

分置

オブジェクト指向強度評価プログラム/  
PARTS-DSの開発

(1) Visual Basic によるプロトタイプを作成

1995年8月

動力炉・核燃料開発事業団  
大洗工学センター

複製又はこの資料の入手については、下記にお問い合わせください。

〒311-13 茨城県東茨城郡大洗町成田町4002

動力炉・核燃料開発事業団

大洗工学センター システム開発推進部・技術管理室

Enquires about copyright and reproduction should be addressed to: Technology Management Section O-arai Engineering Center, Power Reactor and Nuclear Fuel Development Corporation 4002 Narita-cho, O-arai-machi, Higashi-Ibaraki, Ibaraki-ken, 311-13, Japan

動力炉・核燃料開発事業団 (Power Reactor and Nuclear Fuel Development Corporation)

## オブジェクト指向強度評価プログラム/PARTS-D Sの開発

### (1) Visual Basicによるプロトタイプ作成

笠原 直人\*、細貝 広視\*\*

#### 要 旨

損傷計算プログラムは、構造解析結果と試験（運転）条件を入力データとして与えることでクリープ疲労損傷計算処理を行うプログラムである。

計算モジュールはVisual Basicで記述し、データ入出力モジュールは、表計算ソフトMS-EXCELを使用した。

MS-EXCELと計算モジュールのデータ転送はファイルを介さず全てプロセス間通信によって行われる。

プログラム言語は、Windows環境で動作する様にVisual Basic言語を使用した。

本プログラムは、単独での実行の他「過渡熱応力リアルタイムシュミレータPARTS」および「構造物強度データベースシステムSTAR」との連携動作も可能である。

開発作業項目は以下の通りである。

- (1) 損傷計算用基本変数をモジュール間で最適に受け渡すためのパラメータ(構造体)設計
- (2) プロセス間通信(DDE)のプログラム組み込み、動作試験
- (3) 材料特性値算出ライブラリ(FORTRANプログラム)の動的リンクライブラリ(DLL)化
- (4) FORTRANソースコードのVisual Basicコードへの移植
- (5) STARシステムとの連結動作試験

本プログラムは今後、C++を使用して本格的なオブジェクト指向プログラムに移行する予定である。

---

\* 大洗工学センター 基盤技術開発部 構造・材料技術開発室  
\*\* 常陽産業株式会社

## 目 次

1.	緒 言 .....	1
2.	システムの概要 .....	4
2. 1	動作環境 .....	4
2. 2	基本データの流れ .....	6
2. 3	ファイル一覧 .....	6
3.	MS - EXCEL 環境 .....	8
3. 1	機能概要 .....	8
3. 2	入出力仕様 .....	8
3. 2. 1	入力ファイル .....	8
3. 2. 1. 1	試験条件シート .....	8
3. 2. 1. 2	解析結果シート .....	13
3. 2. 2	出力ファイル .....	13
3. 2. 2. 1	損傷計算結果シート .....	13
4.	Visual Basic 環境 .....	19
4. 1	機能概要 .....	19
4. 2	プログラム構成 .....	22
4. 3	入出力仕様 .....	22
4. 4	基本変数ライブラリの項目 .....	23
4. 5	構造体変数 .....	26
4. 5. 1	構造体とは .....	31
4. 6	損傷計算で使した計算式 .....	32

4. 7	モジュール説明	42
5.	F O R T R A N環境	95
5. 1	機能概要	95
5. 2	入出力仕様	95
5. 3	モジュール説明	96
6.	プロセス間通信 (DDE)	117
6. 1	DDEとは	117
6. 2	基本的仕組み	117
6. 2. 1	アプリケーション名	118
6. 2. 2	トピック	118
6. 2. 3	アイテム	119
6. 2. 4	通信でのリンクの種類	119
6. 3	具体的操作	120
7.	動的リンクライブラリ (D L L) の呼び出し	126
7. 1	動的リンクライブラリ (D L L) とは	126
7. 2	インポートライブラリを使わないリンク	128
7. 3	ダイナミックリンクの利点	128
7. 4	複数言語によるプログラミング (M u l t i L a n g u a g e)	132
7. 5	D L L プロシージャの呼び出し	133
7. 5. 1	概要	133
7. 5. 2	D L L プロシージャ呼び出しの特殊なデータ型の利用	136
8.	S T A R - N E T システムとの連結	141

8. 1 親プログラムを再起動させる方法	141
9. 操作方法	142
9. 1 環境設定	142
9. 2 起動方法	142
10. 結 言	144
謝 辞	146
参 考 文 献	147
付 録	149

## 表 リ ス ト

表 1. 1	試験条件ワークシート .....	10
表 1. 2	解析シート項目と構造体変数の対応一覧 .....	14
表 1. 3	損傷計算結果項目の詳細 .....	15
表 2. 1	評価法名称とキーワードの一覧 .....	20
表 2. 2	材料名称とキーワードの一覧 .....	21
表 3. 1	損傷計算用基本変数算出用ライブラリ .....	24
表 3. 2	試験条件シート用構造体 .....	27
表 3. 3	解析結果シート用構造体 .....	28
表 3. 4	基本変数要素用構造体 .....	28
表 3. 5	各種変数要素用構造体 .....	29
表 3. 6	損傷計算結果データ変数用構造体 .....	30
表 6. 1	DDE通信サンプル・コード .....	123
表 7. 1	DLLプロシージャ手続きファイルサンプル .....	138

## 図 リ ス ト

図 1. 1	PARTS-DSの過渡熱応力リアルタイムシミュレータPARTSとの連携 ..	3
図 1. 2	PARTS-DSの構造物強度データベースSTARとの連携 .....	3
図 2. 1	損傷計算プログラムモジュール構成図 .....	5
図 2. 2	損傷プログラムデータの流れ .....	7
図 4. 1	TTS-DSによる評価フロー .....	33
図 4. 2	BDSによる評価フロー .....	34
図 4. 3	BDS-MDによる評価フロー .....	35
図 4. 4	DDS-MDによる評価フロー .....	36
図 4. 5	DDSM-D-WELDによる評価フロー .....	37
図 4. 6	TYPE Iによる評価フロー .....	38
図 4. 7	TYPE IIによる評価フロー .....	39
図 4. 8	EQEFによる評価フロー .....	40
図 4. 9	EQEF-WELDによる評価フロー .....	41
図 3. 1	DDE通信リンクの種類 .....	121
図 3. 2	DDE通信によるセルの対応付け .....	122
図 3. 3	インポート、ライブラリを使わないリンク .....	130
図 3. 4	インポート、ライブラリを使ったリンク .....	131
図 3. 5	複数言語呼び出し .....	132



## 1 緒言

強度評価プログラム/PARTS-DS（以下損傷プログラムと呼ぶ）は、構造解析結果と試験（運転）条件を入力データとして与えることでクリープ疲労損傷計算処理を行うプログラムである。データの入出力は、表計算ソフトMS-EXCELを使用して行う。

MS-EXCELと損傷計算処理プログラムのデータ転送はファイルを介さず全てプロセス間通信によって行われる。本プログラムは、単独で実行するのみでなく「過渡熱応力リアルタイムシミュレータPARTS」（図1.1）および「構造物強度データベースシステムSTAR」（図1.2）とも連携して動作させる事が可能である。

本プログラムは、FORTRAN言語で書かれMS-DOS上で動作していたものをWindows環境でモジュール間のメッセージ通信により同期するようにVisual Basic言語で再構築したものである。

MS-DOS上で動作した損傷プログラムには以下の問題点があった。

STARシステムを例に説明する。

まず2つの大きな要因がある、

(1) 実行環境であるDOS/VパソコンがCPU等の高速化により著しく処理性能が上がった。

(2) OS (MS-Windows) がマルチタスク化した。

これにより親プログラムであるEXCELマクロからMS-DOS版損傷プログラムを呼び出すと、損傷プログラムが終了するのを待たずにEXCELマクロが処理を進め、親プログラムと子プログラムの同期がとれないためシステムが正常に動作しなくなる。

今後パソコンの高性能化・OSの完全なるマルチタスク化を考慮した場合以下の方策が最も適切であると判断した。

(1) 損傷プログラムをWindows対応にする。(Visual CあるいはVisual Basic等の言語でWindowsアプリケーションに書き換える)

(2) OS (Windows) が提供しているプロセスが同期をとるための命令を使用する。(プロセス間通信/DDE)

上記の対策を組み合わせることにより親プログラムと子プログラムの同期がとれない問題は解消され、更にMS-DOSプロンプトを使用しないためシステムの安定動作を確保することも出来た。

更にWindowsアプリケーション化するという事で次の技術も採用した。

(3) 共通モジュールは、動的リンクライブラリ(DLL)化する。

これは、DLLというライブラリ形式のファイルにしておき、実行時必要なライブラリのみ主記憶に呼び出し使用する。そのため主記憶の効率的使用が可能となる。ま

た、プログラム開発も独立して行うためプログラムの汎用性及び独立性を高めることが出来る。

今まで損傷プログラムを実行するには、STARを起動しなければならなかった。単独利用のニーズから、表計算ソフト（MS-EXCEL/マクロ無）と損傷プログラムの組み合わせのみで動作するように損傷プログラムを独立化させた。

本説明書では、損傷プログラムの移植に際しこれを構成するプログラム、ライブラリ、通信方式、ファイルについての仕様・操作方法について説明する。

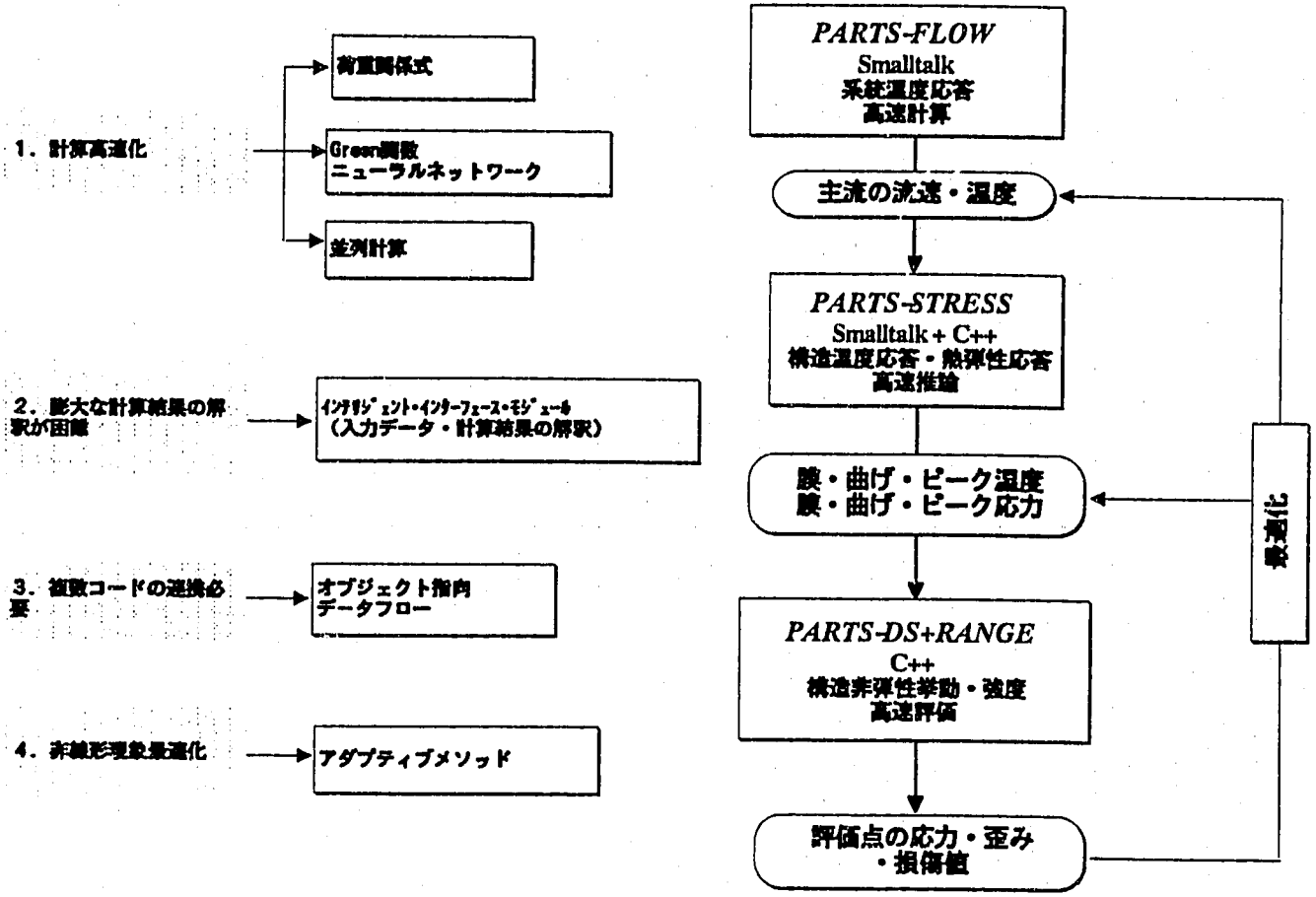


図1.1 PARTS-DSの過渡熱応力解析用シミュレータPARTSとの連携

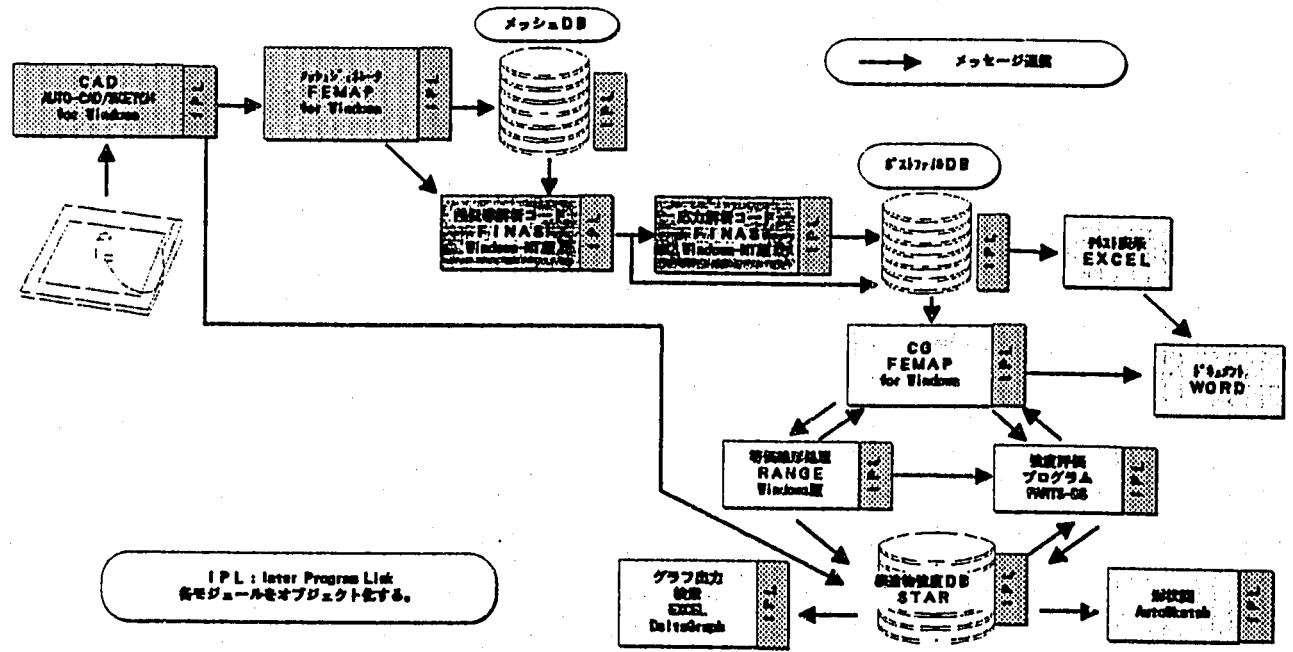


図1.2 PARTS-DSの構造物強度データベースSTARとの連携

## 2 システムの概要

### 2. 1 動作環境

本プログラムの主要部分はVisual Basic言語で記述されている。入出力データの受け渡し及び結果データのグラフ化は、表計算ソフト(MS-EXCEL)が行っている。更に共通性の高いモジュール群は、ライブラリ化してプログラムの汎用性・独立性を高めている。

図2. 1 にプログラム構成図を示す。

以下に本プログラムが動作する環境を示す。

#### <基本環境>

本体 . . . i 4 8 6 以上のCPUを搭載するパーソナルコンピュータ

OS . . . 日本語MS-DOS 5. 0 又は  
IBM DOS J 5. 0 / V 以上

+

日本語MS-Windows 3. 1 以上

メモリ . . . 8MB以上

ハードディスク . . . 100MB以上

ディスプレイ . . . Windowsに対応しているカラーディスプレイ

マウス . . . Windows対応マウス

※ 数値演算プロセッサが装着されていることが望ましい。

上記環境で以下に示すアプリケーションがインストールされていなければならない。

・MS-EXCEL 5. 0 . . . 表計算ソフト

・Visual Basic 2. 0 J . . .

Windows版Visual Basic言語

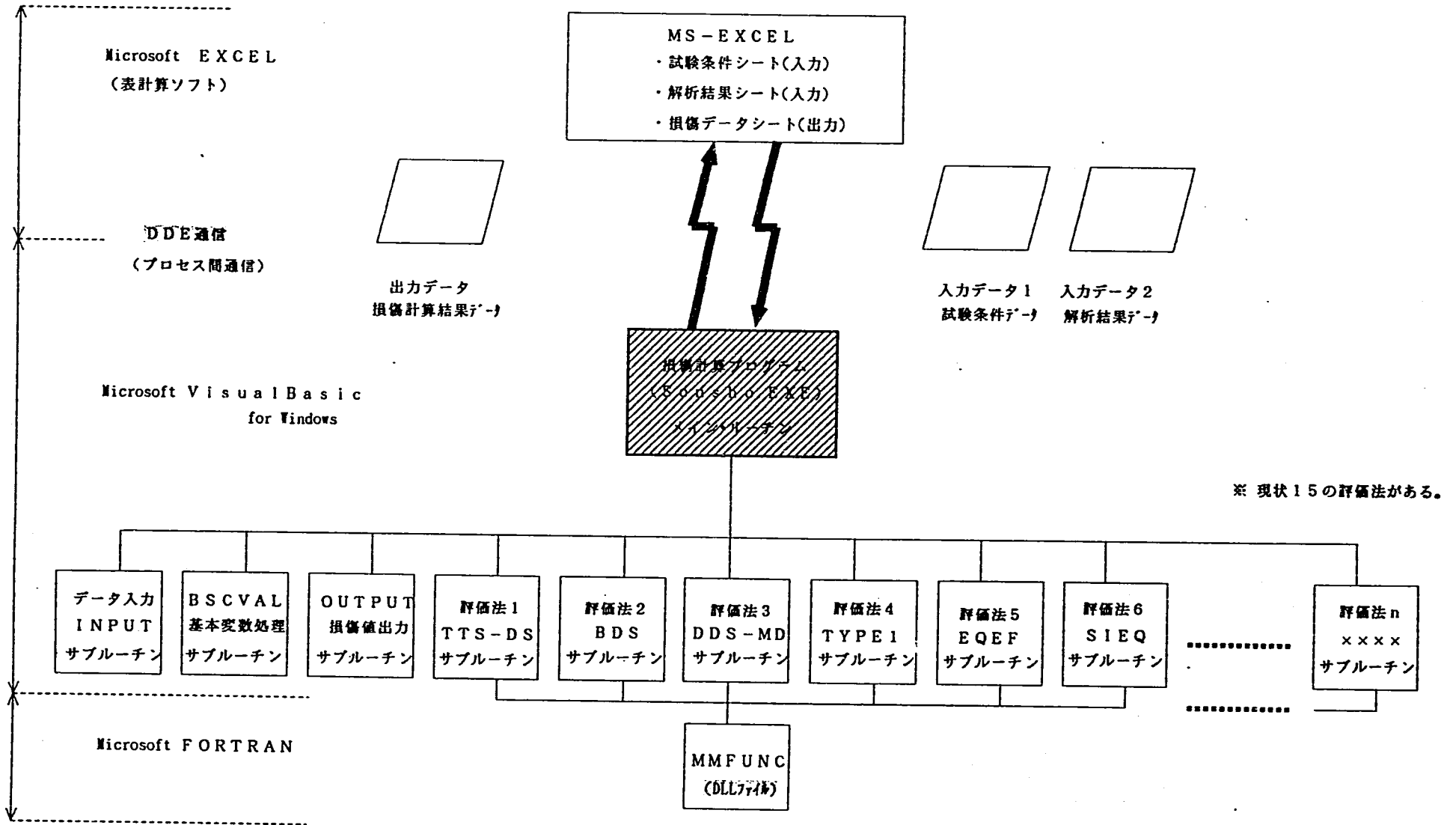


図2.1 損傷計算プログラムモジュール構成図

## 2. 2 基本データの流れ

図 2. 2 に損傷プログラムの基本的なデータの流れを示す。

## 2. 3 ファイル一覧

損傷プログラムで使用するシステムファイルの一覧とその概要を示す。

ファイル名	ディレクトリ	各ファイルの機能概要
Sonsho. bas	S:\Star\sonsho\vb版\	損傷計算プログラムソースコード
Sonsho. exe	S:\Star\sonsho\vb版\	損傷計算プログラム本体
Mmfunc. dll	S:\Star\sonsho\vb版\	材料特性値算出用ライブラリ
試験条件. CND	S:\Star\sonsho\vb版\	試験条件シート／入力用
解析. an?	S:\Star\sonsho\vb版\	解析結果シート／入力用
損傷値. d??	S:\Star\sonsho\vb版\	損傷計算結果シート／出力用
Vbrjp200. dll	S:\Star\sonsho\vb版\	VisualBasicプログラム用のライブラリ
Grid. vbx	S:\Star\sonsho\vb版\	VisualBasicプログラム用の コントロールプログラム
Oleclien. vbx	S:\Star\sonsho\vb版\	VisualBasicプログラム用の コントロールプログラム

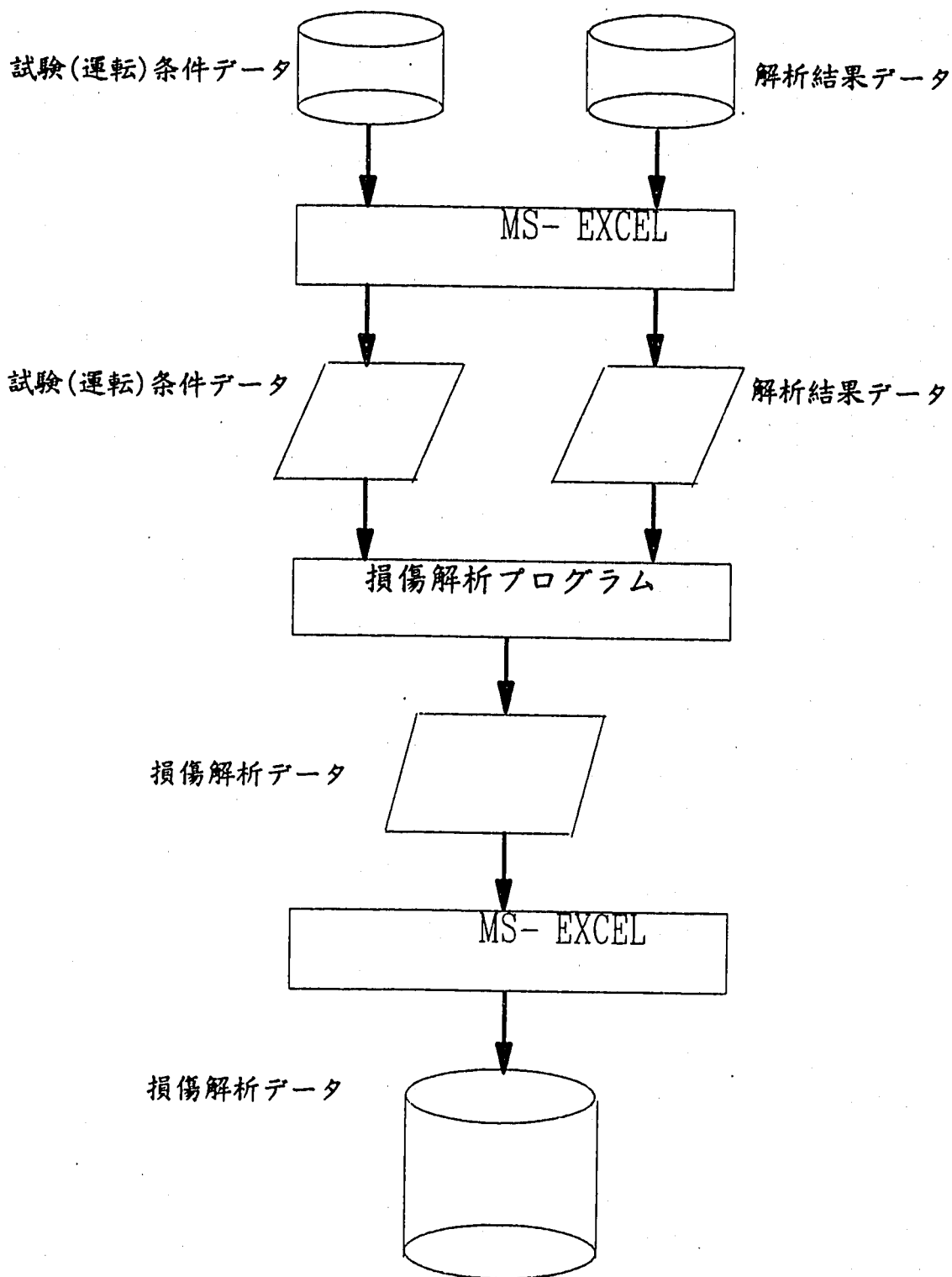


図2. 2 損傷プログラムデータの流れ

### 3 MS-EXCEL環境

#### 3.1 機能概要

本プログラムにおけるMS-EXCEL（表計算ソフト）の役割は、損傷プログラムに対する入出力データの受け渡し及び結果データのグラフ化を行うものである。データの受け渡しの方法はファイルを介さず全てプロセス間通信によって行われる。この通信方法はMS-Windows（OS）が提供する通信方式で通常独立したアプリケーション間で行われるデータの受け渡し方法である。プログラム内でのサブルーチンのパラメータを受け渡すと同様の感覚で比較的自由にデータを受け渡す事が可能である。

※ プロセス間通信（DDE）とは  
本詳細説明書の「6. プロセス間通信（DDE）」を参照の事。

#### 3.2 入出力仕様

MS-EXCELが損傷プログラムと行うデータの入出力に関して説明する。MS-EXCELが扱う入出力ファイルは以下の通りである。

それぞれは、決められたフォーマットで区切られており、MS-EXCEL独自のフォーマット形式で格納されている（バイナリー形式）。

MS-EXCEL Ver 4. 0

- (1) 試験条件シート（入力）
- (2) 解析結果シート（入力）
- (3) 損傷計算結果シート（出力）

##### 3.2.1 入力ファイル

###### 3.2.1.1 試験条件シート

熱過渡強度試験を行うに際し必要な諸条件を格納したデータファイルである。表1. 1 試験条件シート参照。

このデータファイルは、STARシステムが管理・保持している。

損傷プログラムを実行するときは、STARシステム中の試験条件データファイルを使用する。ファイル・フォーマットはMS-EXCELの独自フォーマット（バイナリ形式）になっている、しかもMS-EXCELのバージョンによりフォーマット形式も若干異なる。

ファイル名は固定である。実行する形態によって拡張子が変わる。

- (1) STAR-NETシステムから呼び出す場合



ファイル名 : 試験条件. X J S

(2) 損傷解析プログラム単体で実行する場合

ファイル名 : 試験条件. C N D

表 1. 1 試験条件シートに実際データのサンプルを示す。  
各項目説明は以下の通りである。

試験条件データ											
日付	試験装置	試験体名	材料名	評価部位	試験条件						関連報告書
日付	試験装置	試験体名	MATERIAL	評価部位	COLD温度	HOT温度	C保持時間	H保持時間	機械荷重	サイクル	名前
					TEMPC	TEMPH	HOLDTIMEC	HOLDTIMEH	MECHANIC	CYCLE	名前
90/5/30	T T S	容器型 3 1	SUS316FR		250	600	1	2	0	1055	
	①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	

番号	関連データ	SMAT TAG N	評価法	解析シート			出力シート
番号	関連データ	SMAT TAG N	STANDARD	シート名	シート行	シート桁	シート名
			DDSMDWLD	解析. AN1	150	65	損傷値0. D51
			⑪	⑫	⑬	⑭	⑮

表 1. 1 試験条件ワークシート

## [項目説明]

- ① 試験装置 : 熱過渡強度試験が行われる装置の名称。  
例>T T S (Thermal Transient Test Facility for Structure)
- ② 試験体名 : 試験装置の中で実際に試験される部位の名称。  
例>容器型 3 1 6 モデル
- ③ 材料名 : 試験される部位の金属材質の名称。  
格納変数 : Cond. Material  
例>SUS316FR

鋼種には、以下の様なものがある。

キ-ワ-ト <sup>®</sup> : SUS304	鋼種 : SUS 3 0 4
キ-ワ-ト <sup>®</sup> : SUS316	鋼種 : SUS 3 0 1 6
キ-ワ-ト <sup>®</sup> : SCM V 4	鋼種 : S C M V 4
キ-ワ-ト <sup>®</sup> : STBA 24	鋼種 : S T B A 2 4
キ-ワ-ト <sup>®</sup> : MOD9CR-1MO	鋼種 : 改良 9 C R - 1 M O 鋼
キ-ワ-ト <sup>®</sup> : SUS316FR	鋼種 : S U S 3 1 6 F R

- ④ 評価部位 : 評価の対象となる部位の名称。  
例>

## 試験条件

- ⑤ COLD温度 : 格納変数 : Cond. Tempc  
例>250℃
- ⑥ HOT温度 : 格納変数 : Cond. Temp h  
例>600℃
- ⑦ COLD保持時間 :  
格納変数 : Cond. Hldtmc  
例>1時間
- ⑧ HOT保持時間 :  
格納変数 : Cond. Hldtmh  
例>2時間
- ⑨ 機械荷重 : 格納変数 : Cond. Mecha  
例>0ton

⑩ サイクル数 : 格納変数 : Cond. Cycle  
例>1055

⑪ 評価法 : 損傷計算を行なうための手法。  
格納変数 : Cond. MethodName  
例>DDSMDWLD

評価法には、以下の様なものがある。

キーワード : TTSDS	評価法 : T T S - D S
キーワード : BDS	評価法 : B D S
キーワード : BDSMD	評価法 : B D S - M D
キーワード : DDS	評価法 : D D S
キーワード : DDSMDWLD	評価法 : D D S M D - W E L D
キーワード : TYPE1	評価法 : T Y P E 1
キーワード : TYPE2	評価法 : T Y P E 2
キーワード : EQEF	評価法 : E Q E F
キーワード : EQEFWD	評価法 : E Q E F - W E L D
キーワード : PNC	評価法 : P N C
キーワード : EQEF15	評価法 : E Q E F 改良版
キーワード : EQEF167	評価法 : E Q E F 改良版(qL=1.67)
キーワード : INELAST	評価法 : I N E L A S T I C
キーワード : SIEQ167	評価法 : S I E Q 1 6 7
キーワード : EQEF167W	評価法 : E Q E F 1 6 7 - W E L D

⑫ シート名 : 入力データとなる「解析シート」の名称。  
格納変数 : Cond. AnalSheetName  
例>解析. AN1

⑬ シート行数 : 解析シートの行数。  
格納変数 : Cond. AnalSheetRow  
例>150行

⑭ シート桁数 : 解析シートの桁数。  
格納変数 : Cond. AnalSheetCol  
例>65桁

⑮ 出力シート名 : 損傷計算の結果データを格納するシート名。  
格納変数 : Cond. OutSheetName  
例>損傷値0. D51

### 3. 2. 1. 2 解析結果シート

大型計算機で動作するFINAS（汎用非線形構造解析システム）の数値解析により求められる応力やひずみに関するデータを納めたファイルである。

メッシュモデルや解析方法等の違いにより複数の解析ケースを持つ事が可能である。

このファイルもSTARシステムが管理・保持しているものである。損傷プログラムを実行するときは、STARシステム中の解析ケースに応じた解析結果シートファイルを使用する。ファイル・フォーマットはMS-EXCELの独自フォーマット（バイナリ形式）になっている、しかもMS-EXCELのバージョンによりフォーマット形式も若干異なる。

各項目の詳細は、表1. 2 解析結果シート に示す。

### 3. 2. 2 出力ファイル

#### 3. 2. 2. 1 損傷計算結果シート

損傷プログラムは、試験条件シート及び解析結果シートを入力データとして受け取り指定された評価法・材料に応じた損傷計算を行う。

その結果データをプロセス間通信（DDE）によりMS-EXCELのこのシートに格納する。

このファイルも基本的にSTARシステムが管理・保持するものである。

ファイル・フォーマットはMS-EXCELの独自フォーマット（バイナリ形式）になっている、しかもMS-EXCELのバージョンによりフォーマット形式も若干異なる。

更に評価法により項目の数が異なる。

各項目の詳細は、表1. 3 損傷計算結果シートに示す。

解析の種類		格納変数		備考
評価 断面 座標	起点	r 1 z 1 $\theta$ 1	Analy. R(I, 0) Analy. Z(I, 0) Analy. H(I, 0)	
	終点	r 2 z 2 $\theta$ 2	Analy. R(I, 1) Analy. Z(I, 1) Analy. H(I, 1)	損傷計算は「終点座標」 を必要とせず、メモリ節約 の意味からこの領域 は確保していない
最高金属温度			Analy. Tmpmax(I)	
時点 1	評価時刻		Analy. Ptime(I, 0)	
	表面温度 等価線形温度 平均温度		Analy. Temp(I, 0) Analy. Ltemp(I, 0) Analy. Mtemp(I, 0)	
	表面応力	$\sigma$ R $\sigma$ Z $\sigma$ $\theta$ $\tau$ RZ	Analy. Strs(I, 0, 0) Analy. Strs(I, 1, 0) Analy. Strs(I, 2, 0) Analy. Strs(I, 3, 0)	
	等価線形応力	$\sigma$ R $\sigma$ Z $\sigma$ $\theta$ $\tau$ RZ	Analy. Lstrs(I, 0, 0) Analy. Lstrs(I, 1, 0) Analy. Lstrs(I, 2, 0) Analy. Lstrs(I, 3, 0)	
	膜応力	$\sigma$ R $\sigma$ Z $\sigma$ $\theta$ $\tau$ RZ	Analy. Mstrs(I, 0, 0) Analy. Mstrs(I, 1, 0) Analy. Mstrs(I, 2, 0) Analy. Mstrs(I, 3, 0)	
	弾性歪	$\varepsilon$ R $\varepsilon$ Z $\varepsilon$ $\theta$ $\gamma$ RZ	Analy. Ee(I, 0, 0) Analy. Ee(I, 1, 0) Analy. Ee(I, 2, 0) Analy. Ee(I, 3, 0)	
	塑性歪	$\varepsilon$ R $\varepsilon$ Z $\varepsilon$ $\theta$ $\gamma$ RZ	Analy. Ep(I, 0, 0) Analy. Ep(I, 1, 0) Analy. Ep(I, 2, 0) Analy. Ep(I, 3, 0)	
	クリ-7°歪	$\varepsilon$ R $\varepsilon$ Z $\varepsilon$ $\theta$ $\gamma$ RZ	Analy. EC(I, 0, 0) Analy. EC(I, 1, 0) Analy. EC(I, 2, 0) Analy. EC(I, 3, 0)	
	時点 2	評価時刻		Analy. Ptime(I, 1)
以下時点 1 に同じ。				
時点 3	評価時刻			「時点3」は極希に使用 する程度であり、メモリ 節約の意味からこの 領域は確保していない
	以下時点 1 に同じ。 ただし、時点 3 は存在しない場合もある。			

表 1. 2 解析シート項目と構造体変数の対応一覧

評価法	項目	変数
座標	r	Analy. R(I, 0)
	z	Analy. Z(I, 0)
	$\theta$	Analy. H(I, 0)
TTS-DS	相当応力範囲表面応力	Bscval. Vms(I)
	相当歪範囲弾塑性クリープ歪範囲	Bscval. Eepceq(I)
	歪集中係数	Allmtd. Kep(I)
	熱ピーク歪	Allmtd. Et(I)
	D f	Allmtd. Df(I)
	D c	Allmtd. Dc(I)
	D	Allmtd. D(I)
BDS	表面応力	Allmtd. Spx(I)
	等価線形応力	Allmtd. Sn(I)
	応力集中係数	Allmtd. K(I)
	弾性追従係数	Allmtd. Qn(I)
	K e' 係数	Allmtd. Ked(I)
	NEUBER則	Allmtd. Neuber(I)
	歪集中係数	Allmtd. Ke(I)
	熱ピーク歪	Allmtd. Ef(I)
	全歪範囲	Allmtd. Et(I)
	D f	Allmtd. Df(I)
	D c	Allmtd. Dc(I)
	D c r	Allmtd. Dcr(I)
BDS-MD	表面応力	Allmtd. Spx(I)
	等価線形応力	Allmtd. Sn(I)
	応力集中係数	Allmtd. K(I)
	弾性追従係数	Allmtd. Qn(I)
	K e' 係数	Allmtd. Ked(I)
	NEUBER則	Allmtd. Neuber(I)
	歪集中係数	Allmtd. Ke(I)
	熱ピーク歪	Allmtd. Ef(I)
	全歪範囲	Allmtd. Et(I)
	初期応力	Allmtd. Si(I)
	弾性追従係数	Allmtd. QF(I)
	D f	Allmtd. Df(I)
	D c	Allmtd. Dc(I)
D c r	Allmtd. Dcr(I)	

DDS	表面応力	Allmtd. Spx(I)
	等価線形応力	Allmtd. Sn(I)
	応力集中係数	Allmtd. K(I)
	弾性追従係数	Allmtd. Qn(I)
	K e' 係数	Allmtd. Ked(I)
	NEUBER則	Allmtd. Neuber(I)
	歪集中係数	Allmtd. Ke(I)
	熱ピーク歪	Allmtd. Ef(I)
	全歪範囲	Allmtd. Et(I)
	初期応力	Allmtd. Si(I)
	弾性追従係数	Allmtd. QF(I)
	D f	Allmtd. Df(I)
	D c	Allmtd. Dc(I)
	D	Allmtd. D(I)
DDS-WELD	表面応力	Allmtd. Spx(I)
	等価線形応力	Allmtd. Sn(I)
	応力集中係数	Allmtd. K(I)
	弾性追従係数	Allmtd. Qn(I)
	K e' 係数	Allmtd. Ked(I)
	NEUBER則	Allmtd. Neuber(I)
	歪集中係数	Allmtd. Ke(I)
	熱ピーク歪	Allmtd. Ef(I)
	全歪範囲	Allmtd. Et(I)
	初期応力	Allmtd. Si(I)
	母材に対する弾性追従係数	Allmtd. Qbm(I)
	熔接金属に対する弾性追従係数	Allmtd. Qw(I)
	弾性追従係数	Allmtd. Qeff(I)
	弾性追従係数	Allmtd. QF(I)
	D f	Allmtd. Df(I)
	D c	Allmtd. Dc(I)
D	Allmtd. D(I)	
TYPE1/ TYPE2	表面応力	Allmtd. Spx(I)
	等価線形応力	Allmtd. Sn(I)
	応力集中係数	Allmtd. K(I)
	弾性追従係数	Allmtd. Qn(I)
	K e' 係数	Allmtd. Ked(I)
	NEUBER則	Allmtd. Neuber(I)
	歪集中係数	Allmtd. Ke(I)
	熱ピーク歪	Allmtd. Ef(I)



	全歪範囲	Allmtd. Et(I)
	初期応力	Allmtd. Si(I)
	弾性追従係数	Allmtd. Qeff(I)
	D f	Allmtd. Df(I)
	D c	Allmtd. Dc(I)
	D	Allmtd. D(I)
EQEF	表面応力	Allmtd. Spx(I)
	等価線形応力	Allmtd. Sn(I)
	応力集中係数	Allmtd. K(I)
	K e' 係数	Allmtd. Ked(I)
	歪集中係数	Allmtd. Ke(I)
	総体的歪集中係数	Allmtd. Kedg(I)
	局所的歪集中係数	Allmtd. Kedl(I)
	弾性追従係数	Allmtd. Qn(I)
	K e' 係数	Allmtd. Ked(I)
	NEUBER則	Allmtd. Neuber(I)
	歪集中係数	Allmtd. Ke(I)
	熱ビーク歪	Allmtd. Ef(I)
	全歪範囲	Allmtd. Et(I)
	初期応力	Allmtd. Si(I)
	弾性追従係数	Allmtd. Qeff(I)
	D f	Allmtd. Df(I)
	D c	Allmtd. Dc(I)
D	Allmtd. D(I)	
EQEF-WELD	表面応力	Allmtd. Spx(I)
	等価線形応力	Allmtd. Sn(I)
	応力集中係数	Allmtd. K(I)
	歪集中係数	Allmtd. Ke(I)
	総体的歪集中係数	Allmtd. Kedg(I)
	局所的歪集中係数	Allmtd. Kedl(I)
	熱ビーク歪	Allmtd. Ef(I)
	全歪範囲	Allmtd. Et(I)
	初期応力	Allmtd. Si(I)
	弾性追従係数	Allmtd. Qn(I)
	熔接金属に対する弾性追従係	Allmtd. Qw(I)
	局所的弾性追従係数	Allmtd. Ql(I)
	多軸効果を現す弾性追従係数	Allmtd. Qnu(I)
	弾性追従係数	Allmtd. Qeff(I)
	D f	Allmtd. Df(I)
	D c	Allmtd. Dc(I)

	D D c r	Allmtd. D(I) Allmtd. Dcr(I)
PNC	表面応力	Allmtd. Spx(I)
	等価線形応力	Allmtd. Sn(I)
	応力集中係数	Allmtd. K(I)
	K e' 係数	Allmtd. Ked(I)
	歪集中係数	Allmtd. Ke(I)
	総体的歪集中係数	Allmtd. Kedg(I)
	局所的歪集中係数	Allmtd. Kedl(I)
	熱ピーク歪	Allmtd. Ef(I)
	全歪範囲	Allmtd. Et(I)
	初期応力	Allmtd. Si(I)
	弾性追従係数	Allmtd. Qn(I)
	局所的弾性追従係数	Allmtd. Ql(I)
	多軸効果を現す弾性追従係数	Allmtd. Qnu(I)
	弾性追従係数	Allmtd. Qeff(I)
	D f	Allmtd. Df(I)
	D c	Allmtd. Dc(I)
	D	Allmtd. D(I)
D c r	Allmtd. Dcr(I)	

表 1. 3 損傷計算結果項目の詳細

## 4 Visual Basic環境

### 4. 1 機能概要

損傷プログラム本体は、損傷計算を行うVisual Basic言語で書かれたEXE形式ファイルとFortran言語で書かれた材料特性値算出用ライブラリであるDLL形式ファイルから構成される。

この2つはそれぞれ独立したファイルであり、Visual Basicプログラムは常に主記憶に常駐しているが、Fortranライブラリは必要な時だけ主記憶に呼び出される。

MS-EXCEL（表計算ソフト）から2つの入力データ（試験条件及び解析結果）を通信により受け取る。そこで指定された評価法及び材料等のデータに応じた損傷値を計算し、結果をまたMS-EXCELに通信によって返す。

入力データ／解析結果は、弾性解析に対応している。

損傷プログラムに対する入出力データの受け渡し及び結果データのグラフ化は、表計算ソフト（MS-EXCEL）が行っている。データの受け渡しの方法はファイルを介さず全てプロセス間通信（DDE）によって行われる。

<b>TTSDS</b>	評価法名： T T S - D S
<b>BDS</b>	評価法名： B D S
<b>BDSMD</b>	評価法名： B D S - M D
<b>DDS</b>	評価法名： D D S
<b>DDSMDWLD</b>	評価法名： D D S M D - W E L D
<b>TYPE1</b>	評価法名： T Y P E 1
<b>TYPE2</b>	評価法名： T Y P E 2
<b>EQEF</b>	評価法名： E Q E F
<b>EQEFWD</b>	評価法名： E Q E F - W E L D
<b>PNC</b>	評価法名： P N C
<b>EQEF15</b>	評価法名： E Q E F 1 5
<b>EQEF167</b>	評価法名： E Q E F 1 6 7
<b>INELAST</b>	評価法名： I N E L A S T I C
<b>SIEQ167</b>	評価法名： S I E Q 1 6 7
<b>EQEF167W</b>	評価法名： E Q E F 1 6 7 W

合計 : 15 評価法

表2. 1 評価法名称とキーワードの一覧

<b>SUS304</b>	鋼種 : SUS304
<b>SUS316</b>	鋼種 : SUS3016
<b>SCMV4</b>	鋼種 : SCMV4
<b>STBA24</b>	鋼種 : STBA24
<b>MOD9CR-1MO</b>	鋼種 : 改良9CR-1MO鋼
<b>SUS316FR</b>	鋼種 : SUS316FR

合計 : 6 鋼種

表2.2 材料名称とキーワードの一覧

## 4. 2 プログラム構成

損傷プログラムは、損傷計算を行うプログラム本体と材料特性値算出用のライブラリから構成される。

### 1) 損傷計算を行うコード部

... Visual Basic 言語による EXE 形式ファイル

### 2) 材料特性値算出用ライブラリ部

... Fortran 言語による DLL 形式ファイル

プログラム構成図 2. 1 を参照の事。

本プログラムの主要部分は Visual Basic 言語で記述されている。材料特性算出用ライブラリは、DLL (動的リンクライブラリ) 形式にしてあるため損傷プログラム本体とは独立しており開発も別々に行われる。

このライブラリは、FORTRAN 言語で記述されている。

いずれも (米) Microsoft 社のコンパイラであるため、Visual Basic 言語から FORTRAN 言語を呼び出すという Multi Language 仕様が可能となっている。

つまり DLL 形式ファイルは、ある呼び出し規約に則れば C/VB 等いずれの言語からも呼び出す事が出来る。

損傷プログラムに対する入出力データの受け渡し及び結果データのグラフ化は、表計算ソフト (MS-EXCEL) が行っている。データの受け渡しの方法はファイルを介さず全てプロセス間通信 (DDE) によって行われる。

#### ※ 動的リンクライブラリ (DLL) とは

本詳細説明書の「7. 動的リンクライブラリ (DLL)」を参照の事。

#### ※ Multi Language とは

本詳細説明書の「7. 2 複数言語によるプログラミング

(Multi Language)」を参照の事。

## 4. 3 入出力仕様

損傷プログラムが行うデータの入出力に関して説明する。

大きくは2つのプログラム層とのデータ受け渡しを行う。

(1) MS-EXCEL (表計算ソフト) とのデータ受け渡し

(2) MMFUNC (材料特性値算出用ライブラリ/DLL形式) とのデータ受け渡し

(1) に関しては「3. 2 MS-EXCELの入出力仕様」を参照の事。

損傷プログラムは、MS-EXCELから受け取ったデータを主記憶上に構造体 (異なる型の変数を1まとめにしたもの) として保持する。

EXCEL側のファイルを数ブロックに分割してデータを受け取り処理している。

現状は、50行/1ブロックで処理している。行数パラメータは変更可能。

(2)に於いては、プログラム内のデータ受け渡しであるため通常のサブルーチンのパラメータ受け渡しと同じイメージで処理出来る。

実際のパラメータ形式は構造体である、各構造体は用途に応じて分類してある。4.5以降に(1)(2)それぞれの中で使用している構造体を示す。

#### 4.4 基本変数ライブラリの項目

損傷プログラム内で使用する損傷計算基本変数算出用ライブラリを示す。

表3.1を参照の事。

表 3. 1 損傷計算用基本変数算出用ライブラリ

S T A R表計算シートと計算式、変数とも同一のサブルーチン群

応力成分差	表面応力	$\sigma R$ $\sigma Z$ $\sigma \theta$ $\tau RZ$	$D\sigma R$ $D\sigma Z$ $D\sigma \theta$ $D\tau RZ$	$D\sigma R = \sigma R1 - \sigma R2$ $D\sigma Z = \sigma Z1 - \sigma Z2$ $D\sigma \theta = \sigma \theta 1 - \sigma \theta 2$ $D\tau RZ = \tau RZ1 - \tau RZ2$
	等価線形 応力	$\sigma R$ $\sigma Z$ $\sigma \theta$ $\tau RZ$	$DL\sigma R$ $DL\sigma Z$ $DL\sigma \theta$ $DL\tau RZ$	$DL\sigma R = L\sigma R1 - L\sigma R2$ $DL\sigma Z = L\sigma Z1 - L\sigma Z2$ $DL\sigma \theta = L\sigma \theta 1 - L\sigma \theta 2$ $DL\tau RZ = L\tau RZ1 - L\tau RZ2$
	膜応力	$\sigma R$ $\sigma Z$ $\sigma \theta$ $\tau RZ$	$DM\sigma R$ $DM\sigma Z$ $DM\sigma \theta$ $DM\tau RZ$	$DM\sigma R = M\sigma R1 - M\sigma R2$ $DM\sigma Z = M\sigma Z1 - M\sigma Z2$ $DM\sigma \theta = M\sigma \theta 1 - M\sigma \theta 2$ $DM\tau RZ = M\tau RZ1 - M\tau RZ2$
主応力成分差	表面応力	$\sigma 1$ $\sigma 2$ $\sigma 3$	$D\sigma 1$ $D\sigma 2$ $D\sigma 1$	$PRIN1(D\sigma R, D\sigma Z, D\tau RZ) *1$ $PRIN2(D\sigma R, D\sigma Z, D\tau RZ) *2$ $D\sigma \theta$
	等価線形 応力	$\sigma 1$ $\sigma 2$ $\sigma 3$	$DL\sigma 1$ $DL\sigma 2$ $DL\sigma 1$	$PRIN1(DL\sigma R, DL\sigma Z, DL\tau RZ) *1$ $PRIN2(DL\sigma R, DL\sigma Z, DL\tau RZ) *2$ $DL\sigma \theta$
	膜応力	$\sigma 1$ $\sigma 2$ $\sigma 3$	$DM\sigma 1$ $DM\sigma 2$ $DM\sigma 1$	$PRIN1(DM\sigma R, DM\sigma Z, DM\tau RZ) *1$ $PRIN2(DM\sigma R, DM\sigma Z, DM\tau RZ) *2$ $DM\sigma \theta$
応力成分範囲	表面応力	$\sigma R$ $\sigma Z$ $\sigma \theta$ $\tau RZ$	$\Delta\sigma R$ $\Delta\sigma Z$ $\Delta\sigma \theta$ $\Delta\tau RZ$	$ABS(D\sigma R)$ $ABS(D\sigma Z)$ $ABS(D\sigma \theta)$ $ABS(D\tau RZ)$
	等価線形 応力	$\sigma R$ $\sigma Z$ $\sigma \theta$ $\tau RZ$	$\Delta L\sigma R$ $\Delta L\sigma Z$ $\Delta L\sigma \theta$ $\Delta L\tau RZ$	$ABS(DL\sigma R)$ $ABS(DL\sigma Z)$ $ABS(DL\sigma \theta)$ $ABS(DL\tau RZ)$
	膜応力	$\sigma R$ $\sigma Z$ $\sigma \theta$ $\tau RZ$	$\Delta M\sigma R$ $\Delta M\sigma Z$ $\Delta M\sigma \theta$ $\Delta M\tau RZ$	$ABS(DM\sigma R)$ $ABS(DM\sigma Z)$ $ABS(DM\sigma \theta)$ $ABS(DM\tau RZ)$
主応力成分範囲	表面応力	$\sigma 1$ $\sigma 2$ $\sigma 3$	$\Delta\sigma 1$ $\Delta\sigma 2$ $\Delta\sigma 1$	$ABS(\Delta\sigma 1)$ $ABS(\Delta\sigma 2)$ $ABS(\Delta\sigma 3)$
	等価線形 応力	$\sigma 1$ $\sigma 2$ $\sigma 3$	$\Delta L\sigma 1$ $\Delta L\sigma 2$ $\Delta L\sigma 1$	$ABS(\Delta L\sigma 1)$ $ABS(\Delta L\sigma 2)$ $ABS(\Delta L\sigma 3)$



	膜応力	$\sigma_1$ $\sigma_2$ $\sigma_3$	$\Delta M \sigma_1$ $\Delta M \sigma_2$ $\Delta M \sigma_3$	ABS( $\Delta M \sigma_1$ ) ABS( $\Delta M \sigma_2$ ) ABS( $\Delta M \sigma_3$ )
相当応力範囲	表面応力	ミーゼス トレスカ ランキン	$\Delta VMS$ $\Delta TRES$ $\Delta RANK$	VMS( $D \sigma R, D \sigma Z, D \tau RZ$ ) *3 TRES( $D \sigma_1, -\sigma_2 - \sigma_3$ ) *4 MAX( $\Delta \sigma_1, \Delta \sigma_2, \Delta \sigma_3$ )
	等価線形 応力	ミーゼス トレスカ ランキン	$\Delta LVMS$ $\Delta LTRES$ $\Delta LRANK$	VMS(DL $\sigma R, \dots$ ) *3 TRES(DL $\sigma_1, \dots$ ) *4 MAX( $\Delta L \sigma_1, \Delta L \sigma_2, \Delta L \sigma_3$ )
	膜応力	ミーゼス トレスカ ランキン	$\Delta MVMS$ $\Delta MTRES$ $\Delta MRANK$	VMS(DM $\sigma R, \dots$ ) *3 TRES(DM $\sigma_1, \dots$ ) *4 MAX( $\Delta M \sigma_1, \Delta M \sigma_2, \Delta M \sigma_3$ )
歪成分差	弾性歪	$\epsilon R$ $\epsilon Z$ $\epsilon \theta$ $\gamma RZ$	D $\epsilon R$ D $\epsilon Z$ D $\epsilon \theta$ D $\gamma RZ$	D $\epsilon R = \epsilon R1 - \epsilon R2$ D $\epsilon Z = \epsilon Z1 - \epsilon Z2$ D $\epsilon \theta = \epsilon \theta 1 - \epsilon \theta 2$ D $\gamma RZ = \gamma RZ1 - \gamma RZ2$
	塑性歪	$\epsilon R$ $\epsilon Z$ $\epsilon \theta$ $\gamma RZ$	D $\epsilon PR$ D $\epsilon PZ$ D $\epsilon P\theta$ D $\gamma PRZ$	D $\epsilon PR = \epsilon PR1 - \epsilon PR2$ D $\epsilon PZ = \epsilon PZ1 - \epsilon PZ2$ D $\epsilon P\theta = \epsilon P\theta 1 - \epsilon P\theta 2$ D $\gamma PRZ = \gamma PRZ1 - \gamma PRZ2$
	クリープ歪	$\epsilon R$ $\epsilon Z$ $\epsilon \theta$ $\gamma RZ$	D $\epsilon CR$ D $\epsilon CZ$ D $\epsilon C\theta$ D $\gamma CRZ$	D $\epsilon CR = \epsilon CR1 - \epsilon CR2$ D $\epsilon CZ = \epsilon CZ1 - \epsilon CZ2$ D $\epsilon C\theta = \epsilon C\theta 1 - \epsilon C\theta 2$ D $\gamma CRZ = \gamma CRZ1 - \gamma CRZ2$
相当歪範囲	弾塑性クリープ歪範囲 塑性歪範囲 クリープ歪範囲		$\Delta EEPCEQ$ $\Delta EPEQ$ $\Delta ECEQ$	EEQ(D $\epsilon R, D \epsilon PR, D \epsilon CR, \dots$ ) *5 EEQ(D $\epsilon PR, \dots$ ) *5 EEQ(D $\epsilon CR, \dots$ ) *5

\*1、\*2、\*3、\*4、\*5 は以下の通り

\*1 PRIN1( $\sigma R, \sigma Z, \tau RZ$ )  
 $= (\sigma R + \sigma Z) / 2 + \text{SQRT}(((\sigma R + \sigma Z) / 2) ** 2 + \tau RZ ** 2)$

\*2 PRIN2( $\sigma R, \sigma Z, \tau RZ$ )  
 $= (\sigma R + \sigma Z) / 2 - \text{SQRT}(((\sigma R + \sigma Z) / 2) ** 2 + \tau RZ ** 2)$

\*3 VMS( $\sigma R, \sigma Z, \sigma \theta, \tau RZ$ )  
 $= \text{SQRT}(1/2((\sigma R - \sigma Z) ** 2 + (\sigma Z - \sigma \theta) ** 2 + (\sigma \theta - \sigma R) ** 2 + 6 \tau RZ ** 2))$

\*4 TRES( $\sigma_1, \sigma_2, \sigma_3$ )  
 $= \text{MAX}(\text{ABS}(\sigma_1 - \sigma_2), \text{ABS}(\sigma_2 - \sigma_3), \text{ABS}(\sigma_3 - \sigma_1))$

\*5 EEQ( $\epsilon R, \epsilon Z, \epsilon \theta, \gamma RZ$ )  
 $= 1/3 \text{SQRT}(2((\epsilon R - \epsilon Z) ** 2 + (\epsilon Z - \epsilon \theta) ** 2 + (\epsilon \theta - \epsilon R) ** 2 + 3/2 \gamma RZ ** 2))$

#### 4. 5 構造体変数

MS-EXCELより受け取った試験条件データ及び解析結果データの入力データは主記憶上の構造体に保持される。また処理結果である損傷計算結果データも専用の構造体に格納される。

損傷プログラム内でのサブルーチンのパラメータ受け渡しは殆ど構造体によって受け渡されている。個々のパラメータを羅列すると膨大な数になってしまい、非常に見にくいものになってしまう。ところが構造体の場合タグ名1つをパラメータとして渡せば良いので呼び出しがスマートに行える。

また意味の異なるグループ毎に分類出来るので理解しやすい記述が出来る。

※ 構造体とは

本詳細説明書の「4. 5. 1 構造体」を参照の事。

構造体は、用途に応じて以下の5つに分類されている。

(1) 試験条件データ

表 3. 2 に各種項目の詳細を示す。

(2) 解析結果データ

表 3. 3 に各種項目の詳細を示す。

(3) 基本変数データ

表 3. 4 に各種項目の詳細を示す。

(4) 各種変数データ

表 3. 5 に各種項目の詳細を示す。

(5) 損傷計算結果データ

表 3. 6 に各種項目の詳細を示す。

機能	試験条件シート要素(Jyoken.in)を格納するユーザー定義体				
タグ名	Condition				
変数名	Cond				
メンバ	メンバ名	入出力(I/O)	型	型	意味
	Material	Input	文字型*2B	String	材料名
	Tempc	Input	単精度浮動小数点*4B	Single	Cold温度
	TempH	Input	単精度浮動小数点*4B	Single	Hot 温度
	Hldtmc	Input	単精度浮動小数点*4B	Single	Cold保持時間
	Hldtmh	Input	単精度浮動小数点*4B	Single	Hot 保持時間
	Mecha	Input	単精度浮動小数点*4B	Single	機械荷重
	Cycle	Input	整数型*2B	Integer	サイクル数
	MethodName	Input	文字型*2B	String	評価法名称
	MethodNo	Input	整数型*2B	Integer	評価法番号
	AnalSheetName	Input	文字型*2B	String	解析シート名称
	AnalSheetRow	Input	整数型*2B	Integer	解析シート行数
	AnalSheetCol	Input	整数型*2B	Integer	解析シート桁数
	OutSheetName	Input	文字型*2B	String	出力シート名称
	Imate	Input	整数型*2B	Integer	材料番号
備考					

表3.2 試験条件シート用構造体

機能	Range解析結果シート要素Sonsho.in)を格納するユーザー定義体 補足 : (1) 現在「起点座標」のみ領域確保 理由: 損傷計算処理は「終点座標」を必要としないため。  (2) 現在「時点2」まで領域確保 理由: メモリの削減。 「時点3」は極希に使用する程度であるため。				
タグ名	Analysis				
変数名	Analy				
メンバ	メンバ名	入出力(I/O)	型	型	意味
	R(IN, 0)	Input	単精度浮動小数点*4B	Single	R座標 (起点座標のみ)
	Z(IN, 0)	Input	単精度浮動小数点*4B	Single	Z座標 ( " )
	H(IN, 0)	Input	単精度浮動小数点*4B	Single	θ座標( " )
	Tmpmax(IN)	Input	単精度浮動小数点*4B	Single	最高金属温度
	Ptime(IN, 1)	Input	単精度浮動小数点*4B	Single	評価時刻 (時点2まで)
	Temp(IN, 1)	Input	単精度浮動小数点*4B	Single	表面温度 ( " )
	Ltemp(IN, 1)	Input	単精度浮動小数点*4B	Single	等価線形温度( " )
	Mtemp(IN, 1)	Input	単精度浮動小数点*4B	Single	平均温度 ( " )
	Strs(IN, 3, 1)	Input	単精度浮動小数点*4B	Single	表面応力 ( " )
	Lstrs(IN, 3, 1)	Input	単精度浮動小数点*4B	Single	等価線形応力( " )
	Mstrs(IN, 3, 1)	Input	単精度浮動小数点*4B	Single	膜応力 ( " )
	Ee(IN, 3, 1)	Input	単精度浮動小数点*4B	Single	弾性歪 ( " )
	Ep(IN, 3, 1)	Input	単精度浮動小数点*4B	Single	塑性歪 ( " )
	Ec(IN, 3, 1)	Input	単精度浮動小数点*4B	Single	クリープ歪 ( " )
	Q(IN)	Input	単精度浮動小数点*4B	Single	Q
備考	IN=50、50行/1処理xn 処理				

表3.3 解析結果シート用構造体

機能	基本変数要素				
タグ名	Bscval				
変数名	Bscval				
メンバ	メンバ名	入出力(I/O)	型	型	意味
	Tres(IN)	Output	単精度浮動小数点*4B	Single	相当応力範囲表面応力
	Ltres(IN)	Output	単精度浮動小数点*4B	Single	相当応力範囲等価線形応力
	Vms(IN)	Output	単精度浮動小数点*4B	Single	相当応力範囲表面応力
	Eepceq(IN)	Output	単精度浮動小数点*4B	Single	相当歪範囲弾塑性クリープ歪範囲
	Epeq(IN)	Output	単精度浮動小数点*4B	Single	相当歪範囲塑性歪範囲
	Eceq(IN)	Output	単精度浮動小数点*4B	Single	相当歪範囲クリープ歪範囲
備考	IN=50、50行/1処理xn 処理				

表3.4 基本変数要素用構造体

機能	各種変数要素				
タグ名	Mc				
変数名	Mc				
メンバ	メンバ名	入出力(I/O)	型	型	意味
	Im	Output	整数型*2B	Integer	
	Tc	Output	単精度浮動小数点*4B	Single	
	E	Output	単精度浮動小数点*4B	Single	
	Sy	Output	単精度浮動小数点*4B	Single	
	Sp	Output	単精度浮動小数点*4B	Single	
	Am	Output	単精度浮動小数点*4B	Single	
	Ak	Output	単精度浮動小数点*4B	Single	
	Spc	Output	単精度浮動小数点*4B	Single	
	Amc	Output	単精度浮動小数点*4B	Single	
	Akc	Output	単精度浮動小数点*4B	Single	
	Tr0	Output	単精度浮動小数点*4B	Single	
	Tr1	Output	単精度浮動小数点*4B	Single	
	Tr2	Output	単精度浮動小数点*4B	Single	
	Su	Output	単精度浮動小数点*4B	Single	
	Sm	Output	単精度浮動小数点*4B	Single	
	S1	Output	単精度浮動小数点*4B	Single	
	Alph1	Output	単精度浮動小数点*4B	Single	
	S2	Output	単精度浮動小数点*4B	Single	
	Alph2	Output	単精度浮動小数点*4B	Single	
	G1	Output	単精度浮動小数点*4B	Single	
	Beta1	Output	単精度浮動小数点*4B	Single	
	G2	Output	単精度浮動小数点*4B	Single	
	Beta2	Output	単精度浮動小数点*4B	Single	
	F	Output	単精度浮動小数点*4B	Single	
	Ramda	Output	単精度浮動小数点*4B	Single	
	A0t	Output	単精度浮動小数点*4B	Single	
	A0r	Output	単精度浮動小数点*4B	Single	
	A1t	Output	単精度浮動小数点*4B	Single	
	A1r	Output	単精度浮動小数点*4B	Single	
	A2t	Output	単精度浮動小数点*4B	Single	
	A2r	Output	単精度浮動小数点*4B	Single	
	A4t	Output	単精度浮動小数点*4B	Single	
	A4r	Output	単精度浮動小数点*4B	Single	
備考					

表3.5 各種変数要素用構造体

機能	損傷計算結果 $\sigma$ - $\epsilon$ 変数(全評価法共用)				
タグ名	Methods				
変数名	Allmtd				
メンバ	メンバ名	入出力(I/O)	型	型	意味
	Df(IN)	Output	単精度浮動小数点*4B	Single	D f
	Dc(IN)	Output	単精度浮動小数点*4B	Single	D c
	Dcr(IN)	Output	単精度浮動小数点*4B	Single	D c r
	D(IN)	Output	単精度浮動小数点*4B	Single	D
	Ef(IN)	Output	単精度浮動小数点*4B	Single	熱ピーク歪
	Et(IN)	Output	単精度浮動小数点*4B	Single	全歪範囲
	K(IN)	Output	単精度浮動小数点*4B	Single	応力集中係数
	Ked(IN)	Output	単精度浮動小数点*4B	Single	K $\sigma$ ' 係数
	Ke(IN)	Output	単精度浮動小数点*4B	Single	歪集中係数
	Kedg(IN)	Output	単精度浮動小数点*4B	Single	総体的歪集中係数
	Kedl(IN)	Output	単精度浮動小数点*4B	Single	局所的歪集中係数
	Kep(IN)	Output	単精度浮動小数点*4B	Single	歪集中係数
	Neuber(IN)	Output	単精度浮動小数点*4B	Single	NEUBER則
	Qn(IN)	Output	単精度浮動小数点*4B	Single	弾性追従係数
	Ql(IN)	Output	単精度浮動小数点*4B	Single	局所的弾性追従係数
	Qnu(IN)	Output	単精度浮動小数点*4B	Single	多軸効果を現す弾性追従係数
	Qf(IN)	Output	単精度浮動小数点*4B	Single	弾性追従係数
	Qw(IN)	Output	単精度浮動小数点*4B	Single	熔接金属に対する弾性追従係数
	Qeff(IN)	Output	単精度浮動小数点*4B	Single	弾性追従係数
	Qbm(IN)	Output	単精度浮動小数点*4B	Single	母材に対する弾性追従係数
	Spx(IN)	Output	単精度浮動小数点*4B	Single	表面応力
	Sn(IN)	Output	単精度浮動小数点*4B	Single	等価線形応力
	Si(IN)	Output	単精度浮動小数点*4B	Single	初期応力
備考	IN=50,50行/1処理x n 処理				

表3.6 損傷計算結果 $\sigma$ - $\epsilon$ 変数用構造体

#### 4. 5. 1 構造体とは

Visual Basicでは、異なるデータ型を組み合わせた「ユーザー定義型」(Cの構造体に相当)を作ることができる。ユーザー定義型はいくつかの関連した情報を1つの変数で表わすのに非常に便利である。

ユーザー定義型を定義するには、Typeステートメントを使う。このステートメントは、コードモジュールの宣言セクションに置かなければならない。型の定義は常にグローバルであるから、フォームモジュールでTypeステートメントを使うことはできない。しかし、ユーザー定義体の変数の宣言は、モジュールレベル・プロセスレベルのいずれでも可能であり、変数のスコープ(適用範囲)はそれぞれの宣言の方法に従う。例えば、次のように、コンピュータシステムに関する情報を記録するユーザー定義型を定義することができる。

‘宣言 (コードモジュールの宣言セクションで)

```
Type SystemInfo
```

```
    CPU As Variant
```

```
    Memory As Long
```

```
    VideoColors As Integer
```

```
    PurChaseDate As Variant
```

```
End Type
```

このデータ型の変数は、グローバル変数、モジュールレベル変数、ローカル変数のいずれにも宣言できる。

```
Dim MySystem As SystemInfo, YourSystem As SystemInfo
```

このデータ型の変数は対する値の割り当ておよび取得は、プロパティの設定および取得と似ている。

```
MySystem.Cpu = '486'
```

```
If MySystem.PurChaseDate > #92/1/1# Then
```

同じユーザー定義型であれば、ある変数に代入することもできる。この場合その変数のすべての要素がもう一方の変数のそれぞれの要素に代入される。

```
YourSystem = MySystem
```

ユーザー定義型は内部に通常の配列を(サイズ固定型の配列)を含むことができる。

ユーザー定義型変数に含まれる配列へのアクセスは、プロパティ配列へのアクセスと同様の方法によって行われる。また、ユーザー定義型に別のユーザー

定義型を含むことも可能である。

プロシージャに、引数としてユーザー定義型変数を渡すこともできる。

Functionプロシージャはユーザー定義型の値を返すことはできない。プロシージャからユーザー定義型の値を受け取りたい場合は、常に参照によって渡されるので前に説明したように、プロシージャはその変数の値を変更して呼び出し側のプロシージャに返すことができる。

バリエーション型はさまざまな型の値を保持することができるため、ユーザー定義型の代わりにバリエーション型の配列を使える場合が多くある。バリエーション型の配列は、いつでも各要素の型を変更できる点や、動的配列として宣言することにより必要に応じてサイズを変更できる点で、ユーザー定義型よりも柔軟性に富んでいる。

しかし、バリエーション型はユーザー定義型よりも常に多くのメモリを消費する点に注意しなければならない。

#### 4. 6 損傷計算で使用した計算式

各評価法の中で使用している計算式を示す。

図4. 1～図4. 9に詳細な計算の流れを示す。



Δε<sub>cq</sub>の算出 全ステップの組み合わせの中で最大値

$$\Delta \varepsilon_{cq} = \frac{\sqrt{2}}{3} \left\{ (\Delta \varepsilon_r - \Delta \varepsilon_z)^2 + (\Delta \varepsilon_z - \Delta \varepsilon_\theta)^2 + (\Delta \varepsilon_\theta - \Delta \varepsilon_r)^2 + \frac{3}{2} \Delta \gamma_{rz}^2 \right\}^{1/2}$$

$$\Delta \varepsilon_r = \varepsilon_r^i - \varepsilon_r^j \quad \varepsilon_r^i, \varepsilon_r^j : \text{step } i, j \rightarrow \varepsilon_r \text{ etc}$$

$$\varepsilon_r = \frac{1.5}{1 + \nu} \Delta \varepsilon_{cq}$$

νは2次点の値の平均値  
ν:ポアソン比

$$\varepsilon_t = K_c \varepsilon_r \quad K = 1.2$$

(疲労損傷評価)

(クリープ損傷評価)

歪速度:  $\dot{\varepsilon} = 2 \varepsilon_t / \tau$

τ:周期

BDS最適疲労破損式  
ε<sub>t</sub>, ε → 破損繰返し変数 N<sub>f</sub>

疲労破損  
D<sub>r</sub> = N / N<sub>f</sub>  
N:繰返変数

BDS動的応力ひずみ関係式  
ε<sub>t</sub> → 初期応力 σ<sub>i</sub>

軸力σ<sub>z</sub>がある場合  
初期応力 → σ<sub>i</sub> + σ<sub>z</sub>

BDS主クリープ破断式  
σ(t) → 破断時間 t<sub>r</sub>(t)  
α<sub>R</sub> = 1.0, α<sub>c</sub> = 1.0

クリープ損傷  
 $D_c = N \int_0^{t_N} \frac{dt}{tR(t)}$   
TN: 1サイクル当り高温保持時間

クリープ疲労損傷  
D = D<sub>f</sub> + D<sub>c</sub>

図4.1 TTS-DSによる評価フロー

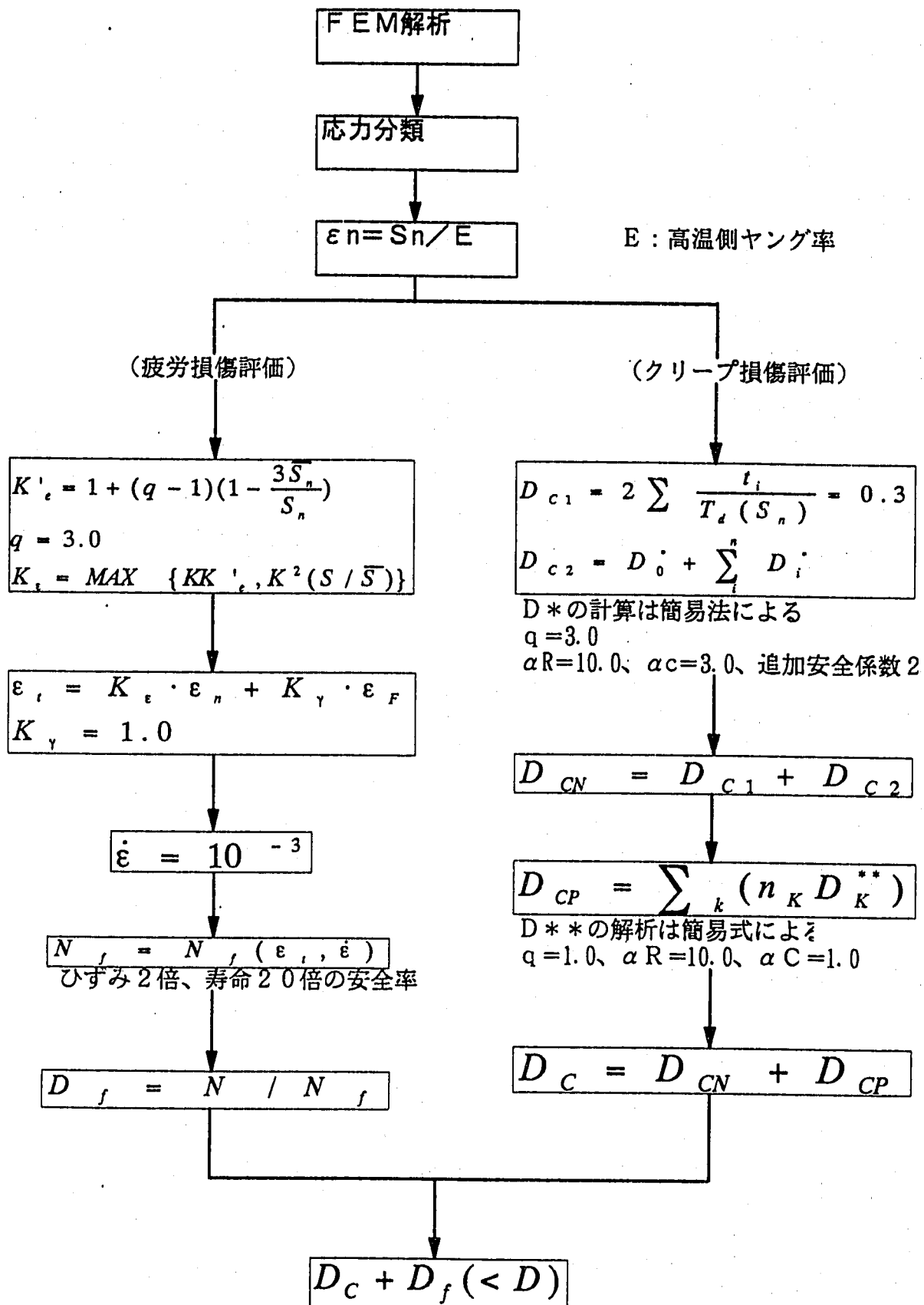


図4.2 BDSによる評価フロー

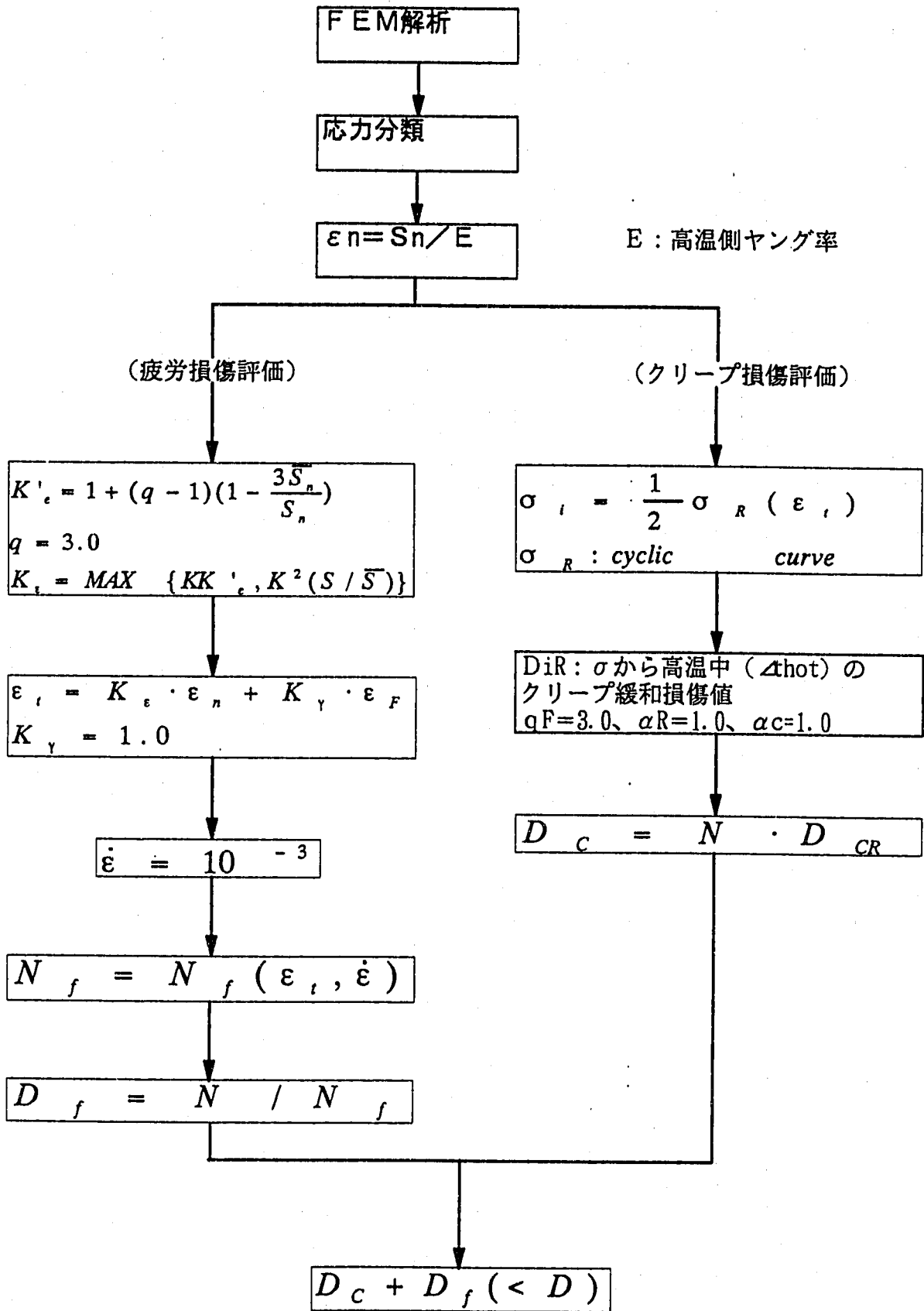


図4.3 BDS-MDによる評価フロー

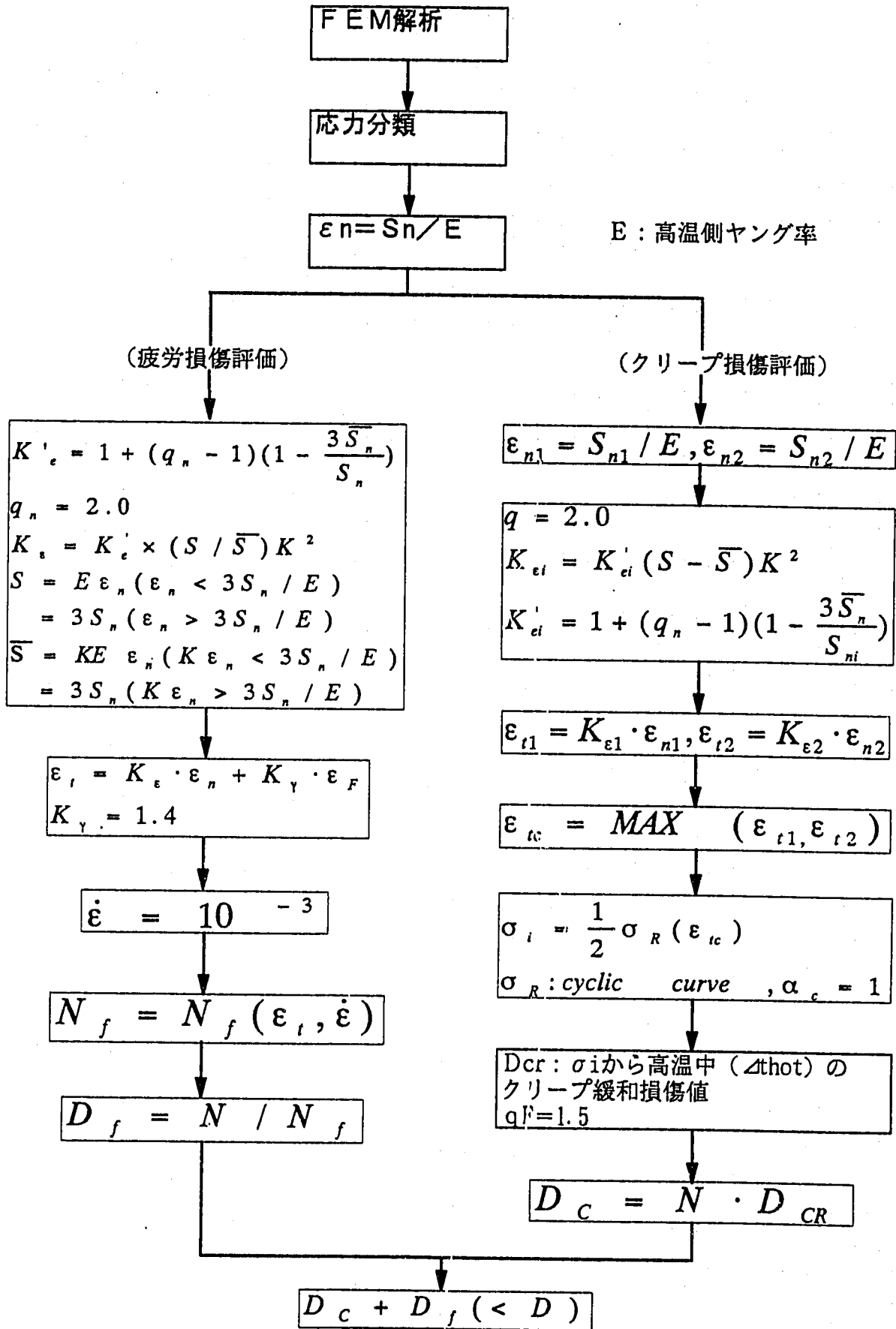


図4.4 DDS-MDによる評価フロー

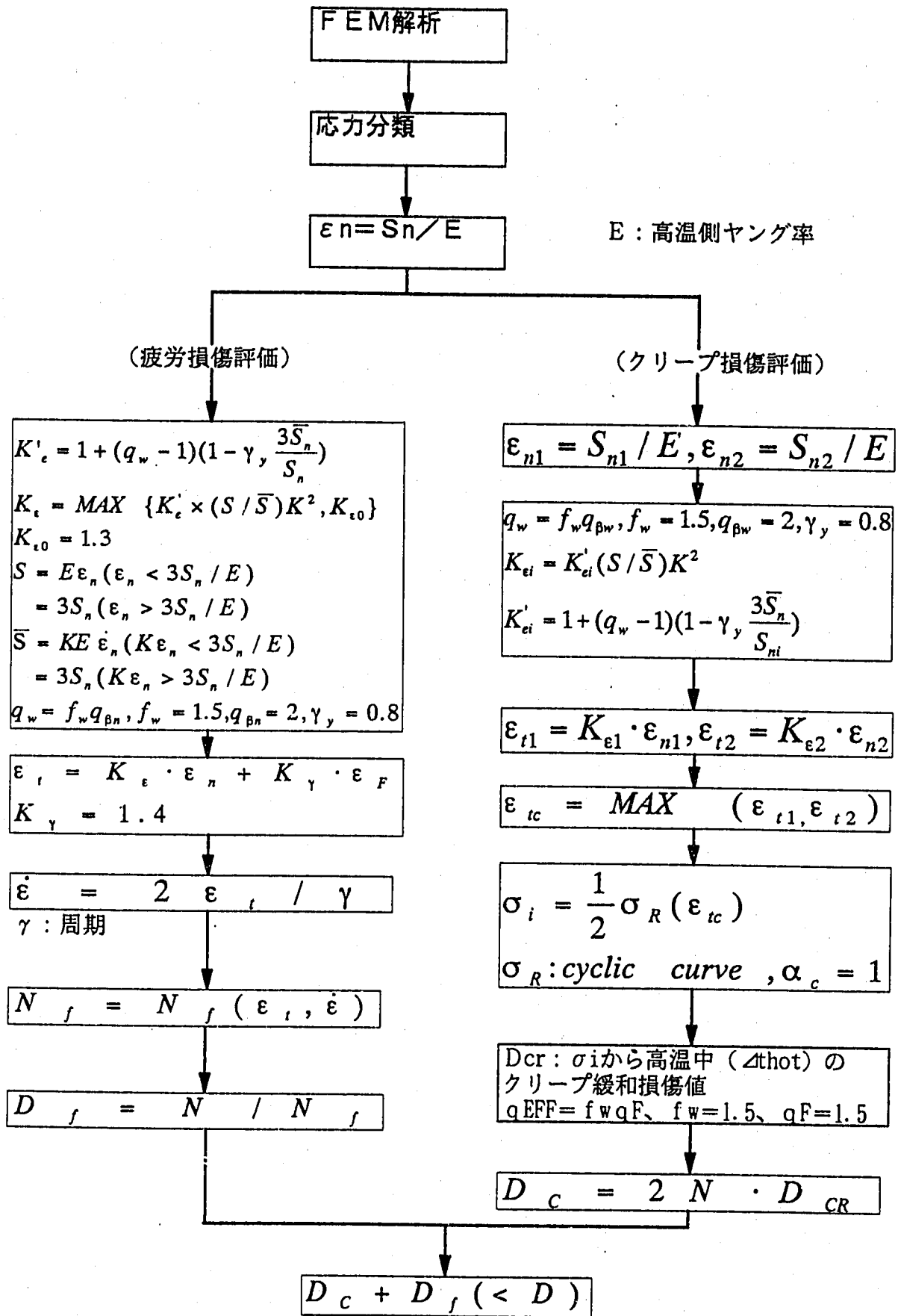


図4.5 DDSMD-WELDによる評価フロー

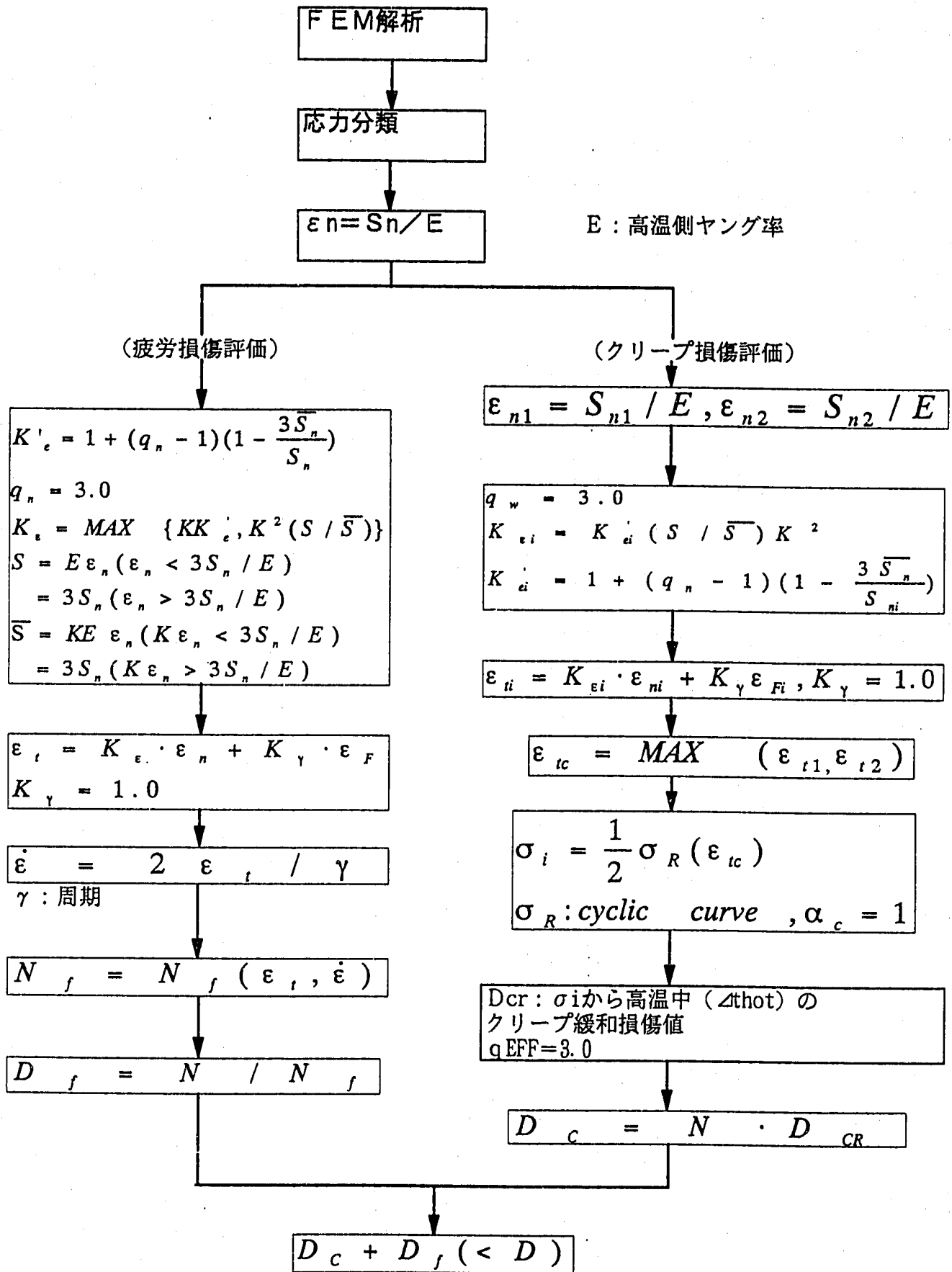


図4.6 TYPE Iによる評価フロー

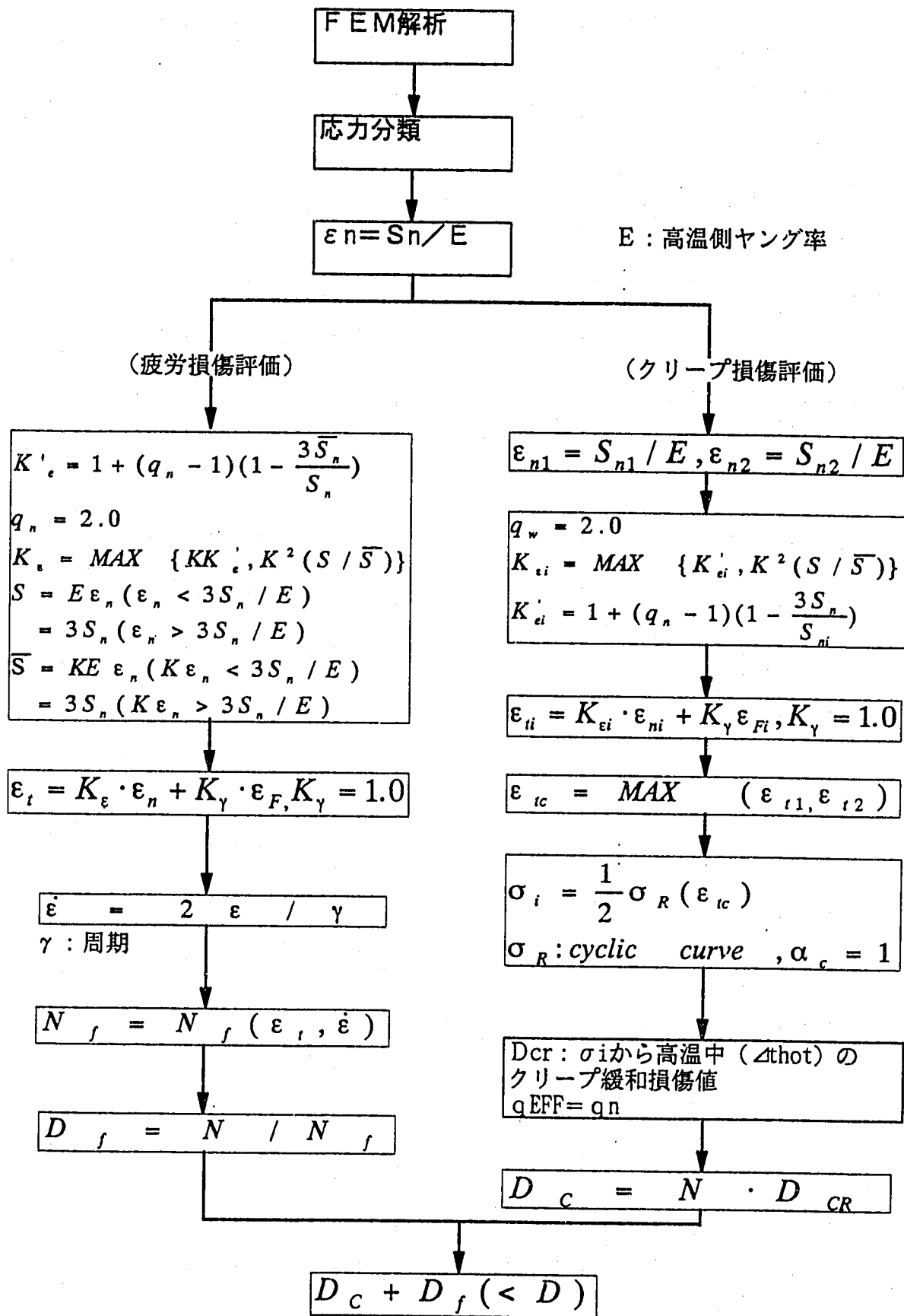


図4.7 TYPE IIによる評価フロー

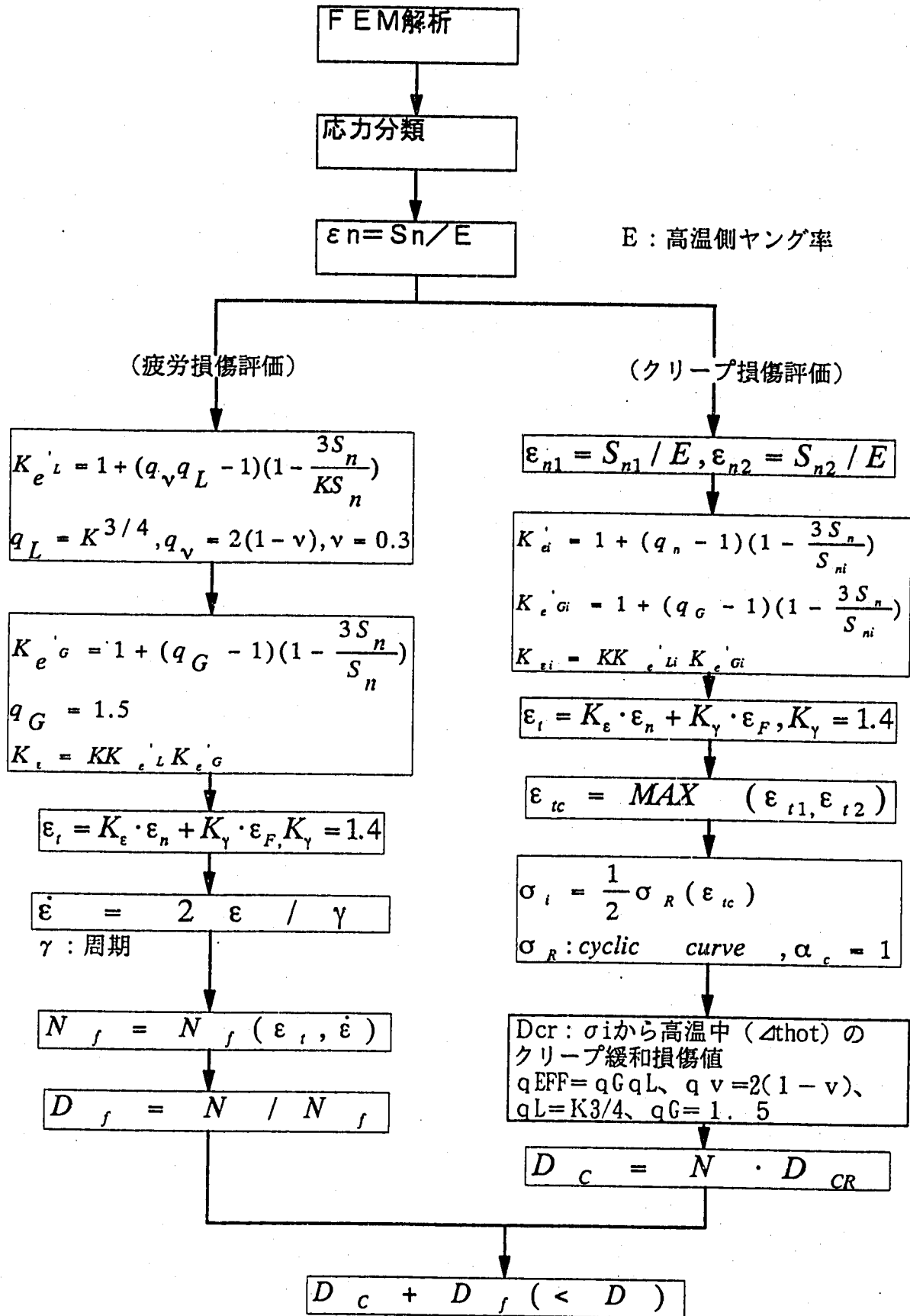


図4.8 EQEFによる評価フロー



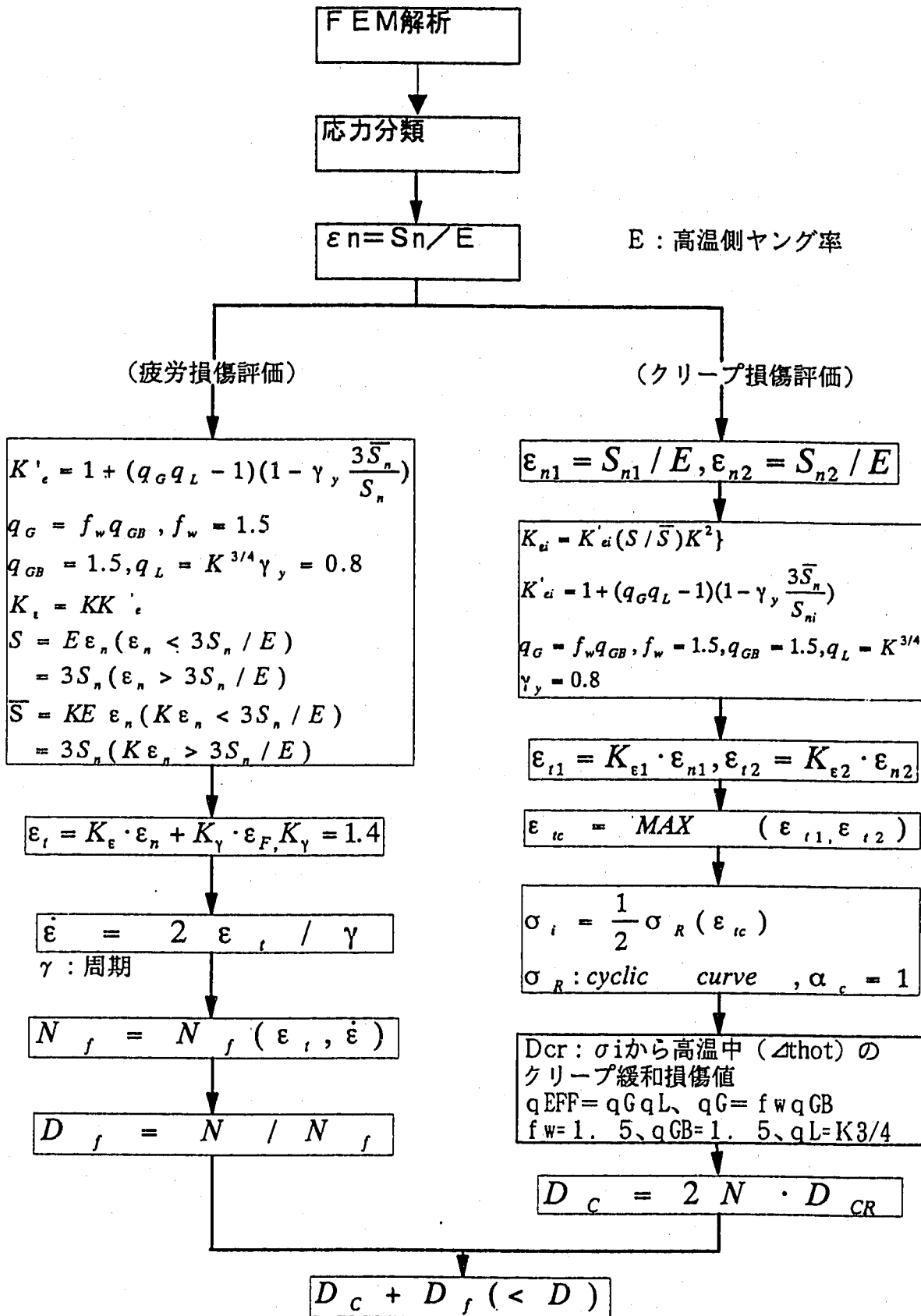


図4.9 EQEF-WELDによる評価フロー

#### 4. 7 モジュール説明

損傷プログラム本体であるVisual Basic言語で記述された個々の関数の説明を行なう。

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	SonshoD			
概要	<p>損傷解析プログラム・メインルーチン          ① 試験条件、② 解析結果(FINAS/RANGEの出力データ)を与える事で          評価法及び鋼種に応じた損傷計算を行なう。          入出力データの取り扱いは、表計算ソフトMS-EXCELが行なう。</p>			
使用モジュール	Initial、Input_Data_1、Form_load、Allmtd_proc、Excel_Restart			
引数	引数名	I/O	型	意味
機能詳細	<p>表計算ソフト(MS-EXCEL)との入出力データの受け渡し、各評価法の処理、材料データ、共通ライブラリ(FORTRAN-DLLファイル)の制御を行っている。          以下のオプションに対応している。          15評価法(TTS-DS・BDS・DDS・TYPE・EQEF・PNC・INELAST等)          6鋼種(SUS304・SUS316・SCMV4・STBA24・改良9CR-1MO等)          入出力データの取り扱いは、表計算ソフトMS-EXCELが行ない          プロセス間通信によって損傷計算プログラムにデータを受け渡す。          そのためファイルを介さず計算結果データを即格納化することが出来る。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Allmtd_proc			
概要	各評価法による解析実行の制御を行う			
使用モジュール	Msg_proc、Input_Data_2、Bscvl、Ttsdsp、Tdsout、Bdsp、Bdsout、Bdsmdp、Bmdout、Ddsp、Ddsout、Ddswldp、Dwdout、Type1p、Typout、Type2p、Eqefp、Eqfout、Eqfwdp、Ewdout、Pncp、Pncout、Eqef15p、Eqfout、Eqef167p、Inelastic、Sieq167p、Eqef16wp、Outdat			
引数	引数名	I/O	型	意味
機能詳細	表計算ソフト(MS-EXCEL)とのDDE通信処理、処理状況表示、解析データ入力、基本変数算出用ライブラリ算出、各評価法による損傷値数値解析を制御する。入力データは、50行単位で処理される。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Bdsp			
概要	評価法：BDSによる損傷解析の制御を行う			
使用モジュール	Mset、Calcnu、Alpha、Prin1、Prin2、Ftres、Nmax、Nmin、Rlx1、FnuBr2、Dfcurv、Dstar、Ddstar、Sepr			
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Output	単精度浮動小数点*4B	表面応力
	Sn	Output	単精度浮動小数点*4B	等価線形応力
	K	Output	単精度浮動小数点*4B	応力集中係数
	Qn	Output	単精度浮動小数点*4B	弾性追従係数
	Ked	Output	単精度浮動小数点*4B	K e' 係数
	Neuber	Output	単精度浮動小数点*4B	NEUBER則
	Ke	Output	単精度浮動小数点*4B	歪集中係数
	Ef	Output	単精度浮動小数点*4B	熱ピーク歪
	Et	Output	単精度浮動小数点*4B	全歪範囲
	Df	Output	単精度浮動小数点*4B	D f
	Dc	Output	単精度浮動小数点*4B	D c
	D	Output	単精度浮動小数点*4B	D
	Dcr	Output	単精度浮動小数点*4B	D c r
機能詳細	歪範囲の計算、最高金属温度による $\epsilon_n$ 、 $S_{rh}$ の計算、コールド温度による $S_{rc}$ の計算、疲労損傷の計算、クリープ損傷の計算、許容値の計算等を行う。 Spx、Sn、K、Qn、Ked等の変数は、構造体「Allmtd.」によってグループ化され外部変数として各プログラム間で受け渡される。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Bdsout			
概要	評価法：BDSの結果データをDDE通信用の変数に格納する			
使用モジュール				
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Input	単精度浮動小数点*4B	表面応力
	Sn	Input	単精度浮動小数点*4B	等価線形応力
	K	Input	単精度浮動小数点*4B	応力集中係数
	Qn	Input	単精度浮動小数点*4B	弾性追従係数
	Ked	Input	単精度浮動小数点*4B	K'e' 係数
	Neuber	Input	単精度浮動小数点*4B	NEUBER則
	Ke	Input	単精度浮動小数点*4B	歪集中係数
	Ef	Input	単精度浮動小数点*4B	熱ピーク歪
	Et	Input	単精度浮動小数点*4B	全歪範囲
	Df	Input	単精度浮動小数点*4B	D f
	Dc	Input	単精度浮動小数点*4B	D c
	D	Input	単精度浮動小数点*4B	D
	Dcr	Input	単精度浮動小数点*4B	D c r
	Grid1.Text	Output	TEXT7°0ハ°ティ	DDE通信用ハッファ
機能詳細	<p>評価法：BDSの結果データをDDE通信用の変数(Grid.Text7°0ハ°ティ)に格納する。</p> <p>主記憶上の構造体に格納してある「BDS」の損傷結果データをDDE通信用ハッファ・エリアForm1/Grid/Text7°0ハ°ティに代入する。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Bdsmdp			
概要	評価法：BDS-MDによる損傷解析の制御を行う			
使用モジュール	Mset、Calcnu、Alpha、Prin1、Prin2、Ftres、Nmax、Rlx1、Fnubr2、Anf、Sepr			
引数	引数名	I/O	型	意味
	lcount Allmtd. Spx Sn K Qn Ked Neuber Ke Ef Et Si Qf Df Dc D Dcr	Input  Output Output Output Output Output Output Output Output Output Output Output Output Output Output Output Output	整数型*2B 構造体名 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B	Excelシート・オフセット行数 損傷結果データ格納用 表面応力 等価線形応力 応力集中係数 弾性追従係数 K e' 係数 NEUBER則 歪集中係数 熱ピーク歪 全歪範囲 初期応力 弾性追従係数 D f D c D D c r
機能詳細	<p>評価法：BDS-MDによる損傷解析の制御を行う 歪範囲の計算、最高金属温度による<math>\epsilon_n</math>、<math>S_{rh}</math>の計算、 コールド温度による<math>S_{rc}</math>の計算、疲労損傷の計算、 クリープ損傷の計算、許容値の計算等を行う。 Spx、Sn、K、Qn、Ked等の変数は、構造体「Allmtd」によってグループ化 され外部変数として各プログラム間で受け渡される。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Bmdout			
概要	評価法：BDS-MDによる損傷解析の制御を行う			
使用モジュール				
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Input	単精度浮動小数点*4B	表面応力
	Sn	Input	単精度浮動小数点*4B	等価線形応力
	K	Input	単精度浮動小数点*4B	応力集中係数
	Qn	Input	単精度浮動小数点*4B	弾性追従係数
	Ked	Input	単精度浮動小数点*4B	K e' 係数
	Neuber	Input	単精度浮動小数点*4B	NEUBER則
	Ke	Input	単精度浮動小数点*4B	歪集中係数
	Ef	Input	単精度浮動小数点*4B	熱ピーク歪
	Et	Input	単精度浮動小数点*4B	全歪範囲
	Si	Input	単精度浮動小数点*4B	初期応力
	Qf	Input	単精度浮動小数点*4B	弾性追従係数
	Df	Input	単精度浮動小数点*4B	D f
	Dc	Input	単精度浮動小数点*4B	D c
	D	Input	単精度浮動小数点*4B	D
	Dcr	Input	単精度浮動小数点*4B	D c r
	Grid1.Text	Output	TEXT7°08°テ	DDE通信用バッファ
機能詳細	評価法：BDS-MDの結果データをDDE通信用の変数 (Grid.Text7°08°テ)に格納する。 主記憶上の構造体に格納してある「BDS」の損傷結果データを DDE通信用バッファ・エリアForm1/Grid/Text7°08°テに代入する。			
備考				



言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Ddsp			
概要	評価法：DDSによる損傷解析の制御を行う			
使用モジュール	Mset、Calcnu、Alpha、Ftres、Prin1、Prin2、Smax、Rlx1、Fnumber、Anf、Sepr			
引数	引数名	I/O	型	意味
	Icount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Output	単精度浮動小数点*4B	表面応力
	Sn	Output	単精度浮動小数点*4B	等価線形応力
	K	Output	単精度浮動小数点*4B	応力集中係数
	Qn	Output	単精度浮動小数点*4B	弾性追従係数
	Ked	Output	単精度浮動小数点*4B	K e' 係数
	Neuber	Output	単精度浮動小数点*4B	NEUBER則
	Ke	Output	単精度浮動小数点*4B	歪集中係数
	Ef	Output	単精度浮動小数点*4B	熱ピーク歪
	Et	Output	単精度浮動小数点*4B	全歪範囲
	Si	Output	単精度浮動小数点*4B	初期応力
	Qf	Output	単精度浮動小数点*4B	弾性追従係数
	Df	Output	単精度浮動小数点*4B	D f
	Dc	Output	単精度浮動小数点*4B	D c
	D	Output	単精度浮動小数点*4B	D
機能詳細	<p>評価法：DDSによる損傷解析の制御を行う                  歪範囲の計算、最高金属温度による<math>\epsilon_n</math>、<math>S_{rh}</math>の計算                  最高金属温度、クリープ損傷の計算、                  クリープ疲労損傷の計算を行う。                  Spx、Sn、K、Qn、Ked等の変数は、構造体「Allmtd」によってグループ化                  され外部変数として各プログラム間で受け渡される。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Ddsout			
概要	評価法：DDSの結果データをDDE通信用の変数に格納する			
使用モジュール				
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Input	単精度浮動小数点*4B	表面応力
	Sn	Input	単精度浮動小数点*4B	等価線形応力
	K	Input	単精度浮動小数点*4B	応力集中係数
	Qn	Input	単精度浮動小数点*4B	弾性追従係数
	Ked	Input	単精度浮動小数点*4B	K e' 係数
	Neuber	Input	単精度浮動小数点*4B	NEUBER則
	Ke	Input	単精度浮動小数点*4B	歪集中係数
	Ef	Input	単精度浮動小数点*4B	熱ピーク歪
	Et	Input	単精度浮動小数点*4B	全歪範囲
	Si	Input	単精度浮動小数点*4B	初期応力
	Qf	Input	単精度浮動小数点*4B	弾性追従係数
	Df	Input	単精度浮動小数点*4B	D f
	Dc	Input	単精度浮動小数点*4B	D c
	D	Input	単精度浮動小数点*4B	D
	Grid1.Text	Output	TEXT7°0ハ°ティ	DDE通信用バッファ
機能詳細	<p>評価法：DDSの結果データをDDE通信用の変数(Grid.Text7°0ハ°ティ)に格納する。</p> <p>主記憶上の構造体に格納してある「BDS」の損傷結果データをDDE通信用バッファ・エリアForm1/Grid/Text7°0ハ°ティに代入する。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Ddswl dp			
概要	評価法：DDS-WELDによる損傷解析の制御を行う			
使用モジュール	Mset、Calcnu、Alpha、Ftres、Prin1、Prin2、Smax、Rlx1、Fnumber、Anf、Sepr			
引数	引数名	I/O	型	意味
	lcount Allmtd. Spx Sn K Ked Neuber Ke Ef Et Si Qw  Qeff Qbm Df Dc D	Input  Output Output Output Output Output Output Output Output Output Output  Output Output Output Output Output	整数型*28 構造体名 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B  単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B	Excelシート・オフセット行数 損傷結果データ格納用 表面応力 等価線形応力 応力集中係数 K <sub>e</sub> 係数 NEUBER則 歪集中係数 熱ピーク歪 全歪範囲 初期応力 熔接金属に対する 弾性追従係数 弾性追従係数 母材の弾性追従係数 Df Dc D
機能詳細	<p>評価法：DDSによる損傷解析の制御を行う                  歪範囲の計算、最高金属温度による<math>\epsilon_n</math>、<math>S_{rh}</math>の計算                  コールド温度による<math>S_{rc}</math>の計算、疲労損傷の計算、                  最高金属温度、クリープ損傷の計算、                  クリープ疲労損傷の計算を行う。                  Sp<sub>x</sub>、Sn、K、Q<sub>n</sub>、Ked等の変数は、構造体「Allmtd」によって                  グループ化され外部変数として各プログラム間で受け渡される。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Dwdout			
概要	評価法：DDS-WELDの結果データをDDE通信用の変数に格納			
使用モジュール				
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オブジェクト行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Input	単精度浮動小数点*4B	表面応力
	Sn	Input	単精度浮動小数点*4B	等価線形応力
	K	Input	単精度浮動小数点*4B	応力集中係数
	Ked	Input	単精度浮動小数点*4B	K <sub>e'</sub> 係数
	Neuber	Input	単精度浮動小数点*4B	NEUBER則
	Ke	Input	単精度浮動小数点*4B	歪集中係数
	Ef	Input	単精度浮動小数点*4B	熱ピーク歪
	Et	Input	単精度浮動小数点*4B	全歪範囲
	Si	Input	単精度浮動小数点*4B	初期応力
	Qw	Input	単精度浮動小数点*4B	溶接金属に対する 弾性追従係数
	Qeff	Input	単精度浮動小数点*4B	弾性追従係数
	Qbm	Input	単精度浮動小数点*4B	母材に対する 弾性追従係数
	Df	Input	単精度浮動小数点*4B	D f
	Dc	Input	単精度浮動小数点*4B	D c
	D	Input	単精度浮動小数点*4B	D
	Gridl. Text	Output	TEXT7°ロハ°ティ	DDE通信用ハッパ
機能詳細	<p>評価法：DDS-WELDの結果データをDDE通信用の変数(Grid. Text7°ロハ°ティ)に格納する。                  主記憶上の構造体に格納してある「BDS」の損傷結果データをDDE通信用ハッパ・エリアForm1/Grid/Text7°ロハ°ティに代入する。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Eqefp			
概要	評価法：EQEFによる損傷解析の制御を行う			
使用モジュール	Mset、Calcnu、Alpha、Ftres、Nmax、Nmin、Rlx1、Prin1、Prin2 Anf、Sepr			
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オブジェクト行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Output	単精度浮動小数点*4B	表面応力
	Sn	Output	単精度浮動小数点*4B	等価線形応力
	K	Output	単精度浮動小数点*4B	応力集中係数
	Ked	Output	単精度浮動小数点*4B	K e' 係数
	Ke	Output	単精度浮動小数点*4B	歪集中係数
	Kedg	Output	単精度浮動小数点*4B	総体的歪集中係数
	Kedl	Output	単精度浮動小数点*4B	局所的歪集中係数
	Ef	Output	単精度浮動小数点*4B	熱ピーク歪
	Et	Output	単精度浮動小数点*4B	全歪範囲
	Si	Output	単精度浮動小数点*4B	初期応力
	Qn	Output	単精度浮動小数点*4B	弾性追従係数
	Ql	Output	単精度浮動小数点*4B	局所的弾性追従係数
	Qnu	Output	単精度浮動小数点*4B	多軸効果を現す 弾性追従係数
	Qeff	Output	単精度浮動小数点*4B	弾性追従係数
	Df	Output	単精度浮動小数点*4B	D f
	Dc	Output	単精度浮動小数点*4B	D c
	D	Output	単精度浮動小数点*4B	D
	Dcr	Output	単精度浮動小数点*4B	D c r
機能詳細	評価法：EQEFによる損傷解析の制御を行う 歪範囲の計算、熱ピークを含むピーク応力強さの計算、 熱ピークを含まないピーク応力強さの計算、 公称応力強さの計算、ピーク応力を含む応力集中係数の計算 Spx、Sn、K、Qn、Ked等の変数は、構造体「Allmtd」によってグループ化 され外部変数として各プログラム間で受け渡される。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Eqfout			
概要	評価法：EQEFの結果データをDDE通信用の変数に格納する			
使用モジュール				
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Input	単精度浮動小数点*4B	表面応力
	Sn	Input	単精度浮動小数点*4B	等価線形応力
	K	Input	単精度浮動小数点*4B	応力集中係数
	Ked	Input	単精度浮動小数点*4B	Ke' 係数
	Ke	Input	単精度浮動小数点*4B	歪集中係数
	Kedg	Input	単精度浮動小数点*4B	総体的歪集中係数
	Kedl	Input	単精度浮動小数点*4B	局所的歪集中係数
	Ef	Input	単精度浮動小数点*4B	熱ピーク歪
	Et	Input	単精度浮動小数点*4B	全歪範囲
	Si	Input	単精度浮動小数点*4B	初期応力
	Qn	Input	単精度浮動小数点*4B	弾性追従係数
	Ql	Input	単精度浮動小数点*4B	局所的弾性追従係数
	Qnu	Input	単精度浮動小数点*4B	多軸効果を現す 弾性追従係数
	Qeff	Input	単精度浮動小数点*4B	弾性追従係数
	Df	Input	単精度浮動小数点*4B	D f
	Dc	Input	単精度浮動小数点*4B	D.c
	D	Input	単精度浮動小数点*4B	D
	Dcr	Input	単精度浮動小数点*4B	D c r
	Grid1.Text	Output	TEXT7*08*ティ	DDE通信用ハック
機能詳細	評価法：EQEFの結果データをDDE通信用の変数 (Grid.Text7*08*ティ)に格納する。 主記憶上の構造体に格納してある「BDS」の損傷結果データを DDE通信用ハック・エリアForm1/Grid/Text7*08*ティに代入する。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Eqef15p			
概要	評価法：EQEF15による損傷解析の制御を行う			
使用モジュール	Mset、Calcnu、Alpha、Ftres、Smax、Nmin、Rlx1、Prin1、Prin2 Anf、Sepr			
引数	引数名	I/O	型	意味
	Icount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Output	単精度浮動小数点*4B	表面応力
	Sn	Output	単精度浮動小数点*4B	等価線形応力
	K	Output	単精度浮動小数点*4B	応力集中係数
	Ked	Output	単精度浮動小数点*4B	K e' 係数
	Ke	Output	単精度浮動小数点*4B	歪集中係数
	Kedg	Output	単精度浮動小数点*4B	総体的歪集中係数
	Kedl	Output	単精度浮動小数点*4B	局所的歪集中係数
	Ef	Output	単精度浮動小数点*4B	熱ピーク歪
	Et	Output	単精度浮動小数点*4B	全歪範囲
	Si	Output	単精度浮動小数点*4B	初期応力
	Qn	Output	単精度浮動小数点*4B	弾性追従係数
	Ql	Output	単精度浮動小数点*4B	局所的弾性追従係数
	Qnu	Output	単精度浮動小数点*4B	多軸効果を現す 弾性追従係数
	Qeff.	Output	単精度浮動小数点*4B	弾性追従係数
	Df	Output	単精度浮動小数点*4B	D f
	Dc	Output	単精度浮動小数点*4B	D c
	D	Output	単精度浮動小数点*4B	D
	Dcr	Output	単精度浮動小数点*4B	D c r
機能詳細	<p>評価法：EQEFによる損傷解析の制御を行う 歪範囲の計算、熱ピークを含むピーク応力強さの計算、 熱ピークを含まないピーク応力強さの計算、 公称応力強さの計算、ピーク応力を含む応力集中係数の計算 等の計算を行う。</p> <p>Spx、Sn、K、Qn、Ked等の変数は、構造体「Allmtd」によってグループ化され外部変数として各プログラム間で受け渡される。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/マクロ形式			
モジュール名	Eqef167p			
概要	評価法：EQEF167による損傷解析の制御を行う			
使用モジュール	Mset、Calcnu、Alpha、Ftres、Nmax、Rlx1、Prin1、Prin2 Anf、Sepr			
引数	引数名	I/O	型	意味
	Icount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Output	単精度浮動小数点*4B	表面応力
	Sn	Output	単精度浮動小数点*4B	等価線形応力
	K	Output	単精度浮動小数点*4B	応力集中係数
	Ked	Output	単精度浮動小数点*4B	Ke' 係数
	Ke	Output	単精度浮動小数点*4B	歪集中係数
	Kedg	Output	単精度浮動小数点*4B	総体的歪集中係数
	Kedl	Output	単精度浮動小数点*4B	局所的歪集中係数
	Ef	Output	単精度浮動小数点*4B	熱ピーク歪
	Et	Output	単精度浮動小数点*4B	全歪範囲
	Si	Output	単精度浮動小数点*4B	初期応力
	Qn	Output	単精度浮動小数点*4B	弾性追従係数
	Ql	Output	単精度浮動小数点*4B	局所的弾性追従係数
	Qnu	Output	単精度浮動小数点*4B	多軸効果を現す 弾性追従係数
	Qeff	Output	単精度浮動小数点*4B	弾性追従係数
	Df	Output	単精度浮動小数点*4B	D f
	Dc	Output	単精度浮動小数点*4B	D c
	D	Output	単精度浮動小数点*4B	D
	Dcr	Output	単精度浮動小数点*4B	D c r
機能詳細	<p>評価法：EQEFによる損傷解析の制御を行う 歪範囲の計算、熱ピークを含むピーク応力強さの計算、 熱ピークを含まないピーク応力強さの計算、 公称応力強さの計算、ピーク応力を含む応力集中係数の計算 等の計算を行う。</p> <p>Spx、Sn、K、Qn、Ked等の変数は、構造体「Allmtd」によってグローバル化され外部変数として各プログラム間で受け渡される。</p>			
備考				



言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/マクロ形式			
モデル名	Eqefwdp			
概要	評価法：EQEF-WELDによる損傷解析の制御を行う			
使用モデル	Mset、Calcnu、Alpha、Ftres、Smax、Rlx1、Prin1、Prin2 Anf、Sepr			
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd		構造体名	損傷結果データ格納用
	Spx	Output	単精度浮動小数点*4B	表面応力
	Sn	Output	単精度浮動小数点*4B	等価線形応力
	K	Output	単精度浮動小数点*4B	応力集中係数
	Ke	Output	単精度浮動小数点*4B	歪集中係数
	Kedg	Output	単精度浮動小数点*4B	総体的歪集中係数
	Kedl	Output	単精度浮動小数点*4B	局所的歪集中係数
	Ef	Output	単精度浮動小数点*4B	熱ピーク歪
	Et	Output	単精度浮動小数点*4B	全歪範囲
	Si	Output	単精度浮動小数点*4B	初期応力
	Qn	Output	単精度浮動小数点*4B	弾性追従係数
	Qw	Output	単精度浮動小数点*4B	熔接金属に対する 弾性追従係数
	Ql	Output	単精度浮動小数点*4B	局所的弾性追従係数
	Qnu	Output	単精度浮動小数点*4B	多軸効果を現す 弾性追従係数
	Qeff	Output	単精度浮動小数点*4B	弾性追従係数
	Df	Output	単精度浮動小数点*4B	D f
	Dc	Output	単精度浮動小数点*4B	D c
	D	Output	単精度浮動小数点*4B	D
	Dcr	Output	単精度浮動小数点*4B	D c r
機能詳細	<p>評価法：EQEFによる損傷解析の制御を行う 歪範囲の計算、熱ピークを含むピーク応力強さの計算、 熱ピークを含まないピーク応力強さの計算、 公称応力強さの計算、ピーク応力を含む応力集中係数の計算 等の計算を行う。</p> <p>Spx、Sn、K、Qn、Ked等の変数は、構造体「Allmtd」によってグローバル化され外部変数として各プログラム間で受け渡される。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/マクロ形式			
モジュール名	Ewdout			
概要	評価法：EQEF-WELDの結果データをDDE通信用の変数に格納する			
使用モジュール				
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Input	単精度浮動小数点*4B	表面応力
	Sn	Input	単精度浮動小数点*4B	等価線形応力
	K	Input	単精度浮動小数点*4B	応力集中係数
	Ke	Input	単精度浮動小数点*4B	歪集中係数
	Kedg	Input	単精度浮動小数点*4B	総体的歪集中係数
	Kedl	Input	単精度浮動小数点*4B	局所的歪集中係数
	Ef	Input	単精度浮動小数点*4B	熱ピーク歪
	Et	Input	単精度浮動小数点*4B	全歪範囲
	Si	Input	単精度浮動小数点*4B	初期応力
	Qn	Input	単精度浮動小数点*4B	弾性追従係数
	Qw	Input	単精度浮動小数点*4B	溶接金属に対する 弾性追従係数
	Ql	Input	単精度浮動小数点*4B	局所的弾性追従係数
	Qnu	Input	単精度浮動小数点*4B	多軸効果を現す 弾性追従係数
	Qeff	Input	単精度浮動小数点*4B	弾性追従係数
	Df	Input	単精度浮動小数点*4B	D f
	Dc	Input	単精度浮動小数点*4B	D c
	D	Input	単精度浮動小数点*4B	D
	Dcr	Input	単精度浮動小数点*4B	D c r
	Gridl.Text	Output	TEXT7°08°ティ	DDE通信用ハック77
機能詳細	評価法：EQEF-WELDの結果データをDDE通信用の変数 (Grid.Text7°08°ティ)に格納する。 主記憶上の構造体に格納してある「BDS」の損傷結果データを DDE通信用ハック77・エリアForm1/Grid/Text7°08°ティに代入する。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モデル名	Eqef16wp			
概要	評価法：EQEF167-WELDによる損傷解析の制御を行う			
使用モデル	Mset、Calcnu、Alpha、Ftres、Smax、Rlx1、Prin1、Prin2 Anf、Sepr			
引数	引数名	I/O	型	意味
	Icount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Output	単精度浮動小数点*4B	表面応力
	Sn	Output	単精度浮動小数点*4B	等価線形応力
	K	Output	単精度浮動小数点*4B	応力集中係数
	Ke	Output	単精度浮動小数点*4B	歪集中係数
	Kedg	Output	単精度浮動小数点*4B	総体的歪集中係数
	Kedl	Output	単精度浮動小数点*4B	局所的歪集中係数
	Ef	Output	単精度浮動小数点*4B	熱ピーク歪
	Et	Output	単精度浮動小数点*4B	全歪範囲
	Si	Output	単精度浮動小数点*4B	初期応力
	Qn	Output	単精度浮動小数点*4B	弾性追従係数
	Qw	Output	単精度浮動小数点*4B	熔接金属に対する 弾性追従係数
	Ql	Output	単精度浮動小数点*4B	局所的弾性追従係数
	Qnu	Output	単精度浮動小数点*4B	多軸効果を現す 弾性追従係数
	Qeff	Output	単精度浮動小数点*4B	弾性追従係数
	Df	Output	単精度浮動小数点*4B	D f
	Dc	Output	単精度浮動小数点*4B	D c
	D	Output	単精度浮動小数点*4B	D
	Dcr	Output	単精度浮動小数点*4B	D c r
機能詳細	<p>評価法：EQEFによる損傷解析の制御を行う 歪範囲の計算、熱ピークを含むピーク応力強さの計算、 熱ピークを含まないピーク応力強さの計算、 公称応力強さの計算、ピーク応力を含む応力集中係数の計算 等の計算を行う。</p> <p>Spx, Sn, K, Qn, Ked等の変数は、構造体「Allmtd」によってグローバル化され外部変数として各プログラム間で受け渡される。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows				
関数種類	Sub/サブルーチン形式				
モジュール名	Form_load				
概要	処理状況表示ルーチン				
使用モジュール					
引数	引数名	I/O	型		意味
	N	Input	整数型*2B		評価法の番号
機能詳細	プログラム起動時に、実行する評価法名を表示する。				
備考					

言語	Microsoft Visual Basic2.0 for Windows				
関数種類	Sub/サブルーチン形式				
モジュール名	Msg_proc0				
概要	処理状況表示ルーチン				
使用モジュール					
引数	引数名	I/O	型		意味
	Message	Input	文字型	String	表示したい メッセージ文字列
機能詳細	メッセージ・ボックス1に任意のメッセージを表示する。				
備考					

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Msg_proc1			
概要	処理状況表示ルーチン			
使用モジュール				
引数	引数名	I/O	型	
	Message	Input	文字型	String
				意味 表示したいメッセージ 文字列
機能詳細	開発時に処理途中の各種データをメッセージ・ボックスに表示する。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Msg_proc			
概要	処理状況表示ルーチン			
使用モジュール				
引数	引数名	I/O	型	意味
	N	Input	整数型*2B	現在の処理状況を示す番号 解析データ(入力)の 行数を50で割った 時の数値 Lcnt=n, i=1~n 、現在のiの値
	Lcnt	Input	整数型*2B	
	i	Input	整数型*2B	
機能詳細	<p>損傷解析プログラムの処理状況を表示する関数。 以下の処理を行っている。</p> <p>解析データ入力中! 基本変数設定中! 損傷値解析中! DDE通信準備中! DDE通信中!</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	inelastic			
概要	評価法：INELASTICによる損傷解析の制御を行う			
使用モジュール	Mset, Calcnu, Anf, Sepr, Rlx1			
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Kep	Output	単精度浮動小数点*4B	歪集中係数
	Et	Output	単精度浮動小数点*4B	全歪範囲
	Df	Output	単精度浮動小数点*4B	D f
	Dc	Output	単精度浮動小数点*4B	D c
	D	Output	単精度浮動小数点*4B	D
機能詳細	<p>評価法：INELASTICによる損傷解析の制御を行う                  歪範囲の計算、疲労損傷の計算、クリープ損傷の計算                  、クリープ疲労損傷の計算を行う。                  Kep、Et、Df、Dc、Dの変数は、構造体「Allmtd.」によってグループ化                  され外部変数として各プログラム間で受け渡される。</p>			
備考				



言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Initial			
概要	全評価法名称を変数に格納する			
使用モジュール				
引数	引数名	I/O	型	意味
	Method_name	Output	文字型 String	評価法名称
機能詳細	全評価法名称を変数に格納する。 BDS・DDS・EQEF・TYPE等の評価法が15ある。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Input_Data_1			
概要	<p>損傷解析処理を実行するために必要な入力データ「試験条件データ」を表計算ソフトMS-EXCELからDDE通信によって受け取る。 そして対応した各変数にデータをセットする。</p>			
使用モジュール				
引数	引数名	I/O	型	意味
	Cond.		構造体名	試験条件シート要素
	Material	Output	文字型 String	材料名
	Tempc	Output	単精度浮動小数点*4B	Cold温度
	TempH	Output	単精度浮動小数点*4B	Hot 温度
	Hldtmc	Output	単精度浮動小数点*4B	Cold保持時間
	Hldtmh	Output	単精度浮動小数点*4B	Hot 保持時間
	Mecha	Output	単精度浮動小数点*4B	機械荷重
	Cycle	Output	整数型*2B	サイクル数
	MethodName	Output	文字型 String	評価法名称
	MethodNo	Output	整数型*2B	評価法番号
	AnalSheetN	Output	文字型 String	解析シート名称
	AnalSheetR	Output	整数型*2B	解析シート行数
	AnalSheetC	Output	整数型*2B	解析シート桁数
	OutSheetName	Output	文字型 String	出力シート名称
	Imate	Output	整数型*2B	材料番号
機能詳細	<p>表計算ソフトであるMS-EXCELとDDE通信を行う。 試験条件シートデータを受け取る。(試験条件. xjs) MS-EXCEL上でのファイル名は固定である。 STAR-NETシステムで実行する場合/単体で実行する場合には拡張子が異なる。(試験条件. XJS/試験条件. CND) 材料名、評価法名、各種試験条件データを受け取り、対応する変数に格納する。</p> <p>※ 表計算ソフト(MS-EXCEL)は、予め起動していなければならない。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Input_Data_2			
概要	DDE通信(入力)、Range解析結果シートデータを取得する、対EXCEL/解析. An?			
使用モジュール				
引数	引数名	I/O	型	意味
	Analy.		構造体名	Range解析結果シート要素
	R	Output	単精度浮動小数点*4B	R座標 (起点座標のみ)
	Z	Output	単精度浮動小数点*4B	Z座標 ( " )
	H	Output	単精度浮動小数点*4B	θ座標( " )
	Tmpmax	Output	単精度浮動小数点*4B	最高金属温度
	Ptime	Output	単精度浮動小数点*4B	評価時刻
	Temp	Output	単精度浮動小数点*4B	表面温度
	Ltemp	Output	単精度浮動小数点*4B	等価線形温度
	Mtemp	Output	単精度浮動小数点*4B	平均温度
	Strs	Output	単精度浮動小数点*4B	表面応力
	Lstrs	Output	単精度浮動小数点*4B	等価線形応力
	Mstrs	Output	単精度浮動小数点*4B	膜応力
	Ee	Output	単精度浮動小数点*4B	弾性歪
	Ep	Output	単精度浮動小数点*4B	塑性歪
	Ec	Output	単精度浮動小数点*4B	クリープ歪
	Q	Output	単精度浮動小数点*4B	Q
機能詳細	表計算ソフトであるMS-EXCELとDDE通信を行う。 Range解析結果シートデータを受け取る。(解析. An?) 起点座標、最高金属温度、評価時刻、表面温度、等価線形応力、 弾性歪等の解析データを受け取る。			
備考				

言語	Microsoft Visual Basic2.0 for Windows																			
関数種類	Sub/サブルーチン形式																			
モジュール名	Mthd																			
概要	評価法名を判定し、対応する番号に変換する。 15評価法(キーワード固定)																			
使用モジュール																				
引数	引数名	I/O	型	意味																
	Cond. MethodName	Input	構造体 文字型 String	試験条件シート 評価法名称																
	Cond. MethodNo	Output	構造体 整数型*2B	試験条件シート 評価法番号																
機能詳細	<p>入力データである試験条件シートデータに含まれている「評価法名称」を判定し、対応する番号を構造体変数に格納する。 評価法の数は、15で以下に示す通りである。</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%;">1. TTSDS</td> <td style="width: 50%;">9. EQEFWD</td> </tr> <tr> <td>2. BDS</td> <td>10. PNC</td> </tr> <tr> <td>3. BDSMD</td> <td>11. EQEF15</td> </tr> <tr> <td>4. DDS</td> <td>12. EQEF167</td> </tr> <tr> <td>5. DDSMDWLD</td> <td>13. INELAST</td> </tr> <tr> <td>6. TYPE1</td> <td>14. SIEQ167</td> </tr> <tr> <td>7. TYPE2</td> <td>15. EQEF167W</td> </tr> <tr> <td>8. EQEF</td> <td></td> </tr> </table>				1. TTSDS	9. EQEFWD	2. BDS	10. PNC	3. BDSMD	11. EQEF15	4. DDS	12. EQEF167	5. DDSMDWLD	13. INELAST	6. TYPE1	14. SIEQ167	7. TYPE2	15. EQEF167W	8. EQEF	
1. TTSDS	9. EQEFWD																			
2. BDS	10. PNC																			
3. BDSMD	11. EQEF15																			
4. DDS	12. EQEF167																			
5. DDSMDWLD	13. INELAST																			
6. TYPE1	14. SIEQ167																			
7. TYPE2	15. EQEF167W																			
8. EQEF																				
備考																				

言語	Microsoft Visual Basic2.0 for Windows																								
関数種類	Sub/サブルーチン形式																								
モジュール名	Mtrl																								
概要	材料名を判定し、対応する番号に変換する。6材料(キーワード固定)																								
使用モジュール																									
引数	引数名	I/O	型	意味																					
	Cond. Material	Input	構造体 文字型	材料名																					
	Mc. Im	Output	構造体 整数型*2B	材料番号																					
機能詳細	<p>入力データである試験条件シートデータに含まれている「材料名称」を判定し、対応する番号を構造体変数に格納する。 材料の数は、6材料ある。</p> <table border="0"> <tr> <td>鋼種</td> <td>キーワード</td> <td>鋼種番号</td> </tr> <tr> <td>鋼種:SUS304</td> <td>SUS304</td> <td>304</td> </tr> <tr> <td>鋼種:SUS316</td> <td>SUS316</td> <td>316</td> </tr> <tr> <td>鋼種:SCMV4</td> <td>SCMV4</td> <td>2250</td> </tr> <tr> <td>鋼種:STBA24</td> <td>STBA24</td> <td>2251</td> </tr> <tr> <td>鋼種:改良9CR-1MO</td> <td>MOD9CR-1MO</td> <td>91</td> </tr> <tr> <td>鋼種:SUS316</td> <td>SUS316</td> <td>3160</td> </tr> </table>				鋼種	キーワード	鋼種番号	鋼種:SUS304	SUS304	304	鋼種:SUS316	SUS316	316	鋼種:SCMV4	SCMV4	2250	鋼種:STBA24	STBA24	2251	鋼種:改良9CR-1MO	MOD9CR-1MO	91	鋼種:SUS316	SUS316	3160
鋼種	キーワード	鋼種番号																							
鋼種:SUS304	SUS304	304																							
鋼種:SUS316	SUS316	316																							
鋼種:SCMV4	SCMV4	2250																							
鋼種:STBA24	STBA24	2251																							
鋼種:改良9CR-1MO	MOD9CR-1MO	91																							
鋼種:SUS316	SUS316	3160																							
備考																									

言語	Microsoft Visual Basic 2.0 for Windows			
関数種類	Sub/マクロ形式			
モジュール名	Outdat			
概要	<p>損傷解析結果データを表計算ソフトMS-EXCELにDDE通信で送信する。 そして対応した各変数にデータをセットする。</p>			
使用モジュール				
引数	引数名	I/O	型	意味
	Ino lcount Colum	Input Input Input	整数型*2B 整数型*2B 整数型*2B	Excelシート・開始行数 出力データ行数 出力データ最終桁数
機能詳細	<p>表計算ソフトであるMS-EXCELとDDE通信を行う。 指定した評価法による損傷解析結果データをEXCELシートに転送する。 その際、50行単位で転送する。 VisualBasic内部にあるテキスト情報格納領域に損傷結果データを保持する。 そこからDDE通信によってMS-EXCELの決まったシートにデータを受け渡す。 出力シート名は、任意である。(fe)損傷値.D??</p> <p>※ 表計算ソフト(MS-EXCEL)は、予め起動しておかなければならない。 そして出力シートも「試験条件シート」で指定した出力シート名と同一であり、更にオープンされていなければならない。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モデル名	Pncp			
概要	評価法：PNCによる損傷解析の制御を行う			
使用モデル	Mset、Calcnu、Alpha、Ftres、Nmax、Nmin、Rlx1、Prin1、Prin2 Anf、Sepr			
引数	引数名	I/O	型	意味
	Icount Allmtd. Spx Sn K Ked Ke Kedg Kedl Ef Et Si Qn Ql Qnu  Qeff Df Dc D Dcr	Input  Output Output Output Output Output Output Output Output Output Output Output Output Output Output  Output Output Output Output Output	整数型*2B 構造体名 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B  単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B	Excelシート・オフセット行数 損傷結果データ格納用 表面応力 等価線形応力 応力集中係数 Ke' 係数 歪集中係数 総合的歪集中係数 局所的歪集中係数 熱ピーク歪 全歪範囲 初期応力 弾性追従係数 局所的弾性追従係数 多軸効果を現す 弾性追従係数 弾性追従係数 Df Dc D Dcr
機能詳細	<p>評価法：PNCによる損傷解析の制御を行う 歪範囲の計算、熱ピークを含むピーク応力強さの計算、 公称応力強さの計算、ピーク応力を含む応力集中係数の計算 等の計算を行う。</p> <p>中間保持考慮、弾性追従パラメータ最適化、ひずみ集中係数改良 EQEFに工学的安全係数追加</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/マクロ形式			
モジュール名	Pncout			
概要	評価法：PNCの結果データをDDE通信用の変数に格納する			
使用モジュール				
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オブジェクト行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spv	Input	単精度浮動小数点*4B	表面応力
	Sn	Input	単精度浮動小数点*4B	等価線形応力
	K	Input	単精度浮動小数点*4B	応力集中係数
	Ked	Input	単精度浮動小数点*4B	Ke' 係数
	Ke	Input	単精度浮動小数点*4B	歪集中係数
	Kedg	Input	単精度浮動小数点*4B	総体的歪集中係数
	Kedl	Input	単精度浮動小数点*4B	局所的歪集中係数
	Ef	Input	単精度浮動小数点*4B	熱ピーク歪
	Et	Input	単精度浮動小数点*4B	全歪範囲
	Si	Input	単精度浮動小数点*4B	初期応力
	Qn	Input	単精度浮動小数点*4B	弾性追従係数
	Ql	Input	単精度浮動小数点*4B	局所的弾性追従係数
	Qnu	Input	単精度浮動小数点*4B	多軸効果を現す 弾性追従係数
	Qeff	Input	単精度浮動小数点*4B	弾性追従係数
	Df	Input	単精度浮動小数点*4B	D f
	Dc	Input	単精度浮動小数点*4B	D c
	D	Input	単精度浮動小数点*4B	D
	Dcr	Input	単精度浮動小数点*4B	D c r
	Grid1.Text	Output	TEXTマクロ	DDE通信用マクロ
機能詳細	<p>評価法：PNCの結果データをDDE通信用の変数(Grid.Textマクロ)に格納する。</p> <p>主記憶上の構造体に格納してある「BDS」の損傷結果データをDDE通信用マクロ・エリアForm1/Grid/Textマクロに代入する。</p>			
備考				



言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Ttsdsp			
概要	評価法：TTS-D Sによる損傷解析の制御を行う			
使用モジュール	Mset、Calcnu、Rlx1、Anf、Sepr			
引数	引数名	I/O	型	意味
	lcount Allmtd. Kep Et Df Dc D	Input Output Output Output Output Output	整数型*2B 構造体名 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B	Excelシート・オフセット行数 損傷結果データ格納用 歪集中係数 全歪範囲 D f D c D
機能詳細	評価法：TTS-D Sによる損傷解析の制御を行う 歪範囲の計算、疲労損傷の計算、ファンクションANFの クリープ疲労損傷の計算			
備考				

言語	Microsoft Visual Basic 2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Tdsout			
概要	評価法：TTS-D Sの結果データをDDE通信用の変数に格納する			
使用モジュール				
引数	引数名	I/O	型	意味
	lcount Alimtd. Kep Et Df Dc D Grid1. Text	Input  Input Input Input Input Input Output	整数型*2B 構造体名 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B TEXTプロパティ	Excelシート・オフセット行数 損傷結果データ格納用 歪集中係数 全歪範囲 Df Dc D DDE通信用バッファ
機能詳細	評価法：TTS-D Sの結果データをDDE通信用の変数 (Grid. Textプロパティ) に格納する。 主記憶上の構造体に格納してある「BDS」の損傷結果データをDDE通信用バッファ・エリアForm1/Grid/Textプロパティに代入する。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/マクロ形式			
モジュール名	Sieq167p			
概要	評価法：S I E Q 1 6 7による損傷解析の制御を行う			
使用モジュール	Mset、Calcnu、Alpha、Ftres、Smax、Nmin、Rlx1、Prin1、Prin2 Anf、Sepr			
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Output	単精度浮動小数点*4B	表面応力
	Sn	Output	単精度浮動小数点*4B	等価線形応力
	K	Output	単精度浮動小数点*4B	応力集中係数
	Ked	Output	単精度浮動小数点*4B	K e' 係数
	Ke	Output	単精度浮動小数点*4B	歪集中係数
	Kedg	Output	単精度浮動小数点*4B	総体的歪集中係数
	Kedl	Output	単精度浮動小数点*4B	局所的歪集中係数
	Ef	Output	単精度浮動小数点*4B	熱ピーク歪
	Et	Output	単精度浮動小数点*4B	全歪範囲
	Si	Output	単精度浮動小数点*4B	初期応力
	Qn	Output	単精度浮動小数点*4B	弾性追従係数
	Ql	Output	単精度浮動小数点*4B	局所的弾性追従係数
	Qnu	Output	単精度浮動小数点*4B	多軸効果を現す 弾性追従係数
	Qeff	Output	単精度浮動小数点*4B	弾性追従係数
	Df	Output	単精度浮動小数点*4B	D f
	Dc	Output	単精度浮動小数点*4B	D c
	D	Output	単精度浮動小数点*4B	D
	Dcr	Output	単精度浮動小数点*4B	D c r
機能詳細	<p>評価法：S I E Q 1 6 7による損傷解析の制御を行う  歪範囲の計算、熱ピークを含むピーク応力強さの計算、  熱ピークを含まないピーク応力強さの計算、  公称応力強さの計算、ピーク応力を含む応力集中係数の計算  等の計算を行う。  Q = 1. 6 7</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/マクロ形式			
モジュール名	Type1p			
概要	評価法：TYPE1による損傷解析の制御を行う			
使用モジュール	Mset、Calcnu、Alpha、Ftres、Smax、Rlx1、Fnuber、Prin1、Prin2 Anf、Sepr			
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Output	単精度浮動小数点*4B	表面応力
	Sn	Output	単精度浮動小数点*4B	等価線形応力
	K	Output	単精度浮動小数点*4B	応力集中係数
	Qn	Output	単精度浮動小数点*4B	弾性追従係数
	Ked	Output	単精度浮動小数点*4B	K e' 係数
	Neuber	Output	単精度浮動小数点*4B	NEUBER則
	Ke	Output	単精度浮動小数点*4B	歪集中係数
	Ef	Output	単精度浮動小数点*4B	熱ピーク歪
	Et	Output	単精度浮動小数点*4B	全歪範囲
	Si	Output	単精度浮動小数点*4B	初期応力
	Qeff	Output	単精度浮動小数点*4B	弾性追従係数
	Df	Output	単精度浮動小数点*4B	D f
	Dc	Output	単精度浮動小数点*4B	D c
	D	Output	単精度浮動小数点*4B	D
機能詳細	<p>評価法：TYPE1による損傷解析の制御を行う 歪範囲の計算、最高金属温度による<math>\epsilon_n</math>、<math>S_r h</math>の計算、 ファンクションANFの計算機能の限界、 クリープ損傷の計算、DCN/DCPの計算、 クリープ疲労損傷の計算等を行う。</p> <p>中間保持考慮</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Typout			
概要	評価法：TYPE?の結果データをDDE通信用の変数に格納する			
使用モジュール				
引数	引数名	I/O	型	意味
	lcount	Input	整数型*2B	Excelシート・オフセット行数
	Allmtd.		構造体名	損傷結果データ格納用
	Spx	Input	単精度浮動小数点*4B	表面応力
	Sn	Input	単精度浮動小数点*4B	等価線形応力
	K	Input	単精度浮動小数点*4B	応力集中係数
	Qn	Input	単精度浮動小数点*4B	弾性追従係数
	Ked	Input	単精度浮動小数点*4B	Ke' 係数
	Neuber	Input	単精度浮動小数点*4B	NEUBER則
	Ke	Input	単精度浮動小数点*4B	歪集中係数
	Ef	Input	単精度浮動小数点*4B	熱ピーク歪
	Et	Input	単精度浮動小数点*4B	全歪範囲
	Si	Input	単精度浮動小数点*4B	初期応力
	Qeff	Input	単精度浮動小数点*4B	弾性追従係数
	Df	Input	単精度浮動小数点*4B	D f
	Dc	Input	単精度浮動小数点*4B	D c
	D	Input	単精度浮動小数点*4B	D
	Grid1.Text	Output	TEXTプロパティ	DDE通信用バッファ
機能詳細	<p>評価法：TYPE?の結果データをDDE通信用の変数 (Grid.Textプロパティ)に格納する。  主記憶上の構造体に格納してある「BDS」の損傷結果データをDDE通信用バッファ・エリアForm1/Grid/Textプロパティに代入する。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Type2p			
概要	評価法：TYPE 2による損傷解析の制御を行う			
使用モジュール	Mset、Calcnu、Alpha、Ftres、Smax、Rlx1、Fnumber、Prin1、Prin2 Anf、Sepr			
引数	引数名	I/O	型	意味
	lcount Allmtd. Spx Sn K Qn Ked Neuber Ke Ef Et Si Qeff Df Dc D	Input  Output Output Output Output Output Output Output Output Output Output Output Output Output Output Output	整数型*2B 構造体名 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B	Excelシート・オフセット行数 損傷結果データ格納用 表面応力 等価線形応力 応力集中係数 弾性追従係数 K e' 係数 NEUBER則 歪集中係数 熱ピーク歪 全歪範囲 初期応力 弾性追従係数 D f D c D
機能詳細	<p>評価法：TYPE 2による損傷解析の制御を行う 歪範囲の計算、最高金属温度による<math>\epsilon_n</math>、<math>S_r h</math>の計算、 ファンクションANFの計算機能の限界、 クリープ損傷の計算、DCN/DCPの計算、 クリープ疲労損傷の計算等を行う。</p> <p>中間保持考慮、弾性追従パラメータ最適化</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Excel_Restart			
概要	STAR-NETシステムから呼び出して実行する場合使用する関数。 一時停止状態にあるSTAR-NETシステム(親プログラム)を再起動させる関数。			
使用モジュール	マクロ再開			
引数	引数名	I/O	型	意味
機能詳細	STAR-NETシステムと連結する時に呼ぶ関数、親(EXCEL)関数を再起動させる。 STAR-NETシステムから起動する時は、自動的に実行される。 そして子プログラムである。損傷解析プログラムが実行し終了すると戻る間に親プログラムを再起動させてから処理を終了する。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Rlx1			
概要	初期応力S0(KG/MM**2)、初期クリープ歪 EC1(MM/MM)、クリープパラメータALPHC クリープ破断係数ALPHR、弾性追従パラメータQ 及び時間DT(HR)での緩和応力S2(KG/MM**2) 及びクリープ損傷を計算するサブルーチン			
使用モジュール	Tcrr, Tcr, Ecrr, Nmin			
引数	引数名	I/O	型	意味
	S1 EC1 ALPHC ALPHR Q TMAX Mc	Input Input Input Input Input Input Input	単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 構造体	各種変数要素
機能詳細	初期応力S0(KG/MM**2)、初期クリープ歪 EC1(MM/MM)、クリープパラメータALPHC クリープ破断係数ALPHR、弾性追従パラメータQ 及び時間DT(HR)での緩和応力S2(KG/MM**2) 及びクリープ損傷を計算するサブルーチン			
備考				



言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Sub/サブルーチン形式			
モジュール名	Debug_disp			
概要	開発時に任意のメッセージを表示し、確認のボタンを押すとその番号を返す。			
使用モジュール				
引数	引数名	I/O	型	意味
	Msg	Input	文字型	メッセージ・ボックスに表示する任意のメッセージ
機能詳細	開発時に変数の値を見たい時に、この関数に変数の値と見出しを与えて表示させる。 選択したボタンに応じて番号を返す。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Ftres			
概要	相当応力範囲(TRESCA)を算出する。			
使用モジュール	Smax			
引数	引数名	I/O	型	意味
	Sig1 Sig2 Sig3	Input Input Input	単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B	△応力1 △応力2 △応力3
関数値	関数値名	型	意味	
	Ftres	単精度浮動小数点*4B	相当応力範囲(TRESCA)値	
機能詳細	入力に△応力1・2・3を与える事により、相当応力範囲			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Fvms			
概要	相当応力範囲(VON MISES)を算出する。			
使用モジュール				
引数	引数名	I/O	型	意味
	R	Input	単精度浮動小数点*4B	R座標
	Z	Input	単精度浮動小数点*4B	Z座標
	H	Input	単精度浮動小数点*4B	H座標
	T	Input	単精度浮動小数点*4B	T座標
関数値	関数値名	型	意味	
	Fvms	単精度浮動小数点*4B	相当応力範囲(VON MISES)値	
機能詳細	入力にR・Z・H・T座標を与える事により、相当応力範囲(VON MISES)値を算出する。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Nmin			
概要	最少値を判定する。入力は単精度浮動小数点。			
使用モジュール				
引数	引数名	I/O	型	意味
	A	Input	単精度浮動小数点*4B	数値1
	B	Input	単精度浮動小数点*4B	数値2
	C	Input	単精度浮動小数点*4B	数値3
関数値	関数値名	型	意味	
	Nmin	単精度浮動小数点*4B	最少値	
機能詳細	<p>入力に単精度浮動小数点を与える事により最少値を判定して結果データを返す。 結果データも単精度浮動小数点である。</p>			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Nmax			
概要	最大値を判定する。入力は単精度浮動小数点。			
使用モジュール				
引数	引数名	I/O	型	意味
	A	Input	単精度浮動小数点*4B	数値1
	B	Input	単精度浮動小数点*4B	数値2
	C	Input	単精度浮動小数点*4B	数値3
関数値	関数値名	型	意味	
	Nmax	単精度浮動小数点*4B	最大値	
機能詳細	入りに△応力1・2・3を与える事により、相当応力範囲			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Smax			
概要	最大値を判定する。入力は単精度浮動小数点。			
使用モジュール	Bds、Bdsmd、Bscval、Dds、Type1、Type2			
引数	引数名	I/O	型	意味
	A	Input	単精度浮動小数点*4B	数値1
	B	Input	単精度浮動小数点*4B	数値2
	C	Input	単精度浮動小数点*4B	数値3
関数値	関数値名	型	意味	
	Smax	単精度浮動小数点*4B	最大値	
機能詳細	入りに単精度浮動小数点を与える事により最大値を判定して結果データを返す。 結果データも単精度浮動小数点である。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Eq			
概要	歪範囲算出プログラム			
使用モジュール				
引数	引数名	I/O	型	意味
	R	Input	単精度浮動小数点*4B	R座標
	Z	Input	単精度浮動小数点*4B	Z座標
	H	Input	単精度浮動小数点*4B	H座標
	T	Input	単精度浮動小数点*4B	T座標
関数値	関数値名	型	意味	
	Eq	単精度浮動小数点*4B	歪範囲値	
機能詳細	入力にR・Z・H・T座標を与える事により、歪み範囲を算出する。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Prin1			
概要	主応力成分差を算出する。			
使用モジュール				
引数	引数名	I/O	型	意味
	R	Input	単精度浮動小数点*4B	R座標
	Z	Input	単精度浮動小数点*4B	Z座標
	T	Input	単精度浮動小数点*4B	T座標
関数値	関数値名	型	意味	
	Prin1	単精度浮動小数点*4B	主応力成分差	
機能詳細	入りにR・Z・T座標を与える事により、主応力成分差を算出する。 結果データも単精度浮動小数点である。			
備考				



言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Prin2			
概要	主応力成分差を算出する。			
使用モジュール				
引数	引数名	I/O	型	意味
	R	Input	単精度浮動小数点*4B	R座標
	Z	Input	単精度浮動小数点*4B	Z座標
	T	Input	単精度浮動小数点*4B	T座標
関数値	関数値名	型	意味	
	Prin2	単精度浮動小数点*4B	主応力成分差	
機能詳細	入力にR・Z・T座標を与える事により、主応力成分差を算出する。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Tcrr			
概要	応力S(MM/KG**2) に対する平均クリープ破断時間(HR) 値を算出する。			
使用モジュール				
引数	引数名	I/O	型	意味
	X	Input	単精度浮動小数点*4B	
関数値	関数値名		型	意味
	Tcrr		単精度浮動小数点*4B	
機能詳細	入力にXを与える事により応力S(MM/KG**2) に対する $TCRR = 10^{**}(\dots\dots\dots)$ の ( $\dots\dots\dots$ )部分の値が20.0以上なら 20.0にする。又、-20.0以下なら-20.0にする。 結果データは、単精度浮動小数点である。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Ecrr			
概要	応力S(KG/MM**2) 時間T(HR) 及びクリープパラメータ ALPHCにおけるクリープひずみ速度(MM/MM/HR)を算出する。			
使用モジュール				
引数	引数名	I/O	型	意味
	X	Input	単精度浮動小数点*4B	
関数値	関数値名	型	意味	
	Ecrr	単精度浮動小数点*4B	クリープひずみ速度	
機能詳細	応力S(KG/MM**2) 時間T(HR) 及びクリープパラメータ ALPHCにおけるクリープひずみ速度(MM/MM/HR)を算出する。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Tcr			
概要	応力S(KG/MM**2) クリープ歪ECO(MM/MM) 及び クリープパラメータALPHCに対応する時間(HR)を算出する。			
使用モジュール	Tcrr			
引数	引数名	I/O	型	意味
	S ECO ALPHC	Input Input Input	単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B	
関数値	関数値名	型	意味	
	Tcr	単精度浮動小数点*4B	クリープパラメータALPHCに対応する時間	
機能詳細	応力S(KG/MM**2) クリープ歪ECO(MM/MM) 及び クリープパラメータALPHCに対応する時間(HR)を算出する。 結果データは、単精度浮動小数点である。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Fnumber			
概要	弾完全塑性体を仮定したNEUBER則を算出する。			
使用モジュール				
引数	引数名	I/O	型	意味
	SN	Input	単精度浮動小数点*4B	SN
	K	Input	単精度浮動小数点*4B	K
	BAR3SM	Input	単精度浮動小数点*4B	3SM BAR
関数値	関数値名	型	意味	
	Fnumber	単精度浮動小数点*4B	NEUBER則値	
機能詳細	弾完全塑性体を仮定したNEUBER則を算出する。 結果データは、単精度浮動小数点である。			
備考				

言語	Microsoft Visual Basic2.0 for Windows			
関数種類	Function/ファンクション形式			
モジュール名	Fnubr2			
概要	単調応力-ひずみ関係を仮定したNEUBER則を算出する。			
使用モジュール	Sepm			
引数	引数名	I/O	型	意味
	SN	Input	単精度浮動小数点*4B	SN
	K	Input	単精度浮動小数点*4B	K
	SRH	Input	単精度浮動小数点*4B	3SM BAR
	EH	Input	単精度浮動小数点*4B	HOT時のE
関数値	関数値名	型		意味
	Fnubr2	単精度浮動小数点*4B	NEUBER則値	
機能詳細	弾完全塑性体を仮定したNEUBER則を算出する。			
備考				

## 5 FORTRAN環境

### 5.1 機能概要

損傷計算を行うプログラム本体は、Microsoft Visual Basic 2.0で作成した。各評価法プログラムから頻繁に呼び出す共通サブルーチンは Microsoft FORTRAN 5.0でDLL形式のファイルとして作成した。各鋼材に対応した材料特性値を算出するためのライブラリ群である。

これらのライブラリは、DLL(Dynamic Link Library)形式のファイルにしてある。この形式にする利点は実行中その関数が必要になった時に初めて主記憶上にロードされる事である。そのため主記憶を圧迫せずに有効に利用することが出来る。DLL形式のファイルに関する詳細は「7. 動的リンクライブラリの呼び出し」で説明する。

### 5.2 入出力仕様

共通サブルーチンが行うデータの入出力に関して説明する。

以下のプログラム層とのデータ受け渡しを行う。

(1) 損傷解析プログラム(VisualBasic)とのデータ受け渡し

(1)の損傷解析プログラムから共通サブルーチンは通常関数呼び出しの形式でコールされる。

そのため入出力データは、関数のパラメータとなる。

文字型/整数型/小数点型/配列/構造体等幅広いタイプのデータが受け渡し出来る。コールされる関数によってパラメータの数・タイプは異なる。

損傷プログラム側は、MS-EXCELから受け取ったデータを主記憶上に構造体(配列の親戚)として保持する。EXCEL側のファイル全部を主記憶上に保持する事は出来ないため数分割してデータを受け取り処理する様にしている。

### 5. 3 モジュール説明

材料特性値算出用ライブラリであるFortran言語で記述された個々の関数の説明を行なう。



言語	Microsoft FORTRAN5.0 for MS-DOS				
関数種類	Sub/サブルーチン形式				
モジュール名	Mset				
概要	<p>材料物性値算出用の定数のセットを行う。          事前に鋼種(IM)及び温度(TC)を規定しておく          必要がある。          本サブルーチンのコールにより、対象鋼種の対象温度での          物性値テーブル(COMMON /MC/MC(10,4))を定める。</p>				
使用モジュール	Suval、Smval				
引数	引数名	I/O	型		意味
	IFLG	Input	整数型*2B		弾塑性における 最小特性値 1以外 平均値 各種共通変数
Mc	Input	構造体			
機能詳細	<p>材料物性値算出用の定数のセットを行う。          事前に鋼種(IM)及び温度(TC)を規定しておく          必要がある。          本サブルーチンのコールにより、対象鋼種の対象温度での          物性値テーブル(COMMON /MC/MC(10,4))を定める。</p>				
備考					

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Sub/サブルーチン形式			
モジュール名	Mclist			
概要	材料特性を定める定数の内容を表示するサブルーチン			
使用モジュール				
引数	引数名	I/O	型	意味
	IFLG	Input	整数型*2B	定数表示指定スイッチ
機能詳細	材料特性を定める定数の内容を表示するサブルーチン IFLG=0 全定数表示 IFLG=1 MC(*,1)の内容表示 IFLG=2 MC(*,2)の内容表示 IFLG=3 MC(*,3)の内容表示 IFLG=4 MC(*,4)の内容表示			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Sub/サブルーチン形式			
モジュール名	R1x1			
概要	材料物性値算出用の定数のセットを行う。 EC1(MM/MM)、クリープパラメータALPHC クリープ破断係数ALPHR、弾性追従パラメータQ 及び時間DT(HR)での緩和応力S2(KG/MM**2) 及びクリープ損傷を計算するサブルーチン			
使用モジュール	Tcrr、Tcr、Ecrr、Nmin			
引数	引数名	I/O	型	意味
	S1	Input	単精度浮動小数点*4B	初期応力
	EC1	Input	単精度浮動小数点*4B	初期クリープ歪
	ALPHC	Input	単精度浮動小数点*4B	クリープパラメータ
	ALPHR	Input	単精度浮動小数点*4B	クリープ破断係数
	Q	Input	単精度浮動小数点*4B	弾性追従パラメータQ
	TMAX	Input	単精度浮動小数点*4B	時間DT(HR)
	S2	Output	単精度浮動小数点*4B	時間DT(HR)での緩和応力
	DC	Output	単精度浮動小数点*4B	クリープ損傷
	Mc	Input	構造体	各種共通変数
機能詳細	材料物性値算出用の定数のセットを行う。 EC1(MM/MM)、クリープパラメータALPHC クリープ破断係数ALPHR、弾性追従パラメータQ 及び時間DT(HR)での緩和応力S2(KG/MM**2) 及びクリープ損傷を計算するサブルーチン			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Sub/サブルーチン形式			
モデル名	R1x2			
概要	初期応力S0(KG/MM**2)、 初期クリープ歪EC1(MM/MM)、 クリープパラメータALPHC、弾性追従パラメータQ及び 緩和応力S2(KG/MM**2)に対応する時間DT(HR)及び クリープ損傷DCを計算するサブルーチン			
使用モデル				
引数	引数名	I/O	型	意味
	S1 EC1 ALPHC ALPHR Q TMAX SMIN DC Mc	Input Input Input Input Input Input Output Output Input	単精度浮動 単精度浮動 単精度浮動 単精度浮動 単精度浮動 単精度浮動 単精度浮動 単精度浮動 構造体	初期応力 初期クリープ歪 クリープパラメータ クリープ破断係数 弾性追従パラメータQ 時間DT(HR) 時間DT(HR)での緩和応力 クリープ損傷 各種共通変数
機能詳細	初期応力S0(KG/MM**2)、 初期クリープ歪EC1(MM/MM)、 クリープパラメータALPHC、弾性追従パラメータQ及び 緩和応力S2(KG/MM**2)に対応する時間DT(HR)及び クリープ損傷DCを計算するサブルーチン			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Sub/サブルーチン形式			
モジュール名	Crlist			
概要	応力S(KG/MM**2) に対するクリープ定数 C1, R1, C2, R2, EMを表示するサブルーチン			
使用モジュール				
引数	引数名	I/O	型	意味
	S	Input	単精度浮動小数点*4B	応力S(KG/MM**2)
機能詳細	応力S(KG/MM**2) に対するクリープ定数 C1, R1, C2, R2, EMを表示するサブルーチン			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Suval			
概要	高温構造設計方針材料強度基準等に規定される設計引張強さ			
使用モジュール				
引数	引数名	I/O	型	意味
	IDUMMY Mc	Input Input	整数型*2B 構造体	ダミー引数 各種共通変数
関数値	関数値名	型	意味	
	Suval	単精度浮動小数点*4B	材料強度基準等に規定される 設計引張強さ	
機能詳細	高温構造設計方針材料強度基準等に規定される設計引張強さ IDUMMY 将来の使用を考慮したダミー引数。現状未使用			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Eepm			
概要	応力X(KG/MM**2)に対する弾塑性歪(MM/MM) (単調負荷)			
使用モジュール				
引数	引数名	I/O	型	意味
	X Mc	Input Input	単精度浮動小数点*4B 構造体	応力(KG/MM**2) 各種共通変数
関数値	関数値名	型		意味
	Eepm	単精度浮動小数点*4B	弾塑性歪(MM/MM) (単調負荷)	
機能詳細	応力X(KG/MM**2)に対する弾塑性歪(MM/MM) (単調負荷)			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Sepr			
概要	弾塑性歪X(MM/MM)に対する応力(KG/MM**2) (単調負荷)			
使用モジュール				
引数	引数名	I/O	型	意味
	X Mc	Input Input	単精度浮動小数点*4B 構造体	弾塑性歪(MM/MM) 各種共通変数
関数値	関数値名	型	意味	
	Sepr	単精度浮動小数点*4B	弾塑性歪X(MM/MM) に対する応力(KG/MM**2)	
機能詳細	弾塑性歪X(MM/MM)に対する応力(KG/MM**2) (単調負荷) 応力範囲(KG/MM**2) (繰返し負荷)			
備考				



言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Tcr			
概要	応力S(KG/MM**2) クリープ歪ECO(MM/MM) 及び クリープパラメータALPHC に対応する時間(HR)			
使用モジュール				
引数	引数名	I/O	型	意味
	S	Input	単精度浮動小数点*4B	応力(KG/MM**2)
	ECO	Input	単精度浮動小数点*4B	クリープ歪(MM/MM)
	ALPHC	Input	単精度浮動小数点*4B	クリープパラメータ
	Mc	Input	構造体	各種共通変数
関数値	関数値名	型	意味	
	Tcr	単精度浮動小数点*4B	ALPHC に対応する時間(HR)	
機能詳細	応力S(KG/MM**2) クリープ歪ECO(MM/MM) 及び クリープパラメータALPHC に対応する時間(HR)			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Tcrr			
概要	応力S(MM/KG**2) に対する平均クリープ破断時間(HR) $Tcrr = 10^{**}(\dots\dots\dots)$ の 20.0にする。 又、-20.0以下なら-20.0にする。			
使用モジュール				
引数	引数名	I/O	型	意味
	X Mc	Input Input	単精度浮動小数点*4B 構造体	応力(MM/KG**2) 各種共通変数
関数値	関数値名	型	意味	
	Tcrr	単精度浮動小数点*4B	応力S(MM/KG**2) に対する 平均クリープ破断時間(HR)	
機能詳細	応力S(MM/KG**2) に対する平均クリープ破断時間(HR) $Tcrr = 10^{**}(\dots\dots\dots)$ の 20.0にする。 又、-20.0以下なら-20.0にする。			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Ecr			
概要	応力S(KG/MM**2) 時間T(HR) 及び クリープパラメータALPHCにおける クリープひずみ(MM/MM)を算出する。			
使用モジュール	Tcrr			
引数	引数名	I/O	型	意味
	S T ALPHC Mc	Input Input Input Input	単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 構造体	応力(KG/MM**2) 時間T(HR) クリープパラメータ 各種共通変数
関数値	関数値名	型		意味
	Ecr	単精度浮動小数点*4B	ALPHCにおける クリープひずみ(MM/MM) を算出する。	
機能詳細	応力S(KG/MM**2) 時間T(HR) 及び クリープパラメータALPHCにおける クリープひずみ(MM/MM)を算出する。			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Eerr			
概要	応力S(KG/MM**2) 時間T(HR) 及び クリープパラメータALPHCにおける クリープひずみ速度(MM/MM/HR)を算出する。			
使用モジュール	Tcrr			
引数	引数名	I/O	型	意味
	S T ALPHC Mc	Input Input Input Input	単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 構造体	応力(KG/MM**2) 時間T(HR) クリープパラメータ 各種共通変数
関数値	関数値名	型		意味
	Suval	単精度浮動小数点*4B	材料強度基準等に規定される 設計引張強さ	
機能詳細	応力S(KG/MM**2) 時間T(HR) 及び クリープパラメータALPHCにおける クリープひずみ速度(MM/MM/HR)を算出する。			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Scrr			
概要	平均クリープ破断時間T(HR) に対する応力(KG/MM**2) を算出する。			
使用モジュール				
引数	引数名	I/O	型	意味
	T	Input	単精度浮動小数点*4B	平均クリープ 破断時間T(HR)
	Mc	Input	構造体	各種共通変数
関数値	関数値名	型		意味
	Scrr	単精度浮動小数点*4B	クリープ破断時間T(HR) に対する応力(KG/MM**2)	
機能詳細	平均クリープ破断時間T(HR) に対する応力(KG/MM**2) を算出する。			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Anf			
概要	全ひずみ範囲ET(MM/MM)及びひずみ速度 EDOT(MM/MM/SEC)に対する平均疲労寿命 Anf = 10**((.....)**(-2))の ((.....)**(-2)部分の値が35.0以上なら 35.0にする。又、-35.0以下なら-35.0にする。			
使用モジュール				
引数	引数名	I/O	型	意味
	ET	Input	単精度浮動小数点*4B	全ひずみ範囲(MM/MM)
	EDOT	Input	単精度浮動小数点*4B	ひずみ速度(MM/MM/SEC)
	Mc	Input	構造体	各種共通変数
関数値	関数値名	型	意味	
	Anf	単精度浮動小数点*4B	EDOT(MM/MM/SEC) に対する平均疲労寿命	
機能詳細	全ひずみ範囲ET(MM/MM)及びひずみ速度 EDOT(MM/MM/SEC)に対する平均疲労寿命  Anf = 10**((.....)**(-2))の ((.....)**(-2)部分の値が35.0以上なら 35.0にする。又、-35.0以下なら-35.0にする。			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Sids			
概要	応力水準SG(KG/MM**2)におけるDstar 算出用初期応力(KG/MM**2)を算出する。			
使用モジュール	Eepm			
引数	引数名	I/O	型	意味
	SG Mc	Input Input	単精度浮動小数点*4B 構造体	応力水準(KG/MM**2) 各種共通変数
関数値	関数値名	型	意味	
	Sids	単精度浮動小数点*4B	Dstar算出用初期応力	
機能詳細	応力水準SG(KG/MM**2)におけるDstar 算出用初期応力(KG/MM**2)を算出する。			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Dstar			
概要	(1次+2次)応力範囲SN(KG/MM**2)、 高温使用時間TSTAR(HR)及び応力水準SGによる クリープ損傷DC1に対する(1次+2次)応力の緩和に 伴うクリープ損傷係数Dstar値 ただし、温度による 補正が無いので注意を要する。			
使用モジュール	Scrr, AMIN1, AMAX1			
引数	引数名	I/O	型	意味
	SN	Input	単精度浮動小数点*4B	(1次+2次)応力範囲 (KG/MM**2)
	TSTAR	Input	単精度浮動小数点*4B	高温使用時間(HR)
	DC1	Input	単精度浮動小数点*4B	クリープ損傷
	Mc	Input	構造体	各種共通変数
関数値	関数値名	型	意味	
	Dstar	単精度浮動小数点*4B	応力の緩和に 伴うクリープ損傷係数Dstar値	
機能詳細	(1次+2次)応力範囲SN(KG/MM**2)、 高温使用時間TSTAR(HR)及び応力水準SGによる  クリープ損傷DC1に対する(1次+2次)応力の緩和に 伴うクリープ損傷係数Dstar値 ただし、温度による 補正が無いので注意を要する。			
備考				



言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Ddstar			
概要	等価歪範囲ET(MM/MM)、高温使用時間TSTAR(HR)及び 応力水準SGによるクリープ損傷DC1 に対する ピーク応力の緩和に伴うクリープ損傷係数DDOUBLE-STAR値 ただし、温度による補正が無いので注意を要する。			
使用モジュール	Scrr, AMAX1, Sepr, Mset, Sepm			
引数	引数名	I/O	型	意味
	ET TSTAR DC1 Mc	Input Input Input Input	単精度浮動小数点*4B 単精度浮動小数点*4B 単精度浮動小数点*4B 構造体	等価歪範囲(MM/MM) 高温使用時間(HR) クリープ損傷 各種共通変数
関数値	関数値名	型	意味	
	Ddstar	単精度浮動小数点*4B	クリープ損傷DC1 に対する ピーク応力の緩和に 伴うクリープ損傷係数	
機能詳細	等価歪範囲ET(MM/MM)、高温使用時間TSTAR(HR)及び 応力水準SGによるクリープ損傷DC1 に対する ピーク応力の緩和に伴うクリープ損傷係数DDOUBLE-STAR値 ただし、温度による補正が無いので注意を要する。			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Alpha			
概要	線膨張係数算出プログラム			
使用モジュール	Xy			
引数	引数名	I/O	型	意味
	Mc	Input	構造体	各種共通変数
関数値	関数値名	型	意味	
	Alpha	単精度浮動小数点*4B	線膨張係数	
機能詳細	線膨張係数算出プログラム			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Xy			
概要	直線内挿入を行う			
使用モジュール				
引数	引数名	I/O	型	意味
	X2	Input	単精度浮動小数点*4B	X2座標
	X1	Input	単精度浮動小数点*4B	X1座標
	Y2	Input	単精度浮動小数点*4B	Y2座標
	Y1	Input	単精度浮動小数点*4B	Y1座標
	XX	Input	単精度浮動小数点*4B	XX
関数値	関数値名	型		意味
	Xy	単精度浮動小数点*4B	直線内挿入	
機能詳細	直線内挿入を行う			
備考				

言語	Microsoft FORTRAN5.0 for MS-DOS			
関数種類	Function/ファンクション形式			
モジュール名	Calcnu			
概要	ポアソン比を換算するサブルーチン			
使用モジュール				
引数	引数名	I/O	型	意味
	Mc	Input	構造体	各種共通変数
関数値	関数値名	型		意味
	Calcnu	単精度浮動小数点*4B	ポアソン比を換算する	
機能詳細	ポアソン比を換算するサブルーチン			
備考				

## 6 プロセス間通信 (DDE)

### 6. 1 DDEとは

ダイナミック・データ・エクスチェンジ(Dynamic Data Exchange)の略語。

DDEとは、2つのアプリケーションが連続的に対話を行いデータを交換する機能である。アプリケーション間のデータの切り取りと貼り付けを自動化し、データの更新を高速に実行する。

この機能を使う前提条件として、対象となるアプリケーションがDDEをサポートしていない場合には利用出来ない。アプリケーションがDDEをサポートしているかどうかは、各アプリケーションのマニュアルを参照することである。

Visual Basicで作成したアプリケーションも含め、Windowsではマルチタスクにより複数のアプリケーションを並行して実行でき、クリップボードを利用したデータのコピーと貼り付けによってアプリケーション間で簡単にデータを交換できる。DDEを使うと、このようなデータの交換を自動的に実行出来る。DDEでは、他のアプリケーションからのデータの取得、新しいデータへの更新、コマンドやキー入力の送出によるアプリケーションのリモートコントロールなどの操作が可能である。

ここでは、DDEの基本的な仕組み・Visual BasicとMS-EXCELとの間で行われるDDE通信の具体的なやり取りを示す。

### 6. 2 基本的仕組み

2つのアプリケーションは、DDE通信によってデータを交換できる。DDE通信を開始する側のアプリケーションを「デスティネーション・アプリケーション」または「デスティネーション」と呼び、デスティネーションに答える側のアプリケーションを「ソース・アプリケーション」または「ソース」と呼ぶ。

デスティネーションやソースになるための特別な条件は不要であるが、1つのアプリケーションがソースにもデスティネーションにもなることが出来る。また、同時に複数のDDE通信を行なうことも出来る。

デスティネーション・アプリケーションでは、次の2つを指定してDDE通信を開始する。

- ① ソースとなるアプリケーションの名前
- ② 通信の主体。これをトピックという。

ソースアプリケーションは認識出来るトピックに対するDDE通信の要求を受けると、要求を出したデスティネーションかソースがアプリケーション名やトピックを変更すると、DDE通信は終了する。

DDE通信実行中は、デスティネーションとソースの間で1つまたは複数のアイテムに関する情報を交換出来る。アイテムとは、2つのアプリケーション間で

交換されるデータを指します。

デスティネーションとソースのどちらも、DDE通信の状態に影響を与えることなくアイテムを変更出来る。

アプリケーション間で交換されるデータは、アプリケーション名、トピック、アイテムの組み合わせを指定することによって決まる。

## 6. 2. 1 アプリケーション名

DDE通信のソースとして使うことが出来るアプリケーションには、それぞれ他のアプリケーションと重複しない名前が与えられている。このアプリケーション名は、通常実行可能ファイル名から拡張子を除いたものである。次の表は、代表的なMicrosoftアプリケーション名を示す。

アプリケーション	DDEアプリケーション名
Microsoft Excel	Excel
Microsoft Word for Windows	WinWord
Microsoft Project	Project
Microsoft Access	MSAccess
Microsoft FoxPro for Windows	FoxPro
Microsoft Windows プログラム・マネージャ	ProgMan

表4. 1 DDEアプリケーション名

Visual BasicアプリケーションのフォームがDDE通信のソースになる場合、アプリケーション名は、アプリケーション実行可能ファイルを作成するときに指定した名前になる。なお、アプリケーションをVisual Basicの中で実行している場合のアプリケーション名は、プロジェクト・ファイル名から拡張子(.MAK)を除いたものになる。

DDEのアプリケーション名では、大文字と小文字を区別する必要はない。

「excel」、「Excel」、「EXCEL」は全て同じである。

## 6. 2. 2 トピック

トピックとは、DDE通信の主体を定義したもので、通常、ソース・アプリケーションで処理されるデータの単位を指す。ほとんどのアプリケーションは、そのアプリケーションで扱う文書名をDDE通信のトピックとして認識できる。

例えば、Microsoft Excelでは拡張子が、XLSまたは、XLCのファイル名（必要ならばパスを含む）を認識する。また、word for Windowsでは拡張子が、DOCまたは、DOTのファイル名を認識する。アプリケーション名の場合と同様に、DDE通信のトピックにも大文字と小文字の区別はない。

Visual Basicアプリケーション中のフォームをDDE通信のソース

にするためには、フォームのLinkTopicプロパティでトピックを指定します。

DDE機能を利用できるアプリケーションの多くは、Systemというトピックをサポートしている。このトピックは、そのアプリケーションが他にどのようなトピックをサポートしているか、などのアプリケーションの情報を要求するために使用する。アプリケーションがsystemトピックをサポートしているかどうかは、そのアプリケーションのマニュアルを参照してすること。

Visual BasicアプリケーションでSystemトピックをサポートするには、まずLinkTopicプロパティの値がSystemであるフォームを作成し、そのフォームにSystemの各アイテムに対応する名前のコントロールを作成する。

### 6. 2. 3 アイテム

アイテムは、DDE通信で実際にやり取りされるデータである。例えば、Excelでは、R1C1のようなセルをDDE通信のアイテムとして認識する。

また、word for Windowsでは、LinkTopicプロパティで指定した文書の中にあるブックマークをDDE通信のアイテムとして認識する。

さらに特殊なブックマーク $\$Doc$ を任意の文書で有効なアイテムとして認識できる。アプリケーション名やトピックの場合と同様に、DDE通信のアイテムにも大文字と小文字を区別する必要はない。

Visual Basicアプリケーション中のコントロールがDDE通信のデスティネーションであれば、LinkItemプロパティの値を設定してDDE通信のアイテムを指定できる。

Visual Basicアプリケーション中のフォームがDDE通信のソースであれば、フォーム上のテキストがボックス、ラベル、ピクチャーボックスの名前がDDE通信のアイテムとして指定できる。

### 6. 2. 4 通信でのリンクの種類

DDE通信は、通信を行う2つのアプリケーションが交換するデータによって結び付けられることから「リンク」と呼ばれることがある。リンクには、データの交換方法の違いによって次の3種類がある。

#### ■ 自動リンク

LinkItemプロパティで定義されるデータが変更されるたびに、ソースからデスティネーションへデータを送る。

#### ■ 手動リンク

デスティネーションが要求したときだけ、ソースからデータを送る。

#### ■ 通知リンク

データが変更されると、そのことをソースからデスティネーションに通知する。

ただし、実際にソースからデータを送るのはデスティネーションが要求したときだけである。

Microsoft Excel	Microsoft Word	VisualBasic
INITIATE	DDEInitiate	LinkMode=自動 LinkMode=手動 LinkMode=通知
REQUEST	DDERequest	LinkRequestメソッド
POKE	DDEPoke	LinkPokeメソッド
EXECUTE	DDEExecute	LinkExecuteメソッド
TERMINATE	DDETerminate	LinkModeメソッド

表 4. 2 Microsoft Excel、Word、VisualBasicのDDEステートメント

図 3. 1 にリンクの種類を示す。

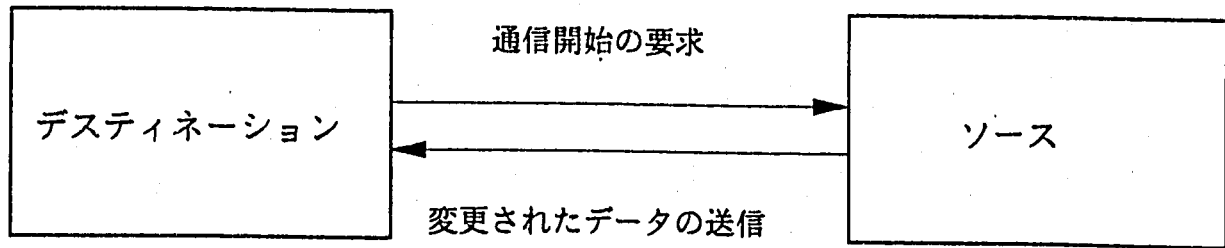
### 6. 3 具体的操作

本プログラムでVisualBasicとMS-EXCELとの間で行なった実際の操作をコードを示して説明する。

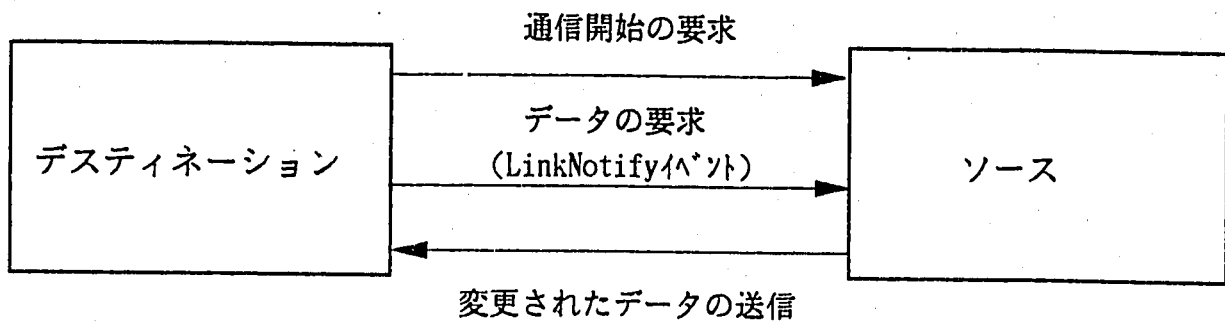
図 3. 2 にDDE通信によるセルの対応付けを示す。  
更に表 6. 1 に「サンプル・コード」をその説明を示す。



### 自動リンク



### 手動リンク



### 通知リンク

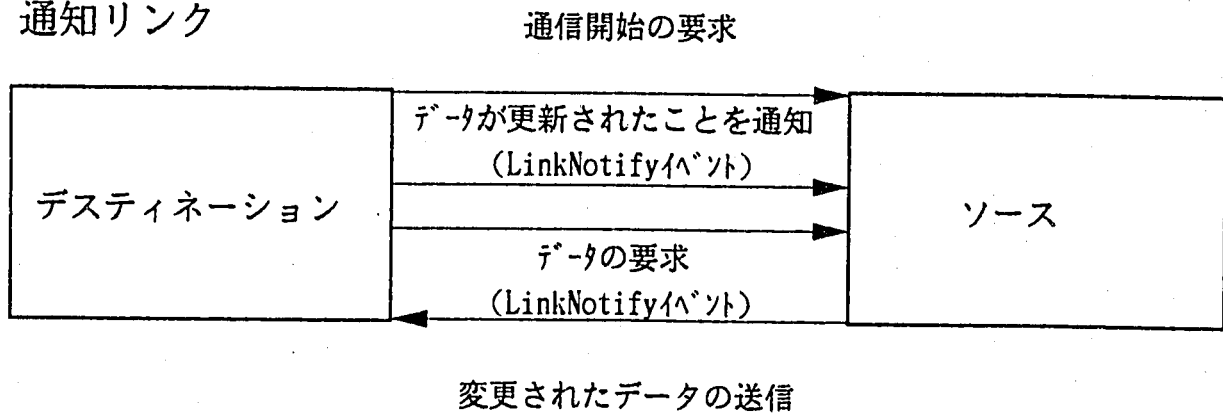
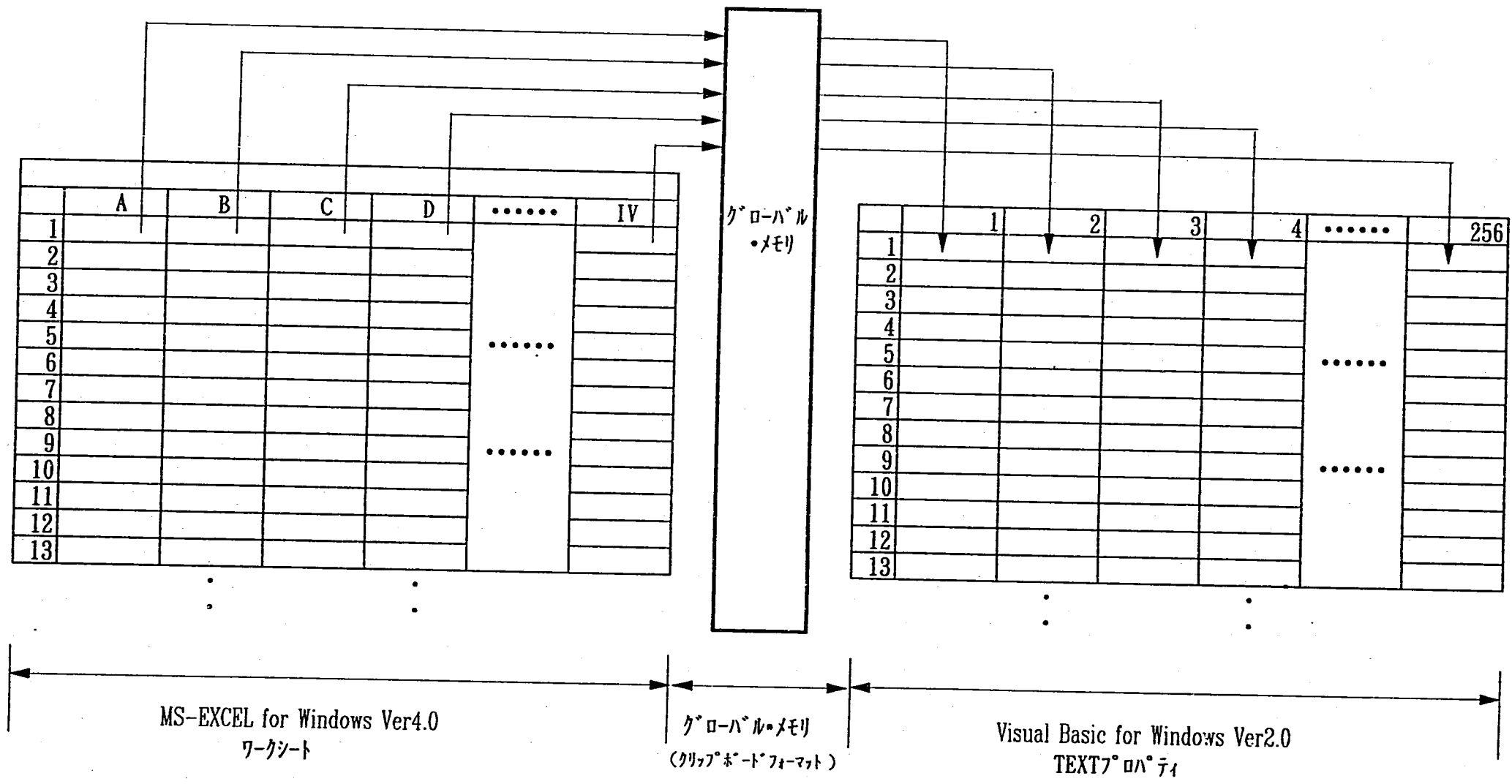


図3. 1 DDE通信リンクの種類



※ 最少単位はセルであり、EXCELとVisual Basicの個々のセルが対応付けられる。  
 その他のDDE対応アプリケーションについても同様のことが行われる。

図3. 2 DDE通信によるセルの対応付け

## 表6. 1 DDE通信サンプル・コード

- ① Form1!Text1.LinkMode = NONE
- ② Form1!Text1.LinkTopic = "Excel|試験条件.CND"
- ③ Form1!Text1.LinkItem = "R5C1:R5C20"
- ④ Form1!Text1.LinkMode = LINK\_AUTOMATIC
- ⑤ Form1!Grid1.SelStartRow = 1
- ⑥ Form1!Grid1.SelStartCol = 1
- ⑦ Form1!Grid1.SelEndRow = 1
- ⑧ Form1!Grid1.SelEndCol = 20
- ⑨ Form1!Grid1.Clip = Form1.Text1.Text
- ⑩ Form1!Grid1.Row = 1
- ⑪ Form1!Grid1.Col = 4
- ⑫ Cond.Material = Form1!Grid1.Text

Call Mtrl

'変換:材料名称->材料番号

Form1!Grid1.Col = 6

Cond.Tempc = Val(Form1!Grid1.Text)

Form1!Grid1.Col = 7

Cond.Temph = Val(Form1!Grid1.Text)

.  
.  
.

Form1!Grid1.Col = 20

Cond.OutSheetName = Form1!Grid1.Text

- ⑬ Form1!Text1.LinkMode = NONE

[説明]

- ① DDE通信でのリンクの種類。リンクには、データの交換方法の違いによって次の3種類がある。

- ◆ 自動リンク : LINK\_AUTOMATIC
- ◆ 手動リンク : LINK\_MANUAL
- ◆ 通知リンク : LINK\_NOTIFY
- ◆ リンクなし : NONE

ここでは、DDE通信の初期化処理として「リンクなし」を指定している。  
詳細は、「6. 2. 4 通信でのリンクの種類」を参照の事。

② トピックとは、DDE通信の主体を定義したもので、通常、ソース・アプリケーションで処理されるデータの単位を指す。

ここで示すようにパラメータとしては、「DDEアプリケーション名/ファイル名」というかたちで指定する。

③ アイテムは、DDE通信で実際にやり取りされるデータである。例えば、Excelでは、R1C1のようなセルをDDE通信のアイテムとして認識する。

これは、MS-EXCELのワークシートのセルの範囲(開始セル位置から終了セル位置)を指定する。

④ DDE通信でのリンクの種類。リンクには、データの交換方法の違いによって次の3種類がある。

ここでは、「自動リンク」を行なっている。

- ◆ 自動リンク : LINK\_AUTOMATIC

この時点でMS-EXCEL(ワークシート)からVisual Basic(TEXTプロパティ)にデータが受け渡された訳である。(全データ群)

次にVisual Basic内で個々のデータを処理するためには、「GRIDプロパティ(配列形式のメモ領域)」にデータをコピーしなければならない。

⑤ GRIDプロパティのどこから、どこまでの配列領域に格納するか指定しなければならない。

これは、GRIDプロパティ開始行番号である。

⑥ GRIDプロパティのどこから、どこまでの配列領域に格納するか指定しなければならない。

これは、GRIDプロパティ開始桁番号である。

⑦ GRIDプロパティのどこから、どこまでの配列領域に格納するか指定しなければならない。

これは、GRIDプロパティ最終行番号である。

⑧ GRIDプロパティのどこから、どこまでの配列領域に格納するか指定しなければならぬ。

これは、GRIDプロパティ最終桁番号である。

⑨ TEXTプロパティからGRIDプロパティに全データをコピーする。

TEXTプロパティの全データを、指定されたGRIDプロパティの領域にコピーした訳である。

データを要素として扱えるようにしたが、膨大なデータを系統立てた配列あるいは変数に格納して使用したほうが効率が良い。そのためGRIDプロパティの各データを配列／変数に再格納する。

⑩ GRIDプロパティの各データの格納位置を指定する。

これは、GRIDプロパティの目的とする行番号である。

⑪ GRIDプロパティの各データの格納位置を指定する。

これは、GRIDプロパティの目的とする桁番号である。

⑫ GRIDプロパティから各配列あるいは変数にデータを再格納する。

⑬ 最後にDDE通信のリンク・モードをクリアする。

## 7 動的リンクライブラリ (DLL) の呼び出し

### 7. 1 動的リンクライブラリ (DLL) とは

ダイナミック・リンク・ライブラリ (Dynamic Link Library) の略語。

DLLとは、コンパイル時でなくロード時または実行時にプログラムにリンクされるルーチンである。MS WINDOWSはダイナミック・リンク・ライブラリをサ

ポートしているが、MS-DOSではサポートしていない。ダイナミック・リンクを理解するために、より一般的なリンク方法であるスタティック・リンクと比較してみる。

スタティックリンクは、次の手順で行う。

1. WRITEなどのFORTRANの標準入出力文を使って、プログラムを作成、コンパイルする。これによって、FORTRANの標準ライブラリ中のこの高レベル文をサポートする、外部関数を参照するオブジェクトファイルが生成される。
2. オブジェクトファイルを、外部関数を含むライブラリファイルとリンクします。このとき、参照された関数のコードを実行可能ファイル内に組み込むことによって、2つのファイルが結合される。言い換えると、外部参照はリンク時に解決される。
3. ステップ2で生成した実行可能ファイルを実行する。この実行可能ファイルは外部関数のコードを含むので、完全に独立していると言える。  
スタティックリンクは便利であるが、短所もいくつかある。たとえば、実行可能コードを改良または変更するには、メインプログラムのオブジェクトファイルと再リンクする必要がある。したがって、プログラミング言語製品の新しいリリースが出ると、実行可能ファイル全体を新しいバージョンと交換しなければならない。

またスタティックリンクでは、共通なコードを共有できない。WRITE文を使う実行可能ファイルは、すべて必要なルーチンと明確にリンクする必要がある。

したがって、WRITEをサポートするのに必要なコードが、ディスク上またはメモリ上にある異なる実行可能ファイルで重視する可能性がある。

DLLを使う実行可能ファイルは外部関数のコードを含まないので、スタティックにリンクされたファイルよりもサイズが小さくなる。プログラムのバージョンアップを行うときは、DLLの新しいバージョンを提供するだけでよく、実行可能ファイルを再リンクする必要はない。

DLLを使う場合の重要な利点は、実行可能ファイル間でコードを共有できることである。共通に使われるルーチンをDLLに置き、何個の実行可能ファイルからでもアクセスできる。

注意：MSWINDOWSは、DLLを広く用いている。MSWINDOWSのシステム関

数はすべて、ユーザープログラムが呼び出すことのできる外部プロシージャとし

て提供される。

ダイナミックリンクを行うときは、次の手順に従うこと。

1. スタティックリンクの場合と同じように、外部関数を参照するプログラムを作成、コンパイルする。
2. メインプログラムがどの関数をDLLからインポートするかを指定する特別なDEF（定義）ファイルを作成する。
3. メインプログラムを適切な”インポートライブラリ”とリンクする。インポートライブラリは、DLL中の外部関数に対する参照を含むファイルである。このファイルは、外部関数のコードを含まない。
5. メインプログラムで参照されるコードを含むDLLを作成する。
6. ステップ3で作成したプログラムを実行する。この実行可能ファイルを実行するとき、MS Windowsはコードをロードし、DLLの特別な参照を見つける。

それぞれの外部DLL参照については、MS Windowsはロード時にDLLディレクトリを検索して外部参照を解決する。

DLL(ダイナミック・リンク・ライブラリ)は、標準ライブラリと同様に共通に使われる関数のための実行可能コードを含む。しかしダイナミック関数は、実行可能ファイルとはリンクされない。そのかわりに実行可能ファイルと共にDLL自身が、実行時にメモリにロードされる。DLLに関する情報は、モジュール定義ファイルかインポートライブラリに記述してリンクに指示する。

それぞれのDLLは、「エクスポート定義」を使ってその関数を他のモジュールで直接使えるようにする。実行時にエクスポートされない関数は、DLL内でのみ呼び出すことができる。エクスポート定義は、DLLを作成するときにモジュール定義ファイル内でリンクに提供される。

一方実行可能ファイルは、それぞれのダイナミックリンク関数が見つかる場所を指示する「インポート定義」を使う必要がある。これを使わないと、MS OS/2またはWindowsは、プログラムの起動時にどのDLLをロードすればよいのかわからない。それぞれのインポート定義は、関数名と関数が存在するDLLを指定する。

インポート定義は、モジュール定義ファイルまたはインポートライブラリでリンクに提供される。IMPLIBは、DLLまたはDLLを作成する時に使ったモジュール定義ファイルからインポートライブラリを作成する。

1つのアプリケーションと1つのDLLを作成するもっとも単純な場合について考えてみる。リンクはダイナミックリンク関数呼び出しのためのエクスポート定義とインポート定義を必要とする。Windowsでは、これらを定義するために2つの方法がある。

■ DLLを作成するためのエクスポート定義を含むモジュール定義ファイル(拡張子: DEF)と、実行可能ファイルを作成するためのインポート定義を含む別のモジュール定義ファイルを作成する。モジュール定義ファイルには、これらの定義をテキスト形式で記述する。

- DLLのためのモジュール定義ファイル(拡張子. DEF)を作成し、DLLまたはDLLを作成するときに使う。DEFファイルからインポートライブラリを生成する。インポートライブラリは、実行可能ファイルとリンクする。

## 7. 2 インポートライブラリを使わないリンク

図3. 3は、最初の方法を使ってダイナミックリンク関数呼び出しに対する定義を提供する方法を示す。DLLとアプリケーションは、それぞれに対応するモジュール定義ファイルを持っている。(モジュール定義ファイルには通常、拡張子. DEFを付ける)

### 主な手順

1. オブジェクトファイル(もしあれば標準ライブラリ)は、DLLを作成するためにモジュール定義ファイルとリンクする。DLLのモジュール定義ファイルは、少なくとも2つの文を含む。最初はLIBRARY文で、実行可能ファイルではなく、DLLを作成するリンクに指示する。もう1つの文は、エクスポート定義リストである。
2. オブジェクト・ファイル(もしあれば標準ライブラリ)は、アプリケーションを作成するために、モジュール定義ファイルをリンクする。このアプリケーションのためのモジュール定義ファイルは、インポート定義のリストを含む。このリスト中のそれぞれの定義は、関数名とDLL名から成り立つ。

## 7. 3 ダイナミックリンクの利点

DLLは標準ライブラリと同じような目的で使われるが、次のような利点がある。

- ディスクスペースを大幅に節約できる。  
printitという、多くの異なるプログラムから呼び出されるライブラリ関数を作成すると仮定する。printitが標準ライブラリにあるときは、関数の実行可能コードはその関数を呼び出す実行可能ファイル中にリンクする必要がある。言い換えると、同じコードがディスク上の多くの異なるファイル上に存在することになる。しかし、printitがDLLに格納されているときは、実行可能ファイルは1つのファイル(DLL自身)にみに存在する。
- コードとデータセグメントを共有できる。  
DLLからロードされるコードとデータセグメントは共有できる。ダイナミックリンクを行わない時には、それぞれのファイルは、そのファイルが使うすべてのコードとデータのコピーを含むために、セグメントを共有でき



ない。したがって、ダイナミックリンクを使ってセグメントを共有すると、メモリをより効率的に使うことができる。

■ ライブラリやアプリケーションの更新が簡単にできる。

DLLは、それを使うアプリケーションを再リンクすることなく、何回でも更新できる。このことは、特にサードパーティーのライブラリを使う時に便利である。 サードパーティーの開発者から更新されたDLLを入手して、新しいライブラリをディスクにコピーするだけですむ。実行時にアプリケーションは、更新されたライブラリ関数を自動的に呼び出すことができる。

■ アプリケーションのリンク時間を短縮できる。

ダイナミックリンクを使うときは、ダイナミックリンク関数の実行可能コードは実行可能ファイルにコピーされず、インポート定義のみがコピーされる。

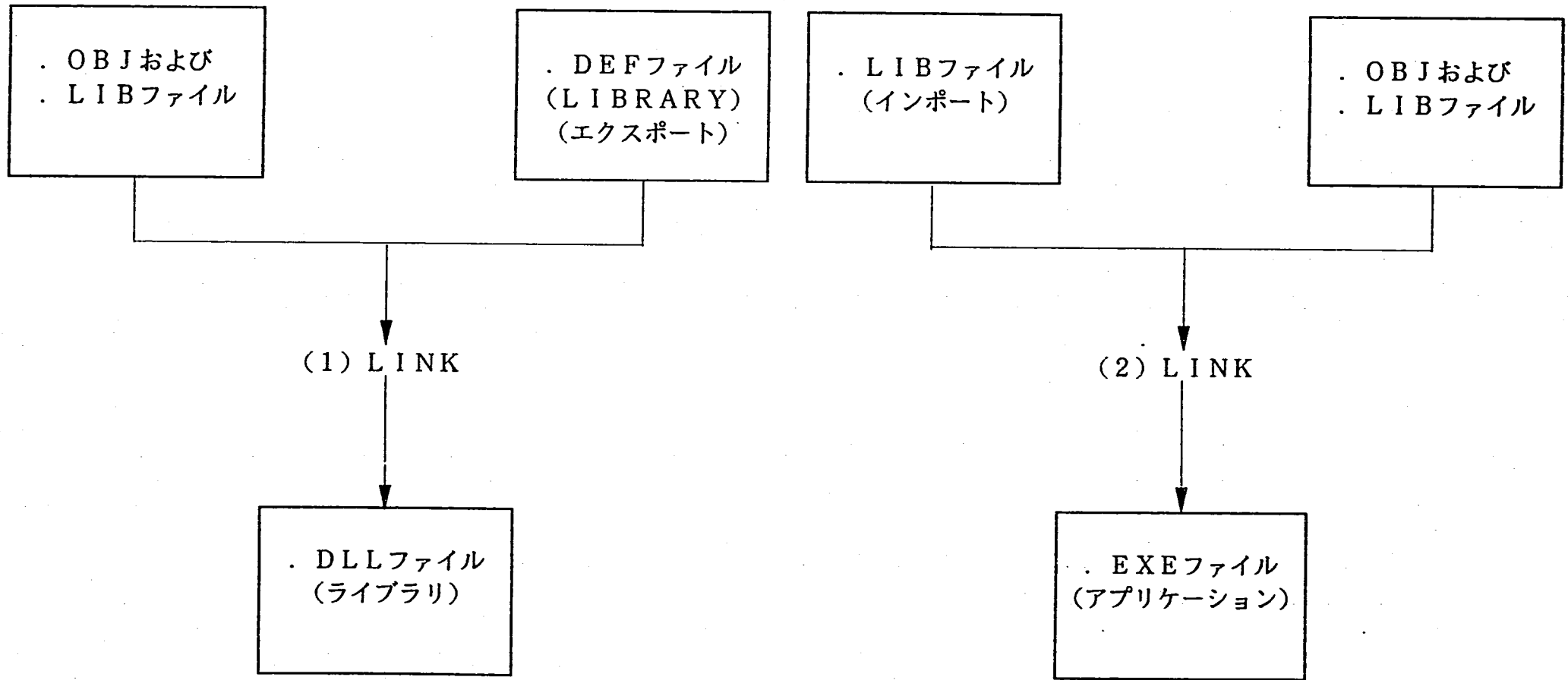


図3. 3 インポート・ライブラリを使わないリンク

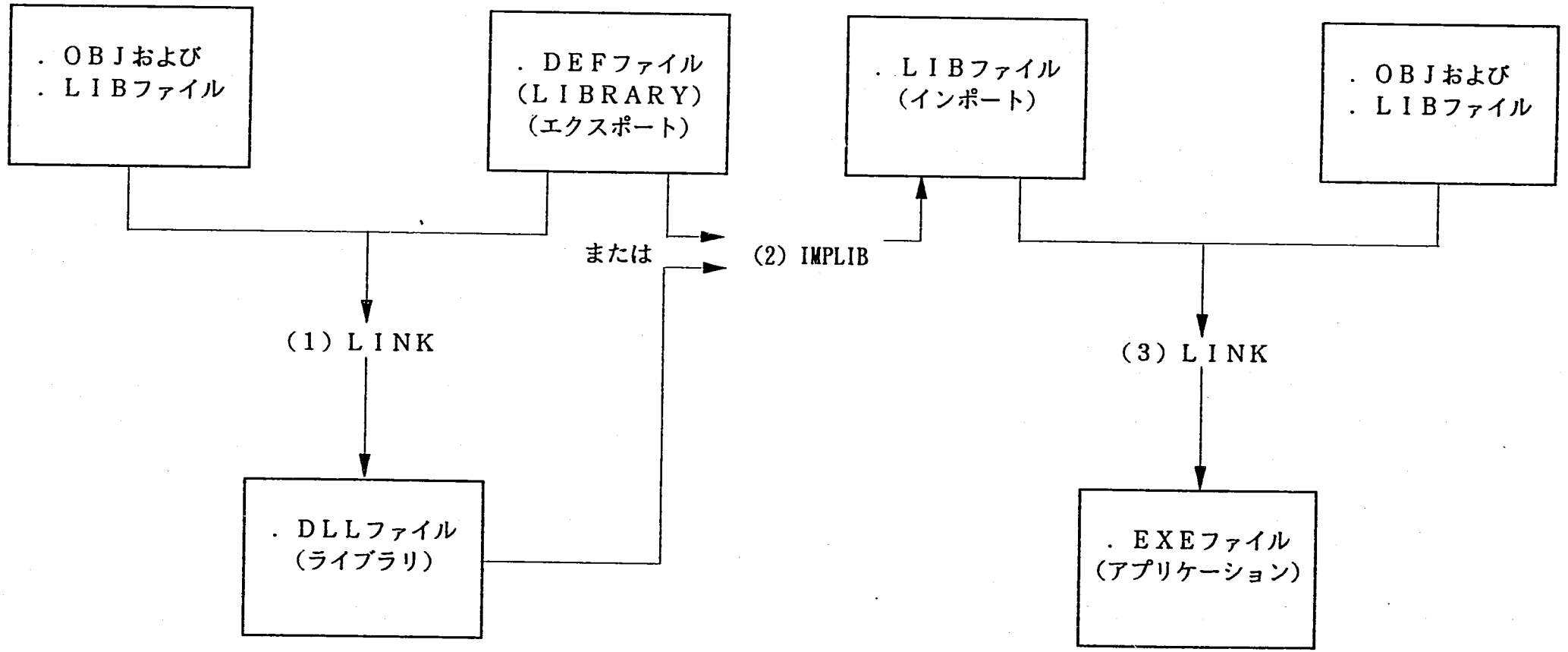


図3. 4 インポート・ライブラリを使ったリンク

## 7. 4 複数言語によるプログラミング(MultiLanguage)

Microsoft Visual Basicは、Microsoft Cで書かれたルーチンの呼び出しをさサポートしている。また、VBのルーチンからFortranのルーチンを呼び出すことができる。以降に複数プログラミングについて、それぞれの言語の違いと、その違いを処理する方法について説明する。

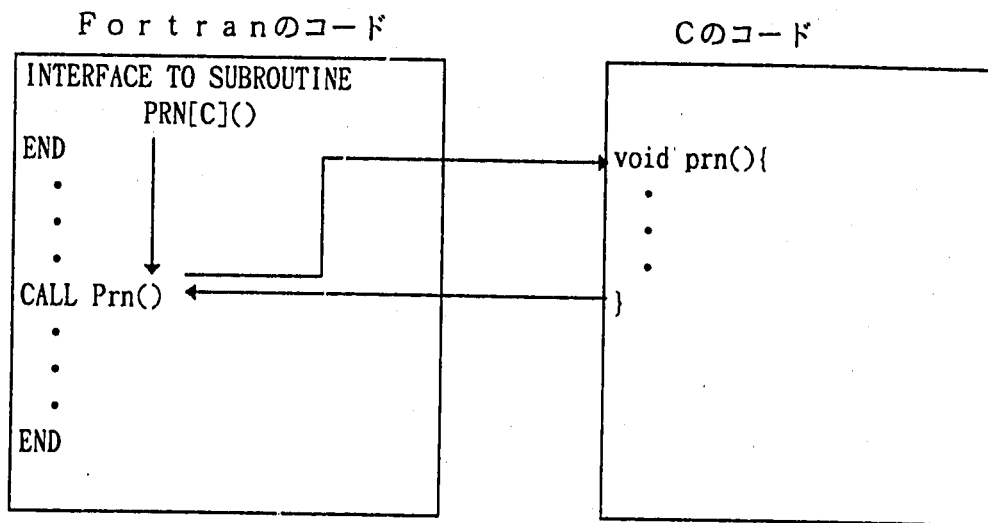
### 複数言語呼び出しの作成

Fortran以外の言語で書かれたルーチンを使いたくなる要求が出てくる。Microsoft Fortranでは新しくルーチンを作成するかわりに、例えばCで書かれた存在の関数を呼び出すことができる。

Microsoft言語で作成した複数言語呼び出しは、必ず複数のモジュールを含む。それぞれのモジュールはそれぞれの言語のコンパイラでコンパイルされている。

上の例では、メインモジュールのソースファイルはFortranのコンパイラでコンパイルし、2番目のソースファイル(Cで書かれたもの)は、Cのコンパイラでコンパイルする。これら2つのオブジェクトファイルは、FLまたはLINKでリンクする。

上で示した例を使って、複数言語呼び出しの構文について図3. 5で説明する。



どのように呼び出されるかを指定する。

図3. 5 複数言語呼び出し

この図で、FortranプログラムからのCの呼び出しはCALL Prn()であり、Fortranのサブプログラムの呼び出しと似ている。しかし、この複数言語呼び出しは、Fortranモジュール間の呼び出しとは、2つの点が異なる。

- サブプログラム Prn () は、標準的な C の構文を使って、C で実現されている。
- Fortran は、C との互換性を保つために INTERFACE 文を使う必要がある。

INTERFACE 文は、複数言語インターフェース文 (Fortran のインターフェース文) の例である。それぞれの言語は、独自のインターフェースの形式を持っている。

構文は違うが C の関数、BASIC のプロシージャ、Microsoft Assembler のルーチン、Fortran のサブルーチンと関数は、似ている点も多くある。主な違いは、ルーチンが値を返すかどうかである。戻り値があるルーチンもないものも、それぞれの言語間で置き換えができる。

表 5. 1 は、さまざまな言語におけるルーチンの呼び出しの対応を示す。

言語	戻り値あり	戻り値なし
Fortran	関数	サブルーチン
C	関数	(void)関数
BASIC	Functionプロシージャ	サブルーチン
Macro Assembler	プロシージャ	プロシージャ

表 5. 1 さまざまな言語におけるルーチンの呼び出しの対応

## 7. 5 DLL プロシージャの呼び出し

DLL (Dynamic Link Libraries) は、アプリケーションが実行時にリンクして利用できるプロシージャのライブラリで、Microsoft Windows の特徴となる機能の 1 つである。コンパイル時にリンクする静的なライブラリと異なり、アプリケーションから独立して更新することもでき、また、多くのアプリケーションで 1 つの DLL を共有することもできる。Microsoft Windows も Windows API (Application Programming Interface) を提供する複数の DLL から構成されていて、すべてのアプリケーションは DLL 内のプロシージャを使ってウィンドウやグラフィックスを表示したり、メモリの管理などを行っている。

これらの DLL 内のプロシージャは、Visual Basic のアプリケーションから呼び出すこともできる。これにより、Visual Basic 内では直接実行できない特殊な操作も実行できるようになる。また、システム内のその他の DLL プロシージャも同様に呼び出すことができる。この章の DLL プロシージャ呼び出しに関するすべての説明は、特にことわらない限り、Microsoft Windows の DLL とその他の DLL の両方に対するものである。

### 7. 5. 1 概要

DLL プロシージャは、Visual BASIC アプリケーションとは別の

ファイルに存在しする。

DLLプロシージャを使うには、`Declare`ステートメントで使いたいプロシージャを宣言する必要がある。一度DLLプロシージャを宣言すれば、それ以後はプログラム内で他のプロシージャと同様に使うことができる。ただし、DLLプロシージャに引数を渡すときは、十分に注意しなければならない。

DLLプロシージャを使うときの基本的な手順は、次のとおりである。

1. DLLプロシージャの使用を`Declare`ステートメントで宣言する。
2. 実際にそのプロシージャを呼び出すDLLプロシージャは、一度だけ宣言する。一度宣言すれば何回でも呼び出せる。

#### DLLプロシージャの宣言

DLLプロシージャを宣言するときは、フォームモジュールまたはコードモジュールの宣言セクションに`Declare`ステートメントを記述する。DLLプロシージャをフォームモジュールの宣言セクションで宣言すると、プロシージャはそのフォーム内からのみ呼び出せるようになる。また、DLLプロシージャをコードモジュールで宣言すると、プロシージャはアプリケーションのどこからも呼び出せる。

プロシージャが値を返さないときは、次のように`Sub`プロシージャとして宣言すること。

```
Declare Sub InvertRect Lib "User" (ByVal hDC As Integer, aRect As Rectangle)
```

プロシージャが値を返すときは、`Function`プロシージャとして宣言する。

```
Declare Function GetSystemMetrics Lib "User" (ByVal n As Integer) As Integer
```

`Declare`ステートメントの`Lib`と`ByVal`キーワードに注目する。この`Declare`ステートメントには、オプションで`Alias`キーワードを含めることもできる。

#### DLLプロシージャの呼び出し

DLLプロシージャは、一度宣言すれば、`Visual Basic`のステートメントまたは関数とまったく同様に呼び出せる。

```
Sub Form_Load()
  Const SM_MOUSEPRESENT = 19
  Show
  If GetSystemMetrics (SM_MOUSEPRESENT) Then Print "ABCDEFGH"
End Sub
```

**重要** DLLプロシージャの呼び出しでは、プロシージャに渡す値が正しいかどうかはチェックされない。不正な値を渡すと、プロシージャの実行に失敗してアプリケーションが異常終了してしまうことがある。また保存していなかったコードなどが失われる可能性がある。DLLプロシージャを呼び出すコードを実行するときは、作

業を頻繁に保存するなど十分に注意する必要がある。

#### DLLプロシージャを宣言するときの注意事項

これまでの例で示したように、通常、DLLプロシージャの宣言はかなり複雑なものになりがちだが、一度DLLプロシージャを適切に宣言すれば、あとはアプリケーション内の値のプロシージャと同様に宣言したプロシージャを呼び出すだけになる。

ただし、DLLプロシージャに値を渡すときには十分に注意すること。

DLLプロシージャに値を渡す方法については、以降に示す。

Microsoft Windowsに含まれていないDLLのプロシージャを呼び出したいときは、プロシージャを適切な形で宣言する必要がある。ここでは、各プロシージャに適した正しい宣言を作成できるように、Declareステートメントの複文を詳しく説明する。

#### プロシージャを含むDLLの指定

DeclareステートメントのLib libname 節では、子のプロシージャを含むDLLを指定する。Microsoft WindowsのDLLの場合は、"User", "GDI", "Kernel", 他のシステムDLL("MMsystem"など)のいずれかである。その他のDLLの場合は、次の例のようにパス指定も含んだファイル名をlibnameに指定する。

```
Declare Function lzCopy& Lib "C:\win\lzexpand.dll" (ByVal S%, ByVal D%)
libname は大文字と小文字を区別しないことに注意すること。
```

#### 引数の値による引渡しと参照による引き返し

VisualBasicのすべての引数は、特に指定しないと、参照による引き渡し（32ビットのfarアドレス）で渡される。しかし、多くのDLLプロシージャでは、引き数を値による引渡しで渡されなければならない。このようなプロシージャに対し、引数を参照による引き渡しで渡すと、そのプロシージャは正常に動作しなくなる。

引き数を値による引き渡しで渡すためには、Declareステートメントの引数宣言の前にByValキーワードを記述する。こうすれば、そのプロシージャが呼び出されるたびに、引き数が必ず値による引渡しで渡されることになる。

例えば、次に示すInvertRectプロシージャは、第1引数を値による引き渡しで受け取り、第2引数を参照による引き渡しで受け取る。

```
Declare Sub InvertRect Lib "User" (ByVal hDC%, aRect As Rectangle)
```

注意 C言語では、配列以外のすべての引数が値による引き渡しで渡される。DLLプロシージャについてのドキュメントが、C言語の構文を使って説明されている場合は、このことに注意しなければならない。

#### 複数のデータ型を受け取る引数の宣言

DLLプロシージャには、1つの引数に複数のデータ型を受け取ることもできるものもある。このようなプロシージャに複数のデータ型を渡せるようにするには、型の制限を取り除くためにその引数を `As Any` として宣言する。

```
Declare Function SendMessage Lib "User" (ByVal hWnd As Integer, ByVal msg As Integer, ByVal wParam As Integer, lp As Any) As Long
```

次の例では、プロシージャの最後の引数に文字列または長整数を与えて呼び出せる。

```
Sub ScrollIt (AnyText As TextBox, Chars As Integer, Lines As Integer
    Const EM_LINESCROLL = WM_USER + 6
    Scroll = chars * 65536 + lines
    Dummy = SendMessage(AnyText.hWnd, EM_LINESCROLL, 0, ByVal Scroll)
End Sub
```

```
Function FindItem (aList As ListBox, target As String) As Integer
    Const WM_USER = &H400, LB_FINDSTRING = WM_USER + 16
    FindItem = SendMessage(aList.hWnd, )Scroll = chars * 65536 + lines
    Dummy = SendMessage(AnyText.hWnd, LB_FINDSTRING, -1, ByVal Target)
End Function
```

`As Any` を使って型の制限を取り除くと、その引数は参照による引渡で渡されることに注意すること。引数を値による引渡で渡すには、そのプロシージャを実際に呼び出すときに、`ByVal` を使う。また、文字列を渡すときには `BAS` `IC` の文字列を `NULL` 文字で終了する文字列に変換するために `ByVal` を使う必要がある。

## 7. 5. 2 DLLプロシージャ呼び出しの特殊なデータ型の利用

### ユーザー定義型

引数としてユーザー定義型を受け取るDLLプロシージャもある。ユーザー定義型は、C言語では「構造体」、Pascalでは「レコード型」として参照される。DLLのドキュメントではC言語の用語が多く使われている。

ユーザー定義型の個々の要素は、配列の場合と同様に、通常の数値変数や文字列変数を渡すときと同じ方法でDLLプロシージャに渡すことができる。

参照による引き渡しで渡すと、ユーザー定義型全体を1つの引数として渡すことができる。このときは、まずユーザー定義型の最初の要素のアドレスが渡され、残りの要素は最初の要素に続くメモリに配置される。ユーザー定義型を値による引き渡しで渡すことはできない。

例えばMicrosoft WindowsのDLLのいくつかのプロシージャは、次のような構造のユーザー定義型を受け取る。



```
Type Rectangle
```

```
Left As Integer
```

```
Top As Integer
```

```
Right As Integer
```

```
Bottom As Integer
```

```
End Type
```

Rectangle型を受け取るプロシージャには、指定した短形領域の外側に点線の外枠を描くDrawFocusRect、指定した短形領域の色を反転させるInvertRectなどがある。これらのプロシージャを使うには、次のような宣言をフォームモジュールまたはコードモジュールの宣言セクションに記述する。

```
Declare Sub DrawFocusRect Lib "User" (ByVal hDc As Integer, aRectMc As Rectangle)
```

このように宣言すると、次のサブプロシージャでこれらの2つのDLLプロシージャを呼び出せる。MouseRectはマウスのドラッグにつれて短形を表示し、InvertRectはマウスボタンを離れたときに短形領域を反転させるものである。

表7. 1 に実際のDLLプロシージャ手続きファイルの例を示す。



```

; in Windows standard or enhanced mode.
;
;STUB          'WINSTUB.EXE'
;
; The STUB statement pulls in an executable file which is run when the
; application is run independently of Windows 3.0. WINSTUB.EXE displays
; the message 'This program requires Microsoft Windows' when executed.
;
CODE           PRELOAD MOVEABLE DISCARDABLE
DATA          PRELOAD MOVEABLE SINGLE
;DATA        PRELOAD MOVEABLE MULTIPLE
;
; The CODE and DATA statements set the attributes of the applications CODE and
; DATA segments. See chapter 22 in the Environment and Tools manual for
; a complete description of the options for these statements.
;
HEAPSIZE 10240
STACKSIZE      8000
;
; The HEAPSIZE and STACKSIZE statements set the applications near heap and
; stack sizes. The values specified are recommended for QuickWin applications.
; See chapter 22 in the Environment and Tools manual for a complete
; description of these statements.
;
;
; Uncomment these lines for DLL module definition file model.
;
EXPORTS
    Mset
;    Rlx1
    Rlx2
;    Mclist
    Crlist
    Sprin
    Suval
    Smval

```

```
Eepm  
Eepr  
Sepm  
Sepr  
; Terr  
Serr  
; Ecr  
; Eerr  
; Ter  
Anf  
Sids  
Dstar  
Ddstar  
Alpha  
Xy  
Calcnu
```

```
;  
; The above section should be uncommented if this .DEF file is to be used as  
; a model for a FORTRAN 5.1 dynamic-link library (DLL) .DEF file. The WEP  
; routine is included in the FORTRAN 5.1 startup code, but the symbol must  
; be explicitly exported. If a DLL already has a user-defined WEP routine,  
; the routine should be renamed __WEP. The startup code will call __WEP,  
; if present, during DLL termination. For additional information concerning  
; the WEP function, see the Windows 3.0 SDK documentation.  
;
```

## 8. STAR-NETシステムとの連結

MS-EXCELのマクロ言語により構築された、構造物強度データベースシステム「STAR」というシステムがあり、この中からも本プログラムを呼び出している。

入出力データ及び受け渡しの方法は、この報告書の中で説明している単体動作でのものと全く同じである。但し起動/終了時は自動的に本プログラムが呼び出され、終了する。

また、本プログラム内部で終了間際に親プログラムを再起動させるための関数を呼び出さなければならない。

### 8. 1 親プログラムを再起動させる方法

Windowsは、ノン・プリエンパティブなマルチタスクOSである。各アプリケーションが自主的にCPU占有権を譲りあってはじめてマルチタスクが実現する方式である。そのため1つのアプリケーションがCPUを独占してしまうと他のアプリケーションはずっと待機していなければならない。

ちなみにUNIXは、プリエンパティブなマルチタスクOSである。アプリケーションに依存するのではなく、OSが強制的に非常に短い時間幅でアプリケーションにCPUの実行時間を割り当てるのである。これをTSS(TimeSharingSystem)/時分割方式という。

Windows環境のマルチタスクは、親プログラムが子プログラムを呼び出すと子プログラムの終了を待たずに先に処理を進めてしまう、そのため制御の流れが制御不能になってしまう。そこで次の様な手順を踏む事でこの問題を回避する事が出来る。

- (1) 親プログラムが子プログラムを実行する。
  - (2) 親プログラムは、自分を待機状態にする。
  - (3) 子プログラムが処理を実行する。
  - (4) 子プログラムは、処理を終了したら最後に親プログラムを再起動する関数を実行してプログラムを終了する。
- 処理は、プロセス間通信/DDEによって実現される。

## 9. 操作方法

### 9. 1 環境設定

※ 以下の準備を行わないと本プログラムは、正常に動作しない。

#### [1] Visual Basicライブラリ・ファイルのコピー

以下の4ファイルをWindows/Systemディレクトリーにコピーする。

- (1) VBRJP200.DLL (R:¥Uty1¥Vb¥)
- (2) GRID.VBX (R:¥Uty1¥Vb¥)
- (3) OLECLIEN.VBX (R:¥Uty1¥Vb¥)
- ※ (4) MMFUNC.DLL (S:¥Star¥Sonsho¥Vb版¥)  
損傷解析プログラム用材料特性ライブラリ

手順：(DOS/Vユーザ-の場合)

```
¥>COPY R:¥UTY1¥VB¥VBRJP200.DLL C:¥WINDOWS¥SYSTEM[Return]
¥>COPY R:¥UTY1¥VB¥GRID.VBX C:¥WINDOWS¥SYSTEM[Return]
¥>COPY R:¥UTY1¥VB¥OLECLIEN.VBX C:¥WINDOWS¥SYSTEM[Return]
¥>COPY S:¥STAR¥SONSHO¥VB版¥MMFUNC.DLL C:¥WINDOWS¥SYSTEM[Return]
```

#### [2] 本プログラムのアイコン登録

- (1) 本プログラムをローカルの任意のディレクトリーにコピーする。

```
¥>COPY S:¥STAR¥SONSHO¥VB版¥sonsho.exe C:¥ [Return]
```

- (2) 本プログラムをWindowsにアイコン登録する。

### 9. 2 起動方法

- (1) アイコン登録した本プログラムをダブル・クリックする。
  - (2) MS-EXCEL Ver4.0を起動する。
- そして以下のファイルをオープンする。

- ① 試験条件シート (入力)
  - ② 解析結果シート (入力)
  - ③ 損傷計算結果シート (出力)
- ※ これらのシートに関しては「3. 2 入出力仕様」を参照の事。

(3) 「損傷計算実行」ボタンをクリックする。

すると損傷プログラムが実行を開始する。MS-EXCELからブロック単位でデータを受け取り、評価法で指定した計算を行い、結果データをまたMS-EXCELに通信で渡す。

この流れを何回か繰り返して全データを処理する。

最終的な結果は、MS-EXCELの結果データ格納シートに保持される。

このデータを基にグラフ化等のデータ整理を行ない、EXCELシートファイルとして保存する。

(4) メニュー画面左上のボタンをクリックし「閉じる」を選択する。

これで「損傷プログラム」を終了する。

MS-EXCELについては、作業後通常の方法で終了する。

## 10 結 言

従来MS-DOS環境で動作していた「損傷計算プログラム」をMS-Windows環境で動作するように移植作業をおこない、その結果当初の目的をほぼ達成する事が出来た。

マルチタスク環境に於いて親プログラムと子プログラムのプロセス連携も問題無く行われ、Windows特有の通信手段であるDDE通信も予想以上に機敏に動作している。機能毎にモジュールの独立化を図る目的で材料特性ライブラリは、DLL形式というライブラリ・ファイルに切離した。

また計算プログラム本体をVisual Basic言語で記述したので、BASIC言語を多少知っていればプログラム・コードをいじり、利用者の要求に対応出来るという配慮も行われている。実行形態も単体動作が出来るようになり、前バージョンに比べれば手軽に損傷結果データを得やすくなった。

通常ソフトウェア開発を行う際、幾つかの基準を満たすよう意識しながら作業を行う。具体的には次の3つの基準(1)拡張性(2)再利用性(3)操作性が挙げられる。その他さまざまな基準はあるが代表的なものとして上記3つを挙げる。

これらの基準がより満たされるという事は、機能の追加・変更がし易い、開発作業工数が減少する、正確さ・頑丈さが増す、より使い易くなる等の効果が得られるということである。

パーソナル・コンピュータを取り巻くハードウェア/ソフトウェア環境を見渡すと、実に目まぐるしく技術は進歩している。とりわけハードウェアの進歩は凄まじいものがある。ソフトウェアの方も大きな変革期に差し掛かっている。「オブジェクト指向によるソフトウェア開発技術」という従来とは全く異なる考え方が導入されつつある。

この考え方は、従来のソフトウェア工法に比べ高い水準で3つの基準を満たす事が出来るといわれている。

本プログラムをあらためて上記の基準に照らし合わせると、いろいろな問題点が浮かんでくる。

- (1) 入出力を担当するアプリケーション(MS-EXCEL)が特定バージョンに依存している。
- (2) 損傷プログラム本体を機能毎にライブラリ形式として独立させる。
- (3) 外部プログラムとの連携を考慮したI/Oインターフェース仕様を作成する。
- (4) 今後の機能/データ追加という拡張に耐えうるだけのモジュール構造にする。
- (5) 材料特性値算出用ライブラリをきれいなモジュール構造に整理する。

更に最新仕様に対応したモジュール構成にする。

等の改善点が挙げられる。これらは体系的に整理して改良計画を立てなければならない。

オブジェクト指向技術を導入すると具体的に以下のような利点を得られる。

- (1) 独立性が高いソフトウェア構造が実現出来る。  
機能毎に独立したオブジェクトという単位に分割し、それらが多数存在し、互いに通信によって連携を取り合い実行を行う。
- (2) 労力を軽減し、生産性を高めることができる。  
標準的な評価法モジュールの型紙を1つ定義しておけば、後はそれに必要な機



能を追加してゆくだけで新たな評価法オブジェクトを作成することができる。

オブジェクト指向の開発技術は、これから広く開発分野に浸透してゆくと思われる。  
なによりこの工法は、応用範囲が広いので是非とも考え方を習得し、他のシステム  
に活かしてゆきたいと願う。

謝 辞

本システムの構築を進めるにあたり（株）I T J 濱田氏には有益なご意見並びに  
多大なる技術的支援を頂きました。  
ここに感謝の意を表します。

## 参 考 文 献

- (1) アスキー出版局 『オブジェクト指向入門』
- (2) マイクロソフト (株) 『Microsoft FORTRAN Ver5.0  
Programming Guide』
- (3) マイクロソフト (株) 『Microsoft FORTRAN Ver5.0  
Users Guide』
- (4) マイクロソフト (株) 『Microsoft FORTRAN Ver5.0  
Language Reference』
- (5) マイクロソフト (株) 『Microsoft FORTRAN Ver5.0  
CodeView Users Guide』
- (6) マイクロソフト (株) 『Microsoft Visual Basic Ver2.0  
Programming Guide』
- (7) マイクロソフト (株) 『Microsoft Visual Basic Ver2.0  
Users Guide』
- (8) マイクロソフト (株) 『Microsoft Visual Basic Ver2.0  
Language Reference』
- (9) マイクロソフト (株) 『Microsoft Visual Basic Ver2.0  
CodeView Users Guide』
- (10) (株) 日本電気 『MS-Windows 3.1 プログラマーズ  
ガイド』
- (11) マイクロソフト (株) 『MS-DOS 5.0 ユーザーズ  
リファレンス マニュアル』
- (12) 技術管理室 『PNC 所内LAN利用者マニュアル』
- (13) (株) アスキー 『DOS/Vスーパーチューニング』
- (14) Winn L. Rosch, Brady Publishing. 『The Winn Rosch Bible 2nd Ed』
- (15) (株) アスキー 『MS-DOS マルチタスク・プログラミング』

(16) (株)技術評論者 『Windows 3.0リアルタイム』

(17) 笠原、他：「過渡熱応力リアルタイムシミュレーションコードPARTSの開発(1)プロトタイプ的设计」. PNC ZN9410 95-

付録A ソース・コードNo. 1 / V i s u a B a s i c 言語  
(損傷プログラム本体)

```

1 .
2 .  損傷解析プログラム for Windows
3 .
4 .      Operating : Microsoft DOS Ver5.0(or IBM DOS/V Ver5.0J) +
5 .          System  Microsoft Windows Ver3.1
6 .
7 .      Language  : Microsoft Visual Basic for Windows Ver2.0 +
8 .                  Microsoft FORTRAN for DOS Ver5.0
9 .
10 .
11 .
12 .
13 .
14 .
15 .
16 .
17 .      LinkMode (forms and controls)
18 Global Const NONE = 0
19 Global Const LINK_SOURCE = 1
20 Global Const LINK_AUTOMATIC = 1
21 Global Const LINK_MANUAL = 2
22 Global Const LINK_NOTIFY = 3
23 .
24 Global Const IN = 50
25 Global Const MNO = 16
26 Global Const METHOD_END = -1
27 .
28 Global Method_name(20, 10) As String
29 Global S2x(), Dcx()
30 Global S2xx(1), Dcxx(1)
31 Global Lcount
32 .
33 .  注意 : 配列の添字は「0」から始まる。
34 .      例えば Mcdim(50) は 0~50の51個の要素を持つ配列となる。
35 .
36 .  構造体宣言/1 : 試験条件シート要素(Jyoken.in)
37 .
38 Type Condition
39     Material      As String
40     Tempc         As Single
41     TempH         As Single
42     Hldtmc        As Single
43     Hldtmh        As Single
44     Mecha         As Single
45     Cycle         As Integer
46     MethodName    As String
47     MethodNo      As Integer
48     Method(MNO+1) As Integer
49     AnalSheetName As String
50     AnalSheetRow  As Integer
51     AnalSheetCol  As Integer
52     OutSheetName  As String
53     Imate         As Integer
54 End Type
55 Global Cond As Condition
56 .
57 .
58 .  構造体宣言/2 : Range解析結果シート要素(Sonsho.in)
59 .
60 .  補 足      : (1) 現在「起点座標」のみ領域確保
61 .              理由 : 損傷計算処理は「終点座標」
62 .                  を必要としないため。
63 .              (2) 現在「時点2」まで領域確保

```

```

64 '          理由：メモリの削減。
65 '          「時点3」は極希に使用する程度であるため。
66 '
67 '   メモリサイズ(バイト) : 61x4xIN=
68 '
69 Type Analysis
70   R(IN, 0)      As Single      ' R座標 (起点座標のみ)
71   Z(IN, 0)      As Single      ' Z座標 ( " )
72   H(IN, 0)      As Single      ' θ座標 ( " )
73   Tmpmax(IN)    As Single      ' 最高金属温度
74   Ptime(IN, 1)  As Single      ' 評価時刻 (時点2まで)
75   Temp(IN, 1)   As Single      ' 表面温度 ( " )
76   Ltemp(IN, 1)  As Single      ' 等価線形温度 ( " )
77   Mtemp(IN, 1)  As Single      ' 平均温度 ( " )
78   Strs(IN, 3, 1) As Single      ' 表面応力 ( " )
79   Lstrs(IN, 3, 1) As Single      ' 等価線形応力 ( " )
80   Mstrs(IN, 3, 1) As Single      ' 膜応力 ( " )
81   Ee(IN, 3, 1)  As Single      ' 弾性歪 ( " )
82   Ep(IN, 3, 1)  As Single      ' 塑性歪 ( " )
83   EC(IN, 3, 1)  As Single      ' クリープ歪 ( " )
84   Q(IN)         As Single      ' Q
85 End Type
86 Global Analy As Analysis
87 '
88 '
89 '   構造体宣言/3 : 基本変数要素
90 '
91 '          メモリサイズ(バイト) : 6x4xIN=
92 '
93 Type Bscval
94   Tres(IN)      As Single      ' 相当応力範囲表面応力
95   Ltres(IN)     As Single      ' 相当応力範囲等価線形応力
96   Vms(IN)       As Single      ' 相当応力範囲表面応力
97   Eepceq(IN)    As Single      ' 相当歪範囲弾塑性クリープ歪範囲
98   Epeq(IN)      As Single      ' 相当歪範囲塑性歪範囲
99   Eceq(IN)      As Single      ' 相当歪範囲クリープ歪範囲
100 End Type
101 Global Bscval As Bscval
102 '
103 '
104 '   構造体宣言/4 : 各種変数要素
105 '
106 '          メモリサイズ(バイト) : 23x4=92
107 '
108 Type Mc
109   Im           As Integer
110   Tc           As Single
111   E            As Single
112   Sy           As Single
113   Sp           As Single
114   Am           As Single
115   Ak           As Single
116   Spc         As Single
117   Amc         As Single
118   Akc         As Single
119   Tr0         As Single
120   Tr1         As Single
121   Tr2         As Single
122   Su          As Single
123   Sm          As Single
124   Sl          As Single
125   Alph1      As Single
126   S2         As Single

```

```

127 Alph2 As Single
128 G1 As Single
129 Beta1 As Single
130 G2 As Single
131 Beta2 As Single
132 F As Single
133 Ramda As Single
134 A0t As Single
135 A0r As Single
136 A1t As Single
137 A1r As Single
138 A2t As Single
139 A2r As Single
140 A4t As Single
141 A4r As Single
142 End Type
143 Global Mc As Mc
144 '
145 '
146 ' 構造体宣言/6(全評価法共通) : 損傷計算結果への変数
147 '
148 ' メモリサイズ(バイト) : 23x4xIN=
149 '
150 Type Methods
151 Df(IN) As Single ' D f
152 Dc(IN) As Single ' D c
153 Dcr(IN) As Single ' D c r
154 D(IN) As Single ' D
155 Ef(IN) As Single ' 熱ピーク歪
156 Et(IN) As Single ' 全歪範囲
157 K(IN) As Single ' 応力集中係数
158 Ked(IN) As Single ' K e 係数
159 Ke(IN) As Single ' 歪集中係数
160 Kedg(IN) As Single ' 総体的歪集中係数
161 Kedl(IN) As Single ' 局所的歪集中係数
162 Kep(IN) As Single ' 歪集中係数
163 Neuber(IN) As Single ' NEUBER 則
164 Qn(IN) As Single ' 弾性追従係数
165 Ql(IN) As Single ' 局所的弾性追従係数
166 Qnu(IN) As Single ' 多軸効果を現す弾性追従係数
167 QF(IN) As Single ' 弾性追従係数
168 Qw(IN) As Single ' 熔接金属に対する弾性追従係数
169 Qeff(IN) As Single ' 弾性追従係数
170 Qbm(IN) As Single ' 母材に対する弾性追従係数
171 Spx(IN) As Single ' 表面応力
172 Sn(IN) As Single ' 等価線形応力
173 Si(IN) As Single ' 初期応力
174 End Type
175 Global Allmtd As Methods
176 '
177 '
178 ' DLLプロシージャの宣言 (from FORTRANライブラリ)
179 '
180 '
181 Declare Sub Mset Lib "Mmfunc.dll" (ByVal Flg As Integer, Mc As Mc)
182 ' Declare Sub R1x1 Lib "Mmfunc.dll" (ByVal S0!, ByVal Ec1!, ByVal Alphc!, ByVal Alpha!, ByVal Qx!,
ByVal Dt!, S2x As Any, Dcx As Any, Mc As Mc)
183 Declare Function Calcnu Lib "Mmfunc.dll" (Mc As Mc) As Single
184 Declare Function Anf Lib "Mmfunc.dll" (ByVal Et!, ByVal Dot!, Mc As Mc) As Single
185 Declare Function Sepm Lib "Mmfunc.dll" (ByVal X!, Mc As Mc) As Single
186 Declare Function Sepr Lib "Mmfunc.dll" (ByVal X!, Mc As Mc) As Single
187 Declare Function Alpha Lib "Mmfunc.dll" (Mc As Mc) As Single
188 Declare Function Dstar Lib "Mmfunc.dll" (ByVal Sn!, ByVal Tstar!, ByVal DC1!, Mc As Mc) As Single

```



```

189 Declare Function Ddstar Lib "Mmfunc.dll" (ByVal Et!, ByVal Tstar!, ByVal DC1!, Mc As Mc) As Single
190
191
192
193  損傷解析管理ルーチン : 50行/1ループ
194
195 Sub Allmtd_proc ()
196   Dim I, Offset1, Res, Loffdat      As Integer
197
198   Loffdat = Cond.AnalSheetRow Mod IN      'Excelシート・オフセット行数2
199
200   For I = 0 To Lcount - 1
201     If (I + 1) = Lcount And Loffdat > 0 Then
202       Offset1 = Loffdat      'Excelシート・オフセット行数2
203     Else
204       Offset1 = IN          'Excelシート・オフセット行数2
205     End If
206
207     Call Msg_proc(0, Lcount, I + 1)      '処理状況表示
208     Call Input_Data_2(I, Offset1)      '解析データ入力
209
210     Call Msg_proc(1, Lcount, I + 1)      '処理状況表示
211     Call Bscvl(Offset1)      '基本変数算出用ライブラリ
212
213     Call Msg_proc(2, Lcount, I + 1)      '処理状況表示
214     Select Case Cond.MethodNo
215       Case 0
216         Call Ttdsp(Offset1)      '損傷値数値解析
217         Call Msg_proc(3, Lcount, I + 1) '処理状況表示
218         Call Tdsout(Offset1)     '結果データDDE通信準備
219         IdNo = 10                '結果項目数
220       Case 1
221         Call Bdsp(Offset1)
222         Call Msg_proc(3, Lcount, I + 1)
223         Call Bdsout(Offset1)
224         IdNo = 16
225       Case 2
226         Call Bdsmdp(Offset1)
227         Call Msg_proc(3, Lcount, I + 1)
228         Call Bmdout(Offset1)
229         IdNo = 18
230       Case 3
231         Call Ddsp(Offset1)
232         Call Msg_proc(3, Lcount, I + 1)
233         Call Ddsout(Offset1)
234         IdNo = 17
235       Case 4
236         Call Ddswldp(Offset1)
237         Call Msg_proc(3, Lcount, I + 1)
238         Call Dwdout(Offset1)
239         IdNo = 18
240       Case 5
241         Call Typelp(Offset1)
242         Call Msg_proc(3, Lcount, I + 1)
243         Call Typout(Offset1)
244         IdNo = 17
245       Case 6
246         Call Type2p(Offset1)
247         Call Msg_proc(3, Lcount, I + 1)
248         Call Typout(Offset1)
249         IdNo = 17
250       Case 7
251         Call Eqefp(Offset1)

```

```

252         Call Msg_proc(3, Lcount, I + 1)
253         Call Eqfout(Offset1)
254         IdNo = 21
255     Case 8                                     'EQEF-WELD START : 21
256         Call Eqefwdp(Offset1)
257         Call Msg_proc(3, Lcount, I + 1)
258         Call Ewdout(Offset1)
259         IdNo = 21
260     Case 9                                     'PNC START : 21
261         Call Pncp(Offset1)
262         Call Msg_proc(3, Lcount, I + 1)
263         Call Pncout(Offset1)
264         IdNo = 21
265     Case 10                                    'EQEF15 START : 21
266         Call Eqef15p(Offset1)
267         Call Msg_proc(3, Lcount, I + 1)
268         Call Eqfout(Offset1)
269         IdNo = 21
270     Case 11                                    'EQEF167 START : 21
271         Call Eqef167p(Offset1)
272         Call Msg_proc(3, Lcount, I + 1)
273         Call Eqfout(Offset1)
274         IdNo = 21
275     Case 12                                    'INELAST START : 10
276         Call Inelastic(Offset1)
277         Call Msg_proc(3, Lcount, I + 1)
278         Call Tdsout(Offset1)
279         IdNo = 10
280     Case 13                                    'SIEQ167 START : 21
281         Call Sieq167p(Offset1)
282         Call Msg_proc(3, Lcount, I + 1)
283         Call Ewdout(Offset1)
284         IdNo = 21
285     Case 14                                    'EQEF167-WELD START : 21
286         Call Eqef16wp(Offset1)
287         Call Msg_proc(3, Lcount, I + 1)
288         Call Ewdout(Offset1)
289         IdNo = 21
290     Case 15                                    'EQEF167-WELD-K START : 21
291         Call Eqef16wk(Offset1)
292         Call Msg_proc(3, Lcount, I + 1)
293         Call Ewdout(Offset1)
294         IdNo = 21
295     Case Else
296         'Msg = " 評価法不明!! "
297         'MsgBox0(Msg)
298     End Select
299     Call Msg_proc(4, Lcount, I + 1)           ' 処理状況表示
300     Call Outdat(I, Offset1, IdNo)           ' 結果データDDE通信対MS-EXCELシート
301
302     Next I
303
304     Call Msg_proc(99, Null, Null)           ' 処理状況表示
305
306 End Sub
307
308
309     引数に指定した数値の常用対数を(10を底とする対数)を計算する
310
311 Function ALog10 (X)
312     Dim X      As Single
313
314     ALog10 = Log(X) / Log(10!)

```

```

315
316 End Function
317
318
319   BDS-MD      熱ピークひずみ算出改訂  1993.3.4  N. K
320
321 Sub Bdsmdp (Icount)
322
323   Dim Icount As Integer
324   Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
325   Dim KEDD As Single
326   Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
327   Dim K1, K2, KE1, KE2 As Single
328   Dim K1D, K2D, KED1D, KED2D, KT As Single
329   Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
330   Dim LSIG1, LSIG2, LSIG3 As Single
331   Dim NU, NU1, NU2, NF           As Single
332   Dim INF                       As Long
333   Dim EC1, ALPHC, ALPHR        As Single
334
335   Iflg = 0
336
337   For I = 0 To Icount - 1
338   -----
339   歪範囲の計算
340   -----
341   -----  T I M E 1
342   Mc.Tc = Analy.Mtemp(I, 0)
343   Call Mset(Iflg, Mc)           'Fortran Library(DLL) 呼び出し
344   E1 = Mc.E
345   NU1 = Calcnu(Mc)             'Fortran Library(DLL) 呼び出し
346   ALPHA1 = Alpha(Mc)
347   -----  T I M E 2
348   Mc.Tc = Analy.Mtemp(I, 1)
349   Call Mset(Iflg, Mc)           'Fortran Library(DLL) 呼び出し
350   E2 = Mc.E
351   NU2 = Calcnu(Mc)             'Fortran Library(DLL) 呼び出し
352   ALPHA2 = Alpha(Mc)
353
354   SF1 = (-E1 * ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0))) / (1 - NU1)
355   SF2 = (-E2 * ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2)
356
357   SANR = Abs(Analy.Strs(I, 0, 1) - Analy.Strs(I, 0, 0))
358   SANZ = Abs(Analy.Strs(I, 1, 1) - Analy.Strs(I, 1, 0))
359   If SANR >= SANZ Then
360     SIGR = (Analy.Strs(I, 0, 0) - SF1) - (Analy.Strs(I, 0, 1) - SF2)
361     SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
362   Else
363     SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
364     SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)
365   End If
366   SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
367   SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
368   SIG1 = Prin1(SIGR, SIGZ, SIGT)
369   SIG2 = Prin2(SIGR, SIGZ, SIGT)
370   SIG3 = SIGH
371   Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
372
373   Allmtd.Sn(I) = Bscval.Ltres(I)
374   ^^ Debug
375   WRITE(6, 8600) SF1, SF2, SIGR, SIGZ, SIGH, Allmtd.Spx(I)
376   8600 FORMAT(1H, 'SF1:', E12.6, ' SF2:', E12.6, ' SIGR:', E12.6
377   1, ' SIGZ:', E12.6, ' SIGH:', E12.6, ' SPX:', E12.6)

```

```

378
379 -----最高金属温度による  $\epsilon_n$ 、S r h の計算
380   Mc.Tc = Analy.Tmpmax(I)
381   Call Mset(Iflg, Mc)                               'Fortran Library(DLL) 呼び出し
382   EH = Mc.E
383   EN = Allmtd.Sn(I) / EH
384
385   If Allmtd.Sn(I) = 0! Then
386     KD = 0!
387   Else
388     KD = Allmtd.Spx(I) / Allmtd.Sn(I)
389   End If
390   Allmtd.K(I) = Nmax(KD, 1!, 1!)
391   Call Rlx1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmh, SRH, DCX, Mc)
392   Call Rlx1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, CSng(Cond.Hldtmh))
393
394 -----コールド温度による S r c の計算
395   Mc.Tc = Analy.Temp(I, 1)
396   Call Mset(Iflg, Mc)                               'Fortran Library(DLL) 呼び出し
397   BAR3SM = S2xx(0) + 1.5 * Mc.Sm
398   Allmtd.Qn(I) = 3!
399   Allmtd.Ked(I) = Nmax(1! + (Allmtd.Qn(I) - 1!) * (1! - BAR3SM / Allmtd.Sn(I)), 1!, 1!)
400   Allmtd.Neuber(I) = FnuBr2(Allmtd.Sn(I), Allmtd.K(I), S2xx(0), EH)
401   Allmtd.Ke(I) = Nmax(Allmtd.K(I) * Allmtd.Ked(I), Allmtd.Neuber(I), Allmtd.Neuber(I))
402
403   KT = 1!
404 -----200°Cにおける E 0,  $\alpha$  0 の計算          1993.3.4 N. K
405   Mc.Tc = 200
406   Call Mset(Iflg, Mc)                               'Fortran Library(DLL) 呼び出し
407   E200 = Mc.E
408   ALPHA200 = Alpha(Mc)
409 -----修正ポアソン比の計算, 許容ひずみ範囲 0.5%固定          1993.3.4 N. K
410   NUB = .5 - (.5 - NU1) * (BAR3SM / (EH * .005))
411 -----熱ピーク歪みの計算          1993.3.4 N. K
412   Allmtd.Ef(I) = E200 * ALPHA200 * Abs((Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) - (Analy.Temp
(I, 1) - Analy.Ltemp(I, 1))) / (EH * (1 - NUB))
413
414   Allmtd.Et(I) = Allmtd.Ke(I) * EN + KT * Allmtd.Ef(I)
415 -----
416   疲労損傷の計算
417 -----
418 -----ファンクション ANF の計算機能の限界
419   If Allmtd.Et(I) < .0005 Then
420     Allmtd.Et(I) = .0005
421   End If
422
423 -----最高金属温度
424   Mc.Tc = Analy.Tmpmax(I)
425   Call Mset(Iflg, Mc)                               'Fortran Library(DLL) 呼び出し
426
427   EDOT = 2! * Allmtd.Et(I) / ((Cond.Hldtmc + Cond.Hldtmh) * 3600!)
428   If EDOT < .000001 Then
429     EDOT = .000001
430   End If
431   NF = Anf(Allmtd.Et(I), EDOT, Mc)                   'Fortran Library(DLL) 呼
432   Allmtd.Df(I) = Cond.Cycle / NF
433 -----
434   クリープ損傷の計算
435 -----
436 -----DCN の計算
437   DCN = 0
438 -----DCP の計算
439

```

```

440 AllmtD.Si(I) = Sepr(AllmtD.Et(I), Mc)
441 AllmtD.Si(I) = Sepr(AllmtD.Et(I), Mc) / 2!
442
443 AllmtD.Si(I) = AllmtD.Si(I) + Cond.Mecha
444
445 EC1 = 0!
446 ALPHC = 1!
447 ALPHR = 1!
448 AllmtD.Qf(I) = 3!
449 DT = Cond.Hldtmh
450 Call R1x1(AllmtD.Si(I), EC1, ALPHC, ALPHR, AllmtD.Qf(I), DT, S2X, DCX, Mc)
451 Call R1x1(AllmtD.Si(I), 0!, 1!, 1!, AllmtD.Qf(I), CSng(Cond.Hldtmh))
452 DCP = Cond.Cycle * Dcxx(0)
453
454 AllmtD.Dc(I) = DCN + DCP
455 -----
456 クリープ疲労損傷の計算
457 -----
458 AllmtD.D(I) = AllmtD.Df(I) + AllmtD.Dc(I)
459 -----
460 許容値の計算
461 -----
462 If AllmtD.Dc(I) >= 0! And AllmtD.Dc(I) <= .3 Then
463 AllmtD.Dcr(I) = 1! - 4! / 3! * AllmtD.Dc(I)
464 Elseif AllmtD.Dc(I) <= 1! Then
465 AllmtD.Dcr(I) = 3! / 7! + 4! / 7! * AllmtD.Dc(I)
466 Elseif AllmtD.Dc(I) > 1! Then
467 AllmtD.Dcr(I) = 1!
468 End If
469
470 Next I
471
472 End Sub
473
474
475 B D S
476
477 損傷計算結果データ -> Grid.Text7°ロパ°テ代入
478
479 Sub Bdsout (Icount)
480 Dim Icount As Integer
481 Dim I As Integer
482
483 For I = 0 To Icount - 1
484 Form1!Grid1.Row = I + 1
485 Form1!Grid1.Col = 1
486 Form1!Grid1.Text = Analy.R(I, 0)
487 Form1!Grid1.Col = 2
488 Form1!Grid1.Text = Analy.Z(I, 0)
489 Form1!Grid1.Col = 3
490 Form1!Grid1.Text = Analy.H(I, 0)
491 Form1!Grid1.Col = 4
492 Form1!Grid1.Text = AllmtD.Spx(I)
493 Form1!Grid1.Col = 5
494 Form1!Grid1.Text = AllmtD.Sn(I)
495 Form1!Grid1.Col = 6
496 Form1!Grid1.Text = AllmtD.K(I)
497 Form1!Grid1.Col = 7
498 Form1!Grid1.Text = AllmtD.Qn(I)
499 Form1!Grid1.Col = 8
500 Form1!Grid1.Text = AllmtD.Ked(I)
501 Form1!Grid1.Col = 9
502 Form1!Grid1.Text = AllmtD.Neuber(I)

```

```

503      Form1!Grid1.Col = 10
504      Form1!Grid1.Text = Allmtd. Ke(1)
505      Form1!Grid1.Col = 11
506      Form1!Grid1.Text = Allmtd. Ef(1)
507      Form1!Grid1.Col = 12
508      Form1!Grid1.Text = Allmtd. Et(1)
509      Form1!Grid1.Col = 13
510      Form1!Grid1.Text = Allmtd. Df(1)
511      Form1!Grid1.Col = 14
512      Form1!Grid1.Text = Allmtd. Dc(1)
513      Form1!Grid1.Col = 15
514      Form1!Grid1.Text = Allmtd. D(1)
515      Form1!Grid1.Col = 16
516      Form1!Grid1.Text = Allmtd. Dcr(1)
517      Next I
518
519 End Sub
520
521
522      B D S
523
524 Sub Bdsp (Icount)
525
526      Dim Icount As Integer
527      Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
528      Dim KEDD As Single
529      Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
530      Dim K1, K2, KE1, KE2 As Single
531      Dim K1D, K2D, KED1D, KED2D, KT As Single
532      Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
533      Dim LSIG1, LSIG2, LSIG3 As Single
534      Dim NU, NU1, NU2, NF          As Single
535      Dim INF                      As Long
536
537      Iflg = 0
538
539      For I = 0 To Icount - 1
540
541          歪範囲の計算
542
543              T I M E 1
544              Mc. Tc = Analy. Mtemp(I, 0)
545              Call Mset(Iflg, Mc)
546              E1 = Mc. E
547              NU1 = Calcnu(Mc)
548              ALPHA1 = Alpha(Mc)
549              T I M E 2
550              Mc. Tc = Analy. Mtemp(I, 1)
551              Call Mset(Iflg, Mc)
552              E2 = Mc. E
553              NU2 = Calcnu(Mc)
554              ALPHA2 = Alpha(Mc)
555
556              SF1 = (-E1 * ALPHA1 * (Analy. Temp(I, 0) - Analy. Ltemp(I, 0))) / (1 - NU1)
557              SF2 = (-E2 * ALPHA2 * (Analy. Temp(I, 1) - Analy. Ltemp(I, 1))) / (1 - NU2)
558
559              SANR = Abs(Analy. Strs(I, 0, 1) - Analy. Strs(I, 0, 0))
560              SANZ = Abs(Analy. Strs(I, 1, 1) - Analy. Strs(I, 1, 0))
561              If SANR >= SANZ Then
562                  SIGR = (Analy. Strs(I, 0, 0) - SF1) - (Analy. Strs(I, 0, 1) - SF2)
563                  SIGZ = Analy. Strs(I, 1, 0) - Analy. Strs(I, 1, 1)
564              Else
565                  SIGR = Analy. Strs(I, 0, 0) - Analy. Strs(I, 0, 1)

```

```

566     SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)
567     End If
568     SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
569     SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
570     SIG1 = Prin1(SIGR, SIGZ, SIGT)
571     SIG2 = Prin2(SIGR, SIGZ, SIGT)
572     SIG3 = SIGH
573     Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
574
575     Allmtd.Sn(I) = Bscval.Ltres(I)
576
577     最高金属温度による  $\epsilon_n$ 、S r h の計算
578     Mc.Tc = Analy.Tmpmax(I)
579     Call Mset(Iflg, Mc)
580     EH = Mc.E
581     EN = Allmtd.Sn(I) / EH
582
583     Res = Debug_disp("[Bdsp]/Spx("&Str(I)&")="&Str(Allmtd.Spx(I))&"/Sn("&Str(i)&")="&Str(Allmtd.Sn(I)))
584     ' for Debug
585     If Res = 9      Then      Exit Sub
586
587     If Allmtd.Sn(I) = 0! Then
588         KD = 0!
589     Else
590         KD = Allmtd.Spx(I) / Allmtd.Sn(I)
591     End If
592     Allmtd.K(I) = Nmax(KD, 1!, 1!)
593     Call R1x1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmh * Cond.Cycle)
594     Call R1x1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmh*Cond.Cycle, S2xx(0), Dcxx(0), Mc)
595
596     コールド温度による S r c の計算
597     Mc.Tc = Analy.Temp(I, 1)
598     Call Mset(Iflg, Mc)
599     BAR3SM = S2xx(0) + 1.5 * Mc.Sm
600     Allmtd.Qn(I) = 3!
601     Allmtd.Ked(I) = Nmax(1! + (Allmtd.Qn(I) - 1!) * (1! - BAR3SM / Allmtd.Sn(I)), 1!, 1!)
602     Allmtd.Neuber(I) = FnuBr2(Allmtd.Sn(I), Allmtd.K(I), S2xx(0), EH)
603     Allmtd.Ke(I) = Nmax(Allmtd.K(I) * Allmtd.Ked(I), Allmtd.Neuber(I), Allmtd.Neuber(I))
604
605     KT = 1!
606     200°Cにおける E0、 $\alpha_0$  の計算      1993.3.4 N. K
607     Mc.Tc = 200
608     Call Mset(Iflg, Mc)
609     E200 = Mc.E
610     ALPHA200 = Alpha(Mc)
611     修正ポアソン比の計算, 許容ひずみ範囲 0.5%固定      1993.3.4 N. K
612     NUB = .5 - (.5 - NU1) * (BAR3SM / (EH * .005))
613     熱ピーク歪みの計算      1993.3.4 N. K
614     Allmtd.Ef(I) = E200 * ALPHA200 * Abs((Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) - (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (EH * (1 - NUB))
615
616     Allmtd.Et(I) = Allmtd.Ke(I) * EN + KT * Allmtd.Ef(I)
617
618     疲労損傷の計算
619
620     ファンクション AN F の計算機能の限界
621     If Allmtd.Et(I) < .0005 Then
622         Allmtd.Et(I) = .0005
623     End If
624
625     最高金属温度
626     Mc.Tc = Analy.Tmpmax(I)
627     Call Mset(Iflg, Mc)

```

```

627
628     EDOT = 2! * Allmtd.Et(I) / ((Cond.Hldtmc + Cond.Hldtmh) * 3600!)
629     If EDOT < .000001 Then
630         EDOT = .000001
631     End If
632     ひずみ範囲2倍、繰返し回数20倍の安全率導入
633     NF = Nmin(Anf(2.0*Allmtd.Et(I), EDOT, Mc), Anf(Allmtd.Et(I), EDOT, Mc)/20.0, Anf(Allmtd.Et(I), ED
OT, Mc)/20.0)
634     ITYPE = 2
635     Call Dfcurv(Int(Analy.Tmpmax(I)), CSng(Allmtd.Et(I)), Int(Cond.Imate), Int(ITYPE), INF)
636     NF = CSng(INF)
637
638     Allmtd.Df(I) = CSng(Cond.Cycle) / NF
639
640     Res = Debug_disp("[Bdsp/Dfcurv]/INF=" & Str(INF) & "/NF=" & Str(NF) & "/Cycle=" & Str(Cond.Cycle))
for Debug
641     If Res = 9      Then      Exit Sub
642
643
644     クリープ損傷の計算
645
646     DCNの計算
647     If Not Im = 3160 Then
648         DC1 = .3
649     D*は簡易式にて計算
650     DSTRX = Dstar(Allmtd.Sn(I), Cond.Cycle * (Cond.Hldtmc + Cond.Hldtmh), DC1, Mc)
651     本来はDBAR3SMを使うべきである
652     If Allmtd.Sn(I) >= BAR3SM Then
653         DC2 = DSTRX + Cond.Cycle * DSTRX
654     Else
655         DC2 = DSTRX
656     End If
657     DCN = DC1 + DC2
658     Else
659     316FRの場合
660     DCN = 0!
661     End If
662     DCPの計算
663     If Not Im = 3160 Then
664     D**は簡易式にて計算
665     本来はMAX(BAR3SM/EH, 2SG/EH)を使うべきである
666     If Allmtd.Et(I) >= BAR3SM / EH Then
667         DDSTRX = Ddstar(Allmtd.Et(I), Cond.Cycle * (Cond.Hldtmc + Cond.Hldtmh), DC1, Mc)
668         DCP = Cond.Cycle * DDSTRX
669     Else
670         DCP = 0!
671     End If
672     Else
673     316FRの場合
674     Si = Sepr(Allmtd.Et(I), Mc)
675     Si = Sepr(Allmtd.Et(I), Mc) / 2!
676     Si = Si + Cond.Mecha
677     EC1 = 0!
678     ALPHC = 1! / 3!
679     ALPHR = 10!
680     QF = 3!
681     DT = Cond.Hldtmh
682     Call R1x1(CSng(Si), 0!, 1! / 3!, 10!, 3!, Cond.Hldtmh)
683     Call R1x1(CSng(SI), EC1, ALPHC, ALPHR, QF, DT, S2x(0), Dcx(0), Mc)
684     DCP = Cond.Cycle * Dcxx(0)
685     End If
686
687     Allmtd.Dc(I) = DCN + DCP

```



```

688
689 クリープ疲労損傷の計算
690
691 Allmtd.D(I) = Allmtd.Df(I) + Allmtd.Dc(I)
692
693 許容値の計算
694
695 If Allmtd.Dc(I) >= 0! And Allmtd.Dc(I) <= .3 Then
696 Allmtd.Dcr(I) = 1! - 4! / 3! * Allmtd.Dc(I)
697 ElseIf Allmtd.Dc(I) <= 1! Then
698 Allmtd.Dcr(I) = 3! / 7! + 4! / 7! * Allmtd.Dc(I)
699 ElseIf Allmtd.Dc(I) > 1! Then
700 Allmtd.Dcr(I) = 1!
701 End If
702
703 Next I
704
705 End Sub
706
707
708 B D S - M D
709
710 損傷計算結果データ -> Grid.Text? の入力代入
711
712 Sub Bmdout (Icount)
713 Dim Icount As Integer
714 Dim I As Integer
715
716 For I = 0 To Icount - 1
717 Form1!Grid1.Row = I + 1
718 Form1!Grid1.Col = 1
719 Form1!Grid1.Text = Analy.R(I, 0)
720 Form1!Grid1.Col = 2
721 Form1!Grid1.Text = Analy.Z(I, 0)
722 Form1!Grid1.Col = 3
723 Form1!Grid1.Text = Analy.H(I, 0)
724 Form1!Grid1.Col = 4
725 Form1!Grid1.Text = Allmtd.Spx(I)
726 Form1!Grid1.Col = 5
727 Form1!Grid1.Text = Allmtd.Sn(I)
728 Form1!Grid1.Col = 6
729 Form1!Grid1.Text = Allmtd.K(I)
730 Form1!Grid1.Col = 7
731 Form1!Grid1.Text = Allmtd.Qn(I)
732 Form1!Grid1.Col = 8
733 Form1!Grid1.Text = Allmtd.Ked(I)
734 Form1!Grid1.Col = 9
735 Form1!Grid1.Text = Allmtd.Neuber(I)
736 Form1!Grid1.Col = 10
737 Form1!Grid1.Text = Allmtd.Ke(I)
738 Form1!Grid1.Col = 11
739 Form1!Grid1.Text = Allmtd.Ef(I)
740 Form1!Grid1.Col = 12
741 Form1!Grid1.Text = Allmtd.Et(I)
742 Form1!Grid1.Col = 13
743 Form1!Grid1.Text = Allmtd.Si(I)
744 Form1!Grid1.Col = 14
745 Form1!Grid1.Text = Allmtd.QF(I)
746 Form1!Grid1.Col = 15
747 Form1!Grid1.Text = Allmtd.Df(I)
748 Form1!Grid1.Col = 16
749 Form1!Grid1.Text = Allmtd.Dc(I)
750 Form1!Grid1.Col = 17

```

```

751 Forml!Grid1.Text = Allmtd.D(I)
752 Forml!Grid1.Col = 18
753 Forml!Grid1.Text = Allmtd.Dcr(I)
754 Next I
755
756 End Sub
757
758
759 損傷計算用基本変数を算出する。
760
761 Sub Bscvl (Icount)
762 Dim Icount As Integer
763 Dim SIGR, SIGZ, SIGH, SIGT, LSIGR, LSIGZ, LSIGH, LSIGT As Single
764 Dim SIG1, SIG2, SIG3, Msigr, Msigz, Msigh, Msigt As Single
765 Dim LSIG1, LSIG2, LSIG3, Dsigr, Dsigz, Dsigh, Dsigt As Single
766 Dim Msig1, Msig2, Msig3, Dlsigr, Dlsigz, Dlsigh, Dlsigt As Single
767 Dim Dsig1, Dsig2, Dsig3, Dmsigr, Dmsigz, Dmsigh, Dmsigt As Single
768 Dim Dlsig1, Dlsig2, Dlsig3, Dmsig1, Dmsig2, Dmsig3 As Single
769 Dim Rank, Lrank, Mrank, Er, Ez, EH, Et, Epr, Epz, Eph, Ept As Single
770 Dim Ecr0, Ecz, Ech, Ect As Single
771 Dim Res As Integer
772
773 For I = 0 To Icount - 1
774
775 応力成分差 (表面応力)
776
777 SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
778 SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
779 SIGH = Analy.Strs(I, 2, 0) - Analy.Strs(I, 2, 1)
780 SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
781
782 応力成分差 (等価線形応力)
783
784 LSIGR = Analy.Lstrs(I, 0, 0) - Analy.Lstrs(I, 0, 1)
785 LSIGZ = Analy.Lstrs(I, 1, 0) - Analy.Lstrs(I, 1, 1)
786 LSIGH = Analy.Lstrs(I, 2, 0) - Analy.Lstrs(I, 2, 1)
787 LSIGT = Analy.Lstrs(I, 3, 0) - Analy.Lstrs(I, 3, 1)
788
789 応力成分差 (膜応力)
790
791 Msigr = Analy.Mstrs(I, 0, 0) - Analy.Mstrs(I, 0, 1)
792 Msigz = Analy.Mstrs(I, 1, 0) - Analy.Mstrs(I, 1, 1)
793 Msigh = Analy.Mstrs(I, 2, 0) - Analy.Mstrs(I, 2, 1)
794 Msigt = Analy.Mstrs(I, 3, 0) - Analy.Mstrs(I, 3, 1)
795
796 主応力成分差 (表面応力)
797
798 SIG1 = Prin1(SIGR, SIGZ, SIGT)
799 SIG2 = Prin2(SIGR, SIGZ, SIGT)
800 SIG3 = SIGH
801
802 主応力成分差 (等価線形応力)
803
804 LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
805 LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
806 LSIG3 = LSIGH
807
808 主応力成分差 (膜応力)
809
810 Msig1 = Prin1(Msigr, Msigz, Msigt)
811 Msig2 = Prin2(Msigr, Msigz, Msigt)
812 Msig3 = Msigh
813

```

```

814  応力成分範囲 (表面応力)
815
816      Dsigr = Abs(SIGR)
817      Dsigz = Abs(SIGZ)
818      Dsigh = Abs(SIGH)
819      Dsigt = Abs(SIGT)
820
821  応力成分範囲 (等価線形応力)
822
823      Dlsigr = Abs(LSIGR)
824      Dlsigz = Abs(LSIGZ)
825      Dlsigh = Abs(LSIGH)
826      Dlsigt = Abs(LSIGT)
827
828  応力成分範囲 (膜応力)
829
830      Dmsigr = Abs(Msigr)
831      Dmsigz = Abs(Msigz)
832      Dmsigh = Abs(Msigh)
833      Dmsigt = Abs(Msigt)
834
835  主応力成分範囲 (表面応力)
836
837      Dsig1 = Abs(SIG1)
838      Dsig2 = Abs(SIG2)
839      Dsig3 = Abs(SIG3)
840
841  主応力成分範囲 (等価線形応力)
842
843      Dlsig1 = Abs(LSIG1)
844      Dlsig2 = Abs(LSIG2)
845      Dlsig3 = Abs(LSIG3)
846
847  主応力成分範囲 (膜応力)
848
849      Dmsig1 = Abs(Msig1)
850      Dmsig2 = Abs(Msig2)
851      Dmsig3 = Abs(Msig3)
852
853  相当応力範囲 (表面応力)
854
855      Bscval.Vms(I) = Fvms(SIGR, SIGZ, SIGH, SIGT)
856      Bscval.Tres(I) = Ftres(SIG1, SIG2, SIG3)
857      Rank = Nmax(Dsig1, Dsig2, Dsig3)
858
859  相当応力範囲 (等価線形応力)
860
861      Lvms = Fvms(LSIGR, LSIGZ, LSIGH, LSIGT)
862      Bscval.Ltres(I) = Ftres(LSIG1, LSIG2, LSIG3)
863      Lrank = Nmax(Dlsig1, Dlsig2, Dlsig3)
864
865  相当応力範囲 (膜応力)
866
867      Mvms = Fvms(Msigr, Msigz, Msigh, Msigt)
868      Mtres = Ftres(Msig1, Msig2, Msig3)
869      Mrank = Nmax(Dmsig1, Dmsig2, Dmsig3)
870
871  歪成分差 (弾性歪)
872
873      Er = Analy.Ee(I, 0, 0) - Analy.Ee(I, 0, 1)
874      Ez = Analy.Ee(I, 1, 0) - Analy.Ee(I, 1, 1)
875      EH = Analy.Ee(I, 2, 0) - Analy.Ee(I, 2, 1)
876      Et = Analy.Ee(I, 3, 0) - Analy.Ee(I, 3, 1)

```

```

877 '
878 '      歪成分差 (塑性歪)
879 '
880 '      Epr = Analy.Ep(I, 0, 0) - Analy.Ep(I, 0, 1)
881 '      Epz = Analy.Ep(I, 1, 0) - Analy.Ep(I, 1, 1)
882 '      Eph = Analy.Ep(I, 2, 0) - Analy.Ep(I, 2, 1)
883 '      Ept = Analy.Ep(I, 3, 0) - Analy.Ep(I, 3, 1)
884 '
885 '      歪成分差 (クリープ歪)
886 '
887 '      Ecr0 = Analy.EC(I, 0, 0) - Analy.EC(I, 0, 1)
888 '      Ecz = Analy.EC(I, 1, 0) - Analy.EC(I, 1, 1)
889 '      Ech = Analy.EC(I, 2, 0) - Analy.EC(I, 2, 1)
890 '      Ect = Analy.EC(I, 3, 0) - Analy.EC(I, 3, 1)
891 '
892 '      相当歪範囲 (弾塑性クリープ歪範囲)
893 '
894 '      Bscval.Eepceq(I) = Eq(Er + Epr + Ecr0, Ez + Epz + Ecz, EH + Eph + Ech, Et + Ept + Ect)
895 '
896 '      相当歪範囲 (塑性歪範囲)
897 '
898 '      Bscval.Epeq(I) = Eq(Epr, Epz, Eph, Ept)
899 '
900 '      相当歪範囲 (クリープ歪範囲)
901 '
902 '      Bscval.Eceq(I) = Eq(Ecr0, Ecz, Ech, Ect)
903 '
904 '      Next I
905 '
906 ' End Sub
907 '
908 '
909 '
910 '
911 ' Sub Date_disp (Flg, I, Dstr1, Dstr2)
912 '   Dim Flg, I As Integer
913 '   Dim Dstr1, Dstr2 As String
914 '   If Cond.MethodNo <= 0 And Cond.MethodNo >= (MNO - 1) Then
915 '     If Flg = 0 Then
916 '       Mstr = Method_name(Cond.MethodNo, 0) & Dstr1
917 '     Else
918 '       Mstr = Method_name(Cond.MethodNo, 0) & Dstr1 & Dstr2
919 '     End If
920 '     Form1.Text2.Text = Mstr
921 '   End If
922 ' End Sub
923 '
924 '
925 '   D D S
926 '
927 '   損傷計算結果データ -> Grid.Text7° 08° 7i 代入
928 '
929 ' Sub Ddsout (Icount)
930 '   Dim Icount As Integer
931 '   Dim I       As Integer
932 '
933 '   For I = 0 To Icount - 1
934 '     Form1!Grid1.Row = I + 1
935 '     Form1!Grid1.Col = 1
936 '     Form1!Grid1.Text = Analy.R(I, 0)
937 '     Form1!Grid1.Col = 2
938 '     Form1!Grid1.Text = Analy.Z(I, 0)
939 '     Form1!Grid1.Col = 3

```

```

940 Forml!Gridl.Text = Analy.H(I, 0)
941 Forml!Gridl.Col = 4
942 Forml!Gridl.Text = Allmtd.Spx(I)
943 Forml!Gridl.Col = 5
944 Forml!Gridl.Text = Allmtd.Sn(I)
945 Forml!Gridl.Col = 6
946 Forml!Gridl.Text = Allmtd.K(I)
947 Forml!Gridl.Col = 7
948 Forml!Gridl.Text = Allmtd.Qn(I)
949 Forml!Gridl.Col = 8
950 Forml!Gridl.Text = Allmtd.Ked(I)
951 Forml!Gridl.Col = 9
952 Forml!Gridl.Text = Allmtd.Neuber(I)
953 Forml!Gridl.Col = 10
954 Forml!Gridl.Text = Allmtd.Ke(I)
955 Forml!Gridl.Col = 11
956 Forml!Gridl.Text = Allmtd.Ef(I)
957 Forml!Gridl.Col = 12
958 Forml!Gridl.Text = Allmtd.Et(I)
959 Forml!Gridl.Col = 13
960 Forml!Gridl.Text = Allmtd.Si(I)
961 Forml!Gridl.Col = 14
962 Forml!Gridl.Text = Allmtd.QF(I)
963 Forml!Gridl.Col = 15
964 Forml!Gridl.Text = Allmtd.Df(I)
965 Forml!Gridl.Col = 16
966 Forml!Gridl.Text = Allmtd.Dc(I)
967 Forml!Gridl.Col = 17
968 Forml!Gridl.Text = Allmtd.D(I)
969 Next I
970
971 End Sub
972
973
974 DDS
975
976 Sub Ddsp (Icount)
977
978 Dim Icount As Integer
979 Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
980 Dim KEDD As Single
981 Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
982 Dim K1, K2, KE1, KE2 As Single
983 Dim K1D, K2D, KED1D, KED2D, KT As Single
984 Dim KED1, KED2 As Single
985 Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
986 Dim LSIG1, LSIG2, LSIG3 As Single
987 Dim NU, NU1, NU2, NF As Single
988 Dim NUBER1, NUBER2 As Single
989 Dim INF As Long
990
991 Iflg = 0
992
993 For I = 0 To Icount - 1
994
995 歪範囲の計算
996
997 TIME 1
998 Mc.Tc = Analy.Mtemp(I, 0)
999 Call Mset(Iflg, Mc)
1000 E1 = Mc.E
1001 NU1 = Calcnu(Mc)
1002 ALPHA1 = Alpha(Mc)

```

```

1003   T I M E 2
1004   Mc.Tc = Analy.Mtemp(I, 1)
1005   Call Mset(Iflg, Mc)
1006   E2 = Mc.E
1007   NU2 = Calcnu(Mc)
1008   ALPHA2 = Alpha(Mc)
1009
1010   SF1 = (-E1 * ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0))) / (1 - NU1)
1011   SF2 = (-E2 * ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2)
1012   Allmtd.Ef(I) = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *
(Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
1013
1014   SANR = Abs(Analy.Strs(I, 0, 1) - Analy.Strs(I, 0, 0))
1015   SANZ = Abs(Analy.Strs(I, 1, 1) - Analy.Strs(I, 1, 0))
1016   If SANR >= SANZ Then
1017     SIGR = (Analy.Strs(I, 0, 0) - SF1) - (Analy.Strs(I, 0, 1) - SF2)
1018     SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
1019   Else
1020     SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
1021     SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)
1022   End If
1023   SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
1024   SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
1025   SIG1 = Prin1(SIGR, SIGZ, SIGT)
1026   SIG2 = Prin2(SIGR, SIGZ, SIGT)
1027   SIG3 = SIGH
1028   Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
1029
1030   Allmtd.Sn(I) = Bscval.Ltres(I)
1031
1032   最高金属温度による  $\epsilon_n$ 、S r h の計算
1033   Mc.Tc = Analy.Tmpmax(I)
1034   Call Mset(Iflg, Mc)
1035   EN = Allmtd.Sn(I) / Mc.E
1036
1037   If Allmtd.Sn(I) = 0! Then
1038     KD = 0!
1039   Else
1040     KD = Allmtd.Spx(I) / Allmtd.Sn(I)
1041   End If
1042   Allmtd.K(I) = Smax(KD, 1!, 1!)
1043   Call R1x1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmh)
1044   Call R1x1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmh, S2xx(0), Dcxx(0), Mc)
1045
1046   コールド温度による S r c の計算
1047   Mc.Tc = Analy.Temp(I, 1)
1048   Call Mset(Iflg, Mc)
1049   BAR3SM = S2xx(0) + 1.5 * Mc.Sm
1050   Allmtd.Qn(I) = Analy.Q(I)
1051   If Allmtd.Sn(I) = 0! Then
1052     KEDD = 0!
1053   Else
1054     KEDD = BAR3SM / Allmtd.Sn(I)
1055   End if
1056   Allmtd.Ked(I) = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KEDD), 1!, 1!)
1057   Allmtd.Neuber(I) = Fneuber(Allmtd.Sn(I), Allmtd.K(I), BAR3SM)
1058   Allmtd.Ke(I) = Allmtd.Ked(I) * Allmtd.Neuber(I)
1059
1060   KT = 1.4
1061
1062   Allmtd.Et(I) = Allmtd.Ke(I) * EN + KT * Allmtd.Ef(I)
1063
1064   疲労損傷の計算

```

```

1065 .
1066 . ファンクションANFの計算機能の限界
1067 .   If Allmtd.Et(I) < .0005 Then Allmtd.Et(I) = .0005
1068 .
1069 . 最高金属温度
1070 .   Mc.Tc = Analy.Tmpmax(I)
1071 .   Call Mset(Iflg, Mc)
1072 .
1073 .   EDOT = .001
1074 .   NF = Anf(Allmtd.Et(I), EDOT, Mc)
1075 .
1076 .   Allmtd.Df(I) = Cond.Cycle / NF
1077 .
1078 . クリープ損傷の計算
1079 .
1080 . DCNの計算
1081 .   DCN = 0
1082 . DCPの計算
1083 . 初期応力の計算
1084 . TIME 1
1085 .   EF1 = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1))
1086 .   If Analy.Strs(I, 0, 0) >= Analy.Strs(I, 1, 0) Then
1087 .     SIGR = Analy.Strs(I, 0, 0) - SF1
1088 .     SIGZ = Analy.Strs(I, 1, 0)
1089 .   Else
1090 .     SIGR = Analy.Strs(I, 0, 0)
1091 .     SIGZ = Analy.Strs(I, 1, 0) - SF1
1092 .   End If
1093 .   SIGH = Analy.Strs(I, 2, 0) - SF1
1094 .   SIGT = Analy.Strs(I, 3, 0)
1095 .   SIG1 = Prin1(SIGR, SIGZ, SIGT)
1096 .   SIG2 = Prin2(SIGR, SIGZ, SIGT)
1097 .   SIG3 = SIGH
1098 .   SP1 = Ftres(SIG1, SIG2, SIG3)
1099 .
1100 .   If Analy.Lstrs(I, 0, 0) >= Analy.Lstrs(I, 1, 0) Then
1101 .     LSIGR = Analy.Lstrs(I, 0, 0) - SF1
1102 .     LSIGZ = Analy.Lstrs(I, 1, 0)
1103 .   Else
1104 .     LSIGR = Analy.Lstrs(I, 0, 0)
1105 .     LSIGZ = Analy.Lstrs(I, 1, 0) - SF1
1106 .   End If
1107 .   LSIGH = Analy.Lstrs(I, 2, 0) - SF1
1108 .   LSIGT = Analy.Lstrs(I, 3, 0)
1109 .   LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
1110 .   LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
1111 .   LSIG3 = LSIGH
1112 .   SN1 = Ftres(LSIG1, LSIG2, LSIG3)
1113 .   EN1 = SN1 / EI
1114 .   If SN1 = 0! Then
1115 .     K1D = 0!
1116 .   Else
1117 .     K1D = SP1 / SN1
1118 .   End If
1119 .   K1 = Smax(K1D, 1!, 1!)
1120 .   If SN1 = 0! Then
1121 .     KED1D = 0!
1122 .   Else
1123 .     KED1D = BAR3SM / SN1
1124 .   End If
1125 .   KED1 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED1D), 1!, 1!)
1126 .   NUBER1 = Fnumber(SN1, K1, BAR3SM)
1127 .   KE1 = KED1 * NUBER1

```

```

1128     ETD1 = KE1 * EN1 + KT * EF1
1129     TIME 2
1130     EF2 = Abs(ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
1131     If Analy.Strs(I, 0, 1) >= Analy.Strs(I, 1, 1) Then
1132         SIGR = Analy.Strs(I, 0, 1) - SF2
1133         SIGZ = Analy.Strs(I, 1, 1)
1134     Else
1135         SIGR = Analy.Strs(I, 0, 1)
1136         SIGZ = Analy.Strs(I, 1, 1) - SF2
1137     End If
1138     SIGH = Analy.Strs(I, 2, 1) - SF2
1139     SIGT = Analy.Strs(I, 3, 1)
1140     SIG1 = Prin1(SIGR, SIGZ, SIGT)
1141     SIG2 = Prin2(SIGR, SIGZ, SIGT)
1142     SIG3 = SIGH
1143     SP2 = Ftres(SIG1, SIG2, SIG3)
1144
1145     If Analy.Lstrs(I, 0, 1) >= Analy.Lstrs(I, 1, 1) Then
1146         LSIGR = Analy.Lstrs(I, 0, 1) - SF2
1147         LSIGZ = Analy.Lstrs(I, 1, 1)
1148     Else
1149         LSIGR = Analy.Lstrs(I, 0, 1)
1150         LSIGZ = Analy.Lstrs(I, 1, 1) - SF2
1151     End If
1152     LSIGH = Analy.Lstrs(I, 2, 1) - SF2
1153     LSIGT = Analy.Lstrs(I, 3, 1)
1154     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
1155     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
1156     LSIG3 = LSIGH
1157     SN2 = Ftres(LSIG1, LSIG2, LSIG3)
1158     EN2 = SN2 / E2
1159     If SN2 = 0! Then
1160         K2D = 0!
1161     Else
1162         K2D = SP2 / SN2
1163     End If
1164     K2 = Smax(K2D, 1!, 1!)
1165     If SN2 = 0! Then
1166         KED2D = 0!
1167     Else
1168         KED2D = BAR3SM / SN2
1169     End If
1170     KED2 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED2D), 1!, 1!)
1171     NUBER2 = Fnumber(SN2, K2, BAR3SM)
1172     KE2 = KED2 * NUBER2
1173     ETD2 = KE2 * EN2 + KT * EF2
1174
1175     ETC = Smax(ETD1, ETD1, ETD2)
1176     ETC = Smax(ETD1, ETD2, ETD2)
1177
1178     Allmtd.Si(I) = Sepr(ETC, Mc)
1179     Allmtd.Si(I) = Sepr(ETC, Mc) / 2!
1180
1181     Allmtd.Si(I) = Allmtd.Si(I) + Cond.Mecha
1182
1183     EC1 = 0!
1184     ALPHC = 1!
1185     ALPHR = 1!
1186     Allmtd.QF(I) = 1.5
1187     DT = Cond.Hldtmh
1188     Call Rlx1(Allmtd.Si(I), 0!, 1!, 1!, Allmtd.QF(I), Cond.Hldtmh)
1189     Call Rlx1(Allmtd.Si(I), EC1, ALPHC, ALPHR, Allmtd.Qf(I), DT, S2xx(0), Dcxx(0), Mc)
1190     DCP = Cond.Cycle * Dcxx(0)

```



```

1191
1192      Allmtd. Dc(I) = DCN + DCP
1193
1194      クリープ疲労損傷の計算
1195
1196      Allmtd. D(I) = Allmtd. Df(I) + Allmtd. Dc(I)
1197
1198      Next I
1199
1200 End Sub
1201
1202
1203      D D S M D - W E L D
1204
1205 Sub Ddswldp (Icount)
1206
1207      Dim Icount As Integer
1208      Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
1209      Dim KEDD As Single
1210      Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
1211      Dim K1, K2, KE1, KE2 As Single
1212      Dim K1D, K2D, KED1D, KED2D, KT As Single
1213      Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
1214      Dim LSIG1, LSIG2, LSIG3 As Single
1215      Dim NU, NU1, NU2, NF          As Single
1216      Dim NUBER1, NUBER2          As Single
1217      Dim INF                      As Long
1218
1219      Iflg = 0
1220
1221      For I = 0 To Icount - 1
1222
1223          歪範囲の計算
1224
1225          T I M E 1
1226          Mc. Tc = Analy. Mtemp(I, 0)
1227          Call Mset(Iflg, Mc)
1228          E1 = Mc. E
1229          NU1 = Calcnu(Mc)
1230          ALPHA1 = Alpha(Mc)
1231          T I M E 2
1232          Mc. Tc = Analy. Mtemp(I, 1)
1233          Call Mset(Iflg, Mc)
1234          E2 = Mc. E
1235          NU2 = Calcnu(Mc)
1236          ALPHA2 = Alpha(Mc)
1237
1238          SF1 = (-E1 * ALPHA1 * (Analy. Temp(I, 0) - Analy. Ltemp(I, 0))) / (1 - NU1)
1239          SF2 = (-E2 * ALPHA2 * (Analy. Temp(I, 1) - Analy. Ltemp(I, 1))) / (1 - NU2)
1240          Allmtd. Ef(I) = Abs(ALPHA1 * (Analy. Temp(I, 0) - Analy. Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *
            (Analy. Temp(I, 1) - Analy. Ltemp(I, 1)) / (1 - NU2))
1241
1242          SANR = Abs(Analy. Strs(I, 0, 1) - Analy. Strs(I, 0, 0))
1243          SANZ = Abs(Analy. Strs(I, 1, 1) - Analy. Strs(I, 1, 0))
1244          If SANR >= SANZ Then
1245              SIGR = (Analy. Strs(I, 0, 0) - SF1) - (Analy. Strs(I, 0, 1) - SF2)
1246              SIGZ = Analy. Strs(I, 1, 0) - Analy. Strs(I, 1, 1)
1247          Else
1248              SIGR = Analy. Strs(I, 0, 0) - Analy. Strs(I, 0, 1)
1249              SIGZ = (Analy. Strs(I, 1, 0) - SF1) - (Analy. Strs(I, 1, 1) - SF2)
1250          End If
1251          SIGH = (Analy. Strs(I, 2, 0) - SF1) - (Analy. Strs(I, 2, 1) - SF2)
1252          SIGT = Analy. Strs(I, 3, 0) - Analy. Strs(I, 3, 1)

```

```

1253     SIG1 = Prin1(SIGR, SIGZ, SIGT)
1254     SIG2 = Prin2(SIGR, SIGZ, SIGT)
1255     SIG3 = SIGH
1256     Allmtd. Spx(I) = Ftres(SIG1, SIG2, SIG3)
1257
1258     Allmtd. Sn(I) = Bscval. Ltres(I)
1259
1260     最高金属温度による  $\epsilon n$ 、S r h の計算
1261     Mc. Tc = Analy. Tmpmax(I)
1262     Call Mset(Iflg, Mc)
1263     EN = Allmtd. Sn(I) / Mc. E
1264
1265     If Allmtd. Sn(I) = 0! Then
1266         KD = 0!
1267     Else
1268         KD = Allmtd. Spx(I) / Allmtd. Sn(I)
1269     End If
1270     Allmtd. K(I) = Smax(KD, 1!, 1!)
1271     Call Rlx1(1.5 * Mc. Sm, 0!, 1!, 1!, 1!, Cond. Hldtmh)
1272     Call Rlx1(1.5*Mc. Sm, 0.0, 1.0, 1.0, 1.0, Cond. Hldtmh, S2xx(0), Dcxx(0), Mc)
1273
1274     コールド温度による S r c の計算
1275     Mc. Tc = Analy. Temp(I, 1)
1276     Call Mset(Iflg, Mc)
1277     BAR3SM = S2xx(0) + 1.5 * Mc. Sm
1278     FW = 1.5
1279     Allmtd. Qbm(I) = Analy. Q(I)
1280     Allmtd. Qw(I) = FW * Allmtd. Qbm(I)
1281
1282     Res = Debug_disp("[Ddswldp]/Q=" & Str(Analy. Q(I)) & "/Qw=" & Str(Allmtd. Qw(I))) ' for Debug
1283     If Res = 9 Then Exit Sub
1284
1285     GAMWAY = .8
1286     KEO = 1.3
1287     If Allmtd. Sn(I) = 0! Then
1288         KEDD = 0!
1289     Else
1290         KEDD = GAMWAY * BAR3SM / Allmtd. Sn(I)
1291     End If
1292     Allmtd. Ked(I) = Smax(1! + (Allmtd. Qw(I) - 1!) * (1! - KEDD), 1!, 1!)
1293     Allmtd. Neuber(I) = Fnuber(Allmtd. Sn(I), Allmtd. K(I), BAR3SM)
1294     Allmtd. Ke(I) = Smax(Allmtd. Ked(I) * Allmtd. Neuber(I), KEO, KEO)
1295
1296     KT = 1.4
1297
1298     Allmtd. Et(I) = Allmtd. Ke(I) * EN + KT * Allmtd. Ef(I)
1299
1300     疲労損傷の計算
1301
1302     ファンクション ANF の計算機能の限界
1303     If Allmtd. Et(I) < .0005 Then Allmtd. Et(I) = .0005
1304
1305     最高金属温度
1306     Mc. Tc = Analy. Tmpmax(I)
1307     Call Mset(Iflg, Mc)
1308
1309     EDOT = 2! * Allmtd. Et(I) / ((Cond. Hldtmc + Cond. Hldtmh) * 3600!)
1310     NF = Anf(Allmtd. Et(I), EDOT, Mc)
1311
1312     Allmtd. Df(I) = Cond. Cycle / NF
1313
1314     クリープ損傷の計算
1315

```

```

1316   DCNの計算
1317   DCN = 0
1318   DCPの計算
1319   初期応力の計算
1320   TIME 1
1321   EF1 = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1))
1322   If Analy.Strs(I, 0, 0) >= Analy.Strs(I, 1, 0) Then
1323     SIGR = Analy.Strs(I, 0, 0) - SF1
1324     SIGZ = Analy.Strs(I, 1, 0)
1325   Else
1326     SIGR = Analy.Strs(I, 0, 0)
1327     SIGZ = Analy.Strs(I, 1, 0) - SF1
1328   End If
1329   SIGH = Analy.Strs(I, 2, 0) - SF1
1330   SIGT = Analy.Strs(I, 3, 0)
1331   SIG1 = Prin1(SIGR, SIGZ, SIGT)
1332   SIG2 = Prin2(SIGR, SIGZ, SIGT)
1333   SIG3 = SIGH
1334   SP1 = Ftres(SIG1, SIG2, SIG3)
1335
1336   If Analy.Lstrs(I, 0, 0) >= Analy.Lstrs(I, 1, 0) Then
1337     LSIGR = Analy.Lstrs(I, 0, 0) - SF1
1338     LSIGZ = Analy.Lstrs(I, 1, 0)
1339   Else
1340     LSIGR = Analy.Lstrs(I, 0, 0)
1341     LSIGZ = Analy.Lstrs(I, 1, 0) - SF1
1342   End If
1343   LSIGH = Analy.Lstrs(I, 2, 0) - SF1
1344   LSIGT = Analy.Lstrs(I, 3, 0)
1345   LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
1346   LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
1347   LSIG3 = LSIGH
1348   SN1 = Ftres(LSIG1, LSIG2, LSIG3)
1349   EN1 = SN1 / E1
1350   If SN1 = 0! Then
1351     K1D = 0!
1352   Else
1353     K1D = SP1 / SN1
1354   End If
1355   K1 = Smax(K1D, 1!, 1!)
1356   If SN1 = 0! Then
1357     KED1D = 0!
1358   Else
1359     KED1D = GAMMAY * BAR3SM / SN1
1360   End If
1361   KED1 = Smax(1! + (Allmtd.Qw(1) - 1!) * (1! - KED1D), 1!, 1!)
1362   NUBER1 = Fnumber(SN1, K1, BAR3SM)
1363   KE1 = KED1 * NUBER1
1364   ETD1 = KE1 * EN1 + KT * EF1
1365   TIME 2
1366   EF2 = Abs(ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
1367   If Analy.Strs(I, 0, 1) >= Analy.Strs(I, 1, 1) Then
1368     SIGR = Analy.Strs(I, 0, 1) - SF2
1369     SIGZ = Analy.Strs(I, 1, 1)
1370   Else
1371     SIGR = Analy.Strs(I, 0, 1)
1372     SIGZ = Analy.Strs(I, 1, 1) - SF2
1373   End If
1374   SIGH = Analy.Strs(I, 2, 1) - SF2
1375   SIGT = Analy.Strs(I, 3, 1)
1376   SIG1 = Prin1(SIGR, SIGZ, SIGT)
1377   SIG2 = Prin2(SIGR, SIGZ, SIGT)
1378   SIG3 = SIGH

```

```

1379      SP2 = Ftres(SIG1, SIG2, SIG3)
1380
1381      If Analy.Lstrs(I, 0, 1) >= Analy.Lstrs(I, 1, 1) Then
1382          LSIGR = Analy.Lstrs(I, 0, 1) - SF2
1383          LSIGZ = Analy.Lstrs(I, 1, 1)
1384      Else
1385          LSIGR = Analy.Lstrs(I, 0, 1)
1386          LSIGZ = Analy.Lstrs(I, 1, 1) - SF2
1387      End If
1388      LSIGH = Analy.Lstrs(I, 2, 1) - SF2
1389      LSIGT = Analy.Lstrs(I, 3, 1)
1390      LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
1391      LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
1392      LSIG3 = LSIGH
1393      SN2 = Ftres(LSIG1, LSIG2, LSIG3)
1394      EN2 = SN2 / E2
1395      If SN2 = 0! Then
1396          K2D = 0!
1397      Else
1398          K2D = SP2 / SN2
1399      End If
1400      K2 = Smax(K2D, 1!, 1!)
1401      If SN2 = 0! Then
1402          KED2D = 0!
1403      Else
1404          KED2D = GAMMAY * BAR3SM / SN2
1405      End If
1406      KED2 = Smax(1! + (Allmtd.Qw(I) - 1!) * (1! - KED2D), 1!, 1!)
1407      NUBER2 = Fnumber(SN2, K2, BAR3SM)
1408      KE2 = KED2 * NUBER2
1409      ETD2 = KE2 * EN2 + KT * EF2
1410
1411      ETC = Smax(ETD1, ETD2, ETD2)
1412
1413      Allmtd.Si(I) = Sepr(ETC, Mc)
1414      Allmtd.Si(I) = Sepr(ETC, Mc) / 2!
1415
1416      Allmtd.Si(I) = Allmtd.Si(I) + Cond.Mecha
1417
1418      EC1 = 0!
1419      ALPHC = 1!
1420      ALPHR = 1!
1421      QF = 1.5
1422      FW = 1.5
1423      Allmtd.Qeff(I) = FW * QF
1424      DT = Cond.Hldtmh
1425      Call R1x1(Allmtd.Si(I), 0!, 1!, 1!, Allmtd.Qeff(I), Cond.Hldtmh)
1426      Call R1x1(Allmtd.Si(I), EC1, ALPHC, ALPHR, Allmtd.Qeff(I), DT, S2xx(0), Dcxx(0), Mc)
1427      DCP = 2! * Cond.Cycle * Dcxx(0)
1428
1429      Allmtd.Dc(I) = DCN + DCP
1430
1431      クリープ疲労損傷の計算
1432
1433      Allmtd.D(I) = Allmtd.Df(I) + Allmtd.Dc(I)
1434
1435      Next I
1436
1437 End Sub
1438
1439
1440      処理状況表示ルーチン
1441

```

```

1442 Function Debug_disp (Msg)
1443     Const MB_OK = 0, MB_OKCANCEL = 1
1444     Const MB_ICONINFORMATION = 64, MB_DEFBUTTON1 = 0
1445     Const IDCANCEL = 2, IDOK = 1
1446     Dim Dgdef, Response As Integer
1447     Dim Dmsg, Title As String
1448     Dgdef = MB_OKCANCEL + MB_ICONINFORMATION + MB_DEFBUTTON1
1449     Title = "処理状況"
1450     Response = MsgBox(Msg, Dgdef, Title)
1451     If Response = IDCANCEL Then
1452         Debug_disp = 9
1453     Else
1454         Debug_disp = 0
1455     End If
1456 End Function
1457
1458
1459     DATA SET DFCURV      AT LEVEL 001 AS OF 03/12/85
1460     DATA SET DFCURV02   AT LEVEL 001 AS OF 08/31/84
1461     DATA SET T5413724   AT LEVEL 002 AS OF 10/03/80
1462     DATA SET T5413724   AT LEVEL 001 AS OF 09/08/80
1463     DATA SET T5413724   AT LEVEL 001 AS OF 12/24/79
1464     DATA SET T8251711   AT LEVEL 009 AS OF 11/15/79
1465
1466 Sub Dfcurv (Temp#, STN!, MATE#, ITYPE#, ND&)
1467     TEMP, MATE, ITYPE   As Integer
1468     STN                  As Single
1469     ND                   As Long
1470
1471     * 設計疲労曲線に歪範囲を適用して許容繰り返し回数N -を求める
1472
1473 Static N(15)                As Long
1474 Static T(9), S(15, 9)       As Single
1475 Static T1(9), T2(7), S1(15, 4), S2(15, 4), S3(15, 4)   As Single
1476 Static S11(15, 4), S22(15, 4), S33(15, 4) As Single
1477 Static S4(15), S5(15)       As Single
1478 Static S6(15, 4), S7(15, 4), S8(15, 4) As Single
1479 Static S66(15, 4), S77(15, 4), S88(15, 4) As Single
1480 Static SA(15, 3), SB(15, 3), SC(15, 3) As Single
1481 Static SAA(15, 3), SBB(15, 3), SCC(15, 3) As Single
1482 Static SD(15), SE(15)      As Single
1483 Dim FN1, FN2, FD, X1, X2, Y1, Y2   As Single
1484
1485     N(0) = 10
1486     N(1) = 20
1487     N(2) = 40
1488     N(3) = 100
1489     N(4) = 200
1490     N(5) = 400
1491     N(6) = 1000
1492     N(7) = 2000
1493     N(8) = 4000
1494     N(9) = 10000
1495     N(10) = 20000
1496     N(11) = 40000
1497     N(12) = 100000
1498     N(13) = 200000
1499     N(14) = 400000
1500     N(15) = 1000000
1501
1502     T1(0) = 425!
1503     T1(1) = 450!
1504     T1(2) = 475!

```

1505 T1(3) = 500!  
 1506 T1(4) = 525!  
 1507 T1(5) = 550!  
 1508 T1(6) = 575!  
 1509 T1(7) = 600!  
 1510 T1(8) = 625!  
 1511 T1(9) = 650!  
 1512  
 1513 T2(0) = 375!  
 1514 T2(1) = 400!  
 1515 T2(2) = 425!  
 1516 T2(3) = 450!  
 1517 T2(4) = 475!  
 1518 T2(5) = 500!  
 1519 T2(6) = 525!  
 1520 T2(7) = 550!  
 1521  
 1522  
 1523 +++ SUS304, SUS321 +++ A, B, C  
 1524  
 1525 S1(0, 0) = .05  
 1526 S1(1, 0) = .0362  
 1527 S1(2, 0) = .0268  
 1528 S1(3, 0) = .0185  
 1529 S1(4, 0) = .0142  
 1530 S1(5, 0) = .0113  
 1531 S1(6, 0) = .00838  
 1532 S1(7, 0) = .00685  
 1533 S1(8, 0) = .00571  
 1534 S1(9, 0) = .00454  
 1535 S1(10, 0) = .00392  
 1536 S1(11, 0) = .00342  
 1537 S1(12, 0) = .00288  
 1538 S1(13, 0) = .00254  
 1539 S1(14, 0) = .00227  
 1540 S1(15, 0) = .002  
 1541  
 1542 S1(0, 1) = .0331  
 1543 S1(1, 1) = .0238  
 1544 S1(2, 1) = .0173  
 1545 S1(3, 1) = .0116  
 1546 S1(4, 1) = .00891  
 1547 S1(5, 1) = .00705  
 1548 S1(6, 1) = .00544  
 1549 S1(7, 1) = .00462  
 1550 S1(8, 1) = .00394  
 1551 S1(9, 1) = .00318  
 1552 S1(10, 1) = .00271  
 1553 S1(11, 1) = .00231  
 1554 S1(12, 1) = .00193  
 1555 S1(13, 1) = .00172  
 1556 S1(14, 1) = .00156  
 1557 S1(15, 1) = .0014  
 1558  
 1559 S1(0, 2) = .0323  
 1560 S1(1, 2) = .0232  
 1561 S1(2, 2) = .0168  
 1562 S1(3, 2) = .0113  
 1563 S1(4, 2) = .00864  
 1564 S1(5, 2) = .00684  
 1565 S1(6, 2) = .00529  
 1566 S1(7, 2) = .0045  
 1567 S1(8, 2) = .00383

1568	$S1(9, 2) = .0031$
1569	$S1(10, 2) = .00264$
1570	$S1(11, 2) = .00225$
1571	$S1(12, 2) = .00189$
1572	$S1(13, 2) = .00169$
1573	$S1(14, 2) = .00153$
1574	$S1(15, 2) = .00137$
1575	
1576	$S1(0, 3) = .0316$
1577	$S1(1, 3) = .0225$
1578	$S1(2, 3) = .0163$
1579	$S1(3, 3) = .0109$
1580	$S1(4, 3) = .00836$
1581	$S1(5, 3) = .00662$
1582	$S1(6, 3) = .00513$
1583	$S1(7, 3) = .00437$
1584	$S1(8, 3) = .00372$
1585	$S1(9, 3) = .00301$
1586	$S1(10, 3) = .00257$
1587	$S1(11, 3) = .00219$
1588	$S1(12, 3) = .00184$
1589	$S1(13, 3) = .00165$
1590	$S1(14, 3) = .0015$
1591	$S1(15, 3) = .00134$
1592	
1593	$S1(0, 4) = .0308$
1594	$S1(1, 4) = .0219$
1595	$S1(2, 4) = .0158$
1596	$S1(3, 4) = .0105$
1597	$S1(4, 4) = .00807$
1598	$S1(5, 4) = .00641$
1599	$S1(6, 4) = .00498$
1600	$S1(7, 4) = .00424$
1601	$S1(8, 4) = .00361$
1602	$S1(9, 4) = .00292$
1603	$S1(10, 4) = .00249$
1604	$S1(11, 4) = .00213$
1605	$S1(12, 4) = .00179$
1606	$S1(13, 4) = .00161$
1607	$S1(14, 4) = .00146$
1608	$S1(15, 4) = .00131$
1609	
1610	
1611	$S11(0, 0) = .03$
1612	$S11(1, 0) = .0212$
1613	$S11(2, 0) = .0152$
1614	$S11(3, 0) = .0102$
1615	$S11(4, 0) = .00779$
1616	$S11(5, 0) = .00619$
1617	$S11(6, 0) = .00482$
1618	$S11(7, 0) = .00411$
1619	$S11(8, 0) = .0035$
1620	$S11(9, 0) = .00283$
1621	$S11(10, 0) = .00241$
1622	$S11(11, 0) = .00207$
1623	$S11(12, 0) = .00174$
1624	$S11(13, 0) = .00157$
1625	$S11(14, 0) = .00143$
1626	$S11(15, 0) = .00128$
1627	
1628	$S11(0, 1) = .0291$
1629	$S11(1, 1) = .0205$
1630	$S11(2, 1) = .0147$

1631	S11(3, 1) = .00978
1632	S11(4, 1) = .0075
1633	S11(5, 1) = .00597
1634	S11(6, 1) = .00467
1635	S11(7, 1) = .00398
1636	S11(8, 1) = .00339
1637	S11(9, 1) = .00274
1638	S11(10, 1) = .00233
1639	S11(11, 1) = .002
1640	S11(12, 1) = .0017
1641	S11(13, 1) = .00153
1642	S11(14, 1) = .00139
1643	S11(15, 1) = .00125
1644	
1645	S11(0, 2) = .0283
1646	S11(1, 2) = .0198
1647	S11(2, 2) = .0141
1648	S11(3, 2) = .0094
1649	S11(4, 2) = .00721
1650	S11(5, 2) = .00576
1651	S11(6, 2) = .00452
1652	S11(7, 2) = .00385
1653	S11(8, 2) = .00328
1654	S11(9, 2) = .00265
1655	S11(10, 2) = .00226
1656	S11(11, 2) = .00194
1657	S11(12, 2) = .00165
1658	S11(13, 2) = .00149
1659	S11(14, 2) = .00136
1660	S11(15, 2) = .00122
1661	
1662	S11(0, 3) = .0274
1663	S11(1, 3) = .0191
1664	S11(2, 3) = .0136
1665	S11(3, 3) = .00902
1666	S11(4, 3) = .00693
1667	S11(5, 3) = .00554
1668	S11(6, 3) = .00436
1669	S11(7, 3) = .00372
1670	S11(8, 3) = .00317
1671	S11(9, 3) = .00256
1672	S11(10, 3) = .00218
1673	S11(11, 3) = .00188
1674	S11(12, 3) = .0016
1675	S11(13, 3) = .00144
1676	S11(14, 3) = .00132
1677	S11(15, 3) = .00119
1678	
1679	S11(0, 4) = .0265
1680	S11(1, 4) = .0184
1681	S11(2, 4) = .013
1682	S11(3, 4) = .00864
1683	S11(4, 4) = .00665
1684	S11(5, 4) = .00533
1685	S11(6, 4) = .00421
1686	S11(7, 4) = .00359
1687	S11(8, 4) = .00306
1688	S11(9, 4) = .00247
1689	S11(10, 4) = .00211
1690	S11(11, 4) = .00182
1691	S11(12, 4) = .00155
1692	S11(13, 4) = .00141
1693	S11(14, 4) = .00129



1694 S11(15, 4) = .00116  
1695  
1696  
1697 S2(0, 0) = .05  
1698 S2(1, 0) = .0362  
1699 S2(2, 0) = .0268  
1700 S2(3, 0) = .0185  
1701 S2(4, 0) = .0142  
1702 S2(5, 0) = .0113  
1703 S2(6, 0) = .00838  
1704 S2(7, 0) = .00685  
1705 S2(8, 0) = .00571  
1706 S2(9, 0) = .00454  
1707 S2(10, 0) = .00392  
1708 S2(11, 0) = .00342  
1709 S2(12, 0) = .00288  
1710 S2(13, 0) = .00254  
1711 S2(14, 0) = .00227  
1712 S2(15, 0) = .002  
1713  
1714 S2(0, 1) = .0265  
1715 S2(1, 1) = .0185  
1716 S2(2, 1) = .0131  
1717 S2(3, 1) = .00875  
1718 S2(4, 1) = .00676  
1719 S2(5, 1) = .00544  
1720 S2(6, 1) = .00431  
1721 S2(7, 1) = .00367  
1722 S2(8, 1) = .00313  
1723 S2(9, 1) = .00253  
1724 S2(10, 1) = .00216  
1725 S2(11, 1) = .00187  
1726 S2(12, 1) = .0016  
1727 S2(13, 1) = .00145  
1728 S2(14, 1) = .00133  
1729 S2(15, 1) = .0012  
1730  
1731 S2(0, 2) = .0252  
1732 S2(1, 2) = .0174  
1733 S2(2, 2) = .0123  
1734 S2(3, 2) = .00819  
1735 S2(4, 2) = .00635  
1736 S2(5, 2) = .00513  
1737 S2(6, 2) = .00409  
1738 S2(7, 2) = .00348  
1739 S2(8, 2) = .00297  
1740 S2(9, 2) = .0024  
1741 S2(10, 2) = .00204  
1742 S2(11, 2) = .00178  
1743 S2(12, 2) = .00153  
1744 S2(13, 2) = .00138  
1745 S2(14, 2) = .00127  
1746 S2(15, 2) = .00115  
1747  
1748 S2(0, 3) = .0238  
1749 S2(1, 3) = .0163  
1750 S2(2, 3) = .0115  
1751 S2(3, 3) = .00765  
1752 S2(4, 3) = .00595  
1753 S2(5, 3) = .00483  
1754 S2(6, 3) = .00387  
1755 S2(7, 3) = .0033  
1756 S2(8, 3) = .00281

1757	S2(9, 3) = .00227
1758	S2(10, 3) = .00193
1759	S2(11, 3) = .00169
1760	S2(12, 3) = .00146
1761	S2(13, 3) = .00132
1762	S2(14, 3) = .00122
1763	S2(15, 3) = .00111
1764	
1765	S2(0, 4) = .0224
1766	S2(1, 4) = .0152
1767	S2(2, 4) = .0107
1768	S2(3, 4) = .00712
1769	S2(4, 4) = .00556
1770	S2(5, 4) = .00454
1771	S2(6, 4) = .00366
1772	S2(7, 4) = .00311
1773	S2(8, 4) = .00265
1774	S2(9, 4) = .00214
1775	S2(10, 4) = .00183
1776	S2(11, 4) = .0016
1777	S2(12, 4) = .00139
1778	S2(13, 4) = .00127
1779	S2(14, 4) = .00117
1780	S2(15, 4) = .00106
1781	
1782	
1783	S22(0, 0) = .021
1784	S22(1, 0) = .0142
1785	S22(2, 0) = .00987
1786	S22(3, 0) = .00662
1787	S22(4, 0) = .00519
1788	S22(5, 0) = .00426
1789	S22(6, 0) = .00344
1790	S22(7, 0) = .00293
1791	S22(8, 0) = .00249
1792	S22(9, 0) = .00201
1793	S22(10, 0) = .00173
1794	S22(11, 0) = .00152
1795	S22(12, 0) = .00132
1796	S22(13, 0) = .00121
1797	S22(14, 0) = .00112
1798	S22(15, 0) = .00102
1799	
1800	
1801	S22(0, 1) = .0196
1802	S22(1, 1) = .0131
1803	S22(2, 1) = .00911
1804	S22(3, 1) = .00613
1805	S22(4, 1) = .00484
1806	S22(5, 1) = .00399
1807	S22(6, 1) = .00323
1808	S22(7, 1) = .00275
1809	S22(8, 1) = .00234
1810	S22(9, 1) = .00189
1811	S22(10, 1) = .00163
1812	S22(11, 1) = .00144
1813	S22(12, 1) = .00126
1814	S22(13, 1) = .00115
1815	S22(14, 1) = .00107
1816	S22(15, 1) = .000978
1817	
1818	S22(0, 2) = .0182
1819	S22(1, 2) = .0121

1820	S22(2, 2) = .00838
1821	S22(3, 2) = .00568
1822	S22(4, 2) = .00451
1823	S22(5, 2) = .00374
1824	S22(6, 2) = .00303
1825	S22(7, 2) = .00258
1826	S22(8, 2) = .00219
1827	S22(9, 2) = .00178
1828	S22(10, 2) = .00154
1829	S22(11, 2) = .00136
1830	S22(12, 2) = .0012
1831	S22(13, 2) = .0011
1832	S22(14, 2) = .00102
1833	S22(15, 2) = .000937
1834	
1835	S22(0, 3) = .0168
1836	S22(1, 3) = .0111
1837	S22(2, 3) = .00769
1838	S22(3, 3) = .00526
1839	S22(4, 3) = .0042
1840	S22(5, 3) = .00351
1841	S22(6, 3) = .00284
1842	S22(7, 3) = .00242
1843	S22(8, 3) = .00206
1844	S22(9, 3) = .00167
1845	S22(10, 3) = .00145
1846	S22(11, 3) = .00129
1847	S22(12, 3) = .00113
1848	S22(13, 3) = .00105
1849	S22(14, 3) = .000974
1850	S22(15, 3) = .000897
1851	
1852	S22(0, 4) = .0155
1853	S22(1, 4) = .0101
1854	S22(2, 4) = .00704
1855	S22(3, 4) = .00485
1856	S22(4, 4) = .00391
1857	S22(5, 4) = .00328
1858	S22(6, 4) = .00266
1859	S22(7, 4) = .00226
1860	S22(8, 4) = .00193
1861	S22(9, 4) = .00156
1862	S22(10, 4) = .00137
1863	S22(11, 4) = .00122
1864	S22(12, 4) = .00108
1865	S22(13, 4) = .000996
1866	S22(14, 4) = .000929
1867	S22(15, 4) = .000858
1868	
1869	
1870	S3(0, 0) = .05
1871	S3(1, 0) = .0362
1872	S3(2, 0) = .0268
1873	S3(3, 0) = .0185
1874	S3(4, 0) = .0142
1875	S3(5, 0) = .0113
1876	S3(6, 0) = .00838
1877	S3(7, 0) = .00685
1878	S3(8, 0) = .00571
1879	S3(9, 0) = .00454
1880	S3(10, 0) = .00392
1881	S3(11, 0) = .00342
1882	S3(12, 0) = .00288

1883	S3(13, 0) = .00254
1884	S3(14, 0) = .00227
1885	S3(15, 0) = .002
1886	
1887	S3(0, 1) = .0227
1888	S3(1, 1) = .0156
1889	S3(2, 1) = .0109
1890	S3(3, 1) = .0073
1891	S3(4, 1) = .00571
1892	S3(5, 1) = .00467
1893	S3(6, 1) = .00377
1894	S3(7, 1) = .00321
1895	S3(8, 1) = .00273
1896	S3(9, 1) = .0022
1897	S3(10, 1) = .00188
1898	S3(11, 1) = .00165
1899	S3(12, 1) = .00143
1900	S3(13, 1) = .0013
1901	S3(14, 1) = .0012
1902	S3(15, 1) = .0011
1903	
1904	S3(0, 2) = .021
1905	S3(1, 2) = .0142
1906	S3(2, 2) = .00995
1907	S3(3, 2) = .0067
1908	S3(4, 2) = .00527
1909	S3(5, 2) = .00434
1910	S3(6, 2) = .0035
1911	S3(7, 2) = .00298
1912	S3(8, 2) = .00254
1913	S3(9, 2) = .00205
1914	S3(10, 2) = .00176
1915	S3(11, 2) = .00155
1916	S3(12, 2) = .00135
1917	S3(13, 2) = .00123
1918	S3(14, 2) = .00114
1919	S3(15, 2) = .00105
1920	
1921	S3(0, 3) = .0194
1922	S3(1, 3) = .013
1923	S3(2, 3) = .00907
1924	S3(3, 3) = .00614
1925	S3(4, 3) = .00486
1926	S3(5, 3) = .00402
1927	S3(6, 3) = .00325
1928	S3(7, 3) = .00277
1929	S3(8, 3) = .00236
1930	S3(9, 3) = .00191
1931	S3(10, 3) = .00165
1932	S3(11, 3) = .00146
1933	S3(12, 3) = .00127
1934	S3(13, 3) = .00117
1935	S3(14, 3) = .00108
1936	S3(15, 3) = .000994
1937	
1938	S3(0, 4) = .0178
1939	S3(1, 4) = .0118
1940	S3(2, 4) = .00823
1941	S3(3, 4) = .00561
1942	S3(4, 4) = .00448
1943	S3(5, 4) = .00373
1944	S3(6, 4) = .00302
1945	S3(7, 4) = .00257

1946	S3(8, 4) = .00219
1947	S3(9, 4) = .00177
1948	S3(10, 4) = .00154
1949	S3(11, 4) = .00137
1950	S3(12, 4) = .0012
1951	S3(13, 4) = .0011
1952	S3(14, 4) = .00103
1953	S3(15, 4) = .000944
1954	
1955	
1956	S33(0, 0) = .0162
1957	S33(1, 0) = .0107
1958	S33(2, 0) = .00744
1959	S33(3, 0) = .00512
1960	S33(4, 0) = .00412
1961	S33(5, 0) = .00345
1962	S33(6, 0) = .0028
1963	S33(7, 0) = .00238
1964	S33(8, 0) = .00203
1965	S33(9, 0) = .00164
1966	S33(10, 0) = .00144
1967	S33(11, 0) = .00128
1968	S33(12, 0) = .00113
1969	S33(13, 0) = .00104
1970	S33(14, 0) = .000972
1971	S33(15, 0) = .000897
1972	
1973	S33(0, 1) = .0146
1974	S33(1, 1) = .00958
1975	S33(2, 1) = .00671
1976	S33(3, 1) = .00467
1977	S33(4, 1) = .00379
1978	S33(5, 1) = .0032
1979	S33(6, 1) = .00259
1980	S33(7, 1) = .00221
1981	S33(8, 1) = .00188
1982	S33(9, 1) = .00152
1983	S33(10, 1) = .00134
1984	S33(11, 1) = .0012
1985	S33(12, 1) = .00106
1986	S33(13, 1) = .000985
1987	S33(14, 1) = .00092
1988	S33(15, 1) = .000851
1989	
1990	S33(0, 2) = .0131
1991	S33(1, 2) = .00857
1992	S33(2, 2) = .00604
1993	S33(3, 2) = .00425
1994	S33(4, 2) = .00348
1995	S33(5, 2) = .00296
1996	S33(6, 2) = .0024
1997	S33(7, 2) = .00204
1998	S33(8, 2) = .00174
1999	S33(9, 2) = .00141
2000	S33(10, 2) = .00125
2001	S33(11, 2) = .00113
2002	S33(12, 2) = .001
2003	S33(13, 2) = .00093
2004	S33(14, 2) = .000871
2005	S33(15, 2) = .000807
2006	
2007	S33(0, 3) = .0117
2008	S33(1, 3) = .00767

2009	S33(2, 3) = .00543
2010	S33(3, 3) = .00388
2011	S33(4, 3) = .0032
2012	S33(5, 3) = .00273
2013	S33(6, 3) = .00221
2014	S33(7, 3) = .00188
2015	S33(8, 3) = .0016
2016	S33(9, 3) = .00131
2017	S33(10, 3) = .00117
2018	S33(11, 3) = .00105
2019	S33(12, 3) = .000942
2020	S33(13, 3) = .000877
2021	S33(14, 3) = .000823
2022	S33(15, 3) = .000765
2023	
2024	S33(0, 4) = .0104
2025	S33(1, 4) = .00679
2026	S33(2, 4) = .00488
2027	S33(3, 4) = .00354
2028	S33(4, 4) = .00295
2029	S33(5, 4) = .00251
2030	S33(6, 4) = .00203
2031	S33(7, 4) = .00173
2032	S33(8, 4) = .00147
2033	S33(9, 4) = .00122
2034	S33(10, 4) = .00109
2035	S33(11, 4) = .000987
2036	S33(12, 4) = .000887
2037	S33(13, 4) = .000828
2038	S33(14, 4) = .000779
2039	S33(15, 4) = .000726
2040	
2041	+++ GAISOU DATA +++ HORYU .
2042	
2043	S4(0) = .0503
2044	S4(1) = .0366
2045	S4(2) = .0273
2046	S4(3) = .0186
2047	S4(4) = .0142
2048	S4(5) = .0109
2049	S4(6) = .00842
2050	S4(7) = .00678
2051	S4(8) = .00568
2052	S4(9) = .0047
2053	S4(10) = .00404
2054	S4(11) = .0035
2055	S4(12) = .00295
2056	S4(13) = .00251
2057	S4(14) = .0023
2058	S4(15) = .00213
2059	
2060	S5(0) = .0503
2061	S5(1) = .0366
2062	S5(2) = .0273
2063	S5(3) = .0186
2064	S5(4) = .0142
2065	S5(5) = .0109
2066	S5(6) = .00842
2067	S5(7) = .00678
2068	S5(8) = .00568
2069	S5(9) = .0047
2070	S5(10) = .00404
2071	S5(11) = .0035

```

2072      S5(12) = .00276
2073      S5(13) = .0025
2074      S5(14) = .0022
2075      S5(15) = .00192
2076      +++
2077      +++ SUS316 A, B, C (OLD)
2078
2079      DATA S6/ 0.0283 ,0.0204 ,0.0151 ,0.0107 ,0.00848 ,0.00694 ,
2080      *      0.00553 ,0.00471 ,0.00401 ,0.00325 ,0.00277 ,0.00239 ,
2081      *      0.00202 ,0.00181 ,0.00165 ,0.00147 ,
2082      *      0.0272 ,0.0195 ,0.0144 ,0.0101 ,0.00803 ,0.00657 ,
2083      *      0.00524 ,0.00446 ,0.00380 ,0.00308 ,0.00262 ,0.00226 ,
2084      *      0.00192 ,0.00172 ,0.00156 ,0.00140 ,
2085      *      0.0260 ,0.0186 ,0.0137 ,0.00956 ,0.00758 ,0.00620 ,
2086      *      0.00495 ,0.00422 ,0.00359 ,0.00290 ,0.00247 ,0.00214 ,
2087      *      0.00182 ,0.00163 ,0.00148 ,0.00133 ,
2088      *      0.0248 ,0.0176 ,0.0129 ,0.00900 ,0.00714 ,0.00584 ,
2089      *      0.00466 ,0.00397 ,0.00338 ,0.00274 ,0.00233 ,0.00202 ,
2090      *      0.00171 ,0.00154 ,0.00140 ,0.00126 ,
2091      *      0.0236 ,0.0167 ,0.0122 ,0.00845 ,0.00669 ,0.00548 ,
2092      *      0.00438 ,0.00373 ,0.00318 ,0.00257 ,0.00219 ,0.00190 ,
2093      *      0.00161 ,0.00145 ,0.00132 ,0.00119 /
2094      DATA S66/0.0223 ,0.0157 ,0.0114 ,0.00791 ,0.00626 ,0.00512 ,
2095      *      0.00410 ,0.00349 ,0.00298 ,0.00241 ,0.00205 ,0.00178 ,
2096      *      0.00152 ,0.00137 ,0.00124 ,0.00112 ,
2097      *      0.0211 ,0.0147 ,0.0107 ,0.00737 ,0.00584 ,0.00478 ,
2098      *      0.00383 ,0.00327 ,0.00278 ,0.00225 ,0.00192 ,0.00167 ,
2099      *      0.00142 ,0.00128 ,0.00117 ,0.00105 ,
2100      *      0.0198 ,0.0138 ,0.00991 ,0.00685 ,0.00542 ,0.00445 ,
2101      *      0.00357 ,0.00304 ,0.00259 ,0.00210 ,0.00179 ,0.00156 ,
2102      *      0.00133 ,0.00120 ,0.00110 ,0.000988 ,
2103      *      0.0186 ,0.0128 ,0.00919 ,0.00634 ,0.00503 ,0.00413 ,
2104      *      0.00332 ,0.00283 ,0.00241 ,0.00195 ,0.00166 ,0.00145 ,
2105      *      0.00124 ,0.00112 ,0.00103 ,0.000927 ,
2106      *      0.0173 ,0.0119 ,0.00849 ,0.00585 ,0.00464 ,0.00382 ,
2107      *      0.00309 ,0.00263 ,0.00223 ,0.00181 ,0.00154 ,0.00135 ,
2108      *      0.00116 ,0.00105 ,0.000960 ,0.000868 /
2109      DATA S7/ 0.0219 ,0.0159 ,0.0120 ,0.0881 ,0.00723 ,0.00611 ,
2110      *      0.00494 ,0.00421 ,0.00359 ,0.00291 ,0.00253 ,0.00224 ,
2111      *      0.00195 ,0.00178 ,0.00164 ,0.00147 ,
2112      *      0.0202 ,0.0146 ,0.0111 ,0.00809 ,0.00666 ,0.00563 ,
2113      *      0.00456 ,0.00388 ,0.00331 ,0.00268 ,0.00234 ,0.00207 ,
2114      *      0.00181 ,0.00165 ,0.00153 ,0.00139 ,
2115      *      0.0186 ,0.0134 ,0.0101 ,0.00740 ,0.00610 ,0.00517 ,
2116      *      0.00419 ,0.00357 ,0.00304 ,0.00246 ,0.00215 ,0.00191 ,
2117      *      0.00167 ,0.00153 ,0.00142 ,0.00129 ,
2118      *      0.0169 ,0.0122 ,0.00917 ,0.00674 ,0.00557 ,0.00474 ,
2119      *      0.00383 ,0.00327 ,0.00279 ,0.00226 ,0.00198 ,0.00176 ,
2120      *      0.00155 ,0.00142 ,0.00131 ,0.00120 ,
2121      *      0.0153 ,0.0110 ,0.00830 ,0.00612 ,0.00507 ,0.00432 ,
2122      *      0.00349 ,0.00298 ,0.00254 ,0.00206 ,0.00181 ,0.00162 ,
2123      *      0.00142 ,0.00131 ,0.00122 ,0.00111 /
2124      DATA S77/0.0138 ,0.00989 ,0.00747 ,0.00553 ,0.00460 ,0.00392 ,
2125      *      0.00317 ,0.00270 ,0.00230 ,0.00188 ,0.00166 ,0.00149 ,
2126      *      0.00131 ,0.00121 ,0.00112 ,0.00103 ,
2127      *      0.0124 ,0.00885 ,0.00670 ,0.00499 ,0.00417 ,0.00355 ,
2128      *      0.00287 ,0.00244 ,0.00208 ,0.00171 ,0.00151 ,0.00136 ,
2129      *      0.00120 ,0.00111 ,0.00103 ,0.000951 ,
2130      *      0.0110 ,0.00788 ,0.00599 ,0.00448 ,0.00376 ,0.00320 ,
2131      *      0.00259 ,0.00221 ,0.00188 ,0.00155 ,0.00138 ,0.00124 ,
2132      *      0.00110 ,0.00102 ,0.000951 ,0.000877 ,
2133      *      0.00973 ,0.00698 ,0.00533 ,0.00402 ,0.00339 ,0.00289 ,
2134      *      0.00234 ,0.00199 ,0.00169 ,0.00141 ,0.00125 ,0.00113 ,

```

```

2135 * 0.00101,0.000935,0.000874,0.000807,
2136 * 0.00856,0.00617,0.00473,0.00360,0.00305,0.00260,
2137 * 0.00210,0.00179,0.00152,0.00127,0.00114,0.00103,
2138 * 0.000921,0.000856,0.000802,0.000742 /
2139 DATA S8/ 0.0187,0.0137,0.0106,0.00796,0.00666,0.00567,
2140 * 0.00459,0.00391,0.00333,0.00273,0.00241,0.00216,
2141 * 0.00191,0.00176,0.00163,0.00147,
2142 * 0.0168,0.0124,0.00954,0.00720,0.00604,0.00515,
2143 * 0.00416,0.00355,0.00302,0.00249,0.00220,0.00198,
2144 * 0.00175,0.00162,0.00151,0.00138,
2145 * 0.0151,0.0111,0.00856,0.00649,0.00547,0.00466,
2146 * 0.00377,0.00321,0.00273,0.00226,0.00201,0.00181,
2147 * 0.00161,0.00149,0.00139,0.00128,
2148 * 0.0134,0.00987,0.00765,0.00583,0.00493,0.00420,
2149 * 0.00340,0.00289,0.00246,0.00205,0.00183,0.00165,
2150 * 0.00147,0.00136,0.00127,0.00118,
2151 * 0.0119,0.00875,0.00680,0.00522,0.00443,0.00377,
2152 * 0.00305,0.00260,0.00222,0.00185,0.00166,0.00150,
2153 * 0.00134,0.00125,0.00117,0.00108 /
2154 DATA S88/0.0104,0.00771,0.00603,0.00466,0.00397,0.00338,
2155 * 0.00273,0.00233,0.00199,0.00167,0.00150,0.00136,
2156 * 0.00122,0.00114,0.00107,0.000990,
2157 * 0.00911,0.00677,0.00533,0.00415,0.00353,0.00301,
2158 * 0.00243,0.00207,0.00178,0.00151,0.00135,0.00124,
2159 * 0.00111,0.00104,0.000975,0.000906,
2160 * 0.00793,0.00593,0.00470,0.00369,0.00314,0.00268,
2161 * 0.00216,0.00184,0.00159,0.00135,0.00122,0.00112,
2162 * 0.00101,0.000944,0.000889,0.000829,
2163 * 0.00687,0.00518,0.00414,0.00327,0.00279,0.00238,
2164 * 0.00192,0.00164,0.00142,0.00122,0.00110,0.00101,
2165 * 0.000916,0.000859,0.000810,0.000757,
2166 * 0.00593,0.00452,0.00364,0.00290,0.00247,0.00211,
2167 * 0.00170,0.00145,0.00127,0.00109,0.000992,0.000913,
2168 * 0.000831,0.000780,0.000738,0.000690 /
2169 +++
2170
2171 +++ SUS316 A, B, C (NEW) +++
2172
2173
2174 S6(0, 0) = .05
2175 S6(1, 0) = .0362
2176 S6(2, 0) = .0268
2177 S6(3, 0) = .0185
2178 S6(4, 0) = .0142
2179 S6(5, 0) = .0113
2180 S6(6, 0) = .00838
2181 S6(7, 0) = .00685
2182 S6(8, 0) = .00571
2183 S6(9, 0) = .00454
2184 S6(10, 0) = .00392
2185 S6(11, 0) = .00342
2186 S6(12, 0) = .00288
2187 S6(13, 0) = .00254
2188 S6(14, 0) = .00227
2189 S6(15, 0) = .002
2190
2191 S6(0, 1) = .0331
2192 S6(1, 1) = .0238
2193 S6(2, 1) = .0173
2194 S6(3, 1) = .0116
2195 S6(4, 1) = .00891
2196 S6(5, 1) = .00705
2197 S6(6, 1) = .00544

```



2198	S6(7, 1) = .00462
2199	S6(8, 1) = .00394
2200	S6(9, 1) = .00318
2201	S6(10, 1) = .00271
2202	S6(11, 1) = .00231
2203	S6(12, 1) = .00193
2204	S6(13, 1) = .00172
2205	S6(14, 1) = .00156
2206	S6(15, 1) = .0014
2207	
2208	S6(0, 2) = .0323
2209	S6(1, 2) = .0232
2210	S6(2, 2) = .0168
2211	S6(3, 2) = .0113
2212	S6(4, 2) = .00864
2213	S6(5, 2) = .00684
2214	S6(6, 2) = .00529
2215	S6(7, 2) = .0045
2216	S6(8, 2) = .00383
2217	S6(9, 2) = .0031
2218	S6(10, 2) = .00264
2219	S6(11, 2) = .00225
2220	S6(12, 2) = .00189
2221	S6(13, 2) = .00169
2222	S6(14, 2) = .00153
2223	S6(15, 2) = .00137
2224	
2225	S6(0, 3) = .0316
2226	S6(1, 3) = .0225
2227	S6(2, 3) = .0163
2228	S6(3, 3) = .0109
2229	S6(4, 3) = .00836
2230	S6(5, 3) = .00662
2231	S6(6, 3) = .00513
2232	S6(7, 3) = .00437
2233	S6(8, 3) = .00372
2234	S6(9, 3) = .00301
2235	S6(10, 3) = .00257
2236	S6(11, 3) = .00219
2237	S6(12, 3) = .00184
2238	S6(13, 3) = .00165
2239	S6(14, 3) = .0015
2240	S6(15, 3) = .00134
2241	
2242	S6(0, 4) = .0308
2243	S6(1, 4) = .0219
2244	S6(2, 4) = .0158
2245	S6(3, 4) = .0105
2246	S6(4, 4) = .00807
2247	S6(5, 4) = .00641
2248	S6(6, 4) = .00498
2249	S6(7, 4) = .00424
2250	S6(8, 4) = .00361
2251	S6(9, 4) = .00292
2252	S6(10, 4) = .00249
2253	S6(11, 4) = .00213
2254	S6(12, 4) = .00179
2255	S6(13, 4) = .00161
2256	S6(14, 4) = .00146
2257	S6(15, 4) = .00131
2258	
2259	
2260	S66(0, 0) = .03

2261	S66(1, 0) = .0212
2262	S66(2, 0) = .0152
2263	S66(3, 0) = .0102
2264	S66(4, 0) = .00779
2265	S66(5, 0) = .00619
2266	S66(6, 0) = .00482
2267	S66(7, 0) = .00411
2268	S66(8, 0) = .0035
2269	S66(9, 0) = .00283
2270	S66(10, 0) = .00241
2271	S66(11, 0) = .00207
2272	S66(12, 0) = .00174
2273	S66(13, 0) = .00157
2274	S66(14, 0) = .00143
2275	S66(15, 0) = .00128
2276	
2277	S66(0, 1) = .0291
2278	S66(1, 1) = .0205
2279	S66(2, 1) = .0147
2280	S66(3, 1) = .00978
2281	S66(4, 1) = .0075
2282	S66(5, 1) = .00597
2283	S66(6, 1) = .00467
2284	S66(7, 1) = .00398
2285	S66(8, 1) = .00339
2286	S66(9, 1) = .00274
2287	S66(10, 1) = .00233
2288	S66(11, 1) = .002
2289	S66(12, 1) = .0017
2290	S66(13, 1) = .00153
2291	S66(14, 1) = .00139
2292	S66(15, 1) = .00125
2293	
2294	S66(0, 2) = .0283
2295	S66(1, 2) = .0198
2296	S66(2, 2) = .0141
2297	S66(3, 2) = .0094
2298	S66(4, 2) = .00721
2299	S66(5, 2) = .00576
2300	S66(6, 2) = .00452
2301	S66(7, 2) = .00385
2302	S66(8, 2) = .00328
2303	S66(9, 2) = .00265
2304	S66(10, 2) = .00226
2305	S66(11, 2) = .00194
2306	S66(12, 2) = .00165
2307	S66(13, 2) = .00149
2308	S66(14, 2) = .00136
2309	S66(15, 2) = .00122
2310	
2311	S66(0, 3) = .0274
2312	S66(1, 3) = .0191
2313	S66(2, 3) = .0136
2314	S66(3, 3) = .00902
2315	S66(4, 3) = .00693
2316	S66(5, 3) = .00554
2317	S66(6, 3) = .00436
2318	S66(7, 3) = .00372
2319	S66(8, 3) = .00317
2320	S66(9, 3) = .00256
2321	S66(10, 3) = .00218
2322	S66(11, 3) = .00188
2323	S66(12, 3) = .0016

2324	S66(13, 3) = .00144
2325	S66(14, 3) = .00132
2326	S66(15, 3) = .00119
2327	
2328	S66(0, 4) = .0265
2329	S66(1, 4) = .0184
2330	S66(2, 4) = .013
2331	S66(3, 4) = .00864
2332	S66(4, 4) = .00665
2333	S66(5, 4) = .00533
2334	S66(6, 4) = .00421
2335	S66(7, 4) = .00359
2336	S66(8, 4) = .00306
2337	S66(9, 4) = .00247
2338	S66(10, 4) = .00211
2339	S66(11, 4) = .00182
2340	S66(12, 4) = .00155
2341	S66(13, 4) = .00141
2342	S66(14, 4) = .00129
2343	S66(15, 4) = .00116
2344	
2345	
2346	S7(0, 0) = .05
2347	S7(1, 0) = .0362
2348	S7(2, 0) = .0268
2349	S7(3, 0) = .0185
2350	S7(4, 0) = .0142
2351	S7(5, 0) = .0113
2352	S7(6, 0) = .00838
2353	S7(7, 0) = .00685
2354	S7(8, 0) = .00571
2355	S7(9, 0) = .00454
2356	S7(10, 0) = .00392
2357	S7(11, 0) = .00342
2358	S7(12, 0) = .00288
2359	S7(13, 0) = .00254
2360	S7(14, 0) = .00227
2361	S7(15, 0) = .002
2362	
2363	S7(0, 1) = .0265
2364	S7(1, 1) = .0185
2365	S7(2, 1) = .0131
2366	S7(3, 1) = .00875
2367	S7(4, 1) = .00676
2368	S7(5, 1) = .00544
2369	S7(6, 1) = .00431
2370	S7(7, 1) = .00367
2371	S7(8, 1) = .00313
2372	S7(9, 1) = .00253
2373	S7(10, 1) = .00216
2374	S7(11, 1) = .00187
2375	S7(12, 1) = .0016
2376	S7(13, 1) = .00145
2377	S7(14, 1) = .00133
2378	S7(15, 1) = .0012
2379	
2380	S7(0, 2) = .0252
2381	S7(1, 2) = .0174
2382	S7(2, 2) = .0123
2383	S7(3, 2) = .00819
2384	S7(4, 2) = .00635
2385	S7(5, 2) = .00513
2386	S7(6, 2) = .00409

2387	S7(7, 2) = .00348
2388	S7(8, 2) = .00297
2389	S7(9, 2) = .0024
2390	S7(10, 2) = .00204
2391	S7(11, 2) = .00178
2392	S7(12, 2) = .00153
2393	S7(13, 2) = .00138
2394	S7(14, 2) = .00127
2395	S7(15, 2) = .00115
2396	
2397	S7(0, 3) = .0238
2398	S7(1, 3) = .0163
2399	S7(2, 3) = .0115
2400	S7(3, 3) = .00765
2401	S7(4, 3) = .00595
2402	S7(5, 3) = .00483
2403	S7(6, 3) = .00387
2404	S7(7, 3) = .0033
2405	S7(8, 3) = .00281
2406	S7(9, 3) = .00227
2407	S7(10, 3) = .00193
2408	S7(11, 3) = .00169
2409	S7(12, 3) = .00146
2410	S7(13, 3) = .00132
2411	S7(14, 3) = .00122
2412	S7(15, 3) = .00111
2413	
2414	S7(0, 4) = .0224
2415	S7(1, 4) = .0152
2416	S7(2, 4) = .0107
2417	S7(3, 4) = .00712
2418	S7(4, 4) = .00556
2419	S7(5, 4) = .00454
2420	S7(6, 4) = .00366
2421	S7(7, 4) = .00311
2422	S7(8, 4) = .00265
2423	S7(9, 4) = .00214
2424	S7(10, 4) = .00183
2425	S7(11, 4) = .0016
2426	S7(12, 4) = .00139
2427	S7(13, 4) = .00127
2428	S7(14, 4) = .00117
2429	S7(15, 4) = .00106
2430	
2431	
2432	S77(0, 0) = .021
2433	S77(1, 0) = .0142
2434	S77(2, 0) = .00987
2435	S77(3, 0) = .00662
2436	S77(4, 0) = .00519
2437	S77(5, 0) = .00426
2438	S77(6, 0) = .00344
2439	S77(7, 0) = .00293
2440	S77(8, 0) = .00249
2441	S77(9, 0) = .00201
2442	S77(10, 0) = .00173
2443	S77(11, 0) = .00152
2444	S77(12, 0) = .00132
2445	S77(13, 0) = .00121
2446	S77(14, 0) = .00112
2447	S77(15, 0) = .00102
2448	
2449	S77(0, 1) = .0196

2450	S77(1, 1) = .0131
2451	S77(2, 1) = .00911
2452	S77(3, 1) = .00613
2453	S77(4, 1) = .00484
2454	S77(5, 1) = .00399
2455	S77(6, 1) = .00323
2456	S77(7, 1) = .00275
2457	S77(8, 1) = .00234
2458	S77(9, 1) = .00189
2459	S77(10, 1) = .00163
2460	S77(11, 1) = .00144
2461	S77(12, 1) = .00126
2462	S77(13, 1) = .00115
2463	S77(14, 1) = .00107
2464	S77(15, 1) = .000978
2465	
2466	S77(0, 2) = .0182
2467	S77(1, 2) = .0121
2468	S77(2, 2) = .00838
2469	S77(3, 2) = .00568
2470	S77(4, 2) = .00451
2471	S77(5, 2) = .00374
2472	S77(6, 2) = .00303
2473	S77(7, 2) = .00258
2474	S77(8, 2) = .00219
2475	S77(9, 2) = .00178
2476	S77(10, 2) = .00154
2477	S77(11, 2) = .00136
2478	S77(12, 2) = .0012
2479	S77(13, 2) = .0011
2480	S77(14, 2) = .00102
2481	S77(15, 2) = .000937
2482	
2483	S77(0, 3) = .0168
2484	S77(1, 3) = .0111
2485	S77(2, 3) = .00769
2486	S77(3, 3) = .00525
2487	S77(4, 3) = .0042
2488	S77(5, 3) = .00351
2489	S77(6, 3) = .00284
2490	S77(7, 3) = .00242
2491	S77(8, 3) = .00206
2492	S77(9, 3) = .00167
2493	S77(10, 3) = .00145
2494	S77(11, 3) = .00129
2495	S77(12, 3) = .00113
2496	S77(13, 3) = .00105
2497	S77(14, 3) = .000974
2498	S77(15, 3) = .000897
2499	
2500	S77(0, 4) = .0155
2501	S77(1, 4) = .0101
2502	S77(2, 4) = .00704
2503	S77(3, 4) = .00485
2504	S77(4, 4) = .00391
2505	S77(5, 4) = .00328
2506	S77(6, 4) = .00266
2507	S77(7, 4) = .00226
2508	S77(8, 4) = .00193
2509	S77(9, 4) = .00156
2510	S77(10, 4) = .00137
2511	S77(11, 4) = .00122
2512	S77(12, 4) = .00108

2513	S77(13, 4) = .000996
2514	S77(14, 4) = .000929
2515	S77(15, 4) = .000858
2516	
2517	
2518	S8(0, 0) = .05
2519	S8(1, 0) = .0362
2520	S8(2, 0) = .0268
2521	S8(3, 0) = .0185
2522	S8(4, 0) = .0142
2523	S8(5, 0) = .0113
2524	S8(6, 0) = .00838
2525	S8(7, 0) = .00685
2526	S8(8, 0) = .00571
2527	S8(9, 0) = .00454
2528	S8(10, 0) = .00392
2529	S8(11, 0) = .00342
2530	S8(12, 0) = .00288
2531	S8(13, 0) = .00254
2532	S8(14, 0) = .00227
2533	S8(15, 0) = .002
2534	
2535	S8(0, 1) = .0227
2536	S8(1, 1) = .0155
2537	S8(2, 1) = .0109
2538	S8(3, 1) = .0073
2539	S8(4, 1) = .00571
2540	S8(5, 1) = .00467
2541	S8(6, 1) = .00377
2542	S8(7, 1) = .00321
2543	S8(8, 1) = .00273
2544	S8(9, 1) = .0022
2545	S8(10, 1) = .00188
2546	S8(11, 1) = .00165
2547	S8(12, 1) = .00143
2548	S8(13, 1) = .0013
2549	S8(14, 1) = .0012
2550	S8(15, 1) = .0011
2551	
2552	S8(0, 2) = .021
2553	S8(1, 2) = .0142
2554	S8(2, 2) = .00995
2555	S8(3, 2) = .0067
2556	S8(4, 2) = .00527
2557	S8(5, 2) = .00434
2558	S8(6, 2) = .0035
2559	S8(7, 2) = .00298
2560	S8(8, 2) = .00254
2561	S8(9, 2) = .00205
2562	S8(10, 2) = .00176
2563	S8(11, 2) = .00155
2564	S8(12, 2) = .00135
2565	S8(13, 2) = .00123
2566	S8(14, 2) = .00114
2567	S8(15, 2) = .00105
2568	
2569	S8(0, 3) = .0194
2570	S8(1, 3) = .013
2571	S8(2, 3) = .00907
2572	S8(3, 3) = .00614
2573	S8(4, 3) = .00486
2574	S8(5, 3) = .00402
2575	S8(6, 3) = .00325

2576	S8(7, 3) = .00277
2577	S8(8, 3) = .00236
2578	S8(9, 3) = .00191
2579	S8(10, 3) = .00165
2580	S8(11, 3) = .00146
2581	S8(12, 3) = .00127
2582	S8(13, 3) = .00117
2583	S8(14, 3) = .00108
2584	S8(15, 3) = .000994
2585	
2586	S8(0, 4) = .0178
2587	S8(1, 4) = .0118
2588	S8(2, 4) = .00823
2589	S8(3, 4) = .00561
2590	S8(4, 4) = .00448
2591	S8(5, 4) = .00373
2592	S8(6, 4) = .00302
2593	S8(7, 4) = .00257
2594	S8(8, 4) = .00219
2595	S8(9, 4) = .00177
2596	S8(10, 4) = .00154
2597	S8(11, 4) = .00137
2598	S8(12, 4) = .0012
2599	S8(13, 4) = .0011
2600	S8(14, 4) = .00103
2601	S8(15, 4) = .000944
2602	
2603	
2604	S88(0, 0) = .0162
2605	S88(1, 0) = .0107
2606	S88(2, 0) = .00744
2607	S88(3, 0) = .00512
2608	S88(4, 0) = .00412
2609	S88(5, 0) = .00345
2610	S88(6, 0) = .0028
2611	S88(7, 0) = .00238
2612	S88(8, 0) = .00203
2613	S88(9, 0) = .00164
2614	S88(10, 0) = .00144
2615	S88(11, 0) = .00128
2616	S88(12, 0) = .00113
2617	S88(13, 0) = .00104
2618	S88(14, 0) = .000972
2619	S88(15, 0) = .000897
2620	
2621	S88(0, 1) = .0146
2622	S88(1, 1) = .00958
2623	S88(2, 1) = .00671
2624	S88(3, 1) = .00467
2625	S88(4, 1) = .00379
2626	S88(5, 1) = .0032
2627	S88(6, 1) = .00259
2628	S88(7, 1) = .00221
2629	S88(8, 1) = .00188
2630	S88(9, 1) = .00152
2631	S88(10, 1) = .00134
2632	S88(11, 1) = .0012
2633	S88(12, 1) = .00106
2634	S88(13, 1) = .000985
2635	S88(14, 1) = .00092
2636	S88(15, 1) = .000851
2637	
2638	S88(0, 2) = .0131

2639	S88(1, 2) = .00857
2640	S88(2, 2) = .00604
2641	S88(3, 2) = .00425
2642	S88(4, 2) = .00348
2643	S88(5, 2) = .00296
2644	S88(6, 2) = .0024
2645	S88(7, 2) = .00204
2646	S88(8, 2) = .00174
2647	S88(9, 2) = .00141
2648	S88(10, 2) = .00125
2649	S88(11, 2) = .00113
2650	S88(12, 2) = .001
2651	S88(13, 2) = .00093
2652	S88(14, 2) = .000871
2653	S88(15, 2) = .000807
2654	
2655	S88(0, 3) = .0117
2656	S88(1, 3) = .00767
2657	S88(2, 3) = .00543
2658	S88(3, 3) = .00388
2659	S88(4, 3) = .0032
2660	S88(5, 3) = .00273
2661	S88(6, 3) = .00221
2662	S88(7, 3) = .00188
2663	S88(8, 3) = .0016
2664	S88(9, 3) = .00131
2665	S88(10, 3) = .00117
2666	S88(11, 3) = .00105
2667	S88(12, 3) = .000942
2668	S88(13, 3) = .000877
2669	S88(14, 3) = .000823
2670	S88(15, 3) = .000765
2671	
2672	S88(0, 4) = .0104
2673	S88(1, 4) = .00679
2674	S88(2, 4) = .00488
2675	S88(3, 4) = .00354
2676	S88(4, 4) = .00295
2677	S88(5, 4) = .00251
2678	S88(6, 4) = .00203
2679	S88(7, 4) = .00173
2680	S88(8, 4) = .00147
2681	S88(9, 4) = .00122
2682	S88(10, 4) = .00109
2683	S88(11, 4) = .000987
2684	S88(12, 4) = .000887
2685	S88(13, 4) = .000828
2686	S88(14, 4) = .000779
2687	S88(15, 4) = .000726
2688	
2689	+++
2690	
2691	
2692	SA(0, 0) = .0387
2693	SA(1, 0) = .0273
2694	SA(2, 0) = .0202
2695	SA(3, 0) = .0137
2696	SA(4, 0) = .0103
2697	SA(5, 0) = .0077
2698	SA(6, 0) = .00553
2699	SA(7, 0) = .00427
2700	SA(8, 0) = .00343
2701	SA(9, 0) = .00253



2702	SA(10, 0) = .00207
2703	SA(11, 0) = .00165
2704	SA(12, 0) = .00133
2705	SA(13, 0) = .0011
2706	SA(14, 0) = .000945
2707	SA(15, 0) = .000833
2708	
2709	SA(0, 1) = .0387
2710	SA(1, 1) = .0258
2711	SA(2, 1) = .0172
2712	SA(3, 1) = .0105
2713	SA(4, 1) = .00794
2714	SA(5, 1) = .00639
2715	SA(6, 1) = .00488
2716	SA(7, 1) = .00398
2717	SA(8, 1) = .00325
2718	SA(9, 1) = .00248
2719	SA(10, 1) = .00202
2720	SA(11, 1) = .00165
2721	SA(12, 1) = .00131
2722	SA(13, 1) = .0011
2723	SA(14, 1) = .000945
2724	SA(15, 1) = .000833
2725	
2726	SA(0, 2) = .0387
2727	SA(1, 2) = .0252
2728	SA(2, 2) = .0164
2729	SA(3, 2) = .00993
2730	SA(4, 2) = .00751
2731	SA(5, 2) = .00603
2732	SA(6, 2) = .00466
2733	SA(7, 2) = .00384
2734	SA(8, 2) = .00316
2735	SA(9, 2) = .00244
2736	SA(10, 2) = .00201
2737	SA(11, 2) = .00165
2738	SA(12, 2) = .00131
2739	SA(13, 2) = .0011
2740	SA(14, 2) = .000945
2741	SA(15, 2) = .000833
2742	
2743	SA(0, 3) = .0387
2744	SA(1, 3) = .0245
2745	SA(2, 3) = .0156
2746	SA(3, 3) = .00938
2747	SA(4, 3) = .00708
2748	SA(5, 3) = .00569
2749	SA(6, 3) = .00445
2750	SA(7, 3) = .00369
2751	SA(8, 3) = .00306
2752	SA(9, 3) = .00239
2753	SA(10, 3) = .00199
2754	SA(11, 3) = .00165
2755	SA(12, 3) = .00131
2756	SA(13, 3) = .0011
2757	SA(14, 3) = .000945
2758	SA(15, 3) = .000833
2759	
2760	
2761	SAA(0, 0) = .0387
2762	SAA(1, 0) = .0239
2763	SAA(2, 0) = .0147
2764	SAA(3, 0) = .00882

2765	SAA(4, 0) = .00665
2766	SAA(5, 0) = .00534
2767	SAA(6, 0) = .00423
2768	SAA(7, 0) = .00354
2769	SAA(8, 0) = .00297
2770	SAA(9, 0) = .00235
2771	SAA(10, 0) = .00197
2772	SAA(11, 0) = .00165
2773	SAA(12, 0) = .00131
2774	SAA(13, 0) = .0011
2775	SAA(14, 0) = .000945
2776	SAA(15, 0) = .000833
2777	
2778	SAA(0, 1) = .0387
2779	SAA(1, 1) = .0232
2780	SAA(2, 1) = .0139
2781	SAA(3, 1) = .00827
2782	SAA(4, 1) = .00623
2783	SAA(5, 1) = .005
2784	SAA(6, 1) = .004
2785	SAA(7, 1) = .00338
2786	SAA(8, 1) = .00285
2787	SAA(9, 1) = .00228
2788	SAA(10, 1) = .00193
2789	SAA(11, 1) = .00163
2790	SAA(12, 1) = .0013
2791	SAA(13, 1) = .0011
2792	SAA(14, 1) = .000945
2793	SAA(15, 1) = .000833
2794	
2795	SAA(0, 2) = .0387
2796	SAA(1, 2) = .0223
2797	SAA(2, 2) = .0131
2798	SAA(3, 2) = .00772
2799	SAA(4, 2) = .00581
2800	SAA(5, 2) = .00467
2801	SAA(6, 2) = .00374
2802	SAA(7, 2) = .00318
2803	SAA(8, 2) = .00271
2804	SAA(9, 2) = .00219
2805	SAA(10, 2) = .00187
2806	SAA(11, 2) = .00159
2807	SAA(12, 2) = .00129
2808	SAA(13, 2) = .0011
2809	SAA(14, 2) = .000945
2810	SAA(15, 2) = .000833
2811	
2812	SAA(0, 3) = .0387
2813	SAA(1, 3) = .0211
2814	SAA(2, 3) = .0122
2815	SAA(3, 3) = .00718
2816	SAA(4, 3) = .0054
2817	SAA(5, 3) = .00434
2818	SAA(6, 3) = .00348
2819	SAA(7, 3) = .00296
2820	SAA(8, 3) = .00253
2821	SAA(9, 3) = .00204
2822	SAA(10, 3) = .00174
2823	SAA(11, 3) = .00151
2824	SAA(12, 3) = .00125
2825	SAA(13, 3) = .00109
2826	SAA(14, 3) = .000945
2827	SAA(15, 3) = .000833

2828	
2829	
2830	SB(0, 0) = .0387
2831	SB(1, 0) = .0273
2832	SB(2, 0) = .0202
2833	SB(3, 0) = .0137
2834	SB(4, 0) = .0103
2835	SB(5, 0) = .0077
2836	SB(6, 0) = .00553
2837	SB(7, 0) = .00427
2838	SB(8, 0) = .00343
2839	SB(9, 0) = .00253
2840	SB(10, 0) = .00207
2841	
2842	SB(11, 0) = .00165
2843	SB(12, 0) = .00133
2844	SB(13, 0) = .0011
2845	SB(14, 0) = .000945
2846	SB(15, 0) = .000833
2847	
2848	SB(0, 1) = .0387
2849	SB(1, 1) = .0229
2850	SB(2, 1) = .0141
2851	SB(3, 1) = .00872
2852	SB(4, 1) = .00669
2853	SB(5, 1) = .00544
2854	SB(6, 1) = .00429
2855	SB(7, 1) = .00358
2856	SB(8, 1) = .003
2857	SB(9, 1) = .00236
2858	SB(10, 1) = .00198
2859	SB(11, 1) = .00165
2860	SB(12, 1) = .00131
2861	SB(13, 1) = .0011
2862	SB(14, 1) = .000945
2863	SB(15, 1) = .000833
2864	
2865	SB(0, 2) = .0383
2866	SB(1, 2) = .0214
2867	SB(2, 2) = .0131
2868	SB(3, 2) = .00808
2869	SB(4, 2) = .00621
2870	SB(5, 2) = .00506
2871	SB(6, 2) = .00404
2872	SB(7, 2) = .00341
2873	SB(8, 2) = .00287
2874	SB(9, 2) = .0023
2875	SB(10, 2) = .00194
2876	SB(11, 2) = .00163
2877	SB(12, 2) = .0013
2878	SB(13, 2) = .0011
2879	SB(14, 2) = .000945
2880	SB(15, 2) = .000833
2881	
2882	SB(0, 3) = .0362
2883	SB(1, 3) = .02
2884	SB(2, 3) = .0121
2885	SB(3, 3) = .00746
2886	SB(4, 3) = .00574
2887	SB(5, 3) = .00468
2888	SB(6, 3) = .00378
2889	SB(7, 3) = .00322
2890	SB(8, 3) = .00274

2891	SB(9, 3) = .00221
2892	SB(10, 3) = .00188
2893	SB(11, 3) = .0016
2894	SB(12, 3) = .00129
2895	SB(13, 3) = .0011
2896	SB(14, 3) = .000945
2897	SB(15, 3) = .000833
2898	
2899	
2900	SBB(0, 0) = .0341
2901	SBB(1, 0) = .0185
2902	SBB(2, 0) = .0111
2903	SBB(3, 0) = .00685
2904	SBB(4, 0) = .00528
2905	SBB(5, 0) = .00432
2906	SBB(6, 0) = .00349
2907	SBB(7, 0) = .00297
2908	SBB(8, 0) = .00253
2909	SBB(9, 0) = .00204
2910	SBB(10, 0) = .00176
2911	SBB(11, 0) = .00152
2912	SBB(12, 0) = .00126
2913	SBB(13, 0) = .00109
2914	SBB(14, 0) = .000945
2915	SBB(15, 0) = .000833
2916	
2917	SBB(0, 1) = .0319
2918	SBB(1, 1) = .017
2919	SBB(2, 1) = .0101
2920	SBB(3, 1) = .00627
2921	SBB(4, 1) = .00484
2922	SBB(5, 1) = .00397
2923	SBB(6, 1) = .00321
2924	SBB(7, 1) = .00273
2925	SBB(8, 1) = .00233
2926	SBB(9, 1) = .00188
2927	SBB(10, 1) = .00162
2928	SBB(11, 1) = .00143
2929	SBB(12, 1) = .00121
2930	SBB(13, 1) = .00107
2931	SBB(14, 1) = .000945
2932	SBB(15, 1) = .000833
2933	
2934	SBB(0, 2) = .0297
2935	SBB(1, 2) = .0155
2936	SBB(2, 2) = .0092
2937	SBB(3, 2) = .0057
2938	SBB(4, 2) = .00442
2939	SBB(5, 2) = .00364
2940	SBB(6, 2) = .00294
2941	SBB(7, 2) = .0025
2942	SBB(8, 2) = .00213
2943	SBB(9, 2) = .00172
2944	SBB(10, 2) = .00149
2945	SBB(11, 2) = .00132
2946	SBB(12, 2) = .00115
2947	SBB(13, 2) = .00104
2948	SBB(14, 2) = .000945
2949	SBB(15, 2) = .000833
2950	
2951	SBB(0, 3) = .0274
2952	SBB(1, 3) = .014
2953	SBB(2, 3) = .00829

2954	SBB(3, 3) = .00517
2955	SBB(4, 3) = .00402
2956	SBB(5, 3) = .00332
2957	SBB(6, 3) = .00268
2958	SBB(7, 3) = .00228
2959	SBB(8, 3) = .00195
2960	SBB(9, 3) = .00157
2961	SBB(10, 3) = .00136
2962	SBB(11, 3) = .00121
2963	SBB(12, 3) = .00106
2964	SBB(13, 3) = .000975
2965	SBB(14, 3) = .000907
2966	SBB(15, 3) = .000833
2967	
2968	
2969	SC(0, 0) = .0387
2970	SC(1, 0) = .0273
2971	SC(2, 0) = .0202
2972	SC(3, 0) = .0137
2973	SC(4, 0) = .0103
2974	SC(5, 0) = .0077
2975	SC(6, 0) = .00553
2976	SC(7, 0) = .00427
2977	SC(8, 0) = .00343
2978	SC(9, 0) = .00253
2979	SC(10, 0) = .00207
2980	SC(11, 0) = .00165
2981	SC(12, 0) = .00133
2982	SC(13, 0) = .0011
2983	SC(14, 0) = .000945
2984	SC(15, 0) = .000833
2985	
2986	SC(0, 1) = .0359
2987	SC(1, 1) = .02
2988	SC(2, 1) = .0123
2989	SC(3, 1) = .00772
2990	SC(4, 1) = .00597
2991	SC(5, 1) = .00489
2992	SC(6, 1) = .00393
2993	SC(7, 1) = .00333
2994	SC(8, 1) = .00282
2995	SC(9, 1) = .00226
2996	SC(10, 1) = .00191
2997	SC(11, 1) = .00162
2998	SC(12, 1) = .0013
2999	SC(13, 1) = .0011
3000	SC(14, 1) = .000945
3001	SC(15, 1) = .000833
3002	
3003	SC(0, 2) = .0335
3004	SC(1, 2) = .0184
3005	SC(2, 2) = .0113
3006	SC(3, 2) = .00706
3007	SC(4, 2) = .00548
3008	SC(5, 2) = .0045
3009	SC(6, 2) = .00364
3010	SC(7, 2) = .0031
3011	SC(8, 2) = .00264
3012	SC(9, 2) = .00213
3013	SC(10, 2) = .00183
3014	SC(11, 2) = .00157
3015	SC(12, 2) = .00128
3016	SC(13, 2) = .0011

3017	SC(14, 2) = .000945
3018	SC(15, 2) = .000833
3019	
3020	SC(0, 3) = .031
3021	SC(1, 3) = .0168
3022	SC(2, 3) = .0102
3023	SC(3, 3) = .00643
3024	SC(4, 3) = .00501
3025	SC(5, 3) = .00413
3026	SC(6, 3) = .00334
3027	SC(7, 3) = .00284
3028	SC(8, 3) = .00242
3029	SC(9, 3) = .00196
3030	SC(10, 3) = .00169
3031	SC(11, 3) = .00148
3032	SC(12, 3) = .00124
3033	SC(13, 3) = .00108
3034	SC(14, 3) = .000945
3035	SC(15, 3) = .000833
3036	
3037	
3038	SCC(0, 0) = .0285
3039	SCC(1, 0) = .0152
3040	SCC(2, 0) = .00921
3041	SCC(3, 0) = .00583
3042	SCC(4, 0) = .00456
3043	SCC(5, 0) = .00377
3044	SCC(6, 0) = .00305
3045	SCC(7, 0) = .0026
3046	SCC(8, 0) = .00221
3047	SCC(9, 0) = .00178
3048	SCC(10, 0) = .00155
3049	SCC(11, 0) = .00138
3050	SCC(12, 0) = .00118
3051	SCC(13, 0) = .00106
3052	SCC(14, 0) = .000945
3053	SCC(15, 0) = .000833
3054	
3055	SCC(0, 1) = .026
3056	SCC(1, 1) = .0136
3057	SCC(2, 1) = .00825
3058	SCC(3, 1) = .00525
3059	SCC(4, 1) = .00413
3060	SCC(5, 1) = .00343
3061	SCC(6, 1) = .00277
3062	SCC(7, 1) = .00236
3063	SCC(8, 1) = .00201
3064	SCC(9, 1) = .00163
3065	SCC(10, 1) = .00141
3066	SCC(11, 1) = .00126
3067	SCC(12, 1) = .0011
3068	SCC(13, 1) = .00102
3069	SCC(14, 1) = .000933
3070	SCC(15, 1) = .000833
3071	
3072	SCC(0, 2) = .0235
3073	SCC(1, 2) = .0121
3074	SCC(2, 2) = .00734
3075	SCC(3, 2) = .00471
3076	SCC(4, 2) = .00373
3077	SCC(5, 2) = .0031
3078	SCC(6, 2) = .00251
3079	SCC(7, 2) = .00214

3080 SCC(8, 2) = .00182  
 3081 SCC(9, 2) = .00148  
 3082 SCC(10, 2) = .00129  
 3083 SCC(11, 2) = .00115  
 3084 SCC(12, 2) = .00101  
 3085 SCC(13, 2) = .000933  
 3086 SCC(14, 2) = .00087  
 3087 SCC(15, 2) = .000803  
 3088  
 3089 SCC(0, 3) = .021  
 3090 SCC(1, 3) = .0106  
 3091 SCC(2, 3) = .00649  
 3092 SCC(3, 3) = .00421  
 3093 SCC(4, 3) = .00335  
 3094 SCC(5, 3) = .0028  
 3095 SCC(6, 3) = .00227  
 3096 SCC(7, 3) = .00193  
 3097 SCC(8, 3) = .00165  
 3098 SCC(9, 3) = .00133  
 3099 SCC(10, 3) = .00117  
 3100 SCC(11, 3) = .00104  
 3101 SCC(12, 3) = .000921  
 3102 SCC(13, 3) = .000851  
 3103 SCC(14, 3) = .000795  
 3104 SCC(15, 3) = .000734  
 3105  
 3106 +++ GAISOU DATA +++ HORYU  
 3107  
 3108 SD(0) = .0376  
 3109 SD(1) = .0283  
 3110 SD(2) = .021  
 3111 SD(3) = .0143  
 3112 SD(4) = .0107  
 3113 SD(5) = .00867  
 3114 SD(6) = .0056  
 3115 SD(7) = .00467  
 3116 SD(8) = .0037  
 3117 SD(9) = .00277  
 3118 SD(10) = .0022  
 3119 SD(11) = .0018  
 3120 SD(12) = .0014  
 3121 SD(13) = .00132  
 3122 SD(14) = .00103  
 3123 SD(15) = .0009  
 3124  
 3125 SE(0) = .038  
 3126 SE(1) = .0287  
 3127 SE(2) = .0213  
 3128 SE(3) = .0147  
 3129 SE(4) = .011  
 3130 SE(5) = .00867  
 3131 SE(6) = .006  
 3132 SE(7) = .00467  
 3133 SE(8) = .0036  
 3134 SE(9) = .00247  
 3135 SE(10) = .002  
 3136 SE(11) = .0016  
 3137 SE(12) = .0012  
 3138 SE(13) = .0011  
 3139 SE(14) = .000933  
 3140 SE(15) = .000867  
 3141  
 3142 +++

```

3143 MATE = 1 --- SUS 304
3144 MATE = 2 --- SUS 316
3145 MATE = 3 --- SUS 321
3146 MATE = 4 --- 2 1/4CR-1MO(NT) BAN
3147 MATE = 5 --- 2 1/4CR-1MO(NT) DEN NETSU KAN
3148 MATE = 6 --- SUS 304 OOGATA-TANKOUHIN
3149 MATE = 7 --- SUS 321 OOGATA-TANKOUHIN
3150
3151 ITYPE = 1 --- (CYCLIC STRAIN RATE). GE. 1. E-03 (MM/MM/SEC)
3152 ITYPE = 2 --- (CYCLIC STRAIN RATE)=(THE OTHER CASES)
3153 ITYPE = 3 --- (CYCLIC STRAIN RATE). GE. 1. E-04 (MM/MM/SEC)
3154 ITYPE = 4 --- (LOW TEMPERATURE DESIGN FATIGUE CURVE)
3155 ITYPE = 5 --- (LOW TEMPERATURE DESIGN FATIGUE CURVE, 79S ADDENDA)
3156
3157 If STN = 0! Then GoTo N950
3158 If STN < 0! Then GoTo N960
3159 If MATE = 4# Or MATE = 5# Then GoTo N1200
3160
3161 For I = 0 To 9
3162     T(I) = T1(I)
3163 Next I
3164 K = 9#
3165 GoTo N1300
3166
3167 N1200:
3168 For I = 0 To 7
3169     T(I) = T2(I)
3170 Next I
3171 K = 7#
3172
3173 N1300:
3174 If ITYPE > 3# Then K = 0#
3175
3176 Select Case MATE
3177     Case 1
3178         GoTo N1500
3179     Case 2
3180         GoTo N2000
3181     Case 3
3182         GoTo N1500
3183     Case 4
3184         GoTo N3000
3185     Case 5
3186         GoTo N3000
3187     Case 6
3188         GoTo N1500
3189     Case 7
3190         GoTo N1500
3191 End Select
3192
3193 N1500:
3194 Select Case ITYPE
3195     Case 1
3196         GoTo N1510
3197     Case 2
3198         GoTo N1600
3199     Case 3
3200         GoTo N1700
3201     Case 4
3202         GoTo N1800
3203     Case 5
3204         GoTo N1900
3205 End Select

```

\* 配列添字 : 9番目

\* 配列添字 : 7番目

\* 材料番号判定

\* MATE = 1 -- SUS 304

\* MATE = 2 -- SUS 316

\* MATE = 3 -- SUS 321

\* MATE = 4 -- 2 1/4CR-1MO(NT) BAN

\* MATE = 5 -- 2 1/4CR-1MO(NT) DENNETSUKAN

\* MATE = 6 --- SUS 304 OOGATA-TANKOUHIN

\* MATE = 7 --- SUS 321 OOGATA-TANKOUHIN



```

3206
3207 N1510:
3208     For I = 0 To 4
3209         For J = 0 To 15
3210             S(J, I) = S1(J, I)
3211         Next J
3212     Next I
3213
3214     For I = 0 To 4
3215         For J = 0 To 15
3216             S(J, I + 5) = S11(J, I)
3217         Next J
3218     Next I
3219
3220     GoTo N2700
3221
3222 N1600:
3223     For I = 0 To 4
3224         For J = 0 To 15
3225             S(J, I) = S2(J, I)
3226         Next J
3227     Next I
3228
3229     For I = 0 To 4
3230         For J = 0 To 15
3231             S(J, I + 5) = S22(J, I)
3232         Next J
3233     Next I
3234
3235     GoTo N2700
3236
3237 N1700:
3238     For I = 0 To 4
3239         For J = 0 To 15
3240             S(J, I) = S3(J, I)
3241         Next J
3242     Next I
3243
3244     For I = 0 To 4
3245         For J = 0 To 15
3246             S(J, I + 5) = S33(J, I)
3247         Next J
3248     Next I
3249
3250     GoTo N2700
3251
3252 N1800:
3253     For J = 0 To 15
3254         S(J, 1) = S4(J)
3255     Next J
3256
3257     GoTo N90
3258
3259 N1900:
3260     For J = 0 To 15
3261         S(J, 1) = S5(J)
3262     Next J
3263
3264     GoTo N90
3265
3266 N2000:
3267     Select Case ITYPE
3268         Case 1             ITYPE = 1 - (CYCLIC STRAIN RATE).GE. 1.E-03 (MM/MM/SEC)

```

```

3269      GoTo N2100
3270      Case 2      ' ITYPE = 2 - (CYCLIC STRAIN RATE)=(THE OTHER CASES)
3271      GoTo N2200
3272      Case 3      ' ITYPE = 3 - (CYCLIC STRAIN RATE). GE. 1. E-04(MM/MM/SEC)
3273      GoTo N2300
3274      Case 4      ' ITYPE = 4 - (LOW TEMPERATURE DESIGN FATIGUE CURVE)
3275      GoTo N1800
3276      Case 5      ' ITYPE = 5 - (LOW TEMPERATURE DESIGN FATIGUE CURVE, 79S ADDENDA)
3277      GoTo N1900
3278      End Select
3279
3280 N2100:
3281      For I = 0 To 4
3282          For J = 0 To 15
3283              S(J, I) = S6(J, I)
3284          Next J
3285      Next I
3286
3287      For I = 0 To 4
3288          For J = 0 To 15
3289              S(J, I + 5) = S66(J, I)
3290          Next J
3291      Next I
3292
3293      GoTo N2700
3294
3295 N2200:
3296      For I = 0 To 4
3297          For J = 0 To 15
3298              S(J, I) = S7(J, I)
3299          Next J
3300      Next I
3301
3302      For I = 0 To 4
3303          For J = 0 To 15
3304              S(J, I + 5) = S77(J, I)
3305          Next J
3306      Next I
3307
3308      GoTo N2700
3309
3310 N2300:
3311      For I = 0 To 4
3312          For J = 0 To 15
3313              S(J, I) = S8(J, I)
3314          Next J
3315      Next I
3316
3317      For I = 0 To 4
3318          For J = 0 To 15
3319              S(J, I + 5) = S88(J, I)
3320          Next J
3321      Next I
3322
3323
3324 N2700:
3325      If Temp <= 650 Then GoTo N90
3326      JJ = K
3327      II = 0
3328      GoTo N500
3329
3330 N3000:
3331      Select Case ITYPE

```

```

3332 Case 1 ' ITYPE = 1 - (CYCLIC STRAIN RATE). GE. 1. E-03 (MM/MM/SEC)
3333 GoTo N3100
3334 Case 2 ' ITYPE = 2 - (CYCLIC STRAIN RATE)=(THE OTHER CASES)
3335 GoTo N3300
3336 Case 3 ' ITYPE = 3 - (CYCLIC STRAIN RATE). GE. 1. E-04(MM/MM/SEC)
3337 GoTo N3500
3338 Case 4 ' ITYPE = 4 - (LOW TEMPERATURE DESIGN FATIGUE CURVE)
3339 GoTo N3700
3340 Case 5 ' ITYPE = 5 - (LOW TEMPERATURE DESIGN FATIGUE CURVE, 79S ADDENDA)
3341 GoTo N3900
3342 End Select
3343
3344 N3100:
3345 For I = 0 To 3
3346 For J = 0 To 15
3347 S(J, I) = SA(J, I)
3348 Next J
3349 Next I
3350
3351 For I = 0 To 3
3352 For J = 0 To 15
3353 S(J, I + 4) = SAA(J, I)
3354 Next J
3355 Next I
3356
3357 GoTo N4100
3358
3359 N3300:
3360 For I = 0 To 3
3361 For J = 0 To 15
3362 S(J, I) = SB(J, I)
3363 Next J
3364 Next I
3365
3366 For I = 0 To 3
3367 For J = 0 To 15
3368 S(J, I + 4) = SBB(J, I)
3369 Next J
3370 Next I
3371
3372 GoTo N4100
3373
3374 N3500:
3375 For I = 0 To 3
3376 For J = 0 To 15
3377 S(J, I) = SC(J, I)
3378 Next J
3379 Next I
3380
3381 For I = 0 To 3
3382 For J = 0 To 15
3383 S(J, I + 4) = SCC(J, I)
3384 Next J
3385 Next I
3386
3387 GoTo N4100
3388
3389 N3700:
3390 For J = 0 To 15
3391 S(J, I) = SD(J)
3392 Next J
3393
3394 GoTo N90

```

```

3395 '
3396 N3900:
3397     For J = 0 To 15
3398         S(J, 1) = SE(J)
3399     Next J
3400
3401     GoTo N90
3402 '
3403 N4100:
3404     If Temp <= 550 Then GoTo N90
3405     JJ = K
3406     II = 0
3407     GoTo N500
3408 N90:
3409 '
3410     For I = 0 To K
3411         If T(I) > Temp Then GoTo N200
3412     Next I
3413
3414     I = K
3415 N200:
3416     JJ = I
3417     II = I - 1
3418     If II = -1 Then GoTo N500
3419     For I = 1 To 15
3420         If S(I, II) < STN Then GoTo N400
3421     Next I
3422
3423     I = 15#
3424 N400:
3425     LL = I
3426     KK = I - 1
3427 '
3428     Res = Debug_disp("[Dfcurv]/I=" & Str(I) & "/STN=" & Str(STN))
3429     'for Debug
3430     If Res = 9 Then Exit Sub
3431 '
3432     If KK = -1 Then GoTo N900
3433     X1 = CSng(N(KK))
3434     X2 = CSng(N(LL))
3435     Y1 = S(KK, II)
3436     Y2 = S(LL, II)
3437     A = ALog10(Y1 / Y2) / ALog10(X1 / X2)
3438     B = 10! ^ (ALog10(Y1) - A * ALog10(X1))
3439     If (ALog10(STN / B) / A) > 10! Then GoTo N450
3440     FN1 = 10# ^ (ALog10(STN / B) / A)
3441     GoTo N500
3442 N450:
3443     FN1 = 1E+10!
3444 N500:
3445     For I = 1 To 15
3446         If S(I, JJ) < STN Then GoTo N700
3447     Next I
3448
3449     I = 15#
3450 N700:
3451     LL = I
3452     KK = I - 1
3453     If KK = -1 Then GoTo N900
3454     X1 = CSng(N(KK))
3455     X2 = CSng(N(LL))
3456     Y1 = S(KK, JJ)
3457     Y2 = S(LL, JJ)

```

```

3458      A = ALog10(Y1 / Y2) / ALog10(X1 / X2)
3459      B = 10! ^ (ALog10(Y1) - A * ALog10(X1))
3460      If ALog10(STN / B) / A > 10! Then GoTo N750
3461      FN2 = 10! ^ (ALog10(STN / B) / A)
3462      GoTo N760
3463 N750:
3464      FN2 = 1E+10!
3465 N760:
3466      If II = -1 Then GoTo N800
3467      DT1 = Temp - T(II)
3468      DT2 = T(JJ) - Temp
3469      FD = 10! ^ ((ALog10(FN1) * DT2 + ALog10(FN2) * DT1) / (DT1 + DT2))
3470      GoTo N850
3471 N800:
3472      FD = FN2
3473 N850:
3474      If FD >= 1E+08! Then FD = 1E+08!
3475      NN = Int(ALog10(FD)) + 1
3476      M = NN - 3
3477      If M <= 0# Then M = 0#
3478      ND = Int(FD / 10! ^ M) * 10! ^ M
3479
3480      Res = Debug_disp("[Dfcurve]/ND=" & Str(ND) & "/M=" & Str(M) & "/NN=" & Str(NN))' for Debug
3481      If Res = 9 Then Exit Sub
3482
3483      If ND < 1# Then ND = 1#
3484      GoTo N1000
3485 N900:
3486      ND = 1#
3487      GoTo N1000
3488 N950:
3489      ND = 100000000#
3490      GoTo N1000
3491 N960:
3492      ND = 1#
3493      WRITE(10, 6000) STN
3494 N6000:
3495      FORMAT(1H0, 40X, 16HERROR --- STN = ,F7.5, 8H < 0.0 )
3496 N1000:
3497
3498 End Sub
3499
3500
3501      DDS - WELD
3502
3503      損傷計算結果データ -> Grid.Text7°ロハ°テ代入
3504
3505 Sub Dwdout (Icount)
3506   Dim Icount As Integer
3507   Dim I      As Integer
3508
3509   For I = 0 To Icount - 1
3510     Form1!Grid1.Row = I + 1
3511     Form1!Grid1.Col = 1
3512     Form1!Grid1.Text = Analy.R(I, 0)
3513     Form1!Grid1.Col = 2
3514     Form1!Grid1.Text = Analy.Z(I, 0)
3515     Form1!Grid1.Col = 3
3516     Form1!Grid1.Text = Analy.H(I, 0)
3517     Form1!Grid1.Col = 4
3518     Form1!Grid1.Text = Allmtd.Spx(I)
3519     Form1!Grid1.Col = 5
3520     Form1!Grid1.Text = Allmtd.Sn(I)

```

```

3521      Forml!Gridl. Col = 6
3522      Forml!Gridl. Text = Allmtd. K(I)
3523      Forml!Gridl. Col = 7
3524      Forml!Gridl. Text = Allmtd. Ked(I)
3525      Forml!Gridl. Col = 8
3526      Forml!Gridl. Text = Allmtd. Neuber(I)
3527      Forml!Gridl. Col = 9
3528      Forml!Gridl. Text = Allmtd. Ke(I)
3529      Forml!Gridl. Col = 10
3530      Forml!Gridl. Text = Allmtd. Ef(I)
3531      Forml!Gridl. Col = 11
3532      Forml!Gridl. Text = Allmtd. Et(I)
3533      Forml!Gridl. Col = 12
3534      Forml!Gridl. Text = Allmtd. Si(I)
3535      Forml!Gridl. Col = 13
3536      Forml!Gridl. Text = Allmtd. Qbm(I)
3537      Forml!Gridl. Col = 14
3538      Forml!Gridl. Text = Allmtd. Qw(I)
3539      Forml!Gridl. Col = 15
3540      Forml!Gridl. Text = Allmtd. Qeff(I)
3541      Forml!Gridl. Col = 16
3542      Forml!Gridl. Text = Allmtd. Df(I)
3543      Forml!Gridl. Col = 17
3544      Forml!Gridl. Text = Allmtd. Dc(I)
3545      Forml!Gridl. Col = 18
3546      Forml!Gridl. Text = Allmtd. D(I)
3547      Next I
3548
3549 End Sub
3550
3551 '
3552 '
3553 '
3554 Function Ecrr (S, T, ALPHC)
3555     Ecrr = 0
3556     If S = 0 Then
3557         Exit Function
3558     End If
3559     Iss = 1
3560     If S < 0! Then
3561         Iss = -1
3562     End If
3563     z = Iss * S
3564     TR = Tcrr(z) / ALPHC
3565     EM = Mc. F * TR ^ Mc. Ramda
3566     R1 = Mc. S1 * TR ^ Mc. Alph1
3567     R2 = Mc. S2 * TR ^ Mc. Alph2
3568     C1 = Mc. G1 * EM ^ Mc. Beta1 / R1
3569     C2 = Mc. G2 * EM ^ Mc. Beta2 / R2
3570     Ecrr = Iss * EM
3571     If R2 * T > 60! Then
3572         Exit Function
3573     End If
3574     Ecrr = Iss * (C2 * R2 * Exp(-R2 * T) + EM)
3575     If R1 * T > 60! Then
3576         Exit Function
3577     End If
3578     Ecrr = Iss * (C1 * R1 * Exp(-R1 * T) + C2 * R2 * Exp(-R2 * T) + EM)
3579 End Function
3580
3581 '
3582 ' 歪範囲算出プログラム
3583 '

```

```

3584 Function Eeq (R, z, H, T)
3585   Dim R, Z, H, T      As Single
3586
3587   Eeq = 1! / 3! * Sqr(2! * ((R - z) ^ 2 + (z - H) ^ 2 + (H - R) ^ 2 + 3! / 2! * T ^ 2))
3588
3589 End Function
3590
3591
3592   EQEF
3593   (中間保持考慮、弾性追従パラメータ最適化、ひずみ集中係数改良)
3594   平成3年11月21日   Ver. 1. 5
3595
3596 Sub Egef15p (Icount)
3597
3598   Dim Icount As Integer
3599   Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
3600   Dim KEDD As Single
3601   Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
3602   Dim K1, K2, KE1, KE2 As Single
3603   Dim K1D, K2D, KED1D, KED2D As Single
3604   Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
3605   Dim LSIG1, LSIG2, LSIG3 As Single
3606   Dim NU, NU1, NU2, NF      As Single
3607
3608   Iflg = 0
3609
3610   For I = 0 To Icount - 1
3611   -----
3612     歪範囲の計算
3613
3614   -----
3615     TIME 1
3616     Mc.Tc = Analy.Mtemp(I, 0)
3617     Call Mset(Iflg, Mc)
3618     E1 = Mc.E
3619
3620     NU1 = Calcnu(Mc)
3621     ALPHA1 = Alpha(Mc)
3622     TIME 2
3623     Mc.Tc = Analy.Mtemp(I, 1)
3624     Call Mset(Iflg, Mc)
3625     E2 = Mc.E
3626     NU2 = Calcnu(Mc)
3627     ALPHA2 = Alpha(Mc)
3628
3629     熱ピークを含むピーク応力強さの計算
3630     SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
3631     SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
3632     SIGH = Analy.Strs(I, 2, 0) - Analy.Strs(I, 2, 1)
3633     SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
3634     SIG1 = Prin1(SIGR, SIGZ, SIGT)
3635     SIG2 = Prin2(SIGR, SIGZ, SIGT)
3636     SIG3 = SIGH
3637
3638     SPP = Ftres(SIG1, SIG2, SIG3)
3639
3640     熱ピークを含まないピーク応力強さの計算
3641     SF1 = (-E1 * ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0))) / (1 - NU1)
3642     SF2 = (-E2 * ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2)
3643     Allmtd.Ef(I) = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *
3644     (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
3645
3645     SANR = Abs(Analy.Strs(I, 0, 1) - Analy.Strs(I, 0, 0))

```

```

3646 SANZ = Abs(Analy.Strs(I, 1, 1) - Analy.Strs(I, 1, 0))
3647 If SANR >= SANZ Then
3648     SIGR = (Analy.Strs(I, 0, 0) - SF1) - (Analy.Strs(I, 0, 1) - SF2)
3649     SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
3650 Else
3651     SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
3652     SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)
3653 End If
3654
3655 SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
3656 SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
3657 SIG1 = Prin1(SIGR, SIGZ, SIGT)
3658 SIG2 = Prin2(SIGR, SIGZ, SIGT)
3659 SIG3 = SIGH
3660
3661 Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
3662 '-----公称応力強さの計算
3663 Allmtd.Sn(I) = Bscval.Ltres(I)
3664 '-----ピーク応力を含む応力集中係数の計算
3665 If Allmtd.Sn(I) = 0! Then
3666     KS = 0!
3667 Else
3668     KS = SPP / Allmtd.Sn(I)
3669 End If
3670
3671 KSTAR = Smax(KS, 1!, 1!)
3672 '-----ピーク応力を含まない応力集中係数の計算
3673 If Allmtd.Sn(I) = 0! Then
3674     KD = 0!
3675 Else
3676     KD = Allmtd.Spx(I) / Allmtd.Sn(I)
3677 End If
3678
3679 Allmtd.K(I) = Smax(KD, 1!, 1!)
3680
3681 '-----最高金属温度によるεn、Smh、Srhの計算
3682 Mc.Tc = Analy.Tmpmax(I)
3683 Call Mset(Iflg, Mc)
3684 EN = Allmtd.Sn(I) / Mc.E
3685 SMH = Mc.Sm
3686
3687 Call R1x1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmh)
3688 Call R1x1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmh, S2xx(0), Dcxx(0), Mc)
3689
3690 '-----コールド温度によるSmc、Srcの計算
3691 Mc.Tc = Analy.Temp(I, 1)
3692 Call Mset(Iflg, Mc)
3693 SMC = Mc.Sm
3694
3695 Call R1x1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmc)
3696 Call R1x1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmc, S2xx(1), Dcxx(1), Mc)
3697
3698 '-----ひずみ集中係数の計算
3699 SM3BAR = S2xx(0) + S2xx(1)
3700 Allmtd.Qn(I) = Analy.Q(I)
3701 NU = .3
3702
3703 Allmtd.Qnu(I) = 2! * (1! - NU)
3704 QNUSTR = 5! / 3!
3705 S1 = Smax(KSTAR ^ (3! / 4!), Allmtd.Qnu(I), Allmtd.Qnu(1))
3706 S2 = Smax(Allmtd.K(I) ^ (3! / 4!), QNUSTR, QNUSTR)
3707 S3 = Smax(Allmtd.K(I) ^ (3! / 4!), QNUSTR, QNUSTR)
3708 Allmtd.Ql(I) = Nmin(S1, S2, S3)

```



```

3709      If Allmtd. Sn(I) = 0! Then
3710          KEDD = 0!
3711      Else
3712          KEDD = SM3BAR / Allmtd. Sn(I)
3713      End If
3714
3715      Allmtd. Kedl(I) = Smax(1! + (Allmtd. Ql(I) - 1!) * (1! - KEDD / KSTAR), 1!, 1!)
3716      Allmtd. Kedg(I) = Smax(1! + (Allmtd. Qn(I) - 1!) * (1! - KEDD), 1!, 1!)
3717
3718      Allmtd. Ke(I) = KSTAR * Allmtd. Kedl(I) * Allmtd. Kedg(I)
3719
3720      Allmtd. Et(I) = Allmtd. Ke(I) * EN
3721
3722
3723      疲労損傷の計算
3724
3725      -----ファンクションANFの計算機能の限界
3726      If Allmtd. Et(I) < .0005 Then Allmtd. Et(I) = .0005
3727
3728      -----最高金属温度
3729      Mc. Tc = Analy. Tmpmax(I)
3730      Call Mset(lflg, Mc)
3731
3732      EDOT = 2! * Allmtd. Et(I) / ((Cond. Hldtmc + Cond. Hldtmh) * 3600!)
3733      NF = Anf(Allmtd. Et(I), EDOT, Mc)
3734
3735      Allmtd. Df(I) = Cond. Cycle / NF
3736
3737      クリープ損傷の計算
3738
3739      -----DCNの計算
3740      DCN = 0
3741
3742      -----DCPの計算
3743      -----初期応力の計算
3744      TIME 1
3745
3746
3747      熱ピークを含むピーク応力強さの計算
3748      SIGR = Analy. Strs(I, 0, 0)
3749      SIGZ = Analy. Strs(I, 1, 0)
3750      SIGH = Analy. Strs(I, 2, 0)
3751      SIGT = Analy. Strs(I, 3, 0)
3752      SIG1 = Prin1(SIGR, SIGZ, SIGT)
3753      SIG2 = Prin2(SIGR, SIGZ, SIGT)
3754      SIG3 = SIGH
3755
3756      SPP1 = Ftres(SIG1, SIG2, SIG3)
3757
3758      熱ピークを含まないピーク応力強さの計算
3759      EF1 = Abs(ALPHA1 * (Analy. Temp(I, 0) - Analy. Ltemp(I, 0)) / (1 - NU1))
3760      If Analy. Strs(I, 0, 0) >= Analy. Strs(I, 1, 0) Then
3761          SIGR = Analy. Strs(I, 0, 0) - SF1
3762          SIGZ = Analy. Strs(I, 1, 0)
3763      Else
3764          SIGR = Analy. Strs(I, 0, 0)
3765          SIGZ = Analy. Strs(I, 1, 0) - SF1
3766      End If
3767
3768      SIGH = Analy. Strs(I, 2, 0) - SF1
3769      SIGT = Analy. Strs(I, 3, 0)
3770      SIG1 = Prin1(SIGR, SIGZ, SIGT)
3771      SIG2 = Prin2(SIGR, SIGZ, SIGT)

```

```

3772     SIG3 = SiGH
3773
3774     SP1 = Ftres(SIG1, SIG2, SIG3)
3775 -----公称応力強さの計算
3776     LSIGR = Analy.Lstrs(I, 0, 0)
3777     LSIGZ = Analy.Lstrs(I, 1, 0)
3778     LSIGH = Analy.Lstrs(I, 2, 0)
3779     LSIGT = Analy.Lstrs(I, 3, 0)
3780     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
3781     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
3782     LSIG3 = LSIGH
3783     SN1 = Ftres(LSIG1, LSIG2, LSIG3)
3784     EN1 = SN1 / E1
3785
3786     ピーク応力を含む応力集中係数の計算
3787     If SN1 = 0! Then
3788         KS1 = 0!
3789     Else
3790         KS1 = SPP1 / SN1
3791     End If
3792
3793     KSTAR1 = Smax(KS1, 1!, 1!)
3794     ピーク応力を含まない応力集中係数の計算
3795     If SN1 = 0! Then
3796         K1D = 0!
3797     Else
3798         K1D = SP1 / SN1
3799     End If
3800
3801     K1 = Smax(K1D, 1!, 1!)
3802     ひずみ集中係数の計算
3803     SM3 = 1.5 * SMH + 1.5 * SMC
3804     If SN1 = 0! Then
3805         KED1D = 0!
3806     Else
3807         KED1D = SM3 / SN1
3808     End If
3809
3810     KEDL1 = Smax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED1D / KSTAR1), 1!, 1!)
3811     KEDG1 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED1D), 1!, 1!)
3812
3813     KE1 = KSTAR1 * KEDL1 * KEDG1
3814     ETD1 = KE1 * EN1
3815 ----- T I M E 2
3816
3817     熱ピークを含むピーク応力強さの計算
3818     SIGR = Analy.Strs(I, 0, 1)
3819     SIGZ = Analy.Strs(I, 1, 1)
3820     SIGH = Analy.Strs(I, 2, 1)
3821     SIGT = Analy.Strs(I, 3, 1)
3822     SIG1 = Prin1(SIGR, SIGZ, SIGT)
3823     SIG2 = Prin2(SIGR, SIGZ, SIGT)
3824     SIG3 = SIGH
3825
3826     SPP2 = Ftres(SIG1, SIG2, SIG3)
3827
3828     熱ピークを含まないピーク応力強さの計算
3829     EF2 = Abs(ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
3830     If Analy.Strs(I, 0, 1) >= Analy.Strs(I, 1, 1) Then
3831         SIGR = Analy.Strs(I, 0, 1) - SF2
3832         SIGZ = Analy.Strs(I, 1, 1)
3833     Else
3834         SIGR = Analy.Strs(I, 0, 1)

```

```

3835     SIGZ = Analy.Strs(I, 1, 1) - SF2
3836     End If
3837
3838     SIGH = Analy.Strs(I, 2, 1) - SF2
3839     SIGT = Analy.Strs(I, 3, 1)
3840     SIG1 = Prin1(SIGR, SIGZ, SIGT)
3841     SIG2 = Prin2(SIGR, SIGZ, SIGT)
3842     SIG3 = SIGH
3843
3844     SP2 = Ftres(SIG1, SIG2, SIG3)
3845     公称応力強さの計算
3846     LSIGR = Analy.Lstrs(I, 0, 1)
3847     LSIGZ = Analy.Lstrs(I, 1, 1)
3848     LSIGH = Analy.Lstrs(I, 2, 1)
3849     LSIGT = Analy.Lstrs(I, 3, 1)
3850     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
3851     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
3852     LSIG3 = LSIGH
3853     SN2 = Ftres(LSIG1, LSIG2, LSIG3)
3854     EN2 = SN2 / E2
3855
3856     ピーク応力を含む応力集中係数の計算
3857     If SN2 = 0! Then
3858         KS2 = 0!
3859     Else
3860         KS2 = SPP2 / SN2
3861     End If
3862
3863     KSTAR2 = Smax(KS2, 1!, 1!)
3864     ピーク応力を含まない応力集中係数の計算
3865     If SN2 = 0! Then
3866         K2D = 0!
3867     Else
3868         K2D = SP2 / SN2
3869     End If
3870
3871     K2 = Smax(K2D, 1!, 1!)
3872     ひずみ集中係数の計算
3873     If SN2 = 0! Then
3874         KED2D = 0!
3875     Else
3876         KED2D = SM3 / SN2
3877     End If
3878     KEDL2 = Smax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED2D / KSTAR2), 1!, 1!)
3879     KEDG2 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED2D), 1!, 1!)
3880
3881     KE2 = KSTAR2 * KEDL2 * KEDG2
3882     ETD2 = KE2 * EN2
3883
3884     ETC = Smax(ETD1, ETD2, ETD2)
3885
3886     Allmtd.Si(I) = Sepr(ETC, Mc)
3887     Allmtd.Si(I) = Sepr(ETC, Mc) / 2!
3888
3889     Allmtd.Si(I) = Allmtd.Si(I) + Cond.Mecha
3890
3891     EC1 = 0!
3892
3893     ALPHC = 1!
3894     ALPHR = 1!
3895     DT = Cond.Hldtmh
3896     Allmtd.Qeff(I) = Allmtd.Q1(I) * Allmtd.Qn(I)
3897     Call Rlx1(Allmtd.Si(I), 0!, 1!, 1!, Allmtd.Qeff(I), Cond.Hldtmh)

```

```

3898 '      Call Rlx1(Allmtd.Si(I), 0. 0. 1. 0. 1. 0, Allmtd. Qeff(I), Cond. Hldtmh, S2xx(0), Dcxx(0), Mc)
3899 '      DCP = Cond. Cycle * Dcxx(0)
3900 '
3901 '      Allmtd. Dc(I) = DCN + DCP
3902 ' -----
3903 '      クリープ疲労損傷の計算
3904 ' -----
3905 '      Allmtd. D(I) = Allmtd. Df(I) + Allmtd. Dc(I)
3906 ' -----
3907 '      許容値の計算
3908 ' -----
3909 '      If Allmtd. Dc(I) >= 0! And Allmtd. Dc(I) <= .3 Then
3910 '          Allmtd. Dcr(I) = 1! - 4! / 3! * Allmtd. Dc(I)
3911 '      ElseIf Allmtd. Dc(I) <= 1! Then
3912 '          Allmtd. Dcr(I) = 3! / 7! + 4! / 7! * Allmtd. Dc(I)
3913 '      ElseIf Allmtd. Dc(I) > 1! Then
3914 '          Allmtd. Dcr(I) = 1!
3915 '      End If
3916 '
3917 '      Next I
3918 '
3919 ' End Sub
3920 '
3921 '
3922 '      EQEF167
3923 '      (中間保持考慮、qL=1. 67、ひずみ集中係数改良 EQEFを改良)
3924 '
3925 ' Sub Eqef167p (Icount)
3926 '     Dim Icount As Integer
3927 '
3928 '     ReDim S2x(1), Dcx(1) As Single
3929 '     Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
3930 '     Dim KEDD As Single
3931 '     Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
3932 '     Dim K1, K2, KE1, KE2 As Single
3933 '     Dim K1D, K2D, KED1D, KED2D As Single
3934 '     Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
3935 '     Dim LSIG1, LSIG2, LSIG3 As Single
3936 '     Dim NU, NU1, NU2, NF          As Single
3937 '
3938 '     Iflg = 0
3939 '
3940 '     For I = 0 To Icount - 1
3941 '
3942 '         歪範囲の計算
3943 '
3944 '         T I M E 1
3945 '         Mc. Tc = Analy. Mtemp(I, 0)
3946 '         Call Mset(Iflg, Mc)
3947 '         E1 = Mc. E
3948 '         NU1 = Calcnu(Mc)
3949 '         ALPHA1 = Alpha(Mc)
3950 '         T I M E 2
3951 '         Mc. Tc = Analy. Mtemp(I, 1)
3952 '         Call Mset(Iflg, Mc)
3953 '         E2 = Mc. E
3954 '         NU2 = Calcnu(Mc)
3955 '         ALPHA2 = Alpha(Mc)
3956 '
3957 '         熱ピークを含むピーク応力強さの計算
3958 '         SIGR = Analy. Strs(I, 0, 0) - Analy. Strs(I, 0, 1)
3959 '         SIGZ = Analy. Strs(I, 1, 0) - Analy. Strs(I, 1, 1)
3960 '         SIGH = Analy. Strs(I, 2, 0) - Analy. Strs(I, 2, 1)

```

```

3961 SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
3962 SIG1 = Prin1(SIGR, SIGZ, SIGT)
3963 SIG2 = Prin2(SIGR, SIGZ, SIGT)
3964 SIG3 = SIGH
3965 SPP = Ftres(SIG1, SIG2, SIG3)
3966
3967 熱ピークを含まないピーク応力強さの計算
3968 SF1 = (-E1 * ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0))) / (1 - NU1)
3969 SF2 = (-E2 * ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2)
3970 Allmtd.Ef(I) = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *
  (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
3971
3972 SANR = Abs(Analy.Strs(I, 0, 1) - Analy.Strs(I, 0, 0))
3973 SANZ = Abs(Analy.Strs(I, 1, 1) - Analy.Strs(I, 1, 0))
3974 If SANR >= SANZ Then
3975     SIGR = (Analy.Strs(I, 0, 0) - SF1) - (Analy.Strs(I, 0, 1) - SF2)
3976     SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
3977 Else
3978     SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
3979     SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)
3980 End If
3981 SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
3982 SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
3983 SIG1 = Prin1(SIGR, SIGZ, SIGT)
3984 SIG2 = Prin2(SIGR, SIGZ, SIGT)
3985 SIG3 = SIGH
3986 Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
3987 公称応力強さの計算
3988 Allmtd.Sn(I) = Bscval.Ltres(I)
3989 ピーク応力を含む応力集中係数の計算
3990 If Allmtd.Sn(I) = 0! Then
3991     KS = 0!
3992 Else
3993     KS = SPP / Allmtd.Sn(I)
3994 End If
3995 KSTAR = Nmax(KS, 1!, 1!)
3996 ピーク応力を含まない応力集中係数の計算
3997 If Allmtd.Sn(I) = 0! Then
3998     KD = 0!
3999 Else
4000     KD = Allmtd.Spx(I) / Allmtd.Sn(I)
4001 End If
4002 Allmtd.K(I) = Nmax(KD, 1!, 1!)
4003
4004 最高金属温度による  $\epsilon_n$ 、 $S_{mh}$ 、 $S_{rh}$  の計算
4005 Mc.Tc = Analy.Tmpmax(I)
4006 Call Mset(Iflg, Mc)
4007
4008 Call Msg_proc0("Eqef167p:" & Str(I))
4009 EN = Allmtd.Sn(I) / Mc.E
4010 SMH = Mc.Sm
4011 Call R1x1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmh, S2x(0), Dcx(0), Mc)
4012 Call R1x1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmh)
4013 S2x1 = S2x(0)
4014 S2x1 = S2xx(0)
4015
4016 コールド温度による  $S_{mc}$ 、 $S_{rc}$  の計算
4017 Mc.Tc = Analy.Temp(I, 1)
4018 Call Mset(Iflg, Mc)
4019 SMC = Mc.Sm
4020 Call R1x1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmc, S2x(0), Dcx(0), Mc)
4021 Call R1x1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmh)
4022

```

```

4023  ひずみ集中係数の計算
4024  SM3BAR = S2x1+S2x(0)
4025  SM3BAR = S2x1 + S2xx(0)
4026  Allmtd. Qn(I) = Q(I)1993. 06. 16
4027  Allmtd. Qn(I) = 1!
4028  NU = .3
4029  Allmtd. Qnu(I) = 2! * (1! - NU)
4030  QNUSTR = 5! / 3!
4031  Allmtd. Q1(I) = MIN(MAX(KSTAR^(3. 0/4. 0), Allmtd. Qnu(I)),      1993. 2. 3
4032  1      MAX(Allmtd. K(I)^(3. 0/4. 0), QNUSTR))1993. 2. 3
4033  Allmtd. Q1(I) = 1. 67
4034  If Allmtd. Sn(I) = 0! Then
4035  KEDD = 0!
4036  Else
4037  KEDD = SM3BAR / Allmtd. Sn(I)
4038  End If
4039  Allmtd. Kedl(I) = Nmax(1! + (Allmtd. Q1(I) - 1!) * (1! - KEDD / KSTAR), 1!, 1!)
4040  Allmtd. Kedg(I) = Nmax(1! + (Allmtd. Qn(I) - 1!) * (1! - KEDD), 1!, 1!)
4041
4042  Allmtd. Ke(I) = KSTAR * Allmtd. Kedl(I) * Allmtd. Kedg(I)
4043
4044  Allmtd. Et(I) = Allmtd. Ke(I) * EN
4045
4046  疲労損傷の計算
4047
4048  ファンクションANFの計算機能の限界
4049  If Allmtd. Et(I) < .0005 Then
4050  Allmtd. Et(I) = .0005
4051  End If
4052
4053  最高金属温度
4054  Mc. Tc = Analy. Tmpmax(I)
4055  Call Mset(Iflg, Mc)
4056
4057  EDOT = 2! * Allmtd. Et(I) / ((Cond. Hldtmc + Cond. Hldtmh) * 3600!)
4058  NF = Anf(Allmtd. Et(I), EDOT, Mc)
4059
4060  Allmtd. Df(I) = Cond. Cycle / NF
4061
4062  クリープ損傷の計算
4063
4064  DCNの計算
4065  DCN = 0
4066  DCPの計算
4067  初期応力の計算
4068  TIME1
4069
4070
4071  熱ピークを含むピーク応力強さの計算
4072  SIGR = Analy. Strs(I, 0, 0)
4073  SIGZ = Analy. Strs(I, 1, 0)
4074  SIGH = Analy. Strs(I, 2, 0)
4075  SIGT = Analy. Strs(I, 3, 0)
4076  SIG1 = Prin1(SIGR, SIGZ, SIGT)
4077  SIG2 = Prin2(SIGR, SIGZ, SIGT)
4078  SIG3 = SIGH
4079  SPP1 = Ftres(SIG1, SIG2, SIG3)
4080
4081  熱ピークを含まないピーク応力強さの計算
4082  EF1 = Abs(ALPHA1 * (Analy. Temp(I, 0) - Analy. Ltemp(I, 0)) / (1 - NU1))
4083  If Analy. Strs(I, 0, 0) >= Analy. Strs(I, 1, 0) Then
4084  SIGR = Analy. Strs(I, 0, 0) - SF1
4085  SIGZ = Analy. Strs(I, 1, 0)

```

```

4086 Else
4087     SIGR = Analy.Strs(I, 0, 0)
4088     SIGZ = Analy.Strs(I, 1, 0) - SF1
4089 End If
4090     SIGH = Analy.Strs(I, 2, 0) - SF1
4091     SIGT = Analy.Strs(I, 3, 0)
4092     SIG1 = Prin1(SIGR, SIGZ, SIGT)
4093     SIG2 = Prin2(SIGR, SIGZ, SIGT)
4094     SIG3 = SIGH
4095     SP1 = Ftres(SIG1, SIG2, SIG3)
4096 公称応力強さの計算
4097     LSIGR = Analy.Lstrs(I, 0, 0)
4098     LSIGZ = Analy.Lstrs(I, 1, 0)
4099     LSIGH = Analy.Lstrs(I, 2, 0)
4100     LSIGT = Analy.Lstrs(I, 3, 0)
4101     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
4102     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
4103     LSIG3 = LSIGH
4104     SN1 = Ftres(LSIG1, LSIG2, LSIG3)
4105     EN1 = SN1 / E1
4106
4107     ピーク応力を含む応力集中係数の計算
4108     If SN1 = 0! Then
4109         KS1 = 0!
4110     Else
4111         KS1 = SPP1 / SN1
4112     End If
4113     KSTAR1 = Nmax(KS1, 1!, 1!)
4114     ピーク応力を含まない応力集中係数の計算
4115     If SN1 = 0! Then
4116         K1D = 0!
4117     Else
4118         K1D = SP1 / SN1
4119     End If
4120     K1 = Nmax(K1D, 1!, 1!)
4121     ひずみ集中係数の計算
4122     SM3 = 1.5 * SMH + 1.5 * SMC
4123     If SN1 = 0! Then
4124         KED1D = 0!
4125     Else
4126         KED1D = SM3 / SN1
4127     End If
4128     KEDL1 = Nmax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED1D / KSTAR1), 1!, 1!)
4129     KEDG1 = Nmax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED1D), 1!, 1!)
4130
4131     KE1 = KSTAR1 * KEDL1 * KEDG1
4132     ETD1 = KE1 * EN1
4133     T I M E 2
4134
4135     熱ピークを含むピーク応力強さの計算
4136     SIGR = Analy.Strs(I, 0, 1)
4137     SIGZ = Analy.Strs(I, 1, 1)
4138     SIGH = Analy.Strs(I, 2, 1)
4139     SIGT = Analy.Strs(I, 3, 1)
4140     SIG1 = Prin1(SIGR, SIGZ, SIGT)
4141     SIG2 = Prin2(SIGR, SIGZ, SIGT)
4142     SIG3 = SIGH
4143     SPP2 = Ftres(SIG1, SIG2, SIG3)
4144
4145     熱ピークを含まないピーク応力強さの計算
4146     EF2 = Abs(ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
4147     If Analy.Strs(I, 0, 1) >= Analy.Strs(I, 1, 1) Then
4148         SIGR = Analy.Strs(I, 0, 1) - SF2

```

```

4149     SIGZ = Analy.Strs(I, 1, 1)
4150     Else
4151         SIGR = Analy.Strs(I, 0, 1)
4152         SIGZ = Analy.Strs(I, 1, 1) - SF2
4153     End If
4154     SIGH = Analy.Strs(I, 2, 1) - SF2
4155     SIGT = Analy.Strs(I, 3, 1)
4156     SIG1 = Prin1(SIGR, SIGZ, SIGT)
4157     SIG2 = Prin2(SIGR, SIGZ, SIGT)
4158     SIG3 = SIGH
4159     SP2 = Ftres(SIG1, SIG2, SIG3)
4160     公称応力強さの計算
4161     LSIGR = Analy.Lstrs(I, 0, 1)
4162     LSIGZ = Analy.Lstrs(I, 1, 1)
4163     LSIGH = Analy.Lstrs(I, 2, 1)
4164     LSIGT = Analy.Lstrs(I, 3, 1)
4165     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
4166     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
4167     LSIG3 = LSIGH
4168     SN2 = Ftres(LSIG1, LSIG2, LSIG3)
4169     EN2 = SN2 / E2
4170
4171     ピーク応力を含む応力集中係数の計算
4172     If SN2 = 0! Then
4173         KS2 = 0!
4174     Else
4175         KS2 = SPP2 / SN2
4176     End If
4177     KSTAR2 = Nmax(KS2, 1!, 1!)
4178     ピーク応力を含まない応力集中係数の計算
4179     If SN2 = 0! Then
4180         K2D = 0!
4181     Else
4182         K2D = SP2 / SN2
4183     End If
4184     K2 = Nmax(K2D, 1!, 1!)
4185     ひずみ集中係数の計算
4186     If SN2 = 0! Then
4187         KED2D = 0!
4188     Else
4189         KED2D = SM3 / SN2
4190     End If
4191     KEDL2 = Nmax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED2D / KSTAR2), 1!, 1!)
4192     KEDG2 = Nmax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED2D), 1!, 1!)
4193
4194     KE2 = KSTAR2 * KEDL2 * KEDG2
4195     ETD2 = KE2 * EN2
4196
4197     ETC = Nmax(ETD1, ETD2, ETD2)
4198
4199     Allmtd.Si(I) = Sepr(ETC, Mc)
4200     Allmtd.Si(I) = Sepr(ETC, Mc) / 2!
4201
4202     Allmtd.Si(I) = Allmtd.Si(I) + Cond.Mecha
4203
4204     EC1 = 0!
4205     ALPHC = 1!
4206     ALPHR = 1!
4207     DT = Cond.Hldtmh
4208     Allmtd.Qeff(I) = Allmtd.Q1(I) * Allmtd.Qn(I)
4209     Call Rlx1(Allmtd.Si(I), EC1, ALPHC, ALPHR, Allmtd.Qeff(I), DT, S2x(0), Dcx(0), Mc)
4210     Call Rlx1(Allmtd.Si(I), 0!, 1!, 1!, Allmtd.Qeff(I), Cond.Hldtmh)
4211     DCP = Cond.Cycle*Dcx(0)

```



```

4212      DCP = CSng(Cond. Cycle) * Dcxx(0)
4213
4214      Allmtd. Dc(I) = DCN + DCP
4215
4216      クリープ疲労損傷の計算
4217
4218      Allmtd. D(I) = Allmtd. Df(I) + Allmtd. Dc(I)
4219
4220      許容値の計算
4221
4222      If Allmtd. Dc(I) >= 0! And Allmtd. Dc(I) <= .3 Then
4223          Allmtd. Dcr(I) = 1! - 4! / 3! * Allmtd. Dc(I)
4224      ElseIf Allmtd. Dc(I) <= 1! Then
4225          Allmtd. Dcr(I) = 3! / 7! + 4! / 7! * Allmtd. Dc(I)
4226      ElseIf Allmtd. Dc(I) > 1! Then
4227          Allmtd. Dcr(I) = 1!
4228      End If
4229
4230      Next I
4231
4232 End Sub
4233
4234
4235      EQEF167 - WELD
4236
4237
4238 Sub Eqef16wp (Icount)
4239
4240      Dim Icount As Integer
4241      Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
4242      Dim KEDD As Single
4243      Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
4244      Dim K1, K2, KE1, KE2 As Single
4245      Dim K1D, K2D, KED1D, KED2D As Single
4246      Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
4247      Dim LSIG1, LSIG2, LSIG3 As Single
4248      Dim NU, NU1, NU2, NF As Single
4249
4250      Iflg = 0
4251
4252      For I = 0 To Icount
4253
4254          歪範囲の計算
4255
4256          TIME 1
4257          Mc. Tc = Analy. Wtemp(I, 0)
4258          Call Mset(Iflg, Mc)
4259          E1 = Mc. E
4260          NU1 = Calcnu(Mc)
4261          ALPHA1 = Alpha(Mc)
4262          TIME 2
4263          Mc. Tc = Analy. Wtemp(I, 1)
4264          Call Mset(Iflg, Mc)
4265          E2 = Mc. E
4266          NU2 = Calcnu(Mc)
4267          ALPHA2 = Alpha(Mc)
4268
4269          熱ピークを含むピーク応力強さの計算
4270          SIGR = Analy. Strs(I, 0, 0) - Analy. Strs(I, 0, 1)
4271          SIGZ = Analy. Strs(I, 1, 0) - Analy. Strs(I, 1, 1)
4272          SIGH = Analy. Strs(I, 2, 0) - Analy. Strs(I, 2, 1)
4273          SIGT = Analy. Strs(I, 3, 0) - Analy. Strs(I, 3, 1)
4274          SIGI = Prinl(SIGR, SIGZ, SIGT)

```

```

4275     SIG2 = Prin2(SIGR, SIGZ, SIGT)
4276     SIG3 = SIGH
4277     SPP = Ftres(SIG1, SIG2, SIG3)
4278
4279     熱ピークを含まないピーク応力強さの計算
4280     SF1 = (-E1 * ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0))) / (1 - NU1)
4281     SF2 = (-E2 * ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2)
4282     Allmtd.Ef(I) = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *
      (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
4283
4284     SANR = Abs(Analy.Strs(I, 0, 1) - Analy.Strs(I, 0, 0))
4285     SANZ = Abs(Analy.Strs(I, 1, 1) - Analy.Strs(I, 1, 0))
4286     If SANR >= SANZ Then
4287         SIGR = (Analy.Strs(I, 0, 0) - SF1) - (Analy.Strs(I, 0, 1) - SF2)
4288         SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
4289     Else
4290         SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
4291         SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)
4292     End If
4293     SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
4294     SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
4295     SIG1 = Prin1(SIGR, SIGZ, SIGT)
4296     SIG2 = Prin2(SIGR, SIGZ, SIGT)
4297     SIG3 = SIGH
4298     Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
4299
4300     Allmtd.Sn(I) = Bscval.Ltres(I)
4301
4302     最高金属温度による ε n、S r h の計算
4303     Mc.Tc = Analy.Tmpmax(I)
4304     Call Mset(Iflg, Mc)
4305     EN = Allmtd.Sn(I) / Mc.E
4306
4307     If Allmtd.Sn(I) = 0! Then
4308         KD = 0!
4309     Else
4310         KD = Allmtd.Spx(I) / Allmtd.Sn(I)
4311     End If
4312     Allmtd.K(I) = Smax(KD, 1!, 1!)
4313     Call Rlx1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmh)
4314     Call Rlx1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmh, SRH, DCX)
4315
4316     コールド温度による S r c の計算
4317     Mc.Tc = Analy.Temp(I, 1)
4318     Call Mset(Iflg, Mc)
4319     NU = .3
4320     Allmtd.Qnu(I) = 2! * (1! - NU)
4321     BAR3SM = S2xx(0) + 1.5 * Mc.Sm
4322     Allmtd.Qn(I) = 1!
4323     Allmtd.Qw(I) = 1.5
4324     Allmtd.Ql(I) = 1.67
4325     GAMMAY = .8
4326     If Allmtd.Sn(I) = 0! Then
4327         KEDD = 0!
4328     Else
4329         KEDD = GAMMAY * BAR3SM / Allmtd.Sn(I)
4330     End If
4331     Allmtd.Kedl(I) = Smax(1! + (Allmtd.Ql(I) - 1!) * (1! - KEDD / Allmtd.K(I)), 1!, 1!)
4332     Allmtd.Kedg(I) = Smax(1! + (Allmtd.Qn(I) * Allmtd.Qw(I) - 1!) * (1! - KEDD), 1!, 1!)
4333
4334     ピーク応力を含む応力集中係数の計算
4335     If Allmtd.Sn(I) = 0! Then
4336         KS = 0!

```

```

4337 Else
4338 KS = SPP / Allmtd. Sn(I)
4339 End If
4340 KSTAR = Smax(KS, 1!, 1!)
4341
4342 Allmtd. Ke(I) = Smax(KSTAR * Allmtd. Ked1(I) * Allmtd. Kedg(I), 1.3, 1.3)
4343
4344 Allmtd. Et(I) = Allmtd. Ke(I) * EN
4345
4346 疲労損傷の計算
4347
4348 ファンクションANFの計算機能の限界
4349 If Allmtd. Et(I) < .0005 Then Allmtd. Et(I) = .0005
4350
4351 最高金属温度
4352 Mc. Tc = Analy. Tmpmax(I)
4353 Call Mset(Iflg, Mc)
4354
4355 EDOT = 2! * Allmtd. Et(I) / ((Cond. Hldtmc + Cond. Hldtmh) * 3600!)
4356 NF = Anf(Allmtd. Et(I), EDOT, Mc) 'Fortran Library(DLL)呼び出し
4357
4358 Allmtd. Df(I) = Cond. Cycle / NF
4359
4360 クリープ損傷の計算
4361
4362 DCNの計算
4363 DCN = 0
4364 DCPの計算
4365 初期応力の計算
4366 TIME 1
4367
4368 熱ピークを含まないピーク応力強さの計算
4369 SIGR = Analy. Strs(I, 0, 0)
4370 SIGZ = Analy. Strs(I, 1, 0)
4371 SIGH = Analy. Strs(I, 2, 0)
4372 SIGT = Analy. Strs(I, 3, 0)
4373 SIG1 = Prin1(SIGR, SIGZ, SIGT)
4374 SIG2 = Prin2(SIGR, SIGZ, SIGT)
4375 SIG3 = SIGH
4376 SPP1 = Ftres(SIG1, SIG2, SIG3)
4377
4378 熱ピークを含まないピーク応力強さの計算
4379 EF1 = Abs(ALPHA1 * (Analy. Temp(I, 0) - Analy. Ltemp(I, 0)) / (1 - NU1))
4380 If Analy. Strs(I, 0, 0) >= Analy. Strs(I, 1, 0) Then
4381 SIGR = Analy. Strs(I, 0, 0) - SF1
4382 SIGZ = Analy. Strs(I, 1, 0)
4383 Else
4384 SIGR = Analy. Strs(I, 0, 0)
4385 SIGZ = Analy. Strs(I, 1, 0) - SF1
4386 End If
4387 SIGH = Analy. Strs(I, 2, 0) - SF1
4388 SIGT = Analy. Strs(I, 3, 0)
4389 SIG1 = Prin1(SIGR, SIGZ, SIGT)
4390 SIG2 = Prin2(SIGR, SIGZ, SIGT)
4391 SIG3 = SIGH
4392 SP1 = Ftres(SIG1, SIG2, SIG3)
4393
4394 If Analy. Lstrs(I, 0, 0) >= Analy. Lstrs(I, 1, 0) Then
4395 LSIGR = Analy. Lstrs(I, 0, 0) - SF1
4396 LSIGZ = Analy. Lstrs(I, 1, 0)
4397 Else
4398 LSIGR = Analy. Lstrs(I, 0, 0)
4399 LSIGZ = Analy. Lstrs(I, 1, 0) - SF1

```

```

4400      End If
4401      LSIGH = Analy.Lstrs(I, 2, 0) - SF1
4402      LSIGT = Analy.Lstrs(I, 3, 0)
4403      LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
4404      LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
4405      LSIG3 = LSIGH
4406      SN1 = Ftres(LSIG1, LSIG2, LSIG3)
4407      EN1 = SN1 / E1
4408      If SN1 = 0! Then
4409          K1D = 0!
4410      Else
4411          K1D = SP1 / SN1
4412      End If
4413      K1 = Smax(K1D, 1!, 1!)
4414      If SN1 = 0! Then
4415          KED1D = 0!
4416      Else
4417          KED1D = GAMMAY * BAR3SM / SN1
4418      End If
4419      KEDL1 = Smax(1.0+(Allmtd.Q1(I)-1.0)*(1.0-KED1D/K1D), 1.0)
4420      KEDL1 = Smax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED1D / K1), 1!, 1!)
4421      KEDG1 = Smax(1! + (Allmtd.Qn(I) * Allmtd.Qw(I) - 1!) * (1! - KED1D), 1!, 1!)
4422
4423      ピーク応力を含む応力集中係数の計算
4424      If SN1 = 0! Then
4425          KS1 = 0!
4426      Else
4427          KS1 = SPPI / SN1
4428      End If
4429      KSTAR1 = Smax(KS1, 1!, 1!)
4430
4431      KE1 = Smax(KSTAR1 * KEDL1 * KEDG1, 1.3, 1.3)
4432      ETD1 = KE1 * EN1
4433      T I M E 2
4434
4435      熱ピークを含むピーク応力強さの計算
4436      SIGR = Analy.Strs(I, 0, 1)
4437      SIGZ = Analy.Strs(I, 1, 1)
4438      SIGH = Analy.Strs(I, 2, 1)
4439      SIGT = Analy.Strs(I, 3, 1)
4440      SIG1 = Prin1(SIGR, SIGZ, SIGT)
4441      SIG2 = Prin2(SIGR, SIGZ, SIGT)
4442      SIG3 = SIGH
4443      SPP2 = Ftres(SIG1, SIG2, SIG3)
4444
4445      熱ピークを含まないピーク応力強さの計算
4446      EF2 = Abs(ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
4447      If Analy.Strs(I, 0, 1) >= Analy.Strs(I, 1, 1) Then
4448          SIGR = Analy.Strs(I, 0, 1) - SF2
4449          SIGZ = Analy.Strs(I, 1, 1)
4450      Else
4451          SIGR = Analy.Strs(I, 0, 1)
4452          SIGZ = Analy.Strs(I, 1, 1) - SF2
4453      End If
4454      SIGH = Analy.Strs(I, 2, 1) - SF2
4455      SIGT = Analy.Strs(I, 3, 1)
4456      SIG1 = Prin1(SIGR, SIGZ, SIGT)
4457      SIG2 = Prin2(SIGR, SIGZ, SIGT)
4458      SIG3 = SIGH
4459      SP2 = Ftres(SIG1, SIG2, SIG3)
4460
4461      If Analy.Lstrs(I, 0, 1) >= Analy.Lstrs(I, 1, 1) Then
4462          LSIGR = Analy.Lstrs(I, 0, 1) - SF2

```

```

4463     LSIGZ = Analy.Lstrs(I, 1, 1)
4464     Else
4465     LSIGR = Analy.Lstrs(I, 0, 1)
4466     LSIGZ = Analy.Lstrs(I, 1, 1) - SF2
4467     End If
4468     LSIGH = Analy.Lstrs(I, 2, 1) - SF2
4469     LSIGT = Analy.Lstrs(I, 3, 1)
4470     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
4471     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
4472     LSIG3 = LSIGH
4473     SN2 = Ftres(LSIG1, LSIG2, LSIG3)
4474     EN2 = SN2 / E2
4475     If SN2 = 0! Then
4476     K2D = 0!
4477     Else
4478     K2D = SP2 / SN2
4479     End If
4480     K2 = Smax(K2D, 1!, 1!)
4481     If SN2 = 0! Then
4482     KED2D = 0!
4483     Else
4484     KED2D = GAMMAY * BAR3SM / SN2
4485     End If
4486     KEDL2 = Smax(1.0+(Allmtd.Q1(I)-1.0)*(1.0-KED2D/K2D), 1.0, 1.0)
4487     KEDL2 = Smax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED2D / K2), 1!, 1!)
4488     KEDG2 = Smax(1! + (Allmtd.Qn(I) * Allmtd.Qw(I) - 1!) * (1! - KED2D), 1!, 1!)
4489
4490     ピーク応力を含む応力集中係数の計算
4491     If SN2 = 0! Then
4492     KS2 = 0!
4493     Else
4494     KS2 = SPP2 / SN2
4495     End If
4496     KSTAR2 = Smax(KS2, 1!, 1!)
4497
4498     KE2 = Smax(KSTAR2 * KEDL2 * KEDG2, 1.3, 1.3)
4499     ETD2 = KE2 * EN2
4500
4501     ETC = Smax(ETD1, ETD2, ETD2)
4502
4503     Allmtd.Si(I) = Sepr(ETC, Mc)
4504     Allmtd.Si(I) = Sepr(ETC, Mc) / 2!
4505
4506     Res = Debug_disp("[Sepr]/ETD1=" & Str(ETD1) & "/ETD2=" & Str(ETD2)) 'for Debug
4507     If Res = 9 Then Exit Sub
4508
4509     Allmtd.Si(I) = Allmtd.Si(I) + Cond.Mecha
4510
4511     EC1 = 0!
4512     ALPHC = 1!
4513     ALPHR = 1!
4514     DT = Cond.Hldtmh
4515     Allmtd.Qeff(I) = (Allmtd.K(I) ^ (3! / 4!)) * Allmtd.Qnu(I) * Allmtd.Qn(I) * Allmtd.Qw(I)
4516     Call R1x1(Allmtd.Si(I), 0!, 1!, 1!, Allmtd.Qeff(I), Cond.Hldtmh)
4517     Call R1x1(Allmtd.Si(I), EC1, ALPHC, ALPHR, Allmtd.Qeff(I), DT, S2X, DCX)
4518     DCP = 2! * Cond.Cycle * Dcxx(0)
4519
4520     Allmtd.Dc(I) = DCN + DCP
4521
4522     クリープ疲労損傷の計算
4523
4524     Allmtd.D(I) = Allmtd.Df(I) + Allmtd.Dc(I)
4525

```

```

4526   許容値の計算
4527
4528       If Allmtd.Dc(I) >= 0! And Allmtd.Dc(I) <= .3 Then
4529           Allmtd.Dcr(I) = 1! - 4! / 3! * Allmtd.Dc(I)
4530       Elseif Allmtd.Dc(I) <= 1! Then
4531           Allmtd.Dcr(I) = 3! / 7! + 4! / 7! * Allmtd.Dc(I)
4532       Elseif Allmtd.Dc(I) > 1! Then
4533           Allmtd.Dcr(I) = 1!
4534       End If
4535
4536   Next I
4537
4538 End Sub
4539
4540
4541   E Q E F
4542   (中間保持考慮、弾性追従パラメータ最適化、ひずみ集中係数改良)
4543   平成3年07月09日   Ver. 1. 4
4544
4545 Sub Eqefp (Icount)
4546   Dim Icount As Integer
4547   Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
4548   Dim KEDD As Single
4549   Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
4550   Dim K1, K2, KE1, KE2 As Single
4551   Dim K1D, K2D, KED1D, KED2D As Single
4552   Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
4553   Dim LSIG1, LSIG2, LSIG3 As Single
4554   Dim NU, NU1, NU2, NF           As Single
4555
4556   Iflg = 0
4557
4558   For I = 0 To Icount - 1
4559
4560       歪範囲の計算
4561
4562       T I M E 1
4563       Mc.Tc = Analy.Mtemp(I, 0)
4564       Call Mset(Iflg, Mc)
4565       E1 = Mc.E
4566       NU1 = Calcnu(Mc)
4567       ALPHA1 = Alpha(Mc)
4568       T I M E 2
4569       Mc.Tc = Analy.Mtemp(I, 1)
4570       Call Mset(Iflg, Mc)
4571       E2 = Mc.E
4572       NU2 = Calcnu(Mc)
4573       ALPHA2 = Alpha(Mc)
4574
4575       熱ピークを含むピーク応力強さの計算
4576       SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
4577       SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
4578       SIGH = Analy.Strs(I, 2, 0) - Analy.Strs(I, 2, 1)
4579       SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
4580       SIG1 = Prin1(SIGR, SIGZ, SIGT)
4581       SIG2 = Prin2(SIGR, SIGZ, SIGT)
4582       SIG3 = SIGH
4583       SPP = Ftres(SIG1, SIG2, SIG3)
4584
4585       熱ピークを含まないピーク応力強さの計算
4586       SF1 = (-E1 * ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0))) / (1 - NU1)
4587       SF2 = (-E2 * ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2)
4588       Allmtd.Ef(I) = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *

```

```

(Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
4589
4590     SANR = Abs(Analy.Strs(I, 0, 1) - Analy.Strs(I, 0, 0))
4591     SANZ = Abs(Analy.Strs(I, 1, 1) - Analy.Strs(I, 1, 0))
4592     If SANR >= SANZ Then
4593         SIGR = (Analy.Strs(I, 0, 0) - SF1) - (Analy.Strs(I, 0, 1) - SF2)
4594         SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
4595     Else
4596         SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
4597         SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)
4598     End If
4599     SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
4600     SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
4601     SIG1 = Prin1(SIGR, SIGZ, SIGT)
4602     SIG2 = Prin2(SIGR, SIGZ, SIGT)
4603     SIG3 = SIGH
4604     Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
4605     公称応力強さの計算
4606     Allmtd.Sn(I) = Bscval.Ltres(I)
4607     ピーク応力を含む応力集中係数の計算
4608     If Allmtd.Sn(I) = 0! Then
4609         KS = 0!
4610     Else
4611         KS = SPP / Allmtd.Sn(I)
4612     End If
4613     KSTAR = Nmax(KS, 1!, 1!)
4614     ピーク応力を含まない応力集中係数の計算
4615     If Allmtd.Sn(I) = 0! Then
4616         KD = 0!
4617     Else
4618         KD = Allmtd.Spx(I) / Allmtd.Sn(I)
4619     End If
4620     Allmtd.K(I) = Nmax(KD, 1!, 1!)
4621
4622     最高金属温度による  $\epsilon_n$ 、 $S_m h$ 、 $S_r h$  の計算
4623     Mc.Tc = Analy.Tmpmax(I)
4624     Call Mset(Iflg, Mc)
4625     EN = Allmtd.Sn(I) / Mc.E
4626     SMH = Mc.Sm
4627     Call R1x1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmh)
4628     Call R1x1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmh, SRH, DCX)
4629     S2x1 = S2xx(0)
4630
4631     コールド温度による  $S_m c$ 、 $S_r c$  の計算
4632     Mc.Tc = Analy.Temp(I, 1)
4633     Call Mset(Iflg, Mc)
4634     SMC = Mc.Sm
4635     Call R1x1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmc)
4636     Call R1x1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmc, SRC, DCY)
4637
4638     ひずみ集中係数の計算
4639     SM3BAR = S2x1 + S2xx(0)
4640     Allmtd.Qn(I) = Analy.Q(I)
4641     NU = .3
4642     Allmtd.Qnu(I) = 2! * (1! - NU)
4643     QNUSTR = 5! / 3!
4644     Allmtd.Q1(I) = Nmin((KSTAR ^ (3! / 4!)) * Allmtd.Qnu(I), (Allmtd.K(I) ^ (3! / 4!)) * QNUS
TR, (Allmtd.K(I) ^ (3! / 4!)) * QNUSTR)
4645     If Allmtd.Sn(I) = 0! Then
4646         KEDD = 0!
4647     Else
4648         KEDD = SM3BAR / Allmtd.Sn(I)
4649     End If

```

```

4650 Allmtd. Ked1(I) = Nmax(!! + (Allmtd. Q1(I) - !!) * (!! - KEDD / KSTAR), !!, !!)
4651 Allmtd. Kedg(I) = Nmax(!! + (Allmtd. Qn(I) - !!) * (!! - KEDD), !!, !!)
4652
4653 Allmtd. Ke(I) = KSTAR * Allmtd. Ked1(I) * Allmtd. Kedg(I)
4654
4655 Allmtd. Et(I) = Allmtd. Ke(I) * EN
4656
4657 疲労損傷の計算
4658
4659 ファンクションANFの計算機能の限界
4660 If Allmtd. Et(I) < .0005 Then
4661     Allmtd. Et(I) = .0005
4662 End If
4663
4664 最高金属温度
4665 Mc. Tc = Analy. Tmpmax(I)
4666 Call Mset(Iflg, Mc)
4667
4668 EDOT = 2! * Allmtd. Et(I) / ((Cond. Hldtmc + Cond. Hldtmh) * 3600!)
4669 NF = Anf(Allmtd. Et(I), EDOT, Mc) 'Fortran Library(DLL)呼び出し
4670 Allmtd. Df(I) = Cond. Cycle / NF
4671
4672 クリープ損傷の計算
4673
4674 DCNの計算
4675 DCN = 0
4676 DCPの計算
4677 初期応力の計算
4678 TIME 1
4679
4680
4681 熱ピークを含むピーク応力強さの計算
4682 SIGR = Analy. Strs(I, 0, 0)
4683 SIGZ = Analy. Strs(I, 1, 0)
4684 SIGH = Analy. Strs(I, 2, 0)
4685 SIGT = Analy. Strs(I, 3, 0)
4686 SIG1 = Prin1(SIGR, SIGZ, SIGT)
4687 SIG2 = Prin2(SIGR, SIGZ, SIGT)
4688 SIG3 = SIGH
4689 SPP1 = Ftres(SIG1, SIG2, SIG3)
4690
4691 熱ピークを含まないピーク応力強さの計算
4692 EF1 = Abs(ALPHA1 * (Analy. Temp(I, 0) - Analy. Ltemp(I, 0)) / (1 - NU1))
4693 If Analy. Strs(I, 0, 0) >= Analy. Strs(I, 1, 0) Then
4694     SIGR = Analy. Strs(I, 0, 0) - SF1
4695     SIGZ = Analy. Strs(I, 1, 0)
4696 Else
4697     SIGR = Analy. Strs(I, 0, 0)
4698     SIGZ = Analy. Strs(I, 1, 0) - SF1
4699 End If
4700 SIGH = Analy. Strs(I, 2, 0) - SF1
4701 SIGT = Analy. Strs(I, 3, 0)
4702 SIG1 = Prin1(SIGR, SIGZ, SIGT)
4703 SIG2 = Prin2(SIGR, SIGZ, SIGT)
4704 SIG3 = SIGH
4705 SPI = Ftres(SIG1, SIG2, SIG3)
4706 公称応力強さの計算
4707 LSIGR = Analy. Lstrs(I, 0, 0)
4708 LSIGZ = Analy. Lstrs(I, 1, 0)
4709 LSIGH = Analy. Lstrs(I, 2, 0)
4710 LSIGT = Analy. Lstrs(I, 3, 0)
4711 LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
4712 LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)

```



```

4713     LSIG3 = LSIGH
4714     SN1 = Ftres(LSIG1, LSIG2, LSIG3)
4715     EN1 = SN1 / E1
4716
4717     ピーク応力を含む応力集中係数の計算
4718     If SN1 = 0! Then
4719         KS1 = 0!
4720     Else
4721         KS1 = SPP1 / SN1
4722     End If
4723     KSTAR1 = Nmax(KS1, 1!, 1!)
4724     ピーク応力を含まない応力集中係数の計算
4725     If SN1 = 0! Then
4726         K1D = 0!
4727     Else
4728         K1D = SP1 / SN1
4729     End If
4730     K1 = Nmax(K1D, 1!, 1!)
4731     ひずみ集中係数の計算
4732     SM3 = 1.5 * SMH + 1.5 * SMC
4733     If SN1 = 0! Then
4734         KED1D = 0!
4735     Else
4736         KED1D = SM3 / SN1
4737     End If
4738     KEDL1 = Nmax(1! + (Allmtd. Q1(I) - 1!) * (1! - KED1D / KSTAR1), 1!, 1!)
4739     KEDG1 = Nmax(1! + (Allmtd. Qn(I) - 1!) * (1! - KED1D), 1!, 1!)
4740
4741     KE1 = KSTAR1 * KEDL1 * KEDG1
4742     ETD1 = KE1 * EN1
4743     T I M E 2
4744
4745     熱ピークを含むピーク応力強さの計算
4746     SIGR = Analy. Strs(I, 0, 1)
4747     SIGZ = Analy. Strs(I, 1, 1)
4748     SIGH = Analy. Strs(I, 2, 1)
4749     SIGT = Analy. Strs(I, 3, 1)
4750     SIG1 = Prin1(SIGR, SIGZ, SIGT)
4751     SIG2 = Prin2(SIGR, SIGZ, SIGT)
4752     SIG3 = SIGH
4753     SPP2 = Ftres(SIG1, SIG2, SIG3)
4754
4755     熱ピークを含まないピーク応力強さの計算
4756     EF2 = Abs(ALPHA2 * (Analy. Temp(I, 1) - Analy. Ltemp(I, 1)) / (1 - NU2))
4757     If Analy. Strs(I, 0, 1) >= Analy. Strs(I, 1, 1) Then
4758         SIGR = Analy. Strs(I, 0, 1) - SF2
4759         SIGZ = Analy. Strs(I, 1, 1)
4760     Else
4761         SIGR = Analy. Strs(I, 0, 1)
4762         SIGZ = Analy. Strs(I, 1, 1) - SF2
4763     End If
4764     SIGH = Analy. Strs(I, 2, 1) - SF2
4765     SIGT = Analy. Strs(I, 3, 1)
4766     SIG1 = Prin1(SIGR, SIGZ, SIGT)
4767     SIG2 = Prin2(SIGR, SIGZ, SIGT)
4768     SIG3 = SIGH
4769     SP2 = Ftres(SIG1, SIG2, SIG3)
4770     公称応力強さの計算
4771     LSIGR = Analy. Lstrs(I, 0, 1)
4772     LSIGZ = Analy. Lstrs(I, 1, 1)
4773     LSIGH = Analy. Lstrs(I, 2, 1)
4774     LSIGT = Analy. Lstrs(I, 3, 1)
4775     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)

```

```

4776     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
4777     LSIG3 = LSIGH
4778     SN2 = Ftres(LSIG1, LSIG2, LSIG3)
4779     EN2 = SN2 / E2
4780
4781     ピーク応力を含む応力集中係数の計算
4782     If SN2 = 0! Then
4783         KS2 = 0!
4784     Else
4785         KS2 = SPP2 / SN2
4786     End If
4787     KSTAR2 = Nmax(KS2, 1!, 1!)
4788     ピーク応力を含まない応力集中係数の計算
4789     If SN2 = 0! Then
4790         K2D = 0!
4791     Else
4792         K2D = SP2 / SN2
4793     End If
4794     K2 = Nmax(K2D, 1!, 1!)
4795     ひずみ集中係数の計算
4796     If SN2 = 0! Then
4797         KED2D = 0!
4798     Else
4799         KED2D = SM3 / SN2
4800     End If
4801     KEDL2 = Nmax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED2D / KSTAR2), 1!, 1!)
4802     KEDG2 = Nmax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED2D), 1!, 1!)
4803
4804     KE2 = KSTAR2 * KEDL2 * KEDG2
4805     ETD2 = KE2 * EN2
4806
4807     ETC = Nmax(ETD1, ETD2, ETD2)
4808
4809     Allmtd.Si(I) = Sepr(ETC, Mc)           'Fortran Library(DLL) 呼び出し
4810     Allmtd.Si(I) = Sepr(ETC, Mc) / 2!    'Fortran Library(DLL) 呼び出し
4811
4812     Allmtd.Si(I) = Allmtd.Si(I) + Cond.Mecha
4813
4814     EC1 = 0!
4815     ALPHC = 1!
4816     ALPHR = 1!
4817     DT = Cond.Hldtmh
4818     Allmtd.Qeff(I) = (Allmtd.K(I) ^ (3! / 4!)) * Allmtd.Qnu(I) * Allmtd.Qn(I)
4819     Call R1x1(Allmtd.Si(I), 0!, 1!, 1!, Allmtd.Qeff(I), Cond.Hldtmh)
4820     Call R1x1(Allmtd.Si(I), EC1, ALPHC, ALPHR, Allmtd.Qeff(I), DT, S2X, DCX)
4821     DCP = CSng(Cond.Cycle) * Dcxx(0)
4822
4823     Allmtd.Dc(I) = DCN + DCP
4824
4825     クリープ疲労損傷の計算
4826
4827     Allmtd.D(I) = Allmtd.Df(I) + Allmtd.Dc(I)
4828
4829     許容値の計算
4830
4831     If Allmtd.Dc(I) >= 0! And Allmtd.Dc(I) <= .3 Then
4832         Allmtd.Dcr(I) = 1! - 4! / 3! * Allmtd.Dc(I)
4833     ElseIf Allmtd.Dc(I) <= 1! Then
4834         Allmtd.Dcr(I) = 3! / 7! + 4! / 7! * Allmtd.Dc(I)
4835     ElseIf Allmtd.Dc(I) > 1! Then
4836         Allmtd.Dcr(I) = 1!
4837     End If
4838

```

```

4839 Next I
4840
4841 End Sub
4842
4843
4844 EQEF - WELD
4845
4846
4847 Sub Eqefwdp (Icount)
4848
4849 Dim Icount As Integer
4850 Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
4851 Dim KEDD As Single
4852 Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
4853 Dim K1, K2, KE1, KE2 As Single
4854 Dim K1D, K2D, KED1D, KED2D As Single
4855 Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
4856 Dim LSIG1, LSIG2, LSIG3 As Single
4857 Dim NU, NU1, NU2, NF As Single
4858
4859 Iflg = 0
4860
4861 For I = 0 To Icount - 1
4862
4863     歪範囲の計算
4864
4865     TIME 1
4866     Mc.Tc = Analy.Mtemp(I, 0)
4867     Call Mset(Iflg, Mc)
4868     E1 = Mc.E
4869     NU1 = Calcnu(Mc)
4870     ALPHA1 = Alpha(Mc)
4871     TIME 2
4872     Mc.Tc = Analy.Mtemp(I, 1)
4873     Call Mset(Iflg, Mc)
4874     E2 = Mc.E
4875     NU2 = Calcnu(Mc)
4876     ALPHA2 = Alpha(Mc)
4877
4878     熱ピークを含むピーク応力強さの計算
4879     SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
4880     SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
4881     SIGH = Analy.Strs(I, 2, 0) - Analy.Strs(I, 2, 1)
4882     SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
4883     SIG1 = Prin1(SIGR, SIGZ, SIGT)
4884     SIG2 = Prin2(SIGR, SIGZ, SIGT)
4885     SIG3 = SIGH
4886     SPP = Ftres(SIG1, SIG2, SIG3)
4887
4888     熱ピークを含まないピーク応力強さの計算
4889     SF1 = (-E1 * ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0))) / (1 - NU1)
4890     SF2 = (-E2 * ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2)
4891     Allmtd.Ef(I) = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *
4892     (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
4893     SANR = Abs(Analy.Strs(I, 0, 1) - Analy.Strs(I, 0, 0))
4894     SANZ = Abs(Analy.Strs(I, 1, 1) - Analy.Strs(I, 1, 0))
4895     If SANR >= SANZ Then
4896         SIGR = (Analy.Strs(I, 0, 0) - SF1) - (Analy.Strs(I, 0, 1) - SF2)
4897         SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
4898     Else
4899         SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
4900         SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)

```

```

4901      End If
4902      SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
4903      SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
4904      SIG1 = Prin1(SIGR, SIGZ, SIGT)
4905      SIG2 = Prin2(SIGR, SIGZ, SIGT)
4906      SIG3 = SIGH
4907      Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
4908
4909      Allmtd.Sn(I) = Bscval.Ltres(I)
4910
4911      最高金属温度による  $\epsilon_n$ 、S r h の計算
4912      Mc.Tc = Analy.Tmpmax(I)
4913      Call Mset(Iflg, Mc)
4914      EN = Allmtd.Sn(I) / Mc.E
4915
4916      If Allmtd.Sn(I) = 0! Then
4917          KD = 0!
4918      Else
4919          KD = Allmtd.Spx(I) / Allmtd.Sn(I)
4920      End If
4921      Allmtd.K(I) = Smax(KD, 1!, 1!)
4922      Call R1x1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmh)
4923      Call R1x1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmh, SRH, DCX)
4924
4925      コールド温度による S r c の計算
4926      Mc.Tc = Analy.Temp(I, 1)
4927      Call Mset(Iflg, Mc)
4928      NU = .3
4929      Allmtd.Qnu(I) = 2! * (1! - NU)
4930      BAR3SM = S2xx(0) + 1.5 * Mc.Sm
4931      Allmtd.Qn(I) = Analy.Q(I)
4932      Allmtd.Qw(I) = 1.5
4933      Allmtd.Q1(I) = Allmtd.K(I) ^ (3! / 4!)
4934      GAMMAY = .8
4935      If Allmtd.Sn(I) = 0! Then
4936          KEDD = 0!
4937      Else
4938          KEDD = GAMMAY * BAR3SM / Allmtd.Sn(I)
4939      End If
4940      Allmtd.Ked1(I) = Smax(1! + (Allmtd.Qnu(I) * Allmtd.Q1(I) - 1!) * (1! - KEDD / Allmtd.K(I)),
4941      1!, 1!)
4942      Allmtd.Kedg(I) = Smax(1! + (Allmtd.Qn(I) * Allmtd.Qw(I) - 1!) * (1! - KEDD), 1!, 1!)
4943
4944      ピーク応力を含む応力集中係数の計算
4945      If Allmtd.Sn(I) = 0! Then
4946          KS = 0!
4947      Else
4948          KS = SPP / Allmtd.Sn(I)
4949      End If
4950      KSTAR = Smax(KS, 1!, 1!)
4951
4952      Allmtd.Ke(I) = Smax(KSTAR * Allmtd.Ked1(I) * Allmtd.Kedg(I), 1.3, 1.3)
4953
4954      Allmtd.Et(I) = Allmtd.Ke(I) * EN
4955
4956      疲労損傷の計算
4957
4958      ファンクション ANF の計算機能の限界
4959      If Allmtd.Et(I) < .0005 Then Allmtd.Et(I) = .0005
4960
4961      最高金属温度
4962      Mc.Tc = Analy.Tmpmax(I)
4963      Call Mset(Iflg, Mc)

```

```

4963
4964     EDOT = 2! * Allmtd.Et(I) / ((Cond.Hldtmc + Cond.Hldtmh) * 3600!)
4965     NF = Anf(Allmtd.Et(I), EDOT, Mc)      'Fortran Library(DLL) 呼び出し
4966
4967     Allmtd.Df(I) = Cond.Cycle / NF
4968
4969     クリープ損傷の計算
4970
4971     DCNの計算
4972     DCN = 0
4973     DCPの計算
4974     初期応力の計算
4975     TIME 1
4976
4977     熱ピークを含まないピーク応力強さの計算
4978     SIGR = Analy.Strs(I, 0, 0)
4979     SIGZ = Analy.Strs(I, 1, 0)
4980     SIGH = Analy.Strs(I, 2, 0)
4981     SIGT = Analy.Strs(I, 3, 0)
4982     SIG1 = Prin1(SIGR, SIGZ, SIGT)
4983     SIG2 = Prin2(SIGR, SIGZ, SIGT)
4984     SIG3 = SIGH
4985     SPP1 = Ftres(SIG1, SIG2, SIG3)
4986
4987     熱ピークを含まないピーク応力強さの計算
4988     EF1 = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1))
4989     If Analy.Strs(I, 0, 0) >= Analy.Strs(I, 1, 0) Then
4990         SIGR = Analy.Strs(I, 0, 0) - SF1
4991         SIGZ = Analy.Strs(I, 1, 0)
4992     Else
4993         SIGR = Analy.Strs(I, 0, 0)
4994         SIGZ = Analy.Strs(I, 1, 0) - SF1
4995     End If
4996     SIGH = Analy.Strs(I, 2, 0) - SF1
4997     SIGT = Analy.Strs(I, 3, 0)
4998     SIG1 = Prin1(SIGR, SIGZ, SIGT)
4999     SIG2 = Prin2(SIGR, SIGZ, SIGT)
5000     SIG3 = SIGH
5001     SP1 = Ftres(SIG1, SIG2, SIG3)
5002
5003     If Analy.Lstrs(I, 0, 0) >= Analy.Lstrs(I, 1, 0) Then
5004         LSIGR = Analy.Lstrs(I, 0, 0) - SF1
5005         LSIGZ = Analy.Lstrs(I, 1, 0)
5006     Else
5007         LSIGR = Analy.Lstrs(I, 0, 0)
5008         LSIGZ = Analy.Lstrs(I, 1, 0) - SF1
5009     End If
5010     LSIGH = Analy.Lstrs(I, 2, 0) - SF1
5011     LSIGT = Analy.Lstrs(I, 3, 0)
5012     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
5013     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
5014     LSIG3 = LSIGH
5015     SNI = Ftres(LSIG1, LSIG2, LSIG3)
5016     ENI = SNI / EI
5017     If SNI = 0! Then
5018         K1D = 0!
5019     Else
5020         K1D = SP1 / SNI
5021     End If
5022     K1 = Smax(K1D, 1!, 1!)
5023     If SNI = 0! Then
5024         KED1D = 0!
5025     Else

```

```

5026      KED1D = GAMWAY * BAR3SM / SN1
5027      End If
5028      KEDL1 = Smax(1.0+(AlllmtD.Qnu(I)*AlllmtD.Q1(I)-1.0)*(1.0-KED1D/K1D), 1.0)
5029      KEDL1 = Smax(1! + (AlllmtD.Qnu(I) * AlllmtD.Q1(I) - 1!) * (1! - KED1D / K1), 1!, 1!)
5030      KEDG1 = Smax(1! + (AlllmtD.Qn(I) * AlllmtD.Qw(I) - 1!) * (1! - KED1D), 1!, 1!)
5031
5032      ピーク応力を含む応力集中係数の計算
5033      If SN1 = 0! Then
5034          KS1 = 0!
5035      Else
5036          KS1 = SPP1 / SN1
5037      End If
5038      KSTAR1 = Smax(KS1, 1!, 1!)
5039
5040      KE1 = Smax(KSTAR1 * KEDL1 * KEDG1, 1.3, 1.3)
5041      ETD1 = KE1 * EN1
5042      T I M E 2
5043
5044      熱ピークを含むピーク応力強さの計算
5045      SIGR = Analy.Strs(I, 0, 1)
5046      SIGZ = Analy.Strs(I, 1, 1)
5047      SIGH = Analy.Strs(I, 2, 1)
5048      SIGT = Analy.Strs(I, 3, 1)
5049      SIG1 = Prin1(SIGR, SIGZ, SIGT)
5050      SIG2 = Prin2(SIGR, SIGZ, SIGT)
5051      SIG3 = SIGH
5052      SPP2 = Ftres(SIG1, SIG2, SIG3)
5053
5054      熱ピークを含まないピーク応力強さの計算
5055      EF2 = Abs(ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
5056      If Analy.Strs(I, 0, 1) >= Analy.Strs(I, 1, 1) Then
5057          SIGR = Analy.Strs(I, 0, 1) - SF2
5058          SIGZ = Analy.Strs(I, 1, 1)
5059      Else
5060          SIGR = Analy.Strs(I, 0, 1)
5061          SIGZ = Analy.Strs(I, 1, 1) - SF2
5062      End If
5063      SIGH = Analy.Strs(I, 2, 1) - SF2
5064      SIGT = Analy.Strs(I, 3, 1)
5065      SIG1 = Prin1(SIGR, SIGZ, SIGT)
5066      SIG2 = Prin2(SIGR, SIGZ, SIGT)
5067      SIG3 = SIGH
5068      SP2 = Ftres(SIG1, SIG2, SIG3)
5069
5070      If Analy.Lstrs(I, 0, 1) >= Analy.Lstrs(I, 1, 1) Then
5071          LSIGR = Analy.Lstrs(I, 0, 1) - SF2
5072          LSIGZ = Analy.Lstrs(I, 1, 1)
5073      Else
5074          LSIGR = Analy.Lstrs(I, 0, 1)
5075          LSIGZ = Analy.Lstrs(I, 1, 1) - SF2
5076      End If
5077      LSIGH = Analy.Lstrs(I, 2, 1) - SF2
5078      LSIGT = Analy.Lstrs(I, 3, 1)
5079      LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
5080      LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
5081      LSIG3 = LSIGH
5082      SN2 = Ftres(LSIG1, LSIG2, LSIG3)
5083      EN2 = SN2 / E2
5084      If SN2 = 0! Then
5085          K2D = 0!
5086      Else
5087          K2D = SP2 / SN2
5088      End If

```

```

5089      K2 = Smax(K2D, 1!, 1!)
5090      If SN2 = 0! Then
5091          KED2D = 0!
5092      Else
5093          KED2D = GANWAY * BAR3SM / SN2
5094      End If
5095      KEDL2 = Smax(1.0+(Allmtd.Qnu(I)*Allmtd.Q1(I)-1.0)*(1.0-KED2D/K2D), 1.0, 1.0)
5096      KEDL2 = Smax(1! + (Allmtd.Qnu(I) * Allmtd.Q1(I) - 1!) * (1! - KED2D / K2), 1!, 1!)
5097      KEDG2 = Smax(1! + (Allmtd.Qn(I) * Allmtd.Qw(I) - 1!) * (1! - KED2D), 1!, 1!)
5098
5099      ピーク応力を含む応力集中係数の計算
5100      If SN2 = 0! Then
5101          KS2 = 0!
5102      Else
5103          KS2 = SPP2 / SN2
5104      End If
5105      KSTAR2 = Smax(KS2, 1!, 1!)
5106
5107      KE2 = Smax(KSTAR2 * KEDL2 * KEDG2, 1.3, 1.3)
5108      ETD2 = KE2 * EN2
5109
5110      ETC = Smax(ETD1, ETD2, ETD2)
5111
5112      Allmtd.Si(I) = Sepr(ETC, Mc)
5113      Allmtd.Si(I) = Sepr(ETC, Mc) / 2!
5114
5115      Res = Debug_disp("[Sepr]/ETD1=" & Str(ETD1) & "/ETD2=" & Str(ETD2)) ' for Debug
5116      If Res = 9 Then Exit Sub
5117
5118      Allmtd.Si(I) = Allmtd.Si(I) + Cond.Mecha
5119
5120      EC1 = 0!
5121      ALPHC = 1!
5122      ALPHR = 1!
5123      DT = Cond.Hldtmh
5124      Allmtd.Qeff(I) = (Allmtd.K(I) ^ (3! / 4!)) * Allmtd.Qnu(I) * Allmtd.Qn(I) * Allmtd.Qw(I)
5125      Call R1x1(Allmtd.Si(I), 0!, 1!, 1!, Allmtd.Qeff(I), Cond.Hldtmh)
5126      Call R1x1(Allmtd.Si(I), EC1, ALPHC, ALPHR, Allmtd.Qeff(I), DT, S2X, DCX)
5127      DCP = 2! * Cond.Cycle * Dcxx(0)
5128
5129      Allmtd.Dc(I) = DCN + DCP
5130
5131      クリープ疲労損傷の計算
5132
5133      Allmtd.D(I) = Allmtd.Df(I) + Allmtd.Dc(I)
5134
5135      許容値の計算
5136
5137      If Allmtd.Dc(I) >= 0! And Allmtd.Dc(I) <= .3 Then
5138          Allmtd.Dcr(I) = 1! - 4! / 3! * Allmtd.Dc(I)
5139      ElseIf Allmtd.Dc(I) <= 1! Then
5140          Allmtd.Dcr(I) = 3! / 7! + 4! / 7! * Allmtd.Dc(I)
5141      ElseIf Allmtd.Dc(I) > 1! Then
5142          Allmtd.Dcr(I) = 1!
5143      End If
5144
5145      Next I
5146
5147 End Sub
5148
5149
5150 EQEF
5151

```

```

5152 '      損傷計算結果データ->Grid.Text7°のハテ代入
5153 '
5154 Sub Eqfout (Icount)
5155 '   Dim Icount As Integer
5156 '   Dim I       As Integer
5157 '
5158 '   For I = 0 To Icount - 1
5159 '       Form1!Grid1.Row = I + 1
5160 '       Form1!Grid1.Col = 1
5161 '       Form1!Grid1.Text = Analy.R(I, 0)
5162 '       Form1!Grid1.Col = 2
5163 '       Form1!Grid1.Text = Analy.Z(I, 0)
5164 '       Form1!Grid1.Col = 3
5165 '       Form1!Grid1.Text = Analy.H(I, 0)
5166 '       Form1!Grid1.Col = 4
5167 '       Form1!Grid1.Text = Allmtd.Spx(I)
5168 '       Form1!Grid1.Col = 5
5169 '       Form1!Grid1.Text = Allmtd.Sn(I)
5170 '       Form1!Grid1.Col = 6
5171 '       Form1!Grid1.Text = Allmtd.K(I)
5172 '       Form1!Grid1.Col = 7
5173 '       Form1!Grid1.Text = Allmtd.Ked(I)
5174 '       Form1!Grid1.Col = 8
5175 '       Form1!Grid1.Text = Allmtd.Ke(I)
5176 '       Form1!Grid1.Col = 9
5177 '       Form1!Grid1.Text = Allmtd.Kedg(I)
5178 '       Form1!Grid1.Col = 10
5179 '       Form1!Grid1.Text = Allmtd.Kedl(I)
5180 '       Form1!Grid1.Col = 11
5181 '       Form1!Grid1.Text = Allmtd.Ef(I)
5182 '       Form1!Grid1.Col = 12
5183 '       Form1!Grid1.Text = Allmtd.Et(I)
5184 '       Form1!Grid1.Col = 13
5185 '       Form1!Grid1.Text = Allmtd.Si(I)
5186 '       Form1!Grid1.Col = 14
5187 '       Form1!Grid1.Text = Allmtd.Qn(I)
5188 '       Form1!Grid1.Col = 15
5189 '       Form1!Grid1.Text = Allmtd.Ql(I)
5190 '       Form1!Grid1.Col = 16
5191 '       Form1!Grid1.Text = Allmtd.Qnu(I)
5192 '       Form1!Grid1.Col = 17
5193 '       Form1!Grid1.Text = Allmtd.Qeff(I)
5194 '       Form1!Grid1.Col = 18
5195 '       Form1!Grid1.Text = Allmtd.Df(I)
5196 '       Form1!Grid1.Col = 19
5197 '       Form1!Grid1.Text = Allmtd.Dc(I)
5198 '       Form1!Grid1.Col = 20
5199 '       Form1!Grid1.Text = Allmtd.D(I)
5200 '       Form1!Grid1.Col = 21
5201 '       Form1!Grid1.Text = Allmtd.Dcr(I)
5202 '   Next I
5203 '
5204 End Sub
5205 '
5206 '
5207 '   EQEF - WELD
5208 '
5209 '      損傷計算結果データ->Grid.Text7°のハテ代入
5210 '
5211 Sub Ewdout (Icount)
5212 '   Dim Icount As Integer
5213 '   Dim I       As Integer
5214 '

```



```
5215 For I = 0 To Icount - 1
5216 Form1!Grid1.Row = I + 1
5217 Form1!Grid1.Col = 1
5218 Form1!Grid1.Text = Analy.R(I, 0)
5219 Form1!Grid1.Col = 2
5220 Form1!Grid1.Text = Analy.Z(I, 0)
5221 Form1!Grid1.Col = 3
5222 Form1!Grid1.Text = Analy.H(I, 0)
5223 Form1!Grid1.Col = 4
5224 Form1!Grid1.Text = Allmtd.Spx(I)
5225 Form1!Grid1.Col = 5
5226 Form1!Grid1.Text = Allmtd.Sn(I)
5227 Form1!Grid1.Col = 6
5228 Form1!Grid1.Text = Allmtd.K(I)
5229 Form1!Grid1.Col = 7
5230 Form1!Grid1.Text = Allmtd.Ke(I)
5231 Form1!Grid1.Col = 8
5232 Form1!Grid1.Text = Allmtd.Kedg(I)
5233 Form1!Grid1.Col = 9
5234 Form1!Grid1.Text = Allmtd.Kedl(I)
5235 Form1!Grid1.Col = 10
5236 Form1!Grid1.Text = Allmtd.Ef(I)
5237 Form1!Grid1.Col = 11
5238 Form1!Grid1.Text = Allmtd.Et(I)
5239 Form1!Grid1.Col = 12
5240 Form1!Grid1.Text = Allmtd.Si(I)
5241 Form1!Grid1.Col = 13
5242 Form1!Grid1.Text = Allmtd.Qn(I)
5243 Form1!Grid1.Col = 14
5244 Form1!Grid1.Text = Allmtd.Qw(I)
5245 Form1!Grid1.Col = 15
5246 Form1!Grid1.Text = Allmtd.Ql(I)
5247 Form1!Grid1.Col = 16
5248 Form1!Grid1.Text = Allmtd.Qnu(I)
5249 Form1!Grid1.Col = 17
5250 Form1!Grid1.Text = Allmtd.Qeff(I)
5251 Form1!Grid1.Col = 18
5252 Form1!Grid1.Text = Allmtd.Df(I)
5253 Form1!Grid1.Col = 19
5254 Form1!Grid1.Text = Allmtd.Dc(I)
5255 Form1!Grid1.Col = 20
5256 Form1!Grid1.Text = Allmtd.D(I)
5257 Form1!Grid1.Col = 21
5258 Form1!Grid1.Text = Allmtd.Dcr(I)
5259 Next I
5260
5261 End Sub
5262
5263 '
5264 ' STAR-NETシステム用プロセス制御関数
5265 ' 損傷計算プログラムが終了と同時に親プログラム(EXCELマクロ)を再起動させる。
5266 '
5267 Sub Excel_Restart ()
5268
5269 AppActivate "PNC STAR"
5270 Form1!Text1.LinkMode = NONE
5271 Form1!Text1.LinkTopic = "Excel|作業.XJS"
5272 Form1!Text1.LinkItem = "R1C1"
5273 Form1!Text1.LinkMode = LINK_MANUAL
5274 Form1!Text1.LinkTimeout = -1
5275 Form1!Text1.LinkExecute "[RUN("STAR制御.XLMマクロ再開")]"
5276 Form1!Text1.LinkMode = NONE
5277 End
```

```

5278
5279 End Sub
5280
5281
5282
5283     NEUBER則算出プログラム (弾完全塑性体を仮定)
5284
5285 Function Fnuber (Sn, FK, BAR3SM)
5286
5287     Dim K As Single
5288
5289     If Sn <= BAR3SM Then
5290         SSTAR = Sn
5291     Else
5292         SSTAR = BAR3SM
5293     End If
5294     If FK * Sn <= BAR3SM Then
5295         SBAR = FK * Sn
5296     Else
5297         SBAR = BAR3SM
5298     End If
5299     If FK = 0 Or SBAR = 0 Then
5300         Fnuber = 1
5301     Else
5302         Fnuber = SSTAR / SBAR * FK * FK
5303     End If
5304
5305 End Function
5306
5307
5308
5309     NEUBER則算出プログラム (単調応力-ひずみ関係を仮定)
5310
5311 Function Fnubr2 (Sn, K, SRH, EH)
5312
5313     Dim K As Single
5314
5315     If Sn <= SRH Then
5316         SSTAR = Sn
5317     Else
5318         SSTAR = Sepm(Sn / EH - SRH / EH, Mc) + SRH
5319     End If
5320     If K * Sn <= SRH Then
5321         SBAR = K * Sn
5322     Else
5323         SBAR = Sepm(K * Sn / EH - SRH / EH, Mc) + SRH
5324     End If
5325     Fnubr2 = SSTAR / SBAR * K * K
5326
5327 End Function
5328
5329
5330     処理状況表示ルーチン
5331
5332 Sub Form_load (N)
5333     Dim N As Integer
5334     Dim I As Integer
5335     For I = 0 To 1
5336         Form1.Text2.Text = Method_name(N, 0)
5337     Next I
5338 End Sub
5339
5340

```

```

5341  TRESCA計算プログラム
5342
5343  Function Ftres (SIG1, SIG2, SIG3)
5344  Dim Sig1, Sig2, Sig3      As Single
5345
5346  Ftres = Smax(Abs(SIG1 - SIG2), Abs(SIG2 - SIG3), Abs(SIG3 - SIG1))
5347
5348  End Function
5349
5350
5351  VON MISES計算プログラム
5352
5353  Function Fvms (R, z, H, T)
5354  Dim R, Z, H, T      As Single
5355
5356  Fvms = Sqr(1! / 2! * ((R - z) ^ 2 + (z - H) ^ 2 + (H - R) ^ 2 + 6! * T ^ 2))
5357
5358  End Function
5359
5360
5361  INELASTIC   TTS-DS を改良   1993.2.3 N.K
5362
5363  Sub Inelastic (Icount)
5364
5365  Dim Icount As Integer
5366  Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
5367  Dim KEDD As Single
5368  Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
5369  Dim K1, K2, KE1, KE2 As Single
5370  Dim K1D, K2D, KED1D, KED2D, KT As Single
5371  Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
5372  Dim LSIG1, LSIG2, LSIG3 As Single
5373  Dim NU, NU1, NU2, NF      As Single
5374  Dim INF                    As Long
5375
5376  Iflg = 0
5377
5378  For I = 0 To Icount - 1
5379  AX(IN*3+I) = Bscval.Vms(I)
5380  Next I
5381
5382  For I = 0 To Icount - 1
5383  AX(IN*4+I) = Bscval.Eepceq(I)
5384  Next I
5385
5386  For I = 0 To Icount - 1
5387  -----
5388  歪範囲の計算
5389  -----
5390  Mc.Tc = Analy.Temp(I, 1)                                1993.2.2
5391  Call Mset(Iflg, Mc)                                       1993.2.2
5392  NUC = Calcnu(Mc)                                           1993.2.2
5393  Mc.Tc = Analy.Temp(I, 2)                                1993.2.2
5394  Call Mset(Iflg, Mc)                                       1993.2.2
5395  NUH = Calcnu(Mc)                                           1993.2.2
5396
5397  NU = (NUC+NUH)/2.0                                         1993.2.2
5398
5399  Mc.Tc = Analy.Tmpmax(I)
5400  Call Mset(Iflg, Mc)
5401  ER = 1.5/(1.0+NU)*Bscval.Eepceq(I)                        1993.2.2
5402  Er = Bscval.Vms(I) / Mc.E + Bscval.Epeq(I) + Bscval.Eceq(I)
5403

```

```

5404      Allmtd. Kep(I) = 1!
5405      Allmtd. Et(I) = Allmtd. Kep(I) * Er
5406  -----
5407      疲労損傷の計算
5408  -----
5409  -----ファンクションANFの計算機能の限界
5410      If Allmtd. Et(I) < .0005 Then Allmtd. Et(I) = .0005
5411      EDOT = 2! * Allmtd. Et(I) / ((Cond. Hldtmc + Cond. Hldtmh) * 3600!)
5412      NF = Anf(Allmtd. Et(I), EDOT, Mc)
5413      Allmtd. Df(I) = Cond. Cycle / NF
5414  -----
5415      クリープ損傷の計算
5416  -----
5417      S0 = Sepr(Allmtd. Et(I), Mc)
5418      S0 = Sepr(Allmtd. Et(I), Mc) / 2!
5419      S0 = S0 + Cond. Mecha
5420      EC1 = 0!
5421      ALPHC = 1!
5422      ALPHR = 1!
5423      Qx = 1!
5424      DT = Cond. Hldtmh
5425      Call R1x1(CSng(S0), 0!, 1!, 1!, 1!, Cond. Hldtmh)
5426      Call R1x1(S0, 0. 0, 1. 0, 1. 0, 1. 0, Cond. Hldtmh, S2xx(0), Dcxx(0), Mc)
5427
5428      Allmtd. Dc(I) = Cond. Cycle * Dcxx(0)
5429  -----
5430      クリープ疲労損傷の計算
5431  -----
5432      Allmtd. D(I) = Allmtd. Df(I) + Allmtd. Dc(I)
5433
5434      Next I
5435
5436 End Sub
5437
5438
5439      評価法名称設定
5440
5441 Sub Initial ()
5442      Method_name(0, 0) = "TTS-DS  "
5443      Method_name(1, 0) = "BDS    "
5444      Method_name(2, 0) = "BDS-MD "
5445      Method_name(3, 0) = "DDS    "
5446      Method_name(4, 0) = "DDS-WELD "
5447      Method_name(5, 0) = "TYPE1  "
5448      Method_name(6, 0) = "TYPE2  "
5449      Method_name(7, 0) = "EQEF   "
5450      Method_name(8, 0) = "EQEF-WELD"
5451      Method_name(9, 0) = "PNC    "
5452      Method_name(10, 0) = "EQEF15 "
5453      Method_name(11, 0) = "EQEF167 "
5454      Method_name(12, 0) = "INELAST "
5455      Method_name(13, 0) = "SIEQ167 "
5456      Method_name(14, 0) = "EQEF167WD"
5457      Method_name(15, 0) = "EQEF167WK"
5458      Method_name(16, 0) = "      "
5459      Method_name(17, 0) = "      "
5460      Method_name(18, 0) = "      "
5461      Method_name(19, 0) = "      "
5462 End Sub
5463
5464
5465      (1) 試験条件シートデータ取得(DDE通信:試験条件.XJS)
5466

```

```

5467 Sub Input_Data_1 ()
5468   Form1!Text1.LinkMode = NONE
5469   Form1!Text1.LinkTopic = "Excel|[ 試験条件.XJS]試験条件"
5470   Form1!Text1.LinkTopic = "Excel|試験条件.XJS"
5471   Form1!Text1.LinkTopic = "Excel|試験条件.CND"
5472   Form1!Text1.LinkItem = "R5C1:R5C20"
5473   Form1!Text1.LinkMode = LINK_AUTOMATIC
5474   Form1!Grid1.SelStartRow = 1
5475   Form1!Grid1.SelStartCol = 1
5476   Form1!Grid1.SelEndRow = 1
5477   Form1!Grid1.SelEndCol = 20
5478   Form1!Grid1.Clip = Form1.Text1.Text
5479
5480   Form1!Grid1.Row = 1
5481
5482   Form1!Grid1.Col = 4
5483   Cond. Material = Form1!Grid1.Text
5484
5485   Call Mtrl                               ' 変換:材料名称->材料番号
5486
5487   Form1!Grid1.Col = 6
5488   Cond. Tempc = Val(Form1!Grid1.Text)
5489
5490   Form1!Grid1.Col = 7
5491   Cond. Temph = Val(Form1!Grid1.Text)
5492
5493   Form1!Grid1.Col = 8
5494   Cond. Hldtmc = Val(Form1!Grid1.Text)
5495
5496   Form1!Grid1.Col = 9
5497   Cond. Hldtmh = Val(Form1!Grid1.Text)
5498
5499   Form1!Grid1.Col = 10
5500   Cond. Mecha = Val(Form1!Grid1.Text)
5501
5502   Form1!Grid1.Col = 11
5503   Cond. Cycle = Val(Form1!Grid1.Text)
5504
5505   Form1!Grid1.Col = 16
5506   Cond. MethodName = Form1!Grid1.Text
5507
5508   Call Mthd                               ' 変換:評価法名->評価法番号
5509
5510   Form1!Grid1.Col = 17
5511   Cond. AnalSheetName = Form1!Grid1.Text
5512
5513   Form1!Grid1.Col = 18
5514   Cond. AnalSheetRow = Val(Form1!Grid1.Text)
5515
5516   Form1!Grid1.Col = 19
5517   Cond. AnalSheetCol = Val(Form1!Grid1.Text)
5518
5519   If Cond. AnalSheetCol > 65 Then
5520     Cond. AnalSheetCol = 65
5521   End If
5522
5523   Form1!Grid1.Col = 20
5524   Cond. OutSheetName = Form1!Grid1.Text
5525
5526   Form1!Text1.LinkMode = NONE
5527
5528   If Cond. AnalSheetRow < IN Then
5529     Lcount = 1

```

```

5530 Elseif (Cond. AnalSheetRow Mod IN) > 0 Then
5531     Lcount = Int(Cond. AnalSheetRow / IN)
5532     Lcount = Int(Cond. AnalSheetRow / IN) + 1
5533 Else
5534     Lcount = Int(Cond. AnalSheetRow / IN)
5535 End If
5536
5537 End Sub
5538
5539
5540 (2) Range 解析結果シート取得 (DDE通信:解析.AN?)
5541
5542 Sub Input_Data_2 (Ino, Ioffset1)
5543     Dim Ino, Ioffset1 As Integer
5544     Dim I, Il, J, L, Ioffset2 As Integer
5545
5546     Il = Ino * IN           ' Excelシート・開始行数
5547     Ioffset2 = 6           ' Excelシート・オフセット行数1 : 6行目~
5548
5549     For I = 1 To Ioffset1
5550         L = I - 1 + Il + Ioffset2
5551         Form1!Text1.LinkMode = NONE
5552         Form1!Text1.LinkTopic = "Excel|[ 解析.AN1]解析"
5553         Form1!Text1.LinkTopic = "Excel|[ " & Cond. AnalSheetName & "]" & Cond. AnalSheetName
5554         Form1!Text1.LinkTopic = "Excel|" & Cond. AnalSheetName
5555         Form1!Text1.LinkItem = "R" & L & "C1:R" & L & "C" & 95
5556         Form1!Text1.LinkItem = "R" & L & "C1:R" & L & "C" & Cond. AnalSheetCol
5557         Form1!Text1.LinkMode = LINK_AUTOMATIC
5558         Form1!Grid1.SelStartRow = I
5559         Form1!Grid1.SelStartCol = 1
5560         Form1!Grid1.SelEndRow = I
5561         Form1!Grid1.SelEndRow = I
5562         Form1!Grid1.SelEndRow = I
5563         Form1!Grid1.SelEndCol = 95
5564         Form1!Grid1.SelEndCol = Cond. AnalSheetCol
5565         Form1!Grid1.Clip = Form1!Text1.Text
5566     Next I
5567
5568     For I = 0 To Ioffset1 - 1
5569         Form1!Grid1.Row = I + 1
5570         Form1!Grid1.Col = 3
5571         Analy.R(I, 0) = Val(Form1!Grid1.Text) ' 起点1:R座標
5572         Form1!Grid1.Col = 4
5573         Analy.Z(I, 0) = Val(Form1!Grid1.Text) ' :Z座標
5574         Form1!Grid1.Col = 5
5575         Analy.H(I, 0) = Val(Form1!Grid1.Text) ' :H座標
5576         Form1!Grid1.Col = 9
5577         Analy.Tmpmax(I) = Val(Form1!Grid1.Text) ' 最高金属温度
5578     Next I
5579
5580     For I = 0 To ((Cond. AnalSheetCol - 9) / 28) - 1
5581
5582         L = 10 + I * 28
5583
5584         For J = 0 To Ioffset1 - 1
5585             Form1!Grid1.Row = J + 1
5586             Form1!Grid1.Col = L
5587             Analy.Ptime(J, I) = Val(Form1!Grid1.Text) ' 時点1:評価時刻
5588             Form1!Grid1.Col = L + 1
5589             Analy.Temp(J, I) = Val(Form1!Grid1.Text) ' :表面温度
5590             Form1!Grid1.Col = L + 2
5591             Analy.Ltemp(J, I) = Val(Form1!Grid1.Text) ' :等価線形温度
5592             Form1!Grid1.Col = L + 3

```

```

5593      Analy. Mtemp(J, 1) = Val(Form1!Grid1.Text) ' : 平均温度
5594
5595      Form1!Grid1.Col = L + 4
5596      Analy. Strs(J, 0, 1) = Val(Form1!Grid1.Text) ' 時点1:表面応力R
5597      Form1!Grid1.Col = L + 5
5598      Analy. Strs(J, 1, 1) = Val(Form1!Grid1.Text) ' : Z
5599      Form1!Grid1.Col = L + 6
5600      Analy. Strs(J, 2, 1) = Val(Form1!Grid1.Text) ' : H
5601      Form1!Grid1.Col = L + 7
5602      Analy. Strs(J, 3, 1) = Val(Form1!Grid1.Text) ' : RZ
5603
5604      Form1!Grid1.Col = L + 8
5605      Analy. Lstrs(J, 0, 1) = Val(Form1!Grid1.Text) ' 時点1:等価線形応力R
5606      Form1!Grid1.Col = L + 9
5607      Analy. Lstrs(J, 1, 1) = Val(Form1!Grid1.Text) ' : Z
5608      Form1!Grid1.Col = L + 10
5609      Analy. Lstrs(J, 2, 1) = Val(Form1!Grid1.Text) ' : H
5610      Form1!Grid1.Col = L + 11
5611      Analy. Lstrs(J, 3, 1) = Val(Form1!Grid1.Text) ' : RZ
5612
5613      Form1!Grid1.Col = L + 12
5614      Analy. Mstrs(J, 0, 1) = Val(Form1!Grid1.Text) ' 時点1:膜応力R
5615      Form1!Grid1.Col = L + 13
5616      Analy. Mstrs(J, 1, 1) = Val(Form1!Grid1.Text) ' : Z
5617      Form1!Grid1.Col = L + 14
5618      Analy. Mstrs(J, 2, 1) = Val(Form1!Grid1.Text) ' : H
5619      Form1!Grid1.Col = L + 15
5620      Analy. Mstrs(J, 3, 1) = Val(Form1!Grid1.Text) ' : RZ
5621
5622      Form1!Grid1.Col = L + 16
5623      Analy. Ee(J, 0, 1) = Val(Form1!Grid1.Text) / 100# ' 時点1:弾性歪R
5624      Form1!Grid1.Col = L + 17
5625      Analy. Ee(J, 1, 1) = Val(Form1!Grid1.Text) / 100# ' : Z
5626      Form1!Grid1.Col = L + 18
5627      Analy. Ee(J, 2, 1) = Val(Form1!Grid1.Text) / 100# ' : H
5628      Form1!Grid1.Col = L + 19
5629      Analy. Ee(J, 3, 1) = Val(Form1!Grid1.Text) / 100# ' : RZ
5630
5631      Form1!Grid1.Col = L + 20
5632      Analy. Ep(J, 0, 1) = Val(Form1!Grid1.Text) / 100# ' 時点1:塑性歪R
5633      Form1!Grid1.Col = L + 21
5634      Analy. Ep(J, 1, 1) = Val(Form1!Grid1.Text) / 100# ' : Z
5635      Form1!Grid1.Col = L + 22
5636      Analy. Ep(J, 2, 1) = Val(Form1!Grid1.Text) / 100# ' : H
5637      Form1!Grid1.Col = L + 23
5638      Analy. Ep(J, 3, 1) = Val(Form1!Grid1.Text) / 100# ' : RZ
5639
5640      Form1!Grid1.Col = L + 24
5641      Analy. EC(J, 0, 1) = Val(Form1!Grid1.Text) / 100# ' 時点1:外-7°歪R
5642      Form1!Grid1.Col = L + 25
5643      Analy. EC(J, 1, 1) = Val(Form1!Grid1.Text) / 100# ' : Z
5644      Form1!Grid1.Col = L + 26
5645      Analy. EC(J, 2, 1) = Val(Form1!Grid1.Text) / 100# ' : H
5646      Form1!Grid1.Col = L + 27
5647      Analy. EC(J, 3, 1) = Val(Form1!Grid1.Text) / 100# ' : RZ
5648
5649      Form1!Grid1.Col = 95 ' 95桁A固定
5650      Analy. Q(J) = Val(Form1!Grid1.Text) ' 時点1:Q
5651
5652      Next J
5653
5654      Next I
5655

```

```

5656 Form1!Text1.LinkMode = NONE
5657
5658 End Sub
5659
5660 '
5661 ' 処理状況表示ルーチン
5662 '
5663 Sub Msg_proc (N, Lcnt, I)
5664 ' Dim N, Lcnt, I As Integer
5665
5666 Select Case N
5667 Case 0
5668 Form1.Text3.Text = "解析データ入力中!(" & Str(I) & "/" & Str(Lcnt) & ")"
5669 Case 1
5670 Form1.Text3.Text = "基本変数設定中!(" & Str(I) & "/" & Str(Lcnt) & ")"
5671 Case 2
5672 Form1.Text3.Text = "損傷値解析中!(" & Str(I) & "/" & Str(Lcnt) & ")"
5673 Case 3
5674 Form1.Text3.Text = "DDE通信準備中!(" & Str(I) & "/" & Str(Lcnt) & ")"
5675 Case 4
5676 Form1.Text3.Text = "DDE通信中!(" & Str(I) & "/" & Str(Lcnt) & ")"
5677 Case 5
5678 Case 6
5679 Case 99
5680 Form1.Text3.Text = "終了!"
5681 Case Else
5682 End Select
5683 End Sub
5684
5685 '
5686 ' 処理状況表示ルーチン
5687 '
5688 Sub Msg_proc0 (Message)
5689 ' Dim N As Integer
5690 Form1!Text1.Text = Message
5691 End Sub
5692
5693 '
5694 ' 処理状況表示ルーチン
5695 '
5696 Sub Msg_procl (Message)
5697 ' Dim N As Integer
5698 Form1!Text4.Text = Message
5699 End Sub
5700
5701 '
5702 ' 評価法名判定 : 16評価法(キーワード以下固定)
5703 '
5704 Sub Mthd ()
5705 If Cond.MethodName = "TSDS" Then ' No. 1
5706 Cond.MethodNo = 0
5707 ElseIf Cond.MethodName = "BDS" Then ' No. 2
5708 Cond.MethodNo = 1
5709 ElseIf Cond.MethodName = "BDSMD" Then ' No. 3
5710 Cond.MethodNo = 2
5711 ElseIf Cond.MethodName = "DDS" Then ' No. 4
5712 Cond.MethodNo = 3
5713 ElseIf Cond.MethodName = "DDSDWLD" Then ' No. 5
5714 Cond.MethodNo = 4
5715 ElseIf Cond.MethodName = "TYPE1" Then ' No. 6
5716 Cond.MethodNo = 5
5717 ElseIf Cond.MethodName = "TYPE2" Then ' No. 7
5718 Cond.MethodNo = 6

```



```

5719   Elseif Cond. MethodName = "EQEF" Then      ' No. 8
5720       Cond. MethodNo = 7
5721   Elseif Cond. MethodName = "EQEFWD" Then    ' No. 9
5722       Cond. MethodNo = 8
5723   Elseif Cond. MethodName = "PNC" Then       ' No. 10
5724       Cond. MethodNo = 9
5725   Elseif Cond. MethodName = "EQEF15" Then    ' No. 11
5726       Cond. MethodNo = 10
5727   Elseif Cond. MethodName = "EQEF167" Then   ' No. 12
5728       Cond. MethodNo = 11
5729   Elseif Cond. MethodName = "INELAST" Then   ' No. 13
5730       Cond. MethodNo = 12
5731   Elseif Cond. MethodName = "SIEQ167" Then   ' No. 14
5732       Cond. MethodNo = 13
5733   Elseif Cond. MethodName = "EQEF167W" Then  ' No. 15
5734       Cond. MethodNo = 14
5735   Elseif Cond. MethodName = "EQEF16WK" Then  ' No. 15
5736       Cond. MethodNo = 15
5737   End If
5738 End Sub
5739
5740
5741   材料名判定 : 6材料(キーワード以下固定)
5742
5743 Sub Mtrl ()
5744   If Cond. Material = "SUS304" Then            ' No. 1
5745       Mc. Im = 304
5746       Cond. Idate = 1
5747   Elseif Cond. Material = "SUS316" Then        ' No. 2
5748       Mc. Im = 316
5749       Cond. Idate = 2
5750   Elseif Cond. Material = "SUS321" Then        ' No. 2
5751       Mc. Im = 321
5752       Cond. Idate = 3
5753   Elseif Cond. Material = "SCMV4" Then         ' No. 3
5754       Mc. Im = 2250
5755       Cond. Idate = 4                          ' True?
5756   Elseif Cond. Material = "STBA24" Then       ' No. 4
5757       Mc. Im = 2251
5758       Cond. Idate = 5                          ' True?
5759   Elseif Cond. Material = "MOD9CR-1W0" Then   ' No. 5
5760       Mc. Im = 91
5761       Cond. Idate = 4
5762   Elseif Cond. Material = "SUS316FR" Then      ' No. 6
5763       Mc. Im = 3160
5764       Cond. Idate = 2
5765   End If
5766 End Sub
5767
5768
5769   最大値判定
5770
5771 Function Nmax (A, B, C)
5772   Dim A, B, C      As Single
5773   Dim D            As Single
5774   If A < B Then
5775       D = B
5776   Else
5777       D = A
5778   End If
5779   If D < C Then
5780       Nmax = C
5781   Else

```

```

5782     Nmax = D
5783     End If
5784 End Function
5785
5786 '
5787 '   最小値判定
5788 '
5789 Function Nmin (A, B, C)
5790 '   Dim A, B, C           As Single
5791   Dim D                 As Single
5792   If A > B Then
5793     D = B
5794   Else
5795     D = A
5796   End If
5797   If D > C Then
5798     Nmin = C
5799   Else
5800     Nmin = D
5801   End If
5802 End Function
5803
5804 '
5805 '   DDE通信(出力) : 損傷解析結果データ 対 MS-EXCELシート
5806 '
5807 Sub Outdat (Ino, Icount, Colum)
5808 '   Dim Ino, Icount, Colum As Integer
5809   Dim I, I1, L, Offset2 As Integer
5810
5811   I1 = Ino * IN           ' Excelシート・開始行数
5812   Offset2 = 6            ' Excelシート・オフセット行数1
5813
5814   For I = 1 To Icount
5815     L = I - 1 + I1 + Offset2
5816     Form1!Grid1.SelStartRow = I
5817     Form1!Grid1.SelStartCol = 1
5818     Form1!Grid1.SelEndRow = I
5819     Form1!Grid1.SelEndCol = Colum
5820     Form1!Text1.Text = Form1!Grid1.Clip
5821     Form1!Text1.LinkMode = NONE
5822     Form1!Text1.LinkTopic = "Excel|[ 損傷値0. D21]損傷値0"
5823     Form1!Text1.LinkTopic = "Excel|[ " & Cond.OutSheetName & "]" & Cond.OutSheetName
5824     Form1!Text1.LinkTopic = "Excel|" & Cond.OutSheetName
5825     Form1!Text1.LinkItem = "R" & L & "C1:R" & L & "C" & Colum
5826     Form1!Text1.LinkMode = LINK_MANUAL
5827     Form1!Text1.LinkPoke
5828   Next I
5829
5830   Form1!Text1.LinkMode = NONE
5831
5832 End Sub
5833
5834 '
5835 '   損傷計算結果データ -> Grid.Text7°ロハ°ティ代入
5836 '
5837 Sub Pncout (Icount)
5838 '   Dim Icount As Integer
5839   Dim I       As Integer
5840
5841   For I = 0 To Icount - 1
5842     Form1!Grid1.Row = I + 1
5843     Form1!Grid1.Col = 1
5844     Form1!Grid1.Text = Analy.R(I, 0)

```

```

5845 Forml!Gridl. Col = 2
5846 Forml!Gridl. Text = Analy. Z(I, 0)
5847 Forml!Gridl. Col = 3
5848 Forml!Gridl. Text = Analy. H(I, 0)
5849 Forml!Gridl. Col = 4
5850 Forml!Gridl. Text = Allmtd. Spx(I)
5851 Forml!Gridl. Col = 5
5852 Forml!Gridl. Text = Allmtd. Sn(I)
5853 Forml!Gridl. Col = 6
5854 Forml!Gridl. Text = Allmtd. K(I)
5855 Forml!Gridl. Col = 7
5856 Forml!Gridl. Text = Allmtd. Ked(I)
5857 Forml!Gridl. Col = 8
5858 Forml!Gridl. Text = Allmtd. Ke(I)
5859 Forml!Gridl. Col = 9
5860 Forml!Gridl. Text = Allmtd. Kedg(I)
5861 Forml!Gridl. Col = 10
5862 Forml!Gridl. Text = Allmtd. Kedl(I)
5863 Forml!Gridl. Col = 11
5864 Forml!Gridl. Text = Allmtd. Ef(I)
5865 Forml!Gridl. Col = 12
5866 Forml!Gridl. Text = Allmtd. Et(I)
5867 Forml!Gridl. Col = 13
5868 Forml!Gridl. Text = Allmtd. Si(I)
5869 Forml!Gridl. Col = 14
5870 Forml!Gridl. Text = Allmtd. Qn(I)
5871 Forml!Gridl. Col = 15
5872 Forml!Gridl. Text = Allmtd. Ql(I)
5873 Forml!Gridl. Col = 16
5874 Forml!Gridl. Text = Allmtd. Qnu(I)
5875 Forml!Gridl. Col = 17
5876 Forml!Gridl. Text = Allmtd. Qeff(I)
5877 Forml!Gridl. Col = 18
5878 Forml!Gridl. Text = Allmtd. Df(I)
5879 Forml!Gridl. Col = 19
5880 Forml!Gridl. Text = Allmtd. Dc(I)
5881 Forml!Gridl. Col = 20
5882 Forml!Gridl. Text = Allmtd. D(I)
5883 Forml!Gridl. Col = 21
5884 Forml!Gridl. Text = Allmtd. Dcr(I)
5885 Next I
5886
5887 End Sub
5888
5889 '
5890 ' PNC
5891 ' (中間保持考慮、弾性追従パラメータ最適化、ひずみ集中係数改良)
5892 ' EQEFに工学的安全係数追加
5893 ' 平成3年11月21日 Ver. 1. 5
5894 '
5895 Sub Pncp (Icount)
5896 '
5897 ' Dim Icount As Integer
5898 ' Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
5899 ' Dim KEDD As Single
5900 ' Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
5901 ' Dim K1, K2, KE1, KE2 As Single
5902 ' Dim K1D, K2D, KED1D, KED2D As Single
5903 ' Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
5904 ' Dim LSIG1, LSIG2, LSIG3 As Single
5905 ' Dim NU, NU1, NU2, NF As Single
5906 '
5907 ' Iflg = 0

```

```

5908
5909   For I = 0 To Icount - 1
5910   -----
5911   歪範囲の計算
5912   -----
5913   -----  T I M E 1
5914       Mc.Tc = Analy.Mtemp(I, 0)
5915       Call Mset(Iflg, Mc)
5916       E1 = Mc.E
5917
5918       NU1 = Calcnu(Mc)
5919       ALPHA1 = Alpha(Mc)
5920   -----  T I M E 2
5921
5922       Mc.Tc = Analy.Mtemp(I, 1)
5923       Call Mset(Iflg, Mc)
5924       E2 = Mc.E
5925
5926       NU2 = Calcnu(Mc)
5927       ALPHA2 = Alpha(Mc)
5928
5929   熱ピークを含むピーク応力強さの計算
5930   SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
5931   SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
5932   SIGH = Analy.Strs(I, 2, 0) - Analy.Strs(I, 2, 1)
5933   SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
5934   SIG1 = Prin1(SIGR, SIGZ, SIGT)
5935   SIG2 = Prin2(SIGR, SIGZ, SIGT)
5936   SIG3 = SIGH
5937
5938   SPP = Ftres(SIG1, SIG2, SIG3)
5939
5940   熱ピークを含まないピーク応力強さの計算
5941   SF1 = (-E1 * ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0))) / (1 - NU1)
5942   SF2 = (-E2 * ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2)
5943   Allmtd.Ef(I) = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *
5944   (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
5945   SANR = Abs(Analy.Strs(I, 0, 1) - Analy.Strs(I, 0, 0))
5946   SANZ = Abs(Analy.Strs(I, 1, 1) - Analy.Strs(I, 1, 0))
5947   If SANR >= SANZ Then
5948       SIGR = (Analy.Strs(I, 0, 0) - SF1) - (Analy.Strs(I, 0, 1) - SF2)
5949       SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
5950   Else
5951       SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
5952       SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)
5953   End If
5954
5955   SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
5956   SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
5957   SIG1 = Prin1(SIGR, SIGZ, SIGT)
5958   SIG2 = Prin2(SIGR, SIGZ, SIGT)
5959   SIG3 = SIGH
5960
5961   Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
5962   ----- 公称応力強さの計算
5963   Allmtd.Sn(I) = Bscval.Ltres(I)
5964   ----- ピーク応力を含む応力集中係数の計算
5965   If Allmtd.Sn(I) = 0! Then
5966       KS = 0!
5967   Else
5968       KS = SPP / Allmtd.Sn(I)
5969   End If

```

```

5970
5971      KSTAR = Smax(KS, 1!, 1!)
5972  -----ピーク応力を含まない応力集中係数の計算
5973      If Allmtd. Sn(I) = 0! Then
5974          KD = 0!
5975      Else
5976          KD = Allmtd. Spx(I) / Allmtd. Sn(I)
5977      End If
5978
5979      Allmtd. K(I) = Smax(KD, 1!, 1!)
5980
5981  -----最高金属温度による  $\epsilon_n$ 、 $S_{mh}$ 、 $S_{rh}$  の計算
5982      Mc. Tc = Analy. Tmpmax(I)
5983      Call Mset(Iflg, Mc)
5984      EN = Allmtd. Sn(I) / Mc. E
5985      SMH = Mc. Sm
5986
5987      Call R1x1(1.5 * Mc. Sm, 0!, 1!, 1!, 1!, Cond. Hldtmh)
5988      Call R1x1(1.5*Mc. Sm, 0. 0. 1. 0. 1. 0. 1. 0, Cond. Hldtmh, S2xx(0), Dcxx(0), Mc)
5989      SX1 = S2xx(0)
5990
5991  -----コールド温度による  $S_{mc}$ 、 $S_{rc}$  の計算
5992      Mc. Tc = Analy. Temp(I, 1)
5993      Call Mset(Iflg, Mc)
5994      SMC = Mc. Sm
5995
5996      Call R1x1(1.5 * Mc. Sm, 0!, 1!, 1!, 1!, Cond. Hldtmc)
5997      Call R1x1(1.5*Mc. Sm, 0. 0. 1. 0. 1. 0. 1. 0, Cond. Hldtmc, S2xx(1), Dcxx(0), Mc)
5998      SX2 = S2xx(0)
5999
6000  -----ひずみ集中係数の計算
6001      SM3BAR = SX1 + SX2
6002      Allmtd. Qn(I) = Analy. Q(I)
6003      NU = .3
6004
6005      Allmtd. Qnu(I) = 2! * (1! - NU)
6006      QNUSTR = 5! / 3!
6007      S1 = Smax(KSTAR ^ (3! / 4!), Allmtd. Qnu(I), Allmtd. Qnu(I))
6008      S2 = Smax(Allmtd. K(I) ^ (3! / 4!), QNUSTR, QNUSTR)
6009      Allmtd. Q1(I) = Nmin(S1, S2, S2)
6010      If Allmtd. Sn(I) = 0! Then
6011          KEDD! = 0!
6012      Else
6013          KEDD! = SM3BAR / Allmtd. Sn(I)
6014      End If
6015
6016      Allmtd. Ked1(I) = Smax(1! + (Allmtd. Q1(I) - 1!) * (1! - KEDD! / KSTAR), 1!, 1!)
6017      Allmtd. Kedg(I) = Smax(1! + (Allmtd. Qn(I) - 1!) * (1! - KEDD!), 1!, 1!)
6018
6019      Allmtd. Ke(I) = KSTAR * Allmtd. Ked1(I) * Allmtd. Kedg(I)
6020
6021      Allmtd. Et(I) = Allmtd. Ke(I) * EN
6022  -----
6023      疲労損傷の計算
6024  -----
6025  -----ファンクションANFの計算機能の限界
6026      If Allmtd. Et(I) < .0005 Then Allmtd. Et(I) = .0005
6027
6028  -----最高金属温度
6029      Mc. Tc = Analy. Tmpmax(I)
6030      Call Mset(Iflg, Mc)
6031
6032      EDOT = 2! * Allmtd. Et(I) / ((Cond. Hldtmc + Cond. Hldtmh) * 3600!)

```

```

6033      If EDOT < .000001 Then EDOT = .000001
6034      S1 = Anf(2! * Allmtd.Et(I), EDOT, Mc)
6035      S2 = 1! / 20! * Anf(Allmtd.Et(I), EDOT, Mc)
6036      NF = Nmin(S1, S2, S2)
6037
6038      Allmtd.Df(I) = Cond.Cycle / NF
6039
6040      クリーブ損傷の計算
6041
6042      DCNの計算
6043      DCN = 0
6044      DCPの計算
6045      初期応力の計算
6046      TIME 1.
6047
6048
6049      熱ピークを含むピーク応力強さの計算
6050      SIGR = Analy.Strs(I, 0, 0)
6051      SIGZ = Analy.Strs(I, 1, 0)
6052      SIGH = Analy.Strs(I, 2, 0)
6053      SIGT = Analy.Strs(I, 3, 0)
6054      SIG1 = Prin1(SIGR, SIGZ, SIGT)
6055      SIG2 = Prin2(SIGR, SIGZ, SIGT)
6056      SIG3 = SIGH
6057
6058      SPP1 = Ftres(SIG1, SIG2, SIG3)
6059
6060      熱ピークを含まないピーク応力強さの計算
6061      EF1 = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1))
6062      If Analy.Strs(I, 0, 0) >= Analy.Strs(I, 1, 0) Then
6063          SIGR = Analy.Strs(I, 0, 0) - SF1
6064          SIGZ = Analy.Strs(I, 1, 0)
6065      Else
6066          SIGR = Analy.Strs(I, 0, 0)
6067          SIGZ = Analy.Strs(I, 1, 0) - SF1
6068      End If
6069
6070      SIGH = Analy.Strs(I, 2, 0) - SF1
6071      SIGT = Analy.Strs(I, 3, 0)
6072      SIG1 = Prin1(SIGR, SIGZ, SIGT)
6073      SIG2 = Prin2(SIGR, SIGZ, SIGT)
6074      SIG3 = SIGH
6075
6076      SP1 = Ftres(SIG1, SIG2, SIG3)
6077      公称応力強さの計算
6078      LSIGR = Analy.Lstrs(I, 0, 0)
6079      LSIGZ = Analy.Lstrs(I, 1, 0)
6080      LSIGH = Analy.Lstrs(I, 2, 0)
6081      LSIGT = Analy.Lstrs(I, 3, 0)
6082      LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
6083      LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
6084      LSIG3 = LSIGH
6085      SN1 = Ftres(LSIG1, LSIG2, LSIG3)
6086      EN1 = SN1 / E1
6087
6088      ピーク応力を含む応力集中係数の計算
6089      If SN1 = 0! Then
6090          KS1 = 0!
6091      Else
6092          KS1 = SPP1 / SN1
6093      End If
6094
6095      KSTAR1 = Smax(KS1, 1!, 1!)

```

```

6096      ピーク応力を含まない応力集中係数の計算
6097      If SN1 = 0! Then
6098          K1D = 0!
6099      Else
6100          K1D = SP1 / SN1
6101      End If
6102
6103      K1 = Smax(K1D, 1!, 1!)
6104      ひずみ集中係数の計算
6105      SM3 = 1.5 * SMH + 1.5 * SMC
6106      If SN1 = 0! Then
6107          KED1D = 0!
6108      Else
6109          KED1D = SM3 / SN1
6110      End If
6111
6112      KEDL1 = Smax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED1D / KSTAR1), 1!, 1!)
6113      KEDG1 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED1D), 1!, 1!)
6114
6115      KE1 = KSTAR1 * KEDL1 * KEDG1
6116      ETD1 = KE1 * EN1
6117      ----- T I M E 2
6118
6119      熱ピークを含むピーク応力強さの計算
6120      SIGR = Analy.Strs(I, 0, 1)
6121      SIGZ = Analy.Strs(I, 1, 1)
6122      SIGH = Analy.Strs(I, 2, 1)
6123      SIGT = Analy.Strs(I, 3, 1)
6124      SIG1 = Prin1(SIGR, SIGZ, SIGT)
6125      SIG2 = Prin2(SIGR, SIGZ, SIGT)
6126      SIG3 = SIGH
6127
6128      SPP2 = Ftres(SIG1, SIG2, SIG3)
6129
6130      熱ピークを含まないピーク応力強さの計算
6131      EF2 = Abs(ALPHA2*(Analy.Temp(I,1)-Analy.Ltemp(I,1)))/(1-NU2))
6132      If Analy.Strs(I, 0, 1) >= Analy.Strs(I, 1, 1) Then
6133          SIGR = Analy.Strs(I, 0, 1) - SF2
6134          SIGZ = Analy.Strs(I, 1, 1)
6135      Else
6136          SIGR = Analy.Strs(I, 0, 1)
6137          SIGZ = Analy.Strs(I, 1, 1) - SF2
6138      End If
6139
6140      SIGH = Analy.Strs(I, 2, 1) - SF2
6141      SIGT = Analy.Strs(I, 3, 1)
6142      SIG1 = Prin1(SIGR, SIGZ, SIGT)
6143      SIG2 = Prin2(SIGR, SIGZ, SIGT)
6144      SIG3 = SIGH
6145
6146      SP2 = Ftres(SIG1, SIG2, SIG3)
6147      公称応力強さの計算
6148      LSIGR = Analy.Lstrs(I, 0, 1)
6149      LSIGZ = Analy.Lstrs(I, 1, 1)
6150      LSIGH = Analy.Lstrs(I, 2, 1)
6151      LSIGT = Analy.Lstrs(I, 3, 1)
6152      LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
6153      LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
6154      LSIG3 = LSIGH
6155      SN2 = Ftres(LSIG1, LSIG2, LSIG3)
6156      EN2 = SN2 / E2
6157
6158      ピーク応力を含む応力集中係数の計算

```

```

6159      If SN2 = 0! Then
6160          KS2 = 0!
6161      Else
6162          KS2 = SPP2 / SN2
6163      End If
6164
6165      KSTAR2 = Smax(KS2, 1!, 1!)
6166      ピーク応力を含まない応力集中係数の計算
6167      If SN2 = 0! Then
6168          K2D = 0!
6169      Else
6170          K2D = SP2 / SN2
6171      End If
6172
6173      K2 = Smax(K2D, 1!, 1!)
6174      ひずみ集中係数の計算
6175      If SN2 = 0! Then
6176          KED2D = 0!
6177      Else
6178          KED2D = SM3 / SN2
6179      End If
6180
6181      KEDL2 = Smax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED2D / KSTAR2), 1!, 1!)
6182      KEDG2 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED2D), 1!, 1!)
6183
6184      KE2 = KSTAR2 * KEDL2 * KEDG2
6185      ETD2 = KE2 * EN2
6186
6187      ETC = Smax(ETD1, ETD2, ETD2)
6188
6189      Allmtd.Si(I) = Sepr(ETC, Mc)
6190      Allmtd.Si(I) = Sepr(ETC, Mc) / 2!
6191
6192      Allmtd.Si(I) = Allmtd.Si(I) + Cond.Mecha
6193
6194      EC1 = 0!
6195
6196      ALPHC = 1! / 3!
6197      ALPHR = 10!
6198      DT = Cond.Hldtmh
6199
6200      Allmtd.Qeff(I) = Allmtd.Q1(I) * Allmtd.Qn(I)
6201      Call R1x1(Allmtd.Si(I), 0!, 1! / 3!, 10!, Allmtd.Qeff(I), Cond.Hldtmh)
6202      Call R1x1(Allmtd.Si(I), EC1, ALPHC, ALPHR, Allmtd.Qeff(I), DT, S2xx(0), Dcxx(0), Mc)
6203      DCP = Cond.Cycle * Dcxx(0)
6204
6205      Allmtd.Dc(I) = DCN + DCP
6206
6207      クリープ疲労損傷の計算
6208
6209      Allmtd.D(I) = Allmtd.Df(I) + Allmtd.Dc(I)
6210
6211      許容値の計算
6212
6213      If Allmtd.Dc(I) >= 0! And Allmtd.Dc(I) <= .3 Then
6214          Allmtd.Dcr(I) = 1! - 4! / 3! * Allmtd.Dc(I)
6215      Elseif Allmtd.Dc(I) <= 1! Then
6216          Allmtd.Dcr(I) = 3! / 7! + 4! / 7! * Allmtd.Dc(I)
6217      Elseif Allmtd.Dc(I) > 1! Then
6218          Allmtd.Dcr(I) = 1!
6219      End If
6220
6221      Next I

```



```

6222
6223 End Sub
6224
6225
6226   主応力成分差算出プログラム1
6227
6228 Function Prin1 (R, z, T)
6229   Dim R, Z, T      As Single
6230
6231   Prin1 = (R + z) / 2! + Sqr(((R - z) / 2!) ^ 2 + T ^ 2)
6232
6233 End Function
6234
6235
6236   主応力成分差算出プログラム2
6237
6238 Function Prin2 (R, z, T)
6239   Dim R, Z, T      As Single
6240
6241   Prin2 = (R + z) / 2! - Sqr(((R - z) / 2!) ^ 2 + T ^ 2)
6242
6243 End Function
6244
6245
6246   1990年02月07日 修正 浜田(FORTRAN for MS-DOS)
6247   1994年05月12日 移植 細貝(Visual Basic for MS-Windows)
6248
6249   S1=0近傍の計算機能を改良          NOV. 8/88
6250   ALPHRを追加                          FEB. 7/90
6251
6252 Sub R1x1 (S1 As Single, EC1 As Single, ALPHC As Single, ALPHR As Single, Q As Single, TMAX As Sin
gle)
6253   Dim S1, EC1, ALPHC, ALPHR, Q, TMAX As Single
6254   Dim S2(2), DC(2) As Single
6255
6256   Dim DDC As Double
6257
6258   Call Msg_procl("R1x1:S1=" & Str(S1) & "/HLDTMH=" & Str(TMAX))
6259   S2xx(0) = S1
6260   Dcxx(0) = 0!
6261
6262   JS = 1
6263   If S1 < 0 Then
6264     JS = -1
6265   End If
6266   S = JS * S1
6267
6268   SMIN = .1
6269   DSMIN = SMIN / 10!
6270   DTMAX = TMAX / 10!
6271
6272   If S <= SMIN Then
6273     Exit Sub
6274   End If
6275
6276   DDC = 0!
6277   TL = TMAX
6278   EC = EC1
6279
6280 103 :
6281
6282   Call Msg_procl("R1x1/Tcrr:S=" & Str(S))
6283

```

```

6284 TR = Tcrr(S) / ALPHR
6285
6286 Call Msg_procl("Rlx1/Tcr:S=" & Str(S) & "/EC=" & Str(EC) & "/ALPHC=" & Str(ALPHC))
6287 TZ = Tcr(S, EC, ALPHC)
6288
6289 Call Msg_procl("Rlx1/Ecrr:S=" & Str(S) & "/TZ=" & Str(TZ) & "/ALPHC=" & Str(ALPHC))
6290 EDOT = Ecrr(S, TZ, ALPHC)
6291
6292 DT = AMIN((0.01*S)/(Mc.E*EDOT), DTMAX)
6293 DT = Nmin((.01 * S) / (Mc.E * EDOT), DTMAX, DTMAX)
6294 If DT >= TL Then
6295     GoTo 203
6296 End If
6297
6298 EC = EC + EDOT * DT
6299 DDC = DDC + DT / TR
6300 TL = TL - DT
6301 S = S - Mc.E * EDOT * DT / Q
6302 GoTo 103
6303
6304 203 :
6305 EC = EC + EDOT * TL
6306 DDC = DDC + TL / TR
6307 S = S - Mc.E * EDOT * TL / Q
6308
6309 S2xx(0) = JS * S
6310 Dcxx(0) = Dcxx(0) + CSng(DDC)
6311
6312 End Sub
6313
6314
6315 S I E Q 1 6 7
6316 山下GL依頼により作成
6317 (板厚内温度勾配緩和初期応力評価 - 2直線近似応力・歪み関係利用
6318 ひずみ速度効果考慮 - HOLDTIMECを利用
6319 qL = 1.67、ひずみ集中係数改良 EQEFを改良)
6320 平成5年6月15日 N.K
6321
6322 Sub Sieq167p (Icount)
6323
6324 Dim Icount As Integer
6325 Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
6326 Dim KEDD As Single
6327 Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
6328 Dim K1, K2, KE1, KE2 As Single
6329 Dim K1D, K2D, KED1D, KED2D, KT As Single
6330 Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
6331 Dim LSIG1, LSIG2, LSIG3 As Single
6332 Dim NU, NU1, NU2, NF, HDASH As Single
6333 Dim INF As Long
6334
6335 Iflg = 0
6336
6337 For I = 0 To Icount - 1
6338 -----
6339 歪範囲の計算
6340 -----
6341 ----- T I M E 1
6342 Mc.Tc = Analy.Mtemp(I, 0)
6343 Call Mset(Iflg, Mc)
6344 E1 = Mc.E
6345 NU1 = Calcnu(Mc)
6346 ALPHA1 = Alpha(Mc)

```

```

6347 '----- T I M E 2
6348 Mc. Tc = Analy. Mtemp(I, 1)
6349 Call Mset(Iflg, Mc)
6350 E2 = Mc. E
6351 NU2 = Calcnu(Mc)
6352 ALPHA2 = Alpha(Mc)
6353 '
6354 ' 熱ピークを含むピーク応力強さの計算
6355 SIGR = Analy. Strs(I, 0, 0) - Analy. Strs(I, 0, 1)
6356 SIGZ = Analy. Strs(I, 1, 0) - Analy. Strs(I, 1, 1)
6357 SIGH = Analy. Strs(I, 2, 0) - Analy. Strs(I, 2, 1)
6358 SIGT = Analy. Strs(I, 3, 0) - Analy. Strs(I, 3, 1)
6359 SIG1 = Prin1(SIGR, SIGZ, SIGT)
6360 SIG2 = Prin2(SIGR, SIGZ, SIGT)
6361 SIG3 = SIGH
6362 SPP = Ftres(SIG1, SIG2, SIG3)
6363 '
6364 ' 熱ピークを含まないピーク応力強さの計算
6365 SF1 = (-E1 * ALPHA1 * (Analy. Temp(I, 0) - Analy. Ltemp(I, 0))) / (1 - NU1)
6366 SF2 = (-E2 * ALPHA2 * (Analy. Temp(I, 1) - Analy. Ltemp(I, 1))) / (1 - NU2)
6367 Allmtd. Ef(I) = Abs(ALPHA1 * (Analy. Temp(I, 0) - Analy. Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *
(Analy. Temp(I, 1) - Analy. Ltemp(I, 1)) / (1 - NU2))
6368 '
6369 SANR = Abs(Analy. Strs(I, 0, 1) - Analy. Strs(I, 0, 0))
6370 SANZ = Abs(Analy. Strs(I, 1, 1) - Analy. Strs(I, 1, 0))
6371 If SANR >= SANZ Then
6372     SIGR = (Analy. Strs(I, 0, 0) - SF1) - (Analy. Strs(I, 0, 1) - SF2)
6373     SIGZ = Analy. Strs(I, 1, 0) - Analy. Strs(I, 1, 1)
6374 Else
6375     SIGR = Analy. Strs(I, 0, 0) - Analy. Strs(I, 0, 1)
6376     SIGZ = (Analy. Strs(I, 1, 0) - SF1) - (Analy. Strs(I, 1, 1) - SF2)
6377 End If
6378 SIGH = (Analy. Strs(I, 2, 0) - SF1) - (Analy. Strs(I, 2, 1) - SF2)
6379 SIGT = Analy. Strs(I, 3, 0) - Analy. Strs(I, 3, 1)
6380 SIG1 = Prin1(SIGR, SIGZ, SIGT)
6381 SIG2 = Prin2(SIGR, SIGZ, SIGT)
6382 SIG3 = SIGH
6383 Allmtd. Spx(I) = Ftres(SIG1, SIG2, SIG3)
6384 '-----公称応力強さの計算
6385 Allmtd. Sn(I) = Bscval. Ltres(I)
6386 '-----ピーク応力を含む応力集中係数の計算
6387 If Allmtd. Sn(I) = 0! Then
6388     KS = 0!
6389 Else
6390     KS = SPP / Allmtd. Sn(I)
6391 End If
6392 KSTAR = Smax(KS, 1!, 1!)
6393 '-----ピーク応力を含まない応力集中係数の計算
6394 If Allmtd. Sn(I) = 0! Then
6395     KD = 0!
6396 Else
6397     KD = Allmtd. Spx(I) / Allmtd. Sn(I)
6398 End If
6399 Allmtd. K(I) = Smax(KD, 1!, 1!)
6400 '-----最高金属温度によるεn、Smh、Srhの計算
6401 Mc. Tc = Analy. Tmpmax(I)
6402 Call Mset(Iflg, Mc)
6403 EN = Allmtd. Sn(I) / Mc. E
6404 SMH = Mc. Sm
6405 Call R1x1(1.5 * Mc. Sm, 0!, 1!, 1!, 1!, Cond. Hldtmh)
6406 Call R1x1(1.5*Mc. Sm, 0.0, 1.0, 1.0, 1.0, Cond. Hldtmh, S2xx(0), Dcxx(0), Mc)
6407 SS1 = S2xx(0)
6408

```

```

6409 '
6410 '-----コールド温度による S m c、S r c の計算
6411     Mc.Tc = Analy.Temp(I, 1)
6412     Call Mset(Iflg, Mc)
6413     SMC = Mc.Sm
6414     Call R1x1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmc)
6415     Call R1x1(1.5*Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmc, S2xx(0), Dcxx(0), Mc)
6416     SS2 = S2xx(0)
6417 '
6418 '-----ひずみ集中係数の計算
6419     SM3BAR = SS1 + SS2
6420     Allmtd.Qn(I) = Analy.Q(I)
6421     Allmtd.Qn(I) = 1!
6422     NU = .3
6423     Allmtd.Qnu(I) = 2! * (1! - NU)
6424     QNUSTR = 5! / 3!
6425     S1 = Smax(KSTAR**(3.0/4.0), Allmtd.Qnu(I), Allmtd.Qnu(I))
6426     S2 = Smax(Allmtd.K(I)**(3.0/4.0), QNUSTR, QNUSTR)
6427     S3 = Smax(Allmtd.K(I)**(3.0/4.0), QNUSTR, QNUSTR)
6428     Allmtd.Q1(I) = Nmin(S1, S2, S3)
6429     Allmtd.Q1(I) = 1.67
6430     If Allmtd.Sn(I) = 0! Then
6431         KEDD = 0!
6432     Else
6433         KEDD = SM3BAR / Allmtd.Sn(I)
6434     End If
6435     Allmtd.Ked1(I) = Smax(1! + (Allmtd.Q1(I) - 1!) * (1! - KEDD / KSTAR), 1!, 1!)
6436     Allmtd.Kedg(I) = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KEDD), 1!, 1!)
6437 '
6438     Allmtd.Ke(I) = KSTAR * Allmtd.Ked1(I) * Allmtd.Kedg(I)
6439 '
6440     Allmtd.Et(I) = Allmtd.Ke(I) * EN
6441 '
6442     Res = Debug_disp("[Sieq157]/Ked1=" & Str(Allmtd.Ked1(I)) & "/Kedg=" & Str(Allmtd.Kedg(I)) & "/KSTAR=" & Str(KSTAR) & "/Ke=" & Str(Allmtd.Ke(I))) for Debug
6443     If Res = 9 Then Exit Sub
6444 '
6445 '-----
6446 ' 疲労損傷の計算
6447 '-----
6448 '-----ファンクションANFの計算機能の限界
6449     If Allmtd.Et(I) < .0005 Then Allmtd.Et(I) = .0005
6450 '
6451 '-----最高金属温度
6452     Mc.Tc = Analy.Tmpmax(I)
6453     Call Mset(Iflg, Mc)
6454 '
6455     EDOT = 2! * Allmtd.Et(I) / (Cond.Hldtmc * 3600!)
6456     NF = Anf(Allmtd.Et(I), EDOT, Mc)
6457 '
6458     Allmtd.Df(I) = Cond.Cycle / NF
6459 '-----
6460 ' クリープ損傷の計算
6461 '-----
6462 '-----DCNの計算
6463     DCN = 0
6464 '-----DCPの計算
6465 '-----初期応力の計算
6466     TIME1
6467 '
6468 '
6469 ' 熱ピークを含むピーク応力強さの計算
6470     SIGR = Analy.Strs(I, 0, 0)

```

1993.06.16

```

6471 SIGZ = Analy.Strs(I, 1, 0)
6472 SIGH = Analy.Strs(I, 2, 0)
6473 SIGT = Analy.Strs(I, 3, 0)
6474 SIG1 = Prin1(SIGR, SIGZ, SIGT)
6475 SIG2 = Prin2(SIGR, SIGZ, SIGT)
6476 SIG3 = SIGH
6477 SPP1 = Ftres(SIG1, SIG2, SIG3)
6478
6479 熱ピークを含まないピーク応力強さの計算
6480 EF1 = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1))
6481 If Analy.Strs(I, 0, 0) >= Analy.Strs(I, 1, 0) Then
6482     SIGR = Analy.Strs(I, 0, 0) - SF1
6483     SIGZ = Analy.Strs(I, 1, 0)
6484 Else
6485     SIGR = Analy.Strs(I, 0, 0)
6486     SIGZ = Analy.Strs(I, 1, 0) - SF1
6487 End If
6488 SIGH = Analy.Strs(I, 2, 0) - SF1
6489 SIGT = Analy.Strs(I, 3, 0)
6490 SIG1 = Prin1(SIGR, SIGZ, SIGT)
6491 SIG2 = Prin2(SIGR, SIGZ, SIGT)
6492 SIG3 = SIGH
6493 SP1 = Ftres(SIG1, SIG2, SIG3)
6494 -----公称応力強さの計算
6495 LSIGR = Analy.Lstrs(I, 0, 0)
6496 LSIGZ = Analy.Lstrs(I, 1, 0)
6497 LSIGH = Analy.Lstrs(I, 2, 0)
6498 LSIGT = Analy.Lstrs(I, 3, 0)
6499 LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
6500 LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
6501 LSIG3 = LSIGH
6502 SN1 = Ftres(LSIG1, LSIG2, LSIG3)
6503 EN1 = SN1 / E1
6504
6505 ピーク応力を含む応力集中係数の計算
6506 If SN1 = 0! Then
6507     KS1 = 0!
6508 Else
6509     KS1 = SPP1 / SN1
6510 End If
6511 KSTAR1 = Smax(KS1, 1!, 1!)
6512 ピーク応力を含まない応力集中係数の計算
6513 If SN1 = 0! Then
6514     K1D = 0!
6515 Else
6516     K1D = SP1 / SN1
6517 End If
6518 K1 = Smax(K1D, 1!, 1!)
6519 ひずみ集中係数の計算
6520 SM3 = 1.5 * SMH + 1.5 * SMC
6521 If SN1 = 0! Then
6522     KED1D = 0!
6523 Else
6524     KED1D = SM3 / SN1
6525 End If
6526 KEDL1 = Smax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED1D / KSTAR1), 1!, 1!)
6527 KEDG1 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED1D), 1!, 1!)
6528
6529 KE1 = KSTAR1 * KEDL1 * KEDG1
6530 ETD1 = KE1 * EN1
6531 ----- T I M E 2
6532
6533 熱ピークを含むピーク応力強さの計算

```

```

6534     SIGR = Analy.Strs(I, 0, 1)
6535     SIGZ = Analy.Strs(I, 1, 1)
6536     SIGH = Analy.Strs(I, 2, 1)
6537     SIGT = Analy.Strs(I, 3, 1)
6538     SIG1 = Prin1(SIGR, SIGZ, SIGT)
6539     SIG2 = Prin2(SIGR, SIGZ, SIGT)
6540     SIG3 = SIGH
6541     SPP2 = Ftres(SIG1, SIG2, SIG3)
6542
6543     熱ピークを含まないピーク応力強さの計算
6544     EF2 = Abs(ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
6545     If Analy.Strs(I, 0, 1) >= Analy.Strs(I, 1, 1) Then
6546         SIGR = Analy.Strs(I, 0, 1) - SF2
6547         SIGZ = Analy.Strs(I, 1, 1)
6548     Else
6549         SIGR = Analy.Strs(I, 0, 1)
6550         SIGZ = Analy.Strs(I, 1, 1) - SF2
6551     End If
6552     SIGH = Analy.Strs(I, 2, 1) - SF2
6553     SIGT = Analy.Strs(I, 3, 1)
6554     SIG1 = Prin1(SIGR, SIGZ, SIGT)
6555     SIG2 = Prin2(SIGR, SIGZ, SIGT)
6556     SIG3 = SIGH
6557     SP2 = Ftres(SIG1, SIG2, SIG3)
6558     公称応力強さの計算
6559     LSIGR = Analy.Lstrs(I, 0, 1)
6560     LSIGZ = Analy.Lstrs(I, 1, 1)
6561     LSIGH = Analy.Lstrs(I, 2, 1)
6562     LSIGT = Analy.Lstrs(I, 3, 1)
6563     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
6564     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
6565     LSIG3 = LSIGH
6566     SN2 = Ftres(LSIG1, LSIG2, LSIG3)
6567     EN2 = SN2 / E2
6568
6569     ピーク応力を含む応力集中係数の計算
6570     If SN2 = 0! Then
6571         KS2 = 0!
6572     Else
6573         KS2 = SPP2 / SN2
6574     End If
6575     KSTAR2 = Smax(KS2, 1!, 1!)
6576     ピーク応力を含まない応力集中係数の計算
6577     If SN2 = 0! Then
6578         K2D = 0!
6579     Else
6580         K2D = SP2 / SN2
6581     End If
6582     K2 = Smax(K2D, 1!, 1!)
6583     ひずみ集中係数の計算
6584     If SN2 = 0! Then
6585         KED2D = 0!
6586     Else
6587         KED2D = SM3 / SN2
6588     End If
6589     KEDL2 = Smax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED2D / KSTAR2), 1!, 1!)
6590     KEDG2 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED2D), 1!, 1!)
6591
6592     KE2 = KSTAR2 * KEDL2 * KEDG2
6593     ETD2 = KE2 * EN2
6594
6595     ETC = Smax(ETD1, ETD2, ETD2)
6596

```

```

6597 Allmtd. Si(I) = Sepr(ETC. Mc)          正解
6598 Allmtd. Si(I) = Sepr(ETC. Mc)/2.0    誤り
6599
6600 Allmtd. Si(I) = Allmtd. Si(I)+Cond. Mecha
6601
6602 初期応力の計算
6603
6604 If Mc. Im = 304 Then GoTo N3040
6605 If Mc. Im = 316 Or Mc. Im = 3160 Then GoTo N3160
6606 WRITE(6,*) ' S I E Q 1 6 7 はSUS304, SUS316, 316FR専用です。'
6607 GoTo N100
6608
6609 N3040:
6610 If Allmtd. Et(I) / 2! <= .002 Then
6611     SIGY = 13.65
6612     HDASH = 10216.3
6613 ElseIf Allmtd. Et(I) / 2! > .002 And Allmtd. Et(I) / 2! <= .003 Then
6614     SIGY = 16.95
6615     HDASH = 5134.4
6616 ElseIf Allmtd. Et(I) / 2! > .003 And Allmtd. Et(I) / 2! <= .004 Then
6617     SIGY = 20.61
6618     HDASH = 3211.4
6619 ElseIf Allmtd. Et(I) / 2! > .004 And Allmtd. Et(I) / 2! <= .005 Then
6620     SIGY = 22.75
6621     HDASH = 2240.8
6622 ElseIf Allmtd. Et(I) / 2! > .005 And Allmtd. Et(I) / 2! <= .006 Then
6623     SIGY = 25.24
6624     HDASH = 1670.9
6625 ElseIf Allmtd. Et(I) / 2! > .006 And Allmtd. Et(I) / 2! <= .007 Then
6626     SIGY = 27.5
6627     HDASH = 1303.5
6628 ElseIf Allmtd. Et(I) / 2! > .007 And Allmtd. Et(I) / 2! <= .008 Then
6629     SIGY = 29.58
6630     HDASH = 1051.1
6631 ElseIf Allmtd. Et(I) / 2! > .008 And Allmtd. Et(I) / 2! <= .009 Then
6632     SIGY = 31.5
6633     HDASH = 869.1
6634 ElseIf Allmtd. Et(I) / 2! > .01 Then
6635     SIGY = 33.3
6636     HDASH = 733.2
6637 End If
6638 GoTo N10000
6639
6640 N3160:
6641 If Allmtd. Et(I) / 2! <= .002 Then
6642     SIGY = 18.64
6643     HDASH = 10216.3
6644 ElseIf Allmtd. Et(I) / 2! > .002 And Allmtd. Et(I) / 2! <= .003 Then
6645     SIGY = 22.27
6646     HDASH = 5134.4
6647 ElseIf Allmtd. Et(I) / 2! > .003 And Allmtd. Et(I) / 2! <= .004 Then
6648     SIGY = 25.36
6649     HDASH = 3211.4
6650 ElseIf Allmtd. Et(I) / 2! > .004 And Allmtd. Et(I) / 2! <= .005 Then
6651     SIGY = 28.06
6652     HDASH = 2240.8
6653 ElseIf Allmtd. Et(I) / 2! > .005 And Allmtd. Et(I) / 2! <= .006 Then
6654     SIGY = 30.59
6655     HDASH = 1670.9
6656 ElseIf Allmtd. Et(I) / 2! > .006 And Allmtd. Et(I) / 2! <= .007 Then
6657     SIGY = 32.79
6658     HDASH = 1303.5
6659 ElseIf Allmtd. Et(I) / 2! > .007 And Allmtd. Et(I) / 2! <= .008 Then

```

```

6660      SIGY = 34.72
6661      HDASH = 1051!
6662      ElseIf Allmtd.Et(I) / 2! > .008 And Allmtd.Et(I) / 2! <= .009 Then
6663          SIGY = 36.56
6664          HDASH = 869.1
6665      ElseIf Allmtd.Et(I) / 2! > .01 Then
6666          SIGY = 38.23
6667          HDASH = 733.2
6668      End If
6669
6670 N10000:
6671      Allmtd.Si(I) = Mc.E / (Mc.E + HDASH) * SIGY
6672
6673      EC1 = 0!
6674      ALPHC = 1!
6675      ALPHR = 1!
6676      DT = Cond.Hldtmh
6677      Allmtd.Qeff(I) = Allmtd.Q1(I) * Allmtd.Qn(I)
6678      Call R1x1(Allmtd.Si(I), 0!, 1!, 1!, Allmtd.Qeff(I), Cond.Hldtmh)
6679      Call R1x1(Allmtd.Si(I), 0.0, 1.0, 1.0, Allmtd.Qeff(I), Cond.Hldtmh, S2xx(0), Dcxx(0), Mc)
6680      DCP = Cond.Cycle * Dcxx(0)
6681
6682      Allmtd.Dc(I) = DCN + DCP
6683
6684      クリープ疲労損傷の計算
6685
6686      Allmtd.D(I) = Allmtd.Df(I) + Allmtd.Dc(I)
6687
6688      許容値の計算
6689
6690      If Allmtd.Dc(I) >= 0! And Allmtd.Dc(I) <= .3 Then
6691          Allmtd.Dcr(I) = 1! - 4! / 3! * Allmtd.Dc(I)
6692      ElseIf Allmtd.Dc(I) <= 1! Then
6693          Allmtd.Dcr(I) = 3! / 7! + 4! / 7! * Allmtd.Dc(I)
6694      ElseIf Allmtd.Dc(I) > 1! Then
6695          Allmtd.Dcr(I) = 1!
6696      End If
6697
6698      Next I
6699
6700 N100:
6701
6702 End Sub
6703
6704
6705      最大値判定
6706
6707 Function Smax (A, B, C)
6708     Dim A, B, C           As Single
6709     Dim D                 As Single
6710     If A < B Then
6711         D = B
6712     Else
6713         D = A
6714     End If
6715     If D < C Then
6716         Smax = C
6717     Else
6718         Smax = D
6719     End If
6720 End Function
6721
6722

```



```

6723 '
6724 '  損傷解析プログラム : メインルーチン
6725 '
6726 '
6727 Sub SonshoD ( )
6728   Dim Id, Res As Integer
6729
6730   Res = 0                                ' for Debug Message
6731
6732   Call Initial                            ' 初期化:見出し変数セット
6733
6734   Call Input_Data_1                       ' DDE 通信:試験条件シート入力
6735   Form_load (Cond.MethodNo)              ' 実行状況BOX表示
6736
6737   Call Allmtd_proc                         ' 損傷値解析処理
6738
6739   Call Excel_Restart                       ' Star-Net システム用プロシージャ関数
6740 End Sub
6741
6742 '
6743 '
6744 '
6745 Function Tcr (S, ECO, ALPHC)
6746   Tcr = 0
6747   If S * ECO <= 0! Then
6748     Exit Function
6749   End If
6750   ZZ = Abs(S)
6751   z = Abs(ECO)
6752   TR = Tcrr(ZZ) / ALPHC
6753   EM = Mc.F * TR ^ Mc.Ramda
6754   R1 = Mc.S1 * TR ^ Mc.Alph1
6755   R2 = Mc.S2 * TR ^ Mc.Alph2
6756   C1 = Mc.G1 * EM ^ Mc.Beta1 / R1
6757   C2 = Mc.G2 * EM ^ Mc.Beta2 / R2
6758   T = z / (C1 * R1 + C2 * R2 + EM)
6759 102 :
6760   EXPR1T = 0
6761   If R1 * T < 60! Then
6762     EXPR1T = Exp(-R1 * T)
6763   End If
6764   EXPR2T = 0
6765   If R2 * T < 60! Then
6766     EXPR2T = Exp(-R2 * T)
6767   End If
6768   DZ = z - EM * T - C1 * (1 - EXPR1T) - C2 * (1 - EXPR2T)
6769   DT = DZ / (C1 * R1 * EXPR1T + C2 * R2 * EXPR2T + EM)
6770   T = T + DT
6771   If DZ / z > .000003 Then
6772     GoTo 102
6773   End If
6774   Tcr = T
6775 End Function
6776
6777 '
6778 '
6779 '
6780 Function Tcrr (X)
6781   If X <= .1 Then
6782     Tcrr = 1E+20
6783   Else
6784     SL = Log(Abs(X)) / Log(10!)
6785     1990年03月06日 修正 浜田 桁溢れへの対策 (PRO-FORTRAN)

```

```

6786   Tcrr=10^(TR0+TR1*SL+TR2*SL^2)
6787   TCRRR = (Mc.Tr0 + Mc.Tr1 * SL + Mc.Tr2 * SL ^ 2)
6788   If TCRRR <= 0! Then
6789     Tcrr = 1!
6790   Else
6791     If TCRRR > 20! Then
6792       TCRRR = 20!
6793     End If
6794     If TCRRR < -20! Then
6795       TCRRR = -20!
6796     End If
6797     Tcrr = 10 ^ (TCRRR)
6798   End If
6799 End If
6800 End Function
6801
6802   T T S - D S
6803
6804
6805   損傷計算結果データ -> Grid.Text7°ロハ°テ代入
6806
6807 Sub Tdsout (Icount)
6808   Dim Icount As Integer
6809   Dim I      As Integer
6810
6811   For I = 0 To Icount - 1
6812     Form1!Grid1.Row = I + 1
6813     Form1!Grid1.Col = 1
6814     Form1!Grid1.Text = Analy.R(I, 0)
6815     Form1!Grid1.Col = 2
6816     Form1!Grid1.Text = Analy.Z(I, 0)
6817     Form1!Grid1.Col = 3
6818     Form1!Grid1.Text = Analy.H(I, 0)
6819     Form1!Grid1.Col = 4
6820     Form1!Grid1.Text = Bscval.Vms(I)
6821     Form1!Grid1.Col = 5
6822     Form1!Grid1.Text = Bscval.Eepceq(I)
6823     Form1!Grid1.Col = 6
6824     Form1!Grid1.Text = Allmtd.Kep(I)
6825     Form1!Grid1.Col = 7
6826     Form1!Grid1.Text = Allmtd.Et(I)
6827     Form1!Grid1.Col = 8
6828     Form1!Grid1.Text = Allmtd.Df(I)
6829     Form1!Grid1.Col = 9
6830     Form1!Grid1.Text = Allmtd.Dc(I)
6831     Form1!Grid1.Col = 10
6832     Form1!Grid1.Text = Allmtd.D(I)
6833   Next I
6834
6835 End Sub
6836
6837
6838   T T S - D S
6839
6840 Sub Ttdsp (Icount)
6841   Dim Icount      As Integer
6842   Dim I, Iflg, Res As Integer
6843   Dim Nuc, Nuh, NU, Er, EDOT, EC1 As Single
6844   Dim ALPHC, ALPHR, NF, SO, Qx, DT As Single
6845   ReDim S2x(1), Dcx(1) As Single
6846
6847   Iflg = 0
6848   Res = 0

```

```

6849
6850 For I = 0 To Icount - 1
6851
6852     歪範囲の計算
6853
6854     Mc.Tc = Analy.Temp(I, 0)
6855     Call Mset(Iflg, Mc) 'Fortran Library(DLL) 呼び出し
6856
6857     Nuc = Calcnu(Mc) 'Fortran Library(DLL) 呼び出し
6858     Mc.Tc = Analy.Temp(I, 1)
6859
6860     Call Mset(Iflg, Mc) 'Fortran Library(DLL) 呼び出し
6861
6862     Nuh = Calcnu(Mc) 'Fortran Library(DLL) 呼び出し
6863     NU = (Nuc + Nuh) / 2!
6864
6865     Mc.Tc = Analy.Tmpmax(I)
6866
6867     Call Mset(Iflg, Mc) 'Fortran Library(DLL) 呼び出し
6868
6869     Er = 1.5 / (1! + NU) * Bscval.Eepceq(I)
6870
6871     Allmtd.Kep(I) = 1.2
6872     Allmtd.Et(I) = Allmtd.Kep(I) * Er
6873
6874     疲労損傷の計算
6875
6876     ファンクションANFの計算機能の限界
6877     If Allmtd.Et(I) < .0005 Then
6878         Allmtd.Et(I) = .0005
6879     End If
6880     EDOT = 2! * Allmtd.Et(I) / ((Cond.Hldtmc + Cond.Hldtmh) * 3600!)
6881     NF = Anf(Allmtd.Et(I), EDOT, Mc) 'Fortran Library(DLL) 呼び出し
6882     Allmtd.Df(I) = Cond.Cycle / NF
6883
6884     クリープ損傷の計算
6885
6886     S0 = Sepr(Allmtd.Et(I), Mc) 'Fortran Library(DLL) 呼び出し
6887     S0 = Sepr(Allmtd.Et(I), Mc) / 2! 'Fortran Library(DLL) 呼び出し
6888
6889     S0 = S0 + Cond.Mecha
6890     EC1 = 0!
6891     ALPHC = 1!
6892     ALPHR = 1!
6893     Qx = 1!
6894     DT = Cond.Hldtmh
6895
6896     Call R1x1(S0, Ec1, Alphc, Alphr, Qx, Dt, S2x(0), Dcx(0), Mc)
6897     Call R1x1(CSng(S0), CSng(EC1), CSng(ALPHC), CSng(ALPHR), CSng(Qx), CSng(DT))
6898     'Fortran Library(DLL) 呼び出し
6899     Allmtd.Dc(I) = Cond.Cycle * Dcx(0)
6900     Allmtd.Dc(I) = Cond.Cycle * Dcxx(0)
6901
6902     クリープ疲労損傷の計算
6903
6904     Allmtd.D(I) = Allmtd.Df(I) + Allmtd.Dc(I)
6905
6906 Next I
6907
6908 End Sub
6909
6910
6911 TYPE 1

```

```

6912 .
6913 .
6914 .   TYPE 1   (中間保持考慮)
6915 .
6916 Sub Typelp (Icount)
6917 .
6918 .   Dim Icount As Integer
6919 .   Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
6920 .   Dim KEDD As Single
6921 .   Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
6922 .   Dim K1, K2, KE1, KE2 As Single
6923 .   Dim K1D, K2D, KED1D, KED2D, KT As Single
6924 .   Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
6925 .   Dim LSIG1, LSIG2, LSIG3 As Single
6926 .   Dim NU, NU1, NU2, NF           As Single
6927 .   Dim NURER1, NUBER2           As Single
6928 .   Dim INF                       As Long
6929 .
6930 .   Iflg = 0
6931 .
6932 .   For I = 0 To Icount - 1
6933 .
6934 .       歪範囲の計算
6935 .
6936 .       T I M E 1
6937 .       Mc.Tc = Analy.Mtemp(I, 0)
6938 .       Call Mset(Iflg, Mc)
6939 .       E1 = Mc.E
6940 .       NU1 = Calcnu(Mc)
6941 .       ALPHA1 = Alpha(Mc)
6942 .       T I M E 2
6943 .       Mc.Tc = Analy.Mtemp(I, 1)
6944 .       Call Mset(Iflg, Mc)
6945 .       E2 = Mc.E
6946 .       NU2 = Calcnu(Mc)
6947 .       ALPHA2 = Alpha(Mc)
6948 .
6949 .       SF1 = (-E1 * ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0))) / (1 - NU1)
6950 .       SF2 = (-E2 * ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2)
6951 .       Allmtd.Ef(I) = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *
        (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
6952 .
6953 .       SANR = Abs(Analy.Strs(I, 0, 1) - Analy.Strs(I, 0, 0))
6954 .       SANZ = Abs(Analy.Strs(I, 1, 1) - Analy.Strs(I, 1, 0))
6955 .       If SANR >= SANZ Then
6956 .           SIGR = (Analy.Strs(I, 0, 0) - SF1) - (Analy.Strs(I, 0, 1) - SF2)
6957 .           SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
6958 .       Else
6959 .           SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
6960 .           SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)
6961 .       End If
6962 .       SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
6963 .       SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
6964 .       SIG1 = Prin1(SIGR, SIGZ, SIGT)
6965 .       SIG2 = Prin2(SIGR, SIGZ, SIGT)
6966 .       SIG3 = SIGH
6967 .       Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
6968 .
6969 .       Allmtd.Sn(I) = Bscval.Ltres(I)
6970 .
6971 .       最高金属温度による ε n、S r h の計算
6972 .       Mc.Tc = Analy.Tmpmax(I)
6973 .       Call Mset(Iflg, Mc)

```

```

6974     EN = Allmtd. Sn(I) / Mc. E
6975
6976     If Allmtd. Sn(I) = 0! Then
6977         KD = 0!
6978     Else
6979         KD = Allmtd. Spx(I) / Allmtd. Sn(I)
6980     End If
6981     Allmtd. K(I) = Smax(Allmtd. Spx(I) / Allmtd. Sn(I), 1!, 1!)
6982     Call Rlx1(1.5 * Mc. Sm, 0!, 1!, 1!, 1!, Cond. Hldtmh)
6983     Call Rlx1(1.5*Mc. Sm, 0. 0. 1. 0. 1. 0. 1. 0, Cond. Hldtmh, S2xx(0), Dcxx(0), Mc)
6984
6985     コールド温度による S r c の計算
6986     Mc. Tc = Analy. Temp(I, 1)
6987     Call Mset(Iflg, Mc)
6988     BAR3SM = S2xx(0) + 1.5 * Mc. Sm
6989     Allmtd. Qn(I) = Analy. Q(I)
6990     If Allmtd. Sn(I) = 0! Then
6991         KEDD = 0!
6992     Else
6993         KEDD = BAR3SM / Allmtd. Sn(I)
6994     End If
6995     Allmtd. Ked(I) = Smax(1! + (Allmtd. Qn(I) - 1!) * (1! - KEDD), 1!, 1!)
6996     Allmtd. Neuber(I) = Fnumber(Allmtd. Sn(I), Allmtd. K(I), BAR3SM)
6997     Allmtd. Ke(I) = Smax(Allmtd. K(I) * Allmtd. Ked(I), Allmtd. Neuber(I), Allmtd. Neuber(I))
6998
6999     KT = 1!
7000
7001     Allmtd. Et(I) = Allmtd. Ke(I) * EN + KT * Allmtd. Ef(I)
7002
7003     疲労損傷の計算
7004
7005     ファンクション AN F の計算機能の限界
7006     If Allmtd. Et(I) < .0005 Then Allmtd. Et(I) = .0005
7007
7008     最高金属温度
7009     Mc. Tc = Analy. Tmpmax(I)
7010     Call Mset(Iflg, Mc)
7011
7012     EDOT = 2! * Allmtd. Et(I) / ((Cond. Hldtmc + Cond. Hldtmh) * 3600!)
7013     NF = Anf(Allmtd. Et(I), EDOT, Mc)
7014
7015     Allmtd. Df(I) = Cond. Cycle / NF
7016
7017     クリープ損傷の計算
7018
7019     D C N の計算
7020     DCN = 0
7021     D C P の計算
7022     初期応力の計算
7023     T I M E 1
7024     EF1 = Abs(ALPHA1 * (Analy. Temp(I, 0) - Analy. Ltemp(I, 0)) / (1 - NU1))
7025     If Analy. Strs(I, 0, 0) >= Analy. Strs(I, 1, 0) Then
7026         SIGR = Analy. Strs(I, 0, 0) - SF1
7027         SIGZ = Analy. Strs(I, 1, 0)
7028     Else
7029         SIGR = Analy. Strs(I, 0, 0)
7030         SIGZ = Analy. Strs(I, 1, 0) - SF1
7031     End If
7032     SIGH = Analy. Strs(I, 2, 0) - SF1
7033     SIGT = Analy. Strs(I, 3, 0)
7034     SIG1 = Prin1(SIGR, SIGZ, SIGT)
7035     SIG2 = Prin2(SIGR, SIGZ, SIGT)
7036     SIG3 = SIGH

```

```

7037     SP1 = Ftres(SIG1, SIG2, SIG3)
7038
7039     If Analy.Lstrs(I, 0, 0) >= Analy.Lstrs(I, 1, 0) Then
7040         LSIGR = Analy.Lstrs(I, 0, 0) - SF1
7041         LSIGZ = Analy.Lstrs(I, 1, 0)
7042     Else
7043         LSIGR = Analy.Lstrs(I, 0, 0)
7044         LSIGZ = Analy.Lstrs(I, 1, 0) - SF1
7045     End If
7046     LSIGH = Analy.Lstrs(I, 2, 0) - SF1
7047     LSIGT = Analy.Lstrs(I, 3, 0)
7048     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
7049     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
7050     LSIG3 = LSIGH
7051     SN1 = Ftres(LSIG1, LSIG2, LSIG3)
7052     EN1 = SN1 / E1
7053     If SN1 = 0! Then
7054         K1D = 0!
7055     Else
7056         K1D = SP1 / SN1
7057     End If
7058     K1 = Smax(K1D, 1!, 1!)
7059     If SN1 = 0! Then
7060         KED1D = 0!
7061     Else
7062         KED1D = BAR3SM / SN1
7063     End If
7064     KED1 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED1D), 1!, 1!)
7065     NUBER1 = Fnumber(SN1, K1, BAR3SM)
7066     KE1 = Smax(K1 * KED1, NUBER1, NUBER1)
7067     ETD1 = KE1 * EN1 + KT * EF1
7068     T I M E 2
7069     EF2 = Abs(ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2))
7070     If Analy.Strs(I, 0, 1) >= Analy.Strs(I, 1, 1) Then
7071         SIGR = Analy.Strs(I, 0, 1) - SF2
7072         SIGZ = Analy.Strs(I, 1, 1)
7073     Else
7074         SIGR = Analy.Strs(I, 0, 1)
7075         SIGZ = Analy.Strs(I, 1, 1) - SF2
7076     End If
7077     SIGH = Analy.Strs(I, 2, 1) - SF2
7078     SIGT = Analy.Strs(I, 3, 1)
7079     SIG1 = Prin1(SIGR, SIGZ, SIGT)
7080     SIG2 = Prin2(SIGR, SIGZ, SIGT)
7081     SIG3 = SIGH
7082     SP2 = Ftres(SIG1, SIG2, SIG3)
7083
7084     If Analy.Lstrs(I, 0, 1) >= Analy.Lstrs(I, 1, 1) Then
7085         LSIGR = Analy.Lstrs(I, 0, 1) - SF2
7086         LSIGZ = Analy.Lstrs(I, 1, 1)
7087     Else
7088         LSIGR = Analy.Lstrs(I, 0, 1)
7089         LSIGZ = Analy.Lstrs(I, 1, 1) - SF2
7090     End If
7091     LSIGH = Analy.Lstrs(I, 2, 1) - SF2
7092     LSIGT = Analy.Lstrs(I, 3, 1)
7093     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
7094     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
7095     LSIG3 = LSIGH
7096     SN2 = Ftres(LSIG1, LSIG2, LSIG3)
7097     EN2 = SN2 / E2
7098     If SN2 = 0! Then
7099         K2D = 0!

```

```

7100 Else
7101     K2D = SP2 / SN2
7102 End If
7103 K2 = Smax(K2D, 1!, 1!)
7104 If SN2 = 0! Then
7105     KED2D = 0!
7106 Else
7107     KED2D = BAR3SM / SN2
7108 End If
7109 KED2 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED2D), 1!, 1!)
7110 NUBER2 = Fnumber(SN2, K2, BAR3SM)
7111 KE2 = Smax(K1 * KED2, NUBER2, NUBER2)
7112 ETD2 = KE2 * EN2 + KT * EF2
7113
7114 ETC = Smax(ETD1, ETD2, ETD2)
7115
7116 Allmtd.Si(I) = Sepr(ETC, Mc)
7117 Allmtd.Si(I) = Sepr(ETC, Mc) / 2!
7118
7119 Allmtd.Si(I) = Allmtd.Si(I) + Cond.Mecha
7120
7121 EC1 = 0!
7122 ALPHC = 1!
7123 ALPHR = 1!
7124 Allmtd.Qeff(I) = 3!
7125 DT = Cond.Hldtmh
7126 Call R1x1(Allmtd.Si(I), 0!, 1!, 1!, Allmtd.Qeff(I), Cond.Hldtmh)
7127 Call R1x1(Allmtd.Si(I), 0.0, 1.0, 1.0, Allmtd.Qeff(I), Cond.Hldtmh, S2xx(0), Dcxx(0), Mc)
7128 DCP = Cond.Cycle * Dcxx(0)
7129
7130 Allmtd.Dc(I) = DCN + DCP
7131
7132 クリープ疲労損傷の計算
7133
7134 Allmtd.D(I) = Allmtd.Df(I) + Allmtd.Dc(I)
7135
7136 Next I
7137
7138 End Sub
7139
7140
7141 TYPE 2
7142
7143
7144 TYPE 2 (中間保持考慮、弾性追従パラメータ最適化)
7145
7146 Sub Type2p (Icount)
7147
7148 Dim Icount As Integer
7149 Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
7150 Dim KEDD As Single
7151 Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
7152 Dim K1, K2, KE1, KE2 As Single
7153 Dim K1D, K2D, KED1D, KED2D, KT As Single
7154 Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
7155 Dim LSIG1, LSIG2, LSIG3 As Single
7156 Dim NU, NU1, NU2, NF As Single
7157 Dim NUBER1, NUBER2 As Single
7158 Dim INF As Long
7159
7160 Iflg = 0
7161
7162 For I = 0 To Icount - 1

```

```

7163
7164   歪範囲の計算
7165
7166   ----- T I M E 1
7167     Mc.Tc = Analy.Mtemp(I, 0)
7168     Call Mset(Iflg, Mc)
7169     E1 = Mc.E
7170     NU1 = Calcnu(Mc)
7171     ALPHA1 = Alpha(Mc)
7172   ----- T I M E 2
7173     Mc.Tc = Analy.Mtemp(I, 1)
7174     Call Mset(Iflg, Mc)
7175     E2 = Mc.E
7176     NU2 = Calcnu(Mc)
7177     ALPHA2 = Alpha(Mc)
7178
7179     SF1 = (-E1 * ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0))) / (1 - NU1)
7180     SF2 = (-E2 * ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2)
7181     Allmtd.Ef(I) = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *
7182     (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
7183     SANR = Abs(Analy.Strs(I, 0, 1) - Analy.Strs(I, 0, 0))
7184     SANZ = Abs(Analy.Strs(I, 1, 1) - Analy.Strs(I, 1, 0))
7185     If SANR >= SANZ Then
7186       SIGR = (Analy.Strs(I, 0, 0) - SF1) - (Analy.Strs(I, 0, 1) - SF2)
7187       SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
7188     Else
7189       SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
7190       SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)
7191     End If
7192     SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
7193     SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
7194     SIG1 = Prin1(SIGR, SIGZ, SIGT)
7195     SIG2 = Prin2(SIGR, SIGZ, SIGT)
7196     SIG3 = SIGH
7197     Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
7198
7199     Allmtd.Sn(I) = Bscval.Ltres(I)
7200
7201   -----最高金属温度による ε n、S r h の計算
7202     Mc.Tc = Analy.Tmpmax(I)
7203     Call Mset(Iflg, Mc)
7204     EN = Allmtd.Sn(I) / Mc.E
7205
7206     If Allmtd.Sn(I) = 0! Then
7207       KD = 0!
7208     Else
7209       KD = Allmtd.Spx(I) / Allmtd.Sn(I)
7210     End If
7211     Allmtd.K(I) = Smax(KD, 1!, 1!)
7212     Call R1x1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmh)
7213     Call R1x1(1.5*Mc.Sm, 0, 0, 1, 0, 1, 0, Cond.Hldtmh, S2xx(0), Dcxx(0), Mc)
7214
7215   -----コールド温度による S r c の計算
7216     Mc.Tc = Analy.Temp(I, 1)
7217     Call Mset(Iflg, Mc)
7218     BAR3SM = S2xx(0) + 1.5 * Mc.Sm
7219     Allmtd.Qn(I) = Analy.Q(I)
7220     Allmtd.Ked(I) = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - BAR3SM / Allmtd.Sn(I)), 1!, 1!)
7221     Allmtd.Neuber(I) = Fneuber(Allmtd.Sn(I), Allmtd.K(I), BAR3SM)
7222     Allmtd.Ke(I) = Smax(Allmtd.K(I) * Allmtd.Ked(I), Allmtd.Neuber(I), Allmtd.Neuber(I))
7223
7224     KT = 1!

```



```

7225 .
7226     Allmtd. Et(I) = Allmtd. Ke(I) * EN + KT * Allmtd. Ef(I)
7227 .
7228     疲労損傷の計算
7229 .
7230     ファンクションANFの計算機能の限界
7231     If Allmtd. Et(I) < .0005 Then Allmtd. Et(I) = .0005
7232 .
7233     -----最高金属温度によるεn、Sr hの計算
7234     Mc. Tc = Analy. Tmpmax(I)
7235     Call Mset(Iflg, Mc)
7236 .
7237     EDOT = 2! * Allmtd. Et(I) / ((Cond. Hldtmc + Cond. Hldtmh) * 3600!)
7238     NF = Anf(Allmtd. Et(I), EDOT, Mc)
7239 .
7240     Allmtd. Df(I) = Cond. Cycle / NF
7241 .
7242     クリープ損傷の計算
7243 .
7244     ----- DCNの計算
7245     DCN = 0
7246     ----- DCPの計算
7247     ----- 初期応力の計算
7248     ----- TIME 1
7249     EF1 = Abs(ALPHA1 * (Analy. Temp(I, 0) - Analy. Ltemp(I, 0)) / (1 - NU1))
7250     If Analy. Strs(I, 0, 0) >= Analy. Strs(I, 1, 0) Then
7251         SIGR = Analy. Strs(I, 0, 0) - SF1
7252         SIGZ = Analy. Strs(I, 1, 0)
7253     Else
7254         SIGR = Analy. Strs(I, 0, 0)
7255         SIGZ = Analy. Strs(I, 1, 0) - SF1
7256     End If
7257     SIGH = Analy. Strs(I, 2, 0) - SF1
7258     SIGT = Analy. Strs(I, 3, 0)
7259     SIG1 = Prin1(SIGR, SIGZ, SIGT)
7260     SIG2 = Prin2(SIGR, SIGZ, SIGT)
7261     SIG3 = SIGH
7262     SP1 = Ftres(SIG1, SIG2, SIG3)
7263 .
7264     If Analy. Lstrs(I, 0, 0) >= Analy. Lstrs(I, 1, 0) Then
7265         LSIGR = Analy. Lstrs(I, 0, 0) - SF1
7266         LSIGZ = Analy. Lstrs(I, 1, 0)
7267     Else
7268         LSIGR = Analy. Lstrs(I, 0, 0)
7269         LSIGZ = Analy. Lstrs(I, 1, 0) - SF1
7270     End If
7271     LSIGH = Analy. Lstrs(I, 2, 0) - SF1
7272     LSIGT = Analy. Lstrs(I, 3, 0)
7273     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
7274     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
7275     LSIG3 = LSIGH
7276     SN1 = Ftres(LSIG1, LSIG2, LSIG3)
7277     EN1 = SN1 / E1
7278     If SN1 = 0! Then
7279         K1D = 0!
7280     Else
7281         K1D = SP1 / SN1
7282     End If
7283     K1 = Smax(K1D, 1!, 1!)
7284     If SN1 = 0! Then
7285         KED1D = 0!
7286     Else
7287         KED1D = BAR3SM / SN1

```

```

7288      End If
7289      KED1 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED1D), 1!, 1!)
7290      NUBER1 = Fnumber(SN1, K1, BAR3SM)
7291      KE1 = Smax(Allmtd.K(I) * KED1, NUBER1, NUBER1)
7292      ETD1 = KE1 * EN1 + KT * EF1
7293      ----- T I M E 2
7294      EF2 = Abs(ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
7295      If Analy.Strs(I, 0, 1) >= Analy.Strs(I, 1, 1) Then
7296          SIGR = Analy.Strs(I, 0, 1) - SF2
7297          SIGZ = Analy.Strs(I, 1, 1)
7298      Else
7299          SIGR = Analy.Strs(I, 0, 1)
7300          SIGZ = Analy.Strs(I, 1, 1) - SF2
7301      End If
7302      SIGH = Analy.Strs(I, 2, 1) - SF2
7303      SIGT = Analy.Strs(I, 3, 1)
7304      SIG1 = Prin1(SIGR, SIGZ, SIGT)
7305      SIG2 = Prin2(SIGR, SIGZ, SIGT)
7306      SIG3 = SIGH
7307      SP2 = Ftres(SIG1, SIG2, SIG3)
7308
7309      If Analy.Lstrs(I, 0, 1) >= Analy.Lstrs(I, 1, 1) Then
7310          LSIGR = Analy.Lstrs(I, 0, 1) - SF2
7311          LSIGZ = Analy.Lstrs(I, 1, 1)
7312      Else
7313          LSIGR = Analy.Lstrs(I, 0, 1)
7314          LSIGZ = Analy.Lstrs(I, 1, 1) - SF2
7315      End If
7316      LSIGH = Analy.Lstrs(I, 2, 1) - SF2
7317      LSIGT = Analy.Lstrs(I, 3, 1)
7318      LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
7319      LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
7320      LSIG3 = LSIGH
7321      SN2 = Ftres(LSIG1, LSIG2, LSIG3)
7322      EN2 = SN2 / E2
7323      If SN2 = 0! Then
7324          K2D = 0!
7325      Else
7326          K2D = SP2 / SN2
7327      End If
7328      K2 = Smax(K2D, 1!, 1!)
7329      If SN2 = 0! Then
7330          KED2D = 0!
7331      Else
7332          KED2D = BAR3SM / SN2
7333      End If
7334      KED2 = Smax(1! + (Allmtd.Qn(I) - 1!) * (1! - KED2D), 1!, 1!)
7335      NUBER2 = Fnumber(SN2, K2, BAR3SM)
7336      KE2 = Smax(Allmtd.K(I) * KED2, NUBER2, NUBER2)
7337      ETD2 = KE2 * EN2 + KT * EF2
7338
7339      ETC = Smax(ETD1, ETD2, ETD2)
7340
7341      Allmtd.Si(I) = Sepr(ETC, Mc)
7342      Allmtd.Si(I) = Sepr(ETC, Mc) / 2!
7343
7344      Res = Debug_disp("[Sepr]/ETD1=" & Str(ETD1) & "/ETD2=" & Str(ETD2)) ' for Debug
7345      If Res = 9 Then Exit Sub
7346
7347      Allmtd.Si(I) = Allmtd.Si(I) + Cond.Mecha
7348
7349      EC1 = 0!
7350      ALPHC = 1!

```

```

7351     ALPHR = 1!
7352     Allmtd.Qeff(I) = Allmtd.Qn(I)
7353     DT = Cond.Hldtmh
7354     Call Rlx1(Allmtd.Si(I), 0!, 1!, 1!, Allmtd.Qeff(I), Cond.Hldtmh)
7355     Call Rlx1(Allmtd.Si(I), 0.0, 1.0, 1.0, Allmtd.Qeff(I), Cond.Hldtmh, S2xx(0), Dcxx(0), Mc)
7356     DCP = Cond.Cycle * Dcxx(0)
7357
7358     Allmtd.Dc(I) = DCN + DCP
7359
7360     クリープ疲労損傷の計算
7361
7362     Allmtd.D(I) = Allmtd.Df(I) + Allmtd.Dc(I)
7363
7364     Next I
7365
7366 End Sub
7367
7368
7369     TYPE 1 及び TYPE 2
7370
7371     損傷計算結果データ -> Grid.Text への代入
7372
7373 Sub Typout (Icount)
7374     Dim Icount As Integer
7375     Dim I       As Integer
7376
7377     For I = 0 To Icount - 1
7378         Form1!Grid1.Row = I + 1
7379         Form1!Grid1.Col = 1
7380         Form1!Grid1.Text = Analy.R(I, 0)
7381         Form1!Grid1.Col = 2
7382         Form1!Grid1.Text = Analy.Z(I, 0)
7383         Form1!Grid1.Col = 3
7384         Form1!Grid1.Text = Analy.H(I, 0)
7385         Form1!Grid1.Col = 4
7386         Form1!Grid1.Text = Allmtd.Spx(I)
7387         Form1!Grid1.Col = 5
7388         Form1!Grid1.Text = Allmtd.Sn(I)
7389         Form1!Grid1.Col = 6
7390         Form1!Grid1.Text = Allmtd.K(I)
7391         Form1!Grid1.Col = 7
7392         Form1!Grid1.Text = Allmtd.Qn(I)
7393         Form1!Grid1.Col = 8
7394         Form1!Grid1.Text = Allmtd.Ked(I)
7395         Form1!Grid1.Col = 9
7396         Form1!Grid1.Text = Allmtd.Neuber(I)
7397         Form1!Grid1.Col = 10
7398         Form1!Grid1.Text = Allmtd.Ke(I)
7399         Form1!Grid1.Col = 11
7400         Form1!Grid1.Text = Allmtd.Ef(I)
7401         Form1!Grid1.Col = 12
7402         Form1!Grid1.Text = Allmtd.Et(I)
7403         Form1!Grid1.Col = 13
7404         Form1!Grid1.Text = Allmtd.Si(I)
7405         Form1!Grid1.Col = 14
7406         Form1!Grid1.Text = Allmtd.Qeff(I)
7407         Form1!Grid1.Col = 15
7408         Form1!Grid1.Text = Allmtd.Df(I)
7409         Form1!Grid1.Col = 16
7410         Form1!Grid1.Text = Allmtd.Dc(I)
7411         Form1!Grid1.Col = 17
7412         Form1!Grid1.Text = Allmtd.D(I)
7413     Next I

```

```

7414
7415 End Sub
7416
7417 EQEF167-WELD-K
7418
7419 EQEF167-WELDの「Qデータ」を1.67/固定から
7420 可変にしたもの。
7421
7422 作成：1995-07-13(木) by N.K, H.H
7423
7424
7425 Sub Eqef16wk (Icount)
7426
7427 Dim Icount As Integer
7428 Dim KD, KS, KS1, KS2, KSTAR, KSTAR1, KSTAR2 As Single
7429 Dim KEDD As Single
7430 Dim KEDL1, KEDG1, KEDL2, KEDG2 As Single
7431 Dim K1, K2, KE1, KE2 As Single
7432 Dim K1D, K2D, KED1D, KED2D As Single
7433 Dim LSIGR, LSIGZ, LSIGH, LSIGT As Single
7434 Dim LSIG1, LSIG2, LSIG3 As Single
7435 Dim NU, NU1, NU2, NF As Single
7436
7437 Iflg = 0
7438
7439 For I = 0 To Icount
7440
7441 歪範囲の計算
7442
7443 TIME 1
7444 Mc.Tc = Analy.Mtemp(I, 0)
7445 Call Mset(Iflg, Mc)
7446 E1 = Mc.E
7447 NU1 = Calcnu(Mc)
7448 ALPHA1 = Alpha(Mc)
7449 TIME 2
7450 Mc.Tc = Analy.Mtemp(I, 1)
7451 Call Mset(Iflg, Mc)
7452 E2 = Mc.E
7453 NU2 = Calcnu(Mc)
7454 ALPHA2 = Alpha(Mc)
7455
7456 熱ピークを含むピーク応力強さの計算
7457 SIGR = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
7458 SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)
7459 SIGH = Analy.Strs(I, 2, 0) - Analy.Strs(I, 2, 1)
7460 SIGT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
7461 SIG1 = Prin1(SIGR, SIGZ, SIGT)
7462 SIG2 = Prin2(SIGR, SIGZ, SIGT)
7463 SIG3 = SIGH
7464 SPP = Ftres(SIG1, SIG2, SIG3)
7465
7466 熱ピークを含まないピーク応力強さの計算
7467 SF1 = (-E1 * ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0))) / (1 - NU1)
7468 SF2 = (-E2 * ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1))) / (1 - NU2)
7469 Allmtd.Ef(I) = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1) - ALPHA2 *
7470 (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
7471
7471 SANR = Abs(Analy.Strs(I, 0, 1) - Analy.Strs(I, 0, 0))
7472 SANZ = Abs(Analy.Strs(I, 1, 1) - Analy.Strs(I, 1, 0))
7473 If SANR >= SANZ Then
7474 SIGR = (Analy.Strs(I, 0, 0) - SF1) - (Analy.Strs(I, 0, 1) - SF2)
7475 SIGZ = Analy.Strs(I, 1, 0) - Analy.Strs(I, 1, 1)

```

```

7476 Else
7477   SIGH = Analy.Strs(I, 0, 0) - Analy.Strs(I, 0, 1)
7478   SIGZ = (Analy.Strs(I, 1, 0) - SF1) - (Analy.Strs(I, 1, 1) - SF2)
7479 End If
7480 SIGH = (Analy.Strs(I, 2, 0) - SF1) - (Analy.Strs(I, 2, 1) - SF2)
7481 SIGHT = Analy.Strs(I, 3, 0) - Analy.Strs(I, 3, 1)
7482 SIG1 = Prin1(SIGR, SIGZ, SIGHT)
7483 SIG2 = Prin2(SIGR, SIGZ, SIGHT)
7484 SIG3 = SIGH
7485 Allmtd.Spx(I) = Ftres(SIG1, SIG2, SIG3)
7486
7487 Allmtd.Sn(I) = Bscval.Ltres(I)
7488
7489 最高金属温度による  $\epsilon_n$ 、S r h の計算
7490 Mc.Tc = Analy.Tmpmax(I)
7491 Call Mset(Iflg, Mc)
7492 EN = Allmtd.Sn(I) / Mc.E
7493
7494 If Allmtd.Sn(I) = 0! Then
7495   KD = 0!
7496 Else
7497   KD = Allmtd.Spx(I) / Allmtd.Sn(I)
7498 End If
7499 Allmtd.K(I) = Smax(KD, 1!, 1!)
7500 Call Rlx1(1.5 * Mc.Sm, 0!, 1!, 1!, 1!, Cond.Hldtmh)
7501 Call Rlx1(1.5 * Mc.Sm, 0.0, 1.0, 1.0, 1.0, Cond.Hldtmh, SRH, DCX)
7502
7503 コールド温度による S r c の計算
7504 Mc.Tc = Analy.Temp(I, 1)
7505 Call Mset(Iflg, Mc)
7506 NU = .3
7507 Allmtd.Qnu(I) = 2! * (1! - NU)
7508 BAR3SM = S2xx(0) + 1.5 * Mc.Sm
7509 Allmtd.Qn(I) = 1!
7510 Allmtd.Qw(I) = 1.5
7511 Allmtd.Ql(I) = 1.67
7512 Allmtd.Ql(I) = Smax(Allmtd.K(I)^(3!/4!), 1.67, 1.67) 追加:1995/0713(木)N.K
7513 GAMMAY = .8
7514 If Allmtd.Sn(I) = 0! Then
7515   KEDD = 0!
7516 Else
7517   KEDD = GAMMAY * BAR3SM / Allmtd.Sn(I)
7518 End If
7519 Allmtd.Ked1(I) = Smax(1! + (Allmtd.Ql(I) - 1!) * (1! - KEDD / Allmtd.K(I)), 1!, 1!)
7520 Allmtd.Kedg(I) = Smax(1! + (Allmtd.Qn(I) * Allmtd.Qw(I) - 1!) * (1! - KEDD), 1!, 1!)
7521
7522 ピーク応力を含む応力集中係数の計算
7523 If Allmtd.Sn(I) = 0! Then
7524   KS = 0!
7525 Else
7526   KS = SPP / Allmtd.Sn(I)
7527 End If
7528 KSTAR = Smax(KS, 1!, 1!)
7529
7530 Allmtd.Ke(I) = Smax(KSTAR * Allmtd.Ked1(I) * Allmtd.Kedg(I), 1.3, 1.3)
7531
7532 Allmtd.Et(I) = Allmtd.Ke(I) * EN
7533
7534 疲労損傷の計算
7535
7536 ファンクション ANF の計算機能の限界
7537 If Allmtd.Et(I) < .0005 Then Allmtd.Et(I) = .0005
7538

```

```

7539 最高金属温度
7540 Mc.Tc = Analy.Tmpmax(I)
7541 Call Mset(Iflg, Mc)
7542
7543 EDOT = 2! * Allmtd.Et(I) / ((Cond.Hldtmc + Cond.Hldtmh) * 3600!)
7544 NF = Anf(Allmtd.Et(I), EDOT, Mc) Fortran Library(DLL) 呼び出し
7545
7546 Allmtd.Df(I) = Cond.Cycle / NF
7547
7548 クリーブ損傷の計算
7549
7550 DCNの計算
7551 DCN = 0
7552 DCPの計算
7553 初期応力の計算
7554 TIME1
7555
7556 熱ピークを含まないピーク応力強さの計算
7557 SIGR = Analy.Strs(I, 0, 0)
7558 SIGZ = Analy.Strs(I, 1, 0)
7559 SIGH = Analy.Strs(I, 2, 0)
7560 SIGT = Analy.Strs(I, 3, 0)
7561 SIG1 = Prin1(SIGR, SIGZ, SIGT)
7562 SIG2 = Prin2(SIGR, SIGZ, SIGT)
7563 SIG3 = SIGH
7564 SPP1 = Ftres(SIG1, SIG2, SIG3)
7565
7566 熱ピークを含まないピーク応力強さの計算
7567 EF1 = Abs(ALPHA1 * (Analy.Temp(I, 0) - Analy.Ltemp(I, 0)) / (1 - NU1))
7568 If Analy.Strs(I, 0, 0) >= Analy.Strs(I, 1, 0) Then
7569     SIGR = Analy.Strs(I, 0, 0) - SF1
7570     SIGZ = Analy.Strs(I, 1, 0)
7571 Else
7572     SIGR = Analy.Strs(I, 0, 0)
7573     SIGZ = Analy.Strs(I, 1, 0) - SF1
7574 End If
7575 SIGH = Analy.Strs(I, 2, 0) - SF1
7576 SIGT = Analy.Strs(I, 3, 0)
7577 SIG1 = Prin1(SIGR, SIGZ, SIGT)
7578 SIG2 = Prin2(SIGR, SIGZ, SIGT)
7579 SIG3 = SIGH
7580 SP1 = Ftres(SIG1, SIG2, SIG3)
7581
7582 If Analy.Lstrs(I, 0, 0) >= Analy.Lstrs(I, 1, 0) Then
7583     LSIGR = Analy.Lstrs(I, 0, 0) - SF1
7584     LSIGZ = Analy.Lstrs(I, 1, 0)
7585 Else
7586     LSIGR = Analy.Lstrs(I, 0, 0)
7587     LSIGZ = Analy.Lstrs(I, 1, 0) - SF1
7588 End If
7589 LSIGH = Analy.Lstrs(I, 2, 0) - SF1
7590 LSIGT = Analy.Lstrs(I, 3, 0)
7591 LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
7592 LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
7593 LSIG3 = LSIGH
7594 SNI = Ftres(LSIG1, LSIG2, LSIG3)
7595 ENI = SNI / EI
7596 If SNI = 0! Then
7597     KID = 0!
7598 Else
7599     KID = SP1 / SNI
7600 End If
7601 KI = Smax(KID, 1!, 1!)

```

```

7602     If SN1 = 0! Then
7603         KED1D = 0!
7604     Else
7605         KED1D = CAMWAY * BAR3SM / SN1
7606     End If
7607     KEDL1 = Smax(1.0+(Allmtd.Q1(I)-1.0)*(1.0-KED1D/K1D), 1.0)
7608     KEDL1 = Smax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED1D / K1), 1!, 1!)
7609     KEDG1 = Smax(1! + (Allmtd.Qn(I) * Allmtd.Qw(I) - 1!) * (1! - KED1D), 1!, 1!)
7610
7611     ピーク応力を含む応力集中係数の計算
7612     If SN1 = 0! Then
7613         KS1 = 0!
7614     Else
7615         KS1 = SPP1 / SN1
7616     End If
7617     KSTAR1 = Smax(KS1, 1!, 1!)
7618
7619     KE1 = Smax(KSTAR1 * KEDL1 * KEDG1, 1.3, 1.3)
7620     ETD1 = KE1 * EN1
7621     T I M E 2
7622
7623     熱ピークを含むピーク応力強さの計算
7624     SIGR = Analy.Strs(I, 0, 1)
7625     SIGZ = Analy.Strs(I, 1, 1)
7626     SIGH = Analy.Strs(I, 2, 1)
7627     SIGT = Analy.Strs(I, 3, 1)
7628     SIG1 = Prin1(SIGR, SIGZ, SIGT)
7629     SIG2 = Prin2(SIGR, SIGZ, SIGT)
7630     SIG3 = SIGH
7631     SPP2 = Ftres(SIG1, SIG2, SIG3)
7632
7633     熱ピークを含まないピーク応力強さの計算
7634     EF2 = Abs(ALPHA2 * (Analy.Temp(I, 1) - Analy.Ltemp(I, 1)) / (1 - NU2))
7635     If Analy.Strs(I, 0, 1) >= Analy.Strs(I, 1, 1) Then
7636         SIGR = Analy.Strs(I, 0, 1) - SF2
7637         SIGZ = Analy.Strs(I, 1, 1)
7638     Else
7639         SIGR = Analy.Strs(I, 0, 1)
7640         SIGZ = Analy.Strs(I, 1, 1) - SF2
7641     End If
7642     SIGH = Analy.Strs(I, 2, 1) - SF2
7643     SIGT = Analy.Strs(I, 3, 1)
7644     SIG1 = Prin1(SIGR, SIGZ, SIGT)
7645     SIG2 = Prin2(SIGR, SIGZ, SIGT)
7646     SIG3 = SIGH
7647     SP2 = Ftres(SIG1, SIG2, SIG3)
7648
7649     If Analy.Lstrs(I, 0, 1) >= Analy.Lstrs(I, 1, 1) Then
7650         LSIGR = Analy.Lstrs(I, 0, 1) - SF2
7651         LSIGZ = Analy.Lstrs(I, 1, 1)
7652     Else
7653         LSIGR = Analy.Lstrs(I, 0, 1)
7654         LSIGZ = Analy.Lstrs(I, 1, 1) - SF2
7655     End If
7656     LSIGH = Analy.Lstrs(I, 2, 1) - SF2
7657     LSIGT = Analy.Lstrs(I, 3, 1)
7658     LSIG1 = Prin1(LSIGR, LSIGZ, LSIGT)
7659     LSIG2 = Prin2(LSIGR, LSIGZ, LSIGT)
7660     LSIG3 = LSIGH
7661     SN2 = Ftres(LSIG1, LSIG2, LSIG3)
7662     EN2 = SN2 / E2
7663     If SN2 = 0! Then
7664         K2D = 0!

```

```

7665 Else
7666     K2D = SP2 / SN2
7667 End If
7668 K2 = Smax(K2D, 1!, 1!)
7669 If SN2 = 0! Then
7670     KED2D = 0!
7671 Else
7672     KED2D = GAMWAY * BAR3SM / SN2
7673 End If
7674 KEDL2 = Smax(1.0+(Allmtd.Q1(I)-1.0)*(1.0-KED2D/K2D), 1.0, 1.0)
7675 KEDL2 = Smax(1! + (Allmtd.Q1(I) - 1!) * (1! - KED2D / K2), 1!, 1!)
7676 KEDG2 = Smax(1! + (Allmtd.Qn(I) * Allmtd.Qw(I) - 1!) * (1! - KED2D), 1!, 1!)
7677
7678 ピーク応力を含む応力集中係数の計算
7679 If SN2 = 0! Then
7680     KS2 = 0!
7681 Else
7682     KS2 = SPP2 / SN2
7683 End If
7684 KSTAR2 = Smax(KS2, 1!, 1!)
7685
7686 KE2 = Smax(KSTAR2 * KEDL2 * KEDG2, 1.3, 1.3)
7687 ETD2 = KE2 * EN2
7688
7689 ETC = Smax(ETD1, ETD2, ETD2)
7690
7691 Allmtd.Si(I) = Sepr(ETC, Mc) ' Fortran Library(DLL) 呼び出し
7692 Allmtd.Si(I) = Sepr(ETC, Mc) / 2! ' Fortran Library(DLL) 呼び出し
7693
7694 Res = Debug_disp("[Sepr]/ETD1=" & Str(ETD1) & "/ETD2=" & Str(ETD2)) ' for Debug
7695 If Res = 9 Then Exit Sub
7696
7697 Allmtd.Si(I) = Allmtd.Si(I) + Cond.Mecha
7698
7699 EC1 = 0!
7700 ALPHC = 1!
7701 ALPHR = 1!
7702 DT = Cond.Hldtmh
7703 Allmtd.Qeff(I) = (Allmtd.K(I) ^ (3! / 4!)) * Allmtd.Qnu(I) * Allmtd.Qn(I) * Allmtd.Qw(I)
7704 Call Rlx1(Allmtd.Si(I), 0!, 1!, 1!, Allmtd.Qeff(I), Cond.Hldtmh)
7705 Call Rlx1(Allmtd.Si(I), EC1, ALPHC, ALPHR, Allmtd.Qeff(I), DT, S2X, DCX)
7706 DCP = 2! * Cond.Cycle * Dcxx(0)
7707
7708 Allmtd.Dc(I) = DCN + DCP
7709
7710 クリープ疲労損傷の計算
7711
7712 Allmtd.D(I) = Allmtd.Df(I) + Allmtd.Dc(I)
7713
7714 許容値の計算
7715
7716 If Allmtd.Dc(I) >= 0! And Allmtd.Dc(I) <= .3 Then
7717     Allmtd.Dcr(I) = 1! - 4! / 3! * Allmtd.Dc(I)
7718 Elseif Allmtd.Dc(I) <= 1! Then
7719     Allmtd.Dcr(I) = 3! / 7! + 4! / 7! * Allmtd.Dc(I)
7720 Elseif Allmtd.Dc(I) > 1! Then
7721     Allmtd.Dcr(I) = 1!
7722 End If
7723
7724 Next I
7725
7726 End Sub

```



付録B ソース・コードNo. 2/FORTRAN言語  
(材料特性値算出用ライブラリ)

```

1 C***** MMFUNC. FORT *****
2 C
3 C   材料特性値算出用ライブラリ
4 C
5 C   適用鋼種      SUS304  SUS316  SUS321
6 C                SCMV4(NT)  STBA24(NT)
7 C                改良9CR1MO鋼  316FR
8 C
9 C   使用単位      応力      KG/MM**2
10 C              ひずみ     MM/MM
11 C              時間      HR
12 C              ひずみ速度   クリープ速度   MM/MM/HR
13 C              疲労強度     疲労強度     MM/MM/SEC
14 C
15 C   **** ライブラリの内容 ****
16 C
17 C   SUBROUTINE      Mset(IFLG, MCDIW)
18 C                 材料物性値算出用の定数のセットを行う。
19 C                 事前に鋼種(IM)及び温度(TC)を規定しておく
20 C                 必要がある。
21 C                 本サブルーチンのコールにより、対象鋼種の対象温度での
22 C                 物性値テーブル(COMMON /MC/MC(10,4)) を定める。
23 C
24 C                 IFLG      IFLG=1 弾塑性における最小特性値
25 C                 1 以外  平均値
26 C
27 C
28 C   FUNCTION        Suval(IDUMMY)
29 C                 高温構造設計方針材料強度基準等に規定される設計引張強さ
30 C
31 C                 IDUMMY  将来の使用を考慮したダミー引数。現状未使用
32 C
33 C
34 C   FUNCTION        Smval(IDUMMY)
35 C                 高温構造設計方針材料強度基準等に規定される設計降伏点
36 C                 (KG/MM**2)
37 C                 IDUMMY  将来の使用を考慮したダミー引数。現状未使用
38 C
39 C
40 C   FUNCTION        Eepm(S)
41 C                 応力S(KG/MM**2)に対する弾塑性歪(MM/MM)
42 C                 (単調負荷)
43 C
44 C
45 C   FUNCTION        Sepm(X)
46 C                 弾塑性歪X(MM/MM)に対する応力(KG/MM**2)
47 C                 (単調負荷)
48 C
49 C
50 C   FUNCTION        Eepr(S)
51 C                 応力範囲S(KG/MM**2) に対する
52 C                 弾塑性ひずみ範囲(MM/MM) (繰返し負荷)
53 C
54 C
55 C   FUNCTION        Sepr(X)
56 C                 弾塑性ひずみ範囲X(MM/MM) に対する
57 C                 応力範囲(KG/MM**2) (繰返し負荷)
58 C
59 C
60 C   FUNCTION        Tcrr(S)
61 C                 応力S(MM/KG**2) に対する平均クリープ破断時間(HR)
62 C
63 C                 Tcrr = 10**(.....)の

```

64 C (.....)部分の値が20.0以上なら  
 65 C 20.0にする。又、-20.0以下なら-20.0にする。  
 66 C 1990年03月06日 修正 アイ・エス・エル 浜田  
 67 C  
 68 C  
 69 C FUNCTION Scrr(T)  
 70 C 平均クリープ破断時間T(HR) に対する応力(KG/MM\*\*2)  
 71 C  
 72 C FUNCTION Ecr(S, T, ALPHC)  
 73 C 応力S(KG/MM\*\*2) 時間T(HR) 及び  
 74 C クリープパラメータALPHCにおける  
 75 C クリープひずみ(MM/MM)  
 76 C  
 77 C  
 78 C FUNCTION Ecrr(S, T, ALPHC)  
 79 C 応力S(KG/MM\*\*2) 時間T(HR) 及び  
 80 C クリープパラメータALPHCにおける  
 81 C クリープひずみ速度(MM/MM/HR)  
 82 C  
 83 C  
 84 C FUNCTION Tcr(S, ECO, ALPHC)  
 85 C 応力S(KG/MM\*\*2) クリープ歪ECO(MM/MM) 及び  
 86 C クリープパラメータALPHC に対応する時間(HR)  
 87 C  
 88 C  
 89 C FUNCTION Anf(ET, EDOT)  
 90 C 全ひずみ範囲ET(MM/MM) 及びひずみ速度  
 91 C EDOT(MM/MM/SEC)に対する平均疲労寿命  
 92 C  
 93 C Anf = 10\*\*((.....)\*\*(-2))の  
 94 C ((.....)\*\*(-2))部分の値が35.0以上なら  
 95 C 35.0にする。又、-35.0以下なら-35.0にする。  
 96 C 1990年03月06日 修正 アイ・エス・エル 浜田  
 97 C  
 98 C  
 99 C FUNCTION Sids(SG)  
 100 C 応力水準SG(KG/MM\*\*2)におけるDstar  
 101 C 算出用初期応力(KG/MM\*\*2)  
 102 C  
 103 C  
 104 C FUNCTION Dstar(SN, TSTAR, DC1)  
 105 C (1次+2次)応力範囲SN(KG/MM\*\*2)、  
 106 C 高温使用時間TSTAR(HR)及び応力水準SGによる  
 107 C クリープ損傷DC1に対する(1次+2次)応力の緩和に  
 108 C 伴うクリープ損傷係数Dstar値 ただし、温度による  
 109 C 補正が無いので注意を要する。  
 110 C  
 111 C  
 112 C FUNCTION Ddstar(ET, TSTAR, DC1)  
 113 C 等価歪範囲ET(MM/MM)、高温使用時間TSTAR(HR)及び  
 114 C 応力水準SGによるクリープ損傷DC1 に対する  
 115 C ピーク応力の緩和に伴うクリープ損傷係数DDOUBLE-STAR値  
 116 C ただし、温度による補正が無いので注意を要する。  
 117 C  
 118 C  
 119 C SUBROUTINE R1x1(S0, EC1, ALPHC, ALPHR, Q, DT, S2, DC)  
 120 C 初期応力S0(KG/MM\*\*2)、初期クリープ歪  
 121 C EC1(MM/MM)、クリープパラメータALPHC、  
 122 C クリープ破断係数ALPHR、  
 123 C 弾性追従パラメータQ 及び時間DT(HR)での緩和応力  
 124 C S2(KG/MM\*\*2)及びクリープ損傷を計算する  
 125 C サブルーチン  
 126 C

```

127 C
128 C SUBROUTINE R1x2(S0, EC1, ALPHC, Q, DT, S2, DC)
129 C 初期応力S0(KG/MM**2)、
130 C 初期クリープ歪EC1(MM/MM)、
131 C クリープパラメータ ALPHC、弾性追従パラメータQ 及び
132 C 緩和応力S2(KG/MM**2)に対応する時間DT(HR)及び
133 C クリープ損傷DCを計算するサブルーチン
134 C
135 C
136 C SUBROUTINE Mclist(IFLG)
137 C 材料特性を定める定数の内容を表示するサブルーチン
138 C IFLG=0 全定数表示
139 C IFLG=1 MC(*,1) の内容表示
140 C IFLG=2 MC(*,2) の内容表示
141 C IFLG=3 MC(*,3) の内容表示
142 C IFLG=4 MC(*,4) の内容表示
143 C
144 C
145 C SUBROUTINE Crlist(S)
146 C 応力S(KG/MM**2) に対するクリープ定数
147 C C1, R1, C2, R2, EMを表示するサブルーチン
148 C
149 C
150 C **** 名前付き共通領域 COMMON /MC/MC(10,4)の内容 ****
151 C
152 C 本ライブラリでは、名前付き共通領域
153 C COMMON / MC / MC(10,4)
154 C に各鋼種及び温度での材料特性算出用定数を記録することにより、
155 C 各サブルーチンの引数の簡素化及びライブラリ構成の簡明化が
156 C 図られている。
157 C
158 C ** MC(*,1) ** 鋼種、温度及び弾塑性特性
159 C EEP=EEL+EPL
160 C EEL=S/E
161 C EPL=((S-SP)/K)**M
162 C MC(1,1) IM 鋼種を規定するパラメータ値
163 C DEFAULT IM=304
164 C IM=304 SUS304
165 C IM=316 SUS316
166 C IM=321 SUS321
167 C IM=2250 SCMV4(NT)
168 C IM=2251 STBA24(NT)
169 C IM=91 改良9CR1MO鋼 (ただし現状暫定値)
170 C IM=3160 316FR
171 C MC(2,1) TC 温度 (°C)
172 C MC(3,1) E 縦弾性係数 (KG/MM**2)
173 C MC(4,1) SY 降伏点 (KG/MM**2)
174 C MC(5,1) SPM 単調負荷時比例限 (KG/MM**2)
175 C MC(6,1) AMM 単調負荷時塑性硬化指数
176 C MC(7,1) AKM 単調負荷時塑性硬化係数 (KG/MM**2)
177 C MC(8,1) SPC 繰返し負荷時比例限 (KG/MM**2)
178 C MC(9,1) AMC 繰返し負荷時塑性硬化指数
179 C MC(10,1) AKC 繰返し負荷時塑性硬化係数 (KG/MM**2)
180 C
181 C
182 C ** MC(*,2) ** クリープ破断特性
183 C TR=10**(TR0+TR1*ALOG10(S)+TR2*(ALOG10(S))**2)
184 C MC(1,2) TR0
185 C MC(2,2) TR1
186 C MC(3,2) TR2
187 C MC(4,2) SU 設計引張り強さSU (KG/MM**2)
188 C MC(5,2) SM 設計応力強さ SM (KG/MM**2)
189 C

```

```

190 C
191 C ** MC(*, 3) **      クリーブ変形特性
192 C      EC=C1*(1-EXP(-R1*T))+C2*(1-EXP(-R2*T))+EM*T
193 C      R1=S1*TR**ALPH1
194 C      R2=S2*TR**ALPH2
195 C      C1=G1*EM**BETA1/R1
196 C      C2=G2*EM**BETA2/R2
197 C      EM=F *TR**RAMDA
198 C      F =FO*EXP(-Q/(R*TK))   R=8.31
199 C      MC(1, 3) S1
200 C      MC(2, 3) ALPH1
201 C      MC(3, 3) S2
202 C      MC(4, 3) ALPH2
203 C      MC(5, 3) G1
204 C      MC(6, 3) BETA1
205 C      MC(7, 3) G2
206 C      MC(8, 3) BETA2
207 C      MC(9, 3) F
208 C      MC(10, 3) RAMDA
209 C
210 C
211 C ** MC(*, 4) **      疲労寿命
212 C      1/SQRT(ALOG10(NF))=(AOT+AOR*R)+(A1T+A1R*R)*ALOG10(ET)
213 C      +(A2T+A2R*R)*ALOG10(ET)**2
214 C      +(A4T+A4R*R)*ALOG10(ET)**4
215 C      MC(1, 4) AOT
216 C      MC(2, 4) AOR
217 C      MC(3, 4) A1T
218 C      MC(4, 4) A1R
219 C      MC(5, 4) A2T
220 C      MC(6, 4) A2R
221 C      MC(7, 4) A4T
222 C      MC(8, 4) A4R
223 C
224 C
225 C
226 C
227 C      INITIAL CODING 2/17/84 BY TN
228 C      FINAL REVISION 6/16/88 BY TN
229 C      (PC98系に装荷するに際してコメント文全面修正)
230 C      2-1/4CR-MO用MODULUS OF ELASTICITY 及び
231 C      繰り返し応力ひずみ関係を修正
232 C      10/27/88 BY TN
233 C      Mod. 9Cr-1Mod データ追加 5/13/93 BY NK
234 C
235 C
236 C *****
237 C
238 C
239 C
240 C *****
241 C
242 C      SUBROUTINE Mset [PASCAL] (IFLG, Mc)
243 C
244 C *****
245 C
246 C      INCLUDE 'STRUCT.FOR'
247 C      INTEGER*2 IFLG, IM
248 C      RECORD /smc/ Mc [FAR, REFERENCE]
249 C      COMMON /MC/IM, TC, E, SY, SP, AM, AK, SPC, AMC, AKC
250 C      #, TR0, TR1, TR2, SU, SM, DUMM2(5)
251 C      #, S1, ALPH1, S2, ALPH2, G1, BETA1, G2, BETA2, F, RAMDA
252 C      #, AOT, AOR, A1T, A1R, A2T, A2R, A4T, A4R, DUMM4(2)

```

```

253 C
254 C
255     TK=Mc. Tc+273. 15
256     IM=Mc. Im
257 C
258 C*****
259 C ** SUS304 **
260 C*****
261 C
262     IF(IM. NE. 304) GO TO 2000
263 C     IF(Mc. Im. NE. 304) GO TO 2000
264     Mc. E=2. 10236E4-9. 71895*Mc. Tc
265 C     WRITE(6,*) ' E = ', Mc. E
266     Mc. Sy=25. 5655-5. 58937E-2*Mc. Tc+1. 04384E-4*Mc. Tc**2
267     #-7. 42535E-8*Mc. Tc**3
268     ALPHY=1. 3
269     IF(Mc. Tc. LT. 500.) ALPHY=1. 3-. 01*(500-Mc. Tc)/25
270     IF(IFLG. NE. 1) ALPHY=1
271     Mc. Sy=Mc. Sy/ALPHY
272 C
273     Mc. Ak=44. 3068-1. 78933E-2*Mc. Tc
274     Mc. Am=. 279395+7. 749E-5*Mc. Tc
275     Mc. Sp=Mc. Sy-Mc. Ak*. 002**Mc. Am
276 C
277     Mc. Akc=10**(. 9772725+. 644735E-2*Mc. Tc-. 4678257E-5*Mc. Tc**2
278     #- . 372179E-8*Mc. Tc**3)
279     Mc. Amc=3. 689759-. 01847746*Mc. Tc+. 3544464E-4*Mc. Tc**2
280     #- . 2297527E-7*Mc. Tc**3
281     Mc. Spc=Mc. Sy*ALPHY-Mc. Ak*. 002**Mc. Am
282 C
283     Mc. Tr0=-17. 54301+26248. 54/TK
284     Mc. Tr1=-6104. 579/TK
285     Mc. Tr2=-425. 0012/TK
286     Mc. Su=Suval(0, Mc)
287     Mc. Sm=Smval(0, Mc)
288 C
289     Mc. S1=103. 37
290     Mc. Alph1=-. 72607
291     Mc. S2=17. 255
292     Mc. Alph2=-. 86775
293     Mc. G1=1. 2692
294     Mc. Beta1=. 74491
295     Mc. G2=. 48449
296     Mc. Beta2=. 81155
297     Mc. F=62. 416*EXP(-40812. /(8. 31*TK))
298     Mc. Ramda=-1. 1335
299 C
300 1900 Mc. A0t=1. 621827
301     Mc. A0r=-. 456785E-7*Mc. Tc**2
302     Mc. A1t=1. 131346+. 8665061E-8*Mc. Tc**2
303     Mc. A1r=0
304     Mc. A2t=. 3439663
305     Mc. A2r=0
306     Mc. A4t=-. 01374387
307     Mc. A4r=. 4910723E-4
308 C
309     RETURN
310 C
311 C*****
312 C ** SUS316 **
313 C*****
314 C
315 2000 IF(Mc. Im. NE. 316) GO TO 3000

```

```

316 Mc. E=2. 10236E4-9. 71895*Mc. Tc
317 Mc. Sy=26. 8073-5. 04547E-2*Mc. Tc+8. 03961E-5*Mc. Tc**2
318 #-5. 11282E-8*Mc. Tc**3
319 ALPHY=1. 3
320 IF(Mc. Tc. LT. 450.) ALPHY=1. 3-. 01*(450-Mc. Tc)/25
321 IF(IFLG. NE. 1) ALPHY=1
322 Mc. Sy=Mc. Sy/ALPHY
323 C
324 Mc. Ak=40. 0909-9. 6999E-3*Mc. Tc
325 Mc. Am=. 326245+6. 13276E-5*Mc. Tc
326 Mc. Sp=Mc. Sy-Mc. Ak*. 002**Mc. Am
327 C
328 Mc. Akc=10**(4. 139556-. 4434273E-2*Mc. Tc+. 1354228E-5*Mc. Tc**2
329 #+. 1593061E-8*Mc. Tc**3)
330 Mc. Amc=2. 171727-. 7045263E-2*Mc. Tc+. 7832692E-5*Mc. Tc**2
331 #- . 20836E-8*Mc. Tc**3
332 Mc. Spc=Mc. Sy*ALPHY-Mc. Ak*. 002**Mc. Am
333 C
334 Mc. Tr0=-17. 37762+22983. 2/TK
335 Mc. Tr1=-798. 521/TK
336 Mc. Tr2=-2163. 93/TK
337 Mc. Su=Suval(0, Mc)
338 Mc. Sm=Smval(0, Mc)
339 C
340 Mc. S1=103. 37
341 Mc. Alph1=-. 72607
342 Mc. S2=17. 255
343 Mc. Alph2=-. 86775
344 Mc. G1=1. 2692
345 Mc. Beta1=. 74491
346 Mc. G2=. 48449
347 Mc. Beta2=. 81155
348 Mc. F=59119. *EXP(-93475. /(8. 31*TK))
349 Mc. Ramda=-1. 1126
350 C
351 GO TO 1900
352 C
353 C*****
354 C ** SUS321 **
355 C*****
356 C
357 3000 IF(Mc. Im. NE. 321) GO TO 4000
358 Mc. E=2. 10236E4-9. 71895*Mc. Tc
359 Mc. Sy=24. 4905-3. 80298E-2*Mc. Tc+6. 15139E-5*Mc. Tc**2
360 #-4. 06897E-8*Mc. Tc**3
361 ALPHY=1. 3
362 IF(Mc. Tc. LT. 500.) ALPHY=1. 3+. 02*(500-Mc. Tc)/25
363 IF(Mc. Tc. LT. 475.) ALPHY=1. 3+. 01*(475-Mc. Tc)/25
364 IF(Mc. Tc. LT. 450.) ALPHY=1. 3+. 02*(450-Mc. Tc)/25
365 IF(IFLG. NE. 1) ALPHY=1
366 Mc. Sy=Mc. Sy/ALPHY
367 C
368 Mc. Ak=59. 517-2. 6872E-2*Mc. Tc
369 Mc. Am=. 3
370 Mc. Sp=Mc. Sy-Mc. Ak*. 002**Mc. Am
371 C
372 Mc. Akc=10**(39. 77021-. 2107718*Mc. Tc+. 3995122E-3*Mc. Tc**2
373 #- . 2548977E-6*Mc. Tc**3)
374 Mc. Amc=20. 53651-. 1135368*Mc. Tc+. 2139214E-3*Mc. Tc**2
375 #- . 1355191E-6*Mc. Tc**3
376 Mc. Spc=Mc. Sy*ALPHY-Mc. Ak*. 002**Mc. Am
377 C
378 Mc. Tr0=-16. 46258+20089. /TK

```

```

379      Mc. Tr1=1953.03/TK
380      Mc. Tr2=-2954.61/TK
381      Mc. Su=Suval(0, Mc)
382      Mc. Sm=Smval(0, Mc)
383 C
384      Mc. S1=103.37
385      Mc. Alph1=-.72607
386      Mc. S2=17.255
387      Mc. Alph2=-.86775
388      Mc. G1=1.2692
389      Mc. Beta1=.74491
390      Mc. G2=.48449
391      Mc. Beta2=.81155
392      Mc. F=.044758*EXP(-4439.8/(8.31*TK))
393      Mc. Ramda=-1.1104
394 C
395      GO TO 1900
396 C
397 C*****
398 C ** 2.25CR-1MO(PLATE) SCNV4 **
399 C*****
400 C
401 4000 IF(Mc. Im. NE. 2250) GO TO 4100
402      Mc. Sy=42.9566-4.59888E-2*Mc. Tc+1.47316E-4*Mc. Tc**2
403      #-1.93388E-7*Mc. Tc**3
404      ALPHY=1.5
405      IF(Mc. Tc. LT. 500.) ALPHY=1.5-.01*(500-Mc. Tc)/25
406      IF(Mc. Tc. LT. 425.) ALPHY=1.47+.01*(425-Mc. Tc)/25
407      IF(Mc. Tc. LT. 400.) ALPHY=1.48+.005*(400-Mc. Tc)/25
408      GO TO 4200
409 C
410 C*****
411 C ** 2.25CR-1MO(HEAT TRANSFER TUBE) STBA24 **
412 C*****
413 C
414 4100 IF(Mc. Im. NE. 2251) GO TO 5000
415      Mc. Sy=35.1902-3.29051E-2*Mc. Tc+6.01783E-5*Mc. Tc**2
416      #-6.32459E-8*Mc. Tc**3
417      ALPHY=1.5
418      IF(Mc. Tc. LT. 500.) ALPHY=1.5-.01*(500-Mc. Tc)/25
419      IF(Mc. Tc. LT. 475.) ALPHY=1.49-.02*(500-Mc. Tc)/25
420      IF(Mc. Tc. LT. 425.) ALPHY=1.45+.02*(425-Mc. Tc)/25
421 C
422 4200 IF(IFLG. NE. 1) ALPHY=1
423      Mc. Sy=Mc. Sy/ALPHY
424 C
425 C 10/27/88 以下3行修正
426      IF(Mc. Tc. LT. 450.)      Mc. E=20100-4*Mc. Tc
427      IF(Mc. Tc. GE. 450.)      Mc. E=.2164011E5-.7060216E1*Mc. Tc
428      #                          +.3684193E-2*Mc. Tc**2-.1014639E-4*Mc. Tc**3
429 C
430      Mc. Ak=83-8.75E-2*Mc. Tc
431      Mc. Am=.417-5.125E-4*Mc. Tc
432      Mc. Sp=Mc. Sy-Mc. Ak*.002**Mc. Am
433 C
434 C 10/27/88 以下3行修正
435      Mc. Akc=10.**(-5.011106+.5259598E-1*Mc. Tc-.1222898E-3*Mc. Tc**2
436      #                          +.9048195E-7*Mc. Tc**3)
437      Mc. Amc=-3.46931+.2953373E-1*Mc. Tc-.715088E-4*Mc. Tc**2
438      #                          +.5518891E-7*Mc. Tc**3
439      Mc. Spc=Mc. Sy*ALPHY-Mc. Ak*.002**Mc. Am
440 C
441      Mc. Tr0=-15.956105+18600.626/TK

```



```

442      Mc. Tr1=659. 76676/TK
443      Mc. Tr2=-2655. 7865/TK
444      Mc. Su=Suval(0, Mc)
445      Mc. Sm=Smval(0, Mc)
446 C
447      Mc. S1=45. 2986
448      Mc. Alph1=-. 56858
449      Mc. S2=14. 3245
450      Mc. Alph2=-. 82278
451      Mc. G1=. 30546
452      Mc. Beta1=. 59235
453      Mc. G2=. 26505
454      Mc. Beta2=. 81657
455      Mc. F=13. 5699*EXP(-33435. /(8. 31*TK))
456      Mc. Ramda=-1. 09319
457 C
458      Mc. A0t=1. 1101
459      Mc. A0r=-. 243973E-7*Mc. Tc**2
460      Mc. A1t=. 5709357
461      Mc. A1r=0
462      Mc. A2t=. 1874343
463      Mc. A2r=0
464      Mc. A4t=-. 1445453E-1+. 7045986E-5*Mc. Tc
465      Mc. A4r=-. 6562716E-4
466 C
467      RETURN
468 C
469 C *****
470 C                      Mod. 9Cr-1Mo
471 C
472 C TEMPORALY DATA 1988. 4. 14 N. K
473 C データ改訂・追加 1993. 5. 13 N. K
474 C 出典 弾塑性応力-ひずみ関係
475 C      繰り返し応力-ひずみ関係
476 C      クリープひずみ曲線
477 C      クリープ破断特性
478 C      以上 青砥他,(1990), 既存鋼種とのアナロジイによる新規開発材料の
479 C                      高温材料特性評価手法, 機論 A, 56巻523号
480 C      最適疲労破損式
481 C      小峰他,(1992), Mod. 9Cr-1Mo鋼(NT)の設計疲労線図(1991年度暫定値)の策定
482 C                      PNC ZN9410 92-089
483 C
484 C *****
485 C
486 5000 IF(Mc. Im. NE. 91) GO TO 6000
487      IF(Mc. Tc. LT. 400.) THEN
488          Mc. E=2. 040E4-8. *Mc. Tc
489      ELSE
490          Mc. E=2. 126E4-10. 125*Mc. Tc
491      ENDIF
492      Mc. Sy=54. 6367-7. 06234E-2*Mc. Tc+2. 56775E-4*Mc. Tc**2
493      #      -3. 52683E-7*Mc. Tc**3
494      ALPHY=1. 3
495      IF(Mc. Tc. LT. 500.) ALPHY=1. 3-. 01*(500-Mc. Tc)/25
496      IF(IFLG. NE. 1) ALPHY=1
497      Mc. Sv=Mc. Sy/ALPHY
498 C
499 C ** 弾塑性応力-ひずみ関係暫定式 **
500      Mc. Ak=83. 0-8. 75E-2*Mc. Tc
501      Mc. Am=. 417-5. 125E-4*Mc. Tc
502      Mc. Sp=Mc. Sy-Mc. Ak*. 002**Mc. Am
503 C
504 C ** 動的応力-ひずみ関係暫定式 **

```

```

505 IF(Mc. Tc. GE. 450. AND. Mc. Tc. LT. 650) THEN
506   Mc. Akc=10**(98. 7399-. 52935*Mc. Tc+. 96421E-3*Mc. Tc**2
507 #   -. 585E-6*Mc. Tc**3)
508   Mc. Amc=49. 895-. 27257*Mc. Tc+. 50036E-3*Mc. Tc**2
509 #   -. 30647E-6*Mc. Tc**3
510   Mc. Spc=Mc. Sy*ALPHY-Mc. Ak*. 002**Mc. Am
511 ELSE
512   Mc. Akc=10. *( -5. 011106+. 5259598E-1*Mc. Tc-. 1222898E-3*Mc. Tc**2
513 #   +. 9048195E-7*Mc. Tc**3)
514   Mc. Amc=-3. 46931+. 2953373E-1*Mc. Tc-. 715088E-4*Mc. Tc**2
515 #   +. 5518891E-7*Mc. Tc**3
516   Mc. Spc=Mc. Sy*ALPHY-Mc. Ak*. 002**Mc. Am
517 ENDIF
518 C Debug
519 C WRITE(6, 8000) Mc. Tc, Mc. Akc, AMC, SPC
520 C8000 FORMAT(1H, 'Mc. Tc:', E12. 4, ' AKC:', E12. 6, ' AMC:', E12. 6, ' SPC:', E12. 6)
521 C
522 C ** 主クリーブ破断関係暫定式 **
523   Mc. Tr0=-29. 11460+31808. 82/TK
524   Mc. Tr1=3055. 520/TK
525   Mc. Tr2=-5148. 248/TK
526   Mc. Su=Suval(0, Mc)
527   Mc. Sm=Smval(0, Mc)
528 C
529 C ** クリーブひずみ暫定式 **
530   Mc. S1=7*45. 2986
531   Mc. Alph1=-. 56858
532   Mc. S2=14. 3245
533   Mc. Alph2=-. 82278
534   Mc. G1=7*. 30546
535   Mc. Beta1=. 59235
536   Mc. G2=3. 5*. 26505
537   Mc. Beta2=. 81657
538   Mc. F=. 77322*EXP(-16956. /(8. 31*TK))
539   Mc. Ramda=-1. 0778
540 C
541 C ** 最適疲労破損式(PNC ZN9410 92-089) **
542   Mc. A0t=1. 0897913
543   Mc. A0r=-. 243973E-7*Mc. Tc**2
544   Mc. A1t=. 5709357
545   Mc. A1r=0
546   Mc. A2t=. 1874343
547   Mc. A2r=0
548   Mc. A4t=-. 1445453E-1+. 7045986E-5*Mc. Tc
549   Mc. A4r=-. 6562716E-4
550 C
551 C
552 GO TO 7000
553 C
554 C*****
555 C ** 316FR ** 90年度暫定式 1991年7月9日 BY 笠原、浜田
556 C*****
557 C
558 6000 IF(Mc. Im. NE. 3160) GO TO 7000
559   Mc. E=2. 10236E4-9. 71895*Mc. Tc
560   Mc. Sy=26. 8073-5. 04547E-2*Mc. Tc+8. 03961E-5*Mc. Tc**2
561 #   -5. 11282E-8*Mc. Tc**3
562   ALPHY=1. 3
563   IF(Mc. Tc. LT. 450.) ALPHY=1. 3-. 01*(450-Mc. Tc)/25
564   IF(IFLG. NE. 1) ALPHY=1
565   Mc. Sy=Mc. Sy/ALPHY
566 C
567   Mc. Ak=40. 0909-9. 6999E-3*Mc. Tc

```

```

568      Mc. Am=. 326245+6. 13276E-5*Mc. Tc
569      Mc. Sp=Mc. Sy-Mc. Ak*. 002**Mc. Am
570 C
571      Mc. Akc=10**(4. 139556-. 4434273E-2*Mc. Tc+. 1354228E-5*Mc. Tc**2
572      #+. 1593061E-8*Mc. Tc**3)
573      Mc. Amc=2. 171727-. 7045263E-2*Mc. Tc+. 7832692E-5*Mc. Tc**2
574      #- . 20836E-8*Mc. Tc**3
575      Mc. Spc=Mc. Sy*ALPHY-Mc. Ak*. 002**Mc. Am
576 C
577      Mc. Tr0=-25. 82042+32232. 27/TK
578      Mc. Tr1=-39. 74271/TK
579      Mc. Tr2=-3481. 803/TK
580      Mc. Su=Suval(0, Mc)
581      Mc. Sm=Smval(0, Mc)
582 C
583      Mc. S1=103. 37
584      Mc. Alph1=-. 72607
585      Mc. S2=17. 255
586      Mc. Alph2=-. 86775
587      Mc. G1=1. 2692
588      Mc. Beta1=. 74491
589      Mc. G2=. 48449
590      Mc. Beta2=. 81155
591 C      Mc. F=241. 33*EXP(-51222. /(8. 31*(TK+273. 15)))
592      Mc. F=241. 33*EXP(-51222. /(8. 31*TK))
593      Mc. Ramda=-1. 1032
594 C
595      GO TO 1900
596 C
597 C ** HERE NOT DEFINED NOW **
598 7000 CONTINUE
599      RETURN
600 C
601 C
602      END
603 C
604 C
605 C *****
606 C
607      FUNCTION Suval (IDUMMY, Mc)
608 C
609 C *****
610 C
611      INCLUDE 'STRUCT. FOR'
612      INTEGER*4 IDUMMY
613 C
614 C      RECORD /smc/ Mc [FAR, REFERENCE]
615 C
616 C      COMMON /MC/IM, TC, AD(38)
617 C
618      DIMENSION ISU(31, 5)
619      DATA ISU/520, 471, 447, 428, 415, 406, 400, 398, 396, 397, 397, 398, 399, 398
620      #, 397, 394, 388, 381, 372, 360, 346, 330, 311, 290, 268, 243, 218, 191, 164, 136
621      #, 109
622      #, 520, 481, 465, 450, 439, 431, 426, 423, 422, 422, 424, 426, 429, 431, 433, 433
623      #, 432, 428, 422, 413, 401, 386, 367, 346, 322, 295, 267, 237, 207, 178, 150
624      #, 520, 464, 443, 426, 413, 404, 398, 394, 392, 392, 393, 394, 396, 398, 399, 399
625      #, 399, 396, 392, 385, 376, 365, 351, 334, 315, 293, 268, 241, 212, 181, 147
626      #, 491, 449, 433, 423, 433, 441, 446, 439, 415, 368, 290, 212, 19*0
627      #, 444, 429, 417, 411, 413, 419, 420, 410, 385, 344, 299, 254, 19*0/
628 C
629      J=IDUMMY
630      J=1

```

```

631      IF(Mc. Im. EQ. 316. OR. Mc. Im. EQ. 3160) J=2
632      IF(Mc. Im. EQ. 321) J=3
633      IF(Mc. Im. EQ. 2250) J=4
634      IF(Mc. Im. EQ. 2251) J=5
635 C
636      Mc. Su=ISU(1, J)
637      IF(Mc. Tc. LE. 20.) GO TO 1000
638 C
639      IF((J-4)*(J-5).EQ. 0) GO TO 200
640      Mc. Su=ISU(1, J)+(ISU(2, J)-ISU(1, J))*(Mc. Tc-20)/55
641      IF(Mc. Tc. LE. 75.) GO TO 1000
642 C
643      K=(Mc. Tc-75)/25+2
644      Mc. Su=ISU(K, J)+(ISU(K+1, J)-ISU(K, J))*(Mc. Tc-25*(K+1))/25
645      SU0=ISU(9, J)
646      IF(K. GE. 9. AND. Mc. Su. GT. SU0) Mc. Su=SU0
647      GO TO 1000
648 C
649 200  Mc. Su=ISU(1, J)+(ISU(2, J)-ISU(1, J))*(Mc. Tc-20)/80
650      IF(Mc. Tc. LE. 100.) GO TO 1000
651 C
652      K=(Mc. Tc-100)/50+2
653      Mc. Su=ISU(K, J)+(ISU(K+1, J)-ISU(K, J))*(Mc. Tc-50*K)/50
654      SU0=ISU(4, J)
655      IF(K. GE. 4. AND. Mc. Su. GT. SU0) Mc. Su=SU0
656 C
657 1000 Suval=Mc. Su/9. 80665
658      RETURN
659      END
660 C
661 C
662 C *****
663 C
664      FUNCTION Smval (IDUMMY, Mc)
665 C
666 C *****
667 C
668 C
669      INCLUDE 'STRUCT.FOR'
670      INTEGER*4 IDUMMY
671 C
672 C      RECORD /smc/ Mc [FAR, REFERENCE]
673 C      COMMON /MC/IM, TC, DUMM(38)
674 C
675      DIMENSION VSM(24, 5)
676 C
677      DATA VSM/4*14., 13. 2, 12. 8, 12. 4, 12., 11. 7, 11. 5, 11. 3, 11. 1, 10. 9
678      #, 10. 6, 10. 4, 10. 2, 10., 9. 9, 9. 7, 9. 5, 9. 4, 9. 2, 2*9.
679      #, 4*14., 13. 5, 13. 2, 12. 8, 12. 4, 12. 1, 11. 9, 11. 6, 11. 4, 11. 2, 11. 1
680      #, 10. 9, 10. 7, 10. 6, 10. 4, 10. 3, 10. 2, 10., 9. 8, 2*9. 6
681      #, 4*14., 13. 1, 12. 7, 12. 2, 12., 11. 7, 11. 4, 11. 3, 11. 1, 11., 10. 9
682      #, 3*10. 9, 10. 8, 10. 7, 10. 5, 10. 4, 10. 3, 2*10. 1
683      #, 17. 7, 15. 9, 15. 8, 15. 2, 14. 9, 14. 8, 4*14. 7, 2*14. 6, 14. 5, 14. 3
684      #, 13. 8, 13. 3, 12. 5, 11. 5, 6*10. 4
685      #, 14., 2*12. 7, 12. 6, 10*12. 4, 12. 1, 11. 8, 11. 5, 10. 9, 6*10. 2/
686 C
687      J=IDUMMY
688      J=1
689      IF(Mc. Im. EQ. 316. OR. Mc. Im. EQ. 3160) J=2
690      IF(Mc. Im. EQ. 321) J=3
691      IF(Mc. Im. EQ. 2250) J=4
692      IF(Mc. Im. EQ. 2251) J=5
693      Smval=VSM(1, J)

```

1995年7月26日 13:15 (MMFUNC. FOR)

```

694     IF(Mc. Tc. LE. 40.) RETURN
695     Smval=VSM(1, J)+(VSM(2, J)-VSM(1, J))*(Mc. Tc-40)/35
696     IF(Mc. Tc. LE. 75.) RETURN
697     Smval=VSM(2, J)+(VSM(3, J)-VSM(2, J))*(Mc. Tc-75)/25
698     IF(Mc. Tc. LE. 100.) RETURN
699     N=Mc. Tc/50+1
700     Smval=VSM(N, J)+(VSM(N+1, J)-VSM(N, J))*(Mc. Tc-50*(N-1))/50
701     IF(Mc. Tc. LE. 200.) RETURN
702     N=Mc. Tc/25-3
703     Smval=VSM(N, J)+(VSM(N+1, J)-VSM(N, J))*(Mc. Tc-25*(N+3))/25
704     RETURN
705     END
706 C
707 C
708 C *****
709 C
710     REAL*4 FUNCTION Eepm [PASCAL] (X, Mc)
711 C
712 C *****
713 C
714 C
715     INCLUDE 'STRUCT. FOR'
716     REAL*4 X
717 C
718 C RECORD /smc/ Mc [FAR, REFERENCE]
719 C COMMON /MC/IM, TC, E, SY, SP, AM, AK, AD(33)
720 C
721     IS=1
722     IF(X. LT. 0.) IS=-1
723     Z=IS*X
724     IF(Z. GT. Mc. Sp) GO TO 100
725     Eepm=X/Mc. E
726     RETURN
727 100 Eepm=X/Mc. E+IS*((Z-Mc. Sp)/Mc. Ak)**(1/Mc. Am)
728     RETURN
729     END
730 C
731 C
732 C *****
733 C
734     REAL*4 FUNCTION Sepm [PASCAL] (X, Mc)
735 C
736 C *****
737 C
738 C
739     INCLUDE 'STRUCT. FOR'
740     REAL*4 X
741 C
742 C RECORD /MC/ Mc
743 C COMMON /MC/IM, TC, E, SY, SP, AM, AK, AD(33)
744 C
745     IS=1
746     IF(X. LT. .0) IS=-1
747     Z=IS*X
748     IF(Mc. E*Z. GT. Mc. Sp) GO TO 100
749     Sepm=Mc. E*X
750     RETURN
751 100 S=Mc. Ak*Z**Mc. Am+Mc. Sp
752     DE=S/Mc. E
753     IF(S. LT. Mc. E*Z) GO TO 120
754     S=Mc. E*Z
755     DE=((S-Mc. Sp)/Mc. Ak)**(1/Mc. Am)
756 120 DS=DE/(1/Mc. E+((S-Mc. Sp)/Mc. Ak)**(1/Mc. Am-1)/(Mc. Am*Mc. Ak))

```

```

757      S=S-DS
758      DE=S/Mc. E+(((S-Mc. Sp)/Mc. Ak)**(1/Mc. Am)-Z)
759      IF(DS/S. GT. 1E-6) GO TO 120
760      Sepm=IS*S
761      RETURN
762      END
763 C
764 C
765 C *****
766 C
767      FUNCTION Eepr(X, Mc)
768 C
769 C *****
770 C
771 C
772      INCLUDE 'STRUCT. FOR'
773      REAL*4 X
774 C
775 C RECORD /MC/ Mc
776 C COMMON /MC/IM, TC, E, AD1(4), SPC, AMC, AKC, AD2(30)
777 C
778      IS=1
779      IF(X. LT. .0) IS=-1
780      Z=IS*X
781      IF(Z. GT. 2. *Mc. Spc) GO TO 100
782      Eepr=X/Mc. E
783      RETURN
784 100 Eepr=X/Mc. E+IS*((Z-2*Mc. Spc)/Mc. Akc)**(1/Mc. Amc)
785      RETURN
786      END
787 C
788 C
789 C *****
790 C
791 C INTERFACE TO REAL*4 FUNCTION Sepr [PASCAL] (X, Mc)
792 REAL*4 FUNCTION Sepr [PASCAL] (X, Mc)
793 C
794 C *****
795 C
796 C
797      INCLUDE 'STRUCT. FOR'
798      REAL*4 X
799 C
800 C RECORD /smc/ Mc [FAR. REFERENCE]
801 C COMMON /MC/IM, TC, E, AD1(4), SPC, AMC, AKC, AD2(30)
802 C
803      IS=1
804      IF(X. LT. .0) IS=-1
805      Z=IS*X
806      IF(Mc. E*Z. GT. 2. *Mc. Spc) GO TO 100
807      Sepr=Mc. E*X
808      RETURN
809 100 S=Mc. Akc*Z**Mc. Amc+2*Mc. Spc
810      DE=S/Mc. E
811      IF(S. LT. Mc. E*Z) GO TO 130
812      S=Mc. E*Z
813      DE=((S-2*Mc. Spc)/Mc. Akc)**(1/Mc. Amc)
814      GO TO 130
815 120 DE=S/Mc. E+(((S-2*Mc. Spc)/Mc. Akc)**(1/Mc. Amc)-Z)
816 130 DS=DE/(1/Mc. E+((S-2*Mc. Spc)/Mc. Akc)**(1/Mc. Amc-1)
817      #/(Mc. Amc*Mc. Akc))
818      S=S-DS
819 C DE=S/Mc. E+(((S-2*Mc. Spc)/Mc. Akc)**(1/Mc. Amc)-Z)

```

```

820      IF(DS/S.GT.3E-6) GO TO 120
821 C      Sepr=IS*S
822      Sepr=IS*S/2.0
823      RETURN
824      END
825 C
826 C
827 C *****
828 C
829      FUNCTION Scrr(X, Mc)
830 C
831 C *****
832 C
833 C
834      INCLUDE 'STRUCT. FOR'
835      REAL*4      X
836 C
837 C      RECORD /MC/ Mc
838 C      COMMON /MC/AD1(10), TR0, TR1, TR2, AD2(27)
839 C
840      Scrr=1E-20
841      IF(X.LE..0) RETURN
842      D=Mc. Tr1**2-4*Mc. Tr2*(Mc. Tr0-ALOG10(X))
843      IF(D.LE..0) RETURN
844      Z=.5*(-Mc. Tr1-SQRT(D))/Mc. Tr2
845      Scrr=10**(Z)
846      RETURN
847      END
848 C
849 C
850 C *****
851 C
852      FUNCTION Ecr(S, T, ALPHC, Mc)
853 C
854 C *****
855 C
856 C
857      INCLUDE 'STRUCT. FOR'
858      REAL*4      S, T, ALPHC
859 C
860 C      RECORD /MC/ Mc
861 C      COMMON /MC/AD1(20), S1, A1, S2, A2, G1, B1, G2, B2, F, R, AD2(10)
862 C
863      Ecr=0
864      IF(S.EQ..0) RETURN
865      IS=1
866      IF(S.LT..0) IS=-1
867      Z=IS*S
868 C      TR=Tcrr(Z, Mc)/ALPHC
869      EM=Mc. F*TR**Mc. Ramda
870      R1=Mc. S1*TR**Mc. Alph1
871      R2=Mc. S2*TR**Mc. Alph2
872      C1=Mc. G1*EM**Mc. Beta1/R1
873      C2=Mc. G2*EM**Mc. Beta2/R2
874      Ecr=IS*(C1+C2+EM*T)
875      IF(R2*T.GT.30.) RETURN
876      Ecr=IS*(C1+C2*(1-EXP(-R2*T))+EM*T)
877      IF(R1*T.GT.30.) RETURN
878      Ecr=IS*(C1*(1-EXP(-R1*T))+C2*(1-EXP(-R2*T))+EM*T)
879      RETURN
880      END
881 C
882 C

```

```

883 C *****
884 C
885 C   INTERFACE TO REAL*4 FUNCTION Anf [PASCAL] (ET, EDOT, Mc)
886 C   REAL*4 FUNCTION Anf [PASCAL] (ET, EDOT, Mc)
887 C
888 C *****
889 C
890 C
891 C   INCLUDE 'STRUCT.FOR'
892 C   REAL*4   ET, EDOT
893 C
894 C   RECORD /smc/ Mc [FAR, REFERENCE]
895 C   COMMON /MC/AD1(30), A0T, A0R, A1T, A1R, A2T, A2R, A4T, A4R, AD2(2)
896 C
897 C   Anf=0.0
898 C   IF(ET*EDOT.EQ.0) RETURN
899 C   Z=ALOG10(ABS(ET))
900 C   R=ALOG10(ABS(EDOT))
901 C   1990年03月06日 修正 浜田 桁溢れへの対策 (PRO-FORTRAN)
902 C   Anf=10.0**((A0T+A0R*R+(A1T+A1R*R)*Z+(A2T+A2R*R)*Z**2
903 C   #+(A4T+A4R*R)*Z**4)**(-2)
904 C   DANF=(Mc.A0t+Mc.A0r*R+(Mc.A1t+Mc.A1r*R)*Z+(Mc.A2t+Mc.A2r*R)
905 C   **Z**2+(Mc.A4t+Mc.A4r*R)*Z**4)**(-2)
906 C   IF(DANF.GT.35.0) DANF = 35.0
907 C   IF(DANF.LT.-35.0) DANF = -35.0
908 C   Anf=10.0**(DANF)
909 C   RETURN
910 C   END
911 C
912 C
913 C *****
914 C
915 C   FUNCTION Sids(SG, Mc)
916 C
917 C *****
918 C
919 C
920 C   INCLUDE 'STRUCT.FOR'
921 C   REAL*4   SG
922 C
923 C   RECORD /MC/ Mc
924 C   COMMON /MC/IM, TC, E, SY, SP, AM, AK, AD(33)
925 C
926 C   S=SG/3+.005/3*Mc.E
927 C   Z=Eepm(S, Mc)
928 C   IF(Z.LE.S/Mc.E) RETURN
929 C   DZ=Z-S/Mc.E
930 100 DS=DZ/(3/Mc.E+((S-Mc.Sp)/Mc.Ak)**(1/Mc.Am-1)/(Mc.Am*Mc.Ak))
931 C   S=S-DS
932 C   DZ=3*S/Mc.E+((S-Mc.Sp)/Mc.Ak)**(1/Mc.Am)-SG/Mc.E-.005
933 C   IF(DS/S.GT.3E-6) GO TO 100
934 C   Sids=S
935 C   RETURN
936 C   END
937 C
938 C
939 C *****
940 C
941 C   FUNCTION Dstar(SN, TSTAR, DC1, Mc)
942 C   REAL*4 FUNCTION Dstar [PASCAL] (SN, TSTAR, DC1, Mc)
943 C
944 C *****
945 C

```



```

946 C
947     INCLUDE 'STRUCT. FOR'
948     REAL*4   SN, TSTAR, DC1
949 C
950 C     RECORD /MC/ Mc
951 C     COMMON /MC/IM, TC, E, DMC1(11), SM, DMC2(25)
952 C
953     TR=2*TSTAR/DC1
954     IF(Mc. Im. NE. 304) GO TO 2000
955     CN1=6E-5
956     CN2=1E-3
957     DNO=1.26E-5
958     AMN=.166
959 1800  TN=1/(CN1*ALOG10(TR)+CN2)-273.15
960     Dstar=0
961     IF(Mc. Tc. LE. TN) RETURN
962     Dstar=DNO*TR**AMN*(Mc. Tc-TN)
963     RETURN
964 C
965 2000  IF(Mc. Im. NE. 316) GO TO 3000
966     CN1=.6E-4
967     CN2=.96E-3
968     DNO=2.1E-6
969     AMN=.35
970     GO TO 1800
971 C
972 3000  IF(Mc. Im. NE. 321) GO TO 4000
973     CN1=15.7
974     CN2=.93
975     DNO=6.0E-4
976     AMN=2.9
977     Dstar=0
978     SG=Scrr(10.*TR, Mc)
979     IF(SG. GE. CN1/CN2) RETURN
980     Dstar=DNO*(CN1-CN2*SG)**AMN
981     RETURN
982 C
983 4000  Dstar=0
984     SG=Scrr(10.*TR, Mc)
985     SI=AMIN1(.5*SN, SN-1.5*Mc. Sm)
986     IF(SI. LE. SG) RETURN
987     D0=5E-3*TR**(-.223)
988     C0=2
989     AM0=2.2
990     X=(Mc. Tc-375)/175
991     BETAN=.28-.25*X
992     D1=.01*(TR/10)**BETAN
993     C1=7
994     AMN=4-1.1*X
995     Dstar=AMAX1(D0*((SI-SG)/C0)**AM0, D1*((SI-SG)/C1)**AMN)
996     RETURN
997     END
998 C *****
999 C
1000 C     FUNCTION Ddstar(ET, TSTAR, DC1, Mc)
1001     REAL*4 FUNCTION Ddstar [PASCAL] (ET, TSTAR, DC1, Mc)
1002 C
1003 C *****
1004 C
1005 C
1006     INCLUDE 'STRUCT. FOR'
1007     REAL*4   ET, TSTAR, DC1
1008 C

```

```

1009 C   RECORD /MC/ Mc
1010 C   COMMON /MC/IM, TC, E, DMCC1(11), SM, DMC2(24), IREN
1011 C
1012     Ddstar=0
1013     TR=2*TSTAR/DC1
1014     SG=Scrr(10. *TR, Mc)
1015     X=AMAX1((1. 5*Mc. Sm+SG)/Mc. E, 2. *SG/Mc. E)
1016     IF(ET. LE. X) RETURN
1017 C
1018     TK0=273. 15
1019     TCO=Mc. Tc
1020 C
1021     IF(Mc. Im. NE. 304) GO TO 2000
1022     DPO=1. 2E-4
1023     CO=1
1024     ANPO=2360/(Mc. Tc+TK0)-1
1025 1200 SI=. 5*Sepr(ET, Mc)
1026     IF(SI. LE. SG) RETURN
1027     D=DPO*((SI-SG)/CO)**ANPO
1028 C
1029     DT=1
1030 1400 Mc. Tc=Mc. Tc-DT
1031 C   IREN = IREN+1
1032     CALL Mset(0, Mc)
1033     SG=Scrr(10. *TR, Mc)
1034     SI=. 5*Sepr(ET, Mc)
1035     IF(SI. LE. SG) GO TO 1900
1036     IF(Mc. Im. EQ. 316. AND. Mc. Tc. LT. 500. )
1037     #           DPO=EXP(7. 8-2. 2E-2*(Mc. Tc+TK0))
1038     IF(Mc. Im. EQ. 304) ANPO=2360/(Mc. Tc+TK0)-1
1039     IF(Mc. Im. EQ. 316) ANPO=5220/(Mc. Tc+TK0)-4. 2
1040     IF(Mc. Im. EQ. 321) ANPO=800/(Mc. Tc+TK0)+1. 6
1041     DI=DPO*((SI-SG)/CO)**ANPO
1042     IF(DI. LE. D) GO TO 1900
1043     D=DI
1044     GO TO 1400
1045 C
1046 1900 Ddstar=D
1047     IF(Ddstar. LT. 1E-4) Ddstar=1E-4
1048     Mc. Tc=TCO
1049     CALL Mset(0, Mc)
1050     RETURN
1051 C
1052 2000 IF(Mc. Im. NE. 316) GO TO 3000
1053     DPO=1E-4
1054     IF(Mc. Tc. LE. 500. ) DPO=EXP(7. 8-2. 2E-2*(Mc. Tc+TK0))
1055     CO=1
1056     ANPO=5220/(Mc. Tc+TK0)-4. 2
1057     GO TO 1200
1058 C
1059 3000 IF(Mc. Im. NE. 321) GO TO 4000
1060     DPO=1. 8E-4
1061     CO=1
1062     ANPO=800/(Mc. Tc+TK0)+1. 6
1063     GO TO 1200
1064 C
1065 4000 DPO=5E-3*TR**(-. 233)
1066     CO=5
1067     ANPO=2
1068     DPI=1E-4
1069     C1=3
1070     ANP1=4-2. 7*(Mc. Tc-375)/175+1. 3*((Mc. Tc-375)/175)**2
1071     SI=Sepm(. 5*ET, Mc)

```

```

1072      IF(SI. LE. SG) RETURN
1073      D=AMAX1(DP0*((SI-SG)/C0)**ANP0, DP1*((SI-SG)/C1)**ANP1)
1074 C
1075      DT=1
1076 4400  Mc. Tc=Mc. Tc-DT
1077      CALL Mset(0, Mc)
1078      SG=Scrr(10. *TR, Mc)
1079      SI=Sepm(. 5*ET, Mc) .
1080      IF(SI. LE. SG) GO TO 1900
1081      ANP1=4-2. 7*(Mc. Tc-375)/175+1. 3*((Mc. Tc-375)/175)**2
1082      D1=AMAX1(DP0*((SI-SG)/C0)**ANP0, DP1*((SI-SG)/C1)**ANP1)
1083      IF(D1. LE. D) GO TO 1900
1084      D=D1
1085      GO TO 4400
1086      END
1087 C
1088 C
1089 C *****
1090 C
1091      SUBROUTINE R1x2(SI, EC1, ALPHC, Q, T, SMIN, DC, Mc)
1092 C
1093 C          S1=0 近傍の計算機能を改良          NOV. 8/88
1094 C          S=SMIN 近傍の誤差を改善          DEC. 10/88
1095 C
1096 C *****
1097 C
1098 C
1099      INCLUDE 'STRUCT. FOR'
1100 C
1101 C      RECORD /MC/ Mc
1102 C          COMMON /MC/IM, TC, E, AD(37)
1103 C          REAL*8 DDC
1104 C
1105 C          DC=0
1106 C          IF(S1*SMIN. LE. 0) RETURN
1107 C          JS=1
1108 C          IF(S1. LE. 0) JS=-1
1109 C          S0=JS*S1
1110 C          DS=. 01
1111 C          IF(S0. LT. JS*SMIN. OR. S0. LT. 10*DS) RETURN
1112 C          IS=0
1113 C          T=0
1114 C          DDC=0
1115 100    S=S0-(IS+. 5)*DS
1116 C          EC=EC1+(IS+. 5)*Q*DS/Mc. E
1117 C          TZ=Tcr(S, EC, ALPHC, Mc)
1118 C          DT=Q*DS/(Mc. E*Ecrr(S, TZ, ALPHC, Mc))
1119 C          IF(S-. 5*DS. LE. JS*SMIN) GOTO 200
1120 C          IS=IS+1
1121 C          T=T+DT
1122 C          Z=DT/Tcrr(S, Mc)
1123 C          DDC=DDC+DBLE(Z)
1124 C          GOTO 100
1125 C
1126 200    T=T+(S+. 5*DS-JS*SMIN)/DS*DT
1127 C          DC=DDC+(S+. 5*DS-JS*SMIN)/DS*DT/Tcrr(. 5*(S+. 5*DS+JS*SMIN), Mc)
1128 C          RETURN
1129 C          END
1130 C
1131 C
1132 C *****
1133 C
1134 C      SUBROUTINE Mc1ist(IFLG)

```

```

1135 C
1136 C *****
1137 C
1138 C
1139 C     INCLUDE 'STRUCT. FOR'
1140 C
1141 C     RECORD /MC/ Mc
1142 C     COMMON /MC/AMC(10,4)
1143 C     EQUIVALENCE (IM,AMC(1,1))
1144 C
1145 C     IF(IFLG*(IFLG-1).EQ.0) WRITE(6,5010) Mc, Im, (Mc, Amc(I,1), I=2,10)
1146 C     IF(IFLG*(IFLG-2).EQ.0) WRITE(6,5020) (Mc, Amc(I,2), I=1,5)
1147 C     IF(IFLG*(IFLG-3).EQ.0) WRITE(6,5030) (Mc, Amc(I,3), I=1,10)
1148 C     IF(IFLG*(IFLG-4).EQ.0) WRITE(6,5040) (Mc, Amc(I,4), I=1,8)
1149 C     RETURN
1150 C
1151 C5010 FORMAT(1H, 'MATERIAL:', I5, 5X, 'TEMP:', F8.3, ' DEG:C', /
1152 C     #, '** ELASTICITY & PLASTICITY **' /
1153 C     #, 1H, 'E:', F10.3, ' KG/MM**2', 5X, 'SY:', F8.3, ' KG/MM**2' /
1154 C     #, 1H, 'MONOTONIC SP:', F8.3, ' KG/MM**2', 5X, 'M:', F8.5, 5X
1155 C     #, 'K:', F8.3, ' KG/MM**2' /
1156 C     #, 1H, 'CYCLIC SP:', F8.3, ' KG/MM**2', 5X, 'M:', F8.5, 5X
1157 C     #, 'K:', F8.3, ' KG/MM**2')
1158 C5020 FORMAT(1H, '** CREEP RUPTURE **' /
1159 C     #, 1H, 'TR0:', E13.6, 5X, 'TR1:', E13.6, 5X, 'TR2:', E13.6 /
1160 C     #, 1H, 'SU:', F10.6, 5X, 'SM:', F10.6)
1161 C5030 FORMAT(1H, '** CREEP STRAIN **' /
1162 C     #, 1H, 'S1:', E13.6, 5X, 'ALPH1:', E13.6 /
1163 C     #, 1H, 'S2:', E13.6, 5X, 'ALPH2:', E13.6 /
1164 C     #, 1H, 'C1:', E13.6, 5X, 'BETA1:', E13.6 /
1165 C     #, 1H, 'C2:', E13.6, 5X, 'BETA2:', E13.6 /
1166 C     #, 1H, 'F:', E13.6, 5X, 'RAMDA:', E13.6)
1167 C5040 FORMAT(1H, '** FATIGUE LIFE **' /
1168 C     #, 1H, 'A0T:', E13.6, 5X, 'A0R:', E13.6 /
1169 C     #, 1H, 'A1T:', E13.6, 5X, 'A1R:', E13.6 /
1170 C     #, 1H, 'A2T:', E13.6, 5X, 'A2R:', E13.6 /
1171 C     #, 1H, 'A4T:', E13.6, 5X, 'A4R:', E13.6)
1172 C
1173 C     END
1174 C
1175 C
1176 C *****
1177 C
1178 C     SUBROUTINE Crlist(S,Mc)
1179 C
1180 C *****
1181 C
1182 C
1183 C     INCLUDE 'STRUCT. FOR'
1184 C     REAL*4 S
1185 C
1186 C     RECORD /MC/ Mc
1187 C     COMMON /MC/AD1(20), S1, ALPH1, S2, ALPH2, G1, BETA1, G2, BETA2
1188 C     #, F, RAMDA, AD2(10)
1189 C
1190 C     IF(S.LE..0) RETURN
1191 C     TR=Tcrr(S,Mc)
1192 C     R1=Mc. S1*TR**Mc. Alph1
1193 C     R2=Mc. S2*TR**Mc. Alph2
1194 C     EM=Mc. F*TR**Mc. Ramda
1195 C     C1=Mc. G1*EM**Mc. Beta1/R1
1196 C     C2=Mc. G2*EM**Mc. Beta2/R2
1197 C     WRITE(6,5000) TR, EM, C1, R1, C2, R2

```

```

1198     RETURN
1199 C
1200 5000  FORMAT(1H , 'TR:', E13. 6, ' HR', 8X, 'EM:', E13. 6, ' MM/MM/HR' /
1201     #. 1H , ' C1:', E13. 6, ' MM/MM', 5X, ' R1:', E13. 6, ' /HR' /
1202     #. 1H , ' C2:', E13. 6, ' MM/MM', 5X, ' R2:', E13. 6, ' /HR' )
1203 C
1204     END
1205 C
1206 C
1207 C *****
1208 C
1209 C     SUBROUTINE Nfcal( NF, EDOT )
1210 C
1211 C *****
1212 C
1213 C
1214 C     INCLUDE 'STRUCT.FOR'
1215 C
1216 C     RECORD /MC/ Mc
1217 C     COMMON /MC/AD1(30), A0T, A0R, A1T, A1R, A2T, A2R, A4T, A4R, AD2(2)
1218 C
1219 C     COMPLEX Z
1220 C     DIMENSION A(5), Z(4), VW(30), WZ(4)
1221 C
1222 C     N=4
1223 C     Anf=FLOAT(NF, Mc, Im)
1224 C     R=ALOG10(ABS(EDOT))
1225 C     A(1)=A4T+A4R*R
1226 C     A(2)=0
1227 C     A(3)=A2T+A2R*R
1228 C     A(4)=A1T+A1R*R
1229 C     A(5)=(A0T+A0R*R)-((ALOG10(Anf))**(-0.5))
1230 C
1231 C     CALL RJETR(A, N, Z, VW, ICON)
1232 C
1233 C     WRITE(6, '(1H , '' N='', I2, '' CONDITION CODE='', I5)') N, ICON
1234 C     IF (ICON.EQ.30000) THEN
1235 C         WRITE(6, '(1H , '' MODULE RJETR ABEND'')')
1236 C     ELSE
1237 C         J=0
1238 C         DO 10 I=1, N
1239 C             CIMG=AIMAG(Z(I))
1240 C             CRAL=REAL(Z(I))
1241 C             IF (CIMG.EQ.0.0) THEN
1242 C                 J=J+1
1243 C                 WZ(J)=CRAL
1244 C             END IF
1245 C 10     CONTINUE
1246 C
1247 C         DO 20 I=1, J
1248 C             WZ(I)=10**(WZ(I))
1249 C             IF (WZ(I).GE.0.0.AND.WZ(I).LE.0.1) WRITE(6, 1000) WZ(I)
1250 C 20     CONTINUE
1251 C
1252 C     END IF
1253 C1000  FORMAT( / 1H , ' ET=', E20. 8 )
1254 C     RETURN
1255 C     END
1256 C *****
1257 C     SUBROUTINE Sprin
1258 C     STRESS RANGE CALCULATION ( TRESCA & MISES )
1259 C *****
1260     SUBROUTINE Sprin(S, TRESCA, MISES, IREN)

```

```

1261 C
1262     DIMENSION S(4,3,100), S1(4), P(5)
1263     REAL*4 MISES(100), TRESCA(100)
1264 C
1265     DO 100 I=1, IREN
1266     DO 10 J=1, 4
1267         S1(J) = S(J, 1, I)-S(J, 2, I)
1268 10 CONTINUE
1269 C
1270     P(3) = S1(3)
1271     SS1 = (S1(1)+S1(2))/2.0
1272     SS2 = (S1(1)-S1(2))/2.0
1273     SS3 = SQRT(SS2**2+S1(4)**2)
1274     P(1) = SS1+SS3
1275     P(2) = SS1-SS3
1276     SS1 = ABS(P(1)-P(2))
1277     SS2 = ABS(P(1)-P(3))
1278     SS3 = ABS(P(2)-P(3))
1279 C TRESCA
1280     TRESCA(I) = AMAX1(SS1, SS2, SS3)
1281 C MISES
1282     MISES(I) = SQRT((SS1**2+SS2**2+SS3**2)/2.0)
1283 100 CONTINUE
1284 C
1285     RETURN
1286 C
1287     END
1288 C
1289 C*****
1290 C
1291 C     線膨張係数算出プログラム
1292     REAL*4 FUNCTION Alpha [PASCAL] (Mc)
1293 C
1294 C *****
1295 C
1296 C
1297     INCLUDE 'STRUCT. FOR'
1298 C
1299 C     RECORD /smc/ Mc [FAR, REFERENCE]
1300 C     COMMON /MC/IM, TC, E, SY, SP, AM, AK, SPC, AMC, AKC
1301 C     #, TR0, TR1, TR2, SU, SM, DUMM2(5)
1302 C     #, S1, ALPH1, S2, ALPH2, G1, BETA1, G2, BETA2, F, RAMDA
1303 C     #, A0T, A0R, A1T, A1R, A2T, A2R, A4T, A4R, DUMM4(2)
1304 C
1305     DIMENSION RTEMP(9), AL(9,2)
1306     EXTERNAL Xy
1307 C
1308     DATA RTEMP/ 250.0, 300.0, 350.0, 400.0, 450.0, 500.0,
1309 *             550.0, 600.0, 650.0 /
1310     DATA AL/ 18.36, 18.79, 19.19, 19.57, 19.93, 20.28, 20.60, 20.87,
1311 *           21.09,
1312 *           14.14, 14.49, 14.75, 15.00, 15.20, 15.36, 15.49, 0.00,
1313 *           0.00 /
1314 C
1315     IF(Mc. Im. EQ. 91. OR. Mc. Im. EQ. 2250) THEN
1316         IC = 7
1317         IAL = 2
1318     ELSE
1319         IC = 9
1320         IAL = 1
1321     ENDIF
1322 C
1323     DO 10 I=1, IC

```

```

1324      IF(RTEMP(I).EQ. Mc. Tc) GO TO 20
1325      IF(RTEMP(I).GT. Mc. Tc) GO TO 30
1326      10 CONTINUE
1327      I = IC
1328      30 IF(I.EQ.1) I = 2
1329      Alpha = Xy(RTEMP(I), RTEMP(I-1), AL(I, IAL), AL(I-1, IAL), Mc. Tc)
1330      Alpha = Alpha*1.0E-6
1331      GO TO 40
1332      20 Alpha = AL(I, IAL)*1.0E-6
1333      40 CONTINUE
1334 C
1335      RETURN
1336 C
1337      END
1338 C
1339 C*****
1340 C      直線内挿入を行う
1341 C
1342      FUNCTION Xy(X2, X1, Y2, Y1, XX)
1343 C
1344 C*****
1345 C
1346      REAL*4 X2, X1, Y2, Y1, XX
1347
1348      Xy = (Y2-Y1)*(XX-X1)/(X2-X1)+Y1
1349 C
1350      RETURN
1351 C
1352      END
1353 C*****
1354 C=====
1355 C      ポアソン比を換算するサブルーチン
1356 C
1357 C      INTERFACE TO REAL*4 FUNCTION Calcnu [PASCAL] (Mc)
1358      REAL*4 FUNCTION Calcnu [PASCAL] (Mc)
1359 C
1360 C=====
1361 C*****
1362 C
1363      INCLUDE 'STRUCT. FOR'
1364 C
1365      RECORD /MC/ Mc
1366 C
1367      COMMON /MC/IM, TC, E, SY, SP, AM, AK, SPC, AMC, AKC
1368 C      #, TRO, TRI, TR2, SU, SM, DUMM2(5)
1369 C      #, S1, ALPH1, S2, ALPH2, G1, BETA1, G2, BETA2, F, RAMDA
1370 C      #, AOT, AOR, A1T, A1R, A2T, A2R, A4T, A4R, DUMM4(2)
1371 C
1372      REAL SUS, SCMV4, CRMO, NU
1373      DIMENSION SUS(14), SCMV4(4), CRMO(7)
1374      DATA SUS / 0.266, 0.268, 0.272, 0.275, 0.279, 0.283, 0.287, 0.291
1375      & .0.295, 0.298, 0.302, 0.306, 0.310, 0.314 /
1376      DATA SCMV4 / 0.26, 0.28, 0.29, 0.30 /
1377      DATA CRMO / 0.300, 0.301, 0.302, 0.304, 0.306, 0.308, 0.310 /
1378
1379      IF(Mc. Im. EQ. 2250) GO TO 1
1380      IF(Mc. Im. EQ. 91 ) GO TO 2
1381
1382 C-----
1383 C      SUS304, SUS316, 316FR
1384 C-----
1385
1386      IF(Mc. Tc. LT. 20. 0) THEN

```

```

1387     NU = SUS(1)
1388     ENDIF
1389
1390     IF(Mc. Tc. GE. 20. 0. AND. Mc. Tc. LT. 50. 0) THEN
1391         NU = SUS(1)+((SUS(2)-SUS(1))/30)*(Mc. Tc-20. 0)
1392     ENDIF
1393
1394     IF(Mc. Tc. GE. 50. 0. AND. Mc. Tc. LT. 100. 0) THEN
1395         NU = SUS(2)+((SUS(3)-SUS(2))/50)*(Mc. Tc-50. 0)
1396     ENDIF
1397
1398     IF(Mc. Tc. GE. 100. 0. AND. Mc. Tc. LT. 150. 0) THEN
1399         NU = SUS(3)+((SUS(4)-SUS(3))/50)*(Mc. Tc-100. 0)
1400     ENDIF
1401
1402     IF(Mc. Tc. GE. 150. 0. AND. Mc. Tc. LT. 200. 0) THEN
1403         NU = SUS(4)+((SUS(5)-SUS(4))/50)*(Mc. Tc-150. 0)
1404     ENDIF
1405
1406     IF(Mc. Tc. GE. 200. 0. AND. Mc. Tc. LT. 250. 0) THEN
1407         NU = SUS(5)+((SUS(6)-SUS(5))/50)*(Mc. Tc-200. 0)
1408     ENDIF
1409
1410     IF(Mc. Tc. GE. 250. 0. AND. Mc. Tc. LT. 300. 0) THEN
1411         NU = SUS(6)+((SUS(7)-SUS(6))/50)*(Mc. Tc-250. 0)
1412     ENDIF
1413
1414     IF(Mc. Tc. GE. 300. 0. AND. Mc. Tc. LT. 350. 0) THEN
1415         NU = SUS(7)+((SUS(8)-SUS(7))/50)*(Mc. Tc-300. 0)
1416     ENDIF
1417
1418     IF(Mc. Tc. GE. 350. 0. AND. Mc. Tc. LT. 400. 0) THEN
1419         NU = SUS(8)+((SUS(9)-SUS(8))/50)*(Mc. Tc-350. 0)
1420     ENDIF
1421
1422     IF(Mc. Tc. GE. 400. 0. AND. Mc. Tc. LT. 450. 0) THEN
1423         NU = SUS(9)+((SUS(10)-SUS(9))/50)*(Mc. Tc-400. 0)
1424     ENDIF
1425
1426     IF(Mc. Tc. GE. 450. 0. AND. Mc. Tc. LT. 500. 0) THEN
1427         NU = SUS(10)+((SUS(11)-SUS(10))/50)*(Mc. Tc-450. 0)
1428     ENDIF
1429
1430     IF(Mc. Tc. GE. 500. 0. AND. Mc. Tc. LT. 550. 0) THEN
1431         NU = SUS(11)+((SUS(12)-SUS(11))/50)*(Mc. Tc-500. 0)
1432     ENDIF
1433
1434     IF(Mc. Tc. GE. 550. 0. AND. Mc. Tc. LT. 600. 0) THEN
1435         NU = SUS(12)+((SUS(13)-SUS(12))/50)*(Mc. Tc-550. 0)
1436     ENDIF
1437
1438     IF(Mc. Tc. GE. 600. 0. AND. Mc. Tc. LT. 650. 0) THEN
1439         NU = SUS(13)+((SUS(14)-SUS(13))/50)*(Mc. Tc-600. 0)
1440     ENDIF
1441
1442     IF(Mc. Tc. GE. 650. 0) THEN
1443         NU = SUS(14)
1444     ENDIF
1445
1446     GO TO 999
1447
1448 C-----
1449 C

```

SCMV4



```

1450 C-----
1451
1452 1 IF(Mc. Tc. LT. 20. 0) THEN
1453     NU = SCMV4(1)
1454     ENDIF
1455
1456     IF(Mc. Tc. GE. 20. 0. AND. Mc. Tc. LT. 50. 0) THEN
1457         NU = SCMV4(1)+((SCMV4(2)-SCMV4(1))/30)*(Mc. Tc-20. 0)
1458     ENDIF
1459
1460     IF(Mc. Tc. GE. 50. 0. AND. Mc. Tc. LT. 100. 0) THEN
1461         NU = SCMV4(2)
1462     ENDIF
1463
1464     IF(Mc. Tc. GE. 100. 0. AND. Mc. Tc. LT. 150. 0) THEN
1465         NU = SCMV4(2)+((SCMV4(3)-SCMV4(2))/50)*(Mc. Tc-100. 0)
1466     ENDIF
1467
1468     IF(Mc. Tc. GE. 150. 0. AND. Mc. Tc. LT. 200. 0) THEN
1469         NU = SCMV4(3)+((SCMV4(4)-SCMV4(3))/50)*(Mc. Tc-150. 0)
1470     ENDIF
1471
1472     IF(Mc. Tc. GE. 200. 0. AND. Mc. Tc. LT. 500. 0) THEN
1473         NU = SCMV4(4)
1474     ENDIF
1475
1476     IF(Mc. Tc. GE. 500. 0. AND. Mc. Tc. LT. 550. 0) THEN
1477         NU = SCMV4(4)+((SCMV4(3)-SCMV4(4))/50)*(Mc. Tc-500. 0)
1478     ENDIF
1479
1480     IF(Mc. Tc. GE. 550. 0. AND. Mc. Tc. LT. 600. 0) THEN
1481         NU = SCMV4(3)+((SCMV4(2)-SCMV4(3))/50)*(Mc. Tc-550. 0)
1482     ENDIF
1483
1484     IF(Mc. Tc. GE. 600. 0) THEN
1485         NU = SCMV4(2)
1486     ENDIF
1487
1488     GO TO 999
1489
1490 C-----
1491 C                                     MOD9CR-1MO
1492 C-----
1493
1494 2 IF(Mc. Tc. LT. 375. 0) THEN
1495     NU = CRMO(1)
1496     ENDIF
1497
1498     IF(Mc. Tc. GE. 375. 0. AND. Mc. Tc. LT. 450. 0) THEN
1499         NU = CRMO(1)
1500     ENDIF
1501
1502     IF(Mc. Tc. GE. 450. 0. AND. Mc. Tc. LT. 475. 0) THEN
1503         NU = CRMO(1)+((CRMO(2)-CRMO(1))/25)*(Mc. Tc-450. 0)
1504     ENDIF
1505
1506     IF(Mc. Tc. GE. 475. 0. AND. Mc. Tc. LT. 500. 0) THEN
1507         NU = CRMO(2)+((CRMO(3)-CRMO(2))/25)*(Mc. Tc-475. 0)
1508     ENDIF
1509
1510     IF(Mc. Tc. GE. 500. 0. AND. Mc. Tc. LT. 525. 0) THEN
1511         NU = CRMO(3)+((CRMO(4)-CRMO(3))/25)*(Mc. Tc-500. 0)
1512     ENDIF

```

```
1512
151 PNC TN9410 95-212 0. AND. Mc. Tc. LT. 550. 0) THEN
1515     NU = CRMO(4)+((CRMO(5)-CRMO(4))/25)*(Mc. Tc-525. 0)
1516     ENDIF
1517
1518     IF(Mc. Tc. GE. 550. 0. AND. Mc. Tc. LT. 575. 0) THEN
1519         NU = CRMO(5)+((CRMO(6)-CRMO(5))/25)*(Mc. Tc-550. 0)
1520     ENDIF
1521
1522     IF(Mc. Tc. GE. 575. 0. AND. Mc. Tc. LT. 600. 0) THEN
1523         NU = CRMO(6)+((CRMO(7)-CRMO(6))/25)*(Mc. Tc-575. 0)
1524     ENDIF
1525
1526     IF(Mc. Tc. GE. 600. 0) THEN
1527         NU = CRMO(7)
1528     ENDIF
1529 C
1530 999 Calcnu = NU
1531 C
1532     RETURN
1533     END
```