NUMERICAL MODEL FOR RANDOM WAVES ON IMPERMEABLE COASTAL STRUCTURES AND BEACHES

by

Andojo Wurjanto

and

Nobuhisa Kobayashi

Sponsored by
U.S. Army Coastal Engineering Research Center

RESEARCH REPORT NO. CACR-91-05

May, 1991

CENTER FOR APPLIED COASTAL RESEARCH

DEPARTMENT OF CIVIL ENGINEERING
UNIVERSITY OF DELAWARE
NEWARK, DELAWARE 19716

ABSTRACT

A computer program called RBREAK is presented in this report. RBREAK is an extended version of our previous computer program IBREAK. The major extension is the development of an automated adjustment procedure to overcome numerical difficulties during the computation of a sufficient duration for random waves. This procedure will allow users to run RBREAK in a more routine manner. This report explains the computational aspects of the computer program RBREAK and a separate computer program called BEFORR developed to implement the automated adjustment procedure.

RBREAK may be used for the design of rough or smooth impermeable coastal structures of arbitrary geometry against normally-incident random waves. Any incident wave train can be specified as input by a user at the seaward boundary of the computation domain. RBREAK computes the reflected wave train at the seaward boundary from which the reflected wave spectrum can be computed. For a subaerial structure, RBREAK computes the waterline oscillation on the seaward slope of the structure or overtopping flow over the crest of the structure if it is not high enough to prevent wave overtopping. For a submerged structure, RBREAK computes the transmitted wave train at the landward boundary of the computation domain from which the transmitted wave spectrum can be computed. In addition to the equations of mass and momentum used to compute the flow field for any structure, an equation of energy is used in RBREAK to estimate the spatial variations of energy dissipation rates due to irregular wave breaking and bottom friction. Furthermore, RBREAK computes the hydraulic stability and sliding motion of individual armor units under the action of the computed flow if the structure is protected with armor units. In addition to coastal structures, RBREAK can be used to predict the irregular wave transformation in the surf zone and the resulting swash oscillation and its spectrum on a beach.

Several examples are presented herein from the published results. These examples may be used to get familiarized with RBREAK. The limitations and capabilities of RBREAK are discussed throughout this report. Users of RBREAK are strongly recommended to read the related papers quoted in the references of this report so that the degree of accuracy of the computed results may be inferred. It is also recommended to calibrate and verify RBREAK if it is to be applied to site-specific problems.

ACKNOWLEDGEMENT

This is the second report resulting from research sponsored by the Coastal Engineering Research Center, U. S. Army Engineer Waterways Experiment Station under contract No. DACW 39-90-K-0006-P001. The authors would like to thank D. L. Ward, J. P. Ahrens, and D. D. Davidson for their enthusiastic support of our continued efforts to improve the capabilities of numerical models for the design of coastal structures.

Contents

Abstract ii

Acknowledgement iii

Part I Introduction

1.1 Background 1

1.2 Previous Work Based on IBREAK 2

1.3 Previous Work Related to RBREAK 4

1.4 Outline of Report 7

Part II Numerical Model

2.1 Governing Equations 8

2.2 Numerical Method 11

2.3 Incident Wave Profile 14

2.3.1 Regular Waves 14
2.3.2 Irregular Waves 16

2.4 Statistical Calculations 17

2.5 Wave Reflection 17

2.6 Wave Run-up 19

2.7 Wave Overtopping 21

2.8 Wave Transmission 23

2.9 Wave Energy Balance 26

2.10 Hydraulic Stability of Armor Units 28

2.11 Movement of Armor Units 29

Part III Computer Program RBREAK

3.1 Introduction 32

3.2 Input Data Files for RBREAK 33

3.3 Main Program RBREAK 35

3.3.1 Reading Input Data 36
3.3.2 Checking FORTRAN PARAMETERs in the Subroutines 36
3.3.3 Groundwork 37
3.3.4 Time-Marching Computation 37
3.3.5 Finishing 38

3.4 Subroutines and Function 39

3.5 Parameters and Variables in COMMON Blocks 44

3.6 Reading the Primary Input Data File 56

3.6.1 Formats 56
3.6.2 Header 56
3.6.3 Input Data 57

3.7 Warning and Error Messages 59

3.8 Output Produced by RBREAK 63

3.8.1 General 63
3.8.2 Subaerial Structures (IJOB<3) 66
3.8.3 Submerged Structures (IJOB=3) 67
3.8.4 Special Request (IREQ=1) 67
3.8.5 Armor Analysis (ISTAB>0) 68
3.8.6 Nodal Time Series (ISAVB=1 and ISAVC=1) 68

PART IV COMPUTER PROGRAM BEFORR

4.1 INTRODUCTION 71

4.2 MAIN PROGRAM BEFORR 71

4.2.1 THREE CRUCIAL COMPUTATION PARAMETERS 72
4.2.2 POTENTIAL NUMERICAL PROBLEMS IN HYDRODYNAMIC COMPUTATION 73
4.2.3 ASSIGNMENT OF VALUES TO NONE AND DELTA 76
4.2.4 COMPUTATION REVERSAL AND STORAGE TIME LEVELS 77

4.3 SUBROUTINES AND FUNCTION 79

4.4 PARAMETERS AND VARIABLES IN COMMON BLOCKS 83

4.5 WARNING AND ERROR MESSAGES 86

4.6 OUTPUT PRODUCED BY BEFORR 89

PART V CONCLUSIONS AND RECOMMENDATIONS 92

REFERENCES 93

APPENDICES

APPENDIX A LISTING OF COMPUTER PROGRAM RBREAK

APPENDIX B LISTING OF COMPUTER PROGRAM BEFORR

APPENDIX C EXAMPLES: REGULAR WAVES

APPENDIX D EXAMPLE: INPUT WAVE TRAIN I FROM EXPERIMENT

APPENDIX E EXAMPLE: INPUT WAVE TRAIN II FROM EXPERIMENT

APPENDIX F EXAMPLE: INPUT WAVE TRAIN FROM TMA SPECTRUM

APPENDIX G EXAMPLE: INPUT WAVE TRAIN FROM USER-SPECIFIED SPECTRUM

APPENDIX H CONTENTS OF THE ACCOMPANYING DISK

APPENDIX I COMPUTATION TIME

List of Figures

1	Wave run-up on a rough slope	9
2	Wave overtopping over a subaerial structure	22
3	Wave trasmission over a submerged stucture	24
4	An overview of the computer programs BEFORR and RBREAK	34
5	Input of dimensional structure geometry for IBOT=1 and IBOT=2	46

List of Tables

1	List of 49 subroutines and one function in computer program RBREAK	40
2	Interdependence among the program units of computer program RBREAK	41
3	List of output files generated by RBREAK	64
4	List of 39 subroutines and one function in computer program BEFORR	80
5	Interdependence among the program units of computer program BEFORR	81
6	List of COMMON blocks in computer program BEFORR	84

PART I
INTRODUCTION

• 1.1 •
BACKGROUND

In the planning and design of rubble mound breakwaters and riprap revetments against incident wind waves, it is generally required to predict irregular wave run-up, overtopping, transmission, and reflection as well as the hydrodynamic stability and movement of armor units (*e.g.*, Shore Protection Manual 1984; Bruun 1985). On the other hand, the quantitative understanding of irregular wave transformation and run-up on beaches is essential for assessing shoreline protection measures against flooding and beach erosion caused by storms (*e.g.*, Kobayashi 1988).

Among the design methods available to predict irregular waves on coastal structures or beaches, numerical models are very new as compared to hydraulic model tests and empirical formulas. Empirical formulas are limited to the specific structure geometry and wave conditions examined in the experiments and are not versatile enough to deal with various combinations of different coastal structures and incident waves. Hydraulic model tests are time-consuming and scale effects such as viscous effects make it difficult to interpret the model test results unless expensive large-scale tests are performed. Furthermore, measurements are normally limited to free surface oscillations and slope profiles because of the difficulties of measuring fluid velocities and hydrodynamic forces acting on armor units. Generation of the specified incident waves is difficult because of wave reflection at the wavemaker and generation of parasitic waves (*e.g.*, Mansard 1988).

On the other hand, numerical models are generally less accurate and versatile than hydraulic model tests, especially for problems involving irregular breaking and broken waves. The advantages of numerical models are low cost, little start-up time, and high spatial and temporal resolutions. During a preliminary design, numerical models may be used together with empirical formulas, if available, to reduce the number of feasible alternatives. During a detailed design, numerical models may be used to reduce the number of hydraulic model tests as well as to estimate the quantities which can not be measured directly. Reversely, the hydraulic model test results may be used to calibrate the empirical coefficients included in numerical models.

Considering the need of numerical models for the problems which may not be modeled using potential flow theory, the authors and co-workers have developed, calibrated, and evaluated numerical models for the last several years. These numerical models are one-dimensional normal to a long coastal structure or shoreline with the assumption of uniformity along the structure or shoreline. The models are time-dependent so as to model the temporal variation of the waterline on the structure or the shoreline on the beach. These models are probably the simplest time-dependent models that can be applied for breaking and broken waves.

The previous work based on the first numerical model called IBREAK and the previous work related to the second numerical model RBREAK are summarized concisely in the following. The papers and reports summarized as the previous work presented mathematical formulations, comparisons with data, and physical interpretations. On the other hand, this report is limited to the computational aspects of our latest numerical model RBREAK. The subroutines used for the spectral and time series analyses for random waves in the previous work are presented in a separate report by Cox, Kobayashi, and Wurjanto (1991).

• 1.2 •
Previous Work Based on IBREAK

Kobayashi and Wurjanto (1989d) presented a computer program called IBREAK. IBREAK may be used for the design of rough or smooth impermeable coastal structures of arbitrary geometry against normally-incident waves. Any incident wave train can be specified as input at the seaward boundary of the computation domain, although the computation was limited to monochromatic and transient waves. IBREAK computes the reflected wave train at the seaward boundary from which the reflection coefficient is predicted. For a subaerial structure, IBREAK computes wave run-up on the seaward slope of the structure or wave overtopping over the crest of the structure if it is not high enough to prevent flow over its crest. For a submerged structure, IBREAK computes the transmitted wave train at the landward boundary of the computation domain from which the transmission coefficient is predicted. In addition to the equations of mass and momentum used to compute the flow field for any structure, an equation of energy is used in IBREAK to estimate the spatial variations of energy dissipation rates due to wave breaking and bottom friction. Furthermore, IBREAK computes the hydraulic stability and sliding motion of individual armor units under the action of the computed flow if the structure is protected with armor units. In addition to coastal structures, IBREAK can be used to predict the wave transformation in the surf zone and the resulting swash oscillation on a beach.

In the following, the previous work based on IBREAK is summarized since the limitations and capabilities of IBREAK and RBREAK are essentially the same except for the capability of RBREAK to overcome numerical difficulties in a self-adjusting manner. Users of RBREAK are recommended to read the related papers quoted in the following since RBREAK has not been applied and calibrated as extensively as IBREAK. However, the monochromatic wave results presented hereafter may not be applicable directly to random waves for which RBREAK is developed.

Kobayashi *et al.* (1986, 1987) developed a numerical flow model to predict the flow characteristics on a rough impermeable slope for specified normally-incident monochromatic waves. The numerical flow model was based on the finite-amplitude shallow-water equations including the effects of bottom friction (Madsen and White 1976) which were solved numerically in the time domain using an explicit dissipative Lax-Wendroff finite difference method (Richtmyer and Morton 1967; Hibberd and Peregrine 1979; Packwood 1980). A recent review of numerical methods developed for flows with shocks was given by Moretti (1987). The adopted numerical method is a shock-capturing method for which a separate treatment of a wave front (shock) is

not required, although it can not describe the detailed behavior of breaking waves (*e.g.*, Peregrine 1983). The numerical flow model was developed in such a way that any incident wave train could be specified at the toe of the slope. The reflected wave train at the toe of the slope was computed from the characteristics advancing seaward. Wave run-up and run-down were predicted from the computed oscillation of the instantaneous waterline on the slope. Comparison was made with available monochromatic wave test data for large-scale uniform riprap slopes (Ahrens 1975; Ahrens and McCartney 1975) and small-scale composite riprap slopes (Kobayashi and Jacobs 1985). The numerical model was shown to predict monochromatic wave run-up, run-down, and reflection fairly well, although the friction factor needs to be calibrated.

Kobayashi and Otta (1987) developed a numerical stability model to predict the hydraulic stability and sliding motion of armor units on a rough impermeable slope under the action of specified normally-incident monochromatic waves. The drag, lift, and inertia forces acting on an armor unit were expressed in terms of the fluid velocity and acceleration predicted separately using the numerical flow model. The numerical stability model predicts the variation of the local stability number along the slope whose minimum value corresponds to the critical stability number for initiation of armor movement. The critical stability number computed for available riprap tests was shown to be in good agreement with the observed zero-damage stability number (Ahrens 1975; Kobayashi and Jacobs 1985), although the lift coefficient used in the model was calibrated within a reasonable range (Sleath 1984)

Kobayashi and Greenwald (1986, 1988) performed an experiment to calibrate and evaluate the developed numerical models in more detail. Eight test runs were conducted in a wave tank using a 1:3 glued gravel slope with an impermeable base. For each run with the specified monochromatic wave train generated in a burst, measurements were made of the free surface oscillation at the toe of the slope, the waterline oscillation on the slope, the temporal variations of dynamic pressure on the base of the slope, and the displacements of loose gravel units placed on the glued gravel slope. The calibrated numerical models were shown to be capable of predicting the measured temporal variations of the free surface and waterline oscillation well. The prediction of the dynamic pressure was reasonable. The spatial variations of the amount of the gravel movement was predicted only qualitatively.

Kobayashi and Watson (1987) applied the developed numerical flow model to predict monochromatic wave reflection and run-up on smooth impermeable slopes by adjusting the friction factor and the water depth specifying visually observed wave run-up. Comparison with available empirical formulas (Seelig 1983; Ahrens and Martin 1985) indicated that the numerical flow model could also predict monochromatic wave reflection and run-up on smooth slopes. Furthermore, the experiment conducted using the 1:3 glued gravel slope was repeated using a 1:3 plywood slope. The adjusted numerical flow model was shown to predict the measured temporal variations of the hydrodynamic quantities on the smooth slope as well.

The numerical flow model developed for coastal structures was also modified slightly to predict the flow characteristics in the swash zone on a beach. The applications of the numerical flow model for predicting the monochromatic or transient wave transformation and swash oscillation on beaches were presented by Kobayashi *et al.* (1988, 1989). On the other hand, the prediction of the sliding motion of individual sand particles was attempted by Kobayashi and DeSilva (1987).

Kobayashi and Wurjanto (1988, 1989a) predicted the monochromatic wave overtopping over the crest of an impermeable coastal structure located on a sloping beach by modifying the

landward and seaward boundary conditions. The modified model accounted for wave shoaling on the sloping beach in front of the structure located in relatively shallow water. The average overtopping rate per unit width was computed from the predicted temporal variations of the velocity and depth of the flow over the crest of the structure. The computed average overtopping rates were shown to be in fair agreement with the small-scale monochromatic wave data of Saville (1955) for which smooth impermeable structures were fronted by a 1:10 slope.

Kobayashi and Wurjanto (1989b, 1989c) and Wurjanto and Kobayashi (1989) predicted the monochromatic wave reflection and transmission over a submerged impermeable breakwater by modifying the landward boundary condition. In addition to the equations of mass and momentum used to compute the flow field, an equation of energy was derived to estimate the rate of energy dissipation due to wave breaking. The computed reflection and transmission coefficients were shown to be in fair agreement with the small-scale monochromatic wave data of Seelig (1980). The numerical model also predicted the spatial variation of the energy dissipation, the mean water level difference, and the time-averaged volume flux per unit width, although available measurements were not sufficient for evaluating the capabilities and limitations of the numerical model for predicting these quantities.

Kobayashi and Wurjanto (1989f) showed that IBREAK could be calibrated and applied to predict the hydrodynamic forces and sliding motions of dolos units at the Crescent City breakwater in California. The calibrated numerical model was used to hindcast the response of the dolos units during a storm which occurred in 1987. The hindcast results were shown to be consistent with the measured results including the upslope movement of poorly interlocked dolos units and the importance of the static and wave forces with negligible impact forces. The numerical model was then used to predict the response of the poorly and well interlocked dolos units under extreme wave conditions. The predicted results have suggested that the wave forces acting on these dolos units may possibly exceed the static forces, while the poorly interlocked dolos units may move considerably, resulting in possible impact forces.

Other researchers (*e.g.* Allsop *et al.* 1988; Thompson 1988; Van der Meer and Breteler 1990) applied the numerical model IBREAK or a similar numerical model to predict the monochromatic wave motion on coastal structures and evaluated the accuracy of the numerical models using their own laboratory measurements. As a whole, their results are consistent with our experiences with IBREAK for the last several years. The numerical model IBREAK is fairly versatile and robust although it is not as accurate as careful hydraulic model tests.

• 1.3 •
Previous Work Related to RBREAK

IBREAK was initially used to simulate irregular waves on the slope of a coastal structure since any incident wave train can be specified as input to IBREAK at the seaward boundary of the computation domain. However, the irregular waterline oscillation on the slope was found to cause numerical difficulties and stoppage during the computation of a sufficient duration for a stationary random sea. The constant time step size Δt for the explicit finite difference method used in IBREAK was reduced to overcome the numerical difficulties. This increased the computation time considerably but did not always work.

To avoid an unnecessary increase in the computation time, it was decided to vary the time step size Δt such that smaller values of Δt should be used for portions of the computation with numerical difficulties. Since the portions with numerical difficulties are not known in advance, the time-marching computation needs to be reversed to an earlier time level before the initiation of the current numerical difficulty and then resumed from the reversed time level using a smaller value of Δt. To reduce the computation time, the value of Δt needs to be increased after overcoming the current numerical difficulty. This adjustment procedure has been improved and automated in the course of work summarized in the following. The computer program RBREAK presented herein may be regarded as an improved version of IBREAK with the automated adjustment procedure which is really essential for making successful computations for incident random waves of sufficient durations in an efficient manner.

Kobayashi, Cox, and Wurjanto (1990) conducted three irregular wave test runs to obtain detailed data on irregular wave reflection and run-up on a 1:3 rough impermeable slope. The test results were also used to evaluate the capabilities and limitations of an earlier version of the computer program RBREAK for predicting the time series and spectral characteristics of the reflected wave and waterline oscillations on the slope. The numerical model was shown to predict the measured time series and spectra reasonably well, including the selective nature of wave reflection and dissipation as well as the appearance of low-frequency components in the waterline oscillations in the 1:3 slope. It is noted that the earlier version of RBREAK required manual interventions to complete the computation for each test run successfully. Two of the three test runs are computed again using the latest version of RBREAK presented in this report. The computed results are presented in Appendices D and E. The computed results for both versions of RBREAK are essentially the same as long as the computation is successful. The automated adjustment procedure of the latest version reduced the computational efforts considerably.

Kobayashi and Wurjanto (1991) derived the one-dimensional equations of mass, momentum, and energy from the two-dimensional continuity and Reynolds equations in order to elucidate the approximations involved in these one-dimensional equations that are used in this report to predict normally-incident wave motions on coastal structures and beaches. The numerical model RBREAK based on these equations was then compared qualitatively with the set-up and swash statistics on a moderately steep beach with a nearshore bar. The numerical model was shown to predict the essential features of the irregular wave transformation and swash oscillation on the barred beach. The computed set-up and swash heights were found to follow the lower bound of the scattered data points partly because of the neglect of low frequency components in the specified incident wave train. A more quantitative comparison was also made with the spectrum of the shoreline oscillation measured on a 1:20 plane beach for which the corresponding wave spectrum was given. RBREAK was shown to predict the dominant low frequency components of the measured spectrum fairly well. Two of the computed runs presented in this paper are included in Appendices F and G in this report.

So far, RBREAK has been applied only to the problems examined in the above two papers. RBREAK could be used to predict irregular wave overtopping and transmission over rough or smooth impermeable structures. Users should calibrate and verify RBREAK if it is to be used to predict irregular wave overtopping and transmission. RBREAK could also be used to predict the hydraulic stability and movement of armor units on rough impermeable structures under the action of irregular waves. The computations of the hydraulic stability of armor units have been made by including a thin permeable underlayer. These related results are summarized in the following to augment the limited calibrations and evaluations made for RBREAK.

Kobayashi and Wurjanto (1990b) developed a numerical model to predict the flow and armor response on a rough permeable slope as well as the flow in a thin permeable underlayer for a normally-incident wave train. In addition to the continuity and momentum equations used to compute the flow field, an equation of energy was used to estimate the rate of energy dissipation due to wave breaking. Computation was made for six test runs to examine the accuracy and capability of the numerical model for simulating the fairly detailed hydrodynamics and armor response under the action of regular waves. The computed critical stability number for initiation of armor movement was compared with the measured stability number corresponding to the start of the damage under irregular wave action to quantify the limitations of the regular wave approximation. The computed wave run-up, run-down, and reflection coefficients were shown to be in qualitative agreement with available empirical formulas based on regular wave tests. Kobayashi and Wurjanto (1989e) applied the developed numerical model to hypothetical permeable slopes corresponding to available impermeable slope tests. The computed results with and without a permeable underlayer indicated that the permeability effects would increase the hydraulic stability of armor units noticeably and decrease wave run-up and reflection slightly. The computed results were qualitatively consistent with available data although they were not extensive and limited to regular waves only.

Kobayashi, Wurjanto, and Cox (1990a) applied the developed numerical model to compute the irregular wave motion on a rough permeable slope. The normally-incident irregular wave train characterized by its spectral density at the toe of the slope was generated numerically for six test runs. The computed critical stability number for initiation of armor movement under the computed irregular wave motion was shown to be in fair agreement with the measured stability number corresponding to the start of the damage. The comparison of the computed armor stability for the incident regular and irregular waves indicated that the armor stability would be reduced appreciably and vary less along the slope under the irregular wave action. On the other hand, the comparison between the computed reflected wave spectrum and the specified incident wave spectrum indicated the reflection of Fourier components with longer periods and the dissipation of Fourier components with shorter periods, while the average reflection coefficient increased with the increase of the surf similarity parameter. The computed waterline oscillations were examined using spectral and time series analyses. The computed spectra of the waterline oscillations showed noticeable low-frequency components, which increased with the decrease of the surf similarity parameter. The statistical analysis of individual wave run-up heights indicated that the computed run-up distribution followed the Rayleigh distribution fairly well for some of the six test runs. The computed maximum wave run-up was in agreement with the empirical formula based on irregular wave run-up tests.

Furthermore, Kobayashi and Wurjanto (1990a) as well as Kobayashi, Wurjanto, and Cox (1990b) analyzed the computed results for the six test runs to examine the critical incident wave profile associated with the minimum rock stability for each run. The minimum rock stability computed for the runs with dominant plunging waves on gentle slopes was caused by the large wave with the maximum crest elevation during its uprush on the slope. The minimum rock stability computed for the runs with dominant surging waves on steeper slopes was caused by the downrushing water with high velocities resulted from a large zero-upcrossing wave with a high crest followed by a deep trough. These computed results may eventually allow one to quantify incident design wave conditions more specifically than the simple approach based on the representative wave height and period. In addition, a simplified model was proposed to predict the eroded area due to the movement and dislodgement of rock units using the probability of

armor movement computed by the numerical model. This model was shown to be in qualitative agreement with the empirical formula for the damage level proposed by Van der Meer (1988).

The numerical model based on the assumption of a thin permeable underlayer was found to be inappropriate for additional three test runs conducted for a 1:3 rough permeable slope with a thick permeable underlayer (Kobayashi, Cox, and Wurjanto 1991). These three test runs corresponded to the three test runs for the 1:3 rough impermeable slope conducted by Kobayashi, Cox, and Wurjanto (1990) except for the presence of the thick permeable underlayer. Efforts are being made to develop a numerical model which is applicable to a rough permeable slope with a thick permeable underlayer. This numerical model will be compared with the permeable slope test results reported by Kobayashi, Cox, and Wurjanto (1991). A user's manual for the numerical model for random waves on permeable coastal structures will be completed in the near future.

• 1.4 •

Outline of Report

This report is written in such a way that users should be able to run RBREAK without reading the report of IBREAK. Part II of this report concisely explains the equations and numerical procedures used in the computer program RBREAK. The computer program RBREAK listed in Appendix A is explained in detail in Part III, although the listed program is written in self-explanatory manners. The additional explanations will facilitate usage of RBREAK and modifications of RBREAK by users if necessary.

Since RBREAK includes a large number of parameters, variables, and subroutines in its present form, a separate computer program called BEFORR is developed to implement the automated adjustment procedure discussed in Section 1.3. The computer program BEFORR listed in Appendix B is explained in Part IV. BEFORR is run before RBREAK to find the time series of appropriate values of the time step size Δt and the small water depth used to define the location of the computational waterline or shoreline. The computed time series of these values are read as input to RBREAK to ensure a successful computation without any numerical problem. This arrangement requires users to run both BEFORR and RBREAK. Our experiences indicate that running both programs should be much easier than running a large number of trial runs to complete the computation of a long duration.

Appendix C presents the computed examples for regular waves which were included in the report of IBREAK. These examples are intended to show that RBREAK can perform all the tasks done by IBREAK with no computational error. Appendices D, E, F, and G present four examples for irregular waves selected from the computed runs given in the papers of Kobayashi, Cox, and Wurjanto (1990) and Kobayashi and Wurjanto (1991). The computed results in these papers which require additional analyses using the subroutines reported by Cox, Kobayashi, and Wurjanto (1991) are not included in these appendices. Appendix H explains the contents of the disk accompanying this report. Appendix I summarizes the computation time for each of the examples included in this report.

PART II
NUMERICAL MODEL

• 2.1 •
GOVERNING EQUATIONS

The wave motion on a rough or smooth impermeable slope is computed for the normally incident wave train specified at the seaward boundary of the computation domain as shown in Fig. 1 for the case of a rough slope. The prime indicates the dimensional variables in the following. The symbols shown in Fig. 1 are as follows:

x' = horizontal coordinate taken to be positive landward with $x' = 0$ at the seaward boundary
z' = vertical coordinate taken to be positive upward with $z' = 0$ at the still water level (SWL)
d'_t = water depth below SWL at the seaward boundary
θ' = local angle of the slope; θ' may vary along the slope
η' = free surface elevation above SWL
h' = water depth above the impermeable slope
u' = depth-averaged horizontal velocity

For the flow over the impermeable slope, the vertically-integrated equations for mass and x'-momentum may be expressed as (Kobayashi, Otta, and Roy 1987)

$$\frac{\partial h'}{\partial t'} + \frac{\partial}{\partial x'}\left(h'u'\right) = 0 \tag{1}$$

$$\frac{\partial}{\partial t'}\left(h'u'\right) + \frac{\partial}{\partial x'}\left(h'u'^2\right) = -gh'\frac{\partial \eta'}{\partial x'} - \frac{1}{2}f'|u'|u' \tag{2}$$

where

t' = time
g = gravitational acceleration
f' = constant friction factor related to the shear stress acting on the slope

In this simplified analysis, the constant friction factor f' accounts for the roughness characteristics of the impermeable slope surface. Moreover, the theoretical bed level for the flow over the rough impermeable slope is difficult to pinpoint as is the case with oscillatory rough turbulent boundary layers (Jonsson 1980).

The following dimensionless variables and parameters are introduced to normalize Eqs. 1 and 2:

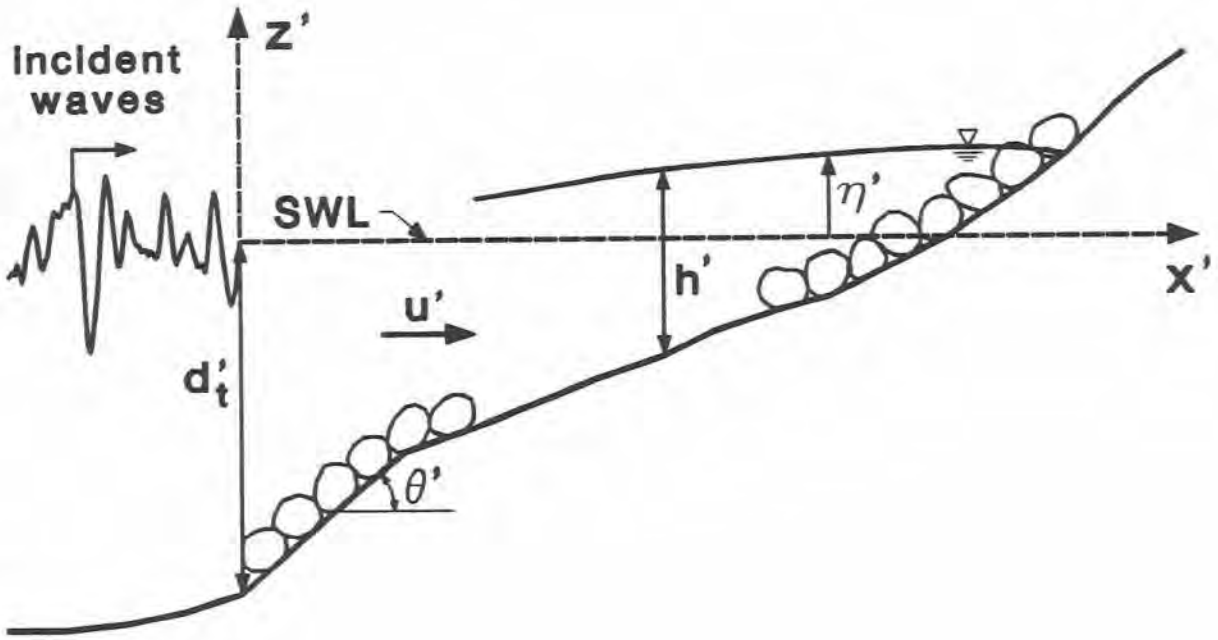


Figure 1. Wave run-up on a rough slope.

$$t = \frac{t'}{T_r'} \quad ; \quad x = \frac{x'}{T_r'\sqrt{gH_r'}} \quad ; \quad u = \frac{u'}{\sqrt{gH_r'}} \tag{3}$$

$$z = \frac{z'}{H_r'} \quad ; \quad h = \frac{h'}{H_r'} \quad ; \quad \eta = \frac{\eta'}{H_r'} \quad ; \quad d_t = \frac{d_t'}{H_r'} \tag{4}$$

$$\sigma = T_r'\sqrt{\frac{g}{H_r'}} \quad ; \quad \theta = \sigma \, tan\theta' \quad ; \quad f = \frac{1}{2}\sigma f' \tag{5}$$

where

T_r' = reference wave period
H_r' = reference wave height
σ = dimensionless parameter related to wave steepness
θ = dimensionless gradient of the slope
f = normalized friction factor

The reference wave period and height used for the normalization can be taken as the period and height used to characterize the incident wave for a particular problem. Substitution of Eqs. 3-5 into Eqs. 1 and 2 yields

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}(hu) = 0 \tag{6}$$

$$\frac{\partial}{\partial t}(hu) + \frac{\partial}{\partial x}\left(hu^2 + \frac{1}{2}h^2\right) = -\theta h - f|u|u \tag{7}$$

where θ and f express the effects of the slope and friction, respectively. For a uniform slope, θ in Eq. 7 can be replaced by the surf similarity parameter, $\xi = \theta/\sqrt{2\pi}$. In terms of the normalized coordinate system, the slope is located at

$$z = \int_0^x \theta \, dx - d_t \quad ; \quad x \geq 0 \tag{8}$$

which reduces to $z = (\theta x - d_t)$ for a uniform slope.

The initial time $t = 0$ for the computation marching forward in time is taken to be the time when the specified incident wave train arrives at the seaward boundary located at $x = 0$ as shown in Fig. 1. The initial conditions for the computation are thus given by $\eta = 0$ and $u = 0$ at $t = 0$ in the region $x \geq 0$. It is noted that h and η are uniquely related for given slope geometry expressed by Eq. 8.

In order to derive appropriate seaward and landward boundary conditions, Eqs. 6 and 7 are expressed in the following characteristic forms:

$$\frac{\partial \alpha}{\partial t} + (u + c)\frac{\partial \alpha}{\partial x} = -\theta - \frac{f|u|u}{h} \quad ; \quad \frac{dx}{dt} = u + c \tag{9}$$

$$\frac{\partial \beta}{\partial t} + (u - c)\frac{\partial \beta}{\partial x} = \theta + \frac{f|u|u}{h} \quad ; \quad \frac{dx}{dt} = u - c \tag{10}$$

with $$c = \sqrt{h} \quad ; \quad \alpha = u + 2c \quad ; \quad \beta = -u + 2c \tag{11}$$

where α and β are the characteristic variables.

Assuming that $u < c$ in the vicinity of the seaward boundary where the normalized water depth below SWL is d_t, α and β represent the characteristics advancing landward and seaward, respectively, in the vicinity of the seaward boundary. The total water depth at the seaward boundary is expressed in the form (Kobayashi, Otta, and Roy 1987)

$$h = d_t + \eta_i(t) + \eta_r(t) \quad \text{at} \quad x = 0 \tag{12}$$

where η_i and η_r are the free surface variations normalized by H_r' at $x = 0$ due to the incident and reflected waves, respectively. The incident wave train is specified by prescribing the variation of η_i with respect to $t \geq 0$. The normalized reflected wave train η_r is approximately expressed in terms of the seaward advancing characteristic β at $x = 0$

$$\eta_r(t) \simeq \frac{1}{2}\sqrt{d_t}\,\beta(t) - d_t - C_t \quad \text{at} \quad x = 0 \tag{13}$$

where β is given by Eq. 10. The correction term C_t in Eq. 13 introduced by Kobayashi, DeSilva, and Watson (1989) to predict wave set-down and set-up on a beach may be expressed as

$$C_t = \frac{1}{2}\sqrt{d_t}\,\frac{\overline{(\eta - \overline{\eta})(u - \overline{u})}}{\overline{h}} \quad \text{at} \quad x = 0 \tag{14}$$

The landward boundary condition of the numerical model depends on the crest height of a structure as will be explained in relation to the numerical procedures for wave run-up, overtopping, and transmission.

• 2.2 •
Numerical Method

Eqs. 6 and 7 are combined and expressed in the following vector form:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \mathbf{G} = 0 \tag{15}$$

with $$\mathbf{U} = \begin{bmatrix} m \\ h \end{bmatrix} \quad ; \quad \mathbf{F} = \begin{bmatrix} mu + \frac{1}{2}h^2 \\ m \end{bmatrix} \quad ; \quad \mathbf{G} = \begin{bmatrix} \theta h + f|u|u \\ 0 \end{bmatrix} \tag{16}$$

where $m = uh$ is the normalized volume flux per unit width. The vectors $\mathbf{F}$ and $\mathbf{G}$ depend on the vector $\mathbf{U}$ for given θ and f.

Eq. 15 is discretized using a finite difference grid of space size Δx and time step Δt based on an explicit dissipative Lax-Wendroff method (*e.g.*, Richtmyer and Morton 1967). The space size Δx is constant during the computation duration, whereas the time step Δt may vary in the following fashion:

The computation duration is divided into *computation units*. Each computation unit is one wave period long except the last unit, the length of which may be less than one wave period. The *wave period* used as reference is the reference wave period, T_r'. The time step Δt is held constant within a computation unit, but may vary from one computation unit to another.

In the following, the known quantities at the node located at $x = (j-1)\Delta x$, with $j = 1, 2, ..., s$, and at the present time t are indicated by subscript $_j$ without superscript. The integer s indicates the wet node next to the moving waterline at the time t for the case of wave run-up, and the node at the specified landward boundary for the case of wave overtopping or transmission. The unknown quantities at the node j and at the next time level $t^* = (t + \Delta t)$ are denoted by subscript $_j$ and an asterisk. The asterisk indicates the quantities at the time t^*. The values of $\mathbf{U}_1^*$ and $\mathbf{U}_j^*$ for $j \geq (s-1)$ are computed using the seaward and landward boundary conditions, respectively. The values of $\mathbf{U}_j^*$ for $j = 2, 3, ..., (s-2)$ are computed using the known values of $\mathbf{U}_{j-1}$, $\mathbf{U}_j$, and $\mathbf{U}_{j+1}$ at the time t (Kobayashi, Otta, and Roy 1987)

$$\begin{aligned} \mathbf{U}_j^* &= \mathbf{U}_j - \lambda \left[\frac{1}{2} (\mathbf{F}_{j+1} - \mathbf{F}_{j-1}) + \Delta x \, \mathbf{G}_j \right] \\ &\quad + \frac{\lambda^2}{2} (\mathbf{g}_j - \mathbf{g}_{j-1} - \Delta x \, \mathbf{S}_j) + \mathbf{D}_j \end{aligned} \tag{17}$$

where $\lambda = \Delta t / \Delta x$. The vector $\mathbf{g}_j$ in Eq. 17 is given by

$$\mathbf{g}_j = \frac{1}{2} [\mathbf{A}_{j+1} + \mathbf{A}_j] \left[(\mathbf{F}_{j+1} - \mathbf{F}_j) + \frac{\Delta x}{2} (\mathbf{G}_{j+1} + \mathbf{G}_j) \right] \tag{18}$$

with

$$\mathbf{A}_j = \begin{bmatrix} 2u_j & (h_j - u_j^2) \\ 1 & 0 \end{bmatrix} \tag{19}$$

The vector $\mathbf{S}_j$ in Eq. 17 is defined as

$$\mathbf{S}_j = \begin{bmatrix} \Delta x \, e_j - \frac{1}{2} \theta_j (m_{j+1} - m_{j-1}) \\ 0 \end{bmatrix} \tag{20}$$

with

$$\begin{aligned} e_j &= 2f \frac{|u_j|}{h_j} \left[\left(u_j^2 - h_j \right) \frac{h_{j+1} - h_{j-1}}{2\Delta x} \right. \\ &\quad \left. - u_j \frac{m_{j+1} - m_{j-1}}{2\Delta x} - \theta_j h_j - f |u_j| u_j \right] \end{aligned} \tag{21}$$

The vector $\mathbf{D}_j$ in Eq. 17 represents the additional term for damping high frequency parasitic waves, which tend to appear at the rear of a breaking wave, and is given by

$$\mathbf{D}_j = \frac{\lambda}{2} [\mathbf{Q}_j (\mathbf{U}_{j+1} - \mathbf{U}_j) - \mathbf{Q}_{j-1} (\mathbf{U}_j - \mathbf{U}_{j-1})] \tag{22}$$

with
$$\mathbf{Q}_j = p_j\,\mathbf{I} + \frac{1}{2} q_j\,(\mathbf{A}_j + \mathbf{A}_{j+1}) \tag{23}$$

where $\mathbf{I}$ = unit matrix; and the coefficients p_j and q_j are given by

$$\begin{aligned} p_j &= -\frac{1}{2\,(c_{j+1}+c_j)}\,[\;\epsilon_1 |v_{j+1}-v_j|\,(w_{j+1}+w_j) \\ &\quad -\epsilon_2 |w_{j+1}-w_j|\,(v_{j+1}+v_j)\;] \end{aligned} \tag{24}$$

$$q_j = \frac{\epsilon_1 |v_{j+1}-v_j| - \epsilon_2 |w_{j+1}-w_j|}{(c_{j+1}+c_j)} \tag{25}$$

with
$$c = \sqrt{h} \quad ; \quad v = u + c \quad ; \quad w = u - c \tag{26}$$

where ϵ_1 and ϵ_2 are the positive damping coefficients determining the amount of numerical damping of high frequency parasitic waves at the rear of a breaking wave.

The numerical stability criterion for this explicit finite difference method is given by (Packwood 1980)

$$\frac{\Delta t}{\Delta x} < \frac{\left[\sqrt{\left(1+\frac{1}{4}\epsilon^2\right)} - \frac{1}{2}\epsilon\right]}{|u_m| + c_m} \tag{27}$$

which requires that $|u_j| < (\Delta x/\Delta t)$ at any time for any j, where

u_j = normalized horizontal fluid velocity at the node j
u_m = maximum value of u expected to be encountered in the flow field
c_m = maximum expected value of $\sqrt{h}$
ϵ = the greater of the positive coefficients ϵ_1 and ϵ_2

The values of Δx, ϵ_1, and ϵ_2 need to be specified by a user. The value of Δt will be assigned automatically by the computer program BEFORR, which should be executed before the computer program RBREAK. The assigned value of Δt is based on the above stability criterion, numerical problems encountered, and convenience for storing computed time series at a specified time interval.

To facilitate the use of Eq. 27, the numerical stability indicator Ω is introduced.

$$\Omega = \frac{\Delta x}{\Delta t}\,\frac{\left[\sqrt{\left(1+\frac{1}{4}\epsilon^2\right)} - \frac{1}{2}\epsilon\right]}{1+\sqrt{d_t}} \tag{28}$$

where the normalized depth at the seaward boundary, d_t, has been defined in Eq. 4. If the following approximations are made

$$\begin{aligned} u_m &\simeq 1 \\ c_m &\simeq \sqrt{d_t} \end{aligned}$$

the numerical stability criterion given by Eq. 27 can then be expressed as

$$\Omega > 1 \tag{29}$$

In the computer program RBREAK, Eq. 29 is actually used in lieu of Eq. 27 to determine whether the numerical stability criterion is violated or not. Detailed explanations for the numerical stability will be presented in Part IV.

• 2.3 •
INCIDENT WAVE PROFILE

One of the required input for the computer program RBREAK is the normalized incident wave profile at the seaward boundary of the computation domain, $\eta_i(t) = \eta_i'(t')/H_r'$, with $t = t'/T_r'$ as defined in Eq. 3, where H_r' and T_r' are the reference wave height and period used for the normalization in Eqs. 3-5. The specification of $\eta_i(t)$ for $t \geq 0$ needs to satisfy the condition $\eta_i = 0$ at $t = 0$ to be consistent with the assumed initial conditions of no wave action in the region of $x \geq 0$ at $t = 0$. The temporal variation of $\eta_i(t)$ can be

1. the profile of monochromatic waves computed (by the computer program RBREAK) using an appropriate wave theory, or
2. a user-specified irregular or transient wave train which may be based on measurements or numerical generation.

For convenience, the former will be referred to as the case of *regular* waves, while the latter is simply called the case of *irregular* waves, although this case can also include monochromatic transient waves.

2.3.1 REGULAR WAVES

For the case of incident regular waves, the periodic variation of $\eta_i(t)$ is computed by the computer program RBREAK using either cnoidal or Stokes second-order wave theory. The height and period of the incident regular waves at the seaward boundary located at $x = 0$ are denoted by H' and T'. The reference wave period T_r' is taken as $T_r' = T'$ for the incident regular waves. The reference wave height H_r' specified as input may be referred to deep water (Kobayashi and Wurjanto 1988, 1989a) or the location of wave measurement (Kobayashi, Strzelecki, and Wurjanto 1988). Since the numerical model is based on the finite-amplitude shallow-water equations given by Eqs. 1 and 2, the seaward boundary should be located in relatively shallow water. As a result, it is not always possible to take $H_r' = H'$. Defining $K_s = H'/H_r'$, the height and period of the regular wave profile $\eta_i(t)$ at $x = 0$ is K_s and unity, respectively.

For Stokes second-order wave theory (*e.g.*, SPM 1984), the incident wave profile $\eta_i(t)$ at $x = 0$ is given by

$$\eta_i(t) = K_s \left\{ \frac{1}{2} cos\left[2\pi(t + t_0)\right] + a_2 cos\left[4\pi(t + t_0)\right] \right\} \quad \text{for } t \geq 0 \tag{30}$$

with

$$a_2 = \frac{2\pi}{L} cosh\left(\frac{2\pi}{L}\right) \left[2 + cosh\left(\frac{4\pi}{L}\right)\right] \left[16 \frac{d_t}{K_s} sinh^3\left(\frac{2\pi}{L}\right)\right]^{-1} \tag{31}$$

$$L = L_0 \, tanh \frac{2\pi}{L} \tag{32}$$

where

t_0 = time shift computed to satisfy the conditions that $\eta_i = 0$ at $t = 0$ and η_i decreases initially
a_2 = normalized amplitude of the second-order harmonic
$L = L'/d'_t$
$L_0 = L'_0/d'_t$
L' = dimensional wavelength at $x = 0$
L'_0 = dimensional linear wavelength in deep water

The normalized wavelength L satisfying Eq. 32 for given L_0 is computed using a Newton-Raphson iteration method. Eq. 31 yields the value of a_2 for given $d_t = d'_t/H'_r$, K_s, and L. Since Eq. 30 satisfies $\eta_i(t + 1) = \eta_i(t)$ and $\eta_i(-t - t_0) = \eta_i(t + t_0)$, it is sufficient to compute the profile $\eta_i(t)$ for $0 \leq (t + t_0) \leq 0.5$. Eq. 30 may be appropriate if the Ursell parameter $U_r < 26$ where $U_r = [H'(L')^2/(d'_t)^3] = (K_s L^2/d_t)$ at $x = 0$. It is noted that the value of U_r based on the normalized wavelength L computed from Eq. 32 is simply used to decide whether cnoidal or Stokes second-order wave theory is applied.

For the case of $U_r \geq 26$, cnoidal wave theory (*e.g.*, Svendsen and Brink-Kjær 1972) is used to compute the incident wave profile $\eta_i(t)$ at $x = 0$

$$\eta_i(t) = \eta_{min} + K_s \, cn^2\left[2K(t + t_0)\right] \quad \text{for } t \geq 0 \tag{33}$$

with

$$\eta_{min} = \frac{K_s}{m}\left(1 - \frac{E}{K}\right) - K_s \tag{34}$$

where

η_{min} = normalized trough elevation below SWL
cn = Jacobian elliptic function
K = complete elliptic integral of the first kind
E = complete elliptic integral of the second kind
m = parameter determining the complete elliptic integrals $K(m)$ and $E(m)$

The parameter m is related to the Ursell parameter U_r

$$U_r = \frac{K_s L^2}{d_t} = \frac{16}{3} m K^2 \tag{35}$$

For $U_r \geq 26$, the parameter m is in the range $0.8 < m < 1$. The parameter m for given σ, d_t, and K_s is computed from

$$\frac{\sigma}{L\sqrt{d_t}} \sqrt{\left[1 + \frac{K_s}{m\, d_t}\left(-m + 2 - 3\frac{E}{K}\right)\right]} - 1 = 0 \tag{36}$$

where the normalized wavelength L is given by Eq. 35 as a function of m for given d_t and K_s. The left hand side of Eq. 36 is a reasonably simple function of m in the range $0.8 < m < 1$. As a result, Eq. 36 can be solved using an iteration method which successively narrows down the range of m bracketing the root of Eq. 36. After the value of m is computed for given σ, d_t, and K_s, the values of U_r and L are computed using Eq. 35, while Eq. 34 yields the value of η_{min}. The incident wave profile $\eta_i(t)$ is computed using Eq. 33 for $0 \leq (t + t_0) \leq 0.5$ where the time shift t_0 and the periodicity and symmetry of the cnoidal wave profile are used in the same manner as the Stokes second-order wave profile given by Eq. 30. It should be mentioned that the Jacobian elliptic function and the complete elliptic integrals of the first and second kinds are computed using the subroutines given by Press *et al.* (1986).

2.3.2 IRREGULAR WAVES

The computer program RBREAK is an improvement over its predecessor, IBREAK (Kobayashi and Wurjanto 1989d), in the sense that RBREAK has the capability of overcoming numerical difficulties for the cases where the wave train of a long duration at the seaward boundary is specified by a user. The specified waves can be *incident* or *total*. The *total* wave is defined as the sum of *incident* and *reflected* waves. It should be stated that the specification of the total wave at the seaward boundary has been found to produce spurious long-period oscillation in the computation domain (Packwood 1980). To avoid confusion, the user-specified wave train at the seaward boundary will hereafter be referred to as *input wave train*. IBREAK has been used only for the wave train of a relatively short duration at the seaward boundary specified by a user (*e.g.*, Kobayashi and Greenwald 1986, 1988). The computer program RBREAK, on the other hand, has been developed for user-specified wave trains of *long* durations. All the options of IBREAK are included in RBREAK, although some of them have not been used for irregular waves. Appendix C shows how to prepare input data for the regular wave computations given as examples in the IBREAK report (Kobayashi and Wurjanto 1989d).

The generation of user-specified wave trains is explained in the separate report by Cox, Kobayashi, and Wurjanto (1991). However, the examples in Appendices D through G show how various kinds of user-specified wave trains may be used in RBREAK. All of these examples feature *random* waves, for which the computer program RBREAK is primarily developed.

Caution should be used to specify the heights and periods of random waves. This numerical model distinguishes the reference wave (H'_r, T'_r) from the representative wave at the seaward boundary (H', T'). The reference wave (H'_r, T'_r) is used to normalize the governing equations (Eqs. 1 and 2), whereas (H', T') is a representative wave of random waves being investigated at the seaward boundary. The representative wave height H' can be, for example, the incident significant wave height (Kobayashi, Cox, and Wurjanto 1990) or the spectral estimate of the incident significant wave height (Kobayashi and Wurjanto 1991). The representative wave period T' can be the mean wave period (Kobayashi, Cox, and Wurjanto 1990) or the spectral peak period (Kobayashi and Wurjanto 1991). Since the relation between the two waves is not simple for the case of random waves, it is recommended that the representative wave at the seaward

boundary (H', T') be taken as the reference wave (H'_r, T'_r), as is the case with the examples given in Appendices D trough G. Thus, $H' = H'_r$, $T' = T'_r$, and $K_s = H'/H'_r = 1$. $K_s = 1$ implies that the normalized height of the representative wave at the seaward boundary is unity.

• 2.4 •
Statistical Calculations

The *statistical calculations* in this report imply the calculations of the time-averaged (mean) value and the extreme (maximum and minimum) values of computed time-series. The statistical calculations are performed over the specified time interval described below.

1. For the case of *regular* waves, the statistical calculations are executed over the last wave period, assuming that the duration of the computation is long enough to reach the periodicity of time-varying quantities. This duration ranges from approximately five wave periods for coastal structures (Kobayashi and Wurjanto 1989d) to about 30 wave periods for beaches (Kobayashi, DeSilva, and Watson 1989).

2. For *irregular* wave computations, the statistical calculations are conducted over most or all of the computation duration. The initial transient waves may be excluded for the case where the *random* sea state is modeled using a spectral approach (Kobayashi and Wurjanto 1991). On the other hand, if measured time series are available, no exclusion may be necessary (Kobayashi, Cox, and Wurjanto 1990).

In the subsequent sections, the time-averaged (mean) value will be denoted by an overbar, whereas the maximum and minimum values are indicated by the subscripts $_{max}$ and $_{min}$. It should be kept in mind that these values are associated with the time interval described above.

• 2.5 •
Wave Reflection

The normalized reflected wave train $\eta_r(t)$ at the seaward boundary is computed using Eq. 13. It is also required to find the unknown value of the vector $\mathbf{U}_1^*$ at $x = 0$ and the next time level $t^* = (t + \Delta t)$ which can not be computed using Eq. 17.

A simple first-order finite difference equation corresponding to Eq. 10 with $f = 0$ is used to find the value of β_1^* at $x = 0$ and the time t^*.

$$\beta_1^* = \beta_1 - \frac{\Delta t}{\Delta x}(u_1 - c_1)(\beta_2 - \beta_1) + \Delta t\, \theta_1 \tag{37}$$

where $\beta_1 = (-u_1 + 2c_1)$ and $\beta_2 = (-u_2 + 2c_2)$. The right hand side of Eq. 37 can be computed for the known values of $\mathbf{U}_j$ with $j = 1$ and 2 at the present time t where the spatial nodes are

located at $x = (j-1)\Delta x$. The value of η_r at the time t^* is calculated using Eq. 13. Eq. 12 yields the value of h_1^*, while $u_1^* = [2\sqrt{(h_1^*)} - \beta_1^*]$ using the definition of β given in Eq. 11. Thus, the values of h_1^*, u_1^*, and $m_1^* = u_1^* h_1^*$ at $x = 0$ and the time t^* are obtained.

The nonlinear correction term C_t given by Eq. 14 needs to be estimated to compute $\eta_r(t)$ using Eq. 13. For incident regular waves on gentle slopes, C_t may be estimated by (Kobayashi, DeSilva, and Watson 1989)

$$C_t = \frac{K_s^2}{16 d_t} \quad \text{for gentle slopes} \tag{38}$$

where the assumptions of linear long wave and negligible wave reflection were made in Eq. 14 to derive Eq. 38. Wave reflection may not be negligible for coastal structures. It is hence suggested to choose the location of the seaward boundary so that C_t may be assumed to be $C_t \simeq 0$ for coastal structures. This assumption may be checked using Eq. 38 as a rough guideline.

The reflection coefficient may be estimated using the following equations (Kobayashi and Wurjanto 1989b, 1989c)

$$r_1 = \frac{1}{K_s}[(\eta_r)_{max} - (\eta_r)_{min}] \tag{39}$$

$$r_2 = \sqrt{\frac{\overline{\eta_r^2}}{\overline{\eta_i^2}}} \tag{40}$$

$$r_3 = \sqrt{\frac{\overline{(\eta_r - \overline{\eta_r})^2}}{\overline{\eta_i^2}}} \tag{41}$$

where the overbar indicates time averaging and the subscripts $_{max}$ and $_{min}$ indicate the maximum and minimum values of $\eta_r(t)$ over the time interval of the statistical calculations (Section 2.4).

K_s represents the normalized height of the representative wave at the seaward boundary. The numerator $[(\eta_r)_{max} - (\eta_r)_{min}]$ in Eq. 39 is the normalized vertical distance between the highest and the lowest surface elevations associated with the reflected wave train. For regular waves, this is the height of the reflected waves. The reflection coefficient r_1 for the case of regular waves is the ratio of the height of the reflected wave to that of the incident wave, For random waves, however, this quantity indicates only the magnitude of the reflected waves and overestimates the degree of wave reflection.

Eqs. 40 and 41 are based on the time-averaged reflected wave energy as compared to the time-averaged incident wave energy, $\overline{\eta_i^2}$. For the case of regular waves, the energy $\overline{\eta_i^2}$ may be estimated using linear wave theory as $\overline{\eta_i^2} = K_s^2/8$. Eq. 41 accounts for the difference $\overline{\eta_r}$ between the still water level and the mean water level at $x = 0$ where $\eta_i(t)$ is specified such that $\overline{\eta_i} = 0$. The method used to compute the reflection coefficient should be consistent with the method used to estimate the reflection coefficient from measured free surface oscillations. If the temporal variations of $\eta_r(t)$ and $\eta_i(t)$ are sinusoidal, Eqs. 39 and 40 yield $r_1 = r_2$. Eqs. 40 and 41 are applicable to regular waves as well as irregular waves. The computer program RBREAK computes all the three reflection coefficients for the case of regular waves, but provides only r_1 for the case of irregular waves since the time series of the incident and reflected waves, from

which r_2 and r_3 can be computed, are available as standard output of the program. Furthermore, these time series allow one to compute the reflection coefficient as a function of the frequency for random waves (*e.g.*, Kobayashi, Cox, and Wurjanto 1990).

• 2.6 •

WAVE RUN-UP

For the case of no wave overtopping on a subaerial coastal structure as shown in Fig. 1, the landward boundary of the numerical model is located at the moving waterline on the slope where the water depth is essentially zero. The kinematic boundary condition requires that the horizontal waterline velocity be the same as the horizontal fluid velocity. In reality, it is difficult to pinpoint the exact location of the moving waterline on the slope. For the computation, the waterline is defined as the location where the normalized instantaneous water depth equals a small value δ, such as $\delta = 10^{-3}$.

The following numerical procedure dealing with the moving waterline located at $h = \delta$ is used to compute the values of $\mathbf{U}_j^*$ at the next time level $t^* = (t + \Delta t)$ for the nodes $j \geq (s-1)$ which are not computed by Eq. 17. It is noted that the procedure is somewhat intuitive and may be improved since the moving waterline tends to cause numerical instability.

1. Compute $h_{s+1} = (2h_s - h_{s-1})$, $u_{s+1} = (2u_s - u_{s-1})$, and $m_{s+1} = h_{s+1}u_{s+1}$ at the present time t where the integer s indicates the wet node next to the moving waterline at the time t such that $h_{s+1} \leq \delta < h_s$.

2. Compute h_j^* and m_j^* at the time t^* for the nodes $j = (s-1)$ and $j = s$, using Eq. 17 without the damping term $\mathbf{D}_j$ since the water depth h can be very small at these nodes.

3. Check whether $h_{s-1}^* \leq \delta$, which may be encountered during a downrush. This is considered a computation failure since the waterline should not move more than Δx because of the numerical stability criterion of the adopted explicit method given by Eq. 27. Denoted by `IPROB=2`, this failure is one of three identified numerical difficulties, which will be elaborated more in Part IV.

4. If $h_s^* > h_{s-1}^*$, use $h_s^* = (2h_{s-1}^* - h_{s-2}^*)$, and $u_s^* = (2u_{s-1}^* - u_{s-2}^*)$, so that the water depth near the waterline decreases landward. The following adjustments are made

   - if $|u_s^*| > |u_{s-1}^*|$, set $u_s^* = 0.9u_{s-1}^*$;
   - if $h_s^* < 0$, set $h_s^* = 0.5h_{s-1}^*$;
   - and if $h_s^* > h_{s-1}^*$, set $h_s^* = 0.9h_{s-1}^*$.

   Then, obtain $m_s^* = h_s^*u_s^*$ based on the *adjusted* values of h_s^* and u_s^*.

5. If $h_s^* \leq \delta$, set $s^* = (s-1)$ and go to Step 13. The integer s^* indicates the wet node next to the waterline at the next time level t^*.

6. If $h_s^* > \delta$, compute $h_{s+1}^* = (2h_s^* - h_{s-1}^*)$, $u_{s+1}^* = (2u_s^* - u_{s-1}^*)$, and $m_{s+1}^* = h_{s+1}^*u_{s+1}^*$.

7. If $h^*_{s+1} \leq \delta$, set $s^* = s$ and go to Step 13.

8. If $h^*_{s+1} > \delta$, compute $\mathbf{U}^{**}_s$ at the time $t^{**} = (t^* + \Delta t)$ using Eq. 17 without the damping term where $\mathbf{U}^*_j$ and $\mathbf{U}_j$ in Eq. 17 are replaced by $\mathbf{U}^{**}_s$ and $\mathbf{U}^*_s$, respectively. Improve the linearly extrapolated values in Step 6 using the following finite difference equations derived from Eqs. 6 and 7 with $f = 0$

$$m^*_{s+1} = m^*_{s-1} - \frac{\Delta x}{\Delta t}(h^{**}_s - h_s) \tag{42}$$

$$u^*_{s+1} = u^*_{s-1} - \frac{1}{u^*_s}\left[\frac{\Delta x}{\Delta t}(u^{**}_s - u_s) + h^*_{s+1} - h^*_{s-1} + 2\Delta x\, \theta_s\right] \tag{43}$$

   The upper limit of the absolute value of $(u^*_s)^{-1}$ in Eq. 43 is taken as δ^{-1} to avoid the division by the very small value. Calculate $h^*_{s+1} = m^*_{s+1}/u^*_{s+1}$.

9. If $|u^*_{s+1}| \leq \delta$, set $s^* = s$ and go to Step 13.

10. If $h^*_{s+1} \leq h^*_s$ and $h^*_{s+1} \leq \delta$, set $s^* = s$ and go to Step 13.

11. If $h^*_{s+1} \leq h^*_s$ and $h^*_{s+1} > \delta$, set $s^* = (s+1)$ and go to Step 13.

12. If $h^*_{s+1} > h^*_s$, the linearly extrapolated values of h^*_{s+1}, u^*_{s+1}, and m^*_{s+1} in Step 6 are adopted in the following instead of those computed in Step 8. Furthermore, set $s^* = s$ if $h^*_{s+1} > h^*_s$ and $s^* = (s+1)$ if $h^*_{s+1} \leq h^*_s$ where h^*_{s+1} is the adopted value given by $h^*_{s+1} = (2h^*_s - h^*_{s-1})$.

13. After $s*$ is obtained, set $h^*_j = 0$, $u^*_j = 0$, and $m^*_j = 0$ for $j \geq (s^* + 1)$ since no water is present above the computational waterline.

Once the normalized water depth h at the given time is known as a function of x, the normalized free surface elevation, $Z_r = Z'_r/H'_r$, where the physical water depth equals a specified value δ'_r, can be computed as long as $\delta_r = (\delta'_r/H'_r) > \delta$. The use of the physical depth δ'_r is related to the use of a waterline meter to measure the waterline oscillation on the slope (*e.g.*, Kobayashi and Greenwald 1986, 1988). The specified depth δ'_r can be regarded as the vertical distance between the waterline meter and the slope, while the corresponding elevation Z'_r is the elevation above SWL of the intersection between the waterline meter and the free surface. The computed oscillations of $Z_r(t)$ for different values of δ'_r can be used to examine the sensitivity to δ'_r of wave run-up and run-down, which are normally defined as the maximum and minimum elevations relative to SWL reached by uprushing and downrushing water on the slope, respectively. The normalized run-up R, run-down R_d, and set-up $\overline{Z_r}$ for given δ'_r are obtained from the computed oscillation of $Z_r(t)$. The computed results such as those presented by Kobayashi, DeSilva, and Watson (1989) indicate that wave run-up is insensitive to δ'_r but wave run-down is very sensitive to δ'_r since a thin layer of water remains on the slope during wave downrush. This implies that wave run-down is difficult to measure visually or using a waterline meter.

• 2.7 •

WAVE OVERTOPPING

Wave overtopping will occur if uprushing water reaches the landward edge of the crest of a subaerial structure as shown in Fig. 2 where $x'_e = x'$-coordinate of the landward edge of the crest. If wave overtopping occurs, the computation domain for the numerical model is limited to the region $0 \leq x \leq x_e$ where $x_e = x'_e/(T'_r\sqrt{gH'_r})$ and the dimensionless variables defined in Eqs. 3-5 remain the same. For the computation, wave overtopping is assumed to occur when the computed water depth h at $x = x_e$ becomes greater than the small value δ, used as the normalized depth of the computational waterline on the slope. It is assumed that water flows over the landward edge freely. The flow approaching the landward edge can be supercritical as well as subcritical since the water depth at $x = x_e$ is relatively small.

For the computation starting from the initial conditions of no wave action in the computation domain, wave overtopping occurs when the integer s indicating the wet node next to the computational waterline at the present time t becomes equal to the given integer j_e indicating the most landward node for the case of wave overtopping. When $s = j_e$ at the time t, the values of $\mathbf{U}_j^*$ at the next time level $t^* = (t + \Delta t)$ for the nodes $j = (s-1)$ and s are computed as follows

1. Compute $\mathbf{U}_{s-1}^*$ using Eq. 17 with $j = (s-1)$ without the damping term $\mathbf{D}_{s-1}$ since the water depth h can be very small at this node.

2. If $u_{s-1} > c_{s-1}$, where $c_{s-1} = \sqrt{h_{s-1}}$, the flow approaching the landward edge is supercritical, and both characteristics given by Eqs. 9 and 10 advance to the landward edge from the computation domain. Since Eqs. 9 and 10 are equivalent to Eqs. 6 and 7, use is made of the following finite difference equations derived from Eqs. 6 and 7 with $f = 0$

$$h_s^* = h_s - \frac{\Delta t}{\Delta x}(m_s - m_{s-1}) \tag{44}$$

$$m_s^* = m_s - \frac{\Delta t}{\Delta x}\left[\left(m_s u_s + \frac{1}{2}h_s^2\right) - \left(m_{s-1}u_{s-1} + \frac{1}{2}h_{s-1}^2\right)\right] - \Delta t\,\theta_s\,h_s \tag{45}$$

   Then, $u_s^* = m_s^*/h_s^*$, and $\mathbf{U}_s^*$ is obtained.

3. If $u_{s-1} \leq c_{s-1}$, the flow approaching the landward edge is subcritical or critical, and only the characteristics α given by Eq. 9 advances to the landward edge from the computation domain. For this case, the flow at the landward edge node is assumed to be critical, that is, $u_s^* = c_s^*$ at the time t^*. Use is made of the following finite difference equation derived from Eq. 9 with $f = 0$

$$\alpha_s^* = \alpha_s - \frac{\Delta t}{\Delta x}(u_s + c_s)(\alpha_s - \alpha_{s-1}) - \Delta t\,\theta_s \tag{46}$$

   where $\alpha_s = (u_s + 2c_s)$ and $\alpha_{s-1} = (u_{s-1} + 2c_{s-1})$. Since $u_s^* = c_s^* = \sqrt{h_s^*}$, then $\alpha_s^* = (u_s^* + 2c_s^*) = 3u_s^*$. Thus, $u_s^* = \alpha_s^*/3$, $h_s^* = (u_s^*)^2$, $m_s^* = u_s^* h_s^*$, and $\mathbf{U}_s^*$ is obtained.

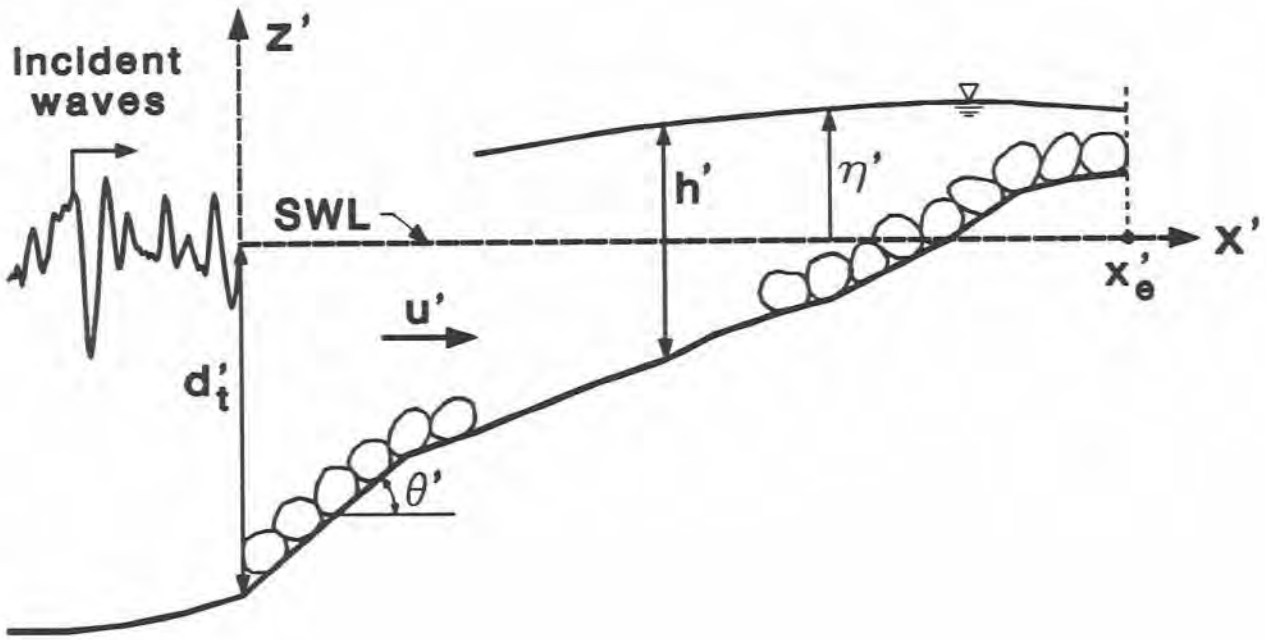


Figure 2. Wave overtopping over a subaerial structure.

4. If $h_s^* \leq \delta$, wave overtopping is assumed to cease. The integer s^* is set to be equal to $(s-1)$ and the landward boundary computation switches to the run-up mode discussed in Section 2.6. The integer s^* indicates the wet node next to the waterline at the time t^*.

5. If $h_s^* > \delta$, wave overtopping continues and $s^* = j_e$.

For incident regular waves, the normalized average overtopping rate per unit width, Q, is obtained from the computed temporal variation of $m = uh$ at $x = x_e$.

$$Q = \frac{Q'}{H_r' \sqrt{g H_r'}} = \overline{m} \quad \text{at } x = x_e \tag{47}$$

where the overbar indicates the time averaging of $m(t,x)$ at $x = x_e$. Eq. 47 does not include the volume flux during the interval when $h \leq \delta$ at $x = x_e$ since the values of m at the nodes landward of the computational waterline are set to be zero during the computation. Eq. 47 is applicable for both regular and irregular waves. The time interval over which the time averaging is performed has been described in Section 2.4.

• 2.8 •
Wave Transmission

For wave transmission over a submerged breakwater, the landward boundary is always located at $x' = x_e'$ as shown in Fig. 3 where x_e' = x'-coordinate of the landward boundary which can be taken to be any convenient location such as the landward toe of the submerged breakwater. The computation domain for the numerical model is the fixed region $0 \leq x \leq x_e$ where $x_e = x_e'/(T_r' \sqrt{g H_r'})$ and the dimensionless variables defined in Eqs. 3-5 remain the same. To avoid the appearance of the waterline in the region $0 \leq x \leq x_e$, the normalized water depth h in the computation domain is taken as $h = \delta$ if the computed value of h becomes less than δ. It is assumed that the transmitted waves propagate landward without being reflected from the shoreline, and the transmitted water flows landward without a return current. If the effects of the shoreline and return current need to be included, it will be required to extend the computation domain to the shoreline in a manner similar to the computations made by Kobayashi, Strzelecki, and Wurjanto (1988) and Kobayashi, DeSilva, and Watson (1989) for the regular wave transformation over a shore-parallel bar and resulting swash oscillation on a beach, as well as Kobayashi and Wurjanto (1991) for the irregular wave transformation over a barred beach.

Assuming that $u < c$ in the vicinity of the landward boundary located at $x = x_e$ where the normalized water depth below SWL is $d_e = d_e'/H_r'$, α and β represent the characteristics advancing landward and seaward, respectively, in the vicinity of the landward boundary. The boundary conditions at $x = x_e$ may then be expressed as (Kobayashi and Wurjanto 1989b, 1989c)

$$h = d_e + \eta_t(t) \quad \text{at } x = x_e \tag{48}$$

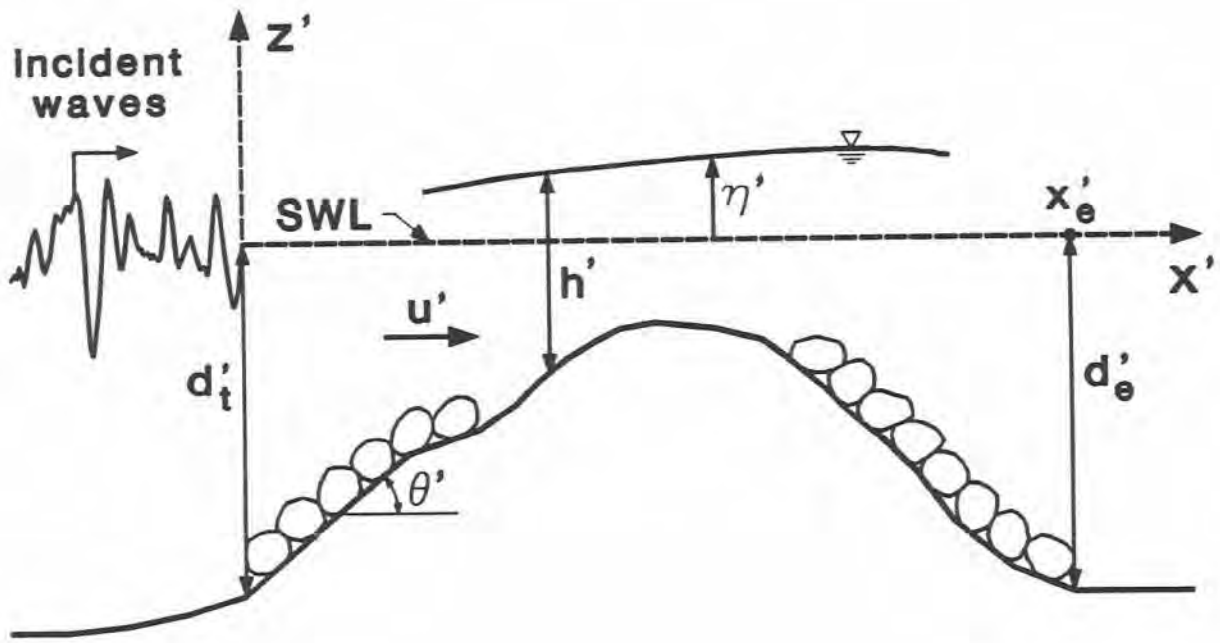


Figure 3. Wave transmission over a submerged structure.

$$\eta_t(t) = \frac{1}{2}\sqrt{d_e}\,\alpha(t) - d_e \quad \text{at } x = x_e \tag{49}$$

where η_t is the free surface oscillation at $x = x_e$ normalized by H_r' due to the transmitted wave, provided that no wave propagates seaward from the region $x > x_e$. Eq. 49 expresses the transmitted wave train η_t in terms of the landward-advancing characteristic α given by Eq. 9 in a manner similar to Eq. 13 for the reflected wave train except that the nonlinear correction term is neglected in Eq. 49.

The following numerical procedure is used to compute the values of $\mathbf{U}_j^*$ at the next time level t^* for the nodes $j = (s-1)$ and s, where the integer s for this case is the landward boundary node j_e located at $x = x_e$

1. Compute $\mathbf{U}_{s-1}^*$ using Eq. 17 with $j = (s-1)$ with the damping term $\mathbf{D}_{s-1}$ since the water depth h is large at this node.

2. Compute α_s^* at the time t^* using Eq. 46.

3. Compute η_t^* at the time t^* using Eq. 49 with $\alpha = \alpha_s^*$. Then, $h_s^* = (d_e + \eta_s^*)$ from Eq. 48, while $u_s^* = (\alpha_s^* - 2\sqrt{h_s^*})$. Thus, $m_s^* = h_s^* u_s^*$, and $\mathbf{U}_s^*$ is obtained.

The transmission coefficient associated with the computed periodic wave train $\eta_t(t)$ may be estimated using the following equations if $d_e = d_t$.

$$T_1 = \frac{1}{K_s}\left[(\eta_t)_{max} - (\eta_t)_{min}\right] \tag{50}$$

$$T_2 = \sqrt{\frac{\overline{\eta_t^2}}{\overline{\eta_i^2}}} \tag{51}$$

$$T_3 = \sqrt{\frac{\overline{(\eta_t - \overline{\eta_t})^2}}{\overline{\eta_i^2}}} \tag{52}$$

Eqs. 50, 51, and 52 correspond to Eqs. 39, 40, and 41, respectively for the reflection coefficients discussed in Section 2.5. For the case of regular waves, the transmission coefficient T_1 represents the ratio of the height of the transmitted wave to that of the incident wave. For random waves, T_1 indicates only the magnitude of the transmitted irregular wave. Eqs. 51 and 52, which are applicable to both regular and irregular waves, are based on the time-averaged transmitted wave energy as compared to the time-averaged incident wave energy, $\overline{\eta_i^2}$. The estimation $\overline{\eta_i^2} = K_s^2/8$ based on linear wave theory may be employed for the case of regular waves. Eq. 52 accounts for the difference $\overline{\eta_t}$ between the still water level and the mean water level at $x = x_e$. The computer program RBREAK computes T_1, T_2, and T_3 for the case of regular waves, and only T_1 for the case of irregular waves. For the latter, T_2 and T_3 can be computed from the irregular incident and transmitted wave trains stored during the computation.

• 2.9 •

Wave Energy Balance

The normalized equations of mass and x-momentum given by Eqs. 6 and 7 are used to compute the flow field. The normalized energy equation corresponding to Eqs. 6 and 7 may be expressed as (Kobayashi and Wurjanto 1989b, 1989c)

$$\frac{\partial E}{\partial t} + \frac{\partial}{\partial x}(E_F) = -D_f - D_B \tag{53}$$

with

$$E = \begin{cases} \frac{1}{2}\left(hu^2 + \eta^2\right) & \text{for } h > \eta \\ \frac{1}{2}\left[hu^2 + \eta^2 - (h-\eta)^2\right] & \text{for } h < \eta \end{cases} \tag{54}$$

$$E_F = uh\left(\frac{1}{2}u^2 + \eta\right) \tag{55}$$

$$D_f = f|u|u^2 \tag{56}$$

where

E = normalized specific energy defined as the sum of kinetic and potential energy per unit horizontal area
E_F = normalized energy flux per unit width
D_f = normalized rate of energy dissipation per unit horizontal area due to bottom friction
D_B = normalized rate of energy dissipation per unit horizontal area due to wave breaking

The dimensional rate D'_B of energy dissipation due to wave breaking is given by $D'_B = (\rho g H'^2_r / T'_r) D_B$ where ρ = fluid density, which is assumed to be constant neglecting air bubbles. The normalized potential energy is taken to be relative to the normalized potential energy at the time $t = 0$ when the incident wave train arrives at $x = 0$ as shown in Figs. 1-3. The first and second parts of Eq. 54 are applicable for the portion of the structure below and above SWL, respectively.

Since the wave energy balance is normally analyzed in terms of the time-averaged quantities, the time-averaged dissipation rate, $\overline{D_B}$, due to wave breaking is computed using the time-averaged energy equation derived from Eq. 53

$$\overline{D_B} = -\frac{d}{dx}(\overline{E_F}) - \overline{D_f} - \frac{1}{t_{end} - t_{begin}}\left[E|_{t_{end}} - E|_{t_{begin}}\right] \tag{57}$$

where the overbar indicates time averaging over the time period spanning from the normalized time t_{begin} to the time t_{end} for the statistical calculations discussed in Section 2.4. For regular waves, $t_{end} = (t_{begin} + 1)$ and the last term of Eq. 57 is zero because of periodicity. For irregular waves, t_{begin} and t_{end} should be chosen such that the last term of Eq. 57 is negligible so as to obtain the time-averaged values for the stationary sea state.

The present numerical model needs to predict that $\overline{D_B}$ is positive or zero depending on whether wave breaking occurs or not. The energy flux $\overline{E_F}$ should decrease with the increase of x, while $\overline{D_f} > 0$ since D_f defined in Eq. 56 is positive. It should be noted that Eq. 57 may be used even for the region which is not always exposed to water since $h = 0$ and $u = 0$ in the absence of water.

For the case of wave overtopping or transmission, integration of Eq. 57 from the seaward boundary to the landward boundary yields the time-averaged energy equation for the region $0 \leq x \leq x_e$.

$$\overline{E_F}(x=0) - \overline{E_F}(x=x_e) = \int_0^{x_e} \left(\overline{D_f} + \overline{D_B}\right) dx + \int_0^{x_e} \frac{1}{t_{end} - t_{begin}} \left[E|_{t_{end}} - E|_{t_{begin}} \right] dx \quad (58)$$

where the first and second terms on the left hand side of Eq. 58 are the values of $\overline{E_F}$ at $x = 0$ and $x = x_e$, respectively. The last term in Eq. 58 is zero (because of periodicity) for the case of regular waves, and should be negligible for the case of irregular waves. Eq. 58 implies that the difference between the net energy fluxes at the seaward and landward boundaries equals the rate of energy dissipation between the two boundaries for the stationary sea state. For the case of wave run-up on a slope, Eq. 58 needs to be modified such that $\overline{E_F}(x = x_e) = 0$ and x_e should be interpreted as the maximum value of x reached by the waterline on the slope.

The specific energy $\overline{E}$ and the energy flux $\overline{E_F}$ for regular waves at the seaward boundary where $\eta = (\eta_i + \eta_r)$ at $x = 0$ from Eq. 12 may approximately be given by (Kobayashi and Wurjanto 1989b, 1989c)

$$\overline{E} \simeq \overline{\eta_i^2} + \overline{(\eta_r - \overline{\eta_r})^2} \quad \text{at } x = 0 \quad (59)$$

$$\overline{E_F} \simeq \sqrt{d_t} \left[\overline{\eta_i^2} - \overline{(\eta_r - \overline{\eta_r})^2} \right] \quad \text{at } x = 0 \quad (60)$$

where $\sqrt{d_t}$ is the normalized group velocity at $x = 0$ based on linear long wave theory. The reflection coefficient r_3 given by Eq. 41 including the effect of $\overline{\eta_r}$ is based on Eqs. 59 and 60. The reflection coefficient r_2 given by Eq. 40 corresponds to Eqs. 59 and 60 with $\overline{\eta_r} = 0$.

For the case of regular wave transmission over a submerged breakwater, the specific energy $\overline{E}$ and the energy flux $\overline{E_F}$ at the landward boundary where $\eta = \eta_t$ at $x = x_e$ may be approximated by (Kobayashi and Wurjanto 1989b, 1989c).

$$\overline{E} \simeq \overline{(\eta_t - \overline{\eta_t})^2} \quad \text{at } x = x_e \quad (61)$$

$$\overline{E_F} \simeq \sqrt{d_e}\ \overline{(\eta_t - \overline{\eta_t})^2} \quad \text{at } x = x_e \quad (62)$$

where $\sqrt{d_e}$ is the normalized group velocity at $x = x_e$ based on linear long wave theory. The transmission coefficient T_3 given by Eq. 52 for the case of $d_e = d_t$ is based on Eqs. 61 and 62, whereas the transmission coefficient T_2 given by Eq. 51 does not include the wave set-up $\overline{\eta_t}$ at $x = x_e$.

• 2.10 •

HYDRAULIC STABILITY OF ARMOR UNITS

The hydraulic stability of armor units is analyzed using the computed flow field on a rough impermeable slope. The drag, lift, and inertia forces acting on individual armor units may be expressed in terms of the fluid velocity and acceleration on the rough impermeable slope. The normalized fluid acceleration, du/dt, is given by

$$\frac{du}{dt} = \frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = -\frac{\partial h}{\partial x} - \theta - \frac{f|u|u}{h} \tag{63}$$

where use is made of Eqs. 6 and 7.

Kobayashi and Otta (1987) expressed the stability condition against sliding or rolling of an armor unit located on the slope with its local slope angle θ' as shown in Fig. 1 in the following form

$$|N_s + E_1| + E_2 N_s \leq E_3 \tag{64}$$

which is applicable for the case of $u \neq 0$. The stability number N_s in Eq. 64 is defined as

$$N_s = \frac{H_r'}{[W'/(\rho s_g)]^{1/3}(s_g - 1)} \tag{65}$$

where

H_r' = reference wave height used for the normalization of the governing equations
s_g = specific gravity of the armor unit whose unit mass is given by ρs_g
W' = median mass of the armor unit

If the stability number N_s required for the hydraulic stability of armor units is known, the required mass W' can be found using Eq. 65 for given H_r'. E_1, E_2, and E_3 in Eq. 64 are defined as

$$E_1 = \frac{2C_3^{2/3}}{C_2 C_D |u|u}\left[\frac{C_M}{(s_g - 1)\sigma}\frac{du}{dt} - sin\theta'\right] \tag{66}$$

$$E_2 = \frac{C_L\, tan\phi}{C_D} \tag{67}$$

$$E_3 = \frac{2C_3^{2/3}}{C_2 C_D u^2}\, cos\theta'\, tan\phi \tag{68}$$

where

C_D = drag coeficient
C_L = lift coeficient
C_M = inertia coeficient
C_2 = area coefficient

C_3 = volume coeficient
ϕ = frictional angle of armor units

Eq. 64 can be solved in terms of N_s

$$N_s \leq N_R(t,x) = \begin{cases} (E_1 + E_3)/(E_2 - 1) & \text{if } E_1 < 0,\ E_2 > 1,\ \text{and } E_3 < (-E_1 E_2) \\ (E_3 - E_1)/(E_2 + 1) & \text{otherwise} \end{cases} \tag{69}$$

where N_R = dimensionless function expressing the degree of the armor unit stability as a function of t and x. For the computation, Eq. 69 is used if $|u| \geq 10^{-3}$ and N_R is set to be $N_R = 1000$ if $|u| < 10^{-3}$.

For the case of $u = 0$, Kobayashi and Otta (1987) expressed the stability condition in the form

$$\left| \frac{C_M}{(s_g - 1)\sigma} \frac{du}{dt} - sin\theta' \right| \leq cos\theta' \, tan\phi \tag{70}$$

The condition given by Eq. 70 is satisfied if the normalized fluid acceleration remain within the following lower and upper bounds

$$\sigma \, a_{min} \leq \frac{du}{dt} \leq \sigma \, a_{max} \tag{71}$$

$$a_{min} \geq -\frac{s_g - 1}{C_M} \frac{sin(\phi - \theta')}{cos\phi} \quad ; \quad a_{max} \leq \frac{s_g - 1}{C_M} \frac{sin(\phi + \theta')}{cos\phi} \tag{72}$$

In terms of the dimensional variables, Eq. 71 can be rewritten as $g a_{min} \leq du'/dt' \leq g a_{max}$ where g = gravitational acceleration. The dimensionless parameters a_{min} and a_{max} need to be chosen so as to satisfy the conditions given by Eq. 72 as discussed by Kobayashi and Otta (1987).

The local stability number $N_{sx}(x)$ for initiation of armor movement at given location x is defined as the minimum value of $N_R(t,x)$ at the same location for a specified duration. For incident regular waves, this duration can be taken as one wave period after the establishment of the periodicity of $N_R(t,x)$ with respect to t. If $N_s \leq N_{sx}(x)$, the armor unit located at given x will not move during the specified duration. The critical stability number N_{sc} for initiation of armor movement is defined as the minimum value of $N_{sx}(x)$ with respect to x in the computation domain. If $N_s \leq N_{sc}$, no armor units in the computation domain will move during the specified duration.

• 2.11 •
Movement of Armor Units

Kobayashi and Otta (1987) also performed a simplified analysis to predict the sliding motion of armor units when the criterion for initiation of armor movement is exceeded. In

the following, the results presented by Kobayashi and Otta (1987) are rearranged so that the computer program attached in Appendix A may be understood without difficulties.

The normalized forces acting on an armor unit are separated into

$$F_D = \frac{C_2 C_D}{2 C_3 d} \sigma |u - u_a| (u - u_a) \tag{73}$$

$$F_L = \frac{C_2 C_L}{2 C_3 d} \sigma (u - u_a)^2 \tag{74}$$

$$F_I = C_M \frac{du}{dt} \tag{75}$$

$$W_c = \sigma (s_g - 1) cos\theta' \tag{76}$$

$$W_s = \sigma (s_g - 1) sin\theta' \tag{77}$$

with

$$d = \frac{d'}{H_r'} = \frac{1}{H_r'} \left(\frac{W'}{C_3 \rho s_g} \right)^{1/3} ; \quad u_a = \frac{u_a'}{\sqrt{g H_r'}} \tag{78}$$

where

F_D = normalized drag force
F_L = normalized lift force
F_I = normalized inertia force due to the fluid acceleration only
W_c = component of the normalized submerged weight downward *normal* to the slope
W_s = component of the normalized submerged weight downward *parallel* to the slope
d = normalized characteristic length of the armor unit
u_a = normalized velocity of the armor unit along the slope

The prime in Eq. 78 indicates the corresponding physical variable. It is simply assumed that the drag and inertia forces act upward or downward parallel to the slope, whereas the lift force acts upward normal to the slope. The normalized forces expressed by Eqs. 73-77 are based on the normalization by $(gW'/\sigma s_g)$. It is noted that the condition given by Eq. 71 is *not* imposed on the value of du/dt in Eq. 75 to account for possibly large fluid accelerations at the point of wave breaking.

The sliding motion of an armor unit starts if the following condition is satisfied

$$|F_D + F_I - W_s| > F_R \tag{79}$$

with

$$F_R = (W_c - F_L) tan\phi \geq 0 \tag{80}$$

where F_R = magnitude of the normalized frictional force acting on the armor unit which is zero if $F_L \geq W_c$. In Eqs. 79 and 80, F_D and F_L are given by Eqs. 73 and 74 with $u_a = 0$, respectively, where $u_a = 0$ for a stationary armor unit. The normalized equation of the sliding motion of the armor unit moving with the normalized velocity u_a along the slope is given by

$$(s_g + C_m) \frac{du_a}{dt} = F_D + F_I - W_c - J\, F_R \tag{81}$$

with

$$J = \frac{u_a}{|u_a|} \tag{82}$$

where C_m = added mass coefficient given by $C_m = (C_M - 1)$ and F_R is assumed to act in the direction opposite to that of the armor movement. The displacement X'_a of the sliding unit along the slope from its initial location is normalized in the following two different ways

$$X_a(t) = \frac{X'_a}{d'} = \frac{\sigma}{d} \int_{t_0}^{t} u_a \, dt \tag{83a}$$

$$X_{aa}(t) = \frac{X'_a}{T'_r \sqrt{g H'_r}} = \int_{t_0}^{t} u_a \, dt \tag{83b}$$

where t_0 = normalized time when the armor unit starts moving. Eq. 83a is used to estimate the degree of the armor movement relative its characteristic length d', whereas Eq. 83b is used to find the x-coordinate of the moving unit since the values of u and du/dt in Eqs. 73-75 should be those at the instantaneous location of the unit.

For the computation of the movement of individual armor units, the grid points used for the computation of the flow fields are used to specify the locations of the units before the armor movement computation. The movement of the armor unit located at the node j starts when the condition given by Eq. 79 is satisfied at the node j. If the armor unit located initially at the node j starts moving, a Lagrangian approach is used by tracking the location of the moving unit identified by its node number j. A forward difference equation in the time t derived from Eq. 81 is used to find the normalized velocity u_a of the identified unit whose instantaneous location is computed using Eq. 83b. The values of u and du/dt in Eqs. 73-75 are evaluated at the node closest to the instantaneous location of the moving unit. The moving unit is assumed to stop when the condition given by Eq. 79 is not satisfied. The stopped unit resumes its movement when the condition given by Eq. 79 is satisfied. The temporal variation of X_a defined by Eq. 83a is also computed for the armor unit identified by its initial location on the slope.

PART III
COMPUTER PROGRAM RBREAK

• 3.1 •
INTRODUCTION

The computer program RBREAK attached in Appendix A consists of the main program, 49 subroutines, and one function. Full double precision mode is used throughout the program to gain maximum numerical accuracy. The program has been tested on IBM 3090 running under VM/XA and on Sun 4/280 operating under UNIX-based SunOS. Written in standard FORTRAN-77, RBREAK should run on other machines.

The term *hydrodynamic computation* in this report refers to the time-marching computation as primarily explained in Section 2.2 with the seaward boundary condition described in Sections 2.3 and 2.5 and the landward boundary conditions in Sections 2.6, 2.7, and 2.8. The hydrodynamic computation constitutes the heart of the computer program RBREAK. Numerical problems, which determine the success or failure of an RBREAK computation, arise mostly from the hydrodynamic computation.

The computer program RBREAK is an extension of its predecessor, IBREAK (Kobayashi and Wurjanto 1989d), which is based on essentially the same theory as described in Part II. The extension is necessary to overcome numerical difficulties which cause abortion of the hydrodynamic computation for irregular waves. These are the same numerical difficulties encountered in the hydrodynamic computation for regular or transient waves using IBREAK. In IBREAK, a single set of the computation parameters such as the time step size is used throughout the computation duration. If a failure occurs, another attempt is carried out using a different set of the computation parameters. This trial and error approach is tolerable for regular or transient waves since the computation duration is typically less than 10 wave periods and very short. This approach becomes very time-consuming for random waves since the required computation duration is long to simulate a representative random wave train which may consist of 200 individual waves.

Potential numerical problems in the computer program RBREAK have been identified and remedial measures have been devised. Numerical problems can be minimized by adjusting the computation parameters during the hydrodynamic computation. The computation duration is divided into *computation units*, and each unit is treated separately. The computation for each unit is considered successful when a set of the computation parameters with no numerical problem is found. A number of trial runs may be required before this set is found. Since RBREAK already contains many options and is too long to make major modifications, a separate program called BEFORR is developed to find the set of the computation parameters for the successful computation of each unit. This program is the forerunner of RBREAK, that is, BEFORR should

be executed before RBREAK. The use of the computation parameters for each unit produced by BEFORR will result in the successful hydrodynamic computation of RBREAK.

The computer program BEFORR will be discussed in Part IV. In the following sections, the inner working of the computer program RBREAK will be discussed with the assumption that the computation parameters for each computation unit have been supplied by the preceding BEFORR computation. If the preceding BEFORR computation is successful, the subsequent hydrodynamic computation by RBREAK should not fail. If a failure occurs, it must be either because input data files have been mixed up or because the non-hydrodynamic parts of RBREAK such as armor movement have encountered numerical difficulties.

The duo BEFORR-RBREAK can do everything that IBREAK can do but does it more efficiently because of the self-adjusting capability for the successful numerical computation embedded in BEFORR-RBREAK. Appendix C presents examples on how to prepare input data files for the four representative cases given as examples in the IBREAK report (Kobayashi and Wurjanto 1989d). RBREAK, however, is primarily intended for irregular wave computations since the self-adjusting capability is crucial for the computation of long duration. The subsequent sections are written with irregular waves in mind.

• 3.2 •

Input Data Files for RBREAK

To execute the computer program RBREAK for the case of irregular waves, three data files are needed.

1. *Primary input data file.* This file contains all the variables and parameters needed to specify the case being investigated, except the input wave train, which is prescribed in the second input data file.

2. File containing the *input wave train* as explained in Section 2.3.2.

3. File `BINPUT` created by the preceding BEFORR computation for the successful hydrodynamic computation of each computation unit.

The first two input data files are prepared by a user, and should have been used in the preceding BEFORR computation. The third is produced by the preceding BEFORR computation. Fig. 4 depicts how the computer programs BEFORR and RBREAK are related through the input/output files for the case of *irregular* waves. The positions of the three input data files listed above are clearly shown in that diagram. The user has the freedom for selecting the names of the first and second input data files which are to be read by RBREAK as the variables `FINP1` and `FINP2`, respectively. The only limitation imposed by RBREAK is that the name should consist of no more than ten characters. The operating system under which RBREAK is running may dictate a certain convention regarding file naming.

The user enters the name of the primary input data file interactively at the beginning of RBREAK computation. The name of the file containing the input wave train is specified in the

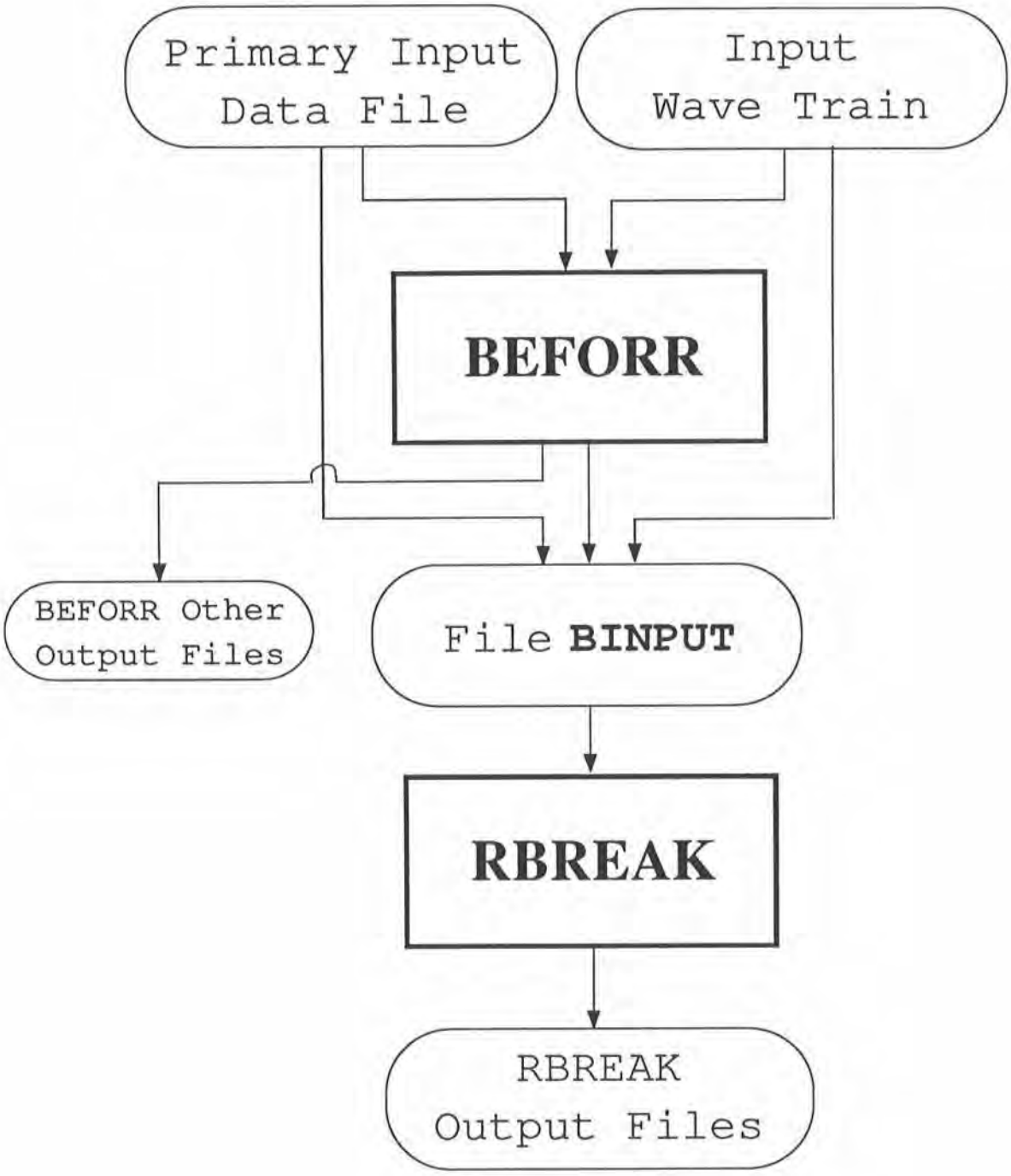


Figure 4. An overview of the computer programs BEFORR and RBREAK.

primary input data file. The preparation of the primary input data file may be best explained by use of the examples given in Appendices C through G. In addition, the order of the the contents of the primary input data file will be presented in Section 3.6. The generation of the file BINPUT will be elaborated in Part IV. In the following, the contents of the second input data file are explained.

The input wave train is read by Subr. 03 INPUT2, where the subroutines are explained in Section 3.4, as follows:

```
      READ (12,9000) NDATA
      READ (12,8000) (ETA(I),I=1,NDATA)
8000 FORMAT (5D15.6)
9000 FORMAT (I8)
```

where

NDATA = number of data points
ETA = array of size NDATA prescribing the free surface elevation at the seaward boundary

It is required that $1 \leq$ NDATA $\leq$ N2, where the parameter N2 is to be explained in the COMMON /DIMENS/ in Section 3.5. The data points of ETA are equally spaced in time and NDATA is given by

$$\text{NDATA} = \frac{\text{TMAX}}{\delta t} + 1 \tag{84}$$

where

TMAX = normalized computation duration specified in the primary input data file by the user
δt = normalized time increment of the data points

It is noted that the values at the very beginning ($t = 0$) and at the very end ($t =$TMAX) of the input wave train need to be specified even if they are the same. In other words, The NDATA points cover the time span $0 \leq t \leq$ TMAX. This caution is applicable to the case where the input wave train is generated for given spectral density using random phases. The wave train generated in this way is periodic over TMAX, in which case the first and NDATA-th data points are identical.

The time increment of the input wave train, δt, is normally much larger than the finite difference time step Δt for the stable numerical computation discussed in Part II. The time step Δt, which may vary from one computation unit to another, has been determined by the preceding BEFORR computation. The computer program RBREAK performs a simple linear interpolation of the input wave train to get the appropriate value at each time level during the time-marching computation. The interpolation is carried out in Subr. 19 SEABC.

• 3.3 •

MAIN PROGRAM RBREAK

The main program lists all the important parameters and variables in the COMMON blocks. These parameters and variables are described in Section 3.5. The main program coordinates tasks which are actually executed by subroutines. The tasks can be categorized into five groups.

1. Reading Input Data.
2. Checking FORTRAN PARAMETERs in the Subroutines.
3. Groundwork.
4. Time-marching Computation.
5. Finishing.

3.3.1 READING INPUT DATA

The first variable read by RBREAK is MREP, which is specified interactively by the user. MREP determines the interval between two consecutive progress messages RBREAK displays on a terminal screen. The progress message is displayed every MREP wave periods where the period herein is the reference wave period. For example, if MREP=4, the following message (without the horizontal lines) will appear on the screen when the computation has just finished 56 wave periods.

```
Finished          56 Wave Periods
```

This reporting is useful in keeping track of the progress of a long computation. If not necessary, however, the reporting can be turned off by specifying MREP=0. It is required that MREP $\geq$ 0.

The next input is the name of the primary input data file, which is also entered interactively. The three input data files explained in Section 3.2 are then read from the prepared files. Input and output files are opened by calling Subr. 01 OPENIO. The contents of the first and the second input data files are read by Subr. 02 INPUT1 with MODE=1, where MODE is used to identify different parts of subroutines, and Subr. 03 INPUT2, respectively, at the beginning of the computation before the time-marching computation begins. A list of the READ statements corresponding to the primary input data file will be presented in Section 3.6. The computation parameters in the file BINPUT are read sequentially (unit by unit) by Subr. 02 INPUT1 with MODE=9 during the time-marching computation.

3.3.2 CHECKING FORTRAN PARAMETERS IN THE SUBROUTINES

In the computer program RBREAK, the dimension of an array is specified using an integer which is independently declared as a PARAMETER by each program unit where the term *program unit* is used herein to represent the main program, subroutines, and functions. The almost

all of RBREAK's variables, which are mostly arrays, are passed between program units using COMMON blocks. Only few variables are passed as arguments of subroutines and functions. This arrangement demands that the dimensions of arrays in the COMMON blocks throughout RBREAK remain the same. Subr. 48 CKSUBR detects possible mismatches in the array dimensions among the program units. Using the PARAMETERs specified in the main program as reference, Subr. 48 CKSUBR calls each applicable subroutine with MODE=0 or ICALL=0 to check if the required agreement is satisfied. In Subr. 10 CPARAM, MODE=0 is also used to ensure that RBREAK reads the correct file BINPUT produced by the preceding BEFORR computation. In this effort, four case-specific parameter values, called *case-signatures* herein, are passed from BEFORR to RBREAK through the file BINPUT to assure that these values are equal to their RBREAK counterparts. A mismatch means that a wrong file BINPUT is being used. The checking of the case-signatures is carried out in Subr. 02 INPUT1 and 03 INPUT2 in addition to Subr. 10 CPARAM.

3.3.3 Groundwork

The tasks performed in preparation for the time-marching computation include

- Computation of the bottom geometry using Subr. 04 BOTTOM.
- Computation of the wave parameters using Subr. 05 WPARAM, and the armor stability parameters using Subr. 40 SPARAM if applicable.
- Assignment of the initial values using Subr. 26 INIT.
- For regular waves, compute the initial regular wave profile based on the minimum allowable value of NONE using Subr. 06 REGWAV where NONE is the number of time steps in one wave period.

3.3.4 Time-Marching Computation

The time-marching computation is executed for MWAVE computation units in sequence. Each computation unit is one wave period long except the last unit, which may be less than one wave period long. One wave period in the time-marching computation is unity on the basis of the normalization by the reference wave period. The time step Δt is constant within each computation unit, but may vary from one computation unit to another. Δt is the inverse of NONE.

$$\Delta t = \frac{1}{\texttt{NONE}} \tag{85}$$

The value of NONE for each computation unit is supplied by the file BINPUT. RBREAK uses the integer M to indicate the computation unit and the integer N to denote the present time level within each computation unit. M is varied from 1 to MWAVE, whereas N is changed from 1 to NEND, where NEND is the final time level in each computation unit. NEND is equal to NONE for the computation units with M = 1,2,...,(MWAVE-1), and may be less than NONE for the final computation unit.

The *present* time level t discussed in Part II should now be denoted as the time level $t = [(\texttt{M-1}) + (\texttt{N-1})\Delta t]$. Similarly, the *next* time level $t^* = (t + \Delta t)$ in Part II is now the time $t^* = [(\texttt{M-1}) + \texttt{N}\Delta t]$.

During the time-marching computation, the unknown quantities at the time $t^* = [(\texttt{M-1}) + \texttt{N}\Delta t]$ are computed from the known quantities at the time $t = [(\texttt{M-1}) + (\texttt{N-1})\Delta t]$.

At the beginning of each computation unit, the following is performed.

- Initialize the quantities used to calculate time-averaged values (Subr. 28 INSTAT)
- Read the computation parameters NONE, NEND, and DELTA (Subr. 02 INPUT1 with MODE=9). NONE and NEND have been described above. DELTA is the normalized depth at the computational waterline.
- If the value of NONE of the present computation unit differs from that of the previous unit:
  - Calculate the computation parameters which are dependent on NONE (Subr. 10 CPARAM).
  - For the case of regular waves: Compute the regular wave profile (Subr. 06 REGWAV).

Time-marching from one time level to the next is done as follows.

1. Estimate h_{s+1}, u_{s+1}, and m_{s+1} at the time $t = [(\texttt{M-1}) + (\texttt{N-1})\Delta t]$ by linear extrapolation for the case of wave run-up where the integer s indicates the wet node next to the moving waterline at $t = [(\texttt{M-1}) + (\texttt{N-1})\Delta t]$ (Subr. 12 EXTRAP).
2. Retain the values of the quantities at the time $t = [(\texttt{M-1}) + (\texttt{N-1})\Delta t]$ which are required for the seaward and landward boundary computations (Subr. 13 RETAIN).
3. Compute $c_j = \sqrt{h_j}$ for $j = 1, 2, ..., s$ used in the characteristic equations given by Eqs. 9 and 10 with Eq. 11 (Subr. 14 CRITV).
4. Compute the unknown quantities at the time $t^* = [(\texttt{M-1}) + \texttt{N}\Delta t]$ (Subr. 15 MARCH, 16 LANDBC, and 19 SEABC).
5. Check the simplified condition of $|u| < (\Delta x/\Delta t)$ that should be satisfied if the numerical stability criterion given by Eq. 27 is satisfied (Subr. 11 NUMSTA).
6. Compute the quantities related to wave energy balance, if applicable (Subr. 30 ENERGY).
7. Compute the statistics of η, u, and $m = uh$ so that the mean, maximum, and minimum values can be found after the time-marching computation (Subr. 41 VECMAT and 35 STAT1).
8. Compute the hydraulic stability or movement of armor units, if applicable (Subr. 31 STABNO or 32 MOVE).

3.3.5 FINISHING

After the time-marching computation, the following tasks are performed:

- Completion of the statistical calculations (Subr. 37 STATC).
- Completion of the computation of the time-averaged energy quantities (Subr. 39 ENERGC).
- Writing the results (Subr. 45 DOC3). It should be noted that some documentation is written before the time-marching computation (Subr. 43 DOC1) and some results are written during the time-marching computation (Subr. 44 DOC2).

• 3.4 •

SUBROUTINES AND FUNCTION

The 49 subroutines and one function arranged in numerical order in the computer program RBREAK are listed in Table 1. The page numbers for the subroutines and function listed in Table 1 correspond to the page numbers in the RBREAK listing presented in Appendix A. Interdependence among the program units are mapped out in Table 2. Each of the subroutines and function are explained concisely in the following. Explanation is given in the format: Number – NAME – Description, where the Number refers to the numerical order in the computer program RBREAK.

01 OPENIO opens the input and output files with the assistance from Subr. 50 OPENF.

02 INPUT1 reads information from the primary input data file and the file BINPUT. It also performs checks on (1) parameters related to the dimensions of arrays and (2) two of the case-signatures, MWAVE and FINP2, as explained in Section 4.6. A rigorous scrutiny on the most of input data has been conducted in the preceding BEFORR computation.

03 INPUT2 reads the input wave train, that is, the wave train at the seaward boundary prescribed by a user as explained in Section 2.3.2.

04 BOTTOM computes the normalized structure geometry and the value of Δx from the dimensional structure geometry specified as input.

05 WPARAM calculates the dimensionless wave parameters used in the other subroutines.

06 REGWAV computes the incident wave profile at the seaward boundary using Eq. 30 or 33 for the case of regular waves.

07 FINDM computes the value of the parameter m which satisfies Eq. 36.

08 CEL computes the values of the complete elliptic integrals K and E used in Eqs. 33-36 for given m. CEL is the only function in the computer program RBREAK.

09 SNCNDN computes the Jacobian elliptic function cn used in Eq. 33.

10 CPARAM computes the parameters which are dependent on NONE for each computation unit. This subroutine also checks one of the case-signatures, NONM, when MODE=0.

11 NUMSTA checks if the numerical stability criterion is violated (IPROB=3).

12 EXTRAP estimates the hydrodynamic quantities at the node immediately landward of the present waterline node by extrapolation.

13 RETAIN retains the quantities at the present time level used for the landward and seaward boundaries.

14 CRITV calculates the critical velocities used in the characteristic variables and checks for negative water depth (IPROB=1).

Table 1: List of 49 subroutines and one function in computer program RBREAK.

No.	Subroutine (S) or Function (F)		Page No. in RBREAK	No.	Subroutine (S) or Function (F)		Page No. in RBREAK
01	S	OPENIO	A-7 - A-10	26	S	INIT	A-46 - A-49
02	S	INPUT1	A-10 - A-17	27	S	INITH	A-49
03	S	INPUT2	A-17 - A-18	28	S	INSTAT	A-49 - A-50
04	S	BOTTOM	A-18 - A-22	29	S	SVSTAT	A-50 - A-51
05	S	WPARAM	A-22 - A-23	30	S	ENERGY	A-51 - A-53
06	S	REGWAV	A-23 - A-25	31	S	STABNO	A-53 - A-55
07	S	FINDM	A-25 - A-26	32	S	MOVE	A-55 - A-57
08	F	CEL	A-26 - A-27	33	S	FORCES	A-57 - A-58
09	S	SNCNDN	A-27 - A-29	34	S	ACCEL	A-58 - A-59
10	S	CPARAM	A-29 - A-30	35	S	STAT1	A-59
11	S	NUMSTA	A-30 - A-31	36	S	STAT2	A-59
12	S	EXTRAP	A-31 - A-32	37	S	STATC	A-59 - A-61
13	S	RETAIN	A-32	38	S	COEF	A-61 - A-62
14	S	CRITV	A-32 - A-33	39	S	ENERGC	A-62 - A-64
15	S	MARCH	A-33 - A-35	40	S	SPARAM	A-64 - A-65
16	S	LANDBC	A-35 - A-38	41	S	VECMAT	A-65
17	S	RUNUP	A-38 - A-40	42	S	DERIV	A-65 - A-66
18	S	OVERT	A-40	43	S	DOC1	A-66 - A-69
19	S	SEABC	A-40 - A-42	44	S	DOC2	A-69 - A-72
20	S	MATAFG	A-42 - A-43	45	S	DOC3	A-72 - A-77
21	S	MATGJR	A-43	46	S	CKFPAR	A-77
22	S	MATS	A-43 - A-44	47	S	CKVAL	A-78
23	S	MATD	A-44 - A-45	48	S	CKSUBR	A-78 - A-80
24	S	MATU	A-45 - A-46	49	S	STOPP	A-80
25	S	NSI	A-46	50	S	OPENF	A-80

Table 2: Interdependence among the program units of computer program RBREAK.

No.	Program Unit	Called from	Makes Call(s) to
00	MAIN	-	01, 02, 03, 04, 05, 06, 10, 11, 12, 13, 14, 15, 16, 19, 26, 28, 29, 30, 31, 32, 35, 37, 39, 40, 41, 43, 44, 45, 48, 49
01	OPENIO	00	50
02	INPUT1	00, 48	46, 47, 49
03	INPUT2	00, 48	46, 47, 49
04	BOTTOM	00, 48	46, 47, 49
05	WPARAM	00, 48	46
06	REGWAV	00, 48	07, 08, 09, 46
07	FINDM	06	08
08	CEL	06, 07	-
09	SNCNDN	06	49
10	CPARAM	00	49
11	NUMSTA	00, 48	25, 46, 49
12	EXTRAP	00, 48	46
13	RETAIN	00, 48	46
14	CRITV	00, 48	25, 46, 49
15	MARCH	00, 48	20, 21, 22, 23, 24, 25, 46, 49
16	LANDBC	00, 48	17, 18, 35, 36, 46
17	RUNUP	16, 48	20, 21, 22, 46
18	OVERT	16, 48	46
19	SEABC	00, 48	36, 46
20	MATAFG	15, 17, 48	46
21	MATGJR	15, 17, 48	46
22	MATS	15, 17, 48	46
23	MATD	15, 48	46
24	MATU	15, 48	46
25	NSI	11, 14, 15	-
26	INIT	00, 48	27, 46
27	INITH	26	-
28	INSTAT	00, 48	46
29	SVSTAT	00, 48	46
30	ENERGY	00, 48	46

Continued on the next page.

Table 2 continued from the previous page.

No.	Program Unit	Called from	Makes Call(s) to
31	STABNO	00, 48	34, 46
32	MOVE	00, 48	33, 34, 46
33	FORCES	32, 48	46
34	ACCEL	31, 32, 48	41, 42, 46
35	STAT1	00, 16	-
36	STAT2	16, 19	-
37	STATC	00, 48	38, 46
38	COEF	37	-
39	ENERGC	00, 48	41, 42, 46
40	SPARAM	00, 48	46
41	VECMAT	00, 34, 39	-
42	DERIV	34, 39	-
43	DOC1	00, 48	46
44	DOC2	00, 48	46
45	DOC3	00, 48	46
46	CKFPAR	All units except for 00, 01, 07, 08, 09, 10, 25, 27, 35, 36, 38, 41, 42, 46, 47, 48, 49, 50	-
47	CKVAL	02, 03, 04	-
48	CKSUBR	00	All units except for 00, 01, 07, 08, 09, 10, 25, 27, 35, 36, 38, 41, 42, 46, 47, 48, 49, 50
49	STOPP	00, 02, 03, 04, 09, 10, 11, 14, 15	-
50	OPENF	01	-

15 MARCH performs the time-marching computation on the basis of Eq. 17.

16 LANDBC manages the landward boundary conditions for wave run-up, overtopping, or transmission, as well as the computation of the normalized free surface elevation Z_r for given δ_r' as discussed in relation to wave run-up.

17 RUNUP computes the waterline movement on the slope of a subaerial structure on the basis of the procedure discussed in Section 2.6.

18 OVERT computes the overtopping flow at the landward edge of the crest of a subaerial structure on the basis of the procedure discussed in Section 2.7.

19 SEABC computes the flow at the seaward boundary using Eq. 37 and the reflected wave train $\eta_r(t)$ using Eq. 13.

20 MATAFG computes the values of the elements of the first row of the matrix **A** defined in Eq. 19, the elements of the vector **F** in Eq. 16, and the first element of the vector **G** in Eq. 16.

21 MATGJR computes the values of the elements of the vector **g** defined by Eq. 18.

22 MATS computes the value of the first element of the vector **S** defined by Eq. 20.

23 MATD computes the values of the elements of the vector **D** defined by Eq. 22.

24 MATU computes the values of the elements of the vector $\mathbf{U}^*$ using Eq. 17.

25 NSI computes the numerical stability indicator Ω defined by Eq. 28.

26 INIT specifies the initial conditions given by $\eta = 0$ and $u = 0$ at $t = 0$, as well as the initial values of various quantities used for the subsequent computation.

27 INITH facilitates the assignment of the initial values in Subr. 26 INIT.

28 INSTAT initializes statistical quantities representing mean values at the beginning of a computation unit.

29 SVSTAT saves statistical quantities related to mean values at the end of a computation unit.

30 ENERGY computes the values of E, E_F, and D_f defined by Eqs. 54, 55, and 56, respectively, in relation to the normalized equation of wave energy. This subroutine also saves the values of E at the normalized time t_{begin} and t_{end}, which are introduced in Eq. 57.

31 STABNO computes the armor stability function $N_R(t,x)$ using Eqs. 66-69 and the local stability number $N_{sx}(x)$ defined as the minimum value of $N_R(t,x)$ at a given location.

32 MOVE computes the displacement of armor units using Eqs. 79-83.

33 FORCES computes the normalized forces given by Eqs. 73-77.

34 ACCEL computes the value of du/dt using Eq. 63.

35 STAT1 is used to calculate the sum, maximum, and minimum values of quantities varying with time.

36 STAT2 is used to calculate the sum, maximum, and minimum values of the surface elevations at the seaward and landward boundaries.

37 STATC completes the statistical calculations after the time-marching computation.

38 COEF computes the wave reflection coefficients given by Eqs. 39-41 as well as the wave transmission coefficients given by Eqs. 50-52 for the case of a submerged structure.

39 ENERGC completes the computation of energy quantities after the time-marching computation.

40 SPARAM calculates the parameters needed for computation of armor stability or movement.

41 VECMAT creates a vector from a specified row of a given matrix

42 DERIV computes the first derivative of a function using a finite difference method.

43 DOC1 documents the input data and dimensionless parameters before the time-marching computation.

44 DOC2 stores some of the computed results at designated time levels during the time-marching computation.

45 DOC3 documents the computed results after the time-marching computation.

46 CKFPAR checks whether the values of the integers N1R, N2R, N3R, N4R, and N5R used to specify the size of matrices and vectors in the main program are equal to the values of the corresponding integers N1, N2, N3, N4, and N5 used in the subroutines.

47 CKVAL checks whether some of the values selected by a user are within the ranges available in the present form of RBREAK.

48 CKSUBR calls subroutines which use any of the PARAMETERs N1, N2, N3, N4, and N5 to check the values of the PARAMETERs being used. The check is actually performed by Subr. 46 CKFPAR. Calls made from Subr. 48 CKSUBR are identified with MODE=1 or ICALL=1.

49 STOPP executes a programmed stop if some of the input requirements for RBREAK are not satisfied.

50 OPENF opens a specified file.

• 3.5 •

Parameters and Variables in COMMON Blocks

The parameters and variables included in the COMMON blocks in the main program RBREAK are explained in the following so that a user may be able to comprehend the computer program RBREAK and modify it if required.

/DIMENS/ contains the integers used to specify the sizes of matrices and vectors.

N1R = N1 = maximum number of spatial nodes allowed in the computation domain.

N2R = N2 = maximum allowed value of (NONE+1) for a regular wave computation, or maximum number of data points allowed in the input wave train. NONE is described in COMMON /CPAR2/ below.

N3R = N3 = maximum number of different values of the physical water depth δ_r' for wave run-up.

N4R = N4 = maximum number of points allowed to specify the structure geometry consisting of linear segments.

N5R = N5 = maximum number of spatial nodes where the time series of certain computed quantities are stored as well as maximum number of specified time levels at which special storing is requested.

The present setting for these integers is N1=800, N2=16000, N3=3, N4=40, N5=40. These integers can be changed as long as the changes are made throughout the program.

/CONSTA/ contains constants.

PI = π = 3.141592...

GRAV = gravitational acceleration $g = 9.81 m/s^2$ or $32.2 ft/s^2$.

/ID/ contains the integers used to specify the user's options.

IJOB indicates the type of the landward boundary condition. IJOB=1 for wave run-up on the seaward slope of a subaerial structure, IJOB=2 for wave overtopping over a subaerial structure, IJOB=3 for wave transmission over a submerged structure.

ISTAB indicates the type of the armor analysis. ISTAB=0 if no computation of armor stability or movement is desired, ISTAB=1 for the computation of armor stability, ISTAB=2 for the computation of armor movement.

ISYST indicates the system of units. ISYST=1 for the International System of Units (SI), ISYST=2 for the U.S. Customary System of Units (USCS).

IBOT indicates the type of input data for the structure geometry. IBOT=1 for the width and slope of linear segments, IBOT=2 for the locations of the seaward end points of linear segments. See Fig. 5 for clarification.

INONCT indicates whether the nonlinear correction term C_t is included or not in Eq. 13 in calculating $\eta_r(t)$. INONCT=0 for $C_t = 0$, INONCT=1 for C_t given by Eq. 38.

IENERG indicates whether the quantities related to wave energy are computed (IENERG=1) or not (IENERG=0).

IWAVE indicates the type of the wave profile specified at the seaward boundary. IWAVE=1 for the incident wave profile $\eta_i(t)$ computed using Subr. 06 REGWAV, IWAVE=2 for the incident wave profile $\eta_i(t)$ read by Subr. 03 INPUT2, IWAVE=3 for the total wave profile $[\eta_i(t) + \eta_r(t)]$ read by Subr. 03 INPUT2.
It is noted that IWAVE=3 corresponds to the free surface oscillation measured at the seaward boundary in the presence of a coastal structure. However, this seaward boundary condition has been found to produce spurious long-period oscillations in the computation domain (Packwood, 1980).

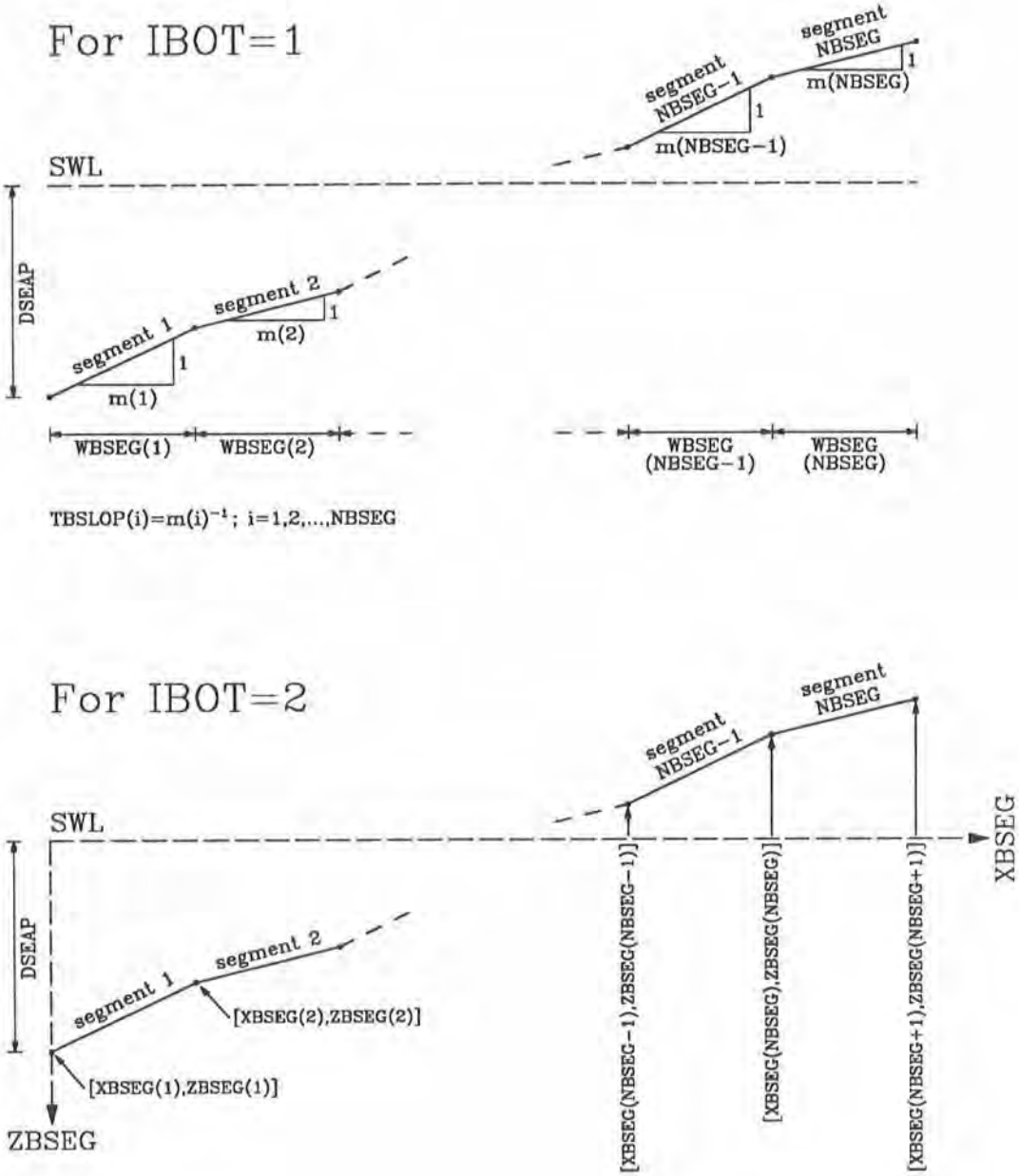


Figure 5. Input of dimensional structure geometry for IBOT=1 and IBOT=2.

ISAVA: ISAVA=1 indicates the storage of the spatial variations of η and u in the computation domain at specified moments. ISAVA=0 indicates no storage.

ISAVB: ISAVB=1 indicates the storage of the time series of m and h at specified nodes. ISAVB=0 indicates no storage.

ISAVC: ISAVC=1 indicates the storage of the time series of the armor stability function $N_R(t,x)$ given by Eq. 69 or the time series of the armor displacement X_a given by Eq. 83a from specified initial nodal locations. ISAVC=0 indicates no storage.

/CPAR1/ contains the computation parameters related to computation units.

MWAVE = number of computation units.

MSTAT: the first MSTAT computation units are excluded from the statistical calculations.

MSTAB: the first MSTAB computation units are excluded from the computation of armor stability or movement.

NTIMES, MSAVA1, MSAVA2: the spatial variations of η and u are stored NTIMES (>1) times at equal intervals per wave period from the computation unit MSAVA1 to MSAVA2, inclusive.

/CPAR2/ contains the computation parameters related to number of time steps.

MULTIF = multiplication factor associated with NONEM. See Eq. 86 in Part IV with additional explanation.

NONEM = minimum allowable value of NONE.

NONM = case-signature read from the file BINPUT, which must be equal to NONEM.

NONE = number of time steps in one wave period.

NEND = number of time steps in one computation unit, which does not exceed NONE.

NRATE = rate (data points per wave period) of storing output time series.

NHOP = NONE/NRATE

/CPAR3/ contains the computation parameters of related to spatial nodes.

INITS = number of the spatial nodes along the bottom below SWL (IJOB<3), or number of the spatial nodes between the seaward and landward boundaries, inclusive (IJOB=3). The spatial nodes are viewed to be located on the surface of the structure. It is required that $1 \leq \text{INITS} \leq (\text{N1} - 1)$.

JE = landward edge node (IJOB<3) or landward boundary node (IJOB=3).

JE1 = JE-1

/CPAR4/ contains the computation parameters related to the finite difference method used in RBREAK.

X1 = damping coefficient ϵ_1 in Eqs. 24 and 25.

X2 = damping coefficient ϵ_2 in Eqs. 24 and 25.

DELTA = normalized water depth defining the computational waterline. The primary input data file for RBREAK specifies an initial value of DELTA. This initial value should not exceed the value of DELTA for each computation unit determined by BEFORR starting from this initial value.

TMAX = normalized duration of the computation.

TIME = normalized present time.

T = time step Δt used for the discretization of Eq. 15. T may vary from one computation unit to another. T=1/NONE as defined in Eq. 85.

X = constant space size Δx used for the discretization of Eq. 15.

TX = $\Delta t/\Delta x$

XT = $\Delta x/\Delta t$

TTX = $(\Delta t)^2/\Delta x$

TTXX = $(\Delta t)^2/(\Delta x)^2$

TWOX = $2\Delta x$

/REQ1/ integers used for the special storage of the spatial variations of specified quantities.

IREQ: IREQ=1 indicates the special storage, IREQ=0 indicates no special storage.

In the following, IELEV, IV, IDUDT, and ISNR are indicators for the storage of the spatial variation of their respective quantities at specified moments: **1** for storage, **0** for no storage.

IELEV: for the normalized free surface elevation η.

IV: for the normalized depth-averaged velocity u.

IDUDT: for the normalized horizontal fluid acceleration du/dt.

ISNR: for the dimensionless armor stability function N_R.

NREQ = number of the specified moments at which the spatial variations of the requested quantities are stored. It is required that $1 \leq$ NREQ $\leq$ N5 when IREQ=1.

/REQ2/ contains the specified moments for the special storage.

TREQ(i) = specified normalized time with i=1,2,...,NREQ at which the spatial variations of the requested quantities are stored.

/WAVE1/ contains physical wave parameters.

HREFP = physical wave height at the "reference" location, specified in *meters* when SI is used or in *feet* when USCS is used.

TP = physical reference wave period in *seconds*.

WLOP = physical deep-water wavelength based on TP.

/WAVE2/ contains dimensionless wave parameters.

KS = KSSEA/KSREF. For regular wave, KS = normalized wave height at the seaward boundary.

KSREF = shoaling coefficient at the "reference" location.

KSSEA = shoaling coefficient at the seaward boundary.

WLO = normalized deep-water wavelength.

WL = normalized wavelength at the seaward boundary.

UR = Ursell number at the seaward boundary.

URPRE = Ursell number at the seaward boundary based on linear wavelength.

KSI = surf similarity parameter.

SIGMA = a measure of wave steepness.

/WAVE3/ contains time series of normalized free surface elevations.

ETA = time series of the incident wave at the seaward boundary, η_i, which is computed by Subr. 06 REGWAV for regular waves (IWAVE=1), or the input wave train for IWAVE>1.

The following variables are used only for regular wave computations. They are used to store their respective quantities during the last wave period.

ETAIS: for the incident wave at the seaward boundary, η_i.

ETARS: for the reflected wave at the seaward boundary, η_r.

ETATS: for the transmitted wave at the landward boundary, η_t.

/WAVE4/ contains the extreme values of ETA in /WAVE3/.

ETAMAX = maximum value of ETA in /WAVE3/.

ETAMIN = minimum value of ETA in /WAVE3/.

/WAVE5/ contains parameters related to cnoidal wave theory.

KCNO = complete elliptic integral of the first kind, $K(m)$, used in Eqs. 33-36.

ECNO = complete elliptic integral of the second kind, $E(m)$, used in Eqs. 34 and 36.

MCNO = parameter m computed from Eq. 36.

KC2 = value of $(1-m)$ used to compute the values of $K(m)$ and $E(m)$ using the Function 08 CEL.

/WAVE6/ contains integers related to ETA in /WAVE3/ for IWAVE>1 only.

NDATA = number of data points in the input wave train.

NDAT = case-signature read from the file BINPUT, which must be equal to NDATA.

/BOT1/ contains dimensional parameters related to the structure.

DSEAP = water depth d'_t below SWL at the seaward boundary.

DLANDP = water depth d'_e below SWL at the landward boundary used for IJOB=3 only.

FWP = constant friction factor f' used in Eq. 2.

/BOT2/ contains normalized parameters related to the structure.

DSEA = normalized water depth, $d_t = d'_t/H'_r$, at the seaward boundary.

DSEAKS = d_t/K_s corresponding to the value of d'_t/H'.

DSEA2 = normalized group velocity $\sqrt{d_t}$ at $x = 0$ based on linear long wave theory.

DLAND = normalized water depth, $d_e = d'_e/H'_r$, at the landward boundary for IJOB=3 only.

DLAND2 = normalized group velocity $\sqrt{d_e}$ at $x = x_e$ based on linear long wave theory.

FW = normalized friction factor f given in Eq. 5.

TSLOPS = tangent of slope, $tan\theta'_\xi$ used to define the surf similarity parameter ξ.

WTOT = normalized horizontal width of the computation domain = $(j_e - 1)\Delta x$.

/BOT3/ contains vectors related to the normalized structure geometry. The spatial nodes are viewed to be located on the surface of the structure. Index j refers to node number with j=1,2,...,JE.

U2INIT(j) = normalized water depth below SWL at the node j, corresponding to the value of $(-z)$ where z is given by Eq. 8.

THETA(j) = dimensionless gradient of the slope, θ_j, at the node j where θ is defined in Eq. 5.

SSLOPE(j) = $sin\theta'_j$ where θ'_j is the local angle of the slope at the node j. This quantity is required for the computation of armor stability and movement.

XB(j) = normalized x-coordinate of the node j

ZB(j) = normalized z-coordinate of the node j, corresponding to the normalized structure geometry given by Eq. 8.

/BOT4/ contains the integer NBSEG described below.

NBSEG = number of linear segments of different inclinations used to specify the structure geometry. It is required that $1 \leq$ NBSEG $\leq$ (N4 $- 1$).

/BOT5/ contains dimensional quantities associated with linear segments of the structure. All quantities listed below are segmental properties. Index i refers to segment number, with i=1,2,...,NBSEG for IBOT=1 (pairs of [WBSEG(i),TBSLOP(i)] specified), or i=1,2,...,(NBSEG+1) for IBOT=2 (pairs of [XBSEG(i),ZBSEG(i)] specified),

WBSEG(i) = horizontal width of the segment i

TBSLOP(i) = tangent of the slope of the segment i which is negative if the slope is downward in the landward direction.

XBSEG(i) = horizontal distance from the seaward boundary located at $x' = 0$ to the seaward end of the segment i.

ZBSEG(i) = vertical distance below SWL at the seaward end of the segment i which is *negative* if the end point is located *above* SWL.

/HYDRO/ contains the hydrodynamic quantities computed by the numerical model.

U(k,j) = components of the vector $\mathbf{U}_j$ defined in Eq. 16 at the node j such that
U(1,j) = m_j and
U(2,j) = h_j, where
m = normalized volume flux per unit width and
h = normalized water depth below the instantaneous free surface.

V(j) = normalized depth-averaged velocity $u_j = m_j/h_j$ at the node j.

ELEV(j) = normalized free surface elevation η_j above SWL at the node j.

C(j) = $c_j = \sqrt{h_j}$ defined in Eq. 11 at the node j.

DUDT(j) = normalized fluid acceleration du/dt given by Eq. 63 at the node j.

/MATRIX/ contains elements of matrices used in the numerical model. In the following, the index k=1,2.

A1(k,j) = elements of the first row of the matrix $\mathbf{A}_j$ defined by Eq. 19 at the node j such that A1(1,j)=$2u_j$ and A1(2,j)=$(h_j - u_j^2)$.

F(k,j) = elements of the vector $\mathbf{F}_j$ defined in Eq. 16 at the node j such that F(1,j)= $(m_j u_j + \frac{1}{2}h_j^2)$ and F(2,j)=m_j.

G1(j) = $[\theta_j h_j + f|u_j|u_j]$ = non-zero element of the vector $\mathbf{G}_j$ defined in Eq. 16 at the node j.

GJR(k,j) = elements of the vector $\mathbf{g}_j$ given by Eq. 18 at the node j.

S1(j) = $\left[\Delta x\, e_j - \frac{1}{2}\theta_j(m_{j+1} - m_{j-1})\right]$ = non-zero element of the vector $\mathbf{S}_j$ given by Eq. 20 at the node j where e_j is given by Eq. 21.

D(k,j) = elements of the vector $\mathbf{D}_j$ defined by Eq. 22 at the node j.

/RUNP1/ contains integers related to wave run-up.

NDELR = number of different values of the physical water depth δ_r' associated with the measured or visual waterline for which the normalized free surface elevation Z_r is computed as discussed in relation to wave run-up. It is required that $1 \leq$ NDELR $\leq$ N3 if IJOB<3. For IJOB=3, RBREAK enforces NDELR=0.

S: For IJOB<3, S is the integer s indicating the wet node next to the moving waterline such that $h_{s+1} \leq \delta < h_s$:

- at the *present* time level [t = (M-1) + (N-1)Δt] *before* Subr. 16 LANDBC is called by the the main program,
- at the *next* time level [t = (M-1) + NΔt] *after* Subr. 16 LANDBC is called by the the main program.

For IJOB=3, S=JE as discussed below.

SM1 = (s–1), used *before* Subr. 16 LANDBC is called by the the main program.

SP1 = (s+1), used *before* Subr. 16 LANDBC is called by the the main program.

JMAX = largest value of S during the time interval of the statistical calculations as explained in Section 2.4.
JMAX$\leq$JE for IJOB=1,
JMAX=JE for IJOB=2, and
JMAX=JE=S for IJOB=3.

/RUNP2/ contains quantities related to wave run-up. In the following, the index i=1,2,..., NDELR.

DELRP(i) = different values of δ_r' being specified in *millimeters* when SI is used and in *inches* when USCS is used. Each value of DELRP is independent of the others.

DELTAR(i) = normalized water depth $\delta_r = \delta_r'/H_r'$ corresponding to the different values of δ_r'. Derived from DELRP, each value of DELTAR is also independent of the others. If an armor analysis is requested (ISTAB>0), the first value, DELTAR(1), determines how far landward the armor analysis is performed. See the explanation for the related integer JSTAB in COMMON /ARMOR3/.

RUNUPS(i) = normalized instantaneous free surface elevation Z_r above SWL at the location of $h = \delta_r$, where RUNUPS(i) corresponds to DELTAR(i).

RSTAT(k,i) = mean, maximum, and minimum values of RUNUPS(i), indicated by k = 1, 2, and 3, respectively, during the time interval of the statistical calculations.

RMEAN(i) = variable used as temporary storage in the process of calculating RSTAT(1,i).

/OVER/ contains quantities related to wave overtopping.

OV(1) = normalized average overtopping rate.

OV(2) = normalized maximum overtopping rate.

OV(3) = normalized overtopping duration.

OV(4) = normalized time when OV(2) occurs.

OVMEAN = variable used as temporary storage in the process of calculating OV(1).

/COEFS/ contains the reflection and transmission coefficients.

RCOEF(k) = wave reflection coefficients defined by Eqs. 39, 40, and 41 for k = 1, 2, and 3, respectively.

TCOEF(k) = wave transmission coefficients defined by Eqs. 50, 51, and 52 for k = 1, 2, and 3, respectively.

/STT1/ contains the statistics of the hydrodynamic quantities computed during the time interval of the statistical calculations as explained in Section 2.4. In the following, the index j refers to node number with j=1,2,...,JE. Other indices are explained following their respective variables.

ELSTAT(k,i) = mean (k=1), maximum (k=2), and minimum (k=3) values of η_i (i=1), η_r (i=2), and η_t (i=3).

U1STAT(j) = mean value of m_j.

ESTAT(k,j) = mean, maximum, and minimum values of η_j indicated by k = 1, 2, and 3, respectively.

VSTAT(k,j) = mean, maximum, and minimum values of u_j indicated by k = 1, 2, and 3, respectively.

ELMEAN(i) = variable used as temporary storage in the process of calculating ELSTAT(1,i), where the index i=1,2,3 indicates the quantities explained for ELSTAT above.

U1MEAN(j) = variable used as temporary storage in the process of calculating U1STAT(j).

EMEAN(j) = variable used as temporary storage in the process of calculating ESTAT(1,j).

VMEAN(j) = variable used as temporary storage in the process of calculating VSTAT(1,j).

/STT2/ contains the parameters used to specify the time interval of the statistical calculations.

TSTAT1 = normalized time at the beginning of the time interval = t_{begin} introduced in Eq. 57.

TSTAT2 = normalized time at the end of the time interval = t_{end} introduced in Eq. 57.

/ENERG/ contains quantities related to wave energy which are computed excluding the first MSTAT computation units. In the following, the index j refers to node number with j=1,2,...,JE.

ENER(i,j) = quantities related to the time-averaged energy equation expressed by Eq. 57.
ENER(i,j) with i = 1, 2, 3, and 4 correspond to the values of $\overline{E}, \overline{E_F}, \overline{E_f}$, and $\overline{D_B}$, respectively. $\overline{E}, \overline{E_F}$, and $\overline{E_f}$ are the time-averaged values of E, E_F, and D_f given by Eqs. 54, 55, and 56, respectively, whereas $\overline{D_B}$ is computed using Eq. 57.
ENER(i,j) with i= 5 and 6 save the values of E at the time $t = t_{begin}$ and $t = t_{end}$, respectively, introduced in Eq. 57.
ENER(7,j) = value of the last term of Eq. 57.

ENERB(m) = quantities related to the time-averaged energy balance in the computation domain expressed by Eqs. 58-62. The index m=1,2,...15 corresponds to the following:
ENERB(1) = $\overline{E_F}(x = 0)$ in Eq. 58.
ENERB(2) = $\overline{E_F}(x = x_e)$ in Eq. 58.
ENERB(3) = $\int_0^{x_e} \overline{D_f} dx$ in Eq. 58.
ENERB(4) = $\int_0^{x_e} \overline{D_B} dx$ in Eq. 58.
ENERB(5) = last integral in Eq. 58.
ENERB(6) = left hand side of Eq. 58.
ENERB(7) = right hand side of Eq. 58.
ENERB(8) = difference between the right and left hand sides of Eq. 58.
ENERB(9) = percentage error defined as 100 x ENERB(8)/ENERB(6).
ENERB(10) = $\sqrt{d_t}\,\overline{\eta_i^2}$.
ENERB(11) = $\sqrt{d_t}\,\overline{(\eta_r - \overline{\eta_r})^2}$.
ENERB(12) = $\sqrt{d_e}\,\overline{(\eta_t - \overline{\eta_t})^2}$.
ENERB(13) = right hand side of Eq. 60.
ENERB(14) = percentage error in the approximate expression given by Eq. 60.
ENERB(15) = percentage error in the approximate expression given by Eq. 62.
The last two quantities may be used to estimate the uncertainties associated with the computed reflection and transmission coefficients.

ENTEMP(k,j) = variable used as temporary storage in the process of calculating ENER(k,j) at the node j with k = 1, 2, and 3.

/ARMOR1/ contains input parameters related to armor stability and movement.

C2 = area coefficient C_2 of the armor unit.

C3 = volume coefficient C_3 of the armor unit.

CD = drag coefficient C_D used in Eq. 73.

CL = lift coefficient C_L used in Eq. 74.

CM = inertia coefficient C_M used in Eq. 75.

SG = specific gravity s_g of the armor unit.

TANPHI = $\tan\phi$, with ϕ = frictional angle of the armor unit.

AMIN = parameter a_{min} specified as input. The condition for a_{min} given in Eq. 72 needs to be satisfied if ISTAB=1.

AMAX = parameter a_{max} specified as input. The condition for a_{max} given in Eq. 72 needs to be satisfied if ISTAB=1.

DAP = characteristic length d' of the armor unit given in Eq. 78 which needs to be specified as input if ISTAB=2.

/ARMOR2/ contains computed parameters related to armor stability and movement.

SG1 = (s_g–1) used in Eqs. 76 and 77.

CTAN(j) = value of ($cos\theta'_j\ tan\phi$) at the node j, where θ'_j = local angle of the slope at the node j.

/ARMOR3/ contains integers related to armor stability and movement.

JSTAB = integer indicating the largest node number for which the computation of armor stability or movement is performed. For ISTAB<3, the node number JSTAB corresponds to the normalized water depth DELTAR(1) which, in turn, corresponds to the *first* value of δ'_r. For ISTAB=3, JSTAB=JE set in Subr. 26 INIT.

JSTABM = largest value of JSTAB during the time interval of the statistical calculations.

/ARMOR4/ contains integers related to armor stability (ISTAB=1).

JSNSC = node number where the minimum armor stability occurs.

JATMIN = value of JSTAB at the time of the minimum armor stability. JSTAB is described above.

/ARMOR5/ contains parameters and values related to armor stability.

CSTAB1 = $[2C_3^{2/3}/(C_2C_D)]$ used in Eq. 66.

CSTAB2 = $[C_M/\,[(s_g-1)\sigma]]$ used in Eqs. 66 and 70.

AMAXS = ($\sigma\ a_{max}$) used in Eq. 71.

AMINS = ($\sigma\ a_{min}$) used in Eq. 71.

E2 = parameter E_2 defined by Eq. 67.

E3PRE(j) = value of $[2C_3^{2/3}\ cos\theta'\ tan\phi/(C_2C_D)]$ at the node j used in Eq. 68.

SNSC = critical stability number N_{sc} for initiation of armor movement.

TSNSC = normalized time when minimum armor stability occurs.

SNR(j) = armor stability function $N_R(t,x)$ at the node j and at the time t^* = [(M-1) + (N)Δt].

SNSX(j) = local stability number $N_{sx}(x)$ at the node j defined as the minimum value of $N_R(t,x)$ at the node j.

TSNSX(j) = normalized time when the local stability number $N_{sx}(x)$ at the node j occurs.

ATMIN(i,j) saves the following quantities at node j at the time of minimum stability:

i=1 for the normalized free surface elevation η
i=2 for the normalized depth-averaged velocity u
i=3 for the normalized horizontal fluid acceleration du/dt
i=4 for the dimensionless armor stability function N_R

/ARMOR6/ contains integers related to armor movement (ISTAB=2).

NMOVE = number of armor units moved from their initial locations where armor movement is computed excluding the first MSTAB computation units.

NSTOP = number of armor units stopped after their movement.

ISTATE(j) = integer indicating the state of the armor unit initially located at the node j, where ISTATE(j) = 0, 1, or 2, depending on whether the armor unit is stationary, moving, or stopped, respectively.

NODFI(j) = node number closest to the armor unit initially located at the node j, at the end of each time level.

/ARMOR7/ contains parameters and values related to armor movement.

CSTAB3 = $[C_2C_D/(2C_3d)]$ used in Eq. 73.

CSTAB4 = $[C_2C_L/(2C_3d)]$ used in Eq. 74.

CM1 = $(C_M - 1)$ = added mass coefficient C_m used in Eq. 81.

DA = normalized characteristic length d of the armor unit defined in Eq. 78.

SIGDA = (σ/d) used in Eq. 83a.

WEIG = $[\sigma\ (s_g - 1)]$ used in Eqs. 76 and 77.

VA(j) = normalized velocity u_a of the armor unit located initially at the node j which is computed using Eq. 81.

XAA(j) = normalized displacement X_{aa} defined by Eq. 83b of the armor unit from its initial location at the node j.

XA(j) = normalized displacement X_a defined by Eq. 83a of the armor unit from its initial location at the node j.

TDIS(j) = normalized time when the armor unit initially located at the node j started moving.

/SAVBC/ contains integers associated with options ISAVB=1 and ISAVC=1.

NNODB = number of nodes for which the time series of m and h are stored if ISAVB=1. It is required that $1 \leq$ NNODB $\leq$ N5 when ISAVB=1.

NNODC = number of nodes for which the time series of either of the following two quantities are stored if ISAVC=1.
(1) For ISTAB=1: the stability function $N_R(t,x)$ given by Eq. 69.
(2) For ISTAB=2: the displacement X_a given by Eq. 83a
It is required that $1 \leq$ NNODC $\leq$ N5 when ISAVC=1.

NODB(i) = node number for the storage of the time series for ISAVB=1. The index i runs from 1 to NNODB. It is required that $1 \leq$ NODB(i) $\leq$ JE, where JE is described in COMMON /CPAR3/.

NODC(i) = node number for the storage of the time series for ISAVC=1. The index i runs from 1 to NNODC. It is required that $1 \leq$ NODC(i) $\leq$ JE, where JE is described in COMMON /CPAR3/.

/VALUEN/ retains some of the values at the *present* time level $t = [(\mathtt{M}-1) + (\mathtt{N}-1)\Delta t]$.

VSN = u_s used for the landward boundary computation.

USN(1) = m_s used for the landward boundary computation.

USN(2) = h_s used for the landward boundary computation.

VMN = u_{s-1} used for the landward boundary computation.

UMN(1) = m_{s-1} used for the landward boundary computation.

V1N = u_1 used in Eq. 37.

V2N = u_2 used in Eq. 37.

• 3.6 •

READING THE PRIMARY INPUT DATA FILE

The contents of the primary input data file is read by Subr. 02 INPUT1. This subroutine provides clear explanations on the input parameters and variables it reads. It is recommended that a user follow the explanations in preparing the primary input data file. This section summarizes concisely the READ statements corresponding to the the primary input data file.

3.6.1 FORMATS

Input data in the primary input data file are read by Subr. 02 INPUT1 using nine FORMATs listed below.

```
1110 FORMAT (3I8)
1120 FORMAT (15A5)
1130 FORMAT (2I1)
1140 FORMAT (I1,I8,2X,A10)
1150 FORMAT (I1,3I8)
1160 FORMAT (5I1,I8)
1170 FORMAT (F15.6)
1180 FORMAT (3F13.6)
1190 FORMAT (5I6)
```

These formats are used in Sections 3.6.2 and 3.6.3.

3.6.2 HEADER

The primary input data file has a header in which a user may write pertinent information or comments. The header is read by Subr. 02 INPUT1 in the following order.

```
      READ (11,1110) NLINES
      DO 110 L = 1,NLINES
         READ (11,1120) (COMMEN(I),I=1,15)
110   CONTINUE
```

where NLINES is the number of lines containing the user's comments. Thus, the first line in the primary input data file contains the integer to be read as NLINES by Subr. 02 INPUT1. The NLINES lines containing the user's comments follow immediately. These comments are then written as the header of the output files ODOC and OMSG.

3.6.3 Input Data

Immediately following the header, input parameters and variables are read according to the formats listed in Section 3.6.1 in the order to be described below. In the following list: (1) READ statements are listed on the left, (2) Notes written on the right show where the parameter(s) or variable(s) read are explained, and (3) Sequential numerals in the middle are provided for convenience.

```
READ (11,1130) IJOB,ISTAB                          [1]     COMMON /ID/

READ (11,1130) ISYST                               [2]     COMMON /ID/

READ (11,1130) IBOT                                [3]     COMMON /ID/

READ (11,1130) INONCT                              [4]     COMMON /ID/

READ (11,1130) IENERG                              [5]     COMMON /ID/

READ (11,1140) IWAVE,MSTAT,FINP2                   [6]     COMMON /ID/
                                                           COMMON /CPAR1/
                                                           Section 3.2

READ (11,1150) ISAVA,MSAVA1,MSAVA2,NTIMES          [7]     COMMON /ID/
                                                           COMMON /CPAR1/

READ (11,1150) ISAVB,NNODB                         [8]     COMMON /ID/
                                                           COMMON /SAVBC/

READ (11,1150) ISAVC,NNODC                         [9]     COMMON /ID/
                                                           COMMON /SAVBC/

READ (11,1160) IREQ,IELEV,IV,IDUDT,ISNR,NREQ       [10]    COMMON /REQ1/

IF (IREQ.EQ.1) THEN                                        COMMON /REQ1/
   READ (11,1180) (TREQ(I),I=1,NREQ)               [11]    COMMON /REQ2/
ENDIF
```

```
      READ (11,1110) INITS                                    [12]    COMMON /CPAR3/

      READ (11,1110) MULTIF                                   [13]    COMMON /CPAR2/

      READ (11,1170) DELTA                                    [14]    COMMON /CPAR4/

      READ (11,1170) TMAX                                     [15]    COMMON /CPAR4/

      READ (11,1110) NRATE                                    [16]    COMMON /CPAR2/

      READ (11,1110) NDELR                                    [17]    COMMON /RUNP1/

      DO 130 L = 1,NDELR                                              COMMON /RUNP1/
         READ (11,1180) DELRP(L)                              [18]    COMMON /RUNP2/
130   CONTINUE

      READ (11,1180) X1,X2                                    [19]    COMMON /CPAR4/

      READ (11,1180) HREFP,TP                                 [20]    COMMON /WAVE1/

      READ (11,1180) KSREF,KSSEA                              [21]    COMMON /WAVE2/

      READ (11,1180) FWP                                      [22]    COMMON /BOT1/

      READ (11,1180) DSEAP                                    [23]    COMMON /BOT1/

      READ (11,1180) TSLOPS                                   [24]    COMMON /BOT2/

      READ (11,1110) NBSEG                                    [25]    COMMON /BOT4/

      IF (IBOT.EQ.1) THEN                                             COMMON /ID/
         DO 140 K = 1,NBSEG                                           COMMON /BOT4/
            READ (11,1180) WBSEG(K),TBSLOP(K)                 [26]    COMMON /BOT5/
140      CONTINUE
      ELSE
         DO 150 K = 1,NBSEG+1                                         COMMON /BOT4/
            READ (11,1180) XBSEG(K),ZBSEG(K)                  [27]    COMMON /BOT5/
150   CONTINUE
      ENDIF

      IF (ISAVB.EQ.1) READ (11,1190) (NODB(I),I=1,NNODB)
                                                              [28]    COMMON /ID/
                                                                      COMMON /SAVBC/

      IF (ISTAB.GT.0) THEN                                            COMMON /ID/
         READ (11,1180) C2,C3,SG                              [29]    COMMON /ARMOR1/
```

```
      READ (11,1180) CD,CL,CM                              [30]    COMMON /ARMOR1/
      READ (11,1180) TANPHI                                [31]    COMMON /ARMOR1/
      READ (11,1180) AMAX,AMIN                             [32]    COMMON /ARMOR1/
      IF (ISAVC.EQ.1) READ (11,1190) (NODC(I),I=1,NNODC)
    ENDIF                                                  [33]    COMMON /ID/
                                                                   COMMON /SAVBC/

    IF (ISTAB.EQ.2) READ (11,1180) DAP                     [34]    COMMON /ID/
                                                                   COMMON /ARMOR1/
```

• 3.7 •

Warning and Error Messages

Warning and error messages are written in the file ODOC and displayed on screen. There is only one warning message that the computer program RBREAK may issue, that is,

```
From Subr.  07 FINDM:
Criterion for parameter m=MCNO not satisfied
```

The above warning is related to the iteration scheme to compute the parameter m using Eq. 36, and thus corresponds to the case of regular cnoidal waves only. This warning has never been experienced by the authors. RBREAK does not automatically cease computation when this warning is issued. The results obtained under this circumstance, however, may not be reliable.

Computation is terminated immediately following any error message. If the preceding BEFORR computation is successful, serious potential errors in an RBREAK execution are likely limited to those related to the dimensions of the vectors and matrices, *i.e.*, the PARAMETERs N1, N2, N3, N4, and N5 explained in COMMON /DIMENS/ in Section 3.5. Here is a list of error messages related to the dimensions. The error messages are self-explanatory.

1. If the value of any of the PARAMETERs N1, N2, N3, N4, and N5 in a subroutine that utilizes the PARAMETERs does not match with the corresponding value specified in the main program:

```
PARAMETER Error:  N· = ··· in Subroutine ···
Correct Value:  N· = ···
```

2. If the requirement $1 \leq \text{INITS} \leq (\text{N1} - 1)$ is not satisfied:

```
Input Error:  INITS = ···
Specify INITS in the range of [1,···]
Change PARAMETER N1 if necessary
```

3. If JE > N1:

```
End Node = ···; N1 = ···
Slope/Structure is too long.
Cut it, or change PARAMETER N1.
```

4. If the requirement $1 \leq$ NDATA $\leq$ N2 when IWAVE>1 is not satisfied:

```
Input Error:  NDATA = ···
Specify NDATA in the range of [1,···]
Change PARAMETER N2 if necessary
```

5. If the requirement $1 \leq$ NDELR $\leq$ N3 when IJOB<3 is not satisfied:

```
Input Error:  NDELR = ···
Specify NDLER in the range of [1,···]
Change PARAMETER N3 if necessary
```

6. If the requirement $1 \leq$ NBSEG $\leq$ (N4 $-$ 1) is not satisfied:

```
Input Error:  NBSEG = ···
Specify NBSEG in the range of [1,···]
Change PARAMETER N4 if necessary
```

7. If the requirement $1 \leq$ NREQ $\leq$ N5 when IREQ=1 is not satisfied:

```
Input Error:  NREQ = ···
Specify NREQ in the range of [1,···]
Change PARAMETER N5 if necessary
```

8. If the requirement $1 \leq$ NNODB $\leq$ N5 when ISAVB=1 is not satisfied:

```
Input Error:  NNODB = ···
Specify NNODB in the range of [1,···]
Change PARAMETER N5 if necessary
```

9. If the requirement $1 \leq$ NNODC $\leq$ N5 when ISAVC=1 is not satisfied:

```
Input Error:  NNODC = ···
Specify NNODC in the range of [1,···]
Change PARAMETER N5 if necessary
```

10. If the requirement $1 \leq$ NODB(i) $\leq$ JE when ISAVB=1 is not satisfied:

```
Input Error:  NODB = ···
Specify NODB in the range of [1,···]
```

11. If the requirement $1 \leq$ NODC(i) $\leq$ JE when ISAVC=1 is not satisfied:

```
Input Error:  NODC = ···
Specify NODC in the range of [1,···]
```

12. For the case of regular waves, the file BINPUT also contains NONEMA = the largest value of NONE used in the preceding BEFORR computation, to be compared with the value of PARAMETER N2 of the computer program RBREAK. It is required that N2>NONEMA. The following message is issued if this condition is not met:

```
From Subr.  O2 INPUT1:
Computation will need NONE as large as ···
Dimension N2 must be greater than NONE.
Currently, N2 = ···
Specify larger N2 as required.
Computation aborted.
```

The parameters and variables involved in the above error messages are explained in Section 3.5 as indicated below.

VARIABLE	DESCRIPTION IN SECTION 3.5
N1,N2,N3,N4,N5	COMMON /DIMENS/
NONE	COMMON /CPAR2/
INITS,JE	COMMON /CPAR3/
NREQ	COMMON /REQ1/
NDATA	COMMON /WAVE6/
NBSEG	COMMON /BOT4/
NDELR	COMMON /RUNP1/
NODB,NODC,NNODB,NNODC	COMMON /SAVBC/

Less serious errors and their causes are listed in the following. Error no. (1) is a simple error. Errors no. (2) to (5) indicate that the case-signatures from the file BINPUT do not match with the corresponding values in RBREAK. Error no. (6) should never happen if the file BINPUT has not been altered.

1. If MREP<0:

```
MREP must be positive or zero.
Programmed Stop.
```

2. If the case-signature MWAV does not match with MWAVE:

```
Computation duration does not match.  Check BINPUT.
Programmed Stop.
```

3. If the case-signature FID2 does not match with FINP2:

```
Second input file name does not match.  Check BINPUT.
Programmed Stop.
```

4. If the case-signature NDAT does not match with NDATA:

```
Number of data points for input wave train
does not match.  Check BINPUT.
Programmed Stop.
```

5. If the case-signature NONM does not match with NONEM:

```
Minimum allowable NONE does not match.  Check BINPUT.
Programmed Stop.
```

6. If the present computation unit M does not match with the sequential number in the file BINPUT:

```
Computation unit not matched.  Check BINPUT.
Programmed Stop.
```

The variables involved in the above error messages are explained previously as indicated below.

VARIABLE	DESCRIPTION
MREP	Described in Section 3.3.1
MWAVE=MWAV	Described in COMMON /CPAR1/ in Section 3.5
FINP2=FID2	Described in Section 3.2
NDATA=NDAT	Described in COMMON /WAVE6/ in Section 3.5
NONEM=NONM	Described in COMMON /CPAR2/ in Section 3.5
M	Described in Section 3.3.4

The computer program RBREAK contains other error messages which are highly unlikely to occur if the duo BEFORR-RBREAK are executed properly. These messages are put in RBREAK to provide programmed escape routes just in case that something unexpected occurs. These error messages are self-explanatory.

• 3.8 •
OUTPUT PRODUCED BY RBREAK

Output from the computer program RBREAK is stored in files whose names start with the letter "O" for easy identification. The number of output files varies depending on the options selected by a user. Table 3 lists the names of all possible output files generated by RBREAK. The first two files in Table 3, ODOC and OMSG, contain information which should be read and checked by the user. The rest contain the computed results that may need further processing to yield useful information for the user. The files OSTAT, OSPACE, OREQ, and OSTAB contain spatial variations. The files OSEAWAV, ORUNUP, OOVER, and OTRANS contain time series covering the time interval $0 < t \leq$ TMAX and so do the families of files OSAV1Bxx, OSAV2Bxx, and OSAVECxx.

Except for the contents of the files ODOC and OMSG, all other output values are written using only two kinds of FORMATs, that is, FORMAT 8000 for double precision values and FORMAT 9000 for integers.

```
8000 FORMAT (5D15.6)
9000 FORMAT (8I8)
```

FORMATs 8000 and 9000 appearing in the WRITE statements in the subsequent sections refer to the above specifications. In addition, the variables involved in the WRITE statements are tabulated with concise descriptions in the following.

3.8.1 GENERAL

The following output files are always generated by RBREAK.

Table 3: List of output files generated by RBREAK.

Output File Name	Condition for Creation	Discussed in Section
ODOC OMSG OSTAT OSPACE OSEAWAV	No Condition	3.8.1
ORUNUP	IJOB<3	3.8.2
OOVER	IJOB=2	3.8.2
OTRANS	IJOB=3	3.8.3
OREQ	IREQ=1	3.8.4
OSTAB	ISTAB>0	3.8.5
OSAV1B01 OSAV1B02 . . and so on	ISAVB=1	3.8.6
OSAV2B01 OSAV2B02 . . and so on	ISAVB=1	3.8.6
OSAVEC01 OSAVEC02 . . and so on	ISAVC=1 and ISTAB>0	3.8.6

File ODOC contains a summary of input data and computed results. The contents of this file can be seen in the examples given in Appendices C through G.

File OMSG contains warning and error messages explained in Section 3.7. There should be no error message generated by a successful RBREAK computation.

File OSTAT stores the statistics of hydrodynamic quantities. The contents of this file are written in Subr. 45 DOC3 in the following order:

```
      WRITE (51,9000) JMAX
      WRITE (51,8000) (U1STAT(J),J=1,JMAX)
      DO 120 I = 1,3
         WRITE (51,8000) (ESTAT(I,J),J=1,JMAX)
         WRITE (51,8000) (VSTAT(I,J),J=1,JMAX)
120   CONTINUE
```

VARIABLE	DESCRIPTION IN SECTION 3.5
JMAX	COMMON /RUNP1/
U1STAT,ESTAT,VSTAT	COMMON /STT1/

File OSPACE stores the normalized structure geometry written in Subr. 43 DOC1 as follows:

```
      WRITE (52,9000) JE
      WRITE (52,8000) (XB(J),ZB(J),J=1,JE)
```

In addition, if ISAVA=1, Subr. 44 DOC2 writes the following output [NTIMESx(MSAVA2-MSAVA1+1)] times.

```
      WRITE (52,9000) S
      WRITE (52,8000) (ELEV(J),V(J),J=1,S),TIME
```

VARIABLE	DESCRIPTION IN SECTION 3.5
ISAVA	COMMON /ID/
NTIMES,MSAVA1,MSAVA2	COMMON /CPAR1/
JE	COMMON /CPAR3/
TIME	COMMON /CPAR4/
XB,ZB	COMMON /BOT3/
S	COMMON /RUNP1/
ELEV,V	COMMON /HYDRO/

File OSEAWAV stores the time series of hydrodynamic quantities at the seaward boundary. The following four quantities are stored at the rate of NRATE data points per wave period in Subr. 44 DOC2:

```
WRITE (61,8000) TIME,ETAI,ETAR,U(1,1)
```

VARIABLE	DESCRIPTION
NRATE	Described in COMMON /CPAR2/ in Section 3.5
TIME	Described in COMMON /CPAR4/ in Section 3.5
ETAI	= η_i at $x = 0$ at t=TIME
ETAR	= η_r at $x = 0$ at t=TIME
U(1,1)	= m at $x = 0$ at t=TIME

3.8.2 SUBAERIAL STRUCTURES (IJOB<3)

If IJOB<3, the file ORUNUP is produced. In addition, if IJOB=2, the file OOVER is also created.

File ORUNUP contains the time series of quantities related to run-up. The time series are written in Subr. 44 DOC2 at the rate of NRATE data points per wave period.

```
WRITE (62,8000) DBLES,(RUNUPS(L),L=1,NDELR),TIME
```

VARIABLE	DESCRIPTION
IJOB	Described in COMMON /ID/ in Section 3.5
NRATE	Described in COMMON /CPAR2/ in Section 3.5
TIME	Described in COMMON /CPAR4/ in Section 3.5
NDELR,S	Described in COMMON /RUNP1/ in Section 3.5
RUNUPS	Described in COMMON /RUNP2/ in Section 3.5
DBLES	= DBLE(S) at t=TIME

File OOVER contains the time series of hydrodynamic quantities at the landward-end node written in Subr. 44 DOC2 at the rate of NRATE data points per wave period.

```
WRITE (63,8000) U(1,JE),U(2,JE)
```

VARIABLE	DESCRIPTION
NRATE	Described in COMMON /CPAR2/ in Section 3.5
U(1,JE)	= m at $x = x_e$
U(2,JE)	= h at $x = x_e$

3.8.3 Submerged Structures (IJOB=3)

File OTRANS is created if IJOB=3. It contains the time series of hydrodynamic quantities at the landward-end node written in Subr. 44 DOC2 at the rate of NRATE data points per wave period.

```
WRITE (64,8000) U(1,JE),U(2,JE),ETAT,TIME
```

Variable	Description
IJOB	Described in COMMON /ID/ in Section 3.5
NRATE	Described in COMMON /CPAR2/ in Section 3.5
TIME	Described in COMMON /CPAR4/ in Section 3.5
U(1,JE)	= m at $x = x_e$ at t=TIME
U(2,JE)	= h at $x = x_e$ at t=TIME
ETAT	= η_t at $x = x_e$ at t=TIME

3.8.4 Special Request (IREQ=1)

File OREQ is reserved for the special request in the computer program RBREAK to store the spatial variations of certain quantities at designated moments. The designated moments are the normalized time TREQ(i) with i=1,2,...NREQ. The special request is activated by specifying IREQ=1. Quantities available for the request are stored if η, u, du/dt, and N_R, which are stored if IELEV=1, IV=1, IDUDT=1, and ISNR=1, respectively. Note that the spatial variation of N_R can be requested only if ISTAB=1. The requested variations at the designated time are written in the file OREQ in Subr. 44 DOC2 as follows:

```
WRITE (55,9000) S
WRITE (55,8000) TIME
IF (IELEV.EQ.1) WRITE (55,8000) (ELEV(J),J=1,S)
IF (IV.EQ.1)    WRITE (55,8000) (V(J),J=1,S)
IF (IDUDT.EQ.1) WRITE (55,8000) (DUDT(J),J=1,S)
IF (ISNR.EQ.1)  WRITE (55,8000) (SNR(J),J=1,S)
```

Variable	Description in Section 3.5
ISTAB	COMMON /ID/
TIME	COMMON /CPAR4/
IREQ,NREQ,IELEV,IV,IDUDT,ISNR	COMMON /REQ1/
TREQ	COMMON /REQ2/
ELEV,V,DUDT	COMMON /HYDRO/
S	COMMON /RUNP1/
SNR	COMMON /ARMOR5/

3.8.5 Armor Analysis (ISTAB>0)

File OSTAB is created if the analysis of either armor stability (ISTAB=1) or armor movement (ISTAB=2) is performed.

For ISTAB=1, Subr. 45 DOC3 writes the following:

```
      WRITE (54,9000) JSTABM
      DO 140 J = 1,JSTABM
         WRITE (54,8000) XB(J),ZB(J),SNSX(J),TSNSX(J)
  140 CONTINUE
      WRITE (54,9000) JATMIN
      WRITE (54,8000) TSNSC
      DO 150 I = 1,4
         WRITE (54,8000) (ATMIN(I,J),J=1,JATMIN)
  150 CONTINUE
```

For ISTAB=2, Subr. 45 DOC3 writes the following:

```
      WRITE (54,9000) NMOVE
      WRITE (54,9000) NSTOP
      WRITE (54,9000) JSTABM
      DO 160 J = 1,JSTABM
         IF (ISTATE(J).GE.1) THEN
            WRITE (54,9000) J,NODFI(J),ISTATE(J)
            WRITE (54,8000) XB(J),ZB(J),XA(J),TDIS(J)
         ENDIF
  160 CONTINUE
```

Variable	Description in Section 3.5
ISTAB	COMMON /ID/
XB,ZB	COMMON /BOT3/
JSTABM	COMMON /ARMOR3/
JATMIN	COMMON /ARMOR4/
SNSX,TSNSX,TSNSC,ATMIN	COMMON /ARMOR5/
NMOVE,NSTOP,NODFI,ISTATE	COMMON /ARMOR6/
XA,TDIS	COMMON /ARMOR7/

3.8.6 Nodal Time Series (ISAVB=1 and ISAVC=1)

The user has the option to obtain the time series of m and h at specified nodes by specifying ISAVB=1. The time series are stored in groups of 100 waves to avoid creating too large an output file. The rate of storing is NRATE data points per wave period. The values at all specified nodes are stored simultaneously in one file at the storage time. The output files are named as follows:

OSAV1Bxx for m, and
OSAV2Bxx for h.

where xx=01 for the file containing the first 100 waves, xx=02 for the file containing the second 100 waves, and so on.

The nodes for the storage of the time series are specified by the node number NODB(i) with i=1,2,...,NNODB. Writing is performed in Subr. 44 DOC2 at each storage time as follows:

```
      DO 110 I = 1,NNODB
         J = NODB(I)
         VONE(I) = U(1,J)
         VTWO(I) = U(2,J)
  110 CONTINUE
      WRITE (71,8000) (VONE(I),I=1,NNODB),TIME
      WRITE (72,8000) (VTWO(I),I=1,NNODB)
```

In addition, if ISTAB>0, the time series of either N_R (for ISTAB=1) or X_a (for ISTAB=2) at specified nodes can be stored by specifying ISAVC=1. The method of storing these time series is similar to that of the option ISAVB=1. The specified nodes are at the node number NODC(i) with i=1,2,...,NNODC. The output file names are OSAVECxx.

The following is written in Subr. 44 DOC2 at each storage time.

```
      IF (ISTAB.EQ.1) THEN
         DO 120 I = 1,NNODC
            J = NODC(I)
            VONE(I) = SNR(J)
  120    CONTINUE
      ELSE
         DO 130 I = 1,NNODC
            J = NODC(I)
            IF (ISTATE(J).EQ.0) THEN
               VONE(I) = 0.D+00
            ELSE
               VONE(I) = XA(J)
            ENDIF
  130    CONTINUE
      ENDIF
      WRITE (73,8000) (VONE(I),I=1,NNODC),TIME
```

VARIABLE	DESCRIPTION IN SECTION 3.5
ISTAB,ISAVB,ISAVC	COMMON /ID/
NRATE	COMMON /CPAR2/
TIME	COMMON /CPAR4/
U	COMMON /HYDRO/
SNR	COMMON /ARMOR5/
XA	COMMON /ARMOR7/
NODB,NODC,NNODB,NNODC	COMMON /SAVBC/

The present setting of the computer program RBREAK enables the storage of the time series associated with ISAVB=1 and ISAVC=1 of up to 2000 waves. If a larger storage is needed, the dimension of the variables FSAV1B, FSAV2B, and FSAVEC in Subr. 01 OPENIO, which is presently set at 20, has to be increased. Consequently, the statements DATA FSAV1B, DATA FSAV2B, and DATA FSAVEC in Subr. 01 OPENIO need to be adjusted accordingly.

PART IV
COMPUTER PROGRAM BEFORR

• 4.1 •
INTRODUCTION

Section 3.1 describes the reason behind the creation of the computer program BEFORR, which is listed in Appendix B. To be compatible with RBREAK, the computer program BEFORR is also written in standard FORTRAN-77 with full double precision mode.

As shown in Fig. 4, BEFORR and RBREAK read the same primary input data file and, for the case of irregular waves, the same input wave train. The computer program BEFORR reads the primary input data file in the same way as the computer program RBREAK, that is, following the sequence and formats listed in Section 3.6.3. It is BEFORR that checks most of the contents of the primary input data file. RBREAK checks only the dimensions of the vectors and matrices. The dimensions of the vectors and matrices `N1`, `N2`, `N3`, `N4`, and `N5` for BEFORR and RBREAK are independently specified in each program, and hence can be different. In the present setting, as listed in Appendices A and B, the values of `N1`, `N2`, `N3`, `N4`, and `N5` for both programs are identical, that is, `N1`=800, `N2`=16000, `N3`=3, `N4`=40, `N5`=40. The other checks are not repeated in RBREAK, assuming that the primary input data file is not altered after a successful BEFORR computation. A list of the requirements enforced by BEFORR is presented in Section 4.5 along with warning and error messages.

BEFORR produces two kinds of computation logs. The first log lists all the numerical problems chronologically as explained in Section 4.2.2. This log is written in the file `BMSG` as well as on screen. The second computation log is written in the file `BDOC` and includes the computation parameters and indicators for each successful computation unit. Some of these parameters and indicators are written on screen every `MREP` wave periods as explained in Section 3.3.1. Together, the two computation logs provide concise output on how the hydrodynamic computation has been executed by BEFORR for each case.

Part IV of this report is written in such a way that items discussed already in Part III are referred to the appropriate Sections of Part III, assuming the user is familiar with RBREAK.

• 4.2 •
MAIN PROGRAM BEFORR

The computer program BEFORR contains essentially the hydrodynamic part of the computer program RBREAK. BEFORR examines the hydrodynamic computation and obtains the

appropriate computation parameters for each computation unit to avoid numerical problems in the hydrodynamic computation for that unit if possible. The main program BEFORR includes a mechanism for adjusting the computation parameters for each computation unit automatically. However, the adjustment may not always be successful in the first attempt as explained in Section 4.2.1.

BEFORR and RBREAK are organized in a similar manner. The tasks performed by the main program of BEFORR can also be categorized into five groups: (1) Reading input data, (2) Checking FORTRAN PARAMETERs in the subroutines, (3) Groundwork, (4) Time-marching computation, and (5) Finishing. The tasks of the groups (1), (2), (3), and (5) for BEFORR are similar to those for RBREAK as explained in Section 3.3.

The self-adjusting mechanism of BEFORR to overcome numerical difficulties during the time-marching computation is explained in detail in the rest of this section. The numerical method for the hydrodynamic computation from one time level to next is the same as that for RBREAK explained in Section 3.3.4.

4.2.1 Three Crucial Computation Parameters

The numerical stability and accuracy of the computation are mainly determined by the following three parameters:

1. INITS = number of the spatial nodes below SWL. A user specifies the value of INITS. INITS should be so large that the spacing between two adjacent nodes is small enough to resolve the spatial variation of the wave motion. INITS=100 to 600 has been used. The examples given in the Appendices C through G suggest the typical values of INITS which have been used so far. A larger value of INITS is desirable for the accuracy of computation, but leads to a longer computation time.

2. NONE = number of time steps in one wave period. NONE is the inverse of the time step Δt, as expressed in Eq. 85. The following three requirements are imposed on the selection of an appropriate value of NONE.

   (a) The numerical stability criterion expressed by Eq. 29 with Eq. 28 needs to be satisfied.

   (b) For storing the computed time series at the rate of NRATE data points per wave period, NONE should be divisible by NRATE.

   (c) For regular wave computations, NONE should be an even number for the symmetry about the crest of the assumed Stokes second-order wave or cnoidal wave profile.

   Denoting the smallest integer satisfying the above three requirements by $\mathcal{N}$ and introducing an integer multiplication factor MULTIF, the minimum allowable value NONEM of NONE is given by

   $$\text{NONEM} = \text{MULTIF} \times \mathcal{N} \tag{86}$$

   Starting from NONEM, the computer program BEFORR attempts to find the value of NONE for each computation unit which does not cause numerical problems. The integer MULTIF should be unity or greater and may be used to adjust the accuracy of the computation since the increase of MULTIF results in the increase of NONE and hence the decrease of the time step Δt. It is suggested to increase MULTIF from unity if the computed results show unrealistic spikes and kinks.

3. DELTA = normalized water depth used to specify the computational waterline. An *initial* value of DELTA needs to be specified by a user. For a regular wave computation, the value of DELTA supplied by the user is not modified. For the case of irregular waves, BEFORR may reduce the initial value of DELTA for subsequent computation units if it is deemed necessary to overcome numerical problems. The values of DELTA ranging from 0.001 to 0.003 have been used. It is suggested that the user specify DELTA=0.001 first. If the resulting computation fails or experiences too many trial runs to overcome numerical difficulties, the value of DELTA should be increased to 0.002 or 0.003.

4.2.2 Potential Numerical Problems in Hydrodynamic Computation

There are three potential numerical problems that may be encountered in the hydrodynamic computation. The three problems are identified by IPROB=1, IPROB=2, and IPROB=3. If any of the three problems occurs in BEFORR, pertinent log messages will be written both in the file BMSG and on terminal screen. If IPROB is not equal to zero, the computation for each computation unit is resumed from the beginning of the unit using a larger value of NONE or a smaller value of DELTA as explained in Section 4.2.3.

1. IPROB=1 identifies negative water depth which may occur at the computational waterline. Subr. 14 CRITV checks whether this problem occurs, and issues the following log message if it occurs.

```
      ICOUNT = 0
      DO 100 J = 1,S
         IF (U(2,J).LT.0.D+00) THEN
            ICOUNT = ICOUNT + 1
         ELSE
            C(J) = DSQRT(U(2,J))
         ENDIF
  100 CONTINUE
      IF (ICOUNT.GT.0) THEN
         IPROB=1
         CALL NSI (OMEGA,NONE)
         WRITE (*,9910) M,S,NONE,OMEGA,TIME,ICOUNT,ITRY,IREV
         WRITE (99,9910) M,S,NONE,OMEGA,TIME,ICOUNT,ITRY,IREV
      ENDIF
 9910 FORMAT (/' From Subr.  14 CRITV: IPROB=1'/
     + ' (Negative water depth)'/
     + ' Computation unit              M =',I8/
     + ' Waterline node                S =',I8/
     + ' Computation parameters:    NONE =',I8/
     + '                           OMEGA =',F11.2/
     + ' Time of occurrence         TIME =',F18.9/
     + ' Number of occurrence     ICOUNT =',I8/
     + ' Indicators             ITRY,IREV =',I4,',',I3)
```

2. IPROB=2 identifies the seaward movement of the computational waterline in excess of one grid spacing over one time step, which violates the numerical stability criterion expressed by Eq. 27. Subr. 15 MARCH checks whether this problem occurs, and issues the following log message if it occurs.

```
      IF (U(2,SM1).LE.DELTA) THEN
         IPROB=2
         U2STOP=U(2,SM1)
         CALL NSI (OMEGA,NONE)
         WRITE (*,9910) M,S,NONE,OMEGA,DELTA,U2STOP,TIME,ITRY,IREV
         WRITE (99,9910) M,S,NONE,OMEGA,DELTA,U2STOP,TIME,ITRY,IREV
         RETURN
      ENDIF
 9910 FORMAT (/' From Subr.  15 MARCH: IPROB=2'/
     + ' (Water depth at (S-1) <or= DELTA)'/
     + ' Computation unit              M =',I8/
     + ' Waterline node                S =',I8/
     + ' Computation parameters:    NONE =',I8/
     + '                           OMEGA =',F11.2/
     + '                           DELTA =',F18.9/
     + ' Water depth at (S-1)  U(2,S-1) =',F18.9/
     + ' Time of occurrence         TIME =',F18.9/
     + ' Indicators            ITRY,IREV =',I4,',',I3)
```

3. IPROB=3 identifies the occurrence of $|u_j| > (\Delta x/\Delta t)$ which implies the violation of the the numerical stability criterion given by Eq. 27. Subr. 11 NUMSTA checks whether this problem occurs, most likely at the computational waterline, and issues the following log message if it occurs.

```
      ICOUNT = 0
      DO 100 J = 1,S
         IF (DABS(V(J)).GT.XT) ICOUNT=ICOUNT+1
  100 CONTINUE
      IF (ICOUNT.GT.0) THEN
         IPROB=3
         CALL NSI (OMEGA,NONE)
         WRITE (*,9910) M,S,NONE,OMEGA,TIME,ICOUNT,ITRY,IREV
         WRITE (99,9910) M,S,NONE,OMEGA,TIME,ICOUNT,ITRY,IREV
      ENDIF
 9910 FORMAT (/' From Subr.  11 NUMSTA: IPROB=3'/
     + ' (Stability criterion violated)'/
     + ' Computation unit              M =',I8/
     + ' Waterline node                S =',I8/
     + ' Computation parameters:    NONE =',I8/
     + '                           OMEGA =',F11.2/
     + ' Time of occurrence         TIME =',F18.9/
     + ' Number of occurrence     ICOUNT =',I8/
     + ' Indicators            ITRY,IREV =',I4,',',I3)
```

The variables involved in these messages are summarized below.

VARIABLE	DESCRIPTION
TIME,XT	Described in COMMON /CPAR4/ in Section 3.5
S	Described in COMMON /RUNP1/ in Section 3.5
U,V,C	Described in COMMON /HYDRO/ in Section 3.5
SM1	= (S-1)
U2STOP	= U(2,SM1)
M	= Computation unit
OMEGA	= Ω given by Eq. 28
NONE	Described in Section 4.2.1
DELTA	Described in Section 4.2.1
ITRY,IREV	Described in COMMON /DIAGID/ in Section 4.4
ICOUNT	Counter counting the number of occurence of numerical problems.

Subr. 25 NSI is called to calculate the numerical stability indicator OMEGA = Ω given by Eq. 28. The log messages, which are written both on terminal screen and in the file BMSG, can be used to assess the numerical difficulties experienced by the BEFORR computation.

4.2.3 Assignment of Values to NONE and DELTA

In a BEFORR computation, the value of the normalized nodal spacing Δx is determined at the beginning of the computation based on the value of INITS supplied by a user. The value of Δx is fixed during the entire computation duration. The value of NONE, which determines the size of the time step Δt, may vary from one computation unit to another. In the case of regular waves, the value of DELTA supplied by the user is used throughout the entire computation duration. In the case of irregular waves, on the other hand, the value of DELTA may be reduced in subsequent computation units as explained in the following.

It is the job of the computer program BEFORR to seek the values of NONE and DELTA that cause no numerical problem for each computation unit. The computation units are examined in sequence. In the first trial run for each computation unit, the values of the pair (NONE,DELTA) are assumed in a certain manner. If there is no numerical problem encountered, the assumed values are adopted. Otherwise, additional trial runs are made by increasing the value of NONE or reducing the value of DELTA incrementally until there is no numerical problem for the computation unit under examination. The number of trial runs for each computation unit is counted by the counter ITRY.

Subr. 10 CPARAM determines the values of NONE and DELTA in each trial run for each computation unit according to the following guidelines:

1. For the first trial run for the first computation unit, use NONE=NONEM and DELTA as specified in the primary input data file.
2. For the second or subsequent trial run because of IPROB = 1, 2, or 3 in the previous trial run for any computation unit:
   (a) For regular wave computations, increase NONE by about 50%. DELTA is not modified.
   (b) For irregular wave computations:
      (b1) If IPROB=2 and $h_{s-1} > 0$: decrease DELTA by about 5%. NONE is not modified.
      (b2) Otherwise, increase NONE by about 50%. DELTA is not modified.
3. For the first trial run for the second and subsequent computation units:
   (a) For regular wave computations, use the value of NONE that has been assigned to the previous computation unit. DELTA is never modified.
   (b) For irregular wave computations: use the value of DELTA that has been assigned to the previous computation unit and choose NONE as follows:

$$\text{NONE}=(\text{IREV}+1)\times\text{NONEM} \tag{87}$$

   where the reverse indicator IREV=0, unless IPROB=9 as explained in the following.
4. It can happen that NONE has been increased several times but numerical problems still occur. There is an upper limit for the value of NONE. The limit is set in terms of the numerical stability indicator Ω given by Eq. 28 which implies that Ω increases with NONE. Whenever $\Omega > 20$, IPROB=9 is set, and additional measures are taken as explained in Section 4.2.4.

5. The following requirements are always imposed:

   (a) NONE is divisible by NRATE.

   (b) DELTA is a multiple (10^{-6}).

   (c) Each trial run starts at the beginning of a computation unit using the stored hydrodynamic quantities at the end of the previous computation unit (LEVEL=1 as explained below).

It should be stated that the above guidelines are based on the experiences gained from various computations made using the computer programs BEFORR-RBREAK. The values of "50%" (increase of NONE), "5%" (decrease of DELTA), and "20" (the upper limit on Ω) could be changed according to the experiences of the user.

4.2.4 Computation Reversal and Storage Time Levels

IPROB=9 indicates that NONE has become so large that the corresponding value of the numerical stability indicator Ω is greater than 20 but numerical problems still exist. Further increases of the value of NONE will usually be ineffective. For *regular* wave computations, this is the end of a BEFORR computation. The following error message, written in Subr. 10 CPARAM, will appear on screen as well as in the file BMSG.

```
OMEGA is too large.  Computation aborted.
Suggested to input larger DELTA.
```

Another attempt may be carried out using a larger initial value of DELTA as explained in Section 4.2.1.

Experiences suggest that for the case of *irregular* waves, the problem IPROB=9 may still be overcome by reversing the computation to a certain time level, setting the counter IREV=IREV+1 and resume the computation. The increased value of IREV will result in the increased value of NONE for the first trial run for the subsequent computation units because of Eq. 87. This computation reversal requires the storage of the hydrodynamic quantities including DELTA at past time levels. To minimize memory requirements and simplify the computational procedures, hydrodynamic quantities are stored at the following three time levels.

- LEVEL=1: the end the *previous* computation unit corresponding to the unit (M-1) where the integer M indicates the present computation unit.
- LEVEL=2: the beginning of the *present* set of five computation units indicated by the integer MM where the computation units are grouped into sets of five units. The last set may contain less than five units. The total number of sets is denoted by MFIVE.
- LEVEL=3: the beginning of the *previous* set of five computation units corresponding to the set (MM-1).

The hydrodynamic quantities at LEVEL=1, LEVEL=2, and LEVEL=3 are stored in the following variables with the index k = 1, 2, and 3, respectively.

- SSAV(k) stores S.
- JMSAV(k) stores JMAX.
- DELSAV(k) stores DELTA.
- VSAV(k,j) stores V(j) at the node j.
- ESAV(k,j) stores ELEV(j) at the node j.
- USAV(2k-1,j) stores U(1,j) at the node j.
- USAV(2k,j) stores U(2,j) at the node j.

The stored variables are explained in the following sections:

VARIABLE	DESCRIPTION
DELTA	Described in COMMON /CPAR4/ in Section 3.5
U,V,ELEV	Described in COMMON /HYDRO/ in Section 3.5
S	Described in COMMON /RUNP1/ in Section 3.5
JMAX	Described in COMMON /RUNP1/ in Section 4.4

These values are stored using Subr. 29 SAVEM for LEVEL=1 and Subr. 30 SAVEMM for LEVEL=2 and LEVEL=3. On the other hand, the stored values are recalled by Subr. 27 INITM for LEVEL=1 and Subr. 28 INITMM for LEVEL=2 and LEVEL=3.

It should be noted that the storing and recalling procedures for LEVEL=1 are necessary and used for trial runs for each computation unit to overcome the numerical problems of IPROB = 1, 2, and 3 discussed in Section 4.2.3.

For the case of IPROB=9, the following measures based on LEVEL=2 and LEVEL=3 are taken:

1. Increase IREV by one. The counter IREV indicates the number of reversals made because of IPROB=9. IREV also increases the value of NONE given by Eq. 87 for the first trial run for the subsequent computation units.
2. If any of the following three conditions is met, reverse the computation to LEVEL=2.
   (a) MM=1, implying that the computation is in the *first* set of five computation units and there is no storage at LEVEL=3 yet.
   (b) ILEV3=1, which indicates that the reversal to LEVEL=3 has already been made and there is no storage at new LEVEL=3 yet.
   (c) The time interval between the end of the present computation unit and the time level of LEVEL=2 is less than 2 wave periods.
3. Otherwise, reverse the computation to LEVEL=3, set ILEV3=1, and substract one from the index MM.

4. Set initial hydrodynamic quantities and DELTA using Subr. 28 INITMM, which differentiate LEVEL=2 and LEVEL=3.

5. Set IPROB=0 and march the computation forward from the designated level of LEVEL=2 or LEVEL=3.

6. If the computation reaches the end of the present set of five computation units without IPROB=9, set IREV=0 and ILEV3=0, which indicate that the problem of IPROB=9 is over and the computation proceeds from one computation unit to the next.

7. If the reversals are made five times (IREV=5), the computation is terminated with the following error message:

```
Computation aborted after IREV reached 5.
Suggested to input larger DELTA.
Programmed stop.
```

• 4.3 •

Subroutines and Function

The computer program BEFORR consists of 41 program units: the main program, 39 subroutines, and one function. Table 4 lists the subroutines and function arranged in numerical order in the computer program BEFORR. The page numbers for the subroutines and function listed in Table 4 correspond to the page numbers in the BEFORR listing presented in Appendix B. Table 5 shows the interdependence among the 41 program units.

The computer programs BEFORR and RBREAK share many subroutine names. The subroutines with the same names do similar or identical tasks. The explanations given in Section 3.4 generally apply to the corresponding subroutines in BEFORR. Explained in the following are only the subroutines that (1) do not exist in RBREAK, or (2) are modified noticeably from the corresponding subroutines in RBREAK. These subroutines are explained in the format: Number – NAME – Description, where the Number refers to the numerical order in the computer program BEFORR. The subroutines and function that are not listed below have already been explained in Section 3.4.

01 OPENIO opens input and output files.

02 INPUT1 reads information from the primary input data file and performs checks on the majority of input data.

10 CPARAM calculates the minimum value of NONE *before* the time-marching computation. *During* the time-marching computation, this subroutine assigns appropriate values to the computation parameters NONE and DELTA for each computation unit following the procedures described in Section 4.2.3. NONE is the number of time steps in one wave period, whereas DELTA is the normalized water depth defining the computational waterline.

Table 4: List of 39 subroutines and one function in computer program BEFORR.

No.	Subroutine (S) or Function (F)		Page No. in BEFORR	No.	Subroutine (S) or Function (F)		Page No. in BEFORR
01	S	OPENIO	B-6 - B-7	21	S	MATGJR	B-40
02	S	INPUT1	B-7 - B-15	22	S	MATS	B-41
03	S	INPUT2	B-15	23	S	MATD	B-41 - B-42
04	S	BOTTOM	B-15 - B-18	24	S	MATU	B-42 - B-43
05	S	WPARAM	B-18 - B-20	25	S	NSI	B-43
06	S	REGWAV	B-20 - B-22	26	S	INIT	B-43 - B-44
07	S	FINDM	B-22 - B-23	27	S	INITM	B-44 - B-45
08	F	CEL	B-23 - B-25	28	S	INITMM	B-45 - B-46
09	S	SNCNDN	B-25 - B-26	29	S	SAVEM	B-46 - B-47
10	S	CPARAM	B-26 - B-29	30	S	SAVEMM	B-47
11	S	NUMSTA	B-29 - B-30	31	S	DOC1	B-47 - B-51
12	S	EXTRAP	B-30 - B-31	32	S	DOC2	B-52
13	S	RETAIN	B-31	33	S	DOC3	B-52 - B-53
14	S	CRITV	B-31 - B-32	34	S	CKFPAR	B-53 - B-54
15	S	MARCH	B-32 - B-34	35	S	CKOPTI	B-54
16	S	LANDBC	B-34 - B-35	36	S	CKVAL	B-54 - B-55
17	S	RUNUP	B-35 - B-37	37	S	CKVAL1	B-55 - B-56
18	S	OVERT	B-37 - B-38	38	S	CKVAL2	B-56
19	S	SEABC	B-38 - B-39	39	S	CKSUBR	B-56 - B-57
20	S	MATAFG	B-39 - B-40	40	S	STOPP	B-57 - B-58

Table 5: Interdependence among the program units of computer program BEFORR.

No.	Program Unit	Called from	Makes Call(s) to
00	MAIN	-	01, 02, 03, 04, 05, 06, 10, 11, 12, 13, 14, 15, 16, 19, 26, 27, 28, 29, 30, 31, 32, 33, 39, 40
01	OPENIO	00	-
02	INPUT1	00, 39	34, 35, 36, 37, 38, 40
03	INPUT2	00, 39	34, 37
04	BOTTOM	00, 39	34, 37, 40
05	WPARAM	00, 39	34
06	REGWAV	00, 39	07, 08, 09, 34
07	FINDM	06	08
08	CEL	06, 07	-
09	SNCNDN	06	40
10	CPARAM	00	25
11	NUMSTA	00, 39	25, 34
12	EXTRAP	00, 39	34
13	RETAIN	00, 39	34
14	CRITV	00, 39	25, 34
15	MARCH	00, 39	20, 21, 22, 23, 24, 25, 34
16	LANDBC	00, 39	17, 18, 34
17	RUNUP	16, 39	20, 21, 22, 34
18	OVERT	16, 39	34
19	SEABC	00, 39	34
20	MATAFG	15, 17, 39	34
21	MATGJR	15, 17, 39	34
22	MATS	15, 17, 39	34
23	MATD	15, 39	34
24	MATU	15, 39	34
25	NSI	10, 11, 14, 15, 25	-
26	INIT	00, 39	34
27	INITM	00, 39	34
28	INITMM	00, 39	34
29	SAVEM	00, 39	34
30	SAVEMM	00, 39	34

Continued on the next page.

Table 5 continued from the previous page.

No.	Program Unit	Called from	Makes Call(s) to
31	DOC1	00, 39	34
32	DOC2	00, 39	25, 34, 40
33	DOC3	00, 39	34
34	CKFPAR	All units except for 00, 01, 07, 08, 09, 10, 25, 34, 35, 36, 37, 38, 39, 40	-
35	CKOPTI	02	-
36	CKVAL	02	38
37	CKVAL1	02, 03, 04	-
38	CKVAL2	02, 36	-
39	CKSUBR	00	All units except for 00, 01, 07, 08, 09, 10, 25, 34, 35, 36, 37, 38, 39, 40
40	STOPP	00, 02, 04, 09, 32	-

16 LANDBC manages the landward boundary conditions for wave run-up, overtopping, or transmission.

26 INIT specifies the initial conditions given by $\eta = 0$ and $u = 0$ at $t = 0$ as well as the initial values for the variables saved at LEVEL=1 and LEVEL=2 as described in Section 4.2.4.

27 INITM sets the initial hydrodynamic quantities at the beginning of a computation unit using the saved values at LEVEL=1, that is, at the end of the preceding computation unit.

28 INITMM sets the initial hydrodynamic quantities at the beginning of one set of five computation units using the saved values at LEVEL=2 or LEVEL=3 as explained in Section 4.2.4.

29 SAVEM saves the hydrodynamic quantities at the end of a computation unit (LEVEL=1).

30 SAVEMM saves the hydrodynamic quantities at the end of one set of five computation units (LEVEL=2 and LEVEL=3).

31 DOC1 documents the input data and dimensionless parameters before the time-marching computation in the file BDOC.

32 DOC2 writes the computation log during the time-marching computation in the file BDOC as well as on screen.

33 DOC3 writes in the file BINPUT and completes the documentation in the file BDOC.

34 CKFPAR checks whether the values of the integers N1R, N2R, N3R, N4R, N5R, and N6R used to specify the size of matrices and vectors in the main program are equal to the values of the corresponding integers N1, N2, N3, N4, N5, and N6 used in the subroutines.

35 CKOPTI checks whether the options selected by a user are within the ranges available in the present form of BEFORR-RBREAK.

36 CKVAL checks some of the user-specified values with the assistance of Subr. 38 CKVAL2. These checks could have been a part of Subr. 02 INPUT1 but are separated because of possible limitations on some FORTRAN compilers in handling a large number of IF statements.

37 CKVAL1 checks whether some of the values selected by a user are within the ranges available in the present form of BEFORR.

38 CKVAL2 facilitates the error count and reporting for Subr. 02 INPUT1 and 36 CKVAL.

39 CKSUBR calls subroutines which use any of the PARAMETERs N1, N2, N3, N4, N5, and N6 to check the values of the PARAMETERs being used. The check is actually performed by Subr. 34 CKFPAR. Calls made from Subr. 39 CKSUBR are identified with MODE=1 or ICALL=1.

40 STOPP executes a programmed stop if some of the input requirements for BEFORR are not satisfied.

• 4.4 •

Parameters and Variables in COMMON Blocks

The computer program BEFORR and RBREAK share many names of COMMON blocks, parameters, and variables. The COMMON blocks, parameters, and variables with the same names have similar or identical definitions.

Table 6 lists the names of all COMMON blocks included in the computer program BEFORR. Also provided in that table is a brief remark on the contents of each block as compared to those of the counterpart in RBREAK counterpart. The *counterpart* herein means COMMON block with the same name from the *other* program. The only parameters and variables added or modified in the COMMON blocks marked with an asterisk in Table 6 are explained in the following since the others have been explained in Section 3.5.

/DIMENS/ contains the integers used to specify the sizes of matrices and vectors.

N6R = N6 = maximum number of computation units allowed. Presently N6=2000.

/CPAR4/ contains the computation parameters related to the finite difference method used herein.

DELTA = normalized water depth defining the computational waterline. An *initial* value of DELTA is specified by a user in the primary input data file. This initial value may be reduced during the BEFORR computation to overcome numerical difficulties.

Table 6: List of COMMON blocks in computer program BEFORR.

No.	Name of COMMON Block	Contents Compared to Those of Its RBREAK Counterpart	
1	/DIMENS/	*	Same except for N6R added in BEFORR
2	/CONSTA/		Same
3	/ID/		Same except for IENERG deleted in BEFORR
4	/CPAR1/		Same
5	/CPAR2/		Same except for NONM, NHOP deleted in BEFORR
6	/CPAR3/		Same
7	/CPAR4/	*	Same except for DELTA modified in BEFORR
8	/REQ1/		Same
9	/REQ2/		Same
10	/WAVE1/		Same
11	/WAVE2/		Same
12	/WAVE3/		Same except for ETAIS,ETARS,ETATS deleted in BEFORR
13	/WAVE4/		Same
14	/WAVE5/		Same
15	/WAVE6/		Same except for NDAT deleted in BEFORR
16	/BOT1/		Same
17	/BOT2/		Same
18	/BOT3/		Same
19	/BOT4/		Same
20	/BOT5/		Same
21	/HYDRO/		Same
22	/MATRIX/		Same
23	/RUNP1/	*	Same except for JMAX modified in BEFORR
34	/RUNP2/		Same except for RUNUPS,RSTAT,RMEAN deleted in BEFORR
24	/ARMOR/		Same as /ARMOR1/ of RBREAK
25	/SAVBC/		Same
26	/VALUEN/		Same
27	/DIAGID/	*	Added in BEFORR
29	/DIAGV1/	*	Added in BEFORR
30	/DIAGV2/	*	Added in BEFORR

/RUNP1/ contains contains parameters and quantities related to wave run-up.

JMAX = largest value of S during the time interval $0 < t \leq$TMAX as compared to JMAX in RBREAK corresponding to the time interval MSTAT$< t \leq$TMAX.

/DIAGID/ contains the identifiers used in the self-adjusting mechanism.

INONE=1 indicates that the regular incident wave profile needs to be re-computed at the beginning of a computation unit because of the increased value of NONE. INONE=0 indicates otherwise.

ITRY counts the number of trial runs carried out for a computation unit.

IREV indicates the number of computational reversals made to overcome IPROB=9 as explained in Section 4.2.4.

LEVEL specifies the past time levels to which the computation is reversed.

/DIAGV1/ contains the integers used as indicator or to save certain quantities at specified past time levels or computation units.

SSAV stores the values of the waterline node S at the specified past time levels.

JMSAV stores the values of JMAX at the specified past time levels.

NONESV(m) = the value of NONE assigned to the computation unit m, where the index m=1,2,...,MWAVE and MWAVE is the number of computation units.

MFIVE = number of sets of five computation units.

/DIAGV2/ contains the double precision variables used to save certain quantities at specified past time levels or computation units.

USAV stores the spatial variations of m and h at the specified past time levels.

VSAV stores the spatial variations of $u = m/h$ at the specified past time levels.

ESAV stores the spatial variations of η at the specified past time levels.

DELSAV stores the values of DELTA at the specified past time levels.

DELTSV(m) = the value of DELTA assigned to the computation unit m, where the index m=1,2,...,MWAVE and MWAVE is the number of computation units.

• 4.5 •

WARNING AND ERROR MESSAGES

The only one warning message that the computer program BEFORR may issue is the same as that of RBREAK described at the beginning of Section 3.7. The first eleven error messages related to the dimensions of the vectors and matrices in Section 3.7 are also included in BEFORR.

The requirements of input data which are checked by the computer program BEFORR are grouped and presented in the following. Any error message will result in the termination of computation.

GROUP 1: Computation parameters MSTAT, MSAVA1, and MSAVA2 explained in COMMONs /CPAR1/ and /CPAR4/ in Section 3.5.

REQUIREMENT	ERROR MESSAGE ISSUED IF VIOLATED
0≤MSTAT≤(TMAX-1)	Input Error: MSTAT = ··· Specify MSTAT in the range of [0,···]
1≤MSAVA2≤TMAX	Input Error: MSAVA2 = ··· Specify MSAVA2 in the range of [1,···]
1≤MSAVA1≤MSAVA2	Input Error: MSAVA1 = ··· Specify MSAVA1 in the range of [···,···]

GROUP 2: Input data values that must be non-negative.

REQUIRE-MENT	ERROR MESSAGE ISSUED IF VIOLATED	REFER TO SECTION 3.5 EXCEPT FOR MREP
XBSEG(i)≥0	XBSEG must be non-negative.	COMMON /BOT5/
CL≥0	CL must be non-negative.	COMMON /ARMOR1/
CM≥0	CM must be non-negative.	COMMON /ARMOR1/
X1≥0	X1 must be non-negative.	COMMON /CPAR4/
X2≥0	X2 must be non-negative.	COMMON /CPAR4/
FWP≥0	FWP must be non-negative.	COMMON /BOT1/
MREP≥0	MREP must be positive or zero.	Sections 4.1 & 3.3.1

GROUP 3: Input data values that must be positive.

REQUIREMENT	ERROR MESSAGE ISSUED IF VIOLATED	REFER TO SECTION 3.5 EXCEPT FOR DELTA
DELRP>0	DELRP must be positive.	COMMON /RUNP1/
TREQ>0	TREQ must be positive.	COMMON /REQ2/
WBSEG>0	WBSEG must be positive.	COMMON /BOT5/
C2>0	C2 must be positive.	COMMON /ARMOR1/
C3>0	C3 must be positive.	COMMON /ARMOR1/
SG>0	SG must be positive.	COMMON /ARMOR1/
CD>0	CD must be positive.	COMMON /ARMOR1/
TANPHI>0	TANPHI must be positive.	COMMON /ARMOR1/
DAP>0	DAP must be positive.	COMMON /ARMOR1/
DELTA>0	DELTA must be positive.	Section 4.2.1
TMAX>0	TMAX must be positive.	COMMON /CPAR4/
HREFP>0	HREFP must be positive.	COMMON /WAVE1/
TP>0	TP must be positive.	COMMON /WAVE1/
KSREF>0	KSREF must be positive.	COMMON /WAVE2/
KSSEA>0	KSSEA must be positive.	COMMON /WAVE2/
DSEAP>0	DSEAP must be positive.	COMMON /BOT1/
TSLOPS>0	TSLOPS must be positive.	COMMON /BOT2/

GROUP 4: Input data integers that must be at least unity where these parameters are explained in COMMONs /CPAR1/ and /CPAR2/ in Section 3.5.

REQUIREMENT	ERROR MESSAGE ISSUED IF VIOLATED
NTIMES$\geq$1	NTIMES must be at least unity.
MULTIF$\geq$1	MULTIF must be at least unity.
NRATE$\geq$1	NRATE must be at least unity.

GROUP 5: Miscellaneous checks.

IF THIS HAPPENS	THIS ERROR MESSAGE WILL BE ISSUED
IWAVE=1 but TMAX not a whole number	TMAX must be a whole number for IWAVE=1. Programmed stop.
IREQ=1 but none of IELEV,IV,IDUDT,ISNR is unity	Special storing requested, but pertinent identifiers not specified correctly. Check identifiers IREQ,IELEV,IV,IDUDT,ISNR. Programmed stop.
TREQ(i)>TMAX	TREQ can not exceed TMAX. Programmed stop.
IJOB<3 but submerged structure specified	SWL is always above the structure. RUNUP/OVERTOPPING computation can not be performed. Programmed stop.
IJOB=3 but subaerial structure specified	Part of the structure is above SWL. TRANSMISSION computation can not be performed. Programmed stop.
Subr. SNCNDN fails	Failure in Subr. 09 SNCNDN. Programmed stop.
Function CEL fails	Failure in Function CEL. Programmed stop.

A couple of error messages related to the self-adjusting mechanism have been discussed in Section 4.2.4.

It is possible that during the run-up computation, the slope geometry specified as input may not be long enough to facilitate the movement of the computational waterline. When this happens, the following message is issued.

```
From Subr.  16 LANDBC:
t = ···; S = ···; End Node = ···
Slope is not long enough to accomodate shoreline movement
Specify longer slope or choose overtopping computation
```

It is also possible that during the time-marching computation for regular waves, the dimension N2 may become too small to accomodate the computation of the regular wave profile. The error message associated with this incident is.

```
Specify larger dimension N2 to facilitate computation
of incident regular wave profile at seaward boundary.
N2 must be greater than NONE.
Currently, N2 = ···
Current try requires NONE = ···
Time of occurrence TIME = ···
Provide enough space for possible increase in NONE.
Computation aborted.
```

• 4.6 •

Output Produced by BEFORR

The names of the four files containing the output from the computer program BEFORR start with the letter "B" for easy identification. The four output files are BDOC, BMSG, BSPACE, and BINPUT.

File BDOC contains (1) a summary of input data, written in the format similar to that of the file ODOC produced by the computer program RBREAK and (2) computation log written by Subr. 32 DOC2.

File BMSG contains the computation log described in Section 4.2.2 as well as the warning and error messages explained in Section 4.5.

File BSPACE stores the normalized structure geometry written in Subr. 31 DOC1 as follows:

```
      WRITE (52,9000) JE
      WRITE (52,8000) (XB(J),ZB(J),J=1,JE)
8000 FORMAT (5D15.6)
9000 FORMAT (I8)
```

The variables involved are explained in Section 3.5.

Variable	Description in Section 3.5
JE	COMMON /CPAR3/
XB,ZB	COMMON /BOT3/

File BINPUT contains the following:

1. Header copied from the primary input data file and written in Subr. 02 INPUT1 as follows:

```
      WRITE (51,1110) NLINES
      DO 110 L = 1,NLINES
         WRITE (51,1120) (COMMEN(I),I=1,15)
  110 CONTINUE
 1110 FORMAT (3I8)
```

2. Case-signatures consisting of MWAVE, FINP2, NDATA, and NONEM, written in Subr. 31 DOC1 in this order:

```
      WRITE (51,9000) MWAVE
      IF (IWAVE.GT.1) THEN
         WRITE (51,5000) FINP2
         WRITE (51,9000) NDATA
      ENDIF
      WRITE (51,9000) NONEM
 5000 FORMAT (A10)
 9000 FORMAT (I8)
```

   These *case-signatures* are to be compared to their counterparts in the subsequent RBREAK computation. It is deemed necessary to do this comparison since any BEFORR computation produces the file BINPUT. The file BINPUT used in the subsequent RBREAK computation should be that produced by the preceding BEFORR computation.

3. Maximum value of NONE for regular waves, which is the value of NONE for the last computation unit, written by Subr. 33 DOC3 as follows:

```
      IF (IWAVE.EQ.1) WRITE (51,9000) NONESV(MWAVE)
 9000 FORMAT (I8)
```

   This value will be used to check the value of PARAMETER N2 used in the computer program RBREAK for the subsequent computation.

4. Computation parameters to be used in the subsequent RBREAK computation. These parameters are written by Subr. 33 DOC3 in the following order:

```
      DO 100 M = 1,MWAVE-1
         WRITE (51,5110) M,NONESV(M),NONESV(M),DELTSV(M)
  100 CONTINUE
      WRITE (51,5110) MWAVE,NONESV(MWAVE),NEND,DELTSV(MWAVE)
 5110 FORMAT (3I8,D18.9)
```

   The last MWAVE lines of the file BINPUT correspond to the MWAVE computation units executed by BEFORR. Each of the MWAVE lines contains the following: (1) Sequential number M, (2) The value of NONE assigned to the computation unit M, (3) The number of time steps in the computation unit M, and (4) The value of DELTA assigned to the computation unit M. The M-th line of the MWAVE lines will be read at the beginning of the computation unit M in the subsequent RBREAK computation.

The variables used above have been explained before as shown below.

VARIABLE	DESCRIPTION
NLINES,COMMEN	Section 3.6.2
FINP2	Name of the file containing the input wave train
NDATA	COMMON /WAVE6/ in Section 3.5
MWAVE	COMMON /CPAR1/ in Section 3.5
NONEM	COMMON /CPAR2/ in Section 3.5
NONESV,DELTSV	COMMONs /DIAGV1/ and /DIAGV2/ in Section 4.4
M	= M-th computation unit

PART V
CONCLUSIONS AND RECOMMENDATIONS

The computer program RBREAK assisted by the computer program BEFORR simulates the interaction of of normally-incident irregular waves with an impermeable coastal structure or beach in the manner similar to hydraulic model tests in a wave flume. RBREAK has been evaluated using three irregular wave test runs for irregular wave reflection and run-up on a 1:3 rough impermeable slope (Kobayashi, Cox, and Wurjanto 1990). RBREAK has also been compared with available field data on wave set-up and run-up on beaches (Kobayashi and Wurjanto 1991). Users of RBREAK are strongly recommended to read these papers which presented more physical interpretations of the computed results using additional spectral and time series analyses. Users will need the subroutines reported by Cox, Kobayashi, and Wurjanto (1991) to perform these additional analyses if the subroutines are not available to users. The interpretations of the computed results are generally much more difficult for irregular waves than for regular waves. Relatedly, spurious computed results are harder to detect for irregular waves than for regular waves. Consequently, it is recommended to check all the output from RBREAK carefully.

RBREAK has never been evaluated for its limitations and capabilities for predicting irregular wave overtopping and transmission. The hydraulic stability and movement of armor units under the action of irregular waves have been computed only by the related numerical model which includes a thin permeable underlayer as described by Kobayashi, Wurjanto, and Cox (1990a,b). The computed hydraulic stability and movement of armor units under the action of irregular waves can be difficult to interpret since the corresponding laboratory data are normally analyzed in terms of the profile change of the slope protected with armor units. Users of RBREAK will need to evaluate and calibrate RBREAK if it is to be applied to the problems for which RBREAK has not been evaluated.

The numerical models RBREAK and IBREAK are based on the approximate governing equations that are time-dependent but one-dimensional normal to a long coastal structure or shoreline with the assumption of uniformity along the structure or shoreline. These models are probably the simplest time-dependent models that can be applied for breaking and broken waves. The computation time for RBREAK for irregular waves listed in Appendix H should be bearable for users in research organizations. Our previous calibrations and evaluations of these models together with those by Van der Meer and Breteler (1990), Allsop *et al.* (1988), and Thompson (1988) indicate that these models predict some quantities accurately but other quantities only marginally. As a result, correct interpretations of the output of RBREAK are essential to avoid misuse of RBREAK. It will eventually be necessary to develop unsteady two-dimensional and three-dimensional models to improve the accuracy of the models, although computational efforts will increase considerably.

REFERENCES

Ahrens, J.P., 1975. Large wave tank tests of riprap stability. *Technical Memo No. 51*, U.S. Army Coastal Engineering Research Center, Ft. Belvoir, Va.

Ahrens, J.P., and McCartney, B.L., 1975. Wave period effect on the stability of riprap. *Proc. Civil Engineering in the Oceans / III*, ASCE, 1019-1034.

Ahrens, J.P., and Martin, F.T., 1985. Wave runup formulas for smooth slopes. *J. Waterway, Port, Coastal, and Ocean Engineering*, ASCE, **111** (1), 128-133.

Allsop, N.W.H., Smallman, J.V., and Stephens, R.V., 1988. Development and application of a mathematical model of wave action on steep slopes. *Proc. 21st Coastal Engineering Conference*, ASCE, 281-291.

Bouws, E., Günther, H., Rosenthal, W., and Vincent, C.L., 1985. Similarity of the wind wave spectrum in finite depth water. 1. Spectral form. *J. Geophysical Research*, **90** (C1), 975-986.

Bruun, P., ed., 1985. *The Design and Construction of Mounds for Breakwaters and Coastal Protection.* Elsevier, Amsterdam, The Netherlands.

Cox, D.T., Kobayashi, N., and Wurjanto, A., 1991. Computer programs for spectral and time series analyses for random waves. *Research Report No. CACR-91-06*, Center for Applied Coastal Research, Univ. of Delaware, Newark, Delaware.

Elgar, S., and Guza, R.T., 1985a. Shoaling gravity waves: comparisons between field observations, linear theory, and a nonlinear model. *J. Fluid Mechanics*, **158**, 47-70.

Elgar, S., and Guza, R.T., 1985b. Observation of bispectra of shoaling surface gravity waves. *J. Fluid Mechanics*, **161**, 425-448.

Guza, R.T., and Thornton, E.B., 1982. Swash oscillations on a natural beach. *J. Geophysical Research*, **87** (C1), 483-491.

Hibberd, S., and Peregrine, D.H., 1979. Surf and run-up on a beach: A uniform bore. *J. Fluid Mechanics*, **95** (2), 323-345.

Holman, R.A., and Sallenger, A.H., Jr., 1985. Setup and swash on a natural beach. *J. Geophysical Research*, **90** (C1), 945-953.

Jonsson, I.G., 1980. A new approach to oscillatory rough turbulent boundary layers. *Ocean Engineering*, **7** (1), 109-152.

Kobayashi, N., and Jacobs, B.K., 1985. Stability of armor units on composite slope. *J. Waterway, Port, Coastal, and Ocean Engineering*, ASCE, **111** (5), 880-894.

Kobayashi, N., Roy, I., and Otta, A.K., 1986. Numerical simulation of wave run-up and armor stability. Paper 5088, *18th Offshore Technology Conference*, 51-56.

Kobayashi, N., and Greenwald, J.H., 1986. Prediction of wave run-up and riprap stability. *Proc. 20th Coastal Engineering Conference*, ASCE, 1958-1971.

Kobayashi, N., and DeSilva, G.S., 1987. Motion of sediment particles in swash zone. *Proc. Coastal Hydrodynamics*, ASCE, 715-730.

Kobayashi, N., and Otta, A.K., 1987. Hydraulic stability analysis of armor units. *J. Waterway, Port, Coastal, and Ocean Engineering*, ASCE, **113** (2), 171-186.

Kobayashi, N., Otta, A.K., and Roy, I., 1987. Wave reflection and run-up on rough slopes. *J. Waterway, Port, Coastal, and Ocean Engineering*, ASCE, **113** (3), 282-298.

Kobayashi, N., and Watson, K.D., 1987. Wave reflection and runup on smooth slopes. *Proc. Coastal Hydrodynamics*, ASCE, 548-563.

Kobayashi, N., 1988. Review of wave transformation and cross-shore sediment transport processes in surf zones. *J. Coastal Research*, **4** (3), 435-445.

Kobayashi, N., and Greenwald, J.H., 1988. Waterline oscillation and riprap movement. *J. Waterway, Port, Coastal, and Ocean Engineering*, ASCE, **114** (3), 281-296.

Kobayashi, N., Strzelecki, M.S., and Wurjanto, A., 1988. Swash oscillation and resulting sediment movement. *Proc. 21st Coastal Engineering Conference*, ASCE, 1167-1181.

Kobayashi, N., and Wurjanto, A., 1988. Numerical prediction of wave overtopping on coastal structures. *Research Report No. CE-88-72*, Ocean Engineering Program, Univ. of Delaware, Newark, Delaware.

Kobayashi, N., DeSilva, G.S., and Watson, K.D., 1989. Wave transformation and swash oscillation on gentle and steep slopes. *J. Geophysical Research*, **94** (C1), 951-966.

Kobayashi, N., and Wurjanto, A., 1989a. Wave overtopping on coastal structures. *J. Waterway, Port, Coastal, and Ocean Engineering*, ASCE, **115** (2), 235-251.

Kobayashi, N., and Wurjanto, A., 1989b. Numerical prediction of wave transmission over submerged breakwaters. *Research Report No. CE-89-73*, Center for Applied Coastal Research, Univ. of Delaware, Newark, Delaware.

Kobayashi, N., and Wurjanto, A., 1989c. Wave transmission over submerged breakwaters. *J. Waterway, Port, Coastal, and Ocean Engineering*, ASCE, **115** (5), 662-680.

Kobayashi, N., and Wurjanto, A., 1989d. Numerical model for design of impermeable coastal structures. *Research Report No. CE-89-75*, Center for Applied Coastal Research, Univ. of Delaware, Newark, Delaware.

Kobayashi, N., and Wurjanto, A., 1989e. Armor stability on rough permeable slopes of marine structures. *Proc. 23rd Congress of International Association for Hydraulic Research*, Ottawa, Canada, **C**, 407-414.

Kobayashi, N., and Wurjanto, A., 1989f. Numerical prediction of hydrodynamic forces and sliding motion of dolos units. *Proc. Stresses in Concrete Armor Units*, ASCE, 355-378.

Kobayashi, N., and Wurjanto, A., 1990a. Numerical prediction of armor stability and movement under irregular wave action. *Research Report No. CACR-90-04*, Center for Applied Coastal Research, Univ. of Delaware, Newark, Delaware.

Kobayashi, N., and Wurjanto, A., 1990b. Numerical model for waves on rough permeable slopes. *J. Coastal Research*, Special Issue No. **7** on Rational Design of Mound Structures, 149-166.

Kobayashi, N., Wurjanto, A., and Cox, D.T., 1990a. Irregular waves on rough permeable slopes. *J. Coastal Research*, Special Issue No. **7** on Rational Design of Mound Structures, 167-184.

Kobayashi, N., Wurjanto, A., and Cox, D.T., 1990b. Rock slopes under irregular wave attack. *Proc. 22nd Coastal Engineering Conference*, ASCE, 1306-1319.

Kobayashi, N., Cox, D.T., and Wurjanto, A., 1990. Irregular wave reflection and run-up on rough impermeable slopes. *J. Waterway, Port, Coastal, and Ocean Engineering*, ASCE, **116** (6), 708-726.

Kobayashi, N., Cox, D.T., and Wurjanto, A., 1991. Permeability effects on irregular wave run-up and reflection. *J. Coastal Research*, **7** (1), 127-136.

Kobayashi, N., and Wurjanto, A., 1991. Irregular wave setup and run-up on beaches. Submitted to *J. Waterway, Port, Coastal, and Ocean Engineering*, ASCE.

Madsen, O.S., and White, S.M., 1976. Energy dissipation on a rough slope. *J. Waterways, Harbors and Coastal Engineering Division*, ASCE, **102** (WW1), 31-48.

Mansard, E.P.D., 1988. Towards a better simulation of sea states for modelling of coastal structures. *Proc. Berm Breakwaters*, ASCE, 1-20.

Moretti, G., 1987. Computation of flows with shocks. *Ann. Rev. Fluid Mechanics*, **19**, 313-337.

Packwood, A.R., 1980. *Surf and run-up on beaches.* Ph.D. Thesis. School of Mathematics, Univ. of Bristol, United Kingdom.

Peregrine, D.H., 1983. Breaking waves on beaches. *Ann. Rev. Fluid Mechanics*, **15**, 149-178.

Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., 1986. *Numerical Recipes: The Art of Scientific Computing.* Cambridge University Press, Cambridge, United Kingdom.

Richtmyer, R.D., and Morton, K.W., 1967. *Difference Methods for Initial-Value Problems*, 2nd ed. Interscience, New York, NY.

Saville, T., Jr., 1955. Laboratory data on wave runup and overtopping on shore structures. *Technical Memo No. 64*, U.S. Army Beach Erosion Board, Document Service Center, Dayton, Ohio.

Seelig, W.N., 1980. Two-dimensional tests of wave transmission and reflection characteristics of laboratory breakwaters. *Technical Report 80-1*, U.S. Army Coastal Engineering Research Center, Ft. Belvoir, Va.

Seelig, W.N., 1983. Wave reflection from coastal structures. *Proc. Coastal Structures '83*, ASCE, 961-973.

Shore Protection Manual, 1984. U.S. Army Engineer Waterways Experiment Station, Coastal Engineering Research Center, U.S. Government Printing Office, Washington, D.C.

Sleath, J.F.A., 1984. *Sea Bed Mechanics.* Wiley-Interscience, New York, NY.

Svendsen, I.A., and Brink-Kjær, O., 1972. Shoaling of cnoidal waves. *Proc. 13th Coastal Engineering Conference*, ASCE, 365-383.

Thompson, A.C., 1988. Numerical model of breakwater wave flows. *Proc. 21st Coastal Engineering Conference*, ASCE, 2013-2027.

Van der Meer, J.W., 1988. Deterministic and probabilistic design of breakwater armor layers. *J. Waterway, Port, Coastal, and Ocean Engineering*, ASCE, **114** (1), 66-80.

Van der Meer, J.W., and Breteler, M.K., 1990. Measurement and computation of wave induced velocities on a smooth slope. *Proc. 22nd Coastal Engineering Conference*, ASCE, 191-204.

Wurjanto, A., and Kobayashi, N., 1989. Wave overtopping and transmission over breakwaters. *Proc. PORTS '89*, ASCE, 247-256.

Appendix A

Listing of Computer Program RBREAK

```
C                                                                       RBR00010
C      #######   #######   #######   ########   ######   ##    ##     RBR00020
C      ##    ##  ##    ##  ##    ##  ##         ##    ##  ##   ##      RBR00030
C      ##    ##  ##    ##  ##    ##  ##         ##    ##  ##  ##       RBR00040
C      #######   #######   #######   #######    ########  ## ##       RBR00050
C      ##  ##    ##    ##  ##  ##    ##         ##    ##  ######      RBR00060
C      ##   ##   ##    ##  ##   ##   ##         ##    ##  ###  ##      RBR00070
C      ##    ##  #######   ##    ##  ########   ##    ##  ##    ##     RBR00080
C                                                                       RBR00090
C                        Numerical Simulation of                        RBR00100
C          Random Waves on Impermeable Breakwaters and Beaches;         RBR00110
C                  Extension of Computer Program IBREAK                 RBR00120
C                                                                       RBR00130
C  Written by Andojo Wurjanto under Supervision of Nobuhisa Kobayashi   RBR00140
C                  Center for Applied Coastal Research                  RBR00150
C                    Department of Civil Engineering                    RBR00160
C              University of Delaware, Newark, Delaware 19716           RBR00170
C                              May  1991                                RBR00180
C                                                                       RBR00190
C ############################  NOTES  ##############################   RBR00200
C                                                                       RBR00210
C     To minimize numerical difficulties, especially at computational   RBR00220
C     waterline, RBREAK should be preceded by BEFORR, which determines  RBR00230
C     two crucial computation parameters for each computation unit:     RBR00240
C         1. NONE  = number of time steps in one wave period            RBR00250
C         2. DELTA = normalized water depth at computational waterline  RBR00260
C     Values of NONE and DELTA obtained by BEFORR are stored in file    RBR00270
C         'BINPUT' and are to be read by RBREAK.                        RBR00280
C     About "computation unit":                                         RBR00290
C       . RBREAK and BEFORR break computation duration into computa-    RBR00300
C         tion units. Each computation unit is one wave period long     RBR00310
C         except for the last unit, the length of which may be less     RBR00320
C         than one wave period.                                         RBR00330
C       . The "wave period" is the reference period used for the        RBR00340
C         normalization of the governing equations.                     RBR00350
C     RBREAK and BEFORR share many subroutine and COMMON block names,   RBR00360
C         but they are independent programs.                            RBR00370
C                                                                       RBR00380
C ########################  MAIN PROGRAM RBREAK  ####################   RBR00390
C                                                                       RBR00400
C     Main program performs time-marching computation using             RBR00410
C     subroutines                                                       RBR00420
C                                                                       RBR00430
      PROGRAM RBREAK                                                    RBR00440
C                                                                       RBR00450
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR00460
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR00470
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI,KCNO,MCNO,KC2                 RBR00480
      CHARACTER*10 FINP1,FINP2                                          RBR00490
      INTEGER S,SM1,SP1                                                 RBR00500
      DIMENSION VDUM(N1)                                                RBR00510
C                                                                       RBR00520
C     COMMONs                                                           RBR00530
C     ------------------------------------------------------------      RBR00540
C      Name     Contents                                                RBR00550
```

```
C     ------------------------------------------------------------------ RBR00560
C     /DIMENS/  The values of FORTRAN PARAMETERs specified in the main  RBR00570
C               program                                                  RBR00580
C     /CONSTA/  Basic constants                                          RBR00590
C     /ID/      Identifiers specifying user's options                    RBR00600
C     /CPAR1/ to /CPAR4/ Computation parameters                          RBR00610
C     /REQ1/ and /REQ2/ Identifiers and values associated with special   RBR00620
C               storing (see Subr. 02 INPUT1 for more explanation)       RBR00630
C     /WAVE1/   Physical wave parameters                                 RBR00640
C     /WAVE2/   Dimensionless wave parameters                            RBR00650
C     /WAVE3/   Norm. free surface elevations as a function of time      RBR00660
C     /WAVE4/   Maximum and minimum of ETA in /WAVE3/                    RBR00670
C     /WAVE5/   Cnoidal wave parameters (K, E, m and 1-m)                RBR00680
C     /WAVE6/   Number of data points specified in input file FINP2      RBR00690
C     /BOT1/    Physical quantities related to structure                 RBR00700
C     /BOT2/    Normalized quantities related to structure               RBR00710
C     /BOT3/    Normalized structure geometry                            RBR00720
C     /BOT4/ and /BOT5/ Physical structure geometry                      RBR00730
C     /HYDRO/   Hydrodynamic quantities computed                         RBR00740
C     /MATRIX/  Elements of matrices used in numerical method            RBR00750
C     /RUNP1/ and /RUNP2/ Quantities related to wave runup               RBR00760
C     /OVER/    Quantities related to wave overtopping                   RBR00770
C     /COEFS/   Reflection and transmission coefficients                 RBR00780
C     /STT1/ and /STT2/ Statistical quantities                           RBR00790
C     /ENERG/   Quantities related to wave energy                        RBR00800
C     /ARMOR1/  Parameters, read as input, for computation of armor      RBR00810
C               stability or movement                                    RBR00820
C     /ARMOR2/ and /ARMOR3/ Computed parameters for computation of       RBR00830
C               armor stability or movement                              RBR00840
C     /ARMOR4/ and /ARMOR5/ Parameters and quantities associated with    RBR00850
C               computation of armor stability                           RBR00860
C     /ARMOR6/ and /ARMOR7/ Parameters and quantities associated with    RBR00870
C               computation of armor movement                            RBR00880
C     /SAVBC/   Node numbers related to options ISAVB=1 and ISAVC=1      RBR00890
C     /FILBC/   File names related to options ISAVB=1 and ISAVC=1        RBR00900
C     /VALUEN/  Values at present time level stored during computation   RBR00910
C     ------------------------------------------------------------------ RBR00920
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                RBR00930
      COMMON /CONSTA/ PI,GRAV                                            RBR00940
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,         RBR00950
     +                ISAVA,ISAVB,ISAVC                                  RBR00960
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES             RBR00970
      COMMON /CPAR2/  MULTIF,NONEM,NONM,NONE,NEND,NRATE,NHOP             RBR00980
      COMMON /CPAR3/  INITS,JE,JE1                                       RBR00990
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX      RBR01000
      COMMON /REQ1/   IREQ,IELEV,IV,IDUDT,ISNR,NREQ                      RBR01010
      COMMON /REQ2/   TREQ(N5)                                           RBR01020
      COMMON /WAVE1/  HREFP,TP,WL0P                                      RBR01030
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA           RBR01040
      COMMON /WAVE3/  ETA(N2),ETAIS(N2),ETARS(N2),ETATS(N2)              RBR01050
      COMMON /WAVE4/  ETAMAX,ETAMIN                                      RBR01060
      COMMON /WAVE5/  KCNO,ECNO,MCNO,KC2                                 RBR01070
      COMMON /WAVE6/  NDATA,NDAT                                         RBR01080
      COMMON /BOT1/   DSEAP,DLANDP,FWP                                   RBR01090
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     RBR01100
```

```
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)      RBR01110
      COMMON /BOT4/   NBSEG                                              RBR01120
      COMMON /BOT5/   WBSEG(N4),TBSLOP(N4),XBSEG(N4),ZBSEG(N4)           RBR01130
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)              RBR01140
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)   RBR01150
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                               RBR01160
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3),RUNUPS(N3),RSTAT(3,N3),       RBR01170
     +                RMEAN(N3)                                          RBR01180
      COMMON /OVER/   OV(4),OVMEAN                                       RBR01190
      COMMON /COEFS/  RCOEF(3),TCOEF(3)                                  RBR01200
      COMMON /STT1/   ELSTAT(3,3),U1STAT(N1),ESTAT(3,N1),VSTAT(3,N1),    RBR01210
     +                ELMEAN(3),U1MEAN(N1),EMEAN(N1),VMEAN(N1)           RBR01220
      COMMON /STT2/   TSTAT1,TSTAT2                                      RBR01230
      COMMON /ENERG/  ENER(7,N1),ENERB(15),ENTEMP(3,N1)                  RBR01240
      COMMON /ARMOR1/ C2,C3,CD,CL,CM,SG,TANPHI,AMIN,AMAX,DAP             RBR01250
      COMMON /ARMOR2/ SG1,CTAN(N1)                                       RBR01260
      COMMON /ARMOR3/ JSTAB,JSTABM                                       RBR01270
      COMMON /ARMOR4/ JSNSC,JATMIN                                       RBR01280
      COMMON /ARMOR5/ CSTAB1,CSTAB2,AMAXS,AMINS,E2,E3PRE(N1),SNR(N1),    RBR01290
     +                SNSX(N1),TSNSX(N1),ATMIN(4,N1),SNSC,TSNSC          RBR01300
      COMMON /ARMOR6/ NMOVE,NSTOP,ISTATE(N1),NODFI(N1)                   RBR01310
      COMMON /ARMOR7/ CSTAB3,CSTAB4,CM1,DA,SIGDA,WEIG,                   RBR01320
     +                VA(N1),XAA(N1),XA(N1),TDIS(N1)                     RBR01330
      COMMON /SAVBC/  NNODB,NNODC,NODB(N5),NODC(N5)                      RBR01340
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                      RBR01350
C                                                                        RBR01360
C             READ INPUT                                                 RBR01370
C             ------------------------------------------------------     RBR01380
C             Subr. 01 OPENIO opens input and output files               RBR01390
C             Subr. 02 INPUT1 reads most of input data                   RBR01400
C             Subr. 03 INPUT2 reads wave profile at seaward boundary     RBR01410
C             Subr. 49 STOPP  stops execution of the computation         RBR01420
C             ------------------------------------------------------     RBR01430
      WRITE (*,*) 'RBREAK reports progress on screen every MREP waves'   RBR01440
      WRITE (*,*) 'Enter MREP (0 if you do not want the report)'         RBR01450
      READ  (*,*)  MREP                                                  RBR01460
      WRITE (*,*) 'Name of primary input-data-file?'                     RBR01470
      READ  (*,5000) FINP1                                               RBR01480
      CALL OPENIO (1,0,FINP1,FINP2)                                      RBR01490
      CALL INPUT1 (1,0,FINP2)                                            RBR01500
      CALL OPENIO (2,0,FINP1,FINP2)                                      RBR01510
      IF (IWAVE.GT.1) CALL INPUT2 (1)                                    RBR01520
      IF (MREP.LT.0)  CALL STOPP (1,1)                                   RBR01530
 5000 FORMAT (A10)                                                       RBR01540
C                                                                        RBR01550
C             CHECK FORTRAN PARAMETERS                                   RBR01560
C             ------------------------------------------------------     RBR01570
C             Each subroutine having FORTRAN "PARAMETER" statement is    RBR01580
C             called by Subroutine 48 CKSUBR to check its "PARAMETER"    RBR01590
C             values, which must be the same as their counterparts in    RBR01600
C             the main program. These checking calls to subroutines      RBR01610
C             are indicated by MODE=0, as opposed to "working" calls     RBR01620
C             indicated by MODE=1 as used above in Call INPUT1 (1,...)   RBR01630
C             and Call INPUT2 (1).                                       RBR01640
C             ------------------------------------------------------     RBR01650
```

```
      CALL CKSUBR (N1,N2,N3,N4,N5)                                      RBR01660
      MODE=1                                                            RBR01670
C                                                                       RBR01680
C             GROUNDWORK                                                RBR01690
C             ----------                                                RBR01700
C           . Subr. 04 BOTTOM computes normalized structure geometry    RBR01710
C           . Subr. 05 WPARAM calculates wave parameters                RBR01720
C           . This call to Subr. 10 CPARAM checks case-signature NONM   RBR01730
C           . Subr. 40 SPARAM calculates parameters needed for compu-   RBR01740
C             tation of armor stability or movement                     RBR01750
C           . Subr. 26 INIT specifies initial conditions                RBR01760
C           . This call to Subr. 06 REGWAV is for initial incident      RBR01770
C             wave profile and documentation purpose in Subr. 43 DOC1   RBR01780
C             where NONE=NONEM is used                                  RBR01790
C           . Subr. 43 DOC1 writes essential information before         RBR01800
C             time-marching computation                                 RBR01810
C             ----------                                                RBR01820
      CALL BOTTOM (MODE)                                                RBR01830
      CALL WPARAM (MODE)                                                RBR01840
      CALL CPARAM (0)                                                   RBR01850
      IF (ISTAB.GT.0) CALL SPARAM (MODE)                                RBR01860
      CALL INIT (MODE)                                                  RBR01870
      IF (IWAVE.EQ.1) CALL REGWAV (MODE)                                RBR01880
      CALL DOC1 (MODE)                                                  RBR01890
      IF (IJOB.EQ.3) SM1=INITS-1                                        RBR01900
C                                                                       RBR01910
C             TIME MARCHING COMPUTATION                                 RBR01920
C             -------------------------                                 RBR01930
C                                                                       RBR01940
      DO 550 M = 1,MWAVE                                                RBR01950
C                                                                       RBR01960
C       ------: 550 Begins :-------------------------------------------- RBR01970
C             : There are MWAVE computation units to execute            RBR01980
C                                                                       RBR01990
C       Subr. 28 INSTAT initializes quantities used in Subr. 29 SVSTAT  RBR02000
C       Subr. 02 INPUT1 reads computation parameters from file 'BINPUT' RBR02010
C       Subr. 06 REGWAV computes incident regular wave profile          RBR02020
C       Subr. 10 CPARAM calculates computation parameters which are     RBR02030
C                       dependent on NONE                               RBR02040
C                                                                       RBR02050
        IF (M.GT.MSTAT) CALL INSTAT (MODE)                              RBR02060
        NONEOL=NONE                                                     RBR02070
        CALL INPUT1 (9,M,FINP2)                                         RBR02080
        IF (NONE.NE.NONEOL) THEN                                        RBR02090
          CALL CPARAM (MODE)                                            RBR02100
          IF (IWAVE.EQ.1) CALL REGWAV (MODE)                            RBR02110
        ENDIF                                                           RBR02120
        IF (ISAVB.EQ.1.OR.ISAVC.EQ.1) CALL OPENIO (3,M,FINP1,FINP2)     RBR02130
C                                                                       RBR02140
        DO 540 N = 1,NEND                                               RBR02150
C                                                                       RBR02160
C         ----: 540 Begins :------------------------------------------- RBR02170
C             : There are NEND time levels to execute                   RBR02180
C                                                                       RBR02190
C       . TIME = normalized time                                        RBR02200
```

```
C       . Subr. 12 EXTRAP estimates hydrodynamic quantities at the      RBR02210
C         node immediately landward of the present waterline node       RBR02220
C       . Subr. 13 RETAIN retains some values at present time level     RBR02230
C         at landward and seaward boundaries                            RBR02240
C       . Subr. 14 CRITV calculates critical velocities used in         RBR02250
C         characteristic variables and checks for negative water        RBR02260
C         depth                                                         RBR02270
C       . Subr. 15 MARCH performs time-marching from present time       RBR02280
C         level (N-1) to next (N) excluding landward and seaward        RBR02290
C         boundaries, which are handled by Subr. 16 LANDBC and          RBR02300
C         19 SEABC, respectively                                        RBR02310
C       . Subr. 11 NUMSTA checks if numerical stability criterion       RBR02320
C         is violated                                                   RBR02330
C                                                                       RBR02340
          TIME = DBLE(M-1) + DBLE(N)/DBLE(NONE)                         RBR02350
          IF (IJOB.LT.3) CALL EXTRAP (MODE)                             RBR02360
          CALL RETAIN (MODE)                                            RBR02370
          CALL CRITV  (MODE,M)                                          RBR02380
          CALL MARCH  (MODE,M)                                          RBR02390
          CALL LANDBC (MODE,M,N,ETAT)                                   RBR02400
          CALL SEABC  (MODE,M,N,ETAI,ETAR)                              RBR02410
          CALL NUMSTA (MODE,M)                                          RBR02420
C                                                                       RBR02430
C         ----: Computation of Energy Quantities :--------------------  RBR02440
C             : Performed by Subr. 30 ENERGY excluding the first MSTAT  RBR02450
C               computation units                                       RBR02460
C                                                                       RBR02470
          IF (IENERG.EQ.1) THEN                                         RBR02480
            IF (IWAVE.GT.1.AND.N.EQ.NEND) THEN                          RBR02490
              IF (M.EQ.MSTAT.OR.M.EQ.MWAVE) CALL ENERGY (9,M)           RBR02500
            ENDIF                                                       RBR02510
            IF (M.GT.MSTAT) CALL ENERGY (1,M)                           RBR02520
          ENDIF                                                         RBR02530
C                                                                       RBR02540
C         ----: Statistical Calculations :----------------------------  RBR02550
C             : Statistical calculations are performed excluding the    RBR02560
C               first MSTAT computation units                           RBR02570
C                                                                       RBR02580
C       . Subr. 41 VECMAT creates a vector from a row of a matrix       RBR02590
C       . Subr. 35 STAT1  finds mean, maximum, and minimum              RBR02600
C       . At node j:                                                    RBR02610
C            U1STAT(j) = mean volume flux                               RBR02620
C            ELEV(j)   = surface elevation above SWL                    RBR02630
C            V(j)      = depth-averaged velocity                        RBR02640
C       . Mean, maximum, and minimum at node j:                         RBR02650
C            ESTAT(1,j),ESTAT(2,j),ESTAT(3,j): for ELEV(j)              RBR02660
C            VSTAT(1,j),VSTAT(2,j),VSTAT(3,j): for V(j)                 RBR02670
C       . JMAX = the largest node number reached by the computational   RBR02680
C            waterline during the period when statistical calculations  RBR02690
C            are performed                                              RBR02700
C            (For IJOB=3, JMAX=INITS)                                   RBR02710
C                                                                       RBR02720
          IF (M.GT.MSTAT) THEN                                          RBR02730
            CALL VECMAT (VDUM,U,2,S,1)                                  RBR02740
            CALL STAT1  (1,U1STAT,VDUM,1,S)                             RBR02750
```

```
            CALL STAT1  (2,ESTAT ,ELEV,3,JE)                          RBR02760
            CALL STAT1  (2,VSTAT ,V   ,3,S)                           RBR02770
            IF (IJOB.LT.3.AND.S.GT.JMAX) JMAX=S                       RBR02780
          ENDIF                                                       RBR02790
C                                                                     RBR02800
C         ----: Armor Stability or Movement :-------------------------RBR02810
C             : Computation is performed excluding the first MSTAB    RBR02820
C               computation units                                     RBR02830
C                                                                     RBR02840
C       :: ISTAB=1: INITIATION OF MOVEMENT OF ARMOR UNITS             RBR02850
C          . SNR at every node is computed by Subr. 31 STABNO         RBR02860
C          . SNR(j) = stability number against rolling/sliding at     RBR02870
C            node j                                                   RBR02880
C       :: ISTAB=2: SLIDING MOTION OF ARMOR UNITS                     RBR02890
C          . Individual armor units are tracked by Subr. 32 MOVE      RBR02900
C          . NMOVE = number of units dislodged from their initial     RBR02910
C            locations                                                RBR02920
C          . NSTOP = number of units stopped after moving             RBR02930
C          . XAA(j),XA(j) = displacement of moving or stopped armor   RBR02940
C            unit number j from its initial location, normalized by   RBR02950
C            TP*sqrt(GRAV*HREFP) and DAP, respectively                RBR02960
C                                                                     RBR02970
          IF (ISTAB.GT.0.AND.M.GT.MSTAB) THEN                         RBR02980
            IF (JSTAB.GT.JSTABM) JSTABM=JSTAB                         RBR02990
            IF (ISTAB.EQ.1) THEN                                      RBR03000
              CALL STABNO (MODE)                                      RBR03010
            ELSE                                                      RBR03020
              CALL MOVE (MODE)                                        RBR03030
            ENDIF                                                     RBR03040
          ENDIF                                                       RBR03050
C                                                                     RBR03060
C         ----: Documentation :---------------------------------------RBR03070
C                                                                     RBR03080
C       . Subr. 44 DOC2 documents computed results at designated      RBR03090
C         moments                                                     RBR03100
C       . Calling DOC2(1,...) is for storing "A", i.e., spatial       RBR03110
C         variations of hydrodynamic quantities as specified in Subr. RBR03120
C         02 INPUT1                                                   RBR03130
C       . Calling DOC2(2,...) is for storing time series at specified RBR03140
C         nodes at rate NRATE per wave period where NHOP=NONE/NRATE   RBR03150
C         in Subr. 10 CPARAM                                          RBR03160
C       . Calling DOC2(3,...) is for storing spatial variations at    RBR03170
C         specified normalized time t=TREQ(i) with i=1,2,...,NREQ     RBR03180
C                                                                     RBR03190
          IF (M.GE.MSAVA1.AND.M.LE.MSAVA2) THEN                       RBR03200
            DO 110 K = 1,NTIMES                                       RBR03210
              NSAVA = K*NONE/NTIMES                                   RBR03220
              IF (N.EQ.NSAVA) CALL DOC2 (1,DUM,DUM,DUM)               RBR03230
  110       CONTINUE                                                  RBR03240
          ENDIF                                                       RBR03250
          IDUM = MOD(N,NHOP)                                          RBR03260
          IF (IDUM.EQ.0) CALL DOC2 (2,ETAI,ETAR,ETAT)                 RBR03270
          IF (IREQ.EQ.1) THEN                                         RBR03280
            DO 120 I = 1,NREQ                                         RBR03290
              ITREQ = INT(TREQ(I))                                    RBR03300
```

```
              RES    = DMOD(TREQ(I),1.D+00)                                  RBR03310
              IF (DABS(RES).LT.1.D-08) THEN                                  RBR03320
                MRQ = ITREQ                                                  RBR03330
                NRQ = NONE                                                   RBR03340
              ELSE                                                           RBR03350
                MRQ = ITREQ+1                                                RBR03360
                NRQ = RES*NONE                                               RBR03370
              ENDIF                                                          RBR03380
              IF (M.EQ.MRQ.AND.N.EQ.NRQ) CALL DOC2 (3,DUM,DUM,DUM)           RBR03390
  120       CONTINUE                                                         RBR03400
          ENDIF                                                            RBR03410
C                                                                          RBR03420
C         ----: 540 Ends :-------------------------------------------      RBR03430
C                                                                          RBR03440
  540   CONTINUE                                                           RBR03450
C                                                                          RBR03460
C       Subr. 29 SVSTAT saves quantities related to time-averaged          RBR03470
C       values                                                             RBR03480
C                                                                          RBR03490
        IF (M.GT.MSTAT) CALL SVSTAT (MODE)                                 RBR03500
        IF (MREP.GT.0) THEN                                                RBR03510
          IDUM = MOD(M,MREP)                                               RBR03520
          IF (IDUM.EQ.0) THEN                                              RBR03530
            IF (M.EQ.1) THEN                                               RBR03540
              WRITE (*,*) ' Finished ',M,' Wave Period'                    RBR03550
            ELSE                                                           RBR03560
              WRITE (*,*) ' Finished ',M,' Wave Periods'                   RBR03570
            ENDIF                                                          RBR03580
          ENDIF                                                            RBR03590
        ENDIF                                                              RBR03600
C                                                                          RBR03610
C       ------: 550 Ends :-----------------------------------------      RBR03620
C                                                                          RBR03630
  550 CONTINUE                                                             RBR03640
C                                                                          RBR03650
C             FINISHING                                                    RBR03660
C             ---------------------------------------------------------    RBR03670
C             Subr. 37 STATC  completes statistical calculations           RBR03680
C             Subr. 39 ENERGC completes computation of energy qtties.      RBR03690
C             Subr. 45 DOC3   writes results                               RBR03700
C             ---------------------------------------------------------    RBR03710
      CALL STATC (MODE)                                                    RBR03720
      IF (IENERG.EQ.1) CALL ENERGC (MODE)                                  RBR03730
      CALL DOC3 (MODE)                                                     RBR03740
C                                                                          RBR03750
      STOP                                                                 RBR03760
      END                                                                  RBR03770
C                                                                          RBR03780
C --------------------  END OF MAIN PROGRAM RBREAK  ------------------     RBR03790
C #01#######################  SUBROUTINE OPENIO  ###############RBREAK#    RBR03800
C                                                                          RBR03810
C     This subroutine opens input and output files                         RBR03820
C                                                                          RBR03830
      SUBROUTINE OPENIO (ICALL,M,FINP1,FINP2)                              RBR03840
C                                                                          RBR03850
```

```
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,        RBR03860
     +                ISAVA,ISAVB,ISAVC                                 RBR03870
      COMMON /REQ1/   IREQ,IELEV,IV,IDUDT,ISNR,NREQ                     RBR03880
      CHARACTER*10 FINP1,FINP2,FSAV1B(20),FSAV2B(20),FSAVEC(20)         RBR03890
      DATA FSAV1B /                                                     RBR03900
     1 'OSAV1B01  ','OSAV1B02  ','OSAV1B03  ','OSAV1B04  ',             RBR03910
     2 'OSAV1B05  ','OSAV1B06  ','OSAV1B07  ','OSAV1B08  ',             RBR03920
     3 'OSAV1B09  ','OSAV1B10  ','OSAV1B11  ','OSAV1B12  ',             RBR03930
     4 'OSAV1B13  ','OSAV1B14  ','OSAV1B15  ','OSAV1B16  ',             RBR03940
     5 'OSAV1B17  ','OSAV1B18  ','OSAV1B19  ','OSAV1B20  '/             RBR03950
      DATA FSAV2B /                                                     RBR03960
     1 'OSAV2B01  ','OSAV2B02  ','OSAV2B03  ','OSAV2B04  ',             RBR03970
     2 'OSAV2B05  ','OSAV2B06  ','OSAV2B07  ','OSAV2B08  ',             RBR03980
     3 'OSAV2B09  ','OSAV2B10  ','OSAV2B11  ','OSAV2B12  ',             RBR03990
     4 'OSAV2B13  ','OSAV2B14  ','OSAV2B15  ','OSAV2B16  ',             RBR04000
     5 'OSAV2B17  ','OSAV2B18  ','OSAV2B19  ','OSAV2B20  '/             RBR04010
      DATA FSAVEC /                                                     RBR04020
     1 'OSAVEC01  ','OSAVEC02  ','OSAVEC03  ','OSAVEC04  ',             RBR04030
     2 'OSAVEC05  ','OSAVEC06  ','OSAVEC07  ','OSAVEC08  ',             RBR04040
     3 'OSAVEC09  ','OSAVEC10  ','OSAVEC11  ','OSAVEC12  ',             RBR04050
     4 'OSAVEC13  ','OSAVEC14  ','OSAVEC15  ','OSAVEC16  ',             RBR04060
     5 'OSAVEC17  ','OSAVEC18  ','OSAVEC19  ','OSAVEC20  '/             RBR04070
C                                                                       RBR04080
C  DESCRIPTION OF INPUT AND OUTPUT FILES                                RBR04090
C  ------------------------------------------------------------------   RBR04100
C  (Category) Unit (Filename) Purpose                                   RBR04110
C  ------------------------------------------------------------------   RBR04120
C  (Input)   11 (FINP1)      contains primary input data                RBR04130
C  (Input)   12 (FINP2)      contains time series of normalized free    RBR04140
C                            surface elevation at seaward boundary if   RBR04150
C                            IWAVE>1                                    RBR04160
C  (Input)   14 ('BINPUT')   contains computation parameters assigned   RBR04170
C                            by BEFORR for RBREAK computation           RBR04180
C                            --> m,NONE(m),NEND(m),DELTA(m)             RBR04190
C                                m = 1,2,...,MWAVE = computation unit   RBR04200
C  (Output)  51 ('OSTAT')    stores statistics of hydrodyn. quantities  RBR04210
C                            --> JMAX                                   RBR04220
C                            --> (U1STAT(j), j=1,JMAX)                  RBR04230
C                            --> (ESTAT(i,j),j=1,JMAX)                  RBR04240
C                            --> (VSTAT(i,j),j=1,JMAX)                  RBR04250
C                                i=1,2,3                                RBR04260
C  (Output)  52 ('OSPACE')   stores structure geometry                  RBR04270
C                            --> JE,(XB(j),ZB(j),j=1,JE)                RBR04280
C                            and "A" (see Subr. 02 INPUT1 for descrip-  RBR04290
C                            tion of "A")                               RBR04300
C  (Output)  53 ('OENERG')   stores energy quantities if IENERG=1       RBR04310
C                            --> JMAX,(ENER(i,j),j=1,JMAX)              RBR04320
C                                i=1,2,...,7                            RBR04330
C  (Output)  54 ('OSTAB')    stores quantities associated with armor    RBR04340
C                            stability or movement                      RBR04350
C  (Output)  55 ('OREQ')     stores specially-requested quantities      RBR04360
C  (Output)  61 ('OSEAWAV')  stores time series of hydrodyn. quantities RBR04370
C                            at seaward boundary                        RBR04380
C  (Output)  62 ('ORUNUP')   stores time series of runup (IJOB<3)       RBR04390
C  (Output)  63 ('OOVER')    stores time series of hydrodyn. quantities RBR04400
```

```
C                              at landward boundary (IJOB=2)                RBR04410
C  (Output) 64 ('OTRANS')     stores time series of hydrodyn. quantities   RBR04420
C                              at landward boundary (IJOB=3)                RBR04430
C  (Output) 71 (FSAV1B)       stores time series of volume flux at         RBR04440
C                              specified nodes (one of "B")                 RBR04450
C  (Output) 72 (FSAV2B)       stores time series of water depth at         RBR04460
C                              specified nodes (the other "B")              RBR04470
C  (Output) 73 (FSAVEC)       stores time series of "C", i.e., time        RBR04480
C                              series of either of the following:           RBR04490
C                              . for ISTAB=1, stability number SNR at       RBR04500
C                                specified nodes                            RBR04510
C                              . for ISTAB=2, displacement of armor units   RBR04520
C                                from specified initial nodal locations     RBR04530
C  (Output) 98 ('ODOC')        stores essential output for concise          RBR04540
C                              documentation                                RBR04550
C  (Output) 99 ('OMSG')        stores messages written under special        RBR04560
C                              circumstances during computation             RBR04570
C  ------------------------------------------------------------------       RBR04580
      IF (ICALL.EQ.1) THEN                                                  RBR04590
        CALL OPENF (11,FINP1,'OLD')                                         RBR04600
        CALL OPENF (14,'BINPUT    ','OLD')                                  RBR04610
        CALL OPENF (51,'OSTAT     ','NEW')                                  RBR04620
        CALL OPENF (52,'OSPACE    ','NEW')                                  RBR04630
        CALL OPENF (61,'OSEAWAV   ','NEW')                                  RBR04640
        CALL OPENF (98,'ODOC      ','NEW')                                  RBR04650
        CALL OPENF (99,'OMSG      ','NEW')                                  RBR04660
      ELSEIF (ICALL.EQ.2) THEN                                              RBR04670
        IF (IJOB.LT.3) THEN                                                 RBR04680
          CALL OPENF (62,'ORUNUP    ','NEW')                                RBR04690
          IF (IJOB.EQ.2) CALL OPENF (63,'OOVER     ','NEW')                 RBR04700
        ELSE                                                                RBR04710
          CALL OPENF (64,'OTRANS    ','NEW')                                RBR04720
        ENDIF                                                               RBR04730
        IF (IWAVE.GT.1)  CALL OPENF (12,FINP2,'OLD')                        RBR04740
        IF (IENERG.EQ.1) CALL OPENF (53,'OENERG    ','NEW')                 RBR04750
        IF (ISTAB.GT.0)  CALL OPENF (54,'OSTAB     ','NEW')                 RBR04760
        IF (IREQ.EQ.1)   CALL OPENF (55,'OREQ      ','NEW')                 RBR04770
      ELSE                                                                  RBR04780
C       ------------------------------------------------------------        RBR04790
C       Storing "B" and "C" is storing a number of time series, each        RBR04800
C         being associated with a specified node, simultaneously. For       RBR04810
C         computation with long duration, a single output file may be       RBR04820
C         too large to store. Therefore, time series are stored in          RBR04830
C         groups of 100 wave periods, i.e., 100 wave periods to an          RBR04840
C         output file.                                                      RBR04850
C       "B" time series of the first kind (volume flux) are stored in       RBR04860
C         files 'OSAV1B01' (the first 100 waves), 'OSAV1B02' (the           RBR04870
C         second 100 waves), and so on (under variable FSAV1B and           RBR04880
C         unit number 71).                                                  RBR04890
C       "B" time series of the second kind (water depth) are stored in      RBR04900
C         files 'OSAV2B01', 'OSAV2B02', and so on (under variable           RBR04910
C         FSAV2B and unit number 72).                                       RBR04920
C       "C" time series are stored in files 'OSAVEC01', 'OSAVEC02',         RBR04930
C         and so on (under variable FSAVEC and unit number 73).             RBR04940
C       In the following, the opening and closing of applicable output      RBR04950
```

```
C         files are performed every 100 wave periods.                    RBR04960
C       ------------------------------------------------------------     RBR04970
        IDUM = MOD(M,100)                                                RBR04980
        IF (IDUM.EQ.1) THEN                                              RBR04990
          MPACK = M/100 + 1                                              RBR05000
          IF (ISAVB.EQ.1) THEN                                           RBR05010
C           ::    For storing "B"                                        RBR05020
            IF (M.GT.1) THEN                                             RBR05030
              CLOSE (71)                                                 RBR05040
              CLOSE (72)                                                 RBR05050
            ENDIF                                                        RBR05060
            CALL OPENF (71,FSAV1B(MPACK),'NEW')                          RBR05070
            CALL OPENF (72,FSAV2B(MPACK),'NEW')                          RBR05080
          ENDIF                                                          RBR05090
          IF (ISAVC.EQ.1) THEN                                           RBR05100
C           ::    For storing "C"                                        RBR05110
            IF (M.GT.1) CLOSE (73)                                       RBR05120
            CALL OPENF (73,FSAVEC(MPACK),'NEW')                          RBR05130
          ENDIF                                                          RBR05140
        ENDIF                                                            RBR05150
      ENDIF                                                              RBR05160
C                                                                        RBR05170
      RETURN                                                             RBR05180
      END                                                                RBR05190
C                                                                        RBR05200
C -01------------------  END OF SUBROUTINE OPENIO ------------RBREAK-    RBR05210
C #02#####################  SUBROUTINE INPUT1  ###############RBREAK#    RBR05220
C                                                                        RBR05230
C     This subroutine reads most of input data                           RBR05240
C                                                                        RBR05250
C     Notes:                                                             RBR05260
C   . User should run RBREAK after a successful run of BEFORR using     RBR05270
C     exactly the same input data set.                                   RBR05280
C   . User-chosen options and most of user-specified values are not      RBR05290
C     checked in RBREAK, assuming that they are unchanged from the       RBR05300
C     corresponding BEFORR computation. Only values related to arrays'   RBR05310
C     dimensions are checked again since RBREAK and BEFORR may have      RBR05320
C     different dimension arrangement.                                   RBR05330
C                                                                        RBR05340
      SUBROUTINE INPUT1 (MODE,M,FINP2)                                   RBR05350
C                                                                        RBR05360
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                RBR05370
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                       RBR05380
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                                RBR05390
      CHARACTER*10 FINP2,FID2                                            RBR05400
      CHARACTER*5 COMMEN(15)                                             RBR05410
      INTEGER S,SM1,SP1                                                  RBR05420
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                RBR05430
      COMMON /CONSTA/ PI,GRAV                                            RBR05440
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,         RBR05450
     +                ISAVA,ISAVB,ISAVC                                  RBR05460
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES             RBR05470
      COMMON /CPAR2/  MULTIF,NONEM,NONM,NONE,NEND,NRATE,NHOP             RBR05480
      COMMON /CPAR3/  INITS,JE,JE1                                       RBR05490
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX      RBR05500
```

```
      COMMON /REQ1/    IREQ,IELEV,IV,IDUDT,ISNR,NREQ                          RBR05510
      COMMON /REQ2/    TREQ(N5)                                               RBR05520
      COMMON /WAVE1/   HREFP,TP,WL0P                                          RBR05530
      COMMON /WAVE2/   KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA              RBR05540
      COMMON /WAVE6/   NDATA,NDAT                                             RBR05550
      COMMON /BOT1/    DSEAP,DLANDP,FWP                                       RBR05560
      COMMON /BOT2/    DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT          RBR05570
      COMMON /BOT4/    NBSEG                                                  RBR05580
      COMMON /BOT5/    WBSEG(N4),TBSLOP(N4),XBSEG(N4),ZBSEG(N4)               RBR05590
      COMMON /RUNP1/   NDELR,S,SM1,SP1,JMAX                                   RBR05600
      COMMON /RUNP2/   DELRP(N3),DELTAR(N3),RUNUPS(N3),RSTAT(3,N3),           RBR05610
     +                 RMEAN(N3)                                             RBR05620
      COMMON /ARMOR1/  C2,C3,CD,CL,CM,SG,TANPHI,AMIN,AMAX,DAP                 RBR05630
      COMMON /SAVBC/   NNODB,NNODC,NODB(N5),NODC(N5)                          RBR05640
      IF (MODE.EQ.0) THEN                                                     RBR05650
C       ::   MODE=0 is used in Subr. 48 CKSUBR                                RBR05660
C       ::   Although not used in any of the above COMMON blocks,             RBR05670
C            N1 and N2 need to be checked in relation with                    RBR05680
C            Call CKVAL (5,...) and checking NONEMA in this subroutine        RBR05690
        CALL CKFPAR (2,1,N1,N1R)                                              RBR05700
        CALL CKFPAR (2,2,N2,N2R)                                              RBR05710
        CALL CKFPAR (2,3,N3,N3R)                                              RBR05720
        CALL CKFPAR (2,4,N4,N4R)                                              RBR05730
        CALL CKFPAR (2,5,N5,N5R)                                              RBR05740
        RETURN                                                                RBR05750
      ELSEIF (MODE.EQ.9) THEN                                                 RBR05760
C       ::   MODE=9 is called from the main program                           RBR05770
C       ::   RBREAK reads NONE and DELTA from file 'BINPUT'                   RBR05780
        READ (14,1410) MI,NONE,NEND,DELTA                                     RBR05790
        IF (M.NE.MI) CALL STOPP (2,2)                                         RBR05800
        RETURN                                                                RBR05810
      ENDIF                                                                   RBR05820
C                                                                             RBR05830
C           COMMENT LINES                                                     RBR05840
C           ------------------------------------------------------------      RBR05850
C           NLINES = number of comment lines preceding primary                RBR05860
C                    input data                                               RBR05870
C           Comment lines of primary input file: read and then written        RBR05880
C           as the heading of output files 'ODOC' and 'OMSG'                  RBR05890
C           ------------------------------------------------------------      RBR05900
      READ  (11,1110) NLINES                                                  RBR05910
      DO 110 L = 1,NLINES                                                     RBR05920
        READ  (11,1120) (COMMEN(I),I=1,15)                                    RBR05930
        WRITE (98,1120) (COMMEN(I),I=1,15)                                    RBR05940
        WRITE (99,1120) (COMMEN(I),I=1,15)                                    RBR05950
  110 CONTINUE                                                                RBR05960
C                                                                             RBR05970
C           OPTIONS                                                           RBR05980
C           ------------------------------------------------------------      RBR05990
C           IJOB=1: RUNUP on impermeable slope                                RBR06000
C               =2: OVERTOPPING over subaerial structure                      RBR06010
C               =3: TRANSMISSION over submerged structure                     RBR06020
C           ISTAB=0: No computation of armor stability or movement            RBR06030
C                =1: Armor stability computation                              RBR06040
C                =2: Armor movement computation                               RBR06050
```

```
C           ISYST=1: International System of Units (SI) is used          RBR06060
C                =2: US Customary System of Units (USCS) is used         RBR06070
C           IBOT=1: "Type 1" bottom data (width-slope)                   RBR06080
C               =2: "Type 2" bottom data (coordinates)                   RBR06090
C           INONCT=0: No correction term in computing ETAR               RBR06100
C                 =1: Correction term for ETAR recommended for           RBR06110
C                     beaches                                            RBR06120
C           IENERG=0: Energy quantities NOT computed                     RBR06130
C                 =1: Energy quantities computed                         RBR06140
C           IWAVE=1: Incident waves at seaward boundary computed         RBR06150
C                =2: Incident waves at seaward boundary given as input   RBR06160
C                =3: Total waves at seaward boundary given as input      RBR06170
C           If IWAVE>1 --> Must specify MSTAT and FINP2                   RBR06180
C             . FINP2 = name of input data file containing time series   RBR06190
C               of normalized free surface elevation at seaw. boundary   RBR06200
C             . Statistical calculations are performed excluding the     RBR06210
C               first MSTAT computation units                            RBR06220
C             . A "computation unit" is one wave period long except      RBR06230
C               for the last unit, the length of which may be less       RBR06240
C               than one wave period                                     RBR06250
C             . The "wave period" is the reference period used for the   RBR06260
C               normalization of the governing equations                 RBR06270
C           "A" = Spatial variations of hydrodynamic quantities at       RBR06280
C                 specified moments                                      RBR06290
C           "B" = Time series of volume flux and water depth at          RBR06300
C                 specified nodes                                        RBR06310
C           "C" = Time series of either of the following:                RBR06320
C               . For ISTAB=1: stability number SNR at specified nodes   RBR06330
C               . For ISTAB=2: displacement of armor units from          RBR06340
C                              specified initial nodal locations         RBR06350
C           "B" and "C" are stored at rate NRATE data points per wave    RBR06360
C                 period                                                 RBR06370
C           ISAVA,ISAVB,ISAVC are identifiers associated with saving     RBR06380
C                 "A","B","C", respectively (1=save; 0=no)               RBR06390
C           MSAVA1, MSAVA2, AND NTIMES:                                  RBR06400
C           If ISAVA=1, "A" is saved NTIMES (>1) times at equal          RBR06410
C                         intervals per wave period from computation     RBR06420
C                         unit MSAVA1 to MSAVA2, inclusive               RBR06430
C           If ISAVB=1 --> Must specify NNODB                            RBR06440
C           If ISAVC=1 --> Must specify NNODC                            RBR06450
C           NNODB,NNODC = number of nodes for which "B","C" is to be     RBR06460
C                         saved                                          RBR06470
C           IREQ=0: No special storing                                   RBR06480
C               =1: Special storing requested                            RBR06490
C           Special storing = storing spatial variations of requested    RBR06500
C               quantities at normalized time TREQ(i)                    RBR06510
C               with i=1,2,...,NREQ                                      RBR06520
C           Quantities available for request:                            RBR06530
C           . ELEV = surface elevation                                   RBR06540
C           . V    = depth-averaged velocity                             RBR06550
C           . DUDT = total fluid acceleration                            RBR06560
C           . SNR  = stability number against rolling/sliding            RBR06570
C           --> requested by IELEV=1, IV=1, IDUDT=1, and ISNR=1,         RBR06580
C               respectively                                             RBR06590
C           Notes: . DUDT can be requested only if ISTAB>0               RBR06600
```

```
C                . SNR  can be requested only if ISTAB=1                RBR06610
C          ------------------------------------------------------------ RBR06620
      READ  (11,1130) IJOB,ISTAB                                        RBR06630
      READ  (11,1130) ISYST                                             RBR06640
      READ  (11,1130) IBOT                                              RBR06650
      READ  (11,1130) INONCT                                            RBR06660
      READ  (11,1130) IENERG                                            RBR06670
      READ  (11,1140) IWAVE,MSTAT,FINP2                                 RBR06680
      READ  (11,1150) ISAVA,MSAVA1,MSAVA2,NTIMES                        RBR06690
      READ  (11,1150) ISAVB,NNODB                                       RBR06700
      READ  (11,1150) ISAVC,NNODC                                       RBR06710
      READ  (11,1160) IREQ,IELEV,IV,IDUDT,ISNR,NREQ                     RBR06720
      IF (IREQ.EQ.1) THEN                                               RBR06730
        READ (11,1180) (TREQ(I),I=1,NREQ)                               RBR06740
        CALL CKVAL (1,NREQ,N5)                                          RBR06750
      ENDIF                                                             RBR06760
      IF (ISAVB.EQ.1) CALL CKVAL (2,NNODB,N5)                           RBR06770
      IF (ISAVC.EQ.1) CALL CKVAL (3,NNODC,N5)                           RBR06780
C                                                                       RBR06790
C          COMPUTATION PARAMETERS                                       RBR06800
C          ------------------------------------------------------------ RBR06810
C          INITS  = . number of spatial nodes along the bottom          RBR06820
C                     below SWL (IJOB<3)                                RBR06830
C                   . number of nodes between seaward and               RBR06840
C                     landward boundaries, inclusive (IJOB=3)           RBR06850
C          MULTIF = multiplication factor associated with NONEM         RBR06860
C                   . NONEM = minimum allowable value of NONE           RBR06870
C                   . NONE  = number of time steps in one wave period   RBR06880
C                   . See Subr. 10 CPARAM for more explanation          RBR06890
C          DELTA as input is an initial value for normalized water      RBR06900
C                depth defining computational waterline                 RBR06910
C          TMAX  = normalized duration of computation                   RBR06920
C          NRATE = rate (data points per per wave period) of storing    RBR06930
C                  output time series                                   RBR06940
C          NDELR = number of "DELRP"s to be specified                   RBR06950
C          DELRP = physical water depth associated with visual or       RBR06960
C                  measured waterline                                   RBR06970
C                  . in millimeters if ISYST=1 (SI)                     RBR06980
C                  . in inches      if ISYST=2 (USCS)                   RBR06990
C          X1,X2 = damping coefficients                                 RBR07000
C          MWAVE = number of computation units                          RBR07010
C          MSTAT: Statistical calculations are performed excluding      RBR07020
C                 the first MSTAT computation units                     RBR07030
C          MSTAB: Computation of armor stability or movement is per-    RBR07040
C                 formed excluding the first MSTAB computation units    RBR07050
C          ------------------------------------------------------------ RBR07060
      READ  (11,1110) INITS                                             RBR07070
      READ  (11,1110) MULTIF                                            RBR07080
      READ  (11,1170) DELTA                                             RBR07090
      READ  (11,1170) TMAX                                              RBR07100
      READ  (11,1110) NRATE                                             RBR07110
      READ  (11,1110) NDELR                                             RBR07120
      CALL CKVAL (5,INITS,N1-1)                                         RBR07130
      IF (IJOB.LT.3) THEN                                               RBR07140
        CALL CKVAL (4,NDELR,N3)                                         RBR07150
```

```
      ELSE                                                              RBR07160
        NDELR = 0                                                       RBR07170
      ENDIF                                                             RBR07180
      DO 130 L = 1,NDELR                                                RBR07190
        READ (11,1180) DELRP(L)                                         RBR07200
  130 CONTINUE                                                          RBR07210
      READ (11,1180) X1,X2                                              RBR07220
      ITMAX = INT(TMAX)                                                 RBR07230
      RES   = DMOD(TMAX,1.D+00)                                         RBR07240
      IF (DABS(RES).LT.1.D-08) THEN                                     RBR07250
        MWAVE = ITMAX                                                   RBR07260
      ELSE                                                              RBR07270
        MWAVE = ITMAX+1                                                 RBR07280
      ENDIF                                                             RBR07290
      IF (IWAVE.EQ.1) THEN                                              RBR07300
        MSTAT = MWAVE-1                                                 RBR07310
      ENDIF                                                             RBR07320
      IF (ISTAB.EQ.1) THEN                                              RBR07330
        MSTAB = MSTAT                                                   RBR07340
      ELSEIF (ISTAB.EQ.2) THEN                                          RBR07350
        MSTAB = 0                                                       RBR07360
      ELSE                                                              RBR07370
        MSTAB = MWAVE+1                                                 RBR07380
        ISAVC = 0                                                       RBR07390
      ENDIF                                                             RBR07400
      IF (ISAVA.EQ.0) MSAVA1=MWAVE+1                                    RBR07410
C                                                                       RBR07420
C           CHECK CASE-SIGNATURES                                       RBR07430
C           to make sure that RBREAK reads the correct file 'BINPUT'    RBR07440
C           ------------------------------------------------------      RBR07450
C           File 'BINPUT' was created by BEFORR. It contains, among     RBR07460
C           other things, four pieces of information which are case-    RBR07470
C           specific, and referred to as "case-signatures".             RBR07480
C           On the left are signatures read from file 'BINPUT', on the  RBR07490
C           right their RBREAK counterparts:                            RBR07500
C           . MWAV - MWAVE                                              RBR07510
C           . FID2 - FINP2                                              RBR07520
C           . NDAT - NDATA                                              RBR07530
C           . NONM - NONEM                                              RBR07540
C           Left must match with right. Check is conducted in:          RBR07550
C           . this subroutine (the first two)                           RBR07560
C           . Subr. 03 INPUT2 (the third)                               RBR07570
C           . Subr. 10 CPARAM (the last)                                RBR07580
C           ------------------------------------------------------      RBR07590
      READ  (14,1110) NLINES                                            RBR07600
      DO 135 L = 1,NLINES                                               RBR07610
        READ (14,1120) (COMMEN(I),I=1,15)                               RBR07620
  135 CONTINUE                                                          RBR07630
      READ  (14,1110) MWAV                                              RBR07640
      IF (MWAV.NE.MWAVE) CALL STOPP (3,3)                               RBR07650
      IF (IWAVE.GT.1) THEN                                              RBR07660
        READ (14,5000) FID2                                             RBR07670
        IF (FID2.NE.FINP2) CALL STOPP (4,4)                             RBR07680
        READ (14,1110) NDAT                                             RBR07690
      ENDIF                                                             RBR07700
```

```
      READ  (14,1110) NONM                                                  RBR07710
C                                                                           RBR07720
C           CHECK PARAMETER N2                                              RBR07730
C           only for regular wave computations                              RBR07740
C           ------------------------------------------------------          RBR07750
C           Read NONEMA from file 'BINPUT'                                  RBR07760
C           NONEMA = largest NONE to be encountered in computation          RBR07770
C                    as determined by BEFORR                                RBR07780
C           NONE   = number of time steps in one wave period                RBR07790
C           PARAMETER N2 provides dimension for variable ETA, which         RBR07800
C           represents wave profile. For regular waves, wave profile        RBR07810
C           is of size (NONE+1). Thus, N2 must be at least (NONEMA+1)       RBR07820
C           ------------------------------------------------------          RBR07830
      IF (IWAVE.EQ.1) THEN                                                  RBR07840
        READ (14,1110) NONEMA                                               RBR07850
        IF (NONEMA.GT.(N2-1)) THEN                                          RBR07860
          WRITE  (*,9910) NONEMA,N2                                         RBR07870
          WRITE (99,9910) NONEMA,N2                                         RBR07880
          STOP                                                              RBR07890
        ENDIF                                                               RBR07900
      ENDIF                                                                 RBR07910
C                                                                           RBR07920
C           CONSTANTS                                                       RBR07930
C           ------------------------------------------------------          RBR07940
C           PI   = 3.141592...                                              RBR07950
C           GRAV = gravitational acceleration                               RBR07960
C                  . in m/sec**2  if ISYST=1 (SI)                           RBR07970
C                  . in ft/sec**2 if ISYST=2 (USCS)                         RBR07980
C           ------------------------------------------------------          RBR07990
      PI = 4.D+00*DATAN(1.D+00)                                             RBR08000
      IF (ISYST.EQ.1) THEN                                                  RBR08010
        GRAV = 9.81D+00                                                     RBR08020
      ELSE                                                                  RBR08030
        GRAV = 32.2D+00                                                     RBR08040
      ENDIF                                                                 RBR08050
C                                                                           RBR08060
C           WAVE PROPERTIES AND FRICTION FACTOR                             RBR08070
C           ------------------------------------------------------          RBR08080
C           HREFP = physical wave height at "reference" location            RBR08090
C                   . in meters if ISYST=1 (SI)                             RBR08100
C                   . in feet   if ISYST=2 (USCS)                           RBR08110
C           TP    = physical reference wave period, in seconds              RBR08120
C           HREFP and TP are used to normalize the governing                RBR08130
C             equations                                                     RBR08140
C           KSREF = shoaling coefficient at "reference" location            RBR08150
C           KSSEA = shoaling coefficient at seaward boundary                RBR08160
C           SIGMA is a measure of wave steepness                            RBR08170
C           FWP   = bottom friction factor                                  RBR08180
C           ------------------------------------------------------          RBR08190
      READ  (11,1180) HREFP,TP                                              RBR08200
      READ  (11,1180) KSREF,KSSEA                                           RBR08210
      READ  (11,1180) FWP                                                   RBR08220
      SIGMA = TP*DSQRT(GRAV/HREFP)                                          RBR08230
C                                                                           RBR08240
C           STRUCTURE GEOMETRY                                              RBR08250
```

```
C           ------------------------------------------------------------ RBR08260
C           The structure geometry is divided into segments of           RBR08270
C           different inclination                                        RBR08280
C           NBSEG  = number of segments                                  RBR08290
C           DSEAP  = physical water depth below SWL at seaward           RBR08300
C                    boundary                                            RBR08310
C           TSLOPS = tangent of slope used to define                     RBR08320
C                    "surf similarity parameter"                         RBR08330
C           For segment i starting from the seaward boundary:            RBR08340
C             WBSEG(i)  = physical horizontal width                      RBR08350
C             TBSLOP(i) = tangent of slope (+ upslope, - downslope)      RBR08360
C             XBSEG(i)  = physical horizontal distance from seaward      RBR08370
C                         boundary to the segment's seaward-end          RBR08380
C             ZBSEG(i)  = physical water depth below SWL (+ below SWL)   RBR08390
C                         at the segment's seaward-end                   RBR08400
C           DSEAP,WBSEG,XBSEG,ZBSEG are in meters if ISYST=1 (SI),       RBR08410
C                                           feet   if ISYST=2 (USCS)     RBR08420
C           ------------------------------------------------------------ RBR08430
      READ  (11,1180) DSEAP                                              RBR08440
      READ  (11,1180) TSLOPS                                             RBR08450
      READ  (11,1110) NBSEG                                              RBR08460
      CALL CKVAL (6,NBSEG,N4-1)                                          RBR08470
      IF (IBOT.EQ.1) THEN                                                RBR08480
        DO 140 K = 1,NBSEG                                               RBR08490
          READ (11,1180) WBSEG(K),TBSLOP(K)                              RBR08500
  140   CONTINUE                                                         RBR08510
      ELSE                                                               RBR08520
        DO 150 K = 1,NBSEG+1                                             RBR08530
          READ (11,1180) XBSEG(K),ZBSEG(K)                               RBR08540
  150   CONTINUE                                                         RBR08550
      ENDIF                                                              RBR08560
C                                                                        RBR08570
C           DATA RELATED TO SAVING "B", i.e., time series of volume      RBR08580
C           flux and total water depth at specified nodes                RBR08590
C           ------------------------------------------------------------ RBR08600
C           NNODB   = no. of nodes for which "B" is to be saved          RBR08610
C           NODB(I) = I-th node number for which "B" is to be saved      RBR08620
C           ------------------------------------------------------------ RBR08630
      IF (ISAVB.EQ.1) READ (11,1190) (NODB(I),I=1,NNODB)                 RBR08640
C                                                                        RBR08650
C           DATA RELATED TO ARMOR STABILITY AND MOVEMENT                 RBR08660
C           ------------------------------------------------------------ RBR08670
C           SG     = specific gravity                                    RBR08680
C           C2     = area coefficient                                    RBR08690
C           C3     = volume coefficient                                  RBR08700
C           CD     = drag coefficient                                    RBR08710
C           CL     = lift coefficient                                    RBR08720
C           CM     = inertia coefficient                                 RBR08730
C           TANPHI = armor friction factor                               RBR08740
C           AMAX,AMIN = upper and lower bounds of fluid acceleration,    RBR08750
C                    normalized by gravitational acceleration, used      RBR08760
C                    only for ISTAB=1                                    RBR08770
C           DAP    = physical armor diameter                             RBR08780
C                    . in meters IF ISYST=1 (SI)                         RBR08790
C                    . in feet   IF ISYST=2 (USCS)                       RBR08800
```

```
C            NNODC   = no. of nodes for which "C" is to be saved          RBR08810
C            NODC(I) = I-th node number for which "C" is to be saved      RBR08820
C            "C" = time series of the following                           RBR08830
C                . For ISTAB=1: stability number SNR at specified nodes   RBR08840
C                . For ISTAB=2: displacement of armor units from          RBR08850
C                               specified initial nodal locations         RBR08860
C            ------------------------------------------------------       RBR08870
C     ::     To compute SNR = stability number against rolling/sliding:   RBR08880
      IF (ISTAB.GT.0) THEN                                                 RBR08890
        READ (11,1180) C2,C3,SG                                            RBR08900
        READ (11,1180) CD,CL,CM                                            RBR08910
        READ (11,1180) TANPHI                                              RBR08920
        READ (11,1180) AMAX,AMIN                                           RBR08930
        IF (ISAVC.EQ.1) READ (11,1190) (NODC(I),I=1,NNODC)                 RBR08940
      ENDIF                                                                RBR08950
C     ::     To compute movement of armor units, additional input is      RBR08960
C            required:                                                    RBR08970
      IF (ISTAB.EQ.2) READ (11,1180) DAP                                   RBR08980
C                                                                          RBR08990
C              FORMATS                                                     RBR09000
C              ------------------------------------------------------      RBR09010
 1110 FORMAT (3I8)                                                         RBR09020
 1120 FORMAT (15A5)                                                        RBR09030
 1130 FORMAT (2I1)                                                         RBR09040
 1140 FORMAT (I1,I8,2X,A10)                                                RBR09050
 1150 FORMAT (I1,3I8)                                                      RBR09060
 1160 FORMAT (5I1,I8)                                                      RBR09070
 1170 FORMAT (F15.6)                                                       RBR09080
 1180 FORMAT (3F13.6)                                                      RBR09090
 1190 FORMAT (5I6)                                                         RBR09100
 1410 FORMAT (3I8,D18.9)                                                   RBR09110
 5000 FORMAT (A10)                                                         RBR09120
 9910 FORMAT (/' From Subr. 02 INPUT1:'/                                   RBR09130
     +         ' Computation will need NONE as large as',I8,'.'/           RBR09140
     +         ' Dimension N2 must be greater than NONE.'/                 RBR09150
     +         ' Currently, N2 =',I8,'.'/                                  RBR09160
     +         ' Specify larger N2 as required.'/                          RBR09170
     +         ' Computation aborted.')                                    RBR09180
C                                                                          RBR09190
      RETURN                                                               RBR09200
      END                                                                  RBR09210
C                                                                          RBR09220
C -02-----------------  END OF SUBROUTINE INPUT1  ------------RBREAK-      RBR09230
C #03#####################  SUBROUTINE INPUT2  ###############RBREAK#     RBR09240
C                                                                          RBR09250
C     This subroutine reads wave profile at seaward boundary               RBR09260
C                                                                          RBR09270
      SUBROUTINE INPUT2 (MODE)                                             RBR09280
C                                                                          RBR09290
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                  RBR09300
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                         RBR09310
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                  RBR09320
      COMMON /WAVE3/  ETA(N2),ETAIS(N2),ETARS(N2),ETATS(N2)                RBR09330
      COMMON /WAVE4/  ETAMAX,ETAMIN                                        RBR09340
      COMMON /WAVE6/  NDATA,NDAT                                           RBR09350
```

```
      IF (MODE.EQ.0) THEN                                               RBR09360
        CALL CKFPAR (3,2,N2,N2R)                                        RBR09370
        RETURN                                                          RBR09380
      ENDIF                                                             RBR09390
C                                                                       RBR09400
C             READ DATA POINTS                                          RBR09410
C             ------------------------------------------------          RBR09420
      READ (12,9000) NDATA                                              RBR09430
      CALL CKVAL (7,NDATA,N2)                                           RBR09440
      READ (12,8000) (ETA(I),I=1,NDATA)                                 RBR09450
 8000 FORMAT (5D15.6)                                                   RBR09460
 9000 FORMAT (I8)                                                       RBR09470
C                                                                       RBR09480
C             CHECK CASE-SIGNATURE NDAT                                 RBR09490
C             ------------------------------------------------          RBR09500
      IF (NDAT.NE.NDATA) CALL STOPP (12,13)                             RBR09510
C                                                                       RBR09520
C             FIND EXTREME VALUES                                       RBR09530
C             ------------------------------------------------          RBR09540
C             ETA = given time series of free surface profile at        RBR09550
C                   seaward boundary                                    RBR09560
C             ETAMAX,ETAMIN = maximum,minimum of ETA                    RBR09570
C             ------------------------------------------------          RBR09580
      ETAMAX = -1.D+03                                                  RBR09590
      ETAMIN =  1.D+03                                                  RBR09600
      DO 100 I = 1,NDATA                                                RBR09610
        IF (ETA(I).GT.ETAMAX) ETAMAX=ETA(I)                             RBR09620
        IF (ETA(I).LT.ETAMIN) ETAMIN=ETA(I)                             RBR09630
  100 CONTINUE                                                          RBR09640
C                                                                       RBR09650
      RETURN                                                            RBR09660
      END                                                               RBR09670
C                                                                       RBR09680
C -03-------------------  END OF SUBROUTINE INPUT2 ------------RBREAK-  RBR09690
C #04#######################  SUBROUTINE BOTTOM  ###############RBREAK#  RBR09700
C                                                                       RBR09710
C     This subroutine calculates normalized structure geometry and      RBR09720
C     delta x between two adjacent nodes                                RBR09730
C                                                                       RBR09740
      SUBROUTINE BOTTOM (MODE)                                          RBR09750
C                                                                       RBR09760
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR09770
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR09780
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                               RBR09790
      DIMENSION TSLOPE(N1)                                              RBR09800
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR09810
      COMMON /CONSTA/ PI,GRAV                                           RBR09820
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,        RBR09830
     +                ISAVA,ISAVB,ISAVC                                 RBR09840
      COMMON /CPAR3/  INITS,JE,JE1                                      RBR09850
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR09860
      COMMON /WAVE1/  HREFP,TP,WL0P                                     RBR09870
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA          RBR09880
      COMMON /BOT1/   DSEAP,DLANDP,FWP                                  RBR09890
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     RBR09900
```

```
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)      RBR09910
      COMMON /BOT4/   NBSEG                                              RBR09920
      COMMON /BOT5/   WBSEG(N4),TBSLOP(N4),XBSEG(N4),ZBSEG(N4)           RBR09930
      COMMON /ARMOR1/ C2,C3,CD,CL,CM,SG,TANPHI,AMIN,AMAX,DAP             RBR09940
      COMMON /ARMOR2/ SG1,CTAN(N1)                                       RBR09950
      COMMON /SAVBC/  NNODB,NNODC,NODB(N5),NODC(N5)                      RBR09960
      IF (MODE.EQ.0) THEN                                                RBR09970
        CALL CKFPAR (4,1,N1,N1R)                                         RBR09980
        CALL CKFPAR (4,4,N4,N4R)                                         RBR09990
        CALL CKFPAR (4,5,N5,N5R)                                         RBR10000
        RETURN                                                           RBR10010
      ENDIF                                                              RBR10020
C                                                                        RBR10030
C           COMPLETE SEGMENT DATA NOT SPECIFIED AS INPUT                 RBR10040
C           (The following variables are dimensional)                    RBR10050
C           ---------------------------------------------------------    RBR10060
C           TSLOPS = tangent of slope used to define                     RBR10070
C                    "surf similarity parameter"                         RBR10080
C           BSWL: . for IJOB<3: physical horizontal distance between     RBR10090
C                   seaward boundary and initial waterline on slope      RBR10100
C                 . for IJOB=3: physical horizontal distance between     RBR10110
C                   seaward and landward boundaries                      RBR10120
C           DSEAP = water depth below SWL at seaward boundary            RBR10130
C           The structure geometry is divided into segments of           RBR10140
C           different inclination                                        RBR10150
C           NBSEG = number of segments                                   RBR10160
C           For segment i starting from the seaward boundary:            RBR10170
C           . WBSEG(i)  = physical horizontal width                      RBR10180
C           . TBSLOP(i) = tangent of slope (+ upslope, - downslope)      RBR10190
C           . XBSEG(i)  = physical horizontal distance from seaward      RBR10200
C                         boundary to the segment's seaward-end          RBR10210
C           . ZBSEG(i)  = physical water depth below SWL                 RBR10220
C                         (+ below SWL) at the segment's seaward-end     RBR10230
C           BSWL,DSEAP,WBSEG,XBSEG,ZBSEG are                             RBR10240
C           . in meters if ISYST=1 (SI)                                  RBR10250
C           . in feet   if ISYST=2 (USCS)                                RBR10260
C           ---------------------------------------------------------    RBR10270
      IF (IBOT.EQ.1) THEN                                                RBR10280
        DCUM     = 0.D+00                                                RBR10290
        XBSEG(1) = 0.D+00                                                RBR10300
        ZBSEG(1) = DSEAP                                                 RBR10310
        DO 110 K = 2,NBSEG+1                                             RBR10320
          DCUM     = DCUM + WBSEG(K-1)*TBSLOP(K-1)                       RBR10330
          XBSEG(K) = XBSEG(K-1) + WBSEG(K-1)                             RBR10340
          ZBSEG(K) = DSEAP - DCUM                                        RBR10350
  110   CONTINUE                                                         RBR10360
      ELSE                                                               RBR10370
        DO 120 K = 1,NBSEG                                               RBR10380
          TBSLOP(K) = -(ZBSEG(K+1)-ZBSEG(K))/(XBSEG(K+1)-XBSEG(K))       RBR10390
  120   CONTINUE                                                         RBR10400
      ENDIF                                                              RBR10410
C                                                                        RBR10420
C           CALCULATE GRID SPACING X BETWEEN TWO ADJACENT NODES          RBR10430
C           (dimensional)                                                RBR10440
C           ---------------------------------------------------------    RBR10450
```

```
C           INITS = . number of spatial nodes along the bottom below    RBR10460
C                     SWL (IJOB<3)                                      RBR10470
C                   . number of nodes between seaward and landward      RBR10480
C                     boundaries, inclusive (IJOB=3)                    RBR10490
C           ------------------------------------------------------------ RBR10500
      IF (IJOB.LT.3) THEN                                                RBR10510
        K = 0                                                            RBR10520
  910   CONTINUE                                                         RBR10530
          IF (K.EQ.NBSEG) CALL STOPP (5,6)                               RBR10540
          K = K+1                                                        RBR10550
          CROSS = ZBSEG(K)*ZBSEG(K+1)                                    RBR10560
        IF (CROSS.GT.0.D+00) GOTO 910                                    RBR10570
        BSWL = XBSEG(K+1) + ZBSEG(K+1)/TBSLOP(K)                         RBR10580
        X    = BSWL/DBLE(INITS)                                          RBR10590
      ELSE                                                               RBR10600
        BSWL = XBSEG(NBSEG+1)                                            RBR10610
        X    = BSWL/DBLE(INITS-1)                                        RBR10620
        DO 130 K = 1,NBSEG+1                                             RBR10630
          IF (ZBSEG(K).LT.0.D+00) CALL STOPP (7,8)                       RBR10640
  130   CONTINUE                                                         RBR10650
      ENDIF                                                              RBR10660
C                                                                        RBR10670
C           CALCULATE STRUCTURE GEOMETRY AT EACH NODE (dimensional)      RBR10680
C           ------------------------------------------------------------ RBR10690
C           JE = landward edge node (IJOB<3) or landward boundary node   RBR10700
C                (IJOB=3)                                                RBR10710
C           U2INIT(j) = water depth below SWL at node j (+ below SWL)    RBR10720
C                     = total water depth U(2,j) at time t=0 (physical,  RBR10730
C                       later normalized under the same name)            RBR10740
C           TSLOPE(j) = tangent of local slope at node j                 RBR10750
C           ------------------------------------------------------------ RBR10760
      IF (IJOB.LT.3) THEN                                                RBR10770
        DUM = XBSEG(NBSEG+1)/X                                           RBR10780
        JE  = INT(DUM)+1                                                 RBR10790
      ELSE                                                               RBR10800
        JE = INITS                                                       RBR10810
      ENDIF                                                              RBR10820
      IF (JE.GT.N1) THEN                                                 RBR10830
        WRITE  (*,9910) JE,N1                                            RBR10840
        WRITE (99,9910) JE,N1                                            RBR10850
        STOP                                                             RBR10860
      ELSE                                                               RBR10870
        JE1 = JE-1                                                       RBR10880
      ENDIF                                                              RBR10890
C     :: Some checks associated with storing time series                 RBR10900
      IF (ISAVB.EQ.1) THEN                                               RBR10910
        DO 140 I = 1,NNODB                                               RBR10920
          CALL CKVAL (8,NODB(I),JE)                                      RBR10930
  140   CONTINUE                                                         RBR10940
      ENDIF                                                              RBR10950
      IF (ISAVC.EQ.1) THEN                                               RBR10960
        DO 150 I = 1,NNODC                                               RBR10970
          CALL CKVAL (9,NODC(I),JE)                                      RBR10980
  150   CONTINUE                                                         RBR10990
      ENDIF                                                              RBR11000
```

```
 9910 FORMAT (/' End Node =',I8,'; N1 =',I8/                          RBR11010
     +        ' Slope/Structure is too long.'/                        RBR11020
     +        ' Cut it, or change PARAMETER N1.')                     RBR11030
C                                                                     RBR11040
      DIST = -X                                                       RBR11050
      K    = 1                                                        RBR11060
      XCUM = XBSEG(K+1)                                               RBR11070
      DO 160 J = 1,JE                                                 RBR11080
        DIST = DIST + X                                               RBR11090
        IF (DIST.GT.XCUM.AND.K.LT.NBSEG) THEN                         RBR11100
  920     CONTINUE                                                    RBR11110
            K    = K+1                                                RBR11120
            XCUM = XBSEG(K+1)                                         RBR11130
          IF (DIST.GT.XCUM.AND.K.LT.NBSEG) GOTO 920                   RBR11140
        ENDIF                                                         RBR11150
        U2INIT(J) = ZBSEG(K) - (DIST-XBSEG(K))*TBSLOP(K)              RBR11160
        TSLOPE(J) = TBSLOP(K)                                         RBR11170
  160 CONTINUE                                                        RBR11180
C                                                                     RBR11190
C           NORMALIZATION                                             RBR11200
C           ------------------------------------------------------    RBR11210
C           WTOT = normalized width of computation domain             RBR11220
C           At node j:                                                RBR11230
C           . U2INIT(j) = normalized water depth below SWL            RBR11240
C                         (+ below SWL)                               RBR11250
C           . THETA(j)  = normalized tangent of local slope           RBR11260
C           . (XB(j),ZB(j)) = normalized coordinates of the structure RBR11270
C           ------------------------------------------------------    RBR11280
      DUM  = TP*DSQRT(GRAV*HREFP)                                     RBR11290
      X    = X/DUM                                                    RBR11300
      DIST = -X                                                       RBR11310
      WTOT = DBLE(JE1)*X                                              RBR11320
      DO 170 J = 1,JE                                                 RBR11330
        U2INIT(J) = U2INIT(J)/HREFP                                   RBR11340
        THETA(J)  = TSLOPE(J)*SIGMA                                   RBR11350
        DIST      = DIST + X                                          RBR11360
        XB(J)     = DIST                                              RBR11370
        ZB(J)     = -U2INIT(J)                                        RBR11380
  170 CONTINUE                                                        RBR11390
C                                                                     RBR11400
C           QUANTITIES NEEDED FOR COMPUTATION OF ARMOR STABILITY AND  RBR11410
C           MOVEMENT                                                  RBR11420
C           ------------------------------------------------------    RBR11430
C           TSLOPE(j) = tangent of local slope at node j              RBR11440
C           SSLOPE(j) = sine of local slope at node j                 RBR11450
C           CSLOPE = cosine of local slope                            RBR11460
C           TANPHI = armor friction factor                            RBR11470
C           ------------------------------------------------------    RBR11480
      IF (ISTAB.GT.0) THEN                                            RBR11490
        DO 180 J = 1,JE                                               RBR11500
          ANGLE     = DATAN(TSLOPE(J))                              RBR11510
          CSLOPE    = DCOS(ANGLE)                                   RBR11520
          SSLOPE(J) = DSIN(ANGLE)                                   RBR11530
          CTAN(J)   = CSLOPE*TANPHI                                 RBR11540
  180   CONTINUE                                                    RBR11550
```

```
      ENDIF                                                              RBR11560
C                                                                        RBR11570
      RETURN                                                             RBR11580
      END                                                                RBR11590
C                                                                        RBR11600
C -04------------------  END OF SUBROUTINE BOTTOM  ------------RBREAK-   RBR11610
C #05#####################  SUBROUTINE WPARAM  ###############RBREAK#   RBR11620
C                                                                        RBR11630
C     This subroutine calculates wave parameters                         RBR11640
C                                                                        RBR11650
      SUBROUTINE WPARAM (MODE)                                           RBR11660
C                                                                        RBR11670
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                RBR11680
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                       RBR11690
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                                RBR11700
      INTEGER S,SM1,SP1                                                  RBR11710
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                RBR11720
      COMMON /CONSTA/ PI,GRAV                                            RBR11730
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,         RBR11740
     +                ISAVA,ISAVB,ISAVC                                  RBR11750
      COMMON /CPAR3/  INITS,JE,JE1                                       RBR11760
      COMMON /WAVE1/  HREFP,TP,WL0P                                      RBR11770
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA           RBR11780
      COMMON /BOT1/   DSEAP,DLANDP,FWP                                   RBR11790
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT      RBR11800
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)      RBR11810
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                               RBR11820
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3),RUNUPS(N3),RSTAT(3,N3),       RBR11830
     +                RMEAN(N3)                                          RBR11840
      IF (MODE.EQ.0) THEN                                                RBR11850
        CALL CKFPAR (5,1,N1,N1R)                                         RBR11860
        CALL CKFPAR (5,3,N3,N3R)                                         RBR11870
        RETURN                                                           RBR11880
      ENDIF                                                              RBR11890
C                                                                        RBR11900
C           PARAMETERS RELATED TO WAVE AND SLOPE CHARACTERISTICS         RBR11910
C           ---------------------------------------------------------    RBR11920
C           KSI    = surf similarity parameter                           RBR11930
C           WL0P   = physical deep-water wavelength                      RBR11940
C           DSEAP  = phys. water depth below SWL at seaward boundary     RBR11950
C           DLANDP = phys. water depth below SWL at landward boundary    RBR11960
C                    (only for IJOB=3)                                   RBR11970
C           FWP    = physical slope friction factor                      RBR11980
C           DELRP  = water depths associated with visual or measured     RBR11990
C                    waterline                                           RBR12000
C           WL0,DSEA,DLAND,FW,DELTAR are the normalized counterparts     RBR12010
C           of WL0P,DSEAP,DLANDP,FWP,DELRP, respectively                 RBR12020
C           ---------------------------------------------------------    RBR12030
      WL0P   = GRAV*(TP*TP)/(2.D+00*PI)                                  RBR12040
      WL0    = WL0P/DSEAP                                                RBR12050
      KS     = KSSEA/KSREF                                               RBR12060
      KSI    = SIGMA*TSLOPS/DSQRT(2.D+00*PI)                             RBR12070
      DSEA   = DSEAP/HREFP                                               RBR12080
      DSEAKS = DSEA/KS                                                   RBR12090
      DSEA2  = DSQRT(DSEA)                                               RBR12100
```

```
      FW     = .5D+00*FWP*SIGMA                                          RBR12110
      DO 110 L = 1,NDELR                                                 RBR12120
        IF (ISYST.EQ.1) THEN                                             RBR12130
          DELTAR(L) = DELRP(L)/(1.D+03*HREFP)                            RBR12140
        ELSE                                                             RBR12150
          DELTAR(L) = DELRP(L)/(12.D+00*HREFP)                           RBR12160
        ENDIF                                                            RBR12170
  110 CONTINUE                                                           RBR12180
      IF (IJOB.EQ.3) THEN                                                RBR12190
        DLAND  = U2INIT(INITS)                                           RBR12200
        DLANDP = DLAND*HREFP                                             RBR12210
        DLAND2 = DSQRT(DLAND)                                            RBR12220
      ENDIF                                                              RBR12230
C                                                                        RBR12240
C           LINEAR WAVELENGTH AND PRELIMINARY URSELL NUMBER              RBR12250
C           ---------------------------------------------------          RBR12260
C           WL = normalized linear wavelength at seaward boundary        RBR12270
C           UR = Ursell number at seaward boundary based on linear       RBR12280
C                wavelength                                              RBR12290
C           ---------------------------------------------------          RBR12300
      TWOPI = 2.D+00*PI                                                  RBR12310
      WL    = WL0                                                        RBR12320
      FUN1  = WL - WL0*DTANH(TWOPI/WL)                                   RBR12330
  900 IF (DABS(FUN1).GT.1.D-04) THEN                                     RBR12340
        FUN2 = 1.D+00 + WL0*TWOPI/(WL*DCOSH(TWOPI/WL))**2                RBR12350
        WL   = WL - FUN1/FUN2                                            RBR12360
        FUN1 = WL - WL0*DTANH(TWOPI/WL)                                  RBR12370
      GOTO 900                                                           RBR12380
      ENDIF                                                              RBR12390
      UR    = WL*WL/DSEAKS                                               RBR12400
      URPRE = UR                                                         RBR12410
C                                                                        RBR12420
      RETURN                                                             RBR12430
      END                                                                RBR12440
C                                                                        RBR12450
C -05------------------  END OF SUBROUTINE WPARAM  ------------RBREAK-   RBR12460
C #06#####################  SUBROUTINE REGWAV  ###############RBREAK#   RBR12470
C                                                                        RBR12480
C     This subroutine computes incident wave profile at seaward          RBR12490
C     boundary if IWAVE=1                                                RBR12500
C     Wave profile: Stokes II if UR<26                                   RBR12510
C                   Cnoidal otherwise                                    RBR12520
C                                                                        RBR12530
      SUBROUTINE REGWAV (MODE)                                           RBR12540
C                                                                        RBR12550
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                RBR12560
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                       RBR12570
      DOUBLE PRECISION KCNO,MCNO,KC2,KC,KS,KSREF,KSSEA,KSI               RBR12580
      DIMENSION ETAU(N2)                                                 RBR12590
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                RBR12600
      COMMON /CONSTA/ PI,GRAV                                            RBR12610
      COMMON /CPAR2/  MULTIF,NONEM,NONM,NONE,NEND,NRATE,NHOP             RBR12620
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX      RBR12630
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA           RBR12640
      COMMON /WAVE3/  ETA(N2),ETAIS(N2),ETARS(N2),ETATS(N2)              RBR12650
```

```
      COMMON /WAVE4/  ETAMAX,ETAMIN                                      RBR12660
      COMMON /WAVE5/  KCNO,ECNO,MCNO,KC2                                 RBR12670
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT      RBR12680
      IF (MODE.EQ.0) THEN                                                RBR12690
        CALL CKFPAR (6,2,N2,N2R)                                         RBR12700
        RETURN                                                           RBR12710
      ENDIF                                                              RBR12720
C                                                                        RBR12730
C           CONSTANTS                                                    RBR12740
C           ---------------------------------------------------------    RBR12750
      TWOPI  = 2.D+00*PI                                                 RBR12760
      FOURPI = 4.D+00*PI                                                 RBR12770
      HALFPI = PI/2.D+00                                                 RBR12780
      NONE1  = NONE+1                                                    RBR12790
      NHALF  = NONE/2                                                    RBR12800
      NHALF1 = NHALF+1                                                   RBR12810
C                                                                        RBR12820
C           COMPUTE HALF OF WAVE PROFILE (unadjusted)                    RBR12830
C           ---------------------------------------------------------    RBR12840
C           ETAMAX = normalized maximum surface elevation                RBR12850
C           ETAMIN = normalized minimum surface elevation                RBR12860
C           ETAU = unadjusted surface elevation                          RBR12870
C           N0 = approx. time level at which surface elevation is zero   RBR12880
C           UR based on linear wave theory is used in the following      RBR12890
C              criterion                                                 RBR12900
C           ---------------------------------------------------------    RBR12910
      IF (UR.LT.26.D+00) THEN                                            RBR12920
C           ::    Stokes II Wave Profile                                 RBR12930
        ARG  = TWOPI/WL                                                  RBR12940
        ARG2 = 2.D+00*ARG                                                RBR12950
        DUM  = 16.D+00*DSEAKS*DSINH(ARG)**3.D+00                         RBR12960
        AMP2 = ARG*DCOSH(ARG)*(2.D+00+DCOSH(ARG2))/DUM                   RBR12970
        N0   = 1                                                         RBR12980
        DO 110 N = 1,NHALF1                                              RBR12990
          TIMEN = DBLE(N-1)*T                                            RBR13000
          ETAU(N) = .5D+00*DCOS(TWOPI*TIMEN)+AMP2*DCOS(FOURPI*TIMEN)     RBR13010
          ETAU(N) = KS*ETAU(N)                                           RBR13020
          IF (N.GT.1) THEN                                               RBR13030
            IF(ETAU(N).LE.0.D+00.AND.ETAU(N-1).GT.0.D+00) N0=N           RBR13040
          ENDIF                                                          RBR13050
  110   CONTINUE                                                         RBR13060
        ETAMIN = ETAU(NHALF1)                                            RBR13070
        ETAMAX = ETAU(1)                                                 RBR13080
      ELSE                                                               RBR13090
C           ::    Cnoidal Wave Profile                                   RBR13100
C                 FINDM is to find the parameter M of the Jacobian       RBR13110
C                 elliptic func. See Func. 08 CEL and Subr. 09 SNCNDN    RBR13120
        CALL FINDM (MCNO)                                                RBR13130
        KC2    = 1.D+00-MCNO                                             RBR13140
        KC     = DSQRT(KC2)                                              RBR13150
        KCNO   = CEL(KC,1.D+00,1.D+00,1.D+00)                            RBR13160
        ECNO   = CEL(KC,1.D+00,1.D+00,KC2)                               RBR13170
        UR     = 16.D+00*MCNO*KCNO*KCNO/3.D+00                           RBR13180
        WL     = DSQRT(UR*DSEAKS)                                        RBR13190
        ETAMIN = (1.D+00-ECNO/KCNO)/MCNO - 1.D+00                        RBR13200
```

```
        ETAMIN = KS*ETAMIN                                                  RBR13210
        ETAMAX = ETAMIN + KS                                                RBR13220
        N0     = 1                                                          RBR13230
        DO 120 N = 1,NHALF1                                                 RBR13240
          TIMEN = DBLE(N-1)*T                                               RBR13250
          TETA = 2.D+00*KCNO*TIMEN                                          RBR13260
          CALL SNCNDN (TETA,KC2,SNU,CNU,DNU)                                RBR13270
          ETAU(N) = ETAMIN + KS*CNU*CNU                                     RBR13280
          IF (N.GT.1) THEN                                                  RBR13290
            IF (ETAU(N).LE.0.D+00.AND.ETAU(N-1).GT.0.D+00) N0=N             RBR13300
          ENDIF                                                             RBR13310
  120   CONTINUE                                                            RBR13320
        ETAU(NHALF1) = ETAMIN                                               RBR13330
      ENDIF                                                                 RBR13340
C                                                                           RBR13350
C           THE OTHER HALF OF WAVE PROFILE                                  RBR13360
C           ------------------------------------------------------------    RBR13370
      DO 130 N = NHALF+2,NONE1                                              RBR13380
        ETAU(N) = ETAU(NONE+2-N)                                            RBR13390
  130 CONTINUE                                                              RBR13400
C                                                                           RBR13410
C           ADJUST WAVE PROFILE so that elevation=0 at time=0 and           RBR13420
C           decreases initially with time                                   RBR13430
C           ------------------------------------------------------------    RBR13440
C           ETAU = unadjusted surface elevation                             RBR13450
C           ETA  = adjusted surface elevation                               RBR13460
C           ------------------------------------------------------------    RBR13470
      NMARK = NONE-N0+2                                                     RBR13480
      DO 140 N = 1,NONE1                                                    RBR13490
        IF (N.LE.NMARK) THEN                                                RBR13500
          ETA(N) = ETAU(N+N0-1)                                             RBR13510
        ELSE                                                                RBR13520
          ETA(N) = ETAU(N-NMARK+1)                                          RBR13530
        ENDIF                                                               RBR13540
  140 CONTINUE                                                              RBR13550
C                                                                           RBR13560
      RETURN                                                                RBR13570
      END                                                                   RBR13580
C                                                                           RBR13590
C -06-----------------  END OF SUBROUTINE REGWAV  ------------RBREAK-       RBR13600
C #07#####################  SUBROUTINE FINDM  ################RBREAK#      RBR13610
C                                                                           RBR13620
C     This subroutine computes the parameter m (MLIL<m<MBIG) of the         RBR13630
C     Jacobian elliptic functions (m=MCNO in this program)                  RBR13640
C                                                                           RBR13650
      SUBROUTINE FINDM (MCNO)                                               RBR13660
C                                                                           RBR13670
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                   RBR13680
      DOUBLE PRECISION KCNO,MCNO,KC2,KC,MSAV,MLIL,MBIG                      RBR13690
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                                   RBR13700
      COMMON /CONSTA/ PI,GRAV                                               RBR13710
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA              RBR13720
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT         RBR13730
      SMALL = 1.D-07                                                        RBR13740
      MLIL  = .8D+00                                                        RBR13750
```

```
      INDI  = 0                                                          RBR13760
      I     = 0                                                          RBR13770
      SIGDT = DSQRT(2.D+00*PI*WL0)                                       RBR13780
      MBIG  = 1.00D+00 - 1.00D-15                                        RBR13790
      MCNO  =  .95D+00                                                   RBR13800
  900 CONTINUE                                                           RBR13810
        I    = I+1                                                       RBR13820
        MSAV = MCNO                                                      RBR13830
        KC2  = 1.D+00-MCNO                                               RBR13840
        KC   = DSQRT(KC2)                                                RBR13850
        KCNO = CEL(KC,1.D+00,1.D+00,1.D+00)                              RBR13860
        ECNO = CEL(KC,1.D+00,1.D+00,KC2)                                 RBR13870
        UR   = 16.D+00*MCNO*KCNO*KCNO/3.D+00                             RBR13880
        WL   = DSQRT(UR*DSEAKS)                                          RBR13890
        FCNO = 1.D+00 + (-MCNO+2.D+00-3.D+00*ECNO/KCNO)/(MCNO*DSEAKS)    RBR13900
        FCNO = SIGDT*DSQRT(FCNO)/WL - 1.D+00                             RBR13910
        IF (FCNO.LT.0.D+00) THEN                                         RBR13920
          MBIG = MCNO                                                    RBR13930
        ELSEIF (FCNO.GT.0.D+00) THEN                                     RBR13940
          MLIL = MCNO                                                    RBR13950
        ELSE                                                             RBR13960
          RETURN                                                         RBR13970
        ENDIF                                                            RBR13980
        MCNO = (MLIL+MBIG)/2.D+00                                        RBR13990
        DIF  = DABS(MSAV-MCNO)                                           RBR14000
        IF (DIF.LT.SMALL) RETURN                                         RBR14010
        IF (INDI.EQ.0) THEN                                              RBR14020
          IF (I.EQ.50) THEN                                              RBR14030
            SMALL = 1.D-13                                               RBR14040
            INDI  = 1                                                    RBR14050
          ELSE                                                           RBR14060
            IF (MCNO.GT..9999D+00) THEN                                  RBR14070
              SMALL = 1.D-13                                             RBR14080
              INDI  = 1                                                  RBR14090
            ENDIF                                                        RBR14100
          ENDIF                                                          RBR14110
        ENDIF                                                            RBR14120
      IF (I.LT.100) GOTO 900                                             RBR14130
      WRITE  (*,9910)                                                    RBR14140
      WRITE (99,9910)                                                    RBR14150
 9910 FORMAT (/' From Subr. 07 FINDM:'/                                  RBR14160
     +        ' Criterion for parameter m=MCNO not satisfied')           RBR14170
C                                                                        RBR14180
      RETURN                                                             RBR14190
      END                                                                RBR14200
C                                                                        RBR14210
C -07------------------  END OF SUBROUTINE FINDM  -----------RBREAK-     RBR14220
C #08#################  DOUBLE PRECISION FUNCTION CEL  #########RBREAK#  RBR14230
C                                                                        RBR14240
C     This function computes the general complete elliptic integral,     RBR14250
C     and is a double precision version of the "Function CEL" from       RBR14260
C     the book:                                                          RBR14270
C       William H. Press et al.                                          RBR14280
C       Numerical Recipes: The Art of Scientific Computing.              RBR14290
C       Cambridge University Press, New York, 1986.                      RBR14300
```

```
C       Pages 187-188.                                                  RBR14310
C                                                                       RBR14320
      DOUBLE PRECISION FUNCTION CEL (QQC,PP,AA,BB)                      RBR14330
C                                                                       RBR14340
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR14350
      PARAMETER (CA=1.D-06,PIO2=1.5707963268D+00)                       RBR14360
      IF (QQC.EQ.0.D+00) THEN                                           RBR14370
        WRITE  (*,*) 'Failure in Function CEL'                          RBR14380
        WRITE (99,*) 'Failure in Function CEL'                          RBR14390
        STOP                                                            RBR14400
      ENDIF                                                             RBR14410
      QC = DABS(QQC)                                                    RBR14420
      A  = AA                                                           RBR14430
      B  = BB                                                           RBR14440
      P  = PP                                                           RBR14450
      E  = QC                                                           RBR14460
      EM = 1.D+00                                                       RBR14470
      IF (P.GT.0.D+00) THEN                                             RBR14480
        P = DSQRT(P)                                                    RBR14490
        B = B/P                                                         RBR14500
      ELSE                                                              RBR14510
        F = QC*QC                                                       RBR14520
        Q = 1.D+00-F                                                    RBR14530
        G = 1.D+00-P                                                    RBR14540
        F = F-P                                                         RBR14550
        Q = Q*(B-A*P)                                                   RBR14560
        P = DSQRT(F/G)                                                  RBR14570
        A = (A-B)/G                                                     RBR14580
        B = -Q/(G*G*P)+A*P                                              RBR14590
      ENDIF                                                             RBR14600
  900 F  = A                                                            RBR14610
      A  = A+B/P                                                        RBR14620
      G  = E/P                                                          RBR14630
      B  = B+F*G                                                        RBR14640
      B  = B+B                                                          RBR14650
      P  = G+P                                                          RBR14660
      G  = EM                                                           RBR14670
      EM = QC+EM                                                        RBR14680
      IF (DABS(G-QC).GT.G*CA) THEN                                      RBR14690
        QC = DSQRT(E)                                                   RBR14700
        QC = QC+QC                                                      RBR14710
        E  = QC*EM                                                      RBR14720
        GOTO 900                                                        RBR14730
      ENDIF                                                             RBR14740
      CEL = PIO2*(B+A*EM)/(EM*(EM+P))                                   RBR14750
C                                                                       RBR14760
      RETURN                                                            RBR14770
      END                                                               RBR14780
C                                                                       RBR14790
C -08------------  END OF DOUBLE PRECISION FUNCTION CEL  ------RBREAK-  RBR14800
C #09#####################  SUBROUTINE SNCNDN  ###############RBREAK#  RBR14810
C                                                                       RBR14820
C     This subroutine computes the Jacobian elliptic functions,         RBR14830
C     and is a double precision version of the "Subroutine SNCNDN"      RBR14840
C     from the book:                                                    RBR14850
```

```
C       William H. Press et al.                                          RBR14860
C       Numerical Recipes: The Art of Scientific Computing.              RBR14870
C       Cambridge University Press, New York, 1986.                      RBR14880
C       Page 189.                                                        RBR14890
C                                                                        RBR14900
      SUBROUTINE SNCNDN (UU,EMMC,SN,CN,DN)                               RBR14910
C                                                                        RBR14920
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                RBR14930
      PARAMETER (CA=1.D-06)                                              RBR14940
      DIMENSION EM(13),EN(13)                                            RBR14950
      LOGICAL BO                                                         RBR14960
      D   = 0.D+00                                                       RBR14970
      EMC = EMMC                                                         RBR14980
      U   = UU                                                           RBR14990
      IF (EMC.NE.0.D+00) THEN                                            RBR15000
        BO = (EMC.LT.0.D+00)                                             RBR15010
        IF (BO) THEN                                                     RBR15020
          D   = 1.D+00-EMC                                               RBR15030
          EMC = -EMC/D                                                   RBR15040
          D   = DSQRT(D)                                                 RBR15050
          U   = D*U                                                      RBR15060
        ENDIF                                                            RBR15070
        A  = 1.D+00                                                      RBR15080
        DN = 1.D+00                                                      RBR15090
        DO 110 I = 1,13                                                  RBR15100
          L = I                                                          RBR15110
          EM(I) = A                                                      RBR15120
          EMC   = DSQRT(EMC)                                             RBR15130
          EN(I) = EMC                                                    RBR15140
          C     = .5D+00*(A+EMC)                                         RBR15150
          IF (DABS(A-EMC).LE.CA*A) GOTO 910                              RBR15160
          EMC = A*EMC                                                    RBR15170
          A = C                                                          RBR15180
  110   CONTINUE                                                         RBR15190
  910   U  = C*U                                                         RBR15200
        SN = DSIN(U)                                                     RBR15210
        CN = DCOS(U)                                                     RBR15220
        IF (SN.EQ.0.D+00) GOTO 920                                       RBR15230
        A = CN/SN                                                        RBR15240
        C = A*C                                                          RBR15250
        DO 120 II = L,1,-1                                               RBR15260
          B  = EM(II)                                                    RBR15270
          A  = C*A                                                       RBR15280
          C  = DN*C                                                      RBR15290
          DN = (EN(II)+A)/(B+A)                                          RBR15300
          A  = C/B                                                       RBR15310
  120   CONTINUE                                                         RBR15320
        A = 1.D+00/DSQRT(C*C+1.D+00)                                     RBR15330
        IF (SN.LT.0.D+00) THEN                                           RBR15340
          SN = -A                                                        RBR15350
        ELSE                                                             RBR15360
          SN = A                                                         RBR15370
        ENDIF                                                            RBR15380
        CN = C*SN                                                        RBR15390
  920   IF (BO) THEN                                                     RBR15400
```

```
          IF (ABS(D).LT.1.D-10) CALL STOPP (9,9)                        RBR15410
          A  = DN                                                       RBR15420
          DN = CN                                                       RBR15430
          CN = A                                                        RBR15440
          SN = SN/D                                                     RBR15450
        ENDIF                                                           RBR15460
      ELSE                                                              RBR15470
        CN = 1.D+00/DCOSH(U)                                            RBR15480
        DN = CN                                                         RBR15490
        SN = DTANH(U)                                                   RBR15500
      ENDIF                                                             RBR15510
C                                                                       RBR15520
      RETURN                                                            RBR15530
      END                                                               RBR15540
C                                                                       RBR15550
C -09------------------  END OF SUBROUTINE SNCNDN  ------------RBREAK-  RBR15560
C #10######################  SUBROUTINE CPARAM  ###############RBREAK#  RBR15570
C                                                                       RBR15580
C     This subroutine                                                   RBR15590
C     . checks case-signature NONM at the beginning of RBREAK           RBR15600
C       computation                                                     RBR15610
C     . calculates computation parameters which are dependent           RBR15620
C       on NONE for each computation unit                               RBR15630
C                                                                       RBR15640
      SUBROUTINE CPARAM (MODE)                                          RBR15650
C                                                                       RBR15660
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR15670
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,        RBR15680
     +                ISAVA,ISAVB,ISAVC                                 RBR15690
      COMMON /CPAR2/  MULTIF,NONEM,NONM,NONE,NEND,NRATE,NHOP            RBR15700
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR15710
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     RBR15720
C                                                                       RBR15730
C           CHECK THE CASE-SIGNATURE NONM                               RBR15740
C           ------------------------------------------------------      RBR15750
C         . NONEM = minimum allowable value of NONE                     RBR15760
C         . NONE  = number of time steps in one wave period             RBR15770
C         . NONE must:                                                  RBR15780
C           + satisfy a numerical stability criterion                   RBR15790
C           + be divisible by NRATE for storing purposes                RBR15800
C           + be an even number, for regular waves (IWAVE=1) only       RBR15810
C         . Below, NDUM3 is the smallest value which satisfies the      RBR15820
C           above criteria. NONEM, the minimum allowable value of NONE, RBR15830
C           is defined as NONEM=MULTIF*NDUM3, where MULTIF is an        RBR15840
C           integer, greater than or equal to unity, specified by user. RBR15850
C           ------------------------------------------------------      RBR15860
      IF (MODE.EQ.0) THEN                                               RBR15870
        EPSI  = DMAX1(X1,X2)                                            RBR15880
        DUM1  = 1.D+00 + EPSI*EPSI/4.D+00                               RBR15890
        DUM2  = DSQRT(DUM1) - EPSI/2.D+00                               RBR15900
        DUM3  = (1.D+00+DSQRT(DSEA))/(DUM2*X)                           RBR15910
        NDUM1 = INT(DUM3) + 1                                           RBR15920
        NDUM2 = NDUM1/NRATE + 1                                         RBR15930
        NDUM3 = NDUM2*NRATE                                             RBR15940
        IF (IWAVE.EQ.1) THEN                                            RBR15950
```

```
          IDUM = MOD(NDUM3,2)                                              RBR15960
          IF (IDUM.NE.0) NDUM3=2*NDUM3                                     RBR15970
        ENDIF                                                              RBR15980
C       :: Minimum allowable value of NONE is NONEM                        RBR15990
        NONEM = MULTIF*NDUM3                                               RBR16000
        IF (NONM.NE.NONEM) CALL STOPP (14,14)                              RBR16010
C       :: Initial value NONE=NONEM serves a practical purpose             RBR16020
        NONE = NONEM                                                       RBR16030
      ENDIF                                                                RBR16040
C                                                                          RBR16050
C           CALCULATE COMPUTATION PARAMETERS                               RBR16060
C           which are dependent on NONE                                    RBR16070
C           ------------------------------------------------------         RBR16080
C           T = constant time step                                         RBR16090
C           X = constant grid spacing between two adjacent nodes           RBR16100
C           NRATE = rate (data points per per wave period) of storing      RBR16110
C                   output time series                                     RBR16120
C           ------------------------------------------------------         RBR16130
      T    = 1.D+00/DBLE(NONE)                                             RBR16140
      TX   = T/X                                                           RBR16150
      XT   = X/T                                                           RBR16160
      TTX  = T*T/X                                                         RBR16170
      TTXX = T*T/(X*X)                                                     RBR16180
      TWOX = 2.D+00*X                                                      RBR16190
      NHOP = NONE/NRATE                                                    RBR16200
C                                                                          RBR16210
      RETURN                                                               RBR16220
      END                                                                  RBR16230
C                                                                          RBR16240
C -10-----------------  END OF SUBROUTINE CPARAM  ------------RBREAK-      RBR16250
C #11#####################  SUBROUTINE NUMSTA  ###############RBREAK#     RBR16260
C                                                                          RBR16270
C     This subroutine checks if numerical stability criterion is           RBR16280
C     violated (IPROB=3)                                                   RBR16290
C                                                                          RBR16300
      SUBROUTINE NUMSTA (MODE,M)                                           RBR16310
C                                                                          RBR16320
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                  RBR16330
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                         RBR16340
      INTEGER S,SM1,SP1                                                    RBR16350
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                  RBR16360
      COMMON /CPAR2/  MULTIF,NONEM,NONM,NONE,NEND,NRATE,NHOP               RBR16370
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX        RBR16380
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)                RBR16390
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                                 RBR16400
      IF (MODE.EQ.0) THEN                                                  RBR16410
        CALL CKFPAR (11,1,N1,N1R)                                          RBR16420
        RETURN                                                             RBR16430
      ENDIF                                                                RBR16440
C           ------------------------------------------------------         RBR16450
C           T = time step; X = spatial grid size; XT = X/T                 RBR16460
C           ------------------------------------------------------         RBR16470
      ICOUNT = 0                                                           RBR16480
      DO 100 J = 1,S                                                       RBR16490
        IF (DABS(V(J)).GT.XT) ICOUNT=ICOUNT+1                              RBR16500
```

```
  100 CONTINUE                                                          RBR16510
      IF (ICOUNT.GT.0) THEN                                             RBR16520
        IPROB=3                                                         RBR16530
        CALL NSI (OMEGA,NONE)                                           RBR16540
        WRITE  (*,9910) M,S,NONE,OMEGA,TIME,ICOUNT                      RBR16550
        WRITE (99,9910) M,S,NONE,OMEGA,TIME,ICOUNT                      RBR16560
        CALL STOPP (10,11)                                              RBR16570
      ENDIF                                                             RBR16580
 9910 FORMAT (/' From Subr. 11 NUMSTA: IPROB=3'/                        RBR16590
     +        ' (Stability criterion violated)'/                        RBR16600
     +        ' Computation unit              M =',I8/                  RBR16610
     +        ' Waterline node                S =',I8/                  RBR16620
     +        ' Computation parameters:    NONE =',I8/                  RBR16630
     +        '                           OMEGA =',F11.2/               RBR16640
     +        ' Time of occurrence         TIME =',F18.9/               RBR16650
     +        ' Number of occurrence     ICOUNT =',I8)                  RBR16660
C                                                                       RBR16670
      RETURN                                                            RBR16680
      END                                                               RBR16690
C                                                                       RBR16700
C -11-----------------  END OF SUBROUTINE NUMSTA  ------------RBREAK-   RBR16710
C #12#####################  SUBROUTINE EXTRAP  ###############RBREAK#   RBR16720
C                                                                       RBR16730
C     This subroutine estimates U(2,SP1) and V(SP1) with SP1=(S+1)      RBR16740
C     by extrapolation                                                  RBR16750
C                                                                       RBR16760
C     S = most landward node at present time level (N-1)                RBR16770
C     The following values at node j are known at time level (N-1)      RBR16780
C     . U(1,j) = volume flux                                            RBR16790
C     . U(2,j) = total water depth                                      RBR16800
C     . V(j)   = depth-averaged velocity                                RBR16810
C                                                                       RBR16820
      SUBROUTINE EXTRAP (MODE)                                          RBR16830
C                                                                       RBR16840
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR16850
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR16860
      INTEGER S,SM1,SP1                                                 RBR16870
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR16880
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR16890
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR16900
      IF (MODE.EQ.0) THEN                                               RBR16910
        CALL CKFPAR (12,1,N1,N1R)                                       RBR16920
        RETURN                                                          RBR16930
      ENDIF                                                             RBR16940
      SM1 = S-1                                                         RBR16950
      SP1 = S+1                                                         RBR16960
      V(SP1)    = 2.D+00*V(S) - V(SM1)                                  RBR16970
      U(2,SP1) = 2.D+00*U(2,S) - U(2,SM1)                               RBR16980
      U(1,SP1) = U(2,SP1)*V(SP1)                                        RBR16990
      IF (U(2,SP1).GT.0.D+00) THEN                                      RBR17000
        C(SP1) = DSQRT(U(2,SP1))                                        RBR17010
      ELSE                                                              RBR17020
        C(SP1) = 0.D+00                                                 RBR17030
      ENDIF                                                             RBR17040
C                                                                       RBR17050
```

```
      RETURN                                                            RBR17060
      END                                                               RBR17070
C                                                                       RBR17080
C -12------------------  END OF SUBROUTINE EXTRAP  ------------RBREAK-  RBR17090
C #13#####################  SUBROUTINE RETAIN  ###############RBREAK#  RBR17100
C                                                                       RBR17110
C     This subroutine retains some values at present time level at      RBR17120
C     landward and seaward boundaries                                   RBR17130
C                                                                       RBR17140
      SUBROUTINE RETAIN (MODE)                                          RBR17150
C                                                                       RBR17160
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR17170
      INTEGER S,SM1,SP1                                                 RBR17180
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR17190
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR17200
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR17210
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR17220
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                     RBR17230
      IF (MODE.EQ.0) THEN                                               RBR17240
        CALL CKFPAR (13,1,N1,N1R)                                       RBR17250
        RETURN                                                          RBR17260
      ENDIF                                                             RBR17270
      VSN    = V(S)                                                     RBR17280
      USN(1) = U(1,S)                                                   RBR17290
      USN(2) = U(2,S)                                                   RBR17300
      VMN    = V(SM1)                                                   RBR17310
      UMN(1) = U(1,SM1)                                                 RBR17320
      V1N    = V(1)                                                     RBR17330
      V2N    = V(2)                                                     RBR17340
C                                                                       RBR17350
      RETURN                                                            RBR17360
      END                                                               RBR17370
C                                                                       RBR17380
C -13------------------  END OF SUBROUTINE RETAIN  ------------RBREAK-  RBR17390
C #14#####################  SUBROUTINE CRITV  ###############RBREAK#  RBR17400
C                                                                       RBR17410
C     This subroutine calculates critical velocities used in            RBR17420
C     characteristic variables and checks for negative water depth      RBR17430
C     (IPROB=1)                                                         RBR17440
C                                                                       RBR17450
      SUBROUTINE CRITV (MODE,M)                                         RBR17460
C                                                                       RBR17470
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR17480
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR17490
      INTEGER S,SM1,SP1                                                 RBR17500
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR17510
      COMMON /CPAR2/  MULTIF,NONEM,NONM,NONE,NEND,NRATE,NHOP            RBR17520
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR17530
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR17540
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR17550
      IF (MODE.EQ.0) THEN                                               RBR17560
        CALL CKFPAR (14,1,N1,N1R)                                       RBR17570
        RETURN                                                          RBR17580
      ENDIF                                                             RBR17590
      ICOUNT = 0                                                        RBR17600
```

```
      DO 100 J = 1,S                                                    RBR17610
        IF (U(2,J).LT.0.D+00) THEN                                      RBR17620
          ICOUNT = ICOUNT + 1                                           RBR17630
        ELSE                                                            RBR17640
          C(J) = DSQRT(U(2,J))                                          RBR17650
        ENDIF                                                           RBR17660
  100 CONTINUE                                                          RBR17670
      IF (ICOUNT.GT.0) THEN                                             RBR17680
        IPROB=1                                                         RBR17690
        CALL NSI (OMEGA,NONE)                                           RBR17700
        WRITE  (*,9910) M,S,NONE,OMEGA,TIME,ICOUNT                      RBR17710
        WRITE (99,9910) M,S,NONE,OMEGA,TIME,ICOUNT                      RBR17720
        CALL STOPP (10,11)                                              RBR17730
      ENDIF                                                             RBR17740
 9910 FORMAT (/' From Subr. 14 CRITV: IPROB=1'/                         RBR17750
     +        ' (Negative water depth)'/                                RBR17760
     +        ' Computation unit              M =',I8/                  RBR17770
     +        ' Waterline node                S =',I8/                  RBR17780
     +        ' Computation parameters:    NONE =',I8/                  RBR17790
     +        '                           OMEGA =',F11.2/              RBR17800
     +        ' Time of occurrence         TIME =',F18.9/              RBR17810
     +        ' Number of occurrence     ICOUNT =',I8)                 RBR17820
C                                                                       RBR17830
      RETURN                                                            RBR17840
      END                                                               RBR17850
C                                                                       RBR17860
C -14-------------------  END OF SUBROUTINE CRITV  ------------RBREAK-  RBR17870
C #15#####################  SUBROUTINE MARCH  ################RBREAK#  RBR17880
C                                                                       RBR17890
C     This subroutine marches the computation from present time level   RBR17900
C     (N-1) to next time level N excluding seaward and landward         RBR17910
C     boundaries, which are treated separately                          RBR17920
C                                                                       RBR17930
      SUBROUTINE MARCH (MODE,M)                                         RBR17940
C                                                                       RBR17950
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR17960
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR17970
      INTEGER S,SM1,SP1                                                 RBR17980
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR17990
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,        RBR18000
     +                ISAVA,ISAVB,ISAVC                                 RBR18010
      COMMON /CPAR2/  MULTIF,NONEM,NONM,NONE,NEND,NRATE,NHOP            RBR18020
      COMMON /CPAR3/  INITS,JE,JE1                                      RBR18030
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR18040
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     RBR18050
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR18060
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR18070
      IF (MODE.EQ.0) THEN                                               RBR18080
        CALL CKFPAR (15,1,N1,N1R)                                       RBR18090
        RETURN                                                          RBR18100
      ENDIF                                                             RBR18110
C                                                                       RBR18120
C          ------------------------------------------------------       RBR18130
C          U(1,j) and U(2,j) at time level N are computed as follows:   RBR18140
C          . at j=2,3,...,JDAM: WITH numerical damping term             RBR18150
```

```
C           . at j=(JDAM+1),(JDAM+2),...,JLAX: NO numer. damping term   RBR18160
C           JE  = landward edge node                                    RBR18170
C           JE1 = JE-1                                                  RBR18180
C           ------------------------------------------------------      RBR18190
      IF (IJOB.LT.3) THEN                                               RBR18200
        JDAM = S-2                                                      RBR18210
        JLAX = S                                                        RBR18220
        IF (IJOB.EQ.2.AND.S.EQ.JE) JLAX=JE1                             RBR18230
      ELSE                                                              RBR18240
        JDAM = JE1                                                      RBR18250
        JLAX = JE1                                                      RBR18260
      ENDIF                                                             RBR18270
      JLAX1 = JLAX+1                                                    RBR18280
C                                                                       RBR18290
C           COMPUTE ELEMENTS OF MATRICES                                RBR18300
C           ------------------------------------------------------      RBR18310
C         . Subr. 20 MATAFG computes non-constant elements of Matrices  RBR18320
C           A and G, and the elements of Matrix F                       RBR18330
C         . Subr. 22 MATS computes the first element of Matrix S        RBR18340
C         . Subr. 21 MATGJR, 23 MATD, and 24 MATU compute the elements  RBR18350
C           of Matrices g, D, and U, respectively                       RBR18360
C         . Subr. 24 MATU computes values of U at time level N using    RBR18370
C           the results obtained from the other four subroutines        RBR18380
C           ------------------------------------------------------      RBR18390
      CALL MATAFG (MODE,1,JLAX1)                                        RBR18400
      CALL MATGJR (MODE,1,JLAX)                                         RBR18410
      CALL MATS   (MODE,2,JLAX)                                         RBR18420
      CALL MATD   (MODE,JDAM,JLAX)                                      RBR18430
      CALL MATU   (MODE,2,JLAX)                                         RBR18440
C                                                                       RBR18450
C           EMERGENCY STOP                                              RBR18460
C           IPROB=2: water depth at (S-1) <or= DELTA                    RBR18470
C           ------------------------------------------------------      RBR18480
      IF (U(2,SM1).LE.DELTA) THEN                                       RBR18490
        IPROB=2                                                         RBR18500
        CALL NSI (OMEGA,NONE)                                           RBR18510
        WRITE  (*,9910) M,S,NONE,OMEGA,DELTA,U(2,SM1),TIME              RBR18520
        WRITE (99,9910) M,S,NONE,OMEGA,DELTA,U(2,SM1),TIME              RBR18530
        CALL STOPP (10,11)                                              RBR18540
      ENDIF                                                             RBR18550
 9910 FORMAT (/' From Subr. 15 MARCH: IPROB=2'/                         RBR18560
     +        ' (Water depth at (S-1) <or= DELTA)'/                     RBR18570
     +        ' Computation unit              M =',I8/                  RBR18580
     +        ' Waterline node                S =',I8/                  RBR18590
     +        ' Computation parameters:    NONE =',I8/                  RBR18600
     +        '                           OMEGA =',F11.2/               RBR18610
     +        '                           DELTA =',F18.9/               RBR18620
     +        ' Water depth at (S-1)  U(2,S-1) =',F18.9/                RBR18630
     +        ' Time of occurrence         TIME =',F18.9)               RBR18640
C                                                                       RBR18650
C           COMPLETE THE COMPUTATION OF HYDRODYNAMIC QUANTITIES         RBR18660
C           ------------------------------------------------------      RBR18670
C           Water depth h is taken to be not less than DELTA for        RBR18680
C           submerged structures                                        RBR18690
C           ------------------------------------------------------      RBR18700
```

```
      DO 100 J = 2,JLAX                                                 RBR18710
        IF (IJOB.EQ.3.AND.U(2,J).LT.DELTA) U(2,J)=DELTA                 RBR18720
        V(J)     = U(1,J)/U(2,J)                                        RBR18730
        ELEV(J) = U(2,J)-U2INIT(J)                                      RBR18740
  100 CONTINUE                                                          RBR18750
C                                                                       RBR18760
      RETURN                                                            RBR18770
      END                                                               RBR18780
C                                                                       RBR18790
C -15------------------  END OF SUBROUTINE MARCH  ------------RBREAK-   RBR18800
C #16#####################  SUBROUTINE LANDBC  ###############RBREAK#   RBR18810
C                                                                       RBR18820
C     This subroutine manages the computation for                       RBR18830
C     landward boundary conditions                                      RBR18840
C                                                                       RBR18850
      SUBROUTINE LANDBC (MODE,M,N,ETAT)                                 RBR18860
C                                                                       RBR18870
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR18880
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR18890
      INTEGER S,SM1,SP1                                                 RBR18900
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR18910
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,        RBR18920
     +                ISAVA,ISAVB,ISAVC                                 RBR18930
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES            RBR18940
      COMMON /CPAR3/  INITS,JE,JE1                                      RBR18950
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR18960
      COMMON /WAVE3/  ETA(N2),ETAIS(N2),ETARS(N2),ETATS(N2)             RBR18970
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     RBR18980
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     RBR18990
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR19000
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR19010
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3),RUNUPS(N3),RSTAT(3,N3),      RBR19020
     +                RMEAN(N3)                                         RBR19030
      COMMON /STT1/   ELSTAT(3,3),U1STAT(N1),ESTAT(3,N1),VSTAT(3,N1),   RBR19040
     +                ELMEAN(3),U1MEAN(N1),EMEAN(N1),VMEAN(N1)          RBR19050
      COMMON /OVER/   OV(4),OVMEAN                                      RBR19060
      COMMON /ARMOR3/ JSTAB,JSTABM                                      RBR19070
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                     RBR19080
      IF (MODE.EQ.0) THEN                                               RBR19090
        CALL CKFPAR (16,1,N1,N1R)                                       RBR19100
        CALL CKFPAR (16,2,N2,N2R)                                       RBR19110
        CALL CKFPAR (16,3,N3,N3R)                                       RBR19120
        RETURN                                                          RBR19130
      ENDIF                                                             RBR19140
C                                                                       RBR19150
C           MANAGE LANDWARD B.C.                                        RBR19160
C           ------------------------------------------------------      RBR19170
C         . Subr. 17 RUNUP computes shoreline movement if               RBR19180
C           computational waterline is on the slope (IJOB<3)            RBR19190
C         . Subr. 18 OVERT computes hydrodynamic quantities at          RBR19200
C           landward edge node if overtopping occurs (IJOB=2)           RBR19210
C         . ALPHA = landward-advancing characteristics                  RBR19220
C         . ETAT  = surface elevation associated with transmitted wave  RBR19230
C           at landward boundary (IJOB=3)                               RBR19240
C         . DLAND = norm. water depth below SWL at landward boundary    RBR19250
```

```
C           (IJOB=3)                                                    RBR19260
C         . S used in Subr. 17 RUNUP is of present time level           RBR19270
C           ---------------------------------------------------------   RBR19280
      IF (IJOB.EQ.1) THEN                                               RBR19290
        CALL RUNUP (MODE)                                               RBR19300
        IF (S.GT.JE1) THEN                                              RBR19310
          WRITE  (*,9910) TIME,S,JE                                     RBR19320
          WRITE (99,9910) TIME,S,JE                                     RBR19330
          STOP                                                          RBR19340
        ENDIF                                                           RBR19350
      ELSEIF (IJOB.EQ.2) THEN                                           RBR19360
        IF (S.LT.JE) THEN                                               RBR19370
          CALL RUNUP (MODE)                                             RBR19380
        ELSE                                                            RBR19390
          CALL OVERT (MODE)                                             RBR19400
        ENDIF                                                           RBR19410
      ELSE                                                              RBR19420
        DUM     = TX*(VSN+C(S))*(VSN-VMN+2.D+00*(C(S)-C(SM1)))          RBR19430
        ALPHA   = VSN+2.D+00*C(S) - DUM - T*THETA(S)                    RBR19440
        ETAT    = ALPHA*DLAND2/2.D+00 - DLAND                           RBR19450
        U(2,S)  = DLAND + ETAT                                          RBR19460
        V(S)    = ALPHA - 2.D+00*DSQRT(U(2,S))                          RBR19470
        U(1,S)  = U(2,S)*V(S)                                           RBR19480
        ELEV(S) = U(2,S) - U2INIT(S)                                    RBR19490
      ENDIF                                                             RBR19500
 9910 FORMAT (/' From Subr. 16 LANDBC:'/                                RBR19510
     +   ' t =',F18.6,'; S =',I8,'; End Node =',I8/                     RBR19520
     +   ' Slope is not long enough to accomodate shoreline movement'/  RBR19530
     +   ' Specify longer slope or choose overtopping computation')     RBR19540
C                                                                       RBR19550
C           CONDITIONS LANDWARD OF NEW WATERLINE NODE S                 RBR19560
C           ---------------------------------------------------------   RBR19570
      IF (IJOB.LT.3) THEN                                               RBR19580
        DO 100 L = S+1,JE                                               RBR19590
          U(1,L)  = 0.D+00                                              RBR19600
          U(2,L)  = 0.D+00                                              RBR19610
          V(L)    = 0.D+00                                              RBR19620
          ELEV(L) = ZB(L)                                               RBR19630
  100   CONTINUE                                                        RBR19640
      ENDIF                                                             RBR19650
C                                                                       RBR19660
C           RUNUPS ASSOCIATED WITH DEPTHS (DELTAR(L),L=1,NDELR)         RBR19670
C           (Assume water depth decreases landward and U(2,S+1)=0.)     RBR19680
C           If IJOB<3, NDELR>0, DO 200 performed                        RBR19690
C           If IJOB=3, NDELR=0, DO 200 void                             RBR19700
C           ---------------------------------------------------------   RBR19710
C         . MSTAB: Computation of armor stability or movement is        RBR19720
C           performed excluding the first MSTAB computation units       RBR19730
C         . JSTAB = the largest node number based on DELTAR(1) for      RBR19740
C           armor stability or movement.                                RBR19750
C           Notes: . If the computed armor stability or movement at     RBR19760
C                    the shoreline is not realistic, it is recommended  RBR19770
C                    to increase the value of DELTAR(1)                 RBR19780
C                  . For IJOB=3, JSTAB=JE, set in Subr. 26 INIT         RBR19790
C         . DELTAR = water depth associated with visual or measured     RBR19800
```

```
C           waterline                                                    RBR19810
C         . RUNUPS = free surface elevation where the water depth        RBR19820
C           equals DELTAR                                                RBR19830
C         . NDELR = number of DELTARs                                    RBR19840
C           ------------------------------------------------------       RBR19850
      DO 200 L = 1,NDELR                                                 RBR19860
        IF (IJOB.EQ.2.AND.S.EQ.JE.AND.U(2,S).GE.DELTAR(L)) THEN          RBR19870
          IF (L.EQ.1.AND.M.GT.MSTAB) JSTAB=S                             RBR19880
          RUNUPS(L) = ZB(S) + U(2,S)                                     RBR19890
        ELSE                                                             RBR19900
          INDIC = 0                                                      RBR19910
          J = -1                                                         RBR19920
  900     CONTINUE                                                       RBR19930
            J = J + 1                                                    RBR19940
            IF (U(2,S-J).GE.DELTAR(L)) THEN                              RBR19950
              INDIC = 1                                                  RBR19960
              NRUN1 = S-J                                                RBR19970
              NRUN2 = S-J+1                                              RBR19980
              IF (L.EQ.1.AND.M.GT.MSTAB) JSTAB=NRUN1                     RBR19990
              DEL1  = U(2,S-J)                                           RBR20000
              DEL2  = U(2,S-J+1)                                         RBR20010
              RUN   = (ZB(NRUN2)-ZB(NRUN1))*(DEL1-DELTAR(L))             RBR20020
              RUN   = RUN/(DEL1-DEL2)                                    RBR20030
              RUN   = RUN + ZB(NRUN1)                                    RBR20040
              RUNUPS(L) = RUN + DELTAR(L)                                RBR20050
            ENDIF                                                        RBR20060
          IF (INDIC.EQ.0) GOTO 900                                       RBR20070
        ENDIF                                                            RBR20080
  200 CONTINUE                                                           RBR20090
C                                                                        RBR20100
C           STATISTICAL CALCULATIONS                                     RBR20110
C           Performed excluding the first MSTAT computation units        RBR20120
C           ------------------------------------------------------       RBR20130
C         . IJOB<3:                                                      RBR20140
C           RSTAT(1,L),RSTAT(2,L),RSTAT(3,L) = mean,maximum,minimum of   RBR20150
C           RUNUPS(L)                                                    RBR20160
C         . IJOB=2:                                                      RBR20170
C           OV(1) = normalized average overtopping rate                  RBR20180
C           OV(2) = normalized maximum overtopping rate                  RBR20190
C           OV(3) = normalized overtopping duration                      RBR20200
C           OV(4) = normalized time when OV(2) occurs                    RBR20210
C         . IJOB=3:                                                      RBR20220
C           For regular wave, during the last computation unit ETAT      RBR20230
C           is saved in time-array ETATS (ETAT = surface elevation       RBR20240
C           associated with transmitted wave at landward boundary)       RBR20250
C           ------------------------------------------------------       RBR20260
      IF (M.GT.MSTAT) THEN                                               RBR20270
        IF (IJOB.LT.3) THEN                                              RBR20280
          CALL STAT1 (2,RSTAT,RUNUPS,3,NDELR)                            RBR20290
          IF (IJOB.EQ.2) THEN                                            RBR20300
            OV(1) = OV(1) + U(1,JE)                                      RBR20310
            IF (U(1,JE).GT.0.D+00) OV(3)=OV(3)+T                         RBR20320
            IF (U(1,JE).GT.OV(2)) THEN                                   RBR20330
              OV(4) = TIME                                               RBR20340
              OV(2) = U(1,JE)                                            RBR20350
```

```
          ENDIF                                                          RBR20360
        ENDIF                                                            RBR20370
      ELSE                                                               RBR20380
        IF (IWAVE.EQ.1) THEN                                             RBR20390
          ETATS(N) = ETAT                                                RBR20400
        ELSE                                                             RBR20410
          CALL STAT2 (ETAT,ELSTAT(1,3),ELSTAT(2,3),ELSTAT(3,3))          RBR20420
        ENDIF                                                            RBR20430
      ENDIF                                                              RBR20440
    ENDIF                                                                RBR20450
C                                                                        RBR20460
    RETURN                                                               RBR20470
    END                                                                  RBR20480
C                                                                        RBR20490
C -16------------------  END OF SUBROUTINE LANDBC  ------------RBREAK-   RBR20500
C #17#####################  SUBROUTINE RUNUP  ################RBREAK#   RBR20510
C                                                                        RBR20520
C     This subroutine computes waterline movement for IJOB=1 and if      RBR20530
C     no overtopping occurs for IJOB=2                                   RBR20540
C                                                                        RBR20550
      SUBROUTINE RUNUP (MODE)                                            RBR20560
C                                                                        RBR20570
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                RBR20580
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                       RBR20590
      DIMENSION USN2(2),US1N1(2)                                         RBR20600
      INTEGER S,SM1,SP1,SNEW                                             RBR20610
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                RBR20620
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX      RBR20630
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)      RBR20640
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)              RBR20650
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)   RBR20660
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                               RBR20670
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                      RBR20680
      IF (MODE.EQ.0) THEN                                                RBR20690
        CALL CKFPAR (17,1,N1,N1R)                                        RBR20700
        RETURN                                                           RBR20710
      ENDIF                                                              RBR20720
C                                                                        RBR20730
C             ADJUST VALUES AT S IF U(2,S)>U(2,S-1)                      RBR20740
C             ------------------------------------------------------     RBR20750
      IF (U(2,S).GE.U(2,SM1)) THEN                                       RBR20760
        V(S)    = 2.D+00*V(SM1) - V(S-2)                                 RBR20770
        U(2,S)  = 2.D+00*U(2,SM1) - U(2,S-2)                             RBR20780
        IF (ABS(V(S)).GT.ABS(V(SM1))) V(S)=.9*V(SM1)                     RBR20790
        IF (U(2,S).LT.0.D+00) U(2,S)=.5*U(2,SM1)                         RBR20800
        IF (U(2,S).GT.U(2,SM1)) U(2,S)=.9*U(2,SM1)                       RBR20810
        U(1,S)  = V(S)*U(2,S)                                            RBR20820
        ELEV(S) = U(2,S) - U2INIT(S)                                     RBR20830
      ENDIF                                                              RBR20840
C                                                                        RBR20850
C             DETERMINE THE NEXT WATERLINE NODE                          RBR20860
C             ------------------------------------------------------     RBR20870
      IF (U(2,S).LE.DELTA) THEN                                          RBR20880
        SNEW = SM1                                                       RBR20890
      ELSE                                                               RBR20900
```

```
      V(SP1)    = 2.D+00*V(S) - V(SM1)                                    RBR20910
      U(2,SP1) = 2.D+00*U(2,S) - U(2,SM1)                                 RBR20920
      U(1,SP1) = V(SP1)*U(2,SP1)                                          RBR20930
      IF (U(2,SP1).LE.DELTA) THEN                                         RBR20940
        SNEW = S                                                          RBR20950
      ELSE                                                                RBR20960
C       ------------------------------------------------------------      RBR20970
C       ::      USN2(i),VSN2 = U(i,S) and V(S), respectively,             RBR20980
C               at time level (N+1), i=1,2                                RBR20990
        CALL MATAFG (MODE,SM1,SP1)                                        RBR21000
        CALL MATGJR (MODE,SM1,S)                                          RBR21010
        CALL MATS   (MODE,S,S)                                            RBR21020
        DUM1 = TX*((F(1,SP1)-F(1,SM1))/2.D+00+X*G1(S))                    RBR21030
        DUM2 = TTXX*(GJR(1,S)-GJR(1,SM1))                                 RBR21040
        DUM3 = TX*(F(2,SP1)-F(2,SM1))                                     RBR21050
        DUM4 = TTXX*(GJR(2,S)-GJR(2,SM1))                                 RBR21060
        USN2(1) = U(1,S)-DUM1+(DUM2-TTX*S1(S))/2.D+00                     RBR21070
        USN2(2) = U(2,S)-(DUM3-DUM4)/2.D+00                               RBR21080
        VSN2 = USN2(1)/USN2(2)                                            RBR21090
C       ------------------------------------------------------------      RBR21100
C       ::      US1N1(i),VS1N1 = U(i,S+1) and V(S+1), respectively,       RBR21110
C               at time level N, i=1,2                                    RBR21120
        VS = V(S)                                                         RBR21130
        IF (DABS(VS).LT.DELTA) VS=DSIGN(DELTA,VS)                         RBR21140
        DUM1  = (XT*(VSN2-VSN)+U(2,SP1)-U(2,SM1)+TWOX*THETA(S))/VS        RBR21150
        VS1N1 = V(SM1)-DUM1                                               RBR21160
        US1N1(1) = U(1,SM1) - XT*(USN2(2)-USN(2))                         RBR21170
        US1N1(2) = US1N1(1)/VS1N1                                         RBR21180
C       ------------------------------------------------------------      RBR21190
        IF (DABS(VS1N1).LE.DELTA) THEN                                    RBR21200
          SNEW = S                                                        RBR21210
        ELSE                                                              RBR21220
          IF (US1N1(2).LE.U(2,S)) THEN                                    RBR21230
            IF (US1N1(2).LE.DELTA) THEN                                   RBR21240
              SNEW = S                                                    RBR21250
            ELSE                                                          RBR21260
              SNEW   = SP1                                                RBR21270
              U(2,SP1) = US1N1(2)                                         RBR21280
              U(1,SP1) = US1N1(1)                                         RBR21290
              V(SP1)   = VS1N1                                            RBR21300
            ENDIF                                                         RBR21310
          ELSE                                                            RBR21320
            IF (U(2,SP1).LE.U(2,S)) THEN                                  RBR21330
              SNEW = SP1                                                  RBR21340
            ELSE                                                          RBR21350
              SNEW = S                                                    RBR21360
            ENDIF                                                         RBR21370
          ENDIF                                                           RBR21380
        ENDIF                                                             RBR21390
      ENDIF                                                               RBR21400
    ENDIF                                                                 RBR21410
    IF (SNEW.EQ.SP1) ELEV(SP1)=U(2,SP1)-U2INIT(SP1)                       RBR21420
    S = SNEW                                                              RBR21430
C                                                                         RBR21440
C   S at next time level has been found                                   RBR21450
```

```
C                                                                       RBR21460
      RETURN                                                            RBR21470
      END                                                               RBR21480
C                                                                       RBR21490
C -17------------------  END OF SUBROUTINE RUNUP  ------------RBREAK-   RBR21500
C #18#####################  SUBROUTINE OVERT  ################RBREAK#   RBR21510
C                                                                       RBR21520
C     This subroutine computes quantities at landward-end node for      RBR21530
C     IJOB=2 if overtopping occurs, that is, S=JE and SM1=(S-1)=JE1     RBR21540
C                                                                       RBR21550
      SUBROUTINE OVERT (MODE)                                           RBR21560
C                                                                       RBR21570
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR21580
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR21590
      INTEGER S,SM1,SP1                                                 RBR21600
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR21610
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR21620
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     RBR21630
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR21640
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)  RBR21650
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR21660
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                     RBR21670
      IF (MODE.EQ.0) THEN                                               RBR21680
        CALL CKFPAR (18,1,N1,N1R)                                       RBR21690
        RETURN                                                          RBR21700
      ENDIF                                                             RBR21710
      IF (VMN.GT.C(SM1)) THEN                                           RBR21720
        U(1,S) = USN(1) - TX*(F(1,S)-F(1,SM1)) - T*(THETA(S)*USN(2))    RBR21730
        U(2,S) = USN(2) - TX*(USN(1)-UMN(1))                            RBR21740
        V(S)   = U(1,S)/U(2,S)                                          RBR21750
      ELSE                                                              RBR21760
        VCS  = VSN + 2.D+00*C(S)                                        RBR21770
        VCM  = VMN + 2.D+00*C(SM1)                                      RBR21780
        V(S) = (VCS-TX*(VSN+C(S))*(VCS-VCM)-T*(THETA(S)))/3.D+00        RBR21790
        U(2,S) = V(S)*V(S)                                              RBR21800
        U(1,S) = V(S)*U(2,S)                                            RBR21810
      ENDIF                                                             RBR21820
      IF (U(2,S).LE.DELTA) THEN                                         RBR21830
        S = SM1                                                         RBR21840
      ELSE                                                              RBR21850
        ELEV(S) = U(2,S) - U2INIT(S)                                    RBR21860
      ENDIF                                                             RBR21870
C                                                                       RBR21880
      RETURN                                                            RBR21890
      END                                                               RBR21900
C                                                                       RBR21910
C -18------------------  END OF SUBROUTINE OVERT  ------------RBREAK-   RBR21920
C #19#####################  SUBROUTINE SEABC  ################RBREAK#   RBR21930
C                                                                       RBR21940
C     This subroutine treats seaward boundary conditions at node j=1    RBR21950
C                                                                       RBR21960
      SUBROUTINE SEABC (MODE,M,N,ETAI,ETAR)                             RBR21970
C                                                                       RBR21980
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR21990
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR22000
```

```
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                                RBR22010
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                RBR22020
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,         RBR22030
     +                ISAVA,ISAVB,ISAVC                                  RBR22040
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES             RBR22050
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX      RBR22060
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA           RBR22070
      COMMON /WAVE3/  ETA(N2),ETAIS(N2),ETARS(N2),ETATS(N2)              RBR22080
      COMMON /WAVE6/  NDATA,NDAT                                         RBR22090
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     RBR22100
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)      RBR22110
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)              RBR22120
      COMMON /STT1/   ELSTAT(3,3),U1STAT(N1),ESTAT(3,N1),VSTAT(3,N1),    RBR22130
     +                ELMEAN(3),U1MEAN(N1),EMEAN(N1),VMEAN(N1)           RBR22140
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                      RBR22150
      IF (MODE.EQ.0) THEN                                                RBR22160
        CALL CKFPAR (19,1,N1,N1R)                                        RBR22170
        CALL CKFPAR (19,2,N2,N2R)                                        RBR22180
        RETURN                                                           RBR22190
      ENDIF                                                              RBR22200
C                                                                        RBR22210
C           ESTIMATE ETAR                                                RBR22220
C           -------------------------------------------------------      RBR22230
C         . BETA = seaward-advancing characteristics                     RBR22240
C         . ETAR = surface elevation associated with reflected wave at   RBR22250
C           seaward boundary                                             RBR22260
C         . A correction term included in ETAR if INONCT=1 to improve    RBR22270
C           prediction of wave set-down and setup on beach               RBR22280
C           -------------------------------------------------------      RBR22290
      VC1 = -V1N+2.D+00*C(1)                                             RBR22300
      VC2 = -V2N+2.D+00*C(2)                                             RBR22310
      BETA = VC1 - TX*(V1N-C(1))*(VC2-VC1) + T*THETA(1)                  RBR22320
      ETAR = BETA*DSEA2/2.D+00 - DSEA                                    RBR22330
      IF (INONCT.EQ.1) ETAR=ETAR-KS*KS/(16.D+00*DSEA)                    RBR22340
C                                                                        RBR22350
C           VALUES AT NODE ONE                                           RBR22360
C           -------------------------------------------------------      RBR22370
      IF (IWAVE.EQ.1) THEN                                               RBR22380
C       :: Regular wave.                                                 RBR22390
C          Surface elevation ETA has been computed by Subr. 06 REGWAV.   RBR22400
        ETAI    = ETA(N+1)                                               RBR22410
        U(2,1)  = DSEA+ETAR+ETAI                                         RBR22420
        ELEV(1) = ETAI+ETAR                                              RBR22430
      ELSE                                                               RBR22440
C       :: Incident or total wave train is specified by user.            RBR22450
C          The specified time series ETA has NDATA points from the       RBR22460
C          normalized time t=0 to t=TMAX.                                RBR22470
C          Interpolate the specified time series to obtain the value     RBR22480
C          at t=TIME.                                                    RBR22490
        DJJ  = DBLE(NDATA-1)*TIME/TMAX                                   RBR22500
        JJ   = INT(DJJ)                                                  RBR22510
        ETA1 = ETA(JJ+1)                                                 RBR22520
        ETA2 = ETA(JJ+2)                                                 RBR22530
        DEL  = DJJ - DBLE(JJ)                                            RBR22540
        ETAI = ETA1 + DEL*(ETA2-ETA1)                                    RBR22550
```

```
        IF (IWAVE.EQ.2) THEN                                              RBR22560
          U(2,1)  = DSEA+ETAI+ETAR                                        RBR22570
          ELEV(1) = ETAI+ETAR                                             RBR22580
        ELSE                                                              RBR22590
          U(2,1)  = DSEA+ETAI                                             RBR22600
          ELEV(1) = ETAI                                                  RBR22610
        ENDIF                                                             RBR22620
      ENDIF                                                               RBR22630
      V(1)   = 2.D+00*DSQRT(U(2,1))-BETA                                  RBR22640
      U(1,1) = U(2,1)*V(1)                                                RBR22650
C                                                                         RBR22660
C           STATISTICAL CALCULATIONS                                      RBR22670
C           Performed excluding the first MSTAT computation units         RBR22680
C           ------------------------------------------------------        RBR22690
C         . ETAI = surface elevation associated with incident wave at     RBR22700
C           seaward boundary                                              RBR22710
C         . ETAR = surface elevation associated with reflected wave at    RBR22720
C           seaward boundary                                              RBR22730
C         . For regular wave (IWAVE=1), during the last computation       RBR22740
C           unit: ETAI,ETAR are saved in time-arrays ETAIS,ETARS          RBR22750
C           ------------------------------------------------------        RBR22760
      IF (M.GT.MSTAT) THEN                                                RBR22770
        IF (IWAVE.EQ.1) THEN                                              RBR22780
          ETAIS(N) = ETAI                                                 RBR22790
          ETARS(N) = ETAR                                                 RBR22800
        ELSE                                                              RBR22810
          IF (IWAVE.EQ.3) ETAI=ETAI-ETAR                                  RBR22820
          CALL STAT2 (ETAI,ELSTAT(1,1),ELSTAT(2,1),ELSTAT(3,1))           RBR22830
          CALL STAT2 (ETAR,ELSTAT(1,2),ELSTAT(2,2),ELSTAT(3,2))           RBR22840
        ENDIF                                                             RBR22850
      ENDIF                                                               RBR22860
C                                                                         RBR22870
      RETURN                                                              RBR22880
      END                                                                 RBR22890
C                                                                         RBR22900
C -19------------------  END OF SUBROUTINE SEABC  ------------RBREAK-     RBR22910
C #20#####################  SUBROUTINE MATAFG  ###############RBREAK#    RBR22920
C                                                                         RBR22930
C     This subroutine computes, for each node,                            RBR22940
C     . the elements of the the first row of Matrix A (2x2)               RBR22950
C                                          --> A1(1,j) and A1(2,j)        RBR22960
C     . the elements of Matrix F (2x1)     --> F(1,j) and F(2,j)          RBR22970
C     . the first element of Matrix G (2x1) --> G1(j)                     RBR22980
C     j=node number                                                       RBR22990
C                                                                         RBR23000
      SUBROUTINE MATAFG (MODE,JBEGIN,JEND)                                RBR23010
C                                                                         RBR23020
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                 RBR23030
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                        RBR23040
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                 RBR23050
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT      RBR23060
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)      RBR23070
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)               RBR23080
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)    RBR23090
      IF (MODE.EQ.0) THEN                                                 RBR23100
```

```
        CALL CKFPAR (20,1,N1,N1R)                                          RBR23110
        RETURN                                                             RBR23120
      ENDIF                                                                RBR23130
      DO 100 J = JBEGIN,JEND                                               RBR23140
        A1(1,J) = 2.D+00*V(J)                                              RBR23150
        A1(2,J) = U(2,J)-V(J)*V(J)                                         RBR23160
        F(1,J)  = V(J)*U(1,J) + U(2,J)*U(2,J)/2.D+00                       RBR23170
        F(2,J)  = U(1,J)                                                   RBR23180
        G1(J)   = THETA(J)*U(2,J) + FW*DABS(V(J))*V(J)                     RBR23190
  100 CONTINUE                                                             RBR23200
C                                                                          RBR23210
      RETURN                                                               RBR23220
      END                                                                  RBR23230
C                                                                          RBR23240
C -20------------------  END OF SUBROUTINE MATAFG  ------------RBREAK-     RBR23250
C #21#######################  SUBROUTINE MATGJR  ###############RBREAK#   RBR23260
C                                                                          RBR23270
C     This subroutine computes, for each node, the elements of             RBR23280
C     Matrix g (2x1) --> GJR(1,j) and GJR(2,j), j=node number              RBR23290
C                                                                          RBR23300
      SUBROUTINE MATGJR (MODE,JBEGIN,JEND)                                 RBR23310
C                                                                          RBR23320
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                  RBR23330
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                         RBR23340
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                  RBR23350
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX       RBR23360
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)     RBR23370
      IF (MODE.EQ.0) THEN                                                  RBR23380
        CALL CKFPAR (21,1,N1,N1R)                                          RBR23390
        RETURN                                                             RBR23400
      ENDIF                                                                RBR23410
C                                                                          RBR23420
      DO 100 J = JBEGIN,JEND                                               RBR23430
        FG1 = F(1,J+1)-F(1,J) + X*(G1(J+1)+G1(J))/2.D+00                   RBR23440
        FG2 = F(2,J+1)-F(2,J)                                              RBR23450
        DUM = (A1(1,J+1)+A1(1,J))*FG1 + (A1(2,J+1)+A1(2,J))*FG2            RBR23460
        GJR(1,J) = DUM/2.D+00                                              RBR23470
        GJR(2,J) = FG1                                                     RBR23480
  100 CONTINUE                                                             RBR23490
C                                                                          RBR23500
      RETURN                                                               RBR23510
      END                                                                  RBR23520
C                                                                          RBR23530
C -21------------------  END OF SUBROUTINE MATGJR  ------------RBREAK-     RBR23540
C #22#######################  SUBROUTINE MATS  ###############RBREAK#     RBR23550
C                                                                          RBR23560
C     This subroutine computes, for each node, the first element of        RBR23570
C     Matrix S (2x1) --> S1(j), j=node number                              RBR23580
C                                                                          RBR23590
      SUBROUTINE MATS (MODE,JBEGIN,JEND)                                   RBR23600
C                                                                          RBR23610
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                  RBR23620
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                         RBR23630
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                  RBR23640
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX       RBR23650
```

```
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     RBR23660
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     RBR23670
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR23680
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)  RBR23690
      IF (MODE.EQ.0) THEN                                               RBR23700
        CALL CKFPAR (22,1,N1,N1R)                                       RBR23710
        RETURN                                                          RBR23720
      ENDIF                                                             RBR23730
C                                                                       RBR23740
      DO 100 J = JBEGIN,JEND                                            RBR23750
        DUM1 = (V(J)*V(J)-U(2,J))*(U(2,J+1)-U(2,J-1))/TWOX              RBR23760
        DUM2 = V(J)*(U(1,J+1)-U(1,J-1))/TWOX                            RBR23770
        DUM3 = THETA(J)*U(2,J)                                          RBR23780
        DUM4 = FW*DABS(V(J))*V(J)                                       RBR23790
        DUM5 = 2.D+00*FW*DABS(V(J))/U(2,J)                              RBR23800
        EJN  = DUM5*(DUM1-DUM2-DUM3-DUM4)                               RBR23810
        S1(J) = X*EJN - THETA(J)*(U(1,J+1)-U(1,J-1))/2.D+00             RBR23820
  100 CONTINUE                                                          RBR23830
C                                                                       RBR23840
      RETURN                                                            RBR23850
      END                                                               RBR23860
C                                                                       RBR23870
C -22-------------------  END OF SUBROUTINE MATS  -------------RBREAK-  RBR23880
C #23#########################  SUBROUTINE MATD  ################RBREAK#  RBR23890
C                                                                       RBR23900
C     This subroutine computes, for each node, the elements of          RBR23910
C     Matrix D (2x1) --> D(1,j) and D(2,j), j=node number               RBR23920
C                                                                       RBR23930
      SUBROUTINE MATD (MODE,JDAM,JEND)                                  RBR23940
C                                                                       RBR23950
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR23960
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR23970
      DIMENSION Q(2,2,N1),UU(2,N1)                                      RBR23980
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR23990
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR24000
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR24010
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)  RBR24020
      IF (MODE.EQ.0) THEN                                               RBR24030
        CALL CKFPAR (23,1,N1,N1R)                                       RBR24040
        RETURN                                                          RBR24050
      ENDIF                                                             RBR24060
C                                                                       RBR24070
      DO 120 J = 1,JDAM                                                 RBR24080
        CC1 = C(J+1)+C(J)                                               RBR24090
        CC2 = C(J+1)-C(J)                                               RBR24100
        VC1 = V(J+1)+V(J)+CC1                                           RBR24110
        VC2 = V(J+1)-V(J)+CC2                                           RBR24120
        VC3 = V(J+1)+V(J)-CC1                                           RBR24130
        VC4 = V(J+1)-V(J)-CC2                                           RBR24140
        PPP = (-X1*DABS(VC2)*VC3+X2*DABS(VC4)*VC1)/(2.D+00*CC1)         RBR24150
        QQQ = (X1*DABS(VC2)-X2*DABS(VC4))/CC1                           RBR24160
        Q(1,1,J) = QQQ*(A1(1,J+1)+A1(1,J))/2.D+00 + PPP                 RBR24170
        Q(1,2,J) = QQQ*(A1(2,J+1)+A1(2,J))/2.D+00                       RBR24180
        Q(2,1,J) = QQQ                                                  RBR24190
        Q(2,2,J) = PPP                                                  RBR24200
```

```
      DO 110 I = 1,2                                                    RBR24210
        UU(I,J) = U(I,J+1)-U(I,J)                                       RBR24220
  110   CONTINUE                                                        RBR24230
  120 CONTINUE                                                          RBR24240
      DO 150 I = 1,2                                                    RBR24250
        DO 140 J = 2,JDAM                                               RBR24260
          D(I,J) = 0.D+00                                               RBR24270
          DO 130 L = 1,2                                                RBR24280
            D(I,J) = D(I,J) + Q(I,L,J)*UU(L,J) - Q(I,L,J-1)*UU(L,J-1)   RBR24290
  130     CONTINUE                                                      RBR24300
          D(I,J) = TX*D(I,J)/2.D+00                                     RBR24310
  140   CONTINUE                                                        RBR24320
  150 CONTINUE                                                          RBR24330
      IF (JEND.GT.JDAM) THEN                                            RBR24340
        DO 170 I = 1,2                                                  RBR24350
          DO 160 J = JDAM+1,JEND                                        RBR24360
            D(I,J) = 0.D+00                                             RBR24370
  160     CONTINUE                                                      RBR24380
  170   CONTINUE                                                        RBR24390
      ENDIF                                                             RBR24400
C                                                                       RBR24410
      RETURN                                                            RBR24420
      END                                                               RBR24430
C                                                                       RBR24440
C -23-------------------  END OF SUBROUTINE MATD  -------------RBREAK-  RBR24450
C #24#########################  SUBROUTINE MATU  ################RBREAK#  RBR24460
C                                                                       RBR24470
C     This subroutine computes the elements of Matrix U (2x1)           RBR24480
C     --> U(1,j) and U(2,j) with j=node number at next time level       RBR24490
C                                                                       RBR24500
      SUBROUTINE MATU (MODE,JBEGIN,JEND)                                RBR24510
C                                                                       RBR24520
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR24530
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR24540
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR24550
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR24560
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR24570
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)  RBR24580
      IF (MODE.EQ.0) THEN                                               RBR24590
        CALL CKFPAR (24,1,N1,N1R)                                       RBR24600
        RETURN                                                          RBR24610
      ENDIF                                                             RBR24620
C                                                                       RBR24630
      DO 100 J = JBEGIN,JEND                                            RBR24640
        DUM1 = TX*((F(1,J+1)-F(1,J-1))/2.D+00+X*G1(J))                  RBR24650
        DUM2 = TTXX*(GJR(1,J)-GJR(1,J-1))                               RBR24660
        DUM3 = TX*(F(2,J+1)-F(2,J-1))                                   RBR24670
        DUM4 = TTXX*(GJR(2,J)-GJR(2,J-1))                               RBR24680
        U(1,J) = U(1,J) - DUM1 + (DUM2-TTX*S1(J))/2.D+00 + D(1,J)       RBR24690
        U(2,J) = U(2,J) - (DUM3-DUM4)/2.D+00 + D(2,J)                   RBR24700
  100 CONTINUE                                                          RBR24710
C                                                                       RBR24720
      RETURN                                                            RBR24730
      END                                                               RBR24740
C                                                                       RBR24750
```

```
C -24-------------------  END OF SUBROUTINE MATU  -------------RBREAK-  RBR24760
C #25#########################  SUBROUTINE NSI  ##################RBREAK#  RBR24770
C                                                                        RBR24780
C     This subroutine calculates numerical stability indicator OMEGA     RBR24790
C                                                                        RBR24800
      SUBROUTINE NSI (OMEGA,NONE)                                        RBR24810
C                                                                        RBR24820
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                RBR24830
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX      RBR24840
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     RBR24850
      EPSI  = DMAX1(X1,X2)                                               RBR24860
      DUM1  = 1.D+00 + EPSI*EPSI/4.D+00                                  RBR24870
      DUM2  = DSQRT(DUM1) - EPSI/2.D+00                                  RBR24880
      OMEGA = DUM2*X*DBLE(NONE)/(1.D+00+DSQRT(DSEA))                     RBR24890
C                                                                        RBR24900
      RETURN                                                             RBR24910
      END                                                                RBR24920
C                                                                        RBR24930
C -25-------------------  END OF SUBROUTINE NSI  -------------RBREAK-  RBR24940
C #26#########################  SUBROUTINE INIT  ##################RBREAK#  RBR24950
C                                                                        RBR24960
C     This subroutine assigns initial values                             RBR24970
C                                                                        RBR24980
      SUBROUTINE INIT (MODE)                                             RBR24990
C                                                                        RBR25000
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                RBR25010
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                       RBR25020
      INTEGER S,SM1,SP1                                                  RBR25030
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                RBR25040
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,         RBR25050
     +                ISAVA,ISAVB,ISAVC                                  RBR25060
      COMMON /CPAR3/  INITS,JE,JE1                                       RBR25070
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX      RBR25080
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)      RBR25090
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)              RBR25100
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                               RBR25110
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3),RUNUPS(N3),RSTAT(3,N3),       RBR25120
     +                RMEAN(N3)                                          RBR25130
      COMMON /OVER/   OV(4),OVMEAN                                       RBR25140
      COMMON /STT1/   ELSTAT(3,3),U1STAT(N1),ESTAT(3,N1),VSTAT(3,N1),    RBR25150
     +                ELMEAN(3),U1MEAN(N1),EMEAN(N1),VMEAN(N1)           RBR25160
      COMMON /ENERG/  ENER(7,N1),ENERB(15),ENTEMP(3,N1)                  RBR25170
      COMMON /ARMOR3/ JSTAB,JSTABM                                       RBR25180
      COMMON /ARMOR4/ JSNSC,JATMIN                                       RBR25190
      COMMON /ARMOR5/ CSTAB1,CSTAB2,AMAXS,AMINS,E2,E3PRE(N1),SNR(N1),    RBR25200
     +                SNSX(N1),TSNSX(N1),ATMIN(4,N1),SNSC,TSNSC          RBR25210
      COMMON /ARMOR6/ NMOVE,NSTOP,ISTATE(N1),NODFI(N1)                   RBR25220
      COMMON /ARMOR7/ CSTAB3,CSTAB4,CM1,DA,SIGDA,WEIG,                   RBR25230
     +                VA(N1),XAA(N1),XA(N1),TDIS(N1)                     RBR25240
      IF (MODE.EQ.0) THEN                                                RBR25250
        CALL CKFPAR (26,1,N1,N1R)                                        RBR25260
        CALL CKFPAR (26,3,N3,N3R)                                        RBR25270
        RETURN                                                           RBR25280
      ENDIF                                                              RBR25290
C                                                                        RBR25300
```

```
C           HYDRODYNAMIC VARIABLES                                          RBR25310
C           At node j:                                                      RBR25320
C           ------------------------------------------------------------    RBR25330
C           U(1,j)   = volume flux                                          RBR25340
C           U(2,j)   = total water depth                                    RBR25350
C                      (not less than DELTA for IJOB=3)                     RBR25360
C           V(j)     = depth-averaged velocity                              RBR25370
C           ELEV(j) = surface elevation above SWL                           RBR25380
C           ------------------------------------------------------------    RBR25390
      DO 110 J = 1,JE                                                       RBR25400
        U(1,J) = 0.D+00                                                     RBR25410
        IF (J.LE.INITS) THEN                                                RBR25420
          U(2,J) = U2INIT(J)                                                RBR25430
          ELEV(J)= 0.D+00                                                   RBR25440
        ELSE                                                                RBR25450
          U(2,J) = 0.D+00                                                   RBR25460
          ELEV(J)= ZB(J)                                                    RBR25470
        ENDIF                                                               RBR25480
        V(J) = 0.D+00                                                       RBR25490
        IF (IJOB.EQ.3.AND.U(2,J).LT.DELTA) U(2,J)=DELTA                     RBR25500
  110 CONTINUE                                                              RBR25510
C                                                                           RBR25520
C           STATISTICAL QUANTITIES                                          RBR25530
C           ------------------------------------------------------------    RBR25540
C         . Subr. 27 INITH is used to specify initial values for            RBR25550
C           statistical quantities                                          RBR25560
C         . U1STAT(j) = mean volume flux at node j                          RBR25570
C         . Mean, maximum, and minimum at node j:                           RBR25580
C               ESTAT(1,j),ESTAT(2,j),ESTAT(3,j): for ELEV(j)               RBR25590
C               VSTAT(1,j),VSTAT(2,j),VSTAT(3,j): for V(j)                  RBR25600
C         . Mean, maximum, and minimum at seaward boundary:                 RBR25610
C               ELSTAT(1,1),ELSTAT(2,1),ELSTAT(3,1):                        RBR25620
C                                              due to incident wave         RBR25630
C               ELSTAT(1,2),ELSTAT(2,2),ELSTAT(3,2):                        RBR25640
C                                              due to reflected wave        RBR25650
C         . Mean, maximum, and minimum at landward boundary:                RBR25660
C               ELSTAT(1,3),ELSTAT(2,3),ELSTAT(3,3):                        RBR25670
C                                              due to transmitted wave      RBR25680
C         . The variable on the left is used as temporary storage in        RBR25690
C           the process of calculating the variable on the right            RBR25700
C           (index j indicates node number):                                RBR25710
C               U1MEAN(j) <-- U1STAT(j)                                     RBR25720
C               EMEAN(j)  <-- ESTAT(1,j)                                    RBR25730
C               VMEAN(j)  <-- VSTAT(1,j)                                    RBR25740
C               ELMEAN(k) <-- ELSTAT(1,k) with k=1,2,3                      RBR25750
C           ------------------------------------------------------------    RBR25760
      CALL INITH (1,U1STAT,1,JE)                                            RBR25770
      CALL INITH (2,ESTAT, 3,JE)                                            RBR25780
      CALL INITH (2,VSTAT, 3,JE)                                            RBR25790
      CALL INITH (2,ELSTAT,3,3)                                             RBR25800
      CALL INITH (1,U1MEAN,1,JE)                                            RBR25810
      CALL INITH (1,EMEAN, 1,JE)                                            RBR25820
      CALL INITH (1,VMEAN, 1,JE)                                            RBR25830
      CALL INITH (1,ELMEAN,1,3)                                             RBR25840
C                                                                           RBR25850
```

```
C           RUNUP                                                       RBR25860
C           ------------------------------------------------------------RBR25870
C         . S = node number closest to the instantaneous computational  RBR25880
C           waterline                                                   RBR25890
C         . JMAX = the largest node number reached by computational     RBR25900
C           waterline                                                   RBR25910
C         . RSTAT(1,k),RSTAT(2,k),RSTAT(3,k) = mean, maximum, and       RBR25920
C           minimum of RUNUPS(k) (See Subr. 16 LANDBC for RUNUPS)       RBR25930
C         . RMEAN(k) is used as temporary storage in the process of     RBR25940
C           calculating RSTAT(1,k)                                      RBR25950
C         . k=1,2,...,NDELR                                             RBR25960
C           ------------------------------------------------------------RBR25970
      S    = INITS                                                      RBR25980
      JMAX = INITS                                                      RBR25990
      IF (IJOB.LT.3) THEN                                               RBR26000
        CALL INITH (2,RSTAT,3,NDELR)                                    RBR26010
        CALL INITH (1,RMEAN,1,NDELR)                                    RBR26020
      ELSE                                                              RBR26030
        IF (ISTAB.GT.1) JSTAB=JE                                        RBR26040
      ENDIF                                                             RBR26050
C                                                                       RBR26060
C           OVERTOPPING                                                 RBR26070
C           ------------------------------------------------------------RBR26080
C           OV(1) = normalized average overtopping rate                 RBR26090
C           OV(2) = normalized maximum overtopping rate                 RBR26100
C           OV(3) = normalized overtopping duration                     RBR26110
C           OV(4) = normalized time when OV(2) occurs                   RBR26120
C           OVMEAN is used as temporary storage in the process of       RBR26130
C                 calculating OV(1)                                     RBR26140
C           ------------------------------------------------------------RBR26150
      IF (IJOB.EQ.2) THEN                                               RBR26160
        CALL INITH (1,OV,1,4)                                           RBR26170
        OVMEAN = 0.D+00                                                 RBR26180
      ENDIF                                                             RBR26190
C                                                                       RBR26200
C           WAVE ENERGY                                                 RBR26210
C           ------------------------------------------------------------RBR26220
C         . See Subr. 30 ENERGY for description of ENER(i,j) with       RBR26230
C           i=1,2,...,7                                                 RBR26240
C         . ENTEMP(k,j) is used as temporary storage in the process of  RBR26250
C           calculating ENER(k,j) at node j with k=1,2,3                RBR26260
C           ------------------------------------------------------------RBR26270
      IF (IENERG.EQ.1) THEN                                             RBR26280
        CALL INITH (1,ENER  ,7,JE)                                      RBR26290
        CALL INITH (1,ENTEMP,3,JE)                                      RBR26300
      ENDIF                                                             RBR26310
C                                                                       RBR26320
C           ARMOR STABILITY OR MOVEMENT                                 RBR26330
C           ------------------------------------------------------------RBR26340
C         . See Subr. 31 STABNO for description of variables            RBR26350
C           associated with ISTAB=1 (computation of armor stability)    RBR26360
C         . See Subr. 32 MOVE for description of variables              RBR26370
C           associated with ISTAB=2 (computation of armor movement)     RBR26380
C           ------------------------------------------------------------RBR26390
      IF (ISTAB.GT.0) JSTABM=0                                          RBR26400
```

```
      IF (ISTAB.EQ.1) THEN                                              RBR26410
        SNSC = 1.D+03                                                   RBR26420
        DO 130 J = 1,JE                                                 RBR26430
          SNSX(J) = 1.D+03                                              RBR26440
  130   CONTINUE                                                        RBR26450
      ELSEIF (ISTAB.EQ.2) THEN                                          RBR26460
        NMOVE = 0                                                       RBR26470
        NSTOP = 0                                                       RBR26480
        DO 140 J = 1,JE                                                 RBR26490
          ISTATE(J) = 0                                                 RBR26500
          NODFI(J)  = J                                                 RBR26510
          VA(J)     = 0.D+00                                            RBR26520
          XAA(J)    = 0.D+00                                            RBR26530
  140   CONTINUE                                                        RBR26540
      ENDIF                                                             RBR26550
C                                                                       RBR26560
      RETURN                                                            RBR26570
      END                                                               RBR26580
C                                                                       RBR26590
C -26-------------------  END OF SUBROUTINE INIT  -------------RBREAK-  RBR26600
C #27#######################  SUBROUTINE INITH  ################RBREAK#  RBR26610
C                                                                       RBR26620
C     This subroutine facilitates assignment of initial values in       RBR26630
C     Subr. 26 INIT                                                     RBR26640
C                                                                       RBR26650
      SUBROUTINE INITH (IMODE,VAL,ND1,ND2)                              RBR26660
C                                                                       RBR26670
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR26680
      DIMENSION VAL(ND1,ND2)                                            RBR26690
      IF (IMODE.EQ.1) THEN                                              RBR26700
        DO 120 I = 1,ND1                                                RBR26710
          DO 110 J = 1,ND2                                              RBR26720
            VAL(I,J) = 0.D+00                                           RBR26730
  110     CONTINUE                                                      RBR26740
  120   CONTINUE                                                        RBR26750
      ELSE                                                              RBR26760
        DO 130 J = 1,ND2                                                RBR26770
          VAL(1,J) =  0.D+00                                            RBR26780
          VAL(2,J) = -1.D+03                                            RBR26790
          VAL(3,J) =  1.D+03                                            RBR26800
  130   CONTINUE                                                        RBR26810
      ENDIF                                                             RBR26820
C                                                                       RBR26830
      RETURN                                                            RBR26840
      END                                                               RBR26850
C                                                                       RBR26860
C -27-------------------  END OF SUBROUTINE INITH  ------------RBREAK-  RBR26870
C #28#######################  SUBROUTINE INSTAT  ################RBREAK#  RBR26880
C                                                                       RBR26890
C     This subroutine initializes statistical quantities representing   RBR26900
C     mean values at the beginning of a computation unit, since         RBR26910
C     delta t = (1/NONE) may vary from one unit to another              RBR26920
C                                                                       RBR26930
      SUBROUTINE INSTAT (MODE)                                          RBR26940
C                                                                       RBR26950
```

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                RBR26960
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                       RBR26970
      INTEGER S,SM1,SP1                                                  RBR26980
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                RBR26990
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,         RBR27000
     +                ISAVA,ISAVB,ISAVC                                  RBR27010
      COMMON /CPAR3/  INITS,JE,JE1                                       RBR27020
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                               RBR27030
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3),RUNUPS(N3),RSTAT(3,N3),       RBR27040
     +                RMEAN(N3)                                          RBR27050
      COMMON /OVER/   OV(4),OVMEAN                                       RBR27060
      COMMON /STT1/   ELSTAT(3,3),U1STAT(N1),ESTAT(3,N1),VSTAT(3,N1),    RBR27070
     +                ELMEAN(3),U1MEAN(N1),EMEAN(N1),VMEAN(N1)           RBR27080
      COMMON /ENERG/  ENER(7,N1),ENERB(15),ENTEMP(3,N1)                  RBR27090
      IF (MODE.EQ.0) THEN                                                RBR27100
        CALL CKFPAR (28,1,N1,N1R)                                        RBR27110
        CALL CKFPAR (28,3,N3,N3R)                                        RBR27120
        RETURN                                                           RBR27130
      ENDIF                                                              RBR27140
C                                                                        RBR27150
      DO 110 J = 1,2                                                     RBR27160
        ELSTAT(1,J) = 0.D+00                                             RBR27170
  110 CONTINUE                                                           RBR27180
      DO 120 J = 1,JE                                                    RBR27190
        U1STAT(J)   = 0.D+00                                             RBR27200
        ESTAT(1,J) = 0.D+00                                              RBR27210
        VSTAT(1,J) = 0.D+00                                              RBR27220
  120 CONTINUE                                                           RBR27230
      DO 130 J = 1,NDELR                                                 RBR27240
        RSTAT(1,J) = 0.D+00                                              RBR27250
  130 CONTINUE                                                           RBR27260
      IF (IJOB.EQ.2) THEN                                                RBR27270
        OV(1) = 0.D+00                                                   RBR27280
      ELSEIF (IJOB.EQ.3) THEN                                            RBR27290
        ELSTAT(1,3) = 0.D+00                                             RBR27300
      ENDIF                                                              RBR27310
      IF (IENERG.EQ.1) THEN                                              RBR27320
        DO 150 L = 1,3                                                   RBR27330
          DO 140 J = 1,JE                                                RBR27340
            ENER(L,J) = 0.D+00                                           RBR27350
  140     CONTINUE                                                       RBR27360
  150   CONTINUE                                                         RBR27370
      ENDIF                                                              RBR27380
C                                                                        RBR27390
      RETURN                                                             RBR27400
      END                                                                RBR27410
C                                                                        RBR27420
C -28------------------  END OF SUBROUTINE INSTAT  ------------RBREAK-   RBR27430
C #29######################  SUBROUTINE SVSTAT  ###############RBREAK#   RBR27440
C                                                                        RBR27450
C     This subroutine saves statistical quantities related to            RBR27460
C     mean values at the end of a computation unit, since                RBR27470
C     delta t = (1/NONE) may vary from one unit to another               RBR27480
C                                                                        RBR27490
      SUBROUTINE SVSTAT (MODE)                                           RBR27500
```

```
C                                                                       RBR27510
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR27520
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR27530
      INTEGER S,SM1,SP1                                                 RBR27540
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR27550
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,        RBR27560
     +                ISAVA,ISAVB,ISAVC                                 RBR27570
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES            RBR27580
      COMMON /CPAR2/  MULTIF,NONEM,NONM,NONE,NEND,NRATE,NHOP            RBR27590
      COMMON /CPAR3/  INITS,JE,JE1                                      RBR27600
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR27610
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR27620
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3),RUNUPS(N3),RSTAT(3,N3),      RBR27630
     +                RMEAN(N3)                                         RBR27640
      COMMON /OVER/   OV(4),OVMEAN                                      RBR27650
      COMMON /STT1/   ELSTAT(3,3),U1STAT(N1),ESTAT(3,N1),VSTAT(3,N1),   RBR27660
     +                ELMEAN(3),U1MEAN(N1),EMEAN(N1),VMEAN(N1)          RBR27670
      COMMON /ENERG/  ENER(7,N1),ENERB(15),ENTEMP(3,N1)                 RBR27680
      IF (MODE.EQ.0) THEN                                               RBR27690
        CALL CKFPAR (29,1,N1,N1R)                                       RBR27700
        CALL CKFPAR (29,1,N3,N3R)                                       RBR27710
        RETURN                                                          RBR27720
      ENDIF                                                             RBR27730
C                                                                       RBR27740
      CONFAC = DBLE(NONE)*(TMAX-DBLE(MSTAT))                            RBR27750
      DO 110 J = 1,2                                                    RBR27760
        ELMEAN(J) = ELMEAN(J) + ELSTAT(1,J)/CONFAC                      RBR27770
  110 CONTINUE                                                          RBR27780
      DO 120 J = 1,JE                                                   RBR27790
        U1MEAN(J) = U1MEAN(J) + U1STAT(J)/CONFAC                        RBR27800
        EMEAN(J)  = EMEAN(J)  + ESTAT(1,J)/CONFAC                       RBR27810
        VMEAN(J)  = VMEAN(J)  + VSTAT(1,J)/CONFAC                       RBR27820
  120 CONTINUE                                                          RBR27830
      DO 130 J = 1,NDELR                                                RBR27840
        RMEAN(J) = RMEAN(J) + RSTAT(1,J)/CONFAC                         RBR27850
  130 CONTINUE                                                          RBR27860
      IF (IJOB.EQ.2) THEN                                               RBR27870
        OVMEAN = OVMEAN + OV(1)/CONFAC                                  RBR27880
      ELSEIF (IJOB.EQ.3) THEN                                           RBR27890
        ELMEAN(3) = ELMEAN(3) + ELSTAT(1,3)/CONFAC                      RBR27900
      ENDIF                                                             RBR27910
      IF (IENERG.EQ.1) THEN                                             RBR27920
        DO 150 L = 1,3                                                  RBR27930
          DO 140 J = 1,JE                                               RBR27940
            ENTEMP(L,J) = ENTEMP(L,J) + ENER(L,J)/CONFAC                RBR27950
  140     CONTINUE                                                      RBR27960
  150   CONTINUE                                                        RBR27970
      ENDIF                                                             RBR27980
C                                                                       RBR27990
      RETURN                                                            RBR28000
      END                                                               RBR28010
C                                                                       RBR28020
C -29------------------  END OF SUBROUTINE SVSTAT  ------------RBREAK-  RBR28030
C #30######################  SUBROUTINE ENERGY  ###############RBREAK#  RBR28040
C                                                                       RBR28050
```

```
C     This subroutine computes quantities related to wave energy          RBR28060
C                                                                         RBR28070
C     At node j:                                                          RBR28080
C       E(i,j) with i=1,2,3 is instantaneous quantity                     RBR28090
C       ENER(i,j) with i=1,2,3,4 is time-averaged quantity                RBR28100
C       E(1,j),ENER(1,j) = norm. energy per unit surface area             RBR28110
C       E(2,j),ENER(2,j) = norm. energy flux per unit width               RBR28120
C       ENER(5,j) saves the value of E(1,j) at t=TSTAT1                   RBR28130
C       ENER(6,j) saves the value of E(1,j) at t=TSTAT2                   RBR28140
C     Normalized rate of energy dissipation at node j:                    RBR28150
C       E(3,j),ENER(3,j): due to bottom friction, per unit bottom area    RBR28160
C       ENER(4,j): due to wave breaking, per unit surface area            RBR28170
C     TSTAT1,TSTAT2 = normalized time at the beginning and at the end,    RBR28180
C       respectively, of statistical calculations. Special treatment     RBR28190
C       at t=TSTAT1 and t=TSTAT2 is provided only for random waves        RBR28200
C       (IWAVE>1) and is handled by ICALL=9.                              RBR28210
C     At node j:                                                          RBR28220
C       ENER(7,j) = (ENER(6,j)-ENER(5,j))/(TSTAT2-TSTAT1) for random      RBR28230
C       waves in Subr. 39 ENERGC                                          RBR28240
C                                                                         RBR28250
      SUBROUTINE ENERGY (ICALL,M)                                         RBR28260
C                                                                         RBR28270
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                 RBR28280
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                        RBR28290
      DIMENSION E(3,N1)                                                   RBR28300
      INTEGER S,SM1,SP1                                                   RBR28310
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                 RBR28320
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES              RBR28330
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX       RBR28340
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)               RBR28350
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT       RBR28360
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                                RBR28370
      COMMON /STT2/   TSTAT1,TSTAT2                                       RBR28380
      COMMON /ENERG/  ENER(7,N1),ENERB(15),ENTEMP(3,N1)                   RBR28390
      IF (ICALL.EQ.0) THEN                                                RBR28400
C                  ::  Checking mode  ::                                  RBR28410
        CALL CKFPAR (30,1,N1,N1R)                                         RBR28420
        RETURN                                                            RBR28430
      ELSEIF (ICALL.EQ.9) THEN                                            RBR28440
C                  ::  Special treatment  ::                              RBR28450
        DO 110 J = 1,S                                                    RBR28460
          E(1,J) = (U(1,J)*V(J) + ELEV(J)*ELEV(J))/2.D+00                 RBR28470
          IF (U(2,J).LT.ELEV(J)) THEN                                     RBR28480
            E(1,J) = E(1,J) - (U(2,J)-ELEV(J))**2/2.D+00                  RBR28490
          ENDIF                                                           RBR28500
  110   CONTINUE                                                          RBR28510
        IF (M.EQ.MSTAT) THEN                                              RBR28520
          TSTAT1 = TIME                                                   RBR28530
          DO 120 J = 1,S                                                  RBR28540
            ENER(5,J) = E(1,J)                                            RBR28550
  120     CONTINUE                                                        RBR28560
        ELSEIF (M.EQ.MWAVE) THEN                                          RBR28570
          TSTAT2 = TIME                                                   RBR28580
          TSPAN  = TSTAT2-TSTAT1                                          RBR28590
          DO 130 J = 1,S                                                  RBR28600
```

```
            ENER(6,J) = E(1,J)                                          RBR28610
  130     CONTINUE                                                      RBR28620
        ENDIF                                                           RBR28630
        RETURN                                                          RBR28640
      ENDIF                                                             RBR28650
C                 ::  Normal mode  ::                                   RBR28660
      DO 220 J = 1,S                                                    RBR28670
        E(1,J) = (U(1,J)*V(J) + ELEV(J)*ELEV(J))/2.D+00                 RBR28680
        IF (U(2,J).LT.ELEV(J)) THEN                                     RBR28690
          E(1,J) = E(1,J) - (U(2,J)-ELEV(J))**2/2.D+00                  RBR28700
        ENDIF                                                           RBR28710
        E(2,J) = U(1,J)*(V(J)*V(J)/2.D+00+ELEV(J))                      RBR28720
        E(3,J) = FW*DABS(V(J))*V(J)*V(J)                                RBR28730
        DO 210 L = 1,3                                                  RBR28740
          ENER(L,J) = ENER(L,J) + E(L,J)                                RBR28750
  210   CONTINUE                                                        RBR28760
  220 CONTINUE                                                          RBR28770
C                                                                       RBR28780
      RETURN                                                            RBR28790
      END                                                               RBR28800
C                                                                       RBR28810
C -30------------------  END OF SUBROUTINE ENERGY  ------------RBREAK-  RBR28820
C #31#####################  SUBROUTINE STABNO  ###############RBREAK#  RBR28830
C                                                                       RBR28840
C     This subroutine computes stability number against                 RBR28850
C     rolling/sliding, SNR                                              RBR28860
C                                                                       RBR28870
      SUBROUTINE STABNO (MODE)                                          RBR28880
C                                                                       RBR28890
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR28900
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR28910
      INTEGER S,SM1,SP1                                                 RBR28920
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR28930
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR28940
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     RBR28950
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     RBR28960
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR28970
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR28980
      COMMON /ARMOR2/ SG1,CTAN(N1)                                      RBR28990
      COMMON /ARMOR3/ JSTAB,JSTABM                                      RBR29000
      COMMON /ARMOR4/ JSNSC,JATMIN                                      RBR29010
      COMMON /ARMOR5/ CSTAB1,CSTAB2,AMAXS,AMINS,E2,E3PRE(N1),SNR(N1),   RBR29020
     +                SNSX(N1),TSNSX(N1),ATMIN(4,N1),SNSC,TSNSC         RBR29030
      IF (MODE.EQ.0) THEN                                               RBR29040
        CALL CKFPAR (31,1,N1,N1R)                                       RBR29050
        RETURN                                                          RBR29060
      ENDIF                                                             RBR29070
C                                                                       RBR29080
C           FLUID ACCELERATION                                          RBR29090
C           Computed in Subr. 34 ACCEL                                  RBR29100
C           ------------------------------------------------------      RBR29110
      CALL ACCEL (MODE)                                                 RBR29120
C                                                                       RBR29130
C           STABILITY NUMBER SNR                                        RBR29140
C           ------------------------------------------------------      RBR29150
```

```
C         . SNR(j) = stability number against rolling/sliding at        RBR29160
C           node j                                                      RBR29170
C         . SNR is computed for j=1,2,...,JSTAB where JSTAB is defined  RBR29180
C           in Subr. 16 LANDBC                                          RBR29190
C           ------------------------------------------------------      RBR29200
      DO 110 J = 1,JSTAB                                                RBR29210
C                                                                       RBR29220
        IF (DABS(V(J)).LT.1.D-03) THEN                                  RBR29230
C         ::            Avoid having very small velocity values       RBR29240
C                       SNR=1000 indicates very stable units          RBR29250
          SNR(J) = 1.D+03                                               RBR29260
        ELSE                                                            RBR29270
C         ::            Impose lower and upper bounds of fluid        RBR29280
C                       acceleration                                  RBR29290
          IF (DUDT(J).GT.AMAXS) DUDT(J)=AMAXS                           RBR29300
          IF (DUDT(J).LT.AMINS) DUDT(J)=AMINS                           RBR29310
C         ::            SNR=-1000 indicates that AMAX and AMIN        RBR29320
C                       specified in Subr. 02 INPUT1 needs to be      RBR29330
C                       modified                                      RBR29340
          VALUE = CSTAB2*DUDT(J)-SSLOPE(J)                              RBR29350
          ABSV  = DABS(VALUE)                                           RBR29360
          IF (ABSV.GT.CTAN(J)) THEN                                     RBR29370
            SNR(J) = -1.D+03                                            RBR29380
            WRITE  (*,2910) TIME,J                                      RBR29390
            WRITE (29,2910) TIME,J                                      RBR29400
            STOP                                                        RBR29410
          ENDIF                                                         RBR29420
C         ::            Compute SNR                                   RBR29430
          E1    = VALUE*CSTAB1/(V(J)*DABS(V(J)))                        RBR29440
          E3    = E3PRE(J)/(V(J)*V(J))                                  RBR29450
          E1E2 = -E1*E2                                                 RBR29460
          IF (E1.LT.0.D+00.AND.E2.GT.1.D+00.AND.E3.LT.E1E2) THEN        RBR29470
            SNR(J) = (E3+E1)/(E2-1.D+00)                                RBR29480
          ELSE                                                          RBR29490
            SNR(J) = (E3-E1)/(E2+1.D+00)                                RBR29500
          ENDIF                                                         RBR29510
        ENDIF                                                           RBR29520
 2910 FORMAT (' From Subr. 31 STABNO'/' Armor stability impossible'/    RBR29530
     +       ' Time =',F18.6,'; J =',I8)                                RBR29540
C                                                                       RBR29550
  110 CONTINUE                                                          RBR29560
C                                                                       RBR29570
C           FIND SNSX                                                   RBR29580
C           ------------------------------------------------------      RBR29590
C           SNSX(j)  = local stability number                           RBR29600
C                    = minimum of SNR at node j                         RBR29610
C           TSNSX(j) = normalized time when SNSX(j) occurs              RBR29620
C           ------------------------------------------------------      RBR29630
      DO 120 J = 1,JSTAB                                                RBR29640
        IF (SNR(J).LT.SNSX(J)) THEN                                     RBR29650
          SNSX(J)  = SNR(J)                                             RBR29660
          TSNSX(J) = TIME                                               RBR29670
        ENDIF                                                           RBR29680
  120 CONTINUE                                                          RBR29690
C                                                                       RBR29700
```

```
C           CRITICAL STABILITY NUMBER, SNSC                                 RBR29710
C           ------------------------------------------------------          RBR29720
C           SNSC  = critical stability number                               RBR29730
C                 = minimum of SNSX along the slope                         RBR29740
C           TSNSC = normalized time when SNSC occurs                        RBR29750
C           JSNSC = node number where SNSC occurs                           RBR29760
C           At the time of minimum stability, i.e., t=TSNSC:                RBR29770
C           . ATMIN(i,j) saves spatial variations, i=1,2,3,4                RBR29780
C           . JATMIN = the value of JSTAB                                   RBR29790
C           ------------------------------------------------------          RBR29800
      INDI = 0                                                              RBR29810
      DO 130 J = 1,JSTAB                                                    RBR29820
        IF (SNSX(J).LT.SNSC) THEN                                           RBR29830
          SNSC  = SNSX(J)                                                   RBR29840
          JSNSC = J                                                         RBR29850
          TSNSC = TSNSX(J)                                                  RBR29860
          INDI  = 1                                                         RBR29870
        ENDIF                                                               RBR29880
  130 CONTINUE                                                              RBR29890
      IF (INDI.EQ.1) THEN                                                   RBR29900
        JATMIN = JSTAB                                                      RBR29910
        DO 140 J = 1,JATMIN                                                 RBR29920
          ATMIN(1,J) = ELEV(J)                                              RBR29930
          ATMIN(2,J) = V(J)                                                 RBR29940
          ATMIN(3,J) = DUDT(J)                                              RBR29950
          ATMIN(4,J) = SNR(J)                                               RBR29960
  140   CONTINUE                                                            RBR29970
      ENDIF                                                                 RBR29980
C                                                                           RBR29990
      RETURN                                                                RBR30000
      END                                                                   RBR30010
C                                                                           RBR30020
C -31------------------  END OF SUBROUTINE STABNO  ------------RBREAK-      RBR30030
C #32#####################  SUBROUTINE MOVE  ################RBREAK#      RBR30040
C                                                                           RBR30050
C     This subroutine computes movement of armor units                      RBR30060
C                                                                           RBR30070
      SUBROUTINE MOVE (MODE)                                                RBR30080
C                                                                           RBR30090
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                   RBR30100
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                          RBR30110
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                   RBR30120
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX         RBR30130
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)                 RBR30140
      COMMON /ARMOR1/ C2,C3,CD,CL,CM,SG,TANPHI,AMIN,AMAX,DAP                RBR30150
      COMMON /ARMOR3/ JSTAB,JSTABM                                          RBR30160
      COMMON /ARMOR6/ NMOVE,NSTOP,ISTATE(N1),NODFI(N1)                      RBR30170
      COMMON /ARMOR7/ CSTAB3,CSTAB4,CM1,DA,SIGDA,WEIG,                      RBR30180
     +                VA(N1),XAA(N1),XA(N1),TDIS(N1)                        RBR30190
      IF (MODE.EQ.0) THEN                                                   RBR30200
        CALL CKFPAR (32,1,N1,N1R)                                           RBR30210
        RETURN                                                              RBR30220
      ENDIF                                                                 RBR30230
C                                                                           RBR30240
C           FLUID ACCELERATION                                              RBR30250
```

```
C           Computed in Subr. 34 ACCEL                                  RBR30260
C           ----------------------------------------------------------  RBR30270
      CALL ACCEL (MODE)                                                 RBR30280
                                                                        RBR30290
C                                                                       RBR30300
C           GENERAL TERMS                                               RBR30310
C           ----------------------------------------------------------  RBR30320
C         . Counters:                                                   RBR30330
C           NMOVE = number of armor units dislodged from their initial  RBR30340
C                   locations                                           RBR30350
C           NSTOP = number of armor units stopped after moving          RBR30360
C         . Node numbers:                                               RBR30370
C           JSTAB = the largest node number for which armor movement    RBR30380
C                   is computed                                         RBR30390
C           For moving/stopped armor unit number j:                     RBR30400
C           NODFI(j) = node number closest to the armor unit at the     RBR30410
C                      end of each time step                            RBR30420
C         . Dynamics:                                                   RBR30430
C           FDES = normalized destabilizing force                       RBR30440
C           FR   = normalized resistance force                          RBR30450
C           FDES and FR are computed in Subr. 33 FORCES                 RBR30460
C           ISTATE(j) indicates the state of armor unit initially       RBR30470
C             located at node j: 0=stationary, 1=moving, 2=stopped      RBR30480
C           For moving/stopped armor unit number j:                     RBR30490
C           + VA(j) = normalized velocity                               RBR30500
C           + XAA(j), XA(j) = displacement from its initial location,   RBR30510
C             normalized by TP*sqrt(GRAV*HREFP) and DAP, respectively   RBR30520
C           + TDIS(j) = norm. time when it starts moving the first      RBR30530
C             time                                                      RBR30540
C         . It is assumed that once an armor unit is dislodged from a   RBR30550
C           node, no other unit will be dislodged from the same node    RBR30560
C           ----------------------------------------------------------  RBR30570
      DO 120 J = 1,JSTAB                                                RBR30580
        IF (ISTATE(J).EQ.0) THEN                                        RBR30590
C         ::      STATIONARY ARMOR UNIT                                 RBR30600
C                 Check whether the unit at node j starts moving        RBR30610
          CALL FORCES (MODE,V(J),J,FDES,FR)                             RBR30620
          IF (DABS(FDES).GT.FR) THEN                                    RBR30630
            IF (FDES.LT.0.D+00) FR=-FR                                  RBR30640
            NMOVE     = NMOVE+1                                         RBR30650
            ISTATE(J) = 1                                               RBR30660
            TDIS(J)   = TIME                                            RBR30670
            VA(J)     = (FDES-FR)*T/(SG+CM1)                            RBR30680
            XAA(J)    = .5D+00*VA(J)*T                                  RBR30690
            NODFI(J)  = J + NINT(XAA(J)/X)                              RBR30700
          ENDIF                                                         RBR30710
        ELSEIF (ISTATE(J).EQ.1) THEN                                    RBR30720
C         ::      MOVING ARMOR UNIT                                     RBR30730
C                 Follow the moving unit initially located at node j    RBR30740
          NOD  = NODFI(J)                                               RBR30750
          VREL = V(NOD) - VA(J)                                         RBR30760
          CALL FORCES (MODE,VREL,NOD,FDES,FR)                           RBR30770
          FR       = FR*(VA(J)/DABS(VA(J)))                             RBR30780
          DVA      = (FDES-FR)*T/(SG+CM1)                               RBR30790
          DXAA     = (VA(J)+.5D+00*DVA)*T                               RBR30800
          VA(J)    = VA(J) + DVA                                        RBR30810
```

```
          XAA(J)    = XAA(J) + DXAA                                      RBR30810
          NODFI(J) = J + NINT(XAA(J)/X)                                  RBR30820
C                 Check whether the moving unit identified by the        RBR30830
C                 initial node j stops at the end of each time step      RBR30840
          IF (DABS(VA(J)).LT.1.D-06.AND.DABS(FDES).LT.DABS(FR)) THEN     RBR30850
            ISTATE(J) = 2                                                RBR30860
            NSTOP     = NSTOP+1                                          RBR30870
          ENDIF                                                          RBR30880
        ELSE                                                             RBR30890
C         ::      STOPPED ARMOR UNIT                                     RBR30900
C                 Check whether the stopped armor unit located           RBR30910
C                 initially at node j resumes movement                   RBR30920
          NOD = NODFI(J)                                                 RBR30930
          CALL FORCES (MODE,V(NOD),NOD,FDES,FR)                          RBR30940
          IF (DABS(FDES).GT.FR) THEN                                     RBR30950
            IF (FDES.LT.0.D+00) FR=-FR                                   RBR30960
            VA(J)     = (FDES-FR)*T/(SG+CM1)                             RBR30970
            XAA(J)    = XAA(J) + .5D+00*VA(J)*T                          RBR30980
            NODFI(J)  = J + NINT(XAA(J)/X)                               RBR30990
            NSTOP     = NSTOP-1                                          RBR31000
            ISTATE(J) = 1                                                RBR31010
          ENDIF                                                          RBR31020
        ENDIF                                                            RBR31030
  120 CONTINUE                                                           RBR31040
C                                                                        RBR31050
C           COMPUTE XA                                                   RBR31060
C           ------------------------------------------------------       RBR31070
      DO 130 J = 1,JSTAB                                                 RBR31080
        IF (ISTATE(J).GE.1) XA(J)=XAA(J)*SIGDA                           RBR31090
  130 CONTINUE                                                           RBR31100
C                                                                        RBR31110
      RETURN                                                             RBR31120
      END                                                                RBR31130
C                                                                        RBR31140
C -32------------------  END OF SUBROUTINE MOVE  ------------RBREAK-     RBR31150
C #33#####################  SUBROUTINE FORCES  ###############RBREAK#   RBR31160
C                                                                        RBR31170
C     This subroutine computes destabilizing force FDES and resistance   RBR31180
C     force FR used in Subr. 32 MOVE                                     RBR31190
C                                                                        RBR31200
      SUBROUTINE FORCES (MODE,VELO,NODE,FDES,FR)                         RBR31210
C                                                                        RBR31220
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                RBR31230
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                       RBR31240
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                                RBR31250
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                RBR31260
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA           RBR31270
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)      RBR31280
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)              RBR31290
      COMMON /ARMOR1/ C2,C3,CD,CL,CM,SG,TANPHI,AMIN,AMAX,DAP             RBR31300
      COMMON /ARMOR7/ CSTAB3,CSTAB4,CM1,DA,SIGDA,WEIG,                   RBR31310
     +                VA(N1),XAA(N1),XA(N1),TDIS(N1)                     RBR31320
      COMMON /ARMOR2/ SG1,CTAN(N1)                                       RBR31330
      IF (MODE.EQ.0) THEN                                                RBR31340
        CALL CKFPAR (33,1,N1,N1R)                                        RBR31350
```

```
        RETURN                                                          RBR31360
      ENDIF                                                             RBR31370
C                                                                       RBR31380
C     WEIG = normalized submerged weight of armor unit defined in       RBR31390
C            Subr. 40 SPARAM                                            RBR31400
C     WSIN = component of WEIG parallel to local slope                  RBR31410
C     WCOS = component of WEIG normal to local slope                    RBR31420
C     FD   = normalized drag force                                      RBR31430
C     FL   = normalized lift force                                      RBR31440
C     FI   = normalized inertia force due to fluid only                 RBR31450
C     FDES = normalized destabilizing force = FD+FI-WSIN                RBR31460
C     FR   = normalized resistance or friction force                    RBR31470
C          = 0 if (WCOS-FL)<0; no contact with other units              RBR31480
C     Note: FR returned to Subr. 32 MOVE is positive or zero            RBR31490
C                                                                       RBR31500
      WSIN = WEIG*SSLOPE(NODE)                                          RBR31510
      WCOS = WEIG*CTAN(NODE)/TANPHI                                     RBR31520
      FD   = SIGMA*CSTAB3*VELO*DABS(VELO)                               RBR31530
      FL   = SIGMA*CSTAB4*VELO*VELO                                     RBR31540
      FI   = CM*DUDT(NODE)                                              RBR31550
      FDES = FD + FI - WSIN                                             RBR31560
      FR   = (WCOS-FL)*TANPHI                                           RBR31570
      IF (FR.LE.0.D+00) FR=0.D+00                                       RBR31580
C                                                                       RBR31590
      RETURN                                                            RBR31600
      END                                                               RBR31610
C                                                                       RBR31620
C -33------------------  END OF SUBROUTINE FORCES  ------------RBREAK-  RBR31630
C #34#######################  SUBROUTINE ACCEL  ################RBREAK#  RBR31640
C                                                                       RBR31650
C     This subroutine computes total fluid acceleration using           RBR31660
C     Subr. 41 VECMAT and Subr. 42 DERIV                                RBR31670
C                                                                       RBR31680
      SUBROUTINE ACCEL (MODE)                                           RBR31690
C                                                                       RBR31700
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR31710
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR31720
      DIMENSION VDUM1(N1),VDUM2(N1)                                     RBR31730
      INTEGER S,SM1,SP1                                                 RBR31740
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR31750
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR31760
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     RBR31770
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     RBR31780
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR31790
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR31800
      IF (MODE.EQ.0) THEN                                               RBR31810
        CALL CKFPAR (34,1,N1,N1R)                                       RBR31820
        RETURN                                                          RBR31830
      ENDIF                                                             RBR31840
      CALL VECMAT (VDUM1,U,2,S,2)                                       RBR31850
      CALL DERIV  (VDUM1,VDUM2,X,S)                                     RBR31860
      DO 100 J = 1,S                                                    RBR31870
        DUDT(J) = -VDUM2(J)-THETA(J)-FW*V(J)*DABS(V(J))/U(2,J)          RBR31880
  100 CONTINUE                                                          RBR31890
C                                                                       RBR31900
```

```
      RETURN                                                            RBR31910
      END                                                               RBR31920
C                                                                       RBR31930
C -34------------------  END OF SUBROUTINE ACCEL  ------------RBREAK-   RBR31940
C #35######################  SUBROUTINE STAT1  ################RBREAK#  RBR31950
C                                                                       RBR31960
C     For MODE=1, VAL1(1,J) is sum of VAL2(J)                           RBR31970
C     For MODE=2, VAL1(1,J) is sum of VAL2(J)                           RBR31980
C                 VAL1(2,J) is maximum of VAL2(J)                       RBR31990
C                 VAL1(3,J) is minimum of VAL2(J)                       RBR32000
C                                                                       RBR32010
      SUBROUTINE STAT1 (IMODE,VAL1,VAL2,ND1,ND2)                        RBR32020
C                                                                       RBR32030
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR32040
      DIMENSION VAL1(ND1,ND2),VAL2(ND2)                                 RBR32050
      IF (IMODE.EQ.1) THEN                                              RBR32060
        DO 120 I = 1,ND1                                                RBR32070
          DO 110 J = 1,ND2                                              RBR32080
            VAL1(I,J) = VAL1(I,J)+VAL2(J)                               RBR32090
  110     CONTINUE                                                      RBR32100
  120   CONTINUE                                                        RBR32110
      ELSE                                                              RBR32120
        DO 130 J = 1,ND2                                                RBR32130
          VAL1(1,J) = VAL1(1,J)+VAL2(J)                                 RBR32140
          IF (VAL2(J).GT.VAL1(2,J)) VAL1(2,J)=VAL2(J)                   RBR32150
          IF (VAL2(J).LT.VAL1(3,J)) VAL1(3,J)=VAL2(J)                   RBR32160
  130   CONTINUE                                                        RBR32170
      ENDIF                                                             RBR32180
      RETURN                                                            RBR32190
      END                                                               RBR32200
C                                                                       RBR32210
C -35------------------  END OF SUBROUTINE STAT1  ------------RBREAK-   RBR32220
C #36######################  SUBROUTINE STAT2  ################RBREAK#  RBR32230
C                                                                       RBR32240
C     This subroutine computes sum, maximum value, and minimum value    RBR32250
C     of time series VAL                                                RBR32260
C                                                                       RBR32270
      SUBROUTINE STAT2 (VAL,VALAVE,VALMAX,VALMIN)                       RBR32280
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR32290
      VALAVE=VALAVE+VAL                                                 RBR32300
      IF (VAL.GT.VALMAX) VALMAX=VAL                                     RBR32310
      IF (VAL.LT.VALMIN) VALMIN=VAL                                     RBR32320
      RETURN                                                            RBR32330
      END                                                               RBR32340
C                                                                       RBR32350
C -36------------------  END OF SUBROUTINE STAT2  ------------RBREAK-   RBR32360
C #37######################  SUBROUTINE STATC  ################RBREAK#  RBR32370
C                                                                       RBR32380
C     This subroutine completes statistical calculations                RBR32390
C                                                                       RBR32400
      SUBROUTINE STATC (MODE)                                           RBR32410
C                                                                       RBR32420
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR32430
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR32440
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                               RBR32450
```

```
      INTEGER S,SM1,SP1                                                 RBR32460
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR32470
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,        RBR32480
     +                ISAVA,ISAVB,ISAVC                                 RBR32490
      COMMON /CPAR2/  MULTIF,NONEM,NONM,NONE,NEND,NRATE,NHOP            RBR32500
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA          RBR32510
      COMMON /WAVE3/  ETA(N2),ETAIS(N2),ETARS(N2),ETATS(N2)             RBR32520
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR32530
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3),RUNUPS(N3),RSTAT(3,N3),      RBR32540
     +                RMEAN(N3)                                         RBR32550
      COMMON /OVER/   OV(4),OVMEAN                                      RBR32560
      COMMON /COEFS/  RCOEF(3),TCOEF(3)                                 RBR32570
      COMMON /STT1/   ELSTAT(3,3),U1STAT(N1),ESTAT(3,N1),VSTAT(3,N1),   RBR32580
     +                ELMEAN(3),U1MEAN(N1),EMEAN(N1),VMEAN(N1)          RBR32590
      IF (MODE.EQ.0) THEN                                               RBR32600
        CALL CKFPAR (37,1,N1,N1R)                                       RBR32610
        CALL CKFPAR (37,2,N2,N2R)                                       RBR32620
        CALL CKFPAR (37,3,N3,N3R)                                       RBR32630
        RETURN                                                          RBR32640
      ENDIF                                                             RBR32650
C                                                                       RBR32660
C           REFLECTION AND TRANSMISSION COEFFICIENTS AND MEAN SURFACE   RBR32670
C           ELEVATIONS AT BOUNDARIES                                    RBR32680
C           ------------------------------------------------------      RBR32690
C         . RCOEF(i),TCOEF(i) = reflection,transmission coefficient of  RBR32700
C           the i-th kind with i=1,2,3                                  RBR32710
C         . For regular waves, all the three kinds of coefficients are  RBR32720
C           calculated using Subr. 38 COEF. For random waves, only the  RBR32730
C           first kind is calculated.                                   RBR32740
C         . Mean surface elevations:                                    RBR32750
C             ELSTAT(1,1): due to incident wave                         RBR32760
C             ELSTAT(1,2): due to reflected wave                        RBR32770
C             ELSTAT(1,3): due to transmitted wave                      RBR32780
C           ------------------------------------------------------      RBR32790
      IF (IWAVE.EQ.1) THEN                                              RBR32800
        CALL COEF (1,DUM  ,DUM1,ETAIS,NONE)                             RBR32810
        CALL COEF (2,RCOEF,DUM2,ETARS,NONE)                             RBR32820
        ELSTAT(1,1) = DUM1                                              RBR32830
        ELSTAT(1,2) = DUM2                                              RBR32840
        IF (IJOB.EQ.3) THEN                                             RBR32850
          CALL COEF (2,TCOEF,DUM3,ETATS,NONE)                           RBR32860
          ELSTAT(1,3) = DUM3                                            RBR32870
        ENDIF                                                           RBR32880
      ELSE                                                              RBR32890
        ELSTAT(1,1) = ELMEAN(1)                                         RBR32900
        ELSTAT(1,2) = ELMEAN(2)                                         RBR32910
        RCOEF(1) = (ELSTAT(2,2)-ELSTAT(3,2))/KS                         RBR32920
        IF (IJOB.EQ.3) THEN                                             RBR32930
          ELSTAT(1,3) = ELMEAN(3)                                       RBR32940
          TCOEF(1) = (ELSTAT(2,3)-ELSTAT(3,3))/KS                       RBR32950
        ENDIF                                                           RBR32960
      ENDIF                                                             RBR32970
C                                                                       RBR32980
C           WAVE SETUP ON SLOPE COMPUTED FROM WATERLINE MOTION          RBR32990
C           ------------------------------------------------------      RBR33000
```

```
      DO 110 J = 1,NDELR                                                RBR33010
        RSTAT(1,J) = RMEAN(J)                                           RBR33020
  110 CONTINUE                                                          RBR33030
C                                                                       RBR33040
C           MEAN HYDRODYNAMIC QUANTITIES                                RBR33050
C           ------------------------------------------------------      RBR33060
C           Mean value at node j:                                       RBR33070
C           . U1STAT(j):  volume flux                                   RBR33080
C           . ESTAT(1,j): surface elevation above SWL                   RBR33090
C           . VSTAT(1,j): depth-averaged velocity                       RBR33100
C           ------------------------------------------------------      RBR33110
      DO 120 J = 1,JMAX                                                 RBR33120
        U1STAT(J)  = U1MEAN(J)                                          RBR33130
        ESTAT(1,J) = EMEAN(J)                                           RBR33140
        VSTAT(1,J) = VMEAN(J)                                           RBR33150
  120 CONTINUE                                                          RBR33160
C                                                                       RBR33170
C           AVERAGE OVERTOPPING RATE                                    RBR33180
C           ------------------------------------------------------      RBR33190
      OV(1) = OVMEAN                                                    RBR33200
C                                                                       RBR33210
      RETURN                                                            RBR33220
      END                                                               RBR33230
C                                                                       RBR33240
C -37------------------  END OF SUBROUTINE STATC  ------------RBREAK-   RBR33250
C #38#########################  SUBROUTINE COEF  ################RBREAK#  RBR33260
C                                                                       RBR33270
C     This subroutine computes:                                         RBR33280
C     . time-averaged value of given quantity, VAL (MODE=1)             RBR33290
C     . reflection or transmission coefficients (three kinds) (MODE=2)  RBR33300
C       for monochromatic incident waves                                RBR33310
C                                                                       RBR33320
      SUBROUTINE COEF (IMODE,COE,AVER1,VAL,ND)                          RBR33330
C                                                                       RBR33340
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR33350
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                               RBR33360
      DIMENSION COE(3),VAL(ND)                                          RBR33370
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA          RBR33380
      SUM1 = 0.D+00                                                     RBR33390
      DO 110 I = 1,ND                                                   RBR33400
        SUM1 = SUM1 + VAL(I)                                            RBR33410
  110 CONTINUE                                                          RBR33420
      AVER1 = SUM1/DBLE(ND)                                             RBR33430
      IF (IMODE.EQ.2) THEN                                              RBR33440
        VALMAX = -1.D+03                                                RBR33450
        VALMIN =  1.D+03                                                RBR33460
        SUM2   =  0.D+00                                                RBZ33470
        SUM3   =  0.D+00                                                RBR33480
        DO 120 I = 1,ND                                                 RBR33490
          IF (VAL(I).GT.VALMAX) VALMAX = VAL(I)                         RBR33500
          IF (VAL(I).LT.VALMIN) VALMIN = VAL(I)                         RBR33510
          SUM2 = SUM2 + VAL(I)*VAL(I)                                   RBR33520
          SUM3 = SUM3 + (VAL(I)-AVER1)**2                               RBR33530
  120   CONTINUE                                                        RBR33540
        AVER2 = SUM2/DBLE(ND)                                           RBR33550
```

```
        AVER3 = SUM3/DBLE(ND)                                              RBR33560
C           ------------------------------------------------------         RBR33570
C           Monochromatic incident wave profile is assumed to be given     RBR33580
C           by ETAI=(KS/2)COS[2*PI*(t+t0)] since linear wave theory is     RBR33590
C           normally used to estimate reflection and transmission          RBR33600
C           coefficients                                                   RBR33610
C           ------------------------------------------------------         RBR33620
        COE(1) = (VALMAX-VALMIN)/KS                                        RBR33630
        COE(2) = DSQRT(8.D+00*AVER2)/KS                                    RBR33640
        COE(3) = DSQRT(8.D+00*AVER3)/KS                                    RBR33650
      ENDIF                                                                RBR33660
C                                                                          RBR33670
      RETURN                                                               RBR33680
      END                                                                  RBR33690
C                                                                          RBR33700
C -38-------------------  END OF SUBROUTINE COEF  -------------RBREAK-     RBR33710
C #39#####################  SUBROUTINE ENERGC  ###############RBREAK#     RBR33720
C                                                                          RBR33730
C     This subroutine completes computation of energy quantities           RBR33740
C                                                                          RBR33750
      SUBROUTINE ENERGC (MODE)                                             RBR33760
C                                                                          RBR33770
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                  RBR33780
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                         RBR33790
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                                  RBR33800
      DIMENSION VDUM1(N1),VDUM2(N1)                                        RBR33810
      INTEGER S,SM1,SP1                                                    RBR33820
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                  RBR33830
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,           RBR33840
     +                ISAVA,ISAVB,ISAVC                                    RBR33850
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX        RBR33860
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT       RBR33870
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA             RBR33880
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                                 RBR33890
      COMMON /COEFS/  RCOEF(3),TCOEF(3)                                    RBR33900
      COMMON /STT2/   TSTAT1,TSTAT2                                        RBR33910
      COMMON /ENERG/  ENER(7,N1),ENERB(15),ENTEMP(3,N1)                    RBR33920
      IF (MODE.EQ.0) THEN                                                  RBR33930
        CALL CKFPAR (39,1,N1,N1R)                                          RBR33940
        RETURN                                                             RBR33950
      ENDIF                                                                RBR33960
C                                                                          RBR33970
C           ------------------------------------------------------         RBR33980
C         . TSTAT1,TSTAT2 = normalized time at the beginning and at        RBR33990
C           the end, respectively, of statistical calculations             RBR34000
C         . Time-averaged quantities at node j:                            RBR34010
C             ENER(1,j) = norm. energy per unit surface area               RBR34020
C             ENER(2,j) = norm. energy flux per unit width                 RBR34030
C             ENER(3,j) = norm. rate of energy dissipation due to          RBR34040
C                         bottom friction, per unit bottom area            RBR34050
C             ENER(4,j) = norm. rate of energy dissipation due to          RBR34060
C                         wave breaking, per unit surface area             RBR34070
C         . Norm. energy per unit surface area at node j:                  RBR34080
C             ENER(5,j): at t=TSTAT1                                       RBR34090
C             ENER(6,j): at t=TSTAT2                                       RBR34100
```

```
C          . At node j:                                                  RBR34110
C              ENER(7,j) = the last term in Eq. 57 of RBREAK manual      RBR34120
C            ------------------------------------------------------      RBR34130
      DO 120 L = 1,3                                                     RBR34140
        DO 110 J = 1,JMAX                                                RBR34150
          ENER(L,J) = ENTEMP(L,J)                                        RBR34160
  110   CONTINUE                                                         RBR34170
  120 CONTINUE                                                           RBR34180
C           :: Use Subr. 41 VECMAT and Subr. 42 DERIV                    RBR34190
      CALL VECMAT (VDUM1,ENER,7,JMAX,2)                                  RBR34200
      CALL DERIV  (VDUM1,VDUM2,X,JMAX)                                   RBR34210
      DO 140 J = 1,JMAX                                                  RBR34220
        ENER(4,J) = -VDUM2(J)-ENER(3,J)                                  RBR34230
  140 CONTINUE                                                           RBR34240
      IF (IWAVE.GT.1) THEN                                               RBR34250
        TSPAN = TSTAT2-TSTAT1                                            RBR34260
        DO 150 J = 1,JMAX                                                RBR34270
          ENER(7,J) = (ENER(6,J)-ENER(5,J))/TSPAN                        RBR34280
          ENER(4,J) = ENER(4,J)-ENER(7,J)                                RBR34290
  150   CONTINUE                                                         RBR34300
      ENDIF                                                              RBR34310
C                                                                        RBR34320
C            ------------------------------------------------------      RBR34330
C          . Normalized energy flux at boundaries:                       RBR34340
C              ENERB(1): at seaward boundary                             RBR34350
C              ENERB(2): at landward boundary                            RBR34360
C          . Normalized rate of energy dissipation in the computation    RBR34370
C            domain:                                                     RBR34380
C              ENERB(3): due to bottom friction                          RBR34390
C              ENERB(4): due to wave breaking                            RBR34400
C          . ENERB(5) = the last integral in Eq. 58 of RBREAK manual     RBR34410
C            ------------------------------------------------------      RBR34420
      ENERB(1) = ENER(2,1)                                               RBR34430
      IF (IJOB.EQ.1) THEN                                                RBR34440
        ENERB(2) = 0.D+00                                                RBR34450
      ELSE                                                               RBR34460
        ENERB(2) = ENER(2,JMAX)                                          RBR34470
      ENDIF                                                              RBR34480
      DO 170 I = 3,5                                                     RBR34490
        IF (I.LT.5) THEN                                                 RBR34500
          IE=I                                                           RBR34510
        ELSE                                                             RBR34520
          IE=7                                                           RBR34530
        ENDIF                                                            RBR34540
        ENERB(I) = (ENER(IE,1)+ENER(IE,JMAX))/2.D+00                     RBR34550
        DO 160 J = 2,JMAX-1                                              RBR34560
          ENERB(I) = ENERB(I) + ENER(IE,J)                               RBR34570
  160   CONTINUE                                                         RBR34580
        ENERB(I) = ENERB(I)*X                                            RBR34590
  170 CONTINUE                                                           RBR34600
C                                                                        RBR34610
      ENERB(6) = ENERB(1) - ENERB(2)                                     RBR34620
      ENERB(7) = ENERB(3) + ENERB(4) + ENERB(5)                          RBR34630
      ENERB(8) = ENERB(7) - ENERB(6)                                     RBR34640
      ENERB(9) = 100.D+00*ENERB(8)/ENERB(6)                              RBR34650
```

```
C                                                                       RBR34660
C          ------------------------------------------------------       RBR34670
C          Approximate energy flux based on regular linear long wave:   RBR34680
C          ENERB(10): due to incident wave at seaward boundary          RBR34690
C          ENERB(11): due to reflected wave at seaward boundary         RBR34700
C          ENERB(12): due to transmitted wave at landward boundary      RBR34710
C          ------------------------------------------------------       RBR34720
      IF (IWAVE.EQ.1) THEN                                              RBR34730
        ENERB(10) = KS*KS*DSEA2/8.D+00                                  RBR34740
        ENERB(11) = DSEA2*(KS*RCOEF(3))**2/8.D+00                       RBR34750
        ENERB(13) = ENERB(10)-ENERB(11)                                 RBR34760
        ENERB(14) = 100.D+00*(ENERB(13)-ENERB(1))/ENERB(1)              RBR34770
        IF (IJOB.EQ.3) THEN                                             RBR34780
          ENERB(12) = DLAND2*(KS*TCOEF(3))**2/8.D+00                    RBR34790
          ENERB(15) = 100.D+00*(ENERB(12)-ENERB(2))/ENERB(2)            RBR34800
        ENDIF                                                           RBR34810
      ENDIF                                                             RBR34820
C                                                                       RBR34830
      RETURN                                                            RBR34840
      END                                                               RBR34850
C                                                                       RBR34860
C -39------------------  END OF SUBROUTINE ENERGC  ------------RBREAK-  RBR34870
C #40#####################  SUBROUTINE SPARAM  ###############RBREAK#  RBR34880
C                                                                       RBR34890
C     This subroutine calculates parameters needed for computation      RBR34900
C     of armor stability or movement                                    RBR34910
C                                                                       RBR34920
      SUBROUTINE SPARAM (MODE)                                          RBR34930
C                                                                       RBR34940
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR34950
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR34960
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                               RBR34970
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR34980
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,        RBR34990
     +                ISAVA,ISAVB,ISAVC                                 RBR35000
      COMMON /CPAR3/  INITS,JE,JE1                                      RBR35010
      COMMON /WAVE1/  HREFP,TP,WL0P                                     RBR35020
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA          RBR35030
      COMMON /ARMOR1/ C2,C3,CD,CL,CM,SG,TANPHI,AMIN,AMAX,DAP            RBR35040
      COMMON /ARMOR2/ SG1,CTAN(N1)                                      RBR35050
      COMMON /ARMOR3/ JSTAB,JSTABM                                      RBR35060
      COMMON /ARMOR5/ CSTAB1,CSTAB2,AMAXS,AMINS,E2,E3PRE(N1),SNR(N1),   RBR35070
     +                SNSX(N1),TSNSX(N1),ATMIN(4,N1),SNSC,TSNSC         RBR35080
      COMMON /ARMOR7/ CSTAB3,CSTAB4,CM1,DA,SIGDA,WEIG,                  RBR35090
     +                VA(N1),XAA(N1),XA(N1),TDIS(N1)                    RBR35100
      IF (MODE.EQ.0) THEN                                               RBR35110
        CALL CKFPAR (40,1,N1,N1R)                                       RBR35120
        RETURN                                                          RBR35130
      ENDIF                                                             RBR35140
C                                                                       RBR35150
C           ------------------------------------------------------      RBR35160
C          . SG = specific gravity                                      RBR35170
C          . Coefficients:                                              RBR35180
C            C2: area, C3: volume, CD: drag, CL: lift, CM: inertia      RBR35190
C          . TANPHI = armor friction factor                             RBR35200
```

```
C           . AMAX,AMIN = upper,lower bounds of fluid acceleration,     RBR35210
C             normalized by gravitational acceleration                 RBR35220
C           . E3PRE is prepared for computing E3 in Subr. 31 STABNO     RBR35230
C           . DAP,DA = physical,normalized armor diameters             RBR35240
C           . WEIG = normalized submerged weight of armor unit         RBR35250
C             ---------------------------------------------------      RBR35260
      SG1 = SG-1.D+00                                                   RBR35270
      IF (IJOB.EQ.3) JSTAB=JE                                           RBR35280
      IF (ISTAB.EQ.1) THEN                                              RBR35290
        CSTAB1 = 2.D+00*C3**(2.D+00/3.D+00)/(C2*CD)                     RBR35300
        CSTAB2 = CM/(SG1*SIGMA)                                         RBR35310
        E2     = CL*TANPHI/CD                                           RBR35320
        AMAXS  = AMAX*SIGMA                                             RBR35330
        AMINS  = AMIN*SIGMA                                             RBR35340
        DO 100 J = 1,JE                                                 RBR35350
          E3PRE(J) = CSTAB1*CTAN(J)                                     RBR35360
  100   CONTINUE                                                        RBR35370
      ELSE                                                              RBR35380
        CM1    = CM - 1.D+00                                            RBR35390
        DA     = DAP/HREFP                                              RBR35400
        SIGDA  = SIGMA/DA                                               RBR35410
        CSTAB3 = C2*CD/(2.D+00*C3*DA)                                   RBR35420
        CSTAB4 = C2*CL/(2.D+00*C3*DA)                                   RBR35430
        WEIG   = SIGMA*SG1                                              RBR35440
      ENDIF                                                             RBR35450
C                                                                       RBR35460
      RETURN                                                            RBR35470
      END                                                               RBR35480
C                                                                       RBR35490
C -40------------------  END OF SUBROUTINE SPARAM  ------------RBREAK-  RBR35500
C #41####################  SUBROUTINE VECMAT  ################RBREAK#  RBR35510
C                                                                       RBR35520
C     This subroutine creates vector VAL1 from row NROW of matrix VAL2  RBR35530
C                                                                       RBR35540
      SUBROUTINE VECMAT (VAL1,VAL2,ND1,ND2,NROW)                        RBR35550
C                                                                       RBR35560
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR35570
      DIMENSION VAL1(ND2),VAL2(ND1,ND2)                                 RBR35580
      DO 100 J = 1,ND2                                                  RBR35590
        VAL1(J) = VAL2(NROW,J)                                          RBR35600
  100 CONTINUE                                                          RBR35610
      RETURN                                                            RBR35620
      END                                                               RBR35630
C                                                                       RBR35640
C -41------------------  END OF SUBROUTINE VECMAT  ------------RBREAK-  RBR35650
C #42####################  SUBROUTINE DERIV  ################RBREAK#  RBR35660
C                                                                       RBR35670
C     This subroutine computes the first derivative, DER, of given     RBR35680
C     quantity, FUN, with respect to given variable, VAR, for           RBR35690
C     J=1,2,...,ND                                                      RBR35700
C                                                                       RBR35710
      SUBROUTINE DERIV (FUN,DER,VAR,ND)                                 RBR35720
C                                                                       RBR35730
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR35740
      DIMENSION FUN(ND),DER(ND)                                         RBR35750
```

```
      VAR2 = 2.D+00*VAR                                                 RBR35760
      DER(1)  = (FUN(2)-FUN(1))/VAR                                     RBR35770
      DER(ND) = (FUN(ND)-FUN(ND-1))/VAR                                 RBR35780
      DO 100 J = 2,ND-1                                                 RBR35790
        DER(J) = (FUN(J+1)-FUN(J-1))/VAR2                               RBR35800
  100 CONTINUE                                                          RBR35810
      RETURN                                                            RBR35820
      END                                                               RBR35830
C                                                                       RBR35840
C -42------------------  END OF SUBROUTINE DERIV  ------------RBREAK-   RBR35850
C #43#######################  SUBROUTINE DOC1  ################RBREAK#  RBR35860
C                                                                       RBR35870
C     This subroutine documents essential information before            RBR35880
C     time-marching computation                                         RBR35890
C                                                                       RBR35900
      SUBROUTINE DOC1 (MODE)                                            RBR35910
C                                                                       RBR35920
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR35930
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR35940
      DOUBLE PRECISION KCNO,MCNO,KC2,KS,KSREF,KSSEA,KSI                 RBR35950
      CHARACTER*7 UL                                                    RBR35960
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR35970
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,        RBR35980
     +                ISAVA,ISAVB,ISAVC                                 RBR35990
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES            RBR36000
      COMMON /CPAR2/  MULTIF,NONEM,NONM,NONE,NEND,NRATE,NHOP            RBR36010
      COMMON /CPAR3/  INITS,JE,JE1                                      RBR36020
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR36030
      COMMON /WAVE1/  HREFP,TP,WL0P                                     RBR36040
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA          RBR36050
      COMMON /WAVE4/  ETAMAX,ETAMIN                                     RBR36060
      COMMON /WAVE5/  KCNO,ECNO,MCNO,KC2                                RBR36070
      COMMON /WAVE6/  NDATA,NDAT                                        RBR36080
      COMMON /BOT1/   DSEAP,DLANDP,FWP                                  RBR36090
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     RBR36100
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     RBR36110
      COMMON /BOT4/   NBSEG                                             RBR36120
      COMMON /BOT5/   WBSEG(N4),TBSLOP(N4),XBSEG(N4),ZBSEG(N4)          RBR36130
      COMMON /ARMOR1/ C2,C3,CD,CL,CM,SG,TANPHI,AMIN,AMAX,DAP            RBR36140
      IF (MODE.EQ.0) THEN                                               RBR36150
        CALL CKFPAR (43,1,N1,N1R)                                       RBR36160
        CALL CKFPAR (43,4,N4,N4R)                                       RBR36170
        RETURN                                                          RBR36180
      ENDIF                                                             RBR36190
C                                                                       RBR36200
C           SYSTEM OF UNITS                                             RBR36210
C           ------------------------------------------------------      RBR36220
      IF (ISYST.EQ.1) THEN                                              RBR36230
        UL = ' meters'                                                  RBR36240
      ELSE                                                              RBR36250
        UL = ' feet  '                                                  RBR36260
      ENDIF                                                             RBR36270
C                                                                       RBR36280
C           WAVE CONDITION                                              RBR36290
C           ------------------------------------------------------      RBR36300
```

```
      WRITE (98,9811)                                                   RBR36310
      IF (IWAVE.EQ.1) THEN                                              RBR36320
        IF (URPRE.LT.26.D+00) THEN                                      RBR36330
          WRITE (98,9812)                                               RBR36340
        ELSE                                                            RBR36350
          WRITE (98,9813) KC2,ECNO,KCNO                                 RBR36360
        ENDIF                                                           RBR36370
      ELSEIF (IWAVE.EQ.2) THEN                                          RBR36380
        WRITE (98,9814) NDATA                                           RBR36390
      ELSE                                                              RBR36400
        WRITE (98,9815) NDATA                                           RBR36410
      ENDIF                                                             RBR36420
      WRITE (98,9816) ETAMAX,ETAMIN                                     RBR36430
      WRITE (98,9817) TP,HREFP,UL,DSEAP,UL,KSREF,KSSEA,KS               RBR36440
      WRITE (98,9818) DSEA,WL,SIGMA,UR,KSI                              RBR36450
      IF (IJOB.EQ.3)  WRITE (98,9819) DLANDP,UL                         RBR36460
 9811 FORMAT (/'WAVE CONDITION')                                        RBR36470
 9812 FORMAT (/'Stokes II Incident Wave at Seaward Boundary')           RBR36480
 9813 FORMAT (/'Cnoidal Incident Wave at Seaward Boundary'/             RBR36490
     +         '1-m = ',D20.9/                                          RBR36500
     +         'E   = ',D20.9/                                          RBR36510
     +         'K   = ',D20.9)                                          RBR36520
 9814 FORMAT (/'Incident Wave at Seaward Boundary Given as Input'/      RBR36530
     +         'Number of Data Points    NDATA =',I8)                   RBR36540
 9815 FORMAT (/'Total Wave at Seaward Boundary Given as Input'/         RBR36550
     +         'Number of Data Points    NDATA =',I8)                   RBR36560
 9816 FORMAT ('Norm. Maximum Surface Elev.     = ',F14.6/               RBR36570
     +        'Norm. Minimum Surface Elev.     = ',F14.6/)              RBR36580
 9817 FORMAT ('Reference Wave Period           = ',F14.6,' sec.'/       RBR36590
     +         'Reference Wave Height          = ',F14.6,A7/            RBR36600
     +         'Depth at Seaward Boundary      = ',F14.6,A7/            RBR36610
     +         'Shoal. Coef. at Reference  Ks1 = ',F11.3/               RBR36620
     +         '             at Seaw. Bdr. Ks2 = ',F11.3/               RBR36630
     +         '             Ks = Ks2/Ks1      = ',F11.3)               RBR36640
 9818 FORMAT ('Norm. Depth at Seaw. Boundary  = ',F11.3/                RBR36650
     +         'Normalized Wave Length         = ',F11.3/               RBR36660
     +         '"Sigma"                        = ',F11.3/               RBR36670
     +         'Ursell Number                  = ',F11.3/               RBR36680
     +         'Surf Similarity Parameter      = ',F11.3)               RBR36690
 9819 FORMAT ('Depth at Landward Boundary     = ',F14.6,A7)             RBR36700
C                                                                       RBR36710
C           STRUCTURE PROPERTIES                                        RBR36720
C           ------------------------------------------------------      RBR36730
      WRITE (98,9821) FWP,FW,WTOT,NBSEG                                 RBR36740
      IF (IBOT.EQ.1) THEN                                               RBR36750
        WRITE (98,9822) UL                                              RBR36760
        WRITE (98,9824) (K,WBSEG(K),TBSLOP(K),K=1,NBSEG)                RBR36770
      ELSE                                                              RBR36780
        WRITE (98,9823) UL,UL                                           RBR36790
        WRITE (98,9824) (K,XBSEG(K),ZBSEG(K),K=1,NBSEG+1)               RBR36800
      ENDIF                                                             RBR36810
      WRITE (98,9825)                                                   RBR36820
 9821 FORMAT (/'SLOPE PROPERTIES'//                                     RBR36830
     +         'Friction Factor            = ',F15.6/                   RBR36840
     +         'Norm. Friction Factor      = ',F15.6/                   RBR36850
```

```
     +        'Norm. Horiz. Length of'/                                      RBR36860
     +        '      Computation Domain    = ',F15.6/                        RBR36870
     +        'Number of Segments          = ',I8)                           RBR36880
 9822 FORMAT (32(1H-)/' SEGMENT    WBSEG(I)    TBSLOP(I)'/                   RBR36890
     +                '       I     ',A7/32(1H-))                           RBR36900
 9823 FORMAT (32(1H-)/' SEGMENT    XBSEG(I)     ZBSEG(I)'/                   RBR36910
     +                '       I     ',A7,'     ',A7/32(1H-))                RBR36920
 9824 FORMAT (I8,2F12.6)                                                     RBR36930
 9825 FORMAT (32(1H-))                                                       RBR36940
C                                                                            RBR36950
C           PARAMETERS FOR ARMOR STABILITY AND MOVEMENT                      RBR36960
C           ------------------------------------------------------------     RBR36970
      IF (ISTAB.GT.0) WRITE (98,9831) TANPHI,SG,C2,C3,CD,CL,CM               RBR36980
      IF (ISTAB.EQ.1) WRITE (98,9832) AMAX,AMIN                              RBR36990
      IF (ISTAB.EQ.2) WRITE (98,9833) DAP,UL                                 RBR37000
 9831 FORMAT (/'PARAMETERS FOR ARMOR STABILITY AND MOVEMENT'//               RBR37010
     +        'Armor Friction Factor     = ',F9.3/                          RBR37020
     +        'Specific Gravity          = ',F9.3/                          RBR37030
     +        'Area Coefficient      C2  = ',F9.3/                          RBR37040
     +        'Volume Coefficient    C3  = ',F9.3/                          RBR37050
     +        'Drag Coefficient      CD  = ',F9.3/                          RBR37060
     +        'Lift Coefficient      CL  = ',F9.3/                          RBR37070
     +        'Inertia Coefficient   CM  = ',F9.3)                          RBR37080
 9832 FORMAT ('Norm. Upper and Lower Bounds of du/dt'/                       RBR37090
     +        '                     AMAX  = ',F9.3/                         RBR37100
     +        '                     AMIN  = ',F9.3)                         RBR37110
 9833 FORMAT ('Armor Diameter             = ',F12.6,A7)                      RBR37120
C                                                                            RBR37130
C           COMPUTATION PARAMETERS                                           RBR37140
C           ------------------------------------------------------------     RBR37150
      WRITE (98,9841) JE                                                     RBR37160
      IF (IJOB.LT.3)  WRITE (98,9842) INITS                                  RBR37170
      WRITE (98,9843) X,X1,X2,TMAX,NONEM,NRATE,MSTAT                         RBR37180
      IF (ISTAB.GT.0) WRITE (98,9844) MSTAB                                  RBR37190
      IF (ISAVA.EQ.1) WRITE (98,9845) NTIMES,MSAVA1,MSAVA2                   RBR37200
 9841 FORMAT (/'COMPUTATION PARAMETERS'//                                    RBR37210
     +  'Total Number of Spatial Nodes      JE = ',I8)                      RBR37220
 9842 FORMAT ('Number of Nodes Along Bottom Below SWL'/                      RBR37230
     +  '                                 INITS = ',I8)                     RBR37240
 9843 FORMAT (                                                               RBR37250
     +  'Normalized Delta x          = ',D19.6/                             RBR37260
     +  'Damping Coefficients    x1 = ',F15.3/                              RBR37270
     +  '                        x2 = ',F15.3/                              RBR37280
     +  'Computation Duration  TMAX = ',F18.6,' Wave Periods'/              RBR37290
     +  'Minimum Allowable NONE          NONEM = ',I8/                      RBR37300
     +  'Time Series Stored at Rate NRATE'/                                 RBR37310
     +  '     per Wave Period with       NRATE = ',I8/                      RBR37320
     +  'Statistical Calculations Are Performed Excluding'/                 RBR37330
     +  '     the First MSTAT Computation Units'/                           RBR37340
     +  '     with                       MSTAT = ',I8)                      RBR37350
 9844 FORMAT (                                                               RBR37360
     +  'Computation of Armor Stability or Movement is Performed'/          RBR37370
     +  '     Excluding the First MSTAB Computation Units'/                 RBR37380
     +  '     with                       MSTAB = ',I8)                      RBR37390
 9845 FORMAT (                                                               RBR37400
```

```
     +  'Spatial Variations are Stored NTIMES at Equal'/                RBR37410
     +  '     Intervals per Wave Period from'/                           RBR37420
     +  '     Computation Unit MSAVA1 to MSAVA2,'/                       RBR37430
     +  '     Inclusive, with          NTIMES = ',I8/                    RBR37440
     +  '                              MSAVA1 = ',I8/                    RBR37450
     +  '                              MSAVA2 = ',I8)                    RBR37460
C                                                                       RBR37470
C          NORMALIZED STRUCTURE GEOMETRY                                RBR37480
C          -------------------------------------------------------      RBR37490
C          File 52 = 'OSPACE'                                           RBR37500
C          (XB(j),ZB(j)) = normalized coordinates of the structure      RBR37510
C                          at node j                                    RBR37520
C          ZB negative below SWL                                        RBR37530
C          -------------------------------------------------------      RBR37540
      WRITE (52,9000) JE                                                RBR37550
      WRITE (52,8000) (XB(J),ZB(J),J=1,JE)                              RBR37560
 9000 FORMAT (I8)                                                       RBR37570
 8000 FORMAT (5D15.6)                                                   RBR37580
C                                                                       RBR37590
      WRITE (*,*) 'Time-marching computation begins'                    RBR37600
C                                                                       RBR37610
      RETURN                                                            RBR37620
      END                                                               RBR37630
C                                                                       RBR37640
C -43-------------------  END OF SUBROUTINE DOC1  -------------RBREAK-  RBR37650
C #44#########################  SUBROUTINE DOC2  ################RBREAK#  RBR37660
C                                                                       RBR37670
C     This subroutine stores computed results at designated time        RBR37680
C     during time-marching computation                                  RBR37690
C                                                                       RBR37700
      SUBROUTINE DOC2 (ICALL,ETAI,ETAR,ETAT)                            RBR37710
C                                                                       RBR37720
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR37730
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR37740
      DIMENSION VONE(N5),VTWO(N5)                                       RBR37750
      INTEGER S,SM1,SP1                                                 RBR37760
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR37770
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,        RBR37780
     +                ISAVA,ISAVB,ISAVC                                 RBR37790
      COMMON /CPAR3/  INITS,JE,JE1                                      RBR37800
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR37810
      COMMON /REQ1/   IREQ,IELEV,IV,IDUDT,ISNR,NREQ                     RBR37820
      COMMON /REQ2/   TREQ(N5)                                          RBR37830
      COMMON /WAVE3/  ETA(N2),ETAIS(N2),ETARS(N2),ETATS(N2)             RBR37840
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             RBR37850
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR37860
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3),RUNUPS(N3),RSTAT(3,N3),      RBR37870
     +                RMEAN(N3)                                         RBR37880
      COMMON /ARMOR5/ CSTAB1,CSTAB2,AMAXS,AMINS,E2,E3PRE(N1),SNR(N1),   RBR37890
     +                SNSX(N1),TSNSX(N1),ATMIN(4,N1),SNSC,TSNSC         RBR37900
      COMMON /ARMOR6/ NMOVE,NSTOP,ISTATE(N1),NODFI(N1)                  RBR37910
      COMMON /ARMOR7/ CSTAB3,CSTAB4,CM1,DA,SIGDA,WEIG,                  RBR37920
     +                VA(N1),XAA(N1),XA(N1),TDIS(N1)                    RBR37930
      COMMON /SAVBC/  NNODB,NNODC,NODB(N5),NODC(N5)                     RBR37940
C                                                                       RBR37950
```

```
      IF (ICALL.EQ.0) THEN                                              RBR37960
C                                                                       RBR37970
C           CHECKING PARAMETERs                                         RBR37980
C           ------------------------------------------------------      RBR37990
        CALL CKFPAR (44,1,N1,N1R)                                       RBR38000
        CALL CKFPAR (44,2,N2,N2R)                                       RBR38010
        CALL CKFPAR (44,3,N3,N3R)                                       RBR38020
        CALL CKFPAR (44,5,N5,N5R)                                       RBR38030
C                                                                       RBR38040
      ELSEIF (ICALL.EQ.1) THEN                                          RBR38050
C                                                                       RBR38060
C           STORING "A"                                                 RBR38070
C           "A" = spatial variations of hydrodynamic quantities         RBR38080
C           ------------------------------------------------------      RBR38090
C           File 52 = 'OSPACE'                                          RBR38100
C           S = waterline node (IJOB<3) or landward-end node (IJOB=3)   RBR38110
C           TIME = current normalized time                              RBR38120
C           At node j:                                                  RBR38130
C              ELEV(j) = surface elevation above SWL                    RBR38140
C              V(j)    = depth-averaged velocity                        RBR38150
C           ------------------------------------------------------      RBR38160
        WRITE (52,9000) S                                               RBR38170
        WRITE (52,8000) (ELEV(J),V(J),J=1,S),TIME                       RBR38180
C                                                                       RBR38190
      ELSEIF (ICALL.EQ.2) THEN                                          RBR38200
C                                                                       RBR38210
C           STORING "B"                                                 RBR38220
C           "B" = time series of volume flux and total water depth at   RBR38230
C                 specified nodes                                       RBR38240
C           ------------------------------------------------------      RBR38250
C           File 71 ('OSAV1Bxx') stores volume flux                     RBR38260
C           File 72 ('OSAV2Bxx') stores total water depth               RBR38270
C           where xx=01 contains the first 100 waves                    RBR38280
C                 xx=02 contains the second 100 waves                   RBR38290
C                 and so on                                             RBR38300
C           ------------------------------------------------------      RBR38310
        IF (ISAVB.EQ.1) THEN                                            RBR38320
          DO 110 I = 1,NNODB                                            RBR38330
            J = NODB(I)                                                 RBR38340
            VONE(I) = U(1,J)                                            RBR38350
            VTWO(I) = U(2,J)                                            RBR38360
  110     CONTINUE                                                      RBR38370
          WRITE (71,8000) (VONE(I),I=1,NNODB),TIME                      RBR38380
          WRITE (72,8000) (VTWO(I),I=1,NNODB)                           RBR38390
        ENDIF                                                           RBR38400
C                                                                       RBR38410
C           STORING "C"                                                 RBR38420
C           "C" = time series of X, where X is                          RBR38430
C               . for ISTAB=1: stability number SNR at specified nodes  RBR38440
C               . for ISTAB=2: displacement of armor units from         RBR38450
C                              specified initial nodal locations        RBR38460
C           ------------------------------------------------------      RBR38470
C           File 73 = 'OSAVECxx'                                        RBR38480
C           where xx=01 contains the first 100 waves                    RBR38490
C                 xx=02 contains the second 100 waves                   RBR38500
```

```
C           ------------------------------------------------------------  RBR38510
          IF (ISAVC.EQ.1) THEN                                            RBR38520
            IF (ISTAB.EQ.1) THEN                                          RBR38530
              DO 120 I = 1,NNODC                                          RBR38540
                J = NODC(I)                                               RBR38550
                VONE(I) = SNR(J)                                          RBR38560
  120         CONTINUE                                                    RBR38570
            ELSE                                                          RBR38580
              DO 130 I = 1,NNODC                                          RBR38590
                J = NODC(I)                                               RBR38600
                IF (ISTATE(J).EQ.0) THEN                                  RBR38610
                  VONE(I) = 0.D+00                                        RBR38620
                ELSE                                                      RBR38630
                  VONE(I) = XA(J)                                         RBR38640
                ENDIF                                                     RBR38650
  130         CONTINUE                                                    RBR38660
            ENDIF                                                         RBR38670
            WRITE (73,8000) (VONE(I),I=1,NNODC),TIME                      RBR38680
          ENDIF                                                           RBR38690
C                                                                         RBR38700
C           STORING VALUES AT LANDWARD-END NODE                           RBR38710
C           ------------------------------------------------------------  RBR38720
C           File 62 = 'ORUNUP'                                            RBR38730
C           File 63 = 'OOVER'                                             RBR38740
C           File 64 = 'OTRANS'                                            RBR38750
C           JE = landward-end node                                        RBR38760
C           S  = waterline node                                           RBR38770
C           RUNUPS = free surface elevation where the water depth         RBR38780
C                    equals DELTAR                                        RBR38790
C           DELTAR = water depth associated with visual or measured       RBR38800
C                    waterline                                            RBR38810
C           NDELR  = number of DELTARs                                    RBR38820
C           ETAT   = surface elevation associated with transmitted        RBR38830
C                    wave at landward boundary                            RBR38840
C           At node j:                                                    RBR38850
C              U(1,j) = volume flux                                       RBR38860
C              U(2,j) = total water depth                                 RBR38870
C           ------------------------------------------------------------  RBR38880
          IF (IJOB.LT.3) THEN                                             RBR38890
            DBLES=DBLE(S)                                                 RBR38900
            WRITE (62,8000) DBLES,(RUNUPS(L),L=1,NDELR),TIME              RBR38910
            IF (IJOB.EQ.2) WRITE (63,8000) U(1,JE),U(2,JE)                RBR38920
          ELSE                                                            RBR38930
            WRITE (64,8000) U(1,JE),U(2,JE),ETAT,TIME                     RBR38940
          ENDIF                                                           RBR38950
C                                                                         RBR38960
C           STORING VALUES AT SEAWARD BOUNDARY                            RBR38970
C           ------------------------------------------------------------  RBR38980
C           File 61 = 'OSEAWAV'                                           RBR38990
C           TIME = current normalized time                                RBR39000
C           Surface elevations at seaward boundary:                       RBR39010
C           . ETAI --> due to incident wave                               RBR39020
C           . ETAR --> due to reflected wave                              RBR39030
C           U(1,1) = volume flux at node 1                                RBR39040
C           ------------------------------------------------------------  RBR39050
```

```
        IF (IWAVE.EQ.3) ETAI=ETAI-ETAR                                  RBR39060
        WRITE (61,8000) TIME,ETAI,ETAR,U(1,1)                           RBR39070
C                                                                       RBR39080
      ELSE                                                              RBR39090
C                                                                       RBR39100
C           SPECIAL STORING IF ICALL=3                                  RBR39110
C           ------------------------------------------------------      RBR39120
C           File 55 = 'OREQ'                                            RBR39130
C           S = waterline node (IJOB<3) or landward-end node (IJOB=3)   RBR39140
C           TIME = current normalized time                              RBR39150
C           At node j for specified time:                               RBR39160
C              ELEV(j) = surface elevation above SWL                    RBR39170
C              V(j)    = depth-averaged velocity                        RBR39180
C              DUDT(j) = total fluid acceleration                       RBR39190
C              SNR(j)  = stability number against rolling/sliding       RBR39200
C           ------------------------------------------------------      RBR39210
        WRITE (55,9000) S                                               RBR39220
        WRITE (55,8000) TIME                                            RBR39230
        IF (IELEV.EQ.1) WRITE (55,8000) (ELEV(J),J=1,S)                 RBR39240
        IF (IV.EQ.1)    WRITE (55,8000) (V(J),J=1,S)                    RBR39250
        IF (IDUDT.EQ.1) WRITE (55,8000) (DUDT(J),J=1,S)                 RBR39260
        IF (ISNR.EQ.1)  WRITE (55,8000) (SNR(J),J=1,S)                  RBR39270
C                                                                       RBR39280
      ENDIF                                                             RBR39290
 9000 FORMAT (2I8)                                                      RBR39300
 8000 FORMAT (5D15.6)                                                   RBR39310
C                                                                       RBR39320
      RETURN                                                            RBR39330
      END                                                               RBR39340
C                                                                       RBR39350
C -44------------------  END OF SUBROUTINE DOC2  -------------RBREAK-   RBR39360
C #45#####################  SUBROUTINE DOC3  ################RBREAK#   RBR39370
C                                                                       RBR39380
C     This subroutine documents essential results after time-marching   RBR39390
C     computation                                                       RBR39400
C                                                                       RBR39410
      SUBROUTINE DOC3 (MODE)                                            RBR39420
C                                                                       RBR39430
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               RBR39440
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40)                      RBR39450
      CHARACTER*7 UL                                                    RBR39460
      INTEGER S,SM1,SP1                                                 RBR39470
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                               RBR39480
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,        RBR39490
     +                ISAVA,ISAVB,ISAVC                                 RBR39500
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES            RBR39510
      COMMON /CPAR3/  INITS,JE,JE1                                      RBR39520
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     RBR39530
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     RBR39540
      COMMON /COEFS/  RCOEF(3),TCOEF(3)                                 RBR39550
      COMMON /STT1/   ELSTAT(3,3),U1STAT(N1),ESTAT(3,N1),VSTAT(3,N1),   RBR39560
     +                ELMEAN(3),U1MEAN(N1),EMEAN(N1),VMEAN(N1)          RBR39570
      COMMON /STT2/   TSTAT1,TSTAT2                                     RBR39580
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              RBR39590
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3),RUNUPS(N3),RSTAT(3,N3),      RBR39600
```

```
     +                RMEAN(N3)                                              RBR39610
      COMMON /OVER/   OV(4),OVMEAN                                           RBR39620
      COMMON /ENERG/  ENER(7,N1),ENERB(15),ENTEMP(3,N1)                      RBR39630
      COMMON /ARMOR3/ JSTAB,JSTABM                                           RBR39640
      COMMON /ARMOR4/ JSNSC,JATMIN                                           RBR39650
      COMMON /ARMOR5/ CSTAB1,CSTAB2,AMAXS,AMINS,E2,E3PRE(N1),SNR(N1),        RBR39660
     +                SNSX(N1),TSNSX(N1),ATMIN(4,N1),SNSC,TSNSC              RBR39670
      COMMON /ARMOR6/ NMOVE,NSTOP,ISTATE(N1),NODFI(N1)                       RBR39680
      COMMON /ARMOR7/ CSTAB3,CSTAB4,CM1,DA,SIGDA,WEIG,                       RBR39690
     +                VA(N1),XAA(N1),XA(N1),TDIS(N1)                         RBR39700
      IF (MODE.EQ.0) THEN                                                    RBR39710
        CALL CKFPAR (45,1,N1,N1R)                                            RBR39720
        CALL CKFPAR (45,3,N3,N3R)                                            RBR39730
        RETURN                                                               RBR39740
      ENDIF                                                                  RBR39750
C                                                                            RBR39760
C           SYSTEM OF UNITS                                                  RBR39770
C           ---------------------------------------------------------        RBR39780
      IF (ISYST.EQ.1) THEN                                                   RBR39790
        UL = '    [mm]'                                                      RBR39800
      ELSE                                                                   RBR39810
        UL = ' [inch]'                                                       RBR39820
      ENDIF                                                                  RBR39830
C                                                                            RBR39840
C           REFLECTION COEFFICIENTS                                          RBR39850
C           ---------------------------------------------------------        RBR39860
      IF (IWAVE.EQ.1) THEN                                                   RBR39870
        WRITE (98,9811) (RCOEF(I),I=1,3)                                     RBR39880
      ELSE                                                                   RBR39890
        WRITE (98,9812) RCOEF(1)                                             RBR39900
      ENDIF                                                                  RBR39910
 9811 FORMAT (/'REFLECTION COEFFICIENTS'//                                   RBR39920
     +         'r1 = ',F9.3/                                                 RBR39930
     +         'r2 = ',F9.3/                                                 RBR39940
     +         'r3 = ',F9.3)                                                 RBR39950
 9812 FORMAT (/'REFLECTION COEFFICIENT'//                                    RBR39960
     +         'r1 = ',F9.3)                                                 RBR39970
C                                                                            RBR39980
C           RUNUP, RUNDOWN, SETUP                                            RBR39990
C           ---------------------------------------------------------        RBR40000
      IF (IJOB.LT.3) THEN                                                    RBR40010
        WRITE (98,9821) JMAX                                                 RBR40020
        WRITE (98,9822) UL                                                   RBR40030
        DO 110 L = 1,NDELR                                                   RBR40040
          WRITE (98,9823) L,DELRP(L),RSTAT(2,L),RSTAT(3,L),RSTAT(1,L)        RBR40050
  110   CONTINUE                                                             RBR40060
        WRITE (98,9824)                                                      RBR40070
      ENDIF                                                                  RBR40080
 9821 FORMAT (/'RUNUP, RUNDOWN, SETUP'//                                     RBR40090
     +'Largest Node Number Reached by Computational Waterline'/              RBR40100
     +'                                          JMAX = ',I8)               RBR40110
 9822 FORMAT (56(1H-)/                                                       RBR40120
     +'       I   DELTAR(I)     RUNUP(I)    RUNDOWN(I)    SETUP(I)'/       RBR40130
     +'           ',A7,'          R            Rd            Zr'/56(1H-))   RBR40140
 9823 FORMAT (I8,1X,F9.3,3(3X,F9.3))                                         RBR40150
```

```
 9824 FORMAT (56(1H-))                                                  RBR40160
C                                                                       RBR40170
C           OVERTOPPING                                                 RBR40180
C           ------------------------------------------------------      RBR40190
      TSPAN = TMAX-DBLE(MSTAT)                                          RBR40200
      IF (IJOB.EQ.2) WRITE (98,9831)                                    RBR40210
     +               OV(1),U1STAT(1),OV(2),OV(4),OV(3),TSPAN,TMAX       RBR40220
 9831 FORMAT (/'OVERTOPPING'//                                          RBR40230
     +'Norm. Average Overtopping Rate          = ',D15.6/               RBR40240
     +'Norm. Average Flow at Seaward Boundary = ',D15.6/                RBR40250
     +'Norm. Maximum Overtopping Rate          = ',D15.6/               RBR40260
     +'      Occuring at Normalized Time    t = ',F18.6/               RBR40270
     +'Norm. Overtopping Duration              = ',F18.6/               RBR40280
     +'Notes: Norm. duration of overtopping computation =',F18.6/       RBR40290
     +'       Norm. duration of RBREAK computation      =',F18.6)       RBR40300
C                                                                       RBR40310
C           WAVE SET-DOWN OR SETUP                                      RBR40320
C           ------------------------------------------------------      RBR40330
      IF (IJOB.EQ.3) THEN                                               RBR40340
        DELMWL = ELSTAT(1,3) - ELSTAT(1,2)                              RBR40350
        WRITE (98,9841) (ELSTAT(1,I),I=1,3),DELMWL                      RBR40360
      ELSE                                                              RBR40370
        WRITE (98,9842) (ELSTAT(1,I),I=1,2)                             RBR40380
      ENDIF                                                             RBR40390
 9841 FORMAT (/'WAVE SET-DOWN OR SETUP'//                               RBR40400
     +         'Average value of ETAI = ',F12.6/                        RBR40410
     +         '                 ETAR = ',F12.6/                        RBR40420
     +         '                 ETAT = ',F12.6/                        RBR40430
     +         'MWL Difference        = ',F12.6)                       RBR40440
 9842 FORMAT (/'WAVE SET-DOWN OR SETUP'//                               RBR40450
     +         'Average value of ETAI = ',F12.6/                        RBR40460
     +         '                 ETAR = ',F12.6)                        RBR40470
C                                                                       RBR40480
C           TRANSMISSION                                                RBR40490
C           ------------------------------------------------------      RBR40500
      IF (IJOB.EQ.3) THEN                                               RBR40510
        QAVER = .5*(U1STAT(1)+U1STAT(JE))                               RBR40520
        IF (IWAVE.EQ.1) THEN                                            RBR40530
          WRITE (98,9851) (TCOEF(I),I=1,3)                              RBR40540
        ELSE                                                            RBR40550
          WRITE (98,9852) TCOEF(1)                                      RBR40560
        ENDIF                                                           RBR40570
        WRITE (98,9853) U1STAT(1),U1STAT(JE),QAVER                      RBR40580
      ENDIF                                                             RBR40590
 9851 FORMAT (/'TRANSMISSION'//                                         RBR40600
     +         'Transmission Coefficient T1 = ',F9.3/                   RBR40610
     +         '                         T2 = ',F9.3/                   RBR40620
     +         '                         T3 = ',F9.3)                   RBR40630
 9852 FORMAT (/'TRANSMISSION'//                                         RBR40640
     +         'Transmission Coefficient T1 = ',F9.3)                   RBR40650
 9853 FORMAT ('Norm. Average Flow at Seaward Boundary  = ',F18.9/       RBR40660
     +         'Norm. Average Flow at Landward Boundary = ',F18.9/      RBR40670
     +         'Average of the Above Two                = ',F18.9)      RBR40680
C                                                                       RBR40690
C           STATISTICS OF HYDRODYNAMIC QUANTITIES                       RBR40700
```

```
C           ------------------------------------------------------------  RBR40710
C           File 51 = 'OSTAT'                                             RBR40720
C           JMAX = the largest node number reached by computational       RBR40730
C              waterline                                                  RBR40740
C           At node j:                                                    RBR40750
C              ELEV(j)    = surface elevation above SWL                   RBR40760
C              V(j)       = depth-averaged velocity                       RBR40770
C              U1STAT(j) = mean volume flux                               RBR40780
C           Mean, maximum, and minimum at node j:                         RBR40790
C              ESTAT(1,j),ESTAT(2,j),ESTAT(3,j): for ELEV(j)              RBR40800
C              VSTAT(1,j),VSTAT(2,j),VSTAT(3,j): for V(j)                 RBR40810
C           ------------------------------------------------------------  RBR40820
      WRITE (51,9000) JMAX                                                RBR40830
      WRITE (51,8000) (U1STAT(J),J=1,JMAX)                                RBR40840
      DO 120 I = 1,3                                                      RBR40850
        WRITE (51,8000) (ESTAT(I,J),J=1,JMAX)                             RBR40860
        WRITE (51,8000) (VSTAT(I,J),J=1,JMAX)                             RBR40870
  120 CONTINUE                                                            RBR40880
C                                                                         RBR40890
C           ENERGY QUANTITIES                                             RBR40900
C           ------------------------------------------------------------  RBR40910
C           File 53 = 'OENERG'                                            RBR40920
C           At node j:                                                    RBR40930
C              ENER(1,j) = norm. energy per unit surface area             RBR40940
C              ENER(2,j) = norm. energy flux per unit width               RBR40950
C           Normalized rate of energy dissipation at node j:              RBR40960
C              ENER(3,j): due to bottom friction, per unit bottom area    RBR40970
C              ENER(4,j): due to wave breaking, per unit surface area     RBR40980
C           ------------------------------------------------------------  RBR40990
      IF (IENERG.EQ.1) THEN                                               RBR41000
        WRITE (98,9861)                                                   RBR41010
        WRITE (98,9862) (ENERB(I),I=1,9)                                  RBR41020
        IF (IWAVE.EQ.1) THEN                                              RBR41030
          WRITE (98,9863) (ENERB(I),I=10,11)                              RBR41040
          IF (IJOB.EQ.3)  WRITE (98,9864) ENERB(12)                       RBR41050
          WRITE (98,9865) (ENERB(I),I=13,14)                              RBR41060
          IF (IJOB.EQ.3)  WRITE (98,9866) ENERB(15)                       RBR41070
        ENDIF                                                             RBR41080
        WRITE (53,9000) JMAX                                              RBR41090
        DO 130 I = 1,7                                                    RBR41100
          WRITE (53,8000) (ENER(I,J),J=1,JMAX)                            RBR41110
  130   CONTINUE                                                          RBR41120
      ENDIF                                                               RBR41130
 9861 FORMAT (/'TIME-AVERAGED ENERGY BALANCE'//                           RBR41140
     +        'Normalized Energy Flux:')                                  RBR41150
 9862 FORMAT ('. at Seaw. Boundary   A =',D14.6/                          RBR41160
     +        '. at Landw. Boundary  B =',D14.6/                          RBR41170
     +        'Normalized Rate of Energy Dissipation'/                    RBR41180
     +        'in the Computation Domain, Due to:'/                       RBR41190
     +        '. Bottom Friction     C =',D14.6/                          RBR41200
     +        '. Wave Breaking       D =',D14.6/                          RBR41210
     +        'Additional Term       E =',D14.6/                          RBR41220
     +        'Calculation 1:'/                                           RBR41230
     +        '                F = A-B =',D14.6/                          RBR41240
     +        '              G = C+D+E =',D14.6/                          RBR41250
```

```
     +        '  Must H=0, but H = G-F =',D14.6/                          RBR41260
     +        '  % error  100G/E =',F14.2)                                RBR41270
 9863 FORMAT ('Approximate Energy Flux, Based on'/                        RBR41280
     +        'Linear Long Wave, Due to:'/                                RBR41290
     +        '. Incident Wave at Seaw. Boundary     P =',D14.6/          RBR41300
     +        '. Reflected Wave at Seaw. Boundary    Q =',D14.6)          RBR41310
 9864 FORMAT ('. Transmitted Wave at Landw. Bndry.   R =',D14.6)          RBR41320
 9865 FORMAT ('Calculation 2:'/                                           RBR41330
     +        'Net Energy Flux at Seaw. Bndry. S = P-Q =',D14.6/          RBR41340
     +        '% Error at Seaward Boundary  100(S-A)/A =',F14.2)          RBR41350
 9866 FORMAT ('% Error at Landward Boundary 100(R-B)/B =',F14.2)          RBR41360
C                                                                         RBR41370
C          ARMOR STABILITY OR MOVEMENT                                    RBR41380
C          ---------------------------------------------------------      RBR41390
C          File 54 = 'OSTAB'                                              RBR41400
C          (XB(j),ZB(j)) = normalized coordinates of the structure at     RBR41410
C                          node j                                         RBR41420
C          ISTAB=1:                                                       RBR41430
C            SNSX(j) = local stability number                             RBR41440
C                    = minimum of SNR at a node j                         RBR41450
C            SNR(j)  = stability number against rolling/sliding at        RBR41460
C                      node j                                             RBR41470
C          ISTAB=2:                                                       RBR41480
C            ISTATE(j) indicates the state of armor unit initially        RBR41490
C              located at node j: 0=stationary, 1=moving, 2=stopped       RBR41500
C            For moving/stopped armor unit number j:                      RBR41510
C              NODFI(j) = node closest to its final location              RBR41520
C              TDIS(j)  = time when it started moving                     RBR41530
C              XA(j)    = displacement from its initial location,         RBR41540
C                         normalized by DAP                               RBR41550
C          ---------------------------------------------------------      RBR41560
      IF (ISTAB.EQ.1) THEN                                                RBR41570
        WRITE (98,9871) JSTABM                                            RBR41580
        WRITE (98,9872) SNSC,JSNSC,TSNSC                                  RBR41590
        WRITE (54,9000) JSTABM                                            RBR41600
        DO 140 J = 1,JSTABM                                               RBR41610
          WRITE (54,8000) XB(J),ZB(J),SNSX(J),TSNSX(J)                    RBR41620
  140   CONTINUE                                                          RBR41630
        WRITE (54,9000) JATMIN                                            RBR41640
        WRITE (54,8000) TSNSC                                             RBR41650
        DO 150 I = 1,4                                                    RBR41660
          WRITE (54,8000) (ATMIN(I,J),J=1,JATMIN)                         RBR41670
  150   CONTINUE                                                          RBR41680
      ELSEIF (ISTAB.EQ.2) THEN                                            RBR41690
        WRITE (98,9873) JSTABM,NMOVE,NSTOP                                RBR41700
        WRITE (54,9000) NMOVE                                             RBR41710
        WRITE (54,9000) NSTOP                                             RBR41720
        WRITE (54,9000) JSTABM                                            RBR41730
        DO 160 J = 1,JSTABM                                               RBR41740
          IF (ISTATE(J).GE.1) THEN                                        RBR41750
            WRITE (54,9000) J,NODFI(J),ISTATE(J)                          RBR41760
            WRITE (54,8000) XB(J),ZB(J),XA(J),TDIS(J)                     RBR41770
          ENDIF                                                           RBR41780
  160   CONTINUE                                                          RBR41790
      ENDIF                                                               RBR41800
```

```
 9871 FORMAT (/'STABILITY NUMBER'//                                      RBR41810
     +         'Largest Node Number for Which Armor Stability'/          RBR41820
     +         '  Computation Performed  JSTABM = ',I11)                 RBR41830
 9872 FORMAT ( 'Critical Stability Number   Nsc = ',F11.3/               RBR41840
     +         'At Node Number                  = ',I11/                 RBR41850
     +         'At Time                         = ',F18.6)               RBR41860
 9873 FORMAT (/'ARMOR UNITS MOVEMENT'//                                  RBR41870
     +         'Largest Node Number for Which Armor Movement'/           RBR41880
     +         '  Computation Performed  JSTABM = ',I8/                  RBR41890
     +         'Number of Units Moved     NMOVE = ',I8/                 RBR41900
     +         'Number of Units Stopped   NSTOP = ',I8)                 RBR41910
C                                                                        RBR41920
 9000 FORMAT (8I8)                                                       RBR41930
 8000 FORMAT (5D15.6)                                                    RBR41940
C                                                                        RBR41950
      RETURN                                                             RBR41960
      END                                                                RBR41970
C                                                                        RBR41980
C -45------------------  END OF SUBROUTINE DOC3  ------------RBREAK-     RBR41990
C #46######################  SUBROUTINE CKFPAR  ##############RBREAK#   RBR42000
C                                                                        RBR42010
C     This subroutine checks if FORTRAN PARAMETER NCHEK=N1,N2,N3,N4,N5   RBR42020
C     specified in given Subroutine (ICALL) matches its counterpart      RBR42030
C     NREF=N1R,N2R,N3R,N4R,N5R from the main program                     RBR42040
C                                                                        RBR42050
      SUBROUTINE CKFPAR (ICALL,NW,NCHEK,NREF)                            RBR42060
C                                                                        RBR42070
      CHARACTER*2 WHICH(5)                                               RBR42080
      CHARACTER*6 SUBR(50)                                               RBR42090
      DATA WHICH /'N1','N2','N3','N4','N5'/                              RBR42100
      DATA SUBR  /'OPENIO','INPUT1','INPUT2','BOTTOM','WPARAM',          RBR42110
     2            'REGWAV','FINDM ','CEL   ','SNCNDN','CPARAM',        RBR42120
     3            'NUMSTA','EXTRAP','RETAIN','CRITV ','MARCH ',        RBR42130
     4            'LANDBC','RUNUP ','OVERT ','SEABC ','MATAFG',        RBR42140
     5            'MATGJR','MATS  ','MATD  ','MATU  ','NSI   ',        RBR42150
     6            'INIT  ','INITH ','INSTAT','SVSTAT','ENERGY',        RBR42160
     7            'STABNO','MOVE  ','FORCES','ACCEL ','STAT1 ',        RBR42170
     8            'STAT2 ','STATC ','COEF  ','ENERGC','SPARAM',        RBR42180
     9            'VECMAT','DERIV ','DOC1  ','DOC2  ','DOC3  ',        RBR42190
     +            'CKFPAR','CKVAL ','CKSUBR','STOPP ','OPENF '/        RBR42200
       IF (NCHEK.NE.NREF) THEN                                          RBR42210
         WRITE  (*,9910)                                                RBR42220
     +          WHICH(NW),NCHEK,ICALL,SUBR(ICALL),WHICH(NW),NREF        RBR42230
         WRITE (99,9910)                                                RBR42240
     +          WHICH(NW),NCHEK,ICALL,SUBR(ICALL),WHICH(NW),NREF        RBR42250
         STOP                                                           RBR42260
       ENDIF                                                            RBR42270
 9910 FORMAT (/                                                         RBR42280
     +' PARAMETER Error: ',A2,' =',I8,' in Subroutine',I3,' ',A6/       RBR42290
     +' Correct Value:   ',A2,' =',I8)                                  RBR42300
C                                                                        RBR42310
      RETURN                                                             RBR42320
      END                                                                RBR42330
C                                                                        RBR42340
C -46------------------  END OF SUBROUTINE CKFPAR  ------------RBREAK-   RBR42350
```

```
C #47#######################  SUBROUTINE CKVAL  ################RBREAK#  RBR42360
C                                                                         RBR42370
C     This subroutine checks user-specified values                        RBR42380
C                                                                         RBR42390
      SUBROUTINE CKVAL (IC,ITEM,IUP)                                      RBR42400
C                                                                         RBR42410
      CHARACTER*2 WHICH(7)                                                RBR42420
      CHARACTER*6 WHAT(9)                                                 RBR42430
      DATA WHICH /'N5','N5','N5','N3','N1','N4','N2'/                     RBR42440
      DATA WHAT  /'NREQ  ','NNODB ','NNODC ','NDELR ','INITS ',           RBR42450
     +            'NBSEG ','NDATA ','NODB  ','NODC  '/                    RBR42460
      IF (IC.LE.7) THEN                                                   RBR42470
        IF (ITEM.LT.1.OR.ITEM.GT.IUP) THEN                                RBR42480
          WRITE  (*,9910) WHAT(IC),ITEM,WHAT(IC),IUP,WHICH(IC)            RBR42490
          WRITE (99,9910) WHAT(IC),ITEM,WHAT(IC),IUP,WHICH(IC)            RBR42500
          STOP                                                            RBR42510
        ENDIF                                                             RBR42520
      ELSE                                                                RBR42530
        IF (ITEM.LT.1.OR.ITEM.GT.IUP) THEN                                RBR42540
          WRITE  (*,9920) WHAT(IC),ITEM,WHAT(IC),IUP                      RBR42550
          WRITE (99,9920) WHAT(IC),ITEM,WHAT(IC),IUP                      RBR42560
          STOP                                                            RBR42570
        ENDIF                                                             RBR42580
      ENDIF                                                               RBR42590
 9910 FORMAT (/' Input Error: ',A6,'=',I8/                                RBR42600
     +         ' Specify ',A6,' in the range of [1,',I8,']'/              RBR42610
     +         ' Change PARAMETER ',A2,' if necessary')                   RBR42620
 9920 FORMAT (/' Input Error: ',A6,'=',I8/                                RBR42630
     +         ' Specify ',A6,' in the range of [1,',I8,']')              RBR42640
C                                                                         RBR42650
      RETURN                                                              RBR42660
      END                                                                 RBR42670
C                                                                         RBR42680
C -47-------------------  END OF SUBROUTINE CKVAL  ------------RBREAK-    RBR42690
C #48#######################  SUBROUTINE CKSUBR  ################RBREAK#  RBR42700
C                                                                         RBR42710
C     This subroutine checks the values of the FORTRAN PARAMETERs         RBR42720
C     specified in subroutines after reading input data file              RBR42730
C                                                                         RBR42740
      SUBROUTINE CKSUBR (N1,N2,N3,N4,N5)                                  RBR42750
C                                                                         RBR42760
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                 RBR42770
      CHARACTER*10 CDUM                                                   RBR42780
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R                                 RBR42790
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IENERG,IWAVE,          RBR42800
     +                ISAVA,ISAVB,ISAVC                                   RBR42810
      DATA IDUM /0/                                                       RBR42820
      DATA DDUM /0.D+00/                                                  RBR42830
      DATA CDUM /'Tcharacter'/                                            RBR42840
C                                                                         RBR42850
C           DUMMIES FOR FORTRAN PARAMETERS USED IN THIS PROGRAM           RBR42860
C           ---------------------------------------------------           RBR42870
C           Variables specified in PARAMETER statement cannot be          RBR42880
C           passed through COMMON statement. The following dummy          RBR42890
C           integers are used in COMMON /DIMENS/.                         RBR42900
```

```
C           ------------------------------------------------------------  RBR42910
      N1R = N1                                                            RBR42920
      N2R = N2                                                            RBR42930
      N3R = N3                                                            RBR42940
      N4R = N4                                                            RBR42950
      N5R = N5                                                            RBR42960
C                                                                         RBR42970
C           CHECK ALL SUBROUTINES POSSESSING FORTRAN "PARAMETER"          RBR42980
C           STATEMENT                                                     RBR42990
C           Checking is indicated by MODE=0                               RBR43000
C           ------------------------------------------------------------  RBR43010
      MODE=0                                                              RBR43020
      CALL INPUT1 (MODE,IDUM,CDUM)                                        RBR43030
      CALL BOTTOM (MODE)                                                  RBR43040
      CALL WPARAM (MODE)                                                  RBR43050
      CALL NUMSTA (MODE,IDUM)                                             RBR43060
      CALL RETAIN (MODE)                                                  RBR43070
      CALL CRITV  (MODE,IDUM)                                             RBR43080
      CALL MARCH  (MODE,IDUM)                                             RBR43090
      CALL LANDBC (MODE,IDUM,IDUM,DDUM)                                   RBR43100
      CALL SEABC  (MODE,IDUM,IDUM,DDUM,DDUM)                              RBR43110
      CALL MATAFG (MODE,IDUM,IDUM)                                        RBR43120
      CALL MATGJR (MODE,IDUM,IDUM)                                        RBR43130
      CALL MATS   (MODE,IDUM,IDUM)                                        RBR43140
      CALL MATD   (MODE,IDUM,IDUM)                                        RBR43150
      CALL MATU   (MODE,IDUM,IDUM)                                        RBR43160
      CALL INIT   (MODE)                                                  RBR43170
      CALL INSTAT (MODE)                                                  RBR43180
      CALL SVSTAT (MODE)                                                  RBR43190
      CALL STATC  (MODE)                                                  RBR43200
      CALL DOC1   (MODE)                                                  RBR43210
      CALL DOC2   (MODE,DDUM,DDUM,DDUM)                                   RBR43220
      CALL DOC3   (MODE)                                                  RBR43230
      IF (IJOB.LT.3) THEN                                                 RBR43240
        CALL EXTRAP (MODE)                                                RBR43250
        CALL RUNUP  (MODE)                                                RBR43260
        IF (IJOB.EQ.2) CALL OVERT (MODE)                                  RBR43270
      ENDIF                                                               RBR43280
      IF (IWAVE.EQ.1) THEN                                                RBR43290
        CALL REGWAV (MODE)                                                RBR43300
      ELSE                                                                RBR43310
        CALL INPUT2 (MODE)                                                RBR43320
      ENDIF                                                               RBR43330
      IF (IENERG.EQ.1) THEN                                               RBR43340
        CALL ENERGY (MODE,IDUM)                                           RBR43350
        CALL ENERGC (MODE)                                                RBR43360
      ENDIF                                                               RBR43370
      IF (ISTAB.GT.0) THEN                                                RBR43380
        CALL SPARAM (MODE)                                                RBR43390
        CALL ACCEL  (MODE)                                                RBR43400
        IF (ISTAB.EQ.1) THEN                                              RBR43410
          CALL STABNO (MODE)                                              RBR43420
        ELSE                                                              RBR43430
          CALL MOVE   (MODE)                                              RBR43440
          CALL FORCES (MODE,DDUM,IDUM,DDUM,DDUM)                          RBR43450
```

```
        ENDIF                                                                RBR43460
      ENDIF                                                                  RBR43470
C                                                                            RBR43480
      RETURN                                                                 RBR43490
      END                                                                    RBR43500
C                                                                            RBR43510
C -48------------------  END OF SUBROUTINE CKSUBR  ------------RBREAK-      RBR43520
C #49#####################  SUBROUTINE STOPP  ################RBREAK#      RBR43530
C                                                                            RBR43540
C     This subroutine executes a programmed stop                             RBR43550
C                                                                            RBR43560
      SUBROUTINE STOPP (IBEGIN,IEND)                                         RBR43570
C                                                                            RBR43580
      CHARACTER*55 MSG(14)                                                   RBR43590
      DATA MSG /                                                             RBR43600
     1 ' MREP must be positive or zero.',                                    RBR43610
     2 ' Computation unit not matched. Check BINPUT.',                       RBR43620
     3 ' Computation duration does not match. Check BINPUT.',                RBR43630
     4 ' Second input file name does not match. Check BINPUT.',              RBR43640
     5 ' SWL is always above the structure.',                                RBR43650
     6 ' RUNUP/OVERTOPPING computation can not be performed.',               RBR43660
     7 ' Part of the structure is above SWL.',                               RBR43670
     8 ' TRANSMISSION computation can not be performed.',                    RBR43680
     9 ' Failure in Subr. 09 SNCNDN.',                                       RBR43690
     + ' IPROB nonzero.',                                                    RBR43700
     1 ' Was corresponding BEFORR computation okay?',                        RBR43710
     2 ' Number of data points for input time series',                       RBR43720
     3 ' does not match. Check BINPUT.',                                     RBR43730
     4 ' Minimum allowable NONE does not match. Check BINPUT.'/              RBR43740
      DO 100 I = IBEGIN,IEND                                                 RBR43750
        WRITE  (*,9910) MSG(I)                                               RBR43760
        WRITE (99,9910) MSG(I)                                               RBR43770
  100 CONTINUE                                                               RBR43780
      WRITE  (*,9920)                                                        RBR43790
      WRITE (99,9920)                                                        RBR43800
 9910 FORMAT (/A55)                                                          RBR43810
 9920 FORMAT (' Programmed Stop.')                                           RBR43820
C                                                                            RBR43830
      STOP                                                                   RBR43840
      END                                                                    RBR43850
C                                                                            RBR43860
C -49------------------  END OF SUBROUTINE STOPP  ------------RBREAK-      RBR43870
C #50#####################  SUBROUTINE OPENF  ################RBREAK#      RBR43880
C                                                                            RBR43890
C     This subroutine opens a specified file                                 RBR43900
C                                                                            RBR43910
      SUBROUTINE OPENF (NUNIT,FNAME,FSTAT)                                   RBR43920
      CHARACTER FNAME*10,FSTAT*3                                             RBR43930
      OPEN (UNIT=NUNIT,FILE=FNAME,STATUS=FSTAT,ACCESS='SEQUENTIAL')          RBR43940
      RETURN                                                                 RBR43950
      END                                                                    RBR43960
C                                                                            RBR43970
C -50------------------  END OF SUBROUTINE OPENF  ------------RBREAK-      RBR43980
```

Appendix B

Listing of Computer Program BEFORR

```
C                                                                       BEF00010
C      #######   ########  ########  #######   #######   #######      BEF00020
C      ##    ##  ##        ##        ##     ##  ##    ##  ##    ##     BEF00030
C      ##    ##  ##        ##        ##     ##  ##    ##  ##    ##     BEF00040
C      #######   #######   #######   ##     ##  #######   #######      BEF00050
C      ##    ##  ##        ##        ##     ##  ##  ##    ##  ##       BEF00060
C      ##    ##  ##        ##        ##     ##  ##   ##   ##   ##      BEF00070
C      #######   ########  ##         #######   ##    ##  ##    ##     BEF00080
C                                                                       BEF00090
C           BEFORe Rbreak for numerically stable computation            BEF00100
C                                                                       BEF00110
C  Written by Andojo Wurjanto under Supervision of Nobuhisa Kobayashi   BEF00120
C                  Center for Applied Coastal Research                  BEF00130
C                    Department of Civil Engineering                    BEF00140
C             University of Delaware, Newark, Delaware 19716            BEF00150
C                              May  1991                                BEF00160
C                                                                       BEF00170
C ############################  PURPOSE  ##############################  BEF00180
C                                                                       BEF00190
C     BEFORR should be executed before computer program RBREAK for      BEF00200
C        numerical simulation of random waves on impermeable            BEF00210
C        breakwaters and beaches.                                       BEF00220
C     There are three crucial computation parameters that need to be    BEF00230
C        specified in RBREAK computation:                               BEF00240
C            1. INITS = number of computational nodes below SWL         BEF00250
C            2. NONE  = number of time steps in one wave period         BEF00260
C            3. DELTA = normalized water depth at computational         BEF00270
C                       waterline                                       BEF00280
C     The value of INITS and an initial value of DELTA are supplied     BEF00290
C        by users.                                                      BEF00300
C     BEFORR and RBREAK break computation duration into computation     BEF00310
C        units. Each computation unit is one wave period long except    BEF00320
C        for the last unit, the length of which may be less than one    BEF00330
C        wave period.                                                   BEF00340
C     The main purpose of BEFORR is to assign the values of NONE and    BEF00350
C        DELTA to each computation unit of RBREAK so as to eliminate    BEF00360
C        numerical difficulties, especially at the computational        BEF00370
C        waterline.                                                     BEF00380
C     The "wave period" is the reference period used for the            BEF00390
C        normalization of the governing equations.                      BEF00400
C     BEFORR reads RBREAK's input data, checks some of them, and        BEF00410
C        writes the computation parameters it assigns in the output     BEF00420
C        file 'BINPUT', which will be the second (if IWAVE=1) or        BEF00430
C        third (if IWAVE>1) input data file for RBREAK computation.     BEF00440
C     Notes:                                                            BEF00450
C        BEFORR and RBREAK share many subroutine and COMMON block       BEF00460
C        names, but they are independent programs.                      BEF00470
C                                                                       BEF00480
C ########################  MAIN PROGRAM BEFORR  ######################  BEF00490
C                                                                       BEF00500
C     Main program performs time-marching computation using             BEF00510
C     subroutines                                                       BEF00520
C                                                                       BEF00530
      PROGRAM BEFORR                                                    BEF00540
C                                                                       BEF00550
```

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                BEF00560
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)               BEF00570
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI,KCNO,MCNO,KC2                  BEF00580
      CHARACTER*10 FINP1,FINP2                                           BEF00590
      INTEGER S,SM1,SP1,SSAV                                             BEF00600
C                                                                        BEF00610
C     COMMONs                                                            BEF00620
C     ------------------------------------------------------------       BEF00630
C      Name     Contents                                                 BEF00640
C     ------------------------------------------------------------       BEF00650
C     /DIMENS/  The values of FORTRAN PARAMETERs specified in the main   BEF00660
C               program                                                  BEF00670
C     /CONSTA/  Basic constants                                          BEF00680
C     /ID/      Identifiers specifying user's options                    BEF00690
C     /CPAR1/ to /CPAR4/ Computation parameters                          BEF00700
C     /REQ1/ and /REQ2/ Identifiers and values associated with special   BEF00710
C               storing (see Subr. 02 INPUT1 for more explanation)       BEF00720
C     /WAVE1/   Physical wave parameters                                 BEF00730
C     /WAVE2/   Dimensionless wave parameters                            BEF00740
C     /WAVE3/   Time series of normalized surface elevation at           BEF00750
C               seaward boundary                                         BEF00760
C     /WAVE4/   Maximum and minimum of time series in /WAVE3/            BEF00770
C     /WAVE5/   Cnoidal wave parameters (K, E, m and 1-m)                BEF00780
C     /WAVE6/   Number of data points specified in input file FINP2      BEF00790
C     /BOT1/    Physical quantities related to structure                 BEF00800
C     /BOT2/    Normalized quantities related to structure               BEF00810
C     /BOT3/    Normalized structure geometry                            BEF00820
C     /BOT4/ and /BOT5/ Physical structure geometry                      BEF00830
C     /HYDRO/   Hydrodynamic quantities computed                         BEF00840
C     /MATRIX/  Elements of matrices used in numerical method            BEF00850
C     /RUNP1/ and /RUNP2/ Quantities related to wave runup               BEF00860
C     /ARMOR/   Armor stability parameters read as input                 BEF00870
C     /SAVBC/   Node numbers related to options ISAVB=1 and ISAVC=1      BEF00880
C     /VALUEN/  Values at present time level stored during computation   BEF00890
C     /DIAGID/, /DIAGV1/, and /DIAGV2/ Indicators and saving variables   BEF00900
C               for diagnostic purposes                                  BEF00910
C     ------------------------------------------------------------       BEF00920
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                            BEF00930
      COMMON /CONSTA/ PI,GRAV                                            BEF00940
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,                BEF00950
     +                ISAVA,ISAVB,ISAVC                                  BEF00960
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES             BEF00970
      COMMON /CPAR2/  MULTIF,NONEM,NONE,NEND,NRATE                       BEF00980
      COMMON /CPAR3/  INITS,JE,JE1                                       BEF00990
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX      BEF01000
      COMMON /REQ1/   IREQ,IELEV,IV,IDUDT,ISNR,NREQ                      BEF01010
      COMMON /REQ2/   TREQ(N5)                                           BEF01020
      COMMON /WAVE1/  HREFP,TP,WL0P                                      BEF01030
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA           BEF01040
      COMMON /WAVE3/  ETA(N2)                                            BEF01050
      COMMON /WAVE4/  ETAMAX,ETAMIN                                      BEF01060
      COMMON /WAVE5/  KCNO,ECNO,MCNO,KC2                                 BEF01070
      COMMON /WAVE6/  NDATA                                              BEF01080
      COMMON /BOT1/   DSEAP,DLANDP,FWP                                   BEF01090
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     BEF01100
```

```
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)      BEF01110
      COMMON /BOT4/   NBSEG                                              BEF01120
      COMMON /BOT5/   WBSEG(N4),TBSLOP(N4),XBSEG(N4),ZBSEG(N4)           BEF01130
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)              BEF01140
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)   BEF01150
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                               BEF01160
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3)                               BEF01170
      COMMON /ARMOR/  C2,C3,CD,CL,CM,SG,TANPHI,AMIN,AMAX,DAP             BEF01180
      COMMON /SAVBC/  NNODB,NNODC,NODB(N5),NODC(N5)                      BEF01190
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                      BEF01200
      COMMON /DIAGID/ INONE,ITRY,IREV,LEVEL                              BEF01210
      COMMON /DIAGV1/ SSAV(3),JMSAV(3),NONESV(N6),MFIVE                  BEF01220
      COMMON /DIAGV2/ USAV(6,N1),VSAV(3,N1),ESAV(3,N1),DELSAV(3),        BEF01230
     +                DELTSV(N6)                                         BEF01240
C                                                                        BEF01250
C             READ INPUT                                                 BEF01260
C             ----------------------------------------------------       BEF01270
C             Subr. 01 OPENIO opens input and output files               BEF01280
C             Subr. 02 INPUT1 reads primary input data                   BEF01290
C             Subr. 03 INPUT2 reads wave profile at seaward boundary     BEF01300
C             ----------------------------------------------------       BEF01310
      WRITE (*,*) 'BEFORR reports progress on screen every MREP waves'   BEF01320
      WRITE (*,*) 'Enter MREP (0 if you do not want the report)'         BEF01330
      READ  (*,*)  MREP                                                  BEF01340
      WRITE (*,*) 'Name of primary input-data-file?'                     BEF01350
      READ  (*,5000) FINP1                                               BEF01360
      CALL OPENIO (1,FINP1,FINP2)                                        BEF01370
      CALL INPUT1 (1,FINP2)                                              BEF01380
      CALL OPENIO (2,FINP1,FINP2)                                        BEF01390
      IF (IWAVE.GT.1) CALL INPUT2 (1)                                    BEF01400
 5000 FORMAT (A10)                                                       BEF01410
C                                                                        BEF01420
C             CHECK FORTRAN PARAMETERS                                   BEF01430
C             ----------------------------------------------------       BEF01440
C             Each subroutine having FORTRAN "PARAMETER" statement is    BEF01450
C             called by Subroutine 39 CKSUBR to check its "PARAMETER"    BEF01460
C             values, which must be the same as their counterparts in    BEF01470
C             the main program. These checking calls to subroutines      BEF01480
C             are indicated by MODE=0, as opposed to "working" calls     BEF01490
C             indicated by MODE=1 as used above in Call INPUT1 (1,...)   BEF01500
C             and Call INPUT2 (1).                                       BEF01510
C             ----------------------------------------------------       BEF01520
      CALL CKSUBR (N1,N2,N3,N4,N5,N6)                                    BEF01530
      MODE=1                                                             BEF01540
C                                                                        BEF01550
C             GROUNDWORK                                                 BEF01560
C             ----------------------------------------------------       BEF01570
C           . Subr. 04 BOTTOM computes normalized structure geometry     BEF01580
C           . Subr. 05 WPARAM calculates wave parameters                 BEF01590
C           . This call to Subr. 10 CPARAM calculates NONEM = mininum    BEF01600
C             NONE                                                       BEF01610
C           . Subr. 26 INIT specifies initial conditions                 BEF01620
C           . This call to Subr. 06 REGWAV is for initial incident       BEF01630
C             wave profile and documentation purpose in Subr. 31 DOC1    BEF01640
C             where NONE=NONEM is used                                   BEF01650
```

```
C           . Subr. 31 DOC1 writes essential information before          BEF01660
C             time-marching computation                                  BEF01670
C             ---------------------------------------------------------  BEF01680
      CALL BOTTOM (MODE)                                                 BEF01690
      CALL WPARAM (MODE)                                                 BEF01700
      CALL CPARAM (0,0,0,0.D+00)                                         BEF01710
      TIME = 0.D+00                                                      BEF01720
      CALL INIT   (MODE)                                                 BEF01730
      IF (IWAVE.EQ.1) CALL REGWAV (MODE)                                 BEF01740
      CALL DOC1   (MODE,FINP2)                                           BEF01750
      IF (IJOB.EQ.3) SM1=INITS-1                                         BEF01760
C                                                                        BEF01770
C             TIME MARCHING COMPUTATION                                  BEF01780
C             ---------------------------------------------------------  BEF01790
C             Loop with bigger number nests to one with smaller number   BEF01800
C             . 510 contains one set of five computation units indi-     BEF01810
C                   cated by MM (MFIVE is defined in Subr. 02 INPUT1)    BEF01820
C             . 520 represents one computation unit indicated by M       BEF01830
C             . 530 constitutes one try for a computation unit,          BEF01840
C                   indicated by ITRY                                    BEF01850
C             . 540 performs time-marching from one time level to the    BEF01860
C                   next, indicated by N                                 BEF01870
C             ---------------------------------------------------------  BEF01880
      M     = 0                                                          BEF01890
      MM    = 0                                                          BEF01900
      ILEV3 = 0                                                          BEF01910
      IPROB = 0                                                          BEF01920
  510 IF (MM.LT.MFIVE) THEN                                              BEF01930
C       ========: 510 Begins :=================================          BEF01940
C       IREV = reverse indicator only for random waves.                  BEF01950
C           IPROB=9 dictates that computation be reversed to either      BEF01960
C           LEVEL=2 or LEVEL=3 (reversal to LEVEL=3 can be made only     BEF01970
C           once when ILEV3=0). IREV indicates how many times reversal   BEF01980
C           has been made.                                               BEF01990
C       LEVEL=2: beginning of the present set of five comp. units        BEF02000
C       LEVEL=3: beginning of the previous set of five comp. units       BEF02010
C       Subr. 28 INITMM sets initial hydrodynamic quantities at the      BEF02020
C           beginning of one set of five computation units using the     BEF02030
C           saved values at LEVEL=2 or LEVEL=3                           BEF02040
C       Subr. 30 SAVEMM saves hydrodynamic quantities at the end of      BEF02050
C           one set of five computation units (LEVEL=2 and LEVEL=3)      BEF02060
C       --------:                                                        BEF02070
        IF (IPROB.EQ.9) THEN                                             BEF02080
          IF (IREV.EQ.5) CALL STOPP (1,2)                                BEF02090
          IREV = IREV+1                                                  BEF02100
          MFAR = M-5*(MM-1)                                              BEF02110
          IF (MFAR.GE.2.OR.ILEV3.EQ.1.OR.MM.EQ.1) THEN                   BEF02120
            LEVEL = 2                                                    BEF02130
          ELSE                                                           BEF02140
            LEVEL = 3                                                    BEF02150
            ILEV3 = 1                                                    BEF02160
            MM = MM-1                                                    BEF02170
          ENDIF                                                          BEF02180
          CALL INITMM (MODE)                                             BEF02190
          IPROB = 0                                                      BEF02200
```

```
      ELSE                                                              BEF02210
        CALL SAVEMM (MODE)                                              BEF02220
        IREV  = 0                                                       BEF02230
        ILEV3 = 0                                                       BEF02240
        MM = MM+1                                                       BEF02250
      ENDIF                                                             BEF02260
      M = (MM-1)*5                                                      BEF02270
      IF (MM.LT.MFIVE) THEN                                             BEF02280
        MEND = MM*5                                                     BEF02290
      ELSE                                                              BEF02300
        MEND = MWAVE                                                    BEF02310
      ENDIF                                                             BEF02320
  520 IF (M.LT.MEND.AND.IPROB.NE.9) THEN                                BEF02330
C       ------: 520 Begins :-----------------------------------------  BEF02340
        M = M+1                                                         BEF02350
        ITRY = 0                                                        BEF02360
  530   CONTINUE                                                        BEF02370
C         ----: 530 begins :-----------------------------------------  BEF02380
C       . Subr. 27 INITM sets initial hydrodynamic quantities at the    BEF02390
C         beginning of a computation unit using the saved values at     BEF02400
C         LEVEL=1                                                       BEF02410
C       . Subr. 10 CPARAM assigns computation parameters NONE and       BEF02420
C         DELTA. When NONE is considered too big, IPROB=9 is set,       BEF02430
C         meaning that computation must be reversed to LEVEL=2 or       BEF02440
C         LEVEL=3. Otherwise, IPROB=0 at the beginning of each trial.   BEF02450
C       . Subr. 06 REGWAV computes incident regular wave profile        BEF02460
C         ----:                                                         BEF02470
          ITRY = ITRY+1                                                 BEF02480
          IF (IPROB.NE.0) CALL INITM (MODE)                             BEF02490
          CALL CPARAM (MODE,M,IPROB,U2STOP)                             BEF02500
          IF (INONE.EQ.1) CALL REGWAV (MODE)                            BEF02510
          N = 0                                                         BEF02520
  540     IF (IPROB.EQ.0.AND.N.LT.NEND) THEN                            BEF02530
C           --: 540 Begins ---------------------------------------      BEF02540
C         . TIME = normalized time                                      BEF02550
C         . Subr. 12 EXTRAP estimates hydrodynamic quantities at the    BEF02560
C           node immediately landward of the present waterline node     BEF02570
C         . Subr. 13 RETAIN retains some values at present time         BEF02580
C           level at landward and seaward boundaries                    BEF02590
C         . Subr. 14 CRITV calculates critical velocities used in       BEF02600
C           characteristic variables and checks for negative water      BEF02610
C           depth (IPROB=1)                                             BEF02620
C         . Subr. 15 MARCH performs time-marching from present time     BEF02630
C           level (N-1) to next (N) excluding landward and seaward      BEF02640
C           boundaries. If the waterline moves seaward more than one    BEF02650
C           grid spacing, IPROB=2 is set.                               BEF02660
C         . Landward and seaward boundary conditions are handled by     BEF02670
C           Subr. 16 LANDBC and 19 SEABC, respectively                  BEF02680
C         . Subr. 11 NUMSTA checks if numerical stability criterion     BEF02690
C           is violated (IPROB=3)                                       BEF02700
C           --:                                                         BEF02710
            N = N+1                                                     BEF02720
            TIME = DBLE(M-1) + DBLE(N)/DBLE(NONE)                       BEF02730
            IF (IJOB.LT.3) CALL EXTRAP (MODE)                           BEF02740
            CALL RETAIN (MODE)                                          BEF02750
```

```
              CALL CRITV  (MODE,M,IPROB)                                  BEF02760
              IF (IPROB.EQ.0) THEN                                        BEF02770
                CALL MARCH (MODE,M,IPROB,U2STOP)                          BEF02780
                IF (IPROB.EQ.0) THEN                                      BEF02790
                  CALL LANDBC (MODE,M)                                    BEF02800
                  CALL SEABC  (MODE,M,N)                                  BEF02810
                  CALL NUMSTA (MODE,M,IPROB)                              BEF02820
                  IF (IJOB.LT.3.AND.S.GT.JMAX) JMAX=S                     BEF02830
                ENDIF                                                     BEF02840
              ENDIF                                                       BEF02850
C             --: 540 Ends :----------------------------------------    BEF02860
            GOTO 540                                                      BEF02870
            ENDIF                                                         BEF02880
C           ----: 530 Ends :--------------------------------------------  BEF02890
          IF (IPROB.EQ.0) THEN                                            BEF02900
C           ::    Subr. 32 DOC2 writes computation log during             BEF02910
C                 time-marching computation                               BEF02920
C           ::    Subr. 29 SAVEM saves hydrodynamic quantities at the     BEF02930
C                 end of a computation unit (LEVEL=1)                     BEF02940
            CALL DOC2  (MODE,M,MREP)                                      BEF02950
            CALL SAVEM (MODE)                                             BEF02960
          ELSE                                                            BEF02970
            IF (IPROB.NE.9) GOTO 530                                      BEF02980
          ENDIF                                                           BEF02990
C         ------: 520 Ends :------------------------------------------   BEF03000
        GOTO 520                                                          BEF03010
        ENDIF                                                             BEF03020
C       ========: 510 Ends :==========================================   BEF03030
      GOTO 510                                                            BEF03040
      ENDIF                                                               BEF03050
C                                                                         BEF03060
C             FINISHING                                                   BEF03070
C             ------------------------------------------------------      BEF03080
C             Subr. 33 DOC3 writes in file 'BINPUT' and completes         BEF03090
C             documentation in file 'BDOC'                                BEF03100
C             ------------------------------------------------------      BEF03110
      CALL DOC3 (MODE)                                                    BEF03120
C                                                                         BEF03130
      STOP                                                                BEF03140
      END                                                                 BEF03150
C                                                                         BEF03160
C ---------------------  END OF MAIN PROGRAM BEFORR ------------------    BEF03170
C #01######################  SUBROUTINE OPENIO  ###############BEFORR#    BEF03180
C                                                                         BEF03190
C     This subroutine opens input and output files                        BEF03200
C                                                                         BEF03210
      SUBROUTINE OPENIO (ICALL,FINP1,FINP2)                               BEF03220
C                                                                         BEF03230
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,                 BEF03240
     +                ISAVA,ISAVB,ISAVC                                   BEF03250
      CHARACTER OL*3,NE*3,SE*10,FINP1*10,FINP2*10                         BEF03260
      DATA OL,NE,SE /'OLD','NEW','SEQUENTIAL'/                            BEF03270
C                                                                         BEF03280
C  DESCRIPTION OF INPUT AND OUTPUT FILES                                  BEF03290
C  ------------------------------------------------------------------     BEF03300
```

```
C  (Category) Unit (Filename) Purpose                                      BEF03310
C  ------------------------------------------------------------------     BEF03320
C  (Input)  11 (FINP1)     contains primary input data                     BEF03330
C  (Input)  12 (FINP2)     contains time series of normalized  surface     BEF03340
C                          elevation at seaward boundary                   BEF03350
C  (Output) 51 ('BINPUT')  stores computation parameters to be used        BEF03360
C                          as input to RBREAK computation                  BEF03370
C                          --> m,NONE(m),NEND(m),DELTA(m)                  BEF03380
C                              m = 1,2,...,MWAVE = computation unit        BEF03390
C  (Output) 52 ('BSPACE')  stores structure geometry                       BEF03400
C                          --> JE,(XB(j),ZB(j),j=1,JE)                     BEF03410
C  (Output) 98 ('BDOC')    stores essential output for concise             BEF03420
C                          documentation                                   BEF03430
C  (Output) 99 ('BMSG')    stores messages written under special           BEF03440
C                          circumstances during computation                BEF03450
C  ------------------------------------------------------------------     BEF03460
      IF (ICALL.EQ.1) THEN                                                 BEF03470
        OPEN (UNIT=11,FILE=FINP1   ,ACCESS=SE,STATUS=OL)                   BEF03480
        OPEN (UNIT=51,FILE='BINPUT',ACCESS=SE,STATUS=NE)                   BEF03490
        OPEN (UNIT=52,FILE='BSPACE',ACCESS=SE,STATUS=NE)                   BEF03500
        OPEN (UNIT=98,FILE='BDOC'  ,ACCESS=SE,STATUS=NE)                   BEF03510
        OPEN (UNIT=99,FILE='BMSG'  ,ACCESS=SE,STATUS=NE)                   BEF03520
      ELSE                                                                 BEF03530
        IF (IWAVE.GT.1) OPEN (UNIT=12,FILE=FINP2,ACCESS=SE,STATUS=OL)      BEF03540
      ENDIF                                                                BEF03550
      RETURN                                                               BEF03560
      END                                                                  BEF03570
C                                                                          BEF03580
C -01------------------  END OF SUBROUTINE OPENIO ------------BEFORR-      BEF03590
C #02####################  SUBROUTINE INPUT1  ###############BEFORR#      BEF03600
C                                                                          BEF03610
C     This subroutine reads data from primary input data file and          BEF03620
C     checks some of them                                                  BEF03630
C                                                                          BEF03640
      SUBROUTINE INPUT1 (MODE,FINP2)                                       BEF03650
C                                                                          BEF03660
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                  BEF03670
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                 BEF03680
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                                  BEF03690
      CHARACTER*10 FINP2                                                   BEF03700
      CHARACTER*5 COMMEN(15)                                               BEF03710
      INTEGER S,SM1,SP1,SSAV                                               BEF03720
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                              BEF03730
      COMMON /CONSTA/ PI,GRAV                                              BEF03740
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,                  BEF03750
     +                ISAVA,ISAVB,ISAVC                                    BEF03760
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES               BEF03770
      COMMON /CPAR2/  MULTIF,NONEM,NONE,NEND,NRATE                         BEF03780
      COMMON /CPAR3/  INITS,JE,JE1                                         BEF03790
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX        BEF03800
      COMMON /REQ1/   IREQ,IELEV,IV,IDUDT,ISNR,NREQ                        BEF03810
      COMMON /REQ2/   TREQ(N5)                                             BEF03820
      COMMON /WAVE1/  HREFP,TP,WL0P                                        BEF03830
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA             BEF03840
      COMMON /BOT1/   DSEAP,DLANDP,FWP                                     BEF03850
```

```
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     BEF03860
      COMMON /BOT4/   NBSEG                                             BEF03870
      COMMON /BOT5/   WBSEG(N4),TBSLOP(N4),XBSEG(N4),ZBSEG(N4)          BEF03880
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              BEF03890
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3)                              BEF03900
      COMMON /ARMOR/  C2,C3,CD,CL,CM,SG,TANPHI,AMIN,AMAX,DAP            BEF03910
      COMMON /SAVBC/  NNODB,NNODC,NODB(N5),NODC(N5)                     BEF03920
      COMMON /DIAGV1/ SSAV(3),JMSAV(3),NONESV(N6),MFIVE                 BEF03930
      IF (MODE.EQ.0) THEN                                               BEF03940
C       ::   MODE=0 is used in Subr. 39 CKSUBR                          BEF03950
C       ::   Although not used in any of the above COMMON blocks, N1    BEF03960
C            needs to be checked in relation with Call CKVAL1 (5,...)   BEF03970
        CALL CKFPAR (2,1,N1,N1R)                                        BEF03980
        CALL CKFPAR (2,3,N3,N3R)                                        BEF03990
        CALL CKFPAR (2,4,N4,N4R)                                        BEF04000
        CALL CKFPAR (2,5,N5,N5R)                                        BEF04010
        CALL CKFPAR (2,6,N6,N6R)                                        BEF04020
        RETURN                                                          BEF04030
      ENDIF                                                             BEF04040
C                                                                       BEF04050
C           COMMENT LINES                                               BEF04060
C           ------------------------------------------------------      BEF04070
C           NLINES = number of comment lines preceding primary          BEF04080
C                    input data                                         BEF04090
C           Comment lines are read from primary input file and then     BEF04100
C           written as the heading of output files 'BINPUT', 'BDOC',    BEF04110
C           and 'BMSG'                                                  BEF04120
C           ------------------------------------------------------      BEF04130
      READ  (11,1110) NLINES                                            BEF04140
      WRITE (51,1110) NLINES                                            BEF04150
      DO 110 L = 1,NLINES                                               BEF04160
        READ  (11,1120) (COMMEN(I),I=1,15)                              BEF04170
        WRITE (51,1120) (COMMEN(I),I=1,15)                              BEF04180
        WRITE (98,1120) (COMMEN(I),I=1,15)                              BEF04190
        WRITE (99,1120) (COMMEN(I),I=1,15)                              BEF04200
  110 CONTINUE                                                          BEF04210
C                                                                       BEF04220
C           OPTIONS                                                     BEF04230
C           ------------------------------------------------------      BEF04240
C           IJOB=1: RUNUP on impermeable slope                          BEF04250
C               =2: OVERTOPPING over subaerial structure                BEF04260
C               =3: TRANSMISSION over submerged structure               BEF04270
C           ISTAB=0: No computation of armor stability or movement      BEF04280
C                =1: Armor stability computation                        BEF04290
C                =2: Armor movement computation                         BEF04300
C           ISYST=1: International System of Units (SI) is used         BEF04310
C                =2: US Customary System of Units (USCS) is used        BEF04320
C           IBOT=1: "Type 1" bottom data (width-slope)                  BEF04330
C               =2: "Type 2" bottom data (coordinates)                  BEF04340
C           INONCT=0: No correction term in computing ETAR              BEF04350
C                  =1: Correction term for ETAR recommended for beaches BEF04360
C           IENERG=0: Energy quantities NOT computed                    BEF04370
C                  =1: Energy quantities computed                       BEF04380
C           IWAVE=1: Incident waves at seaward boundary computed        BEF04390
C                =2: Incident waves at seaward boundary given as input  BEF04400
```

```
C              =3: Total waves at seaward boundary given as input      BEF04410
C           If IWAVE>1 --> Must specify MSTAT and FINP2                 BEF04420
C             . FINP2 = name of input data file containing time series  BEF04430
C               of normalized free surface elevation at seaw. boundary  BEF04440
C             . Statistical calculations are performed excluding the    BEF04450
C               first MSTAT computation units                           BEF04460
C             . A "computation unit" is one wave period long except     BEF04470
C               for the last unit, the length of which may be less      BEF04480
C               than one wave period                                    BEF04490
C             . The "wave period" is the reference period used for the  BEF04500
C               normalization of the governing equations                BEF04510
C           "A" = Spatial variations of hydrodynamic quantities at      BEF04520
C                 specified moments                                     BEF04530
C           "B" = Time series of volume flux and water depth at         BEF04540
C                 specified nodes                                       BEF04550
C           "C" = Time series of the following:                         BEF04560
C               . For ISTAB=1: stability number SNR at specified nodes  BEF04570
C               . For ISTAB=2: displacement of armor units from         BEF04580
C                              specified initial nodal locations        BEF04590
C           "B" and "C" are stored at rate NRATE data points per wave   BEF04600
C                 period                                                BEF04610
C           ISAVA,ISAVB,ISAVC are identifiers associated with saving    BEF04620
C                 "A","B","C", respectively (1=save; 0=no)              BEF04630
C           MSAVA1, MSAVA2, AND NTIMES:                                 BEF04640
C           If ISAVA=1, "A" is saved NTIMES (>1) times at equal         BEF04650
C                       intervals per wave period from computation      BEF04660
C                       unit MSAVA1 to MSAVA2, inclusive                BEF04670
C           If ISAVB=1 --> Must specify NNODB                           BEF04680
C           If ISAVC=1 --> Must specify NNODC                           BEF04690
C           NNODB,NNODC = number of nodes for which "B","C" is to be    BEF04700
C                         saved                                         BEF04710
C           IREQ=0: No special storing                                  BEF04720
C               =1: Special storing requested                           BEF04730
C           Special storing = storing spatial variations of requested   BEF04740
C               quantities at normalized time TREQ(i)                   BEF04750
C               with i=1,2,...,NREQ                                     BEF04760
C           Quantities available for request:                           BEF04770
C           . ELEV = surface elevation                                  BEF04780
C           . V    = depth-averaged velocity                            BEF04790
C           . DUDT = total fluid acceleration                           BEF04800
C           . SNR  = stability number against rolling/sliding           BEF04810
C           --> requested by IELEV=1, IV=1, IDUDT=1, and ISNR=1,        BEF04820
C               respectively                                            BEF04830
C           Notes: . DUDT can be requested only if ISTAB>0              BEF04840
C                  . SNR  can be requested only if ISTAB=1              BEF04850
C           ------------------------------------------------------      BEF04860
      READ  (11,1130) IJOB,ISTAB                                        BEF04870
      READ  (11,1130) ISYST                                             BEF04880
      READ  (11,1130) IBOT                                              BEF04890
      READ  (11,1130) INONCT                                            BEF04900
      READ  (11,1130) IENERG                                            BEF04910
      READ  (11,1140) IWAVE,MSTAT,FINP2                                 BEF04920
      READ  (11,1150) ISAVA,MSAVA1,MSAVA2,NTIMES                        BEF04930
      READ  (11,1150) ISAVB,NNODB                                       BEF04940
      READ  (11,1150) ISAVC,NNODC                                       BEF04950
```

```
      READ  (11,1160) IREQ,IELEV,IV,IDUDT,ISNR,NREQ                    BEF04960
C     ::    Check user-chosen options using Subr. 35 CKOPTI             BEF04970
      INDIC=0                                                           BEF04980
      CALL CKOPTI ( 1,INDIC,IJOB  ,1,3)                                 BEF04990
      CALL CKOPTI ( 2,INDIC,ISTAB ,0,2)                                 BEF05000
      CALL CKOPTI ( 3,INDIC,ISYST ,1,2)                                 BEF05010
      CALL CKOPTI ( 4,INDIC,IBOT  ,1,2)                                 BEF05020
      CALL CKOPTI ( 5,INDIC,INONCT,0,1)                                 BEF05030
      CALL CKOPTI ( 6,INDIC,IENERG,0,1)                                 BEF05040
      CALL CKOPTI ( 7,INDIC,IWAVE ,1,3)                                 BEF05050
      CALL CKOPTI ( 8,INDIC,ISAVA ,0,1)                                 BEF05060
      CALL CKOPTI ( 9,INDIC,ISAVB ,0,1)                                 BEF05070
      CALL CKOPTI (11,INDIC,IREQ  ,0,1)                                 BEF05080
      IF (ISTAB.GT.0) CALL CKOPTI (10,INDIC,ISAVC,0,1)                  BEF05090
      IF (IREQ.EQ.1) THEN                                               BEF05100
        CALL CKOPTI (12,INDIC,IELEV,0,1)                                BEF05110
        CALL CKOPTI (13,INDIC,IV   ,0,1)                                BEF05120
        CALL CKOPTI (14,INDIC,IDUDT,0,1)                                BEF05130
        CALL CKOPTI (15,INDIC,ISNR ,0,1)                                BEF05140
      ENDIF                                                             BEF05150
      IF (INDIC.GT.0) CALL STOPP (3,4)                                  BEF05160
C     ::    Subr. 37,38 CKVAL1,CKVAL2 check user-specified values       BEF05170
C           Subr. 40 STOPP stops execution of the computation           BEF05180
      IF (IREQ.EQ.1) THEN                                               BEF05190
        CALL CKVAL1 (1,NREQ,1,N5)                                       BEF05200
        IF (ISTAB.EQ.0) IDUDT=0                                         BEF05210
        IF (ISTAB.NE.1) ISNR=0                                          BEF05220
        NOREQ = IELEV+IV+IDUDT+ISNR                                     BEF05230
        IF (NOREQ.EQ.0) THEN                                            BEF05240
          CALL STOPP (5,7)                                              BEF05250
        ELSE                                                            BEF05260
          READ (11,1180) (TREQ(I),I=1,NREQ)                             BEF05270
        ENDIF                                                           BEF05280
      ELSE                                                              BEF05290
        NREQ = 0                                                        BEF05300
      ENDIF                                                             BEF05310
      IF (ISAVB.EQ.1) CALL CKVAL1 (2,NNODB,1,N5)                        BEF05320
      IF (ISAVC.EQ.1) CALL CKVAL1 (3,NNODC,1,N5)                        BEF05330
C                                                                       BEF05340
C           COMPUTATION PARAMETERS                                      BEF05350
C           ----------------------------------------------------------  BEF05360
C           INITS = . number of spatial nodes along the bottom          BEF05370
C                     below SWL (IJOB<3)                                BEF05380
C                   . number of nodes between seaward and               BEF05390
C                     landward boundaries, inclusive (IJOB=3)           BEF05400
C           Notes:  . INITS should be so large that delta x between     BEF05410
C                     two adjacent nodes is sufficiently small          BEF05420
C                   . INITS=100 to 600 has been used                    BEF05430
C           MULTIF = multiplication factor associated with NONEM        BEF05440
C           Notes:  . NONEM = minimum allowable value of NONE           BEF05450
C                   . NONE  = number of time steps in one wave period   BEF05460
C                   . See Subr. 10 CPARAM for more explanation about    BEF05470
C                     MULTIF and NONEM                                  BEF05480
C           DELTA as input is an initial value for normalized water     BEF05490
C                 depth defining computational waterline. A range of    BEF05500
```

```
C                values of DELTA from 0.001 to 0.003 has been used.    BEF05510
C           TMAX  = normalized duration of computation                 BEF05520
C           NRATE = rate (data points per per wave period) of storing  BEF05530
C                   of output time series                              BEF05540
C           NDELR = number of "DELRP"s to be specified                 BEF05550
C           DELRP = physical water depth associated with visual or     BEF05560
C                   measured waterline                                 BEF05570
C                   . in millimeters if ISYST=1 (SI)                   BEF05580
C                   . in inches      if ISYST=2 (USCS)                 BEF05590
C           X1,X2 = damping coefficients                               BEF05600
C           ---------------------------------------------------------  BEF05610
      READ  (11,1110) INITS                                            BEF05620
      READ  (11,1110) MULTIF                                           BEF05630
      READ  (11,1170) DELTA                                            BEF05640
      READ  (11,1170) TMAX                                             BEF05650
      READ  (11,1110) NRATE                                            BEF05660
      READ  (11,1110) NDELR                                            BEF05670
      CALL CKVAL1 (5,INITS,1,N1-1)                                     BEF05680
      IF (IJOB.LT.3) THEN                                              BEF05690
        CALL CKVAL1 (4,NDELR,1,N3)                                     BEF05700
      ELSE                                                             BEF05710
        NDELR = 0                                                      BEF05720
      ENDIF                                                            BEF05730
      INDIC=0                                                          BEF05740
      DO 120 L = 1,NDELR                                               BEF05750
        READ (11,1180) DELRP(L)                                        BEF05760
        IF (DELRP(L).LE.0.D+00) CALL CKVAL2 (2,INDIC,'DELRP ')         BEF05770
  120 CONTINUE                                                         BEF05780
      READ (11,1180) X1,X2                                             BEF05790
C                                                                      BEF05800
C           SOME MORE CHECKS                                           BEF05810
C           ---------------------------------------------------------  BEF05820
C           MWAVE = number of computation units                        BEF05830
C           MFIVE = number of sets of five computation units           BEF05840
C           MSTAT: Statistical calculations are performed excluding    BEF05850
C                  the first MSTAT computation units                   BEF05860
C           MSTAB: Computation of armor stability or movement is       BEF05870
C                  performed excluding the first MSTAB computation     BEF05880
C                  units                                               BEF05890
C           ---------------------------------------------------------  BEF05900
      ITMAX = INT(TMAX)                                                BEF05910
      RES   = DMOD(TMAX,1.D+00)                                        BEF05920
      IF (DABS(RES).LT.1.D-08) THEN                                    BEF05930
        MWAVE = ITMAX                                                  BEF05940
      ELSE                                                             BEF05950
        MWAVE = ITMAX+1                                                BEF05960
        IF (IWAVE.EQ.1) CALL STOPP (8,8)                               BEF05970
      ENDIF                                                            BEF05980
      IRES = MOD(MWAVE,5)                                              BEF05990
      IF (IRES.EQ.0) THEN                                              BEF06000
        MFIVE = MWAVE/5                                                BEF06010
      ELSE                                                             BEF06020
        MFIVE = MWAVE/5+1                                              BEF06030
      ENDIF                                                            BEF06040
      DO 130 I = 1,NREQ                                                BEF06050
```

```
        IF (TREQ(I).GT.TMAX)    CALL STOPP  (9,9)                       BEF06060
        IF (TREQ(I).LE.0.D+00) CALL CKVAL2 (2,INDIC,'TREQ  ')           BEF06070
  130 CONTINUE                                                          BEF06080
      CALL CKVAL1 (8,MSTAT,0,ITMAX-1)                                   BEF06090
      IF (IWAVE.EQ.1) THEN                                              BEF06100
        MSTAT = MWAVE-1                                                 BEF06110
      ENDIF                                                             BEF06120
      IF (ISTAB.EQ.1) THEN                                              BEF06130
        MSTAB = MSTAT                                                   BEF06140
      ELSEIF (ISTAB.EQ.2) THEN                                          BEF06150
        MSTAB = 0                                                       BEF06160
      ELSE                                                              BEF06170
        MSTAB = MWAVE+1                                                 BEF06180
        ISAVC = 0                                                       BEF06190
      ENDIF                                                             BEF06200
      IF (ISAVA.EQ.1) THEN                                              BEF06210
        IF (NTIMES.LT.1) CALL CKVAL2 (1,INDIC,'NTIMES')                 BEF06220
        CALL CKVAL1  (9,MSAVA2,1,ITMAX)                                 BEF06230
        CALL CKVAL1 (10,MSAVA1,1,MSAVA2)                                BEF06240
      ELSE                                                              BEF06250
        MSAVA1=MWAVE+1                                                  BEF06260
      ENDIF                                                             BEF06270
C                                                                       BEF06280
C           CONSTANTS                                                   BEF06290
C           ------------------------------------------------------      BEF06300
C           PI   = 3.141592...                                          BEF06310
C           GRAV = gravitational acceleration                           BEF06320
C                  . in m/sec**2  if ISYST=1 (SI)                       BEF06330
C                  . in ft/sec**2 if ISYST=2 (USCS)                     BEF06340
C           ------------------------------------------------------      BEF06350
      PI = 4.D+00*DATAN(1.D+00)                                         BEF06360
      IF (ISYST.EQ.1) THEN                                              BEF06370
        GRAV = 9.81D+00                                                 BEF06380
      ELSE                                                              BEF06390
        GRAV = 32.2D+00                                                 BEF06400
      ENDIF                                                             BEF06410
C                                                                       BEF06420
C           WAVE PROPERTIES AND FRICTION FACTOR                         BEF06430
C           ------------------------------------------------------      BEF06440
C           HREFP = physical wave height at "reference" location        BEF06450
C                   . in meters if ISYST=1 (SI)                         BEF06460
C                   . in feet   if ISYST=2 (USCS)                       BEF06470
C           TP    = physical reference wave period, in seconds          BEF06480
C           HREFP and TP are used to normalize the governing            BEF06490
C                   equations                                           BEF06500
C           KSREF = shoaling coefficient at "reference" location        BEF06510
C           KSSEA = shoaling coefficient at seaward boundary            BEF06520
C           SIGMA is a measure of wave steepness                        BEF06530
C           FWP   = bottom friction factor                              BEF06540
C           ------------------------------------------------------      BEF06550
      READ  (11,1180) HREFP,TP                                          BEF06560
      READ  (11,1180) KSREF,KSSEA                                       BEF06570
      READ  (11,1180) FWP                                               BEF06580
      IF (HREFP.LE.0.D+00) CALL STOPP (10,10)                           BEF06590
      SIGMA = TP*DSQRT(GRAV/HREFP)                                      BEF06600
```

```
C                                                                       BEF06610
C           STRUCTURE GEOMETRY                                          BEF06620
C           ------------------------------------------------------      BEF06630
C           The structure geometry is divided into segments of          BEF06640
C           different inclination                                       BEF06650
C           NBSEG  = number of segments                                 BEF06660
C           DSEAP  = physical water depth below SWL at seaward          BEF06670
C                    boundary                                           BEF06680
C           TSLOPS = tangent of slope used to define                    BEF06690
C                    'surf similarity parameter'                        BEF06700
C           For segment i starting from the seaward boundary:           BEF06710
C             WBSEG(i)  = physical horizontal width                     BEF06720
C             TBSLOP(i) = tangent of slope (+ upslope, - downslope)     BEF06730
C             XBSEG(i)  = physical horizontal distance from seaward     BEF06740
C                         boundary to the segment's seaward-end         BEF06750
C             ZBSEG(i)  = physical water depth below SWL (+ below SWL)  BEF06760
C                         at the segment's seaward-end                  BEF06770
C           DSEAP,WBSEG,XBSEG,ZBSEG are in meters if ISYST=1 (SI),      BEF06780
C                                          feet   if ISYST=2 (USCS)     BEF06790
C           ------------------------------------------------------      BEF06800
      READ  (11,1180) DSEAP                                             BEF06810
      READ  (11,1180) TSLOPS                                            BEF06820
      READ  (11,1110) NBSEG                                             BEF06830
      CALL CKVAL1 (6,NBSEG,1,N4-1)                                      BEF06840
      IF (IBOT.EQ.1) THEN                                               BEF06850
        DO 140 K = 1,NBSEG                                              BEF06860
          READ (11,1180) WBSEG(K),TBSLOP(K)                             BEF06870
          IF (WBSEG(K).LE.0.D+00) CALL CKVAL2 (2,INDIC,'WBSEG ')        BEF06880
  140   CONTINUE                                                        BEF06890
      ELSE                                                              BEF06900
        DO 150 K = 1,NBSEG+1                                            BEF06910
          READ (11,1180) XBSEG(K),ZBSEG(K)                              BEF06920
C         IF (XBSEG(K).LT.0.D+00) CALL CKVAL2 (3,INDIC,'XBSEG ')        BEF06930
  150   CONTINUE                                                        BEF06940
      ENDIF                                                             BEF06950
C                                                                       BEF06960
C           DATA RELATED TO SAVING "B", i.e., time series of volume     BEF06970
C           flux and total water depth at specified nodes               BEF06980
C           ------------------------------------------------------      BEF06990
C           NNODB   = no. of nodes for which "B" is to be saved         BEF07000
C           NODB(I) = I-th node number for which "B" is to be saved     BEF07010
C           ------------------------------------------------------      BEF07020
      IF (ISAVB.EQ.1) READ (11,1190) (NODB(I),I=1,NNODB)                BEF07030
C                                                                       BEF07040
C           DATA RELATED TO ARMOR STABILITY AND MOVEMENT                BEF07050
C           ------------------------------------------------------      BEF07060
C           SG     = specific gravity                                   BEF07070
C           C2     = area coefficient                                   BEF07080
C           C3     = volume coefficient                                 BEF07090
C           CD     = drag coefficient                                   BEF07100
C           CL     = lift coefficient                                   BEF07110
C           CM     = inertia coefficient                                BEF07120
C           TANPHI = armor friction factor                              BEF07130
C           AMAX,AMIN = upper and lower bounds of fluid acceleration,   BEF07140
C                       normalized by gravitational acceleration, used  BEF07150
```

```
C                    only for ISTAB=1                                    BEF07160
C           DAP    = physical armor diameter                             BEF07170
C                    . in meters IF ISYST=1 (SI)                         BEF07180
C                    . in feet   IF ISYST=2 (USCS)                       BEF07190
C           NNODC   = no. of nodes for which "C" is to be saved          BEF07200
C           NODC(I) = I-th node number for which "C" is to be saved      BEF07210
C           "C" = time series of the following                           BEF07220
C               . For ISTAB=1: stability number SNR at specified nodes   BEF07230
C               . For ISTAB=2: displacement of armor units from          BEF07240
C                              specified initial nodal locations         BEF07250
C           ------------------------------------------------------------ BEF07260
C     ::    To compute SNR = stability number against rolling/sliding:   BEF07270
      IF (ISTAB.GT.0) THEN                                               BEF07280
        READ (11,1180) C2,C3,SG                                          BEF07290
        READ (11,1180) CD,CL,CM                                          BEF07300
        READ (11,1180) TANPHI                                            BEF07310
        READ (11,1180) AMAX,AMIN                                         BEF07320
        IF (ISAVC.EQ.1) READ (11,1190) (NODC(I),I=1,NNODC)               BEF07330
        IF (C2    .LE.0.D+00) CALL CKVAL2 (2,INDIC,'C2    ')             BEF07340
        IF (C3    .LE.0.D+00) CALL CKVAL2 (2,INDIC,'C3    ')             BEF07350
        IF (SG    .LE.0.D+00) CALL CKVAL2 (2,INDIC,'SG    ')             BEF07360
        IF (CD    .LE.0.D+00) CALL CKVAL2 (2,INDIC,'CD    ')             BEF07370
        IF (CL    .LT.0.D+00) CALL CKVAL2 (3,INDIC,'CL    ')             BEF07380
        IF (CM    .LT.0.D+00) CALL CKVAL2 (3,INDIC,'CM    ')             BEF07390
        IF (TANPHI.LE.0.D+00) CALL CKVAL2 (2,INDIC,'TANPHI')             BEF07400
      ENDIF                                                              BEF07410
C     ::    To compute movement of armor units, additional input is      BEF07420
C           required:                                                    BEF07430
      IF (ISTAB.EQ.2) THEN                                               BEF07440
        READ (11,1180) DAP                                               BEF07450
        IF (DAP.LE.0.D+00) CALL CKVAL2 (2,INDIC,'DAP   ')                BEF07460
      ENDIF                                                              BEF07470
C     ::    Subroutine 36 CKVAL performs check on a number of user-      BEF07480
C           specified values. The check in that subroutine involves      BEF07490
C           IF statements, and has to be separated from the present      BEF07500
C           subroutine because some FORTRAN compilers may have           BEF07510
C           problems with subroutines having too many IF statements.     BEF07520
      CALL CKVAL (INDIC)                                                 BEF07530
      IF (INDIC.GT.0) CALL STOPP (3,4)                                   BEF07540
C                                                                        BEF07550
C           FORMATS                                                      BEF07560
C           ------------------------------------------------------------ BEF07570
 1110 FORMAT (3I8)                                                       BEF07580
 1120 FORMAT (15A5)                                                      BEF07590
 1130 FORMAT (2I1)                                                       BEF07600
 1140 FORMAT (I1,I8,2X,A10)                                              BEF07610
 1150 FORMAT (I1,3I8)                                                    BEF07620
 1160 FORMAT (5I1,I8)                                                    BEF07630
 1170 FORMAT (F15.6)                                                     BEF07640
 1180 FORMAT (3F13.6)                                                    BEF07650
 1190 FORMAT (5I6)                                                       BEF07660
C                                                                        BEF07670
      RETURN                                                             BEF07680
      END                                                                BEF07690
C                                                                        BEF07700
```

```
C -02------------------  END OF SUBROUTINE INPUT1  ------------BEFORR-  BEF07710
C #03#######################  SUBROUTINE INPUT2  ###############BEFORR#  BEF07720
C                                                                       BEF07730
C     This subroutine reads wave profile at seaward boundary            BEF07740
C                                                                       BEF07750
      SUBROUTINE INPUT2 (MODE)                                          BEF07760
C                                                                       BEF07770
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF07780
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)              BEF07790
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                           BEF07800
      COMMON /WAVE3/  ETA(N2)                                           BEF07810
      COMMON /WAVE4/  ETAMAX,ETAMIN                                     BEF07820
      COMMON /WAVE6/  NDATA                                             BEF07830
      IF (MODE.EQ.0) THEN                                               BEF07840
        CALL CKFPAR (3,2,N2,N2R)                                        BEF07850
        RETURN                                                          BEF07860
      ENDIF                                                             BEF07870
C                                                                       BEF07880
C             READ DATA POINTS                                          BEF07890
C             ------------------------------------------------------    BEF07900
      READ (12,9000) NDATA                                              BEF07910
      CALL CKVAL1 (7,NDATA,1,N2)                                        BEF07920
      READ (12,8000) (ETA(I),I=1,NDATA)                                 BEF07930
 8000 FORMAT (5D15.6)                                                   BEF07940
 9000 FORMAT (I8)                                                       BEF07950
C                                                                       BEF07960
C             FIND EXTREME VALUES                                       BEF07970
C             ------------------------------------------------------    BEF07980
C             ETA = given time series of free surface profile at        BEF07990
C                   seaward boundary                                    BEF08000
C             ETAMAX,ETAMIN = maximum,minimum of ETA                    BEF08010
C             ------------------------------------------------------    BEF08020
      ETAMAX = -1.D+03                                                  BEF08030
      ETAMIN =  1.D+03                                                  BEF08040
      DO 100 I = 1,NDATA                                                BEF08050
        IF (ETA(I).GT.ETAMAX) ETAMAX=ETA(I)                             BEF08060
        IF (ETA(I).LT.ETAMIN) ETAMIN=ETA(I)                             BEF08070
  100 CONTINUE                                                          BEF08080
C                                                                       BEF08090
      RETURN                                                            BEF08100
      END                                                               BEF08110
C                                                                       BEF08120
C -03------------------  END OF SUBROUTINE INPUT2  ------------BEFORR-  BEF08130
C #04#######################  SUBROUTINE BOTTOM  ###############BEFORR#  BEF08140
C                                                                       BEF08150
C     This subroutine calculates normalized structure geometry and      BEF08160
C     delta x between two adjacent nodes                                BEF08170
C                                                                       BEF08180
      SUBROUTINE BOTTOM (MODE)                                          BEF08190
C                                                                       BEF08200
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF08210
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)              BEF08220
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                               BEF08230
      DIMENSION TSLOPE(N1)                                              BEF08240
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                           BEF08250
```

```
      COMMON /CONSTA/ PI,GRAV                                           BEF08260
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,               BEF08270
     +                ISAVA,ISAVB,ISAVC                                 BEF08280
      COMMON /CPAR3/  INITS,JE,JE1                                      BEF08290
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     BEF08300
      COMMON /WAVE1/  HREFP,TP,WL0P                                     BEF08310
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA          BEF08320
      COMMON /BOT1/   DSEAP,DLANDP,FWP                                  BEF08330
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT    BEF08340
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     BEF08350
      COMMON /BOT4/   NBSEG                                             BEF08360
      COMMON /BOT5/   WBSEG(N4),TBSLOP(N4),XBSEG(N4),ZBSEG(N4)          BEF08370
      COMMON /SAVBC/  NNODB,NNODC,NODB(N5),NODC(N5)                     BEF08380
      IF (MODE.EQ.0) THEN                                               BEF08390
        CALL CKFPAR (4,1,N1,N1R)                                        BEF08400
        CALL CKFPAR (4,4,N4,N4R)                                        BEF08410
        CALL CKFPAR (4,5,N5,N5R)                                        BEF08420
        RETURN                                                          BEF08430
      ENDIF                                                             BEF08440
C                                                                       BEF08450
C           COMPLETE SEGMENT DATA NOT SPECIFIED AS INPUT                BEF08460
C           (The following variables are dimensional)                   BEF08470
C           ------------------------------------------------------      BEF08480
C           TSLOPS = tangent of slope used to define                    BEF08490
C                    "surf similarity parameter"                        BEF08500
C           BSWL: . for IJOB<3: physical horizontal distance between    BEF08510
C                   seaward boundary and initial waterline on slope     BEF08520
C                 . for IJOB=3: physical horizontal distance between    BEF08530
C                   seaward and landward boundaries                     BEF08540
C           DSEAP = water depth below SWL at seaward boundary           BEF08550
C           The structure geometry is divided into segments of          BEF08560
C           different inclination                                       BEF08570
C           NBSEG = number of segments                                  BEF08580
C           For segment i starting from the seaward boundary:           BEF08590
C           . WBSEG(i)  = physical horizontal width                     BEF08600
C           . TBSLOP(i) = tangent of slope (+ upslope, - downslope)     BEF08610
C           . XBSEG(i)  = physical horizontal distance from seaward     BEF08620
C                         boundary to the segment's seaward-end         BEF08630
C           . ZBSEG(i)  = physical water depth below SWL                BEF08640
C                         (+ below SWL) at the segment's seaward-end    BEF08650
C           BSWL,DSEAP,WBSEG,XBSEG,ZBSEG are                            BEF08660
C           . in meters if ISYST=1 (SI)                                 BEF08670
C           . in feet   if ISYST=2 (USCS)                               BEF08680
C           ------------------------------------------------------      BEF08690
      IF (IBOT.EQ.1) THEN                                               BEF08700
        DCUM     = 0.D+00                                               BEF08710
        XBSEG(1) = 0.D+00                                               BEF08720
        ZBSEG(1) = DSEAP                                                BEF08730
        DO 110 K = 2,NBSEG+1                                            BEF08740
          DCUM     = DCUM + WBSEG(K-1)*TBSLOP(K-1)                      BEF08750
          XBSEG(K) = XBSEG(K-1) + WBSEG(K-1)                            BEF08760
          ZBSEG(K) = DSEAP - DCUM                                       BEF08770
  110   CONTINUE                                                        BEF08780
      ELSE                                                              BEF08790
        DO 120 K = 1,NBSEG                                              BEF08800
```

```
          TBSLOP(K) = -(ZBSEG(K+1)-ZBSEG(K))/(XBSEG(K+1)-XBSEG(K))      BEF08810
  120   CONTINUE                                                        BEF08820
      ENDIF                                                           BEF08830
C                                                                     BEF08840
C           CALCULATE GRID SPACING X BETWEEN TWO ADJACENT NODES       BEF08850
C           (dimensional)                                             BEF08860
C           ------------------------------------------------------    BEF08870
C           INITS = . number of spatial nodes along the bottom below  BEF08880
C                     SWL (IJOB<3)                                    BEF08890
C                   . number of nodes between seaward and landward    BEF08900
C                     boundaries, inclusive (IJOB=3)                  BEF08910
C           ------------------------------------------------------    BEF08920
      IF (IJOB.LT.3) THEN                                             BEF08930
        K = 0                                                         BEF08940
  910   CONTINUE                                                      BEF08950
          IF (K.EQ.NBSEG) CALL STOPP (11,12)                          BEF08960
          K = K+1                                                     BEF08970
          CROSS = ZBSEG(K)*ZBSEG(K+1)                                 BEF08980
        IF (CROSS.GT.0.D+00) GOTO 910                                 BEF08990
        BSWL = XBSEG(K+1) + ZBSEG(K+1)/TBSLOP(K)                      BEF09000
        X    = BSWL/DBLE(INITS)                                       BEF09010
      ELSE                                                            BEF09020
        BSWL = XBSEG(NBSEG+1)                                         BEF09030
        X    = BSWL/DBLE(INITS-1)                                     BEF09040
        DO 130 K = 1,NBSEG+1                                          BEF09050
          IF (ZBSEG(K).LT.0.D+00) CALL STOPP (13,14)                  BEF09060
  130   CONTINUE                                                      BEF09070
      ENDIF                                                           BEF09080
C                                                                     BEF09090
C           CALCULATE STRUCTURE GEOMETRY AT EACH NODE (dimensional)   BEF09100
C           ------------------------------------------------------    BEF09110
C           JE = landward edge node (IJOB<3) or landward boundary node BEF09120
C                (IJOB=3)                                             BEF09130
C           U2INIT(j) = water depth below SWL at node j (+ below SWL) BEF09140
C                     = total water depth U(2,j) at time t=0 (physical, BEF09150
C                       later normalized under the same name)         BEF09160
C           TSLOPE(j) = tangent of local slope at node j              BEF09170
C           ------------------------------------------------------    BEF09180
      IF (IJOB.LT.3) THEN                                             BEF09190
        DUM = XBSEG(NBSEG+1)/X                                        BEF09200
        JE  = INT(DUM)+1                                              BEF09210
      ELSE                                                            BEF09220
        JE = INITS                                                    BEF09230
      ENDIF                                                           BEF09240
      IF (JE.GT.N1) THEN                                              BEF09250
        WRITE  (*,9910) JE,N1                                         BEF09260
        WRITE (99,9910) JE,N1                                         BEF09270
        STOP                                                          BEF09280
      ELSE                                                            BEF09290
        JE1 = JE-1                                                    BEF09300
      ENDIF                                                           BEF09310
C     :: Some checks associated with storing time series              BEF09320
      IF (ISAVB.EQ.1) THEN                                            BEF09330
        DO 140 I = 1,NNODB                                            BEF09340
          CALL CKVAL1 (11,NODB(I),1,JE)                               BEF09350
```

```
  140   CONTINUE                                                        BEF09360
      ENDIF                                                           BEF09370
      IF (ISAVC.EQ.1) THEN                                            BEF09380
        DO 150 I = 1,NNODC                                            BEF09390
          CALL CKVAL1 (12,NODC(I),1,JE)                               BEF09400
  150   CONTINUE                                                        BEF09410
      ENDIF                                                           BEF09420
 9910 FORMAT (/' End Node =',I8,'; N1 =',I8/                          BEF09430
     +         ' Slope/Structure is too long.'/                       BEF09440
     +         ' Cut it, or change PARAMETER N1.')                    BEF09450
C                                                                     BEF09460
      DIST = -X                                                       BEF09470
      K    = 1                                                        BEF09480
      XCUM = XBSEG(K+1)                                               BEF09490
      DO 160 J = 1,JE                                                 BEF09500
        DIST = DIST + X                                               BEF09510
        IF (DIST.GT.XCUM.AND.K.LT.NBSEG) THEN                         BEF09520
  920     CONTINUE                                                    BEF09530
            K    = K+1                                                BEF09540
            XCUM = XBSEG(K+1)                                         BEF09550
          IF (DIST.GT.XCUM.AND.K.LT.NBSEG) GOTO 920                   BEF09560
        ENDIF                                                         BEF09570
        U2INIT(J) = ZBSEG(K) - (DIST-XBSEG(K))*TBSLOP(K)              BEF09580
        TSLOPE(J) = TBSLOP(K)                                         BEF09590
  160 CONTINUE                                                        BEF09600
C                                                                     BEF09610
C           NORMALIZATION                                             BEF09620
C           ------------------------------------------------------    BEF09630
C           WTOT = normalized width of computation domain             BEF09640
C           At node j:                                                BEF09650
C           . U2INIT(j) = normalized water depth below SWL            BEF09660
C                         (+ below SWL)                               BEF09670
C           . THETA(j)  = normalized tangent of local slope           BEF09680
C           . (XB(j),ZB(j)) = normalized coordinates of the structure BEF09690
C           ------------------------------------------------------    BEF09700
      DUM  = TP*DSQRT(GRAV*HREFP)                                     BEF09710
      X    = X/DUM                                                    BEF09720
      DIST = -X                                                       BEF09730
      WTOT = DBLE(JE1)*X                                              BEF09740
      DO 170 J = 1,JE                                                 BEF09750
        U2INIT(J) = U2INIT(J)/HREFP                                   BEF09760
        THETA(J)  = TSLOPE(J)*SIGMA                                   BEF09770
        DIST      = DIST + X                                          BEF09780
        XB(J)     = DIST                                              BEF09790
        ZB(J)     = -U2INIT(J)                                        BEF09800
  170 CONTINUE                                                        BEF09810
C                                                                     BEF09820
      RETURN                                                          BEF09830
      END                                                             BEF09840
C                                                                     BEF09850
C -04------------------  END OF SUBROUTINE BOTTOM  ------------BEFORR- BEF09860
C #05#######################  SUBROUTINE WPARAM  ###############BEFORR# BEF09870
C                                                                     BEF09880
C     This subroutine calculates wave parameters                      BEF09890
C                                                                     BEF09900
```

```
      SUBROUTINE WPARAM (MODE)                                          BEF09910
C                                                                       BEF09920
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF09930
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)              BEF09940
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                               BEF09950
      INTEGER S,SM1,SP1                                                 BEF09960
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                           BEF09970
      COMMON /CONSTA/ PI,GRAV                                           BEF09980
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,               BEF09990
     +                ISAVA,ISAVB,ISAVC                                 BEF10000
      COMMON /CPAR3/  INITS,JE,JE1                                      BEF10010
      COMMON /WAVE1/  HREFP,TP,WL0P                                     BEF10020
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA          BEF10030
      COMMON /BOT1/   DSEAP,DLANDP,FWP                                  BEF10040
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     BEF10050
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     BEF10060
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              BEF10070
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3)                              BEF10080
      IF (MODE.EQ.0) THEN                                               BEF10090
        CALL CKFPAR (5,1,N1,N1R)                                        BEF10100
        CALL CKFPAR (5,3,N3,N3R)                                        BEF10110
        RETURN                                                          BEF10120
      ENDIF                                                             BEF10130
C                                                                       BEF10140
C           PARAMETERS RELATED TO WAVE AND SLOPE CHARACTERISTICS        BEF10150
C           ----------------------------------------------------------  BEF10160
C           KSI    = surf similarity parameter                          BEF10170
C           WL0P   = physical deep-water wavelength                     BEF10180
C           DSEAP  = phys. water depth below SWL at seaward boundary    BEF10190
C           DLANDP = phys. water depth below SWL at landward boundary   BEF10200
C                    (only for IJOB=3)                                  BEF10210
C           FWP    = physical slope friction factor                     BEF10220
C           DELRP  = water depths associated with visual or measured    BEF10230
C                    waterline                                          BEF10240
C           WL0,DSEA,DLAND,FW,DELTAR are the normalized counterparts    BEF10250
C           of WL0P,DSEAP,DLANDP,FWP,DELRP, respectively                BEF10260
C           ----------------------------------------------------------  BEF10270
      WL0P   = GRAV*(TP*TP)/(2.D+00*PI)                                 BEF10280
      WL0    = WL0P/DSEAP                                               BEF10290
      KS     = KSSEA/KSREF                                              BEF10300
      KSI    = SIGMA*TSLOPS/DSQRT(2.D+00*PI)                            BEF10310
      DSEA   = DSEAP/HREFP                                              BEF10320
      DSEAKS = DSEA/KS                                                  BEF10330
      DSEA2  = DSQRT(DSEA)                                              BEF10340
      FW     = .5D+00*FWP*SIGMA                                         BEF10350
      DO 110 L = 1,NDELR                                                BEF10360
        IF (ISYST.EQ.1) THEN                                            BEF10370
          DELTAR(L) = DELRP(L)/(1.D+03*HREFP)                           BEF10380
        ELSE                                                            BEF10390
          DELTAR(L) = DELRP(L)/(12.D+00*HREFP)                          BEF10400
        ENDIF                                                           BEF10410
  110 CONTINUE                                                          BEF10420
      IF (IJOB.EQ.3) THEN                                               BEF10430
        DLAND  = U2INIT(INITS)                                          BEF10440
        DLANDP = DLAND*HREFP                                            BEF10450
```

```
        DLAND2 = DSQRT(DLAND)                                              BEF10460
      ENDIF                                                                BEF10470
C                                                                          BEF10480
C           LINEAR WAVELENGTH AND PRELIMINARY URSELL NUMBER                BEF10490
C           -----------------------------------------------------          BEF10500
C           WL = normalized linear wavelength at seaward boundary          BEF10510
C           UR = Ursell number at seaward boundary based on linear         BEF10520
C                wavelength                                                BEF10530
C           -----------------------------------------------------          BEF10540
      TWOPI = 2.D+00*PI                                                    BEF10550
      WL    = WL0                                                          BEF10560
      FUN1  = WL - WL0*DTANH(TWOPI/WL)                                     BEF10570
  900 IF (DABS(FUN1).GT.1.D-04) THEN                                       BEF10580
        FUN2 = 1.D+00 + WL0*TWOPI/(WL*DCOSH(TWOPI/WL))**2                  BEF10590
        WL   = WL - FUN1/FUN2                                              BEF10600
        FUN1 = WL - WL0*DTANH(TWOPI/WL)                                    BEF10610
      GOTO 900                                                             BEF10620
      ENDIF                                                                BEF10630
      UR    = WL*WL/DSEAKS                                                 BEF10640
      URPRE = UR                                                           BEF10650
C                                                                          BEF10660
      RETURN                                                               BEF10670
      END                                                                  BEF10680
C                                                                          BEF10690
C -05------------------  END OF SUBROUTINE WPARAM  ------------BEFORR-     BEF10700
C #06#######################  SUBROUTINE REGWAV  ###############BEFORR#   BEF10710
C                                                                          BEF10720
C     This subroutine computes incident regular wave profile at            BEF10730
C     seaward boundary                                                     BEF10740
C     Wave profile: Stokes II if UR<26                                     BEF10750
C                   Cnoidal otherwise                                      BEF10760
C                                                                          BEF10770
      SUBROUTINE REGWAV (MODE)                                             BEF10780
C                                                                          BEF10790
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                  BEF10800
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                 BEF10810
      DOUBLE PRECISION KCNO,MCNO,KC2,KC,KS,KSREF,KSSEA,KSI                BEF10820
      DIMENSION ETAU(N2)                                                   BEF10830
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                              BEF10840
      COMMON /CONSTA/ PI,GRAV                                              BEF10850
      COMMON /CPAR2/  MULTIF,NONEM,NONE,NEND,NRATE                         BEF10860
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX        BEF10870
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA             BEF10880
      COMMON /WAVE3/  ETA(N2)                                              BEF10890
      COMMON /WAVE4/  ETAMAX,ETAMIN                                        BEF10900
      COMMON /WAVE5/  KCNO,ECNO,MCNO,KC2                                   BEF10910
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT       BEF10920
      IF (MODE.EQ.0) THEN                                                  BEF10930
        CALL CKFPAR (6,2,N2,N2R)                                           BEF10940
        RETURN                                                             BEF10950
      ENDIF                                                                BEF10960
C                                                                          BEF10970
C           CHECK DIMENSION N2                                             BEF10980
C           -----------------------------------------------------          BEF10990
      IF (NONE.GT.(N2-1)) THEN                                             BEF11000
```

```
        WRITE  (*,9910) N2,NONE,TIME                                    BEF11010
        WRITE (99,9910) N2,NONE,TIME                                    BEF11020
        STOP                                                            BEF11030
      ENDIF                                                             BEF11040
 9910 FORMAT (/' From Subr. 06 REGWAV:'/                                BEF11050
     +       ' Specify larger dimension N2 to facilitate computation'/  BEF11060
     +       ' of incident regular wave profile at seaward boundary.'/  BEF11070
     +       ' N2 must be greater than NONE.'/                          BEF11080
     +       ' Currently,              N2 =,I8'/                        BEF11090
     +       ' Current try requires NONE =,I8'/                         BEF11100
     +       ' Time of occurrence   TIME =,F15.6'/                      BEF11110
     +       ' Provide enough space for possible increase in NONE.'/    BEF11120
     +       ' Computation aborted.')                                   BEF11130
C                                                                       BEF11140
C           CONSTANTS                                                   BEF11150
C           ------------------------------------------------------      BEF11160
      TWOPI  = 2.D+00*PI                                                BEF11170
      FOURPI = 4.D+00*PI                                                BEF11180
      HALFPI = PI/2.D+00                                                BEF11190
      NONE1  = NONE+1                                                   BEF11200
      NHALF  = NONE/2                                                   BEF11210
      NHALF1 = NHALF+1                                                  BEF11220
C                                                                       BEF11230
C           COMPUTE HALF OF WAVE PROFILE (unadjusted)                   BEF11240
C           ------------------------------------------------------      BEF11250
C           ETAMAX = normalized maximum surface elevation               BEF11260
C           ETAMIN = normalized minimum surface elevation               BEF11270
C           ETAU = unadjusted surface elevation                         BEF11280
C           N0 = approx. time level at which surface elevation is zero  BEF11290
C           UR based on linear wave theory is used in the following     BEF11300
C              criterion                                                BEF11310
C           ------------------------------------------------------      BEF11320
      IF (UR.LT.26.D+00) THEN                                           BEF11330
C           ::    Stokes II Wave Profile                                BEF11340
        ARG  = TWOPI/WL                                                 BEF11350
        ARG2 = 2.D+00*ARG                                               BEF11360
        DUM  = 16.D+00*DSEAKS*DSINH(ARG)**3.D+00                        BEF11370
        AMP2 = ARG*DCOSH(ARG)*(2.D+00+DCOSH(ARG2))/DUM                  BEF11380
        N0   = 1                                                        BEF11390
        DO 110 N = 1,NHALF1                                             BEF11400
          TIMEN = DBLE(N-1)*T                                           BEF11410
          ETAU(N) = .5D+00*DCOS(TWOPI*TIMEN)+AMP2*DCOS(FOURPI*TIMEN)    BEF11420
          ETAU(N) = KS*ETAU(N)                                          BEF11430
          IF (N.GT.1) THEN                                              BEF11440
            IF(ETAU(N).LE.0.D+00.AND.ETAU(N-1).GT.0.D+00) N0=N          BEF11450
          ENDIF                                                         BEF11460
  110   CONTINUE                                                        BEF11470
        ETAMIN = ETAU(NHALF1)                                           BEF11480
        ETAMAX = ETAU(1)                                                BEF11490
      ELSE                                                              BEF11500
C           ::    Cnoidal Wave Profile                                  BEF11510
C                 FINDM is to find the parameter M of the Jacobian      BEF11520
C                 elliptic func. See Func. 08 CEL and Subr. 09 SNCNDN   BEF11530
        CALL FINDM (MCNO)                                               BEF11540
        KC2    = 1.D+00-MCNO                                            BEF11550
```

```
      KC     = DSQRT(KC2)                                                BEF11560
      KCNO   = CEL(KC,1.D+00,1.D+00,1.D+00)                              BEF11570
      ECNO   = CEL(KC,1.D+00,1.D+00,KC2)                                 BEF11580
      UR     = 16.D+00*MCNO*KCNO*KCNO/3.D+00                             BEF11590
      WL     = DSQRT(UR*DSEAKS)                                          BEF11600
      ETAMIN = (1.D+00-ECNO/KCNO)/MCNO - 1.D+00                          BEF11610
      ETAMIN = KS*ETAMIN                                                 BEF11620
      ETAMAX = ETAMIN + KS                                               BEF11630
      N0     = 1                                                         BEF11640
      DO 120 N = 1,NHALF1                                                BEF11650
        TIMEN = DBLE(N-1)*T                                              BEF11660
        TETA  = 2.D+00*KCNO*TIMEN                                        BEF11670
        CALL SNCNDN (TETA,KC2,SNU,CNU,DNU)                               BEF11680
        ETAU(N) = ETAMIN + KS*CNU*CNU                                    BEF11690
        IF (N.GT.1) THEN                                                 BEF11700
          IF (ETAU(N).LE.0.D+00.AND.ETAU(N-1).GT.0.D+00) N0=N            BEF11710
        ENDIF                                                            BEF11720
  120   CONTINUE                                                         BEF11730
      ETAU(NHALF1) = ETAMIN                                              BEF11740
C                                                                        BEF11750
      ENDIF                                                              BEF11760
C           THE OTHER HALF OF WAVE PROFILE                               BEF11770
C           ------------------------------------------------------       BEF11780
      DO 130 N = NHALF+2,NONE1                                           BEF11790
        ETAU(N) = ETAU(NONE+2-N)                                         BEF11800
  130 CONTINUE                                                           BEF11810
C                                                                        BEF11820
C           ADJUST WAVE PROFILE so that elevation=0 at time=0 and        BEF11830
C           decreases initially with time                                BEF11840
C           ------------------------------------------------------       BEF11850
C           ETAU = unadjusted surface elevation                          BEF11860
C           ETA  = adjusted surface elevation                            BEF11870
C           ------------------------------------------------------       BEF11880
      NMARK = NONE-N0+2                                                  BEF11890
      DO 140 N = 1,NONE1                                                 BEF11900
        IF (N.LE.NMARK) THEN                                             BEF11910
          ETA(N) = ETAU(N+N0-1)                                          BEF11920
        ELSE                                                             BEF11930
          ETA(N) = ETAU(N-NMARK+1)                                       BEF11940
        ENDIF                                                            BEF11950
  140 CONTINUE                                                           BEF11960
C                                                                        BEF11970
      RETURN                                                             BEF11980
      END                                                                BEF11990
C                                                                        BEF12000
C -06-----------------  END OF SUBROUTINE REGWAV  -----------BEFORR-      BEF12010
C #07#####################  SUBROUTINE FINDM  ################BEFORR#   BEF12020
C                                                                        BEF12030
C     This subroutine computes the parameter m (MLIL<m<MBIG) of the      BEF12040
C     Jacobian elliptic functions (m=MCNO in this program)               BEF12050
C                                                                        BEF12060
      SUBROUTINE FINDM (MCNO)                                            BEF12070
C                                                                        BEF12080
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                BEF12090
      DOUBLE PRECISION KCNO,MCNO,KC2,KC,MSAV,MLIL,MBIG                   BEF12100
```

```
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                                BEF12110
      COMMON /CONSTA/ PI,GRAV                                            BEF12120
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA           BEF12130
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT      BEF12140
      SMALL = 1.D-07                                                     BEF12150
      MLIL  = .8D+00                                                     BEF12160
      INDI  = 0                                                          BEF12170
      I     = 0                                                          BEF12180
      SIGDT = DSQRT(2.D+00*PI*WL0)                                       BEF12190
      MBIG  = 1.00D+00 - 1.00D-15                                        BEF12200
      MCNO  =  .95D+00                                                   BEF12210
  900 CONTINUE                                                           BEF12220
        I    = I+1                                                       BEF12230
        MSAV = MCNO                                                      BEF12240
        KC2  = 1.D+00-MCNO                                               BEF12250
        KC   = DSQRT(KC2)                                                BEF12260
        KCNO = CEL(KC,1.D+00,1.D+00,1.D+00)                              BEF12270
        ECNO = CEL(KC,1.D+00,1.D+00,KC2)                                 BEF12280
        UR   = 16.D+00*MCNO*KCNO*KCNO/3.D+00                             BEF12290
        WL   = DSQRT(UR*DSEAKS)                                          BEF12300
        FCNO = 1.D+00 + (-MCNO+2.D+00-3.D+00*ECNO/KCNO)/(MCNO*DSEAKS)    BEF12310
        FCNO = SIGDT*DSQRT(FCNO)/WL - 1.D+00                             BEF12320
        IF (FCNO.LT.0.D+00) THEN                                         BEF12330
          MBIG = MCNO                                                    BEF12340
        ELSEIF (FCNO.GT.0.D+00) THEN                                     BEF12350
          MLIL = MCNO                                                    BEF12360
        ELSE                                                             BEF12370
          RETURN                                                         BEF12380
        ENDIF                                                            BEF12390
        MCNO = (MLIL+MBIG)/2.D+00                                        BEF12400
        DIF  = DABS(MSAV-MCNO)                                           BEF12410
        IF (DIF.LT.SMALL) RETURN                                         BEF12420
        IF (INDI.EQ.0) THEN                                              BEF12430
          IF (I.EQ.50) THEN                                              BEF12440
            SMALL = 1.D-13                                               BEF12450
            INDI  = 1                                                    BEF12460
          ELSE                                                           BEF12470
            IF (MCNO.GT..9999D+00) THEN                                  BEF12480
              SMALL = 1.D-13                                             BEF12490
              INDI  = 1                                                  BEF12500
            ENDIF                                                        BEF12510
          ENDIF                                                          BEF12520
        ENDIF                                                            BEF12530
      IF (I.LT.100) GOTO 900                                             BEF12540
      WRITE  (*,9910)                                                    BEF12550
      WRITE (99,9910)                                                    BEF12560
 9910 FORMAT (/' From Subr. 07 FINDM:'/                                  BEF12570
     +        ' Criterion for parameter m=MCNO not satisfied')           BEF12580
C                                                                        BEF12590
      RETURN                                                             BEF12600
      END                                                                BEF12610
C                                                                        BEF12620
C -07------------------  END OF SUBROUTINE FINDM  ------------BEFORR-    BEF12630
C #08#################  DOUBLE PRECISION FUNCTION CEL  #########BEFORR#  BEF12640
C                                                                        BEF12650
```

```
C     This function computes the general complete elliptic integral,     BEF12660
C     and is a double precision version of the "Function CEL" from       BEF12670
C     the book:                                                          BEF12680
C       William H. Press et al.                                          BEF12690
C       Numerical Recipes: The Art of Scientific Computing.              BEF12700
C       Cambridge University Press, New York, 1986.                      BEF12710
C       Pages 187-188.                                                   BEF12720
C                                                                        BEF12730
      DOUBLE PRECISION FUNCTION CEL (QQC,PP,AA,BB)                       BEF12740
C                                                                        BEF12750
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                BEF12760
      PARAMETER (CA=1.D-06,PIO2=1.5707963268D+00)                        BEF12770
      IF (QQC.EQ.0.D+00) THEN                                            BEF12780
        WRITE  (*,*) 'Failure in Function CEL.'                          BEF12790
        WRITE (99,*) 'Failure in Function CEL.'                          BEF12800
        STOP                                                             BEF12810
      ENDIF                                                              BEF12820
      QC = DABS(QQC)                                                     BEF12830
      A  = AA                                                            BEF12840
      B  = BB                                                            BEF12850
      P  = PP                                                            BEF12860
      E  = QC                                                            BEF12870
      EM = 1.D+00                                                        BEF12880
      IF (P.GT.0.D+00) THEN                                              BEF12890
        P = DSQRT(P)                                                     BEF12900
        B = B/P                                                          BEF12910
      ELSE                                                               BEF12920
        F = QC*QC                                                        BEF12930
        Q = 1.D+00-F                                                     BEF12940
        G = 1.D+00-P                                                     BEF12950
        F = F-P                                                          BEF12960
        Q = Q*(B-A*P)                                                    BEF12970
        P = DSQRT(F/G)                                                   BEF12980
        A = (A-B)/G                                                      BEF12990
        B = -Q/(G*G*P)+A*P                                               BEF13000
      ENDIF                                                              BEF13010
  900 F  = A                                                             BEF13020
      A  = A+B/P                                                         BEF13030
      G  = E/P                                                           BEF13040
      B  = B+F*G                                                         BEF13050
      B  = B+B                                                           BEF13060
      P  = G+P                                                           BEF13070
      G  = EM                                                            BEF13080
      EM = QC+EM                                                         BEF13090
      IF (DABS(G-QC).GT.G*CA) THEN                                       BEF13100
        QC = DSQRT(E)                                                    BEF13110
        QC = QC+QC                                                       BEF13120
        E  = QC*EM                                                       BEF13130
        GOTO 900                                                         BEF13140
      ENDIF                                                              BEF13150
      CEL = PIO2*(B+A*EM)/(EM*(EM+P))                                    BEF13160
C                                                                        BEF13170
      RETURN                                                             BEF13180
      END                                                                BEF13190
C                                                                        BEF13200
```

```
C -08------------  END OF DOUBLE PRECISION FUNCTION CEL  ------BEFORR-  BEF13210
C #09######################  SUBROUTINE SNCNDN  ###############BEFORR#  BEF13220
C                                                                       BEF13230
C     This subroutine computes the Jacobian elliptic functions,         BEF13240
C     and is a double precision version of the "Subroutine SNCNDN"      BEF13250
C     from the book:                                                    BEF13260
C       William H. Press et al.                                         BEF13270
C       Numerical Recipes: The Art of Scientific Computing.             BEF13280
C       Cambridge University Press, New York, 1986.                     BEF13290
C       Page 189.                                                       BEF13300
C                                                                       BEF13310
      SUBROUTINE SNCNDN (UU,EMMC,SN,CN,DN)                              BEF13320
C                                                                       BEF13330
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF13340
      PARAMETER (CA=1.D-06)                                             BEF13350
      DIMENSION EM(13),EN(13)                                           BEF13360
      LOGICAL BO                                                        BEF13370
      D   = 0.D+00                                                      BEF13380
      EMC = EMMC                                                        BEF13390
      U   = UU                                                          BEF13400
      IF (EMC.NE.0.D+00) THEN                                           BEF13410
        BO = (EMC.LT.0.D+00)                                            BEF13420
        IF (BO) THEN                                                    BEF13430
          D   = 1.D+00-EMC                                              BEF13440
          EMC = -EMC/D                                                  BEF13450
          D   = DSQRT(D)                                                BEF13460
          U   = D*U                                                     BEF13470
        ENDIF                                                           BEF13480
        A  = 1.D+00                                                     BEF13490
        DN = 1.D+00                                                     BEF13500
        DO 110 I = 1,13                                                 BEF13510
          L = I                                                         BEF13520
          EM(I) = A                                                     BEF13530
          EMC   = DSQRT(EMC)                                            BEF13540
          EN(I) = EMC                                                   BEF13550
          C     = .5D+00*(A+EMC)                                        BEF13560
          IF (DABS(A-EMC).LE.CA*A) GOTO 910                             BEF13570
          EMC = A*EMC                                                   BEF13580
          A = C                                                         BEF13590
  110   CONTINUE                                                        BEF13600
  910   U  = C*U                                                        BEF13610
        SN = DSIN(U)                                                    BEF13620
        CN = DCOS(U)                                                    BEF13630
        IF (SN.EQ.0.D+00) GOTO 920                                      BEF13640
        A = CN/SN                                                       BEF13650
        C = A*C                                                         BEF13660
        DO 120 II = L,1,-1                                              BEF13670
          B  = EM(II)                                                   BEF13680
          A  = C*A                                                      BEF13690
          C  = DN*C                                                     BEF13700
          DN = (EN(II)+A)/(B+A)                                         BEF13710
          A  = C/B                                                      BEF13720
  120   CONTINUE                                                        BEF13730
        A = 1.D+00/DSQRT(C*C+1.D+00)                                    BEF13740
        IF (SN.LT.0.D+00) THEN                                          BEF13750
```

```
          SN = -A                                                          BEF13760
        ELSE                                                               BEF13770
          SN = A                                                           BEF13780
        ENDIF                                                              BEF13790
        CN = C*SN                                                          BEF13800
  920   IF (BO) THEN                                                       BEF13810
          IF (ABS(D).LT.1.D-10) CALL STOPP (15,15)                         BEF13820
          A  = DN                                                          BEF13830
          DN = CN                                                          BEF13840
          CN = A                                                           BEF13850
          SN = SN/D                                                        BEF13860
        ENDIF                                                              BEF13870
      ELSE                                                                 BEF13880
        CN = 1.D+00/DCOSH(U)                                               BEF13890
        DN = CN                                                            BEF13900
        SN = DTANH(U)                                                      BEF13910
      ENDIF                                                                BEF13920
C                                                                          BEF13930
      RETURN                                                               BEF13940
      END                                                                  BEF13950
C                                                                          BEF13960
C -09------------------  END OF SUBROUTINE SNCNDN  ------------BEFORR-     BEF13970
C #10#####################  SUBROUTINE CPARAM  ###############BEFORR#     BEF13980
C                                                                          BEF13990
C     This subroutine                                                      BEF14000
C     . calculates NONEM at the beginning of BEFORR computation            BEF14010
C     . assigns values to NONE and DELTA for each computation unit         BEF14020
C                                                                          BEF14030
      SUBROUTINE CPARAM (MODE,M,IPROB,U2STOP)                              BEF14040
C                                                                          BEF14050
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                  BEF14060
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,                  BEF14070
     +                ISAVA,ISAVB,ISAVC                                    BEF14080
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES               BEF14090
      COMMON /CPAR2/  MULTIF,NONEM,NONE,NEND,NRATE                         BEF14100
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX        BEF14110
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT        BEF14120
      COMMON /DIAGID/ INONE,ITRY,IREV,LEVEL                                BEF14130
C                                                                          BEF14140
C           CALCULATE NONEM                                                BEF14150
C           ---------------------------------------------------------      BEF14160
C         . NONEM = minimum allowable value of NONE                        BEF14170
C         . NONE  = number of time steps in one wave period                BEF14180
C         . NONE must:                                                     BEF14190
C           + satisfy a numerical stability criterion                      BEF14200
C           + be divisible by NRATE for storing purposes                   BEF14210
C           + be an even number, for regular waves (IWAVE=1) only          BEF14220
C         . Below, NDUM3 is the smallest value which satisfies the         BEF14230
C           above criteria. BEFORR could start examining a computation     BEF14240
C           unit with NONE=NDUM3, but this manner is not always            BEF14250
C           efficient. BEFORR will instead begin its diagnosis with        BEF14260
C                     NONE = MULTIF*NDUM3 = NONEM,                         BEF14270
C           where MULTIF is an integer, greater than or equal to unity,    BEF14280
C           specified by user. MULTIF may be increased from unity to       BEF14290
C           improve the temporal resolution of computed variations         BEF14300
```

```
C          . See description of INONE farther below                    BEF14310
C            ---------------------------------------------------------  BEF14320
       IF (MODE.EQ.0) THEN                                              BEF14330
         EPSI  = DMAX1(X1,X2)                                           BEF14340
         DUM1  = 1.D+00 + EPSI*EPSI/4.D+00                              BEF14350
         DUM2  = DSQRT(DUM1) - EPSI/2.D+00                              BEF14360
         DUM3  = (1.D+00+DSQRT(DSEA))/(DUM2*X)                          BEF14370
         NDUM1 = INT(DUM3) + 1                                          BEF14380
         NDUM2 = NDUM1/NRATE + 1                                        BEF14390
         NDUM3 = NDUM2*NRATE                                            BEF14400
         IF (IWAVE.EQ.1) THEN                                           BEF14410
           IDUM = MOD(NDUM3,2)                                          BEF14420
           IF (IDUM.NE.0) NDUM3=2*NDUM3                                 BEF14430
         ENDIF                                                          BEF14440
C        ::  Minimum allowable value of NONE is NONEM                   BEF14450
         NONEM = MULTIF*NDUM3                                           BEF14460
C        ::  Initial value NONE=NONEM serves a practical purpose        BEF14470
         NONE = NONEM                                                   BEF14480
         T    = 1.D+00/DBLE(NONE)                                       BEF14490
         IF (IWAVE.GT.1) INONE=0                                        BEF14500
         RETURN                                                         BEF14510
       ENDIF                                                            BEF14520
C                                                                       BEF14530
C            ASSIGN VALUES TO NONE AND DELTA                            BEF14540
C            NONE  = number of time steps in one wave period            BEF14550
C            DELTA = normalized water depth at computational waterline  BEF14560
C            ---------------------------------------------------------  BEF14570
C            Guidelines                                                 BEF14580
C            ---------------------------------------------------------  BEF14590
C          . General:                                                   BEF14600
C            1. At the very beginning:                                  BEF14610
C               . NONE=NONEM (see above)                                BEF14620
C               . An initial value of DELTA is supplied by user         BEF14630
C            2. Increase NONE by about 50% whenever IPROB>0             BEF14640
C          . Regular waves (IWAVE=1):                                   BEF14650
C            1. The value of DELTA supplied by user is not modified     BEF14660
C            2. At the beginning of a computation unit, use the         BEF14670
C               previous value of NONE                                  BEF14680
C          . Random waves (IWAVE>1):                                    BEF14690
C            1. The value of DELTA may vary during computation          BEF14700
C            2. At the beginning of a computation unit use NONE=        BEF14710
C               (IREV+1)*NONEM and DELTA from previous computation unit BEF14720
C            3. IPROB=2 may be fixed by decreasing DELTA, to a value    BEF14730
C               approximately 95% of the water depth at node (S-1),     BEF14740
C               without increasing NONE                                 BEF14750
C          . IREV = reverse indicator for random waves                  BEF14760
C               (see main program)                                      BEF14770
C            ---------------------------------------------------------  BEF14780
       IF (ITRY.EQ.1) THEN                                              BEF14790
         IF (M.EQ.1.OR.IWAVE.GT.1) NONE=(IREV+1)*NONEM                  BEF14800
       ELSE                                                             BEF14810
         CALL NSI (OMEGA,NONE)                                          BEF14820
         IF (OMEGA.GT.20) THEN                                          BEF14830
           WRITE  (*,9910) M,OMEGA,ITRY,IREV,NONE,DELTA                 BEF14840
           WRITE (99,9910) M,OMEGA,ITRY,IREV,NONE,DELTA                 BEF14850
```

```
          IPROB = 9                                                      BEF14860
          INONE = 0                                                      BEF14870
          IF (IWAVE.EQ.1) THEN                                           BEF14880
            WRITE  (*,9920)                                              BEF14890
            WRITE (99,9920)                                              BEF14900
            STOP                                                         BEF14910
          ELSE                                                           BEF14920
            WRITE  (*,9930)                                              BEF14930
            WRITE (99,9930)                                              BEF14940
            RETURN                                                       BEF14950
          ENDIF                                                          BEF14960
        ENDIF                                                            BEF14970
        IF (IWAVE.GT.1.AND.IPROB.EQ.2.AND.U2STOP.GT.0.D+00) THEN         BEF14980
          DUM1  = .95*U2STOP                                             BEF14990
          DUM2  = DUM1/.000001D+00                                       BEF15000
          IDUM  = INT(DUM2)+1                                            BEF15010
          DELTA = DBLE(IDUM)*.000001D+00                                 BEF15020
        ELSE                                                             BEF15030
          NDUM1 = NONE/(2*NRATE)                                         BEF15040
          NDUM2 = NDUM1*NRATE                                            BEF15050
          NONE  = NONE + NDUM2                                           BEF15060
        ENDIF                                                            BEF15070
      ENDIF                                                              BEF15080
 9910 FORMAT (/' From Subr. 10 CPARAM:'/                                 BEF15090
     +         ' Computation unit                  M =',I8/              BEF15100
     +         ' Numerical stability indicator > 20:'/                   BEF15110
     +         '                               OMEGA =',F11.2/           BEF15120
     +         ' Number of failed trials        ITRY =',I8/              BEF15130
     +         ' Reverse indicator              IREV =',I8/              BEF15140
     +         ' Last used:                     NONE =',I8/              BEF15150
     +         '                               DELTA =',F18.9)           BEF15160
 9920 FORMAT (' OMEGA is too large. Computation aborted.'/               BEF15170
     +         ' Suggested to input larger DELTA.')                      BEF15180
 9930 FORMAT (' Computation reversed to LEVEL=2 or LEVEL=3.')            BEF15190
C                                                                        BEF15200
C           INDICATOR FOR REGULAR WAVE COMPUTATION                       BEF15210
C           -------------------------------------------------------      BEF15220
C         . INONE indicates whether or not regular incident wave         BEF15230
C           profile needs to be re-computed after this routine returns   BEF15240
C           to the main program because of the increase of NONE          BEF15250
C         . INONE=1: regular wave profile re-computed                    BEF15260
C                =0: not re-computed                                     BEF15270
C         . For IWAVE>1, INONE=0 is set above when MODE=0                BEF15280
C         . For IWAVE=1, INONE=1 if NONE has been increased by this      BEF15290
C           routine; otherwise INONE=0                                   BEF15300
C           -------------------------------------------------------      BEF15310
      IF (IWAVE.EQ.1) THEN                                               BEF15320
        IF (ITRY.GT.1) THEN                                              BEF15330
          INONE = 1                                                      BEF15340
        ELSE                                                             BEF15350
          INONE = 0                                                      BEF15360
        ENDIF                                                            BEF15370
      ENDIF                                                              BEF15380
C                                                                        BEF15390
C           DETERMINE NEND                                               BEF15400
```

```
C           NEND = number of time steps for this computation unit       BEF15410
C           NEND = NONE, except for the last unit MWAVE that may be     BEF15420
C                  less than one wave period                            BEF15430
C           ------------------------------------------------------------ BEF15440
      IF (M.LT.MWAVE) THEN                                              BEF15450
        NEND = NONE                                                     BEF15460
      ELSE                                                              BEF15470
        RES = DMOD(TMAX,1.D+00)                                         BEF15480
        IF (DABS(RES).LT.1.D-08) THEN                                   BEF15490
          NEND = NONE                                                   BEF15500
        ELSE                                                            BEF15510
          DEND = RES*DBLE(NONE)                                         BEF15520
          NEND = INT(DEND)                                              BEF15530
        ENDIF                                                           BEF15540
      ENDIF                                                             BEF15550
C                                                                       BEF15560
C           CALCULATE COMPUTATION PARAMETERS                            BEF15570
C           which are dependent on NONE                                 BEF15580
C           ------------------------------------------------------------ BEF15590
C           T = constant time step                                      BEF15600
C           X = constant grid spacing between two adjacent nodes        BEF15610
C           ------------------------------------------------------------ BEF15620
      T    = 1.D+00/DBLE(NONE)                                          BEF15630
      TX   = T/X                                                        BEF15640
      XT   = X/T                                                        BEF15650
      TTX  = T*T/X                                                      BEF15660
      TTXX = T*T/(X*X)                                                  BEF15670
      TWOX = 2.D+00*X                                                   BEF15680
C                                                                       BEF15690
C           INITIALIZE IPROB                                            BEF15700
C           for each trial                                              BEF15710
C           ------------------------------------------------------------ BEF15720
      IPROB = 0                                                         BEF15730
C                                                                       BEF15740
      RETURN                                                            BEF15750
      END                                                               BEF15760
C                                                                       BEF15770
C -10------------------  END OF SUBROUTINE CPARAM  ------------BEFORR-  BEF15780
C #11#####################  SUBROUTINE NUMSTA  ###############BEFORR#  BEF15790
C                                                                       BEF15800
C     This subroutine checks if numerical stability criterion is        BEF15810
C     violated (IPROB=3)                                                BEF15820
C                                                                       BEF15830
      SUBROUTINE NUMSTA (MODE,M,IPROB)                                  BEF15840
C                                                                       BEF15850
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF15860
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)              BEF15870
      INTEGER S,SM1,SP1                                                 BEF15880
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                           BEF15890
      COMMON /CPAR2/  MULTIF,NONEM,NONE,NEND,NRATE                      BEF15900
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     BEF15910
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             BEF15920
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              BEF15930
      COMMON /DIAGID/ INONE,ITRY,IREV,LEVEL                             BEF15940
      IF (MODE.EQ.0) THEN                                               BEF15950
```

```
        CALL CKFPAR (11,1,N1,N1R)                                       BEF15960
        RETURN                                                          BEF15970
      ENDIF                                                             BEF15980
C           ------------------------------------------------------      BEF15990
C           T = time step; X = spatial grid size; XT = X/T              BEF16000
C           ------------------------------------------------------      BEF16010
      ICOUNT = 0                                                        BEF16020
      DO 100 J = 1,S                                                    BEF16030
        IF (DABS(V(J)).GT.XT) ICOUNT=ICOUNT+1                           BEF16040
  100 CONTINUE                                                          BEF16050
C                                                                       BEF16060
      IF (ICOUNT.GT.0) THEN                                             BEF16070
        IPROB=3                                                         BEF16080
        CALL NSI (OMEGA,NONE)                                           BEF16090
        WRITE  (*,9910) M,S,NONE,OMEGA,TIME,ICOUNT,ITRY,IREV            BEF16100
        WRITE (99,9910) M,S,NONE,OMEGA,TIME,ICOUNT,ITRY,IREV            BEF16110
      ENDIF                                                             BEF16120
 9910 FORMAT (/' From Subr. 11 NUMSTA: IPROB=3'/                        BEF16130
     +        ' (Stability criterion violated)'/                        BEF16140
     +        ' Computation unit              M =',I8/                  BEF16150
     +        ' Waterline node                S =',I8/                  BEF16160
     +        ' Computation parameters:    NONE =',I8/                  BEF16170
     +        '                           OMEGA =',F11.2/              BEF16180
     +        ' Time of occurrence         TIME =',F18.9/              BEF16190
     +        ' Number of occurrence     ICOUNT =',I8/                 BEF16200
     +        ' Indicators            ITRY,IREV =',I4,',',I3)           BEF16210
C                                                                       BEF16220
      RETURN                                                            BEF16230
      END                                                               BEF16240
C                                                                       BEF16250
C -11------------------  END OF SUBROUTINE NUMSTA  ------------BEFORR-  BEF16260
C #12#####################  SUBROUTINE EXTRAP  ###############BEFORR#  BEF16270
C                                                                       BEF16280
C     This subroutine estimates U(2,SP1) and V(SP1) with SP1=(S+1)      BEF16290
C     by extrapolation                                                  BEF16300
C                                                                       BEF16310
C     S = most landward node at present time level (N-1)                BEF16320
C     The following values at node j are known at time level (N-1)      BEF16330
C     . U(1,j) = volume flux                                            BEF16340
C     . U(2,j) = total water depth                                      BEF16350
C     . V(j)   = depth-averaged velocity                                BEF16360
C                                                                       BEF16370
      SUBROUTINE EXTRAP (MODE)                                          BEF16380
C                                                                       BEF16390
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF16400
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)              BEF16410
      INTEGER S,SM1,SP1                                                 BEF16420
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                           BEF16430
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             BEF16440
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              BEF16450
      IF (MODE.EQ.0) THEN                                               BEF16460
        CALL CKFPAR (12,1,N1,N1R)                                       BEF16470
        RETURN                                                          BEF16480
      ENDIF                                                             BEF16490
      SM1 = S-1                                                         BEF16500
```

```
      SP1 = S+1                                                          BEF16510
      V(SP1)    = 2.D+00*V(S) - V(SM1)                                   BEF16520
      U(2,SP1) = 2.D+00*U(2,S) - U(2,SM1)                                BEF16530
      U(1,SP1) = U(2,SP1)*V(SP1)                                         BEF16540
      IF (U(2,SP1).GT.0.D+00) THEN                                       BEF16550
        C(SP1) = DSQRT(U(2,SP1))                                         BEF16560
      ELSE                                                               BEF16570
        C(SP1) = 0.D+00                                                  BEF16580
      ENDIF                                                              BEF16590
C                                                                        BEF16600
      RETURN                                                             BEF16610
      END                                                                BEF16620
C                                                                        BEF16630
C -12------------------  END OF SUBROUTINE EXTRAP  ------------BEFORR-   BEF16640
C #13#######################  SUBROUTINE RETAIN  ###############BEFORR#  BEF16650
C                                                                        BEF16660
C     This subroutine retains some values at present time level at       BEF16670
C     landward and seaward boundaries                                    BEF16680
C                                                                        BEF16690
      SUBROUTINE RETAIN (MODE)                                           BEF16700
C                                                                        BEF16710
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                BEF16720
      INTEGER S,SM1,SP1                                                  BEF16730
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)               BEF16740
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                            BEF16750
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)              BEF16760
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                               BEF16770
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                      BEF16780
      IF (MODE.EQ.0) THEN                                                BEF16790
        CALL CKFPAR (13,1,N1,N1R)                                        BEF16800
        RETURN                                                           BEF16810
      ENDIF                                                              BEF16820
      VSN    = V(S)                                                      BEF16830
      USN(1) = U(1,S)                                                    BEF16840
      USN(2) = U(2,S)                                                    BEF16850
      VMN    = V(SM1)                                                    BEF16860
      UMN(1) = U(1,SM1)                                                  BEF16870
      V1N    = V(1)                                                      BEF16880
      V2N    = V(2)                                                      BEF16890
C                                                                        BEF16900
      RETURN                                                             BEF16910
      END                                                                BEF16920
C                                                                        BEF16930
C -13------------------  END OF SUBROUTINE RETAIN  ------------BEFORR-   BEF16940
C #14#######################  SUBROUTINE CRITV  ###############BEFORR#  BEF16950
C                                                                        BEF16960
C     This subroutine calculates critical velocities used in             BEF16970
C     characteristic variables and checks for negative water depth       BEF16980
C     (IPROB=1)                                                          BEF16990
C                                                                        BEF17000
      SUBROUTINE CRITV (MODE,M,IPROB)                                    BEF17010
C                                                                        BEF17020
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                BEF17030
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)               BEF17040
      INTEGER S,SM1,SP1                                                  BEF17050
```

```
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                          BEF17060
      COMMON /CPAR2/  MULTIF,NONEM,NONE,NEND,NRATE                     BEF17070
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX    BEF17080
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)            BEF17090
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                             BEF17100
      COMMON /DIAGID/ INONE,ITRY,IREV,LEVEL                            BEF17110
      IF (MODE.EQ.0) THEN                                              BEF17120
        CALL CKFPAR (14,1,N1,N1R)                                      BEF17130
        RETURN                                                         BEF17140
      ENDIF                                                            BEF17150
      ICOUNT = 0                                                       BEF17160
      DO 100 J = 1,S                                                   BEF17170
        IF (U(2,J).LT.0.D+00) THEN                                     BEF17180
          ICOUNT = ICOUNT + 1                                          BEF17190
        ELSE                                                           BEF17200
          C(J) = DSQRT(U(2,J))                                         BEF17210
        ENDIF                                                          BEF17220
  100 CONTINUE                                                         BEF17230
      IF (ICOUNT.GT.0) THEN                                            BEF17240
        IPROB=1                                                        BEF17250
        CALL NSI (OMEGA,NONE)                                          BEF17260
        WRITE  (*,9910) M,S,NONE,OMEGA,TIME,ICOUNT,ITRY,IREV           BEF17270
        WRITE (99,9910) M,S,NONE,OMEGA,TIME,ICOUNT,ITRY,IREV           BEF17280
      ENDIF                                                            BEF17290
 9910 FORMAT (/' From Subr. 14 CRITV: IPROB=1'/                        BEF17300
     +        ' (Negative water depth)'/                               BEF17310
     +        ' Computation unit              M =',I8/                 BEF17320
     +        ' Waterline node                S =',I8/                 BEF17330
     +        ' Computation parameters:    NONE =',I8/                 BEF17340
     +        '                           OMEGA =',F11.2/              BEF17350
     +        ' Time of occurrence         TIME =',F18.9/              BEF17360
     +        ' Number of occurrence     ICOUNT =',I8/                 BEF17370
     +        ' Indicators            ITRY,IREV =',I4,',',I3)          BEF17380
C                                                                      BEF17390
      RETURN                                                           BEF17400
      END                                                              BEF17410
C                                                                      BEF17420
C -14-------------------  END OF SUBROUTINE CRITV  ------------BEFORR-  BEF17430
C #15#####################  SUBROUTINE MARCH  ################BEFORR#  BEF17440
C                                                                      BEF17450
C     This subroutine marches the computation from present time level  BEF17460
C     (N-1) to next time level N excluding seaward and landward        BEF17470
C     boundaries, which are treated separately                         BEF17480
C                                                                      BEF17490
      SUBROUTINE MARCH (MODE,M,IPROB,U2STOP)                           BEF17500
C                                                                      BEF17510
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                              BEF17520
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)             BEF17530
      INTEGER S,SM1,SP1                                                BEF17540
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                          BEF17550
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,              BEF17560
     +                ISAVA,ISAVB,ISAVC                                BEF17570
      COMMON /CPAR2/  MULTIF,NONEM,NONE,NEND,NRATE                     BEF17580
      COMMON /CPAR3/  INITS,JE,JE1                                     BEF17590
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX    BEF17600
```

```
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     BEF17610
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             BEF17620
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              BEF17630
      COMMON /DIAGID/ INONE,ITRY,IREV,LEVEL                             BEF17640
      IF (MODE.EQ.0) THEN                                               BEF17650
        CALL CKFPAR (15,1,N1,N1R)                                       BEF17660
        RETURN                                                          BEF17670
      ENDIF                                                             BEF17680
C                                                                       BEF17690
C           ------------------------------------------------------      BEF17700
C           U(1,j) and U(2,j) at time level N are computed as follows:  BEF17710
C           . at j=2,3,...,JDAM: WITH numerical damping term            BEF17720
C           . at j=(JDAM+1),(JDAM+2),...,JLAX: NO numer. damping term   BEF17730
C           JE  = landward edge node                                    BEF17740
C           JE1 = JE-1                                                  BEF17750
C           ------------------------------------------------------      BEF17760
      IF (IJOB.LT.3) THEN                                               BEF17770
        JDAM = S-2                                                      BEF17780
        JLAX = S                                                        BEF17790
        IF (IJOB.EQ.2.AND.S.EQ.JE) JLAX=JE1                             BEF17800
      ELSE                                                              BEF17810
        JDAM = JE1                                                      BEF17820
        JLAX = JE1                                                      BEF17830
      ENDIF                                                             BEF17840
      JLAX1 = JLAX+1                                                    BEF17850
C                                                                       BEF17860
C           COMPUTE ELEMENTS OF MATRICES                                BEF17870
C           ------------------------------------------------------      BEF17880
C         . Subr. 20 MATAFG computes non-constant elements of Matrices  BEF17890
C           A and G, and the elements of Matrix F                       BEF17900
C         . Subr. 22 MATS computes the first element of Matrix S        BEF17910
C         . Subr. 21 MATGJR, 23 MATD, and 24 MATU compute the elements  BEF17920
C           of Matrices g, D, and U, respectively                       BEF17930
C         . Subr. 24 MATU computes values of U at time level N using    BEF17940
C           the results obtained from the other four subroutines        BEF17950
C           ------------------------------------------------------      BEF17960
      CALL MATAFG (MODE,1,JLAX1)                                        BEF17970
      CALL MATGJR (MODE,1,JLAX)                                         BEF17980
      CALL MATS   (MODE,2,JLAX)                                         BEF17990
      CALL MATD   (MODE,JDAM,JLAX)                                      BEF18000
      CALL MATU   (MODE,2,JLAX)                                         BEF18010
C                                                                       BEF18020
C           IPROB=2: WATER DEPTH AT (S-1) <or= DELTA                    BEF18030
C           ------------------------------------------------------      BEF18040
      IF (U(2,SM1).LE.DELTA) THEN                                       BEF18050
        IPROB=2                                                         BEF18060
        U2STOP=U(2,SM1)                                                 BEF18070
        CALL NSI (OMEGA,NONE)                                           BEF18080
        WRITE  (*,9910) M,S,NONE,OMEGA,DELTA,U2STOP,TIME,ITRY,IREV      BEF18090
        WRITE (99,9910) M,S,NONE,OMEGA,DELTA,U2STOP,TIME,ITRY,IREV      BEF18100
        RETURN                                                          BEF18110
      ENDIF                                                             BEF18120
 9910 FORMAT (/' From Subr. 15 MARCH: IPROB=2'/                         BEF18130
     +        ' (Water depth at (S-1) <or= DELTA)'/                     BEF18140
     +        ' Computation unit              M =',I8/                  BEF18150
```

```
     +         ' Waterline node                S =',I8/                    BEF18160
     +         ' Computation parameters:   NONE =',I8/                     BEF18170
     +         '                          OMEGA =',F11.2/                  BEF18180
     +         '                          DELTA =',F18.9/                  BEF18190
     +         ' Water depth at (S-1)  U(2,S-1) =',F18.9/                  BEF18200
     +         ' Time of occurrence        TIME =',F18.9/                  BEF18210
     +         ' Indicators           ITRY,IREV =',I4,',',I3)              BEF18220
C                                                                          BEF18230
C          COMPLETE THE COMPUTATION OF HYDRODYNAMIC QUANTITIES             BEF18240
C          ---------------------------------------------------             BEF18250
C          Water depth h is taken to be not less than DELTA for            BEF18260
C          submerged structures                                            BEF18270
C          ---------------------------------------------------             BEF18280
      DO 100 J = 2,JLAX                                                    BEF18290
        IF (IJOB.EQ.3.AND.U(2,J).LT.DELTA) U(2,J)=DELTA                    BEF18300
        V(J)    = U(1,J)/U(2,J)                                            BEF18310
        ELEV(J) = U(2,J)-U2INIT(J)                                         BEF18320
  100 CONTINUE                                                             BEF18330
C                                                                          BEF18340
      RETURN                                                               BEF18350
      END                                                                  BEF18360
C                                                                          BEF18370
C -15------------------  END OF SUBROUTINE MARCH  ------------BEFORR-      BEF18380
C #16#####################  SUBROUTINE LANDBC  ###############BEFORR#     BEF18390
C                                                                          BEF18400
C     This subroutine manages the computation for                          BEF18410
C     landward boundary conditions                                         BEF18420
C                                                                          BEF18430
      SUBROUTINE LANDBC (MODE,M)                                           BEF18440
C                                                                          BEF18450
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                  BEF18460
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                 BEF18470
      INTEGER S,SP1,SM1                                                    BEF18480
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                              BEF18490
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,                  BEF18500
     +                ISAVA,ISAVB,ISAVC                                    BEF18510
      COMMON /CPAR3/  INITS,JE,JE1                                         BEF18520
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX        BEF18530
      COMMON /WAVE3/  ETA(N2)                                              BEF18540
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT       BEF18550
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)        BEF18560
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)                BEF18570
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                                 BEF18580
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                        BEF18590
      IF (MODE.EQ.0) THEN                                                  BEF18600
        CALL CKFPAR (16,1,N1,N1R)                                          BEF18610
        CALL CKFPAR (16,2,N2,N2R)                                          BEF18620
        RETURN                                                             BEF18630
      ENDIF                                                                BEF18640
C                                                                          BEF18650
C          MANAGE LANDWARD B.C.                                            BEF18660
C          ---------------------------------------------------             BEF18670
C        . Subr. 17 RUNUP computes shoreline movement if                   BEF18680
C          computational waterline is on the slope (IJOB<3)                BEF18690
C        . Subr. 18 OVERT computes hydrodynamic quantities at              BEF18700
```

```
C           landward edge node if overtopping occurs (IJOB=2)            BEF18710
C         . ALPHA = landward-advancing characteristics                   BEF18720
C         . ETAT  = surface elevation associated with transmitted wave   BEF18730
C           at landward boundary (IJOB=3)                                BEF18740
C         . DLAND = norm. water depth below SWL at landward boundary     BEF18750
C           (IJOB=3)                                                     BEF18760
C         . S used in Subr. 17 RUNUP is of present time level            BEF18770
C           ------------------------------------------------------       BEF18780
      IF (IJOB.EQ.1) THEN                                                BEF18790
        CALL RUNUP (MODE)                                                BEF18800
        IF (S.GT.JE1) THEN                                               BEF18810
          WRITE  (*,9910) TIME,S,JE                                      BEF18820
          WRITE (99,9910) TIME,S,JE                                      BEF18830
          STOP                                                           BEF18840
        ENDIF                                                            BEF18850
      ELSEIF (IJOB.EQ.2) THEN                                            BEF18860
        IF (S.LT.JE) THEN                                                BEF18870
          CALL RUNUP (MODE)                                              BEF18880
        ELSE                                                             BEF18890
          CALL OVERT (MODE)                                              BEF18900
        ENDIF                                                            BEF18910
      ELSE                                                               BEF18920
        DUM     = TX*(VSN+C(S))*(VSN-VMN+2.D+00*(C(S)-C(SM1)))           BEF18930
        ALPHA   = VSN+2.D+00*C(S) - DUM - T*THETA(S)                     BEF18940
        ETAT    = ALPHA*DLAND2/2.D+00 - DLAND                            BEF18950
        U(2,S)  = DLAND + ETAT                                           BEF18960
        V(S)    = ALPHA - 2.D+00*DSQRT(U(2,S))                           BEF18970
        U(1,S)  = U(2,S)*V(S)                                            BEF18980
        ELEV(S) = U(2,S) - U2INIT(S)                                     BEF18990
      ENDIF                                                              BEF19000
 9910 FORMAT (/' From Subr. 16 LANDBC:'/                                 BEF19010
     +   ' t =',F18.6,'; S =',I8,'; End Node =',I8/                      BEF19020
     +   ' Slope is not long enough to accomodate shoreline movement'/   BEF19030
     +   ' Specify longer slope or choose overtopping computation')      BEF19040
C                                                                        BEF19050
C           CONDITIONS LANDWARD OF NEW WATERLINE NODE S                  BEF19060
C           ------------------------------------------------------       BEF19070
      IF (IJOB.LT.3) THEN                                                BEF19080
        DO 100 L = S+1,JE                                                BEF19090
          U(1,L)  = 0.D+00                                               BEF19100
          U(2,L)  = 0.D+00                                               BEF19110
          V(L)    = 0.D+00                                               BEF19120
          ELEV(L) = ZB(L)                                                BEF19130
  100   CONTINUE                                                         BEF19140
      ENDIF                                                              BEF19150
C                                                                        BEF19160
      RETURN                                                             BEF19170
      END                                                                BEF19180
C                                                                        BEF19190
C -16------------------  END OF SUBROUTINE LANDBC  ------------BEFORR-   BEF19200
C #17#######################  SUBROUTINE RUNUP  ################BEFORR#  BEF19210
C                                                                        BEF19220
C     This subroutine computes waterline movement for IJOB=1 and if      BEF19230
C     no overtopping occurs for IJOB=2                                   BEF19240
C                                                                        BEF19250
```

```
      SUBROUTINE RUNUP (MODE)                                                BEF19260
C                                                                            BEF19270
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                    BEF19280
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                   BEF19290
      DIMENSION USN2(2),US1N1(2)                                             BEF19300
      INTEGER S,SM1,SP1,SNEW                                                 BEF19310
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                                BEF19320
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX          BEF19330
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)          BEF19340
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)                  BEF19350
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)       BEF19360
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                                   BEF19370
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                          BEF19380
      IF (MODE.EQ.0) THEN                                                    BEF19390
        CALL CKFPAR (17,1,N1,N1R)                                            BEF19400
        RETURN                                                               BEF19410
      ENDIF                                                                  BEF19420
C                                                                            BEF19430
C             ADJUST VALUES AT S IF U(2,S)>U(2,S-1)                          BEF19440
C             ------------------------------------------------------         BEF19450
      IF (U(2,S).GE.U(2,SM1)) THEN                                           BEF19460
        V(S)    = 2.D+00*V(SM1) - V(S-2)                                     BEF19470
        U(2,S)  = 2.D+00*U(2,SM1) - U(2,S-2)                                 BEF19480
        IF (ABS(V(S)).GT.ABS(V(SM1))) V(S)=.9*V(SM1)                         BEF19490
        IF (U(2,S).LT.0.D+00) U(2,S)=.5*U(2,SM1)                             BEF19500
        IF (U(2,S).GT.U(2,SM1)) U(2,S)=.9*U(2,SM1)                           BEF19510
        U(1,S)  = V(S)*U(2,S)                                                BEF19520
        ELEV(S) = U(2,S) - U2INIT(S)                                         BEF19530
      ENDIF                                                                  BEF19540
C                                                                            BEF19550
C             DETERMINE THE NEXT WATERLINE NODE                              BEF19560
C             ------------------------------------------------------         BEF19570
      IF (U(2,S).LE.DELTA) THEN                                              BEF19580
        SNEW = SM1                                                           BEF19590
      ELSE                                                                   BEF19600
        V(SP1)    = 2.D+00*V(S) - V(SM1)                                     BEF19610
        U(2,SP1) = 2.D+00*U(2,S) - U(2,SM1)                                  BEF19620
        U(1,SP1) = V(SP1)*U(2,SP1)                                           BEF19630
        IF (U(2,SP1).LE.DELTA) THEN                                          BEF19640
          SNEW = S                                                           BEF19650
        ELSE                                                                 BEF19660
C         ------------------------------------------------------------       BEF19670
C         ::      USN2(i),VSN2 = U(i,S) and V(S), respectively,              BEF19680
C                 at time level (N+1), i=1,2                                 BEF19690
          CALL MATAFG (MODE,SM1,SP1)                                         BEF19700
          CALL MATGJR (MODE,SM1,S)                                           BEF19710
          CALL MATS   (MODE,S,S)                                             BEF19720
          DUM1 = TX*((F(1,SP1)-F(1,SM1))/2.D+00+X*G1(S))                     BEF19730
          DUM2 = TTXX*(GJR(1,S)-GJR(1,SM1))                                  BEF19740
          DUM3 = TX*(F(2,SP1)-F(2,SM1))                                      BEF19750
          DUM4 = TTXX*(GJR(2,S)-GJR(2,SM1))                                  BEF19760
          USN2(1) = U(1,S)-DUM1+(DUM2-TTX*S1(S))/2.D+00                      BEF19770
          USN2(2) = U(2,S)-(DUM3-DUM4)/2.D+00                                BEF19780
          VSN2 = USN2(1)/USN2(2)                                             BEF19790
C         ------------------------------------------------------------       BEF19800
```

```
C         ::      US1N1(i),VS1N1 = U(i,S+1) and V(S+1), respectively,   BEF19810
C                 at time level N, i=1,2                                BEF19820
          VS = V(S)                                                     BEF19830
          IF (DABS(VS).LT.DELTA) VS=DSIGN(DELTA,VS)                     BEF19840
          DUM1  = (XT*(VSN2-VSN)+U(2,SP1)-U(2,SM1)+TWOX*THETA(S))/VS    BEF19850
          VS1N1 = V(SM1)-DUM1                                           BEF19860
          US1N1(1) = U(1,SM1) - XT*(USN2(2)-USN(2))                     BEF19870
          US1N1(2) = US1N1(1)/VS1N1                                     BEF19880
C         ------------------------------------------------------------  BEF19890
          IF (DABS(VS1N1).LE.DELTA) THEN                                BEF19900
            SNEW = S                                                    BEF19910
          ELSE                                                          BEF19920
            IF (US1N1(2).LE.U(2,S)) THEN                                BEF19930
              IF (US1N1(2).LE.DELTA) THEN                               BEF19940
                SNEW = S                                                BEF19950
              ELSE                                                      BEF19960
                SNEW   = SP1                                            BEF19970
                U(2,SP1) = US1N1(2)                                     BEF19980
                U(1,SP1) = US1N1(1)                                     BEF19990
                V(SP1)   = VS1N1                                        BEF20000
              ENDIF                                                     BEF20010
            ELSE                                                        BEF20020
              IF (U(2,SP1).LE.U(2,S)) THEN                              BEF20030
                SNEW = SP1                                              BEF20040
              ELSE                                                      BEF20050
                SNEW = S                                                BEF20060
              ENDIF                                                     BEF20070
            ENDIF                                                       BEF20080
          ENDIF                                                         BEF20090
        ENDIF                                                           BEF20100
      ENDIF                                                             BEF20110
      IF (SNEW.EQ.SP1) ELEV(SP1)=U(2,SP1)-U2INIT(SP1)                   BEF20120
      S = SNEW                                                          BEF20130
C                                                                       BEF20140
C     S at next time level has been found                               BEF20150
C                                                                       BEF20160
      RETURN                                                            BEF20170
      END                                                               BEF20180
C                                                                       BEF20190
C -17------------------  END OF SUBROUTINE RUNUP  ------------BEFORR-   BEF20200
C #18#####################  SUBROUTINE OVERT  ################BEFORR#   BEF20210
C                                                                       BEF20220
C     This subroutine computes quantities at landward-end node for      BEF20230
C     IJOB=2 if overtopping occurs, that is, S=JE and SM1=(S-1)=JE1     BEF20240
C                                                                       BEF20250
      SUBROUTINE OVERT (MODE)                                           BEF20260
C                                                                       BEF20270
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF20280
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)              BEF20290
      INTEGER S,SM1,SP1                                                 BEF20300
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                           BEF20310
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     BEF20320
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     BEF20330
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             BEF20340
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)  BEF20350
```

```
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                                  BEF20360
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                         BEF20370
      IF (MODE.EQ.0) THEN                                                   BEF20380
        CALL CKFPAR (18,1,N1,N1R)                                           BEF20390
        RETURN                                                              BEF20400
      ENDIF                                                                 BEF20410
      IF (VMN.GT.C(SM1)) THEN                                               BEF20420
        U(1,S) = USN(1) - TX*(F(1,S)-F(1,SM1)) - T*(THETA(S)*USN(2))        BEF20430
        U(2,S) = USN(2) - TX*(USN(1)-UMN(1))                                BEF20440
        V(S)   = U(1,S)/U(2,S)                                              BEF20450
      ELSE                                                                  BEF20460
        VCS  = VSN + 2.D+00*C(S)                                            BEF20470
        VCM  = VMN + 2.D+00*C(SM1)                                          BEF20480
        V(S) = (VCS-TX*(VSN+C(S))*(VCS-VCM)-T*(THETA(S)))/3.D+00            BEF20490
        U(2,S) = V(S)*V(S)                                                  BEF20500
        U(1,S) = V(S)*U(2,S)                                                BEF20510
      ENDIF                                                                 BEF20520
      IF (U(2,S).LE.DELTA) THEN                                             BEF20530
        S = SM1                                                             BEF20540
      ELSE                                                                  BEF20550
        ELEV(S) = U(2,S) - U2INIT(S)                                        BEF20560
      ENDIF                                                                 BEF20570
C                                                                           BEF20580
      RETURN                                                                BEF20590
      END                                                                   BEF20600
C                                                                           BEF20610
C -18------------------  END OF SUBROUTINE OVERT  ------------BEFORR-       BEF20620
C #19#####################  SUBROUTINE SEABC  ################BEFORR#      BEF20630
C                                                                           BEF20640
C     This subroutine treats seaward boundary conditions at node j=1        BEF20650
C                                                                           BEF20660
      SUBROUTINE SEABC (MODE,M,N)                                           BEF20670
C                                                                           BEF20680
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                   BEF20690
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                  BEF20700
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                                   BEF20710
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                               BEF20720
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,                   BEF20730
     +                ISAVA,ISAVB,ISAVC                                     BEF20740
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX         BEF20750
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA              BEF20760
      COMMON /WAVE3/  ETA(N2)                                               BEF20770
      COMMON /WAVE6/  NDATA                                                 BEF20780
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT         BEF20790
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)         BEF20800
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)                 BEF20810
      COMMON /VALUEN/ VSN,USN(2),VMN,UMN(1),V1N,V2N                         BEF20820
      IF (MODE.EQ.0) THEN                                                   BEF20830
        CALL CKFPAR (19,1,N1,N1R)                                           BEF20840
        CALL CKFPAR (19,2,N2,N2R)                                           BEF20850
        RETURN                                                              BEF20860
      ENDIF                                                                 BEF20870
C                                                                           BEF20880
C          ESTIMATE ETAR                                                    BEF20890
C          ---------------------------------------------------------        BEF20900
```

```
C         . BETA = seaward-advancing characteristics                    BEF20910
C         . ETAR = surface elevation due to reflected wave at           BEF20920
C           seaward boundary                                            BEF20930
C         . A correction term included in ETAR if INONCT=1 to improve   BEF20940
C           prediction of wave set-down and setup on beach              BEF20950
C           ------------------------------------------------------      BEF20960
      VC1 = -V1N+2.D+00*C(1)                                            BEF20970
      VC2 = -V2N+2.D+00*C(2)                                            BEF20980
      BETA = VC1 - TX*(V1N-C(1))*(VC2-VC1) + T*THETA(1)                 BEF20990
      ETAR = BETA*DSEA2/2.D+00 - DSEA                                   BEF21000
      IF (INONCT.EQ.1) ETAR=ETAR-KS*KS/(16.D+00*DSEA)                   BEF21010
C                                                                       BEF21020
C           VALUES AT NODE ONE                                          BEF21030
C           ------------------------------------------------------      BEF21040
      IF (IWAVE.EQ.1) THEN                                              BEF21050
C       :: Regular wave.                                                BEF21060
C          Surface elevation ETA has been computed by Subr. 06 REGWAV.  BEF21070
        ETAI    = ETA(N+1)                                              BEF21080
        U(2,1)  = DSEA+ETAR+ETAI                                        BEF21090
        ELEV(1) = ETAI+ETAR                                             BEF21100
      ELSE                                                              BEF21110
C       :: Incident or total wave train is specified by user.           BEF21120
C          The specified time series ETA has NDATA points from the      BEF21130
C          normalized time t=0 to t=TMAX.                               BEF21140
C          Interpolate the specified time series to obtain the value    BEF21150
C          at t=TIME.                                                   BEF21160
        DJJ  = DBLE(NDATA-1)*TIME/TMAX                                  BEF21170
        JJ   = INT(DJJ)                                                 BEF21180
        ETA1 = ETA(JJ+1)                                                BEF21190
        ETA2 = ETA(JJ+2)                                                BEF21200
        DEL  = DJJ - DBLE(JJ)                                           BEF21210
        ETAI = ETA1 + DEL*(ETA2-ETA1)                                   BEF21220
        IF (IWAVE.EQ.2) THEN                                            BEF21230
          U(2,1)  = DSEA+ETAI+ETAR                                      BEF21240
          ELEV(1) = ETAI+ETAR                                           BEF21250
        ELSE                                                            BEF21260
          U(2,1)  = DSEA+ETAI                                           BEF21270
          ELEV(1) = ETAI                                                BEF21280
        ENDIF                                                           BEF21290
      ENDIF                                                             BEF21300
      V(1)   = 2.D+00*DSQRT(U(2,1))-BETA                                BEF21310
      U(1,1) = U(2,1)*V(1)                                              BEF21320
C                                                                       BEF21330
      RETURN                                                            BEF21340
      END                                                               BEF21350
C                                                                       BEF21360
C -19------------------  END OF SUBROUTINE SEABC  -----------BEFORR-     BEF21370
C #20#####################  SUBROUTINE MATAFG  ##############BEFORR#    BEF21380
C                                                                       BEF21390
C     This subroutine computes, for each node,                          BEF21400
C     . the elements of the the first row of Matrix A (2x2)             BEF21410
C                                          --> A1(1,j) and A1(2,j)      BEF21420
C     . the elements of Matrix F (2x1)     --> F(1,j) and F(2,j)        BEF21430
C     . the first element of Matrix G (2x1) --> G1(j)                   BEF21440
C     j=node number                                                     BEF21450
```

```
C                                                                       BEF21460
      SUBROUTINE MATAFG (MODE,JBEGIN,JEND)                              BEF21470
C                                                                       BEF21480
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF21490
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)              BEF21500
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                           BEF21510
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT    BEF21520
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     BEF21530
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)             BEF21540
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)  BEF21550
      IF (MODE.EQ.0) THEN                                               BEF21560
        CALL CKFPAR (20,1,N1,N1R)                                       BEF21570
        RETURN                                                          BEF21580
      ENDIF                                                             BEF21590
      DO 100 J = JBEGIN,JEND                                            BEF21600
        A1(1,J) = 2.D+00*V(J)                                           BEF21610
        A1(2,J) = U(2,J)-V(J)*V(J)                                      BEF21620
        F(1,J)  = V(J)*U(1,J) + U(2,J)*U(2,J)/2.D+00                    BEF21630
        F(2,J)  = U(1,J)                                                BEF21640
        G1(J)   = THETA(J)*U(2,J) + FW*DABS(V(J))*V(J)                  BEF21650
  100 CONTINUE                                                          BEF21660
C                                                                       BEF21670
      RETURN                                                            BEF21680
      END                                                               BEF21690
C                                                                       BEF21700
C -20------------------  END OF SUBROUTINE MATAFG  ------------BEFORR-  BEF21710
C #21#######################  SUBROUTINE MATGJR  ###############BEFORR#  BEF21720
C                                                                       BEF21730
C     This subroutine computes, for each node, the elements of          BEF21740
C     Matrix g (2x1) --> GJR(1,j) and GJR(2,j), j=node number           BEF21750
C                                                                       BEF21760
      SUBROUTINE MATGJR (MODE,JBEGIN,JEND)                              BEF21770
C                                                                       BEF21780
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF21790
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)              BEF21800
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                           BEF21810
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     BEF21820
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)  BEF21830
      IF (MODE.EQ.0) THEN                                               BEF21840
        CALL CKFPAR (21,1,N1,N1R)                                       BEF21850
        RETURN                                                          BEF21860
      ENDIF                                                             BEF21870
C                                                                       BEF21880
      DO 100 J = JBEGIN,JEND                                            BEF21890
        FG1 = F(1,J+1)-F(1,J) + X*(G1(J+1)+G1(J))/2.D+00                BEF21900
        FG2 = F(2,J+1)-F(2,J)                                           BEF21910
        DUM = (A1(1,J+1)+A1(1,J))*FG1 + (A1(2,J+1)+A1(2,J))*FG2         BEF21920
        GJR(1,J) = DUM/2.D+00                                           BEF21930
        GJR(2,J) = FG1                                                  BEF21940
  100 CONTINUE                                                          BEF21950
C                                                                       BEF21960
      RETURN                                                            BEF21970
      END                                                               BEF21980
C                                                                       BEF21990
C -21------------------  END OF SUBROUTINE MATGJR  ------------BEFORR-  BEF22000
```

```
C #22########################  SUBROUTINE MATS  ################BEFORR#  BEF22010
C                                                                         BEF22020
C     This subroutine computes, for each node, the first element of       BEF22030
C     Matrix S (2x1) --> S1(j), j=node number                             BEF22040
C                                                                         BEF22050
      SUBROUTINE MATS (MODE,JBEGIN,JEND)                                  BEF22060
C                                                                         BEF22070
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                 BEF22080
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                BEF22090
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                             BEF22100
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX       BEF22110
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT       BEF22120
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)       BEF22130
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)               BEF22140
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)    BEF22150
      IF (MODE.EQ.0) THEN                                                 BEF22160
        CALL CKFPAR (22,1,N1,N1R)                                         BEF22170
        RETURN                                                            BEF22180
      ENDIF                                                               BEF22190
C                                                                         BEF22200
      DO 100 J = JBEGIN,JEND                                              BEF22210
        DUM1 = (V(J)*V(J)-U(2,J))*(U(2,J+1)-U(2,J-1))/TWOX                BEF22220
        DUM2 = V(J)*(U(1,J+1)-U(1,J-1))/TWOX                              BEF22230
        DUM3 = THETA(J)*U(2,J)                                            BEF22240
        DUM4 = FW*DABS(V(J))*V(J)                                         BEF22250
        DUM5 = 2.D+00*FW*DABS(V(J))/U(2,J)                                BEF22260
        EJN  = DUM5*(DUM1-DUM2-DUM3-DUM4)                                 BEF22270
        S1(J) = X*EJN - THETA(J)*(U(1,J+1)-U(1,J-1))/2.D+00               BEF22280
  100 CONTINUE                                                            BEF22290
C                                                                         BEF22300
      RETURN                                                              BEF22310
      END                                                                 BEF22320
C                                                                         BEF22330
C -22------------------  END OF SUBROUTINE MATS  -------------BEFORR-     BEF22340
C #23########################  SUBROUTINE MATD  ################BEFORR#  BEF22350
C                                                                         BEF22360
C     This subroutine computes, for each node, the elements of            BEF22370
C     Matrix D (2x1) --> D(1,j) and D(2,j), j=node number                 BEF22380
C                                                                         BEF22390
      SUBROUTINE MATD (MODE,JDAM,JEND)                                    BEF22400
C                                                                         BEF22410
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                 BEF22420
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                BEF22430
      DIMENSION Q(2,2,N1),UU(2,N1)                                        BEF22440
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                             BEF22450
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX       BEF22460
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)               BEF22470
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)    BEF22480
      IF (MODE.EQ.0) THEN                                                 BEF22490
        CALL CKFPAR (23,1,N1,N1R)                                         BEF22500
        RETURN                                                            BEF22510
      ENDIF                                                               BEF22520
C                                                                         BEF22530
      DO 120 J = 1,JDAM                                                   BEF22540
        CC1 = C(J+1)+C(J)                                                 BEF22550
```

```
        CC2 = C(J+1)-C(J)                                                    BEF22560
        VC1 = V(J+1)+V(J)+CC1                                                BEF22570
        VC2 = V(J+1)-V(J)+CC2                                                BEF22580
        VC3 = V(J+1)+V(J)-CC1                                                BEF22590
        VC4 = V(J+1)-V(J)-CC2                                                BEF22600
        PPP = (-X1*DABS(VC2)*VC3+X2*DABS(VC4)*VC1)/(2.D+00*CC1)              BEF22610
        QQQ = (X1*DABS(VC2)-X2*DABS(VC4))/CC1                                BEF22620
        Q(1,1,J) = QQQ*(A1(1,J+1)+A1(1,J))/2.D+00 + PPP                      BEF22630
        Q(1,2,J) = QQQ*(A1(2,J+1)+A1(2,J))/2.D+00                            BEF22640
        Q(2,1,J) = QQQ                                                       BEF22650
        Q(2,2,J) = PPP                                                       BEF22660
        DO 110 I = 1,2                                                       BEF22670
          UU(I,J) = U(I,J+1)-U(I,J)                                          BEF22680
  110   CONTINUE                                                             BEF22690
  120 CONTINUE                                                               BEF22700
      DO 150 I = 1,2                                                         BEF22710
        DO 140 J = 2,JDAM                                                    BEF22720
          D(I,J) = 0.D+00                                                    BEF22730
          DO 130 L = 1,2                                                     BEF22740
            D(I,J) = D(I,J) + Q(I,L,J)*UU(L,J) - Q(I,L,J-1)*UU(L,J-1)        BEF22750
  130     CONTINUE                                                           BEF22760
          D(I,J) = TX*D(I,J)/2.D+00                                          BEF22770
  140   CONTINUE                                                             BEF22780
  150 CONTINUE                                                               BEF22790
      IF (JEND.GT.JDAM) THEN                                                 BEF22800
        DO 170 I = 1,2                                                       BEF22810
          DO 160 J = JDAM+1,JEND                                             BEF22820
            D(I,J) = 0.D+00                                                  BEF22830
  160     CONTINUE                                                           BEF22840
  170   CONTINUE                                                             BEF22850
      ENDIF                                                                  BEF22860
C                                                                            BEF22870
      RETURN                                                                 BEF22880
      END                                                                    BEF22890
C                                                                            BEF22900
C -23-------------------  END OF SUBROUTINE MATD  -------------BEFORR-       BEF22910
C #24#########################  SUBROUTINE MATU  ################BEFORR#    BEF22920
C                                                                            BEF22930
C     This subroutine computes the elements of Matrix U (2x1)                BEF22940
C     --> U(1,j) and U(2,j) with j=node number at next time level            BEF22950
C                                                                            BEF22960
      SUBROUTINE MATU (MODE,JBEGIN,JEND)                                     BEF22970
C                                                                            BEF22980
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                    BEF22990
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                   BEF23000
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                                BEF23010
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX          BEF23020
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)                  BEF23030
      COMMON /MATRIX/ A1(2,N1),F(2,N1),G1(N1),GJR(2,N1),S1(N1),D(2,N1)       BEF23040
      IF (MODE.EQ.0) THEN                                                    BEF23050
        CALL CKFPAR (24,1,N1,N1R)                                            BEF23060
        RETURN                                                               BEF23070
      ENDIF                                                                  BEF23080
C                                                                            BEF23090
      DO 100 J = JBEGIN,JEND                                                 BEF23100
```

```
        DUM1 = TX*((F(1,J+1)-F(1,J-1))/2.D+00+X*G1(J))                    BEF23110
        DUM2 = TTXX*(GJR(1,J)-GJR(1,J-1))                                 BEF23120
        DUM3 = TX*(F(2,J+1)-F(2,J-1))                                     BEF23130
        DUM4 = TTXX*(GJR(2,J)-GJR(2,J-1))                                 BEF23140
        U(1,J) = U(1,J) - DUM1 + (DUM2-TTX*S1(J))/2.D+00 + D(1,J)         BEF23150
        U(2,J) = U(2,J) - (DUM3-DUM4)/2.D+00 + D(2,J)                     BEF23160
  100 CONTINUE                                                            BEF23170
C                                                                         BEF23180
      RETURN                                                              BEF23190
      END                                                                 BEF23200
C                                                                         BEF23210
C -24-------------------  END OF SUBROUTINE MATU  -------------BEFORR-    BEF23220
C #25#######################  SUBROUTINE NSI  ##################BEFORR#  BEF23230
C                                                                         BEF23240
C     This subroutine calculates numerical stability indicator OMEGA      BEF23250
C                                                                         BEF23260
      SUBROUTINE NSI (OMEGA,NONE)                                         BEF23270
C                                                                         BEF23280
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                 BEF23290
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX       BEF23300
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT       BEF23310
      EPSI  = DMAX1(X1,X2)                                                BEF23320
      DUM1  = 1.D+00 + EPSI*EPSI/4.D+00                                   BEF23330
      DUM2  = DSQRT(DUM1) - EPSI/2.D+00                                   BEF23340
      OMEGA = DUM2*X*DBLE(NONE)/(1.D+00+DSQRT(DSEA))                      BEF23350
      RETURN                                                              BEF23360
      END                                                                 BEF23370
C                                                                         BEF23380
C -25-------------------  END OF SUBROUTINE NSI  -------------BEFORR-    BEF23390
C #26#######################  SUBROUTINE INIT  ##################BEFORR#  BEF23400
C                                                                         BEF23410
C     This subroutine assigns initial values                              BEF23420
C                                                                         BEF23430
      SUBROUTINE INIT (MODE)                                              BEF23440
C                                                                         BEF23450
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                 BEF23460
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                BEF23470
      INTEGER S,SM1,SP1,SSAV                                              BEF23480
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                             BEF23490
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,                 BEF23500
     +                ISAVA,ISAVB,ISAVC                                   BEF23510
      COMMON /CPAR3/  INITS,JE,JE1                                        BEF23520
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX       BEF23530
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)       BEF23540
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)               BEF23550
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                                BEF23560
      COMMON /DIAGV1/ SSAV(3),JMSAV(3),NONESV(N6),MFIVE                   BEF23570
      COMMON /DIAGV2/ USAV(6,N1),VSAV(3,N1),ESAV(3,N1),DELSAV(3),         BEF23580
     +                DELTSV(N6)                                          BEF23590
      IF (MODE.EQ.0) THEN                                                 BEF23600
        CALL CKFPAR (26,1,N1,N1R)                                         BEF23610
        CALL CKFPAR (26,6,N6,N6R)                                         BEF23620
        RETURN                                                            BEF23630
      ENDIF                                                               BEF23640
C                                                                         BEF23650
```

```
C           HYDRODYNAMIC VARIABLES                                          BEF23660
C           ----------------------------------------------------------      BEF23670
C         . S = node number closest to the instantaneous computational      BEF23680
C           waterline                                                       BEF23690
C         . JMAX = the largest node number reached by computational         BEF23700
C           waterline                                                       BEF23710
C         . Hydrodynamic quantities at node j:                              BEF23720
C               U(1,j)  = volume flux                                       BEF23730
C               U(2,j)  = total water depth                                 BEF23740
C                         (not less than DELTA for IJOB=3)                  BEF23750
C               V(j)    = depth-averaged velocity                           BEF23760
C               ELEV(j) = surface elevation above SWL                       BEF23770
C           ----------------------------------------------------------      BEF23780
      S    = INITS                                                          BEF23790
      JMAX = INITS                                                          BEF23800
      DO 110 J = 1,JE                                                       BEF23810
        U(1,J) = 0.D+00                                                     BEF23820
        IF (J.LE.INITS) THEN                                                BEF23830
          U(2,J) = U2INIT(J)                                                BEF23840
          ELEV(J)= 0.D+00                                                   BEF23850
        ELSE                                                                BEF23860
          U(2,J) = 0.D+00                                                   BEF23870
          ELEV(J)= ZB(J)                                                    BEF23880
        ENDIF                                                               BEF23890
        V(J) = 0.D+00                                                       BEF23900
        IF (IJOB.EQ.3.AND.U(2,J).LT.DELTA) U(2,J)=DELTA                     BEF23910
  110 CONTINUE                                                              BEF23920
C                                                                           BEF23930
C           SAVING VARIABLES AT LEVEL=1 AND LEVEL=2                         BEF23940
C           ----------------------------------------------------------      BEF23950
      DO 130 L = 1,2                                                        BEF23960
        SSAV(L)   = S                                                       BEF23970
        JMSAV(L)  = JMAX                                                    BEF23980
        DELSAV(L) = DELTA                                                   BEF23990
        LU1 = 2*L-1                                                         BEF24000
        LU2 = 2*L                                                           BEF24010
        DO 120 J = 1,JE                                                     BEF24020
          USAV(LU1,J) = U(1,J)                                              BEF24030
          USAV(LU2,J) = U(2,J)                                              BEF24040
          VSAV(L,J)   = V(J)                                                BEF24050
          ESAV(L,J)   = ELEV(J)                                             BEF24060
  120   CONTINUE                                                            BEF24070
  130 CONTINUE                                                              BEF24080
C                                                                           BEF24090
      RETURN                                                                BEF24100
      END                                                                   BEF24110
C                                                                           BEF24120
C -26------------------  END OF SUBROUTINE INIT  ------------BEFORR-        BEF24130
C #27#####################  SUBROUTINE INITM  ################BEFORR#      BEF24140
C                                                                           BEF24150
C     This subroutine sets initial hydrodynamic quantities at the be-       BEF24160
C     ginning of a computation unit using the saved values at LEVEL=1       BEF24170
C                                                                           BEF24180
      SUBROUTINE INITM (MODE)                                               BEF24190
C                                                                           BEF24200
```

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                BEF24210
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)               BEF24220
      INTEGER S,SM1,SP1,SSAV                                             BEF24230
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                            BEF24240
      COMMON /CPAR3/  INITS,JE,JE1                                       BEF24250
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX      BEF24260
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)              BEF24270
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                               BEF24280
      COMMON /DIAGV1/ SSAV(3),JMSAV(3),NONESV(N6),MFIVE                  BEF24290
      COMMON /DIAGV2/ USAV(6,N1),VSAV(3,N1),ESAV(3,N1),DELSAV(3),        BEF24300
     +                DELTSV(N6)                                         BEF24310
      IF (MODE.EQ.0) THEN                                                BEF24320
        CALL CKFPAR (27,1,N1,N1R)                                        BEF24330
        CALL CKFPAR (27,6,N6,N6R)                                        BEF24340
        RETURN                                                           BEF24350
      ENDIF                                                              BEF24360
      S     = SSAV(1)                                                    BEF24370
      JMAX  = JMSAV(1)                                                   BEF24380
      DELTA = DELSAV(1)                                                  BEF24390
      DO 100 J = 1,JE                                                    BEF24400
        U(1,J)  = USAV(1,J)                                              BEF24410
        U(2,J)  = USAV(2,J)                                              BEF24420
        V(J)    = VSAV(1,J)                                              BEF24430
        ELEV(J) = ESAV(1,J)                                              BEF24440
  100 CONTINUE                                                           BEF24450
      RETURN                                                             BEF24460
      END                                                                BEF24470
C                                                                        BEF24480
C -27------------------  END OF SUBROUTINE INITM  ------------BEFORR-    BEF24490
C #28######################  SUBROUTINE INITMM  ###############BEFORR#   BEF24500
C                                                                        BEF24510
C     This subroutine sets initial hydrodynamic quantities at the       BEF24520
C     beginning of one set of five computation units using the saved     BEF24530
C     values at LEVEL=2 or LEVEL=3                                       BEF24540
C                                                                        BEF24550
      SUBROUTINE INITMM (MODE)                                           BEF24560
C                                                                        BEF24570
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                BEF24580
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)               BEF24590
      INTEGER S,SM1,SP1,SSAV                                             BEF24600
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                            BEF24610
      COMMON /CPAR3/  INITS,JE,JE1                                       BEF24620
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)              BEF24630
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                               BEF24640
      COMMON /DIAGID/ INONE,ITRY,IREV,LEVEL                              BEF24650
      COMMON /DIAGV1/ SSAV(3),JMSAV(3),NONESV(N6),MFIVE                  BEF24660
      COMMON /DIAGV2/ USAV(6,N1),VSAV(3,N1),ESAV(3,N1),DELSAV(3),        BEF24670
     +                DELTSV(N6)                                         BEF24680
      IF (MODE.EQ.0) THEN                                                BEF24690
        CALL CKFPAR (28,1,N1,N1R)                                        BEF24700
        CALL CKFPAR (28,6,N6,N6R)                                        BEF24710
        RETURN                                                           BEF24720
      ENDIF                                                              BEF24730
      S     = SSAV(LEVEL)                                                BEF24740
      JMAX  = JMSAV(LEVEL)                                               BEF24750
```

```
      DELTA = DELSAV(LEVEL)                                                 BEF24760
      LU1 = 2*LEVEL-1                                                       BEF24770
      LU2 = 2*LEVEL                                                         BEF24780
      DO 110 J = 1,JE                                                       BEF24790
        U(1,J)   = USAV(LU1,J)                                              BEF24800
        U(2,J)   = USAV(LU2,J)                                              BEF24810
        V(J)     = VSAV(LEVEL,J)                                            BEF24820
        ELEV(J)  = ESAV(LEVEL,J)                                            BEF24830
  110 CONTINUE                                                              BEF24840
      IF (LEVEL.EQ.3) THEN                                                  BEF24850
C       ::          Rearrange saving variables                            BEF24860
        DO 120 J = 1,JE                                                     BEF24870
          USAV(3,J) = USAV(5,J)                                             BEF24880
          USAV(4,J) = USAV(6,J)                                             BEF24890
          VSAV(2,J) = VSAV(3,J)                                             BEF24900
          ESAV(2,J) = ESAV(3,J)                                             BEF24910
  120   CONTINUE                                                            BEF24920
      ENDIF                                                                 BEF24930
      RETURN                                                                BEF24940
      END                                                                   BEF24950
C                                                                           BEF24960
C -28------------------  END OF SUBROUTINE INITMM  ------------BEFORR-      BEF24970
C #29#####################  SUBROUTINE SAVEM  ################BEFORR#      BEF24980
C                                                                           BEF24990
C     This subroutine saves hydrodynamic quantities at the end of           BEF25000
C     a computation unit (LEVEL=1)                                          BEF25010
C                                                                           BEF25020
      SUBROUTINE SAVEM (MODE)                                               BEF25030
C                                                                           BEF25040
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                   BEF25050
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                  BEF25060
      INTEGER S,SM1,SP1,SSAV                                                BEF25070
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                               BEF25080
      COMMON /CPAR3/  INITS,JE,JE1                                          BEF25090
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX         BEF25100
      COMMON /HYDRO/  U(2,N1),V(N1),ELEV(N1),C(N1),DUDT(N1)                 BEF25110
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                                  BEF25120
      COMMON /DIAGV1/ SSAV(3),JMSAV(3),NONESV(N6),MFIVE                     BEF25130
      COMMON /DIAGV2/ USAV(6,N1),VSAV(3,N1),ESAV(3,N1),DELSAV(3),           BEF25140
     +                DELTSV(N6)                                           BEF25150
      IF (MODE.EQ.0) THEN                                                   BEF25160
        CALL CKFPAR (29,1,N1,N1R)                                           BEF25170
        CALL CKFPAR (29,6,N6,N6R)                                           BEF25180
        RETURN                                                              BEF25190
      ENDIF                                                                 BEF25200
      SSAV(1)   = S                                                         BEF25210
      JMSAV(1)  = JMAX                                                      BEF25220
      DELSAV(1) = DELTA                                                     BEF25230
      DO 100 J = 1,JE                                                       BEF25240
        USAV(1,J) = U(1,J)                                                  BEF25250
        USAV(2,J) = U(2,J)                                                  BEF25260
        VSAV(1,J) = V(J)                                                    BEF25270
        ESAV(1,J) = ELEV(J)                                                 BEF25280
  100 CONTINUE                                                              BEF25290
      RETURN                                                                BEF25300
```

```
      END                                                                   BEF25310
C                                                                           BEF25320
C -29------------------  END OF SUBROUTINE SAVEM  ------------BEFORR-       BEF25330
C #30######################  SUBROUTINE SAVEMM  ###############BEFORR#     BEF25340
C                                                                           BEF25350
C     This subroutine saves hydrodynamic quantities at the end of one       BEF25360
C     set of five computation units (LEVEL=2 and LEVEL=3)                   BEF25370
C                                                                           BEF25380
      SUBROUTINE SAVEMM (MODE)                                              BEF25390
C                                                                           BEF25400
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                   BEF25410
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                  BEF25420
      INTEGER SSAV                                                          BEF25430
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                               BEF25440
      COMMON /CPAR3/  INITS,JE,JE1                                          BEF25450
      COMMON /DIAGV1/ SSAV(3),JMSAV(3),NONESV(N6),MFIVE                     BEF25460
      COMMON /DIAGV2/ USAV(6,N1),VSAV(3,N1),ESAV(3,N1),DELSAV(3),           BEF25470
     +                DELTSV(N6)                                           BEF25480
      IF (MODE.EQ.0) THEN                                                   BEF25490
        CALL CKFPAR (30,1,N1,N1R)                                           BEF25500
        CALL CKFPAR (30,6,N6,N6R)                                           BEF25510
        RETURN                                                              BEF25520
      ENDIF                                                                 BEF25530
C     ::          Saving variables at LEVEL=3: beginning of the             BEF25540
C                 previous set of five computation units                    BEF25550
      SSAV(3)   = SSAV(2)                                                   BEF25560
      JMSAV(3)  = JMSAV(2)                                                  BEF25570
      DELSAV(3) = DELSAV(2)                                                 BEF25580
      DO 110 J = 1,JE                                                       BEF25590
        USAV(5,J) = USAV(3,J)                                               BEF25600
        USAV(6,J) = USAV(4,J)                                               BEF25610
        VSAV(3,J) = VSAV(2,J)                                               BEF25620
        ESAV(3,J) = ESAV(2,J)                                               BEF25630
  110   CONTINUE                                                            BEF25640
C     ::          Saving variables at LEVEL=2: beginning of the             BEF25650
C                 present set of five computation units                     BEF25660
      SSAV(2)   = SSAV(1)                                                   BEF25670
      JMSAV(2)  = JMSAV(1)                                                  BEF25680
      DELSAV(2) = DELSAV(1)                                                 BEF25690
      DO 120 J = 1,JE                                                       BEF25700
        USAV(3,J) = USAV(1,J)                                               BEF25710
        USAV(4,J) = USAV(2,J)                                               BEF25720
        VSAV(2,J) = VSAV(1,J)                                               BEF25730
        ESAV(2,J) = ESAV(1,J)                                               BEF25740
  120 CONTINUE                                                              BEF25750
      RETURN                                                                BEF25760
      END                                                                   BEF25770
C                                                                           BEF25780
C -30------------------  END OF SUBROUTINE SAVEMM  ------------BEFORR-       BEF25790
C #31########################  SUBROUTINE DOC1  ###############BEFORR#     BEF25800
C                                                                           BEF25810
C     This subroutine documents essential information before                BEF25820
C     time-marching computation                                             BEF25830
C                                                                           BEF25840
      SUBROUTINE DOC1 (MODE,FINP2)                                          BEF25850
```

```
C                                                                       BEF25860
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF25870
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)              BEF25880
      DOUBLE PRECISION KCNO,MCNO,KC2,KS,KSREF,KSSEA,KSI                 BEF25890
      CHARACTER UL1*7,UL2*7,FINP2*10                                    BEF25900
      INTEGER S,SM1,SP1                                                 BEF25910
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                           BEF25920
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,               BEF25930
     +                ISAVA,ISAVB,ISAVC                                 BEF25940
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES            BEF25950
      COMMON /CPAR2/  MULTIF,NONEM,NONE,NEND,NRATE                      BEF25960
      COMMON /CPAR3/  INITS,JE,JE1                                      BEF25970
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX     BEF25980
      COMMON /WAVE1/  HREFP,TP,WL0P                                     BEF25990
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA          BEF26000
      COMMON /WAVE4/  ETAMAX,ETAMIN                                     BEF26010
      COMMON /WAVE5/  KCNO,ECNO,MCNO,KC2                                BEF26020
      COMMON /WAVE6/  NDATA                                             BEF26030
      COMMON /BOT1/   DSEAP,DLANDP,FWP                                  BEF26040
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT     BEF26050
      COMMON /BOT3/   U2INIT(N1),THETA(N1),SSLOPE(N1),XB(N1),ZB(N1)     BEF26060
      COMMON /BOT4/   NBSEG                                             BEF26070
      COMMON /BOT5/   WBSEG(N4),TBSLOP(N4),XBSEG(N4),ZBSEG(N4)          BEF26080
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              BEF26090
      COMMON /RUNP2/  DELRP(N3),DELTAR(N3)                              BEF26100
      COMMON /ARMOR/  C2,C3,CD,CL,CM,SG,TANPHI,AMIN,AMAX,DAP            BEF26110
      IF (MODE.EQ.0) THEN                                               BEF26120
        CALL CKFPAR (31,1,N1,N1R)                                       BEF26130
        CALL CKFPAR (31,3,N3,N3R)                                       BEF26140
        CALL CKFPAR (31,4,N4,N4R)                                       BEF26150
        RETURN                                                          BEF26160
      ENDIF                                                             BEF26170
C                                                                       BEF26180
C           SYSTEM OF UNITS                                             BEF26190
C           ------------------------------------------------------      BEF26200
      IF (ISYST.EQ.1) THEN                                              BEF26210
        UL1 = ' meters'                                                 BEF26220
        UL2 = '    [mm]'                                                BEF26230
      ELSE                                                              BEF26240
        UL1 = ' feet  '                                                 BEF26250
        UL2 = ' [inch]'                                                 BEF26260
      ENDIF                                                             BEF26270
C                                                                       BEF26280
C           WAVE CONDITION                                              BEF26290
C           ------------------------------------------------------      BEF26300
      WRITE (98,9811)                                                   BEF26310
      IF (IWAVE.EQ.1) THEN                                              BEF26320
        IF (URPRE.LT.26.D+00) THEN                                      BEF26330
          WRITE (98,9812)                                               BEF26340
        ELSE                                                            BEF26350
          WRITE (98,9813) KC2,ECNO,KCNO                                 BEF26360
        ENDIF                                                           BEF26370
      ELSEIF (IWAVE.EQ.2) THEN                                          BEF26380
        WRITE (98,9814) NDATA                                           BEF26390
      ELSE                                                              BEF26400
```

```
        WRITE (98,9815) NDATA                                                BEF26410
      ENDIF                                                                  BEF26420
      WRITE (98,9816) ETAMAX,ETAMIN                                          BEF26430
      WRITE (98,9817) TP,HREFP,UL1,DSEAP,UL1,KSREF,KSSEA,KS                  BEF26440
      WRITE (98,9818) DSEA,WL,SIGMA,UR,KSI                                   BEF26450
      IF (IJOB.EQ.3)  WRITE (98,9819) DLANDP,UL1                             BEF26460
 9811 FORMAT (/'WAVE CONDITION')                                             BEF26470
 9812 FORMAT (/'Stokes II Incident Wave at Seaward Boundary')                BEF26480
 9813 FORMAT (/'Cnoidal Incident Wave at Seaward Boundary'/                  BEF26490
     +         '1-m = ',D20.9/                                               BEF26500
     +         'E   = ',D20.9/                                               BEF26510
     +         'K   = ',D20.9)                                               BEF26520
 9814 FORMAT (/'Incident Wave at Seaward Boundary Given as Input'/           BEF26530
     +         'Number of Data Points    NDATA =',I8)                        BEF26540
 9815 FORMAT (/'Total Wave at Seaward Boundary Given as Input'/              BEF26550
     +         'Number of Data Points    NDATA =',I8)                        BEF26560
 9816 FORMAT ('Norm. Maximum Surface Elev.    = ',F14.6/                     BEF26570
     +        'Norm. Minimum Surface Elev.    = ',F14.6/)                    BEF26580
 9817 FORMAT ('Reference Wave Period          = ',F14.6,' sec.'/             BEF26590
     +        'Reference Wave Height          = ',F14.6,A7/                  BEF26600
     +        'Depth at Seaward Boundary      = ',F14.6,A7/                  BEF26610
     +        'Shoal. Coef. at Reference  Ks1 = ',F11.3/                     BEF26620
     +        '             at Seaw. Bdr. Ks2 = ',F11.3/                     BEF26630
     +        '             Ks = Ks2/Ks1      = ',F11.3)                     BEF26640
 9818 FORMAT ('Norm. Depth at Seaw. Bdr.      = ',F11.3/                     BEF26650
     +        'Normalized Wave Length         = ',F11.3/                     BEF26660
     +        '"Sigma"                        = ',F11.3/                     BEF26670
     +        'Ursell Number                  = ',F11.3/                     BEF26680
     +        'Surf Similarity Parameter      = ',F11.3)                     BEF26690
 9819 FORMAT ('Depth at Landward Boundary     = ',F14.6,A7)                  BEF26700
C                                                                            BEF26710
C          STRUCTURE PROPERTIES                                              BEF26720
C          ------------------------------------------------------------      BEF26730
      WRITE (98,9821) FWP,FW,WTOT,NBSEG                                      BEF26740
      IF (IBOT.EQ.1) THEN                                                    BEF26750
        WRITE (98,9822) UL1                                                  BEF26760
        WRITE (98,9824) (K,WBSEG(K),TBSLOP(K),K=1,NBSEG)                     BEF26770
      ELSE                                                                   BEF26780
        WRITE (98,9823) UL1,UL1                                              BEF26790
        WRITE (98,9824) (K,XBSEG(K),ZBSEG(K),K=1,NBSEG+1)                    BEF26800
      ENDIF                                                                  BEF26810
      WRITE (98,9825)                                                        BEF26820
 9821 FORMAT (/'SLOPE PROPERTIES'//                                          BEF26830
     +        'Friction Factor            = ',F15.6/                        BEF26840
     +        'Norm. Friction Factor      = ',F15.6/                        BEF26850
     +        'Norm. Horiz. Length of'/                                     BEF26860
     +        '      Computation Domain   = ',F15.6/                        BEF26870
     +        'Number of Segments         = ',I8)                           BEF26880
 9822 FORMAT (32(1H-)/' SEGMENT    WBSEG(I)   TBSLOP(I)'/                    BEF26890
     +                '       I     ',A7/32(1H-))                          BEF26900
 9823 FORMAT (32(1H-)/' SEGMENT    XBSEG(I)    ZBSEG(I)'/                    BEF26910
     +                '       I     ',A7,'     ',A7/32(1H-))               BEF26920
 9824 FORMAT (I8,2F12.6)                                                     BEF26930
 9825 FORMAT (32(1H-))                                                       BEF26940
C                                                                            BEF26950
```

```
C           PARAMETERS FOR ARMOR STABILITY AND MOVEMENT                    BEF26960
C           ------------------------------------------------------------   BEF26970
      IF (ISTAB.GT.0) WRITE (98,9831) TANPHI,SG,C2,C3,CD,CL,CM             BEF26980
      IF (ISTAB.EQ.1) WRITE (98,9832) AMAX,AMIN                            BEF26990
      IF (ISTAB.EQ.2) WRITE (98,9833) DAP,UL1                              BEF27000
 9831 FORMAT (//'PARAMETERS FOR ARMOR STABILITY AND MOVEMENT'//            BEF27010
     +        'Armor Friction Factor     = ',F9.3/                        BEF27020
     +        'Specific Gravity          = ',F9.3/                        BEF27030
     +        'Area Coefficient      C2  = ',F9.3/                        BEF27040
     +        'Volume Coefficient    C3  = ',F9.3/                        BEF27050
     +        'Drag Coefficient      CD  = ',F9.3/                        BEF27060
     +        'Lift Coefficient      CL  = ',F9.3/                        BEF27070
     +        'Inertia Coefficient   CM  = ',F9.3)                        BEF27080
 9832 FORMAT ('Norm. Upper and Lower Bounds of du/dt'/                     BEF27090
     +        '                     AMAX  = ',F9.3/                       BEF27100
     +        '                     AMIN  = ',F9.3)                       BEF27110
 9833 FORMAT ('Armor Diameter             = ',F12.6,A7)                    BEF27120
C                                                                          BEF27130
C           COMPUTATION PARAMETERS                                         BEF27140
C           ------------------------------------------------------------   BEF27150
      WRITE (98,9841) JE                                                   BEF27160
      IF (IJOB.LT.3)  WRITE (98,9842) INITS                                BEF27170
      WRITE (98,9843) X,X1,X2,TMAX,NONEM,NRATE,MSTAT                       BEF27180
      IF (ISTAB.GT.0) WRITE (98,9844) MSTAB                                BEF27190
      IF (ISAVA.EQ.1) WRITE (98,9845) NTIMES,MSAVA1,MSAVA2                 BEF27200
 9841 FORMAT (//'COMPUTATION PARAMETERS'//                                 BEF27210
     +  'Total Number of Spatial Nodes      JE = ',I8)                    BEF27220
 9842 FORMAT ('Number of Nodes Along Bottom Below SWL'/                    BEF27230
     +  '                                 INITS = ',I8)                   BEF27240
 9843 FORMAT (                                                             BEF27250
     +  'Normalized Delta x          = ',D19.6/                           BEF27260
     +  'Damping Coefficients    x1 = ',F15.3/                            BEF27270
     +  '                        x2 = ',F15.3/                            BEF27280
     +  'Computation Duration  TMAX = ',F18.6,' Wave Periods'/            BEF27290
     +  'Minimum Allowable NONE           NONEM = ',I8/                   BEF27300
     +  '(NONE = number of time steps in one wave period)'/               BEF27310
     +  'Time Series Stored at Rate NRATE'/                               BEF27320
     +  '     per Wave Period with        NRATE = ',I8/                   BEF27330
     +  'Statistical Calculations Are Performed Excluding'/               BEF27340
     +  '     the First MSTAT Computation Units'/                         BEF27350
     +  '     with                        MSTAT = ',I8)                   BEF27360
 9844 FORMAT (                                                             BEF27370
     +  'Armor Stability Computations Are Performed Excluding'/           BEF27380
     +  '     the First MSTAB Computation Units'/                         BEF27390
     +  '     with                        MSTAB = ',I8)                   BEF27400
 9845 FORMAT (                                                             BEF27410
     +  'Spatial Variations are Stored NTIMES at Equal'/                  BEF27420
     +  '     Intervals per Wave Period from'/                            BEF27430
     +  '     Computation Unit MSAVA1 to MSAVA2,'/                        BEF27440
     +  '     Inclusive, with           NTIMES = ',I8/                    BEF27450
     +  '                               MSAVA1 = ',I8/                    BEF27460
     +  '                               MSAVA2 = ',I8)                    BEF27470
C                                                                          BEF27480
C           RUNUP METER                                                    BEF27490
C           ------------------------------------------------------------   BEF27500
```

```
      IF (IJOB.LT.3) THEN                                                BEF27510
        WRITE (98,9851) UL2                                              BEF27520
        DO 110 L = 1,NDELR                                               BEF27530
          WRITE (98,9852) L,DELRP(L)                                     BEF27540
  110   CONTINUE                                                         BEF27550
        WRITE (98,9853)                                                  BEF27560
      ENDIF                                                              BEF27570
 9851 FORMAT (/'RUNUP METER'//5x,17(1H-)/                                BEF27580
     +        '       I   DELTAR(I)'/'           ',A7/5x,17(1H-))       BEF27590
 9852 FORMAT (I8,1X,F9.3)                                                BEF27600
 9853 FORMAT (5x,17(1H-))                                                BEF27610
C                                                                        BEF27620
C           COMPUTATION LOG                                              BEF27630
C           -------------------------------------------------------      BEF27640
      WRITE (98,9861)                                                    BEF27650
 9861 FORMAT (/'COMPUTATION LOG'//68(1H-)/                               BEF27660
     +  1x,'Comp.   NONE    NEND     NEND/       DELTA',                BEF27670
     +      '    Num.     No.     Re-'/                                  BEF27680
     +  1x,'Unit',22x,'NONE',15x,'Stab.      of   verse'/                BEF27690
     + 45x,'Indic.  Trials  Indic.'/68(1H-))                             BEF27700
C                                                                        BEF27710
C           NORMALIZED STRUCTURE GEOMETRY                                BEF27720
C           -------------------------------------------------------      BEF27730
C           File 52 = 'BSPACE'                                           BEF27740
C           (XB(j),ZB(j)) = normalized coordinates of the structure      BEF27750
C                           at node j                                    BEF27760
C           ZB negative below SWL                                        BEF27770
C           -------------------------------------------------------      BEF27780
      WRITE (52,9000) JE                                                 BEF27790
      WRITE (52,8000) (XB(J),ZB(J),J=1,JE)                               BEF27800
 9000 FORMAT (I8)                                                        BEF27810
 8000 FORMAT (5D15.6)                                                    BEF27820
C                                                                        BEF27830
C           WRITE "CASE-SIGNATURES" IN FILE 'BINPUT'                     BEF27840
C           -------------------------------------------------------      BEF27850
      WRITE (51,9000) MWAVE                                              BEF27860
      IF (IWAVE.GT.1) THEN                                               BEF27870
        WRITE (51,5000) FINP2                                            BEF27880
        WRITE (51,9000) NDATA                                            BEF27890
      ENDIF                                                              BEF27900
      WRITE (51,9000) NONEM                                              BEF27910
 5000 FORMAT (A10)                                                       BEF27920
C                                                                        BEF27930
C           CONDITIONAL STOP BEFORE TIME-MARCHING COMPUTATION            BEF27940
C           -------------------------------------------------------      BEF27950
      WRITE (*,6010)                                                     BEF27960
      READ  (*,*) ISTOP                                                  BEF27970
      IF (ISTOP.EQ.1) STOP                                               BEF27980
 6010 FORMAT (' Time-marching computation is about to begin'/            BEF27990
     +        ' 1 = stop here, else = proceed')                           BEF28000
C                                                                        BEF28010
      RETURN                                                             BEF28020
      END                                                                BEF28030
C                                                                        BEF28040
C -31------------------  END OF SUBROUTINE DOC1  ------------BEFORR-     BEF28050
```

```
C #32#########################  SUBROUTINE DOC2  ################BEFORR#  BEF28060
C                                                                          BEF28070
C     This subroutine writes computation log during time-marching          BEF28080
C     computation                                                          BEF28090
C                                                                          BEF28100
      SUBROUTINE DOC2 (MODE,M,MREP)                                        BEF28110
C                                                                          BEF28120
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                                  BEF28130
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)                 BEF28140
      INTEGER SSAV                                                         BEF28150
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                              BEF28160
      COMMON /CPAR2/  MULTIF,NONEM,NONE,NEND,NRATE                         BEF28170
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX        BEF28180
      COMMON /DIAGID/ INONE,ITRY,IREV,LEVEL                                BEF28190
      COMMON /DIAGV1/ SSAV(3),JMSAV(3),NONESV(N6),MFIVE                    BEF28200
      COMMON /DIAGV2/ USAV(6,N1),VSAV(3,N1),ESAV(3,N1),DELSAV(3),          BEF28210
     +                DELTSV(N6)                                           BEF28220
      IF (MODE.EQ.0) THEN                                                  BEF28230
        IF (MREP.LT.0) CALL STOPP (16,16)                                  BEF28240
        CALL CKFPAR (32,1,N1,N1R)                                          BEF28250
        CALL CKFPAR (32,6,N6,N6R)                                          BEF28260
        RETURN                                                             BEF28270
      ENDIF                                                                BEF28280
C                                                                          BEF28290
C           WRITE IN FILE 'BDOC'                                           BEF28300
C           ------------------------------------------------------         BEF28310
      CALL NSI (OMEGA,NONE)                                                BEF28320
      IF (NONE.EQ.NEND) THEN                                               BEF28330
        DUR = 1.D+00                                                       BEF28340
        WRITE (98,9810) M,NONE,NEND,DUR,DELTA,OMEGA,ITRY,IREV              BEF28350
      ELSE                                                                 BEF28360
        DUR = DBLE(NEND)/DBLE(NONE)                                        BEF28370
        WRITE (98,9820) M,NONE,NEND,DUR,DELTA,OMEGA,ITRY,IREV              BEF28380
      ENDIF                                                                BEF28390
      NONESV(M) = NONE                                                     BEF28400
      DELTSV(M) = DELTA                                                    BEF28410
 9810 FORMAT (I5,2I8,F10.1,F12.8,F8.2,2I8)                                 BEF28420
 9820 FORMAT (I5,2I8,F10.6,F12.8,F8.2,2I8)                                 BEF28430
C                                                                          BEF28440
C           WRITE ON SCREEN every MREP waves                               BEF28450
C           ------------------------------------------------------         BEF28460
      IF (MREP.GT.0) THEN                                                  BEF28470
        IDUM = MOD(M,MREP)                                                 BEF28480
        IF (IDUM.EQ.0) WRITE (*,6010) M,NONE,DELTA,ITRY,IREV               BEF28490
      ENDIF                                                                BEF28500
 6010 FORMAT (' M,NONE,DELTA,ITRY,IREV',I5,I8,F12.8,2I4)                   BEF28510
C                                                                          BEF28520
      RETURN                                                               BEF28530
      END                                                                  BEF28540
C                                                                          BEF28550
C -32-------------------  END OF SUBROUTINE DOC2  -------------BEFORR-     BEF28560
C #33#########################  SUBROUTINE DOC3  ################BEFORR#  BEF28570
C                                                                          BEF28580
C     This subroutine writes in file 'BINPUT' and completes                BEF28590
C     documentation in file 'BDOC'                                         BEF28600
```

```
C                                                                       BEF28610
      SUBROUTINE DOC3 (MODE)                                            BEF28620
C                                                                       BEF28630
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF28640
      PARAMETER (N1=800,N2=16000,N3=3,N4=40,N5=40,N6=2000)              BEF28650
      INTEGER S,SM1,SP1,SSAV                                            BEF28660
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                           BEF28670
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,               BEF28680
     +                ISAVA,ISAVB,ISAVC                                 BEF28690
      COMMON /CPAR1/  MWAVE,MSTAT,MSTAB,MSAVA1,MSAVA2,NTIMES            BEF28700
      COMMON /CPAR2/  MULTIF,NONEM,NONE,NEND,NRATE                      BEF28710
      COMMON /RUNP1/  NDELR,S,SM1,SP1,JMAX                              BEF28720
      COMMON /DIAGV1/ SSAV(3),JMSAV(3),NONESV(N6),MFIVE                 BEF28730
      COMMON /DIAGV2/ USAV(6,N1),VSAV(3,N1),ESAV(3,N1),DELSAV(3),       BEF28740
     +                DELTSV(N6)                                        BEF28750
      IF (MODE.EQ.0) THEN                                               BEF28760
        CALL CKFPAR (33,1,N1,N1R)                                       BEF28770
        CALL CKFPAR (33,6,N6,N6R)                                       BEF28780
        RETURN                                                          BEF28790
      ENDIF                                                             BEF28800
C                                                                       BEF28810
      IF (IWAVE.EQ.1) WRITE (51,9000) NONESV(MWAVE)                     BEF28820
      DO 100 M = 1,MWAVE-1                                              BEF28830
        WRITE (51,5110) M,NONESV(M),NONESV(M),DELTSV(M)                 BEF28840
  100 CONTINUE                                                          BEF28850
      WRITE (51,5110) MWAVE,NONESV(MWAVE),NEND,DELTSV(MWAVE)            BEF28860
 5110 FORMAT (3I8,D18.9)                                                BEF28870
 9000 FORMAT (I8)                                                       BEF28880
C                                                                       BEF28890
      IF (IJOB.LT.3) WRITE (98,9811) JMAX                               BEF28900
 9811 FORMAT (68(1H-)//'RUNUP'//                                        BEF28910
     + 'Over the time span 0 < t <or= TMAX,'/                           BEF28920
     + '   the largest node number reached by'/                        BEF28930
     + '   the computational waterline is   =',I8/                     BEF28940
     + 'Notes:'/                                                        BEF28950
     + '1. This information may be useful to figure out the minimum'/   BEF28960
     + '   value of the dimension N1 that should be provided for'/      BEF28970
     + '   the subsequent RBREAK computation.'/                         BEF28980
     + '2. The above number may not be identical to JMAX to be'/        BEF28990
     + '   computed by RBREAK. JMAX is the largest node number'/        BEF29000
     + '   reached by the computational waterline over the'/            BEF29010
     + '   time span MSTAT < t <or= TMAX.')                             BEF29020
C                                                                       BEF29030
      RETURN                                                            BEF29040
      END                                                               BEF29050
C                                                                       BEF29060
C -33-------------------  END OF SUBROUTINE DOC3  ------------BEFORR-   BEF29070
C #34#####################  SUBROUTINE CKFPAR  ##############BEFORR#   BEF29080
C                                                                       BEF29090
C     This subroutine checks if FORTRAN PARAMETER NCHEK=N1,N2,N3,N4,    BEF29100
C     N5,N6 specified in given Subroutine (ICALL) matches its           BEF29110
C     counterpart NREF=N1R,N2R,N3R,N4R,N5R,N6R from the main program    BEF29120
C                                                                       BEF29130
      SUBROUTINE CKFPAR (ICALL,NW,NCHEK,NREF)                           BEF29140
C                                                                       BEF29150
```

```
      CHARACTER*2 WHICH(6)                                              BEF29160
      CHARACTER*6 SUBR(40)                                              BEF29170
      DATA WHICH /'N1','N2','N3','N4','N5','N6'/                        BEF29180
      DATA SUBR  /'OPENIO','INPUT1','INPUT2','BOTTOM','WPARAM',         BEF29190
     2            'REGWAV','FINDM ','CEL   ','SNCNDN','CPARAM',         BEF29200
     3            'NUMSTA','EXTRAP','RETAIN','CRITV ','MARCH ',         BEF29210
     4            'LANDBC','RUNUP ','OVERT ','SEABC ','MATAFG',         BEF29220
     5            'MATGJR','MATS  ','MATD  ','MATU  ','NSI   ',         BEF29230
     6            'INIT  ','INITM ','INITMM','SAVEM ','SAVEMM',         BEF29240
     7            'DOC1  ','DOC2  ','DOC3  ','CKFPAR','CKOPTI',         BEF29250
     8            'CKVAL ','CKVAL1','CKVAL2','CKSUBR','STOPP '/         BEF29260
      IF (NCHEK.NE.NREF) THEN                                           BEF29270
        WRITE  (*,9910)                                                 BEF29280
     +         WHICH(NW),NCHEK,ICALL,SUBR(ICALL),WHICH(NW),NREF         BEF29290
        WRITE (99,9910)                                                 BEF29300
     +         WHICH(NW),NCHEK,ICALL,SUBR(ICALL),WHICH(NW),NREF         BEF29310
        STOP                                                            BEF29320
      ENDIF                                                             BEF29330
 9910 FORMAT (/                                                         BEF29340
     +' PARAMETER Error: ',A2,' =',I8,' in Subroutine',I3,' ',A6/       BEF29350
     +' Correct Value:   ',A2,' =',I8)                                  BEF29360
      RETURN                                                            BEF29370
      END                                                               BEF29380
C                                                                       BEF29390
C -34------------------  END OF SUBROUTINE CKFPAR  ------------BEFORR-  BEF29400
C #35######################  SUBROUTINE CKOPTI  ###############BEFORR#  BEF29410
C                                                                       BEF29420
C     This subroutine checks user-chosen options                        BEF29430
C                                                                       BEF29440
      SUBROUTINE CKOPTI (IC,INDIC,ITEM,ILOW,IUP)                        BEF29450
C                                                                       BEF29460
      CHARACTER*6 OPTI(15)                                              BEF29470
      DATA OPTI  /'IJOB  ','ISTAB ','ISYST ','IBOT  ','INONCT',         BEF29480
     1            'IENERG','IWAVE ','ISAVA ','ISAVB ','ISAVC ',         BEF29490
     2            'IREQ  ','IELEV ','IV    ','IDUDT ','ISNR  '/         BEF29500
      IF (ITEM.LT.ILOW.OR.ITEM.GT.IUP) THEN                             BEF29510
        WRITE  (*,9910) OPTI(IC),ITEM,OPTI(IC),ILOW,IUP                 BEF29520
        WRITE (99,9910) OPTI(IC),ITEM,OPTI(IC),ILOW,IUP                 BEF29530
        INDIC = INDIC+1                                                 BEF29540
      ENDIF                                                             BEF29550
 9910 FORMAT (/' Input Error: ',A6,'=',I1/                              BEF29560
     +         ' Specify ',A6,' in the range of [',I1,',',I1,']')       BEF29570
      RETURN                                                            BEF29580
      END                                                               BEF29590
C                                                                       BEF29600
C -35------------------  END OF SUBROUTINE CKOPTI  ------------BEFORR-  BEF29610
C #36######################  SUBROUTINE CKVAL  ###############BEFORR#  BEF29620
C                                                                       BEF29630
C     This subroutine checks a number of user-specified values          BEF29640
C     Note: Checkings performed in this subroutine are separated from   BEF29650
C           Subr. 02 INPUT1 because some FORTRAN compilers may have     BEF29660
C           problems with subroutines having too many IF statements     BEF29670
C                                                                       BEF29680
      SUBROUTINE CKVAL (INDIC)                                          BEF29690
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF29700
```

```
      DOUBLE PRECISION KS,KSREF,KSSEA,KSI                                BEF29710
      COMMON /CPAR2/  MULTIF,NONEM,NONE,NEND,NRATE                       BEF29720
      COMMON /CPAR4/  X1,X2,DELTA,TMAX,TIME,T,X,TX,XT,TTX,TTXX,TWOX      BEF29730
      COMMON /WAVE1/  HREFP,TP,WL0P                                      BEF29740
      COMMON /WAVE2/  KS,KSREF,KSSEA,WL0,WL,UR,URPRE,KSI,SIGMA           BEF29750
      COMMON /BOT1/   DSEAP,DLANDP,FWP                                   BEF29760
      COMMON /BOT2/   DSEA,DSEAKS,DSEA2,DLAND,DLAND2,FW,TSLOPS,WTOT      BEF29770
      IF (MULTIF.LT.1)       CALL CKVAL2 (1,INDIC,'MULTIF')              BEF29780
      IF (NRATE .LT.1)       CALL CKVAL2 (1,INDIC,'NRATE ')              BEF29790
      IF (DELTA .LE.0.D+00) CALL CKVAL2 (2,INDIC,'DELTA ')               BEF29800
      IF (TMAX  .LE.0.D+00) CALL CKVAL2 (2,INDIC,'TMAX  ')               BEF29810
      IF (X1    .LT.0.D+00) CALL CKVAL2 (3,INDIC,'X1    ')               BEF29820
      IF (X2    .LT.0.D+00) CALL CKVAL2 (3,INDIC,'X2    ')               BEF29830
      IF (TP    .LE.0.D+00) CALL CKVAL2 (2,INDIC,'TP    ')               BEF29840
      IF (KSREF .LE.0.D+00) CALL CKVAL2 (2,INDIC,'KSREF ')               BEF29850
      IF (KSSEA .LE.0.D+00) CALL CKVAL2 (2,INDIC,'KSSEA ')               BEF29860
      IF (FWP   .LT.0.D+00) CALL CKVAL2 (3,INDIC,'FWP   ')               BEF29870
      IF (DSEAP .LE.0.D+00) CALL CKVAL2 (2,INDIC,'DSEAP ')               BEF29880
      IF (TSLOPS.LE.0.D+00) CALL CKVAL2 (2,INDIC,'TSLOPS')               BEF29890
      RETURN                                                             BEF29900
      END                                                                BEF29910
C                                                                        BEF29920
C -36-------------------  END OF SUBROUTINE CKVAL  ------------BEFORR-   BEF29930
C #37#######################  SUBROUTINE CKVAL1  ###############BEFORR#  BEF29940
C                                                                        BEF29950
C     This subroutine checks user-specified values                       BEF29960
C                                                                        BEF29970
      SUBROUTINE CKVAL1 (IC,ITEM,ILOW,IUP)                               BEF29980
C                                                                        BEF29990
      CHARACTER*2 WHICH(7)                                               BEF30000
      CHARACTER*6 WHAT(12)                                               BEF30010
      DATA WHICH /'N5','N5','N5','N3','N1','N4','N2'/                    BEF30020
      DATA WHAT  /'NREQ  ','NNODB ','NNODC ','NDELR ','INITS ',          BEF30030
     +            'NBSEG ','NDATA ','MSTAT ','MSAVA1','MSAVA2',          BEF30040
     +            'NODB  ','NODC  '/                                     BEF30050
      IF (IC.LE.7) THEN                                                  BEF30060
        IF (ITEM.LT.ILOW.OR.ITEM.GT.IUP) THEN                            BEF30070
          WRITE  (*,9910) WHAT(IC),ITEM,WHAT(IC),IUP,WHICH(IC)           BEF30080
          WRITE (99,9910) WHAT(IC),ITEM,WHAT(IC),IUP,WHICH(IC)           BEF30090
          STOP                                                           BEF30100
        ENDIF                                                            BEF30110
      ELSE                                                               BEF30120
        IF (ITEM.LT.ILOW.OR.ITEM.GT.IUP) THEN                            BEF30130
          WRITE  (*,9920) WHAT(IC),ITEM,WHAT(IC),ILOW,IUP                BEF30140
          WRITE (99,9920) WHAT(IC),ITEM,WHAT(IC),ILOW,IUP                BEF30150
          STOP                                                           BEF30160
        ENDIF                                                            BEF30170
      ENDIF                                                              BEF30180
 9910 FORMAT (/' Input Error: ',A6,'=',I8/                               BEF30190
     +        ' Specify ',A6,' in the range of [1,',I8,']'/              BEF30200
     +        ' Change PARAMETER ',A2,' if necessary')                   BEF30210
 9920 FORMAT (/' Input Error: ',A6,'=',I8/                               BEF30220
     +        ' Specify ',A6,' in the range of [',I8,',',I8,']')         BEF30230
      RETURN                                                             BEF30240
      END                                                                BEF30250
```

```
C                                                                       BEF30260
C -37------------------  END OF SUBROUTINE CKVAL1  ------------BEFORR-  BEF30270
C #38######################  SUBROUTINE CKVAL2  ###############BEFORR#  BEF30280
C                                                                       BEF30290
C     This subroutine checks user-specified values                      BEF30300
C                                                                       BEF30310
      SUBROUTINE CKVAL2 (IMSG,INDIC,WHAT)                               BEF30320
C                                                                       BEF30330
      CHARACTER WHAT*6,MSG(3)*24                                        BEF30340
      DATA MSG /' must be at least unity.',                             BEF30350
     +          ' must be positive.      ',                             BEF30360
     +          ' must be non-negative.  '/                            BEF30370
      INDIC = INDIC+1                                                   BEF30380
      WRITE  (*,9910) WHAT,MSG(IMSG)                                    BEF30390
      WRITE (99,9910) WHAT,MSG(IMSG)                                    BEF30400
 9910 FORMAT (/' Input Error: ',A6,A24)                                 BEF30410
      RETURN                                                            BEF30420
      END                                                               BEF30430
C                                                                       BEF30440
C -38------------------  END OF SUBROUTINE CKVAL2  ------------BEFORR-  BEF30450
C #39######################  SUBROUTINE CKSUBR  ###############BEFORR#  BEF30460
C                                                                       BEF30470
C     This subroutine checks the values of the FORTRAN PARAMETERs       BEF30480
C     specified in subroutines                                          BEF30490
C                                                                       BEF30500
      SUBROUTINE CKSUBR (N1,N2,N3,N4,N5,N6)                             BEF30510
C                                                                       BEF30520
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)                               BEF30530
      CHARACTER*10 CDUM                                                 BEF30540
      COMMON /DIMENS/ N1R,N2R,N3R,N4R,N5R,N6R                           BEF30550
      COMMON /ID/     IJOB,ISTAB,ISYST,IBOT,INONCT,IWAVE,               BEF30560
     +                ISAVA,ISAVB,ISAVC                                 BEF30570
      DATA IDUM /0/                                                     BEF30580
      DATA DDUM /0.D+00/                                                BEF30590
      DATA CDUM /'Tcharacter'/                                          BEF30600
C                                                                       BEF30610
C           DUMMIES FOR FORTRAN PARAMETERS USED IN THIS PROGRAM         BEF30620
C           ---------------------------------------------------------   BEF30630
C           Variables specified in PARAMETER statement cannot be        BEF30640
C           passed through COMMON statement. The following dummy        BEF30650
C           integers are used in COMMON /DIMENS/.                       BEF30660
C           ---------------------------------------------------------   BEF30670
      N1R = N1                                                          BEF30680
      N2R = N2                                                          BEF30690
      N3R = N3                                                          BEF30700
      N4R = N4                                                          BEF30710
      N5R = N5                                                          BEF30720
      N6R = N6                                                          BEF30730
C                                                                       BEF30740
C           CHECK ALL SUBROUTINES POSSESING FORTRAN "PARAMETER" STATEMENBEF30750
C           Checking is indicated by MODE=0                             BEF30760
C           ---------------------------------------------------------   BEF30770
      MODE=0                                                            BEF30780
      CALL INPUT1 (MODE,CDUM)                                           BEF30790
      CALL BOTTOM (MODE)                                                BEF30800
```

```
      CALL WPARAM (MODE)                                                BEF30810
      CALL NUMSTA (MODE,IDUM,IDUM)                                      BEF30820
      CALL RETAIN (MODE)                                                BEF30830
      CALL CRITV  (MODE,IDUM,IDUM)                                      BEF30840
      CALL MARCH  (MODE,IDUM,IDUM,DDUM)                                 BEF30850
      CALL LANDBC (MODE,IDUM)                                           BEF30860
      CALL SEABC  (MODE,IDUM,IDUM)                                      BEF30870
      CALL MATAFG (MODE,IDUM,IDUM)                                      BEF30880
      CALL MATGJR (MODE,IDUM,IDUM)                                      BEF30890
      CALL MATS   (MODE,IDUM,IDUM)                                      BEF30900
      CALL MATD   (MODE,IDUM,IDUM)                                      BEF30910
      CALL MATU   (MODE,IDUM,IDUM)                                      BEF30920
      CALL INIT   (MODE)                                                BEF30930
      CALL INITM  (MODE)                                                BEF30940
      CALL INITMM (MODE)                                                BEF30950
      CALL SAVEM  (MODE)                                                BEF30960
      CALL SAVEMM (MODE)                                                BEF30970
      CALL DOC1   (MODE,CDUM)                                           BEF30980
      CALL DOC2   (MODE,IDUM,IDUM)                                      BEF30990
      CALL DOC3   (MODE)                                                BEF31000
      IF (IJOB.LT.3) THEN                                               BEF31010
        CALL EXTRAP (MODE)                                              BEF31020
        CALL RUNUP  (MODE)                                              BEF31030
        IF (IJOB.EQ.2) CALL OVERT (MODE)                                BEF31040
      ENDIF                                                             BEF31050
      IF (IWAVE.EQ.1) THEN                                              BEF31060
        CALL REGWAV (MODE)                                              BEF31070
      ELSE                                                              BEF31080
        CALL INPUT2 (MODE)                                              BEF31090
      ENDIF                                                             BEF31100
C                                                                       BEF31110
      RETURN                                                            BEF31120
      END                                                               BEF31130
C                                                                       BEF31140
C -39------------------  END OF SUBROUTINE CKSUBR  ------------BEFORR-  BEF31150
C #40######################  SUBROUTINE STOPP  ################BEFORR#  BEF31160
C                                                                       BEF31170
C     This subroutine executes a programmed stop                        BEF31180
C                                                                       BEF31190
      SUBROUTINE STOPP (IBEGIN,IEND)                                    BEF31200
C                                                                       BEF31210
      CHARACTER*55 MSG(16)                                              BEF31220
      DATA MSG /                                                        BEF31230
     1 ' Computation aborted after IREV reached 5.',                    BEF31240
     2 ' Suggested to input larger DELTA.',                             BEF31250
     3 ' There is at least one error in primary input data.',           BEF31260
     4 ' See file BMSG to find out the error(s).',                      BEF31270
     5 ' Special storing requested,',                                   BEF31280
     6 ' but pertinent identifiers not specified correctly.',           BEF31290
     7 ' Check identifiers IREQ,IELEV,IV,IDUDT,ISNR.',                  BEF31300
     8 ' TMAX must be a whole number for IWAVE=1.',                     BEF31310
     9 ' TREQ can not exceed TMAX.',                                    BEF31320
     + ' HREFP must be positive.',                                      BEF31330
     1 ' SWL is always above the structure.',                           BEF31340
     2 ' RUNUP/OVERTOPPING computation can not be performed.',          BEF31350
```

```
     3 ' Part of the structure is above SWL.',                          BEF31360
     4 ' TRANSMISSION computation can not be performed.',               BEF31370
     5 ' Failure in Subr. 09 SNCNDN.',                                  BEF31380
     6 ' MREP must be positive or zero.'/                               BEF31390
      DO 100 I = IBEGIN,IEND                                            BEF31400
        WRITE  (*,9910) MSG(I)                                          BEF31410
        WRITE (99,9910) MSG(I)                                          BEF31420
  100 CONTINUE                                                          BEF31430
      WRITE  (*,9920)                                                   BEF31440
      WRITE (99,9920)                                                   BEF31450
 9910 FORMAT (/A55)                                                     BEF31460
 9920 FORMAT (' Programmed Stop.')                                      BEF31470
C                                                                       BEF31480
      STOP                                                              BEF31490
      END                                                               BEF31500
C                                                                       BEF31510
C -40------------------  END OF SUBROUTINE STOPP  ------------BEFORR-   BEF31520
```

APPENDIX C

EXAMPLES: REGULAR WAVES

The IBREAK report (Kobayashi and Wurjanto 1989d) presented four example computations to illustrate the range of capabilities of the computer program IBREAK. This Appendix shows how to prepare the input data files for RBREAK to compute the same four examples as well as the consistency between IBREAK and RBREAK. The BEFORR and RBREAK programs are simply referred to as RBREAK herein. In the following, information is furnished to clarify the differences of the input data for IBREAK and RBREAK. The input variables that are identical in both programs are not discussed. If a user is not familiar with IBREAK, it is recommended to read Appendices B to E of the IBREAK report for the background information.

In IBREAK, NSTAB indicates the time when the armor analysis begins. RBREAK figures it out automatically.

For irregular wave computations, RBREAK requires MSTAT as input. MSTAT determines when the statistical calculations begin. For regular wave computations, like these examples discussed in this Appendix, the statistical calculations are always performed during the last wave period and the value of MSTAT specified by a user is not used in the statistical calculations.

For ISAVA=1, IBREAK uses the input values of NSAVA and NTIMES to specify the storage arrangement. In RBREAK, MSAVA1, MSAVA2, and NTIMES are specified as input as explained in Section 3.8.1. It is noted that NTIMES in IBREAK is *not* identical to that in RBREAK.

The variables NNOD1, NNOD2, NOD1(i), and NOD2(j) with i=1,2,...,NNOD1 and j=1,2,...,NNOD2 in IBREAK are replaced by the variables NNODB, NNODC, NODB(i), and NODC(j), respectively, with i=1,2,...,NNODB and j=1,2,...,NNODC in RBREAK. Furthermore, RBREAK does not require specification of file names for storing computed time series.

The time level NREQ(i) with i=1,2,...,NNREQ in IBREAK and the normalized time TREQ(i) with i=1,2,...,NREQ in RBREAK perform the same function in their respective programs, that is, to indicate moments when the quantities of special request are stored as explained in Section 3.8.4.

The parameter S as input in IBREAK is identical to the parameter INITS in RBREAK.

The value of NONE in RBREAK is not specified as input by a user.

The computation duration in RBREAK is indicated by the normalized time TMAX. TMAX in effect replaces the largest time level NTOP in IBREAK.

The rate of storing computed time series in IBREAK is determined from the integer NJUM1. In RBREAK, this rate is explicitly specified by NRATE.

It is *not* always necessary to supply the *absolute* values of KSSEA and KSREF. It is the *ratio* KS=KSSEA/KSREF that determines the normalized height at the seaward boundary as explained in Section 2.3.1. Therefore, if KSSEA=KSREF, a user may simply specify KSSEA=KSREF=1.

Spatial variations at the time of the minimum armor stability for ISTAB=1 are requested by specifying IREQ=1 in IBREAK. This requires two runs because a user does not know when the minimum armor stability occurs in the first run. RBREAK automatically stores the certain spatial variations at the time of the minimum armor stability when ISTAB=1. The matrix ATMIN explained COMMON/ARMOR5/ in Section 3.5 is added in RBREAK for this purpose.

RBREAK requires MULTIF where MULTIF is explained in Section 4.2.1 under the item NONE.

Table C-1 summarizes the input data files and the files ODOC produced by the two programs for the four examples. The input and output files for the four examples are presented in Tables C-2 to C-21 so that the similarity and difference between IBREAK and RBREAK can be seen.

The comparisons between the output files demonstrate that the computed results from the two programs agree very well, despite the use of different values of the computation parameter NONE. The value of NONE is specified as input in IBREAK, but is automatically assigned by RBREAK.

Table C-1: Summary of input and output files for four examples.

CASE	TABLE	CONTENT OF TABLE	PROGRAM	FILE NAME ON ACCOMPANYING DISK*
Ahrens 1975 Test 12 Zero Damage	C-2	Primary Input Data	IBREAK	Not supplied
	C-3	File ODOC	IBREAK	Not supplied
	C-4	Primary Input Data	RBREAK	appenc1.inp
	C-5	File BINPUT	RBREAK	appenc1.bin
	C-6	File ODOC	RBREAK	appenc1.doc
Ahrens 1975 Test 32 Zero Damage	C-7	Primary Input Data	IBREAK	Not supplied
	C-8	File ODOC	IBREAK	Not supplied
	C-9	Primary Input Data	RBREAK	appenc2.inp
	C-10	File BINPUT	RBREAK	appenc2.bin
	C-11	File ODOC	RBREAK	appenc2.doc
Saville 1955 Run 10	C-12	Primary Input Data	IBREAK	Not supplied
	C-13	File ODOC	IBREAK	Not supplied
	C-14	Primary Input Data	RBREAK	appenc3.inp
	C-15	File BINPUT	RBREAK	appenc3.bin
	C-16	File ODOC	RBREAK	appenc3.doc
Seelig 1980 Run 2	C-17	Primary Input Data	IBREAK	Not supplied
	C-18	File ODOC	IBREAK	Not supplied
	C-19	Primary Input Data	RBREAK	appenc4.inp
	C-20	File BINPUT	RBREAK	appenc4.bin
	C-21	File ODOC	RBREAK	appenc4.doc

* See also Appendix II.

Table C-2: Primary input data file for IBREAK computation of "Ahrens 1975 Test 12 Zero Damage".

```
 1          4                          --> NLINES
 2     ------------------------------------------------------------------------
 3     Ahrens 1975 Test 12 Zero Damage
 4     Computed using IBREAK
 5     ------------------------------------------------------------------------
 6     11   20001                      --> IJOB,ISTAB,NSTAB
 7     2                               --> ISYST
 8     1                               --> IBOT
 9     0                               --> INONCT
10     1                               --> IENERG
11     1  no wave data file            --> IWAVE,FINP2
12     100   20000   5   0   0        --> ISAVA-B-C,NSAVA,NTIMES,NNOD1,NNOD2
13     11111     1                     --> IREQ,IELEV,IV,IDUDT,ISNR,NNREQ
14     21492                           --> NREQ(1)
15        24000                        --> NTOP
16         4000                        --> NONE
17           20                        --> NJUM1
18          100                        --> S
19           .300000                   --> FWP
20          1.000000     1.000000      --> X1,X2
21           .001000                   --> DELTA(normalized)
22            3                        --> NDELR
23           .157480                   --> DELRP(1) (inches)
24           .787402                   --> DELRP(2)
25          1.574803                   --> DELRP(3)
26          3.060000     8.500000      --> HREFP(feet),TP(seconds)
27          1.061400     1.061400      --> KSREF,KSSEA
28         15.000000                   --> DSEAP(feet)
29           .400000                   --> TSLOPS
30            1                        --> NBSEG
31         60.000000      .400000      --> WBSEG(1)(feet),TBSLOP(1)
32            1                                --> NJUM2
33           .900000      .660000     2.710000  --> C2,C3,SG
34           .500000      .180000     1.500000  --> CD,CL,CM
35          1.191754                             --> TANPHI
36          1.000000     -.800000                --> AMAX,AMIN
```

Table C-3: File ODOC produced by IBREAK computation of "Ahrens 1975 Test 12 Zero Damage".

```
 1  ------------------------------------------------------------------------
 2  Ahrens 1975 Test 12 Zero Damage
 3  Computed using IBREAK
 4  ------------------------------------------------------------------------
 5
 6  WAVE CONDITION
 7
 8  Cnoidal Incident Wave at Seaward Boundary
 9  1-m =      0.112815261D+00
10  E   =      0.111509461D+01
11  K   =      0.252149782D+01
12  Norm. Maximum Surface Elev.    =     0.628691
13  Norm. Minimum Surface Elev.    =    -0.371309
14
15  Reference Wave Period          =     8.500000 sec.
16  Reference Wave Height          =     3.060000 feet
17  Depth at Seaward Boundary      =    15.000000 feet
18  Shoal. Coef. at Reference  Ks1 =     1.061
19               at Seaw. Bdr. Ks2 =     1.061
20               Ks = Ks2/Ks1      =     1.000
21  Norm. Depth at Seaw. Bdr.      =     4.902
22  Normalized Wave Length         =    12.144
23  "Sigma"                        =    27.573
24  Ursell Number                  =    30.084
25  Surf Similarity Parameter      =     4.400
26
27  SLOPE PROPERTIES
28
29  Friction Factor            =     0.300000
30  Norm. Friction Factor      =     4.135970
31  Norm. Horiz. Length of
32        Computation Domain   =     0.711121
33  Number of Segments         =        1
34  ------------------------------
35   SEGMENT     WBSEG(I)   TBSLOP(I)
36         I       feet
37  ------------------------------
38         1   60.000000    0.400000
39
40  COMPUTATION PARAMETERS
41
42  Normalized Delta x   =   0.444451D-02
43  Normalized Delta t   =   0.250000D-03
44  Normalized DELTA     =   0.100000E-02
45  Damping Coeff.    x1 =     1.000
46                    x2 =     1.000
47  Num. Stab. Indicator =     3.419
48  Total Number of Time Steps       NTOP =    24000
49  Number of Time Steps in 1 Wave Period
50                                   NONE =     4000
51  Total Number of Spatial Nodes      JE =      161
52  Number of Nodes Along Bottom Below SWL
53                                      S =      100
```

Table C-3 continued

```
 54     Storing Temporal Variations at Every
 55                                        NJUM1 =       20 Time Steps
 56     Armor Stability Number Computed
 57                          at Every      NJUM2 =        1 Time Steps
 58
 59     PARAMETERS FOR ARMOR STABILITY AND MOVEMENT
 60
 61     Armor Friction Factor          =     1.192
 62     Specific Gravity               =     2.710
 63     Area Coefficient          C2   =     0.900
 64     Volume Coefficient        C3   =     0.660
 65     Drag Coefficient          CD   =     0.500
 66     Lift Coefficient          CL   =     0.180
 67     Inertia Coefficient       CM   =     1.500
 68     Norm. Upper and Lower Bounds of du/dt
 69                           AMAX   =     1.000
 70                           AMIN   =    -0.800
 71
 72     REFLECTION COEFFICIENTS
 73
 74     r1 =     0.553
 75     r2 =     0.546
 76     r3 =     0.544
 77
 78     RUNUP, RUNDOWN, SETUP
 79
 80     Largest Node Number Reached by Computational Waterline
 81                                                   JMAX =      139
 82     ------------------------------------------------------------
 83           I   DELTAR(I)     RUNUP(I)    RUNDOWN(I)    SETUP(I)
 84               [inch]          R            Rd           Zr
 85     ------------------------------------------------------------
 86           1     0.157       1.864        0.656        1.318
 87           2     0.787       1.853       -0.283        0.909
 88           3     1.575       1.842       -0.714        0.705
 89
 90     STABILITY NUMBER
 91
 92     Critical Stability Number Nsc =      2.401
 93     At Node Number                J =       85
 94     At Time Level                 N =    21492
 95
 96     WAVE SET-DOWN OR SETUP
 97
 98     Average value of ETAI =     0.000000
 99                      ETAR =     0.015310
100
```

Table C-3 continued

```
101     TIME-AVERAGED ENERGY BALANCE
102
103     Normalized Energy Flux:
104     . at Seaw. Boundary   A =  0.202324D+00
105     . at Landw. Boundary  B =  0.000000D+00
106     Normalized Rate of Energy Dissipation
107     in the Computation Domain, Due to:
108     . bottom friction     C =  0.197916D+00
109     . wave breaking       D =  0.440842D-02
110     Calculation 1:
111                     E = A-B =  0.202324D+00
112                     F = C+D =  0.202324D+00
113       Must G=0, but G = F-E =  0.345043D-06
114       % error  100G/E =          0.00
115     Approximate Energy Flux, Based on
116     Linear Long Wave, Due to:
117     . incident wave at seaw. boundary     P =  0.276755D+00
118     . reflected wave at seaw. boundary    Q =  0.818700D-01
119     Calculation 2:
120     Net Energy Flux at Seaw. Bndry. S = P-Q =  0.194885D+00
121     % Error at Seaward Boundary  100(S-A)/A =         -3.68
```

Table C-4: Primary input data file for RBREAK computation of "Ahrens 1975 Test 12 Zero Damage".

```
 1          3                          --> NLINES
 2    ------------------------------------------------------------------------
 3    Ahrens 1975 Test 12 Zero Damage
 4    ------------------------------------------------------------------------
 5    11                              --> IJOB,ISTAB
 6    2                               --> ISYST
 7    1                               --> IBOT
 8    0                               --> INONCT
 9    1                               --> IENERG
10    1       0  no 2nd data file     --> IWAVE,MSTAT,FINP2
11    1       6       6       4       --> ISAVA,MSAVA1,MSAVA2,NTIMES
12    0       0                       --> ISAVB,NNODB
13    0       0                       --> ISAVC,NNODC
14    00000       0                   --> IREQ,IELEV,IV,IDUDT,ISNR,NREQ
15         100                        --> INITS
16           1                        --> MULTIF
17            .001000                 --> DELTA (normalized)
18           6.000000                 --> TMAX  (normalized)
19         200                        --> NRATE
20           3                        --> NDELR
21          .157480                   --> DELRP(1)(inches)
22          .787402                   -->       (2)
23         1.574803                   -->       (3)
24         1.000000     1.000000      --> X1,X2
25         3.060000     8.500000      --> HREFP(feet),TP(seconds)
26         1.000000     1.000000      --> KSREF,KSSEA
27          .300000                   --> FWP
28        15.000000                   --> DSEAP(feet)
29          .400000                   --> TSLOPS
30           1                        --> NBSEG
31        60.000000      .400000      --> WBSEG(1)(feet),TBSLOP(1)
32          .900000      .660000        2.710000  --> C2,C3,SG
33          .500000      .180000        1.500000  --> CD,CL,CM
34         1.191754                              --> TANPHI
35         1.000000     -.800000                 --> AMAX,AMIN
```

Table C-5: File BINPUT produced by BEFORR computation of "Ahrens 1975 Test 12 Zero Damage".

```
 1          3
 2     ------------------------------------------------------------------------
 3     Ahrens 1975 Test 12 Zero Damage
 4     ------------------------------------------------------------------------
 5          6
 6       1200
 7       2600
 8          1    1200    1200   0.100000000D-02
 9          2    1200    1200   0.100000000D-02
10          3    2600    2600   0.100000000D-02
11          4    2600    2600   0.100000000D-02
12          5    2600    2600   0.100000000D-02
13          6    2600    2600   0.100000000D-02
```

Table C-6: File ODOC produced by RBREAK computation of "Ahrens 1975 Test 12 Zero Damage".

```
 1  ------------------------------------------------------------------------
 2  Ahrens 1975 Test 12 Zero Damage
 3  ------------------------------------------------------------------------
 4
 5  WAVE CONDITION
 6
 7  Cnoidal Incident Wave at Seaward Boundary
 8  1-m =      0.112815261D+00
 9  E   =      0.111509461D+01
10  K   =      0.252149782D+01
11  Norm. Maximum Surface Elev.     =       0.628691
12  Norm. Minimum Surface Elev.     =      -0.371309
13
14  Reference Wave Period           =       8.500000 sec.
15  Reference Wave Height           =       3.060000 feet
16  Depth at Seaward Boundary       =      15.000000 feet
17  Shoal. Coef. at Reference  Ks1  =       1.000
18               at Seaw. Bdr. Ks2  =       1.000
19               Ks = Ks2/Ks1       =       1.000
20  Norm. Depth at Seaw. Boundary   =       4.902
21  Normalized Wave Length          =      12.144
22  "Sigma"                         =      27.573
23  Ursell Number                   =      30.084
24  Surf Similarity Parameter       =       4.400
25
26  SLOPE PROPERTIES
27
28  Friction Factor             =       0.300000
29  Norm. Friction Factor       =       4.135970
30  Norm. Horiz. Length of
31        Computation Domain    =       0.711121
32  Number of Segments          =       1
33  ---------------------------------
34   SEGMENT     WBSEG(I)    TBSLOP(I)
35         I       feet
36  ---------------------------------
37         1   60.000000     0.400000
38  ---------------------------------
39
40  PARAMETERS FOR ARMOR STABILITY AND MOVEMENT
41
42  Armor Friction Factor       =     1.192
43  Specific Gravity            =     2.710
44  Area Coefficient       C2   =     0.900
45  Volume Coefficient     C3   =     0.660
46  Drag Coefficient       CD   =     0.500
47  Lift Coefficient       CL   =     0.180
48  Inertia Coefficient    CM   =     1.500
49  Norm. Upper and Lower Bounds of du/dt
50                         AMAX =     1.000
51                         AMIN =    -0.800
52
```

Table C-6 continued

```
 53     COMPUTATION PARAMETERS
 54
 55     Total Number of Spatial Nodes      JE =      161
 56     Number of Nodes Along Bottom Below SWL
 57                                     INITS =      100
 58     Normalized Delta x          =        0.444451D-02
 59     Damping Coefficients    x1 =           1.000
 60                             x2 =           1.000
 61     Computation Duration  TMAX =           6.000000 Wave Periods
 62     Minimum Allowable NONE          NONEM =     1200
 63     Time Series Stored at Rate NRATE
 64          per Wave Period with       NRATE =      200
 65     Statistical Calculations Are Performed Excluding
 66          the First MSTAT Computation Units
 67          with                       MSTAT =        5
 68     Computation of Armor Stability or Movement is Performed
 69          Excluding the First MSTAB Computation Units
 70          with                       MSTAB =        5
 71     Spatial Variations are Stored NTIMES at Equal
 72          Intervals per Wave Period from
 73          Computation Unit MSAVA1 to MSAVA2,
 74          Inclusive, with           NTIMES =        4
 75                                    MSAVA1 =        6
 76                                    MSAVA2 =        6
 77
 78     REFLECTION COEFFICIENTS
 79
 80     r1 =     0.553
 81     r2 =     0.546
 82     r3 =     0.544
 83
 84     RUNUP, RUNDOWN, SETUP
 85
 86     Largest Node Number Reached by Computational Waterline
 87                                                 JMAX =      140
 88     ------------------------------------------------------------
 89           I   DELTAR(I)     RUNUP(I)    RUNDOWN(I)    SETUP(I)
 90               [inch]          R            Rd            Zr
 91     ------------------------------------------------------------
 92           1     0.157        1.886        0.665        1.326
 93           2     0.787        1.855       -0.283        0.909
 94           3     1.575        1.843       -0.714        0.705
 95     ------------------------------------------------------------
 96
 97     WAVE SET-DOWN OR SETUP
 98
 99     Average value of ETAI =     0.000000
100                      ETAR =     0.015341
101
```

Table C-6 continued

```
102     TIME-AVERAGED ENERGY BALANCE
103
104     Normalized Energy Flux:
105     . at Seaw. Boundary   A =  0.202313D+00
106     . at Landw. Boundary  B =  0.000000D+00
107     Normalized Rate of Energy Dissipation
108     in the Computation Domain, Due to:
109     . Bottom Friction     C =  0.198084D+00
110     . Wave Breaking       D =  0.423137D-02
111     Additional Term       E =  0.000000D+00
112     Calculation 1:
113                   F = A-B =  0.202313D+00
114                 G = C+D+E =  0.202316D+00
115       Must H=0, but H = G-F =  0.231456D-05
116       % error  100G/E =          0.00
117     Approximate Energy Flux, Based on
118     Linear Long Wave, Due to:
119     . Incident Wave at Seaw. Boundary     P =  0.276755D+00
120     . Reflected Wave at Seaw. Boundary    Q =  0.818756D-01
121     Calculation 2:
122     Net Energy Flux at Seaw. Bndry. S = P-Q =  0.194879D+00
123     % Error at Seaward Boundary  100(S-A)/A =         -3.67
124
125     STABILITY NUMBER
126
127     Largest Node Number for Which Armor Stability
128       Computation Performed  JSTABM =        139
129     Critical Stability Number   Nsc =      2.399
130     At Node Number                  =        85
131     At Time                         =      5.372692
```

Table C-7: Primary input data file for IBREAK computation of "Ahrens 1975 Test 32 Zero Damage".

```
 1          4                          --> NLINES
 2     ------------------------------------------------------------------------
 3     Ahrens 1975 Test 32 Zero Damage
 4     Computed using IBREAK
 5     ------------------------------------------------------------------------
 6     12       1                      --> IJOB,ISTAB,NSTAB
 7     2                               --> ISYST
 8     1                               --> IBOT
 9     0                               --> INONCT
10     1                               --> IENERG
11     1  no wave data file            --> IWAVE,FINP2
12     101   12000   5   0   2         --> ISAVA-B-C,NSAVA,NTIMES,NNOD1,NNOD2
13     00000     0                     --> IREQ,IELEV,IV,IDUDT,ISNR,NNREQ
14        14000                        --> NTOP
15         2000                        --> NONE
16           10                        --> NJUM1
17          100                        --> S
18          .300000                    --> FWP
19         1.000000     1.000000       --> X1,X2
20          .001000                    --> DELTA(normalized)
21            3                        --> NDELR
22          .157480                    --> DELRP(1)(inches)
23          .787402                    --> DELRP(2)
24         1.574803                    --> DELRP(3)
25         3.390000     5.700000       --> HREFP(feet),TP(seconds)
26          .942100      .942100       --> KSREF,KSSEA
27        15.000000                    --> DSEAP(feet)
28          .285714                    --> TSLOPS
29            1                        --> NBSEG
30        85.000000      .285714       --> WBSEG(1)(feet),TBSLOP(1)
31            1                                --> NJUM2
32          .900000      .660000     2.710000  --> C2,C3,SG
33          .500000      .400000     1.500000  --> CD,CL,CM
34         1.191754                            --> TANPHI
35         1.000000     -.800000               --> AMAX,AMIN
36         1.030000                            --> DAP(feet)
37          72  O072                  --> NODNO2(1),FNAME2(1)
38         119  O119                  -->       (2)       (2)
```

Table C-8: File ODOC produced by IBREAK computation of "Ahrens 1975 Test 32 Zero Damage".

```
 1  ------------------------------------------------------------------------
 2  Ahrens 1975 Test 32 Zero Damage
 3  Computed using IBREAK
 4  ------------------------------------------------------------------------
 5
 6  WAVE CONDITION
 7
 8  Stokes II Incident Wave at Seaward Boundary
 9  Norm. Maximum Surface Elev.     =     0.594242
10  Norm. Minimum Surface Elev.     =    -0.405758
11
12  Reference Wave Period           =     5.700000 sec.
13  Reference Wave Height           =     3.390000 feet
14  Depth at Seaward Boundary       =    15.000000 feet
15  Shoal. Coef. at Reference  Ks1 =     0.942
16                 at Seaw. Bdr. Ks2 =     0.942
17                 Ks = Ks2/Ks1      =     1.000
18  Norm. Depth at Seaw. Bdr.       =     4.425
19  Normalized Wave Length          =     7.560
20  "Sigma"                         =    17.567
21  Ursell Number                   =    12.917
22  Surf Similarity Parameter       =     2.002
23
24  SLOPE PROPERTIES
25
26  Friction Factor             =     0.300000
27  Norm. Friction Factor       =     2.635082
28  Norm. Horiz. Length of
29        Computation Domain    =     1.419329
30  Number of Segments          =        1
31  --------------------------------
32   SEGMENT     WBSEG(I)    TBSLOP(I)
33         I       feet
34  --------------------------------
35         1   85.000000     0.285714
36
37  COMPUTATION PARAMETERS
38
39  Normalized Delta x    =   0.881571D-02
40  Normalized Delta t    =   0.500000D-03
41  Normalized DELTA      =   0.100000E-02
42  Damping Coeff.     x1 =      1.000
43                     x2 =      1.000
44  Num. Stab. Indicator  =      3.511
45  Total Number of Time Steps        NTOP =     14000
46  Number of Time Steps in 1 Wave Period
47                                    NONE =      2000
48  Total Number of Spatial Nodes       JE =       162
49  Number of Nodes Along Bottom Below SWL
50                                       S =       100
51  Storing Temporal Variations at Every
52                                    NJUM1 =        10 Time Steps
53  Armor Stability Number Computed
54                         at Every   NJUM2 =         1 Time Steps
55
```

Table C-8 continued

PARAMETERS FOR ARMOR STABILITY AND MOVEMENT

Armor Friction Factor		=	1.192
Specific Gravity		=	2.710
Area Coefficient	C2	=	0.900
Volume Coefficient	C3	=	0.660
Drag Coefficient	CD	=	0.500
Lift Coefficient	CL	=	0.400
Inertia Coefficient	CM	=	1.500
Armor Diameter		=	1.030000 feet

REFLECTION COEFFICIENTS

r1 = 0.173
r2 = 0.185
r3 = 0.176

RUNUP, RUNDOWN, SETUP

Largest Node Number Reached by Computational Waterline JMAX = 130

I	DELTAR(I) [inch]	RUNUP(I) R	RUNDOWN(I) Rd	SETUP(I) Zr
1	0.157	1.284	0.892	1.116
2	0.787	1.275	0.538	0.976
3	1.575	1.265	0.309	0.887

ARMOR UNITS MOVEMENT

Number of Units Moved	=	64
Number of Units Stopped	=	34

WAVE SET-DOWN OR SETUP

Average value of ETAI	=	0.000000
ETAR	=	0.019853

Table C-8 continued

```
 95     TIME-AVERAGED ENERGY BALANCE
 96
 97     Normalized Energy Flux:
 98     . at Seaw. Boundary    A =  0.271704D+00
 99     . at Landw. Boundary   B =  0.000000D+00
100     Normalized Rate of Energy Dissipation
101     in the Computation Domain, Due to:
102     . bottom friction      C =  0.205248D+00
103     . wave breaking        D =  0.664560D-01
104     Calculation 1:
105                    E = A-B =  0.271704D+00
106                    F = C+D =  0.271704D+00
107       Must G=0, but G = F-E =  0.512847D-07
108       % error  100G/E =           0.00
109     Approximate Energy Flux, Based on
110     Linear Long Wave, Due to:
111     . incident wave at seaw. boundary      P =  0.262939D+00
112     . reflected wave at seaw. boundary     Q =  0.814467D-02
113     Calculation 2:
114     Net Energy Flux at Seaw. Bndry. S = P-Q =  0.254795D+00
115     % Error at Seaward Boundary  100(S-A)/A =         -6.22
```

Table C-9: Primary input data file for RBREAK computation of "Ahrens 1975 Test 32 Zero Damage".

```
 1          3                          --> NLINES
 2     ------------------------------------------------------------------------
 3     Ahrens 1975 Test 32 Zero Damage
 4     ------------------------------------------------------------------------
 5     12                               --> IJOB,ISTAB
 6     2                                --> ISYST
 7     1                                --> IBOT
 8     0                                --> INONCT
 9     1                                --> IENERG
10     1       0  no 2nd data file      --> IWAVE,MSTAT,FINP2
11     1       6       6       4        --> ISAVA,MSAVA1,MSAVA2,NTIMES
12     0       0                        --> ISAVB,NNODB
13     1       2                        --> ISAVC,NNODC
14     00000       0                    --> IREQ,IELEV,IV,IDUDT,ISNR,NREQ
15          100                         --> INITS
16            1                         --> MULTIF
17             .001000                  --> DELTA (normalized)
18            7.000000                  --> TMAX  (normalized)
19          200                         --> NRATE
20            3                         --> NDELR
21           .157480                    --> DELRP(1)(inches)
22           .787402                    -->       (2)
23          1.574803                    -->       (3)
24          1.000000     1.000000       --> X1,X2
25          3.390000     5.700000       --> HREFP(feet),TP(seconds)
26          1.000000     1.000000       --> KSREF,KSSEA
27           .300000                    --> FWP
28         15.000000                    --> DSEAP(feet)
29           .285714                    --> TSLOPS
30            1                         --> NBSEG
31         85.000000       .285714      --> WBSEG(1)(feet),TBSLOP(1)
32           .900000       .660000        2.710000  --> C2,C3,SG
33           .500000       .400000        1.500000  --> CD,CL,CM
34          1.191754                                 --> TANPHI
35          1.000000     -.800000                    --> AMAX,AMIN
36         72   119                                  --> NODC(1)-(2)
37          1.030000                                 --> DAP(feet)
```

Table C-10: File BINPUT produced by BEFORR computation of "Ahrens 1975 Test 32 Zero Damage".

```
 1          3
 2    ------------------------------------------------------------------------
 3    Ahrens 1975 Test 32 Zero Damage
 4    ------------------------------------------------------------------------
 5          7
 6        600
 7       1200
 8          1     600     600   0.100000000D-02
 9          2    1200    1200   0.100000000D-02
10          3    1200    1200   0.100000000D-02
11          4    1200    1200   0.100000000D-02
12          5    1200    1200   0.100000000D-02
13          6    1200    1200   0.100000000D-02
14          7    1200    1200   0.100000000D-02
```

Table C-11: File ODOC produced by RBREAK computation of "Ahrens 1975 Test 32 Zero Damage".

```
 1  ------------------------------------------------------------------------
 2  Ahrens 1975 Test 32 Zero Damage
 3  ------------------------------------------------------------------------
 4
 5  WAVE CONDITION
 6
 7  Stokes II Incident Wave at Seaward Boundary
 8  Norm. Maximum Surface Elev.     =       0.594242
 9  Norm. Minimum Surface Elev.     =      -0.405758
10
11  Reference Wave Period           =       5.700000 sec.
12  Reference Wave Height           =       3.390000 feet
13  Depth at Seaward Boundary       =      15.000000 feet
14  Shoal. Coef. at Reference  Ks1 =       1.000
15               at Seaw. Bdr. Ks2 =       1.000
16               Ks = Ks2/Ks1      =       1.000
17  Norm. Depth at Seaw. Boundary   =       4.425
18  Normalized Wave Length          =       7.560
19  "Sigma"                         =      17.567
20  Ursell Number                   =      12.917
21  Surf Similarity Parameter       =       2.002
22
23  SLOPE PROPERTIES
24
25  Friction Factor             =        0.300000
26  Norm. Friction Factor       =        2.635082
27  Norm. Horiz. Length of
28        Computation Domain    =        1.419329
29  Number of Segments          =        1
30  --------------------------------
31   SEGMENT     WBSEG(I)    TBSLOP(I)
32         I       feet
33  --------------------------------
34         1   85.000000     0.285714
35  --------------------------------
36
37  PARAMETERS FOR ARMOR STABILITY AND MOVEMENT
38
39  Armor Friction Factor       =     1.192
40  Specific Gravity            =     2.710
41  Area Coefficient       C2   =     0.900
42  Volume Coefficient     C3   =     0.660
43  Drag Coefficient       CD   =     0.500
44  Lift Coefficient       CL   =     0.400
45  Inertia Coefficient    CM   =     1.500
46  Armor Diameter              =     1.030000 feet
47
```

Table C-11 continued

```
 48     COMPUTATION PARAMETERS
 49
 50     Total Number of Spatial Nodes      JE =      162
 51     Number of Nodes Along Bottom Below SWL
 52                                       INITS =      100
 53     Normalized Delta x          =        0.881571D-02
 54     Damping Coefficients    x1 =           1.000
 55                             x2 =           1.000
 56     Computation Duration  TMAX =           7.000000 Wave Periods
 57     Minimum Allowable NONE          NONEM =      600
 58     Time Series Stored at Rate NRATE
 59          per Wave Period with       NRATE =      200
 60     Statistical Calculations Are Performed Excluding
 61          the First MSTAT Computation Units
 62          with                       MSTAT =        6
 63     Computation of Armor Stability or Movement is Performed
 64          Excluding the First MSTAB Computation Units
 65          with                       MSTAB =        0
 66     Spatial Variations are Stored NTIMES at Equal
 67          Intervals per Wave Period from
 68          Computation Unit MSAVA1 to MSAVA2,
 69          Inclusive, with            NTIMES =       4
 70                                     MSAVA1 =       6
 71                                     MSAVA2 =       6
 72
 73     REFLECTION COEFFICIENTS
 74
 75     r1 =     0.173
 76     r2 =     0.185
 77     r3 =     0.176
 78
 79     RUNUP, RUNDOWN, SETUP
 80
 81     Largest Node Number Reached by Computational Waterline
 82                                                 JMAX =      130
 83     ------------------------------------------------------------
 84           I   DELTAR(I)     RUNUP(I)    RUNDOWN(I)    SETUP(I)
 85               [inch]          R            Rd            Zr
 86     ------------------------------------------------------------
 87           1     0.157        1.309        0.888         1.117
 88           2     0.787        1.282        0.538         0.977
 89           3     1.575        1.268        0.311         0.890
 90     ------------------------------------------------------------
 91
 92     WAVE SET-DOWN OR SETUP
 93
 94     Average value of ETAI =     0.000000
 95                      ETAR =     0.019867
 96
```

Table C-11 continued

```
 97     TIME-AVERAGED ENERGY BALANCE
 98
 99     Normalized Energy Flux:
100     . at Seaw. Boundary    A =  0.271688D+00
101     . at Landw. Boundary   B =  0.000000D+00
102     Normalized Rate of Energy Dissipation
103     in the Computation Domain, Due to:
104     . Bottom Friction      C =  0.205318D+00
105     . Wave Breaking        D =  0.663512D-01
106     Additional Term        E =  0.000000D+00
107     Calculation 1:
108                    F = A-B =  0.271688D+00
109                  G = C+D+E =  0.271669D+00
110       Must H=0, but H = G-F = -0.184617D-04
111       % error  100G/E =          -0.01
112     Approximate Energy Flux, Based on
113     Linear Long Wave, Due to:
114     . Incident Wave at Seaw. Boundary      P =  0.262939D+00
115     . Reflected Wave at Seaw. Boundary     Q =  0.815660D-02
116     Calculation 2:
117     Net Energy Flux at Seaw. Bndry. S = P-Q =  0.254783D+00
118     % Error at Seaward Boundary  100(S-A)/A =         -6.22
119
120     ARMOR UNITS MOVEMENT
121
122     Largest Node Number for Which Armor Movement
123       Computation Performed  JSTABM =      130
124     Number of Units Moved     NMOVE =       63
125     Number of Units Stopped   NSTOP =       20
```

Table C-12: Primary input data file for IBREAK computation of "Saville 1955 Run 10".

```
 1          4                        --> NLINES
 2    ------------------------------------------------------------------------
 3    Saville 1955 Run 10
 4    Computed using IBREAK
 5    ------------------------------------------------------------------------
 6    20       0                     --> IJOB,ISTAB,NSTAB
 7    1                              --> ISYST
 8    1                              --> IBOT
 9    0                              --> INONCT
10    1                              --> IENERG
11    1  no wave data file           --> IWAVE,FINP2
12    100   32000   5   0   0        --> ISAVA-B-C,NSAVA,NTIMES,NNOD1,NNOD2
13    00000     0                    --> IREQ,IELEV,IV,IDUDT,ISNR,NNREQ
14       40000                       --> NTOP
15        8000                       --> NONE
16          40                       --> NJUM1
17         120                       --> S
18          .050000                  --> FWP
19         1.000000     1.000000     --> X1,X2
20          .001000                  --> DELTA(normalized)
21           3                       --> NDELR
22         1.000000                  --> DELRP(1) (millimeters)
23         3.000000                  -->       (2)
24         5.000000                  -->       (3)
25          .107576     2.621810     --> HREFP(meters),TP(seconds)
26         1.000000     1.034000     --> KSREF,KSSEA
27          .430000                  --> DSEAP(meters)
28          .333333                  --> TSLOPS
29           3                       --> NBSEG
30         2.689000      .100000     --> WBSEG(1) (meters),TBSLOP(1)
31          .807000      .333333     -->       (2)                 (2)
32          .090000      .000000     -->       (3)                 (3)
```

Table C-13: File ODOC produced by IBREAK computation of "Saville 1955 Run 10".

```
 1  ------------------------------------------------------------------------
 2  Saville 1955 Run 10
 3  Computed using IBREAK
 4  ------------------------------------------------------------------------
 5
 6  WAVE CONDITION
 7
 8  Cnoidal Incident Wave at Seaward Boundary
 9  1-m =       0.608412027D-01
10  E   =       0.107076819D+01
11  K   =       0.281408330D+01
12  Norm. Maximum Surface Elev.     =      0.682057
13  Norm. Minimum Surface Elev.     =     -0.351943
14
15  Reference Wave Period           =      2.621810 sec.
16  Reference Wave Height           =      0.107576 meters
17  Depth at Seaward Boundary       =      0.430000 meters
18  Shoal. Coef. at Reference  Ks1  =      1.000
19               at Seaw. Bdr. Ks2  =      1.034
20               Ks = Ks2/Ks1       =      1.034
21  Norm. Depth at Seaw. Bdr.       =      3.997
22  Normalized Wave Length          =     12.383
23  "Sigma"                         =     25.037
24  Ursell Number                   =     39.665
25  Surf Similarity Parameter       =      3.329
26
27  SLOPE PROPERTIES
28
29  Friction Factor             =      0.050000
30  Norm. Friction Factor       =      0.625919
31  Norm. Horiz. Length of
32        Computation Domain    =      1.325054
33  Number of Segments          =         3
34  --------------------------------
35   SEGMENT     WBSEG(I)    TBSLOP(I)
36         I       meters
37  --------------------------------
38          1    2.689000    0.100000
39          2    0.807000    0.333333
40          3    0.090000    0.000000
41
42  COMPUTATION PARAMETERS
43
44  Normalized Delta x    =   0.981521D-02
45  Normalized Delta t    =   0.125000D-03
46  Normalized DELTA      =   0.100000E-02
47  Damping Coeff.     x1 =      1.000
48                     x2 =      1.000
49  Num. Stab. Indicator =     16.180
50  Total Number of Time Steps        NTOP =     40000
51  Number of Time Steps in 1 Wave Period
52                                    NONE =      8000
53  Total Number of Spatial Nodes       JE =       136
54  Number of Nodes Along Bottom Below SWL
55                                       S =       120
56  Storing Temporal Variations at Every
57                                    NJUM1 =        40 Time Steps
```

Table C-13 continued

```
 58
 59     REFLECTION COEFFICIENTS
 60
 61     r1 =      0.606
 62     r2 =      0.579
 63     r3 =      0.579
 64
 65     RUNUP, RUNDOWN, SETUP
 66
 67     Largest Node Number Reached by Computational Waterline
 68                                                  JMAX =      136
 69     ------------------------------------------------------------
 70           I   DELTAR(I)     RUNUP(I)    RUNDOWN(I)    SETUP(I)
 71                  [mm]          R            Rd           Zr
 72     ------------------------------------------------------------
 73           1      1.000        1.297        0.130        1.048
 74           2      3.000        1.297       -0.937        0.423
 75           3      5.000        1.297       -0.958        0.327
 76
 77     OVERTOPPING
 78
 79     Norm. Avg. Overtopping Rate     =   0.455369D-01
 80     Norm. Avg. Flow at Seaw. Bdr.   =   0.453231D-01
 81     Norm. Max. Overtopping Rate     =   0.256392D+00
 82     Max. Rate Occurs at     0.439000 Within Interval [NSTAT,NTOP]
 83     Overtopping Duration = 1.000000 Within Interval [NSTAT,NTOP]
 84     The last two quantities are relative to the specified
 85       interval taken as unity
 86
 87     WAVE SET-DOWN OR SETUP
 88
 89     Average value of ETAI =      0.000000
 90                      ETAR =     -0.005894
 91
 92     TIME-AVERAGED ENERGY BALANCE
 93
 94     Normalized Energy Flux:
 95     . at Seaw. Boundary   A =  0.186835D+00
 96     . at Landw. Boundary  B =  0.698877D-01
 97     Normalized Rate of Energy Dissipation
 98     in the Computation Domain, Due to:
 99     . bottom friction     C =  0.846857D-01
100     . wave breaking       D =  0.322616D-01
101     Calculation 1:
102                    E = A-B =  0.116947D+00
103                    F = C+D =  0.116947D+00
104       Must G=0, but G = F-E =  0.277556D-16
105       % error  100G/E =           0.00
106     Approximate Energy Flux, Based on
107     Linear Long Wave, Due to:
108     . incident wave at seaw. boundary      P =  0.267195D+00
109     . reflected wave at seaw. boundary     Q =  0.895412D-01
110     Calculation 2:
111     Net Energy Flux at Seaw. Bndry. S = P-Q =  0.177653D+00
112     % Error at Seaward Boundary  100(S-A)/A =         -4.91
```

Table C-14: Primary input data file for RBREAK computation of "Saville 1955 Run 10".

```
 1          3                          --> NLINES
 2     ------------------------------------------------------------------------
 3     Saville 1955 Run 10
 4     ------------------------------------------------------------------------
 5     20                               --> IJOB,ISTAB
 6     1                                --> ISYST
 7     1                                --> IBOT
 8     0                                --> INONCT
 9     1                                --> IENERG
10     1       0  no 2nd data file      --> IWAVE,MSTAT,FINP2
11     1       5       5       4        --> ISAVA,MSAVA1,MSAVA2,NTIMES
12     0       0                        --> ISAVB,NNODB
13     0       0                        --> ISAVC,NNODC
14     00000       0                    --> IREQ,IELEV,IV,IDUDT,ISNR,NREQ
15          120                         --> INITS
16            1                         --> MULTIF
17             .001000                  --> DELTA (normalized)
18            5.000000                  --> TMAX  (normalized)
19          200                         --> NRATE
20            3                         --> NDELR
21          1.000000                    --> DELRP(1)(millimeters)
22          3.000000                    -->       (2)
23          5.000000                    -->       (3)
24          1.000000      1.000000      --> X1,X2
25           .107576      2.621810      --> HREFP(meters),TP(seconds)
26          1.000000      1.034000      --> KSREF,KSSEA
27           .050000                    --> FWP
28           .430000                    --> DSEAP(meters)
29           .333333                    --> TSLOPS
30           3                          --> NBSEG
31          2.689000       .100000      --> WBSEG(1)(meters),TBSLOP(1)
32           .807000       .333333      -->       (2)               (2)
33           .090000       .000000      -->       (3)               (3)
```

Table C-15: File BINPUT produced by BEFORR computation of "Saville 1955 Run 10".

```
1           3
2     ------------------------------------------------------------------------
3     Saville 1955 Run 10
4     ------------------------------------------------------------------------
5           5
6         600
7        3800
8           1    3800    3800   0.100000000D-02
9           2    3800    3800   0.100000000D-02
10          3    3800    3800   0.100000000D-02
11          4    3800    3800   0.100000000D-02
12          5    3800    3800   0.100000000D-02
```

Table C-16: File ODOC produced by RBREAK computation of "Saville 1955 Run 10".

```
 1  ------------------------------------------------------------------------
 2  Saville 1955 Run 10
 3  ------------------------------------------------------------------------
 4
 5  WAVE CONDITION
 6
 7  Cnoidal Incident Wave at Seaward Boundary
 8  1-m =       0.608412027D-01
 9  E   =       0.107076819D+01
10  K   =       0.281408330D+01
11  Norm. Maximum Surface Elev.     =        0.682057
12  Norm. Minimum Surface Elev.     =       -0.351943
13
14  Reference Wave Period           =        2.621810 sec.
15  Reference Wave Height           =        0.107576 meters
16  Depth at Seaward Boundary       =        0.430000 meters
17  Shoal. Coef. at Reference  Ks1  =        1.000
18               at Seaw. Bdr. Ks2  =        1.034
19               Ks = Ks2/Ks1       =        1.034
20  Norm. Depth at Seaw. Boundary   =        3.997
21  Normalized Wave Length          =       12.383
22  "Sigma"                         =       25.037
23  Ursell Number                   =       39.665
24  Surf Similarity Parameter       =        3.329
25
26  SLOPE PROPERTIES
27
28  Friction Factor             =        0.050000
29  Norm. Friction Factor       =        0.625919
30  Norm. Horiz. Length of
31        Computation Domain    =        1.325054
32  Number of Segments          =        3
33  ---------------------------------
34   SEGMENT     WBSEG(I)    TBSLOP(I)
35         I       meters
36  ---------------------------------
37           1    2.689000     0.100000
38           2    0.807000     0.333333
39           3    0.090000     0.000000
40  ---------------------------------
41
```

Table C-16 continued

```
COMPUTATION PARAMETERS

Total Number of Spatial Nodes      JE =      136
Number of Nodes Along Bottom Below SWL
                                   INITS =   120
Normalized Delta x          =        0.981521D-02
Damping Coefficients    x1 =            1.000
                        x2 =            1.000
Computation Duration  TMAX =            5.000000 Wave Periods
Minimum Allowable NONE          NONEM =   600
Time Series Stored at Rate NRATE
     per Wave Period with       NRATE =   200
Statistical Calculations Are Performed Excluding
     the First MSTAT Computation Units
     with                       MSTAT =     4
Spatial Variations are Stored NTIMES at Equal
     Intervals per Wave Period from
     Computation Unit MSAVA1 to MSAVA2,
     Inclusive, with            NTIMES =    4
                                MSAVA1 =    5
                                MSAVA2 =    5

REFLECTION COEFFICIENTS

r1 =     0.605
r2 =     0.579
r3 =     0.579

RUNUP, RUNDOWN, SETUP

Largest Node Number Reached by Computational Waterline
                                                JMAX =     136
-------------------------------------------------------------
       I   DELTAR(I)     RUNUP(I)    RUNDOWN(I)    SETUP(I)
             [mm]          R            Rd           Zr
-------------------------------------------------------------
       1     1.000        1.297        0.143        1.049
       2     3.000        1.297       -0.937        0.422
       3     5.000        1.297       -0.958        0.327
-------------------------------------------------------------

OVERTOPPING

Norm. Average Overtopping Rate          =    0.455060D-01
Norm. Average Flow at Seaward Boundary  =    0.453095D-01
Norm. Maximum Overtopping Rate          =    0.256112D+00
      Occuring at Normalized Time    t =          4.439211
Norm. Overtopping Duration              =          1.000000
Notes: Norm. duration of overtopping computation =          1.000000
       Norm. duration of RBREAK computation      =          5.000000

WAVE SET-DOWN OR SETUP

Average value of ETAI =     0.000000
                  ETAR =    -0.005891
```

Table C-16 continued

```
 98     TIME-AVERAGED ENERGY BALANCE
 99
100     Normalized Energy Flux:
101     . at Seaw. Boundary   A =  0.186805D+00
102     . at Landw. Boundary  B =  0.698364D-01
103     Normalized Rate of Energy Dissipation
104     in the Computation Domain, Due to:
105     . Bottom Friction     C =  0.846254D-01
106     . Wave Breaking       D =  0.323437D-01
107     Additional Term       E =  0.000000D+00
108     Calculation 1:
109                     F = A-B =  0.116969D+00
110                   G = C+D+E =  0.116969D+00
111       Must H=0, but H = G-F = -0.277556D-16
112       % error  100G/E =          0.00
113     Approximate Energy Flux, Based on
114     Linear Long Wave, Due to:
115     . Incident Wave at Seaw. Boundary     P =  0.267195D+00
116     . Reflected Wave at Seaw. Boundary    Q =  0.895754D-01
117     Calculation 2:
118     Net Energy Flux at Seaw. Bndry. S = P-Q =  0.177619D+00
119     % Error at Seaward Boundary  100(S-A)/A =         -4.92
```

Table C-17: Primary input data file for IBREAK computation of "Seelig 1980 Run 2".

```
 1          4                        --> NLINES
 2     ------------------------------------------------------------------------
 3     Seelig 1980 Run 2
 4     Computed using IBREAK
 5     ------------------------------------------------------------------------
 6     30       0                    --> IJOB,ISTAB,NSTAB
 7     1                             --> ISYST
 8     2                             --> IBOT
 9     0                             --> INONCT
10     1                             --> IENERG
11     1  no wave data file          --> IWAVE,FINP2
12     100   12000   5   0   0       --> ISAVA-B-C,NSAVA,NTIMES,NNOD1,NNOD2
13     00000     0                   --> IREQ,IELEV,IV,IDUDT,ISNR,NNREQ
14        15000                      --> NTOP
15         3000                      --> NONE
16           15                      --> NJUM1
17          300                      --> S
18          .050000                  --> FWP
19         2.000000      2.000000    --> X1,X2
20          .001000                  --> DELTA(normalized)
21           0                       --> NDELR
22          .193000      2.390000    --> HREFP(meters),TP(seconds)
23          .931989       .931989    --> KSREF,KSSEA
24          .900000                  --> DSEAP(meters)
25          .666667                  --> TSLOPS
26           3                       --> NBSEG
27          .000000       .900000    --> XBSEG(1)(meters),ZBSEG(1)(meters)
28         1.125000       .150000    -->       (2)               (2)
29         1.425000       .150000    -->       (3)               (3)
30         2.550000       .900000    -->       (4)               (4)
```

Table C-18: File ODOC produced by IBREAK computation of "Seelig 1980 Run 2".

```
 1  ------------------------------------------------------------------------
 2  Seelig 1980 Run 2
 3  Computed using IBREAK
 4  ------------------------------------------------------------------------
 5
 6  WAVE CONDITION
 7
 8  Stokes II Incident Wave at Seaward Boundary
 9  Norm. Maximum Surface Elev.     =     0.582547
10  Norm. Minimum Surface Elev.     =    -0.417453
11
12  Reference Wave Period           =     2.390000 sec.
13  Reference Wave Height           =     0.193000 meters
14  Depth at Seaward Boundary       =     0.900000 meters
15  Shoal. Coef. at Reference  Ks1  =     0.932
16                at Seaw. Bdr. Ks2 =     0.932
17                Ks = Ks2/Ks1      =     1.000
18  Norm. Depth at Seaw. Bdr.       =     4.663
19  Normalized Wave Length          =     7.053
20  "Sigma"                         =    17.039
21  Ursell Number                   =    10.669
22  Surf Similarity Parameter       =     4.532
23  Depth at Landward Boundary      =     0.900000 meters
24
25  SLOPE PROPERTIES
26
27  Friction Factor             =     0.050000
28  Norm. Friction Factor       =     0.425984
29  Norm. Horiz. Length of
30        Computation Domain    =     0.775406
31  Number of Segments          =        3
32  --------------------------------
33   SEGMENT     XBSEG(I)     ZBSEG(I)
34         I       meters       meters
35  --------------------------------
36         1    0.000000     0.900000
37         2    1.125000     0.150000
38         3    1.425000     0.150000
39         4    2.550000     0.900000
40
41  COMPUTATION PARAMETERS
42
43  Normalized Delta x   =   0.259333D-02
44  Normalized Delta t   =   0.333333D-03
45  Normalized DELTA     =   0.100000E-02
46  Damping Coeff.    x1 =     2.000
47                    x2 =     2.000
48  Num. Stab. Indicator =     1.020
49  Total Number of Time Steps        NTOP =    15000
50  Number of Time Steps in 1 Wave Period
51                                    NONE =     3000
52  Total Number of Spatial Nodes       JE =      300
53  Storing Temporal Variations at Every
54                                    NJUM1 =       15 Time Steps
55
```

Table C-18 continued

```
 56     REFLECTION COEFFICIENTS
 57
 58     r1 =      0.494
 59     r2 =      0.550
 60     r3 =      0.504
 61
 62     WAVE SET-DOWN OR SETUP
 63
 64     Average value of ETAI =      0.000000
 65                      ETAR =     -0.077559
 66                      ETAT =      0.073361
 67     MWL Difference       =      0.150920
 68
 69     TRANSMISSION
 70
 71     Transmission Coefficient T1 =      0.741
 72                              T2 =      0.722
 73                              T3 =      0.691
 74     Norm. Avg. Flow at Seaw. Bdr.  =      0.196176
 75     Norm. Avg. Flow at Landw. Bdr. =      0.196442
 76     Average of the Above Two       =      0.196309
 77
 78     TIME-AVERAGED ENERGY BALANCE
 79
 80     Normalized Energy Flux:
 81     . at Seaw. Boundary   A =  0.219144D+00
 82     . at Landw. Boundary  B =  0.163349D+00
 83     Normalized Rate of Energy Dissipation
 84     in the Computation Domain, Due to:
 85     . bottom friction     C =  0.141936D-01
 86     . wave breaking       D =  0.416010D-01
 87     Calculation 1:
 88                    E = A-B =  0.557946D-01
 89                    F = C+D =  0.557946D-01
 90       Must G=0, but G = F-E = -0.138778D-16
 91       % error  100G/E =           0.00
 92     Approximate Energy Flux, Based on
 93     Linear Long Wave, Due to:
 94     . incident wave at seaw. boundary     P =  0.269931D+00
 95     . reflected wave at seaw. boundary    Q =  0.685893D-01
 96     . transmitted wave at landw. bndry.   R =  0.129051D+00
 97     Calculation 2:
 98     Net Energy Flux at Seaw. Bndry. S = P-Q =  0.201342D+00
 99     % Error at Seaward Boundary  100(S-A)/A =         -8.12
100     % Error at Landward Boundary 100(R-B)/B =        -21.00
```

Table C-19: Primary input data file for RBREAK computation of "Seelig 1980 Run 2".

```
 1          3                          --> NLINES
 2    ------------------------------------------------------------------------
 3    Seelig 1980 Run 2
 4    ------------------------------------------------------------------------
 5    30                              --> IJOB,ISTAB
 6    1                               --> ISYST
 7    2                               --> IBOT
 8    0                               --> INONCT
 9    1                               --> IENERG
10    1       0  no 2nd data file     --> IWAVE,MSTAT,FINP2
11    1       5       5       4      --> ISAVA,MSAVA1,MSAVA2,NTIMES
12    0       0                       --> ISAVB,NNODB
13    0       0                       --> ISAVC,NNODC
14    00000       0                   --> IREQ,IELEV,IV,IDUDT,ISNR,NREQ
15         300                        --> INITS
16           1                        --> MULTIF
17            .001000                 --> DELTA (normalized)
18           5.000000                 --> TMAX  (normalized)
19         200                        --> NRATE
20           0                        --> NDELR
21         2.000000     2.000000      --> X1,X2
22          .193000     2.390000      --> HREFP(meters),TP(seconds)
23         1.000000     1.000000      --> KSREF,KSSEA
24          .050000                   --> FWP
25          .900000                   --> DSEAP(meters)
26          .666667                   --> TSLOPS
27          3                         --> NBSEG
28          .000000      .900000      --> XBSEG(1)(meters),ZBSEG(1)(meters)
29         1.125000      .150000      -->      (2)              (2)
30         1.425000      .150000      -->      (3)              (3)
31         2.550000      .900000      -->      (4)              (4)
```

Table C-20: File BINPUT produced by BEFORR computation of "Seelig 1980 Run 2".

```
 1           3
 2     ------------------------------------------------------------------------
 3     Seelig 1980 Run 2
 4     ------------------------------------------------------------------------
 5           5
 6        3000
 7        3000
 8           1    3000    3000   0.100000000D-02
 9           2    3000    3000   0.100000000D-02
10           3    3000    3000   0.100000000D-02
11           4    3000    3000   0.100000000D-02
12           5    3000    3000   0.100000000D-02
```

Table C-21: File ODOC produced by RBREAK computation of "Seelig 1980 Run 2".

```
 1  ------------------------------------------------------------------------
 2  Seelig 1980 Run 2
 3  ------------------------------------------------------------------------
 4
 5  WAVE CONDITION
 6
 7  Stokes II Incident Wave at Seaward Boundary
 8  Norm. Maximum Surface Elev.     =       0.582547
 9  Norm. Minimum Surface Elev.     =      -0.417453
10
11  Reference Wave Period           =       2.390000 sec.
12  Reference Wave Height           =       0.193000 meters
13  Depth at Seaward Boundary       =       0.900000 meters
14  Shoal. Coef. at Reference  Ks1  =       1.000
15               at Seaw. Bdr. Ks2  =       1.000
16               Ks = Ks2/Ks1       =       1.000
17  Norm. Depth at Seaw. Boundary   =       4.663
18  Normalized Wave Length          =       7.053
19  "Sigma"                         =      17.039
20  Ursell Number                   =      10.669
21  Surf Similarity Parameter       =       4.532
22  Depth at Landward Boundary      =       0.900000 meters
23
24  SLOPE PROPERTIES
25
26  Friction Factor             =         0.050000
27  Norm. Friction Factor       =         0.425984
28  Norm. Horiz. Length of
29        Computation Domain    =         0.775406
30  Number of Segments          =         3
31  --------------------------------
32   SEGMENT     XBSEG(I)     ZBSEG(I)
33         I       meters       meters
34  --------------------------------
35         1     0.000000     0.900000
36         2     1.125000     0.150000
37         3     1.425000     0.150000
38         4     2.550000     0.900000
39  --------------------------------
40
41  COMPUTATION PARAMETERS
42
43  Total Number of Spatial Nodes      JE =       300
44  Normalized Delta x           =        0.259333D-02
45  Damping Coefficients    x1 =             2.000
46                          x2 =             2.000
47  Computation Duration  TMAX =             5.000000 Wave Periods
48  Minimum Allowable NONE           NONEM =      3000
49  Time Series Stored at Rate NRATE
50       per Wave Period with       NRATE =       200
```

Table C-21 continued

```
 51     Statistical Calculations Are Performed Excluding
 52          the First MSTAT Computation Units
 53          with                          MSTAT =        4
 54     Spatial Variations are Stored NTIMES at Equal
 55          Intervals per Wave Period from
 56          Computation Unit MSAVA1 to MSAVA2,
 57          Inclusive, with              NTIMES =        4
 58                                       MSAVA1 =        5
 59                                       MSAVA2 =        5
 60
 61     REFLECTION COEFFICIENTS
 62
 63     r1 =      0.494
 64     r2 =      0.550
 65     r3 =      0.504
 66
 67     WAVE SET-DOWN OR SETUP
 68
 69     Average value of ETAI =      0.000000
 70                      ETAR =     -0.077559
 71                      ETAT =      0.073361
 72     MWL Difference       =      0.150920
 73
 74     TRANSMISSION
 75
 76     Transmission Coefficient T1 =      0.741
 77                              T2 =      0.722
 78                              T3 =      0.691
 79     Norm. Average Flow at Seaward Boundary  =         0.196175714
 80     Norm. Average Flow at Landward Boundary =         0.196441562
 81     Average of the Above Two                =         0.196308638
 82
 83     TIME-AVERAGED ENERGY BALANCE
 84
 85     Normalized Energy Flux:
 86     . at Seaw. Boundary   A =  0.219144D+00
 87     . at Landw. Boundary  B =  0.163349D+00
 88     Normalized Rate of Energy Dissipation
 89     in the Computation Domain, Due to:
 90     . Bottom Friction     C =  0.141936D-01
 91     . Wave Breaking       D =  0.416010D-01
 92     Additional Term       E =  0.000000D+00
 93     Calculation 1:
 94                    F = A-B =  0.557946D-01
 95                  G = C+D+E =  0.557946D-01
 96       Must H=0, but H = G-F = -0.138778D-16
 97       % error  100G/E =          0.00
 98     Approximate Energy Flux, Based on
 99     Linear Long Wave, Due to:
100     . Incident Wave at Seaw. Boundary     P =  0.269931D+00
101     . Reflected Wave at Seaw. Boundary    Q =  0.685893D-01
102     . Transmitted Wave at Landw. Bndry.   R =  0.129051D+00
103     Calculation 2:
104     Net Energy Flux at Seaw. Bndry. S = P-Q =  0.201342D+00
105     % Error at Seaward Boundary  100(S-A)/A =         -8.12
106     % Error at Landward Boundary 100(R-B)/B =        -21.00
```

APPENDIX D

EXAMPLE: INPUT WAVE TRAIN I FROM EXPERIMENT

This Appendix shows an example of random wave computations where the input wave train was obtained from a laboratory experiment. This example is Run I1 of Kobayashi, Cox, and Wurjanto (1990).

First, the experiment procedures are discussed so that the input data to the BEFORR-RBREAK computation can be understood.

- Bottom geometry is depicted in Figure D-1.
  - The seaward boundary for the numerical computation is taken to be the toe of the slope, where the water depth is 0.4m $\Longrightarrow$ `DSEAP`=0.4m under `ISYST`=1 (International System of Units).
  - The slope is uniform with an inclination of 1:3. A single (`NBSEG`=1) linear segment is sufficient to represent the slope. `IBOT`=1 (width-slope representation) is selected. `TBSLOP(1)`=$\frac{1}{3}$=.333333. Since the slope is uniform, `TSLOPS`=`TBSLOP(1)`=.333333. The horizontal distance from the toe of the slope to the waterline before wave generation is 3x0.4m=1.2m. To accommodate the waterline movement along the slope, the spatial domain of the numerical computation is ordinarily specified within the range of 1.5 to 2 times this horizontal distance. It is noted that BEFORR will inform a user during the computation if the specified slope is not long enough. For this computation `WBSLOP(1)`=2.1m, which amounts to 1.75 times the horizontal distance 1.2m.
  - The structure is relatively steep. The use of `INONCT`=0 is therefore recommended.
- Slope roughness. The roughness of the slope is represented by to the friction factor `FWP`. The slope in this experiment is similar to that of Kobayashi and Greenwald (1988) where the friction factor `FWP` was calibrated over a wide range of the surf similarity parameter ξ. The value of ξ for Run I1 is 1.68, which suggests the use of `FWP`=.05.
- The wave train at the toe of the slope. In the experiment, wave gages recorded the free surface elevations at the toe and seaward of the slope. The recorded total waves were separated into the incident waves $\eta_i(t)$ and the reflected waves $\eta_r(t)$. The duration of the wave trains in Run I1 was 185 seconds.
  - For the numerical computation, the seaward boundary is taken to be the *reference* location $\Longrightarrow$ `KSREF` and `KSSEA` can simply be taken as unity.
  - The measured *incident* wave train is used as the input wave train $\Longrightarrow$ `IWAVE`=2. `FINP2=tkcwi1` in the primary input data file specifies the name of the file containing

the input wave train. The wave train is plotted in Figure D-2 and is stored in the file `append.iwt` on the disk accompanying this report.

  - The significant wave height of the input wave train is taken to be the reference wave height $\Longrightarrow$ HREFP=0.0705m. The mean period of the input wave train is taken to be the reference wave period $\Longrightarrow$ TP=1.069s [reported as 1.07s in Kobayashi, Cox, and Wurjanto (1990)].
  - The normalized computation duration TMAX is obtained by normalizing the duration of the input wave train (185s) by the reference wave period (1.069s) $\Longrightarrow$ TMAX= 173.058934.

- The water depth used for the waterline measurement in this experiment was 27.5 mm. Accordingly, one of the two (NDELR=2) specified values of DELRP is 27.5mm. This value is assigned to DELRP(2) since DELRP(1) is used to specify the landward limit of the armor analysis.

- The paper by Kobayashi, Cox, and Wurjanto (1990) examined irregular wave reflection and run-up (IJOB=1) with no armor analysis. In this example, armor stability is analyzed $\Longrightarrow$ ISTAB=1. The following input data are specified: the area coefficient C2=0.9; volume coefficient C3=0.6; drag coefficient CD=0.5; inertia coefficient CM=1.5; $tan\phi$=TANPHI=1.191754 (ϕ=50°); the upper and lower limits for the normalized fluid acceleration, AMAX=1.0 and AMIN=−0.8; which are taken from Kobayashi and Otta (1987). The lift coefficient CL=0.18 is based on the calibration reported by Kobayashi and Greenwald (1988).
  The landward limit of the armor stability computation is determined by the water depth DELRP(1), which is taken to be as large as the median diameter of the gravel $\Longrightarrow$ DELRP(1) = 21mm as reported by Kobayashi, Cox, and Wurjanto (1991) who also indicated the specific density of the gravel SG=2.7.

- The grid spacing $\Delta x'$ between adjacent nodes is taken to be 0.003m, which corresponds to INITS=400. $\Delta x'$=0.003m is sufficiently small for the characteristic wave with H'= 0.0705m and T'=1.069s whose linear wavelength L'=1.63m at the seaward boundary.

The other input parameters for this example are as follows:

- MSTAT=0. The statistical calculations are performed during the entire computation duration, $0< t \leq$TMAX.

- NRATE=50. The computed time series are stored at the rate of 50 data points per wave period.

- IENERG=1. The wave energy quantities are computed and stored in the file OENERG.

- ISAVA=1, MSAVA1=170, MSAVA2=173, NTIMES=4. This means that the spatial variations of η and u are stored at the normalized time t = 169.25, 169.50, 169.75, ..., 173.

- ISAVB=ISAVC=0. The time series at spatial nodes are not stored.

- IREQ=0. No special request is made. Accordingly, IELEV, IV, IDUDT, ISNR, and NREQ are all zero. For IBREAK (Kobayashi and Wurjanto 1989d), the only way to store the spatial variations of the hydrodynamic quantities at the time of the minimum armor stability is through this special request. RBREAK stores these automatically.

- X1 and X2. For this computation, the damping coefficients X1=X2=2.0, which are within the range used by Kobayashi and Wurjanto (1989d).
- The values of DELTA and MULTIF are discussed later in relation with the three different computations made for Run I1.

Three computations are performed for Run I1 using different input values of DELTA and MULTIF to show how to adjust these two input parameters.

• COMPUTATION 1 •

Computation 1 is performed with DELTA=0.001 as input and MULTIF=1. The primary input data file for Computation 1 is presented in Table D-1 and in the file append.inp on the accompanying disk. The contents of the accompanying disk are listed in Appendix H. The file ODOC stored in the file append.doc on the disk is presented in Table D-2. The file BINPUT stored in the file append.bin on the disk is written in Table D-3. These files are from Computation 1.

This example demonstrates how the reversal mechanism described in Section 4.2.4 actually works. There was one reversal in Computation 1. Table D-4 contains a portion of the computation log written in the file BDOC that shows when the reversal occurred.

The computation was in the unit M=166 when the problem IPROB=3 occurred because of the violation of the numerical stability criterion. The computer program BEFORR performed nine unsuccessful trial runs to overcome the problem. When the tenth run was initiated, the numerical stability indicator Ω exceeded the upper limit 20, which dictated the reversal. It was judged to be too close to reverse to t=165 (LEVEL=2), which was within the minimum reversal of two wave periods from t=165.016. Consequently, the reversal was made to t=160 (LEVEL=3). Displayed below is the message issued just before the reversal was made where the message was retrieved from the file BMSG.

```
From Subr. 11 NUMSTA: IPROB=3
(Stability criterion violated)
Computation unit                M =     166
Waterline node                  S =     510
Computation parameters:      NONE =   61700
                            OMEGA =      25.50
Time of occurrence           TIME =     165.016337115
Number of occurrence       ICOUNT =       1
Indicators              ITRY,IREV =   9,   0

From Subr. 10 CPARAM:
Computation unit                   M =     166
Numerical stability indicator > 20:
                               OMEGA =      25.50
Number of failed trials         ITRY =      10
Reverse indicator               IREV =       0
```

```
Last used:                    NONE =   61700
                             DELTA =        0.000185000
Computation reversed to LEVEL=2 or LEVEL=3.
```

In the following, figures are plotted from the output files created by RBREAK. These figures are limited to the computed results which do not require additional analyses such as spectral analyses. Physical and probabilistic interpretations of the computed results using time series and spectral analyses were given in the papers and reports summarized in Section 1.3.

Most of the figures presented in this Appendix are from Computation 1. Each of the figures shown hereafter is explained briefly. The index j is used to indicate the spatial node j located at x=(j-1)Δx.

- Figure D-3 shows the time series of the reflected wave train at the seaward boundary, η_r=ETAR. The data points are retrieved from the file OSEAWAV, column 3.
- Figure D-4 shows the time-varying waterline elevation Z_r=RUNUPS(2) corresponding to the physical water depth δ_r'=27.5mm. The data points are retrieved from the file ORUNUP, column 3.
- Figures D-5, D-6, and D-7 show the spatial variations of the statistical values of η, u, and m, respectively. These data points are provided by the file OSTAT. In each of these three figures, the upper panel [Figure (a)] is from Computation 1, while the lower panel [Figure (b)] is from Computation 2. The difference between the two will be discussed under the heading "COMPUTATION 2".
  - In Figure D-5:
    - $\overline{\eta}$ = ESTAT(1,j)
    - η_{max} = ESTAT(2,j)
    - η_{min} = ESTAT(3,j)
  - Also plotted in Figure D-5 is the bottom slope whose geometry is given by x=XB(j) and z=ZB(j) for the node j on the slope. The nodal locations are stored in the file OSPACE.
  - In Figure D-6:
    - $\overline{u}$ = VSTAT(1,j)
    - u_{max} = VSTAT(2,j)
    - u_{min} = VSTAT(3,j)
  - In Figure D-7:
    - $\overline{m}$ = U1STAT(1,j)

    It is noted that $\overline{m}$=0 to satisfy the condition of no net flux into the impermeable slope as explained by Kobayashi, DeSilva, and Watson (1989). Since $|\overline{m}| < 0.002$ in Figure D-7, the condition of $\overline{m}$=0 is regarded to be satisfied within the numerical accuracy.
- Figure D-8 depicts the spatial variations of the time-averaged energy quantities. The data points are obtained from the file OENERG.

- $\overline{E}$ = ENER(1,j)
- $\overline{E_F}$ = ENER(2,j)
- $\overline{D_f}$ = ENER(3,j)
- $\overline{D_B}$ = ENER(4,j)

The upper panel of Figure F-8 is from Computation 1, while the lower panel is from Computation 2.

- Figures D-9 and D-10 show the spatial variations of the instantaneous hydrodynamic quantities η=ELEV(j) and u=V(j), respectively, at the specified moments for the option ISAVA=1. The data points are from the file OSPACE.
- Figures D-11 and D-12 are related to armor stability.
  - Figure D-11 shows the spatial variations of u=ATMIN(2,j), du/dt= ATMIN(3,j), and N_R= ATMIN(4,j) at the time of the minimum armor stability. The left panel [Figure (a)] is from Computation 1, while the right panel [Figure (b)] is from Computation 2. It is noted that the range of x is limited by DELRP(1)=21mm specified as input.
  - Figure D-12 shows the spatial variation of N_{sx}= SNSX(j), the local stability number. The upper panel [Figure (a)] is from Computation 1, whereas the lower panel [Figure (b)] is from Computation 2. The minimum value of N_{sx} along the slope is the critical stability number N_{sc}. Computation 1 yields N_{sc}=0.479, which occurred at the node number 382 at the time t=79.045 as can be found in Table D-2. Computation 2 produces very close results: N_{sc}=0.475, which occurred at the node number 383 at the time t=79.047. It should be stated that the probabilistic nature of armor stability under irregular wave attack will need to be accounted for in interpreting the computed results as discussed by Kobayashi, Wurjanto, and Cox (1990b).

• COMPUTATION 2 •

Sharp spikes occur in the spatial variations of u_{max} and u_{min} for Computation 1 as shown in Figure D-6a. A similar spike also occurs in the spatial variation of D_f shown in Figure D-8a. These spikes are due to the use of the smallest possible NONE for each computation unit in Computation 1 where MULTIF=1 is used in Eq. 86.

The purpose of executing Computation 2 is to show these spikes can be reduced or eliminated by the use of a larger value of MULTIF. Computation 2 uses MULTIF=3. The spatial variations of u_{max}, u_{min}, and D_f for Computation 2 exhibit no spike as can be seen in Figures D-6b and D-8b. No apparent differences between Computations 1 and 2 exist in Figures D-5, D-7, D-11, and D-12 despite the fact that Figures D-6a and D-6b are sharply different. The unrealistically large water particle velocities may occur at some nodes during the computation duration but the time-averaged quantities are barely affected by them. If a user is interested only in the gross or time-averaged quantities, then MULTIF=1 may be sufficient. On the other hand, if sharp spikes occur in the spatial variations of u_{max} and u_{min}, a larger value of MULTIF can be used to reduce or eliminate the spikes, although computation time increases with the increase of MULTIF.

The value of DELTA specified as input for Computation 2 is 0.001 and the same as that for Computation 1. Appendix F will suggest that it may be necessary to increase the value of DELTA to overcome numerical difficulties when the value of MULTIF is increased.

• COMPUTATION 3 •

Computation 3 is performed to show that the use of too small a value of DELTA may lead to a failure of the BEFORR computation. DELTA=0.0002 is used in Computation 3, while MULTIF=1 as in Computation 1. The computation fails within the computation unit 2 after performing five reversals. The last 23 lines of messages written in the file BMSG are shown below.

```
From Subr. 11 NUMSTA: IPROB=3
(Stability criterion violated)
Computation unit              M =       2
Waterline node                S =     402
Computation parameters:    NONE =   49550
                          OMEGA =      20.48
Time of occurrence         TIME =       1.803753784
Number of occurrence     ICOUNT =       1
Indicators            ITRY,IREV =   4,   5

From Subr. 10 CPARAM:
Computation unit                M =       2
Numerical stability indicator > 20:
                           OMEGA =      20.48
Number of failed trials     ITRY =       5
Reverse indicator           IREV =       5
Last used:                  NONE =   49550
                           DELTA =       0.000200000
Computation reversed to LEVEL=2 or LEVEL=3.

Computation aborted after IREV reached 5.
Suggested to input larger DELTA.
Programmed Stop.
```

Table D-1: Primary input data file for BEFORR-RBREAK computation of "Kobayashi, Cox, and Wurjanto 1990 Run I1".

```
 1          3                          --> NLINES
 2    ------------------------------------------------------------------------
 3    Kobayashi, Cox, and Wurjanto 1990 Run I1
 4    ------------------------------------------------------------------------
 5    11                              --> IJOB,ISTAB
 6    1                               --> ISYST
 7    1                               --> IBOT
 8    0                               --> INONCT
 9    1                               --> IENERG
10    2       0  tkcwi1                --> IWAVE,MSTAT,FINP2
11    1     170     173       4       --> ISAVA,MSAVA1,MSAVA2,NTIMES
12    0       0                       --> ISAVB,NNODB
13    0       0                       --> ISAVC,NNODC
14    00000       0                   --> IREQ,IELEV,IV,IDUDT,ISNR,NNREQ
15         400                        --> INITS
16           1                        --> MULTIF
17            .001000                 --> DELTA (normalized)
18         173.058934                 --> TMAX  (normalized)
19          50                        --> NRATE
20           2                        --> NDELR
21        21.000000                   --> DELRP(1)(millimeters)
22        27.500000                   -->       (2)
23         2.000000     2.000000      --> X1,X2
24          .070500     1.069000      --> HREFP(meters),TP(seconds)
25         1.000000     1.000000      --> KSREF,KSSEA
26          .050000                   --> FWP
27          .400000                   --> DSEAP(meters)
28          .333333                   --> TSLOPS
29           1                        --> NBSEG
30         2.100000      .333333      --> WBSEG(1)(meters),TBSLOP(1)
31          .900000      .600000        2.700000  --> C2,C3,SG
32          .500000      .180000        1.500000  --> CD,CL,CM
33         1.191754                                --> TANPHI
34         1.000000     -.800000                   --> AMAX,AMIN
```

Table D-2: File ODOC produced by RBREAK computation of "Kobayashi, Cox, and Wurjanto 1990 Run I1".

```
 1  ------------------------------------------------------------------------
 2  Kobayashi, Cox, and Wurjanto 1990 Run I1
 3  ------------------------------------------------------------------------
 4
 5  WAVE CONDITION
 6
 7  Incident Wave at Seaward Boundary Given as Input
 8  Number of Data Points     NDATA =     5921
 9  Norm. Maximum Surface Elev.     =        0.853900
10  Norm. Minimum Surface Elev.     =       -0.859800
11
12  Reference Wave Period           =        1.069000 sec.
13  Reference Wave Height           =        0.070500 meters
14  Depth at Seaward Boundary       =        0.400000 meters
15  Shoal. Coef. at Reference  Ks1  =        1.000
16               at Seaw. Bdr. Ks2  =        1.000
17               Ks = Ks2/Ks1       =        1.000
18  Norm. Depth at Seaw. Boundary   =        5.674
19  Normalized Wave Length          =        4.071
20  "Sigma"                         =       12.610
21  Ursell Number                   =        2.921
22  Surf Similarity Parameter       =        1.677
23
24  SLOPE PROPERTIES
25
26  Friction Factor             =         0.050000
27  Norm. Friction Factor       =         0.315252
28  Norm. Horiz. Length of
29        Computation Domain    =         2.358805
30  Number of Segments          =         1
31  --------------------------------
32   SEGMENT     WBSEG(I)    TBSLOP(I)
33         I       meters
34  --------------------------------
35         1    2.100000     0.333333
36  --------------------------------
37
38  PARAMETERS FOR ARMOR STABILITY AND MOVEMENT
39
40  Armor Friction Factor       =     1.192
41  Specific Gravity            =     2.700
42  Area Coefficient       C2   =     0.900
43  Volume Coefficient     C3   =     0.600
44  Drag Coefficient       CD   =     0.500
45  Lift Coefficient       CL   =     0.180
46  Inertia Coefficient    CM   =     1.500
47  Norm. Upper and Lower Bounds of du/dt
48                         AMAX =     1.000
49                         AMIN =    -0.800
50
```

Table D-2 continued

```
 51     COMPUTATION PARAMETERS
 52
 53     Total Number of Spatial Nodes      JE =      700
 54     Number of Nodes Along Bottom Below SWL
 55                                      INITS =      400
 56     Normalized Delta x          =        0.337454D-02
 57     Damping Coefficients    x1 =           2.000
 58                             x2 =           2.000
 59     Computation Duration  TMAX =         173.058934 Wave Periods
 60     Minimum Allowable NONE           NONEM =     2450
 61     Time Series Stored at Rate NRATE
 62          per Wave Period with        NRATE =       50
 63     Statistical Calculations Are Performed Excluding
 64          the First MSTAT Computation Units
 65          with                        MSTAT =        0
 66     Computation of Armor Stability or Movement is Performed
 67          Excluding the First MSTAB Computation Units
 68          with                        MSTAB =        0
 69     Spatial Variations are Stored NTIMES at Equal
 70          Intervals per Wave Period from
 71          Computation Unit MSAVA1 to MSAVA2,
 72          Inclusive, with            NTIMES =        4
 73                                     MSAVA1 =      170
 74                                     MSAVA2 =      173
 75
 76     REFLECTION COEFFICIENT
 77
 78     r1 =     0.344
 79
 80     RUNUP, RUNDOWN, SETUP
 81
 82     Largest Node Number Reached by Computational Waterline
 83                                                 JMAX =      575
 84     ------------------------------------------------------------
 85           I   DELTAR(I)     RUNUP(I)    RUNDOWN(I)    SETUP(I)
 86                 [mm]          R            Rd            Zr
 87     ------------------------------------------------------------
 88           1    21.000        1.838       -1.082        0.210
 89           2    27.500        1.583       -1.028        0.128
 90     ------------------------------------------------------------
 91
 92     WAVE SET-DOWN OR SETUP
 93
 94     Average value of ETAI =    -0.000002
 95                       ETAR =     0.007371
 96
```

Table D-2 continued

```
 97     TIME-AVERAGED ENERGY BALANCE
 98
 99     Normalized Energy Flux:
100     . at Seaw. Boundary    A =  0.145671D+00
101     . at Landw. Boundary   B =  0.000000D+00
102     Normalized Rate of Energy Dissipation
103     in the Computation Domain, Due to:
104     . Bottom Friction      C =  0.396148D-01
105     . Wave Breaking        D =  0.104748D+00
106     Additional Term        E =  0.130852D-02
107     Calculation 1:
108                      F = A-B =  0.145671D+00
109                  G = C+D+E =  0.145671D+00
110       Must H=0, but H = G-F = -0.147330D-06
111       % error  100G/E =          0.00
112
113     STABILITY NUMBER
114
115     Largest Node Number for Which Armor Stability
116       Computation Performed  JSTABM =          509
117     Critical Stability Number   Nsc =        0.479
118     At Node Number                 =          382
119     At Time                        =           79.045205
```

Table D-3: File BINPUT produced by BEFORR computation of "Kobayashi, Cox, and Wurjanto 1990 Run I1".

```
1           3
2     ------------------------------------------------------------------------
3     Kobayashi, Cox, and Wurjanto 1990 Run I1
4     ------------------------------------------------------------------------
5          174
6     tkcwi1
7         5921
8         2450
9            1    2450    2450   0.100000000D-02
10           2    2450    2450   0.100000000D-02
11           3    2450    2450   0.100000000D-02
12           4    2450    2450   0.100000000D-02
13           5    2450    2450   0.100000000D-02
14           6    2450    2450   0.100000000D-02
15           7    2450    2450   0.100000000D-02
16           8    2450    2450   0.100000000D-02
17           9    2450    2450   0.100000000D-02
18          10    2450    2450   0.100000000D-02
19          11    2450    2450   0.100000000D-02
20          12    2450    2450   0.100000000D-02
21          13    2450    2450   0.100000000D-02
22          14    2450    2450   0.100000000D-02
23          15    2450    2450   0.100000000D-02
24          16    2450    2450   0.100000000D-02
25          17    2450    2450   0.100000000D-02
26          18    2450    2450   0.100000000D-02
27          19    2450    2450   0.100000000D-02
28          20    2450    2450   0.100000000D-02
29          21    2450    2450   0.100000000D-02
30          22    2450    2450   0.100000000D-02
31          23    2450    2450   0.100000000D-02
32          24    2450    2450   0.100000000D-02
33          25    2450    2450   0.100000000D-02
34          26    2450    2450   0.100000000D-02
35          27    2450    2450   0.100000000D-02
36          28    2450    2450   0.100000000D-02
37          29    2450    2450   0.100000000D-02
38          30    2450    2450   0.100000000D-02
39          31    2450    2450   0.100000000D-02
40          32    2450    2450   0.100000000D-02
41          33    2450    2450   0.100000000D-02
42          34    2450    2450   0.100000000D-02
43          35    2450    2450   0.802000000D-03
44          36    2450    2450   0.802000000D-03
45          37    2450    2450   0.802000000D-03
46          38    2450    2450   0.669000000D-03
47          39    2450    2450   0.669000000D-03
48          40    2450    2450   0.669000000D-03
49          41    2450    2450   0.669000000D-03
50          42    2450    2450   0.669000000D-03
51          43    2450    2450   0.669000000D-03
52          44    2450    2450   0.669000000D-03
53          45    2450    2450   0.669000000D-03
54          46    2450    2450   0.669000000D-03
55          47    2450    2450   0.669000000D-03
56          48    2450    2450   0.630000000D-03
```

Table D-3 continued

57	49	2450	2450	0.630000000D-03
58	50	2450	2450	0.630000000D-03
59	51	2450	2450	0.630000000D-03
60	52	2450	2450	0.630000000D-03
61	53	2450	2450	0.630000000D-03
62	54	2450	2450	0.518000000D-03
63	55	2450	2450	0.518000000D-03
64	56	2450	2450	0.518000000D-03
65	57	2450	2450	0.518000000D-03
66	58	2450	2450	0.518000000D-03
67	59	2450	2450	0.518000000D-03
68	60	2450	2450	0.518000000D-03
69	61	2450	2450	0.518000000D-03
70	62	2450	2450	0.518000000D-03
71	63	2450	2450	0.518000000D-03
72	64	2450	2450	0.518000000D-03
73	65	2450	2450	0.518000000D-03
74	66	2450	2450	0.518000000D-03
75	67	2450	2450	0.518000000D-03
76	68	2450	2450	0.518000000D-03
77	69	2450	2450	0.518000000D-03
78	70	2450	2450	0.518000000D-03
79	71	2450	2450	0.518000000D-03
80	72	2450	2450	0.518000000D-03
81	73	2450	2450	0.518000000D-03
82	74	2450	2450	0.518000000D-03
83	75	2450	2450	0.294000000D-03
84	76	2450	2450	0.294000000D-03
85	77	5450	5450	0.276000000D-03
86	78	3650	3650	0.276000000D-03
87	79	2450	2450	0.276000000D-03
88	80	3650	3650	0.276000000D-03
89	81	2450	2450	0.276000000D-03
90	82	2450	2450	0.276000000D-03
91	83	2450	2450	0.276000000D-03
92	84	3650	3650	0.276000000D-03
93	85	2450	2450	0.276000000D-03
94	86	2450	2450	0.276000000D-03
95	87	2450	2450	0.276000000D-03
96	88	3650	3650	0.276000000D-03
97	89	2450	2450	0.276000000D-03
98	90	2450	2450	0.276000000D-03
99	91	2450	2450	0.276000000D-03
100	92	3650	3650	0.255000000D-03
101	93	2450	2450	0.255000000D-03
102	94	2450	2450	0.255000000D-03
103	95	2450	2450	0.255000000D-03
104	96	2450	2450	0.255000000D-03
105	97	3650	3650	0.255000000D-03
106	98	3650	3650	0.255000000D-03
107	99	2450	2450	0.255000000D-03
108	100	3650	3650	0.241000000D-03
109	101	3650	3650	0.241000000D-03
110	102	5450	5450	0.241000000D-03
111	103	2450	2450	0.241000000D-03
112	104	2450	2450	0.241000000D-03

Table D-3 continued

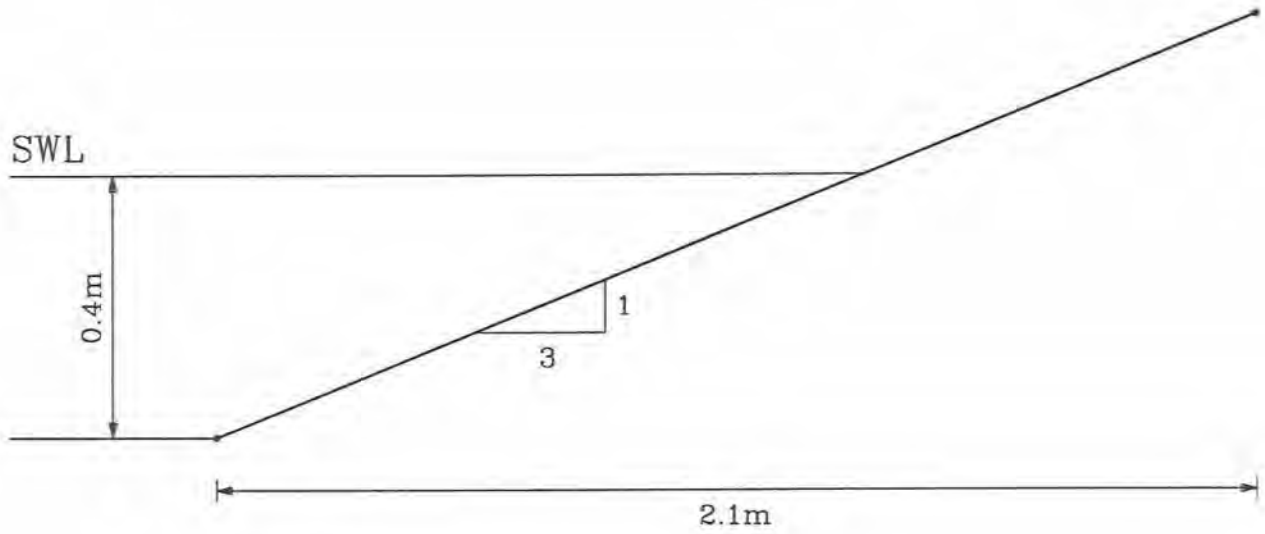
113	105	5450	5450	0.241000000D-03
114	106	3650	3650	0.241000000D-03
115	107	2450	2450	0.241000000D-03
116	108	2450	2450	0.241000000D-03
117	109	2450	2450	0.241000000D-03
118	110	8150	8150	0.241000000D-03
119	111	3650	3650	0.241000000D-03
120	112	2450	2450	0.241000000D-03
121	113	2450	2450	0.241000000D-03
122	114	2450	2450	0.241000000D-03
123	115	2450	2450	0.241000000D-03
124	116	3650	3650	0.241000000D-03
125	117	2450	2450	0.241000000D-03
126	118	3650	3650	0.241000000D-03
127	119	2450	2450	0.241000000D-03
128	120	2450	2450	0.241000000D-03
129	121	2450	2450	0.241000000D-03
130	122	2450	2450	0.241000000D-03
131	123	5450	5450	0.241000000D-03
132	124	2450	2450	0.241000000D-03
133	125	2450	2450	0.241000000D-03
134	126	2450	2450	0.217000000D-03
135	127	3650	3650	0.206000000D-03
136	128	3650	3650	0.206000000D-03
137	129	2450	2450	0.206000000D-03
138	130	3650	3650	0.206000000D-03
139	131	2450	2450	0.206000000D-03
140	132	2450	2450	0.196000000D-03
141	133	5450	5450	0.196000000D-03
142	134	8150	8150	0.196000000D-03
143	135	5450	5450	0.196000000D-03
144	136	41150	41150	0.196000000D-03
145	137	3650	3650	0.196000000D-03
146	138	2450	2450	0.196000000D-03
147	139	12200	12200	0.196000000D-03
148	140	2450	2450	0.196000000D-03
149	141	8150	8150	0.196000000D-03
150	142	2450	2450	0.196000000D-03
151	143	2450	2450	0.196000000D-03
152	144	3650	3650	0.196000000D-03
153	145	8150	8150	0.196000000D-03
154	146	3650	3650	0.196000000D-03
155	147	3650	3650	0.196000000D-03
156	148	2450	2450	0.196000000D-03
157	149	5450	5450	0.196000000D-03
158	150	2450	2450	0.196000000D-03
159	151	3650	3650	0.196000000D-03
160	152	8150	8150	0.196000000D-03
161	153	2450	2450	0.196000000D-03
162	154	2450	2450	0.196000000D-03
163	155	3650	3650	0.196000000D-03
164	156	2450	2450	0.196000000D-03
165	157	2450	2450	0.185000000D-03
166	158	2450	2450	0.185000000D-03
167	159	3650	3650	0.185000000D-03
168	160	2450	2450	0.185000000D-03

Table D-3 continued

169	161	4900	4900	0.185000000D-03
170	162	4900	4900	0.185000000D-03
171	163	4900	4900	0.185000000D-03
172	164	4900	4900	0.175000000D-03
173	165	4900	4900	0.175000000D-03
174	166	18300	18300	0.175000000D-03
175	167	3650	3650	0.175000000D-03
176	168	3650	3650	0.175000000D-03
177	169	2450	2450	0.175000000D-03
178	170	2450	2450	0.175000000D-03
179	171	5450	5450	0.175000000D-03
180	172	2450	2450	0.175000000D-03
181	173	5450	5450	0.175000000D-03
182	174	2450	144	0.175000000D-03

Table D-4: A portion of file BDOC produced by BEFORR computation of "Kobayashi, Cox, and Wurjanto 1990 Run I1".

```
  1  ------------------------------------------------------------------------
  2  Kobayashi, Cox, and Wurjanto 1990 Run I1
  3  ------------------------------------------------------------------------
  4
  .
  .
  .
 86
 87  COMPUTATION LOG
 88
 89  ------------------------------------------------------------------------
 90   Comp.    NONE     NEND      NEND/        DELTA      Num.      No.     Re-
 91   Unit                         NONE                  Stab.      of    verse
 92                                                      Indic.  Trials  Indic.
 93  ------------------------------------------------------------------------
 94      1     2450     2450       1.0   0.00100000     1.01       1       0
 95      2     2450     2450       1.0   0.00100000     1.01       1       0
 96      3     2450     2450       1.0   0.00100000     1.01       1       0
  .      .
  .      .
  .      .
253    160     2450     2450       1.0   0.00018500     1.01       1       0
254    161     3650     3650       1.0   0.00018500     1.51       2       0
255    162     2450     2450       1.0   0.00018500     1.01       1       0
256    163     5450     5450       1.0   0.00018500     2.25       3       0
257    164     5450     5450       1.0   0.00018500     2.25       3       0
258    165     3650     3650       1.0   0.00018500     1.51       2       0
259    161     4900     4900       1.0   0.00018500     2.03       1       1
260    162     4900     4900       1.0   0.00018500     2.03       1       1
261    163     4900     4900       1.0   0.00018500     2.03       1       1
262    164     4900     4900       1.0   0.00017500     2.03       2       1
263    165     4900     4900       1.0   0.00017500     2.03       1       1
264    166    18300    18300       1.0   0.00017500     7.56       6       0
265    167     3650     3650       1.0   0.00017500     1.51       2       0
266    168     3650     3650       1.0   0.00017500     1.51       2       0
267    169     2450     2450       1.0   0.00017500     1.01       1       0
268    170     2450     2450       1.0   0.00017500     1.01       1       0
  .      .
  .      .
  .      .
```



Kobayashi, Cox, and Wurjanto (1990) Runs I1 and I3

Figure D–1

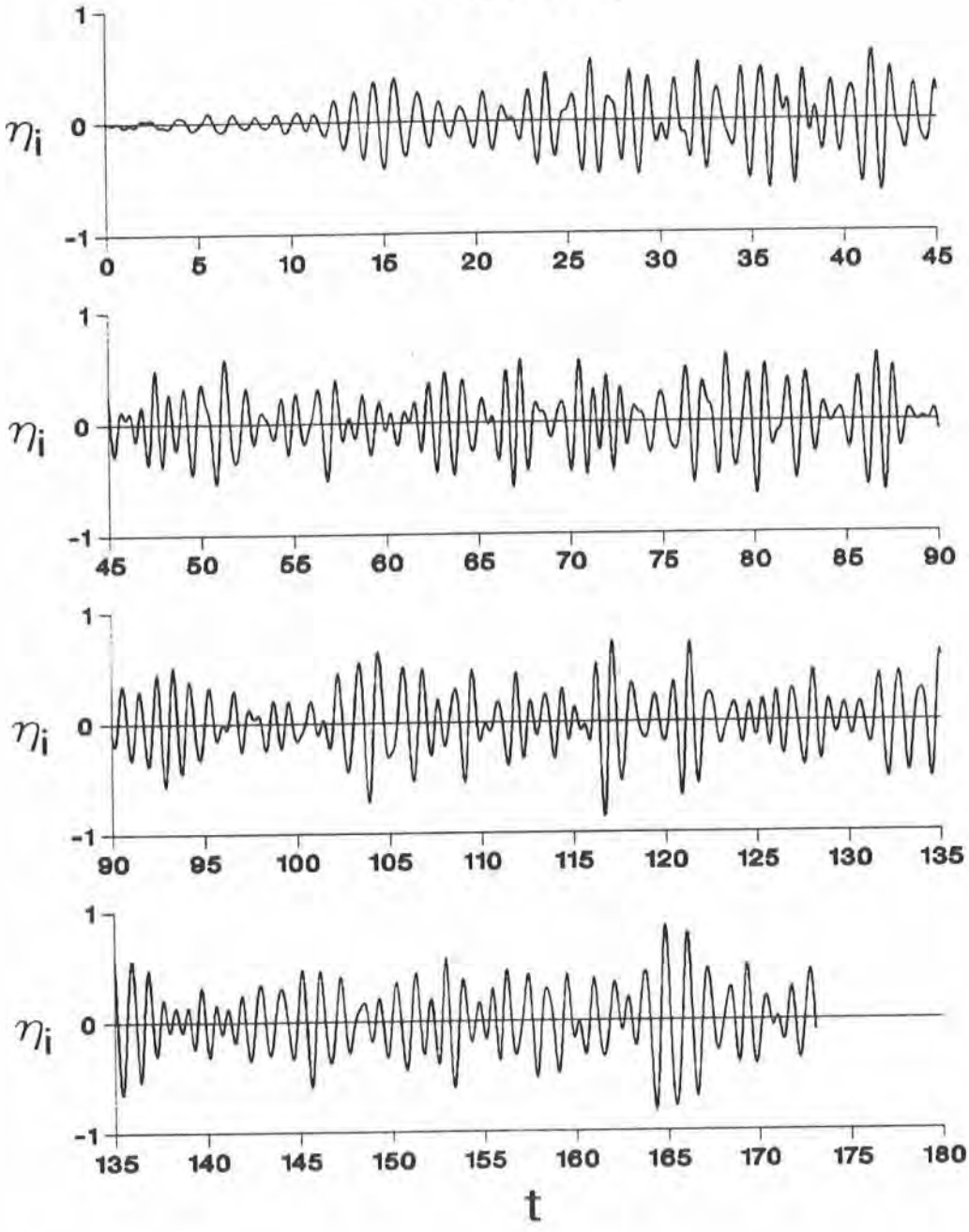

Run I1
η_i
1
0
-1
0 5 10 15 20 25 30 35 40 45
45 50 55 60 65 70 75 80 85 90
90 95 100 105 110 115 120 125 130 135
135 140 145 150 155 160 165 170 175 180
t

Figure D-2

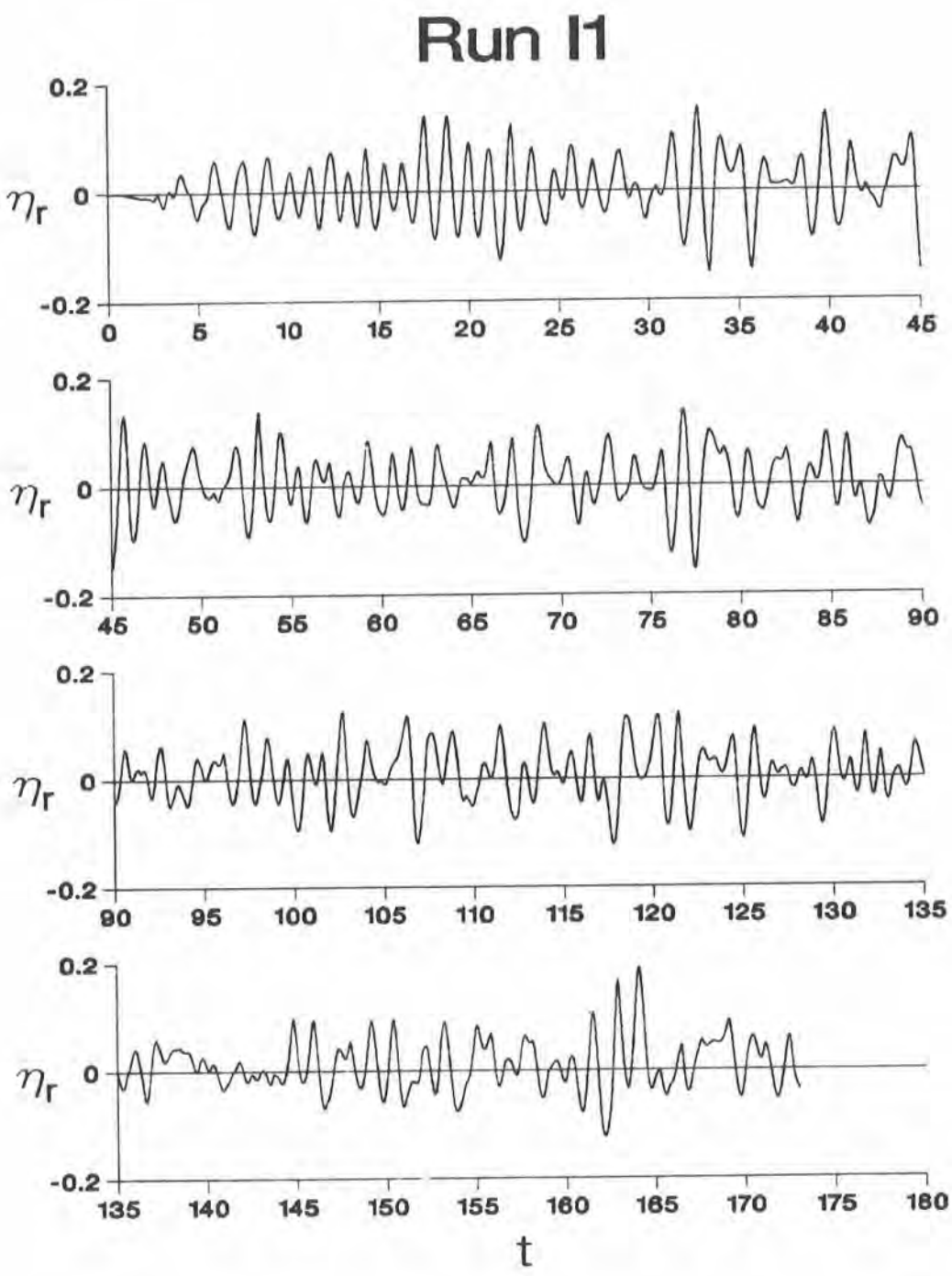

Run I1
0.2
0
-0.2
η_r
0 5 10 15 20 25 30 35 40 45
45 50 55 60 65 70 75 80 85 90
90 95 100 105 110 115 120 125 130 135
135 140 145 150 155 160 165 170 175 180
t

Figure D-3

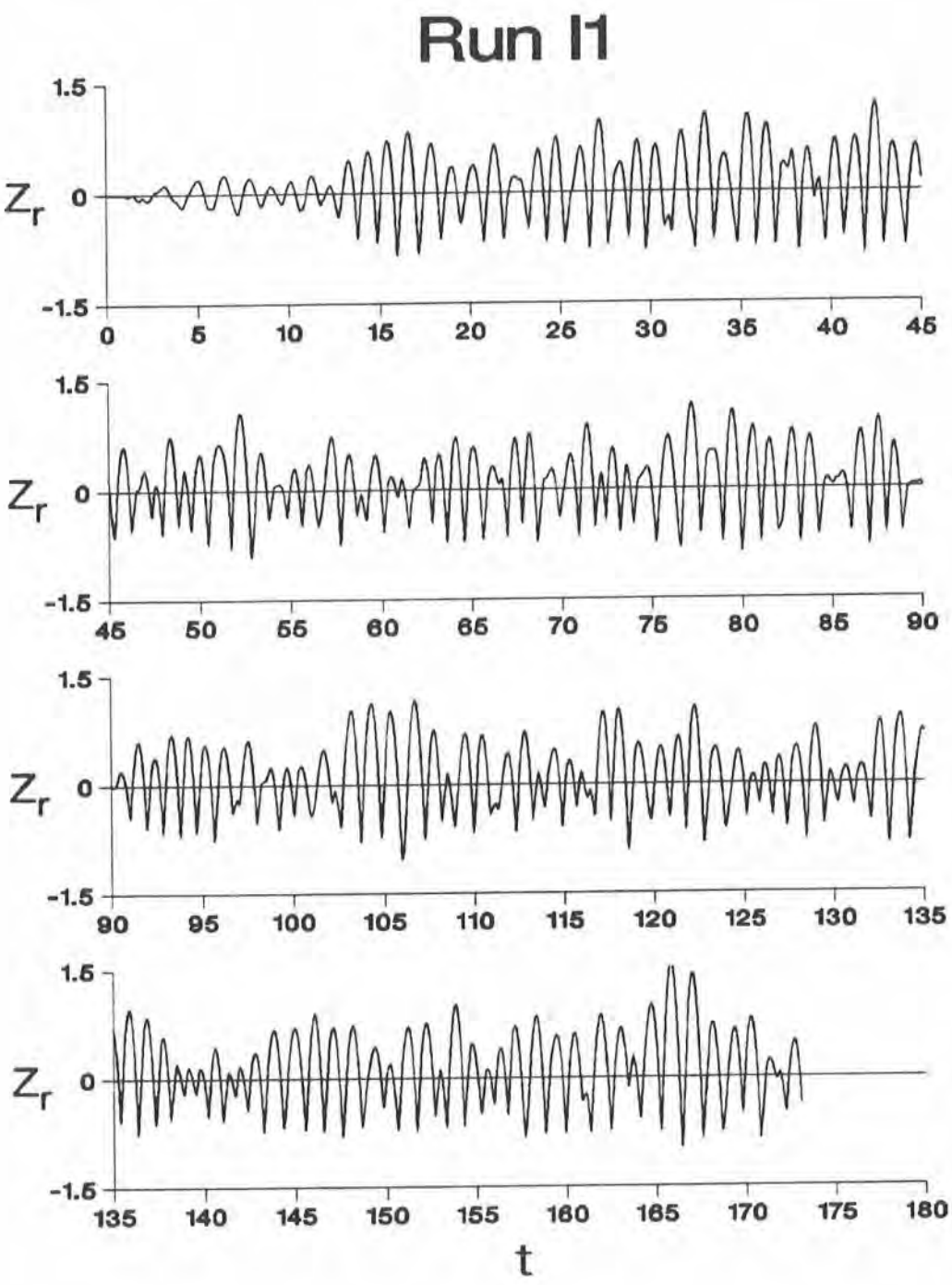

Run I1
1.5
Z_r
0
-1.5
0 5 10 15 20 25 30 35 40 45
45 50 55 60 65 70 75 80 85 90
90 95 100 105 110 115 120 125 130 135
135 140 145 150 155 160 165 170 175 180
t

Figure D-4

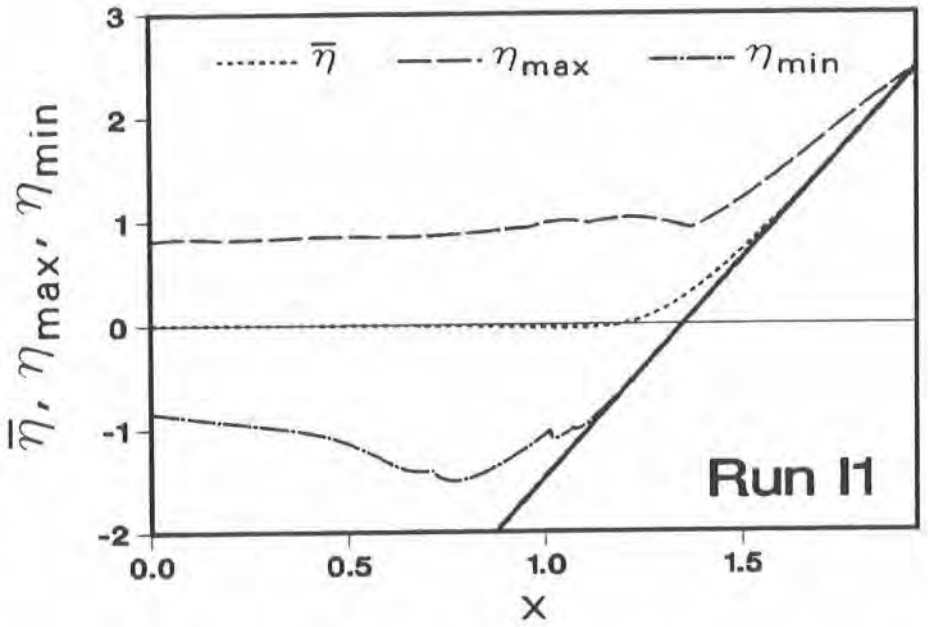


Figure D-5a

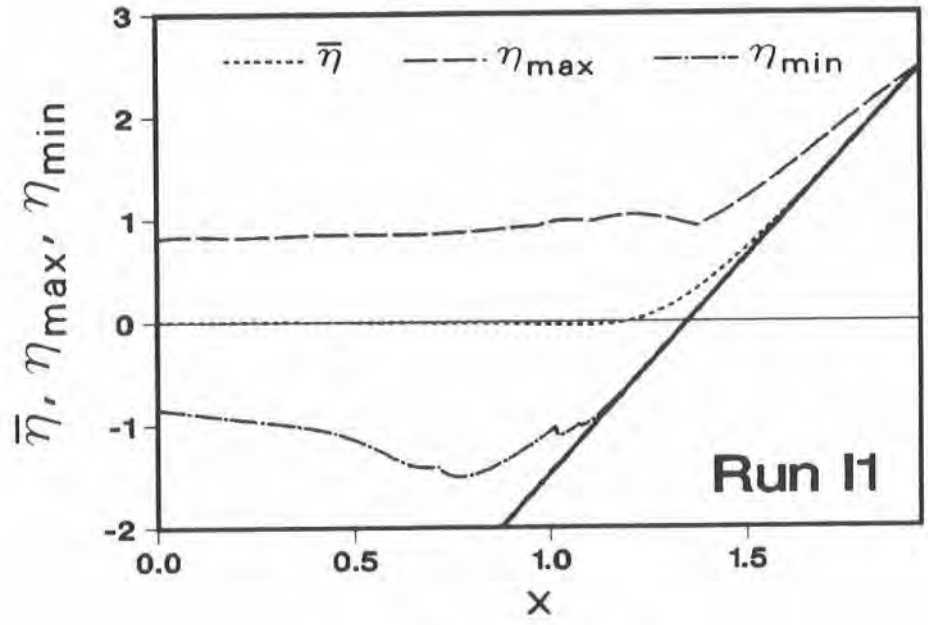


Figure D-5b

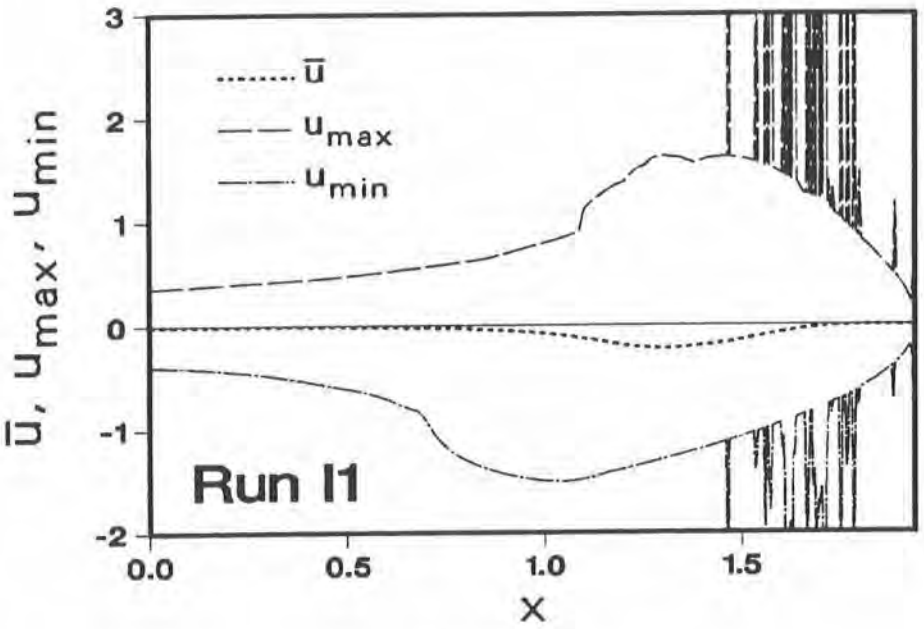


Figure D-6a

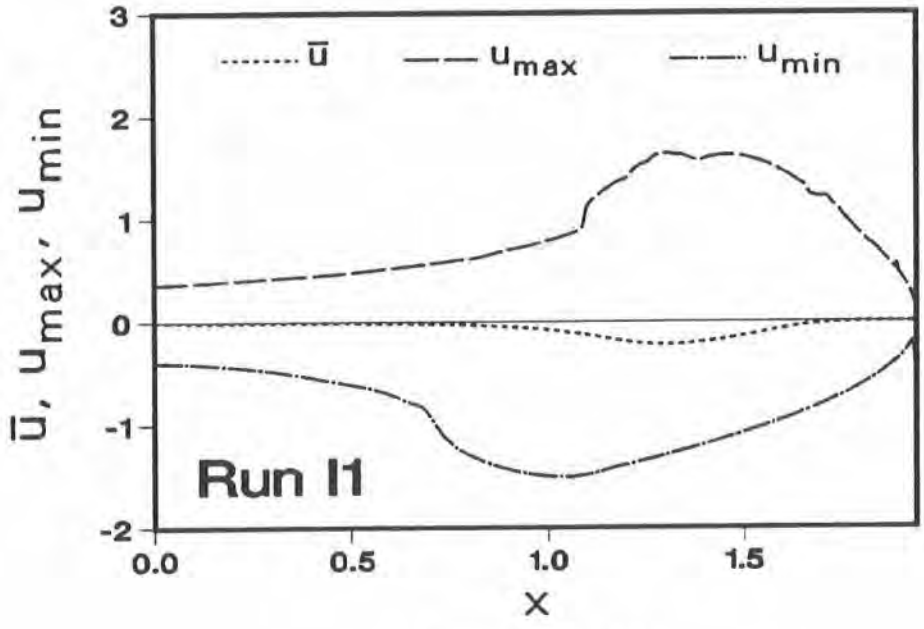


Figure D-6b

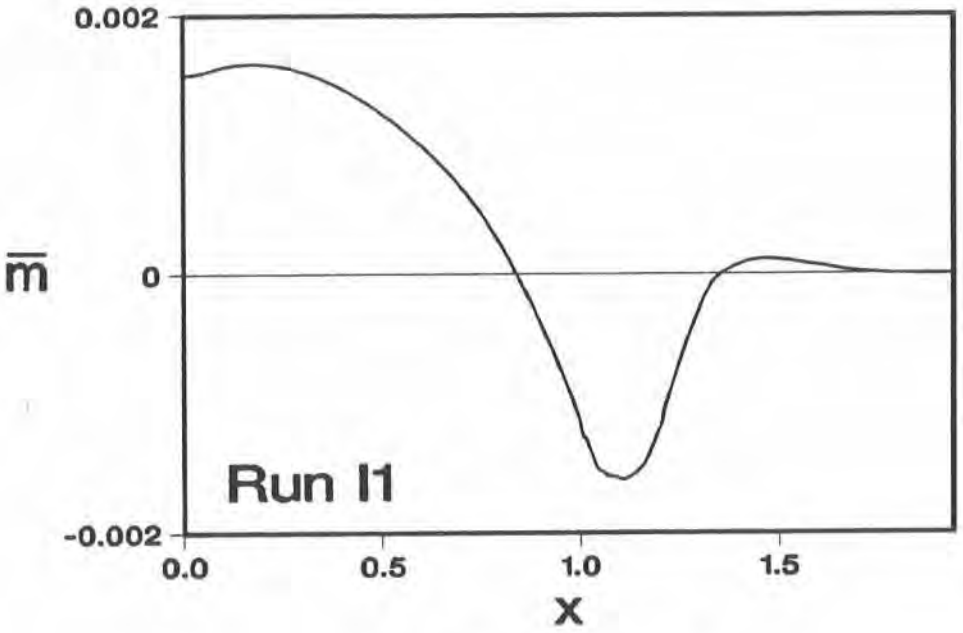


Figure D-7a

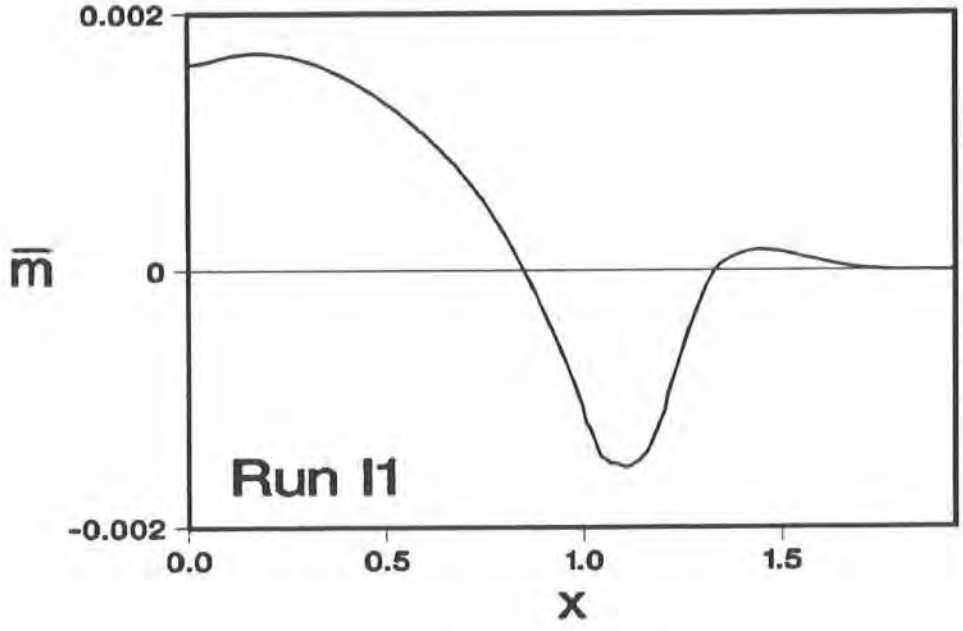


Figure D-7b

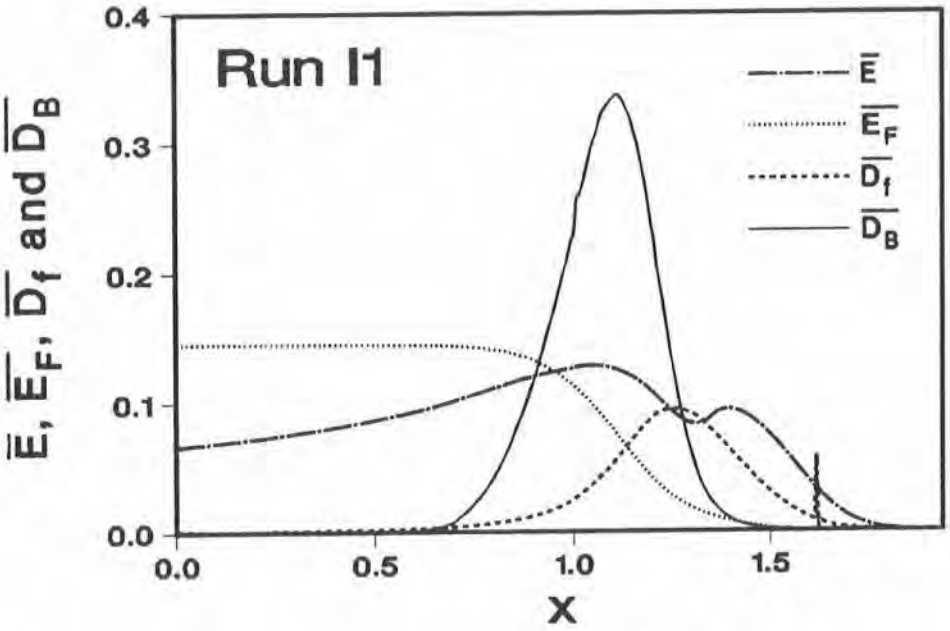


Figure D-8a

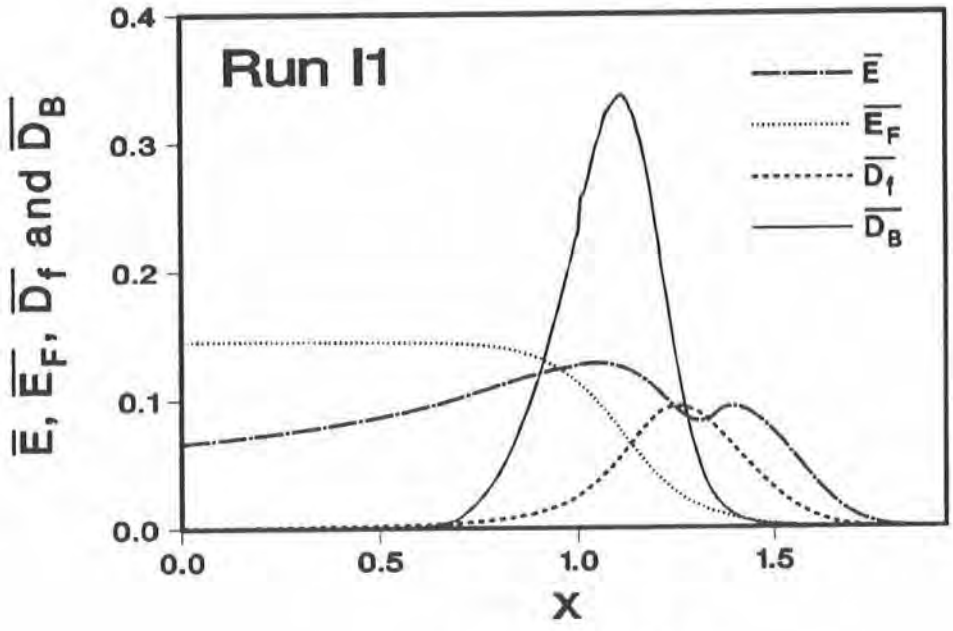


Figure D-8b

Run I1

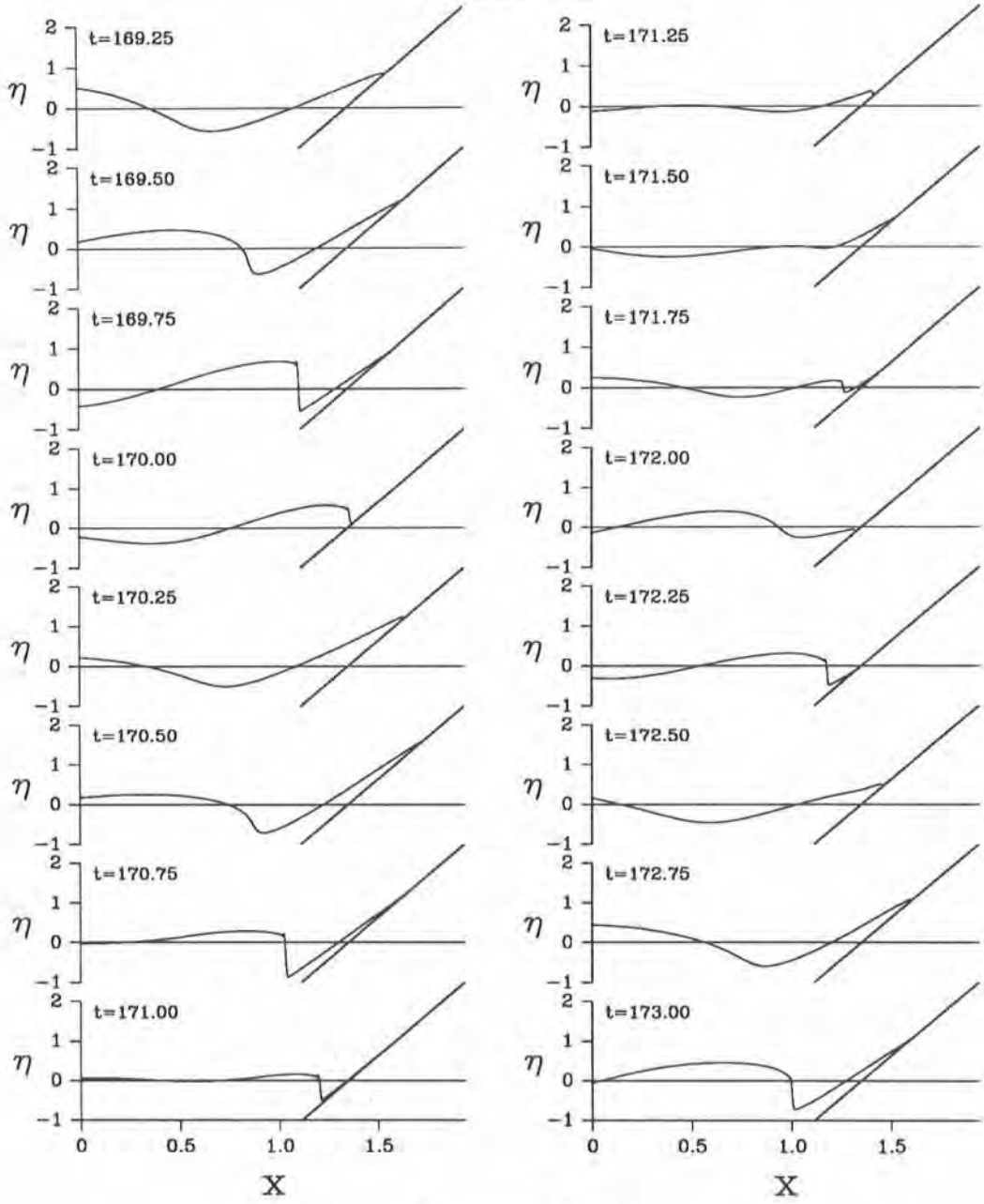


Figure D-9

Run I1

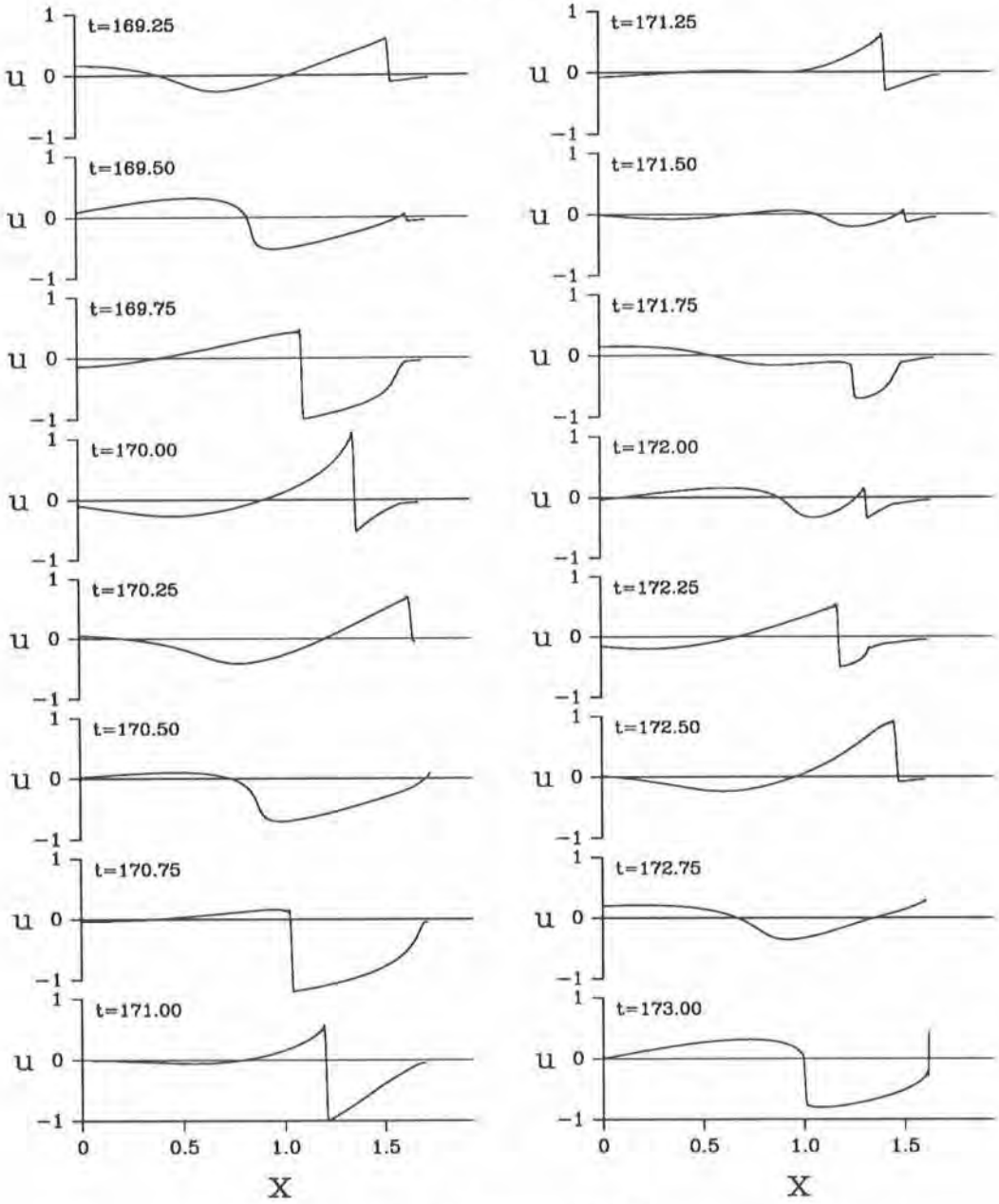


Figure D-10

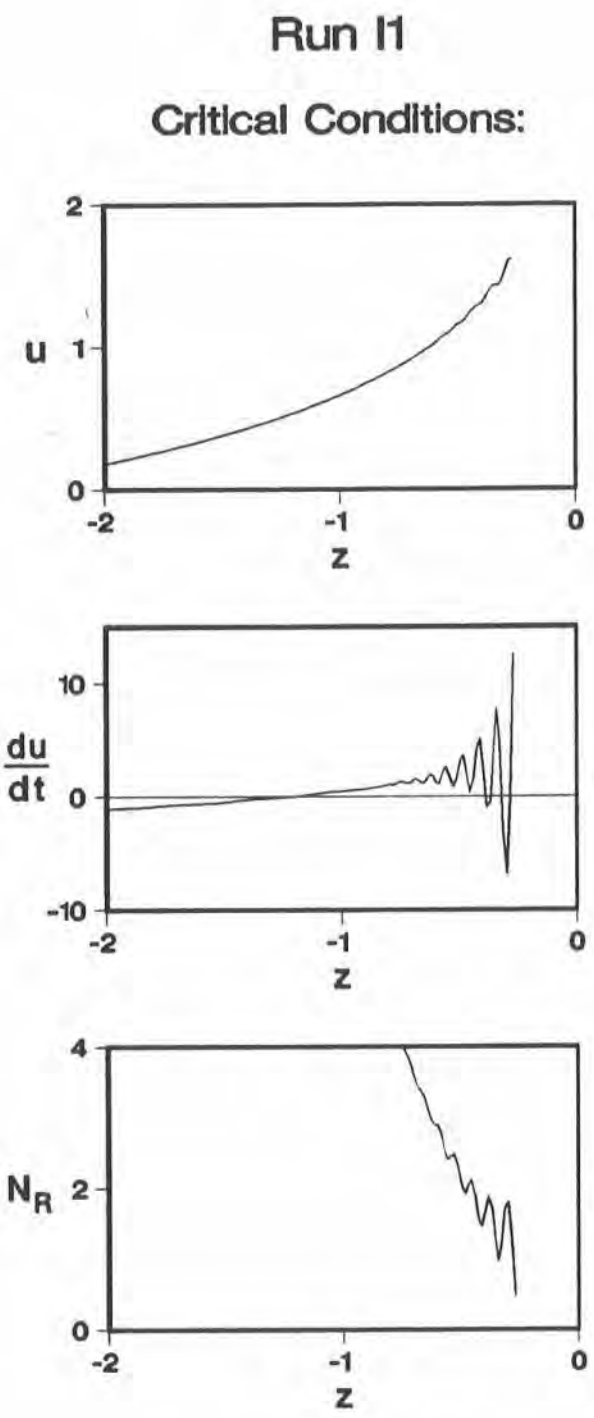


Figure D-11a

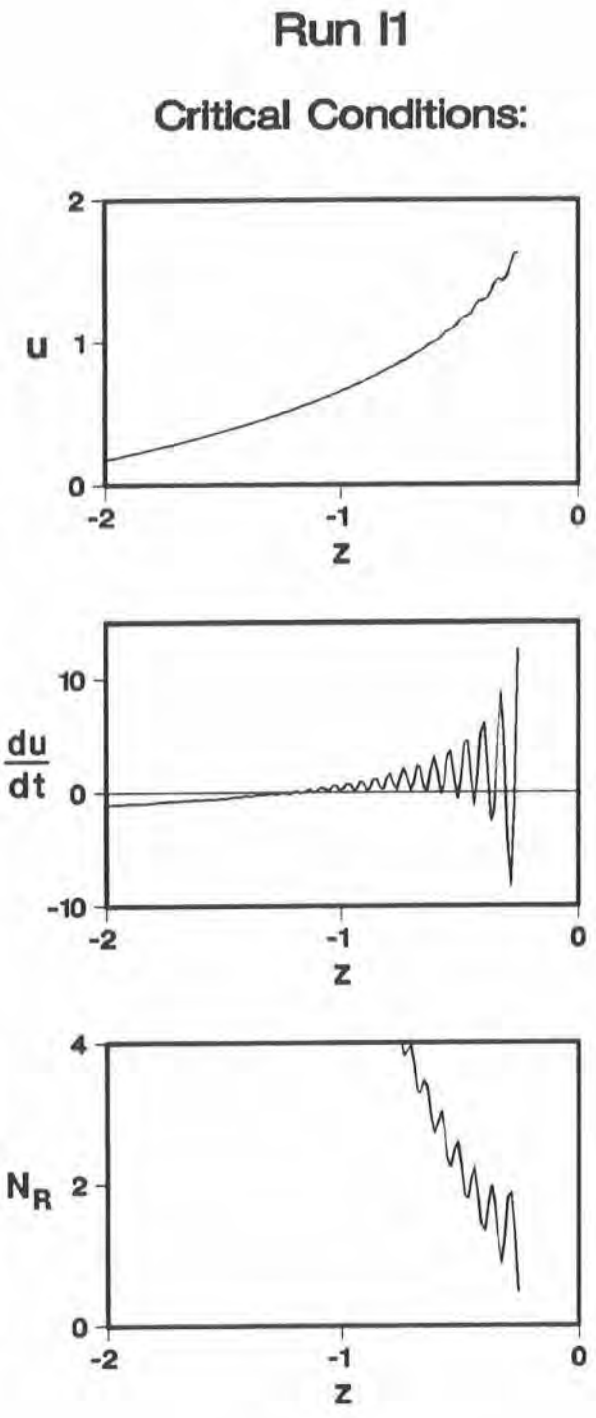


Figure D-11b

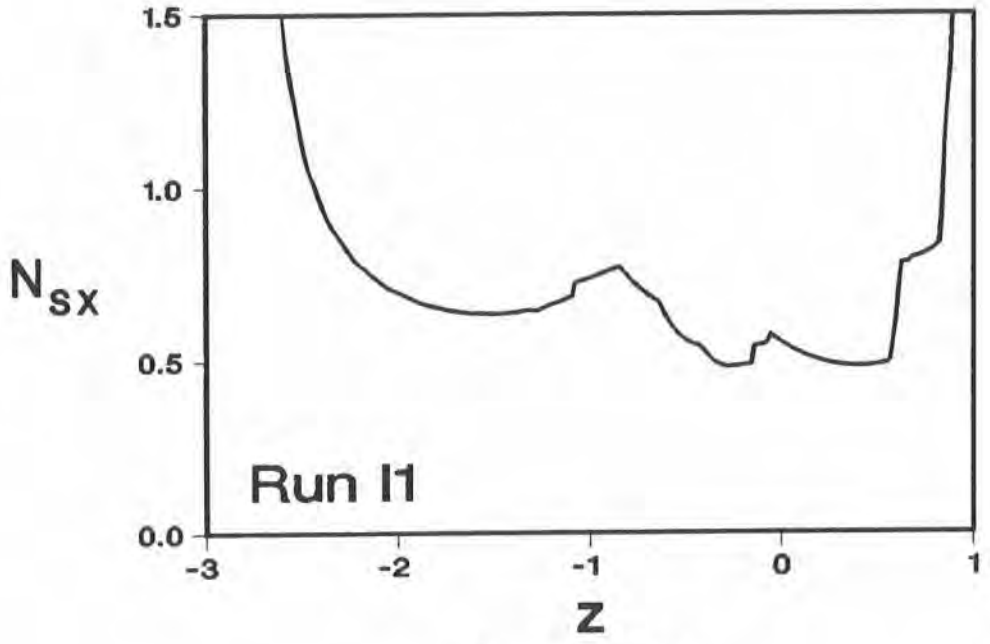


Figure D-12a

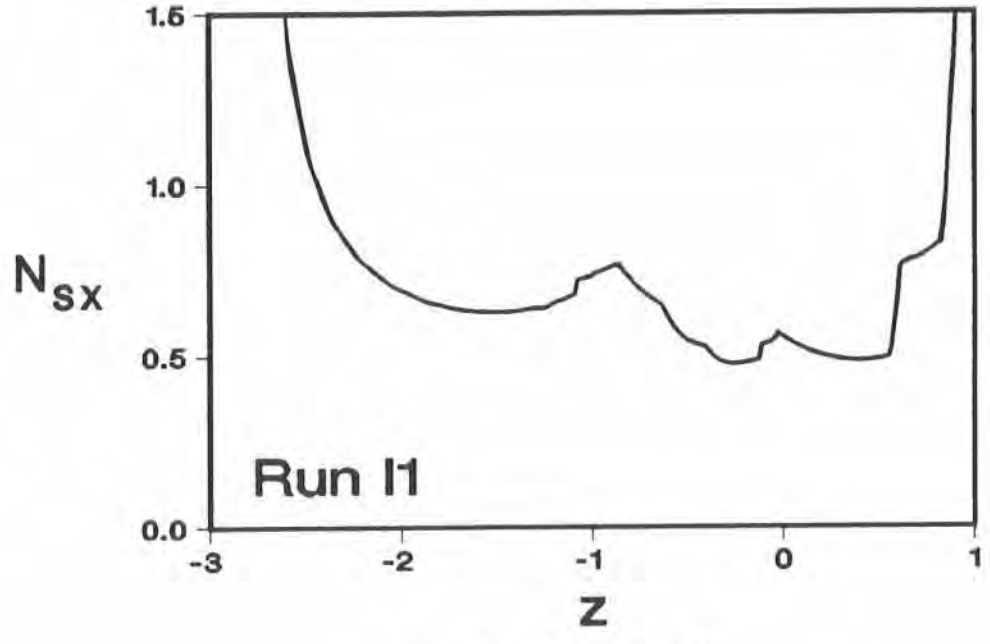


Figure D-12b

APPENDIX E

EXAMPLE: INPUT WAVE TRAIN II FROM EXPERIMENT

This example is Run I3 of Kobayashi, Cox, and Wurjanto (1990), which was similar to the example of Run I1 explained in Appendix D. The slope geometry was the same as that shown in Figure D-1. In fact, only the input wave train and related parameters were different. In the following, this example of Run I3 is explained assuming the familiarity with the example of Run I1 in Appendix D. This example is presented because the computation for Run I3 has been the most difficult among those performed by the authors before the self-adjusting mechanism described in Section 4.2 has been established. This example thus demonstrates that the self-adjusting mechanism does work even for numerically difficult cases.

- Slope roughness. Following the calibration reported by Kobayashi and Greenwald (1988), the bottom friction factor `FWP` is taken to be 0.1 for the surf similarity parameter ξ=3.15.
- Input wave train. The duration of the input wave train for Run I3 was 365 seconds. The train is plotted in Figure E-1 and is stored in the file `appene.iwt` on the accompanying disk.
  - The name of the file containing the input wave train is `FINP2=tkcwi3` for this IBM 3090 run.
  - `HREFP`=0.0508m is the significant wave height of the input wave train, whereas `TP`= 1.706s [reported as 1.71s in Kobayashi, Cox, and Wurjanto (1990)] is the mean period.
  - The normalized computation duration `TMAX` = 365s/1.706s = 213.950762.
- Lift coefficient C_L. Also based on the calibration reported by Kobayashi and Greenwald (1988), the lift coefficient `CL`=0.3 for ξ=3.15.
- `ISAVA`=1, `MSAVA1`=210, `MSAVA2`=213, `NTIMES`=4 so that the spatial variations of η and u will be stored at the normalized time t = 209.25, 209.50, 209.75, ..., 213.

The primary input data file for this computation is presented in Table E-1 and stored in the file `appene.inp` on the accompanying disk. Table E-2 contains the file `ODOC` stored in the file `appene.doc` on the disk. The file `BINPUT` stored in the file `appene.bin` on the disk is presented in Table E-3.

Only one computation is performed for Run I3 using `DELTA`=0.001 as input and `MULTIF`=1. Spikes do appear in the spatial variations of u_{max} and u_{min} (see Figure E-6). A way to reduce or eliminate the spikes has been given in Appendix D.

The BEFORR computation for this example also involves reversals of the computation. The portion of the computation log written in the file BDOC that shows when the reversals occurred is presented in Table E-4. The computation was in the unit M=164 when the problem IPROB=3 for the violation of the numerical stability criterion occurred. BEFORR performed nine unsuccessful trial runs before the numerical stability indicator Ω exceeded 20 at the tenth run. The reversal was made to t=160 (LEVEL=2). The following 19-line message was produced by BEFORR just before the reversal was made.

```
From Subr. 11 NUMSTA: IPROB=3
(Stability criterion violated)
Computation unit              M =     164
Waterline node                S =     453
Computation parameters:    NONE =   94250
                          OMEGA =      25.55
Time of occurrence         TIME =     163.750684350
Number of occurrence     ICOUNT =       1
Indicators            ITRY,IREV =   9,  0

From Subr. 10 CPARAM:
Computation unit                M =     164
Numerical stability indicator > 20:
                           OMEGA =      25.55
Number of failed trials     ITRY =      10
Reverse indicator           IREV =       0
Last used:                  NONE =   94250
                           DELTA =       0.000573000
Computation reversed to LEVEL=2 or LEVEL=3.
```

The computation was reversed again to t=160 (LEVEL=2) at approximately the same time in the computation unit 164. Just before the second reversal was made, the following message was displayed on screen as well as written in the file BMSG.

```
From Subr. 11 NUMSTA: IPROB=3
(Stability criterion violated)
Computation unit              M =     164
Waterline node                S =     453
Computation parameters:    NONE =   84150
                          OMEGA =      22.81
Time of occurrence         TIME =     163.751016043
Number of occurrence     ICOUNT =       2
Indicators            ITRY,IREV =   7,  1

From Subr. 10 CPARAM:
Computation unit                M =     164
Numerical stability indicator > 20:
```

```
                              OMEGA =      22.81
Number of failed trials        ITRY =       8
Reverse indicator              IREV =       1
Last used:                     NONE =   84150
                              DELTA =      0.000573000
Computation reversed to LEVEL=2 or LEVEL=3.
```

In addition to Figure E-1 for the incident wave time series at the seaward boundary, the following time series at the seaward boundary are plotted: the reflected wave train η_r=ETAR in Figure E-2, and the depth-averaged velocity $u(x=0)$=V(1) in Figure E-4. The data points for Figures E-1 and E-2 are retrieved from the columns 2 and 3, respectively, of the file OSEAWAV. Stored in the column 4 of the same file is the time series of the volume flux $m(x=0)$, from which the the depth-averaged velocity $u(x=0)$ can be obtained using the relationship $u(x=0)=m(x=0)/h(x=0)$ where $h(x=0)$ is given by Eq. 12.

Figure E-3 shows the time series of the waterline elevation Z_r=RUNUPS(2) corresponding to the physical water depth δ_r'=27.5mm. The data points are retrieved from the column 3 of the file ORUNUP.

The rest of the figures in this Appendix are the spatial variations arranged in the same order as in Appendix D. The index j is used to indicate the spatial node j located at the normalized location x=(j−1)Δx.

- Figure E-5: the spatial variations of $\overline{\eta}$=ESTAT(1,j), η_{max}=ESTAT(2,j), η_{min}= ESTAT(3,j), and the bottom elevation z=ZB(j).
- Figure E-6: the spatial variations of $\overline{u}$=VSTAT(1,j), u_{max}=VSTAT(2,j), and u_{min}= VSTAT(3,j).
- Figure E-7: the spatial variation of $\overline{m}$=U1STAT(1,j) where $\overline{m} \simeq 0$.
- Figure E-8: the spatial variations of the energy quantities $\overline{E}$=ENER(1,j), $\overline{E_F}$= ENER(2,j), $\overline{D_f}$=ENER(3,j), and $\overline{D_B}$=ENER(4,j).
- Figure E-9: the spatial variations of the instantaneous free surface elevation η=ELEV(j) at the 16 specified moments for the option ISAVA=1.
- Figure E-10: the spatial variations of the instantaneous depth-averaged velocity u=V(j) at the 16 specified moments.
- Figure E-11: the spatial variations of u=ATMIN(2,j), du/dt=ATMIN(3,j), and N_R= ATMIN(4,j) at the time of the minimum armor stability:
- Figure E-12 : the spatial variation of the local stability number N_{sx}= SNSX(j). The critical stability number N_{sc}, which is the smallest value of N_{sx} along the slope, is 0.348, and occurs at the node number 390 at the time t=74.122 as listed in Table E-2.

Table E-1: Primary input data file for BEFORR-RBREAK computation of "Kobayashi, Cox, and Wurjanto 1990 Run I3".

```
 1          3                          --> NLINES
 2     ------------------------------------------------------------------------
 3     Kobayashi, Cox, and Wurjanto 1990 Run I3
 4     ------------------------------------------------------------------------
 5     11                               --> IJOB,ISTAB
 6     1                                --> ISYST
 7     1                                --> IBOT
 8     0                                --> INONCT
 9     1                                --> IENERG
10     2        0  tkcwi3                --> IWAVE,MSTAT,FINP2
11     1      210      213        4     --> ISAVA,MSAVA1,MSAVA2,NTIMES
12     0        0                       --> ISAVB,NNODB
13     0        0                       --> ISAVC,NNODC
14     00000        0                   --> IREQ,IELEV,IV,IDUDT,ISNR,NNREQ
15          400                         --> INITS
16            1                         --> MULTIF
17             .003000                  --> DELTA (normalized)
18          213.950762                  --> TMAX  (normalized)
19           50                         --> NRATE
20            2                         --> NDELR
21         21.000000                    --> DELRP(1)(millimeters)
22         27.500000                    -->       (2)
23          2.000000      2.000000      --> X1,X2
24           .050800      1.706000      --> HREFP(meters),TP(seconds)
25          1.000000      1.000000      --> KSREF,KSSEA
26           .100000                    --> FWP
27           .400000                    --> DSEAP(meters)
28           .333333                    --> TSLOPS
29            1                         --> NBSEG
30          2.100000       .333333      --> WBSEG(1)(meters),TBSLOP(1)
31           .900000       .600000        2.700000  --> C2,C3,SG
32           .500000       .300000        1.500000  --> CD,CL,CM
33          1.191754                                 --> TANPHI
34          1.000000      -.800000                   --> AMAX,AMIN
```

Table E-2: File ODOC produced by RBREAK computation of "Kobayashi, Cox, and Wurjanto 1990 Run I3".

```
 1  ------------------------------------------------------------------------
 2  Kobayashi, Cox, and Wurjanto 1990 Run I3
 3  ------------------------------------------------------------------------
 4
 5  WAVE CONDITION
 6
 7  Incident Wave at Seaward Boundary Given as Input
 8  Number of Data Points    NDATA =    5841
 9  Norm. Maximum Surface Elev.    =       0.972600
10  Norm. Minimum Surface Elev.    =      -0.848400
11
12  Reference Wave Period          =       1.706000 sec.
13  Reference Wave Height          =       0.050800 meters
14  Depth at Seaward Boundary      =       0.400000 meters
15  Shoal. Coef. at Reference  Ks1 =       1.000
16               at Seaw. Bdr. Ks2 =       1.000
17               Ks = Ks2/Ks1      =       1.000
18  Norm. Depth at Seaw. Boundary  =       7.874
19  Normalized Wave Length         =       7.666
20  "Sigma"                        =      23.707
21  Ursell Number                  =       7.464
22  Surf Similarity Parameter      =       3.153
23
24  SLOPE PROPERTIES
25
26  Friction Factor            =        0.100000
27  Norm. Friction Factor      =        1.185364
28  Norm. Horiz. Length of
29        Computation Domain   =        1.741219
30  Number of Segments         =        1
31  --------------------------------
32   SEGMENT     WBSEG(I)    TBSLOP(I)
33         I       meters
34  --------------------------------
35         1    2.100000     0.333333
36  --------------------------------
37
38  PARAMETERS FOR ARMOR STABILITY AND MOVEMENT
39
40  Armor Friction Factor      =     1.192
41  Specific Gravity           =     2.700
42  Area Coefficient       C2  =     0.900
43  Volume Coefficient     C3  =     0.600
44  Drag Coefficient       CD  =     0.500
45  Lift Coefficient       CL  =     0.300
46  Inertia Coefficient    CM  =     1.500
47  Norm. Upper and Lower Bounds of du/dt
48                       AMAX  =     1.000
49                       AMIN  =    -0.800
50
```

Table E-2 continued

```
 51     COMPUTATION PARAMETERS
 52
 53     Total Number of Spatial Nodes      JE =      700
 54     Number of Nodes Along Bottom Below SWL
 55                                       INITS =      400
 56     Normalized Delta x          =        0.249101D-02
 57     Damping Coefficients    x1 =           2.000
 58                             x2 =           2.000
 59     Computation Duration  TMAX =         213.950762 Wave Periods
 60     Minimum Allowable NONE           NONEM =     3700
 61     Time Series Stored at Rate NRATE
 62          per Wave Period with        NRATE =       50
 63     Statistical Calculations Are Performed Excluding
 64          the First MSTAT Computation Units
 65          with                         MSTAT =        0
 66     Computation of Armor Stability or Movement is Performed
 67          Excluding the First MSTAB Computation Units
 68          with                         MSTAB =        0
 69     Spatial Variations are Stored NTIMES at Equal
 70          Intervals per Wave Period from
 71          Computation Unit MSAVA1 to MSAVA2,
 72          Inclusive, with             NTIMES =        4
 73                                      MSAVA1 =      210
 74                                      MSAVA2 =      213
 75
 76     REFLECTION COEFFICIENT
 77
 78     r1 =     1.010
 79
 80     RUNUP, RUNDOWN, SETUP
 81
 82     Largest Node Number Reached by Computational Waterline
 83                                                JMAX =      596
 84     ------------------------------------------------------------
 85            I   DELTAR(I)     RUNUP(I)    RUNDOWN(I)    SETUP(I)
 86                   [mm]          R            Rd           Zr
 87     ------------------------------------------------------------
 88            1    21.000        3.062       -2.317        0.163
 89            2    27.500        2.949       -2.314        0.128
 90     ------------------------------------------------------------
 91
 92     WAVE SET-DOWN OR SETUP
 93
 94     Average value of ETAI =    -0.000001
 95                      ETAR =     0.002231
 96
```

Table E-2 continued

```
 97     TIME-AVERAGED ENERGY BALANCE
 98
 99     Normalized Energy Flux:
100     . at Seaw. Boundary   A =  0.113435D+00
101     . at Landw. Boundary  B =  0.000000D+00
102     Normalized Rate of Energy Dissipation
103     in the Computation Domain, Due to:
104     . Bottom Friction     C =  0.964867D-01
105     . Wave Breaking       D =  0.156039D-01
106     Additional Term       E =  0.134581D-02
107     Calculation 1:
108                     F = A-B =  0.113435D+00
109                   G = C+D+E =  0.113436D+00
110       Must H=0, but H = G-F =  0.154747D-05
111       % error  100G/E =          0.00
112
113     STABILITY NUMBER
114
115     Largest Node Number for Which Armor Stability
116       Computation Performed  JSTABM =          535
117     Critical Stability Number   Nsc =        0.348
118     At Node Number                  =          390
119     At Time                         =           74.122162
```

Table E-3: File BINPUT produced by BEFORR computation of "Kobayashi, Cox, and Wurjanto 1990 Run I3".

```
 1          3
 2    ------------------------------------------------------------------------
 3    Kobayashi, Cox, and Wurjanto 1990 Run I3
 4    ------------------------------------------------------------------------
 5         214
 6    tkcwi3
 7        5841
 8        3700
 9           1    3700    3700   0.100000000D-02
10           2    3700    3700   0.100000000D-02
11           3    3700    3700   0.100000000D-02
12           4    3700    3700   0.100000000D-02
13           5    3700    3700   0.100000000D-02
14           6    3700    3700   0.100000000D-02
15           7    3700    3700   0.100000000D-02
16           8    3700    3700   0.100000000D-02
17           9    3700    3700   0.100000000D-02
18          10    3700    3700   0.100000000D-02
19          11    3700    3700   0.100000000D-02
20          12    3700    3700   0.100000000D-02
21          13    3700    3700   0.100000000D-02
22          14    3700    3700   0.100000000D-02
23          15    3700    3700   0.100000000D-02
24          16    3700    3700   0.100000000D-02
25          17    3700    3700   0.100000000D-02
26          18    3700    3700   0.100000000D-02
27          19    3700    3700   0.100000000D-02
28          20    3700    3700   0.100000000D-02
29          21    3700    3700   0.100000000D-02
30          22    3700    3700   0.100000000D-02
31          23    3700    3700   0.100000000D-02
32          24    3700    3700   0.100000000D-02
33          25    3700    3700   0.100000000D-02
34          26    3700    3700   0.100000000D-02
35          27    3700    3700   0.100000000D-02
36          28    3700    3700   0.100000000D-02
37          29    3700    3700   0.100000000D-02
38          30    3700    3700   0.100000000D-02
39          31    3700    3700   0.100000000D-02
40          32    3700    3700   0.944000000D-03
41          33    3700    3700   0.944000000D-03
42          34    3700    3700   0.944000000D-03
43          35    3700    3700   0.944000000D-03
44          36    3700    3700   0.944000000D-03
45          37    3700    3700   0.944000000D-03
46          38    3700    3700   0.944000000D-03
47          39    3700    3700   0.944000000D-03
48          40    3700    3700   0.944000000D-03
49          41    3700    3700   0.944000000D-03
50          42    3700    3700   0.944000000D-03
51          43    3700    3700   0.944000000D-03
52          44    3700    3700   0.944000000D-03
53          45    3700    3700   0.944000000D-03
54          46    3700    3700   0.944000000D-03
55          47    3700    3700   0.944000000D-03
56          48    3700    3700   0.944000000D-03
```

Table E-3 continued

57	49	3700	3700	0.944000000D-03
58	50	3700	3700	0.944000000D-03
59	51	3700	3700	0.944000000D-03
60	52	3700	3700	0.944000000D-03
61	53	3700	3700	0.944000000D-03
62	54	3700	3700	0.944000000D-03
63	55	3700	3700	0.944000000D-03
64	56	3700	3700	0.944000000D-03
65	57	3700	3700	0.944000000D-03
66	58	3700	3700	0.944000000D-03
67	59	3700	3700	0.944000000D-03
68	60	3700	3700	0.944000000D-03
69	61	3700	3700	0.944000000D-03
70	62	3700	3700	0.944000000D-03
71	63	3700	3700	0.944000000D-03
72	64	3700	3700	0.944000000D-03
73	65	3700	3700	0.944000000D-03
74	66	3700	3700	0.944000000D-03
75	67	3700	3700	0.944000000D-03
76	68	3700	3700	0.944000000D-03
77	69	3700	3700	0.894000000D-03
78	70	3700	3700	0.894000000D-03
79	71	3700	3700	0.894000000D-03
80	72	3700	3700	0.894000000D-03
81	73	3700	3700	0.894000000D-03
82	74	3700	3700	0.894000000D-03
83	75	3700	3700	0.894000000D-03
84	76	3700	3700	0.894000000D-03
85	77	3700	3700	0.894000000D-03
86	78	3700	3700	0.894000000D-03
87	79	3700	3700	0.894000000D-03
88	80	3700	3700	0.894000000D-03
89	81	3700	3700	0.894000000D-03
90	82	3700	3700	0.894000000D-03
91	83	3700	3700	0.894000000D-03
92	84	3700	3700	0.894000000D-03
93	85	3700	3700	0.894000000D-03
94	86	3700	3700	0.894000000D-03
95	87	3700	3700	0.894000000D-03
96	88	3700	3700	0.894000000D-03
97	89	3700	3700	0.894000000D-03
98	90	3700	3700	0.894000000D-03
99	91	3700	3700	0.894000000D-03
100	92	3700	3700	0.894000000D-03
101	93	3700	3700	0.894000000D-03
102	94	3700	3700	0.894000000D-03
103	95	3700	3700	0.894000000D-03
104	96	3700	3700	0.894000000D-03
105	97	3700	3700	0.894000000D-03
106	98	3700	3700	0.894000000D-03
107	99	3700	3700	0.894000000D-03
108	100	3700	3700	0.894000000D-03
109	101	3700	3700	0.894000000D-03
110	102	3700	3700	0.894000000D-03
111	103	3700	3700	0.894000000D-03
112	104	3700	3700	0.894000000D-03

Table E-3 continued

113	105	94250	94250	0.894000000D-03
114	106	3700	3700	0.894000000D-03
115	107	3700	3700	0.894000000D-03
116	108	3700	3700	0.894000000D-03
117	109	3700	3700	0.894000000D-03
118	110	3700	3700	0.894000000D-03
119	111	3700	3700	0.894000000D-03
120	112	3700	3700	0.894000000D-03
121	113	3700	3700	0.894000000D-03
122	114	3700	3700	0.894000000D-03
123	115	3700	3700	0.894000000D-03
124	116	3700	3700	0.894000000D-03
125	117	3700	3700	0.894000000D-03
126	118	3700	3700	0.894000000D-03
127	119	3700	3700	0.894000000D-03
128	120	3700	3700	0.894000000D-03
129	121	3700	3700	0.894000000D-03
130	122	3700	3700	0.894000000D-03
131	123	3700	3700	0.894000000D-03
132	124	3700	3700	0.894000000D-03
133	125	3700	3700	0.894000000D-03
134	126	3700	3700	0.894000000D-03
135	127	3700	3700	0.894000000D-03
136	128	3700	3700	0.894000000D-03
137	129	3700	3700	0.894000000D-03
138	130	3700	3700	0.894000000D-03
139	131	3700	3700	0.894000000D-03
140	132	3700	3700	0.894000000D-03
141	133	3700	3700	0.894000000D-03
142	134	3700	3700	0.894000000D-03
143	135	3700	3700	0.894000000D-03
144	136	3700	3700	0.894000000D-03
145	137	3700	3700	0.894000000D-03
146	138	3700	3700	0.894000000D-03
147	139	3700	3700	0.894000000D-03
148	140	3700	3700	0.894000000D-03
149	141	3700	3700	0.894000000D-03
150	142	3700	3700	0.894000000D-03
151	143	3700	3700	0.894000000D-03
152	144	3700	3700	0.894000000D-03
153	145	3700	3700	0.894000000D-03
154	146	3700	3700	0.894000000D-03
155	147	3700	3700	0.894000000D-03
156	148	3700	3700	0.805000000D-03
157	149	3700	3700	0.805000000D-03
158	150	3700	3700	0.805000000D-03
159	151	3700	3700	0.805000000D-03
160	152	3700	3700	0.805000000D-03
161	153	3700	3700	0.805000000D-03
162	154	3700	3700	0.805000000D-03
163	155	3700	3700	0.805000000D-03
164	156	3700	3700	0.675000000D-03
165	157	5550	5550	0.675000000D-03
166	158	3700	3700	0.622000000D-03
167	159	3700	3700	0.622000000D-03
168	160	3700	3700	0.622000000D-03

Table E-3 continued

169	161	11100	11100	0.573000000D-03
170	162	11100	11100	0.573000000D-03
171	163	11100	11100	0.573000000D-03
172	164	16650	16650	0.573000000D-03
173	165	11100	11100	0.573000000D-03
174	166	3700	3700	0.573000000D-03
175	167	3700	3700	0.573000000D-03
176	168	3700	3700	0.573000000D-03
177	169	3700	3700	0.573000000D-03
178	170	3700	3700	0.573000000D-03
179	171	3700	3700	0.573000000D-03
180	172	3700	3700	0.573000000D-03
181	173	3700	3700	0.573000000D-03
182	174	3700	3700	0.573000000D-03
183	175	3700	3700	0.573000000D-03
184	176	3700	3700	0.573000000D-03
185	177	3700	3700	0.573000000D-03
186	178	3700	3700	0.573000000D-03
187	179	3700	3700	0.573000000D-03
188	180	3700	3700	0.573000000D-03
189	181	3700	3700	0.573000000D-03
190	182	3700	3700	0.573000000D-03
191	183	3700	3700	0.573000000D-03
192	184	3700	3700	0.573000000D-03
193	185	3700	3700	0.573000000D-03
194	186	3700	3700	0.573000000D-03
195	187	3700	3700	0.492000000D-03
196	188	3700	3700	0.492000000D-03
197	189	3700	3700	0.492000000D-03
198	190	3700	3700	0.492000000D-03
199	191	3700	3700	0.492000000D-03
200	192	3700	3700	0.492000000D-03
201	193	8300	8300	0.492000000D-03
202	194	3700	3700	0.492000000D-03
203	195	3700	3700	0.492000000D-03
204	196	3700	3700	0.492000000D-03
205	197	3700	3700	0.492000000D-03
206	198	3700	3700	0.492000000D-03
207	199	3700	3700	0.492000000D-03
208	200	27950	27950	0.492000000D-03
209	201	3700	3700	0.492000000D-03
210	202	3700	3700	0.492000000D-03
211	203	3700	3700	0.492000000D-03
212	204	3700	3700	0.492000000D-03
213	205	3700	3700	0.492000000D-03
214	206	3700	3700	0.492000000D-03
215	207	3700	3700	0.492000000D-03
216	208	3700	3700	0.467000000D-03
217	209	3700	3700	0.467000000D-03
218	210	3700	3700	0.467000000D-03
219	211	3700	3700	0.467000000D-03
220	212	3700	3700	0.467000000D-03
221	213	3700	3700	0.467000000D-03
222	214	3700	3517	0.467000000D-03

Table E-4: A portion of file BDOC produced by BEFORR computation of "Kobayashi, Cox, and Wurjanto 1990 Run I3".

```
1   ------------------------------------------------------------------------
2   Kobayashi, Cox, and Wurjanto 1990 Run I3
3   ------------------------------------------------------------------------
4
.
.
.
86
87  COMPUTATION LOG
88
89  ------------------------------------------------------------------------
90   Comp.   NONE     NEND      NEND/        DELTA      Num.      No.     Re-
91   Unit                       NONE                    Stab.      of    verse
92                                                      Indic.  Trials  Indic.
93  ------------------------------------------------------------------------
94       1    3700     3700       1.0  0.00100000      1.00       1       0
95       2    3700     3700       1.0  0.00100000      1.00       1       0
96       3    3700     3700       1.0  0.00100000      1.00       1       0
 .       .
 .       .
 .       .
253    160    3700     3700       1.0  0.00062200      1.00       1       0
254    161    3700     3700       1.0  0.00057300      1.00       2       0
255    162    5550     5550       1.0  0.00057300      1.50       2       0
256    163    5550     5550       1.0  0.00057300      1.50       2       0
257    161    7400     7400       1.0  0.00057300      2.01       1       1
258    162    7400     7400       1.0  0.00057300      2.01       1       1
259    163    7400     7400       1.0  0.00057300      2.01       1       1
260    161   11100    11100       1.0  0.00057300      3.01       1       2
261    162   11100    11100       1.0  0.00057300      3.01       1       2
262    163   11100    11100       1.0  0.00057300      3.01       1       2
263    164   16650    16650       1.0  0.00057300      4.51       2       2
264    165   11100    11100       1.0  0.00057300      3.01       1       2
265    166    3700     3700       1.0  0.00057300      1.00       1       0
266    167    3700     3700       1.0  0.00057300      1.00       1       0
267    168    3700     3700       1.0  0.00057300      1.00       1       0
268    169    3700     3700       1.0  0.00057300      1.00       1       0
269    170    3700     3700       1.0  0.00057300      1.00       1       0
 .       .
 .       .
 .       .
```

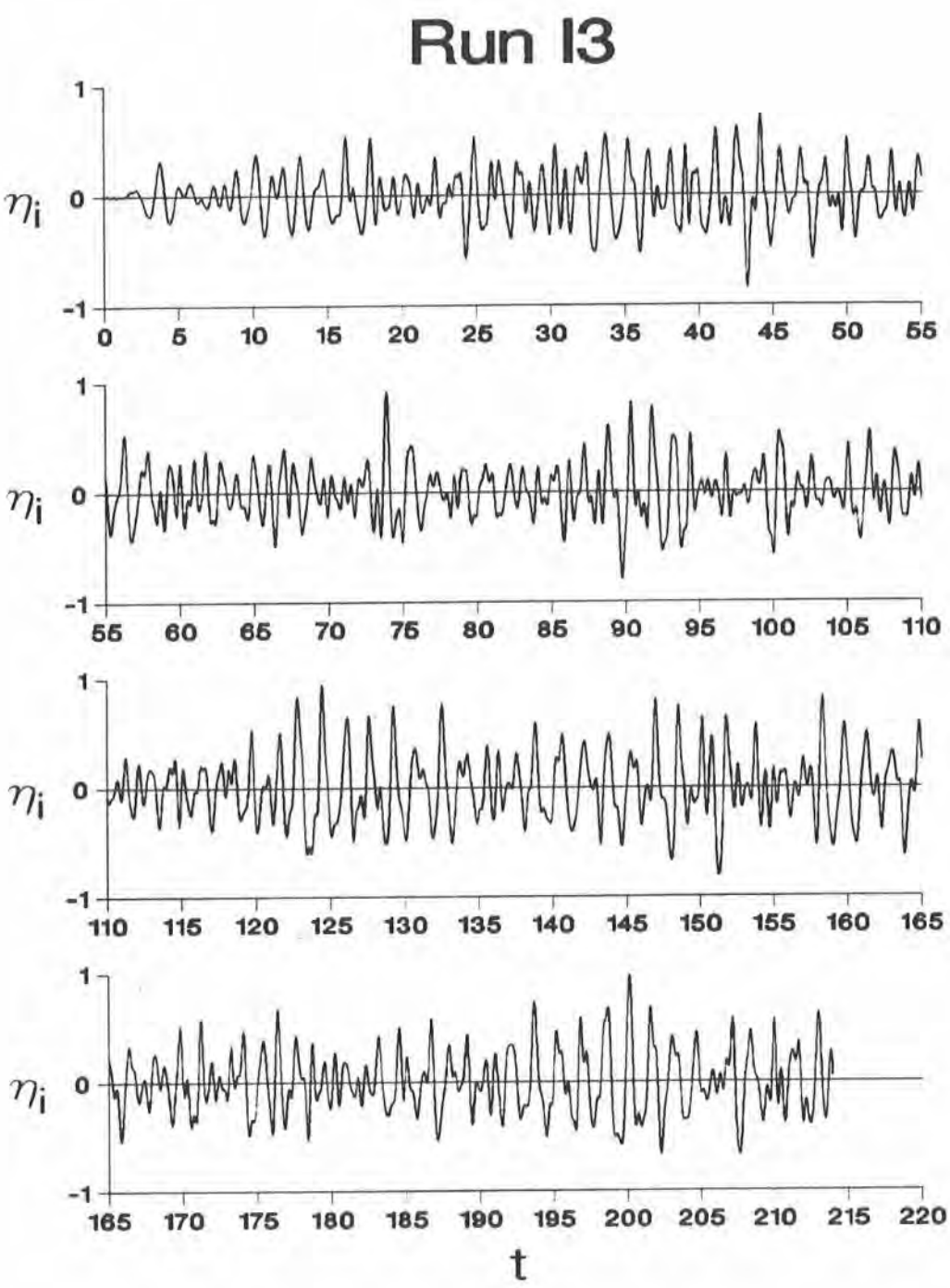

Run I3
η_i
1
0
-1
0 5 10 15 20 25 30 35 40 45 50 55
55 60 65 70 75 80 85 90 95 100 105 110
110 115 120 125 130 135 140 145 150 155 160 165
165 170 175 180 185 190 195 200 205 210 215 220
t

Figure E-1

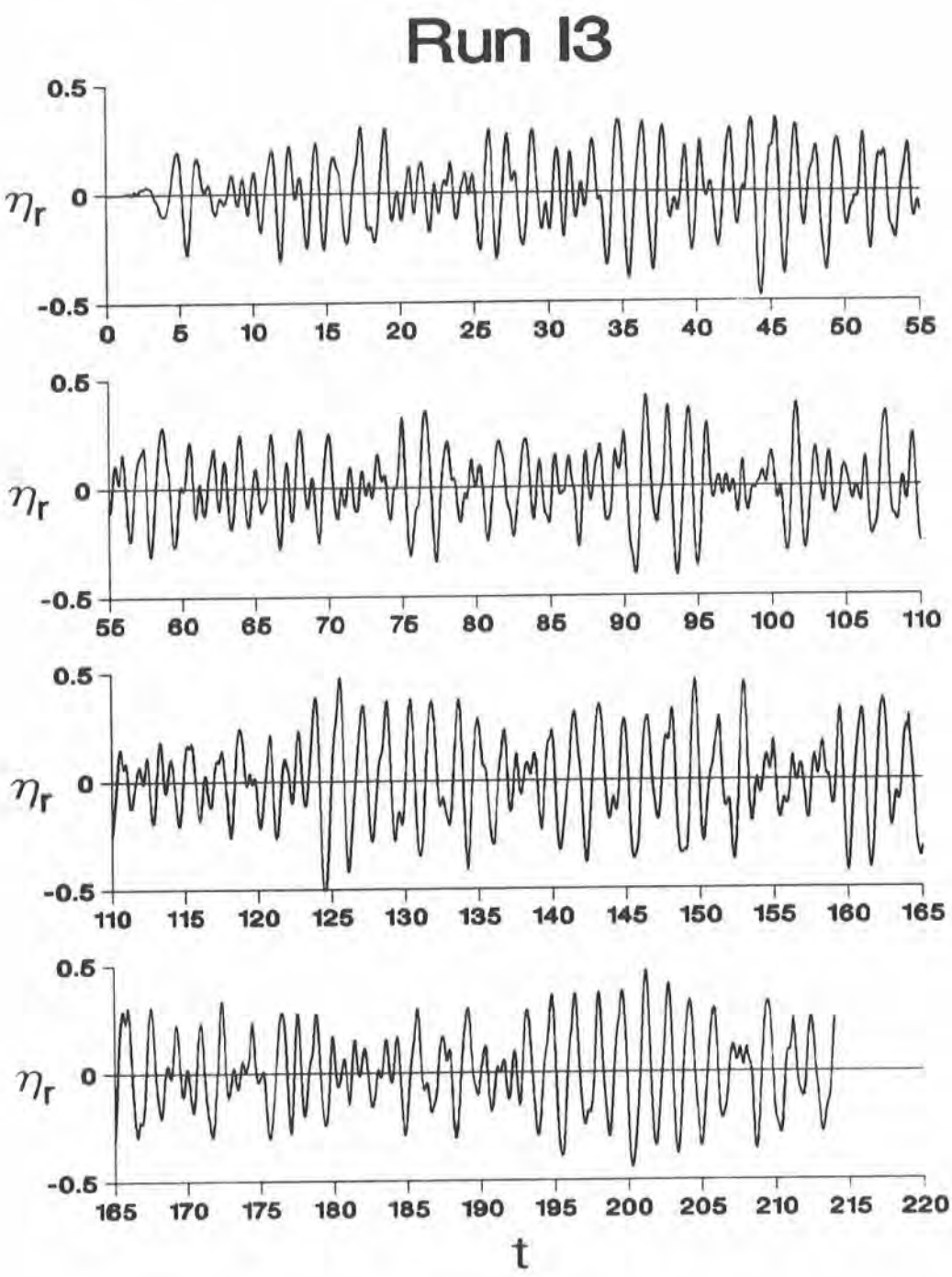

Run I3
0.5
η_r
0
-0.5
0 5 10 15 20 25 30 35 40 45 50 55
0.5
η_r
0
-0.5
55 60 65 70 75 80 85 90 95 100 105 110
0.5
η_r
0
-0.5
110 115 120 125 130 135 140 145 150 155 160 165
0.5
η_r
0
-0.5
165 170 175 180 185 190 195 200 205 210 215 220
t

Figure E-2

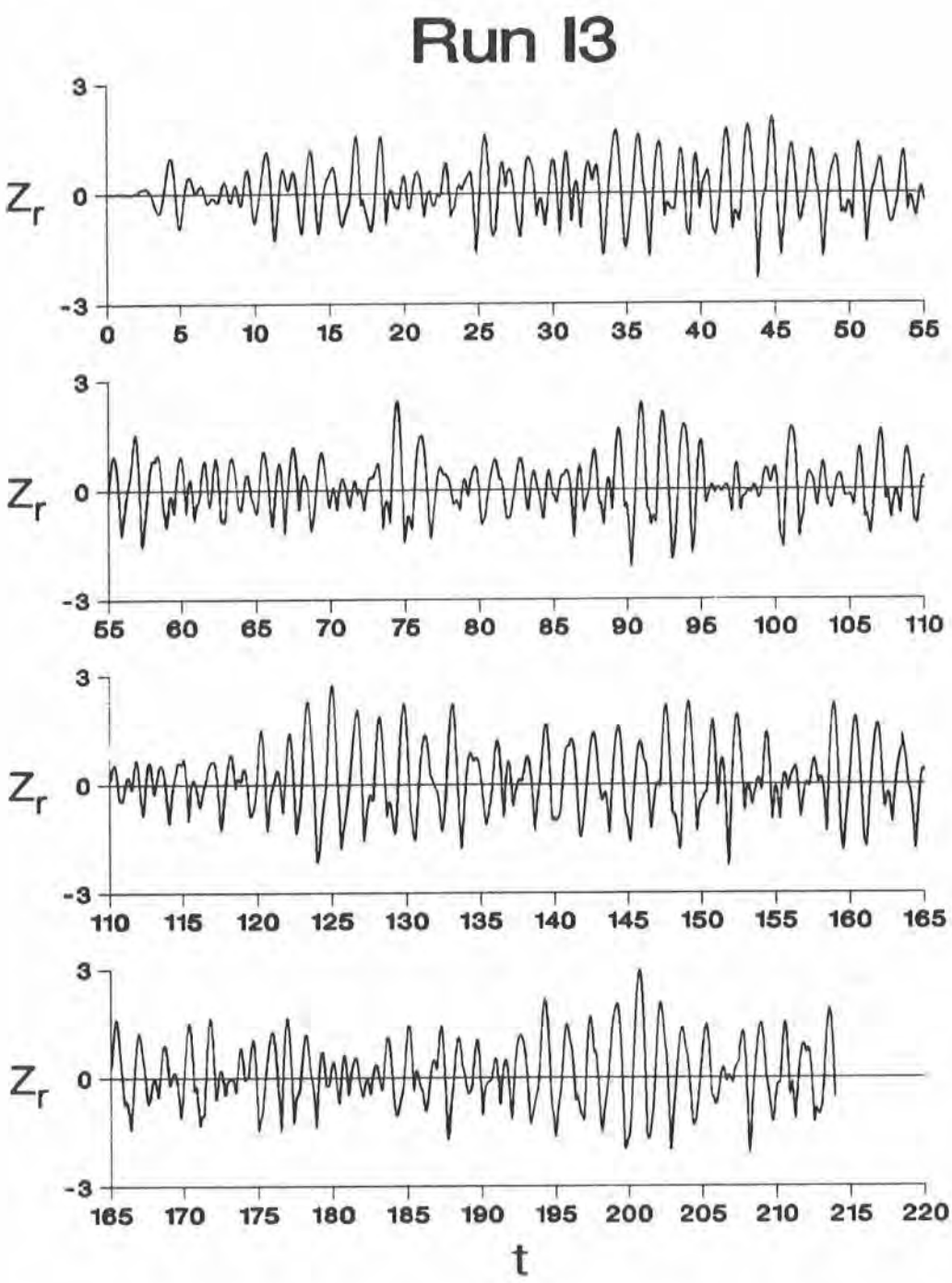

Run I3
3
Z_r 0
-3
0 5 10 15 20 25 30 35 40 45 50 55
3
Z_r 0
-3
55 60 65 70 75 80 85 90 95 100 105 110
3
Z_r 0
-3
110 115 120 125 130 135 140 145 150 155 160 165
3
Z_r 0
-3
165 170 175 180 185 190 195 200 205 210 215 220
t

Figure E-3

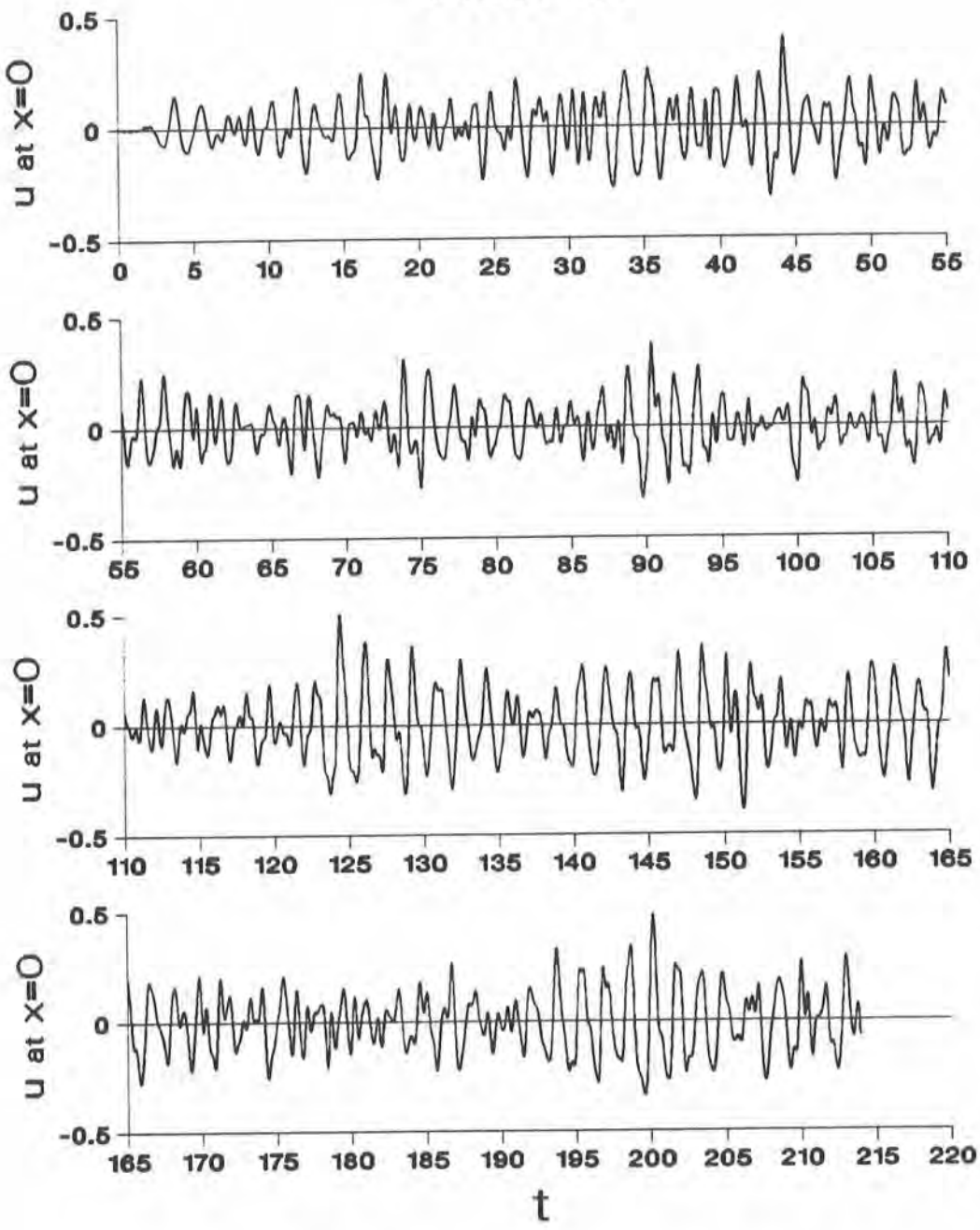

Run I3
u at X=0
0.5
0
-0.5
0 5 10 15 20 25 30 35 40 45 50 55
55 60 65 70 75 80 85 90 95 100 105 110
110 115 120 125 130 135 140 145 150 155 160 165
165 170 175 180 185 190 195 200 205 210 215 220
t

Figure E-4

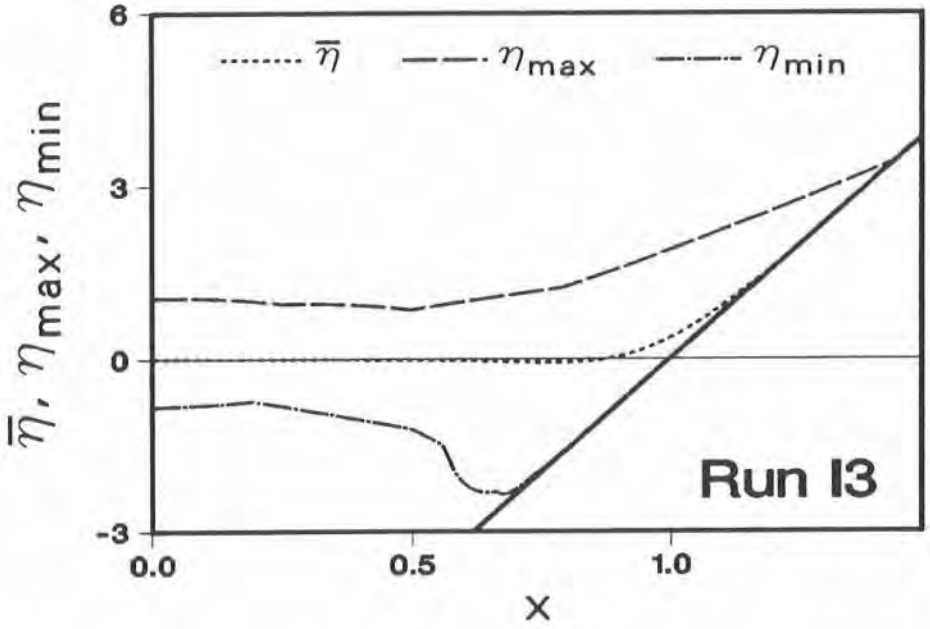


Figure E-5

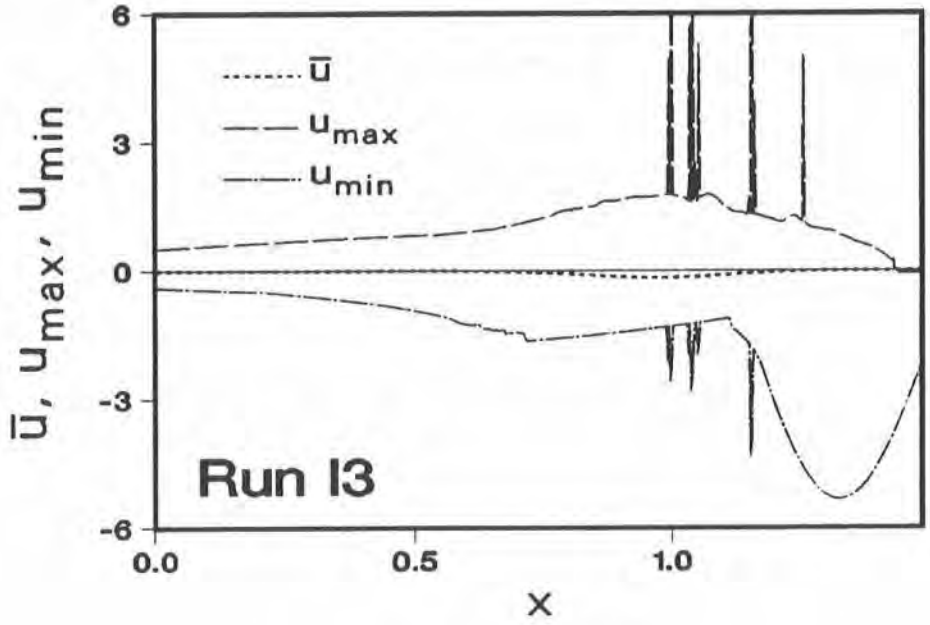


Figure E-6

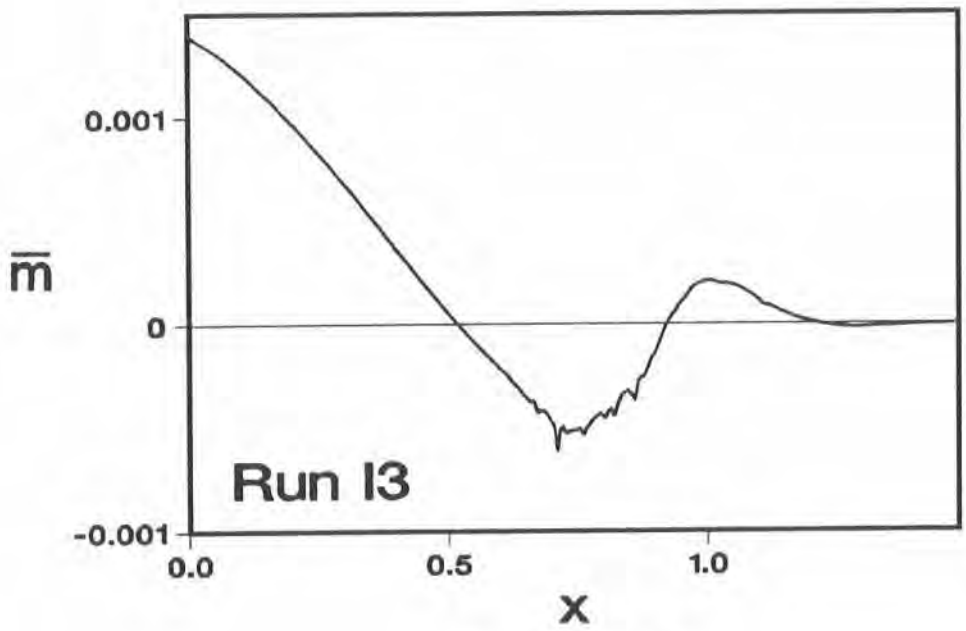


Figure E-7

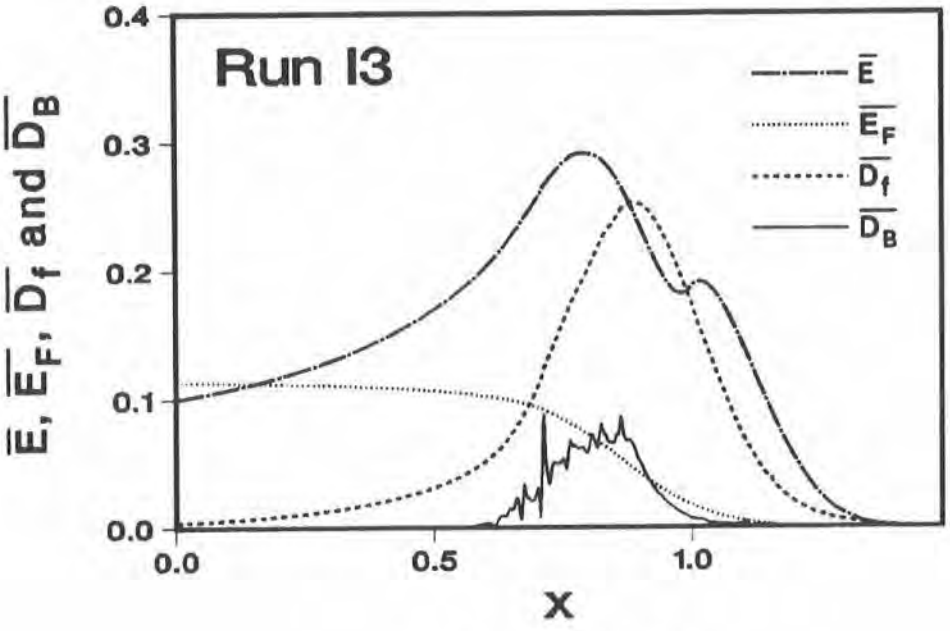


Figure E-8

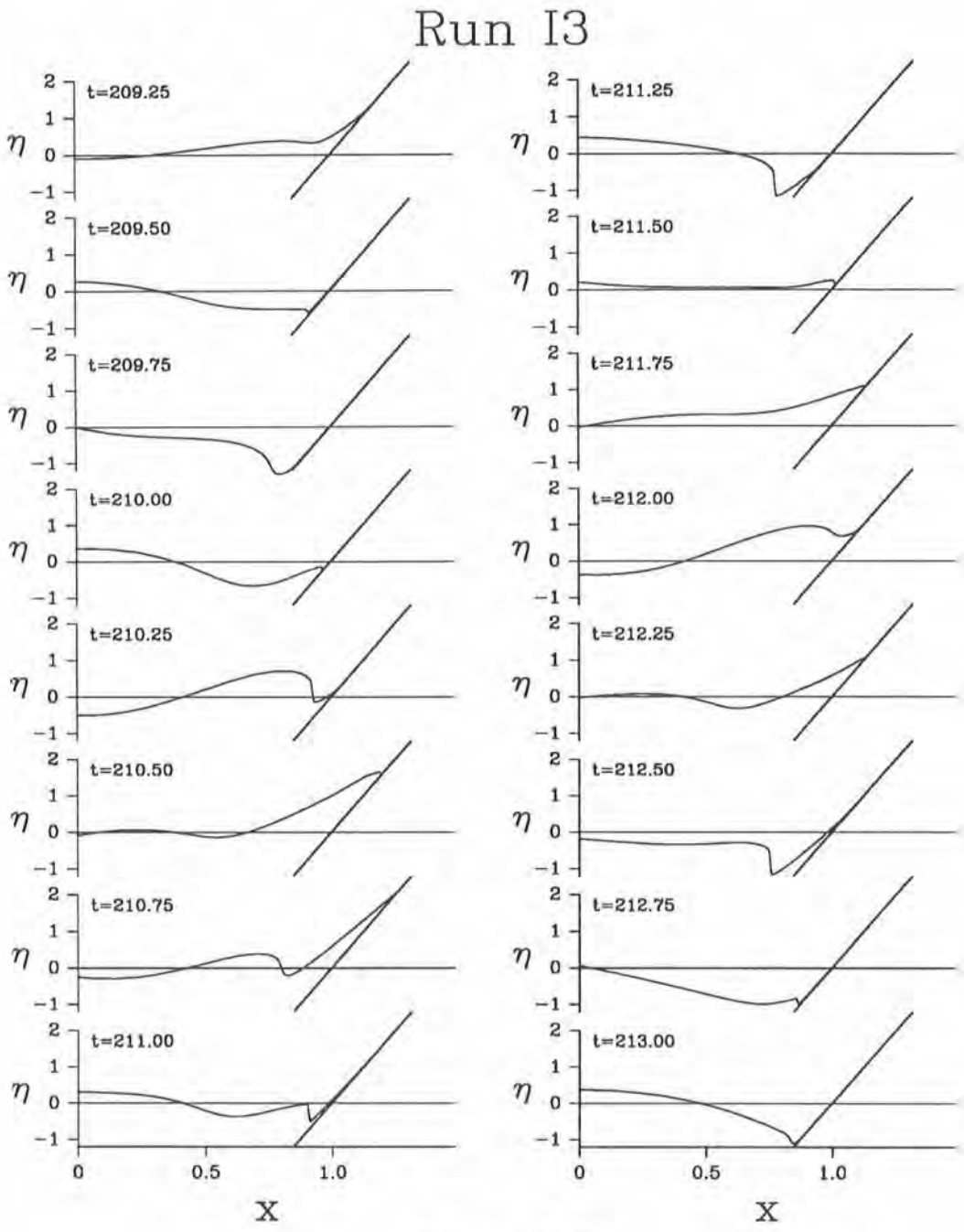

Run I3
t=209.25
t=209.50
t=209.75
t=210.00
t=210.25
t=210.50
t=210.75
t=211.00
t=211.25
t=211.50
t=211.75
t=212.00
t=212.25
t=212.50
t=212.75
t=213.00
η
X
0
0.5
1.0
2
1
0
−1

Figure E-9

Run I3

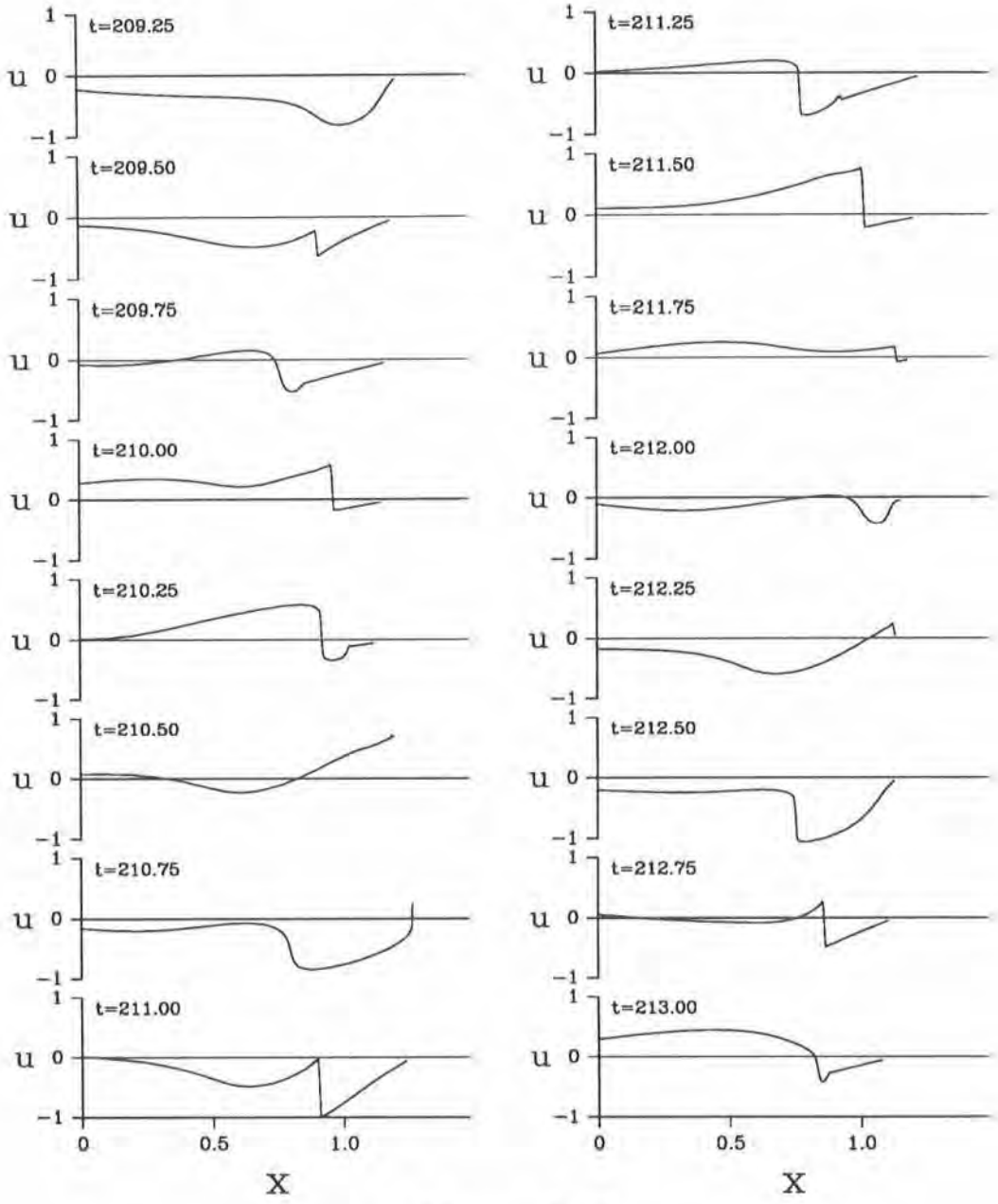


Figure E-10

Run I3

Critical Conditions:

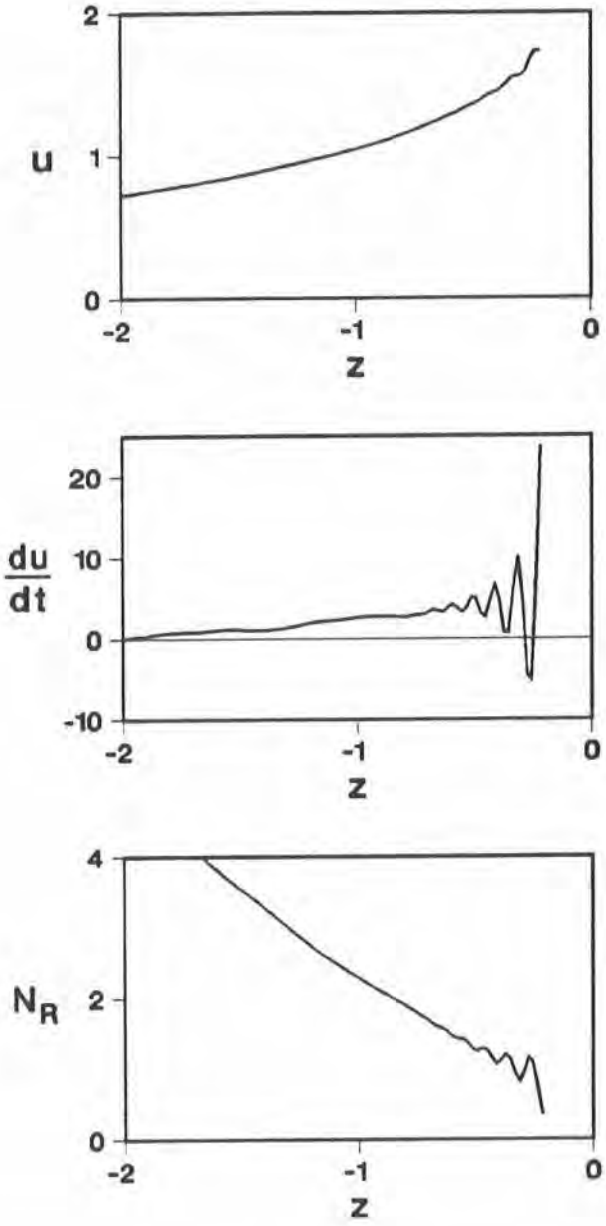


Figure E-11

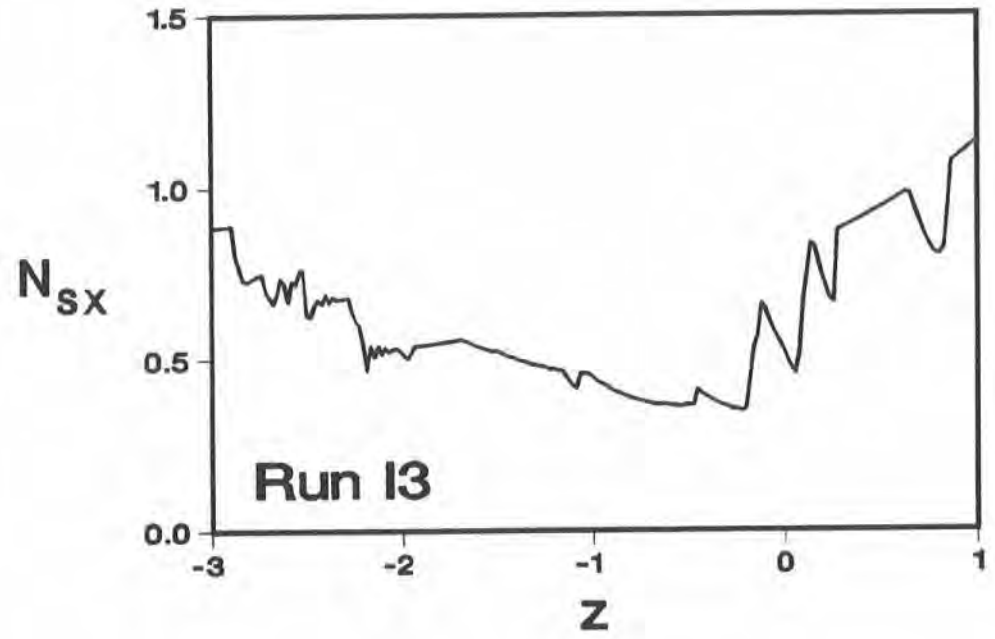


Figure E-12

APPENDIX F

EXAMPLE: INPUT WAVE TRAIN FROM TMA SPECTRUM

This example features the use of (1) an input wave train generated from the TMA spectrum, and (2) irregular bottom geometry. This example is Run 1 in Kobayashi and Wurjanto (1991), which deals with a *beach* in contrast to *coastal structures* discussed in Appendices C, D, and E.

• BOTTOM GEOMETRY •

Figure F-1 shows the bottom geometry, which was digitized from the profile given in Fig. 1 of Holman and Sallenger (1985). The same profile was used by Kobayashi, Strzelecki, and Wurjanto (1988) for regular wave computations. The seaward boundary for the numerical computation is indicated by the vertical line at x'=0. Run 1 corresponded to the "High Tide" condition, where the water depth at the seaward boundary DSEAP=4.15m (International System of Units; ISYST=1). The input bottom geometry comprises 30 linear segments (NBSEG=30), and is specified using IBOT=2. The location of each data point [XBSEG(i),ZBSEG(i)] with i=1,2,...,31 is specified in the primary input data file. The surf similarity parameter ξ was defined using on the average foreshore slope TSLOPS=0.118 (Kobayashi, Strzelecki, and Wurjanto 1988). The bottom friction factor FWP is taken to be 0.05, which has been used for beaches (Kobayashi and Watson 1987; Kobayashi, DeSilva, and Watson 1989).

• INPUT WAVE TRAIN •

The incident random wave train used as the input wave train for this example (IWAVE=2) was generated from the TMA spectrum (Bouws *et al.* 1985) following the procedure described in Kobayashi, Wurjanto, and Cox (1990a) as well as Kobayashi and Wurjanto (1991). The input wave train is depicted in Figure F-2. At the seaward boundary, the *spectral estimate* of the significant wave height is 2.06m, whereas the spectral peak period is 7s. These two quantities are taken as the reference wave height and period, respectively. Thus, HREFP=2.06m, TP=7s, and KSREF=KSSEA=1. The generated wave train is 256 wave periods long $\Longrightarrow$ TMAX=256. The name of the file containing the input wave train is FINP2=tkw1 for this IBM 3090 run and `appenf.iwt` on the accompanying disk.

• OTHER INPUT •

- IJOB=1. The shoreline oscillation on a beach is computed for this example.

- ISTAB=0. The movement of sand particles is not computed, although it could be computed (Kobayashi and DeSilva 1987).
- INONCT=1 for the gentle beach slope.
- IENERG=1. Computation of the energy quantities is made.
- MSTAT=8. The first eight wave periods are excluded from the statistical calculations. This computation is intended to model a stationary random sea state. The transition period from the initial quiescent condition should be excluded.
- ISAVA=1, MSAVA1=253, MSAVA2=256, NTIMES=4. The spatial variations of η and u are stored at the normalized time t = 252.25, 252.50, 252.75, ..., 256.
- ISAVB=1, NNODB=10. The time series of m=U(1,j_B) and h= U(2,j_B) at the ten specified nodes j_B = NODB = 40, 80, 120, ..., 400 are stored during the computation.
- ISAVC=0 because the sediment movement is not computed.
- IREQ=0. No special request is made. IELEV, IV, IDUDT, ISNR, and NREQ are accordingly set to zero.
- NDELR=2. Two water depths are specified depending on how the shoreline oscillations were measured: DELRP(1)=5mm for the photography technique used by Holman and Sallenger (1985), and DELRP(2)=30mm for the wire sensor used by Guza and Thornton (1982).
- NRATE=32. The rate of storing the computed time series is 32 data points per wave period.
- INITS=400. The resulting grid spacing is 0.338m, which is small compared to the characteristic wave height at the seaward boundary, H'=2.06m, and the linear wavelength based on the characteristic wave period at the seaward boundary, L'=41.12m.
- X1=X2=2 is used for this example.
- The values of DELTA and MULTIF are discussed in relation with the two different computations of Run 1 described below.

• COMPUTATION 1 •

The first computation is performed with DELTA=0.001 as input and MULTIF=1. The primary input data file for this computation is presented in Table F-1 and in the file `appenf.inp` on the accompanying disk. The corresponding files ODOC and BINPUT are presented in Tables F-2 and F-3, respectively, and are stored in the files `appenf.doc` and `appenf.bin`, respectively, on the disk accompanying this report.

Most of the figures presented in this Appendix are from Computation 1. The nodal index j appearing in the following corresponds to the normalized location x=(j−1)Δx.

- Figure F-3 shows the time series of the reflected wave train at the seaward boundary, η_r=ETAR. The data points are retrieved from the column 3 of the file OSEAWAV.
- Figure F-4 shows the time series of the shoreline elevation Z_r=RUNUPS(1) corresponding to the physical water depth δ_r'=5mm [=DELRP(1)]. The data points are retrieved from the file ORUNUP, column 2.

- Figures F-5 and F-6 show the time series of the free surface elevation η at the locations x=1.71 (node j_B=160) and x=3.43 (node j_B=320), respectively. The data points for these two figures are obtained from the files OSAV2B01, OSAV2B02, and OSAV2B03 that contain the first 100, the second 100, and the last 56 wave periods, respectively, of the time series of h=U(2,j_B) at the ten specified nodes j_B for the option ISAVB=1. The free surface elevation η at the node j_B can be obtained by adding z=ZB(j_B) to h=U(2,j_B) as explained in Eq. 8.

- Figures F-7, F-8, and F-9 show the spatial variations of the statistical quantities of η, u, and m, respectively. The data points for these figures are provided by the file OSTAT as well as the file OSPACE for the bottom geometry in Figure F-7. In each of these three figures, the upper panel [Figure (a)] is from Computation 1, the lower panel [Figure (b)] is from Computation 2 as explained more under the heading "COMPUTATION 2".

  - In Figure F-7:
    - $\overline{\eta}$ = ESTAT(1,j)
    - η_{max} = ESTAT(2,j)
    - η_{min} = ESTAT(3,j)
    - z=ZB(j)
  - In Figure F-8:
    - $\overline{u}$ = VSTAT(1,j)
    - u_{max} = VSTAT(2,j)
    - u_{min} = VSTAT(3,j)
  - In Figure F-9:
    - $\overline{m}$ = U1STAT(1,j)

- Figure F-10 shows the spatial variations of the time-averaged energy quantities. The data points are retrieved from the file OENERG.

  - $\overline{E}$ = ENER(1,j)
  - $\overline{E_F}$ = ENER(2,j)
  - $\overline{D_f}$ = ENER(3,j)
  - $\overline{D_B}$ = ENER(4,j)

  The upper panel of Figure F-10 is from Computation 1, whereas the lower panel is from Computation 2.

- Figures F-11 and F-12 show the spatial variations of the instantaneous hydrodynamic quantities η=ELEV(j) and u=V(j), respectively, at the 16 specified moments for the option ISAVA=1. The data points are from the file OSPACE.

• COMPUTATION 2 •

Figure F-8a resulting from Computation 1 exhibits unrealistically large spikes of u_{max} and u_{min} above the still water shoreline similar to the problem discussed in Appendix D. Computation 2 is performed to show that these spikes of u_{max} and u_{min} can be reduced or eliminated by using a larger value of the input parameter `MULTIF`. First, `MULTIF`=4 was tried while `DELTA` was kept the same as in Computation 1, that is, `DELTA`=0.001 as input. This attempt failed. The next run was successful using `DELTA`=0.002 as input and `MULTIF`=3. This example points out that it may be necessary to increase the value of `DELTA` when the value of `MULTIF` is increased. The difference between Computation 1 and Computation 2 can be clearly observed in Figure F-8a from Computation 1 as compared to Figure F-8b from Computation 2. The use of a larger value of `MULTIF` does reduce the numerical spikes of u_{max} and u_{min}. There is no significant difference between the upper and lower panels of Figures F-7, F-9, and F-10.

Table F-1: Primary input data file for BEFORR-RBREAK computation of "Kobayashi & Wurjanto 1991 Run 1".

```
 1          3                          --> NLINES
 2    ------------------------------------------------------------------------
 3    Kobayashi & Wurjanto 1991 Run 1
 4    ------------------------------------------------------------------------
 5    10                              --> IJOB,ISTAB
 6    1                               --> ISYST
 7    2                               --> IBOT
 8    1                               --> INONCT
 9    1                               --> IENERG
10    2       8  tkw1                 --> IWAVE,MSTAT,FINP2
11    1     253     256       4       --> ISAVA,MSAVA1,MSAVA2,NTIMES
12    1      10                       --> ISAVB,NNODB
13    0       0                       --> ISAVC,NNODC
14    00000       0                   --> IREQ,IELEV,IV,IDUDT,ISNR,NREQ
15         400                        --> INITS
16           1                        --> MULTIF
17            .001000                 --> DELTA (normalized)
18         256.000000                 --> TMAX  (normalized)
19          32                        --> NRATE
20           2                        --> NDELR
21          5.000000                  --> DELRP(1)(millimeters)
22         30.000000                  -->       (2)
23          2.000000     2.000000     --> X1,X2
24          2.060000     7.000000     --> HREFP(meters),TP(seconds)
25          1.000000     1.000000     --> KSREF,KSSEA
26           .050000                  --> FWP
27          4.15                      --> DSEAP(meters)
28          0.118000                  --> TSLOPS
29           30                       --> NBSEG
30          0.00         4.15         --> XBSEG( 1)(m),ZBSEG( 1)(m)
31          6.67         3.94         -->       ( 2)          ( 2)
32         13.86         3.64         -->       ( 3)          ( 3)
33         20.55         3.39         -->       ( 4)          ( 4)
34         28.03         2.99         -->       ( 5)          ( 5)
35         36.30         2.55         -->       ( 6)          ( 6)
36         47.54         2.15         -->       ( 7)          ( 7)
37         55.49         2.05         -->       ( 8)          ( 8)
38         61.30         2.03         -->       ( 9)          ( 9)
39         66.98         2.14         -->       (10)          (10)
40         72.47         2.27         -->       (11)          (11)
41         78.35         2.50         -->       (12)          (12)
42         84.03         2.73         -->       (13)          (13)
43         90.62         2.90         -->       (14)          (14)
44         97.77         3.01         -->       (15)          (15)
45        105.28         3.11         -->       (16)          (16)
46        109.04         3.09         -->       (17)          (17)
47        112.41         2.90         -->       (18)          (18)
48        115.19         2.67         -->       (19)          (19)
49        117.78         2.42         -->       (20)          (20)
50        120.00         2.15         -->       (21)          (21)
51        122.26         1.80         -->       (22)          (22)
52        123.33         1.54         -->       (23)          (23)
53        125.68         1.31         -->       (24)          (24)
54        126.90         1.02         -->       (25)          (25)
55        128.78         0.80         -->       (26)          (26)
56        132.54         0.29         -->       (27)          (27)
```

Table F-1 continued

57	136.30		-0.10		-->	(28)	(28)
58	143.99		-0.85		-->	(29)	(29)
59	147.18		-1.29		-->	(30)	(30)
60	148.90		-1.57		-->	(31)	(31)
61	40	80	120	160	200	-->	NODB(1)-(2)-(3)-(4)-(5)
62	240	280	320	360	400	-->	(6)-(7)-(8)-(9)-(10)

Table F-2: File ODOC produced by RBREAK computation of "Kobayashi & Wurjanto 1991 Run 1".

```
 1  ------------------------------------------------------------------------
 2  Kobayashi & Wurjanto 1991 Run 1
 3  ------------------------------------------------------------------------
 4
 5  WAVE CONDITION
 6
 7  Incident Wave at Seaward Boundary Given as Input
 8  Number of Data Points    NDATA =    8193
 9  Norm. Maximum Surface Elev.    =       0.811980
10  Norm. Minimum Surface Elev.    =      -0.943159
11
12  Reference Wave Period          =       7.000000 sec.
13  Reference Wave Height          =       2.060000 meters
14  Depth at Seaward Boundary      =       4.150000 meters
15  Shoal. Coef. at Reference  Ks1 =       1.000
16               at Seaw. Bdr. Ks2 =       1.000
17               Ks = Ks2/Ks1      =       1.000
18  Norm. Depth at Seaw. Boundary  =       2.015
19  Normalized Wave Length         =      10.149
20  "Sigma"                        =      15.276
21  Ursell Number                  =      51.124
22  Surf Similarity Parameter      =       0.719
23
24  SLOPE PROPERTIES
25
26  Friction Factor            =       0.050000
27  Norm. Friction Factor      =       0.381891
28  Norm. Horiz. Length of
29        Computation Domain   =       4.730854
30  Number of Segments         =      30
```

Table F-2 continued

SEGMENT I	XBSEG(I) meters	ZBSEG(I) meters
1	0.000000	4.150000
2	6.670000	3.940000
3	13.860000	3.640000
4	20.550000	3.390000
5	28.030000	2.990000
6	36.300000	2.550000
7	47.540000	2.150000
8	55.490000	2.050000
9	61.300000	2.030000
10	66.980000	2.140000
11	72.470000	2.270000
12	78.350000	2.500000
13	84.030000	2.730000
14	90.620000	2.900000
15	97.770000	3.010000
16	105.280000	3.110000
17	109.040000	3.090000
18	112.410000	2.900000
19	115.190000	2.670000
20	117.780000	2.420000
21	120.000000	2.150000
22	122.260000	1.800000
23	123.330000	1.540000
24	125.680000	1.310000
25	126.900000	1.020000
26	128.780000	0.800000
27	132.540000	0.290000
28	136.300000	-0.100000
29	143.990000	-0.850000
30	147.180000	-1.290000
31	148.900000	-1.570000

COMPUTATION PARAMETERS

Total Number of Spatial Nodes JE = 441
Number of Nodes Along Bottom Below SWL
INITS = 400
Normalized Delta x = 0.107519D-01
Damping Coefficients x1 = 2.000
x2 = 2.000
Computation Duration TMAX = 256.000000 Wave Periods
Minimum Allowable NONE NONEM = 576
Time Series Stored at Rate NRATE
per Wave Period with NRATE = 32
Statistical Calculations Are Performed Excluding
the First MSTAT Computation Units
with MSTAT = 8

Table F-2 continued

```
 83     Spatial Variations are Stored NTIMES at Equal
 84          Intervals per Wave Period from
 85          Computation Unit MSAVA1 to MSAVA2,
 86          Inclusive, with              NTIMES =        4
 87                                       MSAVA1 =      253
 88                                       MSAVA2 =      256
 89
 90     REFLECTION COEFFICIENT
 91
 92     r1 =     0.154
 93
 94     RUNUP, RUNDOWN, SETUP
 95
 96     Largest Node Number Reached by Computational Waterline
 97                                                  JMAX =      436
 98     ------------------------------------------------------------
 99           I   DELTAR(I)     RUNUP(I)    RUNDOWN(I)    SETUP(I)
100                  [mm]          R            Rd           Zr
101     ------------------------------------------------------------
102           1     5.000        0.628       -0.100        0.194
103           2    30.000        0.618       -0.165        0.146
104     ------------------------------------------------------------
105
106     WAVE SET-DOWN OR SETUP
107
108     Average value of ETAI =     0.000448
109                       ETAR =   -0.038266
110
111     TIME-AVERAGED ENERGY BALANCE
112
113     Normalized Energy Flux:
114     . at Seaw. Boundary   A =  0.882722D-01
115     . at Landw. Boundary  B =  0.000000D+00
116     Normalized Rate of Energy Dissipation
117     in the Computation Domain, Due to:
118     . Bottom Friction     C =  0.789698D-02
119     . Wave Breaking       D =  0.806223D-01
120     Additional Term       E = -0.247007D-03
121     Calculation 1:
122                     F = A-B =  0.882722D-01
123                   G = C+D+E =  0.882722D-01
124       Must H=0, but H = G-F = -0.584577D-09
125       % error  100G/E =          0.00
```

Table F-3: File BINPUT produced by BEFORR computation of "Kobayashi & Wurjanto 1991 Run 1".

```
 1          3
 2    ------------------------------------------------------------------------
 3    Kobayashi & Wurjanto 1991 Run 1
 4    ------------------------------------------------------------------------
 5         256
 6    tkw1
 7        8193
 8         576
 9           1     576     576   0.100000000D-02
10           2     576     576   0.100000000D-02
11           3     576     576   0.100000000D-02
12           4     576     576   0.100000000D-02
13           5     576     576   0.100000000D-02
14           6     576     576   0.100000000D-02
15           7     576     576   0.100000000D-02
16           8     576     576   0.100000000D-02
17           9     576     576   0.100000000D-02
18          10     576     576   0.100000000D-02
19          11     576     576   0.100000000D-02
20          12     576     576   0.100000000D-02
21          13     576     576   0.100000000D-02
22          14     576     576   0.100000000D-02
23          15     576     576   0.100000000D-02
24          16     576     576   0.100000000D-02
25          17     576     576   0.100000000D-02
26          18     576     576   0.939000000D-03
27          19     576     576   0.939000000D-03
28          20     576     576   0.939000000D-03
29          21     576     576   0.939000000D-03
30          22     576     576   0.939000000D-03
31          23     576     576   0.939000000D-03
32          24     576     576   0.939000000D-03
33          25     576     576   0.939000000D-03
34          26     576     576   0.939000000D-03
35          27     576     576   0.939000000D-03
36          28     576     576   0.939000000D-03
37          29     576     576   0.939000000D-03
38          30     576     576   0.939000000D-03
39          31     576     576   0.939000000D-03
40          32     576     576   0.939000000D-03
41          33     576     576   0.939000000D-03
42          34     576     576   0.939000000D-03
43          35     576     576   0.939000000D-03
44          36     576     576   0.939000000D-03
45          37     576     576   0.939000000D-03
46          38     576     576   0.939000000D-03
47          39     576     576   0.939000000D-03
48          40     576     576   0.939000000D-03
49          41     576     576   0.939000000D-03
50          42     576     576   0.939000000D-03
51          43     576     576   0.939000000D-03
52          44     576     576   0.939000000D-03
53          45     576     576   0.939000000D-03
54          46     576     576   0.939000000D-03
55          47     576     576   0.939000000D-03
56          48    1280    1280   0.939000000D-03
```

Table F-3 continued

57	49	576	576	0.939000000D-03
58	50	576	576	0.939000000D-03
59	51	576	576	0.939000000D-03
60	52	576	576	0.939000000D-03
61	53	576	576	0.939000000D-03
62	54	576	576	0.939000000D-03
63	55	576	576	0.939000000D-03
64	56	576	576	0.883000000D-03
65	57	576	576	0.883000000D-03
66	58	576	576	0.883000000D-03
67	59	576	576	0.883000000D-03
68	60	576	576	0.883000000D-03
69	61	576	576	0.883000000D-03
70	62	576	576	0.883000000D-03
71	63	576	576	0.883000000D-03
72	64	576	576	0.883000000D-03
73	65	576	576	0.883000000D-03
74	66	576	576	0.883000000D-03
75	67	576	576	0.883000000D-03
76	68	576	576	0.883000000D-03
77	69	576	576	0.883000000D-03
78	70	576	576	0.883000000D-03
79	71	576	576	0.883000000D-03
80	72	576	576	0.883000000D-03
81	73	576	576	0.883000000D-03
82	74	576	576	0.883000000D-03
83	75	576	576	0.883000000D-03
84	76	576	576	0.883000000D-03
85	77	576	576	0.883000000D-03
86	78	576	576	0.883000000D-03
87	79	576	576	0.883000000D-03
88	80	576	576	0.883000000D-03
89	81	576	576	0.883000000D-03
90	82	576	576	0.883000000D-03
91	83	576	576	0.883000000D-03
92	84	576	576	0.883000000D-03
93	85	576	576	0.883000000D-03
94	86	576	576	0.883000000D-03
95	87	576	576	0.883000000D-03
96	88	576	576	0.835000000D-03
97	89	576	576	0.835000000D-03
98	90	576	576	0.835000000D-03
99	91	576	576	0.835000000D-03
100	92	576	576	0.835000000D-03
101	93	576	576	0.835000000D-03
102	94	576	576	0.835000000D-03
103	95	576	576	0.835000000D-03
104	96	576	576	0.835000000D-03
105	97	576	576	0.835000000D-03
106	98	576	576	0.835000000D-03
107	99	576	576	0.835000000D-03
108	100	576	576	0.835000000D-03
109	101	576	576	0.835000000D-03
110	102	576	576	0.835000000D-03
111	103	576	576	0.835000000D-03
112	104	576	576	0.835000000D-03

Table F-3 continued

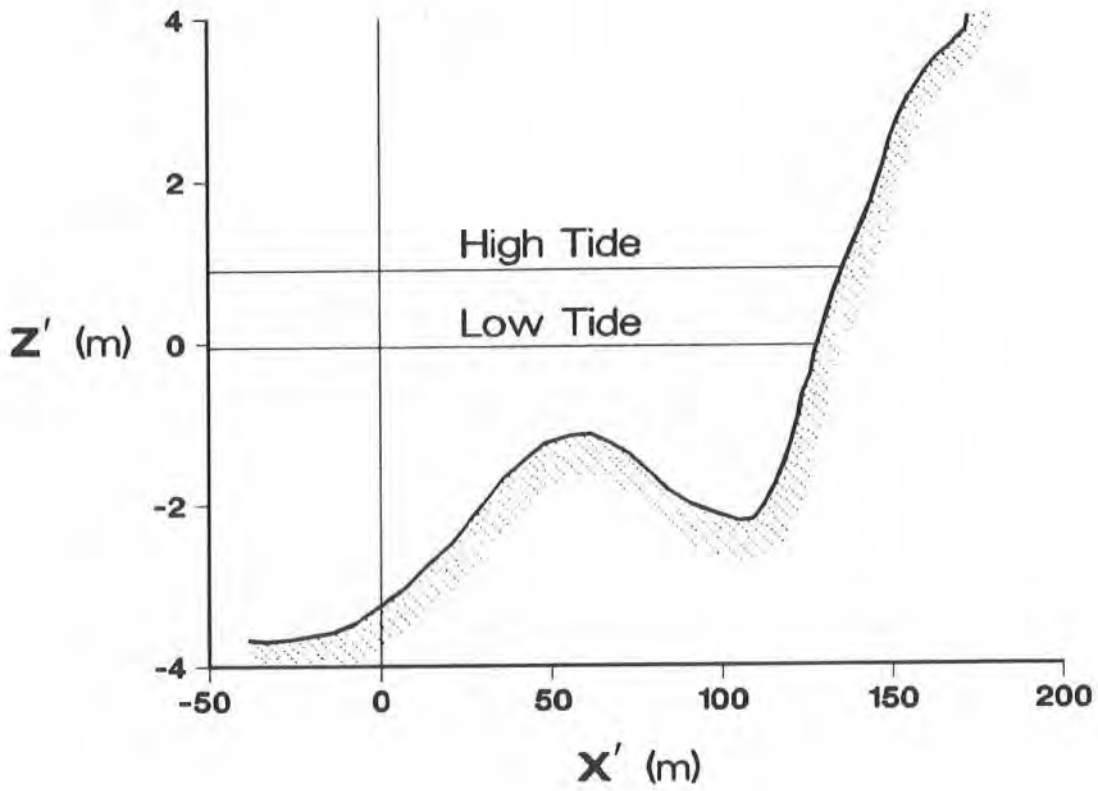
113	105	576	576	0.835000000D-03
114	106	576	576	0.835000000D-03
115	107	576	576	0.777000000D-03
116	108	576	576	0.777000000D-03
117	109	576	576	0.777000000D-03
118	110	576	576	0.777000000D-03
119	111	576	576	0.777000000D-03
120	112	576	576	0.777000000D-03
121	113	576	576	0.777000000D-03
122	114	576	576	0.777000000D-03
123	115	576	576	0.777000000D-03
124	116	576	576	0.777000000D-03
125	117	576	576	0.777000000D-03
126	118	576	576	0.777000000D-03
127	119	576	576	0.777000000D-03
128	120	1280	1280	0.777000000D-03
129	121	576	576	0.777000000D-03
130	122	576	576	0.777000000D-03
131	123	576	576	0.777000000D-03
132	124	576	576	0.777000000D-03
133	125	864	864	0.777000000D-03
134	126	576	576	0.777000000D-03
135	127	576	576	0.777000000D-03
136	128	864	864	0.777000000D-03
137	129	576	576	0.777000000D-03
138	130	576	576	0.777000000D-03
139	131	576	576	0.777000000D-03
140	132	576	576	0.777000000D-03
141	133	576	576	0.777000000D-03
142	134	576	576	0.777000000D-03
143	135	576	576	0.777000000D-03
144	136	576	576	0.777000000D-03
145	137	576	576	0.777000000D-03
146	138	576	576	0.777000000D-03
147	139	576	576	0.777000000D-03
148	140	576	576	0.777000000D-03
149	141	576	576	0.777000000D-03
150	142	576	576	0.777000000D-03
151	143	576	576	0.777000000D-03
152	144	576	576	0.777000000D-03
153	145	576	576	0.777000000D-03
154	146	576	576	0.777000000D-03
155	147	576	576	0.777000000D-03
156	148	576	576	0.777000000D-03
157	149	576	576	0.777000000D-03
158	150	576	576	0.777000000D-03
159	151	576	576	0.777000000D-03
160	152	576	576	0.777000000D-03
161	153	576	576	0.777000000D-03
162	154	576	576	0.777000000D-03
163	155	576	576	0.777000000D-03
164	156	576	576	0.777000000D-03
165	157	576	576	0.777000000D-03
166	158	576	576	0.777000000D-03
167	159	576	576	0.777000000D-03
168	160	576	576	0.777000000D-03

Table F-3 continued

169	161	576	576	0.777000000D-03
170	162	576	576	0.777000000D-03
171	163	576	576	0.777000000D-03
172	164	576	576	0.777000000D-03
173	165	576	576	0.777000000D-03
174	166	864	864	0.777000000D-03
175	167	576	576	0.777000000D-03
176	168	576	576	0.777000000D-03
177	169	576	576	0.777000000D-03
178	170	576	576	0.777000000D-03
179	171	576	576	0.777000000D-03
180	172	576	576	0.777000000D-03
181	173	576	576	0.777000000D-03
182	174	576	576	0.777000000D-03
183	175	576	576	0.777000000D-03
184	176	576	576	0.777000000D-03
185	177	576	576	0.777000000D-03
186	178	576	576	0.777000000D-03
187	179	576	576	0.777000000D-03
188	180	576	576	0.777000000D-03
189	181	576	576	0.777000000D-03
190	182	576	576	0.777000000D-03
191	183	576	576	0.777000000D-03
192	184	576	576	0.777000000D-03
193	185	576	576	0.777000000D-03
194	186	576	576	0.777000000D-03
195	187	576	576	0.777000000D-03
196	188	576	576	0.777000000D-03
197	189	576	576	0.777000000D-03
198	190	576	576	0.777000000D-03
199	191	576	576	0.777000000D-03
200	192	576	576	0.777000000D-03
201	193	576	576	0.777000000D-03
202	194	576	576	0.777000000D-03
203	195	576	576	0.777000000D-03
204	196	576	576	0.777000000D-03
205	197	576	576	0.777000000D-03
206	198	576	576	0.777000000D-03
207	199	576	576	0.777000000D-03
208	200	576	576	0.777000000D-03
209	201	576	576	0.777000000D-03
210	202	576	576	0.777000000D-03
211	203	576	576	0.777000000D-03
212	204	1280	1280	0.777000000D-03
213	205	576	576	0.777000000D-03
214	206	576	576	0.777000000D-03
215	207	576	576	0.777000000D-03
216	208	576	576	0.777000000D-03
217	209	576	576	0.777000000D-03
218	210	576	576	0.777000000D-03
219	211	864	864	0.777000000D-03
220	212	576	576	0.777000000D-03
221	213	576	576	0.777000000D-03
222	214	576	576	0.777000000D-03
223	215	576	576	0.777000000D-03
224	216	576	576	0.777000000D-03

Table F-3 continued

225	217	576	576	0.777000000D-03
226	218	864	864	0.777000000D-03
227	219	576	576	0.777000000D-03
228	220	576	576	0.777000000D-03
229	221	576	576	0.777000000D-03
230	222	576	576	0.777000000D-03
231	223	576	576	0.777000000D-03
232	224	576	576	0.777000000D-03
233	225	576	576	0.777000000D-03
234	226	576	576	0.777000000D-03
235	227	576	576	0.777000000D-03
236	228	576	576	0.777000000D-03
237	229	576	576	0.777000000D-03
238	230	576	576	0.777000000D-03
239	231	576	576	0.777000000D-03
240	232	576	576	0.777000000D-03
241	233	576	576	0.777000000D-03
242	234	576	576	0.777000000D-03
243	235	576	576	0.777000000D-03
244	236	576	576	0.777000000D-03
245	237	576	576	0.777000000D-03
246	238	576	576	0.777000000D-03
247	239	576	576	0.777000000D-03
248	240	576	576	0.777000000D-03
249	241	576	576	0.777000000D-03
250	242	576	576	0.777000000D-03
251	243	576	576	0.777000000D-03
252	244	576	576	0.777000000D-03
253	245	576	576	0.777000000D-03
254	246	576	576	0.777000000D-03
255	247	576	576	0.777000000D-03
256	248	576	576	0.777000000D-03
257	249	576	576	0.777000000D-03
258	250	576	576	0.777000000D-03
259	251	576	576	0.777000000D-03
260	252	576	576	0.777000000D-03
261	253	576	576	0.777000000D-03
262	254	576	576	0.777000000D-03
263	255	576	576	0.777000000D-03
264	256	576	576	0.777000000D-03



Kobayashi and Wurjanto (1991) Run 1

Figure F-1

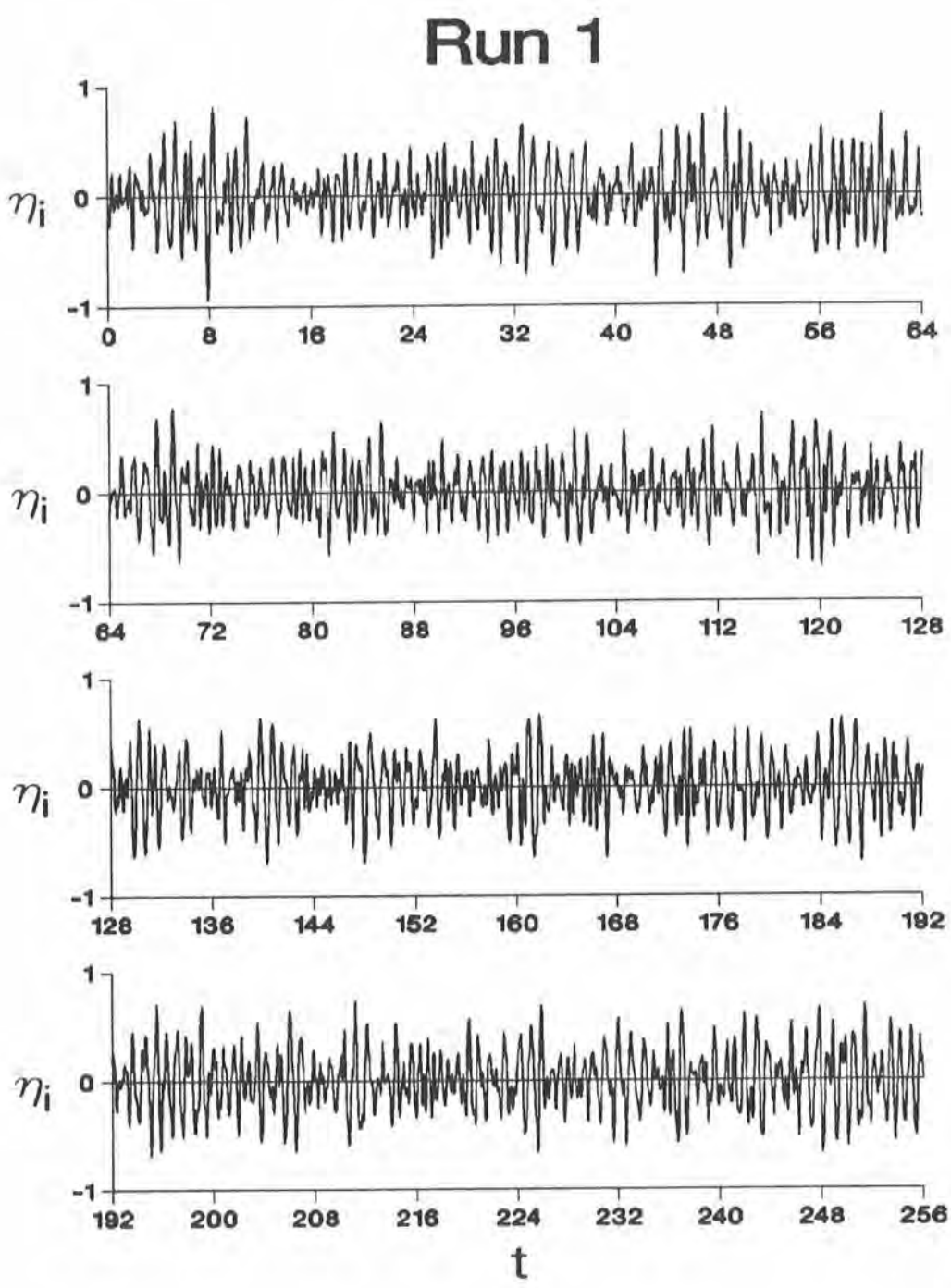

Run 1
η_i
1
0
-1
0
8
16
24
32
40
48
56
64
72
80
88
96
104
112
120
128
136
144
152
160
168
176
184
192
200
208
216
224
232
240
248
256
t

Figure F-2

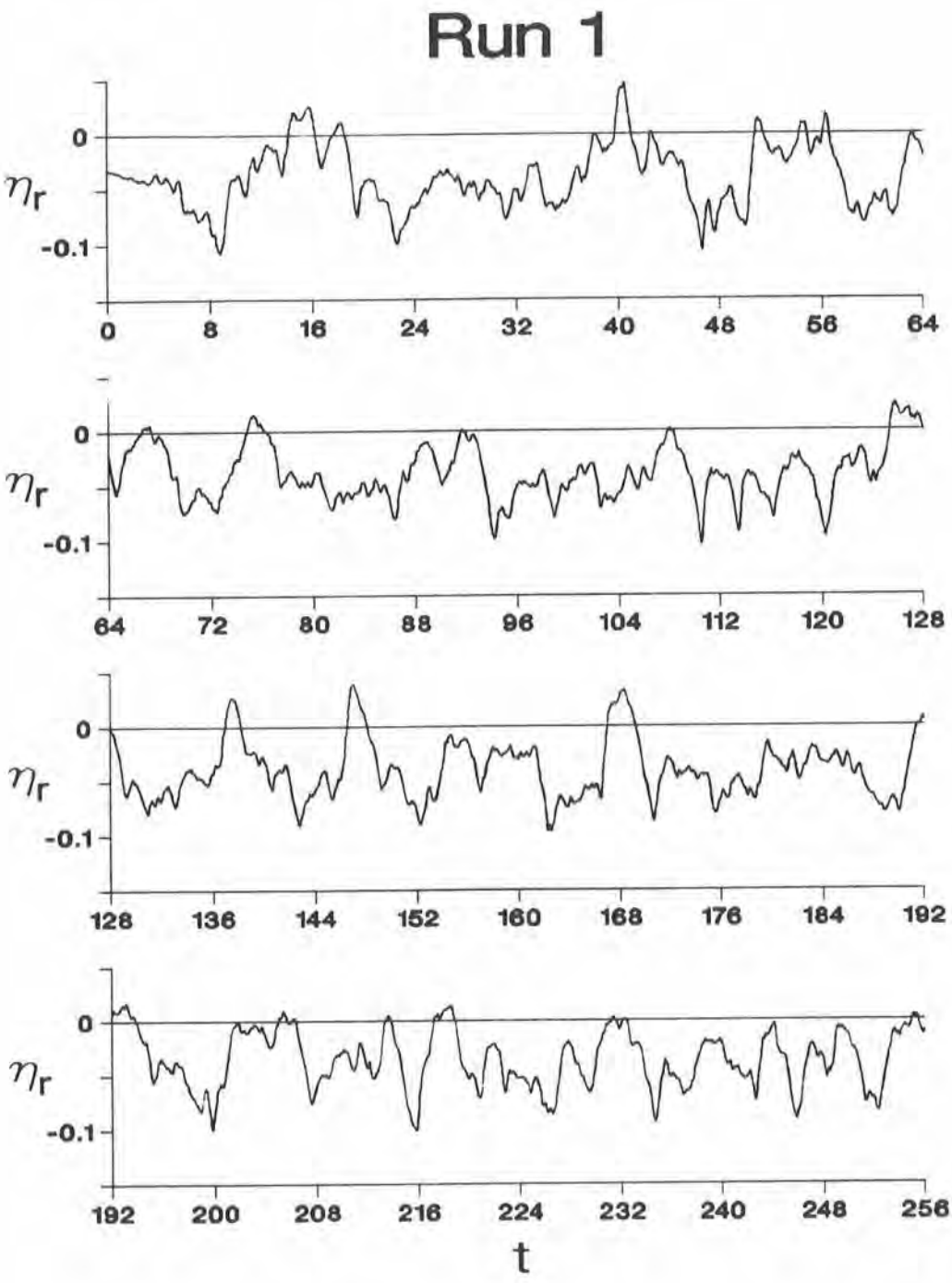

Run 1
η_r
0
-0.1
0 8 16 24 32 40 48 56 64
64 72 80 88 96 104 112 120 128
128 136 144 152 160 168 176 184 192
192 200 208 216 224 232 240 248 256
t

Figure F-3

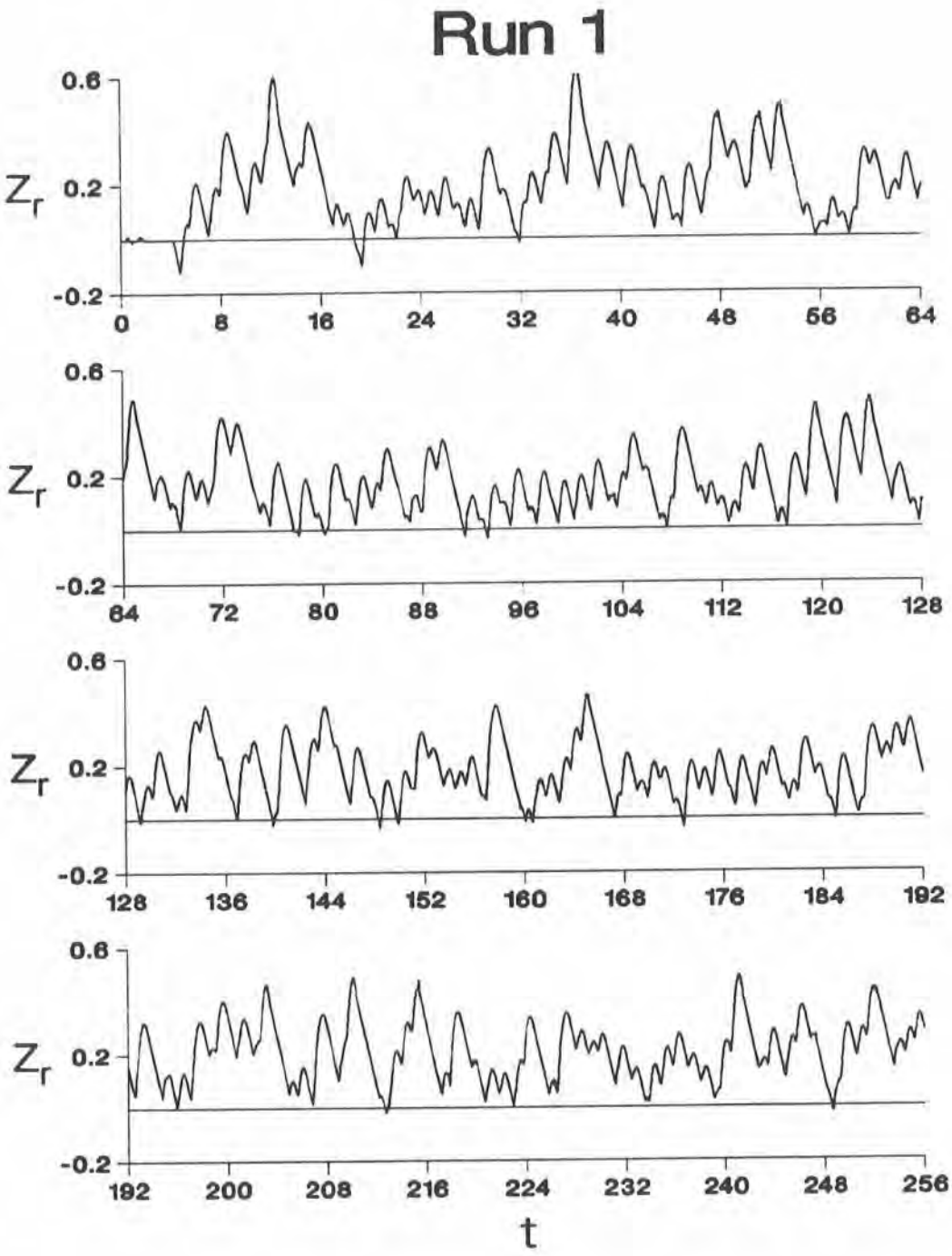

Run 1
0.6
0.2
-0.2
Z_r
0
8
16
24
32
40
48
56
64
64
72
80
88
96
104
112
120
128
128
136
144
152
160
168
176
184
192
192
200
208
216
224
232
240
248
256
t

Figure F-4

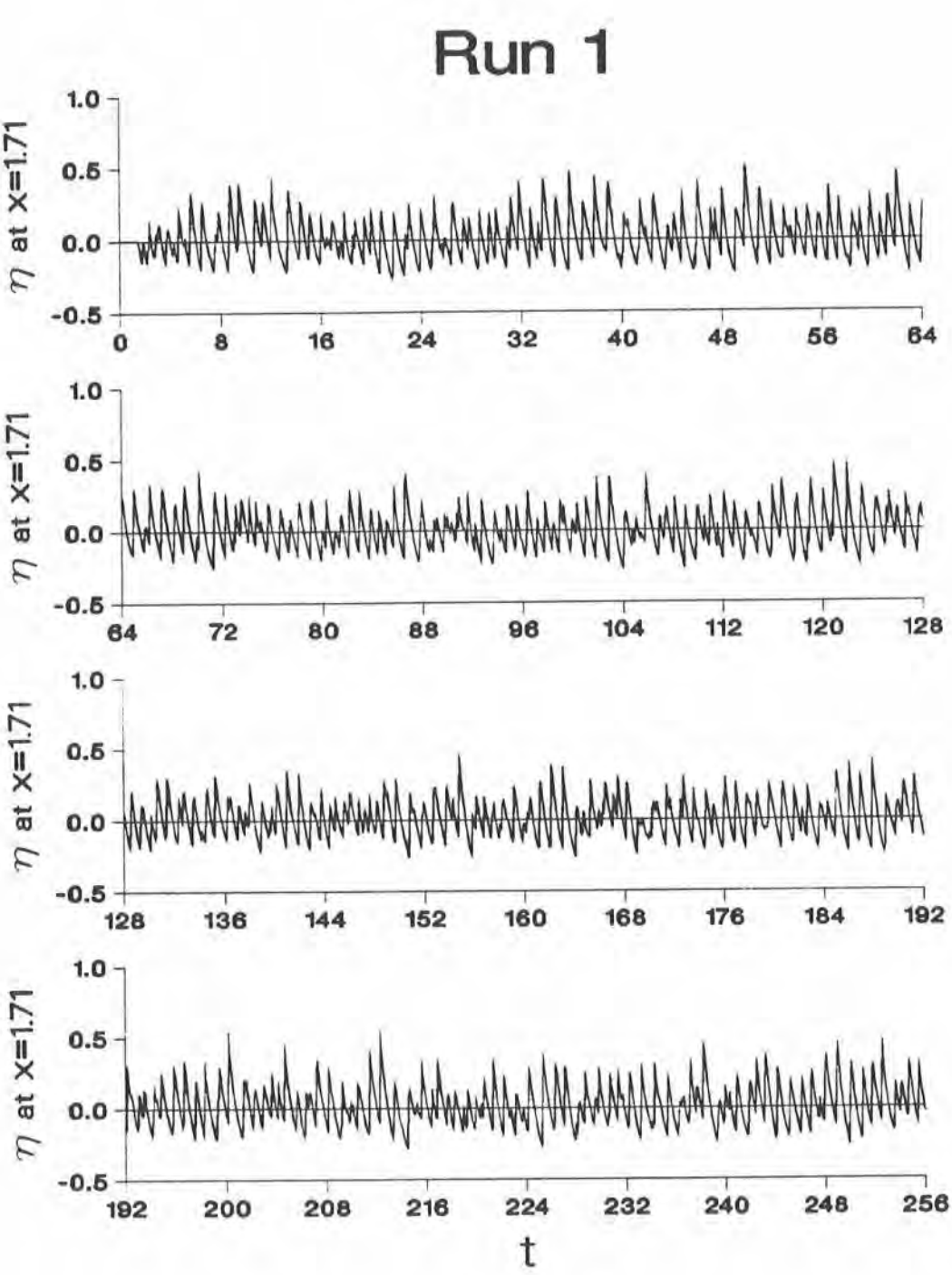

Run 1
η at X=1.71
1.0
0.5
0.0
-0.5
0
8
16
24
32
40
48
56
64
64
72
80
88
96
104
112
120
128
128
136
144
152
160
168
176
184
192
192
200
208
216
224
232
240
248
256
t

Figure F-5

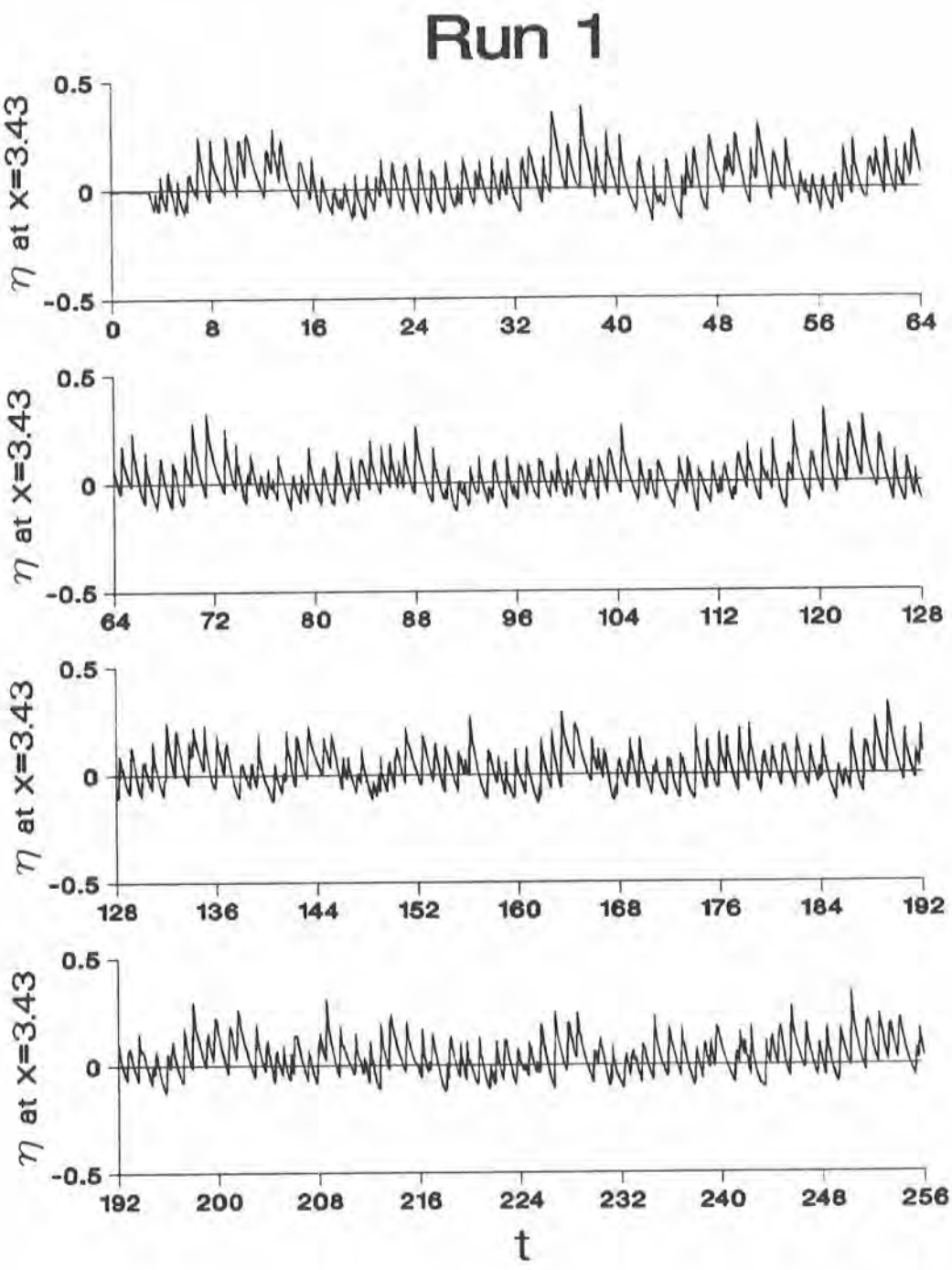

Run 1
η at X=3.43
0.5
0
-0.5
0
8
16
24
32
40
48
56
64
64
72
80
88
96
104
112
120
128
128
136
144
152
160
168
176
184
192
192
200
208
216
224
232
240
248
256
t

Figure F-6

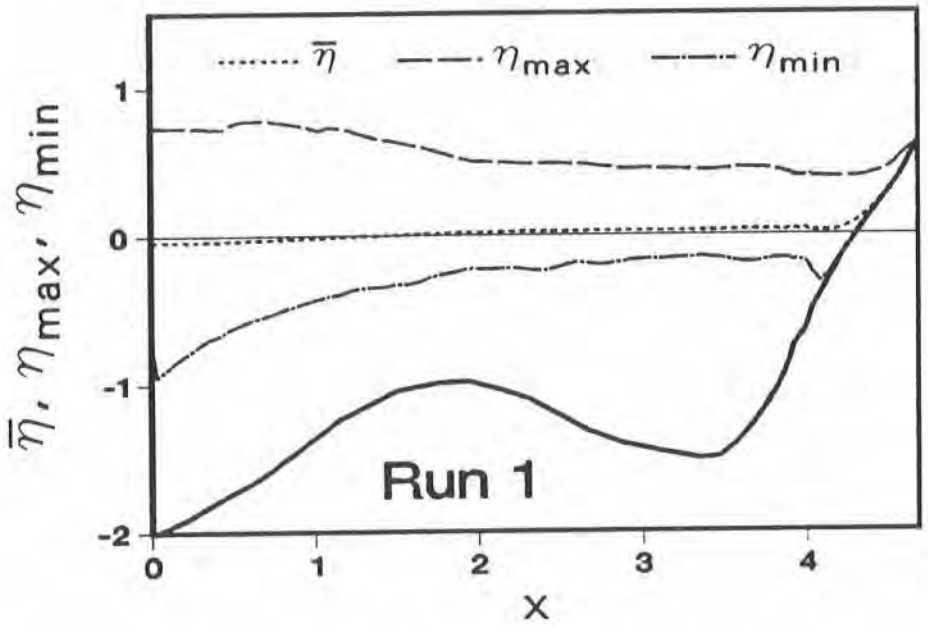


Figure F-7a

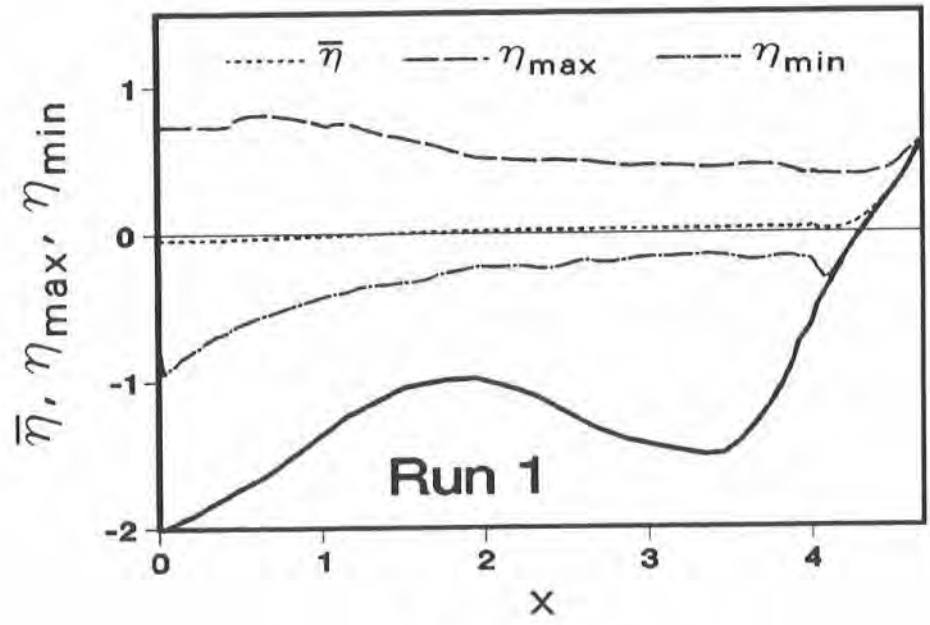


Figure F-7b

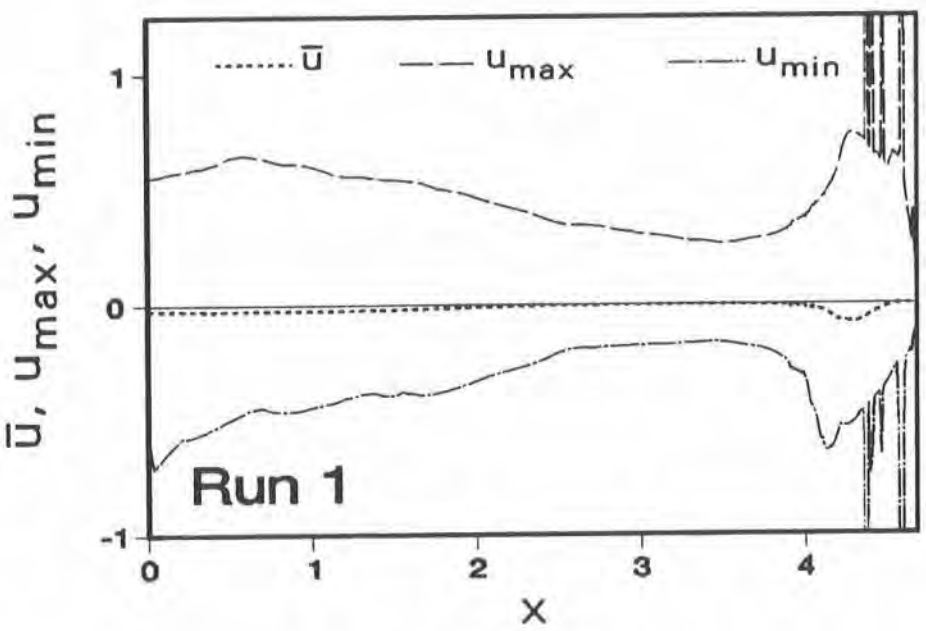


Figure F-8a

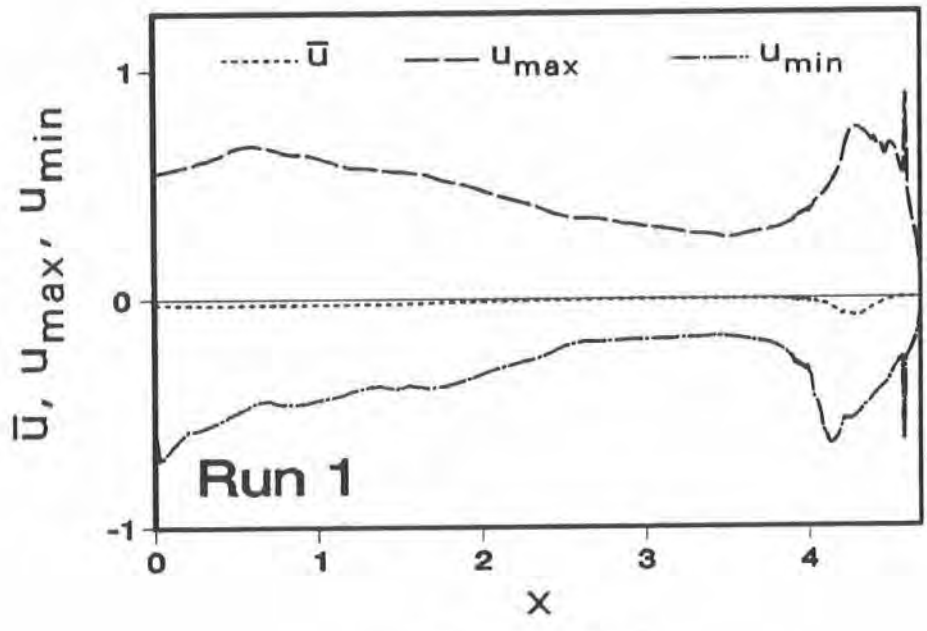


Figure F-8b

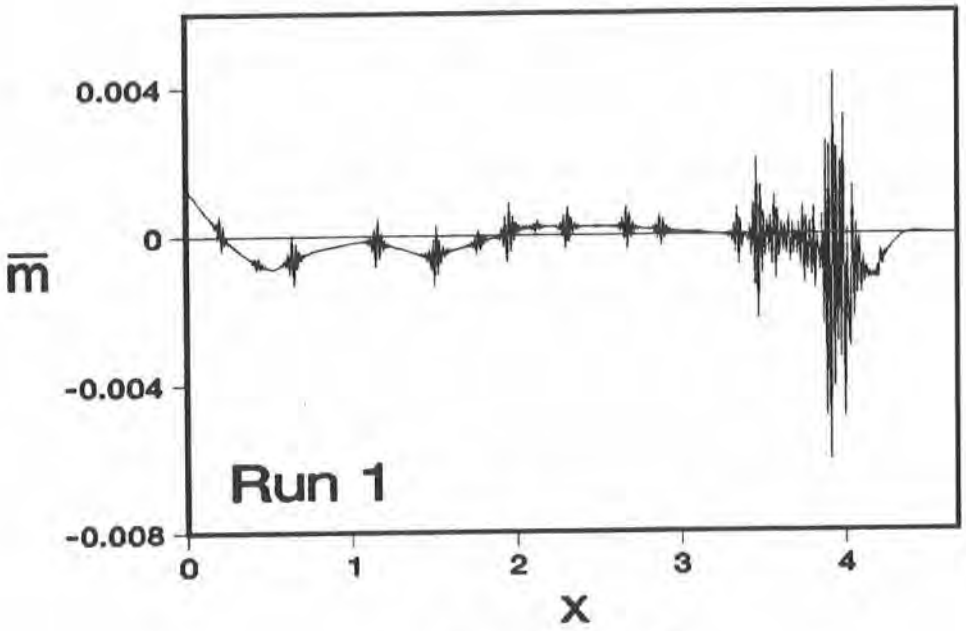


Figure F-9a

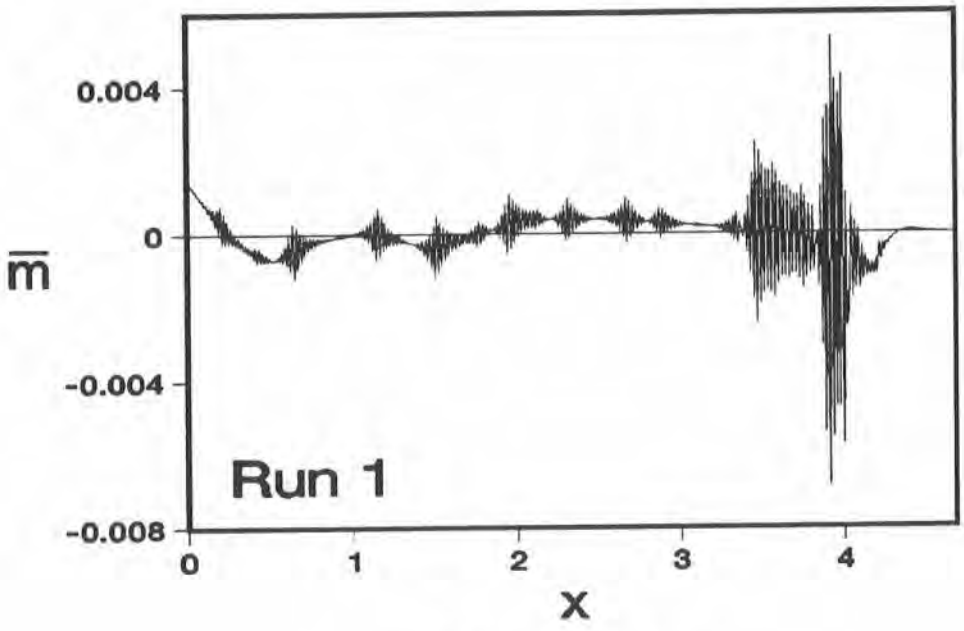


Figure F-9b

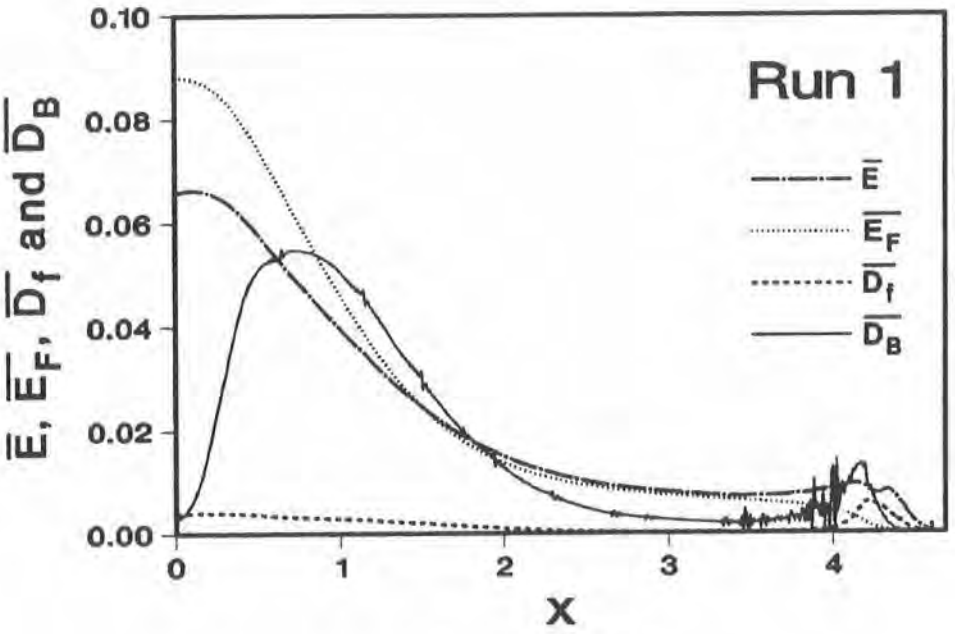


Figure F-10a

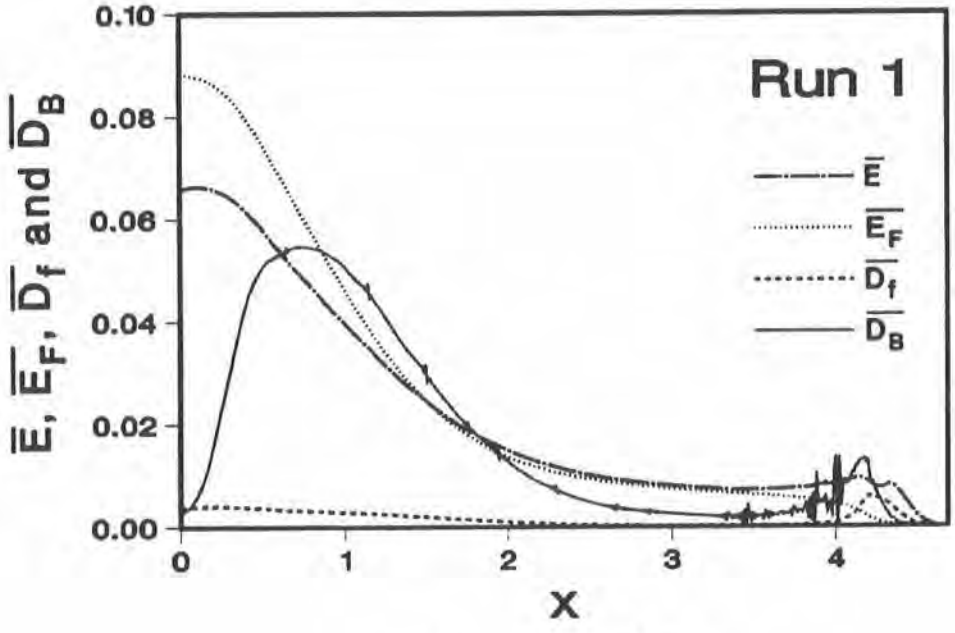


Figure F-10b

Run 1

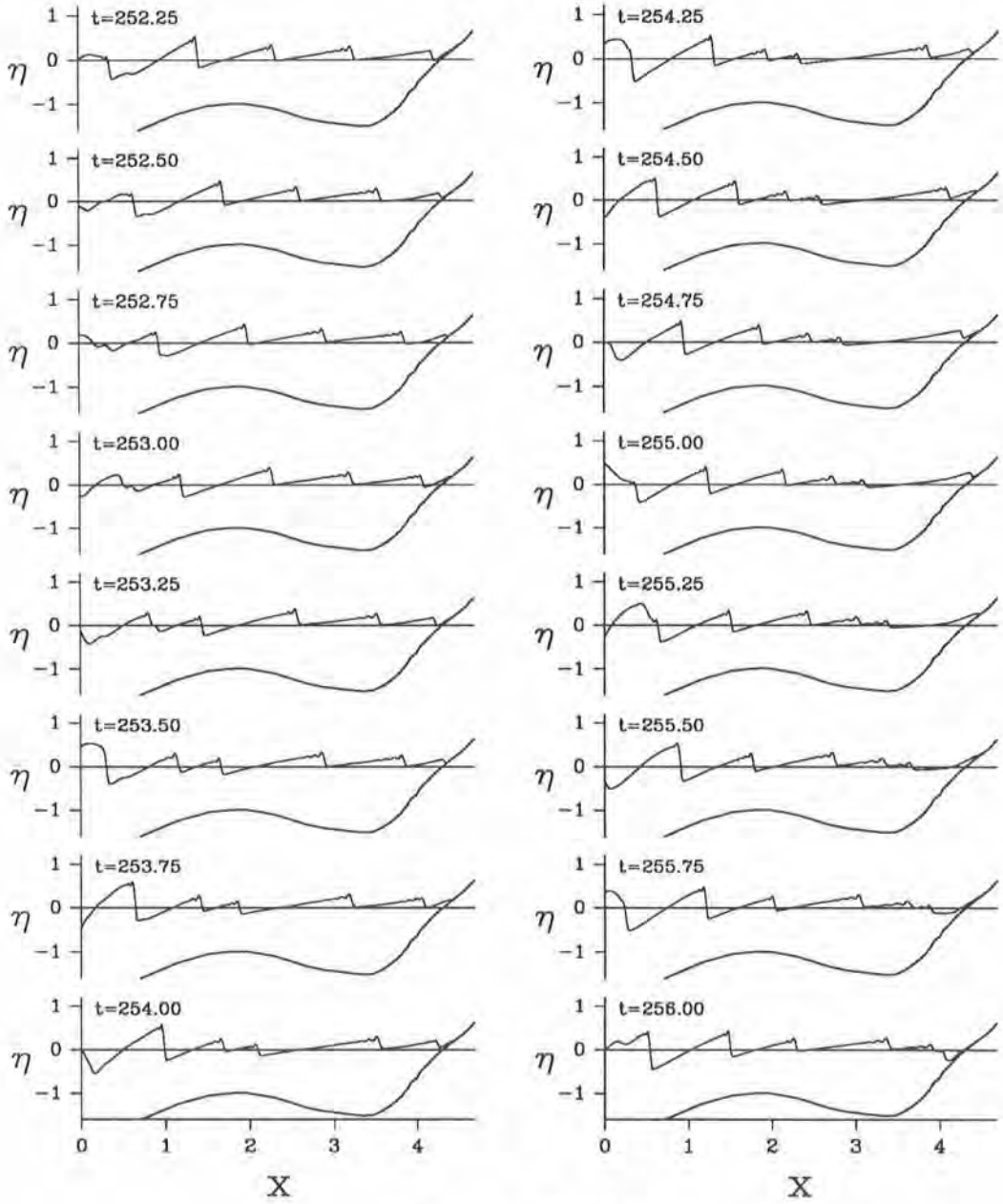


Figure F-11

Run 1

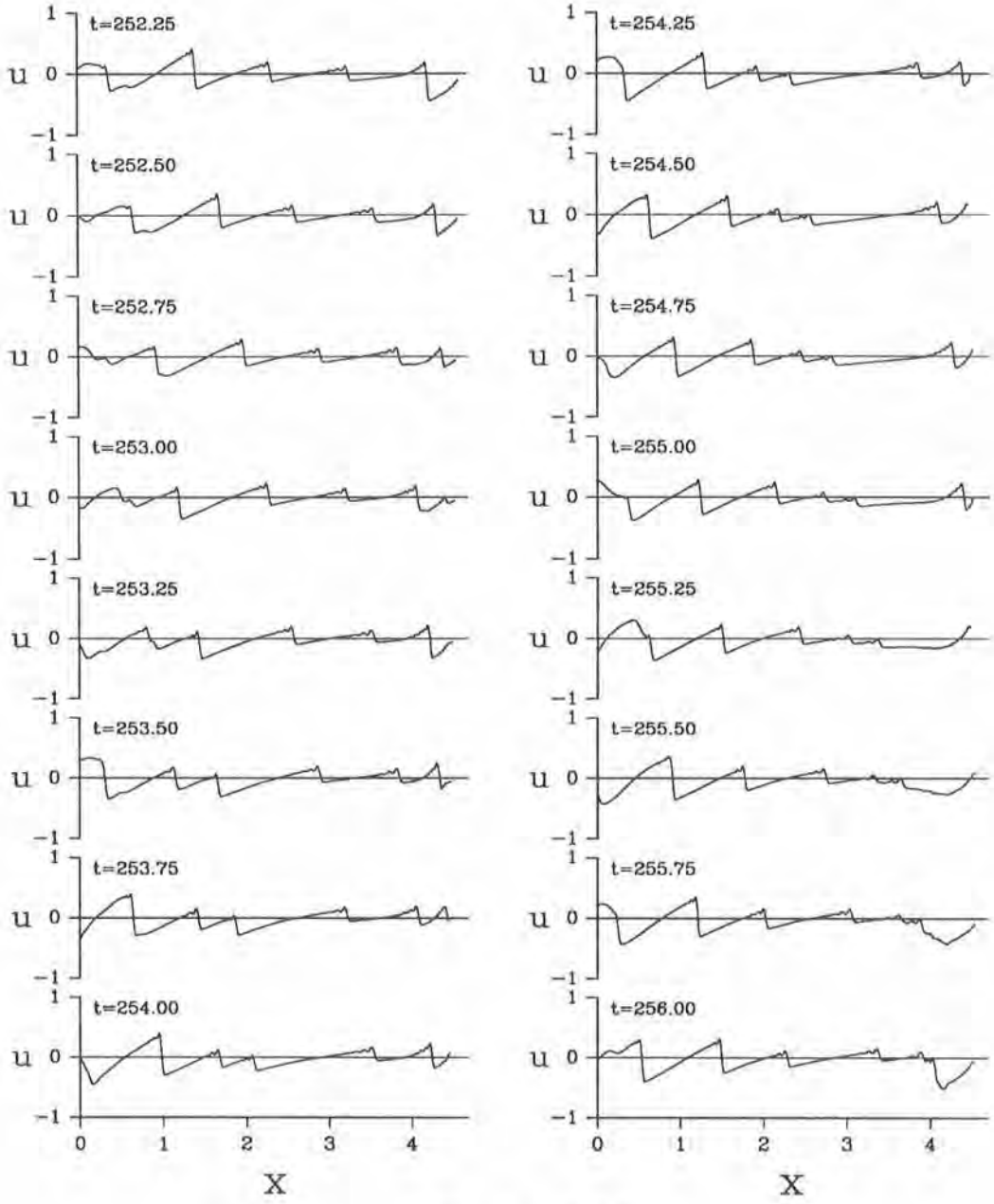


Figure F-12

APPENDIX G

EXAMPLE: INPUT WAVE TRAIN FROM USER-SPECIFIED SPECTRUM

This example is Run SB in Kobayashi and Wurjanto (1991). This run features the use of an input wave train that is generated from an observed wave spectrum. Run SB is intended to simulate the Santa Barbara, California, "Feb 4, 80" field data reported by Elgar and Guza (1985a,b). Figure G-1 depicts the simplified bottom geometry used for the numerical simulation. The seaward boundary for this computation is taken to be at the water depth 1.7m $\Longrightarrow$ DSEAP=1.7m (under ISYST=1, International System of Units) where the spectral estimate of the significant wave height is 0.9m and the spectral peak period is 13.5s. These two quantities are taken as the reference wave height and period, respectively $\Longrightarrow$ HREFP=0.9m, TP=13.5s, and KSREF=KSSEA=1.

Kobayashi and Wurjanto (1991) specifies an *incident* wave spectrum such that the computed *total* wave spectrum at the seaward boundary matches with the reported wave spectrum at the depth 1.7m. The incident wave train generated from the assumed incident wave spectrum is partially plotted in Figure G-2 for the first 160 wave periods out of the total 590 wave periods (TMAX=590). The duration of the computation was very long to provide a large number of degrees of freedom for the spectral analysis presented in their paper.

The other input for the BEFORR-RBREAK computation of Run SB are listed below.

- NBSEG=1, WBSEG(1)=68m, TBSLOP(1)=0.05. A single linear slope segment is used to specify the simplified bottom geometry.
- TSLOPS=TBSLOP(1)=0.05 because the slope is uniform.
- IJOB=1 for run-up computation.
- ISTAB=0 because of no sediment analysis.
- INONCT=1 for the gentle slope.
- IENERG=1 for the computation of energy quantities.
- FINP2=tkwsb specifies the name of the file containing the input wave train for this IBM 3090 run.
- MSTAT=30. The first 30 wave periods are excluded from the statistical calculations.
- ISAVA=1, MSAVA1=587, MSAVA2=590, NTIMES=4. The spatial variations of η and u are stored at the normalized time t = 586.25, 586.50, 586.75, ..., 590.

- ISAVB=1, NNODB=4. The time series of m=U(1,j_B) and h=U(2,j_B) at the four specified nodes j_B = NODB = 1, 61, 121, and 181 are stored during the computation.
- ISAVC=0 because no sediment analysis is performed.
- IREQ=0. IELEV=IV=IDUDT=ISNR=NREQ=0 for no special request.
- NDELR=2. Two water depths are specified: DELRP(1)=5mm and DELRP(2)=30mm.
- NRATE=27. The rate of storing the computed time series is 27 data points per wave period. This number is selected to get the "sampling" interval of 0.5s, which is identical to the field observation.
- INITS=200. The resulting grid spacing between two adjacent nodes is 0.17m, which is reasonably small compared to the characteristic wave height at the seaward boundary, H'=0.9m, and the linear wavelength based on the characteristic wave period at the seaward boundary, L'=54.79m.
- X1=X2=2.
- DELTA=0.001 as input.
- MULTIF=1.

The tables and figures in this Appendix are arranged in the same order as for Run 1 in Appendix F so that the tables and figures in this Appendix may become self-explanatory with the following brief explanations:

- The primary input data the file = the file `appeng.inp` on the disk.
- The input wave train is stored in the file `appeng.iwt` on the disk.
- The file BINPUT = the file `appeng.bin` on the disk.
- The file ODOC = the file `appeng.doc` on the disk.
- The plotted time series in this Appendix cover only the first 160 wave periods out of the total duration TMAX=590.
- The plotted time series Z_r in Figure G-4 corresponds to the physical water depth δ_r'=30mm [=DELRP(2)].
- Figures G-5 and G-6 show the time series of the free surface elevation η at the locations x=0.25 (node j_B=61) and x=0.76 (node j_B=181), respectively. The data points for these two figures are obtained from the files OSAV2B01, OSAV2B02, ..., OSAV2B06 that contain the first 100, the second 100, ..., the fifth 100, and the last 90 wave periods, respectively, of the time series of h=U(2,j_B) at the four specified nodes j_B for the option ISAVB=1. The free surface elevation η at the node j_B can be obtained by adding z=ZB(j_B) to h=U(2,j_B). Please note that Kobayashi and Wurjanto (1991) showed the various time series at x=0.42 (node j_B=101). Their computation of Run SB used a different set of the nodes j_B. Of the six kinds of time series shown in their paper, the time series of $D_B(t)$ can not be obtained

directly from the present form of RBREAK. An additional routine to compute the first derivative of E with respect to time t and the first derivative of E_F with respect to x will be required to compute $D_B(t)$ using Eq. 53.

Table G-1: Primary input data file for BEFORR-RBREAK computation of "Kobayashi & Wurjanto 1991 Run SB".

```
 1          3                          --> NLINES
 2    ------------------------------------------------------------------------
 3    Kobayashi & Wurjanto 1991 Run SB
 4    ------------------------------------------------------------------------
 5    10                              --> IJOB,ISTAB
 6    1                               --> ISYST
 7    1                               --> IBOT
 8    1                               --> INONCT
 9    1                               --> IENERG
10    2       30  tkwsb                --> IWAVE,MSTAT,FINP2
11    1      587      590         4    --> ISAVA,MSAVA1,MSAVA2,NTIMES
12    1        4                      --> ISAVB,NNODB
13    0        0                      --> ISAVC,NNODC
14    00000         0                 --> IREQ,IELEV,IV,IDUDT,ISNR,NREQ
15         200                        --> INITS
16           1                        --> MULTIF
17            .001000                 --> DELTA (normalized)
18         590.000000                 --> TMAX  (normalized)
19          27                        --> NRATE
20           2                        --> NDELR
21         5.000000                   --> DELRP(1)(millimeters)
22        30.000000                   -->      (2)
23         2.000000       2.000000    --> X1,X2
24          .900000      13.500000    --> HREFP(meters),TP(seconds)
25         1.000000       1.000000    --> KSREF,KSSEA
26          .050000                   --> FWP
27         1.70                       --> DSEAP(meters)
28         0.050000                   --> TSLOPS
29           1                        --> NBSEG
30        68.000000       0.050000    --> WBSEG(1)(meters),TBSLOP(1)
31         1    61   121   181        --> NODB( 1)-( 2)-( 3)-( 4)
```

Table G-2: File ODOC produced by RBREAK computation of "Kobayashi & Wurjanto 1991 Run SB".

```
 1  ------------------------------------------------------------------------
 2  Kobayashi & Wurjanto 1991 Run SB
 3  ------------------------------------------------------------------------
 4
 5  WAVE CONDITION
 6
 7  Incident Wave at Seaward Boundary Given as Input
 8  Number of Data Points    NDATA =   15931
 9  Norm. Maximum Surface Elev.    =       0.915365
10  Norm. Minimum Surface Elev.    =      -1.103590
11
12  Reference Wave Period          =      13.500000 sec.
13  Reference Wave Height          =       0.900000 meters
14  Depth at Seaward Boundary      =       1.700000 meters
15  Shoal. Coef. at Reference  Ks1 =       1.000
16               at Seaw. Bdr. Ks2 =       1.000
17               Ks = Ks2/Ks1      =       1.000
18  Norm. Depth at Seaw. Boundary  =       1.889
19  Normalized Wave Length         =      32.227
20  "Sigma"                        =      44.570
21  Ursell Number                  =     549.826
22  Surf Similarity Parameter      =       0.889
23
24  SLOPE PROPERTIES
25
26  Friction Factor            =        0.050000
27  Norm. Friction Factor      =        1.114261
28  Norm. Horiz. Length of
29       Computation Domain    =        1.690956
30  Number of Segments         =        1
31  --------------------------------
32   SEGMENT     WBSEG(I)    TBSLOP(I)
33         I       meters
34  --------------------------------
35         1   68.000000     0.050000
36  --------------------------------
37
38  COMPUTATION PARAMETERS
39
40  Total Number of Spatial Nodes      JE =      400
41  Number of Nodes Along Bottom Below SWL
42                                   INITS =      200
43  Normalized Delta x          =         0.423798D-02
44  Damping Coefficients    x1 =            2.000
45                          x2 =            2.000
46  Computation Duration  TMAX =          590.000000 Wave Periods
47  Minimum Allowable NONE           NONEM =     1377
48  Time Series Stored at Rate NRATE
49       per Wave Period with        NRATE =       27
50  Statistical Calculations Are Performed Excluding
51       the First MSTAT Computation Units
52       with                        MSTAT =       30
```

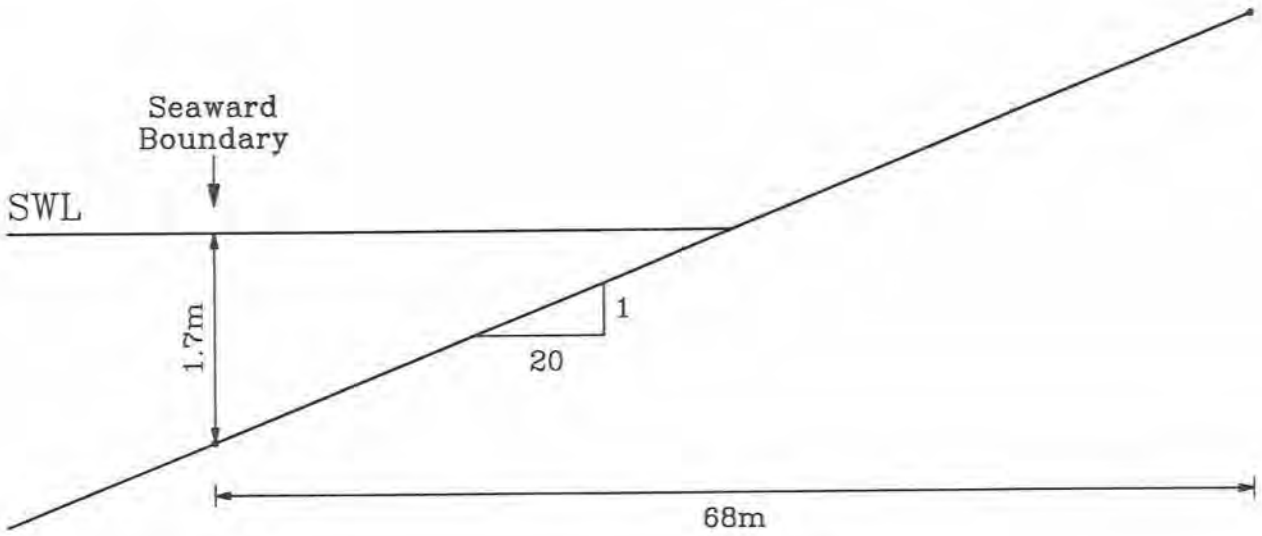
Table G-2 continued

```
 53     Spatial Variations are Stored NTIMES at Equal
 54          Intervals per Wave Period from
 55          Computation Unit MSAVA1 to MSAVA2,
 56          Inclusive, with              NTIMES =        4
 57                                       MSAVA1 =      587
 58                                       MSAVA2 =      590
 59
 60     REFLECTION COEFFICIENT
 61
 62     r1 =     0.365
 63
 64     RUNUP, RUNDOWN, SETUP
 65
 66     Largest Node Number Reached by Computational Waterline
 67                                                  JMAX =      338
 68     ------------------------------------------------------------
 69           I   DELTAR(I)     RUNUP(I)    RUNDOWN(I)    SETUP(I)
 70                  [mm]          R            Rd           Zr
 71     ------------------------------------------------------------
 72           1     5.000        1.292       -0.140        0.428
 73           2    30.000        1.275       -0.532        0.321
 74     ------------------------------------------------------------
 75
 76     WAVE SET-DOWN OR SETUP
 77
 78     Average value of ETAI =    -0.000185
 79                      ETAR =    -0.043959
 80
 81     TIME-AVERAGED ENERGY BALANCE
 82
 83     Normalized Energy Flux:
 84     . at Seaw. Boundary   A =  0.754392D-01
 85     . at Landw. Boundary  B =  0.000000D+00
 86     Normalized Rate of Energy Dissipation
 87     in the Computation Domain, Due to:
 88     . Bottom Friction     C =  0.305872D-01
 89     . Wave Breaking       D =  0.448268D-01
 90     Additional Term       E =  0.251896D-04
 91     Calculation 1:
 92                     F = A-B =  0.754392D-01
 93                   G = C+D+E =  0.754392D-01
 94       Must H=0, but H = G-F = -0.179761D-09
 95       % error  100G/E =          0.00
```

Table G-3: File BINPUT produced by BEFORR computation of "Kobayashi & Wurjanto 1991 Run SB".

Notes:
- The file is not shown in its entirety because the changes of values occured infrequently and only in the last column showing the value of DELTA for each computation unit. Only the parts showing the changes are listed below.
- The number on the left is the line number.
- Starting from the line 9, the number in the first column is sequential.
- Also starting from the line 9, the numbers in the columns 2 and 3 are constant, that is, 1377.

```
  1          3
  2     ------------------------------------------------------------------------
  3     Kobayashi & Wurjanto 1991 Run SB
  4     ------------------------------------------------------------------------
  5         590
  6     tkwsb
  7        15931
  8         1377
  9            1    1377    1377   0.100000000D-02
 10            2    1377    1377   0.100000000D-02
 11            3    1377    1377   0.100000000D-02
  .            .
  .            .
  .            .
132          124    1377    1377   0.100000000D-02
133          125    1377    1377   0.938000000D-03
134          126    1377    1377   0.938000000D-03
135          127    1377    1377   0.938000000D-03
  .            .
  .            .
  .            .
287          279    1377    1377   0.938000000D-03
288          280    1377    1377   0.854000000D-03
289          281    1377    1377   0.854000000D-03
290          282    1377    1377   0.854000000D-03
  .            .
  .            .
  .            .
555          547    1377    1377   0.854000000D-03
556          548    1377    1377   0.812000000D-03
557          549    1377    1377   0.812000000D-03
558          550    1377    1377   0.812000000D-03
  .            .
  .            .
  .            .
598          590    1377    1377   0.812000000D-03
```



Kobayashi and Wurjanto (1991) Run SB

Figure G–1

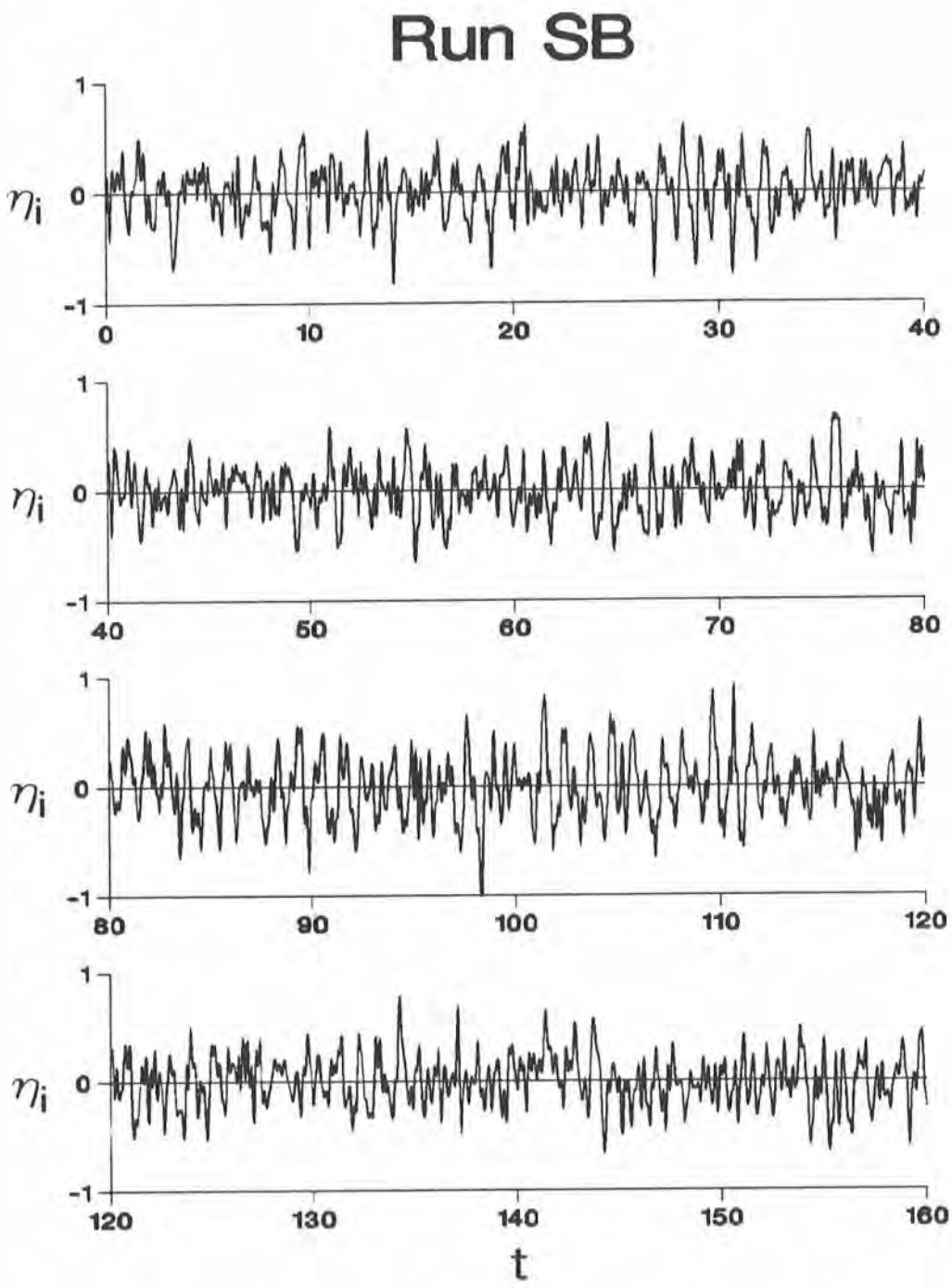

Run SB
η_i
1
0
-1
0
10
20
30
40
50
60
70
80
90
100
110
120
130
140
150
160
t

Figure G-2

Run SB

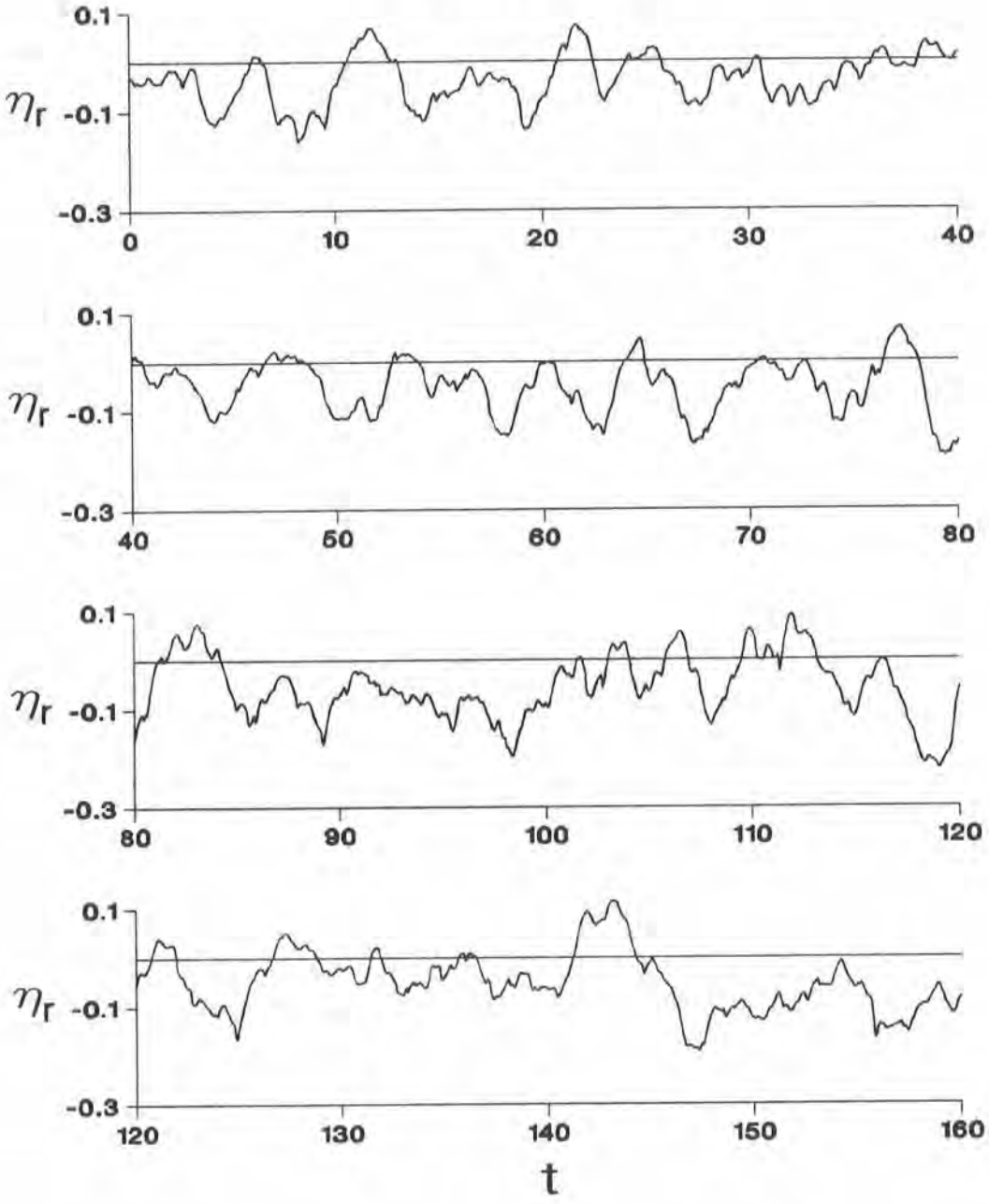


Figure G-3

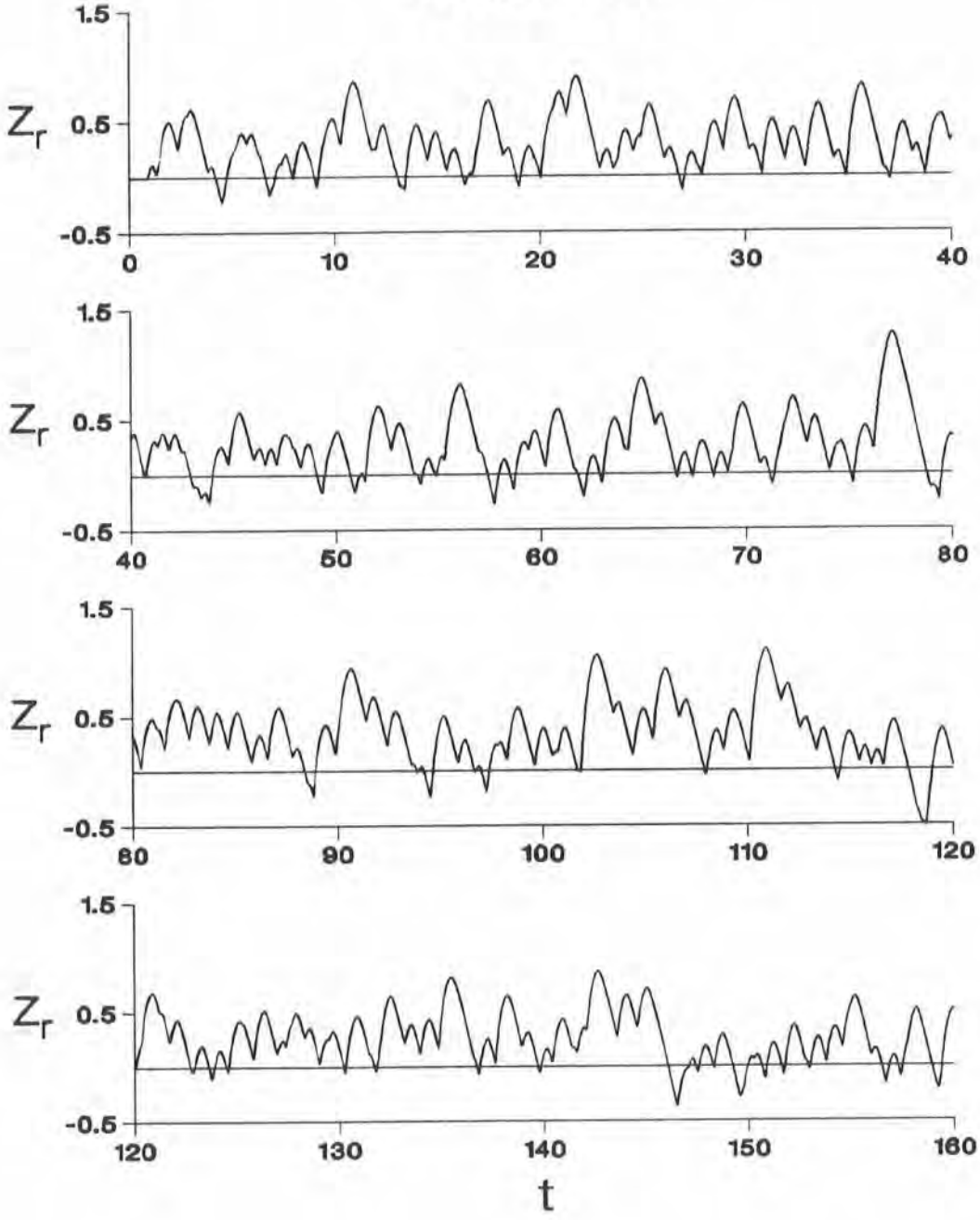

Run SB
1.5
0.5
-0.5
Z_r
0
10
20
30
40
40
50
60
70
80
80
90
100
110
120
120
130
140
150
160
t

Figure G-4

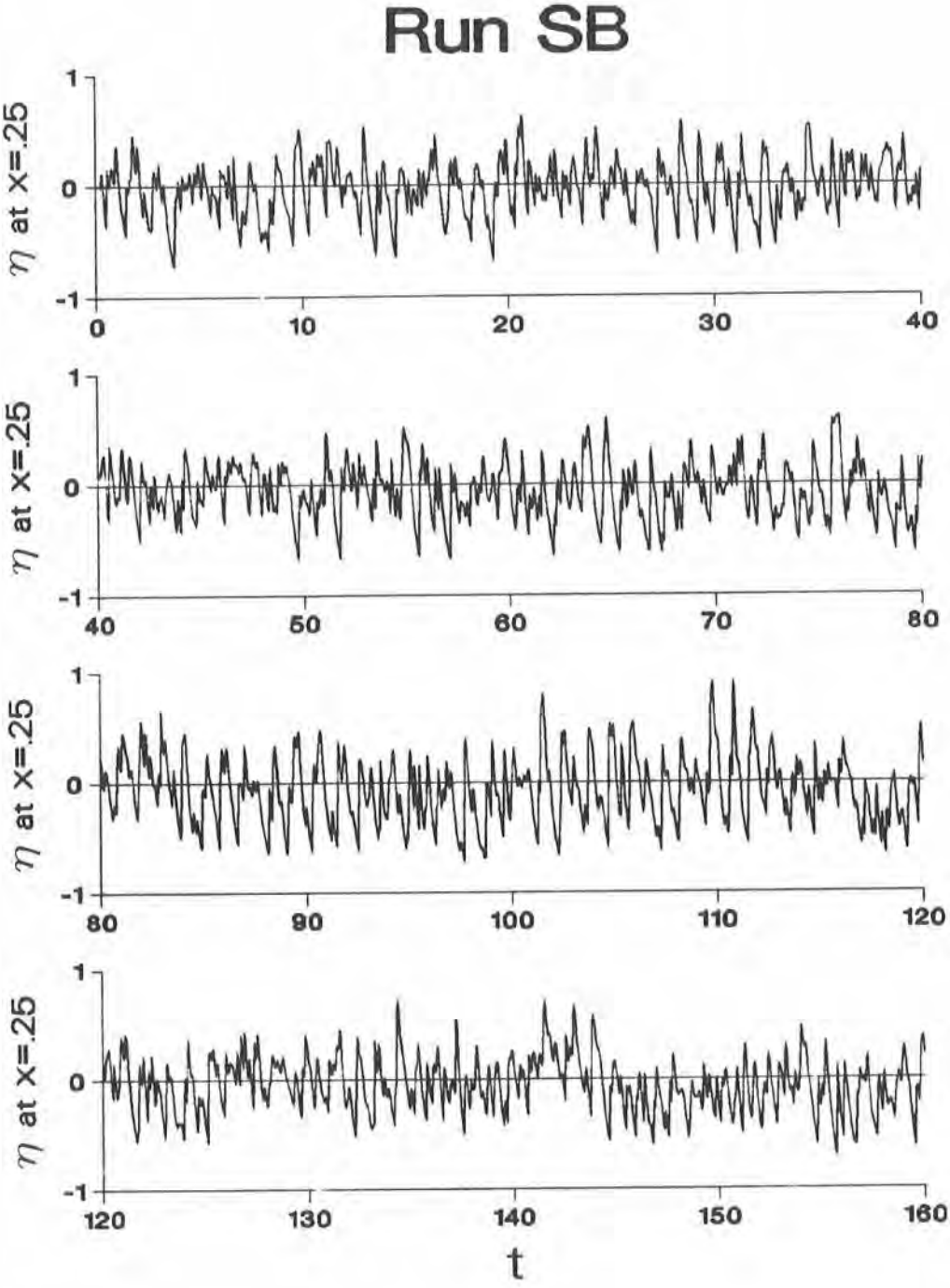

Run SB
η at x=.25
1
0
-1
0
10
20
30
40
η at x=.25
1
0
-1
40
50
60
70
80
η at x=.25
1
0
-1
80
90
100
110
120
η at x=.25
1
0
-1
120
130
140
150
160
t

Figure G-5

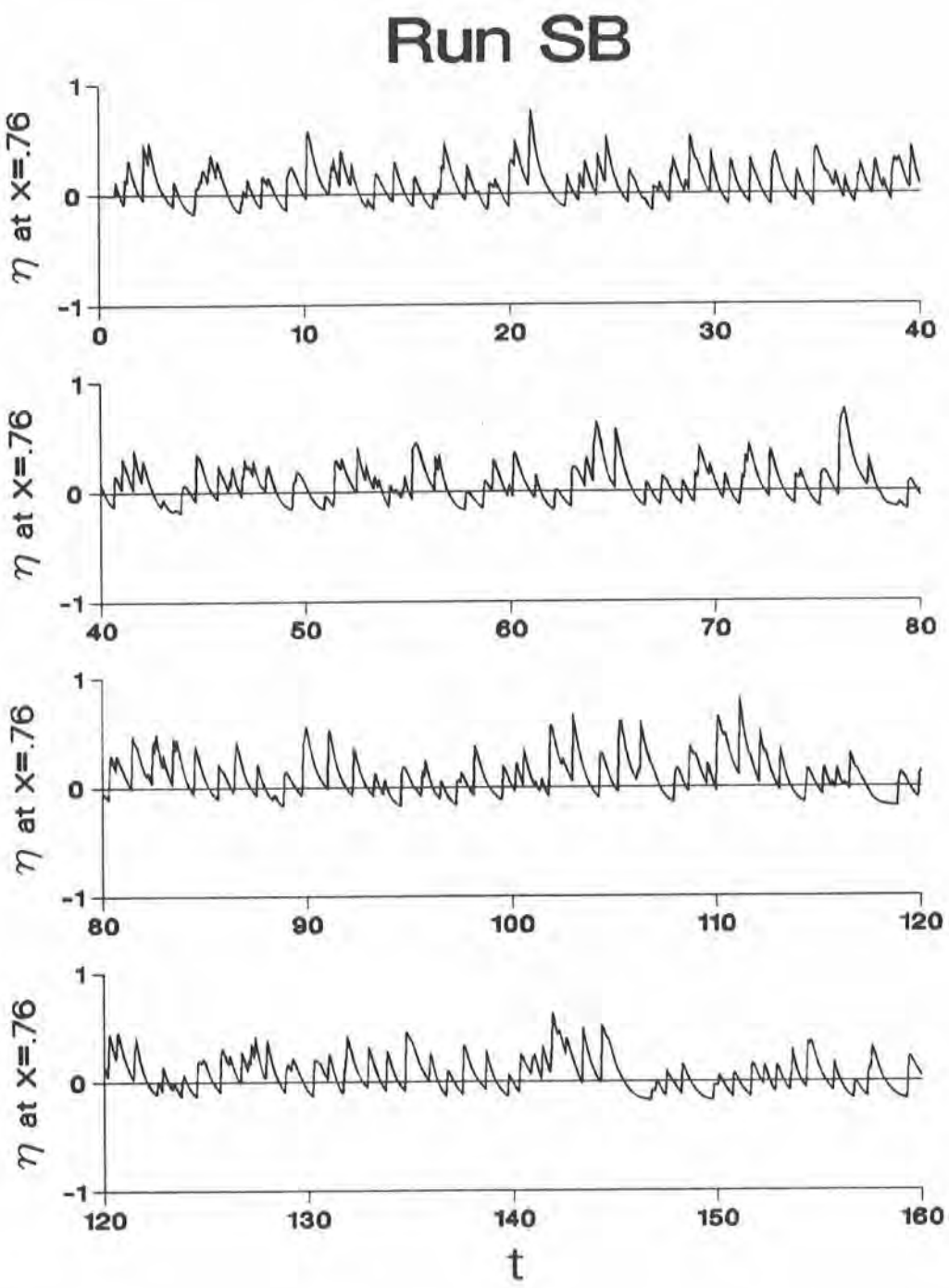

Run SB
η at x=.76
1
0
-1
0
10
20
30
40
η at x=.76
1
0
-1
40
50
60
70
80
η at x=.76
1
0
-1
80
90
100
110
120
η at x=.76
1
0
-1
120
130
140
150
160
t

Figure G-6

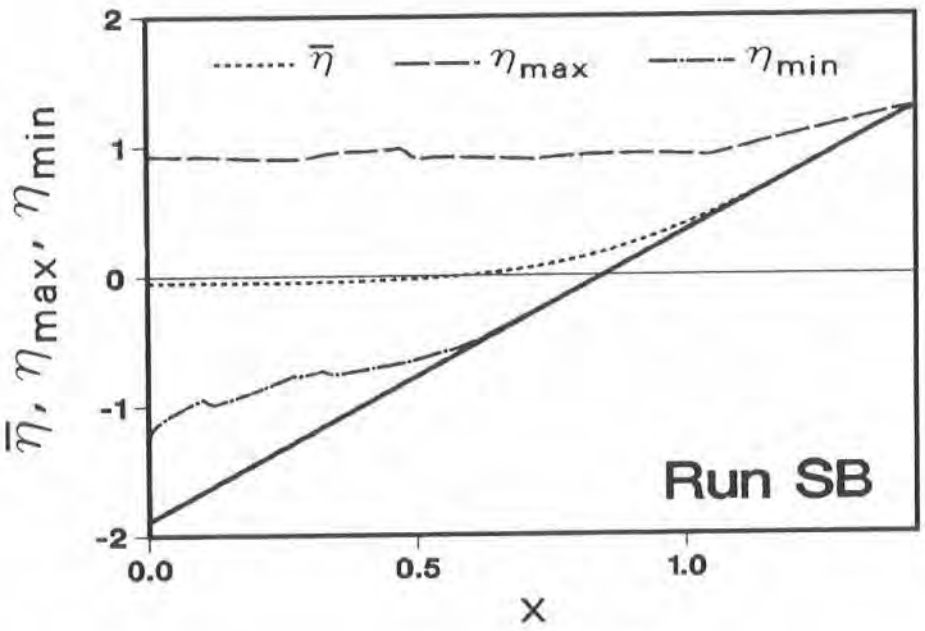


Figure G-7

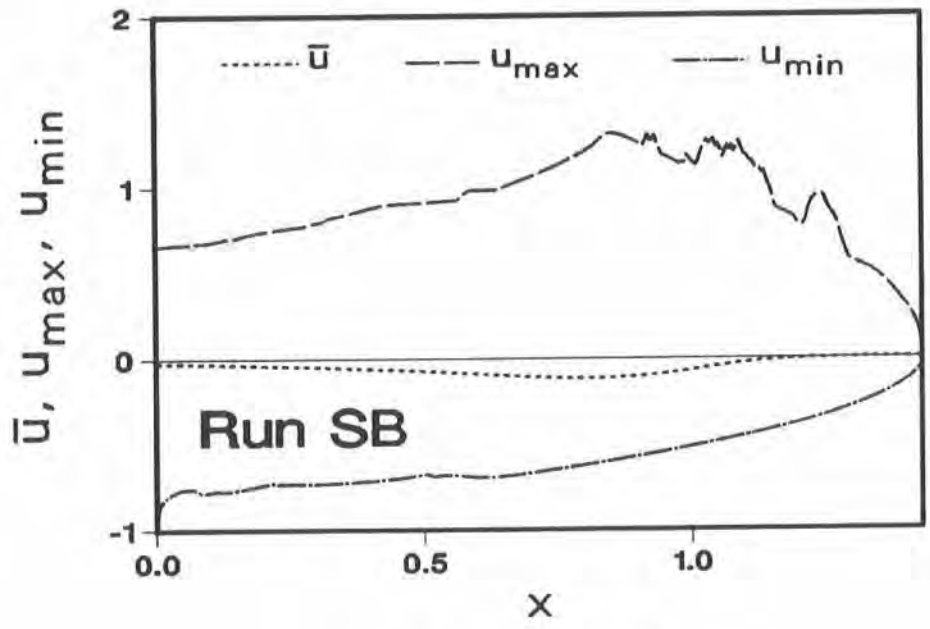


Figure G-8

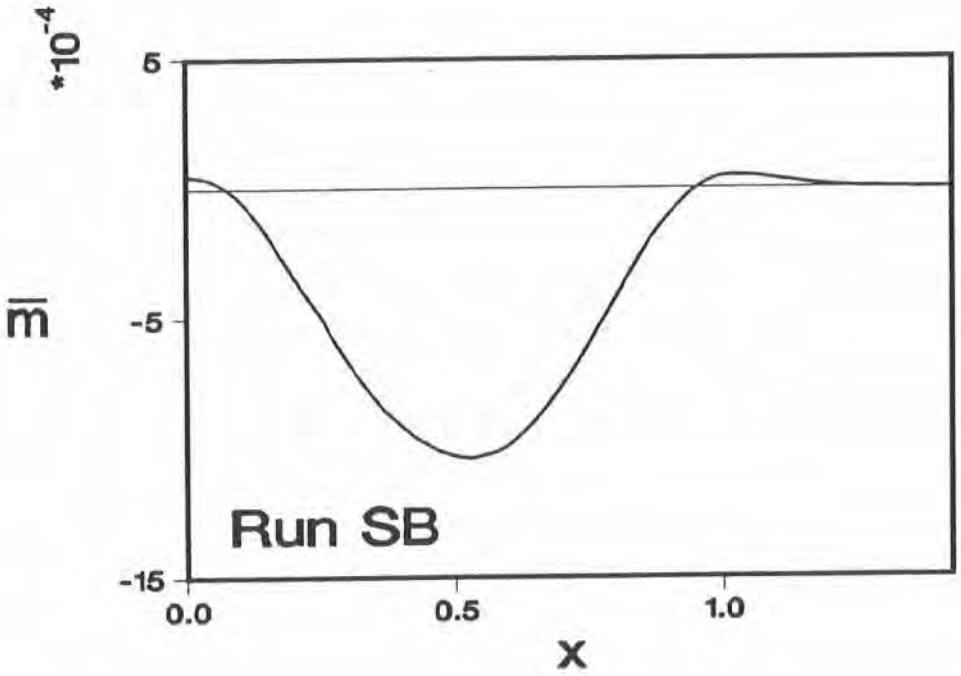


Figure G-9

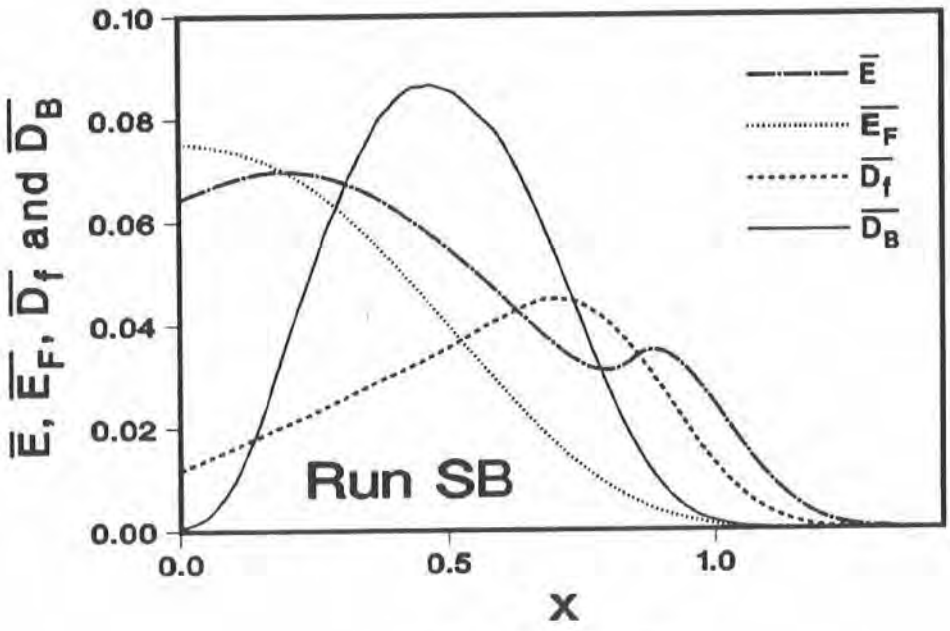


Figure G-10

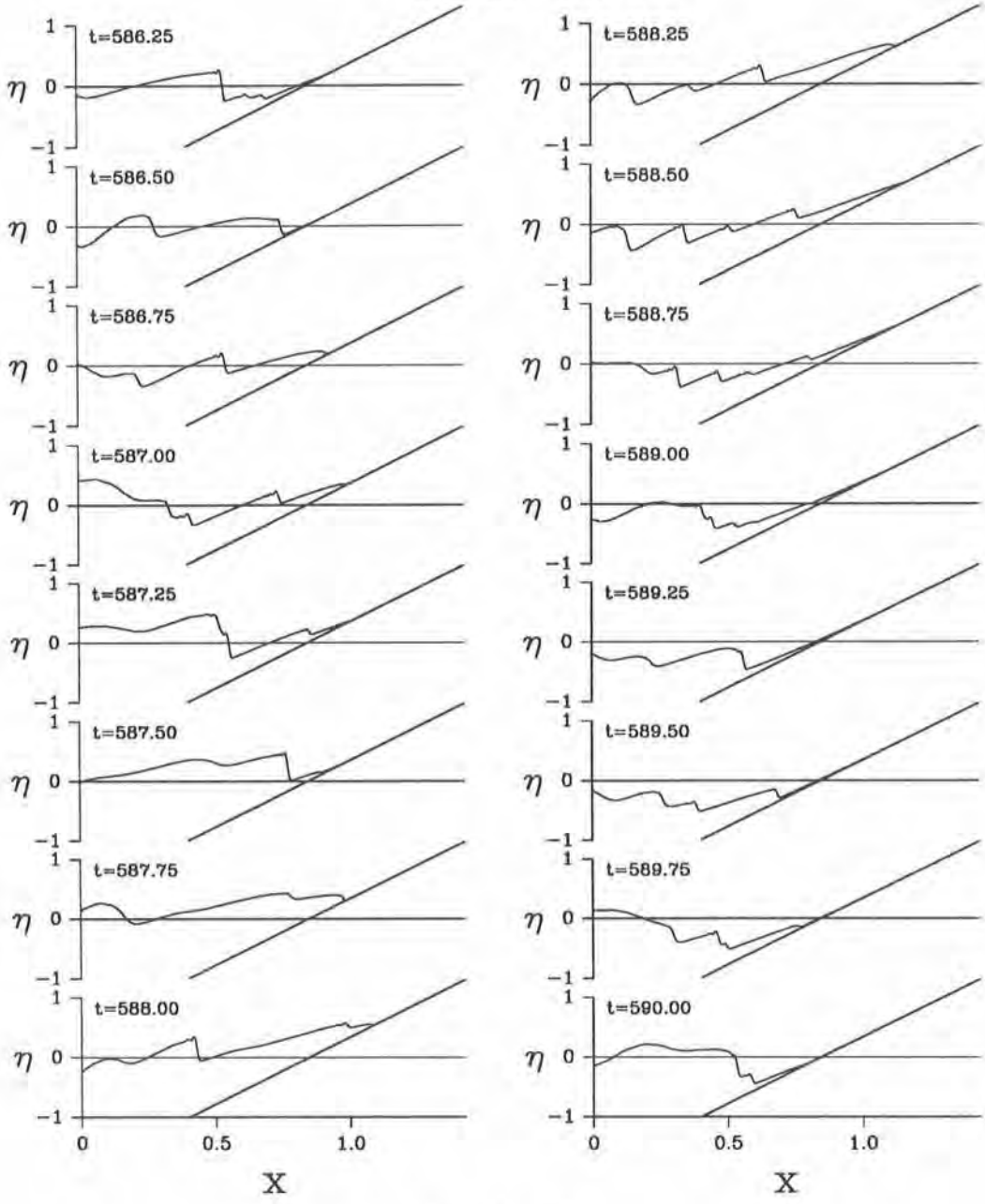

Run SB
t=586.25
t=586.50
t=586.75
t=587.00
t=587.25
t=587.50
t=587.75
t=588.00
t=588.25
t=588.50
t=588.75
t=589.00
t=589.25
t=589.50
t=589.75
t=590.00
η
X
0
0.5
1.0
1
−1

Figure G-11

Run SB

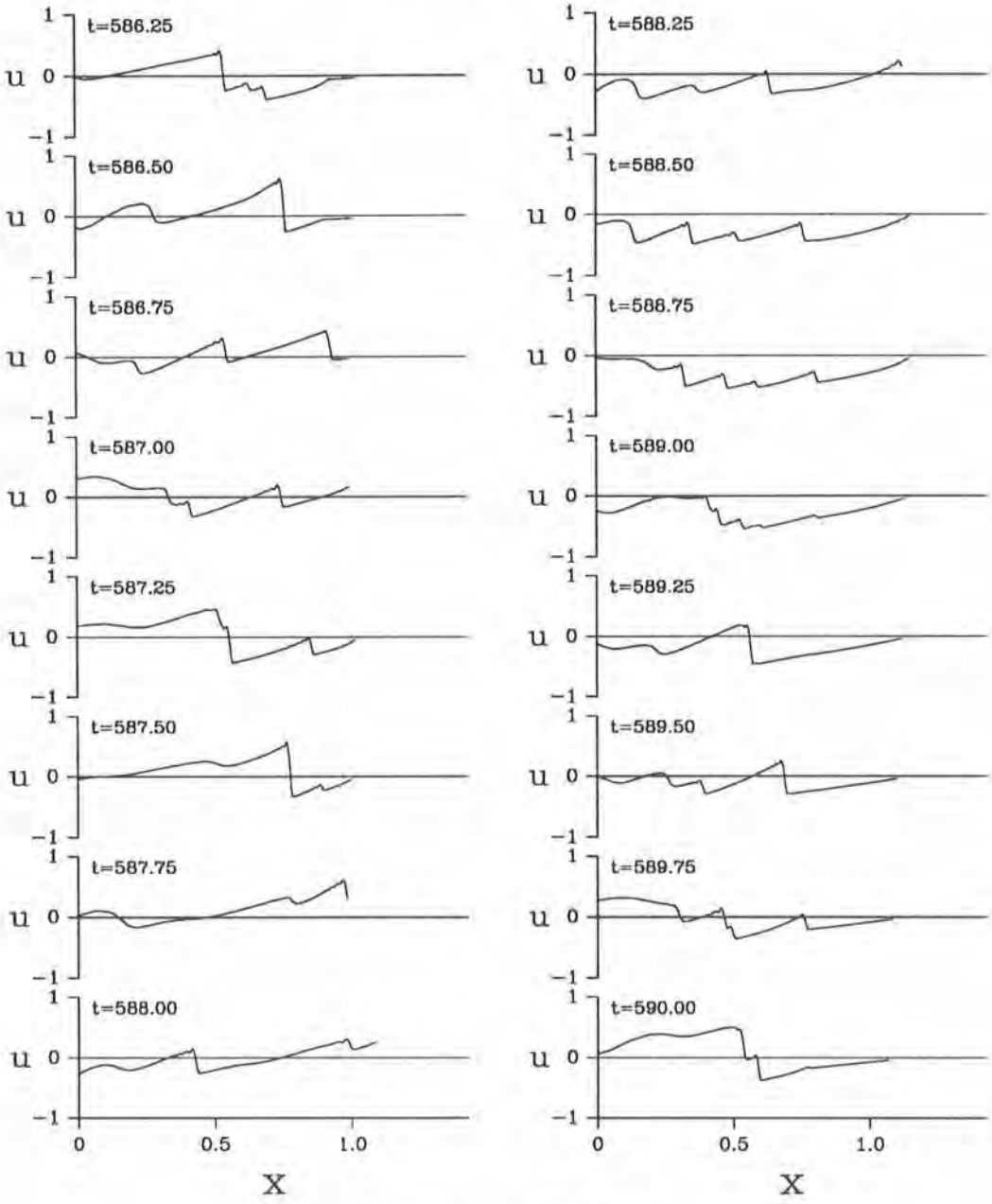


Figure G-12

APPENDIX H

CONTENTS OF THE ACCOMPANYING DISK

The computer programs BEFORR and RBREAK explained in Parts III and IV, the input files for the BEFORR-RBREAK computations discussed in Appendices C through G, the corresponding files BINPUT, and the resulting files ODOC are presented on the accompanying 3.5 inch, double-density floppy disk. The following table lists all the files stored on this disk where [*.for] = FORTRAN program; [*.inp] = the primary input data file; [*.iwt] = the file containing the input wave train; [*.bin] = the file BINPUT; and [*.doc] = the file ODOC.

Table H: Contents of the floppy disk accompanying this report.

DIRECTORY	NAME OF FILE ON DISK				REFER TO
	PROGRAM	INPUT FILE	FILE BINPUT	FILE ODOC	
program	beforr.for rbreak.for				Part III Part IV
appenc		appenc1.inp appenc2.inp appenc3.inp appenc4.inp	appenc1.bin appenc2.bin appenc3.bin appenc4.bin	appenc1.doc appenc2.doc appenc3.doc appenc4.doc	Appendix C
append		append.inp append.iwt	append.bin	append.doc	Appendix D
appene		appene.inp appene.iwt	appene.bin	appene.doc	Appendix E
appenf		appenf.inp appenf.iwt	appenf.bin	appenf.doc	Appendix F
appeng		appeng.inp appeng.iwt	appeng.bin	appeng.doc	Appendix G

APPENDIX I

COMPUTATION TIME

The following table presents the computer time for each of the computed examples discussed in Appendices C through G. The duration listed under "SUN 4/280" indicates the *waiting* time for each computation using SUN 4/280, whereas that under "IBM 3090-300E" is the *CPU* time used for each computation using this computer. The tabulated results are approximate since the observation was done manually. The computer programs BEFORR and RBREAK have been written such that they will run on other computers with no or little adjustment. It is considered impractical to perform the computation for cases with long durations such as those in Appendices D, E, F, and G using SUN 4/280.

Table I: Computation time for the cases discussed in this report.

CASE	COMPUTATION TIME					
	SUN 4/280			IBM 3090-300E		
	IBREAK	BEFORR	RBREAK	IBREAK	BEFORR	RBREAK
Ahrens Test 12 Appendix C	17 min.	9 min.	11 min.	46 sec.	23 sec.	28 sec.
Ahrens Test 32 Appendix C	11 min.	6 min.	7 min.	40 sec.	14 sec.	24 sec.
Saville Run 10 Appendix C	28 min.	17 min.	15 min.	76 sec.	43 sec.	39 sec.
Seelig Run 2 Appendix C	25 min.	21 min.	24 min.	62 sec.	58 sec.	64 sec.
Kobayashi, Cox, & Wurjanto Run I1 Appendix D	N.A.	N.P.	N.P.	N.A.	(1) 75 min. (2) 125 min.	(1) 105 min. (2) 215 min.
Kobayashi, Cox, & Wurjanto Run I3 Appendix E	N.A.	N.P.	N.P.	N.A.	135 min.	160 min.
Kobayashi & Wurjanto Run 1 Appendix F	N.A.	N.P.	N.P.	N.A.	(1) 15 min. (2) 40 min.	(1) 20 min. (2) 55 min.
Kobayashi & Wurjanto Run SB Appendix G	N.A.	N.P.	N.P.	N.A.	45 min.	65 min.

N.A. = Not Applicable
N.P. = Not Performed