

簡易な小型電算機による林 道設計システムの開発(Ⅱ)

林業機械の振動・騒音防止—機械の改良

ページ	行	誤	正
45	上 2	山岳林には	山岳林では
45	"	小型可搬式機械が	小型可搬式機械の
48	下 1	付近では	付近で
49	下 9	騒音レベルは	騒音レベルを
53	下 2	鋸断時の最大	鋸断時の
62	上 6	燃料時間	燃焼時間
74	上 16	図-30	図-31
75	下 6	図-29	図-30
79	上 3	図-30	図-31
"	" 4	図-31	図-33
"	" 6	図-30	図-31
"	" 12	図-33	図-34
"	" 14	図-34	図-35
"	下 14	図-35	図-36
"	" 11	図-36	図-37
"	" 7	図-37	図-38
85	上 1	図-38	図-39
"	" 2	図-39	図-40
"	" 5	図-40、41	図-41、42
86	" 2	図-42	図-43
87	" 7	図-43	図-44
"	" 12	図-48	図-49
"	" 14	図-47	図-48
"	下 8	図-45	図-46
"	" 5	図-46	図-47
91	上 1	図-47	図-48
"	" 2	図-46	図-47
"	" 5	図-48	図-49
"	" 10	図-44	図-45
"	" 11	図-49	図-50
"	" 13	図-50	図-51
"	" 14	図-49	図-50
"	" 16	図-51	図-52
"	下 13	図-48	図-49
"	" 12	静バネ定数	硬さ
"	" 12	図-51	図-52
"	" 11	図-52	図-53
"	" 10	静バネ定数	ゴムの硬さ
"	" 9	図-53	図-54
95	上 7	Kojro	Kojiro
"	" 7	PRotary	Rotary

広葉樹用材林の育成技術

ページ	行	誤	正
99	上 10	試験担当者の項 林業試験場東北支 場 経営第1研究 室のあとに	造林第2研究室 桜井 尚武 大住 克博 を挿入

蓄積経理システムの開発

ページ	行	誤	正
153	上 1	地位級Ⅲ等地0,60 以上であった。	この前に「間伐施業後 の材積生長を考慮した 間伐残存林の適正収量 比数は地位級Ⅰ、Ⅱ等 地0,50以上」を挿入

簡易な小型電算機による林道設計システムの開発(Ⅱ)

ページ	行	誤	正
157	上 12	簡単	簡単
164	" 9	水平行	水平高
166	" 4	Ccostruct	Construct

簡易な小型電算機による林道設計システムの開発(Ⅱ)

I 試験担当者

機械化部林道研究室

福田 章 史

”

市 原 恒 一

II 試験目的

この研究の目的は、パーソナルコンピュータを用いて林道の自動設計を行うものであり、昭和58年度特別会計試験成績報告書「簡易な小型電算機を用いた林道設計システムの開発」に引続き横断図の設計のプログラムの開発を行おうとするものである。

近年パーソナルコンピュータが著しく進歩し、かつ林道関係の事業体にも多く導入されてきているため、その有効な利用をはかり効率的に林道設計が行えるシステムを開発することが望まれている。林道設計においては自然環境を考慮してきめの細かい設計を行わなければならない。そのためには設計と修正を繰り返し多くの図面を描く必要がある。それがコンピュータの画面上で簡単に出来るようにすることが必要である。

大型コンピュータを用いた林道の自動設計はすでに実用化されているが、作設現場の実状に合わせて設計するためには、林道設計を行う技術者が自らコンピュータを使って設計を行うことが望ましい。最近のパーソナルコンピュータの普及によりその条件が整ってきているといえるが、コンピュータの専門家ではない林道技術者でも容易に使える林道設計のためのプログラムが必要である。そのプログラムは、コンピュータについての専門的知識は最小限であっても十分使えるものであると共に、林道技術者の林道設計に対する専門的知識や経験が生かせるものでなくてはならない。そのためパーソナルコンピュータの特質を生かし、設計の途中経過をその都度コンピュータの画面に表示し、設計者の意図に応じてコンピュータに指示を与えていく対話方式が望ましいと言えよう。このことは同時にプログラム化しにくい人間的要素を設計に反映しやすくし、プログラム容量の減少にもつながり、パーソナルコンピュータを用いる上で好都合である。しかし、そのために設計の自動化がある程度犠牲になり、設計者の判断にまかされる部分が多くなる。例えばよう壁の設計において、どの種類のよう壁を選ぶかはコンピュータが決定するのでなく、設計者がその都度選択することが必要となる。

このような点を考慮し、林道技術者が容易に使い、設計の自由度も大きいプログラムの開発を行った。

III 試験の経過と得られた成果

1. 試験概要

本研究以前の既往の研究成果は前報で述べた。また前報では平面測量と縦断測量の図化について研究を行ない、特に縦断線形の決定について、パーソナルコンピュータでも行なえる方法について述べた。しかし横断設計の図化についてはまだプログラムが完成していなかった。さらに前報ではパーソナルコンピュータに SORD M23 mark IV を用いていたが、本報のプログラムの開発には多くの事業所に導入されている機種である NEC PC-9800 シリーズを用い、前報の平面図、縦断面図のプログラムも PC-9801 に移植した。

しかしプログラム開発にあたっての考え方は前報と同様で以下に述べるとおりである。

- (1) 現場の林道技術者がコンピュータの知識があまりなくても容易に利用できるものであること。
- (2) 重要な判断は、主として設計者が行ない、コンピュータの負担を軽くするとともに、設計者の意図、知識、経験が十分設計に生かせるものとする。
- (3) 単純な計算、図化はコンピュータに行なわせ、グラフィックディスプレイ等を用いて、条件を変えても何回でも修正が簡単に行なえること。
- (4) 設計された図面は、できうるかぎり「林道規程」やその他の基準・要領に準ずるものであること。

1) コンピュータの機種および言語

本研究で用いたパーソナルコンピュータを中心とする機器の構成を図-1に示す。またその仕様を表-1に示す。

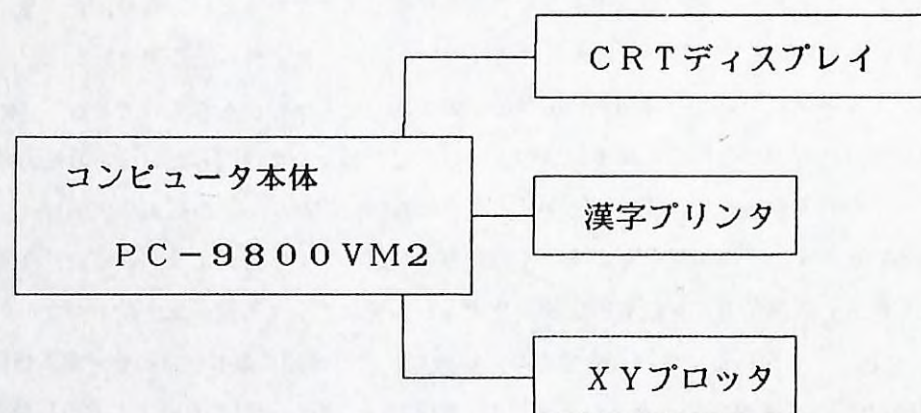


図1 林道設計システムの構成

表-1 林道設計システムの仕様

本体	NEC PC-9801 VM2
CPU	μ PD70116-10L
ユーザメモリ	384KB
増設メモリ	256KB
CRTディスプレイ	NEC PC-KD851
テキスト	80×25字
グラフ	640×400ドット
プリンタ	NEC PC-PR101L
XYプロッタ	Rolland DG DXY-980
X軸	380mm
Y軸	270mm

XYプロッタを除く他の機器は通常のパーソナルコンピュータのシステムである。プリンタは、漢字プリンタの80桁以上のものであれば、どの機種でもよい。

使用言語はBASICである。この言語はパーソナルコンピュータではごく一般的で他の機種にも移植が容易であり、理解しやすいと考える。ただし、使用したBASIC言語はMS-DOS版のN88-BASIC(86) ver 2.00である。

2) 平面、縦断面の設計

これらについては、前報で開発したプログラムをPC-9801に移植した。その使用方法等は前報と同様である。その概略を述べると、平面の設計は平面測量の結果をキーボードから入力しIPの位置や図化する範囲をグラフィック画面で確認した後XYプロッタで図化を行う。同時に曲線表がプリンタに出力される。測量データ等は縦断、横断の設計の際に利用するため、データファイルとしてフロッピーディスクに記憶される。

縦断面図はやはりキーボードより測量結果を入力しグラフィック画面に地形線を表示して、それを見ながら簡単な指示を与えて、計画勾配線を決定していく。一応の計画勾配線が決定したところで、XYプロッタに縦断面図を出力し、横断面の設計に進む。横断面の設計で問題があれば、

縦断面図の設計に戻りこれらを繰り返す。計画線の決定は設計者が行なうことを原則とし、補助的に用いる最適計画線決定のためプログラムも開発した。

3) 横断面の設計

横断面の設計が本研究の中心である。このプログラムは、横断測量の結果を入力し、その後、路面、法面、よう壁、よう壁の基礎等を設計する。これらの構造物はあらかじめデータファイルを作成しておき、必要な構造物の種類を選ぶ方式で設計していく。これらはすべてグラフィック画面上で行なうことができ、最終的には各測点の横断面図をXYプロッタに出力する。

2. 試験経過

1) 昭和59年度

横断面の設計にあたり、県や国有林の林道で用いられている各種構造物の仕様を検討し、横断面設計の資料を得た、それにもとづき各種構造物のデータファイルを作成した。

2) 昭和60年度

新たにPC-9801 コンピュータを購入し、いままでのプログラムをこれに移植するとともに、横断面設計のプログラムの開発を行った。また実際の林道測量のデータを用い横断面の設計を行いプログラムの適否の検討を行った。

3. 試験結果

開発した横断面設計プログラムは、図-2に示す構成となっておりメインプログラムと4つのサブプログラムからなっている。各プログラムはひとつひとつ独立した役割を持っているが、全体としてメインプログラムのメニューから各プログラムが呼び出して実行する。その実行の順序はどのようにでも選べるが、あるプログラムで作成されたデータが他のプログラムで使用されることもあるので、一般的には図-3に示す手順で実行される。すなわち、まず初めにこの設計で使用する構造物のファイルを作成する。ついで、横断測量データの入力を行う。各測点ごとにグラフィック表示される地形線と、縦断面図の設計で決定された路盤高を見ながら構造物の設計を行う。最後に設計された横断面図をXYプロッタに出力する。これらを各測点ごとに繰り返していく。

これらのプログラムについて、以下順に説明する。

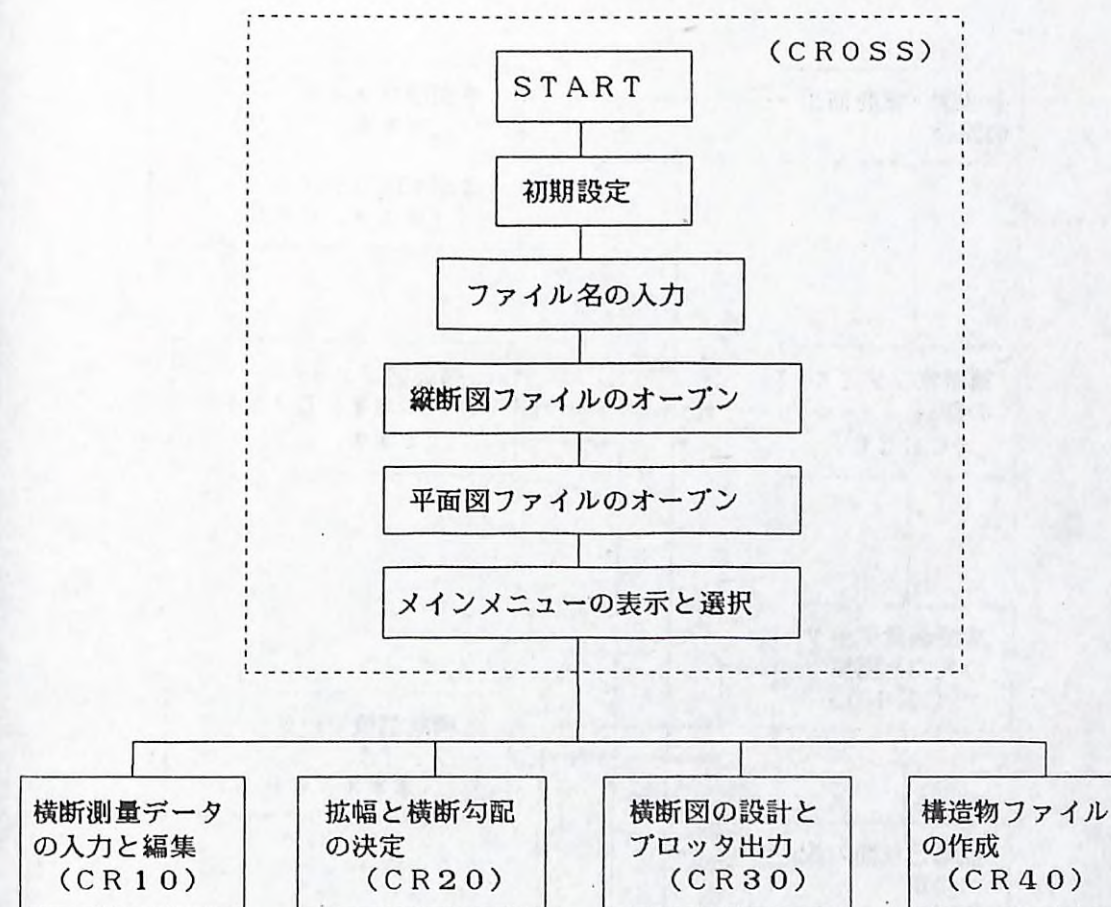


図-2 プログラムの構成

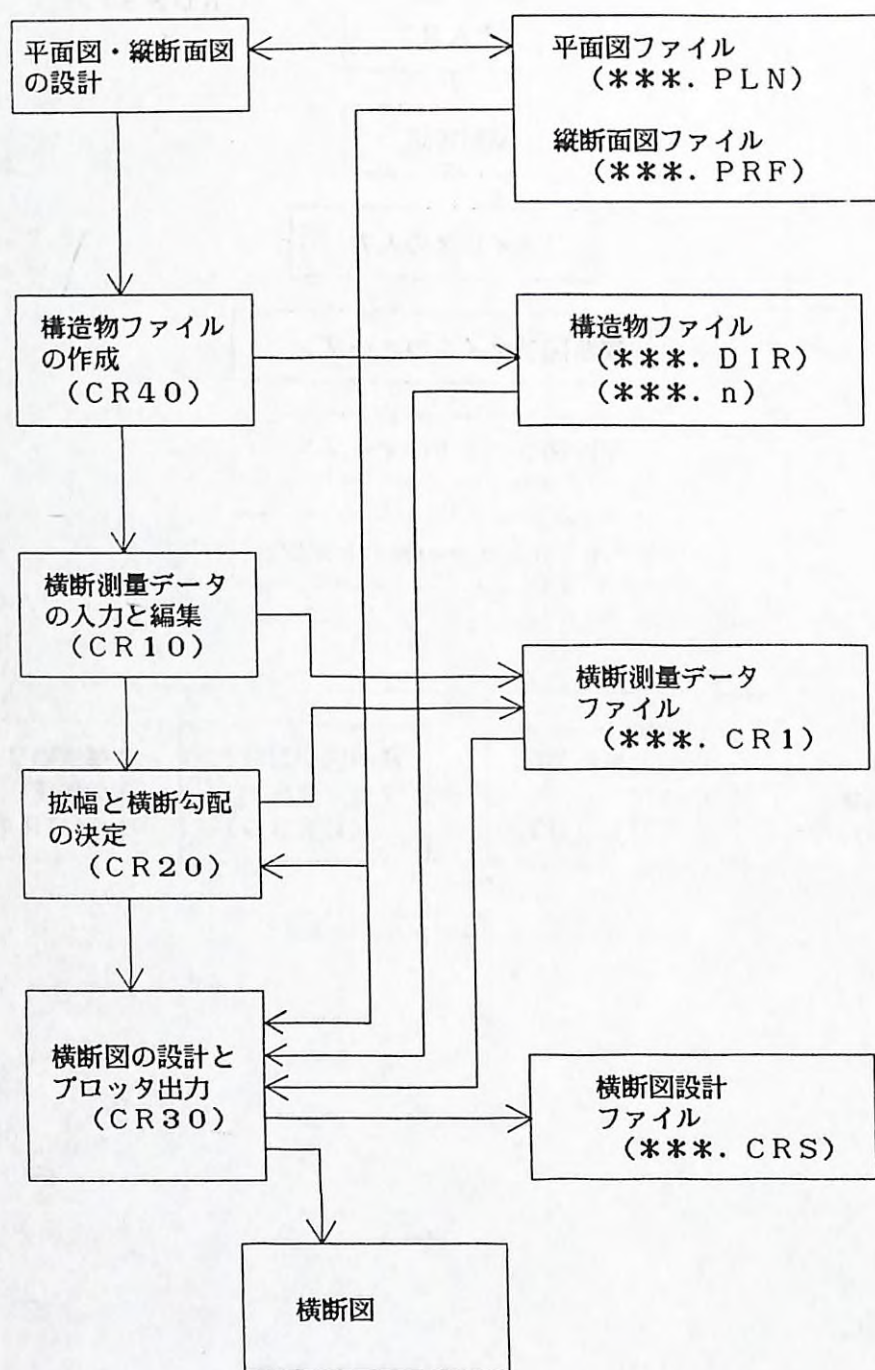


図-3. 作業の手順と作成・使用されるファイル

1) 横断面設計のメインプログラム

図-2で"CROSS"と名付けられたプログラムは、横断面設計のメインプログラムで、初期条件の設定、各サブプログラムへの分岐を行なう。横断面の設計を行うときはまず初めにこのプログラムを起動する。起動にあたっては、MS-DOSからN88-BASIC(86)を"/S:60/F6"のパラメータ(バッファサイズ60バイト、ファイル数6)をつけて立上げる必要がある。また設計路線の平面測量と縦断測量のデータファイルがいずれかのディスクドライブになければならない。プログラムはまずこれらのファイルをオープンして、各サブプログラムに分岐するためのメニューを表示するので、これらを任意に選択する。横断測量データの入力と編集(CR10)以外のプログラムは、CHAIN命令でメインプログラムとおきかわる。その際、総ての変数とオープンしたファイルはそのまま新しいプログラムに引き継がれる。

2) 横断測量データの入力と編集

このサブプログラム(CR10)は、通常まず初めに実行される。しかしすでにデータが入力済みの場合はその必要はない。このプログラムは横断測量のデータファイルを作成したり、修正を行ったりするもので、最終的にその路線に共通したファイル名に".CRI"の拡張子のついたファイルが生成され、それが後に述べる他のプログラムで使用される。

Command Input, Change, Before, Next, Print, Quit

タイトル: 有馬線林道 NO.2
レコード No. 26 全レコード数 30

測点 IP 76									
左側					右側				
-0.4	-1.4	-0.5	-1.4	-1.5	1.7	1.6	0.6	1.5	1.3
-0.3	1.8	0.0	2.0	2.0	2.0	1.0	1.0	0.3	2.0
-1.4	-1.4	-1.4	-0.2	-0.1	1.4	0.4	0.4	0.6	3.2
2.0	2.0	2.0	0.0	0.4	1.4	0.0	0.4	0.0	5.3
			-1.4	-1.4	1.0	1.6	1.8	2.0	1.0
			2.0	2.0	0.0	2.0	2.0	1.0	1.0
					2.0	2.5			
					0.4	1.0			

図-4 横断測量データの入力

メインメニューでこのプログラムが選択されると画面に初めの測点あるいは、いままでに入力した最後の測点のデータがボール横断の野帳と同じ形式で表示される(図-4)最上段にはその時点で選べるコマンドが表示される(コマンドラインという)。そのコマンドラインの英字の頭文字を入力して、各コマンドを実行する。

コマンドのInputはデータの入力、Changeはデータの修正、BeforeとNextは表示されている測点を前あるいは次の測点に変更する。Printは入力されたすべてのデータをプリンタに印刷する。Quitはデータ編集モードから抜けだし、メインプログラムへ戻る。

Inputを選んでデータ入力モードとした場合は、先に述べた野帳と同じ形式の表に右側から垂直高、水平行を順次入力していく。このモードで入力できるコマンドは、数字あるいはカーソル移動キーと次に述べるコマンドの頭文字である。Insertは現在カーソルのあるところに新しいデータを挿入し、Deleteはカーソルの箇所のデータを消去し残りのデータを詰める。Sameは一つ前の測点と全部同じデータを用いる時に入力する。一つの測点のデータの入力が終わったならばEndを入力し、元のデータ編集メニューに戻る。

カーソル移動キーは画面のカーソルを任意に移動しデータの入力位置を決めるために用いるが、連続してデータを入力する場合はカーソルは自動的に次の入力位置に移動する。

入力したデータはランダムアクセスファイルとして、一測点の入力ごとにフロッピーディスクに記録される。データの入りはどの時点でも中止、再開ができる。

3) 拡幅と横断勾配の決定

このサブプログラム(CR 20)は各測点の拡幅量と横断勾配の計算と待避所の設計を行うもので、次節の構造物の設計を行う前に必ず実行しなければならない。

このプログラムは「林道規程」にもとづいた拡幅量と横断勾配の設定を行なうが、プログラム中の設定値を変えて必要に応じた設定とすることも可能であるが、次のような制約がある。

- (1) 緩和区間長は全線で一定で、その値はあらかじめ入力する。緩和区間の一部を曲線内に設定してもよいが、その距離が円曲線のMCを越してはならない。
- (2) 拡幅の位置は各IPごとに、内側、外側、両側のいずれかに設定できる。拡幅量は「林道規程」の値を用いるか、各曲線ごとに入力するかのをいずれかを選択することができる。
- (3) 直線区間の横断勾配は、レベルとするか片勾配とするかを始めに選択できる。ただし屋根形はない。屋根形とする場合は土工定規にその勾配を定めておいて、横断図にはレベルで現わすことにする。曲線区間の横断勾配は設計速度に応じた値とするか(「林道規程」解説とその運用、P. 84)、各IPごとに入力する。
- (4) 緩和線形は、緩和区間の始点と終点を直線で結んだ緩和接線である。
- (5) 横断勾配のすりつけは、拡幅の緩和区間内で行なう。ただし複合曲線のときに、所定のすり

つけ率(「林道規程」解説とその運用、P. 99)より小さくなる場合にはすりつけ率より計算した区間長とする。

(6) 合成勾配のチェックはプログラム中ではしていない。

(7) 視距については考慮していない。

このプログラムの実行は、画面に表示される入力要求に従って、始めに設計速度・林道の等級などの設定を行い、引き続いて各IPごとに必要な値を入力していく。

拡幅と横断勾配の設定を終えると、待避所の設計を行なう。待避所は設計する個数を入力した後、入力要求に従いそれぞれの待避所のついてその位置を累計距離で、また有効長、すりつけ区間長、拡幅量を順に入力する。

以上の設計が終わると各測点のデータが印刷され、メインメニューに戻る。

4) 構造物と法面の設計

このサブプログラム(CR 30)はグラフィック画面上で構造物と法面の設計を行なう。メインメニューからこのプログラムにはいると画面に地形線と路面が表示され、コマンドラインにはConstruct, Plotta, Zoom, Scroll, Before, Next, Quitと表示される(図-5)。Constructで構造物の設計モードになり、Plottaは設計された図面を

Command Construct, Plotta, Zoom, Scroll, Initial, Before, Next, Quit N

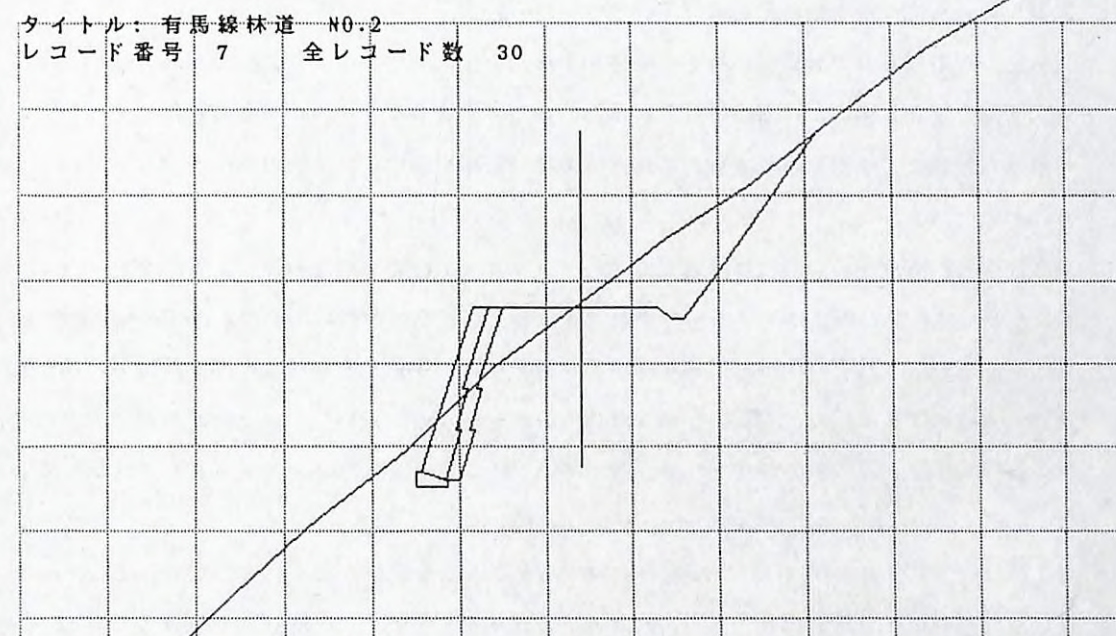


図-5 横断図設計中の画面

ブロックに出力する。Zoom, Scrolle, Initialは画面の表示を拡大・縮小, 移動, 初期状態への復帰を行なう。この場合画面に表示されているメッシュは拡大・縮小をしても常に2m間隔である。その他のコマンドは3-2節と同様である。

Ccostructで構造物の設計モードを選んだ場合は, 画面最上段に表示される入力要求に従って必要な入力を行ないながら設計を進めていく。左側の構造物から設計を始め, もよりの構造物の端点に次の構造物をつなげていく方式で設計を行う。始めに路面の端に側溝を設計し次いで法面, よう壁, 基礎, 第2の法面, よう壁, 基礎の順に設計をしていく。

これらの法面や構造物を設計の要素という。これらの要素は片側について側溝が1箇所, 法面, よう壁, 基礎については各々2箇所が設計できる。これらの要素の設計の順番も定められており, 必ず中心線側から側溝-法面-よう壁-基礎-法面-よう壁-基礎の順で行う。しかしこれらの内, 必要でない要素は設計をしないで次につなげてよく, この方法ではほとんどの林道の設計が可能である。

具体的な設計の方法は, 法面については, もよりの構造物の終点を法面の始点(法肩, 法尻)として最大14までの折線で設計する。犬走りなども法面の設計に含めて行う。各々の法面は始めに法勾配を割の単位で入力し(レベルの場合はL), 次いで法高(Height), 法の水平距離(Xlength), 地表からの垂直距離(Remain height)の何れかを選び数値を入力すると, 画面に設計した法面が表示されるのでそれでよければ次の法面の設計に移る。Exitで法面の設計を終わり次の構造物の設計に移る。

以上の設計のなかで地表面からある距離まで法面を設計するコマンドRemain heightは0を入力すると地表と接した法面が設計され, 他の数値を入力すると地表から垂直距離でその数値だけ離れた位置まで法面を設計する。これは例えば岩石線を想定して法面の勾配を変化させたいときなどに用いる。

法面以外の構造物の設計には構造物のデータファイルがなければならない。このデータファイルについては次節の構造物ファイルの作成で述べる。構造物の設計の場合は, 画面に構造物の種類の名前が表示されその中から適当な構造物を選択する。選択した構造物の断面が画面に描かれそれでよければ次に進み, 不適当な場合は再度設計を行なう。選択した構造物が最適であるかどうかの判断は設計者にまかされており, 例えばよう壁の安定計算等はプログラム中で行なっていない。

これらの作業を各測点について左側と右側で行ない設計を完成させる。ここで設計された結果は, その路線の各設計データファイルに共通したファイル名に".CRS"の拡張子をつけたファイルとして記録される。

コマンドでPlottaを選ぶとその測点の設計結果がプロッタに描かれる。縮尺は1:100に

固定されており, XYプロッタの作図範囲はA3判であるので, 一つの図面に平均して4~6個の測点の横断面図が描ける。各測点の作図位置は, グラフィック画面に表示されるXYプロッタの表示範囲を示す枠を移動させて決定する。

最後にQuitを入力すれば, メインメニューに戻る。

5) 構造物ファイルの作成

その路線で用いる構造物はあらかじめこの構造物ファイルに登録しておく必要がある。構造物ファイルは図-6に示す構造となっている。側溝とよう壁と基礎の三個のファイルに分けられ, 各々は"WALL.DIR"のように".DIR"拡張子のつけられたディレクトリファイルで管理

ディレクトリファイル

データファイル

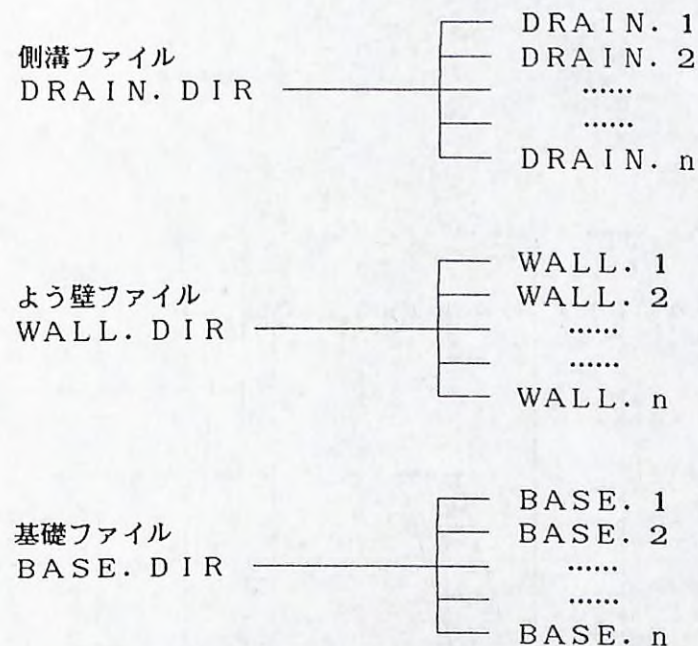


図-6 構造物ファイル

されている。このディレクトリファイルは構造物のタイプの個数と, その個数分のタイプ番号, 名前が入力されている。このタイプ番号と名前が先の構造物の設計時に画面に表示され選択される。

またタイプ番号を拡張子とした, 例えば"WALL.12"といったファイルに各構造物のデータを入力されている。その内容は, 表-2に例を示すようにその構造物の横断面図をXY座標で現

わす。座標の原点は他の構造物あるいは法面に接する点にとり、そこから順次点を結んで構造物の断面を描く。前の点との間を直線で結ぶ点には、継点コードとして1をおき、結ばない場合は断点コードに0をおく。表-2のデータファイルの例では図-7に示す断面が描かれる。

コード	0	1	2	3	4	5	6	7	8	9	10	11	12
X座標	12*	0.9	1.26	1.57	0.96	1.12	0.78	0.5	0.0	0.5	0.85	0.7	1.2
Y座標	3**	3.0	3.0	3.0	1.0	1.0	-0.15	-0.15	0.0	-0.15	1.0	1.0	3.0
継点	-	1	1	1	1	1	1	1	1	0	1	1	1

*全レコード数, **終点レコードNo.

表-2 構造物データファイルのデータ構造の一例

Command Zoom, Scroll, Initial, Exit?

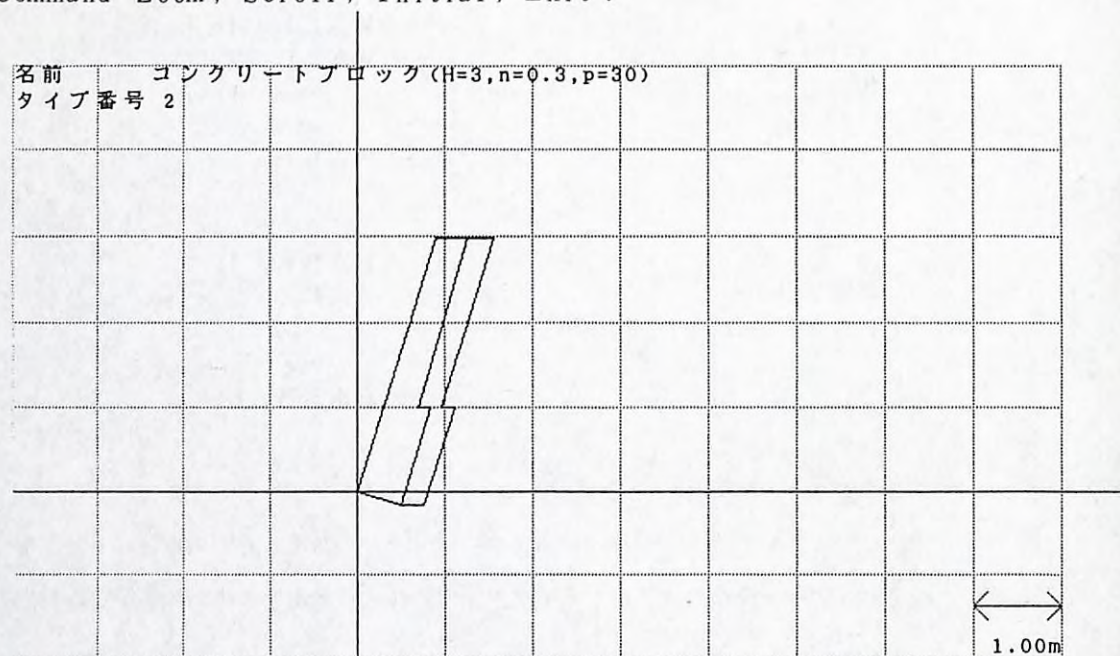


図-7 構造物設計中の画面

各構造物ファイルのデータには前の構造物と接する始点と次の構造物と接する終点が定義されている。始点は常に構造物ファイルの原点 ($X=0, Y=0$) の点であり、終点は各データファイルの2番目のレコードにそのレコード番号が入れられる。なお1番目のレコードには全レコード数が入れている (表-2)。

このサブプログラム (CR 40) を起動すると、コマンドラインに設計する構造物の種類を選ぶコマンドが表示され、そのいずれかを選択する。いずれの場合も画面にメッシュが表示され、データ入力モードとなる。横断設計のプログラムと同様に Zoom, Scroll, Initial のコマンドで画面の表示を変えることができる。構造物の各点のX座標, Y座標を入力し、その点を前の点と結ぶか否かを入力すれば構造物の断面が描かれる。最後に次の構造物との接点になる点の番号を入力すると構造物ファイルが完成する。前出の図-7はその一例である。

構造物は路線の右側にあるものとして図を描く。そのため、構造物によっては始点と終点が設計により入れ変わることもあり、それは横断図の設計の時点で決定する。またよう壁の場合に例えば路側よう壁の天端が幅員に含まれるか否かにより始点の位置を変えなければならない。

横断図の設計の時には、この構造物ファイルのデータがそのまま表示されるので、適当な省略を行うことによって作業の能率があげられる。またよう壁は高さの違うすべてのよう壁の種類をファイルに作成しておかなくてはならない。しかしこの構造物ファイルは一つの路線だけに用いるのでなく、一度作成すれば、他の路線の設計にもそのまま用いることができるので、始めの入力には労力を要するが、一度入力すれば後は何度でも用いることができる。

構造物ファイルは、シーケンシャルファイルでありMS-DOSのTYPEコマンドでその内容を見ることができ、EDLIN等のエディタによってその内容が修正できる。あらかじめ一般に用いられる構造物のファイルを作成しておき、これらのエディタにより必要なファイルを編集すると能率的である。

6) 試験結果

以上のプログラムを用いて、いくつかの実際に測量された路線のデータを入力し設計を行なった。XYプロッタに出力した結果の例を図-8に示す。設計に要した時間は測点数が50点で、約200mの路線でデータの入力と点検・修正に約2時間、横断面の設計に約3時間、プロッタの作図に約1時間を要した。

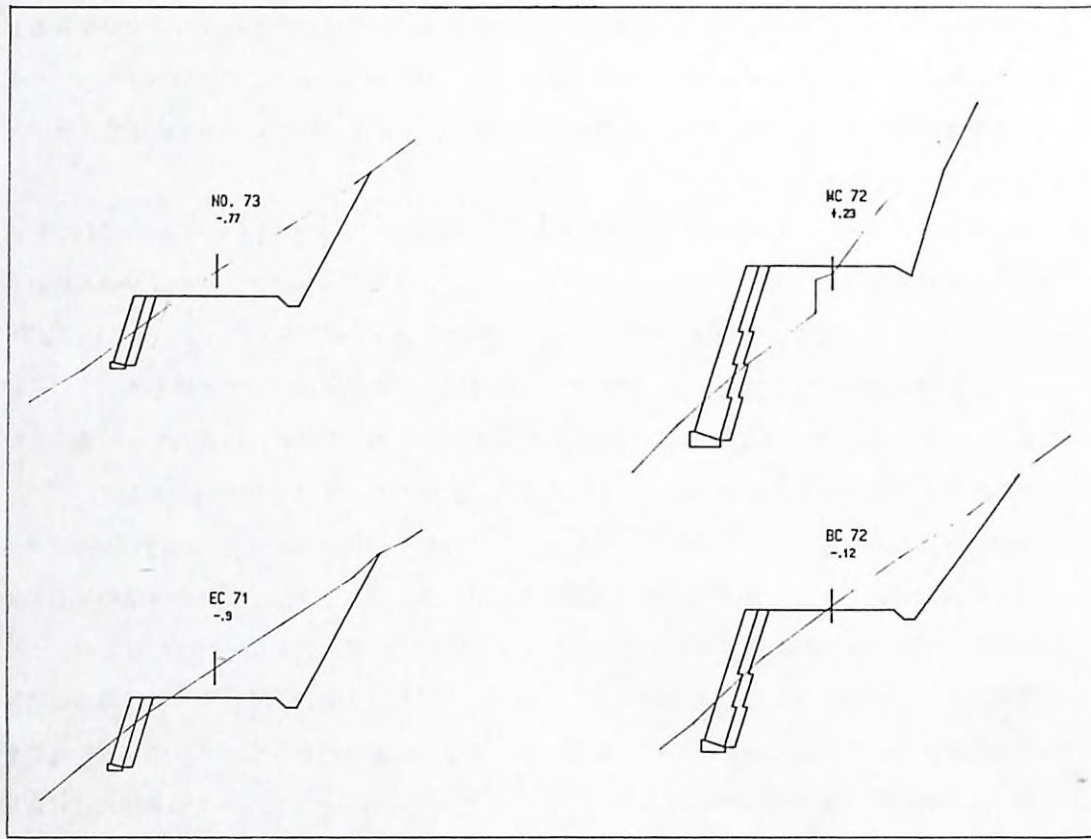


図-8 XYプロッタにより図化した横断面図（原寸はA3判）

4. 今後の課題

以上述べたように、前報とあわせて林道の設計をパーソナルコンピュータを用いて行なうプログラムは完成したが、現在のところ図化を行なうだけの段階である。今後に残された課題としてはつぎのような点があげられる。

- (1) 土工量の計算と、各種数量計算の自動化。
- (2) 数値地形データと結び付けてさらに設計の自動化をはかる。
- (3) 完成したプログラムの改良。

引用文献

- (1) 福田章史・井上源基・市原恒一：簡易な小型電算機による林道システムの開発，昭和58年度国有林野事業特別会計技術開発試験成績報告書，林業試験場，1984．
- (2) 林道技術問題検討委員会：林道規程一解説とその運用，日本林道協会，1983．

付 録 横断設計プログラムリスト

付 録 横断設計プログラムリスト

```

10 *****
20 ( FOREST ROAD DROWING SYSTEM )
30 FILE NAME="CROSS"
40 CROSS PROFILE DROWING MAIN-PROGRAM
50 CREATED BY A.FUKUDA
60 *****
70 ON ERROR GOTO 0
80 CONSOLE 0,24,0,0 : SCREEN 3 : CLS 3
90 LOCATE 15,10 : PRINT "***** F O R D   S Y S T E M *****"
100 LOCATE 8,12 : PRINT "このプログラムは横断図の設計をおこないます。"
110 LOCATE 0,0 : PRINT "Push any key !!"
120 WHILE INKEY$="" : WEND :CLS
130 DEFINT I, J, P, R
140 DIM C(1,45), C$(14), H$(12), H(12)
150 DIM E$(3), E(3), D1$(12), D1(12)
160 DIM WIDEN(1,1), TWIDEN(2,20), CRSA(1), CURV(2)
170 DIM TURN(3,5), ECC(2), ETT(2), BTT(2), BCC(2), MC(2), RAD(2)
180 DIM CONWIDA(1,1), CONWIDB(1,1), CONCRSA(1), CONCRSB(1)
190 DIM DT$(50), WT$(50), BT$(50), KX(50), KY(50), KC(50), X(46), Y(46)
200 DIM K$(14), GX(3), GY(3), DX(2,2), DY(2,2)
210 DIM WLX1(2,2), WLY1(2,2), WLX2(2,2), WLY2(2,2)
220 DIM BX1(2,2), BY1(2,2), BX2(2,2), BY2(2,2)
230 DIM SLX1(2,14), SLY1(2,14), SLX2(2,14), SLY2(2,14)
240 *****
250 'ファイル名の入力
260 *****
270 *FILENAME
280 INPUT "ファイル名 ";FILES$
290 F1$ = FILES$+".CRS"
300 F2$ = FILES$+".CRI"
310 F3$ = FILES$+".PRF"
320 F4$ = FILES$+".NUM"
330 F5$ = FILES$+".PLN"
340 *****
350 'タイトルの入力
360 *****
370 ON ERROR GOTO *ERORUTIN
380 OPEN F4$ FOR INPUT AS #1
390 ON ERROR GOTO 0
400 INPUT #1, T$, DEND%, DUMY1%, PEND, DUMY2%, WIDE2
410 CLOSE #1
420 *****
430 'P R F ファイルのオープン
440 *****
450 OPEN F3$ AS #3
460 *****
470 'P L N ファイルのオープン
480 *****
490 OPEN F5$ AS #6
500 *****
510 '          メインプログラム
520 *****
1000 *MAIN : '***** サブプログラムからのエントリーポイント
1010 DEFINT I, J, P, R
1020 CLS
1030 PRINT "          <<< メインメニュー >>>":PRINT
1040 PRINT "  1) 横断測量データの入力と編集"
1050 PRINT "  2) 拡幅と横断勾配の決定"
1060 PRINT "  3) 構造物と法面の設計"
1070 PRINT "  4) 構造物ファイルの作成"
1080 PRINT "  5) 終了"
1090 PRINT
1100 INPUT "          選んでください "; Q

```

```

1110 IF Q < 1 OR Q > 5 THEN GOTO 1030
1120 ON Q GOTO 2030, 1130, 1140, 1150, 1160
1130 CHAIN "CR20", 1000, ALL
1140 CHAIN "CR30", 1000, ALL
1150 CHAIN "CR40", 1000, ALL
1160 END
2000 '*****
2010 ' サブプログラム NO.1 "CR10"
2020 '*****
2030 CLS
2040 PRINT " <<< 横断測量データの入力と編集 >>>"
2050 PRINT : PRINT
2060 '*****
2070 ' C R 1 ファイルのオープン
2080 '*****
2090 FIELD #2, 4 AS C$(0), 4 AS C$(1), 4 AS C$(2), 4 AS C$(3), 4 AS C$(4),
      4 AS C$(5), 4 AS C$(6), 4 AS C$(7), 4 AS C$(8), 4 AS C$(9),
      4 AS C$(10), 4 AS C$(11), 4 AS C$(12), 4 AS C$(13), 4 AS C$(14)
2100 OPEN F2$ AS #2
2110 PRINT "横断測量ファイル (":F2$;" ) の入力は初めてか (Y/N) ";
2120 Q$ = INKEY$ : IF Q$ = "" THEN GOTO 2120
2130 PRINT Q$
2140 IF Q$ = "N" OR Q$ = "n" THEN GOTO 2210
2150 IF Q$ = "Y" OR Q$ = "y" THEN GOTO 2170
2160 GOTO 2110
2170   REND = 1 : RNO = REND : PNO = 0
2180   C(0,0) = REND
2190   REC = 1 : GOSUB *PUTFILE1
2200   GOTO *PRFOPEN
2210 REC = 1 : GOSUB *GETFILE1
2220 REND = C(0,0)
2230 RNO = REND
2240 REC = REND : GOSUB *GETFILE1
2250 PNO = C(0,44)
2260 '*****
2270 ' P R F ファイルのフィールド
2280 '*****
2290 *PRFOPEN
2300 FIELD #3, 8 AS H$, 4 AS H$(0), 4 AS H$(1), 4 AS H$(2), 4 AS H$(3),
      4 AS H$(4), 4 AS H$(5), 4 AS H$(6), 4 AS H$(7), 4 AS H$(8),
      4 AS H$(9), 4 AS H$(10), 4 AS H$(11), 4 AS H$(12)
2310 IF PNO > 1 THEN REC = PNO : GOSUB *GETFILE2 : GOTO 2350
2320 REC = 1
2330 GOSUB *GETFILE2 : IF H(9) >= 6 THEN REC = REC+1 : GOTO 2330
2340 PNO = REC
2350 '*****
2360 ' データ編集メニュー
2370 '*****
2380 CLS 3 : CONSOLE 0,2 : SCREEN 2,0,0,1
2390 GOSUB *WAKU
2400 GOSUB *DISPDATA
2410 GOSUB *DISPNUM
2420 LOCATE 0,3 : PRINT "タイトル:" ; T$
2430 *MENU
2440 CLS : PRINT "Command I(nput, C(hange, B(efore, N(ext, P(rint, Q(uit ";
2450 A1$ = INKEY$ : IF A1$="" THEN GOTO 2450
2460 PRINT A1$
2470 IF A1$ = "I" OR A1$ = "i" THEN GOTO *INPDATA
2480 IF A1$ = "C" OR A1$ = "c" THEN GOTO *CHANGE
2490 IF A1$ = "B" OR A1$ = "b" THEN GOTO *BEFORE
2500 IF A1$ = "N" OR A1$ = "n" THEN GOTO *NEXT.DATA
2510 IF A1$ = "P" OR A1$ = "p" THEN GOTO *INSATU
2520 IF A1$ = "Q" OR A1$ = "q" THEN GOTO *QUIT

```

```

2530 GOTO *MENU
2540 '*****
2550 ' データの入力
2560 '*****
2570 *INPDATA
2580 REC = REND : GOSUB *GETFILE1
2590 PNO = C(0,44)
2600 *INPDATA1
2610 PNO = PNO+1
2620 IF PNO > PEND THEN CLS : PRINT "全部のデータの入力は終わりました。" : GOSU
      *SLEEP : GOTO *QUIT
2630 REC = PNO : GOSUB *GETFILE2
2640 IF H(9) >= 6 THEN GOTO *INPDATA1
2650 REND = REND+1
2660 RNO = REND
2670 C(0,0) = 0
2680 C(1,0) = 0
2690 GOSUB *DISPNUM
2700 GOSUB *INPT
2710 IF C(0,0) <> 0 AND C(1,0) <> 0 THEN GOTO 2790
2720   CLS : PRINT "データが全部 (または片側) が入力されていません。" : GOSUB *
      LEEP
2730   REND = REND-1 : RNO = REND
2740   REC = RNO : GOSUB *GETFILE1
2750   PNO = C(0,44)
2760   IF PNO > 1 THEN REC = PNO : GOSUB *GETFILE2 : GOTO *MENU
2770   PNO = 0 : REC = 1 : GOSUB *GETFILE2
2780   GOTO *MENU
2790 C(0,44) = PNO
2800 REC = RNO : GOSUB *PUTFILE1
2810 GOTO *MENU
2820 '*****
2830 ' データの変更
2840 '*****
2850 *CHANGE
2860 IF C(0,0) = 0 OR C(1,0) = 0 THEN CLS : PRINT "データが入力されていません。"
      : GOSUB *SLEEP : GOTO *MENU
2870 GOSUB *DISPNUM
2880 GOSUB *INPT
2890 REC = RNO : GOSUB *PUTFILE1
2900 GOTO *MENU
2910 '*****
2920 ' 次のデータ表示
2930 '*****
2940 *NEXT.DATA
2950 RNO = RNO+1
2960 IF RNO > REND THEN RNO = REND
2970 REC = RNO : GOSUB *GETFILE1
2980 PNO = C(0,44) : IF PNO < 1 THEN PNO = 1
2990 REC = PNO : GOSUB *GETFILE2
3000 GOSUB *CLRDATA
3010 GOSUB *DISPNUM
3020 GOSUB *DISPDATA
3030 GOTO *MENU
3040 '*****
3050 ' ひとつ前のデータ表示
3060 '*****
3070 *BEFORE
3080 RNO = RNO-1
3090 IF RNO <= 1 THEN RNO = 2
3100 GOTO 2970
3110 '*****
3120 ' プリンタに印刷

```

```

3130 '*****
3140 *INSATU
3150 IF REND = 0 THEN
    CLS : PRINT "データがありません。" : GOSUB *SLEEP : GOTO *MENU
3160 CLS : PRINT "プリンタの準備はよいか (OK=Push any key) "
3170 WHILE INKEY$="" : WEND
3180 LPRINT
3190 LPRINT TS
3200 LPRINT : LPRINT
3210 FOR J = 2 TO REND
3220 REC = J : GOSUB *GETFILE1
3230 REC = C(0,44) : GOSUB *GETFILE2
3240 LPRINT "測点 "; H1$
3250 FOR I = 1 TO 22 STEP 20
3260 IF I = 1 THEN
    I2 = C(0,0) : LPRINT "右側";
    ELSE I2 = C(1,0)+20 : LPRINT "左側";
3270 I1 = I
3280 LPRINT TAB(10);
3290 I3 = I1 : K = 0
3300 WHILE K < 10 AND I1 <= I2
3310 LPRINT USING "###.# "; C(0,I1);
3320 K = K+1 : I1 = I1+1
3330 WEND
3340 LPRINT : LPRINT TAB(10);
3350 K1 = K-1
3360 FOR K = 0 TO K1
3370 LPRINT STRING$(6,"-"); SPC(1);
3380 NEXT K
3390 LPRINT : LPRINT TAB(10);
3400 I1 = I3
3410 FOR K = 0 TO K1
3420 LPRINT USING "###.# "; C(1,I1);
3430 I1 = I1+1
3440 NEXT K
3450 LPRINT:LPRINT
3460 IF I1 > I2 THEN GOTO 3480
3470 GOTO 3280
3480 NEXT I
3490 NEXT J
3500 GOTO *MENU
3510 '*****
3520 ' 終了
3530 '*****
3540 *QUIT
3550 C(0,0) = REND
3560 REC = 1 : GOSUB *PUTFILE1
3570 SCREEN 3 : CONSOLE 0,25 : CLS 3
3580 CLOSE #2
3590 GOTO *MAIN
10000 '*****
10010 ' エラー処理ルーチン
10020 '*****
10030 *ERRORUTIN
10040 IF ERL = 380 THEN PRINT "ファイル";FILE$;"はありません。" : RESUME *FILENAM
E
10050 IF ERL = 13260 THEN PRINT "縦断面図のデータファイルがありません。" : END
10060 PRINT "ERR="; ERR, "ERL="; ERL
10070 PRINT "マニュアルを見てください"
10080 END
10090 '*****
10100 ' SUB-ROUTINE
10110 '*****

```

```

10120 '
10130 '***** SUB データ入力
10140 '
10150 *INPT
10160 GOSUB *CLRDATA
10170 GOSUB *DISPDATA
10180 N = 0
10190 I = C(0,0)+1
10200 IF I = 21 THEN I = 20
10210 GOSUB *SETXY
10220 *INPT1
10230 IF (I = C(0,0)+1 AND C(0,0) <> 20) OR I = C(1,0)+21 THEN GOTO 10440
10240 '...CHANGE DATA
10250 CLS
10260 PRINT "CHANGE MODE: 矢印(Cursor移動, I(nsert, D(elete, 123(Data入力, E(
d"
10270 *INPT2
10280 GOSUB *DISPXY
10290 LOCATE X,Y+N*2
10300 PRINT USING "###.#"; C(N,I);
10310 LOCATE X,Y+N*2
10320 A1$ = INPUT$(1)
10330 IF ASC(A1$) = &H1C THEN GOSUB *RIGHT.ARROW : GOTO *INPT1
10340 IF ASC(A1$) = &H1E THEN GOSUB *UPPER.ARROW : GOTO *INPT1
10350 IF ASC(A1$) = &H1F THEN GOSUB *LOWER.ARROW : GOTO *INPT1
10360 IF ASC(A1$) = &H1D THEN GOSUB *LEFT.ARROW : GOTO *INPT1
10370 IF A1$ = "I" OR A1$ = "i" THEN GOSUB *INSERT : GOTO *INPT1
10380 IF A1$ = "E" OR A1$ = "e" THEN GOTO *INPFIN
10390 IF A1$ = "D" OR A1$ = "d" THEN GOSUB *DEL : GOTO *INPT1
10400 GOSUB *INPSUB
10410 IF ER% THEN GOTO *INPT2
10420 C(N,I) = C0
10430 GOTO *INPT2
10440 '...INSERT DATA LAST
10450 CLS
10460 PRINT "INPUT MODE: 矢印(Cursor移動, 123(Data入力, S(ame, E(nd "
10470 LOCATE X,Y : PRINT SPC(5) : LOCATE X,Y
10480 A1$ = INPUT$(1)
10490 IF ASC(A1$) = &H1C THEN GOSUB *RIGHT.ARROW : GOTO *INPT1
10500 IF ASC(A1$) = &H1E THEN GOSUB *UPPER.ARROW : GOTO *INPT1
10510 IF ASC(A1$) = &H1F THEN GOSUB *LOWER.ARROW : GOTO *INPT1
10520 IF ASC(A1$) = &H1D THEN GOSUB *LEFT.ARROW : GOTO *INPT1
10530 IF A1$ = "S" OR A1$ = "s" THEN GOSUB *SAME : GOTO *INPT1
10540 IF A1$ = "E" OR A1$ = "e" THEN GOTO *INPFIN
10550 GOSUB *INPSUB
10560 IF ER% THEN GOTO 10470
10570 C(0,I) = C0
10580 CLS : PRINT "INPUT MODE: D(elete, 123(Data入力 "
10590 LOCATE X,Y : PRINT USING "###.#"; C(0,I); : LOCATE X,Y
10600 LOCATE X,Y+2
10610 A1$ = INPUT$(1)
10620 IF A1$ = "D" OR A1$ = "d" THEN LOCATE X,Y+2 : PRINT SPC(7); : GOTO *INPT1
10630 N = 1
10640 GOSUB *INPSUB
10650 IF ER% THEN GOTO 10590
10660 C(1,I) = C0
10670 IF I > 20 THEN C(1,0) = C(1,0)+1 ELSE C(0,0) = C(0,0)+1
10680 GOSUB *DISPXY
10690 IF I = 20 AND C(0,0) = 20 THEN CLS : PRINT "右側のデータ領域はいっぱいです
。" : GOSUB *SLEEP : I = C(1,0)+20
10700 N = 0
10710 I = I+1
10720 IF I > 40 THEN CLS : PRINT "左側のデータ領域はいっぱいです。" : I = 40 : G

```

```

OSUB *SLEEP : GOTO *INPT1
10730 GOSUB *SETXY
10740 GOTO *INPT1
10750 *INPFIN
10760 LOCATE X,Y+N*2
10770 IF I = C(0,0)+1 OR I = C(1,0)+21 THEN PRINT SPC(7); ELSE PRINT USING "###
#"; C(N,I);
10780 RETURN
10790 '
10800 '**** SUB データ表示位置のクリアー
10810 '
10820 *CLRDATA
10830 RESTORE *RDATA
10840 FOR I = 1 TO 40
10850   READ X, Y
10860   LOCATE X,Y : PRINT SPACES(7);
10870   LOCATE X,Y+2 : PRINT SPACES(7);
10880 NEXT I
10890 RETURN
10900 '
10910 '*** SUB 枠の表示
10920 '
10930 *WAKU
10940 LOCATE 15,7 : PRINT "左側"
10950 LOCATE 55,7 : PRINT "右側"
10960 LINE (316,128)-(316,368)
10970 FOR Y = 152 TO 352 STEP 64
10980   FOR X = 0 TO 256 STEP 64
10990     LINE (X,Y)-(X+40,Y)
11000   NEXT X
11010   FOR X = 328 TO 584 STEP 64
11020     LINE (X,Y)-(X+40,Y)
11030   NEXT X
11040 NEXT Y
11050 RETURN
11060 '
11070 '**** SUB データの表示
11080 '
11090 *DISPDATA
11100 RESTORE *RDATA
11110 FOR I = 1 TO C(0,0)
11120   READ X, Y
11130   LOCATE X,Y : PRINT USING "###.#" ; C(0,I)
11140   LOCATE X,Y+2 : PRINT USING "###.#" ; C(1,I)
11150 NEXT I
11160 RESTORE *LDATA
11170 FOR I = 21 TO C(1,0)+20
11180   READ X, Y
11190   LOCATE X,Y : PRINT USING "###.#" ; C(0,I)
11200   LOCATE X,Y+2 : PRINT USING "###.#" ; C(1,I)
11210 NEXT I
11220 RETURN
11230 '
11240 '*** SUB 表示位置の設定
11250 '
11260 *SETXY
11270 RESTORE *RDATA
11280 FOR J = 1 TO I
11290   READ X, Y
11300 NEXT J
11310 LOCATE X,Y
11320 RETURN
11330 '

```

```

11340 '**** SUB カーソルの移動
11350 '
11360 *RIGHT.ARROW
11370 GOSUB *DISPXY
11380 IF I > 20 THEN GOTO 11430
11390   I = I+1
11400   IF I = C(0,0)+1 THEN N = 0 : GOTO 11450
11410   IF I > C(0,0)+1 THEN I = 21
11420   GOTO 11450
11430 I = I-1
11440 IF I = 20 THEN I = 1
11450 GOSUB *SETXY
11460 RETURN
11470 '
11480 *UPPER.ARROW
11490 GOSUB *DISPXY
11500 IF N = 1 THEN N = 0 : RETURN
11510 IF I > 20 THEN GOTO 11540
11520   IF (I-5) > 0 THEN GOTO 11560
11530   RETURN
11540 IF (I-5) > 20 THEN GOTO 11560
11550   RETURN
11560 I = I-5
11570 N = 1
11580 GOSUB *SETXY
11590 RETURN
11600 '
11610 *LOWER.ARROW
11620 GOSUB *DISPXY
11630 IF N = 1 THEN GOTO 11680
11640   IF I = C(0,0)+1 AND C(0,0) <> 20 THEN RETURN
11650   IF I = C(1,0)+21 THEN RETURN
11660   N = 1
11670   RETURN
11680 IF I > 20 THEN GOTO 11710
11690   IF (I+5) > C(0,0)+1 THEN RETURN
11700   GOTO 11720
11710 IF (I+5) > 40 OR (I+5) > C(1,0)+21 THEN RETURN
11720 I = I+5
11730 N = 0
11740 GOSUB *SETXY
11750 RETURN
11760 '
11770 *LEFT.ARROW
11780 GOSUB *DISPXY
11790 IF I > 20 THEN GOTO 11830
11800   I = I-1
11810   IF I = 0 THEN I = 21
11820   GOTO 11870
11830 I = I+1
11840 IF I = 41 THEN I = 1 : GOTO 11870
11850   IF I = C(1,0)+21 THEN N = 0 : GOTO 11870
11860   IF I > C(1,0)+21 THEN I = 1
11870 GOSUB *SETXY
11880 RETURN
11890 '
11900 '***SUB データの挿入
11910 '
11920 *INSERT
11930 IF C(0,0) >= 20 OR C(1,0) >= 20 THEN
      CLS : PRINT "データ領域はいっぱい挿入できません。" :
      RETURN
11940 I1 = I

```

```

11950 IF I > 20 THEN
    I2 = C(1,0)+20 : C(1,0) = C(1,0)+1
    ELSE I2 = C(0,0) : C(0,0) = C(0,0)+1
11960 FOR I = I2 TO I1 STEP -1
11970   C(0,I+1) = C(0,I)
11980   C(1,I+1) = C(1,I)
11990 NEXT I
12000 LOCATE X,Y : PRINT SPC(7);
12010 LOCATE X,Y+2 : PRINT SPC(7);
12020 FOR I = I1+1 TO I2+1
12030   READ X, Y
12040   LOCATE X,Y : PRINT USING "###.#" ; C(0,I);
12050   LOCATE X,Y+2 : PRINT USING "###.#" ; C(1,I);
12060 NEXT I
12070 I = I1
12080 GOSUB *SETXY
12090 CLS : PRINT "Insert Mode D(elete, 123(Data ニュウリョク ";
12100 FOR N = 0 TO 1
12110   LOCATE X,Y+N*2 : PRINT SPC(7);
12120   A1$ = INPUT$(1)
12130   IF A1$ = "D" THEN GOSUB *DEL : GOTO 12200
12140   GOSUB *INPSUB
12150   IF ER% THEN GOTO 12110
12160   C(N,I) = C0
12170   GOSUB *DISPXY
12180 NEXT N
12190 N = 0
12200 RETURN
12210 '
12220 '**** SUB ひとつ前のレコードを入力
12230 '
12240 *SAME
12250 REC = RNO-1 : GOSUB *GETFILE1
12260 GOSUB *CLRDATA
12270 GOSUB *DISPDATA
12280 N = 0
12290 I = C(0,0)+1
12300 IF I = 21 THEN I = 20
12310 GOSUB *SETXY
12320 RETURN
12330 '
12340 '**** SUB カーソル位置にデータ入力
12350 '
12360 *INPSUB
12370 ER% = 0
12380 A$ = A1$
12390 IF ASC(A1$) = 8HD THEN C0 = 2 : RETURN
12400 LOCATE X,Y+N*2
12410 PRINT A1$;
12420 C0 = VAL(A1$)
12430 IF C0 = 0 AND A1$ <> "0" AND A1$ <> "." AND A1$ <> "-" THEN ER% = -1 : RETUR
12440 CC = 1
12450 A1$ = INPUT$(1) : PRINT A1$;
12460 WHILE ASC(A1$) <> 8HD AND CC < 6
12470   IF ASC(A1$) <> 8 AND ASC(A1$) <> 8H7F THEN GOTO 12500
12480   PRINT CHR$(8);CHR$(8);" ";CHR$(8);
12490   A$ = LEFT$(A$,LEN(A$)-1) : GOTO 12520
12500   A$ = A$+A1$
12510   CC = CC+1
12520   A1$ = INPUT$(1) : PRINT A1$;
12530 WEND
12540 C0 = VAL(A$)

```

```

12550 IF C0 = 0 AND A$ <> "0" THEN ER% = -1
12560 RETURN
12570 '
12580 '*** SUB 削除
12590 '
12600 *DEL
12610 IF I = C(0,0)+1 OR I = C(0,0)+21 THEN RETURN
12620 CLS: PRINT "カーソル位置のデータを削除してよいか (Y/N) ";
12630 INPUT A1$
12640 IF A1$ <> "Y" AND A1$ <> "y" THEN RETURN
12650 I1 = I
12660 IF I > 20 THEN
    C(1,0) = C(1,0)-1 : I2 = C(1,0)+20
    ELSE C(0,0) = C(0,0)-1 : I2 = C(0,0)
12670 FOR I = I1 TO I2
12680   C(0,I) = C(0,I+1)
12690   C(1,I) = C(1,I+1)
12700 NEXT I
12710 FOR I = I1 TO I2
12720   LOCATE X,Y : PRINT USING "###.#" ; C(0,I);
12730   LOCATE X,Y+2 : PRINT USING "###.#" ; C(1,I);
12740   READ X, Y
12750 NEXT I
12760 LOCATE X,Y : PRINT SPC(7)
12770 LOCATE X,Y+2 : PRINT SPC(7)
12780 I = I1
12790 GOSUB *SETXY
12800 RETURN
12810 '
12820 '****SUB C(N,I) を X,Y に表示
12830 '
12840 *DISPXY
12850 LOCATE X,Y+N*2 : PRINT SPC(7);
12860 IF C(0,0) <> 20 AND I = C(0,0)+1 THEN GOTO 12890
12870   IF I = C(1,0)+21 THEN GOTO 12890
12880   LOCATE X,Y+N*2 : PRINT USING "###.#" ; C(N,I);
12890 RETURN
12900 '
12910 '*** SUB 測点名他の表示
12920 '
12930 *DISPNUM
12940 LOCATE 34,6 : PRINT "測点 "; H1$
12950 LOCATE 0,4 : PRINT "レコード N o . "; RNO-1; " 全レコード数 "; REND-1;
12960 RETURN
12970 '
12980 ' データファイルの読み込みと書き込み
12990 '
13000 *GETFILE1
13010 I1 = 0
13020 J1 = 0
13030 FOR RR = 0 TO 5
13040   GET #2,(REC-1)*6+RR+1
13050   IF RR = 3 THEN I1 = 1 : J1 = 0
13060   FOR JJ = 0 TO 14
13070     C(I1,J1+JJ) = CVS(C$(JJ))
13080   NEXT JJ
13090   J1 = J1+15
13100 NEXT RR
13110 RETURN
13120 *PUTFILE1
13130 I1 = 0
13140 J1 = 0
13150 FOR RR = 0 TO 5

```

```

13160 IF RR = 3 THEN I1 = 1 : J1 = 0
13170 FOR JJ = 0 TO 14
13180 LSET C$(JJ) = MKS$(C(I1,J1+JJ))
13190 NEXT JJ
13200 J1 = J1+15
13210 PUT #2,(REC-1)*6+RR+1
13220 NEXT RR
13230 RETURN
13240 *GETFILE2
13250 ON ERROR GOTO *ERORUTIN
13260 GET #3, REC
13270 H1$ = HH$
13280 FOR I1 = 0 TO 12
13290 H(I1) = CVS(H$(I1))
13300 NEXT I1
13310 ON ERROR GOTO 0
13320 RETURN
13330 '
13340 ' 表示の休止
13350 '
13360 *SLEEP
13370 FOR IS = 0 TO 5000 : NEXT IS
13380 RETURN
13390 '*****
13400 ' DATA C R T の表示位置
13410 '*****
13420 *RDATA
13430 DATA 41, 8, 49, 8, 57, 8, 65, 8, 73, 8, 41, 12, 49, 12, 57, 12, 65, 12, 73,
, 12, 41, 16, 49, 16, 57, 16, 65, 16, 73, 16, 41, 20, 49, 20, 57, 20, 65, 20, 73,
, 20
13440 *LDATA
13450 DATA 32, 8, 24, 8, 16, 8, 8, 8, 0, 8, 32, 12, 24, 12, 16, 12, 8, 12, 0, 16,
, 32, 16, 24, 16, 16, 16, 8, 16, 0, 16, 32, 20, 24, 20, 16, 20, 8, 20, 0, 20

```

```

10 '*****
20 ' ( FOREST ROAD DROWING SYSTEM )
30 ' FILE NAME="CR20"
40 ' CROSS PROFILE DROWING SUB-PROGRAM NO.2
50 ' CREATED BY A.FUKUDA
60 '*****
1000 *SUB2 : 'サブプログラムのエントリーポイント
1010 CLS
1020 PRINT " <<< 拡幅量と横断勾配の設定 >>>" : PRINT
1030 PRINT " 1:緩和区間は全線で一定です."
1040 PRINT " 2:拡幅の位置は,(1)内側,(2)外側,(3)両側に設定できます."
1050 PRINT " 3:横断勾配は右上がりプラスとします."
1060 '*****
1070 ' C R 1 ファイルのオープン
1080 '*****
1090 FIELD #2, 44 AS DUMY$, 4 AS E$(1), 4 AS E$(2), 4 AS E$(3), 4 AS E$(4)
1100 OPEN F2$ AS #2
1110 '*****
1120 ' P R F ファイルのフィールド
1130 '*****
1140 FIELD #3, 8 AS HH$, 4 AS H$(0), 4 AS H$(1), 4 AS H$(2), 4 AS H$(3),
4 AS H$(4), 4 AS H$(5), 4 AS H$(6), 4 AS H$(7), 4 AS H$(8),
4 AS H$(9), 4 AS H$(10), 4 AS H$(11), 4 AS H$(12)
1150 '*****
1160 ' P L N ファイルのフィールド
1170 '*****
1180 FIELD #6, 4 AS D1$(1), 4 AS D1$(2), 4 AS D1$(3), 4 AS D1$(4),
4 AS D1$(5), 4 AS D1$(6), 4 AS D1$(7), 4 AS D1$(8),
4 AS D1$(9), 4 AS D1$(10), 4 AS D1$(11), 4 AS D1$(12)
1190 '*****
1200 ' 初期設定
1210 '*****
1220 DEFINT I,J,P,R
1230 GOSUB *CONST.ALL: '全体の定数
1240 GOSUB *CONST.BP: 'BPの定数
1250 RNO = 2
1260 PNO = 1
1270 REC = PNO : GOSUB *GETFILE2 : PNO = REC
1280 NWDIST = H(3)
1290 '*****
1300 ' 拡幅量と横断勾配の設定
1310 '*****
1320 FOR DNO% = 2 TO DEND%
1330 GOSUB *SET.BCC: 'BCC,ECC等の設定
1340 '...各IPでの基準量
1350 IF DNO% <> DEND% THEN GOSUB *CONST.CURVE ELSE GOSUB *CONST.EP
1360 GOSUB *KANWA.SESSEN: '緩和接続係数
1370 '...各測点のデータ計算
1380 WHILE NWDIST <= MC(1)
1390 ON TYPE4 GOSUB *SOKUTEN, *FUKUGO, *FUKUGO
1400 E(0) = PNO
1410 REC = RNO : GOSUB *PUTFILE1
1420 ' LPRINT H1$, NWDIST, E(0), E(1), E(2), E(3)
1430 RNO = RNO+1
1440 PNO = PNO+1
1450 REC = PNO : GOSUB *GETFILE2 : PNO = REC
1460 IF PFLG = 1 THEN GOTO 1490
1470 NWDIST = H(3)
1480 WEND
1490 NEXT DNO%
1500 REND = RNO-1
1510 E(0) = REND
1520 FOR I = 1 TO 3

```

```

1530 E(I) = 0
1540 NEXT I
1550 REC = 1 : GOSUB *PUTFILE1
1560 'LPRINT E(0)
1570 '*****
1580 '  退避所,特別な拡幅
1590 '*****
1600 ITURN = 0
1610 INPUT "退避所(車廻し)の設計をしますか.(Y/N) "; N$
1620 IF N$ <> "Y" AND N$ <> "y" THEN GOTO *INSATU
1630 INPUT "退避所の数 = "; ITURN
1640 IF ITURN <= 0 THEN GOTO *INSATU
1650 FOR I = 1 TO ITURN
1660 PRINT "退避所 No."; I
1670 INPUT "  始点(累計距離) = "; TURN(1,I)
1680 REC = DEND% : GOSUB *GETFILE3
1690 IF TURN(1,I) > D1(9) THEN PRINT "データの範囲外です." : GOTO 1670
1700 INPUT "  有効長 = "; LTURN1
1710 INPUT "  取付長 = "; LTURN0
1720 INPUT "  拡幅量(左) = "; TWIDEN(1,I)
1730 INPUT "  拡幅量(右) = "; TWIDEN(2,I)
1740 TURN(0,I) = TURN(1,I) - LTURN0
1750 TURN(2,I) = TURN(1,I) + LTURN1
1760 TURN(3,I) = TURN(2,I) + LTURN0
1770 NEXT I
1780 OPEN FILE$+".TRN" FOR OUTPUT AS #4
1790 WRITE #4, ITURN
1800 FOR I = 1 TO ITURN
1810 WRITE #4, TWIDEN(1,I), TWIDEN(2,I),
      TURN(0,I), TURN(1,I), TURN(2,I), TURN(3,I)
1820 NEXT I
1830 CLOSE #4
1840 PNO = 1
1850 RNO = 1
1860 I = 1
1870 REC = RNO : GOSUB *GETFILE1
1880 REC = E(0) : GOSUB *GETFILE2
1890 WHILE I <= ITURN
1900 LTURN0 = TURN(1,I) - TURN(0,I)
1910 WHILE H(3) <= TURN(0,I)
1920 RNO = RNO+1 : IF RNO > REND THEN GOTO *INSATU
1930 REC = RNO : GOSUB *GETFILE1
1940 REC = E(0) : GOSUB *GETFILE2
1950 WEND
1960 WHILE H(3) < TURN(1,I)
1970 LTURN2 = (H(3) - TURN(0,I)) / LTURN0
1980 FOR J = 1 TO 2
1990 M1 = E(J)
2000 M2 = TWIDEN(J,I) * LTURN2
2010 GOSUB *MAXDATA
2020 E(J) = MAXD
2030 NEXT J
2040 REC = RNO : GOSUB *PUTFILE1
2050 RNO = RNO+1 : IF RNO > REND THEN GOTO *INSATU
2060 REC = RNO : GOSUB *GETFILE1
2070 REC = E(0) : GOSUB *GETFILE2
2080 WEND
2090 WHILE H(3) <= TURN(2,I)
2100 FOR J = 1 TO 2
2110 M1 = E(J)
2120 M2 = TWIDEN(J,I)
2130 GOSUB *MAXDATA
2140 E(J) = MAXD

```

```

2150 NEXT J
2160 REC = RNO : GOSUB *PUTFILE1
2170 RNO = RNO+1 : IF RNO > REND THEN GOTO *INSATU
2180 REC = RNO : GOSUB *GETFILE1
2190 REC = E(0) : GOSUB *GETFILE2
2200 WEND
2210 WHILE H(3) < TURN(3,I)
2220 LTURN2 = (TURN(3,I) - H(3)) / LTURN0
2230 FOR J = 1 TO 2
2240 M1 = E(J)
2250 M2 = TWIDEN(J,I) * LTURN2
2260 GOSUB *MAXDATA
2270 E(J) = MAXD
2280 NEXT J
2290 REC = RNO : GOSUB *PUTFILE1
2300 RNO = RNO+1 : IF RNO > REND THEN GOTO *INSATU
2310 REC = RNO : GOSUB *GETFILE1
2320 REC = E(0) : GOSUB *GETFILE2
2330 WEND
2340 I = I+1
2350 WEND
2360 '*****
2370 '  印刷
2380 '*****
2390 *INSATU
2400 INPUT "拡幅と横断勾配のデータを印刷しますか (Y/N) "; N$
2410 IF N$ <> "Y" AND N$ <> "y" THEN GOTO *QUIT
2420 LPRINT
2430 LPRINT "タイトル", TS
2440 LPRINT "設計速度 = ", S : IF SPEED = 1 THEN LPRINT "規格外"
      ELSE LPRINT SPEED*10; "km/h"
2450 LPRINT "幅員 = ", W : IF WIDE = 1 THEN LPRINT "m"
      ELSE LPRINT W/10; "m"
2460 LPRINT "緩和区間長 = ", L : IF LSECT2 = 1 THEN LPRINT "m"
      ELSE LPRINT L/10; "m"
2470 LPRINT "曲線内の緩和区間長 = ", L : IF LSECT1 = 1 THEN LPRINT "m"
      ELSE LPRINT L/10; "m"
2480 LPRINT "拡幅基準量 = ", B : IF TYPE1 = 1 THEN LPRINT "1,2級の標準"
      ELSE IF TYPE1 = 2 THEN LPRINT "3級の標準"
      ELSE LPRINT "各曲線ごとに設定"
2490 LPRINT "曲線区間の横断勾配 = ", G : IF TYPE3 = 1 THEN LPRINT "標準値"
      ELSE LPRINT "各曲線ごとに設定"
2500 LPRINT "直線区間の横断勾配 = ", G : IF TYPE2 = 1 THEN LPRINT "レベル"
      ELSE LPRINT CRSA; "度"
2510 LPRINT:LPRINT
2520 LPRINT "  測点      累計距離      左拡幅      右拡幅      勾配"
2530 REC = 1 : GOSUB *GETFILE1
2540 REND = E(0)
2550 FOR RNO = 2 TO REND
2560 REC = RNO : GOSUB *GETFILE1
2570 REC = E(0) : GOSUB *GETFILE2
2580 LPRINT USING "&      &      ####.#      ##.##      ##.##      ##.##      "; H1$, H(3), E(1)
      E(2), E(3)
2590 NEXT RNO
2600 '*****待避所
2610 LPRINT:LPRINT:LPRINT
2620 LPRINT "待避所等の拡幅"
2630 IF ITURN = 0 THEN LPRINT "待避所      なし" : GOTO *QUIT
2640 LPRINT "待避所      BCC      BTT      ETT      ECC      左拡幅      右拡幅"
2650 FOR I = 1 TO ITURN
2660 LPRINT USING "      ##      ####.#      ####.#      ####.#      ####.#      ##.##      ##.##";
      I, TURN(0,I), TURN(1,I), TURN(2,I), TURN(3,I), TWIDEN(1,I), TWIDEN(2,I)
2670 NEXT I
2680 '*****
2690 '  終了
2700 '*****

```

```

2710 *QUIT
2720 CLOSE #2
2730 OPEN F4$ FOR OUTPUT AS #2
2740 WRITE #2, T$, DEND%, DUMY1%, PEND, DUMY2%, WIDE2
2750 CLOSE #2
2760 CHAIN "CROSS", 1000, ALL
10000 '*****
10010 ' エラー処理
10020 '*****
10030 *ERORUTIN
10040 PRINT "ERR = ";ERR, "ERL = "; ERL
10050 PRINT "マニュアルを見て下さい."
10060 END
10070 '*****
10080 ' SUB-ROUTIN
10090 '*****
10100 '
10110 '**** SUB 全線の定数
10120 '
10130 *CONST.ALL
10140 '...設計速度
10150 PRINT
10160 INPUT "設計速度 [1=規格外, 2=20km/h, 3=30km/h, 4=40km/h]= "; SPEED
10170 IF SPEED < 1 OR SPEED > 4 THEN GOTO 10160
10180 '...すりつけ率
10190 RESIO = 100*SPEED/4
10200 '...幅員
10210 PRINT
10220 INPUT "車道幅員(m)= "; WIDE
10230 WIDE2 = WIDE/2
10240 '...緩和区間
10250 PRINT
10260 INPUT "緩和区間長 (m)= "; LSECT2
10270 INPUT "曲線内の緩和区間長(m)= "; LSECT1
10280 LSECT0 = LSECT2-LSECT1
10290 '...拡幅量
10300 PRINT
10310 PRINT "拡幅基準量"
10320 PRINT " 1)1,2級の標準"
10330 PRINT " 2)3級の標準"
10340 PRINT " 3)各曲線ごとに入力 ";
10350 INPUT TYPE1
10360 IF TYPE1 < 1 OR TYPE1 > 3 THEN GOTO 10300
10370 '...横断勾配
10380 PRINT
10390 PRINT "直線区間の横断勾配のタイプ"
10400 PRINT " 1)レベル"
10410 PRINT " 2)片勾配 ";
10420 INPUT TYPE2
10430 IF TYPE2 < 1 OR TYPE2 > 2 THEN GOTO 10380
10440 IF TYPE2 = 1 THEN CRSA = 0 : GOTO 10470
10450 PRINT
10460 INPUT "直線区間の横断勾配(%) ="; CRSA
10470 PRINT
10480 PRINT "曲線区間の横断勾配"
10490 PRINT " 1)標準"
10500 PRINT " 2)各曲線ごとに入力 ";
10510 INPUT TYPE3
10520 IF TYPE3 < 1 OR TYPE3 > 2 THEN GOTO 10470
10530 PRINT : INPUT "以上の設定でよい (Y/N) "; N$
10540 IF N$ <> "Y" AND N$ <> "y" THEN GOTO *CONST.ALL
10550 RETURN
10560 '

```

```

10570 '**** SUB 入力用画面
10580 '
10590 *PITEM
10600 CLS
10610 ' ### ### & & ###.## ###.## ### & & ###.##
10620 PRINT "<< 拡幅量と横断勾配の基準量の入力 >>"
10630 LOCATE 0,5
10640 PRINT " IP R 方向 左拡幅 右拡幅 横断(%) タイプ ECC-BCC"
10650 RETURN
10660 '
10670 '**** SUB BPの定数
10680 '
10690 *CONST.BP
10700 GOSUB *PITEM
10710 LOCATE 2,8 : PRINT "BP";
10720 REC = 2 : GOSUB *GETFILE3
10730 IF D1(4) <= 0 THEN CURV(2) = 1 : A1$ = "左" ELSE CURV(2) = 2 : A1$ = "右"
10740 PRADI! = D1(5)
10750 Y = 10
10760 GOSUB *PRINT.DATA
10770 CURV(1) = 1
10780 LOCATE 0,14
10790 INPUT " BPでの拡幅量(m) = "; WIDEN(0,1)
10800 IF WIDEN(0,1) <> 0 THEN GOTO 10880
10810 WIDEN(1,1) = 0
10820 CRSA(1) = CRSA
10830 TYPE0 = 0
10840 RORL = 1
10850 Y = 8
10860 GOSUB *PRINT.DAT1
10870 GOTO 11040
10880 LOCATE 0,15
10890 INPUT " BPからの緩和区間長(m) = "; BPL
10900 WIDEN(0,1) = WIDEN(0,1)*LSECT2/BPL
10910 LOCATE 0,16
10920 INPUT " BPの拡幅のタイプ [ 1)左側 2)右側 3)両側 ] = "; TYPE0
10930 IF TYPE0 < 1 OR TYPE0 > 3 THEN GOTO 10910
10940 WIDEN = WIDEN(0,1)
10950 GOSUB *KAKUFUKU
10960 WIDEN(0,1) = SWID0
10970 WIDEN(1,1) = SWID1
10980 LOCATE 0,17
10990 INPUT " BPでの横断勾配 (+n or -n or 0 %) = "; CRSA(1)
11000 CRSA(1) = CRSA(1)*LSECT2/BPL
11010 Y = 8
11020 RORL = 1
11030 GOSUB *PRINT.DAT1
11040 REC = 1 : GOSUB *GETFILE3
11050 IF WIDEN(0,1) <> 0 OR WIDEN(1,1) <> 0 THEN GOTO 11090
11060 ECC = D1(11)
11070 ETT = ECC
11080 GOTO 11110
11090 ECC = D1(11)-LSECT1
11100 ETT = ECC+LSECT2
11110 MC(0) = ECC-4
11120 RAD!(1) = D1(5)
11130 REC = 2 : GOSUB *GETFILE3
11140 IF D1(5) = 0 THEN
    BTT = D1(9) : BCC = BTT
    ELSE BTT = D1(9)-LSECT0 : BCC = BTT+LSECT2
11150 MC(1) = D1(10)
11160 ECC(2) = ECC
11170 ETT(2) = ETT

```

```

11180 BTT(2) = BTT
11190 BCC(2) = BCC
11200 MC(2) = MC(1)
11210 MC(1) = MC(0)
11220 RAD!(2) = D1(5)
11230 LOCATE 51,9 : PRINT USING "###.##"; BTT-ETT;
11240 LOCATE 0,18
11250 INPUT "設定はこれでよい (Y/N) "; N$
11260 IF N$ <> "Y" AND N$ <> "y" THEN GOSUB *INP.GAMEN : GOTO *CONST.BP
11270 RETURN
11280 '
11290 '*****SUB BCC,ECCほかの設定
11300 '
11310 *SET.BCC
11320 GOSUB *PITEM
11330 CURV(0) = CURV(1)
11340 CURV(1) = CURV(2)
11350 WIDEN(0,0) = WIDEN(0,1)
11360 WIDEN(1,0) = WIDEN(1,1)
11370 CRSA(0) = CRSA(1)
11380 RAD!(0) = RAD!(1)
11390 '...一つ前の I P 表示
11400 IF DNO% <> 2 THEN GOTO 11480
11410 LOCATE 2,8
11420 PRINT "BP"
11430 D1(1) = D1(1)-1
11440 RORL = 0
11450 Y = 8
11460 GOSUB *PRINT.DATA1
11470 GOTO 11550
11480 D1(1) = D1(1)-1
11490 IF CURV(0) = 1 THEN A1$ = "左" ELSE A1$ = "右"
11500 PRADI! = RAD!(0)
11510 RORL = 0
11520 Y = 8
11530 GOSUB *PRINT.DATA
11540 GOSUB *PRINT.DATA1
11550 '...前の I P と現在の I P の間を表示
11560 ECC = ECC(2)
11570 ETT = ETT(2)
11580 BTT = BTT(2)
11590 BCC = BCC(2)
11600 MC(0) = MC(1)
11610 MC(1) = MC(2)
11620 BTET = BTT-ETT
11630 ECC(1) = ECC
11640 ETT(1) = ETT
11650 BTT(1) = BTT
11660 BCC(1) = BCC
11670 IF BTET > 0 THEN TYPE4 = 1 : A2$ = "直線" : GOTO 11690
11680 IF CURV(0) = CURV(1)
    THEN TYPE4 = 2 : A2$ = "複合" ELSE TYPE4 = 3 : A2$ = "S"
11690 IF DNO% = 1 THEN GOTO 11720
11700 Y = 9
11710 GOSUB *PRINT.DATA2
11720 '...現在の I P 表示
11730 IF DNO% = DEND% THEN LOCATE 2,10 : PRINT " EP"; : RETURN
11740 D1(1) = D1(1)+1
11750 RAD!(1) = RAD!(2)
11760 PRADI! = RAD!(1)
11770 IF CURV(1) = 1 THEN A1$ = "左" ELSE A1$ = "右"
11780 RORL = 1
11790 Y = 10

```

```

11800 GOSUB *PRINT.DATA
11810 '...次の I P との間表示
11820 IF D1(5) <> 0 THEN GOTO 11860
11830 ECC(2) = D1(11)
11840 ETT(2) = ECC(2)
11850 GOTO 11880
11860 ECC(2) = D1(11)-LSECT1
11870 ETT(2) = ECC(2)+LSECT2
11880 REC = DNO%+1 : GOSUB *GETFILE3
11890 IF D1(5) <> 0 THEN GOTO 11930
11900 BTT(2) = D1(9)
11910 BCC(2) = BTT(2)
11920 GOTO 11950
11930 BTT(2) = D1(9)-LSECT0
11940 BCC(2) = BTT(2)+LSECT2
11950 MC(2) = D1(10)
11960 IF DNO% = DEND%-1 THEN GOTO 12030
11970 IF D1(4) <= 0
    THEN CURV(2) = 1 : A1$ = "左" ELSE CURV(2) = 2 : A1$ = "右"
11980 IF CURV(1) = CURV(2) THEN A2$ = "複合" ELSE A2$ = "S"
11990 BTET = BTT(2)-ETT(2)
12000 IF BTET > 0 THEN A2$ = "直線"
12010 Y = 11
12020 GOSUB *PRINT.DATA2
12030 '...次の I P 表示
12040 IF DNO% = DEND%-1 THEN LOCATE 2,12 : PRINT "EP" : RETURN
12050 RAD!(2) = D1(5)
12060 PRADI! = RAD!(2)
12070 Y = 12
12080 GOSUB *PRINT.DATA
12090 RETURN
12100 '
12110 '***** SUB 各曲線の定数
12120 '
12130 *CONST.CURVE
12140 IF RAD!(1) <> 0 THEN GOTO 12200
12150 WIDEN(0,1) = 0
12160 WIDEN(1,1) = 0
12170 TYPE0 = 0
12180 CRSA(1) = CRSA
12190 GOTO 12360
12200 IF TYPE1 <> 3 THEN GOTO 12240
12210 LOCATE 0,14
12220 INPUT "拡幅基準量 = "; WIDEN(0,1)
12230 IF WIDEN(0,1) = 0 THEN WIDEN(1,1) = 0 : TYPE0 = 0 : GOTO 12310
12240 LOCATE 0,15
12250 INPUT "拡幅のタイプ [ 1)内側 2)外側 3)両側 ] = "; TYPE0
12260 IF TYPE0 < 1 OR TYPE0 > 3 THEN GOTO 12240
12270 GOSUB *KIKAKU
12280 GOSUB *KAKUFUKU
12290 WIDEN(0,1) = SWID0
12300 WIDEN(1,1) = SWID1
12310 IF TYPE3 <> 2 THEN GOSUB *STD.ODAN : GOTO 12350
12320 LOCATE 0,16
12330 INPUT "横断勾配 (%) = "; CRSA(1)
12340 IF CRSA(1) = 0 THEN CRSA(1) = CRSA : GOTO 12360
12350 IF CURV(1) = 1 THEN CRSA(1) = ABS(CRSA(1)) ELSE CRSA(1) = -ABS(CRSA(1))
12360 RORL = 1
12370 Y = 10
12380 GOSUB *PRINT.DATA1
12390 IF BTT > ETT AND ((WIDEN(0,0) > 0 AND WIDEN(0,1) > 0) OR (WIDEN(1,0) > 0
    AND WIDEN(1,1) > 0)) THEN GOSUB *CHOKUSEN
12400 LOCATE 0,20

```

```

12410 PRINT "IP "; D1(1)-1; "の設定はこれでよい (Y/N) ";
12420 INPUT N$
12430 IF N$ <> "Y" AND N$ <> "y" THEN GOTO *CONST.CURVE
12440 RETURN
12450 '
12460 ' ***** SUB EPの定数
12470 '
12480 *CONST.EP
12490 LOCATE 0,14
12500 INPUT "EPの拡幅量 = "; WIDEN(0,1)
12510 IF WIDEN(0,1) = 0 THEN WIDEN(1,1) = WIDEN(0,1) : TYPE0 = 0 : GOTO 12650
12520 LOCATE 0,15
12530 INPUT "EPまでの緩和区間長 (m) = "; EPL
12540 LOCATE 0,16
12550 INPUT "EPの後の拡幅のタイプ [ 1)左側 2)右側 3)両側 ] = "; TYPE0
12560 IF TYPE0 < 1 OR TYPE0 > 3 THEN GOTO 12540
12570 CURV(1) = 1
12580 WIDEN = WIDEN(0,1)*LSECT2/EPL
12590 GOSUB *KAKUFUKU
12600 WIDEN(0,1) = SWID0
12610 WIDEN(1,1) = SWID1
12620 LOCATE 0,17
12630 INPUT "EPの横断勾配 ( +n% or -n% ) = "; CRSA(1)
12640 CRSA(1) = CRSA(1)*LSECT2/EPL
12650 RORL = 1
12660 Y = 10
12670 GOSUB *PRINT.DATA1
12680 IF BTT > ETT AND ((WIDEN(0,0) > 0 AND WIDEN(0,1) > 0) OR (WIDEN(1,0) > 0 AND WIDEN(1,1) > 0)) THEN GOSUB *CHOKUSEN
12690 LOCATE 0,20
12700 INPUT "EPの設定はこれでよい (Y/N) "; N$
12710 IF N$ <> "Y" THEN GOTO *CONST.EP
12720 RETURN
12730 '
12740 ' ***** SUB 拡幅基準量
12750 '
12760 *KAKUFUKU
12770 ON CURV(1)*TYPE0+1 GOTO 12780, 12810, 12840, 12870, 12810, 12890, 12870
12780 SWID0 = 0
12790 SWID1 = 0
12800 RETURN
12810 SWID0 = WIDEN
12820 SWID1 = 0
12830 RETURN
12840 SWID0 = 0
12850 SWID1 = WIDEN
12860 RETURN
12870 SWID0 = WIDEN/2
12880 SWID1 = SWID0
12890 RETURN
12900 '
12910 ' ***** SUB 「林道規準」 拡幅量
12920 '
12930 *KIKAKU
12940 IF TYPE1 = 2 THEN GOTO 13070
12950 '.... 1,2級林道
12960 IF RAD!(1) >= 45 THEN WIDEN = 0 : RETURN
12970 IF RAD!(1) >= 40 THEN WIDEN = .25 : RETURN
12980 IF RAD!(1) >= 30 THEN WIDEN = .5 : RETURN
12990 IF RAD!(1) >= 25 THEN WIDEN = .75 : RETURN
13000 IF RAD!(1) >= 20 THEN WIDEN = 1 : RETURN
13010 IF RAD!(1) >= 18 THEN WIDEN = 1.25 : RETURN
13020 IF RAD!(1) >= 15 THEN WIDEN = 1.5 : RETURN

```

```

13030 IF RAD!(1) >= 14 THEN WIDEN = 1.75 : RETURN
13040 IF RAD!(1) >= 13 THEN WIDEN = 2 : RETURN
13050 WIDEN = 2.25
13060 RETURN
13070 '...3級林道
13080 IF RAD!(1) >= 60 THEN WIDEN = 0 : RETURN
13090 IF RAD!(1) >= 36 THEN WIDEN = .25 : RETURN
13100 IF RAD!(1) >= 19 THEN WIDEN = .5 : RETURN
13110 IF RAD!(1) >= 13 THEN WIDEN = .75 : RETURN
13120 IF RAD!(1) >= 10 THEN WIDEN = 1 : RETURN
13130 WIDEN = 1.25
13140 RETURN
13150 '
13160 ' ***** SUB 横断勾配の標準
13170 '
13180 *STD.ODUN
13190 ON SPEED GOTO 13290, 13290, 13240, 13200
13200 '....40km/h
13210 IF RAD!(1) > 45 THEN CRSA(1) = CRSA : RETURN
13220 CRSA(1) = 8
13230 RETURN
13240 '....30km/h
13250 IF RAD!(1) >= 35 THEN CRSA(1) = 3 :
      IF CRSA(1) < CRSA THEN CRSA(1) = CRSA : RETURN ELSE RETURN
13260 IF RAD!(1) >= 30 THEN CRSA(1) = 6 : RETURN
13270 CRSA(1) = 8
13280 RETURN
13290 '....20km/h
13300 IF RAD!(1) > 15 THEN CRSA(1) = CRSA : RETURN
13310 IF RAD!(1) > 12 THEN CRSA(1) = 3 :
      IF CRSA(1) < CRSA THEN CRSA(1) = CRSA : RETURN ELSE RETURN
13320 CRSA(1) = 8
13330 RETURN
13340 '....規格外
13350 INPUT "横断勾配 (%)"; CRSA(1)
13360 RETURN
13370 '
13380 ' ***** SUB 緩和接線の係数
13390 '
13400 *KANWA.SESSEN
13410 ECC1 = ECC
13420 BCC1 = BCC
13430 '...重なりがない時の拡幅
13440 FOR I = 0 TO 1
13450   CONWIDA(I,0) = -WIDEN(I,0)/LSECT2
13460   CONWIDB(I,0) = WIDEN(I,0)
13470   CONWIDA(I,1) = WIDEN(I,1)/LSECT2
13480   CONWIDB(I,1) = WIDEN(I,1)-CONWIDA(I,1)*(BCC-ECC)
13490 NEXT I
13500 IF TYPE4 = 1 THEN GOTO 13640
13510 '...重なりのある時の拡幅
13520 FOR I = 0 TO 1
13530   IF WIDEN(I,0) = 0 AND WIDEN(I,1) = 0 THEN GOTO 13630
13540   IF WIDEN(I,0) >= CONWIDB(I,1) THEN GOTO 13590
13550   ECC = (WIDEN(I,0)-CONWIDB(I,1))/CONWIDA(I,1)+ECC
13560   CONWIDA(I,0) = CONWIDA(I,1)
13570   CONWIDB(I,0) = WIDEN(I,0)
13580   GOTO 13630
13590   IF WIDEN(I,1) >= (CONWIDA(I,0)*(ECC-BCC)+CONWIDA(I,1)) THEN GOTO 13620
13600   BCC = (WIDEN(I,1)-CONWIDB(I,0))/CONWIDA(I,0)+ECC
13610   GOTO 13630
13620   CONWIDA(I,0) = (WIDEN(I,1)-WIDEN(I,0))/(BCC-ECC)
13630 NEXT I

```

```

13640 '....重なりが無いときの勾配
13650 CONCRSA(0) = (CRSA-CRSA(0))/LSECT2
13660 CONCRSB(0) = CRSA(0)
13670 CONCRSA(1) = (CRSA(1)-CRSA)/LSECT2
13680 CONCRSB(1) = CRSA(1)-CONCRSA(1)*(BCC1-ECC1)
13690 '....重なりがあるときの勾配
13700 IF SPEED = 1 THEN INPUT "横断勾配の設計区間長は "; L4 : GOTO 13720
13710 L4 = WIDE2*RESIO*ABS(CRSA(1)-CRSA(0))/100
13720 IF L4 = 0 THEN RETURN
13730 ON TYPE4 GOTO 13920, 13740, 13830
13740 '.....複合カーブ
13750 IF (BCC1-ECC1) < L4 THEN GOTO 13780
13760 CONCRSA(0) = (CRSA(1)-CRSA(0))/(BCC1-ECC1)
13770 RETURN
13780 CONCRSA(0) = (CRSA(1)-CRSA(0))/L4
13790 MCC1 = (BCC1-ECC1)/2+ECC1
13800 ECC1 = MCC1-L4/2
13810 BCC1 = MCC1+L4/2
13820 RETURN
13830 '.....S字カーブ
13840 CONCRSA(0) = (CRSA(1)-CRSA(0))/L4
13850 MCC1 = (BCC1-ECC1)/2+ECC1
13860 ECC1 = MCC1-L4/2
13870 BCC1 = MCC1+L4/2
13880 IF ECC1 < MC(0) AND BCC1 > MC(1)
      THEN ECC1 = MC(0) : BCC1 = MC(1) : GOTO 13910
13890 IF ECC1 < MC(0) THEN ECC1 = MC(0) : BCC1 = ECC1+L4 : GOTO 13910
13900 IF BCC1 > MC(1) THEN BCC1 = MC(1) : ECC1 = BCC1-L4
13910 CONCRSA(0) = (CRSA(1)-CRSA(0))/(BCC1-ECC1)
13920 RETURN
13930 '
13940 '*****SUB 重なりのない測点
13950 '
13960 *SOKUTEN
13970 FOR I = 0 TO 1
13980 M1 = CONWIDA(I,0)*(NWDIST-ECC)+CONWIDB(I,0)
13990 M2 = CONWIDA(I,1)*(NWDIST-ECC)+CONWIDB(I,1)
14000 GOSUB *MAXDATA
14010 E(I+1) = WIDEN(I,0)*ABS(NWDIST < ECC)+MAXD*ABS(NWDIST >= ECC AND NWDIST
< BCC)+WIDEN(I,1)*ABS(NWDIST >= BCC)
14020 NEXT I
14030 E(3) = CRSA(0)*ABS(NWDIST < ECC)+(CONCRSA(0)*(NWDIST-ECC)+CONCRSB(0))*ABS(
NWDIST >= ECC AND NWDIST < ETT)+CRSA*ABS(NWDIST >= ETT AND NWDIST <= BTT)+(CONC
SA(1)*(NWDIST-ECC)+CONCRSB(1))*ABS(NWDIST > BTT AND NWDIST < BCC)+CRSA(1)*ABS(N
WDIST >= BCC)
14040 RETURN
14050 '
14060 '***** SUB 複合曲線, S字曲線の測点
14070 '
14080 *FUKUGO
14090 '.....拡幅量
14100 FOR I = 0 TO 1
14110 E(I+1) = WIDEN(I,0)*ABS(NWDIST < ECC)+(CONWIDA(I,0)*(NWDIST-ECC)+CONWID
(I,0))*ABS(NWDIST >= ECC AND NWDIST < BCC)+WIDEN(I,1)*ABS(NWDIST >= BCC)
14120 NEXT I
14130 '.....勾配
14140 E(3) = CRSA(0)*ABS(NWDIST < ECC1)+(CONCRSA(0)*(NWDIST-ECC1)+CONCRSB(0))*A
S(NWDIST >= ECC1 AND NWDIST < BCC1)+CRSA(1)*ABS(NWDIST >= BCC1)
14150 RETURN
14160 '
14170 '***** SUB 直線区間の勾配
14180 '
14190 *CHOKUSEN

```

```

14200 LOCATE 0,18
14210 INPUT "直線区間を緩和区間としますか (Y/N) "; N$
14220 IF N$ = "Y" THEN TYPE4 = 2 : RETURN
14230 TYPE4 = 1
14240 IF TYPE2 <> 2 THEN RETURN
14250 LOCATE 0,19
14260 INPUT "直線区間の横断勾配は [ 1)右上がり 2)左上がり ] "; TYPE5
14270 IF TYPE5 <1 OR TYPE5 >2 THEN GOTO 14250
14280 IF TYPE5 = 1 THEN CRSA = ABS(CRSA) ELSE CRSA = -ABS(CRSA)
14290 RETURN
14300 '
14310 '***** SUB データのプリント
14320 '
14330 *PRINT.DATA
14340 LOCATE 2,Y : PRINT USING "###"; D1(1);
14350 *PRINT.DATA0
14360 LOCATE 7,Y : PRINT USING "####"; PRADI!;
14370 LOCATE 12,Y : PRINT A1$
14380 RETURN
14390 *PRINT.DATA1
14400 LOCATE 18,Y : PRINT USING "###.##"; WIDEN(0,RORL);
14410 LOCATE 26,Y : PRINT USING "###.##"; WIDEN(1,RORL);
14420 LOCATE 34,Y : PRINT USING "####"; CRSA(RORL);
14430 RETURN
14440 *PRINT.DATA2
14450 LOCATE 44,Y : PRINT A2$;
14460 LOCATE 51,Y : PRINT USING "###.##"; BTET;
14470 RETURN
14480 '
14490 '***** SUB 最大データ
14500 '
14510 *MAXDATA
14520 MAXD = M1
14530 IF MAXD < M2 THEN MAXD = M2
14540 RETURN
14550 '
14560 '***** SUB ファイルの入力
14570 '
14580 *GETFILE1
14590 GET #2, REC*6-3
14600 FOR II = 0 TO 3
14610 E(II) = CVS(E$(II))
14620 NEXT II
14630 RETURN
14640 *PUTFILE1
14650 FOR II = 0 TO 3
14660 LSET E$(II) = MKS$(E(II))
14670 NEXT II
14680 PUT #2, REC*6-3
14690 RETURN
14700 *GETFILE2
14710 PFLG = 0
14720 GET #3, REC
14730 H1$ = HH$
14740 FOR II = 0 TO 12
14750 H(II) = CVS(H$(II))
14760 NEXT II
14770 IF H(9) < 6 THEN RETURN
14780 REC = REC +1
14790 IF REC > PEND THEN PFLG = 1 : RETURN
14800 GOTO *GETFILE2
14810 *GETFILE3
14820 GET #6, REC

```

```

14830 FOR II =1 TO 12
14840   D1(II) = CVS(D1$(II))
14850 NEXT II
14860 RETURN

```

```

10 '*****
20 '      ( FOREST ROAD DROWING SYSTEM )
30 '      FILE NAME = "CR30"
40 '      CROSS PROFILE DROWING SUB-PROGRAM NO.3
50 '      CREATED BY A.FUKUDA
60 '*****
1000 *SUB3: 'サブプログラム NO. 3のエントリーポイント
1010 CLS
1020 PRINT "      <<< 構造物と法面の設計 >>>" : PRINT
1030 INPUT "構造物ファイルのパス名 "; PASS$
1040 '*****
1050 '      初期値
1060 '*****
1070 DEF FNPX9(X) = (AX+X)*100
1080 DEF FNPY9(Y) = (AY+Y)*100
1090 WX1 = -64
1100 WX2 = 64
1110 WY1 = -40
1120 WY2 = 40
1130 AX = 11
1140 AY = 9
1150 BLK% = 0
1160 BLUE% = 1
1170 GRN% = 4
1180 SKY% = 5
1190 YELLOW% = 6
1200 WHITE% = 7
1210 '*****
1220 '      構造物ファイルの入力
1230 '*****
1240 *PASS
1250 ON ERROR GOTO *ERDRAIN
1260 OPEN PASS$+"¥DRAIN.DIR" FOR INPUT AS #4
1270 GOTO 1310
1280   *ERDRAIN
1290   IF ERR = 53 THEN DN% = 0 : RESUME *WALL
1300   PRINT "ERR="ERR,"ERL=";ERL : END
1310 INPUT #4, DN%
1320 FOR I = 1 TO DN%
1330   INPUT #4, DT$(I)
1340 NEXT I
1350 CLOSE #4
1360 *WALL
1370 ON ERROR GOTO *ERWALL
1380 OPEN PASS$+"¥WALL.DIR" FOR INPUT AS #4
1390 GOTO 1430
1400   *ERWALL
1410   IF ERR = 53 THEN WN% = 0 : RESUME *BASE
1420   PRINT "ERR="ERR,"ERL=";ERL : END
1430 INPUT #4, WN%
1440 FOR I= 1 TO WN%
1450   INPUT #4, WT$(I)
1460 NEXT I
1470 CLOSE #4
1480 *BASE
1490 ON ERROR GOTO *ERBASE
1500 OPEN PASS$+"¥BASE.DIR" FOR INPUT AS #4
1510 GOTO 1550
1520   *ERBASE
1530   IF ERR = 53 THEN BN% = 0 : RESUME *CRSOPEN
1540   PRINT "ERR="ERR,"ERL=";ERL : END
1550 INPUT #4, BN%
1560 FOR I = 1 TO BN%

```

```

1570 INPUT #4, BT$(1)
1580 NEXT I
1590 CLOSE #4
1600 ON ERROR GOTO 0
1610 '*****
1620 ' C R S ファイルのオープン
1630 '*****
1640 *CRSOPEN
1650 ON ERROR GOTO 0
1660 FIELD #1, 4 AS K$(0), 4 AS K$(1), 4 AS K$(2), 4 AS K$(3), 4 AS K$(4),
      4 AS K$(5), 4 AS K$(6), 4 AS K$(7), 4 AS K$(8), 4 AS K$(9),
      4 AS K$(10), 4 AS K$(11), 4 AS K$(12), 4 AS K$(13), 4 AS K$(14)

1670 OPEN F1$ AS #1
1680 PRINT "横断図設計ファイル("; F1$; ")の使用ははじめてか (Y/N) ";
1690 INPUT N$
1700 IF N$ = "N" OR N$ = "n" THEN CRNO = 1 : GOTO 1770
1710 IF N$ = "Y" OR N$ = "y" THEN GOTO 1730
1720 GOTO 1680
1730 INPUT "このファイルを初期化してよいか (Y/N) "; N$
1740 IF N$ = "N" OR N$ = "n" THEN CRNO = 1 : GOTO 1770
1750 IF N$ = "Y" OR N$ = "y" THEN CRNO = 0 : GOTO 1770
1760 GOTO 1730
1770 '*****
1780 ' C R 1 ファイルのオープン
1790 '*****
1800 FIELD #2, 4 AS C$(0), 4 AS C$(1), 4 AS C$(2), 4 AS C$(3), 4 AS C$(4),
      4 AS C$(5), 4 AS C$(6), 4 AS C$(7), 4 AS C$(8), 4 AS C$(9),
      4 AS C$(10), 4 AS C$(11), 4 AS C$(12), 4 AS C$(13), 4 AS C$(14)

1810 OPEN F2$ AS #2
1820 REC = 1 : GOSUB *GETFILE2 : CREND = C(0,0)-1
1830 IF CRNO = 1 THEN GOTO 1900
1840 PRINT "ファイルを初期化してします。しばらくお待ちください。"
1850 GOSUB *CON.INIT
1860 FOR REC = 1 TO CREND
1870 GOSUB *PUTFILE1
1880 NEXT REC
1890 CRNO = 1
1900 REC = 1 : GOSUB *GETFILE1
1910 REC = 2 : GOSUB *GETFILE2 : PNO = C(0,44)
1920 '*****
1930 ' P R F ファイルのフィールド
1940 '*****
1950 FIELD #3, 8 AS H$(0), 4 AS H$(1), 4 AS H$(2), 4 AS H$(3),
      4 AS H$(4), 4 AS H$(5), 4 AS H$(6), 4 AS H$(7), 4 AS H$(8),
      4 AS H$(9), 4 AS H$(10), 4 AS H$(11), 4 AS H$(12)

1960 REC = PNO : GOSUB *GETFILE3
1970 '*****
1980 ' プロッタのオープン
1990 '*****
2000 REC = 1 : GOSUB *GETFILE1
2010 PRINT "プロッタをオープンします。準備はよいですか (OK=Push any key)";
2020 WHILE INKEY$="" : WEND
2030 OPEN "COM:N82N" AS #5
2040 '*****
2050 ' データ編集メニュー
2060 '*****
2070 CLS : CONSOLE 1,24
2080 GOSUB *GRAPHIC
2090 ON KEY GOSUB *ZOOM, *SCROLL, *INIT
2100 FOR K = 1 TO 3 : KEY (K) ON : NEXT K
2110 *MENU
2120 COMM$ = "Command C(onstruct, P(lotter, Z(oom, S(croll, I(nitial, B(efore, N
(ext, Q(uit " : GOSUB *COMM

```

```

2130 A1$ = INKEY$ : IF A1$ = "" THEN GOTO 2130
2140 PRINT A1$
2150 IF A1$ = "C" OR A1$ = "c" THEN GOSUB *CONST : GOTO *MENU
2160 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PLOTTER : GOTO *MENU
2170 IF A1$ = "B" OR A1$ = "b" THEN GOSUB *BEFORE : GOTO *MENU
2180 IF A1$ = "N" OR A1$ = "n" THEN GOSUB *NEXT.DATA : GOTO *MENU
2190 IF A1$ = "Z" OR A1$ = "z" THEN GOSUB *ZOOM : GOTO *MENU
2200 IF A1$ = "S" OR A1$ = "s" THEN GOSUB *SCROLL : GOTO *MENU
2210 IF A1$ = "I" OR A1$ = "i" THEN GOSUB *INIT : GOTO *MENU
2220 IF A1$ = "Q" OR A1$ = "q" THEN GOTO *QUIT
2230 GOTO *MENU
2240 '*****
2250 ' 構造物の設計
2260 '*****
2270 *CONST
2280 IF CRNO < 2 THEN GOTO 2470
2290 COMM$ = "ひとつ前の測点と同じ設計とするか (Y/N) ? " : GOSUB *COMM
2300 N$ = INKEY$ : IF N$ = "" THEN GOTO 2300
2310 PRINT N$
2320 IF N$ <> "Y" AND N$ <> "y" THEN GOTO 2470
2330 GOSUB *ERASE.CONST
2340 REC = CRNO-1 : GOSUB *GETFILE1
2350 COL% = WHITE% : GOSUB *GRAPHIC1
2360 COMM$ = "これでよいですか。(Y/N) ? " : GOSUB *COMM
2370 N$ = INKEY$ : IF N$ = "" THEN GOTO 2370
2380 PRINT N$
2390 IF N$ = "Y" OR N$ = "y" THEN GOTO 2440
2400 GOSUB *ERASE.CONST
2410 REC = CRNO : GOSUB *GETFILE1
2420 GOSUB *GRAPHIC
2430 GOTO 2470
2440 CRFLG = 1 : CRSAME = CRNO-1
2450 REC = CRNO : GOSUB *PUTFILE1
2460 RETURN
2470 IF CRFLG = 0 THEN GOTO 2530
2480 COMM$ = "現在の設計を変更しますか (Y/N) ? " : GOSUB *COMM
2490 N$ = INKEY$ : IF N$ = "" THEN GOTO 2490
2500 PRINT N$
2510 IF N$ <> "Y" AND N$ <> "y" THEN RETURN
2520 COL% = BLK% : FOR RL = 1 TO 2 : GOSUB *DISP.CONST : NEXT RL
2530 GOSUB *CON.INIT
2540 ' 路面
2550 GOSUB *DISP.ROMEN
2560 FOR RL = 1 TO 2
2570 *CONST1
2580 IF RL = 1 THEN RL$="左側" : SG = -1 ELSE RL$ = "右側" : SG = 1
2590 ' 排水溝
2600 GOSUB *CONST.DRAIN
2610 ' 法面1
2620 SLX2(RL,1) = DX(RL,2)
2630 SLY2(RL,1) = DY(RL,2)
2640 GOSUB *CONST.SLOPE
2650 IF SLX2(RL,0) <> 0 THEN GOTO 2720
2660 SLX1(RL,0) = 0
2670 FOR I1 = 0 TO 2
2680 SLX1(RL,I1) = SLX2(RL,I1)
2690 SLY1(RL,I1) = SLY2(RL,I1)
2700 NEXT I1
2710 GOTO 2760
2720 FOR I1 = 0 TO SLX2(RL,0)
2730 SLX1(RL,I1) = SLX2(RL,I1)
2740 SLY1(RL,I1) = SLY2(RL,I1)
2750 NEXT I1

```

```

2760 IF ENDFLG <> 1 THEN GOTO 2830
2770 WLX1(RL,0) = 0
2780 BX1(RL,0) = 0
2790 SLX2(RL,0) = 0
2800 WLX2(RL,0) = 0
2810 BX2(RL,0) = 0
2820 GOTO *CONFIN
2830 ' よう壁1
2840 WLX2(RL,1) = SLX1(RL,SLY1(RL,0))
2850 WLY2(RL,1) = SLY1(RL,SLY1(RL,0))
2860 GOSUB *CONST.WALL
2870 FOR I1 = 0 TO 2
2880 WLX1(RL,I1) = WLX2(RL,I1)
2890 WLY1(RL,I1) = WLY2(RL,I1)
2900 NEXT I1
2910 WLX1(0,RL) = WLX2(0,RL)
2920 WLX2(RL,0) = 0
2930 IF WLX1(RL,0) = 0 THEN BX1(RL,0) = 0 : GOTO 3070
2940 ' 基礎1
2950 BX2(RL,1) = WLX1(RL,2)
2960 BY2(RL,1) = WLY1(RL,2)
2970 GOSUB *CONST.BASE
2980 FOR I1 = 0 TO 2
2990 BX1(RL,I1) = BX2(RL,I1)
3000 BY1(RL,I1) = BY2(RL,I1)
3010 NEXT I1
3020 IF ENDFLG <> 1 THEN GOTO 3070
3030 SLX2(RL,0) = 0
3040 WLX2(RL,0) = 0
3050 BX2(RL,0) = 0
3060 GOTO *CONFIN
3070 ' 法面2
3080 SLX2(RL,1) = WLX1(RL,2)
3090 SLY2(RL,1) = WLY1(RL,2)
3100 GOSUB *CONST.SLOPE
3110 IF ENDFLG <> 1 THEN GOTO 3150
3120 WLX2(RL,0) = 0
3130 BX2(RL,0) = 0
3140 GOTO *CONFIN
3150 ' よう壁2
3160 WLX2(RL,1) = SLX2(RL,SLY2(RL,0))
3170 WLY2(RL,1) = SLY2(RL,SLY2(RL,0))
3180 GOSUB *CONST.WALL
3190 IF WLX2(RL,0) = 0 THEN BX2(RL,0) = 0 : GOTO 3240
3200 ' 基礎2
3210 BX2(RL,1) = WLX2(RL,2)
3220 BY2(RL,1) = WLY2(RL,2)
3230 GOSUB *CONST.BASE
3240 *CONFIN
3250 LOCATE 0,0 : PRINT RL$;"の設計が終わりました。これでよい (Y/N) ? ";
3260 N1$ = INKEY$ : IF N1$ = "" THEN GOTO 3260
3270 PRINT N1$
3280 IF N1$ = "Y" OR N1$ = "y" THEN GOTO 3320
3290 COL% = BLK% : GOSUB *DISP.CONST
3300 DX(RL,0) = 0 : SLX1(RL,0) = 0 : SLX2(RL,0) = 0 : WLX1(RL,0) = 0 :
    WLX2(RL,0) = 0 : BX1(RL,0) = 0 : BX2(RL,0) = 0
3310 GOTO *CONST1
3320 NEXT RL
3330 CRFLG = 1
3340 CRSAME = 0
3350 REC = CRNO : GOSUB *PUTFILE1
3360 RETURN
3370 '*****

```

```

3380 ' 側溝の設計
3390 '*****
3400 *CONST.DRAIN
3410 DX(RL,1) = GX(RL)
3420 DY(RL,1) = GY(RL)
3430 COMMS = RL$+"の側溝を設計しますか (Y/N) ? " : GOSUB *COMM
3440 N1$ = INKEY$ : IF N1$ = "" THEN GOTO 3440
3450 IF N1$ = "Y" OR N1$ = "y" THEN GOTO *DLIST
3460 IF N1$ = "N" OR N1$ = "n" THEN GOTO 3480
3470 GOTO *CONST.DRAIN
3480 DX(RL,0) = 0
3490 DX(RL,2) = DX(RL,1)
3500 DY(RL,2) = DY(RL,1)
3510 RETURN
3520 *DLIST
3530 IN = 1
3540 *DLIST1
3550 IE = IN + 20 : IF IE > DN% THEN IE = DN%
3560 CLS
3570 LOCATE 0,3
3580 FOR IL = 1 TO IE
3590 PRINT IL; " : ";DT$(IL)
3600 NEXT IL
3610 COMMS = "側溝のタイプを選んでください。(次のリスト=RETURN) " : GOSUB *COMM

3620 INPUT N1$
3630 IF N1$ = "" THEN IN = IE+1 :
    IF IN > DN% THEN GOTO *DLIST ELSE GOTO *DLIST1
3640 DX(RL,0) = VAL(N1$)
3650 IF DX(RL,0) = 0 THEN CLS : GOTO *CONST.DRAIN
3660 COL% = WHITE%
3670 CLS : GOSUB *DISP.DRAIN
3680 COMMS = "これでよい (Y/N) ? " : GOSUB *COMM
3690 N1$ = INKEY$ : IF N1$ = "" THEN GOTO 3690
3700 PRINT N1$
3710 IF N1$ = "Y" OR N1$ = "y" THEN GOTO 3730
3720 COL% = BLK% : GOSUB *DISP.DRAIN : GOTO *DLIST
3730 CLS
3740 GOSUB *TITLE
3750 GOSUB *SOKUTEN
3760 DX(RL,2) = DX(RL,1)+SG*KX1
3770 DY(RL,2) = DY(RL,1)+KY1
3780 RETURN
3790 '*****
3800 ' 法面の設計
3810 '*****
3820 *CONST.SLOPE
3830 COMMS = RL$+"の法面の設計をしますか (Y/N) ? " : GOSUB *COMM
3840 N$ = INKEY$ : IF N$ = "" THEN GOTO 3840
3850 PRINT N$
3860 IF N$ = "Y" OR N$ = "y" THEN GOTO 3940
3870 IF N$ = "N" OR N$ = "n" THEN GOTO 3890
3880 GOTO *CONST.SLOPE
3890 SLX2(RL,0) = 0
3900 SLY2(RL,0) = 2
3910 SLX2(RL,2) = SLX2(RL,1)
3920 SLY2(RL,2) = SLY2(RL,1)
3930 RETURN
3940 N$ = ""
3950 IS = 1
3960 WHILE N$ <> "E" AND N$ <> "e"
3970 *CS1
3980 COMMS = "Command 1.2(法面の勾配, L(ebel, E(xit) " : GOSUB *COMM

```

```

3990 INPUT N$
4000 IF N$ = "E" OR N$ = "e" THEN GOTO *CSEND
4010 IF N$ = "L" OR N$ = "l" THEN GOSUB *XLENGTH : GOTO 4160
4020 IF N$ <> "0" AND VAL(N$) = 0 THEN CLS : GOTO *CSEND
4030 SANGLE = VAL(N$)
4040 *COMCS
4050 COMM$ = "Command H(ight, R(emain hight, X( length " : GOSUB *COMM
4060 N1$ = INKEY$ : IF N1$ = "" THEN GOTO 4060
4070 PRINT N1$
4080 IF N1$ = "H" OR N1$ = "h" THEN GOSUB *HIGHT : GOTO 4160
4090 IF N1$ = "R" OR N1$ = "r" THEN GOSUB *REMAIN : GOTO 4120
4100 IF (N1$ = "X" OR N1$ = "x") AND SANGLE <> 0
      THEN GOSUB *XLENGTH : GOTO 4160

4110 GOTO *COMCS
4120 IS = IS+1
4130 SLX2(RL,IS) = KX1
4140 SLY2(RL,IS) = KY1
4150 GOTO 4190
4160 IS = IS+1
4170 SLX2(RL,IS) = KX+SLX2(RL,IS-1)
4180 SLY2(RL,IS) = KY+SLY2(RL,IS-1)
4190 LINE (SLX2(RL,IS-1),-SLY2(RL,IS-1))- (SLX2(RL,IS),-SLY2(RL,IS)),WHITE%
4200 COMM$ = "これでよいですか (Y/N) ? " : GOSUB *COMM
4210 N2$ = INKEY$ : IF N2$ = "" THEN GOTO 4210
4220 PRINT N2$
4230 IF N2$ = "Y" OR N2$ = "y" THEN GOTO 4270
4240 LINE (SLX2(RL,IS-1),-SLY2(RL,IS-1))- (SLX2(RL,IS),-SLY2(RL,IS)),BLK%
4250 IS = IS-1
4260 GOTO *CS1
4270 SLX = SLX2(RL,IS-1) : SLY = SLY2(RL,IS-1) : HLEN = 0 : GOSUB *CROSS.SLOPE

4280 IF ENDFLG = 0 THEN GOTO *CSEND
4290 COMM$ = "法面は地面と交差しました。 終了しますか (Y/N) ? "
      : GOSUB *COMM
4300 N1$ = INKEY$ : IF N1$ = "" THEN GOTO 4300
4310 PRINT N1$
4320 IF N1$ = "N" OR N1$ = "n" THEN ENDFLG = 0 : GOTO *CSEND
4330 IF N1$ = "Y" OR N1$ = "y" THEN GOTO 4350
4340 GOTO 4290
4350 LINE (SLX2(RL,IS-1),-SLY2(RL,IS-1))- (SLX2(RL,IS),-SLY2(RL,IS)),BLK%
4360 SLX2(RL,IS) = KX1 : SLY2(RL,IS) = KY1
4370 LINE (SLX2(RL,IS-1),-SLY2(RL,IS-1))- (SLX2(RL,IS),-SLY2(RL,IS)),WHITE%
4380 N$ = "E"
4390 *CSEND
4400 WEND
4410 COMM$ = RL$+"の法面の設計はこれでよいですか (Y/N) ? " : GOSUB *COMM
4420 N1$ = INKEY$ : IF N1$ = "" THEN GOTO 4420
4430 PRINT N1$
4440 IF N1$ = "N" OR N1$ = "n" THEN GOTO 4470
4450 IF N1$ = "Y" OR N1$ = "y" THEN GOTO 4520
4460 GOTO 4410
4470 PSET (SLX2(RL,1),-SLY2(RL,1))
4480 FOR II = 2 TO IS
4490 LINE -(SLX2(RL,II),-SLY2(RL,II)),BLK%
4500 NEXT II
4510 GOTO *CONST.SLOPE
4520 SLX2(RL,0) = IS
4530 SLY2(RL,0) = IS
4540 RETURN
4550 '*****
4560 ' 法高より横距の計算
4570 '*****
4580 *HIGHT

```

```

4590 COMM$ = "法高 = " : GOSUB *COMM : INPUT KY
4600 KX = KY*SANGLE
4610 RETURN
4620 '*****
4630 ' 地表よりの距離から横距、縦距の計算
4640 '*****
4650 *REMAIN
4660 COMM$ = "法肩の地表よりの距離 = " : GOSUB *COMM : INPUT HLEN
4670 SLX = SLX2(RL,IS) : SLY = SLY2(RL,IS)
4680 GOSUB *CROSS.TEST
4690 RETURN
4700 '*****
4710 ' 横距より法高の計算
4720 '*****
4730 *XLENGTH
4740 COMM$ = "法の横距 = " : GOSUB *COMM : INPUT KX
4750 IF N$ = "L" THEN KY = 0 : RETURN
4760 KY = KX/SANGLE
4770 RETURN
4780 '*****
4790 ' 地表と法面の交差
4800 '*****
4810 *CROSS.TEST
4820 II = 1
4830 WHILE X(II) <= SLX : II = II+1 : WEND
4840 IF RL = 2 THEN II = II-1
4850 SG1 = 0 : SG2 = 0
4860 WHILE SG1 = SG2 AND II < IEND AND II > 1
4870 IF N$ = "L" OR N$ = "l" THEN SG1 = SGN((Y(II)+HLEN)-SLY) : GOTO 4890
4880 SG1 = SGN((Y(II)+HLEN)-(X(II)-SLX)/SANGLE-SLY)
4890 IF RL = 1 THEN I2 = II-1
4900 IF RL = 2 THEN I2 = II+1
4910 IF N$ = "L" OR N$ = "l" THEN SG2 = SGN((Y(I2)+HLEN)-SLY) : GOTO 4930
4920 SG2 = SGN((Y(I2)+HLEN)-(X(I2)-SLX)/SANGLE-SLY)
4930 IF RL = 1 THEN II = II-1
4940 IF RL = 2 THEN II = II+1
4950 WEND
4960 IF SG1 = SG2 THEN GOTO 5050
4970 IF RL = 1 THEN II = II+1 ELSE II = II-1
4980 C1 = (Y(I2)-Y(II))/(X(I2)-X(II))
4990 D1 = Y(II)+HLEN-C1*X(II)
5000 IF N$ = "L" OR N$ = "l" THEN GOTO 5060
5010 B1 = SLY-SLX/SANGLE
5020 A1 = 1/SANGLE-C1
5030 KX1 = (D1-B1)/A1
5040 KY1 = (D1/SANGLE-B1*C1)/A1
5050 RETURN
5060 KX1 = (SLY-D1)/C1
5070 KY1 = SLY
5080 RETURN
5090 '*****
5100 ' 線分と地表の交差
5110 '*****
5120 *CROSS.SLOPE
5130 GOSUB *CROSS.TEST
5140 ENDFLG = 0
5150 IF SG1 = SG2 THEN RETURN
5160 IF RL = 2 THEN GOTO 5190
5170 IF KX1 >= SLX2(RL,IS) AND KX1 <= SLX THEN ENDFLG = 1
5180 RETURN
5190 IF KX1 <= SLX2(RL,IS) AND KX1 >= SLX THEN ENDFLG = 1
5200 RETURN
5210 '*****

```

```

5220 ' よう壁の設計
5230 '*****
5240 *CONST.WALL
5250 COMM$ = RL$+"のよう壁の設計をしますか (Y/N) ? " : GOSUB *COMM
5260 N$ = INKEY$ : IF N$ = "" THEN GOTO 5260
5270 PRINT N$
5280 IF N$ = "Y" OR N$ = "y" THEN GOTO 5350
5290 IF N$ = "N" OR N$ = "n" THEN GOTO 5310
5300 GOTO *CONST.WALL
5310 WLX2(RL,0) = 0
5320 WLX2(RL,2) = WLX2(RL,1)
5330 WLY2(RL,2) = WLY2(RL,1)
5340 RETURN
5350 *WLIST
5360 IN = 1
5370 *WLIST1
5380 IE = IN + 20 : IF IE > WN% THEN IE = WN%
5390 CLS
5400 LOCATE 0,3
5410 FOR IL = 1 TO IE
5420 PRINT IL; " : ";WT$(IL)
5430 NEXT IL
5440 COMM$ = "よう壁のタイプを選んでください。(次のリスト=RETURN) " : GOSUB *COMM
5450 INPUT N1$
5460 IF N1$ = "" THEN IN = IE+1 :
      IF IN > WN% THEN GOTO *WLIST ELSE GOTO *WLIST1
5470 WLX2(RL,0) = VAL(N1$)
5480 IF WLX2(RL,0) = 0 THEN CLS : GOTO *CONST.WALL
5490 CLS
5500 COMM$ = "前の構造物との接点は 1(始点, 2(終点 " : GOSUB *COMM : INPUT WTYPE
5510 IF WTYPE <> 1 AND WTYPE <> 2 THEN GOTO 5500
5520 WLX2(0,RL) = WTYPE
5530 WLX0% = INT(WLX2(RL,0))
5540 FM$ = RIGHT$(STR$(WLX0%),LEN(STR$(WLX0%))-1)
5550 SHIFTX = WLX2(RL,1) : SHIFTY = WLY2(RL,1)
5560 COL% = WHITE%
5570 ' 線分と地表の交差
5580 GOSUB *DISP.WALL
5590 COMM$ = "これでよいか (Y/N) ? " : GOSUB *COMM
5600 N1$ = INKEY$ : IF N1$ = "" THEN GOTO 5600
5610 PRINT N1$
5620 IF N1$ = "Y" OR N1$ = "y" THEN GOTO 5640
5630 COL% = BLK% : GOSUB *DISP.WALL : GOTO *WLIST
5640 CLS
5650 GOSUB *TITLE
5660 GOSUB *SOKUTEN
5670 WLX2(RL,2) = WLX2(RL,1)+SG*KX(ENO)
5680 IF WTYPE = 2 THEN GOTO 5710
5690 WLY2(RL,2) = WLY2(RL,1)+KY(ENO)
5700 GOTO 5720
5710 WLY2(RL,2) = WLY2(RL,1)-KY(ENO)
5720 RETURN
5730 '*****
5740 ' 基礎の設計
5750 '*****
5760 *CONST.BASE
5770 COMM$ = RL$+"の基礎の設計をしますか (Y/N) ? " : GOSUB *COMM
5780 N$ = INKEY$ : IF N$ = "" THEN GOTO 5780
5790 PRINT N$
5800 IF N$ = "Y" OR N$ = "y" THEN GOTO 5870
5810 IF N$ = "N" OR N$ = "n" THEN GOTO 5830

```

```

5820 GOTO *CONST.BASE
5830 BX2(RL,0) = 0
5840 BX2(RL,2) = BX2(RL,1)
5850 BY2(RL,2) = BY2(RL,1)
5860 RETURN
5870 *BLIST
5880 IN = 1
5890 *BLIST1
5900 IE = IN + 20 : IF IE > BN% THEN IE = BN%
5910 CLS
5920 LOCATE 0,3
5930 FOR IL = 1 TO IE
5940 PRINT IL; " : ";BT$(IL)
5950 NEXT IL
5960 COMM$ = "基礎のタイプを選んでください。(次のリスト=RETURN) " : GOSUB *COMM

5970 INPUT N1$
5980 IF N1$ = "" THEN IN = IE+1 :
      IF IN > BN% THEN GOTO *BLIST ELSE GOTO *BLIST1
5990 BX2(RL,0) = VAL(N1$)
6000 IF BX2(RL,0) = 0 THEN CLS : GOTO *CONST.BASE
6010 BX0% = INT(BX2(RL,0))
6020 FM$ = RIGHT$(STR$(BX0%),LEN(STR$(BX0%))-1)
6030 SHIFTX = BX2(RL,1) : SHIFTY = BY2(RL,1)
6040 CLS
6050 COL% = WHITE% : GOSUB *DISP.BASE
6060 COMM$ = "これでよいか (Y/N) ? " : GOSUB *COMM
6070 N1$ = INKEY$ : IF N1$ = "" THEN GOTO 6070
6080 PRINT N1$
6090 IF N1$ = "Y" OR N1$ = "y" THEN GOTO 6110
6100 COL% = BLK% : GOSUB *DISP.BASE : GOTO *BLIST
6110 CLS
6120 GOSUB *TITLE
6130 GOSUB *SOKUTEN
6140 BX2(RL,2) = BX2(RL,1)+SG*KX(ENO)
6150 BY2(RL,2) = BY2(RL,1)+KY(ENO)
6160 RETURN
6170 '*****
6180 ' X Yプロッタに出力
6190 '*****
6200 *PLOTTER
6210 COMM$ = "Command P(lotter, F(rame, E(xit) : GOSUB *COMM
6220 N$ = INKEY$ : IF N$ = "" THEN GOTO 6220
6230 IF N$ = "P" OR N$ = "p" THEN GOSUB *PLOT : GOTO *PLOTTER
6240 IF N$ = "F" OR N$ = "f" THEN GOSUB *FRAME : GOTO *PLOTTER
6250 IF N$ = "E" OR N$ = "e" THEN GOSUB *PLFIN : RETURN
6260 GOTO *PLOTTER
6270 '*****
6280 ' 地形、構造物のプロット
6290 '*****
6300 *PLOT
6310 IF CRSAME = 0 THEN GOTO 6440
6320 COMM$ = "前の測点と同じ設計です。全データをプロットしますか (Y/N) "
6330 GOSUB *COMM
6340 N$ = INKEY$ : IF N$ = "" THEN GOTO 6340
6350 IF N$ = "Y" OR N$ = "y" THEN GOTO *DISP.WAKU
6360 IF N$ = "N" OR N$ = "n" THEN GOTO 6380
6370 GOTO *PLOT
6380 X = XA
6390 IF H(11) > 0 THEN Y = AY+2.5 ELSE Y = AY-H(11)+2.5
6400 GOSUB *PLOT.SOKUTEN
6410 RETURN
6420 '****表示位置(デフォルト値)

```

```

6430 GOSUB *PMOVE
6440 IF AX > 1 THEN GOTO 6450 ELSE GOTO 6470
6450 IF AY > 19 THEN AX = 11 : AY = 9 : GOTO *DISP.WAKU
6460 AY = 18 : GOTO *DISP.WAKU
6470 IF AX < 12 THEN GOTO *DISP.WAKU
6480 IF AY > 19 THEN AX = 17 : AY = 9 : GOTO *DISP.WAKU
6490 AY = 18
6500 '***枠の表示
6510 *DISP.WAKU
6520 COL% = YELLOW% : GOSUB *SUB.WAKU
6530 '****表示位置の移動
6540 *WAKU.IDOU
6550 COMM$ = "表示位置はこれでよいですか (Y/N) ? " : GOSUB *COMM
6560 N$ = INKEY$ : IF N$ = "" THEN GOTO 6560
6570 IF N$ = "Y" OR N$ = "y" THEN GOTO *SISSER
6580 IF N$ = "N" OR N$ = "n" THEN GOTO 6600
6590 GOTO *DISP.WAKU
6600 COMM$ = "移動距離 (X,Ym) " : GOSUB *COMMS
6610 INPUT SHIFTX,SHIFTY
6620 COL% = BLK% : GOSUB *SUB.WAKU
6630 AX = AX+SHIFTX
6640 AY = AY+SHIFTY
6650 COL% = YELLOW% : GOSUB *SUB.WAKU
6660 GOTO *WAKU.IDOU
6670 '****シザリングの範囲
6680 *SISSER
6690 COMM$ = "表示する範囲を入力してください (左下 Xmin,Ymin) " : GOSUB *COMM
6700 INPUT AX1, AY1
6710 COMM$ = "表示する範囲を入力してください (右上 Xmin,Ymin) " : GOSUB *COMM
6720 INPUT AX2,AY2
6730 IF AX1 > AX2 OR AY1 > AY2
    THEN PRINT "入力誤り" : GOSUB *SLEEP : GOTO *SISSER
6740 LINE (AX1,-AY1)-(AX2,-AY2),SKY%,B
6750 COMM$ = "これでよいですか (Y/N) ? " : GOSUB *COMM
6760 N$ = INKEY$ : IF N$ = "" THEN GOTO 6760
6770 IF N$ = "Y" OR N$ = "y" THEN GOTO *XYPLOT
6780 IF N$ = "N" OR N$ = "n" THEN GOTO 6800
6790 GOTO *SISSER
6800 LINE (AX1,-AY1)-(AX2,-AY2),BLK%,B
6810 GOTO *SISSER
6820 '****プロッタ出力
6830 *XYPLOT
6840 PEN = 1 : GOSUB *PPEN
6850 '***** 測点名
6860 X = 0
6870 IF H(11) > 0 THEN Y = 2 ELSE Y = -H(11)+2
6880 GOSUB *PLOT.SOKUTEN
6890 '***** 切盛高
6900 X = 0
6910 Y = Y-.5
6920 CP% = 1.5
6930 CP$ = STR$(-INT(H(11)*100+.5)/100)
6940 IF H(11) < 0 THEN CP$ = "+" + RIGHT$(CP$, LEN(CP$)-1)
6950 PX = FNPX9(X)-LEN(CP$)*(CP%+1)*.4
6960 PY = FNPY9(Y)
6970 QP% = 0
6980 GOSUB *PALPHA.ROTATE
6990 GOSUB *PALPHA.SCALE
7000 GOSUB *PMOVE
7010 GOSUB *PPRINT
7020 '***** 地形
7030 X1 = 0 : Y1 = .5
7040 X2 = 0 : Y2 = -.5

```

```

7050 GOSUB *PLINE
7060 PEN = 2 : GOSUB *PPEN
7070 FOR I = 2 TO Y(0)
7080 X1 = X(I-1) : Y1 = Y(I-1)
7090 X2 = X(I) : Y2 = Y(I)
7100 GOSUB *SUB.SISSER
7110 IF CCODE% = -999 THEN GOTO 7130
7120 GOSUB *PLINE
7130 NEXT I
7140 PEN = 1 : GOSUB *PPEN
7150 '***** 構造物
7160 GOSUB *DISP.ROMEN
7170 FOR RL = 1 TO 2
7180 IF RL = 1 THEN SG = -1 ELSE SG = 1
7190 GOSUB *DISP.CONST
7200 NEXT RL
7210 '****プロッタ出力終わり
7220 GOSUB *PHOME
7230 RETURN
7240 '*****
7250 ' プロッタに枠をかく
7260 '*****
7270 *FRAME
7280 PEN = 2 : GOSUB *PPEN
7290 PX1 = 100 : PY1 = 100
7300 PX2 = 3800 : PY2 = 100
7310 GOSUB *PLINE1
7320 PX1 = 3800 : PY1 = 100
7330 PX2 = 3800 : PY2 = 2600
7340 GOSUB *PLINE1
7350 PX1 = 3800 : PY1 = 2600
7360 PX2 = 100 : PY2 = 2600
7370 GOSUB *PLINE1
7380 PX1 = 100 : PY1 = 2600
7390 PX2 = 100 : PY2 = 100
7400 GOSUB *PLINE1
7410 PEN = 1 : GOSUB *PPEN
7420 GOSUB *PHOME
7430 RETURN
7440 '*****
7450 ' 測点のプロット
7460 '*****
7470 *PLOT.SOKUTEN
7480 CP% = 3
7490 CP$ = H1$
7500 PX = FNPX9(X)-LEN(CP$)*(CP%+1)*.4
7510 PY = FNPY9(Y)
7520 GOSUB *PMOVE1
7530 QP% = 0
7540 GOSUB *PALPHA.SCALE
7550 GOSUB *PALPHA.ROTATE
7560 GOSUB *PPRINT
7570 RETURN
7580 '*****
7590 ' シザリング処理
7600 '*****
7610 *SUB.SISSER
7620 X = X1 : Y = Y1
7630 GOSUB *CRIP.CODE
7640 CC1% = CCODE%
7650 X = X2 : Y = Y2
7660 GOSUB *CRIP.CODE
7670 CC2% = CCODE%

```

```

7680 *SIS1
7690 IF CC1% = 0 AND CC2% = 0 THEN RETURN
7700 IF ( CC1% AND CC2% ) <> 0 THEN CCODE% = -999 : RETURN
7710 IF CC1% = 0 THEN SWAP X1,X2 : SWAP Y1,Y2 : SWAP CC1%,CC2%
7720 IF ( CC1% AND 1 ) = 0 THEN GOTO *SIS2
7730 IF X1 = X2 THEN X1 = AX1 : GOTO *SIS5
7740 Y1 = Y1+(Y2-Y1)*(AX1-X1)/(X2-X1)
7750 X1 = AX1
7760 GOTO *SIS5
7770 *SIS2
7780 IF ( CC1% AND 2 ) = 0 THEN GOTO *SIS3
7790 IF X1 = X2 THEN X1 = AX2 : GOTO *SIS5
7800 Y1 = Y1+(Y2-Y1)*(AX2-X1)/(X2-X1)
7810 X1 = AX2
7820 GOTO *SIS5
7830 *SIS3
7840 IF ( CC1% AND 4 ) = 0 THEN GOTO *SIS4
7850 IF Y1 = Y2 THEN Y1 = AY1 : GOTO *SIS5
7860 X1 = X1+(X2-X1)*(AY1-Y1)/(Y2-Y1)
7870 Y1 = AY1
7880 GOTO *SIS5
7890 *SIS4
7900 IF Y1 = Y2 THEN Y1 = AY2 : GOTO *SIS5
7910 X1 = X1+(X2-X1)*(AY2-Y1)/(Y2-Y1)
7920 Y1 = AY2
7930 *SIS5
7940 X = X1 : Y = Y1
7950 GOSUB *CRIP.CODE
7960 CC1% = CCODE%
7970 GOTO *SIS1
7980 '*****
7990 ' SUB クリッピングコード
8000 '*****
8010 *CRIP.CODE
8020 IF X >= AX1 AND X <= AX2 THEN CCODE% = 0 : GOTO *CC1
8030 IF X < AX1 THEN CCODE% = 1 : GOTO *CC1
8040 IF X > AX2 THEN CCODE% = 2
8050 *CC1
8060 IF Y >= AY1 AND Y <= AY2 THEN RETURN
8070 IF Y < AY1 THEN CCODE% = CCODE%+4 : RETURN
8080 IF Y > AY2 THEN CCODE% = CCODE%+8
8090 RETURN
8100 '*****
8110 ' プロット終了
8120 '*****
8130 *PLFIN
8140 COL% = BLK% : GOSUB *SUB.WAKU
8150 LINE (AX1,-AY1)-(AX2,-AY2),BLK%,B
8160 RETURN
8170 '*****
8180 ' 地形線と構造物の表示
8190 '*****
8200 *GRAPHIC
8210 CLS 3
8220 GOSUB *KEIKYO.IKYO
8230 GOSUB *WINDOW.SET
8240 GOSUB *WAKU
8250 GOSUB *TITLE
8260 GOSUB *CHIKAI
8270 GOSUB *DISP.ROMEN
8280 *GRAPHIC1
8290 COL% = WHITE%
8300 RLSTACK = RL

```

```

8310 FOR RL = 1 TO 2
8320 IF RL = 1 THEN SG = -1 ELSE SG = 1
8330 GOSUB *DISP.CONST
8340 NEXT RL
8350 RL = RLSTACK
8360 IF RL = 1 THEN SG = -1 ELSE SG = 1
8370 RETURN
8380 '*****
8390 ' 次のデータ表示
8400 '*****
8410 *NEXT.DATA
8420 CRNO = CRNO+1
8430 IF CRNO > CREND THEN CRNO = CREND
8440 *NEXT1
8450 REC = CRNO : GOSUB *GETFILE1
8460 REC = CRNO+1 : GOSUB *GETFILE2
8470 PNO = C(0,44)
8480 REC= PNO : GOSUB *GETFILE3
8490 GOTO *GRAPHIC
8500 '*****
8510 ' 前のデータ表示
8520 '*****
8530 *BEFORE
8540 CRNO = CRNO-1
8550 IF CRNO < 1 THEN CRNO = 1
8560 GOTO *NEXT1
8570 '*****
8580 ' 表示の拡大
8590 '*****
8600 *ZOOM
8610 POSX = POS(0) : POSY = CSRLIN
8620 CLS : LOCATE 0,24
8630 INPUT "拡大、縮小率 " ; SCALE
8640 IF SCALE = 0 THEN GOTO 8620
8650 IF SCALE >1 THEN SCALE = INT(SCALE)
8660 IF SCALE <=1 THEN SCALE = 1/INT(1/SCALE)
8670 WX1 = WX1/SCALE
8680 WY1 = WY1/SCALE
8690 WX2 = WX2/SCALE
8700 WY2 = WY2/SCALE
8710 COL% = WHITE%
8720 GOSUB *GRAPHIC
8730 LOCATE POSX,POSY
8740 RETURN
8750 '*****
8760 ' 表示のスクロール
8770 '*****
8780 *SCROLL
8790 POSX = POS(0) : POSY = CSRLIN
8800 CLS : LOCATE 0,24
8810 INPUT "スクロールする長さ X,Y "; SCRX, SCRY
8820 WX1 = WX1-SCRX
8830 WX2 = WX2-SCRX
8840 WY1 = WY1-SCRY
8850 WY2 = WY2-SCRY
8860 GOTO 8710
8870 '*****
8880 ' 表示の初期化
8890 '*****
8900 *INIT
8910 POSX = POS(0) : POSY = CSRLIN
8920 SCALE% = 1
8930 WX1 = - 64

```

```

8940 WX2 = 64
8950 WY1 = -40
8960 WY2 = 40
8970 GOTO 8710
8980 '*****
8990 ' 岩石線の表示
9000 '*****
9010 *SOIL
9020 PRINT "no programe"
9030 RETURN
9040 '*****
9050 ' 終了
9060 '*****
9070 *QUIT
9080 CLOSE #1
9090 CLOSE #2
9100 CLOSE #5
9110 SCREEN 3 : CONSOLE 0,24 : CLS 3
9120 FOR I = 1 TO 3 : KEY(I) OFF : NEXT I
9130 CHAIN "CROSS", 1000, ALL
10000 '*****
10010 ' エラー処理ルーチン
10020 '*****
10030 *ERORUTIN
10040 IF ERL = 13220 THEN PRINT "横断測定のデータがありません." : END
10050 IF ERL = 13370 THEN PRINT "縦断面のデータがありません." : END
10060 PRINT "ERR="; ERR, "ERL="; ERL
10070 PRINT "マニュアルを見て下さい."
10080 END
10090 '*****
10100 ' SUB-ROUTIN
10110 '*****
10120 '
10130 '**** SUB コマンドラインの表示
10140 '
10150 *COMM
10160 LOCATE 0,0
10170 PRINT "
"; : LOCATE 0,0 : PRINT COMM$;
10180 RETURN
10190 '
10200 '**** SUB 構造物変数の初期化
10210 '
10220 *CON.INIT
10230 ENDFLG = 0
10240 FOR RL = 1 TO 2
10250 DX(RL,0) = 0 : SLX1(RL,0) = 0 : SLX2(RL,0) = 0
10260 WLX1(RL,0) = 0 : WLX2(RL,0) = 0 : BX1(RL,0) = 0 : BX2(RL,0) = 0
10270 NEXT RL
10280 RETURN
10290 '
10300 '***SUB ウィンドウの設定
10310 '
10320 *WINDOW.SET
10330 WINDOW (WX1,WY1)-(WX2,WY2)
10340 RETURN
10350 '
10360 '***SUB 枠の表示
10370 '
10380 *WAKU
10390 WY = MAP(16,3)-MAP(0,3)
10400 IY1 = INT((WY1+WY)*2)/2+1
10410 FOR IY = IY1 TO WY2 STEP 2

```

```

10420 LINE (WX1,IY)-(WX2,IY),BLUE%,, &HAAAA
10430 NEXT IY
10440 FOR IX = WX1 TO WX2 STEP 2
10450 LINE (IX,IY1)-(IX,WY2),BLUE%,, &HAAAA
10460 NEXT IX
10470 RETURN
10480 '
10490 '**** SUB タイトルの表示
10500 '
10510 *TITLE
10520 LOCATE 0,2 : PRINT "タイトル:"; TS
10530 LOCATE 0,3 : PRINT "レコード番号 "; CRNO; SPC(20)
10540 LOCATE 20,3 : PRINT "全レコード数 "; CREND; SPC(20)
10550 RETURN
10560 '
10570 '**** SUB 経距、緯距の計算
10580 '
10590 *KEIKYO.IKYO
10600 X1 = 0
10610 Y1 = 0
10620 I1 = C(1,0)+2
10630 FOR I = 1 TO C(0,0)
10640 X1 = X1+C(1,I)
10650 Y1 = Y1+C(0,I)
10660 X(I1) = X1
10670 Y(I1) = Y1
10680 I1 = I1+1
10690 NEXT I
10700 X(C(1,0)+1) = 0
10710 Y(C(1,0)+1) = 0
10720 X1 = 0
10730 Y1 = 0
10740 I1 = C(1,0)
10750 FOR I = 21 TO C(1,0)+20
10760 X1 = X1-C(1,I)
10770 Y1 = Y1+C(0,I)
10780 X(I1) = X1
10790 Y(I1) = Y1
10800 I1 = I1-1
10810 NEXT I
10820 X(0) = C(1,0)+1
10830 Y(0) = C(0,0)+C(1,0)+1
10840 IEND = C(0,0)+C(1,0)+1
10850 LOCATE 0,15 : PRINT;
10860 ' FOR I = 0 TO IEND
10870 ' PRINT I; " "; X(I); Y(I),
10880 ' NEXT I
10890 RETURN
10900 '
10910 '**** SUB 地形線の表示
10920 '
10930 *CHIKAI
10940 GOSUB *SOKUTEN
10950 LINE (0,4)-(0,-4),WHITE%
10960 FOR I = 2 TO Y(0)
10970 LINE (X(I-1),-Y(I-1))-(X(I),-Y(I)),GRN%
10980 NEXT I
10990 RETURN
11000 '
11010 '**** SUB 測点
11020 '
11030 *SOKUTEN
11040 LX% = MAP(0,0)/8 : LY% = MAP(-8,1)/16

```

```

11050 LX% = LX%-LEN(H1$)/2
11060 IF LX% < 0 OR LX% > 80 OR LY% < 0 OR LY% > 25 THEN GOTO 11080
11070 LOCATE LX%,LY% : PRINT H1$;
11080 RETURN
11090 '
11100 '**** SUB 構造物の表示
11110 '
11120 *DISP.CONST
11130 IF DX(RL,0) = 0 THEN GOTO *SLOPE1
11140 GOSUB *DISP.DRAIN
11150 *SLOPE1
11160 IF SLX1(RL,0) = 0 THEN GOTO *WALL1
11170 X = SLX1(RL,1) : Y = SLY1(RL,1)
11180 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PMOVE : GOTO 11190
11190 PSET (X,-Y),COL%
11200 FOR I1 = 2 TO SLX1(RL,0)
11210 X = SLX1(RL,I1) : Y = SLY1(RL,I1)
11220 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PDROW : GOTO 11240
11230 LINE -(X,-Y),COL%
11240 NEXT I1
11250 *WALL1
11260 IF WLX1(RL,0) = 0 THEN GOTO *SLOPE2
11270 FM% = INT(WLX1(RL,0))
11280 FM$ = RIGHT$(STR$(FM%),LEN(STR$(FM%))-1)
11290 SHIFTX = WLX1(RL,1) : SHIFTY = WLY1(RL,1)
11300 WTYPE = WLX1(0,RL)
11310 GOSUB *DISP.WALL
11320 IF BX1(RL,0) = 0 THEN GOTO *SLOPE2
11330 FM% = INT(BX1(RL,0))
11340 FM$ = RIGHT$(STR$(FM%),LEN(STR$(FM%))-1)
11350 SHIFTX = BX1(RL,1) : SHIFTY = BY1(RL,1)
11360 GOSUB *DISP.BASE
11370 *SLOPE2
11380 IF SLX2(RL,0) = 0 THEN GOTO *WALL2
11390 X = SLX2(RL,1) : Y = SLY2(RL,1)
11400 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PMOVE : GOTO 11420
11410 PSET (X,-Y),COL%
11420 FOR I1 = 2 TO SLX2(RL,0)
11430 X = SLX2(RL,I1) : Y = SLY2(RL,I1)
11440 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PDROW : GOTO 11460
11450 LINE -(X,-Y),COL%
11460 NEXT I1
11470 *WALL2
11480 IF WLX2(RL,0) = 0 THEN GOTO *DCFIN
11490 FL$ = PASS$+"*WALL."+STR$(WLX2(RL,0))
11500 SHIFTX = WLX2(RL,1) : SHIFTY = WLY2(RL,1)
11510 WTYPE = WLX2(0,RL)
11520 GOSUB *DISP.WALL
11530 IF BX2(RL,0) = 0 THEN GOTO *SLOPE2
11540 FL$ = PASS$+"*BASE."+STR$(BX2(RL,0))
11550 SHIFTX = WLX2(RL,1) : SHIFTY = WLY2(RL,1)
11560 GOSUB *DISP.BASE
11570 *DCFIN
11580 RETURN
11590 '
11600 '**** SUB 構造物の消去
11610 '
11620 *ERASE.CONST
11630 COL% = BLK%
11640 RLSTACK = RL
11650 FOR RL = 1 TO 2
11660 IF RL = 1 THEN SG = -1 ELSE SG = 1
11670 GOSUB *DISP.CONST

```

```

11680 NEXT RL
11690 RL = RLSTACK : IF RL = 1 THEN SG = -1 ELSE SG = 1
11700 RETURN
11710 '
11720 '**** SUB 路面の表示
11730 '
11740 *DISP.ROMEN
11750 GX(0) = 0
11760 GY(0) = -H(11)
11770 FOR I1 = 1 TO 2
11780 IF I1 = 1 THEN SG1 = -1 ELSE SG1 = 1
11790 GX(I1) = GX(0)+SG1*(WIDE2+E(I1))
11800 GY(I1) = GY(0)+SG1*(E(3)*WIDE2/100)
11810 IF A1$ = "P" OR A1$ = "p"
      THEN X1 = GX(0) : Y1 = GY(0) : X2 = GX(I1) : Y2 = GY(I1) : GOSUB *PLINE
      : GOTO 11830
11820 LINE (GX(0),-GY(0))-(GX(I1),-GY(I1))
11830 NEXT I1
11840 RETURN
11850 '
11860 '**** SUB 排水溝の表示
11870 '
11880 *DISP.DRAIN
11890 DX0% = INT(DX(RL,0))
11900 FL$ = PASS$+"*DRAIN."+RIGHT$(STR$(DX0%),LEN(STR$(DX0%))-1)
11910 SHIFTX = DX(RL,1) : SHIFTY = DY(RL,1)
11920 IF A1$ = "P" OR A1$ = "p"
      THEN X = SHIFTX : Y = SHIFTY : GOSUB *PMOVE : GOTO 11940
11930 PSET (SHIFTX,-SHIFTY),COL%
11940 OPEN FL$ FOR INPUT AS #4
11950 INPUT #4, NO, ENO, CC
11960 FOR I1 = 1 TO NO
11970 INPUT #4, KX, KY, KC
11980 IF I1 = ENO THEN KX1 = KX : KY1 = KY
11990 X = SG*KX+SHIFTX : Y = KY+SHIFTY
12000 IF KC = 1 THEN GOTO 12030
12010 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PMOVE : GOTO 12050
12020 PSET (X,-Y),COL% : GOTO 12050
12030 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PDROW : GOTO 12050
12040 LINE -(X,-Y),COL% : GOTO 12050
12050 NEXT I1
12060 CLOSE #4
12070 RETURN
12080 '
12090 '**** SUB よう壁の表示
12100 '
12110 *DISP.WALL
12120 SFTX = SHIFTX : SFTY = SHIFTY
12130 FL$ = PASS$+"*WALL."+FM$
12140 OPEN FL$ FOR INPUT AS #4
12150 INPUT #4, NO, ENO, CC
12160 FOR I1 = 1 TO NO
12170 INPUT #4, KX(I1), KY(I1), KC(I1)
12180 NEXT I1
12190 IF WTYPE = 2 THEN SFTX = SHIFTX+SG*KX(ENO) : SFTY = SHIFTY-KY(ENO)
12200 X = SFTX : Y = SFTY
12210 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PMOVE : GOTO 12230
12220 PSET (X,-Y),COL%
12230 FOR I1 = 1 TO NO
12240 IF WTYPE = 2 THEN KX1 = -SG*KX(I1)+SFTX
      ELSE KX1 = SG*KX(I1)+SFTX
12250 KY1 = KY(I1)+SFTY
12260 X = KX1 : Y = KY1

```

```

12270 IF KC(I1) = 1 THEN GOTO 12300
12280 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PMOVE : GOTO 12320
12290 PSET (X,-Y),COL% : GOTO 12320
12300 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PDROW : GOTO 12320
12310 LINE -(X,-Y),COL% : GOTO 12320
12320 NEXT I1
12330 CLOSE #4
12340 RETURN
12350 '
12360 '**** SUB 基礎の表示
12370 '
12380 *DISP.BASE
12390 IF A1$ = "P" OR A1$ = "p"
      THEN X = SHIFTX : Y = SHIFTY : GOSUB *PMOVE : GOTO 12410
12400 PSET (SHIFTX,-SHIFTY),COL%
12410 FL$ = PASS$+"¥BASE."+FM$
12420 OPEN FL$ FOR INPUT AS #4
12430 INPUT #4, NO, ENO, CC
12440 FOR II = 1 TO NO
12450 INPUT #4, KX, KY, KC
12460 IF WTYPE = 2 THEN KX = -SG*KX+SHIFTX ELSE KX = SG*KX+SHIFTX
12470 KY = KY+SHIFTY
12480 X = KX : Y = KY
12490 IF KC = 1 THEN GOTO 12520
12500 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PMOVE : GOTO 12540
12510 PSET (X,-Y),COL% : GOTO 12540
12520 IF A1$ = "P" OR A1$ = "p" THEN GOSUB *PDROW : GOTO 12540
12530 LINE -(X,-Y),COL% : GOTO 12540
12540 NEXT II
12550 CLOSE #4
12560 RETURN
12570 '
12580 '****SUB 表示の中止
12590 '
12600 *SLEEP
12610 FOR SLP = 0 TO 5000 : NEXT SLP
12620 RETURN
12630 '
12640 '**** SUB プロッタの表示位置を C R T にかく
12650 '
12660 *SUB.WAKU
12670 X1 = -AX+1
12680 Y1 = -AY+1
12690 X2 = X1+38
12700 Y2 = Y1+26
12710 LINE (X1,-Y1)-(X2,-Y2),COL%,B
12720 RETURN
12730 '*****
12740 ' プロッタ サブルーチン (ROLAND DXY-980)
12750 '*****
12760 *PMOVE
12770 PX = FNPX9(X)
12780 PY = FNPY9(Y)
12790 *PMOVE1
12800 PRINT #5, "M"; PX, PY
12810 RETURN
12820 *PDROW
12830 PX = FNPX9(X)
12840 PY = FNPY9(Y)
12850 *PDROW1
12860 PRINT #5, "D"; PX, PY
12870 RETURN
12880 *PLINE

```

```

12890 PX1 = FNPX9(X1)
12900 PY1 = FNPY9(Y1)
12910 PX2 = FNPX9(X2)
12920 PY2 = FNPY9(Y2)
12930 *PLINE1
12940 PX = PX1 : PY = PY1 : GOSUB *PMOVE1
12950 PX = PX2 : PY = PY2 : GOSUB *PDROW1
12960 RETURN
12970 *PALPHA.SCALE
12980 PRINT #5, "S"; CP%
12990 RETURN
13000 *PALPHA.ROTATE
13010 PRINT #5, "Q"; QP%
13020 RETURN
13030 *PPRINT
13040 PRINT #5, "U 1"
13050 PRINT #5, "P"; CP$
13060 RETURN
13070 *PPEN
13080 PRINT #5, "J", PEN
13090 RETURN
13100 *PHOME
13110 PRINT #5, "H"
13120 RETURN
13130 '*****
13140 ' SUB データファイルの読み込みと書き込み
13150 '*****
13160 '****横断地形ファイル
13170 *GETFILE2
13180 ON ERROR GOTO *ERORUTIN
13190 I1 = 0
13200 J1 = 0
13210 FOR RR = 0 TO 5
13220 GET #2, (REC-1)*6+RR+1
13230 IF RR = 3 THEN I1 = 1 : J1 = 0
13240 FOR JJ = 0 TO 14
13250 C(I1,J1+JJ) = CVS(C$(JJ))
13260 NEXT JJ
13270 J1 = J1+15
13280 NEXT RR
13290 ON ERROR GOTO 0
13300 E(1) = C(0,41)
13310 E(2) = C(0,42)
13320 E(3) = C(0,43)
13330 RETURN
13340 '****縦断図ファイル
13350 *GETFILE3
13360 ON ERROR GOTO *ERORUTIN
13370 GET #3, REC
13380 H1$ = HH$
13390 FOR I1 = 0 TO 12
13400 H(I1) = CVS(H$(I1))
13410 NEXT I1
13420 ON ERROR GOTO 0
13430 RETURN
13440 '****横断図ファイル
13450 *PUTFILE1
13460 'offset = 1
13470 LSET K$(0) = MKS$(CRFLG)
13480 LSET K$(1) = MKS$(CRSAME)
13490 LSET K$(2) = MKS$(0)
13500 I2 = 0
13510 FOR I1 = 3 TO 5

```

```

13520 LSET K$(I1) = MKS$(GX(I2))
13530 I2 = I2+1
13540 NEXT I1
13550 I2 = 0
13560 FOR I1 = 6 TO 8
13570 LSET K$(I1) = MKS$(GY(I2))
13580 I2 = I2+1
13590 NEXT I1
13600 I2 = 1 : J2 = 0
13610 FOR I1 = 9 TO 14
13620 LSET K$(I1) = MKS$(DX(I2,J2))
13630 J2 = J2+1
13640 IF J2 = 3 THEN I2 = 2 : J2 = 0
13650 NEXT I1
13660 PUT #1, (REC-1)*13+1
13670 'offset = 2
13680 I2 = 1 : J2 = 0
13690 FOR I1 = 0 TO 5
13700 LSET K$(I1) = MKS$(DY(I2,J2))
13710 J2 = J2+1
13720 IF J2 = 3 THEN I2 = 2 : J2 = 0
13730 NEXT I1
13740 J2 = 0
13750 FOR I1 = 6 TO 8
13760 LSET K$(I1) = MKS$(WLX1(0,J2))
13770 J2 = J2+1
13780 NEXT I1
13790 I2 = 1 : J2 = 0
13800 FOR I1 = 9 TO 14
13810 LSET K$(I1) = MKS$(WLX1(I2,J2))
13820 J2 = J2+1
13830 IF J2 = 3 THEN I2 = 2 : J2 = 0
13840 NEXT I1
13850 PUT #1, (REC-1)*13 +2
13860 'offset = 3
13870 I2 = 1 : J2 = 0
13880 FOR I1 = 0 TO 5
13890 LSET K$(I1) = MKS$(WLY1(I2,J2))
13900 J2 = J2+1
13910 IF J2 = 3 THEN I2 = 2 : J2 = 0
13920 NEXT I1
13930 J2 = 0
13940 FOR I1 = 6 TO 8
13950 LSET K$(I1) = MKS$(WLX2(0,J2))
13960 J2 = J2+1
13970 NEXT I1
13980 I2 = 1 : J2 = 0
13990 FOR I1 = 9 TO 14
14000 LSET K$(I1) = MKS$(WLX2(I2,J2))
14010 J2 = J2+1
14020 IF J2 = 3 THEN I2 = 2 : J2 = 0
14030 NEXT I1
14040 PUT #1, (REC-1)*13+3
14050 'offset = 4
14060 I2 = 1 : J2 = 0
14070 FOR I1 = 0 TO 5
14080 LSET K$(I1) = MKS$(WLY2(I2,J2))
14090 J2 = J2+1
14100 IF J2 = 3 THEN I2 = 2 : J2 = 0
14110 NEXT I1
14120 I2 = 1 : J2 = 0
14130 FOR I1 = 6 TO 11
14140 LSET K$(I1) = MKS$(BX1(I2,J2))

```

```

14150 J2 = J2+1
14160 IF J2 = 3 THEN I2 = 2 : J2 = 0
14170 NEXT I1
14180 J2 = 0
14190 FOR I1 = 12 TO 14
14200 LSET K$(I1) = MKS$(BY1(1,J2))
14210 J2 = J2+1
14220 NEXT I1
14230 PUT #1, (REC-1)*13+4
14240 'offset = 5
14250 J2 = 0
14260 FOR I1 = 0 TO 2
14270 LSET K$(I1) = MKS$(BY1(2,J2))
14280 J2 = J2+1
14290 NEXT I1
14300 I2 = 1 : J2 = 0
14310 FOR I1 = 3 TO 8
14320 LSET K$(I1) = MKS$(BX2(I2,J2))
14330 J2 = J2+1
14340 IF J2 = 3 THEN I2 = 2 : J2 = 0
14350 NEXT I1
14360 I2 = 1 : J2 = 0
14370 FOR I1 = 9 TO 14
14380 LSET K$(I1) = MKS$(BY2(I2,J2))
14390 J2 = J2+1
14400 IF J2 = 3 THEN I2 = 2 : J2 = 0
14410 NEXT I1
14420 PUT #1, (REC-1)*13+5
14430 'offset = 6
14440 FOR I1 = 0 TO 14
14450 LSET K$(I1) = MKS$(SLX1(1,I1))
14460 NEXT I1
14470 PUT #1, (REC-1)*13+6
14480 'offset = 7
14490 FOR I1 = 0 TO 14
14500 LSET K$(I1) = MKS$(SLX1(2,I1))
14510 NEXT I1
14520 PUT #1, (REC-1)*13+7
14530 'offset = 8
14540 FOR I1 = 0 TO 14
14550 LSET K$(I1) = MKS$(SLY1(1,I1))
14560 NEXT I1
14570 PUT #1, (REC-1)*13+8
14580 'offset = 9
14590 FOR I1 = 0 TO 14
14600 LSET K$(I1) = MKS$(SLY1(2,I1))
14610 NEXT I1
14620 PUT #1, (REC-1)*13+9
14630 'offset = 10
14640 FOR I1 = 0 TO 14
14650 LSET K$(I1) = MKS$(SLX2(1,I1))
14660 NEXT I1
14670 PUT #1, (REC-1)*13+10
14680 'offset = 11
14690 FOR I1 = 0 TO 14
14700 LSET K$(I1) = MKS$(SLX2(2,I1))
14710 NEXT I1
14720 PUT #1, (REC-1)*13+11
14730 'offset = 12
14740 FOR I1 = 0 TO 14
14750 LSET K$(I1) = MKS$(SLY2(1,I1))
14760 NEXT I1
14770 PUT #1, (REC-1)*13+12

```

```

14780 'offset = 13
14790 FOR I1 = 0 TO 14
14800   LSET K$(I1) = MKS$(SLY2(2,I1))
14810 NEXT I1
14820 PUT #1, (REC-1)*13+13
14830 RETURN
14840 '
14850 *GETFILE1
14860 'offset = 1
14870 GET #1, (REC-1)*13+1
14880 CRFLG = CVS(K$(0))
14890 CRSAME = CVS(K$(1))
14900 I2 = 0
14910 FOR I1 = 3 TO 5
14920   GX(I2) = CVS(K$(I1))
14930   I2 = I2+1
14940 NEXT I1
14950 I2 = 0
14960 FOR I1 = 6 TO 8
14970   GY(I2) = CVS(K$(I1))
14980   I2 = I2+1
14990 NEXT I1
15000 I2 = 1 : J2 = 0
15010 FOR I1 = 9 TO 14
15020   DX(I2,J2) = CVS(K$(I1))
15030   J2 = J2+1
15040   IF J2 = 3 THEN I2 = 2 : J2 = 0
15050 NEXT I1
15060 'offset = 2
15070 GET #1, (REC-1)*13+2
15080 I2 = 1 : J2 = 0
15090 FOR I1 = 0 TO 5
15100   DY(I2,J2) = CVS(K$(I1))
15110   J2 = J2+1
15120   IF J2 = 3 THEN I2 = 2 : J2 = 0
15130 NEXT I1
15140 J2 = 0
15150 FOR I1 = 6 TO 8
15160   WLX1(0,J2) = CVS(K$(I1))
15170   J2 = J2+1
15180 NEXT I1
15190 I2 = 1 : J2 = 0
15200 FOR I1 = 9 TO 14
15210   WLX1(I2,J2) = CVS(K$(I1))
15220   J2 = J2+1
15230   IF J2 = 3 THEN I2 = 2 : J2 = 0
15240 NEXT I1
15250 'offset = 3
15260 GET #1, (REC-1)*13+3
15270 I2 = 1 : J2 = 0
15280 FOR I1 = 0 TO 5
15290   WLY1(I2,J2) = CVS(K$(I1))
15300   J2 = J2+1
15310   IF J2 = 3 THEN I2 = 2 : J2 = 0
15320 NEXT I1
15330 J2 = 0
15340 FOR I1 = 6 TO 8
15350   WLX2(0,J2) = CVS(K$(I1))
15360   J2 = J2+1
15370 NEXT I1
15380 I2 = 1 : J2 = 0
15390 FOR I1 = 9 TO 14
15400   WLX2(I2,J2) = CVS(K$(I1))

```

```

15410   J2 = J2+1
15420   IF J2 = 3 THEN I2 = 2 : J2 = 0
15430 NEXT I1
15440 'offset = 4
15450 GET #1, (REC-1)*13+4
15460 I2 = 1 : J2 = 0
15470 FOR I1 = 0 TO 5
15480   WLY2(I2,J2) = CVS(K$(I1))
15490   J2 = J2+1
15500   IF J2 = 3 THEN I2 = 2 : J2 = 0
15510 NEXT I1
15520 I2 = 1 : J2 = 0
15530 FOR I1 = 6 TO 11
15540   BX1(I2,J2) = CVS(K$(I1))
15550   J2 = J2+1
15560   IF J2 = 3 THEN I2 = 2 : J2 = 0
15570 NEXT I1
15580 J2 = 0
15590 FOR I1 = 12 TO 14
15600   BY1(1,J2) = CVS(K$(I1))
15610   J2 = J2+1
15620 NEXT I1
15630 'offset = 5
15640 GET #1, (REC-1)*13+5
15650 J2 = 0
15660 FOR I1 = 0 TO 2
15670   BY1(2,J2) = CVS(K$(I1))
15680   J2 = J2+1
15690 NEXT I1
15700 I2 = 1 : J2 = 0
15710 FOR I1 = 3 TO 8
15720   BX2(I2,J2) = CVS(K$(I1))
15730   J2 = J2+1
15740   IF J2 = 3 THEN I2 = 2 : J2 = 0
15750 NEXT I1
15760 I2 = 1 : J2 = 0
15770 FOR I1 = 9 TO 14
15780   BY2(I2,J2) = CVS(K$(I1))
15790   J2 = J2+1
15800   IF J2 = 3 THEN I2 = 2 : J2 = 0
15810 NEXT I1
15820 'offset = 6
15830 GET #1, (REC-1)*13+6
15840 FOR I1 = 0 TO 14
15850   SLX1(1,I1) = CVS(K$(I1))
15860 NEXT I1
15870 'offset = 7
15880 GET #1, (REC-1)*13+7
15890 FOR I1 = 0 TO 14
15900   SLX1(2,I1) = CVS(K$(I1))
15910 NEXT I1
15920 'offset = 8
15930 GET #1, (REC-1)*13+8
15940 FOR I1 = 0 TO 14
15950   SLY1(1,I1) = CVS(K$(I1))
15960 NEXT I1
15970 'offset = 9
15980 GET #1, (REC-1)*13+9
15990 FOR I1 = 0 TO 14
16000   SLY1(2,I1) = CVS(K$(I1))
16010 NEXT I1
16020 'offset = 10
16030 GET #1, (REC-1)*13+10

```

```

16040 FOR I1 = 0 TO 14
16050   SLX2(1,I1) = CVS(K$(I1))
16060 NEXT I1
16070 'offset = 11
16080 GET #1, (REC-1)*13+11
16090 FOR I1 = 0 TO 14
16100   SLX2(2,I1) = CVS(K$(I1))
16110 NEXT I1
16120 'offset = 12
16130 GET #1, (REC-1)*13+12
16140 FOR I1 = 0 TO 14
16150   SLY2(1,I1) = CVS(K$(I1))
16160 NEXT I1
16170 'offset = 13
16180 GET #1, (REC-1)*13+13
16190 FOR I1 = 0 TO 14
16200   SLY2(2,I1) = CVS(K$(I1))
16210 NEXT I1
16220 RETURN

```

```

10 '*****
20 '          ( FOREST ROAD DROWING SYSTEM )
30 '          FILE NAME = "CR40"
40 '          CROSS PROFILE DROWING SUB-PROGRAM NO.4
50 '          CREATED BY A.FUKUDA
60 '*****
1000 *SUB4 : 'サブプログラム NO. 4 のエントリーポイント
1010 CLS
1020 PRINT "          <<< 構造物ファイルの作成 >>>" : PRINT
1030 DEFINT I,J,P,R
1040 WX1 = -2
1050 WX2 = 4.4
1060 WY1 = -3
1070 WY2 = 1
1080 BLK% = 0
1090 BLUE% = 1
1100 RED% = 2
1110 '*****
1120 '   バス名の入力
1130 '*****
1140 *PASSNAME
1150 ON ERROR GOTO 0
1160 INPUT "構造物ファイルのバス名 "; PASS$
1170 '*****
1180 '   構造物ファイルのオープンとデータの読み込み
1190 '*****
1200 ON ERROR GOTO *ERDRAIN
1210 OPEN PASS$+"¥DRAIN.DIR" FOR INPUT AS #4
1220 GOTO 1260
1230   *ERDRAIN
1240   IF ERR = 53 THEN DN% = 0 : RESUME *WALL
1250   PRINT "ERR=";ERR,"ERL=";ERL : RESUME *PASSNAME
1260 INPUT #4, DN%
1270 FOR I = 1 TO DN%
1280   INPUT #4, DT$(I)
1290 NEXT I
1300 CLOSE #4
1310 *WALL
1320 ON ERROR GOTO *ERWALL
1330 OPEN PASS$+"¥WALL.DIR" FOR INPUT AS #4
1340 GOTO 1380
1350   *ERWALL
1360   IF ERR = 53 THEN WN% = 0 : RESUME *BASE
1370   PRINT "ERR=";ERR,"ERL=";ERL : RESUME *PASSNAME
1380 INPUT #4, WN%
1390 FOR I = 1 TO WN%
1400   INPUT #4, WT$(I)
1410 NEXT I
1420 CLOSE #4
1430 *BASE
1440 ON ERROR GOTO *ERBASE
1450 OPEN PASS$+"¥BASE.DIR" FOR INPUT AS #4
1460 GOTO 1500
1470   *ERBASE
1480   IF ERR = 53 THEN BN% = 0 : RESUME *MENU
1490   PRINT "ERR=";ERR,"ERL=";ERL : RESUME *PASSNAME
1500 INPUT #4, BN%
1510 FOR I = 1 TO BN%
1520   INPUT #4, BT$(I)
1530 NEXT I
1540 CLOSE #4
1550 '*****
1560 '   構造物データファイル作成   メイン

```

```

1570 '*****
1580 *MENU
1590 ON ERROR GOTO 0
1600 GOSUB *SCRN.SET
1610 GOSUB *MESH
1620 GOSUB *WINDOW.SET
1630 *MENU1
1640 CLS : LOCATE 0,0
1650 PRINT "Command I(nput data, D(isplay data, Q(uit ";
1660 N$ = INKEY$ : IF N$="" THEN GOTO 1660
1670 IF N$ = "I" OR N$ = "i" THEN GOTO *INPDATA
1680 IF N$ = "D" OR N$ = "d" THEN GOTO *DISPLAY
1690 IF N$ = "Q" OR N$ = "q" THEN GOTO *QUIT
1700 GOTO *MENU1
1710 '*****
1720 ' データの入力
1730 '*****
1740 *INPDATA
1750 IP = 1 : 'データのポイント
1760 KX(0) = 0
1770 KY(0) = 0
1780 CLS : INPUT "構造物の種類: 1)側溝 2)よう壁 3)基礎 "; CTYPE%
1790 IF CTYPE% < 1 OR CTYPE% > 3 THEN GOTO *INPDATA
1800 ON CTYPE% GOTO 1810, 1850, 1890
1810 FC1$ = PASS$+"¥DRAIN."
1820 FC2$ = PASS$+"¥DRAIN.DIR"
1830 NS% = DN%
1840 GOTO 1920
1850 FC1$ = PASS$+"¥WALL."
1860 FC2$ = PASS$+"¥WALL.DIR"
1870 NS% = WN%
1880 GOTO 1920
1890 FC1$ = PASS$+"¥BASE."
1900 FC2$ = PASS$+"¥BASE.DIR"
1910 NS% = BN%
1920 LINE INPUT "構造物の名前 "; T$
1930 PRINT "タイプ番号 (現在の最大タイプ番号 =";NS%;") ";
1940 INPUT NS1%
1950 IF NS1% >= NS%+1 THEN NS1% = NS%+1 : NS% = NS%+1 : GOTO 2000
1960 IF NS1% < 1 THEN GOTO 1930
1970 PRINT NS1%;"と置き換えてよいか (Y/N) ?";
1980 N$ = INKEY$ : IF N$ = "" THEN GOTO 1980
1990 IF N$ <> "Y" AND N$ <> "y" THEN GOTO 1940
2000 LOCATE 0,3 : PRINT "名前 "; T$
2010 LOCATE 0,4 : PRINT "タイプ番号"; NS1%
2020 *INPDATA1
2030 CLS
2040 LOCATE 0,0 : PRINT "Command 123(X data, Z(oom, S(croll, I(nitial, E(xit ";
2050 INPUT N$
2060 IF N$ = "Z" OR N$ = "z" THEN GOSUB *ZOOM : GOTO *INPDATA1
2070 IF N$ = "S" OR N$ = "s" THEN GOSUB *SCROLL : GOTO *INPDATA1
2080 IF N$ = "I" OR N$ = "i" THEN GOSUB *INIT : GOTO *INPDATA1
2090 IF N$ = "E" OR N$ = "e" THEN GOTO 2130
2100 X = VAL(N$)
2110 IF X = 0 AND N$ <> "0" THEN GOTO *INPDATA1
2120 GOSUB *XYDATA : GOTO *INPDATA1
2130 GOSUB *INPFIN
2140 IF ER% = 1 THEN
    GOSUB *SCRN.SET : GOSUB *MESH : GOSUB *WINDOW.SET : GOTO *INPDATA
2150 IF ER% = 2 THEN GOTO *INPDATA1
2160 GOTO *MENU
2170 '*****

```

```

2180 ' 画面のズーム表示
2190 '*****
2200 *ZOOM
2210 CLS : LOCATE 0,0
2220 INPUT "ズーム率 (1 = 等倍) "; ZM
2230 IF ZM < 0 THEN GOTO *ZOOM
2240 IF ZM > 1 THEN ZM = INT(ZM)
    ELSE ZM = 1/INT(1/ZM)
2250 WX1 = WX1/ZM
2260 WX2 = WX2/ZM
2270 WY1 = WY1/ZM
2280 WY2 = WY2/ZM
2290 CLS 3
2300 GOSUB *MESH
2310 GOSUB *WINDOW.SET
2320 GOSUB *GRAPHIC
2330 RETURN
2340 '*****
2350 ' 画面のスクロール
2360 '*****
2370 *SCROLL
2380 CLS : LOCATE 0,0
2390 INPUT "スクロールする長さ X = "; SCR X
2400 INPUT "スクロールする長さ Y = "; SCR Y
2410 WX1 = WX1-SCR X
2420 WX2 = WX2-SCR X
2430 WY1 = WY1+SCR Y
2440 WY2 = WY2+SCR Y
2450 GOTO 2290
2460 '*****
2470 ' 画面設定の初期化
2480 '*****
2490 *INIT
2500 WX1 = -2
2510 WX2 = 4.4
2520 WY1 = -3
2530 WY2 = 1
2540 GOTO 2290
2550 '*****
2560 ' データ入力
2570 '*****
2580 *XYDATA
2590 CLS : LOCATE 0,0
2600 INPUT "Y座標 = "; Y
2610 ON ERROR GOTO 2650
2620 RAD! = MAP(3,2)-MAP(0,2) : CIRCLE (X,-Y),RAD!,RED%
2630 ON ERROR GOTO 0
2640 GOTO 2680
2650 IF ERR= 6 THEN PRINT "データが不適当です。" : GOSUB *SLEEP : RESUME 2670
2660 PRINT"ERR=";ERR,"ERL=";ERL : END
2670 RETURN
2680 CLS : PRINT "X = "; X ; " Y = "; Y ; " これでよいか (Y/N) ?";
2690 N$ = INKEY$ : IF N$ = "" THEN GOTO 2690
2700 PRINT N$
2710 IF N$ <> "Y" AND N$ <> "y" THEN CIRCLE (X,-Y),RAD!,BLK% : RETURN
2720 KX(IP) = X : KY(IP) = Y
2730 PRINT "前の点と接続しますか (Y/N) ";
2740 N$ = INKEY$ : IF N$ = "" THEN GOTO 2740
2750 PRINT N$
2760 CIRCLE (X,-Y),RAD!,BLK%
2770 IF N$ = "N" OR N$ = "n" THEN PSET (X,-Y) : KC(IP) = 0 : IP = IP +1 : RETURN
2780 IF N$ = "Y" OR N$ = "y" THEN LINE (KX(IP-1),-KY(IP-1))- (KX(IP),-KY(IP)) : K

```

```

C(IP) = 1 : IP = IP+1 : RETURN
2790 GOTO 2730
2800 '*****
2810 ' データ入力終了
2820 '*****
2830 *INPFIN
2840 ER% = 0
2850 IF IP = 0 THEN PRINT "データが入力されていません。" : GOTO 2900
2860 CLS : PRINT "この構造物のデータの入力を終わります。よろしいですか (Y/N) ?";

2870 N$ = INKEY$ : IF N$ = "" THEN GOTO 2870
2880 PRINT N$
2890 IF N$ = "Y" OR N$ = "y" THEN GOTO 2940
2900 CLS : PRINT "Command S(このデータを取り消して初めから, K(データの入力を
継続 ";
2910 N$ = INKEY$ : IF N$ = "" THEN GOTO 2910
2920 IF N$ = "S" OR N$ = "s" THEN ER% = 1 : RETURN
2930 IF N$ = "K" OR N$ = "k" THEN ER% = 2 : RETURN
2940 INPUT "次の構造物(または法面)との接続点番号は "; IS
2950 IF IS > IP-1 OR IS < 0 THEN GOTO 2940
2960 RAD! = MAP(5,2)-MAP(0,2) : CIRCLE (KX(IS),-KY(IS)),RAD!,RED%
2970 PRINT "この点でよいか (Y/N) ";
2980 N$ = INKEY$ : IF N$ = "" THEN GOTO 2980
2990 PRINT N$
3000 IF N$ = "Y" OR N$ = "y" THEN GOTO 3010
ELSE GOTO 2940
3010 KX(0) = IP-1 : KY(0) = IS
3020 FCON$ = FC1$+RIGHT$(STR$(NS%),LEN(STR$(NS%))-1)
3030 OPEN FCON$ FOR OUTPUT AS #4
3040 FOR I = 0 TO KX(0)
3050 WRITE #4 , KX(I), KY(I), KC(I)
3060 NEXT I
3070 CLOSE #4
3080 OPEN FC2$ FOR OUTPUT AS #4
3090 ON CTYPE% GOTO 3100, 3180, 3260
3100 DN% = NS%
3110 DT$(NS1%) = T$
3120 WRITE #4, DN%
3130 FOR I = 1 TO DN%
3140 WRITE #4, DT$(I)
3150 NEXT I
3160 CLOSE #4
3170 RETURN
3180 WN% = NS%
3190 WT$(NS1%) = T$
3200 WRITE #4, WN%
3210 FOR I = 1 TO WN%
3220 WRITE #4, WT$(I)
3230 NEXT I
3240 CLOSE #4
3250 RETURN
3260 BN% = NS%
3270 BT$(NS1%) = T$
3280 WRITE #4, BN%
3290 FOR I = 1 TO BN%
3300 WRITE #4, BT$(I)
3310 NEXT I
3320 CLOSE #4
3330 RETURN
3340 '*****
3350 ' 構造物ファイルの表示
3360 '*****
3370 *DISPLAY

```

```

3380 CLS : INPUT "表示する構造物の種類: 1)側溝 2)よう壁 3)基礎 "; CTYPE%
3390 IF CTYPE% < 1 OR CTYPE% > 3 THEN GOTO *DISPLAY
3400 ON CTYPE% GOTO 3420,3470,3520
3410 '*** 側溝
3420 IF DN% = 0 THEN PRINT "データがありません。" : GOSUB *SLEEP : GOTO *MENU
3430 FC1$ = PASS$+"¥DRAIN."
3440 NS% = DN%
3450 GOTO 3550
3460 '*** よう壁
3470 IF WN% = 0 THEN PRINT "データがありません。" : GOSUB *SLEEP : GOTO *MENU
3480 FC1$ = PASS$+"¥WALL."
3490 NS% = WN%
3500 GOTO 3550
3510 '*** 基礎
3520 IF BN% = 0 THEN PRINT "データがありません。" : GOSUB *SLEEP : GOTO *MENU
3530 FC1$ = PASS$+"¥BASE."
3540 NS% = BN%
3550 PRINT "タイプ番号 ( 1 -";NS%;") = ";
3560 INPUT NS1%
3570 IF NS1% > NS% THEN GOTO 3550
3580 IF CTYPE% = 1 THEN T$ = DT$(NS1%)
ELSE IF CTYPE% = 2 THEN T$ = WT$(NS1%)
ELSE T$ = BT$(NS1%)
3590 FCON$ = FC1$+RIGHT$(STR$(NS1%),LEN(STR$(NS1%))-1)
3600 OPEN FCON$ FOR INPUT AS #4
3610 I = 0
3620 WHILE NOT(EOF(4))
3630 INPUT #4 , KX(I), KY(I), KC(I)
3640 I = I+1
3650 WEND
3660 CLOSE #4
3670 IP = KX(0)
3680 GOSUB *GRAPHIC
3690 CLS : LOCATE 0,0 : PRINT "Command Z(zoom, S(scroll, I(nitial, E(xit ";
3700 INPUT N$
3710 IF N$ = "Z" OR N$ = "z" THEN GOSUB *ZOOM : GOTO 3690
3720 IF N$ = "S" OR N$ = "s" THEN GOSUB *SCROLL : GOTO 3690
3730 IF N$ = "I" OR N$ = "i" THEN GOSUB *INIT : GOTO 3690
3740 IF N$ = "E" OR N$ = "e" THEN GOTO *MENU
3750 GOTO 3690
3760 '*****
3770 ' 終了
3780 '*****
3790 *QUIT
3800 CLOSE #4
3810 CONSOLE 0,25 : CLS 3 : CONSOLE 0,24
3820 CHAIN "CROSS", 1000, ALL
10000 '*****
10010 ' SUB-ROUTIN
10020 '*****
10030 '
10040 '***SUB ウィンドウの設定
10050 '
10060 *WINDOW.SET
10070 WINDOW (WX1,WY1)-(WX2,WY2)
10080 RETURN
10090 '
10100 '***SUB メッシュの表示
10110 '
10120 *MESH
10130 WINDOW (0,0)-(639,399)
10140 SX1% = 0
10150 SX2% = 600

```

```

10160 FOR SY% = 49 TO 399 STEP 50
10170   LINE (SX1%,SY%)-(SX2%,SY%),BLUE%, ,&HAAAA
10180 NEXT SY%
10190 SY1% = 49
10200 SY2% = 399
10210 FOR SX% = 0 TO 600 STEP 50
10220   LINE (SX%,SY1%)-(SX%,SY2%),BLUE%, ,&HAAAA
10230 NEXT SX%
10240 LINE (550,367)-(600,367)
10250 LINE (550,367)-(558,359)
10260 LINE (550,367)-(558,375)
10270 LINE (600,367)-(592,359)
10280 LINE (600,367)-(592,375)
10290 LOCATE 69,24 : PRINT USING"###.##m"; (WY2-WY1)/8
10300 GOSUB *WINDOW.SET
10310 LINE (0,WY1)-(0,WY2),BLUE%
10320 LINE (WX1,0)-(WX2,0),BLUE%
10330 RETURN
10340 '
10350 '*** SUB 構造物の表示
10360 '
10370 *GRAPHIC
10380 LOCATE 0,3 : PRINT "名前          "; T$
10390 LOCATE 0,4 : PRINT "タイプ番号"; NS1%
10400 PSET (0,0)
10410 FOR II = 1 TO IP
10420   IF KC(II) = 1 THEN LINE -(KX(II),-KY(II))
10430   IF KC(II) = 0 THEN PSET (KX(II),-KY(II))
10440 NEXT II
10450 RETURN
10460 '
10470 '**** SUB スクリーンのクリアーと設定
10480 '
10490 *SCRN.SET
10500 CONSOLE 0,25 : CLS 3
10510 CONSOLE 0,2 : SCREEN 3,0,0,1
10520 RETURN
10530 '
10540 '**** SUB 表示の停止
10550 '
10560 *SLEEP
10570 FOR II = 0 TO 5000 : NEXT II
10580 RETURN

```