

WT1600FC

**インピーダンスメータ
通信インタフェース**

U S E R ' S M A N U A L

ユーザーズマニュアル

はじめに

このたびは、インピーダンスメータWT1600FCをお買い上げいただきましてありがとうございました。

この通信インタフェースユーザーズマニュアルは、GP-IB/シリアル/イーサネットインタフェースの機能やコマンドについて説明したものです。ご使用前にこのマニュアルをよくお読みいただき、正しくお使いください。

お読みになったあとは大切に保存してください。ご使用中に操作がわからなくなったときなどにきっとお役に立ちます。

なお、WT1600FCのマニュアルとして、このマニュアルを含め、次の2冊があります。あわせてお読みください。

マニュアル名	マニュアルNo.	内 容
WT1600FCインピーダンスメータ ユーザーズマニュアル	IM 760151-01	通信機能を除く全機能とその操作方法について説明しています。
WT1600FCインピーダンスメータ 通信インタフェース ユーザーズマニュアル	IM 760151-17	本CD-Rです。GP-IB/シリアル/ イーサネットインタフェースの通信 機能について説明しています。

ご注意

- 本書の内容は、性能・機能の向上などにより、将来予告なしに変更することがあります。
- 本書の内容に関しては万全を期していますが、万一ご不審の点や誤りなどお気づきのことがありましたら、お手数ですが、最寄りの当社支社・支店・営業所までご連絡ください。
- 本書の内容の全部または一部を無断で転載、複製することは禁止されています。

商 標

- MS-DOSまたはVisual Basicは、米国Microsoft Corporationの、米国およびその他の国における登録商標または商標です。
- Adobe, Adobe AcrobatおよびPostScriptは、Adobe Systems Incorporated(アドビシステムズ社)の商標または登録商標です。
- 本文中の各社の登録商標または商標には、TM, ®マークは表示していません。
- その他、本文中に使われている会社名、商品名は、各社の登録商標または商標です。

履 歴

- 2003年5月 初版発行

このマニュアルの利用方法

このマニュアルの構成

このユーザズマニュアルは、以下に示す1章～7章および付録・索引で構成されています。

第1章 GP-IBインタフェースについて

GP-IBインタフェースの機能・仕様などについて説明しています。

第2章 シリアルインタフェースについて

シリアルインタフェースの機能・仕様などについて説明しています。

第3章 イーサネットインタフェースについて

イーサネットインタフェースの機能・仕様などについて説明しています。

第4章 プログラムを組む前に

コマンドを送るときの書式などについて説明しています。

第5章 通信コマンドを使う

使用できる全コマンドについて1つずつ説明しています。

第6章 ステータスレポート

ステータスバイトや各種レジスタ、キューなどについて説明しています。

第7章 サンプルプログラム

Windows PC(GP-IBボードはNI社製のAT-GPIB/TNT IEEE-488.2を使用)を使用して、Visual Basicで組んだプログラム例を紹介しています。

付 録

ASCIIキャラクタコード表などの参考資料を紹介しています。

索 引

アルファベット-50音順の索引があります。

このマニュアルで使用している記号

● 注記やキーを表す記号

種類	記号	意 味
単位	k	1000 例：100kHz
	K	1024 例：459Kバイト(ファイルのデータサイズ)
注記	Note	本機器を取り扱ううえで重要な情報が記載されています。
キー	Communication	操作対象の画面上の文字を表します。

● 構文の記号

主に5章の構文で使用している記号を下表に示します。なお、これはBNF(Backus-Naur Form)記号と呼ばれるものです。詳しいデータについては、4-5～4-6ページを参照してください。

記号	意 味	例	入力例
<x>	定義された値	ELEMeNt<x> <x>=1～5	->ELEMENT2
{ } 	{ }内から1つを選択 排他的論理和	HCOPY: {TIFF BMP} ?	->HCOPY:TIFF?
[]	省略可能	CURSOr [:TYPE]	->CURSor

目次

はじめに	i
このマニュアルの利用方法	ii
第1章 GP-IBインタフェースについて	
1.1 各部の名称と機能	1-1
1.2 GP-IBケーブルの接続方法	1-2
1.3 GP-IBインタフェースの機能	1-3
1.4 GP-IBインタフェースの仕様	1-4
1.5 アドレスの設定	1-5
1.6 インタフェースメッセージに対する応答	1-6
第2章 シリアルインタフェースについて	
2.1 各部の名称と機能	2-1
2.2 シリアルインタフェースの機能と仕様	2-2
2.3 シリアルインタフェースによる接続	2-3
2.4 ハンドシェイク方式の組み合わせ	2-5
2.5 データフォーマットの組み合わせ	2-7
2.6 シリアル通信の設定	2-8
第3章 イーサネットインタフェースについて	
3.1 各部の名称と機能	3-1
3.2 イーサネットインタフェースの機能と仕様	3-2
3.3 イーサネットコントロールの設定	3-4
第4章 プログラムを組む前に	
4.1 メッセージ	4-1
4.2 命令	4-3
4.3 応答	4-4
4.4 データ	4-5
4.5 コントローラとの同期	4-7
第5章 通信コマンド	
5.1 コマンド一覧表	5-1
5.2 COMMunicate グループ	5-10
5.3 CURSorグループ	5-12
5.4 DISPlayグループ	5-15
5.5 FILEグループ	5-25
5.6 HCOPyグループ	5-30
5.7 HOLDグループ	5-34
5.8 IMAGeグループ	5-35
5.9 IMPedanceグループ	5-36
5.10 INPutグループ	5-43
5.11 INTEGrateグループ	5-53
5.12 MEASureグループ	5-57
5.13 NUMericグループ	5-62
5.14 RATEグループ	5-68
5.15 STATus グループ	5-69

5.16	STOReグループ	5-71
5.17	SYSTemグループ	5-76
5.18	WAVeformグループ	5-79
5.19	WSETup (Wave SETup)グループ	5-81
5.20	共通コマンドグループ	5-84

第6章 ステータスレポート

6.1	ステータスレポートについて	6-1
6.2	ステータスバイト	6-2
6.3	標準イベントレジスタ	6-3
6.4	拡張イベントレジスタ	6-4
6.5	出力キューとエラーキュー	6-5

第7章 サンプルプログラム

7.1	プログラムを組む前に	7-1
7.2	サンプルプログラムイメージ	7-2
7.3	初期化・エラー・実行関数	7-3
7.4	電力測定データの出力	7-6
7.5	インピーダンス測定データの出力	7-10
7.6	波形データ(ASCII形式)の出力	7-14
7.7	波形データ(FLOAT形式)の出力	7-17

付録

付録1	ASCIIキャラクタコード	付-1
付録2	エラーメッセージ	付-2
付録3	IEEE 488.2-1987について	付-5

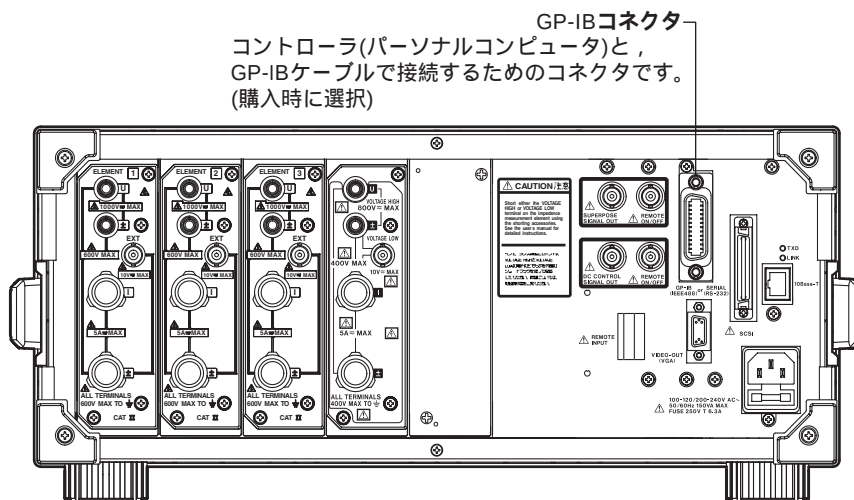
索引

1.1 各部の名称と機能

フロントパネル



リアパネル



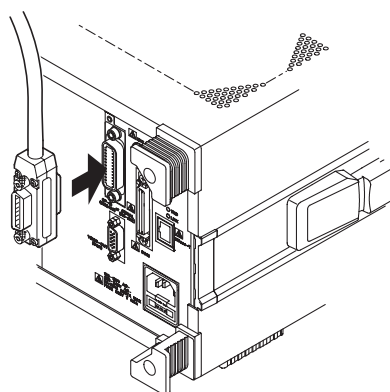
1.2 GP-IBケーブルの接続方法

GP-IBケーブル

本機器のGP-IBコネクタは、IEEE St'd 488-1978規格の24ピンコネクタです。GP-IBケーブルは、IEEE St'd 488-1978に合ったものを使用してください。

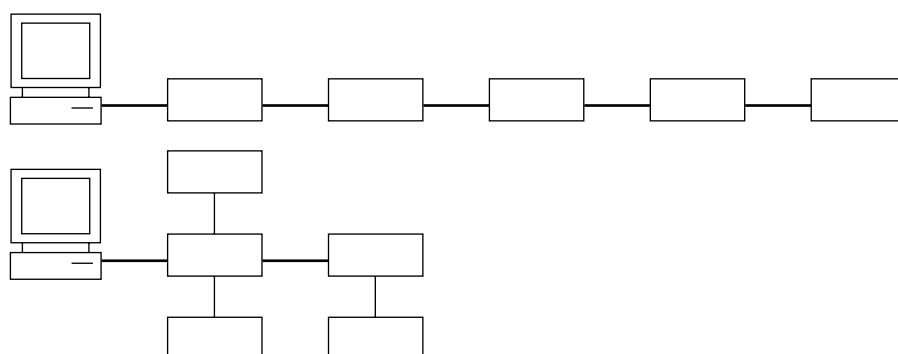
接続方法

下図のように接続してください。



接続時の注意

- GP-IBケーブルのコネクタに付いているねじは、しっかりと固定してください。
- 何本かのケーブルを接続して、複数の機器を接続することができます。ただし、1つのバス上にコントローラを含め15台以上の機器を接続することはできません。
- 複数の機器を接続するときは、それぞれのアドレスを同じに設定することはできません。
- 機器間をつなぐケーブルは2m以下のものを使用してください。
- ケーブルの長さは合計で20mを超えないようにしてください。
- 通信を行っているときは、少なくとも全体の2/3以上の機器の電源をONにしておいてください。
- 複数の機器を接続するときは、下図に示すようなスター形またはリニア形の結線にしてください。ループ形やパラレル形の結線はできません。



注 意

通信ケーブルを接続したり、取り外したりするときは、必ずパーソナルコンピュータおよび本機器の電源をOFFにしてください。OFFにしないと、誤動作を生じたり、内部回路を破損することがあります。

1.3 GP-IB インタフェースの機能

GP-IB インタフェースの機能

- リスナ機能

- ・ 電源のON/OFFと通信の設定を除き、本機器のキー操作で設定できる同じ内容の設定ができます。
- ・ 設定情報や測定/演算データ、波形データなどの、コントローラからの出力指令を受けることができます。
- ・ その他、ステータスレポートに関するコマンドなどを受けることができます。

- トーカ機能

- ・ 設定情報や測定/演算データ、波形データなどを出力することができます。

Note

トークオンリ、リスンオンリおよびコントローラ機能はありません。

リモート/ローカル切り替え時の動作

- ローカル→リモート切り替え時

ローカル状態のときにコントローラからREN(Remote Enable)のメッセージを受け取ると、リモート状態になります。

- ・ REMOTEインジケータが点灯します。
- ・ LOCAL以外はキーが効かなくなります。
- ・ ローカル状態での設定は、リモート状態になっても保持されます。

- リモート→ローカル切り替え時

リモート状態のときにLOCALを押すと、ローカル状態になります。ただし、コントローラによりLocal Lockout(1-6ページ参照)になっているときは無効です。

- ・ REMOTEインジケータが消灯します。
- ・ キー操作が可能になります。
- ・ リモート状態での設定は、ローカル状態になっても保持されます。

1.4 GP-IBインタフェースの仕様

GP-IBインタフェースの仕様

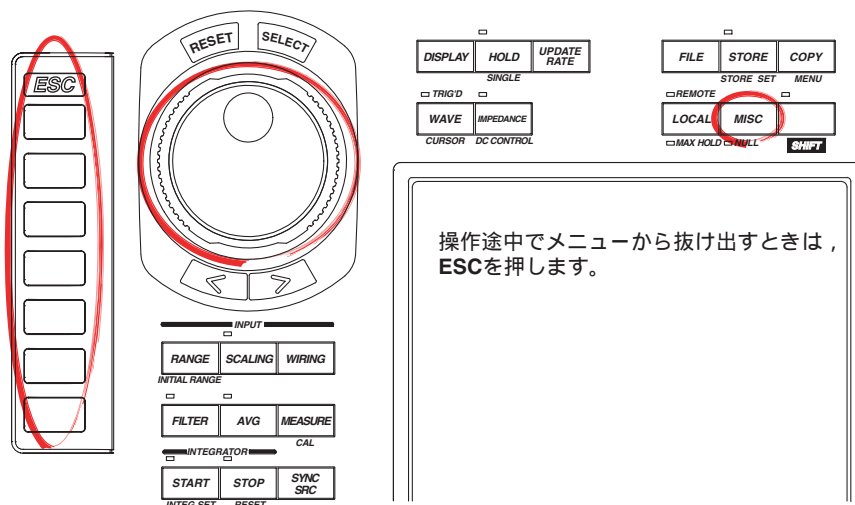
電氣的・機械的仕様	: IEEE St'd 488-1978に準拠
機能的仕様	: 下表
プロトコル	: IEEE St'd 488.2-1987に準拠
使用コード	: ISO(ASCII)コード
モード	: アドレスサブルモード
アドレス設定	: MISCのGP-IBの設定画面で、0～30のアドレスを設定可能。
リモート状態解除	: LOCALを押すことで、リモート状態の解除可能。ただし、コントローラによりLocal Lockoutされているときは無効。

機能的仕様

機能	サブセット名	内 容
ソースハンドシェーク	SH1	送信ハンドシェークの全機能あり
アクセプタ ハンドシェーク	AH1	受信ハンドシェークの全機能あり
トーカ	T6	基本トーカ機能、シリアルボール、MLA(My Listen Address)によるトーカ解除機能あり、トークオンリ機能なし
リスナ	L4	基本リスナ機能、MTA(My Talk Address)によるリスナ解除機能あり、リスンオンリ機能なし
サービスリクエスト	SR1	サービスリクエストの全機能あり
リモートローカル	RL1	リモート/ローカルの全機能あり
パラレルボール	PP0	パラレルボール機能なし
デバイスクリア	DC1	デバイスクリアの全機能あり
デバイストリガ	DT1	デバイストリガ機能あり
コントローラ	C0	コントローラ機能なし
電気特性	E1	オープンコレクタ

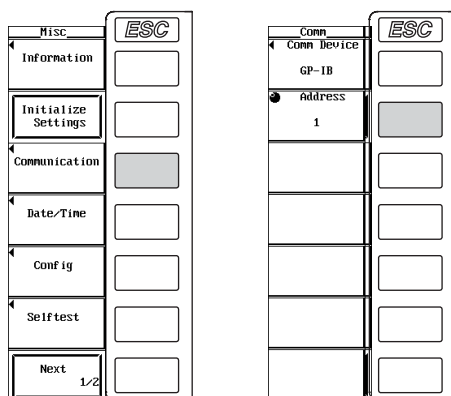
1.5 アドレスの設定

操作キー



操 作

1. MISCを押します。Miscメニューが表示されます。
2. Communicationのソフトキーを押します。
3. Comm Deviceのソフトキーを押して、GP-IBを選択します。
4. ジョグシャトルを回して、アドレスを設定します。



解 説

本機器のキー操作で設定できる内容をコントローラで設定するときや、コントローラに設定情報や波形表示データを出力するときは、下記の設定をします。

アドレスの設定

アドレスابلモードのときの、本機器のアドレスを次の範囲で設定します。

0 ~ 30

GP-IBで接続できる各装置は、GP-IBシステム内で固有のアドレスを持ちます。このアドレスによって他の装置と識別されます。したがって、本機器をパーソナルコンピュータなどに接続するときは、本機器のアドレスを他の機器と重ならないように設定する必要があります。

Note

コントローラが他のデバイスも含めて、GP-IBを使用中はアドレスを変更しないでください。

1.6 インタフェースメッセージに対する応答

インタフェースメッセージに対する応答

● ユニラインメッセージに対する応答

- IFC(Interface Clear)

トーカ，リスナを解除します。データ出力中のときは出力を中止します。

- REN(Remote Enable)

リモート状態/ローカル状態を切り替えます。

IDY(Identify)はサポートしていません。

● マルチラインメッセージ(アドレスコマンド)に対する応答

- GTL(Go To Local)

ローカル状態へ移行します。

- SDC(Selected Device Clear)

- ・ 受信中のプログラムメッセージ(コマンド)と，出力キュー(6-5ページ参照)をクリアします。

- ・ 実行中の*OPC，*OPC?は無効になります。

- ・ *WAI，COMMunicate:WAITは直ちに終了します。

- GET(Group Execute Trigger)

*TRGと同じ動作をします。

PPC(Parallel Poll Configure)，TCT(Take Control)はサポートしていません。

● マルチラインメッセージ(ユニバーサルコマンド)に対する応答

- LLO(Local Lockout)

フロントパネルのLOCALの操作を無効にし，ローカル状態への移行を禁止します。

- DCL(Device Clear)

SDCと同じ動作をします。

- SPE(Serial Poll Enable)

バス上のすべての機器のトーカ機能をシリアルポールモードにします。コントローラは各機器を順番にポーリングします。

- SPD(Serial Poll Disable)

バス上のすべての機器のトーカ機能のシリアルポールモードを解除します。

PPU(Parallel Poll Unconfigure)はサポートしていません。

インタフェースメッセージとは

インタフェースメッセージは，インタフェースコマンドまたはバスコマンドとも呼ばれ，コントローラから発せられるコマンドのことです。次のような分類になっています。

● ユニラインメッセージ

1本の管理ラインを経由してメッセージを送ります。次の3種類があります。

- ・ IFC(Interface Clear)

- ・ REN(Remote Enable)

- ・ IDY(Identify)

● マルチラインメッセージ

8本のデータラインを経由してメッセージを送ります。次のように分類されます。

● アドレスコマンド

機器がリスナあるいはトーカに指定されているときに有効なコマンドです。次の5種類があります。

リスナに指定している機器に有効なコマンド

- ・ GTL(Go To Local)
- ・ SDC(Selected Device Clear)
- ・ PPC(Parallel Poll Configure)
- ・ GET(Group Execute Trigger)

トーカに指定している機器に有効なコマンド

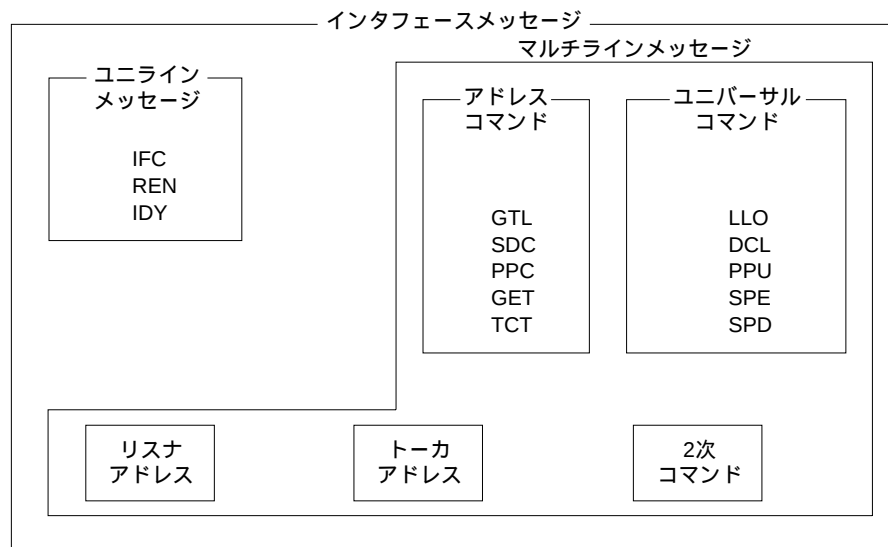
- ・ TCT(Take Control)

● ユニバーサルコマンド

リスナ・トーカの指定の有無に関わらず、すべての機器に有効です。次の5種類があります。

- ・ LLO(Local Lockout)
- ・ DCL(Device Clear)
- ・ PPU(Parallel Poll Unconfigure)
- ・ SPE(Serial Poll Enable)
- ・ SPD(Serial Poll Disable)

その他、インタフェースメッセージとして、リスナアドレス、トーカアドレス、2次コマンドがあります。



印は本機器でサポートしているインタフェースメッセージです。

Note

SDCとDCLの違い

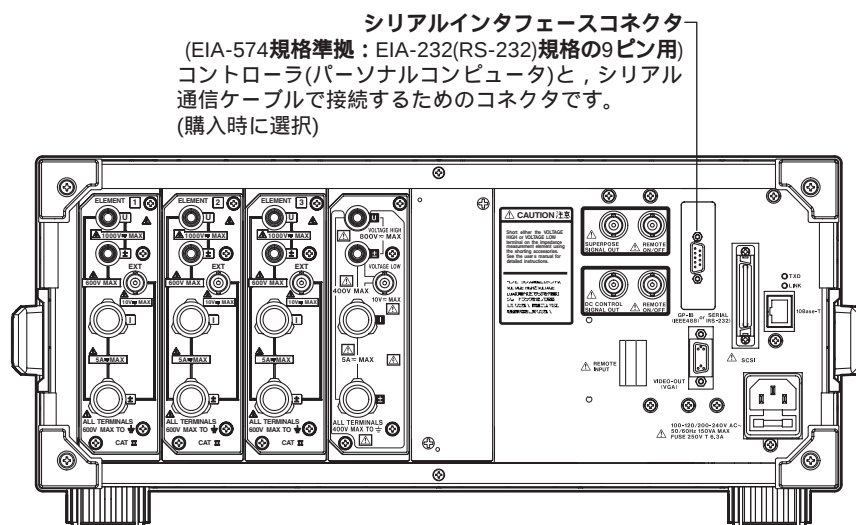
マルチラインメッセージのうち、SDCはトーカ・リスナの指定が必要なアドレスコマンド、DCLはトーカ・リスナの指定が不要なユニバーサルコマンドです。したがって、SDCはある特定の機器を対象にしますが、DCLはバス上のすべての機器を対象にします。

2.1 各部の名称と機能

フロントパネル



リアパネル



2.2 シリアルインタフェースの機能と仕様

受信機能

フロントパネルのキー操作による設定と同じ設定ができます。
測定/演算データ、パネルの設定情報、エラーコードの出力要求を受けられます。

送信機能

測定/演算データを出力できます。
パネルの設定情報、ステータスバイトを出力できます。
発生したエラーコードを出力できます。

シリアルインタフェースの仕様

電気的特性	: EIA-574規格に準拠(EIA-232(RS-232)規格の9ピン用)
接続方式	: ポイント対ポイント
通信方式	: 全2重
同期方式	: 調歩同期式
ボーレート	: 1200, 2400, 4800, 9600, 19200
スタートビット	: 1ビット固定
データ長	: 7または8ビット
パリティ	: 偶数(EVEN), 奇数(ODD), パリティなし
ストップビット	: 1または2ビット
コネクタ	: DELC-J9PAF-13L6(JAEまたは相当品)
ハードウェアハンドシェイク	: CA, CB信号について、常にTRUEか制御線として使用するかのどちらかを選択できます。
ソフトウェアハンドシェイク	: データ送信時、送信データをX-ON, X-OFF信号によって制御するか、送受信ともX-ON, X-OFF信号によって制御するか選択できます。 X-ON(ASCII 11H) X-OFF(ASCII 13H)
受信バッファ長	: 256バイト

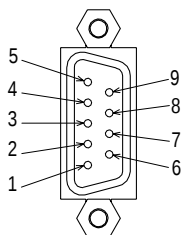
リモート/ローカル切り替え時の動作

- **ローカル→リモート切り替え時**
ローカル状態のときにコンピュータから「:COMMunicate:REMOte ON」コマンドを受信すると、リモート状態になります。
 - ・ REMOTEインジケータが点灯します。
 - ・ LOCAL以外はキーが効かなくなります。
 - ・ ローカル状態での設定は、リモート状態になっても保持されます。
- **リモート→ローカル切り替え時**
リモート状態のときにLOCALを押すとローカル状態になります。ただし、コンピュータから「:COMMunicate:LOCKout ON」コマンドを受信している(ローカルロックアウト状態)ときは無効です。コンピュータから「:COMMunicate:REMOte OFF」コマンドを受信したときは、ローカルロックアウト状態に関係なくローカル状態になります。
 - ・ REMOTEインジケータが消灯します。
 - ・ キー操作が可能になります。
 - ・ リモート状態での設定は、ローカル状態になっても保持されます。

2.3 シリアルインタフェースによる接続

本機器をコンピュータと接続するときは、ハンドシェークの方法、データ転送速度、データフォーマットなどをコンピュータ側と整合するように設定する必要があります。設定の詳細は以下のページをご覧ください。また、インタフェースケーブルは本機器の仕様にあったものをご使用ください。

コネクタと信号名

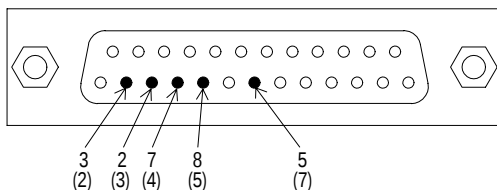


DELIC-J9PAF-13L6 相当品

- | | |
|-----------------------|---|
| 2 RD(Received Data) | : パーソナルコンピュータからの受信データです。
信号方向.....入力 |
| 3 SD(Send Data) | : パーソナルコンピュータへの送信データです。
信号方向.....出力 |
| 5 SG(Signal Ground) | : 信号用接地です。 |
| 7 RS(Request to Send) | : パーソナルコンピュータからデータを受信するときのハンドシェーク方式です。
信号方向.....出力 |
| 8 CS(Clear to Send) | : パーソナルコンピュータへデータを送信するときのハンドシェーク方式です。
信号方向.....入力 |

* 1, 4, 6, 9ピンは使用しません。

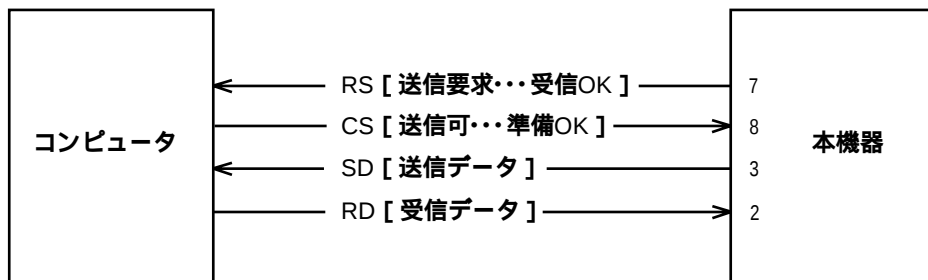
9ピン-25ピン変換コネクタと信号名



カッコ内の数字は、25ピンコネクタのピン番号です。

信号の方向

本機器のシリアルインタフェースで使用する信号の方向を下図に示します。

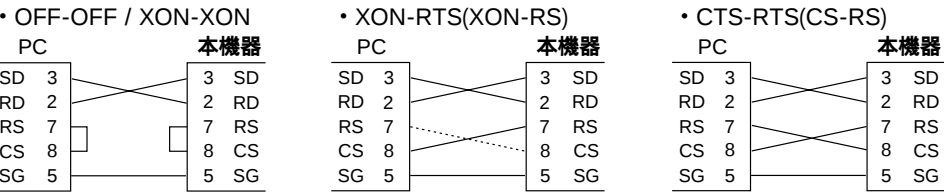


RS-232規定の信号一覧とJISおよびCCITT規定の略号
信 号 表

ピン番号 (9ピンコネクタ)	略 号			名 称
	RS-232	CCITT	JIS	
5	AB (GND)	102	SG	信号用接地
3	BA (TXD)	103	SD	送信データ
2	BB (RXD)	104	RD	受信データ
7	CA (RTS)	105	RS	送信要求
8	CB (CTS)	106	CS	送信可

信号線の結線例

ピン番号は、9ピンコネクタのものです。
一般的には、クロスケーブルを使用してください。



2.4 ハンドシェーク方式の組み合わせ

シリアルインタフェースを使用してコンピュータ通信をするときは、確実にデータの受け渡しができるように、お互いの取り決めによって電気信号上いろいろな手続きをする必要があります。この手続きをハンドシェークといいます。ハンドシェークはコンピュータとの組み合わせでいろいろな方法がありますので、本機器とコンピュータの方式を一致させる必要があります。

本機器では、下表に示すような4通りの方式を選べます。

ハンドシェーク方式の組み合わせ表（・・・・・・機能あり）

ハンドシェーク方式	本機器のメニュー	送信データ制御 (コンピュータへデータを送るときの制御方式)			受信データ制御 (コンピュータからデータを受けるときの制御方式)		
		ソフトハンド シェーク	ハードハンド シェーク	ハンド シェーク なし	ソフトハンド シェーク	ハードハンド シェーク	ハンド シェーク なし
		X-OFF受信で 送信をやめ、 X-ON受信で送 信を再開する	CB(CTS)がFalse で送信をやめ、True で送信を再開する		受信のバッファ のデータが3/4 でX-OFFを送信 し、受信バッフ アのデータが 1/4でX-ONを 送信する	受信のバッファの データが3/4で CA(RTS)をFalse にし、1/4で CA(RTS)をTrue にする	
OFF-OFF	NO-NO			○			○
XON-XON	XON-XON	○			○		
XON-RS	XON-RTS	○				○	
CS-RS	CTS-RTS		○			○	

OFF-OFFの場合

● 送信データ制御

本機器とパーソナルコンピュータの間でハンドシェークは行われません。パーソナルコンピュータからの“X-OFF”、“X-ON”はデータとして扱い、CSは無視します。

● 受信データ制御

本機器とパーソナルコンピュータの間でハンドシェークは行われません。本機器の受信バッファがFULLになると、あふれたデータは捨てます。
RS = True固定。

XON-XONの場合

● 送信データ制御

本機器とパーソナルコンピュータの間でソフトウェアハンドシェークが行われます。本機器がデータ送信中にパーソナルコンピュータからの“X-OFF”コードを受信するとデータの送信を止め、次の“X-ON”コードを受信すると送信を再開します。パーソナルコンピュータからのCSは無視します。

● 受信データ制御

本機器とパーソナルコンピュータの間でソフトウェアハンドシェークが行われます。本機器の受信バッファの空きが64バイトになったらパーソナルコンピュータに“X-OFF”コードを送信し、バッファの空きが192バイトになったら“X-ON”コードを送信します。
RS = True固定。

XON-RSの場合

● 送信データ制御

本機器とパーソナルコンピュータの間でソフトウェアハンドシェークが行われます。本機器がデータ送信中にパーソナルコンピュータからの“X-OFF”コードを受信するとデータの送信を止め、次の“X-ON”コードを受信すると送信を再開します。パーソナルコンピュータからのCSは無視します。

● 受信データ制御

本機器とパーソナルコンピュータの間でハードウェアハンドシェークが行われます。本機器の受信バッファの空きが64バイトになったら“RS=False”とし、バッファの空きが192バイトになったら“RS=True”とします。

CS-RSの場合

● 送信データ制御

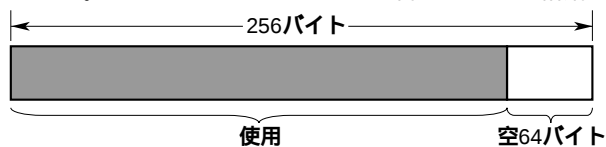
本機器とパーソナルコンピュータの間でハードウェアハンドシェークが行われます。本機器がデータ送信中に“CS=False”となったらデータの送信を止め、次に“CS=True”になったら送信を再開します。パーソナルコンピュータからの“X-OFF”、“X-ON”はデータとして扱います。

● 受信データ制御

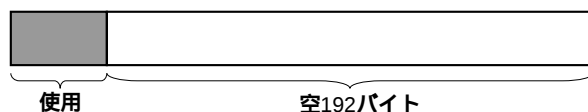
本機器とパーソナルコンピュータの間でハードウェアハンドシェークが行われます。本機器の受信バッファの空きが64バイトになったら“RS=False”とし、バッファの空きが192バイトになったら“RS=True”とします。

データ受信制御に関する注意

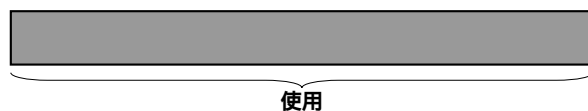
受信データの制御をハンドシェークで行っているときに、受信バッファの空きが64バイト以下になっているのに、コンピュータからデータが来ることがあります。このとき、ハンドシェークの有無に関わらず、受信バッファがFULLになると、あふれたデータは捨てられます。バッファに空きができると再びデータを格納します。



ハンドシェークを行う場合、内部へのデータの受け渡しが間に合わず、バッファの空きが64バイトになったら受信をやめます。



上記の状態の後、データを内部へ渡し続け、バッファの空きが192バイトになったら、受信を再開します。



ハンドシェークに関わらず、もしバッファがFULLになったら、あふれたデータは格納せずに捨てられます。

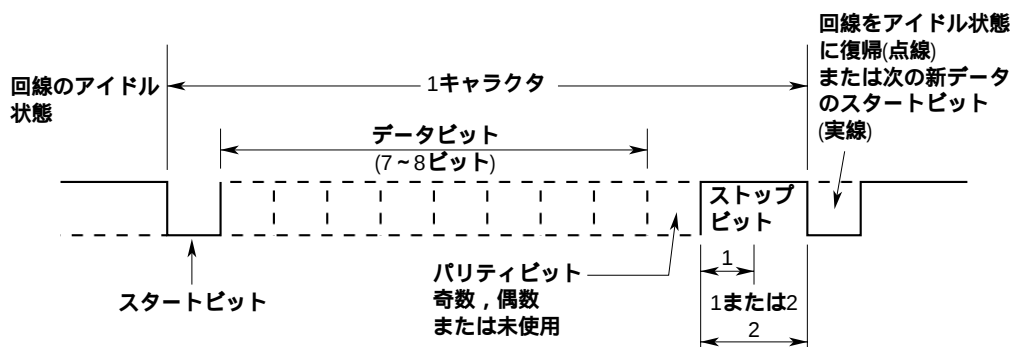
ハンドシェークによるデータ受信制御

Note

本機器とパーソナルコンピュータのそれぞれの受信バッファがFULLにならないように、パーソナルコンピュータのプログラムを作る必要があります。

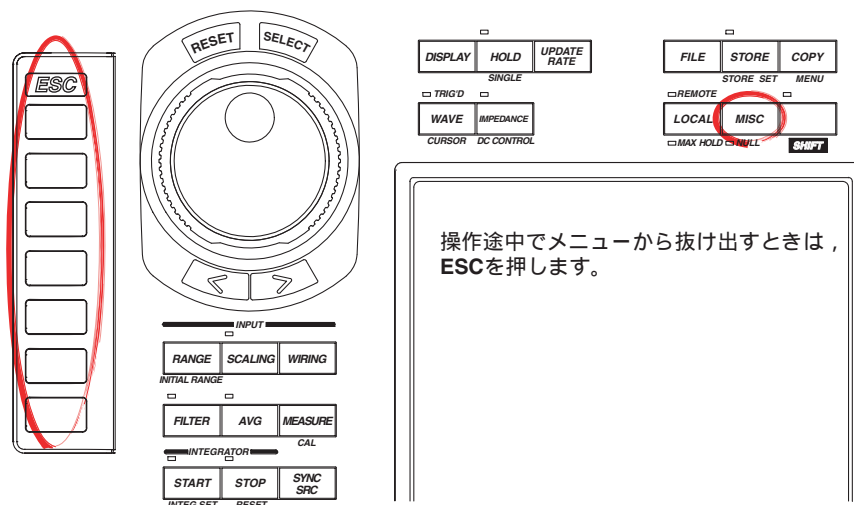
2.5 データフォーマットの組み合わせ

本機器のシリアルインタフェースは、調歩同期式で通信を行います。調歩同期式は、1キャラクタ(1文字)を転送するたびにスタートビットを付け、以降順にデータビット、パリティビット、ストップビットを付加します(下図参照)。



2.6 シリアル通信の設定

操作キー



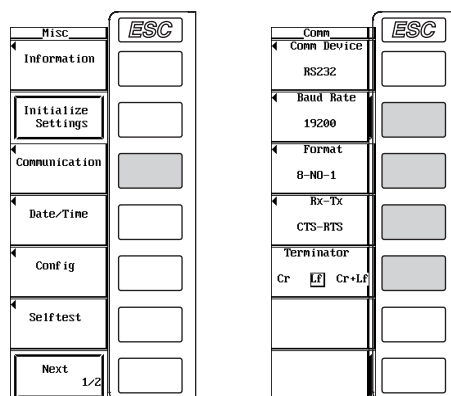
操 作

● シリアル通信(RS-232)メニューの表示

1. MISCを押します。Miscメニューが表示されます。
2. Communicationのソフトキーを押します。
3. Comm Deviceのソフトキーを押して、RS-232を選択します。

● ボーレート、データフォーマットなどの選択

4. Baud Rate(ボーレート), Format(データフォーマット), Rx-Tx(ハンドシェーク方式), Terminator(ターミネータ)のそれぞれのソフトキーを押してから、それぞれの項目を選択します。



解 説

本機器のキー操作で設定できる内容をコントローラで設定するときや、コントローラに設定情報や波形データを出力するときは、下記の設定をします。

- **ボーレートの選択**

次の中から選択します。

1200 , 2400 , 4800 , 9600 , 19200

- **データフォーマットの選択**

データ長-パリティ-ストップビットの組み合わせを、次の中から選択します。

8-NO-1 , 7-EVEN-1 , 7-ODD-1 , 7-NO-2

- **ハンドシェーク方式の選択**

送信データ制御-受信データ制御を、次の中から選択します。

NO-NO , XON-XON , XON-RTS , CTS-RTS

- **ターミネータの選択**

次の中から選択します。本機器のメニューでは、本機器からデータを送信するときのターミネータを選択します。本機器でデータを受信するときのターミネータは「Lf」 , 「Cr+Lf」のどちらかを使用してください。

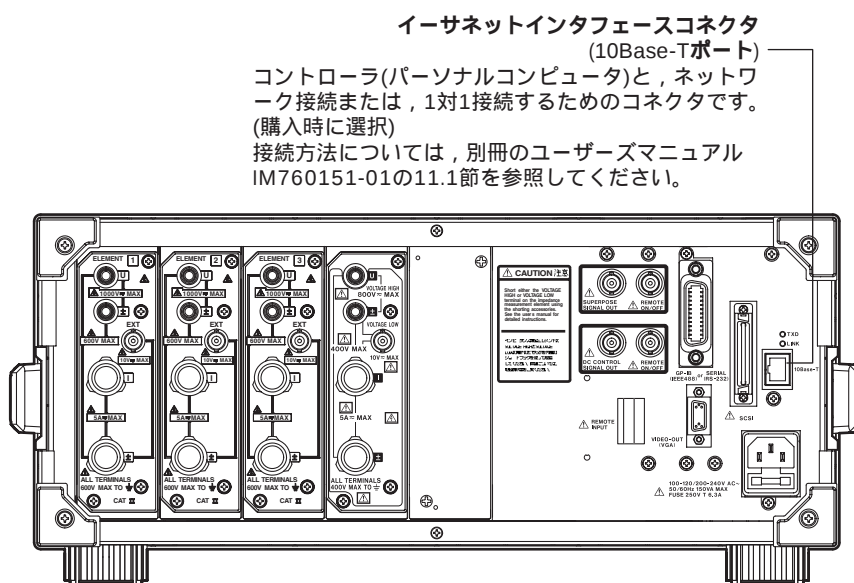
Cr , Lf , Cr+Lf

3.1 各部の名称と機能

フロントパネル



リアパネル



3.2 イーサネットインタフェースの機能と仕様

受信機能

本機器では、イーサネット通信を使って、パーソナルコンピュータで本機器をコントロールできます。以下にその機能と設定操作について説明します。

フロントパネルのキー操作による設定と同じ設定ができます。
測定/演算データ、パネルの設定情報、エラーコードの出力要求を受けられます。

送信機能

測定/演算データを出力できます。
パネルの設定情報、ステータスバイトを出力できます。
発生したエラーコードを出力できます。

イーサネットインタフェースの仕様

電氣的・機械的仕様 : IEEE 802.3準拠

同時接続数 : 1

ポート番号 : 10001/tcp

その他の仕様については、WT1600FCインピーダンスメータユーザズマニュアルIM760151-01(以降、ユーザズマニュアルIM760151-01と略します)の15.13節をご覧ください。

リモート/ローカル切り替え時の動作

● ローカル → リモート切り替え時

ローカル状態のときにコンピュータから「:COMMunicate:REMOte ON」コマンドを受信すると、リモート状態になります。

- ・ REMOTEインジケータが点灯します。
- ・ LOCAL以外はキーが効かなくなります。
- ・ ローカル状態での設定は、リモート状態になっても保持されます。

● リモート → ローカル切り替え時

リモート状態のときにLOCALを押すとローカル状態になります。ただし、コンピュータから「:COMMunicate:LOCKout ON」コマンドを受信している(ローカルロックアウト状態)ときは無効です。コンピュータから「:COMMunicate:REMOte OFF」コマンドを受信したときは、ローカルロックアウト状態に関係なくローカル状態になります。

- ・ REMOTEインジケータが消灯します。
- ・ キー操作が可能になります。
- ・ リモート状態での設定は、ローカル状態になっても保持されます。

Note

イーサネットインタフェースは、他の通信インタフェース(GP-IB、シリアル(RS-232)インタフェース)と同時に使用できません。

ユーザー認証機能

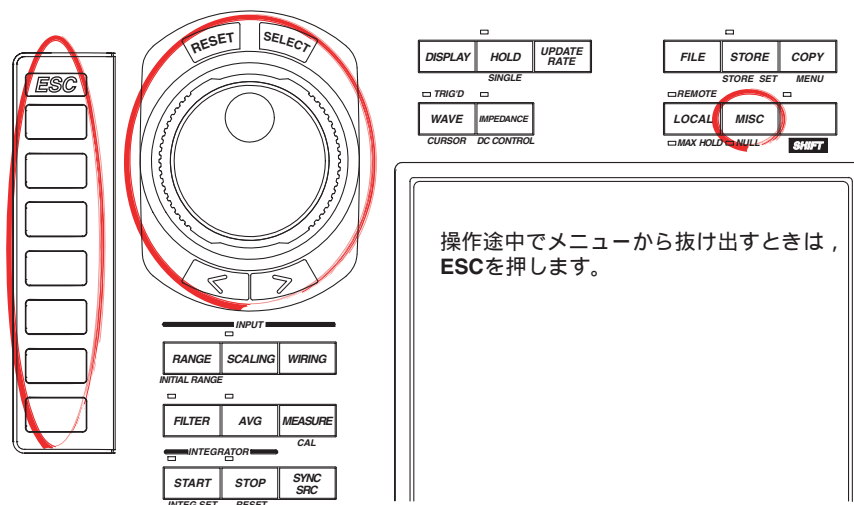
イーサネットインタフェースを使って、パーソナルコンピュータから本機器にアクセスするには、ユーザー名とパスワードを入力する必要があります。本機器のMISCメニューの中のUser Account設定画面で、本機器にアクセスするときのユーザー名とパスワードを設定してください。詳細は、次節の「イーサネットコントロールの設定」をご覧ください。

本機器とパーソナルコンピュータの接続

本機器をパーソナルコンピュータに接続する方法については、ユーザーズマニュアルIM760151-01の11.1節をご覧ください。

3.3 イーサネットコントロールの設定

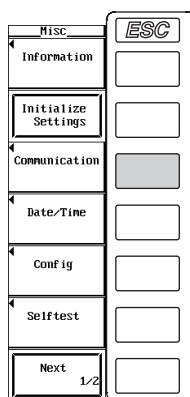
操作キー



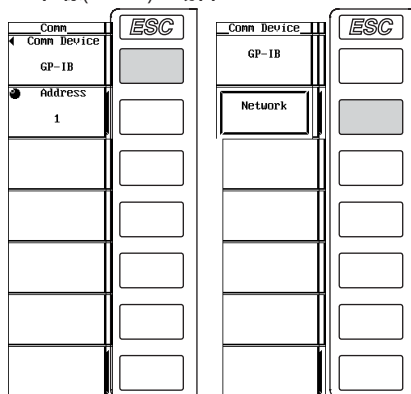
操 作

● 本機器をコントロールする通信インターフェースを選択する

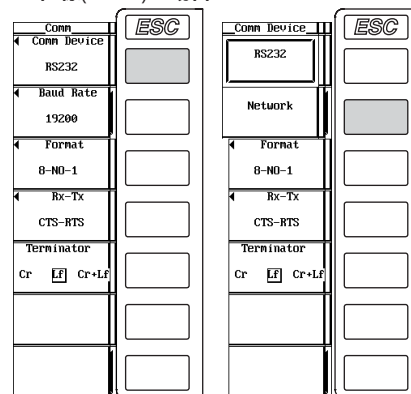
1. MISCを押します。Miscメニューが表示されます。
2. Communicationのソフトキーを押します。Commメニューが表示されます。
3. Comm Deviceのソフトキーを押します。Comm Deviceメニューが表示されます。
4. Networkのソフトキーを押します。この操作で、本機器をコントロールする通信インターフェースの種類として、イーサネットインターフェースを選択しました。



-C1仕様(GP-IB)の場合



-C2仕様(Serial)の場合



Note

Comm Deviceメニューで選択した通信インターフェースだけを使用できます。選択していない他の通信インターフェースにコマンドを送信しても、本機器は受け付けません。

● ユーザー名とパスワードを設定する

5. User Accountのソフトキーを押します。User Accountダイアログボックスが表示されます。

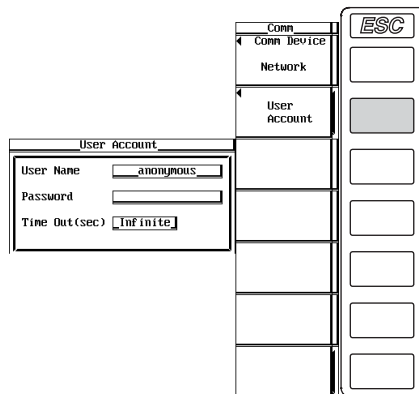
Note

FTPサーバ機能(ユーザーズマニュアルIM760151-01の11.6節参照)を使用するときのUser AccountとPasswordは別の設定です。FTPサーバの設定と同じにすることを推奨します。

6. ジョグシャトルを回して、User Nameを選択します。
7. SELECTを押します。キーボードが表示されます。
8. キーボードを操作して、ユーザー名を入力します。
キーボードの操作については、ユーザーズマニュアルIM760151-01の3.8節をご覧ください。
9. ジョグシャトルを回して、Passwordを選択します(設定は2回繰り返します)。
10. SELECTを押します。キーボードが表示されます。
11. キーボードを操作して、パスワードを入力します。ユーザー名が「anonymous」の場合は、パスワードの設定は不要です。
キーボードの操作については、ユーザーズマニュアルIM760151-01の3.8節をご覧ください。

● タイムアウト時間を設定する

12. ジョグシャトルを回して、Time Outを選択します。
13. SELECTを押します。タイムアウト時間設定ボックスが表示されます。
14. ジョグシャトルを回して、タイムアウト時間を設定します。
ジョグシャトルによる入力方法については、ユーザーズマニュアルIM760151-01の3.8節をご覧ください。
15. SELECTまたはESCを押して、設定ボックスを閉じます。



● TCP/IPの設定をする

イーサネットインタフェースを使って、パーソナルコンピュータで本機器をコントロールするには、TCP/IPの設定をする必要があります。設定方法については、ユーザーズマニュアルIM760151-01の11.2節をご覧ください。

解 説

イーサネットインタフェースを使って、パーソナルコンピュータで本機器をコントロールできます。コントロールするためには、上記の設定操作に加えて、パーソナルコンピュータに当社で用意した専用ソフトウェアがインストールされている必要があります。

無償ソフトウェア

FcEvaluation バージョン1.01以降。

次のURLからダウンロードできます。

<http://www.yokogawa.co.jp/Measurement/F-SOFT/>

● ユーザー名の設定

- ・ 本機器にアクセスを許可するユーザー名を設定します。
- ・ 15文字以内の英数字を入力できます。
- ・ 使用できる文字は、0～9、A～Z、%、_、()(カッコ)、-(マイナス)です。
- ・ anonymousを設定すると、パスワードなしで、外部(PC)から本機器にアクセスできます。

● パスワードの設定

- ・ 本機器にアクセスを許可するユーザー名のパスワードを設定します。
- ・ 15文字以内の英数字を入力できます。
- ・ 使用できる文字は、0～9、A～Z、%、_、()(カッコ)、-(マイナス)です。
- ・ ユーザー名をanonymousにすると、パスワードなしで、外部(PC)から本機器にアクセスできます。
- ・ パスワードの設定は2回繰り返します。

● タイムアウト時間の設定

ある一定時間(タイムアウト時間)過ぎても本機器へのアクセスがない場合、本機器がネットワークとの接続を閉じます。

0～3600s、Infinite(制限時間なし)の範囲で設定できます。初期値はInfiniteです。

Note

設定した内容を反映するには、本機器の電源を入れなおす必要があります。

4.1 メッセージ

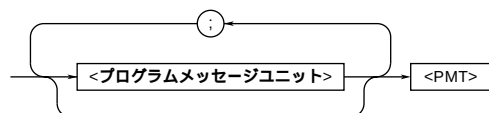
メッセージ

コントローラと本機器の間の送受信は、メッセージという単位で行います。コントローラから本機器に送信するメッセージをプログラムメッセージといい、コントローラが本機器から受信するメッセージを応答メッセージといいます。

プログラムメッセージの中に応答を要求するメッセージユニット(クエリといいます)があるときは、本機器はプログラムメッセージを受信したあとに、応答メッセージを送信します。1つのプログラムメッセージに対する応答は、必ず1つの応答メッセージになります。

プログラムメッセージ

プログラムメッセージの書式は次のようになります。



<プログラムメッセージユニット>

プログラムメッセージは、1つ以上のプログラムメッセージユニットをつないだものです。プログラムメッセージユニットが1つの命令に相当します。本機器は受信した順序で命令を実行していきます。

プログラムメッセージユニットは「;」(セミコロン)で区切ります。

プログラムメッセージの書式については、次項を参照してください。

例

```
:IMPedance:STATe ON;MEASure:TYPE STABle<PMT>
```

ユニット
ユニット

<PMT>

プログラムメッセージのターミネータです。次の3種類があります。

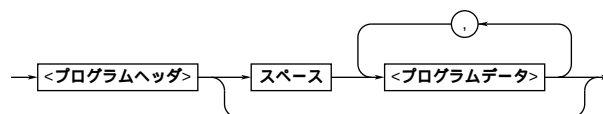
NL(ニューライン) : LF(ラインフィード)と同じ、ASCIIコード「0AH」の1文字

^END : IEEE488.1で定義されているENDメッセージ(EOI信号)
(ENDメッセージと同時に送信されたデータバイトは、プログラムメッセージの最後のデータになります)

NL^END : ENDメッセージが付加されたNL
(NLはプログラムメッセージには含まれません)

● プログラムメッセージユニットの書式

プログラムメッセージユニットの書式は次のようになります。



<プログラムヘッダ>

プログラムヘッダは命令の種類を表わします。詳しくは、4-3ページを参照してください。

<プログラムデータ>

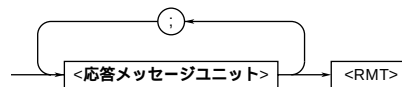
命令を実行するときに必要な条件などがあるときは、プログラムデータを付けます。プログラムデータを付けるときは、ヘッダとデータをスペース(ASCIIコード「20H」)で区切ります。複数のデータがあるときは、データとデータの間に「,」(カンマ)で区切ります。詳しくは、4-5ページを参照してください。

例 :IMPedance:STATe ON<PMT>

ヘッダ
データ

応答メッセージ

応答メッセージの書式は次のようになります。



<応答メッセージユニット>

応答メッセージは、1つ以上の応答メッセージユニットをつないだものです。応答メッセージユニットが1つの応答に相当します。

応答メッセージユニットは「;」(セミコロン)で区切られます。

応答メッセージの書式については、次ページを参照してください。

例

```
:IMPEDANCE:STATE 1;MEASURE:TYPE STABLE<RMT>
```

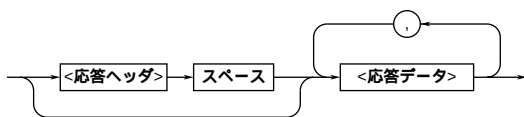
ユニット
ユニット

<RMT>

応答メッセージのターミネータで、NL^ENDです。

● 応答メッセージユニットの書式

応答メッセージユニットの書式は次のようになります。



<応答ヘッダ>

応答データの前に応答ヘッダが付くことがあります。
ヘッダとデータの間は、1文字のスペースで区切られます。
詳しくは、4-4ページを参照してください。

< 応答データ >

応答データは、応答の内容を示します。複数のデータがあるときは、データとデータの間は「,」（カンマ）で区切られます。詳しくは、4-5ページを参照してください。

例 100.00E-03<RMT> :DISPLAY:FORMAT WAVE<RMT>

データ ヘッダ データ

プログラムメッセージに複数のクエリがある場合、応答の順序はクエリの順序に従います。クエリの多くは1つの応答メッセージユニットを返しますが、複数のユニットを返すものもあります。1番目のクエリの応答は1番目のユニットですが、n番目の応答はn番目のユニットとは限りません。確実に応答を取り出したいときは、プログラムメッセージを分けるようにしてください。

メッセージの送受信時の注意

- ・クエリを含まないプログラムメッセージを送信したときは、いつでも次のプログラムメッセージを送信できます。
- ・クエリを含むプログラムメッセージを送信したときは、次のプログラムメッセージを送信する前に応答メッセージを受信しなければなりません。もし、応答メッセージを受信しないか、途中までしか受信せずに次のプログラムメッセージを送信したときは、エラーになります。受信されなかった応答メッセージは捨てられます。
- ・コントローラが応答メッセージがないのに受信しようとしたときは、エラーになります。もし、コントローラがプログラムメッセージを送信し終わる前に応答メッセージを受信しようすると、エラーになります。

- ・メッセージにユニットが複数あるプログラムメッセージを送信したときに、その中に不完全なプログラムユニットが存在すると、本機器は完全と思われるプログラムメッセージユニットを拾い上げて実行を試みますが、必ずしも成功するとは限りません。また、その中にクエリが含まれていても、必ずしも応答が返るとは限りません。

デッドロック状態

本機器は、送受信とも最低1024バイトのメッセージをバッファに蓄えておくことができます(バイト数は、動作状態によって増減することがあります)。このバッファが送受信と同時にいっぱいになると、本機器はそのままでは通信動作を続行できなくなります。これをデッドロック状態といいます。このときは、応答メッセージを捨てることで回復します。

プログラムメッセージを<PMT>も含めて1024バイト以下にしておけば、デッドロックすることはありません。また、クエリがないプログラムメッセージは、デッドロックすることはありません。

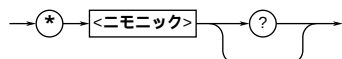
4.2 命令

命令

コントローラから本機器に送信される命令(プログラムヘッダ)には、以下に示す3種類があります。それぞれプログラムヘッダの書式が異なります。

共通コマンドヘッダ

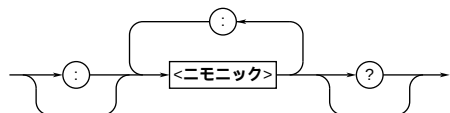
IEEE 488.2-1987で規定されている命令を共通コマンドといいます。共通コマンドのヘッダの書式は次のようになります。先頭に必ず「*」(アスタリスク)を付けます。



共通コマンドの例 *CLS

複合ヘッダ

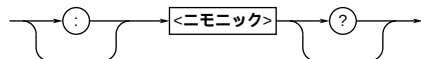
共通コマンド以外の本機器固有の命令は、機能ごとに分類されて、階層化されています。複合ヘッダの書式は次のようになります。下の階層を記述するときは、必ず「:」(コロン)を付けます。



複合ヘッダの例 :DISPlay:FORMat

単純ヘッダ

機能的に独立した、階層を持たない命令です。ヘッダの書式は次のようになります。



単純ヘッダの例 :HOLD

Note

<ニモニック>とは、アルファベットと数字からなる文字列です。

命令を続けて記述する場合

● グループについて

ヘッダが階層化された共通の複合ヘッダを持つコマンド群をグループといいます。グループの中にさらに小さいグループが存在することもあります。

例 インピーダンス測定の表示に関するグループ

```
:DISPlay:IMPedance?
:DISPlay:IMPedance:TYPE
:DISPlay:IMPedance:OBJect
:DISPlay:IMPedance:ICURsor
```

● 同じグループの命令を続けて記述する場合

本機器は、実行している命令がどの階層の命令であるかを記憶し、次に送信した命令も同じ階層に属しているものと仮定して解析を行っています。したがって、同じグループの命令は、共通のヘッダの部分を省略することができます。

```
例 :DISPlay:IMPedance:TYPE ZR_ZI;
    OBJect 5<PMT>
```

● 違うグループの命令を続けて記述する場合

グループが違う命令を後ろに記述するときは、ヘッダの先頭に「:」(コロン)を付けます。

```
例 :DISPlay:IMPedance:TYPE ZR_ZI;;
    DISPlay:FORMat NUMeric<PMT>
```

● 単純ヘッダを続けて記述する場合

他の命令に単純ヘッダを続けるときは、単純ヘッダの先頭に「:」(コロン)を付けます。

```
例 :DISPlay:IMPedance:TYPE ZR_ZI;;
    HOLD ON<PMT>
```

● 共通コマンドを続けて記述する場合

IEEE 488.2-1987で定義された共通コマンドは、階層には無関係です。「:」(コロン)はつける必要はありません。

```
例 :DISPlay:IMPedance:TYPE ZR_ZI;*CLS;
    OBJect 5<PMT>
```

● コマンド間を<PMT>で区切った場合

ターミネータで区切ると、2つのプログラムメッセージを送信することになります。したがって、同じグループでのコマンドを続ける場合でも、共通のヘッダを省略することはできません。

```
例 :DISPlay:IMPedance:TYPE ZR_ZI<PMT>;
    DISPlay:IMPedance:OBJect 5<PMT>
```

上位クエリ

初めて出てくるグループの最上位のコマンドに「?」を付けたクエリを上位クエリといいます。この上位クエリを実行すると、そのグループで設定できるすべての設定をまとめて受信することができます。階層が3階層以上あるグループで、下の階層をすべて出力するものもあります。

例 :DISPlay[:NUMeric]:IMPedance?<PMT> ->
:DISPLAY:NUMERIC:IMPEDANCE:TYPE ZR_ZI;
OBJECT 5;ICURSOR 1<RMT>

上位クエリの応答は、そのまま本機器にプログラムメッセージとして送信することができます。送信すると、上位クエリを行ったときの設定を再現できます。ただし、上位クエリでは現在使われていない設定情報を返さないものもあります。必ずしもそのグループのすべての情報が応答として出力されるわけではないので、注意してください。

ヘッダの解釈の規則

本機器は、受信したヘッダを次の規則に従って解釈します。

- ・二モニックのアルファベットの太文字/小文字は区別しません。
例 「CURSor」 -> 「cursor」「CURsor」でも可
- ・小文字の部分は省略できます。
例 「CURSor」 -> 「CURSO」「CURS」でも可
- ・ヘッダの最後の「?」(クエスチョンマーク)は、クエリであることを示します。「?」は省略できません。
例 「CURSor?» -> 最小の省略形は「CURS?」
- ・二モニックの最後に付いている<x>(数値)を省略すると、x=1と解釈します。
例 「ELEMent<x>」 -> 「ELEM」とすると
「ELEMent1」の意味
- ・[]で囲まれた部分は省略できます。
例 [:INPut]:SCALing[:STATe] ON ->
「SCALing ON」でも可
ただし上位クエリの場合、最後の部分は省略できません。
例 「SCALing?」と「SCALing:STATe?」は別のクエリになる。

4.3 応答

コントローラが「?」の付いた命令であるクエリを送信すると、本機器はそのクエリに対する応答メッセージを返します。返される形式は、次の2つに分けられます。

・ヘッダ+データの応答

応答をそのままプログラムメッセージとして利用できるものは、命令のヘッダを付けて返されます。

例 :DISPlay:FORMat?<PMT> -> :DISPLAY:
FORMAT WAVE<RMT>

・データだけの応答

そのままプログラムメッセージとして利用できないもの(クエリ専用の命令)は、ヘッダを付けずにデータだけで返されます。ただし、ヘッダを付けて返すクエリ専用の命令もあります。

例 [:INPut]:POVer?<PMT> -> 0<RMT>

・ヘッダを付けない応答を返したい場合

「ヘッダ+データ」で返されるものでも、ヘッダを強制的に付けないようにすることができます。これには、「COMMunicate:HEADer」命令を使用します。

・省略形について

応答のヘッダは、通常は小文字の部分を省略した形で返されます。これを省略しないフルスペルにすることもできます。これには、「COMMunicate:VERBose」命令を使用します。また、省略形のときは[]で囲まれた部分も省略されます。

4.4 データ

データ

データとは、ヘッダの後ろにスペースを空けて記述する条件や数値です。データは次のように分類されます。

データ	意味
<10進数>	10進数で表された数値 (例: PT比の設定 -> [:INPUT]:SCALing:PT:ELEMeNt1 100)
<電圧><電流> <時間><周波数>	物理的な次元を持った数値 (例: 電圧レンジの設定 -> [:INPUT]:VOLTAge:RANGE: ELEMeNt1 100V)
<Register>	2, 8, 10, 16進数のどれかで表されたレジスタ値 (例: 拡張イベントレジスタ値 -> STATUS:ESE #HFE)
<文字データ>	規定された文字列(二モニック)。{}内から選択 (例: トリガモードの選択 -> WSEtup:TRIGger:MODE {AUTO NORMAl})
<Boolean>	ON/OFFを表す。「ON」「OFF」または数値で設定 (例: データホールドをON ->:HOLD ON)
<文字列データ>	任意の文字列 (例: ユーザ定義ファンクション -> MEASure:FUNCTION1: EXPRession "URMS(E1)")
<Filename>	ファイル名を表す (例: 保存ファイル名 -> FILE:SAVE:WAVE[:EXECute] "CASE1")
<ブロックデータ>	任意の8ビットの値を持つデータ (例: 取り込んだ波形データの応答 -> #40012ABCDEFGHJKL)

<10進数>

<10進数>は下表のように10進数で表現された数値です。なお、これはANSI X3.42-1975で規定されているNR形式で記述します。

記号	意味	例
<NR1>	整数	125 -1 +1000
<NR2>	固定小数点数	125.0 -.90 +001.
<NR3>	浮動小数点数	125.0E+0 -9E-1 +.1E4
<NRf>	<NR1> ~ <NR3>のどれでも可能	

- 本機器がコントローラから送られた10進数を受け取るときは、<NR1> ~ <NR3>のどの形式でも受け付けます。これを<NRf>で表します。
- 本機器からコントローラに返される応答メッセージは、<NR1> ~ <NR3>のどれを使用するかはクエリごとに決められています。値の大きさによって使用する形式が変わることはありません。
- <NR3>形式の場合、「E」のあとの「+」は省略できます。「-」は省略できません。
- 設定範囲外の値を記述したときは、設定できる値でいちばん近い値になります。
- 精度以上の値を記述したときは、四捨五入します。

<電圧> , <電流> , <時間> , <周波数>

<電圧> , <電流> , <時間> , <周波数>は、<10進数>のうち物理的な次元を持ったデータです。前述の<NRf>形式に<乗数>および<単位>を付けることができます。次の書式のどれかで記述します。

書式	例
<NRf><乗数><単位>	5MV
<NRf><単位>	5E-3V
<NRf><乗数>	5M
<NRf>	5E-3

・<乗数>

使用できる<乗数>は下表のとおりです。

記号	読み	乗数
EX	エクサ	10^{18}
PE	ペタ	10^{15}
T	テラ	10^{12}
G	ギガ	10^9
MA	メガ	10^6
K	キロ	10^3
M	ミリ	10^{-3}
U	マイクロ	10^{-6}
N	ナノ	10^{-9}
P	ピコ	10^{-12}
F	フェムト	10^{-15}

・<単位>

使用できる<単位>は下表のとおりです。

記号	読み	意味
V	ボルト	電圧
A	アンペア	電流
S	セカンド	時間
HZ	ヘルツ	周波数
MHZ	メガヘルツ	周波数

- <乗数>と<単位>は、大文字/小文字の区別がありません。
- マイクロの「m」は「U」で表します。
- メガの「M」はミリと区別するため、「MA」で表します。ただし、電流のときは「MA」はミリアンペアと解釈します。また、メガヘルツの場合は、「MHZ」で表します。したがって、周波数のときは乗数に「M(ミリ)」は使用できません。
- <乗数>も<単位>も省略したときは、基本単位(V, A, S, HZ)になります。
- 応答メッセージは必ず<NR3>形式になります。また、<乗数>および<単位>をつけずに基本単位で返します。

4.4 データ

<Register>

<Register>は整数ですが、<10進数>のほかに<16進数><8進数><2進数>でも表現できるデータです。数値がビットごとに意味を持つときに使用します。次の書式のどれかで記述します。

書式	例
<NRf>	1
#H<0～9, A～Fからなる16進数>	#H0F
#Q<0～7からなる8進数>	#Q777
#B<0または1からなる2進数>	#B001100

- ・<Register>は、大文字/小文字の区別はありません。
- ・応答メッセージは必ず<NR1>で返されます。

<文字データ>

<文字データ>は、規定された文字(二モニック)のデータです。主に選択肢を表現するときに使用され、{}内の文字列からどれか1つを選んで記述します。データの解釈のしかたは、4-4ページの「ヘッダ解釈の規則」と同様です。

書式	例
{<AUTO NORMAL>}	AUTO

- ・応答メッセージでは、ヘッダと同様に「COMMunicate:VERBose」を使って、フルスペルで返すか、省略形で返すかを選ぶことができます。
- ・「COMMunicate:HEADer」の設定は<文字データ>には影響しません。

<Boolean>

<Boolean>は、ONまたはOFFを示すデータです。次の書式のどれかで記述します。

書式	例
{<ON OFF <NRf>}&	ON OFF 1 0

- ・<NRf>で表す場合は、整数に丸めた値が「0」のときがOFF、「0以外」のときがONになります。
- ・応答メッセージは必ず、ONのときは「1」、OFFのときは「0」で返されます。

<文字列データ>

<文字列データ>は、<文字データ>のように規定された文字列ではなく、任意の綴りの文字列です。次のように、「'」（シングルクォーテーション）または「"」（ダブルクォーテーション）で囲った書式で記述します。

書式	例
<文字列データ>	'ABC' "IEEE488.2-1987"

- ・「"」内に文字列として「'」があるときは、「"」で表します。「'」のときも同様です。
- ・応答メッセージは、必ず「"」（ダブルクォーテーション）で囲って返されます。
- ・<文字列データ>は任意の綴りなので、最後の「'」（シングルクォーテーション）または「"」（ダブルクォーテーション）がないと、本機器は残りのプログラムメッセージユニットを<文字列データ>の一部と解釈してしまい、エラーが正しく検出できない場合があります。

<Filename>

<Filename>は、ファイル名を表すデータです。次の書式のどれかで記述します。

書式	例
{<NRf> <文字データ> <文字列データ>}	1 CASE "CASE"

- ・<NRf>は整数に丸めた8桁の値をASCIIコードに直したものがファイル名になります(例: 1 -> "00000001")。ただし、負の値は使えません。
- ・<文字データ>、<文字列データ>は先頭の8文字がファイル名になります。
- ・応答メッセージは、必ず<文字列データ>で返されます。

<ブロックデータ>

<ブロックデータ>は、任意の8ビットの値を持つデータです。本機器では、応答メッセージだけに使用されます。書式は次のとおりです。

書式	例
#N<N桁の10進数><データバイトの並び>	#40012ABCDEFGHijkl

- ・#N
<ブロックデータ>であることを表します。「N」は次に続くデータバイト数を表わすASCIIコードの文字数(桁)を示します。
- ・<N桁の10進数>
データのバイト数を表します(例: 0012 = 12 バイト)。
- ・<データバイトの並び>
実際のデータを表します(例: ABCDEFGHIJKL)。
- ・データは8ビットでとり得る値(0～255)です。したがって、「NL」を示すASCIIコード「0AH」もデータになることがありますので、コントローラ側では注意が必要です。

4.5 コントローラとの同期

オーバーラップコマンドとシーケンシャルコマンド

コマンドには、オーバーラップコマンドとシーケンシャルコマンドの2種類があります。オーバーラップコマンドの場合は、先に送信したコマンドによる動作が完了する前に、次のコマンドによる動作が開始される場合があります。

たとえば、電圧レンジを指定してその結果を問い合わせるときに、次のプログラムメッセージを送信すると、応答は常に最新の設定値(この場合は100V)を返します。

```
:INPut:VOLTage:RANge:ELEMent1 100V;
ELEMent?<PMT>
```

これは、「INPut:VOLTage:RANge:ELEMent1」が自身の処理を終えるまで、次の命令を待たせているためです。このような命令をシーケンシャルコマンドといいます。

これに対して、たとえばファイルロードを実行して、その結果の電圧レンジを問い合わせたいときに、次のプログラムメッセージを送信すると、

```
:FILE:LOAD:SETup "FILE1";:INPut:VOLTage:
RANge:ELEMent1?
```

「:INPut:VOLTage:RANge:ELEMent1?」はファイルロードが終了する前に実行されてしまい、応答される電圧レンジはファイルロードする前の値になってしまいます。

「FILE:LOAD:SETup」のように、自身の処理が終わる前に次の命令を実行させることをオーバーラップ動作といい、オーバーラップ動作する命令を、オーバーラップコマンドといいます。

このようなときは、以下に示す方法でオーバーラップ動作を回避できます。

オーバーラップコマンドとの同期をとる方法

• *WAIコマンドを使う

*WAIコマンドは、オーバーラップコマンドが終了するまで、*WAIに続く命令を待つコマンドです。

```
例:COMMunicate:OPSE #H0040;:FILE:LOAD:
SETup "FILE1";*WAI;:INPut:VOLTage:
RANge:ELEMent1?<PMT>
```

「COMMunicate:OPSE」は「*WAI」の対象を選ぶ命令です。ここではメディアアクセスだけを対象に指定しています。

「:INPut:VOLTage:RANge:ELEMent?」の直前で

「*WAI」を実行しているため、「:INPut:

VOLTage:RANge:ELEMent1?」は、ファイルロードが終了するまで実行されません。

• COMMunicate:OVERlapコマンドを使う

COMMunicate:OVERlapコマンドは、オーバーラップ動作を許可(または禁止)する命令です。

```
例:COMMunicate:OVERlap #HFFBF;:FILE:
LOAD:SETup "FILE1";:INPut:VOLTage:
RANge:ELEMent1?<PMT>
```

「COMMunicate:OVERlap #HFFBF」は、メディアアクセス以外のオーバーラップ動作を許可していません。ファイルロードはオーバーラップ動作を許可されていないので、次の「FILE:LOAD:SETup」は、シーケンシャルコマンドと同じ動作をします。したがって、

「:INPut:VOLTage:RANge:ELEMent1?」は、ファイルロードが終了するまで実行されません。

• *OPCコマンドを使う

*OPCコマンドは、オーバーラップ動作が終了したときに、標準イベントレジスタ(6-3ページ参照)のビット0であるOPCビットを1にする命令です。

```
例:COMMunicate:OPSE #H0040;*ESE 1;
*ESR?;*SRE 32;:FILE:LOAD:
SETup "FILE1";*OPC<PMT>
(*ESR?の応答を読む)
(サービスリクエストの発生を待つ)
```

```
:INPut:VOLTage:RANge:ELEMent1?<PMT>
```

「COMMunicate:OPSE」は「*OPC」の対象を選ぶ命令です。ここではメディアアクセスだけを対象に指定しています。

「*ESE 1」と「*SRE 32」は、OPCビットが1になったときだけ、サービスリクエストを発生することを示しています。

「*ESR?」は、標準イベントレジスタをクリアします。上の例では、

「:INPut:VOLTage:RANge:ELEMent1?」は、サービスリクエストが発生するまで実行されません。

• *OPC?クエリを使う

*OPC?クエリは、オーバーラップ動作が終了したときに応答を生成する命令です。

```
例:COMMunicate:OPSE #H0040;:FILE:LOAD:
SETup "FILE1";*OPC?<PMT>
(*OPC?の応答を読む)
:INPut:VOLTage:RANge:ELEMent?<PMT>
```

「COMMunicate:OPSE」は「*OPC?」の対象を選ぶ命令です。ここではメディアアクセスだけを対象に指定しています。

「*OPC?」はオーバーラップ動作が終了するまで応答を作成しないので、「*OPC?」の応答を読み終えたときには、ファイルロードは終了しています。

Note

命令のほとんどはシーケンシャルコマンドです。オーバーラップコマンドは、5章でオーバーラップコマンドと明記しています。それ以外はすべてシーケンシャルコマンドです。

オーバーラップコマンド以外の同期をとる方法

シーケンシャルコマンドの場合でも、測定データを正しく問い合わせるために同期をとる必要がある場合もあります。

たとえば、測定データの更新ごとに最新の数値データを問い合わせたいとき、任意のタイミングで

「:NUMeric[:NORMal]:VALue?」を送信していると、測定データの更新終了いかんにかかわらず現在の測定データを応答するため、前回と同じデータを受信してしまう可能性があります。

このときは、次に示す方法で測定データの更新が終了したときのタイミングをとる必要があります。

- **STATus:CONDition?クエリを使う**

「STATus:CONDition?」は状態レジスタ(6-4ページ参照)の内容を問い合わせる命令です。測定データの更新中かそうでないかは、状態レジスタのビット0を読むことで判断できます。状態レジスタのビット0が「1」なら測定データの更新中、「0」なら測定データの問い合わせ可能を示します。

- **拡張イベントレジスタを使う**

状態レジスタの変化は、拡張イベントレジスタ(6-4ページ)に反映させることができます。

例 :STATus:FILTer1 FALL;:STATus:EESR 1;
EESR?;*SRE 8<PMT>
(STATus:EESR?の応答を読み出す)
LOOP
(サービスリクエストの発生を待つ)
:NUMeric[:NORMal]:VALue?<PMT>
(:NUMeric[:NORMal]:VALue?の応答を読み出す)
:STATus:EESR?<PMT>
(:STATus:EESR?の応答を読み出す)
(LOOPに戻る)

「STATus:FILTer1 FALL」は、状態レジスタのビット0が「1」から「0」に変化したときに、拡張イベントレジスタのビット0(FILTer1)を「1」にセットするように、遷移フィルタを設定することを示しています。

「STATus:EESR 1」は、拡張イベントレジスタのビット0だけをステータスバイトに反映するようにする命令です。

「STATus:EESR?」は、拡張イベントレジスタをクリアするために行っています。

「*SRE 8」は、拡張イベントレジスタの原因だけでサービスリクエストが発生するようにする命令です。

「:NUMeric[:NORMal]:VALue?」は、サービスリクエストが発生するまで実行されません。

- **COMMunicate:WAITコマンドを使う**

「COMMunicate:WAIT」は、特定のイベントが発生するのを待つ命令です。

例 :STATus:FILTer1 FALL;:STATus:EESR?
<PMT>
(STATus:EESR?の応答を読み出す)
LOOP
COMMunicate:WAIT 1<PMT>
:NUMeric[:NORMal]:VALue?<PMT>
(:NUMeric[:NORMal]:VALue?の応答を読み出す)
:STATus:EESR?<PMT>
(:STATus:EESR?の応答を読み出す)
(LOOPに戻る)

「STATus:FILTer1 FALL」および「STATus:EESR?」の意味は、前述の拡張イベントレジスタの場合と同一です。

「COMMunicate:WAIT 1」は、拡張イベントレジスタのビット0が「1」にセットされるのを待つことを示しています。

「:NUMeric[:NORMal]:VALue?」は、拡張イベントレジスタのビット0が「1」になるまで実行されません。

5.1 コマンド一覧表

コマンド	機能	ページ
COMMunicate グループ		
:COMMunicate?	通信に関する全設定値の問い合わせ	5-10
:COMMunicate:HEADer	クエリに対する応答を、ヘッダ付きか、ヘッダ付きでないかの設定/問い合わせ	5-10
:COMMunicate:LOCKout	ローカルロックアウトの設定/解除	5-10
:COMMunicate:OPSE	*OPC, *OPC?, *WAIの対象となるオーバーラップコマンドの設定/問い合わせ	5-11
:COMMunicate:OPSR?	オペレーションペンディングステータスレジスタの値の問い合わせ	5-11
:COMMunicate:OVERlap	オーバーラップ動作にするコマンドの設定/問い合わせ	5-11
:COMMunicate:REMOte	リモート/ローカルの設定	5-11
:COMMunicate:STATus?	回線固有のステータスの問い合わせ	5-11
:COMMunicate:VERBose	クエリに対する応答を、フルスベルか、省略形かの設定/問い合わせ	5-11
:COMMunicate:WAIT	指定された拡張イベントの待機	5-11
:COMMunicate:WAIT?	指定された拡張イベントのどれかが発生したときの応答の作成	5-11
CURSorグループ		
:CURSor?	カーソル測定に関する全設定値の問い合わせ	5-13
:CURSor:TREND?	トレンドにおけるカーソル測定に関する全設定値の問い合わせ	5-13
:CURSor:TREND:POStion<x>	トレンドにおけるカーソル位置の設定/問い合わせ	5-13
:CURSor:TREND[:STATe]	トレンドにおけるカーソル表示のON/OFFの設定/問い合わせ	5-13
:CURSor:TREND:TRACe<x>	トレンドにおけるカーソルの対象の設定/問い合わせ	5-13
:CURSor:TREND:{X<x> Y<x> DY}?	トレンドにおけるカーソル測定値の問い合わせ	5-13
:CURSor:WAVE?	波形表示におけるカーソル測定に関する全設定値の問い合わせ	5-13
:CURSor:WAVE:PATH	波形表示におけるカーソルパスの設定/問い合わせ	5-13
:CURSor:WAVE:POStion<x>	波形表示におけるカーソル位置の設定/問い合わせ	5-13
:CURSor:WAVE[:STATe]	波形表示におけるカーソル表示のON/OFFの設定/問い合わせ	5-14
:CURSor:WAVE:TRACe<x>	波形表示におけるカーソルの対象の設定/問い合わせ	5-14
:CURSor:WAVE:{X<x> DX PERDt Y<x> DY}?	波形表示におけるカーソル測定値の問い合わせ	5-14
DISPlayグループ		
:DISPlay?	画面表示に関する全設定値の問い合わせ	5-18
:DISPlay:FORMat	表示方式の設定/問い合わせ	5-18
:DISPlay:NUMeric?	数値表示に関する全設定値の問い合わせ	5-18
:DISPlay[:NUMeric]:IMPedance?	インピーダンス測定時における数値表示に関する全設定値の問い合わせ	5-18
:DISPlay[:NUMeric]:IMPedance:ICURsor	インピーダンス測定時における数値表示のカーソル位置の設定/問い合わせ	5-18
:DISPlay[:NUMeric]:IMPedance:OBJect	インピーダンス測定時における数値表示対象エレメントの設定/問い合わせ	5-18
:DISPlay[:NUMeric]:IMPedance:TYPE	インピーダンス測定時における数値表示項目の設定/問い合わせ	5-18
:DISPlay[:NUMeric]:NORMal?	電力測定時における数値表示に関する全設定値の問い合わせ	5-19
:DISPlay[:NUMeric]:NORMal:FCURsor	電力測定時における数値表示(全表示)のカーソル位置の設定/問い合わせ	5-19
:DISPlay[:NUMeric]:NORMal:IAMount	電力測定時における数値表示の方式の設定/問い合わせ	5-19
:DISPlay[:NUMeric]:NORMal:ICURsor	電力測定時における数値表示(分割表示)のカーソル位置の設定/問い合わせ	5-19
:DISPlay[:NUMeric]:NORMal:ITEM<x>	電力測定時における数値表示項目の設定/問い合わせ	5-20
:DISPlay[:NUMeric]:NORMal:PRESet	電力測定時における数値表示項目の表示順パターンのプリセット	5-20
:DISPlay:TREND?	トレンドに関する全設定値の問い合わせ	5-20
:DISPlay:TREND:ALL	全トレンドのON/OFFの一括設定	5-20
:DISPlay:TREND:FORMat	トレンドの表示フォーマットの設定/問い合わせ	5-20
:DISPlay:TREND:NORMal?	電力測定時における全トレンドに関する全設定値の問い合わせ	5-21
:DISPlay:TREND:NORMal:ITEM<x>?	電力測定時におけるトレンドに関する全設定値の問い合わせ	5-21

5.1 コマンド一覧表

コマンド	機能	ページ
:DISPlay:TREND:NORMAL:ITEM<x>[:FUNCTION]	電力測定時におけるトレンド項目の設定/問い合わせ	5-21
:DISPlay:TREND:NORMAL:ITEM<x>:SCALing?	電力測定時におけるトレンドのスケーリングに関する全設定値の問い合わせ	5-21
:DISPlay:TREND:NORMAL:ITEM<x>:SCALing:MODE	電力測定時におけるトレンドのスケーリング方式の設定/問い合わせ	5-21
:DISPlay:TREND:NORMAL:ITEM<x>:SCALing:VALue	電力測定時におけるトレンドのマニュアルスケーリング上下限値の設定/問い合わせ	5-22
:DISPlay:TREND:PDIV	トレンドの横軸(Point/div)の設定/問い合わせ	5-22
:DISPlay:TREND:REStart	トレンドをリスタートします。	5-22
:DISPlay:TREND[:SAMPling]	トレンド波形サンプリングのON/OFFの設定/問い合わせ	5-22
:DISPlay:TREND:Tdiv	電力測定時におけるトレンドの横軸(T/div)の設定/問い合わせ	5-22
:DISPlay:TREND:T<x>	トレンドのON/OFFの設定/問い合わせ	5-22
:DISPlay:WAVE?	波形表示に関する全設定値の問い合わせ	5-23
:DISPlay:WAVE:ALL	全波形表示のON/OFFの一括設定	5-23
:DISPlay:WAVE:FORMat	波形の表示フォーマットの設定/問い合わせ	5-23
:DISPlay:WAVE:GRATicule	グラティクル(目盛り)のタイプの設定/問い合わせ	5-23
:DISPlay:WAVE:INTerpolate	波形の補間方式の設定/問い合わせ	5-23
:DISPlay:WAVE:MAPPING?	分割フォーマットへの波形の割り付けに関する全設定値の問い合わせ	5-23
:DISPlay:WAVE:MAPPING[:MODE]	分割フォーマットへの波形の割り付け方法の設定/問い合わせ	5-23
:DISPlay:WAVE:MAPPING:{U<x> I<x>}	分割フォーマットへの波形の割り付けの設定/問い合わせ	5-23
:DISPlay:WAVE:SVALue	スケール値表示のON/OFFの設定/問い合わせ	5-24
:DISPlay:WAVE:TLABel	波形ラベル名表示のON/OFFの設定/問い合わせ	5-24
:DISPlay:WAVE:{U<x> I<x>}	波形表示のON/OFFの設定/問い合わせ	5-24
FILEグループ		
:FILE?	ファイル操作に関する全設定値の問い合わせ	5-26
:FILE:CDIRectory	カレントディレクトリの変更	5-26
:FILE:DELeTe:IMAGe:{TIFF BMP PSCript}	画面イメージデータファイルの削除	5-26
:FILE:DELeTe:NUMeric:{ASCIi FLOat}	数値データファイルの削除	5-26
:FILE:DELeTe:SETup	設定情報ファイルの削除	5-26
:FILE:DELeTe:WAVE:{BINary ASCIi FLOat}	波形表示データファイルの削除	5-26
:FILE:DRive	対象ドライブの設定	5-26
:FILE:FORMat	フロッピーディスクのフォーマットの実行	5-26
:FILE:FREE?	対象ドライブの空き容量の問い合わせ	5-27
:FILE:LOAD:ABORt	ファイルのロードの中止	5-27
:FILE:LOAD:FGWave	インピーダンス測定用負荷電流のパターン波形ファイルのロード	5-27
:FILE:LOAD:SETup	設定情報ファイルのロード	5-27
:FILE:MDIRectory	ディレクトリの作成	5-27
:FILE:PATH?	カレントディレクトリの絶対パスの問い合わせ	5-27
:FILE:SAVE?	ファイルの保存に関する全設定値の問い合わせ	5-27
:FILE:SAVE:ABORt	ファイルの保存の中止	5-27
:FILE:SAVE:ANAMing	保存するファイル名を自動的につけるかどうかの設定/問い合わせ	5-27
:FILE:SAVE:COMMeNt	保存するファイルに付加するコメントの設定/問い合わせ	5-27
:FILE:SAVE:NUMeric?	数値データのファイル保存に関する全設定値の問い合わせ	5-28
:FILE:SAVE:NUMeric[:EXECute]	数値データのファイルの保存	5-28
:FILE:SAVE:NUMeric:NORMAL?	電力測定時における数値データのファイル保存に関する全設定値の問い合わせ	5-28
:FILE:SAVE:NUMeric:NORMAL:ALL	電力測定時に数値データをファイルに保存するときの全エレメント・ファンクションの出力ON/OFFの一括設定	5-28

コマンド	機能	ページ
:FILE:SAVE:NUMeric:NORMal:{ELEment<x> SIGMA SIGMB SIGMC}	電力測定時に数値データをファイルに保存するときの{各エレメント ΣA ΣB ΣC}の出力ON/OFFの設定/問い合わせ	5-28
:FILE:SAVE:NUMeric:NORMal:PRESet<x>	電力測定時に数値データをファイルに保存するときのエレメント・ファンクションの出力ON/OFFパターンのプリセット	5-28
:FILE:SAVE:NUMeric:NORMal:<電力測定ファンクション>	電力測定時に数値データをファイルに保存するときの各ファンクションの出力ON/OFFの設定/問い合わせ	5-29
:FILE:SAVE:NUMeric:TYPE	保存する数値データの形式の設定/問い合わせ	5-29
:FILE:SAVE:SETup[:EXECute]	設定情報のファイル保存の実行	5-29
:FILE:SAVE:WAVE?	波形表示データのファイル保存に関する全設定値の問い合わせ	5-29
:FILE:SAVE:WAVE[:EXECute]	波形表示データのファイル保存の実行	5-29
:FILE:SAVE:WAVE:TRACE	ファイル保存の対象とする波形の設定/問い合わせ	5-29
:FILE:SAVE:WAVE:TYPE	保存する波形表示データの形式の設定/問い合わせ	5-29
HCOPYグループ		
:HCOPY?	画面イメージデータの出力に関する全設定値の問い合わせ	5-31
:HCOPY:ABORt	画面イメージデータ出力、紙送りの中止	5-31
:HCOPY:BMP?	BMPフォーマットに関する全設定値の問い合わせ	5-31
:HCOPY:BMP:COLor	BMPフォーマットの色調の設定/問い合わせ	5-31
:HCOPY:BMP:COMPression	BMPフォーマットのデータ圧縮の設定/問い合わせ	5-31
:HCOPY:COMMeNt	画面下部に表示するコメントの設定/問い合わせ	5-31
:HCOPY:DIRection	画面イメージデータの出力先の設定/問い合わせ	5-32
:HCOPY:EXECute	画面イメージデータ出力の実行	5-32
:HCOPY:FORMat	保存する画面イメージデータのファイル形式の設定/問い合わせ	5-32
:HCOPY:PRINter?	内蔵プリンタ出力に関する全設定値の問い合わせ	5-32
:HCOPY:PRINter:DLISt?	数値データリストの内蔵プリンタでの印字に関する全設定値の問い合わせ	5-32
:HCOPY:PRINter:DLISt[:EXECute]	数値データリストの内蔵プリンタでの印字実行	5-32
:HCOPY:PRINter:DLISt:INFORmation	数値データリストを内蔵プリンタで印字するときに設定情報を付加する/しないの設定/問い合わせ	5-32
:HCOPY:PRINter:DLISt:NORMal?	電力測定時における数値データリストの印字に関する全設定値の問い合わせ	5-32
:HCOPY:PRINter:DLISt:NORMal:ALL	電力測定時に数値データリストを内蔵プリンタで印字するときの全エレメント・ファンクションの出力ON/OFFの一括設定	5-33
:HCOPY:PRINter:DLISt:NORMal:{ELEment<x> SIGMA SIGMB SIGMC}	電力測定時に数値データリストを内蔵プリンタで印字するときの{各エレメント ΣA ΣB ΣC}の出力ON/OFFの設定/問い合わせ	5-33
:HCOPY:PRINter:DLISt:NORMal:PRESet<x>	電力測定時に数値データリストを内蔵プリンタで印字するときのエレメント・ファンクションの出力ON/OFFパターンのプリセット	5-33
:HCOPY:PRINter:DLISt:NORMal:<電力測定ファンクション>	電力測定時に数値データリストを内蔵プリンタで印字するときの各ファンクションの出力ON/OFFの設定/問い合わせ	5-33
:HCOPY:PRINter:FEED	内蔵プリンタの紙送りの実行	5-33
:HCOPY:SAVE?	ファイル保存に関する全設定値の問い合わせ	5-33
:HCOPY:SAVE:ANAMing	保存するファイル名を自動的につけるかどうかの設定/問い合わせ	5-33
:HCOPY:SAVE:COMMeNt	保存するファイルに付加するコメントの設定/問い合わせ	5-33
:HCOPY:SAVE:NAME	保存するファイル名の設定/問い合わせ	5-34
:HCOPY:TIFF?	TIFFフォーマットに関する全設定値の問い合わせ	5-34
:HCOPY:TIFF:COLor	TIFFフォーマットの色調の設定/問い合わせ	5-34
HOLDグループ		
:HOLD	出力データ(表示・通信など)のホールドの設定/問い合わせ	5-34

5.1 コマンド一覧表

コマンド	機能	ページ
IMAGeグループ		
:IMAGe?	画面イメージデータの出力の全設定値の問い合わせ	5-35
:IMAGe:COLor	出力する画面イメージデータの色調の設定/問い合わせ	5-35
:IMAGe:FORMat	出力する画面イメージデータの出力形式の設定/問い合わせ	5-35
:IMAGe:SEND?	画面イメージデータの問い合わせ	5-35
IMPedanceグループ		
:IMPedance?	インピーダンス測定に関する全設定値の問い合わせ	5-38
:IMPedance:CURRent:MRANge?	現在の電流レンジの問い合わせ	5-38
:IMPedance:DCControl?	直流負荷電流に関する全設定値の問い合わせ	5-38
:IMPedance:DCControl:DETail?	直流負荷電流の詳細設定に関する全設定値の問い合わせ	5-38
:IMPedance:DCControl:DETail:HOLD	ホールドしたときの本機器の動作(直流電子負荷装置への制御信号の扱い)の設定/問い合わせ	5-39
:IMPedance:DCControl:DETail:LIMit	直流負荷電流の範囲の設定/問い合わせ	5-39
:IMPedance:DCControl:DETail:RATio	直流電子負荷装置への制御信号1(V)あたりの電流値の設定/問い合わせ	5-39
:IMPedance:DCControl:OFFSet	直流負荷電流の電流値の設定/問い合わせ	5-39
:IMPedance:DCControl:OUTPut	直流負荷電流のON/OFFの設定/問い合わせ	5-39
:IMPedance:MEASure?	インピーダンス測定のFFT窓幅の問い合わせ	5-39
:IMPedance:MEASure:ANALysis?	インピーダンス測定モードの種類の問い合わせ	5-39
:IMPedance:MEASure:ARRay?	ロードされたパターンファイルの配列情報の問い合わせ	5-39
:IMPedance:MEASure:TYPE	インピーダンス測定のFFT窓幅の設定/問い合わせ	5-39
:IMPedance[:STATe]	インピーダンス測定モードのON/OFFの設定/問い合わせ	5-40
:IMPedance:SUPerpose?	インピーダンス測定用負荷電流に関する全設定値の問い合わせ	5-40
:IMPedance:SUPerpose:AMPLitude	インピーダンス測定用負荷電流の振幅の設定/問い合わせ	5-40
:IMPedance:SUPerpose:DETail?	インピーダンス測定用負荷電流の詳細設定に関する全設定値の問い合わせ	5-40
:IMPedance:SUPerpose:DETail:HOLD	ホールドしたときの本機器の動作(インピーダンス測定用電子負荷装置への制御信号の扱い)の設定/問い合わせ	5-40
:IMPedance:SUPerpose:DETail:LIMit	インピーダンス測定用負荷電流の範囲の設定/問い合わせ	5-40
:IMPedance:SUPerpose:DETail:RATio	インピーダンス測定用電子負荷装置への制御信号1(V)あたりの電流値の設定/問い合わせ	5-40
:IMPedance:SUPerpose:DETail:WAVEform	インピーダンス測定用負荷電流の波形の設定/問い合わせ	5-40
:IMPedance:SUPerpose:FREQuency?	インピーダンス測定用負荷電流の周波数に関する全設定値の問い合わせ	5-40
:IMPedance:SUPerpose:FREQuency:RANge	インピーダンス測定用負荷電流の周波数レンジの設定/問い合わせ	5-41
:IMPedance:SUPerpose:FREQuency:VALue	インピーダンス測定用負荷電流の周波数の設定/問い合わせ	5-41
:IMPedance:SUPerpose:OFFSet	インピーダンス測定用負荷電流の直流成分の大きさの設定/問い合わせ	5-41
:IMPedance:SUPerpose:OUTPut?	インピーダンス測定用負荷電流の出力に関する全設定値の問い合わせ	5-41
:IMPedance:SUPerpose:OUTPut[:STATe]	インピーダンス測定用負荷電流のON/OFFの設定/問い合わせ	5-41
:IMPedance:SUPerpose:OUTPut:TYPE	インピーダンス測定用負荷電流の出力タイプの設定/問い合わせ	5-41
:IMPedance:VOLTag?	インピーダンス測定の電圧センス入力に関する全設定値の問い合わせ	5-41
:IMPedance:VOLTag:ESTimate?	全インピーダンス測定エレメントのインピーダンス目安値の問い合わせ	5-41
:IMPedance:VOLTag:ESTimate[:ALL]	全インピーダンス測定エレメントのインピーダンス目安値の一括設定	5-41
:IMPedance:VOLTag:ESTimate:ELEMeNt<x>	各インピーダンス測定エレメントのインピーダンス目安値の設定/問い合わせ	5-42

コマンド	機能	ページ
:IMPedance:VOLTage:INITialize	電圧レンジを初期レンジに設定	5-42
:IMPedance:VOLTage:MRANge?	現在の電圧測定レンジの問い合わせ	5-42
:IMPedance:VOLTage:RANge?	全インピーダンス測定エレメントの電圧レンジモードの問い合わせ	5-42
:IMPedance:VOLTage:RANge[:ALL]	全インピーダンス測定エレメントの電圧レンジモードの一括設定	5-42
:IMPedance:VOLTage:RANge:ELEment<x>	各インピーダンス測定エレメントの電圧レンジモードの設定/問い合わせ	5-42
:IMPedance:VOLTage:TERMinal?	全インピーダンス測定エレメントの電圧入力端子の問い合わせ	5-42
:IMPedance:VOLTage:TERMinal[:ALL]	全インピーダンス測定エレメントの電圧入力端子の一括設定	5-42
:IMPedance:VOLTage:TERMinal:ELEment<x>	各インピーダンス測定エレメントの電圧入力端子の設定/問い合わせ	5-42
INPutグループ		
:INPut?	入力エレメントに関する全設定値の問い合わせ	5-45
[:INPut]:CURRent?	電流測定に関する全設定値の問い合わせ	5-45
[:INPut]:CURRent:AUTO[:ALL]	全電力測定エレメントの電流オートレンジON/OFFの一括設定	5-45
[:INPut]:CURRent:AUTO:ELEment<x>	各電力測定エレメントの電流オートレンジON/OFFの設定/問い合わせ	5-46
[:INPut]:CURRent:MRANge?	現在の電流測定レンジの問い合わせ	5-46
[:INPut]:CURRent:RANge?	全電力測定エレメントの電流レンジの問い合わせ	5-46
[:INPut]:CURRent:RANge[:ALL]	全電力測定エレメントの電流レンジの一括設定	5-46
[:INPut]:CURRent:RANge:ELEment<x>	各電力測定エレメントの電流レンジの設定/問い合わせ	5-47
[:INPut]:CURRent:SRATio?	全電力測定エレメントの電流センサスケール定数の問い合わせ	5-47
[:INPut]:CURRent:SRATio[:ALL]	全電力測定エレメントの電流センサスケール定数の一括設定	5-47
[:INPut]:CURRent:SRATio:ELEment<x>	各電力測定エレメントの電流センサスケール定数の設定/問い合わせ	5-47
[:INPut]:CURRent:TERMinal?	全電力測定エレメントの電流測定端子の問い合わせ	5-47
[:INPut]:CURRent:TERMinal[:ALL]	全電力測定エレメントの電流測定端子の一括設定	5-47
[:INPut]:CURRent:TERMinal:ELEment<x>	各電力測定エレメントの電流測定端子の設定/問い合わせ	5-48
[:INPut]:FILTer?	フィルタに関する全設定値の問い合わせ	5-48
[:INPut]:FILTer:LINE?	全エレメントのラインフィルタの設定の問い合わせ	5-48
[:INPut]:FILTer[:LINE][:ALL]	全エレメントのラインフィルタの一括設定	5-48
[:INPut]:FILTer[:LINE]:ELEment<x>	各エレメントのラインフィルタの設定/問い合わせ	5-48
[:INPut]:FILTer:ZCRoss?	全電力測定エレメントのゼロクロスフィルタの設定の問い合わせ	5-48
[:INPut]:FILTer:ZCRoss[:ALL]	全電力測定エレメントのゼロクロスフィルタの一括設定	5-48
[:INPut]:FILTer:ZCRoss:ELEment<x>	各電力測定エレメントのゼロクロスフィルタの設定/問い合わせ	5-48
[:INPut]:MODUle?	入力エレメントタイプの問い合わせ	5-49
[:INPut]:NULL	NULL機能のON/OFFの設定/問い合わせ	5-49
[:INPut]:POVer?	ピークオーバ情報の問い合わせ	5-49
[:INPut]:SCALing?	スケールリングに関する全設定値の問い合わせ	5-49
[:INPut]:SCALing:{PT CT SFACTOR}?	全エレメントのスケールリング定数の問い合わせ	5-49
[:INPut]:SCALing:{PT CT SFACTOR}[:ALL]	全エレメントのスケールリング定数の一括設定	5-49
[:INPut]:SCALing:{PT CT SFACTOR}:ELEment<x>	各エレメントのスケールリング定数の設定/問い合わせ	5-49
[:INPut]:SCALing:STATe?	全エレメントのスケールリングON/OFFの問い合わせ	5-50
[:INPut]:SCALing[:STATe][:ALL]	全エレメントのスケールリングON/OFFの一括設定	5-50
[:INPut]:SCALing[:STATe]:ELEment<x>	各エレメントのスケールリングON/OFFの設定/問い合わせ	5-50
[:INPut]:SYNChronize?	全電力測定エレメントの同期ソースの問い合わせ	5-50
[:INPut]:SYNChronize[:ALL]	全電力測定エレメントの同期ソースの一括設定	5-50
[:INPut]:SYNChronize:ELEment<x>	各電力測定エレメントの同期ソースの設定/問い合わせ	5-50
[:INPut]:VOLTage?	電圧測定に関する全設定値の問い合わせ	5-50
[:INPut]:VOLTage:AUTO[:ALL]	全電力測定エレメントの電圧オートレンジON/OFFの一括設定	5-50
[:INPut]:VOLTage:AUTO:ELEment<x>	各電力測定エレメントの電圧オートレンジON/OFFの設定/問い合わせ	5-51
[:INPut]:VOLTage:MRANge?	現在の電圧測定レンジの問い合わせ	5-51
[:INPut]:VOLTage:RANge?	全電力測定エレメントの電圧レンジの問い合わせ	5-51

5.1 コマンド一覧表

コマンド	機能	ページ
[:INPut] :VOLTagE :RANGe [:ALL]	全電力測定エレメントの電圧レンジの一括設定	5-51
[:INPut] :VOLTagE :RANGe :ELEMEnt <x>	各電力測定エレメントの電圧レンジの設定/問い合わせ	5-51
[:INPut] :WIRing	結線方式の設定/問い合わせ	5-52
INTEGrateグループ		
:INTEGrate?	積算に関する全設定値の問い合わせ	5-54
:INTEGrate:ACAL	オートキャリブレーションのON/OFFの設定/問い合わせ	5-54
:INTEGrate:CURREnt?	全電力測定エレメントの電流積算の電流モードの問い合わせ	5-54
:INTEGrate:CURREnt [:ALL]	全電力測定エレメントの電流積算の電流モードの一括設定	5-54
:INTEGrate:CURREnt :ELEMEnt <x>	各電力測定エレメントの電流積算の電流モードの設定/問い合わせ	5-54
:INTEGrate:INdependent	エレメント別積算のON/OFFの設定/問い合わせ	5-54
:INTEGrate:MODE	積算モードの設定/問い合わせ	5-54
:INTEGrate:RESet	積算値のリセット	5-55
:INTEGrate:RTIME <x>?	実時間制御積算モードにおける積算スタート/ストップ時刻の問い合わせ	5-55
:INTEGrate:RTIME <x> : {START END}	実時間制御積算モードにおける積算{スタート ストップ}時刻の設定/問い合わせ	5-55
:INTEGrate:START	積算のスタート	5-56
:INTEGrate:STATe?	積算状態の問い合わせ	5-56
:INTEGrate:STOP	積算のストップ	5-56
:INTEGrate:TIMer <x>	積算タイマ時間の設定/問い合わせ	5-56
MEASureグループ		
:MEASure?	測定に関する全設定値の問い合わせ	5-58
:MEASure:AVERaging?	アベレージングに関する全設定値の問い合わせ	5-58
:MEASure:AVERaging:COUNT	電力測定時におけるアベレージング係数の設定/問い合わせ	5-58
:MEASure:AVERaging [:STATe]	アベレージングのON/OFFの設定/問い合わせ	5-58
:MEASure:AVERaging:TYPE	電力測定時におけるアベレージングのタイプの設定/問い合わせ	5-59
:MEASure:DMeasure?	デルタ演算に関する全設定値の問い合わせ	5-59
:MEASure:DMeasure:OBJect	デルタ演算対象の設定/問い合わせ	5-59
:MEASure:DMeasure:TYPE	デルタ演算モードの設定/問い合わせ	5-59
:MEASure:FREQuency?	周波数測定に関する全設定値の問い合わせ	5-59
:MEASure:FREQuency:ITEM	周波数測定対象の設定/問い合わせ	5-59
:MEASure:FUNCTION <x>?	ユーザ定義ファンクションに関する全設定値の問い合わせ	5-60
:MEASure:FUNCTION <x> : EXPReSSion	ユーザ定義ファンクションの演算式の設定/問い合わせ	5-60
:MEASure:FUNCTION <x> [:STATe]	ユーザ定義ファンクションの有効(ON)/無効(OFF)の設定/問い合わせ	5-60
:MEASure:FUNCTION <x> : UNIT	ユーザ定義ファンクションの演算結果に付加する単位の設定/問い合わせ	5-60
:MEASure:MHOLd	MAX HOLD機能のON/OFFの設定/問い合わせ	5-60
:MEASure:PC?	Pc(Corrected Power)の算出に関する全設定値の問い合わせ	5-60
:MEASure:PC:IEC	Pc(Corrected Power)の算出式の設定/問い合わせ	5-60
:MEASure:PC:P <x>	Pc(Corrected Power)算出のためのパラメータの設定/問い合わせ	5-61
:MEASure:PHASe	位相差の表示形式の設定/問い合わせ	5-61
:MEASure:SFORMula	S(無効電力)の算出式の設定/問い合わせ	5-61
NUMericグループ		
:NUMeric?	数値データの出力に関する全情報の問い合わせ	5-63
:NUMeric:FORMat	「:NUMeric:{NORMal HARMonics LIST}:VALue?」で送信される数値データのフォーマットの設定/問い合わせ	5-63
:NUMeric:IMPedance?	インピーダンス測定時における数値データの出力に関する全設定値の問い合わせ	5-63
:NUMeric:IMPedance:ARRAy	配列型ファンクションを出力する際のデータ数(配列数)の設定/問い合わせ	5-63
:NUMeric:IMPedance:CLEAr	インピーダンス測定時における数値データ出力項目のクリア	5-63
:NUMeric:IMPedance:ITEM <x>	インピーダンス測定時における数値データ出力項目の設定/問い合わせ	5-64
:NUMeric:IMPedance:NUMber	「:NUMeric:IMPedance:VALue?」で送信される数値データの項目数の設定/問い合わせ	5-64
:NUMeric:IMPedance:PRESet	インピーダンス測定時における数値データ出力項目パターンのプリセット	5-64
:NUMeric:IMPedance:VALue?	インピーダンス測定時における数値データの問い合わせ	5-64

コマンド	機能	ページ
:NUMeric:NORMal?	電力測定時における数値データの出力に関する全設定値の問い合わせ	5-64
:NUMeric[:NORMal]:CLear	電力測定時における数値データの出力項目のクリア	5-65
:NUMeric[:NORMal]:ITEM<x>	電力測定時における数値データ出力項目の設定/問い合わせ	5-65
:NUMeric[:NORMal]:NUMBER	「:NUMeric:NORMal:VALue?」で送信される数値データの個数の設定/問い合わせ	5-65
:NUMeric[:NORMal]:PRESet	電力測定時における数値データの出力項目パターンのプリセット	5-65
:NUMeric[:NORMal]:VALue?	電力測定時における数値データの問い合わせ	5-65
RATEグループ		
:RATE	電力測定時におけるデータ更新レートの設定/問い合わせ	5-68
STATusグループ		
:STATus?	通信のステータス機能に関する全設定値の問い合わせ	5-69
:STATus:CONDition?	状態レジスタの内容の問い合わせ	5-69
:STATus:EESSE	拡張イベントイネーブルレジスタの設定/問い合わせ	5-69
:STATus:EESR?	拡張イベントレジスタの内容の問い合わせ、レジスタのクリア	5-69
:STATus:ERRor?	発生したエラーのコードとメッセージ内容(エラーキューの先頭)の問い合わせ	5-70
:STATus:FILTer<x>	遷移フィルタの設定/問い合わせ	5-70
:STATus:QENable	エラー以外のメッセージをエラーキューに格納する(ON)/しない(OFF)の設定/問い合わせ	5-70
:STATus:QMESsage	「:STATus:ERRor?」の応答にメッセージ内容を付ける(ON)/付けない(OFF)の設定/問い合わせ	5-70
:STATus:SPOLi?	シリアルポールの実行	5-70
STOReグループ		
:STORe?	ストア/リコールに関する全設定値の問い合わせ	5-72
:STORe:COUNt	ストア回数の設定/問い合わせ	5-72
:STORe:DIRection	ストア先の設定/問い合わせ	5-72
:STORe:FILE?	ファイルへのストアしたデータの保存に関する全設定値の問い合わせ	5-72
:STORe:FILE:ANAMing	ストアしたデータを保存するときのファイル名を自動的につけるかどうかの設定/問い合わせ	5-72
:STORe:FILE:COMMeNt	ストアしたデータを保存するときのファイルに付加するコメントの設定/問い合わせ	5-72
:STORe:FILE:NAME	ストアしたデータを保存するときのファイル名の設定/問い合わせ	5-73
:STORe:INTerval	ストアインターバルの設定/問い合わせ	5-73
:STORe:ITEM	ストアする内容の設定/問い合わせ	5-73
:STORe:MEMory:CONVert:ABORt	ストアデータのメモリからファイルへのコンバートの中止	5-73
:STORe:MEMory:CONVert:EXECute	ストアデータのメモリからファイルへのコンバートの実行	5-73
:STORe:MEMory:INITialize	ストアメモリの初期化の実行	5-73
:STORe:MODE	データのストア/リコールの設定/問い合わせ	5-73
:STORe:NUMeric?	数値データのストアに関する全設定値の問い合わせ	5-74
:STORe:NUMeric:NORMal?	電力測定時における数値データのストアに関する全設定値の問い合わせ	5-74
:STORe:NUMeric:NORMal:ALL	電力測定時に数値データをストアするときの全エレメント・ファンクションの出力ON/OFFの一括設定	5-74
:STORe:NUMeric:NORMal:{ELEMeNt<x> SIGMA SIGMB SIGMC}	電力測定時に数値データをストアするときの{各エレメント ΣA ΣB ΣC}の出力ON/OFFの設定/問い合わせ	5-74
:STORe:NUMeric:NORMal:PRESet<x>	電力測定時に数値データをストアするときのエレメント・ファンクションの出力ON/OFFパターンのプリセット	5-74
:STORe:NUMeric:NORMal:<電力測定ファンクション>	電力測定時に数値データをストアするときの各ファンクションの出力ON/OFFの設定/問い合わせ	5-74
:STORe:RECall	リコールするデータ番号の設定/問い合わせ	5-74
:STORe:RTIME?	実時間ストアモードにおけるストアスタート/ストップ日時との問い合わせ	5-75

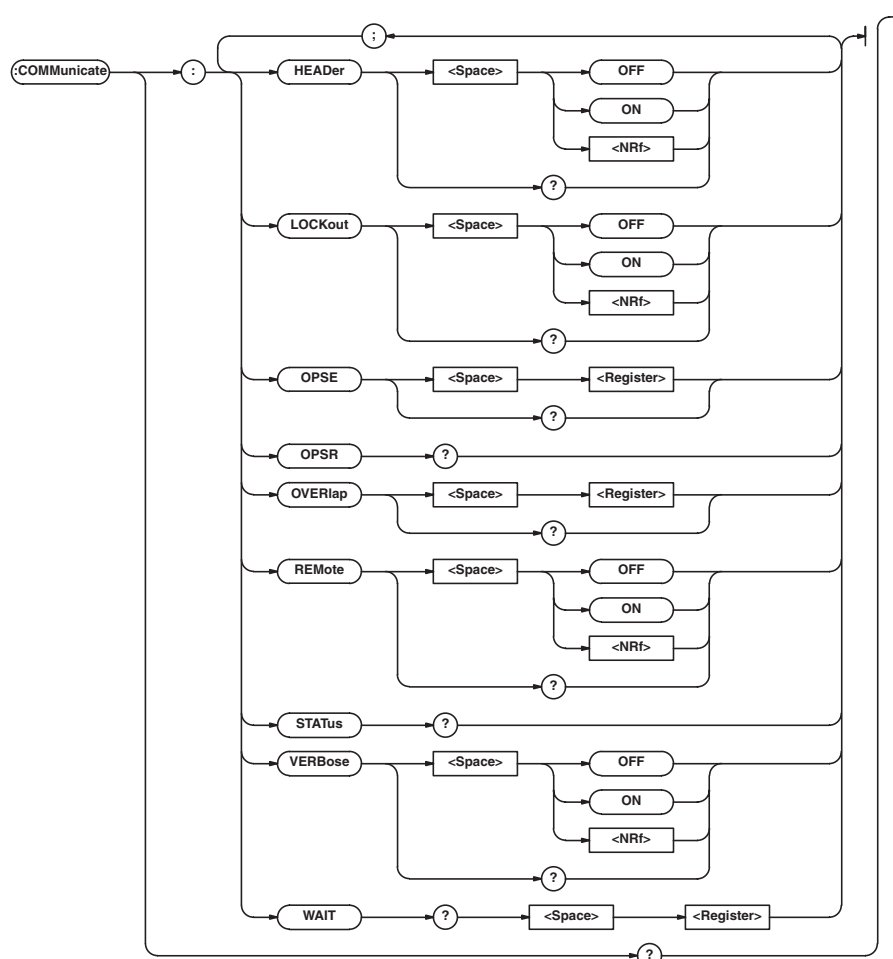
5.1 コマンド一覧表

コマンド	機能	ページ
:STORe:RTIME:{START END}	実時間ストアモードにおけるストア{スタート ストップ}日時の設定/問い合わせ	5-75
:STORe:SMODE	ストアモードの設定/問い合わせ	5-75
:STORe:START	データのストアの開始	5-75
:STORe:STOP	データのストアの終了	5-75
:STORe:WAVE?	波形表示データのストアに関する全設定値の問い合わせ	5-75
:STORe:WAVE:ALL	波形表示データをストアするときの全波形の出力ON/OFFの一括設定	5-75
:STORe:WAVE:{U<x> I<x>}	波形表示データをストアするときの各波形の出力ON/OFFの設定/問い合わせ	5-75
SYSTemグループ		
:SYSTem?	システムに関する全設定値の問い合わせ	5-76
:SYSTem:DATE	日付の設定/問い合わせ	5-76
:SYSTem:LANGUage	メッセージの言語の設定/問い合わせ	5-77
:SYSTem:LCD?	LCD画面に関する全設定値の問い合わせ	5-77
:SYSTem:LCD:BRIGHtness	LCD画面の輝度設定/問い合わせ	5-77
:SYSTem:LCD:COLor?	LCD画面の表示色に関する全設定値の問い合わせ	5-77
:SYSTem:LCD:COLor:GRAPh?	グラフィック項目の表示色に関する全設定値の問い合わせ	5-77
:SYSTem:LCD:COLor:GRAPh:{BACKground GRATicule CURSor U<x> I<x>}	{背景 目盛り カーソル 電圧波形 電流波形}の表示色の設定/問い合わせ	5-77
:SYSTem:LCD:COLor:GRAPh:MODE	グラフィック項目の表示色のモードの設定/問い合わせ	5-77
:SYSTem:LCD:COLor:TEXT?	テキスト項目の表示色に関する全設定値の問い合わせ	5-77
:SYSTem:LCD:COLor:TEXT:{LETTer BACKground BOX SUB SELected}	{文字(Menu Fore) メニューの背景(Menu Back) 選択されているメニュー(Select Box) ポップアップメニュー(Sub Menu) 選択されているキー(Selected Key)}の表示色の設定/問い合わせ	5-78
:SYSTem:LCD:COLor:TEXT:MODE	テキスト項目の表示色のモードの設定/問い合わせ	5-78
:SYSTem:SCSI?	SCSI-IDに関する全設定値の問い合わせ	5-78
:SYSTem:SCSI:HDMotor	内蔵ハードディスクのモーターのON/OFFの設定/問い合わせ	5-78
:SYSTem:SCSI:INITialize	SCSI関連のイニシャライズの実行	5-78
:SYSTem:SCSI:INTernalid	内蔵ハードディスクのSCSI-IDの設定/問い合わせ	5-78
:SYSTem:SCSI:OWNid	本機器のSCSI-IDの設定/問い合わせ	5-78
:SYSTem:TIME	時刻の設定/問い合わせ	5-78
WAVeformグループ		
:WAVeform?	波形表示データの全情報の問い合わせ	5-79
:WAVeform:BYTeorder	「:WAVeform:SEND?」で送信される波形表示データ(FLOAT形式)のバイト出力順序の設定/問い合わせ	5-79
:WAVeform:END	「:WAVeform:SEND?」で送信される波形表示データの出力終了点の設定/問い合わせ	5-79
:WAVeform:FORMat	「:WAVeform:SEND?」で送信される波形表示データのフォーマットの設定/問い合わせ	5-80
:WAVeform:LENGth?	「:WAVeform:TRACe」で指定した波形の全データ点数の問い合わせ	5-80
:WAVeform:SEND?	「:WAVeform:TRACe」で指定した波形表示データの問い合わせ	5-80
:WAVeform:SRAtE?	取り込んだ波形のサンプルレートの問い合わせ	5-80
:WAVeform:START	「:WAVeform:SEND?」で送信される波形表示データの出力開始点の設定/問い合わせ	5-80
:WAVeform:TRACe	WAVeformグループで対象となる波形の設定/問い合わせ	5-80
:WAVeform:TRIGger?	取り込んだ波形のトリガボジションの問い合わせ	5-80
WSETup (Wave SETup)グループ		
:WSETup?	波形観測に関する全設定値の問い合わせ	5-82
:WSETup:POSition?	波形の垂直ボジション(GND位置)に関する全設定値の問い合わせ	5-82
:WSETup:POSition:{UALL IALL}	全電力測定エレメントの{電圧 電流}波形の垂直ボジション(中心位置のレベル)の一括設定	5-82

コマンド	機能	ページ
:WSETup:POSiTion:{U<x> I<x>}	各電力測定エレメントの{電圧 電流}波形の垂直ポジション(中心位置のレベル)の設定/問い合わせ	5-82
:WSETup[:SAmPling]	波形サンプリングのON/OFFの設定/問い合わせ	5-82
:WSETup:TDIV	波形のTime/div値の設定/問い合わせ	5-82
:WSETup:TRIGger?	トリガに関する全設定値の問い合わせ	5-82
:WSETup:TRIGger:LeVel	トリガレベルの設定/問い合わせ	5-82
:WSETup:TRIGger:MODe	トリガモードの設定/問い合わせ	5-82
:WSETup:TRIGger:SLOPe	トリガスロープの設定/問い合わせ	5-82
:WSETup:TRIGger:SOURce	トリガソースの設定/問い合わせ	5-83
:WSETup:VZoom?	波形の垂直方向のズーム率に関する全設定値の問い合わせ	5-83
:WSETup:VZoom:{UALL IALL}	全電力測定エレメントの{電圧 電流}波形の垂直方向のズーム率の一括設定	5-83
:WSETup:VZoom:{U<x> I<x>}	各電力測定エレメントの{電圧 電流}波形の垂直方向のズーム率の設定/問い合わせ	5-83
共通コマンドグループ		
*CAL?	キャリブレーション(ゼロレベル補正, CAL(SHIFT+MEASURE)を押したのと同じ動作)の実行と, 結果の問い合わせ	5-84
*CLS	標準イベントレジスタ, 拡張イベントレジスタ, エラーキューのクリア	5-84
*ESE	標準イベントイネーブルレジスタの値の設定/問い合わせ	5-84
*ESR?	標準イベントレジスタの値の問い合わせ	5-85
*IDN?	機種種の問い合わせ	5-85
*OPC	指定したオーバーラップコマンド終了時に, 標準イベントレジスタのビット0(OPCビット)を1にセット	5-85
	整数に丸めた値が「0」以外のときクリア	5-85
*OPC?	*OPC?の送信で, 指定したオーバーラップコマンドが終了しているとき, ASCIIコードの「1」を返信	5-85
*OPT?	装備しているオプションの問い合わせ	5-85
*PSC	電源ON時にレジスタをクリアするかしないかの設定/問い合わせ	5-85
*RST	設定の初期化の実行	5-85
*SRE	サービスリクエストイネーブルレジスタの値の設定/問い合わせ	5-86
*STB?	ステータスバイトレジスタの値の問い合わせ	5-86
*TRG	SINGLE(SHIFT+HOLD)を押したのと同じ動作を実行	5-86
*TST?	セルフテストの実行と, 結果の問い合わせ	5-86
*WAI	指定したオーバーラップコマンド終了まで, *WAIに続く命令を待機	5-86

5.2 COMMunicate グループ

COMMunicateグループは、通信に関するグループです。
このグループに相当するフロントパネルのキーはありません。



:COMMunicate?

機能 通信に関するすべての設定値を問い合わせます。

構文 :COMMunicate?

例 :COMMUNICATE? -> :COMMUNICATE:

HEADER 1;OPSE 96;OVERLAP 96;

VERBOSE 1

:COMMunicate:HEADer

機能 クエリに対する応答を、ヘッダを付けて返送するか(例 DISPLAY:FORMAT NUMERIC)、付けないで返送するか(例 NUMERIC)を設定/問い合わせします。

構文 :COMMunicate:HEADer {<Boolean>}

:COMMunicate:HEADer?

例 :COMMUNICATE:HEADER ON

:COMMUNICATE:HEADER? ->

:COMMUNICATE:HEADER 1

:COMMunicate:LOCKout

機能 ローカルロックアウトを設定/解除します。

構文 :COMMunicate:LOCKout {<Boolean>}

:COMMunicate:LOCKout?

例 :COMMUNICATE:LOCKOUT ON

:COMMUNICATE:LOCKOUT? ->

:COMMUNICATE:LOCKOUT 1

解説

シリアル(RS-232)インタフェース専用のコマンドです。GP-IBインタフェースの場合、インタフェースメッセージとして用意されています。

:COMMunicate:OPSE**(Operation Pending Status Enable register)**

機能 *OPC, *OPC?, *WAIの対象となるオーバーラップコマンドを設定/問い合わせします。

構文 :COMMunicate:OPSE <Register>
:COMMunicate:OPSE?
<Register> = 0 ~ 65535, :COMMunicate:WAIT? コマンドの図参照

例 :COMMunicate:OPSE 65535
:COMMunicate:OPSE? -> :COMMunicate:OPSE 96

解説 上の例では、全ビットを1にして、すべてのオーバーラップコマンドを対象にしています。ただし、0固定のビットは1にならないので、問い合わせに対してはビット5、6だけが1になっています。

:COMMunicate:OPSR?**(Operation Pending Status Register)**

機能 オペレーションペンディングステータスレジスタの値を問い合わせます。

構文 :COMMunicate:OPSR?

例 :COMMunicate:OPSR? -> 0

解説 オペレーションペンディングステータスレジスタについては、:COMMunicate:WAIT? コマンドの図を参照してください。

:COMMunicate:OVERlap

機能 オーバーラップ動作にするコマンドを設定/問い合わせします。

構文 :COMMunicate:OVERlap <Register>
:COMMunicate:OVERlap?
<Register> = 0 ~ 65535, :COMMunicate:WAIT? コマンドの図参照

例 :COMMunicate:OVERlap 65535
:COMMunicate:OVERlap? -> :COMMunicate:OVERlap 96

解説

- ・上の例では、全ビットを1にして、すべてのオーバーラップコマンドを対象にしています。ただし、0固定のビットは1にならないので、問い合わせに対してはビット5、6だけが1になっています。
- ・COMMunicate:OVERlapを使った同期のとり方については、4-7ページを参照してください。
- ・上の例では、ビット5、6を1にして、すべてのオーバーラップコマンドを対象にしています (:COMMunicate:WAIT? コマンドの図参照)。

:COMMunicate:REMOte

機能 リモート/ローカルを設定します。ONのときにリモートになります。

構文 :COMMunicate:REMOte {<Boolean>}
:COMMunicate:REMOte?

例 :COMMunicate:REMOte ON
:COMMunicate:REMOte? -> :COMMunicate:REMOte 1

解説 シリアル(RS-232)インタフェース専用のコマンドです。GP-IBインタフェースの場合、インタフェースメッセージとして用意されています。

:COMMunicate:STATus?

機能 回線固有のステータスを問い合わせます。

構文 :COMMunicate:STATus?

例 :COMMunicate:STATus? -> :COMMunicate:STATus 0

解説 ステータスの各ビットの意味は次のとおりです。

ビット	GP-IB	RS-232
0	回復不能な送信エラー	パリティエラー
1	常に0	フレーミングエラー
2	常に0	ブレークキャラクタ検出
3 ~	常に0	常に0

ステータスは要因が発生したときに該当するビットがセットされ、読むとクリアされます。

:COMMunicate:VERBoSe

機能 クエリに対する応答を、フルスペルで返送するか(例 DISPLAY:FORMAT NUMERIC)、省略形で返送するか(例 DISP:FORM NUM)を設定/問い合わせします。

構文 :COMMunicate:VERBoSe {<Boolean>}
:COMMunicate:VERBoSe?

例 :COMMunicate:VERBoSe ON
:COMMunicate:VERBoSe? -> :COMMunicate:VERBoSe 1

:COMMunicate:WAIT

機能 指定された拡張イベントのどれかが発生するのを待ちます。

構文 :COMMunicate:WAIT <Register>
<Register> = 0 ~ 65535 (拡張イベントレジスタ, 6-4ページ参照)

例 :COMMunicate:WAIT 1

解説 COMMunicate:WAITを使った同期のとり方については、4-8ページを参照してください。

:COMMunicate:WAIT?

機能 指定された拡張イベントのどれかが発生したときに応答を作成します。

構文 :COMMunicate:WAIT? <Register>
<Register> = 0 ~ 65535 (拡張イベントレジスタ, 6-4ページ参照)

例 :COMMunicate:WAIT? 65535 -> 1

オペレーションペンディングステータスレジスタ/オーバーラップイネーブルレジスタ

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	ACS	PRN	0	0	0	0	0

ビット5(PRN) = 1のとき:

内蔵プリンタ動作未完了, ネットワークプリンタ出力未完了

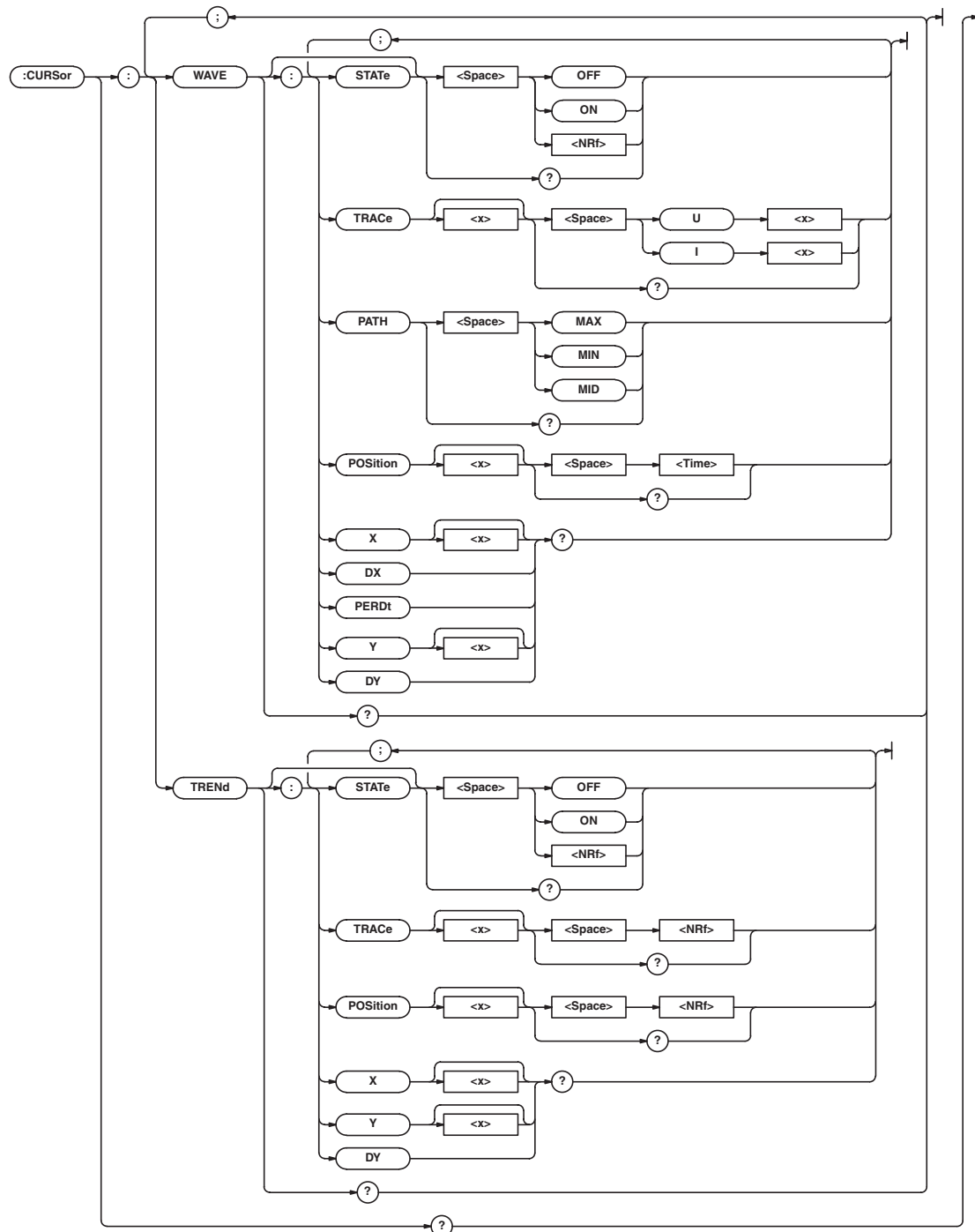
ビット6(ACS) = 1のとき:

メディアへのアクセス未完了

5.3 CURSorグループ

CURSorグループは、カーソル測定に関するグループです。フロントパネルのCURSOR(SHIFT+WAVE)と同じ設定、および設定内容の問い合わせができます。

ただし、インピーダンス測定エレメントだけが装備されている製品では、このグループのコマンドは無効です。



:CURSor?

機能 カーソル測定に関するすべての設定値を問い合わせます。

構文 :CURSor?

例 :CURSOR? -> :CURSOR:WAVE:STATE 0;
TRACE1 U1;TRACE2 I1;PATH MAX;
POSITION1 2.0E-03;
POSITION2 8.0E-03;:CURSOR:TREND:
STATE 0;TRACE1 1;TRACE2 2;
POSITION1 6;POSITION2 54

:CURSor:TREnd?

機能 トレンドにおけるカーソル測定に関するすべての設定値を問い合わせます。

構文 :CURSor:TREnd?

例 :CURSOR:TREND? ->
:CURSOR:TREND:STATE 1;TRACE1 1;
TRACE2 2;POSITION1 6;POSITION2 54

:CURSor:TREnd:POsition<x>

機能 トレンドにおけるカーソル位置を設定/問い合わせします。

構文 :CURSor:TREnd:POsition<x> {<Nrf>}
:CURSor:TREnd:POsition<x>?
<x> = 1, 2
<Nrf> = 0 ~ 500

例 :CURSOR:TREND:POSITION1 10
:CURSOR:TREND:POSITION1?->
:CURSOR:TREND:POSITION1 10

:CURSor:TREnd[:STATe]

機能 トレンドにおけるカーソル表示のON/OFFを設定/問い合わせします。

構文 :CURSor:TREnd[:STATe] {<Boolean>}
:CURSor:TREnd:STATe?

例 :CURSOR:TREND:STATE ON
:CURSOR:TREND:STATE? -> :CURSOR:
TREND:STATE 1

:CURSor:TREnd:TRACe<x>

機能 トレンドにおけるカーソルの対象を設定/問い合わせします。

構文 :CURSor:TREnd:TRACe<x> {<Nrf>}
:CURSor:TREnd:TRACe<x>?
<x> = 1, 2
<Nrf> = 1 ~ 16

例 :CURSOR:TREND:TRACE1 1
:CURSOR:TREND:TRACE1? -> :CURSOR:
TREND:TRACE1 1

:CURSor:TREnd:{X<x>|Y<x>|DY}?

機能 トレンドにおけるカーソル測定値を問い合わせます。

構文 :CURSor:TREnd:{X<x>|Y<x>|DY}?

X<x> = カーソル位置のトレンド時刻文字列(X1 = D+ , X2 = Dx)

Y<x> = カーソル位置のY軸値(Y1 = Y+ , Y2 = Yx)

DY = カーソル間のY軸値(ΔY)

<x> = 1, 2

例 :CURSOR:TREND:X1? -> "2003/04/01
12:34:56"

:CURSOR:TREND:Y1? -> 78.628E+00

解説 トレンドでカーソル表示がONになっていないときは、以下のようになります。

X<x>の場合: 「****/**/** **.*.***」を返します。

Y<x>, DYの場合: 「NAN(Not A Number)」を返します。

:CURSor:WAVE?

機能 波形表示におけるカーソル測定に関するすべての設定値を問い合わせます。

構文 :CURSor:WAVE?

例 :CURSOR:WAVE? -> :CURSOR:WAVE:
STATE 1;TRACE1 U1;TRACE2 I1;
PATH MAX;POSITION1 2.0E-03;
POSITION2 8.0E-03

:CURSor:WAVE:PATH

機能 波形表示におけるカーソルパスを設定/問い合わせします。

構文 :CURSor:WAVE:PATH {MAX|MIN|MID}
:CURSor:WAVE:PATH?

例 :CURSOR:WAVE:PATH MAX
:CURSOR:WAVE:PATH? -> :CURSOR:WAVE:
PATH MAX

:CURSor:WAVE:POsition<x>

機能 波形表示におけるカーソル位置を設定/問い合わせします。

構文 :CURSor:WAVE:POsition<x> {<時間>}
:CURSor:WAVE:POsition<x>?
<x> = 1, 2

<時間> = 0 ~ 5.00s

例 :CURSOR:WAVE:POSITION1 2MS
:CURSOR:WAVE:POSITION1? -> :CURSOR:
WAVE:POSITION1 2.0E-03

解説 <時間>の設定範囲および分解能は、波形のTime/div値(:WSETup:TDiv)によって決まります。

:CURSor:WAVE[:STATe]

機能 波形表示におけるカーソル表示のON/OFFを設定/問い合わせします。

構文 :CURSor:WAVE[:STATe] {<Boolean>}
:CURSor:WAVE:STATe?

例 :CURSOR:WAVE:STATE ON
:CURSOR:WAVE:STATE? -> :CURSOR:
WAVE:STATE 1

:CURSor:WAVE:TRACe<x>

機能 波形表示におけるカーソルの対象を設定/問い合わせします。

構文 :CURSor:WAVE:TRACe<x> {U<x>|I<x>}
:CURSor:WAVE:TRACe<x>?
TRACe<x>の<x>=1,2
U<x>,I<x>の<x>=1~4(電力測定エレメント)

例 :CURSOR:WAVE:TRACE1 U1
:CURSOR:WAVE:TRACE1? ->
:CURSOR:WAVE:TRACE1 U1

:CURSor:WAVE:{X<x>|DX|PERDt|Y<x>|DY}?

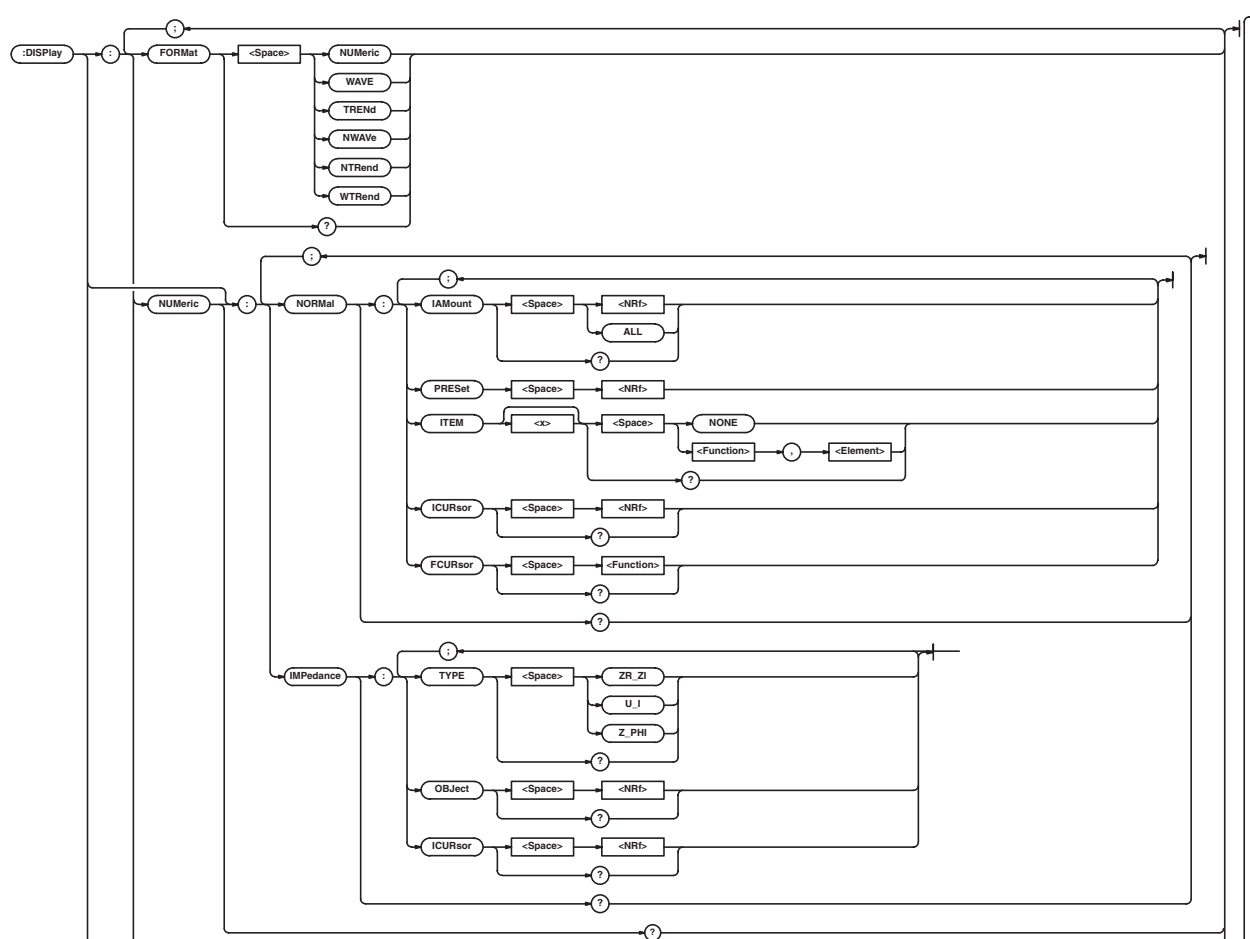
機能 波形表示におけるカーソル測定値を問い合わせます。

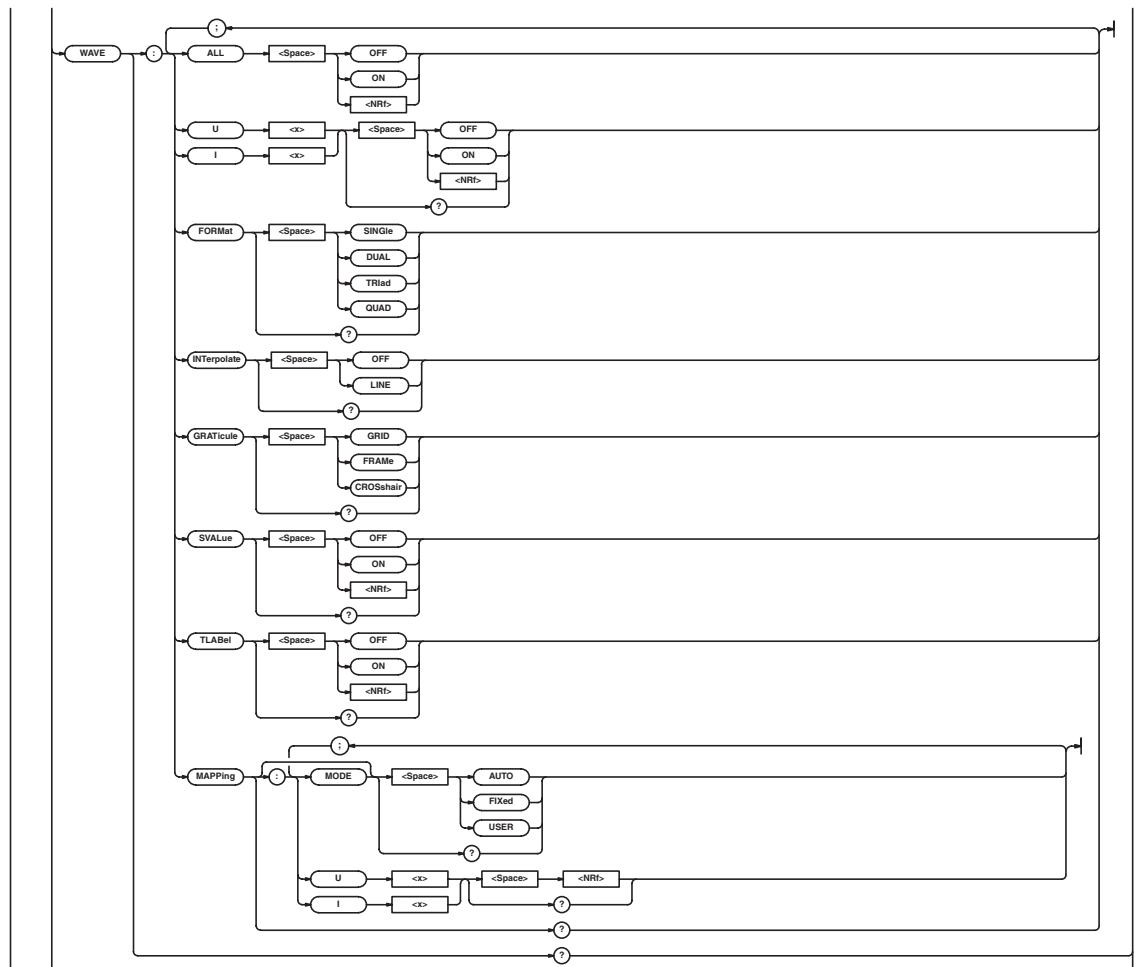
構文 :CURSor:WAVE:{X<x>|DX|PERDt|Y<x>|DY}?
X<x>=カーソル位置のX軸値($X1 = X +$, $X2 = Xx$)
DX=カーソル間のX軸値(ΔX)
PERDt=カーソル間の $1/\Delta T$ 値($1/\Delta X$)
Y<x>=カーソル位置のY軸値($Y1 = Y +$, $Y2 = Yx$)
DY=カーソル間のY軸値(ΔY)
<x>=1,2

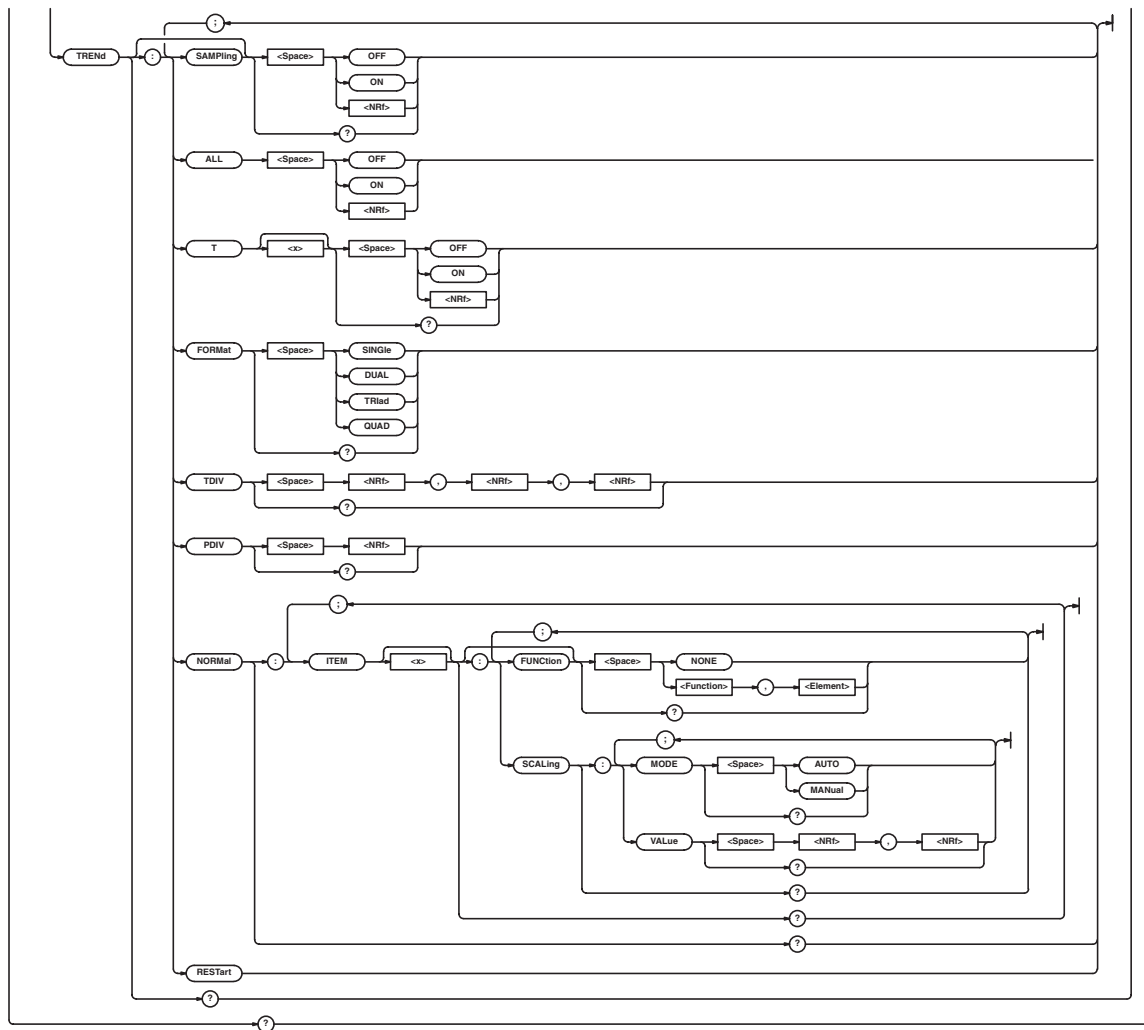
例 :CURSOR:WAVE:Y1? -> 78.628E+00
解説 波形表示でカーソル表示がONになっていないときは「NAN(Not A Number)」を返します。

5.4 DISPlayグループ

DISPlayグループは、画面表示に関するグループです。
フロントパネルのDISPLAYと同じ設定、および設定内容の問い合わせができます。







:DISPlay?

機能 画面表示に関するすべての設定値を問い合わせます。

構文 :DISPlay?

例 ・表示方式(:DISPlay:FORMat)が「NWAVE」のときの例
 :DISPlay? -> :DISPlay:FORMat
 NWAVE; (「:DISPlay:NUMeric?」の応答から最初の「:DISPlay:」の部分を除いたもの); (「:DISPlay:WAVE?」の応答)

:DISPlay:FORMat

機能 表示方式を設定/問い合わせします。

構文 :DISPlay:FORMat {NUMeric|WAVE|TREnd|NWAVE|NTREnd|WTREnd}

:DISPlay:FORMat?

NUMeric = 数値のみ表示

WAVE = 波形のみ表示

TREnd = トレンド

NWAVE = 数値と波形を両方表示

NTREnd = 数値とトレンドを両方表示

WTREnd = 波形とトレンドを両方表示

例 :DISPlay:FORMat NUMeric

:DISPlay:FORMat? -> :DISPlay:FORMat NUMeric

解説 ・電力測定時のみ有効な設定です。インピーダンス測定時は、この設定に関係なくインピーダンス測定専用表示となります。
 ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:NUMeric?

機能 数値表示に関するすべての設定値を問い合わせます。

構文 :DISPlay:NUMeric?

例 ・電力測定するとき(:IMPedance[:STATe]の設定が「OFF(0)」)
 :DISPlay:NUMeric? -> (「:DISPlay[:NUMeric]:NORMal?」の応答と同じ)
 ・インピーダンス測定するとき(:IMPedance[:STATe]の設定が「ON(1)」)
 :DISPlay:NUMeric? -> (「:DISPlay[:NUMeric]:IMPedance?」の応答と同じ)

解説 ・電力測定時のみ有効な設定です。インピーダンス測定時は、この設定に関係なくインピーダンス測定専用表示となります。
 ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay[:NUMeric]:IMPedance?

機能 インピーダンス測定時における数値表示に関するすべての設定値を問い合わせます。

構文 :DISPlay[:NUMeric]:IMPedance?

例 :DISPlay:NUMeric:IMPedance? ->

:DISPlay:NUMeric:IMPedance:

TYPE ZR_ZI;OBJECT 5;ICURSor 1

:DISPlay[:NUMeric]:IMPedance:ICURSor

機能 インピーダンス測定時における数値表示のカーソル位置を設定/問い合わせします。

構文 :DISPlay[:NUMeric]:IMPedance:

ICURSor {<NRf>}

:DISPlay[:NUMeric]:IMPedance:

ICURSor?

<NRf> = 1 ~ 100

例 :DISPlay:NUMeric:IMPedance:

ICURSor 1

:DISPlay:NUMeric:IMPedance:

ICURSor? -> :DISPlay:NUMeric:

IMPedance:ICURSor 1

解説 カーソル位置は項目番号で指定します。

:DISPlay[:NUMeric]:IMPedance:OBJect

機能 インピーダンス測定時における数値表示対象エレメントを設定/問い合わせします。

構文 :DISPlay[:NUMeric]:IMPedance:

OBJect {<NRf>}

:DISPlay[:NUMeric]:IMPedance:

OBJect?

<NRf> = 1 ~ 5(インピーダンス測定エレメント)

例 :DISPlay:NUMeric:IMPedance:OBJect 5

:DISPlay:NUMeric:IMPedance:OBJect?

-> :DISPlay:NUMeric:IMPedance:

OBJect 5

:DISPlay[:NUMeric]:IMPedance:TYPE

機能 インピーダンス測定時における数値表示の方式を設定/問い合わせします。

構文 :DISPlay[:NUMeric]:IMPedance:

TYPE {ZR_ZI|U_I|Z_PHI}

:DISPlay[:NUMeric]:IMPedance:TYPE?

例 :DISPlay:NUMeric:IMPedance:

TYPE ZR_ZI

:DISPlay:NUMeric:IMPedance:TYPE? ->

:DISPlay:NUMeric:IMPedance:

TYPE ZR_ZI

:DISPlay[:NUMeric]:NORMal?

機能 電力測定時における数値表示に関するすべての設定値を問い合わせます。

構文 :DISPlay[:NUMeric]:NORMal?
例 ・数値表示の方式(:DISPlay[:NUMeric]:NORMal:IAMount)が「<NRf>(分割表示)」のときの例
:DISPLAY:NUMERIC:NORMAL? ->
:DISPLAY:NUMERIC:NORMAL:
IAMOUNT 4;ITEM1 URMS,1;
ITEM2 UMN,1;ITEM3 UDC,1;...(中略)
...;ITEM100 NONE;ICURSOR 1
・数値表示の方式(:DISPlay[:NUMeric]:NORMal:IAMount)が「ALL(全表示)」のときの例
:DISPLAY:NUMERIC:NORMAL? ->
:DISPLAY:NUMERIC:NORMAL:
IAMOUNT ALL;FCURSOR URMS

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay[:NUMeric]:NORMal:FCURsor

機能 電力測定時における数値表示(全表示)のカーソル位置を設定/問い合わせします。

構文 :DISPlay[:NUMeric]:NORMal:FCURsor {<Function>}
:DISPlay[:NUMeric]:NORMal:FCURsor?
<Function> = {URMS | UMN | UDC | UAC | IRMS | ...}(ファンクション選択肢一覧(1)を参照)
例 :DISPLAY:NUMERIC:NORMAL:FCURSOR URMS
:DISPLAY:NUMERIC:NORMAL:FCURSOR? ->
:DISPLAY:NUMERIC:NORMAL:FCURSOR URMS

解説 ・カーソル位置はファンクションで指定します。
・数値表示の方式(:DISPlay[:NUMeric]:NORMal:IAMount)が「ALL(全表示)」のときに有効な設定です。
・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay[:NUMeric]:NORMal:IAMount

機能 電力測定時における数値表示の方式を設定/問い合わせします。

構文 :DISPlay[:NUMeric]:NORMal:IAMount {<NRf>|ALL}
:DISPlay[:NUMeric]:NORMal:IAMount?
<NRf> = 4, 8, 16, 42, 78

例 :DISPLAY:NUMERIC:NORMAL:IAMOUNT 4
:DISPLAY:NUMERIC:NORMAL:IAMOUNT? ->
:DISPLAY:NUMERIC:NORMAL:IAMOUNT 4

解説 ・表示される測定データの内容は、数値表示の方式の設定により、以下のようになります。
<NRf> = 数値表示項目を、項目番号順に表示します。(NRfは1つの画面に表示される項目数を表します)
ALL = 全電力測定ファンクションをエレメントごとに並べて表示します。
・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay[:NUMeric]:NORMal:ICURsor

機能 電力測定時における数値表示(分割表示)のカーソル位置を設定/問い合わせします。

構文 :DISPlay[:NUMeric]:NORMal:ICURsor {<NRf>}
:DISPlay[:NUMeric]:NORMal:ICURsor?
<NRf> = 1 ~ 100

例 :DISPLAY:NUMERIC:NORMAL:ICURSOR 1
:DISPLAY:NUMERIC:NORMAL:ICURSOR? ->
:DISPLAY:NUMERIC:NORMAL:ICURSOR 1

解説 ・カーソル位置は項目番号で指定します。
・数値表示の方式(:DISPlay[:NUMeric]:NORMal:IAMount)が「<NRf>(分割表示)」のときに有効な設定です。
・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay[:NUMERIC]:NORMAL:ITEM<x>

機能 電力測定時における数値表示項目を設定/問い合わせします。

構文 :DISPlay[:NUMERIC]:NORMAL:ITEM<x> {NONE|<Function>,<Element>}
:DISPlay[:NUMERIC]:NORMAL:ITEM<x>?
<x> = 1 ~ 100(項目番号)
NONE = 表示項目なし
<Function> = {URMS|UMN|UDC|UAC|IRMS|...}(ファンクション選択肢一覧(1)を参照)
<Element> = {<NRf>|SIGMA|SIGMB|SIGMC}
(<NRf> = 1 ~ 4(電力測定エレメント))

例 :DISPLAY:NUMERIC:NORMAL:
ITEM1 URMS,1
:DISPLAY:NUMERIC:NORMAL:ITEM1? ->
:DISPLAY:NUMERIC:NORMAL:
ITEM1 URMS,1

解説

- ・数値表示の方式(:DISPlay[:NUMERIC]:NORMAL:IAMount)が「<NRf>(分割表示)」のときに有効な設定です。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay[:NUMERIC]:NORMAL:PRESet

機能 電力測定時における数値表示項目を決められたパターンにプリセットします。

構文 :DISPlay[:NUMERIC]:NORMAL:PRESet {<NRf>}
<NRf> = 1 ~ 4(パターン番号)

例 :DISPLAY:NUMERIC:NORMAL:PRESET 1

解説

- ・<NRf>に1 ~ 4のどれを設定しても、数値表示項目の表示パターン(順序)は、本体画面に表示されるDisplay設定メニューのリセット実行(Reset List Exec)をしたときと同じ表示順序になります。リセット実行をしたときの表示順序については、本体ユーザーズマニュアルを参照してください。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREND?

機能 トレンドに関するすべての設定値を問い合わせます。

構文 :DISPlay:TREND?
例 :DISPLAY:TREND? -> :DISPLAY:TREND:
SAMPLING 1;T1 1;T2 1;T3 1;T4 1;
T5 1;T6 1;T7 1;T8 1;T9 0;T10 0;
T11 0;T12 0;T13 0;T14 0;T15 0;
T16 0;FORMAT SINGLE;TDIV 0,0,3;
NORMAL:ITEM1:FUNCTION URMS,1;
SCALING:MODE AUTO;VALUE 100.00E+00,
-100.00E+00;:DISPLAY:TREND:NORMAL:
ITEM2:FUNCTION IRMS,1;SCALING:
MODE AUTO;VALUE 100.00E+00,
-100.00E+00;...(中略)...;:DISPLAY:
TREND:NORMAL:ITEM16:FUNCTION FU,2;
SCALING:MODE AUTO;VALUE 100.00E+00,
-100.00E+00

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREND:ALL

機能 すべてのトレンドのON/OFFを一括設定します。

構文 :DISPlay:TREND:ALL {<Boolean>}

例 :DISPLAY:TREND:ALL ON

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREND:FORMat

機能 トレンドの表示フォーマットを設定/問い合わせします。

構文 :DISPlay:TREND:FORMat {SINGLE|DUAL|TRIad|QUAD}
:DISPlay:TREND:FORMat?

例 :DISPLAY:TREND:FORMAT SINGLE
:DISPLAY:TREND:FORMAT? ->

:DISPLAY:TREND:FORMAT SINGLE

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREND:NORMAL?

機能 電力測定時における全トレンドに関するすべての設定値を問い合わせます。

構文 :DISPlay:TREND:NORMAL?

例 :DISPLAY:TREND:NORMAL? ->
:DISPLAY:TREND:NORMAL:ITEM1:
FUNCTION URMS,1;SCALING:MODE AUTO;
VALUE 100.00E+00,-100.00E+00;;
DISPLAY:TREND:NORMAL:ITEM2:
FUNCTION IRMS,1;SCALING:MODE AUTO;
VALUE 100.00E+00,-100.00E+00;...
(中略)...;:DISPLAY:TREND:NORMAL:
ITEM16:FUNCTION FU,2;SCALING:
MODE AUTO;VALUE 100.00E+00,
-100.00E+00

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREND:NORMAL:ITEM<x>?

機能 電力測定時におけるトレンドに関するすべての設定値を問い合わせます。

構文 :DISPlay:TREND:NORMAL:ITEM<x>?

例 <x> = 1 ~ 16(項目番号)
:DISPLAY:TREND:NORMAL:ITEM1? ->
:DISPLAY:TREND:NORMAL:ITEM1:
FUNCTION URMS,1;SCALING:MODE AUTO;
VALUE 100.00E+00,-100.00E+00

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREND:NORMAL:ITEM<x>[:FUNCTION]

機能 電力測定時におけるトレンド項目を設定/問い合わせします。

構文 :DISPlay:TREND:NORMAL:ITEM<x>
[:FUNCTION] {NONE|<Function>,
<Element>}
:DISPlay:TREND:NORMAL:ITEM<x>:
FUNCTION?
<x> = 1 ~ 16(項目番号)
NONE = 表示項目なし
<Function> = {URMS|UMN|UDC|UAC|IRMS|...}(ファンクション選択肢一覧(1)を参照)
<Element> = {<NRf>|SIGMA|SIGMB|SIGMC}
(<NRf> = 1 ~ 4)(電力測定エレメント)

例 :DISPLAY:TREND:NORMAL:ITEM1:
FUNCTION URMS,1
:DISPLAY:TREND:NORMAL:ITEM1:
FUNCTION? -> :DISPLAY:TREND:NORMAL:
ITEM1:FUNCTION URMS,1

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREND:NORMAL:ITEM<x>:SCALing?

機能 電力測定時におけるトレンドのスケールングに関するすべての設定値を問い合わせます。

構文 :DISPlay:TREND:NORMAL:ITEM<x>:
SCALing?

例 <x> = 1 ~ 16(項目番号)
:DISPLAY:TREND:NORMAL:ITEM1:
SCALING? -> :DISPLAY:TREND:NORMAL:
ITEM1:SCALING:MODEAUTO;
VALUE 100.00E+00,-100.00E+00

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREND:NORMAL:ITEM<x>:SCALing:**MODE**

機能 電力測定時におけるトレンドのスケールング方式を設定/問い合わせします。

構文 :DISPlay:TREND:NORMAL:ITEM<x>:
SCALing:MODE {AUTO|MANual}
:DISPlay:TREND:NORMAL:ITEM<x>:
SCALing:MODE?
<x> = 1 ~ 16(項目番号)

例 :DISPLAY:TREND:NORMAL:ITEM1:
SCALING:MODE AUTO
:DISPLAY:TREND:NORMAL:ITEM1:
SCALING:MODE? -> :DISPLAY:TREND:
NORMAL:ITEM1:SCALING:MODE AUTO

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

5.4 DISPlayグループ

:DISPlay:TREnd:NORMal:ITEM<x>:SCALing:

VALue

機能	電力測定時におけるトレンドのマニュアルスケーリング上下限值を設定/問い合わせします。
構文	:DISPlay:TREnd:NORMal:ITEM<x>: SCALing:VALue {<NRf>,<NRf>} :DISPlay:TREnd:NORMal:ITEM<x>: SCALing:VALue?
例	<x> = 1 ~ 16(項目番号) <NRf> = - 9.9999E+30 ~ 9.9999E+30 :DISPlay:TREnd:NORMal:ITEM1: SCALing:VALue 100,-100 :DISPlay:TREnd:NORMal:ITEM1: SCALing:VALue? ->
解説	・ 上限値, 下限値の順で設定します。 ・ トレンドのスケーリング方式 (:DISPlay:TREnd:NORMal:ITEM<x>: SCALing:MODE)が「MANual」のときに有効な設定です。 ・ インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREnd:PDIV

機能	トレンドの横軸(Point/div)を設定/問い合わせします。
構文	:DISPlay:TREnd:PDIV {<NRf>} :DISPlay:TREnd:PDIV? <NRf> = 1, 2, 5, 10, 20, 50, 100, 200, 500
例	:DISPlay:TREnd:PDIV 50 :DISPlay:TREnd:PDIV? -> :DISPlay: TREnd:PDIV 50
解説	・ 電力測定時の波形サンプリング (:WSEtup[:SAMPling])がONのときに有効な設定です。 ・ インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREnd:REStart

機能	トレンドをリスタートします。
構文	:DISPlay:TREnd:REStart
例	:DISPlay:TREnd:REStart
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREnd[:SAMPling]

機能	トレンド波形サンプリングのON/OFFを設定/問い合わせします。
構文	:DISPlay:TREnd: [:SAMPling] {<Boolean>} :DISPlay:TREnd[:SAMPling]?
例	:DISPlay:TREnd:SAMPling ON :DISPlay:TREnd:SAMPling? -> :DISPlay:TREnd:SAMPling 1
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREnd:TDIV

機能	電力測定時におけるトレンドの横軸(T/div)を設定/問い合わせします。
構文	:DISPlay:TREnd:TDIV {<NRf>,<NRf>,<NRf>} :DISPlay:TREnd:TDIV? {<NRf>,<NRf>,<NRf>} = 0, 0, 3 ~ 24, 0, 0 1つ目の<NRf> = 1, 3, 6, 12, 24 (時間) 2つ目の<NRf> = 1, 3, 6, 10, 30 (分) 3つ目の<NRf> = 3, 6, 10, 30 (秒)
例	:DISPlay:TREnd:TDIV 0,0,3 :DISPlay:TREnd:TDIV? -> :DISPlay: TREnd:TDIV 0,0,3
解説	・ 3つの<NRf>のうち、1つが0以外となり、他の2つは0となるように設定します。 ・ 波形サンプリング(:WSEtup[:SAMPling])がOFFのときに有効な設定です。 ・ インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:TREnd:T<x>

機能	トレンドのON/OFFを設定/問い合わせします。
構文	:DISPlay:TREnd:T<x> {<Boolean>} :DISPlay:TREnd:T<x>? <x> = 1 ~ 16(項目番号)
例	:DISPlay:TREnd:T1 ON :DISPlay:TREnd:T1? -> :DISPlay:TREnd:T1 1
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:WAVE?

機能	波形表示に関するすべての設定値を問い合わせます。
構文	:DISPlay:WAVE?
例	:DISPlay:WAVE? -> :DISPlay:WAVE: U1 1;U2 1;U3 1;U4 1;I1 1;I2 1;I3 1; I4 1;FORMAT SINGLE; INTERPOLATE LINE;GRATICULE GRID; SVALUE 1;TLABEL 0;MAPPING:MODE AUTO
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:WAVE:ALL

機能	すべての波形表示のON/OFFを一括設定します。
構文	:DISPlay:WAVE:ALL {<Boolean>}
例	:DISPlay:WAVE:ALL ON
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:WAVE:FORMAT

機能	波形の表示フォーマットを設定/問い合わせします。
構文	:DISPlay:WAVE:FORMAt {SINGLe DUAL TRIad QUAD}
例	:DISPlay:WAVE:FORMAt? :DISPlay:WAVE:FORMAt SINGLE :DISPlay:WAVE:FORMAt? -> :DISPlay: WAVE:FORMAt SINGLE
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:WAVE:GRATicule

機能	グラティクル(目盛り)のタイプを設定/問い合わせします。
構文	:DISPlay:WAVE:GRATicule {GRID FRAMe CROSShair}
例	:DISPlay:WAVE:GRATicule? :DISPlay:WAVE:GRATicule? -> :DISPlay:WAVE:GRATicule GRID
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:WAVE:INTERpolate

機能	波形の補間方式を設定/問い合わせします。
構文	:DISPlay:WAVE:INTERpolate {OFF LINE}
例	:DISPlay:WAVE:INTERpolate? :DISPlay:WAVE:INTERPOLATE LINE :DISPlay:WAVE:INTERPOLATE? -> :DISPlay:WAVE:INTERPOLATE LINE
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:WAVE:MAPPING?

機能	分割フォーマットへの波形の割り付けに関するすべての設定値を問い合わせます。
構文	:DISPlay:WAVE:MAPPING?
例	:DISPlay:WAVE:MAPPING? -> :DISPlay: WAVE:MAPPING:MODE USER;U1 0;U2 1; U3 2;U4 3;I1 0;I2 1;I3 2;I4 3
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:WAVE:MAPPING[:MODE]

機能	分割フォーマットへの波形の割り付け方法を設定/問い合わせします。
構文	:DISPlay:WAVE:MAPPING[:MODE] {AUTO FIXed USER}
例	:DISPlay:WAVE:MAPPING:MODE? :DISPlay:WAVE:MAPPING:MODE? -> :DISPlay:WAVE:MAPPING:MODE AUTO
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:WAVE:MAPPING:{U<x>|I<x>}

機能	分割フォーマットへの{電圧 電流}波形の割り付けを設定/問い合わせします。
構文	:DISPlay:WAVE:MAPPING: {U<x> I<x>} {<Nrf>} :DISPlay:WAVE:MAPPING:{U<x> I<x>}? <x> = 1 ~ 4(電力測定エレメント) <Nrf> = 0 ~ 3
例	:DISPlay:WAVE:MAPPING:U1 0 :DISPlay:WAVE:MAPPING:U1? -> :DISPlay:WAVE:MAPPING:U1 0
解説	・ 波形の割り付け方法(:DISPlay:WAVE:MAPPING[:MODE])が「USER」のときに有効な設定です。 ・ インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:WAVE:SVALue (Scale VALue)

機能 スケール値表示のON/OFFを設定/問い合わせします。

構文 :DISPlay:WAVE:SVALue {<Boolean>}
:DISPlay:WAVE:SVALue?

例 :DISPlay:WAVE:SVALue ON
:DISPlay:WAVE:SVALue? -> :DISPlay:
WAVE:SVALue 1

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:WAVE:TLABel (Trace LABel)

機能 波形ラベル名表示のON/OFFを設定/問い合わせします。

構文 :DISPlay:WAVE:TLABel {<Boolean>}
:DISPlay:WAVE:TLABel?

例 :DISPlay:WAVE:TLABel OFF
:DISPlay:WAVE:TLABel? -> :DISPlay:
WAVE:TLABel 0

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:DISPlay:WAVE:{U<x>|I<x>}

機能 {電圧|電流}波形表示のON/OFFを設定/問い合わせします。

構文 :DISPlay:WAVE:
{U<x>|I<x>} {<Boolean>}
:DISPlay:WAVE:{U<x>|I<x>}?
<x> = 1 ~ 4(電力測定エレメント)

例 :DISPlay:WAVE:U1 ON
:DISPlay:WAVE:U1? -> :DISPlay:WAVE:
U1 1

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

ファンクション選択肢(<Function>)一覧**(1) 電力測定モードにおけるファンクション**

対象コマンド

```
:DISPlay[:NUMeric]:NORMal:FCURsor
:DISPlay[:NUMeric]:NORMal:ITEM<x>
:DISPlay:TREND:NORMal:ITEM<x>[:FUNCTION]
:FILE:SAVE:NUMeric:NORMal:...
:HCOpy:PRINter:DLISt:NORMal:...
:NUMeric[:NORMal]:ITEM<x>
:STORe:NUMeric:NORMal:...
```

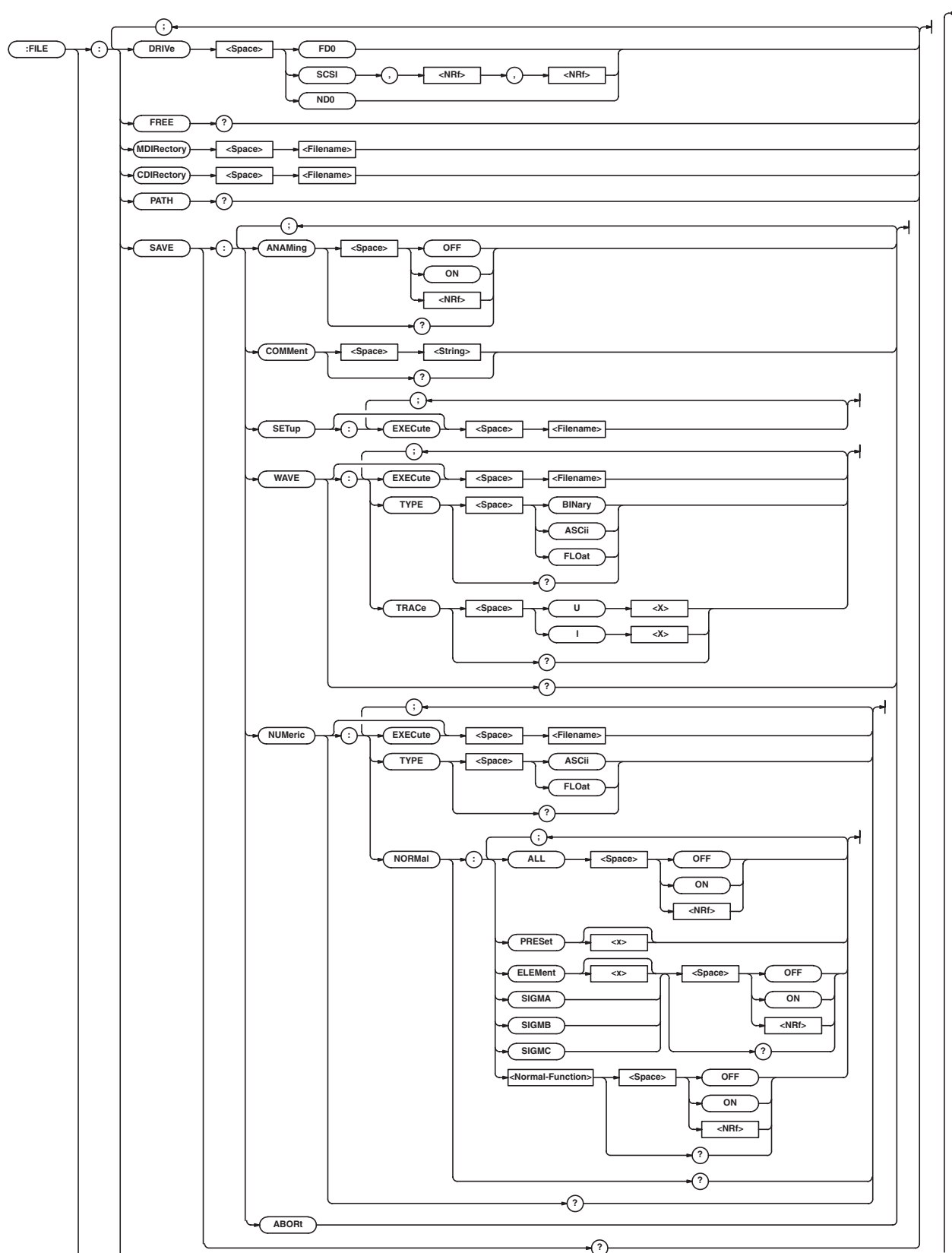
コマンドで使用する : メニューで使用している
ファンクション名 ファンクション名
(数値表示ヘッダ名)

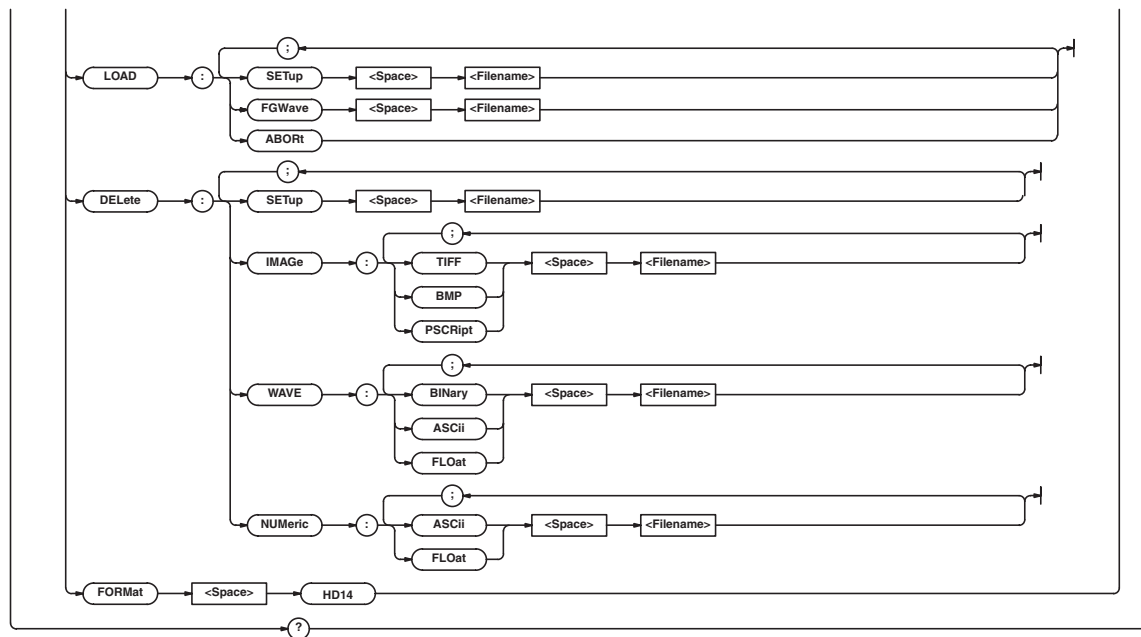
```
URMS : Urms
UMN : Umean
UDC : Udc
UAC : Uac
IRMS : Irms
```

```
IMN : lmean
IDC : ldc
IAC : lac
P : P
S : S
Q : Q
LAMBda : λ
PHI : φ
FU : FreqU (fU)
FI : FreqI (fI)
UPPeak : U+peak (U+pk)
UMPeak : U-peak (U-pk)
IPPeak : I+peak (I+pk)
IMPeak : I-peak (I-pk)
CFU : CfU
CFI : Cfl
FFU : FfU
FFI : Ffl
Z : Z
RS : Rs
XS : Xs
RP : Rp
XP : Xp
PC : Pc
TIME : I-Time
WH : Wp
WHP : Wp+
WHM : Wp-
AH : q
AHP : q+
AHM : q-
ETA : η
SETA : 1/η
F1 : F1
F2 : F2
F3 : F3
F4 : F4
DURMS : ΔUrms
DUMN : ΔUmean
DUDC : ΔUdc
DUAC : ΔUac
DIRMS : ΔIrms
DIMN : Δlmean
DIDC : Δldc
DIAC : Δlac
```

5.5 FILEグループ

FILEグループは、ファイル操作に関するグループです。
フロントパネルのFILEと同じ設定、および設定内容の問い合わせができます。



**:FILE?**

機能 ファイル操作に関するすべての設定値を問い合わせます。

構文 :FILE?

例 :FILE? -> (「:FILE:SAVE?」の応答と同じ)

:FILE:CDIRectory

機能 カレントディレクトリを変更します。

構文 :FILE:CDIRectory {<Filename>}

<Filename> = ディレクトリ名

例 :FILE:CDIRECTORY "IMAGE"

解説 上のディレクトリに移動するには, ".."を指定します。

:FILE:DElete:IMAGe:{TIFF|BMP|PSCript}

機能 画面イメージデータファイルを削除します。

構文 :FILE:DElete:IMAGe:{TIFF|BMP|PSCript} {<Filename>}

例 :FILE:DELETE:IMAG:TIFF "IMAGE1"

解説 ファイル名は, 拡張子を付けずに指定してください。

:FILE:DElete:NUMeric:{ASCii|FLOat}

機能 数値データファイルを削除します。

構文 :FILE:DElete:NUMeric:{ASCii|FLOat} {<Filename>}

例 :FILE:DELETE:NUMERIC:ASCII "NUM1"

解説 ファイル名は, 拡張子を付けずに指定してください。

:FILE:DElete:SETup

機能 設定情報ファイルを削除します。

構文 :FILE:DElete:SETup {<Filename>}

例 :FILE:DELETE:SETUP "SETUP1"

解説 ファイル名は, 拡張子を付けずに指定してください。

:FILE:DElete:WAVE:{BINary|ASCii|FLOat}

機能 波形表示データファイルを削除します。

構文 :FILE:DElete:WAVE:{BINary|ASCii|FLOat} {<Filename>}

例 :FILE:DELETE:WAVE:BINARY "WAVE1"

解説 ファイル名は, 拡張子を付けずに指定してください。

:FILE:DRIVE

機能 対象ドライブを設定します。

構文 :FILE:DRIVE {FD0|SCSI,<Nrf>[,<Nrf>]|ND0}
FD0 = フロッピーディスクドライブ
SCSI = SCSIデバイス
最初の<Nrf> = SCSIアドレス(0~7)
2番目の<Nrf> = パーティション(0~9)

ND0 = ネットワークドライブ

例 :FILE:DRIVE FD0

解説 パーティションを設定していないドライブの場合, 2番目の<Nrf>は省略してください。

:FILE:FORMat

機能 フロッピーディスクのフォーマットを実行します。

構文 :FILE:FORMat {HD14}

例 :FILE:FORMAT HD14

:FILE:FREE?

機能 対象ドライブの空き容量(byte)を問い合わせます。
構文 :FILE:FREE?
例 :FILE:FREE? -> 163840

:FILE:LOAD:ABORt

機能 ファイルのロードを中止します。
構文 :FILE:LOAD:ABORt
例 :FILE:LOAD:ABORt

:FILE:LOAD:FGWave

機能 インピーダンス測定用負荷電流のパターン波形ファイルをロードします。
構文 :FILE:LOAD:FGWave {<Filename>}
例 :FILE:LOAD:FGWAVE "FGWAVE1"
解説 ファイル名は、拡張子を付けずに指定してください。

:FILE:LOAD:SETup

機能 設定情報ファイルをロードします。
構文 :FILE:LOAD:SETup {<Filename>}
例 :FILE:LOAD:SETUP "SETUP1"
解説
・ ファイル名は、拡張子を付けずに指定してください。
・ このコマンドはオーバーラップコマンドです。

:FILE:MDIRectory

機能 ディレクトリを作成します。
構文 :FILE:MDIRectory {<Filename>}
<Filename> = ディレクトリ名
例 :FILE:MDIRECTORY "TEST"

:FILE:PATH?

機能 カレントディレクトリを絶対パスで問い合わせます。
構文 :FILE:PATH?
例 :FILE:PATH? -> "FD0¥IMAGE"

:FILE:SAVE?

機能 ファイルの保存に関するすべての設定値を問い合わせます。
構文 :FILE:SAVE?
例 :FILE:SAVE? -> :FILE:SAVE:
ANAMING 1;COMMENT "";WAVE:
TYPEBINARY;:FILE:SAVE:NUMERIC:
TYPE ASCII;NORMAL:ELEMENT1 1;
ELEMENT2 0;ELEMENT3 0;ELEMENT4 0;
SIGMA 0;SIGMB 0;SIGMC 0;URMS 1;
UMN 1;UDC 1;UAC 1;IRMS 1;IMN 1;
IDC 1;IAC 1;P 1;S 1;Q 1;LAMBDA 1;
PHI 1;FU 1;FI 1;UPPEAK 1;UMPEAK 1;
IPPEAK 1;IMPEAK 1;CFU 1;CFI 1;
FFU 1;FFI 1;Z 1;RS 1;XS 1;RP 1;
XP 1;PC 1;TIME 0;WH 0;WHP 0;WHM 0;
AH 0;AHP 0;AHM 0;ETA 0;SETA 0;F1 0;
F2 0;F3 0;F4 0;DURMS 0;DUMN 0;
DUDC 0;DUAC 0;DIRMS 0;DIMN 0;
DIDC 0;DIAC 0

:FILE:SAVE:ABORt

機能 ファイルの保存を中止します。
構文 :FILE:SAVE:ABORt
例 :FILE:SAVE:ABORt

:FILE:SAVE:ANAMing

機能 保存するファイル名を自動的につけるかどうかを設定/問い合わせします。
構文 :FILE:SAVE:ANAMing {<Boolean>}
:FILE:SAVE:ANAMing?
例 :FILE:SAVE:ANAMINGON
:FILE:SAVE:ANAMING? -> :FILE:SAVE:
ANAMING 1

:FILE:SAVE:COMMeNt

機能 保存するファイルに付加するコメントを設定/問い合わせします。
構文 :FILE:SAVE:COMMeNt {<文字列>}
:FILE:SAVE:COMMeNt?
<文字列> = 30文字以内
例 :FILE:SAVE:COMMENT "CASE1"
:FILE:SAVE:COMMENT? -> :FILE:SAVE:
COMMENT "CASE1"

:FILE:SAVE:NUMERIC?

機能 数値データのファイル保存に関するすべての設定値を問い合わせます。

構文 :FILE:SAVE:NUMERIC?

例 :FILE:SAVE:NUMERIC? ->

```
:FILE:SAVE:NUMERIC:TYPE ASCII;
NORMAL:ELEMENT1 1;ELEMENT2 0;
ELEMENT3 0;ELEMENT4 0;SIGMA 0;
SIGMB 0;SIGMC 0;URMS 1;UMN 1;UDC 1;
UAC 1;IRMS 1;IMN 1;IDC 1;IAC 1;P 1;
S 1;Q 1;LAMBDA 1;PHI 1;FU 1;FI 1;
UPPEAK 1;UMPEAK 1;IPPEAK 1;
IMPEAK 1;CFU 1;CFI 1;FFU 1;FFI 1;
Z 1;RS 1;XS 1;RP 1;XP 1;PC 1;
TIME 0;WH 0;WHP 0;WHM 0;AH 0;AHP 0;
AHM 0;ETA 0;SETA 0;F1 0;F2 0;F3 0;
F4 0;DURMS 0;DUMN 0;DUDC 0;DUAC 0;
DIRMS 0;DIMN 0;DIDC 0;DIAC 0
```

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:FILE:SAVE:NUMERIC[:EXECUTE]

機能 数値データをファイルに保存します。

構文 :FILE:SAVE:NUMERIC

```
[:EXECUTE] {<Filename>}
```

例 :FILE:SAVE:NUMERIC:EXECUTE "NUM1"

解説

- ・ファイル名は、拡張子を付けずに指定してください。
- ・このコマンドはオーバーラップコマンドです。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:FILE:SAVE:NUMERIC:NORMAL?

機能 電力測定時における数値データのファイル保存に関するすべての設定値を問い合わせます。

構文 :FILE:SAVE:NUMERIC:NORMAL?

例 :FILE:SAVE:NUMERIC:NORMAL? ->

```
:FILE:SAVE:NUMERIC:NORMAL:
ELEMENT1 1;ELEMENT2 0;ELEMENT3 0;
ELEMENT4 0;SIGMA 0;SIGMB 0;SIGMC 0;
URMS 1;UMN 1;UDC 1;UAC 1;IRMS 1;
IMN 1;IDC 1;IAC 1;P 1;S 1;Q 1;
LAMBDA 1;PHI 1;FU 1;FI 1;UPPEAK 1;
UMPEAK 1;IPPEAK 1;IMPEAK 1;CFU 1;
CFI 1;FFU 1;FFI 1;Z 1;RS 1;XS 1;
RP 1;XP 1;PC 1;TIME 0;WH 0;WHP 0;
WHM 0;AH 0;AHP 0;AHM 0;ETA 0;
SETA 0;F1 0;F2 0;F3 0;F4 0;DURMS 0;
DUMN 0;DUDC 0;DUAC 0;DIRMS 0;
DIMN 0;DIDC 0;DIAC 0
```

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:FILE:SAVE:NUMERIC:NORMAL:ALL

機能 電力測定時に数値データをファイルに保存するときのすべての電力測定エレメント・ファンクションの出力ON/OFFを一括設定します。

構文 :FILE:SAVE:NUMERIC:NORMAL:

```
ALL {<Boolean>}
```

例 :FILE:SAVE:NUMERIC:NORMAL:ALL ON

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:FILE:SAVE:NUMERIC:NORMAL:{ELEMENT<x>|SIGMA|SIGMB|SIGMC}

機能 電力測定時に数値データをファイルに保存するときの{各電力測定エレメント|ΣA|ΣB|ΣC}の出力ON/OFFを設定/問い合わせします。

構文 :FILE:SAVE:NUMERIC:NORMAL:

```
{ELEMENT<x>|SIGMA|SIGMB|
```

```
SIGMC} {<Boolean>}
```

```
:FILE:SAVE:NUMERIC:NORMAL:
```

```
{ELEMENT<x>|SIGMA|SIGMB|SIGMC}?
```

```
<x> = 1 ~ 4(電力測定エレメント)
```

例 :FILE:SAVE:NUMERIC:NORMAL:

```
ELEMENT1 ON
```

```
:FILE:SAVE:NUMERIC:NORMAL:
```

```
ELEMENT1? -> :FILE:SAVE:NUMERIC:
```

```
NORMAL:ELEMENT1 1
```

解説

- ・「:FILE:SAVE:NUMERIC:NORMAL:SIGMB」の設定/問い合わせは、電力測定エレメントが2つ以上のモデルで有効です。
- ・「:FILE:SAVE:NUMERIC:NORMAL:SIGMC」の設定/問い合わせは、電力測定エレメントが3つ以上のモデルで有効です。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:FILE:SAVE:NUMERIC:NORMAL:PRESET<x>

機能 電力測定時に数値データをファイルに保存するときの電力測定エレメント・ファンクションの出力ON/OFFを決められたパターンにプリセットします。

構文 :FILE:SAVE:NUMERIC:NORMAL:PRESET<x>

```
<x> = 1 ~ 2(プリセットパターン番号)
```

例 :FILE:SAVE:NUMERIC:NORMAL:PRESET1

解説

- ・プリセット実行をしたときの出力設定パターンについては、本体ユーザーズマニュアルを参照してください。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:FILE:SAVE:NUMERIC:NORMAL:<電力測定ファンクション>

機能	電力測定時に数値データをファイルに保存するときの各ファンクションの出力ON/OFFを設定/問い合わせします。
構文	:FILE:SAVE:NUMERIC:NORMAL:<電力測定ファンクション> {<Boolean>} :FILE:SAVE:NUMERIC:NORMAL:<電力測定ファンクション>? <電力測定ファンクション> = {URMS UMN UDC UAC IRMS ...}(DISPLAYグループのファンクション選択肢一覧(1)を参照)
例	:FILE:SAVE:NUMERIC:NORMAL:URMS ON :FILE:SAVE:NUMERIC:NORMAL:URMS? -> :FILE:SAVE:NUMERIC:NORMAL:URMS 1
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:FILE:SAVE:NUMERIC:TYPE

機能	保存する数値データの形式を設定/問い合わせします。
構文	:FILE:SAVE:NUMERIC:TYPE {ASCIi FLOat} :FILE:SAVE:NUMERIC:TYPE? :FILE:SAVE:NUMERIC:TYPE ASCII :FILE:SAVE:NUMERIC:TYPE?-> :FILE:SAVE:NUMERIC:TYPE ASCII
例	:FILE:SAVE:NUMERIC:TYPE ASCII
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:FILE:SAVE:SETUP[:EXECute]

機能	設定情報をファイルに保存します。
構文	:FILE:SAVE:SETUP [:EXECute] {<Filename>}
例	:FILE:SAVE:SETUP:EXECUTE "SETUP1"
解説	・ ファイル名は、拡張子を付けずに指定してください。 ・ このコマンドはオーバーラップコマンドです。

:FILE:SAVE:WAVE?

機能	波形表示データのファイル保存に関するすべての設定値を問い合わせます。
構文	:FILE:SAVE:WAVE?
例	:FILE:SAVE:WAVE? -> :FILE:SAVE:WAVE:TYPE BINARY
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:FILE:SAVE:WAVE[:EXECute]

機能	波形表示データをファイルに保存します。
構文	:FILE:SAVE:WAVE [:EXECute] {<Filename>}
例	:FILE:SAVE:WAVE:EXECUTE "WAVE1"
解説	・ ファイル名は、拡張子を付けずに指定してください。 ・ このコマンドはオーバーラップコマンドです。 ・ インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:FILE:SAVE:WAVE:TRACE

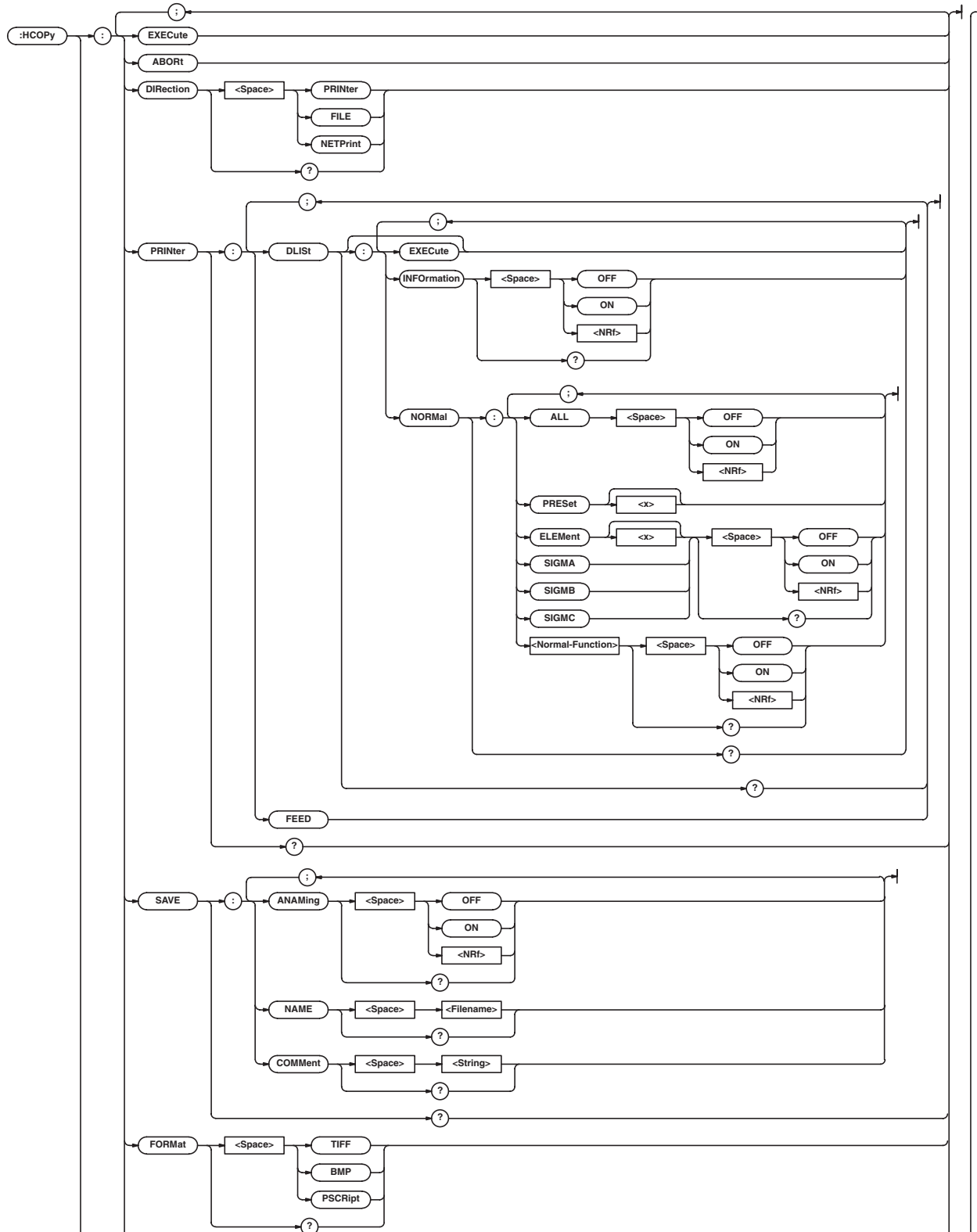
機能	ファイル保存の対象とする波形を設定/問い合わせします。
構文	:FILE:SAVE:WAVE:TRACE {U<x> I<x>} :FILE:SAVE:WAVE:TRACE? <x> = 1 ~ 4(電力測定エレメント)
例	:FILE:SAVE:WAVE:TRACE U1 :FILE:SAVE:WAVE:TRACE? -> :FILE:SAVE:WAVE:TRACE U1
解説	・ 保存する波形表示データの形式(:FILE:SAVE:WAVE:TYPE)が「FLOat」のときに有効な設定です。{BINARY ASCIi}のときは、表示がONになっている波形がすべて対象となります。 ・ インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

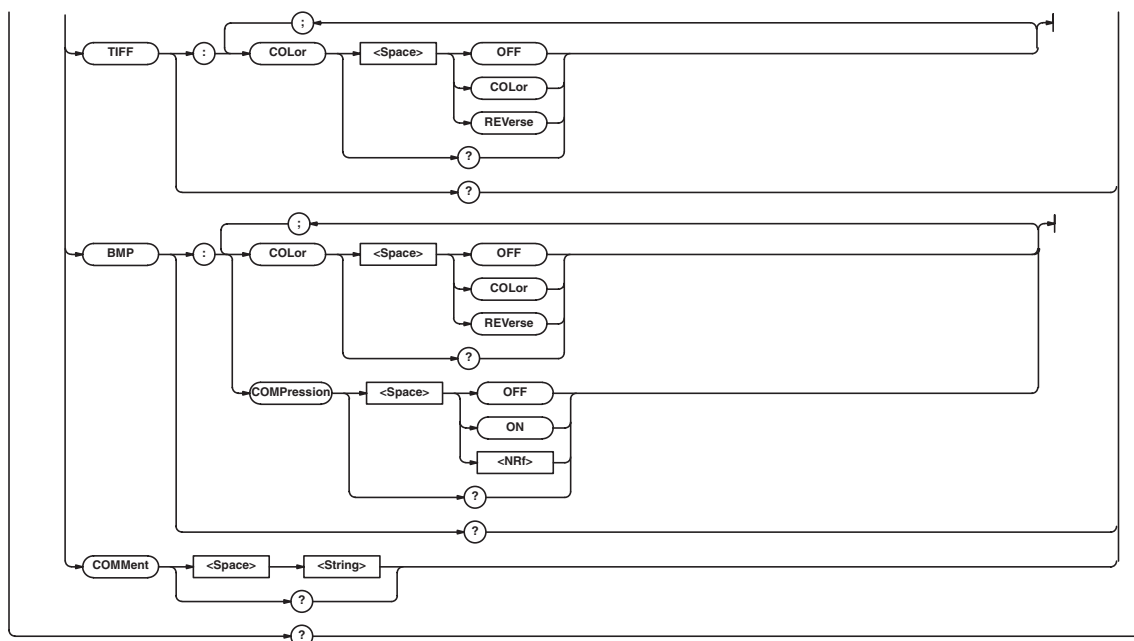
:FILE:SAVE:WAVE:TYPE

機能	保存する波形表示データの形式を設定/問い合わせします。
構文	:FILE:SAVE:WAVE:TYPE {BINARY ASCIi FLOat} :FILE:SAVE:WAVE:TYPE? :FILE:SAVE:WAVE:TYPE BINARY :FILE:SAVE:WAVE:TYPE?-> :FILE:SAVE:WAVE:TYPE BINARY
例	:FILE:SAVE:WAVE:TYPE BINARY
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

5.6 HCOPyグループ

HCOPYグループは、内蔵プリンタ(オプション)などへの画面データの出力に関するグループです。
フロントパネルのCOPY, MENU(SHIFT+COPY)と同じ設定、および設定内容の問い合わせができます。



**:HCOpy?**

機能 画面イメージデータの出力に関するすべての設定値を問い合わせます。

構文 :HCOpy?

例 :HCOpy? -> :HCOpy:

```
DIRECTION PRINTER;PRINTER:DLIST:
INFORMATION 1;NORMAL:ELEMENT1 1;
ELEMENT2 0;ELEMENT3 0;ELEMENT4 0;
SIGMA 0;SIGMB 0;SIGMC 0;URMS 1;
UMN 1;UDC 1;UAC 1;IRMS 1;IMN 1;
IDC 1;IAC 1;P 1;S 1;Q 1;LAMBDA 1;
PHI 1;FU 1;FI 1;UPPEAK 1;UMPEAK 1;
IPPEAK 1;IMPEAK 1;CFU 1;CFI 1;
FFU 1;FFI 1;Z 1;RS 1;XS 1;RP 1;
XP 1;PC 1;TIME 0;WH 0;WHP 0;WHM 0;
AH 0;AHP 0;AHM 0;ETA 0;SETA 0;F1 0;
F2 0;F3 0;F4 0;DURMS 0;DUMN 0;
DUDC 0;DUAC 0;DIRMS 0;DIMN 0;
DIDC 0;DIAC 0;:HCOpy:
COMMENT "THIS IS TEST."
```

:HCOpy:ABORT

機能 画面イメージデータ出力，紙送りを中止します。

構文 :HCOpy:ABORT

例 :HCOpy:ABORT

:HCOpy:BMP?

機能 BMPフォーマットに関するすべての設定値を問い合わせます。

構文 :HCOpy:BMP?

例 :HCOpy:BMP? -> :HCOpy:BMP:
COLOR COLOR;COMPRESSION 0

:HCOpy:BMP:COLOR

機能 BMPフォーマットの色調を設定/問い合わせします。

構文 :HCOpy:BMP:COLOR {OFF|COLOR|
REVerse}

例 :HCOpy:BMP:COLOR?

:HCOpy:BMP:COLOR COLOR
:HCOpy:BMP:COLOR? -> :HCOpy:BMP:COLOR
COLOR

:HCOpy:BMP:COMPRESSION

機能 BMPフォーマットのデータを圧縮するかどうかを設定/問い合わせします。

構文 :HCOpy:BMP:COMPRESSION {<Boolean>}
:HCOpy:BMP:COMPRESSION?

例 :HCOpy:BMP:COMPRESSION OFF

:HCOpy:BMP:COMPRESSION? -> :HCOpy:
BMP:COMPRESSION 0

解説 色調(:HCOpy:BMP:COLOR)が{COLOR|
REVerse}の場合に有効な設定です。

:HCOpy:COMMENT

機能 画面下部に表示するコメントを設定/問い合わせします。

構文 :HCOpy:COMMENT {<文字列>}
:HCOpy:COMMENT?

<文字列> = 25文字以内(ただし，表示は20文字
までです。)

例 :HCOpy:COMMENT "THIS IS TEST."

:HCOpy:COMMENT? -> :HCOpy:
COMMENT "THIS IS TEST."

解説 本体画面に表示されるキーボード以外の文字や
記号は使用できません。

:HCOpy:DIRection

機能	画面イメージデータの出力先を設定/問い合わせします。
構文	:HCOpy:DIRection {PRINter FILE NETPrint} :HCOpy:DIRection?
例	:HCOpy:DIRECTION PRINTER :HCOpy:DIRECTION? -> :HCOpy:DIRECTION PRINTER
解説	・ {PRINter}は、内蔵プリンタ(オプション, /B5)搭載時のみ有効です。 ・ {NETPrint}は、イーサネット(オプション, /C10)搭載時のみ有効です。

:HCOpy:EXECute

機能	画面イメージデータ出力を実行します。
構文	:HCOpy:EXECute
例	:HCOpy:EXECUTE
解説	このコマンドはオーバーラップコマンドです。

:HCOpy:FORMat

機能	保存する画面イメージデータのファイル形式を設定/問い合わせします。
構文	:HCOpy:FORMat {TIFF BMP PSCript} :HCOpy:FORMat?
例	:HCOpy:FORMAT TIFF :HCOpy:FORMAT? -> :HCOpy:FORMAT TIFF
解説	データ出力先(:HCOpy:DIRection)が「PRINter」の場合は、無意味な設定になります。

:HCOpy:PRINter?

機能	内蔵プリンタでの印字に関するすべての設定値を問い合わせます。
構文	:HCOpy:PRINter?
例	:HCOpy:PRINTER? -> (「:HCOpy:PRINter:DLIST?」の応答と同じ)

:HCOpy:PRINter:DLIST?

機能	数値データリストの内蔵プリンタでの印字に関するすべての設定値を問い合わせます。
構文	:HCOpy:PRINter:DLIST?
例	:HCOpy:PRINTER:DLIST? -> :HCOpy:PRINTER:DLIST:INFORMATION 1; NORMAL:ELEMENT1 1;ELEMENT2 0; ELEMENT3 0;ELEMENT4 0;SIGMA 0; SIGMB 0;SIGMC 0;URMS 1;UMN 1;UDC 1; UAC 1;IRMS 1;IMN 1;IDC 1;IAC 1;P 1; S 1;Q 1;LAMBDA 1;PHI 1;FU 1;FI 1; UPPEAK 1;UMPEAK 1;IPPEAK 1; IMPEAK 1;CFU 1;CFI 1;FFU 1;FFI 1; Z 1;RS 1;XS 1;RP 1;XP 1;PC 1; TIME 0;WH 0;WHP 0;WHM 0;AH 0;AHP 0; AHM 0;ETA 0;SETA 0;F1 0;F2 0;F3 0; F4 0;DURMS 0;DUMN 0;DUDC 0;DUAC 0; DIRMS 0;DIMN 0;DIDC 0;DIAC 0

:HCOpy:PRINter:DLIST[:EXECute]

機能	数値データリストを内蔵プリンタで印字します。
構文	:HCOpy:PRINter:DLIST[:EXECute]
例	:HCOpy:PRINTER:DLIST:EXECUTE
解説	このコマンドはオーバーラップコマンドです。

:HCOpy:PRINter:DLIST:INFORMatIon

機能	数値データリストを内蔵プリンタで印字するときに設定情報を付加する/しないを設定/問い合わせします。
構文	:HCOpy:PRINter:DLIST: INFORMatIon {<Boolean>}
例	:HCOpy:PRINTER:DLIST:INFORMATION? :HCOpy:PRINTER:DLIST:INFORMATION ON -> :HCOpy:PRINTER:DLIST: INFORMATION 1

:HCOpy:PRINter:DLIST:NORMAL?

機能	電力測定時における数値データリストの印字に関するすべての設定値を問い合わせます。
構文	:HCOpy:PRINter:DLIST:NORMAL?
例	:HCOpy:PRINTER:DLIST:NORMAL? -> :HCOpy:PRINTER:DLIST:NORMAL: ELEMENT1 1;ELEMENT2 0;ELEMENT3 0; ELEMENT4 0;SIGMA 0;SIGMB 0;SIGMC 0; URMS 1;UMN 1;UDC 1;UAC 1;IRMS 1; IMN 1;IDC 1;IAC 1;P 1;S 1;Q 1; LAMBDA 1;PHI 1;FU 1;FI 1;UPPEAK 1; UMPEAK 1;IPPEAK 1;IMPEAK 1;CFU 1; CFI 1;FFU 1;FFI 1;Z 1;RS 1;XS 1; RP 1;XP 1;PC 1;TIME 0;WH 0;WHP 0; WHM 0;AH 0;AHP 0;AHM 0;ETA 0; SETA 0;F1 0;F2 0;F3 0;F4 0;DURMS 0; DUMN 0;DUDC 0;DUAC 0;DIRMS 0; DIMN 0;DIDC 0;DIAC 0

:HCOpy:PRINter:DLISt:NORMal:ALL

機能 電力測定時に数値データリストを内蔵プリンタで印字するときのすべての電力測定エレメント・ファンクションの出力ON/OFFを一括設定します。

構文 :HCOpy:PRINter:DLISt:NORMal:
ALL {<Boolean>}

例 :HCOpy:PRINter:DLISt:NORMal:ALL ON

:HCOpy:PRINter:DLISt:NORMal:{ELEMeNt<x>|SIGMA|SIGMB|SIGMC}

機能 電力測定時に数値データリストを内蔵プリンタで印字するときの{各電力測定エレメント|ΣA|ΣB|ΣC}の出力ON/OFFを設定/問い合わせします。

構文 :HCOpy:PRINter:DLISt:NORMal:
{ELEMeNt<x>|SIGMA|SIGMB|
SIGMC} {<Boolean>}
:HCOpy:PRINter:DLISt:NORMal:
{ELEMeNt<x>|SIGMA|SIGMB|SIGMC}?
<x> = 1 ~ 4(電力測定エレメント)

例 :HCOpy:PRINter:DLISt:NORMal:
ELEMENT1 ON
:HCOpy:PRINter:DLISt:NORMal:
ELEMENT1? -> :HCOpy:PRINter:DLISt:
NORMAL:ELEMENT1 1

解説

- ・「:HCOpy:PRINter:DLISt:NORMal:SIGMB」の設定/問い合わせは、電力測定エレメントが2つ以上のモデルで有効です。
- ・「:HCOpy:PRINter:DLISt:NORMal:SIGMC」の設定/問い合わせは、電力測定エレメントが3つ以上のモデルで有効です。

:HCOpy:PRINter:DLISt:NORMal:PRESet<x>

機能 電力測定時に数値データリストを内蔵プリンタで印字するときの電力測定エレメント・ファンクションの出力ON/OFFを決められたパターンにプリセットします。

構文 :HCOpy:PRINter:DLISt:NORMal:
PRESet<x>
<x> = 1 ~ 2(プリセットパターン番号)

例 :HCOpy:PRINter:DLISt:NORMal:PRESet1
プリセット実行をしたときの印字パターンについては、本体ユーザーズマニュアルを参照してください。

:HCOpy:PRINter:DLISt:NORMal:<電力測定ファンクション>

機能 電力測定時に数値データリストを内蔵プリンタで印字するときの各ファンクションの出力ON/OFFを設定/問い合わせします。

構文 :HCOpy:PRINter:DLISt:NORMal:<電力測定
ファンクション> {<Boolean>}
:HCOpy:PRINter:DLISt:NORMal:<電力測定
ファンクション>?
<電力測定ファンクション> = {URMS|UMN|
UDC|UAC|IRMS|...}(DISPlayグループのファン
クション選択肢一覧(1)を参照)

例 :HCOpy:PRINter:DLISt:NORMal:URMS ON
:HCOpy:PRINter:DLISt:NORMal:URMS?
-> :HCOpy:PRINter:DLISt:NORMal:
URMS 1

:HCOpy:PRINter:FEED

機能 内蔵プリンタの紙送りを実行します。

構文 :HCOpy:PRINter:FEED

例 :HCOpy:PRINter:FEED

解説 このコマンドはオーバーラップコマンドです。

:HCOpy:SAVE?

機能 ファイル保存に関するすべての設定値を問い合わせます。

構文 :HCOpy:SAVE?

例 :HCOpy:SAVE? -> :HCOpy:SAVE:
ANAMING 1;NAME "DATA1";
COMMENT "CASE1"

:HCOpy:SAVE:ANAMing

機能 保存するファイル名を自動的につけるかどうかを設定/問い合わせします。

構文 :HCOpy:SAVE:ANAMing {<Boolean>}
:HCOpy:SAVE:ANAMing?

例 :HCOpy:SAVE:ANAMING ON
:HCOpy:SAVE:ANAMING? -> :HCOpy:
SAVE:ANAMING 1

:HCOpy:SAVE:COMMeNt

機能 保存するファイルに付加するコメントを設定/問い合わせします。

構文 :HCOpy:SAVE:COMMeNt {<文字列>}
:HCOpy:SAVE:COMMeNt?
<文字列> = 25文字以内

例 :HCOpy:SAVE:COMMENT "CASE1"
:HCOpy:SAVE:COMMENT? ->
:HCOpy:SAVE:COMMENT "CASE1"

解説

- ・本体画面に表示されるキーボード以外の文字や記号は使用できません。
- ・データの出力先(:HCOpy:DIRection)が「FILE」のときに有効な設定です。

:HCOpy:SAVE:NAME

機能 保存するファイル名を設定/問い合わせします。

構文 :HCOpy:SAVE:NAME {<Filename>}

例 :HCOpy:SAVE:NAME "DATA1"

解説

- 画面データの保存先は、
 - 対象ドライブは「:FILE:DRIVE」コマンド
 - ディレクトリは「:FILE:CDIRECTORY」コマンドでそれぞれ設定します。保存先のパスは「:FILE:PATH?」で問い合わせできます。
 - ファイル名は、拡張子を付けずに指定してください。

:HCOpy:TIFf?

機能 TIFFフォーマットに関するすべての設定値を問い合わせます。

構文 :HCOpy:TIFf?

例 :HCOpy:TIFf? -> :HCOpy:TIFf:COLOR COLOR

:HCOpy:TIFf:COLOr

機能 TIFFフォーマットの色調を設定/問い合わせします。

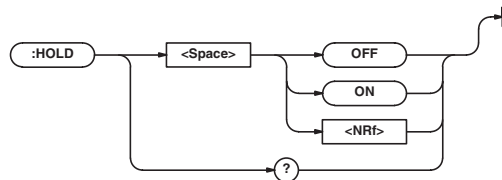
構文 :HCOpy:TIFf:COLOr {OFF|COLOr|REVERSE}

例 :HCOpy:TIFf:COLOr COLOR

:HCOpy:TIFf:COLOr? -> :HCOpy:TIFf:COLOR COLOR

5.7 HOLDグループ

HOLDグループは、出力データのホールド機能に関するグループです。フロントパネルのHOLDと同じ設定、および設定内容の問い合わせができます。

**:HOLD**

機能 出力データ(表示・通信など)のホールドを設定/問い合わせします。

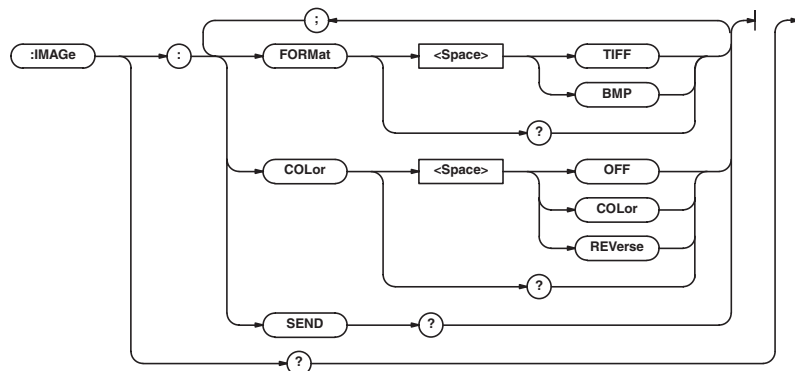
構文 :HOLD {<Boolean>}

例 :HOLD OFF

:HOLD? -> :HOLD 0

5.8 IMAGEグループ

IMAGEグループは、画面イメージデータの出力に関するグループです。
このグループに相当するフロントパネルのキーはありません。



: IMAGE?

機能 画面イメージデータの出力のすべての設定値を問い合わせます。

構文 : IMAGE?

例 : IMAGE? -> : IMAGE: FORMAt TIFF;
COLOR OFF

: IMAGE: COLor

機能 出力する画面イメージデータの色調を設定/問い合わせします。

構文 : IMAGE: COLor {OFF | COLor | REVerse}

: IMAGE: COLor?

例 : IMAGE: COLOR OFF

: IMAGE: COLOR? -> : IMAGE: COLOR OFF

: IMAGE: FORMat

機能 出力する画面イメージデータの出力形式を設定/問い合わせします。

構文 : IMAGE: FORMat {TIFF | BMP}

: IMAGE: FORMat?

例 : IMAGE: FORMAt TIFF

: IMAGE: FORMAt? -> : IMAGE:
FORMAt TIFF

: IMAGE: SEND?

機能 画面イメージデータを問い合わせます。

構文 : IMAGE: SEND?

例 : IMAGE: SEND? -> #6(6桁のバイト数)(データバイトの並び)

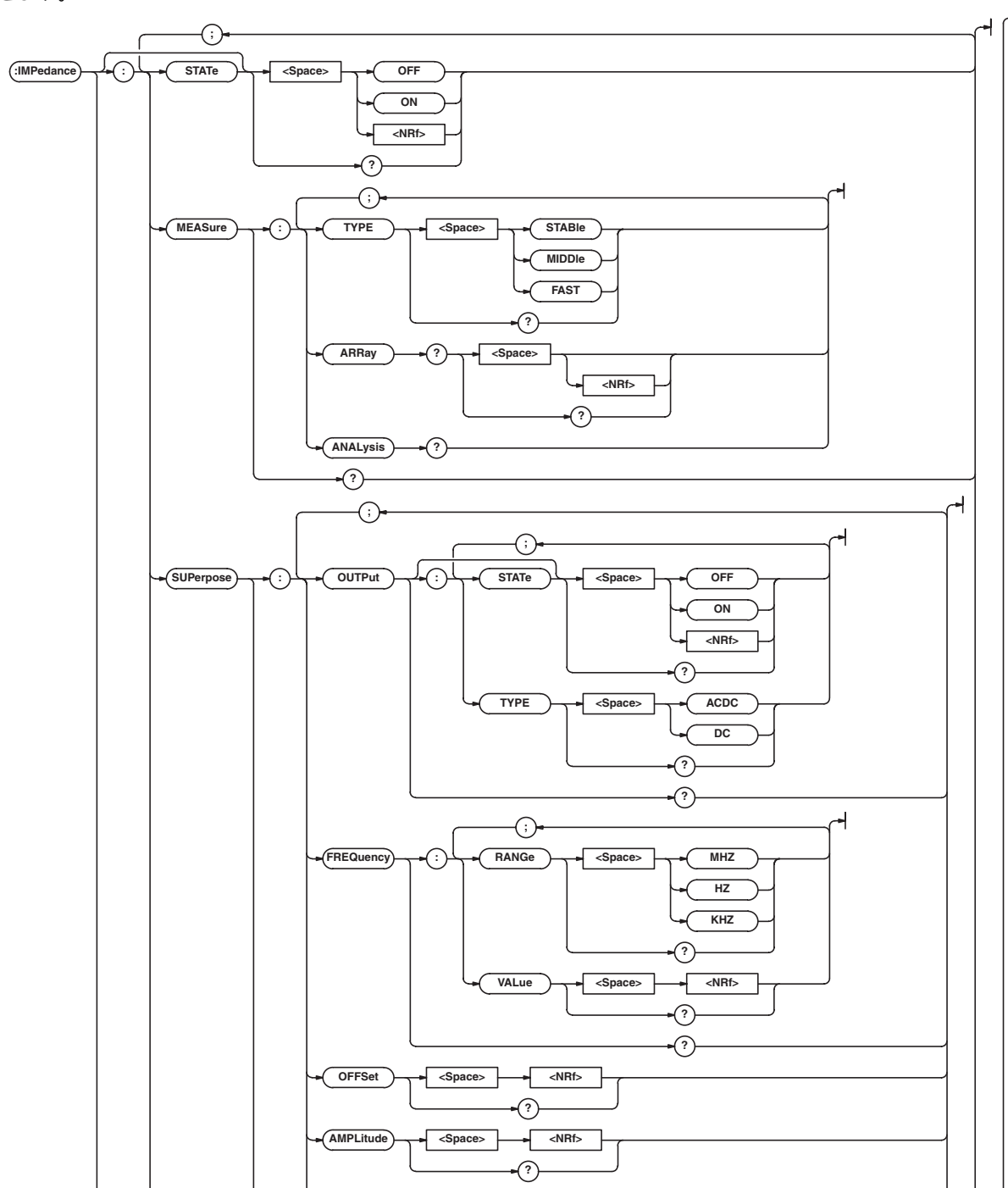
解説

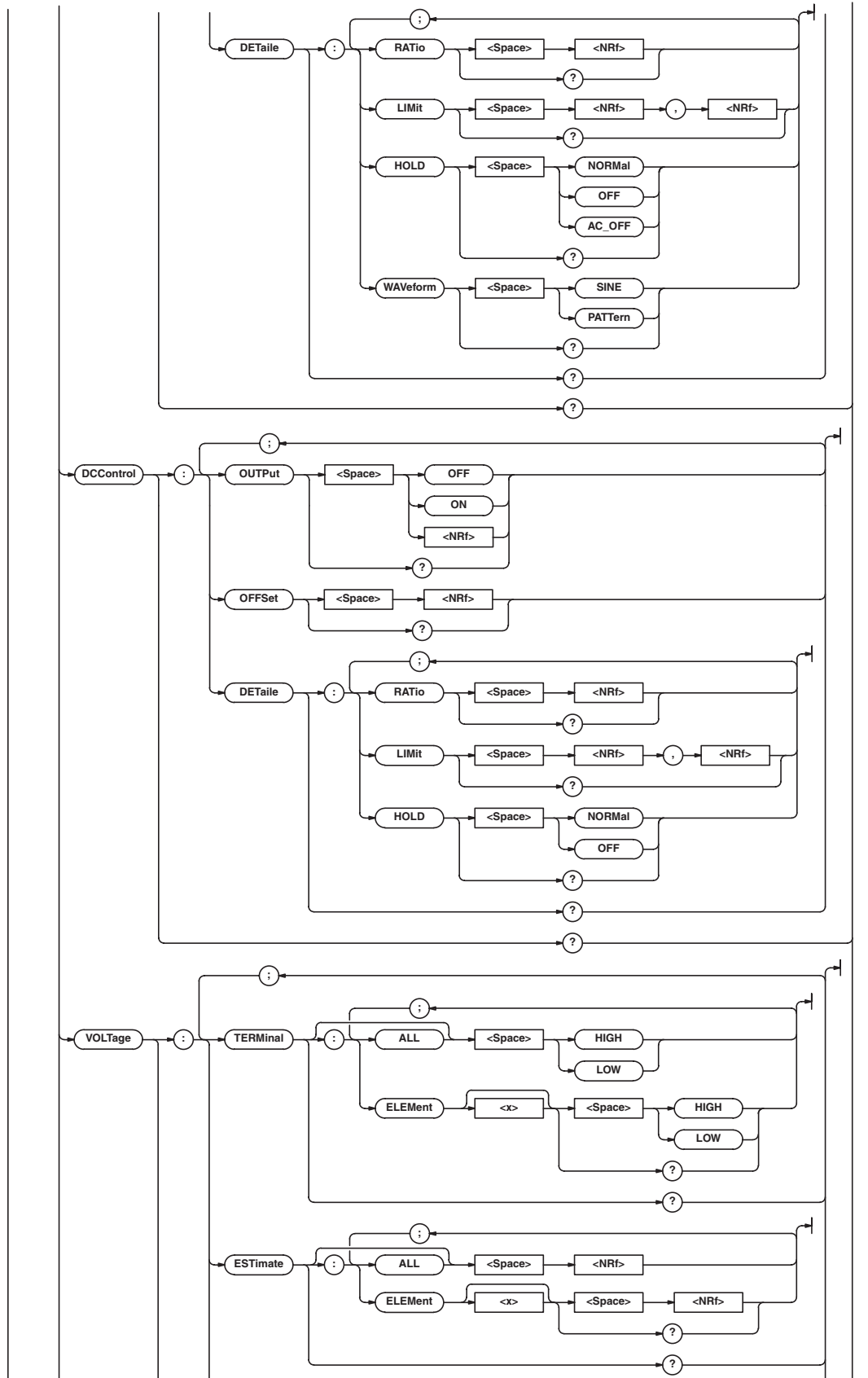
- ・ <ブロックデータ>のバイト数は、{2+6+データ数+1(デリミタ)}です。
- ・ <ブロックデータ>については、4-6ページを参照してください。

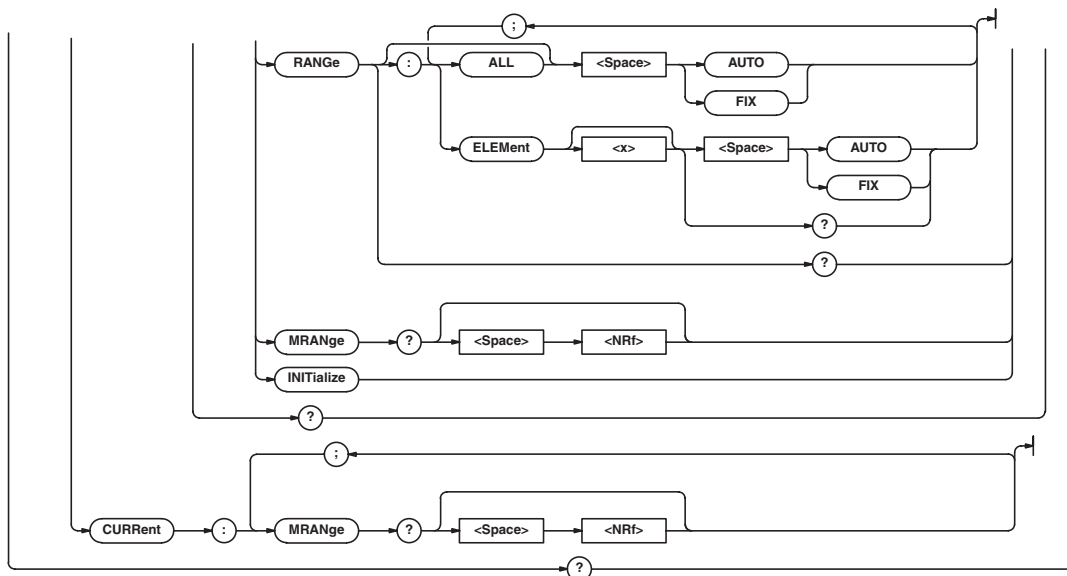
5.9 IMPedanceグループ

IMPedanceグループは、インピーダンス測定に関するグループです。

フロントパネルのIMPEDANCE、DC CONTROL(SHIFT+IMPEDANCE)と同じ設定、および設定内容の問い合わせができます。





**:IMPedance?**

機能 インピーダンス測定に関するすべての設定値を問い合わせます。

構文 :IMPedance?

例 :IMPEDANCE? -> :IMPEDANCE:STATE:1;
MEASURE:TYPE STABLE;;IMPEDANCE:
SUPERPOSE:OUTPUT:STATE 0;TYPE DC;;
IMPEDANCE:SUPERPOSE:FREQUENCY:
RANGE HZ;VALUE 1.0002;;IMPEDANCE:
SUPERPOSE:OFFSET 0.000;
AMPLITUDE 0.000;DETAILS:
RATIO 1.000;LIMIT 1000.000,
-1000.000;HOLD NORMAL;
WAVEFORM SINE;;IMPEDANCE:DCCONTROL:
OUTPUT 0;OFFSET 0.000;DETAILS:
RATIO 1.000;LIMIT 1000.000,
-1000.000;HOLD NORMAL;;IMPEDANCE:
VOLTAGE:TERMINAL:ELEMENT5 HIGH;;
IMPEDANCE:VOLTAGE:ESTIMATE:
ELEMENT5 0.0010;;IMPEDANCE:VOLTAGE:
RANGE:ELEMENT5 AUTO

:IMPedance:CURRENT:MRANGE? (Measured RANGE)

機能 現在の測定値に対する電流レンジを問い合わせます。

構文 :IMPedance:CURRENT:MRANGE? {<NRf>}

例 <NRf> = 1~5(インピーダンス測定エレメント)
(インピーダンス測定エレメント(Z5)が2と3に装
備されている場合の例)

:IMPEDANCE:CURRENT:MRANGE? 2

-> 5.000E+00

:IMPEDANCE:CURRENT:MRANGE?

-> 5.000E+00,5.000E+00

解説 ・画面右側に表示されている測定レンジを返しま
す。

・パラメータを省略した場合、装備されているす
べてのインピーダンス測定エレメントの測定レ
ンジを、小さいエレメント番号の方から順に出
力します。

:IMPedance:DCControl?

機能 直流負荷電流に関するすべての設定値を問い合
わせます。

構文 :IMPedance:DCControl?

例 :IMPEDANCE:DCCONTROL? ->

:IMPEDANCE:DCCONTROL:OUTPUT 0;

OFFSET 0.000;DETAILS:RATIO 1.000;

LIMIT 1000.000,-1000.000;

HOLD NORMAL

:IMPedance:DCControl:DETAile?

機能 直流負荷電流の詳細設定に関するすべての設定値
を問い合わせます。

構文 :IMPedance:DCControl:DETAile?

例 :IMPEDANCE:DCCONTROL:DETAILE? ->

:IMPEDANCE:DCCONTROL:DETAILE:

RATIO 1.000;LIMIT 1000.000,

-1000.000;HOLD NORMAL

:IMPedance:DCControl:DETAile:HOLD

機能 ホールドしたときの本機器の動作(直流電子負荷装置への制御信号の扱い)を設定/問い合わせします。

構文 :IMPedance:DCControl:DETAile:
HOLD {NORMAL|OFF}
:IMPedance:DCControl:DETAile:HOLD?
例 :IMPEDANCE:DCCONTROL:DETAILE:
HOLD NORMAL
:IMPEDANCE:DCCONTROL:DETAILE:HOLD?
-> :IMPEDANCE:DCCONTROL:DETAILE:
HOLD NORMAL

:IMPedance:DCControl:DETAile:LIMit

機能 直流負荷電流の範囲を設定/問い合わせします。

構文 :IMPedance:DCControl:DETAile:
LIMit {<NRf>,<NRf>}
:IMPedance:DCControl:DETAile:LIMit?
<NRf> = - 1000.000 ~ 1000.000(A)
例 :IMPEDANCE:DCCONTROL:DETAILE:
LIMIT 1000,-1000
:IMPEDANCE:DCCONTROL:DETAILE:LIMIT?
-> :IMPEDANCE:DCCONTROL:DETAILE:
LIMIT 1000.000,-1000.000
解説 上限値, 下限値の順で設定します。

:IMPedance:DCControl:DETAile:RATio

機能 直流電子負荷装置への制御信号1(V)あたりの電流値を設定/問い合わせします。

構文 :IMPedance:DCControl:DETAile:
RATio {<NRf>}
:IMPedance:DCControl:DETAile:RATio?
<NRf> = 0.001 ~ 1000.000
例 :IMPEDANCE:DCCONTROL:DETAILE:
RATIO 1
:IMPEDANCE:DCCONTROL:DETAILE:RATIO?
-> :IMPEDANCE:DCCONTROL:DETAILE:
RATIO 1.000

:IMPedance:DCControl:OFFSet

機能 直流負荷電流の電流値を設定/問い合わせします。

構文 :IMPedance:DCControl:OFFSet {<NRf>}
:IMPedance:DCControl:OFFSet?
<NRf> = - 1000.000 ~ 1000.000(A)
例 :IMPEDANCE:DCCONTROL:OFFSET 0
:IMPEDANCE:DCCONTROL:OFFSET? ->
:IMPEDANCE:DCCONTROL:OFFSET 0.000

:IMPedance:DCControl:OUTPut

機能 直流負荷電流のON/OFFを設定/問い合わせします。

構文 :IMPedance:DCControl:
OUTPut {<Boolean>}
:IMPedance:DCControl:OUTPut?
例 :IMPEDANCE:DCCONTROL:OUTPUT ON
:IMPEDANCE:DCCONTROL:OUTPUT? ->
:IMPEDANCE:DCCONTROL:OUTPUT 1

:IMPedance:MEASure?

機能 インピーダンス測定のFFT窓幅を問い合わせします。

構文 :IMPedance:MEASure?
例 :IMPEDANCE:MEASURE? ->
:IMPEDANCE:MEASURE:TYPE STABLE

:IMPedance:MEASure:ANALysis?

機能 インピーダンス測定モードの種類を問い合わせします。

構文 :IMPedance:MEASure:ANALysis?
例 :IMPEDANCE:MEASURE:ANALYSIS? ->
NORMAL

解説 ・ 応答の内容は以下のとおりです。
NORMAL : 通常モード
WIDE : 広帯域モード
・ モードの詳細については、本体ユーザズマニュアルをご覧ください。

:IMPedance:MEASure:ARRAY?

機能 ロードされたパターンファイルの配列情報を問い合わせします。

構文 :IMPedance:MEASure:ARRAY? {<NRf>}
<NRf> = 1 ~ 100(配列番号)
例 :IMPEDANCE:MEASURE:ARRAY? 1 -> 1
:IMPEDANCE:MEASURE:ARRAY? ->
1,2,5,10,-1,-1,...

解説 ・ 指定された配列番号の次数を出力します。
・ パラメータを省略した場合、配列1 ~ 100の次数を順に(カンマ区切り)出力します。

:IMPedance:MEASure:TYPE

機能 インピーダンス測定のFFT窓幅を設定/問い合わせします。

構文 :IMPedance:MEASure:TYPE {STABLE|
MIDDLE|FAST}
:IMPedance:MEASure:TYPE?
例 :IMPEDANCE:MEASURE:TYPE STABLE
:IMPEDANCE:MEASURE:TYPE? ->
:IMPEDANCE:MEASURE:TYPE STABLE

:IMPedance[:STATe]

機能 インピーダンス測定モードのON/OFFを設定/問い合わせします。

構文 :IMPedance[:STATe] {<Boolean>
:IMPedance:STATe?

例 :IMPEDANCE:STATE ON
:IMPEDANCE:STATE? ->
:IMPEDANCE:STATE 1

:IMPedance:SUPerpose?

機能 インピーダンス測定用負荷電流に関するすべての設定値を問い合わせます。

構文 :IMPedance:SUPerpose?

例 :IMPEDANCE:SUPERPOSE? ->
:IMPEDANCE:SUPERPOSE:OUTPUT:
STATE 0;TYPE DC;:IMPEDANCE:
SUPERPOSE:FREQUENCY:RANGE HZ;
VALUE 1.0002;:IMPEDANCE:SUPERPOSE:
OFFSET 0.000;AMPLITUDE 0.000;
DETAILE:RATIO 1.000;LIMIT 1000.000,
-1000.000;HOLD NORMAL;WAVEFORM SINE

:IMPedance:SUPerpose:AMPLitude

機能 インピーダンス測定用負荷電流の振幅を設定/問い合わせします。

構文 :IMPedance:SUPerpose:
AMPLitude {<NRf>}
:IMPedance:SUPerpose:AMPLitude?
<NRf> = 0.000 ~ 2000.000(App)

例 :IMPEDANCE:SUPERPOSE:AMPLITUDE 0
:IMPEDANCE:SUPERPOSE:AMPLITUDE? ->
:IMPEDANCE:SUPERPOSE:
AMPLITUDE 0.000

:IMPedance:SUPerpose:DETAile?

機能 インピーダンス測定用負荷電流の詳細設定に関するすべての設定値を問い合わせます。

構文 :IMPedance:SUPerpose:DETAile?

例 :IMPEDANCE:SUPERPOSE:DETAILE? ->
:IMPEDANCE:SUPERPOSE:DETAILE:
RATIO 1.000;LIMIT 1000.000,
-1000.000;HOLD NORMAL;WAVEFORM SINE

:IMPedance:SUPerpose:DETAile:HOLD

機能 ホールドしたときの本機器の動作(インピーダンス測定用電子負荷装置への制御信号の扱い)を設定/問い合わせします。

構文 :IMPedance:SUPerpose:DETAile:
HOLD {NORMAL|OFF|AC_OFF}
:IMPedance:SUPerpose:DETAile:HOLD?

例 :IMPEDANCE:SUPERPOSE:DETAILE:
HOLD NORMAL
:IMPEDANCE:SUPERPOSE:DETAILE:HOLD?
-> :IMPEDANCE:SUPERPOSE:DETAILE:
HOLD NORMAL

:IMPedance:SUPerpose:DETAile:LIMit

機能 インピーダンス測定用負荷電流の範囲を設定/問い合わせします。

構文 :IMPedance:SUPerpose:DETAile:
LIMit {<NRf>,<NRf>}
:IMPedance:SUPerpose:DETAile:LIMit?
<NRf> = - 1000.000 ~ 1000.000(A)

例 :IMPEDANCE:SUPERPOSE:DETAILE:
LIMIT 1000,-1000
:IMPEDANCE:SUPERPOSE:DETAILE:LIMIT?
-> :IMPEDANCE:SUPERPOSE:DETAILE:
LIMIT 1000.000,-1000.000

解説 上限値, 下限値の順で設定します。

:IMPedance:SUPerpose:DETAile:RATio

機能 インピーダンス測定用電子負荷装置への制御信号1(V)あたりの電流値を設定/問い合わせします。

構文 :IMPedance:SUPerpose:DETAile:
RATio {<NRf>}
:IMPedance:SUPerpose:DETAile:RATio?
<NRf> = 0.001 ~ 1000.000

例 :IMPEDANCE:SUPERPOSE:DETAILE:
RATIO 1
:IMPEDANCE:SUPERPOSE:DETAILE:RATIO?
-> :IMPEDANCE:SUPERPOSE:DETAILE:
RATIO 1.000

:IMPedance:SUPerpose:DETAile:WAVEform

機能 インピーダンス測定用負荷電流の波形を設定/問い合わせします。

構文 :IMPedance:SUPerpose:DETAile:
WAVEform {SINE|PATTern}
:IMPedance:SUPerpose:DETAile:
WAVEform?

例 :IMPEDANCE:SUPERPOSE:DETAILE:
WAVEFORM SINE
:IMPEDANCE:SUPERPOSE:DETAILE:
WAVEFORM? -> :IMPEDANCE:SUPERPOSE:
DETAILE: WAVEFORM SINE

:IMPedance:SUPerpose:FREQuency?

機能 インピーダンス測定用負荷電流の周波数に関するすべての設定値を問い合わせます。

構文 :IMPedance:SUPerpose:FREQuency?

例 :IMPEDANCE:SUPERPOSE:FREQUENCY? ->
:IMPEDANCE:SUPERPOSE:FREQUENCY:
RANGE HZ;VALUE 100.00

:IMPedance:Superpose:FREQuency:RANGe

機能 インピーダンス測定用負荷電流の周波数レンジを設定/問い合わせします。

構文 :IMPedance:Superpose:FREQuency:RANGe {MHZ|HZ|KHZ}

:IMPedance:Superpose:FREQuency:RANGe?

例 :IMPEDANCE:SuperPOSE:FREQUENCY:RANGE HZ

:IMPEDANCE:SuperPOSE:FREQUENCY:RANGE? -> :IMPEDANCE:SuperPOSE:FREQUENCY:RANGE HZ

:IMPedance:Superpose:FREQuency:VALue

機能 インピーダンス測定用負荷電流の周波数を設定/問い合わせします。

構文 :IMPedance:Superpose:FREQuency:VALue {<NRf>}

:IMPedance:Superpose:FREQuency:VALue?

<NRf> = 0.9313 ~ 891.28 (mHz)(周波数レンジ = mHzの場合)

<NRf> = 1.0002 ~ 992.06 (Hz)(周波数レンジ = Hzの場合)

<NRf> = 1.1161 ~ 50.000 (kHz)(周波数レンジ = kHzの場合)

例 :IMPEDANCE:SuperPOSE:FREQUENCY:VALUE 1.0002

:IMPEDANCE:SuperPOSE:FREQUENCY:VALUE? -> :IMPEDANCE:SuperPOSE:FREQUENCY:VALUE 1.0002

:IMPedance:Superpose:OFFSet

機能 インピーダンス測定用負荷電流の直流成分の大きさを設定/問い合わせします。

構文 :IMPedance:Superpose:OFFSet {<NRf>}

:IMPedance:Superpose:OFFSet?

<NRf> = - 1000.000 ~ 1000.000(A)

例 :IMPEDANCE:SuperPOSE:OFFSET 0

:IMPEDANCE:SuperPOSE:OFFSET? ->

:IMPEDANCE:SuperPOSE:OFFSET 0.000

:IMPedance:Superpose:OUTPut?

機能 インピーダンス測定用負荷電流の出力に関するすべての設定値を問い合わせます。

構文 :IMPedance:Superpose:OUTPut?

例 :IMPEDANCE:SuperPOSE:OUTPUT? ->

:IMPEDANCE:SuperPOSE:OUTPUT:STATE 0;TYPE DC

:IMPedance:Superpose:OUTPut[:STATE]

機能 インピーダンス測定用負荷電流のON/OFFを設定/問い合わせします。

構文 :IMPedance:Superpose:OUTPut[:STATE] {<Boolean>}

:IMPedance:Superpose:OUTPut:STATE?

例 :IMPEDANCE:SuperPOSE:OUTPUT:STATE ON

:IMPEDANCE:SuperPOSE:OUTPUT:STATE?

-> :IMPEDANCE:SuperPOSE:OUTPUT:

STATE 1

:IMPedance:Superpose:OUTPut:TYPE

機能 インピーダンス測定用負荷電流の出力タイプを設定/問い合わせします。

構文 :IMPedance:Superpose:OUTPut:TYPE {ACDC|DC}

:IMPedance:Superpose:OUTPut:TYPE?

例 :IMPEDANCE:SuperPOSE:OUTPUT:TYPE DC

:IMPEDANCE:SuperPOSE:OUTPUT:TYPE?

-> :IMPEDANCE:SuperPOSE:OUTPUT:

TYPE DC

:IMPedance:VOLTage?

機能 インピーダンス測定の電圧センス入力に関するすべての設定値を問い合わせます。

構文 :IMPedance:VOLTage?

例 :IMPEDANCE:VOLTAGE? -> :IMPEDANCE:

VOLTAGE:TERMINAL:ELEMENT5 HIGH;;

IMPEDANCE:VOLTAGE:ESTIMATE:

ELEMENT5 0.0010;;IMPEDANCE:VOLTAGE:

RANGE:ELEMENT5 AUTO

:IMPedance:VOLTage:ESTimate?

機能 すべてのインピーダンス測定エレメントのインピーダンス目安値を問い合わせます。

構文 :IMPedance:VOLTage:ESTimate?

例 :IMPEDANCE:VOLTAGE:ESTIMATE? ->

:IMPEDANCE:VOLTAGE:ESTIMATE:

ELEMENT5 0.0010

:IMPedance:VOLTage:ESTimate[:ALL]

機能 すべてのインピーダンス測定エレメントのインピーダンス目安値を一括設定します。

構文 :IMPedance:VOLTage:ESTimate[:ALL] {<NRf>}

<NRf> = 0.0001 ~ 10000.0000(Ω)

例 :IMPEDANCE:VOLTAGE:ESTIMATE:

ALL 0.001

: IMPedance: VOLTage: ESTimate: ELEMeNt<x>

機能 各インピーダンス測定エレメントのインピーダンス目安値を設定/問い合わせします。

構文 : IMPedance: VOLTage: ESTimate:
ELEMeNt<x> {<NRf>}
: IMPedance: VOLTage: ESTimate:
ELEMeNt<x>?
<x> = 1 ~ 5(インピーダンス測定エレメント)
<NRf> = 0.0001 ~ 10000.0000(Ω)

例 : IMPEDANCE: VOLTAGE: ESTIMATE:
ELEMENT5 0.001
: IMPEDANCE: VOLTAGE: ESTIMATE:
ELEMENT5? -> : IMPEDANCE: VOLTAGE:
ESTIMATE: ELEMENT5 0.0010

: IMPedance: VOLTage: INITialize

機能 電圧レンジを初期レンジに設定します。

構文 : IMPedance: VOLTage: INITialize

例 : IMPEDANCE: VOLTAGE: INITIALIZE

: IMPedance: VOLTage: MRANge? (Measured RANGE)

機能 現在の測定値に対する電圧レンジを問い合わせます。

構文 : IMPedance: VOLTage: MRANge? {<NRf>}
<NRf> = 1 ~ 5(インピーダンス測定エレメント)
例 (インピーダンス測定エレメント(Terminal=High)が2と3に装備されている場合の例)

: IMPEDANCE: VOLTAGE: MRANGE? 2 ->
150.0E-03
: IMPEDANCE: VOLTAGE: MRANGE? ->
150.0E-03, 150.0E-03

解説 ・画面右側に表示されている測定レンジを返します。
・パラメータを省略した場合、装備されているすべてのインピーダンス測定エレメントの測定レンジを、小さいエレメント番号の方から順に出力します。

: IMPedance: VOLTage: RANge?

機能 すべてのインピーダンス測定エレメントの電圧レンジモードを問い合わせます。

構文 : IMPedance: VOLTage: RANge?

例 : IMPEDANCE: VOLTAGE: RANGE? ->
: IMPEDANCE: VOLTAGE: RANGE:
ELEMENT5 AUTO

: IMPedance: VOLTage: RANge[: ALL]

機能 すべてのインピーダンス測定エレメントの電圧レンジモードを一括設定します。

構文 : IMPedance: VOLTage: RANge[:
ALL] {AUTO|FIX}

例 : IMPEDANCE: VOLTAGE: RANGE: ALL AUTO

: IMPedance: VOLTage: RANge: ELEMeNt<x>

機能 各インピーダンス測定エレメントの電圧レンジモードを設定/問い合わせします。

構文 : IMPedance: VOLTage: RANge:
ELEMeNt<x> {AUTO|FIX}
: IMPedance: VOLTage: RANge:
ELEMeNt<x>?
<x> = 1 ~ 5(インピーダンス測定エレメント)

例 : IMPEDANCE: VOLTAGE: RANGE:
ELEMENT5 AUTO
: IMPEDANCE: VOLTAGE: RANGE:
ELEMENT5? -> : IMPEDANCE: VOLTAGE:
RANGE: ELEMENT5 AUTO

: IMPedance: VOLTage: TERMinal?

機能 すべてのインピーダンス測定エレメントの電圧入力端子を問い合わせます。

構文 : IMPedance: VOLTage: TERMinal?

例 : IMPEDANCE: VOLTAGE: TERMINAL? ->
: IMPEDANCE: VOLTAGE: TERMINAL:
ELEMENT5 HIGH

: IMPedance: VOLTage: TERMinal[: ALL]

機能 すべてのインピーダンス測定エレメントの電圧入力端子を一括設定します。

構文 : IMPedance: VOLTage: TERMinal[:
ALL] {HIGH|LOW}

例 : IMPEDANCE: VOLTAGE: TERMINAL: ALL
HIGH

: IMPedance: VOLTage: TERMinal: ELEMeNt<x>

機能 各インピーダンス測定エレメントの電圧入力端子を設定/問い合わせします。

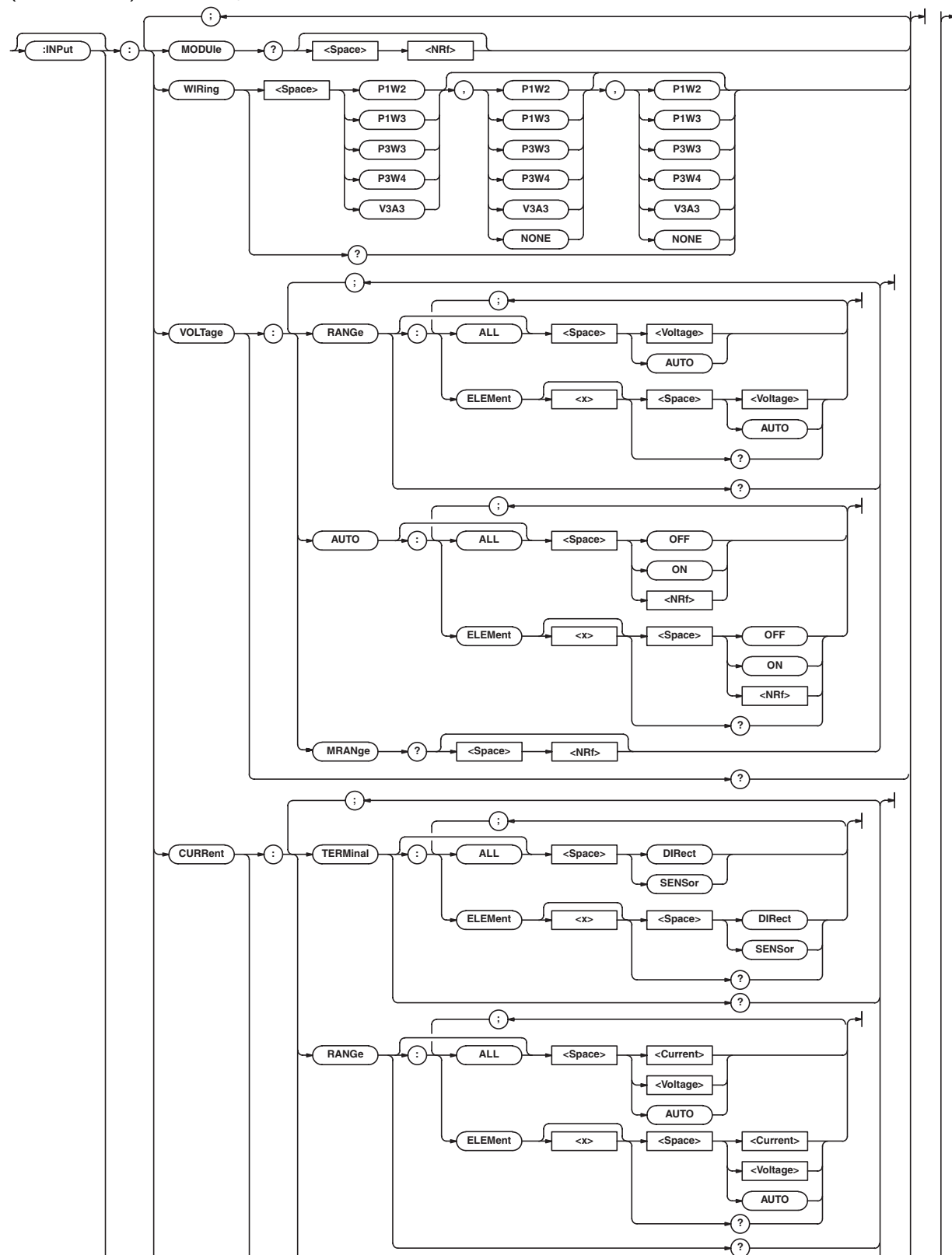
構文 : IMPedance: VOLTage: TERMinal:
ELEMeNt<x> {HIGH|LOW}
: IMPedance: VOLTage: TERMinal:
ELEMeNt<x>?
<x> = 1 ~ 5(インピーダンス測定エレメント)

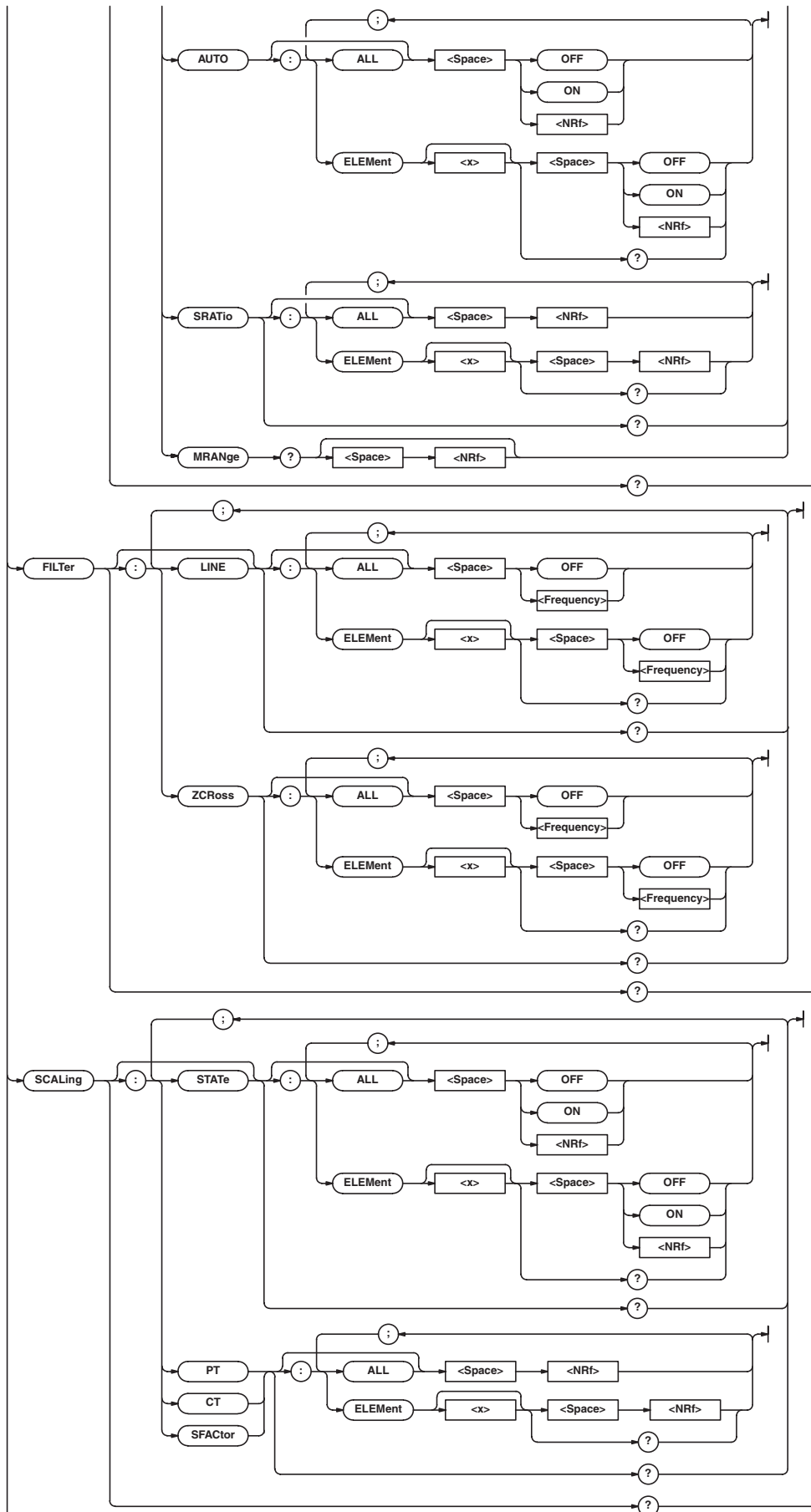
例 : IMPEDANCE: VOLTAGE: TERMINAL:
ELEMENT5 HIGH
: IMPEDANCE: VOLTAGE: TERMINAL:
ELEMENT5? -> : IMPEDANCE: VOLTAGE:
TERMINAL: ELEMENT5 HIGH

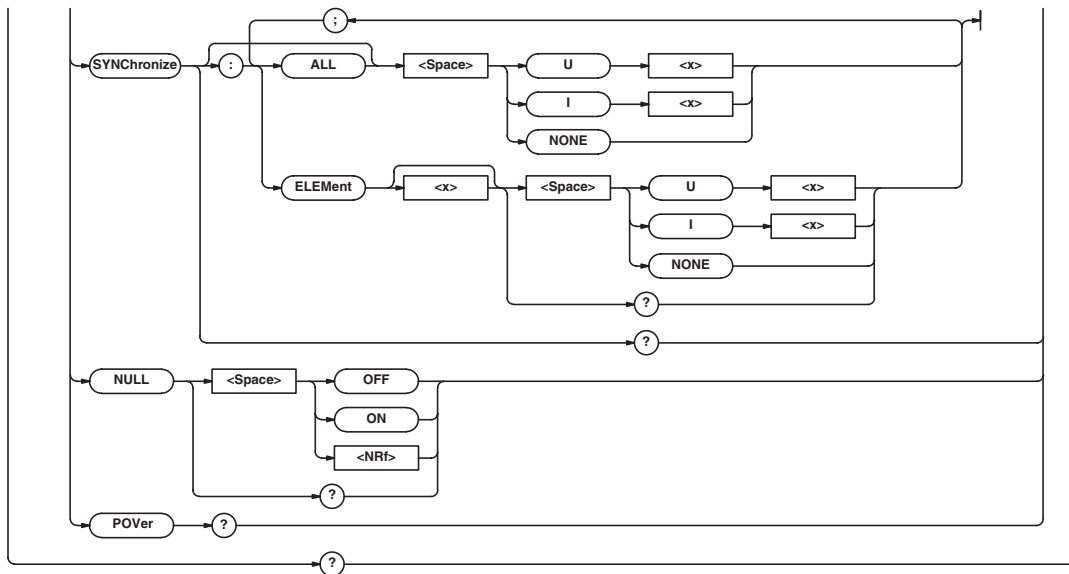
5.10 INPutグループ

INPutグループは、入力エレメントの測定条件に関するグループです。

フロントパネルのINPUTグループのRANGE、SCALING、WIRING、FILTER、SYNC SRC、およびNULL (SHIFT+MISC)と同じ設定、および設定内容の問い合わせができます。





**:INPut?**

機能 入力エレメントに関するすべての設定値を問い合わせます。

構文 :INPut?

例 :INPUT? -> :INPUT:

```
WIRING P1W2,P1W2,P1W2;VOLTAGE:
RANGE:ELEMENT1 1.0000E+03;
ELEMENT2 1.0000E+03;
ELEMENT3 1.0000E+03;
ELEMENT4 1.0000E+03;:INPUT:CURRENT:
TERMINAL:ELEMENT1 DIRECT;
ELEMENT2 DIRECT;ELEMENT3 DIRECT;
ELEMENT4 DIRECT;:INPUT:CURRENT:
RANGE:ELEMENT1 5.00E+00;
ELEMENT2 5.00E+00;ELEMENT3 5.00E+00;
ELEMENT4 5.00E+00;:INPUT:CURRENT:
SRATIO:ELEMENT1 10.0000;
ELEMENT2 10.0000;ELEMENT3 10.0000;
ELEMENT4 10.0000;:INPUT:FILTER:
LINE:ELEMENT1 OFF;ELEMENT2 OFF;
ELEMENT3 OFF;ELEMENT4 OFF;
ELEMENT5 OFF;:INPUT:FILTER:ZCROSS:
ELEMENT1 OFF;ELEMENT2 OFF;
ELEMENT3 OFF;ELEMENT4 OFF;:INPUT:
SCALING:STATE:ELEMENT1 0;ELEMENT2 0;
ELEMENT3 0;ELEMENT4 0;ELEMENT5 0;:
INPUT:SCALING:PT:ELEMENT1 1.0000;
ELEMENT2 1.0000;ELEMENT3 1.0000;
ELEMENT4 1.0000;ELEMENT5 1.0000;:
INPUT:SCALING:CT:ELEMENT1 1.0000;
ELEMENT2 1.0000;ELEMENT3 1.0000;
ELEMENT4 1.0000;ELEMENT5 1.0000;:
INPUT:SCALING:SFACTOR:
ELEMENT1 1.0000;ELEMENT2 1.0000;
ELEMENT3 1.0000;ELEMENT4 1.0000;:
INPUT:SYNCHRONIZE:ELEMENT1 I1;
ELEMENT2 I2;ELEMENT3 I3;
ELEMENT4 I4:INPUT:NULL 0
```

[:INPut] :CURRent?

機能 電流測定に関するすべての設定値を問い合わせます。

構文 [:INPut] :CURRent?

例 :INPUT:CURRENT? -> :INPUT:CURRENT:

```
TERMINAL:ELEMENT1 DIRECT;
ELEMENT2 DIRECT;ELEMENT3 DIRECT;
ELEMENT4 DIRECT;:INPUT:CURRENT:
RANGE:ELEMENT1 5.00E+00;
ELEMENT2 5.00E+00;
ELEMENT3 5.00E+00;
ELEMENT4 5.00E+00;:INPUT:CURRENT:
SRATIO:ELEMENT1 10.0000;
ELEMENT2 10.0000;ELEMENT3 10.0000;
ELEMENT4 10.0000;
```

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[:INPut] :CURRent :AUTO [:ALL]

機能 すべての電力測定エレメントの電流オートレンジ ON/OFFを一括設定します。

構文 [:INPut] :CURRent :AUTO

[:ALL] { <Boolean> }

例 :INPUT:CURRENT:AUTO:ALL ON

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : CURRent : AUTO : ELEMeNt < x >

機能 各電力測定エレメントの電流オートレンジON/OFFを設定/問い合わせします。

構文 [: INPut] : CURRent : AUTO : ELEMeNt < x > { < Boolean > }
[: INPut] : CURRent : AUTO : ELEMeNt < x > ?
< x > = 1 ~ 4 (電力測定エレメント)

例 : INPUT:CURRENT:AUTO:ELEMENT1 ON
: INPUT:CURRENT:AUTO:ELEMENT1? ->
: INPUT:CURRENT:AUTO:ELEMENT1 1

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : CURRent : MRANge? (Measured RANge)

機能 現在の測定値に対する電流レンジを問い合わせます。

構文 [: INPut] : CURRent : MRANge? { < NRf > }
< NRf > = 1 ~ 4 (電力測定エレメント)

例 : INPUT:CURRENT:MRANGE? 1 ->
5.00E+00
: INPUT:CURRENT:MRANGE? ->
5.00E+00,5.00E+00,5.00E+00,5.00E+00

解説 ・画面右側に表示されている測定レンジを返します。したがってオートレンジの場合でも現在の測定レンジが取得できます。
・パラメータを省略した場合、装備されているすべての電力測定エレメントの測定レンジをエレメント1から順に出力します。
・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : CURRent : RANge?

機能 すべての電力測定エレメントの電流レンジを問い合わせます。

構文 [: INPut] : CURRent : RANge?
例 : INPUT:CURRENT:RANGE? -> : INPUT:
CURRENT:RANGE:ELEMENT1 5.00E+00;
ELEMENT2 5.00E+00;
ELEMENT3 5.00E+00;
ELEMENT4 5.00E+00

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : CURRent : RANge [: ALL]

機能 すべての電力測定エレメントの電流レンジを一括設定します。

構文 [: INPut] : CURRent : RANge
[: ALL] { < 電流 > | < 電圧 > | AUTO }
・5A入力エレメントの時
< 電流 > = 10, 20, 50, 100, 200, 500 (mA), 1, 2, 5(A) (TERMinal = DIReCtの時)
< 電圧 > = 50, 100, 250, 500(mV), 1, 2.5, 5, 10(V) (TERMinal = SENsOrの時)
AUTO = オートレンジ
・50A入力エレメントの時
< 電流 > = 1, 2, 5, 10, 20, 50(A) (TERMinal = DIReCtの時)
< 電圧 > = 50, 100, 250, 500(mV), 1, 2.5, 5, 10(V) (TERMinal = SENsOrの時)
AUTO = オートレンジ

例 : INPUT:CURRENT:RANGE:ALL 5A
解説 ・エレメント1の入力エレメントタイプ(5A/50A)と電流測定端子の設定([: INPut] : CURRent : TERMinal : ELEMeNt1)によって選択できるレンジが決まります。したがって、エレメント1と同じ入力エレメントタイプで、かつ同じ電流測定端子の設定になっているエレメントのみ設定されます。
・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[:INPut] :CURRent:RANGe:ELEMent<x>

機能 各電力測定エレメントの電流レンジを設定/問い合わせします。

構文 **[:INPut] :CURRent:RANGe:ELEMent<x>** {<電流>|<電圧>|AUTO}
[:INPut] :CURRent:RANGe:ELEMent<x>?
 <x> = 1 ~ 4(電力測定エレメント)
 ・ 5A入力エレメントの時
 <電流> = 10, 20, 50, 100, 200, 500 (mA), 1, 2, 5(A) (TERMinal = DIReCtの時)
 <電圧> = 50, 100, 250, 500(mV), 1, 2.5, 5, 10(V) (TERMinal = SENSOorの時)
 AUTO = オートレンジ
 ・ 50A入力エレメントの時
 <電流> = 1, 2, 5, 10, 20, 50(A) (TERMinal = DIReCtの時)
 <電圧> = 50, 100, 250, 500(mV), 1, 2.5, 5, 10(V) (TERMinal = SENSOorの時)
 AUTO = オートレンジ

例 **:INPUT:CURRENT:RANGE:ELEMENT1 5A**
:INPUT:CURRENT:RANGE:ELEMENT1? ->
:INPUT:CURRENT:RANGE:ELEMENT1 5.00E+00

解説
 ・ 対象エレメントの入力エレメントタイプ(5A/50A)と電流測定端子の設定([:INPut] :CURRent:TERMinal:ELEMent<x>)によって選択できるレンジが決まります。
 ・ このコマンドで「AUTO」を設定すると、「[:INPut] :CURRent:AUTO:ELEMent<x>」を「ON」に設定したのと同じことになります。
 ・ インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[:INPut] :CURRent:SRATio?

機能 すべての電力測定エレメントの電流センサスケリング定数を問い合わせます。

構文 **[:INPut] :CURRent:SRATio?**
 例 **:INPUT:CURRENT:SRATIO? -> :INPUT:CURRENT:SRATIO:ELEMENT1 10.0000;**
ELEMENT2 10.0000;ELEMENT3 10.0000;
ELEMENT4 10.0000

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[:INPut] :CURRent:SRATio[:ALL]

機能 すべての電力測定エレメントの電流センサスケリング定数を一括設定します。

構文 **[:INPut] :CURRent:SRATio[:ALL]** {<NRf>}
 <NRf> = 0.0001 ~ 99999.9999

例 **:INPUT:CURRENT:SRATIO:ALL 10**
 解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[:INPut] :CURRent:SRATio:ELEMent<x>

機能 各電力測定エレメントの電流センサスケリング定数を設定/問い合わせします。

構文 **[:INPut] :CURRent:SRATio:ELEMent<x>** {<NRf>}
[:INPut] :CURRent:SRATio:ELEMent<x>?
 <x> = 1 ~ 4(電力測定エレメント)
 <NRf> = 0.0001 ~ 99999.9999

例 **:INPUT:CURRENT:SRATIO:ELEMENT1 10**
:INPUT:CURRENT:SRATIO:ELEMENT1? ->
:INPUT:CURRENT:SRATIO:ELEMENT1 10.0000

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[:INPut] :CURRent:TERMinal?

機能 すべての電力測定エレメントの電流測定端子を問い合わせます。

構文 **[:INPut] :CURRent:TERMinal?**
 例 **:INPUT:CURRENT:TERMINAL? -> :INPUT:CURRENT:TERMINAL:ELEMENT1 DIRECT;**
ELEMENT2 DIRECT;ELEMENT3 DIRECT;
ELEMENT4 DIRECT

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[:INPut] :CURRent:TERMinal[:ALL]

機能 すべての電力測定エレメントの電流測定端子を一括設定します。

構文 **[:INPut] :CURRent:TERMinal[:ALL]** {DIReCt|SENSOr}
 DIReCt = 直接入力
 SENSOor = 電流センサ入力

例 **:INPUT:CURRENT:TERMINAL:ALL DIRECT**
 解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : CURRent : TERMinAl : ELEMenT < x >

機能 各電力測定エレメントの電流測定端子を設定/問い合わせします。

構文 [: INPut] : CURRent : TERMinAl : ELEMenT < x > { DiREct | SENsOr }
[: INPut] : CURRent : TERMinAl : ELEMenT < x > ?
< x > = 1 ~ 4 (電力測定エレメント)
DiREct = 直接入力
SENsOr = 電流センサ入力

例 : INPUT:CURRENT:TERMINAL: ELEMENT1 DIRECT
: INPUT:CURRENT:TERMINAL: ELEMENT1? -> : INPUT:CURRENT:TERMINAL: ELEMENT1 DIRECT

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : FILTer ?

機能 フィルタに関するすべての設定値を問い合わせます。

構文 [: INPut] : FILTer ?
例 : INPUT:FILTER? -> : INPUT:FILTER: LINE: ELEMENT1 OFF; ELEMENT2 OFF; ELEMENT3 OFF; ELEMENT4 OFF; ELEMENT5 OFF; : INPUT:FILTER: ZCROSS: ELEMENT1 OFF; ELEMENT2 OFF; ELEMENT3 OFF; ELEMENT4 OFF

[: INPut] : FILTer : LINE ?

機能 すべてのエレメントのラインフィルタの設定を問い合わせます。

構文 [: INPut] : FILTer : LINE ?
例 : INPUT:FILTER: LINE? -> : INPUT: FILTER: LINE: ELEMENT1 OFF; ELEMENT2 OFF; ELEMENT3 OFF; ELEMENT4 OFF; ELEMENT5 OFF

[: INPut] : FILTer [: LINE] [: ALL]

機能 すべてのエレメントのラインフィルタを一括設定します。

構文 [: INPut] : FILTer [: LINE] [: ALL] { OFF | < 周波数 > }
OFF = ラインフィルタOFF
< 周波数 > = 500Hz, 5.5kHz (ラインフィルタON, カットオフ周波数)

例 : INPUT:FILTER: LINE: ALL OFF
解説 すべての電力測定エレメントに対して一括設定されます。インピーダンス測定エレメントに設定するには、「[: INPut] : FILTer [: LINE] : ELEMenT < x >」をお使いください。ただし、インピーダンス測定エレメントだけが装備されている製品に限り、このコマンドでインピーダンス測定エレメントに設定することができます。

[: INPut] : FILTer [: LINE] : ELEMenT < x >

機能 各エレメントのラインフィルタを設定/問い合わせします。

構文 [: INPut] : FILTer [: LINE] : ELEMenT < x > { OFF | < 周波数 > }
[: INPut] : FILTer [: LINE] : ELEMenT < x > ?
< x > = 1 ~ 5 (エレメント)
OFF = ラインフィルタOFF
< 周波数 > = 500Hz, 5.5kHz (ラインフィルタON, カットオフ周波数)

例 : INPUT:FILTER: LINE: ELEMENT1 OFF
: INPUT:FILTER: LINE: ELEMENT1? -> : INPUT:FILTER: LINE: ELEMENT1 OFF

[: INPut] : FILTer : ZCRoss ?

機能 すべての電力測定エレメントのゼロクロスフィルタの設定を問い合わせます。

構文 [: INPut] : FILTer : ZCRoss ?
例 : INPUT:FILTER: ZCROSS? -> : INPUT:FILTER: ZCROSS: ELEMENT1 OFF; ELEMENT2 OFF; ELEMENT3 OFF; ELEMENT4 OFF

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : FILTer : ZCRoss [: ALL]

機能 すべての電力測定エレメントのゼロクロスフィルタを一括設定します。

構文 [: INPut] : FILTer : ZCRoss [: ALL] { OFF | < 周波数 > }
OFF = ゼロクロスフィルタOFF
< 周波数 > = 500Hz (ゼロクロスフィルタON, カットオフ周波数)

例 : INPUT:FILTER: ZCROSS: ALL OFF
解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : FILTer : ZCRoss : ELEMenT < x >

機能 各電力測定エレメントのゼロクロスフィルタを設定/問い合わせします。

構文 [: INPut] : FILTer : ZCRoss : ELEMenT < x > { OFF | < 周波数 > }
[: INPut] : FILTer : ZCRoss : ELEMenT < x > ?
< x > = 1 ~ 4 (電力測定エレメント)
OFF = ゼロクロスフィルタOFF
< 周波数 > = 500Hz (ゼロクロスフィルタON, カットオフ周波数)

例 : INPUT:FILTER: ZCROSS: ELEMENT1 OFF
: INPUT:FILTER: ZCROSS: ELEMENT1? -> : INPUT:FILTER: ZCROSS: ELEMENT1 OFF
解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[:INPut] :MODUle?

機能 入力エレメントタイプを問い合わせます。

構文 `[ :INPut ] :MODUle? { <NRf> }`
`[ :INPut ] :MODUle?`
`<NRf> = 1 ~ 6 (エレメント)`

例 `:INPUT:MODULE? 1 -> 5`
`:INPUT:MODULE? -> 5,5,5,5,Z5,0`

解説

- ・ 応答の内容は次のとおりです。
5 = 電力測定5A入力エレメント
50 = 電力測定50A入力エレメント
Z5 = インピーダンス測定5A入力エレメント
Z20 = インピーダンス測定20A入力エレメント
0 = 入力エレメントなし
- ・ パラメータを省略した場合、すべてのエレメントの入力エレメントタイプをエレメント1から順に出力します。

[:INPut] :NULL

機能 NULL機能のON/OFFを設定/問い合わせします。

構文 `[ :INPut ] :NULL { <Boolean> }`
`[ :INPut ] :NULL?`

例 `:INPUT:NULL ON`
`:INPUT:NULL? -> :INPUT:NULL 1`

[:INPut] :POVer?

機能 ピークオーバ情報を問い合わせます。

構文 `[ :INPut ] :POVer?`

例 `:INPUT:POVER? -> 0`

解説

- ・ 各エレメントのピークオーバ情報を下記のとおり割り付けています。応答は、各ビットの10進数の和が返されます。
- ・ 例えば、応答が「16」の場合、U3にピークオーバが発生していることになります。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	15	U5	14	U4	13	U3	12	U2	11	U1

[:INPut] :SCALing?

機能 スケーリングに関するすべての設定値を問い合わせます。

構文 `[ :INPut ] :SCALing?`

例 `:INPUT:SCALING? -> :INPUT:SCALING:`
`STATE:ELEMENT1 0;ELEMENT2 0;`
`ELEMENT3 0;ELEMENT4 0;ELEMENT5 0;:`
`INPUT:SCALING:PT:ELEMENT1 1.0000;`
`ELEMENT2 1.0000;ELEMENT3 1.0000;`
`ELEMENT4 1.0000;ELEMENT5 1.0000;:`
`INPUT:SCALING:CT:ELEMENT1 1.0000;`
`ELEMENT2 1.0000;ELEMENT3 1.0000;`
`ELEMENT4 1.0000;ELEMENT5 1.0000;:`
`INPUT:SCALING:SFACTOR:`
`ELEMENT1 1.0000;ELEMENT2 1.0000;`
`ELEMENT3 1.0000;ELEMENT4 1.0000`

[:INPut] :SCALing: {PT|CT|SFActor}?

機能 すべてのエレメントの{電圧|電流|電力}スケール定数を問い合わせます。

構文 `[ :INPut ] :SCALing: {PT|CT|SFActor}?`

例 `:INPUT:SCALING:PT? -> :INPUT:`
`SCALING:PT:ELEMENT1 1.0000;`
`ELEMENT2 1.0000;ELEMENT3 1.0000;`
`ELEMENT4 1.0000;ELEMENT5 1.0000`

解説 `[ :INPut ] :SCALing:SFACTOR..` のコマンドは、インピーダンス測定エレメントだけが装備されている製品では無効です。

[:INPut] :SCALing: {PT|CT|SFActor} [:ALL]

機能 すべてのエレメントの{電圧|電流|電力}スケール定数を一括設定します。

構文 `[ :INPut ] :SCALing: {PT|CT|SFActor}`
`[ :ALL ] { <NRf> }`
`<NRf> = 0.0001 ~ 99999.9999`

例 `:INPUT:SCALING:PT:ALL 1`

解説

- ・ 「`[ :INPut ] :SCALing: {PT|CT} [ :ALL ]`」コマンドは、すべての電力測定エレメントに対して一括設定されます。インピーダンス測定エレメントに設定するには、「`[ :INPut ] :SCALing: {PT|CT} :ELEMENT<x>`」をお使いください。ただし、インピーダンス測定エレメントだけが装備されている製品に限り、このコマンドでインピーダンス測定エレメントに設定することができます。
- ・ `[ :INPut ] :SCALing:SFACTOR..` のコマンドは、インピーダンス測定エレメントだけが装備されている製品では無効です。

[:INPut] :SCALing: {PT|CT|SFActor} :**ELEMENT<x>**

機能 各エレメントの{電圧|電流|電力}スケール定数を設定/問い合わせします。

構文 `[ :INPut ] :SCALing: {PT|CT|SFActor} :`
`ELEMENT<x> { <NRf> }`
`[ :INPut ] :SCALing: {PT|CT|SFActor} :`
`ELEMENT<x>?`

- ・ `[ :INPut ] :SCALing: {PT|CT} :ELEMENT<x>` のとき
`<x> = 1 ~ 5 (エレメント)`
- ・ `[ :INPut ] :SCALing: {SFACTOR} :ELEMENT<x>` のとき
`<x> = 1 ~ 4 (電力測定エレメント)`

`<NRf> = 0.0001 ~ 99999.9999`

例 `:INPUT:SCALING:PT:ELEMENT1 1`
`:INPUT:SCALING:PT:ELEMENT1? ->`
`:INPUT:SCALING:PT:ELEMENT1 1.0000`

解説 `[ :INPut ] :SCALing:SFACTOR..` のコマンドは、インピーダンス測定エレメントだけが装備されている製品では無効です。

[: INPut] : SCALing : STATe ?

機能 すべてのエレメントのスケールON/OFFを問い合わせます。

構文 `[ : INPut ] : SCALing : STATe ?`
 例 `: INPUT : SCALING : STATE ? -> : INPUT : SCALING : STATE : ELEMENT1 0 ; ELEMENT2 0 ; ELEMENT3 0 ; ELEMENT4 0 ; ELEMENT5 0`

[: INPut] : SCALing [: STATe] [: ALL]

機能 すべてのエレメントのスケールON/OFFを一括設定します。

構文 `[ : INPut ] : SCALing [ : STATe ] [ : ALL ] { < Boolean > }`
 例 `: INPUT : SCALING : STATE : ALL OFF`
 解説 すべての電力測定エレメントに対して一括設定されます。インピーダンス測定エレメントに設定するには、「`[ : INPut ] : SCALing [ : STATe ] : ELEMENT < x > ?`」をお使いください。ただし、インピーダンス測定エレメントだけが装備されている製品に限り、このコマンドでインピーダンス測定エレメントに設定することができます。

[: INPut] : SCALing [: STATe] : ELEMENT < x > ?

機能 各エレメントのスケールON/OFFを設定/問い合わせします。

構文 `[ : INPut ] : SCALing [ : STATe ] : ELEMENT < x > { < Boolean > }`
`[ : INPut ] : SCALing [ : STATe ] : ELEMENT < x > ?`
`< x > = 1 ~ 5 (エレメント)`
 例 `: INPUT : SCALING : STATE : ELEMENT1 OFF`
`: INPUT : SCALING : STATE : ELEMENT1 ? ->`
`: INPUT : SCALING : STATE : ELEMENT1 0`

[: INPut] : SYNChronize ?

機能 すべての電力測定エレメントの同期ソースを問い合わせます。

構文 `[ : INPut ] : SYNChronize ?`
 例 `INPUT : SYNCHRONIZE ? -> : INPUT : SYNCHRONIZE : ELEMENT1 I1 ; ELEMENT2 I2 ; ELEMENT3 I3 ; ELEMENT4 I4`
 解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : SYNChronize [: ALL]

機能 すべての電力測定エレメントの同期ソースを一括設定します。

構文 `[ : INPut ] : SYNChronize [ : ALL ] { U < x > | I < x > | NONE }`
`< x > = 1 ~ 4 (電力測定エレメント)`
`NONE = 同期ソースなし`
 例 `: INPUT : SYNCHRONIZE : ALL I1`
 解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : SYNChronize : ELEMENT < x > ?

機能 各電力測定エレメントの同期ソースを設定/問い合わせします。

構文 `[ : INPut ] : SYNChronize : ELEMENT < x > { U < x > | I < x > | NONE }`
`[ : INPut ] : SYNChronize : ELEMENT < x > ?`
`< x > = 1 ~ 4 (電力測定エレメント)`
`NONE = 同期ソースなし`
 例 `: INPUT : SYNCHRONIZE : ELEMENT1 I1`
`: INPUT : SYNCHRONIZE : ELEMENT1 ? ->`
`: INPUT : SYNCHRONIZE : ELEMENT1 I1`
 解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : VOLTage ?

機能 電圧測定に関するすべての設定値を問い合わせます。

構文 `[ : INPut ] : VOLTage ?`
 例 `: INPUT : VOLTAGE ? -> : INPUT : VOLTAGE : RANGE : ELEMENT1 1.0000E+03 ; ELEMENT2 1.0000E+03 ; ELEMENT3 1.0000E+03 ; ELEMENT4 1.0000E+03 ;`
 解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : VOLTage : AUTO [: ALL]

機能 すべての電力測定エレメントの電圧オートレンジON/OFFを一括設定します。

構文 `[ : INPut ] : VOLTage : AUTO [ : ALL ] { < Boolean > }`
 例 `: INPUT : VOLTAGE : AUTO : ALL ON`
 解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : VOLTage : AUTO : ELEMeNt <x>

機能	各電力測定エレメントの電圧オートレンジON/OFFを設定/問い合わせします。
構文	[: INPut] : VOLTage : AUTO : ELEMeNt <x> { <Boolean> } [: INPut] : VOLTage : AUTO : ELEMeNt <x> ? <x> = 1 ~ 4 (電力測定エレメント)
例	: INPUT : VOLTAGE : AUTO : ELEMENT1 ON : INPUT : VOLTAGE : AUTO : ELEMENT1 ? -> : INPUT : VOLTAGE : AUTO : ELEMENT1 1
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : VOLTage : MRANge ? (Measured RANge)

機能	現在の測定値に対する電圧レンジを問い合わせます。
構文	[: INPut] : VOLTage : MRANge ? { <NRF> } <NRF> = 1 ~ 4 (電力測定エレメント)
例	: INPUT : VOLTAGE : MRANGE ? 1 -> 1.0000E+03 : INPUT : VOLTAGE : MRANGE ? -> 1.0000E+03, 1.0000E+03, 1.0000E+03, 1.0000E+03
解説	- 画面右側に表示されている測定レンジを返します。したがってオートレンジのONの場合でも現在の測定レンジが取得できます。 - パラメータを省略した場合、装備されているすべての電力測定エレメントの測定レンジをエレメント1から順に出力します。 - インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : VOLTage : RANge ?

機能	すべての電力測定エレメントの電圧レンジを問い合わせます。
構文	[: INPut] : VOLTage : RANge ?
例	: INPUT : VOLTAGE : RANGE ? -> : INPUT : VOLTAGE : RANGE : ELEMENT1 1.0000E+03; ELEMENT2 1.0000E+03; ELEMENT3 1.0000E+03; ELEMENT4 1.0000E+03
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : VOLTage : RANge [: ALL]

機能	すべての電力測定エレメントの電圧レンジを一括設定します。
構文	[: INPut] : VOLTage : RANge [: ALL] { <電圧> AUTO } <電圧> = 1.5, 3, 6, 10, 15, 30, 60, 100, 150, 300, 600, 1000(V) AUTO = オートレンジ
例	: INPUT : VOLTAGE : RANGE : ALL 1000V
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : VOLTage : RANge : ELEMeNt <x>

機能	各電力測定エレメントの電圧レンジを設定/問い合わせします。
構文	[: INPut] : VOLTage : RANge : ELEMeNt <x> { <電圧> AUTO } [: INPut] : VOLTage : RANge : ELEMeNt <x> ? <x> = 1 ~ 4 (電力測定エレメント) <電圧> = 1.5, 3, 6, 10, 15, 30, 60, 100, 150, 300, 600, 1000(V) AUTO = オートレンジ
例	: INPUT : VOLTAGE : RANGE : ELEMENT1 1000V : INPUT : VOLTAGE : RANGE : ELEMENT1 ? -> : INPUT : VOLTAGE : RANGE : ELEMENT1 1.0000E+03
解説	- このコマンドで「AUTO」を設定すると、「[: INPut] : VOLTage : AUTO : ELEMeNt <x>」を「ON」に設定したのと同じことになります。 - インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

[: INPut] : WIRing

機能 結線方式を設定/問い合わせします。

構文 `[: INPut] : WIRing { (P1W2 | P1W3 | P3W3 |
P3W4 | V3A3) [, (P1W2 | P1W3 | P3W3 | P3W4 |
V3A3 | NONE)] [, (P1W2 | P1W3 | P3W3 | P3W4 |
V3A3 | NONE)] }`
`[ : INPut ] : WIRing?`
P1W2 = 単相2線式
P1W3 = 単相3線式
P3W3 = 三相3線式
P3W4 = 三相4線式
V3A3 = 3電圧3電流計法
NONE = 結線なし

例 `: INPUT: WIRING P1W2, P1W2, P1W2`
`: INPUT: WIRING? -> : INPUT:`
`WIRING P1W2, P1W2, P1W2`

解説

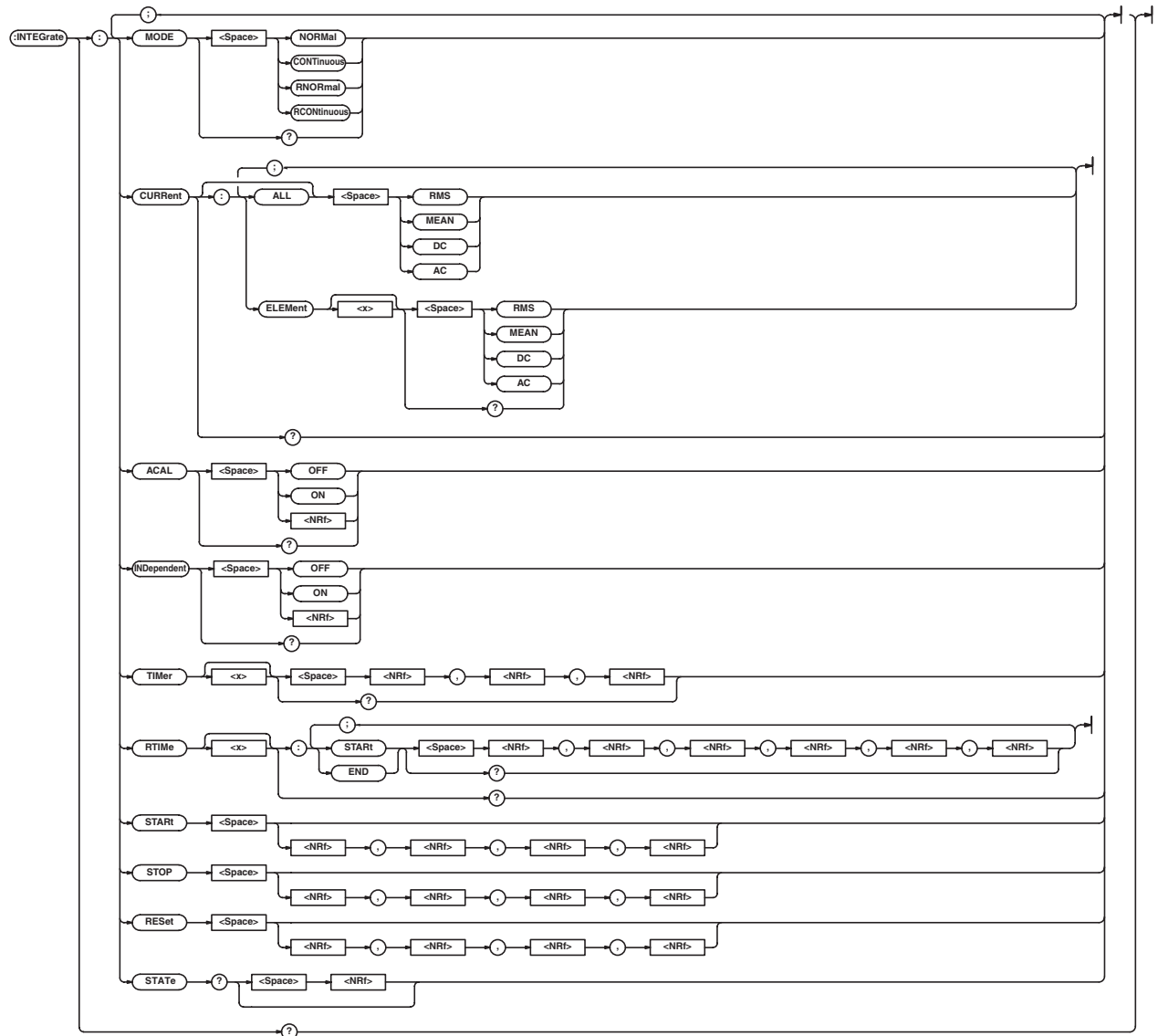
- ・ ΣA , ΣB , ΣC の順で設定します。
- ・ ΣB あるいは ΣC が設定できない組み合わせの場合は、省略することができます。
- ・ モデルタイプによっては、選択できない結線方式の組み合わせがあります。結線方式の組み合わせについては、本体ユーザズマニュアルを参照してください。
- ・ 電力測定エレメントが1つのモデルの場合、 $\Sigma A = P1W2$ 固定、 ΣB , ΣC の設定はできません。
- ・ インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

5.11 INTEGrateグループ

INTEGrateグループは、積算に関するグループです。

一部のコマンドを除き、フロントパネルのINTEGRATORグループのSTART, STOP, RESET(SHIFT+STOP), INTEG SET(SHIFT+START)と同じ設定、および設定内容の問い合わせができます。

ただし、インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このグループのコマンドは無効です。



:INTEGrate?

機能	積算に関するすべての設定値を問い合わせます。
構文	:INTEGrate?
例	- 標準積算モード、電力測定エレメント別積算OFFの場合の例 :INTEGRATE? -> :INTEGRATE: MODE NORMAL;CURRENT:ELEMENT1 RMS; ELEMENT2 RMS;ELEMENT3 RMS; ELEMENT4 RMS;;INTEGRATE:ACAL 0; INDEPENDENT 0;TIMER1 1,0,0 - 実時間制御標準積算モード、電力測定エレメント別積算OFFの場合の例 :INTEGRATE? -> :INTEGRATE: MODE RNORMAL;CURRENT: ELEMENT1 RMS;ELEMENT2 RMS; ELEMENT3 RMS;ELEMENT4 RMS;; INTEGRATE:ACAL 0;INDEPENDENT 0; TIMER1 1,0,0;RTIME1: START 2001,1,1,0,0,0; END 2001,1,1,1,0,0 - 標準積算モード、電力測定エレメント別積算ONの場合の例 :INTEGRATE? -> :INTEGRATE: MODE NORMAL;CURRENT:ELEMENT1 RMS; ELEMENT2 RMS;ELEMENT3 RMS; ELEMENT4 RMS;;INTEGRATE:ACAL 0; INDEPENDENT 1;TIMER1 1,0,0; TIMER2 1,0,0;TIMER3 1,0,0; TIMER4 1,0,0 - 実時間制御標準積算モード、電力測定エレメント別積算ONの場合の例 :INTEGRATE? -> :INTEGRATE: MODE RNORMAL;CURRENT: ELEMENT1 RMS;ELEMENT2 RMS; ELEMENT3 RMS;ELEMENT4 RMS;; INTEGRATE:ACAL 0;INDEPENDENT 1; TIMER1 1,0,0;TIMER2 1,0,0; TIMER3 1,0,0;TIMER4 1,0,0; RTIME1:START 2001,1,1,0,0,0; END 2001,1,1,1,0,0;;INTEGRATE: RTIME2:START 2001,1,1,0,0,0; END 2001,1,1,1,0,0;;INTEGRATE: RTIME3:START 2001,1,1,0,0,0; END 2001,1,1,1,0,0;;INTEGRATE: RTIME4:START 2001,1,1,0,0,0; END 2001,1,1,1,0,0

:INTEGrate:ACAL

機能	オートキャリブレーションのON/OFFを設定/問い合わせします。
構文	:INTEGrate:ACAL {<Boolean>}
例	:INTEGrate:ACAL? :INTEGRATE:ACAL OFF :INTEGRATE:ACAL? -> :INTEGRATE: ACAL 0

:INTEGrate:CURRent?

機能	すべての電力測定エレメントの電流積算の電流モードを問い合わせます。
構文	:INTEGrate:CURRent?
例	:INTEGRATE:CURRENT? -> :INTEGRATE: CURRENT:ELEMENT1 RMS;ELEMENT2 RMS; ELEMENT3 RMS;ELEMENT4 RMS

:INTEGrate:CURRent[:ALL]

機能	すべての電力測定エレメントの電流積算の電流モードを一括設定します。
構文	:INTEGrate:CURRent[:ALL] {RMS MEAN DC AC}
例	:INTEGRATE:CURRENT:ALL RMS

:INTEGrate:CURRent:ELEMEnt<x>

機能	各電力測定エレメントの電流積算の電流モードを設定/問い合わせします。
構文	:INTEGrate:CURRent:ELEMEnt<x> {RMS MEAN DC AC}
	:INTEGrate:CURRent:ELEMEnt<x>?
	<x> = 1 ~ 4(電力測定エレメント)
例	:INTEGRATE:CURRENT:ELEMENT1 RMS :INTEGRATE:CURRENT:ELEMENT1? -> :INTEGRATE:CURRENT:ELEMENT1 RMS
解説	電力測定エレメント別積算 (:INTEGrate:INDEpendent) のON/OFFに関係なく、各電力測定エレメントの電流積算の電流モードの設定により動作します。

:INTEGrate:INDEpendent

機能	電力測定エレメント別積算のON/OFFを設定/問い合わせします。
構文	:INTEGrate:INDEpendent {<Boolean>}
	:INTEGrate:INDEpendent?
例	:INTEGRATE:INDEPENDENT OFF :INTEGRATE:INDEPENDENT? -> :INTEGRATE:INDEPENDENT 0

:INTEGrate:MODE

機能	積算モードを設定/問い合わせします。
構文	:INTEGrate:MODE {NORMAL CONTInuous RNORMAL RCONTInuous}
	:INTEGrate:MODE?
	NORMAL = 標準積算モード CONTInuous = 連続積算モード RNORMAL = 実時間制御標準積算モード RCONTInuous = 実時間制御連続積算モード
例	:INTEGRATE:MODE NORMAL :INTEGRATE:MODE? -> :INTEGRATE: MODE NORMAL

:INTEGrate:RESet

機能 積算値をリセットします。

構文 `:INTEGrate:RESet {<NRf>,<NRf>,<NRf>,<NRf>}`
`<NRf> = 1 ~ 4(実行する電力測定エレメント)`

例

- 電力測定エレメント別積算(:INTEGrate:INDependent)が「ON(1)」のときの例
`:INTEGRATE:RESET`
(全電力測定エレメント実行)
`:INTEGRATE:RESET 1,2,3`
(電力測定エレメントを指定して実行)
- 電力測定エレメント別積算(:INTEGrate:INDependent)が「OFF(0)」のときの例
`:INTEGRATE:RESET`
(全電力測定エレメント実行)

解説

- 電力測定エレメント別積算(:INTEGrate:INDependent)が「ON(1)」のときは、実行する電力測定エレメントをパラメータとして4つまで指定することができます。ただし、この方法は、通信でのみ行えます。この方法に相当するフロントパネルのキーはありません。パラメータを省略した場合は、すべての電力測定エレメントを指定したのと同じことになります。
- 電力測定エレメント別積算(:INTEGrate:INDependent)が「OFF(0)」のときは、パラメータを指定することはできません。エラーとなります。

:INTEGrate:RTIME<x>?

機能 実時間制御積算モードにおける積算スタート/ストップ時刻を問い合わせます。

構文 `:INTEGrate:RTIME<x>?`
`<x> = 1 ~ 4(電力測定エレメント)`

例 `:INTEGRATE:RTIME1? -> :INTEGRATE:RTIME1:START 2001,1,1,0,0,0;`
`END 2001,1,1,1,0,0`

解説 電力測定エレメント別積算(:INTEGrate:INDependent)が「OFF(0)」のときは、電力測定エレメント1の積算スタート/ストップ時刻の設定により動作します。他のエレメントへの問い合わせは、エラーとなります。

:INTEGrate:RTIME<x>:{START|END}

機能 実時間制御積算モードにおける積算{スタート|ストップ}時刻を設定/問い合わせします。

構文 `:INTEGrate:RTIME<x>:{START|END} {<NRf>,<NRf>,<NRf>,<NRf>,<NRf>,<NRf>}`
`:INTEGrate:RTIME<x>:{START|END}?`
`<x> = 1 ~ 4(電力測定エレメント)`
`{<NRf>,<NRf>,<NRf>,<NRf>,<NRf>,<NRf>}` = 2001,1,1,0,0,0 ~ 2099,12,31,23,59,59

1つ目の<NRf> = 2001 ~ 2099 (年)
2つ目の<NRf> = 1 ~ 12 (月)
3つ目の<NRf> = 1 ~ 31 (日)
4つ目の<NRf> = 0 ~ 23 (時)
5つ目の<NRf> = 0 ~ 59 (分)
6つ目の<NRf> = 0 ~ 59 (秒)

例 `:INTEGRATE:RTIME1:`
`START 2001,1,1,0,0,0`
`:INTEGRATE:RTIME1:START? ->`
`:INTEGRATE:RTIME1:`
`START 2001,1,1,0,0,0`

解説 電力測定エレメント別積算(:INTEGrate:INDependent)が「OFF(0)」のときは、電力測定エレメント1の積算スタート/ストップ時刻の設定により動作します。他のエレメントへの設定/問い合わせは、エラーとなります。

:INTEGrate:START

機能 積算をスタートします。

構文 :INTEGrate:START {<NRf>,<NRf>,<NRf>,<NRf>}
<NRf> = 1 ~ 4(実行する電力測定エレメント)

例 ・電力測定エレメント別積算
(:INTEGrate:INDependent)が「ON(1)」のときの例
:INTEGRATE:START
(全電力測定エレメント実行)
:INTEGRATE:START 1,2,3
(電力測定エレメントを指定して実行)

解説 ・電力測定エレメント別積算
(:INTEGrate:INDependent)が「OFF(0)」のときの例
:INTEGRATE:START
(全電力測定エレメント実行)
・電力測定エレメント別積算(:INTEGrate:INDependent)が「ON(1)」のときは、実行する電力測定エレメントをパラメータとして4つまで指定することができます。ただし、この方法は、通信でのみ行えます。この方法に相当するフロントパネルのキーはありません。パラメータを省略した場合は、すべての電力測定エレメントを指定したのと同じこととなります。
・電力測定エレメント別積算(:INTEGrate:INDependent)が「OFF(0)」のときは、パラメータを指定することはできません。エラーとなります。

:INTEGrate:STATE?

機能 積算状態を問い合わせます。

構文 :INTEGrate:STATE? {<NRf>}
<NRf> = 1 ~ 4(問い合わせる電力測定エレメント)

例 :INTEGRATE:STATE? 1 -> RESET (指定した電力測定エレメントを問い合わせ)
:INTEGRATE:STATE? ->
RESET,RESET,RESET,RESET
(全電力測定エレメントを問い合わせ)

解説 ・応答の内容は次のとおりです。
RESEt = 積算リセット
READy = 待機中(実時間制御積算モード)
STARt = 積算実行中
STOP = 積算ストップ
ERRor = 積算異常終了(積算値オーバフロー、停電)
TImeup = 積算タイマ時間による積算ストップ
・パラメータを省略した場合は、装備されているすべての電力測定エレメントの状態を問い合わせます。装備されていない電力測定エレメントをパラメータに指定した場合、エラーとなります。

:INTEGrate:STOP

機能 積算をストップします。

構文 :INTEGrate:STOP {<NRf>,<NRf>,<NRf>,<NRf>}
<NRf> = 1 ~ 4(実行する電力測定エレメント)

例 ・電力測定エレメント別積算(:INTEGrate:INDependent)が「ON(1)」のときの例
:INTEGRATE:STOP
(全電力測定エレメント実行)
:INTEGRATE:STOP 1,2,3
(電力測定エレメントを指定して実行)

解説 ・電力測定エレメント別積算(:INTEGrate:INDependent)が「OFF(0)」のときは、実行する電力測定エレメントをパラメータとして4つまで指定することができます。ただし、この方法は、通信でのみ行えます。この方法に相当するフロントパネルのキーはありません。パラメータを省略した場合は、すべての電力測定エレメントを指定したのと同じこととなります。
・電力測定エレメント別積算(:INTEGrate:INDependent)が「OFF(0)」のときは、パラメータを指定することはできません。エラーとなります。

:INTEGrate:TMER<x>

機能 積算タイマ時間を設定/問い合わせします。

構文 :INTEGrate:TMER<x> {<NRf>,<NRf>,<NRf>}
:INTEGrate:TMER<x>?
<x> = 1 ~ 4(電力測定エレメント)
{<NRf>,<NRf>,<NRf>} = 0,0,0 ~ 10000,0,0
1つ目の<NRf> = 0 ~ 10000 (時間)
2つ目の<NRf> = 0 ~ 59 (分)
3つ目の<NRf> = 0 ~ 59 (秒)

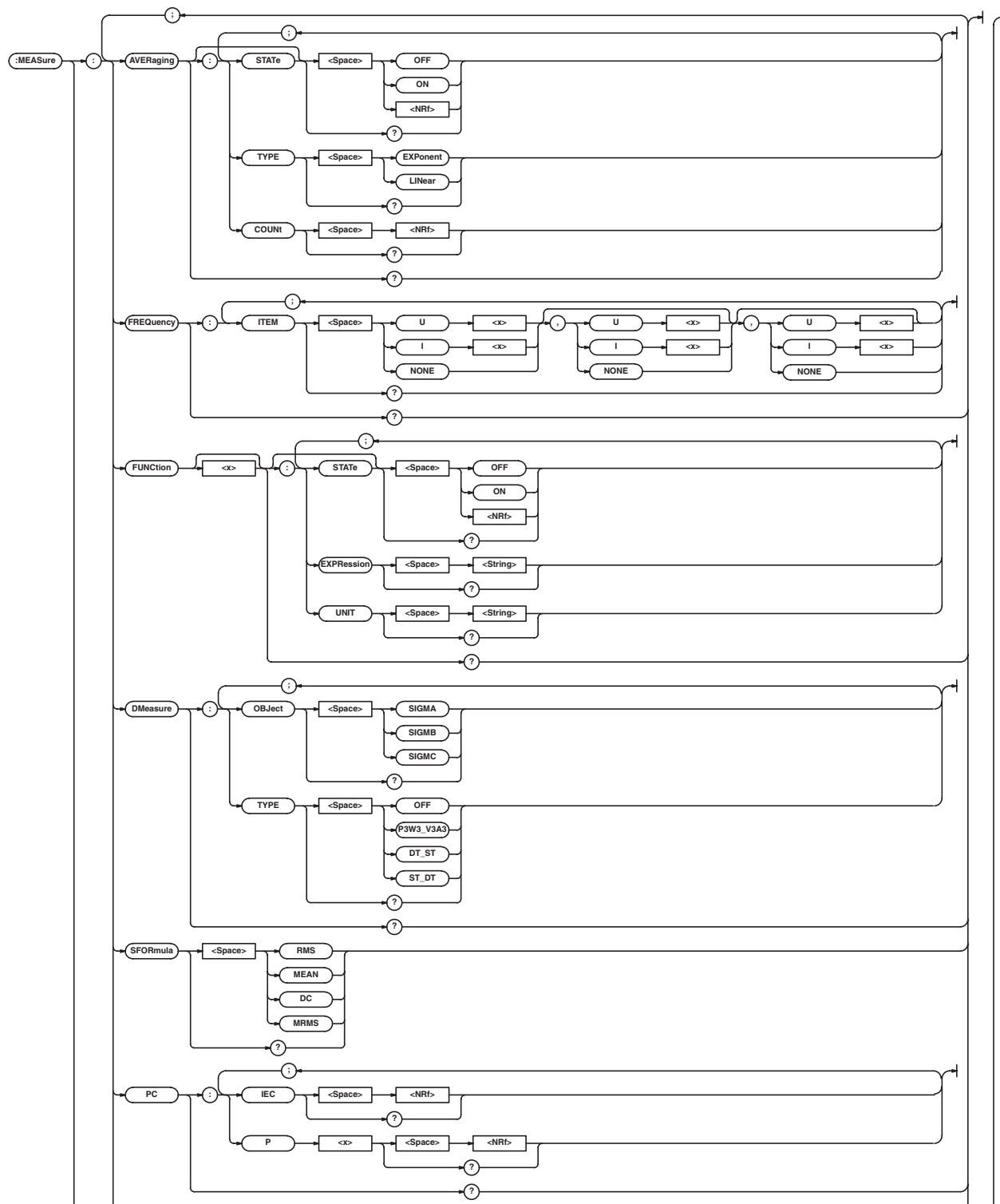
例 :INTEGRATE:TIMER1 1,0,0
:INTEGRATE:TIMER1? -> :INTEGRATE:TIMER1 1,0,0

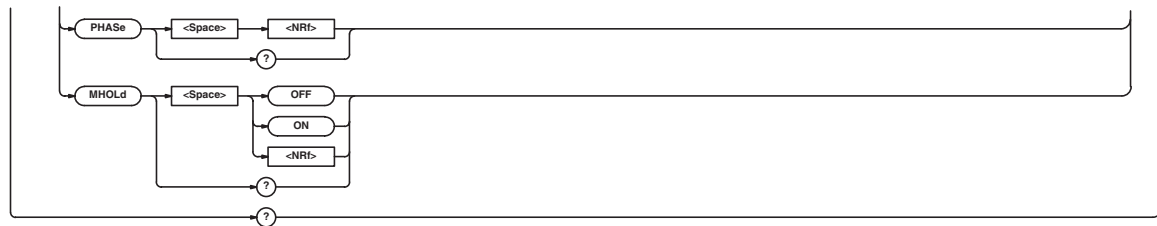
解説 電力測定エレメント別積算(:INTEGrate:INDependent)が「OFF(0)」のときは、電力測定エレメント1の積算タイマ時間の設定により動作します。他のエレメントへの設定/問い合わせは、エラーとなります。

5.12 MEASureグループ

MEASureグループは、測定に関するグループです。

フロントパネルのMEASURE, AVG, MAX HOLD(SHIFT+LOCAL)と同じ設定、および設定内容の問い合わせができます。



**:MEASure?**

機能 測定に関するすべての設定値を問い合わせます。

構文 :MEASure?

例 ・電力測定の場合の例

```
:MEASURE? -> :MEASURE: AVERAGING:
STATE 0; TYPE EXPONENT; COUNT 2;
MEASURE: FREQUENCY: ITEM U1, I1, U2;
MEASURE: FUNCTION1: STATE 0;
EXPRESSION "URMS(E1)"; UNIT "V";
MEASURE: FUNCTION2: STATE 0;
EXPRESSION "IRMS(E1)"; UNIT "A";
MEASURE: FUNCTION3: STATE 0;
EXPRESSION "UPPK(E1)"; UNIT "V";
MEASURE: FUNCTION4: STATE 0;
EXPRESSION "IPPK(E1)"; UNIT "A";
MEASURE: DMEASURE: OBJECT SIGMA;
TYPE OFF; :MEASURE: SFORMULA RMS;
PC: IEC 1976; P1 0.5000; P2 0.5000;
MEASURE: PHASE 180; MHOLD 0
```

・インピーダンス測定の場合の例

```
:MEASURE? -> :MEASURE: AVERAGING:
STATE 0
```

:MEASure: AVERaging?

機能 アベレーシングに関するすべての設定値を問い合わせます。

構文 :MEASure: AVERaging?

例 ・電力測定の場合の例

```
:MEASURE: AVERAGING? -> :MEASURE:
AVERAGING: STATE 1; TYPE EXPONENT;
COUNT 2
```

・インピーダンス測定の場合の例

```
:MEASURE: AVERAGING? -> :MEASURE:
AVERAGING: STATE 1
```

:MEASure: AVERaging: COUNT

機能 電力測定時におけるアベレーシング係数を設定/問い合わせします。

構文 :MEASure: AVERaging: COUNT {<NRf>}

:MEASure: AVERaging: COUNT?

<NRf> = 2, 4, 8, 16, 32, 64

(TYPE = EXPONENTの時)

<NRf> = 8, 16, 32, 64, 128, 256

(TYPE = LINearの時)

例 :MEASURE: AVERAGING: COUNT 2

:MEASURE: AVERAGING: COUNT? ->

:MEASURE: AVERAGING: COUNT 2

解説

- ・電力測定時のみ有効な設定です。インピーダンス測定モード時は設定できません。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。
- ・インピーダンス測定時のアベレーシング係数(減衰定数)については、本体ユーザーズマニュアルを参照してください。

:MEASure: AVERaging[:STATe]

機能 アベレーシングのON/OFFを設定/問い合わせします。

構文 :MEASure: AVERaging

[:STATe] {<Boolean>}

:MEASure: AVERaging: STATe?

例 :MEASURE: AVERAGING: STATE ON

:MEASURE: AVERAGING: STATE? ->

:MEASURE: AVERAGING: STATE 1

解説

インピーダンス測定時のアベレーシングは、ON/OFFの設定のみ可能です。インピーダンス測定時のアベレーシングについては、本体ユーザーズマニュアルを参照してください。

:MEASure:AVERaging:TYPE

機能	電力測定時におけるアベレーシング方式を設定/問い合わせします。
構文	:MEASure:AVERaging:TYPE {EXPonent LINear}
例	:MEASure:AVERaging:TYPE? :MEASURE:AVERAGING:TYPE EXPONENT :MEASURE:AVERAGING:TYPE? -> :MEASURE:AVERAGING:TYPE EXPONENT
解説	- 電力測定時のみ有効な設定です。インピーダンス測定モード時は設定できません。 - インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。 - インピーダンス測定時のアベレーシング方式については、本体ユーザーズマニュアルを参照してください。

:MEASure:DMeasure?

機能	デルタ演算に関するすべての設定値を問い合わせます。
構文	:MEASure:DMeasure?
例	:MEASURE:DMEASURE? -> :MEASURE:DMEASURE:OBJECT SIGMA;TYPE OFF
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:DMeasure:OBject

機能	デルタ演算対象を設定/問い合わせします。
構文	:MEASure:DMeasure:OBject {SIGMA SIGMB SIGMC} :MEASure:DMeasure:OBject? SIGMA = ΣA SIGMB = ΣB(電力測定エレメント2つ以上で選択可) SIGMC = ΣC(電力測定エレメント3つ以上で選択可)
例	:MEASURE:DMEASURE:OBJECT SIGMA :MEASURE:DMEASURE:OBJECT? -> :MEASURE:DMEASURE:OBJECT SIGMA
解説	- 電力測定時のみ有効な設定です。 - インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:DMeasure:TYPE

機能	デルタ演算モードを設定/問い合わせします。
構文	:MEASure:DMeasure:TYPE {OFF P3W3_V3A3 DT_ST ST_DT} :MEASure:DMeasure:TYPE?
例	:MEASURE:DMEASURE:TYPE OFF :MEASURE:DMEASURE:TYPE? -> :MEASURE:DMEASURE:TYPE OFF
解説	- 電力測定時のみ有効な設定です。 - 選択肢の内容はそれぞれ次のとおりです。 OFF = デルタ演算を行わない P3W3_V3A3 = 3P3W -> 3V3A変換 DT_ST = Delta -> Star変換 ST_DT = Star -> Delta変換 - 指定されたデルタ演算対象(:MEASure:DMeasure:OBject)の結線方式によっては、選択できないものがあります。 - インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:FREquency?

機能	周波数測定に関するすべての設定値を問い合わせます。
構文	:MEASure:FREquency?
例	:MEASURE:FREQUENCY? -> :MEASURE:FREQUENCY:ITEM U1,I1,U2
解説	インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:FREquency:ITEM

機能	周波数測定対象を設定/問い合わせします。
構文	:MEASure:FREquency:ITEM {(U<x> I<x> NONE)[,(U<x> I<x> NONE)] [,(U<x> I<x> NONE)]} :MEASure:FREquency:ITEM? <x> = 1 ~ 4(電力測定エレメント)
例	:MEASURE:FREQUENCY:ITEM U1,I1,U2 :MEASURE:FREQUENCY:ITEM? -> :MEASURE:FREQUENCY:ITEM U1,I1,U2
解説	- 電力測定時のみ有効な設定です。 - 周波数測定対象は3つまで指定することができます。 - 周波数測定対象を指定しないときは「NONE」を設定します。2つ目、3つ目については省略することもできます。 - インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:FUNCTION<x>?

機能 ユーザ定義ファンクションに関するすべての設定値を問い合わせます。

構文 :MEASure:FUNCTION<x>?

<x> = 1 ~ 4

例 :MEASURE:FUNCTION1? ->

:MEASURE:FUNCTION1:STATE 1;

EXPRESSION "URMS(E1)";UNIT "V"

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:FUNCTION<x>:EXPRESSION

機能 ユーザ定義ファンクションの演算式を設定/問い合わせします。

構文 :MEASure:FUNCTION<x>:EXPRESSION {<文字列>}

:MEASure:FUNCTION<x>:EXPRESSION?

<x> = 1 ~ 4

<文字列> = 50文字以内

例 :MEASURE:FUNCTION1:

EXPRESSION "URMS(E1)"

:MEASURE:FUNCTION1:EXPRESSION? ->

:MEASURE:FUNCTION1:

EXPRESSION "URMS(E1)"

解説

- ・電力測定時のみ有効な設定です。
- ・本体画面に表示されるキーボード以外の文字や記号は使用できません。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:FUNCTION<x>[:STATE]

機能 ユーザ定義ファンクションの有効(ON)/無効(OFF)を設定/問い合わせします。

構文 :MEASure:FUNCTION<x>

[:STATE] {<Boolean>}

:MEASure:FUNCTION<x>:STATE?

<x> = 1 ~ 4

例 :MEASURE:FUNCTION1:STATE ON

:MEASURE:FUNCTION1:STATE? ->

:MEASURE:FUNCTION1:STATE 1

解説

- ・電力測定時のみ有効な設定です。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:FUNCTION<x>:UNIT

機能 ユーザ定義ファンクションの演算結果に付加する単位を設定/問い合わせします。

構文 :MEASure:FUNCTION<x>:UNIT {<文字列>}

:MEASure:FUNCTION<x>:UNIT?

<x> = 1 ~ 4

<文字列> = 8文字以内

例 :MEASURE:FUNCTION1:UNIT "V"

:MEASURE:FUNCTION1:UNIT? ->

:MEASURE:FUNCTION1:UNIT "V"

解説

- ・電力測定時のみ有効な設定です。
- ・本体画面に表示されるキーボード以外の文字や記号は使用できません。
- ・演算結果に影響を及ぼすことはありません。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:MHOLD

機能 MAX HOLD機能のON/OFFを設定/問い合わせします。

構文 :MEASure:MHOLD {<Boolean>}

:MEASure:MHOLD?

例 :MEASURE:MHOLD ON

:MEASURE:MHOLD? -> :MEASURE:MHOLD 1

解説

- ・電力測定時のみ有効な設定です。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:PC?

機能 Pc(Corrected Power)の算出に関するすべての設定値を問い合わせます。

構文 :MEASure:PC?

例 :MEASURE:PC? -> :MEASURE:PC:

IEC 1976;P1 0.5000;P2 0.5000

解説 インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:PC:IEC

機能 Pc(Corrected Power)の算出式を設定/問い合わせします。

構文 :MEASure:PC:IEC {<Nrf>}

:MEASure:PC:IEC?

<Nrf> = 1976, 1993

例 :MEASURE:PC:IEC 1976

:MEASURE:PC:IEC? -> :MEASURE:PC:

IEC 1976

解説

- ・電力測定時のみ有効な設定です。
- ・IEC76-1で記述されているPc算出式の発行年を指定します。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:PC:P<x>

機能 Pc(Corrected Power)算出のためのパラメータを設定/問い合わせします。

構文 :MEASure:PC:P<x> {<NRf>}
:MEASure:PC:P<x>?
<x> = 1, 2

<NRf> = 0.0000 ~ 9.9999

例 :MEASURE:PC:P1 0.5

:MEASURE:PC:P1? -> :MEASURE:PC:
P1 0.5000

- 解説
- ・電力測定時のみ有効な設定です。
 - ・このパラメータは、「:MEASure:PC:IEC」の設定が「1976(IEC76-1(1976))」のときに使用します。
 - ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:PHASe

機能 位相差の表示形式を設定/問い合わせします。

構文 :MEASure:PHASe {<NRf>}
:MEASure:PHASe?
<NRf> = 180, 360

例 :MEASURE:PHASE 180

:MEASURE:PHASE? -> :MEASURE:
PHASE 180

- 解説
- ・電力測定時のみ有効な設定です。
 - ・「180」のときは $\pm 0 \sim 180^\circ$ (Lead/Lag)で、「360」のときは $0 \sim 360^\circ$ で表示します。
 - ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

:MEASure:SFORMula

機能 S(無効電力)の算出式を設定/問い合わせします。

構文 :MEASure:SFORMula {RMS|MEAN|DC|MRMS}

:MEASure:SFORMula?

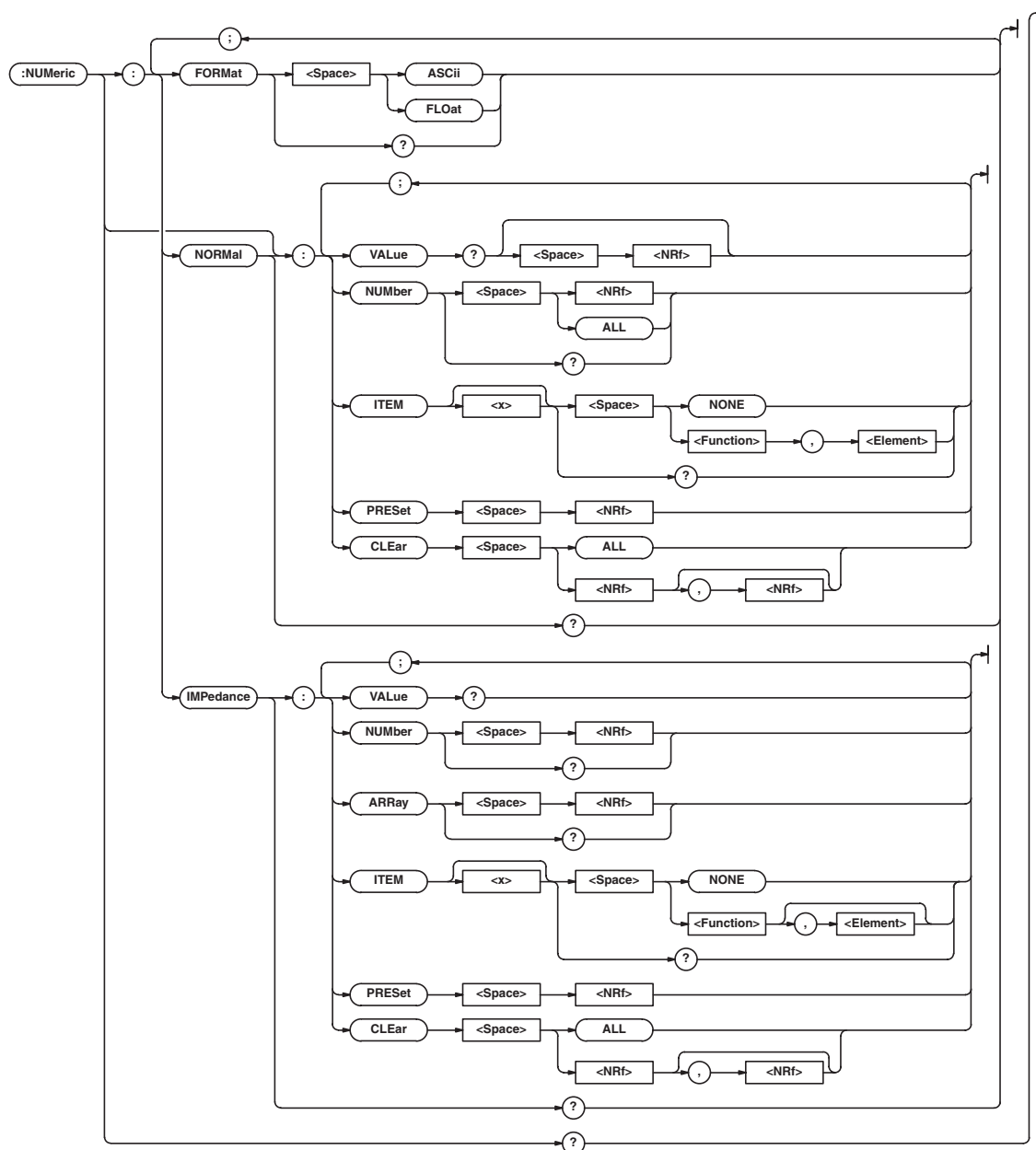
例 :MEASURE:SFORMULA RMS

:MEASURE:SFORMULA? -> :MEASURE:
SFORMULA RMS

- 解説
- ・電力測定時のみ有効な設定です。
 - ・選択肢に対応する算出式はそれぞれ次のとおりです。
RMS : $S = U_{rms} * I_{rms}$
MEAN : $S = U_{mean} * I_{mean}$
DC : $S = U_{dc} * I_{dc}$
MRMS : $S = U_{mean} * I_{rms}$
 - ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

5.13 NUMericグループ

NUMericグループは、数値データの出力に関するグループです。
このグループに相当するフロントパネルのキーはありません。



:NUMeric?

機能 数値データの出力に関するすべての情報を問い合わせます。

構文 :NUMeric?

例 ・電力測定の場合の例

```
:NUMERIC? -> :NUMERIC:
FORMAT ASCII;NORMAL:NUMBER 15;
ITEM1 URMS,1;ITEM2 UMN,1;
ITEM3 UDC,1;ITEM4 UAC,1;
ITEM5 IRMS,1;ITEM6 IMN,1;
ITEM7 IDC,1;ITEM8 IAC,1;
ITEM9 P,1;ITEM10 S,1;ITEM11 Q,1;
ITEM12 LAMBDA,1;ITEM13 PHI,1;
ITEM14 FU,1;ITEM15 FI,1
・インピーダンス測定の場合の例
:NUMERIC? -> :NUMERIC:
FORMAT ASCII;IMPEDANCE:NUMBER 6;
ARRAY 1;ITEM1 BU,4;ITEM2 BI,4;
ITEM3 BP,4;ITEM4 FREQ;ITEM5 ZR,5;
ITEM6 ZI,5
```

:NUMeric:FORMat

機能 「:NUMeric:{NORMAL|IMPedance}:VALue?」で送信される数値データのフォーマットを設定/問い合わせします。

構文 :NUMeric:FORMat {ASCIi|FLOat}

例 :NUMeric:FORMat?

例 :NUMERIC:FORMAT ASCII
:NUMERIC:FORMAT? -> :NUMERIC:
FORMAT ASCII

解説 ・出力される数値データの形式は、
「:NUMeric:FORMat」の設定によって次のように変わります。

- (1)「ASCIi」のとき
物理値を<NR3>形式で出力します。(積算経過時間(TIME)のみ<NR1>形式)
各項目のデータはカンマ(,)で区切られます。
- (2)「FLOat」のとき
数値データブロックの先頭に6バイトのヘッダ(例"#40060")が付きます。
ヘッダに続いて、物理値をIEEE単精度浮動小数点(4byte)形式で連続して出力します。
各項目のデータのバイト出力順序は、MSB Firstです。

・個々の数値データの形式については、このグループの最後にある「数値データのフォーマット」を参照してください。

:NUMeric:IMPedance?

機能 インピーダンス測定時における数値データの出力に関するすべての設定値を問い合わせます。

構文 :NUMeric:IMPedance?

例 :NUMERIC:IMPEDANCE? ->
:NUMERIC:IMPEDANCE:NUMBER 6;
ARRAY 1;ITEM1 BU,4;ITEM2 BI,4;
ITEM3 BP,4;ITEM4 FREQ;ITEM5 ZR,5;
ITEM6 ZI,5

解説 「:NUMeric:IMPedance:ITEM<x>」に関する設定値については、「:NUMeric:IMPedance:NUMBER」で設定されている数だけの数値データ出力項目を出力します。

:NUMeric:IMPedance:ARRay

機能 「:NUMeric:IMPedance:VALue?」で出力される数値データのうち、配列型ファンクション()を出力する際のデータ数(配列数)を設定/問い合わせします。

(配列型ファンクション = {ZR|ZI|Z|PHI|U|I|FREQ})

構文 :NUMeric:IMPedance:ARRay {<NRf>}

:NUMeric:IMPedance:ARRay?

<NRf> = 1 ~ 100

例 :NUMERIC:IMPEDANCE:ARRAY 1
:NUMERIC:IMPEDANCE:ARRAY? ->
:NUMERIC:IMPEDANCE:ARRAY 1

解説 ・この設定により「:NUMeric:IMPedance:VALue?」で1つの配列型ファンクションに対して1~(設定値)の数値データを順に出力します。
・初期設定では、出力データ数(配列数)は「1」が設定されています。

:NUMeric:IMPedance:CLEar

機能 インピーダンス測定時における数値データの出力項目をクリア(「NONE」に設定)します。

構文 :NUMeric:IMPedance:CLEar {ALL|

<NRf>[,<NRf>]}

ALL = すべての項目をクリア

1つ目の<NRf> = 1 ~ 16(クリアを開始する項目番号)

2つ目の<NRf> = 1 ~ 16(クリアを終了する項目番号)

例 :NUMERIC:IMPEDANCE:CLEAR ALL

解説 2つ目の<NRf>を省略した場合、クリア開始番号から最後(16)までの出力項目をクリアします。

:NUMeric:IMPedance:ITEM<x>

機能	電力測定時における数値データ出力項目を設定/問い合わせします。
構文	:NUMeric:IMPedance:ITEM<x> {NONE <Function>[,<Element>]} :NUMeric:IMPedance:ITEM<x>? <x> = 1 ~ 16(項目番号) NONE = 出力項目なし <Function> = {BU BI BP ZR ZI Z PHI U I FREQ} <Element> = {<NRf>}(<NRf> = 1 ~ 5)
例	:NUMERIC:IMPEDANCE:ITEM1 BU,4 :NUMERIC:IMPEDANCE:ITEM1? -> :NUMERIC:IMPEDANCE:ITEM1 BU,4
解説	・ <Function>で指定する項目の内容は、以下のとおりです。 BU = 電池電力測定エレメントの電圧U(dc)(表示: Battery Voltage) BI = 電池電力測定エレメントの電流I(dc)(表示: Battery Current) BP = 電池電力測定エレメントの電力P(dc)(表示: Battery Power) ZR = 回路のインピーダンス実数部(表示: Z') ZI = 回路のインピーダンス虚数部(表示: Z") Z = 回路のインピーダンス絶対値(表示: Z) PHI = 位相差(表示: Φ) U = インピーダンス測定エレメントの電圧(表示: U) I = インピーダンス測定エレメントの電流(表示: I) FREQ = 周波数(表示: Freq) ・ <Function>が「FREQ」の場合、<Element>は省略可能です。

:NUMeric:IMPedance:NUMBER

機能	「:NUMeric:IMPedance:VALue?」で出力される数値データの項目数を設定/問い合わせします。
構文	:NUMeric:IMPedance:NUMBER {<NRf>} :NUMeric:IMPedance:NUMBER? <NRf> = 1 ~ 16
例	:NUMERIC:IMPEDANCE:NUMBER 6 :NUMERIC:IMPEDANCE:NUMBER? -> :NUMERIC:IMPEDANCE:NUMBER 6
解説	・ この設定により「:NUMeric:IMPedance:VALue?」で1~(設定値)の数値データを順に出力します。 ・ 初期設定では、数値データの項目数は「6」が設定されています。

:NUMeric:IMPedance:PRESet

機能	インピーダンス測定時における数値データの出力項目を決められたパターンにプリセットします。
構文	:NUMeric:IMPedance:PRESet {<NRf>} <NRf> = 1 ~ 4(プリセットパターン番号)
例	:NUMERIC:IMPEDANCE:PRESET 1
解説	・ プリセットされる出力項目の内容については、このグループの最後にある「(2) インピーダンス測定数値データ出力項目のプリセットパターン」を参照してください。 ・ 初期設定では、「パターン2」の出力項目が設定されています。

:NUMeric:IMPedance:VALue?

機能	インピーダンス測定時における数値データを問い合わせます。
構文	:NUMeric:IMPedance:VALue?
例	・ 「:NUMeric:FORMat」の設定が「AScii」の場合の例 :NUMERIC:IMPEDANCE:VALUE? -> 1 0 4 . 7 5 E + 0 0 , 1 0 5 . 0 2 E + 0 0 , - 0 . 3 8 E + 0 0 , . . (中略) . . , 4 9 . 8 6 8 E + 0 0 ・ 「:NUMeric:FORMat」の設定が「FLOat」の場合の例 :NUMERIC:IMPEDANCE:VALUE? -> #4(4桁のバイト数)(データバイトの並び)
解説	・ 1 ~ 「:NUMeric:IMPedance:NUMBER」の項目番号の数値データを順に出力します。 ・ 配列型ファンクション({ZR ZI Z PHI U I FREQ})の場合、1つの項目に対して1 ~ 「:NUMeric:IMPedance:ARRay」の配列数の数値データを順に出力します。 ・ 出力される個々の数値データの形式については、このグループの最後にある「数値データのフォーマット」を参照してください。

:NUMeric:NORMAL?

機能	電力測定時における数値データの出力に関するすべての設定値を問い合わせます。
構文	:NUMeric:NORMAL?
例	:NUMERIC:NORMAL? -> :NUMERIC: NORMAL:NUMBER 15;ITEM1 URM,1; ITEM2 UMN,1;ITEM3 UDC,1; ITEM4 UAC,1;ITEM5 IRMS,1; ITEM6 IMN,1;ITEM7 IDC,1; ITEM8 IAC,1;ITEM9 P,1;ITEM10 S,1; ITEM11 Q,1;ITEM12 LAMBDA,1; ITEM13 PHI,1;ITEM14 FU,1; ITEM15 FI,1
解説	「:NUMeric[:NORMAL]:ITEM<x>」に関する設定値については、「:NUMeric[:NORMAL]:NUMBER」で設定されている数だけの数値データ出力項目を出力します。

:NUMeric[:NORMal]:CLEar

機能	電力測定時における数値データの出力項目をクリア(「NONE」に設定)します。
構文	:NUMeric[:NORMal]:CLEar {ALL <NRf>[,<NRf>]} ALL = すべての項目をクリア 1つ目の<NRf> = 1 ~ 255(クリアを開始する項目番号) 2つ目の<NRf> = 1 ~ 255(クリアを終了する項目番号)
例	:NUMERIC:NORMAL:CLEAR ALL
解説	2つ目の<NRf>を省略した場合、クリア開始番号から最後(255)までの出力項目をクリアします。

:NUMeric[:NORMal]:ITEM<x>

機能	電力測定時における数値データ出力項目を設定/問い合わせします。
構文	:NUMeric[:NORMal]:ITEM<x> {NONE <Function>,<Element>} :NUMeric[:NORMal]:ITEM<x>? <x> = 1 ~ 255(項目番号) NONE = 出力項目なし <Function> = {URMS UMN UDC UAC IRMS ...}(「DISPlayグループ」のファンクション選択肢一覧(1)を参照) <Element> = {<NRf> SIGMA SIGMB SIGMC} (<NRf> = 1 ~ 4)(電力測定エレメント)
例	:NUMERIC:NORMAL:ITEM1 URMS,1 :NUMERIC:NORMAL:ITEM1? -> :NUMERIC: NORMAL:ITEM1 URMS,1

:NUMeric[:NORMal]:NUMBER

機能	「:NUMeric:NORMAL:VALue?」で送信される数値データの個数を設定/問い合わせします。
構文	:NUMeric[:NORMal]:NUMBER {<NRf> ALL} :NUMeric[:NORMal]:NUMBER? <NRf> = 1 ~ 255(ALL)
例	:NUMERIC:NORMAL:NUMBER 15 :NUMERIC:NORMAL:NUMBER -> :NUMERIC:NORMAL:NUMBER 15
解説	・「:NUMeric:NORMAL:VALue?」でパラメータを省略すると、1 ~ (設定値)の数値データを順に出力します。 ・初期設定では、数値データの個数は「15」が設定されています。

:NUMeric[:NORMal]:PRESet

機能	電力測定時における数値データの出力項目を決められたパターンにプリセットします。
構文	:NUMeric[:NORMal]:PRESet {<NRf>} <NRf> = 1 ~ 4(プリセットパターン番号)
例	:NUMERIC:NORMAL:PRESET 1
解説	・プリセットされる出力項目の内容については、このグループの最後にある「(1) 電力測定数値データ出力項目のプリセットパターン」を参照してください。 ・初期設定では、「パターン2」の出力項目が設定されています。

:NUMeric[:NORMal]:VALue?

機能	電力測定時における数値データを問い合わせます。
構文	:NUMeric[:NORMal]:VALue? {<NRf>} <NRf> = 1 ~ 255(項目番号)
例	・<NRf>を指定した場合の例 :NUMERIC:NORMAL:VALUE? 1 -> 104.75E+00 ・<NRf>を省略した場合の例 :NUMERIC:NORMAL:VALUE? -> 104.75E+00,105.02E+00,-0.38E+00, ..(中略)..,49.868E+00 ・「:NUMeric:FORMat」の設定が「FLOat」の場合の例 :NUMERIC:NORMAL:VALUE? -> #4(4桁のバイト数)(データバイトの並び)
解説	・<NRf>を指定した場合、その項目番号の数値データのみを出力します。 ・<NRf>を省略した場合、1 ~ 「:NUMeric[:NORMal]:NUMBER」の項目番号の数値データを順に出力します。 ・出力される個々の数値データの形式については、このグループの最後にある「数値データのフォーマット」を参照してください。

数値データのフォーマット

(1) 正常時のデータ

- 電力測定エレメントの180° (Lead/Lag)表示の位相差 ϕ (PHI)
ASCII: 「D/G」+ <NR3>形式(仮数部:有効最大5桁, 指数部:2桁, 例: G90.00E+00)
FLOAT: IEEE単精度浮動小数点(4byte)形式
- 電力値(P, S, Q, PC)のΣ
- 積算値(WH, WHP, WHM, AH, AHP, AHM)
ASCII: <NR3>形式(仮数部:有効最大6桁, 指数部:2桁, 例: [-]123.456E+00)
FLOAT: IEEE単精度浮動小数点(4byte)形式
- 積算経過時間(TIME)
ASCII: 秒単位の<NR1>形式(例: 1時間(1:00:00)の場合, 3600)
FLOAT: IEEE単精度浮動小数点(4byte)形式で秒単位(例: 1時間(1:00:00)の場合, 0x45610000)
- 項目無し(NONE)
ASCII: 「NAN」(Not A Number)
FLOAT: 0x7E951BEE(9.91E+37)
- 上記以外
ASCII: <NR3>形式(仮数部:有効最大5桁, 指数部:2桁, 例: [-]123.45E+00)
FLOAT: IEEE単精度浮動小数点(4byte)形式

(2) 異常時のデータ

- データが存在しない(表示: "-----")
ASCII: 「NAN」(Not A Number)
FLOAT: 0x7E951BEE(9.91E+37)
- オーバレンジ(表示: "---O L---")
- オーバフロー(表示: "---O F---")
- データオーバ(表示: " Error ")
ASCII: 「INF」(INFINITY)
FLOAT: 0x7E94F56A(9.9E+37)

プリセットされる数値データ出力項目一覧

コマンドで使用するファンクション(Function)名と、本機器の画面メニューで使用しているファンクション名の対比リストが、DISPlayグループのファンクション選択肢一覧にあります。

Note

「プリセットされる数値データ出力項目一覧」は、各項目番号(ITEM<x>)に割り当てられている測定ファンクション(Function)とエレメント(Element)を示しています。測定対象に設定されていない項目のところでは、データが存在しないときと同じ表示または出力になります。たとえば、エレメント2の電流の周波数FIが測定対象になっていない場合、項目番号 ITEM19のところでは、データが存在しないときの出力(ASCIIではNAN)になります。

(1) 電力測定数値データ出力項目のプリセットパターン

対象コマンド「:NUMeric[:NORMal]:PRESet」

・パターン1

ITEM<x>	<Function>	<Element>
1	URMS,	1
2	IRMS,	1
3	P,	1
4	S,	1
5	Q,	1
6	LAMBda,	1
7	PHI,	1
8	FU,	1
9	FI,	1
10	NONE	
11 ~ 19	URMS ~ FI,	2
20	NONE	
21 ~ 29	URMS ~ FI,	3
30	NONE	
31 ~ 39	URMS ~ FI,	4
40	NONE	
41 ~ 49	URMS ~ FI,	5
50	NONE	
51 ~ 59	URMS ~ FI,	6
60	NONE	
61 ~ 69	URMS ~ FI,	SIGMA
70	NONE	
71 ~ 79	URMS ~ FI,	SIGMB
80	NONE	
81 ~ 89	URMS ~ FI,	SIGMC
90	NONE	
91 ~ 255	NONE	

・パターン2

ITEM<x>	<Function>	<Element>
1	URMS,	1
2	UMN,	1
3	UDC,	1
4	UAC,	1
5	IRMS,	1
6	IMN,	1
7	IDC,	1
8	IAC,	1
9	P,	1
10	S,	1
11	Q,	1
12	LAMBda,	1
13	PHI,	1
14	FU,	1
15	FI,	1
16 ~ 30	URMS ~ FI,	2
31 ~ 45	URMS ~ FI,	3
46 ~ 60	URMS ~ FI,	4
61 ~ 75	URMS ~ FI,	5
76 ~ 90	URMS ~ FI,	6
91 ~ 105	URMS ~ FI,	SIGMA
106 ~ 120	URMS ~ FI,	SIGMB
121 ~ 135	URMS ~ FI,	SIGMC
136 ~ 255	NONE	

・ パターン3

ITEM<x>	<Function>	<Element>
1	URMS,	1
2	UMN,	1
3	UDC,	1
4	UAC,	1
5	IRMS,	1
6	IMN,	1
7	IDC,	1
8	IAC,	1
9	P,	1
10	S,	1
11	Q,	1
12	LAMBda,	1
13	PHI,	1
14	FU,	1
15	FI,	1
16	UPPeak,	1
17	UMPeak,	1
18	IPPeak,	1
19	IMPeak,	1
20	NONE	
21 ~ 39	URMS ~ IMPeak,	2
40	NONE	
41 ~ 59	URMS ~ IMPeak,	3
60	NONE	
61 ~ 79	URMS ~ IMPeak,	4
80	NONE	
81 ~ 99	URMS ~ IMPeak,	5
100	NONE	
101 ~ 119	URMS ~ IMPeak,	6
120	NONE	
121 ~ 139	URMS ~ IMPeak,	SIGMA
140	NONE	
141 ~ 159	URMS ~ IMPeak,	SIGMB
160	NONE	
161 ~ 179	URMS ~ IMPeak,	SIGMB
180	NONE	
181 ~ 255	NONE	

・ パターン4

ITEM<x>	<Function>	<Element>
1	URMS,	1
2	UMN,	1
3	UDC,	1
4	UAC,	1
5	IRMS,	1
6	IMN,	1
7	IDC,	1
8	IAC,	1
9	P,	1
10	S,	1
11	Q,	1
12	FU,	1
13	FI,	1
14	TIME,	1
15	WH,	1

16	WHP,	1
17	WHM,	1
18	AH,	1
19	AHP,	1
20	AHM,	1
21 ~ 40	URMS ~ AHM,	2
41 ~ 60	URMS ~ AHM,	3
61 ~ 80	URMS ~ AHM,	4
81 ~ 100	URMS ~ AHM,	5
101 ~ 120	URMS ~ AHM,	6
121 ~ 140	URMS ~ AHM,	SIGMA
141 ~ 160	URMS ~ AHM,	SIGMB
161 ~ 180	URMS ~ AHM,	SIGMC
181 ~ 255	NONE	

(2) インピーダンス測定数値データ出力項目のプリセットパターン

対象コマンド「:NUMeric:IMPedance:PRESet」

・ パターン1

ITEM<x>	<Function>	<Element>
1	FREQ,	
2	ZR,	Impedance
3	ZI,	Impedance
4 ~ 16	NONE	

・ パターン2

ITEM<x>	<Function>	<Element>
1	BU,	Battery
2	BI,	Battery
3	BP,	Battery
4	FREQ,	
5	ZR,	Impedance
6	ZI,	Impedance
7 ~ 16	NONE	

・ パターン3

ITEM<x>	<Function>	<Element>
1	BU,	Battery
2	BI,	Battery
3	BP,	Battery
4	FREQ,	
5	U,	Impedance
6	I,	Impedance
7 ~ 16	NONE	

・ パターン4

ITEM<x>	<Function>	<Element>
1	BU,	Battery
2	BI,	Battery
3	BP,	Battery
4	FREQ,	
5	Z,	Impedance
6	PHI,	Impedance
7 ~ 16	NONE	

<Element>の解説

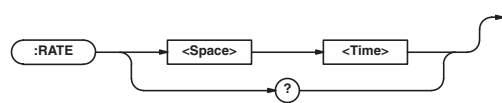
Impedance = インピーダンス測定エレメント(複数ある場合は、小さい番号の方)

Battery = 電池電力測定エレメント

(インピーダンス測定エレメントに隣接している電力測定エレメント1つだけ)

5.14 RATEグループ

RATEグループは、データ更新レートに関するグループです。
フロントパネルのUPDATE RATEと同じ設定、および設定内容の問い合わせができます。



:RATE

機能 電力測定時におけるデータ更新レートを設定/問い合わせします。

構文 :RATE {<時間>}

:RATE?

<時間> = 50, 100, 200, 500(ms), 1, 2, 5 (s)

例 :RATE 200MS

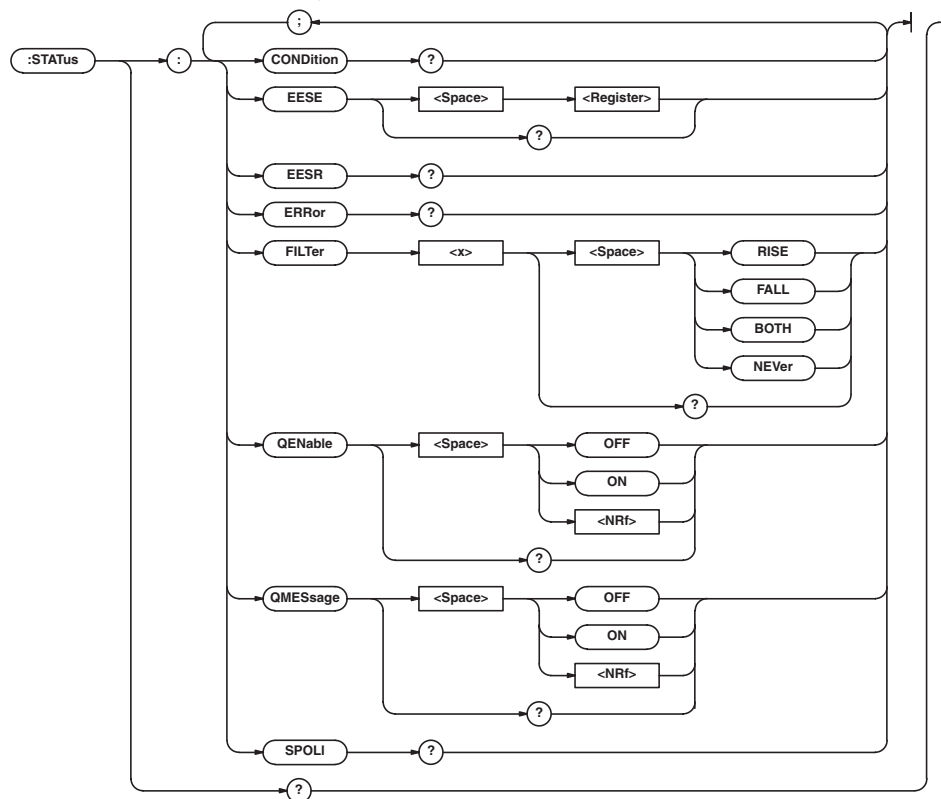
:RATE? -> :RATE 200.0E-03

解説

- ・電力測定時のみ有効な設定です。インピーダンス測定モード時は設定できません。
- ・インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このコマンドは無効です。

5.15 STATus グループ

STATusグループは、ステータスレポートに関する設定と問い合わせを行うグループです。
このグループに相当するフロントパネルのキーはありません。
ステータスレポートについては、6章を参照してください



:STATus?

機能 通信のステータス機能に関するすべての設定値を問い合わせます。

構文 :STATus?

例 :STATus? -> :STATus:EESR 0;FILTer1
NEVER;FILTer2 NEVER;FILTer3 NEVER;
FILTer4 NEVER;FILTer5 NEVER;
FILTer6 NEVER;FILTer7 NEVER;
FILTer8 NEVER;FILTer9 NEVER;
FILTer10 NEVER;FILTer11 NEVER;
FILTer12 NEVER;FILTer13 NEVER;
FILTer14 NEVER;FILTer15 NEVER;
FILTer16 NEVER;QENABLE 0;QMESSAGE 1

:STATus:CONDition?

機能 状態レジスタの内容を問い合わせます。

構文 :STATus:CONDition?

例 :STATus:CONDition? -> 16

解説 :STATus:CONDitionを使った同期のとり方については、4-8ページを参照してください。

:STATus:EESR

(Extended Event Status Enable register)

機能 拡張イベントイネーブルレジスタを設定/問い合わせします。

構文 :STATus:EESR <Register>

:STATus:EESR?

<Register> = 0 ~ 65535

例 :STATus:EESR #B00000000000000000

:STATus:EESR? -> :STATus:EESR 0

:STATus:EESR?

(Extended Event Status Register)

機能 拡張イベントレジスタの内容を問い合わせ、レジスタをクリアします。

構文 :STATus:EESR?

例 :STATus:EESR? -> 0

:STATUS:ERROR?

機能	発生したエラーのコードとメッセージ内容(エラーキューの先頭)を問い合わせます。
構文	:STATUS:ERROR?
例	:STATUS:ERROR? -> 113,"Underfined Header"
解説	・エラーが発生していないときは「0,"No error"」が返されます。 ・メッセージ内容を日本語で返すことはできません。 ・「STATUS:QMESSAGE」で、メッセージ内容を付けるか付けないかを設定できます。

:STATUS:FILTER<x>

機能	遷移フィルタを設定/問い合わせします。
構文	:STATUS:FILTER<x> {RISE FALL BOTH NEVER} :STATUS:FILTER<x>?
例	<x>=1~16 :STATUS:FILTER2 RISE :STATUS:FILTER2? -> :STATUS: FILTER2 RISE
解説	状態レジスタの各ビットがどのように変化したときにイベントをセットするかを決めます。「RISE」なら「0」から「1」になったときにイベントをセットします。

:STATUS:QENABLE

機能	エラー以外のメッセージをエラーキューに格納する(ON)/しない(OFF)を設定/問い合わせします。
構文	:STATUS:QENABLE {<Boolean>} :STATUS:QENABLE?
例	:STATUS:QENABLE ON :STATUS:QENABLE? -> :STATUS: QENABLE 1

:STATUS:QMESSAGE

機能	「STATUS:ERROR?」の応答にメッセージ内容を付ける(ON)/付けない(OFF)を設定/問い合わせします。
構文	:STATUS:QMESSAGE {<Boolean>} :STATUS:QMESSAGE?
例	:STATUS:QMESSAGE ON :STATUS:QMESSAGE? -> :STATUS: QMESSAGE 1

:STATUS:SPOLI? (Serial Poll)

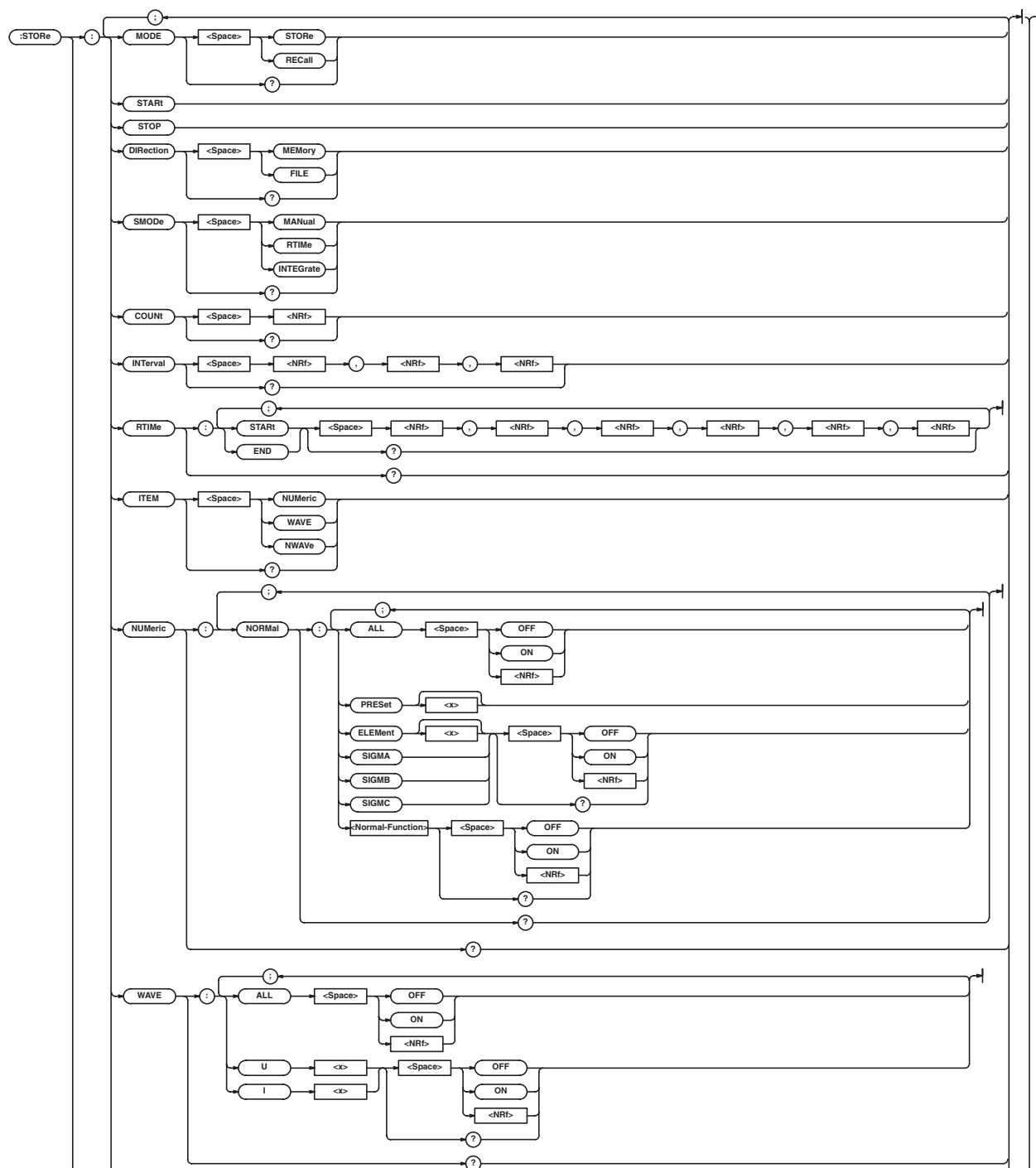
機能	シリアルポールを実行します。
構文	:STATUS:SPOLL?
例	:STATUS:SPOLL? -> :STATUS:SPOLL 0
解説	シリアル(RS-232)インタフェース専用のコマンドです。GP-IBインタフェースの場合、インタフェースメッセージとして用意されています。

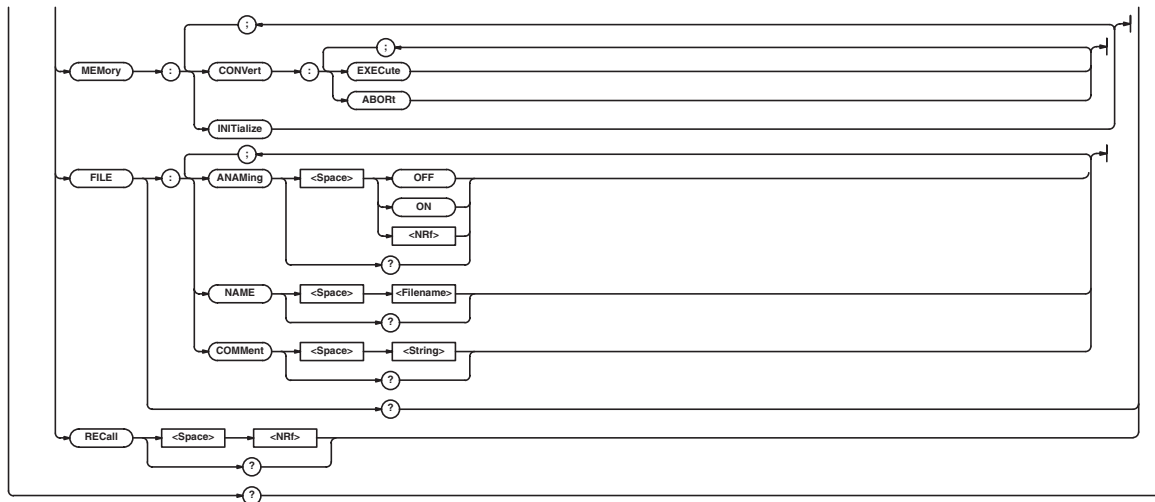
5.16 STOReグループ

STOReグループは、ストア/リコールに関するグループです。

フロントパネルのSTORE，STORE SET(SHIFT+STORE)と同じ設定，および設定内容の問い合わせができます。

ただし、インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このグループのコマンドは無効です。



**:STORe?**

機能 ストア/リコールに関するすべての設定値を問い合わせます。

構文 :STORe?

例 :STORE? -> STORE:MODE STORE;
DIRECTION MEMORY;SMODE MANUAL;
COUNT 100;INTERVAL 0,0,0;
ITEM NUMERIC;NUMERIC:NORMAL:
ELEMENT1 1;ELEMENT2 0;ELEMENT3 0;
ELEMENT4 0;SIGMA 0;SIGMB 0;SIGMC 0;
URMS 1;UMN 1;UDC 1;UAC 1;IRMS 1;
IMN 1;IDC 1;IAC 1;P 1;S 1;Q 1;
LAMBDA 1;PHI 1;FU 1;FI 1;UPPEAK 1;
UMPEAK 1;IPPEAK 1;IMPEAK 1;CFU 1;
CFI 1;FFU 1;FFI 1;Z 1;RS 1;XS 1;
RP 1;XP 1;PC 1;TIME 0;WH 0;WHP 0;
WHM 0;AH 0;AHP 0;AHM 0;ETA 0;
SETA 0;F1 0;F2 0;F3 0;F4 0;DURMS 0;
DUMN 0;DUDC 0;DUAC 0;DIRMS 0;
DIMN 0;DIDC 0;DIAC 0

:STORe:COUNT

機能 ストア回数を設定/問い合わせします。

構文 :STORe:COUNT {<NRf>}

:STORe:COUNT?

<NRf> = 1 ~ 999999

例 :STORE:COUNT 100

:STORE:COUNT? -> :STORE:COUNT 100

:STORe:DIREction

機能 ストア先を設定/問い合わせします。

構文 :STORe:DIREction {MEMory|FILE}

:STORe:DIREction?

例 :STORE:DIRECTION MEMORY

:STORE:DIRECTION? -> :STORE:

DIRECTION MEMORY

:STORe:FILE?

機能 ストアしたデータの保存に関するすべての設定値を問い合わせます。

構文 :STORe:FILE?

例 :STORE:FILE? -> :STORE:FILE:

ANAMING 1;NAME "DATA1";

COMMENT "CASE1"

:STORe:FILE:ANAMing

機能 ストアしたデータを保存するときのファイル名を自動的につけるかどうかを設定/問い合わせします。

構文 :STORe:FILE:ANAMing {<Boolean>}

:STORe:FILE:ANAMing?

例 :STORE:FILE:ANAMING ON

:STORE:FILE:ANAMING? -> :STORE:

FILE:ANAMING 1

:STORe:FILE:COMMeNt

機能 ストアしたデータを保存するときのファイルに付加するコメントを設定/問い合わせします。

構文 :STORe:FILE:COMMeNt {<文字列>}

:STORe:FILE:COMMeNt?

<文字列> = 25文字以内

例 :STORE:FILE:COMMENT "CASE1"

:STORE:FILE:COMMENT? ->

:STORE:FILE:COMMENT "CASE1"

:STORE:FILE:NAME

機能 ストアしたデータを保存するときのファイル名を設定/問い合わせします。

構文 :STORE:FILE:NAME {<Filename>}
:STORE:FILE:NAME?

例 :STORE:FILE:NAME "DATA1"
:STORE:FILE:NAME? -> :STORE:FILE:
NAME "DATA1"

解説 ストアしたデータの保存先は、
・対象ドライブは「:FILE:DRIVE」コマンド
・ディレクトリは「:FILE:CDIRECTORY」コマンド
でそれぞれ設定します。保存先のパスは
「:FILE:PATH?」で問い合わせできます。

:STORE:INTERVAL

機能 ストアインターバルを設定/問い合わせします。

構文 :STORE:INTERVAL {<NRf>,<NRf>,<NRf>}
:STORE:INTERVAL?

1つ目の<NRf> = 0 ~ 99(時間)
2つ目の<NRf> = 0 ~ 59(分)
3つ目の<NRf> = 1 ~ 59(秒)

例 :STORE:INTERVAL 0,0,0
:STORE:INTERVAL? -> :STORE:
INTERVAL 0,0,0

:STORE:ITEM

機能 ストアする内容を設定/問い合わせします。

構文 :STORE:ITEM {NUMERIC|WAVE|NWAVE}
:STORE:ITEM?
NUMERIC = 数値のみストア
WAVE = 波形のみストア
NWAVE = 数値と波形を両方ストア

例 :STORE:ITEM NUMERIC
:STORE:ITEM? -> :STORE:ITEM NUMERIC

:STORE:MEMORY:CONVERT:ABORT

機能 ストアデータのメモリからファイルへのコンバートを中止します。

構文 :STORE:MEMORY:CONVERT:ABORT

例 :STORE:MEMORY:CONVERT:ABORT

:STORE:MEMORY:CONVERT:EXECUTE

機能 ストアデータのメモリからファイルへのコンバートを実行します。

構文 :STORE:MEMORY:CONVERT:EXECUTE

例 :STORE:MEMORY:CONVERT:EXECUTE

解説
・コンバート先のファイルの設定は、
「:STORE:FILE: ...」で行います。
・ファイルコンバートを実行するとき、本機器
はファイルに2回アクセスしています。
ファイルコンバートを実行しその終了を確認
する場合は、「COMMUNICATE:WAIT 64」
コマンド(状態レジスタのビット6(ACS)の変
化をチェック)を使って、WT1600FCのファ
イルアクセス終了を2回確認してください。
下記に例を示します。

```
"STATUS:EESR?"
(拡張イベントレジスタをクリア)
"STORE:MEMORY:CONVERT:EXECUTE"
(ファイルコンバート開始)
"COMMUNICATE:WAIT 64"
(コンバート終了を待つ, 1回目)
"STATUS:EESR?"
(拡張イベントレジスタをクリア)
"COMMUNICATE:WAIT 64"
(コンバート終了を待つ, 2回目)
"STATUS:EESR?"
(拡張イベントレジスタをクリア)
```

:STORE:MEMORY:INITIALIZE

機能 ストアメモリの初期化を実行します。

構文 :STORE:MEMORY:INITIALIZE

例 :STORE:MEMORY:INITIALIZE

:STORE:MODE

機能 データのストア/リコールを設定/問い合わせしま
す。

構文 :STORE:MODE {STORE|RECALL}

:STORE:MODE?

例 :STORE:MODE STORE

:STORE:MODE? -> :STORE:MODE STORE

:STORE:NUMERIC?

機能 数値データのストアに関するすべての設定値を問い合わせます。

構文 :STORE:NUMERIC?

例 :STORE:NUMERIC? -> :STORE:NUMERIC:
NORMAL:ELEMENT1 1;ELEMENT2 0;
ELEMENT3 0;ELEMENT4 0;SIGMA 0;
SIGMB 0;SIGMC 0;URMS 1;UMN 1;UDC 1;
UAC 1;IRMS 1;IMN 1;IDC 1;IAC 1;P 1;
S 1;Q 1;LAMBDA 1;PHI 1;FU 1;FI 1;
UPPEAK 1;UMPEAK 1;IPPEAK 1;
IMPEAK 1;CFU 1;CFI 1;FFU 1;FFI 1;
Z 1;RS 1;XS 1;RP 1;XP 1;PC 1;
TIME 0;WH 0;WHP 0;WHM 0;AH 0;AHP 0;
AHM 0;ETA 0;SETA 0;F1 0;F2 0;F3 0;
F4 0;DURMS 0;DUMN 0;DUDC 0;DUAC 0;
DIRMS 0;DIMN 0;DIDC 0;DIAC 0

:STORE:NUMERIC:NORMAL?

機能 電力測定時における数値データのストアに関するすべての設定値を問い合わせます。

構文 :STORE:NUMERIC:NORMAL?

例 :STORE:NUMERIC:NORMAL? -> :STORE:
NUMERIC:NORMAL:ELEMENT1 1;
ELEMENT2 0;ELEMENT3 0;ELEMENT4 0;
SIGMA 0;SIGMB 0;SIGMC 0;URMS 1;
UMN 1;UDC 1;UAC 1;IRMS 1;IMN 1;
IDC 1;IAC 1;P 1;S 1;Q 1;LAMBDA 1;
PHI 1;FU 1;FI 1;UPPEAK 1;UMPEAK 1;
IPPEAK 1;IMPEAK 1;CFU 1;CFI 1;
FFU 1;FFI 1;Z 1;RS 1;XS 1;RP 1;
XP 1;PC 1;TIME 0;WH 0;WHP 0;WHM 0;
AH 0;AHP 0;AHM 0;ETA 0;SETA 0;F1 0;
F2 0;F3 0;F4 0;DURMS 0;DUMN 0;
DUDC 0;DUAC 0;DIRMS 0;DIMN 0;
DIDC 0;DIAC 0

:STORE:NUMERIC:NORMAL:ALL

機能 電力測定時に数値データをストアするときのすべての電力測定エレメント・ファンクションの出力ON/OFFを一括設定します。

構文 :STORE:NUMERIC:NORMAL:
ALL {<Boolean>}

例 :STORE:NUMERIC:NORMAL:ALL ON

:STORE:NUMERIC:NORMAL:{ELEMENT<x>|SIGMA|SIGMB|SIGMC}

機能 電力測定時に数値データをストアするときの{各電力測定エレメント|ΣA|ΣB|ΣC}の出力ON/OFFを設定/問い合わせします。

構文 :STORE:NUMERIC:NORMAL:{ELEMENT<x>|
SIGMA|SIGMB|SIGMC} {<Boolean>}
:STORE:NUMERIC:NORMAL:{ELEMENT<x>|
SIGMA|SIGMB|SIGMC}?
<x>=1~4(電力測定エレメント)

例 :STORE:NUMERIC:NORMAL:ELEMENTION
:STORE:NUMERIC:NORMAL:ELEMENT1? ->
:STORE:NUMERIC:NORMAL:ELEMENT1 1

解説
・「:STORE:NUMERIC:NORMAL:SIGMB」の設定/問い合わせは、電力測定エレメントが2つ以上のモデルで有効です。
・「:STORE:NUMERIC:NORMAL:SIGMC」の設定/問い合わせは、電力測定エレメントが3つ以上のモデルで有効です。

:STORE:NUMERIC:NORMAL:PRESET<x>

機能 電力測定時に数値データをストアするときの電力測定エレメント・ファンクションの出力ON/OFFを決められたパターンにプリセットします。

構文 :STORE:NUMERIC:NORMAL:PRESET<x>
<x>=1~2(プリセットパターン番号)

例 :STORE:NUMERIC:NORMAL:PRESET1

解説 プリセット実行をしたときのストア設定パターンについては、本体ユーザーズマニュアルを参照してください。

:STORE:NUMERIC:NORMAL:<電力測定ファンクション>

機能 電力測定時に数値データをストアするときの各ファンクションの出力ON/OFFを設定/問い合わせします。

構文 :STORE:NUMERIC:NORMAL:<電力測定ファン
クション> {<Boolean>}
:STORE:NUMERIC:NORMAL:<電力測定ファン
クション>?
<電力測定ファンクション>={URMS|UMN|
UDC|UAC|IRMS|...}(DISPLAYグループのファン
クション選択肢一覧(1)を参照)

例 :STORE:NUMERIC:NORMAL:URMS ON
:STORE:NUMERIC:NORMAL:URMS? ->
:STORE:NUMERIC:NORMAL:URMS 1

:STORE:RECALL

機能 リコールするデータ番号を設定/問い合わせします。

構文 :STORE:RECALL {<Nrf>}
:STORE:RECALL?
<Nrf>=1~999999

例 :STORE:RECALL 1
:STORE:RECALL? -> :STORE:RECALL 1

:STORE:RTIME?

機能 実時間ストアモードにおけるストアスタート/ストップ日時を問い合わせます。

構文 :STORE:RTIME?

例 :STORE:RTIME? -> :STORE:RTIME:START
2001,1,1,0,0,0;END2001,1,1,1,0,0

:STORE:RTIME:{START|END}

機能 実時間ストアモードにおけるストア(スタート|ストップ)日時を設定/問い合わせします。

構文 :STORE:RTIME:{START|END} {<NRf>,<NRf>,<NRf>,<NRf>,<NRf>,<NRf>}
:STORE:RTIME:{START|END}?
{<NRf>,<NRf>,<NRf>,<NRf>,<NRf>,<NRf>} = 2001,1,1,0,0,0 ~ 2099,12,31,23,59,59
1つ目の<NRf> = 2001 ~ 2099 (年)
2つ目の<NRf> = 1 ~ 12 (月)
3つ目の<NRf> = 1 ~ 31 (日)
4つ目の<NRf> = 0 ~ 23 (時)
5つ目の<NRf> = 0 ~ 59 (分)
6つ目の<NRf> = 0 ~ 59 (秒)

例 :STORE:RTIME:START 2001,1,1,0,0,0
:STORE:RTIME:START? -> :STORE:RTIME:START2001,1,1,0,0,0

:STORE:SMODE

機能 スタアモードを設定/問い合わせします。

構文 :STORE:SMODE {MANual|RTIME|INTEGrate}
:STORE:SMODE?
MANual = マニュアルストアモード
RTIME = 実時間ストアモード
INTEGrate = 積算同期ストアモード

例 :STORE:SMODE MANUAL
:STORE:SMODE? -> :STORE:SMODE MANUAL

:STORE:START

機能 データのストアを開始します。

構文 :STORE:START

例 :STORE:START

解説 「:STORE:SMODE」がMANualのときはストアを実行します。{RTIME|INTEGrate}のときはストア待機状態になります。

:STORE:STOP

機能 データのストアを終了します。

構文 :STORE:STOP

例 :STORE:STOP

:STORE:WAVE?

機能 波形表示データのストアに関するすべての設定値を問い合わせます。

構文 :STORE:WAVE?

例 :STORE:WAVE? -> :STORE:WAVE:U1 1;
U2 0;U3 0;U4 0;I1 1;I2 0;I3 0;I4 0

:STORE:WAVE:ALL

機能 波形表示データをストアするときのすべての波形の出力ON/OFFを一括設定します。

構文 :STORE:WAVE:ALL {<Boolean>}

例 :STORE:WAVE:ALL ON

:STORE:WAVE:{U<x>|I<x>}

機能 波形表示データをストアするときの各波形の出力ON/OFFを設定/問い合わせします。

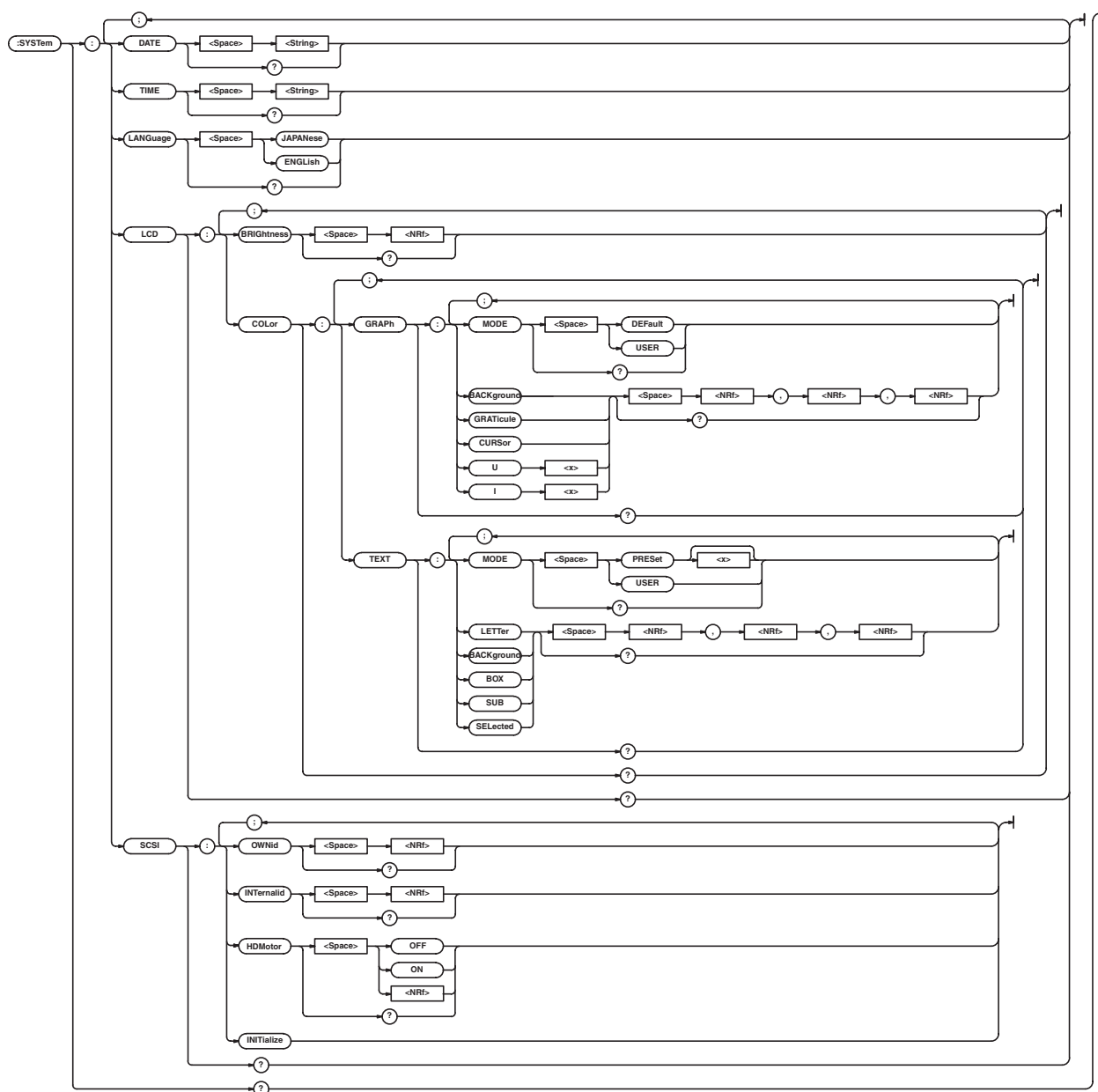
構文 :STORE:WAVE:{U<x>|I<x>} {<Boolean>}
:STORE:WAVE:{U<x>|I<x>}?

<x> = 1 ~ 4(電力測定エレメント)

例 :STORE:WAVE:U1 ON
:STORE:WAVE:U1? -> :STORE:WAVE:U1 1

5.17 SYSTemグループ

SYSTemグループは、システムに関するグループです。
フロントパネルのMISCと同じ設定、および設定内容の問い合わせができます。



:SYSTem?

機能 システムに関するすべての設定値を問い合わせます。

構文 :SYSTem?

例 :SYSTem? -> :SYSTem:
LANGUAGE ENGLISH;LCD:BRIGHTNESS 2;
COLOR:GRAPH:MODEDEFAULT;:SYSTem:
LCD:COLOR:TEXT:MODEPRESET1;:
SYSTem:SCSI:OWNID 6;INTERNALID 4;
HDMOTOR 1

:SYSTem:DATE

機能 日付を設定/問い合わせします。

構文 :SYSTem:DATE {<文字列>}

:SYSTem:DATE?

<文字列> = "YY/MM/DD"(YY = 年, MM = 月, DD = 日)

例 :SYSTem:DATE "00/01/01"

:SYSTem:DATE? -> "00/01/01"

解説 「年」は西暦の下2桁です。

:SYSTem:LANGUage

機能 メッセージの言語を設定/問い合わせします。

構文 :SYSTem:LANGUage {JAPANEse|ENGLish}
:SYSTem:LANGUage?

例 :SYSTem:LANGUage ENGLISH
:SYSTem:LANGUage? -> :SYSTem:
LANGUage ENGLISH

:SYSTem:LCD?

機能 LCD画面に関するすべての設定値を問い合わせます。

構文 :SYSTem:LCD?

例 :SYSTem:LCD? -> :SYSTem:LCD:
BRIGHTNESS 2;COLOR:GRAPH:
MODE DEFAULT;:SYSTem:LCD:COLOR:
TEXT:MODE PRESET1

:SYSTem:LCD:BRIGhtness

機能 LCD画面の輝度を設定/問い合わせします。

構文 :SYSTem:LCD:BRIGhtness {<NRf>}
:SYSTem:LCD:BRIGhtness?
<NRf> = - 1 ~ 3

例 :SYSTem:LCD:BRIGhtness 2
:SYSTem:LCD:BRIGhtness? -> :SYSTem:
LCD:BRIGhtness 2

:SYSTem:LCD:COLor?

機能 LCD画面の表示色に関するすべての設定値を問い合わせます。

構文 :SYSTem:LCD:COLor?

例 :SYSTem:LCD:COLor? -> :SYSTem:LCD:
COLOR:GRAPH:MODE DEFAULT;:SYSTem:
LCD:COLOR:TEXT:MODE PRESET1

:SYSTem:LCD:COLor:GRAPH?

機能 グラフィック項目の表示色に関するすべての設定値を問い合わせます。

構文 :SYSTem:LCD:COLor:GRAPH?

例 :SYSTem:LCD:COLor:GRAPH? ->
:SYSTem:LCD:COLor:GRAPH:MODE USER;
BACKGROUND 0,0,0;GRATICULE 6,6,6;
CURSOR 7,7,7;U1 7,7,0;U2 7,0,7;
U3 7,0,0;U4 0,4,7;I1 0,7,0;
I2 0,7,7;I3 7,4,0;I4 5,5,5

:SYSTem:LCD:COLor:GRAPH: {BACKground|GRATICule|CURSor|U<x>|I<x>}

機能 {背景|目盛り|カーソル|電圧波形|電流波形}の表示色を設定/問い合わせします。

構文 :SYSTem:LCD:COLor:GRAPH:
{BACKground|GRATICule|CURSor|U<x>|
I<x>} {<NRf>,<NRf>,<NRf>}
:SYSTem:LCD:COLor:GRAPH:
{BACKground|GRATICule|CURSor|U<x>|
I<x>}?

<x> = 1 ~ 4(電力測定エレメント)
<NRf> = 0 ~ 7

例 :SYSTem:LCD:COLor:GRAPH:
BACKGROUND 0,0,0
:SYSTem:LCD:COLor:GRAPH:
BACKGROUND? -> :SYSTem:LCD:COLor:
GRAPH:BACKGROUND 0,0,0
R, G, Bの順で設定します。

解説 グラフィック項目の表示色のモード(:SYSTem:LCD:COLor:GRAPH:MODE)が「USER」のときに有効な設定です。

:SYSTem:LCD:COLor:GRAPH:MODE

機能 グラフィック項目の表示色のモードを設定/問い合わせします。

構文 :SYSTem:LCD:COLor:GRAPH:
MODE {DEFault|USER}
:SYSTem:LCD:COLor:GRAPH:MODE?

例 :SYSTem:LCD:COLor:GRAPH:
MODE DEFAULT
:SYSTem:LCD:COLor:GRAPH:MODE? ->
:SYSTem:LCD:COLor:GRAPH:
MODE DEFAULT

:SYSTem:LCD:COLor:TEXT?

機能 テキスト項目の表示色に関するすべての設定値を問い合わせます。

構文 :SYSTem:LCD:COLor:TEXT?

例 :SYSTem:LCD:COLor:TEXT? -> :SYSTem:
LCD:COLOR:TEXT:MODE USER;
LETTER 7,7,7;BACKGROUND 2,2,6;
BOX 0,0,7;SUB 3,3,3;SELECTED 0,4,7

:SYSTem:LCD:COLor:TEXT:{LETter|BACKground|BOX|SUB|SELEcted}

機能	{文字(Menu Fore) メニューの背景(Menu Back) 選択されているメニュー(Select Box) ポップアップメニュー(Sub Menu) 選択されているキー(Selected Key)}の表示色を設定/問い合わせします。
構文	:SYSTem:LCD:COLor:TEXT:{LETter BACKground BOX SUB SELEcted} {<Nrf>,<Nrf>,<Nrf>} :SYSTem:LCD:COLor:TEXT:{LETter BACKground BOX SUB SELEcted}? <Nrf> = 0 ~ 7
例	:SYSTEM:LCD:COLOR:TEXT:LETTER 7,7,7 :SYSTEM:LCD:COLOR:TEXT:LETTER? -> :SYSTEM:LCD:COLOR:TEXT:LETTER 7,7,7
解説	R, G, Bの順で設定します。 テキスト項目の表示色のモード (:SYSTem:LCD:COLor:TEXT:MODE)が「USER」のときに有効な設定です。

:SYSTem:LCD:COLor:TEXT:MODE

機能	テキスト項目の表示色のモードを設定/問い合わせします。
構文	:SYSTem:LCD:COLor:TEXT:MODE {PRESet<x> USER} :SYSTem:LCD:COLor:TEXT:MODE? <x> = 1 ~ 3
例	:SYSTEM:LCD:COLOR:TEXT:MODE PRESET1 :SYSTEM:LCD:COLOR:TEXT:MODE? -> :SYSTEM:LCD:COLOR:TEXT:MODE PRESET1

:SYSTem:SCSI?

機能	SCSI-IDに関するすべての設定値を問い合わせます。
構文	:SYSTem:SCSI?
例	:SYSTEM:SCSI? -> :SYSTEM:SCSI:OWNID 6;INTERNALID 4;HDMOTOR 1
解説	SCSIインタフェース(オプション)を装着していない時は、エラーになります。

:SYSTem:SCSI:HDMotor

機能	内蔵ハードディスクのモータのON/OFFを設定/問い合わせします。
構文	:SYSTem:SCSI:HDMotor {<Boolean>} :SYSTem:SCSI:HDMotor?
例	:SYSTEM:SCSI:HDMOTOR ON :SYSTEM:SCSI:HDMOTOR? -> :SYSTEM:SCSI:HDMOTOR 1
解説	SCSIインタフェース(オプション)を装着していない時は、エラーになります。

:SYSTem:SCSI:INITialize

機能	SCSI関連のイニシャライズを実行します。
構文	:SYSTem:SCSI:INITialize
例	:SYSTEM:SCSI:INITIALIZE
解説	・SCSIインタフェース(オプション)を装着していない時は、エラーになります。 ・「:SYSTem:SCSI:OWNid」で本機器のSCSI-IDを変更した場合には、必ずこのコマンドを実行してください。

:SYSTem:SCSI:INTernalid

機能	内蔵ハードディスクのSCSI-IDを設定/問い合わせします。
構文	:SYSTem:SCSI:INTernalid {<Nrf>} :SYSTem:SCSI:INTernalid? <Nrf> = 4 (固定)
例	:SYSTEM:SCSI:INTERNALID 4 :SYSTEM:SCSI:INTERNALID? -> :SYSTEM:SCSI:INTERNALID 4
解説	SCSIインタフェース(オプション)を装着していない時は、エラーになります。

:SYSTem:SCSI:OWNid

機能	本機器のSCSI-IDを設定/問い合わせします。
構文	:SYSTem:SCSI:OWNid {<Nrf>} :SYSTem:SCSI:OWNid? <Nrf> = 0 ~ 7
例	:SYSTEM:SCSI:OWNID 6 :SYSTEM:SCSI:OWNID? -> :SYSTEM:SCSI:OWNID 6
解説	SCSIインタフェース(オプション)を装着していない時は、エラーになります。

:SYSTem:TIME

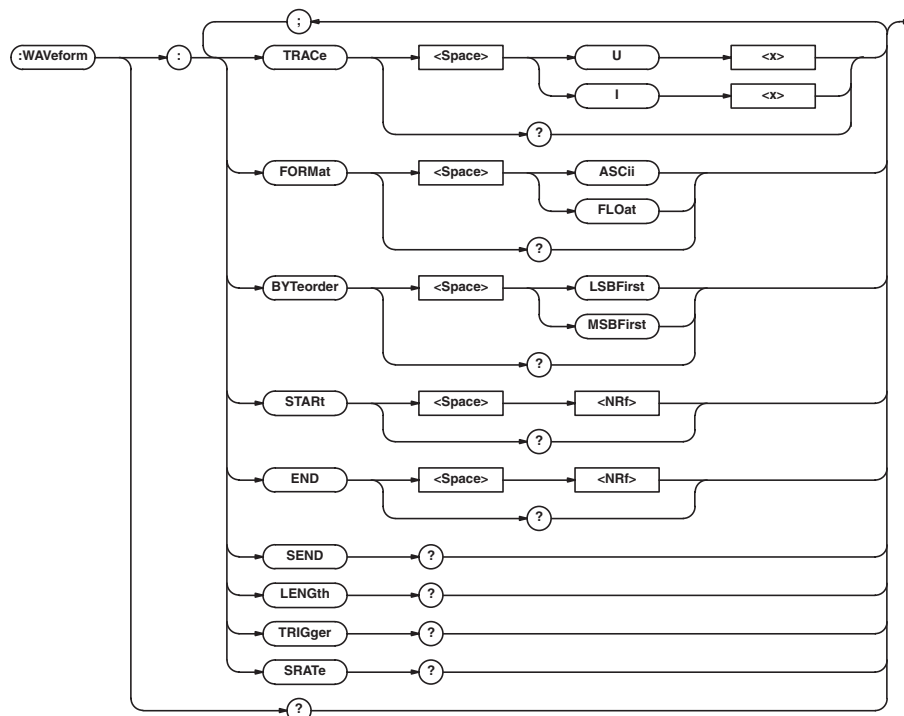
機能	時刻を設定/問い合わせします。
構文	:SYSTem:TIME {<文字列>} :SYSTem:TIME? <文字列> = "HH:MM:SS"(HH = 時間, MM = 分, SS = 秒)
例	:SYSTEM:TIME "14:30:00" :SYSTEM:TIME? -> "14:30:00"

5.18 WAVEformグループ

WAVEformグループは、取り込んだ波形表示データの出力に関するグループです。

このグループに相当するフロントパネルのキーはありません。

ただし、インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このグループのコマンドは無効です。



:WAVEform?

機能 波形表示データのすべての情報を問い合わせます。

構文 :WAVEform?

例 :WAVEFORM? -> :WAVEFORM:TRACE U1;
FORMAT ASCII;START 0;END 1001

:WAVEform:BYTeorder

機能 「:WAVEform:SEND?」で送信される波形表示データ(FLOAT形式)のバイト出力順序を設定/問い合わせします。

構文 :WAVEform:BYTeorder {LSBFirst|MSBFirst}

:WAVEform:BYTeorder?

例 :WAVEFORM:BYTEORDER LSBFIRST
:WAVEFORM:BYTEORDER? -> :WAVEFORM:BYTEORDER LSBFIRST

解説 この設定値は、「:WAVEform:FORMat」が{FLOat}のときに有効です。

:WAVEform:END

機能 「:WAVEform:SEND?」で送信される波形表示データの出力終了点を設定/問い合わせします。

構文 :WAVEform:END {<NRf>}

:WAVEform:END?

<NRf> = 0 ~ (全データ点数-1)

例 :WAVEFORM:END 1001

:WAVEFORM:END? -> :WAVEFORM:END 1001

解説 (全データ点数)は、「:WAVEform:LENGth?」で問い合わせることができます。

:WAVEform:FORMat

機能 「:WAVEform:SEND?」で送信される波形表示データのフォーマットを設定/問い合わせします。

構文 :WAVEform:FORMat {AScii|FLOat}
:WAVEform:FORMat?

例 :WAVEFORM:FORMAT FLOAT
:WAVEFORM:FORMAT? -> :WAVEFORM:FORMAT FLOAT

解説 フォーマットの設定による波形表示データ出力の違いについては、「:WAVEform:SEND?」の解説を参照してください。

:WAVEform:LENGth?

機能 「:WAVEform:TRACE」で指定した波形の全データ点数を問い合わせます。

構文 :WAVEform:LENGth?

例 :WAVEFORM:LENGTH? -> 1002

解説 データ点数は固定です。常に「1002」を返します。

:WAVEform:SEND?

機能 「:WAVEform:TRACE」で指定した波形表示データを問い合わせます。

構文 :WAVEform:SEND?

例 ・「:WAVEform:FORMat」が{AScii}のとき
:WAVEFORM:SEND? ->
<NR3>,<NR3>,...
・「:WAVEform:FORMat」が{FLOat}のとき
:WAVEFORM:SEND? -> #4(4桁のバイト数)
(データバイトの並び)

解説 出力される波形表示データの形式は、「:WAVEform:FORMat」の設定によって次のようになります。

(1)「AScii」のとき
物理値を<NR3>形式で出力します。各点のデータはカンマ(,)で区切られます。

(2)「FLOat」のとき
物理値をIEEE単精度浮動小数点(4byte)形式で出力します。
各点のデータのバイト出力順序は、「:WAVEform:BYTeorder」の設定に従います。

:WAVEform:SRATE?

機能 取り込んだ波形のサンプルレートを問い合わせます。

構文 :WAVEform:SRATE?

例 :WAVEFORM:SRATE? -> 200.000E+03

:WAVEform:START

機能 「:WAVEform:SEND?」で送信される波形表示データの出力開始点を設定/問い合わせします。

構文 :WAVEform:START {<NRf>}
:WAVEform:START?

例 <NRf> = 0 ~ (全データ点数 - 1)
:WAVEFORM:START 0
:WAVEFORM:START? -> :WAVEFORM:START 0

解説 (全データ点数)は、「:WAVEform:LENGth?」で問い合わせることができます。

:WAVEform:TRACe

機能 「WAVEform:LENGth?」や「WAVEform:SEND?」コマンドの対象となる波形を設定/問い合わせします。

構文 :WAVEform:TRACe {U<x>|I<x>}
:WAVEform:TRACe?

例 <x> = 1 ~ 4(電力測定エレメント)
:WAVEFORM:TRACE U1
:WAVEFORM:TRACE? -> :WAVEFORM:TRACE U1

:WAVEform:TRIGger?

機能 取り込んだ波形のトリガポジションを問い合わせます。

構文 :WAVEform:TRIGger?

例 :WAVEFORM:TRIGGER? -> 0

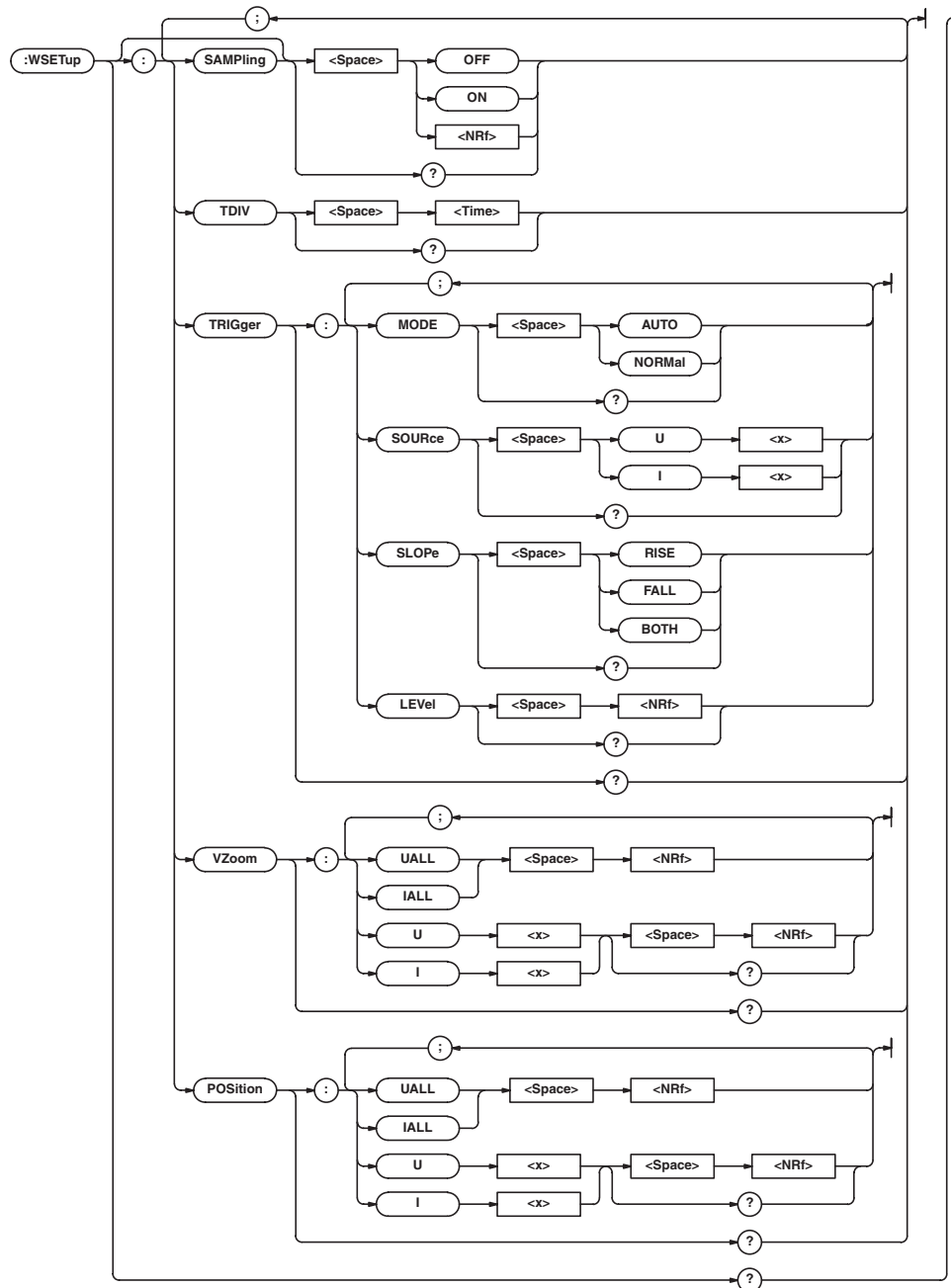
解説 トリガポジションは常に波形表示データの先頭になるので、「0」を返します。

5.19 WSETup (Wave SETup)グループ

WSETupグループは、波形観測に関するグループです。

フロントパネルのWAVEと同じ設定、および設定内容の問い合わせができます。

ただし、インピーダンス測定エレメントだけが装備されている製品では、インピーダンス測定モードに固定のため、このグループのコマンドは無効です。



:WSETup?

機能 波形観測に関するすべての設定値を問い合わせます。

構文 :WSETup?

例 :WSETUP? -> :WSETUP:SAMPLING 0;
TDIV 500.0E-06;TRIGGER:MODE AUTO;
SOURCE U1;SLOPE RISE;LEVEL 0.0;;
WSETUP:VZOOM:U1 1.00;U2 1.00;
U3 1.00;U4 1.00;I1 1.00;I2 1.00;
I3 1.00;I4 1.00;;WSETUP:POSITION:
U1 0.000;U2 0.000;U3 0.000;
U4 0.000;I1 0.000;I2 0.000;
I3 0.000;I4 0.000

:WSETup:POSition?

機能 波形の垂直ポジション(GND位置)に関するすべての設定値を問い合わせます。

構文 :WSETup:POSition?

例 :WSETUP:POSITION? -> :WSETUP:
POSITION:U1 0.000;U2 0.000;
U3 0.000;U4 0.000;I1 0.000;
I2 0.000;I3 0.000;I4 0.000

:WSETup:POSition:{UALL| IALL}

機能 すべての電力測定エレメントの{電圧|電流}波形の垂直ポジション(中心位置のレベル)を一括設定します。

構文 :WSETup:POSition:{UALL|
IALL} {<NRf>}
<NRf> = - 130.000 ~ 130.000(%)

例 :WSETUP:POSITION:UALL 0

:WSETup:POSition:{U<x>| I<x>}

機能 各電力測定エレメントの{電圧|電流}波形の垂直ポジション(中心位置のレベル)を設定/問い合わせします。

構文 :WSETup:POSition:{U<x>|
I<x>} {<NRf>}
:WSETup:POSition:{U<x>| I<x>}?
<x> = 1 ~ 4(電力測定エレメント)
<NRf> = - 130.000 ~ 130.000(%)

例 :WSETUP:POSITION:U1 0
:WSETUP:POSITION:U1? -> :WSETUP:
POSITION:U1 0.000

:WSETup[:SAMPling]

機能 波形サンプリングのON/OFFを設定/問い合わせします。

構文 :WSETup[:SAMPling] {<Boolean>}
:WSETup:SAMPling?

例 :WSETUP:SAMPLING ON
:WSETUP:SAMPLING? -> :WSETUP:
SAMPLING 1

:WSETup:TDIV

機能 波形のTime/div値を設定/問い合わせします。

構文 :WSETup:TDIV {<時間>}

:WSETup:TDIV?
<時間> = 0.5, 1, 2, 5, 10, 20, 50, 100,
200, 500(ms)

例 :WSETUP:TDIV 0.5MS

:WSETup:TDIV? -> :WSETUP:
TDIV 500.0E-06

解説 設定することのできるTime/div値は、データ更新周期(:RATE)の1/10までです。

:WSETup:TRIGger?

機能 トリガに関するすべての設定値を問い合わせます。

構文 :WSETup:TRIGger?

例 :WSETUP:TRIGGER? ->
:WSETUP:TRIGGER:MODE AUTO;
SOURCE U1;SLOPE RISE;LEVEL 0.0

:WSETup:TRIGger:LEVel

機能 トリガレベルを設定/問い合わせします。

構文 :WSETup:TRIGger:LEVel {<NRf>}
:WSETup:TRIGger:LEVel?
<NRf> = - 100.0 ~ 100.0(%) (設定分解能は0.1
(%))

例 :WSETUP:TRIGGER:LEVEL 0
:WSETUP:TRIGGER:LEVEL? -> :WSETUP:
TRIGGER:LEVEL 0.0

解説 画面表示されているフルスケール値に対する%で設定します。

:WSETup:TRIGger:MODE

機能 トリガモードを設定/問い合わせします。

構文 :WSETup:TRIGger:MODE {AUTO|NORMal}
:WSETup:TRIGger:MODE?

例 :WSETUP:TRIGGER:MODE AUTO
:WSETUP:TRIGGER:MODE? -> :WSETUP:
TRIGGER:MODE AUTO

:WSETup:TRIGger:SLOPe

機能 トリガスロープを設定/問い合わせします。

構文 :WSETup:TRIGger:SLOPe {RISE|FALL|
BOTH}
:WSETup:TRIGger:SLOPe?

例 :WSETUP:TRIGGER:SLOPE RISE
:WSETUP:TRIGGER:SLOPE? -> :WSETUP:
TRIGGER:SLOPE RISE

:WSETup:TRIGger:SOURce

機能 トリガソースを設定/問い合わせします。

構文 :WSETup:TRIGger:SOURce {U<x>|I<x>}
:WSETup:TRIGger:SOURce?
<x> = 1 ~ 4(電力測定エレメント)

例 :WSETUP:TRIGGER:SOURCE U1
:WSETUP:TRIGGER:SOURCE? -> :WSETUP:
TRIGGER:SOURCE U1

:WSETup:VZoom?

機能 波形の垂直方向のズーム率に関するすべての設定値を問い合わせます。

構文 :WSETup:VZoom?

例 :WSETUP:VZOOM? -> :WSETUP:VZOOM:
U1 1.00;U2 1.00;U3 1.00;U4 1.00;
I1 1.00;I2 1.00;I3 1.00;I4 1.00

:WSETup:VZoom:{UALL|IALL}

機能 すべての電力測定エレメントの{電圧|電流}波形の垂直方向のズーム率を一括設定します。

構文 :WSETup:VZoom:{UALL|IALL} {<NRf>}
<NRf> = 0.1 ~ 100(本体ユーザーズマニュアル参照)

例 :WSETUP:VZOOM:UALL 1

:WSETup:VZoom:{U<x>|I<x>}

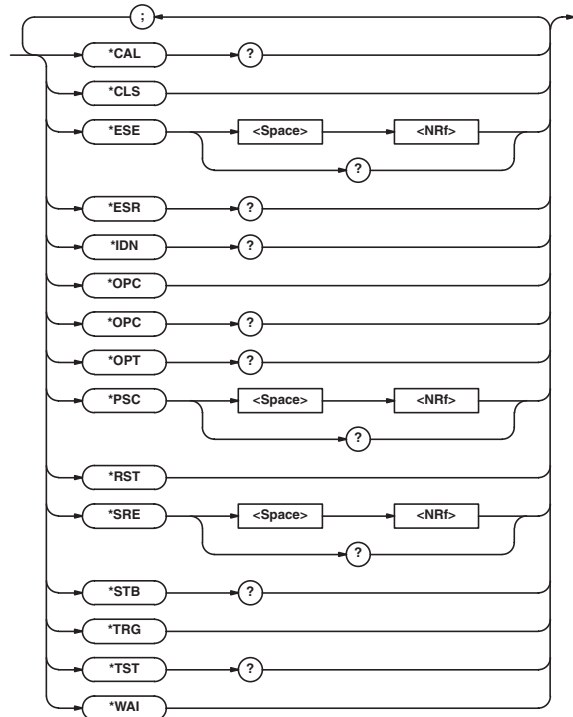
機能 各電力測定エレメントの{電圧|電流}波形の垂直方向のズーム率を設定/問い合わせします。

構文 :WSETup:VZoom:{U<x>|I<x>} {<NRf>}
:WSETup:VZoom:{U<x>|I<x>}?
<x> = 1 ~ 4(電力測定エレメント)
<NRf> = 0.1 ~ 100(本体ユーザーズマニュアル参照)

例 :WSETUP:VZOOM:U1 1
:WSETUP:VZOOM:U1? -> :WSETUP:VZOOM:
U1 1.00

5.20 共通コマンドグループ

共通コマンドグループは、IEEE488.2-1987で規定されている，機種固有の機能に依存しないコマンドのグループです。このグループに相当するフロントパネルのキーはありません。



*CAL? (CALibrate)

機能 キャリブレーション(ゼロレベル補正，CAL (SHIFT+MEASURE)を押したのと同じ動作)を実行し，結果を問い合わせます。

構文 *CAL?

例 *CAL? -> 0

解説 キャリブレーションが正常に終了したときは「0」，異常があるときは「1」が返されます。

*CLS (Clear Status)

機能 標準イベントレジスタ，拡張イベントレジスタ，エラーキューをクリアします。

構文 *CLS

例 *CLS

解説

- ・*CLSコマンドがプログラムメッセージターミネータのすぐ後ろにあるときは，出力キューもクリアされます。
- ・各レジスタ，キューについては，6章を参照してください。

*ESE

(standard Event Status Enable register)

機能 標準イベントイネーブルレジスタの値を設定/問い合わせします。

構文 *ESE {<NRf>}

例 *ESE 251

解説

- ・各ビットの10進数の和で設定します。
- ・たとえば，「*ESE 251」とすると，標準イベントイネーブルレジスタを「11111011」にセットします。つまり，標準イベントレジスタのビット2を無効にし，「問い合わせエラー」が起こってもステータスバイトレジスタのビット5(ESB)を「1」にしません。
- ・初期値は「*ESE 0」(全ビット無効)です。
- ・*ESE?で問い合わせても，標準イベントイネーブルレジスタの内容はクリアされません。
- ・標準イベントイネーブルレジスタについては，6-3ページを参照してください。

***ESR? (standard Event Status Register)**

機能	標準イベントレジスタの値を問い合わせ、同時にクリアします。
構文	*ESR?
例	*ESR? -> 32
解説	- 各ビットの10進数の和が返されます。 - SRQが発生しているときに、どんな種類のイベントが起こったかを調べることができます。 - たとえば、「32」が返されると、標準イベントレジスタが「00100000」にセットされていることを示します。つまり、「コマンド文法エラー」が起こったためにSRQが発生したことがわかります。 *ESR?で問い合わせると、標準イベントレジスタの内容がクリアされます。 - 標準イベントレジスタについては、6-3ページを参照してください。

***IDN? (IDeNtify)**

機能	機種を問い合わせます。
構文	*IDN?
例	*IDN? -> YOKOGAWA,760151-0401,0,F1.01
解説	<製造者>,<機種>,<シリアルNo.>,<ファームウェアのバージョン>の形式で返されます。<シリアルNo.>は実際には返していません(常に0)。

***OPC (OPeration Complete)**

機能	指定したオーバーラップコマンドが終了したときに、標準イベントレジスタのビット0(OPCビット)を1にセットします。
構文	*OPC
例	*OPC
解説	*OPCを使った同期のとり方については、4-7ページを参照してください。 - オーバーラップコマンドの指定は、「COMMunicate:OPSE」で行います。 - メッセージの最後でない*OPCの動作は保証されません。

***OPC? (OPeration Complete)**

機能	指定したオーバーラップコマンドが終了していれば、ASCIIコードの「1」を返します。
構文	*OPC?
例	*OPC? -> 1
解説	*OPC?を使った同期のとり方については、4-7ページを参照してください。 - オーバーラップコマンドの指定は「COMMunicate:OPSE」で行います。 - メッセージの最後でない*OPC?の動作は保証されません。

***OPT? (OPTion)**

機能	装備しているオプションを問い合わせます。
構文	*OPT?
例	*OPT? -> B5,C10
解説	- 内蔵プリンタ(/B5)、SCSIインタフェース(/C7)またはイーサネット+SCSI+内蔵HDD(/C10)の有無が返されます。 - いずれのオプションも装備していない場合は、ASCIIコードの「0」を返します。 「*OPT?」はプログラムメッセージの最後のクエリ(問い合わせ)でなければなりません。後ろにほかのクエリがあるときは、エラーになります。

***PSC (Power-on Status Clear)**

機能	電源ON時に以下のレジスタをクリアするかしないかを設定/問い合わせします。整数に丸めた値が「0」以外のときにクリアされます。
	- 標準イベントイネーブルレジスタ - 拡張イベントイネーブルレジスタ - 遷移フィルタ
構文	*PSC {<NRf>}
	*PSC?
	<NRf>=0(クリアしない)、0以外(クリアする)
例	*PSC 1
	*PSC? -> 1
解説	各レジスタについては6章を参照してください。

***RST (ReSeT)**

機能	設定の初期化を行います。
構文	*RST
例	*RST
解説	- 以前に送った*OPCおよび*OPC?も取り消します。 - 通信の設定を除くすべての設定を工場出荷時の設定値にします。

5.20 共通コマンドグループ

*SRE (Service Request Enable register)

機能	サービスリクエストイネーブルレジスタの値を設定/問い合わせします。
構文	*SRE <Nrf> *SRE?
例	<Nrf> = 0 ~ 255 *SRE 239 *SRE? -> 175(ビット6(MSS)の設定が無視されるため)
解説	・各ビットの10進数の和で設定します。 ・たとえば、「*SRE 239」とすると、サービスリクエストイネーブルレジスタを「11101111」にセットします。つまり、サービスリクエストイネーブルレジスタのビット4を無効にし、「出力キューが空でない」ときでもステータスバイトレジスタのビット4(MAV)を「1」にしません。 ・ただし、ステータスバイトレジスタのビット6(MSS)はMSSビット自身なので、無視されます。 ・初期値は「*SRE 0」(全ビット無効)です。 ・*SRE?で問い合わせても、サービスリクエストイネーブルレジスタの内容はクリアされません。 ・サービスリクエストイネーブルレジスタについては、6-2ページを参照してください。

*STB? (Status Byte)

機能	ステータスバイトレジスタの値を問い合わせます。
構文	*STB?
例	*STB? -> 4
解説	・各ビットの和が10進数で返されます。 ・シリアルポールを実行せずにレジスタを読むので、ビット6はRQSではなくMSSビットになります。 ・たとえば、「4」が返されると、ステータスバイトレジスタが「00000100」にセットされていることを示します。つまり、「エラーキューが空でない」(エラーが発生した)ことがわかります。 ・*STB?で問い合わせても、ステータスバイトレジスタの内容はクリアされません。 ・ステータスバイトレジスタについては、6-2ページを参照してください。

*TRG (TRiGger)

機能	SINGLE(SHIFT+HOLD)を押したのと同じ動作をします。
構文	*TRG
例	*TRG
解説	マルチラインメッセージGET(Group Execute Trigger)も、このコマンドと同じ動作をします。

*TST? (TeSt)

機能	セルフテストを実行し、結果を問い合わせます。
構文	*TST?
例	*TST? -> 0
解説	・セルフテストの内容は、内部の各メモリのテストです。 ・セルフテスト結果が正常なときは「0」、異常があるときは、「1」が返されます。

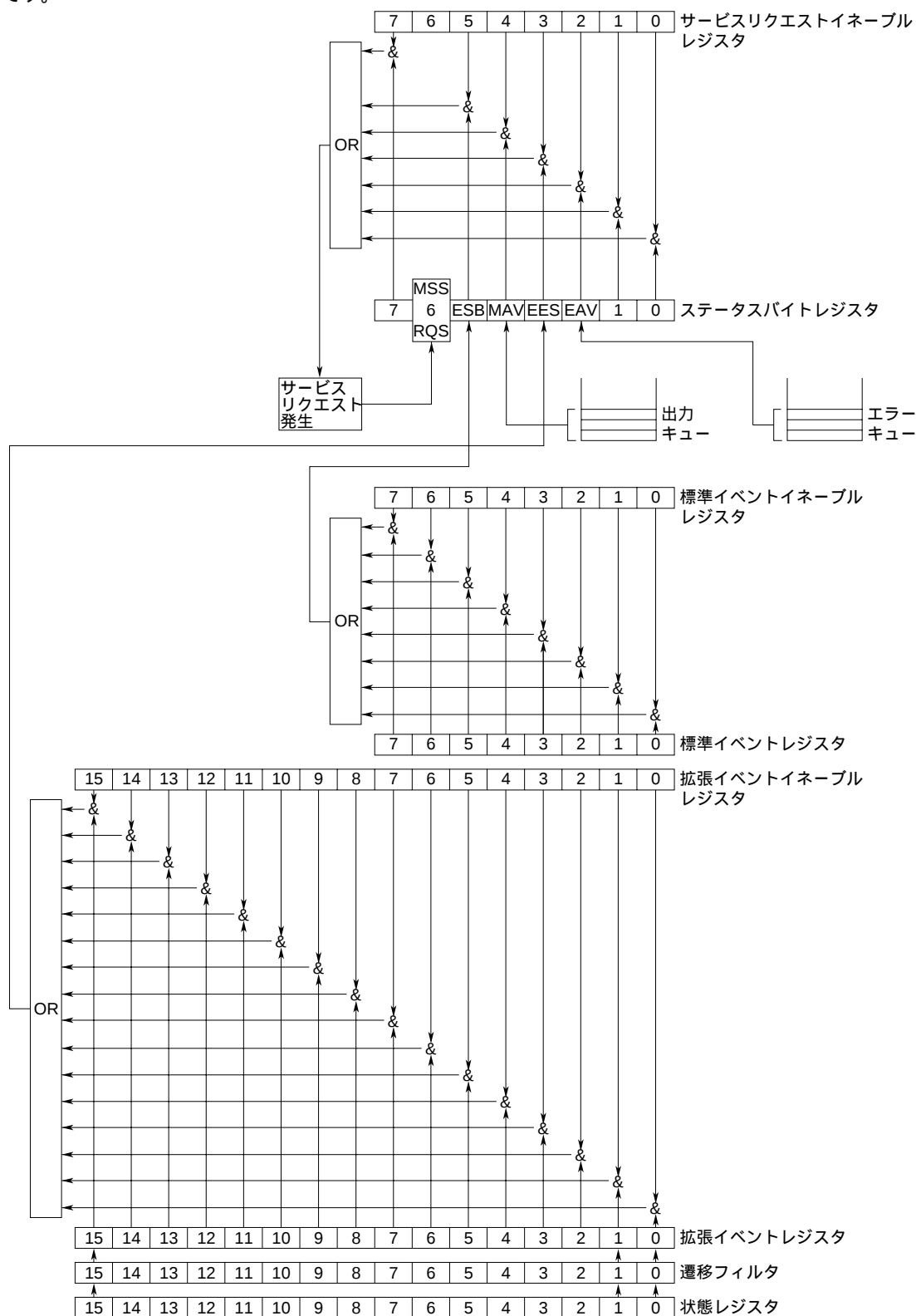
*WAI (WAIt)

機能	指定したオーバーラップコマンドが終了するまで、*WAIに続く命令を待ちます。
構文	*WAI
例	*WAI
解説	・*WAIを使った同期のとり方については、4-7ページを参照してください。 ・オーバーラップコマンドの指定は、「COMMunicate:OPSE」で行います。

6.1 ステータスレポートについて

ステータスレポート

シリアルポータルで読まれるステータスレポートは下図のとおりです。これは、IEEE 488.2-1987で規定されたものを拡張したものです。



各レジスタ・キューの概要

名称	機能	書き込み	読み出し
ステータスバイト		-	シリアルボール (RQS), *STB?(MSS)
サービスリクエストイネーブルレジスタ	ステータスバイトのマスク	*SRE	*SRE?
標準イベントレジスタ	機器の状態の変化	-	*ESR?
標準イベントイネーブルレジスタ	標準イベントレジスタのマスク	*ESE	*ESE?
拡張イベントレジスタ	機器の状態の変化	-	STATUS:EESR?
拡張イベントイネーブルレジスタ	拡張イベントレジスタのマスク	STATUS:ESEE	STATUS:ESEE?
状態レジスタ	現在の機器の状態	-	STATUS:CONdITION?
遷移フィルタ	拡張イベントレジスタの変化の条件	STATUS:FILTER<x>	STATUS:FILTer<x>?
出力キュー	問い合わせに対する応答メッセージを格納	各問い合わせコマンド	
エラーキュー	エラーNo.とメッセージを格納	-	STATUS:ERRor?

ステータスバイトに影響を与えるレジスタとキュー

ステータスバイトの各ビットに影響を与えるレジスタを整理すると、次のようになります。

- 標準イベントレジスタ：ステータスバイトのビット5 (ESB)を1/0にセット
- 出力キュー：ステータスバイトのビット4(MAV)を1/0にセット
- 拡張イベントレジスタ：ステータスバイトのビット3 (EES)を1/0にセット
- エラーキュー：ステータスバイトのビット2(EAV)を1/0にセット

各イネーブルレジスタ

各ビットをマスクして、そのビットが1であってもステータスバイトの要因にしないようにできるレジスタを整理すると、次のようになります。

- ステータスバイト：サービスリクエストイネーブルレジスタにより、各ビットをマスク
- 標準イベントレジスタ：標準イベントイネーブルレジスタにより、各ビットをマスク
- 拡張イベントレジスタ：拡張イベントイネーブルレジスタにより、各ビットをマスク

各レジスタの書き込み/読み出し

たとえば、標準イベントイネーブルレジスタの各ビットを1または0にするには、*ESEコマンドを使います。また、標準イベントイネーブルレジスタの各ビットが1であるか0であるかを確認するには、*ESE?コマンドを使います。これらの各コマンドについては、第5章で詳しく説明しています。

6.2 ステータスバイト

ステータスバイト

	RQS						
7	6	ESB	MAV	EES	EAV	1	0
	MSS						

- **ビット0, 1, 7**
未使用(常に0)
- **ビット2 EAV(Error Available)**
エラーキューが空でないときに1にセットされます。つまり、エラーが発生すると1になります。6-5ページを参照してください。
- **ビット3 EES(Extend Event Summary Bit)**
拡張イベントレジスタと、そのイネーブルレジスタの論理積が0でないときに、1にセットされます。つまり、機器の内部であるイベントが起こったときに1になります。6-4ページを参照してください。
- **ビット4 MAV(Message Available)**
出力キューが空でないときに1にセットされます。つまり、問い合わせを行って出力すべきデータがあるときに1になります。6-5ページを参照してください。
- **ビット5 ESB(Event Summary Bit)**
標準イベントレジスタと、そのイネーブルレジスタの論理積が0でないときに、1にセットされます。つまり、機器の内部であるイベントが起こったときに1になります。6-3ページを参照してください。
- **ビット6 RQS(Request Service)/MSS(Master Status Summary)**
ビット6以外のステータスバイトと、サービスリクエストイネーブルレジスタの論理積が0でないときに、1にセットされます。つまり、機器がコントローラにサービス要求をしているときに1になります。
RQSは、MSSが0から1になったときに1にセットされ、シリアルボールか、MSSが0になったときにクリアされます。

各ビットのマスク

ステータスバイトのあるビットをマスクしてSRQの要因にしたくないときには、サービスリクエストイネーブルレジスタの対応するビットを0にします。

たとえば、ビット2(EAV)をマスクして、エラーが発生してもサービスを要求しないようにするには、サービスリクエストイネーブルレジスタのビット2を0にします。これは*SREコマンドで行います。また、サービスリクエストイネーブルレジスタの各ビットが1であるか0であるかは、*SRE?で問い合わせられます。*SREコマンドについては、5章をお読みください。

ステータスバイトの動作

ステータスバイトのビット6が1になると、サービスリクエストを発生します。ビット6以外のどれかのビットが1になると、ビット6が1になります(サービスリクエストイネーブルレジスタの対応するビットも1のとき)。たとえば、何かのイベントが起こって、標準イベントレジスタとそのイネーブルレジスタの各ビットの論理積のうち、いずれかが1になったときは、ビット5(ESB)が1にセットされます。このとき、サービスリクエストイネーブルレジスタのビット5が1であれば、ビット6(MSS)が1にセットされ、コントローラにサービスを要求します。

また、ステータスバイトの内容を読むことにより、どんな種類のイベントが起こったのかを確認することができます。

ステータスバイトの読み出し

ステータスバイトの内容を読み出すには、次の2つの方法があります。

- ***STB?による問い合わせ**
*STB?で問い合わせると、ビット6はMSSになります。したがって、MSSを読み出すことになります。読み出したあとは、ステータスバイトのどのビットもクリアしません。
- **シリアルボール**
シリアルボールを実行すると、ビット6はRQSになります。したがって、RQSを読み出すことになります。読み出したあと、RQSだけをクリアします。シリアルボールではMSSを読み出すことはできません。

ステータスバイトのクリア

ステータスバイトの全ビットを強制的にクリアする方法はありません。各動作に対してクリアされるビットを以下に示します。

- ***STB?で問い合わせたとき**
どのビットもクリアされません。
- **シリアルボールを実行したとき**
RQSビットだけがクリアされます。
- ***CLSコマンドを受信したとき**
*CLSコマンドを受信すると、ステータスバイト自体はクリアされませんが、各ビットに影響する標準イベントレジスタなどの内容がクリアされます。その結果、それに対応したステータスバイトのビットがクリアされます。ただし、出力キューは*CLSコマンドではクリアできないので、ステータスバイトのビット4(MAV)は影響を受けません。ただし、*CLSコマンドをプログラムメッセージターミネータのすぐあとに受信したときは、出力キューもクリアされます。

6.3 標準イベントレジスタ

標準イベントレジスタ

7	6	5	4	3	2	1	0
PON	URQ	CME	EXE	DDE	QYE	RQC	OPC

- **ビット7 PON(Power ON) 電源ON**
本機器の電源がONになったときに、1になります。
- **ビット6 URQ(User Request) ユーザリクエスト**
未使用(常に0)
- **ビット5 CME(Command Error) コマンド文法エラー**
コマンドの文法に誤りがあるときに、1になります。
例 コマンド名のつづりの誤り、選択肢にない文字データを受信した。
- **ビット4 EXE(Execution Error) コマンド実行エラー**
コマンドの文法は正しいが、現在の状態では実行不可能なときに、1になります。
例 パラメータが設定範囲外、未装着のオプションに関するコマンドを受信した。
- **ビット3 DDE(Device Error) 機器特有のエラー**
コマンド文法エラー、コマンド実行エラー以外の機器の内部的原因で、コマンドが実行できなかったときに、1になります。
- **ビット2 QYE(Query Error) 問い合わせエラー**
問い合わせコマンドを送信したが、出力キューが空かデータが失われていたときに、1になります。
例 応答データがない、出力キューがあふれてデータが失われた
- **ビット1 RQC(Request Control) リクエストコントロール**
未使用(常に0)
- **ビット0 OPC(Operation Complete) 操作終了**
*OPCコマンド(5章参照)によって指定された動作が終了したときに、1になります。

各ビットのマスク

標準イベントレジスタのあるビットをマスクして、ステータスバイトのビット5(ESB)の要因にしたいときには、標準イベントイネーブルレジスタの対応するビットを0にします。

たとえば、ビット2(QYE)をマスクして問い合わせエラーが発生してもESBを1にしないようにするには、標準イベントイネーブルレジスタのビット2を0にします。これは*ESEコマンドで行います。また、標準イベントイネーブルレジスタの各ビットが1であるか0であるかは、*ESE?で問い合わせられます。*ESEコマンドについては、5章をお読みください。

標準イベントレジスタの動作

標準イベントレジスタは、機器の内部に起こった8種類のイベントに対するレジスタです。どれかのビットが1になると、ステータスバイトのビット5(ESB)を1にセットします(標準イベントイネーブルレジスタの対応するビットも1のとき)。

例

1. 問い合わせエラー発生
2. ビット2(QYE)が1にセットされる
3. 標準イベントイネーブルレジスタのビット2が1ならば、ステータスバイトのビット5(ESB)が1にセットされる

また、標準イベントレジスタの内容を読むことにより、機器の内部に起こったイベントの種類を確認することができます。

標準イベントレジスタの読み出し

標準イベントレジスタの内容は、*ESR?で読み出すことができます。読み出されたあとは、レジスタはクリアされます。

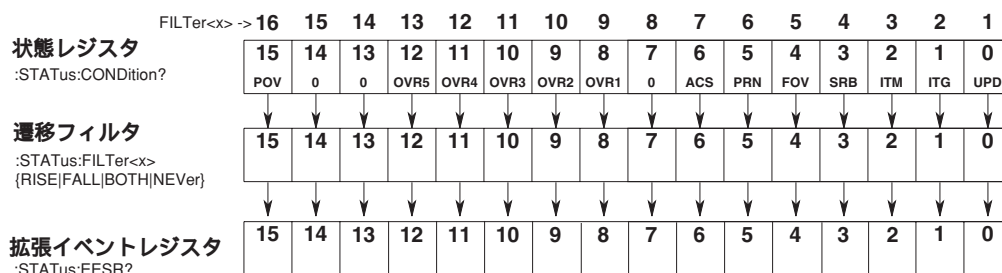
標準イベントレジスタのクリア

標準イベントレジスタがクリアされるのは、次の3つの場合です。

- ・ *ESR?で標準イベントレジスタの内容が読み出されたとき
- ・ *CLSコマンドを受信したとき
- ・ 電源再投入時

6.4 拡張イベントレジスタ

拡張イベントレジスタは、機器の内部状態を表す状態レジスタの状態変化が、遷移フィルタでエッジ検出された結果が入ります。



状態レジスタの各ビットの意味は、次の通りです。

ビット0	UPD (Updating)	測定データ更新中のときに「1」になります。 UPDの立ち下がり(1 → 0)が更新終了を意味します。
ビット1	ITG (Integrate Busy)	積算中のときに「1」になります。
ビット2	ITM (Integrate Timer Busy)	積算タイマ動作中のときに「1」になります。
ビット3	SRB (Store/Recall Busy)	ストアまたはリコール中のときに「1」になります。
ビット4	FOV (Frequency Over)	周波数がエラーのときに「1」になります。
ビット5	PRN (Printing)	内蔵プリンタ動作中、ネットワークプリンタにデータを出力中に1になります。
ビット6	ACS (Accessing)	フロッピーディスク、内蔵ハードディスク、および外部(SCSIデバイス/ネットワーク)の各ドライブのアクセス中に1になります。
ビット8	OVR1 (Element1 Measured Data Over)	エレメント1の電圧値/電流値のいずれかがオーバレンジのときに「1」になります。
ビット9	OVR2 (Element2 Measured Data Over)	エレメント2の電圧値/電流値のいずれかがオーバレンジのときに「1」になります。
ビット10	OVR3 (Element3 Measured Data Over)	エレメント3の電圧値/電流値のいずれかがオーバレンジのときに「1」になります。
ビット11	OVR4 (Element4 Measured Data Over)	エレメント4の電圧値/電流値のいずれかがオーバレンジのときに「1」になります。
ビット12	OVR5 (Element5 Measured Data Over)	エレメント5の電圧値/電流値のいずれかがオーバレンジのときに「1」になります。
ビット15	POV (ElementX Input Peak Over)	いずれかのエレメントでピークオーバを検出したときに「1」になります。

遷移フィルタのパラメータは、状態レジスタの指定されたビット(数値サフィックス1~16)の変化を次のように抽出し、拡張イベントレジスタを書き換えます。

RISE	0→1の変化で、拡張イベントレジスタの指定ビットを、1にします。
FALL	1→0の変化で、拡張イベントレジスタの指定ビットを、1にします。
BOTH	0→1または1→0の変化で、拡張イベントレジスタの指定ビットを、1にします。
NEVer	常に0。

6.5 出力キューとエラーキュー

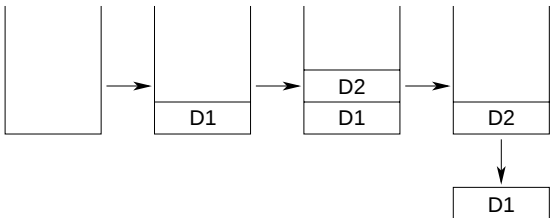
出力キュー

出力キューは、問い合わせ(クエリ)に対する応答メッセージを格納します。たとえば、取り込んだ波形データの出力を要求するWAVEform:SEND?を送信すると、そのデータはそれが読み出されるまで出力キューに蓄えられます。

下図のように、データは順番に蓄えられ、古いものから読み出されます。読み出す以外にも、次のときに出力キューは空になります。

- ・新しいメッセージをコントローラから受信したとき
- ・デッドロック状態になったとき(4-2ページ参照)
- ・デバイスクリア(DCLまたはSDC)を受信したとき
- ・電源の再投入

なお、*CLSコマンドでは出力キューを空にすることはできません。出力キューが空であるかどうかは、ステータスバイトのビット4(MAV)で確認できます。



エラーキュー

エラーキューは、エラーが発生したときにその番号とメッセージを格納します。たとえば、コントローラが間違ったプログラムメッセージを送信したら、エラーが表示されたときに「113, "Undefined header"」という番号とエラーメッセージがエラーキューに格納されます。エラーキューの内容は、STATus:ERRor?クエリで読み出すことができます。エラーキューは出力キューと同様に古いものから読み出されます。

エラーキューがあふれたときは、最後のメッセージを「350, "Queue overflow"」というメッセージに置き換えます。

読み出す以外にも次のときにエラーキューは空になります。

- ・*CLSコマンドを受信したとき
- ・電源の再投入

なお、エラーキューが空であるかどうかは、ステータスバイトのビット2(EAV)で確認できます。

7.1 プログラムを組む前に

対象PCと言語

対象PC : Windows PC
対象言語 : Visual Basic Ver5.0 Professional Edition以上
GPIBボード : NI社製 AT-GPIB/TNT IEEE-488.2

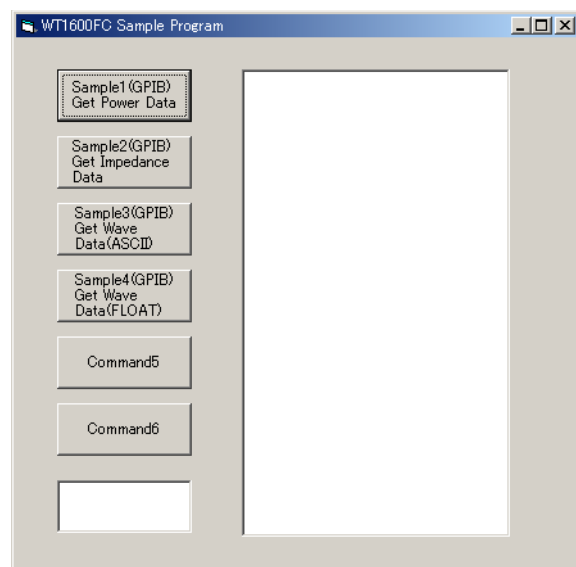
Visual Basic上での設定

使用標準モジュール : Niglobal.bas
Vbib-32.bas

本機器の設定

GPIB
この章のサンプルプログラムは、すべて本機器のGPIBアドレスを「1」として作成しています。
1-5ページの操作説明に従って、GPIBアドレスを「1」に設定してください。

7.2 サンプルプログラムイメージ



7.3 初期化・エラー・実行関数

```
-----
Option Explicit
Dim StartFlag As Integer           'Start Flag
Dim addr As Integer               'GPiB Address
Dim Timeout As Integer            'Timeout
Dim Dev As Integer                'Device ID(GPiB)
Dim term As String                'Terminator
Dim Query(1100) As String         'Query String
Dim Dummy As Integer
-----

Private Function InitGpib() As Integer
    Dim eos As Integer             'EOS
    Dim eot As Integer             'EOI
    Dim brd As Integer             'GPiB Board ID
    Dim sts As Integer

    eos = &HC0A                    'Terminator = LF
    eot = 1                        'EOI = Enable
    term = Chr(10)                  'Timeout = 10s
    Timeout = T10s

    brd = ilfind("GPiB0")
    If (brd < 0) Then
        Call DisplayGPiBError(brd, "ilfind")
        InitGpib = 1
        Exit Function
    End If
    Dev = ildev(0, addr, 0, Timeout, eot, eos)
    If (Dev < 0) Then
        Call DisplayGPiBError(Dev, "ildev")
        InitGpib = 1
        Exit Function
    End If
    sts = ilsic(brd)                'Set IFC
    If (sts < 0) Then
        Call DisplayGPiBError(sts, "ilsic")
        InitGpib = 1
        Exit Function
    End If
    InitGpib = 0
End Function
-----

Private Sub DisplayGPiBError(ByVal sts As Integer, ByVal msg As String)
    Dim wrn As String
    Dim ers As String
    Dim ern As Integer

    If (sts And TIMO) Then
        wrn = "Time out" + Chr(13)
    Else
        wrn = ""
    End If
    If (sts And EERR) Then
        ern = iberr
        If (ern = EDVR) Then
            ers = "EDVR:System error"
        ElseIf (ern = ECIC) Then
            ers = "ECIC:Function requires GPiB board to be CIC"
        ElseIf (ern = ENOL) Then
            ers = "ENOL:No Listeners on the GPiB"
        ElseIf (ern = EADR) Then
            ers = "EADR:GPiB board not addressed correctly"
        ElseIf (ern = EARG) Then
            ers = "EARG:Invalid argument to function call"
        ElseIf (ern = ESAC) Then
            ers = "ESAC:GPiB board not System Controller as required"
        ElseIf (ern = EABO) Then
            ers = "EABO:I/O operation aborted(timeout)"
        ElseIf (ern = ENEB) Then
            ers = "ENEB:Nonexistent GPiB board"
        ElseIf (ern = EDMA) Then
            ers = "EDMA:DMA error"
        ElseIf (ern = EOIP) Then
            ers = "EOIP:I/O operation started before previous operation completed"
        ElseIf (ern = ECAP) Then
            ers = "ECAP:No capability for intended operation"
        ElseIf (ern = EFSO) Then
            ers = "EFSO:File system operation error"
        ElseIf (ern = EBUS) Then
            ers = "EBUS:GPiB bus error"
        ElseIf (ern = ESTB) Then
            ers = "ESTB:Serial poll status byte queue overflow"
        ElseIf (ern = ESRQ) Then
            ers = "ESRQ:SRQ remains asserted"
        End If
    End If
End Sub
```

```

        ElseIf (ern = ETAB) Then
            ers = "ETAB:The return buffer is full"
        ElseIf (ern = ELCK) Then
            ers = "ELCK:Address or board is locked"
        Else
            ers = ""
        End If
    Else
        ers = ""
    End If

    MsgBox ("Status No. " + Str(sts) + Chr(13) + wrn + "Error No. " + Str(ern) + Chr(13)
+ ers + Chr(13) + msg), vbExclamation, "Error!"
    Call ibonl(Dev, 0)
    Dev = -1
End Sub
-----

Private Sub Command1_Click()
    Dim sts As Integer

    If (StartFlag = 1) Then
        Exit Sub
    End If
    StartFlag = 1
    Text1.Text = "START"
    List1.Clear
    Dummy = DoEvents()
    sts = GpibPower                                'Run Sample1(GPIB) Get Numerical Data
(Power)
    If (sts = 0) Then
        Text1.Text = "END"
    Else
        Text1.Text = "ERROR"
    End If
    StartFlag = 0
End Sub
-----

Private Sub Command2_Click()
    Dim sts As Integer

    If (StartFlag = 1) Then
        Exit Sub
    End If
    StartFlag = 1
    Text1.Text = "START"
    List1.Clear
    Dummy = DoEvents()
    sts = GpibImpedance                            'Run Sample2(GPIB) Get Numerical Data
(Impedance)
    If (sts = 0) Then
        Text1.Text = "END"
    Else
        Text1.Text = "ERROR"
    End If
    StartFlag = 0
End Sub
-----

Private Sub Command3_Click()
    Dim sts As Integer

    If (StartFlag = 1) Then
        Exit Sub
    End If
    StartFlag = 1
    Text1.Text = "START"
    List1.Clear
    Dummy = DoEvents()
    sts = GpibWaveAscii                            'Run Sample3(GPIB) Get Waveform data
(ASCII)
    If (sts = 0) Then
        Text1.Text = "END"
    Else
        Text1.Text = "ERROR"
    End If
    StartFlag = 0
End Sub
-----

Private Sub Command4_Click()
    Dim sts As Integer

    If (StartFlag = 1) Then
        Exit Sub
    End If
    StartFlag = 1
    Text1.Text = "START"
    List1.Clear

```

```

        Dummy = DoEvents()
        sts = GpibWaveFloat
        If (sts = 0) Then
            Text1.Text = "END"
        Else
            Text1.Text = "ERROR"
        End If
        StartFlag = 0
    End Sub
-----

Private Sub Command5_Click()
    Dim sts As Integer

    If (StartFlag = 1) Then
        Exit Sub
    End If
    StartFlag = 1
    Text1.Text = "START"
    List1.Clear
    List1.AddItem "NOT MAKE"
    Text1.Text = "END"
    StartFlag = 0
End Sub
-----

Private Sub Command6_Click()
    Dim sts As Integer

    If (StartFlag = 1) Then
        Exit Sub
    End If
    StartFlag = 1
    Text1.Text = "START"
    List1.Clear
    List1.AddItem "NOT MAKE"
    Text1.Text = "END"
    StartFlag = 0
End Sub
-----

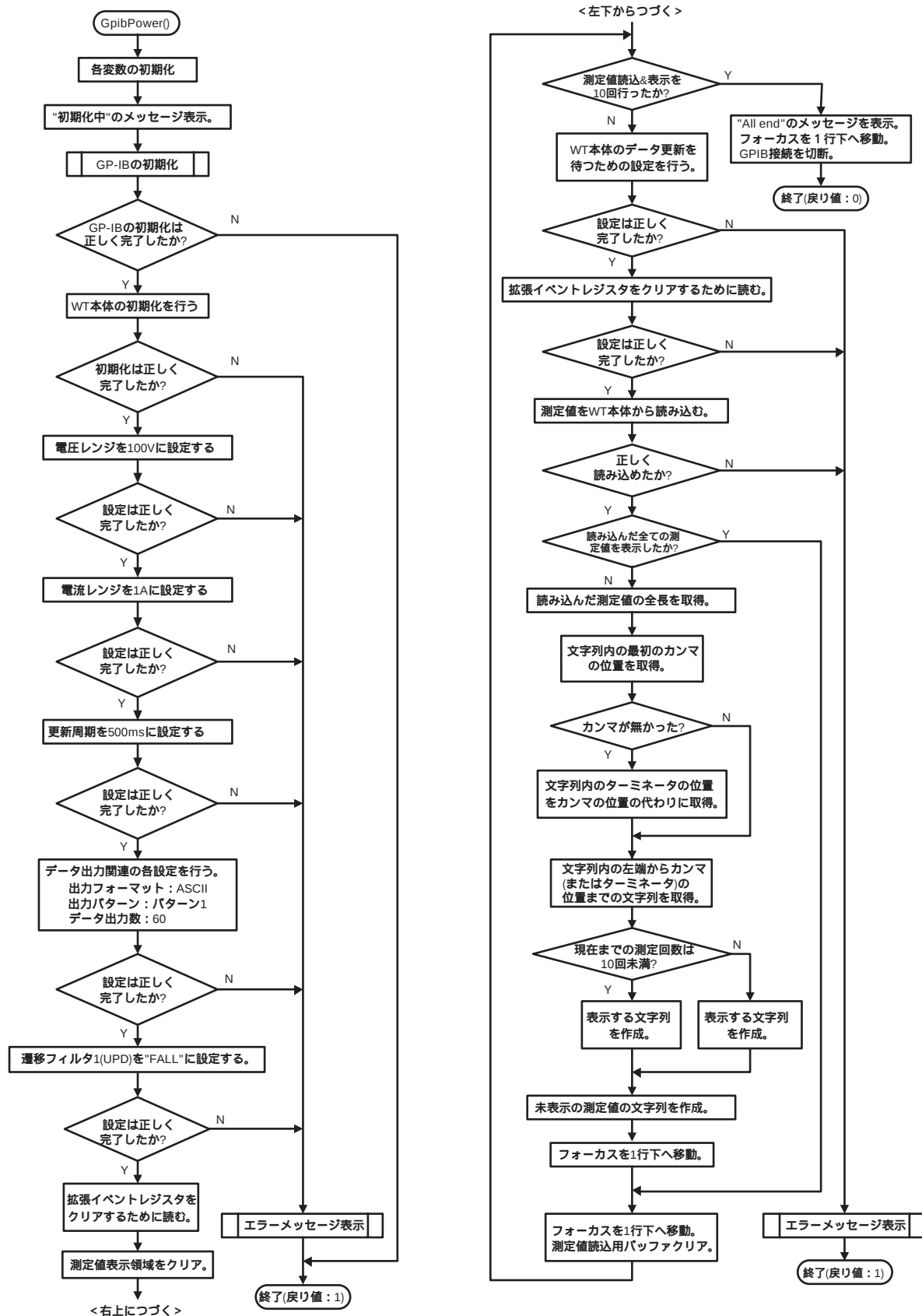
Private Sub Form_Load()

    StartFlag = 0
    Dev = -1
    addr = 1
    Command1.Caption = "Sample1(GPIB)" + Chr(13) + "Get Power Data"
    Command2.Caption = "Sample2(GPIB)" + Chr(13) + "Get Impedance Data"
    Command3.Caption = "Sample3(GPIB)" + Chr(13) + "Get Wave Data(ASCII)"
    Command4.Caption = "Sample4(GPIB)" + Chr(13) + "Get Wave Data(FLOAT)"
    Text1.Text = ""

End Sub
-----

```

7.4 電力測定データの出力



```

Sample1(GPIB) Get Power Data
-----
Private Function GpibPower() As Integer
    Dim msg As String                'Command buffer
    Dim qry As String                'Query buffer
    Dim sts As Integer
    Dim item As Integer
    Dim comma As Integer
    Dim length As Integer
    Dim cnt As Integer

    term = Chr$(10)                  'terminator
    msg = Space$(100)
    qry = Space$(900)

    List1.AddItem "Now Initializing. Wait a moment."
    Dummy = DoEvents()

    sts = InitGpib                    'Initialize GPIB
    If (sts <> 0) Then
        GpibPower = 1
        Exit Function
    End If

    'Initialize the settings
    msg = "*RST" + term              'Initialize the settings
    sts = ilwrt(Dev, msg, Len(msg))  'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibPower = 1
        Exit Function
    End If

    'Set the measurment condition
    msg = "VOLTAGE:RANGE 100V" + term 'Voltage range = 100V
    sts = ilwrt(Dev, msg, Len(msg))  'Send Command

    Call DisplayGPIBError(sts, msg)
    GpibPower = 1
    Exit Function
End If

    msg = "CURRENT:RANGE 1A" + term  'Current range = 1A
    sts = ilwrt(Dev, msg, Len(msg))  'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibPower = 1
        Exit Function
    End If

    msg = "RATE 500MS" + term         'Update Rate = 500ms
    sts = ilwrt(Dev, msg, Len(msg))  'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibPower = 1
        Exit Function
    End If

    'Set the numerical data output items
    'ASCII format, Preset pattern1, Number of data = 60
    msg = "NUMERIC:FORMAT ASCII;NORMAL:PRESET 1;NUMBER 60" + term
    sts = ilwrt(Dev, msg, Len(msg))  'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibPower = 1
        Exit Function
    End If

    'Set the transition filter used to detect the completion of the data updating
    msg = "STATUS:FILTER1 FALL" + term 'Falling edge of bit0(UPD)
    sts = ilwrt(Dev, msg, Len(msg))  'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibPower = 1
        Exit Function
    End If

    'Clear the extended event register (Read and trash the response)
    msg = "STATUS:EESR?" + term
    sts = ilwrt(Dev, msg, Len(msg))  'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibPower = 1
        Exit Function
    End If

    sts = ilrd(Dev, qry, Len(qry))    'Receive Query
    If (sts < 0) Then

```

```

        Call DisplayGPIBError(sts, msg)
        GpibPower = 1
        Exit Function
    End If

    List1.Clear
    'Read and display the numerical data (It is repeated 10 times in this program)
    For cnt = 1 To 10
        'Wait for the completion of the data updating
        msg = "COMMUNICATE:WAIT 1" + term
        sts = ilwrt(Dev, msg, Len(msg))          'Send Command
        If (sts < 0) Then
            Call DisplayGPIBError(sts, msg)
            GpibPower = 1
            Exit Function
        End If

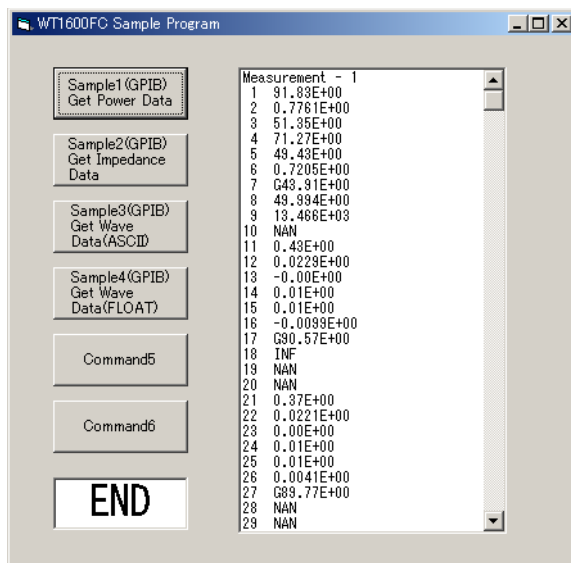
        'Clear the extended event register (Read and trash the response)
        msg = "STATUS:EESR?" + term
        sts = ilwrt(Dev, msg, Len(msg))          'Send Command
        If (sts < 0) Then
            Call DisplayGPIBError(sts, msg)
            GpibPower = 1
            Exit Function
        End If
        sts = ilrd(Dev, qry, Len(qry))           'Receive Query
        If (sts < 0) Then
            Call DisplayGPIBError(sts, msg)
            GpibPower = 1
            Exit Function
        End If

        'Read out numerical data
        msg = "NUMERIC:NORMAL:VALUE?" + term
        sts = ilwrt(Dev, msg, Len(msg))          'Send Command
        If (sts < 0) Then
            Call DisplayGPIBError(sts, msg)
            GpibPower = 1
            Exit Function
        End If
        sts = ilrd(Dev, qry, Len(qry))           'Receive Query
        If (sts < 0) Then
            Call DisplayGPIBError(sts, msg)
            GpibPower = 1
            Exit Function
        End If

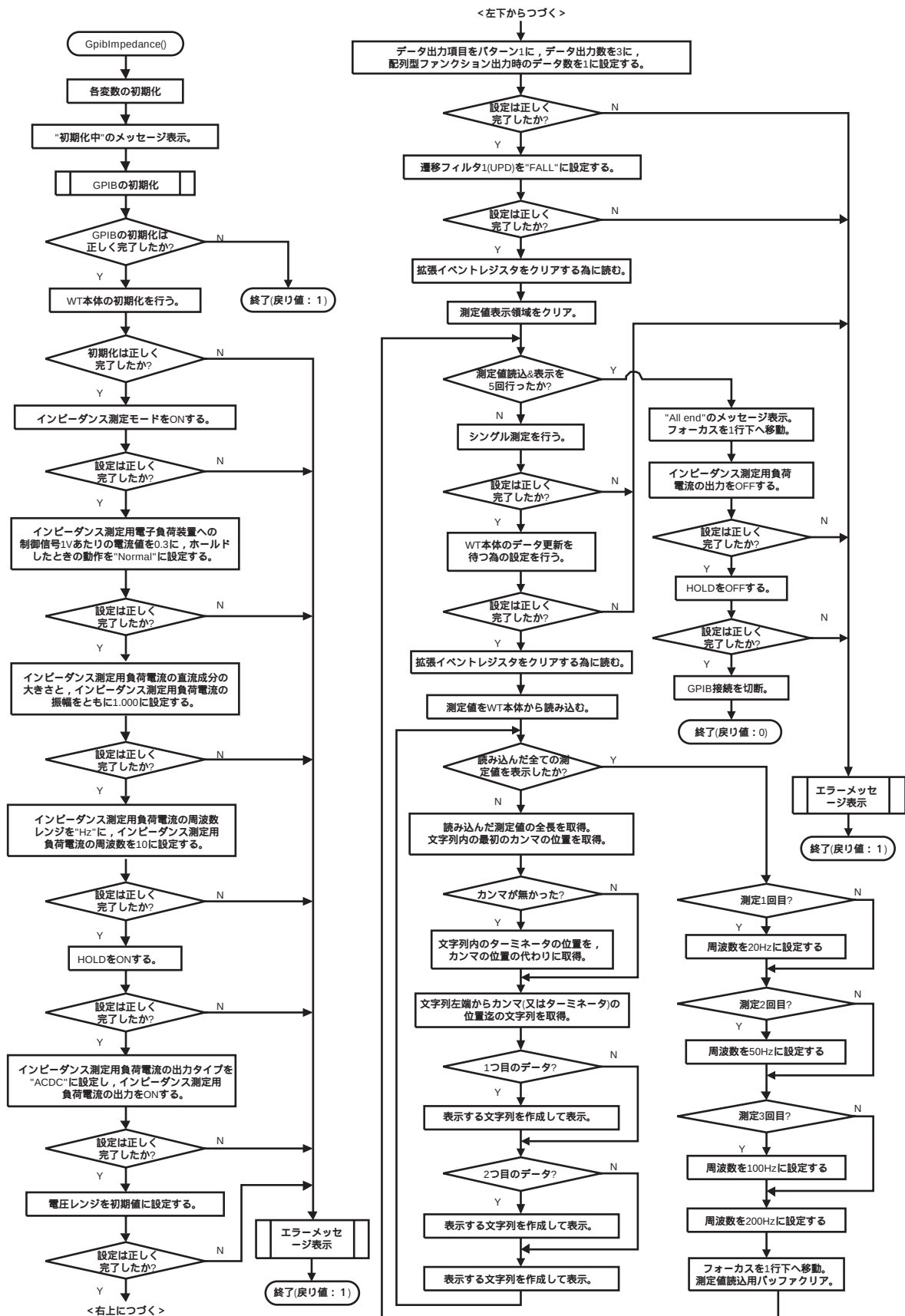
        'Extract items that are separated by commas(,) from the received data
        List1.AddItem "Measurement - " + CStr(cnt)
        List1.ListIndex = List1.ListIndex + 1
        For item = 1 To 60
            length = Len(qry)
            comma = InStr(qry, ",")
            If (comma = 0) Then comma = InStr(qry, term)
            Query(item) = Left(qry, comma - 1)
            If item < 10 Then
                List1.AddItem " " + CStr(item) + " " + Query(item)
            Else
                List1.AddItem CStr(item) + " " + Query(item)
            End If
            qry = Mid(qry, comma + 1)
            List1.ListIndex = List1.ListIndex + 1
        Next item
        List1.AddItem ""
        List1.ListIndex = List1.ListIndex + 1
        qry = Space$(900)
        Dummy = DoEvents()
    Next cnt

    List1.AddItem " All end"
    List1.ListIndex = List1.ListIndex + 1
    Call ibonl(Dev, 0)
    GpibPower = 0
End Function
-----

```



7.5 インピーダンス測定データの出力



```

Sample2(GPIB) Get Impedance Data
-----
Private Function GpibImpedance() As Integer
    Dim msg As String                'Command buffer
    Dim qry As String                'Query buffer
    Dim sts As Integer
    Dim wait As Integer
    Dim item As Integer
    Dim comma As Integer
    Dim length As Integer
    Dim cnt As Integer

    term = Chr$(10)                  'terminator
    msg = Space$(100)
    qry = Space$(1200)

    List1.AddItem "Now Initializing. Wait a moment."
    Dummy = DoEvents()

    sts = InitGpib                    'Initialize GPIB
    If (sts <> 0) Then
        GpibImpedance = 1
        Exit Function
    End If

    'Initialize the settings
    msg = "*RST" + term               'Initialize the settings
    sts = ilwrt(Dev, msg, Len(msg))   'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    'Set the measurment condition
    'Mode = impedance mode
    msg = "IMPEDANCE:STATE ON" + term
    sts = ilwrt(Dev, msg, Len(msg))   'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    'Superpose Output Ratio = 0.3, Hold Action = Normal
    msg = "IMPEDANCE:SUPERPOSE:DETAIL:RATIO 0.3;HOLD NORMAL" + term
    sts = ilwrt(Dev, msg, Len(msg))   'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    'Superpose DC Offset = 1.000, Superpose AC Amp = 1.000
    msg = "IMPEDANCE:SUPERPOSE:OFFSET 1;AMPLITUDE 1" + term
    sts = ilwrt(Dev, msg, Len(msg))   'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    'Frequency Range = Hz, Frequency Value = 10
    msg = "IMPEDANCE:SUPERPOSE:FREQUENCY:RANGE HZ;VALUE 10" + term
    sts = ilwrt(Dev, msg, Len(msg))   'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    'Hold On
    msg = "HOLD ON" + term
    sts = ilwrt(Dev, msg, Len(msg))   'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    'Superpose Output Type = AC/DC, Superpose Output = ON
    msg = "IMPEDANCE:SUPERPOSE:OUTPUT:TYPE ACDC;STATE ON" + term
    sts = ilwrt(Dev, msg, Len(msg))   'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    'Initialize voltage range
    msg = "IMPEDANCE:VOLTAGE:INITIALIZE" + term
    sts = ilwrt(Dev, msg, Len(msg))   'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If
End If

```

7.5 インピーダンス測定データの出力

```
'Preset pattern1, Number of data = 3, Number of data of array function = 1
msg = "NUMERIC:IMPEDANCE:PRESET 1;NUMBER 3;ARRAY 1" + term
sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
If (sts < 0) Then
    Call DisplayGPIBError(sts, msg)
    GpibImpedance = 1
    Exit Function
End If

'Set the transition filter used to detect the completion of the data updating
msg = "STATUS:FILTER1 FALL" + term 'Falling edge of bit0(UPD)
sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
If (sts < 0) Then
    Call DisplayGPIBError(sts, msg)
    GpibImpedance = 1
    Exit Function
End If

'Clear the extended event register (Read and trash the response)
msg = "STATUS:EESR?" + term
sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
If (sts < 0) Then
    Call DisplayGPIBError(sts, msg)
    GpibImpedance = 1
    Exit Function
End If
sts = ilrd(Dev, qry, Len(qry)) 'Receive Query
If (sts < 0) Then
    Call DisplayGPIBError(sts, msg)
    GpibImpedance = 1
    Exit Function
End If

List1.Clear
'Read and display the numerical data (It is repeated 5 times in this program)
For cnt = 1 To 5
    msg = "*TRG" + term 'Single trigger
    sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    'Wait for the completion of the data updating
    msg = "COMMUNICATE:WAIT 1" + term
    sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    'Clear the extended event register (Read and trash the response)
    msg = "STATUS:EESR?" + term
    sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If
    sts = ilrd(Dev, qry, Len(qry)) 'Receive Query
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    'Read out numerical data
    msg = "NUMERIC:IMPEDANCE:VALUE?" + term
    sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If
    sts = ilrd(Dev, qry, Len(qry)) 'Receive Query
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    'Extract items that are separated by commas(,) from the received data
    List1.AddItem "Measurement - " + CStr(cnt)
    List1.ListIndex = List1.ListIndex + 1
    For item = 1 To 3
        length = Len(qry)
        comma = InStr(qry, ",")
        If (comma = 0) Then comma = InStr(qry, term)
        Query(item) = Left(qry, comma - 1)
        If (item = 1) Then
            List1.AddItem "Freq : " + Query(item)
```

```

        ElseIf (item = 2) Then
            List1.AddItem "Z'   " : " + Query(item)
        Else
            List1.AddItem "Z''  " : " + Query(item)
        End If

        qry = Mid(qry, comma + 1)
        List1.ListIndex = List1.ListIndex + 1
    Next item

    'Change Frequency
    If cnt = 1 Then
        msg = "IMPEDANCE:SUPERPOSE:FREQUENCY:VALUE 20" + term      'Frequency Value
= 20
    ElseIf cnt = 2 Then
        msg = "IMPEDANCE:SUPERPOSE:FREQUENCY:VALUE 50" + term      'Frequency Value
= 50
    ElseIf cnt = 3 Then
        msg = "IMPEDANCE:SUPERPOSE:FREQUENCY:VALUE 100" + term      'Frequency Value
= 100
    Else
        msg = "IMPEDANCE:SUPERPOSE:FREQUENCY:VALUE 200" + term      'Frequency Value
= 200
    End If
    sts = ilwrt(Dev, msg, Len(msg))                                'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibImpedance = 1
        Exit Function
    End If

    List1.AddItem ""
    List1.ListIndex = List1.ListIndex + 1
    qry = Space$(900)
    Dummy = DoEvents()
Next cnt

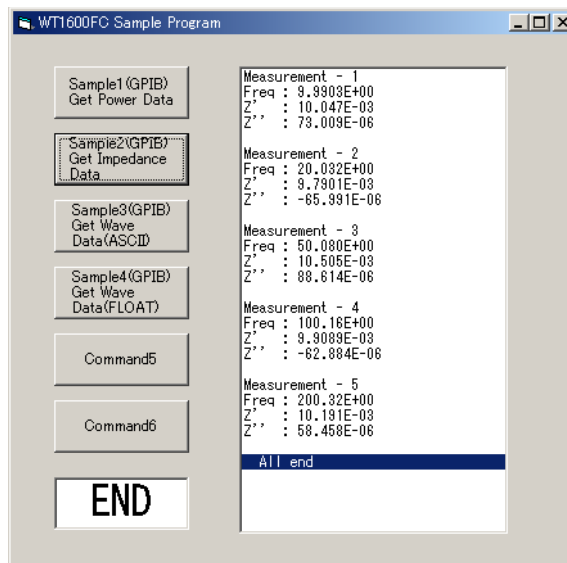
List1.AddItem " All end"
List1.ListIndex = List1.ListIndex + 1

msg = "IMPEDANCE:SUPERPOSE:OUTPUT:STATE OFF" + term              'Superspose
Output = OFF
sts = ilwrt(Dev, msg, Len(msg))                                'Send Command
If (sts < 0) Then
    Call DisplayGPIBError(sts, msg)
    GpibImpedance = 1
    Exit Function
End If

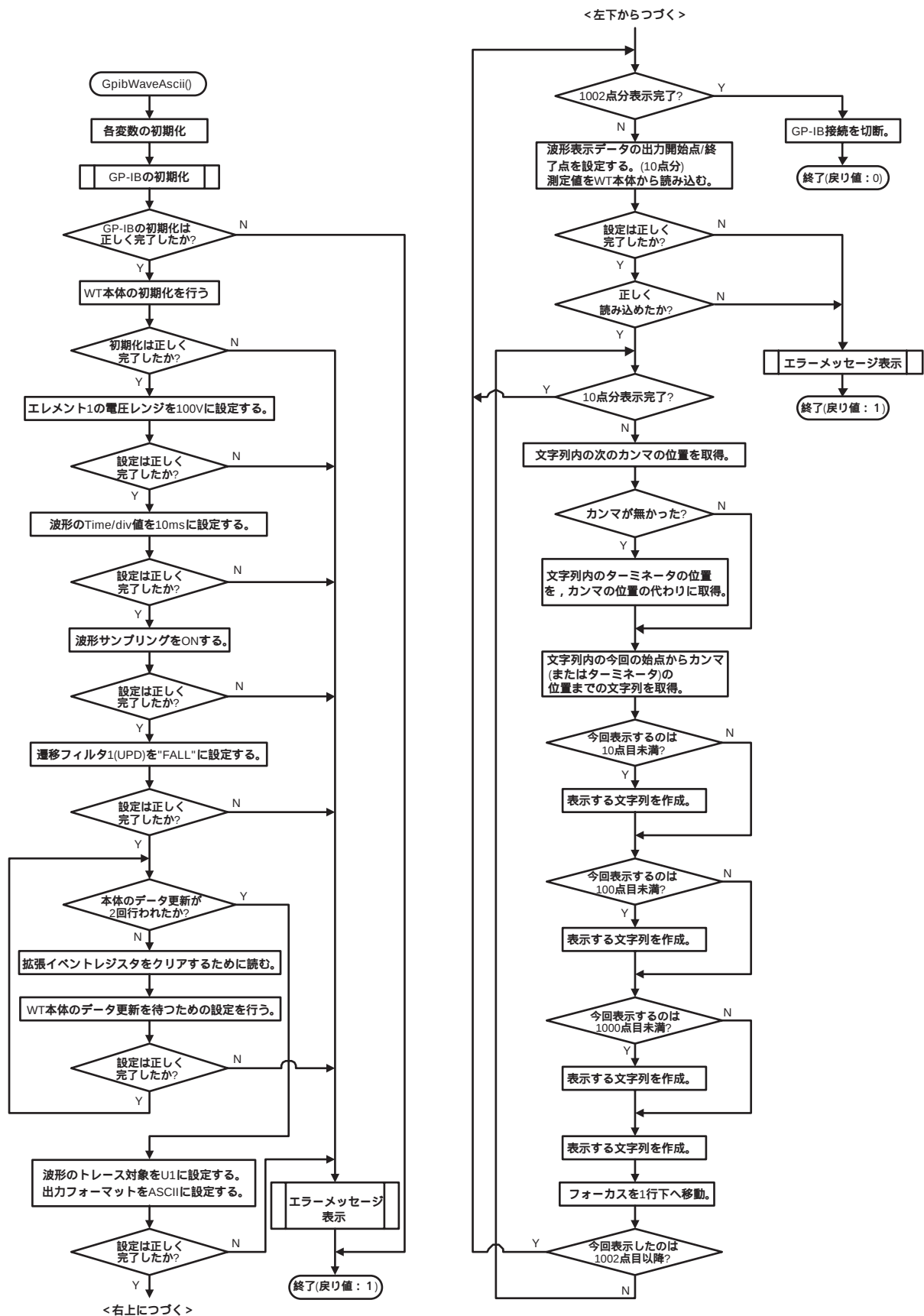
msg = "HOLD OFF" + term                                          'Hold Off
sts = ilwrt(Dev, msg, Len(msg))                                'Send Command
If (sts < 0) Then
    Call DisplayGPIBError(sts, msg)
    GpibImpedance = 1
    Exit Function
End If

Call ibon1(Dev, 0)
GpibImpedance = 0
End Function

```



7.6 波形データ(ASCII形式)の出力



```

Sample3(GPIB) Get Wave Data (ASCII)
-----
Private Function GpibWaveAscii() As Integer
    Dim msg As String           'Command buffer
    Dim qry As String           'Query buffer
    Dim sts As Integer
    Dim wait As Integer
    Dim pnt1 As Integer
    Dim num As Integer
    Dim i As Integer
    Dim j As Integer
    Dim k As Integer
    Dim comma As Integer

    term = Chr$(10)             'terminator
    msg = Space$(100)
    qry = Space$(200)

    sts = InitGpib              'Initialize GPIB
    If (sts <> 0) Then
        GpibWaveAscii = 1
        Exit Function
    End If

    'Initialize the settings
    msg = "*RST" + term         'Initialize the settings
    sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveAscii = 1
        Exit Function
    End If

    'Set the measurment condition
    msg = "VOLTAGE:RANGE:ELEMENT1 100V" + term 'Voltage range = 100V
    sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveAscii = 1
        Exit Function
    End If

    msg = "WSETUP:TDIV 10MS" + term 'Time/div = 10ms
    sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveAscii = 1
        Exit Function
    End If

    msg = "WSETUP:SAMPLING ON" + term 'Wave sampling start
    sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveAscii = 1
        Exit Function
    End If

    'Set the transition filter used to detect the completion of the data updating
    msg = "STATUS:FILTER1 FALL" + term 'Falling edge of bit0(UPD)
    sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveAscii = 1
        Exit Function
    End If

    'Wait until waveform measure is stable (2 samples in this program)
    For wait = 1 To 2
        'Clear the extended event register (Read and trash the response)
        msg = "STATUS:EESR?" + term
        sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
        If (sts < 0) Then
            Call DisplayGPIBError(sts, msg)
            GpibWaveAscii = 1
            Exit Function
        End If

        sts = ilrd(Dev, qry, Len(qry)) 'Receive Query
        If (sts < 0) Then
            Call DisplayGPIBError(sts, msg)
            GpibWaveAscii = 1
            Exit Function
        End If

        'Wait for the completion of the data updating
        msg = "COMMUNICATE:WAIT 1" + term
        sts = ilwrt(Dev, msg, Len(msg)) 'Send Command
        If (sts < 0) Then

```

7.6 波形データ(ASCII形式)の出力

```

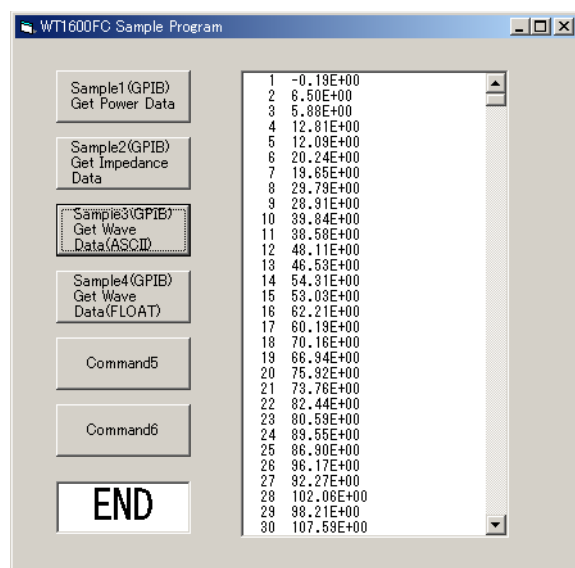
        Call DisplayGPIBError(sts, msg)
        GpibWaveAscii = 1
        Exit Function
    End If
Next wait

'Set conditions for reading the waveform
'ASCII format, Trace = U1
msg = "WAVEFORM:TRACE U1;FORMAT ASCII" + term
sts = ilwrt(Dev, msg, Len(msg))          'Send Command
If (sts < 0) Then
    Call DisplayGPIBError(sts, msg)
    GpibWaveAscii = 1
    Exit Function
End If

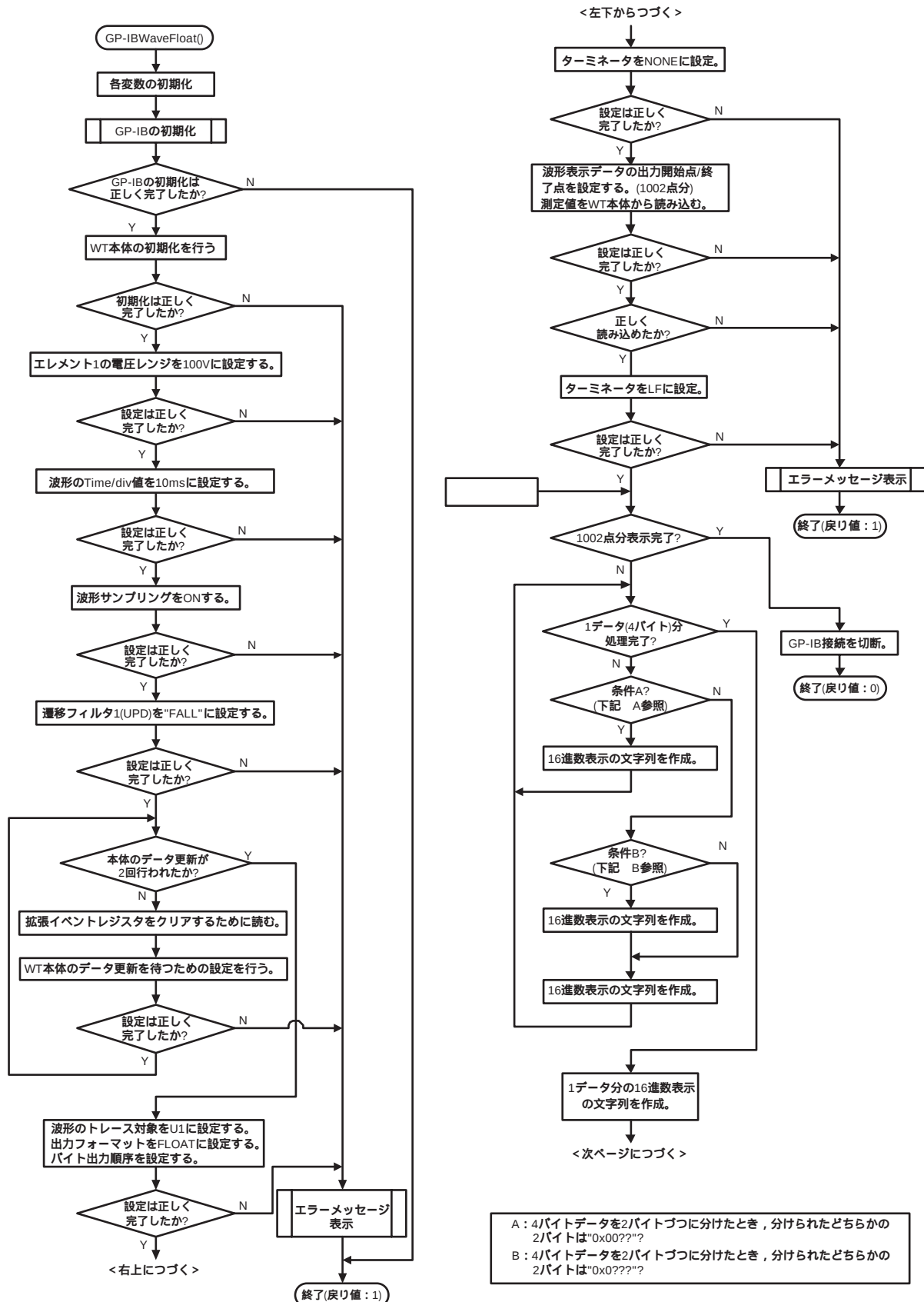
'Read and display the waveform data
pntl = 1002
num = 0
For i = 0 To pntl Step 10
    'Read in the waveform data 10 data points at a time
    msg = "WAVEFORM:START" + Str(i) + ";END" + Str(i + 9) + ";SEND?" + term
    sts = ilwrt(Dev, msg, Len(msg))      'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveAscii = 1
        Exit Function
    End If
    sts = ilrd(Dev, qry, Len(qry))        'Receive Query
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveAscii = 1
        Exit Function
    End If
    k = 1
    'Extract items that are separated by commas(,) from the received data
    For j = 0 To 9
        comma = InStr(k, qry, ",")
        If (comma = 0) Then comma = InStr(k, qry, term)
        num = num + 1
        Query(num) = Mid(qry, k, (comma - k))
        If (num < 10) Then
            List1.AddItem " " + CStr(num) + " " + Query(num)
        ElseIf (num < 100) Then
            List1.AddItem " " + CStr(num) + " " + Query(num)
        ElseIf (num < 1000) Then
            List1.AddItem " " + CStr(num) + " " + Query(num)
        Else
            List1.AddItem CStr(num) + " " + Query(num)
        End If
        k = comma + 1
        List1.ListIndex = List1.ListIndex + 1
        If (num >= pntl) Then Exit For
    Next j
    qry = Space$(200)
    Dummy = DoEvents()
Next i

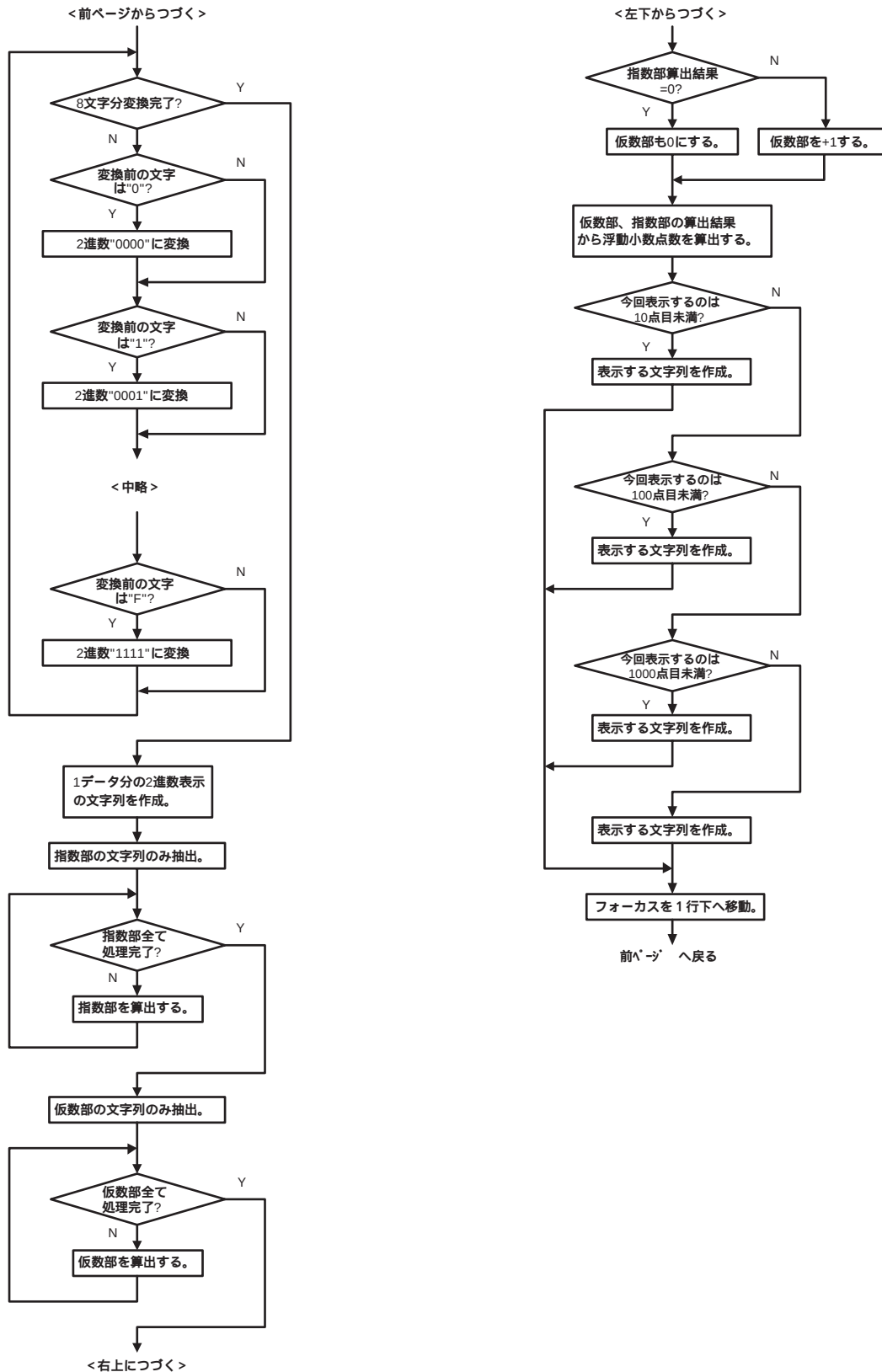
Call ibonl(Dev, 0)
GpibWaveAscii = 0
End Function

```



7.7 波形データ(FLOAT形式)の出力





```

Sample4(GPIB) Get Wave Data (FLOAT)
-----
Private Function GpibWaveFloat() As Integer
    Dim msg As String
    Dim qry As String
    Dim wait As Integer
    Dim eos As Integer
    Dim w As String
    Dim a(8) As String
    Dim b(8) As String
    Dim buf As String
    Dim all As String
    Dim allb As String
    Dim stre As String
    Dim sts As Integer
    Dim pnt1 As Integer
    Dim i As Integer
    Dim j As Integer
    Dim k As Integer
    Dim l As Integer
    Dim m As Integer
    Dim valu As Integer
    Dim vale As Integer
    Dim bufv(2007) As Integer
    Dim valf As Single
    Dim flo As Single

    term = Chr$(10)                                'terminator
    msg = Space$(100)
    qry = Space$(200)

    sts = InitGpib                                  'Initialize GPIB
    If (sts <> 0) Then
        GpibWaveFloat = 1
        Exit Function
    End If

    'Initialize the settings
    msg = "*RST" + term                              'Initialize the settings
    sts = ilwrt(Dev, msg, Len(msg))                  'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveFloat = 1
        Exit Function
    End If

    'Set the measurment condition
    msg = "VOLTAGE:RANGE:ELEMENT1 100V" + term        'Voltage range = 100V
    sts = ilwrt(Dev, msg, Len(msg))                  'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveFloat = 1
        Exit Function
    End If

    msg = "WSETUP:TDIV 10MS" + term                  'Time/div = 10ms
    sts = ilwrt(Dev, msg, Len(msg))                  'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveFloat = 1
        Exit Function
    End If

    msg = "WSETUP:SAMPLING ON" + term                 'Wave sampling start
    sts = ilwrt(Dev, msg, Len(msg))                  'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveFloat = 1
        Exit Function
    End If

    'Set the transition filter used to detect the completion of the data updating
    msg = "STATUS:FILTER1 FALL" + term               'Falling edge of bit0(UPD)
    sts = ilwrt(Dev, msg, Len(msg))                  'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveFloat = 1
        Exit Function
    End If

    'Wait until waveform measure is stable (2 samples in this program)
    For wait = 1 To 2
        'Clear the extended event register (Read and trash the response)
        msg = "STATUS:EESR?" + term
        sts = ilwrt(Dev, msg, Len(msg))              'Send Command
        If (sts < 0) Then
            Call DisplayGPIBError(sts, msg)

```

```

        GpibWaveFloat = 1
        Exit Function
    End If
    sts = ilrd(Dev, qry, Len(qry))          'Receive Query
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveFloat = 1
        Exit Function
    End If
    'Wait for the completion of the data updating
    msg = "COMMUNICATE:WAIT 1" + term
    sts = ilwrt(Dev, msg, Len(msg))        'Send Command
    If (sts < 0) Then
        Call DisplayGPIBError(sts, msg)
        GpibWaveFloat = 1
        Exit Function
    End If
Next wait

'Set conditions for reading the waveform
'FLOAT(MSB first) format, Trace = U1
msg = "WAVEFORM:TRACE U1;FORMAT FLOAT;BYTEORDER MSBFIRST" + term
sts = ilwrt(Dev, msg, Len(msg))          'Send Command
If (sts < 0) Then
    Call DisplayGPIBError(sts, msg)
    GpibWaveFloat = 1
    Exit Function
End If

'Read and display the waveform data
eos = 0
sts = ileos(Dev, eos)                    'Terminator = None(for Binary Data)
If (sts < 0) Then
    Call DisplayGPIBError(sts, "ileos")
    GpibWaveFloat = 1
    Exit Function
End If
'Read in the waveform data
pntl = 1002
msg = "WAVEFORM:START 0;END 1001;SEND?" + term
sts = ilwrt(Dev, msg, Len(msg))          'Send Command
If (sts < 0) Then
    Call DisplayGPIBError(sts, msg)
    GpibWaveFloat = 1
    Exit Function
End If
sts = ilrdi(Dev, bufv(), 6 + 1002 * 4 + 1) 'Receive Query(Integer data)
If (sts < 0) Then
    Call DisplayGPIBError(sts, msg)
    GpibWaveFloat = 1
    Exit Function
End If
eos = &HCOA
sts = ileos(Dev, eos)                    'Terminator = LF
If (sts < 0) Then
    Call DisplayGPIBError(sts, "ileos")
    GpibWaveFloat = 1
    Exit Function
End If

For i = 1 To pntl
    buf = ""
    For j = 1 To 2
        If Left(Right("00" + Hex$(bufv((i * 2) + j)), 4), 2) = "00" Then
            buf = buf + Right("00" + Hex$(bufv((i * 2) + j)), 4)
        ElseIf Left(Right("0" + Hex$(bufv((i * 2) + j)), 4), 1) = "0" Then
            buf = buf + Right("0" + Hex$(bufv((i * 2) + j)), 4)
        Else
            buf = buf + Hex$(bufv(2 + ((i - 1) * 2) + j))
        End If
    Next j

    all = Mid(buf, 3, 2) + Mid(buf, 1, 2) + Mid(buf, 7, 2) + Mid(buf, 5, 2)
    For k = 1 To 8
        a(k) = Mid$(all, k, 1)
        If a(k) = "0" Then b(k) = "0000"
        If a(k) = "1" Then b(k) = "0001"
        If a(k) = "2" Then b(k) = "0010"
        If a(k) = "3" Then b(k) = "0011"
        If a(k) = "4" Then b(k) = "0100"
        If a(k) = "5" Then b(k) = "0101"
        If a(k) = "6" Then b(k) = "0110"
        If a(k) = "7" Then b(k) = "0111"
        If a(k) = "8" Then b(k) = "1000"
        If a(k) = "9" Then b(k) = "1001"
        If a(k) = "A" Then b(k) = "1010"
        If a(k) = "B" Then b(k) = "1011"
        If a(k) = "C" Then b(k) = "1100"
        If a(k) = "D" Then b(k) = "1101"
    Next k

```

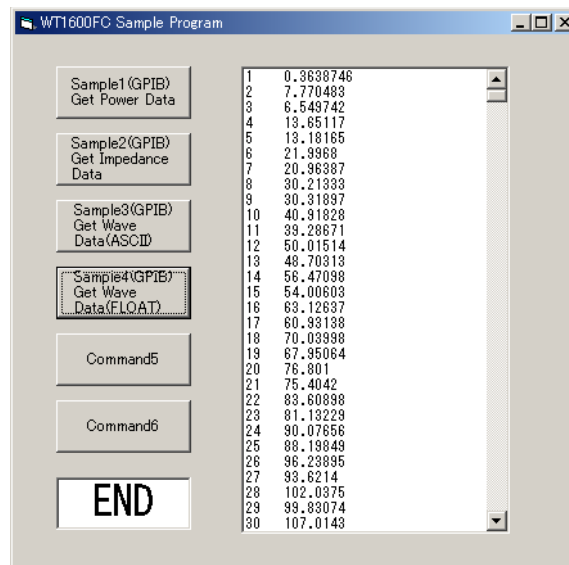
```

        If a(k) = "E" Then b(k) = "1110"
        If a(k) = "F" Then b(k) = "1111"
    Next k
    allb = b(1) + b(2) + b(3) + b(4) + b(5) + b(6) + b(7) + b(8)
    vale = 0
    valf = 0
    valu = Val(Left$(allb, 1))
    stre = Mid$(allb, 2, 8)
    For l = 0 To 7
        vale = vale + (2 ^ l) * Val(Mid$(stre, (8 - l), 1))
    Next l
    w = Mid$(allb, 10, 23)
    For m = 1 To 23
        valf = valf + (2 ^ (-m)) * Val(Mid$(w, m, 1))
    Next m
    If (vale = 0) Then valf = 0 Else: valf = valf + 1
    flo = ((-1) ^ valu) * (2 ^ (vale - 127)) * valf
    If i < 10 Then
        List1.AddItem CStr(i) + "    " + CStr(flo)
    ElseIf i < 100 Then
        List1.AddItem CStr(i) + "  " + CStr(flo)
    ElseIf i < 1000 Then
        List1.AddItem CStr(i) + " " + CStr(flo)
    Else
        List1.AddItem CStr(i) + " " + CStr(flo)
    End If
    List1.ListIndex = List1.ListIndex + 1

    gry = Space$(200)
    Dummy = DoEvents()
Next i

Call ibon1(Dev, 0)
GpibWaveFloat = 0
End Function

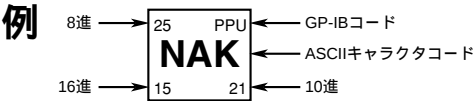
```



付録1 ASCIIキャラクタコード

ここでは、ASCIIのキャラクタコード表を紹介しています。

	0	1	2	3	4	5	6	7
0	0 NUL	20 DEL	40 SP	60 0	100 @	120 P	140 '	160 p
1	1 SOH	21 DC1	41 !	61 1	101 A	121 Q	141 a	161 q
2	2 STX	22 DC2	42 "	62 2	102 B	122 R	142 b	162 r
3	3 ETX	23 DC3	43 #	63 3	103 C	123 S	143 c	163 s
4	4 EOT	24 DC4	44 \$	64 4	104 D	124 T	144 d	164 t
5	5 ENQ	25 NAK	45 %	65 5	105 E	125 U	145 e	165 u
6	6 ACK	26 SYN	46 &	66 6	106 F	126 V	146 f	166 v
7	7 BEL	27 ETB	47 ,	67 7	107 G	127 W	147 g	167 w
8	10 BS	30 CAN	50 (	70 8	110 H	130 X	150 h	170 x
9	11 HT	31 EM	51)	71 9	111 I	131 Y	151 i	171 y
A	12 LF	32 SUB	52 *	72 :	112 J	132 Z	152 j	172 z
B	13 VT	33 ESC	53 +	73 ;	113 K	133 [	153 k	173 {
C	14 FF	34 FS	54 ,	74 <	114 L	134 \	154 l	174 l
D	15 CR	35 GS	55 -	75 =	115 M	135]	155 m	175 }
E	16 SO	36 RS	56 .	76 >	116 N	136 ^	156 n	176 ~
F	17 SI	37 US	57 /	77 ?	117 O	137 _	157 o	177 DEL (RUBOUT)
	F アドレス コマンド	1F ユニバーサル コマンド	2F リスナ アドレス	3F トーカー アドレス	4F 2次 コマンド	5F 2次 コマンド	6F 2次 コマンド	7F 2次 コマンド



付録2 エラーメッセージ

ここでは、通信に関するエラーメッセージについて説明しています。

- 本機器本体には日本語/英語のどちらでも表示することができますが、パーソナルコンピュータなどで読み出したときは英語で表示されます。
- サービスが必要なときは、お買い求め先または横河エンジニアリングサービス(株)まで修理をお申し付けください。
- ここに記載しているエラーメッセージは、通信に関するものに限定しています。通信以外のエラーメッセージについては、ユーザーズマニュアルIM760151-01を参照してください。

通信文法エラー(100～199)

Error in communication command

コード	メッセージ	対処方法	ページ
102	構文が間違っています。 Syntax error	以下のコード以外で構文が間違っています。	4章， 5章
103	<DATA SEPARATOR>がありません。 Invalid separator	データとデータは「,」(カンマ)で区切ってください。	4-1
104	<DATA>の種類が間違っています。 Data type error	4-5～4-6ページを参照して、正しいデータ形式で記述してください。	4-5～ 4-6
108	<DATA>が多すぎます。 Parameter not allowed	データの数を確認してください。	4-5， 5章
109	必要な<DATA>がありません。 Missing parameter	必要なデータを記述してください。	4-5， 5章
111	<HEADER SEPARATOR>がありません。 Header separator error	ヘッダとデータはスペースで区切ってください。	4-1
112	<mnemonic>が長すぎます。 Program mnemonic too long	ニモニック(アルファベットと数字からなる文字列)を確認してください。	5章
113	そのような命令はありません。 Undefined header	ヘッダを確認してください。	5章
114	<HEADER>の数値が間違っています。 Header suffix out of range	ヘッダを確認してください。	5章
120	数値の仮数部分がありません。 Numeric data error	<NRf>形式のときは数字が必要です。	4-5
123	指数が大きすぎます。 Exponent too large	<NR3>形式のときの「E」のあとの指数を小さくしてください。	4-5， 5章
124	有効桁数が多すぎます。 Too many digits	数字は255桁以内にしてください。	4-5， 5章
128	数値データは使えません。 Numeric data not allowed	<NRf>形式以外のデータ形式で記述してください。	4-5， 5章
131	単位が間違っています。 Invalid suffix	<電圧>，<時間>，<周波数>の単位を確認してください。	4-5
134	単位のつづりが長すぎます。 Suffix too long	<電圧>，<時間>，<周波数>の単位を確認してください。	4-5
138	単位は使えません。 Suffix not allowed	<電圧>，<時間>，<周波数>以外では単位は使えません。	4-5
141	そのような選択肢はありません。 Invalid character data	{...}の中にある文字列を記述してください。	5章
144	<CHARACTER DATA>のつづりが長すぎます。 Character data too long	{...}の文字列のつづりを確認してください。	5章

コード	メッセージ	対処方法	ページ
148	<CHARACTER DATA>は使えません。 Character data not allowed	{...}以外のデータ形式で記述してください。	5章
150	<STRING DATA>の右の区切りがありません。 String data error	<文字列>の場合は「"」または「'」で囲ってください。	4-6
151	<STRING DATA>の内容が不適当です。 Invalid string data	<文字列>が長すぎるか、使用不可能な文字があります。	5章
158	<STRING DATA>は使えません。 String data not allowed	<文字列>以外のデータ形式で記述してください。	5章
161	<BLOCK DATA>のデータ長が合っていない。 Invalid block data	<ブロックデータ>は使用できません。	4-6 , 5章
168	<BLOCK DATA>は使えません。 Block data not allowed	<ブロックデータ>は使用できません。	4-6 , 5章
171	<EXPRESSION DATA>の中に許されない文字があります。 Invalid expression	演算式は使用できません。	5章
178	<EXPRESSION DATA>は使えません。 Expression data not allowed	演算式は使用できません。	5章
181	プレースホルダがマクロの外にあります。 Invalid outside macro definition	IEEE488.2のマクロ機能には対応していません。	-

通信実行エラー(200 ~ 299)

Error in communication execution

コード	メッセージ	対処方法	ページ
221	設定内容に矛盾があります。 Setting conflict	関連のある設定値を確認してください。	5章
222	データの値が範囲外です。 Data out of range	設定範囲を確認してください。	5章
223	データのバイト長が長すぎます。 Too much data	データのバイト長を確認してください。	5章
224	データの値が不適当です。 Illegal parameter value	設定範囲を確認してください。	5章
241	ハードウェアが実装されていません。 Hardware missing	オプションの有無を確認してください。	-
260	<EXPRESSION DATA>が間違っています。 Expression error	演算式は使用できません。	-
270	マクロのネストが深すぎます。 Macro error	IEEE488.2のマクロ機能には対応していません。	-
272	マクロでは使用できません。 Macro execution error	IEEE488.2のマクロ機能には対応していません。	-
273	マクロラベルが不適当です。 Illegal macro label	IEEE488.2のマクロ機能には対応していません。	-
275	マクロが長すぎます。 Macro definition too long	IEEE488.2のマクロ機能には対応していません。	-
276	マクロが再帰呼び出しされました。 Macro recursion error	IEEE488.2のマクロ機能には対応していません。	-
277	マクロの二重定義はできません。 Macro redefinition not allowed	IEEE488.2のマクロ機能には対応していません。	-
278	そのようなマクロは定義されていません。 Macro header not found	IEEE488.2のマクロ機能には対応していません。	-

通信クエリエラー(400 ~ 499) Error in communication Query

コード	メッセージ	対処方法	ページ
410	応答の送信が中断されました。 Query INTERRUPTED	送受信の順序を確認してください。	4-2
420	送信できる応答がありません。 Query UNTERMINATED	送受信の順序を確認してください。	4-2
430	送受信がデッドロックしました。 送信を中止します。 Query DEADLOCKED	プログラムメッセージは<PMT>も含めて 1024バイト以下にしてください。	4-2
440	応答を要求する順番が間違っています。 Query UNTERMINATED after indefinite response	*IDN?, *OPT?の後ろにはクエリを記述 しないでください。	-

システムエラー(通信) (912 ~ 914) Error in System Operation

コード	メッセージ	対処方法	ページ
912	通信ドライバエラー Fatal error in Communication-driver	サービスが必要です。	-

警告(通信) (5) Warning

コード	メッセージ	対処方法	ページ
5	*OPC/?がメッセージの途中にあります。 *OPC/? exists in message	*OPCまたは*OPC?は、プログラムメッ セージの最後においてください。	-

その他(350 , 390)

コード	メッセージ	対処方法	ページ
350	Queue overflow	エラーキューを読み出してください。	6-5
390	Overrun error(シリアル(RS-232)のみ)	ボーレートを下げて実行してください。	-

Note

コード350はエラーキューがあふれたときに発生します。STATus:ERRor?クエリにのみ出力さ
れるエラーで、画面には表示されません。

付録3 IEEE 488.2-1987について

本機器のGP-IBインタフェースは、IEEE 488.2-1987規格に準じています。この規格では、以下の23の項目について「ドキュメントに記載しなければならない」としています。ここでは、これらについて説明しています。

- (1) IEEE 488.1インタフェース機能のうち、サポートしているサブセット
「1.4 GP-IBインタフェースの仕様」を参照してください。
- (2) アドレスが0～30以外に設定されたときのデバイスの動作
本機器では、アドレスを0～30以外に設定することはできません。
- (3) ユーザがアドレス変更をしたときの動作
アドレスの変更はMISCキーメニューでアドレスを設定した時点で行われます。設定したアドレスは、次に変更するまで有効です。
- (4) 電源ON時のデバイスのセッティング。電源ON時に使用可能なコマンド
基本的には、以前の設定(その前に電源をOFFにしたときの設定)になります。
電源ON時に実行を制限されるコマンドはありません。
- (5) メッセージ交換のオプション
 - (a) 入力バッファのサイズ
1024バイト
 - (b) 複数の応答メッセージユニットを返すクエリ
5章の各コマンドの例を参照してください。
 - (c) 構文解析時に応答データを作成するクエリ
すべてのクエリは、構文を解析すると応答データを作成します。
 - (d) 受信時に応答データを作成するクエリ
コントローラが受信する時点で応答データを作成するクエリはありません。
 - (e) 制限しあうパラメータを有するコマンド
5章の各コマンドの例を参照してください。
- (6) コマンドを構成する機能エレメントおよび複合ヘッダのエレメントに含まれるもの
4章および5章を参照してください。
- (7) ブロックデータの転送に影響するバッファのサイズ
ブロックデータの送信時には、そのサイズに合わせて出力キューを拡張します。
- (8) 演算式で使えるプログラムデータのエレメントの一覧と、そのネストの制限
演算式は使えません。
- (9) 各問い合わせに対する応答の構文
5章の各コマンドの例を参照してください。
- (10) 応答の文法に従わないデバイス間の通信について
ありません。

- (11) 応答データのブロックデータのサイズ
1 ~ 308922バイト
- (12) サポートしている共通コマンドの一覧
「5.20 共通コマンドグループ」を参照してください。
- (13) キャリブレーション正常終了時のデバイスの状態
測定を実行中の状態になります。
- (14) *DDTのトリガマクロの定義で使えるブロックデータの最大長
サポートしていません。
- (15) マクロ定義のマクロラベルの最大長, マクロ定義で使えるブロックデータの最大長, マクロ定義で再帰を使ったときの処理
マクロ機能は対応していません。
- (16) *IDN?に対する返送
「5.20 共通コマンドグループ」を参照してください。
- (17) *PUD, *PUD?のプロテクトユーザデータの保存エリアのサイズ
*PUD, *PUD?はサポートしていません。
- (18) *RDT, *RDT?のリソース名の長さ
*RDT, *RDT?はサポートしていません。
- (19) *RST, *LRN?, *RCL, *SAVによる状態の変化
*RST
「5.20 共通コマンドグループ」を参照してください。

*LRN?, *RCL, *SAV
これらの共通コマンドはサポートしていません。
- (20) *TST?によるセルフテストの実行範囲
MISCのSelf TestメニューのMEMORYテストのすべて(内部の各メモリ)を実行します。
- (21) 拡張されたリターンステータスの構造
6章を参照してください。
- (22) 各コマンドの処理がオーバーラップするか, シーケンシャルに行われるか
「4.5 コントローラとの同期」および5章を参照してください。
- (23) 各コマンドの実行内容
5章の各コマンドの機能とユーザーズマニュアルIM760151-01を参照してください。

索引

記号

3P3W -> 3V3A 変換 5-59

B

Baud Rate 2-8
BMP フォーマット 5-31
Boolean 4-6

C

CCITT 2-4
COMMunicate グループ 5-10
Communication 1-5
CS-RS 2-6
CURSor グループ 5-12

D

DCL 1-6
Delta -> Star 変換 5-59
DISPlay グループ 5-15

F

FFT 窓幅 5-39
Filename 4-6
FILE グループ 5-25
Format 2-8

G

GET 1-6
GP-IB インタフェースの機能 1-3
GP-IB インタフェースの仕様 1-4
GP-IB ケーブル 1-2
GTL 1-6

H

HCOPy グループ 5-30
HOLD グループ 5-34

I

IEEE 488.2-1987 について 付-5
IFC 1-6
IMAGe グループ 5-35
IMPedance グループ 5-36
INPUt グループ 5-43
INTEGrate グループ 5-53

L

LCD 画面 5-77
LLO 1-6

M

MAX HOLD 5-60
MEASure グループ 5-57
MISC 1-5

N

NL^END 4-1
NULL 機能 5-49
NUMeric グループ 5-62

O

OFF-OFF 2-5

P

Pc(Corrected Power) 5-60
PMT 4-1

R

RATE グループ 5-68
Register 4-6
REN 1-6
RMT 4-1
RS-232 規定の信号一覧 2-4
Rx-Tx 2-8

S

SCSI-ID 5-78
SDC 1-6
SDC と DCL の違い 1-7
SINGLE 5-86
SPD 1-6
SPE 1-6
Star -> Delta 変換 5-59
STATus グループ 5-69
STORe グループ 5-71
SYSTem グループ 5-76

T

TCP/IP 3-5
Terminator 2-8
TIFF フォーマット 5-34
Time/div 5-82

V

Visual Basic 7-1

W

WAVeform グループ 5-79

WSETup (Wave SETup)グループ 5-81

X

XON-RS 2-6

XON-XON 2-5

ア

空き容量 5-27

圧縮 5-31

アドレス 1-5

アドレスコマンド 1-7

アペレーシング 5-58

イ

イーサネットインタフェース 3-2

イーサネットコントロールの設定 3-4

位相差 5-61

イニシャライズ 5-78

イネーブルレジスタ 6-2

インタフェースメッセージ 1-6

インピーダンス測定用負荷電流 5-40

インピーダンス目安値 5-41

エ

エラーキュー 5-70, 6-2, 6-5

エラーキューのクリア 5-84

エラーメッセージ 付-2

エレメント別積算 5-54

オ

オートキャリブレーション 5-54

オートレンジ 5-45, 5-50

オーバラップイネーブルレジスタ 5-11

オーバラップコマンド 4-7

応答データ 4-2

応答ヘッダ 4-2

応答メッセージ 4-1

オプション 5-85

オペレーションペンディングステータスレジスタ 5-11

カ

カーソル位置 5-13

カーソル測定 5-13

カーソルの対象 5-13, 5-14

カーソルパス 5-13

カーソル表示 5-13, 5-14

拡張イベントイネーブルレジスタ 5-69, 6-2

拡張イベントレジスタ 5-69, 6-2, 6-4

各部の名称 1-1, 2-1, 3-1

カットオフ周波数 5-48

紙送り 5-33

画面イメージ 5-31, 5-35

画面表示 5-18

カレントディレクトリ 5-26

キ

記号	ii
機種	5-85
輝度	5-77
基本単位	4-5
キャリブレーション	5-84
共通コマンドグループ	5-84
共通コマンドヘッダ	4-3

ク

クエリ	4-4
拡張イベントレジスタ	4-8
グラフィカル	5-23

ケ

結線方式	5-52
言語	7-1

コ

工場出荷時の設定値	5-85
構文の記号	ii
コネクタと信号名(シリアル)	2-3
コマンド一覧表	5-1
コメント	5-27, 5-33, 5-72
コントローラとの同期	4-7
コンバート	5-73

サ

サービスリクエストイネーブルレジスタ	5-86, 6-2
削除	5-26
サンプルプログラムイメージ	7-2
サンプルレート	5-80

シ

シーケンシャルコマンド	4-7
色調	5-31, 5-34, 5-35
時刻	5-78
システム	5-76
実時間ストアモード	5-75
実時間制御標準積算モード	5-54
周波数測定	5-59
周波数レンジ	5-41
受信機能	2-2, 3-2
受信バッファ	2-6
出力キュー	6-2, 6-5
出力形式	5-35
出力タイプ	5-41
上位クエリ	4-4
乗数	4-5
状態レジスタ	4-8, 5-69, 6-4
省略形	4-4
初期化	5-73, 5-85

シリアルインタフェースの仕様	2-2
シリアル通信	2-8
シリアルポール	5-70, 6-3

ス

ズーム率	5-83
垂直ポジション	5-82
数値データの出力	5-63
数値データのフォーマット	5-66
数値表示	5-18
数値表示の方式	5-19
スケーリング	5-49
スケーリング定数	5-47
スケール値表示	5-24
ステータスバイト	6-2
ステータスバイトレジスタ	5-86
ステータスレポート	6-1
ストア	5-72
ストアインタバル	5-73
ストア回数	5-72
ストア先	5-72
ストアメモリ	5-73
ストアモード	5-75

セ

制御信号	5-39, 5-40
積算タイム	5-56
積算のスタート	5-56
積算のストップ	5-56
積算モード	5-54
絶対パス	5-27
接続	3-3
セルフテスト	5-86
ゼロクロスフィルタ	5-48
遷移フィルタ	5-70, 6-4

ソ

送信機能	2-2, 3-2
測定モードの種類	5-39
ソフトウェアハンドシェーク	2-5

タ

ターミネータ	2-8
対象PC	7-1
対象ドライブ	5-26
タイムアウト時間	3-5, 3-6
単位	4-5
単純ヘッダ	4-3

チ

直接入力	5-47
直流成分	5-41
直流負荷電流	5-38

テ

データ	4-5
データ更新レート	5-68
データバイトの並び	4-6
データ番号	5-74
データフォーマット	2-7, 2-8
ディレクトリ	5-27
デッドロック	4-2
デルタ演算	5-59
電圧センス入力	5-41
電圧測定	5-50
電圧入力端子	5-42
電圧レンジ	5-42, 5-51
電流センサ入力	5-47
電流測定	5-45
電流モード	5-54
電流レンジ	5-38, 5-46

ト

トーカ機能	1-3
同期ソース	5-50
トリガ	5-82
トレンド	5-21

ナ

内蔵ハードディスクのモータ	5-78
内蔵プリンタでの印字	5-32

ニ

入力エレメント	5-45
入力エレメントタイプ	5-49

ハ

ハードウェアハンドシェーク	2-6
バイト出力順序	5-79
波形観測	5-82
波形サンプリング	5-82
波形の全データ点数	5-80
波形の割り付け方法	5-23
波形表示	5-23
波形表示データ	5-79
波形ラベル名表示	5-24
パスワード	3-5, 3-6
ハンドシェーク	2-5

ヒ

ピークオーバ	5-49
日付	5-76
表示色	5-77
表示色のモード	5-78
表示フォーマット	5-23
表示方式	5-18
標準イベントイネーブルレジスタ	5-84, 6-2
標準イベントレジスタ	5-85, 6-2, 6-3
標準積算モード	5-54

フ

ファイル操作	5-26
ファイルの保存	5-27
ファイル名	5-27, 5-33, 5-72
ファンクション選択肢(<Function>)一覧	5-24
負荷電流の ON/OFF	5-39, 5-41
負荷電流の周波数	5-40, 5-41
負荷電流の振幅	5-40
負荷電流の電流値	5-39
負荷電流の波形	5-40
複合ヘッダ	4-3
物理的な次元を持ったデータ	4-5
プリセットパターン	5-67
プログラムデータ	4-1
プログラムヘッダ	4-1
プログラムメッセージ	4-1
ブロックデータ	4-6
フロッピーディスクのフォーマット	5-26
プロトコル	1-4
フロントパネル	1-1, 2-1, 3-1
分割フォーマット	5-23

ヘ

ヘッダ	4-1
ヘッダの解釈	4-4

ホ

ホールド	5-34, 5-39, 5-40
ボーレート	2-8
補間方式	5-23

マ

マスク	6-2, 6-3
マニュアルで使用している記号	ii
マルチラインメッセージ	1-7

ム

無効電力	5-61
無償ソフトウェア	3-6

メ

メッセージの言語 5-77

モ

文字データ 4-6

文字列データ 4-6

ユ

ユーザー認証機能 3-3

ユーザー名 3-5, 3-6

ユーザ定義ファンクション 5-60

ユニバーサルコマンド 1-7

ユニラインメッセージ 1-6

ヨ

横軸(T/div)(トレンドの) 5-22

ラ

ラインフィルタ 5-48

リ

リアパネル 1-1, 2-1, 3-1

リコール 5-72

リスタート 5-22

リスナ機能 1-3

リモート/ローカル切り替え 1-3, 2-2, 3-2

レ

レジスタのクリア 5-85

レンジモード 5-42

ロ

ローカルロックアウト 5-10

ロード 5-27

コマンドリスト

:COMMunicate:HEADer 5-10

:COMMunicate:LOCKout 5-10

:COMMunicate:OPSE 5-11

:COMMunicate:OPSR? 5-11

:COMMunicate:OVERlap 5-11

:COMMunicate:REMOte 5-11

:COMMunicate:STATus? 5-11

:COMMunicate:VERBoSe 5-11

:COMMunicate:WAIT 5-11

:COMMunicate:WAIT? 5-11

:COMMunicate? 5-10

:CURSor:TRENd: 5-13

:CURSor:TRENd:POSition<x> 5-13

:CURSor:TRENd:TRACe<x> 5-13

:CURSor:TRENd? 5-13

:CURSor:TRENd[:STATe] 5-13

:CURSor:WAVE: 5-14

:CURSor:WAVE:PATH 5-13

:CURSor:WAVE:POSition<x> 5-13

:CURSor:WAVE:TRACe<x> 5-14

:CURSor:WAVE? 5-13

:CURSor:WAVE[:STATe] 5-14

:CURSor? 5-13

:DISPlay:FORMat 5-18

:DISPlay:NUMeric? 5-18

:DISPlay:TRENd:ALL 5-20

:DISPlay:TRENd:FORMat 5-20

:DISPlay:TRENd:NORMal:ITEM<x>:SCALing:MODE 5-21

:DISPlay:TRENd:NORMal:ITEM<x>:SCALing:VALue 5-22

:DISPlay:TRENd:NORMal:ITEM<x>:SCALing? 5-21

:DISPlay:TRENd:NORMal:ITEM<x>? 5-21

:DISPlay:TRENd:NORMal:ITEM<x>[:FUNCTion] 5-21

:DISPlay:TRENd:NORMal? 5-21

:DISPlay:TRENd:PDIV 5-22

:DISPlay:TRENd:RESTart 5-22

:DISPlay:TRENd:T<x> 5-22

:DISPlay:TRENd:TDIV 5-22

:DISPlay:TRENd? 5-20

:DISPlay:TRENd[:SAMPLing] 5-22

:DISPlay:WAVE: 5-24

:DISPlay:WAVE:ALL 5-23

:DISPlay:WAVE:FORMat 5-23

:DISPlay:WAVE:GRATicule 5-23

:DISPlay:WAVE:INTerpolate 5-23

:DISPlay:WAVE:MAPPing: 5-23

:DISPlay:WAVE:MAPPing? 5-23

:DISPlay:WAVE:MAPPing[:MODE] 5-23

:DISPlay:WAVE:SVALue 5-24

:DISPlay:WAVE:TLABel 5-24

:DISPlay:WAVE? 5-23

:DISPlay? 5-18

:DISPlay[:NUMeric]:IMPedance:ICURsor 5-18

:DISPlay[:NUMeric]:IMPedance:OBJect 5-18

:DISPlay[:NUMeric]:IMPedance:TYPE 5-18

:DISPlay[:NUMeric]:IMPedance? 5-18

:DISPlay[:NUMeric]:NORMal:FCURsor	5-19	:HCOPy:TIFF:COLor	5-34
:DISPlay[:NUMeric]:NORMal:IAMount	5-19	:HCOPy:TIFF?	5-34
:DISPlay[:NUMeric]:NORMal:ICURsor	5-19	:HCOPy?	5-31
:DISPlay[:NUMeric]:NORMal:ITEM<x>	5-20	:HOLD	5-34
:DISPlay[:NUMeric]:NORMal:PRESet	5-20	:IMAGe:COLor	5-35
:DISPlay[:NUMeric]:NORMal?	5-19	:IMAGe:FORMat	5-35
:FILE:CDIRectory	5-26	:IMAGe:SEND?	5-35
:FILE:DELete:IMAGe:	5-26	:IMAGe?	5-35
:FILE:DELete:NUMeric:	5-26	:IMPedance:CURRent:MRANge?	5-38
:FILE:DELete:SETup	5-26	:IMPedance:DCControl:DETaile:HOLD	5-39
:FILE:DELete:WAVE:	5-26	:IMPedance:DCControl:DETaile:LIMit	5-39
:FILE:DRIVE	5-26	:IMPedance:DCControl:DETaile:RATio	5-39
:FILE:FORMat	5-26	:IMPedance:DCControl:DETaile?	5-38
:FILE:FREE?	5-27	:IMPedance:DCControl:OFFSet	5-39
:FILE:LOAD:ABORt	5-27	:IMPedance:DCControl:OUTPut	5-39
:FILE:LOAD:FGWave	5-27	:IMPedance:DCControl?	5-38
:FILE:LOAD:SETup	5-27	:IMPedance:MEASure:ANALysis?	5-39
:FILE:MDIRectory	5-27	:IMPedance:MEASure:ARRay?	5-39
:FILE:PATH?	5-27	:IMPedance:MEASure:TYPE	5-39
:FILE:SAVE:ABORt	5-27	:IMPedance:MEASure?	5-39
:FILE:SAVE:ANAMing	5-27	:IMPedance:SUPERpose:AMPLitude	5-40
:FILE:SAVE:COMMeNt	5-27	:IMPedance:SUPERpose:DETaile:HOLD	5-40
:FILE:SAVE:NUMeric:NORMal:	5-28, 5-29	:IMPedance:SUPERpose:DETaile:LIMit	5-40
:FILE:SAVE:NUMeric:NORMal:ALL	5-28	:IMPedance:SUPERpose:DETaile:RATio	5-40
:FILE:SAVE:NUMeric:NORMal:PRESet<x>	5-28	:IMPedance:SUPERpose:DETaile:WAVEform	5-40
:FILE:SAVE:NUMeric:NORMal?	5-28	:IMPedance:SUPERpose:DETaile?	5-40
:FILE:SAVE:NUMeric:TYPE	5-29	:IMPedance:SUPERpose:FREQuency:RANGe	5-41
:FILE:SAVE:NUMeric?	5-28	:IMPedance:SUPERpose:FREQuency:VALue	5-41
:FILE:SAVE:NUMeric[:EXECute]	5-28	:IMPedance:SUPERpose:FREQuency?	5-40
:FILE:SAVE:SETup[:EXECute]	5-29	:IMPedance:SUPERpose:OFFSet	5-41
:FILE:SAVE:WAVE:TRACe	5-29	:IMPedance:SUPERpose:OUTPut:TYPE	5-41
:FILE:SAVE:WAVE:TYPE	5-29	:IMPedance:SUPERpose:OUTPut?	5-41
:FILE:SAVE:WAVE?	5-29	:IMPedance:SUPERpose:OUTPut[:STATe]	5-41
:FILE:SAVE:WAVE[:EXECute]	5-29	:IMPedance:SUPERpose?	5-40
:FILE:SAVE?	5-27	:IMPedance:VOL TAge:ESTimate:ELEMent<x>	5-42
:FILE?	5-26	:IMPedance:VOL TAge:ESTimate?	5-41
:HCOPy:ABORt	5-31	:IMPedance:VOL TAge:ESTimate[:ALL]	5-41
:HCOPy:BMP:COLor	5-31	:IMPedance:VOL TAge:INITialize	5-42
:HCOPy:BMP:COMPression	5-31	:IMPedance:VOL TAge:MRANge?	5-42
:HCOPy:BMP?	5-31	:IMPedance:VOL TAge:RANGe:ELEMent<x>	5-42
:HCOPy:COMMeNt	5-31	:IMPedance:VOL TAge:RANGe?	5-42
:HCOPy:DIRectioN	5-32	:IMPedance:VOL TAge:RANGe[:ALL]	5-42
:HCOPy:EXECute	5-32	:IMPedance:VOL TAge:TERMinal:ELEMent<x>	5-42
:HCOPy:FORMat	5-32	:IMPedance:VOL TAge:TERMinal?	5-42
:HCOPy:PRINter:DLISt:INFORmation	5-32	:IMPedance:VOL TAge:TERMinal[:ALL]	5-42
:HCOPy:PRINter:DLISt:NORMal:	5-33	:IMPedance:VOL TAge?	5-41
:HCOPy:PRINter:DLISt:NORMal:ALL	5-33	:IMPedance?	5-38
:HCOPy:PRINter:DLISt:NORMal:PRESet<x>	5-33	:IMPedance[:STATe]	5-40
:HCOPy:PRINter:DLISt:NORMal?	5-32	:INPut?	5-45
:HCOPy:PRINter:DLISt?	5-32	:INTEGrate:ACAL	5-54
:HCOPy:PRINter:DLISt[:EXECute]	5-32	:INTEGrate:CURRent:ELEMent<x>	5-54
:HCOPy:PRINter:FEED	5-33	:INTEGrate:CURRent?	5-54
:HCOPy:PRINter?	5-32	:INTEGrate:CURRent[:ALL]	5-54
:HCOPy:SAVE:ANAMing	5-33	:INTEGrate:INDependent	5-54
:HCOPy:SAVE:COMMeNt	5-33	:INTEGrate:MODE	5-54
:HCOPy:SAVE:NAME	5-34	:INTEGrate:RESet	5-55
:HCOPy:SAVE?	5-33	:INTEGrate:RTIME<x>:	5-55

:INTEGrate:RTIME<x>?	5-55	:STORE:FILE?	5-72
:INTEGrate:START	5-56	:STORE:INTerval	5-73
:INTEGrate:STATE?	5-56	:STORE:ITEM	5-73
:INTEGrate:STOP	5-56	:STORE:MEMory:CONVert:ABORt	5-73
:INTEGrate:TIMer<x>	5-56	:STORE:MEMory:CONVert:EXECute	5-73
:INTEGrate?	5-54	:STORE:MEMory:INITialize	5-73
:MEASure:AVERaging:COUNT	5-58	:STORE:MODE	5-73
:MEASure:AVERaging:TYPE	5-59	:STORE:NUMeric:NORMal:	5-74
:MEASure:AVERaging?	5-58	:STORE:NUMeric:NORMal:ALL	5-74
:MEASure:AVERaging[:STATE]	5-58	:STORE:NUMeric:NORMal:PRESet<x>	5-74
:MEASure:DMeasure:OBJect	5-59	:STORE:NUMeric:NORMal?	5-74
:MEASure:DMeasure:TYPE	5-59	:STORE:NUMeric?	5-74
:MEASure:DMeasure?	5-59	:STORE:RECall	5-74
:MEASure:FREQuency:ITEM	5-59	:STORE:RTIME:	5-75
:MEASure:FREQuency?	5-59	:STORE:RTIME?	5-75
:MEASure:FUNCTion<x>:EXPReSSion	5-60	:STORE:SMODE	5-75
:MEASure:FUNCTion<x>:UNIT	5-60	:STORE:START	5-75
:MEASure:FUNCTion<x>?	5-60	:STORE:STOP	5-75
:MEASure:FUNCTion<x>[:STATE]	5-60	:STORE:WAVE:	5-75
:MEASure:MHOLd	5-60	:STORE:WAVE:ALL	5-75
:MEASure:PC:IEC	5-60	:STORE:WAVE?	5-75
:MEASure:PC:P<x>	5-61	:STORE?	5-72
:MEASure:PC?	5-60	:SYSTem:DATE	5-76
:MEASure:PHASe	5-61	:SYSTem:LANGuage	5-77
:MEASure:SFORMula	5-61	:SYSTem:LCD:BRIGhtness	5-77
:MEASure?	5-58	:SYSTem:LCD:COLor:GRAPh:	5-77
:NUMeric:FORMat	5-63	:SYSTem:LCD:COLor:GRAPh:MODE	5-77
:NUMeric:IMPedance:ARRay	5-63	:SYSTem:LCD:COLor:GRAPh?	5-77
:NUMeric:IMPedance:CLEar	5-63	:SYSTem:LCD:COLor:TEXT:	5-78
:NUMeric:IMPedance:ITEM<x>	5-64	:SYSTem:LCD:COLor:TEXT:MODE	5-78
:NUMeric:IMPedance:NUMber	5-64	:SYSTem:LCD:COLor:TEXT?	5-77
:NUMeric:IMPedance:PRESet	5-64	:SYSTem:LCD:COLor?	5-77
:NUMeric:IMPedance:VALue?	5-64	:SYSTem:LCD?	5-77
:NUMeric:IMPedance?	5-63	:SYSTem:SCSI:HDMotor	5-78
:NUMeric:NORMal?	5-64	:SYSTem:SCSI:INITialize	5-78
:NUMeric?	5-63	:SYSTem:SCSI:INTernalid	5-78
:NUMeric[:NORMal]:CLEar	5-65	:SYSTem:SCSI:OWNid	5-78
:NUMeric[:NORMal]:ITEM<x>	5-65	:SYSTem:SCSI?	5-78
:NUMeric[:NORMal]:NUMber	5-65	:SYSTem:TIME	5-78
:NUMeric[:NORMal]:PRESet	5-65	:SYSTem?	5-76
:NUMeric[:NORMal]:VALue?	5-65	:WAVEform:BYTeorder	5-79
:RATE	5-68	:WAVEform:END	5-79
:STATus:CONDition?	5-69	:WAVEform:FORMat	5-80
:STATus:EESe	5-69	:WAVEform:LENGth?	5-80
:STATus:EESR?	5-69	:WAVEform:SEND?	5-80
:STATus:ERRor?	5-70	:WAVEform:SRATe?	5-80
:STATus:FILTer<x>	5-70	:WAVEform:START	5-80
:STATus:QENable	5-70	:WAVEform:TRACe	5-80
:STATus:QMESsage	5-70	:WAVEform:TRIGger?	5-80
:STATus:SPOLI?	5-70	:WAVEform?	5-79
:STATus?	5-69	:WSETup:POSition:	5-82
:STORE:COUNT	5-72	:WSETup:POSition?	5-82
:STORE:DIRection	5-72	:WSETup:TDiv	5-82
:STORE:FILE:ANAMing	5-72	:WSETup:TRIGger:LEVel	5-82
:STORE:FILE:COMMeNt	5-72	:WSETup:TRIGger:MODE	5-82
:STORE:FILE:NAME	5-73	:WSETup:TRIGger:SLOPe	5-82

:WSEtup:TRIGger:SOURce	5-83	*STB?	5-86
:WSEtup:TRIGger?	5-82	*TRG	5-86
:WSEtup:VZoom:	5-83	*TST?	5-86
:WSEtup:VZoom?	5-83	*WAI	5-86
:WSEtup?	5-82		
:WSEtup[:SAMPling]	5-82		
[:INPut]:CURRent:AUTO:ELEMent<x>	5-46		
[:INPut]:CURRent:AUTO[:ALL]	5-45		
[:INPut]:CURRent:MRANge?	5-46		
[:INPut]:CURRent:RANGe:ELEMent<x>	5-47		
[:INPut]:CURRent:RANGe?	5-46		
[:INPut]:CURRent:RANGe[:ALL]	5-46		
[:INPut]:CURRent:SRATio:ELEMent<x>	5-47		
[:INPut]:CURRent:SRATio?	5-47		
[:INPut]:CURRent:SRATio[:ALL]	5-47		
[:INPut]:CURRent:TERMinal:ELEMent<x>	5-48		
[:INPut]:CURRent:TERMinal?	5-47		
[:INPut]:CURRent:TERMinal[:ALL]	5-47		
[:INPut]:CURRent?	5-45		
[:INPut]:FILTer:LINE?	5-48		
[:INPut]:FILTer:ZCRoss:ELEMent<x>	5-48		
[:INPut]:FILTer:ZCRoss?	5-48		
[:INPut]:FILTer:ZCRoss[:ALL]	5-48		
[:INPut]:FILTer?	5-48		
[:INPut]:FILTer[:LINE]:ELEMent<x>	5-48		
[:INPut]:FILTer[:LINE][:ALL]	5-48		
[:INPut]:MODUle?	5-49		
[:INPut]:NULL	5-49		
[:INPut]:POVer?	5-49		
[:INPut]:SCALing:	5-49		
[:INPut]:SCALing:STATe?	5-50		
[:INPut]:SCALing?	5-49		
[:INPut]:SCALing[:STATe]:ELEMent<x>	5-50		
[:INPut]:SCALing[:STATe][:ALL]	5-50		
[:INPut]:SYNChronize:ELEMent<x>	5-50		
[:INPut]:SYNChronize?	5-50		
[:INPut]:SYNChronize[:ALL]	5-50		
[:INPut]:VOLTage:AUTO:ELEMent<x>	5-51		
[:INPut]:VOLTage:AUTO[:ALL]	5-50		
[:INPut]:VOLTage:MRANge?	5-51		
[:INPut]:VOLTage:RANGe:ELEMent<x>	5-51		
[:INPut]:VOLTage:RANGe?	5-51		
[:INPut]:VOLTage:RANGe[:ALL]	5-51		
[:INPut]:VOLTage?	5-50		
[:INPut]:WIRing	5-52		
*CAL?	5-84		
*CLS	5-84		
*ESE	5-84		
*ESR?	5-85		
*IDN?	5-85		
*OPC	5-85		
*OPC?	5-85		
*OPT?	5-85		
*PSC	5-85		
*RST	5-85		
*SRE	5-86		