

WT1800

プレシジョンパワーアナライザ
通信インタフェース

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

ユーザーズマニュアル

はじめに

このたびは、プレシジョンパワーアナライザ WT1800 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、下記の各インタフェースの機能やコマンドについて説明したものです。

- ・イーサネットインタフェース
- ・USB インタフェース
- ・GP-IB インタフェース

ご使用前にこのマニュアルをよくお読みいただき、正しくお使いください。

お読みになったあとは大切に保存してください。ご使用中に操作がわからなくなったときなどにきつとお役に立ちます。なお、WT1800 のマニュアルとして、このマニュアルを含め、次のものがあります。あわせてお読みください。

マニュアル名	マニュアル No	内容
WT1800 プレシジョンパワーアナライザ ユーザーズマニュアル [機能編]	IM WT1801-01JA	付属の CD に pdf データが納められています。通信インタフェースの機能を除く、本機器の全機能について説明しています。
WT1800 プレシジョンパワーアナライザ ユーザーズマニュアル [操作編]	IM WT1801-02JA	付属の CD に pdf データが納められています。本機器の各設定操作について説明しています。
WT1800 プレシジョンパワーアナライザ スタートガイド	IM WT1801-03JA	本機器の取り扱い上の注意、機能概要、基本的な操作や仕様について、説明しています。
WT1800 プレシ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	IM WT1801-17JA	本書です。付属の CD に pdf データが納められています。本機器の通信インタフェースの機能について、その操作方法を説明しています。
Model WT1800 Precision Power Analyzer	IM WT1801-92Z1	中国向け文書

マニュアル No. の「JA」、「Z1」は言語コードです。

各国や地域の当社営業拠点の連絡先は、次のシートに記載されています。

ドキュメント No.	内容
PIM 113-01Z2	国内海外の連絡先一覧

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、実際の画面表示内容が本書に記載の画面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、お買い求め先か、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。
- ・保証書が付いています。再発行はいたしません。よくお読みいただき、ご理解のうえ大切に保存してください。
- ・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。

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または商標です。
- ・Adobe、Acrobat は、アドビシステムズ社の登録商標または商標です。
- ・本文中の各社の登録商標または商標には、®、TM マークは表示していません。
- ・その他、本文中に使われている会社名、商品名は、各社の登録商標または商標です。

履歴

- ・ 2011 年 2 月 初版発行
- ・ 2011 年 8 月 2 版発行
- ・ 2015 年 12 月 3 版発行
- ・ 2017 年 10 月 4 版発行

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。
 - WT1800 用ライブラリ (TMCTL)
 - PC-WT1800 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。
 - WT1800 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 Web サイトの提供ページからダウンロードできます。
<http://www.yokogawa.com/jp-ymi/tm/F-SOFT/>

このマニュアルの利用方法

このマニュアルの構成

このユーザーズマニュアルは、以下に示す第 1 章～第 6 章および付録で構成されています。

第 1 章 イーサネットインタフェースについて

イーサネットインタフェースの機能・仕様などについて説明しています。

第 2 章 USB インタフェースについて

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 3 章 GP-IB インタフェースについて

GP-IB インタフェースの機能・仕様などについて説明しています。

第 4 章 プログラムを組む前に

コマンドを送るときの書式などについて説明しています。

第 5 章 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 6 章 ステータスレポート

ステータスバイトや各種レジスタ、キューなどについて説明しています。

付 録

エラーメッセージなどについて説明しています。

索 引

50 音順、アルファベット順の 2 種類の索引があります。

このマニュアルで使用している記号と表記法

注記

このマニュアルでは、注記を以下のようなシンボルで区別しています。

警告 取り扱いを誤った場合に、使用者が死亡または重傷を負う危険があるときに、その危険を避けるための注意事項が記載されています。

注意 取り扱いを誤った場合に、使用者が軽傷を負うか、または物的損害のみが発生する危険があるときに、それを避けるための注意事項が記載されています。

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。

文字の表記法

太文字の操作キー名とソフトキー名

操作対象になる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。押した操作キーの下に紫色で記されている項目の設定メニューが画面に表示されます。

単位

k	「1000」の意味です。使用例：100kHz
K	「1024」の意味です。使用例：720K バイト (ファイルのデータサイズ)

構文の記号

主に第 4、5 章の構文で使用している記号を下表に示します。なお、これは BNF(Backus-Naur Form) 記号と呼ばれるものです。詳細データについては、4-6 ～ 4-7 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 ～ 6	ELEMENT2
{ } 	{ } 内から 1 つを選択 排他的論理和	SQFormula {TYPE1 TYPE2 TYPE3}	SQFORMULA TYPE1
[]	省略可能	NUMeric[:NORMal]:VALue?	NUMERIC:VALUE?

目次

このマニュアルの利用方法	iii
--------------------	-----

第 1 章

イーサネットインタフェースについて

1.1 各部の名称と機能	1-1
1.2 イーサネットインタフェースの機能と仕様	1-2
1.3 イーサネットインタフェースによる接続	1-3
1.4 本体の設定 (イーサネット)	1-4

第 2 章

USB インタフェースについて

2.1 各部の名称と機能	2-1
2.2 USB インタフェースの機能と仕様	2-2
2.3 USB インタフェースによる接続	2-3
2.4 本体の設定 (USB)	2-4

第 3 章

GP-IB インタフェースについて

3.1 各部の名称と機能	3-1
3.2 GP-IB インタフェースの機能と仕様	3-2
3.3 GP-IB インタフェースによる接続	3-4
3.4 本体の設定 (GP-IB)	3-6
3.5 インタフェースメッセージに対する応答	3-7

第 4 章

プログラムを組む前に

4.1 メッセージ	4-1
4.2 命令	4-3
4.3 応答	4-5
4.4 データ	4-6
4.5 コントローラとの同期	4-8

第 5 章

コマンド

5.1 コマンド一覧表	5-1
5.2 AOUPut グループ	5-16
5.3 AUX グループ	5-17
5.4 COMMunicate グループ	5-19
5.5 CURSor グループ	5-21
5.6 DISPlay グループ	5-23
5.7 FILE グループ	5-42
5.8 HARMonics グループ	5-45
5.9 HCOPy グループ	5-46
5.10 HOLD グループ	5-48
5.11 HSPeed グループ	5-49
5.12 IMAGe グループ	5-55
5.13 INPut グループ	5-57
5.14 INTEGrate グループ	5-67
5.15 MEASure グループ	5-70
5.16 MOTor グループ	5-75
5.17 NUMeric グループ	5-80
5.18 RATE グループ	5-92
5.19 STATus グループ	5-93
5.20 STORe グループ	5-94

目次

5.21	SYSTem グループ	5-97
5.22	WAVeform グループ	5-100
5.23	共通コマンドグループ	5-102

第 6 章 ステータスレポート

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

付録

付録 1	エラーメッセージ	付 -1
付録 2	IEEE 488.2-1992 について	付 -4

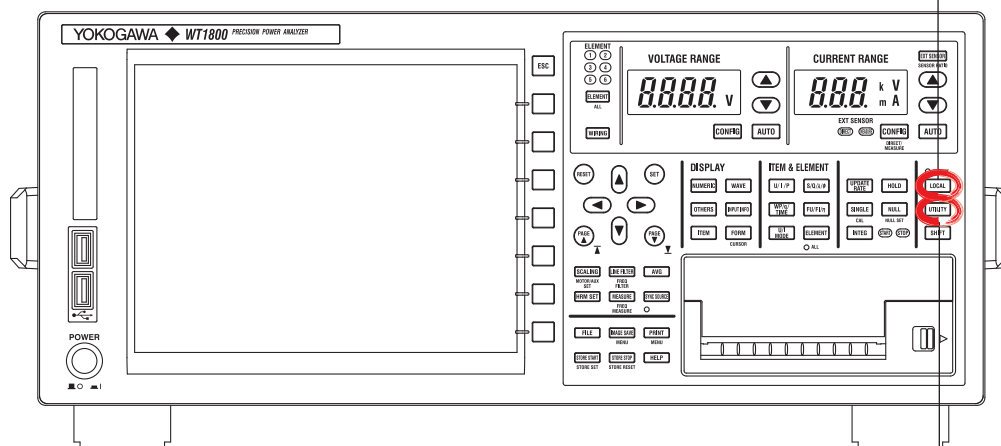
索引

1.1 各部の名称と機能

フロントパネル

LOCALキー

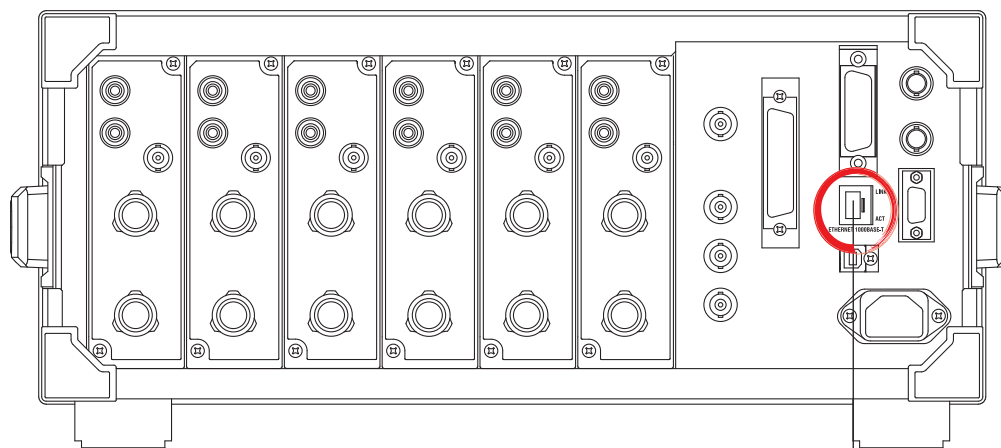
通信によるリモート状態を解除し、キー操作を可能にするローカル状態にするときに押します。ただし、コントローラによりローカルロックアウト状態(1-2ページ参照)になっているときは無効です。



UTILITYキー(1-5ページ)

ネットワーク接続のタイムアウト時間の設定をするときに押します。

リアパネル



イーサネットポート

コントローラ(PCなど)と、イーサネットケーブルで接続するためのコネクタです。接続の方法は、1-4ページをお読みください。

1.2 イーサネットインタフェースの機能と仕様

イーサネットインタフェースの機能

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。

イーサネットインタフェースの仕様

電氣的・機械的仕様 : IEEE802.3 に準拠

同時接続数 : 1

通信プロトコル : TCP/IP(VXI-11)

コネクタ形状 : RJ-45 コネクタ

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 PC から「:COMMunicate:REMOte ON」コマンドを受け取ると、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、PC から「:COMMunicate:LOCKout ON」コマンドを受信しているとき（ローカルロックアウト状態）は無効です。PC 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、他のインタフェース (GP-IB、USB インタフェース) と同時に使用できません。

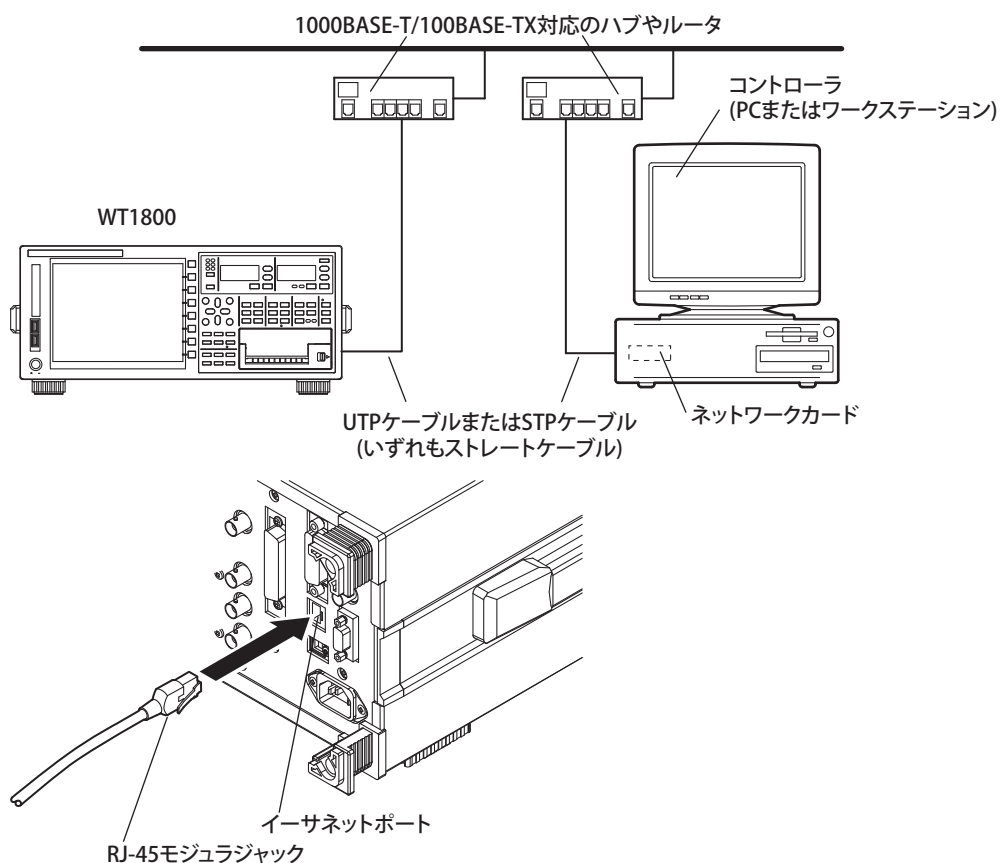
タイムアウト時間の設定

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 の範囲で設定できます。初期値は 0s です。設定の方法については、「1.4 本体の設定 (イーサネット)」をご覧ください。

1.3 イーサネットインタフェースによる接続

接続方法

ハブなどに接続された UTP(Unshielded Twisted-Pair) ケーブルまたは STP(Shielded Twisted-Pair) ケーブルを本機器のリアパネルにあるイーサネットポートに接続します。



接続時の注意

- ・ 本機器と PC との接続には、必ずハブやルータを介してストレートケーブルを使用してください。クロスケーブルでの 1 対 1 の接続では、動作を保証することができません。
- ・ ご使用のネットワーク環境 (伝送速度) に対応したケーブルを使用してください。

Note

接続方法の詳細については、WT1800 ユーザーズマニュアル [操作編] (IM WT1801-02JA) の「19.1 本機器をネットワークに接続する」をご覧ください。

1.4 本体の設定 (イーサネット)

ここでは、イーサネットインタフェースでリモートコントロールをするときの次の設定について説明しています。

- ・ ネットワーク接続のタイムアウト時間の設定

UTILITY_ Remote Control メニュー

UTILITY キー > Remote Control > Network のソフトキーを押します。次の画面が表示されます。



Note

GP-IB、USB、Network のどれか 1 つの通信インタフェースだけを使用してください。他の通信インタフェースを使って同時にコマンドを送信すると、コマンドが正常に実行されません。

TCP/IP の設定

イーサネットインタフェース機能を利用するには、TCP/IP の以下の設定が必要です。

- ・ IP アドレス
- ・ サブネットマスク
- ・ デフォルトゲートウェイ

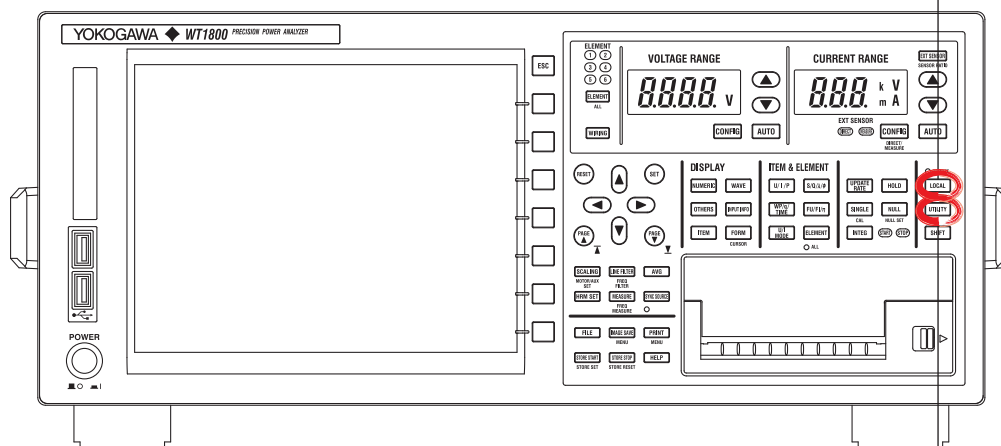
これらの設定方法の詳細については、WT1800 ユーザーズマニュアル [操作編](IM WT1801-02JA) の「20.2 TCP/IP の設定をする」をご覧ください。

2.1 各部の名称と機能

フロントパネル

LOCALキー

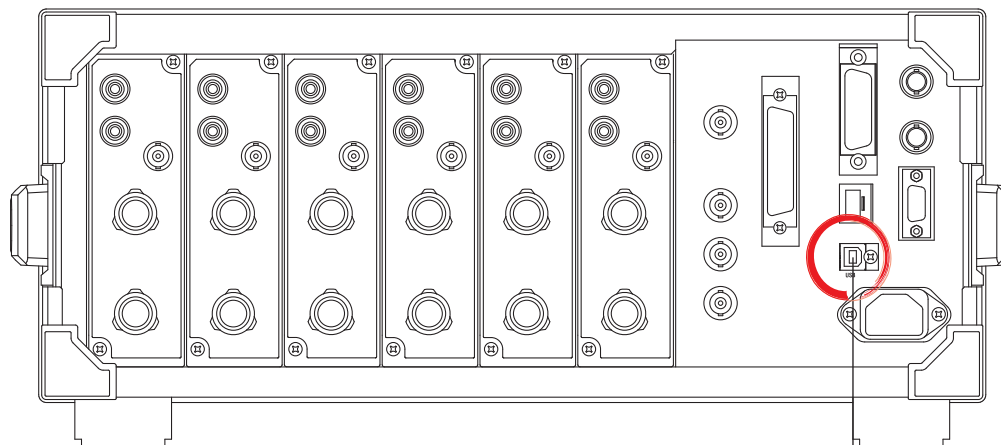
通信によるリモート状態を解除し、キー操作を可能にするローカル状態にするときに押します。ただし、コントローラによりローカルロックアウト状態(2-2ページ参照)になっているときは無効です。



UTILITYキー(2-4ページ)

USB TMCによる通信で使用する計器番号を確認するときに押します。

リアパネル



USBポート

コントローラ(PCなど)と、USBケーブルで接続するためのコネクタです。接続の方法は、2-3ページをお読みください。

2.2 USB インタフェースの機能と仕様

USB インタフェースの機能

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。

USB インタフェースの仕様

電氣的・機械的仕様	: USB Rev.2.0 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	: Windows7(32bit)、Windows Vista(32bit)、Windows XP(SP2 以降の 32bit) で動作し、USB ポートが標準装備されている機種 (PC との接続には、別途デバイスドライバが必要)

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 PC から「:COMMunicate:REMOte ON」コマンドを受け取ると、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、PC から「:COMMunicate:LOCKout ON」コマンドを受信しているとき (ローカルロックアウト状態) は無効です。PC 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、他のインタフェース (GP-IB、イーサネットインタフェース) と同時に使用できません。

2.3 USB インタフェースによる接続

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 ケーブルを抜き差ししないでください。本機器を損傷する恐れがあります。

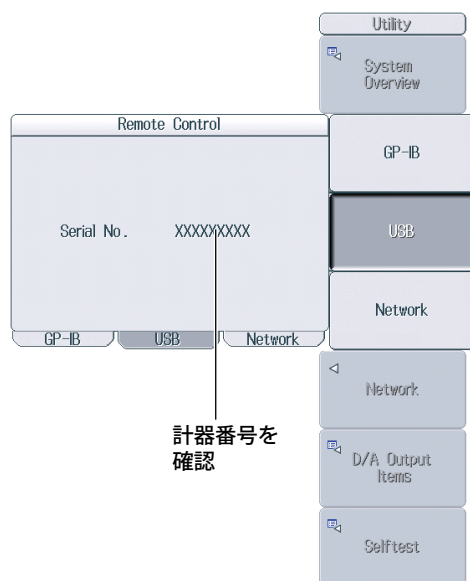
2.4 本体の設定 (USB)

ここでは、USB インタフェースでリモートコントロールをするときの次の設定について説明しています。

- USB TMC による通信で使用する計器番号の確認

UTILITY_Remote Control メニュー

UTILITY キー > Remote Control > USB のソフトキーを押します。次の画面が表示されます。



Note

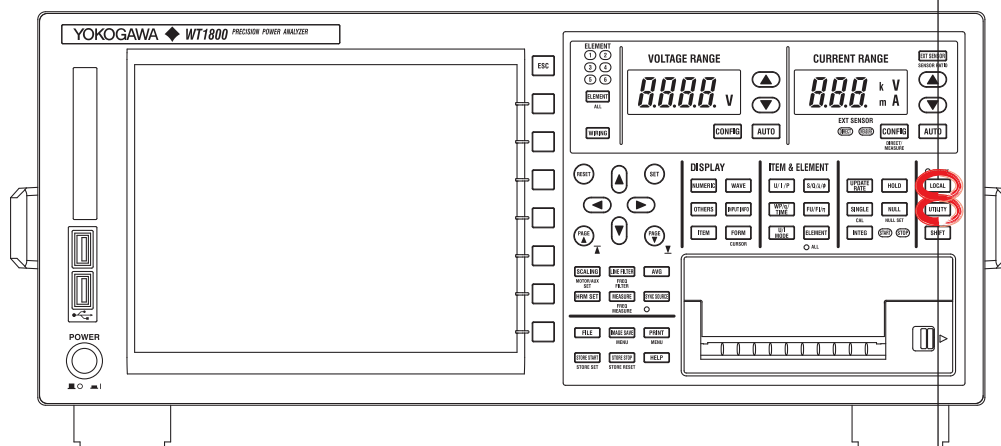
- GP-IB、USB、Network のどれか 1 つの通信インタフェースだけを使用してください。他の通信インタフェースを使って同時にコマンドを送信すると、コマンドが正常に実行されません。
- 当社の USB TMC (Test and Measurement Class) 用ドライバを PC にインストールしてください。当社の USB TMC 用ドライバの入手方法については、お買い求め先にお問い合わせいただくか、下記の当社 Web サイトから USB ドライバ提供ページにアクセスし、USB TMC 用ドライバをダウンロードしてください。
<http://www.yokogawa.com/jp-yumi/tm/F-SOFT/>
- 当社以外の USB TMC 用ドライバ (またはソフトウェア) は、使用しないでください。

3.1 各部の名称と機能

フロントパネル

LOCALキー

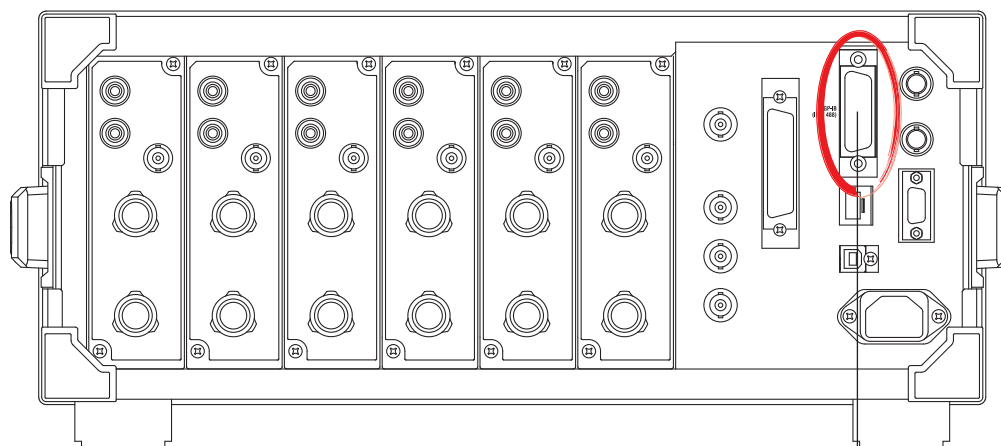
通信によるリモート状態を解除し、キー操作を可能にするローカル状態にするときに押します。ただし、コントローラによりローカルロックアウト状態(3-7ページ参照)になっているときは無効です。



UTILITYキー(3-6ページ)

GP-IBアドレスの設定をするときに押します。

リアパネル



GP-IBポート

コントローラ(PCなど)と、GP-IBケーブルで接続するためのコネクタです。接続の方法は、3-4ページをお読みください。

3.2 GP-IB インタフェースの機能と仕様

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。
- ・ 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。
- ・ 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

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。

Note

リスンオンリ、トークオンリ、およびコントローラ機能はありません。

GP-IB インタフェースの仕様

対応機器	: NATIONAL INSTRUMENTS 社
	・ PCI-GPIB および PCI-GPIB +
	・ PCIe-GPIB および PCIe-GPIB +
	・ PCMCIA-GPIB および PCMCIA-GPIB +
	・ GPIB-USB-HS
	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	: アドレスサブモード
アドレス設定	: UTILITY キー > Remote Control メニューの通信インタフェース (Device) を GP-IB にし、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
トーカー	T6	基本トーカー機能、シリアルポール、MLA(My Listen Address) によるトーカー解除機能あり、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
デバイスクリア	DC1	デバイスクリアの全機能あり
デバイストリガ	DT1	デバイストリガ機能あり
コントローラ	C0	コントローラ機能なし
電気特性	E1	オープンコレクタ

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 PC 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(3-7 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、他のインタフェース (USB、イーサネットインタフェース) と同時に使用できません。

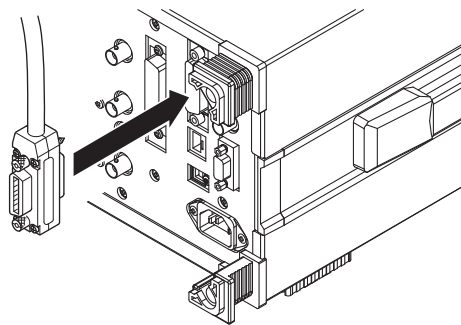
3.3 GP-IB インタフェースによる接続

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。

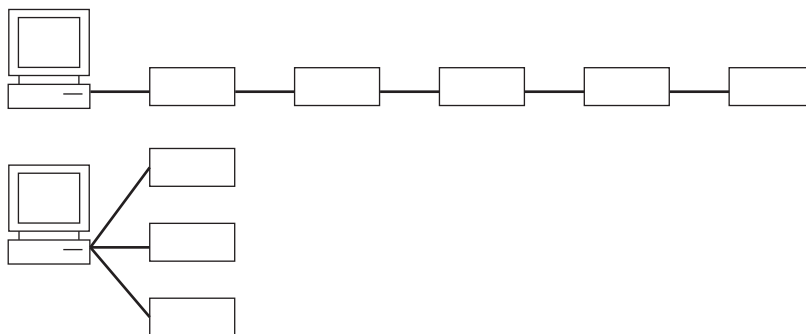
接続方法

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、3.2 節をご覧ください。
- WT1800 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、下図に示すようなリニア形またはスター形の結線にしてください。その組み合わせも可能です。ループ形の結線はできません。



注 意

通信ケーブルを接続したり、取り外したりするときは、必ず PC 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。

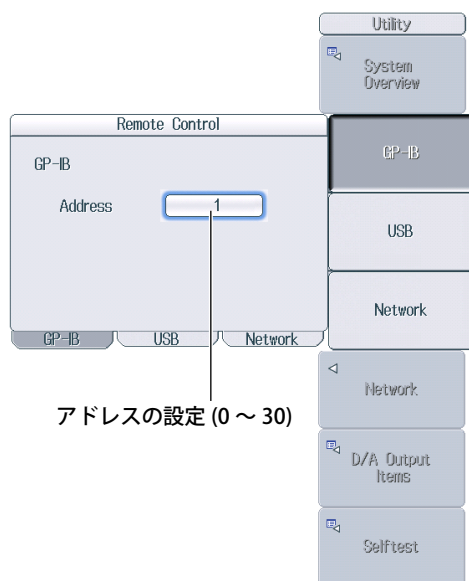
3.4 本体の設定 (GP-IB)

ここでは、GP-IB インタフェースでリモートコントロールをするときの次の設定について説明しています。

- GP-IB アドレスの設定

UTILITY_ Remote Control メニュー

UTILITY キー > Remote Control > GP-IB のソフトキーを押します。次の画面が表示されます。



Note

- GP-IB、USB、Networkのどれか1つの通信インタフェースだけを使用してください。他の通信インタフェースも同時にコマンドを送信すると、コマンドが正常に実行されません。
- GP-IB を介してコントローラが GP-IB で本機器または他のデバイスと通信しているときは、アドレスを変更しないでください。
- 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 PC 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。

3.5 インタフェースメッセージに対する応答

インタフェースメッセージに対する応答

ユニラインメッセージに対する応答

- **IFC(Interface Clear)**
トーカ、リスナを解除します。データ出力中のときは出力を中止します。
- **REN(Remote Enable)**
リモート状態 / ローカル状態を切り替えます。

IDY(Identify) はサポートしていません。

マルチラインメッセージ (アドレスコマンド) に対する応答

- **GTL(Go To Local)**
ローカル状態へ移行します。
- **SDC(Selected Device Clear)**
 - 受信中のプログラムメッセージ (コマンド) と、出力キュー (6-6 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** キーの操作を無効にし、ローカル状態への移行を禁止します。
- **DCL(Device Clear)**
SDC と同じ動作をします。
- **SPE(Serial Poll Enable)**
バス上のすべての機器のトーカ機能をシリアルポールモードにします。コントローラは各機器を順番にポーリングします。
- **SPD(Serial Poll Disable)**
バス上のすべての機器のトーカ機能のシリアルポールモードを解除します。

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

インタフェースメッセージとは

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

ユニラインメッセージ

- 1 本の管理ラインを経由してメッセージを送ります。次の 3 種類があります。
- IFC(Interface Clear)
 - REN(Remote Enable)
 - IDY(Identify)

マルチラインメッセージ

8本のデータラインを経由してメッセージを送ります。次のように分類されます。

・ アドレスコマンド

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

リスナに指定している機器に有効なコマンド

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

トーカに指定している機器に有効なコマンド

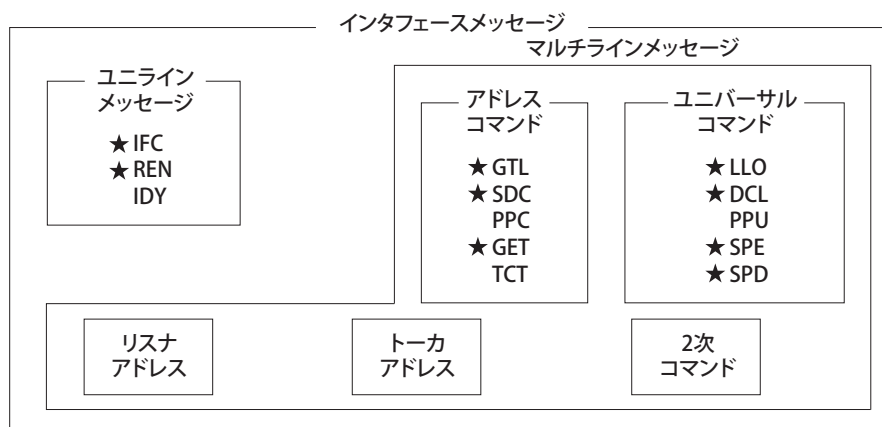
- ・ TCT(Take Control)

・ ユニバーサルコマンド

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

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

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



★印は本機器でサポートしているインタフェースメッセージです。

Note

SDC と DCL の違い

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

4.1 メッセージ

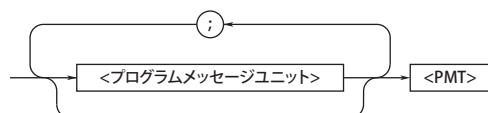
メッセージ

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

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

プログラムメッセージ

プログラムメッセージの書式は次のようになります。



＜プログラムメッセージユニット＞

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

プログラムメッセージユニットは「;」(セミコロン)で区切ります。

プログラムメッセージの書式については、次項を参照してください。

例 :INPut:CFACtor 3;INDependent OFF<PMT>
 ユニット ユニット

<PMT>

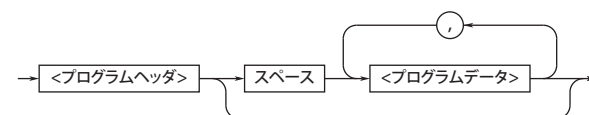
プログラムメッセージのターミネータです。次の3種類があります。

NL(ニューライン)：LF(ラインフィード)と同じ、ASCIIコード「0AH」の一文字
^END：IEEE488.1で定義されているENDメッセージ
(ENDメッセージと同時に送信されたデータバイトは、プログラムメッセージの最後のデータになります)

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

プログラムメッセージユニットの書式

プログラムメッセージユニットの書式は次のようになります。



< プログラムヘッダ >

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

< プログラムデータ >

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

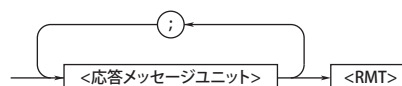
詳しくは、4-6 ページを参照してください。

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

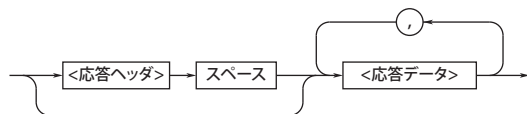
<RMT>

応答メッセージのターミネータで、NL^END です。

4.1 メッセージ

応答メッセージユニットの書式

応答メッセージユニットの書式は次のようになります。



< 応答ヘッダ >

応答データの前に応答ヘッダが付くことがあります。ヘッダとデータの間は、1文字のスペースで区切られます。詳しくは、4-5 ページを参照してください。

< 応答データ >

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

例

100.00E-03<RMT>	:	DISPLAY: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ときは、プログラムメッセージを分けるようにしてください。

メッセージの送受信時の注意

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

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

デッドロック状態

本機器は、送受信とも最低 1024 バイトのメッセージをバッファに蓄えておくことができます（バイト数は、動作状態によって増減することがあります）。このバッファが送受信と同時にいっぱいになると、本機器は動作不能状態になります。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動作不能から回復します。

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

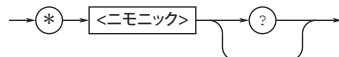
4.2 命令

命令

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

共通コマンドヘッダ

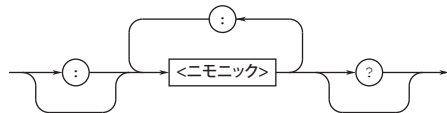
IEEE 488.2-1992 で規定されている命令を共通コマンドといいます。共通コマンドのヘッダの書式は次のようになります。先頭に必ず「*」(アスタリスク) 付けます。



共通コマンドの例 *CLS

複合ヘッダ

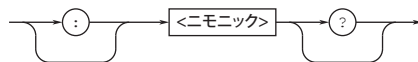
共通コマンド以外の本機器固有の命令は、機能ごとに分類されて、階層化されています。複合ヘッダの書式は次のようになります。下の階層を記述するときは、必ず「:」(コロン) を付けます。



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 (一部)

```
:INTEGrate?
:INTEGrate:MODE
:INTEGrate:TIMer
:INTEGrate:RTIME?
:INTEGrate:RTIME:START
:INTEGrate:RTIME:END
:INTEGrate:ACAL
: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 (省略することはできません)。

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 (省略することはできません)。

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>

4.2 命令

上位クエリ

グループの上位のコマンドに「?」を付けたクエリを上位クエリといいます。最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;TIMER 0,0,0;ACAL 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。

- ・二モニックのアルファベットの太文字 / 小文字は区別しません。
例 「CURSor」 -> 「cursor」「Cursor」でも可
- ・小文字の部分は省略できます。
例 「CURSor」 -> 「CURSO」「CURS」でも可
- ・ヘッダの最後の「?」(クエスチョンマーク)は、クエリであることを示します。「?」は省略できません。
例 「CURSor?» -> 最小の省略形は「CURS?」
- ・二モニックの最後に付いている <x>(数値)を省略すると、x = 1 と解釈します。
例 「ELEMeNt<x>」 -> 「ELEM」とすると
「ELEMeNt1」の意味
- ・[] で囲まれた部分は省略できます。
例 [:INPut]:SCALing[:STATe][: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。

4.3 応答

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。

4.4 データ

データ

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

データ	意味
<10 進数>	10 進数で表された数値 (例: 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:EES E #HFE)
<文字データ>	規定された文字列 (二モニック)。{} 内から選択 (例: トリガモードの選択 ->:DISPlay:WAVE:TRIGger: MODE {AUTO NORMa OFF})
<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。

<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 (ニモニック) のデータです。主に選択肢を表現するときに使用され、{} 内の文字列からどれか 1 つを選んで記述します。データの解釈のしかたは、4-4 ページの「ヘッダの解釈の規則」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、整数に四捨五入した値が「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。

<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>	#8000000010ABCDEFGHIJ

- ・ #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: 00000010 = 10 バイト)。
- ・ <データバイトの並び>
実際のデータを表します (例: ABCDEFGHIJ)。
- ・ データは 8 ビットでとり得る値 (0～255) です。したがって、「NL」を示す ASCII コード「0AH」もデータになることがありますので、コントローラ側では注意が必要です。

4.5 コントローラとの同期

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

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

```
:INPut:VOLTage:RANge:ELEMenT1 100V;  
ELEMenT?<PMT>
```

これは、「:INPut:VOLTage:RANge:ELEMenT1」が自身の処理を終えるまで、次の命令を待たせているためです。このような命令をシーケンシャルコマンドといいます。

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

```
:FILE:LOAD:SETup "FILE1";:INPut:VOLTage:  
RANge:ELEMenT1?
```

「:INPut:VOLTage:RANge:ELEMenT1?」はファイルロードが終了する前に実行されてしまい、応答される電圧レンジはファイルロードする前の値になってしまいます。

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

このようなときは、以下に示す方法でオーバーラップ動作を回避できます。

オーバーラップコマンドとの同期をとる方法

• *WAI コマンドを使う

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

```
例:COMMunicate:OPSE #H0040;:FILE:LOAD:  
SETup "FILE1";*WAI;:INPut:VOLTage:  
RANge:ELEMenT1?<PMT>
```

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

「:INPut:VOLTage:RANge: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、オーバーラップ動作を許可(または禁止)する命令です。

```
例:COMMunicate:OVERlap #HFFBF;:FILE:  
LOAD:SETup "FILE1";:INPut:VOLTage:  
RANge:ELEMenT1?<PMT>
```

「COMMunicate:OVERlap #HFFBF」は、メディアアクセス以外のオーバーラップ動作を許可しています。ファイルロードはオーバーラップ動作を許可されていないので、次の「FILE:LOAD:SETup」は、シーケンシャルコマンドと同じ動作をします。したがって、「:INPut:VOLTage:RANge:ELEMenT1?」は、ファイルロードが終了するまで実行されません。

• *OPC コマンドを使う

*OPC コマンドは、オーバーラップ動作が終了したときに、標準イベントレジスタ(6-4ページ参照)のビット0であるOPCビットを1にする命令です。

```
例:COMMunicate:OPSE #H0040;  
*ESE 1;*ESR?;*SRE 32;:FILE:LOAD:  
SETup "FILE1";*OPC<PMT>  
(*ESR? の応答を読む)  
(サービスリクエストの発生を待つ)  
:INPut:VOLTage:RANge:ELEMenT1?<PMT>
```

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

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

「*ESR?」は、標準イベントレジスタをクリアします。上の例では、「:INPut:VOLTage:RANge:ELEMenT1?」は、サービスリクエストが発生するまで実行されません。

• *OPC? クエリを使う

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

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


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

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

Note

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

オーバーラップコマンド以外の同期をとる方法

シーケンシャルコマンドの場合でも、測定データを正しく問い合わせるために同期をとる必要がある場合もあります。たとえば、測定データの更新ごとに最新の数値データを問い合わせたいとき、任意のタイミングで「:NUMeric[:NORMal]:VALue?」を送信していると、測定データの更新終了いかんにかかわらず現在の測定データを応答するため、前回と同じデータを受信してしまう可能性があります。このときは、次に示す方法で測定データの更新が終了したときのタイミングをとる必要があります。

• STATUS:CONDition? クエリを使う

「STATUS:CONDition?」は状態レジスタ (6-5 ページ参照) の内容を問い合わせる命令です。測定データの更新中かそうでないかは、状態レジスタのビット 0 を読むことで判断できます。状態レジスタのビット 0 が「1」なら測定データの更新中、「0」なら測定データの問い合わせ可能を示します。

• 拡張イベントレジスタを使う

状態レジスタの変化は、拡張イベントレジスタ (6-5 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? の応答を読む)
    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? の応答を読む)
    Loop
    COMMunicate:WAIT 1<PMT>
    :NUMeric[:NORMal]:VALue?<PMT>
    (:NUMeric[:NORMal]:VALue? の応答を読み出す)
    :STATus:EESR?<PMT>
    (:STATus:EESR? の応答を読み出す)
    (Loop に戻る)
```

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

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

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

5.1 コマンド一覧表

コマンド	機能	ページ
AOUTput グループ		
:AOUTput?	D/A 出力に関するすべての設定値を問い合わせます。	5-16
:AOUTput:NORMal?	D/A 出力に関するすべての設定値を問い合わせます。	5-16
:AOUTput[:NORMal]:CHANnel<x>	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。	5-16
:AOUTput[:NORMal]:IRTime	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。	5-16
:AOUTput[:NORMal]:MODE<x>	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。	5-16
:AOUTput[:NORMal]:RATE<x>	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。	5-16
AUX グループ		
:AUX<x>?	外部信号入力に関するすべての設定値を問い合わせます。	5-17
:AUX<x>:AUTO	外部信号入力の電圧オートレンジ ON/OFF を設定 / 問い合わせします。	5-17
:AUX<x>:FILTer?	外部信号入力の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。	5-17
:AUX<x>:FILTer[:LINE]	外部信号入力のラインフィルタを設定 / 問い合わせします。	5-17
:AUX<x>:LSCale?	外部信号入力のリニアスケールに関するすべての設定値を問い合わせます。	5-17
:AUX<x>:LSCale:AVALue	外部信号入力のリニアスケールの傾き A を設定 / 問い合わせします。	5-17
:AUX<x>:LSCale:BVALue	外部信号入力のリニアスケールのオフセット値 B を設定 / 問い合わせします。	5-17
:AUX<x>:LSCale:CALCulate?	外部信号入力のリニアスケールのパラメータ計算に関するすべての設定値を問い合わせます。	5-17
:AUX<x>:LSCale:CALCulate:{P1X P1Y P2X P2Y}	外部信号入力のリニアスケールのパラメータ計算のためのデータ {Point1X Point1Y Point2X Point2Y} を設定 / 問い合わせします。	5-17
:AUX<x>:LSCale:CALCulate:EXECute	外部信号入力のリニアスケールのパラメータ計算を実行します。	5-18
:AUX<x>:NAME	外部信号入力の名前を設定 / 問い合わせします。	5-18
:AUX<x>:RANGe	外部信号入力の電圧レンジを設定 / 問い合わせします。	5-18
:AUX<x>:SCALing	外部信号入力のスケール係数を設定 / 問い合わせします。	5-18
:AUX<x>:UNIT	外部信号入力に付加する単位を設定 / 問い合わせします。	5-18
COMMunicate グループ		
:COMMunicate?	通信に関するすべての設定値を問い合わせます。	5-19
:COMMunicate:HEADer	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。	5-19
:COMMunicate:LOCKout	ローカルロックアウトを設定 / 解除します。	5-19
:COMMunicate:OPSE	*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。	5-19
: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を問い合わせます。	5-19
:COMMunicate:OVERlap	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。	5-19
:COMMunicate:REMote	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。	5-20
:COMMunicate:VERBose	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。	5-20
:COMMunicate:WAIT	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。	5-20
:COMMunicate:WAIT?	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。	5-20
CURSor グループ		
:CURSor?	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。	5-21
:CURSor:BAR?	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。	5-21
:CURSor:BAR:LINKage	バーグラフ表示のカーソル位置のリンケージ ON/OFF を設定 / 問い合わせします。	5-21
:CURSor:BAR:POSition<x>	バーグラフ表示のカーソル位置を設定 / 問い合わせします。	5-21
:CURSor:BAR[:STATe]	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。	5-21
:CURSor:BAR:{Y<x> DY}?	バーグラフ表示のカーソル測定値を問い合わせます。	5-21
:CURSor:TREND?	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ます。	5-21

5.1 コマンド一覧表

コマンド	機能	ページ
:CURSor:TREND:LINKage	トレンド表示のカーソル位置のリンケージ ON/OFF を設定 / 問い合わせします。	5-21
:CURSor:TREND:POSition<x>	トレンド表示のカーソル位置を設定 / 問い合わせします。	5-21
:CURSor:TREND[:STATe]	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。	5-21
:CURSor:TREND:TRACe<x>	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。	5-22
:CURSor:TREND:{X<x> Y<x> DY}?	トレンド表示のカーソル測定値を問い合わせます。	5-22
:CURSor:WAVE?	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。	5-22
:CURSor:WAVE:LINKage	波形表示のカーソル位置のリンケージ ON/OFF を設定 / 問い合わせします。	5-22
:CURSor:WAVE:PATH	波形表示のカーソルパスを設定 / 問い合わせします。	5-22
:CURSor:WAVE:POSition<x>	波形表示のカーソル位置を設定 / 問い合わせします。	5-22
:CURSor:WAVE[:STATe]	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。	5-22
:CURSor:WAVE:TRACe<x>	波形表示のカーソルの対象を設定 / 問い合わせします。	5-22
:CURSor:WAVE: {X<x> DX PERDt Y<x> DY}?	波形表示のカーソル測定値を問い合わせます。	5-22

DISPlay グループ

:DISPlay?	画面表示に関するすべての設定値を問い合わせます。	5-23
:DISPlay:BAR?	バーグラフ表示に関するすべての設定値を問い合わせます。	5-23
:DISPlay:BAR:FORMat	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。	5-23
:DISPlay:BAR:ITEM<x>?	各バーグラフに関するすべての設定値を問い合わせます。	5-23
:DISPlay:BAR:ITEM<x>[:FUNCTioN]	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。	5-23
:DISPlay:BAR:ITEM<x>:SCALing?	バーグラフのスケールリングに関するすべての設定値を問い合わせます。	5-23
:DISPlay:BAR:ITEM<x>:SCALing:MODE	バーグラフのスケールリング方式を設定 / 問い合わせします。	5-23
:DISPlay:BAR:ITEM<x>:SCALing:VA Lue	バーグラフのマニュアルスケールリング上限値を設定 / 問い合わせします。	5-23
:DISPlay:BAR:ITEM<x>:SCALing:VE RTical	バーグラフの垂直スケール形式を設定 / 問い合わせします。	5-24
:DISPlay:BAR:ITEM<x>:SCALing:XA Xis	バーグラフの X 軸の位置を設定 / 問い合わせします。	5-24
:DISPlay:BAR:ORDER	バーグラフの表示開始 / 終了次数を設定 / 問い合わせします。	5-24
:DISPlay:HSPeet?	高速データ収集モードの表示に関するすべての設定値を問い合わせます。	5-24
:DISPlay:HSPeet:COLumn?	高速データ収集モードのカラムに関するすべての設定値を問い合わせます。	5-24
:DISPlay:HSPeet:COLumn:ITEM<x>	高速データ収集モードのカラム表示項目を設定 / 問い合わせします。	5-24
:DISPlay:HSPeet:COLumn:NUMber	高速データ収集モードの表示カラム数を設定 / 問い合わせします。	5-24
:DISPlay:HSPeet:COLumn:RESet	高速データ収集モードのカラム表示項目を初期値にリセットします。	5-24
:DISPlay:HSPeet:FRAMe	高速データ収集モードのデータ部のフレーム表示 ON/OFF を設定 / 問い合わせします。	5-24
:DISPlay:HSPeet:PAGE	高速データ収集モードの表示ページを設定 / 問い合わせします。	5-25
:DISPlay:HSPeet:POVer	高速データ収集モードのピークオーバー発生情報の表示 ON/OFF を設定 / 問い合わせします。	5-25
:DISPlay:INFORMatioN?	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。	5-25
:DISPlay:INFORMatioN:PAGE	設定情報の一覧表示における表示ページを設定 / 問い合わせします。	5-25
:DISPlay:INFORMatioN[:STATe]	設定情報の一覧表示の ON/OFF を設定 / 問い合わせします。	5-25
:DISPlay:MODE	表示方式を設定 / 問い合わせします。	5-25
:DISPlay:NUMeric?	数値表示に関するすべての設定値を問い合わせます。	5-25
:DISPlay:NUMeric:CUSTom?	数値表示 (カスタム表示) に関するすべての設定値を問い合わせます。	5-25
:DISPlay:NUMeric:CUSTom:FILE:CD IRectory	数値表示 (カスタム表示) のファイル読み込み / 保存先ディレクトリを変更します。	5-25
:DISPlay:NUMeric:CUSTom:FILE:DR IVE	数値表示 (カスタム表示) のファイル読み込み / 保存先ドライブを設定します。	5-26
:DISPlay:NUMeric:CUSTom:FILE:FR EE?	数値表示 (カスタム表示) のファイル読み込み / 保存先ドライブの空き容量 (byte) を問い合わせます。	5-26
:DISPlay:NUMeric:CUSTom:FILE:LO AD:ABORt	数値表示 (カスタム表示) のファイルの読み込みを中止します。	5-26
:DISPlay:NUMeric:CUSTom:FILE:LO AD:BMP	数値表示 (カスタム表示) の背景ファイルの読み込みを実行します。	5-26
:DISPlay:NUMeric:CUSTom:FILE:LO AD:BOTH	数値表示 (カスタム表示) の表示構成 & 背景ファイルの読み込みを実行します。	5-26
:DISPlay:NUMeric:CUSTom:FILE:LO AD:ITEM	数値表示 (カスタム表示) の表示構成ファイルの読み込みを実行します。	5-26
:DISPlay:NUMeric:CUSTom:FILE:PA TH?	数値表示 (カスタム表示) のファイル読み込み / 保存先を絶対パスで問い合わせます。	5-26

5.1 コマンド一覧表

コマンド	機能	ページ
:DISPlay:NUMeric:CUSTom:FILE:SAVE:ANAMing	数値表示 (カスタム表示) の表示構成を保存するファイル名の自動生成機能を設定 / 問い合わせします。	5-26
:DISPlay:NUMeric:CUSTom:FILE:SAVE:ITEM	数値表示 (カスタム表示) の表示構成ファイルの保存を実行します。	5-26
:DISPlay:NUMeric:CUSTom:ITEM<x>?	数値表示 (カスタム表示) の各表示項目に関するすべての設定値を問い合わせます。	5-26
:DISPlay:NUMeric:CUSTom:ITEM<x>:COLOr	数値表示 (カスタム表示) の表示項目の文字色を設定 / 問い合わせします。	5-27
:DISPlay:NUMeric:CUSTom:ITEM<x>[:FUNCTion]	数値表示 (カスタム表示) の表示項目 (数値項目または文字列) を設定 / 問い合わせします。	5-27
:DISPlay:NUMeric:CUSTom:ITEM<x>:POSition	数値表示 (カスタム表示) の表示項目の表示位置を設定 / 問い合わせします。	5-28
:DISPlay:NUMeric:CUSTom:ITEM<x>:SIZE	数値表示 (カスタム表示) の表示項目の文字サイズを設定 / 問い合わせします。	5-28
:DISPlay:NUMeric:CUSTom:PAGE	数値表示 (カスタム表示) の表示ページを設定 / 問い合わせします。	5-28
:DISPlay:NUMeric:CUSTom:PERPage	数値表示 (カスタム表示) の 1 ページあたりの項目数を設定 / 問い合わせします。	5-28
:DISPlay:NUMeric:CUSTom:TOTal	数値表示 (カスタム表示) の全表示項目数を設定 / 問い合わせします。	5-28
:DISPlay:NUMeric:FRAMe	数値表示のデータ部のフレーム表示 ON/OFF を設定 / 問い合わせします。	5-28
:DISPlay:NUMeric:NORMal?	数値表示に関するすべての設定値を問い合わせます。	5-28
:DISPlay:NUMeric[:NORMal]:ALL?	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。	5-28
:DISPlay:NUMeric[:NORMal]:ALL:COlumn?	数値表示 (全表示) のカラムに関するすべての設定値を問い合わせます。	5-29
:DISPlay:NUMeric[:NORMal]:ALL:COlumn:DAELem	数値表示 (全表示) のカラム全体表示機能の ON/OFF を設定 / 問い合わせします。	5-29
:DISPlay:NUMeric[:NORMal]:ALL:COlumn:SCRoll	数値表示 (全表示) のカラムのスクロールを設定 / 問い合わせします。	5-29
:DISPlay:NUMeric[:NORMal]:ALL:CURSor	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。	5-29
: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を設定 / 問い合わせします。	5-29
:DISPlay:NUMeric[:NORMal]:ALL:PAGE	数値表示 (全表示) の表示ページを設定 / 問い合わせします。	5-29
:DISPlay:NUMeric[:NORMal]:FORMat	数値表示の方式を設定 / 問い合わせします。	5-30
:DISPlay:NUMeric[:NORMal]:LIST?	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。	5-30
:DISPlay:NUMeric[:NORMal]:LIST:CURSor	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。	5-30
: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を設定 / 問い合わせします。	5-30
: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) を設定 / 問い合わせします。	5-30
: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を設定 / 問い合わせします。	5-31
:DISPlay:NUMeric[:NORMal]:MATRix?	数値表示 (マトリクス表示) に関するすべての設定値を問い合わせます。	5-31
:DISPlay:NUMeric[:NORMal]:MATRix:COlumn?	数値表示 (マトリクス表示) のカラムに関するすべての設定値を問い合わせます。	5-31
:DISPlay:NUMeric[:NORMal]:MATRix:COlumn:ITEM<x>	数値表示 (マトリクス表示) のカラム表示項目を設定 / 問い合わせします。	5-31
:DISPlay:NUMeric[:NORMal]:MATRix:COlumn:NUMber	数値表示 (マトリクス表示) のカラム数を設定 / 問い合わせします。	5-31
:DISPlay:NUMeric[:NORMal]:MATRix:COlumn:RESet	数値表示 (マトリクス表示) のカラム表示項目を初期値にリセットします。	5-31
:DISPlay:NUMeric[:NORMal]:MATRix:CURSor	数値表示 (マトリクス表示) のカーソル位置を設定 / 問い合わせします。	5-31
:DISPlay:NUMeric[:NORMal]:MATRix:ITEM<x>	数値表示 (マトリクス表示) の表示項目 (ファンクション・次数) を設定 / 問い合わせします。	5-32
:DISPlay:NUMeric[:NORMal]:MATRix:PAGE	数値表示 (マトリクス表示) の表示ページを設定 / 問い合わせします。	5-32
:DISPlay:NUMeric[:NORMal]:MATRix:PRESet	数値表示 (マトリクス表示) の表示項目を決められたパターンにプリセットします。	5-32
:DISPlay:NUMeric[:NORMal]:{VAL4 VAL8 VAL16}?	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。	5-32

5.1 コマンド一覧表

コマンド	機能	ページ
:DISPlay:NUMeric[:NORMal]: {VAL4 VAL8 VAL16}:CURSor	数値表示 (4 値 8 値 16 値) 表示) のカーソル位置を設定 / 問い合わせします。	5-32
: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・ 5-33 次数) を設定 / 問い合わせします。	
:DISPlay:NUMeric[:NORMal]: {VAL4 VAL8 VAL16}:PAGE	数値表示 (4 値 8 値 16 値) 表示) の表示ページを設定 / 問い合わせします。	5-33
:DISPlay:NUMeric[:NORMal]: {VAL4 VAL8 VAL16}:PRESet	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 5-33 セットします。	
:DISPlay:TREND?	トレンド表示に関するすべての設定値を問い合わせます。	5-33
:DISPlay:TREND:ALL	すべてのトレンドの ON/OFF を一括設定します。	5-33
:DISPlay:TREND:CLEAr	トレンドをクリアします。	5-34
:DISPlay:TREND:FORMAt	トレンドの表示フォーマットを設定 / 問い合わせします。	5-34
:DISPlay:TREND:ITEM<x>?	各トレンドに関するすべての設定値を問い合わせます。	5-34
:DISPlay:TREND:ITEM<x>[:FUNcti on]	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 5-34 ます。	
:DISPlay:TREND:ITEM<x>:SCALing?	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。	5-34
:DISPlay:TREND:ITEM<x>:SCALing: MODE	トレンドのスケーリング方式を設定 / 問い合わせします。	5-34
: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を設定 / 問い合わせします。	5-34
:DISPlay:TREND:T<x>	各トレンドの ON/OFF を設定 / 問い合わせします。	5-34
:DISPlay:TREND:TDiv	トレンドの横軸 (T/div) を設定 / 問い合わせします。	5-35
:DISPlay:VECTor?	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。	5-35
:DISPlay:VECTor:FORMAt	ベクトルの表示フォーマットを設定 / 問い合わせします。	5-35
:DISPlay:VECTor:ITEM<x>?	各ベクトルに関するすべての設定値を問い合わせます。	5-35
:DISPlay:VECTor:ITEM<x>:OBJect	ベクトル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。	5-35
:DISPlay:VECTor:ITEM<x>:{UMAG I MAG}	ベクトル表示の { 電圧 電流 } ズーム率を設定 / 問い合わせします。	5-35
:DISPlay:VECTor:NUMeric	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。	5-35
:DISPlay:WAVE?	波形表示に関するすべての設定値を問い合わせます。	5-35
:DISPlay:WAVE:ALL	すべての波形表示の ON/OFF を一括設定します。	5-35
:DISPlay:WAVE:FORMAt	波形の表示フォーマットを設定 / 問い合わせします。	5-35
:DISPlay:WAVE:GRATicule	グラティクル (目盛り) のタイプを設定 / 問い合わせします。	5-35
:DISPlay:WAVE:INTerpolate	波形の補間方式を設定 / 問い合わせします。	5-36
:DISPlay:WAVE:MAPPING?	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 5-36 せします。	
:DISPlay:WAVE:MAPPING[:MODE]	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。	5-36
:DISPlay:WAVE:MAPPING:{U<x> I <x> SPEed TORQue AUX<x>}	分割フォーマットへの { 電圧 電流 回転速度 トルク 外部信号 } 波形の割 5-36 り付けを設定 / 問い合わせします。	
:DISPlay:WAVE:POSition?	波形の垂直ポジション (中心位置のレベル) に関するすべての設定値を問い 5-36 合わせます。	
: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) を 5-36 設定 / 問い合わせします。	
:DISPlay:WAVE:POSition:{UALL IA LL}	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 5-36 ベル) を一括設定します。	
:DISPlay:WAVE:SVALue	スケール値表示の ON/OFF を設定 / 問い合わせします。	5-36
:DISPlay:WAVE:TDiv	波形の Time/div 値を設定 / 問い合わせします。	5-36
:DISPlay:WAVE:TLABel	波形ラベル名表示の ON/OFF を設定 / 問い合わせします。	5-36
:DISPlay:WAVE:TRIGger?	トリガに関するすべての設定値を問い合わせます。	5-37
:DISPlay:WAVE:TRIGger:LEVel	トリガレベルを設定 / 問い合わせします。	5-37
:DISPlay:WAVE:TRIGger:MODE	トリガモードを設定 / 問い合わせします。	5-37
:DISPlay:WAVE:TRIGger:SLOPe	トリガスロープを設定 / 問い合わせします。	5-37
:DISPlay:WAVE:TRIGger:SOURce	トリガソースを設定 / 問い合わせします。	5-37
:DISPlay:WAVE:{U<x> I<x> SPEed TORQue AUX<x>}	{ 電圧 電流 回転速度 トルク 外部信号 } 波形表示の ON/OFF を設定 / 問 5-37 い合わせします。	
:DISPlay:WAVE:VZoom?	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。	5-37
: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を設定 / 問い合わ 5-37 せします。	
:DISPlay:WAVE:VZoom:{UALL IALL}	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 5-37 します。	

5.1 コマンド一覧表

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すべての設定値を問い合わせます。	5-42
:FILE:CDIRectory	ファイル操作の対象ディレクトリを変更します。	5-42
:FILE:DELeTe:IMAGe:{BMP PNG JPEG}	画面イメージデータファイルを削除します。	5-42
:FILE:DELeTe:NUMeric:AScii	数値データファイルを削除します。	5-42
:FILE:DELeTe:SEtUp	設定情報ファイルを削除します。	5-42
:FILE:DELeTe:STORe:{DATA Header}	ストアされた数値データファイルを削除します。	5-42
:FILE:DELeTe:WAVE:AScii	波形表示データファイルを削除します。	5-42
:FILE:DRIVE	ファイル操作の対象ドライブを設定します。	5-42
:FILE:FILTer	ファイルリストのフィルタを設定 / 問い合わせします。	5-42
:FILE:FREE?	対象ドライブの空き容量 (byte) を問い合わせます。	5-42
:FILE:LOAD:ABORt	ファイルの読み込みを中止します。	5-42
:FILE:LOAD:SEtUp	設定情報ファイルの読み込みを実行します。	5-42
:FILE:PATH?	対象ディレクトリを絶対パスで問い合わせます。	5-42
:FILE:SAVE?	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。	5-43
:FILE:SAVE:ABORt	ファイルの保存を中止します。	5-43
:FILE:SAVE:ANAMing	保存ファイル名の自動生成機能を設定 / 問い合わせします。	5-43
:FILE:SAVE:COMMeNt	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。	5-43
:FILE:SAVE:NUMeric[:EXECute]	数値データをファイルに保存します。	5-43
:FILE:SAVE:NUMeric:ITEM	数値データをファイル保存するときの保存項目の選択方式を設定 / 問い合わせします。	5-43
:FILE:SAVE:NUMeric:NORMal?	数値データのファイル保存 (保存項目マニュアル選択方式) に関するすべての設定値を問い合わせます。	5-43
:FILE:SAVE:NUMeric:NORMal:ALL	数値データをファイル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。	5-43
:FILE:SAVE:NUMeric:NORMal:{ELEMeNt<x> SIGMA SIGMB SIGMC}	数値データをファイル保存するときの各エレメント 結線ユニット {ΣA ΣB ΣC} の出力 ON/OFF を設定 / 問い合わせします。	5-43
:FILE:SAVE:NUMeric:NORMal:<Function>	数値データをファイル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。	5-43
:FILE:SAVE:NUMeric:NORMal:PRESeT<x>	数値データをファイル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。	5-44
:FILE:SAVE:SEtUp[:EXECute]	設定情報をファイルに保存します。	5-44
:FILE:SAVE:WAVE[:EXECute]	波形表示データをファイルに保存します。	5-44
HARMonics グループ		
:HARMonics<x>?	高調波測定に関するすべての設定値を問い合わせます。	5-45
:HARMonics<x>:CONFIgure?	すべてのエレメントの高調波測定グループを問い合わせます。	5-45
:HARMonics<x>:CONFIgure[:ALL]	すべてのエレメントの高調波測定グループを一括設定します。	5-45
:HARMonics<x>:CONFIgure:ELEMeNt<x>	各エレメントの高調波測定グループを設定 / 問い合わせします。	5-45
:HARMonics<x>:CONFIgure:{SIGMA SIGMB SIGMC}	結線ユニット {ΣA ΣB ΣC} に属するエレメントの高調波測定グループを一括設定します。	5-45
:HARMonics<x>:ORDeR	最小 / 最大解析次数を設定 / 問い合わせします。	5-45
:HARMonics<x>:PLLSource	PLL ソースを設定 / 問い合わせします。	5-45
:HARMonics<x>:THD	THD (高調波ひずみ率) の算出式を設定 / 問い合わせします。	5-45
HCOPy グループ		
:HCOPy?	プリントに関するすべての設定値を問い合わせます。	5-46
:HCOPy:ABORt	プリントを中断します。	5-46
:HCOPy:AUTO?	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。	5-46
:HCOPy:AUTO:COUNt	オートプリントの印刷回数を設定 / 問い合わせします。	5-46
:HCOPy:AUTO:INTeRval	オートプリントの印刷インターバルを設定 / 問い合わせします。	5-46
:HCOPy:AUTO:MODe	オートプリントの動作方式を設定 / 問い合わせします。	5-46
:HCOPy:AUTO:PAStArT	オートプリント開始時に印刷するかどうかを設定 / 問い合わせします。	5-46
:HCOPy:AUTO:{STArT END}	実時間制御印刷モードの印刷 {開始 終了} 予約時刻を設定 / 問い合わせします。	5-46
:HCOPy:AUTO[:STATe]	オートプリントの ON/OFF を設定 / 問い合わせします。	5-47

5.1 コマンド一覧表

コマンド	機能	ページ
:HCOpy:AUTO:TEVent	イベント同期印刷モードのトリガとなるイベントを設定 / 問い合わせします。	5-47
:HCOpy:COMMeNt	画面下部に表示するコメントを設定 / 問い合わせします。	5-47
:HCOpy:EXECute	プリントを実行します。	5-47
:HCOpy:PRINter?	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。	5-47
:HCOpy:PRINter:FEED	内蔵プリンタの紙送りを実行します。	5-47
:HCOpy:PRINter:FORMat	内蔵プリンタで印刷する内容を設定 / 問い合わせします。	5-47

HOLD グループ

:HOLD	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。	5-48
-------	--------------------------------------	------

HSPEED グループ

:HSPEED?	高速データ収集機能に関するすべての設定値を問い合わせます。	5-49
:HSPEED:CAPTUREd?	高速データ収集の収集済み回数を問い合わせます。	5-49
:HSPEED:COUNT	データ収集回数を設定 / 問い合わせします。	5-49
:HSPEED:DISPLAY?	高速データ収集モードの表示に関するすべての設定値を問い合わせます。	5-49
:HSPEED:DISPLAY:COLumn?	高速データ収集モードのカラムに関するすべての設定値を問い合わせます。	5-49
:HSPEED:DISPLAY:COLumn:ITEM<x>	高速データ収集モードのカラム表示項目を設定 / 問い合わせします。	5-49
:HSPEED:DISPLAY:COLumn:NUMBER	高速データ収集モードの表示カラム数を設定 / 問い合わせします。	5-49
:HSPEED:DISPLAY:COLumn:RESet	高速データ収集モードのカラム表示項目を初期値にリセットします。	5-49
:HSPEED:DISPLAY:FRAME	高速データ収集モードのデータ部のフレーム表示 ON/OFF を設定 / 問い合わせします。	5-49
:HSPEED:DISPLAY:PAGE	高速データ収集モードの表示ページを設定 / 問い合わせします。	5-50
:HSPEED:DISPLAY:POVer	高速データ収集モードのピークオーバー発生情報の表示 ON/OFF を設定 / 問い合わせします。	5-50
:HSPEED:EXTSync	高速データ収集の外部信号同期の ON/OFF を設定 / 問い合わせします。	5-50
:HSPEED:FILTer?	高速データ収集のフィルタに関するすべての設定値を問い合わせます。	5-50
:HSPEED:FILTer[:HS]	高速データ収集のデジタルフィルタ (HS Filter) を設定 / 問い合わせします。	5-50
:HSPEED:FILTer:LINE?	高速データ収集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。	5-50
:HSPEED:FILTer:LINE[:ALL]	高速データ収集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。	5-50
:HSPEED:FILTer:LINE:ELEMeNt<x>	高速データ収集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。	5-50
:HSPEED:MAXCount?	データ収集回数の最大値を問い合わせます。	5-50
:HSPEED:MEASuring?	高速データ収集の電圧 / 電流モードに関するすべての設定値を問い合わせます。	5-50
:HSPEED:MEASuring[:ALL]	すべての電圧 / 電流モードを一括設定します。	5-50
:HSPEED:MEASuring:{U<x> I<x>}	{電圧 電流} モードを設定 / 問い合わせします。	5-51
:HSPEED:MEASuring:{UALL IALL}	すべての {電圧 電流} モードを一括設定します。	5-51
:HSPEED:POVer?	高速データ収集のピークオーバー発生情報を問い合わせます。	5-51
:HSPEED:RECOrd?	高速データ収集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。	5-51
:HSPEED:RECOrd:FILE?	収集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。	5-51
:HSPEED:RECOrd:FILE:ANAMing	収集した数値データの保存ファイル名の自動生成機能を設定 / 問い合わせします。	5-51
:HSPEED:RECOrd:FILE:CDIRectory	収集した数値データの保存先ディレクトリを変更します。	5-51
:HSPEED:RECOrd:FILE:CONVerT?	収集した数値データファイルの CSV 形式への変換に関するすべての設定値を問い合わせます。	5-51
:HSPEED:RECOrd:FILE:CONVerT:ABORt	収集した数値データファイルの CSV 形式への変換を中止します。	5-51
:HSPEED:RECOrd:FILE:CONVerT:AUTO	収集した数値データファイルの CSV 形式への自動変換 ON/OFF を設定 / 問い合わせします。	5-51
:HSPEED:RECOrd:FILE:CONVerT:EXECute	収集した数値データファイルの CSV 形式への変換を実行します。	5-51
:HSPEED:RECOrd:FILE:DRIVE	収集した数値データの保存先ドライブを設定します。	5-52
:HSPEED:RECOrd:FILE:FREE?	収集した数値データの保存先ドライブの空き容量 (byte) を問い合わせます。	5-52
:HSPEED:RECOrd:FILE:NAME	収集した数値データの保存ファイル名を設定 / 問い合わせします。	5-52
:HSPEED:RECOrd:FILE:PATH?	収集した数値データの保存先を絶対パスで問い合わせます。	5-52
:HSPEED:RECOrd:FILE:STATE?	収集した数値データのファイル保存の処理ステータスを問い合わせます。	5-52
:HSPEED:RECOrd:ITEM?	ファイル保存する数値データ項目に関するすべての設定値を問い合わせます。	5-52
:HSPEED:RECOrd:ITEM:AUX<x>	数値データ (外部信号入力) の保存する / しないを設定 / 問い合わせします。	5-52

5.1 コマンド一覧表

コマンド	機能	ページ
:HSPeed:RECOrd:ITEM:{I<x> IA IB IC}	数値データ(電流)の各エレメント 結線ユニット{ΣA ΣB ΣC}の保存する / しないを設定 / 問い合わせします。	5-52
:HSPeed:RECOrd:ITEM:{P<x> PA PB PC}	数値データ(有効電力)の各エレメント 結線ユニット{ΣA ΣB ΣC}の保存する / しないを設定 / 問い合わせします。	5-52
:HSPeed:RECOrd:ITEM:{SPEed TORQ ue PM}	数値データ(モータ)の{回転速度 トルク モータ出力}の保存する / しないを設定 / 問い合わせします。	5-52
:HSPeed:RECOrd:ITEM:{U<x> UA UB UC}	数値データ(電圧)の各エレメント 結線ユニット{ΣA ΣB ΣC}の保存する / しないを設定 / 問い合わせします。	5-53
:HSPeed:RECOrd:ITEM:PRESet:ALL	数値データのすべての項目の保存する / しないを一括設定します。	5-53
:HSPeed:RECOrd:ITEM:PRESet:{ELE Ment<x> SIGMA SIGMB SIGMC}	数値データの各エレメント 結線ユニット{ΣA ΣB ΣC}の保存する / しないを一括設定します。	5-53
:HSPeed:RECOrd:ITEM:PRESet:{U I P MOTor AUX}	数値データの各ファンクションの保存する / しないを一括設定します。	5-53
:HSPeed:RECOrd[:STATe]	収集した数値データをファイルに保存する / しないを設定 / 問い合わせします。	5-53
:HSPeed:STARt	データ収集を開始します。	5-53
:HSPeed:STATe?	高速データ収集のステータスを問い合わせます。	5-53
:HSPeed:STOP	データ収集を終了します。	5-53
:HSPeed:TRIGger?	高速データ収集のトリガに関するすべての設定値を問い合わせます。	5-53
:HSPeed:TRIGger:LEVel	トリガレベルを設定 / 問い合わせします。	5-53
:HSPeed:TRIGger:MODE	トリガモードを設定 / 問い合わせします。	5-53
:HSPeed:TRIGger:SLOPe	トリガスロープを設定 / 問い合わせします。	5-54
:HSPeed:TRIGger:SOURce	トリガソースを設定 / 問い合わせします。	5-54

IMAGe グループ

:IMAGe?	画面イメージデータの出力に関するすべての設定値を問い合わせます。	5-55
:IMAGe:ABORt	画面イメージデータの出力を中止します。	5-55
:IMAGe:COLor	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。	5-55
:IMAGe:COMMeNt	画面下部に表示するコメントを設定 / 問い合わせします。	5-55
:IMAGe:EXECute	画面イメージデータの出力を実行します。	5-55
:IMAGe:FORMat	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。	5-55
:IMAGe:SAVE?	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。	5-55
:IMAGe:SAVE:ANAMing	保存ファイル名の自動生成機能を設定 / 問い合わせします。	5-55
:IMAGe:SAVE:CDIRectory	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。	5-55
:IMAGe:SAVE:DRIVE	画面イメージデータの保存先ドライブを設定します。	5-55
:IMAGe:SAVE:FREE?	画面イメージデータの保存先ドライブの空き容量(byte)を問い合わせます。	5-55
:IMAGe:SAVE:NAME	保存ファイル名を設定 / 問い合わせします。	5-56
:IMAGe:SAVE:PATH?	画面イメージデータの保存先を絶対パスで問い合わせます。	5-56
:IMAGe:SEND?	画面イメージデータを問い合わせます。	5-56

INPut グループ

:INPut?	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。	5-57
[:INPut] :CFACtor	クレストファクタを設定 / 問い合わせます。	5-57
[:INPut] :CURRent?	電流測定に関するすべての設定値を問い合わせます。	5-57
[:INPut] :CURRent:AUTO?	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。	5-57
[:INPut] :CURRent:AUTO[:ALL]	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。	5-57
[:INPut] :CURRent:AUTO:ELEMeNt <x>	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。	5-57
[:INPut] :CURRent:AUTO:{SIGMA SIGMB SIGMC}	結線ユニット{ΣA ΣB ΣC}に属するエレメントの電流オートレンジ ON/OFF を一括設定します。	5-57
[:INPut] :CURRent:CONFIg?	すべてのエレメントの有効な電流レンジを問い合わせます。	5-57
[:INPut] :CURRent:CONFIg[:ALL]	すべてのエレメントの有効な電流レンジを一括設定します。	5-57
[:INPut] :CURRent:CONFIg:ELEMeNt <x>	各エレメントの有効な電流レンジを設定 / 問い合わせします。	5-58
[:INPut] :CURRent:EXTSensor?	外部電流センサレンジに関するすべての設定値を問い合わせます。	5-58
[:INPut] :CURRent:EXTSensor:CONFIg?	すべてのエレメントの有効な外部電流センサレンジを問い合わせます。	5-58
[:INPut] :CURRent:EXTSensor:CONFIg[:ALL]	すべてのエレメントの有効な外部電流センサレンジを一括設定します。	5-58
[:INPut] :CURRent:EXTSensor:CONFIg:ELEMeNt <x>	各エレメントの有効な外部電流センサレンジを設定 / 問い合わせします。	5-58

5.1 コマンド一覧表

コマンド	機能	ページ
[:INPut]:CURRent:EXTSensor:DISP lay	外部電流センサレンジの表示方式を設定 / 問い合わせします。	5-58
[:INPut]:CURRent:EXTSensor:POJump?	すべてのエレメントの電流ピークオーバー発生時のジャンプ先レンジを問い合わせます。	5-58
[:INPut]:CURRent:EXTSensor:POJump[:ALL]	すべてのエレメントの電流ピークオーバー発生時のジャンプ先レンジを一括設定します。	5-59
[:INPut]:CURRent:EXTSensor:POJump:ELEMeNt<x>	各エレメントの電流ピークオーバー発生時のジャンプ先レンジを設定 / 問い合わせします。	5-59
[:INPut]:CURRent:POJump?	すべてのエレメントの電流ピークオーバー発生時のジャンプ先レンジを問い合わせます。	5-59
[:INPut]:CURRent:POJump[:ALL]	すべてのエレメントの電流ピークオーバー発生時のジャンプ先レンジを一括設定します。	5-59
[:INPut]:CURRent:POJump:ELEMeNt<x>	各エレメントの電流ピークオーバー発生時のジャンプ先レンジを設定 / 問い合わせします。	5-59
[:INPut]:CURRent:RANGe?	すべてのエレメントの電流レンジを問い合わせます。	5-59
[:INPut]:CURRent:RANGe[:ALL]	すべてのエレメントの電流レンジを一括設定します。	5-60
[:INPut]:CURRent:RANGe:ELEMeNt<x>	各エレメントの電流レンジを設定 / 問い合わせします。	5-60
[:INPut]:CURRent:RANGe:{SIGMA SIGMB SIGMC}	結線ユニット{Σ A Σ B Σ C}に属するエレメントの電流レンジを一括設定します。	5-60
[:INPut]:CURRent:SRATio?	すべてのエレメントの外部電流センサ換算比を問い合わせます。	5-60
[:INPut]:CURRent:SRATio[:ALL]	すべてのエレメントの外部電流センサ換算比を一括設定します。	5-60
[:INPut]:CURRent:SRATio:ELEMeNt<x>	各エレメントの外部電流センサ換算比を設定 / 問い合わせします。	5-60
[:INPut]:CURRent:SRATio:{SIGMA SIGMB SIGMC}	結線ユニット{Σ A Σ B Σ C}に属するエレメントの外部電流センサ換算比を一括設定します。	5-61
[:INPut]:ESElect	測定レンジの設定対象を設定 / 問い合わせします。	5-61
[:INPut]:FILTeR?	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。	5-61
[:INPut]:FILTeR:FREQuency?	すべてのエレメントの周波数フィルタを問い合わせます。	5-61
[:INPut]:FILTeR:FREQuency[:ALL]	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。	5-61
[:INPut]:FILTeR:FREQuency:ELEMeNt<x>	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。	5-61
[:INPut]:FILTeR:LINE?	すべてのエレメントのラインフィルタを問い合わせます。	5-61
[:INPut]:FILTeR[:LINE][:ALL]	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。	5-61
[:INPut]:FILTeR[:LINE]:ELEMeNt<x>	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。	5-61
[:INPut]:FILTeR[:LINE]:{SIGMA SIGMB SIGMC}	結線ユニット{Σ A Σ B Σ C}に属するエレメントのラインフィルタを一括設定します。	5-62
[:INPut]:INDePeNdeNt	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。	5-62
[:INPut]:MODUle?	入力エレメントタイプを問い合わせます。	5-62
[:INPut]:NULL:CONDition:{SPEEd TORQue AUX<x>}	{回転速度 トルク AUX} の NULL 動作状態を問い合わせます。	5-62
[:INPut]:NULL:CONDition:{U<x> I<x>}	各エレメントの{電圧 電流} の NULL 動作状態を問い合わせます。	5-62
[:INPut]:NULL[:STATe]	NULL 機能の ON/OFF を設定 / 問い合わせします。	5-62
[:INPut]:NULL:TARGet?	NULL 機能の動作対象に関するすべての設定値を問い合わせます。	5-62
[:INPut]:NULL:TARGet[:MODE]	NULL 機能の動作対象の選択モードを設定 / 問い合わせします。	5-62
[:INPut]:NULL:TARGet:{SPEEd TORQue AUX<x>}	{回転速度 トルク AUX} の NULL 動作対象を設定 / 問い合わせします。	5-62
[:INPut]:NULL:TARGet:{U<x> I<x>}	各エレメントの{電圧 電流} の NULL 動作対象を設定 / 問い合わせします。	5-63
[:INPut]:NULL:TARGet:{UALL IALL}	すべてのエレメントの{電圧 電流} の NULL 動作対象を一括設定します。	5-63
[:INPut]:POVer?	ピークオーバー情報を問い合わせます。	5-63
[:INPut]:SCALing?	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。	5-63
[:INPut]:SCALing:STATe?	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。	5-63
[:INPut]:SCALing[:STATe][:ALL]	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。	5-63
[:INPut]:SCALing[:STATe]:ELEMeNt<x>	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。	5-63
[:INPut]:SCALing:{VT CT SFACtor}?	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。	5-63

5.1 コマンド一覧表

コマンド	機能	ページ
[:INPut] :SCALing : { VT CT SFACtor } [:ALL]	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。	5-63
[:INPut] :SCALing : { VT CT SFACtor } :ELEMEnt <x>	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。	5-63
[:INPut] :SCALing : { VT CT SFACtor } : { SIGMA SIGMB SIGMC }	結線ユニット { ΣA ΣB ΣC } に属するエレメントの {VT 比 CT 比 電力係数} を一括設定します。	5-64
[:INPut] :SYNChronize?	すべてのエレメントの同期ソースを問い合わせます。	5-64
[:INPut] :SYNChronize [:ALL]	すべてのエレメントの同期ソースを一括設定します。	5-64
[:INPut] :SYNChronize :ELEMEnt <x>	各エレメントの同期ソースを設定 / 問い合わせします。	5-64
[:INPut] :SYNChronize : { SIGMA SIGMB SIGMC }	結線ユニット { ΣA ΣB ΣC } に属するエレメントの同期ソースを一括設定します。	5-64
[:INPut] :VOLTag?	電圧測定に関するすべての設定値を問い合わせます。	5-64
[:INPut] :VOLTag :AUTO?	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。	5-64
[:INPut] :VOLTag :AUTO [:ALL]	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。	5-64
[:INPut] :VOLTag :AUTO :ELEMEnt <x>	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。	5-64
[:INPut] :VOLTag :AUTO : { SIGMA SIGMB SIGMC }	結線ユニット { ΣA ΣB ΣC } に属するエレメントの電圧オートレンジ ON/OFF を一括設定します。	5-64
[:INPut] :VOLTag :CONFIg?	すべてのエレメントの有効な電圧レンジを問い合わせます。	5-64
[:INPut] :VOLTag :CONFIg [:ALL]	すべてのエレメントの有効な電圧レンジを一括設定します。	5-65
[:INPut] :VOLTag :CONFIg :ELEMEnt <x>	各エレメントの有効な電圧レンジを設定 / 問い合わせします。	5-65
[:INPut] :VOLTag :POJump?	すべてのエレメントの電圧ピークオーバ発生時のジャンプ先レンジを問い合わせます。	5-65
[:INPut] :VOLTag :POJump [:ALL]	すべてのエレメントの電圧ピークオーバ発生時のジャンプ先レンジを一括設定します。	5-65
[:INPut] :VOLTag :POJump :ELEMEnt <x>	各エレメントの電圧ピークオーバ発生時のジャンプ先レンジを設定 / 問い合わせします。	5-65
[:INPut] :VOLTag :RANGE?	すべてのエレメントの電圧レンジを問い合わせます。	5-65
[:INPut] :VOLTag :RANGE [:ALL]	すべてのエレメントの電圧レンジを一括設定します。	5-65
[:INPut] :VOLTag :RANGE :ELEMEnt <x>	各エレメントの電圧レンジを設定 / 問い合わせします。	5-66
[:INPut] :VOLTag :RANGE : { SIGMA SIGMB SIGMC }	結線ユニット { ΣA ΣB ΣC } に属するエレメントの電圧レンジを一括設定します。	5-66
[:INPut] :WIRing	結線方式を設定 / 問い合わせします。	5-66

INTEGrate グループ

:INTEGrate?	積算に関するすべての設定値を問い合わせます。	5-67
:INTEGrate :ACAL	積算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。	5-67
:INTEGrate :INdependent	エレメント別積算の ON/OFF を設定 / 問い合わせします。	5-67
:INTEGrate :MODE	積算モードを設定 / 問い合わせします。	5-67
:INTEGrate :QMODE?	すべてのエレメントの電流積算の電流モードを問い合わせます。	5-67
:INTEGrate :QMODE [:ALL]	すべてのエレメントの電流積算の電流モードを一括設定します。	5-67
:INTEGrate :QMODE :ELEMEnt <x>	各エレメントの電流積算の電流モードを設定 / 問い合わせします。	5-67
:INTEGrate :RESet	積算値をリセットします。	5-67
:INTEGrate :RTALl : { START END }	実時間制御積算モードにおけるすべてのエレメントの積算 { 開始 終了 } 予約時刻を一括設定します。	5-67
:INTEGrate :RTIME <x>?	実時間制御積算モードにおける積算開始 / 終了予約時刻を問い合わせます。	5-68
:INTEGrate :RTIME <x> : { START END }	実時間制御積算モードにおける積算 { 開始 終了 } 予約時刻を設定 / 問い合わせします。	5-68
:INTEGrate :START	積算をスタートします。	5-68
:INTEGrate :STAT?	積算状態を問い合わせます。	5-68
:INTEGrate :STOP	積算をストップします。	5-69
:INTEGrate :TIme <x>	積算タイマ時間を設定 / 問い合わせします。	5-69
:INTEGrate :TMALl	すべてのエレメントの積算タイマ時間を一括設定します。	5-69
:INTEGrate :WPTYPE?	すべてのエレメントの極性別電力量 (WP+/WP-) の演算方式を問い合わせます。	5-69
:INTEGrate :WPTYPE [:ALL]	すべてのエレメントの極性別電力量 (WP+/WP-) の演算方式を一括設定します。	5-69

5.1 コマンド一覧表

コマンド	機能	ページ
:INTEGrate:WPTyPe:ELEMeNt<x>	各エレメントの極性別電力量 (WP+/WP-) の演算方式を設定 / 問い合わせします。	5-69

MEASure グループ

:MEASure?	演算に関するすべての設定値を問い合わせます。	5-70
:MEASure:AVERaging?	アベレージングに関するすべての設定値を問い合わせます。	5-70
:MEASure:AVERaging:COUNt	アベレージング係数を設定 / 問い合わせします。	5-70
:MEASure:AVERaging[:STATe]	アベレージングの ON/OFF を設定 / 問い合わせします。	5-70
:MEASure:AVERaging:TYPE	アベレージングのタイプを設定 / 問い合わせします。	5-70
:MEASure:DMeasure?	デルタ演算に関するすべての設定値を問い合わせます。	5-70
:MEASure:DMeasure:MODE	デルタ演算の対象とする電圧 / 電流モードを設定 / 問い合わせします。	5-70
:MEASure:DMeasure:{SIGMA SIGMB SIGMC}	結線ユニット {Σ A Σ B Σ C} に対するデルタ演算タイプを設定 / 問い合わせします。	5-70
:MEASure:EFFiciency?	効率の演算に関するすべての設定値を問い合わせます。	5-71
:MEASure:EFFiciency:ETA<x>	効率の演算式を設定 / 問い合わせします。	5-71
: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を設定 / 問い合わせします。	5-71
:MEASure:EVENT<x>?	ユーザー定義イベントに関するすべての設定値を問い合わせます。	5-71
:MEASure:EVENT<x>:EXPRession?	ユーザー定義イベントの条件式に関するすべての設定値を問い合わせます。	5-71
:MEASure:EVENT<x>:EXPRession:CO NDition	ユーザー定義イベントの条件式 (条件合成タイプ) を設定 / 問い合わせします。	5-71
:MEASure:EVENT<x>:EXPRession:IN Verse	ユーザー定義イベントの条件式 (条件合成タイプ) の論理反転 ON/OFF を設定 / 問い合わせします。	5-71
:MEASure:EVENT<x>:EXPRession:IT EM	ユーザー定義イベントの条件式 (範囲指定タイプ) の対象項目を設定 / 問い合わせします。	5-72
:MEASure:EVENT<x>:EXPRession:LI Mit<x>	ユーザー定義イベントの条件式 (範囲指定タイプ) の範囲を設定 / 問い合わせします。	5-72
:MEASure:EVENT<x>:EXPRession:ST Ring?	ユーザー定義イベントの条件式を文字列形式で問い合わせます。	5-72
:MEASure:EVENT<x>:EXPRession:TY PE	ユーザー定義イベントの条件式タイプを設定 / 問い合わせします。	5-72
:MEASure:EVENT<x>:FLABel	ユーザー定義イベントの条件不適合 (False) 時の表示データ文字列を設定 / 問い合わせします。	5-72
:MEASure:EVENT<x>:NAME	ユーザー定義イベントの名前を設定 / 問い合わせします。	5-72
:MEASure:EVENT<x>[:STATe]	ユーザー定義イベントの有効 (ON)/ 無効 (OFF) を設定 / 問い合わせします。	5-72
:MEASure:EVENT<x>:TLABel	ユーザー定義イベントの条件適合 (True) 時の表示データ文字列を設定 / 問い合わせします。	5-73
:MEASure:FREQuency?	周波数測定に関するすべての設定値を問い合わせます。	5-73
:MEASure:FREQuency:ITeM<x>	周波数測定対象を設定 / 問い合わせします。	5-73
:MEASure:FUNcTion<x>?	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。	5-73
:MEASure:FUNcTion<x>:EXPRession	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。	5-73
:MEASure:FUNcTion<x>:NAME	ユーザー定義ファンクションの名前を設定 / 問い合わせします。	5-73
:MEASure:FUNcTion<x>[:STATe]	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。	5-73
: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を設定 / 問い合わせします。	5-73
: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) を設定 / 問い合わせします。	5-73
:MEASure:PC?	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。	5-74
:MEASure:PC:IEC	Pc(Corrected Power) の演算式を設定 / 問い合わせします。	5-74
:MEASure:PC:P<x>	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。	5-74
:MEASure:PHASe	位相差の表示形式を設定 / 問い合わせします。	5-74
:MEASure:SAMPling	サンプリング周波数を設定 / 問い合わせします。	5-74
:MEASure:SFORMula	S(皮相電力) の演算式を設定 / 問い合わせします。	5-74
:MEASure:SQFormula	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。	5-74

5.1 コマンド一覧表

コマンド	機能	ページ
:MEASure:SYNChronize	同期測定モードを設定 / 問い合わせします。	5-74
MOTor グループ		
:MOTor?	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。	5-75
:MOTor:EANgle?	電気角測定に関するすべての設定値を問い合わせます。	5-75
:MOTor:EANgle:CORRection?	電気角の補正值設定に関するすべての設定値を問い合わせます。	5-75
:MOTor:EANgle:CORRection:AENTer?	電気角の補正值の自動入力に関するすべての設定値を問い合わせます。	5-75
:MOTor:EANgle:CORRection:AENTer[:EXECute]	電気角の補正值の自動入力を実行します。	5-75
:MOTor:EANgle:CORRection:AENTer:TARGet	電気角の補正值を自動入力する対象ソースを設定 / 問い合わせします。	5-75
:MOTor:EANgle:CORRection:CLEar	電気角の補正值をクリアします。	5-75
:MOTor:EANgle:CORRection[:VALue]	電気角の補正值を設定 / 問い合わせします。	5-75
:MOTor:EANgle[:STATe]	電気角測定の ON/OFF を設定 / 問い合わせします。	5-75
:MOTor:FILTer?	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。	5-75
:MOTor:FILTer[:LINE]	ラインフィルタを設定 / 問い合わせします。	5-75
:MOTor:PM?	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。	5-75
:MOTor:PM:SCALing	モータ出力演算のスケール係数を設定 / 問い合わせします。	5-76
:MOTor:PM:UNIT	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。	5-76
:MOTor:POLE	モータの極数を設定 / 問い合わせします。	5-76
:MOTor:SPEed?	回転速度 (Speed) に関するすべての設定値を問い合わせます。	5-76
:MOTor:SPEed:AUTO	回転信号 (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。	5-76
:MOTor:SPEed:LSCale?	回転信号 (アナログ入力方式) のリニアスケールに関するすべての設定値を問い合わせます。	5-76
:MOTor:SPEed:LSCale:AVALue	回転信号 (アナログ入力方式) のリニアスケールの傾き A を設定 / 問い合わせします。	5-76
:MOTor:SPEed:LSCale:BVALue	回転信号 (アナログ入力方式) のリニアスケールのオフセット値 B を設定 / 問い合わせします。	5-76
:MOTor:SPEed:LSCale:CALCulate?	回転信号 (アナログ入力方式) のリニアスケールのパラメータ計算に関するすべての設定値を問い合わせます。	5-76
:MOTor:SPEed:LSCale:CALCulate:{P1X P1Y P2X P2Y}	回転信号 (アナログ入力方式) のリニアスケールのパラメータ計算のためのデータ {Point1X Point1Y Point2X Point2Y} を設定 / 問い合わせします。	5-76
:MOTor:SPEed:LSCale:CALCulate:EXECute	回転信号 (アナログ入力方式) のリニアスケールのパラメータ計算を実行します。	5-76
:MOTor:SPEed:PRANge	回転信号 (パルス入力方式) のレンジを設定 / 問い合わせします。	5-77
:MOTor:SPEed:PULSe	回転信号 (パルス入力方式) のパルス数を設定 / 問い合わせします。	5-77
:MOTor:SPEed:RANGe	回転信号 (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。	5-77
:MOTor:SPEed:SCALing	回転速度演算のスケール係数を設定 / 問い合わせします。	5-77
:MOTor:SPEed:TYPE	回転信号の入力タイプを設定 / 問い合わせします。	5-77
:MOTor:SPEed:UNIT	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。	5-77
:MOTor:SSPeed	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。	5-77
:MOTor:SYNChronize	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。	5-77
:MOTor:TORQue?	トルク (Torque) に関するすべての設定値を問い合わせます。	5-77
:MOTor:TORQue:AUTO	トルク信号 (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。	5-78
:MOTor:TORQue:LSCale?	トルク信号 (アナログ入力方式) のリニアスケールに関するすべての設定値を問い合わせます。	5-78
:MOTor:TORQue:LSCale:AVALue	トルク信号 (アナログ入力方式) のリニアスケールの傾き A を設定 / 問い合わせします。	5-78
:MOTor:TORQue:LSCale:BVALue	トルク信号 (アナログ入力方式) のリニアスケールのオフセット値 B を設定 / 問い合わせします。	5-78
:MOTor:TORQue:LSCale:CALCulate?	トルク信号 (アナログ入力方式) のリニアスケールのパラメータ計算に関するすべての設定値を問い合わせます。	5-78
:MOTor:TORQue:LSCale:CALCulate:{P1X P1Y P2X P2Y}	トルク信号 (アナログ入力方式) のリニアスケールのパラメータ計算のためのデータ {Point1X Point1Y Point2X Point2Y} を設定 / 問い合わせします。	5-78
:MOTor:TORQue:LSCale:CALCulate:EXECute	トルク信号 (アナログ入力方式) のリニアスケールのパラメータ計算を実行します。	5-78
:MOTor:TORQue:PRANge	トルク信号 (パルス入力方式) のレンジを設定 / 問い合わせします。	5-78

5.1 コマンド一覧表

コマンド	機能	ページ
:MOTor:TORQue:RANGe	トルク信号 (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。	5-78
:MOTor:TORQue:RATE?	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。	5-79
:MOTor:TORQue:RATE:{UPPer LOWer}	トルク信号 (パルス入力方式) の { 上限 下限 } 定格値を設定 / 問い合わせします。	5-79
:MOTor:TORQue:SCALIng	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。	5-79
:MOTor:TORQue:TYPE	トルク信号の入力タイプを設定 / 問い合わせします。	5-79
:MOTor:TORQue:UNIT	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。	5-79

NUMeric グループ

:NUMeric?	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。	5-80
:NUMeric:FORMat	数値データのフォーマットを設定 / 問い合わせします。	5-80
:NUMeric:HSPEed?	高速データ収集モード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。	5-80
:NUMeric:HSPEed:CLEar	高速データ収集モードの数値データの出力項目をクリア (「NONE」に設定) します。	5-80
:NUMeric:HSPEed:DELeTe	高速データ収集モードの数値データの出力項目を削除します。	5-80
:NUMeric:HSPEed:HEADer?	高速データ収集モードの数値データのヘッダを問い合わせます。	5-81
:NUMeric:HSPEed:ITEM<x>	高速データ収集モードの数値データの出力項目 (ファンクション・エレメント) を設定 / 問い合わせします。	5-81
:NUMeric:HSPEed:{MAXimum MINimum}?	高速データ収集モードの数値データの { 最大値 最小値 } を問い合わせます。	5-81
:NUMeric:HSPEed:NUMBER	「:NUMeric:HSPEed:VALue?」で送信される数値データの項目数を設定 / 問い合わせします。	5-81
:NUMeric:HSPEed:PRESet	高速データ収集モードの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。	5-82
:NUMeric:HSPEed:VALue?	高速データ収集モードの数値データを問い合わせます。	5-82
:NUMeric:HOLD	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。	5-83
:NUMeric:LIST?	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。	5-83
:NUMeric:LIST:CLEar	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。	5-83
:NUMeric:LIST:DELeTe	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。	5-83
: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) を設定 / 問い合わせします。	5-84
:NUMeric:LIST:NUMBER	「: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。	5-84
:NUMeric:LIST:ORDER	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。	5-84
:NUMeric:LIST:PRESet	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。	5-84
:NUMeric:LIST:SELeCt	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。	5-84
:NUMeric:LIST:VALue?	高調波測定の数値リストデータを問い合わせます。	5-85
:NUMeric:NORMal?	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。	5-85
:NUMeric[:NORMal]:CLEar	数値データの出力項目をクリア (「NONE」に設定) します。	5-85
:NUMeric[:NORMal]:DELeTe	数値データの出力項目を削除します。	5-85
: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) を設定 / 問い合わせします。	5-86
:NUMeric[:NORMal]:NUMBER	「: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。	5-86
:NUMeric[:NORMal]:PRESet	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。	5-86
:NUMeric[:NORMal]:VALue?	数値データを問い合わせます。	5-86

RATE グループ

:RATE	データ更新レートを設定 / 問い合わせします。	5-92
-------	-------------------------	------

5.1 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すべての設定値を問い合わせます。	5-93
:STATus:CONDition?	状態レジスタの内容を問い合わせます。	5-93
:STATus:EESE	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。	5-93
:STATus:EESR?	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。	5-93
: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) を問い合わせます。	5-93
:STATus:FILTer<x>	遷移フィルタを設定 / 問い合わせします。	5-93
: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) を設定 / 問い合わせします。	5-93
: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) を設定 / 問い合わせします。	5-93
:STATus:SPOLL?	シリアルポールを実行します。	5-93
STORe グループ		
:STORe?	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。	5-94
:STORe:COUNT	ストア回数を設定 / 問い合わせします。	5-94
:STORe:FILE?	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。	5-94
:STORe:FILE:ANAMing	ストアされた数値データの保存ファイル名の自動生成機能を設定 / 問い合わせします。	5-94
:STORe:FILE:CDIRectory	ストアされた数値データの保存先ディレクトリを変更します。	5-94
:STORe:FILE:CONVert?	ストアされた数値データファイルの CSV 形式への変換に関するすべての設定値を問い合わせます。	5-94
:STORe:FILE:CONVert:ABORt	ストアされた数値データファイルの CSV 形式への変換を中止します。	5-94
:STORe:FILE:CONVert:AUTO	ストアされた数値データファイルの CSV 形式への自動変換 ON/OFF を設定 / 問い合わせします。	5-94
:STORe:FILE:CONVert:EXECute	ストアされた数値データファイルの CSV 形式への変換を実行します。	5-94
:STORe:FILE:DRIVe	ストアされた数値データの保存先ドライブを設定します。	5-94
:STORe:FILE:FREE?	ストアされた数値データの保存先ドライブの空き容量 (byte) を問い合わせます。	5-94
:STORe:FILE:NAME	ストアされた数値データの保存ファイル名を設定 / 問い合わせします。	5-95
:STORe:FILE:PATH?	ストアされた数値データの保存先を絶対パスで問い合わせます。	5-95
:STORe:INTerval	ストアインタバルを設定 / 問い合わせします。	5-95
:STORe:NUMeric?	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。	5-95
:STORe:NUMeric:ITEM	数値データのストア項目の選択方式を設定 / 問い合わせします。	5-95
:STORe:NUMeric:NORMal?	数値データのストア項目 (マニュアル選択方式) に関するすべての設定値を問い合わせます。	5-95
: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 を一括設定します。	5-95
:STORe:NUMeric[:NORMal]:{ELEMeNt<x> SIGMA SIGMB SIGMC}	数値データをストアするときの各エレメント 結線ユニット {ΣA ΣB ΣC} の出力 ON/OFF を設定 / 問い合わせします。	5-95
: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 を設定 / 問い合わせします。	5-95
: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します。	5-96
:STORe:RESet	数値データのストアを初期化します。	5-96
:STORe:RTIME?	実時間モードのストア開始 / 終了予約時刻を問い合わせます。	5-96
:STORe:RTIME:{START END}	実時間モードのストア {開始 終了} 予約時刻を設定 / 問い合わせします。	5-96
:STORe:SAStart	ストア開始時の数値データをストアするかどうかを設定 / 問い合わせします。	5-96
:STORe:SMODE	ストアモードを設定 / 問い合わせします。	5-96
:STORe:START	数値データのストアを開始します。	5-96
:STORe:STATe?	ストア状態を問い合わせます。	5-96
:STORe:STOP	数値データのストアを終了します。	5-96
:STORe:TEVent	イベント同期ストアモードのトリガとなるイベントを設定 / 問い合わせします。	5-96

5.1 コマンド一覧表

コマンド	機能	ページ
SYSTem グループ		
:SYSTem?	システムに関するすべての設定値を問い合わせます。	5-97
:SYSTem:CLOCK?	日付 / 時刻に関するすべての設定値を問い合わせます。	5-97
:SYSTem:CLOCK:DISPlay	日付 / 時刻表示の ON/OFF を設定 / 問い合わせします。	5-97
:SYSTem:CLOCK:SNTP?	SNTP による日付 / 時刻の設定に関するすべての設定値を問い合わせます。	5-97
:SYSTem:CLOCK:SNTP[:EXECute]	SNTP による日付 / 時刻の設定を実行します。	5-97
:SYSTem:CLOCK:SNTP:GMTTime	グリニッジ標準時との時差を設定 / 問い合わせします。	5-97
:SYSTem:CLOCK:TYPE	日付 / 時刻の設定方法を設定 / 問い合わせします。	5-97
:SYSTem:DATE	日付を設定 / 問い合わせします。	5-97
:SYSTem:DFLow:FREQuency	低周波数入力 (無入力) 時の周波数データの表示方法を設定 / 問い合わせします。	5-97
:SYSTem:DFLow:MOTor	パルス無入力時のモータデータの表示方法を設定 / 問い合わせします。	5-97
:SYSTem:DPOint	各種データをアスキー (CSV) 形式で保存する場合の小数点の種類を設定 / 問い合わせします。	5-97
:SYSTem:EClear	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。	5-97
:SYSTem:FONT	メニューおよびメッセージのフォントの大きさを設定 / 問い合わせします。	5-98
:SYSTem:KLOCK	キーロックの ON/OFF を設定 / 問い合わせします。	5-98
:SYSTem:LANGUage?	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。	5-98
:SYSTem:LANGUage:MENU	メニューの言語を設定 / 問い合わせします。	5-98
:SYSTem:LANGUage:MESSage	メッセージの言語を設定 / 問い合わせします。	5-98
:SYSTem:LCD?	液晶画面に関するすべての設定値を問い合わせます。	5-98
:SYSTem:LCD:AOff?	バックライトのオートオフ機能に関するすべての設定値を問い合わせます。	5-98
:SYSTem:LCD:AOff[:STATe]	バックライトのオートオフ機能の ON/OFF を設定 / 問い合わせします。	5-98
:SYSTem:LCD:AOff:TIME	バックライトのオートオフまでの時間を設定 / 問い合わせします。	5-98
:SYSTem:LCD:BRIGhtness	液晶画面の輝度を設定 / 問い合わせします。	5-98
:SYSTem:LCD:COLor?	液晶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。	5-98
:SYSTem:LCD:COLor:BASecolor	画面 (メニュー) の基本色を設定 / 問い合わせします。	5-98
:SYSTem:LCD:COLor:GRAPh?	波形の表示色に関するすべての設定値を問い合わせます。	5-98
:SYSTem:LCD:COLor:GRAPh:CHANnel <x>	各波形の表示色を設定 / 問い合わせします。	5-99
:SYSTem:LCD:COLor:GRAPh:PRESet	波形の表示色を決められたパターンにプリセットします。	5-99
:SYSTem:LCD:COLor:INTENSity:GRID	グリッドの輝度を設定 / 問い合わせします。	5-99
:SYSTem:LCD[:STATe]	バックライトの ON/OFF を設定 / 問い合わせします。	5-99
:SYSTem:MODel?	モデルコード (形名) を問い合わせます。	5-99
:SYSTem:RESolution	数値データの表示分解能を設定 / 問い合わせします。	5-99
:SYSTem:SERial?	シリアル番号を問い合わせます。	5-99
:SYSTem:SUFFix?	サフィックス (仕様コード) を問い合わせます。	5-99
:SYSTem:TIME	時刻を設定 / 問い合わせします。	5-99
:SYSTem:USBKeyboard	USB キーボードの種類を設定 / 問い合わせします。	5-99

WAVeform グループ

:WAVeform?	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。	5-100
:WAVeform:BYTeorder	「:WAVeform:SEND?」で送信される波形表示データ (FLOAT 形式) のバイト出力順序を設定 / 問い合わせします。	5-100
:WAVeform:END	「: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。	5-100
:WAVeform:FORMat	「: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。	5-100
:WAVeform:HOLD	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。	5-100
:WAVeform:LENGth?	「:WAVeform:TRACe」で指定した波形の全データ点数を問い合わせます。	5-100
:WAVeform:SEND?	「:WAVeform:TRACe」で指定した波形表示データを問い合わせます。	5-101
:WAVeform:SRATe?	取り込んだ波形のサンプルレートを問い合わせます。	5-101
:WAVeform:STARt	「: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。	5-101
:WAVeform:TRACe	「:WAVeform:SEND?」の対象となる波形を設定 / 問い合わせします。	5-101
:WAVeform:TRIGger?	取り込んだ波形のトリガポジションを問い合わせます。	5-101

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) を実行し、結果を問い合わせます。	5-102
*CLS	標準イベントレジスタ、拡張イベントレジスタ、エラーキューをクリアします。	5-102
*ESE	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。	5-102
*ESR?	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。	5-102
*IDN?	機種を問い合わせます。	5-102
*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 にセットします。	5-102
*OPC?	指定したオーバーラップコマンドが終了していれば、ASCII コードの「1」を返します。	5-103
*OPT?	装備しているオプションを問い合わせます。	5-103
*RST	設定の初期化を行います。	5-103
*SRE	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。	5-103
*STB?	ステータスバイトレジスタの値を問い合わせます。	5-103
*TRG	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。	5-103
*TST?	セルフテストを実行し、結果を問い合わせます。	5-104
*WAI	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。	5-104

5.2 AOUPut グループ

AOUPut 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 UTILITY キーの「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。

:AOUPut?

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 :AOUPut?

:AOUPut: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 :AOUPut:NORMAL?

:AOUPut[: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 :AOUPut[:NORMAL]:CHANNEL<x> {NONE|<Function>[,<Element>][,<Order>]}
:AOUPut[:NORMAL]:CHANNEL<x>?
<x> = 1 ~ 20 (出力チャネル)
NONE = 出力項目なし
<Function> = {URMS|IRMS|P|S|Q|...}
<Element> = {<Nrf>|SIGMA|SIGMA|SIGMC}
(<Nrf> = 1 ~ 6)
<Order> = {TOTAL|DC|<Nrf>}
(<Nrf> = 1 ~ 500)

例 :AOUTPUT:NORMAL:CHANNEL1 URMS,1
:AOUTPUT:NORMAL:CHANNEL1? ->
:AOUTPUT:NORMAL:CHANNEL1 URMS,1
:AOUTPUT:NORMAL:CHANNEL1 UK,1,1
:AOUTPUT:NORMAL:CHANNEL1? ->
:AOUTPUT:NORMAL:CHANNEL1 UK,1,1

解説

- ・ <Function> の選択肢については、「DISPLAY グループ」のファンクション選択肢一覧 (1)、5-38 ページを参照してください。
- ・ <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。

:AOUPut[: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 :AOUPut[:NORMAL]:IRTime
{<Nrf>,<Nrf>,<Nrf>}
:AOUPut[: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

:AOUPut[: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 :AOUPut[:NORMAL]:
MODE<x>{FIXED|MANUAL}
:AOUPut[:NORMAL]:MODE<x>?
<x> = 1 ~ 20 (出力チャネル)
例 :AOUTPUT:NORMAL:MODE1 FIXED
:AOUTPUT:NORMAL:MODE1? ->
:AOUTPUT:NORMAL:MODE1 FIXED

:AOUPut[: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 :AOUPut[:NORMAL]:RATE<x>{<Nrf>,<Nrf>}
:AOUPut[:NORMAL]:RATE<x>?
<x> = 1 ~ 20 (出力チャネル)
<Nrf> = -9.999E+12 ~ 9.999E+12
例 :AOUTPUT:NORMAL:RATE1 100,-100
:AOUTPUT:NORMAL:RATE1? ->
:AOUTPUT:NORMAL:
RATE1 100.0E+00,-100.0E+00

解説

- ・ 上限値、下限値の順で設定します。
- ・ D/A 出力の定格値設定方式 (:AOUPut[:NORMAL]:MODE<x>) が「MANUAL」のときに有効な設定です。

5.3 AUX グループ

AUX グループは、外部信号入力機能に関するグループです。

フロントパネルの MOTOR/AUX SET(SHIFT+SCALING) 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。ただし、このグループのコマンドは、外部信号入力機能 (オプション、/AUX) 搭載時のみ有効です。

:AUX<x>?

機能 外部信号入力に関するすべての設定値を問い合わせます。

構文 :AUX<x>?
<x> = 1 ~ 2 (AUX 入力チャネル)

:AUX<x>:AUTO

機能 外部信号入力の電圧オートレンジ ON/OFF を設定 / 問い合わせします。

構文 :AUX<x>:AUTO {<Boolean>}
:AUX<x>:AUTO?

<x> = 1 ~ 2 (AUX 入力チャネル)

例 :AUX1:AUTO ON
:AUX1:AUTO? -> :AUX1:AUTO 1

:AUX<x>:FILTer?

機能 外部信号入力の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 :AUX<x>:FILTer?
解説 AUX<x> の <x> は、設定 / 問い合わせに関係ありません。

:AUX<x>:FILTer[:LINE]

機能 外部信号入力のラインフィルタを設定 / 問い合わせします。

構文 :AUX<x>:FILTer[:LINE] {OFF|<周波数>}
:AUX<x>:FILTer:LINE?

OFF = ラインフィルタ OFF

<周波数> = 100Hz、1kHz (ラインフィルタ ON、カットオフ周波数)

例 :AUX:FILTER:LINE OFF
:AUX:FILTER:LINE? ->
:AUX1:FILTER:LINE OFF

解説 AUX<x> の <x> は、設定 / 問い合わせに関係ありません。

:AUX<x>:LSCale?

機能 外部信号入力のリニア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 :AUX<x>:LSCale?
<x> = 1 ~ 2 (AUX 入力チャネル)

:AUX<x>:LSCale:AVALue

機能 外部信号入力のリニアスケールの傾き A を設定 / 問い合わせします。

構文 :AUX<x>:LSCale:AVALue {<NRf>}
:AUX<x>:LSCale:AVALue?
<x> = 1 ~ 2 (AUX 入力チャネル)
<NRf> = 1.000E-03 ~ 1.000E+06

例 :AUX1:LSCALE:AVALUE 1.000
:AUX1:LSCALE:AVALUE? ->
:AUX1:LSCALE:AVALUE 1.000E+00

:AUX<x>:LSCale:BVALue

機能 外部信号入力のリニアスケールのオフセット値 B を設定 / 問い合わせします。

構文 :AUX<x>:LSCale:BVALue {<NRf>}
:AUX<x>:LSCale:BVALue?
<x> = 1 ~ 2 (AUX 入力チャネル)
<NRf> = -1.000E+06 ~ 1.000E+06

例 :AUX1:LSCALE:BVALUE 0
:AUX1:LSCALE:BVALUE? ->
:AUX1:LSCALE:BVALUE 0.000E+00

:AUX<x>:LSCale:CALCulate?

機能 外部信号入力のリニアスケールのパラメータ計算に関するすべての設定値を問い合わせます。

構文 :AUX<x>:LSCale:CALCulate?
<x> = 1 ~ 2 (AUX 入力チャネル)

:AUX<x>:LSCale:CALCulate:

{P1X|P1Y|P2X|P2Y}

機能 外部信号入力のリニアスケールのパラメータ計算のためのデータ {Point1X|Point1Y|Point2X|Point2Y} を設定 / 問い合わせします。

構文 :AUX<x>:LSCale:CALCulate:
{P1X|P1Y|P2X|P2Y} {<NRf>}
:AUX<x>:LSCale:CALCulate:
{P1X|P1Y|P2X|P2Y}?

<x> = 1 ~ 2 (AUX 入力チャネル)
<NRf> = -1.000E+12 ~ 1.000E+12

例 :AUX1:LSCALE:CALCULATE:P1X 0
:AUX1:LSCALE:CALCULATE:P1X? ->
:AUX1:LSCALE:CALCULATE:P1X 0.000E+00

5.3 AUX グループ

:AUX<x>:LSCale:CALCulate:EXECute

機能	外部信号入力のリニアスケールのパラメータ計算を実行します。
構文	:AUX<x>:LSCale:CALCulate:EXECute <x> = 1 ~ 2 (AUX 入力チャネル)
例	:AUX1:LSCALE:CALCULATE:EXECUTE
解説	あらかじめ設定されたデータ (Point1X,Point1Y,Point2X,Point2Y) から、リニアスケールの傾き A およびオフセット値 B を計算し、設定します。

:AUX<x>:NAME

機能	外部信号入力の名前を設定 / 問い合わせします。
構文	:AUX<x>:NAME {< 文字列 >} :AUX<x>:NAME? <x> = 1 ~ 2 (AUX 入力チャネル) < 文字列 > = 8 文字以内
例	:AUX1:NAME "AUX1" :AUX1:NAME? -> :AUX1:NAME "AUX1"

:AUX<x>:RANGe

機能	外部信号入力の電圧レンジを設定 / 問い合わせします。
構文	:AUX<x>:RANGe {< 電圧 >} :AUX<x>:RANGe? <x> = 1 ~ 2 (AUX 入力チャネル) < 電圧 > = 50、100、200、500 (mV)、1、2、5、10、20 (V)
例	:AUX1:RANGE 20V :AUX1:RANGE? -> :AUX1:RANGE 20.00E+00

:AUX<x>:SCALing

機能	外部信号入力のスケーリング係数を設定 / 問い合わせします。
構文	:AUX<x>:SCALing {<NRf>} :AUX<x>:SCALing? <x> = 1 ~ 2 (AUX 入力チャネル) <NRf> = 0.0001 ~ 99999.9999
例	:AUX1:SCALING 1 :AUX1:SCALING? -> :AUX1:SCALING 1.0000

:AUX<x>:UNIT

機能	外部信号入力に付加する単位を設定 / 問い合わせします。
構文	:AUX<x>:UNIT {< 文字列 >} :AUX<x>:UNIT? <x> = 1 ~ 2 (AUX 入力チャネル) < 文字列 > = 8 文字以内
例	:AUX1:UNIT "kW/m2" :AUX1:UNIT? -> :AUX1:UNIT "kW/m2"
解説	演算結果に影響を及ぼすことはありません。

5.4 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?

: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

:COMMunicate:OPSE

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:OPSR? コマンドの図を参照
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?

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 / オーバ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

: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:OPSR? コマンドの図を参照
例 :COMMUNICATE:OVERLAP 65535
:COMMUNICATE:OVERLAP? ->
:COMMUNICATE:OVERLAP 96

解説

- 上の例では、全ビットを 1 にして、すべてのオーバーラップコマンドを対象にしています。ただし、0 固定のビットは 1 にならないので、問い合わせに対してはビット 5、6 だけが 1 になっています。
- COMMunicate:OVERlap を使った同期のとり方については、4-8 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:OPSR? コマンドの図を参照)。

5.4 COMMunicate グループ

: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

: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、6-5 ページ参照)

例 :COMMUNICATE:WAIT 1

解説 COMMunicate:WAIT を使った同期のとり方については、4-9 ページを参照してください。

:COMMunicate:WAIT?

機能 指定された拡張イベントのどれかが発生したときに応答を作成します。

構文 :COMMunicate:WAIT? <Register>
<Register> = 0 ~ 65535 (拡張イベントレジスタ、6-5 ページ参照)

例 :COMMUNICATE:WAIT? 65535 -> 1

5.5 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+FORM) キーと同じ設定、および設定内容・測定値を問い合わせることができます。

: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?

: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?

解説 バーグラフ表示のカーソル機能は、高調波測定機能 (オプション、/G5 または /G6) 搭載時のみ有効です。

:CURSor:BAR:LINKage

機能 バーグラフ表示のカーソル位置のリンケージ ON/OFF を設定 / 問い合わせします。

構文 :CURSor:BAR:LINKage {<Boolean>}

例 :CURSor:BAR:LINKage?

:CURSor:BAR:LINKage OFF

:CURSor:BAR:LINKage? ->

:CURSor:BAR:LINKage 0

:CURSor:BAR:POSition<x>

機能 バーグラフ表示のカーソル位置を設定 / 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 ~ 500

例 :CURSor:BAR:POSition1 1

:CURSor:BAR:POSition1? ->

:CURSor:BAR:POSition1 1

: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

: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 (DY1, DY2, DY3)

例 :CURSor:BAR:Y1? -> 78.628E+00

解説

- 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: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ます。

構文 :CURSor:TREnd?

:CURSor:TREnd:LINKage

機能 トレンド表示のカーソル位置のリンケージ ON/OFF を設定 / 問い合わせします。

構文 :CURSor:TREnd:LINKage {<Boolean>}

:CURSor:TREnd:LINKage?

例 :CURSor:TREnd:LINKage OFF

:CURSor:TREnd:LINKage? ->

:CURSor:TREnd:LINKage 0

: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 ~ 1601

例 :CURSor:TREnd:POSition1 160

:CURSor:TREnd:POSition1? ->

:CURSor:TREnd:POSition1 160

: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

5.5 CURSor グループ

: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 (DY)

例 :CURSOR:TREND:X1? ->
"2010/01/01 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、次のようになります。
X<x> の場合: "*****/**/** **.*.***" を返します。
Y<x>、DY の場合: "NAN(Not A Number)" を返します。

:CURSor:WAVE?

機能 波形表示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?

:CURSor:WAVE:LINKage

機能 波形表示のカーソル位置のリンケージ ON/OFF を設定 / 問い合わせします。

構文 :CURSor:WAVE:LINKage {<Boolean>}
:CURSor:WAVE:LINKage?

例 :CURSOR:WAVE:LINKAGE OFF
:CURSOR:WAVE:LINKAGE? ->
:CURSOR:WAVE:LINKAGE 0

: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> {<NRf>}
:CURSor:WAVE:POSition<x>?
<x> = 1、2 (1 = C1 +、2 = C2 x)
<NRf> = 0 ~ 800

例 :CURSOR:WAVE:POSITION1 160
:CURSOR:WAVE:POSITION1? ->
:CURSOR:WAVE:POSITION1 160

: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|AUX<x>}
:CURSor:WAVE:TRACe<x>?
TRACe<x> の <x> = 1、2 (1 = C1 +、2 = C2 x)
U<x>、I<x> の <x> = 1 ~ 6 (エレメント)
AUX<x> の <x> = 1 ~ 2 (AUX 入力チャネル)

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

解説 • {SPEEd|TORQue} は、モータ評価機能 (オプション、/MTR) 搭載時のみ選択可能です。
• AUX<x> は、外部信号入力 (オプション、/AUX) 搭載時のみ選択可能です。

: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 (DX)
PERDt = カーソル間の 1/DT 値 (1/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 (DY)

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

5.6 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?
解説 現在の表示方式 (:DISPlay:MODE) に関係するすべての設定値を返します。

:DISPlay:BAR?

機能 バーグラフ表示に関するすべての設定値を問い合わせます。
構文 :DISPlay:BAR?
解説 バーグラフ表示は、高調波測定機能 (オプション、/G5 または /G6) 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}
:DISPlay:BAR:FORMat?
例 :DISPlay:BAR:FORMat SINGle
:DISPlay:BAR:FORMat? ->
:DISPlay:BAR:FORMat SINGle

:DISPlay:BAR:ITEM<x>?

機能 各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:ITEM<x>?
<x> = 1 ~ 3 (項目番号)

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

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] {<Function>,<Element>}
:DISPlay:BAR:ITEM<x>:FUNCTION?
<x> = 1 ~ 3 (項目番号)
<Function> = {U|I|P|S|Q|LAMBda|PHI|PHIU|PHII|Z|RS|XS|RP|XP}
<Element> = 1 ~ 6
例 :DISPlay:BAR:ITEM1 U,1
:DISPlay:BAR:ITEM1? ->
:DISPlay:BAR:ITEM1 U,1
解説 <Function> の選択肢については、ファンクション選択肢一覧 (2)、5-41 ページを参照してください。

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

機能 バーグラフのスケール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:BAR:ITEM<x>:SCALing?
<x> = 1 ~ 3 (項目番号)

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

機能 バーグラフのスケールリング方式を設定 / 問い合わせします。
構文 :DISPlay:BAR:ITEM<x>:SCALing:MODE {FIXed|MANual}
:DISPlay:BAR:ITEM<x>:SCALing:MODE?
<x> = 1 ~ 3 (項目番号)
例 :DISPlay:BAR:ITEM1:SCALing:MODE FIXED
:DISPlay:BAR:ITEM1:SCALing:MODE? ->
:DISPlay:BAR:ITEM1:SCALing:MODE FIXED

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

機能 バーグラフのマニュアルスケールリング上限値を設定 / 問い合わせします。
構文 :DISPlay:BAR:ITEM<x>:SCALing:VALue {<NRf>}
:DISPlay:BAR:ITEM<x>:SCALing:VALue?
<x> = 1 ~ 3 (項目番号)
<NRf> = 0 ~ 9.999E+12
例 :DISPlay:BAR:ITEM1:SCALing:VALue 100
:DISPlay:BAR:ITEM1:SCALing:VALue? ->
:DISPlay:BAR:ITEM1:SCALing:VALue 100.0E+00
解説
・ バーグラフのスケールリング方式 (:DISPlay:BAR:ITEM<x>:SCALing:MODE) が「MANual」のときに有効な設定です。
・ 上限値のみ設定します。下限値は、垂直スケール形式 (:DISPlay:BAR:ITEM<x>:SCALing:VERTical) の設定により、以下のとおり自動的に決まります。
(1) 「LINear」のとき→ X 軸の位置 (:DISPlay:BAR:ITEM<x>:SCALing:XAXis) が「BOTTom」のとき「0」、「CENTer」のとき「- 上限値」となります。
(2) 「LOG」のとき→「上限値 / 10000」となります。

5.6 DISPLAY グループ

:DISPLAY:BAR:ITEM<x>:SCALing:VERTic al

機能 バーグラフの垂直スケール形式を設定 / 問い合わせします。

構文 :DISPLAY:BAR:ITEM<x>:SCALing:VERTic al {LINear|LOG}
:DISPLAY:BAR:ITEM<x>:SCALing:VERTic al?

例 <x> = 1 ~ 3 (項目番号)

```
:DISPLAY:BAR:ITEM1:SCALING:
VERTICAL LOG
:DISPLAY:BAR:ITEM1:SCALING:
VERTICAL? ->
:DISPLAY:BAR:ITEM1:SCALING:
VERTICAL LOG
```

解説 バーグラフのスケール方式 (:DISPLAY:BAR:ITEM<x>:SCALing:MODE) が「MANual」のときに有効な設定です。

:DISPLAY:BAR:ITEM<x>:SCALing:XAXis

機能 バーグラフの X 軸の位置を設定 / 問い合わせします。

構文 :DISPLAY:BAR:ITEM<x>:SCALing:XAXis {BOTTom|CENTer}
:DISPLAY:BAR:ITEM<x>:SCALing:XAXis?

例 <x> = 1 ~ 3 (項目番号)

```
:DISPLAY:BAR:ITEM1:SCALING:
XAXIS BOTTOM
:DISPLAY:BAR:ITEM1:SCALING:XAXIS? ->
:DISPLAY:BAR:ITEM1:SCALING:
XAXIS BOTTOM
```

解説 バーグラフのスケール方式 (:DISPLAY:BAR:ITEM<x>:SCALing:MODE) が「MANual」で、垂直スケール形式 (:DISPLAY:BAR:ITEM<x>:SCALing:VERTic al) が「LINear」のときに有効な設定です。

:DISPLAY:BAR:ORDER

機能 バーグラフの表示開始 / 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 ~ 490 (表示開始次数)
2 つ目の <NRf> = 10 ~ 500 (表示終了次数)

例 :DISPLAY:BAR:ORDER 1,100
:DISPLAY:BAR:ORDER? ->
:DISPLAY:BAR:ORDER 1,100

解説

- ・ 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。

:DISPLAY:HSPeed?

機能 高速データ収集モードの表示に関するすべての設定値を問い合わせます。

構文 :DISPLAY:HSPeed?

解説 高速データ収集モード表示は、高速データ収集機能 (オプション、/HS) 搭載時のみ有効です。

:DISPLAY:HSPeed:COLumn?

機能 高速データ収集モードのカラムに関するすべての設定値を問い合わせます。

構文 :DISPLAY:HSPeed:COLumn?

:DISPLAY:HSPeed:COLumn:ITEM<x>

機能 高速データ収集モードのカラム表示項目を設定 / 問い合わせします。

構文 :DISPLAY:HSPeed:COLumn:ITEM<x> {NONE|<Element>}
:DISPLAY:HSPeed:COLumn:ITEM<x>?
<x> = 1 ~ 6 (カラム番号)
<Element> = {<NRf>|SIGMa|SIGMB|SIGMC}
(<NRf> = 1 ~ 6)

例 :DISPLAY:HSPeed:COLumn:ITEM1 1
:DISPLAY:HSPeed:COLumn:ITEM1? ->
:DISPLAY:HSPeed:COLumn:ITEM1 1

:DISPLAY:HSPeed:COLumn:NUMBER

機能 高速データ収集モードの表示カラム数を設定 / 問い合わせします。

構文 :DISPLAY:HSPeed:COLumn:NUMBER {<NRf>}
:DISPLAY:HSPeed:COLumn:NUMBER?
<NRf> = 4、6

例 :DISPLAY:HSPeed:COLumn:NUMBER 4
:DISPLAY:HSPeed:COLumn:NUMBER? ->
:DISPLAY:HSPeed:COLumn:NUMBER 4

:DISPLAY:HSPeed:COLumn:RESet

機能 高速データ収集モードのカラム表示項目を初期値にリセットします。

構文 :DISPLAY:HSPeed:COLumn:RESet
例 :DISPLAY:HSPeed:COLumn:RESet

:DISPLAY:HSPeed:FRAMe

機能 高速データ収集モードのデータ部のフレーム表示 ON/OFF を設定 / 問い合わせします。

構文 :DISPLAY:HSPeed:FRAMe {<Boolean>}
:DISPLAY:HSPeed:FRAMe?

例 :DISPLAY:HSPeed:FRAMe ON
:DISPLAY:HSPeed:FRAMe? ->
:DISPLAY:HSPeed:FRAMe 1

解説 「:DISPLAY:NUMeric:FRAMe」と共通の設定です。

:DISPlay:HSPEED:PAGE

機能 高速データ収集モードの表示ページを設定 / 問い合わせします。

構文 :DISPlay:HSPEED:PAGE {<NRf>}
 :DISPlay:HSPEED:PAGE?
 <NRf> = 1 ~ 2 (ページ番号)
 <NRf> = 1 ~ 4 (モータ評価機能 (オプション、/MTR) または外部信号入力 (オプション、/AUX) 搭載時)

例 :DISPLAY:HSPEED:PAGE 1
 :DISPLAY:HSPEED:PAGE? ->
 :DISPLAY:HSPEED:PAGE 1

:DISPlay:HSPEED:POVer

機能 高速データ収集モードのピークオーバー発生情報の表示 ON/OFF を設定 / 問い合わせします。

構文 :DISPlay:HSPEED:POVer {<Boolean>}
 :DISPlay:HSPEED:POVer?

例 :DISPLAY:HSPEED:POVER OFF
 :DISPLAY:HSPEED:POVER? ->
 :DISPLAY:HSPEED:POVER 0

: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?

:DISPlay:INfOrMation:PAGE

機能 設定情報の一覧表示における表示ページを設定 / 問い合わせします。

構文 :DISPlay:INfOrMation:PAGE {Power|
 RANGE|<NRf>}
 :DISPlay:INfOrMation:PAGE?
 Power (または <NRf> = 1) =
 各エレメントの測定条件の対応表 (Power
 Element Settings)
 RANGE (または <NRf> = 2) =
 各エレメントの電圧 / 電流レンジ設定状態を表
 すインジケータ (Range Settings)

例 :DISPLAY:INFORMATION:PAGE POWER
 :DISPLAY:INFORMATION:PAGE? ->
 :DISPLAY:INFORMATION:PAGE POWER

: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:MODE

機能 表示方式を設定 / 問い合わせします。

構文 :DISPlay:MODE {NUMeric|WAVE|TREnd|
 BAR|VECTor|NWAve|NTREnd|NBAR|
 NVECTor|WNUMeric|WTREnd|WBAR|
 WVECTor|TNUMeric|TWAve|TBAR|TVECTor|
 HSPeet}
 :DISPlay:MODE?
 NUMeric = 数値表示
 WAVE = 波形表示
 TREnd = トレンド
 BAR = バーグラフ
 VECTor = ベクトル

NWAve = 数値と波形を両方表示
 NTREnd = 数値とトレンドを両方表示
 NBAR = 数値とバーグラフを両方表示
 NVECTor = 数値とベクトルを両方表示
 WNUMeric = 波形と数値を両方表示
 WTREnd = 波形とトレンドを両方表示
 WBAR = 波形とバーグラフを両方表示
 WVECTor = 波形とベクトルを両方表示
 TNUMeric = トレンドと数値を両方表示
 TWAve = トレンドと波形を両方表示
 TBAR = トレンドとバーグラフを両方表示
 TVECTor = トレンドとベクトルを両方表示
 HSPeet = 高速データ収集モード表示 (数値)

例 :DISPLAY:MODE NUMERIC
 :DISPLAY:MODE? ->
 :DISPLAY:MODE NUMERIC

解説 • {BAR|VECTor|NBAR|NVECTor|WBAR|WVECTor|TBAR|TVECTor} は、高調波測定機能 (オプション、/G5 または /G6) 搭載時のみ選択可能です。
 • HSPeet は、高速データ収集 (オプション、/HS) 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?

:DISPlay:NUMeric:CUSTOm?

機能 数値表示 (カスタム表示) に関するすべての設定値を問い合わせます。

構文 :DISPlay:NUMeric:CUSTOm?

:DISPlay:NUMeric:CUSTOm:FILE:CDIRectory

機能 数値表示 (カスタム表示) のファイル読み込み / 保存先ディレクトリを変更します。

構文 :DISPlay:NUMeric:CUSTOm:FILE:CDIRectory {<文字列>}

<文字列> = ディレクトリ名

例 :DISPLAY:NUMERIC:CUSTOM:FILE:CDIRECTORY "CUSTOM"

解説 上のディレクトリに移動するには、"." を指定します。

5.6 DISPlay グループ

:DISPlay:NUMeric:CUSTom:FILE:DRIVE

機能 数値表示 (カスタム表示) のファイル読み込み / 保存先ドライブを設定します。

構文 :DISPlay:NUMeric:CUSTom:FILE:DRIVE {RAM|USB[,<Nrf>]|NETWork}
RAM = 内蔵 RAM ドライブ
USB = USB メモリドライブ、
<Nrf> = 0、1 (ドライブ番号)
NETWork = ネットワークドライブ

例 :DISPlay:NUMeric:CUSTom:FILE:DRIVE USB,0

:DISPlay:NUMeric:CUSTom:FILE:FREE?

機能 数値表示 (カスタム表示) のファイル読み込み / 保存先ドライブの空き容量 (byte) を問い合わせます。

構文 :DISPlay:NUMeric:CUSTom:FILE:FREE?

例 :DISPlay:NUMeric:CUSTom:FILE:FREE? -> 20912128

:DISPlay:NUMeric:CUSTom:FILE:LOAD:ABORT

機能 数値表示 (カスタム表示) のファイルの読み込みを中止します。

構文 :DISPlay:NUMeric:CUSTom:FILE:LOAD:ABORT

例 :DISPlay:NUMeric:CUSTom:FILE:LOAD:ABORT

:DISPlay:NUMeric:CUSTom:FILE:LOAD:BACKGROUND

機能 数値表示 (カスタム表示) の背景ファイルの読み込みを実行します。

構文 :DISPlay:NUMeric:CUSTom:FILE:LOAD:BACKGROUND {<文字列>}
<文字列> = ファイル名

例 :DISPlay:NUMeric:CUSTom:FILE:LOAD:BACKGROUND "CUSTOM1"

解説

- ・ ファイル名は、拡張子 (.bmp) 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。

:DISPlay:NUMeric:CUSTom:FILE:LOAD:BACKGROUND

機能 数値表示 (カスタム表示) の表示構成 & 背景ファイルの読み込みを実行します。

構文 :DISPlay:NUMeric:CUSTom:FILE:LOAD:BACKGROUND {<文字列>}
<文字列> = ファイル名

例 :DISPlay:NUMeric:CUSTom:FILE:LOAD:BACKGROUND "CUSTOM1"

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。

:DISPlay:NUMeric:CUSTom:FILE:LOAD:ITEM

機能 数値表示 (カスタム表示) の表示構成ファイルの読み込みを実行します。

構文 :DISPlay:NUMeric:CUSTom:FILE:LOAD:ITEM {<文字列>}
<文字列> = ファイル名

例 :DISPlay:NUMeric:CUSTom:FILE:LOAD:ITEM "CUSTOM1"

解説

- ・ ファイル名は、拡張子 (.txt) 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。

:DISPlay:NUMeric:CUSTom:FILE:PATH?

機能 数値表示 (カスタム表示) のファイル読み込み / 保存先を絶対パスで問い合わせます。

構文 :DISPlay:NUMeric:CUSTom:FILE:PATH?

例 :DISPlay:NUMeric:CUSTom:FILE:PATH? -> "USB-0/CUSTOM"

:DISPlay:NUMeric:CUSTom:FILE:SAVE:ANAMING

機能 数値表示 (カスタム表示) の表示構成を保存するファイル名の自動生成機能を設定 / 問い合わせします。

構文 :DISPlay:NUMeric:CUSTom:FILE:SAVE:ANAMING {OFF|NUMBERING|DATE}
:DISPlay:NUMeric:CUSTom:FILE:SAVE:ANAMING?

例 :DISPlay:NUMeric:CUSTom:FILE:SAVE:ANAMING NUMBERING
:DISPlay:NUMeric:CUSTom:FILE:SAVE:ANAMING? ->
:DISPlay:NUMeric:CUSTom:FILE:SAVE:ANAMING NUMBERING

:DISPlay:NUMeric:CUSTom:FILE:SAVE:ITEM

機能 数値表示 (カスタム表示) の表示構成ファイルの保存を実行します。

構文 :DISPlay:NUMeric:CUSTom:FILE:SAVE:ITEM {<文字列>}
<文字列> = ファイル名

例 :DISPlay:NUMeric:CUSTom:FILE:SAVE:ITEM "CUSTOM1"

解説

- ・ ファイル名は、拡張子 (.txt) 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。

:DISPlay:NUMeric:CUSTom:ITEM<x>?

機能 数値表示 (カスタム表示) の各表示項目に関するすべての設定値を問い合わせます。

構文 :DISPlay:NUMeric:CUSTom:ITEM<x>?
<x> = 1 ~ 192 (項目番号)

:DISPLAY:NUMERIC:CUSTOM:ITEM<x>:COLOR

機能 数値表示 (カスタム表示) の表示項目の文字色を設定 / 問い合わせします。

構文 :DISPLAY:NUMERIC:CUSTOM:ITEM<x>:COLOR {YELLOW|GREEN|MAGENTA|CYAN|RED|ORANGE|LBLUE|PURPLE|BLUE|PINK|LGREEN|DBLUE|BGREEN|SPINK|MGREEN|GRAY|WHITE|DGRAY|BGRAY|BLACK}
:DISPLAY:NUMERIC:CUSTOM:ITEM<x>:COLOR?
<x> = 1 ~ 192 (項目番号)
YELLOW = 黄
GREEN = 緑
MAGENTA = マゼンタ
CYAN = シアン
RED = 赤
ORANGE = オレンジ
LBLUE = 明るい青 (Light Blue)
PURPLE = 紫
BLUE = 青
PINK = ピンク
LGREEN = 明るい緑 (Light Green)
DBLUE = 暗い青 (Dark Blue)
BGREEN = 青緑 (Blue Green)
SPINK = サーマン (Salmon Pink)
MGREEN = 薄い緑 (Mild Green)
GRAY = グレー
WHITE = 白
DGRAY = 暗いグレー (Dark Gray)
BGRAY = 青いグレー (Blue Gray)
BLACK = 黒

例 :DISPLAY:NUMERIC:CUSTOM:ITEM1:COLOR WHITE
:DISPLAY:NUMERIC:CUSTOM:ITEM1:COLOR? ->
:DISPLAY:NUMERIC:CUSTOM:ITEM1:COLOR WHITE

:DISPLAY:NUMERIC:CUSTOM:ITEM<x>[:FUNCTION]

機能 数値表示 (カスタム表示) の表示項目 (数値項目または文字列) を設定 / 問い合わせします。

構文 :DISPLAY:NUMERIC:CUSTOM:ITEM<x>[:FUNCTION] {<Function>[,<Element>][,<Order>]|<文字列>}
:DISPLAY:NUMERIC:CUSTOM:ITEM<x>[:FUNCTION]?
<x> = 1 ~ 192 (項目番号)
・数値項目を設定する場合
<Function> = {URMS|IRMS|P|S|Q|...}
<Element> = {<NRF>|SIGMA|SIGMB|SIGMC}
(<NRF> = 1 ~ 6)
<Order> = {TOTAL|DC|<NRF>}
(<NRF> = 1 ~ 500)
・文字列を設定する場合
<文字列> = 16 文字以内
・数値項目を設定する場合

例 :DISPLAY:NUMERIC:CUSTOM:ITEM1:FUNCTION URMS,1
:DISPLAY:NUMERIC:CUSTOM:ITEM1:FUNCTION? ->
:DISPLAY:NUMERIC:CUSTOM:ITEM1:FUNCTION URMS,1
:DISPLAY:NUMERIC:CUSTOM:ITEM1:FUNCTION UK,1,1
:DISPLAY:NUMERIC:CUSTOM:ITEM1:FUNCTION? ->
:DISPLAY:NUMERIC:CUSTOM:ITEM1:FUNCTION UK,1,1
・文字列を設定する場合
:DISPLAY:NUMERIC:CUSTOM:ITEM1:FUNCTION "YOKOGAWA"
:DISPLAY:NUMERIC:CUSTOM:ITEM1:FUNCTION? ->
:DISPLAY:NUMERIC:CUSTOM:ITEM1:FUNCTION "YOKOGAWA"

解説
・表示項目として、数値項目または文字列のどちらかを設定します。
(1) 数値項目を設定する場合
・<Function> の選択肢については、ファンクション選択肢一覧 (1)、5-38 ページを参照してください。
・<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。
(2) 文字列を設定する場合
・数値項目のヘッダや単位など、任意の文字列を表示させることができます。

5.6 DISPLAY グループ

:DISPlay:NUMeric:CUSTom:ITEM<x>:POSITION

機能	数値表示 (カスタム表示) の表示項目の表示位置を設定 / 問い合わせします。
構文	:DISPlay:NUMeric:CUSTom:ITEM<x>:POSITION {<NRf>,<NRf>} :DISPlay:NUMeric:CUSTom:ITEM<x>:POSITION?
例	<x> = 1 ~ 192 (項目番号) 1 つめの <NRf> = 0 ~ 800 (X 座標) 2 つめの <NRf> = 0 ~ 672 (Y 座標) :DISPlay:NUMeric:CUSTom:ITEM1:POSITION 0,0 :DISPlay:NUMeric:CUSTom:ITEM1:POSITION? -> :DISPlay:NUMeric:CUSTom:ITEM1:POSITION 0,0
解説	数値データ表示エリアの左上を原点として、各表示項目の左上の座標を指定します。

:DISPlay:NUMeric:CUSTom:ITEM<x>:SIZE

機能	数値表示 (カスタム表示) の表示項目の文字サイズを設定 / 問い合わせします。
構文	:DISPlay:NUMeric:CUSTom:ITEM<x>:SIZE {<NRf>} :DISPlay:NUMeric:CUSTom:ITEM<x>:SIZE?
例	<x> = 1 ~ 192 (項目番号) <NRf> = 14、16、20、24、32、48、64、96、128 :DISPlay:NUMeric:CUSTom:ITEM1:SIZE 20 :DISPlay:NUMeric:CUSTom:ITEM1:SIZE? -> :DISPlay:NUMeric:CUSTom:ITEM1:SIZE 20

:DISPlay:NUMeric:CUSTom:PAGE

機能	数値表示 (カスタム表示) の表示ページを設定 / 問い合わせします。
構文	:DISPlay:NUMeric:CUSTom:PAGE {<NRf>} :DISPlay:NUMeric:CUSTom:PAGE? <NRf> = 1 ~ 12 (ページ番号)
例	:DISPlay:NUMeric:CUSTom:PAGE 1 :DISPlay:NUMeric:CUSTom:PAGE? -> :DISPlay:NUMeric:CUSTom:PAGE 1
解説	表示ページの最大値は、全表示項目数および 1 ページあたりの項目数で決まります。

:DISPlay:NUMeric:CUSTom:PERPage

機能	数値表示 (カスタム表示) の 1 ページあたりの項目数を設定 / 問い合わせします。
構文	:DISPlay:NUMeric:CUSTom:PERPage {<NRf>} :DISPlay:NUMeric:CUSTom:PERPage? <NRf> = 1 ~ 全表示項目数
例	:DISPlay:NUMeric:CUSTom:PERPage 5 :DISPlay:NUMeric:CUSTom:PERPage? -> :DISPlay:NUMeric:CUSTom:PERPage 5
解説	1 ページあたりの項目数の最小値は、全表示項目数 (:DISPlay:NUMeric:CUSTom:TOTal) / 12 となります。

:DISPlay:NUMeric:CUSTom:TOTal

機能	数値表示 (カスタム表示) の全表示項目数を設定 / 問い合わせします。
構文	:DISPlay:NUMeric:CUSTom:TOTal {<NRf>} :DISPlay:NUMeric:CUSTom:TOTal? <NRf> = 1 ~ 192 (項目数)
例	:DISPlay:NUMeric:CUSTom:TOTal 20 :DISPlay:NUMeric:CUSTom:TOTal? -> :DISPlay:NUMeric:CUSTom:TOTal 20
解説	全表示項目数の最大値は、1 ページあたりの項目数 (:DISPlay:NUMeric:CUSTom:PERPage) × 12 となります。

:DISPlay:NUMeric:FRAME

機能	数値表示のデータ部のフレーム表示 ON/OFF を設定 / 問い合わせします。
構文	:DISPlay:NUMeric:FRAME {<Boolean>} :DISPlay:NUMeric:FRAME?
例	:DISPlay:NUMeric:FRAME ON :DISPlay:NUMeric:FRAME? -> :DISPlay:NUMeric:FRAME 1

: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します。
構文	:DISPlay:NUMeric:NORMal?
解説	現在の数値表示方式 (:DISPlay:NUMeric[:NORMal]:FORMat) に関係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します。
構文	:DISPlay:NUMeric[:NORMal]:ALL?

**:DISPlay:NUMeric[:NORMal]:ALL:COLu
mn?**

機能	数値表示 (全表示) のカラム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:COLu mn?
解説	カラム表示に関する設定は、5 エlement以上のモデルのみ有効です。

**:DISPlay:NUMeric[:NORMal]:ALL:COLumn
:DAELem**

機能	数値表示 (全表示) のカラム全体表示機能の ON/OFF を設定 / 問い合わせします。
構文	:DISPlay:NUMeric[:NORMal]:ALL:COLumn :DAELem {<Boolean>}
例	:DISPlay:NUMeric[:NORMal]:ALL:COLumn :DAELem? :DISPlay:NUMeric[:NORMal]:ALL:COLumn :DAELem? :DISPlay:NUMeric[:NORMal]:ALL:COLumn:D AELEM ON :DISPlay:NUMeric[:NORMal]:ALL:COLumn:D AELEM? -> :DISPlay:NUMeric[:NORMal]:ALL:COLumn:D AELEM 1
解説	結線方式の設定により表示すべきカラム (エレメント / Σ) 数が 6 を超えたとき、自動的に文字の大きさを小さくしてカラム全体を表示する機能です。 (Display All Elements)

**:DISPlay:NUMeric[:NORMal]:ALL:COLumn
:SCROLL**

機能	数値表示 (全表示) のカラムのスクロールを設定 / 問い合わせします。
構文	:DISPlay:NUMeric[:NORMal]:ALL:COLumn :SCROLL {<Nrf>}
例	:DISPlay:NUMeric[:NORMal]:ALL:COLumn :SCROLL? <Nrf> = 0 ~ 3 (スクロール量) :DISPlay:NUMeric[:NORMal]:ALL:COLumn:S CROLL 0 :DISPlay:NUMeric[:NORMal]:ALL:COLumn:S CROLL? -> :DISPlay:NUMeric[:NORMal]:ALL:COLumn:S CROLL 0
解説	カラム全体表示機能 (:DISPlay:NUMeric[:NORMal]:ALL:COLumn:DAELem) が OFF のときに有効な設定です。

: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>}
例	:DISPlay:NUMeric[:NORMal]:ALL:CURS or? <Function> = {URMS IRMS P S Q ...} :DISPlay:NUMeric[:NORMal]:ALL:CURSOR P :DISPlay:NUMeric[:NORMal]:ALL: CURSOR? -> :DISPlay:NUMeric[:NORMal]:ALL:CURSOR P
解説	- カーソル位置はファンクション名で指定します。 <Function> の選択肢については、ファンクション選択肢一覧 (1)、5-38 ページを参照してください。

: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	:DISPlay:NUMeric[:NORMal]:ALL: ORDER {<Order>}
例	:DISPlay:NUMeric[:NORMal]:ALL: ORDER? <Order> = {TOTal DC <Nrf>} (<Nrf> = 1 ~ 500) :DISPlay:NUMeric[:NORMal]:ALL:ORDER 1 :DISPlay:NUMeric[:NORMal]:ALL: ORDER? -> :DISPlay:NUMeric[:NORMal]:ALL:ORDER 1
解説	- 高調波測定機能 (オプション、/G5 または /G6) 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 9 ~ 10 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) の表示ページを設定 / 問い合わせします。
構文	:DISPlay:NUMeric[:NORMal]:ALL:PAGE {<Nrf>}
例	:DISPlay:NUMeric[:NORMal]:ALL:PAGE? <Nrf> = 1 ~ 7 (ページ番号) <Nrf> = 1 ~ 8 (デルタ演算機能 (オプション、/DT) 搭載時) <Nrf> = 1 ~ 12 (高調波測定機能 (オプション、/G5 または /G6) 搭載時)
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。

5.6 DISPlay グループ

: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	:DISPlay:NUMeric[:NORMal]:FORMat {VAL4 VAL8 VAL16 MATRIX ALL SINGLE DUAL CUSTom}
例	:DISPlay:NUMeric[:NORMal]:FORMat? :DISPlay:NUMeric[:NORMal]:FORMat VAL4 :DISPlay:NUMeric[:NORMal]:FORMat? -> :DISPlay:NUMeric[:NORMal]:FORMat VAL4
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) MATRIX = 任意のファンクションをエレメントごとに並べて表示します。 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。 CUSTom = 任意の数値表示項目を、指定されたビットマップ背景上に表示します。 {SINGLE DUAL} は、高調波測定機能 (オプション、/G5 または /G6) 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	:DISPlay:NUMeric[:NORMal]:LIST?
解説	リスト表示は、高調波測定機能 (オプション、/G5 または /G6) 搭載時のみ有効です。

:DISPlay:NUMeric[:NORMal]:LIST:CURS or

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	:DISPlay:NUMeric[:NORMal]:LIST:CURS or {HEADer ORDER} :DISPlay:NUMeric[:NORMal]:LIST:CURS or?
例	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。 :DISPlay:NUMeric[:NORMal]:LIST: CURSOR ORDER :DISPlay:NUMeric[:NORMal]:LIST: CURSOR? -> :DISPlay:NUMeric[:NORMal]:LIST: CURSOR ORDER

:DISPlay:NUMeric[:NORMal]:LIST:HEAD er

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	:DISPlay:NUMeric[:NORMal]:LIST:HEAD er {<NRf>} :DISPlay:NUMeric[:NORMal]:LIST:HEAD er?
例	<NRf> = 1 ~ 155 (ヘッダ行) :DISPlay:NUMeric[:NORMal]:LIST: HEADER 1 :DISPlay:NUMeric[:NORMal]:LIST: HEADER? -> :DISPlay:NUMeric[:NORMal]:LIST: HEADER 1
解説	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	:DISPlay:NUMeric[:NORMal]:LIST:ITEM <x> {<Function>,<Element>} :DISPlay:NUMeric[:NORMal]:LIST:ITEM <x>?
例	<x> = 1、2 (項目番号) <Function> = {U I P S Q LAMBda PHI PHIU PHII Z RS XS RP XP} <Element> = {<NRf> SIGMa SIGMB SIGMC} (<NRf> = 1 ~ 6) :DISPlay:NUMeric[:NORMal]:LIST: ITEM1 U,1 :DISPlay:NUMeric[:NORMal]:LIST: ITEM1? -> :DISPlay:NUMeric[:NORMal]:LIST: ITEM1 U,1
解説	<Function> の選択肢については、ファンクション選択肢一覧 (2)、5-41 ページを参照してください。

: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 :DISPlay:NUMeric[:NORMal]:LIST:ORDER {<NRf>}
:DISPlay:NUMeric[:NORMal]:LIST:ORDER?
<NRf> = 1 ~ 500 (次数)

例 :DISPlay:NUMeric:NORMal:LIST:ORDER 1
:DISPlay:NUMeric:NORMal:LIST:ORDER? ->
:DISPlay:NUMeric:NORMal:LIST:ORDER 1

解説 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]:MATRix?

機能 数値表示 (マトリクス表示) に関するすべての設定値を問い合わせます。

構文 :DISPlay:NUMeric[:NORMal]:MATRix?

:DISPlay:NUMeric[:NORMal]:MATRix:COLumn?

機能 数値表示 (マトリクス表示) のコラムに関するすべての設定値を問い合わせます。

構文 :DISPlay:NUMeric[:NORMal]:MATRix:COLumn?

:DISPlay:NUMeric[:NORMal]:MATRix:COLumn:ITEM<x>

機能 数値表示 (マトリクス表示) のコラム表示項目を設定 / 問い合わせします。

構文 :DISPlay:NUMeric[:NORMal]:MATRix:COLumn:ITEM<x> {NONE|<Element>}
:DISPlay:NUMeric[:NORMal]:MATRix:COLumn:ITEM<x>?
<x> = 1 ~ 6 (カラム番号)
<Element> = {<NRf>|SIGMa|SIGMB|SIGMC} (<NRf> = 1 ~ 6)

例 :DISPlay:NUMeric:NORMal:MATRIX:COLUMN:ITEM1 1
:DISPlay:NUMeric:NORMal:MATRIX:COLUMN:ITEM1? ->
:DISPlay:NUMeric:NORMal:MATRIX:COLUMN:ITEM1 1

:DISPlay:NUMeric[:NORMal]:MATRix:COLumn:NUMBER

機能 数値表示 (マトリクス表示) のコラム数を設定 / 問い合わせします。

構文 :DISPlay:NUMeric[:NORMal]:MATRix:COLumn:NUMBER {<NRf>}
:DISPlay:NUMeric[:NORMal]:MATRix:COLumn:NUMBER?
<NRf> = 4、6

例 :DISPlay:NUMeric:NORMal:MATRIX:COLUMN:NUMBER 4
:DISPlay:NUMeric:NORMal:MATRIX:COLUMN:NUMBER? ->
:DISPlay:NUMeric:NORMal:MATRIX:COLUMN:NUMBER 4

:DISPlay:NUMeric[:NORMal]:MATRix:COLumn:RESet

機能 数値表示 (マトリクス表示) のコラム表示項目を初期値にリセットします。

構文 :DISPlay:NUMeric[:NORMal]:MATRix:COLumn:RESet

例 :DISPlay:NUMeric:NORMal:MATRIX:COLUMN:RESET

:DISPlay:NUMeric[:NORMal]:MATRix:CURSor

機能 数値表示 (マトリクス表示) のカーソル位置を設定 / 問い合わせします。

構文 :DISPlay:NUMeric[:NORMal]:MATRix:CURSor {<NRf>}
:DISPlay:NUMeric[:NORMal]:MATRix:CURSor?
<NRf> = 1 ~ 81 (項目番号)

例 :DISPlay:NUMeric:NORMal:MATRIX:CURSOR 1
:DISPlay:NUMeric:NORMal:MATRIX:CURSOR? ->
:DISPlay:NUMeric:NORMal:MATRIX:CURSOR 1

解説 カーソル位置は項目番号で指定します。

5.6 DISPlay グループ

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

機能	数値表示 (マトリクス表示) の表示項目 (ファンクション・次数) を設定 / 問い合わせします。
構文	:DISPlay:NUMeric[:NORMal]:MATRix:ITEM<x> {NONE <Function>[,<Element>][,<Order>]}
	:DISPlay:NUMeric[:NORMal]:MATRix:ITEM<x>?
	<x> = 1 ~ 81 (項目番号)
	NONE = 表示項目なし
	<Function> = {URMS IRMS P S Q ...}
	<Element> = {<Nrf> SIGMa SIGMB SIGMC}
	(<Nrf> = 1 ~ 6)
	<Order> = {TOTal DC <Nrf>}
	(<Nrf> = 1 ~ 500)
例	:DISPlay:NUMeric:NORMal:MATRIX:ITEM1 URMS
	:DISPlay:NUMeric:NORMal:MATRIX:ITEM1? ->
	:DISPlay:NUMeric:NORMal:MATRIX:ITEM1 URMS,1
	:DISPlay:NUMeric:NORMal:MATRIX:ITEM1 UK,1,1
	:DISPlay:NUMeric:NORMal:MATRIX:ITEM1? ->
	:DISPlay:NUMeric:NORMal:MATRIX:ITEM1 UK,1,1
解説	・ <Function> の選択肢については、ファンクション選択肢一覧 (1)、5-38 ページを参照してください。 ・ <Element> の設定は表示に影響しません。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]:MATRix:PAGE

機能	数値表示 (マトリクス表示) の表示ページを設定 / 問い合わせします。
構文	:DISPlay:NUMeric[:NORMal]:MATRix:PAGE {<Nrf>}
	:DISPlay:NUMeric[:NORMal]:MATRix:PAGE?
	<Nrf> = 1 ~ 9 (ページ番号)
例	:DISPlay:NUMeric:NORMal:MATRIX:PAGE 1
	:DISPlay:NUMeric:NORMal:MATRIX:PAGE? ->
	:DISPlay:NUMeric:NORMal:MATRIX: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]:MATRix:PRESet

機能	数値表示 (マトリクス表示) 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]:MATRix:PRESet {<Nrf> EORigin FORigin CLRPage CLRA11}
	<Nrf> = 1 または EORigin = エレメント基準のリセットパタン (Element Origin)
	<Nrf> = 2 または FORigin = ファンクション基準のリセットパタン (Function Origin)
	<Nrf> = 3 または CLRPage = 現在のページの表示項目をクリア (Clear Current Page)
	<Nrf> = 4 または CLRA11 = すべてのページの表示項目をクリア (Clear All Pages)
例	:DISPlay:NUMeric:NORMal:MATRIX:PRESET 1
	:DISPlay:NUMeric:NORMal:MATRIX:PRESET EORIGIN
解説	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、スタートガイド IM WT1801-03JA を参照してください。

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

: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 ~ 48 (項目番号、VAL4 のとき)
	<Nrf> = 1 ~ 96 (項目番号、VAL8 のとき)
	<Nrf> = 1 ~ 192 (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 ~ 48 (項目番号、VAL4 のとき)
<x> = 1 ~ 96 (項目番号、VAL8 のとき)
<x> = 1 ~ 192 (項目番号、VAL16 のとき)
NONE = 表示項目なし
<Function> = {URMS|IRMS|P|S|Q|...}
<Element> = {<Nrf>|SIGMA|SIGMB|SIGMC}
(<Nrf> = 1 ~ 6)
<Order> = {TOTAL|DC|<Nrf>}
(<Nrf> = 1 ~ 500)

例 :DISPLAY:NUMERIC:NORMAL:VAL4:
ITEM1 URMS,1
:DISPLAY:NUMERIC:NORMAL:VAL4:
ITEM1? ->
:DISPLAY:NUMERIC:NORMAL:VAL4:
ITEM1 URMS,1
:DISPLAY:NUMERIC:NORMAL:VAL4:
ITEM1 UK,1,1
:DISPLAY:NUMERIC:NORMAL:VAL4:
ITEM1? ->
:DISPLAY:NUMERIC:NORMAL:VAL4:
ITEM1 UK,1,1

解説

- ・ <Function> の選択肢については、ファンクション選択肢一覧 (1)、5-38 ページを参照してください。
- ・ <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 ~ 12 (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>|
EORIGIN|FORIGIN|CLRPage|CLRAll}
<Nrf> = 1 または EORIGIN = エレメント基準
のリセットパターン (Element Origin)
<Nrf> = 2 または FORIGIN = ファンクション
基準のリセットパターン (Function Origin)
<Nrf> = 3 または CLRPage = 現在のページの
表示項目をクリア (Clear Current Page)
<Nrf> = 4 または CLRAll = すべてのページの
表示項目をクリア (Clear All Pages)

例 :DISPLAY:NUMERIC:NORMAL:VAL4:
PRESET 1
:DISPLAY:NUMERIC:NORMAL:VAL4:
PRESET EORIGIN

解説 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、スタートガイド IM WT1801-03JA を参照してください。

:DISPLAY:TREND?

機能 トレンド表示に関するすべての設定値を問い合わせます。

構文 :DISPLAY:TREND?

: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

5.6 DISPlay グループ

: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)

: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]
 {<Function>,<Element>[,<Order>]}
 :DISPlay:TREND:ITEM<x>:FUNCTION?
 <x> = 1 ~ 16 (項目番号)
 <Function> = {URMS|IRMS|P|S|Q|...}
 <Element> = {<NRf>|SIGMa|SIGMB|SIGMC}
 (<NRf> = 1 ~ 6)
 <Order> = {TOTal|DC|<NRf>}
 (<NRf> = 1 ~ 500)
例 :DISPlay:TREND:ITEM1:FUNCTION URMS,1
 :DISPlay:TREND:ITEM1:FUNCTION? ->
 :DISPlay:TREND:ITEM1:FUNCTION URMS,1
 :DISPlay:TREND:ITEM1:FUNCTION UK,1,1
 :DISPlay:TREND:ITEM1:FUNCTION? ->
 :DISPlay:TREND:ITEM1:FUNCTION UK,1,1
解説 ・ <Function> の選択肢については、ファンク
 ション選択肢一覧 (1)、5-38 ページを参照して
 ください。
 ・ <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)

: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
 します。
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:VAL
 ue?
 <x> = 1 ~ 16 (項目番号)
 <NRf> = - 9.999E+12 ~ 9.999E+12
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:I
 TEM<x>:SCALIng:MODE) が「MANual」のとき
 に有効な設定です。

:DISPlay:TREND:T<x>

機能 各トレンドの 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: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: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?

解説 ベクトル表示は、高調波測定機能 (オプション、/G5 または /G6) 搭載時のみ有効です。

:DISPlay:VECTor:FORMat

機能 ベクトルの表示フォーマットを設定 / 問い合わせします。

構文 :DISPlay:VECTor:FORMat {SINGLE|DUAL}
 :DISPlay:VECTor:FORMat?

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

:DISPlay:VECTor:ITEM<x>?

機能 各ベクトルに関するすべての設定値を問い合わせます。

構文 :DISPlay:VECTor:ITEM<x>?
 <x> = 1 ~ 2 (項目番号)

:DISPlay:VECTor:ITEM<x>:OBJect

機能 ベクトル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:ITEM<x>:
 OBJect {<Element>}
 :DISPlay:VECTor:ITEM<x>:OBJect?
 <x> = 1 ~ 2 (項目番号)
 <Element> = {<NRf>|SIGMa|SIGMB|SIGMC}
 (<NRf> = 1 ~ 6)

例 :DISPLAY:VECTOR:ITEM1:OBJECT SIGMA
 :DISPLAY:VECTOR:ITEM1:OBJECT? ->
 :DISPLAY:VECTOR:ITEM1:OBJECT SIGMA

:DISPlay:VECTor:ITEM<x>: {UMAG|IMAG}

機能 ベクトル表示の {電圧 | 電流} ズーム率を設定 / 問い合わせします。

構文 :DISPlay:VECTor:ITEM<x>: {UMAG|IMAG}
 {<NRf>}
 :DISPlay:VECTor:ITEM<x>: {UMAG|IMAG}?
 <x> = 1 ~ 2 (項目番号)
 <NRf> = 0.100 ~ 100.000

例 :DISPLAY:VECTOR:ITEM1:UMAG 1
 :DISPLAY:VECTOR:ITEM1:UMAG? ->
 :DISPLAY:VECTOR:ITEM1:UMAG 1.000

: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

: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?

: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|HEXa}
 :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

5.6 DISPlay グループ

: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します。

構文 :DISPlay:WAVE:MAPPING?

: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|AUX<x>}

機能 分割フォーマットへの { 電圧 | 電流 | 回転速度 | トルク | 外部信号 } 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|AUX<x>} {<Nrf>}
:DISPlay:WAVE:MAPPING:{U<x>|I<x>|SPEed|TORQue|AUX<x>}?

U<x>、I<x> の <x> = 1 ~ 6 (エLEMENT)
AUX<x> の <x> = 1 ~ 2 (AUX 入力チャンネル)
<Nrf> = 0 ~ 5

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、モータ評価機能 (オプション、/MTR) 搭載時のみ有効です。
- ・ AUX<x> は、外部信号入力 (オプション、/AUX) 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します。

構文 :DISPlay:WAVE:POSition?

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

機能 各エLEMENT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 ~ 6 (エLEMENT)
<Nrf> = - 130.000 ~ 130.000 (%)

例 :DISPlay:WAVE:POSition:U1 0
:DISPlay:WAVE:POSition:U1? ->
:DISPlay:WAVE:POSition:U1 0.000

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

機能 すべてのエLEMENT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:SVALue

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> = 0.05、0.1、0.2、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 5.00E-03

解説 設定することのできる Time/div 値は、データ更新レート (:RATE) の 1/10 までです。

:DISPlay:WAVE:TLabel

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?

: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 (%)

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|OFF}

:DISPlay:WAVE:TRIGger:MODE?<NRf> = - 100.0 ~ 100.0 (%)
例 :DISPlay:WAVE:TRIGger:MODE AUTO
:DISPlay:WAVE:TRIGger:MODE? ->
:DISPlay:WAVE:TRIGger:MODE AUTO

解説 波形表示を選択し、トリガモードを Auto/Normal に設定した場合は、データ更新周期はトリガの動作に依存します。

: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?<NRf> = - 100.0 ~ 100.0 (%)
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}<NRf> = - 100.0 ~ 100.0 (%)
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|AUX<x>}

機能 {電圧 | 電流 | 回転速度 | トルク | 外部信号} 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|AUX<x>} {<Boolean>}<NRf> = - 100.0 ~ 100.0 (%)
:DISPlay:WAVE:{U<x>|I<x>|SPEed|TORQue|AUX<x>}?<NRf> = - 100.0 ~ 100.0 (%)
U<x>、I<x> の <x> = 1 ~ 6 (エレメント)
AUX<x> の <x> = 1 ~ 2 (AUX 入力チャネル)

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

解説 • {SPEed|TORQue} は、モータ評価機能 (オプション、/MTR) 搭載時のみ有効です。
• AUX<x> は、外部信号入力 (オプション、/AUX) 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?

: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>}<NRf> = - 100.0 ~ 100.0 (%)
:DISPlay:WAVE:VZoom:{U<x>|I<x>}?<NRf> = - 100.0 ~ 100.0 (%)

<x> = 1 ~ 6 (エレメント)
例 :DISPlay:WAVE:VZoom:U1 1
:DISPlay:WAVE:VZoom:U1? ->
:DISPlay:WAVE:VZoom:U1 1.00

解説 ズーム率の選択肢については、ユーザーズマニュアル [機能編] IM WT1801-01JA を参照してください。

: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> = - 100.0 ~ 100.0 (%)

例 :DISPlay:WAVE:VZoom:UALL 1
解説 ズーム率の選択肢については、ユーザーズマニュアル [機能編] IM WT1801-01JA を参照してください。

※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

```

:AOUtput[:NORMal]:CHANnel<x> {NONE|<Function>[,<Element>][,<Order>]}
:DISPlay:NUMeric:CUSTom:ITEM<x>[:FUNctioN] {<Function>[,<Element>]
[,<Order>]|<文字列>}
:DISPlay:NUMeric[:NORMal]:{VAL4|VAL8|VAL16}:ITEM<x> {NONE|<Function>
[,<Element>][,<Order>]}
:DISPlay:NUMeric[:NORMal]:MATRix:ITEM<x> {NONE|<Function>[,<Element>]
[,<Order>]}
:DISPlay:NUMeric[:NORMal]:ALL:CURSor {<Function>}
:DISPlay:TREnd:ITEM<x>[:FUNctioN] {<Function>,<Element>[,<Order>]}
:FILE:SAVE:NUMeric:NORMal:<Function> {<Boolean>}
:MEASure:EVENT<x>:EXPRession:ITEM {<Function>[,<Element>][,<Order>]}
:NUMeric[:NORMal]:ITEM<x> {NONE|<Function>[,<Element>][,<Order>]}
:STORe:NUMeric[:NORMal]:<Function> {<Boolean>}

```

<Function>	メニュー上のファンクション表記 (数値表示ヘッダ名)	<Element>(エレメント)		<Order>(次数)
		○必要 ×不要	○必要 ×不要	
URMS	Urms	○	×	
UMN	Umn	○	×	
UDC	Udc	○	×	
URMN	Urmn	○	×	
UAC	Uac	○	×	
IRMS	lrms	○	×	
IMN	lmn	○	×	
IDC	ldc	○	×	
IRMN	lrnm	○	×	
IAC	lac	○	×	
P	P	○	×	
S	S	○	×	
Q	Q	○	×	
LAMBda	λ	○	×	
PHI	φ	○	×	
FU	FreqU(fU)	○	×	
FI	FreqI(fI)	○	×	
UPPeak	U+peak(U+pk)	○	×	
UMPeak	U-peak(U-pk)	○	×	
IPPeak	I+peak(I+pk)	○	×	
IMPeak	I-peak(I-pk)	○	×	
CFU	CfU	○	×	
CFI	CfI	○	×	
PC	Pc	○	×	
PPPeak	P+peak(P+pk)	○	×	
PMPeak	P-peak(P-pk)	○	×	
TIME	Time	○	×	
WH	WP	○	×	
WHP	WP+	○	×	
WHM	WP-	○	×	
AH	q	○	×	
AHP	q+	○	×	
AHM	q-	○	×	
WS	WS	○	×	
WQ	WQ	○	×	
ETA1 ~ ETA4	η 1 ~ η 4	×	×	
F1 ~ F20	F1 ~ F20	×	×	
EV1 ~ EV8	Event1 ~ Event8	×	×	

高調波測定機能 (オプション、/G5 または /G6) 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	λ (k)	○	○
PHIK	φ (k)	○	○
PHIUk	φ U(k)	○	○
PHIIk	φ I(k)	○	○
Zk	Z(k)	○	○
RSk	Rs(k)	○	○
XSk	Xs(k)	○	○
RPk	Rp(k)	○	○
XPk	Xp(k)	○	○
UHDFk	Uhdf(k)	○	○
IHDFk	Ihdf(k)	○	○
PHDFk	Phdf(k)	○	○
UTHD	Uthd	○	×
ITHD	Ithd	○	×
PTHD	Pthd	○	×
UTHF	Uthf	○	×
ITHF	Ithf	○	×
UTIF	Utif	○	×
ITIF	Itif	○	×
HVF	hvf	○	×
HCF	hcf	○	×
KFACtor	K-factor	○	×
PHI_U1U2	φ Ui-Uj	○	×
PHI_U1U3	φ Ui-Uk	○	×
PHI_U1I1	φ Ui-Ii	○	×
PHI_U2I2	φ Uj-Ij	○	×
PHI_U3I3	φ Uj-Ik	○	×
FPLL1	fPLL1	×	×
FPLL2	fPLL2	×	×

5.6 DISPlay グループ

デルタ演算機能 (オプション、/DT) が必要なファンクション			
DU1	Δ U1	○ (Σ のみ)	×
DU2	Δ U2	○ (Σ のみ)	×
DU3	Δ U3	○ (Σ のみ)	×
DUS	Δ U Σ	○ (Σ のみ)	×
DI	Δ I	○ (Σ のみ)	×
DP1	Δ P1	○ (Σ のみ)	×
DP2	Δ P2	○ (Σ のみ)	×
DP3	Δ P3	○ (Σ のみ)	×
DPS	Δ P Σ	○ (Σ のみ)	×
モータ評価機能 (オプション、/MTR) が必要なファンクション			
SPeEd	Speed	×	×
TORQue	Torque	×	×
SYNCsp	SyncSp	×	×
SLIP	Slip	×	×
PM	Pm	×	×
EAU	EaU	○	×
EAI	EaI	○	×
外部信号入力機能 (オプション、/AUX) が必要なファンクション			
AUX1	Aux1	×	×
AUX2	Aux2	×	×

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、「TOTaI」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 (高調波測定機能 (オプション) が必要)

対象コマンド

```
:DISPlay:BAR:ITEM<x>[:FUNCTION] {<Function>,<Element>}
:DISPlay:NUMeric[:NORMAL]:LIST:ITEM<x> {<Function>,<Element>}
:NUMeric:LIST:ITEM<x> {NONE|<Function>,<Element>}
```

<Function>	メニュー上のファンクション表記
U	U
I	I
P	P
S	S
Q	Q
LAMBda	λ
PHI	φ
PHIU	φ U
PHII	φ I
Z	Z
RS	Rs
XS	Xs
RP	Rp
XP	Xp
以下は、:NUMeric:LIST:ITEM<x> のみ有効	
UHDF	Uhdf
IHDF	Ihdf
PHDF	Phdf

5.7 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 キーと同じ設定・実行、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?

:FILE:CDIRectory

機能 ファイル操作の対象ディレクトリを変更します。

構文 :FILE:CDIRectory {<文字列>}

<文字列> = ディレクトリ名

例 :FILE:CDIRECTORY "TEST"

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:{BMP|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:
{BMP|PNG|JPEG} {<文字列>}

<文字列> = ファイル名

例 :FILE:DELETE:IMAGE:BMP "IMAGE1"

解説 ファイル名は、拡張子を付けずに指定してください。

:FILE:DELeTe:NUMeric:ASCii

機能 数値データファイルを削除します。

構文 :FILE:DELeTe:NUMeric:ASCii {<文字列>}

<文字列> = ファイル名

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 {<文字列>}

<文字列> = ファイル名

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:STORe:{DATA|HEADer}

機能 ストアされた数値データファイルを削除します。

構文 :FILE:DELeTe:STORe:{DATA|HEADer}
{<文字列>}

<文字列> = ファイル名

例 :FILE:DELETE:STORE:DATA "STR1"

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:ASCii

機能 波形表示データファイルを削除します。

構文 :FILE:DELeTe:WAVE:ASCii {<文字列>}

<文字列> = ファイル名

例 :FILE:DELETE:WAVE:ASCII "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 ファイル操作の対象ドライブを設定します。

構文 :FILE:DRIVE {RAM|USB[,<NRf>]|NETWork}

RAM = 内蔵 RAM ドライブ

USB = USB メモリドライブ、<NRf> = 0、1

(ドライブ番号)

NETWork = ネットワークドライブ

例 :FILE:DRIVE RAM

:FILE:FILTer

機能 ファイルリストのフィルタを設定 / 問い合わせします。

構文 :FILE:FILTer {ALL|ITEM}

:FILE:FILTer?

例 :FILE:FILTER ALL

:FILE:FILTER? ->

:FILE:FILTER ALL

: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? -> 20912128

: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 {<文字列>}

<文字列> = ファイル名

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:PATH?

機能 対象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? -> "USB-0/TEST"

: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?

:FILE:SAVE:ABORT

機能 ファイルの保存を中止します。

構文 :FILE:SAVE:ABORT

例 :FILE:SAVE:ABORT

:FILE:SAVE:ANAMing

機能 保存ファイル名の自動生成機能を設定 / 問い合わせします。

構文 :FILE:SAVE:ANAMing {OFF|NUMBERing|DATE}
:FILE:SAVE:ANAMing?

例 :FILE:SAVE:ANAMING NUMBERING

:FILE:SAVE:ANAMING? ->

:FILE:SAVE:ANAMING NUMBERING

:FILE:SAVE:COMMeNt

機能 保存するファイルに付加するコメントを設定 / 問い合わせします。

構文 :FILE:SAVE:COMMeNt {<文字列>}
:FILE:SAVE:COMMeNt?
<文字列> = 30 文字以内

例 :FILE:SAVE:COMMENT "CASE1"

:FILE:SAVE:COMMENT? ->

:FILE:SAVE:COMMENT "CASE1"

:FILE:SAVE:NUMERIC[: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]
{<文字列>}
<文字列> = ファイル名

例 :FILE:SAVE:NUMERIC:EXECUTE "NUM1"

解説

- ・ ファイル名は、拡張子を付けずに指定してください。
- ・ このコマンドはオーバーラップコマンドです。

:FILE:SAVE:NUMERIC:ITEM

機能 数値データをファイル保存するときの保存項目の選択方式を設定 / 問い合わせします。

構文 :FILE:SAVE:NUMERIC:ITEM {DISPlayed|SELEcted}
:FILE:SAVE:NUMERIC:ITEM?

DISPlayed = 画面表示項目自動選択方式

SELEcted = マニュアル選択方式

例 :FILE:SAVE:NUMERIC:ITEM SELECTED

:FILE:SAVE:NUMERIC:ITEM? ->

:FILE:SAVE:NUMERIC:ITEM SELECTED

解説

選択肢の意味は、次のとおりです。

DISPlayed = 画面に表示されている数値項目をファイル保存

SELEcted = 「:FILE:SAVE:NUMERIC:NORMAL:…」以下のコマンドで個別設定された数値項目をファイル保存

:FILE:SAVE:NUMERIC:NORMAL?

機能 数値データのファイル保存 (保存項目マニュアル選択方式) 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?

解説 保存項目の選択方式 (:FILE:SAVE:NUMERIC:ITEM) がマニュアル選択方式 (SELEcted) のときに有効な設定です。

:FILE:SAVE:NUMERIC:NORMAL:ALL

機能 数値データをファイル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|SIGMC}**

機能 数値データをファイル保存するときの各エレメント|結線ユニット{Σ A|Σ B|Σ C}の出力 ON/OFF を設定 / 問い合わせします。

構文 :FILE:SAVE:NUMERIC:NORMAL:
{ELEMENT<x>|SIGMA|SIGMB|SIGMC}
{<Boolean>}
:FILE:SAVE:NUMERIC:NORMAL:
{ELEMENT<x>|SIGMA|SIGMB|SIGMC}?
<x> = 1 ~ 6

例 :FILE:SAVE:NUMERIC:NORMAL:

ELEMENT1 ON

:FILE:SAVE:NUMERIC:NORMAL:

ELEMENT1? ->

:FILE:SAVE:NUMERIC:NORMAL:

ELEMENT1 1

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

機能 数値データをファイル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> = {URMS|IRMS|P|S|Q|...}

例 :FILE:SAVE:NUMERIC:NORMAL:URMS ON

:FILE:SAVE:NUMERIC:NORMAL:URMS? ->

:FILE:SAVE:NUMERIC:NORMAL:URMS 1

解説 <Function> の選択肢については、「DISPlay グループ」のファンクション選択肢一覧 (1)、5-38 ページを参照してください。

5.7 FILE グループ

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

機能 数値データをファイル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[機能編]IM WT1801-01JA を参照してください。

: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] {<文字列>}
<文字列> = ファイル名

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ラップコマンドです。

:FILE:SAVE:WAVE[:EXECute]

機能 波形表示データをファイルに保存します。

構文 :FILE:SAVE:WAVE[:EXECute] {<文字列>}
<文字列> = ファイル名

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ラップコマンドです。

5.8 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 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。

ただし、このグループのコマンドは、2 系統同時高調波測定 (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<x>?

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<x>?

<x> = 1 ~ 2 (高調波測定グループ)

解説 「HARMonics2」は、2 系統同時高調波測定機能 (オプション、/G6) 搭載時だけ有効です。

:HARMonics<x>:CONFigure?

機能 すべてのエレメントの高調波測定グループを問い合わせます。

構文 :HARMonics<x>:CONFigure?

解説

- 2 系統同時高調波測定機能 (オプション、/G6) 搭載時だけ有効です。
- HARMonics<x> の <x> は、問い合わせに関係ありません。

:HARMonics<x>:CONFigure[:ALL]

機能 すべてのエレメントの高調波測定グループを一括設定します。

構文 :HARMonics<x>:CONFigure[:ALL]
{<NRf>}

<NRf> = 1 (Hrm1)、2 (Hrm2)

例 :HARMONICS:CONFIGURE:ALL 1

解説

- 2 系統同時高調波測定機能 (オプション、/G6) 搭載時だけ有効です。
- HARMonics<x> の <x> は、設定に関係ありません。

:HARMonics<x>:CONFigure:ELEMeNt<x>

機能 各エレメントの高調波測定グループを設定 / 問い合わせします。

構文 :HARMonics<x>:CONFigure:
ELEMeNt<x> {<NRf>}
:HARMonics<x>:CONFigure:ELEMeNt<x>?
ELEMeNt<x> の <x> = 1 ~ 6 (エレメント)
<NRf> = 1 (Hrm1)、2 (Hrm2)

例 :HARMONICS:CONFIGURE:ELEMENT1 1
:HARMONICS:CONFIGURE:ELEMENT1? ->
:HARMONICS1:CONFIGURE:ELEMENT1 1

解説

- 2 系統同時高調波測定機能 (オプション、/G6) 搭載時だけ有効です。
- HARMonics<x> の <x> は、設定 / 問い合わせに関係ありません。

:HARMonics<x>:CONFigure: {SIGMA|SIGMB|SIGMC}

機能 結線ユニット {Σ A|Σ B|Σ C} に属するエレメントの高調波測定グループを一括設定します。

構文 :HARMonics<x>:CONFigure:
{SIGMA|SIGMB|SIGMC} {<NRf>}
<NRf> = 1 (Hrm1)、2 (Hrm2)

例 :HARMONICS:CONFIGURE:SIGMA 1

解説

- 2 系統同時高調波測定機能 (オプション、/G6) 搭載時だけ有効です。
- HARMonics<x> の <x> は、設定に関係ありません。

:HARMonics<x>:ORDeR

機能 最小 / 最大解析次数を設定 / 問い合わせします。

構文 :HARMonics<x>:ORDeR {<NRf>,<NRf>}
:HARMonics<x>:ORDeR?
<x> = 1 ~ 2 (高調波測定グループ)

1 つ目の <NRf> = 0、1 (最小解析次数)

2 つ目の <NRf> = 1 ~ 500 (最大解析次数)

例 :HARMONICS:ORDER 1,100

:HARMONICS:ORDER? ->

:HARMONICS1:ORDER 1,100

:HARMonics<x>:PLLSource

機能 PLL ソースを設定 / 問い合わせします。

構文 :HARMonics<x>:PLLSource {U<x>|I<x>|
EXTeRnal}
:HARMonics<x>:PLLSource?
HARMonics<x> の <x> = 1 ~ 2 (高調波測定グループ)

U<x>、I<x> の <x> = 1 ~ 6 (エレメント)

EXTeRnal = 外部クロック入力 (Ext Clk)

例 :HARMONICS:PLLSOURCE U1

:HARMONICS:PLLSOURCE? ->

:HARMONICS1:PLLSOURCE U1

:HARMonics<x>: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<x>:THD {TOTAl|
FUNDamental}
:HARMonics<x>:THD?
<x> = 1 ~ 2 (高調波測定グループ)

例 :HARMONICS:THD TOTAL

:HARMONICS:THD? ->

:HARMONICS1:THD TOTAL

5.9 HCOPY グループ

HCOPY グループは、内蔵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) 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。

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) 搭載時だけ有効です。

:HCOPY?

機能 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?

:HCOPY:ABORT

機能 プリントを中断します。

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?

:HCOPY:AUTO:COUNT

機能 オートプリントの印刷回数を設定 / 問い合わせします。

構文 :HCOPY:AUTO:COUNT {<NRf>|INFINITE}

:HCOPY:AUTO:COUNT?

<NRf> = 1 ~ 9999

INFINITE = 限度なし

例 :HCOPY:AUTO:COUNT INFINITE

:HCOPY:AUTO:COUNT? ->

:HCOPY:AUTO:COUNT INFINITE

解説 オートプリントの動作方式 (:HCOPY:AUTO:MODE) が {INTERVAL|RTIME|INTEGRATE} のときに有効な設定です。

:HCOPY:AUTO:INTERVAL

機能 オートプリントの印刷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

解説 オートプリントの動作方式 (:HCOPY:AUTO:MODE) が {INTERVAL|RTIME|INTEGRATE} のときに有効な設定です。

:HCOPY:AUTO:MODE

機能 オートプリントの動作方式を設定 / 問い合わせします。

構文 :HCOPY:AUTO:MODE {INTERVAL|RTIME|

INTEGRATE|EVENT}

:HCOPY:AUTO:MODE?

INTERVAL = インタバル印刷モード

RTIME = 実時間制御印刷モード

INTEGRATE = 積算同期印刷モード

EVENT = イベント同期印刷モード

例 :HCOPY:AUTO:MODE INTERVAL

:HCOPY:AUTO:MODE? ->

:HCOPY:AUTO:MODE INTERVAL

:HCOPY:AUTO:PASTART

機能 オートプリント開始時に印刷するかどうかを設定 / 問い合わせします。

構文 :HCOPY:AUTO:PASTART {<Boolean>}

:HCOPY:AUTO:PASTART?

例 :HCOPY:AUTO:PASTART OFF

:HCOPY:AUTO:PASTART? ->

:HCOPY:AUTO:PASTART 0

解説 オートプリントの動作方式 (:HCOPY:AUTO:MODE) が {INTERVAL|RTIME|INTEGRATE} のときに有効な設定です。

:HCOPY:AUTO:{START|END}

機能 実時間制御印刷モードの印刷 {開始 | 終了} 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>}

<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 2011,1,1,0,0,0

:HCOPY:AUTO:START? ->

:HCOPY:AUTO:START 2011,1,1,0,0,0

解説 オートプリントの動作方式 (:HCOPY:AUTO:MODE) が RTIME のときに有効な設定です。

5.9 HCOpy グループ

: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 OFF
:HCOpy:AUTO:STATe? ->
:HCOpy:AUTO:STATe 0

:HCOpy:AUTO:TEVent

機能 イベント同期印刷モードのトリガとなるイベントを設定 / 問い合わせします。

構文 :HCOpy:AUTO:TEVent {<NRf>}
:HCOpy:AUTO:TEVent?
<NRf> = 1 ~ 8 (イベント番号)

例 :HCOpy:AUTO:TEVENT 1
:HCOpy:AUTO:TEVENT? ->
:HCOpy:AUTO:TEVENT 1

解説 オートプリントの動作方式 (:HCOpy:AUTO:MODE) が EVENT のときに有効な設定です。

:HCOpy:COMMeNt

機能 画面下部に表示するコメントを設定 / 問い合わせします。

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

<文字列> = 30 文字以内
例 :HCOpy:COMMENT "THIS IS TEST."
:HCOpy:COMMENT? ->
:HCOpy:COMMENT "THIS IS TEST."

:HCOpy:EXECute

機能 プリントを実行します。

構文 :HCOpy:EXECute

例 :HCOpy:EXECUTE

解説 オーバラップコマンドです。

: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?

: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 オーバ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 {SCREEN|LIST}
:HCOpy:PRINter:FORMat?

SCREEN = 画面イメージ

LIST = 数値データリスト

例 :HCOpy:PRINter:FORMAT SCREEN
:HCOpy:PRINter:FORMAT? ->
:HCOpy:PRINter SCREEN

5.10 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 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。

: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

5.11 HSPeet グループ

HSPeet グループは、高速データ収集に関するグループです。

フロントパネルの ITEM の「HS Items」メニューおよび FORM の「HS Settings」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。

このグループのコマンドは、高速データ収集機能 (オプション、/HS) 搭載時のみ有効です。

:HSPeet?

機能 高速データ収集機能に関するすべての設定値を問い合わせます。

構文 :HSPeet?

解説 このグループのコマンドは、高速データ収集機能 (オプション、/HS) 搭載時のみ有効です。

:HSPeet:CAPTured?

機能 高速データ収集の収集済み回数を問い合わせます。

構文 :HSPeet:CAPTured?

例 :HSPEED:CAPTURED? -> 200

解説 収集実行中に画面上部のステータス画面の「Captured」に表示されている数値を返します。

:HSPeet:COUNT

機能 データ収集回数を設定 / 問い合わせします。

構文 :HSPeet:COUNT {<NRf>|INFinite}

:HSPeet:COUNT?

<NRf> = 1 ~ 10000000

INFinite = 限度なし

例 :HSPEED:COUNT INFINITE

:HSPEED:COUNT? ->

:HSPEED:COUNT INFINITE

:HSPeet:DISPlay?

機能 高速データ収集モードの表示に関するすべての設定値を問い合わせます。

構文 :HSPeet:DISPlay?

解説 「:HSPeet:DISPlay:...」は「:DISPlay:HSPeet:...」と共通の設定 / 問い合わせとなります。

:HSPeet:DISPlay:COLumn?

機能 高速データ収集モードのカラムに関するすべての設定値を問い合わせます。

構文 :HSPeet:DISPlay:COLumn?

:HSPeet:DISPlay:COLumn:ITEM<x>

機能 高速データ収集モードのカラム表示項目を設定 / 問い合わせします。

構文 :HSPeet:DISPlay:COLumn:ITEM<x>

{NONE|<Element>}

:HSPeet:DISPlay:COLumn:ITEM<x>?

<x> = 1 ~ 6 (カラム番号)

<Element> = {<NRf>|SIGMa|SIGMB|SIGMC}

(<NRf> = 1 ~ 6)

例 :HSPEED:DISPLAY:COLUMN:ITEM1 1

:HSPEED:DISPLAY:COLUMN:ITEM1? ->

:HSPEED:DISPLAY:COLUMN:ITEM1 1

:HSPeet:DISPlay:COLumn:NUMBER

機能 高速データ収集モードの表示カラム数を設定 / 問い合わせします。

構文 :HSPeet:DISPlay:COLumn:NUMBER

{<NRf>}

:HSPeet:DISPlay:COLumn:NUMBER?

<NRf> = 4、6

例 :HSPEED:DISPLAY:COLUMN:NUMBER 4

:HSPEED:DISPLAY:COLUMN:NUMBER? ->

:HSPEED:DISPLAY:COLUMN:NUMBER 4

:HSPeet:DISPlay:COLumn:RESet

機能 高速データ収集モードのカラム表示項目を初期値にリセットします。

構文 :HSPeet:DISPlay:COLumn:RESet

例 :HSPEED:DISPLAY:COLUMN:RESET

:HSPeet:DISPlay:FRAMe

機能 高速データ収集モードのデータ部のフレーム表示 ON/OFF を設定 / 問い合わせします。

構文 :HSPeet:DISPlay:FRAMe {<Boolean>}

:HSPeet:DISPlay:FRAMe?

例 :HSPEED:DISPLAY:FRAME ON

:HSPEED:DISPLAY:FRAME? ->

:HSPEED:DISPLAY:FRAME 1

解説 「:DISPlay:NUMeric:FRAME」と共通の設定です。

5.11 HSPeet グループ

:HSPeet:DISPlay:PAGE

機能 高速データ収集モードの表示ページを設定 / 問い合わせします。

構文 :HSPeet:DISPlay:PAGE{<NRf>}
:HSPeet:DISPlay:PAGE?
<NRf> = 1 ~ 2 (ページ番号)
<NRf> = 1 ~ 4 (モータ評価機能 (オプション、/MTR) または外部信号入力 (オプション、/AUX) 搭載時)

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

:HSPeet:DISPlay:POVer

機能 高速データ収集モードのピークオーバー発生情報の表示 ON/OFF を設定 / 問い合わせします。

構文 :HSPeet:DISPlay:POVer {<Boolean>}
:HSPeet:DISPlay:POVer?

例 :HSPeet:DISPlay:POVer OFF
:HSPeet:DISPlay:POVer? ->
:HSPeet:DISPlay:POVer 0

:HSPeet:EXTSync

機能 高速データ収集の外部信号同期の ON/OFF を設定 / 問い合わせします。

構文 :HSPeet:EXTSync {<Boolean>}
:HSPeet:EXTSync?

例 :HSPeet:EXTSync OFF
:HSPeet:EXTSync ? ->
:HSPeet:EXTSync 0

:HSPeet:FILTer?

機能 高速データ収集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 :HSPeet:FILTer?

:HSPeet:FILTer[:HS]

機能 高速データ収集のデジタルフィルタ (HS Filter) を設定 / 問い合わせします。

構文 :HSPeet:FILTer[:HS] {OFF|<周波数>}
:HSPeet:FILTer:HS?
OFF = デジタルフィルタ OFF
<周波数> = 1 ~ 1000 (Hz) (デジタルフィルタ ON、カットオフ周波数)

例 :HSPeet:FILTer:HS 100HZ
:HSPeet:FILTer:HS? ->
:HSPeet:FILTer:HS 100.0E+00

:HSPeet:FILTer:LINE?

機能 高速データ収集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 :HSPeet:FILTer:LINE?

:HSPeet:FILTer:LINE[:ALL]

機能 高速データ収集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 :HSPeet:FILTer:LINE[:ALL] {<周波数>}
<周波数> = 0.1 ~ 100.0 (kHz)、300kHz
(カットオフ周波数)

例 :HSPeet:FILTer:LINE:ALL 300KHZ
解説
・高速データ収集モードでは、ラインフィルタは常に ON となります。
・ここではカットオフ周波数を設定します。0.1 ~ 100.0(kHz) の間は、0.1(kHz) の分解能で設定できます。

:HSPeet:FILTer:LINE:ELEment<x>

機能 高速データ収集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 :HSPeet:FILTer:LINE:
ELEment<x> {<周波数>}
:HSPeet:FILTer:LINE:ELEment<x>?
<x> = 1 ~ 6 (エレメント)
<周波数> = 0.1 ~ 100.0 (kHz)、300kHz
(カットオフ周波数)

例 :HSPeet:FILTer:LINE:ELEment1 300KHZ
:HSPeet:FILTer:LINE:ELEment1? ->
:HSPeet:FILTer:LINE:
ELEment1 300.000E+03
解説
・高速データ収集モードでは、ラインフィルタは常に ON となります。
・ここではカットオフ周波数を設定します。0.1 ~ 100.0(kHz) の間は、0.1(kHz) の分解能で設定できます。

:HSPeet:MAXCount?

機能 データ収集回数の最大値を問い合わせます。

構文 :HSPeet:MAXCount?

例 :HSPeet:MAXCOUNT? -> 506811

:HSPeet:MEASuring?

機能 高速データ収集の電圧 / 電流モードに関するすべての設定値を問い合わせます。

構文 :HSPeet:MEASuring?

:HSPeet:MEASuring[:ALL]

機能 すべての電圧 / 電流モードを一括設定します。

構文 :HSPeet:MEASuring[:ALL] {RMS|MEAN|
DC|RMEAN}

例 :HSPeet:MEASURING:ALL RMS

:HSPeet:MEASuring:{U<x>|I<x>}

機能 { 電圧 | 電流 } モードを設定 / 問い合わせします。

構文 :HSPeet:MEASuring:{U<x>|I<x>} {RMS|
MEAN|DC|RMEAN}
:HSPeet:MEASuring:{U<x>|I<x>}?
<x> = 1 ~ 6 (エレメント)

例 :HSPeet:MEASURING:U1 RMS
:HSPeet:MEASURING:U1? ->
:HSPeet:MEASURING:U1 RMS

:HSPeet:MEASuring:{UALL|IALL}

機能 すべての { 電圧 | 電流 } モードを一括設定します。

構文 :HSPeet:MEASuring:{UALL|IALL} {RMS|
MEAN|DC|RMEAN}

例 :HSPeet:MEASURING:UALL RMS

:HSPeet:POVer?

機能 高速データ収集のピークオーバー発生情報を問い合わせます。

構文 :HSPeet:POVer?

例 :HSPeet:POVer? -> 0

解説

- ・収集実行中のピークオーバー発生情報 (画面中
段の「Peak Over Status」) を返します。
- ・各エレメントのピークオーバー発生情報を下記
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
--	--	Tq	Sp	I6	U6	I5	U5	I4	U4	I3	U3	I2	U2	I1	U1

Sp: 回転速度、または AUX1

Tq: トルク、または AUX2

:HSPeet:RECORD?

機能 高速データ収集のファイル保存に関するすべての
の設定値を問い合わせます。

構文 :HSPeet:RECORD?

:HSPeet:RECORD:FILE?

機能 収集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 :HSPeet:RECORD:FILE?

解説 「:HSPeet:RECORD:FILE:...」は「:STORE:FILE:...」
と共通の設定 / 操作となります。
(「:STORE:FILE:CONVERT:...」を除く)

:HSPeet:RECORD:FILE:ANAMing

機能 収集した数値データの保存ファイル名の自動生
成機能を設定 / 問い合わせします。

構文 :HSPeet:RECORD:FILE:ANAMing {OFF|
NUMBERing|DATE}
:HSPeet:RECORD:FILE:ANAMing?

例 :HSPeet:RECORD:FILE
:ANAMING NUMBERING
:HSPeet:RECORD:FILE:ANAMING? ->
:HSPeet:RECORD:FILE
:ANAMING NUMBERING

:HSPeet:RECORD:FILE:CDIRECTory

機能 収集した数値データの保存先ディレクトリを変
更します。

構文 :HSPeet:RECORD:FILE
:CDIRECTory{<文字列>}
<文字列> = ディレクトリ名

例 :HSPeet:RECORD:FILE
:CDIRECTORY "RECORD"

解説 上のディレクトリに移動するには、"." を指定し
ます。

:HSPeet:RECORD:FILE:CONVERT?

機能 収集した数値データファイルの CSV 形式への変
換に関するすべての設定値を問い合わせます。

構文 :HSPeet:RECORD:FILE:CONVERT?

:HSPeet:RECORD:FILE:CONVERT:ABORT

機能 収集した数値データファイルの CSV 形式への変
換を中止します。

構文 :HSPeet:RECORD:FILE:CONVERT:ABORT

例 :HSPeet:RECORD:FILE:CONVERT:ABORT

:HSPeet:RECORD:FILE:CONVERT:AUTO

機能 収集した数値データファイルの CSV 形式への自
動変換 ON/OFF を設定 / 問い合わせします。

構文 :HSPeet:RECORD:FILE:CONVERT:
AUTO {<Boolean>}

:HSPeet:RECORD:FILE:CONVERT:AUTO?

例 :HSPeet:RECORD:FILE:CONVERT:AUTO ON
:HSPeet:RECORD:FILE:CONVERT:AUTO? ->
:HSPeet:RECORD:FILE:CONVERT:AUTO 1

:HSPeet:RECORD:FILE:CONVERT:EXECute

機能 収集した数値データファイルの CSV 形式への変
換を実行します。

構文 :HSPeet:RECORD:FILE:CONVERT:
EXECute {<文字列>}

<文字列> = ファイル名

例 :HSPeet:RECORD:FILE:CONVERT:
EXECUTE "RECORD1"

解説

- ・ファイル名は、括弧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。

5.11 HSPeEd グループ

:HSPeEd:RECORD:FILE:DRIVE

機能 収集した数値データの保存先ドライブを設定します。

構文 :HSPeEd:RECORD:FILE:DRIVE{RAM|
USB[, <NRf>]|NETWork}
RAM = 内蔵 RAM ドライブ
USB = USB メモリドライブ、
<NRf> = 0、1 (ドライブ番号)
NETWork = ネットワークドライブ

例 :HSPEED:RECORD:FILE:DRIVE RAM

:HSPeEd:RECORD:FILE:FREE?

機能 収集した数値データの保存先ドライブの空き容量 (byte) を問い合わせます。

構文 :HSPeEd:RECORD:FILE:FREE?

例 :HSPEED:RECORD:FILE:FREE? -> 20912128

:HSPeEd:RECORD:FILE:NAME

機能 収集した数値データの保存ファイル名を設定 / 問い合わせします。

構文 :HSPeEd:RECORD:FILE:NAME {<文字列>}
:HSPeEd:RECORD:FILE:NAME?
<文字列> = ファイル名

例 :HSPEED:RECORD:FILE:NAME "RECORD1"
:HSPEED:RECORD:FILE:NAME? ->
:HSPEED:RECORD:FILE:NAME "RECORD1"

:HSPeEd:RECORD:FILE:PATH?

機能 収集した数値データの保存先を絶対パスで問い合わせます。

構文 :HSPeEd:RECORD:FILE:PATH?

例 :HSPEED:RECORD:FILE:PATH? ->
"USB-0/RECORD"

:HSPeEd:RECORD:FILE:STATE?

機能 収集した数値データのファイル保存の処理ステータスを問い合わせます。

構文 :HSPeEd:RECORD:FILE:STATE?

例 :HSPEED:RECORD:FILE:STATE? -> READY

解説 応答の内容は次のとおりです。

- READY = ファイルクローズ (収集待機中、または収集・ファイル保存完了)
- RECOrd = ファイルオープン (収集実行中)
- STOP = ファイルクローズ (収集実行中だが、エラー発生によるファイル保存終了)
- CONVert = CSV 形式ファイルへの変換中
- OFF = ファイル保存しないとき、または高速データ収集モード以外のとき

:HSPeEd:RECORD:ITEM?

機能 ファイル保存する数値データ項目に関するすべての設定値を問い合わせます。

構文 :HSPeEd:RECORD:ITEM?

:HSPeEd:RECORD:ITEM:AUX<x>

機能 数値データ (外部信号入力) の保存する / しないを設定 / 問い合わせします。

構文 :HSPeEd:RECORD:ITEM:
AUX<x> {<Boolean>}
:HSPeEd:RECORD:ITEM:AUX<x>?
<x> = 1 ~ 2 (AUX 入力チャネル)

例 :HSPEED:RECORD:ITEM:AUX1 ON
:HSPEED:RECORD:ITEM:AUX1? ->
:HSPEED:RECORD:ITEM:AUX1 1

解説 外部信号入力 (オプション、/AUX) 搭載時のみ有効です。

:HSPeEd:RECORD:ITEM:{I<x>|IA|IB|IC}

機能 数値データ (電流) の各エレメント | 結線ユニット {Σ A|Σ B|Σ C} の保存する / しないを設定 / 問い合わせします。

構文 :HSPeEd:RECORD:ITEM:{I<x>|IA|IB|
IC} {<Boolean>}
:HSPeEd:RECORD:ITEM:{I<x>|IA|IB|IC}?
<x> = 1 ~ 6 (エレメント)

IA、IB、IC = IΣA、IΣB、IΣC
例 :HSPEED:RECORD:ITEM:I1 ON
:HSPEED:RECORD:ITEM:I1? ->
:HSPEED:RECORD:ITEM:I1 1

:HSPeEd:RECORD:ITEM:{P<x>|PA|PB|PC}

機能 数値データ (有効電力) の各エレメント | 結線ユニット {Σ A|Σ B|Σ C} の保存する / しないを設定 / 問い合わせします。

構文 :HSPeEd:RECORD:ITEM:{P<x>|PA|PB|
PC} {<Boolean>}
:HSPeEd:RECORD:ITEM:{P<x>|PA|PB|PC}?
<x> = 1 ~ 6 (エレメント)

PA、PB、PC = PΣA、PΣB、PΣC
例 :HSPEED:RECORD:ITEM:P1 ON
:HSPEED:RECORD:ITEM:P1? ->
:HSPEED:RECORD:ITEM:P1 1

:HSPeEd:RECORD:ITEM:{SPEEd|TORQue|PM}

機能 数値データ (モータ) の {回転速度 | トルク | モータ出力} の保存する / しないを設定 / 問い合わせします。

構文 :HSPeEd:RECORD:ITEM:{SPEEd|TORQue|
PM} {<Boolean>}
:HSPeEd:RECORD:ITEM:{SPEEd|TORQue|
PM}?

例 :HSPEED:RECORD:ITEM:SPEED ON
:HSPEED:RECORD:ITEM:SPEED? ->
:HSPEED:RECORD:ITEM:SPEED 1

解説 モータ評価機能 (オプション、/MTR) 搭載時のみ有効です。

:HSPeEd:RECOrd:ITEM:{U<x>|UA|UB|UC}

機能 数値データ（電圧）の各エレメント | 結線ユニット {Σ A | Σ B | Σ C} の保存する / しないを設定 / 問い合わせします。

構文 :HSPeEd:RECOrd:ITEM:{U<x>|UA|UB|UC} {<Boolean>}
 :HSPeEd:RECOrd:ITEM:{U<x>|UA|UB|UC}?
 <x> = 1 ~ 6 (エレメント)
 UA、UB、UC = U_{EA}、U_{EB}、U_{EC}

例 :HSPeEd:RECOrd:ITEM:U1 ON
 :HSPeEd:RECOrd:ITEM:U1? ->
 :HSPeEd:RECOrd:ITEM:U1 1

:HSPeEd:RECOrd:ITEM:PRESet:ALL

機能 数値データのすべての項目の保存する / しないを一括設定します。

構文 :HSPeEd:RECOrd:ITEM:PRESet:
 ALL {<Boolean>}

例 :HSPeEd:RECOrd:ITEM:PRESet:ALL ON

:HSPeEd:RECOrd:ITEM:PRESet:{ELEMENT<x>|SIGMA|SIGMB|SIGMC}

機能 数値データの各エレメント | 結線ユニット {Σ A | Σ B | Σ C} の保存する / しないを一括設定します。

構文 :HSPeEd:RECOrd:ITEM:PRESet:
 {ELEMENT<x>|SIGMA|SIGMB|SIGMC} {<Boolean>}
 <x> = 1 ~ 6

例 :HSPeEd:RECOrd:ITEM:PRESet:
 ELEMENT1 ON

:HSPeEd:RECOrd:ITEM:PRESet:{U|I|P|MOTor|AUX}

機能 数値データの各ファンクションの保存する / しないを一括設定します。

構文 :HSPeEd:RECOrd:ITEM:PRESet:{U|I|P|
 MOTor|AUX} {<Boolean>}

例 :HSPeEd:RECOrd:ITEM:PRESet:U ON

解説

- ・MOTor は、モータ評価機能 (オプション、/MTR) 搭載時のみ有効です。
- ・AUX は、外部信号入力 (オプション、/AUX) 搭載時のみ有効です。

:HSPeEd:RECOrd[:STATE]

機能 収集した数値データをファイルに保存する / しないを設定 / 問い合わせします。

構文 :HSPeEd:RECOrd[:STATE] {<Boolean>}
 :HSPeEd:RECOrd:STATE?

例 :HSPeEd:RECOrd:STATE ON
 :HSPeEd:RECOrd:STATE? ->
 :HSPeEd:RECOrd:STATE 1

:HSPeEd:START

機能 データ収集を開始します。

構文 :HSPeEd:START
 例 :HSPeEd:START

:HSPeEd:STATE?

機能 高速データ収集のステータスを問い合わせます。

構文 :HSPeEd:STATE?
 例 :HSPeEd:STATE? -> READY
 解説

- ・応答の内容は次のとおりです。
 INIT = 初期化中
 READy = 収集待機中
 START = 収集実行中
 OFF = 高速データ収集モード以外するとき

:HSPeEd:STOP

機能 データ収集を終了します。

構文 :HSPeEd:STOP
 例 :HSPeEd:STOP

:HSPeEd:TRIGger?

機能 高速データ収集のトリガに関するすべての設定値を問い合わせます。

構文 :HSPeEd:TRIGger?

:HSPeEd:TRIGger:LEVel

機能 トリガレベルを設定 / 問い合わせします。

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

例 :HSPeEd:TRIGger:LEVel 0
 :HSPeEd:TRIGger:LEVel? ->
 :HSPeEd:TRIGger:LEVel 0.0

解説 「:DISPlay:WAVE:TRIGger:LEVel」と共通の設定です。

:HSPeEd:TRIGger:MODE

機能 トリガモードを設定 / 問い合わせします。

構文 :HSPeEd:TRIGger:MODE {AUTO|NORMal|
 OFF}
 :HSPeEd:TRIGger:MODE?

例 :HSPeEd:TRIGger:MODE AUTO
 :HSPeEd:TRIGger:MODE? ->
 :HSPeEd:TRIGger:MODE AUTO

解説 「:DISPlay:WAVE:TRIGger:MODE」と共通の設定です。

5.11 HSPeEd グループ

:HSPeEd:TRIGger:SLOPe

機能	トリガスロープを設定 / 問い合わせします。
構文	:HSPeEd:TRIGger:SLOPe {RISE FALL BOTH} :HSPeEd:TRIGger:SLOPe?
例	:HSPEED:TRIGGER:SLOPE RISE :HSPEED:TRIGGER:SLOPE? -> :HSPEED:TRIGGER:SLOPE RISE
解説	「:DISPlay:WAVE:TRIGger:SLOPe」と共通の設定です。

:HSPeEd:TRIGger:SOURce

機能	トリガソースを設定 / 問い合わせします。
構文	:HSPeEd:TRIGger:SOURce {U<x> I<x> EXtErnal} :HSPeEd:TRIGger:SOURce? <x> = 1 ~ 6 (エレメント) EXtErnal = 外部トリガ入力 (Ext Clk)
例	:HSPEED:TRIGGER:SOURCE U1 :HSPEED:TRIGGER:SOURCE? -> :HSPEED:TRIGGER:SOURCE U1
解説	「:DISPlay:WAVE:TRIGger:SOURce」と共通の設定です。

5.12 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、および MENU(SHIFT+IMAGE SAVE) キーと同じ設定・実行、および設定内容の変更の問い合わせができます。

: IMAGE?

機能 画面イメージデータの出力に関するすべての設定値を問い合わせます。

構文 : IMAGE?

: IMAGE: ABORT

機能 画面イメージデータの出力を中止します。

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

: 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> = 30 文字以内

例 : IMAGE: COMMENT "THIS IS TEST."

: IMAGE: COMMENT? ->

: IMAGE: COMMENT "THIS IS TEST."

: IMAGE: EXECute

機能 画面イメージデータの出力を実行します。

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 {BMP|PNG|JPEG}

: IMAGE: FORMat?

例 : IMAGE: FORMAT BMP

: IMAGE: FORMAT? ->

: IMAGE: FORMAT BMP

: 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?

: IMAGE: SAVE: ANAMing

機能 保存ファイル名の自動生成機能を設定 / 問い合わせします。

構文 : IMAGE: SAVE: ANAMing {OFF|NUMBer|DATE}

: IMAGE: SAVE: ANAMing?

例 : IMAGE: SAVE: ANAMING NUMBERING

: IMAGE: SAVE: ANAMING? ->

: IMAGE: SAVE: ANAMING NUMBERING

: 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: SAVE: CDIRectory {<文字列>}

<文字列> = ディレクトリ名

例 : IMAGE: SAVE: 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 {RAM|

USB[, <Nrf>] | NETWork}

RAM = 内蔵 RAM ドライブ

USB = USB メモリドライブ、

<Nrf> = 0、1 (ドライブ番号)

NETWork = ネットワークドライブ

例 : IMAGE: SAVE: DRIVE RAM

: IMAGE: SAVE: FREE?

機能 画面イメージデータの保存先ドライブの空き容量 (byte) を問い合わせます。

構文 : IMAGE: SAVE: FREE?

例 : IMAGE: SAVE: FREE? -> 20912128

5.12 IMAGE グループ

: IMAGE : SAVE : NAME

機能	保存ファイル名を設定 / 問い合わせします。
構文	: IMAGE : SAVE : NAME { < 文字列 > } : IMAGE : SAVE : NAME ? < 文字列 > = ファイル名
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 : CDiractory」コマンドでそれぞれ設定します。 - 保存先のパスは「: IMAGE : SAVE : PATH ?」で問い合わせできます。 - ファイル名は、拡張子を付けずに指定してください。

: IMAGE : SAVE : PATH ?

機能	画面イメージデータの保存先を絶対パスで問い合わせます。
構文	: IMAGE : SAVE : PATH ?
例	: IMAGE : SAVE : PATH ? -> "USB-0 / IMAGE"

: 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 ? -> #N (N 桁のバイト数) (データバイトの並び)
解説	データバイト数の桁数 N は、出力するデータサイズにより異なります。

5.13 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、NULL、および NULL SET (SHIFT+NULL) 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。

: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?

[:INPut] :CFACtor

機能 クレストファクタを設定 / 問い合わせます。

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?

[: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?

[: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 ~ 6 (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|SIGMC}

機能 結線ユニット {Σ A|Σ B|Σ C} に属する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|SIGMC} { <Boolean> }

例 :INPUT:CURRENT:AUTO:SIGMA ON

解説 結線方式の設定 ([:INPut] :WIRing) により結線ユニット {Σ A|Σ B|Σ C} が存在しないとき、{SIGMA|SIGMB|SIGMC} は無効です。

[:INPut] :CURRent:CONFIg?

機能 すべてのエレメントの有効な電流レンジを問い合わせます。

構文 [:INPut] :CURRent:CONFIg?

[:INPut] :CURRent:CONFIg [:ALL]

機能 すべてのエレメントの有効な電流レンジを一括設定します。

構文 [:INPut] :CURRent:CONFIg [:ALL] { ALL | <電流> [, <電流>] [, <電流>] ... }

ALL = 全レンジ有効

・ 50A 入力エレメントのとき

・ クレストファクタを「3」に設定しているとき
<電流> = 1、2、5、10、20、50(A)

・ クレストファクタを「6」に設定しているとき
<電流> = 500(mA)、1、2.5、5、10、25(A)

・ 5A 入力エレメントのとき

・ クレストファクタを「3」に設定しているとき
<電流> = 10、20、50、100、200、500(mA)、1、2、5(A)

・ クレストファクタを「6」に設定しているとき
<電流> = 5、10、25、50、100、250、500(mA)、1、2.5(A)

例 :INPUT:CURRENT:CONFIG:ALL ALL

:INPUT:CURRENT:CONFIG:ALL 50,10,5,1

解説 パラメータには、有効にする電流レンジをすべて並べて記述します。全レンジ有効にする場合は、パラメータ「ALL」を指定します。

5.13 INPut グループ

[: INPut] : CURRent : CONFig : ELEMeNt < x >	
機能	各エレメントの有効な電流レンジを設定 / 問い合わせします。
構文	<pre>[: INPut] : CURRent : CONFig : ELEMeNt < x > { ALL < 電流 > [, < 電流 >] [, < 電流 >] ... } [: INPut] : CURRent : CONFig : ELEMeNt < x > ? < x > = 1 ~ 6 (エレメント) ALL = 全レンジ有効 < 電流 > = ([: INPut] : CURRent : CONFig [: ALL]) を参照</pre>
例	<pre>: INPUT : CURRENT : CONFIG : ELEMENT1 ALL : INPUT : CURRENT : CONFIG : ELEMENT1 ? -> : INPUT : CURRENT : CONFIG : ELEMENT1 ALL : INPUT : CURRENT : CONFIG : ELEMENT1 50,10,5,1 : INPUT : CURRENT : CONFIG : ELEMENT1 ? -> : INPUT : CURRENT : CONFIG : ELEMENT1 50.0E+00,10.0E+00,5.0E+00,1.0E+00</pre>
解説	パラメータには、有効にする電流レンジをすべて並べて記述します。全レンジ有効にする場合は、パラメータ「ALL」を指定します。

[: INPut] : CURRent : EXTSeNsor ?	
機能	外部電流センサレンジに関するすべての設定値を問い合わせます。
構文	<pre>[: INPut] : CURRent : EXTSeNsor ?</pre>
解説	外部電流センサ入力 (オプション、/EX) 搭載時のみ有効です。

[: INPut] : CURRent : EXTSeNsor : CONFig ?	
機能	すべてのエレメントの有効な外部電流センサレンジを問い合わせます。
構文	<pre>[: INPut] : CURRent : EXTSeNsor : CONFig ?</pre>

[: INPut] : CURRent : EXTSeNsor : CONFig [: ALL]	
機能	すべてのエレメントの有効な外部電流センサレンジを一括設定します。
構文	<pre>[: INPut] : CURRent : EXTSeNsor : CONFig [: ALL] { ALL < 電圧 > [, < 電圧 >] [, < 電圧 >] ... } ALL = 全レンジ有効 ・ クレストファクタを「3」に設定しているとき < 電圧 > = 50、100、200、500(mV)、1、2、5、10(V) ・ クレストファクタを「6」に設定しているとき < 電圧 > = 25、50、100、250、500(mV)、1、2.5、5(V)</pre>
例	<pre>: INPUT : CURRENT : EXTSENSOR : CONFIG : ALL ALL : INPUT : CURRENT : EXTSENSOR : CONFIG : ALL 10,5,2,1</pre>
解説	パラメータには、有効にする外部電流センサレンジをすべて並べて記述します。全レンジ有効にする場合は、パラメータ「ALL」を指定します。

[: INPut] : CURRent : EXTSeNsor : CONFig : ELEMeNt < x >	
機能	各エレメントの有効な外部電流センサレンジを設定 / 問い合わせします。
構文	<pre>[: INPut] : CURRent : EXTSeNsor : CONFig : ELEMeNt < x > { ALL < 電圧 > [, < 電圧 >] [, < 電圧 >] ... } [: INPut] : CURRent : EXTSeNsor : CONFig : ELEMeNt < x > ? < x > = 1 ~ 6 (エレメント) ALL = 全レンジ有効 < 電圧 > = ([: INPut] : CURRent : EXTSeNsor : CONFig [: ALL]) を参照</pre>
例	<pre>: INPUT : CURRENT : EXTSENSOR : CONFIG : ELEMENT1 ALL : INPUT : CURRENT : EXTSENSOR : CONFIG : ELEMENT1 ? -> : INPUT : CURRENT : EXTSENSOR : CONFIG : ELEMENT1 ALL : INPUT : CURRENT : EXTSENSOR : CONFIG : ELEMENT1 10,5,2,1 : INPUT : CURRENT : EXTSENSOR : CONFIG : ELEMENT1 ? -> : INPUT : CURRENT : EXTSENSOR : CONFIG : ELEMENT1 10.00E+00,5.00E+00, 2.00E+00,1.00E+00</pre>
解説	パラメータには、有効にする外部電流センサレンジをすべて並べて記述します。全レンジ有効にする場合は、パラメータ「ALL」を指定します。

[: INPut] : CURRent : EXTSeNsor : DISPlay	
機能	外部電流センサレンジの表示方式を設定 / 問い合わせします。
構文	<pre>[: INPut] : CURRent : EXTSeNsor : DISPlay { DiRect MEASure } [: INPut] : CURRent : EXTSeNsor : DISPlay ?</pre>
例	<pre>: INPUT : CURRENT : EXTSENSOR : DISPLAY DIRECT : INPUT : CURRENT : EXTSENSOR : DISPLAY ? -> : INPUT : CURRENT : EXTSENSOR : DISPLAY DIRECT</pre>

[: INPut] : CURRent : EXTSeNsor : POJump ?	
機能	すべてのエレメントの電流ピークオーバー発生時のジャンプ先レンジを問い合わせます。
構文	<pre>[: INPut] : CURRent : EXTSeNsor : POJump ?</pre>

[: INPut] : CURRent : EXTSensor :**POJump [: ALL]**

機能 すべてのエレメントの電流ピークオーバー発生時のジャンプ先レンジを一括設定します。

構文 [: INPut] : CURRent : EXTSensor :
POJump [: ALL] { OFF | < 電圧 > }
OFF = ジャンプ先電流レンジなし
・ クレストファクタを「3」に設定しているとき
< 電圧 > = 50、100、200、500(mV)、1、2、5、10(V)
・ クレストファクタを「6」に設定しているとき
< 電圧 > = 25、50、100、250、500(mV)、1、2.5、5(V)

例 : INPUT:CURRENT:EXTSENSOR:POJUMP:
ALL OFF

[: INPut] : CURRent : EXTSensor : POJump : ELEMent<x>

機能 各エレメントの電流ピークオーバー発生時のジャンプ先レンジを設定 / 問い合わせします。

構文 [: INPut] : CURRent : EXTSensor : POJump : ELEMent<x> { OFF | < 電圧 > }
[: INPut] : CURRent : EXTSensor : POJump : ELEMent<x>?
<x> = 1 ~ 6 (エレメント)
OFF = ジャンプ先電流レンジなし
< 電圧 > = ([: INPut] : CURRent : EXTSensor : POJump [: ALL]) を参照

例 : INPUT:CURRENT:EXTSENSOR:POJUMP:ELEM
ENT1 10V
: INPUT:CURRENT:EXTSENSOR:POJUMP:ELEM
ENT1? ->
: INPUT:CURRENT:EXTSENSOR:POJUMP:ELEM
ENT1 10.00E+00

[: INPut] : CURRent : POJump ?

機能 すべてのエレメントの電流ピークオーバー発生時のジャンプ先レンジを問い合わせます。

構文 [: INPut] : CURRent : POJump ?

[: INPut] : CURRent : POJump [: ALL]

機能 すべてのエレメントの電流ピークオーバー発生時のジャンプ先レンジを一括設定します。

構文 [: INPut] : CURRent :
POJump [: ALL] { OFF | < 電流 > }
OFF = ジャンプ先電流レンジなし
・ 50A 入力エレメントのとき
・ クレストファクタを「3」に設定しているとき
< 電流 > = 1、2、5、10、20、50(A)
・ クレストファクタを「6」に設定しているとき
< 電流 > = 500(mA)、1、2.5、5、10、25(A)
・ 5A 入力エレメントのとき
・ クレストファクタを「3」に設定しているとき
< 電流 > = 10、20、50、100、200、500(mA)、1、2、5(A)
・ クレストファクタを「6」に設定しているとき
< 電流 > = 5、10、25、50、100、250、500(mA)、1、2.5(A)

例 : INPUT:CURRENT:POJUMP:ALL OFF

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

機能 各エレメントの電流ピークオーバー発生時のジャンプ先レンジを設定 / 問い合わせします。

構文 [: INPut] : CURRent : POJump : ELEMent<x>
{ OFF | < 電流 > }
[: INPut] : CURRent : POJump : ELEMent<x>?
<x> = 1 ~ 6 (エレメント)
OFF = ジャンプ先電流レンジなし
< 電流 > = ([: INPut] : CURRent : POJump [: ALL]) を参照

例 : INPUT:CURRENT:POJUMP:ELEMENT1 50A
: INPUT:CURRENT:POJUMP:ELEMENT1? ->
: INPUT:CURRENT:POJUMP:
ELEMENT1 50.0E+00

[: 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?

5.13 INPut グループ

[: 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	<pre>[: INPut] : CURRent : RANGE [: ALL] { < 電流 > (EXTeRnal , < 電圧 >) }</pre> 50A 入力エレメントのとき - クレストファクタを「3」に設定しているとき < 電流 > = 1、2、5、10、20、50(A)(電流直接入力 のとき) < 電圧 > = 50、100、200、500(mV)、1、2、5、10(V)(外部電流センサ入力 のとき) - クレストファクタを「6」に設定しているとき < 電流 > = 500(mA)、1、2.5、5、10、25(A)(電流直接入力 のとき) < 電圧 > = 25、50、100、250、500(mV)、1、2.5、5(V)(外部電流センサ入力 のとき) 5A 入力エレメントのとき - クレストファクタを「3」に設定しているとき < 電流 > = 10、20、50、100、200、500(mA)、1、2、5(A)(電流直接入力 のとき) < 電圧 > = 50、100、200、500(mV)、1、2、5、10(V)(外部電流センサ入力 のとき) - クレストファクタを「6」に設定しているとき < 電流 > = 5、10、25、50、100、250、500(mA)、1、2.5(A)(電流直接入力 のとき) < 電圧 > = 25、50、100、250、500(mV)、1、2.5、5(V)(外部電流センサ入力 のとき)
例	<pre>: INPUT : CURRENT : RANGE : ALL 50A : INPUT : CURRENT : RANGE : ALL EXTERNAL , 10V</pre>
解説	「EXTeRnal, < 電圧 >」は外部電流センサ入力 (オプション、/EX) 搭載時のみ選択可能です。

[: 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	<pre>[: INPut] : CURRent : RANGE : ELEMeNt < x > { < 電流 > (EXTeRnal , < 電圧 >) }</pre> <pre>[: INPut] : CURRent : RANGE : ELEMeNt < x > ?</pre> < x > = 1 ~ 6 (エレメント) < 電流 > , < 電圧 > = ([: INPut] : CURRent : RANGE [: ALL]) を参照
例	<pre>: INPUT : CURRENT : RANGE : ELEMENT1 50A : INPUT : CURRENT : RANGE : ELEMENT1 ? -> : INPUT : CURRENT : RANGE : ELEMENT1 50.0E+00 : INPUT : CURRENT : RANGE : ELEMENT1 EXTERNAL , 10V : INPUT : CURRENT : RANGE : ELEMENT1 ? -> : INPUT : CURRENT : RANGE : ELEMENT1 EXTERNAL , 10.00E+00</pre>

[: INPut] : CURRent : RANGE :

機能	結線ユニット { Σ A Σ B Σ C } に属するエレメントの電流レンジを一括設定します。
構文	<pre>[: INPut] : CURRent : RANGE : { SIGMA SIGMB SIGMC } { < 電流 > (EXTeRnal , < 電圧 >) }</pre> < 電流 > , < 電圧 > = ([: INPut] : CURRent : RANGE [: ALL]) を参照
例	<pre>: INPUT : CURRENT : RANGE : SIGMA 50A : INPUT : CURRENT : RANGE : SIGMA EXTERNAL , 10V</pre>
解説	結線方式の設定 ([: INPut] : WIRing) により結線ユニット { Σ A Σ B Σ C } が存在しないとき、{ SIGMA SIGMB SIGMC } は無効です。

[: INPut] : CURRent : SRATio ?

機能	すべてのエレメントの外部電流センサ換算比を問い合わせます。
構文	[: INPut] : CURRent : SRATio ?
解説	外部電流センサ入力 (オプション、/EX) 搭載時のみ有効です。

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

機能	すべてのエレメントの外部電流センサ換算比を一括設定します。
構文	<pre>[: INPut] : CURRent : SRATio [: ALL]</pre> { < NRf > } < NRf > = 0.0001 ~ 99999.9999
例	<pre>: INPUT : CURRENT : SRATIO : ALL 10</pre>

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

機能	各エレメントの外部電流センサ換算比を設定 / 問い合わせします。
構文	<pre>[: INPut] : CURRent : SRATio : ELEMeNt < x > { < NRf > }</pre> <pre>[: INPut] : CURRent : SRATio : ELEMeNt < x > ?</pre> < x > = 1 ~ 6 (エレメント) < NRf > = 0.0001 ~ 99999.9999
例	<pre>: INPUT : CURRENT : SRATIO : ELEMENT1 10 : INPUT : CURRENT : SRATIO : ELEMENT1 ? -> : INPUT : CURRENT : SRATIO : ELEMENT1 10.0000</pre>

[: INPut] : CURRent : SRATio : { SIGMA | SIGMB | SIGMC }

機能 結線ユニット { ΣA | ΣB | ΣC } に属するエレメントの外部電流センサ換算比を一括設定します。

構文 **[: INPut] : CURRent : SRATio :**
{ SIGMA | SIGMB | SIGMC } { < NRf > }
< NRf > = 0.0001 ~ 99999.9999

例 **: INPUT : CURRENT : SRATIO : SIGMA 10**
解説 結線方式の設定 ([: INPut] : WIRing) により結線ユニット { ΣA | ΣB | ΣC } が存在しないとき、{ SIGMA | SIGMB | SIGMC } は無効です。

[: INPut] : ESElect

機能 測定レンジの設定対象を設定 / 問い合わせします。

構文 **[: INPut] : ESElect { < NRf > | ALL }**
[: INPut] : ESElect ?
< NRf > = 1 ~ 6 (エレメント)

例 **: INPUT : ESELECT 1**
: INPUT : ESELECT ? -> : INPUT : ESELECT 1

[: INPut] : 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 **[: INPut] : FILTer ?**

[: INPut] : FILTer : FREQuency ?

機能 すべてのエレメントの周波数フィルタを問い合わせます。

構文 **[: INPut] : FILTer : FREQuency ?**

[: 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]**
{ OFF | < 周波数 > }
OFF = 周波数フィルタ OFF
< 周波数 > = 100Hz、1kHz (周波数フィルタ ON、カットオフ周波数)

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 > { OFF | < 周波数 > }
[: INPut] : FILTer : FREQuency : ELEMeNt
< x > ?

< x > = 1 ~ 6 (エレメント)
OFF = 周波数フィルタ OFF
< 周波数 > = 100Hz、1kHz (周波数フィルタ ON、カットオフ周波数)

例 **: INPUT : FILTER : FREQUENCY :**
ELEMENT1 100HZ
: INPUT : FILTER : FREQUENCY : ELEMENT1 ? ->
: INPUT : FILTER : FREQUENCY :
ELEMENT1 100.0E+00

[: INPut] : FILTer : LINE ?

機能 すべてのエレメントのラインフィルタを問い合わせます。

構文 **[: INPut] : FILTer : LINE ?**

[: 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 > = 0.1 ~ 100.0 (kHz)、300kHz、1MHz (ラインフィルタ ON、カットオフ周波数)

例 **: INPUT : FILTER : LINE : ALL OFF**
解説 0.1 ~ 100.0(kHz) の間は、0.1(kHz) の分解能で設定できます。

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

機能 各エレメントのラインフィルタを設定 / 問い合わせします。

構文 **[: INPut] : FILTer [: LINE] :**
ELEMeNt < x > { OFF | < 周波数 > }
[: INPut] : FILTer [: LINE] :
ELEMeNt < x > ?

< x > = 1 ~ 6 (エレメント)
OFF = ラインフィルタ OFF
< 周波数 > = 0.1 ~ 100.0 (kHz)、300kHz、1MHz (ラインフィルタ ON、カットオフ周波数)

例 **: INPUT : FILTER : LINE : ELEMENT1 0.5KHZ**
: INPUT : FILTER : LINE : ELEMENT1 ? ->
: INPUT : FILTER : LINE :
ELEMENT1 500.0E+00

解説 0.1 ~ 100.0(kHz) の間は、0.1(kHz) の分解能で設定できます。

5.13 INPut グループ

[: INPut] : FILTeR [: LINE] : { SIGMA | SIGMB | SIGMC }

機能	結線ユニット { Σ A Σ B Σ C } に属する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] : { SIGMA SIGMB SIGMC } { OFF < 周波数 > } OFF = ラインフィルタ OFF < 周波数 > = 0.1 ~ 100.0 (kHz)、300kHz、1MHz (ラインフィルタ ON、カットオフ周波数)
例	: INPUT: FILTER: LINE: SIGMA 300KHZ
解説	0.1 ~ 100.0(kHz) の間は、0.1(kHz) の分解能で設定できます。

[: 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 ~ 6 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 ~ 6 (エレメント)
例	: INPUT: MODULE? 1 -> 50 : INPUT: MODULE? -> 50, 50, 50, 50, 50, 50
解説	・ 応答の内容は次のとおりです。 50 = 50A 入力エレメント (最大電流レンジ = 50A) 5 = 5A 入力エレメント (最大電流レンジ = 5A) 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 : CoNDition : { SPEEd | ToRQue | AuX < x > }

機能	{ 回転速度 トルク AUX } の NULL 動作状態を問い合わせます。
構文	[: INPut] : NuLL : CoNDition : { SPEEd ToRQue AuX < x > } ? < x > = 1 ~ 2 (AUX 入力チャネル)
例	: INPUT: NULL: CONDITION: SPEED? -> 1
解説	・ 応答の内容は次のとおりです。 0 = NULL 機能オフ 1 = NULL 機能動作中 ・ { SPEEd ToRQue } は、モータ評価機能 (オプション、/MTR) 搭載時のみ有効です。 ・ AUX < x > は、外部信号入力 (オプション、/AUX) 搭載時のみ有効です。

[: INPut] : NuLL : CoNDition : { U < x > | I < x > }

機能	各エレメントの { 電圧 電流 } の NULL 動作状態を問い合わせます。
構文	[: INPut] : NuLL : CoNDition : { U < x > I < x > } ? < x > = 1 ~ 6 (エレメント)
例	: INPUT: NULL: CONDITION: U1? -> 1
解説	応答の内容は次のとおりです。 0 = NULL 機能オフ 1 = NULL 機能動作中

[: INPut] : NuLL [: STAtE]

機能	NULL 機能の ON/OFF を設定 / 問い合わせします。
構文	[: INPut] : NuLL [: STAtE] { < Boolean > } [: INPut] : NuLL : STAtE ?
例	: INPUT: NULL: STATE ON : INPUT: NULL: STATE? -> : INPUT: NULL: STATE 1

[: INPut] : NuLL : TARGeT ?

機能	NULL 機能の動作対象に関するすべての設定値を問い合わせます。
構文	[: INPut] : NuLL : TARGeT ?

[: INPut] : NuLL : TARGeT [: MoDE]

機能	NULL 機能の動作対象の選択モードを設定 / 問い合わせします。
構文	[: INPut] : NuLL : TARGeT [: MoDE] { ALL SElect }
例	: INPUT: NULL: TARGET: MODE? : INPUT: NULL: TARGET: MODE ALL : INPUT: NULL: TARGET: MODE? -> : INPUT: NULL: TARGET: MODE ALL

[: INPut] : NuLL : TARGeT : { SPEEd | ToRQue | AuX < x > }

機能	{ 回転速度 トルク AUX } の NULL 動作対象を設定 / 問い合わせします。
構文	[: INPut] : NuLL : TARGeT : { SPEEd ToRQue AuX < x > } { ON HoLD OFF } [: INPut] : NuLL : TARGeT : { SPEEd ToRQue AuX < x > } ? < x > = 1 ~ 2 (AUX 入力チャネル) ON = NULL 機能有効 (NULL ON 時に NULL 値を新規取得) HoLD = NULL 機能有効 (NULL ON 時に前回 NULL 値を継続) OFF = NULL 機能無効 (NULL 動作対象外)
例	: INPUT: NULL: TARGET: SPEED ON : INPUT: NULL: TARGET: SPEED? -> : INPUT: NULL: TARGET: SPEED ON
解説	・ { SPEEd ToRQue } は、モータ評価機能 (オプション、/MTR) 搭載時のみ有効です。 ・ AUX < x > は、外部信号入力 (オプション、/AUX) 搭載時のみ有効です。

[: INPut] : NULL : TARGET : { U<x> | I<x> }

機能 各エレメントの { 電圧 | 電流 } の NULL 動作対象を設定 / 問い合わせします。

構文 [: INPut] : NULL : TARGET : { U<x> | I<x> } { ON | HOLD | OFF }
[: INPut] : NULL : TARGET : { U<x> | I<x> } ?
<x> = 1 ~ 6 (エレメント)
ON = NULL 機能有効 (NULL ON 時に NULL 値を新規取得)
HOLD = NULL 機能有効 (NULL ON 時に前回 NULL 値を継続)
OFF = NULL 機能無効 (NULL 動作対象外)

例 : INPUT:NULL:TARGET:U1 ON
: INPUT:NULL:TARGET:U1? ->
: INPUT:NULL:TARGET:U1 ON

[: INPut] : NULL : TARGET : { UALL | IALL }

機能 すべてのエレメントの { 電圧 | 電流 } の NULL 動作対象を一括設定します。

構文 [: INPut] : NULL : TARGET : { UALL | IALL } { ON | HOLD | OFF }
ON = NULL 機能有効 (NULL ON 時に NULL 値を新規取得)
HOLD = NULL 機能有効 (NULL ON 時に前回 NULL 値を継続)
OFF = NULL 機能無効 (NULL 動作対象外)

例 : INPUT:NULL:TARGET:UALL ON

[: 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
A2	A1	Tq	Sp	I6	U6	I5	U5	I4	U4	I3	U3	I2	U2	I1	U1

Sp: 回転速度

Tq: トルク

A1: AUX1

A2: AUX2

[: 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?

[: 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?

[: 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 ~ 6 (エレメント)

例 : INPUT:SCALING:STATE:ELEMENT1 OFF
: INPUT:SCALING:STATE:ELEMENT1? ->
: INPUT:SCALING:STATE:ELEMENT1 0

[: 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 } ?

[: 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 ~ 6 (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

5.13 INPut グループ

[: INPut] : SCALing : { VT | CT | SFACtor } : { SIGMA | SIGMB | SIGMC }

機能 結線ユニット { Σ A | Σ B | Σ C } に属するエレメントの { VT 比 | CT 比 | 電力係数 } を一括設定します。

構文 `[ : INPut ] : SCALing : { VT | CT | SFACtor } : { SIGMA | SIGMB | SIGMC } { <NRf> }`
`<NRf> = 0.0001 ~ 99999.9999`

例 `: INPUT: SCALING: VT: SIGMA 1`

解説 結線方式の設定 ([: INPut]: WIRing) により結線ユニット { Σ A | Σ B | Σ C } が存在しないとき、{ SIGMA | SIGMB | SIGMC } は無効です。

[: 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 ?`

[: 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 ~ 6 ( 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 ~ 6 ( 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 | SIGMC }

機能 結線ユニット { Σ A | Σ B | Σ C } に属する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 | SIGMC } { U<x> | I<x> | EXTernal | NONE }`
`<x> = 1 ~ 6 ( 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: SIGMA I1`

解説 結線方式の設定 ([: INPut]: WIRing) により結線ユニット { Σ A | Σ B | Σ C } が存在しないとき、{ SIGMA | SIGMB | SIGMC } 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 ?`

[: 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 ?`

[: 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 ~ 6 ( 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 | SIGMC }

機能 結線ユニット { Σ A | Σ B | Σ C } に属する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 | SIGMC } { <Boolean> }`

例 `: INPUT: VOLTAGE: AUTO: SIGMA ON`

解説 結線方式の設定 ([: INPut]: WIRing) により結線ユニット { Σ A | Σ B | Σ C } が存在しないとき、{ SIGMA | SIGMB | SIGMC } は無効です。

[: INPut] : VOLTage: CONFIg ?

機能 すべてのエレメントの有効な電圧レンジを問い合わせます。

構文 `[ : INPut ] : VOLTage: CONFIg ?`

[: INPut] : VOLTage : CONFig [: ALL]

機能 すべてのエレメントの有効な電圧レンジを一括設定します。

構文 [: INPut] : VOLTage : CONFig [: ALL]
 { ALL | < 電圧 > [, < 電圧 >] [, < 電圧 >] ... }
 ALL = 全レンジ有効
 ・ クレストファクタを「3」に設定しているとき
 < 電圧 > = 1.5、3、6、10、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 > = 0.75、1.5、3、5、7.5、15、30、50、75、150、300、500(V)

例 : INPUT:VOLTAGE:CONFIG:ALL ALL
 : INPUT:VOLTAGE:CONFIG:ALL
 1000,300,100,30,10

解説 パラメータには、有効にする電圧レンジをすべて並べて記述します。全レンジ有効にする場合は、パラメータ「ALL」を指定します。

[: INPut] : VOLTage : CONFig : ELEMEnt < x >

機能 各エレメントの有効な電圧レンジを設定 / 問い合わせします。

構文 [: INPut] : VOLTage : CONFig : ELEMEnt < x >
 { ALL | < 電圧 > [, < 電圧 >] [, < 電圧 >] ... }
 [: INPut] : VOLTage : CONFig : ELEMEnt < x > ?
 < x > = 1 ~ 6 (エレメント)
 ALL = 全レンジ有効
 < 電圧 > = ([: INPut] : VOLTage : CONFig [: ALL]) を参照

例 : INPUT:VOLTAGE:CONFIG:ELEMENT1 ALL
 : INPUT:VOLTAGE:CONFIG:ELEMENT1? ->
 : INPUT:VOLTAGE:CONFIG:ELEMENT1 ALL
 : INPUT:VOLTAGE:CONFIG:ELEMENT1
 1000,300,100,30,10
 : INPUT:VOLTAGE:CONFIG:ELEMENT1? ->
 : INPUT:VOLTAGE:CONFIG:ELEMENT1
 1.0000E+03,300.0E+00,100.0E+00,30.0E+00,10.0E+00

解説 パラメータには、有効にする電圧レンジをすべて並べて記述します。全レンジ有効にする場合は、パラメータ「ALL」を指定します。

[: INPut] : VOLTage : POJump ?

機能 すべてのエレメントの電圧ピークオーバー発生時のジャンプ先レンジを問い合わせます。

構文 [: INPut] : VOLTage : POJump ?

[: INPut] : VOLTage : POJump [: ALL]

機能 すべてのエレメントの電圧ピークオーバー発生時のジャンプ先レンジを一括設定します。

構文 [: INPut] : VOLTage : POJump [: ALL]
 { OFF | < 電圧 > }
 OFF = ジャンプ先電圧レンジなし
 ・ クレストファクタを「3」に設定しているとき
 < 電圧 > = 1.5、3、6、10、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 > = 0.75、1.5、3、5、7.5、15、30、50、75、150、300、500(V)

例 : INPUT:VOLTAGE:POJUMP:ALL OFF

[: INPut] : VOLTage : POJump : ELEMEnt < x >

機能 各エレメントの電圧ピークオーバー発生時のジャンプ先レンジを設定 / 問い合わせします。

構文 [: INPut] : VOLTage : POJump : ELEMEnt < x >
 { OFF | < 電圧 > }
 [: INPut] : VOLTage : POJump : ELEMEnt < x > ?
 < x > = 1 ~ 6 (エレメント)
 OFF = ジャンプ先電圧レンジなし
 < 電圧 > = ([: INPut] : VOLTage : POJump [: ALL]) を参照

例 : INPUT:VOLTAGE:POJUMP:ELEMENT1 1000V
 : INPUT:VOLTAGE:POJUMP:ELEMENT1? ->
 : INPUT:VOLTAGE:POJUMP:
 ELEMENT1 1.0000E+03

[: 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 ?

[: 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 > = 1.5、3、6、10、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 > = 0.75、1.5、3、5、7.5、15、30、50、75、150、300、500(V)

例 : INPUT:VOLTAGE:RANGE:ALL 1000V

5.13 INPut グループ

[: 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 ~ 6 (エレメント)
< 電圧 > = ([: INPut] : VOLTage : RANGE [: ALL]) を参照

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

[: INPut] : VOLTage : RANGE :

{ SIGMA | SIGMB | SIGMC }

機能 結線ユニット { Σ A | Σ B | Σ C } に属する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 | SIGMC } { < 電圧 > }
< 電圧 > = ([: INPut] : VOLTage : RANGE [: ALL]) を参照

例 : INPUT:VOLTAGE:RANGE:SIGMA 1000V
解説 結線方式の設定 ([: INPut] : WIRing) により結線ユニット { Σ A | Σ B | Σ C } が存在しないとき、{ SIGMA | SIGMB | SIGMC } 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)] [, (P1W2 | P1W3 | P3W3 | P3W4 | V3A3)] [, (P1W2 | P1W3 | P3W3 | P3W4 | V3A3)] [, (P1W2 | P1W3 | P3W3)] [, P1W2] }
[: 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)]

例 ・ 6 エレメントモデルのときの例
INPUT:WIRING P1W2,P1W2,P1W2,P1W2,P1W2,P1W2
INPUT:WIRING? -> : INPUT:WIRING
P1W2,P1W2,P1W2,P1W2,P1W2,P1W2
INPUT:WIRING P1W3,P1W3,P1W3
INPUT:WIRING? ->
: INPUT:WIRING P1W3,P1W3,P1W3
INPUT:WIRING P3W4,V3A3
INPUT:WIRING? ->
: INPUT:WIRING P3W4,V3A3
・ 3 エレメントモデルのときの例
INPUT:WIRING P1W2,P3W3
INPUT:WIRING? ->
: INPUT:WIRING P1W2,P3W3
INPUT:WIRING P3W4
INPUT:WIRING? ->
: INPUT:WIRING P3W4
解説 ・ 結線方式のパタ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ンがあります。結線方式のパターンについては、ユーザズマニュアル [機能編] IM WT1801-01JA 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。

5.14 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 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。

: 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?

: INTEGrate:ACAL

機能 積算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:INdependent

機能 エLEMENT別積算の ON/OFF を設定 / 問い合わせします。
構文 : INTEGrate:INdependent {<Boolean>}
: INTEGrate:INdependent?
例 : INTEGrate:INDEPENDENT OFF
: INTEGrate:INDEPENDENT? ->
: INTEGrate:INDEPENDENT 0

: INTEGrate:MODE

機能 積算モードを設定 / 問い合わせします。
構文 : INTEGrate:MODE {NORMAL|CONTinuous|RNORMAL|RCONTinuous}
: INTEGrate:MODE?
NORMAL = 標準積算モード
CONTinuous = 繰り返し積算モード
RNORMAL = 実時間制御標準積算モード
RCONTinuous = 実時間制御繰り返し積算モード
例 : INTEGrate:MODE NORMAL
: INTEGrate:MODE? ->
: INTEGrate:MODE NORMAL

: INTEGrate:QMODE?

機能 すべてのELEMENTの電流積算の電流モードを問い合わせます。
構文 : INTEGrate:QMODE?

: INTEGrate:QMODE[:ALL]

機能 すべてのELEMENTの電流積算の電流モードを一括設定します。
構文 : INTEGrate:QMODE[:ALL]
{RMS|MEAN|DC|RMEAN|AC}
例 : INTEGrate:QMODE:ALL DC

: INTEGrate:QMODE:ELEMENT<x>

機能 各ELEMENTの電流積算の電流モードを設定 / 問い合わせします。
構文 : INTEGrate:QMODE:ELEMENT<x>
{RMS|MEAN|DC|RMEAN|AC}
: INTEGrate:QMODE:ELEMENT<x>?
<x> = 1 ~ 6 (ELEMENT)
例 : INTEGrate:QMODE:ELEMENT1 DC
: INTEGrate:QMODE:ELEMENT1? ->
: INTEGrate:QMODE:ELEMENT1 DC
解説 エLEMENT別積算 (「:INTEGrate:INdependent」) の ON/OFF に関係なく、各ELEMENTの電流モードの設定により動作します。

: INTEGrate:RESet

機能 積算値をリセットします。
構文 : INTEGrate:RESet {[<NRf>][,<NRf>][,<NRf>][,<NRf>][,<NRf>][,<NRf>]}
<NRf> = 1 ~ 6 (ELEMENT)
例 : INTEGrate:RESet (全ELEMENT実行)
: INTEGrate:RESet 1,2,3 (ELEMENTを指定して実行)
解説
・ELEMENT別積算 (「:INTEGrate:INdependent」) が ON(1) のとき、パラメータ (実行するELEMENT) を指定することができます。パラメータを省略すると、すべてのELEMENTを指定したことになります。
・ELEMENT別積算 (「:INTEGrate:INdependent」) が OFF(0) のときは、パラメータの指定はできません。

: INTEGrate:RTALL:{START|END}

機能 実時間制御積算モードにおけるすべてのELEMENTの積算 { 開始 | 終了 } 予約時刻を一括設定します。
構文 : INTEGrate:RTALL:{START|END} {<NRf>,<NRf>,<NRf>,<NRf>,<NRf>,<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:RTALL:START
2010,1,1,0,0,0

5.14 INTEGrate グループ

:INTEGrate:RTIME<x>?

機能 実時間制御積算モードにおける積算開始/終了予約時刻を問い合わせます。

構文 :INTEGrate:RTIME<x>?
<x> = 1 ~ 6 (エLEMENT)

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

機能 実時間制御積算モードにおける積算 { 開始 | 終了 } 予約時刻を設定 / 問い合わせします。

構文 :INTEGrate:RTIME<x>:{START|END} {<NRf>,<NRf>,<NRf>,<NRf>,<NRf>,<NRf>}
:INTEGrate:RTIME<x>:{START|END}?
<x> = 1 ~ 6 (エLEMENT)
{<NRf>,<NRf>,<NRf>,<NRf>,<NRf>,<NRf>} = 2001、1、1、0、0、0 ~ 2099、12、31、23、59、59
1 つ目の <NRf> = 2001 ~ 2099 (年)
2 つ目の <NRf> = 1 ~ 12 (月)
3 つ目の <NRf> = 1 ~ 31 (日)
4 つ目の <NRf> = 0 ~ 23 (時)
5 つ目の <NRf> = 0 ~ 59 (分)
6 つ目の <NRf> = 0 ~ 59 (秒)

例 :INTEGRATE:RTIME1:START 2010,1,1,0,0,0
:INTEGRATE:RTIME1:START? ->
:INTEGRATE:RTIME1:START 2010,1,1,0,0,0

解説

- 積算モード (:INTEGrate:MODE) が実時間制御積算モード ({RNORmal|RCONtinuous}) のときに有効な設定です。
- ELEMENT別積算 (:INTEGrate:INDepeNdeNt) が OFF(0) のとき、<x> は省略できます (<x>=1)。ELEMENT 1 の積算 { 開始 | 終了 } 予約時刻の設定により、すべてのELEMENTが動作します。

:INTEGrate:START

機能 積算をスタートします。

構文 :INTEGrate:START {[<NRf>][,<NRf>][,<NRf>][,<NRf>][,<NRf>]}
<NRf> = 1 ~ 6 (エLEMENT)

例 :INTEGRATE:START (全ELEMENT実行)
:INTEGRATE:START 1,2,3 (ELEMENTを指定して実行)

解説

- ELEMENT別積算 (:INTEGrate:INDepeNdeNt) が ON(1) のとき、パラメータ (実行するELEMENT) を指定することができます。パラメータを省略すると、すべてのELEMENTを指定したことになります。
- ELEMENT別積算 (:INTEGrate:INDepeNdeNt) が OFF(0) のときは、パラメータの指定はできません。

:INTEGrate:STATE?

機能 積算状態を問い合わせます。

構文 :INTEGrate:STATE? {<NRf>}
:INTEGrate:STATE?
<NRf> = 1 ~ 6 (エLEMENT)

例

- ELEMENT別積算 ((:INTEGrate:INDepeNdeNt) が OFF(0) のとき
:INTEGRATE:STATE? -> RESET
- ELEMENT別積算 ((:INTEGrate:INDepeNdeNt) が ON(1) のとき
:INTEGRATE:STATE? 1 -> RESET
:INTEGRATE:STATE? ->
RESET,RESET,RESET,RESET,RESET,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)
TIMeup = 積算タイマ時間による積算ストップ
- ELEMENT別積算 (:INTEGrate:INDepeNdeNt) が OFF(0) のときは、パラメータ (問い合わせるELEMENT) の指定はできません
- ELEMENT別積算 (:INTEGrate:INDepeNdeNt) が ON(1) のとき、パラメータを指定することができます。パラメータを省略した場合、すべてのELEMENTの積算状態をELEMENT 1 から順に出力します。

: INTEGrate: STOP

機能 積算をストップします。

構文 :INTEGrate:STOP {[<NRf>][,<NRf>]
[,<NRf>][,<NRf>][,<NRf>][,<NRf>]}
<NRf> = 1 ~ 6 (エLEMENT)

例 :INTEGRATE:STOP (全ELEMENT実行)
:INTEGRATE:STOP 1,2,3 (ELEMENTを指定して実行)

解説

- ・ ELEMENT別積算 (「:INTEGrate:INDependent」) が ON(1) のとき、パラメータ (実行するELEMENT) を指定することができます。パラメータを省略すると、すべてのELEMENTを指定したことになります。
- ・ ELEMENT別積算 (「:INTEGrate:INDependent」) が OFF(0) のときは、パラメータの指定はできません。

: INTEGrate: TImEr<x>

機能 積算タイマ時間を設定 / 問い合わせします。

構文 :INTEGrate:TImEr<x> {<NRf>,<NRf>,<NRf>}
:INTEGrate:TImEr<x>?
<x> = 1 ~ 6 (ELEMENT)
{<NRf>,<NRf>,<NRf>} = 0、0、0 ~ 10000、0、0
1 つ目の <NRf> = 0 ~ 10000 (時間)
2 つ目の <NRf> = 0 ~ 59 (分)
3 つ目の <NRf> = 0 ~ 59 (秒)

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

解説 ELEMENT別積算 (「:INTEGrate:INDependent」) が OFF(0) のとき、<x> は省略できます (<x>=1)。ELEMENT 1 の積算タイマ時間の設定により、すべてのELEMENTが動作します。

: INTEGrate: TMALl

機能 すべてのELEMENTの積算タイマ時間を一括設定します。

構文 :INTEGrate:TMALl {<NRf>,<NRf>,<NRf>}
{<NRf>,<NRf>,<NRf>} = Q Q 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:TMALL 1,0,0

: INTEGrate: WPTYPE?

機能 すべてのELEMENTの極性別電力量 (WP+/WP-) の演算方式を問い合わせます。

構文 :INTEGrate:WPTYPE?

: INTEGrate: WPTYPE[: ALL]

機能 すべてのELEMENTの極性別電力量 (WP+/WP-) の演算方式を一括設定します。

構文 :INTEGrate:WPTYPE[:ALL]
{CHARGE|SOLD}

例 :INTEGRATE:WPTYPE:ALL CHARGE

: INTEGrate: WPTYPE: ELEMEnt<x>

機能 各ELEMENTの極性別電力量 (WP+/WP-) の演算方式を設定 / 問い合わせします。

構文 :INTEGrate:WPTYPE:
ELEMEnt<x> {CHARGE|SOLD}
:INTEGrate:WPTYPE:ELEMEnt<x>?
<x> = 1 ~ 6 (ELEMENT)
CHARGE = 充放電
SOLD = 売買電

例 :INTEGRATE:WPTYPE:ELEMENT1 CHARGE
:INTEGRATE:WPTYPE:ELEMENT1? ->
:INTEGRATE:WPTYPE:ELEMENT1 CHARGE

解説 ELEMENT別積算 (「:INTEGrate:INDependent」) の ON/OFF に関係なく、各ELEMENTの演算方式の設定により動作します。

5.15 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、FREQ MEASURE(SHIFT+MEASURE)、AVG、WIRING キーの「 η Formula」、「 Δ Measure」(オプション、/DT)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?

: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?

: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 ~ 64 (TYPE = EXponent のとき、減衰定数)
<NRf> = 8 ~ 64 (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 [機能編] IM WT1801-01JA を参照してください。

:MEASure:AVERaging[:STATe]

機能 アベレーシングの ON/OFF を設定 / 問い合わせします。
構文 :MEASure:AVERaging[:STATe] {<Boolean>}
:MEASure:AVERaging:STATe?
例 :MEASURE:AVERAGING:STATE ON
:MEASURE:AVERAGING:STATE? ->
:MEASURE:AVERAGING:STATE 1

: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}
:MEASure:AVERaging:TYPE?
例 :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 [機能編] IM WT1801-01JA を参照してください。

:MEASure:DMeasure?

機能 デルタ演算に関するすべての設定値を問い合わせます。
構文 :MEASure:DMeasure?
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:MODE

機能 デルタ演算の対象とする電圧 / 電流モードを設定 / 問い合わせします。
構文 :MEASure:DMeasure:MODE {RMS | MEAN | DC | RMEAN | AC}
:MEASure:DMeasure:MODE?
例 :MEASURE:DMEASURE:MODE RMS
:MEASURE:DMEASURE:MODE? ->
:MEASURE:DMEASURE:MODE RMS
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 | SIGMB | SIGMC}

機能 結線ユニット { ΣA | ΣB | ΣC } に対するデルタ演算タイプを設定 / 問い合わせします。
構文 :MEASure:DMeasure:{SIGMA | SIGMB | SIGMC} {OFF | DIFFerence | P3W3_V3A3 | ST_DT | DT_ST}
:MEASure:DMeasure:{SIGMA | SIGMB | SIGMC}?
例 :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 | ΣB | ΣC } 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: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?

: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|PC|PM|UDEF<x>)
[, (OFF|P<x>|PA|PB|PC|PM|UDEF<x>)] }
:MEASure:EFFiciency:ETA<x>?
ETA<x> の <x> = 1 ~ 4 ($\eta_1 \sim \eta_4$)
OFF = 演算なし
P<x> の <x> = 1 ~ 6 (エレメント)
PA、PB、PC = P Σ A、P Σ B、P Σ C (エレメント数により選択可能)
PM = P_m (モータ出力、モータ評価機能 (オプション、/MTR) 搭載時のみ)
UDEF<x> の <x> = 1 ~ 2 (Udef1 ~ Udef2)

例 :MEASURE:EFFICIENCY:ETA1 P3,PA
:MEASURE:EFFICIENCY:ETA1? ->
:MEASURE:EFFICIENCY:ETA1 P3,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「OFF」となります。
- 問い合わせの応答では、分母が「OFF」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|PC|PM)
[, (NONE|P<x>|PA|PB|PC|PM)]
[, (NONE|P<x>|PA|PB|PC|PM)]
[, (NONE|P<x>|PA|PB|PC|PM)] }
:MEASure:EFFiciency:UDEF<x>?
UDEF<x> の <x> = 1 ~ 2 (Udef1 ~ Udef2)
NONE = 演算項なし
P<x> の <x> = 1 ~ 6 (エレメント)
PA、PB、PC = P Σ A、P Σ B、P Σ C (エレメント数により選択可能)
PM = P_m (モータ出力、モータ評価機能 (オプション、/MTR) 搭載時のみ)

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:EVENT<x>?

機能 ユーザー定義イベントに関するすべての設定値を問い合わせます。

構文 :MEASure:EVENT<x>?
<x> = 1 ~ 8 (Event1 ~ Event8)

:MEASure:EVENT<x>:EXPRession?

機能 ユーザー定義イベントの条件式に関するすべての設定値を問い合わせます。

構文 :MEASure:EVENT<x>:EXPRession?
<x> = 1 ~ 8 (Event1 ~ Event8)

:MEASure:EVENT<x>:EXPRession:CONDition

機能 ユーザー定義イベントの条件式 (条件合成タイプ) を設定 / 問い合わせします。

構文 :MEASure:EVENT<x>:EXPRession:CONDition {<Event>[,<Logic>,<Event>]
[,<Logic>,<Event>]...}
:MEASure:EVENT<x>:EXPRession:CONDition?
<x> = 1 ~ 8 (Event1 ~ Event8)
<Event> = {<Nrf>} (<Nrf> = 1 ~ 8 (Event1 ~ Event8))
<Logic> = {AND|OR}
例 :MEASURE:EVENT1:EXPRESSION:
CONDITION 1,AND,2
:MEASURE:EVENT1:EXPRESSION:CONDITI
ON? ->
:MEASURE:EVENT1:EXPRESSION:
CONDITION 1,AND,2
解説 条件式タイプ (:MEASure:EVENT<x>:EXPRession:TY
PE) が「CONDition」のときに有効な設定です。

:MEASure:EVENT<x>:EXPRession:INVerse

機能 ユーザー定義イベントの条件式 (条件合成タイプ) の論理反転 ON/OFF を設定 / 問い合わせします。

構文 :MEASure:EVENT<x>:EXPRession:
INVerse {<Boolean>}
:MEASure:EVENT<x>:EXPRession:INVer
se?
<x> = 1 ~ 8 (Event1 ~ Event8)
例 :MEASURE:EVENT1:EXPRESSION:
INVERSE OFF
:MEASURE:EVENT1:EXPRESSION:
INVERSE? ->
:MEASURE:EVENT1:EXPRESSION:INVERSE 0
解説 条件式タイプ (:MEASure:EVENT<x>:EXPRession:TY
PE) が「CONDition」のときに有効な設定です。

5.15 MEASure グループ

:MEASure:EVENT<x>:EXPRession:ITEM

機能	ユーザー定義イベントの条件式 (範囲指定タイプ) の対象項目を設定 / 問い合わせします。
構文	:MEASure:EVENT<x>:EXPRession:ITEM {<Function>[,<Element>][,<Order>]}
例	:MEASure:EVENT<x>:EXPRession:ITEM? <x> = 1 ~ 8 (Event1 ~ Event8) <Function> = {URMS IRMS P S Q ...} <Element> = {<NRf> SIGMa SIGMB SIGMC} (<NRf> = 1 ~ 6) <Order> = {TOTal DC <NRf>} (<NRf> = 1 ~ 500)
解説	- 条件式タイプ (:MEASure:EVENT<x>:EXPRession:TYPE) が「RANge」のときに有効な設定です。 <Function> の選択肢については、「DISPlay グループ」のファンクション選択肢一覧 (1)、5-38 ページを参照してください。 <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。

:MEASure:EVENT<x>:EXPRession:LIMit<x>

機能	ユーザー定義イベントの条件式 (範囲指定タイプ) の範囲を設定 / 問い合わせします。
構文	:MEASure:EVENT<x>:EXPRession:LIMit<x> {<Operand>,<NRf>}
例	:MEASure:EVENT<x>:EXPRession:LIMit<x>? EVENT<x> の <x> = 1 ~ 8 (Event1 ~ Event8) LIMit<x> の <x> = 1 ~ 2 <Operand> = {OFF LESS LEQual EQual Great GEQual NEQual} <NRf> = -1.0000E+12 ~ 1.0000E+12
解説	- 条件式タイプ (:MEASure:EVENT<x>:EXPRession:TYPE) が「RANge」のときに有効な設定です。 <Operand> = OFF のとき、<NRf> は省略します。

:MEASure:EVENT<x>:EXPRession:STRing?

機能	ユーザー定義イベントの条件式を文字列形式で問い合わせます。
構文	:MEASure:EVENT<x>:EXPRession:STRing? <x> = 1 ~ 8 (Event1 ~ Event8)
例	:MEASure:EVENT1:EXPRession:STRing? -> "TEMP < 100.00000"
解説	条件式が無い場合は「"No Expression"」を返します。

:MEASure:EVENT<x>:EXPRession:TYPE

機能	ユーザー定義イベントの条件式タイプを設定 / 問い合わせします。
構文	:MEASure:EVENT<x>:EXPRession:TYPE {RANge CONDition}
例	:MEASure:EVENT<x>:EXPRession:TYPE? <x> = 1 ~ 8 (Event1 ~ Event8) RANge = 範囲指定タイプ CONDition = イベント合成タイプ
解説	条件式タイプ (:MEASure:EVENT<x>:EXPRession:TYPE) が「RANge」のときに有効な設定です。

:MEASure:EVENT<x>:FLABel

機能	ユーザー定義イベントの条件不適合 (False) 時の表示データ文字列を設定 / 問い合わせします。
構文	:MEASure:EVENT<x>:FLABel {<文字列>}
例	:MEASure:EVENT<x>:FLABel? <x> = 1 ~ 8 (Event1 ~ Event8) <文字列> = 6 文字以内
解説	条件式タイプ (:MEASure:EVENT<x>:EXPRession:TYPE) が「RANge」のときに有効な設定です。

:MEASure:EVENT<x>:NAME

機能	ユーザー定義イベントの名前を設定 / 問い合わせします。
構文	:MEASure:EVENT<x>:NAME {<文字列>}
例	:MEASure:EVENT<x>:NAME? <x> = 1 ~ 8 (Event1 ~ Event8) <文字列> = 8 文字以内
解説	条件式タイプ (:MEASure:EVENT<x>:EXPRession:TYPE) が「RANge」のときに有効な設定です。

:MEASure:EVENT<x>[:STATe]

機能	ユーザー定義イベントの有効 (ON)/ 無効 (OFF) を設定 / 問い合わせします。
構文	:MEASure:EVENT<x>[:STATe] {<Boolean>}
例	:MEASure:EVENT<x>:STATe? <x> = 1 ~ 8 (Event1 ~ Event8)
解説	条件式タイプ (:MEASure:EVENT<x>:EXPRession:TYPE) が「RANge」のときに有効な設定です。

:MEASure:EVENT<x>:TLABEL

機能 ユーザー定義イベントの条件適合 (True) 時の表示データ文字列を設定 / 問い合わせします。

構文 :MEASure:EVENT<x>:TLABEL {<文字列>}
:MEASure:EVENT<x>:TLABEL?
<x> = 1 ~ 8 (Event1 ~ Event8)
<文字列> = 6 文字以内

例 :MEASURE:EVENT1:TLABEL "True"
:MEASURE:EVENT1:TLABEL? ->
:MEASURE:EVENT1:TLABEL "True"

: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?

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なため、問い合わせ不要です。

: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 ~ 3 (Freq.1 ~ Freq.3)
U<x>, I<x> の <x> = 1 ~ 6 (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なため、設定不要です。

: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)

: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 "WH(E1)/TI(E1)*3600"
:MEASURE:FUNCTION1:EXPRESSION? ->
:MEASURE:FUNCTION1:
EXPRESSION "WH(E1)/TI(E1)*3600"

:MEASure:FUNCTION<x>:NAME

機能 ユーザー定義ファンクションの名前を設定 / 問い合わせします。

構文 :MEASure:FUNCTION<x>:NAME {<文字列>}
:MEASure:FUNCTION<x>:NAME?
<x> = 1 ~ 20 (F1 ~ F20)
<文字列> = 8 文字以内

例 :MEASURE:FUNCTION1:NAME "F1"
:MEASURE:FUNCTION1:NAME? ->
:MEASURE:FUNCTION1:NAME "F1"

: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 "W"
:MEASURE:FUNCTION1:UNIT? ->
:MEASURE:FUNCTION1:UNIT "W"

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 [機能編] IM WT1801-01JA を参照してください。

5.15 MEASure グループ

: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?

: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。

: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 $\pm 0 \sim 180^\circ$ (Lead/Lag) で、「360」のときは $0 \sim 360^\circ$ 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	選択肢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 [機能編] IM WT1801-01JA を参照してください。

:MEASure:SFORMula

機能	S(皮相電力) の演算式を設定 / 問い合わせします。
構文	:MEASure:SFORMula {RMS MEAN DC MRMS RMEAN} :MEASure:SFORMula?
例	:MEASURE:SFORMULA RMS :MEASURE:SFORMULA? -> :MEASURE:SFORMULA RMS
解説	選択肢に対応する演算式はそれぞれ次のとおりです。 RMS: $S = U_{rms} \times I_{rms}$ MEAN : $S = U_{mean} \times I_{mean}$ DC : $S = U_{dc} \times I_{dc}$ MRMS : $S = U_{mean} \times I_{rms}$ RMEAN: $S = U_{rmean} \times I_{rmean}$

: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 [機能編] IM WT1801-01JA を参照してください。 ・ 「TYPE3」は高調波測定機能 (オプション、/G5 または /G6) 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

5.16 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/AUX SET(SHIFT+SCALING) 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。ただし、このグループのコマンドは、モータ評価機能 (オプション、/MTR) の搭載時のみ有効です。

: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?

:MOTor:EANgLe?

機能 電気角測定に関するすべての設定値を問い合わせます。

構文 :MOTor:EANgLe?

解説 電気角測定機能は、高調波測定機能 (オプション、/G5 または /G6) 搭載時だけ有効です。

:MOTor:EANgLe:CORRection?

機能 電気角の補正值設定に関するすべての設定値を問い合わせます。

構文 :MOTor:EANgLe:CORRection?

:MOTor:EANgLe:CORRection:AENter?

機能 電気角の補正值の自動入力に関するすべての設定値を問い合わせます。

構文 :MOTor:EANgLe:CORRection:AENter?

:MOTor:EANgLe:CORRection:AENter[:EXECute]

機能 電気角の補正值の自動入力を実行します。

構文 :MOTor:EANgLe:CORRection:AENter[:EXECute]

例 :MOTOR:EANGLE:CORRECTION:AENTER:EXECUTE

解説 対象ソース (:MOTor:EANgLe:CORRection:AENter:TARGet) の電気角測定値が補正值として入力されます。

:MOTor:EANgLe:CORRection:AENter:TARGet

機能 電気角の補正值を自動入力する対象ソースを設定 / 問い合わせします。

構文 :MOTor:EANgLe:CORRection:AENter:TARGet {U<x>|I<x>}
:MOTor:EANgLe:CORRection:AENter:TARGet?

<x> = 1 ~ 6 (エレメント)

例 :MOTOR:EANGLE:CORRECTION:AENTER:TARGET U1
:MOTOR:EANGLE:CORRECTION:AENTER:TARGET? ->
:MOTOR:EANGLE:CORRECTION:AENTER:TARGET U1

:MOTor:EANgLe:CORRection:CLEar

機能 電気角の補正值をクリアします。

構文 :MOTor:EANgLe:CORRection:CLEar

例 :MOTOR:EANGLE:CORRECTION:CLEAR

:MOTor:EANgLe:CORRection[:VALue]

機能 電気角の補正值を設定 / 問い合わせします。

構文 :MOTor:EANgLe:CORRection[:VALue]
{<NRF>}

:MOTor:EANgLe:CORRection:VALue?

<NRF> = -180.00 ~ 180.00

例 :MOTOR:EANGLE:CORRECTION:VALUE 0

:MOTOR:EANGLE:CORRECTION:VALUE? ->

:MOTOR:EANGLE:CORRECTION:VALUE 0.00

:MOTor:EANgLe[:STATe]

機能 電気角測定の ON/OFF を設定 / 問い合わせします。

構文 :MOTor:EANgLe[:STATe] {<Boolean>}
:MOTor:EANgLe:STATe?

例 :MOTOR:EANGLE:STATE ON

:MOTOR:EANGLE:STATE? ->

:MOTOR:EANGLE:STATE 1

: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?

: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、1kHz (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?

5.16 MOTor グループ

: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?

:MOTor:SPEed:AUTO

機能 回転信号 (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。

:MOTor:SPEed:LSCale?

機能 回転信号 (アナログ入力方式) のリニア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 :MOTor:SPEed:LSCale?

解説 リニアスケーリングに関する設定値は、回転信号入力タイプ (:MOTor:SPEed:TYPE) が「ANALog (アナログ入力)」のときに有効です。

:MOTor:SPEed:LSCale:AVALue

機能 回転信号 (アナログ入力方式) のリニアスケールの傾き A を設定 / 問い合わせします。

構文 :MOTor:SPEed:LSCale:AVALue {<NRf>}
:MOTor:SPEed:LSCale:AVALue?
<NRf> = 1.000E-03 ~ 1.000E+06

例 :MOTOR:SPEED:LSCALE:AVALUE 1.000
:MOTOR:SPEED:LSCALE:AVALUE? ->
:MOTOR:SPEED:LSCALE:AVALUE 1.000E+00

:MOTor:SPEed:LSCale:BVALue

機能 回転信号 (アナログ入力方式) のリニアスケールのオフセット値 B を設定 / 問い合わせします。

構文 :MOTor:SPEed:LSCale:BVALue {<NRf>}
:MOTor:SPEed:LSCale:BVALue?
<NRf> = -1.000E+06 ~ 1.000E+06

例 :MOTOR:SPEED:LSCALE:BVALUE 0
:MOTOR:SPEED:LSCALE:BVALUE? ->
:MOTOR:SPEED:LSCALE:BVALUE 0.000E+00

:MOTor:SPEed:LSCale:CALCulate?

機能 回転信号 (アナログ入力方式) のリニアスケールのパラメータ計算に関するすべての設定値を問い合わせます。

構文 :MOTor:SPEed:LSCale:CALCulate?

:MOTor:SPEed:LSCale:CALCulate: {P1X|P1Y|P2X|P2Y}

機能 回転信号 (アナログ入力方式) のリニアスケールのパラメータ計算のためのデータ {Point1X|Point1Y|Point2X|Point2Y} を設定 / 問い合わせします。

構文 :MOTor:SPEed:LSCale:CALCulate:
{P1X|P1Y|P2X|P2Y} {<NRf>}
:MOTor:SPEed:LSCale:CALCulate:
{P1X|P1Y|P2X|P2Y}?

<NRf> = -1.000E+12 ~ 1.000E+12
例 :MOTOR:SPEED:LSCALE:CALCULATE:P1X 0
:MOTOR:SPEED:LSCALE:CALCULATE:
P1X? ->
:MOTOR:SPEED:LSCALE:CALCULATE:
P1X 0.000E+00

:MOTor:SPEed:LSCale:CALCulate:EXECu te

機能 回転信号 (アナログ入力方式) のリニアスケールのパラメータ計算を実行します。

構文 :MOTor:SPEed:LSCale:CALCulate:EXECu
te

例 :MOTOR:SPEED:LSCALE:CALCULATE:EXECU
TE

解説 あらかじめ設定されたデータ (Point1X,Point1Y,Point2X,Point2Y) から、リニアスケールの傾き A およびオフセット値 B を計算し、設定します。

:MOTor:SPEEd:PRANge

機能 回転信号 (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 (パルス入力方式) 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 (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リング係数を設定 / 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

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 ~ 6 (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 ~ 6 (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?

5.16 MOTO Tor グループ

:MOTO Tor:TORQue:AUTO

機能 トルク信号 (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 :MOTO Tor:TORQue:AUTO {<Boolean>}

:MOTO Tor:TORQue:AUTO?

例 :MOTO Tor:TORQue:AUTO ON

:MOTO Tor:TORQue:AUTO? ->

:MOTO Tor:TORQue:AUTO 1

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

:MOTO Tor:TORQue:LSCale?

機能 トルク信号 (アナログ入力方式) のリニアスケールリングに関するすべての設定値を問い合わせます。

構文 :MOTO Tor:TORQue:LSCale?

解説 リニアスケールリングに関する設定値は、トルク信号入力タイプ (:MOTO Tor:TORQue:TYPE) が「ANALog(アナログ入力)」のときに有効です。

:MOTO Tor:TORQue:LSCale:AVALue

機能 トルク信号 (アナログ入力方式) のリニアスケールの傾き A を設定 / 問い合わせします。

構文 :MOTO Tor:TORQue:LSCale:AVALue {<Nrf>}

:MOTO Tor:TORQue:LSCale:AVALue?

<Nrf> = 1.000E-03 ~ 1.000E+06

例 :MOTO Tor:TORQue:LSCale:AVALUE 1.000

:MOTO Tor:TORQue:LSCale:AVALUE? ->

:MOTO Tor:TORQue:LSCale:

AVALUE 1.000E+00

:MOTO Tor:TORQue:LSCale:BVALue

機能 トルク信号 (アナログ入力方式) のリニアスケールのオフセット値 B を設定 / 問い合わせします。

構文 :MOTO Tor:TORQue:LSCale:BVALue {<Nrf>}

:MOTO Tor:TORQue:LSCale:BVALue?

<Nrf> = -1.000E+06 ~ 1.000E+06

例 :MOTO Tor:TORQue:LSCale:BVALUE 0

:MOTO Tor:TORQue:LSCale:BVALUE? ->

:MOTO Tor:TORQue:LSCale:

BVALUE 0.000E+00

:MOTO Tor:TORQue:LSCale:CALCulate?

機能 トルク信号 (アナログ入力方式) のリニアスケールのパラメータ計算に関するすべての設定値を問い合わせます。

構文 :MOTO Tor:TORQue:LSCale:CALCulate?

:MOTO Tor:TORQue:LSCale:CALCulate:

{P1X|P1Y|P2X|P2Y}

機能 トルク信号 (アナログ入力方式) のリニアスケールのパラメータ計算のためのデータ {Point1X|Point1Y|Point2X|Point2Y} を設定 / 問い合わせします。

構文 :MOTO Tor:TORQue:LSCale:CALCulate:

{P1X|P1Y|P2X|P2Y} {<Nrf>}

:MOTO Tor:TORQue:LSCale:CALCulate:

{P1X|P1Y|P2X|P2Y}?

<Nrf> = -1.000E+12 ~ 1.000E+12

例 :MOTO Tor:TORQue:LSCale:CALCULATE:P1X 0

:MOTO Tor:TORQue:LSCale:CALCULATE:

P1X? ->

:MOTO Tor:TORQue:LSCale:CALCULATE:

P1X 0.000E+00

:MOTO Tor:TORQue:LSCale:CALCulate:EXECute

機能 トルク信号 (アナログ入力方式) のリニアスケールのパラメータ計算を実行します。

構文 :MOTO Tor:TORQue:LSCale:CALCulate:EXECute

例 :MOTO Tor:TORQue:LSCale:CALCULATE:EXECUTE

解説 あらかじめ設定されたデータ (Point1X,Point1Y,Point2X,Point2Y) から、リニアスケールの傾き A およびオフセット値 B を計算し、設定します。

:MOTO Tor:TORQue:PRANge

機能 トルク信号 (パルス入力方式) のレンジを設定 / 問い合わせします。

構文 :MOTO Tor:TORQue:PRANge {<Nrf>,<Nrf>}

:MOTO Tor:TORQue:PRANge?

<Nrf> = -10000.0000 ~ 10000.0000

例 :MOTO Tor:TORQue:PRANGE 50,-50

:MOTO Tor:TORQue:PRANGE? ->

:MOTO Tor: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 (:MOTO Tor:TORQue:TYPE) が「PULSe(パルス入力)」のときに有効な設定です。

:MOTO Tor:TORQue:RANge

機能 トルク信号 (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 :MOTO Tor:TORQue:RANge {<電圧>}

:MOTO Tor:TORQue:RANge?

<電圧> = 1、2、5、10、20(V)

例 :MOTO Tor:TORQue:RANGE 20V

:MOTO Tor:TORQue:RANGE? ->

:MOTO Tor:TORQue:RANGE 20.0E+00

解説 トルク信号入力タイプ (:MOTO Tor: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?

: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 > }
:MOTor:TORQue:RATE:{UPPer|LOWer}?
<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 トルク信号の入力タイプを設定 / 問い合わせします。

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。

5.17 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?

:NUMeric:FORMat

機能 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。
 - (1) 「ASCIi」のとき
物理値を <NR3> 形式で出力します。(積算経過時間 (TIME) のみ <NR1> 形式)
各項目のデータはカンマ (,) で区切られます。
 - (2) 「FLOat」のとき
数値データブロックの先頭に数バイトのヘッダ (例 "#260", "#3208") 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」(5-87 ページ) を参照してください。

:NUMeric:HSPeed?

機能 高速データ収集モード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:HSPeed?

解説

- 高速データ収集機能 (オプション、/HS) 搭載時だけ有効です。
- 「:NUMeric:HSPeed:ITEM<x>」に関する設定値については、「:NUMeric:HSPeed:NUMber」で設定されている数だけの数値データ項目を出力します。

:NUMeric:HSPeed:CLEar

機能 高速データ収集モードの数値データの出力項目をクリア (「NONE」に設定) します。

構文 :NUMeric:HSPeed:CLEar {ALL|<NRf>[,<NRf>]}
ALL = すべての項目をクリア
1 つ目の <NRf> = 1 ~ 30 (クリアを開始する項目番号)
2 つ目の <NRf> = 1 ~ 30 (クリアを終了する項目番号)

例 :NUMERIC:HSPPEED:CLEAR ALL

解説

- 高速データ収集機能 (オプション、/HS) 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 (30) までの出力項目をクリアします。

:NUMeric:HSPeed:DELeTe

機能 高速データ収集モードの数値データの出力項目を削除します。

構文 :NUMeric:HSPeed:DELeTe {<NRf>[,<NRf>]}
1 つ目の <NRf> = 1 ~ 30 (削除を開始する項目番号)
2 つ目の <NRf> = 1 ~ 30 (削除を終了する項目番号)

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

解説

- 高速データ収集機能 (オプション、/HS) 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:HSPEED:HEADER?

機能	高速データ収集モードの数値データのヘッダを問い合わせます。
構文	:NUMeric:HSPEED:HEADER? {<NRf>} <NRf> = 1 ~ 30 (項目番号)
例	・<NRf> を指定した場合の例 :NUMERIC:HSPEED:HEADER? 1 -> U-E1 ・<NRf> を省略した場合の例 (「:NUMeric:HSPEED:NUMBER」の設定が「3」のとき) :NUMERIC:HSPEED:HEADER? -> U-E1,I-E1,P-E1
解説	・高速データ収集機能 (オプション、/HS) 搭載時だけ有効です。 ・出力項目のデータ名 (ヘッダ) を出力します。 ・<NRf> を指定した場合、その項目番号のデータ名のみを出力します。 ・<NRf> を省略した場合、1 ~ 「:NUMeric:HSPEED:NUMBER」の項目番号のデータ名を順に出力します。

:NUMeric:HSPEED:ITEM<x>

機能	高速データ収集モードの数値データの出力項目 (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:HSPEED:ITEM<x> {NONE <Function>[, <Element>]} :NUMeric:HSPEED:ITEM<x>? <x> = 1 ~ 30 (項目番号) NONE = 出力項目なし <Function> = {U I P SPEED TORQUE PM AUX<x>} (<x> = 1 ~ 2) <Element> = {<NRf> SIGMA SIGMB SIGMC} (<NRf> = 1 ~ 6)
例	:NUMERIC:HSPEED:ITEM1 U,1 :NUMERIC:HSPEED:ITEM1? -> :NUMERIC:HSPEED:ITEM1 U,1
解説	・高速データ収集機能 (オプション、/HS) 搭載時だけ有効です。 ・<Element> を省略したときは、エレメント 1 が設定されます。 ・<Element> が不要なファンクションの応答は、<Element> が省略されます。 ・{SPEED TORQUE PM} は、モータ評価機能 (オプション、/MTR) 搭載時のみ有効です。 ・AUX<x> は、外部信号入力 (オプション、/AUX) 搭載時のみ有効です。

:NUMeric:HSPEED:{MAXimum|MINimum}?

機能	高速データ収集モードの数値データの { 最大値 最小値 } を問い合わせます。
構文	:NUMeric:HSPEED:{MAXimum MINimum}? {<NRf>} <NRf> = 1 ~ 30 (項目番号)
例	・<NRf> を指定した場合の例 :NUMERIC:HSPEED:MAXIMUM? 1 -> 103.79E+00 ・<NRf> を省略した場合の例 (「:NUMeric:HSPEED:NUMBER」の設定が「3」のとき) :NUMERIC:HSPEED:MAXIMUM? -> 103.79E+00,1.0185E+00,105.27E+00
解説	・「:NUMeric:FORMAT」の設定が {FLOAT} の場合の例 :NUMERIC:HSPEED:MAXIMUM? -> #N(N桁のバイト数) (データバイトの並び) ・高速データ収集機能 (オプション、/HS) 搭載時だけ有効です。 ・収集開始時から現在のデータ更新までの中の数値データの { 最大値 最小値 } を出力します。 ・<NRf> を指定した場合、その項目番号の数値データの { 最大値 最小値 } のみを出力します。 ・<NRf> を省略した場合、1 ~ 「:NUMeric:HSPEED:NUMBER」の項目番号の数値データの { 最大値 最小値 } を順に出力します。

:NUMeric:HSPEED:NUMBER

機能	「:NUMeric:HSPEED:VALUE?」で送信される数値データの項目数を設定 / 問い合わせします。
構文	:NUMeric:HSPEED:NUMBER {<NRf>} :NUMeric:HSPEED:NUMBER? <NRf> = 1 ~ 30
例	:NUMERIC:HSPEED:NUMBER 3 :NUMERIC:HSPEED:NUMBER -> :NUMERIC:HSPEED:NUMBER 3
解説	・高速データ収集機能 (オプション、/HS) 搭載時だけ有効です。 ・「:NUMeric:HSPEED:VALUE?」コマンドのパラメータを省略すると、1 ~ (設定値) の項目番号の数値データを 1 つのデータレコードとして、1 回のデータ更新周期で収集した個数のデータレコードを順に出力します。 ・初期設定では、数値データの個数は「3」が設定されています。

5.17 NUMeric グループ

:NUMeric:HSPEED:PRESet

機能	高速データ収集モードの数値データの出力項目を決められたパターンにプリセットします。
構文	:NUMeric:HSPEED:PRESet {DEFAULT RECORD}
	DEFAULT = 初期設定 RECORD = ファイル出力と同じ設定
例	:NUMERIC:HSPEED:PRESET DEFAULT
解説	- 高速データ収集機能 (オプション、/HS) 搭載時だけ有効です。 - 初期設定 (DEFAULT) の内容については、このグループの最後にある「(3) 高速データ収集モードの数値データ出力項目のプリセットパターン」(5-91 ページ) を参照してください。

:NUMeric:HSPEED:VALUE?

機能	高速データ収集モードの数値データを問い合わせます。
構文	:NUMeric:HSPEED:VALUE? {<NRF>} <NRF> = 1 ~ 30 (項目番号)
例	1 回のデータ更新周期で収集したデータレコード数 = 100 の場合 ・ <NRF> を指定した場合の例 :NUMERIC:HSPEED:VALUE? 1 -> 103.79E+00,103.26E+00,... (中略) ..., 103.53E+00 (U1[1],U1[2],... (中略) ...,U1[100]: 100 個のデータ) ・ <NRF> を省略した場合の例 (「:NUMeric:HSPEED:NUMBER」の設定が「3」のとき) :NUMERIC:HSPEED:VALUE? -> 103.79E+00,1.0143E+00,105.27E+00, 103.26E+00,1.0185E+00,105.17E+00, ... (中略) ...,103.53E+00,1.0164E+00, 105.23E+00 (U1[1],I1[1],P1[1],U1[2],I1[2], P1[2],... (中略) ...,U1[100],I1[100], P1[100]:300 = 3×100 個のデータ) ・ 「:NUMeric:FORMat」の設定が {FLOat} の場合の例 :NUMERIC:HSPEED:VALUE? -> #N(N 桁のバイト数) (データバイトの並び) ・ 高速データ収集機能 (オプション、/HS) 搭載時だけ有効です。 ・ 1 回のデータ更新周期で収集した個数のデータレコードを収集した順に出力します。 ・ 1 つの数値データレコードは、以下の数値データで構成されます。 <NRF> を省略した場合、1 ~ 「:NUMeric:HSPEED:NUMBER」の項目番号の数値データ (最大 30 個) で構成されます。 <NRF> を指定した場合、 「:NUMeric:HSPEED:NUMBER」の設定に関係なく、指定された項目番号の数値データ 1 個となります。 ・ ASCII フォーマットの場合、数値データの区切りおよびデータレコードの区切りは、いずれも「カンマ (,)」です。 ・ 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」(5-87 ページ) を参照してください。 ・ データ収集開始前または測定条件変更後は、収集個数が「0」のためデータレコードがありません。応答は以下のようになります。 ASCII フォーマットの場合、応答はありません。 <RMT> のみとなります。 FLOAT フォーマットの場合、「#10」(データバイト数 = 0 を表すヘッダのみ) となります。
解説	

: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。 <pre>:NUMeric:HOLD ON :NUMeric[:NORMal]: ITEM1 URMS,1;ITEM2 IRMS,1;... (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 URMS,2;ITEM2 IRMS,2;... (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) :NUMeric[:NORMal]: ITEM1 URMS,3;ITEM2 IRMS,3;... (エレメント 3 の数値データ項目を設定) :NUMeric[:NORMal]:VALue? (エレメント 3 の数値データを受信) :NUMeric:HOLD OFF</pre> :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?
解説	- 高調波測定機能 (オプション、/G5 または /G6) 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	- 高調波測定機能 (オプション、/G5 または /G6) 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	- 高調波測定機能 (オプション、/G5 または /G6) 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。

5.17 NUMeric グループ

: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|SIGMC} (<Nrf> = 1 ~ 6)

例 :NUMERIC:LIST:ITEM1 U,1
:NUMERIC:LIST:ITEM1? ->
:NUMERIC:LIST:ITEM1 U,1

解説

- 高調波測定機能 (オプション、/G5 または /G6) 搭載時のみ有効です。
- <Function> の選択肢については、ファンクション選択肢一覧 (2)、5-41 ページを参照してください。

: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

- 高調波測定機能 (オプション、/G5 または /G6) 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 ~ 500 (ALL)

例 :NUMERIC:LIST:ORDER 100
:NUMERIC:LIST:ORDER? ->
:NUMERIC:LIST:ORDER 100

解説 高調波測定機能 (オプション、/G5 または /G6) 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

- 高調波測定機能 (オプション、/G5 または /G6) 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」(5-90 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

- 高調波測定機能 (オプション、/G5 または /G6) 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 502 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 502*5 = 2510 個のデータ)
- ・ 「:NUMeric:FORMat」の設定が {FLOat} の場合
の例
:NUMERIC:LIST:VALUE? -> #N(N桁のバイ
ト数) (データバイトの並び)

解説

- ・ 高調波測定機能 (オプション、/G5 または /G6)
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 502
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 502 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 502*
「: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」(5-87 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 :NUMeric:NORMal?

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 :NUMeric[:NORMal]:CLEar {ALL|
<NRf>[,<NRf>]}

ALL = すべての項目をクリア
1 つ目の <NRf> = 1 ~ 255 (クリアを開始する
項目番号)
2 つ目の <NRf> = 1 ~ 255 (クリアを終了する
項目番号)

例 :NUMERIC:NORMAL:CLEAR ALL

解説 2 つ目の <NRf> を省略した場合、クリア開始番
号から最後 (255) までの出力項目をクリアしま
す。

:NUMeric[:NORMal]:DELeTe

機能 数値データの出力項目を削除します。

構文 :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)

例 :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)

解説

- ・ 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。

5.17 NUMeric グループ

:NUMeric[:NORMal]:ITEM<x>

機能 数値データの出力項目（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 :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> = {URMS|IRMS|P|S|Q|...}
<Element> = {<Nrf>|SIGMa|SIGMB|SIGMC}
(<Nrf> = 1 ~ 6)
<Order> = {TOTal|DC|<Nrf>}
(<Nrf> = 1 ~ 500)

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

解説

- ・ <Function> の選択肢については、「DISPlay グループ」のファンクション選択肢一覧 (1)、5-38 ページを参照してください。
- ・ <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。

:NUMeric[:NORMal]:NUMBER

機能 「:NUMeric[:NORMal]:VALue?」で送信される数値データの個数を設定 / 問い合わせします。

構文 :NUMeric[:NORMal]:NUMBER {<Nrf>|ALL}
:NUMeric[:NORMal]:NUMBER?
<Nrf> = 1 ~ 255 (ALL)

例 :NUMERIC:NORMAL:NUMBER 15
:NUMERIC:NORMAL:NUMBER ->
:NUMERIC:NORMAL:NUMBER 15

解説

- ・ 「:NUMeric[:NORMal]:VALue?」でパラメータを省略すると、1 ~ (設定値) の数値データを順に出力します。
- ・ 初期設定では、数値データの個数は「15」が設定されています。

:NUMeric[:NORMal]:PRESet

機能 数値データの出力項目を決められたパターンにプリセットします。

構文 :NUMeric[:NORMal]:PRESet {<Nrf>}
<Nrf> = 1 ~ 4

例 :NUMERIC:NORMAL:PRESET 1

解説

- ・ プリセットされる出力項目の内容については、このグループの最後にある「(1) 数値データ出力項目のプリセットパターン」(5-88 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 ->
103.79E+00
- ・ <Nrf> を省略した場合の例
:NUMERIC:NORMAL:VALUE? -> 103.79E+00
,1.0143E+00,105.27E+00,...
(中略)...,1.428E+00
- ・ 「:NUMeric:FORMat」の設定が {FLOat} の場合の例
:NUMERIC:NORMAL:VALUE? -> #N(N 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」(5-87 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

- 電力値 (P、S、Q) の Σ
- 積算値 (WH、WHP、WHM、AH、AHP、AHM、WS、WQ)
- 効率 (ETA1、ETA2、ETA3、ETA4)、含有率 (UHDFk、IHDFk、PHDFk)、ひずみ率 (UTHD、ITHD、PTHD)
ASCII: <NR3> 形式 (仮数部: 有効最大 6 桁、指数部: 2 桁、例: [-]123.456E+00)
FLOAT: IEEE 単精度浮動小数点 (4byte) 形式
- 積算経過時間 (TIME)
ASCII: 秒単位の <NR1> 形式 (例: 1 時間 (1:00:00) の場合、3600)
FLOAT: IEEE 単精度浮動小数点 (4byte) 形式で秒単位 (例: 1 時間 (1:00:00) の場合、0x45610000)
- ユーザー定義イベント (EV1 ~ EV8)
ASCII: 成立 / 不成立時にそれぞれ設定されている文字列 (初期値は " True" / " False")
FLOAT: 成立時 0x3F800000(1)、不成立時 0x00000000(0)
- 項目無し (NONE)
ASCII: 「NAN」 (Not A Number)
FLOAT: 0x7E951BEE(9.91E+37)
- 上記以外
ASCII: <NR3> 形式 (仮数部: 有効最大 5 桁、指数部: 2 桁、例: [-]123.45E+00)
FLOAT: IEEE 単精度浮動小数点 (4byte) 形式

(2) 異常時のデータ

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

Note

- エレメント 1 ~ 6 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 -180.00 ~ 180.00 の範囲で出力します。
- 電力値 (P、S、Q) 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 6 桁となる場合があります。電力レンジの一覧表がスタートガイド IM WT1801-03JA に掲載されていますので、参照してください。
- 効率 (ETA1、ETA2、ETA3、ETA4)、含有率 (UHDFk、IHDFk、PHDFk)、ひずみ率 (UTHD、ITHD、PTHD) は、小数点以下 3 桁固定です。値が 100[%] 以上になると、仮数部の桁数が 6 桁となります。

※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名 <Function> と、本機器のメニュー画面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、パターン 1 の項目番号 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>
1	URMS	1
2	IRMS	1
3	P	1
4	S	1
5	Q	1
6	LAMBda	1
7	PHI	1
8	FU	1
9	FI	1
10	NONE	
11 ~ 19	URMS ~ FI	2
20	NONE	
21 ~ 29	URMS ~ FI	3
30	NONE	
31 ~ 39	URMS ~ FI	4
40	NONE	
41 ~ 49	URMS ~ FI	5
50	NONE	
51 ~ 59	URMS ~ FI	6
60	NONE	
61 ~ 69	URMS ~ FI	SIGMA
70	NONE	
71 ~ 79	URMS ~ FI	SIGMB
80	NONE	
81 ~ 89	URMS ~ FI	SIGMC
90	NONE	
91 ~ 255	NONE	

パターン 2

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

16 ~ 30	URMS ~ FI	2
31 ~ 45	URMS ~ FI	3
46 ~ 60	URMS ~ FI	4
61 ~ 75	URMS ~ FI	5
76 ~ 90	URMS ~ FI	6
91 ~ 105	URMS ~ FI	SIGMA
106 ~ 120	URMS ~ FI	SIGMB
121 ~ 135	URMS ~ FI	SIGMC
136 ~ 255	NONE	

パターン 3

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

パターン 4

ITEM<x>	<Function>	<Element>
1	URMS	1
2	UMN	1
3	UDC	1
4	UAC	1
5	IRMS	1
6	IMN	1
7	IDC	1
8	IAC	1
9	P	1
10	S	1

5.17 NUMeric グループ

11	Q	1
12	FU	1
13	FI	1
14	TIME	1
15	WH	1
16	WHP	1
17	WHM	1
18	AH	1
19	AHP	1
20	AHM	1
21 ~ 40	URMS ~ AHM	2
41 ~ 60	URMS ~ AHM	3
61 ~ 80	URMS ~ AHM	4
81 ~ 100	URMS ~ AHM	5
101 ~ 120	URMS ~ AHM	6
121 ~ 140	URMS ~ AHM	SIGMA
141 ~ 160	URMS ~ AHM	SIGMB
161 ~ 180	URMS ~ AHM	SIGMC
181 ~ 255	NONE	

(2) 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 ~ 15	U ~ P	5
16 ~ 18	U ~ P	6
19 ~ 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 ~ 25	U ~ PHII	5
26 ~ 30	U ~ PHII	6
31 ~ 64	NONE	

パターン 3

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 ~ 18	U ~ PHII	2
19 ~ 27	U ~ PHII	3
28 ~ 36	U ~ PHII	4
37 ~ 45	U ~ PHII	5
46 ~ 54	U ~ PHII	6
55 ~ 64	NONE	

パターン 4

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 ~ 45	U ~ XP	5
46 ~ 54	U ~ XP	6
55 ~ 64	NONE	

(3) 高速データ収集モードの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:HSPeed:PRESet」

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 ~ 15	U ~ P	5
16 ~ 18	U ~ P	6
19 ~ 21	U ~ P	SIGMA
22 ~ 24	U ~ P	SIGMB
25 ~ 27	U ~ P	SIGMC
28 ~ 30	NONE	

5.18 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 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。

:RATE

機能	データ更新レートを設定 / 問い合わせします。
構文	:RATE {<時間>} :RATE? <時間> = 50、100、200、500 (ms)、1、2、5、 10、20 (s)
例	:RATE 500MS :RATE? -> :RATE 500.0E-03
解説	波形表示を選択し、トリガモードを Auto/Normal に設定した場合は、データ更新周期はトリガの動 作に依存します。

5.19 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、第 6 章をご覧ください。

: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?

: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、6 章のステータスレポートをご覧ください。

:STATus:EESE

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:EESE <Register>
:STATus:EESE?
<Register> = 0 ~ 65535

例 :STATUS:EESE #B00000000000000000
:STATus:EESE? ->
:STATUS:EESE 0

解説 拡張イベントイネーブルレジスタについては、6 章のステータスレポートをご覧ください。

:STATus:EESR?

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、6 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, "Noerror"」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、6 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?

機能 シリアルポールを実行します。

構文 :STATus:SPOLl?

例 :STATUS:SPOLL? -> :STATUS:SPOLL 0

5.20 STORE グループ

STORE グループは、ストアに関するグループです。

フロントパネルの STORE START、STORE STOP、STORE RESET(SHIFT + STORE STOP)、STORE SET(SHIFT + STORE START) 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。

:STORE?

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?

:STORE:COUNT

機能 スタア回数を設定 / 問い合わせします。

構文 :STORE:COUNT {<NRf>|INFinite}

:STORE:COUNT?

<NRf> = 1 ~ 9999999

INFinite = 限度なし

例 :STORE:COUNT 100

:STORE:COUNT? ->

:STORE:COUNT 100

:STORE:FILE?

機能 スタ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?

:STORE:FILE:ANAMing

機能 スタアされた数値データの保存ファイル名の自動生成機能を設定 / 問い合わせします。

構文 :STORE:FILE:ANAMing {OFF|

NUMBERing|DATE}

:STORE:FILE:ANAMing?

例 :STORE:FILE:ANAMING NUMBERING

:STORE:FILE:ANAMING? ->

:STORE:FILE:ANAMING NUMBERING

:STORE:FILE:CDIRECTory

機能 スタアされた数値データの保存先ディレクトリを変更します。

構文 :STORE:FILE:CDIRECTory {<文字列>}

<文字列> = ディレクトリ名

例 :STORE:FILE:CDIRECTORY "STORE"

解説 上のディレクトリに移動するには、"." を指定します。

:STORE:FILE:CONVert?

機能 スタアされた数値データファイルの CSV 形式への変換に関するすべての設定値を問い合わせます。

構文 :STORE:FILE:CONVert?

:STORE:FILE:CONVert:ABORt

機能 スタアされた数値データファイルの CSV 形式への変換を中止します。

構文 :STORE:FILE:CONVert:ABORt

例 :STORE:FILE:CONVERT:ABORT

:STORE:FILE:CONVert:AUTO

機能 スタアされた数値データファイルの CSV 形式への自動変換 ON/OFF を設定 / 問い合わせします。

構文 :STORE:FILE:CONVert:AUTO {<Boolean>}

:STORE:FILE:CONVert:AUTO?

例 :STORE:FILE:CONVERT:AUTO ON

:STORE:FILE:CONVERT:AUTO? ->

:STORE:FILE:CONVERT:AUTO 1

:STORE:FILE:CONVert:EXECute

機能 スタアされた数値データファイルの CSV 形式への変換を実行します。

構文 :STORE:FILE:CONVert:

EXECute {<文字列>}

<文字列> = ファイル名

例 :STORE:FILE:CONVERT:EXECUTE "STORE1"

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。

:STORE:FILE:DRIVE

機能 スタアされた数値データの保存先ドライブを設定します。

構文 :STORE:FILE:DRIVE {RAM|USB[, <NRf>]|

NETWork}

RAM = 内蔵 RAM ドライブ

USB = USB メモリドライブ、<NRf> = 0, 1 (ドライブ番号)

NETWork = ネットワークドライブ

例 :STORE:FILE:DRIVE RAM

:STORE:FILE:FREE?

機能 スタアされた数値データの保存先ドライブの空き容量 (byte) を問い合わせます。

構文 :STORE:FILE:FREE?

例 :STORE:FILE:FREE? -> 20912128

:STORe:FILE:NAME

機能 ストアされた数値データの保存ファイル名を設定 / 問い合わせします。

構文 :STORe:FILE:NAME {<文字列>}
:STORe:FILE:NAME?
<文字列> = ファイル名

例 :STORe:FILE:NAME "STORE1"
:STORe:FILE:NAME? ->
:STORe:FILE:NAME "STORE1"

:STORe:FILE:PATH?

機能 ストアされた数値データの保存先を絶対パスで問い合わせます。

構文 :STORe:FILE:PATH?
例 :STORe:FILE:PATH? -> "USB-0/STORE"

: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

解説 ストアモード (:STORe:SMODE) が {MANual|RTIME|INTEGRate} のときに有効な設定です。

:STORe:NUMeric?

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?

:STORe:NUMeric:ITEM

機能 数値データのストア項目の選択方式を設定 / 問い合わせします。

構文 :STORe:NUMeric:ITEM {DISPlayed|SELEcted}
:STORe:NUMeric:ITEM?

DISPlayed = 画面表示項目自動選択方式
SELEcted = マニュアル選択方式

例 :STORe:NUMERIC:ITEM SELECTED
:STORe:NUMERIC:ITEM? ->
:STORe:NUMERIC:ITEM SELECTED

解説 選択肢の意味は、次のとおりです。
DISPlayed = 画面に表示されている数値項目をファイル保存
SELEcted = 「:STORe:NUMeric:NORMAL:…」以下のコマンドで個別設定された数値項目をファイル保存

:STORe:NUMeric:NORMAL?

機能 数値データのストア項目 (マニュアル選択方式) に関するすべての設定値を問い合わせます。

構文 :STORe:NUMeric:NORMAL?

解説 ストア項目の選択方式 (:STORe:NUMeric:ITEM) がマニュアル選択方式 (SELEcted) のときに有効な設定です。

: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|SIGMC}**

機能 数値データをストアするときの各エレメント | 結線ユニット {Σ A|Σ B|Σ C} の出力 ON/OFF を設定 / 問い合わせします。

構文 :STORe:NUMeric[:NORMAL]:{ELEMENT<x>|SIGMA|SIGMB|SIGMC} {<Boolean>}
:STORe:NUMeric[:NORMAL]:{ELEMENT<x>|SIGMA|SIGMB|SIGMC}?
<x> = 1 ~ 6

例 :STORe:NUMERIC:NORMAL:ELEMENT1 ON
:STORe:NUMERIC:NORMAL:ELEMENT1? ->
:STORe:NUMERIC:NORMAL:ELEMENT1 1

: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> = {URMS|IRMS|P|S|Q|...}

例 :STORe:NUMERIC:NORMAL:URMS ON
:STORe:NUMERIC:NORMAL:URMS? ->
:STORe:NUMERIC:NORMAL:URMS 1

解説 <Function> の選択肢については、「DISPlay グループ」のファンクション選択肢一覧 (1)、5-38 ページを参照してください。

5.20 STORE グループ

: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項目設定パターンについては、ユーザーズマニュアル[機能編]IM WT1801-01JA を参照してください。

:STORE:RESet

機能 数値データのストアを初期化します。

構文 :STORE:RESet

例 :STORE:RESet

:STORE:RTIME?

機能 実時間モードのストア開始 / 終了予約時刻を問い合わせます。

構文 :STORE:RTIME?

:STORE:RTIME:{START|END}

機能 実時間モードのストア { 開始 | 終了 } 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>,<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 2011,1,1,0,0,0
:STORE:RTIME:START? ->
:STORE:RTIME:START 2011,1,1,0,0,0

解説 ストアモード (:STORE:SMODE) が RTIME のときに有効な設定です。

:STORE:SASStart

機能 ストア開始時の数値データをストアするかどうかを設定 / 問い合わせします。

構文 :STORE:SASStart {<Boolean>}
:STORE:SASStart?

例 :STORE:SASStart OFF
:STORE:SASStart? ->
:STORE:SASStart 0

解説 ストアモード (:STORE:SMODE) が {MANual|RTIME|INTEGrate}{(MANual|RTIME) は、ストアインターバルが 0 以外) のときに有効な設定です。

:STORE:SMODE

機能 ストアモードを設定 / 問い合わせします。

構文 :STORE:SMODE {MANual|RTIME|INTEGrate|EVENT|SINGLE}
:STORE:SMODE?
MANual = マニュアルストアモード
RTIME = 実時間制御ストアモード
INTEGrate = 積算同期ストアモード
EVENT = イベント同期ストアモード
SINGLE = シングルショットストアモード

例 :STORE:SMODE MANUAL
:STORE:SMODE? ->
:STORE:SMODE MANUAL

:STORE:START

機能 数値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|EVENT} のときはストア待機状態になります。SINGLE のときはストアを実行します。

:STORE:STATE?

機能 ストア状態を問い合わせます。

構文 :STORE:STATE?

例 :STORE:STATE? -> RESET

解説
・ 応答の内容は次のとおりです。
RESet = ストアリセット状態
READY = ストア待機中
START = ストア実行中
STOP = ストアストップ状態
COMPLETE = ストア完了 (またはエラー発生による終了)
CONVert = CSV 形式ファイルへの変換中
CLOSe = WTS/HDS 形式ファイルの終了処理中

:STORE:STOP

機能 数値データのストアを終了します。

構文 :STORE:STOP

例 :STORE:STOP

:STORE:TEVent

機能 イベント同期ストアモードのトリガとなるイベントを設定 / 問い合わせします。

構文 :STORE:TEVent {<NRf>}
:STORE:TEVent?
<NRf> = 1 ~ 8 (イベント番号)

例 :STORE:TEVENT 1
:STORE:TEVENT? ->
:STORE:TEVENT 1

解説 ストアモード (:STORE:SMODE) が EVENT のときに有効な設定です。

5.21 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 UTILITY キーの System Config 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。

: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?

: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?

: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 / 時刻の設定に関するすべての設定値を問い合わせます。

構文 :SYSTem:CLOCK:SNTP?

:SYSTem:CLOCK:SNTP[:EXECute]

機能 SNTP による日付 / 時刻の設定を実行します。

構文 :SYSTem:CLOCK:SNTP[:EXECute]

例 :SYSTEM:CLOCK:SNTP:EXECUTE

: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"

: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

: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 "11/01/01"
:SYSTEM:DATE? -> "11/01/01"

解説 「年」は西暦の下 2 桁です。

:SYSTem:DFlow:FREQuency

機能 低周波数入力 (無入力) 時の周波数データの表示方法を設定 / 問い合わせします。

構文 :SYSTem:DFlow:FREQuency {0|ERRor}
:SYSTem:DFlow:FREQuency?

例 :SYSTEM:DFlow:FREQUENCY ERROR
:SYSTEM:DFlow:FREQUENCY? ->
:SYSTEM:DFlow:FREQUENCY ERROR

:SYSTem:DFlow:MOTor

機能 パルス無入力時のモータデータの表示方法を設定 / 問い合わせします。

構文 :SYSTem:DFlow:MOTor {0|ERRor}
:SYSTem:DFlow:MOTor?

例 :SYSTEM:DFlow:MOTOR ERROR
:SYSTEM:DFlow:MOTOR? ->
:SYSTEM:DFlow:MOTOR ERROR

解説 モータ評価機能 (オプション、/MTR) 搭載時のみ有効です。

:SYSTem:DPOint

機能 各種データをアスキー (CSV) 形式で保存する場合の小数点の種類を設定 / 問い合わせします。

構文 :SYSTem:DPOint {PERiod|COMma}
:SYSTem:DPOint?

例 :SYSTEM:DPOINT PERIOD
:SYSTEM:DPOINT? ->
:SYSTEM:DPOINT PERIOD

: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

5.21 SYSTem グループ

:SYSTem:FONT

機能 メニューおよびメッセージのフォントの大きさを設定 / 問い合わせします。

構文 :SYSTem:FONT {SMALL|LARGE}
:SYSTem:FONT?

例 :SYSTEM:FONT LARGE
:SYSTEM:FONT? ->
:SYSTEM:FONT LARGE

: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?

: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|CHINEse|GERMan}
: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|CHINEse|GERMan}
:SYSTem:LANGuage:MESSAge?

例 :SYSTEM:LANGUAGE:MESSAGE ENGLISH
:SYSTEM:LANGUAGE:MESSAGE? ->
:SYSTEM:LANGUAGE:MESSAGE ENGLISH

:SYSTem:LCD?

機能 液晶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?

:SYSTem:LCD:AOff?

機能 バックライトのオートオフ機能に関するすべての設定値を問い合わせます。

構文 :SYSTem:LCD:AOff?

:SYSTem:LCD:AOff[:STATe]

機能 バックライトのオートオフ機能の ON/OFF を設定 / 問い合わせします。

構文 :SYSTem:LCD:AOff[:STATe] {<Boolean>}
:SYSTem:LCD:AOff:STATe?

例 :SYSTEM:LCD:AOff:STATE ON
:SYSTEM:LCD:AOff:STATE? ->
:SYSTEM:LCD:AOff:STATE 1

:SYSTem:LCD:AOff:TIME

機能 バックライトのオートオフまでの時間を設定 / 問い合わせします。

構文 :SYSTem:LCD:AOff:TIME {<Nrf>}
:SYSTem:LCD:AOff:TIME?
<Nrf> = 1 ~ 60 (分)

例 :SYSTEM:LCD:AOff:TIME 5
:SYSTEM:LCD:AOff:TIME? ->
:SYSTEM:LCD:AOff:TIME 5

:SYSTem:LCD:BRIGhtness

機能 液晶画面の輝度を設定 / 問い合わせします。

構文 :SYSTem:LCD:BRIGhtness {<Nrf>}
:SYSTem:LCD:BRIGhtness?
<Nrf> = 1 ~ 10

例 :SYSTEM:LCD:BRIGHTNESS 7
:SYSTEM:LCD:BRIGHTNESS? ->
:SYSTEM:LCD:BRIGHTNESS 7

:SYSTem:LCD:COLor?

機能 液晶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?

:SYSTem:LCD:COLor:BASEcolor

機能 画面 (メニュー) の基本色を設定 / 問い合わせします。

構文 :SYSTem:LCD:COLor:BASEcolor {BLUE|
GRAY}
:SYSTem:LCD:COLor:BASEcolor?

例 :SYSTEM:LCD:COLOR:BASECOLOR BLUE
:SYSTEM:LCD:COLOR:BASECOLOR? ->
:SYSTEM:LCD:COLOR:BASECOLOR BLUE

:SYSTem:LCD:COLor:GRAPh?

機能 波形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?

:SYSTem:LCD:COLOr:GRAPh:CHANnel<x>

機能	各波形の表示色を設定 / 問い合わせします。
構文	:SYSTem:LCD:COLOr:GRAPh:CHANnel<x> {YELLow GREen MAGenta CYAN RED ORANge LBLue PURPle BLUE PINK LGRen DBLue BGRen SPINk MGRen GRAY} :SYSTem:LCD:COLOr:GRAPh:CHANnel<x>? <x> = 1 ~ 16 (波形チャンネル) YELLow = 黄 GREen = 緑 MAGenta = マゼンタ CYAN = シアン RED = 赤 ORANge = オレンジ LBLue = 明るい青 (Light Blue) PURPle = 紫 BLUE = 青 PINK = ピンク LGRen = 明るい緑 (Light Green) DBLue = 暗い青 (Dark Blue) BGRen = 青緑 (Blue Green) SPINk = サーモン (Salmon Pink) MGRen = 薄い緑 (Mild Green) GRAY = グレー
例	:SYSTem:LCD:COLOr:GRAPh: CHANNEL1 YELLOW :SYSTem:LCD:COLOr:GRAPh: CHANNEL1? -> :SYSTem:LCD:COLOr:GRAPh: CHANNEL1 YELLOW

:SYSTem:LCD:COLOr:GRAPh:PRESet

機能	波形の表示色を決められたパターンにプリセットします。
構文	:SYSTem:LCD:COLOr:GRAPh:PRESet {DEFault CLASSic}
例	:SYSTem:LCD:COLOr:GRAPh:PRESet DEFAULT

:SYSTem:LCD:COLOr:INTENsity:GRID

機能	グリッドの輝度を設定 / 問い合わせします。
構文	:SYSTem:LCD:COLOr:INTENsity: GRID {<NRf>} :SYSTem:LCD:COLOr:INTENsity:GRID? <NRf> = 1 ~ 8 (グリッドの輝度)
例	:SYSTem:LCD:COLOr:INTENsity:GRID 4 :SYSTem:LCD:COLOr:INTENsity:GRID? -> :SYSTem:LCD:COLOr:INTENsity:GRID 4

:SYSTem:LCD[:STATe]

機能	バックライトの ON/OFF を設定 / 問い合わせします。
構文	:SYSTem:LCD[:STATe] {<Boolean>} :SYSTem:LCD:STATe?
例	:SYSTem:LCD:STATe ON :SYSTem:LCD:STATe? -> :SYSTem:LCD:STATe 1

:SYSTem:MODeL?

機能	モデルコード (形名) を問い合わせます。
構文	:SYSTem:MODeL?
例	:SYSTem:MODeL? -> :SYSTem:MODeL "WT1806"
解説	UTILITY → System Overview 画面の Model 項目を返します。

:SYSTem:RESolution

機能	数値データの表示分解能を設定 / 問い合わせします。
構文	:SYSTem:RESolution {<NRf>} :SYSTem:RESolution? <NRf> = 4、5 (digit)
例	:SYSTem:RESolution 5 :SYSTem:RESolution? -> :SYSTem:RESolution 5

:SYSTem:SERial?

機能	シリアル番号を問い合わせます。
構文	:SYSTem:SERial?
例	:SYSTem:SERial? -> :SYSTem:SERial "123456789"
解説	UTILITY → System Overview 画面の No. 項目の文字列を返します。

:SYSTem:SUFFix?

機能	サフィックス (仕様コード) を問い合わせます。
構文	:SYSTem:SUFFix?
例	:SYSTem:SUFFix? -> :SYSTem:SUFFix "-33-M-HE/EX6/B5/G6/ DT/FQ/V1/DA/MTR"
解説	UTILITY → System Overview 画面の Suffix 項目の文字列を返します。

:SYSTem:TIME

機能	時刻を設定 / 問い合わせします。
構文	:SYSTem:TIME {<文字列>} :SYSTem:TIME? <文字列> = "HH:MM:SS" (HH = 時間、MM = 分、SS = 秒)
例	:SYSTem:TIME "14:30:00" :SYSTem:TIME? -> "14:30:00"

:SYSTem:USBKeyboard

機能	USB キーボードの種類を設定 / 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

5.22 WAVEform グループ

WAVEform グループは、取り込んだ波形データ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?

: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}

例 :WAVEform:BYTeorder?

:WAVEFORM:BYTEORDER MSBFIRST

:WAVEFORM:BYTEORDER? ->

:WAVEFORM:BYTEORDER MSBFIRST

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 ~ 1601

例 :WAVEform:END 1601

:WAVEFORM:END? ->

:WAVEFORM:END 1601

: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? -> 1602

解説 データ点数は固定です。常に「1602」を返します。

5.22 WAVEform グループ

: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。

: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? -> 32.000E+03

:WAVEform:STARt

機能	「:WAVEform:SEND?」で送信される波形表示デー タの出力開始点を設定 / 問い合わせします。
構文	:WAVEform:STARt {<NRf>} :WAVEform:STARt?
例	<NRf> = 0 ~ 1601 :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 AUX<x>} :WAVEform:TRACe?
例	U<x>、I<x>の<x> = 1 ~ 6 (エレメント) AUX<x>の<x> = 1 ~ 2 (AUX入力チャネル) :WAVEFORM:TRACE U1 :WAVEFORM:TRACE? -> :WAVEFORM:TRACE U1
解説	・{SPEed TORQue}は、モータ評価機能(オプション、/MTR)搭載時のみ選択可能です。 ・AUX<x>は、外部信号入力(オプション、/ AUX)搭載時のみ選択可能です。

: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。

5.23 共通コマンドグループ

共通コマンドグループは、IEEE 488.2-1992 で規定されている、機器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?

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

機能	標準イベントレジスタ、拡張イベントレジスタ、エラーキューをクリアします。
構文	*CLS
例	*CLS
解説	• *CLS コマンドがプログラムメッセージターミネータのすぐ後ろにあるときは、出力キューもクリアされます。 • 各レジスタ、キューについては、6 章を参照してください。

*ESE

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?
例	<NRf> = 0 ~ 255 *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、6-4 ページを参照してください。

*ESR?

機能	標準イベントレジスタの値を問い合わせ、同時にクリアします。
構文	*ESR?
例	*ESR? -> 32
解説	• 各ビットの 10 進数の和が返されます。 • SRQ が発生しているときに、どんな種類のイベントが起こったかを調べることができます。 • たとえば、「32」が返されると、標準イベントレジスタが「00100000」にセットされていることを示します。つまり、「コマンド文法エラー」が起こったために SRQ が発生したことがわかります。 • *ESR? で問い合わせると、標準イベントレジスタの内容がクリアされます。 • 標準イベントレジスタについては、6-4 ページを参照してください。

*IDN?

機能	機種を問い合わせます。
構文	*IDN?
例	*IDN? -> YOKOGAWA,WT1806-06,123456789,F1.01
解説	• <製造者>、<機種>、<シリアル No.>、<ファームウェアのバージョン> の形式で返されます。 • <機種> は「形名 (6 桁) - エLEMENT 構成 (2 桁)」という形式になります。形名と ELEMENT 構成については、スタートガイド IM WT1801-03JA の梱包内容の確認をご覧ください。

*OPC

機能	指定したオーバーラップコマンドが終了したときに、標準イベントレジスタのビット 0 (OPC ビット) を 1 にセットします。
構文	*OPC
例	*OPC
解説	• *OPC を使った同期のとり方については、4-8 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?

機能	指定したオーバーラップコマンドが終了していれば、ASCII コードの「1」を返します。
構文	*OPC?
例	*OPC? -> 1
解説	・*OPC? を使った同期のとり方については、4-8 ページを参照してください。 ・オーバーラップコマンドの指定は「COMMunicate:OPSE」で行います。 ・メッセージの最後でない *OPC? の動作は保証されません。

*OPT?

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? -> EX6,B5,G5,DT,FQ,V1,DA,MTR
解説	・外部電流センサ入力 (EX1、WT1801 用)、外部電流センサ入力 (EX2、WT1802 用)、外部電流センサ入力 (EX3、WT1803 用)、外部電流センサ入力 (EX4、WT1804 用)、外部電流センサ入力 (EX5、WT1805 用)、外部電流センサ入力 (EX6、WT1806 用)、内蔵プリンタ (B5)、高調波測定 (G5)、2 系統同時高調波測定 (G6)、デルタ演算 (DT)、周波数測定追加 (FQ)、RGB 出力 (V1)、20 チャネル DA 出力 (DA)、モータ評価機能 (MTR)、外部信号入力 (AUX)、および高速データ収集 (HS) 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。

*RST

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

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、6-3 ページを参照してください。

*STB?

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。 ・シリアルボールを実行せずにレジスタを読むので、ビット 6 は RQS ではなく MSS ビットになります。 ・たとえば、「4」が返されると、ステータスバイトレジスタが「00000100」にセットされていることを示します。つまり、「エラーキューが空でない」(エラーが発生した)ことがわかります。 ・*STB? で問い合わせても、ステータスバイトレジスタの内容はクリアされません。 ・ステータスバイトレジスタについては、6-3 ページを参照してください。

*TRG

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。

5.23 共通コマンドグループ

*TST?

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。また、本機器の状態（積算中、ストア中）により、セルフテストが実行できなかったときは、対応するエラーコードを返します。 - テストが終了するまで約 30 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。

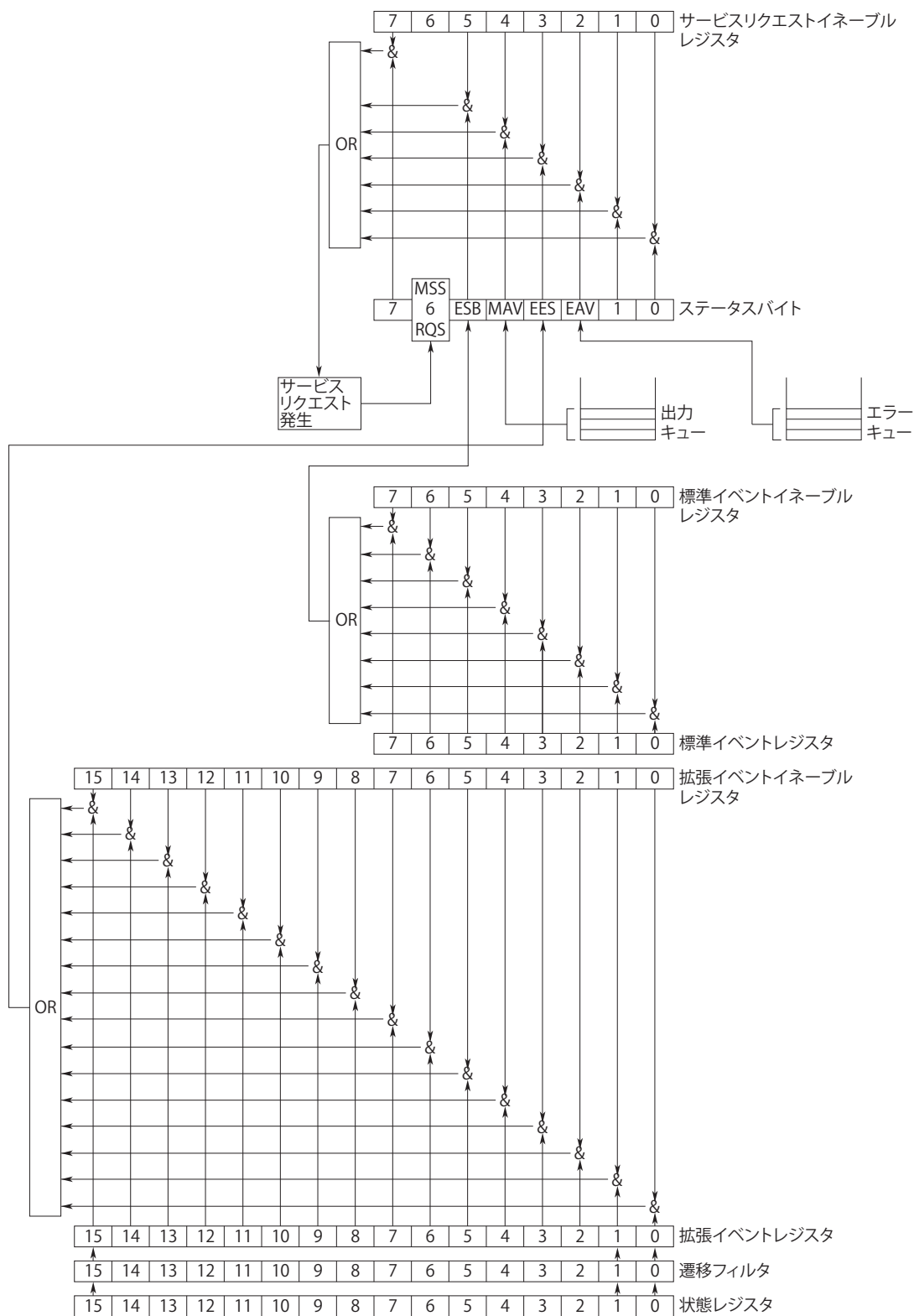
*WAI

機能	指定したオーバラップコマンドが終了するまで、*WAI に続く命令を待ちます。
構文	*WAI
例	*WAI
解説	*WAI を使った同期のとり方については、4-8 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。

6.1 ステータスレポートについて

ステータスレポート

シリアルポー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。



6.1 ステータスレポートについて

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。

レジスタ	影響を与えるステータスバイトのビット
標準イベントレジスタ	ビット 5(ESB) を 1/0 にセット
出力キュー	ビット 4(MAV) を 1/0 にセット
拡張イベントレジスタ	ビット 3(EES) を 1/0 にセット
エラーキュー	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。

マスクされるレジスタ	マスクするレジスタ
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
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
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

各レジスタの書き込み / 読み出し

たとえば、標準イベントイネーブルレジスタの各ビットを1または0にするには、*ESE コマンドを使います。また、標準イベントイネーブルレジスタの各ビットが1であるか0であるかを確認するには、*ESE? コマンドを使います。これらの各コマンドについては、第5章で詳しく説明しています。

6.2 ステータスバイト

ステータスバイト

	RQS						
7	6	ESB	MAV	EES	EAV	1	0
	MSS						

- **ビット 0、1、7**
未使用 (常に 0)
- **ビット 2 EAV(Error Available)**
エラーキューが空でないときに 1 にセットされます。つまり、エラーが発生すると 1 になります。6-6 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、拡張イベン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、機器の内部であるイベントが起こったときに 1 になります。6-5 ページを参照してください。
- **ビット 4 MAV(Message Available)**
出力キューが空でないときに 1 にセットされます。つまり、問い合わせを行って出力すべきデータがあるときに 1 になります。6-6 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、標準イベン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、機器の内部であるイベントが起こったときに 1 になります。6-4 ページを参照してください。
- **ビット 6 RQS(Request Service)/MSS(MasterStatus 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、シリアルポールされたときか、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、第 5 章をお読みください。

ステータスバイトの動作

ステータスバイトのビット 6 が 1 になると、サービスリクエストが発生します。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和が 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。
- **シリアルポール**
シリアルポールを実行すると、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。シリアルポールでは 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。
- **シリアルポールを実行したとき**
RQS ビットだけがクリアされます。
- ***CLS コマンドを受信したとき**
*CLS コマンドを受信すると、ステータスバイト自体はクリアされませんが、各ビットに影響する標準イベントレジスタなどの内容がクリアされます。その結果、それに対応したステータスバイトのビットがクリアされます。ただし、出力キューは *CLS コマンドではクリアできないので、ステータスバイトのビット 4(MAV) は影響を受けません。ただし、*CLS コマンドをプログラムメッセージターミネータのすぐあとに受信したときは、出力キューもクリアされます。

6.3 標準イベントレジスタ

標準イベントレジスタ

7	6	5	4	3	2	1	0
PON	URQ	CME	EXE	DDE	QYE	RQC	OPC

- **ビット 7 PON(Power ON) 電源 ON**
本機器の電源が ON になったときに、1 になります。
- **ビット 6 URQ(User Request) ユーザーリクエスト**
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 (第 5 章参照) によって指定された動作が終了したときに、1 になります。

各ビットのマスク

標準イベントレジスタのあるビットをマスクして、ステータスバイトのビット 5(ESB) の要因にしたいときには、標準イベントイネーブルレジスタの対応するビットを 0 にします。

たとえば、ビット 2(QYE) をマスクして問い合わせエラーが発生しても ESB を 1 にしないようにするには、標準イベントイネーブルレジスタのビット 2 を 0 にします。これは *ESE コマンドで行います。また、標準イベントイネーブルレジスタの各ビットが 1 であるか 0 であるかは、*ESE? で問い合わせられます。*ESE コマンドについては、第 5 章をお読みください。

標準イベントレジスタの動作

標準イベントレジスタは、機器の内部に起こった 8 種類のイベントに対するレジスタです。どれかのビットが 1 になると、ステータスバイトのビット 5(ESB) を 1 にセットします (標準イベントイネーブルレジスタの対応するビットも 1 のとき)。

例

1. 問い合わせエラー発生
2. ビット 2(QYE) が 1 にセットされる
3. 標準イベントイネーブルレジスタのビット 2 が 1 ならば、ステータスバイトのビット 5(ESB) が 1 にセットされる

また、標準イベントレジスタの内容を読むことにより、機器の内部に起こったイベントの種類を確認することができます。

標準イベントレジスタの読み出し

標準イベントレジスタの内容は、*ESR? で読み出すことができます。読み出されたあとは、レジスタはクリアされます。

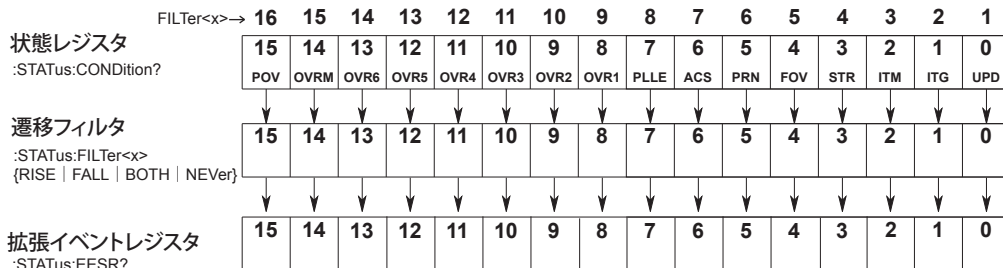
標準イベントレジスタのクリア

標準イベントレジスタがクリアされるのは、次の 3 つの場合です。

- *ESR? で標準イベントレジスタの内容が読み出されたとき
- *CLS コマンドを受信したとき
- 電源再投入時

6.4 拡張イベントレジスタ

拡張イベントレジスタは、機器の内部状態を表す状態レジスタの状態変化が、遷移フィルタでエッジ検出された結果が入ります。



状態レジスタの各ビットの意味は、次の通りです。

ビット 0	UPD(Updating)	測定データ更新中のときに「1」になります。 UPDの立ち下がり (1 → 0) が更新終了を意味します。
ビット 1	ITG(Integrate Busy)	積算中のときに「1」になります。
ビット 2	ITM (Integrate Timer Busy)	積算タイマ動作中のときに「1」になります。
ビット 3	STR (Store Busy)	ストア中のときに「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)	内蔵プリンタの動作中に 1 になります。
ビット 6	ACS(Accessing)	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 / 電流値のいずれ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 / 電流値のいずれ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 / 電流値のいずれ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 / 電流値のいずれかがオーバーレンジのときに「1」になります。
ビット 12	OVR5(Element5 Measured Data Over)	エレメント 5 の電圧値 / 電流値のいずれかがオーバーレンジのときに「1」になります。
ビット 13	OVR6(Element6 Measured Data Over)	エレメント 6 の電圧値 / 電流値のいずれ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)	モータ入力のスPEED / トルク、または外部信号入力の AUX1/AUX2 のいずれ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)	いずれかのエレメントでピークオーバを検出したときに「1」になります。

遷移フィルタのパラメータは、状態レジスタの指定されたビット (数値サフィックス 1 ~ 16) の変化を次のように抽出し、拡張イベントレジスタを書き換えます。

RISE	0 → 1 の変化で、拡張イベントレジスタの指定ビットを、1 にします。
FALL	1 → 0 の変化で、拡張イベントレジスタの指定ビットを、1 にします。
BOTH	0 → 1 または 1 → 0 の変化で、拡張イベントレジスタの指定ビットを、1 にします。
NEVer	常に 0。

6.5 出力キューとエラーキュー

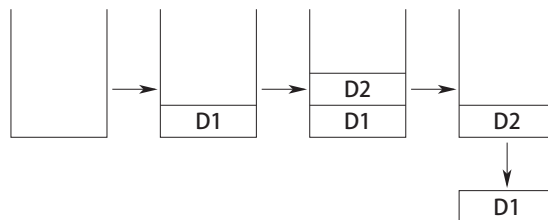
出力キュー

出力キューは、問い合わせ（クエリ）に対する応答メッセージを格納します。たとえば、取り込んだ波形データの出力を要求する `:WAVEform:SEND?` を送信すると、そのデータはそれが読み出されるまで出力キューに蓄えられます。

下図のように、データは順番に蓄えられ、古いものから読み出されます。読み出す以外にも、次のときに出力キューは空になります。

- 新しいメッセージをコントローラから受信したとき
- デッドロック状態になったとき (4-2 ページ参照)
- デバイスクリア (DCL または SDC) を受信したとき
- 電源の再投入

なお、`*CLS` コマンドでは出力キューを空にすることはできません。出力キューが空であるかどうかは、ステータスバイトのビット 4(MAV) で確認できます。



エラーキュー

エラーキューは、エラーが発生したときにその番号とメッセージを格納します。たとえば、コントローラが間違ったプログラムメッセージを送信したら、エラーが表示されたときに「113、"Undefined header"」という番号とエラーメッセージがエラーキューに格納されます。

エラーキューの内容は、`:STATus:ERRor?` クエリで読み出すことができます。エラーキューは出力キューと同様に古いものから読み出されます。

エラーキューがあふれたときは、最後のメッセージを「350、"Queue overflow"」というメッセージに置き換えます。

読み出す以外にも次のときにエラーキューは空になります。

- `*CLS` コマンドを受信したとき
- 電源の再投入

なお、エラーキューが空であるかどうかは、ステータスバイトのビット 2(EAV) で確認できます。

付録 1 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。

- メッセージは本機器本体にはシステム設定メニューの「メニュー言語 (Menu Language)」で設定した言語で表示できますが、PCなどで読み出したときは英語で表示されます。
- サービスが必要なときは、お買い求め先ま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、ユーザーズマニュアル [操作編] IM WT1801-02JA をご覧ください。

• 通信文法エラー	100～199	} 以下に記載
• 通信実行エラー	200～299	
• 機種固有(その他)	300～399	
• 通信クエリエラー	400～499	
• システムエラー(通信)	300、399	

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.	通信文法エラーコード (100 ～ 199) 以外で構文が間違っています。	4 章、 5 章
103	<DATA SEPARATOR> がありません。 Invalid separator.	データとデータは「,」(カンマ)で区切ってください。	4-1
104	<DATA> の種類が間違っています。 Data type error.	4-6 ～ 4-7 ページを参照して、正しいデータ形式で記述してください。	4-6 ～ 4-7
108	<DATA> が多すぎます。 Parameter not allowed.	データの数を確認してください。	4-6、 5 章
109	必要な <DATA> がありません。 Missing parameter.	必要なデータを記述してください。	4-6、 5 章
111	<HEADER SEPARATOR> がありません。 Header separator error.	ヘッダとデータはスペースで区切ってください。	4-1
112	<mnemonic> が長すぎます。 Program mnemonic too long.	二モニック (アルファベットと数字からなる文字列) を確認してください。	5 章
113	そのような命令がありません。 Undefined header.	ヘッダを確認してください。	5 章
114	<HEADER> の数値が間違っています。 Header suffix out of range.	ヘッダを確認してください。	5 章
120	数値の仮数部分がありません。 Numeric data error.	<NRf> 形式のときは数字が必要です。	4-6
123	指数が大きすぎます。 Exponent too large.	<NR3> 形式のときの「E」のあとの指数を小さくしてください。	4-6、 5 章
124	有効桁数が多すぎます。 Too many digits.	数字は 255 桁以内にしてください。	4-6、 5 章
128	数値データは使えません。 Numeric data not allowed.	<NRf> 形式以外のデータ形式で記述してください。	4-6、 5 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。	4-6
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。	4-6
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。	4-6
141	そのような選択肢はありません。 Invalid character data.	{...} 中にある文字列を記述してください。	4 章、 5 章
144	<CHARACTER DATA> のつづりが長すぎます。 Character data too long.	{...} の文字列のつづりを確認してください。	5 章
148	<CHARACTER DATA> は使えません。 Character data not allowed.	{...} 以外のデータ形式で記述してください。	5 章
150	<STRING DATA> の右の区切りがありません。 String data error.	<文字列> の場合は「"」または「'」で囲ってください。	4-7

付録1 エラーメッセージ

コード	メッセージ	対処方法	ページ
151	<STRING DATA> の内容が不適当です。 Invalid string data.	<文字列> が長すぎるか、使用不可能な文字があります。5 章	
158	<STRING DATA> は使えません。 String data not allowed.	<文字列> 以外のデータ形式で記述してください。	5 章
161	<BLOCK DATA> のデータ長が合っていない。 Invalid block data.	<ブロックデータ> は使用できません。	4-7、 5 章
168	<BLOCK DATA> は使えません。 Block data not allowed.	<ブロックデータ> は使用できません。	4-7、 5 章
171	<EXPRESSION> の右括弧がありません。 Missing Right	演算式は使用できません。	—
172	<EXPRESSION DATA> の中に 許されない文字があります。 Invalid expression.	演算式は使用できません。	5 章
178	<EXPRESSION DATA> は使えません。 Expression data not allowed.	演算式は使用できません。	5 章
181	プレースホルダがマクロの外にあります。 Invalid outside macro definition.	IEEE488.2 のマクロ機能には対応していません。	—

通信実行エラー (200 ～ 299)

Error in communication execution

コード	メッセージ	対処方法	ページ
221	設定内容に矛盾があります。 Setting conflict.	関連のある設定値を確認してください。	5 章
222	データの値が範囲外です。 Data out of range.	設定範囲を確認してください。	5 章
223	データのバイト長が長すぎます。 Too much data.	データのバイト長を確認してください。	5 章
224	データの値が不適当です。 Illegal parameter value.	設定範囲を確認してください。	5 章
225	オーバーフロー Overflow.	プログラムメッセージは <PMT> も含めて 1024 バイト 以下にしてください。	4-2
226	データの格納領域が足りません。 Out Of Memory.	プログラムメッセージは <PMT> も含めて 1024 バイト 以下にしてください。	4-2
241	ハードウェアが実装されていません。 Hardware missing.	オプションの有無を確認してください。	—
260	<EXPRESSION DATA> が間違っています。 Expression error.	演算式は使用できません。	—
270	マクロのネストが深すぎます。 Macro error.	IEEE488.2 のマクロ機能には対応していません。	—
272	マクロでは使用できません。 Macro execution error.	IEEE488.2 のマクロ機能には対応していません。	—
273	マクロラベルが不適当です。 Illegal macro label.	IEEE488.2 のマクロ機能には対応していません。	—
275	マクロが長すぎます。 Macro definition too long.	IEEE488.2 のマクロ機能には対応していません。	—
276	マクロが再帰呼び出しされました。 Macro recursion error.	IEEE488.2 のマクロ機能には対応していません。	—
277	マクロの二重定義はできません。 Macro redefinition not allowed.	IEEE488.2 のマクロ機能には対応していません。	—
278	そのようなマクロは定義されていません。 Macro header not found.	IEEE488.2 のマクロ機能には対応していません。	—

通信クエリエラー (400 ~ 499) Error in communication Query

コード	メッセージ	対処方法	ページ
410	応答の送信が中断されました。 Query INTERRUPTED.	送受信の順序を確認してください。	4-2
420	送信できる応答がありません。 Query UNTERMINATED.	送受信の順序を確認してください。	4-2
430	送受信がデッドロックしました。 送信を中止します。 Query DEADLOCKED.	プログラムメッセージは <PMT> も含めて 1024 バイト以下にしてください。	4-2
440	応答を要求する順番が間違っています。 Query UNTERMINATED after indefinite response.	*IDN?、*OPT? の後ろにはクエリを記述しないでください。	—

システムエラー (通信) (300、399) Error in System Operation

コード	メッセージ	対処方法	ページ
300	通信デバイスエラー。 Communication device-specific error.	サービスが必要です。	—
399	通信ドライバーエラー。 Fatal error in the communication driver.	サービスが必要です。	—

警告 (通信) (50) Warning

コード	メッセージ	対処方法	ページ
50	*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)

コード	メッセージ	対処方法	ページ
350	Queue overflow.	エラーキューを読み出してください。	6-6

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 2 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

「3.2 GP-IB インタフェースの機能と仕様」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) ユーザーがアドレス変更をしたときの動作

アドレスの変更は UTILITY キー → Remote Ctrl メニューでアドレスを設定した時点で行われます。設定したアドレスは、次に変更するまで有効です。

(4) 電源 ON 時のデバイスのセッティング。電源 ON 時に使用可能なコマンド

基本的には、以前の設定（その前に電源を OFF にしたときの設定）になります。

電源 ON 時に実行を制限されるコマンドはありません。

(5) メッセージ交換のオプション

(a) 入力バッファのサイズ

1024 バイト

(b) 複数の応答メッセージユニットを返すクエリ

5 章の各コマンドの例を参照してください。

(c) 構文解析時に応答データを作成するクエリ

すべてのクエリは、構文を解析すると応答データを作成します。

(d) 受信時に応答データを作成するクエリ

コントローラが受信する時点で応答データを作成するクエリはありません。

(e) 制限しあうパラメータを有するコマンド

5 章の各コマンドの例を参照してください。

(6) コマンドを構成する機能エレメントおよび複合ヘッダのエレメントに含まれるもの

4 章および、5 章を参照してください。

(7) ブロックデータの転送に影響するバッファのサイズ

ブロックデータの送信時には、そのサイズに合わせて出力キューを拡張します。

(8) 演算式で使えるプログラムデータのエレメントの一覧と、そのネストの制限

演算式は使えません。

(9) 各問い合わせに対する応答の構文

5 章の各コマンドの例を参照してください。

(10) 応答の文法に従わないデバイス間の通信についてサポートしていません。

(11) 応答データのブロックデータのサイズ

1 ～ 2000000

(12) サポートしている共通コマンドの一覧

「5.23 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利用でき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、マクロ定義で利用できるブロックデータの最大長、マクロ定義で再帰を使ったときの処理

マクロ機能は対応していません。

(16) *IDN? に対する返送

「5.23 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ーデータの保存エリアのサイズ

*PUD、*PUD? はサポートしていません。

(18) *RDT、*RDT? のリソース名の長さ

*RDT、*RDT? はサポートしていません。

(19) *RST、*LRN?、*RCL、*SAV による状態の変化 *RST

「5.23 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

UTILITY の Self Test メニューの MEMORY テストのすべて（内部の各メモリ）を実行します。

(21) 拡張されたリターンステータスの構造

6 章を参照してください。

(22) 各コマンドの処理がオーバーラップするか、シーケンシャルに行われるか

「4.5 コントローラとの同期」および 5 章を参照してください。

(23) 各コマンドの実行内容

5 章の各コマンドの機能とユーザーズマニュアル IM WT1801-01JA/IM WT1801-02JA を参照してください。

索引

数字

	ページ
1P2W.....	5-66
1P3W.....	5-66
3P3W.....	5-66
3P3W(3V3A).....	5-66
3P3W->3V3A 変換.....	5-70
3P4W.....	5-66
3 電圧 3 電流計法.....	5-66
4 値.....	5-32
8 値.....	5-32
16 値.....	5-32

B

	ページ
Boolean.....	4-7

C

	ページ
CAL.....	5-102
Corrected Power.....	5-74
CSV 変換.....	5-51, 5-94
CT 比.....	5-63

D

	ページ
D/A 出力項目.....	5-16
DCL(Device Clear).....	3-7
Delta->Star 変換.....	5-70

G

	ページ
GET(Group Execute Trigger).....	3-7
GP-IB インタフェース.....	3-2
GP-IB ボード.....	3-4
GTL(Go To Local).....	3-7

H

	ページ
HS Filter.....	5-50

I

	ページ
IFC(Interface Clear).....	3-7

L

	ページ
LLO(Local Lockout).....	3-7

M

	ページ
MAX HOLD.....	5-73

N

	ページ
NRF.....	4-7
NULL.....	5-62

P

	ページ
Pc.....	5-74
PLL ソース.....	5-45

R

	ページ
Register.....	4-7
REN(Remote Enable).....	3-7

S

	ページ
SDC(Selected Device Clear).....	3-7
SNTP.....	5-97
SPD(Serial Poll Disable).....	3-7
Speed.....	5-76
SPE(Serial Poll Enable).....	3-7
Star->Delta 変換.....	5-70
SyncSp.....	5-77

T

	ページ
TCP/IP.....	1-4
THD.....	5-45
TMCTL.....	ii
Torque.....	5-77

U

	ページ
USB インタフェース.....	2-2
USB キーボードの種類.....	5-99
USB ケーブル.....	2-3
USB ハブ.....	2-3

V

	ページ
VT 比.....	5-63

ア

	ページ
アドレス.....	3-2
アドレスサブルモード.....	3-2
アベレージング.....	5-70
アベレージング係数.....	5-70
アベレージングのタイプ.....	5-70

イ

	ページ
イーサネットインタフェース.....	1-2
位相差の表示形式.....	5-74
移動平均個数.....	5-70
イベント同期印刷モード.....	5-46
イベント同期ストアモード.....	5-96
イメージデータの色調.....	5-55
イメージデータの保存.....	5-55
印刷インタバル.....	5-46
印刷回数.....	5-46
印刷予約時刻.....	5-46
インタバル印刷モード.....	5-46

エ

	ページ
エラーキュー.....	6-6
エラーメッセージ.....	付-1
エレメント別積算.....	5-67

オ

	ページ
応答.....	4-5
応答メッセージ.....	4-1
オートプリント.....	5-46
オートプリント開始時に印刷.....	5-46
オートプリントの動作方式.....	5-46
オーバーラップコマンド.....	4-8
オプション.....	5-103

索引

カ ページ

回転信号の入力タイプ	5-77
回転速度	5-76
外部信号同期	5-50
外部信号入力に付加する単位	5-18
外部信号入力のスケーリング係数	5-18
外部信号入力の電圧オートレンジ	5-17
外部信号入力の電圧レンジ	5-18
外部信号入力の名前	5-18
外部信号入力のラインフィルタ	5-17
外部電流センサ換算比	5-60
外部電流センサレンジ	5-58
拡張イベントイネーブルレジスタ	5-93
拡張イベントレジスタ	5-93, 6-5
形名	5-99
紙送り	5-47
画面の輝度	5-98
画面の表示色	5-98
カラム数	5-31, 5-49

キ ページ

キーロック	5-98
機種	5-102
キャリブレーション	5-102
共通コマンドヘッダ	4-3
極性別電力量の演算方式	5-69

ク ページ

クエリ	4-1
グラディカル (目盛り)	5-35
繰り返し積算モード	5-67
グリッドの輝度	5-99
グリニッジ標準時	5-97
グループ	4-3
クレストファクタ	5-57

ケ ページ

結線方式	5-66
減衰定数	5-70

コ ページ

高速データ収集	5-49
高調波ひずみ率	5-45
高調波測定対象チャンネル	5-45
効率	5-71
効率の演算式	5-71
コメント	5-47

サ ページ

サービスリクエストイネーブルレジスタ	5-103
最小解析次数	5-45
最大解析次数	5-45
差動電圧	5-70
差動電流	5-70
サフィックス	5-99
三相 3 線式	5-66
三相 4 線式	5-66
サンプリング周波数	5-74
サンプルレート	5-101

シ ページ

時間	4-6
実時間制御印刷モード	5-46
実時間制御繰り返し積算モード	5-67

実時間制御ストアモード	5-96
実時間制御標準積算モード	5-67
ジャンプ先レンジ	5-58, 5-65
収集回数	5-49
周波数	4-6
周波数測定	5-73
周波数フィルタ	5-61
出力キュー	6-6
上位クエリ	4-4
仕様コード	5-99
小数点の種類	5-97
状態レジスタ	5-93
省略形	4-5
初期化	5-103
シリアル番号	5-99
シリアルポール	5-93
シングルショットストアモード	5-96
シングル測定	5-103

ス ページ

垂直ポジション	5-36
数値データ	5-80
数値データの出力項目	5-86
数値データのフォーマット	5-80, 5-87
数値表示 (4 値 8 値 16 値 } 表示) の表示項目	5-33
数値表示 (カスタム表示) の表示項目	5-27
数値表示の方式	5-30
数値表示 (マトリクス表示) の表示項目	5-32
数値表示 (リスト表示) の表示項目	5-30
スケーリング	5-63
スケール値表示	5-36
ステータスバイト	6-3
ステータスバイトレジスタ	5-103
ステータスレポート	5-93, 6-1
ストア	5-94
ストアインタバル	5-95
ストア開始時の数値データをストア	5-96
ストア項目	5-95
ストア項目の選択方式	5-95
ストアモード	5-96
ストア予約時刻	5-96

セ ページ

積算	5-67
積算オートキャリブレーション	5-67
積算タイマ	5-69
積算値をリセット	5-67
積算定格時間	5-16
積算同期印刷モード	5-46
積算同期ストアモード	5-96
積算モード	5-67
積算予約時刻	5-68
積算をスタート	5-68
積算をストップ	5-69
設定情報の一覧表示	5-25
セルフテスト	5-104
ゼロレベル補正	5-102
遷移フィルタ	5-93, 6-5

タ ページ

タイムアウト時間	1-2
単相 2 線式	5-66
単相 3 線式	5-66

テ	ページ
定格値設定方式	5-16
データ	4-6
データ更新レート	5-92
デッドロック	4-2
デルタ演算	5-70
デルタ演算タイプ	5-70
デルタ演算モード	5-70
電圧	4-6
電圧オートレンジ	5-64
電圧 / 電流モード	5-50
電圧レンジ	5-65
電気角	5-75
電気角のオフセット値	5-75
電気角のオフセット値の自動入力	5-75
電流	4-6
電流オートレンジ	5-57
電流積算の電流モード	5-67
電力係数	5-63
ト	ページ
同期ソース	5-64
同期測定モード	5-74
同期速度	5-77
トリガイイベント	5-47
トリガスロープ	5-37, 5-54
トリガソース	5-37, 5-54
トリガポジション	5-101
トリガモード	5-37, 5-53
トリガレベル	5-37, 5-53
トルク	5-77
トルク演算のスケーリング係数	5-79
トルク信号の入力タイプ	5-79
トレンド項目	5-34
トレンドのスケーリング	5-34
トレンドのスケーリング方式	5-34
トレンドの表示フォーマット	5-34
トレンドの横軸 (T/div)	5-35
トレンド表示のカーソル表示	5-21
ニ	ページ
入力エレメントタイプ	5-62
入力エレメントの個別設定	5-62
入力フィルタ	5-75
ハ	ページ
バーグラフ項目	5-23
バーグラフの垂直スケール形式	5-24
バーグラフのスケーリング方式	5-23
バーグラフの表示フォーマット	5-23
バーグラフ表示のカーソル表示	5-21
波形の Time/div	5-36
波形の垂直方向のズーム率	5-37
波形の表示フォーマット	5-35
波形の補間方式	5-36
波形表示データ	5-100
波形表示データのフォーマット	5-100
波形表示のカーソル表示	5-22
波形ラベル	5-36
バックライトのオートオフ	5-98
ヒ	ページ
ピークオーバー	5-63
ピークオーバー発生情報	5-50, 5-51
皮相電力の演算式	5-74
日付 / 時刻	5-97

索引	
表示分解能	5-99
表示方式	5-25
標準イベントイネーブルレジスタ	5-102
標準イベントレジスタ	5-102
標準イベントレジタ	6-4
標準積算モード	5-67

フ	ページ
ファイル保存	5-43, 5-51
ファイルリストのフィルタ	5-42
ファンクション選択肢	5-38
フォントの大きさ	5-98
プログラムメッセージ	4-1
ブロックデータ	4-7
フロントパネル	1-1, 2-1, 3-1
分割フォーマット	5-36

ヘ	ページ
ベクトルの表示フォーマット	5-35
ヘッダを付けない応答	4-5

ホ	ページ
ホールド	5-48

マ	ページ
マニュアルストアモード	5-96

メ	ページ
命令	4-3
メッセージ	4-1
メニュー言語	5-98

モ	ページ
モータ出力	5-75
モータ出力演算のスケーリング係数	5-76
モータの極数	5-76
モータ評価機能	5-75
文字データ	4-7
文字の表記法	iv
文字列データ	4-7
モデルコード	5-99

ユ	ページ
有効電圧レンジ	5-64
有効電流レンジ	5-57
ユーザー定義イベント	5-71
ユーザー定義イベントの条件式	5-71
ユーザー定義イベントの条件式タイプ	5-72
ユーザー定義イベントの名前	5-72
ユーザー定義ファンクション	5-73
ユーザー定義ファンクションの演算式	5-73
ユーザー定義ファンクションの名前	5-73

ラ	ページ
ラインフィルタ	5-50, 5-61, 5-75

リ	ページ
リアパネル	1-1, 2-1, 3-1
リモート / ローカル切り替え	1-2, 2-2, 3-3

ロ	ページ
ローカルロックアウト	5-19