

WT3000

プレシジョンパワーアナライザ
通信インタフェース

USER'S MANUAL

ユーザーズマニュアル

はじめに

このたびは、WT3000プレジジョンパワーアナライザをお買い上げいただきましてありがとうございます。

この通信インタフェースユーザーズマニュアルは、GP-IB/RS-232/USB/イーサネットインタフェースの機能や通信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、ご使用時にすぐにご覧になれるところに、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、WT3000のマニュアルは、このマニュアルを含め3冊あります。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
WT3000プレジジョンパワーアナライザ ユーザーズマニュアル (Vol 1/3)	IM 760301-01	WT3000の通信機能と拡張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。
WT3000プレジジョンパワーアナライザ 通信インタフェース ユーザーズマニュアル(CD-ROM) (Vol 2/3)	IM 760301-17	本書です。通信コマンドを使って本機器を制御する機能について説明しています。
WT3000プレジジョンパワーアナライザ 拡張機能ユーザーズマニュアル (Vol 3/3)	IM 760301-51	本機器の拡張機能(オプション)とその操作方法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。また、実際の表示内容が本書に記載の表示内容と多少異なることがあります。
- ・ 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、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。
- ・ 本製品のTCP/IPソフトウェア、およびTCP/IPソフトウェアに関するドキュメントは、カリフォルニア大学からライセンスされたBSD Networking Software, Release 1をもとに当社で開発/作成したものです。

USBインタフェースおよびイーサネットインタフェースについて

- USBインタフェースによる通信機能を使用するには、PC側に下記が必要です。
 - ・ DL/WTシリーズ用ライブラリ(TMCTL)
 - ・ PC-WT3000間のUSB接続デバイスドライバ
- イーサネットインタフェースによる通信機能を使用するには、PC側に下記が必要です。
 - ・ DL/WTシリーズ用ライブラリ(TMCTL)

上記のライブラリおよびドライバは、下記ホームページからダウンロードできます。

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

商標

- ・ Microsoft, Internet Explorer, MS-DOS, Windows, Windows NT, および Windows XPは, 米国Microsoft Corporationの, 米国およびその他の国における登録商標です。
- ・ Adobe, Acrobat, およびPostScriptは, 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, 各社の商標または登録商標です。

履歴

- ・ 2004年 12月 初版発行
- ・ 2005年 6月 2版発行
- ・ 2006年 1月 3版発行
- ・ 2006年 12月 4版発行
- ・ 2007年 3月 5版発行

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

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 RS-232Cインタフェース(オプション)

RS-232C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 USBインタフェース(オプション)

USB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 イーサネットインタフェース(オプション)

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5章 プログラムを組む前に

コマンドを送る時の書式などについて説明しています。

第6章 通信コマンド

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7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。

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

単位と注記

種類	記号	意 味
単位	k	1000 例：100kHz
	K	1024 例：640K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。

操作説明ページで使用しているシンボル

操作説明をしているページでは、説明内容を区別するために、次のようなシンボル/表示文字/用語を使用しています。

操 作

数字で示す順序で各操作をしてください。ここでは、初めて操作をすることを前提に手順を説明しています。したがって設定内容を変更する場合は、すべての操作を必要としない場合があります。

解 説

操作に関連する設定内容や限定事項について説明しています。

操作説明中の表示文字と用語

操作キーとソフトキー

操作説明のところに記載されている太字の英数字は、操作対象のパネル上の操作キーの文字や、画面に表示されるソフトキー/メニューの文字を示します。

SHIFT+操作キー

SHIFTキーを押して、SHIFTキーを点灯させてから、操作キーを押すという意味です。押した操作キー下側にある紫色文字の設定メニューが画面に表示されます。

構文の記号

主に6章の構文で使用している偽構文記号を表に示します。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、5-5～5-6ページを参照してください。

記号	意 味	例	入力例
<>	定義された値	ELEMEnt<x> <x>=1～4	->ELEMENT2
{ } 	{ }内から1つを選択 排他的論理和	MODE:{RMS MEAN DC RMEAN}?	->MODE:RMS?
[]	省略可能	NUMERIC[:NORMAl]:VALue?	->NUMERIC:VALUE?

目次

	はじめに	i
	このマニュアルの構成	iii
	このマニュアルで使用している記号	iv
第1章	GP-IBインタフェース	
	1.1 各部の名称	1-1
	1.2 GP-IBインタフェースの機能	1-2
	1.3 GP-IBインタフェースの仕様	1-3
	1.4 GP-IBケーブルの接続方法	1-4
	1.5 GP-IBコントロールの設定	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	RS-232インタフェース(オプション)	
	2.1 各部の名称	2-1
	2.2 RS-232インタフェースの機能と仕様	2-2
	2.3 ハンドシェイク方式の組み合わせ	2-3
	2.4 データフォーマットの組み合わせ	2-5
	2.5 RS-232インタフェースによる接続	2-6
	2.6 RS-232コントロールの設定	2-8
第3章	USBインタフェース(オプション)	
	3.1 各部の名称	3-1
	3.2 USBインタフェースの機能と仕様	3-2
	3.3 USBインタフェースによる接続	3-3
	3.4 USBコントロールの設定	3-4
第4章	イーサネットインタフェース(オプション)	
	4.1 各部の名称	4-1
	4.2 イーサネットインタフェースの機能と仕様	4-2
	4.3 イーサネットコントロールの設定	4-4
第5章	プログラムを組む前に	
	5.1 メッセージ	5-1
	5.2 命令	5-3
	5.3 応答	5-4
	5.4 データ	5-5
	5.5 コントローラとの同期	5-7
第6章	通信コマンド	
	6.1 コマンド一覧表	6-1
	6.2 ACQuisitionグループ	6-15
	6.3 AOUPutグループ	6-17
	6.4 CBCycleグループ	6-19
	6.5 COMMunicate グループ	6-22
	6.6 CURSorグループ	6-24
	6.7 DISPlayグループ	6-27
	6.8 FILEグループ	6-44

6.9	FLICkerグループ	6-49
6.10	HARMonicsグループ	6-53
6.11	HCOPyグループ	6-55
6.12	HOLDグループ	6-59
6.13	IMAGeグループ	6-60
6.14	INPutグループ	6-62
6.15	INTEGrateグループ	6-70
6.16	MEASureグループ	6-72
6.17	MOTorグループ	6-77
6.18	NUMericグループ	6-80
6.19	RATEグループ	6-97
6.20	STATus グループ	6-98
6.21	STOReグループ	6-100
6.22	SYSTemグループ	6-104
6.23	WAVeformグループ	6-107
6.24	共通コマンドグループ	6-109

第7章

ステータスレポート

7.1	ステータスレポートについて	7-1
7.2	ステータスパイト	7-3
7.3	標準イベントレジスタ	7-5
7.4	拡張イベントレジスタ	7-7
7.5	出力キューとエラーキュー	7-8

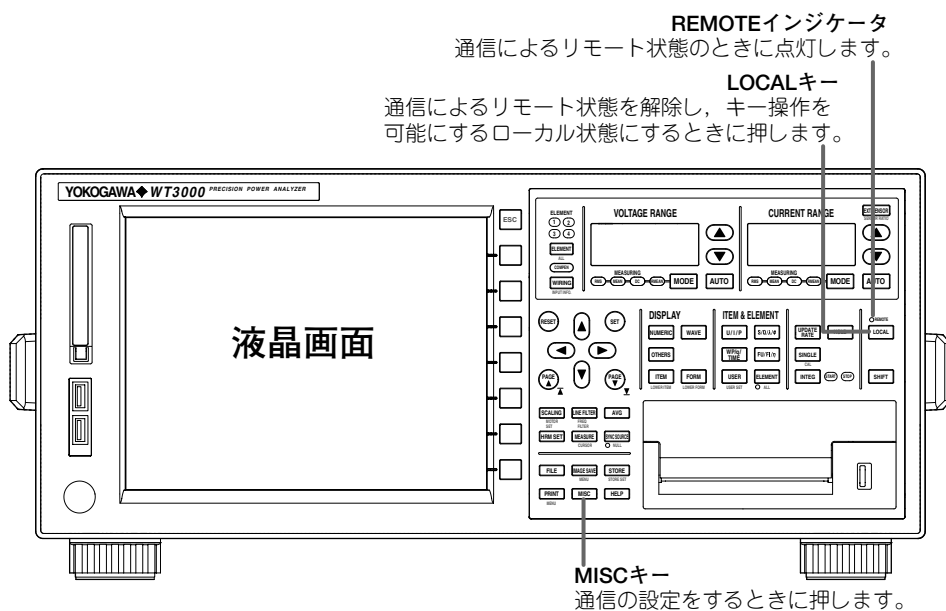
付録

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 488.2-1992について	付-5

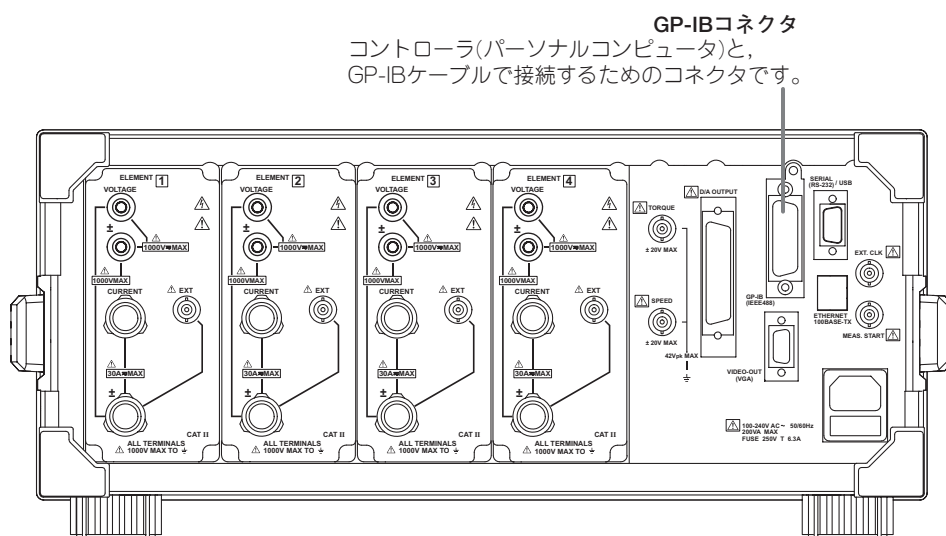
索引

1.1 各部の名称

フロントパネル



リアパネル



1.2 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。

トーカー機能

- ・ 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キー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キーを押すと、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。

Note

GP-IBインタフェースは、他の通信インタフェース(RS-232, USB, およびイーサネット)と同時に使用できません。

1.3 GP-IBインタフェースの仕様

GP-IBインタフェースの仕様

対応機器	： NATIONAL INSTRUMENTS社 ・ AT-GPIB ・ PCI-GPIBおよびPCI-GPIB+ ・ PCMCIA-GPIBおよびPCMCIA-GPIB+ドライバNI-488.2M Version 1.60以降
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 488.2-1992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	： 0～30のアドレスを設定可能。
リモート状態解除	： LOCALキーを押すことで、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、シリアルポール、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
パラレルポール	PP0	パラレルポール機能なし
デバースクリア	DC1	デバースクリアの全機能あり
デバーストリガ	DT1	デバーストリガ機能あり
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

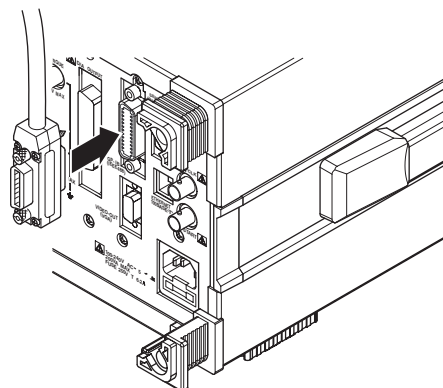
1.4 GP-IBケーブルの接続方法

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。

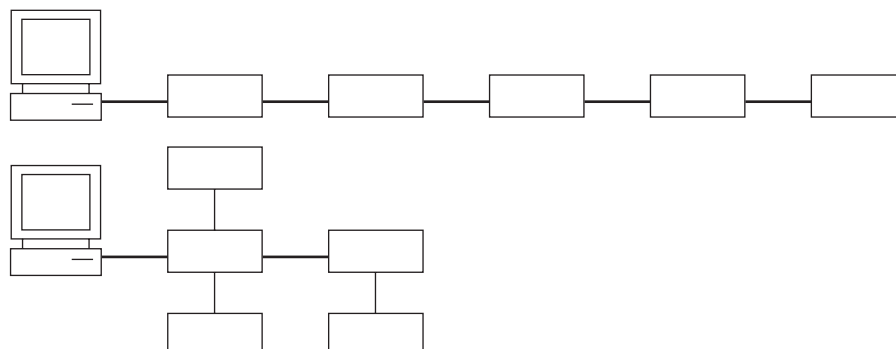
接続方法

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。
- ・ PC側のGP-IBボード(またはカード)には、NI(ナショナルインスツルメンツ)社製をご使用ください。詳細については、1.3節をご覧ください。
- ・ WT本体とPC間を接続する通信ケーブルの途中に変換器を介した場合(たとえば、GP-IBとUSB変換のように)、正常に動作しないときがあります。詳細については、お買い求め先にお問い合わせください。
- ・ 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。

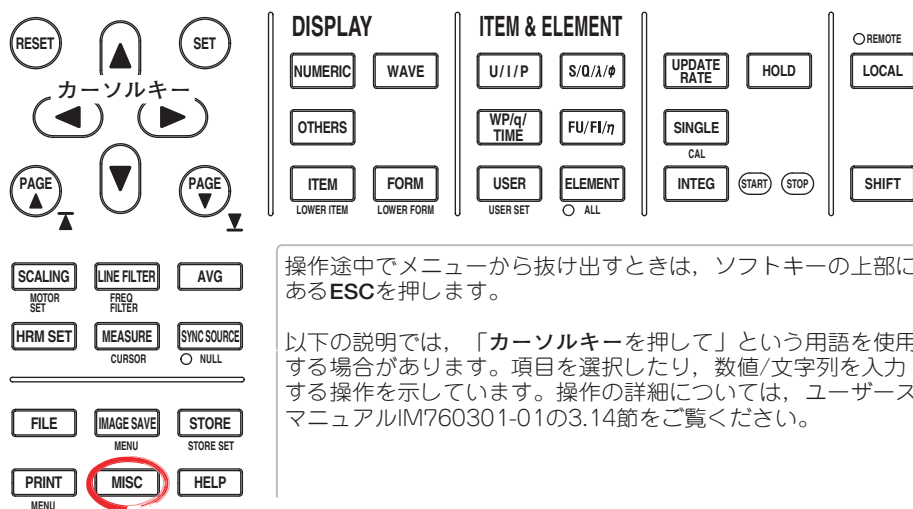


注 意

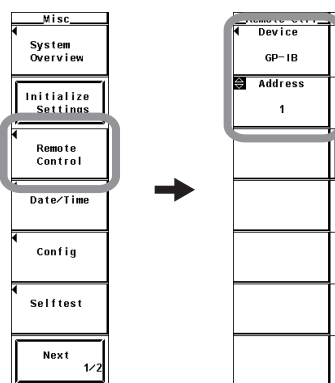
通信ケーブルの接続や取り外しのときは、必ずパーソナルコンピュータおよび本機器の電源をOFFにしてください。OFFにしないと、誤動作を生じたり、内部回路を破損することがあります。

1.5 GP-IBコントロールの設定

操 作



1. **MISC**を押します。Miscメニューが表示されます。
2. **Remote Control**のソフトキーを押します。Remote Ctrlメニューが表示されます。
3. **Device**のソフトキーを押して、GP-IBを選択します。
ここで選択した通信インタフェースだけが有効です。選択していない他の通信インタフェースでコマンドを受信しても本機器は受け付けません。
4. **カーソルキー**を押して、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

アドレスサブルモードのときの、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

コントローラや他のデバイスも含めて、GP-IB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(7-8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。

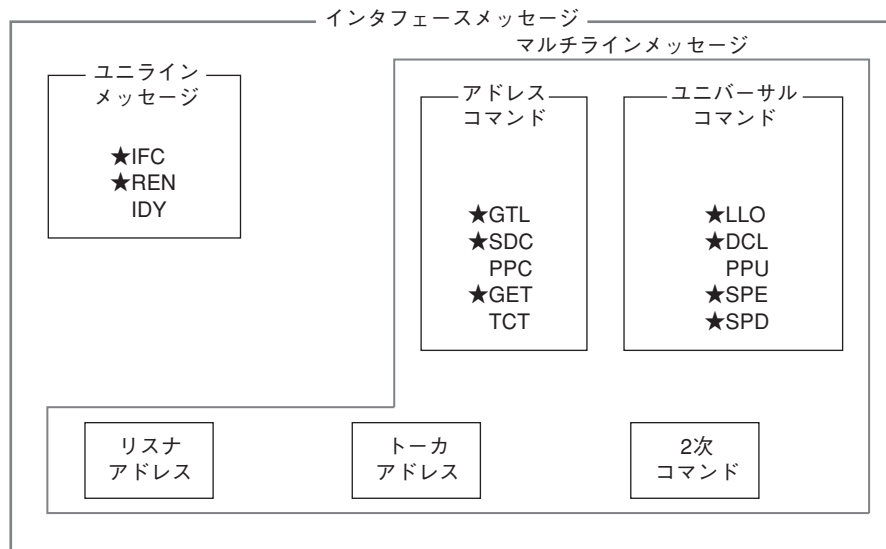
- ・ リスナに指定している機器に有効なコマンド
 - ・ GTL(Go To Local)
 - ・ SDC(Selected Device Clear)
 - ・ PPC(Parallel Poll Configure)
 - ・ GET(Group Execute Trigger)
- ・ トーカに指定している機器に有効なコマンド
 - TCT(Take Control)

- ・ **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)

そのほか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。



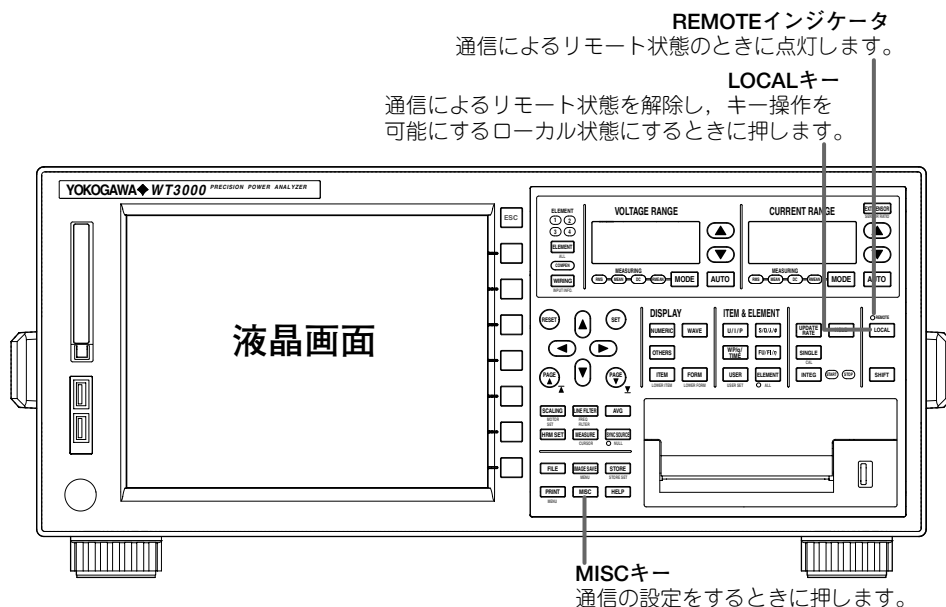
★印は本機器でサポートしているインタフェースメッセージです。

Note**SDCとDCLの違い**

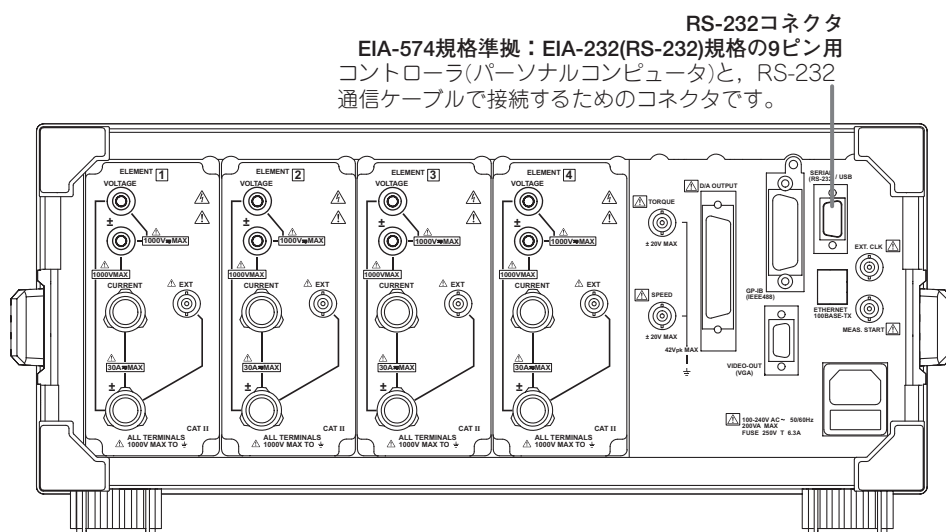
マルチラインメッセージのうち、SDCはトーカ・リスナの指定が必要なアドレスコマンド、DCLはトーカ・リスナの指定が不要なユニバーサルコマンドです。したがって、SDCはある特定の機器を対象にしますが、DCLはバス上のすべての機器を対象にします。

2.1 各部の名称

フロントパネル



リアパネル



2.2 RS-232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。

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キー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キー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

RS-232インタフェースは、他の通信インタフェース(GP-IB, USB, およびイーサネット)と同時に使用できません。

RS-232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, 38400
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ーク	: CA, CB信号について、常にTRUEが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ーク	: 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

2.3 ハンドシェーク方式の組み合わせ

RS-232C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 (〇・・・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			〇			〇
XON-XON	XON-XON	〇			〇		
XON-RS	XON-RTS	〇				〇	
CS-RS	CTS-RTS		〇			〇	

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。

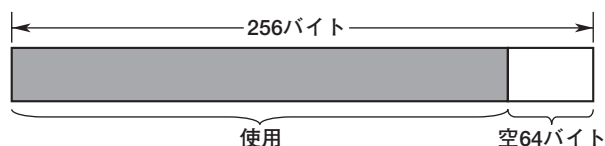
受信データ制御

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。

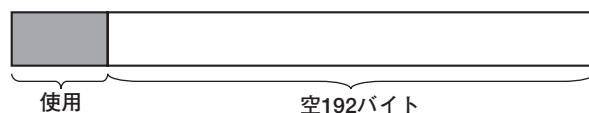
データ受信制御に関する注意

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。

ハンドシェークによるデータ受信制御



ハンドシェークを行う場合、内部へのデータの受け渡しに間に合わず、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。



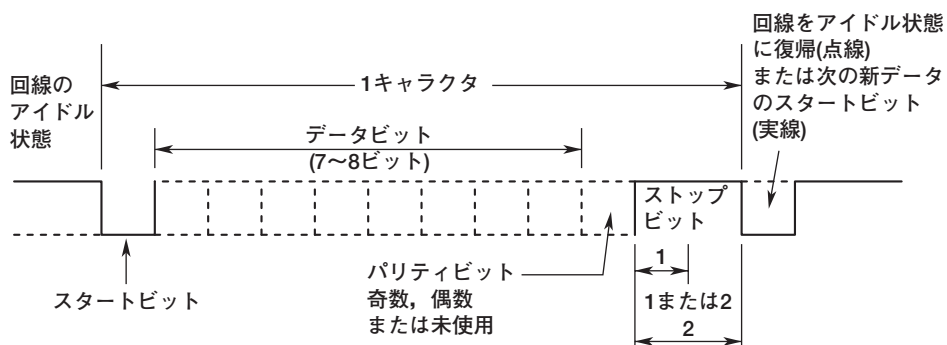
ハンドシェークに関わらず、もしバッファがFULLになったら、あふれたデータは格納せずに捨てられます。

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。

2.4 データフォーマットの組み合わせ

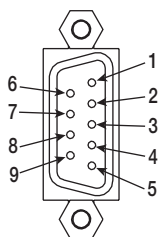
本機器のRS-232Cインタフェースは、調歩同期式で通信を行います。調歩同期式は、1キャラクタ(1文字)を転送するたびにスタートビットを付け、以降順にデータビット、パリティビット、ストップビットを付加します(下図参照)。



2.5 RS-232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

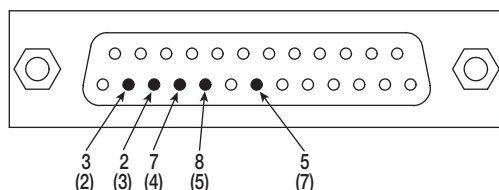


DELC-J9PAF-13L6 相当品

- | | |
|-----------------------|---|
| 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信号です。
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信号です。
信号方向.....入力 |

* 1, 4, 6, 9ピンは使用しません。

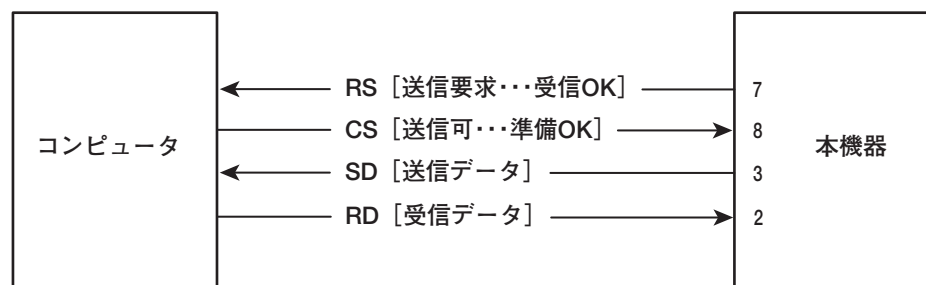
9ピン-25ピン変換コネクタと信号名



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RS-232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。



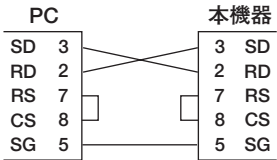
RS-232規定の信号一覧とJISおよびCCITT規定の略号

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

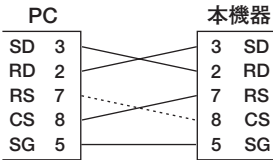
信号線の結線例

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。

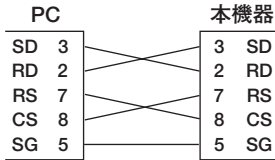
・ OFF-OFF / XON-XON



・ XON-RTS(XON-RS)

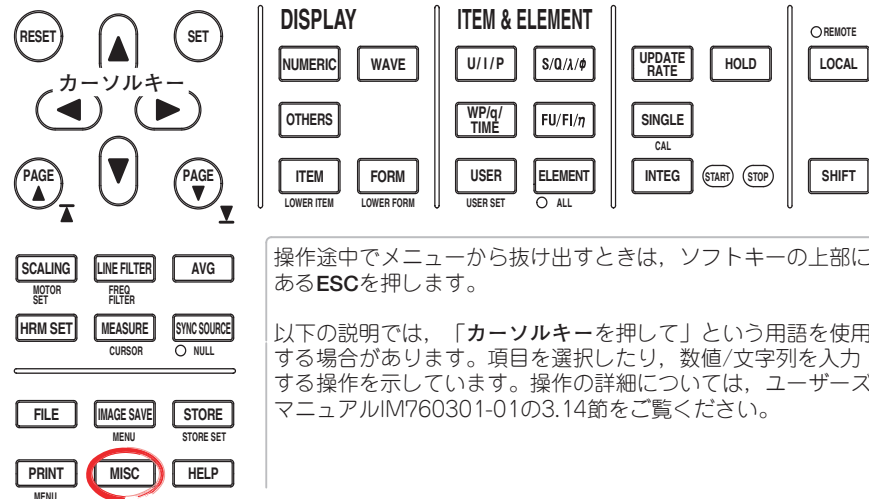


・ CTS-RTS(CS-RS)



2.6 RS-232コントロールの設定

操 作



RS-232コントロールの選択

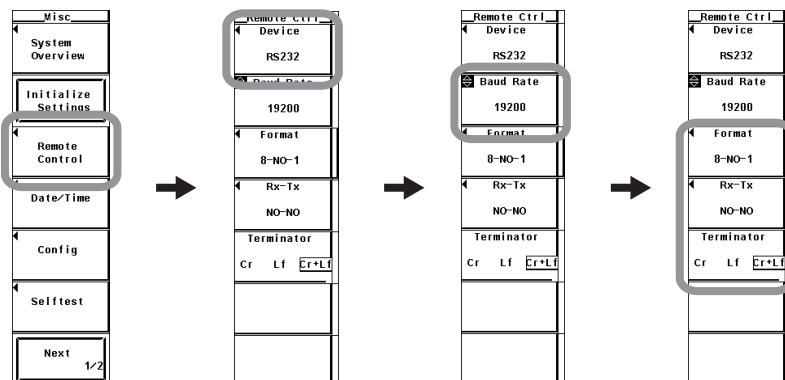
1. **MISC**を押します。Miscメニューが表示されます。
2. **Remote Control**のソフトキーを押します。Remote Ctrlメニューが表示されます。
3. **Device**のソフトキーを押して、RS232を選択します。
ここで選択した通信インタフェースだけが有効です。選択していない他の通信インタフェースでコマンドを受信しても、本機器は受け付けません。

ボーレートの選択

4. **カーソルキー**を押して、Baud Rate(ボーレート)を選択します。

データフォーマット、ハンドシェーク方式、およびターミネータの選択

4. **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

次の中から選択します。

1200, 2400, 4800, 9600, 19200, 38400

データフォーマット

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

ハンドシェイク方式

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

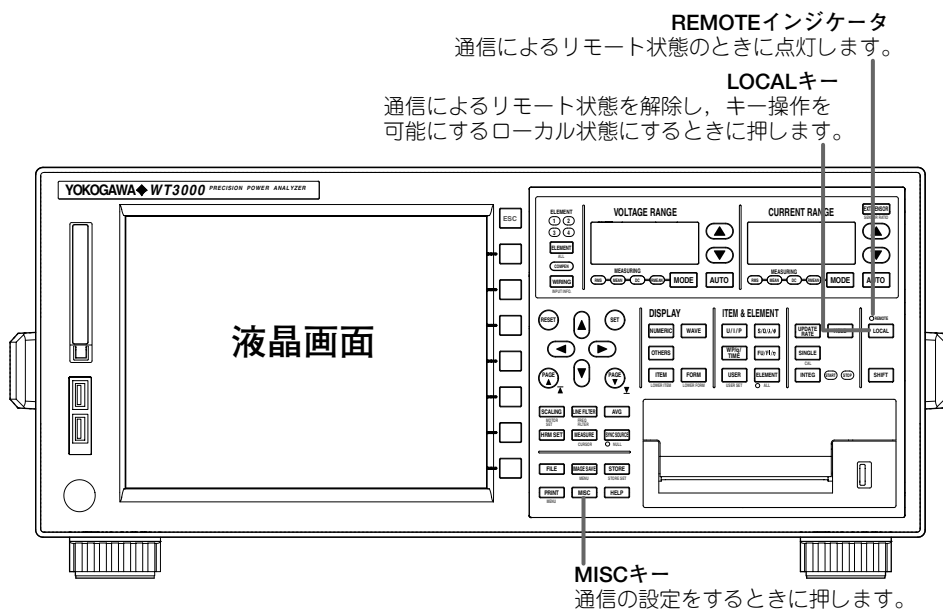
ターミネータ

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。

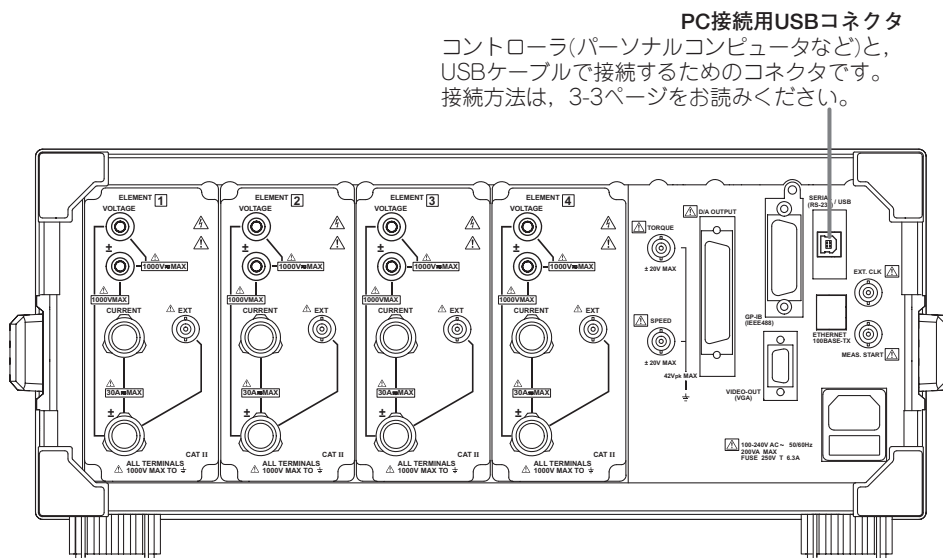
Cr, Lf, Cr+Lf

3.1 各部の名称

フロントパネル



リアパネル



3.2 USB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、設定データ、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。

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キー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キー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

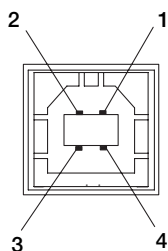
USBインタフェースは、他の通信インタフェース(GP-IB, RS-232, およびイーサネット通信インタフェース)と同時に使用できません。

USBインタフェースの仕様

電氣的・機械的仕様	: USB Rev.1.1に準拠
コネクタ	: タイプBコネクタ(レセプタクル)
ポート数	: 1
電源	: セルフパワー
対応システム環境	: Windows 98 SE, Windows Me, Windows 2000, または Windows XPで動作し、USBポートが標準装備されているパーソナルコンピュータなどのコントローラ(パーソナルコンピュータとの接続には、別途デバイスドライバが必要)

3.3 USBインタフェースによる接続

コネクタと信号名



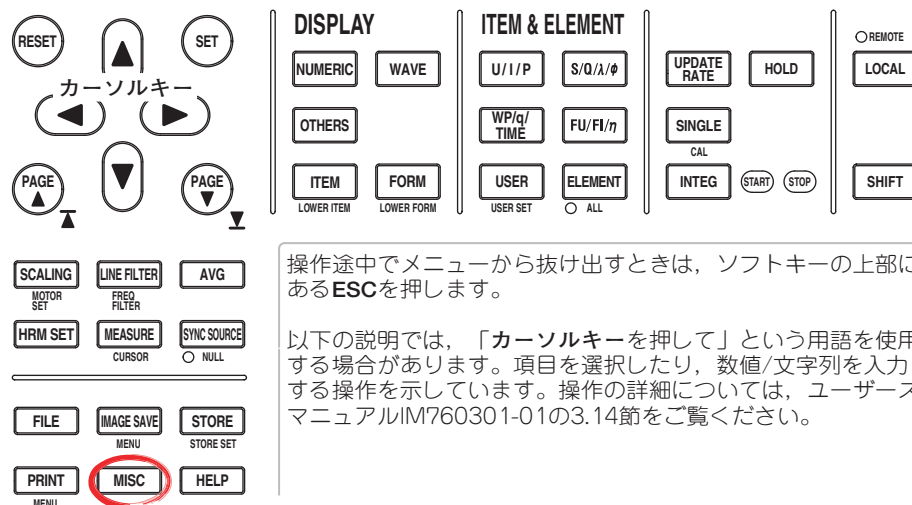
ピン番号	信号名
1	V _{BUS} : +5V
2	D ⁻ : -Data
3	D ⁺ : +Data
4	GND : グランド

接続時の注意

- ・ USBケーブルは、USBコネクタに奥までしっかりと差し込んで接続してください。
- ・ USBハブを使って複数の機器を接続する場合は、本機器をコントローラに最も近いUSBハブに接続してください。
- ・ 本機器の電源を投入してから操作が可能になるまでの間(約20～30秒)は、USBケーブルを抜き挿ししないでください。本機器を損傷する恐れがあります。

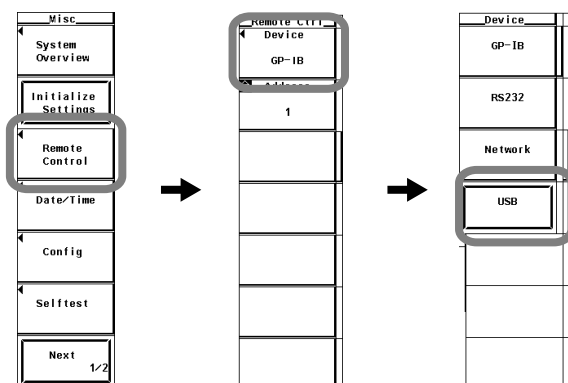
3.4 USBコントロールの設定

操 作



USBコントロールの選択

1. **MISC**を押します。Miscメニューが表示されます。
2. **Remote Control**のソフトキーを押します。Remote Ctrlメニューが表示されます。
3. **Device**のソフトキーを押して、USBを選択します。
ここで選択した通信インターフェースだけが有効です。選択していない他の通信インターフェースでコマンドを受信しても、本機器は受け付けません。
4. **カーソルキー**を押して、識別のためのID値を設定します。



解 説

USB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。本機器での設定に加えて、パーソナルコンピュータに当社で用意したUSB接続デバイスドライバと専用ライブラリソフトウェア(TMCTL)がインストールされていることが必要です。

ID値の設定

本機器のID値を次の範囲で設定します。

1～127

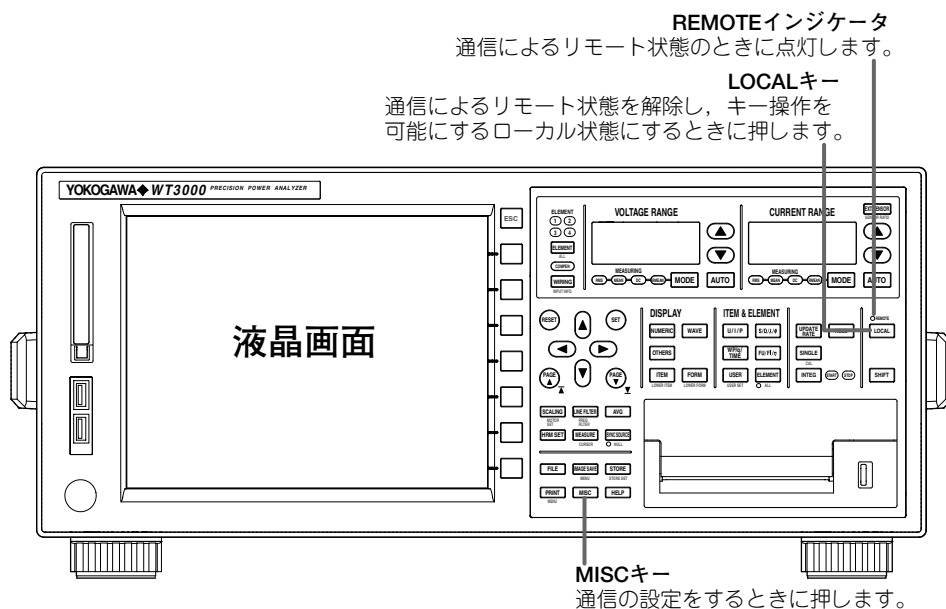
USBインタフェースでは、コントローラに対して複数の機器を接続することができます。そのため、1つのUSBシステム内において、機器が複数接続されている場合には、コントローラから各機器を識別するためにID値を使用します。したがって、1つのシステム内での機器同士のID値は、重ならないように設定する必要があります。

Note

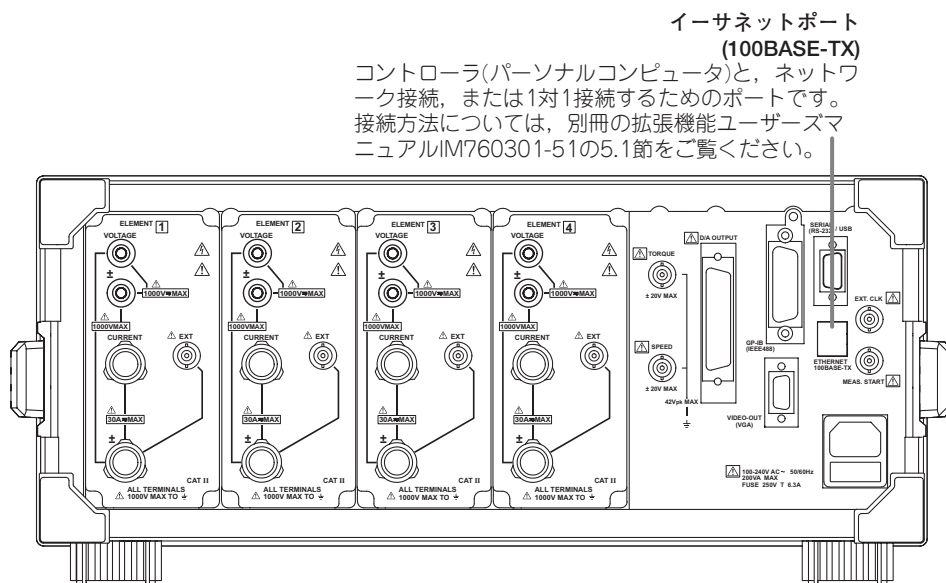
USBインタフェースで通信中は、ID値を変更しないでください。

4.1 各部の名称

フロントパネル



リアパネル



4.2 イーサネットインタフェースの機能と仕様

受信機能

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。

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キー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キー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, RS-232, およびUSB)と同時に使用できません。

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、拡張機能ユーザーズマニュアルIM760301-51の5章をご覧ください。

ユーザー認証機能

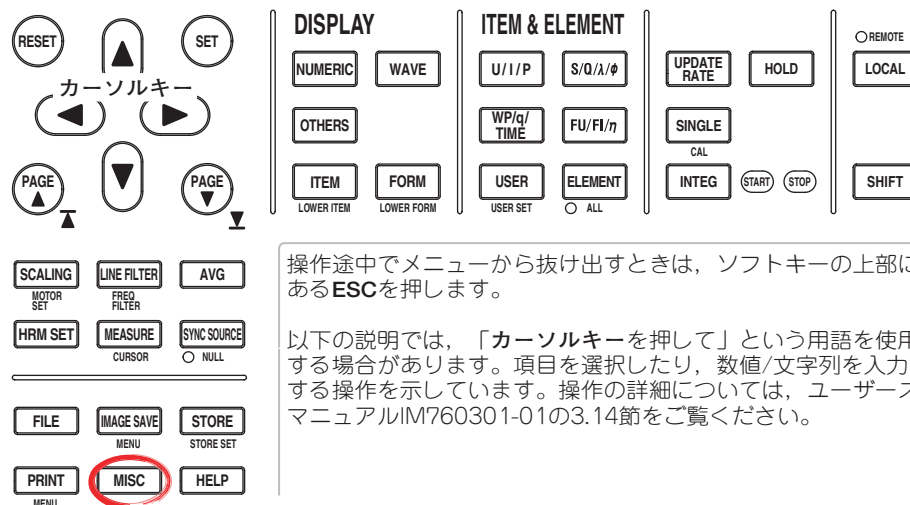
イーサネットインタフェースを使って、パーソナルコンピュータから本機器にアクセスするには、ユーザー名とパスワードを入力する必要があります。本機器のMiscメニューの中のUser Account設定画面で、本機器にアクセスするときのユーザー名とパスワードを設定してください。詳細は、次節の「イーサネットコントロールの設定」をご覧ください。

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、拡張機能ユーザーズマニュアルIM760301-51の5.1節をご覧ください。

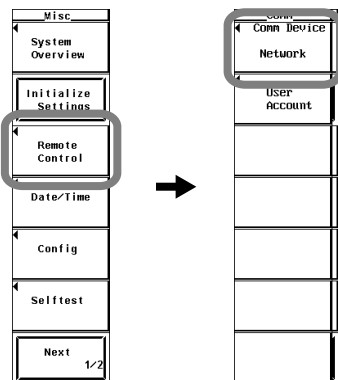
4.3 イーサネットコントロールの設定

操 作



イーサネットコントロールの選択

1. **MISC**を押します。Miscメニューが表示されます。
2. **Remote Control**のソフトキーを押します。Remote Ctrlメニューが表示されます。
3. **Device**のソフトキーを押して、Networkを選択します。
ここで選択した通信インターフェースだけが有効です。選択していない他の通信インターフェースでコマンドを受信しても、本機器は受け付けません。

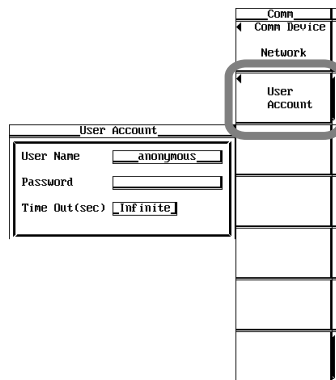


ユーザーアカウント(ユーザー名とパスワード)の設定

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。
6. **カーソルキー**を押して、User Nameを選択します。
7. **SET**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IM760301-01の「3.14 数値や文字列の入力」をご覧ください。
9. **カーソルキー**を押して、Passwordを選択します(設定は2回繰り返します)。
10. **SET**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IM760301-01の「3.14 数値や文字列の入力」をご覧ください。

タイムアウト時間の設定

12. **カーソルキー**を押して、Time Outを選択します。
13. **SET**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. **カーソルキー**を押して、タイムアウト時間を設定します。
15. **SET**または**ESC**を押して、設定ボックスを閉じます。



TCP/IPの設定

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、拡張機能ユーザーズマニュアルIM760301-51の5.2節をご覧ください。

本機器の再起動

ユーザーアカウント、タイムアウト時間、およびTCP/IPを新たに設定あるいは変更したあとは、設定内容を反映するため、本機器を再起動します。

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。本機器での設定に加えて、パーソナルコンピュータに当社で用意した専用ライブラリソフトウェア(TMCTL)がインストールされていることが必要です。

有償ソフトウェア

WTViewer(形名760121) バージョン4.01以降。
体験版を用意しております。次のURLからダウンロードできます。
<http://www.yokogawa.co.jp/tm/Bu/760121>

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。

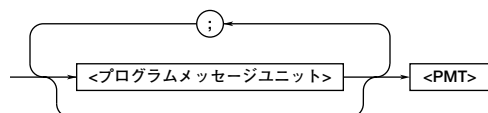
5.1 メッセージ

メッセージとクエリ

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、本機器からコントローラへの応答を要求するプログラムメッセージをクエリといいます。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クエリがあるときは、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 `:INPut:CFactor 3;INdependent OFF<PMT>`
 ユニット ユニット

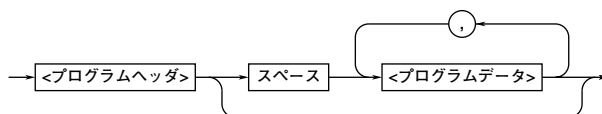
<PMT>

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、5-3ページを参照してください。

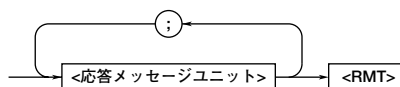
<プログラムデータ>

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、5-5ページを参照してください。

例 `:INPut:CFactor 3<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。

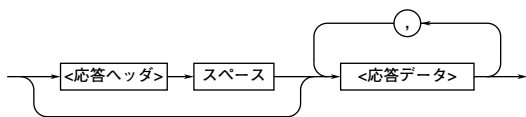
例 `:INPUT:CFactor 3;INDEPENDENT 0<RMT>`
 ユニット ユニット

<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、5-4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、5-5ページを参照してください。

例

```
100.00E-03<RMT> :DISPLAY:MODE 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とする、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。

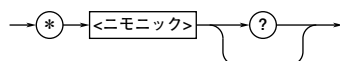
5.2 命令

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

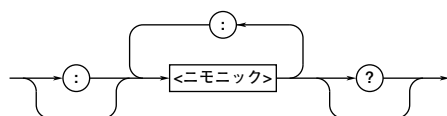
IEEE 488.2-1992で規定されている命令を共通コマンドといいます。共通コマンドのヘッダの書式は次のようになります。先頭に必ず「*」(アスタリスク)を付けます。



共通コマンドの例 *CLS

複合ヘッダ

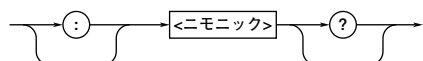
共通コマンド以外の本機器固有の命令は、機能ごとに分類されて、階層化されています。複合ヘッダの書式は次のようになります。下の階層を記述するときは、必ず「:」(コロン)を付けます。



複合ヘッダの例 :DISPlay:MODE

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 積算に関するグループ

```
:INTEGrate?
:INTEGrate:MODE
:INTEGrate:ACAL
:INTEGrate:TiMEr
:INTEGrate:RTiMe?
:INTEGrate:RTiMe:STArT
:INTEGrate:RTiMe:ENd
:INTEGrate:STArT
:INTEGrate:STOP
:INTEGrate:RESet
```

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 :INTEGrate:MODE NORMal;ACAL ON <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 :INTEGrate:MODE NORMal;;DISPlay:
MODE 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 :INTEGrate:MODE NORMal;;HOLD ON<PMT>

共通コマンドを続けて記述する場合

IEEE 488.2-1992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 :INTEGrate:MODE NORMal;*CLS;
ACAL ON<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 :INTEGrate:MODE NORMal<PMT>;
INTEGrate:ACAL ON<PMT>

上位クエリ

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 `:INTEGrate?<PMT> -> :INTEGRATE:
MODE NORMAL;ACAL 0;TIMER 0,0,0<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。

- ・ニ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?」`
- ・ニ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][:ALL] ON -> 「SCAL 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。

5.3 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:MODE?<PMT> -> :DISPLAY:
MODE 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。

5.4 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: VT比の設定 -> [:INPUT]:SCALing:VT: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:EESE #HFE)
<文字データ>	規定された文字列(ニモニック)。{}内から選択 (例: トリガモードの選択 -> :DISPlay:WAVE: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: ユーザー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:SETup[: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 -> #40012ABCDEFGHJKLM)

<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「μ」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。

5.4 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、5-4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' "IEEE488.2-1992"

- ・ 「'''」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>	#40012ABCDEFGHJKLM

- ・ #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。

5.5 コントローラとの同期

オーバ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ラップ動作と
いい、オーバラップ動作する命令を、オーバラップコ
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ラップ動
作を回避できます。

オーバ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:ELEMenT1?」の直前で
「*WAI」を実行しているので、「: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ラップ動作を許可しています。
ファイルロードはオーバ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、オーバラップ動作が終了したとき
に、標準イベントレジスタ(7-5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: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。「*OPC?」はオーバラップ動作
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。オーバラップコ
マンドは、6章でオーバ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があります。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(7-7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(7-7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。

6.1 コマンド一覧表

コマンド	機能	ページ
ACQuisitionグループ		
:ACQuisition?	波形サンプリングデータの出力に関する全設定値の問い合わせ	6-15
:ACQuisition:BYTeorder	「:ACQuisition:SEND?」で送信される波形サンプリングデータ (FLOAT形式)のバイト出力順序の設定/問い合わせ	6-15
:ACQuisition:END	「:ACQuisition:SEND?」で送信される波形サンプリングデータの出力終了点の設定/問い合わせ	6-15
:ACQuisition:FORMat	「:ACQuisition:SEND?」で送信される波形サンプリングデータのフォーマットの設定/問い合わせ	6-15
:ACQuisition:HOLD	すべての波形サンプリングデータを保持する(ON)/解除する(OFF)の設定/問い合わせ	6-15
:ACQuisition:LENGth?	「:ACQuisition:TRAcE」で指定した波形サンプリングデータの全データ点数の問い合わせ	6-16
:ACQuisition:SEND?	「:ACQuisition:TRAcE」で指定した波形サンプリングデータの問い合わせ	6-16
:ACQuisition:SRAtE?	取り込んだデータのサンプルレートの問い合わせ	6-16
:ACQuisition:STARt	「:ACQuisition:SEND?」で送信される波形サンプリングデータの出力開始点の設定/問い合わせ	6-16
:ACQuisition:TRAcE	「:ACQuisition:SEND?」の対象の設定/問い合わせ	6-16
AOUPutグループ		
:AOUPut?	D/A出力に関する全設定値の問い合わせ	6-17
:AOUPut:NORMal?	D/A出力に関する全設定値の問い合わせ	6-17
:AOUPut[:NORMal]:CHANnel<x>	D/A出力項目(ファンクション・エレメント・次数)の設定/問い合わせ	6-17
:AOUPut[:NORMal]:IRTime	積算値のD/A出力における積算定格時間の設定/問い合わせ	6-17
:AOUPut[:NORMal]:MODE<x>	D/A出力項目に対する定格値設定方式の設定/問い合わせ	6-18
:AOUPut[:NORMal]:RATE<x>	D/A出力項目に対する定格最大・最小値のマニュアル設定/問い合わせ	6-18
CBCycle グループ		
:CBCycle?	Cycle by Cycle測定機能に関する全設定値の問い合わせ	6-19
:CBCycle:COUnT	Cycle by Cycle測定の測定サイクル数の設定/問い合わせ	6-19
:CBCycle:DISPlay?	Cycle by Cycle表示に関する全設定値の問い合わせ	6-19
:CBCycle:DISPlay:CURSor	Cycle by Cycle表示のカーソル位置の設定/問い合わせ	6-19
:CBCycle:DISPlay:ITeM<x>	Cycle by Cycle表示の表示項目(ファンクション・エレメント)の設定/問い合わせ	6-19
:CBCycle:DISPlay:PAGE	Cycle by Cycle表示の表示ページ番号の設定/問い合わせ	6-19
:CBCycle:FILTer?	Cycle by Cycle測定のフィルタに関する全設定値の問い合わせ	6-19
:CBCycle:FILTer:LINE?	Cycle by Cycle測定のラインフィルタに関する全設定値の問い合わせ	6-20
:CBCycle:FILTer[:LINE][:ALL]	Cycle by Cycle測定の全エレメントのラインフィルタの一括設定	6-20
:CBCycle:FILTer[:LINE]:ELEMeNt<x>	Cycle by Cycle測定の各エレメントのラインフィルタの設定/問い合わせ	6-20
:CBCycle:FILTer[:LINE]:MOTor	Cycle by Cycle測定のモータ入力ラインフィルタの設定/問い合わせ	6-20
:CBCycle:RESet	Cycle by Cycle測定のリセット	6-20
:CBCycle:STARt	Cycle by Cycle測定の開始	6-20
:CBCycle:STATe?	Cycle by Cycle測定ステータスの問い合わせ	6-20
:CBCycle:SYNChronize?	Cycle by Cycle測定の同期ソースに関する全設定値の問い合わせ	6-20
:CBCycle:SYNChronize:SLOPe	Cycle by Cycle測定の同期ソースのスロープの設定/問い合わせ	6-20
:CBCycle:SYNChronize:SOURce	Cycle by Cycle測定の同期ソースの設定/問い合わせ	6-20
:CBCycle:TIMEout	Cycle by Cycle測定のタイムアウト時間の設定/問い合わせ	6-20
:CBCycle:TRIGger?	トリガに関する全設定値の問い合わせ	6-21
:CBCycle:TRIGger:LEVeL	トリガレベルの設定/問い合わせ	6-21
:CBCycle:TRIGger:MODE	トリガモードの設定/問い合わせ	6-21
:CBCycle:TRIGger:SLOPe	トリガスロープの設定/問い合わせ	6-21
:CBCycle:TRIGger:SOURce	トリガソースの設定/問い合わせ	6-21

6.1 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	6-22
: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	6-22
:COMMunicate:LOCKout	ローカルロックアウトの設定/解除	6-22
: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	6-22
: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	6-22
: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	6-22
:COMMunicate:REMOte	リモート/ローカルの設定	6-22
:COMMunicate:STATus?	回線固有のステータスの問い合わせ	6-23
:COMMunicate:VERBoSe	クエリに対する応答を、フルスペルか、省略形かの設定/問い合わせ	6-23
:COMMunicate:WAIT	指定された拡張イベントの待機	6-23
: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	6-23
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	6-24
:CURSor:BAR?	バーグラフ表示のカーソル測定に関する全設定値の問い合わせ	6-24
:CURSor:BAR:POSition<x>	バーグラフ表示のカーソル位置(次数)の設定/問い合わせ	6-24
:CURSor:BAR[:STATe]	バーグラフ表示のカーソル表示のON/OFFの設定/問い合わせ	6-24
:CURSor:BAR:{Y<x> DY}?	バーグラフ表示のカーソル測定値の問い合わせ	6-24
:CURSor:FFT?	FFT波形表示のカーソル測定に関する全設定値の問い合わせ	6-24
:CURSor:FFT:POSition<x>	FFT波形表示のカーソル位置の設定/問い合わせ	6-24
:CURSor:FFT[:STATe]	FFT波形表示のカーソル表示のON/OFFの設定/問い合わせ	6-25
:CURSor:FFT:TRACe<x>	FFT波形表示のカーソルの対象の設定/問い合わせ	6-25
:CURSor:FFT:{X<x> DX Y<x> DY}?	FFT波形表示のカーソル測定値の問い合わせ	6-25
:CURSor:TREnd?	トレンド表示のカーソル測定に関する全設定値の問い合わせ	6-25
:CURSor:TREnd:POSition<x>	トレンド表示のカーソル位置の設定/問い合わせ	6-25
:CURSor:TREnd[:STATe]	トレンド表示のカーソル表示のON/OFFの設定/問い合わせ	6-25
:CURSor:TREnd:TRACe<x>	トレンド表示のカーソルの対象の設定/問い合わせ	6-25
:CURSor:TREnd:{X<x> Y<x> DY}?	トレンド表示のカーソル測定値の問い合わせ	6-25
:CURSor:WAVE?	波形表示のカーソル測定に関する全設定値の問い合わせ	6-26
:CURSor:WAVE:PATH	波形表示のカーソルパスの設定/問い合わせ	6-26
:CURSor:WAVE:POSition<x>	波形表示のカーソル位置の設定/問い合わせ	6-26
:CURSor:WAVE[:STATe]	波形表示のカーソル表示のON/OFFの設定/問い合わせ	6-26
:CURSor:WAVE:TRACe<x>	波形表示のカーソルの対象の設定/問い合わせ	6-26
:CURSor:WAVE:{X<x> DX PERDt Y<x> DY}?	波形表示のカーソル測定値の問い合わせ	6-26
DISPlayグループ		
:DISPlay?	画面表示に関する全設定値の問い合わせ	6-27
:DISPlay:BAR?	バーグラフに関する全設定値の問い合わせ	6-27
:DISPlay:BAR:FORMat	バーグラフの表示フォーマットの設定/問い合わせ	6-27
:DISPlay:BAR:ITEM<x>	バーグラフ項目(ファンクション・エレメント)の設定/問い合わせ	6-27
:DISPlay:BAR:ORDer	バーグラフの開始/終了次数の設定/問い合わせ	6-27
:DISPlay:CBCycle?	Cycle by Cycle表示に関する全設定値の問い合わせ	6-28
:DISPlay:CBCycle:CURSor	Cycle by Cycle表示のカーソル位置の設定/問い合わせ	6-28
:DISPlay:CBCycle:ITEM<x>	Cycle by Cycle表示の表示項目(ファンクション・エレメント)の設定/問い合わせ	6-28
:DISPlay:CBCycle:PAGE	Cycle by Cycle表示の表示ページ番号の設定/問い合わせ	6-28
:DISPlay:FFT?	FFT波形表示に関する全設定値の問い合わせ	6-28
:DISPlay:FFT:FFT<x>?	FFT波形に関する全設定値の問い合わせ	6-28
:DISPlay:FFT:FFT<x>:LABel	FFT波形のラベル名の設定/問い合わせ	6-28
:DISPlay:FFT:FFT<x>:OBJect	FFT演算の対象波形の設定/問い合わせ	6-29
:DISPlay:FFT:FFT<x>[:STATe]	FFT波形表示のON/OFFの設定/問い合わせ	6-29

コマンド	機能	ページ
:DISPlay:FFT:FORMat	FFT波形の表示フォーマットの設定/問い合わせ	6-29
:DISPlay:FFT:POINt	FFT演算の点数の設定/問い合わせ	6-29
:DISPlay:FFT:SCOPE	FFT波形の表示範囲の設定/問い合わせ	6-29
:DISPlay:FFT:SPECTrum	FFT波形の表示スペクトル形式の設定/問い合わせ	6-29
:DISPlay:FFT:VSCale	FFT波形の垂直軸(Vertical)の表示スケールの設定/問い合わせ	6-29
:DISPlay:FFT:WINDow	FFT演算の窓関数の設定/問い合わせ	6-29
:DISPlay:FLICKer?	フリッカ測定表示に関する全設定値の問い合わせ	6-29
:DISPlay:FLICKer:ELEMent	フリッカ測定表示の表示対象エレメントの設定/問い合わせ	6-30
:DISPlay:FLICKer:PAGE	フリッカ測定表示の表示ページ番号の設定/問い合わせ	6-30
:DISPlay:FLICKer:PERiod	フリッカ測定表示の表示観測期間番号の設定/問い合わせ	6-30
:DISPlay:INFORMation?	設定情報の一覧表示に関する全設定値の問い合わせ	6-30
:DISPlay:INFORMation:PAGE	設定情報の一覧表示における表示ページ番号の設定/問い合わせ	6-30
:DISPlay:INFORMation[:STATe]	設定情報の一覧表示のON/OFFの設定/問い合わせ	6-30
:DISPlay:MATH?	演算波形表示に関する全設定値の問い合わせ	6-30
:DISPlay:MATH:CONSTant<x>	波形演算式で使用する定数の設定/問い合わせ	6-30
:DISPlay:MATH:MATH<x>?	演算波形に関する全設定値の問い合わせ	6-31
:DISPlay:MATH:MATH<x>:EXPReSSion	波形演算の演算式の設定/問い合わせ	6-31
:DISPlay:MATH:MATH<x>:LABel	演算波形のラベル名の設定/問い合わせ	6-31
:DISPlay:MATH:MATH<x>:SCALing?	演算波形のスケールリングに関する全設定値の問い合わせ	6-31
:DISPlay:MATH:MATH<x>:SCALing:CENter	演算波形のマニュアルスケールリングの中心値の設定/問い合わせ	6-31
:DISPlay:MATH:MATH<x>:SCALing:MODE	演算波形のスケールリング方式の設定/問い合わせ	6-31
:DISPlay:MATH:MATH<x>:SCALing:SDIV	演算波形のマニュアルスケールリングのScale/div値の設定/問い合わせ	6-31
:DISPlay:MATH:MATH<x>:UNIT	波形演算結果に付加する単位の設定/問い合わせ	6-32
:DISPlay:MODE	表示方式の設定/問い合わせ	6-32
:DISPlay:NUMeric?	数値表示に関する全設定値の問い合わせ (:DISPlay:NUMeric:NORMAL?と同じ)	6-32
:DISPlay:NUMeric:NORMAL?	数値表示に関する全設定値の問い合わせ	6-32
:DISPlay:NUMeric[:NORMAL]:ALL?	数値表示(全表示)に関する全設定値の問い合わせ	6-32
:DISPlay:NUMeric[:NORMAL]:ALL:CURSor	数値表示(全表示)のカーソル位置の設定/問い合わせ	6-32
:DISPlay:NUMeric[:NORMAL]:ALL:ORder	数値表示(全表示)の高調波測定ファンクション表示ページにおける 表示次数の設定/問い合わせ	6-33
:DISPlay:NUMeric[:NORMAL]:ALL:PAGE	数値表示(全表示)の表示ページ番号の設定/問い合わせ	6-33
:DISPlay:NUMeric[:NORMAL]:FORMat	数値表示の方式の設定/問い合わせ	6-33
:DISPlay:NUMeric[:NORMAL]:LIST?	数値表示(リスト表示)に関する全設定値の問い合わせ	6-33
:DISPlay:NUMeric[:NORMAL]:LIST:CURSor	数値表示(リスト表示)のカーソル位置の設定/問い合わせ	6-33
:DISPlay:NUMeric[:NORMAL]:LIST:HEADer	数値表示(リスト表示)のヘッダ部のカーソル位置の設定/問い合わせ	6-34
:DISPlay:NUMeric[:NORMAL]:LIST:ITEM<x>	数値表示(リスト表示)の表示項目(ファンクション・エレメント)の 設定/問い合わせ	6-34
:DISPlay:NUMeric[:NORMAL]:LIST:ORder	数値表示(リスト表示)のデータ部の次数カーソル位置の設定/問い合わせ	6-34
:DISPlay:NUMeric[:NORMAL]:{VAL4 VAL8 VAL16}?	数値表示({4値 8値 16値}表示)に関する全設定値の問い合わせ	6-34
:DISPlay:NUMeric[:NORMAL]:{VAL4 VAL8 VAL16}:CURSor	数値表示({4値 8値 16値}表示)のカーソル位置の設定/問い合わせ	6-35
:DISPlay:NUMeric[:NORMAL]:{VAL4 VAL8 VAL16}:ITEM<x>	数値表示({4値 8値 16値}表示)の表示項目 (ファンクション・エレメント・次数)の設定/問い合わせ	6-35

6.1 コマンド一覧表

コマンド	機能	ページ
:DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}:PAGE	数値表示({4値 8値 16値}表示)の表示ページ番号の設定/問い合わせ	6-35
:DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}:PRESet	数値表示({4値 8値 16値}表示)の表示項目の表示順パターンのプリセット	6-35
:DISPlay:TREND?	トレンドに関する全設定値の問い合わせ	6-35
:DISPlay:TREND:ALL	全トレンドのON/OFFの一括設定	6-36
:DISPlay:TREND:CLEar	トレンドのクリアの実行	6-36
:DISPlay:TREND:FORMat	トレンドの表示フォーマットの設定/問い合わせ	6-36
:DISPlay:TREND:ITEM<x>?	各トレンドに関する全設定値の問い合わせ	6-36
:DISPlay:TREND:ITEM<x>[:FUNCTION]	トレンド項目(ファンクション・エレメント・次数)の設定/問い合わせ	6-36
:DISPlay:TREND:ITEM<x>:SCALing?	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	6-36
:DISPlay:TREND:ITEM<x>:SCALing:MODE	トレンドのスケーリング方式の設定/問い合わせ	6-36
:DISPlay:TREND:ITEM<x>:SCALing:VALue	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	6-37
:DISPlay:TREND:TDiv	トレンドの横軸(T/div)の設定/問い合わせ	6-37
:DISPlay:TREND:T<x>	トレンドのON/OFFの設定/問い合わせ	6-37
:DISPlay:VECTor?	ベクトル表示に関する全設定値の問い合わせ	6-37
:DISPlay:VECTor:NUMeric	ベクトル表示の数値データ表示のON/OFFの設定/問い合わせ	6-37
:DISPlay:VECTor:OBject	ベクトル表示の表示対象とする結線ユニットの設定/問い合わせ	6-37
:DISPlay:VECTor:{UMAG IMAG}	ベクトル表示のズーム率の設定/問い合わせ	6-37
:DISPlay:WAVE?	波形表示に関する全設定値の問い合わせ	6-37
:DISPlay:WAVE:ALL	全波形表示のON/OFFの一括設定	6-38
:DISPlay:WAVE:FORMat	波形の表示フォーマットの設定/問い合わせ	6-38
:DISPlay:WAVE:GRATicule	グラティクル(目盛り)のタイプの設定/問い合わせ	6-38
:DISPlay:WAVE:INTerpolate	波形の補間方式の設定/問い合わせ	6-38
: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	6-38
: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	6-38
:DISPlay:WAVE:MAPPing:{U<x> I<x> SPEed TORQue MATH<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	6-38
:DISPlay:WAVE:POSition?	波形の垂直ポジション(中心位置のレベル)に関する全設定値の問い合わせ	6-38
:DISPlay:WAVE:POSition:{UALL IALL}	全エレメントの{電圧 電流}波形の垂直ポジション(中心位置のレベル)の一括設定	6-38
:DISPlay:WAVE:POSition:{U<x> I<x>}	各エレメントの{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	6-38
:DISPlay:WAVE:SVALue	スケール値表示のON/OFFの設定/問い合わせ	6-39
:DISPlay:WAVE:TDiv	波形のTime/div値の設定/問い合わせ	6-39
:DISPlay:WAVE:TLABel	波形ラベル名表示のON/OFFの設定/問い合わせ	6-39
:DISPlay:WAVE:TRIGger?	トリガに関する全設定値の問い合わせ	6-39
:DISPlay:WAVE:TRIGger:LEVel	トリガレベルの設定/問い合わせ	6-39
:DISPlay:WAVE:TRIGger:MODE	トリガモードの設定/問い合わせ	6-39
:DISPlay:WAVE:TRIGger:SLOPe	トリガスロープの設定/問い合わせ	6-39
:DISPlay:WAVE:TRIGger:SOURce	トリガソースの設定/問い合わせ	6-39
:DISPlay:WAVE:{U<x> I<x> SPEed TORQue MATH<x>}	波形表示のON/OFFの設定/問い合わせ	6-39
:DISPlay:WAVE: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	6-39
:DISPlay:WAVE:VZoom:{UALL IALL}	全エレメントの{電圧 電流}波形の垂直方向のズーム率の一括設定	6-40
:DISPlay:WAVE:VZoom:{U<x> I<x>}	各エレメントの{電圧 電流}波形の垂直方向のズーム率の設定/問い合わせ	6-40

コマンド	機能	ページ
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	6-44
:FILE:CDIRectory	カレントディレクトリの変更	6-44
:FILE:DELeTe:IMAGe:{TIFF BMP PSCript PNG JPEG}	画面イメージデータファイルの削除	6-44
:FILE:DELeTe:NUMeric:{ASCIi FLOat}	数値データファイルの削除	6-44
:FILE:DELeTe:SETup	設定情報ファイルの削除	6-44
:FILE:DELeTe:WAVE:{BINary ASCIi FLOat}	波形表示データファイルの削除	6-44
:FILE:DRIVE	対象ドライブの設定	6-44
:FILE:FORMat:EXECute	PCカードの初期化の実行	6-44
:FILE:FREE?	対象ドライブの空き容量の問い合わせ	6-44
:FILE:LOAD:ABORT	ファイルの読み込みの中止	6-44
:FILE:LOAD:SETup	設定情報ファイルの読み込みの実行	6-44
:FILE:MDIRectory	ディレクトリの作成	6-45
:FILE:PATH?	カレントディレクトリの絶対パスの問い合わせ	6-45
:FILE:SAVE?	ファイルの保存に関する全設定値の問い合わせ	6-45
:FILE:SAVE:ABORT	ファイルの保存の中止	6-45
:FILE:SAVE:ACQuisition?	波形サンプリングデータのファイル保存に関する全設定値の問い合わせ	6-45
:FILE:SAVE:ACQuisition[:EXECute]	波形サンプリングデータをファイルに保存	6-45
:FILE:SAVE:ACQuisition:TRACe	ファイル保存の対象とする波形サンプリングデータの設定/問い合わせ	6-45
:FILE:SAVE:ACQuisition:TYPE	保存する波形サンプリングデータの形式の設定/問い合わせ	6-45
: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	6-45
: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	6-46
:FILE:SAVE:NUMeric?	数値データのファイル保存に関するすべての設定値の問い合わせ	6-46
:FILE:SAVE:NUMeric:CBCCycle?	Cycle by Cycle測定の数値データのファイル保存項目に関するすべての設定値の問い合わせ	6-46
:FILE:SAVE:NUMeric:CBCCycle:ALL	Cycle by Cycle測定の数値データをファイルに保存するときのすべてのエレメント・ファンクションの出力ON/OFFの一括設定	6-46
:FILE:SAVE:NUMeric:CBCCycle:{ELEMeNt<x> SIGMA SIGMB}	Cycle by Cycle測定の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 }の出力ON/OFFの設定/問い合わせ	6-46
:FILE:SAVE:NUMeric:CBCCycle:<Function>	Cycle by Cycle測定の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	6-46
:FILE:SAVE:NUMeric[:EXECute]	数値データをファイルに保存	6-47
:FILE:SAVE:NUMeric:NORMal?	数値データのファイル保存項目に関するすべての設定値の問い合わせ	6-47
:FILE:SAVE:NUMeric:NORMal:ALL	数値データをファイルに保存するときのすべてのエレメント・ファンクションの出力ON/OFFの一括設定	6-47
:FILE:SAVE:NUMeric:NORMal:{ELEMeNt<x> SIGMA SIGMB}	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 }の出力ON/OFFの設定/問い合わせ	6-47
:FILE:SAVE:NUMeric:NORMal:PRESet<x>	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を決められたパターンにプリセット	6-47
:FILE:SAVE:NUMeric:NORMal:<Function>	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	6-47
:FILE:SAVE:NUMeric:TYPE	保存する数値データの形式の設定/問い合わせ	6-48
:FILE:SAVE:SETup[:EXECute]	設定情報のファイル保存の実行	6-48
:FILE:SAVE:WAVE?	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の問い合わせ	6-48
:FILE:SAVE:WAVE[:EXECute]	波形表示データをファイルに保存	6-48
: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	6-48
: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	6-48

6.1 コマンド一覧表

コマンド	機能	ページ
FLICKer グループ		
:FLICKer?	フリッカ測定に関する全設定値の問い合わせ	6-49
:FLICKer:COUNT	短期間フリッカ値Pstの測定回数の設定/問い合わせ	6-49
:FLICKer:DC?	相対定常電圧変化dcに関する全設定値の問い合わせ	6-49
:FLICKer:DC:LIMit	相対定常電圧変化dcの限度値の設定/問い合わせ	6-49
:FLICKer:DC[:STATe]	相対定常電圧変化dcの判定ON/OFFの設定/問い合わせ	6-49
:FLICKer:DISPlay?	フリッカ測定表示に関する全設定値の問い合わせ	6-49
:FLICKer:DISPlay:ELEMent	フリッカ測定表示の表示対象エレメントの設定/問い合わせ	6-49
:FLICKer:DISPlay:PAGE	フリッカ測定表示の表示ページ番号の設定/問い合わせ	6-49
:FLICKer:DISPlay:PERiod	フリッカ測定表示の表示観測期間番号の設定/問い合わせ	6-50
:FLICKer:DMAX?	最大相対電圧変化dmaxに関する全設定値の問い合わせ	6-50
:FLICKer:DMAX:LIMit	最大相対電圧変化dmaxの限度値の設定/問い合わせ	6-50
:FLICKer:DMAX[:STATe]	最大相対電圧変化dmaxの判定ON/OFFの設定/問い合わせ	6-50
:FLICKer:DMIN?	定常範囲dminに関する全設定値の問い合わせ	6-50
:FLICKer:DMIN:LIMit	定常範囲dminの限度値の設定/問い合わせ	6-50
:FLICKer:DT?	相対電圧変化時間d(t)に関する全設定値の問い合わせ	6-50
:FLICKer:DT:LIMit	相対電圧変化時間d(t)の限度値の設定/問い合わせ	6-50
:FLICKer:DT[:STATe]	相対電圧変化時間d(t)の判定ON/OFFの設定/問い合わせ	6-50
:FLICKer:ELEMent<x>	フリッカ測定対象エレメントの設定/問い合わせ	6-50
:FLICKer:FREQuency	フリッカ測定対象周波数の設定/問い合わせ	6-50
:FLICKer:INITialize	フリッカ測定の初期化の実行	6-50
:FLICKer:INTerval	短期間フリッカ値Pstの1回の測定時間の設定/問い合わせ	6-51
:FLICKer:JUDGE	手動スイッチングdmax測定の完了と判定の実行	6-51
:FLICKer:MEASurement	フリッカ測定方法の設定/問い合わせ	6-51
:FLICKer:MOVE	手動スイッチングdmax測定の観測期間番号の移動	6-51
:FLICKer:PLT?	長期間フリッカ値Pltに関する全設定値の問い合わせ	6-51
:FLICKer:PLT:LIMit	長期間フリッカ値Pltの限度値の設定/問い合わせ	6-51
:FLICKer:PLT:NVALue	長期間フリッカ値Pltの算出演算式の定数Nの設定/問い合わせ	6-51
:FLICKer:PLT[:STATe]	長期間フリッカ値Pltの判定ON/OFFの設定/問い合わせ	6-51
:FLICKer:PST?	短期間フリッカ値Pstに関する全設定値の問い合わせ	6-51
:FLICKer:PST:LIMit	短期間フリッカ値Pstの限度値の設定/問い合わせ	6-51
:FLICKer:PST[:STATe]	短期間フリッカ値Pstの判定ON/OFFの設定/問い合わせ	6-52
:FLICKer:RESet	フリッカの測定データのリセット	6-52
:FLICKer:STARt	フリッカ測定の開始	6-52
:FLICKer:STATe?	フリッカ測定ステータスの問い合わせ	6-52
:FLICKer:UN?	定格電圧Unに関する全設定値の問い合わせ	6-52
:FLICKer:UN:MODE	定格電圧Unの取得方式の設定/問い合わせ	6-52
:FLICKer:UN:VALue	定格電圧Unの既定値の設定/問い合わせ	6-52
HARMonicsグループ		
:HARMonics?	高調波測定に関する全設定値の問い合わせ	6-53
:HARMonics:FBAND	高調波測定の周波数帯域の設定/問い合わせ	6-53
:HARMonics:IEC?	IEC高調波測定に関する全設定値の問い合わせ	6-53
:HARMonics:IEC:OBJect	IEC高調波測定対象の設定/問い合わせ	6-53
:HARMonics:IEC:{UGRouping IGRouping}	IEC高調波測定の{電圧 電流}グルーピングの設定/問い合わせ	6-53
:HARMonics:ORDer	最小/最大測定次数の設定/問い合わせ	6-53
:HARMonics:PLLSource	PLLソースの設定/問い合わせ	6-53
:HARMonics:PLLWarning?	PLLソースの警告メッセージに関する全設定値の問い合わせ	6-54
:HARMonics:PLLWarning[:STATe]	PLLソース未入力時の警告メッセージを出す/出さないの設定/問い合わせ	6-54
:HARMonics:THD	THD(高調波ひずみ率)の算出式の設定/問い合わせ	6-54

コマンド	機能	ページ
HCOPYグループ		
:HCOPY?	印刷に関する全設定値の問い合わせ	6-55
:HCOPY:ABORt	印刷または紙送りの中止	6-55
:HCOPY:AUTO?	オートプリントに関するすべての設定値の問い合わせ	6-55
:HCOPY:AUTO:INTERval	オートプリントの印字インターバルの設定/問い合わせ	6-55
:HCOPY:AUTO:{START END}	オートプリントの{スタート ストップ}予約時刻の設定/問い合わせ	6-55
:HCOPY:AUTO[:STATE]	オートプリントのON/OFFの設定/問い合わせ	6-55
:HCOPY:AUTO:SYNChronize	オートプリントの同期方式の設定/問い合わせ	6-55
:HCOPY:COMMeNt	画面下部に表示するコメントの設定/問い合わせ	6-56
:HCOPY:DIRection	プリンタの設定/問い合わせ	6-56
:HCOPY:EXECute	印刷の実行	6-56
:HCOPY:NETPrint?	ネットワークプリンタでの印刷に関するすべての設定値の問い合わせ	6-56
:HCOPY:NETPrint:COLor	ネットワークプリンタで印刷するときのカラー出力のON/OFFの設定/問い合わせ	6-56
:HCOPY:NETPrint:FORMat	ネットワークプリンタで印刷するときのプリンタ記述言語の設定/問い合わせ	6-56
:HCOPY:PRINter?	内蔵プリンタでの印刷に関する全設定値の問い合わせ	6-56
:HCOPY:PRINter:FEED	内蔵プリンタの紙送りの実行	6-56
:HCOPY:PRINter:FORMat	内蔵プリンタで印刷する内容の設定/問い合わせ	6-56
:HCOPY:PRINter:LIST?	内蔵プリンタでの数値データリストの印字に関するすべての設定値の問い合わせ	6-57
:HCOPY:PRINter:LIST:INFORMatIon	内蔵プリンタで数値データリストを印字するときに、設定情報を付加する/しないの設定/問い合わせ	6-57
:HCOPY:PRINter:LIST:NORMal?	内蔵プリンタでの数値データリストの印字項目に関するすべての設定値の問い合わせ	6-57
:HCOPY:PRINter:LIST[:NORMal]:ALL	内蔵プリンタで数値データリストを印字するときのすべてのエレメント・ファンクションの出力ON/OFFの一括設定	6-57
:HCOPY:PRINter:LIST[:NORMal]:{ELEMeNt<x> SIGMA SIGMB}	内蔵プリンタで数値データリストを印字するときの{各エレメント ΣA ΣB}の出力ON/OFFの設定/問い合わせ	6-57
:HCOPY:PRINter:LIST[:NORMal]:PRESet<x>	内蔵プリンタで数値データリストを印字するときのエレメント・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	6-57
:HCOPY:PRINter:LIST[:NORMal]:<Function>	内蔵プリンタで数値データリストを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	6-58
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	6-59
IMAGeグループ		
:IMAGe?	画面イメージデータの保存に関する全設定値の問い合わせ	6-60
:IMAGe:ABORt	画面イメージデータの保存の中止	6-60
:IMAGe:COLor	保存する画面イメージデータの色調の設定/問い合わせ	6-60
:IMAGe:COMMeNt	画面下部に表示するコメントの設定/問い合わせ	6-60
:IMAGe:COMPreSSion	画面イメージデータのBMPフォーマット時のデータ圧縮の設定/問い合わせ	6-60
:IMAGe:EXECute	画面イメージデータの保存の実行	6-60
:IMAGe:FORMat	保存する画面イメージデータの出力形式の設定/問い合わせ	6-60
:IMAGe:SAVE?	画面イメージデータの保存に関する全設定値の問い合わせ	6-60
:IMAGe:SAVE:ANAMing	画面イメージデータの保存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	6-60
:IMAGe:SAVE:CDIRectory	画面イメージデータの保存先ディレクトリの変更	6-60
:IMAGe:SAVE:DRIVE	画面イメージデータの保存先ドライブの設定	6-61
:IMAGe:SAVE:NAME	画面イメージデータの保存ファイル名の設定/問い合わせ	6-61
:IMAGe:SEND?	画面イメージデータの問い合わせ	6-61

6.1 コマンド一覧表

コマンド	機能	ページ
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	6-62
[:INPut] :CFACtor	クレストファクタの設定/問い合わせ	6-62
[:INPut] :CURRent?	電流測定に関する全設定値の問い合わせ	6-62
[:INPut] :CURRent:AUTO?	全エレメントの電流オートレンジON/OFFの問い合わせ	6-62
[:INPut] :CURRent:AUTO[:ALL]	全エレメントの電流オートレンジON/OFFの一括設定	6-62
[:INPut] :CURRent:AUTO:ELEMeNt<x>	各エレメントの電流オートレンジON/OFFの設定/問い合わせ	6-63
[:INPut] :CURRent:AUTO: {SIGMA SIGMB}	結線ユニット{ΣA ΣB}に属する全エレメントの電流オートレンジON/OFFの一括設定	6-63
[:INPut] :CURRent:MODE?	全エレメントの電流モードの問い合わせ	6-63
[:INPut] :CURRent:MODE[:ALL]	全エレメントの電流モードの一括設定	6-63
[:INPut] :CURRent:MODE:ELEMeNt<x>	各エレメントの電流モードの設定/問い合わせ	6-63
[:INPut] :CURRent:MODE: {SIGMA SIGMB}	結線ユニット{ΣA ΣB}に属する全エレメントの電流モードの一括設定	6-63
[:INPut] :CURRent:RANGE?	全エレメントの電流レンジの問い合わせ	6-63
[:INPut] :CURRent:RANGE[:ALL]	全エレメントの電流レンジの一括設定	6-64
[:INPut] :CURRent:RANGE:ELEMeNt<x>	各エレメントの電流レンジの設定/問い合わせ	6-64
[:INPut] :CURRent:RANGE: {SIGMA SIGMB}	結線ユニット{ΣA ΣB}に属する全エレメントの電流レンジの一括設定	6-65
[:INPut] :CURRent:SRATio?	全エレメントの電流センサスケール定数の問い合わせ	6-65
[:INPut] :CURRent:SRATio[:ALL]	全エレメントの電流センサスケール定数の一括設定	6-65
[:INPut] :CURRent:SRATio:ELEMeNt<x>	各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	6-65
[:INPut] :FILTer?	入力フィルタに関する全設定値の問い合わせ	6-65
[:INPut] :FILTer:FREQuency?	全エレメントの周波数フィルタON/OFFの問い合わせ	6-65
[:INPut] :FILTer:FREQuency[:ALL]	全エレメントの周波数フィルタON/OFFの一括設定	6-65
[:INPut] :FILTer:FREQuency:ELEMeNt<x>	各エレメントの周波数フィルタON/OFFの設定/問い合わせ	6-66
[: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	6-66
[:INPut] :FILTer[:LINE] [:ALL]	全エレメントのラインフィルタの一括設定	6-66
[: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	6-66
[:INPut] :INDePeNdeNt	入力エレメントの個別設定ON/OFFの設定/問い合わせ	6-66
[:INPut] :MODUle?	入力エレメントタイプの問い合わせ	6-66
[:INPut] :NULL	NULL機能のON/OFFの設定/問い合わせ	6-66
[:INPut] :POVer?	ピークオーバ情報の問い合わせ	6-66
[:INPut] :SCALing?	スケールリングに関する全設定値の問い合わせ	6-66
[:INPut] :SCALing: {VT CT SFACtor}?	全エレメントのスケールリング定数の問い合わせ	6-67
[:INPut] :SCALing: {VT CT SFACtor}[:ALL]	全エレメントのスケールリング定数の一括設定	6-67
[:INPut] :SCALing: {VT 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	6-67
[: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	6-67
[: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	6-67
[: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	6-67
[:INPut] :SYNChronize?	全エレメントの同期ソースの問い合わせ	6-67
[:INPut] :SYNChronize[:ALL]	全エレメントの同期ソースの一括設定	6-67
[:INPut] :SYNChronize:ELEMeNt<x>	各エレメントの同期ソースの設定/問い合わせ	6-67
[:INPut] :SYNChronize: {SIGMA SIGMB}	結線ユニット{ΣA ΣB}に属する全エレメントの同期ソースの一括設定	6-67
[:INPut] :VOLTage?	電圧測定に関する全設定値の問い合わせ	6-68

コマンド	機能	ページ
[:INPut]:VOLTage:AUTO?	全エレメントの電圧オートレンジON/OFFの問い合わせ	6-68
[:INPut]:VOLTage:AUTO[:ALL]	全エレメントの電圧オートレンジON/OFFの一括設定	6-68
[:INPut]:VOLTage:AUTO:ELEment<x>	各エレメントの電圧オートレンジON/OFFの設定/問い合わせ	6-68
[:INPut]:VOLTage:AUTO:{SIGMA SIGMB}	結線ユニット{ΣA ΣB}に属する全エレメントの電圧オートレンジON/OFFの一括設定	6-68
[:INPut]:VOLTage:MODE?	全エレメントの電圧モードの問い合わせ	6-68
[:INPut]:VOLTage:MODE[:ALL]	全エレメントの電圧モードの一括設定	6-68
[:INPut]:VOLTage:MODE:ELEment<x>	各エレメントの電圧モードの設定/問い合わせ	6-68
[:INPut]:VOLTage:MODE:{SIGMA SIGMB}	結線ユニット{ΣA ΣB}に属する全エレメントの電圧モードの一括設定	6-68
[:INPut]:VOLTage:RANGE?	全エレメントの電圧レンジの問い合わせ	6-68
[:INPut]:VOLTage:RANGE[:ALL]	全エレメントの電圧レンジの一括設定	6-69
[:INPut]:VOLTage:RANGE:ELEment<x>	各エレメントの電圧レンジの設定/問い合わせ	6-69
[:INPut]:VOLTage:RANGE:{SIGMA SIGMB}	結線ユニット{ΣA ΣB}に属する全エレメントの電圧レンジの一括設定	6-69
[:INPut]:WIRing	結線方式の設定/問い合わせ	6-69
INTEGrateグループ		
:INTEGrate?	積算に関する全設定値の問い合わせ	6-70
: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	6-70
:INTEGrate:MODE	積算モードの設定/問い合わせ	6-70
:INTEGrate:RESet	積算値のリセット	6-70
:INTEGrate:RTIME?	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	6-70
:INTEGrate:RTIME:{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	6-70
:INTEGrate:START	積算のスタート	6-70
:INTEGrate:STAtE?	積算状態の問い合わせ	6-70
:INTEGrate:STOP	積算のストップ	6-70
:INTEGrate:TIMer	積算タイマ時間の設定/問い合わせ	6-71
MEASureグループ		
:MEASure?	演算に関する全設定値の問い合わせ	6-72
:MEASure:AVERaging?	アベレーシングに関する全設定値の問い合わせ	6-72
:MEASure:AVERaging:COUNT	アベレーシング係数の設定/問い合わせ	6-72
:MEASure:AVERaging[:STATe]	アベレーシングのON/OFFの設定/問い合わせ	6-72
:MEASure:AVERaging:TYPE	アベレーシングのタイプの設定/問い合わせ	6-73
:MEASure:COMPensation?	補正演算に関する全設定値の問い合わせ	6-73
:MEASure:COMPensation:EFFiciency	効率補正のON/OFFの設定/問い合わせ	6-73
:MEASure:COMPensation:V3A3	2電力計法の補正のON/OFFの設定/問い合わせ	6-73
:MEASure:COMPensation:WIRing?	結線補正に関する全設定値の問い合わせ	6-73
:MEASure:COMPensation:WIRing:ELEment<x>	各エレメントの結線補正の設定/問い合わせ	6-73
:MEASure:DMeasure?	デルタ演算に関する全設定値の問い合わせ	6-73
:MEASure:DMeasure[:SIGMA]	結線ユニットΣAに対するデルタ演算モードの設定/問い合わせ	6-73
:MEASure:DMeasure:SIGMB	結線ユニットΣBに対するデルタ演算モードの設定/問い合わせ	6-74
:MEASure:EFFiciency?	効率の演算に関する全設定値の問い合わせ	6-74
:MEASure:EFFiciency:ETA<x>	効率の演算式の設定/問い合わせ	6-74
:MEASure:EFFiciency:UDEF<x>	効率の演算式で使用するユーザー定義パラメータの設定/問い合わせ	6-74
:MEASure:FREQuency?	周波数測定に関する全設定値の問い合わせ	6-74
:MEASure:FREQuency:ITEM<x>	周波数測定対象の設定/問い合わせ	6-74
:MEASure:FUNCTion<x>?	ユーザー定義ファンクションに関する全設定値の問い合わせ	6-75
:MEASure:FUNCTion<x>:EXPRession	ユーザー定義ファンクションの演算式の設定/問い合わせ	6-75
:MEASure:FUNCTion<x>[:STATe]	ユーザー定義ファンクションの有効(ON)/無効(OFF)の設定/問い合わせ	6-75

6.1 コマンド一覧表

コマンド	機能	ページ
:MEASure:FUNCTION<x>:UNIT	ユーザー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	6-75
:MEASure:MHOLD	ユーザー定義ファンクションで使用するMAX HOLDファンクションの有効(ON)/無効(OFF)の設定/問い合わせ	6-75
:MEASure:PC?	Pc(Corrected Power)の演算に関する全設定値の問い合わせ	6-75
:MEASure:PC:IEC	Pc(Corrected Power)の演算式の設定/問い合わせ	6-75
:MEASure:PC:P<x>	Pc(Corrected Power)演算のためのパラメータの設定/問い合わせ	6-75
:MEASure:PHASE	位相差の表示形式の設定/問い合わせ	6-76
:MEASure:SAMPLing	サンプリング周波数の設定/問い合わせ	6-76
:MEASure:SQFormula	S(皮相電力)Q(無効電力)の演算式の設定/問い合わせ	6-76
:MEASure:SYNChronize	同期測定モードの設定/問い合わせ	6-76
MOTorグループ		
:MOTor?	モータ評価機能に関する全設定値の問い合わせ	6-77
:MOTor:FILTer?	入力フィルタに関する全設定値の問い合わせ	6-77
:MOTor:FILTer[:LINE]	ラインフィルタの設定/問い合わせ	6-77
:MOTor:PM?	モータ出力(Pm)に関する全設定値の問い合わせ	6-77
:MOTor:PM:SCALing	モータ出力演算のスケーリング係数の設定/問い合わせ	6-77
:MOTor:PM:UNIT	モータ出力演算結果に付加する単位の設定/問い合わせ	6-77
:MOTor:POLE	モータの極数の設定/問い合わせ	6-77
:MOTor:SPEEd?	回転速度(Speed)に関する全設定値の問い合わせ	6-77
:MOTor:SPEEd:AUTO	回転信号入力(アナログ入力方式)の電圧オートレンジON/OFFの設定/問い合わせ	6-77
:MOTor:SPEEd:PRANge	回転速度(パルス入力方式)のレンジの設定/問い合わせ	6-78
:MOTor:SPEEd:PULSe	回転信号入力(パルス入力)のパルス数の設定/問い合わせ	6-78
:MOTor:SPEEd:RANGe	回転信号入力(アナログ入力方式)の電圧レンジの設定/問い合わせ	6-78
:MOTor:SPEEd:SCALing	回転速度演算のスケーリング係数の設定/問い合わせ	6-78
:MOTor:SPEEd:TYPE	回転信号入力の入力タイプの設定/問い合わせ	6-78
:MOTor:SPEEd:UNIT	回転速度演算結果に付加する単位の設定/問い合わせ	6-78
:MOTor:SSpeed	同期速度(SyncSp)演算のための周波数測定ソースの設定/問い合わせ	6-78
:MOTor:SYNChronize	回転速度(Speed)/トルク(Torque)演算のための同期ソースの設定/問い合わせ	6-78
:MOTor:TORQue?	トルク(Torque)に関する全設定値の問い合わせ	6-78
:MOTor:TORQue:AUTO	トルク信号入力(アナログ入力方式)の電圧オートレンジON/OFFの設定/問い合わせ	6-79
:MOTor:TORQue:PRANge	トルク(パルス入力方式)のレンジの設定/問い合わせ	6-79
:MOTor:TORQue:RANGe	トルク信号入力(アナログ入力方式)の電圧レンジの設定/問い合わせ	6-79
:MOTor:TORQue:RATE?	トルク信号(パルス入力方式)の定格値に関する全設定値の問い合わせ	6-79
:MOTor:TORQue:RATE:{UPPer LOWer}	トルク信号(パルス入力方式)の{上限 下限}定格値の設定/問い合わせ	6-79
:MOTor:TORQue:SCALing	トルク演算のスケーリング係数の設定/問い合わせ	6-79
:MOTor:TORQue:TYPE	トルク信号入力の入力タイプの設定/問い合わせ	6-79
:MOTor:TORQue:UNIT	トルク演算結果に付加する単位の設定/問い合わせ	6-79
NUMericグループ		
:NUMeric?	数値データの出力に関する全設定値の問い合わせ	6-80
:NUMeric:CBCycle?	Cycle by Cycle測定の数値リストデータの出力に関する全設定値の問い合わせ	6-80
:NUMeric:CBCycle:END	「:NUMeric:CBCycle:VALue?」で出力される数値リストデータの出力終了サイクルの設定/問い合わせ	6-80
:NUMeric:CBCycle:ITEM	Cycle by Cycle測定の数値リストデータの出力項目(ファンクション・エレメント)の設定/問い合わせ	6-80
:NUMeric:CBCycle:START	「:NUMeric:CBCycle:VALue?」で出力される数値リストデータの出力開始サイクルの設定/問い合わせ	6-80
:NUMeric:CBCycle:VALue?	Cycle by Cycle測定の数値リストデータの問い合わせ	6-81
:NUMeric:FLICKer?	フリッカ測定の数値データの出力に関する全設定値の問い合わせ	6-81
:NUMeric:FLICKer:COUNT?	設定した観測期間で、フリッカ測定が完了した回数	6-81

コマンド	機能	ページ
:NUMERIC:FLICKer:FUNCTION?	フリッカ測定データ(可変フォーマット)の出力に関する全設定値の問い合わせ	6-81
:NUMERIC:FLICKer:FUNCTION:CLear	フリッカ測定データ(可変フォーマット)の出力項目のクリア	6-82
:NUMERIC:FLICKer:FUNCTION:DELeTe	フリッカ測定データ(可変フォーマット)の出力項目の削除	6-82
:NUMERIC:FLICKer:FUNCTION:ITEM<x>	フリッカ測定データ(可変フォーマット)の出力項目(ファンクション・エレメント・観測期間)の設定/問い合わせ	6-82
:NUMERIC:FLICKer:FUNCTION:NUMBER	「:NUMERIC:FLICKer:FUNCTION:VALue?」で送信されるフリッカ測定データの個数の設定/問い合わせ	6-83
:NUMERIC:FLICKer:FUNCTION:VALue?	フリッカ測定データ(可変フォーマット)の問い合わせ	6-83
:NUMERIC:FLICKer:INFORMatiOn?	フリッカ判定結果(可変フォーマット)の出力に関する全設定値の問い合わせ	6-83
:NUMERIC:FLICKer:INFORMatiOn:CLear	フリッカ判定結果(可変フォーマット)の出力項目のクリア	6-84
:NUMERIC:FLICKer:INFORMatiOn:DELeTe	フリッカ判定結果(可変フォーマット)の出力項目の削除	6-84
:NUMERIC:FLICKer:INFORMatiOn:ITEM<x>	フリッカ判定結果(可変フォーマット)の出力項目(ファンクション・エレメント・観測期間)の設定/問い合わせ	6-84
:NUMERIC:FLICKer:INFORMatiOn:NUMBER	「:NUMERIC:FLICKer:INFORMatiOn:VALue?」で送信されるフリッカ判定結果のデータの個数の設定/問い合わせ	6-85
:NUMERIC:FLICKer:INFORMatiOn:VALue?	フリッカ判定結果(可変フォーマット)の問い合わせ	6-85
:NUMERIC:FLICKer:JUDGement?	フリッカ判定結果(固定フォーマット)の問い合わせ	6-85
:NUMERIC:FLICKer:PERiod?	フリッカ測定における現在測定中の観測期間番号の問い合わせ	6-86
:NUMERIC:FLICKer:VALue?	フリッカ測定データ(固定フォーマット)の問い合わせ	6-86
:NUMERIC:FORMat	「:NUMERIC[:NORMal]:VALue?」または「:NUMERIC: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	6-87
:NUMERIC:HOLD	すべての数値データを保持する(ON)/解除する(OFF)の設定/問い合わせ	6-87
:NUMERIC:LIST?	高調波測定の数値リストデータの出力に関する全設定値の問い合わせ	6-87
:NUMERIC:LIST:CLear	高調波測定の数値リストデータの出力項目のクリア	6-88
:NUMERIC:LIST:DELeTe	高調波測定の数値リストデータの出力項目の削除	6-88
:NUMERIC:LIST:ITEM<x>	高調波測定の数値リストデータの出力項目(ファンクション・エレメント)の設定/問い合わせ	6-88
:NUMERIC:LIST:NUMBER	「:NUMERIC:LIST:VALue?」で送信される数値リストデータの個数の設定/問い合わせ	6-88
:NUMERIC:LIST:ORDer	高調波測定の数値リストデータの出力最高次数の設定/問い合わせ	6-88
:NUMERIC:LIST:PRESet	高調波測定の数値リストデータの出力項目パターンのプリセット	6-89
:NUMERIC:LIST:SELeCt	高調波測定の数値リストデータの出力成分の設定/問い合わせ	6-89
:NUMERIC:LIST:VALue?	高調波測定の数値リストデータの問い合わせ	6-89
:NUMERIC:NORMal?	数値データの出力に関する全設定値の問い合わせ	6-90
:NUMERIC[:NORMal]:CLear	数値データの出力項目のクリア	6-90
:NUMERIC[:NORMal]:DELeTe	数値データの出力項目の削除	6-90
:NUMERIC[:NORMal]:ITEM<x>	数値データの出力項目(ファンクション・エレメント・次数)の設定/問い合わせ	6-90
: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	6-90
:NUMERIC[:NORMal]:PRESet	数値データの出力項目パターンのプリセット	6-90
:NUMERIC[:NORMal]:VALue?	数値データの問い合わせ	6-91
RATEグループ		
:RATE	データ更新周期の設定/問い合わせ	6-97

6.1 コマンド一覧表

コマンド	機能	ページ
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	6-98
:STATus:CONDition?	状態レジスタの内容の問い合わせ	6-98
:STATus:EESE	拡張イベントイネーブルレジスタの設定/問い合わせ	6-98
: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	6-98
: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	6-98
:STATus:FILTer<x>	遷移フィルタの設定/問い合わせ	6-98
: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	6-98
: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	6-98
:STATus:SPOLl?	シリアルポールの実行	6-99
STOReグループ		
:STORe?	ストア/リコールに関するすべての設定値の問い合わせ	6-100
:STORe:COUNT	ストア回数の設定/問い合わせ	6-100
:STORe:DIRection	ストア先の設定/問い合わせ	6-100
:STORe:FILE?	ストアしたデータの保存に関するすべての設定値の問い合わせ	6-100
: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	6-100
: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	6-100
: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	6-100
:STORe:FILE:TYPE	ストアしたデータを保存するときのデータ形式の設定/問い合わせ	6-100
:STORe:INTerval	ストアインターバルの設定/問い合わせ	6-101
:STORe:ITEM	ストア対象の設定/問い合わせ	6-101
:STORe:MEMory?	ストアメモリに関するすべての設定値の問い合わせ	6-101
:STORe:MEMory:ALERT	ストアメモリを初期化するときの確認メッセージを出す/出さないの設定/問い合わせ	6-101
: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	6-101
: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	6-101
:STORe:MEMory:INITialize	ストアメモリの初期化の実行	6-101
:STORe:MODE	データのストア/リコールの設定/問い合わせ	6-101
:STORe:NUMeric?	数値データのストアに関するすべての設定値の問い合わせ	6-102
:STORe:NUMeric:NORMal?	数値データのストア項目に関するすべての設定値の問い合わせ	6-102
:STORe:NUMeric[:NORMal]:ALL	数値データをストアするときのすべてのエレメント・ファンクションの出力ON/OFFの一括設定	6-102
:STORe:NUMeric[:NORMal]:{ELEMeNt<x> SIGMA SIGMB}	数値データをストアするときの{各エレメント ΣA ΣB}の出力ON/OFFの設定/問い合わせ	6-102
:STORe:NUMeric[:NORMal]:PRESet<x>	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を決められたパターンにプリセット	6-102
:STORe:NUMeric[:NORMal]:<Function>	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	6-102
:STORe:RECall	リコールするデータ番号の設定/問い合わせ	6-103
:STORe:RTIME?	実時間ストアモードにおけるストア予約時刻の問い合わせ	6-103
: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}予約時刻の設定/問い合わせ	6-103
:STORe:SMODE	ストアモードの設定/問い合わせ	6-103
:STORe:START	データのストアの開始	6-103
:STORe:STOP	データのストアの終了	6-103
:STORe:WAVE?	波形表示データのストアに関するすべての設定値の問い合わせ	6-103

コマンド	機能	ページ
:STORE:WAVE:ALL	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の一括設定	6-103
:STORE:WAVE:{U<x> I<x> SPEEd TORQue}	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	6-103
SYSTemグループ		
:SYSTem?	システムに関する全設定値の問い合わせ	6-104
:SYSTem:CLOCK?	日付/時刻に関する全設定値の問い合わせ	6-104
:SYSTem:CLOCK:DISPlay	日付/時刻表示のON/OFFの設定/問い合わせ	6-104
:SYSTem:CLOCK:SNTP?	SNTPによる日付/時刻に関する全設定値の問い合わせ	6-104
:SYSTem:CLOCK:SNTP[:EXECute]	SNTPによる日付/時刻の設定	6-104
:SYSTem:CLOCK:SNTP:GMTTime	グリニッジ標準時との時差の設定/問い合わせ	6-104
:SYSTem:CLOCK:TYPE	日付/時刻の設定方法の設定/問い合わせ	6-104
:SYSTem:DATE	日付の設定/問い合わせ	6-104
:SYSTem:EClear	画面上に表示されているエラーメッセージのクリア	6-104
:SYSTem:FONT	画面表示フォントの設定/問い合わせ	6-104
:SYSTem:KLOCK	キーロックのON/OFFの設定/問い合わせ	6-105
:SYSTem:LANguage?	画面表示言語に関する全設定値の問い合わせ	6-105
:SYSTem:LANguage:MENU	メニューの言語の設定/問い合わせ	6-105
:SYSTem:LANguage:MESSAge	メッセージの言語の設定/問い合わせ	6-105
:SYSTem:LCD?	LCD画面に関する全設定値の問い合わせ	6-105
:SYSTem:LCD:BRIGhtness	LCD画面の輝度の設定/問い合わせ	6-105
:SYSTem:LCD:COLor?	LCD画面の表示色に関する全設定値の問い合わせ	6-105
: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	6-105
: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	6-105
: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	6-105
: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	6-106
: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	6-106
: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	6-106
:SYSTem:SLOCK	SHIFTキーのON状態を継続する(ON)/しない(OFF)の設定/問い合わせ	6-106
:SYSTem:TIME	時刻の設定/問い合わせ	6-106
:SYSTem:USBKeyboard	USBキーボードの種類の設定/問い合わせ	6-106
WAVeformグループ		
:WAVeform?	波形表示データの出力に関する全設定値の問い合わせ	6-107
: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	6-107
: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	6-107
: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	6-107
:WAVeform:HOLD	すべての波形表示データを保持する(ON)/解除する(OFF)の設定/問い合わせ	6-107
:WAVeform:LENGth?	「:WAVeform:TRACe」で指定した波形の全データ点数の問い合わせ	6-107
:WAVeform:SEND?	「:WAVeform:TRACe」で指定した波形表示データの問い合わせ	6-108
:WAVeform:SRAtE?	取り込んだ波形のサンプリングレートの問い合わせ	6-108
: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	6-108
:WAVeform:TRACe	WAVeformグループで対象となる波形の設定/問い合わせ	6-108
:WAVeform:TRIGger?	取り込んだ波形のトリガポジションの問い合わせ	6-108

6.1 コマンド一覧表

コマンド	機能	ページ
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+SINGLE)を押したのと同じ動作)の実行と, 結果の問い合わせ	6-109
*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	6-109
*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	6-109
*ESR?	標準イベントレジスタの値の問い合わせ	6-109
*IDN?	機種種の問い合わせ	6-109
*OPC	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	6-109
*OPC?	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	6-110
*OPT?	装備しているオプションの問い合わせ	6-110
*PSC	電源ON時に各レジスタをクリアするかしないかの設定/問い合わせ	6-110
*RST	設定の初期化の実行	6-110
*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	6-110
*STB?	ステータスバイトレジスタの値の問い合わせ	6-110
*TRG	シングル測定(SINGLEキーを押したのと同じ動作)を実行	6-110
*TST?	セルフテストの実行と, 結果の問い合わせ	6-111
*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	6-111

6.2 ACQquisitionグループ

ACQquisitionグループは、波形サンプリングデータ(アキュイジションデータ)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。

このグループのコマンドは、高度演算機能(オプション、/G6)搭載時のみ有効です。

:ACQquisition?

機能 波形サンプリング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 :ACQquisition?

例 :ACQUISITION? -> :ACQUISITION:
TRACE U1;FORMAT ASCII;START 0;
END 199999;HOLD 0

:ACQquisition:BYTeorder

機能 「:ACQquisition:SEND?」で送信される波形サンプリングデータ(FLOAT形式)のバイト出力順序を設定/問い合わせします。

構文 :ACQquisition:BYTeorder {LSBFirst|MSBFirst}
:ACQquisition:BYTeorder?

例 :ACQUISITION:BYTEORDER LSBFIRST
:ACQUISITION:BYTEORDER? ->
:ACQUISITION:BYTEORDER LSBFIRST

解説 この設定値は、「:ACQquisition:FORMat」がFLOatのときに有効です。

:ACQquisition:END

機能 「:ACQquisition:SEND?」で送信される波形サンプリングデータの出力終了点を設定/問い合わせします。

構文 :ACQquisition:END {<NRf>}
:ACQquisition:END?
<NRf>=0~3999999

例 :ACQUISITION:END 199999
:ACQUISITION:END? ->
:ACQUISITION:END 199999

解説 (全データ点数-1)までの範囲で設定してください。全データ点数は「:ACQquisition:LENGth?」で問い合わせることができます。

:ACQquisition:FORMat

機能 「:ACQquisition:SEND?」で送信される波形サンプリングデータのフォーマットを設定/問い合わせします。

構文 :ACQquisition:FORMat {AScii|FLOat}
:ACQquisition:FORMat?

例 :ACQUISITION:FORMAT FLOAT
:ACQUISITION:FORMAT? ->
:ACQUISITION:FORMAT FLOAT

解説 フォーマットの設定による波形サンプリングデータ出力の違いについては、「:ACQquisition:SEND?」の解説を参照してください。

:ACQquisition:HOLD

機能 すべての波形サンプリングデータを保持する(ON)/解除する(OFF)を設定/問い合わせします。

構文 :ACQquisition:HOLD {<Boolean>}
:ACQquisition:HOLD?

例 :ACQUISITION:HOLD ON
:ACQUISITION:HOLD? ->
:ACQUISITION:HOLD 1

解説 ・測定モードが「MATH」または「FFT」のときに有効です。これ以外のときはエラーとなります。

・:ACQquisition:HOLDを「ON」にすると、波形サンプリングデータのサンプリングを停止し、その時点のすべての波形サンプリングデータを内部に保持します。

「:ACQquisition:SEND?」を実行するときには、その前に必ず:ACQquisition:HOLDを「ON」にしてください。

・たとえば、同じ時点でのU1とI1の波形サンプリングデータを取得したいときには、次のようにします。

```
:ACQquisition:HOLD ON
:ACQquisition:TRACe U1
:ACQquisition:SEND?
(U1の波形サンプリングデータを受信)
:ACQquisition:TRACe I1
:ACQquisition:SEND?
(I1の波形サンプリングデータを受信)
:ACQquisition:HOLD OFF
```

・新たな波形サンプリングデータを取得するときは、一旦:ACQquisition:HOLDを「OFF」にしてサンプリングを再開させてから、再度:ACQquisition:HOLDを「ON」にしてください。

6.2 ACQuisitionグループ

:ACQuisition:LENGth?

- 機能 「:ACQuisition:TRACe」で指定した波形サンプリングデータの全データ点数を問い合わせます。
- 構文例 **:ACQuisition:LENGth?**
:ACQUISITION:LENGTH? -> 100000
- 解説
- ・測定モードが「MATH」のとき、データ点数はデータ更新周期(:RATE)の設定により決まります。
length=rate(sec)×200000
 - ・測定モードが「FFT」のとき、データ点数は、FFT演算点数(:DISPlay:FFT:POINT)の設定により次のように決まります。
(1)「:ACQuisition:TRACe」がFFT<x>のとき
length = point/2+1 = 10001 or 100001
(2)「:ACQuisition:TRACe」がFFT<x>以外のとき
length = point = 20000 or 200000
 - ・「MATH」または「FFT」以外の測定モードでは、アキュイジションデータが無いためエラーとなります。「0」を返します。

:ACQuisition:SEnD?

- 機能 「:ACQuisition:TRACe」で指定した波形サンプリングデータを問い合わせます。
- 構文例 **:ACQuisition:SEnD?**
・「:ACQuisition:FORMat」が{ASCIi}のとき
:ACQUISITION:SEnD? ->
<NR3>,<NR3>,...
・「:ACQuisition:FORMat」が{FLOat}のとき
:ACQUISITION:SEnD? -> #8(8桁のバイト数)(データバイトの並び)
- 解説
- ・出力される波形サンプリングデータの形式は、「:ACQuisition: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。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、「:ACQuisition:BYTeorder」の設定に従います。
 - ・「:ACQuisition:{StARt|ENd}」で指定された範囲の波形サンプリングデータを出力します。ただし、波形サンプリングデータの範囲「0～(全データ点数-1)」を超える部分については出力されません。

- ・測定モードが「MATH」または「FFT」で波形サンプリングデータが内部に保持されている(:ACQuisition:HOLD ON)ときに有効です。これ以外のときは波形サンプリングデータが無いためエラーとなります。出力は次のようになります。
(1)「ASCIi」のとき
「NAN」を出力します。
(2)「FLOat」のとき
「#8000000000」を出力します。

:ACQuisition:SRATe?

- 機能 取り込んだデータのサンプルレートを問い合わせます。
- 構文例 **:ACQuisition:SRATe?**
:ACQUISITION:SRATE? -> 195.312E+03
- 解説
- ・測定モードが「MATH」または「FFT」のときに有効です。これ以外のときは波形サンプリングデータが無いためエラーとなります。「NAN」を返します。

:ACQuisition:StARt

- 機能 「:ACQuisition:SEnD?」で送信される波形サンプリングデータの出力開始点を設定/問い合わせします。
- 構文例 **:ACQuisition:StARt <NRf>**
:ACQuisition:StARt?
<NRf>=0~3999999
- 例 **:ACQUISITION:StARt 0**
:ACQUISITION:StARt? ->
:ACQUISITION:StARt 0
- 解説 (全データ点数-1)までの範囲で設定してください。全データ点数は「:ACQuisition:LENGth?」で問い合わせることができます。

:ACQuisition:TRACe

- 機能 「:ACQuisition:SEnD?」の対象を設定/問い合わせします。
- 構文例 **:ACQuisition:TRACe {U<x>|I<x>|SPEEd|TORQue|MATH<x>|FFT<x>}**
:ACQuisition:TRACe?
U<x>, I<x>の<x>=1~4(エレメント)
MATH<x>の<x>=1, 2(MATH)
FFT<x>の<x>=1, 2(FFT)
- 例 **:ACQUISITION:TRACe U1**
:ACQUISITION:TRACe? ->
:ACQUISITION:TRACe U1
- 解説 {SPEEd|TORQue}は、モータバージョン(-MV)のみ有効です。

6.3 AOUTputグループ

AOUTputグループは、D/A出力に関するグループです。

フロントパネルのMISCの「D/A Output Items」メニューと同じ設定、および設定内容の問い合わせができます。

ただし、このグループのコマンドは、D/A出力(オプション、/DA)搭載時のみ有効です。

:AOUTput?

機能 D/A出力に関するすべての設定値を問い合わせます。

構文 :AOUTput?

例 :AOUTPUT? -> 「:AOUTput:NORMAL?」の応答と同じ

:AOUTput:NORMAL?

機能 D/A出力に関するすべての設定値を問い合わせます。

構文 :AOUTput:NORMAL?

例 :AOUTPUT:NORMAL? -> :AOUTPUT:
NORMAL:CHANNEL1 U,1,TOTAL;
CHANNEL2 I,1,TOTAL;
CHANNEL3 P,1,TOTAL;
CHANNEL4 S,1,TOTAL;
CHANNEL5 Q,1,TOTAL;
CHANNEL6 LAMBDA,1,TOTAL;
CHANNEL7 PHI,1,TOTAL;CHANNEL8 FU,1;
CHANNEL9 FI,1;CHANNEL10 NONE;
CHANNEL11 NONE;CHANNEL12 NONE;
CHANNEL13 NONE;CHANNEL14 NONE;
CHANNEL15 NONE;CHANNEL16 NONE;
CHANNEL17 NONE;CHANNEL18 NONE;
CHANNEL19 NONE;CHANNEL20 NONE;
MODE1 FIXED;MODE2 FIXED;
MODE3 FIXED;MODE4 FIXED;
MODE5 FIXED;MODE6 FIXED;
MODE7 FIXED;MODE8 FIXED;
MODE9 FIXED;MODE10 FIXED;
MODE11 FIXED;MODE12 FIXED;
MODE13 FIXED;MODE14 FIXED;
MODE15 FIXED;MODE16 FIXED;
MODE17 FIXED;MODE18 FIXED;
MODE19 FIXED;MODE20 FIXED;
RATE1 100.0E+00,-100.0E+00;
RATE2 100.0E+00,-100.0E+00;
RATE3 100.0E+00,-100.0E+00;
RATE4 100.0E+00,-100.0E+00;
RATE5 100.0E+00,-100.0E+00;
RATE6 100.0E+00,-100.0E+00;
RATE7 100.0E+00,-100.0E+00;
RATE8 100.0E+00,-100.0E+00;
RATE9 100.0E+00,-100.0E+00;
RATE10 100.0E+00,-100.0E+00;
RATE11 100.0E+00,-100.0E+00;
RATE12 100.0E+00,-100.0E+00;
RATE13 100.0E+00,-100.0E+00;
RATE14 100.0E+00,-100.0E+00;

RATE15 100.0E+00,-100.0E+00;
RATE16 100.0E+00,-100.0E+00;
RATE17 100.0E+00,-100.0E+00;
RATE18 100.0E+00,-100.0E+00;
RATE19 100.0E+00,-100.0E+00;
RATE20 100.0E+00,-100.0E+00;
IRTIME 1,0,0

:AOUTput[:NORMAL]:CHANNEL<x>

機能 D/A出力項目(ファンクション・エレメント・次数)を設定/問い合わせします。

構文 :AOUTput[:NORMAL]:CHANNEL<x> {NONE |
<Function>,<Element>,<Order>}
:AOUTput[:NORMAL]:CHANNEL<x>?
<x>=1~20(出力チャンネル)
NONE=出力項目なし

<Function>={U|I|P|S|Q|...} (「DISPlayグループ」のファンクション選択肢一覧(1)、6-41ページ参照)

<Element>={<NRf>|SIGMA|SIGMB}(<NRf>=1~4)

<Order>={TOTAL|DC|<NRf>}(<NRf>=1~100)

例 :AOUTPUT:NORMAL:CHANNEL1 U,1,TOTAL
:AOUTPUT:NORMAL:CHANNEL1? ->

:AOUTPUT:NORMAL:CHANNEL1 U,1,TOTAL
解説 ・<Element>を省略したときは、エレメント1が設定されます。
・<Order>を省略したときは、TOTALが設定されます。
・<Element>または<Order>が不要なファンクションの応答は、<Element>または<Order>が省略されます。

:AOUTput[:NORMAL]:IRTime

機能 積算値のD/A出力における積算定格時間を設定/問い合わせします。

構文 :AOUTput[:NORMAL]:IRTime {<NRf>,<NRf>,<NRf>}
:AOUTput[:NORMAL]:IRTime?
{<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 :AOUTPUT:NORMAL:IRTIME 1,0,0
:AOUTPUT:NORMAL:IRTIME? ->
:AOUTPUT:NORMAL:IRTIME 1,0,0

6.3 AOUTputグループ

:AOUTput[:NORMAl]:MODE<x>

機能 D/A出力項目に対する定格値設定方式を設定/問い合わせします。

構文 :AOUTput[:NORMAl]:MODE<x> {FIXed|MANual}

:AOUTput[:NORMAl]:MODE<x>?

<x>=1~20(出力チャネル)

例 :AOUTPUT:NORMAL:MODE1 FIXED

:AOUTPUT:NORMAL:MODE1? ->

:AOUTPUT:NORMAL:MODE1 FIXED

:AOUTput[:NORMAl]:RATE<x>

機能 D/A出力項目に対する定格最大・最小値をマニュアル設定/問い合わせします。

構文 :AOUTput[:NORMAl]:MODE<x> {<NRf>,<NRf>}

:AOUTput[:NORMAl]:MODE<x>?

<x>=1~20(出力チャネル)

<NRf>=-9.999E+30~9.999E+30 (定格値)

例 :AOUTPUT:NORMAL:RATE1 100,-100

:AOUTPUT:NORMAL:RATE1? ->

:AOUTPUT:NORMAL:

RATE1 100.0E+00,-100.0E+00

解説 ・ 最大値, 最小値の順で設定します。

・ 定格値設定方式

(:AOUTput[:NORMAl]:MODE<x>)が「FIXed」のときに有効な設定です。

6.4 CBCycleグループ

CBCycleグループは、Cycle by Cycle測定機能に関するグループです。

フロントパネルのITEMの「CbyC Items」メニューおよびFORMの「CbyC Form」メニューと同じ設定、および設定内容の問い合わせができます。

ただし、このグループのコマンドは、Cycle by Cycle機能(オプション、/CC)搭載時のみ有効です。

:CBCycle?

機能 Cycle by Cycle測定機能に関するすべての設定値を問い合わせます。

構文 :CBCycle?

例 :CBCYCLE? -> :CBCYCLE:SYNCHRONIZE:
SOURCE U1;SLOPE RISE;:CBCYCLE:
TRIGGER:MODE AUTO;SOURCE U1;
SLOPE RISE;LEVEL 0.0;:CBCYCLE:
COUNT 100;TIMEOUT 10;FILTER:LINE:
ELEMENT1 50.0E+03;
ELEMENT2 50.0E+03;
ELEMENT3 50.0E+03;ELEMENT4 50.0E+03

:CBCycle:COUNT

機能 Cycle by Cycle測定の測定サイクル数を設定/問い合わせします。

構文 :CBCycle:COUNT {<NRf>}

:CBCycle:COUNT?

<NRf>=1~3000(測定サイクル数)

例 :CBCYCLE:COUNT 100
:CBCYCLE:COUNT? ->
:CBCYCLE:COUNT 100

:CBCycle:DISPlay?

機能 Cycle by Cycle表示に関するすべての設定値を問い合わせます。

構文 :CBCycle:DISPlay?

例 :CBCYCLE:DISPlay? ->

:CBCYCLE:DISPlay:ITEM1 FREQ;
ITEM2 U,1;ITEM3 I,1;ITEM4 P,1;
ITEM5 S,1;CURSOR 1

:CBCycle:DISPlay:CURSor

機能 Cycle by Cycle表示のカーソル位置を設定/問い合わせします。

構文 :CBCycle:DISPlay:CURSor {<NRf>}

:CBCycle:DISPlay:CURSor?

<NRf>=1~3000(カーソル位置)

例 :CBCYCLE:DISPlay:CURSOR 1
:CBCYCLE:DISPlay:CURSOR? ->
:CBCYCLE:DISPlay:CURSOR 1

解説
・カーソル位置はサイクル番号で指定します。
・「:DISPlay:CBCycle:CURSor」でも同じ設定/問い合わせができます。

:CBCycle:DISPlay:ITEM<x>

機能 Cycle by Cycle表示の表示項目(ファンクション・エレメント)を設定/問い合わせします。

構文 :CBCycle:DISPlay:

ITEM<x> {<Function>,<Element>}

:CBCycle:DISPlay:ITEM<x>?

<x>=1~5(項目番号)

<Function>={FREQ|U|I|P|S|Q|LAMBda|SPEed|TORQue|PM}

<Element>={<NRf>|SIGMA|SIGMB}(<NRf>=1~4)

例 :CBCYCLE:DISPlay:ITEM1 U,1
:CBCYCLE:DISPlay:ITEM1? ->
:CBCYCLE:DISPlay:ITEM1 U,1

解説
・<Function>={FREQ|SPEed|TORQue|PM}のとき、<Element>の指定は不要です。応答は<Element>が省略されます。
・<Element>を省略したときは、エレメント1が設定されます。
・{SPEed|TORQue|PM}は、モータバージョン(-MV)のみ有効です。
・「:DISPlay:CBCycle:ITEM<x>」でも同じ設定/問い合わせができます。

:CBCycle:DISPlay:PAGE

機能 Cycle by Cycle表示の表示ページ番号を設定/問い合わせします。

構文 :CBCycle:DISPlay:PAGE {<NRf>}

:CBCycle:DISPlay:PAGE?

<NRf>=1~150(ページ番号)

例 :CBCYCLE:DISPlay:PAGE 1
:CBCYCLE:DISPlay:PAGE? ->
:CBCYCLE:DISPlay:PAGE 1

解説
・ページ番号を設定すると、カーソル位置は設定したページの先頭に移動します。
・「:DISPlay:CBCycle:PAGE」でも同じ設定/問い合わせができます。

:CBCycle:FILTer?

機能 Cycle by Cycle測定の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 :CBCycle:FILTer?

例 :CBCYCLE:FILTer? -> :CBCYCLE:

FILTER:LINE:ELEMENT1 50.0E+03;

ELEMENT2 50.0E+03;

ELEMENT3 50.0E+03;ELEMENT4 50.0E+03

6.4 CBCycleグループ

:CBCycle:FILTer:LINE?

機能 Cycle by Cycle測定のライン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 **:CBCycle:FILTer:LINE?**

例 **:CBCYCLE:FILTER:LINE? -> :CBCYCLE:FILTER:LINE:ELEMENT1 50.0E+03; ELEMENT2 50.0E+03; ELEMENT3 50.0E+03; ELEMENT4 50.0E+03**

:CBCycle:FILTer[:LINE][:ALL]

機能 Cycle by Cycle測定の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 **:CBCycle:FILTer[:LINE][:ALL] {OFF|<周波数>}**
OFF=ラインフィルタOFF
<周波数>=500Hz, 5.5kHz, 50kHz(ラインフィルタON, カットオフ周波数)

例 **:CBCYCLE:FILTER:LINE:ALL 50KHZ**

解説 モータ入力ラインフィルタには設定されません。

:CBCycle:FILTer[:LINE]:ELEMENT<x>

機能 Cycle by Cycle測定の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 **:CBCycle:FILTer[:LINE]:ELEMENT<x> {OFF|<周波数>}**
:CBCycle:FILTer[:LINE]:ELEMENT<x>? <x>=1~4(エレメント)
OFF=ラインフィルタOFF
<周波数>=500Hz, 5.5kHz, 50kHz(ラインフィルタON, カットオフ周波数)

例 **:CBCYCLE:FILTER:LINE:ELEMENT1 50KHZ**
:CBCYCLE:FILTER:LINE:ELEMENT1? ->
:CBCYCLE:FILTER:LINE:ELEMENT1 50.0E+03

:CBCycle:FILTer[:LINE]:MOTor

機能 Cycle by Cycle測定のモータ入力ラインフィルタを設定/問い合わせします。

構文 **:CBCycle:FILTer[:LINE]:MOTor {OFF|<周波数>}**
:CBCycle:FILTer[:LINE]:MOTor? <周波数>=100Hz, 50kHz(ラインフィルタON, カットオフ周波数)

解説 モータバージョン(-MV)のみ有効です。

:CBCycle:RESet

機能 Cycle by Cycle測定をリセットします。

構文 **:CBCycle:RESet**

例 **:CBCYCLE:RESET**

:CBCycle:STARt

機能 Cycle by Cycle測定を開始します。

構文 **:CBCycle:STARt**

例 **:CBCYCLE:START**

:CBCycle:STATe?

機能 Cycle by Cycle測定ステータスを問い合わせます。

構文 **:CBCycle:STATe?**

例 **:CBCYCLE:STATE? -> RESET**

解説 応答の内容は次のとおりです。
RESet=リセット状態
STARt=測定中
COMPlEte=測定終了・測定結果表示
TIMEout=タイムアウト
ERRFreq=測定終了(周波数測定エラー発生)

:CBCycle:SYNChronize?

機能 Cycle by Cycle測定の同期ソースに関するすべての設定値を問い合わせます。

構文 **:CBCycle:SYNChronize?**

例 **:CBCYCLE:SYNCHRONIZE? ->**
:CBCYCLE:SYNCHRONIZE:SOURCE U1;
SLOPE RISE

:CBCycle:SYNChronize:SLOPe

機能 Cycle by Cycle測定の同期ソースのスロープを設定/問い合わせします。

構文 **:CBCycle:SYNChronize:SLOPe {RISE|FALL}**
:CBCycle:SYNChronize:SLOPe?

例 **:CBCYCLE:SYNCHRONIZE:SLOPE RISE**
:CBCYCLE:SYNCHRONIZE:SLOPE? ->
:CBCYCLE:SYNCHRONIZE:SLOPE RISE

:CBCycle:SYNChronize:SOURce

機能 Cycle by Cycle測定の同期ソースを設定/問い合わせします。

構文 **:CBCycle:SYNChronize:SOURce {U<x>|I<x>|EXTErnal}**
:CBCycle:SYNChronize:SOURce? <x>=1~4(エレメント)
EXTErnal=外部クロック入力(Ext Clk)

例 **:CBCYCLE:SYNCHRONIZE:SOURCE U1**
:CBCYCLE:SYNCHRONIZE:SOURCE? ->
:CBCYCLE:SYNCHRONIZE:SOURCE U1

:CBCycle:TIMEout

機能 Cycle by Cycle測定のタイムアウト時間を設定/問い合わせします。

構文 **:CBCycle:TIMEout {<NRf>}**
:CBCycle:TIMEout? <NRf>=0~3600(秒)
(0はタイムアウトなし)

例 **:CBCYCLE:TIMEOUT 10**
:CBCYCLE:TIMEOUT? ->
:CBCYCLE:TIMEOUT 10

:CBCycle: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 :CBCycle:TRIGger?

例 :CBCYCLE:TRIGGER? ->
:CBCYCLE:TRIGGER:MODE AUTO;
SOURCE U1;SLOPE RISE;LEVEL 0.0

解説 「:DISPlay:WAVE:TRIGger?」と同じ問い合わせとなります。

:CBCycle:TRIGger:LEVel

機能 トリガレベルを設定/問い合わせします。

構文 :CBCycle:TRIGger:LEVel {<Nrf>}
:CBCycle:TRIGger:LEVel?
<Nrf>=-100.0~100.0(%)

例 :CBCYCLE:TRIGGER:LEVEL 0
:CBCYCLE:TRIGGER:LEVEL? ->
:CBCYCLE:TRIGGER:LEVEL 0.0

解説 「:DISPlay:WAVE:TRIGger:LEVel」と同じ設定/問い合わせとなります。

:CBCycle:TRIGger:MODE

機能 トリガモードを設定/問い合わせします。

構文 :CBCycle:TRIGger:MODE {AUTO|NORMal}
:CBCycle:TRIGger:MODE?

例 :CBCYCLE:TRIGGER:MODE AUTO
:CBCYCLE:TRIGGER:MODE? ->
:CBCYCLE:TRIGGER:MODE AUTO

解説 「:DISPlay:WAVE:TRIGger:MODE」と同じ設定/問い合わせとなります。

:CBCycle:TRIGger:SLOPe

機能 トリガスロープを設定/問い合わせします。

構文 :CBCycle:TRIGger:SLOPe {RISE|FALL|BOTH}
:CBCycle:TRIGger:SLOPe?

例 :CBCYCLE:TRIGGER:SLOPE RISE
:CBCYCLE:TRIGGER:SLOPE? ->
:CBCYCLE:TRIGGER:SLOPE RISE

解説 「:DISPlay:WAVE:TRIGger:SLOPe」と同じ設定/問い合わせとなります。

:CBCycle:TRIGger:SOURce

機能 トリガソースを設定/問い合わせします。

構文 :CBCycle:TRIGger:SOURce {U<x>|I<x>|EXTErnal}
:CBCycle:TRIGger:SOURce?
<x>=1~4(エレメント)
EXTErnal=外部トリガ入力(Ext Clk)

例 :CBCYCLE:TRIGGER:SOURCE U1
:CBCYCLE:TRIGGER:SOURCE? ->
:CBCYCLE:TRIGGER:SOURCE U1

解説 「:DISPlay:WAVE:TRIGger:SOURce」と同じ設定/問い合わせとなります。

6.5 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:MODE NUMERIC)、付けずに返送するか(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 RS-232/USB/イーサネット(いずれもオプション)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, 6-23
ページ参照

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(6-23ページ)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, 6-23
ページ参照

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、5-7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(6-23ページ)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 RS-232/USB/イーサネット(いずれもオプション)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

USB/イーサネット(いずれもオプション)インタフェースでは、常に0を返します。

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 **:INPUT:VOLTAGE:RANGE:ELEMENT1 1.000E+03**)、省略形で返送するか(例 **:VOLT:RANG:ELEM 1.000E+03**)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, 7-7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、5-8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, 7-7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

ビット6(ACS)=1のとき:
 メディアへのアクセス未完了

6.6 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+MEASURE)と同じ設定、および設定内容の問い合わせができます。

: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 10.0E-03; POSITION2 40.0E-03;;CURSOR:BAR: STATE 0;POSITION1 1;POSITION2 15;; CURSOR:TREND:STATE 0;TRACE1 1; TRACE2 2;POSITION1 100; POSITION2 900

:CURSor:BAR?

機能	バーグラフ表示の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:BAR?
例	:CURSOR:BAR? -> :CURSOR:BAR: STATE 1;POSITION1 1;POSITION2 15
解説	高度演算機能(オプション、/G6)または高調波測定機能(オプション、/G5)搭載時のみ有効です。

:CURSor:BAR:POSition<x>

機能	バーグラフ表示のカーソル位置(次数)を設定/問い合わせします。
構文	:CURSor:BAR:POSition<x> {<NRf>} :CURSor:BAR:POSition<x>? <x>=1, 2(1=C1+, 2=C2 x) <NRf>=0~100
例	:CURSOR:BAR:POSITION1 1 :CURSOR:BAR:POSITION1? -> :CURSOR:BAR:POSITION1 1
解説	高度演算機能(オプション、/G6)または高調波測定機能(オプション、/G5)搭載時のみ有効です。

:CURSor:BAR[:STATe]

機能	バーグラフ表示のカーソル表示のON/OFFを設定/問い合わせします。
構文	:CURSor:BAR[:STATe] {<Boolean>} :CURSor:BAR:STATe?
例	:CURSOR:BAR:STATE ON :CURSOR:BAR:STATE? -> :CURSOR:BAR:STATE 1
解説	高度演算機能(オプション、/G6)または高調波測定機能(オプション、/G5)搭載時のみ有効です。

:CURSor:BAR:{Y<x>|DY}?

機能	バーグラフ表示のカーソル測定値を問い合わせます。
構文	:CURSor:BAR:{Y<x> DY}? Y<x>=カーソル位置のY軸値(Y1=Y1+, Y2+, Y3+ Y2=Y1x, Y2x, Y3x) DY=カーソル間のY軸値($\Delta Y1$, $\Delta Y2$, $\Delta Y3$) <x>=1, 2(1=C1+, 2=C2 x)
例	:CURSOR:BAR:Y1? -> 78.628E+00
解説	・ 高度演算機能(オプション、/G6)または高調波測定機能(オプション、/G5)搭載時のみ有効です。 ・ 複数のバーグラフが表示されている場合は、それぞれのバーグラフのカーソル測定値を順に返します。 ・ バーグラフでカーソル表示がONになっていないときは「NAN(Not A Number)」を返します。

:CURSor:FFT?

機能	FFT波形表示の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:FFT?
例	:CURSOR:FFT? -> :CURSOR:FFT: STATE 0;TRACE1 FFT1;TRACE2 FFT2; POSITION1 100;POSITION2 900
解説	高度演算機能(オプション、/G6)搭載時のみ有効です。

:CURSor:FFT:POSition<x>

機能	FFT波形表示のカーソル位置を設定/問い合わせします。
構文	:CURSor:FFT:POSition<x> {<NRf>} :CURSor:FFT:POSition<x>? <x>=1, 2(1=C1+, 2=C2 x) <NRf>=0~1001
例	:CURSOR:FFT:POSITION1 20 :CURSOR:FFT:POSITION1? -> :CURSOR:FFT:POSITION1 20
解説	高度演算機能(オプション、/G6)搭載時のみ有効です。

:CURSor:FFT[:STATe]

機能 FFT波形表示のカーソル表示のON/OFFを設定/問い合わせします。

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

例 :CURSOR:FFT:STATE OFF
:CURSOR:FFT:STATE? ->
:CURSOR:FFT:STATE 0

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:CURSor:FFT:TRACe<x>

機能 FFT波形表示のカーソルの対象を設定/問い合わせします。

構文 :CURSor:FFT:TRACe<x> {FFT<x>}
:CURSor:FFT:TRACe<x>?
TRACe<x>の<x>=1, 2(1=C1 +, 2=C2 x)
FFT<x>の<x>=1, 2(FFT)

例 :CURSOR:FFT:TRACE1 FFT1
:CURSOR:FFT:TRACE1? ->
:CURSOR:FFT:TRACE1 FFT1

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

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

機能 FFT波形表示のカーソル測定値を問い合わせします。

構文 :CURSor:FFT:{X<x>|DX|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)
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(1=C1 +, 2=C2 x)

例 :CURSOR:FFT:Y1? -> 78.628E+00

解説 ・高度演算機能(オプション, /G6)搭載時のみ有効です。
・FFT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。

:CURSor:TREnd?

機能 トレンド表示のカーソル測定に関するすべての設定値を問い合わせします。

構文 :CURSor:TREnd?

例 :CURSOR:TREND? -> :CURSOR:TREND:
STATE 1;TRACE1 1;TRACE2 2;
POSITION1 100;POSITION2 900

:CURSor:TREnd:POSition<x>

機能 トレンド表示の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(1=C1 +, 2=C2 x)
<Nrf>=0~1001

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

:CURSor:TREnd[:STATe]

機能 トレンド表示のカーソル表示の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 トレンド表示の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(1=C1 +, 2=C2 x)
<Nrf>=1~16 (T1~T16)

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

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

機能 トレンド表示のカーソル測定値を問い合わせします。

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(1=C1 +, 2=C2 x)

例 :CURSOR:TREND:X1? ->
"2005/01/01 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。

6.6 CURSorグループ

:CURSor:WAVE?

機能 波形表示の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 10.0E-03;
POSITION2 40.0E-03

:CURSor:WAVE:PATH

機能 波形表示の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 波形表示の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(1=C1+, 2=C2 x)

<時間>=0~20.00s

例 :CURSOR:WAVE:POSITION1 10MS

:CURSOR:WAVE:POSITION1? ->

:CURSOR:WAVE:POSITION1 10.0E-03

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

:CURSor:WAVE[:STATe]

機能 波形表示のカーソル表示の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 波形表示の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>|SPEEd|TORQue|MATH<x>}

:CURSor:WAVE:TRACe<x>?

TRACe<x>の<x>=1, 2(1=C1+, 2=C2 x)

U<x>, I<x>の<x>=1~4

MATH<x>の<x>=1, 2(MATH)

例 :CURSOR:WAVE:TRACE1 U1

:CURSOR:WAVE:TRACE1? ->

:CURSOR:WAVE:TRACE1 U1

解説 ・{SPEEd|TORQue}は、モータバージョン(-MV)のみ有効です。
・MATH<x>は、高度演算機能(オプション、/G6)搭載時のみ有効です。

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

機能 波形表示の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/DT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(1=C1+, 2=C2 x)

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。

6.7 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エリアおよびITEM & ELEMENTエリアの各種キーと同じ設定、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:MODE)が「NUMeric(数値表示)」のときの例 :DISPlay? -> :DISPLAY:MODE NUMERIC;(「:DISPlay:NUMeric?」の応答から最初の「:DISPLAY:」の部分を除いたもの);:DISPLAY:INFORMATION:STATE 0; PAGE 1 - 表示方式(:DISPlay:MODE)が「WAVE(波形表示)」のときの例 :DISPlay? -> :DISPLAY:MODE WAVE; (「:DISPlay:WAVE?」の応答から最初の「:DISPLAY:」の部分を除いたもの); ;:DISPLAY:INFORMATION:STATE 0; PAGE 1 - 表示方式(:DISPlay:MODE)が「NWAVE」のときの例 :DISPlay? -> :DISPLAY:MODE NWAVE; (「:DISPlay:NUMeric?」の応答から最初の「:DISPLAY:」の部分を除いたもの); (「:DISPlay:WAVE?」の応答と同じ) ;:DISPLAY:INFORMATION:STATE 0; PAGE 1
解説	表示方式(:DISPlay:MODE)に該当するすべての設定値を返します。

:DISPlay:BAR?

機能	バーグラフに関するすべての設定値を問い合わせます。
構文	:DISPlay:BAR?
例	:DISPLAY:BAR? -> :DISPLAY:BAR:FORMAT SINGLE;ITEM1 U,1;ITEM2 I,1;ITEM3 P,1;ORDER 1,100
解説	高度演算機能(オプション、/G6)または高調波測定機能(オプション、/G5)搭載時のみ有効です。

:DISPlay:BAR:FORMat

機能	バーグラフの表示フォーマットを設定/問い合わせします。
構文	:DISPlay:BAR:FORMat {SINGLE DUAL TRIad}
例	:DISPlay:BAR:FORMat SINGLE :DISPLAY:BAR:FORMat? -> :DISPLAY:BAR:FORMat SINGLE
解説	高度演算機能(オプション、/G6)または高調波測定機能(オプション、/G5)搭載時のみ有効です。

:DISPlay:BAR:ITEM<x>

機能	バーグラフ項目(ファンクション・エレメント)を設定/問い合わせします。
構文	:DISPlay:BAR:ITEM<x> {<Function>,<Element>} :DISPlay:BAR:ITEM<x>? <x>=1~3(項目番号) <Function>={U I P S Q LAMBda ...} (ファンクション選択肢一覧(2)、6-43ページ参照) <Element>=1~4
例	:DISPLAY:BAR:ITEM1 U,1 :DISPLAY:BAR:ITEM1? -> :DISPLAY:BAR:ITEM1 U,1
解説	高度演算機能(オプション、/G6)または高調波測定機能(オプション、/G5)搭載時のみ有効です。

:DISPlay:BAR:ORDER

機能	バーグラフの開始/終了次数を設定/問い合わせします。
構文	:DISPlay:BAR:ORDER {<NRf>,<NRf>} :DISPlay:BAR:ORDER? 1つ目の<NRf>=0~90(表示開始次数) 2つ目の<NRf>=10~100(表示終了次数)
例	:DISPLAY:BAR:ORDER 1,100 :DISPLAY:BAR:ORDER? -> :DISPLAY:BAR:ORDER 1,100
解説	- 高度演算機能(オプション、/G6)または高調波測定機能(オプション、/G5)搭載時のみ有効です。 - 開始次数、終了次数の順で設定します。 - 終了次数は(開始次数+10)以上になるように設定してください。

6.7 DISPlayグループ

:DISPlay:CBCycle?

機能	Cycle by Cycle表示に関するすべての設定値を問い合わせます。
構文	:DISPlay:CBCycle?
例	:DISPlay:CBCycle? -> :DISPlay:CBCycle:ITEM1 FREQ;ITEM2 U,1;ITEM3 I,1;ITEM4 P,1;ITEM5 S,1;CURSOR 1
解説	Cycle by Cycle機能(オプション, /CC)搭載時のみ有効です。

:DISPlay:CBCycle:CUSor

機能	Cycle by Cycle表示のカーソル位置を設定/問い合わせします。
構文	:DISPlay:CBCycle:CUSor {<Nrf>} :DISPlay:CBCycle:CUSor?
例	<Nrf>=1~3000(カーソル位置) :DISPlay:CBCycle:CUSor 1 :DISPlay:CBCycle:CUSor? -> :DISPlay:CBCycle:CUSor 1
解説	・ Cycle by Cycle機能(オプション, /CC)搭載時のみ有効です。 ・ カーソル位置はサイクル番号で指定します。

:DISPlay:CBCycle:ITEM<x>

機能	Cycle by Cycle表示の表示項目(ファンクション・エレメント)を設定/問い合わせします。
構文	:DISPlay:CBCycle: ITEM<x> {<Function>,<Element>} :DISPlay:CBCycle:ITEM<x>? <x>=1~5(項目番号) <Function>={FREQ U I P S Q LAMBda SPEEd TORQue PM} <Element>={<Nrf> SIGMA SIGMB} (<Nrf>=1~4)
例	:DISPlay:CBCycle:ITEM1 U,1 :DISPlay:CBCycle:ITEM1? -> :DISPlay:CBCycle:ITEM1 U,1
解説	・ Cycle by Cycle機能(オプション, /CC)搭載時のみ有効です。 ・ <Function>={FREQ SPEEd TORQue PM}のとき、<Element>の指定は不要です。応答は<Element>が省略されます。 ・ <Element>を省略したときは、エレメント1が設定されます。 ・ {SPEEd TORQue PM}は、モータバージョン(-MV)のみ有効です。

:DISPlay:CBCycle:PAGE

機能	Cycle by Cycle表示の表示ページ番号を設定/問い合わせします。
構文	:DISPlay:CBCycle:PAGE {<Nrf>} :DISPlay:CBCycle:PAGE? <Nrf>=1~150(ページ番号)
例	:DISPlay:CBCycle:PAGE 1 :DISPlay:CBCycle:PAGE? -> :DISPlay:CBCycle:PAGE 1
解説	・ Cycle by Cycle機能(オプション, /CC)搭載時のみ有効です。 ・ ページ番号を設定すると、カーソル位置は設定したページの先頭に移動します。

:DISPlay:FFT?

機能	FFT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:FFT?
例	:DISPlay:FFT? -> :DISPlay:FFT: FORMAT SINGLE;POINT 20000; WINDOW RECTANGLE;SCOPE 0,10000; VSCALE LOG;SPECTRUM LINE;FFT1: STATE 1;OBJECT U1;LABEL "FFT1";: DISPlay:FFT:FFT2:STATE 1;OBJECT I1; LABEL "FFT2"
解説	高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:FFT:FFT<x>?

機能	FFT波形に関するすべての設定値を問い合わせます。
構文	:DISPlay:FFT:FFT<x>? <x>=1, 2(FFT)
例	:DISPlay:FFT:FFT1? -> :DISPlay:FFT:FFT1:STATE 1;OBJECT U1;LABEL "FFT1"
解説	高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:FFT:FFT<x>:LABel

機能	FFT波形のラベル名を設定/問い合わせします。
構文	:DISPlay:FFT:FFT<x>:LABel {<文字列>} :DISPlay:FFT:FFT<x>:LABel? <x>=1, 2(FFT) <文字列>=8文字以内
例	:DISPlay:FFT:FFT1:LABel "FFT1" :DISPlay:FFT:FFT1:LABel? -> :DISPlay:FFT:FFT1:LABel "FFT1"
解説	高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:FFT:FFT<x>:OBJect

機能 FFT演算の対象波形を設定/問い合わせします。

構文 **:DISPlay:FFT:FFT<x>:OBJect** {U<x>|I<x>|P<x>|PA|PB|Q<x>|QA|QB|SPEed|TORQue}
:DISPlay:FFT:FFT<x>:OBJect?
 FFT<x>の<x>=1, 2(FFT)
 U<x>, I<x>, P<x>, Q<x>の<x>=1~4(エレメント)
 PA, QA=PΣA, QΣA(2~4エレメントモデルのみ)
 PB, QB=PΣB, QΣB(4エレメントモデルのみ)

例 **:DISPLAY:FFT:FFT1:OBJECT U1**
:DISPLAY:FFT:FFT1:OBJECT? ->
:DISPLAY:FFT:FFT1:OBJECT U1

解説
 ・高度演算機能(オプション, /G6)搭載時のみ有効です。
 ・{SPEed|TORQue}は、モータバージョン(-MV)のみ有効です。

:DISPlay:FFT:FFT<x>[:STATE]

機能 FFT波形表示のON/OFFを設定/問い合わせします。

構文 **:DISPlay:FFT:**
FFT<x>[:STATE] {<Boolean>}
:DISPlay:FFT:FFT<x>:STATE?
 <x>=1, 2(FFT)

例 **:DISPLAY:FFT:FFT1:STATE ON**
:DISPLAY:FFT:FFT1:STATE? ->
:DISPLAY:FFT:FFT1:STATE 1

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:FFT:FORMat

機能 FFT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:FFT:FORMat** {SINGLE|DUAL}
:DISPlay:FFT:FORMat?

例 **:DISPLAY:FFT:FORMAT SINGLE**
:DISPLAY:FFT:FORMAT? ->

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:FFT:POINT

機能 FFT演算の点数を設定/問い合わせします。

構文 **:DISPlay:FFT:POINT** {<Nrf>}
:DISPlay:FFT:POINT?
 <Nrf>=20000, 200000

例 **:DISPLAY:FFT:POINT 20000**
:DISPLAY:FFT:POINT? ->
:DISPLAY:FFT:POINT 20000

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:FFT:SCOpe

機能 FFT波形の表示範囲を設定/問い合わせします。

構文 **:DISPlay:FFT:SCOpe** {<Nrf>, <Nrf>}
:DISPlay:FFT:SCOpe?
 1つ目の<Nrf>=0~99990(表示開始点)
 2つ目の<Nrf>=10~100000(表示終了点)

例 **:DISPLAY:FFT:SCOPE 0,10000**
:DISPLAY:FFT:SCOPE? ->
:DISPLAY:FFT:SCOPE 0,10000

解説
 ・高度演算機能(オプション, /G6)搭載時のみ有効です。
 ・開始点, 終了点の順で設定します。
 ・終了点は(開始点+10)以上になるように設定してください。

:DISPlay:FFT:SPECTrum

機能 FFT波形の表示スペクトル形式を設定/問い合わせします。

構文 **:DISPlay:FFT:SPECTrum** {LINE|BAR}
:DISPlay:FFT:SPECTrum?

例 **:DISPLAY:FFT:SPECTRUM LINE**
:DISPLAY:FFT:SPECTRUM? ->

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:FFT:VScale

機能 FFT波形の垂直軸(Vertical)の表示スケールを設定/問い合わせします。

構文 **:DISPlay:FFT:VScale** {LINEar|LOG}
:DISPlay:FFT:VScale?

例 **:DISPLAY:FFT:VSCALE LOG**
:DISPLAY:FFT:VSCALE? ->
:DISPLAY:FFT:VSCALE LOG

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:FFT:WINDOW

機能 FFT演算の窓関数を設定/問い合わせします。

構文 **:DISPlay:FFT:WINDOW** {RECTangle|HANNing|FLATtop}
:DISPlay:FFT:WINDOW?

例 **:DISPLAY:FFT:WINDOW RECTANGLE**
:DISPLAY:FFT:WINDOW? ->

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:FLICKer?

機能 フリッカ測定表示に関するすべての設定値を問い合わせします。

構文 **:DISPlay:FLICKer?**

例 **:DISPLAY:FLICKER? ->**

解説 **:DISPLAY:FLICKER:ELEMENT 1;PERIOD 1**
 フリッカ測定機能(オプション, /FL)搭載時のみ有効です。

6.7 DISPlayグループ

:DISPlay:FLICKer:ELEMeNt

機能	フリッカ測定表示の表示対象エレメントを設定/問い合わせします。
構文	:DISPlay:FLICKer:ELEMeNt {<NRf>} :DISPlay:FLICKer:ELEMeNt? <NRf>=1~4(エレメント)
例	:DISPLAY:FLICKER:ELEMENT 1 :DISPLAY:FLICKER:ELEMENT? -> :DISPLAY:FLICKER:ELEMENT 1
解説	フリッカ測定機能(オプション, /FL)搭載時のみ有効です。

:DISPlay:FLICKer:PAGE

機能	フリッカ測定表示の表示ページ番号を設定/問い合わせします。
構文	:DISPlay:FLICKer:PAGE {<NRf>} :DISPlay:FLICKer:PAGE? <NRf>=1~9(ページ番号)
例	:DISPLAY:FLICKER:PAGE 1 :DISPLAY:FLICKER:PAGE? -> :DISPLAY:FLICKER:PAGE 1
解説	フリッカ測定機能(オプション, /FL)搭載時のみ有効です。

:DISPlay:FLICKer:PERIOD

機能	フリッカ測定表示の表示観測期間番号を設定/問い合わせします。
構文	:DISPlay:FLICKer:PERIOD {<NRf>} :DISPlay:FLICKer:PERIOD? <NRf>=1~99(観測期間番号)
例	:DISPLAY:FLICKER:PERIOD 1 :DISPLAY:FLICKER:PERIOD? -> :DISPLAY:FLICKER:PERIOD 1
解説	フリッカ測定機能(オプション, /FL)搭載時のみ有効です。

:DISPlay:INFORmation?

機能	設定情報の一覧表示に関するすべての設定値を問い合わせます。
構文	:DISPlay:INFORmation?
例	:DISPLAY:INFORMATION? -> :DISPLAY:INFORMATION:STATE 0;PAGE 1

:DISPlay:INFORmation:PAGE

機能	設定情報の一覧表示における表示ページ番号を設定/問い合わせします。
構文	:DISPlay:INFORmation {<NRf>} :DISPlay:INFORmation? <NRf>=1~4(ページ番号)
例	:DISPLAY:INFORMATION:PAGE 1 :DISPLAY:INFORMATION:PAGE? -> :DISPLAY:INFORMATION:PAGE 1

:DISPlay:INFORmation[:STATe]

機能	設定情報の一覧表示のON/OFFを設定/問い合わせします。
構文	:DISPlay:INFORmation [:STATe] {<Boolean>} :DISPlay:INFORmation:STATe?
例	:DISPLAY:INFORMATION:STATE ON :DISPLAY:INFORMATION:STATE? -> :DISPLAY:INFORMATION:STATE 1

:DISPlay:MATH?

機能	演算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:MATH?
例	:DISPLAY:MATH? -> :DISPLAY:MATH: MATH1:EXPRESSION "U1*I1";SCALING: MODE AUTO;CENTER 0.0000E+00; SDIV 25.000E+00;:DISPLAY:MATH: MATH1:UNIT "W":LABEL "Math1";: DISPLAY:MATH:MATH2: EXPRESSION "ABS(U1)";SCALING: MODE AUTO;CENTER 0.0000E+00; SDIV 25.000E+00;:DISPLAY:MATH: MATH2:UNIT "V":LABEL "Math2";: DISPLAY:MATH:CONSTANT1 1.0000E+00; CONSTANT2 2.0000E+00; CONSTANT3 3.0000E+00; CONSTANT4 4.0000E+00; CONSTANT5 5.0000E+00; CONSTANT6 6.0000E+00; CONSTANT7 7.0000E+00; CONSTANT8 8.0000E+00
解説	高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:MATH:CONStant<x>

機能	波形演算式で使用する定数を設定/問い合わせします。
構文	:DISPlay:MATH:CONStant<x> {<NRf>} :DISPlay:MATH:CONStant<x>? <x>=1~8(K1~K8) <NRf>=-9.9999E+30~9.9999E+30
例	:DISPLAY:MATH:CONSTANT1 1.0000E+00 :DISPLAY:MATH:CONSTANT1? -> :DISPLAY:MATH:CONSTANT1 1.0000E+00
解説	高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:MATH:MATH<x>?

機能 演算波形に関するすべての設定値を問い合わせます。

構文 :DISPlay:MATH:MATH<x>?
<x>=1, 2(MATH)

例 :DISPLAY:MATH:MATH1? -> :DISPLAY:
MATH:MATH1:EXPRESSION "U1*I1";
SCALING:MODE AUTO;
CENTER 0.0000E+00;SDIV 25.000E+00;;
DISPLAY:MATH:MATH1:UNIT "W":
LABEL "Math1"

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:MATH:MATH<x>:EXPRession

機能 波形演算の演算式を設定/問い合わせします。

構文 :DISPlay:MATH:MATH<x>:EXPRession {<文字列>}
:DISPlay:MATH:MATH<x>:EXPRession?
<x>=1, 2(MATH)
<文字列>=50文字以内

例 :DISPLAY:MATH:MATH1:
EXPRESSION "U1*I1"
:DISPLAY:MATH:MATH1:EXPRESSION? ->
:DISPLAY:MATH:MATH1:
EXPRESSION "U1*I1"

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:MATH:MATH<x>:LABel

機能 演算波形のラベル名を設定/問い合わせします。

構文 :DISPlay:MATH:MATH<x>:LABel {<文字列>}
:DISPlay:MATH:MATH<x>:LABel?
<x>=1, 2(MATH)
<文字列>=8文字以内

例 :DISPLAY:MATH:MATH1:LABEL "Math1"
:DISPLAY:MATH:MATH1:LABEL? ->
:DISPLAY:MATH:MATH1:LABEL "Math1"

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:MATH:MATH<x>:SCALing?

機能 演算波形のスケールに関するすべての設定値を問い合わせます。

構文 :DISPlay:MATH:MATH<x>:SCALing?
<x>=1, 2(MATH)

例 :DISPLAY:MATH:MATH1? -> :DISPLAY:
MATH:MATH1:SCALING:MODE AUTO;
CENTER 0.0000E+00;SDIV 25.000E+00

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:MATH:MATH<x>:SCALing:CENTer

機能 演算波形のマニュアルスケールリングの中心値を設定/問い合わせします。

構文 :DISPlay:MATH:MATH<x>:SCALing:
CENTer {<Nrf>}
:DISPlay:MATH:MATH<x>:SCALing:
CENTer?
<x>=1, 2(MATH)
<Nrf>=-9.9999E+30~9.9999E+30

例 :DISPLAY:MATH:MATH1:SCALING:
CENTER 0.0000E+00
:DISPLAY:MATH:MATH1:SCALING:CENTer?
-> :DISPLAY:MATH:MATH1:SCALING:
CENTER 0.0000E+00

解説 ・高度演算機能(オプション, /G6)搭載時のみ有効です。
・演算波形のスケールリング方式
(:DISPlay:MATH:MATH<x>:SCALing:
MODE)が「MANual」のときに有効な設定です。

:DISPlay:MATH:MATH<x>:SCALing:MODE

機能 演算波形のスケールリング方式を設定/問い合わせします。

構文 :DISPlay:MATH:MATH<x>:SCALing:
MODE {AUTO|MANual}
:DISPlay:MATH:MATH<x>:SCALing:MODE?
<x>=1, 2(MATH)

例 :DISPLAY:MATH:MATH1:SCALING:
MODE AUTO
:DISPLAY:MATH:MATH1:SCALING:MODE?
-> :DISPLAY:MATH:MATH1:SCALING:
MODE AUTO

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:MATH:MATH<x>:SCALing:SDIV

機能 演算波形のマニュアルスケールリングのScale/div値を設定/問い合わせします。

構文 :DISPlay:MATH:MATH<x>:SCALing:
SDIV {<Nrf>}
:DISPlay:MATH:MATH<x>:SCALing:SDIV?
<x>=1, 2(MATH)
<Nrf>=-9.9999E+30~9.9999E+30

例 :DISPLAY:MATH:MATH1:SCALING:
SDIV 2.5000E+01
:DISPLAY:MATH:MATH1:SCALING:SDIV?
-> :DISPLAY:MATH:MATH1:SCALING:
SDIV 25.000E+00

解説 ・高度演算機能(オプション, /G6)搭載時のみ有効です。
・演算波形のスケールリング方式
(:DISPlay:MATH:MATH<x>:SCALing:
MODE)が「MANual」のときに有効な設定です。

6.7 DISPlayグループ

:DISPlay:MATH:MATH<x>:UNIT

機能	波形演算結果に付加する単位を設定/問い合わせします。
構文	:DISPlay:MATH:MATH<x>:UNIT {<文字列>} :DISPlay:MATH:MATH<x>:UNIT? <x>=1, 2(MATH) <文字列>=8文字以内
例	:DISPlay:MATH:MATH1:UNIT "W" :DISPlay:MATH:MATH1:UNIT? -> :DISPlay:MATH:MATH1:UNIT "W"
解説	高度演算機能(オプション, /G6)搭載時のみ有効です。

:DISPlay:MODE

機能	表示方式を設定/問い合わせします。
構文	:DISPlay:MODE {NUMeric WAVE BAR VECTor TREND NWAVE NBAR NTREnd WBAR WTREnd BTREnd MATH NMATH FFT NFFT WFFT IECHarm CBCycle FLICKer} :DISPlay:MODE? NUMeric=数値のみ表示 WAVE=波形のみ表示 BAR=バーグラフ VECTor=ベクトル表示 TREND=トレンド NWAVE=数値と波形を両方表示 NBAR=数値とバーグラフを両方表示 NTREnd=数値とトレンドを両方表示 WBAR=波形とバーグラフを両方表示 WTREnd=波形とトレンドを両方表示 BTREnd=バーグラフとトレンドを両方表示 MATH=波形(演算波形を含む)のみ表示 NMATH=数値と波形(演算波形を含む)を両方表示 FFT=FFT波形のみ表示 NFFT=数値とFFT波形を両方表示 WFFT=波形とFFT波形を両方表示 IECHarm=IEC高調波測定モード表示(数値) CBCycle=Cycle by Cycleモード表示(数値) FLICKer=フリッカ測定モード表示(数値)
例	:DISPlay:MODE NUMERIC :DISPlay:MODE? -> :DISPlay:MODE NUMERIC
解説	・ {BAR VECTor NBAR WBAR BTREnd}は、高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ選択可能です。 ・ {MATH NMATH FFT NFFT WFFT IECHarm}は、高度演算機能(オプション, /G6)搭載時のみ選択可能です。 ・ {CBCycle}は、Cycle by Cycle機能(オプション, /CC)搭載時のみ選択可能です。 ・ {FLICKer}は、フリッカ測定機能(オプション, /FL)搭載時のみ選択可能です。

: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	:DISPlay:NUMeric? -> (「:DISPlay:NUMeric:NORMAL?」の応答と同じ)

:DISPlay:NUMeric:NORMAL?

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:NORMAL?
例	・ 数値表示の方式(:DISPlay:NUMeric[:NORMAL]:FORMAT)が「VAL4(4値表示)」のときの例 :DISPlay:NUMeric:NORMAL? -> :DISPlay:NUMeric:NORMAL:FORMAT VAL4;VAL4:ITEM1 U,1,TOTAL;ITEM2 I,1,TOTAL;ITEM3 P,1,TOTAL;...(中略)...;ITEM35 ETA3;ITEM36 ETA4;CURSOR 1 ・ 数値表示の方式(:DISPlay:NUMeric[:NORMAL]:FORMAT)が「ALL(全表示)」のときの例 :DISPlay:NUMeric:NORMAL? -> :DISPlay:NUMeric:NORMAL:FORMAT ALL;ALL:CURSOR U
解説	数値表示の方式(:DISPlay:NUMeric[:NORMAL]:FORMAT)に該当するすべての設定値を返します。

:DISPlay:NUMeric[:NORMAL]:ALL?

機能	数値表示(全表示)に関するすべての設定値を問い合わせます。
構文	:DISPlay:NUMeric[:NORMAL]:ALL?
例	:DISPlay:NUMeric[:NORMAL]:ALL? -> :DISPlay:NUMeric[:NORMAL]:ALL:CURSOR U

:DISPlay:NUMeric[:NORMAL]:ALL:CURSOR

機能	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]:ALL:CURSOR {<Function>} :DISPlay:NUMeric[:NORMAL]:ALL:CURSOR?
例	<Function>={U I P S Q ...}(ファンクション選択肢一覧(1), 6-41ページ参照) :DISPlay:NUMeric[:NORMAL]:ALL:CURSOR U :DISPlay:NUMeric[:NORMAL]:ALL:CURSOR? -> :DISPlay:NUMeric[:NORMAL]:ALL:CURSOR U
解説	カーソル位置はファンクション名で指定します。

:DISPlay:NUMeric[:NORMal]:ALL:ORDER

機能	数値表示(全表示)の高調波測定ファンクション表示ページにおける表示次数を設定/問い合わせします。
構文	<pre>:DISPlay:NUMeric[:NORMal]:ALL: ORDER {<Order>} :DISPlay:NUMeric[:NORMal]:ALL: ORDER? <Order>={TOTal DC <NRf>}(<NRf>=1~100)</pre>
例	<pre>:DISPlay:NUMeric[:NORMal]:ALL:ORDER 1 :DISPlay:NUMeric[:NORMal]:ALL:ORDER? -> :DISPlay:NUMeric[:NORMal]:ALL:ORDER 1</pre>
解説	- 高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ有効です。 - 数値表示(全表示)における表示ページ番号(:DISPlay:NUMeric[:NORMal]:ALL:PAGE)が6~7のときに有効な設定です。

:DISPlay:NUMeric[:NORMal]:ALL:PAGE

機能	数値表示(全表示)の表示ページ番号を設定/問い合わせします。
構文	<pre>:DISPlay:NUMeric[:NORMal]:ALL: PAGE {<NRf>} :DISPlay:NUMeric[:NORMal]:ALL:PAGE? <NRf>=1~5(ページ番号) <NRf>=1~9(高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時)</pre>
例	<pre>:DISPlay:NUMeric[:NORMal]:ALL:PAGE 1 :DISPlay:NUMeric[:NORMal]:ALL:PAGE? -> :DISPlay:NUMeric[:NORMal]:ALL:PAGE 1</pre>
解説	ページ番号を設定すると、カーソル位置は設定したページの先頭に移動します。

:DISPlay:NUMeric[:NORMal]:FORMat

機能	数値表示の方式を設定/問い合わせします。
構文	<pre>:DISPlay:NUMeric[:NORMal]: FORMat {VAL4 VAL8 VAL16 ALL SINGle DUAL} :DISPlay:NUMeric[:NORMal]:FORMat? :DISPlay:NUMeric[:NORMal]:FORMat VAL4 :DISPlay:NUMeric[:NORMal]:FORMat? -> :DISPlay:NUMeric[:NORMal]:FORMat VAL4</pre>
例	<pre>:DISPlay:NUMeric[:NORMal]:FORMat VAL4 :DISPlay:NUMeric[:NORMal]:FORMat? -> :DISPlay:NUMeric[:NORMal]:FORMat VAL4</pre>
解説	- 表示される数値データの内容は以下のようになります。 {VAL4 VAL8 VAL16}=数値表示項目を、項目番号順に表示します。(数字は1つの画面(ページ)に表示される項目数を表します) - ALL=全ファンクションをエレメントごとに並べて表示します。 - SINGle=1つのリスト表示項目を、EVEN/ODDに分けてリスト表示します。 - DUAL=2つのリスト表示項目を、次数順に並べてリスト表示します。 {SINGle DUAL}は、高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ選択可能です。

:DISPlay:NUMeric[:NORMal]:LIST?

機能	数値表示(リスト表示)に関するすべての設定値を問い合わせます。
構文	<pre>:DISPlay:NUMeric[:NORMal]:LIST? :DISPlay:NUMeric[:NORMal]:LIST?</pre>
例	<pre>:DISPlay:NUMeric[:NORMal]:LIST? -> :DISPlay:NUMeric[:NORMal]:LIST: ITEM1 U,1;ITEM2 I,1;CURSOR ORDER; HEADER 1;ORDER 1</pre>
解説	高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ有効です。

:DISPlay:NUMeric[:NORMal]:LIST:CURSor

機能	数値表示(リスト表示)のカーソル位置を設定/問い合わせします。
構文	<pre>:DISPlay:NUMeric[:NORMal]:LIST: CURSor {HEADer ORDER} :DISPlay:NUMeric[:NORMal]:LIST: CURSor?</pre>
例	<pre>:DISPlay:NUMeric[:NORMal]:LIST: CURSOR ORDER :DISPlay:NUMeric[:NORMal]:LIST: CURSOR? -> :DISPlay:NUMeric[:NORMal]:LIST:CURSOR ORDER</pre>
解説	- HEADer=ヘッダ部(高調波全体に関するデータ、画面左側)にカーソルが移動します。 - ORDER=データ部(各高調波の数値データ、画面右側)にカーソルが移動します。

6.7 DISPlayグループ

:DISPlay:NUMeric[:NORMal]:LIST:HEADer

機能	数値表示(リスト表示)のヘッダ部のカーソル位置を設定/問い合わせします。
構文	<pre>:DISPlay:NUMeric[:NORMal]:LIST: HEADer {<NRf>} :DISPlay:NUMeric[:NORMal]:LIST: HEADer? <NRf>=1~98</pre>
例	<pre>:DISPlay:NUMeric:NORMal:LIST: HEADer 1 :DISPlay:NUMeric:NORMal:LIST: HEADer? -> :DISPlay:NUMeric:NORMal: LIST:HEADer 1</pre>
解説	・高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ有効です。 ・数値表示(リスト表示)におけるカーソル位置(:DISPlay:NUMeric[:NORMal]:LIST:CURSor)が「HEADer」のときに有効な設定です。

:DISPlay:NUMeric[:NORMal]:LIST:ITEM<x>

機能	数値表示(リスト表示)の表示項目(ファンクション・エレメント)を設定/問い合わせします。
構文	<pre>:DISPlay:NUMeric[:NORMal]:LIST: ITEM<x> {<Function>,<Element>} :DISPlay:NUMeric[:NORMal]:LIST: ITEM<x>? <x>=1, 2(項目番号) <Function>={U I P S Q LAMBda ...} (ファンクション選択肢一覧(2), 6-43ページ参照) <Element>={<NRf> SIGMA SIGMB}(<NRf>=1~4)</pre>
例	<pre>:DISPlay:NUMeric:NORMal:LIST: ITEM1 U,1 :DISPlay:NUMeric:NORMal:LIST:ITEM1? -> :DISPlay:NUMeric:NORMal:LIST: ITEM1 U,1</pre>
解説	高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ有効です。

:DISPlay:NUMeric[:NORMal]:LIST:ORDER

機能	数値表示(リスト表示)のデータ部の次数カーソル位置を設定/問い合わせします。
構文	<pre>:DISPlay:NUMeric[:NORMal]:LIST: ORDER {<NRf>} :DISPlay:NUMeric[:NORMal]:LIST: ORDER? <NRf>=1~100(次数)</pre>
例	<pre>:DISPlay:NUMeric:NORMal:LIST: ORDER 1 :DISPlay:NUMeric:NORMal:LIST:ORDER? -> :DISPlay:NUMeric:NORMal:LIST: ORDER 1</pre>
解説	・高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ有効です。 ・数値表示(リスト表示)におけるカーソル位置(:DISPlay:NUMeric[:NORMal]:LIST:CURSor)が「ORDER」のときに有効な設定です。

:DISPlay:NUMeric[:NORMal]:{VAL4|VAL8|VAL16}?

機能	数値表示({4値 8値 16値}表示)に関するすべての設定値を問い合わせます。
構文	<pre>:DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}?</pre>
例	<pre>:DISPlay:NUMeric:NORMal:VAL4? -> :DISPlay:NUMeric:NORMal:VAL4: ITEM1 U,1,TOTAL;ITEM2 I,1,TOTAL; ITEM3 P,1,TOTAL; ITEM4 LAMBDA,1,TOTAL; ITEM5 U,2,TOTAL;ITEM6 I,2,TOTAL; ITEM7 P,2,TOTAL; ITEM8 LAMBDA,2,TOTAL; ITEM9 U,3,TOTAL;ITEM10 I,3,TOTAL; ITEM11 P,3,TOTAL; ITEM12 LAMBDA,3,TOTAL; ITEM13 U,4,TOTAL;ITEM14 I,4,TOTAL; ITEM15 P,4,TOTAL; ITEM16 LAMBDA,4,TOTAL; ITEM17 U,SIGMA,TOTAL; ITEM18 I,SIGMA,TOTAL; ITEM19 P,SIGMA,TOTAL; ITEM20 LAMBDA,SIGMA,TOTAL; ITEM21 U,SIGMB,TOTAL; ITEM22 I,SIGMB,TOTAL; ITEM23 P,SIGMB,TOTAL; ITEM24 LAMBDA,SIGMB,TOTAL; ITEM25 WH,1;ITEM26 WH,2; ITEM27 WH,3;ITEM28 WH,4; ITEM29 WH,SIGMA;ITEM30 WH,SIGMB; ITEM31 F1;ITEM32 F2;ITEM33 ETA1; ITEM34 ETA2;ITEM35 ETA3; ITEM36 ETA4;CURSOR 1</pre>

:DISPlay:NUMeric[:NORMal]:{VAL4|VAL8|VAL16}:CURSor

機能	数値表示({4値 8値 16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[:NORMal]:{VAL4 VAL8 VAL16}:CURSor {<Nrf>} :DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}:CURSor? <Nrf>=1~36(項目番号, VAL4のとき) <Nrf>=1~72(項目番号, VAL8のとき) <Nrf>=1~144(項目番号, VAL16のとき)
例	:DISPlay:NUMeric:NORMAL:VAL4: CURSOR 1 :DISPlay:NUMeric:NORMAL:VAL4: CURSOR? -> :DISPlay:NUMeric:NORMAL: VAL4:CURSOR 1
解説	カーソル位置は項目番号で指定します。

:DISPlay:NUMeric[:NORMal]:{VAL4|VAL8|VAL16}:ITEM<x>

機能	数値表示({4値 8値 16値}表示)の表示項目(ファンクション・エレメント・次数)を設定/問い合わせします。
構文	:DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}:ITEM<x> {NONE <Function>,<Element>[,<Order>]} :DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}:ITEM<x>? <x>=1~36(項目番号, VAL4のとき) <x>=1~72(項目番号, VAL8のとき) <x>=1~144(項目番号, VAL16のとき) NONE=表示項目なし <Function>={U I P S Q ...}(ファンクション選択肢一覧(1), 6-41ページ参照) <Element>={<Nrf> SIGMA SIGMB}(<Nrf>=1~4) <Order>={TOTAL DC <Nrf>}(<Nrf>=1~100)
例	:DISPlay:NUMeric:NORMAL:VAL4: ITEM1 U,1,TOTAL :DISPlay:NUMeric:NORMAL:VAL4:ITEM1? -> :DISPlay:NUMeric:NORMAL:VAL4: ITEM1 U,1,TOTAL
解説	・<Element>を省略したときは, エレメント1が設定されます。 ・<Order>を省略したときは, TOTALが設定されます。 ・<Element>または<Order>が不要なファンクションの応答は, <Element>または<Order>が省略されます。

:DISPlay:NUMeric[:NORMal]:{VAL4|VAL8|VAL16}:PAGE

機能	数値表示({4値 8値 16値}表示)の表示ページ番号を設定/問い合わせします。
構文	:DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}:PAGE {<Nrf>} :DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}:PAGE? <Nrf>=1~9(ページ番号)
例	:DISPlay:NUMeric:NORMAL:VAL4:PAGE 1 :DISPlay:NUMeric:NORMAL:VAL4:PAGE? -> :DISPlay:NUMeric:NORMAL:VAL4: PAGE 1
解説	ページ番号を設定すると, カーソル位置は設定したページの先頭に移動します。

:DISPlay:NUMeric[:NORMal]:{VAL4|VAL8|VAL16}:PRESet

機能	数値表示({4値 8値 16値}表示)の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]:{VAL4 VAL8 VAL16}:PRESet {<Nrf>} <Nrf>=1~4
例	:DISPlay:NUMeric:NORMAL:VAL4: 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ITEM設定メニューのリセット実行(Reset Items 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, ユーザーズマニュアルIM760301-01を参照してください。

: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: FORMAT SINGLE;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;TDIV 0,0,3;ITEM1: FUNCTION U,1,TOTAL;SCALING: MODE AUTO; VALUE 100.0E+00,-100.0E+00;; DISPlay:TREnd:ITEM2: FUNCTION I,1,TOTAL;SCALING: MODE AUTO; VALUE 100.0E+00,-100.0E+00;... (中略)... :DISPlay:TREnd:NORMAL: ITEM16:FUNCTION AH,1;SCALING: MODE AUTO; VALUE 100.0E+00,-100.0E+00

6.7 DISPlayグループ

: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

:DISPlay:TREnd:CLear

機能 トレンドをクリアします。

構文 :DISPlay:TREnd:CLear

例 :DISPLAY:TREND:CLEAR

: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

:DISPlay:TREnd:ITEM<x>?

機能 各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:ITEM<x>?

<x>=1~16(項目番号)

例 :DISPLAY:TREND:ITEM1? -> :DISPLAY:

TREND:ITEM1:FUNCTION U,1,TOTAL;

SCALING:MODE AUTO;

VALUE 100.0E+00,-100.0E+00

:DISPlay:TREnd:ITEM<x>[:FUNCTION]

機能 トレンド項目(ファンクション・エレメント・次数)を設定/問い合わせします。

構文 :DISPlay:TREnd:ITEM<x>

[:FUNCTION] {NONE|<Function>,
<Element>[,<Order>]}

:DISPlay:TREnd:ITEM<x>:FUNCTION?
<x>=1~16(項目番号)

NONE=表示項目なし

<Function>={U|I|P|S|Q|...}(ファンク
ション選択肢一覧(1), 6-41ページ参照)

<Element>={<NRf>|SIGMA|SIGMB}(<NRf>
=1~4)

<Order>={TOTal|DC|<NRf>}(<NRf>=1~
100)

例 :DISPLAY:TREND:ITEM1:

FUNCTION U,1,TOTAL

:DISPLAY:TREND:ITEM1:FUNCTION? ->

:DISPLAY:TREND:ITEM1:

FUNCTION U,1,TOTAL

解説
・ <Element>を省略したときは、エレメント1
が設定されます。
・ <Order>を省略したときは、TOTalが設定さ
れます。
・ <Element>または<Order>が不要なファンク
ションの応答は、<Element>または<Order>
が省略されます。

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

機能 トレンドの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 :DISPlay:TREnd:ITEM<x>:SCALing?

<x>=1~16(項目番号)

例 :DISPLAY:TREND:ITEM1:SCALING? ->

:DISPLAY:TREND:ITEM1:SCALING:

MODE AUTO;

VALUE 100.0E+00,-100.0E+00

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

機能 トレンドのスケーリング方式を設定/問い合
わせます。

構文 :DISPlay:TREnd:ITEM<x>:SCALing:

MODE {AUTO|MANual}

:DISPlay:TREnd:ITEM<x>:SCALing:
MODE?

<x>=1~16(項目番号)

例 :DISPLAY:TREND:ITEM1:SCALING:

MODE AUTO

:DISPLAY:TREND:ITEM1:SCALING:MODE?

-> :DISPLAY:TREND:ITEM1:SCALING:

MODE AUTO

:DISPlay:TREND:ITEM<x>:SCALing:VALue

機能 トレンドのマニュアルスケーリング上下限値を設定/問い合わせします。

構文 :DISPlay:TREND:ITEM<x>:SCALing:VALue {<NRf>,<NRf>}

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

<x>=1~16(項目番号)

<NRf>=-9.999E+30~9.999E+30

例 :DISPLAY:TREND:ITEM1:SCALING:

VALUE 100,-100

:DISPLAY:TREND:ITEM1:SCALING:VALUE?

-> :DISPLAY:TREND:ITEM1:SCALING:

VALUE 100.0E+00,-100.0E+00

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:ITEM<x>:SCALing:MODE)が「MANual」のときに有効な設定です。

:DISPlay:TREND:TDIV

機能 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。

: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

:DISPlay:VECTor?

機能 ベクトル表示に関するすべての設定値を問い合わせます。

構文 :DISPlay:VECTor?

例 :DISPLAY:VECTOR? -> :DISPLAY:

VECTOR:OBJECT SIGMA;NUMERIC 1;

UMAG 1.000;IMAG 1.000

解説 高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ有効です。

:DISPlay:VECTor:NUMeric

機能 ベクトル表示の数値データ表示のON/OFFを設定/問い合わせします。

構文 :DISPlay:VECTor:NUMeric {<Boolean>}

:DISPlay:VECTor:NUMeric?

例 :DISPLAY:VECTOR:NUMERIC ON

:DISPLAY:VECTOR:NUMERIC? ->

:DISPLAY:VECTOR:NUMERIC 1

解説 高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ有効です。

:DISPlay:VECTor:OBject

機能 ベクトル表示の表示対象とする結線ユニットを設定/問い合わせします。

構文 :DISPlay:VECTor:OBject {SIGMA|SIGMB}

:DISPlay:VECTor:OBject?

例 :DISPLAY:VECTOR:OBJECT SIGMA

:DISPLAY:VECTOR:OBJECT? ->

:DISPLAY:VECTOR:OBJECT SIGMA

解説 ・ 高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ有効です。

・ SIGMBは, 4エレメントモデルのみ選択可能です。

:DISPlay:VECTor:{UMAG|IMAG}

機能 ベクトル表示の{電圧|電流}表示のズーム率を設定/問い合わせします。

構文 :DISPlay:VECTor:{UMAG|IMAG} {<NRf>}

:DISPlay:VECTor:{UMAG|IMAG}?

<NRf>=0.100~100.000

例 :DISPLAY:VECTOR:UMAG 1

:DISPLAY:VECTOR:UMAG? ->

:DISPLAY:VECTOR:UMAG 1.000

解説 高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ有効です。

: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:

FORMAT SINGLE;U1 1;U2 1;U3 1;U4 1;

I1 1;I2 1;I3 1;I4 1;TDIV 5.0E-03;

TRIGGER:MODE AUTO;SOURCE U1;

SLOPE RISE;LEVEL 0.0;:DISPLAY:WAVE:

INTERPOLATE LINE;GRATICULE GRID;

SVALUE 1;TLABEL 0;MAPPING:

MODE AUTO;:DISPLAY:WAVE: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;:

DISPLAY:WAVE: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

6.7 DISPlayグループ

: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

: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}
:DISPlay:WAVE:FORMat?

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

: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}
:DISPlay:WAVE:GRATicule?

例 :DISPLAY:WAVE:GRATICULE GRID
:DISPLAY:WAVE:GRATICULE? ->
:DISPLAY:WAVE:GRATICULE GRID

: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}
:DISPlay:WAVE:INTERpolate?

例 :DISPLAY:WAVE:INTERPOLATE LINE
:DISPLAY:WAVE:INTERPOLATE? ->
:DISPLAY:WAVE:INTERPOLATE LINE

: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

: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}
:DISPlay:WAVE:MAPPING:MODE?

例 :DISPLAY:WAVE:MAPPING:MODE AUTO
:DISPLAY:WAVE:MAPPING:MODE? ->
:DISPLAY:WAVE:MAPPING:MODE AUTO

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

機能 分割フォーマットへの{電圧|電流|回転速度|トルク|波形演算}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>|SPEed|TORQue|MATH<x>} {<Nrf>}
:DISPlay:WAVE:MAPPING:{U<x>|I<x>|SPEed|TORQue|MATH<x>}?
U<x>, I<x>の<x>=1~4(エレメント)
MATH<x>の<x>=1, 2(MATH)
<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。
・ {SPEed|TORQue}は、モータバージョン(-MV)のみ有効です。
・ MATH<x>は、高度演算機能(オプション、/G6)搭載時のみ有効です。

:DISPlay:WAVE:POSition?

機能 波形の垂直ポジション(中心位置のレベル)に関するすべての設定値を問い合わせます。

構文 :DISPlay:WAVE:POSition?

例 :DISPLAY:WAVE:POSITION? ->
:DISPLAY:WAVE: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

:DISPlay:WAVE:POSition:{UALL|IALL}

機能 すべてのエレメントの{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 :DISPlay:WAVE:POSition:{UALL|IALL} {<Nrf>}

<Nrf>=-130.000~130.000(%)

例 :DISPLAY:WAVE:POSITION:UALL 0

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

機能 各エレメントの{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 :DISPlay:WAVE:POSition:{U<x>|I<x>} {<Nrf>}
:DISPlay:WAVE:POSition:{U<x>|I<x>}?
<x>=1~4(エレメント)
<Nrf>=-130.000~130.000(%)

例 :DISPLAY:WAVE:POSITION:U1 0
:DISPLAY:WAVE:POSITION:U1? ->
:DISPLAY:WAVE:POSITION:U1 0.000

: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

:DISPlay:WAVE:TDIV

機能 波形のTime/div値を設定/問い合わせします。

構文 :DISPlay:WAVE:TDIV {<時間>}
:DISPlay:WAVE:TDIV?
<時間>=0.5, 1, 2, 5, 10, 20, 50, 100,
200, 500(ms), 1, 2(s)

例 :DISPlay:WAVE:TDIV 5MS
:DISPlay:WAVE:TDIV? ->
:DISPlay:WAVE:TDIV 5.0E-03

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。

: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

:DISPlay:WAVE: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 :DISPlay:WAVE:TRIGger?

例 :DISPlay:WAVE:TRIGger? ->
:DISPlay:WAVE:TRIGGER:MODE AUTO;
SOURCE U1;SLOPE RISE;LEVEL 0.0

:DISPlay:WAVE:TRIGger:LEVel

機能 トリガレベルを設定/問い合わせします。

構文 :DISPlay:WAVE:TRIGger:LEVel {<NRF>}
:DISPlay:WAVE:TRIGger:LEVel?
<NRF>=-100.0~100.0(%)(設定分解能は0.1(%))

例 :DISPlay:WAVE:TRIGGER:LEVEL 0
:DISPlay:WAVE:TRIGGER:LEVEL? ->
:DISPlay:WAVE: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。

:DISPlay:WAVE:TRIGger:MODE

機能 トリガモードを設定/問い合わせします。

構文 :DISPlay:WAVE:TRIGger:MODE {AUTO|NORMAL}

:DISPlay:WAVE:TRIGger:MODE?

例 :DISPlay:WAVE:TRIGGER:MODE AUTO
:DISPlay:WAVE:TRIGGER:MODE? ->
:DISPlay:WAVE:TRIGGER:MODE AUTO

:DISPlay:WAVE:TRIGger:SLOPe

機能 トリガスロープを設定/問い合わせします。

構文 :DISPlay:WAVE:TRIGger:SLOPe {RISE|FALL|BOTH}

:DISPlay:WAVE:TRIGger:SLOPe?

例 :DISPlay:WAVE:TRIGGER:SLOPE RISE
:DISPlay:WAVE:TRIGGER:SLOPE? ->
:DISPlay:WAVE:TRIGGER:SLOPE RISE

:DISPlay:WAVE:TRIGger:SOURce

機能 トリガソースを設定/問い合わせします。

構文 :DISPlay:WAVE:TRIGger:SOURce {U<x>|I<x>|EXTERNAL}

:DISPlay:WAVE:TRIGger:SOURce?

<x>=1~4(エレメント)

EXTERNAL=外部トリガ入力(Ext Clk)

例 :DISPlay:WAVE:TRIGGER:SOURCE U1
:DISPlay:WAVE:TRIGGER:SOURCE? ->
:DISPlay:WAVE:TRIGGER:SOURCE U1

:DISPlay:WAVE:{U<x>|I<x>|SPEed|TORQue|MATH<x>}

機能 {電圧|電流|回転速度|トルク|波形演算}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>|SPEed|TORQue|MATH<x>} {<Boolean>}

:DISPlay:WAVE:{U<x>|I<x>|SPEed|TORQue|MATH<x>}?
U<x>, I<x>の<x>=1~4(エレメント)

MATH<x>の<x>=1, 2(MATH)

例 :DISPlay:WAVE:U1 ON

:DISPlay:WAVE:U1? ->

:DISPlay:WAVE:U1 1

解説 ・ {SPEed|TORQue}は、モータバージョン(-MV)のみ有効です。
・ MATH<x>は、高度演算機能(オプション、/G6)搭載時のみ有効です。

:DISPlay:WAVE: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 :DISPlay:WAVE:VZoom?

例 :DISPlay:WAVE:VZOOM? ->

:DISPlay:WAVE: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

6.7 DISPlayグループ

:DISPlay:WAVE:VZoom:{UALL|IALL}

機能 すべてのエレメントの{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 :DISPlay:WAVE:VZoom:{UALL|IALL}
{<NRf>}

<NRf>=0.1~100(ユーザーズマニュアル
IM760301-01参照)

例 :DISPLAY:WAVE:VZOOM:UALL 1

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

機能 各エレメントの{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 :DISPlay:WAVE:VZoom:{U<x>|I<x>}
{<NRf>}

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

<x>=1~4(エレメント)

<NRf>=0.1~100(ユーザーズマニュアル
IM760301-01参照)

例 :DISPLAY:WAVE:VZOOM:U1 1
:DISPLAY:WAVE:VZOOM:U1? ->
:DISPLAY:WAVE:VZOOM:U1 1.00

※ファンクション選択肢(<Function>)一覧

(1) 数値データのファンクション

対象コマンド*

:AOUPut[:NORMal]:CHANnel<x>

:DISPlay:NUMeric[:NORMal]:{VAL4|VAL8|VAL16}:ITEM<x>

:DISPlay:TREnd:ITEM<x>[:FUNction]

:NUMeric[:NORMal]:ITEM<x>

コマンドで使用する ファンクション名	メニューで使用する ファンクション名 (数値表示ヘッダ名)	エレメント ○必要 ×不要	次数 ○必要 ×不要
U	U (Urms/Umn/Udc/Urmn)	○	○
I	I (Irms/Imn/Idc/Irmn)	○	○
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	CfI	○	×
PC	Pc	○	×
TIME	Time	○	×
WH	WP	○	×
WHP	WP+	○	×
WHM	WP-	○	×
AH	q	○	×
AHP	q+	○	×
AHM	q-	○	×
WS	WS	○	×
WQ	WQ	○	×
ETA1	η1	×	×
ETA2	η2	×	×
ETA3	η3	×	×
ETA4	η4	×	×
F1	F1	×	×
F2	F2	×	×
F3	F3	×	×
F4	F4	×	×
F5	F5	×	×
F6	F6	×	×
F7	F7	×	×
F8	F8	×	×
F9	F9	×	×
F10	F10	×	×
F11	F11	×	×
F12	F12	×	×
F13	F13	×	×
F14	F14	×	×
F15	F15	×	×
F16	F16	×	×
F17	F17	×	×
F18	F18	×	×
F19	F19	×	×
F20	F20	×	×

高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)が必要なファンクション

PHIU	ϕU	○	○
PHII	ϕI	○	○
Z	Z	○	○
RS	Rs	○	○
XS	Xs	○	○
RP	Rp	○	○
XP	Xp	○	○
UHDF	Uhdf	○	○
IHDF	Ihdf	○	○
PHDF	Phdf	○	○
UTHD	Uthd	○	×
ITHD	Ithd	○	×
PTHD	Pthd	○	×
UTHF	Uthf	○	×
ITHF	Ithf	○	×
UTIF	Utif	○	×
ITIF	Itif	○	×
HVF	hvf	○	×
HCF	hcf	○	×
PHI_U1U2	ϕU_i-U_j	○	×
PHI_U1U3	ϕU_i-U_k	○	×
PHI_U1I1	ϕU_i-I_i	○	×
PHI_U1I2	ϕU_i-I_j	○	×
PHI_U1I3	ϕU_i-I_k	○	×

デルタ演算機能(オプション, /DT)が必要なファンクション

DELTA1	$\Delta F1$	×	×
DELTA2	$\Delta F2$	×	×
DELTA3	$\Delta F3$	×	×
DELTA4	$\Delta F4$	×	×

モータバージョン(-MV)が必要なファンクション

SPEEd	Speed	×	×
TORQue	Torque	×	×
SYNCsp	SyncSp	×	×
SLIP	Slip	×	×
PM	Pm	×	×

次のコマンドでは、さらに下記のファンクションも使用可能

:DISPlay:NUMeric[:NORMal]:ALL:CURSor
:FILE:SAVE:NUMeric[:NORMal]:<Function>
:HCOpy:PRINter:LIST[:NORMal]:<Function>
:STORe:NUMeric[:NORMal]:<Function>

(高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)が必要なファンクション)

UK	U(k)
IK	I(k)
PK	P(k)
SK	S(k)
QK	Q(k)
LAMBDK	$\lambda(k)$
PHIK	$\phi(k)$
PHIUk	$\phi U(k)$
PHIIk	$\phi I(k)$
Zk	Z(k)
RSk	Rs(k)
XSk	Xs(k)
RPk	Rp(k)
XPk	Xp(k)

Note

- ・上記の選択肢一覧でエレメントの指定が不要なファンクションについては、エレメントを指定するパラメータ(<Element>)を持つコマンドでは、「1」を設定するか、または省略してください。
- ・同様に、次数の指定が不要なファンクションについても、次数を指定するパラメータ(<Order>)を持つコマンドでは、「TOTal」を設定するか、または省略してください。

(2) 数値リストデータのファンクション(高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)が必要)

対象コマンド

```
:DISPlay:BAR:ITEM<x>
:DISPlay:NUMeric[:NORMal]:LIST:ITEM<x>
```

コマンドで使用する ファンクション名	メニューで使用している ファンクション名 (数値表示ヘッダ名)
U	U
I	I
P	P
S	S
Q	Q
LAMBda	λ
PHI	ϕ
PHIU	ϕU
PHII	ϕI
Z	Z
RS	Rs
XS	Xs
RP	Rp
XP	Xp

次のコマンドでは、さらに下記のファンクションも使用可能

:NUMeric:LIST:ITEM<x>	
UHDF	Uhdf
IHDF	Ihdf
PHDF	Phdf

6.8 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|PNG|JPEG}

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 PNG JPEG} {<Filename>}
例	:FILE:DELeTe:IMAGe:TIFF "IMAG1"
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 {PCCard[,<Nrf>] NETWork USB,<Nrf>[,<Nrf>][,<Nrf>]} PCCard=PCカードドライブ <Nrf>=パーティション(0~3) NETWork=ネットワークドライブ USB=USBメモリドライブ 最初の<Nrf>=ID番号(アドレス) 2番目の<Nrf>=パーティション(0~3)または LUN(論理ユニット番号:0~3) 3番目の<Nrf>=LUNを指定したときのパー ティション(0~3)

例 :FILE:DRIVE PCCARD

- 解説
- ・パーティションを設定していないドライブの場合、パーティションの<Nrf>は省略してください。
 - ・「NETWork」はイーサネット(オプション、/C7)搭載時に使用できます。
 - ・「USB」はUSBポート(周辺機器)(オプション、/C5)搭載時に使用できます。
 - ・「USB」の場合の2, 3番目の<Nrf>は、パーティションまたはLUNで区切られていないときは省略可能です。

:FILE:FORMat:EXECute

機能	PCカードの初期化を実行します。
構文	:FILE:FORMat:EXECute
例	:FILE:FORMat:EXECUTE

: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	ファイルの読み込みを中止します。
構文	:FILE:LOAD:ABORt
例	:FILE:LOAD:ABORT

:FILE:LOAD:SETup

機能	設定情報ファイルの読み込みを実行します。
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? -> "PC_Card\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:
 TYPE BINARY;;FILE:SAVE:NUMERIC:
 TYPE ASCII;NORMAL:ELEMENT1 1;
 ELEMENT2 0;ELEMENT3 0;ELEMENT4 0;
 SIGMA 0;SIGMB 0;U 1;I 1;P 1;S 1;
 Q 1;LAMBDA 1;PHI 1;FU 1;FI 1;
 UPPEAK 0;UMPEAK 0;IPPEAK 0;
 IMPEAK 0;CFU 0;CFI 0;PC 0;TIME 0;
 WH 0;WHP 0;WHM 0;AH 0;AHP 0;AHM 0;
 WS 0;WQ 0;ETA1 0;ETA2 0;ETA3 0;
 ETA4 0;F1 0;F2 0;F3 0;F4 0;F5 0;
 F6 0;F7 0;F8 0;F9 0;F10 0;F11 0;
 F12 0;F13 0;F14 0;F15 0;F16 0;
 F17 0;F18 0;F19 0;F20 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:ACQuisition?

機能 波形サンプリング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:ACQuisition?**

例 **:FILE:SAVE:ACQUISITION? ->
 :FILE:SAVE:ACQUISITION:TYPE FLOAT;
 TRACE U1**

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

:FILE:SAVE:ACQuisition[:EXECute]

機能 波形サンプリングデータをファイルに保存します。

構文 **:FILE:SAVE:ACQuisition
 [:EXECute] {<Filename>}**

例 **:FILE:SAVE:ACQUISITION:
 EXECUTE "ACQ1"**

解説

- ・高度演算機能(オプション, /G6)搭載時のみ有効です。
- ・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:ACQuisition:TRACe

機能 ファイル保存の対象とする波形サンプリングデータを設定/問い合わせします。

構文 **:FILE:SAVE:ACQuisition:TRACe {U<x>|
 I<x>|SPEEd|TORQue|MATH<x>|FFT<x>}
 :FILE:SAVE:ACQuisition:TRACe?
 U<x>, I<x>の<x>=1~4(エレメント)
 MATH<x>の<x>=1, 2(MATH)
 FFT<x>の<x>=1, 2(FFT)**

例 **:FILE:SAVE:ACQUISITION:TRACE U1
 :FILE:SAVE:ACQUISITION:TRACE? ->
 :FILE:SAVE:ACQUISITION:TRACE U1**

解説

- ・高度演算機能(オプション, /G6)搭載時のみ有効です。
- ・保存する波形サンプリングデータの形式(:FILE:SAVE:ACQuisition:TYPE)が「FLOat」のときに有効な設定です。
- ・{BINary|AScii}のときは、波形表示がONになっている波形サンプリングデータがすべて対象となります。
- ・{SPEEd|TORQue}は、モータバージョン(-MV)のみ有効です。

:FILE:SAVE:ACQuisition:TYPE

機能 保存する波形サンプリングデータの形式を設定/問い合わせします。

構文 **:FILE:SAVE:ACQuisition:
 TYPE {BINary|AScii|FLOat}
 :FILE:SAVE:ACQuisition:TYPE?
 :FILE:SAVE:ACQUISITION:TYPE FLOAT
 :FILE:SAVE:ACQUISITION:TYPE? ->
 :FILE:SAVE:ACQUISITION:TYPE FLOAT**

解説 高度演算機能(オプション, /G6)搭載時のみ有効です。

: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>}**

例 **:FILE:SAVE:ANAMing?
 :FILE:SAVE:ANAMING ON
 :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>=25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;U 1;I 1;P 1;S 1;Q 1;
LAMBDA 1;PHI 1;FU 1;FI 1;UPPEAK 0;
UMPEAK 0;IPPEAK 0;IMPEAK 0;CFU 0;
CFI 0;PC 0;TIME 0;WH 0;WHP 0;WHM 0;
AH 0;AHP 0;AHM 0;WS 0;WQ 0;ETA1 0;
ETA2 0;ETA3 0;ETA4 0;F1 0;F2 0;
F3 0;F4 0;F5 0;F6 0;F7 0;F8 0;F9 0;
F10 0;F11 0;F12 0;F13 0;F14 0;
F15 0;F16 0;F17 0;F18 0;F19 0;F20 0

:FILE:SAVE:NUMeric:CBCycle?

機能 Cycle by Cycle測定の数値データのファイル保存項目に関するすべての設定値を問い合わせます。

構文 **:FILE:SAVE:NUMeric:CBCycle?**

例 **:FILE:SAVE:NUMeric:CBCycle? ->**

:FILE:SAVE:NUMeric:CBCycle
:ELEMENT1 1;ELEMENT2 0;ELEMENT3 0;
ELEMENT4 0;SIGMA 0;SIGMB 0;FREQ 1;
U 1;I 1;P 1;S 1;Q 1;LAMBDA 1

解説 Cycle by Cycle測定機能(オプション, /CC)搭載時のみ有効です。

:FILE:SAVE:NUMeric:CBCycle:ALL

機能 Cycle by Cycle測定の数値データをファイルに保存するときのすべてのエレメント・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:CBCycle:**

ALL {<Boolean>}

例 **:FILE:SAVE:NUMeric:CBCycle:ALL ON**

解説 Cycle by Cycle測定機能(オプション, /CC)搭載時のみ有効です。

:FILE:SAVE:NUMeric:CBCycle:{ELEMeNt<x>|SIGMA|SIGMB}

機能 Cycle by Cycle測定の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}の出力ON/OFFを設定/問い合わせします。

構文 **:FILE:SAVE:NUMeric:**

CBCycle:{ELEMeNt<x>|SIGMA|SIGMB}

{<Boolean>}

:FILE:SAVE:NUMeric:

CBCycle:{ELEMeNt<x>|SIGMA|SIGMB}?

<x>=1~4

例 **:FILE:SAVE:NUMeric:CBCycle:**

ELEMENT1 ON

:FILE:SAVE:NUMeric:CBCycle:

ELEMENT1? -> :FILE:SAVE:NUMeric:

CBCycle:ELEMENT1 1

解説 ・ Cycle by Cycle測定機能(オプション, /CC)搭載時のみ有効です。

・ 「**:FILE:SAVE:NUMeric:CBCycle:SIGMA**」は、2エレメント以上のモデルで有効です。また、出力をONに設定するためには、あらかじめ結線方式の設定([**INPut**]WIRingコマンド)により結線ユニットΣAが存在する必要があります。

・ 「**:FILE:SAVE:NUMeric:CBCycle:SIGMB**」は、4エレメントモデルのみ有効です。また、出力をONに設定するためには、あらかじめ結線方式の設定([**INPut**]WIRingコマンド)により結線ユニットΣBが存在する必要があります。

:FILE:SAVE:NUMeric:CBCycle:<Function>

機能 Cycle by Cycle測定の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:**

CBCycle:<Function> {<Boolean>}

:FILE:SAVE:NUMeric:

CBCycle:<Function>?

<Function>={**FREQ**|**U**|**I**|**P**|**S**|**Q**|**LAMBda**|**SPEEd**|**TORQue**|**PM**}

例 **:FILE:SAVE:NUMeric:CBCycle:U ON**

:FILE:SAVE:NUMeric:CBCycle:U? ->

:FILE:SAVE:NUMeric:CBCycle:U 1

解説 ・ Cycle by Cycle機能(オプション, /CC)搭載時のみ有効です。

・ {**SPEEd**|**TORQue**|**PM**}は、モータバージョン(-MV)のみ有効です。

: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ラップコマンドです。

:FILE:SAVE:NUMERIC:NORMAL?

機能 数値データのファイル保存項目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;U 1;I 1;
P 1;S 1;Q 1;LAMBDA 1;PHI 1;FU 1;
FI 1;UPPEAK 0;UMPEAK 0;IPPEAK 0;
IMPEAK 0;CFU 0;CFI 0;PC 0;TIME 0;
WH 0;WHP 0;WHM 0;AH 0;AHP 0;AHM 0;
WS 0;WQ 0;ETA1 0;ETA2 0;ETA3 0;
ETA4 0;F1 0;F2 0;F3 0;F4 0;F5 0;
F6 0;F7 0;F8 0;F9 0;F10 0;F11 0;
F12 0;F13 0;F14 0;F15 0;F16 0;
F17 0;F18 0;F19 0;F20 0

:FILE:SAVE:NUMERIC:NORMAL:ALL

機能 数値データをファイルに保存するときのすべてのエレメント・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**

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

機能 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 }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} {<Boolean>}
:FILE:SAVE:NUMERIC:NORMAL:
{ELEMENT<x>|SIGMA|SIGMB}?<x>=1~4

例 **:FILE:SAVE:NUMERIC:NORMAL:**
ELEMENT1 ON
:FILE:SAVE:NUMERIC:NORMAL:ELEMENT1?
-> :FILE:SAVE:NUMERIC:NORMAL:
ELEMENT1 1

解説
 ・「**:FILE:SAVE:NUMERIC:NORMAL:**
SIGMA」は、2エレメント以上のモデルで有効です。また、出力をONに設定するためには、あらかじめ結線方式の設定 (**[:INPUT] WIRING** コマンド)により結線ユニット ΣA が存在する必要があります。
 ・「**:FILE:SAVE:NUMERIC:NORMAL:**
SIGMB」は、4エレメントモデルのみ有効です。また、出力をONに設定するためには、あらかじめ結線方式の設定 (**[:INPUT] WIRING** コマンド)により結線ユニット ΣB が存在する必要があります。

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

機能 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、ユーザーズマニュアルIM760301-01を参照してください。

:FILE:SAVE:NUMERIC:NORMAL:<Function>

機能 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:**
<Function> {<Boolean>}
:FILE:SAVE:NUMERIC:NORMAL:
<Function>?<Function>={U|I|P|S|Q|...}(「DISPLAYグループ」のファンクション選択肢一覧(1)、6-41ページ参照)

例 **:FILE:SAVE:NUMERIC:NORMAL:U ON**
:FILE:SAVE:NUMERIC:NORMAL:U? ->
:FILE:SAVE:NUMERIC:NORMAL:U 1

6.8 FILEグループ

: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}

例 :FILE:SAVE:NUMERIC:TYPE?

:FILE:SAVE:NUMERIC:TYPE ASCII

:FILE:SAVE:NUMERIC:TYPE? ->

:FILE:SAVE:NUMERIC:TYPE ASCII

: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

: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。

:FILE:SAVE:WAVE:TRACe

機能 保存対象の波形を設定/問い合わせします。

構文 :FILE:SAVE:WAVE:TRACe {U<x>|I<x>|SPEed|TORQue|MATH<x>}

:FILE:SAVE:WAVE:TRACe?

U<x>, I<x>の<x>=1~4(エレメント)

MATH<x>の<x>=1, 2 (MATH)

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。
- ・{SPEed|TORQue}は、モータバージョン(-MV)のみ有効です。
- ・MATH<x>は、高度演算機能(オプション、/G6)搭載時のみ有効です。

: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}

例 :FILE:SAVE:WAVE:TYPE?

:FILE:SAVE:WAVE:TYPE BINARY

:FILE:SAVE:WAVE:TYPE? ->

:FILE:SAVE:WAVE:TYPE BINARY

6.9 FLICKerグループ

FLICKerグループは、フリッカ測定に関するグループです。

フロントパネルのITEMの「Flicker Items」メニューおよびFORMの「Flicker Form」メニューと同じ設定、および設定内容の問い合わせができます。

ただし、このグループのコマンドは、フリッカ測定機能(オプション、/FL)搭載時のみ有効です。

:FLICKer?

機能 フリッカ測定に関するすべての設定値を問い合わせます。

構文 :FLICKer?

例 :FLICKer? -> :FLICKer:
MEASUREMENT FLICKer;ELEMENT1 1;
ELEMENT2 0;ELEMENT3 0;
INTERVAL 10,0;COUNT 12;
FREQUENCY 50;UN:MODE AUTO;
VALUE 230.00;;FLICKer:DC:STATE 1;
LIMIT 3.30;;FLICKer:DMAX:STATE 1;
LIMIT 4.00;;FLICKer:DT:STATE 1;
LIMIT 500,3.30;;FLICKer:PST:
STATE 1;LIMIT 1.00;;FLICKer:PLT:
STATE 1;LIMIT 0.65;NVALUE 12;;
FLICKer:DMIN:LIMIT 0.10

:FLICKer:COUNT

機能 短期間フリッカ値Pstの測定回数を設定/問い合わせします。

構文 :FLICKer:COUNT {<Nrf>}

:FLICKer:COUNT?
<Nrf>=1~99(測定回数)

例 :FLICKer:COUNT 12
:FLICKer:COUNT? ->
:FLICKer:COUNT 12

解説 通常のフリッカ測定(:FLICKer:MEASurement FLICKer)のときに有効な設定です。
手動スイッチングdmax測定(:FLICKer:MEASurement DMAX)のときの測定回数は、24(回)固定です。

:FLICKer:DC?

機能 相対定常電圧変化dcに関するすべての設定値を問い合わせます。

構文 :FLICKer:DC?

例 :FLICKer:DC? ->
:FLICKer:DC:STATE 1;LIMIT 3.30

:FLICKer:DC:LIMit

機能 相対定常電圧変化dcの限度値を設定/問い合わせします。

構文 :FLICKer:DC:LIMit {<Nrf>}

:FLICKer:DC:LIMit?
<Nrf>=1.00~99.99(限度値[%])

例 :FLICKer:DC:LIMIT 3.30
:FLICKer:DC:LIMIT? ->
:FLICKer:DC:LIMIT 3.30

:FLICKer:DC[:STATE]

機能 相対定常電圧変化dcの判定ON/OFFを設定/問い合わせします。

構文 :FLICKer:DC[:STATE] {<Boolean>}

:FLICKer:DC:STATE?

例 :FLICKer:DC:STATE ON
:FLICKer:DC:STATE? ->
:FLICKer:DC:STATE 1

:FLICKer:DISPlay?

機能 フリッカ測定表示に関するすべての設定値を問い合わせます。

構文 :FLICKer:DISPlay?

例 :FLICKer:DISPlay? ->
:FLICKer:DISPlay:ELEMENT 1;PERIOD 1

:FLICKer:DISPlay:ELEMent

機能 フリッカ測定表示の表示対象エレメントを設定/問い合わせします。

構文 :FLICKer:DISPlay:ELEMent {<Nrf>}

:FLICKer:DISPlay:ELEMent?
<Nrf>=1~4(エレメント)

例 :FLICKer:DISPlay:ELEMENT 1
:FLICKer:DISPlay:ELEMENT? ->
:FLICKer:DISPlay:ELEMENT 1

解説 「:DISPlay:FLICKer:ELEMent」でも同じ設定/問い合わせができます。

:FLICKer:DISPlay:PAGE

機能 フリッカ測定表示の表示ページ番号を設定/問い合わせします。

構文 :FLICKer:DISPlay:PAGE {<Nrf>}

:FLICKer:DISPlay:PAGE?
<Nrf>=1~9(ページ番号)

例 :FLICKer:DISPlay:PAGE 1
:FLICKer:DISPlay:PAGE? ->
:FLICKer:DISPlay:PAGE 1

解説 「:DISPlay:FLICKer:PAGE」でも同じ設定/問い合わせができます。

:FLICKer:DISPlay:PERiod

機能 フリッカ測定表示の表示観測期間番号を設定/問い合わせします。

構文 :FLICKer:DISPlay:PERiod {<NRf>}
:FLICKer:DISPlay:PERiod?
<NRf>=1~99(観測期間番号)

例 :FLICKER:DISPLAY:PERIOD 1
:FLICKER:DISPLAY:PERIOD? ->
:FLICKER:DISPLAY:PERIOD 1

解説 「:DISPlay:FLICKer:PERiod」でも同じ設定/問い合わせができます。

:FLICKer:DMAX?

機能 最大相対電圧変化dmaxに関するすべての設定値を問い合わせます。

構文 :FLICKer:DMAX?
例 :FLICKER:DMAX? ->
:FLICKER:DMAX:STATE 1;LIMIT 4.00

:FLICKer:DMAX:LIMit

機能 最大相対電圧変化dmaxの限度値を設定/問い合わせします。

構文 :FLICKer:DMAX:LIMit {<NRf>}
:FLICKer:DMAX:LIMit?
<NRf>=1.00~99.99(限度値[%])

例 :FLICKER:DMAX:LIMIT 4.00
:FLICKER:DMAX:LIMIT? ->
:FLICKER:DMAX:LIMIT 4.00

:FLICKer:DMAX[:STATE]

機能 最大相対電圧変化dmaxの判定ON/OFFを設定/問い合わせします。

構文 :FLICKer:DMAX[:STATE] {<Boolean>}
:FLICKer:DMAX:STATE?
例 :FLICKER:DMAX:STATE ON
:FLICKER:DMAX:STATE? ->
:FLICKER:DMAX:STATE 1

:FLICKer:DMIN?

機能 定常範囲dminに関するすべての設定値を問い合わせます。

構文 :FLICKer:DMIN?
例 :FLICKER:DMIN? ->
:FLICKER:DMIN:LIMIT 0.10

:FLICKer:DMIN:LIMit

機能 定常範囲dminの限度値を設定/問い合わせします。

構文 :FLICKer:DMIN:LIMit {<NRf>}
:FLICKer:DMIN:LIMit?
<NRf>=0.10~9.99(限度値[%])
例 :FLICKER:DMIN:LIMIT 0.10
:FLICKER:DMIN:LIMIT? ->
:FLICKER:DMIN:LIMIT 0.10

:FLICKer:DT?

機能 相対電圧変化時間d(t)に関するすべての設定値を問い合わせます。

構文 :FLICKer:DT?
例 :FLICKER:DT? ->
:FLICKER:DT:STATE 1;LIMIT 500,3.30

:FLICKer:DT:LIMit

機能 相対電圧変化時間d(t)の限度値を設定/問い合わせします。

構文 :FLICKer:DT:LIMit {<NRf>[,<NRf>]}
:FLICKer:DT:LIMit?
1つ目の<NRf>=1~99999(限度値[ms])
2つ目の<NRf>=1.00~99.99(スレッシュホールドレベル[%])

例 :FLICKER:DT:LIMIT 500,3.30
:FLICKER:DT:LIMIT? ->
:FLICKER:DT:LIMIT 500,3.30

解説 2つ目のパラメータ(スレッシュホールドレベル)の設定をしないときは、省略することができます。

:FLICKer:DT[:STATE]

機能 相対電圧変化時間d(t)の判定ON/OFFを設定/問い合わせします。

構文 :FLICKer:DT[:STATE] {<Boolean>}
:FLICKer:DT:STATE?
例 :FLICKER:DT:STATE ON
:FLICKER:DT:STATE? ->
:FLICKER:DT:STATE 1

:FLICKer:ELEment<x>

機能 フリッカ測定対象エレメントを設定/問い合わせします。

構文 :FLICKer:ELEment<x> {<Boolean>}
:FLICKer:ELEment<x>?
<x>=1~4(エレメント)

例 :FLICKER:ELEMENT1 ON
:FLICKER:ELEMENT1? ->
:FLICKER:ELEMENT1 1

解説 「ON(1)」に設定すると、そのエレメントはフリッカ測定の対象となります。

:FLICKer:FREquency

機能 フリッカ測定対象周波数を設定/問い合わせします。

構文 :FLICKer:FREquency {<NRf>}
:FLICKer:FREquency?
<NRf>=50, 60(対象周波数[Hz])
例 :FLICKER:FREQUENCY 50
:FLICKER:FREQUENCY? ->
:FLICKER:FREQUENCY 50

:FLICKer:INITialize

機能 フリッカ測定の初期化を実行します。

構文 :FLICKer:INITialize
例 :FLICKER:INITIALIZE

:FLICKer:INTERval

機能	短期間フリッカ値Pstの1回の測定時間を設定/問い合わせします。
構文	:FLICKer:INTERval {<NRf>,<NRf>} :FLICKer:INTERval? <NRf>,<NRf>=0, 30~15, 00(測定時間:分, 秒)
例	:FLICKer:INTERval 10,00 :FLICKer:INTERval? -> :FLICKer:INTERval 10,00
解説	・通常のフリッカ測定(:FLICKer:MEASurement FLICKer)のときに有効な設定です。手動スイッチングdmax測定(:FLICKer:MEASurement DMAX)のときの1回の測定時間は、1(分)00(秒)固定です。 ・測定時間の設定分解能は2秒です。奇数の秒を設定したときは、+1(秒)に切り上げられます。

:FLICKer:JUDGE

機能	手動スイッチングdmax測定を完了し判定をおこないます。
構文	:FLICKer:JUDGE
例	:FLICKer:JUDGE
解説	手動スイッチングdmax測定(:FLICKer:MEASurement DMAX)のときに実行可能です。通常のフリッカ測定(:FLICKer:MEASurement FLICKer)のときはエラーが発生します。

:FLICKer:MEASurement

機能	フリッカ測定方法を設定/問い合わせします。
構文	:FLICKer:MEASurement { FLICKer DMAX } :FLICKer:MEASurement? FLICKer=通常のフリッカ測定 DMAX=手動スイッチングdmax測定
例	:FLICKer:MEASUREMENT FLICKER :FLICKer:MEASUREMENT? -> :FLICKer:MEASUREMENT FLICKER

:FLICKer:MOVE

機能	手動スイッチングdmax測定の観測期間番号を移動します。
構文	:FLICKer:MOVE {<NRf>} <NRf>=1~24(移動先の観測期間番号)
例	:FLICKer:MOVE 1
解説	・ある観測期間のdmax測定が正しく実行されなかった場合、再測定を行いたいときに実行します。 ・手動スイッチングdmax測定(:FLICKer:MEASurement DMAX)のときに実行可能です。通常のフリッカ測定(:FLICKer:MEASurement FLICKer)のときはエラーが発生します。

:FLICKer:PLT?

機能	長期間フリッカ値Pltに関するすべての設定値を問い合わせます。
構文	:FLICKer:PLT?
例	:FLICKer:PLT? -> :FLICKer:PLT:STATE 1;LIMIT 0.65;NVALUE 12

:FLICKer:PLT:LIMit

機能	長期間フリッカ値Pltの限度値を設定/問い合わせします。
構文	:FLICKer:PLT:LIMit {<NRf>} :FLICKer:PLT:LIMit? <NRf>=0.10~99.99(限度値)
例	:FLICKer:PLT:LIMIT 0.65 :FLICKer:PLT:LIMIT? -> :FLICKer:PLT:LIMIT 0.65

:FLICKer:PLT:NVALue

機能	長期間フリッカ値Pltの算出演算式の定数Nを設定/問い合わせします。
構文	:FLICKer:PLT:NVALue {<NRf>} :FLICKer:PLT:NVALue? <NRf>=1~99(定数N)
例	:FLICKer:PLT:NVALUE 12 :FLICKer:PLT:NVALUE? -> :FLICKer:PLT:NVALUE 12

:FLICKer:PLT[:STATE]

機能	長期間フリッカ値Pltの判定ON/OFFを設定/問い合わせします。
構文	:FLICKer:PLT[:STATE] {<Boolean>} :FLICKer:PLT:STATE?
例	:FLICKer:PLT:STATE ON :FLICKer:PLT:STATE? -> :FLICKer:PLT:STATE 1

:FLICKer:PST?

機能	短期間フリッカ値Pstに関するすべての設定値を問い合わせます。
構文	:FLICKer:PST?
例	:FLICKer:PST? -> :FLICKer:PST:STATE 1;LIMIT 1.00

:FLICKer:PST:LIMit

機能	短期間フリッカ値Pstの限度値を設定/問い合わせします。
構文	:FLICKer:PST:LIMit {<NRf>} :FLICKer:PST:LIMit? <NRf>=0.10~99.99(限度値)
例	:FLICKer:PST:LIMIT 1.00 :FLICKer:PST:LIMIT? -> :FLICKer:PST:LIMIT 1.00

6.9 FLICKerグループ

:FLICKer:PST[:STATe]

機能 短期間フリッカ値Pstの判定ON/OFFを設定/問い合わせします。

構文 **:FLICKer:PST[:STATe]** {<Boolean>}
:FLICKer:PST:STATe?

例 **:FLICKer:PST:STATe ON**
:FLICKer:PST:STATe? ->
:FLICKer:PST:STATe 1

:FLICKer:RESet

機能 フリッカの測定データをリセットします。

構文 **:FLICKer:RESet**

例 **:FLICKer:RESet**

:FLICKer:STARt

機能 フリッカ測定を開始します。

構文 **:FLICKer:STARt**

例 **:FLICKer:STARt**

:FLICKer:STATe?

機能 フリッカ測定ステータスを問い合わせます。

構文 **:FLICKer:STATe?**

例 **:FLICKer:STATe? -> RESet**

解説 応答の内容は次のとおりです。

RESet=リセット状態

INITialize=初期化中

READY=測定開始待機状態

STARt=測定中

COMPLete=測定終了・判定結果表示

:FLICKer:UN?

機能 定格電圧Unに関するすべての設定値を問い合わせます。

構文 **:FLICKer:UN?**

例 **:FLICKer:UN? -> :FLICKer:UN:**
MODE AUTO;VALUE 230.00

:FLICKer:UN:MODE

機能 定格電圧Unの取得方式を設定/問い合わせします。

構文 **:FLICKer:UN:MODE** {AUTO|SET}
:FLICKer:UN:MODE?

AUTO=測定開始時の測定電圧値を使用する

SET=既定値(**:FLICKer:UN:VALue**)を使用する

例 **:FLICKer:UN:MODE AUTO**
:FLICKer:UN:MODE? ->
:FLICKer:UN:MODE AUTO

:FLICKer:UN:VALue

機能 定格電圧Unの既定値を設定/問い合わせします。

構文 **:FLICKer:UN:VALue** {<NRf>}
:FLICKer:UN:VALue?
<NRf>=0.01~999.99(既定値[V])

例 **:FLICKer:UN:VALue 230.00**
:FLICKer:UN:VALue? ->
:FLICKer:UN:VALue 230.00

6.10 HARMonicsグループ

HARMonicsグループは、高調波測定に関するグループです。

フロントパネルのHRM SET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、このグループのコマンドは、高度演算機能(オプション、/G6)または高調波測定機能(オプション、/G5)の搭載時のみ有効です。

:HARMonics?

機能 高調波測定に関するすべての設定値を問い合わせます。

構文 :HARMonics?

例 :HARMONICS? -> :HARMONICS:
FBAND NORMAL;PLLSOURCE U1;
ORDER 1,100;THD TOTAL;IEC:
OBJECT ELEMENT1;UGROUPING OFF;
IGROUPING OFF;:HARMONICS:
PLLWARNING:STATE 1

:HARMonics:FBAND

機能 高調波測定の周波数帯域を設定/問い合わせします。

構文 :HARMonics:FBAND {NORMAL|WIDE}
:HARMonics:FBAND?

例 :HARMONICS:FBAND NORMAL
:HARMONICS:FBAND? ->

:HARMONICS:FBAND NORMAL
解説 ・ 高度演算機能(オプション、/G6)搭載時のみ有効です。
・ {NORMAL|WIDE}に対応する周波数帯域については、拡張機能ユーザーズマニュアルIM760301-51を参照してください。

:HARMonics:IEC?

機能 IEC高調波測定に関するすべての設定値を問い合わせます。

構文 :HARMonics:IEC?

例 :HARMONICS:IEC? ->
:HARMONICS:IEC:OBJECT
ELEMENT1;UGROUPING OFF;IGROUPING
OFF

解説 高度演算機能(オプション、/G6)搭載時のみ有効です。

:HARMonics:IEC:OBject

機能 IEC高調波測定対象を設定/問い合わせします。

構文 :HARMonics:IEC:OBject {ELEMENT<x>|
SIGMA|SIGMB}
:HARMonics:IEC:OBject?
<x>=1~4(エレメント)

例 :HARMONICS:IEC:OBJECT ELEMENT1
:HARMONICS:IEC:OBJECT? ->

:HARMONICS:IEC:OBJECT ELEMENT1
解説 高度演算機能(オプション、/G6)搭載時のみ有効です。

:HARMonics:IEC:{UGrouping|IGrouping}

機能 IEC高調波測定の{電圧|電流}グルーピングを設定/問い合わせします。

構文 :HARMonics:IEC:{UGrouping|
IGrouping} {OFF|TYPE1|TYPE2}
:HARMonics:IEC:{UGrouping|
IGrouping}?

例 :HARMONICS:IEC:UGROUPING OFF
:HARMONICS:IEC:UGROUPING? ->
:HARMONICS:IEC:UGROUPING OFF

解説 ・ 高度演算機能(オプション、/G6)搭載時のみ有効です。
・ {OFF|TYPE1|TYPE2}に対応するグルーピング設定については、拡張機能ユーザーズマニュアルIM760301-51を参照してください。

:HARMonics:ORDER

機能 最小/最大測定次数を設定/問い合わせします。

構文 :HARMonics:ORDER {<Nrf>,<Nrf>}
:HARMonics:ORDER?
1つ目の<Nrf>=0, 1(最小測定次数)
2つ目の<Nrf>=1~100(最大測定次数)

例 :HARMONICS:ORDER 1,100
:HARMONICS:ORDER? ->
:HARMONICS:ORDER 1,100

:HARMonics:PLLSource

機能 PLLソースを設定/問い合わせします。

構文 :HARMonics:PLLSource {U<x>|I<x>|
EXTernal|SAMPle}
:HARMonics:PLLSource?
<x>=1~4(エレメント)
EXTernal=外部クロック入力(Ext Clk)
SAMPle=サンプリングクロック入力(Smp Clk)

例 :HARMONICS:PLLSOURCE U1
:HARMONICS:PLLSOURCE? ->
:HARMONICS:PLLSOURCE U1

解説 ・ 「SAMPle」は高度演算機能(オプション、/G6)搭載時のみ選択できます。
・ 「SAMPle」を選択した場合は、広帯域高調波測定モード(Wide-Band Harmonics)のときに「SAMPle」が採用されます。それ以外の測定モードでは「EXTernal」が採用されます。問い合わせの応答にも「EXTernal」を返します。

6.10 HARMonicsグループ

:HARMonics:PLLWarning?

機能	PLLソースの警告メッセージに関するすべての設定値を問い合わせます。
構文	:HARMonics:PLLWarning?
例	:HARMONICS:PLLWARNING? -> :HARMONICS:PLLWARNING:STATE 1
解説	高度演算機能(オプション, /G6)搭載時のみ有効です。

:HARMonics:PLLWarning[:STATe]

機能	PLLソース未入力時の警告メッセージを出す/出さないを設定/問い合わせします。
構文	:HARMonics:PLLWarning [:STATe] {<Boolean>} :HARMonics:PLLWarning:STATe?
例	:HARMONICS:PLLWARNING:STATE ON :HARMONICS:PLLWARNING:STATE? -> :HARMONICS:PLLWARNING:STATE 1
解説	・ 高度演算機能(オプション, /G6)搭載時のみ有効です。 ・ 広帯域高調波測定モード(Wide-Band Harmonics)のときに有効な設定です。

:HARMonics:THD

機能	THD(高調波ひずみ率)の算出式を設定/問い合わせします。
構文	:HARMonics:THD {TOTal FUNDamental} :HARMonics:THD?
例	:HARMONICS:THD TOTAL :HARMONICS:THD? -> :HARMONICS:THD TOTAL

6.11 HCOpyグループ

HCOpyグループは、内蔵プリンタやネットワークプリンタでの印刷に関するグループです。
フロントパネルのPRINT、MENU(SHIFT+PRINT)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、このグループのコマンドは、内蔵プリンタ(オプション、/B5)またはイーサネット(オプション、/C7)搭載時のみ有効です。

:HCOpy? (Hard COPY)

機能 印刷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:
FORMAT HCOpy;:HCOpy:AUTO:STATE 0;;
HCOpy:COMMENT "THIS IS TEST."

:HCOpy:ABORT

機能 印刷または紙送りを中止します。

構文 :HCOpy:ABORT

例 :HCOpy:ABORT

:HCOpy:AUTO?

機能 オートプリントに関するすべての設定値を問い合わせます。

構文 :HCOpy:AUTO?

例 :HCOpy:AUTO? -> :HCOpy:AUTO:
STATE 1;SYNCHRONIZE TIMER;
START 2005,1,1,0,0,0;
END 2005,1,1,1,0,0;INTERVAL 0,0,10

:HCOpy:AUTO:INTERVAL

機能 オートプリントの印字インターバルを設定/問い合わせします。

構文 :HCOpy:AUTO:INTERVAL
{<NRf>,<NRf>,<NRf>}
:HCOpy:AUTO:INTERVAL?
{<NRf>,<NRf>,<NRf>}=0, 0, 10~99,
59, 59
1つ目の<NRf>=0~99 (時間)
2つ目の<NRf>=0~59 (分)
3つ目の<NRf>=0~59 (秒)

例 :HCOpy:AUTO:INTERVAL 0,0,10
:HCOpy:AUTO:INTERVAL? ->
:HCOpy:AUTO:INTERVAL 0,0,10

:HCOpy:AUTO:{START|END}

機能 オートプリントの{スタート|ストップ}予約時刻を設定/問い合わせします。

構文 :HCOpy:AUTO:{START|END}
{<NRf>,<NRf>,<NRf>,<NRf>,<NRf>,<NRf>}
:HCOpy:AUTO:{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 :HCOpy:AUTO:START 2005,1,1,0,0,0
:HCOpy:AUTO:START? ->
:HCOpy:AUTO:START 2005,1,1,0,0,0

解説 同期方式(:HCOpy:AUTO:SYNCHRONIZE)が「TIme(タイマ同期印字)」のときに有効な設定です。

:HCOpy:AUTO[:STATE]

機能 オートプリントのON/OFFを設定/問い合わせします。

構文 :HCOpy:AUTO[:STATE] {<Boolean>}
:HCOpy:AUTO:STATE?

例 :HCOpy:AUTO:STATE ON
:HCOpy:AUTO:STATE? ->
:HCOpy:AUTO:STATE 1

:HCOpy:AUTO:SYNCHRONIZE

機能 オートプリントの同期方式を設定/問い合わせします。

構文 :HCOpy:AUTO:SYNCHRONIZE {TIme|INTEGrate}
:HCOpy:AUTO:SYNCHRONIZE?
TIme=タイマ同期印字
INTEGrate=積算同期印字

例 :HCOpy:AUTO:SYNCHRONIZE TIMER
:HCOpy:AUTO:SYNCHRONIZE? ->
:HCOpy:AUTO:SYNCHRONIZE TIMER

: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."

:HCOpy:DIRectiOn

機能	プリンタを設定/問い合わせします。
構文	:HCOpy:DIRectiOn {PRINter NETPrint} :HCOpy:DIRectiOn? PRINter=内蔵プリンタ NETPrint=ネットワークプリンタ
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、/C7)搭載時のみ有効です。

:HCOpy:EXECute

機能	印刷を実行します。
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:NETPrint?

機能	ネットワークプリンタでの印刷に関するすべての設定値を問い合わせます。
構文	:HCOpy:NETPrint?
例	:HCOpy:NETPrint? -> :HCOpy:NETPRINT:FORMAT BJ,180;COLOR 0
解説	イーサネット(オプション、/C7)搭載時のみ有効です。

:HCOpy:NETPrint:COLor

機能	ネットワークプリンタで印刷するときのカラー出力のON/OFFを設定/問い合わせします。
構文	:HCOpy:NETPrint:COLor {<Boolean>} :HCOpy:NETPrint:COLor?
例	:HCOpy:NETPrint:COLor OFF :HCOpy:NETPrint:COLor? -> :HCOpy:NETPrint:COLor 0
解説	イーサネット(オプション、/C7)搭載時のみ有効です。

:HCOpy:NETPrint:FORMat

機能	ネットワークプリンタで印刷するときのプリンタ記述言語を設定/問い合わせします。
構文	:HCOpy:NETPrint:FORMat {PCL5 LIPS3 BJ,<NRf>} :HCOpy:NETPrint:FORMat? <NRf>=180, 300, 360(dpi, 解像度)
例	:HCOpy:NETPrint:FORMat BJ,180 :HCOpy:NETPrint:FORMat? -> :HCOpy:NETPrint:FORMat BJ,180
解説	・<NRf>は、「BJ」を選択したときだけ設定します。 ・イーサネット(オプション、/C7)搭載時のみ有効です。

:HCOpy:PRINter?

機能	内蔵プリンタでの印刷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:FORMat HCOpy
解説	内蔵プリンタ(オプション、/B5)搭載時のみ有効です。

: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	・内蔵プリンタ(オプション、/B5)搭載時のみ有効です。 ・このコマンドはオーバーラップコマンドです。

:HCOpy:PRINter:FORMat

機能	内蔵プリンタで印刷する内容を設定/問い合わせします。
構文	:HCOpy:PRINter:FORMat {HCOpy LIST} :HCOpy:PRINter:FORMat? HCOpy=画面イメージデータ LIST=数値データリスト
例	:HCOpy:PRINter:FORMat HCOpy :HCOpy:PRINter:FORMat? -> :HCOpy:PRINter:FORMat HCOpy
解説	内蔵プリンタ(オプション、/B5)搭載時のみ有効です。

:HCOpy:PRINter:LIST?

機能 内蔵プリンタでの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:LIST?

例 :HCOpy:PRINter:LIST? -> :HCOpy:PRINter:LIST:INFORMATION 1;NORMAL: ELEMENT1 1;ELEMENT2 0;ELEMENT3 0; ELEMENT4 0;SIGMA 0;SIGMB 0;U 1;I 1; P 1;S 1;Q 1;LAMBDA 1;PHI 1;FU 1; FI 1;UPPEAK 0;UMPEAK 0;IPPEAK 0; IMPEAK 0;CFU 0;CFI 0;PC 0;TIME 0; WH 0;WHP 0;WHM 0;AH 0;AHP 0;AHM 0; WS 0;WQ 0;ETA1 0;ETA2 0;ETA3 0; ETA4 0;F1 0;F2 0;F3 0;F4 0;F5 0; F6 0;F7 0;F8 0;F9 0;F10 0;F11 0; F12 0;F13 0;F14 0;F15 0;F16 0; F17 0;F18 0;F19 0;F20 0

解説 内蔵プリンタ(オプション、/B5)搭載時のみ有効です。

:HCOpy:PRINter:LIST:INformation

機能 内蔵プリンタで数値データリストを印字するときに、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:LIST:INformation {<Boolean>}

例 :HCOpy:PRINter:LIST:INformation? :HCOpy:PRINter:LIST:INformation ON :HCOpy:PRINter:LIST:INformation? -> :HCOpy:PRINter:LIST:INformation 1

解説 内蔵プリンタ(オプション、/B5)搭載時のみ有効です。

:HCOpy:PRINter:LIST:NORMAL?

機能 内蔵プリンタでの数値データリストの印字項目に関するすべての設定値を問い合わせます。

構文 :HCOpy:PRINter:LIST:NORMAL?

例 :HCOpy:PRINter:LIST:NORMAL? -> :HCOpy:PRINter:LIST:NORMAL: ELEMENT1 1;ELEMENT2 0;ELEMENT3 0; ELEMENT4 0;SIGMA 0;SIGMB 0;U 1;I 1; P 1;S 1;Q 1;LAMBDA 1;PHI 1;FU 1; FI 1;UPPEAK0;UMPEAK 0;IPPEAK 0; IMPEAK 0;CFU 0;CFI 0;PC 0;TIME 0; WH 0;WHP 0;WHM 0;AH 0;AHP 0;AHM 0; WS 0;WQ 0;ETA1 0;ETA2 0;ETA3 0; ETA4 0;F1 0;F2 0;F3 0;F4 0;F5 0; F6 0;F7 0;F8 0;F9 0;F10 0;F11 0; F12 0;F13 0;F14 0;F15 0;F16 0; F17 0;F18 0;F19 0;F20 0

解説 内蔵プリンタ(オプション、/B5)搭載時のみ有効です。

:HCOpy:PRINter:LIST[:NORMAL]:ALL

機能 内蔵プリンタで数値データリストを印字するときのすべてのエレメント・ファンクションの出力ON/OFFを一括設定します。

構文 :HCOpy:PRINter:LIST[:NORMAL]: ALL {<Boolean>}

例 :HCOpy:PRINter:LIST:NORMAL:ALL ON
解説 内蔵プリンタ(オプション、/B5)搭載時のみ有効です。

:HCOpy:PRINter:LIST[:NORMAL]: {ELEMENT<x>|SIGMA|SIGMB}

機能 内蔵プリンタで数値データリストを印字するときの{各エレメント|ΣA|ΣB}の出力ON/OFFを設定/問い合わせします。

構文 :HCOpy:PRINter:LIST[:NORMAL]: {ELEMENT<x>|SIGMA|SIGMB} {<Boolean>} :HCOpy:PRINter:LIST[:NORMAL]: {ELEMENT<x>|SIGMA|SIGMB}? <x>=1~4

例 :HCOpy:PRINter:LIST:NORMAL: ELEMENT1 ON :HCOpy:PRINter:LIST:NORMAL: ELEMENT1? -> :HCOpy:PRINter:LIST: NORMAL:ELEMENT1 1

解説
・内蔵プリンタ(オプション、/B5)搭載時のみ有効です。
・「:HCOpy:PRINter:LIST[:NORMAL]: SIGMA」は、2エレメント以上のモデルで有効です。また、出力をONに設定するためには、あらかじめ結線方式の設定 ([:INPut]WIRingコマンド)により結線ユニットΣAが存在する必要があります。
・「:HCOpy:PRINter:LIST[:NORMAL]: SIGMB」は、4エレメントモデルのみ有効です。また、出力をONに設定するためには、あらかじめ結線方式の設定 ([:INPut]WIRingコマンド)により結線ユニットΣBが存在する必要があります。

:HCOpy:PRINter:LIST[:NORMAL]:PRESet<x>

機能 内蔵プリンタで数値データリストを印字するときのエレメント・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:LIST[:NORMAL]: PRESet<x> <x>=1~2(プリセットパターン番号)

例 :HCOpy:PRINter:LIST:NORMAL:PRESET1
解説
・内蔵プリンタ(オプション、/B5)搭載時のみ有効です。
・プリセット実行をしたときの印字パターンについては、拡張機能ユーザズマニュアル IM760301-51を参照してください。

:HCOPY:PRINTER:LIST[:NORMAL]:<Function>

機能	内蔵プリンタで数値データリストを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:LIST[:NORMAL]: <Function> {<Boolean>} :HCOPY:PRINTER:LIST[:NORMAL]: <Function>? <Function>={U I P S Q ...} (「DISPlayグループ」のファンクション選択肢一覧(1), 6-41ページ参照)
例	:HCOPY:PRINTER:LIST:NORMAL:U ON :HCOPY:PRINTER:LIST:NORMAL:U? -> :HCOPY:PRINTER:LIST:NORMAL:U 1
解説	内蔵プリンタ(オプション, /B5)搭載時のみ有効です。

6.12 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>}**

:HOLD?

例 **:HOLD OFF**

:HOLD? -> :HOLD 0

6.13 IMAGEグループ

IMAGEグループは、画面イメージデータの保存に関するグループです。
フロントパネルのIMAGE SAVE、MENU(SHIFT+ IMAGE SAVE)と同じ設定、および設定内容の問い合わせができます。

: IMAGE?

機能 画面イメージ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 : IMAGE?

例 : IMAGE? -> : IMAGE:FORMAT TIFF;
COLOR OFF;COMMENT "THIS IS TEST.";
SAVE:ANAMING 1;NAME ""

: IMAGE:ABORT

機能 画面イメージデータの保存を中止します。

構文 : IMAGE:ABORT

例 : IMAGE:ABORT

: IMAGE:COLor

機能 保存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|GRAY}

: IMAGE:COLor?

例 : IMAGE:COLOR OFF

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

解説 形式(: IMAGE:FORMat)がPSCript以外のときに有効な設定です。

: IMAGE: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 : IMAGE:COMMeNt {<文字列>}

: IMAGE:COMMeNt?

<文字列>=25文字以内(ただし、表示は20文字までです。)

例 : IMAGE:COMMENT "THIS IS TEST."

: IMAGE:COMMENT? -> : IMAGE:

COMMENT "THIS IS TEST."

: IMAGE:COMPreSSion

機能 画面イメージデータのBMPフォーマット時のデータ圧縮の有無を設定/問い合わせします。

構文 : IMAGE:COMPreSSion {<Boolean>}

: IMAGE:COMPreSSion?

例 : IMAGE:COMPRESSION ON

: IMAGE:COMPRESSION? ->

: IMAGE:COMPRESSION 1

解説 形式(: IMAGE:FORMat)がBMPで、色調(: IMAGE:COLor)が{COLor|REVerse|GRAY}のときに有効な設定です。

: IMAGE:EXECute

機能 画面イメージデータの保存を実行します。

構文 : IMAGE:EXECute

例 : IMAGE:EXECUTE

: IMAGE:FORMat

機能 保存する画面イメージデータの形式を設定/問い合わせします。

構文 : IMAGE:FORMat {TIFF|BMP|PSCript|PNG|JPEG}

: IMAGE:FORMat?

例 : IMAGE:FORMAT TIFF

: IMAGE:FORMAT? ->

: IMAGE:FORMAT TIFF

: IMAGE:SAVE?

機能 画面イメージ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 : IMAGE:SAVE?

例 : IMAGE:SAVE? ->

: IMAGE:SAVE:ANAMING 1;NAME ""

: IMAGE:SAVE:ANAMing

機能 画面イメージデータの保存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 : IMAGE:SAVE:ANAMing {<Boolean>}

: IMAGE:SAVE:ANAMing?

例 : IMAGE:SAVE:ANAMING ON

: IMAGE:SAVE:ANAMING? ->

: IMAGE:SAVE:ANAMING 1

: IMAGE:SAVE:CDIRectory

機能 画面イメージデータの保存先ディレクトリを変更します。

構文 : IMAGE:CDIRectory {<Filename>}

<Filename>=ディレクトリ名

例 : IMAGE: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。

: IMAGE:SAVE:DRIVE

機能 画面イメージデータの保存先ドライブを設定します。

構文 **:IMAGE:SAVE:DRIVE {PCCard[,<Nrf>] | NETWork|USB,<Nrf>[,<Nrf>][,<Nrf>]}**
 PCCard=PCカードドライブ
 <Nrf>=パーティション(0~3)
 NETWork=ネットワークドライブ
 USB=USBメモリドライブ
 最初の<Nrf>=ID番号(アドレス)
 2番目の<Nrf>=パーティション(0~3)またはLUN(論理ユニット番号: 0~3)
 3番目の<Nrf>=LUNを指定したときのパーティション(0~3)

例 **:IMAGE:SAVE:DRIVE PCCARD**
 解説
 ・パーティションを設定していないドライブの場合、パーティションの<Nrf>は省略してください。
 ・「NETWork」はイーサネット(オプション, /C7)搭載時に使用できます。
 ・「USB」はUSBポート(周辺機器)(オプション, /C5)搭載時に使用できます。
 ・「USB」の場合の2, 3番目の<Nrf>は、パーティションまたはLUNで区切られていないときは省略可能です。

: IMAGE:SAVE:NAME

機能 画面イメージデータの保存ファイル名を設定/問い合わせします。

構文 **:IMAGE:SAVE:NAME {<Filename>}**
:IMAGE:SAVE:NAME?

例 **:IMAGE:SAVE:NAME "IMAGE1"**
:IMAGE:SAVE:NAME? ->
:IMAGE:SAVE:NAME "IMAGE1"

解説
 ・画面イメージデータの保存先の、対象ドライブは「:IMAGE:SAVE:DRIVE」コマンドで、ディレクトリは「:IMAGE:SAVE:CDIRectory」コマンドでそれぞれ設定します。
 ・ファイル名は、拡張子を付けずに指定してください。

: 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、5-6ページを参照してください。

6.14 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。

フロントパネルの測定条件設定エリア(水色で囲まれたエリア)の各種キー、SCALING、LINE FILTER、FREQ FILTER (SHIFT+LINE FILTER)、SYNC SOURCE、およびNULL(SHIFT+SYNC SOURCE)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:CFACTOR 3;
WIRING P1W2,P1W2,P1W2,P1W2;
INDEPENDENT 0;VOLTAGE:RANGE:
ELEMENT1 1.000E+03;
ELEMENT2 1.000E+03;
ELEMENT3 1.000E+03;
ELEMENT4 1.000E+03;;INPUT:VOLTAGE:
AUTO:ELEMENT1 0;ELEMENT2 0;
ELEMENT3 0;ELEMENT4 0;;INPUT:
VOLTAGE:MODE:ELEMENT1 RMS;
ELEMENT2 RMS;ELEMENT3 RMS;
ELEMENT4 RMS;;INPUT:CURRENT:RANGE:
ELEMENT1 30.0E+00;
ELEMENT2 30.0E+00;
ELEMENT3 30.0E+00;
ELEMENT4 30.0E+00;;INPUT:CURRENT:
AUTO:ELEMENT1 0;ELEMENT2 0;
ELEMENT3 0;ELEMENT4 0;;INPUT:
CURRENT:MODE:ELEMENT1 RMS;
ELEMENT2 RMS;ELEMENT3 RMS;
ELEMENT4 RMS;;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;;INPUT:FILTER:
FREQUENCY:ELEMENT1 0;ELEMENT2 0;
ELEMENT3 0;ELEMENT4 0;;INPUT:
SCALING:STATE:ELEMENT1 0;
ELEMENT2 0;ELEMENT3 0;ELEMENT4 0;;
INPUT:SCALING:VT:ELEMENT1 1.0000;
ELEMENT2 1.0000;ELEMENT3 1.0000;
ELEMENT4 1.0000;;INPUT:SCALING:CT:
ELEMENT1 1.0000;ELEMENT2 1.0000;
ELEMENT3 1.0000;ELEMENT4 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] :CFActor

機能 クレストファクタを設定/問い合わせします。

構文 **[:INPut] :CFActor {<Nrf>}**
[:INPut] :CFActor?
<Nrf>=3, 6

例 **:INPUT:CFACTOR 3
:INPUT:CFACTOR? -> :INPUT:CFACTOR 3**

[: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:
RANGE:ELEMENT1 30.0E+00;
ELEMENT2 30.0E+00;
ELEMENT3 30.0E+00;
ELEMENT4 30.0E+00;;INPUT:CURRENT:
AUTO:ELEMENT1 0;ELEMENT2 0;
ELEMENT3 0;ELEMENT4 0;;INPUT:
CURRENT:MODE:ELEMENT1 RMS;
ELEMENT2 RMS;ELEMENT3 RMS;
ELEMENT4 RMS;;INPUT:CURRENT:SRATIO:
ELEMENT1 10.0000;ELEMENT2 10.0000;
ELEMENT3 10.0000;ELEMENT4 10.0000**

[:INPut] :CURRent:AUTO?

機能 すべてのエレメントの電流オートレンジON/OFFを問い合わせます。

構文 **[:INPut] :CURRent:AUTO?**
例 **:INPUT:CURRENT:AUTO? ->
:INPUT:CURRENT:AUTO:ELEMENT1 0;
ELEMENT2 0;ELEMENT3 0;ELEMENT4 0**

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

機能 すべての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**

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

機能 各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(エレメント)

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

[: INPut] : CURRent : AUTO : { SIGMA | SIGMB }

機能 結線ユニット{ΣA|ΣB}に属するすべての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 : { SIGMA | SIGMB } { < Boolean > }

例 : INPUT:CURRENT:AUTO:SIGMA ON
解説 ・ [: INPut] : CURRent : AUTO : SIGMAは、2~4エレメントモデルのみ有効です。
・ [: INPut] : CURRent : AUTO : SIGMBは、4エレメントモデルのみ有効です。
・ 結線方式の設定([: INPut] : WIRing)により結線ユニット{ΣA|ΣB}が存在しないときは、このコマンドは無効です。

[: INPut] : CURRent : MODE ?

機能 すべてのエレメントの電流モードを問い合わせます。

構文 [: INPut] : CURRent : MODE ?

例 : INPUT:CURRENT:MODE? ->
: INPUT:CURRENT:MODE:ELEMENT1 RMS;
ELEMENT2 RMS;ELEMENT3 RMS;
ELEMENT4 RMS

[: INPut] : CURRent : MODE [: ALL]

機能 すべてのエレメントの電流モードを一括設定します。

構文 [: INPut] : CURRent : MODE [: ALL] { RMS | MEAN | DC | RMEAN }

例 : INPUT:CURRENT:MODE:ALL RMS

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

機能 各エレメントの電流モードを設定/問い合わせします。

構文 [: INPut] : CURRent : MODE : ELEMeNt < x > { RMS | MEAN | DC | RMEAN }
[: INPut] : CURRent : MODE : ELEMeNt < x > ?
< x > = 1~4(エレメント)

例 : INPUT:CURRENT:MODE:ELEMENT1 RMS
: INPUT:CURRENT:MODE:ELEMENT1? ->
: INPUT:CURRENT:MODE:ELEMENT1 RMS

[: INPut] : CURRent : MODE : { SIGMA | SIGMB }

機能 結線ユニット{ΣA|ΣB}に属する全エレメントの電流モードを一括設定します。

構文 [: INPut] : CURRent : MODE : { SIGMA | SIGMB } { RMS | MEAN | DC | RMEAN }

例 : INPUT:CURRENT:MODE:SIGMA RMS
解説 ・ [: INPut] : CURRent : MODE : SIGMAは、2~4エレメントモデルのみ有効です。
・ [: INPut] : CURRent : MODE : SIGMBは、4エレメントモデルのみ有効です。
・ 結線方式の設定([: INPut] : WIRing)により結線ユニット{ΣA|ΣB}が存在しないときは、このコマンドは無効です。

[: INPut] : CURRent : RANGE ?

機能 すべての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 30.0E+00;
ELEMENT2 30.0E+00;
ELEMENT3 30.0E+00;
ELEMENT4 30.0E+00

[: INPut] : CURRent : RANGe [: ALL]

機能	すべての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> (EXTeRnal , <電圧>) } (標準)電力エレメントモデル(760301-01～760304-04)のとき ・ クレストファクタを「3」に設定しているとき <電流>=500(mA), 1, 2, 5, 10, 20, 30 (A) (電流直接入力の場合) <電圧>=50, 100, 200, 500(mV), 1, 2, 5, 10(V) (電流外部センサ入力の場合) ・ クレストファクタを「6」に設定しているとき <電流>=250, 500(mA), 1, 2.5, 5, 10, 15(A) (電流直接入力の場合) <電圧>=25, 50, 100, 250, 500(mV), 1, 2.5, 5(V) (電流外部センサ入力の場合) 低電流レンジ電力エレメントモデル(760301-10～760304-40)のとき ・ クレストファクタを「3」に設定しているとき <電流>=5, 10, 20, 50, 100, 200, 500 (mA), 1, 2(A) (電流直接入力の場合) <電圧>=50, 100, 200, 500(mV), 1, 2, 5, 10(V) (電流外部センサ入力の場合) ・ クレストファクタを「6」に設定しているとき <電流>=2.5, 5, 10, 25, 50, 100, 250, 500(mA), 1(A) (電流直接入力の場合) <電圧>=25, 50, 100, 250, 500(mV), 1, 2.5, 5(V) (電流外部センサ入力の場合)
例	: INPUT:CURRENT:RANGE:ALL 30A : INPUT:CURRENT:RANGE: ALL EXTERNAL,10V

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

機能	各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> (EXTeRnal , <電圧>) } [: INPut] : CURRent : RANGe : ELEMeNt <x> ? <x>=1～4(エレメント) (標準)電力エレメントモデル(760301-01～760304-04)のとき ・ クレストファクタを「3」に設定しているとき <電流>=500(mA), 1, 2, 5, 10, 20, 30 (A) (電流直接入力の場合) <電圧>=50, 100, 200, 500(mV), 1, 2, 5, 10(V) (電流外部センサ入力の場合) ・ クレストファクタを「6」に設定しているとき <電流>=250, 500(mA), 1, 2.5, 5, 10, 15(A) (電流直接入力の場合) <電圧>=25, 50, 100, 250, 500(mV), 1, 2.5, 5(V) (電流外部センサ入力の場合) 低電流レンジ電力エレメントモデル(760301-10～760304-40)のとき ・ クレストファクタを「3」に設定しているとき <電流>=5, 10, 20, 50, 100, 200, 500 (mA), 1, 2(A) (電流直接入力の場合) <電圧>=50, 100, 200, 500(mV), 1, 2, 5, 10(V) (電流外部センサ入力の場合) ・ クレストファクタを「6」に設定しているとき <電流>=2.5, 5, 10, 25, 50, 100, 250, 500(mA), 1(A) (電流直接入力の場合) <電圧>=25, 50, 100, 250, 500(mV), 1, 2.5, 5(V) (電流外部センサ入力の場合)
例	: INPUT:CURRENT:RANGE:ELEMENT1 30A : INPUT:CURRENT:RANGE:ELEMENT1? -> : INPUT:CURRENT:RANGE: ELEMENT1 30.0E+00 : INPUT:CURRENT:RANGE: ELEMENT1 EXTERNAL,10V : INPUT:CURRENT:RANGE:ELEMENT1? -> : INPUT:CURRENT:RANGE: ELEMENT1 EXTERNAL,10.00E+00

[:INPut] :CURRent:RANGe: {SIGMA|SIGMB}

機能 結線ユニット{ΣA|ΣB}に属するすべてのエレメントの電流レンジを一括設定します。

構文 **[:INPut] :CURRent:RANGe: {SIGMA|SIGMB} {<電流>| (EXternal,<電圧>)} (標準)電力エレメントモデル(760301-01~760304-04)のとき**

- ・ クレストファクタを「3」に設定しているとき
<電流>=500(mA), 1, 2, 5, 10, 20, 30 (A) (電流直接入力の場合)
<電圧>=50, 100, 200, 500(mV), 1, 2, 5, 10(V) (電流外部センサ入力の場合)
- ・ クレストファクタを「6」に設定しているとき
<電流>=250, 500(mA), 1, 2.5, 5, 10, 15(A) (電流直接入力の場合)
<電圧>=25, 50, 100, 250, 500(mV), 1, 2.5, 5(V) (電流外部センサ入力の場合)

低電流レンジ電力エレメントモデル(760301-10~760304-40)のとき

- ・ クレストファクタを「3」に設定しているとき
<電流>=5, 10, 20, 50, 100, 200, 500 (mA), 1, 2(A) (電流直接入力の場合)
<電圧>=50, 100, 200, 500(mV), 1, 2, 5, 10(V) (電流外部センサ入力の場合)
- ・ クレストファクタを「6」に設定しているとき
<電流>=2.5, 5, 10, 25, 50, 100, 250, 500(mA), 1(A) (電流直接入力の場合)
<電圧>=25, 50, 100, 250, 500(mV), 1, 2.5, 5(V) (電流外部センサ入力の場合)

例 **:INPUT:CURRENT:RANGE:SIGMA 30A**

- 解説
- ・ **[:INPut] :CURRent:RANGe:SIGMA**は、2~4エレメントモデルのみ有効です。
 - ・ **[:INPut] :CURRent:RANGe:SIGMB**は、4エレメントモデルのみ有効です。
 - ・ 結線方式の設定(**[:INPut] :WIRing**)により結線ユニット{ΣA|ΣB}が存在しないときは、このコマンドは無効です。

[:INPut] :CURRent:SRATio?

機能 すべてのエレメントの電流センサスケール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**

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

機能 すべてのエレメントの電流センサスケール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**

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

機能 各エレメントの電流センサスケール定数を設定/問い合わせします。

構文 **[:INPut] :CURRent:SRATio:**

ELEMeNt<x> {<NRF>}

[:INPut] :CURRent:SRATio:ELEMeNt<x>?

<x>=1~4(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

[: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;;INPUT:FILTER:FREQUENCY:ELEMENT1 0; ELEMENT2 0;ELEMENT3 0;ELEMENT4 0**

[:INPut] :FILTer:FREQuency?

機能 すべてのエレメントの周波数フィルタの設定を問い合わせます。

構文 **[:INPut] :FILTer:FREQuency?**

例 **:INPUT:FILTER:FREQUENCY? ->**

:INPUT:FILTER:FREQUENCY:ELEMENT1 0;

ELEMENT2 0;ELEMENT3 0;ELEMENT4 0

[:INPut] :FILTer:FREQuency [:ALL]

機能 すべてのエレメントの周波数フィルタを一括設定します。

構文 **[:INPut] :FILTer:FREQuency**

[:ALL] {<Boolean>}

例 **:INPUT:FILTER:FREQUENCY:ALL OFF**

[: INPut] : FILTer : FREQuency : ELEMeNt < x >

機能 各エレメントの周波数フィルタを設定/問い合わせします。

構文 **[: INPut] : FILTer : FREQuency : ELEMeNt < x >** {<Boolean>}
[: INPut] : FILTer : FREQuency : ELEMeNt < x > ?
 <x>=1~4(エレメント)

例 **: INPUT:FILTER:FREQUENCY:ELEMENT1 ON**
: INPUT:FILTER:FREQUENCY:ELEMENT1?
-> : INPUT:FILTER:FREQUENCY: ELEMENT1 1

[: 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

[: 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, 50kHz(ラインフィルタON, カットオフ周波数)

例 **: INPUT:FILTER:LINE:ALL OFF**

[: 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>=1~4(エレメント)
 OFF=ラインフィルタOFF
 <周波数>=500Hz, 5.5kHz, 50kHz(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] : INDependent

機能 入力エレメントの個別設定ON/OFFを設定/問い合わせします。

構文 **[: INPut] : INDependent** {<Boolean>}
[: INPut] : INDependent ?

例 **: INPUT:INDEPENDENT OFF**
: INPUT:INDEPENDENT? ->
: INPUT:INDEPENDENT 0

解説 2~4エレメントモデルのみ有効です。

[: 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>=1~4(エレメント)

例 **: INPUT:MODULE? 1 -> 30**

: INPUT:MODULE? -> 30,30,30,30

解説 ・ 応答の内容は次のとおりです。
 30=(標準)電力エレメント(最大電流レンジ=30A)
 2=低電流レンジ電力エレメント(最大電流レンジ=2A)
 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 ピークオーバー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ー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。
 ・ たとえば、応答が「16」の場合、U3にピークオーバー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	Tq	Sp	0	0	0	0	14	U4	U3	U2	U1	U1	U1	U1

Sp: 回転速度
Tq: トルク

[: 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;;INPUT: SCALING:VT:ELEMENT1 1.0000; ELEMENT2 1.0000;ELEMENT3 1.0000; ELEMENT4 1.0000;;INPUT:SCALING:CT: ELEMENT1 1.0000;ELEMENT2 1.0000; ELEMENT3 1.0000;ELEMENT4 1.0000;; INPUT:SCALING:SFACTOR: ELEMENT1 1.0000;ELEMENT2 1.0000; ELEMENT3 1.0000;ELEMENT4 1.0000**

[: INPut] : SCALing : { VT | CT | SFACtor } ?

機能 すべてのエレメントの{VT比|CT比|電力係数}を問い合わせます。

構文 **[: INPut] : SCALing : { VT | CT | SFACtor } ?**
 例 **: INPUT : SCALING : VT ? ->**
: INPUT : SCALING : VT : ELEMENT1 1.0000 ;
ELEMENT2 1.0000 ; ELEMENT3 1.0000 ;
ELEMENT4 1.0000

[: INPut] : SCALing : { VT | CT | SFACtor } [: ALL]

機能 すべてのエレメントの{VT比|CT比|電力係数}を一括設定します。

構文 **[: INPut] : SCALing : { VT | CT | SFACtor } [: ALL] { < NRf > }**
< NRf > = 0.0001 ~ 99999.9999
 例 **: INPUT : SCALING : VT : ALL 1**

[: INPut] : SCALing : { VT | CT | SFACtor } : ELEMENT < x >

機能 各エレメントの{VT比|CT比|電力係数}を設定/問い合わせします。

構文 **[: INPut] : SCALing : { VT | CT | SFACtor } : ELEMENT < x > { < NRf > }**
[: INPut] : SCALing : { VT | CT | SFACtor } : ELEMENT < x > ?
< x > = 1 ~ 4 (エレメント)
< NRf > = 0.0001 ~ 99999.9999
 例 **: INPUT : SCALING : VT : ELEMENT1 1**
: INPUT : SCALING : VT : ELEMENT1 ? ->
: INPUT : SCALING : VT : ELEMENT1 1.0000

[: INPut] : SCALing : STATE ?

機能 すべてのエレメントのスケーリング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

[: INPut] : SCALing [: STATE] [: ALL]

機能 すべてのエレメントのスケーリングON/OFFを一括設定します。

構文 **[: INPut] : SCALing [: STATE] [: ALL] { < Boolean > }**
 例 **: INPUT : SCALING : STATE : ALL OFF**

[: INPut] : SCALing [: STATE] : ELEMENT < x >

機能 各エレメントのスケーリング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 ~ 4 (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

[: INPut] : SYNChronize [: ALL]

機能 すべての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 > | EXTeRnal | NONE }**
< x > = 1 ~ 4 (エレメント)
EXTeRnal = 外部クロック入力 (Ext Clk)
NONE = 同期ソースなし
 例 **: INPUT : SYNCHRONIZE : ALL I1**

[: INPut] : SYNChronize : ELEMENT < x >

機能 各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 > | EXTeRnal | NONE }**
[: INPut] : SYNChronize : ELEMENT < x > ?
< x > = 1 ~ 4 (エレメント)
EXTeRnal = 外部クロック入力 (Ext Clk)
NONE = 同期ソースなし
 例 **: INPUT : SYNCHRONIZE : ELEMENT1 I1**
: INPUT : SYNCHRONIZE : ELEMENT1 ? ->
: INPUT : SYNCHRONIZE : ELEMENT1 I1

[: INPut] : SYNChronize : { SIGMA | SIGMB }

機能 結線ユニット { ΣA | ΣB } に属する全エレメントの同期ソースを一括設定します。

構文 **[: INPut] : SYNChronize : { SIGMA | SIGMB } { U < x > | I < x > | EXTeRnal | NONE }**
 例 **: INPUT : SYNCHRONIZE : SIGMA I1**
 解説

- ・ **[: INPut] : SYNChronize : SIGMA** は、2~4 エレメントモデルのみ有効です。
- ・ **[: INPut] : SYNChronize : SIGMB** は、4 エレメントモデルのみ有効です。
- ・ 結線方式の設定 (**[: INPut] : WIRing**) により結線ユニット { ΣA | ΣB } が存在しないときは、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 1.000E+03; ELEMENT2 1.000E+03; ELEMENT3 1.000E+03; ELEMENT4 1.000E+03; : INPUT: VOLTAGE: AUTO: ELEMENT1 0; ELEMENT2 0; ELEMENT3 0; ELEMENT4 0; : INPUT: VOLTAGE: MODE: ELEMENT1 RMS; ELEMENT2 RMS; ELEMENT3 RMS; ELEMENT4 RMS**

[: INPut] : VOLTage: AUTO?

機能 すべてのエレメントの電圧オートレンジON/OFFを問い合わせます。

構文 **[: INPut] : VOLTage: AUTO?**
 例 **: INPUT: VOLTAGE: AUTO? -> : INPUT: VOLTAGE: AUTO: ELEMENT1 0; ELEMENT2 0; ELEMENT3 0; ELEMENT4 0**

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

機能 すべての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**

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

機能 各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(エレメント)
 例 **: INPUT: VOLTAGE: AUTO: ELEMENT1 ON : INPUT: VOLTAGE: AUTO: ELEMENT1? -> : INPUT: VOLTAGE: AUTO: ELEMENT1 1**

[: INPut] : VOLTage: AUTO: { SIGMA | SIGMB }

機能 結線ユニット{ΣA|ΣB}に属するすべての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: { SIGMA | SIGMB } { <Boolean> }**
 例 **: INPUT: VOLTAGE: AUTO: SIGMA ON**
 解説

- ・ **[: INPut] : VOLTage: AUTO: SIGMA**は、2～4エレメントモデルのみ有効です。
- ・ **[: INPut] : VOLTage: AUTO: SIGMB**は、4エレメントモデルのみ有効です。
- ・ 結線方式の設定(**[: INPut] : WIRing**)により結線ユニット{ΣA|ΣB}が存在しないときは、このコマンドは無効です。

[: INPut] : VOLTage: MODE?

機能 すべてのエレメントの電圧モードを問い合わせます。

構文 **[: INPut] : VOLTage: MODE?**
 例 **: INPUT: VOLTAGE: MODE? -> : INPUT: VOLTAGE: MODE: ELEMENT1 RMS; ELEMENT2 RMS; ELEMENT3 RMS; ELEMENT4 RMS**

[: INPut] : VOLTage: MODE [: ALL]

機能 すべてのエレメントの電圧モードを一括設定します。

構文 **[: INPut] : VOLTage: MODE [: ALL] { RMS | MEAN | DC | RMEAN }**
 例 **: INPUT: VOLTAGE: MODE: ALL RMS**

[: INPut] : VOLTage: MODE: ELEMENT<x>

機能 各エレメントの電圧モードを設定/問い合わせします。

構文 **[: INPut] : VOLTage: MODE: ELEMENT<x> { RMS | MEAN | DC | RMEAN }**
[: INPut] : VOLTage: MODE: ELEMENT<x>? <x>=1~4(エレメント)
 例 **: INPUT: VOLTAGE: MODE: ELEMENT1 RMS : INPUT: VOLTAGE: MODE: ELEMENT1? -> : INPUT: VOLTAGE: MODE: ELEMENT1 RMS**

[: INPut] : VOLTage: MODE: { SIGMA | SIGMB }

機能 結線ユニット{ΣA|ΣB}に属する全エレメントの電圧モードを一括設定します。

構文 **[: INPut] : VOLTage: MODE: { SIGMA | SIGMB } { RMS | MEAN | DC | RMEAN }**
 例 **: INPUT: VOLTAGE: MODE: SIGMA RMS**
 解説

- ・ **[: INPut] : VOLTage: MODE: SIGMA**は、2～4エレメントモデルのみ有効です。
- ・ **[: INPut] : VOLTage: MODE: SIGMB**は、4エレメントモデルのみ有効です。
- ・ 結線方式の設定(**[: INPut] : WIRing**)により結線ユニット{ΣA|ΣB}が存在しないときは、このコマンドは無効です。

[: INPut] : VOLTage: RANGE?

機能 すべての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 1.000E+03; ELEMENT2 1.000E+03; ELEMENT3 1.000E+03; ELEMENT4 1.000E+03**

[: INPut] : VOLTage : RANGE [: ALL]

機能 すべての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> }**
 ・ クレストファクタを「3」に設定しているとき
 <電圧> = 15, 30, 60, 100, 150, 300, 600, 1000(V)
 ・ クレストファクタを「6」に設定しているとき
 <電圧> = 7.5, 15, 30, 50, 75, 150, 300, 500(V)

例 **: INPUT:VOLTAGE:RANGE:ALL 1000V**

[: INPut] : VOLTage : RANGE : ELEMENT <x>

機能 各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> }**
[: INPut] : VOLTage : RANGE : ELEMENT <x> ?
 <x> = 1~4(エレメント)
 ・ クレストファクタを「3」に設定しているとき
 <電圧> = 15, 30, 60, 100, 150, 300, 600, 1000(V)
 ・ クレストファクタを「6」に設定しているとき
 <電圧> = 7.5, 15, 30, 50, 75, 150, 300, 500(V)

例 **: INPUT:VOLTAGE:RANGE:ELEMENT1 1000V**
: INPUT:VOLTAGE:RANGE:ELEMENT1? ->
: INPUT:VOLTAGE:RANGE:ELEMENT1
1.000E+03

[: INPut] : VOLTage : RANGE : { SIGMA | SIGMB }

機能 結線ユニット{ΣA|ΣB}に属するすべてのエレメントの電圧レンジを一括設定します。

構文 **[: INPut] : VOLTage : RANGE : { SIGMA | SIGMB } { <電圧> }**
 ・ クレストファクタを「3」に設定しているとき
 <電圧> = 15, 30, 60, 100, 150, 300, 600, 1000(V)
 ・ クレストファクタを「6」に設定しているとき
 <電圧> = 7.5, 15, 30, 50, 75, 150, 300, 500(V)

例 **: INPUT:VOLTAGE:RANGE:SIGMA 1000V**

解説
 ・ **[: INPut] : VOLTage : RANGE : SIGMA**は、2~4エレメントモデルのみ有効です。
 ・ **[: INPut] : VOLTage : RANGE : SIGMB**は、4エレメントモデルのみ有効です。
 ・ 結線方式の設定(**[: INPut] : WIRing**)により結線ユニット{ΣA|ΣB}が存在しないときは、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|NONE)] [, (P1W2|NONE)] }**
[: INPut] : WIRing?
 P1W2=単相2線式[1P2W]
 P1W3=単相3線式[1P3W]
 P3W3=三相3線式[3P3W]
 P3W4=三相4線式[3P4W]
 V3A3=三相3線(3電圧3電流計法)[3P3W(3V3A)]
 NONE=結線なし

例
 ・ 4エレメントモデルのときの例
: INPUT:WIRING P1W2,P1W2,P1W2,P1W2
: INPUT:WIRING? -> : INPUT:
WIRING P1W2,P1W2,P1W2,P1W2

: INPUT:WIRING P1W3,P3W3

: INPUT:WIRING? ->

: INPUT:WIRING P1W3,P3W3

・ 3エレメントモデルのときの例

: INPUT:WIRING P3W3,P1W2

: INPUT:WIRING? -> : INPUT:

WIRING P3W3,P1W2

: INPUT:WIRING P3W4

: INPUT:WIRING? ->

: INPUT:WIRING P3W4

解説
 ・ 結線方式のパターンをエレメントの若い方から順に並べて設定します。
 ・ モデルタイプによっては、選択できない結線方式のパターンがあります。結線方式のパターンについては、ユーザーズマニュアル IM760301-01を参照してください。
 ・ 1エレメントモデルの場合は、P1W2固定です。それ以外の設定はできません。

6.15 INTEGrateグループ

INTEGrateグループは、積算に関するグループです。
フロントパネルのINTEGと同じ設定、および設定内容の問い合わせができます。

: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 :INTEGRATE? -> :INTEGRATE:
MODE NORMAL;ACAL 0;TIMER 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>}

:INTEGrate:ACAL?

例 :INTEGRATE:ACAL OFF
:INTEGRATE:ACAL? ->
:INTEGRATE:ACAL 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

例 :INTEGRATE:RESET

:INTEGrate:RTIME?

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>?

例 :INTEGRATE:RTIME? ->

:INTEGRATE:RTIME:

START 2005,1,1,0,0,0;

END 2005,1,1,1,0,0

:INTEGrate:RTIME:{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:{START|
END} {<NRf>,<NRf>,<NRf>,<NRf>,<NRf>,<NRf>}

:INTEGrate:RTIME:{START|END}?

{<NRf>,<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:RTIME:
START 2005,1,1,0,0,0
:INTEGRATE:RTIME:START? ->
:INTEGRATE:RTIME:
START 2005,1,1,0,0,0

:INTEGrate:START

機能 積算をスタートします。

構文 :INTEGrate:START

例 :INTEGRATE:START

:INTEGrate:STATE?

機能 積算状態を問い合わせます。

構文 :INTEGrate:STATE?

例 :INTEGRATE:STATE? -> RESET

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)

TI梅up=積算タイマ時間による積算ストップ

:INTEGrate:STOP

機能 積算をストップします。

構文 :INTEGrate:STOP

例 :INTEGRATE:STOP

: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 {<NRf>,<NRf>,<NRf>}

:INTEGrate:TMer?

{<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:TIMER 1,0,0

:INTEGRATE:TIMER? ->

:INTEGRATE:TIMER 1,0,0

6.16 MEASureグループ

MEASureグループは、演算に関するグループです。

フロントパネルのMEASURE、AVG、ITEMの「Frequency Meas. Item」メニュー、WIRINGの「 η Formula」、
「Compensation」、 Δ Measure」の各メニューと同じ設定、および設定内容の問い合わせができます。

:MEASure?

機能 演算に関するすべての設定値を問い合わせます。

構文 :MEASure?

例 :MEASURE? -> :MEASURE: AVERAGING:
STATE 0; TYPE EXPONENT; COUNT 2; :
MEASURE: FREQUENCY: ITEM1 U1;
ITEM2 I1; :MEASURE: SAMPLING AUTO;
SQFORMULA TYPE1; PC: IEC 1976;
P1 0.5000; P2 0.5000; :MEASURE:
EFFICIENCY: ETA1 PB, PA; ETA2 PA, PB;
ETA3 OFF; ETA4 OFF; UDEF1 P1;
UDEF2 P1; :MEASURE: FUNCTION1:
STATE 0; EXPRESSION "UMN(E1)";
UNIT "V"; :MEASURE: FUNCTION2:
STATE 0; EXPRESSION "UMN(E2)";
UNIT "V"; :MEASURE: FUNCTION3:
STATE 0; EXPRESSION "UMN(E3)";
UNIT "V"; :MEASURE: FUNCTION4:
STATE 0; EXPRESSION "UMN(E4)";
UNIT "V"; :MEASURE: FUNCTION5:
STATE 0; EXPRESSION "U(E1, ORT)";
UNIT "V"; :MEASURE: FUNCTION6:
STATE 0; EXPRESSION "I(E1, ORT)";
UNIT "A"; :MEASURE: FUNCTION7:
STATE 0; EXPRESSION "U(E1, ORT)";
UNIT "V"; :MEASURE: FUNCTION8:
STATE 0; EXPRESSION "I(E1, ORT)";
UNIT "A"; :MEASURE: FUNCTION9:
STATE 0; EXPRESSION "U(E1, ORT)";
UNIT "V"; :MEASURE: FUNCTION10:
STATE 0; EXPRESSION "I(E1, ORT)";
UNIT "A"; :MEASURE: FUNCTION11:
STATE 0; EXPRESSION "U(E1, ORT)";
UNIT "V"; :MEASURE: FUNCTION12:
STATE 0; EXPRESSION "I(E1, ORT)";
UNIT "A"; :MEASURE: FUNCTION13:
STATE 0; EXPRESSION "U(E1, ORT)";
UNIT "V"; :MEASURE: FUNCTION14:
STATE 0; EXPRESSION "I(E1, ORT)";
UNIT "A"; :MEASURE: FUNCTION15:
STATE 0; EXPRESSION "U(E1, ORT)";
UNIT "V"; :MEASURE: FUNCTION16:
STATE 0; EXPRESSION "I(E1, ORT)";
UNIT "A"; :MEASURE: FUNCTION17:
STATE 0; EXPRESSION "U(E1, ORT)";
UNIT "V"; :MEASURE: FUNCTION18:
STATE 0; EXPRESSION "I(E1, ORT)";
UNIT "A"; :MEASURE: FUNCTION19:

```
STATE 0; EXPRESSION "U(E1, ORT)";  
UNIT "V"; :MEASURE: FUNCTION20:  
STATE 0; EXPRESSION "I(E1, ORT)";  
UNIT "A"; :MEASURE: PHASE 180;  
SYNCHRONIZE MASTER; MHOLD 0;  
COMPENSATION: WIRING: ELEMENT1 OFF;  
ELEMENT2 OFF; ELEMENT3 OFF;  
ELEMENT4 OFF; :MEASURE: COMPENSATION:  
EFFICIENCY 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 :MEASURE: AVERAGING? ->
:MEASURE: AVERAGING: STATE 0;
TYPE EXPONENT; COUNT 2

:MEASure: AVERaging: COUNT

機能 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のとき、減衰定数)
<Nrf>=8, 16, 32, 64, 128, 256
(TYPE=LINearのとき、移動平均個数)

例 :MEASURE: AVERAGING: COUNT 2
:MEASURE: AVERAGING: COUNT? ->
:MEASURE: AVERAGING: COUNT 2

解説 高調波測定(オプション)の測定ファンクション
に対するアベレーシングは、TYPE=EXPonent
のとき(減衰定数)のみ有効です。詳しくは、
ユーザーズマニュアルIM760301-01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>}

例 :MEASure: AVERaging: STATE?
:MEASURE: AVERAGING: STATE ON
:MEASURE: AVERAGING: STATE? ->
:MEASURE: AVERAGING: STATE 1

:MEASure:AVERaging:TYPE

機能	アベレーシングのタイプを設定/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	高調波測定(オプション)の測定ファンクションに対するアベレーシングは「EXPonent」のみ有効です。詳しくは、ユーザーズマニュアル IM760301-01を参照してください。

:MEASure:COMPensation?

機能	補正演算に関するすべての設定値を問い合わせします。
構文	:MEASure:COMPensation?
例	:MEASURE:COMPENSATION? -> :MEASURE:COMPENSATION:WIRING: ELEMENT1 OFF;ELEMENT2 OFF; ELEMENT3 OFF;ELEMENT4 OFF;;MEASURE: COMPENSATION:EFFICIENCY 0

:MEASure:COMPensation:EFFiciency

機能	効率補正のON/OFFを設定/問い合わせします。
構文	:MEASure:COMPensation: EFFiciency {<Boolean>}
例	:MEASURE:COMPENSATION:EFFICIENCY ON :MEASURE:COMPENSATION:EFFICIENCY? - > :MEASURE:COMPENSATION: EFFICIENCY 1

:MEASure:COMPensation:V3A3

機能	2電力計法の補正のON/OFFを設定/問い合わせします。
構文	:MEASure:COMPensation: V3A3 {<Boolean>}
例	:MEASURE:COMPENSATION:V3A3 ON :MEASURE:COMPENSATION:V3A3? -> :MEASURE:COMPENSATION:V3A3 1
解説	・3～4エレメントモデルのみ有効です。 ・デルタ演算機能(オプション, /DT)搭載時のみ有効です。 ・結線方式([:INPut]:WIRing)に「V3A3」が設定されているときに有効な設定です。

:MEASure:COMPensation:WIRing?

機能	結線補正に関するすべての設定値を問い合わせします。
構文	:MEASure:COMPensation:WIRing?
例	:MEASURE:COMPENSATION:WIRING? -> :MEASURE:COMPENSATION:WIRING: ELEMENT1 OFF;ELEMENT2 OFF; ELEMENT3 OFF;ELEMENT4 OFF

:MEASure:COMPensation:WIRing:ELEment<x>

機能	各エレメントの結線補正を設定/問い合わせします。
構文	:MEASure:COMPensation:WIRing: ELEment<x> {OFF U_I I_U}
例	:MEASure:COMPensation:WIRing: ELEment<x>? <x>=1~4(エレメント) :MEASURE:COMPENSATION:WIRING: ELEMENT1 OFF :MEASURE:COMPENSATION:WIRING: ELEMENT1? -> :MEASURE:COMPENSATION: WIRING:ELEMENT1 OFF

: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します。
構文	:MEASure:DMeasure?
例	:MEASURE:DMEASURE? -> :MEASURE:DMEASURE:SIGMA OFF; SIGMB OFF
解説	デルタ演算機能(オプション, /DT)搭載時のみ有効です。

:MEASure:DMeasure[:SIGMA]

機能	結線ユニットΣAに対するデルタ演算モードを設定/問い合わせします。
構文	:MEASure:DMeasure[:SIGMA] {OFF DIFFerence P3W3_V3A3 ST_DT DT_ST}
例	:MEASure:DMeasure:SIGMA? :MEASURE:DMEASURE:SIGMA OFF :MEASURE:DMEASURE:SIGMA? -> :MEASURE:DMEASURE:SIGMA OFF
解説	・デルタ演算機能(オプション, /DT)搭載時のみ有効です。 ・選択肢の内容はそれぞれ次のとおりです。結線ユニットΣAの結線方式によって、選択できるものが決まります。 - OFF=デルタ演算なし(単相2線式(1P2W)のみ) - DIFFerence=差動電圧, 差動電流(単相3線式(1P3W), 三相3線式(3P3W)のみ) - P3W3_V3A3=3P3W->3V3A変換(単相3線式(1P3W), 三相3線式(3P3W)のみ) - ST_DT=Star->Delta変換(三相4線式(3P4W)のみ) - DT_ST=Delta->Star変換(三相3線(3電圧3電流計法)[3P3W(3V3A)]のみ)

:MEASure:DMeasure:SIGMB

機能	結線ユニットΣBに対するデルタ演算モードを設定/問い合わせします。
構文	:MEASure:DMeasure:SIGMB {OFF DIFFerence P3W3_V3A3 ST_DT DT_ST} :MEASure:DMeasure:SIGMB?
例	:MEASURE:DMEASURE:SIGMB OFF :MEASURE:DMEASURE:SIGMB? -> :MEASURE:DMEASURE:SIGMB OFF
解説	・デルタ演算機能(オプション, /DT)搭載時かつ4エレメントモデルのみ有効です。 ・選択肢の内容は 「 :MEASure:DMeasure[:SIGMA] 」コマンドと同じです。結線ユニットΣBの結線方式によって、選択できるものが決まります。

:MEASure:EFFiciency?

機能	効率の演算に関するすべての設定値を問い合わせます。
構文	:MEASure:EFFiciency?
例	:MEASURE:EFFICIENCY? -> :MEASURE:EFFICIENCY:ETA1 PB,PA; ETA2 PA,PB;ETA3 OFF;ETA4 OFF; UDEF1 P1;UDEF2 P1

:MEASure:EFFiciency:ETA<x>

機能	効率の演算式を設定/問い合わせします。
構文	:MEASure:EFFiciency:ETA<x> {(OFF P<x> PA PB PM UDEF<x>)[,(P<x> PA PB PM UDEF<x>)]} :MEASure:EFFiciency:ETA<x>? ETA<x>の<x>=1~4(η1~η4) OFF=演算なし(分母は無視されます) P<x>の<x>=1~4(エレメント) PA=PΣA(2~4エレメントモデルのみ) PB=PΣB(4エレメントモデルのみ) PM=Pm(モータ出力, モータバージョン(-MV)のみ) UDEF<x>の<x>=1~2(Udef1~Udef2)
例	:MEASURE:EFFICIENCY:ETA1 PB,PA :MEASURE:EFFICIENCY:ETA1? -> :MEASURE:EFFICIENCY:ETA1 PB,PA
解説	・分子, 分母の順で設定します。 ・分母は省略することができます。省略したときの分母は「1」となります。 ・問い合わせの応答では, 分母が「1」のときは省略されます。

:MEASure:EFFiciency:UDEF<x>

機能	効率の演算式で使用するユーザー定義パラメータを設定/問い合わせします。
構文	:MEASure:EFFiciency:UDEF<x> {(NONE P<x> PA PB PM)[,(NONE P<x> PA PB PM)][(NONE P<x> PA PB PM)][(NONE P<x> PA PB PM)]} :MEASure:EFFiciency:UDEF<x>? UDEF<x>の<x>=1~2(Udef1~Udef2) NONE=演算項なし P<x>の<x>=1~4(エレメント) PA=PΣA(2~4エレメントモデルのみ) PB=PΣB(4エレメントモデルのみ) PM=Pm(モータ出力, モータバージョン(-MV)のみ)
例	:MEASURE:EFFICIENCY:UDEF1 P1,P2,P3 :MEASURE:EFFICIENCY:UDEF1? -> :MEASURE:EFFICIENCY:UDEF1 P1,P2,P3
解説	・第1項, 第2項, 第3項, 第4項の順で設定します。 ・第2~4項は省略することができます。省略項は「NONE」となります。 ・問い合わせの応答では, 第2~4項については, それ以降の項がすべて「NONE」のときは省略されます。

: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:ITEM1 U1; ITEM2 I1
解説	周波数測定追加(オプション, /FQ)のときは, 全入力エレメントの同時測定が可能のため, このコマンドは無効です。

:MEASure:FREquency:ITEM<x>

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<x> {U<x> I<x>} :MEASure:FREquency:ITEM<x>? ITEM<x>の<x>=1~2(Freq.1~Freq.2) U<x>, I<x>の<x>=1~4(エレメント)
例	:MEASURE:FREQUENCY:ITEM1 U1 :MEASURE:FREQUENCY:ITEM1? -> :MEASURE:FREQUENCY:ITEM1 U1
解説	周波数測定追加(オプション, /FQ)のときは, 全入力エレメントの同時測定が可能のため, このコマンドは無効です。

:MEASure:FUNCTION<x>?

機能 ユーザー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>=1~20(F1~F20)

例 :MEASURE:FUNCTION1? ->

:MEASURE:FUNCTION1:STATE 1;
EXPRESSION "UMN(E1)";UNIT "V"

:MEASure:FUNCTION<x>:EXPRESSION

機能 ユーザー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>=1~20(F1~F20)

<文字列>=50文字以内

例 :MEASURE:FUNCTION1:

EXPRESSION "UMN(E1)"

:MEASURE:FUNCTION1:EXPRESSION? ->

:MEASURE:FUNCTION1:

EXPRESSION "UMN(E1)"

:MEASure:FUNCTION<x>[:STATE]

機能 ユーザー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>=1~20(F1~F20)

例 :MEASURE:FUNCTION1:STATE ON

:MEASURE:FUNCTION1:STATE? ->

:MEASURE:FUNCTION1:STATE 1

:MEASure:FUNCTION<x>:UNIT

機能 ユーザー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>=1~20(F1~F20)

<文字列>=8文字以内

例 :MEASURE:FUNCTION1:UNIT "V"

:MEASURE:FUNCTION1:UNIT? ->

:MEASURE:FUNCTION1:UNIT "V"

解説 演算結果に影響を及ぼすことはありません。

:MEASure:MHOLD

機能 ユーザー定義ファンクションで使用するMAX HOLD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:MHOLD {<Boolean>}

:MEASure:MHOLD?

例 :MEASURE:MHOLD ON

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

解説

- ・ユーザー定義ファンクションでMAX HOLDファンクションを指定し、
:MEASure:MHOLDを「ON」にした時点からMAX HOLD動作を開始します。
- ・:MEASure:MHOLDを「OFF」にするとMAX HOLD動作を終了し、MAX HOLD値は「データ無し」となります。
- ・:MEASure:MHOLDを「ON」にした状態で「ON」を設定すると、MAX HOLD値を一旦リセットし、再度MAX HOLD動作を開始します。
- ・MAX HOLDファンクションの指定方法については、ユーザーズマニュアルIM760301-01を参照してください。

:MEASure:PC?

機能 Pc(Corrected Power)の演算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

:MEASure:PC:IEC

機能 Pc(Corrected Power)の演算式を設定/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 IEC76-1で記述されているPc算出式の発行年を指定します。

:MEASure:PC:P<x>

機能 Pc(Corrected Power)演算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(P1, P2)

<NRf>=0.0001~9.9999

例 :MEASURE:PC:P1 0.5

:MEASURE:PC:P1? ->

:MEASURE:PC:P1 0.5000

解説 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。

6.16 MEASureグループ

: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 「180」のときは±0～180° (Lead/Lag)で、
「360」のときは0～360° で表示します。

:MEASure:SAMPling

機能 サンプリング周波数を設定/問い合わせします。

構文 **:MEASure:SAMPling** {AUTO|CLKA|CLKB|CLKC}

:MEASure:SAMPling?

例 **:MEASURE:SAMPLING** AUTO

:MEASURE:SAMPLING? ->

:MEASURE:SAMPLING AUTO

解説 {AUTO|CLKA|CLKB|CLKC}に対応するサンプリング周波数については、ユーザーズマニュアルIM760301-01を参照してください。

:MEASure:SQFormula

機能 S(皮相電力), Q(無効電力)の演算式を設定/問い合わせします。

構文 **:MEASure:SQFormula** {TYPE1|TYPE2|TYPE3}

:MEASure:SQFormula?

例 **:MEASURE:SQFORMULA** TYPE1

:MEASURE:SQFORMULA? ->

:MEASURE:SQFORMULA TYPE1

解説 ・ {TYPE1|TYPE2|TYPE3}に対応する演算式については、ユーザーズマニュアルIM760301-01を参照してください。
・ 「TYPE3」は高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時のみ選択可能です。

:MEASure:SYNChronize

機能 同期測定モードを設定/問い合わせします。

構文 **:MEASure:SYNChronize** {MASTER|SLAVE}

:MEASure:SYNChronize?

例 **:MEASURE:SYNCHRONIZE** MASTER

:MEASURE:SYNCHRONIZE? ->

:MEASURE:SYNCHRONIZE MASTER

6.17 MOTorグループ

MOTorグループは、モータ評価機能に関するグループです。
フロントパネルのMOTOR SET(SHIFT+SCALING)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、このグループのコマンドは、モータバージョン(-MV)のみ有効です。

:MOTor?

機能 モータ評価機能に関するすべての設定値を問い合わせます。

構文 :MOTor?

例 :MOTOR? -> :MOTOR:SPEED:
TYPE ANALOG;RANGE 20.0E+00;AUTO 0;
SCALING 1.0000;UNIT "rpm";:MOTOR:
TORQUE:TYPE ANALOG;RANGE 20.0E+00;
AUTO 0;SCALING 1.0000;UNIT "Nm";:
MOTOR:PM:SCALING 1.0000;UNIT "W";:
MOTOR:FILTER:LINE OFF;:MOTOR:
SYNCHRONIZE NONE;POLE 2;SSPEED I1

:MOTor:FILTer?

機能 入力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 :MOTor:FILTer?

例 :MOTOR:FILTER? ->
:MOTOR:FILTER:LINE OFF

:MOTor:FILTer[:LINE]

機能 ラインフィルタを設定/問い合わせします。

構文 :MOTor:FILTer[:LINE] {OFF|<周波数>}
:MOTor:FILTer:LINE?
OFF=ラインフィルタOFF
<周波数>=100Hz, 50kHz(ラインフィルタ
ON, カットオフ周波数)

例 :MOTOR:FILTER:LINE OFF
:MOTOR:FILTER:LINE? ->
:MOTOR:FILTER:LINE OFF

:MOTor:PM?

機能 モータ出力(Pm)に関するすべての設定値を問い合わせます。

構文 :MOTor:PM?

例 :MOTOR:PM? -> :MOTOR:PM:
SCALING 1.0000;UNIT "W"

:MOTor:PM:SCALing

機能 モータ出力演算のスケーリング係数を設定/問い合わせします。

構文 :MOTor:PM:SCALing {<Nrf>}
:MOTor:PM:SCALing?
<Nrf>=0.0001~99999.9999

例 :MOTOR:PM:SCALING 1
:MOTOR:PM:SCALING? ->
:MOTOR:PM:SCALING 1.0000

:MOTor:PM:UNIT

機能 モータ出力演算結果に付加する単位を設定/問い合わせします。

構文 :MOTor:PM:UNIT {<文字列>}
:MOTor:PM:UNIT?
<文字列>=8文字以内

例 :MOTOR:PM:UNIT "W"
:MOTOR:PM:UNIT? ->
:MOTOR:PM:UNIT "W"

解説 演算結果に影響を及ぼすことはありません。

:MOTor:POLE

機能 モータの極数を設定/問い合わせします。

構文 :MOTor:POLE {<Nrf>}
:MOTor:POLE?
<Nrf>=1~99
例 :MOTOR:POLE 2
:MOTOR:POLE? -> :MOTOR:POLE 2

:MOTor:SPEEd?

機能 回転速度(Speed)に関するすべての設定値を問い合わせます。

構文 :MOTor:SPEEd?
例 :MOTOR:SPEED? -> :MOTOR:SPEED:
TYPE ANALOG;RANGE 20.0E+00;AUTO 0;
SCALING 1.0000;UNIT "rpm"

:MOTor:SPEEd:AUTO

機能 回転信号入力(アナログ入力方式)の電圧オートレンジON/OFFを設定/問い合わせします。

構文 :MOTor:SPEEd:AUTO {<Boolean>}
:MOTor:SPEEd:AUTO?

例 :MOTOR:SPEED:AUTO ON
:MOTOR:SPEED:AUTO? ->
:MOTOR:SPEED:AUTO 1

解説 回転信号入力タイプ(:MOTor:SPEEd:TYPE)が「ANALog(アナログ入力)」のときに有効な設定です。

6.17 MOTorグループ

:MOTor:SPEEd:PRANge

機能	回転速度(パルス入力方式)のレンジを設定/問い合わせします。
構文	:MOTor:SPEEd:PRANge {<NRf>,<NRf>} :MOTor:SPEEd:PRANge? <NRf>=0.0000~99999.9999
例	:MOTOR:SPEED:PRANGE 10000,0 :MOTOR:SPEED:PRANGE? -> :MOTOR:SPEED: PRANGE 10000.0000,0.0000
解説	・ 上限値, 下限値の順で設定します。 ・ 回転信号入力タイプ(:MOTor:SPEEd:TYPE)が「PULSe(パルス入力)」のときに有効な設定です。

:MOTor:SPEEd:PULSe

機能	回転信号入力(パルス入力)のパルス数を設定/問い合わせします。
構文	:MOTor:SPEEd:PULSe {<NRf>} :MOTor:SPEEd:PULSe? <NRf>=1~9999
例	:MOTOR:SPEED:PULSE 60 :MOTOR:SPEED:PULSE? -> :MOTOR:SPEED:PULSE 60
解説	回転信号入力タイプ(:MOTor:SPEEd:TYPE)が「PULSe(パルス入力)」のときに有効な設定です。

:MOTor:SPEEd:RANGe

機能	回転信号入力(アナログ入力方式)の電圧レンジを設定/問い合わせします。
構文	:MOTor:SPEEd:RANGe {<電圧>} :MOTor:SPEEd:RANGe? <電圧>=1, 2, 5, 10, 20(V)
例	:MOTOR:SPEED:RANGE 20V :MOTOR:SPEED:RANGE? -> :MOTOR:SPEED:RANGE 20.0E+00
解説	回転信号入力タイプ(:MOTor:SPEEd:TYPE)が「ANALog(アナログ入力)」のときに有効な設定です。

:MOTor:SPEEd:SCALing

機能	回転速度演算のスケール係数を設定/問い合わせします。
構文	:MOTor:SPEEd:SCALing {<NRf>} :MOTor:SPEEd:SCALing? <NRf>=0.0001~99999.9999
例	:MOTOR:SPEED:SCALING 1 :MOTOR:SPEED:SCALING? -> :MOTOR:SPEED:SCALING 1.0000

:MOTor:SPEEd:TYPE

機能	回転信号入力の入力タイプを設定/問い合わせします。
構文	:MOTor:SPEEd:TYPE {ANALog PULSe} :MOTor:SPEEd:TYPE?
例	:MOTOR:SPEED:TYPE ANALOG :MOTOR:SPEED:TYPE? -> :MOTOR:SPEED:TYPE ANALOG

:MOTor:SPEEd:UNIT

機能	回転速度演算結果に付加する単位を設定/問い合わせします。
構文	:MOTor:SPEEd:UNIT {<文字列>} :MOTor:SPEEd:UNIT? <文字列>=8文字以内
例	:MOTOR:SPEED:UNIT "rpm" :MOTOR:SPEED:UNIT? -> :MOTOR:SPEED:UNIT "rpm"
解説	演算結果に影響を及ぼすことはありません。

:MOTor:SSPeEd(Sync Speed source)

機能	同期速度(SyncSp)演算のための周波数測定ソースを設定/問い合わせします。
構文	:MOTor:SSPeEd {U<x> I<x>} :MOTor:SSPeEd? <x>=1~4(エレメント)
例	:MOTOR:SSPEED I1 :MOTOR:SSPEED? -> :MOTOR:SSPEED I1

:MOTor:SYNChronize

機能	回転速度(Speed)/トルク(Torque)演算のための同期ソースを設定/問い合わせします。
構文	:MOTor:SYNChronize {U<x> I<x> EXTErnal NONE} :MOTor:SYNChronize? <x>=1~4(エレメント) EXTErnal=外部クロック入力(Ext Clk) NONE=同期ソースなし
例	:MOTOR:SYNCHRONIZE NONE :MOTOR:SYNCHRONIZE? -> :MOTOR:SYNCHRONIZE NONE

:MOTor:TORQue?

機能	トルク(Torque)に関するすべての設定値を問い合わせます。
構文	:MOTor:TORQue?
例	:MOTOR:TORQUE? -> :MOTOR:TORQUE: TYPE ANALOG;RANGE 20.0E+00;AUTO 0; SCALING 1.0000;UNIT "Nm"

:MOTor:TORQue:AUTO

機能 トルク信号入力(アナログ入力方式)の電圧オー
トレンジON/OFFを設定/問い合わせします。

構文 :MOTor:TORQue:AUTO {<Boolean>}
:MOTor:TORQue:AUTO?

例 :MOTOR:TORQUE:AUTO ON
:MOTOR:TORQUE:AUTO? ->
:MOTOR:TORQUE:AUTO 1

解説 トルク信号入力タイプ(:MOTor:TORQue:
TYPE)が「ANALog(アナログ入力)」のときに
有効な設定です。

:MOTor:TORQue:PRANge

機能 トルク(パルス入力方式)のレンジを設定/問い合
わせします。

構文 :MOTor:TORQue:PRANge {<NRf>,<NRf>}
:MOTor:TORQue:PRANge?
<NRf>=-10000.0000~10000.0000

例 :MOTOR:TORQUE:PRANGE 50,-50
:MOTOR:TORQUE:PRANGE? ->
:MOTOR:TORQUE:

PRANGE 50.0000,-50.0000
解説 ・ 上限値, 下限値の順で設定します。
・ トルク信号入力タイプ(:MOTor:TORQue:
TYPE)が「PULSe(パルス入力)」のときに有
効な設定です。

:MOTor:TORQue:RANge

機能 トルク信号入力(アナログ入力方式)の電圧レン
ジを設定/問い合わせします。

構文 :MOTor:TORQue:RANge {<電圧>}
:MOTor:TORQue:RANge?
<電圧>=1, 2, 5, 10, 20(V)

例 :MOTOR:TORQUE:RANGE 20V
:MOTOR:TORQUE:RANGE? ->
:MOTOR:TORQUE:RANGE 20.0E+00

解説 トルク信号入力タイプ(:MOTor:TORQue:
TYPE)が「ANALog(アナログ入力)」のときに
有効な設定です。

:MOTor:TORQue:RATE?

機能 トルク信号(パルス入力方式)の定格値に関する
すべての設定値を問い合わせます。

構文 :MOTor:TORQue:RATE?
例 :MOTOR:TORQUE:RATE? ->
:MOTOR:TORQUE:RATE:
UPPER 50.0000,15.000E+03;
LOWER -50.0000,5.000E+03

:MOTor:TORQue:RATE:{UPPer|LOWer}

機能 トルク信号(パルス入力方式)の{上限|下限}定格
値を設定/問い合わせします。

構文 :MOTor:TORQue:RATE:{UPPer|
LOWer} {<NRf>,<周波数>}
<NRf>=-10000.0000~10000.0000
<周波数>=1Hz~100MHz

例 :MOTOR:TORQUE:RATE:UPPER 50,15kHz
:MOTOR:TORQUE:RATE:UPPER? ->
:MOTOR:TORQUE:RATE:UPPER
50.0000,15.000E+03

解説 トルク信号入力タイプ(:MOTor:TORQue:
TYPE)が「PULSe(パルス入力)」のときに有効
な設定です。

:MOTor:TORQue:SCALing

機能 トルク演算のスケーリング係数を設定/問い合
わせします。

構文 :MOTor:TORQue:SCALing {<NRf>}
:MOTor:TORQue:SCALing?
<NRf>=0.0001~99999.9999

例 :MOTOR:TORQUE:SCALING 1
:MOTOR:TORQUE:SCALING? ->
:MOTOR:TORQUE:SCALING 1.0000

:MOTor:TORQue:TYPE

機能 トルク信号入力の入力タイプを設定/問い合
わせます。

構文 :MOTor:TORQue:TYPE {ANALog|PULSe}
:MOTor:TORQue:TYPE?
例 :MOTOR:TORQUE:TYPE ANALOG
:MOTOR:TORQUE:TYPE? ->
:MOTOR:TORQUE:TYPE ANALOG

:MOTor:TORQue:UNIT

機能 トルク演算結果に付加する単位を設定/問い合
わせします。

構文 :MOTor:TORQue:UNIT {<文字列>}
:MOTor:TORQue:UNIT?
<文字列>=8文字以内

例 :MOTOR:TORQUE:UNIT "Nm"
:MOTOR:TORQUE:UNIT? ->
:MOTOR:TORQUE:UNIT "Nm"

解説 演算結果に影響を及ぼすことはありません。

6.18 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。フロントパネルのNUMERICキーと同じ設定、および設定内容の問い合わせはDISPlayグループのコマンドを使用します。

: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 :NUMERIC? -> :NUMERIC:FORMAT ASCII;
NORMAL:NUMBER 15;ITEM1 U,1,TOTAL;
ITEM2 I,1,TOTAL;ITEM3 P,1,TOTAL;
ITEM4 S,1,TOTAL;ITEM5 Q,1,TOTAL;
ITEM6 LAMBDA,1,TOTAL;
ITEM7 PHI,1,TOTAL;ITEM8 FU,1;
ITEM9 FI,1;ITEM10 UPPEAK,1;
ITEM11 UMPEAK,1;ITEM12 IPPEAK,1;
ITEM13 IMPEAK,1;ITEM14 CFU,1;
ITEM15 CFI,1;:NUMERIC:HOLD 0

:NUMeric:CBCycle?

機能 Cycle by Cycle測定の数値リスト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:CBCycle?

例 :NUMERIC:CBCYCLE? ->
:NUMERIC:CBCYCLE:ITEM U,1;
START 1;END 100

解説 Cycle by Cycle測定機能(オプション、/CC)搭載時だけ有効です。

:NUMeric:CBCycle:END

機能 「:NUMeric:CBCycle:VALue?」で出力される数値リストデータの出力終了サイクルを設定/問い合わせします。

構文 :NUMeric:CBCycle:END {<NRf>}
:NUMeric:CBCycle:END?

<NRf>= 1~3000(サイクル番号)

例 :NUMERIC:CBCYCLE:END 100
:NUMERIC:CBCYCLE:END ->
:NUMERIC:CBCYCLE:END 100

解説 Cycle by Cycle測定機能(オプション、/CC)搭載時だけ有効です。

:NUMeric:CBCycle:ITEM

機能 Cycle by Cycle測定の数値リストデータの出力項目(ファンクション・エレメント)を設定/問い合わせします。

構文 :NUMeric:CBCycle:ITEM {<Function>,
<Element>}
:NUMeric:CBCycle:ITEM?
<Function>={FREQ|U|I|P|S|Q|LAMBda|
SPEEd|TORQue|PM|PKU|PKI|PKSpeed|
PKTorque}
<Element>={<NRf>|SIGMA|SIGMB}<NRf>
=1~4)

例 :NUMERIC:CBCYCLE:ITEM U,1
:NUMERIC:CBCYCLE:ITEM? ->
:NUMERIC:CBCYCLE:ITEM U,1

解説
・ Cycle by Cycle機能(オプション、/CC)搭載時だけ有効です。
・ <Function>={FREQ|SPEEd|TORQue|PM|PKSpeed|PKTorque}のとき、<Element>の指定は不要です。応答は<Element>が省略されます。
・ <Element>を省略したときは、エレメント1が設定されます。
・ {SPEEd|TORQue|PM|PKSpeed|PKTorque}は、モータバージョン(-MV)のみ有効です。

:NUMeric:CBCycle:START

機能 「:NUMeric:CBCycle:VALue?」で出力される数値リストデータの出力開始サイクルを設定/問い合わせします。

構文 :NUMeric:CBCycle:START {<NRf>}
:NUMeric:CBCycle:START?
<NRf>= 1~3000(サイクル番号)

例 :NUMERIC:CBCYCLE:START 1
:NUMERIC:CBCYCLE:START ->
:NUMERIC:CBCYCLE:START 1

解説 Cycle by Cycle測定機能(オプション、/CC)搭載時だけ有効です。

:NUMeric:CBcYcle:VALue?

機能 Cycle by Cycle測定の数値リストデータを問い合わせます。

構文 **:NUMeric:CBcYcle:VALue?**
 {<Function>,<Element>}
:NUMeric:CBcYcle:VALue?
 <Function>={FREQ|U|I|P|S|Q|LAMBda|
 SPEEd|TORQue|PM|PKU|PKI|PKSPeEd|
 PKTorque}
 <Element>={<NRf>|SIGMA|SIGMB}<NRf>
 =1~4)

例

- ・パラメータを指定した場合の例
 (:NUMeric:CBcYcle:STARt 1;END 100
 の場合)
:NUMERIC:CBcYCLE:VALUE? U,1 ->
 100.001E+00,100.002E+00,
 100.003E+00,100.004E+00,
 ..(中略)..,100.099E+00,100.100E+00
- ・パラメータを省略した場合の例
 (:NUMeric:CBcYcle:ITEM U,1の場合)
:NUMERIC:CBcYCLE:VALUE? ->
 100.001E+00,100.002E+00,
 100.003E+00,100.004E+00,
 ..(中略)..,100.099E+00,100.100E+00
- ・「:NUMeric:FORMat」の設定が「FLOat」
 の場合の例
**:NUMERIC:CBcYCLE:VALUE? -> #6(6桁
 のバイト数)(データバイトの並び)**

解説

- ・Cycle by Cycle機能(オプション、/CC)搭載
 時だけ有効です。
- ・数値リストデータは、「:NUMeric:
 CBcYcle:STARt」から「:NUMeric:
 CBcYcle:END」までの最大3000個の数値
 データで構成されています。
- ・パラメータを指定すると、その項目の数値リ
 ストデータを出力します。<Function>および
 <Element>の内容については「:NUMeric:
 CBcYcle:ITEM」の解説をご覧ください。
- ・パラメータを省略した場合は「:NUMeric:
 CBcYcle:ITEM」で設定されている項目の数
 値リストデータを出力します。
- ・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」(6-92ページ)を参照してく
 ださい。

:NUMeric:FLICKer?

機能 フリッカ測定の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:FLICKer?**
 例 **:NUMERIC:FLICKER? -> :NUMERIC:
 FLICKER:FUNCTION:NUMBER 8;
 ITEM1 TIME;ITEM2 UN,1;ITEM3 FU,1;
 ITEM4 DC,1,CURRENT;ITEM5 DMAX,1,
 CURRENT;ITEM6 DT,1,CURRENT;
 ITEM7 PST,1,CURRENT;ITEM8 PLT,1;;
 NUMERIC:FLICKER:INFORMATION:
 NUMBER 7;ITEM1 JTOTAL,TOTAL;
 ITEM2 JDC,1,ALL;ITEM3 JDMAX,1,ALL;
 ITEM4 JDT,1,ALL;ITEM5 JPST,1,ALL;
 ITEM6 JPLT,1;ITEM7 JTOTAL,1**

解説 フリッカ測定機能(オプション、/FL)搭載時だけ
 有効です。

:NUMeric:FLICKer:COUNT?

機能 設定した観測期間で、フリッカ測定が完了した
 回数を問い合わせます。

構文 **:NUMeric:FLICKer:COUNT?**
 例 **:NUMERIC:FLICKER:COUNT? -> 0**
 解説

- ・フリッカ測定機能(オプション、/FL)搭載時
 だけ有効です。
- ・フリッカ測定表示画面の「Count」バーグラ
 フの右側に表示されている回数を返します。

:NUMeric:FLICKer:FUNCTION?

機能 フリッカ測定データ(可変フォーマット)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:FLICKer:FUNCTION?**
 例 **:NUMERIC:FLICKER:FUNCTION? ->
 :NUMERIC:FLICKER:FUNCTION:NUMBER 8;
 ITEM1 TIME;ITEM2 UN,1;ITEM3 FU,1;
 ITEM4 DC,1,CURRENT;ITEM5 DMAX,1,
 CURRENT;ITEM6 DT,1,CURRENT;
 ITEM7 PST,1,CURRENT;ITEM8 PLT,1**

解説

- ・フリッカ測定機能(オプション、/FL)搭載時
 だけ有効です。
- ・「:NUMeric:FLICKer:FUNCTION:
 ITEM<x>」に関する設定値については、
 「:NUMeric:FLICKer:FUNCTION:
 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:FLICKer:FUNCTION:CLEar

機能 フリッカ測定データ(可変フォーマット)の出力項目をクリア(「NONE」に設定)します。

構文 **:NUMeric:FLICKer:FUNCTION:CLEar** {ALL|<NRf>[,<NRf>]}
 ALL=すべての項目をクリア
 1つ目の<NRf>=1~32(クリアを開始する項目番号)
 2つ目の<NRf>=1~32(クリアを終了する項目番号)

例 **:NUMERIC:FLICKER:FUNCTION:CLEAR ALL**
 ・フリッカ測定機能(オプション、/FL)搭載時だけ有効です。
 ・2つ目の<NRf>を省略した場合、クリア開始番号から最後(32)までの出力項目をクリアします。

:NUMeric:FLICKer:FUNCTION:DELeTe

機能 フリッカ測定データ(可変フォーマット)の出力項目を削除します。

構文 **:NUMeric:FLICKer:FUNCTION:DELeTe** {<NRf>[,<NRf>]}
 1つ目の<NRf>=1~32(削除を開始する項目番号)
 2つ目の<NRf>=1~32(削除を終了する項目番号)

例 **:NUMERIC:FLICKER:FUNCTION:DELETE 1**
 (ITEM1を削除し、ITEM2以降を前へ詰める)
:NUMERIC:FLICKER:FUNCTION:DELETE 1,3
 (ITEM1~3を削除し、ITEM4以降を前へ詰める)

解説
 ・フリッカ測定機能(オプション、/FL)搭載時だけ有効です。
 ・削除された出力項目の位置には、それ以降の出力項目が順次詰められ、最後の空いた部分には「NONE」が設定されます。
 ・2つ目の<NRf>を省略した場合、削除開始番号の出力項目のみを削除します。

:NUMeric:FLICKer:FUNCTION:ITEM<x>

機能 フリッカ測定データ(可変フォーマット)の出力項目(ファンクション・エレメント・観測期間)を設定/問い合わせします。

構文 **:NUMeric:FLICKer:FUNCTION:ITEM<x>** {NONE|<Function>,<Element>[,<Period>]}
:NUMeric:FLICKer:FUNCTION:ITEM<x>?
 <x>=1~32(項目番号)
 NONE=出力項目なし
 <Function>={TIME|UN|FU|DC|DMAX|DT|PST|PLT}
 <Element>={<NRf>}(<NRf>=1~4)
 <Period>={CURRENT|<NRf>|ALL}(<NRf>=1~99)

例 **:NUMERIC:FLICKER:FUNCTION:ITEM1 DC,1,1**
:NUMERIC:FLICKER:FUNCTION:ITEM1? ->
:NUMERIC:FLICKER:FUNCTION:ITEM1 DC,1,1

解説
 ・フリッカ測定機能(オプション、/FL)搭載時だけ有効です。
 ・<Function>の選択肢に対する出力内容は次のとおりです。

<Function> 出力内容		<Element> 指定	<Period> 指定
TIME	測定経過時間 [sec] (画面右上の「Flicker:」横に表示される時間)	不要	不要
UN	電圧実効値(定格電圧) Un[V]	必要	不要
FU	電圧周波数 Freq[Hz]	必要	不要
DC	相対定常電圧変化 dc[%]	必要	必要
DMAX	最大相対電圧変化 dmax[%]	必要	必要
DT	相対電圧変化時間 dt[ms]	必要	必要
PST	短期間フリッカ値 Pst	必要	必要
PLT	長期間フリッカ値 Plt	必要	不要

- ・<Element>を省略したときは、エレメント1が設定されます。
- ・<Period>の選択肢の内容は次のとおりです。
 <Period>を省略したときは、CURRENTが設定されます。
 CURRENT 現在測定中の観測期間(フリッカ測定表示画面の数値リストの「*」の付いた行)
 測定完了(Complete)状態のときは、ALLと同じ
 ALL 観測期間全体(フリッカ測定表示画面の数値リストのResult行)
 <NRf>=1~99 指定された観測期間
- ・上記の表で、<Element>または<Period>の指定が不要な出力項目の応答は、<Element>または<Period>が省略されます。

:NUMeric:FLICKer:FUNCTion:NUMBER

- 機能 「:NUMeric:FLICKer:FUNCTion:VALue?」で送信されるフリッカ測定データの個数を設定/問い合わせします。
- 構文 **:NUMeric:FLICKer:FUNCTion:NUMBER {<NRf>|ALL}**
:NUMeric:FLICKer:FUNCTion:NUMBER? <NRf>=1~32(ALL)
- 例 **:NUMERIC:FLICKER:FUNCTION:NUMBER 8**
:NUMERIC:FLICKER:FUNCTION:NUMBER ->
:NUMERIC:FLICKER:FUNCTION:NUMBER 8
- 解説 ・フリッカ測定機能(オプション, /FL)搭載時だけ有効です。
 ・「:NUMeric:FLICKer:FUNCTion: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「8」が設定されています。

:NUMeric:FLICKer:FUNCTion:VALue?

- 機能 フリッカ測定データ(可変フォーマット)を問い合わせます。
- 構文 **:NUMeric:FLICKer:FUNCTion:VALue? {<NRf>}**
:NUMeric:FLICKer:FUNCTion:VALue? <NRf>=1~32(項目番号)
- 例 ・<NRf>を指定した場合の例
:NUMERIC:FLICKER:FUNCTION:VALUE? 4 -> 1.52E+00
 ・<NRf>を省略した場合の例
:NUMERIC:FLICKER:FUNCTION:VALUE? -> 600,229.75E+00, 50.000E+00,1.52E+00,1.56E+00, 3E+00,0.43E+00,0.17E+00
 ・「:NUMeric:FORMat」の設定が「FLOat」の場合の例
:NUMERIC:FLICKER:FUNCTION:VALUE? -> #4(4桁のバイト数)(データバイトの並び)
- 解説 ・フリッカ測定機能(オプション, /FL)搭載時だけ有効です。
 ・<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:FLICKer:FUNCTion: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は次のとおりです。

(1)正常時のデータ

- 測定経過時間(TIME)
 ASCII: 秒単位の<NR 1>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: 5桁, 指数部: 2桁, 例: 229.87E+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")
- 定常状態なし(表示: "Undef")
 ASCII: 「INF」(INFinity)
 FLOAT: 0x7E94F56A(9.9E+37)

:NUMeric:FLICKer:INFORMation?

- 機能 フリッカ判定結果(可変フォーマット)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:FLICKer:INFORMation?**
- 例 **:NUMERIC:FLICKER:INFORMATION? ->**
:NUMERIC:FLICKER:INFORMATION:
NUMBER 7;ITEM1 JTOTAL,TOTAL;
ITEM2 JDC,1,ALL;ITEM3 JDMAX,1,ALL;
ITEM4 JDT,1,ALL;ITEM5 JPST,1,ALL;
ITEM6 JPLT,1;ITEM7 JTOTAL,1
- 解説 ・フリッカ測定機能(オプション, /FL)搭載時だけ有効です。
 ・「:NUMeric:FLICKer:INFORMation:ITEM<x>」に関する設定値については, 「:NUMeric:FLICKer:INFORMation: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:FLICKer:INfOrMation:CLear

機能 フリッカ判定結果(可変フォーマット)の出力項目をクリア(「NONE」に設定)します。

構文 **:NUMeric:FLICKer:INfOrMation:CLear** {ALL|<NRf>[,<NRf>]}
 ALL=すべての項目をクリア
 1つ目の<NRf>=1~32(クリアを開始する項目番号)
 2つ目の<NRf>=1~32(クリアを終了する項目番号)

例 **:NUMERIC:FLICKER:INFORMATION: CLEAR ALL**

解説
 ・フリッカ測定機能(オプション, /FL)搭載時だけ有効です。
 ・2つ目の<NRf>を省略した場合, クリア開始番号から最後(32)までの出力項目をクリアします。

:NUMeric:FLICKer:INfOrMation:DELeTe

機能 フリッカ判定結果(可変フォーマット)の出力項目を削除します。

構文 **:NUMeric:FLICKer:INfOrMation:DELeTe** {<NRf>[,<NRf>]}
 1つ目の<NRf>=1~32(削除を開始する項目番号)
 2つ目の<NRf>=1~32(削除を終了する項目番号)

例 **:NUMERIC:FLICKER:INFORMATION: DELETE 1**
 (ITEM1を削除し, ITEM2以降を前へ詰める)
:NUMERIC:FLICKER:INFORMATION: DELETE 1,3
 (ITEM1~3を削除し, ITEM4以降を前へ詰める)

解説
 ・フリッカ測定機能(オプション, /FL)搭載時だけ有効です。
 ・削除された出力項目の位置には, それ以降の出力項目が順次詰められ, 最後の空いた部分には「NONE」が設定されます。
 ・2つ目の<NRf>を省略した場合, 削除開始番号の出力項目のみを削除します。

:NUMeric:FLICKer:INfOrMation:ITEM<x>

機能 フリッカ判定結果(可変フォーマット)の出力項目(ファンクション・エレメント・観測期間)を設定/問い合わせします。

構文 **:NUMeric:FLICKer:INfOrMation:ITEM<x>** {NONE|<Function>,<Element>[,<Period>]}
:NUMeric:FLICKer:INfOrMation:ITEM<x>?
 <x>=1~32(項目番号)
 NONE=出力項目なし
 <Function>={JTOTal|JDC|JDMAX|JDT|JPST|JPLT}
 <Element>={<NRf>|TOTal}(<NRf>=1~4)
 <Period>={<NRf>|ALL}(<NRf>=1~99)

例 **:NUMERIC:FLICKER:INFORMATION: ITEM1 JDC,1,1**
:NUMERIC:FLICKER:INFORMATION: ITEM1? -> :NUMERIC:FLICKER: INFORMATION:ITEM1 JDC,1,1

解説
 ・フリッカ測定機能(オプション, /FL)搭載時だけ有効です。
 ・<Function>の選択肢に対する出力内容は次のとおりです。

<Function>	出力内容	<Element> 指定	<Period> 指定
JTOTal	dc, dmax, d(t), pst, pltの 総合判定結果 (画面右上の「Element# Judgement:」に表示される判定結果)	必要	不要
JDC	相対定常電圧変化dcの 判定結果	必要	必要
JDMAX	最大相対電圧変化dmaxの 判定結果	必要	必要
JDT	相対電圧変化時間d(t)の 判定結果	必要	必要
JPST	短期間フリッカ値Pstの 判定結果	必要	必要
JPLT	長期間フリッカ値Pltの 判定結果	必要	不要

・<Element>の選択肢の内容は次のとおりです。<Element>を省略したときは, エレメント1が設定されます。
 TOTal 測定対象エレメント全体の総合判定結果, <Function>=JTOTalのときのみに有効
 (画面右上の「Total Judgement:」に表示される判定結果)
 <NRf>=1~4 指定されたエレメント
 ・<Period>の選択肢の内容は次のとおりです。<Period>を省略したときは, ALLが設定されます。
 ALL 観測期間全体(フリッカ測定表示画面の数値リストの結果行)
 <NRf>=1~99 指定された観測期間
 ・上記の表で, <Period>の指定が不要な出力項目の応答は, <Period>が省略されます。

:NUMeric:FLICKer:INFORmation:NUMBER

機能 「:NUMeric:FLICKer:INFORmation:VALUE?」で送信されるフリッカ判定結果の個数を設定/問い合わせします。

構文 :NUMeric:FLICKer:INFORmation:NUMBER {<NRf>|ALL}
:NUMeric:FLICKer:INFORmation:NUMBER?
<NRf>=1~32(ALL)

例 :NUMERIC:FLICKER:INFORMATION:NUMBER 7
:NUMERIC:FLICKER:INFORMATION:NUMBER -> :NUMERIC:FLICKER:FUNCTION:NUMBER 7

解説 ・フリッカ測定機能(オプション, /FL)搭載時だけ有効です。
・「:NUMeric:FLICKer:INFORmation:VALUE?」でパラメータを省略すると, 1~(設定値)の判定結果を順に出力します。
・初期設定では, 判定結果のデータの個数は「7」が設定されています。

:NUMeric:FLICKer:INFORmation:VALUE?

機能 フリッカ判定結果(可変フォーマット)を問い合わせます。

構文 :NUMeric:FLICKer:INFORmation:VALUE? {<NRf>}
:NUMeric:FLICKer:INFORmation:VALUE?
<NRf>=1~32(項目番号)

例 ・<NRf>を指定した場合の例
:NUMERIC:FLICKER:INFORMATION:VALUE? 1 -> 0
・<NRf>を省略した場合の例
:NUMERIC:FLICKER:INFORMATION:VALUE? -> 0,0,0,0,0,0,0
・「:NUMeric:FORMat」の設定が「FLOat」の場合の例
:NUMERIC:FLICKER:INFORMATION:VALUE? -> #4(4桁のバイト数)(データバイトの並び)

解説 ・フリッカ測定機能(オプション, /FL)搭載時だけ有効です。
・<NRf>を指定した場合, その項目番号の判定結果のみを出力します。
・<NRf>を省略した場合, 1~「:NUMeric:FLICKer:INFORmation:NUMBER」の項目番号の判定結果を順に出力します。

- ・出力される個々の判定結果の形式は次のとおりです。
- ・判定結果(JTOTal, JDC, JDMAX, JDT, JPST, JPLT)

	ASCII (<NR1>形式)	FLOAT (IEEE単精度浮動小数点(4byte)形式)
Pass :	「0」	0x00000000 (0)
Fail :	「-1」	0xBF800000 (-1)
Error :	「-2」	0xC0000000 (-2)
---- :	「1」	0x3F800000 (1)
(空白) :	「1」	0x3F800000 (1)

- ・項目無し(NONE)
ASCII : 「NaN」 (Not A Number)
FLOAT : 0x7E951BEE(9.91E+37)

:NUMeric:FLICKer:JUDGement?

機能 フリッカ判定結果(固定フォーマット)を問い合わせます。

構文 :NUMeric:FLICKer:JUDGement? {<NRf>|ALL}
:NUMeric:FLICKer:JUDGement?
<NRf>=1~9(観測期間番号)
ALL=観測期間全体(Result)

例 ・「:NUMeric:FORMat」の設定が「AScii」の場合の例
:NUMERIC:FLICKER:JUDGEMENT? 1 -> 0,0,0,0,0,0,0,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1
・「:NUMeric:FORMat」の設定が「FLOat」の場合の例
:NUMERIC:FLICKER:JUDGEMENT? -> #4 (4桁のバイト数)(データバイトの並び)

解説 ・フリッカ測定機能(オプション, /FL)搭載時だけ有効です。
・出力される判定結果の内容および順序は, 以下のとおり固定フォーマットとなります。出力内容および順序を独自に構成したい場合は, 「:NUMeric:FLICKer:INFORmation」系のコマンドをお使いください。

※判定結果の出力内容および出力順序

ファンクション名は「:NUMeric:FLICker:INfOrmation:ITEM<x>」の解説を参照してください。

数字はエレメントを表します。(ALL)は測定対象エレメント全体の総合判定結果です。

JTOTal(ALL)→
JDC1→JDMAX1→JDT1→JPST1→
JPLT1→JTOTal1→
JDC2→JDMAX2→JDT2→JPST2→
JPLT2→JTOTal2→ (2～4エレメントモデルのみ出力されます)
JDC3→JDMAX3→JDT3→JPST3→
JPLT3→JTOTal3→ (3～4エレメントモデルのみ出力されます)
JDC4→JDMAX4→JDT4→JPST4→
JPLT4→JTOTal4 (4エレメントモデルのみ出力されます)

1エレメントモデルの場合、JTOTal(ALL)～JTOTal1の7データが出力されます。

2エレメントモデルの場合、JTOTal(ALL)～JTOTal2の13データが出力されます。

3エレメントモデルの場合、JTOTal(ALL)～JTOTal3の19データが出力されます。

4エレメントモデルの場合、JTOTal(ALL)～JTOTal4の25データが出力されます。

- ・パラメータを指定した場合、指定された観測期間の判定結果を出力します。
- ・パラメータを省略した場合、観測期間全体(Result)の判定結果を出力します。(パラメータ「ALL」を指定したのと同じ出力となります)
- ・個々の数値データの形式については、
「:NUMeric:FLICker:INfOrmation:VALue?」の解説を参照してください。

:NUMeric:FLICker:PERiod?

機能 フリッカ測定における現在測定中の観測期間番号を問い合わせます。

構文 :NUMeric:FLICker:PERiod?

例 :NUMERIC:FLICKER:PERIOD? -> 5

解説
・フリッカ測定機能(オプション、/FL)搭載時だけ有効です。
・フリッカ測定表示画面の数値リストの「No.」の横に「*」を表示している観測期間番号を返します。「*」を表示していない状態(リセット状態や初期化中など)のときは、「0」を返します。

:NUMeric:FLICker:VALue?

機能 フリッカ測定データ(固定フォーマット)を問い合わせます。

構文 :NUMeric:FLICker:VALue? {<NRf>|ALL}

:NUMeric:FLICker:VALue?

<NRf>= 1～99(観測期間番号)

ALL=観測期間全体(Result)

例
・「:NUMeric:FORMat」の設定が「AScii」の場合の例

```
:NUMERIC:FLICKER:VALUE? 1 ->
600,229.75E+00,50.000E+00,
1.52E+00,1.56E+00,3E+00,...
```

・「:NUMeric:FORMat」の設定が「FLOat」の場合の例

```
:NUMERIC:FLICKER:VALUE? -> #4(4桁のバイト数)(データバイトの並び)
```

解説
・フリッカ測定機能(オプション、/FL)搭載時だけ有効です。
・出力される数値データの内容および順序は、以下のとおり固定フォーマットとなります。出力内容および順序を独自に構成したい場合は、「:NUMeric:FLICker:FUNCTion」系のコマンドをお使いください。

※数値データの出力内容および出力順序

ファンクション名は「:NUMeric:FLICker:FUNCTion:ITEM<x>」の解説を参照してください。

数字はエレメントを表します。

TIME→
UN1→FU1→DC1→DMAX1→DT1→
PST1→PLT1→
UN2→FU2→DC2→DMAX2→DT2→
PST2→PLT2→ (2～4エレメントモデルのみ出力されます)
UN3→FU3→DC3→DMAX3→DT3→
PST3→PLT3→ (3～4エレメントモデルのみ出力されます)
UN4→FU4→DC4→DMAX4→DT4→
PST4→PLT4 (4エレメントモデルのみ出力されます)

1エレメントモデルの場合、TIME～PLT1の8データが出力されます。

2エレメントモデルの場合、TIME～PLT2の15データが出力されます。

3エレメントモデルの場合、TIME～PLT3の22データが出力されます。

4エレメントモデルの場合、TIME～PLT4の29データが出力されます。

- ・パラメータを指定した場合、指定された観測期間の測定データを出力します。
- ・パラメータを省略した場合、現在測定中の観測期間の測定データを出力します。測定完了(Complete)状態のときは、観測期間全体(Result)の測定データとなります。
- ・個々の数値データの形式については、
「:NUMeric:FLICker:FUNCTion:VALue?」の解説を参照してください。

:NUMeric:FORMat

- 機能 「:NUMeric[:NORMal]:VALue?」または「:NUMeric:LIST: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}**
: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。
 - 「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。
 - 「FLOat」のとき
数値データブロックの先頭に6または8バイトのヘッダ(例 "#40060", "#6000408")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」(6-92ページ)を参照してください。

:NUMeric:HOLD

- 機能 すべての数値データを保持する(ON)/解除する(OFF)を設定/問い合わせします。
- 構文 **:NUMeric:HOLD {<Boolean>}**
:NUMeric:HOLD?
- 例 **:NUMERIC:HOLD ON**
:NUMERIC:HOLD? -> :NUMERIC:HOLD 1
- 解説
 - 「:NUMeric[:NORMal]:VALue?」または「:NUMeric:LIST:VALue?」を実行する前に**:NUMeric:HOLD**を「ON」にすると、その時点のすべての数値データを内部に保持することができます。
 - 表示画面上では数値データが更新されていても、**:NUMeric:HOLD**を「OFF」にしない限り、数値データは保持されます。
 - たとえば、同じ時点でのエレメントごとの各種数値データを取得したいときには、次のようにします。
:NUMeric:HOLD ON
:NUMeric[:NORMal]:ITEM1 U,1,TOTAL; ITEM2 I,1,TOTAL;... (エレメント1の数値データ項目を設定)
:NUMeric[:NORMal]:VALue? (エレメント1の数値データを受信)
:NUMeric[:NORMal]:ITEM1 U,2,TOTAL; ITEM2 I,2,TOTAL;... (エレメント2の数値データ項目を設定)
:NUMeric[:NORMal]:VALue? (エレメント2の数値データを受信)
 ... (中略) ...
:NUMeric[:NORMal]:ITEM1 U,4,TOTAL; ITEM2 I,4,TOTAL;... (エレメント4の数値データ項目を設定)
:NUMeric[:NORMal]:VALue? (エレメント4の数値データを受信)
:NUMeric:HOLD OFF
 - :NUMeric:HOLD**を「ON」にした状態で「ON」を設定すると、数値データを一旦解除し、再度最新の数値データを内部に保持します。数値データを連続的に取得するときは、この方法により、毎回**:NUMeric:HOLD**を「OFF」にする必要がなくなります。

:NUMeric:LIST?

- 機能 高調波測定の数値リスト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:LIST?**
- 例 **:NUMERIC:LIST? ->**
:NUMERIC:LIST:NUMBER 1;
ORDER 100;SELECT ALL;ITEM1 U,1
- 解説
 - 高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時だけ有効です。
 - 「:NUMeric:LIST:ITEM<x>」に関する設定値については、「:NUMeric:LIST:NUMBER」で設定されている数だけの数値リストデータ出力項目を出力します。

:NUMeric:LIST:CLear

機能	高調波測定の数値リストデータの出力項目をクリア(「NONE」に設定)します。
構文	:NUMeric:LIST:CLear {ALL <NRf>[,<NRf>]}
	ALL=すべての項目をクリア 1つ目の<NRf>=1~64(クリアを開始する項目番号) 2つ目の<NRf>=1~64(クリアを終了する項目番号)
例	:NUMERIC:LIST:CLEAR ALL
解説	- 高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時だけ有効です。 2つ目の<NRf>を省略した場合, クリア開始番号から最後(64)までの出力項目をクリアします。

:NUMeric:LIST:DElete

機能	高調波測定の数値リストデータの出力項目を削除します。
構文	:NUMeric:LIST:DElete {<NRf>[,<NRf>]}
	1つ目の<NRf>=1~64(削除を開始する項目番号) 2つ目の<NRf>=1~64(削除を終了する項目番号)
例	:NUMERIC:LIST:DELETE 1 (ITEM1を削除し, ITEM2以降を前へ詰める) :NUMERIC:LIST:DELETE 1,3 (ITEM1~3を削除し, ITEM4以降を前へ詰める)
解説	- 高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時だけ有効です。 - 削除された出力項目の位置には, それ以降の出力項目が順次詰められ, 最後の開いた部分には「NONE」が設定されます。 2つ目の<NRf>を省略した場合, 削除開始番号の出力項目のみを削除します。

:NUMeric:LIST:ITEM<x>

機能	高調波測定の数値リストデータの出力項目(ファンクション・エレメント)を設定/問い合わせします。
構文	:NUMeric:LIST:ITEM<x> {NONE <Function>,<Element>}
	:NUMeric:LIST:ITEM<x>? <x>=1~64(項目番号) NONE=出力項目なし <Function>={U I P S Q LAMBda PHI PHIU PHII Z RS XS RP XP UHDF IHDF PHDF}
	<Element>={<NRf> SIGMA SIGMB}<NRf>=1~4)
例	:NUMERIC:LIST:ITEM1 U,1 :NUMERIC:LIST:ITEM1? -> :NUMERIC:LIST:ITEM1 U,1
解説	高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時だけ有効です。

:NUMeric:LIST:NUMBER

機能	「:NUMeric:LIST:VALue?」で送信される数値リストデータの個数を設定/問い合わせします。
構文	:NUMeric:LIST:NUMBER {<NRf> ALL}
	:NUMeric:LIST:NUMBER? <NRf>=1~64(ALL)
例	:NUMERIC:LIST:NUMBER 5 :NUMERIC:LIST:NUMBER -> :NUMERIC:LIST:NUMBER 5
解説	- 高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時だけ有効です。 「:NUMeric:LIST:VALue?」コマンドのパラメータを省略すると, 1~(設定値)の数値リスト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「1」が設定されています。

:NUMeric:LIST:ORDER

機能	高調波測定の数値リストデータの出力最高次数を設定/問い合わせします。
構文	:NUMeric:LIST:ORDER {<NRf> ALL}
	:NUMeric:LIST:ORDER? <NRf>=1~100(ALL)
例	:NUMERIC:LIST:ORDER 100 :NUMERIC:LIST:ORDER? -> :NUMERIC:LIST:ORDER 100
解説	高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時だけ有効です。

:NUMeric:LIST:PRESet

機能 高調波測定の数値リスト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:LIST:PRESet {<NRf>}**
 <NRf>=1~4

例 **:NUMERIC:LIST:PRESET 1**

解説

- ・高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時だけ有効です。
- ・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) 高調波測定の数値リストデータ出力項目のプリセットパターン」(6-95ページ)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:LIST:SElect

機能 高調波測定の数値リストデータの出力成分を設定/問い合わせします。

構文 **:NUMeric:LIST:SElect {EVEN|ODD|ALL}**
:NUMeric:LIST:SElect?

例 **:NUMERIC:LIST:SELECT ALL**

:NUMERIC:LIST:SELECT? ->

:NUMERIC:LIST:SELECT ALL

解説

- ・高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時だけ有効です。
- ・選択肢の内容はそれぞれ次のとおりです。
 EVEN=TOTAL, DC, 偶数次の各成分を出力
 ODD=TOTAL, DC, 奇数次の各成分を出力
 ALL=すべての成分を出力

:NUMeric:LIST:VALue?

機能 高調波測定の数値リストデータを問い合わせます。

構文 **:NUMeric:LIST:VALue? {<NRf>}**
 <NRf>=1~64(項目番号)

例

- ・<NRf>を指定した場合の例

:NUMERIC:LIST:VALUE? 1 ->
 103.58E+00,0.00E+00,103.53E+00,
 0.09E+00,2.07E+00,0.04E+00,...(中略)
 ...,0.01E+00,0.01E+00 (最大102個のデータ)

- ・<NRf>を省略した場合の例
 (「:NUMeric:LIST:NUMBER」の設定が「5」のとき)

:NUMERIC:LIST:VALUE? ->
 103.58E+00,0.00E+00,103.53E+00,
 0.09E+00,2.07E+00,0.04E+00,...(中略)
 ...,0.00E+00,0.00E+00 (最大102*5=510個のデータ)

- ・「:NUMeric:FORMat」の設定が「FLOat」の場合の例

:NUMERIC:LIST:VALUE? -> #6(6桁のバイト数)(データバイトの並び)

解説

- ・高度演算機能(オプション, /G6)または高調波測定機能(オプション, /G5)搭載時だけ有効です。

- ・1つの数値リストデータは、TOTAL, DC, 1次~「:NUMeric:LIST:ORDER」の順に、最大102個の数値データで構成されています。

- ・<NRf>を指定した場合、その項目番号の数値リストデータのみを出力します。(最大102個のデータ)

- ・<NRf>を省略した場合、1~

「:NUMeric:LIST:NUMBER」の項目番号の数値リストデータを順に出力します。(最大102*「:NUMeric:LIST:NUMBER」個のデータ)

- ・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」(6-92ページ)を参照してください。

:NUMeric:NORMal?

機能	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 例	:NUMeric:NORMal? :NUMERIC:NORMAL? -> :NUMERIC: NORMAL:NUMBER 15;ITEM1 U,1,TOTAL; ITEM2 I,1,TOTAL;ITEM3 P,1,TOTAL; ITEM4 S,1,TOTAL;ITEM5 Q,1,TOTAL; ITEM6 LAMBDA,1,TOTAL; ITEM7 PHI,1,TOTAL;ITEM8 FU,1; ITEM9 FI,1;ITEM10 UPPEAK,1; ITEM11 UMPEAK,1;ITEM12 IPPEAK,1; ITEM13 IMPEAK,1;ITEM14 CFU,1; ITEM15 CFI,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	数値データの出力項目をクリア(「NONE」に設 定)します。
構文 例 解説	: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) :NUMERIC:NORMAL:CLEAR ALL 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]:DELeTe

機能	数値データの出力項目を削除します。
構文 例 解説	:NUMeric[:NORMal]:DELeTe {<Nrf> [,<Nrf>]} 1つ目の<Nrf>=1~255(削除を開始する項目 番号) 2つ目の<Nrf>=1~255(削除を終了する項目 番号) :NUMERIC:NORMAL:DELETE 1 (ITEM1を削除 し、ITEM2以降を前へ詰める) :NUMERIC:NORMAL:DELETE 1,3 (ITEM1~3 を削除し、ITEM4以降を前へ詰める) ・ 削除された出力項目の位置には、それ以降の 出力項目が順次詰められ、最後の開いた部分 には「NONE」が設定されます。 ・ 2つ目の<Nrf>を省略した場合、削除開始番 号の出力項目のみを削除します。

:NUMeric[:NORMal]:ITEM<x>

機能	数値データ出力項目(ファンクション・エレメン ト・次数)を設定/問い合わせします。
構文 例 解説	:NUMeric[:NORMal]:ITEM<x> {NONE <Function>,<Element>[,<Order>]} :NUMeric[:NORMal]:ITEM<x>? <x>=1~255(項目番号) NONE=出力項目なし <Function>={U I P S Q ...}(「DISPlayグ ループ」のファンクション選択肢一覧(1), 6- 41ページ参照) <Element>={<Nrf> SIGMA SIGMB}(<Nrf> =1~4) <Order>={TOTaI DC <Nrf>}(<Nrf>=1~ 100) :NUMERIC:NORMAL:ITEM1 U,1,TOTAL :NUMERIC:NORMAL:ITEM1? -> :NUMERIC:NORMAL:ITEM1 U,1,TOTAL ・ <Element>を省略したときは、エレメント1 が設定されます。 ・ <Order>を省略したときは、TOTaIが設定さ れます。 ・ <Element>または<Order>が不要なファンク ションの応答は、<Element>または<Order> が省略されます。

: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 例 解説	:NUMeric[:NORMal]:NUMBER {<Nrf> ALL} :NUMeric[:NORMal]:NUMBER? <Nrf>=1~255(ALL) :NUMERIC:NORMAL:NUMBER 15 :NUMERIC:NORMAL:NUMBER -> :NUMERIC:NORMAL:NUMBER 15 ・ 「: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	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 例 解説	:NUMeric[:NORMal]:PRESet {<Nrf>} <Nrf>=1~4 :NUMERIC:NORMAL:PRESET 1 ・ 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)数値デー タ出力項目のプリセットパターン」(6-93 ページ)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 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」(6-92ページ)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

- ・ 周波数(FU, FI, FREQ)
ASCII: <NR3>形式(仮数部: 5桁, 指数部: 2桁, 例: 50.000E+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)
- ・ Cycle by Cycle測定のピーク情報(PKU, PKI, PKSpeed, PKTorque)

	ASCII (<NR1>形式)	FLOAT (IEEE単精度浮動小数点(4byte)形式)
ピーク発生無し「 ☐ 」:	「0」	0x00000000 (0)
正側ピーク発生「↑+」:	「1」	0x3F800000 (1)
負側ピーク発生「↓-」:	「2」	0x40000000 (2)
両側ピーク発生「↓±」:	「3」	0x40400000 (3)

Note

測定期間内に本機器本体にあるピークオーバ検出機能で検知した場合, 上記数値に4が加えられます。

- ・ 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6桁, 指数部: 2桁, 例: [-]123.456E+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)

Note

- ・ エLEMENT1~4の位相差φ(PHI)については, 180° (Lead/Lag)表示の場合, Lead(D)をマイナス, Lag(G)をプラスとして-180.000~180.000の範囲で出力します。
- ・ 電力値(P, S, Q, PC)のΣについては, 電圧レンジと電流レンジの組み合わせ(電力レンジ)により, 仮数部の桁数が有効最大7桁となる場合があります。電力レンジの一覧表がユーザーズマニュアルIM760301-01に掲載されていますので, 参照してください。

※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) 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>	<Order>
1	U,	1,	TOTal
2	I,	1,	TOTal
3	P,	1,	TOTal
4	S,	1,	TOTal
5	Q,	1,	TOTal
6	LAMBda,	1,	TOTal
7	PHI,	1,	TOTal
8	FU,	1,	(TOTal)
9	FI,	1,	(TOTal)
10	NONE,		
11~19	U~FI,	2,	TOTal
20	NONE,		
21~29	U~FI,	3,	TOTal
30	NONE,		
31~39	U~FI,	4,	TOTal
40	NONE,		
41~49	U~FI,	SIGMA,	TOTal
50	NONE,		
51~59	U~FI,	SIGMB,	TOTal
60	NONE,		
61~255	NONE,		

・パターン2

ITEM<x>	<Function>	<Element>	<Order>
1	U,	1,	TOTal
2	I,	1,	TOTal
3	P,	1,	TOTal
4	S,	1,	TOTal
5	Q,	1,	TOTal
6	LAMBda,	1,	TOTal
7	PHI,	1,	TOTal
8	FU,	1,	(TOTal)
9	FI,	1,	(TOTal)
10	UPPeak,	1,	(TOTal)
11	UMPeak,	1,	(TOTal)

12	IPPeak,	1,	(TOTal)
13	IMPeak,	1,	(TOTal)
14	CFU,	1,	(TOTal)
15	CFI,	1,	(TOTal)
16~30	U~CFI,	2,	TOTal
31~45	U~CFI,	3,	TOTal
46~60	U~CFI,	4,	TOTal
61~75	U~CFI,	SIGMA,	TOTal
76~90	U~CFI,	SIGMB,	TOTal
91~255	NONE,		

・パターン3

ITEM<x>	<Function>	<Element>	<Order>
1	U,	1,	TOTal
2	I,	1,	TOTal
3	P,	1,	TOTal
4	S,	1,	TOTal
5	Q,	1,	TOTal
6	TIME,	1,	(TOTal)
7	WH,	1,	(TOTal)
8	WHP,	1,	(TOTal)
9	WHM,	1,	(TOTal)
10	AH,	1,	(TOTal)
11	AHP,	1,	(TOTal)
12	AHM,	1,	(TOTal)
13	WS,	1,	(TOTal)
14	WQ,	1,	(TOTal)
15	NONE,		
16~29	U~WQ,	2,	TOTal
30	NONE,		
31~44	U~WQ,	3,	TOTal
45	NONE,		
46~59	U~WQ,	4,	TOTal
60	NONE,		
61~74	U~WQ,	SIGMA,	TOTal
75	NONE,		
76~89	U~WQ,	SIGMB,	TOTal
90	NONE,		
91~255	NONE,		

・パターン4

ITEM<x>	<Function>	<Element>	<Order>
1	U,	1,	TOTal
2	I,	1,	TOTal
3	P,	1,	TOTal
4	S,	1,	TOTal
5	Q,	1,	TOTal
6	LAMBda,	1,	TOTal
7	PHI,	1,	TOTal
8	FU,	1,	(TOTal)

9	FI,	1,	(TOTal)
10	UPPeak,	1,	(TOTal)
11	UMPeak,	1,	(TOTal)
12	IPPeak,	1,	(TOTal)
13	IMPeak,	1,	(TOTal)
14	CFU,	1,	(TOTal)
15	CFI,	1,	(TOTal)
16	PC,	1,	(TOTal)
17	TIME,	1,	(TOTal)
18	WH,	1,	(TOTal)
19	WHP,	1,	(TOTal)
20	WHM,	1,	(TOTal)
21	AH,	1,	(TOTal)
22	AHP,	1,	(TOTal)
23	AHM,	1,	(TOTal)
24	WS,	1,	(TOTal)
25	WQ,	1,	(TOTal)
26~50	U~WQ,	2,	TOTal
51~75	U~WQ,	3,	TOTal
76~100	U~WQ,	4,	TOTal
101~125	U~WQ,	SIGMA,	TOTal
126~150	U~WQ,	SIGMB,	TOTal
151~255	NONE,		

(2) 高調波測定の数値リストデータ出力項目のプリセットパターン

対象コマンド「:NUMeric:LIST:PRESet」

・パターン1

ITEM<x>	<Function>	<Element>
1	U,	1
2	I,	1
3	P,	1
4~6	U~P,	2
7~9	U~P,	3
10~12	U~P,	4
13~64	NONE,	

・パターン2

ITEM<x>	<Function>	<Element>
1	U,	1
2	I,	1
3	P,	1
4	PHIU,	1
5	PHII,	1
6~10	U~PHII,	2
11~15	U~PHII,	3
16~20	U~PHII,	4
21~64	NONE,	

• パターン3

ITEM<x>	<Function>	<Element>
1	U,	1
2	I,	1
3	P,	1
4	Q,	1
5	Z,	1
6	RS,	1
7	XS,	1
8	RP,	1
9	XP,	1
10~18	U~XP,	2
19~27	U~XP,	3
28~36	U~XP,	4
37~64	NONE,	

• パターン4

ITEM<x>	<Function>	<Element>
1	U,	1
2	I,	1
3	P,	1
4	S,	1
5	Q,	1
6	LAMBda,	1
7	PHI,	1
8	PHIU,	1
9	PHII,	1
10	Z,	1
11	RS,	1
12	XS,	1
13	RP,	1
14	XP,	1
15~28	U~XP,	2
29~42	U~XP,	3
43~56	U~XP,	4
57~64	NONE,	

6.19 RATEグループ

RATEグループは、データ更新周期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	データ更新周期を設定/問い合わせします。
構文	:RATE {<時間>} :RATE? <時間>=50, 100, 250, 500(ms), 1, 2, 5, 10, 20(s)
例	:RATE 500MS :RATE? -> :RATE 500.0E-03

6.20 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、7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:EES 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 1;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 状態レジスタについては、7章のステータスレポートをご覧ください。

:STATus:EES

(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:EES <Register>
:STATus:EES?

例 :STATUS:EES #B00000000000000000
:STATUS:EES? -> :STATUS:EES 0

解説 拡張イベントイネーブルレジスタについては、7章のステータスレポートをご覧ください。

: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

解説 拡張イベントレジスタについては、7章のステータスレポートをご覧ください。

: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>?
<x>=1~16

例 :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。
- ・遷移フィルタについては、7章のステータスレポートをご覧ください。

: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:SPOLL? (Serial Poll)

機能	シリアルポーラを実行します。
構文	:STATus:SPOLL?
例	:STATUS:SPOLL? -> :STATUS:SPOLL 0
解説	RS-232/USB/イーサネット(いずれもオプション)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。

6.21 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。

: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;U 1;I 1; P 1;S 1;Q 1;LAMBDA 1;PHI 1;FU 1; FI 1;UPPEAK 0;UMPEAK 0;IPPEAK 0; IMPEAK 0;CFU 0;CFI 0;PC 0;TIME 0; WH 0;WHP 0;WHM 0;AH 0;AHP 0;AHM 0; WS 0;WQ 0;ETA1 0;ETA2 0;ETA3 0; ETA4 0;F1 0;F2 0;F3 0;F4 0;F5 0; F6 0;F7 0;F8 0;F9 0;F10 0;F11 0; F12 0;F13 0;F14 0;F15 0;F16 0; F17 0;F18 0;F19 0;F20 0;;STORE: MEMORY:ALERT 1

: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: TYPE ASCII;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の対象ドライブとディレクトリは、それぞれ次のコマンドで設定します。 ・対象ドライブ:「:FILE:DRIVE」 ・ディレクトリ:「:FILE:CDIRECTORY」 保存先のパスは「:FILE:PATH?」で問い合わせできます。

:STORE:FILE:TYPE

機能	ストアしたデータを保存するときのデータ形式を設定/問い合わせします。
構文	:STORE:FILE:TYPE {ASCIi FLOat} :STORE:FILE:TYPE?
例	:STORE:FILE:TYPE ASCII :STORE:FILE:TYPE? -> :STORE:FILE:TYPE ASCII

:STORE:INTERVAL

機能 ストアインタバルを設定/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 ストア対象を設定/問い合わせします。

構文 **:STORE:ITEM {NUMERIC|WAVE|NWAve}**
:STORE:ITEM?
 NUMERIC=数値データのみストア
 WAVE=波形表示データのみストア
 NWAve=数値データと波形表示データを両方ストア

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

:STORE:MEMORY?

機能 ストアメモリに関するすべての設定値を問い合わせます。

構文 **:STORE:MEMORY?**

例 **:STORE:MEMORY? ->**
:STORE:MEMORY:ALERT 1

:STORE:MEMORY:ALERT

機能 ストアメモリを初期化するときを確認メッセージを出す/出さないを設定/問い合わせします。

構文 **:STORE:MEMORY:ALERT {<Boolean>}**
:STORE:MEMORY:ALERT?

例 **:STORE:MEMORY:ALERT ON**
:STORE:MEMORY:ALERT? ->
:STORE:MEMORY:ALERT 1

解説 「:STORE:MEMORY:INITIALIZE」コマンドでストアメモリを初期化するときは、この設定に関係なく直ちに実行します。

: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、WT3000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;U 1;I 1;P 1;S 1;Q 1;
LAMBDA 1;PHI 1;FU 1;FI 1;UPPEAK 0;
UMPEAK 0;IPPEAK 0;IMPEAK 0;CFU 0;
CFI 0;PC 0;TIME 0;WH 0;WHP 0;WHM 0;
AH 0;AHP 0;AHM 0;WS 0;WQ 0;ETA1 0;
ETA2 0;ETA3 0;ETA4 0;F1 0;F2 0;
F3 0;F4 0;F5 0;F6 0;F7 0;F8 0;F9 0;
F10 0;F11 0;F12 0;F13 0;F14 0;
F15 0;F16 0;F17 0;F18 0;F19 0;F20 0

:STORE:NUMERIC:NORMAL?

機能 数値データのストア項目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;U 1;I 1;P 1;S 1;
Q 1;LAMBDA 1;PHI 1;FU 1;FI 1;
UPPEAK 0;UMPEAK 0;IPPEAK 0;
IMPEAK 0;CFU 0;CFI 0;PC 0;TIME 0;
WH 0;WHP 0;WHM 0;AH 0;AHP 0;AHM 0;
WS 0;WQ 0;ETA1 0;ETA2 0;ETA3 0;
ETA4 0;F1 0;F2 0;F3 0;F4 0;F5 0;
F6 0;F7 0;F8 0;F9 0;F10 0;F11 0;
F12 0;F13 0;F14 0;F15 0;F16 0;
F17 0;F18 0;F19 0;F20 0

:STORE:NUMERIC[:NORMAL]:ALL

機能 数値データをストアするときのすべてのエレメント・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}

機能 数値データをストアするときの{各エレメント|ΣA|ΣB}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}

{<Boolean>}

:STORE:NUMERIC[:NORMAL]:

{ELEMENT<x>|SIGMA|SIGMB}?

<x>=1~4

例 :STORE:NUMERIC:NORMAL:ELEMENT1 ON

:STORE:NUMERIC:NORMAL:ELEMENT1? ->

:STORE:NUMERIC:NORMAL:ELEMENT1 1

解説
・「:STORE:NUMERIC[:NORMAL]:SIGMA」は、2エレメント以上のモデルで有効です。また、出力をONに設定するためには、あらかじめ結線方式の設定([:INPUT]WIRINGコマンド)により結線ユニットΣAが存在する必要があります。

・「:STORE:NUMERIC[:NORMAL]:SIGMB」は、4エレメントモデルのみ有効です。また、出力をONに設定するためには、あらかじめ結線方式の設定([:INPUT]WIRINGコマンド)により結線ユニットΣBが存在する必要があります。

:STORE:NUMERIC[:NORMAL]:PRESET<x>

機能 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、ユーザーズマニュアル IM760301-01を参照してください。

:STORE:NUMERIC[:NORMAL]:<Function>

機能 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]:<Function>

{<Boolean>}

:STORE:NUMERIC[:NORMAL]:<Function>?

<Function>={U|I|P|S|Q|...}(「DISPlayグループ」のファンクション選択肢一覧(1), 6-41ページ参照)

例 :STORE:NUMERIC:NORMAL:U ON

:STORE:NUMERIC:NORMAL:U? ->

:STORE:NUMERIC:NORMAL:U 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予約時刻を問い合わせます。

構文 **:STORE:RTIME?**

例 **:STORE:RTIME? -> :STORE:RTIME:**
START 2005,1,1,0,0,0;
END 2005,1,1,1,0,0

: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}予約時刻を設定/問い合わせします。

構文 **:STORE:RTIME:{START|END} {<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 2005,1,1,0,0,0**
:STORE:RTIME:START? ->
:STORE:RTIME:START 2005,1,1,0,0,0

解説 スタモード(:STORE:SMODE)が「RTIME(実時間ストアモード)」のときに有効な設定です。

:STORE:SMODE

機能 スタモードを設定/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>|SPEEd|TORQue}

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>|SPEEd|TORQue} {<Boolean>}**
:STORE:WAVE:{U<x>|I<x>|SPEEd|TORQue}?
 <x>=1~4

例 **:STORE:WAVE:U1 ON**
:STORE:WAVE:U1? -> :STORE:WAVE:U1 1

解説 {SPEEd|TORQue}は、モータバージョン(-MV)のみ有効です。

6.22 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:
MESSAGE ENGLISH;MENU ENGLISH;;
SYSTEM:FONT GOTHIC;KLOCK 0;SLOCK 0;
LCD:BRIGHTNESS 2;COLOR:GRAPH:
MODE DEFAULT;;SYSTEM:LCD:COLOR:
TEXT:MODE PRESET1

:SYSTem:CLOCK?

機能 日付/時刻に関するすべての設定値を問い合わせます。

構文 :SYSTem:CLOCK?

例 :SYSTEM:CLOCK? ->
:SYSTEM:CLOCK:DISPLAY 1;TYPE MANUAL

:SYSTem:CLOCK:DISPlay

機能 日付/時刻表示のON/OFFを設定/問い合わせします。

構文 :SYSTem:CLOCK:DISPlay {<Boolean>}
:SYSTem:CLOCK:DISPlay?

例 :SYSTEM:CLOCK:DISPLAY ON
:SYSTEM:CLOCK:DISPLAY? ->
:SYSTEM:CLOCK:DISPLAY 1

:SYSTem:CLOCK:SNTP?

機能 SNTPによる日付/時刻に関するすべての設定値を問い合わせます。

構文 :SYSTem:CLOCK:SNTP?

例 :SYSTEM:CLOCK:SNTP? ->
:SYSTEM:CLOCK:SNTP:GMTTIME "09:00"

解説 イーサネット(オプション、/C7)搭載時だけ有効です。

:SYSTem:CLOCK:SNTP[:EXECute]

機能 SNTPによる日付/時刻を設定します。

構文 :SYSTem:CLOCK:SNTP[:EXECute]

例 :SYSTEM:CLOCK:SNTP:EXECUTE

解説 イーサネット(オプション、/C7)搭載時だけ有効です。

:SYSTem:CLOCK:SNTP:GMTTime

機能 グリニッジ標準時との時差を設定/問い合わせします。

構文 :SYSTem:CLOCK:SNTP:GMTTime {<文字列>}
>}

:SYSTem:CLOCK:SNTP:GMTTime?

<文字列>="HH:MM"(HH=時間, MM=分)

例 :SYSTEM:CLOCK:SNTP:GMTTIME "09:00"

:SYSTEM:CLOCK:SNTP:GMTTIME? ->

:SYSTEM:CLOCK:SNTP:GMTTIME "09:00"

解説 ・イーサネット(オプション、/C7)搭載時だけ有効です。

・日付/時刻の設定方法(:SYSTem:CLOCK:TYPE)が「SNTP」のときに有効な設定です。

:SYSTem:CLOCK:TYPE

機能 日付/時刻の設定方法を設定/問い合わせします。

構文 :SYSTem:CLOCK:TYPE {MANual|SNTP}
:SYSTem:CLOCK:TYPE?

例 :SYSTEM:CLOCK:TYPE MANUAL
:SYSTEM:CLOCK:TYPE? ->

:SYSTEM:CLOCK:TYPE MANUAL

解説 「SNTP」は、イーサネット(オプション、/C7)搭載時だけ有効です。

: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 "05/01/01"

:SYSTEM:DATE? -> "05/01/01"

解説 「年」は西暦の下2桁です。

:SYSTem:EClear

機能 画面上に表示されているエラーメッセージをクリアします。

構文 :SYSTem:EClear

例 :SYSTEM:ECLEAR

:SYSTem:FONT

機能 画面表示フォントを設定/問い合わせします。

構文 :SYSTem:FONT {GOTHic|ROMan}

:SYSTem:FONT?

例 :SYSTEM:FONT GOTHIC

:SYSTEM:FONT? ->

:SYSTEM:FONT GOTHIC

:SYSTem:KLOCK

機能 キーロックのON/OFFを設定/問い合わせします。

構文 :SYSTem:KLOCK {<Boolean>}
:SYSTem:KLOCK?

例 :SYSTEM:KLOCK OFF
:SYSTEM:KLOCK? -> :SYSTEM:KLOCK 0

:SYSTem:LANGuage?

機能 画面表示言語に関するすべての設定値を問い合わせます。

構文 :SYSTem:LANGuage?

例 :SYSTEM:LANGUAGE? ->
:SYSTEM:LANGUAGE:MESSAGE ENGLISH;
MENU ENGLISH

:SYSTem:LANGuage:MENU

機能 メニューの言語を設定/問い合わせします。

構文 :SYSTem:LANGuage:MENU {JAPANEse|
ENGLISH}
:SYSTem:LANGuage:MENU?

例 :SYSTEM:LANGUAGE:MENU ENGLISH
:SYSTEM:LANGUAGE:MENU? ->
:SYSTEM:LANGUAGE:MENU ENGLISH

:SYSTem:LANGuage:MESSAge

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:MESSAge {JAPANEse|
ENGLISH}
:SYSTem:LANGuage:MESSAge?

例 :SYSTEM:LANGUAGE:MESSAGE ENGLISH
:SYSTEM:LANGUAGE:MESSAGE? ->
:SYSTEM:LANGUAGE:MESSAGE 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
<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

解説 R, G, Bの順で設定します。
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:SLOCK

機能 SHIFTキーのON状態を継続する(ON)/しない(OFF)を設定/問い合わせします。

構文 :SYSTem:SLOCK {<Boolean>}
:SYSTem:SLOCK?

例 :SYSTEM:SLOCK OFF
:SYSTEM:SLOCK? -> :SYSTEM:SLOCK 0

: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"

:SYSTem:USBKeyboard

機能 USBキーボードの種類(言語)を設定/問い合わせします。

構文 :SYSTem:USBKeyboard {JAPANEse|ENGLISH}

:SYSTem:USBKeyboard?

例 :SYSTEM:USBKEYBOARD JAPANESE
:SYSTEM:USBKEYBOARD? ->

:SYSTEM:USBKEYBOARD JAPANESE

解説 USBポート(周辺機器)(オプション, /C5)搭載時
だけ有効です。

6.23 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。

:WAVEform?

機能 波形表示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 :WAVEform?

例 :WAVEFORM? -> :WAVEFORM:TRACE U1;
FORMAT ASCII;START 0;END 1001;
HOLD 0

: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>=0~1001

例 :WAVEFORM:END 1001

:WAVEFORM:END? ->

:WAVEFORM:END 1001

: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:HOLD

機能 すべての波形表示データを保持する(ON)/解除する(OFF)を設定/問い合わせします。

構文 :WAVEform:HOLD {<Boolean>}

:WAVEform:HOLD?

例 :WAVEFORM:HOLD ON

:WAVEFORM:HOLD? -> :WAVEFORM:HOLD 1

解説

- ・「:WAVEform:SEND?」を実行する前に
:WAVEform:HOLDを「ON」にすると、その時点のすべての波形データを内部に保持することができます。

- ・表示画面上では波形表示が更新されていても、:WAVEform:HOLDを「OFF」にしない限り、波形データは保持されます。

- ・たとえば、同じ時点でのU1とI1の波形表示データを取得したいときには、次のようにします。

:WAVEform:HOLD ON

:WAVEform:TRACe U1

:WAVEform:SEND?

(U1の波形表示データを受信)

:WAVEform:TRACe I1

:WAVEform:SEND?

(I1の波形表示データを受信)

:WAVEform:HOLD OFF

- ・:WAVEform:HOLDを「ON」にした状態で「ON」を設定すると、波形表示データを一旦解除し、再度最新の波形データを内部に保持します。波形表示データを連続的に取得するときは、この方法により、毎回
:WAVEform:HOLDを「OFF」にする必要がなくなります。

: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変わります。
(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。
・表示方式(:DISPlay:MODE)の設定で波形を表示する設定になっていても、波形表示データが無い場合、次のようになります。
(1)「AScii」のとき
各点のデータはすべて「NAN」になります。
(2)「FLOat」のとき
各点のデータはすべて、「0
(0x00000000)」になります。

: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?
<NRf>=0~1001
- 例 :WAVEFORM:START 0
:WAVEFORM:START? ->
:WAVEFORM:START 0

:WAVEform:TRACe

- 機能 「:WAVEform:SEND?」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>|SPEed|TORQue|MATH<x>}
:WAVEform:TRACe?
U<x>, I<x>の<x>=1~4(エレメント)
MATH<x>の<x>=1, 2(MATH)
- 例 :WAVEFORM:TRACE U1
:WAVEFORM:TRACE? ->
:WAVEFORM:TRACE U1
- 解説 ・{SPEed|TORQue}は、モータバージョン(-MV)のみ有効です。
・MATH<x>は、高度演算機能(オプション、/G6)搭載時のみ有効です。

: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。

6.24 共通コマンドグループ

共通コマンドグループは、IEEE488.2-1992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+SINGLE)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、7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>} *ESE? <Nrf>=0~255
例	*ESE 251 *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、7-5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、7-5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,760304-04-SV,0,F4.01
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。 ・ <機種>は「形名(6桁)-エレメント構成(2桁)-バージョン名」という形式になります。形名とエレメント構成は、760301-01, 760302-02, 760303-03, 760304-04, 760301-10, 760302-20, 760303-30, 760304-40のどれかになります。バージョン名は、スタンダードバージョンがSV, モータバージョンがMVとなります。 ・ <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、5-7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、5-7ページを参照してください。 ・オーバ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? -> G6,B5,DT,FQ,DA,V1,C2,C7,C5,CC,FL
解説	・高調波測定(G5)または高度演算機能(G6)、内蔵プリンタ(B5)、デルタ演算(DT)、周波数測定追加(FQ)、20chDA出力(DA)、VGA出力(V1)、RS-232通信(C2)、USBポート(PC用、C12)、イーサネット(C7)、USBポート(周辺機器用、C5)、Cycle by Cycle機能(CC)、およびフリッカ測定(FL)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7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。

***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? <Nrf>=0~255
例	*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、7-3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、7-3ページを参照してください。

***TRG (TRiGger)**

機能	シングル測定(SINGLEキーを押したのと同じ動作)を実行します。
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。 ・テストが終了するまで約90秒かかります。本機器からの応答を受信するときには、タイムアウト時間を長めに設定してください。

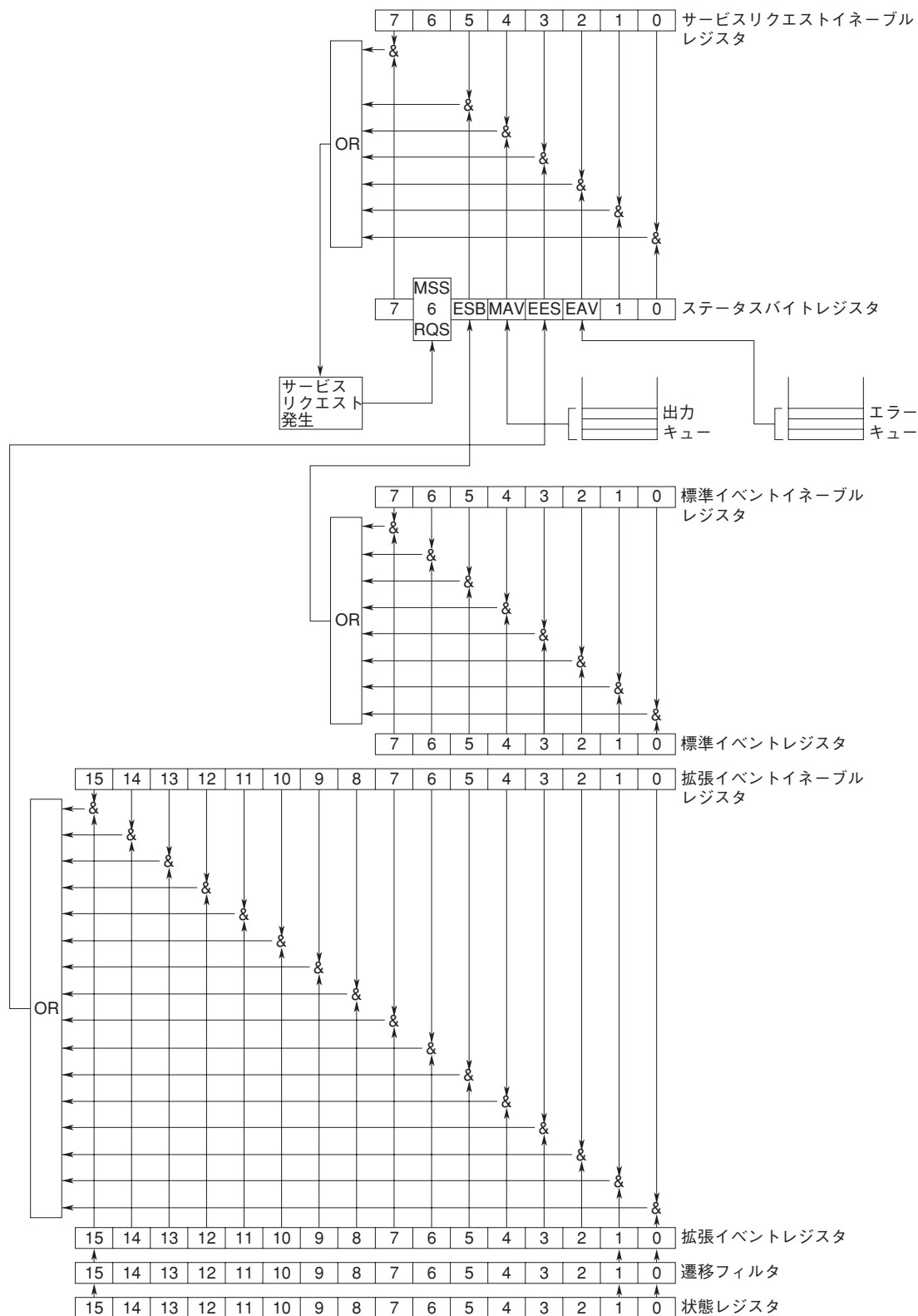
***WAI (WAIt)**

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、5-7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。

7.1 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 488.2-1992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	—	シリアルポール(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:EESE	:STATUS:EESE?
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、6章で詳しく説明しています。

7.2 ステータスバイト

ステータスバイト



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。7-8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。7-7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。7-8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。7-5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、シリアルポールか、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、6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が発生します。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。

7.3 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(6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)の要因にたくないときには, 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, 6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。

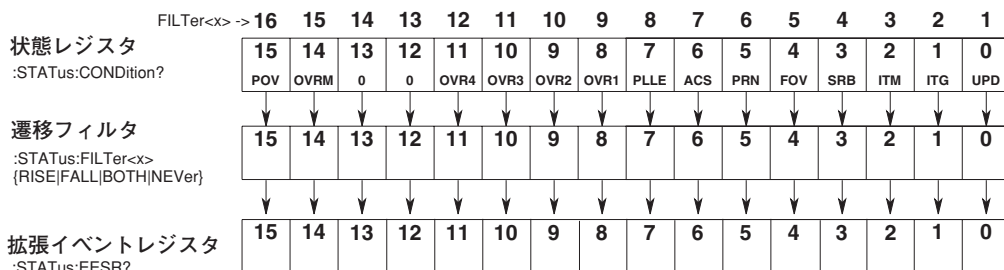
標準イベントレジスタのクリア

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

7.4 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、次のとおりです。

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)

PCカードインタフェース、USBストレージメディア、およびネットワークの各ドライブへのアクセス中に「1」になります。

ビット7 PLLE (PLL Source Input Error)

高調波測定時、PLLソースに入力がなく、同期がかからないときに「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/電流値のどちらかがオーバーレンジのときに「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/電流値のどちらかがオーバーレンジのときに「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/電流値のどちらかがオーバーレンジのときに「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/電流値のどちらかがオーバーレンジのときに「1」になります。

ビット14 OVRM (Motor Measured Data Over)

モータ入力スピード/トルクのどちらかがオーバーレンジのときに「1」になります。

ビット15 POV (ElementX Input Peak Over)

どれかのエレメントでピークオーバーを検出したときに「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」。

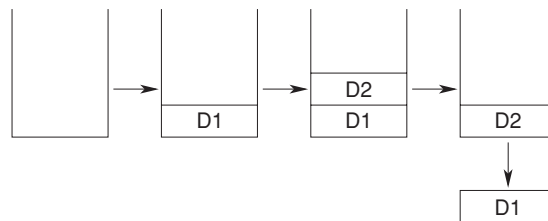
7.5 出力キューとエラーキュー

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(5-2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。

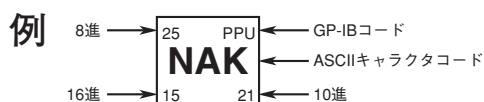
- ・ ***CLS** コマンドを受信したとき
- ・ 電源の再投入

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。

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	80 @	100 P	120 '	140 p
1	1 SOH	21 DC1	41 !	61 1	81 A	101 Q	121 a	141 q
2	2 STX	22 DC2	42 "	62 2	82 B	102 R	122 b	142 r
3	3 ETX	23 DC3	43 #	63 3	83 C	103 S	123 c	143 s
4	4 EOT	24 DC4	44 \$	64 4	84 D	104 T	124 d	144 t
5	5 ENQ	25 NAK	45 %	65 5	85 E	105 U	125 e	145 u
6	6 ACK	26 SYN	46 &	66 6	86 F	106 V	126 f	146 v
7	7 BEL	27 ETB	47 ,	67 7	87 G	107 W	127 g	147 w
8	10 BS	30 CAN	50 (	70 8	90 H	110 X	130 h	150 x
9	11 HT	31 EM	51)	71 9	91 I	111 Y	131 i	151 y
A	12 LF	32 SUB	52 *	72 :	92 J	112 Z	132 j	152 z
B	13 VT	33 ESC	53 +	73 ;	93 K	113 [	133 k	153 {
C	14 FF	34 FS	54 ,	74 <	94 L	114 \	134 l	154
D	15 CR	35 GS	55 -	75 =	95 M	115]	135 m	155 }
E	16 SO	36 RS	56 .	76 >	96 N	116 ^	136 n	156 ~
F	17 SI	37 US	57 /	77 ?	97 O	117 _	137 o	157 DEL (RUBOUT)
	アドレス コマンド	ユニバーサル コマンド	リスナ アドレス		トーカー アドレス		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、:STATus:ERRor?コマンドで問い合わせたパーソナルコンピュータに表示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、お買い求め先まで修理をお申しつけください。
- ・ 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、ユーザズマニュアルIM760301-01をご覧ください。

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。	5章, 6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。	5-1
104	<DATA>の種類が間違っています。 Data type error	5-5~5-6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。	5-5~ 5-6
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。	5-5, 6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。	5-5, 6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。	5-1
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。	6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。	6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。	6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。	5-5
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。	5-5, 6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。	5-5, 6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。	5-5, 6章
131	単位が間違っています。 Invalid suffix	<電圧>、<電流>、<時間>、<周波数>の単位を確認してください。	5-5
134	単位のつづりが長すぎます。 Suffix too long	<電圧>、<電流>、<時間>、<周波数>の単位を確認してください。	5-5
138	単位は使えません。 Suffix not allowed	<電圧>、<電流>、<時間>、<周波数>以外では単位は使えません。	5-5
141	そのような選端& `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。	6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。	6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。	6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。	5-6
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。	6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。	6章
161	<BLOCK DATA>のデータ長が合いません。 Invalid block data	<ブロックデータ>は使用できません。	5-6, 6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。	5-6, 6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。	6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。	6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。	6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。	6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。	6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。	6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。	5-2
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。	5-2
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。	5-2
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	通信ドライバーエラー 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。	7-8
390	Overrun error(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 488.2-1992について

本機器のGP-IBインタフェースは、IEEE 488.2-1992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
「1.3 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
6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
6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
5章および、6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
6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
「6.24 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
「6.24 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
「6.24 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
7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
「5.5 コントローラとの同期」および6章を参照してください。
- (23) 各コマンドの実行内容
6章の各コマンドの機能とユーザズマニュアルIM760301-01, 拡張機能ユー
ザズマニュアルIM760301-51を参照してください。

索引

記号	ページ
^END	5-1
16 値	6-34
1P2W	6-69
1P3W	6-69
2 電力計法の補正	6-73
3P3W	6-69
3P3W->3V3A 変換	6-73
3P4W	6-69
3V3A	6-69
4 値	6-34
8 値	6-34

A	ページ
ACQuisition グループ	6-15
AOUPut グループ	6-17
ASCII キャラクタコード	付-1

B	ページ
Baud Rate	2-8
BMP フォーマット	6-60
Boolean	5-6

C	ページ
CBCycle グループ	6-19
CCITT	2-7
COMMunicate グループ	6-22
CS-RS	2-4
CT 比	6-67
CURSor グループ	6-24

D	ページ
D/A 出力	6-17
D/A 出力項目	6-17, 6-18
d(t)	6-50
DC	6-82
dc	6-49
DCL	1-6
Delta->Star 変換	6-73
Device	1-5, 2-8, 4-4
DIFFerence	6-73
DISPlay グループ	6-27
DMAX	6-82
dmax	6-50
dmin	6-50
DT	6-82

F	ページ
Filename	5-6
FILE グループ	6-44
FLICKer グループ	6-49
Format	2-8
FU	6-82

G	ページ
GET	1-6
GP-IB インタフェースの機能	1-2
GP-IB インタフェースの仕様	1-3
GP-IB ケーブル	1-4
GTL	1-6

H	ページ
HARMonics グループ	6-53
HCOPy グループ	6-55
HOLD グループ	6-59

I	ページ
ID 値	3-5
IEC 高調波測定	6-53
IEEE 488.2-1992 について	付-5
IFC	1-6
IMAGe グループ	6-60
INPut グループ	6-62
INTEGrate グループ	6-70

J	ページ
JPLT	6-84
JPST	6-84
JTOTal	6-84

L	ページ
LCD 画面	6-105
LLO	1-6

M	ページ
MAX HOLD	6-75
MEASure グループ	6-72
MISC	1-5
MOTor グループ	6-77

N	ページ
NL^END	5-1
NL(ニューライン)	5-1
NULL 機能	6-66
NUMeric グループ	6-80

O	ページ
OFF-OFF	2-3

P	ページ
Pc(Corrected Power)	6-75
PC カードの初期化	6-44
PLL ソース	6-53
PLL ソースの警告メッセージ	6-54
PLT	6-82
Plt	6-51
PMT	5-1
PST	6-82
Pst	6-51

R	ページ
RATE グループ	6-97
Register	5-6
Remote Control	1-5, 2-8, 4-4
REN	1-6
RMT	5-1
RS-232 インタフェースの仕様	2-2
RS-232 規定の信号一覧	2-7
RS-232 コントロール	2-8
Rx-Tx	2-8

S	ページ
SDC	1-6
SDC と DCL の違い	1-7
SINGLE	6-110
SNTP	6-104
SPD	1-6
SPE	1-6
Speed	6-77
Star->Delta 変換	6-73
STATus グループ	6-98
STORe グループ	6-100
SyncSpd	6-78
SYSTem グループ	6-104

T	ページ
TCP/IP	4-5
Terminator	2-8
THD	6-54
TIME	6-82
Time/div	6-39
Torque	6-78

U	ページ
UN	6-82
Un	6-52
USB インタフェースの仕様	3-2
USB コントロール	3-4
User Account	4-5

V	ページ
VT 比	6-67

W	ページ
WAVeform グループ	6-107
WT3000 のマニュアル	i

X	ページ
XON-RS	2-4
XON-XON	2-3

ア	ページ
空き容量	6-44
アキュイジションデータ	6-15
圧縮	6-60
アドレス	1-5
アドレスコマンド	1-7
アドレスサブルモード	1-3
アナログ入力	6-78
アペレージング	6-72

イ	ページ
イーサネットインタフェース	4-2
イーサネットコントロールの設定	4-4
異常時のデータ	6-92
位相差	6-76
イネーブルレジスタ	7-2
印刷	6-55
インタフェースメッセージ	1-6

エ	ページ
エラーキュー	6-98, 7-2, 7-8
エラーキューのクリア	6-109
エラーメッセージ	付-2
演算	6-72

オ	ページ
オートキャリブレーション	6-70
オートレンジ	6-62, 6-68
オーバラップイネーブルレジスタ	6-23
オーバラップコマンド	5-7
応答データ	5-2
応答ヘッダ	5-2
応答メッセージ	5-1
オプション	6-110
オペレーションペンディングステータスレジスタ	6-23

カ	ページ
カーソル測定	6-24
開始次数	6-27
回転信号	6-77
回転速度	6-77, 6-78
拡張イベントイネーブルレジスタ	6-98, 7-2
拡張イベントレジスタ	5-8, 6-98, 7-2, 7-7
各部の名称	1-1, 2-1, 3-1, 4-1
カットオフ周波数	6-66, 6-77
紙送り	6-56
画面イメージ	6-56, 6-60
画面イメージデータの保存	6-60
画面表示	6-27
カレントディレクトリ	6-44
観測期間	6-50, 6-82

キ	ページ
記号	iv
機種	6-109
輝度	6-105
基本単位	5-5
キャリブレーション	6-109
共通コマンドグループ	6-109
共通コマンドヘッダ	5-3
極数	6-77

ク	ページ
クエリ	5-4, 5-1
グラフィカル	6-38
グリニッジ標準時	6-104
グルーピング	6-53
クレストファクタ	6-62

ケ	ページ
結線方式	6-69
結線補正	6-73
言語	6-105

コ	ページ
工場出荷時の設定値	6-110
高調波測定	6-53
高調波測定の周波数帯域	6-53
高調波ひずみ率	6-54
構文の記号	iv
効率の演算	6-74
効率補正	6-73
コネクタと信号名(RS-232 の)	2-6
個別設定	6-66
コマンド一覧表	6-1
コメント	6-46, 6-56, 6-60
コントローラとの同期	5-7

サ	ページ
サービスリクエストイネーブルレジスタ	6-110, 7-2
サイクル番号	6-19
最小解析次数	6-53
最大解析次数	6-53
最大相対電圧変化	6-50, 6-82
削除	6-44
差動電圧	6-73
差動電流	6-73
サンプリング周波数	6-76
サンプルレート	6-108

シ	ページ
シーケンシャルコマンド	5-7
色調	6-60
時刻	6-106
システム	6-104
実時間制御積算モード	6-70
周波数測定	6-74
周波数フィルタ	6-65
終了次数	6-27
受信機能	2-2, 4-2
受信バッファ	2-4
出力開始点	6-16
出力キュー	7-2, 7-8
出力終了点	6-15
手動スイッチング	6-51
取得方式	6-52
使用コード	1-3
上位クエリ	5-4
乗数	5-5
状態レジスタ	5-8, 6-98, 7-7

索引

商標	ii
省略形	5-4
初期化	6-110
シリアルポート	6-99, 7-4
シンボル	iv

ス ページ

ズーム率	6-37, 6-39
垂直ポジション	6-38
数値データの出力	6-80
数値データのフォーマット	6-92
数値表示	6-32
数値表示の方式	6-33
数値リストデータ	6-87
スケーリング	6-66
スケーリング(トレンド表示の)	6-36
スケーリング係数	6-77, 6-78, 6-79
スケーリング定数	6-65
スケール値表示	6-39
ステータスバイト	7-3
ステータスバイトレジスタ	6-110
ステータスレポート	7-1
スペクトル形式	6-29

セ ページ

セーブ	6-45, 6-60
積算タイマ	6-71
積算値のリセット	6-70
積算定格時間	6-17
積算のスタート	6-70
積算のストップ	6-70
積算モード	6-70
絶対パス	6-45
設定情報の一覧表示	6-30
設定情報ファイル	6-44
接続	1-4, 2-6, 4-3
接続方式	2-2
セルフテスト	6-111
遷移フィルタ	6-98, 7-7

ソ ページ

送信機能	2-2, 4-2
相対定常電圧変化	6-49, 6-82
相対電圧変化時間	6-50, 6-82
測定回数	6-49
測定経過時間	6-82
測定次数	6-53
ソフトウェアハンドシェーク	2-3

タ ページ

ターミネータ	2-9
対象ドライブ	6-44
タイムアウト時間	4-5, 4-6, 6-20
単位	iv, 5-5
短期間フリッカ値	6-51, 6-82
単純ヘッダ	5-3

チ ページ

注記	iv
中心位置のレベル	6-38
長期間フリッカ値	6-51, 6-82

ツ ページ

通信コマンド	6-1
通信方式	2-2

テ ページ

データ	5-5
データ更新周期	6-97
データバイトの並び	5-6
データフォーマット	2-5, 2-9
定格電圧	6-52
定常範囲	6-50
ディレクトリ	6-45
デッドロック	5-2
デルタ演算	6-73
電圧実効値	6-82
電圧周波数	6-82
電圧測定	6-68
電圧レンジ	6-68, 6-78
電流センサスケーリング定数	6-65
電流測定	6-62
電流モード	6-63
電流レンジ	6-63
電力係数	6-67

ト ページ

トータ機能	1-2
同期ソース	6-20, 6-67
同期測定モード	6-76
同期速度	6-78
同期方式	2-2
トリガ	6-21, 6-39
トルク	6-78
トレンド	6-35

ナ ページ

内蔵プリンタでの印刷	6-56
------------------	------

ニ	ページ
入力エレメント	6-62
入力エレメントタイプ	6-66
入力エレメントの個別設定	6-66
入力タイプ	6-78
入力フィルタ	6-77

ハ	ページ
バーグラフ	6-27
ハードウェアハンドシェーク	2-4
バイト出力順序	6-107
波形サンプリングデータ	6-15
波形の全データ点数	6-107
波形の割り付け方法	6-38
波形表示	6-37
波形表示データ	6-107
波形ラベル名表示	6-39
パスワード	4-5, 4-6
パターン	6-93, 6-95
パルス数	6-78
パルス入力	6-78
ハンドシェーク	2-3, 2-9

ヒ	ページ
ピークオーバー	6-66
ピーク情報	6-92
皮相電力	6-76
日付	6-104
表示色	6-105
表示色のモード	6-105
表示フォーマット(トレンドの)	6-36
表示フォーマット(バーグラフの)	6-27
表示フォーマット(波形の)	6-38
表示方式	6-32
標準イベントイネーブルレジスタ	6-109, 7-2
標準イベントレジスタ	6-109, 7-2, 7-5
標準積算モード	6-70

フ	ページ
ファイル操作	6-44
ファイルの保存	6-45
ファイルの読み込み	6-44
ファイル名	6-45, 6-61
ファンクション選択肢(<Function>)一覧	6-41
ファンクション名	6-41, 6-43
フィルタ	6-19, 6-65
複合ヘッダ	5-3
物理的な次元を持ったデータ	5-5
プリセットパターン	6-93, 6-95
フリッカ測定対象周波数	6-50
フリッカ測定方法	6-51
フリッカ判定結果	6-84

プリンタ	6-56
プログラムデータ	5-1
プログラムヘッダ	5-1
プログラムメッセージ	5-1
ブロックデータ	5-6
プロトコル	1-3
フロントパネル	1-1, 2-1, 3-1, 4-1
分割フォーマット	6-38

ヘ	ページ
ベクトル表示	6-37
ヘッダ	5-1, 5-3
ヘッダの解釈	5-4

ホ	ページ
ホールド	6-59
ボーレート	2-9
補間方式	6-38
補正演算	6-73
保存	6-45, 6-60

マ	ページ
マスク	7-3, 7-5
窓関数	6-29
マニュアル	i
マニュアルスケーリング(トレンドの)	6-37
マニュアルで使用している記号	iv
マルチラインメッセージ	1-7

ム	ページ
無効電力	6-76

メ	ページ
メッセージの言語	6-105
メニューの言語	6-105

モ	ページ
モータ出力	6-77
モータ評価	6-77
文字データ	5-6
文字列データ	5-6

ユ	ページ
ユーザー定義ファンクション	6-75
ユーザー認証機能	4-3
ユーザー名	4-5, 4-6
有償ソフトウェア	4-6
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	ページ
横軸(T/div)(トレンドの)	6-37
読み込み	6-44

ラ	ページ
ラインフィルタ	6-20, 6-66, 6-77

リ	ページ
リアパネル	1-1, 2-1, 3-1, 4-1
リスト表示	6-33
リスナ機能	1-2
リモート / ローカル	6-22
リモート / ローカル切り替え	1-2, 2-2, 4-2
履歴	ii

レ	ページ
レジスタのクリア	6-110
連続積算モード	6-70

ロ	ページ
ローカルロックアウト	6-22
ロード	6-44

コマンドリスト	ページ
:ACquisition:BYTeorder	6-15
:ACquisition:END	6-15
:ACquisition:FORMat	6-15
:ACquisition:HOLD	6-15
:ACquisition:LENGth?	6-16
:ACquisition:SEND?	6-16
:ACquisition:SRATe?	6-16
:ACquisition:STARt	6-16
:ACquisition:TRACe	6-16
:ACquisition?	6-15
:AOUTput:NORMal?	6-17
:AOUTput?	6-17
:AOUTput[:NORMal]:CHANnel<x>	6-17
:AOUTput[:NORMal]:IRTime	6-17
:AOUTput[:NORMal]:MODE<x>	6-18
:AOUTput[:NORMal]:RATE<x>	6-18
:CBCycle:COUNt	6-19
:CBCycle:DISPlay:CURSor	6-19
:CBCycle:DISPlay:ITEM<x>	6-19
:CBCycle:DISPlay:PAGE	6-19
:CBCycle:DISPlay?	6-19
:CBCycle:FILTer:LINE?	6-20
:CBCycle:FILTer?	6-19
:CBCycle:FILTer[:LINE]:ELEMeNt<x>	6-20
:CBCycle:FILTer[:LINE]:MOTor	6-20
:CBCycle:FILTer[:LINE][:ALL]	6-20
:CBCycle:RESet	6-20
:CBCycle:STATe?	6-20
:CBCycle:SYNChronize:SLOPe	6-20
:CBCycle:SYNChronize:SOURce	6-20
:CBCycle:SYNChronize?	6-20
:CBCycle:TIMEout	6-20
:CBCycle:TRIGger:LEVel	6-21
:CBCycle:TRIGger:MODE	6-21
:CBCycle:TRIGger:SLOPe	6-21
:CBCycle:TRIGger:SOURce	6-21
:CBCycle:TRIGger?	6-21
:CBCycle?	6-19
:COMMunicate:HEADer	6-22
:COMMunicate:LOCKout	6-22
:COMMunicate:OPSE	6-22
:COMMunicate:OPSR?	6-22
:COMMunicate:OVERlap	6-22
:COMMunicate:REMOte	6-22
:COMMunicate:STATus?	6-23
:COMMunicate:VERBoSe	6-23
:COMMunicate:WAIT	6-23
:COMMunicate:WAIT?	6-23
:COMMunicate?	6-22
:CURSor:BAR:{Y<x> DY}?	6-24
:CURSor:BAR:POSition<x>	6-24
:CURSor:BAR?	6-24
:CURSor:BAR[:STATe]	6-24
:CURSor:FFT:{X<x> DX Y<x> DY}?	6-25
:CURSor:FFT:POSition<x>	6-24

:CURSor:FFT:TRACe<x>	6-25	ITEM<x>	6-35
:CURSor:FFT?	6-24	:DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}:	
:CURSor:FFT[:STATe]	6-25	PAGE	6-35
:CURSor:TREnd:{X<x> Y<x> DY}?	6-25	:DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}:	
:CURSor:TREnd:POSition<x>	6-25	PRESet	6-35
:CURSor:TREnd:TRACe<x>	6-25	:DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}? ..	6-34
:CURSor:TREnd?	6-25	:DISPlay:NUMeric[:NORMal]:ALL:CURSor	6-32
:CURSor:TREnd[:STATe]	6-25	:DISPlay:NUMeric[:NORMal]:ALL:ORDer	6-33
:CURSor:WAVE:{X<x> DX PERDt Y<x> DY}?	6-26	:DISPlay:NUMeric[:NORMal]:ALL:PAGE	6-33
:CURSor:WAVE:PATH	6-26	:DISPlay:NUMeric[:NORMal]:ALL?	6-32
:CURSor:WAVE:POSition<x>	6-26	:DISPlay:NUMeric[:NORMal]:FORMat	6-33
:CURSor:WAVE:TRACe<x>	6-26	:DISPlay:NUMeric[:NORMal]:LIST:CURSor	6-33
:CURSor:WAVE?	6-26	:DISPlay:NUMeric[:NORMal]:LIST:HEADer	6-34
:CURSor:WAVE[:STATe]	6-26	:DISPlay:NUMeric[:NORMal]:LIST:ITEM<x>	6-34
:CURSor?	6-24	:DISPlay:NUMeric[:NORMal]:LIST:ORDer	6-34
:DISPlay:BAR:FORMat	6-27	:DISPlay:NUMeric[:NORMal]:LIST?	6-33
:DISPlay:BAR:ITEM<x>	6-27	:DISPlay:TREnd:ALL	6-36
:DISPlay:BAR:ORDer	6-27	:DISPlay:TREnd:CLEar	6-36
:DISPlay:BAR?	6-27	:DISPlay:TREnd:FORMat	6-36
:DISPlay:CBCycle:CURSor	6-28	:DISPlay:TREnd:ITEM<x>:SCALing:MODE	6-36
:DISPlay:CBCycle:ITEM<x>	6-28	:DISPlay:TREnd:ITEM<x>:SCALing:VALue	6-37
:DISPlay:CBCycle:PAGE	6-28	:DISPlay:TREnd:ITEM<x>:SCALing?	6-36
:DISPlay:CBCycle?	6-28	:DISPlay:TREnd:ITEM<x>?	6-36
:DISPlay:FFT:FFT<x>:LABel	6-28	:DISPlay:TREnd:ITEM<x>[:FUNCTION]	6-36
:DISPlay:FFT:FFT<x>:OBJect	6-29	:DISPlay:TREnd:T<x>	6-37
:DISPlay:FFT:FFT<x>?	6-28	:DISPlay:TREnd:TDIV	6-37
:DISPlay:FFT:FFT<x>[:STATe]	6-29	:DISPlay:TREnd?	6-35
:DISPlay:FFT:FORMat	6-29	:DISPlay:VECTor:{UMAG IMAG}	6-37
:DISPlay:FFT:POINt	6-29	:DISPlay:VECTor:NUMeric	6-37
:DISPlay:FFT:SCOPE	6-29	:DISPlay:VECTor:OBJect	6-37
:DISPlay:FFT:SPECTrum	6-29	:DISPlay:VECTor?	6-37
:DISPlay:FFT:VSCale	6-29	:DISPlay:WAVE:{U<x> I<x> SPEed TORQue MATH<x>}	6-39
:DISPlay:FFT:WINDow	6-29		6-39
:DISPlay:FFT?	6-28	:DISPlay:WAVE:ALL	6-38
:DISPlay:FLICKer:ELEMent	6-30	:DISPlay:WAVE:FORMat	6-38
:DISPlay:FLICKer:PAGE	6-30	:DISPlay:WAVE:GRATicule	6-38
:DISPlay:FLICKer:PERiod	6-30	:DISPlay:WAVE:INTERpolate	6-38
:DISPlay:FLICKer?	6-29	:DISPlay:WAVE:MAPPING:	
:DISPlay:INFORmation:PAGE	6-30	{U<x> I<x> SPEed TORQue MATH}	6-38
:DISPlay:INFORmation?	6-30	:DISPlay:WAVE:MAPPING?	6-38
:DISPlay:INFORmation[:STATe]	6-30	:DISPlay:WAVE:MAPPING[:MODE]	6-38
:DISPlay:MATH:CONStant<x>	6-30	:DISPlay:WAVE:POSition:{U<x> I<x>}	6-38
:DISPlay:MATH:MATH<x>:EXPRession	6-31	:DISPlay:WAVE:POSition:{UALL IALL}	6-38
:DISPlay:MATH:MATH<x>:LABel	6-31	:DISPlay:WAVE:POSition?	6-38
:DISPlay:MATH:MATH<x>:SCALing:CENTer	6-31	:DISPlay:WAVE:SVALue	6-39
:DISPlay:MATH:MATH<x>:SCALing:MODE	6-31	:DISPlay:WAVE:TDIV	6-39
:DISPlay:MATH:MATH<x>:SCALing:SDIV	6-31	:DISPlay:WAVE:TLABel	6-39
:DISPlay:MATH:MATH<x>:SCALing?	6-31	:DISPlay:WAVE:TRIGger:LEVel	6-39
:DISPlay:MATH:MATH<x>:UNIT	6-32	:DISPlay:WAVE:TRIGger:MODE	6-39
:DISPlay:MATH:MATH<x>?	6-31	:DISPlay:WAVE:TRIGger:SLOPe	6-39
:DISPlay:MATH?	6-30	:DISPlay:WAVE:TRIGger:SOURce	6-39
:DISPlay:MODE	6-32	:DISPlay:WAVE:TRIGger?	6-39
:DISPlay:NUMeric:NORMal?	6-32	:DISPlay:WAVE:VZoom:{U<x> I<x>}	6-40
:DISPlay:NUMeric?	6-32	:DISPlay:WAVE:VZoom:{UALL IALL}	6-40
:DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}:		:DISPlay:WAVE:VZoom?	6-39
CURSor	6-35	:DISPlay:WAVE?	6-37
:DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}:		:DISPlay?	6-27

:FILE:CDIRectory	6-44	:FLICKer:ELEMeNt<x>	6-50
:FILE:DELeTe:IMAGe:{TIFF BMP PScriPt PNG JPEG}	6-44	:FLICKer:FREQuency	6-50
:FILE:DELeTe:NUMeric:{ASCIi FLOat}	6-44	:FLICKer:INIitialize	6-50
:FILE:DELeTe:SEtUp	6-44	:FLICKer:INTErval	6-51
:FILE:DELeTe:WAVE:{BINary ASCIi FLOat}	6-44	:FLICKer:JUDGE	6-51
:FILE:DRIVE	6-44	:FLICKer:MEASurement	6-51
:FILE:FORMat:EXECute	6-44	:FLICKer:MOVE	6-51
:FILE:FREE?	6-44	:FLICKer:PLT:LIMit	6-51
:FILE:LOAD:ABORT	6-44	:FLICKer:PLT:NVALue	6-51
:FILE:LOAD:SEtUp	6-44	:FLICKer:PLT?	6-51
:FILE:MDIRectory	6-45	:FLICKer:PLT[:STATe]	6-51
:FILE:PATH?	6-45	:FLICKer:PST:LIMit	6-51
:FILE:SAVE:ABORT	6-45	:FLICKer:PST?	6-51
:FILE:SAVE:ACQuisiTion:TRACe	6-45	:FLICKer:PST[:STATe]	6-52
:FILE:SAVE:ACQuisiTion:TYPE	6-45	:FLICKer:RESeT	6-52
:FILE:SAVE:ACQuisiTion?	6-45	:FLICKer:STARt	6-52
:FILE:SAVE:ACQuisiTion[:EXECute]	6-45	:FLICKer:STATe?	6-52
:FILE:SAVE:ANAMing	6-45	:FLICKer:UN:MODE	6-52
:FILE:SAVE:COMMeNt	6-46	:FLICKer:UN:VALue	6-52
:FILE:SAVE:NUMeric:CBCCycle:		:FLICKer:UN?	6-52
{ELEMeNt<x> SIGMA SIGMB}	6-46	:FLICKer?	6-49
:FILE:SAVE:NUMeric:CBCCycle:<Function>	6-46	:HARMonics:FBAND	6-53
:FILE:SAVE:NUMeric:CBCCycle:ALL	6-46	:HARMonics:IEC:{UGRouping IGRouping}	6-53
:FILE:SAVE:NUMeric:CBCCycle?	6-46	:HARMonics:IEC:OBJect	6-53
:FILE:SAVE:NUMeric:NORMal?	6-47	:HARMonics:IEC?	6-53
:FILE:SAVE:NUMeric:TYPE	6-48	:HARMonics:ORDER	6-53
:FILE:SAVE:NUMeric?	6-46	:HARMonics:PLLSource	6-53
:FILE:SAVE:NUMeric[:EXECute]	6-47	:HARMonics:PLLWarning?	6-54
:FILE:SAVE:NUMeric[:NORMal]:		:HARMonics:PLLWarning[:STATe]	6-54
{ELEMeNt<x> SIGMA SIGMB}	6-47	:HARMonics:THD	6-54
:FILE:SAVE:NUMeric[:NORMal]:<Function>	6-47	:HARMonics?	6-53
:FILE:SAVE:NUMeric[:NORMal]:ALL	6-47	:HCOPy:ABORT	6-55
:FILE:SAVE:NUMeric[:NORMal]:PRESeT<x>	6-47	:HCOPy:AUTO:{STARt END}	6-55
:FILE:SAVE:SEtUp[:EXECute]	6-48	:HCOPy:AUTO:INTErval	6-55
:FILE:SAVE:WAVE:TRACe	6-48	:HCOPy:AUTO:SYNChronize	6-55
:FILE:SAVE:WAVE:TYPE	6-48	:HCOPy:AUTO?	6-55
:FILE:SAVE:WAVE?	6-48	:HCOPy:AUTO[:STATe]	6-55
:FILE:SAVE:WAVE[:EXECute]	6-48	:HCOPy:COMMeNt	6-56
:FILE:SAVE?	6-45	:HCOPy:DIRectiOn	6-56
:FILE?	6-44	:HCOPy:EXECute	6-56
:FLICKer:COUNT	6-49	:HCOPy:NETPrint:COLor	6-56
:FLICKer:DC:LIMit	6-49	:HCOPy:NETPrint:FORMat	6-56
:FLICKer:DC?	6-49	:HCOPy:NETPrint?	6-56
:FLICKer:DC[:STATe]	6-49	:HCOPy:PRINter:FEED	6-56
:FLICKer:DISPlay:ELEMeNt	6-49	:HCOPy:PRINter:FORMat	6-56
:FLICKer:DISPlay:PAGE	6-49	:HCOPy:PRINter:LIST:INFORMATION	6-57
:FLICKer:DISPlay:PERiod	6-50	:HCOPy:PRINter:LIST:NORMal?	6-57
:FLICKer:DISPlay?	6-49	:HCOPy:PRINter:LIST?	6-57
:FLICKer:DMAX:LIMit	6-50	:HCOPy:PRINter:LIST[:NORMal]:	
:FLICKer:DMAX?	6-50	{ELEMeNt<x> SIGMA SIGMB}	6-57
:FLICKer:DMAX[:STATe]	6-50	:HCOPy:PRINter:LIST[:NORMal]:<Function>	6-58
:FLICKer:DMIN:LIMit	6-50	:HCOPy:PRINter:LIST[:NORMal]:ALL	6-57
:FLICKer:DMIN?	6-50	:HCOPy:PRINter:LIST[:NORMal]:PRESeT<x>	6-57
:FLICKer:DT:LIMit	6-50	:HCOPy:PRINter?	6-56
:FLICKer:DT?	6-50	:HCOPy?	6-55
:FLICKer:DT[:STATe]	6-50	:HOLD	6-59
		:IMAGe:ABORT	6-60

:IMAGe:COLor	6-60	:MOTor:PM?	6-77
:IMAGe:COMMeNt	6-60	:MOTor:POLE	6-77
:IMAGe:COMPreSSion	6-60	:MOTor:SPEed:AUTO	6-78
:IMAGe:EXECute	6-60	:MOTor:SPEed:PRANge	6-78
:IMAGe:FORMat	6-60	:MOTor:SPEed:PULSe	6-78
:IMAGe:SAVE:ANAMing	6-60	:MOTor:SPEed:RANge	6-78
:IMAGe:SAVE:CDIRectory	6-60	:MOTor:SPEed:SCALing	6-78
:IMAGe:SAVE:DRIVE	6-61	:MOTor:SPEed:TYPE	6-78
:IMAGe:SAVE:NAME	6-61	:MOTor:SPEed:UNIT	6-78
:IMAGe:SAVE?	6-60	:MOTor:SPEed?	6-77
:IMAGe:SEND?	6-61	:MOTor:SSpeed	6-78
:IMAGe?	6-60	:MOTor:TORQue:AUTO	6-79
:INPut?	6-62	:MOTor:TORQue:PRANge	6-79
:INTEGrate:ACAL	6-70	:MOTor:TORQue:RANge	6-79
:INTEGrate:MODE	6-70	:MOTor:TORQue:RATE:{UPPer LOWer}	6-79
:INTEGrate:RESet	6-70	:MOTor:TORQue:RATE?	6-79
:INTEGrate:RTIME:{STARt ENd}	6-70	:MOTor:TORQue:SCALing	6-79
:INTEGrate:RTIME?	6-70	:MOTor:TORQue:TYPE	6-79
:INTEGrate:STARt	6-70	:MOTor:TORQue:UNIT	6-80
:INTEGrate:STATe?	6-70	:MOTor:TORQue?	6-79
:INTEGrate:STOP	6-70	:MOTor?	6-77
:INTEGrate:TImer<x>	6-71	:NUMeric:CBCycle:END	6-80
:INTEGrate?	6-70	:NUMeric:CBCycle:ITEM	6-80
:MEASure:AVERaging:COUNT	6-72	:NUMeric:CBCycle:STARt	6-80
:MEASure:AVERaging:TYPE	6-73	:NUMeric:CBCycle:VALue?	6-81
:MEASure:AVERaging?	6-72	:NUMeric:CBCycle?	6-80
:MEASure:AVERaging[:STATe]	6-72	:NUMeric:FLICKer:COUNT?	6-81
:MEASure:COMPensation:EFFiciency	6-73	:NUMeric:FLICKer:FUNCTion:CLear	6-82
:MEASure:COMPensation:V3A3	6-73	:NUMeric:FLICKer:FUNCTion:DELeTe	6-82
:MEASure:COMPensation:WIRing:ELEMent<x>	6-73	:NUMeric:FLICKer:FUNCTion:ITEM<x>	6-82
:MEASure:COMPensation:WIRing?	6-73	:NUMeric:FLICKer:FUNCTion:NUMBER	6-83
:MEASure:COMPensation?	6-73	:NUMeric:FLICKer:FUNCTion:VALue?	6-83
:MEASure:DMeasure:SIGMB	6-74	:NUMeric:FLICKer:FUNCTion?	6-81
:MEASure:DMeasure?	6-73	:NUMeric:FLICKer:INFORMation:CLear	6-84
:MEASure:DMeasure[:SIGMA]	6-73	:NUMeric:FLICKer:INFORMation:DELeTe	6-84
:MEASure:EFFiciency:ETA<x>	6-74	:NUMeric:FLICKer:INFORMation:ITEM<x>	6-84
:MEASure:EFFiciency:UDEF<x>	6-74	:NUMeric:FLICKer:INFORMation:NUMBER	6-85
:MEASure:EFFiciency?	6-74	:NUMeric:FLICKer:INFORMation:VALue?	6-85
:MEASure:FREQUency:ITEM<x>	6-74	:NUMeric:FLICKer:INFORMation?	6-83
:MEASure:FREQUency?	6-74	:NUMeric:FLICKer:JUDGement?	6-85
:MEASure:FUNCTion<x>:EXPRession	6-75	:NUMeric:FLICKer:PERiod?	6-86
:MEASure:FUNCTion<x>:UNIT	6-75	:NUMeric:FLICKer:VALue?	6-86
:MEASure:FUNCTion<x>?	6-75	:NUMeric:FLICKer?	6-81
:MEASure:FUNCTion<x>[:STATe]	6-75	:NUMeric:FORMat	6-87
:MEASure:MHOLd	6-75	:NUMeric:HOLd	6-87
:MEASure:PC:IEC	6-75	:NUMeric:LIST:CLear	6-88
:MEASure:PC:P<x>	6-76	:NUMeric:LIST:DELeTe	6-88
:MEASure:PC?	6-75	:NUMeric:LIST:ITEM<x>	6-88
:MEASure:PHASe	6-76	:NUMeric:LIST:NUMBER	6-88
:MEASure:SAMPLing	6-76	:NUMeric:LIST:ORDER	6-89
:MEASure:SQFormula	6-76	:NUMeric:LIST:PRESet	6-89
:MEASure:SYNChronize	6-77	:NUMeric:LIST:SELeCT	6-89
:MEASure?	6-72	:NUMeric:LIST:VALue?	6-89
:MOTor:FILTer?	6-77	:NUMeric:LIST?	6-88
:MOTor:FILTer[:LINE]	6-77	:NUMeric:NORMal?	6-90
:MOTor:PM:SCALing	6-77	:NUMeric?	6-80
:MOTor:PM:UNIT	6-77	:NUMeric[:NORMal]:CLear	6-90

:NUMeric[:NORMal]:DELeTe	6-90	:SYSTem:LANGUage:MENU	6-105
:NUMeric[:NORMal]:ITEM<x>	6-90	:SYSTem:LANGUage:MESSage	6-105
:NUMeric[:NORMal]:NUMber	6-90	:SYSTem:LANGUage?	6-105
:NUMeric[:NORMal]:PRESet	6-90	:SYSTem:LCD:BRIGhtness	6-105
:NUMeric[:NORMal]:VALue?	6-91	:SYSTem:LCD:COLor:GRAPh:	
:RATE	6-98	{BACKground GRATicule CURSor}	6-105
:STATus:CONDition?	6-98	:SYSTem:LCD:COLor:GRAPh:MODE	6-106
:STATus:EESE	6-98	:SYSTem:LCD:COLor:GRAPh?	6-105
:STATus:EESR?	6-98	:SYSTem:LCD:COLor:TEXT:	
:STATus:ERRor?	6-98	{LETter BACKground BOX SUB SELecte}	6-106
:STATus:FILTer<x>	6-98	:SYSTem:LCD:COLor:TEXT:MODE	6-106
:STATus:QENable	6-98	:SYSTem:LCD:COLor:TEXT?	6-106
:STATus:QMESsage	6-99	:SYSTem:LCD:COLor?	6-105
:STATus:SPOLi?	6-104	:SYSTem:LCD?	6-105
:STATus?	6-98	:SYSTem:SLOCK	6-106
:STORe:COUNT	6-100	:SYSTem:TIME	6-107
:STORe:DIRection	6-100	:SYSTem:USBKeyboard	6-106
:STORe:FILE:ANAMing	6-100	:SYSTem?	6-104
:STORe:FILE:COMMeNt	6-100	:WAVEform:BYTeorder	6-107
:STORe:FILE:NAME	6-100	:WAVEform:END	6-107
:STORe:FILE:TYPE	6-100	:WAVEform:FORMat	6-107
:STORe:FILE?	6-100	:WAVEform:HOLD	6-107
:STORe:INTerval	6-101	:WAVEform:LENGth?	6-108
:STORe:ITEM	6-101	:WAVEform:SEND?	6-108
:STORe:MEMory:ALERT	6-101	:WAVEform:SRATe?	6-108
:STORe:MEMory:CONVert:ABORT	6-101	:WAVEform:STARt	6-108
:STORe:MEMory:CONVert:EXECute	6-101	:WAVEform:TRACe	6-108
:STORe:MEMory:INITialize	6-101	:WAVEform:TRIGger?	6-109
:STORe:MEMory?	6-101	:WAVEform?	6-107
:STORe:MODE	6-101	[[:INPut]:CFACtor	6-62
:STORe:NUMeric:NORMal?	6-102	[[:INPut]:CURRent:AUTO:{SIGMA SIGMB}	6-63
:STORe:NUMeric?	6-102	[[:INPut]:CURRent:AUTO:ELEMeNt<x>	6-63
:STORe:NUMeric[:NORMal]:		[[:INPut]:CURRent:AUTO?	6-62
{ELEMeNt<x> SIGMA SIGMB}	6-102	[[:INPut]:CURRent:AUTO[:ALL]	6-62
:STORe:NUMeric[:NORMal]:<Function>	6-102	[[:INPut]:CURRent:MODE:{SIGMA SIGMB}	6-63
:STORe:NUMeric[:NORMal]:ALL	6-102	[[:INPut]:CURRent:MODE:ELEMeNt<x>	6-63
:STORe:NUMeric[:NORMal]:PRESet<x>	6-102	[[:INPut]:CURRent:MODE?	6-63
:STORe:RECall	6-103	[[:INPut]:CURRent:MODE[:ALL]	6-63
:STORe:RTIME:{STARt END}	6-103	[[:INPut]:CURRent:RANGe:{SIGMA SIGMB}	6-65
:STORe:RTIME?	6-103	[[:INPut]:CURRent:RANGe:ELEMeNt<x>	6-64
:STORe:SMODE	6-103	[[:INPut]:CURRent:RANGe?	6-63
:STORe:STARt	6-103	[[:INPut]:CURRent:RANGe[:ALL]	6-64
:STORe:STOP	6-103	[[:INPut]:CURRent:SRATio:ELEMeNt<x>	6-65
:STORe:WAVE:{U<x> I<x> SPEEd TORQue}	6-103	[[:INPut]:CURRent:SRATio?	6-65
:STORe:WAVE:ALL	6-103	[[:INPut]:CURRent:SRATio[:ALL]	6-65
:STORe:WAVE?	6-103	[[:INPut]:CURRent?	6-62
:STORe?	6-100	[[:INPut]:FILTer:FREQuency:ELEMeNt<x>	6-66
:SYSTem:CLOCK:DISPlay	6-104	[[:INPut]:FILTer:FREQuency?	6-65
:SYSTem:CLOCK:SNTP:GMTTime	6-104	[[:INPut]:FILTer:FREQuency[:ALL]	6-65
:SYSTem:CLOCK:SNTP?	6-104	[[:INPut]:FILTer:LINE?	6-66
:SYSTem:CLOCK:SNTP[:EXECute]	6-104	[[:INPut]:FILTer?	6-65
:SYSTem:CLOCK:TYPE	6-104	[[:INPut]:FILTer[:LINE]:ELEMeNt<x>	6-66
:SYSTem:CLOCK?	6-104	[[:INPut]:FILTer[:LINE][:ALL]	6-66
:SYSTem:DATE	6-104	[[:INPut]:INDepeNdeNt	6-66
:SYSTem:ECLear	6-105	[[:INPut]:MODULE?	6-66
:SYSTem:FONT	6-105	[[:INPut]:NULL	6-66
:SYSTem:KLOCK	6-105	[[:INPut]:POVer?	6-66

[:INPut]:SCALing:{VT CT SFACtor}:ELEMeNt<x>	6-67
[:INPut]:SCALing:{VT CT SFACtor}?	6-67
[:INPut]:SCALing:{VT CT SFACtor}[:ALL]	6-67
[:INPut]:SCALing:STATe?	6-67
[:INPut]:SCALing?	6-66
[:INPut]:SCALing[:STATe]:ELEMeNt<x>	6-67
[:INPut]:SCALing[:STATe][:ALL]	6-67
[:INPut]:SYNChronize:{SIGMA SIGMB}	6-67
[:INPut]:SYNChronize:ELEMeNt<x>	6-67
[:INPut]:SYNChronize?	6-67
[:INPut]:SYNChronize[:ALL]	6-67
[:INPut]:VOL TAge:AUTO:{SIGMA SIGMB}	6-68
[:INPut]:VOL TAge:AUTO:ELEMeNt<x>	6-68
[:INPut]:VOL TAge:AUTO?	6-68
[:INPut]:VOL TAge:AUTO[:ALL]	6-68
[:INPut]:VOL TAge:MODE:{SIGMA SIGMB}	6-68
[:INPut]:VOL TAge:MODE:ELEMeNt<x>	6-68
[:INPut]:VOL TAge:MODE?	6-68
[:INPut]:VOL TAge:MODE[:ALL]	6-68
[:INPut]:VOL TAge:RANGe:{SIGMA SIGMB}	6-69
[:INPut]:VOL TAge:RANGe:ELEMeNt<x>	6-69
[:INPut]:VOL TAge:RANGe?	6-68
[:INPut]:VOL TAge:RANGe[:ALL]	6-69
[:INPut]:VOL TAge?	6-68
[:INPut]:WIRing	6-69
*CAL ?	6-109
*CLS	6-109
*ESE	6-109
*ESR?	6-109
*IDN?	6-109
*OPC	6-110
*OPT?	6-110
*PSC	6-110
*RST	6-110
*SRE	6-110
*STB?	6-110
*TRG	6-111
*TST?	6-111
*WAI	6-111