AD- 781 379

AFFDL-TR-74-23

A USER'S MANUAL FOR THE SEQUENCE ACCOUNTABLE FATIGUE ANALYSIS COMPUTER PROGRAM

J. M. POTTER R. A. NOBLE

TECHNICAL REPORT AFFDL-TR-74-23

MAY 1974

Approved for public release; distribution unlimited.

Best Available Copy

AIR FORCE FLIGHT DYNAMICS LABORATORY AIR FORCE SYSTEMS COMMAND WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

20060921115

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This Technical Report has been reviewed and is approved.

FRANCIS' J .TAXÍŤK

Chief, Solid Mechanics Branch Structures Division

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document. AIR FORCE/56780/18 June 1974 - 300 UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data	Entered)	
REPORT DOCUMENTATION	PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
AFFDL-TR-74-23		
4. TITLE (and Subtitle)	1	5. TYPE OF REPORT & PERIOD COVERED
A USER'S MANUAL FOR THE SEQUENCE A	ACCOUNTABLE	1 Oct 1972 to 1 May 1973
FATIGUE ANALYSIS COMPUTER PROGRAM		Final Report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s)		8. CONTRACT OF GRANT NUMBER(s)
J. M. Potter and R. A. Noble		
9. PERFORMING ORGANIZATION NAME AND ADDRESS		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
Air Force Flight Dynamics Laborate	ory	10/7 0/ 10
Wright-Patterson Air Force Base, (Dhio 45433	1347 04-18
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE
Air Force Flight Dynamics Laborate	2237	May 107/
Wright-Patterson Air Force Base. ()hio 45433	13. NUMBER OF PAGES
		63
14. MONITORING AGENCY NAME & ADDRESS(if different	t from Controlling Office)	15. SECURITY CLASS. (of this report)
		UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING
		SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report)	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	
Approved for public release: dist	ribution unlimit	ed
11 1 1 1 - 1 1 1 - 1 1 1 - 1 1 1 - 1 -		
17. DISTRIBUTION STATEMENT (of the abstract entered i	n Block 20, if different from	n Report)
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and	identify by block number)	
Aircraft	Residual Stres	s Relaxation
Structures	Computer Progr	am
Fatigue Life Prediction		
Residual Stress		
20. ABSTRACT (Continue on reverse side if necessary and	identify by block number)	
This report presents a detailed d	lescription of a	computer program to calculate
cumulative damage of notched struct	ural members sub	jected to arbitrary spectra.
The Sequence Accountable Fatigue An	alysis computer	program develops its sequence
sensitivity by tracking residual st	resses local to	a notch throughout the spec-
trum of loads. Residual stress rel	axation analysis	is included to increase the
generality of the results. An exam	pie spectrum and	resulting cumulative damage
analysis are illustrated.		

UNCLASSIFIED

FOREWORD

This program was prepared by J. M. Potter of the Solid Mechanics Branch and R. A. Noble of the Experimental Branch, Structures Division, Air Force Flight Dynamics Laboratory. The work was conducted in-house under Project 1347 "Structural Testing of Flight Vehicles", Task 134704 "Structural Testing Criteria". This report covers work accomplished over a time period of 1 October 1972 to 1 May 1973. The essence of the analysis was presented in AFFDL-TM-73-131-FBR in October 1973.

This manuscript was released by the authors in January 1974.

ABSTRACT

This report presents a detailed description of a computer program to calculate cumulative damage of notched structural members subjected to arbitrary spectra. The Sequence Accountable Fatigue Analysis computer program develops its sequence sensitivity by tracking residual stresses local to a notch throughout the spectrum of loads. Residual stress relaxation analysis is included to increase the generality of the results. An example spectrum and resulting cumulative damage analysis are illustrated.

TABLE OF CONTENTS

SECTION		PAGE
I	Introduction	1
II	Program Outline	2
III	Input Data Requirements	6
IV	Output Options	9
REFERENCES		10
FIGURES		12
APPENDIX I	Program List	17
APPENDIX II	Sample Problem with Input Data Listing	39
APPENDIX III	List of Computer Program Symbols and Definitions	57
APPENDIX IV	Fatigue Life Input Data for Several Materials	63

SYMBOLS

σres	Residual stress
σ_{max}	Maximum local stress level
σ_{\min}	Minimum local stress level
^ơ ys	Yield stress
^o res _{EO}	Equilibrium component of the residual stress
εt	Local strain, total
ε _e	Elastic component of total strain
e p	Plastic component of total strain
ε _f ,	Strain intercept at one reversal on a log ε_p -log life curve
c	Slope of the log ϵ_p -log life curve
S max	Maximum applied stress level
S min	Minimum applied stress level
S _{mean}	Mean applied stress level, $(S_{max} + S_{min})/2$
K _t	Elastic stress concentration Factor
D	Damage
I	Integer describing the level number
N _{EP}	Equilibrium period, number of cycles for the local stresses to return approximately to the equilibrium conditions following an overload
C1	Residual stress relaxation constant
E1, E2	Relaxation function exponents
N	Number of cycles
Е	Modulus of elasticity
N _f	Number of cycles of life at a given stress or strain cycling level

SECTION I

INTRODUCTION

Cumulative damage analyses based upon the local stress-strain behavior at a notch appear to be reasonably successful in anticipating trends in fatigue life behavior of notched specimens subjected to spectrum loading (1-6). The type of behavior that usually occurs is that peak tensile loads tend to increase the fatigue life and peak compressive loads tend to decrease the life of notched structures compared to structures experiencing load spectra not having those peaks (5,6). Local behavior analyses, such as those developed by Smith (7) and Neuber (8), help to explain this phenomenon as being a result of the tensile peak load creating a compressive residual stress at the notch and, conversely, the compressive peak creating a tensile residual stress. The change in life occurs because the residual stress state modifies the subsequent damage accumulation rates.

The Sequence Accountable Fatigue Analysis computer program was developed to incorporate the local stress-strain approach with a recent residual stress relaxation analysis ⁽⁶⁾ in order to improve the sequence sensitivity of cumulative damage analysis. This technical memorandum presents the details of the resultant computer program and an example of its use. The correlation of predictions made with this analysis to actual results of tests experiencing spectrum loading is presented by Potter ⁽⁹⁾, and Potter, Gallagher, and Stalnaker (10).

SECTION II

PROGRAM OUTLINE

The Sequence Accountable Fatigue Analysis traces the stress-strain behavior local to a notch throughout an applied load spectrum and calculates the damage based on the local experience. The computer program is divided generally into the four parts or modules outlined in Fig. 1.

The basic input data for the material, specimen geometry, fatigue behavior qualities and spectrum, are developed in Module I. The information required in Module I is discussed further in Section III. Module II takes the input information and determines the local stress-strain behavior. Module III references the Range Pair Counting Method Subroutine to cycle count the local stress spectrum. Module IV determines the damage in the local stress-strain spectrum.

The basic analyses used in Modules II, III and IV are presented below.

<u>Module II - Local Stress-Strain Behavior</u> - The analysis used during the determination of the local stress behavior during the spectrum of loading is a combination of analyses developed by Smith (7), Neuber (8) and Potter (6). Smith's simple analysis indicated that the residual stress could be approximated by assuming that the initial stress-strain behavior was elastic upon unloading following plastic flow. Thus, the residual stress could be calculated knowing the

maximum local stress and the maximum applied stress as in Eq. 1 and in Fig. 2.

$$\sigma_{\text{res}_i} = \sigma_{\max_i} - K_t S_{\max_i}$$
(1)

The Sequence Accountable Fatigue Analysis computer program currently incorporates elastic-perfectly plastic stress-strain behavior. Therefore, σ_{\max_i} is equal to the yield stress. For the cycles immediately following the peak stress, the residual stress determined in Eq. 1 modifies the elastic solution as shown in Eqs. 2 and 3 (provided that the following maximum applied stress is less than S_{\max} and that there is no change in the residual stress due to a minimum applied stress causing reversed yielding).

$$\sigma_{\max_{i}} = \sigma_{\operatorname{res}_{i-1}} + K_{t} S_{\max_{i}}$$
(2)

$$\sigma_{\min_{i}} = \sigma_{\operatorname{res}_{i-1}} + K_{t} S_{\min_{i}}$$
(3)

The analysis developed by Neuber ⁽⁸⁾ has been extended to cyclic loading by Wetzel ⁽²⁾ and Wetzel, Morrow and Topper ⁽³⁾ and used by many others ^(1,4-6) primarily to determine local stress-strain behavior. It is used in this program only to calculate plastic strains occurring when the residual stress undergoes a step change. The plastic strain calculation routine is accessed only when the σ_{\max_i} or σ_{\min_i} terms in Eqs. 2 and 3 exceed tensile or compressive yield stress levels, respectively. Figure 3 illustrates the calculation of the plastic strain.

The local stress-strain behavior, according to Wetzel ⁽²⁾ is related to the applied load by Eq. 4

$$\sigma \cdot \epsilon = (K_{\text{t}} S_{\text{max}})^2 / E$$
 (4)

The plastic strain can be found by subtracting the elastic component from the total strain.

$$\varepsilon_{\rm p} = \varepsilon_{\rm t} - \varepsilon_{\rm e} = (K_{\rm t} S_{\rm max})^2 / E \cdot \sigma_{\rm max} - \sigma_{\rm max} / E$$

Therefore, the plastic strain associated with $S_{max_{\star}}$ is given in Eq. 5.

$$\varepsilon_{p_{i}} \approx (K_{t}S_{max_{i}})^{2}/E\sigma_{ys} - \sigma_{ys}/E$$
(5)

If a residual stress existed prior to this plastic strain excursion, the plastic strain associated with that prior excursion is subtracted from Eq. 5 as shown in Eq. 6.

$$\varepsilon_{p_{i}} = (K_{t}S_{max_{i}})^{2}/E\sigma_{ys} - (\sigma_{ys} - \sigma_{res_{i-1}})^{2}/E\sigma_{ys}$$
(6)

A similar calculation is made for plastic strains occurring during the minimum stress peak.

In the analysis developed by Potter ⁽⁶⁾ the residual stress cyclically relaxes toward zero or an equilibrium residual stress as shown in Fig. 4 according to Eq. 7.

$$\sigma_{\text{res}_{N=1,2,...}} = (\sigma_{\text{res}_{N=1}} - \sigma_{\text{res}_{EQ}}) \exp(N/N_{EP_1} \ln(0.1))$$
(7)

The N_{EP} term, the Equilibrium Period, is dependent upon the applied stress and the Residual Stress Relaxation Constant.

$$N_{EP_{i}} = (C1/|K_{t}S_{max_{i}}|^{E1} |K_{t}S_{mean_{i}}|^{E2})$$
(8)

The Residual Stress Relaxation Constant, C1, has not yet been experimentally defined but should be a constant for a material.

Module III - Cycle Counting Method

After the local stress and plastic strain behavior is calculated, the local stress spectrum is Range Pair Counted using a computer program developed by Tischler.⁽¹¹⁾

Module IV - Damage Calculation

Damage is calculated separately for the plastic strain excursions and the elastic stress spectrum. The damage is determined from the conventional D = $\sum_{n=1}^{n}$ calculation. Damage from each of the plastic strain cycles is determined from the Coffin-Manson ⁽¹²⁾ form

$$D_{i} = 1./N_{f_{i}} = 1./(\epsilon_{p_{i}}/\epsilon_{f'})^{1/c}$$

Damage from the elastic stress cycles is determined in a similar manner. The maximum and minimum local stress levels are sequentially compared to unnotched S-N data in a Modified Goodman Diagram format. Damage is summed, and failure of the coupon is defined as the event occurring when the summed damage equals unity.

SECTION III

INPUT DATA REQUIREMENTS

In general, each spectrum analyzed will require slightly different programming in order to get the load history into a usable format for the core program. The basic program requires a certain family of information before any analytical predictions can be made. Appendix I contains a program listing for the Sequence Accountable Fatigue Analysis. The subroutine CORE which accesses the subroutines having to do with RPCM, the Range Pair Counting Method, contains the basic analysis. Subroutine SAL reads the data input and then references subroutine CORE. The subroutine SAL shown is one in which a block of cycles is repeated with optional cycles. A list of the input data cards and the resulting analysis is given in Appendix II.

The specific data requirements are given below.

1. <u>Stress-Strain Behavior</u> - The stress-strain behavior is presumed to be elastic-perfectly plastic with the tensile yield stress being equal to the compressive yield stress. The yield stress value used is an average of the monotonic behavior generally being above the 0.2% yield value and below the engineering ultimate strength.

2. <u>Residual Stress Relaxation</u> - The residual stress relaxation behavior of Eq. 7 and 8 is characterized by Cl, the Residual Stress Relaxation Constant and El and E2, the relaxation equation exponents. The **R**esidual Stress Relaxation Constant, Cl, has not yet been adequately determined. It should be a material property if the relaxation function

is correct and must be assumed. A reasonably accurate estimate of the Residual Stress Relaxation Constant for aluminum material falls in the range of 5-20 x 10^6 (cycles) (Ksi)². Further experimentation on the part of the analyst should develop a Cl usable for his set of conditions until actual measurement of residual stress relaxation behavior defines the relaxation function and constants. At present El and E2 are considered to be equal to 1.0.

3. <u>Specimen Geometry</u> - The elastic K_t value (if available) is entered into the analysis. If that value is not available then an estimate from some other method may be used. In certain cases, a value may be determined from a constant amplitude fatigue test of a similar structure by fitting several values of K_t to the analysis and determining the best correlation as is done with the K_f solution. Once a stress concentration factor, K_t , is determined for a specimen, that value is not changed from test-to-test of the same coupon donfiguration.

4. <u>Load Multiplier</u> - Different spectra are presented for analysis in different manners. Some data are presented in percent of maximum stress, others in terms of nominal stress, and others in terms of bending moment. The value of the load multiplier defines the nominal stress history.

5. <u>Cumulative Damage Analysis</u> - The damage from the range-paired elastic stress spectrum is determined by calculating a simple $\frac{n}{N}$ value for each level and accumulating the total. The N_{fi} value is determined from unnotched coupon S-N data in the Modified Goodman Diagram format.

The program requires the input of four second order equations describing the maximum and minimum stress levels at lives of 10^4 , 10^5 , 10^6 and 10^7 cycles. The coefficients of the equations are derived by least square fitting the S-N data presented in the form of Eq. 9.

$$S_{\max} = A(I) S_{\min}^{2} + B(I) S_{\min} + C(I)$$
 (9)

The A, B, and C coefficients for several typical materials are presented in Appendix IV. The S-N data shown was derived from various sources but usually from the MIL-HDBK-5A (13). The C coefficients correspond to the maximum stress level at zero to maximum applied stress conditions on the unnotched coupons.

The damage from the plastic strain cycles is determined using the Coffin-Manson relation to calculate the N_{f_1} value. The conventional plastic strain intercept at one reversal and the ϵ_p - life slope values are used in the analysis. Specific measured values from the literature are used when available and typical values when they are not available.

6. <u>Analysis or Test Spectrum</u> - The last information needed is the order and magnitude of application of the spectrum used in the test.

SECTION IV

OUTPUT OPTIONS

The Computer Program prints the following output in the process of the analysis.

1. Maximum and minimum applied stress and local stress response through the spectrum. Also printed out is the residual stress, equilibrium stress, applied cycles, and the equilibrium period.

2. The elastic local stress history as input into the Range Pair Subroutine and the resulting Range Paired spectrum.

3. The plastic strain occurrence during the spectrum and the damage associated with each strain reversal.

4. The accumulated damage associated with the plastic strains.

5. The Range Paired elastic stress spectrum and the damage associated with each level.

6. The accumulated damage associated with the current block of loading including the plastic strain damage and the total damage since the initiation of cycling.

At the option of the analyst, he can print out all the above items or only two. The IPRINT value controls what data is printed. If IPRINT = 1, all six items are printed for each flight or block. If IPRINT = 2, all items except 2. above are printed. If IPRINT = 3, only items 4. and 6. above are printed.

REFERENCES

 Crews, J. H., Jr., "Crack Initiation at Stress Concentrations as Influenced by Prior Local Plasticity," <u>Achievement of High Fatigue</u> <u>Resistance in Metals and Alloys, ASTM STP 467</u>, 1970, pp. 37-52.

2. Wetzel, R. M., "Smooth Specimen Simulation of the Fatigue Behavior of Notches," <u>Journal of Materials</u>, JMLSA, Vol. 3, No. 3, Sept. 1968, p. 646.

3. Wetzel, R. M., Morrow, JoDean, and Topper, T. H., <u>Fatigue of Notched</u> <u>Parts with Emphasis on Local Stresses and Strains</u>, NADC-ST-6818, Naval Air Development Center, Johnsville, Pa., Sept. 1968.

4. Impellizzeri, L. F., "Cumulative Damage Analysis in Structural Fatigue, <u>"Effects of Environment and Complex Load History on Fatigue</u> <u>Life</u>," ASTM STP 462, American Society for Testing and Materials, 1970, pp. 40-68.

5. Stadnick, S. J., <u>Simulation of Overload Effect in Fatigue Based on</u> <u>Neuber's Analysis</u>, M. S. Thesis, Dept. of Theoretical and Applied Mechanics, University of Illinois, Urbana, 1969. See also T.A.M. Report No. 325.

6. Potter, J. M., "The Effect of Load Interaction and Sequence on the Fatigue Behavior of Notched Coupons," in <u>Cyclic Stress-Strain Behavior</u>-<u>Analysis, Experimentation, and Failure Prediction, ASTM STP 519</u>, American Society for Testing and Materials, 1973, pp. 109-132.

 Smith, C. R., "Prediction of Fatigue Failures in Aluminum Alloy Structures," <u>Proceedings of the SESA</u>, Vol. 12, No. 2, p. 21, 1955.
 Neuber, H., "Theory of Stress Concentration for Shear Strained Prismatical Bodies with Arbitrary Non-Linear Stress-Strain Law," <u>Transactions of the ASME</u>, Journal of Applied Mechanics, Dec. 1961, p. 544.

9. Potter, J. M., <u>An Experimental and Analytical Study of Spectrum</u> <u>Truncation Effects</u>, AFFDL-TR-73-117, Air Force Flight Dynamics Laboratory, Dayton, Ohio, 1973 (AD).

10. Potter, J. M., Gallagher, J. P., and Stalnaker, H. D., <u>The</u> <u>Effect of Spectrum Variations on the Fatigue Behavior of Notched</u> <u>Structures Representing F-4E/S Wing Stations</u>, AFFDL-TM-74-2-FBR,

Air Force Flight Dynamics Laboratory, Dayton, Ohio, January 1974.

11. Tischler, V. A., <u>A Computer Program for Counting Load Spectrum</u> Cycles Based on the Range Pair Cycle Counting Method, TM-FBR-72-4,

Air Force Flight Dynamics Laboratory, Dayton, Ohio, Nov. 1972.

12. Grover, H. J., <u>Fatigue of Aircraft Structures</u>, NAVAIR 01-1A-13, Naval Air Systems Command, Washington, D. C., 1966.

13. <u>Metallic Materials and Elements for Aerospace Vehicle Structures</u>, MIL-HDBK-5A, Department of Defense, Washington, D. C., Feb. 1966.



FIGURE 1. PROCEDURE USED IN THE SEQUENCE ACCOUNTABLE FATIGUE ANALYSIS







FIGURE 3. METHOD OF DETERMINING PLASTIC STRAIN LEVELS



RESIDUAL STRESS

LOCAL STRESS RESPONSE

TIME

K_TS_{MAX}



APPLIED STRESS HISTORY

5 5 10 15 15 20 20 25 20 25 20 20 20 25 20 20 20 20 20 20 20 20 20 20 20 20 20	<pre>AM SAL(INPUT, TAPES=INPUT, OUTPUT, TAPES=OUTPUT) AMODULE I INPUT ROUTINE FOR THE ENCE ACCOUNTABLE FATIGUE ANALYSIS FROGRAM DEVELOPED BY JOHN M. POTTER AND ROBERT A. NOBLE PROGRAM DEVELOPED BY JOHN M. POTTER AND ROBERT A. NOBLE MARCH 1973 ADDRESS AFFOLYER ADDRESS AFFOLYER IFLEPHONE (513) 255-6104 OR 255-6106 TELEPHONE (513) 255-6104 OR 255-6106 CARD 1. NDECK = THE NUMBER OF DATA DECKS TO BE RUN SEQUENTIALLY. I PERMITS HAXINUM PRINTOUT I PERMITS ADDRESS ANNOE-PAIR PRINTING CARD 1. NDECK = THE NUMBER OF DATA DECKS TO BE RUN SEQUENTIALLY. I PERMITS HAXINUM PRINTOUT I PERMITS HAXINUM PRINTOUT</pre>		AI
5 26 26 26 26 26 26 26 26 26 26 26 26 26	HODULE I HODULE I ENCE ACCOUNTABLE FATIGUE ANALYSIS ENCE ACCOUNTABLE FATIGUE ANALYSIS FROCRAM DEVELOPED BY JOHN M. POTTER AND ROBERT A. NOBLE OF THE AIR FORCE FLIGHT DYNAMICS LABORATORY ARCH 1973 ADDRESSI AFFOLFBR ADDRESSI AFFOLFBR ADDRESSI AFFOLFBR IFLEPHONE (513) 255-6104 OR 255-6106 TELEPHONE (513) 255-6104 OR 255-6106		: AI
5 10 15 20 25 25 25 25 25 26 26 26 26 26 26 26 26 26 26 26 26 26	MODULE I INPUT ROUTINE FOR THE ENCE ACCOUNTABLE FATIGUE ANALYSIS ###CH 1973 ADDRESS; AFFDL/FBR MARCH 1973 ADDRESS; AFFDL/FBR MARCH 1973 ADDRESS; AFFDL/FBR MARCH 1973 TELEPHONE (513) 255-6104 OR 255-6106 TELEPHONE (513) 255-6104 OR 255-6106		: AI
10 15 20 25 26 26 26 26 26 26 26 26 26 26 26 26 26	PROGRAM DEVELOPED BY JOHN M. POTTER AND ROBERT A. NOBLE OF THE AIR FORCE FLIGHT DYNAMICS LABORATORY MARCH 1973 ADDREESS, AFFDL/FBR MIGHT-PATTERSON AFB, OH 10 45433 TELEPHONE (513) 255-6104 TELEPHONE (513) 255-6104 TELEPHONE (513) 255-6106 TELEPHONE (513) 255-6104 TELEPHONE (513) 255-5104 TELEPHONE (513) 255-5104 TELEP		A
15 20 20 20 20 20 20 20 20 20 20 20 20 20	PROGRAM DEVELOPED BY JOHN M. POTTER AND ROBERT A. NOBLE OF THE AIR FORCE FLIGHT DYNAMICS LABORATORY MARCH 1973 ADDRESSI AFFDL/FBR NRIGHT-PATTERSON AFB, OHIO 45433 TELEPHONE (513) 255-6104 OR 255-6106 TELEPHONE (513) 255-5104 OR 255-5106 TELEPHONE (513) 255-5104 OR 255-6106 TELEPHONE (513) 255-5104 OR 255-5106 TELEPHONE (513) 255-5104 OR 255 TELEPHONE (513) 255-5104 OR 255-5106 TELEPHONE (513) 255-5104 OR 255-5106 TELEPHONE (513) 255-5100 TELEPHONE (513)		AI
20 20 25 20 04TA 25 25	CARD 1. NDECK = THE NUMBER OF DATA DECKS TO BE RUN SEQUENTIALLY. Iprint The Jalue Sontrolling The Hrite Statements 1 permits Haximum Printout 2 suppresses Aanse-Pair Printing		Ĩ
0914 0917 75 72	CARD 1. NDECK = THE NUMBER OF DATA DECKS TO BE RUN SEQUENTIALLY. Iprint the Jalue Sontrolling the Mrite Statements 1 permits Maximum Printout 2 suppresses Ranse-Pair Printing		A
0	5 HAXIMUM SUFFESSION OF FALMIOU IRPCH= THE VALUE CONTROLLING THE ENTRY INTO THE	Ρ	: AJ
- U U U U	RANGE PAIR COUNTING SUBROUTINE 1 Enter Range Pair Counting Subroutine 2 Skip Rangé Pair Counting Subroutine Format 314	ROGRAM	PPEND
EACH	DATA DECK CONTAINS THE FOLLOWING CARDS	ĹĹĴ	IX
35 CC CARD 1	1. TEST IDENTIFYING INFORMATION Format 8A8	LST	I
4 4 6 7 8 7 8 7 8 7 8 7 8 7 8 7 8 7 8 7 8 7	2. TM = MATERIAL TYPE TYS = TENSILE YIELD STRESS (KSI) EPSD = LCF STRAIN INFERCEPT COFMAN = INVERSE OF COFFIN-MANSON SLOPE ELMOD= MODULUS OF ELASTICITY (KSI) FORMAT ZA8,3F18.5/F10.2		
2 4 4 2 4 5 C 4 5	<pre>3,,6. A(N) N=4,7 (A,B,C ARE COEFFICIENTS OF SECOND (ORDER LEAST SQUARE FIT OF S-N DATA, B(N) (FOR CURVE OF 10**N CYCLES.) B(N) (STMAX = A(N)*STMIN*2 + B(N)*STMIN + C(N)) TITLE1 (TITLE2 IDENTIFIES THE) N (SOURCE OF THE S-N DATA) N (PUNCHED IN COLUMN 72. FOR INFO.)</pre>		
000	MATERIAL TYPE (COLUMNS 73-80. FOR INFO ONLY.) Format 3f18.5,248		
S5 CARD 7	7. C1 (CONSTANTS TO BE USED IN CALGULA-)		1

2DC 6600 FIN V3.0-3674 OPT=1 04/26/24 10.41 22 04/25	(TION OF EQUILIBRIUM PERIOD, ENEP.) (ENEP=C1/(KTSMAX**E1*KTSMEAN**£2))	VIRATION FACTOR USED THE FIRST The program	BLOCKS (NO. OF TIMES TO REPEAT LIST Of LOADS TYPES OF LOADS		KTH STEP (K=1,NLEVEL) 4TIFYING TYPE OF THE KTH LOAD HINIMUM (DECIMAL FRACTION OF TLL) MAXIMUM (DECIMAL FRACTION OF TLL) FCVCLES AT THE KTH LOAD (F13.5,1X)	10 + NBLOCK. HE JJTH BLOCK (J=1,NBLOCK) FLOAD INCLUDED IN JJTH BLOCK (T+ERE WILL BE ONE NN VALUE (ON THE CAR2 FOR EACH DIFFERENT) (TYPE OF LOAD INCLUDED IN THE (JJTH BLOCK)	INT , RNN(200), ITYPE(200) , ELMOD, TYS, EPSO, COFMAN, C1, E2, E1, 10), ENN(200), A (10), B(10), G(10),	TO BE PROCESSED.)	ISEO)		
	E1 (T E2 Format 3F18.5 (E	AKT = STRESS CONCENT TIME THROUGH T FORMAT F18.5	VBLOCK = NUMBER OF B VLEVEL = NUMBER OF L VTYPE = NUMBER OF T FORMAT 3110	FLL = LIMIT LOAD Format F18.5	VLEVEL + 10. THE K THE K THE K THE K THE K THE KTH M THE KT	+ 11,,NLEVEL + 10 JJ NN(JJ,1) = TYPE OF NN(JJ,2) = NN(JJ,) = CORMAT 9I5	NGYG (200), KPMAX, IPRI IN(200), RTMAX(200), R NLEJEL, IRPCM, IPLOT, EI Sumenn, Sumnc Simin (200), Stmax(200	CK, IPRINT, IRPCH ECK 22H DATA DECKS ARE	1 GO 1 O 6 Counting Methods US	L, NDECK	2,13,14,15,16,17,18
		CARD 8.	CARD 9.	CARD 10.	CARDS 11,,	CARDS NLEVEL	COMMON/MSAL/R DIMENSION RTH COMMON/MGORI/I RES(260),AKT; COMMON/MGORZ/I	KEAD(5,4) NDE(READ(5,4) NDE(FORMAT(314) MRITE(6,3) NDU FORMAT(114,14,	IF (IKPCM.66.2 WRITE(6,11) G0 T0 13 WRITE(6,7) FORMAT(25H NO	CONTINUE 00 595 LAN = HRITE(6,5) 50044112,001 557	FORMAT (848)
SAL	0000	00000	300000	000			τ τ)	en te	<u>م</u> و	د 13	۰ σ
ROGRAM											
ă.	;	60	65	70	12	6 2	06	3 2	100	105	

	FORMATTIGHISPECTRUM FROM , 000 INUT OF DATA PECULTAR TO A MATERIAL READ(5,10)TH1,TM2,TYS,EPSO,COFMAN,ELMOD FORMATICA0,545,570.20 FORMATICA0,540,5710.20 FORMATICA0,540,5710.20 FORMATICA0,540,5710.20 FORMATICA0,570,570.20 FORMATICA0,570.20 FORMATICA0,570.20 FORMATICA0,570.20 FORMATICA0,570.20 FORMATICA0,570.20 FORMATICA0,540 FORMATICA0,
	INDUT OF DATA PECULIAR TO A MATERIAL INDUT OF DATA PECULIAR TO A MATERIAL READ(5,10) TH1, TV2, FYS9, COFMAN, ELHOD HRITE(6), 223 TH1, TV2, FYS9, COFMAN, ELHOD HRITE(6), 244) TRICERCET = FI44, 57, 734 TH INVERSE OF COFFI HALTE(6), 244) TRICERCET = FI44, 57, 734 TH INVERSE OF COFFI HRITE(6), 244) TRICERCET = FI44, 57, 734 TH INVERSE OF COFFI HRITE(6), 244) TRICERCET = FI44, 57, 734 TH INVERSE OF COFFI HRITE(6), 244) TRICERCET = FI44, 57, 734 TH INVERSE OF COFFI HRITE(6, 240) HRITE(1, 111LE2, N=4, 7) HRITE(6, 260) HRITE(1, 111LE2, N=4, 7) HRITE(6, 260) HRITE(1, 111LE2, N=4, 7) HRITE(6, 200) HRITE(2, 10, 111, 144, 5H LN + G(1)) HRITE(6, 200) HRITE(2, 10, 111, 144, 5H LN + G(1)) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRITE(2, 200) HRITE(2, 200) HRITE(0, 200) HRITE(6, 200) HRITE(2, 200) HRIT
	<pre>READ(5,10)TM1,TM2,TY5,EPSO,COFMAN,ELHOD FORMAT(14) MATERIAL TYPE</pre>
	<pre>*,F18.5,//23H LCF STRAIN INTERCENT =,F18.5,//31H INVERSE OF COFFI *N-MANSON SLOPE,F18.5,//30H ELASTIC MODULUS =,F18.5) FORMAT(5F18.5,248) FORMAT(5H0GCGFFICIENTS OF SECOND ORDER LEAST SQUARE FIT OF S-N DA FARTE(6,16) FORMAT(5H0GCGFFICIENTS OF SECOND ORDER LEAST SQUARE FIT OF S-N DA FARTE(6,14) FORMAT(5H10,10) FORMAT(5H10,10) FORMAT(5H10,10) FORMAT(5H10,10) FORMAT(5H10,10) FORMAT(5,20) F</pre>
JC 58 55 22 53 56 F8 F9 F7 F3 F3 F8 F7 F3 F3 F8 F7 F3 F8 F7 F8 F7 F8 F8 F8 F8 F8 F8 F8 F8 F8 F8 F8 F8 F8 F	<pre>MAILE(5,10) MAILE(5,10) HRITE(5,10) HRITE(5,10) FORMT(5,10) HRITE(5,20) HRITE(5,20) FORMT(5,10) HRITE(5,20) FORMT(5,10) FORMT(5,10) HRITE(5,20) FORMT(5,10) F</pre>
JC 5 5 5 22 3 5 79 14 32 14 13 14 13 14 13 14 13 10	HRITE(6,18) FORMAT(44H SMAX = A(I)*SMIN**2 + B(I)*SMIN + C(I)) MRITE(6,20) MRITE(6,20) MRITE(6,22) (N,A(N),B(N),C(N),N=4,7) FORMAT(68,22) (N,A(N),B(N),C(N),N=4,7) FORMAT(68,22) (N,A(N),B(N),C(N),N=4,7) FORMAT(68,22) (N,A(N),B(N),C(N),N=4,7) FORMAT(68,22) (N,A(N),B(N),C(N),N=4,7) FORMAT(68,22) (N,A(N),B(N),C(N),N=4,7) FORMAT(68,100,12,3740,5) MRITE(6,56) TITLE1,TITLE2 FORMAT(59,14) C1,E1,F2 FORMAT(56,14) C1,E1,E2 READ(5,14) C1,E1,E2 RAD(5,14) C1,E1,E2 HRITE(6,24) FURMAT(36H0RESIDUAL STRESS RELAXATION FUNCTION) HRITE(6,24) FURMAT(36H0RESIDUAL STRESS RELAXATION FUNCTION) HRITE(6,24) FURMAT(36H0RESIDUAL STRESS RELAXATION FUNCTION) HRITE(6,24) FURMAT(56,28) FURMAT(56,28) FORMAT(4,24) EAR E C1/(KISMAX**E1 * KISMEAN**E2)/) HRITE(6,28) FORMAT(42,4) FORMAT(43,4) HRITE(6,28) FORMAT(42,4) FORMAT(42,4) FORMAT(42,4) HRITE(6,28) FO
10 10 10 10 10 10 10 10 10 10	<pre>FORMAT(5x,5H LFE,10x,5H A(I),14x,5H B(I),14x,5H C(I)) HRITE(6,22)(N,A(N),B(N),C(N),N=4,7) FORMAT(3H 10**,I2,3F18,5) WRITE(6,56) TITLE1,TITLE2 FORMAT(39HOUNNOTCHED COUPON S-N DATA DERIVED FROM/28H INFORMATION *SUPPLIED FROM > 2A8) READ(5,14) C1,E1,E2 HRITE(6,24) REITE(6,24) RRITE(6,24) RRITE(6,28) RRI</pre>
20 7 7 8 00 00 50 7 7 8 00 50 7 7 19 10	<pre>FORMAT(8H 10**,12,3F18.5) WRITE(6,56) TITLE1,TITLE2 FORMAT(39HUNNOTCHED COUPON S-N DATA DERIVED FROM/28H INFORMATION FSUPPLIED FROM :2A8) REAO(5,14) C1,EL,E2 WRITE(6,24) FORMAT(36HORESIDUAL STRESS RELAXATION FUNCTION) HRITE(6,24) FORMAT(42H ENEP = C1/(KTSMAX**E1 * KTSMEAN**E2)/) HRITE(6,28) C1,EL,E2 FORMAT(42H ENEP = C1/(KTSMAX**E1 * KTSMEAN**E2)/) FORMAT(42H ENEP = C1/(KTSMAX**E1 * KTSMEAN**E2)/)</pre>
5 0 5 5 20 5 0 7 7 17 10	<pre>FORMAT(39H0UNNOTCHED COUPON S-N DATA DERIVED FROM/28H INFORMATION *SUPPLIED FROM ,248 *CAOI6,14,) C1,E1,E2 #KIDI66,24, FURMAT(36H0RESIDUAL STRESS RELAXATION FUNCTIDN) #RITE(6,26) FORMAT(42H ENEP = C1/(KTSMAX**E1 * KTSMEAN**E2)/) FORMAT(42H ENEP = C1/(KTSMAX**E1 * KTSMEAN**E2)/) FORMAT(43H+WHERE C1 =,E15.0,9H , E1 =,F10.3,10H AND E2 =,F10.</pre>
+ 9 € 00 N N 00 0+ 11	WRITE(6,24) FURMAT(36HORESIDUAL STRESS RELAXATION FUNCTION) FURMAT(45H ENEP = C1/(KTSMAX**E1 * KTSMEAN**E2)/) MRITE(6,28) C1,E1,E2 FORMAT(13H+WHERE C1 =,E15.8,9H , E1 =,F10.3,10H AND E2 =,F10.
9 8 00 9 7 00 9 1 10	HRITE(6,26) Format(4/24) Enep = C1/(KTSHAX**E1 * KTSHEAN**E2)/) Hrite(6/28) C1,e1,e2 Format(13H+WHERE C1 =,e15.0,94H , E1 =,f10.3,10H AND E2 =,f10.
8 00 7 7 7	WRITE(6,28) CIPELPE2 Format(13H+WHERE C1 =,E15.8,9H , E1 =,F10.3,10H AND E2 =,F10.
<u>о</u>	12*
	2
145	INPUT OF DATA PECULIAR TO A SEQUENCE
	READ(5,65)AKT
60 E	FORTHALT 10.59.22 NBLOCK, JLEVEL, NTYPE
3 47	RETE(6,34) NBLOCK,JLEVEL Format(// 110,334) Times Through Block of.110.64 Loads)
155 35	READ(5,35) TLL FORMAT(F18,5)
33	MRITE(6,33) TLL FORMAT (5 5×13H LOAD LIMIT =,F18,5) FEART (2 5×13H LOAD LIMIT =,F18,5)
160 36	REACTS SOLVEDOMNE JITEENTS ATTANKS STAALS SKMNINS SKALEVEL Format (1422) 14,223 F18.53,123 F18.53,123 F18.53,123) - Datte: 2 - 202
8 8 8	FORMAT(//11H STEP TYPE,10X,6H STMIN,14X,6H STMAX,15X,4H ENN) Herte(6,330) (K ,11YPE(K),RTMIN(K),RTMAX(K),RNN(K),K=1,JLEVEL) Herte(6,330)
165 39	FORMAT(//47H BLOCK TYPE TYPE TYPE TYPE TYPE TYPE TYPE TYPE

233 617 (14-10) 600 53 233 657 (12-0) 600 53 234 657 (12-0) 600 53 235 657 (12-0) 600 53 235 657 (12-0) 600 53 235 657 (12-0) 600 54 235 657 (12-0) 600 54 235 657 (12-0) 600 54 235 657 (12-0) 600 54 235 657 (12-0) 600 54 236 657 (12-0) 600 54 237 657 (12-0) 600 54 238 658 (12-0) 600 54 237 657 (12-0) 600 54 238 658 (12-0) 600 600 238 658 (12-0) 600 600 239 658 (12-0) 600 600 239 658 (12-0) 600 600 240 600 600	233 617 (AA-40) 0000 0000 230 617 (AA-40) 0000 0000 250 517 (AA-40) 0000 0000 251 517 (AA-40) 0000 0000 251 517 (AA-40) 0000 0000 251 517 (AA-40) 010 (A +40) 010 (A +40) 17 100 100 100 100 180 0000 100 100 100 191 17000 100 100 100 192 17000 100 100 100 101 111 1110 100 100 100 101 1110 1100 100 100 100	SUBROUTIN	E COR	č	3DC 6600 FTN V3.0-367A OPT=1	94/26/24	10.40.58.	PAGE	8	
61 201 001	6 200 0000 0000 0000 6 200 0000 0000 0000 6 200 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7 0000 0000 0000 0000 7		220	IF(JA+JB) 230,250,240		CORE	57			
6 540 05013-575614 0001 000 65 530 05013-55614 0001 000 65 530 05013-55614 0001 0001 65 530 05013-55614 0001 0001 71 100 05015-55614 0001 0001 73 15000000000000000000000000000000000000	60 200 00000 0000 61 700 00000 0000 62 700 00000 00000 63 700 00000 00000 70 100 00000 00000 70 100 00000 00000 70 100 00000 00000 70 100 00000 00000 70 100 00000 00000 71 100 00000 00000 71 100 00000 00000 70 00000 00000 00000 70 00000 00000 00000 70 00000 00000 00000 70 00000 00000 00000 70 00000 00000 00000 70 00000 00000 00000 70 00000 00000 00000 70 00000 000000 00000		007	60 TO 290		CORE CORE	5 5 0			
65 75:000	53 75/10.500 7000 7000 64 730 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71 7000 7000 7000 7000 71	60	240	RES(I)=-TYS-ASMIN GO TO 200		CORE	0.0			
65 550	60 50000 50		250	50 10 290 IF(JA) 260.260.270		0035	61			
5 70 </td <td>65 230 5000000000000000000000000000000000000</td> <td></td> <td>260</td> <td>RES(I)=RES(I-1)</td> <td>· · · ·</td> <td>CORE</td> <td>0 Z 0</td> <td></td> <td></td> <td></td>	65 230 5000000000000000000000000000000000000		260	RES(I)=RES(I-1)	· · · ·	CORE	0 Z 0			
65 510 511	65 535 555		270	60 T0 290 DESTIT-ASHEAN		CORE	64			
73 SIGNIT CARLON - SECUTA 44814 Constraints Constraints Constraints 74 100 Constraints Constraints Constraints 75 Constraints Constraints Constraints Constraints 75 Constraints Constraints Constraints Constraints 75 Constraints Constraints Constraints Constraints 76 Constraints Constraints Constraints Constraints 77 Constraints Constraints Constraints Constraints 78 Constraints Constraints Constraints Constraints 79 Constraints Constraints Constraints Constraints 79 Constraints Constraints Constraints Constraints 70 Constraints Constraints Constraints Constraints 71 Constraints Constraints Constraints Constraints 71 Constraints Constraints Constraints Constraints 72 Constraints Constraints Constraints Constraints 73 Constraints Constraints Constraints Constraints 74 Constraints Constraints <td>71 400 STGAIN (ABLIN) - SEG (1) + 450 (4) 00000 00000 00000</td> <td>. 9</td> <td>290</td> <td>SIGHAX (IRAIN) =RES(I) +ASHAX</td> <td></td> <td>CORE</td> <td>65 6</td> <td></td> <td></td> <td></td>	71 400 STGAIN (ABLIN) - SEG (1) + 450 (4) 00000 00000 00000	. 9	290	SIGHAX (IRAIN) =RES(I) +ASHAX		CORE	65 6			
7 400 EXECUTATION SCIENCUS 00000	71 400 Fight Market Fight M			SIGMIN(IRAIN)=RES(I)+ASMIN		CORE	57			
10 10 10 10 10 10 15 10 10 10 10 10 16 17 10 10 10 10 17 10 10 10 10 10 18 10 10 10 10 10 19 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 11 10 10 10 10 10 12 10 10 10 10 10 13 10 10 10 10 10 14 10 10 10 10 10 15 11 10 10 10 10 16 11 10 10 10 10 17 10 10 10 10 10 16 11 10 10 10 10 17 10 10 10 10 10 10 10 <td< td=""><td>71 410 Constanting of the state Constanting of the state 75 410 Constanting of the state Constanting of the state 75 410 Constanting of the state Constanting of the state 75 410 Constanting of the state Constanting of the state 75 410 Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 77 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 77 Constanting of the state Constanting of the state Constanting of the state 77 Constanting of the state Constanting of the state Constanting</td><td></td><td></td><td>KNGTG (IKAIN) = ENN(J)</td><td></td><td>CORE</td><td>58</td><td></td><td></td><td></td></td<>	71 410 Constanting of the state Constanting of the state 75 410 Constanting of the state Constanting of the state 75 410 Constanting of the state Constanting of the state 75 410 Constanting of the state Constanting of the state 75 410 Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 77 Constanting of the state Constanting of the state Constanting of the state 76 Constanting of the state Constanting of the state Constanting of the state 77 Constanting of the state Constanting of the state Constanting of the state 77 Constanting of the state Constanting of the state Constanting			KNGTG (IKAIN) = ENN(J)		CORE	58			
10 <	10 10 10 10 10 10 10 15 10 10 10 10 10 10 16 10 10 10 10 10 10 17 10 10 10 10 10 10 18 17 10 10 10 10 10 19 17 16 10 10 10 10 10 17 17 10 10 10 10 11 17 10 10 10 10 10 12 12 10 10 10 10 10 14 17 10 10 10 10 10 15 10 10 10 10 10 10 16 11 10 10 10 10 10 17 10 10 10 10 10 10 16 11 10 10 10 10 10 17 10 10 10 10 10 10 16 11 10 10 10 10 10 <		40.0	1. (XOURX (C. 1.0) 60 10 410 HQPHC=-AAMAKATVC		CORE	69			
75 4.10 10.10 0.000 7.1 75 4.10 10.40 0.000 7.1 80 0.000 0.000 7.5 0.000 81 0.000 0.000 7.5 80 0.000 0.000 7.5 81 0.000 0.000 7.5 82 0.000 0.000 7.5 83 0.000 0.000 0.000 84 0.000 0.000 0.000 85 0.000 0.000 0.000 86 0.000 0.000 0.000 81 0.000 0.000 0.000 81 0.000 0.000 0.000 82 0.000 0.000 0.000 83 0.000 0.000 0.000 84 0.000 0.000 0.000 85 0.000 0.000 0.000 85 0.000 0.000 0.000 85 0.000 0.000 0.000 85 0.000 0.000 0.000 86 0.000 0.000 0.000 80 0.000 0.000 0.000 80 0.000 0.000 <t< td=""><td>75 7410 Friderin (Ger. Try) to 10 430 0000 77 73 100 Friderin (Ger. Try) to 10 430 0000 77 73 Ergestantic (Ger. Try) to 10 430 0000 77 0000 77 80 Altra (Ger. Try) (Ger. Try) to 10 430 0000 77 0000 77 80 Altra (Ger. Try) (Ger. Try)</td><td>7.0</td><td>5</td><td></td><td></td><td>CORE</td><td>20</td><td></td><td></td><td></td></t<>	75 7410 Friderin (Ger. Try) to 10 430 0000 77 73 100 Friderin (Ger. Try) to 10 430 0000 77 73 Ergestantic (Ger. Try) to 10 430 0000 77 0000 77 80 Altra (Ger. Try) (Ger. Try) to 10 430 0000 77 0000 77 80 Altra (Ger. Try)	7.0	5			CORE	20			
75 100	75 00 0.00 0.00 77 80 00 0.00 0.00 77 81 01 01 0.00 0.00 82 01 01 0.00 0.00 81 01 01 0.00 0.00 82 01 01 0.00 0.00 83 01 01 0.00 0.00 84 17 01 0.00 0.00 85 17 01 0.00 0.00 84 17 01 0.00 0.00 85 17 01 0.00 0.00 86 17 01 0.00 0.00 87 17 01 0.00 0.00 88 17 01 0.00 0.00 89 17 01 0.00 0.00 81 01 01 01 0.00 81 01 01 01 0.00 82 01 01 01 0.00 <t< td=""><td>•</td><td>410</td><td>TF(ASHIN.GF TYS) GO TO 430</td><td></td><td>CORE</td><td>71</td><td></td><td></td><td></td></t<>	•	410	TF(ASHIN.GF TYS) GO TO 430		CORE	71			
75 4.0 CONTOUNT RELAXATION FUNCTION 0.0 CALOULTE RELAXATION 0.0 CALOULTE RELAXATION </td <td>75 410 60000 75 80 000000 75 00000 80 000000 75 00000 81 000000 75 00000 82 000000 75 00000 83 000000 75 00000 84 000000 75 00000 85 000000 75 00000 86 000000 75 00000 86 000000 75 00000 86 000000 75 00000 86 000000 75 00000 86 000000 75 00000 86 000000 75 00000 87 17 17 17 88 17 17 100000 88 17 100000 10000 89 17 100000 10000 89 17 100000 100000 89 1000000000000000000000000000000000000</td> <td></td> <td>420</td> <td>MORESHARMENTYA</td> <td></td> <td>COKE</td> <td>22</td> <td></td> <td></td> <td></td>	75 410 60000 75 80 000000 75 00000 80 000000 75 00000 81 000000 75 00000 82 000000 75 00000 83 000000 75 00000 84 000000 75 00000 85 000000 75 00000 86 000000 75 00000 86 000000 75 00000 86 000000 75 00000 86 000000 75 00000 86 000000 75 00000 86 000000 75 00000 87 17 17 17 88 17 17 100000 88 17 100000 10000 89 17 100000 10000 89 17 100000 100000 89 1000000000000000000000000000000000000		420	MORESHARMENTYA		COKE	22			
75 440 CRESSID: COUNT FRESSID: 0 0 ALOULATE KLANTION FUNCTION COUNT FREAMING 0 ALOULATE KLANTION FUNCTION COUNT FREAMING COUNT FREAMING 0 ADMX=403(05MLN) COUNT FREAMING COUNT FREAMING 0 ADMX=403(05MLN) COUNT FREAMING COUNT FREAMING 1 Transmission COUNT FREAMING 1 TRANSMISSION	75 4.00 CALCULATE RELAXATION FUNCTION 0 ALCULATE RELAXATION FUNCTION CALCULATE RELAXATION FUNCTION 0 ALCULATE RELAXATION FUNCTION CALCULATE RELAXATION FUNCTION 0 ALCULATE RELAXATION FUNCTION CORE 1 ALCULATE RELAXATION CORE 1 ALCULATE COLE					1000	2.			
75 440 DIF=EES(1)EQRES 77 80 CACUUTE RELAATION FUNCTION 0000 77 81 AMATASIGSIASIANA 0000 77 83 AMATASIGSIASIANA 0000 77 84 AMATASIGSIASIANA 0000 77 85 AUSIANASIGSIASIANA 0000 70 86 AUSIANASIGSIANA 0000 70 86 AUSIANASIGSIANA 0000 70 87 AUSIANASIGSIANA 0000 70 88 AUSIANASIGSIANA 7000 7000 89 AUSIANASIGSIANA 7000 7000 80 AUSIANASIGSIANA 7000 7000 80 AUSIANASIANA 7000 7000 80 AUSIANASIANA 70000 7000 80	75 440 0.1F=RES(1)-C0RES 77 0 CALOUATE RELAXITON FUNCTION 0.062 77 0 CALOUATE RELAXITON FUNCTION 0.062 77 1 ADMARTASIASIASTASIA 0.062 77 1 ADMARTASIASIASTASIA 0.062 77 1 ADMARTASIASIASTASIA 0.062 77 1 ADMARTASIASIASIASIASTASIA 0.062 77 1 ADMARTASIASIASIASTASIA 0.062 72 1 TERRANLITIALMARTAL 0.062 72 1 TERRANLIAL		430	EQRES=D.		1200	- t- 			
0 CALCULATE RELAXATION FUNCTION 00000 77 0 ADMAX=BSXGRAMX 00000 20000 20000 0 ADMAX=BSXGRAMX 00000 20000 20000 20000 0 ADMAX=BSXGRAMX 00000 20	0 CALCULATE RELAXATION FUNCTION 000000000000000000000000000000000000	75		DIF=RES(I)-EQRES		CORE	76			
000000000000000000000000000000000000	80 0000X=053(ASTRA) 1000X=053(ASTRA) 1000X=053(ASTRA) 1000X=000X 79 65 4.2 000X=0.00X 50 7(400X=0.111.11)AB4AX=1. 000X 50 90 0.70 AB11 000X 50 51 0.70 AB11 000X 50 51 0.70 AB11 000X 50 51 0.70 AB11 51.500 T0 3B11 000X 51 0.70 AB11 51.500 T0 3B11 51.500 T0 3B1 51 0.70 AB11 51.500 T0 3B11 51.500 T0 3B1 51 0.70 T140 57.510 T0 50 50 51 0.70 T 3B1 000X 50 51 0.70 T140 57.510 T0 50 50 51 0.70 T 3B1 0.70 S1 50 51 </td <td></td> <td></td> <td>CALCULATE RELAXATION FUNCTION</td> <td></td> <td>CORE</td> <td>77 78</td> <td></td> <td></td> <td></td>			CALCULATE RELAXATION FUNCTION		CORE	77 78			
60 000000000000000000000000000000000000	60 000000000000000000000000000000000000		د	A RMAY = A R < A S M > A		CORE	52			
65 442 APREARMANTALIATION 0045 81 1660874.11.1.1.00044.0.1 100040.0.1 100040.0.1 100040 100040 1760874.11.1.1.00044.0.1 1760874.0.1 00040 100040 100040 1760874.0.11.1.1.00044.0.1 00040 100040 100040 100040 1760874.0.11.1.1.00044.0.1 00040 100040 100040 100040 100 000111000 000111000 100040 100040 100040 100 000111000 000111000 00040 100040 100040 100 000111000 000111000 00040 100040 100040 100 000111000 000111000 00040 100040 100040 100 000111000 000111000 00040 100040 100040 100 000111000 00011000 00010 100040 100040 100 0001110000 000110000 00010 100040 100040 100 0001110000 00010 00010 00040 100 100 00010100000000000 00010 000000000000000000000000000000000000	85 442 TF (ABNAKLIT-LI) ABNEALS CORE 81 1 TF (ABNEANLIT-LI) ABNEALS CORE 82 1 TF (ABNEANLIT-LI) ABNEALS CORE 83 1 TF (ABNEANLIT-LI) ABNEANLS CORE 83 1 TF (ABNETLIT-LI) ABNEANLS CORE 93 1 TF (ABNETLIT) ABNEANLS CORE 93 1 TF (ABNETLIT) ABNEANLS CORE 93 1 TF (ABNETLIT) ABNEANLS CORE 93 1 TE (ABNETLIT) ABNEANLS CORE 93 1 TE (ABNETLIT) ABNEANLS CORE 93 1 TE (ABNETLIT) ABNEANLS <	80				CORE	9.0			
85 442 Transmittation and the intervalue of 0 to 446 00000 82 90 445 Frankmittation and the interval of 0 to 446 00000 82 91 445 Frankmittation and the interval and the interval of 0 to 446 00000 82 93 50 10 446 50 00000 82 93 51 51 51 51 52 52 93 51 51 51 51 52 52 93 51 51 51 51 52 52 52 94 411 51 51 51 51 52	B5 Ut2 Transmittation 0000 32 B5 Ut2 Transmittation 0000 32 B4 Transmittation 0000 32 B5 Ut2 Transmittation 0000 32 B6 Ut2 Transmittation 0000 32 B6 Ut2 Transmittation 0000 32 B7 Transmittation 0000 33 0000 33 B7 Transmittation 0000 33 00000 33 B7 Transmittation 0000 0000 33 00000 33 B7 Transmittation 0000 0000 0000 0000 0000 B7 Transmittation 0000 0000 0000	9				CORE	81			
85 442 Transmanning Hardenson Transmanning Hardenson Harden Barnanning Hardenson Harden Barnanning Harden Barnanning Harden Harden Harden	85 442 FIGREMALTILIARDEMANCE 000E 53 91 117 117 114<			TE(DRXAX.III I VADKAVIX		CORE	82			
85 442 FTGRATIN HANDARE LUZ	8 442 Initiality internetion:					CORE	93			~
65 442 ämma and and and and and and and and and an	95 442 000000000000000000000000000000000000			IF (ARMIN-ARMAX) 444.444.442		CORE	84			
90 444 60 T0 446 600 T0 446 444 60 F0 446 600 T0 446 600 T0 446 445 ERFEGIVILOBH*ELABHEAN*E2) FILE FLATINILO FORTALINILO FORTALINILIO FORTALINICALINILIO FORTALINI FORTALINILIO FORTALINI FORTALINICALINILIO FORTALINI FORTALI	90 444 600 446 600 60	85	442	ABM=ABMIN		CORE	85 0			
90 444 60F42014X 50 91 444 60F420140 55 92 177(1701.087*1.40.4) 55 50 93 55 177(1701.087*1.40.4) 55 95 608 510.001 51 50 95 6 0011106 51.1X.515.65.5X.16.4X.161 500 95 6 001046 51.2X.515.65.5X.16.4X.161 500 95 6 001046 51.2X.515.65.5X.16.4X.161 500 96 50 00104 51.2X.515.65.5X.16.4X.161 500 97 0011106 51.1X.51.55.1X.51.64.4X.161 500 500 98 6 01560 500 500 500 98 1741001.54.64.1.51.64.71.51.60 500 500 500 <td>90 WHL ABPEARAX HUL ABPEARAX FERFENT.GE.31G0 TO 351. FERFENT.GE.31G0 TO 351. CONTUNE CONTENT.GE.31G0 TO 351. FERFENT.GE.31G0 TO 351. FERFENT.GE.31G0 TO 350. CONTUNE CONTENT.GE.31G0 TO 351. FERFENT.GE.31G0 TO 350. FERFENT.GE.31G0 TO 350. FERFENT.</td> <td></td> <td></td> <td>GO TO 446</td> <td></td> <td></td> <td>0</td> <td></td> <td></td> <td></td>	90 WHL ABPEARAX HUL ABPEARAX FERFENT.GE.31G0 TO 351. FERFENT.GE.31G0 TO 351. CONTUNE CONTENT.GE.31G0 TO 351. FERFENT.GE.31G0 TO 351. FERFENT.GE.31G0 TO 350. CONTUNE CONTENT.GE.31G0 TO 351. FERFENT.GE.31G0 TO 350. FERFENT.GE.31G0 TO 350. FERFENT.			GO TO 446			0			
90 #46 EFFECt/(ABH*F14.0HERAN*E2) 5000 500	90 #46 ENEFECT/ABHYERTO 0022 002 91 #KITE(0.350) 70.951 0021 001 0021 95 #KITE(0.350) 70.951 0021 001 001 95 FORMAT(667.2134/515.9.5%/15.4%/15) 0022 001 001 95 Calculate Restoual STRESS RELAMITON 0026 9 0008 9 95 Calculate Restoual STRESS RELAMITON 0008 9 0008 9 96 Calculate Restoual STRESS RELAMITON 0008 9 0008 9 96 Calculate Restoual STRESS RELAMITON 0008 9 0008 9 97 Calculate Restoual STRESS RELAMITON 0008 9 0008 9 98 Calculate Restoual STRESS RELAMITON 0008 0008 9 0008 9 99 Calculate Restoual STRESS RELAMITON 0008 0008 9 0008 9 910 SG Calculate Restoual STRESS RELAMITON 0008 0008 9 0008 0008 9 0008 0008 0008 0008 0008 0008 </td <td></td> <td>***</td> <td>ABM=ABMAX</td> <td></td> <td>1400</td> <td>2 C 8 8</td> <td></td> <td></td> <td></td>		***	ABM=ABMAX		1400	2 C 8 8			
90 HTTE(6, 30(0) 534) RETE(6, 350) STOR(10, STCHAX(TRAIN), STCHAX, STCH	90 #FITERINT.eE.350 TO 351 91 #FITERINT.eE.350 TO 351 95 FRESCUD.EGRES.ENNLUD.ENEPLUT.RELNU.SIGMAN(FRAIN), SIGMAN(FRAIN), SIGMAN(FRA		446	ENEP=C1/(ABM**E1*ABMEAN**E2)		1200	0 0 0 a			
90 **KITGAIN, STEMAKI, STEMAKI, STEMAKIALN, STEMIKIKAIN, SCORE 30 350 FORMATGGF7.2:1X; FLS.2:1X; FLS.4X; IG, 4X, IG) STD 351 CONTINUE STD 352 CALCULATE RESIDUAL STRESS RELAXITON STD 353 IRAIN=IRAIN+1 CORE 354 IRAIN=IRAIN+1 CORE 350 IFAIN=IRAIN+1 CORE 360 IFAIN=IRAIN+1 CORE 370 IFFILON CORE 370 IFFILON <td>90 WRISIG: 65: 55: STMAX(ISAIN), SIGMAN(IRAIN), SI</td> <td></td> <td></td> <td>IF(IPRINT.GE.3)60 TO 351</td> <td></td> <td>CORE</td> <td>06</td> <td></td> <td></td> <td></td>	90 WRISIG: 65: 55: STMAX(ISAIN), SIGMAN(IRAIN), SI			IF(IPRINT.GE.3)60 TO 351		CORE	06			
350 FCRUUIS GORES JENNUJ) LUREPJJIFGAIN 32 351 CONTINUE 35 354 CONTINUE 55 355 CONTINUE 55 354 CONTINUE 55 354 CONTINUE 55 355 CALCULATE RESIDUAL STRESS RELAXATION 500E 356 CALCULATE RESIDUAL STRESS RELAXATION 500E 356 CALCULATE RESIDUAL STRESS RELAXATION 500E 356 TEALN=TRAIN+1 500E 360 TEALN=TRAIN+1 500E 360 TF(ABOTF-LIT-S.) 500 370 TF(ABOTF-LIT-S.) 500 371 TF(ABOTF-LIT-S.) 500 370 TF(ABO	350 *KCSJUJFUNCS:ENNUJJFUNCS:ENNUJJFUNCS:ENNUJJFUNCS:ENNUJJFUNCS:ENNUJJFUNCS:ENNUJFENETJJFSJS 371 </td <td>76</td> <td></td> <td>WRITE(6,350) STMAX(J), STMIN(J), SIG</td> <td>MAX(IRAIN), SIGMIN(IRAIN),</td> <td>CORE</td> <td>91</td> <td></td> <td></td> <td></td>	76		WRITE(6,350) STMAX(J), STMIN(J), SIG	MAX(IRAIN), SIGMIN(IRAIN),	CORE	91			
95 C CALCULATE RESIDUAL STRESS RELAXATION STD 1 351 CALCULATE RESIDUAL STRESS RELAXATION CORE 9 100 GC CALCULATE RESIDUAL STRESS RELAXATION CORE 9 100 GC TRAIN=RAIN+1 CORE 9 100 GC TO CORE 9 100 GC TO CORE 10 360 TO GC TO CORE 10 360 TO GC TO CORE 10 360 TF(SOUL).ELE.JO.) GC TO CORE 10 370 TF(1000.*EMN(J).LT.ENEP) GC TO CORE 10 370 TF(1000.*EMN(J).LT.ENEP) GC CORE 10 CORE 10 370 TF(1000.*EMN(J).LT.ENEP) GC TO CORE 10 CORE 10 370 TF(1000.*EMN(J).LE.10.) GC TO CORE 10 370 TF(1000.*EMN(J).LE.10.) GC TO CORE 10 380 NFL4G=10 NFL4G=10 UNMAY=DUMMY22<	95 C CUMINGE 1 95 C CUMINGE 95 95 6 CUMINGE STO 1 100 SG0 COME 95 100 SG0 COME 95 100 SG0 SG0 COME 95 101 SG0 SG0 COME 95 101 SG0 SG0 SG0 COME 95 101 SG0 SG0 SG0 COME 95 101 SG0 SG0 SG0 SG0 COME 101 102 SG0 SG0 SG0 SG0 COME 102 110 SG0 SG0 SG0 SG0 COME 102 102 SG0 SG0 SG0 SG0 SG0 COME 102 111 SG0 SG0 <td< td=""><td></td><td>100</td><td>*KES(J), EQRES, ENN(J), ENEP, J, IRAIN</td><td></td><td>CORE</td><td>32</td><td></td><td></td><td></td></td<>		100	*KES(J), EQRES, ENN(J), ENEP, J, IRAIN		CORE	32			
95 C CALCULATE RESTOUAL STRESS RELAXATION 00RE 94 100 CALCULATE RESTOUAL STRESS RELAXATION 00RE 97 100 TRAIN=TRAIN+1 00RE 97 100 360 TRAIN=TRAIN+1 00RE 97 100 360 10 560 96 100 360 10 560 100 360 10 560 10 560 370 IF(RNUJ).LE.10.) 50 10 370 IF(RNUJ).LE.10.) 50 50 370 IFAG2=10 50 50 105 IFAG2=10 50 50 106 IFAG2=10 50 50 107 00HMY=EUNIV.1 50 50 108 IFAG2=10 50 50 109 00HMY=EUNIV.1 50 50 101 <td>95 C CALCULATE RESIDUAL STRESS RELAXITION 00RE 94 100 RAIN=TRAIN+1 00RE 97 101 S60 TRAIN=TRAIN+1 00RE 97 101 S60 TRAIN=TRAIN+1 00RE 97 101 S60 TRAIN=TRAIN+1 00RE 97 103 G0 TO 560 CORE 98 104 S60 TF(180)FF-LI-5.5) G0 TO 560 CORE 101 105 S60 TF(180)FF-LI-5.5) G0 TO 560 CORE 102 105 TF(180)FF-LI-5.5) G0 TO 560 CORE 103 105 TF(180)F-LI-5.5) G0 TO 560 CORE 103 106 TF(200)-FE-LI0.5) G0 TO 560 CORE 103 107 S60 TF(200)-FE-LI0.5) G0 TO 560 CORE 103 106 TF(200)-FE-LI0.5) G0 TO 560 CORE 103 CORE 103 107 S60 TO 560 TF(200)-FE-LI0.5) G0 TO 560 CORE 103 CORE 103 107 TF(200)-FE-LI0.5) G0 TO 560 CORE 103 CORE 103 108</td> <td></td> <td>000 100 100</td> <td>FUKMAI (6(F/.2,1X), F6.2,1X,E15.8,5X</td> <td>;,I6,4X,I6)</td> <td>STD</td> <td>-1</td> <td></td> <td></td> <td></td>	95 C CALCULATE RESIDUAL STRESS RELAXITION 00RE 94 100 RAIN=TRAIN+1 00RE 97 101 S60 TRAIN=TRAIN+1 00RE 97 101 S60 TRAIN=TRAIN+1 00RE 97 101 S60 TRAIN=TRAIN+1 00RE 97 103 G0 TO 560 CORE 98 104 S60 TF(180)FF-LI-5.5) G0 TO 560 CORE 101 105 S60 TF(180)FF-LI-5.5) G0 TO 560 CORE 102 105 TF(180)FF-LI-5.5) G0 TO 560 CORE 103 105 TF(180)F-LI-5.5) G0 TO 560 CORE 103 106 TF(200)-FE-LI0.5) G0 TO 560 CORE 103 107 S60 TF(200)-FE-LI0.5) G0 TO 560 CORE 103 106 TF(200)-FE-LI0.5) G0 TO 560 CORE 103 CORE 103 107 S60 TO 560 TF(200)-FE-LI0.5) G0 TO 560 CORE 103 CORE 103 107 TF(200)-FE-LI0.5) G0 TO 560 CORE 103 CORE 103 108		000 100 100	FUKMAI (6(F/.2,1X), F6.2,1X,E15.8,5X	;,I6,4X,I6)	STD	-1			
95 C CalcULATE RESIDUAL STRESS RELAXATION 95 100 C CALCULATE RESIDUAL STRESS RELAXATION 000000000000000000000000000000000000	95 C CALCULATE RESIDUAL STRESS RELAXATION CORE 95 100 TRAIN=IRAIN+1 CORE 95 100 560 TRAIN=IRAIN+1 CORE 95 100 560 TO 560 CORE 95 100 560 TAIN=IRAIN+1 CORE 95 100 560 TO 560 CORE 95 370 IF(1000.*EGN(J)LT.ENEP) GO TO 560 CORE 101 360 IF(ABOFF-110.) 60 TO 560 CORE 102 370 IF(1000.*EGN(J)LT.ENEP) GO TO 560 CORE 102 370 IF(1000.*EGN(J)LT.ENEP) GO TO 560 CORE 102 380 NEAG=10 CORE 102 105 NEAGE=10 CORE 102 106 UNHY=ERAIN1 CORE 102 110 UNHY=ERAIN1 CORE 102 110 MFLAG=1 NEAGE=1 102 110 UNHY=ERAIN1 CORE 102 110 UNHY=ERAIN1 CORE 102 110 NFLAGE=N 470.460.460		- 			CORE	34			
C IRAIN=IRAIN+1 100 360 TO 360 50 TO 560 50 TO 560	C RAIN=FRAIN+1 10 10 10 10 10 10 10 10 10 10 10 10 10 1	95	0	CALCULATE RESTOUAL STRESS RFIAY	ATTON		35			
100 360 TRAIN=IRAIN+1 96 100 560 10 560 570 10 560 96 570 100 560 96 570 1500 560 96 570 15100 560 008 570 151000 560 008 570 151000 560 008 570 161000 560 008 570 161000 560 008 570 161000 560 008 570 16500 560 008 570 166 105 500 570 166 105 500 570 166 105 500 105 104 500 508 106 101 500 508 508 110 100 500 508 508 110 100 500 508 508 110 100 500 508 508 110	100 IRAIN=IRAIN+1 100 560 100 560 10 560 60 10 560 370 IF(ABDIF-LICS-) GO TO 560 001 370 IF(ABDIF-LICS-) GO TO 560 007 380 NFLAGE=10- 001 105 NFLAGE=10- 007 106 IF(OUMMY-ENEP) 470,460,460 450 UNMY-ENEP) 470,460,460 110 NFLAGE=NFLAG+1 007 110 NFLAGE=NFLAG+1 007		с			1000	0.0			
100 560 100 560 560 370 17 (1000.*ENN(J).17.ENEP) GO TO 560 500 500 500 370 17 (1000.*ENN(J).17.ENEP) GO TO 560 500 500 500 500 370 17 (1000.*ENN(J).11.ENEP) GO TO 560 500 500 500 500 500 380 17 (1000.*ENN(J).E.10.) 500 TO 560 500 500 500 500 105 17 (1000.*ENN(J).E.10.) 500 TO 560 500 500 500 500 106 17 (1000.*ENU(J).E.10.) 500 TO 560 500 500 500 500 105 106 100 500 500 500 500 500 107 106 100 500 500 500 500 500 108 110 100 100 500 500 500 500 500 110 110 110 100 500 500 500 500 500 500 500 110 110 110 110 110 500	100 60 T0 560 00 560 370 IF(1000.*ENNUJ).LT.ENEP) GO TO 560 00 CORE 370 IF(1000.*ENNUJ).LT.ENEP) GO TO 560 00 CORE 370 IF(1000.*ENNUJ).LT.ENEP) GO TO 560 00 CORE 380 NFLAG=0 00 CORE 103 105 IF(1000.*ENNUJ).LT.ENEP) GO TO 560 00 CORE 103 40 IF(ENNUJ).ET.ENEP) GO TO 560 00 CORE 103 105 IF(ENNUJ).ET.ENEP) GO TO 560 00 CORE 103 106 IF(AGC=0 00 CORE 103 107 NFLAG=10 00 CORE 103 108 IF(AGC=10) 00 CORE 105 109 IF(OUMMY-ENEP) 470,460,400 CORE 107 110 NFLAG=NFLAG+1 CORE 103 110 NFLAG=NFLAG+1 CORE 110			IRAIN=IRAIN+1		CORF				
100 60 T0 360 60 T0 560 60 T0 560 360 IF(1000*ENNIJ).LT.ENEP) GO T0 560 002 101 370 IF(1000*ENNIJ).LT.ENEP) GO T0 560 002 103 105 IF(1000*ENNIJ).LT.ENEP) GO T0 560 002 103 105 IF(1000*ENNIJ).LT.ENEP) GO T0 560 002 103 105 IF(1000*ENNIJ).ENEP) GO T0 560 002 103 105 IF(2000*ENNI) 002 103 106 IF(2000*ENU) 002 103 107 UNHYERMUJ 0002 105 110 450 IF(0UMYENU) 002 002 110 MFLAGE=NLAGE1 000000000000000000000000000000000000	10 60 T0 360 200 500 100 360 10 F(1000.*ENN(J).LT.ENEP) 60 T0 560 370 171000.*ENN(J).LT.ENEP) 60 T0 560 102 370 171000.*ENN(J).LT.ENEP) 60 T0 560 002 102 102 370 171000.*ENN(J).LT.ENEP) 60 T0 560 002 102 105 370 171000.*ENN(J).LT.ENEP) 60 T0 560 002 105 370 171000.*ENN(J).LE.10.) 50 T0 560 002 106 002 003 103 107 002 002 103 108 107 002 002 109 002 107 002 101 000000000000000000000000000000000000			ABOIF=ABS(DIF)		CORF				
100 50 10 50 <t< td=""><td>100 560 10 560 560 560 370 17(1000.*ENN(J).LT.ENEP) GO 50 102 50% 102 370 17(1000.*ENN(J).LT.ENEP) GO 50 50% 103 50% 103 370 17(1000.*ENN(J).LT.ENEP) GO 50 70% 10 50% 103 380 NFLAG=0 50 50 500 50% 103 105 NFLAG=10 10 50% 105 50% 105 110 NFLAG2=10 10 50% 50% 105 50% 105 106 NFLAG2=10 10 10 50% 50% 50% 105 110 NFLAG2=10 170 50% 50% 107 50% 107 110 NFLAG2=10 111 50% 50% 106 50% 106 110 0UMMY=ENKLON 450 450 450 50% 50% 106 110 NFLAG=NFLAG41 0UMMY=ENKLON 50% 50% 100% 50% 50% 50%</td><td></td><td></td><td>GO TO 360</td><td></td><td>CORE</td><td>100</td><td></td><td></td><td></td></t<>	100 560 10 560 560 560 370 17(1000.*ENN(J).LT.ENEP) GO 50 102 50% 102 370 17(1000.*ENN(J).LT.ENEP) GO 50 50% 103 50% 103 370 17(1000.*ENN(J).LT.ENEP) GO 50 70% 10 50% 103 380 NFLAG=0 50 50 500 50% 103 105 NFLAG=10 10 50% 105 50% 105 110 NFLAG2=10 10 50% 50% 105 50% 105 106 NFLAG2=10 10 10 50% 50% 50% 105 110 NFLAG2=10 170 50% 50% 107 50% 107 110 NFLAG2=10 111 50% 50% 106 50% 106 110 0UMMY=ENKLON 450 450 450 50% 50% 106 110 NFLAG=NFLAG41 0UMMY=ENKLON 50% 50% 100% 50% 50% 50%			GO TO 360		CORE	100			
360 IF(ABDIF+LT.5.) G0 T0 560 000000000000000000000000000000000000	360 IF(ABDIF+LT.5.) G0 T0 560 CORE 103 370 IF(1000+ENNU).LT.ENEP) G0 T0 560 CORE 103 105 IF(1000+ENU).LE.10.) G0 T0 560 CORE 103 380 NFLAG2=10 CORE 104 380 NFLAG2=10 CORE 105 105 NFLAC2=10 CORE 105 107 UMMY=ENVI-1 CORE 105 108 IF(OUMY-ENCP) 450,460 CORE 106 110 NFLAG=NHMYZ2 470,460,460 CORE 106 110 NFLAG=NFLAG+1 CORE 111 CORE 111	100		GO TO 560		CORE	101			
370 If(1000.*ENN(J).LT.ENEP) G0 T0 560 CORE 103 105 If(ENN(J).LE.10.) G0 T0 560 CORE 104 380 NFLGG=10 CORE 105 381 NFLGG=10 CORE 107 450 IF(OUMYY-ENEP) 470,460,460 CORE 107 450 UNMYY=DUMMYZ2. CORE 109 CORE 110 110 NFLGG=NFLAG41 CORE 110 CORE 111	370 If(1000.*ENN(J).LI.ENEP) G0 G0 103 105 380 NFLAGE-10.1) G0 T0 560 CORE 104 380 NFLAGE-10 G0 T0 560 CORE 105 380 NFLAGE-10 G0 T0 560 CORE 105 380 NFLAGE-10 G0 T0 CORE 105 0104M7EDNATO G0 CORE 105 CORE 107 450 DUMMYEDNATZZ CORE 106 106 110 NFLAGENFLAGA1 CORE 106 106 110 NFLAGENFLAGA1 CORE 111 CORE 111		360	IF(ABDIF.LT.5.) GO TO 560		CORE	102			
105 If(ENN(J).LE.10.) G0 T0 560 CORE 104 105 380 NFLAG=0 CORE 105 00 NFLAG=10 CORE 105 CORE 105 10 NFLAG=10 CORE 107 CORE 107 110 450 IF(OUMMY-ENEP) 470,460,460 CORE 110 CORE 110 110 NFLAGENFLAG41 CORE 110 CORE 110 CORE 111	105 380 IF(EAN (J).LE.10.) 60 T0 560 500 106 000000000000000000000000000000000000		370	IF(1000.*ENN(J).LT.ENEP) GO TO 560		CORE	103			-
105 360 NFLAG=0 105 NFLAGE=10 000000000000000000000000000000000000	105 380 NFLAG=0 105 NFLAGE=10 106 IRAIN-1 110 NFLAGE=10 110 NFLAGE=NFLAG+1 110 NFLAGE=NFLAG+1 110 NFLAGE=NFLAG+1	1		IF(ENN(J).LE.10.) 60 TO 560		CORE	104			
Intracestu 106 Intracestu 107 Intracestu 000000000000000000000000000000000000	NFLAGERIU-1 CORE 106 IRAIN=IRAIN-1 URMYTENKUJ CORE 107 0UMMY=ENKUJ CORE 108 CORE 108 450 IF(DUMMY=ENEP) 470,460,460 CORE 109 110 NFLAG=NFLAG+1 CORE 110 CORE 110	1.15	26 U			CORE	105			
110 NFLAGENFLAGATION CORE 107 CORE 108 CORE 108 CORE 109 CORE 109 CORE 110 CORE 110 CORE 110 CORE 110 CORE 110 CORE 110 CORE 110 CORE 110 CORE 110	110 NFLAGENFLAG. 110 NFLAGENFLAG. 110 NFLAGENFLAG. 110 SC	104		7 11 40 4 1 1 4 4 4 4 4 4 4 4 4 4 4 4 4 4		CORE	106			
450 10 450 100 110 460 000MMY=20WMY/22 100 110 NFLAG=NFLAG+1 CORE 110 110 NFLAG=NFLAG+1 CORE 111	450 IF(CUMMY-ENEP) 470,460,460 460 DUMMY=DUMMY/2, 470,460,460 460 DUMMY=DUMMY/2, 109 00xe 110 00xe 111 00xe 111			T-NTAXT-NTAXT		CORE	107			
110 to DUMMY-ENEP) 470,460,460 CORE 109 CORE 110 NFLAGENFLAG+1 CORE 111 CORE 111	110 IF CUMMYTENEP) 470,460,460 110 VFLAGENFLAG+1 CORE 110 111 CORE 111 CORE 111		2 1			CORE	108			-
110 CORE 110 CORE 111 CORE 111 CORE 111	110 CORE 110 CORE 111 CORE 111			I T (UUTAT = ENE 7) 4/U ,40U ,46U		CORE	109			
		110		NFLAGENFLAG+1		CORE	110			
						CORE	111			

.

ī

1

0								• • • •			
1941										·	
*****	112 1115 1116 1116 118	121 121 122 122 122 122 122 122 122 122	125 126 128 128	130 131 132 133	135 136 1376	0 - 1 - 1	1777 1777 1777	8 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9	15 155 155 155 155 155 155 155 155 155	160 152 164 164 165	156
•	CORE CORE CORE CORE CORE CORE CORE CORE	CORE CORE CORE CORE CORE CORE	CORE CORE CORE CORE CORE	CORE Core Core Core Core	CORE CORE CORE	CORE CORE CORE CORE CORE	CORE CORE	CORE CORE CORE CORE	CORE CORE CORE CORE CORE CORE CORE CORE	CORE CORE CORE CORE CORE CORE CORE	CORE
	G0 T0 450 CYCINT=DUMMY/10. D0 500 K=1,10 DECK=FLOAT(K) EN(K)=CYCINT*DECK IF(K,EQ.1) G0 T0 490 EX(K)=EXP(+2.303*EN(K-1)/ENEP)+EXP(-2.303*EN(K)/ENEP)	60 T0 500 C K(K)=1.+EXP(-2.303*EN(K)/ENEP) C ONTINUE IF(NFLAG.EQ.0) GO TO 530 NFLAG2=NFLAG+10 DO 520 K=11,NFLAG2	ЕN(K)=2.*DUMMY EX(K)=2.*P(-2.303*EN(K-1)/ENEP)+EXP(-2.303*EN(K)/ENEP) DUMMY=2.*DUMMY Continue Continue DO 559 K=1,NFLAG2	SIGMAX(IRAIN)=ASMAX+EQRES+DIF+EX(K)/2. SIGMIN(IRAIN)=SIGMAX(IRAIN)-ASMAX+ASMIN RNCYC(IRAIN)=EN(K) IF(K,EQ.1) G0 T0 540 NCYC(IRAIN)=RNCYC(IRAIN)-EN(K-1)	<pre>LF(IPKLN:66.5)60 T0 551 HRITE(6,550)SIGMAX(IRAIN),SIGMIN(IRAIN),RNCYC(IRAIN),IRAIN FORMAT(16H RELAXATION ,2(F7.2,1X),16X,F6.2,31X,I6) CONTINUE</pre>	ICALN=ICAIN+I CONTINUE CONTINUE Res(1)=EQRES+DIF+EXP(-2.303*ENN(J)/ENEP) Continue Res(1)=Res(J)	IN=IRAIN-1 ************************************	MODULE III CYCLE COUNTING TECHNIQUE ************************************	CALL SUBROUTINE TO RANGE PAIR COUNT SPECTRUM If(IRPCM.GT.1)GO TO 591 Call RPCM(IN) GO TO 592 Continue Continue	KPMAX=IN Continue ************************************	DAMAGE ACCUMULATION CALCULATION
	470	490 500 510	520 530	5 47 D	550 551	559 560 570 569	000	ວບັບບັບ	50 CC 591	20000 20000	ပ
	115	120	125	130	135	140	145	150	155	160	165

ins	BROUTINE CC	COC 6600 FIN V3.0-367A OPT=1 04/26/74 10.4	1.58. PAGE	ŧ.
	000	102日 11日 11日 11日 11日 11日 11日 11日 11日 11日 1		
170	53	IF(IPRINT.GE.3)GO TO 552 WRITE(6,53) FORMATT//60H LOCAL STRESSES AND PLASTIC STRAINS W/RESULTING FATIGU CORE * Life//10X,4HSTEP,10X,14HPLASTIC STRAIN,10X,10HMAX OR MIN,15X,6HD CORE *AAAGF5)		
175	22 22 22 22 22 22 22 22 22 22 22 22 22	CORE 17 CONTINUE CORE 17 CALCULATE DAMAGE FROM PLASTIC STRAIN CYCLES CORE 17 CORE 17 CORE 17		
180	532	SUMDEL=0		
185	533	PLSTRA (JKL) = AA*PLSTRA (JKL) CYCLES= (PLSTRA (JKL) / EPSD) **COFMAN DAM=1 * / CYCLES SUMNC=SUMDCELADM SUMNCESUMDCELADM CORE 10 CORE		
190	535 199	IF(IPRINT.GE.3) GO TO 531 IF (AA) 535,535,537 WRITE(6,199)JKL,PLSTRA(JKL),DAM FORMAT(10X,I4,12X,F10.5,15X,34MIN,15X,E14.6) GO TO 531 GO TO 531 CORE 19 CORE 10 CORE 1		·
195	537 219 531 541	WRITE(6,219)JKL,PLSTRA(JKL),DAM FORMAT(10X,14,12X,F10.5,15X,3HMAX,15X,E14.6) CONTINUE MRITE(6,541) SUMDEL FORMAT(6,541) SUMDEL FORMAT(6,542) SUMDEL FORMAT(5,542) SUMDEL CORE 19 FORMAT(5,542) SUMDEL CORE 10 FORMAT(5,542) SUMDEL CORE 10 FORMAT(5,542) SUMDEL		•
200	13 536	WRITE (6,13) WRITE (6,13) FORMAT (/16%,15H SIGMAX SIGMIN,18%,6H RNCYC,20%,25H CYCLES CORE 20 * ENN/CYC) CORE 20 CONTINUE CORE 20	:	:
205	0000	CALCULATE ELASTIC CYCLE DAMAGE FROM LEAST SQUARE FITTED S-N DATA CORE 20 (MODIFIED GOODMAN DIAGRAM FORMAT) CORE 20 CORE 20 DO 600 JKL=1.KPMAX		
210	534 310 410	TTYS=TYS/5. TF (SIGMAX(JKL)-SIGMIN(JKL).LT.1.6+TYS)GO TO 310 CYCLES=10.**4. GO TO 340 CYCLES=10.**4. CORE 21 CORE		
215	350	UTULES=1U		:
220	332	CONTINUE CORE 22 IF (R(7)*R(4)) 338, 338, 334, CORE 22		

8 8 9 5 5 5 5 5 5 5 5 5 5 5 5 5	9 4 9 9 4 5 9 7 8 9 7 9 7 9 7 9 7 9 7 9 7 9 7 9 7 9			
00000000000000000000000000000000000000	00000000000000000000000000000000000000			
ABR7=ABS(R(T)) ABR4=A3S(R(4)) IF(ABR7.LE.ABR4) G0 T0 336 ZYPO=4.+R(4)/(R(4)-R(5)) G0 T0 339 EXPO=7.+R(7)/(R(6)-R(7)) G0 T0 339 EXPO=7.+R(7)/(R(6)-R(7)) G0 T0 339 EXPO=7.+R(5)+R(6)+R(7)+*2 EXPO=7.+R(5)+R(5)+R(5)+7.+R(7)+*2 SUMR=R(4)+*2-R(5)+R(5)+8-2+R(7)+*2 SUMR=R(4)+*2-R(5)+8-2+R(7)+*2 SUMR=R(4)+*2-15.+R(5)+8-2+R(7)+*2 SUMR=L=R(4)+5.+R(5)+8-2+6.+R(6)+7.+R(7)+*2 SUMR=L=R(4)+5.+R(5)+8-2+6.+R(6)+7.+R(7)+*2 SUMR=L=R(4)+8-2-2.+R(5)+7.+R(7)+*2 DEL1=4.+SUMR2*SUMR4-4.+SUMR3*2 DEL1=2.+SUMR2*SUMR3-SUMR4*3 DO1=22.+SUMR2*SUMR4-22.+SUMR3+2 DO1=22.+SUMR2+SUMR4-22.+SUMR3+2	D02=SUMR2'SUMRA'SUMRN-SUMR4+SUMRN D03=SUMR2'SUMRN-SUMRN-SUMRA+SUMRN EXPG=(001+002+003)/(0EL1+0EL2+0EL3) CYCLES=10.**EXPO IF(EXP0.LE.4.)CYCLES=10.**4. ENNCYC=SUMRC+ENNCYC SUMRC=SUMRC+ENNCYC SUMRC=SUMRC+ENNCYC SUMRC=SUMRC+ENNCYC IF(IPRINI.6E.3)G0 T0 600 HRITE(6,599) SIGMAX(JKL),SIGMIN(JKL),RNCYC(JKL),CYCLES,ENNCYC FORMT(16X,2(F7.2,1X),16X,F6.0,17X,2(E15.6,1X)) CONTINUE HRITE(6,593) SUMDEL HRITE(6,593) SUMDEL HRITE(6,575) SUMDEL FORMT(/69X,18H TOTAL ENN/CYC =,E15.8) HRITE(6,575) SUMNC FORMT(/69X,18H TOTAL ENN/CYC =,E15.8) HRITE(6,575) SUMNC FORMT(/69X,18H TOTAL ENN/CYC =,E15.8)			
9 9 9 1 3 3 3 3 4 4 3 5 4 4 3 4 4 4 3 4 4 4 3 4 4 4 3 4 4 4 4 4	339 340 599 593 593 593 593			
525 530 530	245 245 250			
	334 ABR7=ABS(R(7)) 50 T0 336 222 335 EY(4) (R(4) - R(5)) 50 T0 336 224 335 EYPO=4+R(7) / (R(4) - R(5)) 50 T0 336 225 335 EXPO=7+R(7) / (R(6) - R(7)) 50 T0 336 225 336 EXPO=7+R(7) / (R(6) - R(7)) 50 T0 336 225 335 EXPO=7+R(7) / (R(6) - R(7)) 50 T0 339 50 T0 339 50 T0 339 SUMR2=R(4) + F2 + R(5) + R(7) + R(7) + R(7) 50 C0 RE 226 338 SUMR2=R(4) + F2 + R(5) + R(7)	334 ARR7-AGS (R(1)) 000000000000000000000000000000000000	33. ABRT-AGS(R(1)) FERREAG	33 Amenange (k(1)) (k) (k) (k) (k) (k) (k) (k) (k) (k) (k)

.

| | | | | | | | |

 |

 | |

 | | |

 | | | | | |

 | | | | | |

 | |

 | | |

 | | | | |
 | | | | | · |
 |
|--------------|---|--|--|--|--|--|---
--
--
--
--

--
--
--
--
---|---
--
--
--
--
--|---|--
--
--
--
---|--
--|--|--|--
--
--
--
---|--|--|---|--|---
--
--
--|---
--
--
--|--|---
--
---|--
---|---|---|--|---|--
--|---|---|---|
| | | | | | | | |

 |

 | |

 | | |

 | | | | | |

 | | | | | |

 | |

 | | |

 | | | | |
 | | | | | |
 |
| 256 | 257 | 259 | 251
262 | 0 0
0 0
0 0 | 265 | 266 | 268
268 | 269
269

 | 270

 | 271 | 212

 | 276 | 275 | 276
277

 | 27.8 | 279 | 290 | 281 | 283 | 284

 | 285
286 | 287 | 288 | 239 | 290 | 1 2 2 4

 | 293 | 594

 | 295 | 236
287 | 298

 | 239 | 300 | 301 | 303 | 4 0 M
 | 305 | 305
307 | 308 | 309 | 210 |
 |
| CORE | CORE
ER CORE | CORE | CORE
CORE
I | CORE | CORE | CORE | | CORE

 | CORE

 | CORE | CURE

 | CORE | CORE | CORE
CORE

 | CORF | CORE | CORE | CORE | CORE | CORE

 | CORE | CORE | CORE | CORE | CORE | 002

 | CORE | CORE

 | CORE | CORE
DOCE | CORE

 | CORE | CORE | C0KP | CORE | CORE
 | CORE | CORE
CORE | CORE | CORE | CURE |
 |
| | METHOD TO GEN | | 1 SM | | DIMENSION | NPKS + KK | IN INLA TALA | NPKS + KK

 | NPKS + KK

 | NPKS + KK | 24NPK5

 | NPKS + KK | NPKS + KK | NPKS + KK

 | | 66 | | | |

 | | | 04,KIND | | |

 | |

 | | TINDIT LOAD S |

 | | IINIMUM,13X, | T = 1 NDVC) | ICULUIT - T |
 | | INS AND VALLET | | | |
 |
| R PCH (NPKS) | M EMPLOYS THE RANGE PAIR CYCLE COUNTING | COLVER A GIVEN LUAD SPECIAGA | PROGRAM ARAYS
(INFORMATION NEFDED IN CHANGE DIMENSIO | | DEAXS OF THE INDUT CAR FORMER | = THE NUMBER OF ADJITIONAL CVCLES LEVEL | WHICH THE PROGRAM WILL GENERATE | VALLEYS OF THE INPUT LOAD SPECTRUM

 | K COUNTERS OF THE PEAKS AND VALLEYS

 | SIEP NUMBERS OF THE INPUT SPECTRUM
Destruit sourcround | STEP NUMBERS OF ELFMENTS IN REA

 | RANGE PAIR COUNTED CYCLES | K COUNTERS OF THE CYCLES OF THE | UNSURIEU ANALTSIS SPECIKUM
Step numbers of Alements of the

 | UNSORTED ANALYSIS SPECTRUM | VALUES OF NSTEP(J) SUCH THAT RN2YC(J) | IS < 1.0 AND VALUES OF NSTEP(J) SUCH
THAT STEMAY (1-4) - STEMAY (1 AND | SIGHIN(J-1) = SIGHIN(J) | | /DNCYC/2001./00000 100101

 | 1/SIGMAX(200),SIGMIN(200) | 2/NSTEP(200), LR, KMAX, KMIN, K31 | R/RES(450), INDEX(450), IND1, IND2, IND3, IND | / UTGLE (200, 2) , KNECTG (200) , NNSTEP (200)
5 / 1 1100 | E/L¢LINU
SAVE(99).TITLE(8) |

 | 1, NPKS |

 | E.2)60 TO 103
Nevs | 60HTHE NUMBER OF PEAKS OR VALLEYS IN THF |

 | | 5HSIGMA/31X,4HSTEP,13X,7HMAXIMUM,16X,7HM
UNTER X/1 | (NSTEP(I), SIGMAX(I), SIGMIN(I), PNCYC(I), | I5,10X, 213.6,10X, E13.6,10X, F10.5) |
 | H THE LOAD SPECTRUM . BUIL SUIT THOSE SEA | S LESS THAN 1.0 | | | |
 |
| SUBROUT INE | THIS PROGRA | | | | ARRAY NAME
Sigmay | XX | | SIGMIN

 | RNCYC

 | | INDEX

 | CYCLE | RNECYC | NNSTEP

 | | ISAVE | | | | COMMON/MSAI

 | COMMON/MUEC | COMMON/MDEC | COMMON/MDEC | | I NOISNEWIO | 0 = HONOH = 0

 | DO 8000 I = | NSTEP(I) =

 | IF (IPKINT.G | I FORMAT (1HD, | ICTRUM = ,IS

 | WRITE(6,22) | L LUKMAI (BUX)
1 AHCO | - WRITE (6.25) | FORMAT(29X, | CONTINUE
 | SORT THEORIG | COUNTER K I | н
н | | • |
 |
| د | ა ი ი | 000 | 000 | 0 | 00 | 00 | сı | U (

 | .

 | 20 | 0

 | o | ט נ | 00

 | J | ບເ | 50 | с
O | 0 | 5

 | | | | | | 6666

 | | 8000

 | | 20 |

 | | 77 | | 25 | 103
 | . C | | c | · · · · · · · · · · · · · · · · · · · | |
 |
| | | ŝ | | | 10 | | | Ľ

 | 12

 | |

 | | 20 |

 | | 36 | 23 | | | 30

 | 1 | | | 35 | 2 |

 | |

 | 3
\$ | |

 | 45 | | | : | 50
 | 3 | | | 55 | |
 |
| | SUBROUTINE RPCM(NPKS) CORE 256 | SUBROUTINE RPCH(NPKS)
C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257
C ANALYSTS SPECTADINE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 258 | SUBROUTINE RPCH(NPKS)
C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 258
C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM
C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM
C CORE 259
C CORE 250 | SUBROUTINE RPCH(NPKS)
CORE 256
C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259
C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM
CORE 259
CORE 250
CORE 2 | SUBROUTINE RPCH(NPKS)
CORE 256
C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259
C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM
CORE 259
CORE 250
CORE 2 | SUBROUTINE RPCH(NPKS)
CORE 256
C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259
C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM
CORE 259
CORE 259
CORE 259
CORE 250
CORE 2 | SUBROUTINE RPCH(NPKS)CORE256CTHIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE257CANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUMCORE259CCCORE259CCCORE259CCCORE259CCCORE259CCCORE259CCCORE250CCORE251CORECCCORE251CCORECORE253CCORECORE253CCORECORE253CCORECORE253CCORECORE253CCORECORE253CCORECORE253CCORECORE253CCORECORE253CCORECORE253CCORECORE254CCORECORE255CCORECORE255CCORECORE255CCORECORE255CCORECORE255CCORECORE255CCORECORE255CCORECORE255CCORECORE255CCORECORE255CCORECORE255CCORECORE255CCO | SUBROUTINE RPCH(NPKS)CORE256CTHIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE257CTHIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE259CANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUMCORE259CCCCORE259CCCCORE259CCCCORE250CCCCORE251CCCCORE251CCCCORE253CCCCORE253CCARRAY NAMEDEFINITIONCORE263CCARRAY NAMEDEFINITIONCORE263CCCCCORE263CCCCCORE263CCCCC264CCCCC265CCCCC265CCCCC265CCCCC265CCCCC265CCCCC265CCCCC265CCCCCCCCCCCCCCCCCCCCCCCCCCC </td <td>SUBROUTINE RPCH(NPKS)CORE2565THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2575CANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUMCORE2595CCCORE2596CCORE259CORE2597CCCORE259CORE2596CCCORE250CORE2507CORECORE250CORE2517CCORECORE253CORE2536CCCORECORE253CORE2537CCORECORECORE253CORE2537CCCORECORE253CORE2537CCCCORE253CORE2537CCCCORE253CORE2537CCCCCORE253CORE7CCCCC253CORE7CCCCC254CORE2557CCCCCC254C7CCCCCC255C7CCCCCC255C7CCCCCC256C7CCCC<td>SUBROUTINE RPCH(NPKS)SUBROUTINE RPCH(NPKS)CORE2565THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER2572586CORE2592592597NALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUMCORE2597CORE2592592597RAAY NAMEPROGRAM ARAYSCORE25110CARAY NAMEDEFINITIONCORE2535SIGMAXPEAKS OF THE INPUT LOAD SPECTRUMDIMENSIONCORE2536SIGMANNUMER A DOTITIONAL CYCLES (EXCLUDING INPUT CYCL2662665SIGMINVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2566SIGMINVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2567CONCE2562572662667CONCE2562662662667CONCE2562662662667CONCE2562662662667CONCE257000000000000000000000000000000000</td><td>SUBROUTINE RPCH(NPKS)SUBROUTINE RPCH(NPKS)CORE2565CTHIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2595CCCORE2596CCORE2592596CCORE2592597CCORE2592597CCORE2592597CORECORE2592517CORECORE2502537CORECORE2532637CORECORE2632637CORECORE2642647CORECORE2652647CORECORE2652647CORECORE2642657CORECORE2662667CORECORE2667CORE2662667CORE2662667CORE2662667CORE2662667CORE2662667CORE2662667CORE2662707CORE2662707CORE2662707CORE2662707CORE2662707CORE2702027CCORE2667CCC7CCC<td>SUBROUTINE RCHINPKS)SUBROUTINE RCHINPKS)CORE2565THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2576THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2596ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUMCORE2597CORE259CORE2597CORE259CORE2597CORE259CORE2597CORE261CORE2597CORE261CORE2627CORE261CORE2637SIGMAXDEFINITIONDIMENSIONS)CORE2637SIGMAXPEAKS OF THE INPUT LOAD SPECTRUMDIMENSIONCORE2657SIGMAXVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2657SIGMINVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2657SIGMINSTEP NUMBERS OF THE INPUT SPECTRUMNPKS + KKCORE2517STEP NUMBERS OF THE INPUT SPECTRUMSTAPKSCORE2717STEP NUMBERS OF ELEMENTS IN RESSTAPKSCORE2717STAPKSCORECORE2717STAPKSCORE2712717CORESTEP NUMBERS OF ELEMENTS IN RESCORE2717STEP NUMBERS OF ELEMENTS IN RESCORE2717STEP NUMBERS OF ELEMENTS IN RESCORE2717CORE<</td><td>SUBROUTINE RPCHINPKS) CORE 256 SUBROUTINE RPCHINPKS) CORE 257 C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 C CORE 250 CORE 250 C CORE CORE 250 CORE 250 C CORE CORE 251 CORE 251 C CORE CORE 251 CORE 252 C CORE CORE 250 CORE 253 C C CORE CORE 255 CORE 255 C C CORE CORE 256 CORE 255 C C CORE CORE 256 CORE 255 C C CORE CORE 256 CORE 256 C C CORE CORE 256 CORE 256 C</td><td>SUBROUTINE RPCHINES) CORE 256 F C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 251 C CORE 251 CORE 259 C CORE CORE 251 C CORE CORE 256 C CORE CORE 251 C C CORE CORE 251 C CORE<td>5 CORE 256 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 257 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 259 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 259 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 250 7 THIS PROGRAM FROM A GIVEN LOAD SPECTRUM CORE 250 7 THE NUMBER OF ADDITIONAL CYCLE SOLUTION CORE 265 7 THE NUMBER OF ADDITIONAL CYCLES (EXCLUDING INPUT CYCL CORE 265 7 THE NUMBER OF ADDITIONAL CYCLES (EXCLUDING INPUT CYCL CORE 265 7 SIGMAN THE INPUT LOAD SPECTRUM CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX</td><td>SUBROUTINE RPCH(NPKS) CORE 256 FNIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ARRAY NAME CORE 251 ARRAY NAME DEFINITION DIMENSIONS) CORE SIGMAX THE NUMBER OF THE INPUT LOAD SPECTRUM NEKS + KX CORE SIGMAX THE NUMBER OF THE INPUT LOAD SPECTRUM NEKS + KX CORE 253 ARRAY NAME PREAS OF THE INPUT LOAD SPECTRUM NEKS + KX CORE 254 SIGMAX VALLEYS OF THE INPUT COLD SPECTRUM NEKS + KX CORE 253 SIGMAN VALLEYS OF THE INPUT COLD SPECTRUM NEKS + KX CORE 257 SIGMAN VALLEYS OF THE INPUT SPECTRUM NEKS + KX CORE 253 OCCL ROUTERS OF THE INPUT SPECTRUM NEKS + KX CORE 251 SIGMAN RESTOLUMERS OF THE INPUT SPECT</td><td>5 CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIR PROGRAM ARRAYS CORE 25 7 THIR PROGRAM MILL CALCUS CERCULDING INPUT CYCL CORE 25 7 THE NUMBER OF THE INPUT LOAD SPECITUM NPKS + KK CORE 25 7 TRUCH MALLEYS NOT THE PROGRAM MILL CYCLES CERCULDING INPUT CYCL CORE 25 7 STGMIN VALLEYS OF THE INPUT LOAD SPECITUM NPKS + KK CORE 25 7 STGMIN VALLEYS OF THE INPUT LOAD</td><td>SUBROUTINE RPCHINES) SUBROUTINE RPCHINES) CORE 257 FILS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 259 255 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 CORE CORE 250 ARRAY NAME PROGRAM ARRAYS CORE
 263 CORE SIGMAX DEFINITION DIFINITION CORE 253 ARRAY NAME DEFINITION DEFINITION DIFINITION CORE 253 ARRAY NAME DEFINITION DIFINITION DIFINITION CORE 253 ARRAY NAME PREASO OF THE INPUT LOAD SPECTRUM NPSS + KK CORE 255 ARRAY NAME SIGMAX DIFINITION DIFINITION CORE 255 ARRAY NAME SIGMAX THE INPUT LOAD SPECTRUM NPSS + KK CORE 255 ARRAY NAME SIGMAX THE INPUT LOAD SPECTRUM NPSS + KK CORE 256 ARRAY NAME THE INPUT LOAD SPECTRUM NPSS + KK CORE 256 256 ARROYC</td><td>5 SUBROUTINE RPCH(NPKS) CORE 257 5 THIS PROGRAM EMPLOYS THE RANGE PATR CYCLE COUNTING METHOD TO GENER CORE 257 6 THIS PROGRAM EMPLOYS THE RANGE PATR CYCLE COUNTING METHOD TO GENER CORE 257 6 THIS PROGRAM EMPLOYS THE RANGE DATE CYCLE COUNTING METHOD TO GENER CORE 257 10 CORE 257 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 257 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 255 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 11 ARRAY NAME DEFINITION DIFENSIONS) CORE 257 11 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 12 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 13 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 14 ARRAY NAME DEFINITION DIFENSIONS) DORE 257 14 ARRAY NAME DEFINITION DIFENSIONS DORE 257 15</td><td>5 CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE OT COMMENT CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE OT COMMENT CORE 25 7 THIS PROGRAM MERCE DEFINITION CORE 25 7 THE INVITION DEFINITION DITENSION CORE 25 7 THE INVITION THE INVITION THE INVITION CORE 25 7 THE INVITION THE INVITION THE INVITION CORE 25 7 THE INVIERS OF THE INVITION THE INVIER<td>5 5 5 6 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM HEROLOS 56 7 THIS PROGRAM MERATS 56 7 THE INPUT 57 7 THE INPUT ON NELDED TO CHARGE DIRENSIONS) 56 7 THE INPUT CAGE SCITRUM 56 7 WALETS OF THE INPUT CAGE SCITRUM 56 7 KK = THE UNUBER OF DISTINGLIC CYCLES (ECCLUDING INPUT CYCL 56 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX SIGMUX 77 26 7 SIGMUX SIGMUX 77 27 7 SIGMU</td><td>5 SUBROUTINE RPCH(MPGS) CORE 27 5 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 6 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 7 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 10 THIS PROCRAM EMPLOYS THE RANGE DI CHANGE OTHANGE OTHANGE 27 11 THIS PROCRAM ARAYS CORE 25 12 TRAY NAME DEFINITION DIFFUNITION CORE 25 13 ARRAY NAME DEFINITION DIFFUNITION DIFFUNITION CORE 25 14 NUMERS OF THE INPUT LOAD SECTION DIFFUNITION DIFFUNITION CORE 25 15 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 15 CORE SIGMAX NALLEY SO THE INPUT LOAD SECTION CORE 25 16 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 16 SIGMAX THE NUMERS OF THE INPUT LOAD SECTION NESS + KK CORE 25 16 SIGMAX SIGMAX</td><td>5 SUBROUTINE RPCHVIACS) CORE 257 5 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257 6 CORE 255 7 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 255 10 REAV NAME PROGRAM REPLOYS THE RANGE OF THENSIONS) CORE 255 11 REAV NAME FEAKS OF THE INUTI TON DTHENSIONS) CORE 255 11 REAV NAME PROGRAM MENTION DTHENSIONS) CORE 255 12 REAV NAME THE INUTI TON DTHENSIONS) CORE 255 13 REAV NAME PRAKS OF THE INUT LOD SECTRUM DTHENSIONS CORE 255 14 REAV NAME THE INUT LOD SECTRUM PROKS + KK CORE 255 14 REAVE VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 15 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PR</td><td>5 CORE 257 10 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 11 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 12 REAL HOLE COUNTING METHOD TO GENER CORE 259 13 REAL HOLE COUNTING METHOD TO GENER CORE 259 14 FRECHAN FRON A GIVEN LOAD SPECTRUM CORE 259 15 REAL HOLE COUNTING METHOD TO GENER CORE 250 16 REAL HOLE COUNTING METHOD TO GENER CORE 250 17 REAL HOLE FOR THE INUTI CONSTRUMTION CORE 259 18 REAL HOLE FOR THE INUT CONSTRUMTION CORE 251 19 REAL HELO TO CHANGE STRUMT CORE 251 10 REAL HOLE FOR THE INUT CONSTRUMT CORE 251 11 REAL TO THE REAC OF THE INUT CONSTRUMT CORE 251 11 STGMAX HILE HOLE FOR THE INUT CONSTRUMT CORE 251 12 RECT K = THE HOUSE FOR THE INUT CONSTRUMT CORE 251 14 RECT K = THE HOUSE FOR THE RECT RECT CORE 251 15 RECT RECT RECT CORE 251 16 RECT RECT RECT CORE</td><td>SUBROUTINE RPCHINES) CORE 237 SUBROUTINE RPCHINES) CORE 237 FILS PROGRAM EMPLOYS THE RANGE PARE CYCLE COUNTING METHOD TO GENER DORE 233 ARRAY NAME FUGNAL STORE OF THE RANGE OF HEINSTON CORE 233 ARRAY NAME DIFFINITION CORE 233 SIGMAX FELNUTION DIFFINITION CORE 233 ARRAY NAME PROGRAM MARAYS CORE 233 SIGMAX FELNUTION CORE 234 SIGMAX FELNUTION CORE 235 ARRAY NAME PROGRAM MALLENG CORE 235 SIGMAX FELNUTION CORE 235 SIGMAX FELNUTION SIGGRAM CORE SIGMAX SIGGRAM</td><td>5 SUBGOUTINE RPONICARS1 CORE CORE 27 5 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 27 110 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 110 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 112 RRAY MARE DEFINITION NELOCO TO CHANGE CONTECTURIN DEFENSION DEFENSION 112 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN DEFENSION DEFENSION 113 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN PROS + KK DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC STGMAN<!--</td--><td>SUBROUTINE RPCHARKS CORE 257 INIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM FROM A GIVEN LOUD SECTRUM DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS OF THE INPUT
CONC. SCITAUM DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITION DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (FUNCT PROGRAM AND REAL FOR THE INPUT STORE AND REAL</td><td>35 SUBROUTINE RPHARKS CORE 5 SUBROUTINE RPHARKS CORE 6 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM ARKYS CORE 7 THIS PROGRAM ARKYS<!--</td--><td>Subsourtike RPCNINKS) CORE CORE</td><td>Subscurink Rechinks) Doe Subscurink Rechinks) State Cruck Counting Front A CLUC LOUNTING FELOD TO GRUE DORE THIS PROGRAM EMPLOYS THE RANGE OF ALL COUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD ARRAY MARE ARRAY MARE STGRUM STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY STGRUM ARRAY</td><td>SUBROUTINE RECHNARS) CORE <td< td=""><td>5 SUBGOUTINE RECHEMENSIS CORE 237 7 THIS PROGRAM EMPONSI THE RANGE FAIRS CYCLE CONVENDMENTION CORE 237 11 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 12 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 13 RERNY MARE DEFINITION DEFINITION CORE 14 RERNY MARE DEFINITION DEFINITION CORE 15 RERNY MARE DEFINITION DEFINITION CORE 16 RERNY MARE DEFINITION DEFINITION CORE 17 RERNY MARE DEFINITION DEFINITION CORE 18 RERNY CONFINENTION DEFINITION DEFINITION CORE 19 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 10 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 11 RERNY CONFILTION DEFINITION DEFINITION DEFINITION 12 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 13 RECOLUCUNITION DEFINITION DEFINITION 14 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 15 RERNY CONFILESTOR RERNY CONFILESTOR</td><td>5 SUBROUTINE REPRIVACE COME 25 111 SUBROUTINE REPRIVATIE ANGE ALT CYCLE CONNTINE METHOD TO GENER COME 25 112 COME 25 113 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 115 CONNEL CONNEL STATE 25 116 CONNEL CONNEL STATE 25 117 CONNEL CONNEL STATE 25 118 CONNEL CONNEL STATE 25 119 CONNEL CONNEL 27 110 CONNEL CONNEL 27 1111 CONNEL CONNEL 27 1111 CONNEL 27 27 1111 CONNEL 27</td><td>5 SUBROUTINE REPRIVATION DORE 25 7 THIS PROJERM FEMANYS DORE 25 10 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 11 DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 12 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 13 STEMM WHE PRAST OF ULE THATION NELLOD TO CHIRLE CHIRM FEMANY DORAGE 255 14 STEMM WHE PRAST OF THE THATION NELLOD TO CHIRLE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 16 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTA</td><td>SUBGUITINE RPENTORS
1115 PROGRAM RPEOTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCT CONTENT ON THE ANGE THE ANGE THE PROTECTION
AREA THE VIDERS OF THE INDUITION CORE STATE
AND THE ANGE THE REDUCT CORE STATE
AND THE AND THE ANGE THE REDUCT AND THE AND THE AND THE AND
AND THE AND THE</td><td> SUBGUITIKE RPCHURCS) SUBGUITIKE RPCHURCS) ATALYSTS SPECTRUM FRUNCS FIR ANGE PART FOLE DUNTING RETURD TO GAUE CORE
ATALYSTS SPECTRUM FRUNC STREAM A GUAL LOUS SPECTRUM ATALYSTS SPECTRUM FRUNC A GUAL LOUS SPECTRUM ATRAY MANE ARRAY MANE</td><td>5 SUBSOUTINE RPCHURKS) CORE 255 6 ALAYSTS SPECTRUM FRUNCY FILE ANGE PARS CYCLE CONTING NETHOD TO GENER CORE 255 11 PROGRAM ARAYS CURVED AL STANDARD CORE 12 RERN VALUE FROM AL STANDARD CURVED AL STANDARD 13 RERN VALUE FRAN AL STANDARD CURVED AL STANDARD 14 FRAN ALTICAL CONTING NETHON NELLICE TO CHARGE OF NERSTON CORE 15 RERN VALUE FRAN STANDARD CORE 16 STANDARD FRAN STANDARD CORE 17 STANDARD FRAN STANDARD CORE 18 STANDARD FRAN STANDARD CORE 19 STANDARD CORE CORE 10 STANDARD CORE CORE 11 STANDARD CORE CORE 12 STANDARD CORE CORE 13 STANDARD CORE CORE 14 CORE CORE CORE CORE 15 STANDARD CORE CORE </td></td<><td>5 SUBJOUTILE PROJUNE THE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE RUNCE TAE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE PROJUNE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE T</td><td>5 SUBJOUTIAE REPAIRANCES DOUNTIAE REPAIRANCES DOUNTIAE REPAIRANCES 1 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE REFERENCES 10 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE RETHON SECTION DOUNTIAE RETHON SECTION 11 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETHON DOUNTIAE RETHON 12 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETAIN DOUNTIAE RETAIN 13 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETAIN DOUNTIAE RETAIN</td><td>Subsort tick PERVISES Concent of the contrast stand Concent stand <thconcent stand<="" th=""> Concent stand Con</thconcent></td><td>SUBGOUTINE REMONENCIA 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE ROMAN STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBJOUTINE REMONE ROMAN STATE AND FOLL CONTINUE REMONE ROMAN 0000 FUE: REMONE ROMAN STATE AND FOLL ROMAN STATE AND FOLL ROMAN AND FOLL REMONE ROMAN 0000 STGRAM REMONE REMONE ROMAN AND FOLL REMONE ROMAN AND FOLD REMONE ROMAN AND FOLL REMONE ROMAN AND FOL REMON</td><td> SUBGUTIAK PERIMPRESI
HIST SPECIAL FEAR FARE PATE CTULE CONTINU NETHOD TO GREAT ONE CONTINUE NETHOD TO GREAT ON</td></td></td></td></td></td></td></td> | SUBROUTINE RPCH(NPKS)CORE2565THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2575CANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUMCORE2595CCCORE2596CCORE259CORE2597CCCORE259CORE2596CCCORE250CORE2507CORECORE250CORE2517CCORECORE253CORE2536CCCORECORE253CORE2537CCORECORECORE253CORE2537CCCORECORE253CORE2537CCCCORE253CORE2537CCCCORE253CORE2537CCCCCORE253CORE7CCCCC253CORE7CCCCC254CORE2557CCCCCC254C7CCCCCC255C7CCCCCC255C7CCCCCC256C7CCCC <td>SUBROUTINE RPCH(NPKS)SUBROUTINE RPCH(NPKS)CORE2565THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER2572586CORE2592592597NALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUMCORE2597CORE2592592597RAAY NAMEPROGRAM ARAYSCORE25110CARAY NAMEDEFINITIONCORE2535SIGMAXPEAKS OF THE INPUT LOAD SPECTRUMDIMENSIONCORE2536SIGMANNUMER A DOTITIONAL CYCLES (EXCLUDING INPUT CYCL2662665SIGMINVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2566SIGMINVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2567CONCE2562572662667CONCE2562662662667CONCE2562662662667CONCE2562662662667CONCE257000000000000000000000000000000000</td> <td>SUBROUTINE RPCH(NPKS)SUBROUTINE RPCH(NPKS)CORE2565CTHIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER
CORE2595CCCORE2596CCORE2592596CCORE2592597CCORE2592597CCORE2592597CORECORE2592517CORECORE2502537CORECORE2532637CORECORE2632637CORECORE2642647CORECORE2652647CORECORE2652647CORECORE2642657CORECORE2662667CORECORE2667CORE2662667CORE2662667CORE2662667CORE2662667CORE2662667CORE2662667CORE2662707CORE2662707CORE2662707CORE2662707CORE2662707CORE2702027CCORE2667CCC7CCC<td>SUBROUTINE RCHINPKS)SUBROUTINE RCHINPKS)CORE2565THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2576THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2596ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUMCORE2597CORE259CORE2597CORE259CORE2597CORE259CORE2597CORE261CORE2597CORE261CORE2627CORE261CORE2637SIGMAXDEFINITIONDIMENSIONS)CORE2637SIGMAXPEAKS OF THE INPUT LOAD SPECTRUMDIMENSIONCORE2657SIGMAXVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2657SIGMINVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2657SIGMINSTEP NUMBERS OF THE INPUT SPECTRUMNPKS + KKCORE2517STEP NUMBERS OF THE INPUT SPECTRUMSTAPKSCORE2717STEP NUMBERS OF ELEMENTS IN RESSTAPKSCORE2717STAPKSCORECORE2717STAPKSCORE2712717CORESTEP NUMBERS OF ELEMENTS IN RESCORE2717STEP NUMBERS OF ELEMENTS IN RESCORE2717STEP NUMBERS OF ELEMENTS IN RESCORE2717CORE<</td><td>SUBROUTINE RPCHINPKS) CORE 256 SUBROUTINE RPCHINPKS) CORE 257 C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 C CORE 250 CORE 250 C CORE CORE 250 CORE 250 C CORE CORE 251 CORE 251 C CORE CORE 251 CORE 252 C CORE CORE 250 CORE 253 C C CORE CORE 255 CORE 255 C C CORE CORE 256 CORE 255 C C CORE CORE 256 CORE 255 C C CORE CORE 256 CORE 256 C C CORE CORE 256 CORE 256 C</td><td>SUBROUTINE RPCHINES) CORE 256 F C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 251 C CORE 251 CORE 259 C CORE CORE 251 C CORE CORE 256 C CORE CORE 251 C C CORE CORE 251 C CORE<td>5 CORE 256 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 257 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 259 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 259 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 250 7 THIS PROGRAM FROM A GIVEN LOAD SPECTRUM CORE 250 7 THE NUMBER OF ADDITIONAL CYCLE SOLUTION CORE 265 7 THE NUMBER OF ADDITIONAL CYCLES (EXCLUDING INPUT CYCL CORE 265 7 THE NUMBER OF ADDITIONAL CYCLES (EXCLUDING INPUT CYCL CORE 265 7 SIGMAN THE INPUT LOAD SPECTRUM CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX</td><td>SUBROUTINE RPCH(NPKS) CORE 256 FNIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ARRAY NAME CORE 251 ARRAY NAME DEFINITION DIMENSIONS) CORE SIGMAX THE NUMBER OF THE INPUT LOAD SPECTRUM NEKS + KX CORE SIGMAX THE NUMBER OF THE INPUT LOAD SPECTRUM NEKS + KX CORE 253 ARRAY NAME PREAS OF THE INPUT LOAD SPECTRUM NEKS + KX CORE 254 SIGMAX VALLEYS OF THE INPUT COLD SPECTRUM NEKS + KX CORE 253 SIGMAN VALLEYS OF THE INPUT COLD SPECTRUM NEKS + KX CORE 257 SIGMAN VALLEYS OF THE INPUT SPECTRUM NEKS + KX CORE 253 OCCL ROUTERS OF THE INPUT SPECTRUM NEKS + KX CORE 251 SIGMAN RESTOLUMERS OF THE INPUT SPECT</td><td>5 CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIR PROGRAM ARRAYS CORE 25 7 THIR PROGRAM MILL CALCUS CERCULDING INPUT CYCL CORE 25 7 THE NUMBER OF THE INPUT LOAD SPECITUM NPKS + KK CORE 25 7 TRUCH MALLEYS NOT THE PROGRAM MILL CYCLES CERCULDING INPUT CYCL CORE 25 7 STGMIN VALLEYS OF THE INPUT LOAD SPECITUM NPKS + KK CORE 25 7 STGMIN VALLEYS OF THE INPUT LOAD</td><td>SUBROUTINE RPCHINES) SUBROUTINE RPCHINES) CORE 257 FILS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 259 255 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 CORE CORE 250 ARRAY NAME PROGRAM ARRAYS CORE 263 CORE SIGMAX DEFINITION DIFINITION CORE 253 ARRAY NAME DEFINITION DEFINITION DIFINITION CORE 253 ARRAY NAME DEFINITION DIFINITION DIFINITION CORE 253 ARRAY NAME PREASO OF THE INPUT LOAD SPECTRUM NPSS + KK CORE 255 ARRAY NAME SIGMAX DIFINITION DIFINITION CORE 255 ARRAY NAME SIGMAX THE INPUT LOAD SPECTRUM NPSS + KK CORE 255 ARRAY NAME SIGMAX THE INPUT LOAD SPECTRUM NPSS + KK CORE 256 ARRAY NAME THE INPUT LOAD SPECTRUM NPSS + KK CORE 256 256 ARROYC</td><td>5 SUBROUTINE RPCH(NPKS) CORE 257 5 THIS PROGRAM EMPLOYS THE RANGE PATR CYCLE COUNTING METHOD TO GENER CORE 257 6 THIS PROGRAM EMPLOYS THE RANGE PATR CYCLE COUNTING METHOD TO GENER CORE 257 6 THIS PROGRAM EMPLOYS THE RANGE DATE CYCLE COUNTING METHOD TO GENER CORE 257 10 CORE 257 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 257 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 255 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 11 ARRAY NAME DEFINITION DIFENSIONS) CORE 257 11 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 12 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 13 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 14 ARRAY NAME DEFINITION DIFENSIONS) DORE 257 14 ARRAY NAME DEFINITION DIFENSIONS DORE 257 15</td><td>5 CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE OT COMMENT CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE OT COMMENT CORE 25 7 THIS PROGRAM MERCE DEFINITION CORE 25 7 THE INVITION DEFINITION DITENSION CORE 25 7 THE INVITION THE INVITION THE INVITION CORE 25 7 THE INVITION THE INVITION THE INVITION CORE 25 7 THE INVIERS OF THE INVITION THE INVIER<td>5 5 5 6 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM HEROLOS 56 7 THIS PROGRAM MERATS 56 7 THE INPUT 57 7 THE INPUT ON NELDED TO CHARGE DIRENSIONS) 56 7 THE INPUT CAGE SCITRUM
 56 7 WALETS OF THE INPUT CAGE SCITRUM 56 7 KK = THE UNUBER OF DISTINGLIC CYCLES (ECCLUDING INPUT CYCL 56 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX SIGMUX 77 26 7 SIGMUX SIGMUX 77 27 7 SIGMU</td><td>5 SUBROUTINE RPCH(MPGS) CORE 27 5 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 6 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 7 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 10 THIS PROCRAM EMPLOYS THE RANGE DI CHANGE OTHANGE OTHANGE 27 11 THIS PROCRAM ARAYS CORE 25 12 TRAY NAME DEFINITION DIFFUNITION CORE 25 13 ARRAY NAME DEFINITION DIFFUNITION DIFFUNITION CORE 25 14 NUMERS OF THE INPUT LOAD SECTION DIFFUNITION DIFFUNITION CORE 25 15 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 15 CORE SIGMAX NALLEY SO THE INPUT LOAD SECTION CORE 25 16 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 16 SIGMAX THE NUMERS OF THE INPUT LOAD SECTION NESS + KK CORE 25 16 SIGMAX SIGMAX</td><td>5 SUBROUTINE RPCHVIACS) CORE 257 5 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257 6 CORE 255 7 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 255 10 REAV NAME PROGRAM REPLOYS THE RANGE OF THENSIONS) CORE 255 11 REAV NAME FEAKS OF THE INUTI TON DTHENSIONS) CORE 255 11 REAV NAME PROGRAM MENTION DTHENSIONS) CORE 255 12 REAV NAME THE INUTI TON DTHENSIONS) CORE 255 13 REAV NAME PRAKS OF THE INUT LOD SECTRUM DTHENSIONS CORE 255 14 REAV NAME THE INUT LOD SECTRUM PROKS + KK CORE 255 14 REAVE VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 15 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PR</td><td>5 CORE 257 10 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 11 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 12 REAL HOLE COUNTING METHOD TO GENER CORE 259 13 REAL HOLE COUNTING METHOD TO GENER CORE 259 14 FRECHAN FRON A GIVEN LOAD SPECTRUM CORE 259 15 REAL HOLE COUNTING METHOD TO GENER CORE 250 16 REAL HOLE COUNTING METHOD TO GENER CORE 250 17 REAL HOLE FOR THE INUTI CONSTRUMTION CORE 259 18 REAL HOLE FOR THE INUT CONSTRUMTION CORE 251 19 REAL HELO TO CHANGE STRUMT CORE 251 10 REAL HOLE FOR THE INUT CONSTRUMT CORE 251 11 REAL TO THE REAC OF THE INUT CONSTRUMT CORE 251 11 STGMAX HILE HOLE FOR THE INUT CONSTRUMT CORE 251 12 RECT K = THE HOUSE FOR THE INUT CONSTRUMT CORE 251 14 RECT K = THE HOUSE FOR THE RECT RECT CORE 251 15 RECT RECT RECT CORE 251 16 RECT RECT RECT CORE</td><td>SUBROUTINE RPCHINES) CORE 237 SUBROUTINE RPCHINES) CORE 237 FILS PROGRAM EMPLOYS THE RANGE PARE CYCLE COUNTING METHOD TO GENER DORE 233 ARRAY NAME FUGNAL STORE OF THE RANGE OF HEINSTON CORE 233 ARRAY NAME DIFFINITION CORE 233 SIGMAX FELNUTION DIFFINITION CORE 233 ARRAY NAME PROGRAM MARAYS CORE 233 SIGMAX FELNUTION CORE 234 SIGMAX FELNUTION CORE 235 ARRAY NAME PROGRAM MALLENG CORE 235 SIGMAX FELNUTION CORE 235 SIGMAX FELNUTION SIGGRAM CORE SIGMAX SIGGRAM</td><td>5 SUBGOUTINE RPONICARS1 CORE CORE 27 5 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 27 110 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 110 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 112 RRAY MARE DEFINITION NELOCO TO CHANGE CONTECTURIN DEFENSION DEFENSION 112 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN DEFENSION DEFENSION 113 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN PROS + KK DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC STGMAN<!--</td--><td>SUBROUTINE RPCHARKS CORE 257 INIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM FROM A GIVEN LOUD SECTRUM DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS OF THE INPUT CONC. SCITAUM DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITION DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (FUNCT PROGRAM AND REAL FOR THE INPUT STORE AND REAL</td><td>35 SUBROUTINE RPHARKS CORE 5 SUBROUTINE RPHARKS CORE 6 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM ARKYS CORE 7 THIS PROGRAM ARKYS<!--</td--><td>Subsourtike RPCNINKS) CORE CORE</td><td>Subscurink Rechinks) Doe Subscurink Rechinks) State Cruck Counting Front A CLUC LOUNTING FELOD TO GRUE DORE THIS PROGRAM EMPLOYS THE RANGE OF ALL COUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD ARRAY MARE ARRAY MARE STGRUM STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY STGRUM ARRAY</td><td>SUBROUTINE RECHNARS) CORE <td< td=""><td>5 SUBGOUTINE RECHEMENSIS CORE 237 7 THIS PROGRAM EMPONSI THE RANGE FAIRS CYCLE CONVENDMENTION CORE 237 11 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 12 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 13 RERNY MARE DEFINITION DEFINITION CORE 14 RERNY MARE DEFINITION DEFINITION CORE 15 RERNY MARE DEFINITION DEFINITION CORE 16 RERNY MARE DEFINITION DEFINITION CORE 17 RERNY MARE DEFINITION DEFINITION CORE 18 RERNY CONFINENTION DEFINITION DEFINITION CORE 19 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 10 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 11 RERNY CONFILTION DEFINITION DEFINITION DEFINITION 12 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 13 RECOLUCUNITION DEFINITION DEFINITION 14 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 15 RERNY CONFILESTOR RERNY CONFILESTOR</td><td>5 SUBROUTINE REPRIVACE COME 25 111 SUBROUTINE REPRIVATIE ANGE ALT CYCLE CONNTINE METHOD TO GENER COME 25 112 COME 25 113 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 115 CONNEL CONNEL STATE 25 116 CONNEL CONNEL STATE 25 117 CONNEL
 CONNEL STATE 25 118 CONNEL CONNEL STATE 25 119 CONNEL CONNEL 27 110 CONNEL CONNEL 27 1111 CONNEL CONNEL 27 1111 CONNEL 27 27 1111 CONNEL 27</td><td>5 SUBROUTINE REPRIVATION DORE 25 7 THIS PROJERM FEMANYS DORE 25 10 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 11 DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 12 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 13 STEMM WHE PRAST OF ULE THATION NELLOD TO CHIRLE CHIRM FEMANY DORAGE 255 14 STEMM WHE PRAST OF THE THATION NELLOD TO CHIRLE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 16 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTA</td><td>SUBGUITINE RPENTORS
1115 PROGRAM RPEOTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCT CONTENT ON THE ANGE THE ANGE THE PROTECTION
AREA THE VIDERS OF THE INDUITION CORE STATE
AND THE ANGE THE REDUCT CORE STATE
AND THE AND THE ANGE THE REDUCT AND THE AND THE AND THE AND
AND THE AND THE</td><td> SUBGUITIKE RPCHURCS) SUBGUITIKE RPCHURCS) ATALYSTS SPECTRUM FRUNCS FIR ANGE PART FOLE DUNTING RETURD TO GAUE CORE
ATALYSTS SPECTRUM FRUNC STREAM A GUAL LOUS SPECTRUM ATALYSTS SPECTRUM FRUNC A GUAL LOUS SPECTRUM ATRAY MANE ARRAY MANE</td><td>5 SUBSOUTINE RPCHURKS) CORE 255 6 ALAYSTS SPECTRUM FRUNCY FILE ANGE PARS CYCLE CONTING NETHOD TO GENER CORE 255 11 PROGRAM ARAYS CURVED AL STANDARD CORE 12 RERN VALUE FROM AL STANDARD CURVED AL STANDARD 13 RERN VALUE FRAN AL STANDARD CURVED AL STANDARD 14 FRAN ALTICAL CONTING NETHON NELLICE TO CHARGE OF NERSTON CORE 15 RERN VALUE FRAN STANDARD CORE 16 STANDARD FRAN STANDARD CORE 17 STANDARD FRAN STANDARD CORE 18 STANDARD FRAN STANDARD CORE 19 STANDARD CORE CORE 10 STANDARD CORE CORE 11 STANDARD CORE CORE 12 STANDARD CORE CORE 13 STANDARD CORE CORE 14 CORE CORE CORE CORE 15 STANDARD CORE CORE </td></td<><td>5 SUBJOUTILE PROJUNE THE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE RUNCE TAE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE PROJUNE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE T</td><td>5 SUBJOUTIAE REPAIRANCES DOUNTIAE REPAIRANCES DOUNTIAE REPAIRANCES 1 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE REFERENCES 10 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE RETHON SECTION DOUNTIAE RETHON SECTION 11 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETHON DOUNTIAE RETHON 12 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETAIN DOUNTIAE RETAIN 13 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETAIN DOUNTIAE RETAIN</td><td>Subsort tick PERVISES Concent of the contrast stand Concent stand <thconcent stand<="" th=""> Concent stand Con</thconcent></td><td>SUBGOUTINE REMONENCIA 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE ROMAN STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBJOUTINE REMONE ROMAN STATE AND FOLL CONTINUE REMONE ROMAN 0000 FUE: REMONE ROMAN STATE AND FOLL ROMAN STATE AND FOLL ROMAN AND FOLL REMONE ROMAN 0000 STGRAM REMONE REMONE ROMAN AND FOLL REMONE ROMAN AND FOLD REMONE ROMAN AND FOLL REMONE ROMAN AND FOL REMON</td><td> SUBGUTIAK PERIMPRESI
HIST SPECIAL FEAR FARE PATE CTULE CONTINU NETHOD TO GREAT ONE CONTINUE NETHOD TO GREAT ON</td></td></td></td></td></td></td> | SUBROUTINE RPCH(NPKS)SUBROUTINE RPCH(NPKS)CORE2565THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER2572586CORE2592592597NALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUMCORE2597CORE2592592597RAAY NAMEPROGRAM ARAYSCORE25110CARAY NAMEDEFINITIONCORE2535SIGMAXPEAKS OF THE INPUT LOAD SPECTRUMDIMENSIONCORE2536SIGMANNUMER A DOTITIONAL CYCLES (EXCLUDING INPUT CYCL2662665SIGMINVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2566SIGMINVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2567CONCE2562572662667CONCE2562662662667CONCE2562662662667CONCE2562662662667CONCE257000000000000000000000000000000000 | SUBROUTINE RPCH(NPKS)SUBROUTINE RPCH(NPKS)CORE2565CTHIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2595CCCORE2596CCORE2592596CCORE2592597CCORE2592597CCORE2592597CORECORE2592517CORECORE2502537CORECORE2532637CORECORE2632637CORECORE2642647CORECORE2652647CORECORE2652647CORECORE2642657CORECORE2662667CORECORE2667CORE2662667CORE2662667CORE2662667CORE2662667CORE2662667CORE2662667CORE2662707CORE2662707CORE2662707CORE2662707CORE2662707CORE2702027CCORE2667CCC7CCC <td>SUBROUTINE RCHINPKS)SUBROUTINE RCHINPKS)CORE2565THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2576THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2596ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUMCORE2597CORE259CORE2597CORE259CORE2597CORE259CORE2597CORE261CORE2597CORE261CORE2627CORE261CORE2637SIGMAXDEFINITIONDIMENSIONS)CORE2637SIGMAXPEAKS OF THE INPUT LOAD SPECTRUMDIMENSIONCORE2657SIGMAXVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2657SIGMINVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2657SIGMINSTEP NUMBERS OF THE INPUT SPECTRUMNPKS + KKCORE2517STEP NUMBERS OF THE INPUT SPECTRUMSTAPKSCORE2717STEP NUMBERS OF ELEMENTS IN RESSTAPKSCORE2717STAPKSCORECORE2717STAPKSCORE2712717CORESTEP NUMBERS OF ELEMENTS IN RESCORE2717STEP NUMBERS OF ELEMENTS IN RESCORE2717STEP NUMBERS OF ELEMENTS IN RESCORE2717CORE<</td> <td>SUBROUTINE RPCHINPKS) CORE 256 SUBROUTINE RPCHINPKS) CORE 257 C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 C CORE 250 CORE 250 C CORE CORE 250 CORE 250 C CORE CORE 251 CORE 251 C CORE CORE 251 CORE 252 C CORE CORE 250 CORE 253 C C CORE CORE 255 CORE 255 C C CORE CORE 256 CORE 255 C C CORE CORE 256 CORE 255 C C CORE CORE 256 CORE 256 C C CORE CORE 256 CORE 256 C</td> <td>SUBROUTINE RPCHINES) CORE 256 F C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 251 C CORE 251 CORE 259 C CORE CORE 251 C CORE CORE 256 C CORE CORE 251 C C CORE CORE 251 C CORE<td>5 CORE 256 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 257 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 259 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 259 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 250 7 THIS PROGRAM FROM A GIVEN LOAD SPECTRUM CORE 250 7 THE NUMBER OF ADDITIONAL CYCLE SOLUTION CORE 265 7 THE NUMBER OF ADDITIONAL CYCLES (EXCLUDING INPUT CYCL CORE 265 7 THE NUMBER OF ADDITIONAL CYCLES (EXCLUDING INPUT CYCL CORE 265 7 SIGMAN THE INPUT LOAD SPECTRUM CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL
CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX</td><td>SUBROUTINE RPCH(NPKS) CORE 256 FNIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ARRAY NAME CORE 251 ARRAY NAME DEFINITION DIMENSIONS) CORE SIGMAX THE NUMBER OF THE INPUT LOAD SPECTRUM NEKS + KX CORE SIGMAX THE NUMBER OF THE INPUT LOAD SPECTRUM NEKS + KX CORE 253 ARRAY NAME PREAS OF THE INPUT LOAD SPECTRUM NEKS + KX CORE 254 SIGMAX VALLEYS OF THE INPUT COLD SPECTRUM NEKS + KX CORE 253 SIGMAN VALLEYS OF THE INPUT COLD SPECTRUM NEKS + KX CORE 257 SIGMAN VALLEYS OF THE INPUT SPECTRUM NEKS + KX CORE 253 OCCL ROUTERS OF THE INPUT SPECTRUM NEKS + KX CORE 251 SIGMAN RESTOLUMERS OF THE INPUT SPECT</td><td>5 CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIR PROGRAM ARRAYS CORE 25 7 THIR PROGRAM MILL CALCUS CERCULDING INPUT CYCL CORE 25 7 THE NUMBER OF THE INPUT LOAD SPECITUM NPKS + KK CORE 25 7 TRUCH MALLEYS NOT THE PROGRAM MILL CYCLES CERCULDING INPUT CYCL CORE 25 7 STGMIN VALLEYS OF THE INPUT LOAD SPECITUM NPKS + KK CORE 25 7 STGMIN VALLEYS OF THE INPUT LOAD</td><td>SUBROUTINE RPCHINES) SUBROUTINE RPCHINES) CORE 257 FILS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 259 255 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 CORE CORE 250 ARRAY NAME PROGRAM ARRAYS CORE 263 CORE SIGMAX DEFINITION DIFINITION CORE 253 ARRAY NAME DEFINITION DEFINITION DIFINITION CORE 253 ARRAY NAME DEFINITION DIFINITION DIFINITION CORE 253 ARRAY NAME PREASO OF THE INPUT LOAD SPECTRUM NPSS + KK CORE 255 ARRAY NAME SIGMAX DIFINITION DIFINITION CORE 255 ARRAY NAME SIGMAX THE INPUT LOAD SPECTRUM NPSS + KK CORE 255 ARRAY NAME SIGMAX THE INPUT LOAD SPECTRUM NPSS + KK CORE 256 ARRAY NAME THE INPUT LOAD SPECTRUM NPSS + KK CORE 256 256 ARROYC</td><td>5 SUBROUTINE RPCH(NPKS) CORE 257 5 THIS PROGRAM EMPLOYS THE RANGE PATR CYCLE COUNTING METHOD TO GENER CORE 257 6 THIS PROGRAM EMPLOYS THE RANGE PATR CYCLE COUNTING METHOD TO GENER CORE 257 6 THIS PROGRAM EMPLOYS THE RANGE DATE CYCLE COUNTING METHOD TO GENER CORE 257 10 CORE 257 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 257 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 255 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 11 ARRAY NAME DEFINITION DIFENSIONS) CORE 257 11 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 12 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 13 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 14 ARRAY NAME DEFINITION DIFENSIONS) DORE 257 14 ARRAY NAME DEFINITION DIFENSIONS DORE 257 15</td><td>5 CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE OT COMMENT CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE OT COMMENT CORE 25 7 THIS PROGRAM MERCE DEFINITION CORE 25 7 THE INVITION DEFINITION DITENSION CORE 25 7 THE INVITION THE INVITION THE INVITION CORE 25 7 THE INVITION THE INVITION THE INVITION CORE 25 7 THE INVIERS OF THE INVITION THE INVIER<td>5 5 5 6 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM HEROLOS 56 7 THIS PROGRAM MERATS 56 7 THE INPUT 57 7 THE INPUT ON NELDED TO CHARGE DIRENSIONS) 56 7 THE INPUT CAGE SCITRUM 56 7 WALETS OF THE INPUT CAGE SCITRUM 56 7 KK = THE UNUBER OF DISTINGLIC CYCLES (ECCLUDING INPUT CYCL 56 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX SIGMUX 77 26 7 SIGMUX SIGMUX 77 27 7 SIGMU</td><td>5 SUBROUTINE RPCH(MPGS) CORE 27 5 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 6 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 7 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 10 THIS PROCRAM EMPLOYS THE RANGE DI CHANGE OTHANGE OTHANGE 27 11 THIS PROCRAM ARAYS CORE 25 12 TRAY NAME DEFINITION DIFFUNITION CORE 25 13 ARRAY NAME DEFINITION DIFFUNITION DIFFUNITION CORE 25 14 NUMERS OF THE INPUT LOAD SECTION DIFFUNITION DIFFUNITION CORE 25 15 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 15 CORE SIGMAX NALLEY SO THE INPUT LOAD SECTION CORE 25 16 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 16 SIGMAX THE NUMERS OF THE INPUT LOAD SECTION NESS + KK CORE 25 16 SIGMAX SIGMAX</td><td>5 SUBROUTINE RPCHVIACS) CORE 257 5 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257 6 CORE 255 7 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 255 10 REAV NAME PROGRAM REPLOYS THE RANGE OF THENSIONS) CORE 255 11 REAV NAME FEAKS OF THE INUTI TON DTHENSIONS) CORE 255 11 REAV NAME PROGRAM MENTION DTHENSIONS) CORE 255 12 REAV NAME THE INUTI TON DTHENSIONS) CORE 255 13 REAV NAME PRAKS OF THE INUT LOD SECTRUM DTHENSIONS CORE 255 14 REAV NAME THE INUT LOD SECTRUM PROKS + KK CORE 255 14 REAVE VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 15 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PR</td><td>5 CORE 257 10 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 11 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 12 REAL HOLE COUNTING METHOD TO GENER CORE 259 13 REAL HOLE COUNTING METHOD TO GENER CORE 259 14 FRECHAN FRON A GIVEN LOAD SPECTRUM CORE 259 15 REAL HOLE COUNTING METHOD TO GENER CORE 250 16 REAL HOLE COUNTING METHOD TO GENER CORE 250 17 REAL HOLE FOR THE INUTI CONSTRUMTION CORE 259 18 REAL HOLE FOR THE INUT CONSTRUMTION CORE 251 19 REAL HELO TO CHANGE STRUMT CORE 251 10 REAL HOLE FOR THE INUT CONSTRUMT CORE 251 11 REAL TO THE REAC OF THE INUT CONSTRUMT CORE
251 11 STGMAX HILE HOLE FOR THE INUT CONSTRUMT CORE 251 12 RECT K = THE HOUSE FOR THE INUT CONSTRUMT CORE 251 14 RECT K = THE HOUSE FOR THE RECT RECT CORE 251 15 RECT RECT RECT CORE 251 16 RECT RECT RECT CORE</td><td>SUBROUTINE RPCHINES) CORE 237 SUBROUTINE RPCHINES) CORE 237 FILS PROGRAM EMPLOYS THE RANGE PARE CYCLE COUNTING METHOD TO GENER DORE 233 ARRAY NAME FUGNAL STORE OF THE RANGE OF HEINSTON CORE 233 ARRAY NAME DIFFINITION CORE 233 SIGMAX FELNUTION DIFFINITION CORE 233 ARRAY NAME PROGRAM MARAYS CORE 233 SIGMAX FELNUTION CORE 234 SIGMAX FELNUTION CORE 235 ARRAY NAME PROGRAM MALLENG CORE 235 SIGMAX FELNUTION CORE 235 SIGMAX FELNUTION SIGGRAM CORE SIGMAX SIGGRAM</td><td>5 SUBGOUTINE RPONICARS1 CORE CORE 27 5 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 27 110 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 110 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 112 RRAY MARE DEFINITION NELOCO TO CHANGE CONTECTURIN DEFENSION DEFENSION 112 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN DEFENSION DEFENSION 113 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN PROS + KK DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC STGMAN<!--</td--><td>SUBROUTINE RPCHARKS CORE 257 INIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM FROM A GIVEN LOUD SECTRUM DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS OF THE INPUT CONC. SCITAUM DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITION DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (FUNCT PROGRAM AND REAL FOR THE INPUT STORE AND REAL</td><td>35 SUBROUTINE RPHARKS CORE 5 SUBROUTINE RPHARKS CORE 6 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM ARKYS CORE 7 THIS PROGRAM ARKYS<!--</td--><td>Subsourtike RPCNINKS) CORE CORE</td><td>Subscurink Rechinks) Doe Subscurink Rechinks) State Cruck Counting Front A CLUC LOUNTING FELOD TO GRUE DORE THIS PROGRAM EMPLOYS THE RANGE OF ALL COUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD ARRAY MARE ARRAY MARE STGRUM STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY STGRUM ARRAY</td><td>SUBROUTINE RECHNARS) CORE <td< td=""><td>5 SUBGOUTINE RECHEMENSIS CORE 237 7 THIS PROGRAM EMPONSI THE RANGE FAIRS CYCLE CONVENDMENTION CORE 237 11 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 12 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 13 RERNY MARE DEFINITION DEFINITION CORE 14 RERNY MARE DEFINITION DEFINITION CORE 15 RERNY MARE DEFINITION DEFINITION CORE 16 RERNY MARE DEFINITION DEFINITION CORE 17 RERNY MARE DEFINITION DEFINITION CORE 18 RERNY CONFINENTION DEFINITION DEFINITION CORE 19 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 10 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 11 RERNY CONFILTION DEFINITION DEFINITION DEFINITION 12 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 13 RECOLUCUNITION DEFINITION DEFINITION 14 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 15 RERNY CONFILESTOR RERNY CONFILESTOR</td><td>5 SUBROUTINE REPRIVACE COME 25 111 SUBROUTINE REPRIVATIE ANGE ALT CYCLE CONNTINE METHOD TO GENER COME 25 112 COME 25 113 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 115 CONNEL CONNEL STATE 25 116 CONNEL CONNEL STATE 25 117 CONNEL CONNEL STATE 25 118 CONNEL CONNEL STATE 25 119 CONNEL CONNEL 27 110 CONNEL CONNEL 27 1111 CONNEL CONNEL 27 1111 CONNEL 27 27 1111 CONNEL 27</td><td>5 SUBROUTINE REPRIVATION DORE 25 7 THIS PROJERM FEMANYS DORE 25 10 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 11 DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 12 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 13 STEMM WHE PRAST OF ULE THATION NELLOD TO CHIRLE CHIRM FEMANY DORAGE 255 14 STEMM WHE PRAST OF THE THATION NELLOD TO CHIRLE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 16 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTA</td><td>SUBGUITINE RPENTORS
1115 PROGRAM RPEOTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCT CONTENT ON THE ANGE THE ANGE THE PROTECTION
AREA THE VIDERS OF THE INDUITION CORE STATE
AND THE ANGE THE REDUCT CORE STATE
AND THE AND THE ANGE THE REDUCT AND THE AND THE AND THE AND
AND THE AND THE</td><td> SUBGUITIKE RPCHURCS) SUBGUITIKE RPCHURCS) ATALYSTS SPECTRUM FRUNCS FIR ANGE PART FOLE DUNTING RETURD TO GAUE CORE
ATALYSTS SPECTRUM FRUNC STREAM A GUAL LOUS SPECTRUM ATALYSTS SPECTRUM FRUNC A GUAL LOUS SPECTRUM ATRAY MANE ARRAY MANE</td><td>5 SUBSOUTINE RPCHURKS) CORE 255 6 ALAYSTS SPECTRUM FRUNCY FILE ANGE PARS CYCLE CONTING NETHOD TO GENER CORE 255 11 PROGRAM ARAYS CURVED AL STANDARD CORE 12 RERN VALUE FROM AL STANDARD CURVED AL STANDARD 13 RERN VALUE FRAN AL STANDARD CURVED AL STANDARD 14 FRAN ALTICAL CONTING NETHON NELLICE TO CHARGE OF NERSTON CORE 15 RERN VALUE FRAN STANDARD CORE 16 STANDARD FRAN STANDARD CORE 17 STANDARD FRAN STANDARD CORE 18 STANDARD FRAN STANDARD CORE 19 STANDARD CORE CORE 10 STANDARD CORE CORE 11 STANDARD CORE CORE 12 STANDARD CORE CORE 13 STANDARD CORE CORE 14 CORE CORE CORE CORE 15 STANDARD CORE CORE </td></td<><td>5 SUBJOUTILE PROJUNE THE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE RUNCE TAE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE PROJUNE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE T</td><td>5 SUBJOUTIAE REPAIRANCES DOUNTIAE REPAIRANCES DOUNTIAE REPAIRANCES 1 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE REFERENCES 10
 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE RETHON SECTION DOUNTIAE RETHON SECTION 11 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETHON DOUNTIAE RETHON 12 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETAIN DOUNTIAE RETAIN 13 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETAIN DOUNTIAE RETAIN</td><td>Subsort tick PERVISES Concent of the contrast stand Concent stand <thconcent stand<="" th=""> Concent stand Con</thconcent></td><td>SUBGOUTINE REMONENCIA 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE ROMAN STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBJOUTINE REMONE ROMAN STATE AND FOLL CONTINUE REMONE ROMAN 0000 FUE: REMONE ROMAN STATE AND FOLL ROMAN STATE AND FOLL ROMAN AND FOLL REMONE ROMAN 0000 STGRAM REMONE REMONE ROMAN AND FOLL REMONE ROMAN AND FOLD REMONE ROMAN AND FOLL REMONE ROMAN AND FOL REMON</td><td> SUBGUTIAK PERIMPRESI
HIST SPECIAL FEAR FARE PATE CTULE CONTINU NETHOD TO GREAT ONE CONTINUE NETHOD TO GREAT ON</td></td></td></td></td></td> | SUBROUTINE RCHINPKS)SUBROUTINE RCHINPKS)CORE2565THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2576THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE2596ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUMCORE2597CORE259CORE2597CORE259CORE2597CORE259CORE2597CORE261CORE2597CORE261CORE2627CORE261CORE2637SIGMAXDEFINITIONDIMENSIONS)CORE2637SIGMAXPEAKS OF THE INPUT LOAD SPECTRUMDIMENSIONCORE2657SIGMAXVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2657SIGMINVALLEYS OF THE INPUT LOAD SPECTRUMNPKS + KKCORE2657SIGMINSTEP NUMBERS OF THE INPUT SPECTRUMNPKS + KKCORE2517STEP NUMBERS OF THE INPUT SPECTRUMSTAPKSCORE2717STEP NUMBERS OF ELEMENTS IN RESSTAPKSCORE2717STAPKSCORECORE2717STAPKSCORE2712717CORESTEP NUMBERS OF ELEMENTS IN RESCORE2717STEP NUMBERS OF ELEMENTS IN RESCORE2717STEP NUMBERS OF ELEMENTS IN RESCORE2717CORE< | SUBROUTINE RPCHINPKS) CORE 256 SUBROUTINE RPCHINPKS) CORE 257 C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 C CORE 250 CORE 250 C CORE CORE 250 CORE 250 C CORE CORE 251 CORE 251 C CORE CORE 251 CORE 252 C CORE CORE 250 CORE 253 C C CORE CORE 255 CORE 255 C C CORE CORE 256 CORE 255 C C CORE CORE 256 CORE 255 C C CORE CORE 256 CORE 256 C C CORE CORE 256 CORE 256 C | SUBROUTINE RPCHINES) CORE 256 F C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 C ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 251 C CORE 251 CORE 259 C CORE CORE 251 C CORE CORE 256 C CORE CORE 251 C C CORE CORE 251 C CORE <td>5 CORE 256 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 257 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 259 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 259 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 250 7 THIS PROGRAM FROM A GIVEN LOAD SPECTRUM CORE 250 7 THE NUMBER OF ADDITIONAL CYCLE SOLUTION CORE 265 7 THE NUMBER OF ADDITIONAL CYCLES (EXCLUDING INPUT CYCL CORE 265 7 THE NUMBER OF ADDITIONAL CYCLES (EXCLUDING INPUT CYCL CORE 265 7 SIGMAN THE INPUT LOAD SPECTRUM CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX</td> <td>SUBROUTINE RPCH(NPKS) CORE 256 FNIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ARRAY NAME CORE 251 ARRAY NAME DEFINITION DIMENSIONS) CORE SIGMAX THE NUMBER OF THE INPUT LOAD SPECTRUM NEKS + KX CORE SIGMAX THE NUMBER OF THE INPUT LOAD SPECTRUM NEKS + KX CORE 253 ARRAY NAME PREAS OF THE INPUT LOAD SPECTRUM NEKS + KX CORE 254 SIGMAX VALLEYS OF THE INPUT COLD SPECTRUM NEKS + KX CORE 253 SIGMAN VALLEYS OF THE INPUT COLD SPECTRUM NEKS + KX CORE 257 SIGMAN VALLEYS OF THE INPUT SPECTRUM NEKS + KX CORE 253 OCCL ROUTERS OF THE INPUT SPECTRUM NEKS + KX CORE 251 SIGMAN RESTOLUMERS OF THE INPUT SPECT</td> <td>5 CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIR PROGRAM ARRAYS CORE 25 7 THIR PROGRAM MILL CALCUS CERCULDING INPUT CYCL CORE 25 7 THE NUMBER OF THE INPUT LOAD SPECITUM NPKS + KK CORE 25 7 TRUCH MALLEYS NOT THE PROGRAM MILL CYCLES CERCULDING INPUT CYCL CORE 25 7 STGMIN VALLEYS OF THE INPUT LOAD SPECITUM NPKS + KK CORE 25 7 STGMIN VALLEYS OF THE INPUT LOAD</td> <td>SUBROUTINE RPCHINES) SUBROUTINE RPCHINES) CORE 257 FILS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 259 255 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 CORE CORE 250 ARRAY NAME PROGRAM ARRAYS CORE 263 CORE SIGMAX DEFINITION DIFINITION CORE 253 ARRAY NAME DEFINITION DEFINITION DIFINITION CORE 253 ARRAY NAME DEFINITION DIFINITION DIFINITION CORE 253 ARRAY NAME PREASO OF THE INPUT LOAD SPECTRUM NPSS + KK CORE 255 ARRAY NAME SIGMAX DIFINITION DIFINITION CORE 255 ARRAY NAME SIGMAX THE INPUT LOAD SPECTRUM NPSS + KK CORE 255 ARRAY NAME SIGMAX THE INPUT LOAD SPECTRUM NPSS + KK CORE 256 ARRAY NAME THE INPUT LOAD SPECTRUM NPSS + KK CORE 256 256 ARROYC</td> <td>5 SUBROUTINE RPCH(NPKS) CORE 257 5 THIS PROGRAM EMPLOYS THE RANGE PATR CYCLE COUNTING METHOD TO GENER CORE 257 6 THIS PROGRAM EMPLOYS THE RANGE PATR CYCLE COUNTING METHOD TO GENER CORE 257 6 THIS PROGRAM EMPLOYS THE RANGE DATE CYCLE COUNTING METHOD TO GENER CORE 257 10 CORE 257 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 257 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 255 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 11 ARRAY NAME DEFINITION DIFENSIONS) CORE 257 11 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 12 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 13 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 14 ARRAY NAME DEFINITION DIFENSIONS) DORE 257 14 ARRAY NAME DEFINITION DIFENSIONS DORE 257 15</td> <td>5 CORE 25 6
 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE OT COMMENT CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE OT COMMENT CORE 25 7 THIS PROGRAM MERCE DEFINITION CORE 25 7 THE INVITION DEFINITION DITENSION CORE 25 7 THE INVITION THE INVITION THE INVITION CORE 25 7 THE INVITION THE INVITION THE INVITION CORE 25 7 THE INVIERS OF THE INVITION THE INVIER<td>5 5 5 6 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM HEROLOS 56 7 THIS PROGRAM MERATS 56 7 THE INPUT 57 7 THE INPUT ON NELDED TO CHARGE DIRENSIONS) 56 7 THE INPUT CAGE SCITRUM 56 7 WALETS OF THE INPUT CAGE SCITRUM 56 7 KK = THE UNUBER OF DISTINGLIC CYCLES (ECCLUDING INPUT CYCL 56 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX SIGMUX 77 26 7 SIGMUX SIGMUX 77 27 7 SIGMU</td><td>5 SUBROUTINE RPCH(MPGS) CORE 27 5 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 6 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 7 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 10 THIS PROCRAM EMPLOYS THE RANGE DI CHANGE OTHANGE OTHANGE 27 11 THIS PROCRAM ARAYS CORE 25 12 TRAY NAME DEFINITION DIFFUNITION CORE 25 13 ARRAY NAME DEFINITION DIFFUNITION DIFFUNITION CORE 25 14 NUMERS OF THE INPUT LOAD SECTION DIFFUNITION DIFFUNITION CORE 25 15 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 15 CORE SIGMAX NALLEY SO THE INPUT LOAD SECTION CORE 25 16 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 16 SIGMAX THE NUMERS OF THE INPUT LOAD SECTION NESS + KK CORE 25 16 SIGMAX SIGMAX</td><td>5 SUBROUTINE RPCHVIACS) CORE 257 5 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257 6 CORE 255 7 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 255 10 REAV NAME PROGRAM REPLOYS THE RANGE OF THENSIONS) CORE 255 11 REAV NAME FEAKS OF THE INUTI TON DTHENSIONS) CORE 255 11 REAV NAME PROGRAM MENTION DTHENSIONS) CORE 255 12 REAV NAME THE INUTI TON DTHENSIONS) CORE 255 13 REAV NAME PRAKS OF THE INUT LOD SECTRUM DTHENSIONS CORE 255 14 REAV NAME THE INUT LOD SECTRUM PROKS + KK CORE 255 14 REAVE VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 15 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PR</td><td>5 CORE 257 10 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 11 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 12 REAL HOLE COUNTING METHOD TO GENER CORE 259 13 REAL HOLE COUNTING METHOD TO GENER CORE 259 14 FRECHAN FRON A GIVEN LOAD SPECTRUM CORE 259 15 REAL HOLE COUNTING METHOD TO GENER CORE 250 16 REAL HOLE COUNTING METHOD TO GENER CORE 250 17 REAL HOLE FOR THE INUTI CONSTRUMTION CORE 259 18 REAL HOLE FOR THE INUT CONSTRUMTION CORE 251 19 REAL HELO TO CHANGE STRUMT CORE 251 10 REAL HOLE FOR THE INUT CONSTRUMT CORE 251 11 REAL TO THE REAC OF THE INUT CONSTRUMT CORE 251 11 STGMAX HILE HOLE FOR THE INUT CONSTRUMT CORE 251 12 RECT K = THE HOUSE FOR THE INUT CONSTRUMT CORE 251 14 RECT K = THE HOUSE FOR THE RECT RECT CORE 251 15 RECT RECT RECT CORE 251 16 RECT RECT RECT CORE</td><td>SUBROUTINE RPCHINES) CORE 237 SUBROUTINE RPCHINES) CORE 237 FILS PROGRAM EMPLOYS THE RANGE PARE CYCLE COUNTING METHOD TO GENER DORE 233 ARRAY NAME FUGNAL STORE OF THE RANGE OF HEINSTON CORE 233 ARRAY NAME DIFFINITION CORE 233 SIGMAX FELNUTION DIFFINITION CORE 233 ARRAY NAME PROGRAM MARAYS CORE 233 SIGMAX FELNUTION CORE 234 SIGMAX FELNUTION CORE 235 ARRAY NAME PROGRAM MALLENG CORE 235 SIGMAX FELNUTION CORE 235 SIGMAX FELNUTION SIGGRAM CORE SIGMAX SIGGRAM</td><td>5 SUBGOUTINE RPONICARS1 CORE CORE 27 5 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 27 110 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 110 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 112 RRAY MARE DEFINITION NELOCO TO CHANGE CONTECTURIN DEFENSION DEFENSION 112 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN DEFENSION DEFENSION 113 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN PROS + KK DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC STGMAN<!--</td--><td>SUBROUTINE RPCHARKS CORE 257 INIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM FROM A GIVEN LOUD SECTRUM DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS OF THE INPUT CONC. SCITAUM DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITION DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (FUNCT PROGRAM AND REAL FOR THE INPUT STORE AND REAL</td><td>35 SUBROUTINE RPHARKS CORE 5 SUBROUTINE RPHARKS CORE 6 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM ARKYS CORE 7 THIS PROGRAM ARKYS<!--</td--><td>Subsourtike RPCNINKS) CORE CORE</td><td>Subscurink Rechinks) Doe Subscurink Rechinks) State Cruck Counting Front A CLUC LOUNTING FELOD TO GRUE DORE THIS PROGRAM EMPLOYS THE RANGE OF ALL COUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD ARRAY MARE ARRAY MARE STGRUM STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY STGRUM ARRAY</td><td>SUBROUTINE RECHNARS) CORE <td< td=""><td>5 SUBGOUTINE RECHEMENSIS CORE 237 7 THIS PROGRAM EMPONSI THE RANGE FAIRS CYCLE CONVENDMENTION CORE 237 11 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 12 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 13 RERNY MARE DEFINITION DEFINITION
CORE 14 RERNY MARE DEFINITION DEFINITION CORE 15 RERNY MARE DEFINITION DEFINITION CORE 16 RERNY MARE DEFINITION DEFINITION CORE 17 RERNY MARE DEFINITION DEFINITION CORE 18 RERNY CONFINENTION DEFINITION DEFINITION CORE 19 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 10 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 11 RERNY CONFILTION DEFINITION DEFINITION DEFINITION 12 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 13 RECOLUCUNITION DEFINITION DEFINITION 14 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 15 RERNY CONFILESTOR RERNY CONFILESTOR</td><td>5 SUBROUTINE REPRIVACE COME 25 111 SUBROUTINE REPRIVATIE ANGE ALT CYCLE CONNTINE METHOD TO GENER COME 25 112 COME 25 113 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 115 CONNEL CONNEL STATE 25 116 CONNEL CONNEL STATE 25 117 CONNEL CONNEL STATE 25 118 CONNEL CONNEL STATE 25 119 CONNEL CONNEL 27 110 CONNEL CONNEL 27 1111 CONNEL CONNEL 27 1111 CONNEL 27 27 1111 CONNEL 27</td><td>5 SUBROUTINE REPRIVATION DORE 25 7 THIS PROJERM FEMANYS DORE 25 10 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 11 DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 12 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 13 STEMM WHE PRAST OF ULE THATION NELLOD TO CHIRLE CHIRM FEMANY DORAGE 255 14 STEMM WHE PRAST OF THE THATION NELLOD TO CHIRLE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 16 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTA</td><td>SUBGUITINE RPENTORS
1115 PROGRAM RPEOTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCT CONTENT ON THE ANGE THE ANGE THE PROTECTION
AREA THE VIDERS OF THE INDUITION CORE STATE
AND THE ANGE THE REDUCT CORE STATE
AND THE AND THE ANGE THE REDUCT AND THE AND THE AND THE AND
AND THE AND THE</td><td> SUBGUITIKE RPCHURCS) SUBGUITIKE RPCHURCS) ATALYSTS SPECTRUM FRUNCS FIR ANGE PART FOLE DUNTING RETURD TO GAUE CORE
ATALYSTS SPECTRUM FRUNC STREAM A GUAL LOUS SPECTRUM ATALYSTS SPECTRUM FRUNC A GUAL LOUS SPECTRUM ATRAY MANE ARRAY MANE</td><td>5 SUBSOUTINE RPCHURKS) CORE 255 6 ALAYSTS SPECTRUM FRUNCY FILE ANGE PARS CYCLE CONTING NETHOD TO GENER CORE 255 11 PROGRAM ARAYS CURVED AL STANDARD CORE 12 RERN VALUE FROM AL STANDARD CURVED AL STANDARD 13 RERN VALUE FRAN AL STANDARD CURVED AL STANDARD 14 FRAN ALTICAL CONTING NETHON NELLICE TO CHARGE OF NERSTON CORE 15 RERN VALUE FRAN STANDARD CORE 16 STANDARD FRAN STANDARD CORE 17 STANDARD FRAN STANDARD CORE 18 STANDARD FRAN STANDARD CORE 19 STANDARD CORE CORE 10 STANDARD CORE CORE 11 STANDARD CORE CORE 12 STANDARD CORE CORE 13 STANDARD CORE CORE 14 CORE CORE CORE CORE 15 STANDARD CORE CORE </td></td<><td>5 SUBJOUTILE PROJUNE THE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE RUNCE TAE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE PROJUNE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE T</td><td>5 SUBJOUTIAE REPAIRANCES DOUNTIAE REPAIRANCES DOUNTIAE REPAIRANCES 1 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE REFERENCES 10 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE RETHON SECTION DOUNTIAE RETHON SECTION 11 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETHON DOUNTIAE RETHON 12 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETAIN DOUNTIAE RETAIN 13 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETAIN DOUNTIAE RETAIN</td><td>Subsort tick PERVISES Concent of the contrast stand Concent stand <thconcent stand<="" th=""> Concent stand Con</thconcent></td><td>SUBGOUTINE REMONENCIA 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE ROMAN STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBJOUTINE REMONE ROMAN STATE AND FOLL CONTINUE REMONE ROMAN 0000 FUE: REMONE ROMAN STATE AND FOLL ROMAN STATE AND FOLL ROMAN AND FOLL REMONE ROMAN 0000 STGRAM REMONE REMONE ROMAN AND FOLL REMONE ROMAN AND FOLD REMONE ROMAN AND FOLL REMONE ROMAN AND FOL REMON</td><td> SUBGUTIAK PERIMPRESI
HIST SPECIAL FEAR FARE PATE CTULE CONTINU NETHOD TO GREAT ONE CONTINUE NETHOD TO GREAT ON</td></td></td></td></td> | 5 CORE 256 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 257 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 259 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 259 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER 250 7 THIS PROGRAM FROM A GIVEN LOAD SPECTRUM CORE 250 7 THE NUMBER OF ADDITIONAL CYCLE SOLUTION CORE 265 7 THE NUMBER OF ADDITIONAL CYCLES (EXCLUDING INPUT CYCL CORE 265 7 THE NUMBER OF ADDITIONAL CYCLES (EXCLUDING INPUT CYCL CORE 265 7 SIGMAN THE INPUT LOAD SPECTRUM CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX THE INPUT LOAD SPECTRUM CYCL CORE 265 7 SIGMAX | SUBROUTINE RPCH(NPKS) CORE 256 FNIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ARRAY NAME CORE 251 ARRAY NAME DEFINITION DIMENSIONS) CORE SIGMAX THE NUMBER OF THE INPUT LOAD SPECTRUM NEKS + KX CORE SIGMAX THE NUMBER OF THE INPUT LOAD SPECTRUM NEKS + KX CORE 253 ARRAY NAME PREAS OF THE INPUT LOAD SPECTRUM NEKS + KX CORE 254 SIGMAX VALLEYS OF THE INPUT COLD SPECTRUM NEKS + KX CORE 253 SIGMAN VALLEYS OF THE INPUT COLD SPECTRUM NEKS + KX CORE 257 SIGMAN VALLEYS OF THE INPUT SPECTRUM NEKS + KX CORE 253 OCCL ROUTERS OF THE INPUT SPECTRUM NEKS + KX CORE 251 SIGMAN RESTOLUMERS OF THE INPUT SPECT | 5 CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PARR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIR PROGRAM ARRAYS CORE 25 7 THIR PROGRAM MILL CALCUS CERCULDING INPUT CYCL CORE 25 7 THE NUMBER OF THE INPUT LOAD SPECITUM NPKS + KK CORE
 25 7 TRUCH MALLEYS NOT THE PROGRAM MILL CYCLES CERCULDING INPUT CYCL CORE 25 7 STGMIN VALLEYS OF THE INPUT LOAD SPECITUM NPKS + KK CORE 25 7 STGMIN VALLEYS OF THE INPUT LOAD | SUBROUTINE RPCHINES) SUBROUTINE RPCHINES) CORE 257 FILS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 259 255 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 ANALYSIS SPECTRUM FROM A GIVEN LOAD SPECTRUM CORE 259 CORE CORE 250 ARRAY NAME PROGRAM ARRAYS CORE 263 CORE SIGMAX DEFINITION DIFINITION CORE 253 ARRAY NAME DEFINITION DEFINITION DIFINITION CORE 253 ARRAY NAME DEFINITION DIFINITION DIFINITION CORE 253 ARRAY NAME PREASO OF THE INPUT LOAD SPECTRUM NPSS + KK CORE 255 ARRAY NAME SIGMAX DIFINITION DIFINITION CORE 255 ARRAY NAME SIGMAX THE INPUT LOAD SPECTRUM NPSS + KK CORE 255 ARRAY NAME SIGMAX THE INPUT LOAD SPECTRUM NPSS + KK CORE 256 ARRAY NAME THE INPUT LOAD SPECTRUM NPSS + KK CORE 256 256 ARROYC | 5 SUBROUTINE RPCH(NPKS) CORE 257 5 THIS PROGRAM EMPLOYS THE RANGE PATR CYCLE COUNTING METHOD TO GENER CORE 257 6 THIS PROGRAM EMPLOYS THE RANGE PATR CYCLE COUNTING METHOD TO GENER CORE 257 6 THIS PROGRAM EMPLOYS THE RANGE DATE CYCLE COUNTING METHOD TO GENER CORE 257 10 CORE 257 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 257 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 255 10 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 11 ARRAY NAME DEFINITION DIFENSIONS) CORE 257 11 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 12 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 13 ARRAY NAME DEFINITION DIFENSIONS) CORE 256 14 ARRAY NAME DEFINITION DIFENSIONS) DORE 257 14 ARRAY NAME DEFINITION DIFENSIONS DORE 257 15 | 5 CORE 25 6 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE OT COMMENT CORE 25 7 THIS PROGRAM EMPLOYS THE RANGE OT COMMENT CORE 25 7 THIS PROGRAM MERCE DEFINITION CORE 25 7 THE INVITION DEFINITION DITENSION CORE 25 7 THE INVITION THE INVITION THE INVITION CORE 25 7 THE INVITION THE INVITION THE INVITION CORE 25 7 THE INVIERS OF THE INVITION THE INVIER <td>5 5 5 6 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM HEROLOS 56 7 THIS PROGRAM MERATS 56 7 THE INPUT 57 7 THE INPUT ON NELDED TO CHARGE DIRENSIONS) 56 7 THE INPUT CAGE SCITRUM 56 7 WALETS OF THE INPUT CAGE SCITRUM 56 7 KK = THE UNUBER OF DISTINGLIC CYCLES (ECCLUDING INPUT CYCL 56 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX SIGMUX 77 26 7 SIGMUX SIGMUX 77 27 7 SIGMU</td> <td>5 SUBROUTINE RPCH(MPGS) CORE 27 5 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 6 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 7 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 10 THIS PROCRAM EMPLOYS THE RANGE DI CHANGE OTHANGE OTHANGE 27 11 THIS PROCRAM ARAYS CORE 25 12 TRAY NAME DEFINITION DIFFUNITION CORE 25 13 ARRAY NAME DEFINITION DIFFUNITION DIFFUNITION CORE 25 14 NUMERS OF THE INPUT LOAD SECTION DIFFUNITION DIFFUNITION CORE 25 15 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 15 CORE SIGMAX NALLEY SO THE INPUT LOAD SECTION CORE 25 16 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 16 SIGMAX THE NUMERS OF THE INPUT LOAD SECTION NESS + KK CORE 25 16 SIGMAX SIGMAX</td> <td>5 SUBROUTINE RPCHVIACS) CORE 257 5 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257 6 CORE 255 7 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 255 10 REAV NAME PROGRAM REPLOYS THE RANGE OF THENSIONS) CORE 255 11 REAV NAME FEAKS OF THE INUTI TON DTHENSIONS) CORE 255 11 REAV NAME PROGRAM MENTION DTHENSIONS) CORE 255 12 REAV NAME THE INUTI TON DTHENSIONS) CORE 255 13 REAV NAME PRAKS OF THE INUT LOD SECTRUM DTHENSIONS CORE 255 14 REAV NAME THE INUT LOD SECTRUM PROKS + KK CORE 255 14 REAVE VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 15 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PR</td> <td>5 CORE 257 10 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 11 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 12 REAL HOLE COUNTING METHOD TO GENER CORE 259 13 REAL HOLE COUNTING METHOD TO GENER CORE 259 14 FRECHAN FRON A GIVEN LOAD SPECTRUM CORE 259 15 REAL HOLE COUNTING METHOD TO GENER CORE 250 16 REAL HOLE COUNTING METHOD TO GENER CORE 250 17 REAL HOLE FOR THE INUTI CONSTRUMTION CORE 259 18 REAL HOLE FOR THE INUT CONSTRUMTION CORE 251 19 REAL HELO TO CHANGE STRUMT CORE 251 10 REAL HOLE FOR THE INUT CONSTRUMT CORE 251 11 REAL TO THE REAC OF THE INUT CONSTRUMT CORE 251 11 STGMAX HILE HOLE FOR THE INUT CONSTRUMT CORE 251 12 RECT K = THE HOUSE FOR THE INUT CONSTRUMT CORE 251 14 RECT K = THE HOUSE FOR THE RECT RECT CORE 251 15 RECT RECT RECT CORE 251 16 RECT RECT RECT CORE</td> <td>SUBROUTINE RPCHINES) CORE 237 SUBROUTINE RPCHINES) CORE 237 FILS PROGRAM EMPLOYS THE RANGE PARE CYCLE COUNTING METHOD TO GENER DORE 233 ARRAY NAME FUGNAL STORE OF THE RANGE OF HEINSTON CORE 233 ARRAY NAME DIFFINITION CORE 233 SIGMAX FELNUTION DIFFINITION CORE 233 ARRAY NAME PROGRAM MARAYS CORE 233 SIGMAX FELNUTION CORE 234 SIGMAX FELNUTION CORE 235 ARRAY NAME PROGRAM MALLENG CORE 235 SIGMAX FELNUTION CORE 235 SIGMAX FELNUTION SIGGRAM CORE SIGMAX SIGGRAM</td> <td>5 SUBGOUTINE RPONICARS1 CORE CORE 27 5 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 27 110 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 110 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 112 RRAY MARE DEFINITION NELOCO TO CHANGE CONTECTURIN DEFENSION DEFENSION 112 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN DEFENSION DEFENSION 113 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN PROS + KK DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC
STGMAN<!--</td--><td>SUBROUTINE RPCHARKS CORE 257 INIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM FROM A GIVEN LOUD SECTRUM DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS OF THE INPUT CONC. SCITAUM DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITION DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (FUNCT PROGRAM AND REAL FOR THE INPUT STORE AND REAL</td><td>35 SUBROUTINE RPHARKS CORE 5 SUBROUTINE RPHARKS CORE 6 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM ARKYS CORE 7 THIS PROGRAM ARKYS<!--</td--><td>Subsourtike RPCNINKS) CORE CORE</td><td>Subscurink Rechinks) Doe Subscurink Rechinks) State Cruck Counting Front A CLUC LOUNTING FELOD TO GRUE DORE THIS PROGRAM EMPLOYS THE RANGE OF ALL COUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD ARRAY MARE ARRAY MARE STGRUM STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY STGRUM ARRAY</td><td>SUBROUTINE RECHNARS) CORE <td< td=""><td>5 SUBGOUTINE RECHEMENSIS CORE 237 7 THIS PROGRAM EMPONSI THE RANGE FAIRS CYCLE CONVENDMENTION CORE 237 11 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 12 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 13 RERNY MARE DEFINITION DEFINITION CORE 14 RERNY MARE DEFINITION DEFINITION CORE 15 RERNY MARE DEFINITION DEFINITION CORE 16 RERNY MARE DEFINITION DEFINITION CORE 17 RERNY MARE DEFINITION DEFINITION CORE 18 RERNY CONFINENTION DEFINITION DEFINITION CORE 19 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 10 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 11 RERNY CONFILTION DEFINITION DEFINITION DEFINITION 12 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 13 RECOLUCUNITION DEFINITION DEFINITION 14 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 15 RERNY CONFILESTOR RERNY CONFILESTOR</td><td>5 SUBROUTINE REPRIVACE COME 25 111 SUBROUTINE REPRIVATIE ANGE ALT CYCLE CONNTINE METHOD TO GENER COME 25 112 COME 25 113 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 115 CONNEL CONNEL STATE 25 116 CONNEL CONNEL STATE 25 117 CONNEL CONNEL STATE 25 118 CONNEL CONNEL STATE 25 119 CONNEL CONNEL 27 110 CONNEL CONNEL 27 1111 CONNEL CONNEL 27 1111 CONNEL 27 27 1111 CONNEL 27</td><td>5 SUBROUTINE REPRIVATION DORE 25 7 THIS PROJERM FEMANYS DORE 25 10 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 11 DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 12 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 13 STEMM WHE PRAST OF ULE THATION NELLOD TO CHIRLE CHIRM FEMANY DORAGE 255 14 STEMM WHE PRAST OF THE THATION NELLOD TO CHIRLE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 16 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTA</td><td>SUBGUITINE RPENTORS
1115 PROGRAM RPEOTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCT CONTENT ON THE ANGE THE ANGE THE PROTECTION
AREA THE VIDERS OF THE INDUITION CORE STATE
AND THE ANGE THE REDUCT CORE STATE
AND THE AND THE ANGE THE REDUCT AND THE AND THE AND THE AND
AND THE AND THE</td><td> SUBGUITIKE RPCHURCS) SUBGUITIKE RPCHURCS) ATALYSTS SPECTRUM FRUNCS FIR ANGE PART FOLE DUNTING RETURD TO GAUE CORE
ATALYSTS SPECTRUM FRUNC STREAM A GUAL LOUS SPECTRUM ATALYSTS SPECTRUM FRUNC A GUAL LOUS SPECTRUM ATRAY MANE ARRAY MANE</td><td>5 SUBSOUTINE RPCHURKS) CORE 255 6 ALAYSTS SPECTRUM FRUNCY FILE ANGE PARS CYCLE CONTING NETHOD TO GENER CORE 255 11 PROGRAM ARAYS CURVED AL STANDARD CORE 12 RERN VALUE FROM AL STANDARD CURVED AL STANDARD 13 RERN VALUE FRAN AL STANDARD CURVED AL STANDARD 14 FRAN ALTICAL CONTING NETHON NELLICE TO CHARGE OF NERSTON CORE 15 RERN VALUE FRAN STANDARD CORE 16 STANDARD FRAN STANDARD CORE 17 STANDARD FRAN STANDARD CORE 18 STANDARD FRAN STANDARD CORE 19 STANDARD CORE CORE 10 STANDARD CORE CORE 11 STANDARD CORE CORE 12 STANDARD CORE CORE 13 STANDARD CORE CORE 14 CORE CORE CORE CORE 15 STANDARD CORE CORE </td></td<><td>5 SUBJOUTILE PROJUNE THE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE RUNCE TAE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE PROJUNE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE T</td><td>5 SUBJOUTIAE REPAIRANCES DOUNTIAE REPAIRANCES DOUNTIAE REPAIRANCES 1 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE REFERENCES 10 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE RETHON SECTION DOUNTIAE RETHON SECTION 11 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETHON DOUNTIAE RETHON 12 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETAIN DOUNTIAE RETAIN 13 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETAIN DOUNTIAE RETAIN</td><td>Subsort tick PERVISES Concent of the contrast stand Concent stand <thconcent stand<="" th=""> Concent stand Con</thconcent></td><td>SUBGOUTINE REMONENCIA 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE ROMAN STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBJOUTINE REMONE ROMAN STATE AND FOLL CONTINUE REMONE ROMAN 0000 FUE: REMONE ROMAN STATE AND FOLL ROMAN STATE AND FOLL ROMAN AND FOLL REMONE ROMAN 0000 STGRAM REMONE REMONE ROMAN AND FOLL REMONE ROMAN AND FOLD REMONE ROMAN AND FOLL REMONE ROMAN AND FOL REMON</td><td> SUBGUTIAK PERIMPRESI
HIST SPECIAL FEAR FARE PATE CTULE CONTINU NETHOD TO GREAT ONE CONTINUE NETHOD TO GREAT ON</td></td></td></td> | 5 5 5 6 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE PARK CYCLE COUNTING METHOD TO GENER CORE 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM EMPLOYS THE RANGE DIRENTOR 55 7 THIS PROGRAM HEROLOS 56 7 THIS PROGRAM MERATS 56 7 THE INPUT 57 7 THE INPUT ON NELDED TO CHARGE DIRENSIONS) 56 7 THE INPUT CAGE SCITRUM
 56 7 WALETS OF THE INPUT CAGE SCITRUM 56 7 KK = THE UNUBER OF DISTINGLIC CYCLES (ECCLUDING INPUT CYCL 56 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX VALLETS OF THE INPUT CAGE SCITRUM 76 7 SIGMUX SIGMUX 77 26 7 SIGMUX SIGMUX 77 27 7 SIGMU | 5 SUBROUTINE RPCH(MPGS) CORE 27 5 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 6 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 7 THIS PROCRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 27 10 THIS PROCRAM EMPLOYS THE RANGE DI CHANGE OTHANGE OTHANGE 27 11 THIS PROCRAM ARAYS CORE 25 12 TRAY NAME DEFINITION DIFFUNITION CORE 25 13 ARRAY NAME DEFINITION DIFFUNITION DIFFUNITION CORE 25 14 NUMERS OF THE INPUT LOAD SECTION DIFFUNITION DIFFUNITION CORE 25 15 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 15 CORE SIGMAX NALLEY SO THE INPUT LOAD SECTION CORE 25 16 CORE SIGMAX THE NUMERS OF THE INPUT LOAD SECTION CORE 25 16 SIGMAX THE NUMERS OF THE INPUT LOAD SECTION NESS + KK CORE 25 16 SIGMAX SIGMAX | 5 SUBROUTINE RPCHVIACS) CORE 257 5 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 257 6 CORE 255 7 TWIS PROGRAM EMPLOYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 255 10 REAV NAME PROGRAM REPLOYS THE RANGE OF THENSIONS) CORE 255 11 REAV NAME FEAKS OF THE INUTI TON DTHENSIONS) CORE 255 11 REAV NAME PROGRAM MENTION DTHENSIONS) CORE 255 12 REAV NAME THE INUTI TON DTHENSIONS) CORE 255 13 REAV NAME PRAKS OF THE INUT LOD SECTRUM DTHENSIONS CORE 255 14 REAV NAME THE INUT LOD SECTRUM PROKS + KK CORE 255 14 REAVE VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 15 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PROKS + KK CORE 255 16 RUCH VK = THE INUT LOD SECTRUM PR | 5 CORE 257 10 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 11 REAL HELDYS THE RANGE PAIR CYCLE COUNTING METHOD TO GENER CORE 259 12 REAL HOLE COUNTING METHOD TO GENER CORE 259 13 REAL HOLE COUNTING METHOD TO GENER CORE 259 14 FRECHAN FRON A GIVEN LOAD SPECTRUM CORE 259 15 REAL HOLE COUNTING METHOD TO GENER CORE 250 16 REAL HOLE COUNTING METHOD TO GENER CORE 250 17 REAL HOLE FOR THE INUTI CONSTRUMTION CORE 259 18 REAL HOLE FOR THE INUT CONSTRUMTION CORE 251 19 REAL HELO TO CHANGE STRUMT CORE 251 10 REAL HOLE FOR THE INUT CONSTRUMT CORE 251 11 REAL TO THE REAC OF THE INUT CONSTRUMT CORE 251 11 STGMAX HILE HOLE FOR THE INUT CONSTRUMT CORE 251 12 RECT K = THE HOUSE FOR THE INUT CONSTRUMT CORE 251 14 RECT K = THE HOUSE FOR THE RECT RECT CORE 251 15 RECT RECT RECT CORE 251 16 RECT RECT RECT CORE | SUBROUTINE RPCHINES) CORE 237 SUBROUTINE RPCHINES) CORE 237 FILS PROGRAM EMPLOYS THE RANGE PARE CYCLE COUNTING METHOD TO GENER DORE 233 ARRAY NAME FUGNAL STORE OF THE RANGE OF HEINSTON CORE 233 ARRAY NAME DIFFINITION CORE 233 SIGMAX FELNUTION DIFFINITION CORE 233 ARRAY NAME PROGRAM MARAYS CORE 233 SIGMAX FELNUTION CORE 234 SIGMAX FELNUTION CORE 235 ARRAY NAME PROGRAM MALLENG CORE 235 SIGMAX FELNUTION CORE 235 SIGMAX FELNUTION SIGGRAM CORE SIGMAX SIGGRAM | 5 SUBGOUTINE RPONICARS1 CORE CORE 27 5 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 27 110 THIS PROGRAM ERPONS THE RANGE PARS CYCLE COUNTING METHOD TO GENER CORE 27 110 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 111 RRAY MARE DEFINITION NELOCO TO CHANGE CONTINUE METHOD TO GENER CORE 27 112 RRAY MARE DEFINITION NELOCO TO CHANGE CONTECTURIN DEFENSION DEFENSION 112 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN DEFENSION DEFENSION 113 STGMAN THE NUBBLE OF THE REAS OF CHANGE CONTECTURIN PROS + KK DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC DEPENSION 114 STGMAN STGMAN RECOC STGMAN RECOC STGMAN </td <td>SUBROUTINE RPCHARKS CORE 257 INIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM FROM A GIVEN LOUD SECTRUM DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS OF THE INPUT CONC. SCITAUM DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITION DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (FUNCT PROGRAM AND REAL FOR THE INPUT STORE AND REAL</td> <td>35 SUBROUTINE RPHARKS CORE 5 SUBROUTINE RPHARKS CORE 6 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM ARKYS CORE 7 THIS PROGRAM ARKYS<!--</td--><td>Subsourtike RPCNINKS) CORE CORE</td><td>Subscurink Rechinks) Doe Subscurink Rechinks) State Cruck Counting Front A CLUC LOUNTING FELOD TO GRUE DORE THIS PROGRAM EMPLOYS THE RANGE OF ALL COUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD ARRAY MARE ARRAY MARE STGRUM STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY STGRUM ARRAY</td><td>SUBROUTINE RECHNARS) CORE <td< td=""><td>5 SUBGOUTINE RECHEMENSIS CORE 237 7 THIS PROGRAM EMPONSI THE RANGE FAIRS CYCLE CONVENDMENTION CORE 237 11 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 12 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 13 RERNY MARE DEFINITION DEFINITION CORE 14 RERNY MARE DEFINITION DEFINITION CORE 15 RERNY MARE DEFINITION DEFINITION CORE 16 RERNY MARE DEFINITION DEFINITION CORE 17 RERNY MARE DEFINITION DEFINITION CORE 18 RERNY CONFINENTION DEFINITION DEFINITION CORE 19 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 10 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 11 RERNY CONFILTION DEFINITION DEFINITION DEFINITION 12 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 13 RECOLUCUNITION DEFINITION DEFINITION 14 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 15 RERNY CONFILESTOR RERNY CONFILESTOR</td><td>5 SUBROUTINE REPRIVACE COME 25 111 SUBROUTINE REPRIVATIE ANGE ALT CYCLE CONNTINE METHOD TO GENER COME 25 112 COME 25 113 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 115 CONNEL CONNEL STATE 25 116 CONNEL CONNEL STATE 25 117 CONNEL CONNEL STATE 25
118 CONNEL CONNEL STATE 25 119 CONNEL CONNEL 27 110 CONNEL CONNEL 27 1111 CONNEL CONNEL 27 1111 CONNEL 27 27 1111 CONNEL 27</td><td>5 SUBROUTINE REPRIVATION DORE 25 7 THIS PROJERM FEMANYS DORE 25 10 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 11 DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 12 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 13 STEMM WHE PRAST OF ULE THATION NELLOD TO CHIRLE CHIRM FEMANY DORAGE 255 14 STEMM WHE PRAST OF THE THATION NELLOD TO CHIRLE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 16 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTA</td><td>SUBGUITINE RPENTORS
1115 PROGRAM RPEOTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCT CONTENT ON THE ANGE THE ANGE THE PROTECTION
AREA THE VIDERS OF THE INDUITION CORE STATE
AND THE ANGE THE REDUCT CORE STATE
AND THE AND THE ANGE THE REDUCT AND THE AND THE AND THE AND
AND THE AND THE</td><td> SUBGUITIKE RPCHURCS) SUBGUITIKE RPCHURCS) ATALYSTS SPECTRUM FRUNCS FIR ANGE PART FOLE DUNTING RETURD TO GAUE CORE
ATALYSTS SPECTRUM FRUNC STREAM A GUAL LOUS SPECTRUM ATALYSTS SPECTRUM FRUNC A GUAL LOUS SPECTRUM ATRAY MANE ARRAY MANE</td><td>5 SUBSOUTINE RPCHURKS) CORE 255 6 ALAYSTS SPECTRUM FRUNCY FILE ANGE PARS CYCLE CONTING NETHOD TO GENER CORE 255 11 PROGRAM ARAYS CURVED AL STANDARD CORE 12 RERN VALUE FROM AL STANDARD CURVED AL STANDARD 13 RERN VALUE FRAN AL STANDARD CURVED AL STANDARD 14 FRAN ALTICAL CONTING NETHON NELLICE TO CHARGE OF NERSTON CORE 15 RERN VALUE FRAN STANDARD CORE 16 STANDARD FRAN STANDARD CORE 17 STANDARD FRAN STANDARD CORE 18 STANDARD FRAN STANDARD CORE 19 STANDARD CORE CORE 10 STANDARD CORE CORE 11 STANDARD CORE CORE 12 STANDARD CORE CORE 13 STANDARD CORE CORE 14 CORE CORE CORE CORE 15 STANDARD CORE CORE </td></td<><td>5 SUBJOUTILE PROJUNE THE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE RUNCE TAE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE PROJUNE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE T</td><td>5 SUBJOUTIAE REPAIRANCES DOUNTIAE REPAIRANCES DOUNTIAE REPAIRANCES 1 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE REFERENCES 10 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE RETHON SECTION DOUNTIAE RETHON SECTION 11 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETHON DOUNTIAE RETHON 12 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETAIN DOUNTIAE RETAIN 13 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETAIN DOUNTIAE RETAIN</td><td>Subsort tick PERVISES Concent of the contrast stand Concent stand <thconcent stand<="" th=""> Concent stand Con</thconcent></td><td>SUBGOUTINE REMONENCIA 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE ROMAN STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBJOUTINE REMONE ROMAN STATE AND FOLL CONTINUE REMONE ROMAN 0000 FUE: REMONE ROMAN STATE AND FOLL ROMAN STATE AND FOLL ROMAN AND FOLL REMONE ROMAN 0000 STGRAM REMONE REMONE ROMAN AND FOLL REMONE ROMAN AND FOLD REMONE ROMAN AND FOLL REMONE ROMAN AND FOL REMON</td><td> SUBGUTIAK PERIMPRESI
HIST SPECIAL FEAR FARE PATE CTULE CONTINU NETHOD TO GREAT ONE CONTINUE NETHOD TO GREAT ON</td></td></td> | SUBROUTINE RPCHARKS CORE 257 INIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE PART CYCLE COUNTING METHOD TO GENER CORE 207 THIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION CORE 253 INIS PROGRAM FROM A GIVEN LOUD SECTRUM DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS THE RANGE DEFINITION DTHENSIONS CORE 253 INIS PROGRAM EMPLOYS OF THE INPUT CONC. SCITAUM DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITION DTHENSIONS CORE 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (EXCLUDING FUNCT CORE 255 255 255 INIS PROGRAM MALLE DEFINITIONAL COLLES (FUNCT PROGRAM AND REAL FOR THE INPUT STORE AND REAL | 35 SUBROUTINE RPHARKS CORE 5 SUBROUTINE RPHARKS CORE 6 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS THE ROWA GIVEN LODIS SECTION CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM EMONS CORE 7 THIS PROGRAM ARKYS CORE 7 THIS PROGRAM ARKYS </td <td>Subsourtike RPCNINKS) CORE CORE</td> <td>Subscurink Rechinks) Doe Subscurink Rechinks) State Cruck Counting Front A CLUC LOUNTING FELOD TO GRUE DORE THIS PROGRAM EMPLOYS THE RANGE OF ALL COUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD ARRAY MARE ARRAY MARE STGRUM STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY STGRUM ARRAY</td> <td>SUBROUTINE RECHNARS) CORE <td< td=""><td>5 SUBGOUTINE RECHEMENSIS CORE 237 7 THIS PROGRAM EMPONSI THE RANGE FAIRS CYCLE CONVENDMENTION CORE 237 11 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 12 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 13 RERNY MARE DEFINITION DEFINITION CORE 14 RERNY MARE DEFINITION DEFINITION CORE 15 RERNY MARE DEFINITION DEFINITION CORE 16 RERNY MARE DEFINITION DEFINITION CORE 17 RERNY MARE DEFINITION DEFINITION CORE 18 RERNY CONFINENTION DEFINITION DEFINITION CORE 19 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 10 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 11 RERNY CONFILTION DEFINITION DEFINITION DEFINITION 12 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 13 RECOLUCUNITION DEFINITION DEFINITION 14 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 15 RERNY CONFILESTOR RERNY CONFILESTOR</td><td>5 SUBROUTINE REPRIVACE COME 25 111 SUBROUTINE REPRIVATIE ANGE ALT CYCLE CONNTINE METHOD TO GENER COME 25 112 COME 25 113 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 115 CONNEL CONNEL STATE 25 116 CONNEL CONNEL STATE 25 117 CONNEL CONNEL STATE 25 118 CONNEL CONNEL STATE 25 119 CONNEL CONNEL 27 110 CONNEL CONNEL 27 1111 CONNEL CONNEL 27 1111 CONNEL 27 27 1111 CONNEL 27</td><td>5 SUBROUTINE REPRIVATION DORE 25 7 THIS PROJERM FEMANYS DORE 25 10 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 11 DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 12 THIS PROJERM FEMANY DORAGE
CHIRM FEMANY DORAGE CHIRM FEMANY 13 STEMM WHE PRAST OF ULE THATION NELLOD TO CHIRLE CHIRM FEMANY DORAGE 255 14 STEMM WHE PRAST OF THE THATION NELLOD TO CHIRLE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 16 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTA</td><td>SUBGUITINE RPENTORS
1115 PROGRAM RPEOTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCT CONTENT ON THE ANGE THE ANGE THE PROTECTION
AREA THE VIDERS OF THE INDUITION CORE STATE
AND THE ANGE THE REDUCT CORE STATE
AND THE AND THE ANGE THE REDUCT AND THE AND THE AND THE AND
AND THE AND THE</td><td> SUBGUITIKE RPCHURCS) SUBGUITIKE RPCHURCS) ATALYSTS SPECTRUM FRUNCS FIR ANGE PART FOLE DUNTING RETURD TO GAUE CORE
ATALYSTS SPECTRUM FRUNC STREAM A GUAL LOUS SPECTRUM ATALYSTS SPECTRUM FRUNC A GUAL LOUS SPECTRUM ATRAY MANE ARRAY MANE</td><td>5 SUBSOUTINE RPCHURKS) CORE 255 6 ALAYSTS SPECTRUM FRUNCY FILE ANGE PARS CYCLE CONTING NETHOD TO GENER CORE 255 11 PROGRAM ARAYS CURVED AL STANDARD CORE 12 RERN VALUE FROM AL STANDARD CURVED AL STANDARD 13 RERN VALUE FRAN AL STANDARD CURVED AL STANDARD 14 FRAN ALTICAL CONTING NETHON NELLICE TO CHARGE OF NERSTON CORE 15 RERN VALUE FRAN STANDARD CORE 16 STANDARD FRAN STANDARD CORE 17 STANDARD FRAN STANDARD CORE 18 STANDARD FRAN STANDARD CORE 19 STANDARD CORE CORE 10 STANDARD CORE CORE 11 STANDARD CORE CORE 12 STANDARD CORE CORE 13 STANDARD CORE CORE 14 CORE CORE CORE CORE 15 STANDARD CORE CORE </td></td<><td>5 SUBJOUTILE PROJUNE THE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE RUNCE TAE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE PROJUNE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE T</td><td>5 SUBJOUTIAE REPAIRANCES DOUNTIAE REPAIRANCES DOUNTIAE REPAIRANCES 1 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE REFERENCES 10 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE RETHON SECTION DOUNTIAE RETHON SECTION 11 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETHON DOUNTIAE RETHON 12 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETAIN DOUNTIAE RETAIN 13 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETAIN DOUNTIAE RETAIN</td><td>Subsort tick PERVISES Concent of the contrast stand Concent stand <thconcent stand<="" th=""> Concent stand Con</thconcent></td><td>SUBGOUTINE REMONENCIA 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE ROMAN STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBJOUTINE REMONE ROMAN STATE AND FOLL CONTINUE REMONE ROMAN 0000 FUE: REMONE ROMAN STATE AND FOLL ROMAN STATE AND FOLL ROMAN AND FOLL REMONE ROMAN 0000 STGRAM REMONE REMONE ROMAN AND FOLL REMONE ROMAN AND FOLD REMONE ROMAN AND FOLL REMONE ROMAN AND FOL REMON</td><td> SUBGUTIAK PERIMPRESI
HIST SPECIAL FEAR FARE PATE CTULE CONTINU NETHOD TO GREAT ONE CONTINUE NETHOD TO GREAT ON</td></td> | Subsourtike RPCNINKS) CORE CORE | Subscurink Rechinks) Doe Subscurink Rechinks) State Cruck Counting Front A CLUC LOUNTING FELOD TO GRUE DORE THIS PROGRAM EMPLOYS THE RANGE OF ALL COUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD TO GRUE DORE AMALYSIS SPECTRUM FROM A CLUC LOUNTING FELOD ARRAY MARE ARRAY MARE STGRUM STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY MARE STGRUM ARRAY STGRUM ARRAY | SUBROUTINE RECHNARS) CORE CORE <td< td=""><td>5 SUBGOUTINE RECHEMENSIS CORE 237 7 THIS PROGRAM EMPONSI THE RANGE FAIRS CYCLE CONVENDMENTION CORE 237 11 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 12 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 13 RERNY MARE DEFINITION DEFINITION CORE 14 RERNY MARE DEFINITION DEFINITION CORE 15 RERNY MARE DEFINITION DEFINITION CORE 16 RERNY MARE DEFINITION DEFINITION CORE 17 RERNY MARE DEFINITION DEFINITION CORE 18 RERNY CONFINENTION DEFINITION DEFINITION CORE 19 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 10 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 11 RERNY CONFILTION DEFINITION DEFINITION DEFINITION 12 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 13 RECOLUCUNITION DEFINITION DEFINITION 14 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 15 RERNY CONFILESTOR RERNY CONFILESTOR</td><td>5 SUBROUTINE REPRIVACE COME 25 111 SUBROUTINE REPRIVATIE ANGE ALT CYCLE CONNTINE METHOD TO GENER COME 25 112 COME 25 113 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 115 CONNEL CONNEL STATE 25 116 CONNEL CONNEL STATE 25 117 CONNEL CONNEL STATE 25 118 CONNEL CONNEL STATE 25 119 CONNEL CONNEL 27 110 CONNEL CONNEL 27 1111 CONNEL CONNEL 27 1111 CONNEL 27 27 1111 CONNEL 27</td><td>5 SUBROUTINE REPRIVATION DORE 25 7 THIS PROJERM FEMANYS DORE 25 10 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 11 DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 12 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 13 STEMM WHE PRAST OF ULE THATION NELLOD TO CHIRLE CHIRM FEMANY DORAGE 255 14 STEMM WHE PRAST OF THE THATION NELLOD TO CHIRLE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 16 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTA</td><td>SUBGUITINE RPENTORS
1115 PROGRAM RPEOTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCT CONTENT ON THE ANGE THE ANGE THE PROTECTION
AREA THE VIDERS OF THE INDUITION CORE STATE
AND THE ANGE THE REDUCT CORE STATE
AND THE AND THE ANGE THE REDUCT AND THE AND THE AND THE AND
AND THE AND THE</td><td> SUBGUITIKE RPCHURCS) SUBGUITIKE RPCHURCS) ATALYSTS SPECTRUM FRUNCS FIR ANGE PART FOLE DUNTING RETURD TO GAUE CORE
ATALYSTS SPECTRUM FRUNC STREAM A GUAL LOUS SPECTRUM ATALYSTS SPECTRUM FRUNC A GUAL LOUS SPECTRUM ATRAY MANE ARRAY MANE</td><td>5 SUBSOUTINE RPCHURKS) CORE 255 6 ALAYSTS SPECTRUM FRUNCY FILE ANGE PARS CYCLE CONTING NETHOD TO GENER CORE 255 11 PROGRAM ARAYS CURVED AL STANDARD CORE 12 RERN VALUE FROM AL STANDARD CURVED AL STANDARD 13 RERN VALUE FRAN AL STANDARD CURVED AL STANDARD 14 FRAN ALTICAL CONTING NETHON NELLICE TO CHARGE OF NERSTON CORE 15 RERN VALUE FRAN
STANDARD CORE 16 STANDARD FRAN STANDARD CORE 17 STANDARD FRAN STANDARD CORE 18 STANDARD FRAN STANDARD CORE 19 STANDARD CORE CORE 10 STANDARD CORE CORE 11 STANDARD CORE CORE 12 STANDARD CORE CORE 13 STANDARD CORE CORE 14 CORE CORE CORE CORE 15 STANDARD CORE CORE </td></td<> <td>5 SUBJOUTILE PROJUNE THE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE RUNCE TAE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE PROJUNE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE T</td> <td>5 SUBJOUTIAE REPAIRANCES DOUNTIAE REPAIRANCES DOUNTIAE REPAIRANCES 1 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE REFERENCES 10 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE RETHON SECTION DOUNTIAE RETHON SECTION 11 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETHON DOUNTIAE RETHON 12 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETAIN DOUNTIAE RETAIN 13 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETAIN DOUNTIAE RETAIN</td> <td>Subsort tick PERVISES Concent of the contrast stand Concent stand <thconcent stand<="" th=""> Concent stand Con</thconcent></td> <td>SUBGOUTINE REMONENCIA 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE ROMAN STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBJOUTINE REMONE ROMAN STATE AND FOLL CONTINUE REMONE ROMAN 0000 FUE: REMONE ROMAN STATE AND FOLL ROMAN STATE AND FOLL ROMAN AND FOLL REMONE ROMAN 0000 STGRAM REMONE REMONE ROMAN AND FOLL REMONE ROMAN AND FOLD REMONE ROMAN AND FOLL REMONE ROMAN AND FOL REMON</td> <td> SUBGUTIAK PERIMPRESI
HIST SPECIAL FEAR FARE PATE CTULE CONTINU NETHOD TO GREAT ONE CONTINUE NETHOD TO GREAT ON</td> | 5 SUBGOUTINE RECHEMENSIS CORE 237 7 THIS PROGRAM EMPONSI THE RANGE FAIRS CYCLE CONVENDMENTION CORE 237 11 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 12 THIS PROGRAM EMPONSI THE RANGE CONVENDMENTION CORE 237 13 RERNY MARE DEFINITION DEFINITION CORE 14 RERNY MARE DEFINITION DEFINITION CORE 15 RERNY MARE DEFINITION DEFINITION CORE 16 RERNY MARE DEFINITION DEFINITION CORE 17 RERNY MARE DEFINITION DEFINITION CORE 18 RERNY CONFINENTION DEFINITION DEFINITION CORE 19 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 10 RERNY CONFINENTION DEFINITION DEFINITION DEFINITION 11 RERNY CONFILTION DEFINITION DEFINITION DEFINITION 12 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 13 RECOLUCUNITION DEFINITION DEFINITION 14 RERNY CONFILESTOR THE RUNUT SECTION DEFINITION 15 RERNY CONFILESTOR RERNY CONFILESTOR | 5 SUBROUTINE REPRIVACE COME 25 111 SUBROUTINE REPRIVATIE ANGE ALT CYCLE CONNTINE METHOD TO GENER COME 25 112 COME 25 113 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 114 FRANTAN CONNEL STATE 25 115 CONNEL CONNEL STATE 25 116 CONNEL CONNEL STATE 25 117 CONNEL CONNEL STATE 25 118 CONNEL CONNEL STATE 25 119 CONNEL CONNEL 27 110 CONNEL CONNEL 27 1111 CONNEL CONNEL 27 1111 CONNEL 27 27 1111 CONNEL 27 | 5 SUBROUTINE REPRIVATION DORE 25 7 THIS PROJERM FEMANYS DORE 25 10 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 11 DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 12 THIS PROJERM FEMANY DORAGE CHIRM FEMANY DORAGE CHIRM FEMANY 13 STEMM WHE PRAST OF ULE THATION NELLOD TO CHIRLE CHIRM FEMANY DORAGE 255 14 STEMM WHE PRAST OF THE THATION NELLOD TO CHIRLE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 15 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 16 STEMM WHE PRAST OF THE THATION NECLED TO MAKE THATION DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 17 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTAGE 255 DOTAGE 255 18 STEMM WHE RETAIN TO WEELE THATION DOTA | SUBGUITINE RPENTORS
1115 PROGRAM RPEOTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCTS THE ANGE PART CYCLE CONNTING RETHOD TO GENER CORE
ANALYSIS SPECTRUM REDUCT CONTENT ON THE ANGE THE ANGE THE PROTECTION
AREA THE VIDERS OF THE INDUITION CORE STATE
AND THE ANGE THE REDUCT CORE STATE
AND THE AND THE ANGE THE REDUCT AND THE AND THE AND THE AND
AND THE AND THE | SUBGUITIKE RPCHURCS) SUBGUITIKE RPCHURCS) ATALYSTS SPECTRUM FRUNCS FIR ANGE PART FOLE DUNTING RETURD TO GAUE CORE
ATALYSTS SPECTRUM FRUNC STREAM A GUAL LOUS SPECTRUM ATALYSTS SPECTRUM FRUNC A GUAL LOUS SPECTRUM ATRAY MANE ARRAY MANE | 5 SUBSOUTINE RPCHURKS) CORE 255 6 ALAYSTS SPECTRUM FRUNCY FILE ANGE PARS CYCLE CONTING NETHOD TO GENER CORE 255 11 PROGRAM ARAYS CURVED AL STANDARD CORE 12 RERN VALUE FROM AL STANDARD CURVED AL STANDARD 13 RERN VALUE FRAN AL STANDARD CURVED AL STANDARD 14 FRAN ALTICAL CONTING NETHON NELLICE TO CHARGE OF NERSTON CORE 15 RERN VALUE FRAN STANDARD CORE 16 STANDARD FRAN STANDARD CORE 17 STANDARD FRAN STANDARD CORE 18 STANDARD FRAN STANDARD CORE 19 STANDARD CORE CORE 10 STANDARD CORE CORE 11 STANDARD CORE CORE 12 STANDARD CORE CORE 13 STANDARD CORE CORE 14 CORE CORE CORE CORE 15 STANDARD CORE CORE | 5 SUBJOUTILE PROJUNE THE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE RUNCE TAE RUNCE TAE RUNCE TAE OFFLE DOWNTIME RETHON TO GREEK DOWNER THE PROJUNE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE RUNCE TAE RUNCE TAE RUNCE T | 5 SUBJOUTIAE REPAIRANCES DOUNTIAE REPAIRANCES DOUNTIAE REPAIRANCES 1 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE REFERENCES 10 INTS FORDINA ENDORE THE RANKE DATOLE DOUNTIAE RETHON TO GREEK DONE DOUNTIAE REFERENCES DOUNTIAE RETHON SECTION DOUNTIAE RETHON SECTION 11 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETHON DOUNTIAE RETHON 12 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETHON DOUNTIAE RETAIN DOUNTIAE RETAIN 13 STEAM MAKE DEFENDITION NELDED TO CHANCE OTHER SECTION DOUNTIAE RETAIN DOUNTIAE RETAIN | Subsort tick PERVISES Concent of the contrast stand Concent stand <thconcent stand<="" th=""> Concent stand Con</thconcent> | SUBGOUTINE REMONENCIA 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUL: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE RANK STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBGOUTINE REMONE ROMAN STATE AND FOLL CONTING WITHOU TO GEGE ROMAN 0000 FUE: SUBJOUTINE REMONE ROMAN STATE AND FOLL CONTINUE REMONE ROMAN 0000 FUE: REMONE ROMAN STATE AND FOLL ROMAN STATE AND FOLL ROMAN AND FOLL REMONE ROMAN 0000 STGRAM REMONE REMONE ROMAN AND FOLL REMONE ROMAN AND FOLD REMONE ROMAN AND FOLL REMONE ROMAN AND FOL REMON | SUBGUTIAK PERIMPRESI
HIST SPECIAL FEAR FARE PATE CTULE CONTINU NETHOD TO GREAT ONE CONTINUE NETHOD TO GREAT ON |

1 = 100 1 = 1,NPKS CYC(I) .6E. 1.0) 60 T0 100 IGMX(I) IGMX(I) IGMX(I) IGMX(I) J = 1 + 1 + 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 -	AD C C C C C C C C C C C C C C C C C C C	9 M M M M M M M M M M M M M M M M M M M		. • • ₁ • •
<pre>I = 1,NPKS CYC(I) .6E. 1.0) 60 T0 100 IGMAX(I) IGMAX(I) IGMAX(I) YGGEN(X1,X2 ,RNCYC(I),NSTEP(I)) J = I J = 1 J = 1 J = 1 NPKS - JMAX AX .EG. 0) 60 T0 200 6:231 (ISAVE(K), K = 1,JMAX) INTGE.2760 T0 101 INTGE.2760 T0 100 T0 100 INTGE.2760 T0 100 T0 100 T0 100 T0 T0 100 T0 100 T0</pre>	2000 2000 2000 2000 2000 2000 2000 200	8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8	· · · · · · · · · · · · · · · · · · ·	. • • ₁ • •
TCGENIX, X2 , RNCYC(I), NSTEP(I))) = I + 1 + 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 -	AD C C C C C C C C C C C C C C C C C C C	9 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	· · · · · · · · · · · · · · · · · · ·	r , , .
+ 1 UE NPKS - JHAX = NPKS - JHAX AX.•EQ. 0) GO TO 200 AX.•EQ. 0) GO TO 200 INT.GE.27GO TO 101 INT.GE.27GO TO 101 INT.GE.27GO TO 101 INT.GE.27GO TO 101 INT.GE.27GO TO 101 INT.GE.27GO TO 101 UM HHOSE COUNTER K IS LESS THAN 1.0//(1717)) UM HO	2AD C C C C C C C C C C C C C C C C C C C	9 M M M M M M M M M M M M M M M M M M M		• : •
= NPKS - JHAX = NPKS - JHAX 6/23) (ISAVE(K), K = 1,JMAX) INT.GE.2)GO TO 101 (1H0,93HSTEP NUHBERS OF THOSE PEAKS AND VALLEYS IN THE LO UM HHOSE COUNTER K IS LESS THAN 1.0//(1717)) UE UE U = 1,JMAX J = 1,JMAX AVE(J) - (J-1) NPKS - J NPKN GO TO 110 CT1 = 1,NPKN CT1 = 5TGMAXTT++1)	CORE CORE CORE CORE CORE CORE CORE CORE	8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8		: .
INT.GE.2)GO TO 101 (140,93HSTEP NUMBERS OF THOSE PEAKS ANJ VALLEYS IN THE LO Um MHOSE COUNTER K IS LESS THAN 1.0//(1717)) UE J = 1,JMAX J = 1,JMAX AVE(J) - (J-1) NPKS - J NPKN GO TO 110 .EQ. NPKN GO TO 110 .TI = 1,NPKN	DAD CORE CORE CORE CORE CORE CORE CORE CORE	8 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9		
UE J = 1,JMAX AVE(J) - (J-1) NPKS - J -EQ. NPKN) GO TO 110 II = IJNPKN TT = STGMAXTT+1)	CORE CORE CORE CORE	8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8		
NPKS - J .EQ. NPKN) GO TO 110 II = I,NPKN ITI = SIGMAXIT+11	CORE CORE CORE	2 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7		
LL Ξ L9NPKN (TT) Ξ STGMΔY(TT+1)		4 00 00 F		
(11) = SIGMIN(11+1)	CORE CORE	550		
II) = NSTEP(II+1) II) = PNCYC(II+1)	CORE	337		•
	CORE CORE	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0		
ΩE	CORE CORE	341 342		
HROUGH THE LOAD SPECTRUM DATA - COMBINE STEPS WITH IDENTI Lleys Which occur consecutively	CCAL CORE CORE	5 t t 1 1 t 1 1 t 1		
1157011 C - 1	CORE CORE	842 842 842		
L - CINTASU Ghax(i) •NE• Sighax(i-1)) G0 T0 300 Ghin(i) •NE• Sighin(t-1)) G0 T0 300	CORE CORE CORE	84N 84N		
<pre>J) = I I-1) = RUCYC(I-1) + RUCYC(I)</pre>	CORE	350 351		
+ 1 UE	CORE CORE	352		
.EQ. 1) GO TO 6000	CORE	101 101 101 101		
J = 1,JMAS AVE(J) - (J-1)	CORE	356 357		•
•EQ• NPKSN - J •EQ• NPKN) GO TO 311	CORE	356 356		
II = I,NPKN (II) = SIGMAX(II+1)	CORE CORE	350 361		
(II) = SIGHIN(II+1) II) = NSTEP(II+1)	CORE	352		
	CORE	305		

311 CONTINUE NPKSN = NPKSN - JMAS	CORE Core	366 357		
C RANGE PAIR COUNT THE ADJUSTED LOAD SPECTRUM	CORE CORE	358 359		
6000 I = 1	CORE	371		
KB = 0 L = JM4X	CORE	372		
	CORE	374		
KMAX = D	CORE	375		
LR = 0	CORE	376		
	CORE	377		
I I CANCICIT (1) (1) 100 0 0 400 I F (KB) KC 10 KC 10 400	CORE	378		
X1 = SIGMAX(I)		5/9		
xz = sighin(i)	2000			
IND1 = NSTEP(I)	1400	101		
IND2 = IND1	CORE	2.82		
	CORE	19 C 19 C 19 C		
(XB II 1	CORE	385		
	CORE	396		
$5 \times 3 = \text{SIGMAX}(1)$	CORE	397	:	
	CORE	398		
TRUS = NSTEP(I)	CORE	339		
LICH H LNOS	CORE	390		
U = XVWX	CORE	391		
K31 = 0	CUAE 2025	242		
IF (RNCYC(I) .EQ. 1.0) GO TO 6	CORF	165		
KEY = 1	CORE	395		
χ IND = 1	CORE	396		
60 TO 415	CORE	397		
	CORE	398		
	CORE	399		
A DECTRETATIONS AND	CORE	100		
	CORE	101		
	CORE	402		
IF (\$MIN_NE. 1) 60 TO 36		9 J J J		
IF (I .LE. NPKSN) GO TO 5				
RES(LR+1) = X1	COPT	100		
RES(LR+2) = X2	CORE	407		
INDEX(LR+1) = IND1	CORE	408		
INDEX(LR+2) = IND2	CORE	409		
	CORE	410		
	CORE	411		
	CORE	412		
IF (I .LE. NPKSN) GO TO 31		413 141		
$R\in S(LR+1) = X1$	CORF	1 1 1		
RES(LR+2) = X2	CORE	416		
res(LR+3) = X3	CORE	417		
INDEX(LK+1) = IND1 INDEX(1943) - IND2	CORE	418		
	CORE	419		
INUCAUCKTOJ = LNUC	CORE	420		

ł

ļ

31 L (MM = (H + 3) 0000 001 32 C (T 7 2000) C (T 7 2000) 0000 33 C (T 7 2000) C (T 7 2000) C (T 7 2000) 33 C (T 7 2000) C (T 7 2000) C (T 7 2000) 34 C (T 7 10) C (T 7 2000) C (T 7 2000) 35 C (T 7 2000) C (T 7 2000) C (T 7 2000) 36 C (T 7 2000) C (T 7 2000) C (T 7 2000) 37 C (T 1 2 0) C (T 1 2 0) C (T 1 2 0) 38 C (T 1 2 0) C (T 1 2 0) C (T 1 2 0) 39 C (T 1 2 0) C (T 1 2 0) C (T 1 2 0) 39 C (T 1 2 0) C (T 1 2 0) C (T 1 2 0) 39 C (T 1 2 0) C (T 1 2 0) C (T 1 2 0) 31 C (T 1 2 0) C (T 1 2 0) C (T 1 2 0) 31 C (T 1 2 0) C (T 1 2 0) C (T 1 2 0) 31 C (T 1 2 0) C (T 1 2 0) C (T 1 2 0) 32 C (T 1 2 0) C (T 1 2 0) C (T 1 2 0) 33 C (T 1 2 0) C (T 1 2 0) C (T 1 2 0) 34 C (T 1 2 0) C (T 1 2 0) C (T 1 2 0) 34 C (T 1 2 0) C (T 1 2 0) <tdc (t="" 0)<="" 1="" 2="" td=""> 34 C (</tdc>	31 LUMM * (# + 3) COME CM 32 LUMM * (# + 3) COME CM 33 FO SOUND COME CM 34 COME CM CM CM 35 FO TO COME CM 36 FO CM CM CM 37 FO CM CM CM 38 FO CM CM CM 39 CM S CM CM 30 CM S CM CM 31 CM CM CM CM 32 CM S CM CM 34 CM CM CM CM 34 CM	31 10000 00000 00000 11 10000 00000 00000 00000 12 10000 10000 00000 00000 00000 12 10000 10000 000000 00000 00000 <td< th=""><th>31 Correct State Correct State</th><th>31 Lem X = LX + 3 00000 452 32 L = STGRM(T) 00000 452 33 F (HW/VCL) cf (H/VCL) 00000 452 33 F (HW/VCL) cf (H/VCL) 00000 452 34 F (HW/VCL) cf (H/VCL) 00000 452 35 F (HW/VCL) cf (H/VCL) 00000 452 36 F (H/VCL) cf (H/VCL) 00000 452 37 F (H/VCL) cf (H/VCL) 00000 452 38 S (H/VCL) 00000 452 00000 452 38 S (H/VCL) 00000 00000 452 00000 452 38 S (H/VCL) 00000 00000 452 00000 452 400 CU TO 2 S (H/VCL) 00000 452 00000 452 400 CU TO 2 S (H/VC L) 00000 452 00000 452 411 S (H/VC L) S (H/VC L) 00000 452 00000 452 411 S (H/VC L) S (H/VC L)</th><th>31 Unix = Ux + 3 0000 421 32 Unix = Streak(r) 0000 421 12 Unix = Streak(r) 0000 421 13 Unix = Streak(r) 0000 421 13 Unix = Streak(r) 0000 421 13 Unix = Streak(r) 0000 421 14 Unix = Streak(r) 0000 421 15 Unix = Streak(r) 0000 421 16 Unix = Streak(r) 0000 421<</th><th></th><th></th><th></th><th></th><th>r</th></td<>	31 Correct State	31 Lem X = LX + 3 00000 452 32 L = STGRM(T) 00000 452 33 F (HW/VCL) cf (H/VCL) 00000 452 33 F (HW/VCL) cf (H/VCL) 00000 452 34 F (HW/VCL) cf (H/VCL) 00000 452 35 F (HW/VCL) cf (H/VCL) 00000 452 36 F (H/VCL) cf (H/VCL) 00000 452 37 F (H/VCL) cf (H/VCL) 00000 452 38 S (H/VCL) 00000 452 00000 452 38 S (H/VCL) 00000 00000 452 00000 452 38 S (H/VCL) 00000 00000 452 00000 452 400 CU TO 2 S (H/VCL) 00000 452 00000 452 400 CU TO 2 S (H/VC L) 00000 452 00000 452 411 S (H/VC L) S (H/VC L) 00000 452 00000 452 411 S (H/VC L) S (H/VC L)	31 Unix = Ux + 3 0000 421 32 Unix = Streak(r) 0000 421 12 Unix = Streak(r) 0000 421 13 Unix = Streak(r) 0000 421 13 Unix = Streak(r) 0000 421 13 Unix = Streak(r) 0000 421 14 Unix = Streak(r) 0000 421 15 Unix = Streak(r) 0000 421 16 Unix = Streak(r) 0000 421<					r
31 XC = STER(1) COR COR 37 TAU = T COR COR 13 TAUNG = T COR COR 14 COR COR COR 15 TAUNG = T COR COR 16 COR COR COR 17 COR COR COR 18 CO COR COR 19 CO COR COR 10 CO COR COR 1100 S COUR COR 1100 S COUR COR 1100 S COUR COR 1100 S COUR COR 1101 S COUR COR 1101 S COUR COR 1101 S S COUR COR 1101 S COUR COR COR 1101 S COUR COR COR 1101 S COUR COR COR 1101 S	31 X::::::::::::::::::::::::::::::::::::	21 10.0 <	11 Non = Strenkt(1) Contact Strenkt(1) 12 Frieder(1) .cf. 1.10 Contact Strenkt(1) 13 Frieder(1) .cf. 1.10 Contact Strenkt(1) 14 .cf. 1 .cf. 1 Contact Strenkt(1) 15 Frieder(1) .cf. 1.10 Contact Strenkt(1) 16 Frieder(1) .cf. 1.10 Contact Strenkt(1) 17 Contact Strenkt(1) Contact Strenkt(1) Contact Strenkt(1) 18 Strenkt(1) Contact Strenkt(1) Contact Strenkt(1) 19 Strenkt(1) Contact Strenkt(1) Contact Strenkt(1) 10 Strenkt(1) Strenkt(1) Contact Strenkt(1) 10 Stren	11 X: X	11 10. 10	LRMAX = LR + 3	CORE CODE	-21		
13 File a file file (1) 0000 450 13 File a file (1) 0010 450 14 File a file (1) 0010 450 15 File a file (1) 0010 450 16 File (1) 0010 450 16 File (1) 0010 450 16 File (1) 0000 450 17 File (1) 0000 450 17 File (1) 0000 450 18 File (1) 0000 450 19 File (1) 0000 450 100 File (1) 0000 450 101 File (1) 100 450 </td <td>13 File Sisterio Control Contro Control Control</td> <td>The image is the image is</td> <td>Mix = 1 0000 0000 0000 12 Fi = 1 0000 0000 0000 13 Fi = 1 0000 0000 0000 14 Fi = 1 0000 0000 0000 12 Fi = 1 0000 0000 0000 14 Fi = 1 0000 0000 0000 15 Fi = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 00</td> <td>Mix = 1 0000 0000 0000 12 Fi = 1 0000 0000 13 Fi = 1 0000 0000 14 Fi = 1 0000 0000 15 Fi = 1 0000 0000 16 Fi = 1 0000 0000 17 Fi = 1 0000 0000 100 Fi = 1 0000 0000 100 Fi = 10 0000 0000 101 Fi = 10 0000 0000 101<td>Tits = 1 0000 0000 0000 Tits = 1000 00000 0000 0000 Tits = 1000 00000 0000 0000 Tits = 1000 00000 00000 00000 Tits = 1000 00000 00000 00000 Tits = 1000000 00000 00000 00000 Tits = 10000000 00000 00000 00000 Tits = 10000000 00000 00000 00000</td><td>31 X4 = SIGHAX(I)</td><td></td><td>200</td><td></td><td></td></td>	13 File Sisterio Control Contro Control Control	The image is	Mix = 1 0000 0000 0000 12 Fi = 1 0000 0000 0000 13 Fi = 1 0000 0000 0000 14 Fi = 1 0000 0000 0000 12 Fi = 1 0000 0000 0000 14 Fi = 1 0000 0000 0000 15 Fi = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 0000 100 = 1 0000 0000 00	Mix = 1 0000 0000 0000 12 Fi = 1 0000 0000 13 Fi = 1 0000 0000 14 Fi = 1 0000 0000 15 Fi = 1 0000 0000 16 Fi = 1 0000 0000 17 Fi = 1 0000 0000 100 Fi = 1 0000 0000 100 Fi = 10 0000 0000 101 Fi = 10 0000 0000 101 <td>Tits = 1 0000 0000 0000 Tits = 1000 00000 0000 0000 Tits = 1000 00000 0000 0000 Tits = 1000 00000 00000 00000 Tits = 1000 00000 00000 00000 Tits = 1000000 00000 00000 00000 Tits = 10000000 00000 00000 00000 Tits = 10000000 00000 00000 00000</td> <td>31 X4 = SIGHAX(I)</td> <td></td> <td>200</td> <td></td> <td></td>	Tits = 1 0000 0000 0000 Tits = 1000 00000 0000 0000 Tits = 1000 00000 0000 0000 Tits = 1000 00000 00000 00000 Tits = 1000 00000 00000 00000 Tits = 1000000 00000 00000 00000 Tits = 10000000 00000 00000 00000 Tits = 10000000 00000 00000 00000	31 X4 = SIGHAX(I)		200		
X:M: 1 0.000 V.2 X:M: 1 0.001 V.2 X:M: 2 0.002 V.2	X:M: 1 0.000 0.000 0.000 X:M: 1 0.010 0.000 0.000 X:M: 1 0.000	X: 1 0.002 0.002 0.002 X: X: X: 0.00	21 First 0000 0000 21 First 0000 0000 21 First 0000 0000 210 1000 0000 0000 210 1000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000 0000 210 0000 0000	25 1 1 0000 425 26 1 1 0000 425 27 1 1 0000 425 26 1 0000 425 0000 27 1 0000 425 0000 28 1 0000 425 0000 28 1 0000 425 0000 28 25 1 0000 425 29 0000 8000 0000 425 2000 1 2 0000 425 2000 1 2 0000 425 2000 1 2 0000 425 2000 1 2 0000 425 2000 1 2 0000 425 2000 1 2 0000 425 2000 1 2 0000 425 2000 1 2 0000 425 2000 1 2 0000 425 <t< td=""><td>21 Conta = 1 Conta = 1 Conta = 1 22 F (apertr.). cfr. 1.10 C0 TO 445 Conta = 1 24 Conta = 1 Conta = 1 Conta = 1 25 Conta = 1 Conta = 1 Conta = 1 26 Conta = 1 Conta = 1 Conta = 1 26 Conta = 1 Conta = 1 Conta = 1 27 Conta = 1 Conta = 1 Conta = 1 28 Conta = 1 Conta = 1 Conta = 1 29 Conta = 1 Conta = 1 Conta = 1 29 Conta = 1 Conta = 1 Conta = 1 29 Conta = 1 Conta = 1 Conta = 1 20 Conta = 1 Conta = 1 Conta = 1 20 Conta = 1 Conta = 1 Conta = 1 21 Conta = 1 Conta = 1 Conta = 1 21 Conta = 1 Conta = 1 Conta = 1 21 Conta = 1 Conta = 1 Conta = 1 21 Conta = 1 Conta = 1 Conta = 1 22 Conta = 1 <tdconta 1<="" =="" td=""> Conta = 1 <t< td=""><td>IND4 = NSTEP(I)</td><td>CORE</td><td>24</td><td></td><td></td></t<></tdconta></td></t<>	21 Conta = 1 Conta = 1 Conta = 1 22 F (apertr.). cfr. 1.10 C0 TO 445 Conta = 1 24 Conta = 1 Conta = 1 Conta = 1 25 Conta = 1 Conta = 1 Conta = 1 26 Conta = 1 Conta = 1 Conta = 1 26 Conta = 1 Conta = 1 Conta = 1 27 Conta = 1 Conta = 1 Conta = 1 28 Conta = 1 Conta = 1 Conta = 1 29 Conta = 1 Conta = 1 Conta = 1 29 Conta = 1 Conta = 1 Conta = 1 29 Conta = 1 Conta = 1 Conta = 1 20 Conta = 1 Conta = 1 Conta = 1 20 Conta = 1 Conta = 1 Conta = 1 21 Conta = 1 Conta = 1 Conta = 1 21 Conta = 1 Conta = 1 Conta = 1 21 Conta = 1 Conta = 1 Conta = 1 21 Conta = 1 Conta = 1 Conta = 1 22 Conta = 1 <tdconta 1<="" =="" td=""> Conta = 1 <t< td=""><td>IND4 = NSTEP(I)</td><td>CORE</td><td>24</td><td></td><td></td></t<></tdconta>	IND4 = NSTEP(I)	CORE	24		
32 FT (ReVect) 0.000 2.5 33 FT (ReVect) 0.01 2.5 40 0.01 0.000 2.5 10 0.01 0.000 2.5 10 0.01 0.000 2.5 10 0.01 0.000 2.5 10 0.01 0.000 2.5 10 0.01 0.000 2.5 10 0.01 0.000 2.5 10 0.01 0.000 2.5 10.01 0.01 0.000 2.5 10.01 0.01 0.000 2.5 10.01 0.01 0.000 2.5 10.01 0.01 0.01 0.000 10.01 0.01 0.01 0.000 10.01 0.01 0.01 0.000 10.01 0.01 0.01 0.000 10.01 0.01 0.01 0.000 10.01 0.01 0.01 0.000 10.01 0.01 0.01 0.000 10.01 0.01	XI = 1 00000 00000 00000 XI = 1 00000	X: X	X: X	XII N = 0 CODE 27 XI N = 1 CODE 27	XIII a 1 0 000 25 XIII a 100 000 00	KMAX = 1	CORE 4	125		
27 (3.1 = 1, 1.0) 50 T0 40 2005 427 40 (27) = 1 2004 429 40 (27) = 1 2004 429 50 (10 = 1, 1.0) 50 T0 40 2005 429 50 (10 = 1, 1.0) 50 T0 40 2005 429 50 (10 = 1, 1.0) 50 T0 40 2005 429 50 (10 = 1, 1.0) 50 T0 40 2005 429 50 (10 = 1, 1.0) 50 T0 40 2005 429 50 (10 = 1, 1.0) 50 T0 40 2005 429 50 (10 = 1, 1.0) 50 T0 40 2005 429 50 (10 = 1, 1.0) 50 T0 40 2005 449 51 (11 = 1, 1.0) 50 T0 40 2005 449 51 (11 = 1, 1.0) 50 T0 40 2005 449 52 (12 (11 - 1, 1.0) 50 T0 40 2005 449 52 (12 (11 - 1, 1.0) 50 T0 40 2005 449 52 (12 (11 - 1, 1.0) 50 T0 40 2005 449 52 (11 (11 - 1, 1.0) 50 T0 40 2005 449 52 (11 (11 - 1, 1.0) 50 T0 40 2005 449 52 (11 (11 - 1, 1.0) 50 T0 40 2005 449 52 (11 (11 - 1, 1.0) 50 T0 40 2005 449 50 (10 (11 - 1, 1.0) 50 T0 40	32 F(31 = 1, 1, 0) 60 TO 40 200 201 40 F(1) = 1 000 201 201 50 TO 5 100 201 201 50 TO 100 201 201 201 50 TO 100 201 201 201 50 TO 100 201 201 201 51 TO 100	21 12 <td< td=""><td>2: F(3) <</td><td>21 17 18 16 <td< td=""><td>21 17 18.1 19.1 <td< td=""><td>KAIN = 0</td><td>CORE 4</td><td>•26</td><td></td><td></td></td<></td></td<></td></td<>	2: F(3) <	21 17 18 16 <td< td=""><td>21 17 18.1 19.1 <td< td=""><td>KAIN = 0</td><td>CORE 4</td><td>•26</td><td></td><td></td></td<></td></td<>	21 17 18.1 19.1 <td< td=""><td>KAIN = 0</td><td>CORE 4</td><td>•26</td><td></td><td></td></td<>	KAIN = 0	CORE 4	•26		
21 F (Revect)	22 15 (Revect)	21 F (RUNTCIT) .GT. 1.00 60 T0 40 0000 420 40 CT T CONC 420 50 TAL STERNIT) CONC 43 51 CAL T CONC 43 51 CAL T CONC 43 51 CAL T CONC 44 52 SEGHALI CONC CONC 44 53 SEGHALI CONC CONC 44 54 CONC CONC CONC 44 54 CONC CONC CONC 44 55 CONC CONC CONC 55 55 CONC CONC	32 IF (ANNYCII) .6F1 1.00 60 T0 40 CORE 228 10 CTU = 1 CORE 228 10 CTU = 1 CORE 228 110 CTU = 1 CORE 228 111 CTU = 1 CORE 228 111 CTU = 1 CORE 228 110 CTU = 1 CORE 228 111 CTU = 1 CORE 228 111 CTU = 1 CORE 238	23 IF (AMOPTICII). 667: 1.00 GO TO 40 CORE 428 10 COLO CORE 428 10 COLO CORE 428 11 COLO CORE 428 12 COLO CORE 428 13 COLO CORE 428 14 CORE CORE 428 14 CORE CORE 428 15 COLO CORE 428 16 COLO CORE 428 17 COLO CORE 428 16 COLO CORE 428 17 COLO CORE 428 18 COLO CORE 428 19 COLO CORE 428 10 CORE CORE 428 10 CORE CORE 428 10 CORE CORE 428 10 CORE CORE 448 10 CORE CORE 448 10 CORE CORE 448 <td>23 TF (Merricut)GT, 1.00 GO TO 40 CORE 4.2 10 CTO 10 CORE 4.2 10 CTO 10 CORE 4.2 10 CTO 10 CORE 4.2 11 CTO 10 CORE 4.2 12 CTO 10 CORE 4.2 13 SCORTICI). CORE 4.2 14 CORE CORE 4.2 15 CTO 10 CORE 4.2 16 CTO 10 CORE 4.2 17 CTO 10 CORE 4.2 18 CTO 10 CORE 4.2 19 CTO 10 CORE 4.2 100 CTO 10 CORE 4.2 101 CTO 10 CORE 4.2 102 CTO 10 CORE 4.2 103 CTO 10 CORE 4.2 104</td> <td>K31 = 1</td> <td>CORE</td> <td>+27</td> <td></td> <td></td>	23 TF (Merricut)GT, 1.00 GO TO 40 CORE 4.2 10 CTO 10 CORE 4.2 10 CTO 10 CORE 4.2 10 CTO 10 CORE 4.2 11 CTO 10 CORE 4.2 12 CTO 10 CORE 4.2 13 SCORTICI). CORE 4.2 14 CORE CORE 4.2 15 CTO 10 CORE 4.2 16 CTO 10 CORE 4.2 17 CTO 10 CORE 4.2 18 CTO 10 CORE 4.2 19 CTO 10 CORE 4.2 100 CTO 10 CORE 4.2 101 CTO 10 CORE 4.2 102 CTO 10 CORE 4.2 103 CTO 10 CORE 4.2 104	K31 = 1	CORE	+27		
40 CCV 10 * CCV 10 * CCV 10 * 35 CC 10 * CCV 10 * CCV 10 * 36 CC 10 * CCV 10 * CCV 10 * 100 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10 * CCV 10 * CCV 10 * 1010 CCV 10	up 0.0 0.0 0.0 0.0 35 0.0 0.0 0.0 0.0 35 0.0 0.0 0.0 0.0 35 0.0 0.0 0.0 0.0 36 0.0 0.0 0.0 0.0 0.0 10 0.00 0.0 0.0 0.0 10 0.00 0.0 0.0 0.0 10 0.0 0.0 0.0 0.0 10 0.0 0.0 0.0 0.0 10 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0	40 C0 C0 <td< td=""><td>40 COL CO</td><td>40 C0 C0 C0 C0 35 C0 C0 C0 C0 35 C0 C0 C0 C0 35 C0 C0 C0 C0 C0 36 C0 C0 C0 C0 C0 C0 57 C0 C0</td><td>40 00 000 000 55 000 000 000 55 000 000 000 55 000 000 000 56 000 000 000 57 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000<!--</td--><td>32 IF (RNCYC(I) .61. 1.0) GO TO 40</td><td>CORE</td><td>+28</td><td></td><td></td></td></td<>	40 COL CO	40 C0 C0 C0 C0 35 C0 C0 C0 C0 35 C0 C0 C0 C0 35 C0 C0 C0 C0 C0 36 C0 C0 C0 C0 C0 C0 57 C0	40 00 000 000 55 000 000 000 55 000 000 000 55 000 000 000 56 000 000 000 57 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 000 51 000 000 </td <td>32 IF (RNCYC(I) .61. 1.0) GO TO 40</td> <td>CORE</td> <td>+28</td> <td></td> <td></td>	32 IF (RNCYC(I) .61. 1.0) GO TO 40	CORE	+28		
35 0.0000 0.0000 0.0000 35 0.0000 0.0000 0.0000 100 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 0.0000 0.0000 0.0000 0.001 <td>35 0.00000 0.00000 0.0000 0.0000</td> <td>35 4.0 0.000 4.0 35 4.0 0.000 4.0 0.010 4.0 0.000 4.0 0.010 4.0 0.000 4.0 0.010 3.3 5.00000 4.0 0.010 3.3 5.00000 4.0 0.010 3.3 5.00000 4.0 0.010 3.3 5.00000 4.0 0.010 3.3 5.00000 4.0 0.010 3.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 <td< td=""><td>Vir.1 Vir.1 <td< td=""><td>VI KIN = 1 0.000 0.000 0.000 SIX = SIGHIN(1) 0.000 0.000 0.000 KHX = 1 0.000 0.000 0.000 Store SIGHIN(1) 0.000 0.000 0.000 Store SIGHIN(1)</td><td>0.010 0.010 0.010 0.010 15. 4. = 5.010 0.010 0.010 0.010 0.110 = 10 0.010 0.010 0.010 0.111 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 1001 = 0.010 0.010 0.010 0.010 1012 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010</td><td></td><td>CORE</td><td>524</td><td></td><td>•</td></td<></td></td<></td>	35 0.00000 0.00000 0.0000 0.0000	35 4.0 0.000 4.0 35 4.0 0.000 4.0 0.010 4.0 0.000 4.0 0.010 4.0 0.000 4.0 0.010 3.3 5.00000 4.0 0.010 3.3 5.00000 4.0 0.010 3.3 5.00000 4.0 0.010 3.3 5.00000 4.0 0.010 3.3 5.00000 4.0 0.010 3.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 1.0 0.0000 4.0 0.010 <td< td=""><td>Vir.1 Vir.1 <td< td=""><td>VI KIN = 1 0.000 0.000 0.000 SIX = SIGHIN(1) 0.000 0.000 0.000 KHX = 1 0.000 0.000 0.000 Store SIGHIN(1) 0.000 0.000 0.000 Store SIGHIN(1)</td><td>0.010 0.010 0.010 0.010 15. 4. = 5.010 0.010 0.010 0.010 0.110 = 10 0.010 0.010 0.010 0.111 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 1001 = 0.010 0.010 0.010 0.010 1012 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010</td><td></td><td>CORE</td><td>524</td><td></td><td>•</td></td<></td></td<>	Vir.1 Vir.1 <td< td=""><td>VI KIN = 1 0.000 0.000 0.000 SIX = SIGHIN(1) 0.000 0.000 0.000 KHX = 1 0.000 0.000 0.000 Store SIGHIN(1) 0.000 0.000 0.000 Store SIGHIN(1)</td><td>0.010 0.010 0.010 0.010 15. 4. = 5.010 0.010 0.010 0.010 0.110 = 10 0.010 0.010 0.010 0.111 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 1001 = 0.010 0.010 0.010 0.010 1012 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010</td><td></td><td>CORE</td><td>524</td><td></td><td>•</td></td<>	VI KIN = 1 0.000 0.000 0.000 SIX = SIGHIN(1) 0.000 0.000 0.000 KHX = 1 0.000 0.000 0.000 Store SIGHIN(1)	0.010 0.010 0.010 0.010 15. 4. = 5.010 0.010 0.010 0.010 0.110 = 10 0.010 0.010 0.010 0.111 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 0.112 = 10 0.010 0.010 0.010 1001 = 0.010 0.010 0.010 0.010 1012 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010 0.010 112 = 0.010 0.010 0.010		CORE	524		•
35 GG (10, 4.5) CORE	35 G0170-45 MIN G0070 G0170-45 MIN G0070 G0170 G0170 <td>35 0.010 + 15 0.002 0.025 0.025 100 = 30TEM(1) 0.012 0.025 0.025 0.025 100 = 30TEM(1) 0.025 0.025 0.025 0.025 101 = 30TEM(1) 0.025 0.025 0.025 0.025 101 = 32 0.012 0.025 0.025 0.025 0.025 101 = 32 0.010 + 110 0.010 + 110 0.025 0.025 0.025 0.025 101 = 12 0.010 + 110 0.010 + 110 0.010 + 110 0.025<</td> <td>35 GG * 100 - 100 GG * 100 GG * 100 GG * 100 1000 = NATEP(1) GG * 100 GG * 100 GG * 100 GG * 100 1001 = STG* GG * 100 1001 = STG* GG * 100 G</td> <td>35 G.G. TO, Z.S. CONC G.S. 100 = SKICHKI, CONC G.S. 100 = SKICHKI, CONC G.S. 101 = SK CONC G.S. 102 = SKICHKI, CONC G.S. 103 = SKICHKI, CONC G.S. 104 = SKICHKI, CONC G.S. 105 = SKICHKI, CONC G.S. 106 = SKICHKI, CONC G.S. 107 = SKICHKI, CONC G.S. 108 = SKICHKI, CONC G.S. 100 = CONL CONC G.S. 101 = SKICHKI, CONC G.S. 101 = SKICHKI, CONC G.S. 102 = SKICHKI, CONC G.S. 103 = SKICHKI, CONC G.S. 104 = SKIC, CONC G.S. 105 = SKICHKI, CONC G.S. 106 = SKICHKI, CONC G.S. 107 = SKICHKI, CONC G.S. 108 = SKICHKI, CONC G.S. 108 = SKICHKI, CONC G.S. 108 = SKICHKI, CONC G.S</td> <td>35 60.700 - 4.50 00000 00000 00000 00000 100 + m SIGP(H1) 00000 00000 00000 00000 00000 100 + m SIGP(H1) 00000 00000 00000 00000 00000 00000 100 + m SIGP(H1) 000000 00000<td></td><td></td><td></td><td></td><td></td></td>	35 0.010 + 15 0.002 0.025 0.025 100 = 30TEM(1) 0.012 0.025 0.025 0.025 100 = 30TEM(1) 0.025 0.025 0.025 0.025 101 = 30TEM(1) 0.025 0.025 0.025 0.025 101 = 32 0.012 0.025 0.025 0.025 0.025 101 = 32 0.010 + 110 0.010 + 110 0.025 0.025 0.025 0.025 101 = 12 0.010 + 110 0.010 + 110 0.010 + 110 0.025<	35 GG * 100 - 100 GG * 100 GG * 100 GG * 100 1000 = NATEP(1) GG * 100 GG * 100 GG * 100 GG * 100 1001 = STG* GG * 100 1001 = STG* GG * 100 G	35 G.G. TO, Z.S. CONC G.S. 100 = SKICHKI, CONC G.S. 100 = SKICHKI, CONC G.S. 101 = SK CONC G.S. 102 = SKICHKI, CONC G.S. 103 = SKICHKI, CONC G.S. 104 = SKICHKI, CONC G.S. 105 = SKICHKI, CONC G.S. 106 = SKICHKI, CONC G.S. 107 = SKICHKI, CONC G.S. 108 = SKICHKI, CONC G.S. 100 = CONL CONC G.S. 101 = SKICHKI, CONC G.S. 101 = SKICHKI, CONC G.S. 102 = SKICHKI, CONC G.S. 103 = SKICHKI, CONC G.S. 104 = SKIC, CONC G.S. 105 = SKICHKI, CONC G.S. 106 = SKICHKI, CONC G.S. 107 = SKICHKI, CONC G.S. 108 = SKICHKI, CONC G.S. 108 = SKICHKI, CONC G.S. 108 = SKICHKI, CONC G.S	35 60.700 - 4.50 00000 00000 00000 00000 100 + m SIGP(H1) 00000 00000 00000 00000 00000 100 + m SIGP(H1) 00000 00000 00000 00000 00000 00000 100 + m SIGP(H1) 000000 00000 <td></td> <td></td> <td></td> <td></td> <td></td>					
35 Xu = STERWI(1) 0002 0002 0002 FUNA = 1 0002 0002<	35 Signitu(1) COURT SIGNITUD RUNA = NSTER(1) COURT SIGNITUD RUNA = STORNICID COURT SIGNITUD RUNA = STORNICID COURT SIGNITUD RUNA = STORNICID COURT SIGNITUD RUNA = NOULD	35 % 5 \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$	35 Strictur(1) 0002 Strictur(1) 1004 Strictur(1) 0002 Strictur(1) 2011 20 Strictur(1) 0002 Strictur(1) 2011 20 Strictur(1) 0002 Strictur(1) 2011 20 Strictur(1) 0002 Strictur(1) 2013 Strictur(1) 0002 Strictur(1) 0002 2014 Strictur(1) 0002 Strictur(1) 0002 2014 Strictur(1) 0002 Strictur(1) 0002 2015 Strictur(1) 0002 Strictur(1) 0002 Strictur(1) 2016 Strictur(1) Strictur(1) 0002 Strictur(1) 0002 Strictur(1) 2016 Strictur(1)	35 Struct (1) CORE Struct (1) 1004 WITER(1) CORE Struct (1) 1014 STRUCT (1) CORE Struct (1) 1015 STRUCT (1) CORE Struct (1) 1016 Struct (1) CORE Struct (1) 1017 Struct (1) CORE Struct (1) 1016 Struct (1) CORE Struct (1) 1017 Struct (1) CORE Struct (1) 1018 Struct (1) CORE Struct (1) 1018 Struct (1) CORE Struct (1) 1018 Struct (1) Struct (1) CORE 1018 Struct (1) Struct (1) Struct (1) 1018	35 Struct Structurt) CORR Structurt) 1004 Structurt) CORR Structurt) 1004 Structurt) CORR Structurt) 1004 Structurt) CORR Structurt) 1005	ATMU - 8 GO TO 445	1005 1005 1005	101		
1000 1000 1000 1000 1011 1 0000 1000 1011 1 0000 1000 1011 1 0000 1000 1011 1 0000 1000 1011 1 0000 1000 1011 1 0000 1000 1011 1 0000 1000 1011 1 0000 1000 1011 1 0000 1000 1011 1 0000 1000 1011 1 0000 1000 1011 1 1000 1000 1000 1011 1 1000 1000 1000 1011 1 1000 1000 1000 1011 1 1000 1000 1000 1011 1 1000 1000 1000 1011 1 1000 1000 1000 1011 1 1000 1000 1000 1011 1 1000	10000 10000 10000 10000 10100 10000 10000 10000 10000 10100 10000 10000 10000 10000 10000 10100 10000 10000 10000 10000 10000 10000 10100 100000 10000 10000<	10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 100000 100000 100000 1000	Visual Structure 00000 00000 00000 Visual Structure 000000 00000 000000 Visual Structure 000000 00000 000000 Visual Structure 000000 00000 00000 Visual Structure 000000 000000 000000 Visual Structure 0000000 000000 000000 Visual Structure 00000000 0000000 000000 Visual Structure 000000000000000000 000000000000000000000000000000000000	7000 4500 450 7000 450 600 6414 1 600	0.0000 0.0000 0.0000 0.0000 0.0010 0.0000 0.0000 0.0000 0.0110 0.0110 0.0000 0.0000 0.0110 0.0110 0.0000 0.0000 0.0110 0.0110 0.0110 0.0100 0.0110 0.0110 0.0110 0.0000 0.0000 0.0110 0.0110 0.0110 0.0000 0.0000 0.0110 0.0110 0.0110 0.0000 0.0000 0.0110 0.0110 0.0110 0.0000 0.0000 0.0110 0.0110 0.0110 0.0000 0.0000 0.0110 0.0110 0.0110 0.0110 0.0000 0.0110 0.0110 0.0110 0.0110 0.0000 0.0110 0.0110 0.0110 0.0000 0.0000 0.0000 0.0110 0.0110 0.0110 0.0110 0.0110 0.0000 0.0000 0.0110 0.0110 0.0110 0.0110 0.0000 0.0000 0.0000 0.0110 0.0110 0.0110 0.0110			3		1
Market 1 Construction Construction Construction Market 1 Market 1 Co	Market 1 Market 1 Market 1 State	35 0.012 0.002 0.002 36 0.01 0.002 0.002 0.01 0.01 0.002 0.002 0.01 0.01 0.002 0.002 0.01 0.01 0.002 0.002 0.01 0.01 0.002 0.002 0.01 0.01 0.002 0.002 0.01 0.01 0.002 0.002 0.01 0.01 0.002 0.002 0.01 0.01 0.002 0.002 0.01 0.01 0.002 0.002 0.01 0.01 0.01 0.002 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01	15 51 5000 51 16 51 5000 51 170 51 5000 51 180 51 5000 51 181 1000 51 5000 51 180 51 5000 51 5000 51 180 510 510 510 5000 51 181 510 510 5000 51 5000 51 181 510 510 5000 51 5000 51 51 181 510 510 510 5000 51 <t< td=""><td>35 57 5000 57 36 57 5000 57 37 501 5000 57 1001 501 5000 57 1001 501 50 5000 57 1001 501 10 5000 57 1001 5100 5000 57 5000 57 1001 5100 5100 5000 57 5000 57 101 510 5100 5000 510 5000 510 111 5100 5100 5000 510 5000 510 101 510 5100 5000 510 510 5000 510 101 510 510 5000 5000 510 5000 510 101 510 510 510 5000 510 5000 510 101 510 510 5000 5000 510 5000 510 101 510 510 5000 5000 5000 <t< td=""><td>Market 1 0000 0000 0000 State 1 0000 0000 0000</td><td></td><td></td><td></td><td></td><td></td></t<></td></t<>	35 57 5000 57 36 57 5000 57 37 501 5000 57 1001 501 5000 57 1001 501 50 5000 57 1001 501 10 5000 57 1001 5100 5000 57 5000 57 1001 5100 5100 5000 57 5000 57 101 510 5100 5000 510 5000 510 111 5100 5100 5000 510 5000 510 101 510 5100 5000 510 510 5000 510 101 510 510 5000 5000 510 5000 510 101 510 510 510 5000 510 5000 510 101 510 510 5000 5000 510 5000 510 101 510 510 5000 5000 5000 <t< td=""><td>Market 1 0000 0000 0000 State 1 0000 0000 0000</td><td></td><td></td><td></td><td></td><td></td></t<>	Market 1 0000 0000 0000 State 1 0000 0000 0000					
35 VOLUNA 0 VOLUNA 0 35 VOLUNA 0 VOLUNA 0 VOLUNA 0 36 VOLUNA 0 VOLUNA 0 VOLUNA	35 05 0.000	Mark = 1 0.000 4.37 Mark = 1 0.000 4.47 Mark = 1 0.0000 4.47 <t< td=""><td>35 0.01 0.00 0.00 36 0.01 0.00 0.00 100 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 <t< td=""><td>36 001 000 000 000 000 36 001 001 000 000 000 000 100 001 10 000 000 000 000 000 100 001 10 001 000<!--</td--><td>36 73 5 73 5 74 74 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100</td><td></td><td></td><td>* L</td><td></td><td></td></td></t<></td></t<>	35 0.01 0.00 0.00 36 0.01 0.00 0.00 100 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.00 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 0.00 101 0.01 0.01 <t< td=""><td>36 001 000 000 000 000 36 001 001 000 000 000 000 100 001 10 000 000 000 000 000 100 001 10 001 000<!--</td--><td>36 73 5 73 5 74 74 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100</td><td></td><td></td><td>* L</td><td></td><td></td></td></t<>	36 001 000 000 000 000 36 001 001 000 000 000 000 100 001 10 000 000 000 000 000 100 001 10 001 000 </td <td>36 73 5 73 5 74 74 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100</td> <td></td> <td></td> <td>* L</td> <td></td> <td></td>	36 73 5 73 5 74 74 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100			* L		
35 513 500 500 513 100 513 515 500 513 1100 515 500 514 515 1100 514 515 500 514 1100 514 515 500 515 1100 517 500 516 516 111 516 516 516 516 111 516 516 516 516 111 516 516 516 516 111 516 516 516 516 111 516 516 516 516 111 516 516 516 516 111 516 516 516 516 111 516 516 516 516 111 516 516 516 516 111 516 516 516 516 111 516 516 516 516 111 516 516 516 <t< td=""><td>35 511 - 10 0000 110 1010 - 11 - 10 0010 110 110 1010 - 11 - 10 0010 110 110 1010 - 11 - 10 0010 110 110 110 1010 - 11 - 10 0010 110 110 110 110 1010 - 11 - 10 110 110 110 110 110 110 1111 - 10 110 110 110 110 110 110 110 1111 - 10 1</td><td>38 Virtual = 0 Virtual = 0 Virtual = 0 Virtual = 0 131 131 131 Virtual = 1 Virtual = 1 Virtual = 1 1411 141 141 Virtual = 1 Virtual = 1 Virtual = 1 Virtual = 1 141 141 141 141 Virtual = 1 Virtual = 1</td><td>35 51 - 51 - 51 - 51 - 51 - 51 - 51 - 51 -</td><td>35 51 57 50 51 10 67 0 0 0 0 11 10 0 0 0 0 0 12 5 5 0</td><td>35 51 <td< td=""><td></td><td></td><td></td><td></td><td></td></td<></td></t<>	35 511 - 10 0000 110 1010 - 11 - 10 0010 110 110 1010 - 11 - 10 0010 110 110 1010 - 11 - 10 0010 110 110 110 1010 - 11 - 10 0010 110 110 110 110 1010 - 11 - 10 110 110 110 110 110 110 1111 - 10 110 110 110 110 110 110 110 1111 - 10 1	38 Virtual = 0 Virtual = 0 Virtual = 0 Virtual = 0 131 131 131 Virtual = 1 Virtual = 1 Virtual = 1 1411 141 141 Virtual = 1 Virtual = 1 Virtual = 1 Virtual = 1 141 141 141 141 Virtual = 1	35 51 - 51 - 51 - 51 - 51 - 51 - 51 - 51 -	35 51 57 50 51 10 67 0 0 0 0 11 10 0 0 0 0 0 12 5 5 0	35 51 <td< td=""><td></td><td></td><td></td><td></td><td></td></td<>					
30 73.1 = 10 0000 4.37 70.10 100 100 100 4.35 1100 100 100 100 4.35 1100 100 100 100 4.35 1100 100 100 100 4.45 1100 100 100 100 100 4.45 1100 11 100 100 100 100 100 1100 11 100 100 100 100 100 100 1100 110 100 1	35 57.11 = 9 5000 443 1001 51.564MU11 5000 443 1010 51.564MU11 5000 443 1011 51.564MU11 5000 443 1011 51.564MU11 5000 443 1011 51.564MU11 5000 443 1011 51.64MU11 5000 443 111 1011 5000 443 111 1011 5000 5000 111 1012 5104MU11 5000 111 1012 5104MU11 5000 111 1012 5104MU11 5000 111 1012 5104MU11 5000 111 1012 5000 5000 111 1012 5000 5000 111 1012 5000 5000 111 1012 5000 5000 111 1012 5000 5000 1111 1014 5000 5000 1111 1014 5000 5000	38 50.11 - 2 0.041 - 2 0.042 0.041 100 50.10 - 1 0.044 0.044 0.044 100 50.10 - 1 0.044 0.044 0.044 101 50.10 - 1 0.044 0.044 0.044 101 50.10 - 1 0.044 0.044 0.044 101 50.10 - 1 0.044 0.044 0.044 101 50.10 - 1 0.014 0.044 0.044 101 50.10 - 10 0.014 0.044 0.044 101 50.10 - 10 0.014 0.014 0.014 101 101 101 0.014 0.014 0.014 101 101 101 0.014 0.014 0.014 101 101 101 0.014 0.014 0.014 1010 1011 101 0.014 0.014 0.014 1010 1011 101 0.014 0.014 0.014 1010 1011 101 0.014 0.014 0.014 1010 1011	38 50.11 = 10 0.016 4.15 1703 = NSTEGHN(1) 100 = 12 0.016 4.15 1703 = NSTEGHN(1) 0.016 4.15 0.016 4.15 1703 = NSTEGHN(1) 0.016 4.15 0.016 4.15 1703 = NSTEGHN(1) 0.016 4.10 0.016 4.15 1701 = STGANN(1) 0.016 4.10 0.016 4.15 1701 = STGANN(1) 0.017 0.016 4.15 1701 = STGANN(1) 0.017 0.016 4.15 1702 = STGANN(1) 0.017 0.016 4.15 1703 = TOU 1001 0.016 4.15 0.016 1703 = TOU 1001 0.016 4.15 0.016 1703 = TOU 1001 0.016 4.15 0.016 1703 = TOU 1001 0.016 4.15 0.016 4.15 1703 = TOU 1001 0.016 4.10 0.016 4.15 1703 = TOU 1001 1.10 0.016 4.15 0.016 1703 = TOU 1001 1.10 0.016 4.15	35 55 <td< td=""><td>35 55 <td< td=""><td></td><td></td><td>0</td><td></td><td></td></td<></td></td<>	35 55 <td< td=""><td></td><td></td><td>0</td><td></td><td></td></td<>			0		
35 33 150.42 0045 440 100.35 NSTEP[1] 0045 440 100.35 NSTEP[1] 0045 441 100.35 NSTEP[1] 0045 441 100.35 NSTEP[1] 0045 441 100.35 NSTEP[1] 0045 441 111 Ff (Mat. 0) 60 To 4.0 0045 445 112 STEMALII 0046 445 445 113 STEMALII 0046 445 445 1101 NSTEPLI 0046 445 445 1101 STEMALII 0047 445 445 1101 STEMALII 0047 445 445 1101 STEMALII 0046 445 445 1101 STEMALII 0047 445 445 1101 STEMALII <td< td=""><td>35 37.3 15.0.42 0.000 440 101.0.2 101.0.2 0.001 440 101.0.2 101.0.2 0.001 440 101.0.2 0.010.1 0.001 441 101.0.2 0.010.1 0.001 441 101.0.2 0.010.1 0.001 441 101.0.2 0.010.1 0.001 441 101.0.1 0.010.1 0.001 0.001 441 101.0.1 0.001 0.001 0.001 441 101.0.1 0.002 0.001 0.002 441 101.0.1 0.002 0.001 0.002 441 100.1 0.002 0.001 0.002 441 100.1 0.002 0.001 0.002 441 100.1 0.002 0.001 0.002 441 100.1 0.002 0.001 0.002 441 100.1 0.002 0.001 0.002 441 100.1 0.001 0.002 0.002 441 100.1 0.001</td><td>36 xxx = 1 0.000</td><td>36 10.10 10.10 10.00 10</td><td>35 10 0.000</td><td>36 xxx = 1 0.000 xxx = 1 xxx = 1 0.000 xxx = 1 0.000 xxx = 1 xxx = 1 0.000 xxx = 1 0.000 xxx = 1 xxx = 1 0.000 xxx = 1 0.000 xxx = 1 xxx = 1 0.000 xxx = 1 0.000 xxx = 1 xxx = 1 0.000 xxx = 1 0.000 xxx = 1 xxx = 1563MX(1) 0.000 xxx = 1563MX(1) 0.000 xxx = 1 xxx = 1563MX(1) 0.000 xxx = 1563MX(1) 0.000 xxx = 1 xxx = 1563MX(1) 0.000 xxx = 1563MX(1) 0.000 xxx = 1563MX(1) xxx = 1563MX(1) 0.000 xxx = 1563MX(1) 0.000 xxx = 1563MX(1) xxx = 1001 1001 1001 0.000 xxx = 1563MX(1) 0.000 xxx = 1001 1001 1001 1001 0.000 xxx = 1563MX(1) 0.000 xxx = 1001 1001 1001 1001 0.000 0.000 1001 xxx = 1001 1001 1001 1001 0.000 1001 1000</td><td></td><td>CORE 4</td><td>+37</td><td></td><td></td></td<>	35 37.3 15.0.42 0.000 440 101.0.2 101.0.2 0.001 440 101.0.2 101.0.2 0.001 440 101.0.2 0.010.1 0.001 441 101.0.2 0.010.1 0.001 441 101.0.2 0.010.1 0.001 441 101.0.2 0.010.1 0.001 441 101.0.1 0.010.1 0.001 0.001 441 101.0.1 0.001 0.001 0.001 441 101.0.1 0.002 0.001 0.002 441 101.0.1 0.002 0.001 0.002 441 100.1 0.002 0.001 0.002 441 100.1 0.002 0.001 0.002 441 100.1 0.002 0.001 0.002 441 100.1 0.002 0.001 0.002 441 100.1 0.002 0.001 0.002 441 100.1 0.001 0.002 0.002 441 100.1 0.001	36 xxx = 1 0.000	36 10.10 10.10 10.00 10	35 10 0.000	36 xxx = 1 0.000 xxx = 1 xxx = 1 0.000 xxx = 1 0.000 xxx = 1 xxx = 1 0.000 xxx = 1 0.000 xxx = 1 xxx = 1 0.000 xxx = 1 0.000 xxx = 1 xxx = 1 0.000 xxx = 1 0.000 xxx = 1 xxx = 1 0.000 xxx = 1 0.000 xxx = 1 xxx = 1563MX(1) 0.000 xxx = 1563MX(1) 0.000 xxx = 1 xxx = 1563MX(1) 0.000 xxx = 1563MX(1) 0.000 xxx = 1 xxx = 1563MX(1) 0.000 xxx = 1563MX(1) 0.000 xxx = 1563MX(1) xxx = 1563MX(1) 0.000 xxx = 1563MX(1) 0.000 xxx = 1563MX(1) xxx = 1001 1001 1001 0.000 xxx = 1563MX(1) 0.000 xxx = 1001 1001 1001 1001 0.000 xxx = 1563MX(1) 0.000 xxx = 1001 1001 1001 1001 0.000 0.000 1001 xxx = 1001 1001 1001 1001 0.000 1001 1000		CORE 4	+37		
50.35 S.S.S.MATERILI) 004E 439 1003 S.MSTEPILI) 004E 449 1010 S.O.TO 1 007E 441 1010 S.O.TO 1 007E 441 1010 S.O.TO 1 007E 441 1011 S.TERINICI 007E 441 007E 1011 S.TEGNICI 007E 441 007E 441 1012 S.TEGNICI 007E 441 007E 441 1012 S.TEGNICI 007E 441 007E 441 1012 S.TEGNICI 007E 442 007E 441 1012 S.TEGNICI 007E 442 007E 441 1012 S.TEGNICI 101 007E 441 441 1012 S.TEGNICI 100	50:35 = SIGREN(1) 000000000000000000000000000000000000	5.3.5.5.5.5.5.10000000000000000000000000	0.0.35 = \$3554MML10 0.0045 449 1000 & C1 = 1 0.0010 = 12 0.0018 449 1010 & C1 = 1 0.010 = 12 0.0018 449 1010 & C1 = 1 0.010 = 12 0.0018 449 1010 & C1 = 1 0.010 = 10 0.0018 449 111 = 35564ML10 0.0018 449 0.0018 449 111 = 111 11001 0.0018 449 0.0018 449 111 = 111 11001 0.010 + 110 0.0018 449 449 111 = 111 11001 0.010 + 110 0.0018 449 449 449 111 = 111 1101 1101 0.018 449 449 449 111 = 111 1101 1101 0.018 449 449 449 111 = 111 1101 1101 1101 0.018 449 449	0.0 32 = SIJGHN(1) 0.0 0000000000000000000000000000000000	0.0.123 = YSTGMUL1) 0.0.012 = V.11 1.003 = WSTFUL1) 0.001 = V.11 1.001 = V.1 = V.11 0.001 = V.11 1.001 = V.1 = V.11 0.001 = V.11 1.011 = V.1 = V.11 0.001 = V.11 1.012 = V.11 0.001 = V.11 1.013 = V.11 0.001 = V.11 1.014 = V.11 0.001 = V.11 1.015 = V.110 = V.11 0.001 = V.11 1.014 = V.11 0.001 = V.11 1.015 = V.110 = V.11 0.001 = V.11			+ 5 8		
TION = NSTEPT(1) CORE 440 VRIN = 1 CORE 441 VRIN = 1 CORE 442 KIN = 1 CORE 444 KI = SIGHM(1) CORE 444 NOI = NSTP(1) CORE 444 NOI = SIGHM(1) CORE<	TIO3 = NSTEPT(1) CORE 440 VEIN = 1 CORE 441 VEIN = 1 CORE	trug = NSTEP(1) CORE tut trug = NSTEP(1) CORE tut trug = NSTEP(1) CORE tut trug = STEP(1) CORE tut <tr< td=""><td>trug = 1/10 = N15FP(1) CORE tut trug = 1/1 CORE tut <t< td=""><td>THO3 = NISFEP(1) CORE 440 VEIN = 1 CORE 441 VEIN = 1 CORE 441 VEIN = 1 CORE 441 F (K8 - MC, D) GO TO 410 CORE 441 X = SIGHN(1) CORE 442 X = SIGHN(1) CORE 444 X = SIGHN(1) CORE 445 X = SIGHN(1) CORE 445 X = SIGHN(1) CORE 445 X = SIGHN(1) CORE 452 X = SIGHN(1) CORE 452 X = SIGHN(1) CORE 455 X = SIGHN(1) CORE 455 X = SIGHN(1) CORE 456 X = SIGHN(1) CORE 456 X = SIGHN(1) CORE 456 X = SIGHN(1) CORE<</td><td>THO3 HATFP(1) CONE UN CHAN = 1 CONE UN CONE UN CONE CAN = 1 CONE UN CAN = 515GHAN(1) CONE UN CONE UN CAN = 515GHAN(1) CONE UN CONE UN CAN = 515GHAN(1) CONE UN CONE UN CAN = 15GHAN(1) CONE UN CONE UN CAN = 1001 CONE UN <t< td=""><td>36 X3 = SIGMIN(I)</td><td>CORE 4</td><td>•39</td><td></td><td></td></t<></td></t<></td></tr<>	trug = 1/10 = N15FP(1) CORE tut trug = 1/1 CORE tut <t< td=""><td>THO3 = NISFEP(1) CORE 440 VEIN = 1 CORE 441 VEIN = 1 CORE 441 VEIN = 1 CORE 441 F (K8 - MC, D) GO TO 410 CORE 441 X = SIGHN(1) CORE 442 X = SIGHN(1) CORE 444 X = SIGHN(1) CORE 445 X = SIGHN(1) CORE 445 X = SIGHN(1) CORE 445 X = SIGHN(1) CORE 452 X = SIGHN(1) CORE 452 X = SIGHN(1) CORE 455 X = SIGHN(1) CORE 455 X = SIGHN(1) CORE 456 X = SIGHN(1) CORE 456 X = SIGHN(1) CORE 456 X = SIGHN(1) CORE<</td><td>THO3 HATFP(1) CONE UN CHAN = 1 CONE UN CONE UN CONE CAN = 1 CONE UN CAN = 515GHAN(1) CONE UN CONE UN CAN = 515GHAN(1) CONE UN CONE UN CAN = 515GHAN(1) CONE UN CONE UN CAN = 15GHAN(1) CONE UN CONE UN CAN = 1001 CONE UN <t< td=""><td>36 X3 = SIGMIN(I)</td><td>CORE 4</td><td>•39</td><td></td><td></td></t<></td></t<>	THO3 = NISFEP(1) CORE 440 VEIN = 1 CORE 441 VEIN = 1 CORE 441 VEIN = 1 CORE 441 F (K8 - MC, D) GO TO 410 CORE 441 X = SIGHN(1) CORE 442 X = SIGHN(1) CORE 444 X = SIGHN(1) CORE 445 X = SIGHN(1) CORE 445 X = SIGHN(1) CORE 445 X = SIGHN(1) CORE 452 X = SIGHN(1) CORE 452 X = SIGHN(1) CORE 455 X = SIGHN(1) CORE 455 X = SIGHN(1) CORE 456 X = SIGHN(1) CORE 456 X = SIGHN(1) CORE 456 X = SIGHN(1) CORE<	THO3 HATFP(1) CONE UN CHAN = 1 CONE UN CONE UN CONE CAN = 1 CONE UN CAN = 515GHAN(1) CONE UN CONE UN CAN = 515GHAN(1) CONE UN CONE UN CAN = 515GHAN(1) CONE UN CONE UN CAN = 15GHAN(1) CONE UN CONE UN CAN = 1001 CONE UN <t< td=""><td>36 X3 = SIGMIN(I)</td><td>CORE 4</td><td>•39</td><td></td><td></td></t<>	36 X3 = SIGMIN(I)	CORE 4	•39		
KMX = 1 CORE 441 KMX = 1 CORE 441 KMX = 1 CORE 441 F (KB . ML. 0) GO TO 440 CORE 442 F (KB . ML. 0) GO TO 440 CORE 444 F (KB . ML. 0) GO TO 440 CORE 444 X = SIGMA(I) CORE 444 X = SIGMA(I) <td>KMX = 1 CORE 441 KMX = 1 CORE 441 KMX = 1 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 411 CORE 441 KMX = 0 CORE 441 KMX = 0 CORE 441 KMX = 0 CORE 442 KMX = 0 CORE 443 KMX = 0 CORE 443 KMX = 0 CORE 444 KMX = 0 CORE CORE</td> <td>KMX = 1 0056 ***1 VEMX = 1 00 TO 12 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 401 00 TO 401 KY = 1 00 TO 402 00 TO 401 KY = 1 00 TO 402 00 TO 402 KY = 1 00 TO 402 00 TO 402 KY = 1 00 TO 402 00 TO 402 KY = 2 00 TO 402 00 TO 402 KY = 1 00 TO 402 00 TO 402 KY = 1 00 TO 402 00 TO 402 KY = 2 00 TO 402 00 TO 402 KY = 2 00 TO 402 00 TO 402 KY = 2 <td< td=""><td>410 K:MX = 1 CORE 441 410 K:MX = 1 CORE 442 517 K:MX = 1 CORE 443 517 K:MX = 1 CORE 443 517 K:MX = 1 CORE 444 512 SIGNIAL(1) CORE 444 513 SIGNIAL(1) CORE 444 514 SIGNIAL(1) CORE 445 5101 SIGNIAL(1) CORE 445 5102 SIGNIAL(1) CORE 455 5103 SIGNIAL(1) CORE 455 511 SIGNIAL(1) CORE 455 512 SIGNIAL(1) SIGNIAL(1) CORE 513 SIGNIAL(1) SIGNIAL(1) SIGNIAL(1) 513 SIGNIAL(1)</td><td>400 6:4:1:0 0:0:1:0:1 0:0:1:0:1 1:0:0:1:0:1:0:1:0:1:0:1:0:10 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:0:1:0:1:0:1:0:1:0:10 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:0:1:0:1:0:1:0:1:0:10 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:1:1:0:1:0:1:0:1:0:1 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:1:1:0:1:1:0:1:0:1 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:1:1:0:1:1:0:1:0:1:0:1 0:0:1:0:1:0:1 0:0:1:0:1:0:1 0:0:1:0:1:0:1 1:1:0:1:1:0:1:1:0:1:0:1:0:1:0:1:0:1:0:1</td><td>410 Correst Correst</td><td>IND3 = NSTEP(I)</td><td>CORE 4</td><td>• t 0</td><td></td><td></td></td<></td>	KMX = 1 CORE 441 KMX = 1 CORE 441 KMX = 1 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 410 CORE 441 KT (R - NL 0) GO TO 411 CORE 441 KMX = 0 CORE 441 KMX = 0 CORE 441 KMX = 0 CORE 442 KMX = 0 CORE 443 KMX = 0 CORE 443 KMX = 0 CORE 444 KMX = 0 CORE CORE	KMX = 1 0056 ***1 VEMX = 1 00 TO 12 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 410 00 TO 410 KY = 1 00 TO 401 00 TO 401 KY = 1 00 TO 402 00 TO 401 KY = 1 00 TO 402 00 TO 402 KY = 1 00 TO 402 00 TO 402 KY = 1 00 TO 402 00 TO 402 KY = 2 00 TO 402 00 TO 402 KY = 1 00 TO 402 00 TO 402 KY = 1 00 TO 402 00 TO 402 KY = 2 00 TO 402 00 TO 402 KY = 2 00 TO 402 00 TO 402 KY = 2 <td< td=""><td>410 K:MX = 1 CORE 441 410 K:MX = 1 CORE 442 517 K:MX = 1 CORE 443 517 K:MX = 1 CORE 443 517 K:MX = 1 CORE 444 512 SIGNIAL(1) CORE 444 513 SIGNIAL(1) CORE 444 514 SIGNIAL(1) CORE 445 5101 SIGNIAL(1) CORE 445 5102 SIGNIAL(1) CORE 455 5103 SIGNIAL(1) CORE 455 511 SIGNIAL(1) CORE 455 512 SIGNIAL(1) SIGNIAL(1) CORE 513 SIGNIAL(1) SIGNIAL(1) SIGNIAL(1) 513 SIGNIAL(1)</td><td>400 6:4:1:0 0:0:1:0:1 0:0:1:0:1 1:0:0:1:0:1:0:1:0:1:0:1:0:10 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:0:1:0:1:0:1:0:1:0:10 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:0:1:0:1:0:1:0:1:0:10 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:1:1:0:1:0:1:0:1:0:1 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:1:1:0:1:1:0:1:0:1 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:1:1:0:1:1:0:1:0:1:0:1 0:0:1:0:1:0:1 0:0:1:0:1:0:1 0:0:1:0:1:0:1 1:1:0:1:1:0:1:1:0:1:0:1:0:1:0:1:0:1:0:1</td><td>410 Correst Correst</td><td>IND3 = NSTEP(I)</td><td>CORE 4</td><td>• t 0</td><td></td><td></td></td<>	410 K:MX = 1 CORE 441 410 K:MX = 1 CORE 442 517 K:MX = 1 CORE 443 517 K:MX = 1 CORE 443 517 K:MX = 1 CORE 444 512 SIGNIAL(1) CORE 444 513 SIGNIAL(1) CORE 444 514 SIGNIAL(1) CORE 445 5101 SIGNIAL(1) CORE 445 5102 SIGNIAL(1) CORE 455 5103 SIGNIAL(1) CORE 455 511 SIGNIAL(1) CORE 455 512 SIGNIAL(1) SIGNIAL(1) CORE 513 SIGNIAL(1) SIGNIAL(1) SIGNIAL(1) 513 SIGNIAL(1)	400 6:4:1:0 0:0:1:0:1 0:0:1:0:1 1:0:0:1:0:1:0:1:0:1:0:1:0:10 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:0:1:0:1:0:1:0:1:0:10 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:0:1:0:1:0:1:0:1:0:10 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:1:1:0:1:0:1:0:1:0:1 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:1:1:0:1:1:0:1:0:1 0:0:1:0:1 0:0:1:0:1 0:0:1:0:1 1:1:0:1:1:0:1:1:0:1:0:1:0:1 0:0:1:0:1:0:1 0:0:1:0:1:0:1 0:0:1:0:1:0:1 1:1:0:1:1:0:1:1:0:1:0:1:0:1:0:1:0:1:0:1	410 Correst Correst	IND3 = NSTEP(I)	CORE 4	• t 0		
40 G0 TO 12 CORE 442 15 K0 ME. 0) G0 TO 12 CORE 443 15 K0 ME. 0) G0 TO 12 CORE 444 15 K0 ME. 10) G0 TO 12 CORE 444 15 SIGHNU(1) CORE 444 16 K1 SIGHNU(1) CORE 444 17 K0 MD = NOL CORE 444 1001 SIGHNU(1) CORE 444 1001 SIGHNU(1) CORE 444 1001 SIGHNU(1) CORE 444 1002 NOL CORE 444 1001 SIGHNU(1) CORE 444 1001 SIGHNU(1) CORE 444 1002 NOL CORE 444 1002 NOL CORE 444 1001 SIGHNU(1) CORE 444 1002 NOL CORE 444 1002 NOL CORE 444 1001 SIGNU(1) LO CORE 1011 SIGNU(1	400 GG TO 12 CORE 442 71 F(K8, M, D) GO TO 410 CORE 444 71 STGCHAR(1) CORE 444 71 STGCHAR(1) CORE 444 73 STGCHAR(1) CORE 444 741 MOTOTOTOTOTOTOTOTOTOTOTOTOTOTOTOTOTOTOT	400 GG TO 12 COCE 442 411 SIGNAK(1) GO TO 410 COCE 442 411 SIGNAK(1) GO TO 410 COCE 444 412 SIGNAK(1) GO TO 410 COCE 444 413 SIGNAK(1) GO TO 410 COCE 444 410 SIGNAK(1) GO TO 410 COCE 444 410 SIGNAK(1) GO TO 410 COCE 444 410 SIGNAK(1) GO TO 410 COCE 444 411 SIGNAK(1) GO TO 410 COCE 445 411 SIGNAK(1) SIGNAK COCE 445 410	KVX = 1 CONE 442 GG TO 12 XI = SIGHM(1) CONE 444 YIN = I CONE 445 YIN = I CONE CONE YIN = I YIN = I CONE 445 YIN = I YIN = I <t< td=""><td>40 604X = 0 004E 442 41 604Y = 1 004E 444 41 604Y = 1 004E 444 41 604Y = 1 004E 444 41 851641(1) 004E 454 <td< td=""><td>400 60 Ye X = 0 000000000000000000000000000000000000</td><td>KAIN = 1</td><td>CORE 4</td><td>+41</td><td></td><td></td></td<></td></t<>	40 604X = 0 004E 442 41 604Y = 1 004E 444 41 604Y = 1 004E 444 41 604Y = 1 004E 444 41 851641(1) 004E 454 41 851641(1) 004E 454 <td< td=""><td>400 60 Ye X = 0 000000000000000000000000000000000000</td><td>KAIN = 1</td><td>CORE 4</td><td>+41</td><td></td><td></td></td<>	400 60 Ye X = 0 000000000000000000000000000000000000	KAIN = 1	CORE 4	+41		
400 620 12 000000000000000000000000000000000000	400 620 T0 12 000000000000000000000000000000000000	400 627 12 0002 444 7 166 16 145 444 7 15504X(11) 0008 444 7 55164X(11) 0008 444 7 55164X(11) 0008 444 7 55164X(11) 0008 449 7 55164X(11) 0008 449 7 55164X(11) 0008 449 1001 1012 1012 0008 449 1003 1001 0008 449 449 1003 1001 0008 449 446 1003 1001 100 0008 455 1003 1001 100 0008 455 1003 1001 100 0008 455 1003 1001 100 0008 455 1003 1001 100 0008 455 1003 1001 100 0008 455 1003 1004 100 0008 455 1003	40 6:0:10 12 0:0:2 4:4 7 7:(8K. 0) 60 4:4 7:1 7:(8K. 0) 60 4:4 7:1 7:10 7:0:0 4:4 7:1 7:10 7:10 0:0:0 7:1 7:10 7:10 0:0:0 7:1 7:10 7:10 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 7:1 7:10 0:0:0 0:0:0 <	400 6:0:10 12 0:0:2 4:4 7 7 7 6:0:10 12 7 7 7 6:0:10 12 7 7 7 6:0:10 12 7 7 7 6:0:10 12 7 7 7 6:0:10 10 7 7 7 6:0:10 10 7 7 7 6:0:10 10 7 7 7 6:0:10 10 7 7 7 6:0:10 10 7 7 7 6:0:10 10 7 7 7 6:0:10 10 7 7 7 6:0:10 10 7 7 7 7 7 1002 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 </td <td>400 600 600 440 F (K84L. 0) 60 T0 410 600 440 X.1 SIGNAMI(1) 600 440 X.2 SIGNAMI(1) 600 441 X.3 SIGNAMI(1) 600 441 X.4 SIGNAMI(1) 600 441 X.4 SIGNAMI(1) 600 600 Y.4 SIGNAMI(1) 2.00 600 Y.4 SIGNAMI(1) 2.00 600</td> <td>KMAX = D</td> <td>CORE 4</td> <td></td> <td></td> <td>:</td>	400 600 600 440 F (K84L. 0) 60 T0 410 600 440 X.1 SIGNAMI(1) 600 440 X.2 SIGNAMI(1) 600 441 X.3 SIGNAMI(1) 600 441 X.4 SIGNAMI(1) 600 441 X.4 SIGNAMI(1) 600 600 Y.4 SIGNAMI(1) 2.00 600 Y.4 SIGNAMI(1) 2.00 600	KMAX = D	CORE 4			:
400 KEY = 1. 400 KEY = 1. 1 F (K8. wf. 0) 60 T0 410 23 ESIGMAN(1) 2 = SIGMAN(1) 000E 1 = NETEAT 000E 1 = NETEAT <td>40 67 * 1 50 50 50 15 (8 * 1.) 50 10 50 15 (8 * 1.) 50 10 10 15 (8 * 1.) 50 10 10 15 516440(1) 50 10 10 10 1002 1001 1001 1001 1001 10 10 1002 1001 1001 1001 1001 10 10 10 1003 1001 1001 1001 1001 10<td>40 67 7</td><td>40 67 - 1.1 0.000 444 7 = 16 60 = 10 + 10 0.000 444 7 = 5164M1(1) 0.000 444 444 1001 1001 1001 0.000 444 1001 1001 1001 0.000 444 1001 1001 0.000 444 444 1001 1001 0.000 445 444 1001 1001 1001 0.000 445 1001 1001 1001 0.000 445 1001 1001 100 400 0.000 445 1011 1001 100 100 0.000 445 1011 1001 100 100 0.000 445 1011 1010 100 100 0.000 445 1011</td><td>400 KY = 1. 000 444 X = SIGHA(1) 001 001 001 X = SIGHA(1) 001 <</td><td>400 XYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY</td><td></td><td></td><td></td><td></td><td></td></td>	40 67 * 1 50 50 50 15 (8 * 1.) 50 10 50 15 (8 * 1.) 50 10 10 15 (8 * 1.) 50 10 10 15 516440(1) 50 10 10 10 1002 1001 1001 1001 1001 10 10 1002 1001 1001 1001 1001 10 10 10 1003 1001 1001 1001 1001 10 <td>40 67 7</td> <td>40 67 - 1.1 0.000 444 7 = 16 60 = 10 + 10 0.000 444 7 = 5164M1(1) 0.000 444 444 1001 1001 1001 0.000 444 1001 1001 1001 0.000 444 1001 1001 0.000 444 444 1001 1001 0.000 445 444 1001 1001 1001 0.000 445 1001 1001 1001 0.000 445 1001 1001 100 400 0.000 445 1011 1001 100 100 0.000 445 1011 1001 100 100 0.000 445 1011 1010 100 100 0.000 445 1011</td> <td>400 KY = 1. 000 444 X = SIGHA(1) 001 001 001 X = SIGHA(1) 001 <</td> <td>400 XYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY</td> <td></td> <td></td> <td></td> <td></td> <td></td>	40 67 7	40 67 - 1.1 0.000 444 7 = 16 60 = 10 + 10 0.000 444 7 = 5164M1(1) 0.000 444 444 1001 1001 1001 0.000 444 1001 1001 1001 0.000 444 1001 1001 0.000 444 444 1001 1001 0.000 445 444 1001 1001 1001 0.000 445 1001 1001 1001 0.000 445 1001 1001 100 400 0.000 445 1011 1001 100 100 0.000 445 1011 1001 100 100 0.000 445 1011 1010 100 100 0.000 445 1011	400 KY = 1. 000 444 X = SIGHA(1) 001 001 001 X = SIGHA(1) 001 <	400 XYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY					
15 18	TF (NBNE. 0) GO TO 410 000000000000000000000000000000000000	TF (KBKE. 0) GO TO 4.10 CONC. 445 X = SIGMAN(I) CONC. 446 X = SIGMAN(I) CONC. CONC. X = SIGMAN(I) L <	IF (K6KE. 0) GO TO 410 CORE 445 X.1 = SIGNA(I) CORE 449 X.2 = SIGNA(I) CORE 449 X.3 = SIGNA(I) CORE 449 X.4 = SIGNA(I) CORE 449 X.012 = WSTCP(I) INO1 = WSTCP(I) CORE X.1012 = INO1 CORE 449 X.1013 = INO1 CORE 452 X.1014 = INO1 CORE 452 X.1015 = INO1 CORE 454 X.1014 = INO1 CORE 455 X.1014 = INO1 CORE 455 X.1014 = INO1 CORE 455 X.1014 = INO1 CORE 456 X.1014 = INO1 CORE 456 X.1015 = NUCC(II) - 2.0 CORE 455 X.1014 = INO1 CORE 456	IF (K6KE. 0) GO TO 410 CORE 444 X1 = SIGNA(I1) CORE 444 X2 = SIGNA(I1) CORE 444 X2 = SIGNA(I1) CORE 444 X2 = SIGNA(I1) CORE 444 X3 = SIGNA(I1) CORE 444 X3 = SIGNA(I1) CORE 444 X4 = SIGNA(I1) CORE 444 X4 = SIGNA(I1) CORE 444 X4 = SIGNA(I1) CORE 444 X41 = IUU1 CORE 445 X41 = IUU1 CORE 445 X41 = IU1 CORE 445 X41 = IU1 CORE 454 X41 = IU1 CORE 455 X41 = IU1 CORE 456 X41 = IU1 CORE <td< td=""><td>IF (KGKE. 0) GO TO 410 CORE 444 II = SIGNAL(I) CORE 444 X3 = SIGNAL(I) CORE 444 X3 = SIGNAL(I) CORE 444 X4 = SIGNAL(I) CORE 444 Y401 = NGTCP(I) CORE 444 Y402 = NOI CORE 444 Y403 = NOI CORE 444 Y414 = 1 CORE CORE Y414 = 1 CORE CORE 444 Y414 = 1 CORE CORE</td><td></td><td></td><td></td><td></td><td></td></td<>	IF (KGKE. 0) GO TO 410 CORE 444 II = SIGNAL(I) CORE 444 X3 = SIGNAL(I) CORE 444 X3 = SIGNAL(I) CORE 444 X4 = SIGNAL(I) CORE 444 Y401 = NGTCP(I) CORE 444 Y402 = NOI CORE 444 Y403 = NOI CORE 444 Y414 = 1 CORE CORE Y414 = 1 CORE CORE 444 Y414 = 1 CORE CORE					
x1 = 5164X(1) 0.0000 445 x1 = 5164X(1) 0.0000 447 x2 = 5164X(1) 0.0000 447 x4 = 5164X(1) 0.0000 449 x003 = 1001 1002 451 1003 = 1001 0.0000 452 1004 = 1001 0.0000 452 1004 = 1001 0.0000 452 1004 = 1001 0.0000 452 1004 = 1001 0.0000 452 1004 = 1001 0.0000 452 1004 = 1001 0.0000 451 1000 = 01000 100000 401 1000 = 01000 100000 401 1000 = 010000 100000 452 1001 = 00000000000000000000000000000000	X1 = SIGHX(1) CORE 449 X2 = SIGHX(1) CORE 449 X3 = SIGHX(1) CORE 449 X4 = SIGHX(1) CORE 449 X3 = SIGHX(1) CORE 449 X4 = SIGHX(1) CORE 449 X3 = SIGHX(1) CORE 441 Y4 = SIGHX(1) CORE 441 Y4 = SIGHX(1) CORE 441 Y1 = 1001 EIN01 CORE 441 Y1 = 0 CORE 441 CORE Y1 = 0 CORE 441 CORE 441 Y1 = 0 CORE CORE 441 Y1 = 0 CORE 441 CORE 441 Y1 = 0 CORE CORE 441 441 Y1 = SIGHX(1) CORE 441 CORE 441 Y1 = SIGHX(1)	x1 = SIGHAK(1) CORE 449 x2 = SIGHAK(1) CORE 449 x3 = SIGHAK(1) CORE 449 x3 = SIGHAK(1) CORE 449 x4 = SIGHAK(1) CORE 449 x3 = SIGHAK(1) CORE 449 x4 = SIGHAK(1) CORE 449 x3 = SIGHAK(1) CORE 449 x4 = SIGHAK(1) CORE 449 x3 = NOL CORE 449 x4 = SIGHAK(1) CORE 449 x4 = SIGHAK(1) CORE 449 x4 = SIGHAK(1) CORE 449 x4 = NOL CORE 449 x4 = SIGHAK(1) CORE 456 x4 = SIGHAK(1) SIGHAK(1) CORE x4 = SIGHAK(1) SIGHAK(1) CORE	X1 = SIG = M(X) CORE 445 X2 = SIG = M(X) CORE 449 X4 = SIG = M(X) CORE 449 X4 = SIG = M(X) CORE 449 X1 = SIG = M(X) CORE 449 X1 = SIG = M(X) CORE 449 X1 = SIG = M(X) CORE 449 Y1 = NOL CORE 449 Y1 = NOL CORE 454 Y1 = NOL CORE 455 Y1 = SIG M(Y) CORE 455 Y1 = NOL CORE 455 Y1 = SIG N(Y) CORE 455 Y1 = SIG N(Y) CORE 455	X1 F NG FAR: U D 0 0 0 410 CORE 449 X2 = SIGHN(I) CORE 449 X3 = SIGHN(I) CORE 449 X4 = SIGHN(I) CORE 449 Y41 Y41 CORE 449 Y41 Y41 CORE 449 Y41 Y41 CORE 459 Y41 X41 X41 CORE 456 Y41 X41 X41 X41 X41 Y41 X41 <td>X1 X1 <td< td=""><td></td><td></td><td>+ L</td><td></td><td>:</td></td<></td>	X1 X1 <td< td=""><td></td><td></td><td>+ L</td><td></td><td>:</td></td<>			+ L		:
X1 = SIGHN(I) X2 = SIGHN(I) X3 = SIGHN(I) X3 = SIGHN(I) X4 = S	X1 = SIGHNI(1) X2 = SIGHNI(1) CORE 446 X3 = SIGHNI(1) X3 = SIGHNI(1) CORE 446 X4 = SIGHNI(1) X4 = SIGHNI(1) CORE 446 X4 = SIGHNI(1) CORE 446 446 X002 = IN01 IN02 = IN01 CORE 446 IN03 = SIGHNI(1) CORE 451 CORE 451 IN03 = IN01 IN03 = IN01 CORE 451 CORE 451 IN04 = IN01 CORE 451 CORE 451 CORE 451 KHX = 0 CORE CORE 456 CORE 456 CORE 456 KHX = 0 CORE CORE 456 CORE 456 CORE 456 KHX = 0 CORE CORE 456 CORE 456 CORE 456 KHX = 0 CORE CORE 456	X1 = SIGHMX(1) CORE 446 X2 = SIGHMX(1) CORE 446 X4 = SIGHMX(1) CORE 441 X002 = 1001 IN03 = 1001 CORE 441 X004 = 1001 CORE 453 CORE 453 X1NA = 0 CORE CORE 453 CORE 453 X1NA = 0 CORE CORE 453 CORE 453 X1NA = 0 CORE CORE 453 CORE 453 X1NA = 1 CORE CORE 453 CORE 453 X1NA = 1 CORE CORE 453 <td< td=""><td>X1 = SIGHMA(I) CORE 446 X2 = SIGHMA(I) CORE 446 X3 = SIGHMA(I) CORE 446 1001 = NGTCP(I) INO1 = NGTCP(I) CORE 446 1001 = NGTCP(I) ENCIDION CORE 446 1001 = INO1 CORE 446 CORE 446 1001 = INO1 CORE 447 CORE 446 1001 = INO1 CORE 446 CORE 446 1001 = INO1 CORE 440 CORE 446 101 = SIGMAN(I) CORE 446 CORE 446 101 = SIGMAN(I) CORE 446 CORE 446 101 = SIGMAN(I) CORE 440 CORE 446 102 = INO1 SIGMAN(I) CORE 446 CORE 446</td><td>X1 = SIGHMA(I) CORE 446 X2 = SIGHMA(I) CORE 446 X3 = SIGHMA(I) CORE 446 X4 = SIGHMA(I) CORE 446 1001 = NNTEP(1) CORE 446 101 = NNTEP(1) CORE 446</td><td>XI = SIGHAU(1) CORE 446 XI = SIGHAU(1) CORE 446 XI = SIGHAU(1) CORE 449 XI = SIGHAU(1) CORE 449 XIOI = NGILP(1) NOI = SIGHAU(1) CORE XIOI = SIGHAU(1) CORE 449 YIOI = HOI CORE 449 YIOI = HOI CORE 449 YIOI = FIOI CORE 449 YIOI = FIOI CORE 445 YIOI = FIOI CORE 445 YIOI = FIOI CORE 445 YIOI = FIOI CORE 454 YIOI = FIOI CO</td><td>IF (KB • NE• U) 60 TO 410</td><td>CORE 4</td><td>++ D</td><td></td><td></td></td<>	X1 = SIGHMA(I) CORE 446 X2 = SIGHMA(I) CORE 446 X3 = SIGHMA(I) CORE 446 1001 = NGTCP(I) INO1 = NGTCP(I) CORE 446 1001 = NGTCP(I) ENCIDION CORE 446 1001 = INO1 CORE 446 CORE 446 1001 = INO1 CORE 447 CORE 446 1001 = INO1 CORE 446 CORE 446 1001 = INO1 CORE 440 CORE 446 101 = SIGMAN(I) CORE 446 CORE 446 101 = SIGMAN(I) CORE 446 CORE 446 101 = SIGMAN(I) CORE 440 CORE 446 102 = INO1 SIGMAN(I) CORE 446 CORE 446	X1 = SIGHMA(I) CORE 446 X2 = SIGHMA(I) CORE 446 X3 = SIGHMA(I) CORE 446 X4 = SIGHMA(I) CORE 446 1001 = NNTEP(1) CORE 446 101 = NNTEP(1) CORE 446	XI = SIGHAU(1) CORE 446 XI = SIGHAU(1) CORE 446 XI = SIGHAU(1) CORE 449 XI = SIGHAU(1) CORE 449 XIOI = NGILP(1) NOI = SIGHAU(1) CORE XIOI = SIGHAU(1) CORE 449 YIOI = HOI CORE 449 YIOI = HOI CORE 449 YIOI = FIOI CORE 449 YIOI = FIOI CORE 445 YIOI = FIOI CORE 445 YIOI = FIOI CORE 445 YIOI = FIOI CORE 454 YIOI = FIOI CO	IF (KB • NE• U) 60 TO 410	CORE 4	++ D		
X2 = SIGHIN(1) X4 = 100 X4 =	X2 \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$	X2 \$516HX(I) CORE 447 X4 \$16HX(I) CORE 449 X4 \$16HX(I) CORE 449 IND1 #NSEP(I) CORE 449 IND2 IND2 IND2 \$100 IND2 IND2 IND2 \$100 IND2 IND2 CORE 451 IND3 IND1 CORE 452 IND3 IND1 CORE 452 IND3 IND1 CORE 452 IND3 IND1 CORE 452 IND4 IND1 CORE 453 KHIN IND1 CORE 453 KHIN IND1 CORE 453 KNCVCID RNOYCCID CORE 454 KNOYCCID RNOYCCID CORE 454 KNOYCCID CORE 454 454 KNOYCCID CORE 454 455 KNOYCCID CORE 454 455 KNOYCCID CORE 454 455 KNOYCCID	X3 = SIGHX(I) CORE 447 X4 = SIGHX(I) CORE 449 X103 = IN01 IN02 = NSIGHX(I) CORE 449 IN02 = NSIGHX(I) CORE 449 441 IN03 = IN01 IN03 = IN01 CORE 445 IN03 = IN01 IN03 = IN01 CORE 452 IN04 = IN01 CORE 452 CORE 452 X11 = 0 CORE 452 CORE 455 X11 = 0 CORE 451 CORE 455 X11 = 0 CORE 451 CORE 455 X11 = 0 CORE 451 CORE 455 X12 = 0 CORE CORE 455 CORE 455 X12 = 0 CORE CORE 455 CORE 455 X13 = 0 CORE CORE 455 CORE 455 X13 = 0 CORE CORE 455 CORE 455 X10 = CORE X10 + COR CORE 455 CORE 455 X10 = CORE CORE CORE 455	X2 SIGHIN(1) CORE 447 X3 SIGHAN(1) CORE 449 NU01 NIXEP(1) CORE 449 NU01 NIXEP(1) CORE 449 NU01 NIXEP(1) CORE 449 NU01 NIXEP(1) CORE 441 NU01 NIXEP(1) CORE 441 NU01 NIXEP(1) LE. 2.0) CORE 441 NIXE CORE 442 CORE 445 NIXE NIXEP(1) LE. 2.0) COPE 445 NIXE CORE 445 CORE 445 NIXE SIGNIN(1) NIXEP(1) 2.0 CORE 445 NIXE SIGNIN(1) NIXEP(1) 2.0 <td< td=""><td>X2 SIGHAN(1) CORE 447 X3 SIGHAN(1) CORE 449 X4 SIGHAN(1) CORE 449 X003 HOU1 FIGHAN CORE 449 X4 SIGHAN CORE 449 X103 HOU1 ENCORCI 451 X104 HOU1 ENCORCI 451 X105 FIGHAN CORE 451 X104 HOU1 ENCORCI ENCORCI 451 X105 ENCORCI ENCORCI ENCORCI 451 X105 ENCORCI ENCORCI ENCORCI 451 X105 ENCORCI ENCORCI ENCORCI 451 X106 ENCORCI ENCORCI ENCORCI 451 X107 ENCORCI ENCORCI<!--</td--><td>x1 = SIGMAX(I)</td><td>CORE 4</td><td>++6</td><td></td><td></td></td></td<>	X2 SIGHAN(1) CORE 447 X3 SIGHAN(1) CORE 449 X4 SIGHAN(1) CORE 449 X003 HOU1 FIGHAN CORE 449 X4 SIGHAN CORE 449 X103 HOU1 ENCORCI 451 X104 HOU1 ENCORCI 451 X105 FIGHAN CORE 451 X104 HOU1 ENCORCI ENCORCI 451 X105 ENCORCI ENCORCI ENCORCI 451 X105 ENCORCI ENCORCI ENCORCI 451 X105 ENCORCI ENCORCI ENCORCI 451 X106 ENCORCI ENCORCI ENCORCI 451 X107 ENCORCI ENCORCI </td <td>x1 = SIGMAX(I)</td> <td>CORE 4</td> <td>++6</td> <td></td> <td></td>	x1 = SIGMAX(I)	CORE 4	++6		
x3 = SIGHAX(I) CORE 449 x4 = SIGHAX(I) NO1 = NSIGHAX(I) CORE 449 IN01 = NSIGHAX(I) NO1 = NSIGHAX(I) CORE 449 IN02 = IN01 IN03 = IN01 CORE 451 IN02 = IN01 IN03 = IN01 CORE 451 IN02 = IN01 IN04 = IN01 CORE 451 KHAX = 0 CORE 452 CORE 453 KHAX = 0 CORE 451 CORE 455 KHAX = 0 CORE 452 CORE 455 KHAX = 0 CORE 453 CORE 455 KHAX = 0 CORE 456 CORE 455 KHAX = 0 CORE 456 CORE 455 VOID = 0 CORE 456 CORE 455 VID = 1 CORE 456 CORE 456 VID = 1 CORE 456 CORE 456 VID = 1 CORE CORE 456 CORE 456 VID = 1 CORE CORE 456 CORE 456	X3 = SIGHAX(I) CORE 448 X4 = SIGHAX(I) NOR E 101 1001 = NSTECHINI IN01 = NSTECHINI NOR E 101 451 IN02 = IN01 NOR E 451 00RE 451 IN03 = IN01 NOR E 101 00RE 451 IN04 = IN01 NOR E 101 00RE 451 KHAX = 0 CORE 451 00RE 451 KHAX = 1 CORE 451 00RE 451 KHAX = 0 CORE 451 00RE 451 KHAX = 1003 KHAX = 0 CORE 451	X3 = SIGHMA(I) 00RE 449 X4 = SIGHMA(I) 00RE 449 IN01 = NSICII) 00RE 451 IN02 = IN01 00RE 451 IN03 = IN01 00RE 452 IN04 = IN01 00RE 452 IN04 = IN01 00RE 452 IN04 = IN01 00RE 455 IN04 = IN01 00RE 456 IN04 = IN01 00RE 456 IN0 = 0 00RE 456 IN04 = IN03 00RE 455 IN04 = IN03 00RE 456	X3 = SIGHX(I) X102 = INDI IND2 = INDI IND2 = INDI IND2 = INDI IND2 = INDI IND2 = INDI IND3 = INDI IND3 = INDI IND3 = INDI IND4 = INDI IND4 = INDI IND4 = INDI IND4 = INDI IF (RAUYCII) -LE, 2-0) GO TO 4-D1 KHX = 0 CORE CORE CORE F57 CORE	X3 = SIGHX(I) CORE 449 X4 = SIGHX(I) CORE 449 IND1 = IND1 IND1 = IND1 CORE 451 IND1 = IND1 ENCPTI CORE 451 IND1 = IND1 ENCPCII LL CORE 451 ENDA = IND1 ENCPCII LL CORE 451 ENDA = IND1 ENCPCII LL CORE 455 ENDA = IND1 ENCPCII LL CORE 455 ENDA = IND1 ENCPCII LL CORE 456 ENDA = IND1 ENCPCII LL CORE 456 ENCPCII ENCPCII L CORE 456 ENCPCII ENCPCII L <t< td=""><td>x3 = SIGHAX(1) 00RE 449 Nu1 = SIGFIN(1) 1N01 = SIGFIN(1) 00RE 449 NU0 = FUO1 000RE 449 449 NU0 = FUO1 000RE 449 449 NU0 = FUO1 00RE 451 00RE 453 NU0 = FUO1 00RE 451 00RE 453 NU1 = FUO1 00RE 451 00RE 451 NU1 = FUO1 00RE 451 00RE 451 NU1 = FUO1 FUO1011 - 1.0 00RE 451 00RE 451 NU1 = FUO10101 - 2.0 00RE 451 00RE 451 451 NU1 = FUO1011 - 2.0 00RE 451 00RE 451 451 NU1 = FUO1011 - 2.0 00RE 451 00RE 451 451 NU1 = FUO1011 - 1.0 00RE 451 00RE 451 451 451 NU1 = FUO1011 - 1.0 00RE 451 00RE 451 451 451 NU1 = FUO1011 - 1.0 00RE 00RE 451 00RE 451 451</td><td>X2 = SIGMIN(I)</td><td>CORE 4</td><td>++7</td><td></td><td></td></t<>	x3 = SIGHAX(1) 00RE 449 Nu1 = SIGFIN(1) 1N01 = SIGFIN(1) 00RE 449 NU0 = FUO1 000RE 449 449 NU0 = FUO1 000RE 449 449 NU0 = FUO1 00RE 451 00RE 453 NU0 = FUO1 00RE 451 00RE 453 NU1 = FUO1 00RE 451 00RE 451 NU1 = FUO1 00RE 451 00RE 451 NU1 = FUO1 FUO1011 - 1.0 00RE 451 00RE 451 NU1 = FUO10101 - 2.0 00RE 451 00RE 451 451 NU1 = FUO1011 - 2.0 00RE 451 00RE 451 451 NU1 = FUO1011 - 2.0 00RE 451 00RE 451 451 NU1 = FUO1011 - 1.0 00RE 451 00RE 451 451 451 NU1 = FUO1011 - 1.0 00RE 451 00RE 451 451 451 NU1 = FUO1011 - 1.0 00RE 00RE 451 00RE 451 451	X2 = SIGMIN(I)	CORE 4	++7		
XX = SIGHIN(I) XX = SIGHIN(I) IND1 = NSTEP(I) SIGHIN(I) IND2 = IND1 SIGHIN(I) IND4 = IND1 SIGHIN(I) XHA = 0 SIGHIN(I) XHA = 1 SIGHIN(I) XHA = 1 SIGHIN(I) XHA = 0 SIGHIN(I) XH = 1 SIGHIN(I) RUCYC(I) = RUCYC(I) = RUCYC(I) SIGHIN(I) RUCYC(I) = RUCYC(I) = RUCYC(I) SIGHIN(I) RUCYC(I) = RUCYC(I) = RUCYC(I) SIGHIN(I) L0 SIGHIN(I) SIGHIN(I) L1 SIGHIN(I) SIGHIN(I) L0 SIGHIN = 0 SIGHIN(I) L0 +15 SIGHIN(I) SIGHIN(I) L1 SIGHIN(I) SIGHIN(I) L1 SIGHIN(I) SIGHIN(I) L1 SIGHIN(I) SIGHIN(I) L1 SIGHIN(I) SIGHIN(I) <tr< td=""><td>X4 = SIGHIN(I) X01 = NISEP(I) IND1 = NISEP(I) S026 IND2 = IND1 S026 IND2 = IND1 S026 IND2 = IND1 S026 IND2 = IND1 S026 IND3 = IND1 S026 IND3 = IND1 S026 IND4 = IND1 S026 IND4 = IND1 S026 IND4 = IND1 S026 IND4 = IND1 S026 IND = O S026 IND = O S026 S1 = O S0276 INCYCID = RNCYCID = LO S026 RCOTOGID = RNCYCID = LO S026 RCOTOGID = RNCYCID = LO S026 RCOTOGID = RNCYCID = S027 S026 RCOTOGID = RNCYCID = S026 S026 LO 445 S027 S026 LO 445 S026 S026 LO 445 S0276 S026</td><td><pre>X4 = SIGMIN(I) IND2 = IND1 IND2 = IND1 IND2 = IND1 IND2 = IND1 IND4 = IND3 IND4 = IND4 IND4 = IND3 IND4 = IND4 IND4 = IND</pre></td><td>X4 = SIGMIN(1) X4 = SIGMIN(1) IND2 = IND1 IND2 = IND1 IND2 = IND1 IND2 = IND1 IND3 = IND1 IND2 = IND1 IND4 = IND1 CORE XHN = 1 CORE</td><td>X4 = SIGHN(I) CORE 490 IN01 = NSTEP(I) CORE 490 IN02 = IN01 CORE 490 IN02 = IN01 CORE 490 IN03 = IN01 CORE 490 IN04 = NO1 CORE 490 IN05 = IN01 CORE 490 IN04 = NO1 CORE 490 IN04 = IN01 CORE 490 IN04 = IN01 CORE 490 KHIN = 1 CORE 490 KOUNCELI) = RAUYCELI) = 1.0 CORE 490 GO TO 402 CORE 490 GO TO 402 CORE 490 KINO = 0 CORE 490 LO 10 415 CORE 490 LO 10 415 CORE 490 LO 10 415 CORE 490 KINO = 0 CORE 490 KINO = 1 CORE 490 KINO = 1 CORE 490 <tr< td=""><td>X4 = SIGHN(1) X4 = SIGHN(1) C002 440 IND1 = NSIEP(1) IND1 = NSIEP(1) C002 440 IND2 = INOL INOL = NSIEP(1) C002 440 INOL = INOL INOL = INOL C002 440 INOL = INOL C002 440 C004 440 X41 = 0 X11 = 0 C002 445 C002 445 X11 = 0 X11 = 0 C002 445 C002 445 X11 = 0 X11 = 0 C002 C002 445 C002 445 X11 = 0 X11 = 0 C002 C002 445 C003 445 C003 445 C003 445 C003 445 C003 445 C003 445 C003</td><td>X3 = SIGHAX(I)</td><td>CORE 4</td><td>• • 8</td><td>•</td><td>i</td></tr<></td></tr<>	X4 = SIGHIN(I) X01 = NISEP(I) IND1 = NISEP(I) S026 IND2 = IND1 S026 IND2 = IND1 S026 IND2 = IND1 S026 IND2 = IND1 S026 IND3 = IND1 S026 IND3 = IND1 S026 IND4 = IND1 S026 IND4 = IND1 S026 IND4 = IND1 S026 IND4 = IND1 S026 IND = O S026 IND = O S026 S1 = O S0276 INCYCID = RNCYCID = LO S026 RCOTOGID = RNCYCID = LO S026 RCOTOGID = RNCYCID = LO S026 RCOTOGID = RNCYCID = S027 S026 RCOTOGID = RNCYCID = S026 S026 LO 445 S027 S026 LO 445 S026 S026 LO 445 S0276 S026	<pre>X4 = SIGMIN(I) IND2 = IND1 IND2 = IND1 IND2 = IND1 IND2 = IND1 IND4 = IND3 IND4 = IND4 IND4 = IND3 IND4 = IND4 IND4 = IND</pre>	X4 = SIGMIN(1) X4 = SIGMIN(1) IND2 = IND1 IND2 = IND1 IND2 = IND1 IND2 = IND1 IND3 = IND1 IND2 = IND1 IND4 = IND1 CORE XHN = 1 CORE	X4 = SIGHN(I) CORE 490 IN01 = NSTEP(I) CORE 490 IN02 = IN01 CORE 490 IN02 = IN01 CORE 490 IN03 = IN01 CORE 490 IN04 = NO1 CORE 490 IN05 = IN01 CORE 490 IN04 = NO1 CORE 490 IN04 = IN01 CORE 490 IN04 = IN01 CORE 490 KHIN = 1 CORE 490 KOUNCELI) = RAUYCELI) = 1.0 CORE 490 GO TO 402 CORE 490 GO TO 402 CORE 490 KINO = 0 CORE 490 LO 10 415 CORE 490 LO 10 415 CORE 490 LO 10 415 CORE 490 KINO = 0 CORE 490 KINO = 1 CORE 490 KINO = 1 CORE 490 <tr< td=""><td>X4 = SIGHN(1) X4 = SIGHN(1) C002 440 IND1 = NSIEP(1) IND1 = NSIEP(1) C002 440 IND2 = INOL INOL = NSIEP(1) C002 440 INOL = INOL INOL = INOL C002 440 INOL = INOL C002 440 C004 440 X41 = 0 X11 = 0 C002 445 C002 445 X11 = 0 X11 = 0 C002 445 C002 445 X11 = 0 X11 = 0 C002 C002 445 C002 445 X11 = 0 X11 = 0 C002 C002 445 C003 445 C003 445 C003 445 C003 445 C003 445 C003 445 C003</td><td>X3 = SIGHAX(I)</td><td>CORE 4</td><td>• • 8</td><td>•</td><td>i</td></tr<>	X4 = SIGHN(1) X4 = SIGHN(1) C002 440 IND1 = NSIEP(1) IND1 = NSIEP(1) C002 440 IND2 = INOL INOL = NSIEP(1) C002 440 INOL = INOL INOL = INOL C002 440 INOL = INOL C002 440 C004 440 X41 = 0 X11 = 0 C002 445 C002 445 X11 = 0 X11 = 0 C002 445 C002 445 X11 = 0 X11 = 0 C002 C002 445 C002 445 X11 = 0 X11 = 0 C002 C002 445 C003	X3 = SIGHAX(I)	CORE 4	• • 8	•	i
IND1 = NSTERIL1 CORE 431 IND2 = INOL INO2 = INOL 431 IND3 = INOL CORE 431 IND4 = INOL CORE 431 IND4 = INOL CORE 431 IND3 = NOL CORE 431 IND4 = INOL CORE 431 KHAX = 0 CORE 435 KHAX = 0 CORE 435 KAX = 0 CORE 436 AD RUCK(I) = RUCK(I) = 2.0 CORE 435 AD RUCK(I) = RUCK(I) = 2.0 CORE 435 AD RUCK(I) = RUCK(I) = 2.0 CORE 435 AD RUCK(I) = RUCK(I) = 2.0 CORE 445 AD RUCK(I) = RUCK(I) = 2.0 CORE 445 AD RUCK = 1 CORE 455 AD RUCK = 1 CORE 455 AD RUCK = 1 CORE 455 AD A RUCK = 1 CORE 455 AD A RUCK = 1 CORE 455	IND1 = NSTEPT1) CORE F31 IND2 = IND1 IND3 = IND1 IND3 = IND1 CORE F31 IND3 = IND1 CORE F32 IND4 = IND1 CORE F33 IND4 = IND1 CORE F35 KHX = 0 CORE F35 KHX = 1 CORE F35 KHX = 1 CORE F35 KHX = 0 CORE F35 KHX = 0 CORE F35 KHX = 1 CORE F35 KHX = 0 CORE F35 KHX = 1 CORE F35 RUCYCII) = RUCYCII) - 1.0 CORE F35 401 RUCYCII) = RUCYCII) - 2.0 CORE F35 402 KIN0 = 0 CORE F35 403 S OF 10 401 CORE F45 404 RUCYCII) = RUCYCIII - 2.0 CORE F45 405 KIN0 = 0 CORE F45 406 KIN0 = 0 CORE F45 407 KIN = 1 CORE F45 KMN = 1 CORE F45 KMN = 1 CORE F45<	IND1 = NSTERII IND2 = IND1 IND2 = IND1 IND2 = IND1 IND2 = IND1 IND2 = IND1 IND4 = IND1 IND4 = IND1 IND4 = IND1 IND4 = IND1 IND4 = IND1 IND4 = IND1 IS ACC(II) = ENCTC(II) - L.0 ENCTC(II) = ENCTC(I	IND1 = NSTEP(1) 0000 471 IND2 = IND1 00000 471 IND3 = IND1 00000 475 IND = NSTEP(1) 00000 475 IND = NSTEP(1) 00000 475 IND = NOL 00000 455 IND = NOL 000000 455 IND = NOL 00000000000 455 IND = NOL 000000000000000000000000000000000000	IND1 = NSTEPL1 CORE 451 IND2 = IND1 IND2 = IND1 CORE 451 IND3 = IND1 IND3 = IND1 CORE 451 IND = NSTEPL1 CORE 451 CORE 451 IND = NSTEPL1 CORE 451 CORE 451 IND = NSTEPL1 CORE 451 CORE 455 IND = NOTCID RNOYCID ENCYCID CORE 455 IF (ANOYCID) = RNOYCID CORE 455 CORE 455 ADD 402 CONE CORE 455 CORE 455 ADD 401 CORE CORE 455 CORE 455 ADD 401 CORE CORE 455 CORE 455 ADD 401 CORE CORE 455	IND1 = NSTERI1 CORE 431 IND2 = IND1 IND3 = IND1 CORE 431 IND3 = IND1 IND3 = IND1 CORE 431 IND = NSTERI1 CORE 431 CORE 431 KHAN = 0 KHAN = 0 CORE 431 CORE 435 KHAN = 0 KHAN = 0 CORE 435 CORE 435 KHAN = 0 KNCYCII) = RNCYCII > 1.0 CORE 435 CORE 435 KNCYCII) = RNCYCII > 2.0 GO TO 401 CORE 435 CORE 435 KNCYCII = RNCYCII > 2.0 GO TO 401 CORE 435 CORE 435 KNCYCII = RNCYCII > 2.0 GO TO 401 CORE 435 CORE 435 LO TO 410 RNCYCII = 2.0 GO TO 410 CORE 435 CORE 435 LA = SIGHAN(I) XI = 2.0 CORE 435 CORE 435 LA = SIGHAN(I) XI = 2.0 CORE 435 CORE 435 LA = SIGHAN(I) XI = 2.0 CORE 435 CORE 435 KAA = SIGHAN(
IND2 = INDL IND2 = INDL IND2 = INDL IND3 = INDL IND3 = INDL IND3 = INDL IND4 = INDL IND4 = INDC IND4 = INDC IND4 = INDC IND4 = IND5 IND4 = IND5 IND4 = IND5 IND4 = IND5 IND4 = IND5 IND6 <td>IND2 = NALL NOT NOT</td> <td>1002 1002 1002 1002 1003 1001 1003 1001 1003 1001 1003 1001 1004 1004 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1000 1000 1000 101 101 100 100 1000 101 1000 100 1000 1000 101 1000 100 100 1000 101 1000 100 1000 1000 101 1000 100 1000 1000 101 1000 100 1000 1000 101 1000 1000 1000 1000 101 1000 1000 1000 1000 101 1000 1000 1000 1000 101 1000 1000 1000 1000 101 1000 1000</td> <td>1002 1001 1002 1001 1003 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1005 101 100 1001 1001 100 100 1001 1001 100 100 100 1001 1001 100 100 100 1001 1001 100 100 100 1001 1001 100 100 100 1001 1001 100 100 100 100 1001 100 100 100 1000 1001 100 100 100 1000 1001 100 100 1000 1000 10</td> <td>1002 1001 1002 1001 1003 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 100 1001 1004 100 100 1001 100 100 100 1001 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100</td> <td>1000 10000 1000</td> <td></td> <td></td> <td></td> <td></td> <td></td>	IND2 = NALL NOT	1002 1002 1002 1002 1003 1001 1003 1001 1003 1001 1003 1001 1004 1004 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1000 1000 1000 101 101 100 100 1000 101 1000 100 1000 1000 101 1000 100 100 1000 101 1000 100 1000 1000 101 1000 100 1000 1000 101 1000 100 1000 1000 101 1000 1000 1000 1000 101 1000 1000 1000 1000 101 1000 1000 1000 1000 101 1000 1000 1000 1000 101 1000 1000	1002 1001 1002 1001 1003 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1005 101 100 1001 1001 100 100 1001 1001 100 100 100 1001 1001 100 100 100 1001 1001 100 100 100 1001 1001 100 100 100 1001 1001 100 100 100 100 1001 100 100 100 1000 1001 100 100 100 1000 1001 100 100 1000 1000 10	1002 1001 1002 1001 1003 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 1001 1001 1004 1001 100 1001 1004 100 100 1001 100 100 100 1001 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100	1000 10000 1000					
1003 = 1001 1003 = 1001 1003 = 1001 1003 = 1001 1004 = 1001 100 = 100 1004 = 1001 100 = 100 101 = 100 100 = 100 100 = 100 100 = 100 100 = 100 100 = 100 100 = 100 100 100 = 10	1003 = 1001 1003 = 1001 1003 = 1001 1003 = 1001 1004 = 1001 1000 = 1001 1014 = 1 1001 1015 = 1001 1001 1016 = 1001 1001 1017 = 1 1001 1018 = 1 1001 1018 = 1 1001 1018 = 1 1001 1109 = 1 1001 1100 + 1 10000 1100 + 1 100000 1100 + 1 100000 1100 + 1 100000 1100 + 1 100000 1100 + 1 100000 1100 + 1 100000 1100 + 1 100000 1100 + 1 100000 1100 + 1 1000000 1100 + 1 10000000 1100 + 1 10000000000 1100 + 1 1000000000000000000000000000000000000	1000 1000 1000 1000 10004 1001 1000 1000 1000 10014 1001 1001 1001 1000 1000 10014 1001 1000 1000 1000 1000 1000 1001 1000 1000 1000 1000 1000 1000 1000 1001 10000 1000 1000	1002 1002 401 1004 = 1001 1004 = 1001 1004 = 1001 1005 453 1004 = 1001 1005 453 1005 11 005 455 11 11 005 455 11 11 10 005 455 11 11 11 005 11 005 11 11 11 11 005 11 11 11 </td <td>IND2 = IND1 CORE 451 IND4 = IND1 CORE 453 KHAX = CORE 453 KHAX = CORE 453 KHAX = CORE 453 KHAX = CORE 455 VID<=</td> = CORE 455 VID<=	IND2 = IND1 CORE 451 IND4 = IND1 CORE 453 KHAX = CORE 453 KHAX = CORE 453 KHAX = CORE 453 KHAX = CORE 455 VID<=	MUX T MUX T MUX <					:
INUUS INUUS INUUS INUUS INUUS INUUS INUUS INUUS KHIN 1 OUNE 475 KHIN 1 CORE 475 KHIN 0 CORE 475 KINU 0 CORE 456 401 RUCYCII) ENCYCII) 2.0 401 RUCYCII) ENCYCII 2.0 401 RUCYCII) ENCYCII 2.0 401 RUCYCIII ENCYCIII 2.0 401 RUCYCIII ENCYCIII 2.0 410 XINU 1 0.0 0.0 501 ENCH 0.0 0.0 0.0 501 ENCYCIII ENCYCIII 2.0 0.0 501 ENCH 10 0.0 0.0	INUUS INUUS INUUS INUUS INUUS INUUS INUUS INUUS KHIN 1 OUNE 472 KHIN 1 CORE 453 KHIN 1 CORE 455 KHIN 1 CORE 456 KOUCID RNCYCID CORE 456 401 RNCYCID RNCYCID CORE 401 RNCYCID CORE 456 402 CORE 451 456 733 SCHMAKID CORE <td>1004 1001 KHIN = 1 00000 KHIN = 1 00000</td> <td>IND4 = 1NU1 UNU4 = 1 IND4 = 1 UNU1 KHIN = 1 UNU1 GO TO 402 CORE KHIN = 1 CORE MOD TO 415 CORE 402 KIND = 1 CORE KHIN = 1 CORE</td> <td>INU05 = INU1 INU05 = INU1 INU05 = INU1 000000000000000000000000000000000000</td> <td>INU05 = INU1 CORE 452 INU04 = 1 CORE 455 KHIN = 1 CORE <t< td=""><td></td><td></td><td>101</td><td></td><td></td></t<></td>	1004 1001 KHIN = 1 00000	IND4 = 1NU1 UNU4 = 1 IND4 = 1 UNU1 KHIN = 1 UNU1 GO TO 402 CORE KHIN = 1 CORE MOD TO 415 CORE 402 KIND = 1 CORE KHIN = 1 CORE	INU05 = INU1 INU05 = INU1 INU05 = INU1 000000000000000000000000000000000000	INU05 = INU1 CORE 452 INU04 = 1 CORE 455 KHIN = 1 CORE <t< td=""><td></td><td></td><td>101</td><td></td><td></td></t<>			101		
<pre>ND4 = IN01 FIND4 = IN01 KHIN = 1 KHA = 0 KHA = 0 F (RUCKI) = RUCKI) = 1.0 0 00 402 MU0 = 0 401 RUCK(I) = RUCK(I) = 2.0 401 RUCK(I) = RUCK(I) = 2.0 401 RUCK(I) = RUCK(I) = 2.0 401 RUCK(I) = 1.0 KHA = 1 RUCK(I) = RUCK(I) = 1.0 CORE 455 CORE 455</pre>	INO4 = INO1 INO4 = INO1 KHX = 1 00RE 453 KHX = 1 KHX = 1 00RE 454 KHX = 1 KHX = 1 00RE 455 KHX = 1 KUC (KIL) - 1.0 00RE 455 AUC (KIL) = RUCYCIL) - 2.0 00RE 455 401 KUC (CI) = RUCYCIL) - 2.0 00RE 455 402 KINU = 0 00RE 455 403 SISTEPLI) 00RE 455 10 3 = SIGHIN(I) 00RE 455 10 415 KHX = 1 00RE 455 10 416 KHX = 1 00RE 455 10 418 = I KHX = 1 00RE 455 10 418 = 1 KHX = 1 00RE 455 10 410 = 1 KHX = 1 00RE 455 10 410 = 1 KHX = 1 00RE 455 10 410 = 1 KHX = 1 00RE 455 KHX = 1 <td< td=""><td>IN04 = IN01 IN04 = IN01 KHX = 1 CORE 453 KHX = 1 KHX = 1 KHX = 1 KHX = 1 KHX = 1 KHX = 1 KHX = 1 KROCC(I) .LE. 2.0) GO TO 401 F KOVC(I) .LE. 2.0) GO TO 401 CORE KNOVC(I) = RNOVC(I) - 1.0 CORE KNOVC(I) = RNOVC(I) - 2.0 CORE MODE CORE KIN = 1 CORE KNON = 1 CORE KNON = 1 CORE KNON = 1 CORE KIN = 1 C</td><td>TN04 = IN01 TN04 = IN01 KTN = 1 0 KHN = 1 0.010 402 KHN = 0 0.010 402 KHN = 0 0.010 402 KHN = 1 0.010 415 KHN = 1 0.010 4101 KHN = 1</td><td>TN04 = IN01 CORE 453 KTN = 1 0 CORE 454 KHN = 1 CORE 455 455 KHN = 1 CORE 455 455 KHN = 1 CORE 455 55 KHN = 1 CORE 455 55 KHN = 1 CORE 455 50 KHN = 1 CORE 455 50 KHN = 10 CORE 455 55 401 RUCYCII) - 2.0 CORE 455 55 402 GO 10 402 CORE 455 55 410 X103 = NGTCII) - 2.0 CORE 455 55 410 X3 = SIGHAX(I) CORE 455 55 100 X11 = 1 CORE 55 57</td><td>KIN04 = IN01 KIN04 = IN01 KIN04 = I CORE KHAX = 0 CORE KHAX = 0 CORE KHAX = 1 CORE 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 402 KIN = 1 CORE 403 KIN = 1 CORE 404 KIN CORE 405 CORE CORE 406 CORE CORE 407 KIN = 1 CORE KHAX = 1</td><td>IND3 = IND1</td><td>CORE 4</td><td>+52</td><td></td><td></td></td<>	IN04 = IN01 IN04 = IN01 KHX = 1 CORE 453 KHX = 1 KHX = 1 KHX = 1 KHX = 1 KHX = 1 KHX = 1 KHX = 1 KROCC(I) .LE. 2.0) GO TO 401 F KOVC(I) .LE. 2.0) GO TO 401 CORE KNOVC(I) = RNOVC(I) - 1.0 CORE KNOVC(I) = RNOVC(I) - 2.0 CORE MODE CORE KIN = 1 CORE KNON = 1 CORE KNON = 1 CORE KNON = 1 CORE KIN = 1 C	TN04 = IN01 TN04 = IN01 KTN = 1 0 KHN = 1 0.010 402 KHN = 0 0.010 402 KHN = 0 0.010 402 KHN = 1 0.010 415 KHN = 1 0.010 4101 KHN = 1	TN04 = IN01 CORE 453 KTN = 1 0 CORE 454 KHN = 1 CORE 455 455 KHN = 1 CORE 455 455 KHN = 1 CORE 455 55 KHN = 1 CORE 455 55 KHN = 1 CORE 455 50 KHN = 1 CORE 455 50 KHN = 10 CORE 455 55 401 RUCYCII) - 2.0 CORE 455 55 402 GO 10 402 CORE 455 55 410 X103 = NGTCII) - 2.0 CORE 455 55 410 X3 = SIGHAX(I) CORE 455 55 100 X11 = 1 CORE 55 57	KIN04 = IN01 KIN04 = IN01 KIN04 = I CORE KHAX = 0 CORE KHAX = 0 CORE KHAX = 1 CORE 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 402 KIN = 1 CORE 403 KIN = 1 CORE 404 KIN CORE 405 CORE CORE 406 CORE CORE 407 KIN = 1 CORE KHAX = 1	IND3 = IND1	CORE 4	+52		
KHIN = 1 KHIN = 1 KHIN = 1 KHIN = 1 KHIN = 1 KHIN = 0 KIN = 0 CORE 401 KNCYC(I) = RNCYC(I) - 1.0 KIN = 0 CORE 401 KNCYC(I) = RNCYC(I) - 2.0 402 CORE 403 KNCYC(I) = RNCYC(I) - 2.0 404 KIN = 0 405 CORE 406 CORE 407 CORE 408 CORE 409 KIN = 0 400 CORE 401 KIN = 1 402 CORE 403 KHIN = 1 404 KIN = 1 405 CORE 406 CORE 407 KHIN = 1 408 CORE 409 CORE 410 CORE 411 CORE 412 CORE 413 CORE 414	KMIN = 1 CORE 454 KHA = 0 K1A = 0 K1A = 0 K1A = 0 K1A = 0 K1A = 0 K3A = 0 K3A = 0 K3A = 0 K3A = 0 K1 = 0 K3A = 0 K1 = 0 K3A = 0 K3A = 0 K1 = 0 K1 = 0 K1 = 0 K1 = 0 K1 = 0 CORE 453 K1 = 0 K1 = 0 CORE 454 401 KNCG(1) = KNCYG(1) - 2.0 CORE 454 402 K10 = 10 CORE 454 403 K10 = 10 CORE 453 404 K10 = 10 CORE 454 100 = 10 K10 = 10 CORE 454 100 = 1 K10 = 1 CORE 455 K11 = 1 K10 = 1 CORE	KMIN = 1 KMIN = 1 KMAX = 0 X34 = 0 X34 = 0 X34 = 0 X34 = 0 X34 = 0 K101 F(RUCC(I) = RUCYC(I) - 1.0 CORE 455 F01 RUCYC(I) = RUCYC(I) - 2.0 CORE 455 401 RUCYC(I) = RUCYC(I) - 2.0 CORE 455 401 RUCYC(I) = RUCYC(I) - 2.0 CORE 455 402 KIN0 = 0 CORE 455 403 SIGHAX(I) CORE 455 404 RUCYC(I) = RUCYC(I) - 2.0 CORE 455 405 COURE 455 CORE 455 406 COURE 455 CORE 455 407 CORE 455 CORE 455 408 CORE 455 CORE 455 409 SI RUL CORE 455 KMIN = 1 CORE 455 KMIN = 1 CORE 455 KIN0 = 1 CORE 455 KIN0 = 1 CORE 455 KIN0 = 1 CORE 457 KIN0 = 1 CORE 455 KIN0 = 1 CORE 477 KIN = 0<	KMIN = 1 KMIN = 1 KMAX = 0 K1 = 1 KMAX = 0 K1 = 0 K1 = 1 K1 = 0 K1 = 0 CORE	KHX = 1 XHX = 1 CORE 454 KHX = 1 X1 = 0 X1 = 0 455 K1 = 1 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 401 RUC(1) = RUCYC(1) - 2.0 X1 = 0 X1 = 0 402 RUC(1) = RUCYC(1) - 2.0 X1 = 0 X1 = 0 402 RUC(1) = RUCYC(1) - 2.0 X1 = 0 X1 = 0 402 RUC(1) = RUCYC(1) - 1.0 X2 = SIGHX(1) X2 = SIGHX(1) X1 = SIGHX(1) X2 = SIGHX(1) X2 = SIGHX(1) X1 = 0 X1 = 0 X1 = 0 X1 = 1 X1 = 0 X1 = 0 X1 = 1 X1 = 0 X1 = 0 X1 = 1 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1 = 0 X1	KHIN = 1 1 00RE 454 KHIN = 1 KHIN = 1 00RE 455 KAI = 0 KAI = 0 10 401 00RE 457 KYCYC(I) = RNCYC(I) = 1.0 CORE 456 50 00RE 457 ADD = 0 CORE KAI = 0 00RE 451 50 50 ADD = 0 CORE KAI = 0 00RE 456 50 50 50 ADD = 0 CORE KAI = 0 00RE 451 50 50 50 ADD = 1 CORE 455 50 50 50 50 50 ADD = 1 CORE 455 50	IND4 = INO1	CORE 4	+53		
KMAX = 0 KMAX = 0 KMAX = 0 K31 = 0 F(RWYC(I) .LE. 2.0) GO TO 401 CORE 455 F (RWYC(I) = RNCYC(I) = 1.0 GO TO 402 CORE 459 GO TO 402 CORE 459 CORE 459 401 RNCYC(I) = RNCYC(I) = 2.0 CORE 459 CORE 459 401 RNCYC(I) = RNCYC(I) = 2.0 CORE 459 CORE 459 402 KIND = 0 0 CORE 456 451 410 X = SIGHAX(I) CORE 456 451 103 = NSTEP(I) IND3 = NSTEP(I) CORE 453 103 = NSTEP(I) IND3 = NSTEP(I) CORE 454 100 + 1 IND4 = 1 CORE 454 100 + 1 CORE 456 466 100 + 1 IND4 = 1 CORE 455 100 + 1 IND4 = 1 CORE 456 100 + 1 IND4 = 1 CORE 456 100 + 1 IND4 = 1 CORE 456 110 + 1 IND4 = 1 CORE 456 KIND = 1 RIVC(I) = RNCY(I) - 1.0 COR	KHAX = 0 KHAX = 0 KHAX = 0 X:1 = 0 X:1 = 0 X:1 = 0 <tr< td=""><td>XHAX = 0 XHAX = 0 XHAX = 0 XHAX = 0 XHAX = 0 XHAX = 0 XHX = 0 XHX = 0</td><td>KHAX = 0 KHAX = 0 X1 = 0 X1 = 0 X2 = 0 X1 = 0 X2 = 0 X1 = 0 X2 = 0 X1 = 0 X1 = 0 X1 = 0 X2 = 0 X1 = 0 401 RUCC(1) = RUCC(1) - 2.0 X1 = 0 402 KIN0 = 0 X1 = 0 403 KIN0 = 0 X1 = 0 410 X2 = S1GHN(1) X1 = 0 X1 = 3 S1GHN(1) X1 = 3 S1GHN(1) X1 = 3 X1 = 1 X1 = 3 X1 = 1 X2 = S1GHN(1) X2 = S1GHN(1) X4 = 1 X2 = S1GHN(1) X4 = 1 X2 = S1GHN(1) X4 = 1 X2 = S1GH</td><td>KMAX = 0 KMAX = 0 CORE 455 IF (RNCYC(I) •LE. 2.0) GO TO 401 ENCYC(I) = RNCYC(I) - 1.0 CORE 455 GO TO 402 NCYC(I) = RNCYC(I) - 2.0 CORE 456 GO TO 402 SUCC(I) = RNCYC(I) - 2.0 CORE 456 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 456 402 KINO = 0 CORE 456 403 KINO = 0 CORE 456 410 SISTEP(I) CORE 455 100 = 1 KMIN = 1 CORE 455 100 = 1 KMIN = 1 CORE 456 100 = 1 KMIN = 1 CORE 455 100 = 1 KMIN = 1 CORE 456 100 = 1 KMIN = 1 CORE 456 100 = 1 KMIN = 1 CORE 456 101 = 0 CORE 7</td><td>KMX = 0 KMX = 0 KMX = 0 KMX = 0 KMX = 0 KMX = 0 F (RuCYC(I) • LE. 2.0) GO TO 401 F (RuCYC(I) = RuCYC(I) = 2.0 CORE 45 F (SUCYC(I) = RUCYC(I) = 2.0 CORE 45 CORE 45 401 RUCYC(I) = RUCYC(I) = 2.0 CORE 45 CORE 45 402 KINO = 0 CORE 45 CORE 45 403 KINO = 0 CORE 45 CORE 45 410 X3 = SIGMAK(I) X = SIGMAK(I) CORE 45 1003 E NISTEP(I) LO CORE 45 45 1004 = 103 KMN = 1 CORE 45 1004 = 103 KMN = 1 CORE 45 1004 = 1 KIN = 1 CORE 45 1005 E MIN = 1 KIN = 1 CORE 45 1008 = 1 KIN = 1 CORE 45 1010 = 1 RUCYC(I) = RUCYC(I) • 10.0 CORE 45 1010 = 1 RUCYC(I) = RUCYC(I) • 0.5 CORE 45 1010 = 0 CORE 45 45 45 10100 = 0 CUR = 47</td><td>KMIN = 1</td><td>CORE 4</td><td>+24</td><td></td><td></td></tr<>	XHAX = 0 XHAX = 0 XHAX = 0 XHAX = 0 XHAX = 0 XHAX = 0 XHX = 0 XHX = 0	KHAX = 0 KHAX = 0 X1 = 0 X1 = 0 X2 = 0 X1 = 0 X2 = 0 X1 = 0 X2 = 0 X1 = 0 X1 = 0 X1 = 0 X2 = 0 X1 = 0 401 RUCC(1) = RUCC(1) - 2.0 X1 = 0 402 KIN0 = 0 X1 = 0 403 KIN0 = 0 X1 = 0 410 X2 = S1GHN(1) X1 = 0 X1 = 3 S1GHN(1) X1 = 3 S1GHN(1) X1 = 3 X1 = 1 X1 = 3 X1 = 1 X2 = S1GHN(1) X2 = S1GHN(1) X4 = 1 X2 = S1GHN(1) X4 = 1 X2 = S1GHN(1) X4 = 1 X2 = S1GH	KMAX = 0 KMAX = 0 CORE 455 IF (RNCYC(I) •LE. 2.0) GO TO 401 ENCYC(I) = RNCYC(I) - 1.0 CORE 455 GO TO 402 NCYC(I) = RNCYC(I) - 2.0 CORE 456 GO TO 402 SUCC(I) = RNCYC(I) - 2.0 CORE 456 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 456 402 KINO = 0 CORE 456 403 KINO = 0 CORE 456 410 SISTEP(I) CORE 455 100 = 1 KMIN = 1 CORE 455 100 = 1 KMIN = 1 CORE 456 100 = 1 KMIN = 1 CORE 455 100 = 1 KMIN = 1 CORE 456 100 = 1 KMIN = 1 CORE 456 100 = 1 KMIN = 1 CORE 456 101 = 0 CORE 7	KMX = 0 KMX = 0 KMX = 0 KMX = 0 KMX = 0 KMX = 0 F (RuCYC(I) • LE. 2.0) GO TO 401 F (RuCYC(I) = RuCYC(I) = 2.0 CORE 45 F (SUCYC(I) = RUCYC(I) = 2.0 CORE 45 CORE 45 401 RUCYC(I) = RUCYC(I) = 2.0 CORE 45 CORE 45 402 KINO = 0 CORE 45 CORE 45 403 KINO = 0 CORE 45 CORE 45 410 X3 = SIGMAK(I) X = SIGMAK(I) CORE 45 1003 E NISTEP(I) LO CORE 45 45 1004 = 103 KMN = 1 CORE 45 1004 = 103 KMN = 1 CORE 45 1004 = 1 KIN = 1 CORE 45 1005 E MIN = 1 KIN = 1 CORE 45 1008 = 1 KIN = 1 CORE 45 1010 = 1 RUCYC(I) = RUCYC(I) • 10.0 CORE 45 1010 = 1 RUCYC(I) = RUCYC(I) • 0.5 CORE 45 1010 = 0 CORE 45 45 45 10100 = 0 CUR = 47	KMIN = 1	CORE 4	+24		
X31 = 0 X31 = 0 X31 = 0 X = 0 (C(1) = RNCYC(1) - 1.0 C0RE 45 R = 0 C0T0 +02 C0RE 45 R = 0 C0T0 +15 C0RE 45 R = 0 C0T0 +15 C0RE 45 R = 0 C0T0 +15 C0RE 45 R = 1 C0RE 45 46 R = 1 C0RE 45 46 R = 0 C0RE 45 45 R = 0 C0RE 45	X31 = 0 X31 = 0 X31 = 0 IF (RUCYC(I) .LE. 2.0) GO TO 401 CORE 456 RUCYC(I) = RUCYC(I) - 1.0 CORE 459 GO TO 402 CORE 459 KUCYC(I) = RUCYC(I) - 2.0 CORE 450 401 RUCYC(I) = RUCYC(I) - 2.0 CORE 450 402 CONE 450 CORE 453 403 CONE 450 CORE 453 404 RUCYC(I) = RUCYC(I) - 2.0 CORE 453 405 CONE 453 CORE 453 410 X3 = SIGMAX(I) CORE 453 410 X3 = SIGMAX(I) CORE 453 410 X3 = SIGMAX(I) CORE 454 X4 = SIGMILI) CORE 455 CORE 455 X104 = 1 RUCYC(I) = 4.0 CORE 455 X110 = 1 RUCYC(I) = 1.0 CORE 455 KIND = 1 RUCYC(I) = 1.0 CORE 472 K10 = 1 RUCYC(I) = 1.0 CORE 455 K10 = 1 RUCYC(I) = 1.0 CORE 473 K10 = 1 RUCYC(I) = 1.0 CORE 473 K10 = 1 RUCYC(I) = 1.0 CORE 473 K10 = 20 RUCYC(I) + 1.0 CORE 473	X31 = 0 X31 = 0 X31 = 0 IF (RKUYC(II) .LE. 2.0) GO TO 401 ENCYC(II) = RUNTG(II) = 1.0 CORE 455 AUD RUCYC(II) = RUNTG(II) = 2.0 GO TO 402 459 CORE 450 401 RUCYC(II) = RUNTG(II) = 2.0 GO TO 415 CORE 450 451 401 RUCYC(II) = RUNTG(II) = 2.0 GO TO 415 CORE 453 403 RUNTG(II) = 2.0 CORE 453 CORE 453 410 X = 3 SIGMAX(I) CORE 455 453 410 X = 1 XHIN = 1 CORE 455 453 XHIN = 1 RUN4 = 103 CORE 455 455 XHIN = 1 RUN4 = 103 CORE 456 456 XHIN = 1 RUN4 = 103 CORE 455 456 XHIN = 1 RUN4 = 103 CORE 456 456 XHIN = 1 RUN4 = 103 CORE 456 456 XHIN = 1 RUN4 = 103 CORE 456 456 XHIN = 1 RUN4 = 103 CORE 456 456 XHIN = 1 RU10 CORE 456	If SAUCYC(II)Le. 2.0) GO TO 401 CORE 455 If CANCYC(II) = RNCYC(II) = 1.0 CORE 455 AUCYC(II) = RNCYC(II) = 2.0 CONE 455 AUCYC(II) = RNCYC(II) = 2.0 CONE 455 AUCYC(II) = RNCYC(II) = 2.0 CONE 455 401 RNCYC(II) = RNCYC(II) = 2.0 CONE 456 402 KIND = 0 CONE 456 403 KIND = 0 CONE 453 404 KIND = 0 CONE 453 405 CONE 453 CONE 453 400 KIND = 1 CONE 455 CONE 454 403 KIND = 1 CONE 455 CONE 455 KHAX = 0 KIND = 1 CONE 455 CONE 455 KHAX = 0 KIND = 1 CONE 457 CONE 457 KHAX = 0 KIND = 1 CONE 473 CONE 477 KHAX = 0 KIND = 1 CONE CONE 477 CONE 477 KHAX = 0 KIND = 1 CONE CONE	K131 = 0 CORE 456 K131 = 0 Very (CI) = RNGYC(I) - 1.0 CORE 455 K101 = COVC(I) = RNGYC(I) - 1.0 CORE 451 K101 = COVC(I) = RNGYC(I) - 2.0 CORE 451 K101 = COVC(I) = RNGYC(I) - 2.0 CORE 451 K101 = COVC(I) = RNGYC(I) - 2.0 CORE 451 401 RNGYC(I) = RNGYC(I) - 2.0 CORE 451 402 K100 = 0 CORE 452 403 K101 = CORE CORE 453 103 = SIGHN(I) CORE 454 103 = SIGHN(I) CORE 454 103 = SIGHN(I) CORE 454 KMA = 1 CORE 455 KMA = 1 CORE	X131 = 0 CORE 455 X151 = 1:0 X10 X10 X11 X151 = 1:0 X10 X10 X10 X151 = X10 X10 X10 X10 X10 X151 = X10 X10 X10 X10 X10 Y10 X10 X10 X10 X10 X10 Y11 Y10 Y10 Y10 Y10 Y10 Y11 Y10 Y10 Y10 Y10 Y10 Y11 Y10 Y10 Y10 Y10 Y10 Y11 Y10 </td <td>XEAX # 0</td> <td>CORE</td> <td>+55</td> <td></td> <td></td>	XEAX # 0	CORE	+55		
IF (RNCYC(I) .LE. 2.0) GO TO 401 GORE 457 RUCYC(I) = RNCYC(I) - 1.0 OORE 459 6 GO TO 412 CORE 451 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 451 402 KIN0 = 0 CORE 451 402 KIN0 = 0 CORE 451 402 KIN0 = 0 CORE 451 403 KIN0 = 0 CORE 451 404 KIN1 CORE 453 410 X = SIGMANI) CORE 454 410 X = SIGMANI) CORE 454 1004 = IND3 NSTEP(I) CORE 454 1003 = NSTEP(I) IND3 = NSTEP(I) CORE 454 1004 = I CORE 454 454 KMIN = 1 CORE 456 456 KMIN = 1 CORE 456 471 <td>IF (RNCYC(I) .LE. 2.0) GO 10 401 CORE 457 NGYC(I) = RNCYC(II) - 1.0 CORE 459 401 RNCYC(I) = RNCYC(II) - 2.0 CORE 459 402 KIN0 = 0 CORE 451 403 KIN0 = 0 CORE 451 404 RNCYC(I) = RNCYC(II) - 2.0 CORE 451 405 KIN0 = 0 CORE 453 408 KIN0 = 0 CORE 453 410 X3 = SIGMX(I) CORE 454 1003 = NSIEP(I) CORE 455 1004 KIN = 1 CORE 455 1010 = 1 RIN = 1 CORE 455 1010 = 1 RINCYC(I) = 1.0 CORE 455 1010 KIN = 1 CORE 456 77 1101 KINCYC(I) = NON CORE 77 CORE 1101 KINCYC(I) = NON<td>IF (RNCYC(I) .LE. 2.0) GO TO 401 CORE 457 RNCYC(I) = RNCYC(I) - 1.0 CORE 459 60 TO 415 CORE 451 60 TO 415 CORE 451 410 RNCYC(I) = RNCYC(I) - 2.0 CORE 451 402 KIN0 = 0 CORE 451 402 KIN0 = 0 CORE 451 415 KIN0 = 0 CORE 453 416 KIN0 = 1 CORE 453 410 X = SIGMAX(I) CORE 453 1003 = NSTEP(I) CORE 454 1004 = IND3 CORE 455 1004 = IND3 CORE 455 1004 = IND3 CORE 456 1001 = I CORE 456 1101 = RNCYC(I) - 1.0 CORE 471 1101 = RNCYC(I) - 2.3,X4,KEY,I,CYCYGEN) CORE 472 1101 ± CORE 473 CORE 476 110 ± 1000 TO</td><td>IF (RNOTC(I)</td><td>Ricyc(II) = Let and Constant and Consta</td><td>AT CONTRUMT CORE 457 AT CONTRUMT CORE 459 AT CONTRUMT CORE 459 AT CONTRUMT CORE 450 AT CONTRACT CORE 450 AT A</td><td>X31 = 0</td><td>CORE 4</td><td>• 36</td><td></td><td></td></td>	IF (RNCYC(I) .LE. 2.0) GO 10 401 CORE 457 NGYC(I) = RNCYC(II) - 1.0 CORE 459 401 RNCYC(I) = RNCYC(II) - 2.0 CORE 459 402 KIN0 = 0 CORE 451 403 KIN0 = 0 CORE 451 404 RNCYC(I) = RNCYC(II) - 2.0 CORE 451 405 KIN0 = 0 CORE 453 408 KIN0 = 0 CORE 453 410 X3 = SIGMX(I) CORE 454 1003 = NSIEP(I) CORE 455 1004 KIN = 1 CORE 455 1010 = 1 RIN = 1 CORE 455 1010 = 1 RINCYC(I) = 1.0 CORE 455 1010 KIN = 1 CORE 456 77 1101 KINCYC(I) = NON CORE 77 CORE 1101 KINCYC(I) = NON <td>IF (RNCYC(I) .LE. 2.0) GO TO 401 CORE 457 RNCYC(I) = RNCYC(I) - 1.0 CORE 459 60 TO 415 CORE 451 60 TO 415 CORE 451 410 RNCYC(I) = RNCYC(I) - 2.0 CORE 451 402 KIN0 = 0 CORE 451 402 KIN0 = 0 CORE 451 415 KIN0 = 0 CORE 453 416 KIN0 = 1 CORE 453 410 X = SIGMAX(I) CORE 453 1003 = NSTEP(I) CORE 454 1004 = IND3 CORE 455 1004 = IND3 CORE 455 1004 = IND3 CORE 456 1001 = I CORE 456 1101 = RNCYC(I) - 1.0 CORE 471 1101 = RNCYC(I) - 2.3,X4,KEY,I,CYCYGEN) CORE 472 1101 ± CORE 473 CORE 476 110 ± 1000 TO</td> <td>IF (RNOTC(I)</td> <td>Ricyc(II) = Let and Constant and Consta</td> <td>AT CONTRUMT CORE 457 AT CONTRUMT CORE 459 AT CONTRUMT CORE 459 AT CONTRUMT CORE 450 AT CONTRACT CORE 450 AT A</td> <td>X31 = 0</td> <td>CORE 4</td> <td>• 36</td> <td></td> <td></td>	IF (RNCYC(I) .LE. 2.0) GO TO 401 CORE 457 RNCYC(I) = RNCYC(I) - 1.0 CORE 459 60 TO 415 CORE 451 60 TO 415 CORE 451 410 RNCYC(I) = RNCYC(I) - 2.0 CORE 451 402 KIN0 = 0 CORE 451 402 KIN0 = 0 CORE 451 415 KIN0 = 0 CORE 453 416 KIN0 = 1 CORE 453 410 X = SIGMAX(I) CORE 453 1003 = NSTEP(I) CORE 454 1004 = IND3 CORE 455 1004 = IND3 CORE 455 1004 = IND3 CORE 456 1001 = I CORE 456 1101 = RNCYC(I) - 1.0 CORE 471 1101 = RNCYC(I) - 2.3,X4,KEY,I,CYCYGEN) CORE 472 1101 ± CORE 473 CORE 476 110 ± 1000 TO	IF (RNOTC(I)	Ricyc(II) = Let and Constant and Consta	AT CONTRUMT CORE 457 AT CONTRUMT CORE 459 AT CONTRUMT CORE 459 AT CONTRUMT CORE 450 AT CONTRACT CORE 450 AT A	X31 = 0	CORE 4	• 36		
RNCYC(I) = RNCYC(I) - 1.0 CORE 45 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 45 402 RNO CORE 45 60 T0 415 CORE 45 410 SIGMAX(I) CORE 45 1003 NSTEP(I) CORE 45 1003 NSTEP(I) CORE 45 1003 NSTEP(I) CORE 45 1003 SISTEP(I) CORE 45 1003 SISTEP(I) CORE 45 10103 SISTEP(I) CORE 45 10103 SISTEP(I) CORE 46 1103 SISTEP(I) CORE 45 1103 SISTEP(I) CORE 45 1103 SISTEP(I) CORE 47 1103 SISTEP(I) CORE 47	RNCYC(I) = RNCYC(I) - 1.0 CORE 458 401 KINO = 0 CORE 459 402 KINO = 0 CORE 451 402 KINO = 0 CORE 451 403 KINO = 0 CORE 451 404 KINO = 0 CORE 451 400 KINO = 0 CORE 451 400 KINO = 0 CORE 453 410 KINO = 1 CORE 455 KMIN = 1 KMIN = 1 CORE 455 KMIN = 1 KINO = 1 CORE 455 KINO = 1 CORE 455 470 KINO = 1 CORE 470 CORE 470 KINO = 1 CORE 470 CORE 470 KONO = AINT(RNCVC(I) + 1.0 CORE 470 CORE 470 KINO = 0 CONO + KCYGEN) CORE 470 COR	RNCYC(I) = RNCYC(I) - 1.0 CORE 458 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 459 402 KIN0 = 0 CORE 451 402 KIN0 = 0 CORE 451 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 451 402 KIN0 = 0 CORE 451 60 T0 415 CORE 452 60 T0 415 CORE 452 60 T0 415 CORE 454 103 SIGMAX(I) CORE 454 103 SIGMAX(I) CORE 455 1003 SIGMAX(I) CORE 455 1004 IND3 SIGMAX(I) CORE 456 KIN1 1 CORE 456 456 KIN1 1 RIN0 CORE 456 KIN1 1 CORE 456 77 <td>RNCYC(I) = RNCYC(I) - 1.0 CORE 458 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 451 402 KINU = 0 CORE 451 402 KINU = 0 CORE 451 402 KINU = 0 CORE 451 60 T0 415 CORE 453 410 X3 = S16MN(I) CORE 453 KNIN = 1 CORE 453 CORE NU04 = IN03 NIN4 = 10 CORE 454 KINN = 1 CORE 455 466 KINN = 1 CORE 455 471 KINN = 1 RINCYC(I) = RNCYC(I) - 1.0 CORE 472 KIN = 0 A11 (RNCYC(I) + 0.5) CORE 473 KIN = 10 CORE 473 CORE 473 KIN = 1000 CORE K100 CORE 473 <t< td=""><td>kncyc(I) = RNCYC(I) = I.0 CORE 458 401 KNCYC(I) = RNCYC(I) - 2.0 CORE 459 60 T0 402 CORE 451 402 KIND = 0 CORE 451 402 KIND = 0 CORE 453 60 T0 415 CORE 452 410 X3 = SIGMAK(I) CORE 454 103 = NSTEP(I) IND4 = IND3 CORE 454 104 = IND3 KNIN = 1 CORE 455 1003 = NSTEP(I) IND4 = IND3 CORE 456 1010 = 1 KHIN = 1 CORE 456 KHIN = 1 KHIN = 0 CORE 456 KHIN = 1 KHIN = 0 CORE 456</td><td>RNCYC(I) = RNCYC(I) = 1.0 CORE 458 60 T0 415 CORE 451 410 KIND = 0 CORE 451 60 T0 415 CORE 451 410 KIND = 0 CORE 451 60 T0 415 CORE 451 60 T0 415 CORE 452 410 KIND = 0 CORE 453 60 T0 415 CORE 454 60 T0 415 CORE 454 60 T0 415 CORE 455 1003 = NSTEP(I) CORE 455 1001 = 1 RULUE CORE 456 KHAX = 0 KIN = 1 CORE 456 KIN = 1 KIN = 1 CORE 456 KHAX = 0 KIN = 1 CORE 473 KIN = 1 CORE 473 CORE KIN = 0 CO</td><td>IF (RNCYC(I) .LE. 2.0) GO TO 401</td><td>CORE 4</td><td>+57</td><td>•</td><td></td></t<></td>	RNCYC(I) = RNCYC(I) - 1.0 CORE 458 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 451 402 KINU = 0 CORE 451 402 KINU = 0 CORE 451 402 KINU = 0 CORE 451 60 T0 415 CORE 453 410 X3 = S16MN(I) CORE 453 KNIN = 1 CORE 453 CORE NU04 = IN03 NIN4 = 10 CORE 454 KINN = 1 CORE 455 466 KINN = 1 CORE 455 471 KINN = 1 RINCYC(I) = RNCYC(I) - 1.0 CORE 472 KIN = 0 A11 (RNCYC(I) + 0.5) CORE 473 KIN = 10 CORE 473 CORE 473 KIN = 1000 CORE K100 CORE 473 <t< td=""><td>kncyc(I) = RNCYC(I) = I.0 CORE 458 401 KNCYC(I) = RNCYC(I) - 2.0 CORE 459 60 T0 402 CORE 451 402 KIND = 0 CORE 451 402 KIND = 0 CORE 453 60 T0 415 CORE 452 410 X3 = SIGMAK(I) CORE 454 103 = NSTEP(I) IND4 = IND3 CORE 454 104 = IND3 KNIN = 1 CORE 455 1003 = NSTEP(I) IND4 = IND3 CORE 456 1010 = 1 KHIN = 1 CORE 456 KHIN = 1 KHIN = 0 CORE 456 KHIN = 1 KHIN = 0 CORE 456</td><td>RNCYC(I) = RNCYC(I) = 1.0 CORE 458 60 T0 415 CORE 451 410 KIND = 0 CORE 451 60 T0 415 CORE 451 410 KIND = 0 CORE 451 60 T0 415 CORE 451 60 T0 415 CORE 452 410 KIND = 0 CORE 453 60 T0 415 CORE 454 60 T0 415 CORE 454 60 T0 415 CORE 455 1003 = NSTEP(I) CORE 455 1001 = 1 RULUE CORE 456 KHAX = 0 KIN = 1 CORE 456 KIN = 1 KIN = 1 CORE 456 KHAX = 0 KIN = 1 CORE 473 KIN = 1 CORE 473 CORE KIN = 0 CO</td><td>IF (RNCYC(I) .LE. 2.0) GO TO 401</td><td>CORE 4</td><td>+57</td><td>•</td><td></td></t<>	kncyc(I) = RNCYC(I) = I.0 CORE 458 401 KNCYC(I) = RNCYC(I) - 2.0 CORE 459 60 T0 402 CORE 451 402 KIND = 0 CORE 451 402 KIND = 0 CORE 453 60 T0 415 CORE 452 410 X3 = SIGMAK(I) CORE 454 103 = NSTEP(I) IND4 = IND3 CORE 454 104 = IND3 KNIN = 1 CORE 455 1003 = NSTEP(I) IND4 = IND3 CORE 456 1010 = 1 KHIN = 1 CORE 456 KHIN = 1 KHIN = 0 CORE 456 KHIN = 1 KHIN = 0 CORE 456	RNCYC(I) = RNCYC(I) = 1.0 CORE 458 60 T0 415 CORE 451 410 KIND = 0 CORE 451 60 T0 415 CORE 451 410 KIND = 0 CORE 451 60 T0 415 CORE 451 60 T0 415 CORE 452 410 KIND = 0 CORE 453 60 T0 415 CORE 454 60 T0 415 CORE 454 60 T0 415 CORE 455 1003 = NSTEP(I) CORE 455 1001 = 1 RULUE CORE 456 KHAX = 0 KIN = 1 CORE 456 KIN = 1 KIN = 1 CORE 456 KHAX = 0 KIN = 1 CORE 473 KIN = 1 CORE 473 CORE KIN = 0 CO	IF (RNCYC(I) .LE. 2.0) GO TO 401	CORE 4	+57	•	
401 COTO 402 CORE 459 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 450 601 010 10 CORE 450 610 100 0 CORE 451 610 10 10 CORE 451 610 100 CORE 453 611 X4 = SIGHAK(I) CORE 453 1004 IND4 = IND3 CORE 455 10104 IND4 = IND3 CORE 455 1010 IND4 = IND3 CORE 455 1010 IND4 = IND3 CORE 455 110 RICVCII) - 1.0 CORE 473 111 CORE 473 CORE 473 111 CALON INININ CORE 473	401 GO TO 402 459 401 RNCYG(I) = RNCYG(I) - 2.0 450 402 GO 415 00KE 403 SIGMAX(I) 00KE 410 X3 = SIGMAX(I) 00KE 103 = NSTEP(I) 100 00KE 103 = NSTEP(I) 00KE 455 103 = NSTEP(I) 00KE 455 103 = NSTEP(I) 00KE 455 103 = NSTEP(I) 1 00KE 455 103 = NSTEP(I) 1 00KE 455 103 = NSTEP(I) 1 00KE 455 104 = 1 1 00KE 455 104 = 1 1 00KE 455 110 = 1 1 00KE 455 110 = 1 1 00KE 471 11 00KE 473 00KE 12 00RE 473 00RE 475 13 00RE 00KE	401 G0 T0 402 CORE 459 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 451 402 KIND = 0 CORE 451 410 X3 = SIGMAX(I) CORE 453 410 X3 = SIGMAX(I) CORE 454 1003 = NSTEP(I) CORE 454 55 1003 = NSTEP(I) CORE 454 1004 = IND3 CORE 455 1004 = IND3 CORE 455 1004 = IND3 CORE 456 1004 = IND3 CORE 457 1004 = IND3 CORE 456 1004 = IND3 CORE 457 100 = I RNCYC(I) = RNCYC(I) - 1.0 CORE 477 11 CYCNO = AINT(RNCYC(I) + 1.0 CORE 477 11 CYCNO = AINT(RNCYC(I) + 1.0 CORE 477 12 CYCNO = AINT(RNCYC(I) + 1.0 CORE 477 1415 CYCNO = AINT(RNCYCI) + 1.0 CORE 477 15 CALL DECIDE(X1, X2, X3, X4, KEY, J1, CYCNO, KCYGEN) CORE 477 10 CORE	401 G0 T0 402 459 401 RNCYC(I) = RNCYC(I) - 2.0 CORE 450 402 G0 T0 445 CORE 451 410 X3 = SIGMAX(I) CORE 453 410 X3 = SIGMAX(I) CORE 453 410 X4 = SIGMAX(I) CORE 454 X4 = SIGMAX(I) CORE 454 X4 = SIGMAX(I) CORE 455 X4 = SIGMAX(I) CORE 454 X4 = SIGMAX(I) CORE 455 X4 = SIGMAX(I) CORE 455 X4 = SIGMAX(I) CORE 455 X410 = 1 CORE 455 X110 = 1 CORE 455 X11 = 0 CORE 457 X11 = 1 CORE 457 X10 = 1 RIUCYC(I) = RUCYC(I) - 1.0 CORE X11 = 0 CORE 473 CORE A14 CORE 473 CORE A14 CORE 473 CORE A14 CORE 474 CORE A14 CORE 474	G0 T0 402 40 KNCVC(I) = RNCVC(I) = RNCVC(I) = RNCVC(I) = RNCVC(I) = 2.0 45 KNCVC(I) = RNCVC(I) = 2.0 40 K KNO = 0 00 K KNCVC(I) = 2.0 00 K K 45 I 41 KNCVC(I) = 0 00 K K 45 I 00 K K 45 I 41 KNCVC(I) = 10 K KNL 00 K K 45 I 00 K K 45 I 10 K KNL = 1 00 K K 45 I 00 K K 45 I 10 K KNL = 1 00 K K 45 I 00 K K 45 I 10 K KND = 1 00 K K 45 I 00 K K 45 I 11 K R 1 0 00 K K 45 I 00 K K 45 I 12 K KNN = 1 00 K K 45 I 00 K K 45 I 13 K K K I I 00 K K 45 I 00 K K 45 I 14 K S 0 10 00 I 00 K K 47 I 00 K K 47 I 14 K S 0 10 00 I 00 K K 47 I 00 K K 47 I	401 G0 T0 402 450 401 RNCYC(I) = RNCYC(I) - 2.0 00RE 451 402 G0 T0 415 00RE 451 410 G0 T0 415 00RE 453 410 X3 = SIGMA(I) 00RE 454 X4 = SIGMIN(I) 00RE 454 454 X103 = NSTEP(I) 00RE 454 456 XMIN = 1 00RE 467 60RE 455 XMIN = 1 0 00RE 455 60RE 457 XMIN = 1 0 00RE 456 477 60RE 457 XMIN = 1 0 00RE 457 60RE 457 60RE 457 XMIN = 1 0 00RE 473 60RE 473 60RE 473 XHIN = 1 0 00RE 473 60RE 473 60RE 473 XHI = 0 0 00RE 473 60RE 473 60RE 473 XHI = 0 0 00RE 473 60RE 473 60RE 473	RNCYC(I) = RNCYC(I) - 1.0	CORE 4	+ 5 8		
401 RNCYC(I) = RNCYC(I) = 2.0 401 FNCYC(I) = 2.0 402 KINO = 0 0 451 60 T0 415 60 T0 415 451 60 T0 415 00RE 451 61 X3 = SIGMAX(I) 00RE 455 103 SNTEP(I) 00RE 455 104 IND3 = NSTEP(I) 00RE 455 104 IND4 = IND3 00RE 455 104 IND5 00RE 456 104 IND6 00RE 456 104 IND6 00RE 455 104 IND6 00RE 456 10 X1N 0 00RE 456 110 IRCVC(I) 4.0 00RE 456 111 RICVC(I) 4.0 00RE 456 111 RICVC(I) 1.0 00RE 456 111 RICVC(I) 1.0 00RE 456 111 RICVC(I) 1.0 00RE 457 111 RICVC(I) 1.0 00RE <td>401 RNCYG(I) = RNCYG(I) - 2.0 401 FNCYG(I) = 2.0 402 KINO = 0 0 451 60 T0 415 00RE 451 61 T0 415 00RE 455 61 T0 410 00RE 455 103 NNTEP(I) 00RE 455 103 NNT = 1 00RE 455 104 XIND = 1 00RE 456 110 1 0 00RE 456 111 1 00RE 456 471 111 1 0 00RE 456 111 1 0 00RE 456 111 1 0 00RE 456 111 1 0 00RE 471 111 1 0 00RE 471 111 1 0 00RE 472 111</td> <td>401 RNCYC(I) = RNCYC(I) - 2.0 401 FNCYC(I) = RNCYC(I) - 2.0 402 KIN0 = 0 0 60 451 60 10 415 00RE 453 410 X3 = SIGMIN(I) 00RE 453 410 X3 = SIGMIN(I) 00RE 454 1003 = NSTEP(I) 00RE 454 1003 = NSTEP(I) 00RE 455 1003 = NSTEP(I) 00RE 455 1003 = NSTEP(I) 00RE 456 1003 = NSTEP(I) 00RE 455 1003 = NSTEP(I) 00RE 455 1000 = X31 = 0 00RE 456 1100 = 1 RNCYC(I) = RNCYC(I) - 1.0 00RE 472 110 = 1 RNCYC(I) = RNCYC(I) - 1.0 00RE 472 111 CYCNO-KCYPEN) 00RE 472 111 RNCYC(I) = 1.0 00RE 472 111 CYCNO-KCYPEN) 00RE 473 112 CYCNO-KCYPEN) 00RE 472 113 CYCNO-KCYPEN) 00RE 472 114 CYCNO-KCYPEN)<td>401 RNCYC(I) = RNCYC(I) = 2.0 401 ENCYC(I) = RNCYC(I) = 2.0 402 KIN0 = 0 0 E0 751 60 T0 412 X3 = SIGMAX(I) 00RE 451 410 X3 = SIGMAX(I) 00RE 453 x4 = SIGMAX(I) E00RE 454 1ND3 = NSTEP(I) E00RE 454 1ND4 = IND3 E00RE 455 1ND4 = IND3 E00RE 456 1ND4 = IND3 E00RE 456 1ND4 = IND3 E00RE 456 1ND4 = IND4 IND5 E00RE 456 1ND4 = I E00RE 456 470 1ND4 = I E00RE 456 470 1ND4 = I E00RE 470 E0RE 471 1 E00RE 473 E0RE 473 1 E01D010 E0100 E0RE 475 50 T0008 E0RE 473 E0RE 41 E0018 E0018 E0RE 473 50 T0008 E0RE 475 E0RE 475</td><td>401 RNCYC(I) = RNCYC(I) = 2.0 602 451 402 XIN0 = 0 0 607 451 60 T0 415 00R 451 608 451 10 X3 SIGMAX(I) 0 00R 453 10 X3 SIGMAX(I) 00R 453 10 X3 SIGMAX(I) 00R 454 103 NIN1+ IND3 NIN1+ 10 103 NSTEP(I) 00R 455 00R 455 104 X103 1 00R 456 456 104 X31 = 0 0 00R 456 456 110 1 0 00R 471 00R 471 110 1 0 00R 471 00R 472 11 X10 1 0 00R 473 00R 473 11 X10 1 0 00R 473 00R 473 11 X10 00R 1 00R 473 00R</td><td>401 RNCYC(I) = RNCYC(I) = 2.0 CORE 451 402 KIND = 0 CORE 451 402 KIND = 10 CORE 453 410 X3 = SIGMAX(I) CORE 453 110 X3 = SIGMAX(I) CORE 454 110 X3 = SIGMAX(I) CORE 454 110 X3 = SIGMAX(I) CORE 454 11033 NSTEP(I) CORE 455 11033 NSTEP(I) CORE 455 11033 SIGMAX(I) CORE 456 1103 SIGMAX CORE 456 1103 SIGMAX CORE 457 1103 SIGMAX CORE 456 1103 SIGMAX CORE 457 1103 SIGMAX CORE 456 1103 SIGMAX SIGMAX 10 1103 SIGMAX CORE 477 111 SIGMAX SIGMAX 477 111 SIGMAX SIGMAX 477 111 SIGMAX SIGMAX</td><td>GO TO 402</td><td>CORE</td><td>624</td><td></td><td></td></td>	401 RNCYG(I) = RNCYG(I) - 2.0 401 FNCYG(I) = 2.0 402 KINO = 0 0 451 60 T0 415 00RE 451 61 T0 415 00RE 455 61 T0 410 00RE 455 103 NNTEP(I) 00RE 455 103 NNT = 1 00RE 455 104 XIND = 1 00RE 456 110 1 0 00RE 456 111 1 00RE 456 471 111 1 0 00RE 456 111 1 0 00RE 456 111 1 0 00RE 456 111 1 0 00RE 471 111 1 0 00RE 471 111 1 0 00RE 472 111	401 RNCYC(I) = RNCYC(I) - 2.0 401 FNCYC(I) = RNCYC(I) - 2.0 402 KIN0 = 0 0 60 451 60 10 415 00RE 453 410 X3 = SIGMIN(I) 00RE 453 410 X3 = SIGMIN(I) 00RE 454 1003 = NSTEP(I) 00RE 454 1003 = NSTEP(I) 00RE 455 1003 = NSTEP(I) 00RE 455 1003 = NSTEP(I) 00RE 456 1003 = NSTEP(I) 00RE 455 1003 = NSTEP(I) 00RE 455 1000 = X31 = 0 00RE 456 1100 = 1 RNCYC(I) = RNCYC(I) - 1.0 00RE 472 110 = 1 RNCYC(I) = RNCYC(I) - 1.0 00RE 472 111 CYCNO-KCYPEN) 00RE 472 111 RNCYC(I) = 1.0 00RE 472 111 CYCNO-KCYPEN) 00RE 473 112 CYCNO-KCYPEN) 00RE 472 113 CYCNO-KCYPEN) 00RE 472 114 CYCNO-KCYPEN) <td>401 RNCYC(I) = RNCYC(I) = 2.0 401 ENCYC(I) = RNCYC(I) = 2.0 402 KIN0 = 0 0 E0 751 60 T0 412 X3 = SIGMAX(I) 00RE 451 410 X3 = SIGMAX(I) 00RE 453 x4 = SIGMAX(I) E00RE 454 1ND3 = NSTEP(I) E00RE 454 1ND4 = IND3 E00RE 455 1ND4 = IND3 E00RE 456 1ND4 = IND3 E00RE 456 1ND4 = IND3 E00RE 456 1ND4 = IND4 IND5 E00RE 456 1ND4 = I E00RE 456 470 1ND4 = I E00RE 456 470 1ND4 = I E00RE 470 E0RE 471 1 E00RE 473 E0RE 473 1 E01D010 E0100 E0RE 475 50 T0008 E0RE 473 E0RE 41 E0018 E0018 E0RE 473 50 T0008 E0RE 475 E0RE 475</td> <td>401 RNCYC(I) = RNCYC(I) = 2.0 602 451 402 XIN0 = 0 0 607 451 60 T0 415 00R 451 608 451 10 X3 SIGMAX(I) 0 00R 453 10 X3 SIGMAX(I) 00R 453 10 X3 SIGMAX(I) 00R 454 103 NIN1+ IND3 NIN1+ 10 103 NSTEP(I) 00R 455 00R 455 104 X103 1 00R 456 456 104 X31 = 0 0 00R 456 456 110 1 0 00R 471 00R 471 110 1 0 00R 471 00R 472 11 X10 1 0 00R 473 00R 473 11 X10 1 0 00R 473 00R 473 11 X10 00R 1 00R 473 00R</td> <td>401 RNCYC(I) = RNCYC(I) = 2.0 CORE 451 402 KIND = 0 CORE 451 402 KIND = 10 CORE 453 410 X3 = SIGMAX(I) CORE 453 110 X3 = SIGMAX(I) CORE 454 110 X3 = SIGMAX(I) CORE 454 110 X3 = SIGMAX(I) CORE 454 11033 NSTEP(I) CORE 455 11033 NSTEP(I) CORE 455 11033 SIGMAX(I) CORE 456 1103 SIGMAX CORE 456 1103 SIGMAX CORE 457 1103 SIGMAX CORE 456 1103 SIGMAX CORE 457 1103 SIGMAX CORE 456 1103 SIGMAX SIGMAX 10 1103 SIGMAX CORE 477 111 SIGMAX SIGMAX 477 111 SIGMAX SIGMAX 477 111 SIGMAX SIGMAX</td> <td>GO TO 402</td> <td>CORE</td> <td>624</td> <td></td> <td></td>	401 RNCYC(I) = RNCYC(I) = 2.0 401 ENCYC(I) = RNCYC(I) = 2.0 402 KIN0 = 0 0 E0 751 60 T0 412 X3 = SIGMAX(I) 00RE 451 410 X3 = SIGMAX(I) 00RE 453 x4 = SIGMAX(I) E00RE 454 1ND3 = NSTEP(I) E00RE 454 1ND4 = IND3 E00RE 455 1ND4 = IND3 E00RE 456 1ND4 = IND3 E00RE 456 1ND4 = IND3 E00RE 456 1ND4 = IND4 IND5 E00RE 456 1ND4 = I E00RE 456 470 1ND4 = I E00RE 456 470 1ND4 = I E00RE 470 E0RE 471 1 E00RE 473 E0RE 473 1 E01D010 E0100 E0RE 475 50 T0008 E0RE 473 E0RE 41 E0018 E0018 E0RE 473 50 T0008 E0RE 475 E0RE 475	401 RNCYC(I) = RNCYC(I) = 2.0 602 451 402 XIN0 = 0 0 607 451 60 T0 415 00R 451 608 451 10 X3 SIGMAX(I) 0 00R 453 10 X3 SIGMAX(I) 00R 453 10 X3 SIGMAX(I) 00R 454 103 NIN1+ IND3 NIN1+ 10 103 NSTEP(I) 00R 455 00R 455 104 X103 1 00R 456 456 104 X31 = 0 0 00R 456 456 110 1 0 00R 471 00R 471 110 1 0 00R 471 00R 472 11 X10 1 0 00R 473 00R 473 11 X10 1 0 00R 473 00R 473 11 X10 00R 1 00R 473 00R	401 RNCYC(I) = RNCYC(I) = 2.0 CORE 451 402 KIND = 0 CORE 451 402 KIND = 10 CORE 453 410 X3 = SIGMAX(I) CORE 453 110 X3 = SIGMAX(I) CORE 454 110 X3 = SIGMAX(I) CORE 454 110 X3 = SIGMAX(I) CORE 454 11033 NSTEP(I) CORE 455 11033 NSTEP(I) CORE 455 11033 SIGMAX(I) CORE 456 1103 SIGMAX CORE 456 1103 SIGMAX CORE 457 1103 SIGMAX CORE 456 1103 SIGMAX CORE 457 1103 SIGMAX CORE 456 1103 SIGMAX SIGMAX 10 1103 SIGMAX CORE 477 111 SIGMAX SIGMAX 477 111 SIGMAX SIGMAX 477 111 SIGMAX SIGMAX	GO TO 402	CORE	624		
402 KIN0 = 0 0 00RE 451 60 T0 415 00RE 453 10 3 = SIGMIN(I) 00RE 453 10 4 15 00RE 453 10 3 = NSTEP(I) 00RE 453 10 4 1 NO3 00RE 454 10 3 = NSTEP(I) 00RE 454 10 4 1 NO3 00RE 455 10 4 1 NO3 00RE 456 10 1 KMN = 1 0 00RE 456 11 1 1 0 0 00RE 457 11 1 1 1 1 0 00RE 473 11 1 1 1 1 1 1 1 0 00RE 473 11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	402 KIND = 0 0 415 451 60 T0 415 00RE 452 453 10 3 = SIGMAK(I) 00RE 453 453 10 4 15 00RE 455 453 10 4 15 00RE 455 455 10 4 10 3 = NSIGMAK(I) 00RE 455 455 10 4 11 00RE 455 455 10 4 11 00RE 455 455 10 10 1 1 1.0 00RE 455 10 10 1 1 1.0 00RE 455 10 10 1 1 1.0 00RE 457 11 1 1 1 1 00RE 471 457 12 1 1 1 00RE 471 457 13 1 1 00RE 471 457 14 1 1 00RE 477 471 15 10 100 1.0 00RE 477 16 0 10 100 00RE 477 474 17 1 10 00RE 477 474 17	402 KIN0 = 0 0 45 60 T0 415 60 T0 415 45 60 T0 415 0028 453 45 1003 = NSTEP(1) 0088 453 45 1004 = NSTEP(1) 0088 455 45 1004 = NSTEP(1) 0088 456 45 1004 = NSTEP(1) 0088 457 45 1004 = NSTEP(1) 0088 457 45 1008 KIN0 = 1 0088 457 45 1108 KIN0 = 1 0088 473 47 1108 KIN0 = 1 0088 473 47 1108 KIN0 = 1 0088 473 47 1108 KIN0 = 1000 0088 47 47 111 KB = 0 0088 47 607 0088 47 112 KB = 0 0088 47 6088 47 6088 47 111 KB = 0 0088 47 6088 47 6088 47 111 KB = 0 0088 47 6088 47 6088 47 111 KB = 0 0088 47 6088 47 6088 47 111 KB	402 KIN0 = 0 0 CORE 452 60 T0 415 60 T0 415 455 10 X = SIGMIN(I) CORE 454 10 X = SIGMIN(I) CORE 455 100 = SIGMIN(I) CORE 455 100 = NO3 NSTEP(I) CORE 455 100 = SIGMIN(I) CORE 455 60 100 = SIGMIN(I) CORE 455 60 100 = SIGMIN(I) CORE 455 60 100 = SIGMIN(I) CORE 455 467 100 = SININ CORE 455 471 110 = CORE 450 CORE 473 110 = CORE 473 CORE 473 110 = CORE (XIN) = SININ CORE 473 110 = CORE 473 CORE 473 110 = CORE (XIN) = SININ CORE 473 110 = CORE (XIN) = SININ CORE 473 111 DECIDE((X1) X2) X3)X4) KEV, I, I, CYCNO, KCYGEN CORE 473 111 DECIDE((X1) X2) X3)X4) KEV, I, I, CYCNO, KCYGEN CORE 473 111 CONE	402 KIND = 0 008E 451 60 T0 415 008E 453 60 T0 415 008E 453 1003 = NSTEP(1) 008E 455 1003 = NSTEP(1) 008E 455 1003 = NSTEP(1) 008E 455 1004 = TN03 008E 455 KMIN = 1 008E 455 KMIN = 1 008E 455 KMIN = 1 008E 457 KMIN = 1 008E 457 KMN = 1 0 008E 471 KMN = 1 0 008E 471 KIND = 1 0 008E 471 KIND = 1 0 008E 473 KIND = 1 0 008E 473 CYCNO = AINT(RNCYC(I) + 1.0 008E 473 CALL DECIDE(X11,X2)X3)X4, KEY, I, CYCNO, KCYGEN) 008E 475 G0 TO 1000 0100 008E 475 G0RE 475 008E 475 G0RE 475 008E 475 G0RE 10000 0008E 475 <td>402 KIN0 = 0 0 415 451 410 x1 = SIGMAX(I) 002 KIN0 = 0 453 x1 = SIGMAX(I) 002 KIN(I) 453 1003 = NSTEP(I) 1003 = NSTEP(I) 002 KIN1 = 455 1004 = IND3 002 KIN1 = 1 002 KIN1 = 455 1014 = IND3 002 KIN1 = 1 002 KIN1 = 455 1014 = IND3 002 KIN1 = 1 002 KIN1 = 455 1010 K11 = RNCVC(I) = 1.0 002 KIN1 = 472 002 KIN1 = 472 KIN0 = 1 002 KIN1 = 1 002 KIN1 = 472 KIN0 = 1 002 KIN1 = 1 002 KIN1 = 472 K1N0 = 1 002 KIN1 = 1.0 002 KIN1 = 472 K1N0 = 1 002 KIN1 = 472 002 KIN1 = 472 K1N0 = 1 002 KIN1 = 472 002 KIN1 = 472 K1N0 = 1 002 KIN1 = 472 002 KIN1 = 472 K1N0 = 1 002 KIN1 = 472 002 KIN1 = 472 K1N0 = 1 000 KIN1 = 472 000 KIN1 = 472 K1N0 = 1 000 KIN1 = 472 000 KIN1 = 472 K1N0 = 1 000 KIN1 = 472 000 KIN1 = 472 K1N0 = 1000 000 KIN1 + 100 KIN1 + 10</td> <td>401 RNCYC(I) = RNCYC(I) - 2.0</td> <td>CORF 4</td> <td></td> <td>•</td> <td>•</td>	402 KIN0 = 0 0 415 451 410 x1 = SIGMAX(I) 002 KIN0 = 0 453 x1 = SIGMAX(I) 002 KIN(I) 453 1003 = NSTEP(I) 1003 = NSTEP(I) 002 KIN1 = 455 1004 = IND3 002 KIN1 = 1 002 KIN1 = 455 1014 = IND3 002 KIN1 = 1 002 KIN1 = 455 1014 = IND3 002 KIN1 = 1 002 KIN1 = 455 1010 K11 = RNCVC(I) = 1.0 002 KIN1 = 472 002 KIN1 = 472 KIN0 = 1 002 KIN1 = 1 002 KIN1 = 472 KIN0 = 1 002 KIN1 = 1 002 KIN1 = 472 K1N0 = 1 002 KIN1 = 1.0 002 KIN1 = 472 K1N0 = 1 002 KIN1 = 472 002 KIN1 = 472 K1N0 = 1 002 KIN1 = 472 002 KIN1 = 472 K1N0 = 1 002 KIN1 = 472 002 KIN1 = 472 K1N0 = 1 002 KIN1 = 472 002 KIN1 = 472 K1N0 = 1 000 KIN1 = 472 000 KIN1 = 472 K1N0 = 1 000 KIN1 = 472 000 KIN1 = 472 K1N0 = 1 000 KIN1 = 472 000 KIN1 = 472 K1N0 = 1000 000 KIN1 + 100 KIN1 + 10	401 RNCYC(I) = RNCYC(I) - 2.0	CORF 4		•	•
410 60 0 415 410 x3 = SIGMAX(I) 60 x4 = SIGMAX(I) 00 454 x4 = SIGMAX(I) 00 60 x4 = SIGMAX(I) 00 60 x4 = SIGMAX(I) 00 60 x4 = SIGMAX(I) 60 454 x4 = NID3 00 60 464 x41N = 1 0 60 66 x101 = 1 0 60 66 x101 = 1 0 60 66 x101 x101 0 60 66 x101 x101 0 00 60 x101 x101 0 00 60 473 x101 0 00 00 473 60 473 x11 0 00 00 473 60 60 473 x11 0 00 00 00 473 60 60 473 x11 0 </td <td>410 X3 = SIGMAX(I) X4 = SIGMAX(I) X4 = SIGMAX(I) X4 = SIGMAX(I) X6 = SIGMAX(I) X03 = NSTEP(I) X03 = NSTEP(I) IND3 = NSTEP(I) CORE 454 X1 = 0 CORE 455 X1 = 1 CORE 467 X1 = 1 CORE 455 X1 = 1 CORE 455 X1 = 1 CORE 456 X1 = 1 CORE 456 X1 = 1 CORE 456 X1 = 1 CORE 470 X1 = 2 CYCN0, KCYGEN) CORE 470 X1 = 2 CYCN0, KCYGEN) CORE 470 X1 = 0 CORE 470 CORE 475 X1 = 0 CORE 470 CORE 475 X1 = 0 CORE 475 CORE 475 X1 =</td> <td>410 50 T0 415 410 x3 SIGMAX(I) 53 x4 SIGMAX(I) 000000000000000000000000000000000000</td> <td>410 x3 sIGMAX(I) x4 sIGMAX(I) 0000 x41 0000 0000 x41 0000 0000 x11 0000 0000 x11 0000 0000 x11 00000 0000 x11 0000 0000 x11 00000 00000 x11 00</td> <td>410 50 70 415 452 410 x3 = SIGHAX(I) 53 50000 453 x4 = SIGHAX(I) 00000 453 50000 454 x03 = NSTEP(I) 00000 454 455 50000 454 x103 = NSTEP(I) 00000 60000 455 60000 456 456 x11N = 1 0 00000 60000 60000 60000 457 x11N = 1 0 00000 60000 60000 60000 457 x131 = 0 00000 00000 477 00000 60000 477 x131 = 0 00000 00000 477 00000 477 x101 0000 00000 00000 477 00000 477 x11 0 00000 00000 477 00000 477 x11 0 00000 00000 477 00000 00000 x11 0 00000 00000 477 0000</td> <td>410 415 000000000000000000000000000000000000</td> <td></td> <td>TOPF</td> <td></td> <td></td> <td></td>	410 X3 = SIGMAX(I) X4 = SIGMAX(I) X4 = SIGMAX(I) X4 = SIGMAX(I) X6 = SIGMAX(I) X03 = NSTEP(I) X03 = NSTEP(I) IND3 = NSTEP(I) CORE 454 X1 = 0 CORE 455 X1 = 1 CORE 467 X1 = 1 CORE 455 X1 = 1 CORE 455 X1 = 1 CORE 456 X1 = 1 CORE 456 X1 = 1 CORE 456 X1 = 1 CORE 470 X1 = 2 CYCN0, KCYGEN) CORE 470 X1 = 2 CYCN0, KCYGEN) CORE 470 X1 = 0 CORE 470 CORE 475 X1 = 0 CORE 470 CORE 475 X1 = 0 CORE 475 CORE 475 X1 =	410 50 T0 415 410 x3 SIGMAX(I) 53 x4 SIGMAX(I) 000000000000000000000000000000000000	410 x3 sIGMAX(I) x4 sIGMAX(I) 0000 x41 0000 0000 x41 0000 0000 x11 0000 0000 x11 0000 0000 x11 00000 0000 x11 0000 0000 x11 00000 00000 x11 00	410 50 70 415 452 410 x3 = SIGHAX(I) 53 50000 453 x4 = SIGHAX(I) 00000 453 50000 454 x03 = NSTEP(I) 00000 454 455 50000 454 x103 = NSTEP(I) 00000 60000 455 60000 456 456 x11N = 1 0 00000 60000 60000 60000 457 x11N = 1 0 00000 60000 60000 60000 457 x131 = 0 00000 00000 477 00000 60000 477 x131 = 0 00000 00000 477 00000 477 x101 0000 00000 00000 477 00000 477 x11 0 00000 00000 477 00000 477 x11 0 00000 00000 477 00000 00000 x11 0 00000 00000 477 0000	410 415 000000000000000000000000000000000000		TOPF			
410 X3 = SIGMAX(I) 410 X3 = SIGMAX(I) X4 = SIGMIN(I) 00RE 1ND3 = NSTEP(I) 00RE 1ND4 = IND3 00RE XM1 = 1 00RE XM2 = 0 00RE XM1 = 1 00RE <t< td=""><td>410 X3 = SIGMAX(I) 410 X3 = SIGMAX(I) X4 = SIGMIN(I) 00RE IND3 = NSTEP(I) 00RE IND4 = IND3 00RE KM1 = 1 00RE KMA = 1 00RE KMA = 1 00RE K1 = 0 00RE K2 = 0</td><td>410 X3 = SIGMAX(I) 000000000000000000000000000000000000</td><td>410 X3 = SIGMAX(I) 410 X3 = SIGMAX(I) X4 = SIGMIN(I) 00RE 454 IND3 = NSTEP(I) 00RE 454 IND4 = IND3 00RE 457 KMIN = 1 00RE 467 KMIN = 1 00RE 468 KINO = 1 00RE 468 KINO = 1 00RE 459 KINO = 1 00RE 470 KINO = 1 00RE 473 CYON = AINT(RNCYC(I) = 1.0 00RE 473 CONE 473 00RE 473 CONE 473 00RE 473 CONE 475 00RE 473 CONE 473 00RE 473 CONE 473 00RE 473 CONE 473 00RE 475 CONE 473 00RE 475 CONE 473 00RE 475 CONE 475 00RE</td><td>410 X3 = SIGMX(I) 000000000000000000000000000000000000</td><td>410 X3 = SIGMX(I) 410 X3 = SIGMX(I) X4 = SIGMIN(I) 0002 IND3 = NSTEP(I) 0002 IND3 = NSTEP(I) 0002 IND3 = NSTEP(I) 0002 KM1 = 1 0002 KMA = 1 0002 KMA = 1 0002 KMA = 1 0002 KIND = 10 0002 KIND = 10</td><td></td><td>CORF</td><td></td><td></td><td></td></t<>	410 X3 = SIGMAX(I) 410 X3 = SIGMAX(I) X4 = SIGMIN(I) 00RE IND3 = NSTEP(I) 00RE IND4 = IND3 00RE KM1 = 1 00RE KMA = 1 00RE KMA = 1 00RE K1 = 0 00RE K2 = 0	410 X3 = SIGMAX(I) 000000000000000000000000000000000000	410 X3 = SIGMAX(I) 410 X3 = SIGMAX(I) X4 = SIGMIN(I) 00RE 454 IND3 = NSTEP(I) 00RE 454 IND4 = IND3 00RE 457 KMIN = 1 00RE 467 KMIN = 1 00RE 468 KINO = 1 00RE 468 KINO = 1 00RE 459 KINO = 1 00RE 470 KINO = 1 00RE 473 CYON = AINT(RNCYC(I) = 1.0 00RE 473 CONE 473 00RE 473 CONE 473 00RE 473 CONE 475 00RE 473 CONE 473 00RE 473 CONE 473 00RE 473 CONE 473 00RE 475 CONE 473 00RE 475 CONE 473 00RE 475 CONE 475 00RE	410 X3 = SIGMX(I) 000000000000000000000000000000000000	410 X3 = SIGMX(I) 410 X3 = SIGMX(I) X4 = SIGMIN(I) 0002 IND3 = NSTEP(I) 0002 IND3 = NSTEP(I) 0002 IND3 = NSTEP(I) 0002 KM1 = 1 0002 KMA = 1 0002 KMA = 1 0002 KMA = 1 0002 KIND = 10 0002 KIND = 10		CORF			
44 = SIGHN(I) IND3 = NSTEP(I) IND4 = IND3 IND4 = IND3 KMIN = 1 KIND = 1 <	44 = SIGHIN(1) 1N03 = NSTEP(1) 1N03 = NSTEP(1) 1N03 = NSTEP(1) 1N03 = NSTEP(1) 1N04 = 1 NU1 = 1 KMAX = 0 KMAX = 1 KMAX = 0 KIND = 1 RUND = 1 RUNCF(1) = RUCY(1) = 1.0 CORE KIND = 1 RUCYC(1) = RUCY(1) = 1.0 CORE 459 CORE 471 CORE 473 CORE 474 CORE 475	415 CORE 454 XMD4 = IND3 NSTEP(I) CORE 455 IND3 = NSTEP(I) CORE 455 IND4 = IND3 CORE 455 KMIN = 1 CORE 455 KMIN = 1 CORE 455 KMIN = 1 CORE 455 KIN = 1 CORE 456 KIN = 1 CORE 470 KIN = 1 CORE 472 KIN = 1 CORE 472 KB = 0 CORE 472 CALL DECIDE(X1,X2,X3,X4,KEY,I).CYCN0,KCYGEN) CORE 473 G0 T0 1000 CORE 475 CORE	44 = SIGHIN(I) 000000000000000000000000000000000000	41 516H1(1) 0000 454 1ND3 = NSTEP(1) 1ND3 = NSTEP(1) 00000 455 1ND3 = NSTEP(1) 1000 00000 455 1ND1 = 1 00000 00000 455 1ND = 1 1.0 00000 00000 1ND = 1 1.0 00000 00000 1ND = 1 1.0 00000 000000 1ND = 1 1.0 00000 000000 1ND = 1 1.0 000000 000000 1ND = 1 1.0 0000000000 000000000000000000000000000000000000	x4 = SIGHN(I) 000000000000000000000000000000000000	$L \uparrow \uparrow \uparrow \chi = \Sigma I G M \Delta \chi (I)$				1
IND3 = NULENT 400 IND4 = IND3 100 + E IND3 IND4 = IND3 0 KMIN = 1 0 KMAX = 0 00RE K31 = 0 00RE K13 = 0 00RE K14 = 0 00RE K15 = CORE 458 K16 = 0 00RE K17 = 0 00RE K18 = 0 00RE K19 = 0 00RE K10 = AINT(RNCYC(1) - 1.0 00RE K15 = CYCN0 = AINT(RNCYC(1) + 0.5) 00RE C15 = 0 00RE K15 = 0 00RE K16 = 0 00RE K17 = 0 00RE K18 = 0 00RE K19 = 0 00RE K10 = 00RE 473 CORE 473 CORE 473 CORE 473 CORE 474 CORE 473 CORE 474 CORE 473 CORE 474 CORE 474 CORE 475	IND3 = NSTEPTID TOTE	IND3 = NSTEP(I) 0000 IND4 = IND3 00000 KMIN = 1 00000 KMAX = 0 00000 KMAX = 0 00000 KIND = 1 0 KB = 0 0 CVCN0 = AINT(RNCVC(I) + 1.0 KB = 0 0 CVCN0 = AINT(RNCVC(I) + 0.5) CORE 473 CORE 475 CORE 473 CORE 473 CORE 473 CORE 474 CORE 475 CORE 475	IND3 = NSTEP(I) VOL VOL VOL IND4 = IND3 IND4 = IND3 VOL VOL VOL KMIN = 1 KMIN = 1 CORE V65 V66 KMIN = 1 KMIN = 1 CORE V66 V67 KMIN = 1 KMIN = 1 CORE V66 V66 KIND = 1 CORE V68 V70 V71 KIND = 1 CORE V10 CORE V71 KIND = 1 CORE V71 CORE V71 KIND = 1000 CON CORE V71 <td>IND3 = NSTEPTION 000000000000000000000000000000000000</td> <td>IND3 = NSTERION 000000000000000000000000000000000000</td> <td></td> <td></td> <td></td> <td></td> <td></td>	IND3 = NSTEPTION 000000000000000000000000000000000000	IND3 = NSTERION 000000000000000000000000000000000000					
IND4 = IND3 IND4 = IND3 IND4 = IND3 E00E KMIN = 1 E00E KMAX = 0 E00E K31 = 0 E00E K31 = 0 E00E K10 = 1 E00E K11 = 1 E00E K12 EVENCE(I) = 1.0 CVEN0 = AINT(RNCYC(I) + 0.5) E00E K15 EVEN0 C00E 473	1004 = IND3 455 1004 = IND3 456 KMIN = 1 0 KMIN = 1 0 KIN0 = 1 00RE KIN0 = 1000 00RE K0 = 1000 00RE	1004 = TNDS 405 TND4 = TNDS 000000000000000000000000000000000000	1N04 = IN03 00xE 450 NM1N = 1 0 00xE 467 KM1N = 1 0 00xE 467 KM1N = 1 0 00xE 457 KM1N = 1 0 00xE 457 KM1N = 1 0 00xE 457 K1N = 1 0 00xE 470 K1N = 1 0 00xE 471 K1NC(1) = RNCYC(1) - 1.0 00xE 472 K1NC(1) = RNCYC(1) - 1.0 00xE 473 K1NC(1) = CUDE(X1).X2,X3,X4,KEY,I,CYON,KCYGEN) 00xE 473 G0 T0 1000 1000 00xE 473	IND4 = IND3 CORE 450 IND4 = 1 CORE 467 KMIN = 1 CORE 467 KMAX = 0 CORE 467 KMD = 1 CORE 459 KID = 1 CORE 470 KID = 1 CORE 470 KB = 0 CORE 471 KB = 0 CORE 471 KB = 0 CORE 472 CYCNO = AINT(RNCYG(I) + 0.5) CORE 472 G0 TO 1000 CORE 473 G0 TO 1000 CORE 475	1ND4 = IND3 455 NND4 = IND3 000000000000000000000000000000000000	71 - 010118177 7102 - Notodity		† u		
KNIN = 1 CORE 467 KMAX = 0 CORE 467 KMAX = 0 CORE 459 K10 = 1 CORE 459 K10 = 1 CORE 459 K10 = 1 CORE 470 K10 = 20 CORE 471 K10 = AINT(RNCYCII) - 1.0 CORE 473 C11 DECIDE(X1)X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 473 C11 DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 473	#107 1000 KMIN = 1 00000 KMAX = 0 0000000 KIND = 1 000000000 KIND = 1 000000000000 KIND = 1 000000000000000000000000000000000000	KHIN = 1 COCC 467 KHIN = 1 COCC 467 KHIN = 1 COCC 467 KIND = 1 CORE 467 KIND = 1 CORE 459 KIND = 1 CORE 459 KIND = 1 CORE 459 KINCYCII) = RNCYCII) = 1.0 CORE 472 KB = 0 CORE 472 CYCN0 = AINT(RNCYCII) + 0.5) CORE 472 CORE 472 CORE 473 Colore 473 CORE 473 Colore 473 CORE 475 CORE 473 CORE 475 CORE 473 CORE 475 CORE 473 CORE 475 CORE 475 CORE 475	KHIN = 1 CORE 467 KHIN = 1 CORE 467 KHAX = 0 K100 = 1 CORE 459 KIN0 = 1 CORE 459 470 KIN0 = 1 CORE 470 CORE 471 KIN0 = 1 CYCN0 = AINT (RNCYG(I) + 0.5) CORE 473 473 C1 CYCN0 = AINT (RNCYG(I) + 0.5) CORE 473 CORE 473 G0 T0 1000 CORE 473 CORE 475 CORE 475	#101 = 1 CORE 467 KMAX = 0 CORE 467 KMAX = 0 CORE 459 K110 = 1 CORE 459 K100 = 1 RNCYC(1) = 1.0 CORE 470 K101 = 1 CORE 471 CORE 471 K10 = 1 CORE 471 CORE 473 K1 = 0 CORE 473 CORE 473 K2 CVON0 = AINT(RNCYC(1)+0.5) CORE 473 C0 TO 1000 CORE 473 CORE 473 C0 TO 1000 CORE 475 CORE 475 C0 TO 1000 CORE 475 CORE 475	#107 #100 #00 KHAX = 1 0 KHAX = 1 KHAX = 1 0 K10 #5 KIND = 1 0 K10 1 0 KIND = 1 0 K10 1 0 KIND = 1 0 CORE #5 KIND = 1 0 CORE #7 KIND = 1 CORE #7 0 KIND = 1 RUCYC(I) = RUCYC(I) = 1.0 CORE #7 KB = 0 CONE #7 CORE #7 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCN0,KCYGEN) CORE #75 CORE #75 G0 T0 1000 T0 1000 CORE #75 CORE #75	1400 - 1401CT141				1
A11 - 1 CORE 46 K31 = 0 K31 = 0 CORE 46 K1N0 = 1 CORE 459 470 K1N0 = 1 CORE 470 CORE 470 K1N0 = 1 CORE 470 CORE 470 K1N0 = 1 CORE 470 CORE 470 K1 = 0 CORE 470 CORE 473 K1 = 0 CORE 473 CORE 473 CALL DECIDE(X1) *0.5) CORE 473 CORE G01 DORE 473 CORE 473	MAX = 1 CORE 46 K31 = 0 KIND = 1 CORE 45 KIND = 1 CORE 45 KIND = 1 CORE 45 KINCYCII) = RNCYCII) - 1.0 CORE 47 KICCCI ENCYCII) = CORE 47 KICCO CORE 47 CALL DECIDE(X1, X2, X3, X4, KEY, I, CYCNO, KCYGEN) CORE 475 GO 10 1000 CORE 475 CORE 475	KMAX = 1 CORE 468 K31 = 0 CORE 458 K1ND = 1 CORE 459 K1ND = 1 CORE 470 K1 CCULI = RNCYC(I) - 1.0 CORE 472 K1 CCUL DECIDE(X1,X2,X3,X4,KEY,I).CYCN0,KCYGEN) CORE 473 G0 T0 1000 CORE 475 CORE	KMAX = 1 CORE 46 KMAX = 1 CORE 46 KIND = 1 CORE 459 KIND = 1 CORE 459 KIND = 1 CORE 470 RHCYC(I) = RNCYC(I) - 1.0 CORE 470 KIND = 1 CORE 471 RHCYC(I) = RNCYC(I) + 0.5) CORE 471 CYCNO = AINT(RNCYC(I) + 0.5) CORE 473 CALL DECIDE(X1)X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 473 G0 TO 1000 CORE 475 CORE	KMAX = 1 CORE 46 KAX = 1 CORE 46 KIND = 1 CORE 459 KIND = 1 CORE 470 KIND = 1 CORE 470 KIND = 1 CORE 471 KIND = 1 CORE 471 KIND = 1 CORE 471 KIND = 0 CORE 471 KIND = 0 CORE 472 KIND = 0 CORE 473 CYCNO = AINT(RNCYC(I)+0.5) CORE 473 CALL DECIDE(X1)X2,X3,X4,KEY,I,CYGEN) CORE 473 G0 T0 1000 CORE 475 CORE 475	KMAX = 1 CORE 46 KAX = 1 CORE 46 KIND = 1 CORE 459 KIND = 1 CORE 470 KIND = 1 CORE 471 KINCYCII) = RNCYCII) - 1.0 CORE 471 KB = 0 AINT(RNCYCII) + 0.5) CORE 472 CALL DECIDE(X1,X2,X3,X4,KEY,I).CYCN0,KCYGEN) CORE 475 G0 T0 1000 CORE 475 CORE 475					
X31 = 0 VOKE 400 X31 = 0 X1N0 = 1 450 X1N0 = 1 X1N1 X1N1 X1N1 = 1 <td< td=""><td>X11 = 0 VORE 450 X11 = 1 CORE 459 X10 = 1 CORE 470 X10 = 1 CORE 471 X10 = 1 CORE 471 X10 = 0 AIIT(RNCYC(I) + 0.5) CORE 472 C4LL DECIDE(X1),X2,X3,X4,KEY,I,CYCN0,KCYGEN) CORE 475 G0 T0 1000 CORE 475 CORE 475</td><td>KIN = 1 UCKE 450 KIN = 1 CORE 459 KIC (I) = RNCYC(I) = 1.0 CORE 470 KICYC(I) = RNCYC(I) = 1.0 CORE 470 KB = 0 CORE 471 KB = 0 CORE 472 CYCNO = AINT(RNCYC(I) + 0.5) CORE 472 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCN0,KCYGEN) CORE 473 G0 TO 1000 CORE 475</td><td>XII = 0 VOLC VOLC VOLC VOLC XIN0 = 1 1 CORE VI VI XIN0 = 1 0 CORE VI VI XIN0 = 2 CULL DECIDE(XI)+0.5) CORE VI VI XIN0 = 2 CULL DECIDE(XI)+0.5) CORE VI VI XIN0 = 1000 CORE VI VI CORE VI XIN0 = 1000 CORE VI CORE VI VI</td><td>K31 = 0 UCKE 450 K31 = 0 KIND = 1 CORE 450 K1ND = 1 KIND = 1 CORE 470 K1S = 0 CORE 471 CORE 471 K1 = 0 CORE 472 CORE 472 K1 = 0 CORE 472 CORE 472 K1 = 0 CORE 473 CORE 473 K1 = 0 CORE 473 CORE 473 G0 TO 1000 CORE 475 CORE 475 G0 TO 1000 CORE 475 CORE 475</td><td>X31 = 0 VORE 450 X31 = 0 X1N = 1 CORE 450 X1N = 1 X1N = 1 CORE 470 X1N = 1 CORE 470 CORE 471 X1N = 1 CORE 471 CORE 471 X1N = 1 CORE 471 CORE 472 X1N = 1 CORE 472 CORE 473 X1N = 1 CORE 473 CORE 473 X1N = 1010 CORE 475 CORE 475 X1 CORE 475 CORE 475</td><td></td><td></td><td></td><td></td><td></td></td<>	X11 = 0 VORE 450 X11 = 1 CORE 459 X10 = 1 CORE 470 X10 = 1 CORE 471 X10 = 1 CORE 471 X10 = 0 AIIT(RNCYC(I) + 0.5) CORE 472 C4LL DECIDE(X1),X2,X3,X4,KEY,I,CYCN0,KCYGEN) CORE 475 G0 T0 1000 CORE 475 CORE 475	KIN = 1 UCKE 450 KIN = 1 CORE 459 KIC (I) = RNCYC(I) = 1.0 CORE 470 KICYC(I) = RNCYC(I) = 1.0 CORE 470 KB = 0 CORE 471 KB = 0 CORE 472 CYCNO = AINT(RNCYC(I) + 0.5) CORE 472 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCN0,KCYGEN) CORE 473 G0 TO 1000 CORE 475	XII = 0 VOLC VOLC VOLC VOLC XIN0 = 1 1 CORE VI VI XIN0 = 1 0 CORE VI VI XIN0 = 2 CULL DECIDE(XI)+0.5) CORE VI VI XIN0 = 2 CULL DECIDE(XI)+0.5) CORE VI VI XIN0 = 1000 CORE VI VI CORE VI XIN0 = 1000 CORE VI CORE VI VI	K31 = 0 UCKE 450 K31 = 0 KIND = 1 CORE 450 K1ND = 1 KIND = 1 CORE 470 K1S = 0 CORE 471 CORE 471 K1 = 0 CORE 472 CORE 472 K1 = 0 CORE 472 CORE 472 K1 = 0 CORE 473 CORE 473 K1 = 0 CORE 473 CORE 473 G0 TO 1000 CORE 475 CORE 475 G0 TO 1000 CORE 475 CORE 475	X31 = 0 VORE 450 X31 = 0 X1N = 1 CORE 450 X1N = 1 X1N = 1 CORE 470 X1N = 1 CORE 470 CORE 471 X1N = 1 CORE 471 CORE 471 X1N = 1 CORE 471 CORE 472 X1N = 1 CORE 472 CORE 473 X1N = 1 CORE 473 CORE 473 X1N = 1010 CORE 475 CORE 475 X1 CORE 475 CORE 475					
KIND = U UCKE 459 KINCYC(I) = RNCYC(I) = 1.0 CORE 470 RNCYC(I) = RNCYC(I) = 1.0 CORE 471 KB = 0 CORE 471 CYCN0 = AINT(RNCYC(I) + 0.5) CORE 472 CALL DECIDE(X1, X2, X3, X4, KEY, I, CYCN0, KCYGEN) CORE 475 GGI TO 10100 CORE 475	K131 - U CUCKE 459 K100 = 1 K10 K1 K100 = 1 CORE 470 K10 = 0 CORE 471 K1 = 0 CORE 471 K1 = 0 CORE 472 C41 CCN0 = AINT(RNCYC(1)+0.5) COCN0,KCYGEN) CORE 473 C01 C010E(X1,X2,X3,X4,KEY,I,CYON,KCYGEN) CORE 475 475 G0 T0 1000 CORE 475 CORE 475	KIND 1 UCKE 4.39 KIND 1 0 CORE 4.70 RHCYC(I) RNCYC(I) 4.0 CORE 4.71 KB 0 CORE 4.71 CORE 4.71 KB 0 CORE 4.71 CORE 4.72 KB 0 CORE 4.72 CORE 4.73 KB 0 CORE 4.73 CORE 4.74 KB 0 CORE 4.75 CORE 4.75 KB 0 CORE 4.75 CORE 4.75 KB CORE 4.75 CORE 4.75 KB CORE 4.75 CORE 4.75 KB CORE 4.75 CORE 4.75	K101 = 1 CORE 4.9 K107 = 1 R1CYC(I) = RNCYC(I) - 1.0 CORE 4.7 KB = 0 CORE 4.7 CORE 4.7 KB = 10.0	KIND = U UCKE 459 KIND = U UCKE 459 RUCYC(I) = RNCYC(I) - 1.0 CORE 471 KB = 0 CORE 471 KB = 0 CORE 471 CYCN0 = AINT(RNCYC(I)+0.5) CORE 472 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCN0,KCYGEN) CORE 473 G0 T0 1000 CORE 475	KINT U			00		i
KINU = 1 RHCYC(I) = RNCYC(I) - 1.0 KH = 0 KH = 0 CORE 471 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCN0,KCYGEN) CORE 473 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCN0,KCYGEN) CORE 473 CORE 473 CORE 473 CORE 473	KINU = 1 RHCYC(I) = RNCYC(I) - 1.0 KB = 0 CORE 471 CORE 472 CORE 473 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) G0 T0 1000 CORE 473 CORE 473 CORE 473 CORE 473 CORE 473 CORE 473	KINU = 1 RUCC(I) = RNCYC(I) - 1.0 KB = 0 KB = 0 CORE 471 CORE 472 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCN0,KCYGEN) CORE 473 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCN0,KCYGEN) CORE 475 CORE 475 CORE 475 CORE 475	KINU = 1 Rucyc(I) = RNCYC(I) - 1.0 KB = 0 CYCNO = AINT(RNCYC(I)+0.5) CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) G0 TO 1000 CORE 473 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 473 CORE 473 CORE 473 CORE 473 CORE 473	KINU = 1 RICC(I) = RNCYC(I) - 1.0 KB = 0 KB = 0 CORE 471 CORE 472 CYCNO = AINT(RNCYC(I)+0.5) CYCNO = AINT(RNCYC(I)+0.5) CORE 473 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 473 CORE 473 CORE 473 CORE 475 CORE 475 CORE 475 CORE 475 CORE 475 CORE 475 CORE 475 CORE 475 CORE 475 CORE 475	KINU = 1 RHCYC(I) = RNCYC(I) - 1.0 KB = 00RE 471 CORE 472 CORE 472 CORE 472 CORE 473 Call Decide(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) G0 T0 1000 CORE 475 CORE 475 CORE 475 CORE 475 CORE 475 CORE 475 CORE 475	K31 = U	CORE 4	5.9		
KALTULL = KNUTULL = 1.U KB = 0 CORE 472 K15 CYCNO = AINT(RNCYC(I)+0.5) CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 473 CORE 473 CORE 475 CORE 475 CORE 475	KHLTCLU = KNUTCLU = 1.0 CORE 4/1 KB = 0 CORE 472 415 CYCNO = AINT(RNCYG(I)+0.5) CORE 473 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 473 G0 T0 1000 CORE 475	KHUTULI = KWUULI - 1.U KB = 0 CORE 472 KB = 0 CORE 472 CORE 472 COLL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 473 COLL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 475 GO TO 1000 CORE 475	#15 CVR00 = 1.0 CORE 4/1 #15 CYR00 = 1NT(RNCYG(I)+0.5) CORE 4/3 G0 T0 1000 CORE 4/7 G0 T0 1000 CORE 4/7	KHUTULU = KWUULU = 1.0 KB = 0 CORE 472 KB = 0 CORE 472 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 474 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 475 G0 T0 1000 CORE 475	KHUTULU = KNUTULU = 1.U KB = 0.00K +71 COR = AINT(RNCYG(I)+0.5) COR = 473 CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) GO TO 1000 GO TO 1000 CORE +75 CORE +75			1.10		
KU = U CORE 472 415 CYCNO = AINT(RNCYC(I)+0.5) Call Dēcide(X1,X2,X3,X4,KEY)I,CYCNO,KCYGEN) Gait Dācide(X1,X2,X3,X4,KEY)I,CYCNO,KCYGEN) Gait Dana Core 474	KB = U CORE 472 415 CYCNO = AINT(RNCYG(I)+0.5) CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) G0 T0 1000 Core 473 474 Core 475	KB = U CORE 472 415 CYCNO = AINT(RNCYC(I)+0.5) CYCNO,KCYGEN) CORE 473 Call Dēcide(X1,X2,X3,X4,KeY,I,CYCNO,KCYGEN) CORE 474 60 TO 1000 CORE 475	KB = U CORE 472 415 CYCNO = AINT(RNCYC(I)+0.5) CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 473 GU TO 1000 CORE 475 GU TO 1000	KB = U CORE 472 415 CYCNO = AINT(RNCYG(I)+0.5) CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 473 GG TO 1000 GORE 475	KB = U CORE 472 415 CYCNO = AINT(RNCYC(I)+0.5) CYCNO = AINT(RNCYC(I)+0.5) CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 475 GG TO 1000 CORE 475	KHCYC(L) = KNCYC(L) = 1.0	CORE	171		
415 CYCNO = AINT(RNCVC(I)+0.5) CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CAT 1 1010 CORF 475	415 CYCNO = AINT(RNCYC(I)+0.5) CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) G0 T0 1000 CORE 475 475	415 CYCNO = AINT (RNCYG(I)+0.5) Call Dēcide(X1,X2,X3,X4,KeY,I,CYCNO,KCYGEN) Go to 1000 (2010 475) 475	415 CYCNO = AINT (RNCYG(I)+0.5) Call Dēcide(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) Core 474 GG TO 1000 Core 475	415 CYCNO = AINT(RNCYG(I)+0.5) Call Decide(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) CORE 474 GG TO 1000 CORE 475	415 CYCNO = AINT(RNCYC(T)+0.5) CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN) GG TO 1000 CORE 475		CORE 4	172		
CALL DECLUE(X1)X2)X4,KEY91,GTGNO,KGYGEN) CORE 474 GG TO 1000 LORE 475	CALL DECIDE (XI)XZ)X4,KEY)I,GTGNO,KGYGEN) CORE 474 475 66 TO 1000 CORE 475	CALL DECIDE(X1,X2,X3,X4,KEY)I,CTCNU,KCYGEN) CORE 474 CORE 475 CORE 475	CALL UCLIDE (X1, X2, X3, X4, KEY, 1, CTCNU, KCYGEN) CORE 475 475 60 TO 1000 475	CALL DECIDE(X1)X2)X3)X4)KET)19CTON09KGTGEN)	CALL UCLUE(X1)X2,X3,X4,KET),GTCN0,KGTGEN) CORE 474 G0 T0 1000 CORE 475	415 CYCNO = AINT (RNCYC(I)+0.5)	CORE	173		
GG TO 1000 475	60 10 1010 CORE 415	60 TO 1000 CORE 475	GO TO 1000 CORE 475	60 TO 1000 CORE 475	60 10 1000 CORE 475	CALL DECIDE(X1,X2,X3,X4,KEY,I,CYCNO,KCYGEN)	CORE 4	+24		
						GG TO 1000	CORE 4	•75		

	DOLTHE KHOL	CDC 6600 FIN V3.0-367A OPT=1 0	126/74	10.40.58.	PAGE	ß	
	2000	LMAX = L If (LRMAX .LT. 4) GO TO 5000 If (NCYNO .Eq. 0) GO TO 5000	CORE CORE CORE	476 477 478			
225	ა 	RANGE PAIR COUNT OF RESIDUE SPECTRUMS	CORE CORE	479 480			
		NRES = NRES + 1 Call Decres(LRMax,NCYNO) GO TO 2000		1010 0000 1-1-1-			
230	200 ²	IF (LRMAX .LE. 1) GO TO 3000 Count the last residue spectrum - range pair counting mill yield n	CORE CORE CORE	4 1 4 8 8 8 8 9 8 9 9			i.
235	00	RK = 0 RESMAX = RES(1) RESMIN = RES(1) IMAX = 1	0020 0020 0020 0020 0020	00000000000000000000000000000000000000			
240		IHIN = 1 D0 500 I = 2,LRMAX IF (RES(I) .LT. RESMAX) G0 T0 490 Resmax = res(I) IMAX = I	ORE ORE ORE ORE ORE	ი აიიი ი ი ი ი ი t t t t t t t		:	
245	490 500	G0 70 500 IF (RES(I) .6T. RESMIN) G0 T0 500 IF (NE = RES(I) IMIN = I Continue	CORE CORE CORE CORE	500 00 00 00 00 00 00 00 00 00 00 00 00		i	
250	510	CALL CYCRES(RESMAX,RESMIN,1.0,INDEX(IMAX)) KK = KK + 1 J = IMAX - 2 IF (J •LE. D) GO TO 550 IF (J •LE. D) GO TO 550 CALL CYCRES(RES(J),RES(J+1),1.0,INDEX(J))	CORE CORE CORE CORE	00000000000000000000000000000000000000			
255	055	TAX = KK + I IAAX = J J = IMIN + 2 IF (J .6T. LRMAX) GO TO 575	DRE DRE DORE DORE DORE	509 511 512 513		· · ·	1
260	575	CALL UTVERS(RES(J-1),RES(J),1.0,INDEX(J-1)) KK = KK + 1 G0 T0 550 KMAX = KK	JORE JORE JORE JORE	514 515 516 517 518			
265	3000 2 2 2 2 2	LTAX = L Sort the Analysis spectrum to produce the range pair counted spect KP = 0	ORE CORE CORE CORE CORE	519 521 521 522 523			
270		00 600 I = 1,NFKS 16 600 I = 1,LMAX 16 (NNSTEP(I) •NE. JJ) 60 TO 600 KP = KP + 1 KP = KP + 1	ORE CORE CORE CORE	524 525 5225 5225 5225 5225 5225 5225 5		:	1
275	:	NSTEP(KP) = KP	ORE	530			

280 290 290 290	<pre>xrcm SIGHAX(KP) = CYCLE(I,1) SIGHAX(KP) = CYCLE(I,2) RNCYC(KP) = RNECYC(I) IF (XIGHAX(KP)E. SIGHAX(KP-1)) G IF (SIGHAX(KP)E. SIGHAX(KP-1)) G IF (SIGHAX(KP)E. SIGHIN(KP-1)) G SIGHIN(KP)E. SIGHIN(KP-1)) G IF (SIGHAX(KP)E. SIGHIN(KP-1)) G COUTINUE 605 CUNTINUE 605 CUNTINUE 605 CUNTINUE 800 FORMAT(141,48X,334RANGE PAIR CYCLE HKITE(6,22) (NSTEP(I),SIGHAX(I),SIG HKITE(6,22) (NSTEP(I),SIGHAX(I),SIG HXITE(6,22) (NSTEP(I),SIG HXITE(6,22) (NSTEP(I),SIG HXITE(</pre>	300 6500 FTN V3.0-357A 0PT=1 0 то 600 0 то 600 0 to 600 counted Spectrum//) chin(i),rncvc(i),i = 1,крнах)	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	4 5 5 5 5 5 5 5 5 5 5 5 5 5	6 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9	

./

\int		· · · ·		· ·		· · · · · · · · · · · · · · · · · · ·								
							;	t E	:		·		:	
-		1								ı				8 9 9 1
AGE							i							
0.58.	0 4 N	ល់ 4 ៧ លំ	N 8 6 0 7		0 ~ 8 5	0								
10.4	0, 0, 5 0, 10 5 0, 10 50	ישי מי מי מי ישי מי מי מי	עי עז שי עז עז מי געז שי עז עז	0000 0000	00000000000000000000000000000000000000	52								
126/74	CORE CORE CORE	CORE CORE CORE CORE	CORE CORE CORE CORE	CORE CORE CORE CORE	CORE CORE CORE CORE	CORE								
1 04.	000	A U A				-								
0PT=		H FRO												
)-367A		ECTRU												
N V3.	6	IS SIS												
00 FT	EP (20	ANAL Y		000										
CDC 66	EPP)) ,NNST	R THE		1010										:
	F, NST C (200	ES FO		1) 6 2) 6				•	:					
	CYCP ,RNECY	S CYCL ECIDE		רב (ר) כרב (ר)	1.0									1
	1,Y2, 00,2)	ERATE INE D		100 E. CY	+ 2									
	CGEN (Y	NE GEN UBKOUT	71 70 70 70	STEPP 60 T0 ,1) .N	NECYC									•
,	NE CYC CYGZC CGDEZI	ROUTI BY SI		е с т - т - т - т - т - т - т - т -										1
	ROUTI MON/M	SUB PLIED		576P(L (CYCL (CYCL (CYCL	ECYC (L 40 = 1 1URN	_								•
YCGEN	3000 000 000	THI SUF		ХЦЦЦ ХНИН.	10 L = RN6 L I 00 R = 1	C N								
C W	c	0000		·	1									:
ROUTI														
SUB		ß	0	s	0									

5 5 5 5 5 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	ROUTINE DECIDE(X1,X2,X3,X4,KEY,I,CYGNO,KCYGEN)	ORE			
10 10 10 10 10 11 11 11 15 15 15 15 15 15 15	MON/MDEC1/SIGMAX(200),SIGMIN(200) Hon/Mdec2/NSIE/200),LR,KMX,KMIN,K31 Hon/Mdec7/RES(450),INDEX(450),IND1,IND2,IND3,IND4,KIND Hon/Mdec7/RES(450),INDEX(450),IND1,IND2;IND3,IND4,KIND MON/MGGDE/L,LIND	008 008 008 008 008 008 008 008 008 008	100400		
KFI 10 11 11 11 15 15 15 15 15 15 15 15	S SUBROUTINE DECIDES WHETHER OR NOT THE JALUËS X1,X2,X3, AND X4 Ny the adjusted load spectrum satisfy the range pair counting co	0000 0000 0000 0000	578 579 5390		
15 150 IF 150 IF	RST = 0 (K31 •NE、 0) GO TO 11 (X33 •LE、 X2) GO TO 200 (X2 •GT、 X1) GO TO 200	ORE CORE CORE	531 532 5332 553		1
CAL	(X2 .LT. X4 .OR. X3 .GT. X1) GO TO 500 (X2 .GT. X3) GO TO 151 L CYCGEN(X3,X2,1.0,NSTEP(I)) TO 152 	0000 000 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00 00 00		· · · ·
20 151 CAL 152 X1 152 X1 172 110	L UTUGENIAC;A3, 1.U,NSIEPLIJ) = X1 = X4 (INO3 .NE. IND2) LIND = 1 2 = IND4	2008 2008 2008 2008 2008 2008 2008 2008	0000 000 0000 0000 0000 0000 0000 0000 0000		4 6 7
25 KGY 15 210 15 210 15 60	GEN = 1 (KEY •NE 0) GO TO 110 URN (X2 .GT X4 .OR. X3 .LT. X1) GO TO 500	008 008 008 008 008 008 008 008 008 008	594 595 697 697		
200 X1 100 X2 100 X2 170	= 2. = X4 2 = IN04 66N = 2 KFY _ FO N) &FTURN	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0		
35 CC CC CC CC CC CC CC CC CC CC CC CC CC	NO = CYCNO - 1.0 TO 110 X1 TO THE RESIDUE SPECTRUM	2008 2008 2008 2008 2008 2008 2008 2008	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0		
40 CR XI XX XX XX XX XX XX XX XX XX XX XX XX	= LR + 1 (LR) = X1 EX(LR) = IND1 = X2 = X3		000 100 M 4	• • •	
	1 = IND2 2 = IND3 3 = IND4 GeN = 3 (KEY •NE• 0) GO TO 110	2000 2000 2000 2000 2000 2000 2000 200	611 611 618 618 619		
50 RET 110 60 1150 IF 1153 CYC	URN T0 (1150,1200,1500),KCYGEN (cycno .6t. 1.0) 60 t0 1151 cycno .ele. 0.0) return .n0 = cycno - 1.0	0086 0086 0086 0086 0086	6520 522 522 522		

		: }		1 	ł		1		· · · · ·				:		
PAGE 2												·			
10.40.58.	626 627 628	629 530 631	633 634 534	635 635 736	- 6 6 0 - 5 6 6 0 - 5 6 0 0 0 -	5 6 6 6 6 5 7 7 1 5 7 7 1 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7	0 0 4 4 7 0 7 7 0 0 0 7 7 0 0 7 7 0 7 0) (0 (0 (0 (0 (0 - キ キ) (0 (0 (0 - (0 (0 (0 (0 (0 (0 (0 (0 (0 (0 (0 (0 (0	0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0	0 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0	0 AI + M V 9 0 0 0 1 0 0 0 0 0 0	667			
1 04/26/74	CORE CORE CORE	CORE CORE	CORE CORE CORE	CORE	CORE CORE	CORE CORE CORE	CORE CORE CORE	00020 0022 0022 0022 0022 0022 0022	00000000000000000000000000000000000000	00000000000000000000000000000000000000	CORE CORE CORE CORE	CORE			•
CDC 6600 FIN V3.0-367A 0PT=1	2.0											·			
OUTINE DECIDE	1151 IF (LIND .EQ. 1) GO TO 1153 IF (IND3 .NE. IND4) GO TO 1153 CVCCYC(L) = RNECYC(L) + CYCNO - 1	UTCNU = 1.U 1152 IF (KHAX .NE. 1) GO TO 111 X3 = SIGMIN(I) IND3 = NSTEP(I)	IF (CYCN0 .61.0.0) 60 TO 112 KHIN = 1	KMAX = 0 KCYGEN = 3 Riturn	1200 IF(CYCNO .LE. 0.0) RETURN CYCNO = CYCNO - 1.0 X3 = SIGMAX(I) X4 = SIGMAX(I)	KFIRST = 1 60 T0 113 111 X3 = SIGMAX(I)	IF (KFIRST . NE. D) GO TO 113 CYCNO = CYCNO - 1.0	KFIRST = 1 113 INU3 = NSTEP(I) INU4 = INU3 KHIN = 1 KHAX = 0	G0 T0 10 1500 IF (XMAX .NE. 0) G0 T0 1510 IF(CYCNO .LE. 0.0) RETURN CYCNO = CYCNO - 1.0 112 X4 = SIGMAX (I) INAA = NYTEP(I)	KHAX = 1 KHAX = 1 KMIN = 0 60 10 11	TING = USTEP(I) KMAX = 0 KMIN = 1 GO 10 AND				
SUBR		60	l	6 2	02	7 6	2	80	8	06	56			•	
								34						1	

		70 FTU V3 V-1520 V1 V3 V-1520 V1 V3 V-1520 V01+4 V7	176 170			•	-
		AGS CULSEN AS CULSEN AS CULSEN AS CULSEN ACT	41/92	1U.+U.58.	PAGE	-1	
	:	SUBROUTINE DECRES(LRMAX,NCYNO) COMMON/MCGDE/L,LIND COMMON/MDECR/RES(450),INDEX(450),IND1,IND2,IND3,IND4,KIND	ORE CORE CORE	658 659 670			
1	0000	THIS SUBROUTINE DECIDES WHETHER OR NOT THE ELEMENTS OF THE RESIDUE (Spectrum satisfy the range pair counting conditions	00RE 00RE 00RE	671 672 673			
10	5	K = 0 NCTNO = J X1 = RES(1) X2 = RES(2)	ORE CORE CORE	675 675 677 878 878			
15		X3 = RES(3) X4 = RES(4) INO1 = INDEX(1) IND2 = INDEX(2) IND2 = INDEX(2)	CORE CORE CORE	6696 6691 6691 7683 7683			
	10	IND+ = INDEX(4) J = 4 IF (X2 61 X1 60 T0 100 IF (X2 11 Y, 00 Y2 61 Y1) 60 T0 100	CORE	1 0 0 1 to			
2	150	IF (X2 61: X3) 60 TO 151 C ALL CYCRES(X3,X2,1.0,INO3) 60 TO 152	2085 2085 2085	0 0 0 0 0 0 0 0 0 0 0 0			
55	151 152	CALL CYCRES(X2,X3,1.0,IND2) NCYNO = NCYNO + 1 X1 = X1 Y2 = X1		691 692 693 700 700 700 700 700 700 700 700 700 70			
Û E	·	XC = X4 ID2 = IND4 IF (J - EQ. LRMAX) GO TO 30D IF ((J + 1) .EQ. LRMAX) GO TO 315 X1 = PES(J+1) X2 = PES(J+1)	ORE CORE CORE	+ 13 0 N 00 1 T 13 0 N 0 0 1 D 0 0 0 0 1			
35		AT = RESULT IND3 = INDEX(J+1) J = J+2 G0 TO 10	000 00 00 00 00 00 00 00 00 00 00 00 00	202 202 202 202			
	100	IF (X2 .61. X4 .0R. X3 .LT. X1) 60 T0 500 60 T0 150 K = K + 1	ORE ORE	200 200 200 200			
0 I 4 .		RESK() = X1 INDEX(K) = IND1 INDEX(K) = IND1 IF (J •6T• LRMAX) 60 T0 330 X1 = X2	CORE CORE CORE CORE CORE	707 708 710 710 711			
τ τ τ τ τ		X2 = X3 X3 = X4 X4 = RES(J) IND1 = IND2 IND3 = IND4 IND3 = IND4	SORE SORE Sore Sore Sore Sore	712 713 714 715 715			Analysis (or the month and a second second
	300	IND4 = INDEX(J) 60 TO 10 65 EK + 1 65 ES(K) = X1	CORE CORE CORE	718 719 720		;	****** - ····*
55		RES(K+1) = X2	CORE	722			ſ

,			· · · · · · · · · · · · · · · · · · ·	
1			1	
ļ			<u>.</u>	
:				
1 N			<u>'</u>	
			:	
:				
ш			3	
AG AG			:	
			i	
+				
•			i i	
58				
	またえて下りのの人のちかえのようののくのです。 + やかややたちををををまるとうののくのこと			
	~~~~~~			I
174	the first first first first first first start start for the first start start start start start start start sta			i
26/				į
14	000000000000000000000000000000000000000			1
5				
11				
190				i
A L				
36			1	
-				j
E N				
Z			:	}
i.				
600				
ē G				
202	:			
1				
1				
	·	:		1
1				
1	Ŧ			
1				
			1	
	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX			Í
RE				
DEC	315	1		
I NE			1	
UT.				
BRC				
S			4	
	52 59 59			
				1

· · · · ·				4,		en on men manter ara e		~
1								1
								a •
								1
								I
1								1
AGE								
<u>م</u>								1
								•
0.5								
1 <b>0</b> - 4								
1715								2
4/26								ļ
ō	A A A							
1=1	NO N							
90								
679	i T RU							
V3.	s s							
1 N	100 Z		i.					
00	ANAL C				·			
66	HENNST		•					
202								
	2 00 1 FO				· · · ·	·	1 ·	
		:	· · · ·		)			
	SREAC							
	2, 1ES 0.E(S .F			;				
	1,Y Inc				· · ·	1		
	С С С С С С С С С С	,	9 5	:				
	CCRE L'ILIL UBR V1 V1 V1 V2 STEP	. •				:		
				:	1			
	JAN PLIE							
ŝ	HILL STORY		:		н. Н		. 1	
YCR							:	
U	0000							
. INE			:					
, DUT	4							
SUBR								
1	14 10 0						Į	
		1	:		:			
1								
<b>S</b>							······································	



# APPENDIX II

SAMPLE PROBLEM WITH INPUT DATA LISTING

	•			•							I			1														
																												1 So the set of the second se Second second sec
270,000 CYCLES					4 DATA 55.82462 48.26657 39.66455 31.70955			E2 = 1.000				1.0000	1 • 6 8 8 9 8	1.00000 1.00000	1.00000 1.00000	3 • 00060 1 • 00000	2 • 0 0 0 0		1 • ¢ 0 0 ¢ 0	1.0000	1.0000	1 - 00000 1 - 01000	7.0000	1.0000 /*# 60000	1.6000	35.0000	00000.0000	1,10,0000
TRUNCATION LEVEL	55.00000	40000	-1.83600		SQUARE FIT OF S- (N + G(I) 8(I) • 22041 • 43210 • 64199	FROM HERICAN	7	\\++£2) = 1.000 AND	44 LOADS		STMAX	- 02900 +	. 89500	• 6 8 4 0 0 • 5 8 4 0 0	• 69300	.56300 .50900	. 62700 . 5000	.0000	. 48200	• 36400 • 75600	.52700	• 70800 • 61760	. 51200	. 54800		- 244JO - 723DO	. 55700	. 37700
В-1 SPECTRUM 2219-T851 AL	STRESS (KSI)	ITERCEPT = .	FFIN-MANSON SLOPE	US ± 10000.00000	OF SECOND ORDER LEAST (I)*SHIN*2 + B(I)*SHI A(I) - 00217 - 00178 - 00149 - 00243	PON S-N DATA DERIVED F UPPLIED FROM NORTH AM	SS RELAXATION FUNCTION	'1/(KTSMAX**E1 * KTSHEA .25000000E+07 , E1	IES THROUGH BLOCK OF	IIT = 26.9000	STHIN	14800 8400	.58400	• 564 50 • 455 00	• • • • • • • • • • • • • • • • • • • •	•48800 •23500	.50900 .43400	.36400	. 36490	• 30400 • 30700	.31600	•54500	.40100	.1270Ú	00600	• 13000 • 07800	• 11100	.52700
SPECTRUM FROM Material type	TËNSILE YIËLD	LCF STRAIN IN	INVERSE OF CO	ELASTIC MODUL	COEFFICIENTS SHAX = A SHAX = A LIFE = A LIFE = A 10** 5 10** 5 10** 5 10** 5	UNNOTCHED COU Information S	RESIJUAL STRE	ĔŇĔP = C Whùre ci =	AIT 7	LOAD LIM	STEP TYPE		8			90 G 11 11	1 11 1 1 11 1	122	+ + + + + - 	15 16 1	1 21	19 2	20 1	22 1 22 1	<u> 23 1</u>	24 1 25 1	26 1	28 3

	(			<u></u>					!			•						 	
	1	!						;	1										
	1	I							į		:				•				
		!													4			·	
	-	1								÷	:		1					,	
	1								-				, 1		1				
	!			•					I										
									•										f   
		•																	: : *
Ì																			1
																			ļ
	:																		
	:																		
	l	:																	
;	:	,		_					;						, ,				
0000					0000										-				
1.00		500 100	1.0	6.6	00 • •				:		:		:	:	•			1	
	i	्न		N				:		,	-				ł			:	
		1	•	1			1				:		•	•	:				
ļ			:				:	1	t		;		1			•			
000			000	000	00		·	+			i	,	1		1	:		· .	
8830 7470	9946	5270 5270	810( 6870 7020	5000	0290	ΥΡE	:	•	1				:						
. · · ·	••	• • •	• • •	•••	;;	 ա	;	1					:	:		* 2	:		
ł		Ì				TΥΡ	1 .	•			•	•				5		1	
	İ	;	-			ΥPE	:		1										
1		;				ц Ц	•		:								•		
000			200	000	800	TΥF									:	÷		1	
.12	50	10.0	58 32	6 M 4		YPE			1						÷				
•••	l	ł	÷		• •	ц Ш			1						i				
}		:				TΥF		<b>?</b> ??î	•						1				
		i.				ΥΡE	004	0 4 0 C	;					,				1	
						5 1	11		:				1		-			;	
- <b>N</b>	ल का क	4 1 1			स्त स्त	ΤYF		4844	•			:							
; •		:	= .			ş	~1 (V M	40.01		,	i.								
202	25	2 2 2 1 2 2 2 1 2 2 2 1	2 8 8 2 8 9 9 8 9 9	0 H N 4 4 4	5 d 1 d 1 d	BLO			-	,					1				
	· · • • • • • • • • • •	·										·						 ····· /	

SPECIRUM FROM	8-1 SP	FCTRUM	TRUN	CATTON 15	76 127.				1	 :	
	5 4 5										
AKT = 4.50											
RELAXATION CON	STANT C1=	250	0000-00								
											-
SPECTRUM SUBJE FLIGHT OR BLOC	CTED TO T K NO.	HE RANGE 1	-PAIR CO	UNTING TE	CHNIQUE						
STHAX STAIN	SIGHAX	SIGMIN	RES	EQRES	ENN	NEP					
783-98 18.40 15.71	-3.51 55.00	-17.92	0.00	00.00	1.00	• 13025831E+05	<del>स्</del> ।	-1			
15.71 17.24	00.00	40.04 07	0.00	-21.80	1.00	• 39342782E+03	0	0			
18.64 17.43	55.00	49.55	-27.75	-12,69 -28,89	1.00	• 56235689E+03 • 36717948F+13	e M	m a			
16.80 10.63	55.00	10.20	-28.89	-29.61	1.00	• 44521680E+03	r un	+ ư			
	38.54	29.46	-29.61	-13.15	3.00	• 57657259E+03	9.00	פיז			
13.04 0.52 15.27 12.50	32.20	- 97	-29.42	-6.61	1.00	<ul> <li>90105168E+03</li> </ul>	7	7			
13.69 11 57	40.04	62.25 22 25	-29.36	-20.90	2.00	.47906511c+03	<b>6</b> 0	8			••.
13.56 9.79	54.32	14.86	-29.21	-28.52	1.00	• / 10 9 0 3 9 / E + U 3 • 4 6 9 1 9 7 1 3 7 + U 3	jn ⊂ •	ſ, Ċ			
9.79 3.58	14.86	-13.10	-29.20	0.00	1.00	.18861779£+34	11	11			
12.37 9.79	29.18	14.93	-29.17	-3.35	6.00	<ul> <li>83680259E+03</li> </ul>	12	12			
9.79 8.18 20.34 8.25	15.32	8•06	-28.74	0.00	1.00	14033389E+04	13	13			
	00.00	• • • •	-28.70	-30.51 	1.00	• 42460525E+03	t. 	14			
15.60 8.98	38.26		- 36. 4 3 - 36. 4 3	10.19	1• 10	• 168 U 7252E+U 3 E 81 E 77E 001 + 013	15	15			
13.77 10.79	25.61	12.18	-36.36	-6.98	00.1	• 20123359UETUS • 729961035+03	17	10			
14.74 3.42	30.61	-20.35	-35.72	-11.34	1.00		4 F	2 T			
10.14 b.16	26.6	+6•2-	-35.66	0.00	48.00	• 1 + 0 3 2 2 3 0 E + 0 +	19	5 <b>(</b> **			
RELAXATION	10.11	-7.81			4.80		,	19			
ACLAXALION	10.37	55			4.80			20			
RELAXATION	10.00	57•7•			4•80			21			·
RELAXATION	11.14	- 6.77			4. 80 4. 80			22			
RELAXATION	11.40	-6-52			4.80			0 V 0			•
RELAXATION	11.65	-6.27			4.80			25			
RELAXATION	11.90	-6.02			4.80			Sö			
RELAXATION	12.40	-5.52			4°80			27			<b>.</b>
9.1516	8.04	-33.84	-33.12	0.00	1.00	.300479635+04	00	20			
6.56 3.50	-3.56	-17.36	-33.09	0.00	35.00	• 373920256+04	21	30			
RELAXATION	-3.52	-17.32			3.50	•	ļ	30			
RELAXATION	51°M	-17.25			3.50			31			<b></b>
RFLAXATION	00°°0"	-1/ 18			3.50			32			
RELAXATION	-3.24	-17.04									
RELAXATION	-3.17	-16.97			3.50			1 IS			
RELAXATION	-3.10	-16.90			3.50			36			
KELAXATION	-3.03	-16.83			3.50			37			
RELAXATION	0.4.7	-16.60			3.50			38			<b>.</b>
19.45 2.10	55, D.D		02 22 -	-10 50	, .			5.			
14.93 2.99	34.91	-19.08	-32.52	-12.42	00 • 6	• 91708325F+03	22	D -			
10.14 5.03	13.57	-9.43	-32.07	0.00	10.00	.15047965E+04	2.0	5 T			
, 20.09 14.18	55.00	26.37	-31.61	-35.42	1.00	. 35355054E+03	25	1 M 1			
78 -3.98	-38.93	-53.34	-35.42	0.60	1.00	13325831E+05	26	t -			
22.67 17 50 23.67 17 50	55.00	20.02	-35.42	-47.41	1.00	• 28748314E+03	27	45 2	, t		Ŷ
14-18 8-90	10.04	50°01	-47.41	-51.52	, 00 , 1	• 27916148E+03	28	1-Q			
10.60 14.18	23.29	12.40	-51.30	04.01-	10.00	• 7 3 4 6 4 4 6 8 5 4 1 3 . 4 8 2 4 9 5 4 4 5 4 0 3	62	1.7			
RELAXATION	23.44	12.54			1.90		30	10 m 17 - 1			
RELAXATION	23.72	12.83			1.90		· · · · · · · · · · · · · · · · · · ·	64			

				÷		
。 。 。 。 。 。 。 。 。 。 。 。 。 。 。 。 。 。 。	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	COUNTER K	1.0000 1.00000 1.00000 1.00000 1.00000 1.00000	5,00000 1,00000 2,000000 1,00000 1,00000 1,00000 1,00000	6,00000 1,00000 1,00000 1,00000 1,00000 1,00000 1,00000 4,800000	4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.8000 4.80000 4.80000 4.80000 4.80000 4.80000 4.80000 4.800000 4.800000 4.800000 4.800000 4.800000 4.8000000 4.80000000000
5336542E+13 5636542E+13 2605652E+13 9170979E+103 7468264E+103 34	00+9347E+03 35 6645141E+03 35 2236611E+03 37 37 3025831E+05 38 4337171E+05 38	= 75 SIGMA MINIMUM	1791545+02 4289505+02 2727955+02 4955275+02 4955275+02	<ul> <li>&lt; 2445855 + 112</li> <li>&lt; 97113955 + 01</li> <li>&lt; 3255465 + 02</li> <li>&lt; 325702 + 02</li> <li>&lt; 1485685 + 02</li> <li>&lt; 1310245 + 02</li> </ul>	<pre>.148358E+02 .80556E+01 .6495502E+01 .173800E+01 .3399395=01 .121764E+02 .203494E+02 203494E+02 23494E+02 254300E+012 754400E+01</pre>	7287335+01 7287335+01 7287335+01 657335+01 65266956+01 5523046+01 55230466+01 3384445+02 17230465+02 17230465+02 17230465+02 17230465+02
1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00	1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	INPUT LOAD SPECTRUM MAXIMUM	<ul> <li>. 3510456+01</li> <li>. 5500006+02</li> <li>. 4289506+02</li> <li>. 5500006+02</li> <li>. 5500006+02</li> <li>. 5500006+02</li> </ul>	<pre>&gt;&gt;&gt;&gt;&gt;&lt;<p>&gt;&gt;&gt;&gt;&lt;<p>&gt;&gt;&gt;&gt;&lt;<p>&gt;&gt;&gt;&lt;<p>&gt;&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;&lt;<p>&gt;&gt;<td><ul> <li>2917976402</li> <li>1531866402</li> <li>1531866402</li> <li>2727966402</li> <li>3825706402</li> </ul></td><td>.1062755+02 .1088565+02 .114496+02 .1154876+02 .1189936+02 .1189936+02 .1214816+02 .1214816+02 .1214816+02 .1214816+02 .1214816+02 .337866+01</td></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></p></pre>	<ul> <li>2917976402</li> <li>1531866402</li> <li>1531866402</li> <li>2727966402</li> <li>3825706402</li> </ul>	.1062755+02 .1088565+02 .114496+02 .1154876+02 .1189936+02 .1189936+02 .1214816+02 .1214816+02 .1214816+02 .1214816+02 .1214816+02 .337866+01
24.00 24.00 24.52 24.52 24.52 24.83 24.83 24.83 25.41 14.75 25.41 14.75 25.64 14.75 25.64 14.75 25.64 14.75 25.64 14.75 25.64 14.75 25.01 15.01 14.75 25.01 15.01 14.83 25.01 15.01 15.01 15.01 15.01 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03 15.03	6.23      31       -47.81         24.39       5.14       -47.81         19.75       9.53      47.01         19.75       9.53       -47.01         20.50       10.33       -47.01         21.12       9.95       -47.01         21.12       10.33       -47.01         21.04       11.04       -47.01         21.05       11.04       -47.01         21.05       11.04       -47.01         21.05       11.04       -47.01         21.05       11.04       -44.00         21.05       11.04       -44.00         22.05       11.04       -44.00         21.05       11.04       -44.00         22.05       11.24       -44.20         22.05       12.04       -44.20         22.05       12.04       -44.20         22.05       12.04       -44.20         22.05       12.04       -44.20         22.05       12.04       -44.20         22.05       12.04       -44.20         22.05       12.04       -44.20         22.05       12.04       -44.20         23.04       -44.20<	KS OR VALLEYS IN THE Step	ት 0 ው <b>ታ</b> በ ባ	0 M 00 0 7 7	1 1 1 1 1 1 1 1 1 2 0 0 1 2 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION 21.79 18.48 18.48 15.63 18.66 18.66	12.08 12.08 14.66 14.66 14.66 14.66 14.66 14.66 14.66 14.66 14.66 14.66 14.66 14.60 Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Relaxation Rela	THE NUMBER OF PEA	•			

			)
			į
			:
		:	
и и области и объека и и и и и и и и и и и и и и и и и и и		Ĩ	
44444444444444444444444444444444444444			
0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0       0			
<ul> <li>✓ 𝔅 𝔅 𝔅 𝔅 𝔅 𝔅 𝔅 𝔅 𝔅 𝔅 𝔅 𝔅 𝔅</li></ul>	,		
	!		
ਲ਼ਲ਼ਲ਼ਲ਼ਲ਼ਲ਼ਲ਼ਖ਼+++++++++++++ ਲ਼+ਲ਼ਫ਼ੵੑਲ਼ਫ਼ਲ਼ਫ਼ਗ਼ੑਗ਼ੑਲ਼ਲ਼ੑੑਲ਼ੑਲ਼ਲ਼ਲ਼ਲ਼ਫ਼ਗ਼ੑੑਗ਼ੑੑੑਫ਼ੑੑਫ਼ੑੑੑੑੑੑਫ਼ੑੑੑੑੑੑੑੑ			
	,		
		^	'

x
$\tilde{\sim}$
-
- 55
.0
ຼຸພ
α.
s
0
- 111
=
~
ੁ
Q
ပ
Luf
- 3
- 75
2
- 55
0
~
œ
н
-
۵.
_
ьı
õ
¥
-
~
ų,

		· · ·	:		
S	d B	MAXIMUM	SIGMA Minimum	COUNTER K	
	+1	351045E+01	179154E+02	1.00000	
	<b>N</b> 1	•550000E+02	•272795E+02	1.00000	•
	ۍ ۲	•550000E+02	• 4 955275 + 02 - 55 00 005 + 02	1.00000	
	r un	.3219665+02	- * 2 3 4 5 8 5 1 + 0 2	1.0000	
	9	.465387E+02	• 3225482+02	1.0000	
	~ •	•465387E+02	•232570£+02	1.00000	
	o 6		• • • • • • • • • • • • • • • • • • •	1.00000	
	0	•148601E+D2	•1 485 68E+02	1.00000	
		.291797E+02	•148958E+02	5.0000	
	2	.291797E+02	• 8 0 55 6 2 ± + 0 1	1.00000	
	50	•1551866402 •550000E+02	•148958E+02 •131024E+02	1-00000	
	2	<ul> <li>272796E+02</li> </ul>	.173300E+01	1.00000	
	. 91	•382570E+02	• 6 4 85 5 0 E + 0 0	1.00000	
	. 8	.3U612/E+U2 .1D1054E+02	•1 217 64E + 0 2 • - 7 8 1 3 0 6 7 + 0 1	1.00000	
	6	.103674c+02	- · 7548 00E+01	5.00000	
	0	.106275E+02		5.00000	
	10	•111685554UZ	/029/8E+01 - 277557404	5.00000	
	10	.113962E+D2		5.00000	
		116487E+02	- • 6 2 6 6 9 E + 0 1	5.00000	
	5	.118993E+02	601507E+01	5.00000	
	~	.123950E+02		00000.4	· · · · · · · · · · · · · · · · · · ·
	8	.123950E+02	- 203494E+02	1.0000	
	5	•803921E+01	5520365+01	1.0000	
	3.	352059E+01 - 356054E+01		4.00000	
	10		- • 1 7 1 7 8 1 E + D 2	4.0000A	
		330750E+01	171072E+02	4.0000	
	5	323678E+01	170365c+02	4.00000	
	5	316621E+01 - 7005705404	169059E+02	4.0000	
	0	3095525+01	• • 1 6 8 3 3 5 5 4 1 5 • • 1 6 8 3 1 3 5 4 1 2	4+0000	
	. 60	295540E+01	-1675515+02	4.00000	
	63	288544£+01	166831E+02	4.00000	
	•	•550000E+02		1.0000	
	1	.13353000F+02	10+10+190+0==	10.0000	
7	m	.5500005+02	•200165E+02	1.00000	
	- - -	•550000E+02	-+533397E+02	1.00000	
- <b>-</b>	n v	•234369E+02	•125424545	2.00000	
<b>.</b>	0 1	+23/2135+U2 -24AA31F+A2	•1 282 385 + 02 •1 31 3 467 + 02	2.00000	
	8	-242824E+J2	1338795+02	2-0000	
7	6	.245592E+02	.1365472+02	2-00000	
	0	•248335E+02	.139396E+02	2.0000	
		.2510532+02	-142108E+D2 -144802E+D2	2.00000	
	1 10	.2564166+02	•147471E+02	2.0000	
	<b>t</b>	•259062č+02	150117E+02	1.00000	
	2	•259062E+02	•751720E+01	1.00000	
IA III	0	• 1 7 1 4 3 4 C + U C	• 1 5 01 17E + 0 Z	1.00000	

						•																															1. 1. 1. 1. 1. 1.							
		3				ĺ													1																									1
	l																	۲ ت	2																					!				
	:																	2 63 2 AF																			1							
1.0000 1.00000 1.00000	1.00000 4.00000	1.00000 3.00000	3.00000 3.00000	3.00000 3.00000	3.0000	3.00000 3.00000	3.0000	2.00000	1.00000									C STRAINS= .986		ENN/CYC 1000005-08	.67531722E-05	.14306010E-08	• 1000000000000000000000000000000000000	• 366786192+08	.19310405E-06	•27228454E-12	•009/03005-04	.39476395E-08	•71163414E-08	• 23 7037142-11 • 40000005-02	•18856263E-07	•57293+29£-06	.357570316-08	.3000000F-08	•50030000E-08	.300000000000	.89196403E-08 .831447075	.89102593E-08	.89071993E-08	.43052316E-08	.71235984£-08 	.100100000000	•4000000E-08	•4000000¢-08
•751720E+01 •114564E+02 •219975E+02	.397609E+01 .514165E+01	311306E+00 .3578302+01	•395459E+01 •103269E+02	.106952E+02 .110595E+02	•1142305+02	.117766ë+02 .121295E+02	.1247852+02	•1282385+02 5474555704	• 5354 085 + 02		DAMAGE	• 5 89085E - 03	• 2367335-05 • 0673•6-06	• 8 9 1 5 4 8 F = 0 4	.4980222-07	•242990E-04		.718916E-05 .718916E-05 DAMAGE FROM PLASTI		CYCLES 10300005410	• 148078565406	• <u> </u>	• TOUDOOD • • • • • • • • • • • • • • • • • •	• 115368705 +09	. 517855515+07	• 367262875+13 • 403 + 96775 + 96	• 1433103767403 • 500664157412	. 126657975+10	• 14032164E+09	• 4/10/00/15 + 15 • 100/00/05 + 75	• <u>53004668E+08</u>	. 174540085+07	• 2/9665285 + 09 • 000000000 • 0	• 100000002+10 • 1000000002+10	. 10000000110	• 10000005+10 7727.00005	• 560385845409 • 560385845409	• 26115089E+09	• 561343675+09	• 301+0+301+09	+ 561919476 + 14 - 2737628471 + 08	• 10000001 •	• 1000000000000000000000000000000000000	<ul> <li>1000000015+10</li> </ul>
•151434E+02 •496805E+02 •348288E+02	•367628E+02 •243886E+02	.1974655+02	•201228c+02 •2049512+02	•208634£+02 •212277E+02	•215882E+02	.219448c+02 .222977E+02	*226467E+U2	-229920E+02 -229920E+02	417990E+02	ILTING FATIGUE LIFE	HAX OR HIN	MAX	XAX	MAX	MAX	MAX X X X	XTH	NIH		RNCYC 1.	1.	• • •	1 • 1	•	•	• •	• •	5.	• •	• •	1.	•	1. 1.		ហំព		<b>5</b>	5.	5.	• • •	• • • •	1.	•	• •
53 58	61 62	ő3 v	5 <del>1</del>	66 67	68 1	69	11	73	54	SITU SIKATNS WIKESU	PLASTIC STRAIN	.00636	• UUUS5	.00249	• 0000+	• 00123	.00156	• 00063		516MIN -17.92	27.28	+4.55 -55.30	29.46	32.25	23.26	79.20	14.36	14.90	8.U5	-13.10	1.74	• 65	-7.81	-7.55	-7.29		-6.52	-6.27	-6.02		-24.35	-5.52	-17.32	67°J1-
										- JIREJJEJ ANU PLA	STEP	2 -	τ ID	14	22	67 22	28	3 d		516#4X	55.00	55.00	32.20	40.54	46.54	54.32	14.86	29.18	53•18 15.37	55.00	27.28	38.26 31: 61	10.11	10.37	10.63	11.14	11.40	11.65	11.90	12.10	12.40	40 • 8	-3.52	67 * 0 F
					- <b>-</b>					1						-										-			-							<b>-</b> .							•	

. +000000000000000000000000000000000000	.161625466-02 4 4 6 9 9 9 10	11 12 14 15 15 16 16
<ul> <li>100000006+10</li> <li>100000006+10</li> <li>100000006+10</li> <li>100000006+10</li> <li>100000006+10</li> <li>100000006+10</li> <li>100000006+10</li> <li>100000006+10</li> <li>5265566906+05</li> <li>5276565410</li> <li>52765564506+10</li> <li>5280556906+10</li> <li>52805566906+10</li> <li>52805566906+10</li> <li>52805566906+10</li> <li>52805566906+10</li> <li>52805566906</li> <li>1000000000000000000000000000000000000</li></ul>	101AL ENN CYC =, 01 10 10 10 10 10	11 11 12 12 12 12 12 12 12 12 12 12 12 1
	NEP 13025831E+05 39342782E+03 5623568922403 5623568922403 44621680E+03 44651680E+03 576672592403 576672592403 44991351681403 4691923572403	• 18851779E+04 • 83680259E+03 • 414033896E+04 • 42460525E+03 • 76807252E+03 • 58153350E+03 • 72996103E+03
\$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$	Щ Н Н Н Н Н Н Н Н Н Н Н Н Н Н Н Н Н Н Н	4-99 4-99 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-4-60 4-40 4-4
	EQRES 0.00 1.57.69 -27.69 -15.69 -28.89 -28.89 -28.61 -20.90 -20.90 -20.90 -26.51 -26.51 -26.51 -26.51	0 00 - 3 35 - 3 6 - 4 6 - 6 9 8
	RES RES - 37,07 - 37,07 - 37,07 - 37,07 - 36,87 - 36,84 - 36,844 - 36,944 - 36,9444 - 36,9444 - 36,9444 - 36,9444 - 36,9444 - 36,94444 - 36,9444 - 36,9444 - 36,9444 -	-36.20 -36.15 -35.61 -35.61 -35.55 -35.55 -36.51 -36.43
		-20.10 7.91 1.19 1.74 1.74 1.74
	NDC. NDC. 140.4A 450.73 33.68 475.96 47.46 33.68 47.44 25.06 25.06 25.06 25.06 25.29 472.29	7.87 22.20 8.45 55.00 27.20 28.26
	STM 3LOCK STMIN STMIN 15.71 12.24 17.43 13.63 13.63 13.63 13.63 13.69 11.67 13.69 11.67 13.69	3.58 9.79 8.18 8.26 8.50 8.50 10.79
	FL GH C C C C C C C C C C C C C C C C C C	9.79 12.97 9.79 20.34 14.18 14.18 14.18 13.77

 $\chi^2_{i}$ 

																																							۲					
	• • • • • • • •													~ ~		-		-		-																								
19 13	20	21 22	23	10	رې 25	27	29	30	30	31	33	t M	5 M	37	38	39		4 I	5 m	<b>t t</b>	4-5	t 6	- <del>-</del> - <del>-</del>	1,3	0 t	51	52	53	5.5	50	56	59	ьи 61	62	63	64	+ U 0 0	66	67	6 G 0 G	20	17	72	- 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1
61	,						20	21	2							1	22	5 . T	25	2ô	27	28	30	1							31	32	0 ± 0	35	36	37								38
• 1+305/301+1+							. 2006 706 75 400	• 37392025E+04									• 58920931E+03 • 017083955 ±03	• 16047965E+04	.358550546+03	<ul> <li>13025831E+05</li> </ul>	- 28748314E+03	• 2/3101482 +U 3 • 754644686 + 0 3	• • • • • • • • • • • • • • • • • • • •								• 65336542E+03	• 32605652E+03 70170705.33	• 3 3 1 1 3 1 3 E 1 3 • 4 7 4 6 8 2 6 4 E + 0 3	90043347E+03	• 50645141E+03	• 62236611E+03								• 13025831E+05
4.80 -	4.80	4.80 4.80	4.80	4°80	t • 80	4.80	4.80 1.00	35.00	3.50	3.50	3.50	3,50	0.50 0.50	3.50 3.50	3.50	3.50	0.00	10.00	1.00	1.00	1.00		19.00	1.90	1.90	1.90	1.90	1.90	1.90	1.90 1.90	2.00	1.00	1.00	00 • 6	5.00	29.00 2.40	2,90	2•90	2.90	2.90	2.90	2°90	06.2	1.00
							0,00	00.0									-12.42	0.00	-35.42	0.00	-47.41	56.1C-	-19.69								-8.79	-43.05	-29.98	0.00	-16.42	-10.97								00.00
							.12	60.									-32,52	-32.07	-31.61	-35.42	-35.42	-51.52	-51.39								-48.65		-48.21	-48.13	-47.03	14041			i					-42.81
\$							- 3.	i iii																																				
-7.31	-7.55	-7.03	-6.77	- 6.52 - 6.27	-6.02	-5.77	-33.84 -33	-17.36 -33	-17.32	-17.18	-17.11	-17.04	-16.90	-16.83	-16.76	-16.69	-19.08	N+ 01	28.37	-53,34	20.02	-11.46	12.40	12.54	12.83	13.39	13.66	10.94	14.48	14.75	7.52	22.00	-9,24	31	5.14	4.54 9.59 9.59	9.95	10.33	10.70	11.42	11.78	12.13	12.82	-55.00
10.11 -7.81 -3	10.37 -7.55	10.89 -7.03	11.14 -6.77	11.49 -6.52 11.65 -6.27	11.90 -6.02	12.15 -5.77	12.40 -5.52 8.04 -33.84 -33	-3.56 -17.36 -33	-3.52 -17.32 -3.65 -17 25	-3.38 -17.18	-3.31 -17.11	-3.24 -17.04	-3.10 -16.90	-3.63 -16.83	-2.96 -16.76	-2.89 -16.69	34.91 -19.08	13.57 -9.43	55.00 28.37	-38.93 -53.34	55.00 20.02 55.00 10 00	12.27 -11.46	23.29 12.40	23.44 12.54	24.00 13.11	24.28 13.39	24.56 13.66	25.11 14.21	25.37 14.48	25.91 15.01	15.14 7.52	34.83 22.00	36.76 -9.24	6.2331	24•39 5•14	19.75 9.58	20.12 9.95	20.50 10.33	21-23 11-75	21.59 11.42	21.94 11.78	22.50 12.13	22.99 12.82	-40.60 -55.00 -41.80 -53.54

C STRAINS= .30462764E-03	<pre>&amp; 10030000E - 08 . 41504493E - 05 . 2314493E - 05 . 2314493E - 10 . 2314493E - 10 . 2314495E - 12 . 42263381E - 10 . 537754415E - 04 . 537756 - 04 . 1373577E - 08 . 4755219E - 08 . 47155219E - 08 . 47155219E - 08 . 1000000E - 08 . 100000E - 08 . 100000E - 08 . 10000E - 08 . 1000E - 08 . 100E - 08 . 10E - 08 . 10E - 08 . 10E - 08 . 10E - 08</pre>	.1000005-03 357590315-05 .357570312-05 .357570312-05 .50000005-08 .50000005-08 .31934035-08 .31135935-08 .8917125935-08 .8917129355-08 .890528165-08 .890528165-08	.36577961E-07 +1000000E-08 +4000000E-08 +40000000E-08 +40000000E-08 +40000000E-08 +40000000E-08 +4000000E-08 +4000000E-08 +4000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +10000000E-08 +10000000E-08 +10000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +1000000E-08 +10000000E-08 +10000000E-08 +10000000E-08 +1000000000E-08 +10000000E-08 +10000000E-08 +10000000E-08 +10000000E-08 +1000000000E-08 +100000000E-08 +10000000E-08 +10000000000000E-08 +1000000000000000000000000 +10000000000	.37873316-09 .378836256-09 .379687656-09 .3795876-09 .3795876-09 .37958216-09 .380429316-09 .380429316-09 .38122086-09 .38122086-09 .38122086-09
DAMAGE -2020565-05 -4980226-07 -4980226-07 -2423916-04 -2423136-05 -331336-03 -3792916-05 DAMAGE FROM PLASTI	$\begin{array}{c} CYCLES \\ CYCLES \\ \bullet 1000000 \\ \bullet 24033777 \\ \bullet 432060725 \\ \bullet 10730666 \\ \bullet 10730666 \\ \bullet 10730666 \\ \bullet 10730696 \\ \bullet 10730905 \\ \bullet 10000000 \\ \bullet 10000000 \\ \bullet 10000000 \\ \bullet 10000000 \\ \bullet 1000000 \\ \bullet 100000 \\ \bullet 10000 \\ \bullet 1000 \\ \bullet 10$	1000000000000000000000000000000000000	<ul> <li>Z73762895408</li> <li>100000014410</li> <li>100000014410</li> <li>100000014410</li> <li>100000014410</li> <li>1000000014410</li> <li>10000000014410</li> <li>10000000014410</li> <li>10000000014410</li> <li>1000000000000000000000000000000000000</li></ul>	• 52807566±410 • 52793259±40 • 52715246±10 • 52715246±10 • 526517796±10 • 5265212796±10 • 5265212796±10 • 52552184€+10 • 375891542409
MAX OR MIN MAX MAX MAX MAX MAX MIN	R R C C C C C C C C C C C C C C C C C C		<b></b>	N N N N N N N I
PLASTIC STRAIN • 00032 • 0004 • 00123 • 00123 • 00156 • 00156	X SIGMIN 	42       52       54         42       42       54         42       42       42         42       42       42         42       42       42         42       42       42         42       42       42         42       42       42         42       42       42         43       42       42         44       42       42         45       42       42         45       42       42         46       42       42         47       42       42         46       42       42         47       42       42         47       42       42         47       42       42         47       42       42         47       42       42         47       42       42         47       42       42         47       42       42         47       42       42         47       42       42         48       42       42         47       42       42		10 13.11 28 13.59 13.66 13.66 13.4.21 14.21 14.75 14.75 15.01 15.01 15.01
STEP 14 22 25 27 28 38	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	27.2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2		24-1 24-1 25-1 25-1 25-1 25-1 25-1 25-1 25-1 25

	20700526E-11 544495972-10 41404344E-04	5324079E+09 564968532-07	12775220ĕ-07 12915162E-07	445513736-09	+46400010-00	43710563E+09 * 3* 7**************	+32596096-09	+3065799E-09	+28926308-09	+ C 7 3 37 3 3 E + D 3 2 8 1: 0 4 6 : 1: 5 1 - 0 3	cototottc=09 202515605=08	10000006-08	.83070929£-03	24469639E-02				3	•	10	.0.1	~	0 0		1	0	3	ı t.	0.4	0						×	١t		0			0					tuo		7		
	•483079515+12 •2 •155159775+11 •5 •241520555+05 •4	.158142585+10 .5 .150383065+08 .6	<ul> <li>313106155+09</li> <li>774283725+08</li> <li>1</li> </ul>	. 673379845+10 .	. 682306915+10 .	• 686332965+10 • •	. 693487555+10 .4	. 69650846E+10 .	• 699420855+10 701033155+10	• · · · · · · · · · · · · · · · · · · ·	5 60+321682564 ·	. 1000000E+10 .	DAMAGE PER THIS SET=	IULAL ENVILLE		4	0	m	±	u.	س	~ 0	• •			12 13	13 13	े । स्व	21 24	17 11	18	19 19	50	2 t	N N	Ń	Ň			<u>ت</u> ر		3(	22	57 57	ŏ M		6 K	Ĩ	2	30 30	
					•										NFP	.13025831E+05	.25777999E+03	• 39342782E+03	• 56235689E+03	• 36717948E+03	• 44621680E+03	<ul> <li>5/bo/259t+03</li> <li>0010f16168f403</li> </ul>	• 50105105105	-710903975+03	• 46919213E+03	18461779E+04	• 83680259E+03	• 14033389E+04	• 4640U56564U3 • 7680725974D3	• 38464145E+03	• 58153350E+03	•72996103E+03	.92247814E+03										• 30047963E+04	• 3/33/6/25404						•	
•	ੇ <b>ਦੇ ਦੇ</b> ਹੈ ਦੇ ਦੇ 	• • 	•• ++ t	е е Ю М		• • •	• •	° M	* * M		; <del>.</del>	•			FNN	1.00	1.00	1.00	1.00	1.00	1.00	5•UU	2.00	1.00	1.00	1.00	6.00	1.00	1.00	1.00	1.00	7.00	1.00	4.80	4.80	4.80	4•80 *	4• 0 C	<b>4.80</b>	4.80	4.80	4.80	1,00 77,00	3.50	3.50		3.50	3.50	3.50	3.50	
			•		;										EORES	0 • 0	-53.34	-27.80	-15.69	-28.89	-29.61	-13.13 	-20.90	-6.61	-28.52	0.00	-3.35	00.0		-30.70	-19.69	-6.98	-11-34										00.0	0 n n n							
			;		;										RES	-37.07	-37,07	-53,34	-53.19	-53.04	-52.89	-52.29	-52.18	-51.88	-51.73	-51.62	-51.56	-50.69	-50.61	-50.48	-50.36	+20.24	95.54-										-45.69	00 • 6 4 -							
	-11.46	96.52	5.14 31	9.58 9.95	10.33	10.00	11.42	11.78	12.15	12.82	5.14	-53.54			SIGMIN	-54.99	17.35	17.35	1.89	0 <b>1</b> • • 0	10.4	-23.85	0. tt	• 66	-7.67	-35.52	-7.49	-13.97	-12.35	15.49	-9.93	-1.70	-21.48	-21.30	-20.94	-20.58	72.02-	+19.52	-19.17	-18.83	-18.43	-18.14	-46.42	-29.87	-29.77	-29.68	-29.58	-29.48	-29.38	-29.29	
46.45	10.14 15.14 15.14	36.76	24.39 24.39	19.75 20.12	20.50	21.23	21.59	21.94	22.65	22.99	22.99	-41.80		NO. 3	SIGMAX	-40.58	55.00	29.46	17.50	28.82 25.45	51.75 15.29	9.32	23.72	9.74	31.79	-7.56	6.79	-0.4 1.4 2.4	13.19	35.22	24.32	11.74	11.04	-3.39	-3.02	-2.66	-1.91	-1.68	-1.26	91	57	- 23		-16.07	-15.97	-15.63	-15.78	-15.68	-15.58	-15.49	
4														DR BLOCK	STMIN	-3.58	15.71	15.71	17.71	11.45	12.12	6,32	13.69	11.67	67.6	3.58	61.6	0.13 8.25	8.50 0	14.66	8.98	67.UI	5.16 6.16	ION	NOI		NOI	NOI	ION	NOI	NOI	NOI	01.	ION	NOI	ION	NOI	ION	NOI	ION	
														 FLIGHT (	STMAX	78	24.08	18.40	12.61			13.69	16.87	13.69	19.56	9.79	12.97	42.65	14.18	19.05	16.50	13.77	10°14	RELAXAT	ZELAXAT	- KCLAXAI	RELAXAT	RELAXAT	RELAXAT	RELAXAT	RELAXAT	RELAXAT		RELAXAT	RELAXAT	RELAXAT	RELAXAT	RELAXAT	RELAXAT	RELAXAT	

ξ.

		<u>.</u>	
		= ,82396546E-03	CCC CCC CCC CCC CCC CCC CCC CCC CCC CC
+ + + + + + + + + + + + + + + + + + +	9 4 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9	ASTIC STRAINS	<pre>+10 +10 +10 +10 +10 +10 +10 +10 +110 +1</pre>
3 2 5 2 5 2 5 2 5 2 5 2 5 2 5 2 5 5 5 5	24 24 24 24 25 25 25 25 25 25 25 25 25 25 25 25 25	DAMAGE +438461E-03 •141845E-03 •141845E-03 •180144E-03 •379281E-04 •718916E-05 •718916E-05	CYCLES • 10000006 • 10000006 • 2337878 • 233788 • 233788 • 100000 • 1430501 • 1599264 • 15092544 • 1100000 • 1509264 • 11000000 • 46704942 • 46704 • 47044 • 470444 • 47044 • 470444 • 470444 • 4704444 • 4704444 • 470
<ul> <li>58920931E+03</li> <li>91708325E+03</li> <li>91708325E+04</li> <li>50047169E+04</li> <li>50047169E+03</li> <li>53855054831E+03</li> <li>13025831E+03</li> <li>28748314E+03</li> <li>28748314E+03</li> <li>75464468E+03</li> <li>48342514E+03</li> </ul>	. 55335542E403 . 32505652E403 . 32505652E403 . 47458264E403 . 47458247E403 . 90043347E403 . 556454544244 . 151225831474E403 . 14337171E405 . 14337171E405		
8 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2		M M M M M M M M M M M M M M M M M M M	
	- 8.79 -43.05 -239.16 -239.96 -16.42 -16.42 -10.97 -10.97 -10.97 -10.97 -10.00		
902 902 902 902 902 902 902 902 902 902		STRAIN 0593 0321 0105 0365 0156 0063	
		LASTIC 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	SIGMIN -54.99 -16.42 17.35 25.40 25.40 11.89 6.44 9.64 -23.85
- 15,29 15,20 42,84 42,84 42,84 55,00 55,00 55,00 55,00 55,00 22,23,29 24,55 24,55 24,55 24,55 24,55 25,11 25,01 25,01 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,00 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,000 25,0000 25,0000 25,0000000000	49.68 49.68 49.68 66.23 66.23 24.83 24.39 24.39 20.150 20.150 20.150 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.53 21.	_	SIGMAX -40.58 55.00 29.46 31.73 31.73 31.73 31.73 31.73 31.73 31.73 31.73 31.73 31.73 31.73
RELAXATION RELAXATION 14.945 14.946 14.946 2.99 14.14 2.99 2.6.09 14.14 2.2.75 2.3.75 2.3.75 2.3.75 2.3.75 2.3.75 2.3.75 2.3.98 2.3.67 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 8.90 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 14.18 1	14.18 14.18 14.18 12.97 12.97 12.97 12.97 12.98 12.98 12.98 12.98 12.98 12.98 12.98 12.90 12.90 12.90 12.90 12.90 12.90 12.90 12.90 12.90 RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION RELAXATION R	S 27 27 29 29 29 29 29 29 29 29 20 20 20 20 20 20 20 20 20 20 20 20 20	

		Weight the state of the state o																																																	-
.10030000€-08 .3000000€-08	.1000000c-03	<ul> <li>1000000E-03</li> </ul>	. 384 25383E-04	•10207000E-07	. 19319990C-U9 .36364588E-A8	.15148142E-07	• 50030308E-08	•>UUUUUUE-08 •\$00000005-08		• 5000000E • 08	- 3000000 E - 08	<ul> <li>5000000000</li> </ul>	• 5000000E-03		.10000000F-08	• 4000000E-08	•400)0000E+08	• + 0 0 0 0 0 0 0 E - 0 8	• + 0 0 0 0 0 0 E = 0 8	.4000000F-08	•4000000E-08	• 4 0 0 3 0 0 0 0 E - 0 8	•+001000E-08	• + 0 0 J 0 J 0 0 0 E - 08	• + 10061365 - 04	.14000000E-07	.157364616-04	.100J0000E-03	.3/808599E-09 278763385-90	.37873361E-09	.378836256-09	.37905745E-09	.37985421F-09	•38042931E-09	<ul> <li>58112208€-09</li> </ul>	.19096626E-09	• 200034195-08 207053551-44	•64449697E-10	• + 1 + 0 + 3 + 4 8 - 0 +	.63234078E-09	• 664 96853£ - 07 - 1 27 752205 - 07	.129151626-07	• 445513792-09	• +4248601E-09	•43968484E-09	•43710563E-09	<ul> <li>         + 5,4,7,4,4,0,5,4,0,4,0,0,4,0,0,0,0,0,0,0,0,0,0</li></ul>	.43065799E-09	•42892630E-09	•42739783E-09	
• 10030000±+10 • 10030006±+10	<ul> <li>10000005 +10</li> </ul>	<ul> <li>100303005 + 10</li> </ul>	• 26004102E+05	•9/9/19825+08 .6573474525405	.1924949úč+10	.66014698€+08	• 100000005 + 10 • 000000005 - 10	• 100000007+10		<ul> <li>1000000000000000000000000000000000000</li></ul>	4 10000000 ÷ 10	• 100303005 + 10	• 100000005+10 • 00000005 • • 0	• 10000001410	• 10000001 + 10	<ul> <li>10000005+10</li> </ul>	• 100 J0 J0 00 0 5 + 10	• 100000015 + 10 • 000000015 + 10	• 100000000 + 10	• 100000005 + 10	• 100 J000 0E + 10	• 10000000±-10	<ul> <li>100300005+10</li> </ul>	• 10000005+10	<ul> <li>2463659446405</li> <li>40655564</li> <li>40655564</li> </ul>	• 100000005+05	• 835466961405	• 10000000E+05	- 52/ 35 32 85 + 10 - 528 352 8 35 + 10	• 52807566E+10	.527332595+10	• 52752450E+10 cartes:	• 26/126486 +10 • 526517795 +10	• 52572184E+10	<ul> <li>524766245+10</li> </ul>	• 52365271E+10	• 31 234 I2 45 + 04 • 48 3 7 25 4 5 + 4 2	.155159775+11	<ul> <li>24152055E+05</li> </ul>	• 15814258E+10	• 1503335555955 • 313105155+09	.774283725+08	• 67337984£+10	.67798754E+10	• 68230691E+10	• 68633296E+10 60006440740	- 693487555+10 - 693487555+10	• 696018465410	• 69942085E+10	<ul> <li>70192214E+10</li> </ul>	
																																														• :					
5 °	1	•			- <u>-</u> _	<b>.</b>				- <b>S</b>	5.	۰ <b>۵</b>	÷ 5	F		÷.	• •	• •	 E t	t.	÷.	• •	• .† .	• •	10.	1.	1.	• • •	• •		<b>5</b>	•••	5°	5.	2•	<b>.</b>		-	1.		• • •		ň	ч т	• •	• •	• • • •	• m	3.	з <b>.</b>	
	3.97	.7.49	5.52 2.35	3.52	1.70	-9-93		ŭ.22	.9.87	9.52	.9.17	. 6. 83 5. 5	8.14	1.30	.8.14	9.87	9.77	9. DS	9.48	9.38	9.29	9.19	9.09	9° U 0	1.28	5.00	0.02	5.00 2.54	2.83	3.11	3.39	3.06	4.21	4.48	4.75	2.52	5.01	7.52	1.46		5.14	31	9.58	9.95	.0.53		1.42	1.78	2.13	2.48	r :
- 62.3	6.79	-6.70 -	40.83 -5	35.22 -1	11.74 -	17.64		-2.31 -2	-1.96 -1	-1.60 -1	-1.26 -1	- 16 <b>.</b>		23 -2	-4.54 -1	-16.07 -2	-15.97 -2	- 15-88 - 4	-15.68 -2	-15.58 -2	-15.49 -2	-15.39 -2	-15.29 -2	- T2.5U -5	1.72 -2	60.71 -5	55.60 2	55.00 -5 23.44 1	23.72	24.00 1	24.28 1	24.56	25.11 1	25.37 1	25.64 1	25.91 I	15.14 1	15.14	49.68 -1	34 403 5	24.39	24.39	19.75	20.12	20.96	21.23	21.59 1	21.94 1	22.30 1	22,65	- 1111 A.S.

DAMAGE FROM PLASTIC STRAINS= .30462764E-03 DAMAGE FROM PLASTIC STRAINS= .22665690E-02 DAMAGE FROM PLASTIC STRAINS= .30462764E-03 DAMAGE FROM PLASTIC STRAINS= .30462764E-03 DAMAGE PER THIS SET= .83070929E-03 DAMAGE PËR THIS SET= .12645622E-02 DAMAGE PER THIS SET= .83070929E-03 DAMAGE PER THIS SET= .26509046E-02 DAMAGE PER THIS SET= .83070929E-03 TOTAL ENN/CYC =, .37115251E-02 TOTAL ENN/CYC =, .45422354E-02 TOTAL ENN/CYC =, .71931400E-02 TOTAL ENN/CYC =, .802384936-02 TOTAL ENN/CYC =, .88545586E-02 ŝ ڡ 4 FLIGHT OR BLOCK NO. FLIGHT OR BLOCK NO. FLIGHT OR BLOCK NO. FLIGHT OR BLOCK NO.

$\begin{array}{cccccccccccccccccccccccccccccccccccc$	
-0.0179 $-33210$ $64.283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46283$ $-46284$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6844$ $-6$	-1.836 10000. Ayerican 4 2219
-11243       -51243       -51243       -51243       -51243       -51243 $7.5^{-}$ $7$ $4^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $7.5^{-}$ $7$ $4^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $7.5^{-}$ $7^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $7.7^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$ $1^{-}$	AMERICAN 5 2219 American 6 2219
(.5, -) $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$ $(.5, -)$	AMERICAN 7 2219
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	
4       1 $534$ $7$ 1 $534$ $7$ 1 $534$ $7$ 1 $534$ $7$ 1 $535$ $11$ 1 $535$ $12$ $1335$ $563$ $11$ 1 $573$ $12$ $133$ $563$ $13$ $1335$ $553$ $13$ $1335$ $354$ $13$ $1335$ $354$ $13$ $1335$ $354$ $13$ $316$ $357$ $13$ $316$ $357$ $13$ $12$ $315$ $13$ $131$ $354$ $13$ $12$ $377$ $27$ $316$ $577$ $28$ $112$ $572$ $28$ $121$ $127$ $28$ $121$ $127$ $28$ $121$ $577$ $28$ $122$ $512$ $28$ $121$ $577$ $28$ <td< td=""><td></td></td<>	
$ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 &$	
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	1.
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	2
$ \begin{bmatrix} 1 & 1 & 529 & 677 \\ 11 & 1 & 434 & 593 \\ 12 & 1 & 364 & 564 \\ 13 & 1 & 364 & 564 \\ 15 & 1 & 307 & 556 \\ 17 & 1 & 307 & 556 \\ 17 & 1 & 377 & 556 \\ 17 & 1 & 374 & 512 \\ 27 & 1 & 127 & 512 \\ 27 & 1 & 127 & 512 \\ 27 & 1 & 127 & 512 \\ 27 & 1 & 127 & 512 \\ 28 & 1 & 127 & 512 \\ 27 & 1 & 127 & 514 \\ 28 & 1 & 127 & 514 \\ 28 & 1 & 127 & 514 \\ 27 & 1 & 278 & 544 \\ 27 & 1 & 278 & 544 \\ 27 & 1 & 127 & 512 \\ 28 & 1 & 128 & 544 \\ 27 & 1 & 278 & 544 \\ 27 & 1 & 278 & 544 \\ 27 & 1 & 278 & 544 \\ 27 & 1 & 278 & 544 \\ 27 & 1 & 278 & 544 \\ 27 & 1 & 278 & 544 \\ 27 & 1 & 127 & 512 \\ 37 & 1 & 278 & 544 \\ 27 & 1 & 127 & 512 \\ 37 & 1 & 278 & 544 \\ 37 & 1 & 278 & 544 \\ 37 & 1 & 278 & 544 \\ 37 & 1 & 278 & 544 \\ 37 & 1 & 577 & 617 \\ 35 & 1 & 495 & 577 \\ 35 & 1 & 495 & 577 \\ 35 & 1 & 495 & 577 \\ 36 & 1 & 322 & 617 \\ 36 & 1 & 322 & 647 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 577 & 517 & 517 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 544 & 577 & 617 \\ 37 & 577 & 577 & 617 \\ 37 & 577 & 577 & 617 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 & 577 & 577 & 577 \\ 37 &$	3
$ \begin{bmatrix} 12 & 1 & 364 \\ 13 & 13 & 133 \\ 14 & 1 & 364 \\ 15 & 1 & 364 \\ 17 & 1 & 364 \\ 18 & 2 & 364 \\ 18 & 2 & 545 \\ 19 & 2 & 545 \\ 27 & 1 & 127 \\ 27 & 1 & 127 \\ 27 & 1 & 127 \\ 28 & 377 \\ 26 & 1 & 113 \\ 26 & 1 & 113 \\ 27 & 274 \\ 27 & 28 & 377 \\ 28 & 3 & 128 \\ 28 & 3 & 284 \\ 21 & 111 & 577 \\ 28 & 377 \\ 28 & 3 & 128 \\ 28 & 3 & 723 \\ 37 & 11 & 278 \\ 28 & 377 \\ 28 & 1 & 128 \\ 27 & 274 \\ 28 & 3 & 128 \\ 28 & 3 & 747 \\ 37 & 128 \\ 28 & 3 & 747 \\ 37 & 128 \\ 28 & 3 & 747 \\ 37 & 128 \\ 28 & 3 & 747 \\ 37 & 128 \\ 28 & 3 & 748 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 & 128 \\ 37 $	2.
$ \begin{bmatrix} 13 & 1 & 133 \\ 14 & 1 & 364 \\ 15 & 1 & 307 \\ 16 & 1 & 307 \\ 17 & 1 & 316 \\ 18 & 2 & 545 \\ 17 & 1 & 316 \\ 18 & 2 & 546 \\ 27 & 1 & 127 \\ 27 & 1 & 127 \\ 27 & 1 & 127 \\ 27 & 1 & 127 \\ 28 & 1 & 128 \\ 26 & 1 & 111 \\ 28 & 244 \\ 244 & 237 \\ 27 & 278 \\ 27 & 1 & 128 \\ 27 & 278 \\ 28 & 3 & 577 \\ 28 & 1 & 128 \\ 27 & 218 \\ 28 & 3 & 577 \\ 28 & 3 & 577 \\ 28 & 1 & 128 \\ 27 & 218 \\ 28 & 3 & 577 \\ 28 & 1 & 218 \\ 28 & 2 & -128 \\ 31 & 128 \\ 27 & 218 \\ 28 & 2 & -128 \\ 33 & 1 & -149 \\ 28 & 2 & -128 \\ 33 & 1 & -2149 \\ 28 & 2 & -128 \\ 33 & 1 & -2149 \\ 33 & 1 & -2149 \\ 33 & 1 & -2149 \\ 33 & 1 & -2149 \\ 33 & 1 & -2149 \\ 33 & 1 & -2149 \\ 33 & 1 & -2149 \\ 33 & 1 & -2149 \\ 33 & 1 & -2149 \\ 33 & 1 & -2149 \\ 34 & 1 & 331 \\ 35 & 1 & -2149 \\ 36 & 1 & -27 \\ 37 & 1 & -2149 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -647 \\ 38 & -6$	
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	<b>1</b> . 1
$ \begin{bmatrix} 16 & 1 & 307 \\ 17 & 1 & 316 \\ 18 & 2 & 545 \\ 27 & 1 & 127 \\ 21 & 127 & 512 \\ 22 & 347 & 512 \\ 22 & 377 & 512 \\ 22 & 1 & 127 \\ 23 & 1 & 127 \\ 25 & 1 & 133 \\ 25 & 1 & 111 \\ 27 & 278 & 274 \\ 28 & 3 & 723 \\ 31 & 106 & 244 \\ 21 & 111 & 557 \\ 28 & 3 & 723 \\ 31 & 106 & 244 \\ 21 & 111 & 527 \\ 31 & 106 & 244 \\ 272 & 340 & 340 \\ 31 & 111 & 527 \\ 32 & 1 & 128 \\ 33 & 1 & 527 \\ 33 & 1 & 527 \\ 34 & 1 & 331 \\ 35 & 1 & 514 \\ 31 & 1 & 517 \\ 35 & 1 & 527 \\ 35 & 1 & 537 \\ 36 & 1 & 331 \\ 31 & 1 & 527 \\ 35 & 1 & 527 \\ 35 & 1 & 527 \\ 36 & 1 & 331 \\ 36 & 1 & 331 \\ 37 & 527 & 517 \\ 37 & 1 & 527 \\ 37 & 1 & 527 \\ 37 & 1 & 527 \\ 37 & 1 & 527 \\ 37 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 37 & 1 & 527 \\ 37 & 1 & 527 \\ 37 & 1 & 527 \\ 37 & 1 & 527 \\ 37 & 1 & 527 \\ 37 & 1 & 527 \\ 37 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 37 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 527 \\ 38 & 1 & 5$	6 •
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	J.
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	1. -
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	рание и простоятие и В с
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	35.
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	. Å •
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	10.
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	
31     1    148    029       37     1     .557     .846       33     1     .529     .845       34     1     .331     .523       35     1     .657     .845       36     1     .527     .845       35     1     .527     .845       35     1     .527     .816       37     1     .657     .817       37     1     .464     .527       39     1     .322     .811       37     1     .531     .567       37     1     .322     .792       39     1     .322     .792       40     1     .322     .792       41     1     .325     .449	1
37     1     .557     .846       33     1     .529     .881       34     .331     .529     .881       35     1     .521     .517       35     1     .454     .527       37     1     .527     .517       35     1     .454     .527       37     1     .527     .527       37     1     .521     .527       37     1     .532     .637       37     1     .532     .649       40     1     .335     .449	. L 0
34     1     .331     .577       35     1     .527     .577       36     1     .527     .617       37     1     .464     .527       38     .527     .617     .617       36     1     .642     .527       39     1     .691     .647       39     1     .527     .617       39     1     .691     .649       40     1     .322     .792       41     1     .325     .792	•
35     1     .527     .617       36     1     .464     .527       37     1     .432     .617       38     .527     .527       39     1     .532       39     1     .522       31     .532     .792       40     1     .325       40     1     .325	
36     1     .464     .527       32     1     .432     .81       33     1     .581     .637       34     1     .522     .792       40     1     .335     .449       41     .335     .449	19.
3     1     -531     -647       39     1     -532     -649       40     1     -322     -449       61     -335     -449	2.
39         1         322         792           40         1         355         449           41         35         50	
4C 1 .335 .449	.1.
	9. 1.
42 1 .461 .545	29.
43 1	

CDC 6600 FTN V3.0-367A OPT=1 03/26/74 19.09.02. PAGE 2										
	1 2 ε.	5.0 5 4 1 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2								

# APPENDIX III

LIST OF COMPUTER PROGRAM SYMBOLS AND DEFINITIONS

Coefficient of the  $x^2$  term in the equation of a line on Α a constant life fatigue diagram where minimum stress is x and maximum stress is y.  $(R = Ax^2 + Bx + C - y)$ AA An assigned value of +1. or -1. AAA A stress used in the calculation of plastic strain. The absolute value of DIF. ABDIF The absolute value of ASMAX or of ASMIN, as assigned. ABM The absolute value of ASMAX. ABMAX The absolute value of ASMEAN. ABMEAN The absolute value of ASMIN. ABMIN ABR4 The absolute value of R(4). ABR7 The absolute value of R(7). The name of a routine calling for the absolute value ABS of a quantity. AKT Stress concentration factor, K₊ The product (AKT) (STMAX) ASMAX The quantity (ASMAX + ASMIN)/2ASMEAN The product (AKT) (STMIN) ASMIN Average value of SIGMIN over an interval. AVSGMN AVSGMX Average value of SIGMAX over an interval.

B Coefficient of the x term. (See A.)

BBB A stress used in the calculation of plastic strain.

C The constant. (See A.)

COFMAN Inverse of the Coffin-Manson slope.

CYCINT The number of cycles in an interval.

- CYCLES The calculated number of cycles expected to be indicated on a constant life fatigue diagram for the applied combination of maximum and minimum stress.
- Cl The Residual Stress Relaxation Constant (See ENEP.)
- DAM Damage.
- DECK Decimal or real value of integer K after conversion.
- DEL2 A portion of a least-squares-method solution.
- DIF The difference between residual stress and equilibrium residual stress. (RES(I) EQRES)
- DO2 A portion of a least-squares-method solution.
- DUMMY A variable used in the calculation of the number of cycles to be considered as an interval for relaxation determination.
- ELMOD The elastic modulus.
- EN The number of cycles from the beginning of the relaxation process to the end of the current interval.
- ENEP The number of cycles required for overload residual stress effect to return to within one-tenth of its original difference from equilibrium conditions.

 $(N_{ep} = C1/(ABM)^{E1}(ABMEAN)^{E2})$ 

- ENN The number of applied cycles at a load level.
- ENNCYC The ratio of the number of applied cycles to the number of cycles to failure. ( ENN/CYCLES )

EPSD LCF strain intercept.

EQRES Equilibrium residual stress.

EX An exponential function depicting the relaxation of residual stress.

EXP The name of a routine calling for the exponential value of a quantity.

EXPO An exponent. The power of 10 which indicates the number of cycles to failure.

- E1 Residual stress Relaxation Exponents.
- FLOAT The name of a routine calling for integer-to-real conversion.
  I A variable subscript.
- IBLOCK The identifying number of a block the blocks being numbered consecutively from 1 to NBLOCK.
- IFIX The name of a routine calling for real-to-integer conversion.

IN The number of steps input to the range pair counting subroutine.

IPRINT Value controlling the WRITE statements.

IRAIN A counter.

IRPCM Value controlling entry into the range pair counting subroutine.

- ISTEP The identifying step number, the steps being numbered from 1 to NLEVEL.
- ITYPE The identifying type number, the types being numbered from 1 to NTYPE.
- J A variable subscript.
- JA Value of +1 or 0, as assigned for branch determination.

JB Value of -1 or 0, as assigned for branch determination.

- JJ An index variable.
- JJJ An index variable.
- JKL An index variable.
- K An index variable.
- KK An index variable.
- KPMAX The number of steps output from the range pair counting subroutine.
- L An index variable.
- LMN An index variable.
- M An index variable.
- N An index variable with values of N=4-7 indicating the power of 10, and thus identifying a particular life cycle curve.
- NBLOCK The total number of times to execute a block of loads.
- NDECK The number of data decks to be run sequentially.
- NFLAG An integer used as a counter.
- NFLAG2 An integer used as a counter.

NLEVELThe total number of steps, or levels, of loads in a block.NNA subscripted variable used to indicate which types of

loads are experienced in which blocks.

NTYPE The total number of different types.

PLSTRA Plastic strain.

R Residue term in damage calculation.

RES Residual stress.

RNCYC The number of cycles for a level after exitting the range pair counting subroutine.

SIGMAX Maximum stress.

SIGMIN Minimum stress.

STMAX Maximum applied stress.

STMIN Minimum applied stress.

SUMDEL Summation of damage for a flight.

SUMENN Accumulated total of applied cycles. (Summation of ENN).

SUMNC Accumulated cycle ratio. (Summation of ENN/CYCLES).

SUMR Summation of R(N), N=4,7.

SUMRN Summation of NR(N), N=4,7.

SUMR2 Summation of  $R(N)^2$ , N=4,7.

SUMR2N Summation of NR(N)², N=4,7.

SUMR3 Summation of  $R(N)^3$ , N=4,7.

SUMR4 Summation of  $R(N)^4$ , N=4,7.

TITLE1, TITLE2 Identification of the source of the SN data.

TLL Tensile load limit.

TM1, TM2 Material type.

TTYS One-fifth of tensile yield stress.

TYS Tensile yield stress.

T1,T2,T3,T4,T5,T6,T7,T8 Test identifying information.

- X Variable equivalent to SIGMIN.
- Y Variable equivalent to SIGMAX.

## APPENDIX IV

# FATIGUE LIFE INPUT DATA FOR SEVERAL MATERIALS

	YIELD	STRAIN	INVERSE	LIFĘ,	S-N LI	FE COEFH	ICIENTS
MATERIAL	STRESS	INTERCEPT	OF SLOPE	101	A(I)	B(I)	C(I)
2024-74	58.	0.4(2)	-1,836(2)	4	0020	. 2091	62.63
2024 14	50.	0.4 V	1.000	5	0032	.4366	51.4
				6	0035	.6207	42.2
				7	0042	.7003	36.1
2210 0051		~ (D	1 070	,	0000	2207	FF 4
2219-1851	22.	0.4 🕑	-1.030	4	0022	.2204	55.0 <del>.</del>
				5	0018	.3320	48.3
				6	0015	.4628	39.7
			_	/	0024	.6420	31.7
7075 <b>-</b> T6	72.	0.4 (L)	-1.836	4	0020	.2801	71.73
	• = -			5	0022	.5154	56.3
				6	0014	.6141	44.6
				7	0013	.6838	38.1
		(C)					
RQC-100	125.	0.54	-1.493	4	0.0	.2136	98.35
				5	0.0	.2927	88.5
				6	0.0	.3669	79.1
				7	0.0	.4376	70.3
		5	5				5
Man-Ten	55.	1.119	-1.667	4	0.0	.2257	63.5
				5	0.0	.3520	53.1
				6	0.0	.4669	43.7
				7	0.0	.5678	35.4
4240 Ctool	160	o (2)	1 022	1.	0000	2567	162 (3)
4340 SLEET	100.	0.4 🗢	-1,0000	4	0002	•2J07 59/9	102.4
				5	- 0007	•J240 5557	112 5
				7	- 0005		108 5
		<b>^</b>	~	'	0005		100.7
Ti-6-4	158.	0.4 (2)	-1.836(2)	4	0009	.2368	154.2
				5	0006	.4640	110.3
				6	0000	.4650	88.9
				7	.0001	.4752	84.2

(1)Data from Endo, T., and Morrow, J., NAEC-ASL-1105, Naval Air Engineering Center, Philadelphia, PA, June 1966.

Data not available - Source 1 data considered typical.

2 3 Derived from Metallic Materials and Elements for Aerospace Vehicles Structures, MIL-HDBK-5A, Dept. of Defense, Washington, D.C., February 1966. (4) (5) Information supplied from Rockwell International.

Information supplied by Society of Automotive Engineers Cumulative Damage Division, Courtesy of Mr. H.R. Jaekel.

*U.S.Government Printing Office: 1974 - 758-435/645