

# 取扱説明書

NA-29/NA-29E用  
RS-232-Cインタフェース

 **リオン株式会社**

## 当社製品と国際規格及びJISにおける量記号の表記

量記号はISO 1996, 3891、IEC Pub. 804及びJIS Z 8202, 8731より抜粋しました。

当社製品の表記		名 称	周波数補正回路	ISOの表記	IECの表記	JISの表記
$L_p$		音圧レベル	FLAT特性	$L_p$	—	$L_p$
		騒音レベル	A特性	$L_{pA}$	—	$L_A$
		音圧レベル	C特性	$L_{pC}$	—	—
$L_{eq}$		等価騒音レベル	A特性	$L_{Aeq,T}$	$L_{Aeq,T}$	$L_{Aeq,T}$
		等価音圧レベル	C特性	$L_{Ceq,T}$	$L_{Ceq,T}$	—
$L_{AE}$		単発騒音暴露レベル	A特性	$L_{AE}$	$L_{EA,T}$	$L_{AE}$
$L_x$	$L_5$ $L_{10}$ $L_{50}$ $L_{90}$ $L_{95}$	時間率騒音レベル	A特性	$L_{AN,T}$	—	$L_x$
$L_{max}$		騒音レベルの最大値	A特性	$L_{max}$	—	—

## 目次

	ページ
1. 概要	1
2. 転送方式	1
3. データ転送手順	2
3.1 PCからNA-29へコマンドを送信する場合の手順	2
3.2 データ転送を要求するコマンドを送る場合の手順	3
3.3 エラー処理	5
3.4 コマンドフォーマット	6
3.5 誤ったコマンドを送信した場合の動作	6
4. コマンドリスト	7
5. 出力データフォーマット	14
5.1 表示データフォーマット	14
5.2 マーカー値	16
5.3 D_Lデータセット	17
5.4 ディレクトリー	17
5.5 表示画面, メモリー情報	18
6. サンプルプログラム (使用コンピューター NEC PC-9801シリーズ)	25
6.1 RS-232Cチェックプログラム	25
6.2 デモンストレーションプログラム	27



## 1. 概要

NA-29にはRS-232Cインタフェースが内蔵されています。このインタフェースを介してNA-29にストアされているデータをコンピューター（以後PCと呼ぶ）に送信したり、PCで測定条件などを設定することができます。

なお、本インタフェースでは以下に示す項目の操作は出来ません。

- ・プリンターへの画面出力
- ・ボーレートの設定
- ・設定画面の表示
- ・ディレクトリー画面の表示

## 2. 転送方式

・転送制御手順	有手順
・通信方式	半二重
・X_パラメーター	NA-29が送信中のみ有効
・データビット長	8ビット
・スタートビット	1ビット
・ストップビット	2ビット
・パリティチェック	なし
・ボーレート	1200, 2400, 4800, 9600 BPS 選択

### ○ X\_パラメーター

NA-29がデータ送信中に限って、PCがこのデータの受信を停止または再開するために装置制御コードDC3(X\_off, CTRL+S=13H)またはDC1(X\_on, CTRL+Q=11H)を送信することができます。

- ・DC3を受信するとNA-29は直ちにPCに対してデータ送信を停止
- ・DC1を受信すると、DC3受信後停止していたPCに対してデータ送信を再開

### ○ リモート／ローカル機能

リモートモードではRS-232Cインタフェースのコマンドで動作し、手動操作は出来なくなります。手動操作を可能にするためには、PCよりコマンドLOCを送信するか、NA-29のEXITキーを押してローカルモードにします。

ローカルモードではパネルキーによる操作、RS-232Cインタフェースによる操作どちらでも可能です。

### 3. データ転送手順

#### 3.1 PCからNA-29へコマンドを送信する場合の手順

PCでNA-29をコントロールしたり、そのデータを読み出すためには決められたコマンドを送信しなければなりません。しかし、PCが適切なタイミングでコマンドを送信してもNA-29がそのコマンドを見落とす危険があります。

そこで、NA-29ではデータの送受信について一定の手順を踏みながら実行する方法をとっています。

NA-29へコマンドを送信するためには、まず〈ENQ〉を送信しなければなりません。NA-29はこの〈ENQ〉に反応して〈ACK〉READY〈CR〉〈LF〉をPCへ返信します。PCがこの〈ACK〉READY〈CR〉〈LF〉を確認して初めてコマンド送信が可能になり、4秒以内にコマンド送信を完了しなければなりません。

- 【注意】
1. NA-29のOPEキーを押した直後は、ストアデータの二次処理を実行する間コマンドの受信が出来ません。従って、この場合はコマンドエラー、またはタイムアウトエラーとなります。
  2. コマンドの送信はPCが〈ACK〉READY〈CR〉〈LF〉を受信した後10 ms以上経過してから行って下さい。

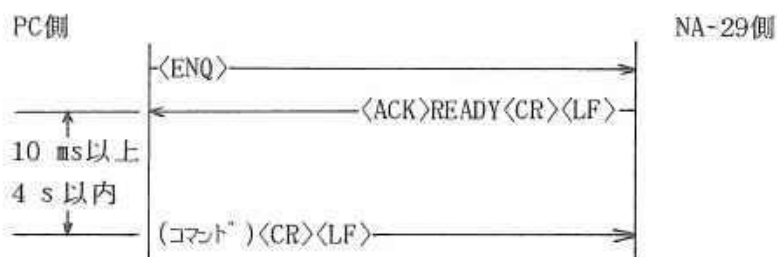


図1 コマンド転送の手順

表1 各コードの数値

コード	16進数	10進数
〈ENQ〉	05H	5
〈ACK〉	06H	6
〈EOT〉	04H	4
〈CR〉	0DH	13
〈LF〉	0AH	10

### 3.2 データ転送を要求するコマンドを送る場合の手順

NA-29は、コマンド受信を終了（〈CR〉コードでコマンドの終了を判定）すると直ちにコマンド解釈とその実行にとりかかります。送信するデータ数が250バイト以下なら1回、250バイト以上なら1回のデータ数が250バイト以内になるように数回に分けて送信します。

数回に分けてデータを送信する場合は1回分のデータを送信し、次のデータの送信はPCが〈ACK〉NEXT〈CR〉〈LF〉を返信するまで実行されません。

PCは1回分のデータを受信した後、次のデータを必要とする場合は4秒以内に〈ACK〉NEXT〈CR〉〈LF〉をNA-29に送信しなければなりません。

1回分のデータフォーマットは次のようになります。

#### ○ 1回でデータ送信が出来る場合

`(DATA)〈EOT〉〈CR〉〈LF〉`

〈DATA〉：コマンドが要求したデータ本体で、総数は250バイト以下

〈EOT〉：転送制御キャラクター CTRL+D=04H

#### ○ データ総数が多いためN回に分けてデータ転送する場合

<code>(DATA)<sub>1</sub>〈CR〉〈LF〉</code>	1回目
<code>(DATA)<sub>2</sub>〈CR〉〈LF〉</code>	2回目
⋮	⋮
<code>(DATA)<sub>N</sub>〈EOT〉〈CR〉〈LF〉</code>	N回目（最後）

いずれの場合も、残りデータがない場合に〈DATA〉の後に〈EOT〉が付加されます。PC側では残りのデータがあるか否かを〈DATA〉の後のコードが〈EOT〉であるか否かで判定し、もし〈CR〉であればNA-29に〈ACK〉NEXT〈CR〉〈LF〉を送信して残りの〈DATA〉を受け取らなければなりません。

図2に〈DATA〉を3回に分けて転送するときの例を示します。



図2 データ転送と要求するコマンドを送る場合の手順

- 【注意】** 1. OPE (ストアデータの二次処理) を実行させるコマンドを送信した場合、NA-29は最大8秒間 RS-232Cインタフェースの通信を停止します。従って、続いてOPE結果を要求するデータを送信していた場合にはデータ受信まで最大8秒間待たなければなりません。
2. `<ACK>NEXT<CR><LF>`の送信は`<DATA><CR><LF>`を受信後、10 ms以上待ってから実行して下さい。



### 3.3 エラー処理

NA-29とPCとの通信を正しく実行するために一定の手順が必要です。  
この手順を外れた場合のPCとNA-29の対応について以下に示します。

A) PCからコマンド<ENQ>を送信したが応答がない場合

2秒程度待って再度<ENQ>を送信します。

これを5～6回繰り返してもNA-29から応答がない場合には次のことが考えられます。

- 1) 転送方式が整合していない
- 2) RS-232Cケーブルの異常
- 3) NA-29の電源が入っていない

B) <ACK>READY<CR><LF>を受信後、約4秒以内にコマンド転送を完了しなかった場合

NA-29はコマンド待ちを放棄(タイムアウト1)します。従って<ENQ>からやり直します。

C) 残りデータを受け取るための<ACK>NEXT<CR><LF>送信を4秒以内に実行しなかった場合

NA-29は残りデータの送信を放棄(タイムアウト2)します。

更に、今のデータ要求コマンドと一緒に複数個コマンドが送られていた場合、それらのコマンドはすべて無視されます。

[例]

コマンド1/コマンド2/コマンド3.....<CR><LF>

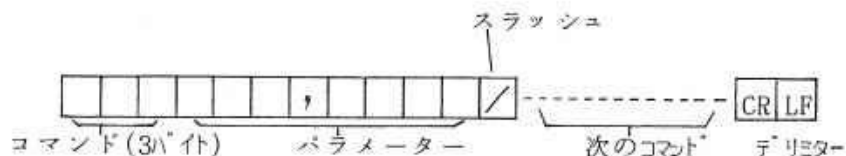
コマンド2でタイムアウトが発生した場合、コマンド3以降は無視されます。

D) コマンドのフォーマットが規則外である場合

規則に外れたコマンド以降のコマンドは無視されます。

上記 B)～D)のどれかが発生した場合、NA-29は警告ブザーを2回鳴らします。

### 3.4 コマンドフォーマット



- 【注意】
1. パラメーターの区切りにはカンマを uses.
  2. コマンドの区切りとして / を入れても省略してもかまいません。
  3. 全体で256バイトまでのコマンドが同時に送信出来ます。
  4. 設定画面またはディレクトリー画面表示中にコマンドを送信すると自動的に測定画面に変わります。

### 3.5 誤ったコマンドを送信した場合の動作

以下に示す誤ったコマンドを送信した場合、NA-29はその誤ったコマンドからCR、LFまでのコマンドを無視し、警告ブザーを2回鳴らします。

- 1) 登録されていないコマンド
- 2) パラメーターの個数の誤り
- 3) パラメーターとして指定範囲外の値

#### 4. コマンドリスト

##### ○ 一般設定

コマンド	パラメーター	内 容
SRT	X <sub>1</sub> X <sub>2</sub> X <sub>3</sub>	Leq, Lmax, LAE 演算開始 X <sub>1</sub> X <sub>2</sub> : 実測時間 01~59 (X <sub>3</sub> =0, 1の場合) 01~24 (X <sub>3</sub> =2の場合) X <sub>3</sub> : 実測時間の単位 0: 秒 1: 分 2: 時
STP	なし	Leq, Lmax, LAE 演算中止または オートストア中止
PSE	なし	データ取り込みのポーズ
CNT	なし	ポーズ解除
DSP	X <sub>1</sub>	表示データ選択(Lp, Leq, Lmax, LAE) X <sub>1</sub> =0: Lp 1: Lmax 2: Leq 3: LAE
OCT	なし	騒音分析モードに設定
SLM	なし	騒音測定モードに設定
GRP	なし	グラフ表示
NUM	なし	数値表示
TMC	X <sub>1</sub>	時定数設定 X <sub>1</sub> =0: FAST 1: SLOW 2: 10 ms
WGT	X <sub>1</sub>	周波数補正回路設定 X <sub>1</sub> =0: A特性 1: C特性 2: F特性

コマンド	パラメーター	内 容
RNG	X1	測定レンジ設定 X1=0: フルスケール 70 dB 1: 80 dB ⋮ 6: 130 dB 7: 140 dB
CAL	なし	校正モードに設定
CAF	なし	校正モード解除
TRG	X1, X2X3	トリガー設定 X1=0: トリガ OFF 1: 内部トリガ ON 2: 外部トリガ ON X2X3: 内部トリガ ONのときの設定レベル 内部トリガ ON以外の場合もデフォルトパラメーターを入れる
MKP	X1X2	マーカー位置設定 X1X2: LEVEL/FREQの場合 00: 31.5 Hz 01: 63 Hz ⋮ 09: AP LEVEL/TIMEの場合 00~74
REF	X1, X2X3	レファレンスレベル設定 X1=0: OFF 1: ON X2X3=07: フルスケールを70 dBに設定 08: 80 dB ⋮ 20: 200 dB
PON	なし	電源の自動遮断
POF	なし	解除
RMT	なし	リモートモードに設定
LOC	なし	ローカルモードに設定

○ データストア

コマンド	パラメーター	内 容
STA	X1, X2X3X4	<p>オートストア実行, ストアタイプ変更</p> <p>X1=0:ストアデータの種類 Lp</p> <p>1: " Lmax</p> <p>2: " Leq</p> <p>3: " LAE</p> <p>X2X3X4: Lpの場合のストア間隔</p> <p>0: 2 ms, 1: 5 ms, ……</p> <p>……………10: 5 s, 11: 10 s</p> <p>Lp以外の場合</p> <p>X2X3: 01~59または01~24</p> <p>X4=0: 秒</p> <p>1: 分</p> <p>2: 時</p>
STM	X1X2X3X4	<p>マニュアルストア実行, ストアタイプ変更</p> <p>X1X2X3X4: アドレス番号設定</p> <p>(0001~1500)</p>
STS	X1X2X3	<p>D_Lストア実行, ストアタイプ変更</p> <p>X1X2X3: アドレス番号設定</p> <p>(001~250)</p>

○ データリコール

コマンド	パラメーター	内 容
RCF	X1X2X3X4, X5	オートまたはマニュアルストアデータを LEVEL/FREQとしてリコール X1X2X3X4:リコールするアドレス番号 (0001~1500) X5:二重表示の設定, 解除 1: 設定 0: 解除
RCT	X1X2X3X4, X5, X6	オートストアデータをLEVEL/TIMEとして リコール X1X2X3X4:リコールする先頭のアドレ ス番号(0001~1500) X5:表示周波数 0: 31.5 Hz, 1: 63 Hz, ..... 9: AP X6:アドレス番号の間引き数 0: 1, 1: 2, 2: 5, 3: 10, 4: 20
RCS	X1X2X3, X4	D_Lデータのリコール X1X2X3:リコールするアドレス番号 (001~250) X4:測定ポイント 0: BGN 1~5:測定ポイント
EDR	なし	リコール終了

○ メモリー操作

コマンド	パラメーター	内 容
CLR	X1X2X3X4, X5X6X7X8	データクリア X1X2X3X4：クリアする最初のアドレス番号 X5X6X7X8：クリアする最終アドレス番号
CMT	X1X2X3X4, X5X6X7X8X9X10X11X12	コメント入力 X1X2X3X4：アドレス番号 X5～X12：コメント文
OOF OON	X1X2X3X4, X5X6X7X8	パワー平均を実行する場合の演算データの指定 X1X2X3X4：最初のアドレス番号設定 X5X6X7X8：最終アドレス番号設定 OOF：演算除外 OON：演算実行
DRM	X1X2X3, X4X5X6X7, X8	室間平均音圧レベル差測定の場合のルーム番号入力 X1X2X3：アドレス番号(001～250) X4X5X6X7：ルーム番号(0001～9999) X8=S：音源室データ R：受音室データ
LRM	X1X2X3, X4X5X6X7, X8	床衝撃音レベル測定の場合のルーム番号入力 X1X2X3：アドレス番号(001～250) X4X5X6X7：ルーム番号(0001～9999) X8：整理番号(1～5)

- 【注意】
1. CMTコマンドのコメント文(X5～X12)は必ず8文字入力して下さい。
  2. DRM, LRMコマンドのルーム番号(X4～X7)は必ず4文字入力して下さい。

○ ストアデータの二次演算

コマンド	パラメーター	内 容
OPE	X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> X <sub>4</sub> , X <sub>5</sub> , X <sub>6</sub> X <sub>7</sub> X <sub>8</sub> X <sub>9</sub>	オートストアデータの二次演算を実行 X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> X <sub>4</sub> : 演算開始アドレス X <sub>5</sub> =0: L <sub>5</sub> の演算 1: L <sub>10</sub> の演算 2: L <sub>50</sub> の演算 3: L <sub>90</sub> の演算 4: L <sub>95</sub> の演算 5: Leqの演算 6: パワー平均 X <sub>6</sub> X <sub>7</sub> X <sub>8</sub> X <sub>9</sub> : 演算するデータ数
SDR	X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> X <sub>4</sub>	ルーム番号による空間平均音圧レベル差の演算 X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> X <sub>4</sub> : DRMコマンドで入力したルーム番号
SDA	X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> , X <sub>4</sub> X <sub>5</sub> X <sub>6</sub>	アドレス番号による空間平均音圧レベル差の演算 X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> : 音源室データのアドレス番号(001~250) X <sub>4</sub> X <sub>5</sub> X <sub>6</sub> : 受音室データのアドレス番号(001~250)
SLR	X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> X <sub>4</sub>	ルーム番号による床衝撃音レベルの演算 X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> X <sub>4</sub> : LRMコマンドで入力したルーム番号
SLA	X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> , X <sub>4</sub> X <sub>5</sub> X <sub>6</sub> , X <sub>7</sub> X <sub>8</sub> X <sub>9</sub> , X <sub>10</sub> X <sub>11</sub> X <sub>12</sub> , X <sub>13</sub> X <sub>14</sub> X <sub>15</sub>	アドレス番号による床衝撃音レベルの演算 X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> X <sub>4</sub> X <sub>5</sub> X <sub>6</sub> X <sub>7</sub> X <sub>8</sub> X <sub>9</sub> X <sub>10</sub> X <sub>11</sub> X <sub>12</sub> X <sub>13</sub> X <sub>14</sub> X <sub>15</sub> } 演算する5組のアドレス番号
EDO	なし	二次演算, D_L演算終了



○ D\_L測定

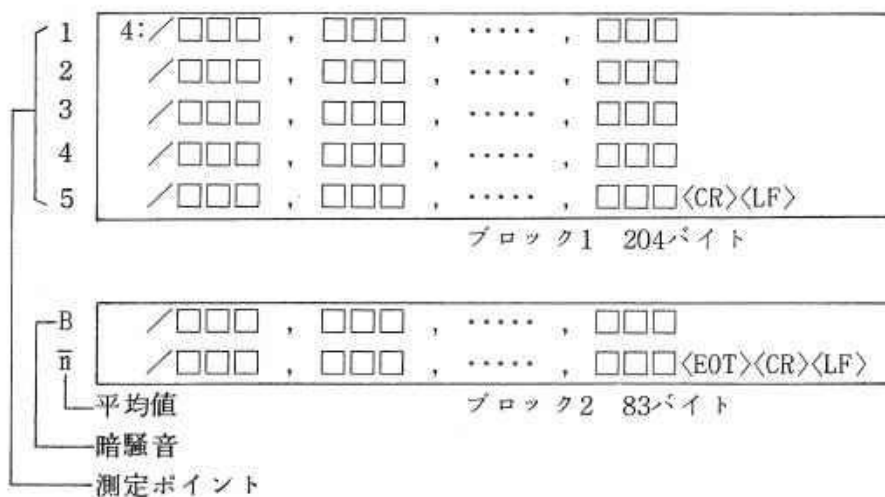
コマンド	パラメーター	内 容
SON	なし	D_L測定モードの設定
SOF	なし	D_L測定モードの設定解除
SMC	X1 , X2X3	D_L測定条件の設定 X1=0: 空間平均音圧レベル差 1: 重量床衝撃音レベル 2: 軽量床衝撃音レベル X2X3: 1測定ポイント平均回数 (01~99)
ENT	X1X2, X3	測定データの取り込み X1X2=0: 31.5 Hzのデータ 01: 63 Hz ⋮ 09: AP 10: 全周波数データを同時 X3=0: 暗騒音データ 1~5: 測定ポイントごとに取り込む

○ データ出力

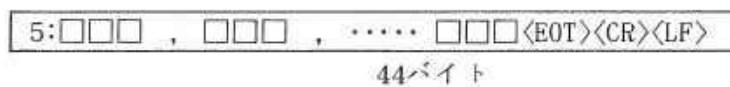
コマンド	パラメーター	内 容
D0D	なし	表示されているデータの出力
PSI	X1	キー設定状態の情報出力 X1=0: 表示状態 1: 入力状態 2: 表示パラメーター 3: メモリー情報 4: ストアデータの二次演算 5: D_L測定情報 6: その他
MKV	なし	マーカー値出力
DDL	なし	D_L測定データセット出力
DIR	X1X2X3X4, X5X6X7X8	ディレクトリー出力 X1X2X3X4: 出力開始アドレス番号 X5X6X7X8: 出力最終アドレス番号



○ 騒音分析モード (D\_L設定での数値表示)



○ 騒音分析モード (D\_L演算表示)



## 5.2 マーカー値

MKVコマンドによりNA-29からPCに転送されるデータフォーマットは以下のようになります。

- 騒音測定モード

0:  $\underbrace{\square\square\square\square}_{\text{レベル}} \cdot \square \langle \text{EOT} \rangle \langle \text{CR} \rangle \langle \text{LF} \rangle$  10バイト

- 騒音分析モード (LEVEL\_FREQ)

1:  $\underbrace{\square\square\square\square\square}_{\text{周波数}}, \underbrace{\square\square\square \cdot \square}_{\text{レベル}} \langle \text{EOT} \rangle \langle \text{CR} \rangle \langle \text{LF} \rangle$  17バイト

- 騒音分析モード (LEVEL\_TIME リコール時のみ)

2:  $\underbrace{\square\square\square\square\square}_{\text{周波数}}, \underbrace{\square\square\square \cdot \square}_{\text{レベル}}, \underbrace{\square\square\square\square}_{\text{マーカーのアドレス番号}} \langle \text{EOT} \rangle \langle \text{CR} \rangle \langle \text{LF} \rangle$   
22バイト

- 騒音分析モード (D\_L設定での数値表示)

4:  $\langle \text{EOT} \rangle \langle \text{CR} \rangle \langle \text{LF} \rangle$  5バイト

- 騒音分析モード (D\_L演算表示)

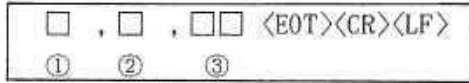
5:  $\langle \text{EOT} \rangle \langle \text{CR} \rangle \langle \text{LF} \rangle$  5バイト



## 5.5 表示画面, メモリー情報

PSInコマンドによりNA-29からPCに転送されるデータフォーマットは以下のようになります。

### ○ 表示状態(PS10)



9バイト

- ① 0: 騒音分析モード      1: 騒音測定モード
- ② 0: CUR(D\_L OFF)      1: D\_L ON      2: RCL  
 3: ストアデータの二次演算中
- ③ ②が0のとき (CUR D\_L OFF)
- 0: Lp, Lmax, Leq, LAE 表示中 (Lmax, Leq, LAE は測定完了しているものとみなす)
  - 1: Leq測定中
  - 2: Lpデータをオートストア中
  - 3: Lmax      ♪
  - 4: Leq      ♪
  - 5: LAE      ♪
- トリガー待ちのときは各々11,12,13,14,15になる
- ②が1のとき (D\_L ON)
- 0: Lp, Lmax, Leq, LAE 表示中 (Lmax, Leq, L は測定完了しているものとみなす)
  - 1: Lmax, Leq, LAE測定中
  - 11: Lmax, Leq, LAE 測定のトリガー待ち
- ②が2のとき (RCL)
- 0: レベル, 周波数または騒音データ表示
  - 1: レベル・タイム表示
  - 2: D\_Lデータ表示
  - 3: ストアデータなし(NO DATA)
- ②が3のとき (ストアデータの二次演算中)
- 0: L5    1: L10    2: L50    3: L90
  - 4: L95    6: P\_AVE
  - 20: 空間平均音圧レベル差
  - 21: 床衝撃音レベル







- ⑩ レベル・タイムにおける表示周波数  
0: 31.5 Hz    1: 63 Hz.....9: AP
- ⑪ レベル・タイムにおけるリコールデータのアドレス番号の間引き数  
0: 1/1    1: 1/2    2: 1/5    3: 1/10    4: 1/20

○ メモリー情報(PSI3)

/□	, □□□□	, □	, □□□/□	, □□	, □□□/□□□□	,
①	②	③	④	⑤	⑥	⑦
□□<EOT><CR><LF>						
⑨						33バイト

・ストアデータに関するもの

- ① ストアタイプ      0: AUTO      1: MANU      2: D\_L
- ② スストアデータ数
- ③ スストアデータの種類    0: Lp    1: Lmax    2: Leq    3: LAE
- ④ スストア間隔  
     ストアデータがLpのとき  
         0: 2 ms    1: 5 ms.....11: 10 s  
     ストアデータがLeqのとき  
         実測時間設定値(1~59)及び単位(0:秒 1:分 2:時)

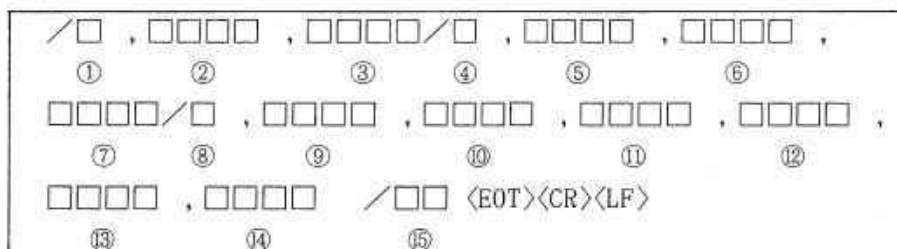
・新たにストアするとき有効となるパラメーター

- ⑤ ストアタイプ      0: AUTO      1: MANU      2: D\_L
- ⑥ Lpのストア間隔  
         0: 2 ms              4: 50 ms              8: 1 s  
         1: 5 ms              5: 100 ms             9: 2 s  
         2: 10 ms             6: 200 ms            10: 5 s  
         3: 20 ms             7: 500 ms            11: 10 s
- ⑦ Leqのストア間隔(Leq\_TIME)  
         実測時間設定値(1~59)及び単位(0:秒 1:分 2:時)

・その他

- ⑧ 現在のアドレス番号
- ⑨ そのアドレス番号にストアされているデータの種類  
         0: Lp                              20: D  
         1: Lmax                            21: Lu  
         2: Leq                             22: Ll  
         3: LAE                             99: NO DATA

○ ストアデータの二次演算に関する情報(PSI4)



67バイト

- ① 0: L5      1: L10.....5: Leq      6: P\_AVE
- ② Lx, Leqのデータ数
- ③ P\_AVEのデータ数
  
- ④ <D>用INDEX      0: アドレス番号      1: ルーム番号
- ⑤ ルーム番号
- ⑥ アドレス番号 (音源室)
- ⑦ アドレス番号 (受信室)
  
- ⑧ <L>用INDEX
- ⑨ ルーム番号
- ⑩~⑭ アドレス番号
  
- ⑮ NA-29のOPEキーを押して実行する二次演算の種類
  - 0: L5      1: L10.....5: Leq      6: P\_AVE
  - 20: <D>    21: <L>
  - 99: マニュアルストアデータ



6. サンプルプログラム (使用コンピューター NEC PC9801シリーズ)

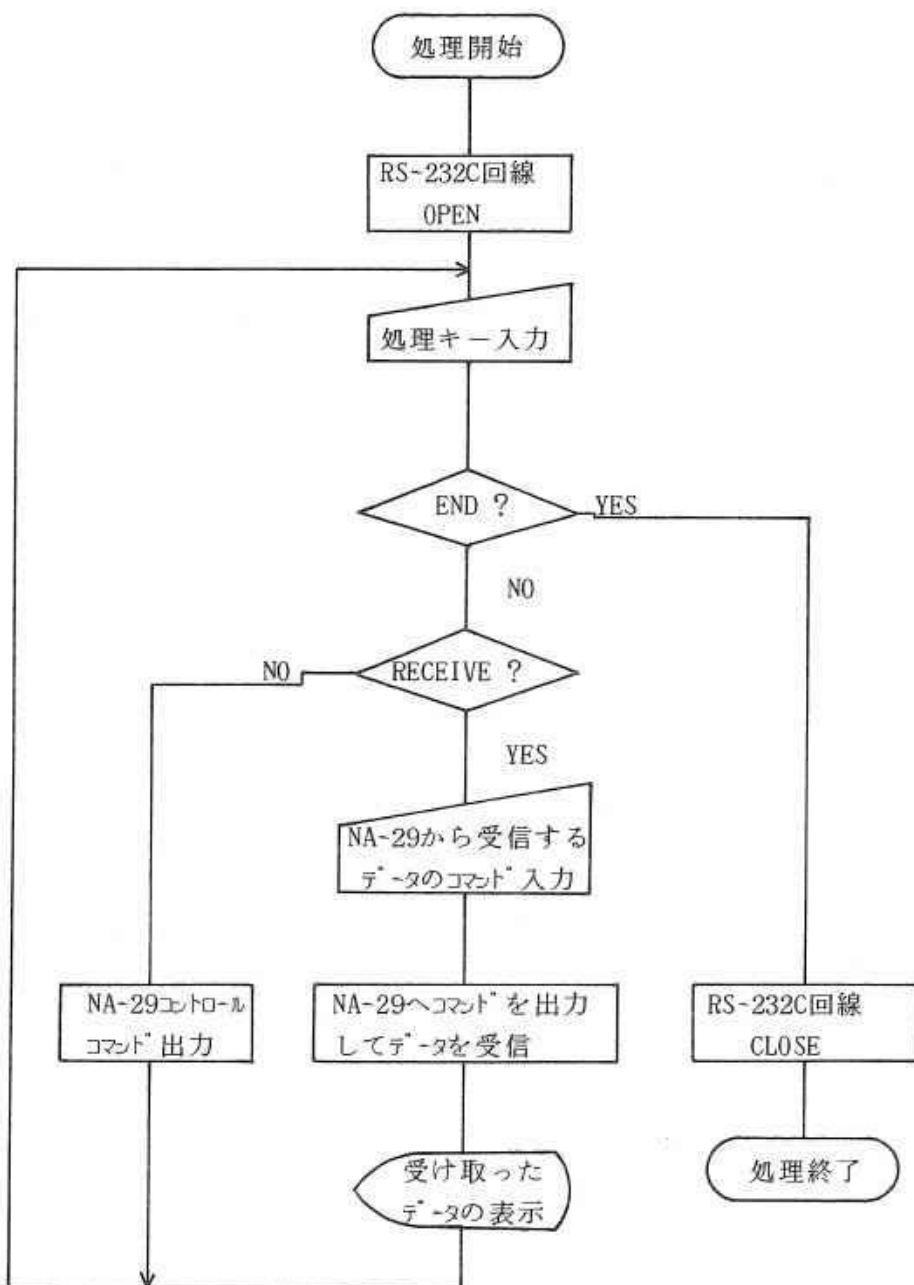
```

1000 REM      NA-29 RS-232Cチェック
1010 REM
1020 REM      1988 / 6          By RION Co., LTD.
1030 REM
1040 REM *****
1050 REM
1060 CONSOLE 0,25,0,1
1070 SCREEN 3,0,0,1
1080 WIDTH 80,25
1090 CLS 3
1100 OPEN "COM:" AS #2
1110 COLOR 5:PRINT SPC(17);"※※ NA-29 RS-232C チェック ※※"
1120 PRINT:PRINT SPC(10);"Command :":COLOR 7
1130 PRINT SPC(15);"END          : 処理終了"
1140 PRINT SPC(15);"RECEIVE     : データ受信"
1150 PRINT SPC(15);"コマンド    : NA-29 コントロール"
1160 COLOR 1:PRINT "-----"

1170 CONSOLE 8,17 : CLS 3
1180 PRINT : COLOR 7:PRINT SPC(15)"※ コマンドを入力して下さい。";
1190 COLOR 4:LINE INPUT COMMAND$:BEEP
1200 IF COMMAND$<>"END" THEN 1230
1210 CLOSE #2
1220 END
1230 IF COMMAND$="RECEIVE" THEN GOSUB *RECEIVE:GOTO 1180
1240 REM
1250 REM =====
1260 REM      コマンド送信手順
1270 REM =====
1280 GOSUB *START: IF TIMEOUT=1 THEN 1180
1290 PRINT #2,COMMAND$:GOTO 1180
1300 REM
1310 REM =====
1320 REM      データ受信手順
1330 REM =====
1340 *RECEIVE
1350 PRINT:COLOR 7
1360 PRINT SPC(10);"★ データを要求すべきコマンドを入力して下さい。"
1370 PRINT SPC(20);"制御コマンドは、1種類のみ ";
1380 COLOR 6:LINE INPUT COMMAND$:BEEP
1390 GOSUB *START: IF TIMEOUT=1 THEN RETURN
1400 PRINT #2,COMMAND$
1410 COLOR 4 : PRINT : PRINT SPC(10);"入力データ"
1420 GOSUB *RECV.DATA
1430 RETURN
1440 REM
1450 REM ++++++
1460 REM      データ受信
1470 REM ++++++
1480 *RECV.DATA
1490 COLOR 6 : H$="":TOUT = 1000 : TIMEOUT = 0
1500 TOUT = TOUT - 1
1510 IF TOUT=0 THEN COLOR 2:PRINT SPC(20);"** Not receive (Error) !! **":TIME
OUT=1:RETURN
1520 IF LOC(2)=0 THEN 1500
1530 A$=INPUT$(1,#2)
1540 H$=H$+A$
1550 IF ASC(A$)>10 THEN TOUT=1000:GOTO 1500
1560 A = LEN(H$)
1570 IF MID$(H$, A-2, 1) = CHR$(8H4) THEN 1610
1580 PRINT #2,CHR$(8H6);
1590 PRINT #2,"NEXT" : H$=LEFT$(H$,LEN(H$)-2)
1600 PRINT SPC(10);H$:GOTO 1490
1610 H$ = LEFT$(H$,LEN(H$)-3) : PRINT SPC(10);H$ : RETURN
1620 REM
1630 REM *****
1640 REM      E N Q   A C K の ハ ン ド シ ェ イ ク
1650 REM *****
1660 *START
1670 H$="":PRINT #2,CHR$(8H5);
1680 TOUT = 1000 : TIMEOUT = 0
1690 IF LOC(2) = 0 THEN TOUT = TOUT-1 : IF TOUT = 0 THEN 1730 ELSE GOTO 1690
1700 A$=INPUT$(1,#2):H$=H$+A$:IF ASC(A$)>10 THEN 1690
1710 RETURN
1720 REM
1730 TIMEOUT = 1
1740 PRINT:COLOR 2:PRINT SPC(20);"** Not receive (Error) !! **"
1750 FOR I=1 TO 5000:NEXT
1760 RETURN

```

○ RS-232Cチェック フローチャート



【注意】 PCのCRT上にエラーメッセージ “\*\* Not recieve (Error)!! \*\*” が表示された場合は、PCとNA-29との通信に異常が発生したことを示します。

## 6.2 デモンストレーションプログラム

ここに示すデモンストレーションプログラムを実行するためには、以下のOS及びユーティリティーが必要です。

### OS

MS-DOS                    マイクロソフト社製 PC-9801用    バージョン3.10以上

### ユーティリティー

   SORT.EXE                (MS-DOSユーティリティー)  
                             ディスクに書き込まれたファイル検索時に使用

   SWITCH.EXE             (MS-DOSユーティリティー)  
                             NA-29とPC-9801の通信モード（ボーレートなど）の設定時に使用

   N88BASIC.EXE            NEC社製 MS-DOS用BASIC    バージョン3.0以上  
                             デモンストレーションプログラムを実行するときの  
                             BASICユーティリティー

● 作業用システムディスク、データ格納用ディスクの作成手順

生フロッピーディスクを2枚用意し、作業用システムディスク及びデータ格納用ディスクを作成します。

1. PCの電源を入れます。
2. MS-DOSシステムディスクをドライブA(1)に入れ、システムを起動します。
3. 生フロッピーディスクをドライブB(2)に入れます。
4. MS-DOSのコマンドラインで次のコマンドを入力し、ディスクのフォーマット及びMS-DOSのシステムファイルをコピーします。

```
FORMAT_B:/S [RETURN]
```

【注意】 コマンド中の\_はスペースを意味します。(以下同様)

5. キーボード上の任意のキーを押した後、ディスクのタイプ(640 KBまたは1 MB)を選択し、[RETURN]キーを押します。  
フォーマットを開始し、終了後MS-DOSのシステムがコピーされ、次に別のディスクをフォーマットする(Y)か、しない(N)かのメッセージが表示されます。
6. Nキー(フォーマットしない)を押し、続いて[RETURN]キーを押します。
7. 次のコマンドを入力し、ソートユーティリティーをコピーします。  
COPY\_SORT.EXE\_B: [RETURN]
8. 次のコマンドを入力し、通信モード設定用ユーティリティーをコピーします。  
COPY\_SWITCH.EXE\_B: [RETURN]
9. ドライブ1からMS-DOSシステムディスクを抜き取り、MS-DOS用N88日本語BASICシステムディスクを入れます。
10. 次のコマンドを入力し、N88日本語BASICをコピーします。  
COPY\_N88BASIC.EXE\_B: [RETURN]
11. ドライブ1からMS-DOS用N88日本語BASICシステムディスクを抜き取り、再度MS-DOSシステムディスクをドライブ1に入れます。
12. ドライブ2から今作成した作業用システムディスクを抜き取り、生フロッピーディスク(データ格納用)を入れます。
13. 次のコマンドを入力し、データ格納用ディスクのフォーマットをします。  
FORMAT\_B: [RETURN]

以上で作業用システムディスク及びデータ格納用ディスクの作成は終了です。



以後、作業用システムディスクでBASICを起動し、デモンストレーションプログラムを入力してNA-29のデータ処理を行います。

【注意】 OS、ユーティリティー及びMS-DOS用N88日本語BASICについては、それぞれに付属の説明書を参照して下さい。

● プログラムによる処理内容

このデモンストレーションプログラムは

- ① 1/1オクターブモードにおける10 s間のLeq測定
- ② 2 ms間隔でのLpデータのストア

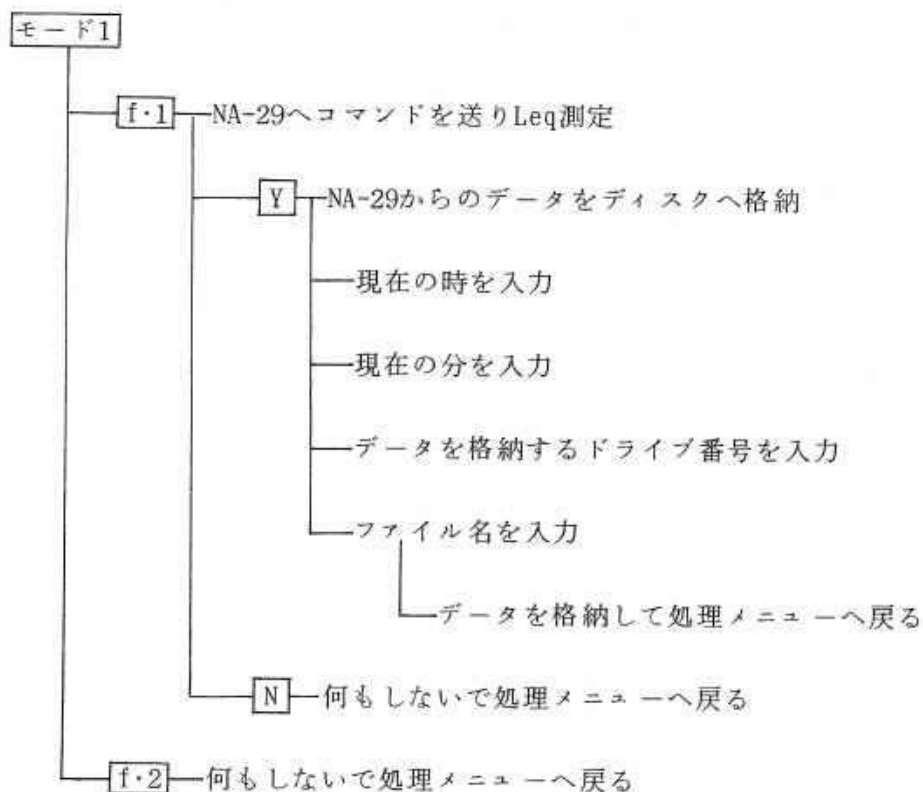
の作業を行い、それぞれのデータをPCに転送してCRTに表示します。

なお、このプログラムは、処理項目が以下の7つに分けられています。

処理項目	操作キー
1) モード1	f・1
2) モード2	f・2
3) データ表示	f・3
4) コマンド入力	f・4
5) RS-232C通信モード設定	f・5
6) MS-DOSシステムの呼び出し	f・6
7) NA-29処理終了	f・7

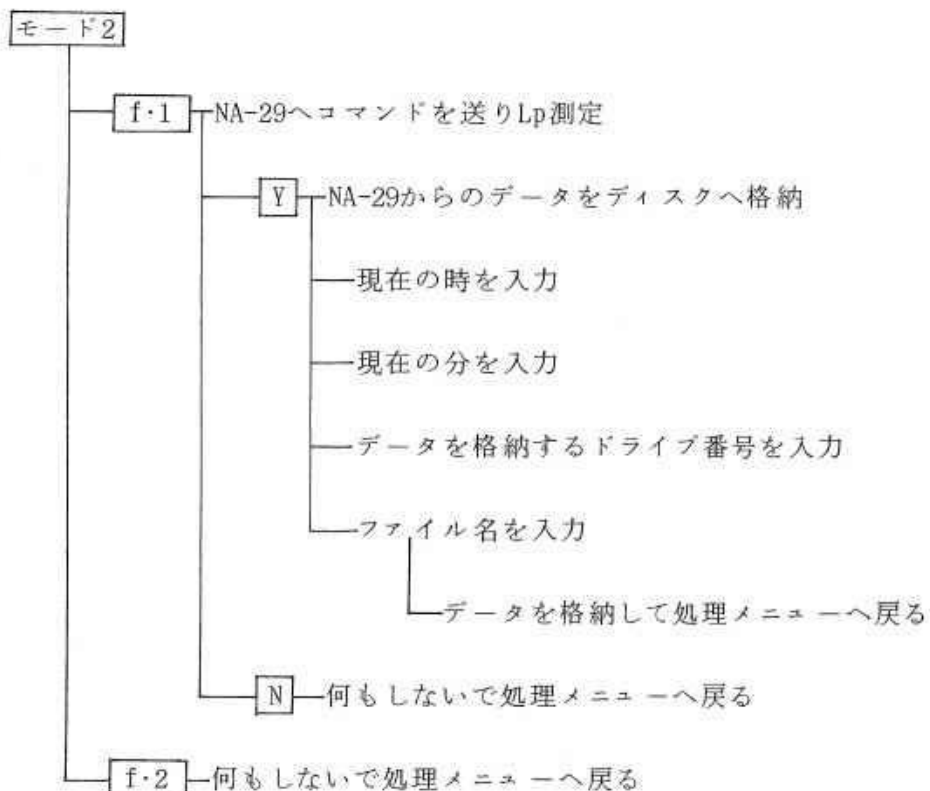
1) モード1

1/1オクターブモードで10 s間のLeqを測定し、そのデータをディスクへ格納します。



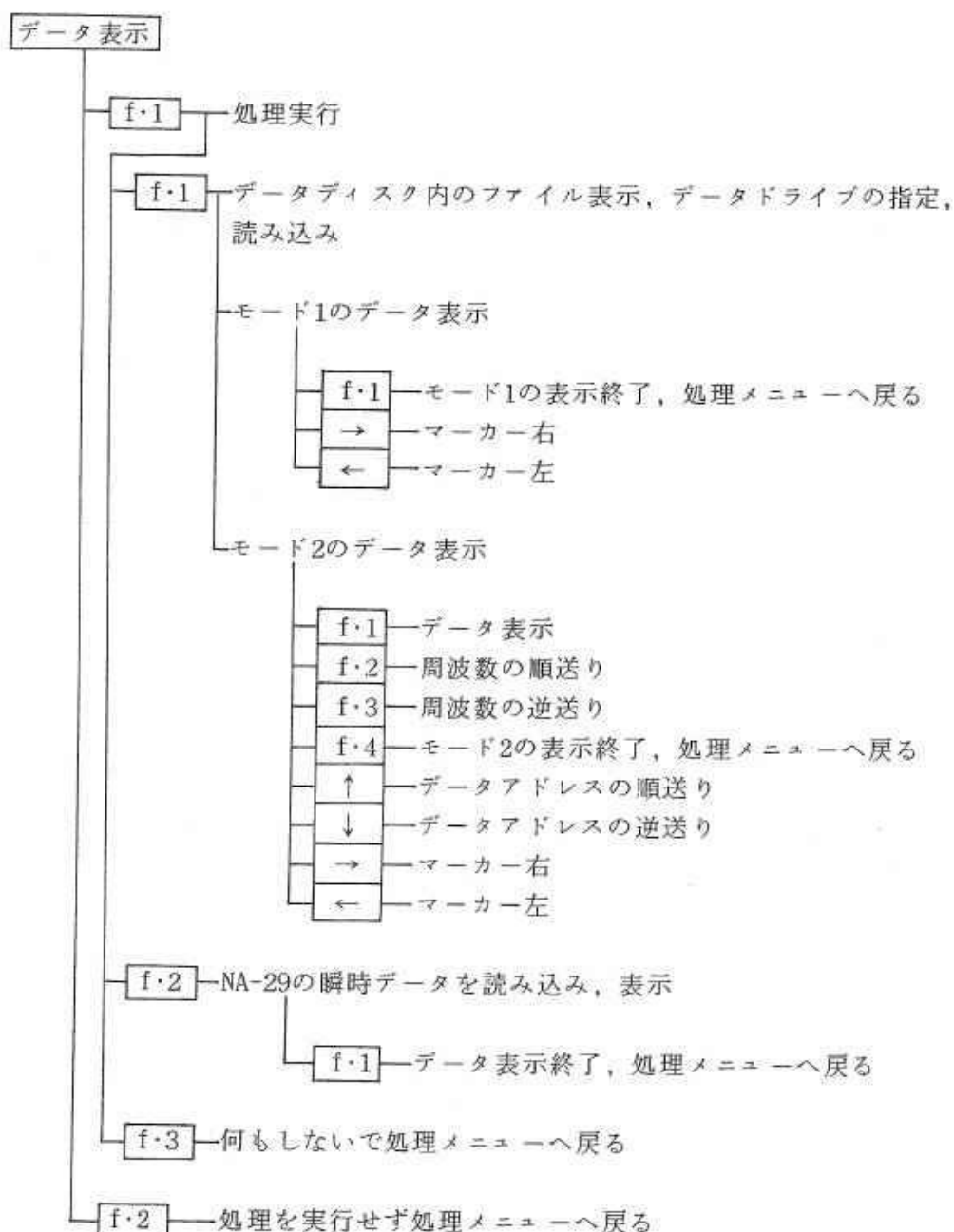
## 2) モード2

2 ms間隔でLpデータをNA-29のメモリー内にストアした後、レベル・タイムとしてリコールし、そのデータをディスクへ格納します。



### 3) データ表示

モード1, モード2でディスクに格納したデータの表示を行います。



○ データ表示例

・データファイルの選択

\*\*\* NA-29 \*\*\*

サンプル プログラム (Version 1.0) RION CO., LTD.

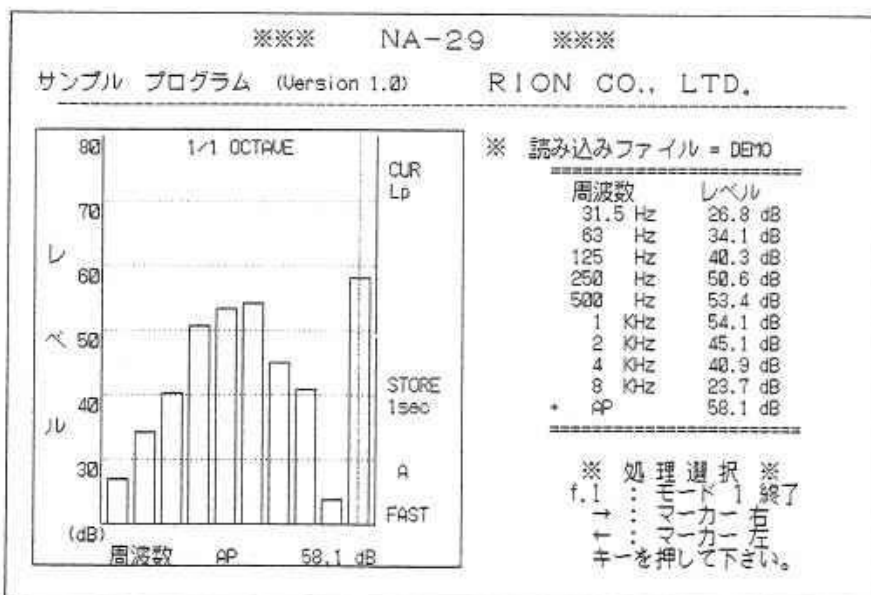
---

\*\*\*\*\* DIRECTORY \*\*\*\*\*

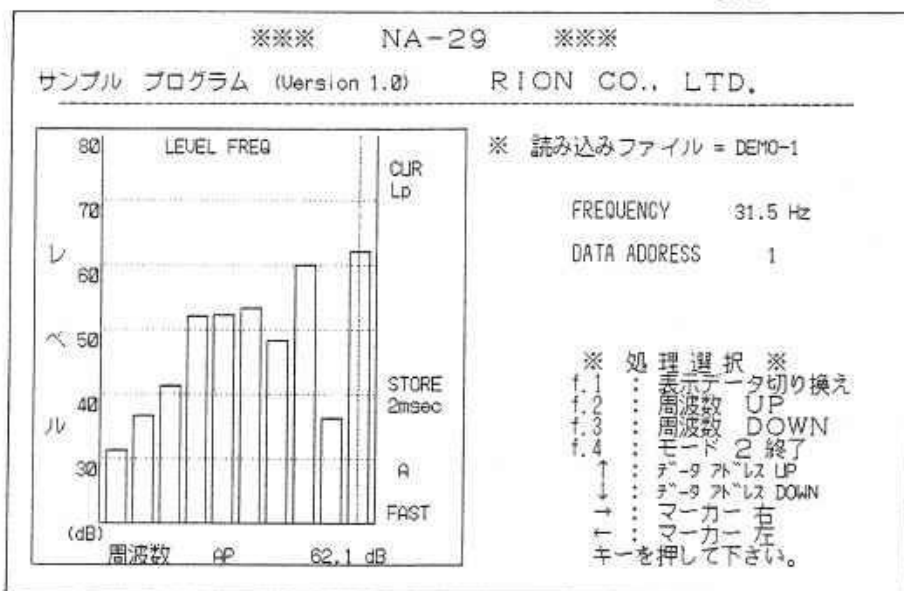
NO	ファイル名	容量	日付	時間
1	DEMO	163	88-07-05	13:50
2	DEMO-1	75100	88-07-06	14:25
3	NA-29001	163	88-06-15	0:00
4	NA-29002	75100	88-06-15	0:22
5	NA-29003	163	88-06-15	0:00
6	NA-29004	75100	88-06-15	0:23
7	NA-29005	163	88-06-15	0:00
8	NA-29006	75100	88-06-20	0:10
9	NA-29007	163	88-06-15	0:00
10	NA-29008	75100	88-06-15	0:26

f.1 : 次ページ f.2 : 前ページ f.3 : 読み込み f.4 : 中 断 ↑ : UP ↓ : DOWN

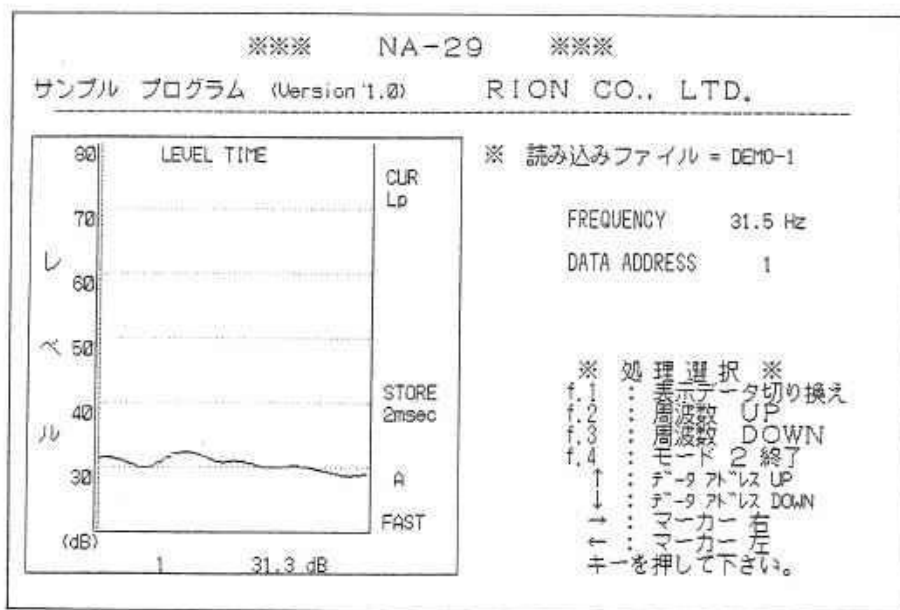
・モード1で入力したデータ



・モード2で入力したデータのレベル・フリケンシー表示



・モード2で入力したデータのレベル・タイム表示

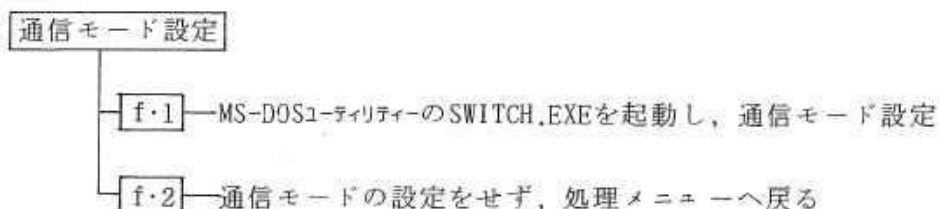


#### 4) NA-29 コマンド入力

NA-29をコントロールするコマンドを入力します。  
データが転送されると処理メニューに戻ります。

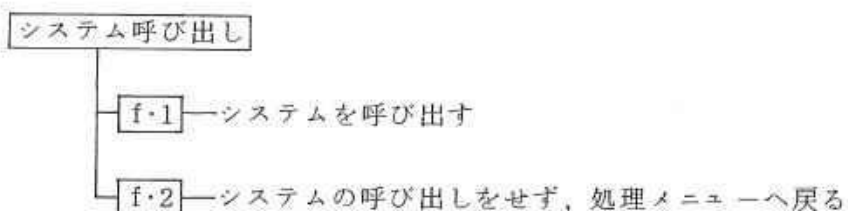
#### 5) RS-232C 通信モード設定

NA-29とPC間の通信モードの設定を行います。



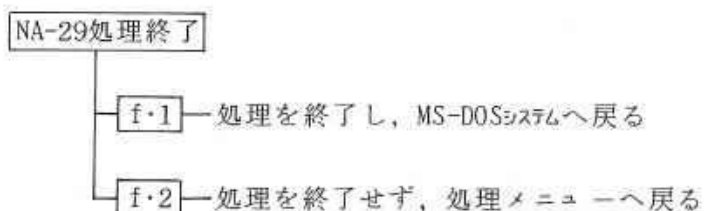
#### 6) システム 呼び出し

システムを呼び出し、処理を継続します。  
システムから抜ける場合はEXIT [RETURN]を入力します。



#### 7) NA-29 処理終了

NA-29のサンプル処理を終了します。



● デモンストレーションプログラム

ここに掲載したプログラムは一例ですので、これを参考にして各ユーザーがより使いやすいプログラムを作成して下さい。

```

1000 REM *****
1010 REM *****
1020 REM ***** < NA-29 デモンストレーション > *****
1030 REM *****
1040 REM ***** Version 1.0 *****
1050 REM *****
1060 REM ***** 1988 / 6 By RION Co., LTD. *****
1070 REM ***** Program name "NA-29.BAS" *****
1080 REM *****
1090 REM *****
1100 REM
1110 CLEAR &H100,&H100,&H2000,&H1000 ' System map set
1120 OPTION BASE 0 ' Dimension base 0
1130 DIM DOD$(10),DIRDAT$(100),DIR$(100) ' Dimension set
1140 DIM F.TABLE%(10,1500) ' "
1150 GOSUB *INITIALIZE ' RS-232C,KEY open
1160 GOSUB *MENU.SUB ' Farst Menu
1170 REM -----
1180 *PROCESS
1190 GOSUB *MENU ' Process menu
1200 FKEY=7:GOSUB *KEY.IN ' Function key in
1210 IF FM=1 THEN GOSUB *MODE1:GOTO 1180 ' Mode 1
1220 IF FM=2 THEN GOSUB *MODE2:GOTO 1180 ' Mode 2
1230 IF FM=3 THEN GOSUB *HYOUJI:GOTO 1180 ' Data display
1240 IF FM=4 THEN GOSUB *COMMAND.IN:GOTO 1180 ' Command input
1250 IF FM=5 THEN GOSUB *RS232C ' RS-232C CALL
1260 IF FM=6 THEN GOSUB *SYSTEM.CALL ' SYSTEM CALL
1270 IF FM=7 THEN GOSUB *SYSTEM.END ' NA-29 process end
1280 GOTO 1180
1290 REM
1300 REM
1310 REM *****
1320 REM ***** F. 1 : モード 1 *****
1330 REM *****
1340 *MODE1
1350 CLS : COLOR 3
1360 PRINT SPC(20);"1 / 1 オクターブモードで10秒LEQを測定し、"
1370 PRINT
1380 PRINT SPC(20);"その結果を読み込みます。"
1390 PRINT : PRINT : COLOR 5
1400 PRINT SPC(20);"f. 1 ";:COLOR 7:PRINT "処理を行う。"
1410 PRINT : COLOR 5
1420 PRINT SPC(20);"f. 2 ";:COLOR 7:PRINT "処理は行わない。"
1430 COLOR 6 : PRINT
1440 PRINT : PRINT SPC(20);"f. 1 又は、f. 2 を選択して下さい。"
1450 FKEY=2:GOSUB *KEY.IN
1460 IF FM=2 THEN RETURN
1470 GOSUB *START : IF TIMEOUT=1 THEN RETURN '
1480 PRINT #2,"EDR"
1490 REM EDR : リコール解除
1500 GOSUB *START : IF TIMEOUT=1 THEN RETURN '
1510 PRINT #2,"EDO"
1520 REM EDO : 二次演算解除

```



```

1530 CLS : PRINT : COLOR 4
1540 PRINT SPC(10);"↓ 初期データ送信 ↓":GOSUB *PBEEP
1550 GOSUB *START : IF TIMEOUT=1 THEN RETURN
1560 PRINT #2,"TRG0,0/TMC0/WGT0/RNG1/REF0,0"
1570 REM TRG0,0 : トリガー OFF
1580 REM TMC0 : 時定数 FAST
1590 REM WGT0 : ウェイト A
1600 REM RNG1 : レンジ FS = 80 dB
1610 REM REF0,0 : リファレンス OFF
1620 REM
1630 FOR I=1 TO 500:NEXT:PRINT : COLOR 5
1640 PRINT SPC(10);"↓ 10sec Leq 測定開始 ↓":GOSUB *PBEEP
1650 GOSUB *START : IF TIMEOUT=1 THEN RETURN
1660 PRINT #2,"SRT100"
1670 REM SRT100 : [XX] 時間
1680 REM : [X] 単位
1690 REM
1700 CK$="1":GOSUB *CHECK.PSI : IF TIMEOUT=1 THEN RETURN
1710 PRINT : COLOR 6
1720 PRINT SPC(10);"↓ 測定結果の受信 ↓":GOSUB *PBEEP
1730 GOSUB *START : IF TIMEOUT = 1 THEN RETURN
1740 PRINT #2,"PSI0/PSI1/PSI2/PSI3"
1750 REM PSI0 : 表示パラメーター 0
1760 REM PSI1 : 表示パラメーター 1
1770 REM PSI2 : 表示パラメーター 2
1780 REM PSI3 : 表示パラメーター 3
1790 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' receive
1800 DOD$(1) = H$
1810 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' receive
1820 DOD$(2) = H$
1830 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' receive
1840 DOD$(3) = H$
1850 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' receive
1860 DOD$(4) = H$
1870 GOSUB *START : IF TIMEOUT = 1 THEN RETURN
1880 PRINT #2,"DSP2/DOD"
1890 REM DSP2 : 表示データの種別
1900 REM : 0 = Lp
1910 REM : 1 = Lmax
1920 REM : 2 = Leq
1930 REM : 3 = LAE
1940 REM DOD : 表示データ要求
1950 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' 1 sample read
1960 DOD$(5) = H$ : COUNTER = 5 : MODE = 1
1970 GOSUB *START : IF TIMEOUT = 1 THEN RETURN
1980 PRINT #2,"DSPO"
1990 REM DSPO : 画面表示切り換え
2000 REM : 0 = Lp
2010 GOSUB *DATA.SAVE ' Data save
2020 RETURN
2030 REM
2040 REM

```

```

2050 REM *****
2060 REM F. 2 : モード 2
2070 REM *****
2080 *MODE2
2090 CLS : COLOR 3
2100 PRINT SPC(10);"2 m s e c 間隔で L p データをストアした後、レベルタイム!
で"
2110 PRINT
2120 PRINT SPC(10);"リコールしてそのデータを読み込みます。"
2130 PRINT
2140 PRINT SPC(10);"又、リコールは A P , 1 / 1 , スタートアドレスは 1 とする。
2150 PRINT : PRINT : COLOR 5
2160 PRINT SPC(20);" f . 1 " ; : COLOR 7 : PRINT "処理を行う。"
2170 PRINT : COLOR 5
2180 PRINT SPC(20);" f . 1 " ; : COLOR 7 : PRINT "処理は行わない。"
2190 COLOR 6 : PRINT
2200 PRINT : PRINT SPC(20);" f . 1 又は、 f . 2 を選択して下さい。"
2210 FKEY=2:GOSUB *KEY.IN
2220 IF FM=2 THEN RETURN
2230 CLS : PRINT : COLOR 4
2240 PRINT SPC(10);"↓ 初期 データ 送 信 ↓":GOSUB *PBEEP
2250 GOSUB *START : IF TIMEOUT=1 THEN RETURN
2260 PRINT #2,"EDR"
2270 REM EDR : リコール解除
2280 GOSUB *START : IF TIMEOUT=1 THEN RETURN
2290 PRINT #2,"EDO"
2300 REM EDO : 二次演算解除
2310 GOSUB *START : IF TIMEOUT=1 THEN RETURN
2320 PRINT #2,"TRG0,0/TMC0/WGT0/RNG1/REF0,0"
2330 REM TRG0,0 : トリガー OFF
2340 REM TMC0 : 時定数 FAST
2350 REM WGT0 : ウェイト FLAT
2360 REM RNG1 : レンジ FS = 80 dB
2370 REM REF0,0 : リフレッシュ OFF
2380 REM
2390 FOR I=1 TO 500:NEXT:PRINT : COLOR 5
2400 PRINT SPC(10);"↓ 2 msec L p のオートストア開始 ↓":GOSUB *PBEEP
2410 GOSUB *START : IF TIMEOUT = 1 THEN RETURN
2420 PRINT #2,"STAO,0"
2430 REM STAX,Y : X スタアデータの種別
2440 REM : 0 = L P
2450 REM : Y スタア間隔
2460 REM : 0 = 2 m s e c
2470 REM
2480 CK$="2":GOSUB *CHECK.PSI : IF TIMEOUT = 1 THEN RETURN
2490 PRINT : COLOR 6
2500 PRINT SPC(10);"↓ 測定結果の受信 ↓":GOSUB *PBEEP
2510 GOSUB *START : IF TIMEOUT = 1 THEN RETURN
2520 PRINT #2,"PS10/PS11/PS12/PS13"
2530 REM PS10 : 表示パラメーター 0
2540 REM PS11 : 表示パラメーター 1
2550 REM PS12 : 表示パラメーター 2
2560 REM PS13 : 表示パラメーター 3
2570 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' receive

```

```

2580 DOD$(1) = H$
2590 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' receive
2600 DOD$(2) = H$
2610 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' receive
2620 DOD$(3) = H$
2630 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' receive
2640 DOD$(4) = H$
2650 RESTORE *FREQ.TABLE
2660 FOR LI1=1 TO 10
2670 READ FREQ$
2680 FOR LI2=1 TO 1426 STEP 75
2690 CM1$=STR$(LI2):CM1$=RIGHT$(CM1$,LEN(CM1$)-1)
2700 CM2$=STR$(LI1-1):CM2$=RIGHT$(CM2$,1)
2710 GOSUB *START : IF TIMEOUT = 1 THEN RETURN
2720 PRINT #2,"RCT"+CM1$+", "+CM2$+",0/DOD"
2730 REM RCTX,Y,Z
2740 REM : X 先頭アドレス
2750 REM : 1 --> 1500
2760 REM : Y 周波数
2770 REM : 0: 31.5 --> 9: AP
2780 REM : Z 間引数
2790 REM : 0 = 1
2800 REM : 1 = 2個毎
2810 REM : 2 = 5 "
2820 REM : 3 =10 "
2830 REM : 4 =20 "
2840 REM
2850 LOCATE 14,15 : COLOR 7 : PRINT "読み込み周波数 = ";
2860 COLOR 6 : PRINT FREQ$
2870 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' 1 sample read
2880 A$=RIGHT$(I$,LEN(I$)-2)
2890 LOCATE 14,17 : COLOR 7 : PRINT "アドレス = " : COLOR 6
2900 FOR I=1 TO 40
2910 F.TABLE%(LI1,LI2+I-1) = VAL(MID$(A$,(I-1)*6+1,5))* 10
2920 LOCATE 36,17 : PRINT USING "#####";LI2 + I - 1
2930 NEXT
2940 FOR I=1 TO 35
2950 F.TABLE%(LI1,LI2+I+39) = VAL(MID$(H$,(I-1)*6+1,5))* 10
2960 LOCATE 36,17 : PRINT USING "#####";LI2 + I + 39
2970 NEXT
2980 NOKORI = 100 - ((LI1-1) * 1500 + LI2+I+39) / 150
2990 LOCATE 14,19 : COLOR 7 : PRINT "読み込み終了まであと " : COLOR 6
3000 LOCATE 36,19 : PRINT USING "#####.#";NOKORI;;PRINT " %"
3010 GOSUB *PBEEP
3020 NEXT
3030 NEXT
3040 COUNTER = 4 : MODE = 2
3050 GOSUB *START : IF TIMEOUT = 1 THEN RETURN
3060 PRINT #2,"EDR"
3070 REM EDR : リコール モード オフ
3080 GOSUB *DATA.SAVE ' Data save
3090 RETURN
3100 REM

```

```

3110 REM *****
3120 REM F. 3: データ表示
3130 REM *****
3140 *HYOUJI
3150 CLS : COLOR 3
3160 PRINT SPC(14);"モード1ならびにモード2でファイルに書き込まれた"
3170 PRINT SPC(14);"データの表示を行います。"
3180 PRINT : PRINT : COLOR 5
3190 PRINT SPC(20);"f. 1 ";:COLOR 7:PRINT "データの表示を行う。"
3200 PRINT : COLOR 5
3210 PRINT SPC(20);"f. 2 ";:COLOR 7:PRINT "データの表示は行わない。"
3220 COLOR 6 : PRINT
3230 PRINT : PRINT SPC(20);"f. 1 又は、f. 2 を選択して下さい。"
3240 FKEY=2:GOSUB *KEY.IN
3250 IF FM=2 THEN RETURN
3260 CLS : COLOR 3
3270 PRINT SPC(14);"データ ディスクに入っているデータ、又は、"
3280 PRINT SPC(14);"N A - 2 9 の瞬時データの表示を行います。"
3290 PRINT : PRINT : COLOR 5
3300 PRINT SPC(20);"f. 1 ";:COLOR 7:PRINT "データ ディスクの表示"
3310 PRINT : COLOR 5
3320 PRINT SPC(20);"f. 2 ";:COLOR 7:PRINT "N A - 2 9 の瞬時データの表示"
3330 PRINT : COLOR 5
3340 PRINT SPC(20);"f. 3 ";:COLOR 7:PRINT "データの表示は行わない。"
3350 COLOR 6 : PRINT
3360 PRINT : PRINT SPC(20);"f. 1 から、f. 3 を選択して下さい。"
3370 FKEY=3:GOSUB *KEY.IN
3380 IF FM=3 THEN RETURN
3390 IF FM=2 THEN GOSUB *SYUNJI.DISPLAY:RETURN
3400 CLS : PRINT
3410 COLOR 7 : PRINT "※ 読み込むデータのドライブNOを入力して下さい。(1
2:B 3:C 4:Cancel) ";
3420 COLOR 4:INPUT A:GOSUB *PBEEP:ON ERROR GOTO 3550
3430 IF A>0 AND A<5 THEN 3440 ELSE GOSUB *EBEEP:GOTO 3410
3440 IF A=4 THEN RETURN
3450 RESTORE *DRV.TABLE:FOR I=1 TO A:READ DRV$:NEXT : CLOSE #2
3460 OPEN DRV$+"$$$" FOR OUTPUT AS #2:CLOSE #2:KILL DRV$+"$$$"
3470 CHILD "DIR "+DRV$+" ; SORT > DIR.MAP"
3480 GOSUB *INITIALIZE
3490 GOSUB *MENU.SUB
3500 GOSUB *FILE.IN : IF ABORT=1 THEN 3410
3510 CLOSE #2 : OPEN "COM:" AS #2
3520 IF CANCEL = 1 THEN RETURN
3530 GOSUB *DISPLAY
3540 RETURN
3550 ER=ERR:EL=ERL:RESUME 3560
3560 ON ERROR GOTO 0
3570 COLOR 2:PRINT:PRINT SPC(16);"××× ドライブ指定が誤りです ×××":
UB *EBEEP
3580 PRINT:GOTO 3410

```

```

3590 REM
3600 REM *****
3610 REM F. 4: コマンド入力
3620 REM *****
3630 *COMMAND.IN
3640 CLS : COLOR 3
3650 PRINT SPC(10);"N A - 2 9 をコントロールするコマンドの入力を行います。"
3660 PRINT : PRINT : COLOR 7
3670 PRINT SPC(10);"★ コマンドを入力して下さい。 [中断は";:COLOR 6:PRINT
"[RETURN KEY]";:COLOR 7:PRINT "のみ]";
3680 COLOR 4:LINE INPUT F$:GOSUB *PBEEP
3690 IF F$="" THEN 3720
3700 GOSUB *START:IF TIMEOUT=1 THEN RETURN
3710 PRINT #2,F$ ' Command out
3720 FM = 0 :RETURN
3730 REM
3740 REM
3750 REM *****
3760 REM F. 5: R S - 2 3 2 C C A L L
3770 REM *****
3780 *RS232C
3790 CLS : COLOR 3
3800 PRINT SPC(15);"R S - 2 3 2 C の各データの設定を行います。"
3810 PRINT : PRINT : COLOR 5
3820 PRINT SPC(15);"f . 1 ";:COLOR 7:PRINT "R S - 2 3 2 C の設定を行う。"
3830 PRINT : COLOR 5
3840 PRINT SPC(15);"f . 2 ";:COLOR 7:PRINT "R S - 2 3 2 C の設定は行わない
。"
3850 COLOR 6 : PRINT
3860 PRINT : PRINT SPC(20);"f . 1 又は、f . 2 を選択して下さい。"
3870 FKEY=2:GOSUB *KEY.IN
3880 IF FM=2 THEN RETURN
3890 CONSOLE 0,25:CHILD "SWITCH" ' System call
3900 GOSUB *PBEEP:RUN
3910 REM
3920 REM
3930 REM *****
3940 REM F. 6: S Y S T E M C A L L
3950 REM *****
3960 *SYSTEM.CALL
3970 CLS : COLOR 3: PRINT SPC(15);"M S - D O S のシステム呼び出しを行います。
"
3980 PRINT : PRINT : COLOR 5
3990 PRINT SPC(15);"f . 1 ";:COLOR 7:PRINT "システムの呼び出しを行う。"
4000 PRINT : COLOR 5
4010 PRINT SPC(15);"f . 2 ";:COLOR 7:PRINT "システムの呼び出しは行わない。"
4020 COLOR 6 : PRINT
4030 PRINT : PRINT SPC(20);"f . 1 又は、f . 2 を選択して下さい。"
4040 FKEY=2:GOSUB *KEY.IN
4050 IF FM=2 THEN RETURN
4060 COLOR 4 : PRINT : PRINT
4070 PRINT SPC(10);"※ システムへ処理を移行します。何かキーを押して下さい
。"
4080 PRINT:PRINT SPC(18);"戻る時には、" E X I T " + [RETURN KEY]を押して下さい。";
4090 A$=INPUT$(1):GOSUB *PBEEP

```



```

4600 REM
4610 REM =====
4620 REM B E E P
4630 REM =====
4640 *PBEEP
4650 BEEP 1
4660 FOR B=1 TO 50:NEXT
4670 BEEP 0
4680 RETURN
4690 REM
4700 REM =====
4710 REM E R R O R B E E P
4720 REM =====
4730 *EBEEP
4740 FOR BB=1 TO 3
4750 BEEP 1
4760 FOR B=1 TO 100:NEXT
4770 BEEP 0
4780 NEXT
4790 RETURN
4800 REM
4810 REM =====
4820 REM 処 理 メ ニ ュ ー
4830 REM =====
4840 *MENU.SUB
4850 CONSOLE 0,25,0,1 ' Char line 25,not menu
4860 SCREEN 3,0,0,1 ' CRT mode 640x400
4870 WIDTH 80,25 ' Char size 80x25
4880 CLS 3 ' CRT Clear
4890 COLOR 5
4900 PRINT SPC(20);"*** N A - 2 9 ***";COLOR 7:PRINT
4910 PRINT "サンプル プログラム (Version 1.0) ";
4920 COLOR 2:PRINT "R I O N C O . , L T D . ";
4930 PRINT : COLOR 1 : FOR I=1 TO 10:KEY I,"":NEXT
4940 PRINT " -----
"
4950 CONSOLE 5,20
4960 RETURN
4970 REM
4980 *MENU
4990 CLS 3 : PRINT : COLOR 4: PRINT SPC(25);"《 処理メニュー 》"
5000 PRINT : COLOR 5
5010 PRINT SPC(20);"f . 1 : ";:COLOR 7:PRINT " モード 1 "
5020 PRINT : COLOR 5
5030 PRINT SPC(20);"f . 2 : ";:COLOR 7:PRINT " モード 2 "
5040 PRINT : COLOR 5
5050 PRINT SPC(20);"f . 3 : ";:COLOR 7:PRINT " データ表示"
5060 PRINT : COLOR 5
5070 PRINT SPC(20);"f . 4 : ";:COLOR 7:PRINT " N A - 2 9 コマンド入力"
5080 PRINT : COLOR 5
5090 PRINT SPC(20);"f . 5 : ";:COLOR 7:PRINT " 通信モードセット
5100 PRINT : COLOR 5
5110 PRINT SPC(20);"f . 6 : ";:COLOR 7:PRINT " システム呼び出し"
5120 PRINT : COLOR 5
5130 PRINT SPC(20);"f . 7 : ";:COLOR 7:PRINT " N A - 2 9 処理終了"
5140 PRINT : PRINT : COLOR 6

```

```

5150 PRINT SPC(20);"f. 1からf. 7までのキーを選択して下さい。";
5160 DSP.FLAG=0 : RETURN
5170 REM
5180 REM
5190 REM =====
5200 REM I N I T I A L I Z E
5210 REM =====
5220 *INITIALIZE
5230 OPEN "KYBD:" AS #1 ' Key in open
5240 OPEN "COM:" AS #2 ' RS-232C open
5250 ' Baudrate 9600bps
5260 ' Character size 8 bit
5270 ' Stop bit 2
5280 ' Parity none
5290 ' X parameter off
5300 ' ON STOP GOSUB *STOP.SUB : STOP ON
5310 ' ON HELP GOSUB *HELP.SUB : HELP ON
5320 RETURN
5330 *STOP.SUB : RETURN
5340 *HELP.SUB : RETURN
5350 REM
5360 REM =====
5370 REM コマンドの受信
5380 REM =====
5390 *RECV.COM:
5400 TIMEOUT = 0 : H$ = ""
5410 TIME$ = "00:00:00"
5420 IF LOC(2) <> 0 THEN 5520
5430 B$ = TIME$ : IF VAL(RIGHT$(B$,2)) < 4 THEN 5420
5440 IF DSP.FLAG=1 THEN 5460
5450 PRINT : COLOR 2 : PRINT SPC(20);"** Not receive (Error) !! **";
5460 GOSUB *EBEEP : TIMEOUT = 1 : TIME$ = "00:00:00"
5470 A$ = TIME$ : IF VAL(RIGHT$(A$,2)) < 2 THEN 5470
5480 CLOSE #2 ' Error close
5490 OPEN "COM:" AS #2 ' RS-232C open
5500 RETURN
5510 '
5520 A$ = INPUT$(1,#2)
5530 H$ = H$ + A$
5540 IF ASC(A$) <> 10 THEN 5410 ' Check if
5550 RETURN
5560 REM
5570 REM =====
5580 REM データの受信
5590 REM =====
5600 *RECV.DATA
5610 H$="" : TIMEOUT = 0
5620 TIME$ = "00:00:00"
5630 IF LOC(2) <> 0 THEN 5730
5640 A$ = TIME$ : IF VAL(RIGHT$(A$,2)) < 4 THEN 5630
5650 IF DSP.FLAG=1 THEN 5670
5660 PRINT : COLOR 2 : PRINT SPC(20);"** Not receive (Error) !! **";
5670 GOSUB *EBEEP : TIMEOUT = 1 : TIME$ = "00:00:00"

```



```

5680  A$ = TIMES : IF VAL(RIGHT$(A$,2)) < 2 THEN 5680
5690  CLOSE #2
5700  OPEN "COM:" AS #2
5710  RETURN
5720  '
5730  A$=INPUT$(1,#2)
5740  H$=H$+A$
5750  IF ASC(A$)<>10 THEN 5620
5760  A = LEN(H$)
5770  IF MID$(H$, A-2, 1) = CHR$(8H4) THEN 5810
5780  PRINT #2,CHR$(8H6);
5790  PRINT #2,"NEXT" : COLOR 7 : I$=LEFT$(H$,LEN(H$)-2) ' SEND ACK
5800  GOTO 5610
5810  H$ = LEFT$(H$,LEN(H$)-3) : RETURN
5820  REM
5830  REM =====
5840  REM                      測 定 デ ー タ の 格 納
5850  REM =====
5860 *DATA.SAVE
5870  CLS : COLOR 7 : GOSUB *PBEEP
5880  PRINT SPC(10);"※ 今、読み込まれたデータをディスクに格納しますか。"
5890  PRINT SPC(25);" [ Y : はい N : いいえ ] ";
5900  COLOR 4:INPUT A$:GOSUB *PBEEP
5910  IF A$="Y" OR A$="y" THEN 5920 ELSE RETURN
5920  COLOR 7 : PRINT : PRINT SPC(20);"※ 時を入力して下さい。 "; : COLOR 4
5930  INPUT J1 : GOSUB *PBEEP
5940  IF J1>0 AND J1<24 THEN 5950 ELSE GOSUB *EBEEP : GOTO 5920
5950  COLOR 7 : PRINT : PRINT SPC(20);"※ 分を入力して下さい。 "; : COLOR 4
5960  INPUT FUN : GOSUB *PBEEP
5970  IF FUN<60 THEN 5980 ELSE GOSUB *EBEEP : GOTO 5950
5980  J1$=STR$(J1) : J1$=RIGHT$(J1$,LEN(J1$)-1)
5990  IF LEN(J1$)<2 THEN J1$="0"+J1$ : GOTO 5990
6000  FUN$=STR$(FUN) : FUN$=RIGHT$(FUN$,LEN(FUN$)-1)
6010  IF LEN(FUN$)<2 THEN FUN$="0"+FUN$ : GOTO 6010
6020  TIME$=J1$+" ":"+FUN$+":00"
6030  COLOR 7:PRINT:PRINT SPC(10);"※ 格納するドライブNOを入力して下さい。
1:A 2:B 3:C) ";
6040  COLOR 4:INPUT A:GOSUB *PBEEP:ON ERROR GOTO 6360
6050  IF A>0 AND A<4 THEN 6060 ELSE GOSUB *EBEEP:GOTO 5980
6060  RESTORE *DRV.TABLE:FOR I=1 TO A:READ DRV$:NEXT : CLOSE #2
6070  OPEN DRV$+"$$$" FOR OUTPUT AS #2:CLOSE #2:KILL DRV$+"$$$"
6080  COLOR 7:PRINT:PRINT SPC(10);"※ 格納するファイル名を入力して下さい。
(最大8桁) ";
6090  COLOR 4:INPUT NAMA$:GOSUB *PBEEP
6100  IF LEN(NAMA$)>8 THEN GOSUB *EBEEP:GOTO 6080
6110  ON ERROR GOTO *DRV.ER1
6120  OPEN DRV$+NAMA$ FOR INPUT AS #2:CLOSE #2:GOSUB *EBEEP:GOTO 6300
6130  OPEN DRV$+NAMA$ FOR OUTPUT AS #2
6140  COLOR 6:LOCATE 20,20:PRINT " [  ただ今データを書き込み中です。  ] "
6150  PRINT #2,MODE
6160  FOR I1=1 TO COUNTER:PRINT #2,DOD$(I1):NEXT
6170  IF MODE = 1 THEN 6230
6180  FOR I=1 TO 10

```

```

6190     FOR J=1 TO 1500
6200         PRINT #2,F.TABLE%(I,J);
6210     NEXT
6220 NEXT
6230 CLOSE #2
6240 PRINT "     ディスクにデータが格納されました。"
6250 OPEN "COM:" AS #2:RETURN
6260 *DRV.TABLE:DATA "A:","B:","C:"
6270 *DRV.ER1
6280 ER=ERR:EL=ERL:RESUME 6290
6290 ON ERROR GOTO 0:GOTO 6130
6300 COLOR 2:PRINT "     ××× 同一のファイル名が存在しています。 ××
×"
6310 PRINT "     選択して下さい  [ 1 : 再入力  2 : 重ね書き ] ";
6320 COLOR 4:INPUT A:GOSUB *PBEEP
6330 IF A>0 AND A<3 THEN 6340 ELSE GOSUB *EBEEP:GOTO 6310
6340 IF A=1 THEN 6080
6350 KILL DRV$+NAMAES:GOTO 6130
6360 ER=ERR:EL=ERL:RESUME 6370
6370 ON ERROR GOTO 0
6380 COLOR 2:PRINT "     ××× ドライブ指定が誤りです ×××":GOSUB *1
EEP
6390 GOTO 5820
6400 REM
6410 REM =====
6420 REM           E N Q / A C K の ハ ン ド シ ェ イ ク
6430 REM =====
6440 *START
6450 H$="":PRINT #2,CHR$(&H5);                                ' SEND ENQ
6460 TIME$="00:00:00" : TIMEOUT = 0
6470 IF LOC(2) <> 0 THEN 6490
6480 A$=TIME$:IF VAL(RIGHT$(A$,2)) > 4 THEN 6510 ELSE 6470
6490 A$=INPUT$(1,#2):H$=H$+A$:IF ASC(A$)<>10 THEN 6470      ' Check LF
6500 RETURN
6510 IF DSP.FLAG=1 THEN 6540
6520 TIMEOUT=1 : PRINT : COLOR 2
6530 PRINT SPC(20);"** Not receive (Error) !! **";
6540 GOSUB *PBEEP : TIME$="00:00:00"
6550 A$=TIME$:IF VAL(RIGHT$(A$,2))> 2 THEN 6550
6560 CLOSE #2                                                ' Error close
6570 OPEN "COM:" AS #2                                       ' RS-232C open
6580 RETURN
6590 REM
6600 REM =====
6610 REM           デ ー タ フ ァ イ ル 読 み 込 み & 選 択
6620 REM =====
6630 *FILE.IN
6640 ON ERROR GOTO 7690 : ABORT = 0
6650 CLOSE #2:OPEN "DIR.MAP" FOR INPUT AS #2
6660 DIR.CNT=1:FOR I=1 TO 6:INPUT #2,A$:NEXT:FM=0
6670 INPUT #2,A$:IF EOF(2) THEN DIR$(DIR.CNT)=A$:DIR.CNT=DIR.CNT+1:GOTO 6690
6680 DIR$(DIR.CNT)=A$:DIR.CNT = DIR.CNT + 1:GOTO 6670
6690 K=1

```

```

6700 FOR I=1 TO DIR.CNT-1
6710 IF MID$(DIR$(I),19, 3)=" 0" THEN 6730
6720 IF MID$(DIR$(I), 9, 4)=" " THEN DIRDATA$(K)=DIR$(I):K=K+1
6730 NEXT
6740 DIR.CNT = K : IF K > 1 THEN 6780
6750 COLOR 2 : LOCATE 20, 10 : PRINT "■ データが入っていません。 ■"
6760 GOSUB *EBEEP : TIMES = "00:00:00" : CANCEL = 1
6770 A$ = TIMES : IF VAL(RIGHT$(A$,2)) < 2 THEN 6770 ELSE RETURN
6780 COUNT=1 : PAGE=1
6790 CLOSE #2:CLS : LOCATE 14,6 : DEF SEG=&HA200
6800 COLOR 4:PRINT "***** D I R E C T O R Y *****"
6810 PRINT : COLOR 6
6820 PRINT SPC(14);"NO ファイル名 容量 日付 時間"
6830 COLOR 1 : PRINT SPC(14);"-----"
6840 LOCATE 0,24:COLOR 5:PRINT "f.1 : ";;COLOR 7
6850 PRINT "次ページ ";;COLOR 5
6860 PRINT "f.2 : ";;COLOR 7:PRINT "前ページ ";;COLOR 5
6870 PRINT "f.3 : ";;COLOR 7:PRINT "読み込み ";;COLOR 5
6880 PRINT "f.4 : ";;COLOR 7:PRINT "中 断 ";;COLOR 5
6890 PRINT "↑ : ";;COLOR 7:PRINT "U P ";;COLOR 5
6900 PRINT "↓ : ";;COLOR 7:PRINT "D O W N";
6910 '
6920 COLOR 6 : CANCEL=0
6930 Y.AXIS =10 : YM =10 : LA = COUNT : PPOINT = 1
6940 COLOR 5:LOCATE 13,Y.AXIS:PRINT USING "### ";;COUNT;
6950 COLOR 7:PRINT DIRDATA$(COUNT)
6960 COUNT=COUNT+1:YM=YM+1:Y.AXIS=Y.AXIS+1
6970 IF COUNT>=DIR.CNT THEN 6990
6980 IF YM<20 THEN 6940
6990 MA = YM - 1
7000 KP=PPOINT*160+160*9+40
7010 FOR K=KP TO KP+80 STEP 2:POKE K,&HC5:NEXT
7020 FOR I=5 TO 10 : KEY (I) OFF : NEXT
7030 FOR I=1 TO 4 : KEY (I) ON : NEXT
7040 ON KEY GOSUB *F1,*F2,*F3,*F4
7050 IF LOC(1)=0 THEN 7120
7060 A$=INPUT$(1):GOSUB *PBEEP
7070 IF ASC(A$)=31 THEN GOSUB *UP:GOTO 7090
7080 IF ASC(A$)=30 THEN GOSUB *DOWN
7090 A=LOC(1):IF A=0 THEN 7040
7100 FOR I=1 TO A:A$=INPUT$(1):NEXT
7110 GOTO 7040
7120 IF FM=3 OR FM=4 THEN ON ERROR GOTO 0:RETURN
7130 IF FM=1 OR FM=2 THEN FM=0:CLS:GOTO 6790 ELSE 7040
7140 *F1
7150 IF COUNT>=DIR.CNT THEN FM=0:GOSUB *EBEEP ELSE FM=1:PAGE=PAGE+1:COUNT=(PAGE-1)*10+1:GOSUB *PBEEP
7160 RETURN
7170 *F2
7180 IF PAGE=1 THEN GOSUB *EBEEP:RETURN
7190 IF COUNT<>11 THEN PAGE=PAGE-1:COUNT=(PAGE-1)*10+1:FM=2:GOSUB *PBEEP ELSE
GOSUB *EBEEP
7200 RETURN

```

```

7210 *F3
7220 COLOR 6:LOCATE 20,22:PRINT " [   ただ今データを読み込み中です。   ] "
7230 FOR I=1 TO 10 : KEY (I) OFF : NEXT
7240 FM=3:GOSUB *PBEEP
7250 RFILE$=LEFT$(DIRDATA$((PAGE-1)*10+PPOINT),8):A=LEN(RFILE$)
7260 FOR I=1 TO A
7270   IF RIGHTS(RFILE$,1)<>" " THEN 7300
7280   A$=LEFT$(RFILE$,LEN(RFILE$)-1)
7290 NEXT
7300 ON ERROR GOTO *DO.ERR
7310 OPEN DRV$+RFILE$ FOR INPUT AS #2
7320 INPUT #2,MODE
7330 IF MODE=1 THEN 7410
7340 FOR I=1 TO 4:LINE INPUT #2,DOD$(I):NEXT:DR.CNT=4
7350 FOR I=1 TO 10
7360   FOR J=1 TO 1500
7370     INPUT #2,F.TABLE%(I,J)
7380   NEXT
7390 NEXT
7400 CLOSE #2:RETURN
7410 DR.CNT=1
7420 LINE INPUT #2,DOD$(DR.CNT):IF EOF(2) THEN CLOSE #2:RETURN
7430 DR.CNT=DR.CNT+1:GOTO 7420
7440 *DO.ERR
7450 ER=ERR:EL=ERL:RESUME 7460
7460 ON ERROR GOTO 0:LOCATE 0,18:GOSUB *EBEEP
7470 COLOR 2:PRINT "           X X           ファイル エラー           確認して下さい。
X X "
7480 COLOR 7:PRINT "                               ( 1 : 中断           2 : 再処理 ) ";
7490 COLOR 4:INPUT A:GOSUB *PBEEP
7500 IF A>0 AND A<3 THEN 7510 ELSE GOSUB *EBEEP:GOTO 7480
7510 IF A=1 THEN CANCEL=1 ELSE CANCEL=0:LOCATE 0,20:PRINT SPACE$(160);:FM=0
7520 OPEN "COM:" AS #2:RETURN
7530 *F4
7540 FM=4:GOSUB *PBEEP:CANCEL=1:RETURN
7550 *UP
7560 IF PPOINT = MA-9 THEN GOSUB *EBEEP:RETURN
7570 KP=PPOINT*160+160*9 +40
7580 FOR K=KP TO KP+80 STEP 2:POKE K,&HE1:NEXT
7590 PPOINT = PPOINT + 1:KP=PPOINT*160+160*9 +40
7600 FOR K=KP TO KP+80 STEP 2:POKE K,&HC5:NEXT
7610 RETURN
7620 *DOWN
7630 IF PPOINT = 1 THEN GOSUB *EBEEP:RETURN
7640 KP=PPOINT*160+160*9 +40
7650 FOR K=KP TO KP+80 STEP 2:POKE K,&HE1:NEXT
7660 PPOINT=PPOINT-1:KP=PPOINT*160+160*9 +40
7670 FOR K=KP TO KP+80 STEP 2:POKE K,&HC5:NEXT
7680 RETURN
7690 ER = ERR : ERN = ERL : RESUME 7700
7700 ON ERROR GOTO 0
7710 PRINT : COLOR 2
7720 PRINT SPC(13);"X X   データ ディスクではありません。 X X" : PRINT

```

```

7730 PRINT SPC(19);"確認して下さい。" : PRINT : ABORT = 1
7740 GOSUB *EBEEP : FOR I=1 TO 1000 : NEXT : RETURN
7750 REM
7760 REM =====
7770 REM                      L E Q の 終 了 確 認
7780 REM =====
7790 *CHECK.PSI
7800 TIME$="00:00:00" : OK = 0 : L = 0
7810 ON TIME$="00:00:02" GOSUB *TIME.CHECK
7820 TIME$ ON
7830 IF OK=1 THEN TIME$ OFF:RETURN ELSE GOTO 7810
7840 *TIME.CHECK
7850 GOSUB *START : IF TIMEOUT = 1 THEN RETURN
7860 PRINT #2,"PSI0"
7870 REM                      PSI0   : L E Q 終 了 確 認
7880 GOSUB *RECV.COM
7890 TIME$="00:00:00"
7900 IF TIMEOUT = 1 THEN L = L + 1 : IF L > 3 THEN TIME$ OFF:RETURN ELSE 7810
7910 IF MID$(H$,6,1)=CK$ THEN OK = 0 ELSE OK = 1
7920 RETURN
7930 REM
7940 REM =====
7950 REM                      測 定 デ ー タ の 表 示
7960 REM =====
7970 *DISPLAY
7980 CLS 3 : COLOR 7 : LOCATE 42,5
7990 PRINT "※ 読み込みファイル = "; : COLOR 6
8000 PRINT RFILE$
8010 LINE( 50, 80)-( 50,368),7
8020 LINE(255, 80)-(255,368),7
8030 LINE( 50,368)-(255,368),7
8040 FOR I=8*16 TO 8*16+48*4 STEP 16*3
8050   LINE( 50,I)-(255,I),7,,&H1111
8060 NEXT
8070 ON MODE GOSUB *MODE1.DISP,*MODE2.DISP
8080 RETURN
8090 REM
8100 REM =====
8110 REM                      モ ー ド 1 デ ー タ 表 示
8120 REM =====
8130 *MODE1.DISP
8140 COLOR 4:LOCATE 14,5:PRINT "1/1 OCTAVE"
8150 GOSUB *LINE.DISP           ' Line display
8160 GOSUB *REF.DISP           ' Reference display
8170 GOSUB *MARKER.DISP       ' Marker display
8180 GOSUB *WEIGHT.DISP       ' Weight display
8190 GOSUB *JITEISU.DISP      ' 時定数 表示
8200 GOSUB *TIME.DISP         ' Time display
8210 GOSUB *MODE.DISP         ' Mode display
8220 GOSUB *GRAPHICS.DISP     ' Graphics display
8230 GOSUB *NUMERIC.DISP      ' Numeric display
8240 COLOR 4 : LOCATE 51,20 : PRINT "※ 処理選択 ※"
8250 LOCATE 50,21:COLOR 5:PRINT "f.1" : "":COLOR 7:PRINT "モード 1 終了"

```

```

"
8260 LOCATE 50,22:COLOR 2:PRINT " → : ";:COLOR 7:PRINT "マーカー 右"
8270 LOCATE 50,23:COLOR 2:PRINT " ← : ";:COLOR 7:PRINT "マーカー 左"
8280 LOCATE 52,24:COLOR 6:PRINT "キーを押して下さい。";
8290 KEY (1) ON : FM = 0 : GOSUB *LOC.CLEAR
8300 ON KEY GOSUB *FK1 : IF FM=1 THEN KEY (1) OFF:RETURN
8310 IF LOC(1)=0 THEN 8300
8320 A$=INPUT$(1)
8330 IF ASC(A$)=29 OR ASC(A$)=30 THEN GOSUB *MRK.LEFT : GOTO 8290
8340 IF ASC(A$)=28 OR ASC(A$)=31 THEN GOSUB *MRK.RIGHT: GOTO 8290
8350 GOSUB *EBEEP : GOTO 8290
8360 REM
8370 REM          +++++ Function key 1 +++++
8380 *FK1
8390 GOSUB *PBEEP : FM=1 : RETURN
8400 REM
8410 REM          +++++ Marker left +++++
8420 *MRK.LEFT
8430 IF PT = 0 THEN RETURN
8440 GOSUB *PBEEP:LINE(62+PT*20,367)-(62+PT*20,80),0
8450 G = VAL(MID$(DOD$(5),PT*6+3,5))-LOW.LEVEL
8460 F = 368 - 4.8 * G : IF F < 80 THEN F = 80
8470 LINE(55+PT*20,F)-(70+PT*20,F),6
8480 LOCATE 48,PT+8:PRINT " "
8490 PT = PT - 1
8500 RESTORE *FREQ.TABLE
8510 FOR I=1 TO PT + 1 : READ FREQ$ : NEXT
8520 LOCATE 15,24 :COLOR 6 : PRINT FREQ$;
8530 B$=MID$(DOD$(5),PT*6+3,5)
8540 LINE(62+PT*20,367)-(62+PT*20,80),2,,&H5555
8550 LOCATE 24,24:PRINT B$;:COLOR 7:PRINT " dB";
8560 COLOR 6:LOCATE 48,PT+8:PRINT "*"
8570 RETURN
8580 REM
8590 REM          +++++ Marker right +++++
8600 *MRK.RIGHT
8610 IF PT = 9 THEN RETURN
8620 GOSUB *PBEEP:LINE(62+PT*20,367)-(62+PT*20,80),0
8630 G = VAL(MID$(DOD$(5),PT*6+3,5))-LOW.LEVEL
8640 F = 368 - 4.8 * G : IF F < 80 THEN F = 80
8650 LINE(55+PT*20,F)-(70+PT*20,F),6
8660 LOCATE 48,PT+8:PRINT " "
8670 PT = PT + 1
8680 RESTORE *FREQ.TABLE
8690 FOR I=1 TO PT + 1 : READ FREQ$ : NEXT
8700 LOCATE 15,24 :COLOR 6 : PRINT FREQ$;
8710 B$=MID$(DOD$(5),PT*6+3,5)
8720 LINE(62+PT*20,367)-(62+PT*20,80),2,,&H5555
8730 LOCATE 24,24:PRINT B$;:COLOR 7:PRINT " dB";
8740 COLOR 6:LOCATE 48,PT+8:PRINT "*"
8750 RETURN

```

```

8760 REM
8770 REM =====
8780 REM                      モ ー ド 2 デ ー タ 表 示
8790 REM =====
8800 *MODE2.DISP
8810 D.MODE = 1 : FREQ.POINT = 1 : FREQ.ADD = 0 : MARKER.POINT = 1 :PT=0
8820 COLOR 4 : LOCATE 12,5 : PRINT "LEVEL TIME"
8830 GOSUB *LINE.DISP                      ' Line display
8840 GOSUB *REF.DISP                      ' Reference display
8850 GOSUB *MARKER21.DISP                 ' Marker display
8860 GOSUB *WEIGHT.DISP                  ' Weight display
8870 GOSUB *JITEISU.DISP                 ' 時定数 表示
8880 GOSUB *TIME.DISP                    ' Time display
8890 GOSUB *MODE.DISP                    ' Mode display
8900 GOSUB *DATA.UP                      ' Data display
8910 GOSUB *FREQ.DISP                    ' Frequency display
8920 COLOR 4 : LOCATE 51,15 : PRINT "※ 処 理 選 択 ※"
8930 LOCATE 50,16:COLOR 5
8940 PRINT "f.1 : ";;COLOR 7:PRINT "表示データ切り換え"
8950 LOCATE 50,17:COLOR 5
8960 PRINT "f.2 : ";;COLOR 7:PRINT "周波数 UP"
8970 LOCATE 50,18:COLOR 5
8980 PRINT "f.3 : ";;COLOR 7:PRINT "周波数 DOWN"
8990 LOCATE 50,19:COLOR 5
9000 PRINT "f.4 : ";;COLOR 7:PRINT "モード 2 終了"
9010 LOCATE 50,20:COLOR 5
9020 PRINT " ↑ : ";;COLOR 7:PRINT "データアドレス UP"
9030 LOCATE 50,21:COLOR 5
9040 PRINT " ↓ : ";;COLOR 7:PRINT "データアドレス DOWN"
9050 LOCATE 50,22:COLOR 2
9060 PRINT " → : ";;COLOR 7:PRINT "マーカー 右"
9070 LOCATE 50,23:COLOR 2
9080 PRINT " ← : ";;COLOR 7:PRINT "マーカー 左"
9090 LOCATE 52,24:COLOR 6:PRINT "キーを押して下さい。";
9100 FOR K=1 TO 4:KEY (K) ON:NEXT
9110 GOSUB *LOC.CLEAR
9120 ON KEY GOSUB *FK21,*FK22,*FK23,*FK24
9130 IF FM=4 THEN FOR I=1 TO 4:KEY (I) OFF:NEXT:RETURN
9140 IF FM<4 AND FM>0 THEN FM=0 : GOTO 9100
9150 IF LOC(1)=0 THEN 9110
9160 A$=INPUT$(1)
9170 IF ASC(A$)=31 THEN GOSUB *DATA.DOWN :GOTO 9100
9180 IF ASC(A$)=30 THEN GOSUB *DATA.UP :GOTO 9100
9190 IF ASC(A$)=29 THEN GOSUB *MRK.LEFT2 : GOTO 9100
9200 IF ASC(A$)=28 THEN GOSUB *MRK.RIGHT2: GOTO 9100
9210 GOSUB *EBEEP : GOTO 9110
9220 REM
9230 REM          +++++ Function key 1 +++++
9240 *FK21
9250 GOSUB *PBEEP
9260 COLOR 4 : IF D.MODE=1 THEN D.MODE=2 ELSE D.MODE=1
9270 FOR I=1 TO 4 : KEY (I) OFF : NEXT
9280 IF D.MODE=2 THEN 9320
9290 LOCATE 12,5:PRINT "LEVEL TIME" : FREQ.ADD = FFF
9300 LOCATE 64,10:COLOR 6 :PRINT USING "#####";FREQ.ADD

```

```

9310 GOSUB *GRAPHICS1.DISP : FM=1 : RETURN
9320 LOCATE 12,5:PRINT "LEVEL FREQ"
9330 FFF = FREQ.ADD : FREQ.ADD=FREQ.ADD+MARKER.POINT-1
9340 IF FREQ.ADD>1500 THEN FREQ.ADD=1500
9350 LOCATE 64,10:COLOR 6 :PRINT USING "#####";FREQ.ADD
9360 GOSUB *GRAPHICS2.DISP : FM=1
9370 RETURN
9380 REM
9390 REM          +++++ Function key 2 +++++
9400 *FK22
9410 FM=2:FOR I=1 TO 4:KEY (I) OFF:NEXT
9420 IF FREQ.POINT = 10 THEN FREQ.POINT=0
9430 GOSUB *PBEEP:FREQ.POINT=FREQ.POINT+1
9440 GOSUB *FREQ.DISP
9450 IF D.MODE=2 THEN GOSUB *GRAPHICS2.DISP ELSE GOSUB *GRAPHICS1.DISP
9460 RETURN
9470 REM
9480 REM          +++++ Function key 3 +++++
9490 *FK23
9500 FM=3:FOR I=1 TO 4:KEY (I) OFF:NEXT
9510 IF FREQ.POINT = 1 THEN FREQ.POINT=11
9520 GOSUB *PBEEP:FREQ.POINT=FREQ.POINT-1
9530 GOSUB *FREQ.DISP
9540 IF D.MODE=2 THEN GOSUB *GRAPHICS2.DISP ELSE GOSUB *GRAPHICS1.DISP
9550 RETURN
9560 REM
9570 REM          +++++ Function key 4 +++++
9580 *FK24
9590 GOSUB *PBEEP : FM = 4 : RETURN
9600 REM
9610 REM          +++++ Data up +++++
9620 *DATA.UP
9630 AA=1:BB=1
9640 IF FREQ.ADD=1500 THEN RETURN
9650 FREQ.ADD=FREQ.ADD+1 : FOR I=1 TO 4 : KEY (I) OFF : NEXT
9660 GOSUB *PBEEP:LOCATE 50,10:COLOR 7:PRINT "DATA ADDRESS"
9670 LOCATE 64,10:COLOR 6 :PRINT USING "#####";FREQ.ADD
9680 IF D.MODE=1 THEN GOSUB *GRAPHICS1.DISP ELSE GOSUB *GRAPHICS2.DISP
9690 IF LOC(1)=0 THEN 9770
9700 IF FREQ.ADD=1500 THEN 9770
9710 S$=INPUT$(1):GOSUB *LOC.CLEAR : IF ASC(S$)<>30 THEN RETURN
9720 AA=AA+1:IF AA<5 THEN 9650
9730 AA=5:IF FREQ.ADD+10>1500 THEN 9770
9740 BB=BB+1:IF BB<5 THEN FREQ.ADD=FREQ.ADD+10:GOTO 9660
9750 BB=5:IF FREQ.ADD+100>1500 THEN BB=0:GOTO 9740
9760 FREQ.ADD=FREQ.ADD+100 : GOSUB *PBEEP : GOTO 9660
9770 FFF=FREQ.ADD : RETURN
9780 REM
9790 REM          +++++ Data down +++++
9800 *DATA.DOWN
9810 AA=1:BB=1
9820 IF FREQ.ADD=1 THEN RETURN
9830 FREQ.ADD=FREQ.ADD-1 : FOR I=1 TO 4 : KEY (I) OFF : NEXT

```



```

9840 GOSUB *PBEEP:LOCATE 50,10:COLOR 7:PRINT "DATA ADDRESS"
9850 LOCATE 64,10:COLOR 6 :PRINT USING "#####";FREQ.ADD
9860 IF D.MODE=1 THEN GOSUB *GRAPHICS1.DISPLAY ELSE GOSUB *GRAPHICS2.DISPLAY
9870 IF LOC(1)=0 THEN 9950
9880 IF FREQ.ADD=1 THEN 9950
9890 S$=INPUT$(1):GOSUB *LOC.CLEAR:IF ASC(S$)<>31 THEN RETURN
9900 AA=AA+1:IF AA<5 THEN 9830
9910 AA=5:IF FREQ.ADD-10<1 THEN 9950
9920 BB=BB+1:IF BB<5 THEN FREQ.ADD=FREQ.ADD-10:GOTO 9840
9930 BB=5:IF FREQ.ADD-100<1 THEN BB=0:GOTO 9920
9940 FREQ.ADD=FREQ.ADD-100 : GOSUB *PBEEP : GOTO 9840
9950 FFF=FREQ.ADD : RETURN
9960 REM
9970 REM          +++++ Marker left +++++
9980 *MRK.LEFT2
9990 IF D.MODE=2 THEN 10300
10000 AA=1
10010 IF MARKER.POINT=1 THEN RETURN ELSE GOSUB *PBEEP
10020 LINE(52+MARKER.POINT,367)-(52+MARKER.POINT,80),0
10030 S=FREQ.ADD+MARKER.POINT-1:IF S>1500 THEN G=0:GOTO 10050
10040 G = F.TABLE%(FREQ.POINT,S)/10-LOW.LEVEL
10050 F = 368 - 4.8 * G : IF F < 80 THEN F = 80
10060 PSET(52+MARKER.POINT,F),6
10070 MARKER.POINT = MARKER.POINT - 1
10080 S=FREQ.ADD+MARKER.POINT-1
10090 LINE(52+MARKER.POINT,367)-(52+MARKER.POINT,80),2,,&H5555
10100 LOCATE 7,24:COLOR 6
10110 IF S>1500 THEN PRINT "データ アドレス オーバー ";GOTO 10130
10120 PRINT USING "   ####          ####.# dB           ";MARKER.POINT+FREQ.ADD-1,F.T
ABLE%(FREQ.POINT,S)/10;
10130 FOR I=8*16 TO 8*16+48*4 STEP 16*3
10140 LINE( 50,I)-(255,I),7,,&H1111
10150 NEXT
10160 LINE( 50,368)-(255,368),7
10170 FOR I=1 TO 100:NEXT
10180 IF LOC(1)=0 THEN RETURN
10190 IF MARKER.POINT=1 THEN RETURN
10200 S$=INPUT$(1):GOSUB *LOC.CLEAR : IF ASC(S$)<>29 THEN RETURN
10210 AA=AA+1:IF AA<10 THEN GOSUB *PBEEP:GOTO 10020
10220 AA=9:IF MARKER.POINT-10<1 THEN RETURN
10230 LINE(52+MARKER.POINT,367)-(52+MARKER.POINT,80),0
10240 S=FREQ.ADD+MARKER.POINT-1:IF S>1500 THEN G=0:GOTO 10260
10250 G = F.TABLE%(FREQ.POINT,S)/10-LOW.LEVEL
10260 F = 368 - 4.8 * G : IF F < 80 THEN F = 80
10270 PSET(52+MARKER.POINT,F),6
10280 MARKER.POINT=MARKER.POINT-10 : GOSUB *PBEEP : GOTO 10080
10290 '
10300 IF PT = 0 THEN RETURN
10310 GOSUB *PBEEP:LINE(62+PT*20,367)-(62+PT*20,80),0
10320 G = F.TABLE%(PT+1,FREQ.ADD)/10-LOW.LEVEL
10330 F = 368 - 4.8 * G : IF F < 80 THEN F = 80
10340 LINE(55+PT*20,F)-(70+PT*20,F),6
10350 PT = PT - 1

```

```

10360 RESTORE *FREQ.TABLE
10370 FOR I=1 TO PT + 1 : READ FREQ$ : NEXT
10380 LOCATE 7,24 :COLOR 7:PRINT "周波数 " ;:COLOR 6 : PRINT FREQ$;" ";
10390 F=F.TABLE%(PT+1,FREQ.ADD)/10
10400 LINE(62+PT*20,367)-(62+PT*20,80),2,,&H5555
10410 LOCATE 24,24:PRINT USING "####.# dB";F;
10420 FREQ.POINT=PT+1:GOSUB *FREQ.DISP:RETURN
10430 REM
10440 REM          +++++ Marker right +++++
10450 *MRK.RIGHT2
10460 IF D.MODE=2 THEN 10770
10470 AA=1
10480 IF MARKER.POINT=200 THEN RETURN ELSE GOSUB *PBEEP
10490 LINE(52+MARKER.POINT,367)-(52+MARKER.POINT,80),0
10500 S=FREQ.ADD+MARKER.POINT-1:IF S>1500 THEN G=0:GOTO 10520
10510 G = F.TABLE%(FREQ.POINT,S)/10-LOW.LEVEL
10520 F = 368 - 4.8 * G : IF F < 80 THEN F = 80
10530 PSET(52+MARKER.POINT,F),6
10540 MARKER.POINT = MARKER.POINT + 1
10550 S=FREQ.ADD+MARKER.POINT-1
10560 LINE(52+MARKER.POINT,367)-(52+MARKER.POINT,80),2,,&H5555
10570 LOCATE 7,24:COLOR 6
10580 IF S>1500 THEN PRINT "データ アドレス オーバー " ;:GOTO 10600
10590 PRINT USING " ####          ####.# dB          ";MARKER.POINT+FREQ.ADD-1,F.
ABLE%(FREQ.POINT,S)/10;
10600 FOR I=8*16 TO 8*16+48*4 STEP 16*3
10610 LINE( 50,I)-(255,I),7,,&H1111
10620 NEXT
10630 LINE( 50,368)-(255,368),7
10640 FOR I=1 TO 100:NEXT
10650 IF LOC(1)=0 THEN RETURN
10660 IF MARKER.POINT=200 THEN RETURN
10670 S$=INPUT$(1):GOSUB *LOC.CLEAR : IF ASC(S$)<>28 THEN RETURN
10680 AA=AA+1:IF AA<10 THEN GOSUB *PBEEP:GOTO 10490
10690 AA=9:IF MARKER.POINT+10>200 THEN RETURN
10700 LINE(52+MARKER.POINT,367)-(52+MARKER.POINT,80),0
10710 S=FREQ.ADD+MARKER.POINT-1:IF S>1500 THEN G=0:GOTO 10730
10720 G = F.TABLE%(FREQ.POINT,S)/10-LOW.LEVEL
10730 F = 368 - 4.8 * G : IF F < 80 THEN F = 80
10740 PSET(52+MARKER.POINT,F),6
10750 MARKER.POINT=MARKER.POINT+10 : GOSUB *PBEEP :GOTO 10550
10760 '
10770 IF PT = 9 THEN RETURN
10780 GOSUB *PBEEP:LINE(62+PT*20,367)-(62+PT*20,80),0
10790 G=F.TABLE%(PT+1,FREQ.ADD)/10 - LOW.LEVEL
10800 F = 368 - 4.8 * G : IF F < 80 THEN F = 80
10810 LINE(55+PT*20,F)-(70+PT*20,F),6
10820 PT = PT + 1
10830 RESTORE *FREQ.TABLE
10840 FOR I=1 TO PT + 1 : READ FREQ$ : NEXT
10850 LOCATE 7,24 :COLOR 7:PRINT "周波数 " ;:COLOR 6 : PRINT FREQ$;" ";
10860 F=F.TABLE%(PT+1,FREQ.ADD)/10
10870 LINE(62+PT*20,367)-(62+PT*20,80),2,,&H5555

```

```

10880 LOCATE 24,24:PRINT USING "####.# dB";F;
10890 FREQ.POINT=PT+1:GOSUB *FREQ.DISP:RETURN
10900 REM
10910 REM =====
10920 REM Frequency display
10930 REM =====
10940 *FREQ.DISP
10950 LOCATE 50,8:COLOR 7:PRINT "FREQUENCY"
10960 RESTORE *FREQ.TABLE
10970 FOR I=1 TO FREQ.POINT:READ A$:NEXT
10980 LOCATE 64,8:COLOR 6:PRINT A$ :RETURN
10990 REM
11000 REM =====
11010 REM Line display
11020 REM =====
11030 *LINE.DISP
11040 LINE(0,75)-(320,399),7,B
11050 LINE( 50, 80)-( 50,368),7
11060 LINE(255, 80)-(255,368),7
11070 LINE( 50,368)-(255,368),7
11080 FOR I=8*16 TO 8*16+48*4 STEP 16*3
11090 LINE( 50,I)-(255,I),7,,&H1111
11100 NEXT
11110 RETURN
11120 REM
11130 REM =====
11140 REM Reference display
11150 REM =====
11160 *REF.DISP
11170 IF MID$(DOD$(3),6,1)="0" THEN 11190
11180 A = VAL(MID$(DOD$(3),8,3)):GOTO 11220
11190 REF=7-VAL(MID$(DOD$(2),5,1))
11200 RESTORE *REF.TABLE
11210 FOR I=1 TO REF+1:READ A:NEXT
11220 COLOR 6:LOCATE 2, 5:PRINT USING "####";A:A=A-10
11230 LOCATE 2, 8:PRINT USING "####";A:A=A-10
11240 LOCATE 2,11:PRINT USING "####";A:A=A-10
11250 LOCATE 2,14:PRINT USING "####";A:A=A-10
11260 LOCATE 2,17:PRINT USING "####";A:A=A-10
11270 LOCATE 2,20:PRINT USING "####";A:LOW.LEVEL=A-10
11280 LOCATE 3,23 : COLOR 7 : PRINT "(dB)"
11290 LOCATE 1,10:PRINT "レ":LOCATE 1,14:PRINT "ベ":LOCATE 1,18:PRINT "ル"
11300 RETURN
11310 REM
11320 REM =====
11330 REM Marker display
11340 REM =====
11350 *MARKER.DISP
11360 RESTORE *FREQ.TABLE
11370 PT=VAL(MID$(DOD$(3),3,2)) : FOR I=1 TO PT + 1 : READ FREQ$ : NEXT
11380 LOCATE 7,24 : COLOR 7 : PRINT "周波数 ";:COLOR 6 : PRINT FREQ$;" ";
11390 B$=MID$(DOD$(5),PT*6+3,5) : LINE(62+PT*20,367)-(62+PT*20,80),2,,&H5555
11400 LOCATE 24,24 : PRINT B$;:COLOR 7:PRINT " dB";

```

```

11410 COLOR 6:LOCATE 48,PT+8:PRINT "*"
11420 RETURN
11430 *MARKER1.DISP
11440 S=FREQ.ADD+MARKER.POINT-1:IF S>1500 THEN F=0:GOTO 11460
11450 F = F.TABLE*(FREQ.POINT,S)/10-LOW.LEVEL
11460 LINE(52+MARKER.POINT,367)-(52+MARKER.POINT,80),2,,&H5555
11470 LOCATE 7,24:COLOR 6
11480 IF S>1500 THEN PRINT "データ アドレス オーバー ";:RETURN
11490 PRINT USING " ####      ##.# dB      ";MARKER.POINT+FREQ.ADD-1,F
ABLE*(FREQ.POINT,FREQ.ADD+MARKER.POINT-1)/10;
11500 RETURN
11510 *MARKER21.DISP
11520 RESTORE *FREQ.TABLE
11530 PT=VAL(MID$(DOD$(3),3,2)):FOR I=1 TO PT + 1 : READ FREQ$: NEXT
11540 LOCATE 7,24 : COLOR 7 : PRINT "周波数 ";:COLOR 6 : PRINT FREQ$;"
11550 BS=MID$(DOD$(5),PT*6+3,5) : LINE(62+PT*20,367)-(62+PT*20,80),2,,&H5555
11560 LOCATE 24,24 : PRINT BS$:;COLOR 7:PRINT " dB";
11570 RETURN
11580 REM =====
11590 REM                               Weight display
11600 REM =====
11610 *WEIGHT.DISP
11620 RESTORE *WEIGHT.TABLE
11630 FOR I=1 TO VAL(MID$(DOD$(2),12,1))+1 : READ WEIGHT$: NEXT
11640 LOCATE 33,20 : PRINT WEIGHT$
11650 RETURN
11660 REM
11670 REM =====
11680 REM                               時定数 Display
11690 REM =====
11700 *JITEISU.DISP
11710 RESTORE *JITEISU.TABLE
11720 FOR I=1 TO VAL(MID$(DOD$(2),14,1))+1 : READ JITEISU$: NEXT
11730 LOCATE 33,22 : PRINT JITEISU$
11740 RETURN
11750 REM
11760 REM =====
11770 REM                               Sample Time Display
11780 REM =====
11790 *TIME.DISP
11800 RESTORE *SAMPLE.TABLE
11810 FOR I=1 TO VAL(MID$(DOD$(4),17,2))+1 : READ SAMPLE.TABLE$: NEXT
11820 LOCATE 33,17 : PRINT SAMPLE.TABLE$
11830 LOCATE 33,16 : PRINT "STORE"
11840 LOCATE 33,6:PRINT "CUR"
11850 RETURN
11860 REM
11870 REM =====
11880 REM                               Mode Display
11890 REM =====
11900 *MODE.DISP
11910 RESTORE *HYOUJI.TABLE
11920 FOR I=1 TO VAL(MID$(DOD$(1),3,1))+1 : READ HYOUJI$: NEXT

```

```

11930 LOCATE 33, 7 : PRINT HYOUJIS
11940 RETURN
11950 REM
11960 REM =====
11970 REM Graphic Display
11980 REM =====
11990 *GRAPHICS.DISP
12000 FOR I=1 TO 10
12010 G = VAL(MID$(DOD$(5), (I-1)*6+3,5))-LOW.LEVEL
12020 F = 368 - 4.8 * G : IF F < 80 THEN F = 80
12030 LINE(55+(I-1)*20,F)-(70+(I-1)*20,F),6
12040 LINE(55+(I-1)*20,368)-(55+(I-1)*20,F),6
12050 LINE(70+(I-1)*20,368)-(70+(I-1)*20,F),6
12060 NEXT
12070 RETURN
12080 *GRAPHICS1.DISP
12090 J=1:LINE(51,77)-(254,367),0,BF
12100 FOR I=8*16 TO 8*16+48*4 STEP 16*3
12110 LINE( 50,I)-(255,I),7,,&H1111
12120 NEXT
12130 GOSUB *MARKER1.DISP
12140 FOR I=FREQ.ADD TO FREQ.ADD+199
12150 IF I>1500 THEN 12200
12160 F = F.TABLE%(FREQ.POINT,I)/10-LOW.LEVEL
12170 G = 368 - 4.8 * F : IF G < 80 THEN G = 80
12180 PSET (52+J,G),6:J=J+1
12190 NEXT
12200 LINE( 50,368)-(255,368),7
12210 RETURN
12220 *GRAPHICS2.DISP
12230 LINE(51,77)-(254,367),0,BF
12240 FOR I=1 TO 10
12250 IF FREQ.ADD>1500 THEN 12330
12260 F = F.TABLE%(I,FREQ.ADD)/10-LOW.LEVEL
12270 G = 368-4.8*F : IF G < 80 THEN G = 80
12280 LINE(55+(I-1)*20,G)-(70+(I-1)*20,G),6
12290 LINE(55+(I-1)*20,368)-(55+(I-1)*20,G),6
12300 LINE(70+(I-1)*20,368)-(70+(I-1)*20,G),6
12310 NEXT
12320 F=F.TABLE%(PT+1,FREQ.ADD)/10
12330 LINE( 50,368)-(255,368),7
12340 RESTORE *FREQ.TABLE
12350 FOR I=1 TO PT+1:READ FREQ$;NEXT
12360 LOCATE 7,24 :COLOR 7:PRINT "周波数 ";:COLOR 6 : PRINT FREQ$;" ";
12370 LOCATE 24,24:PRINT USING "####.# dB";F;
12380 FOR I=8*16 TO 8*16+48*4 STEP 16*3
12390 LINE( 50,I)-(255,I),7,,&H1111
12400 NEXT
12410 LINE(62+PT*20,367)-(62+PT*20,80),2,,&H5555
12420 RETURN

```

```

12430 REM
12440 REM =====
12450 REM Numeric Display
12460 REM =====
12470 *NUMERIC.DISP
12480 LOCATE 48, 6:COLOR 4:PRINT "=====
12490 LOCATE 50,7 : COLOR 7 : PRINT "周波数 レベル"
12500 RESTORE *FREQ.TABLE
12510 FOR I=1 TO 10
12520 READ FREQ$:COLOR 5 : LOCATE 50,7+I : PRINT FREQ$
12530 COLOR 6 : LOCATE 62,7+I : PRINT MID$(DOD$(5),3+(I-1)*6,5);
12540 COLOR 5:PRINT " dB"
12550 NEXT
12560 LOCATE 48,18:COLOR 4:PRINT "=====
12570 RETURN
12580 REM
12590 REM =====
12600 REM Inst data display
12610 REM =====
12620 *SYUNJI.DISP
12630 CLS : COLOR 4
12640 PRINT SPC(10);"↓ 初期データ送信 ↓":GOSUB *PBEEP
12650 GOSUB *START : IF TIMEOUT=1 THEN RETURN
12660 PRINT #2,"TRG0,0/OCT/TMCO/WGT0/RNG1/REF0,0/DSP0"
12670 REM TRG0,0 : トリガー OFF
12680 REM OCT : 1/1 OCT測定モード
12690 REM TMCO : 時定数 FAST
12700 REM WGT0 : ウェイト FLAT
12710 REM RNG1 : レンジ FS = 80 dB
12720 REM REF0,0 : リファレンス OFF
12730 REM DSP0 : Lp 表示"
12740 REM
12750 FOR I=1 TO 1000:NEXT:PRINT:COLOR 5
12760 PRINT SPC(10);"↓ 瞬時データの受信 ↓":GOSUB *PBEEP
12770 GOSUB *START : IF TIMEOUT=1 THEN RETURN
12780 PRINT #2,"RMT/PSI0/PSI1/PSI2/PSI3/DOD"
12790 REM RMT : リモート
12800 REM PSI0 : 表示パラメーター 0
12810 REM PSI1 : 表示パラメーター 1
12820 REM PSI2 : 表示パラメーター 2
12830 REM PSI3 : 表示パラメーター 3
12840 REM DOD : データ要求
12850 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' Data receive
12860 DOD$(1) = H$
12870 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' Data receive
12880 DOD$(2) = H$
12890 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' Data receive
12900 DOD$(3) = H$
12910 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' Data receive
12920 DOD$(4) = H$
12930 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN RETURN ' Data receive
12940 DOD$(5) = H$:CLS
12950 LINE( 50, 80)-( 50,368),7
12960 LINE(255, 80)-(255,368),7
12970 LINE( 50,368)-(255,368),7
12980 GOSUB *LINE.DISP ' Line display

```

```

12990 GOSUB *REF.DISPLAY          ' Reference display
13000 GOSUB *WEIGHT.DISPLAY      ' Weight display
13010 GOSUB *JITEISU.DISPLAY    ' 時定数 表示
13020 GOSUB *TIME.DISPLAY       ' Time display
13030 GOSUB *MODE.DISPLAY       ' Mode display
13040 KEY (1) ON : FM=0 : DSP.FLAG = 1
13050 COLOR 4 : LOCATE 51,20 : PRINT "※ 処理選択 ※"
13060 LOCATE 48,22:COLOR 5:PRINT "f . 1 ";;COLOR 7:PRINT "データ表示終了"
13070 COLOR 7:LOCATE 50,5:PRINT "★ 瞬時データ表示中 ★"
13080 GOSUB *GRAPHICS.DISPLAY    ' Graphics display
13090 GOSUB *MARKER.DISPLAY     ' Marker display
13100 GOSUB *NUMERIC.DISPLAY    ' Numeric display
13110 GOSUB *PBEEP
13120 GOSUB *START : IF TIMEOUT=1 THEN RETURN
13130 PRINT #2,"RMT/DOD"
13140 REM                          RMT      : REMOTE
13150 REM                          DOD      : データ要求
13160 LOCATE 40,24 : PRINT SPACE$(39);
13170 GOSUB *RECV.DATA : IF TIMEOUT=1 THEN 13120
13180 DOD$(5) = H$
13190 LINE(51,76)-(254,367),0,BF
13200 LINE( 50,368)-(255,368),7
13210 FOR I=8*16 TO 8*16+48*4 STEP 16*3
13220   LINE( 50,I)-(255,I),7,,&H1111
13230 NEXT
13240 ON KEY GOSUB *FK1
13250 IF FM<>1 THEN GOSUB *LOC.CLEAR:GOTO 13080
13260 GOSUB *PBEEP
13270 GOSUB *START : IF TIMEOUT=1 THEN RETURN
13280 PRINT #2,"LOC"
13290 REM                          LOC      : LOCAL
13300 RETURN
13310 REM
13320 REM ***** Data table *****
13330 REM
13340 REM Level Range
13350 *REF.TABLE
13360 DATA 140,130,120,110,100, 90, 80, 70, 60, 50, 40, 30, 20, 10
13370 REM Weight
13380 *WEIGHT.TABLE
13390 DATA " A ", " C ", "FLAT"
13400 REM 時定数
13410 *JITEISU.TABLE
13420 DATA "FAST", "SLOW", "10msec"
13430 REM 表示データの種別
13440 *HYOUJI.TABLE
13450 DATA "Lp", "Lmax", "Leq", "LAE"
13460 REM 周波数
13470 *FREQ.TABLE
13480 DATA " 31.5 Hz", " 63 Hz", "125 Hz", "250 Hz", "500 Hz"
13490 DATA " 1 KHz", " 2 KHz", " 4 KHz", " 8 KHz", " AP "
13500 *SAMPLE.TABLE
13510 DATA "2msec", "5msec", "10msec", "20msec", "50msec"
13520 DATA "100msec", "0.2sec"
13530 DATA "0.5sec", "1sec", "2sec", "5sec", "10sec"

```



## リオン株式会社

本社/営業部 東京都国分寺市東元町3丁目20番41号  
☎185-8533 TEL (042) 359-7887 (代表)  
FAX (042) 359-7441

- 東京支店/東京都渋谷区代々木2丁目7番7号 池田ビル  
☎151-0053 TEL(03)3379-5521 (代表) FAX(03)3370-4830
- 大阪営業所/大阪市北区西天満6丁目8番7号 電子会館ビル  
☎530-0047 TEL(06)6364-3671 (代表) FAX(06)6364-3673
- 仙台営業所/仙台市太白区南大野田25番地13  
☎982-0015 TEL(022)249-5533 (代表) FAX(022)249-5535
- 名古屋営業所/名古屋市中区丸の内2丁目3番23号 和波ビル  
☎460-0002 TEL(052)232-0470 (代表) FAX(052)232-0458
- 広島営業所/広島市中区宝町1番15号 宝町ビル  
☎730-0044 TEL(082)243-8899 (代表) FAX(082)243-8845
- 九州リオン株式会社/福岡市博多区店屋町5-22 朝日生命福岡第2ビル  
☎812-0025 TEL(092)281-5366 (代表) FAX(092)291-2847