



# U6200A 6 GHz ユニバーサルカウンター



# & U6220A 400 MHz 周波数カウンター



# 目次

<b>1 一般情報</b> .....	<b>5</b>
1.1 特徴概要 .....	5
1.2 保証情報 .....	6
1.3 捜査上の注意 .....	7
1.4 U6200A/220A のメンテナンス .....	8
1.5 安全性情報 .....	8
1.6 記号および用語 .....	9
1.7 点検 .....	9
1.8 オプションおよび付属品 .....	11
1.9 U6200A/220A 機能サポートリスト .....	11
<b>2 概要</b> .....	<b>13</b>
2.1 U6200A/220A のセットアップ .....	13
2.1.1 ハンドルの調節 .....	13
2.2 U6200A/220A を初めて使う .....	16
2.2.1 フロントパネル .....	16
2.2.2 リアパネル .....	18
<b>3 測定の実施</b> .....	<b>19</b>
3.1 周波数測定 .....	19
3.2 周波数レシオ測定 .....	20
3.3 時間間隔の測定 .....	21
3.4 周期測定 .....	22
3.5 立ち上がり/立ち下がり時間の測定 .....	23
3.6 パルス幅測定 .....	25
3.7 デューティサイクルの測定 .....	26
3.8 トータライズ測定 .....	27
3.9 フェーズの測定 .....	28
3.10 ピーク電圧の測定 .....	30
<b>4 フロントパネルの操作</b> .....	<b>32</b>
4.1 測定結果と状態の表示 .....	32
4.2 キーの機能 .....	33
4.2.1 数字キーパッドと矢印キー .....	33

4.2.2 測定制御キー (Run および Stop/Single) .....	34
4.3 ゲート及び外部アームの操作 .....	34
4.4 倍率およびオフセットの設定 .....	42
4.5 リミット機能 .....	44
4.6 統計演算 .....	46
4.7 チャンネル入力条件の設定 .....	47
4.8 トリガー設定 .....	48
4.9 保存と再呼び出し機能 .....	51
4.10 ユーティリティ操作 .....	53
4.11 表示のヒント .....	57
4.11.1 リモートコマンドエラーの表示 .....	57
4.11.3 電源投入時の初期設定リセット .....	58
<b>5 リモートインターフェイス操作 .....</b>	<b>60</b>
5.1 USB コネクタ経由での PASS/FAIL (合否) 出力 .....	60
5.2 UP I/O のセットアップ .....	61
5.3 リモートインターフェイスコマンド .....	64
<b>6 手動キャリブレーション手順 .....</b>	<b>75</b>
6.1 CH1 のオフセットキャリブレーションを行う方法 .....	75
6.2 CH2 のオフセットキャリブレーション方法 .....	75
6.3 CH1 の ATTx10 オフセットキャリブレーション方法 .....	76
6.4 CH2 の ATTx10 オフセットキャリブレーション方法 .....	76
6.5 CH1 のゲインキャリブレーション方法 .....	77
6.6 CH2 のゲインキャリブレーション方法 .....	77
6.7 CH1 の ATTx10 ゲインキャリブレーション方法 .....	77
6.8 CH2 の ATTx10 ゲインキャリブレーション方法 .....	78
6.9 タイムドメイン・キャリブレーションの方法 (外部アームを含む) .....	78
6.10 時間間隔キャリブレーションの方法 .....	79
6.11 時間間隔キャリブレーションの方法 (外部アームを含む) .....	80
6.12 標準タイムベースキャリブレーションの方法 .....	81
<b>7 エラーメッセージ .....</b>	<b>82</b>
7.1 コマンドエラー .....	82
7.2 実行エラー .....	84
7.3 デバイス固有エラー .....	86
7.4 クエリエラー .....	86
7.5 カウンター固有エラー .....	86

付録 .....	88
A. 仕様リスト .....	88
A.1 RMS 分解能 .....	92
B. プリセット値とセーブ/リコール .....	93
C. 一般仕様 .....	94
D. リモートインターフェイス・リファレンス .....	95
D.1 SCPI 言語への導入 .....	95
D.2 コマンドレファレンス .....	99
D.3 IEEE-488 共通コマンド .....	149
D.4 IEEE-488 コンプライアンス 情報 .....	155
E. アプリケーション・プログラムについて .....	161

# 1 一般情報

---

このセクションは PICOTEST U6200A ユニバーサル カウンターおよび U6220A 周波数カウンター (省略表記 U6200A/220A)にかかわる一般情報を記載しています。情報の内容は次の通りです:

- ・特長概要
- ・保証情報
- ・操作上の注意 U6200A/220A のメンテナンス
- ・安全性情報
- ・記号および用語
- ・点検
- ・オプションおよび付属品

保証,サービスまたはテクニカルサポート情報については次の電話番号から東洋計測器までご連絡ください。

電話番号: 03-3255-8026

ウェブサイト <https://keisokuki-land.co.jp>

---

## 1.1 特長概要

U6200A/220A:

- U6200A チャンネル 1 & 2; DC 1mHz~400 MHz
- U6200A チャンネル 3; 375MHz~6GHz
- 12 桁分解能、ゲート時間 1s
- 40 ps タイムドメイン分解能
- インターフェース; USB、イーサネット(10M/100M)、GPIB (オプション)
- 測定結果のマルチパラメータ表示
- 使いやすいショートカットキー
- 無償のPC アプリケーション

- ※ 注: U6220A はチャンネル 2&3 (緑色)は提供していません。
  - ※ 注: U6200A/220A の全仕様は付録 A に記載されています。
  - ※ 注: Picotest はフリーウェア U6200A/220A PT-LINK (Ver 1.0)を提供しています。  
このソフトウェアの GUI (グラフィカル・ユーザー・インターフェイス)ツールで測定結果を取得し、マイクロソフト Excel または Word 用データに変換し分析や編集可能です。
- 

## 1.2 保証情報

本機器を指定されている方法で使用しないと、機器の保護機能が損傷を受けることがあります。保証の詳細に関しては以下を参照して下さい。

1. **Warranty:** PICOTEST CORP. guarantees that this product meets its published specifications at the time of shipment from the factory. Under proper installation it should work as expected.

2. **Warranty Period:** This equipment is warranted against defects in material and workmanship for a period of one year from date of shipment. During the warranty period, PICOTEST is responsible for necessary repairs as long as the product can be proved to be defective.

For warranty service or repair, this product must be returned to a service facility designated by PICOTEST. Please contact your local service representative.

3. **Excluded Items:** This warranty does not include consumptive parts such as fuses, buttons and relays. Neither does this warranty cover defects caused by improper installation, improper or insufficient maintenance, unauthorized modification, improper operation, ignorance of environmental specifications or improper software or interfacing.

4. Remarks:

- No other warranty is expressed or implied, except for the above mentioned.
- The remedies provided herein are the buyer's sole and exclusive remedies.

PICOTEST shall not be liable for any direct, indirect, special, incidental or consequential damages.

## 保証の制限

保証の制限に関しては、下記の英文を参照して下さい。

1. Our warranties do not cover any damage resulting from unauthorized modification or misuse.
  2. Unless mentioned elsewhere in this document, our warranty does not apply to fuses, probes, and problems arising from normal wear or user's failure to follow instructions.
  3. Our warranties do not apply on any direct, incidental, special, or consequential damages.
  4. The above warranties are exclusive and no other warranty is expressed or implied. Picotest disclaims any implied warranties of MERCHANTABILITY, SATISFACTORY QUALITY, and FITNESS for any particular reasons.
- 

### 1.3 操作上の注意

- この機器を操作する前に本マニュアルをよく読んでください。
- このマニュアルは参考用としてお使いください。より具体的な支援については、担当の代理店にお問い合わせください。
- このマニュアルの内容は予告無しに修正されることがあります。
- 機器が損傷を受けるおそれがありますので、資格のない人に分解させないでください。
- 機器は当社の工場から出荷される前に厳格な品質検査を受けていますが、損傷の恐れのある危険な状況では使用しないでください。
- 本機器は予期されない人による使用を防止するため、安全な場所に保管してください。
- 背面保護導体端子は有効なアースに接続してください。そうしないと感電することがあります。
- 製品にかかわる特許および関連文書は PICOTEST CORP の財産です。いかなる複製も非合法です。

---

## 1.4 U6200A/220A のメンテナンス

U6200A ユニバーサル・カウンターは耐久性があり耐候性に優れていますが、大きな衝撃や圧力を掛けないようご注意ください。

- U6200A/220A は、水や湿気の多い環境から離してください。
- 校正を毎年行うことをお勧めします。詳しくは担当サービス代理店にお問い合わせください。
- 不正表示や、異常ビープ音が鳴った場合、直ちに機器の使用を停止してください。
- 爆発性ガスや可燃性蒸気の近くで使用しないでください。
- U6200A ユニバーサル・カウンターの表面は、乾燥した清潔な布で拭いてください。

---

## 1.5 安全性情報



注意！ 機器を使用する前に次の安全性情報をよく読んでください。

感電や負傷を防止するため、以下のガイドラインをよく読み従ってください:

- このマニュアルに記載のガイドラインに従い、ケースが損傷している場合は、使用しないでください。機器筐体およびターミナルを確認し、全てが適正な状況で使ってください。
- 過度の電圧を印加しないでください。電圧は定格範囲のみとしてください。
  - ・ 30V RMS、42 V ピークまたは 60V DC を超える電圧を測定する際はよく注意してください。これらの電圧は感電の危険があります。
- 主電源モジュールには定格 3.15A/250V のヒューズが搭載されています。  
ヒューズ(BUSSMANN F3.15A250V)
- 機器 が損傷している場合、操作しようとししないでください。機器の電源を抜き、担当サービス代理店にご相談ください。

---

## 1.6 記号および用語



この記号は機器の損傷または人員の負傷さえ引き起こすような危険を意味しています。



この記号は高電圧が存在する可能性を意味しています。いかなる動作でも行う前に特別の注意をしてください。



この記号はフレームまたはシャーシターミナルが有効な接地に接続される必要があることを意味しています。



この記号は「保護用導体端子」を意味しています。



UL規格<sup>®</sup>。



この記号は接地(アース)端子を意味しています。



この記号は本製品が安全、健康、環境および消費者保護にかかわる基本要件または適用される欧州法または指令に準拠していることを意味しています。

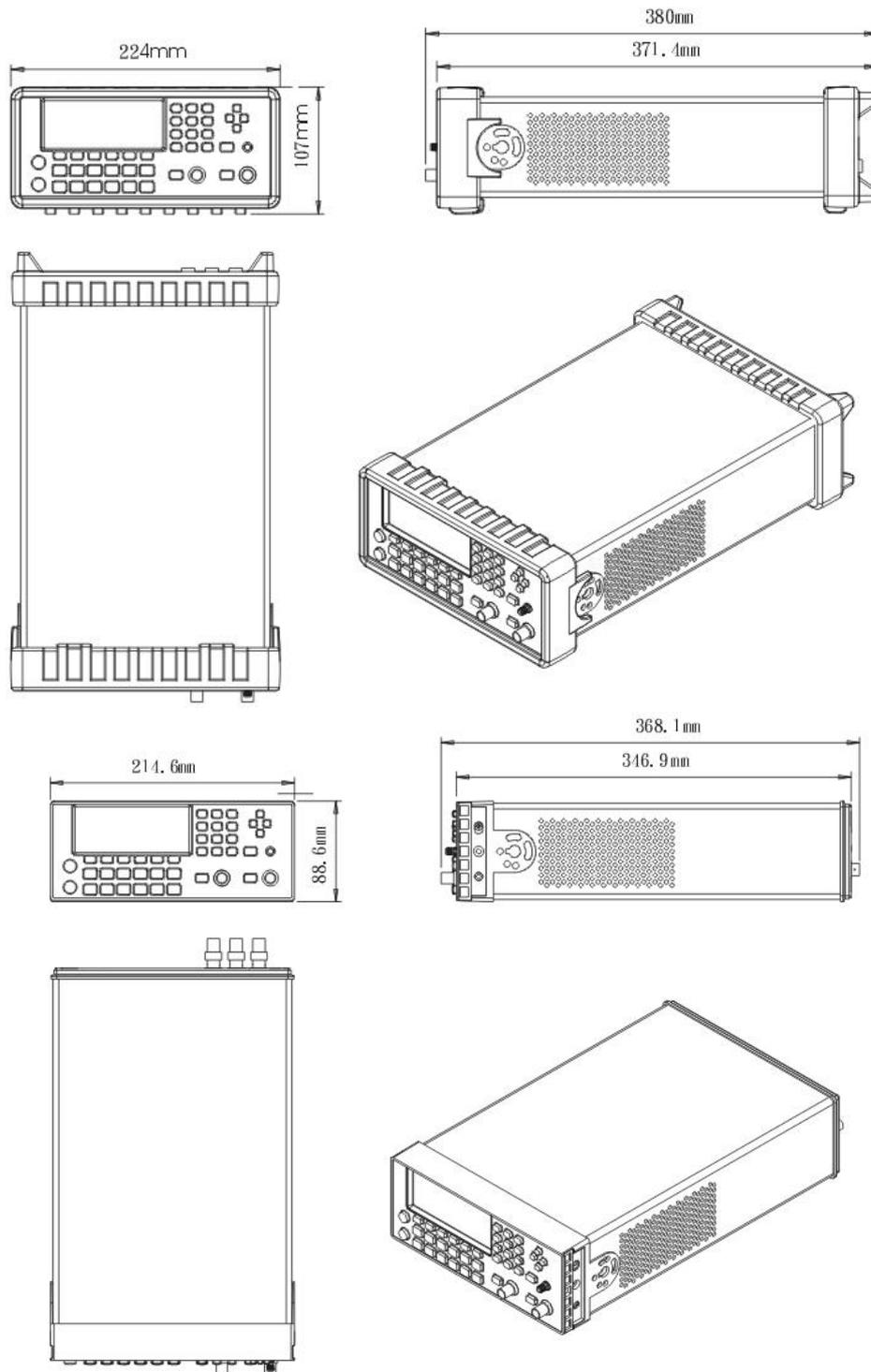
---

## 1.7 点検

製品パッケージには次のアイテムが同梱されています：

- U6200A/220A 1台 [224 (W) x 107 (H) x 380 (D) mm、フロントおよびリアバンパー付き、約 3416g] / [214.6 (W) x 88.6(H) x 346.9 (D) mm、フロントおよびリアバンパー無し、約 3130g]
- 電源コード 1本。

- USBケーブル 1本。
- CD 1枚(英文取扱説明書およびソフトウェアアプリケーション)。
- オプション付属品 (ご注文の場合)
- GPIBインターフェイスカード (オプション)。



---

## 1.8 オプションおよび付属品

U6200Aと使用するために次のオプションおよび付属品がPicotestから入手可能です。  
表 1-1 を参照ください。

表 1-1 付属品

部品名称	部品番号
リアパネル入力モジュール(CH1/CH2)	U6200-opt04
リアパネル入力モジュール (CH1/CH2/CH3)	U6200-opt05
GPIBカード	M3500-opt04

※ 注: 黒字の付属品はU6220A用です。

---

## 1.9 U6200A/220A 機能サポートリスト

機能	チャンネル	U6200A	U6220A
周波数	1	V	V
周波数	2	V	
周波数	3	V	
周期	1	V	V
周期	2	V	
周期	3	V	
レシオ	1,2	V	
レシオ	2,1	V	
レシオ	1,3	V	
レシオ	3,1	V	
立ち上がり時間		V	
立ち下がり時間		V	
POS 幅		V	

NEG 幅		V	
時間間隔		V	
フェーズ		V	
デューティ		V	
トータライズ		V	V
VOLT_PTP	1	V	V
VOLT_PTP	2	V	
VOLT_MAX	1	V	V
VOLT_MAX	2	V	
VOLT_MIN	1	V	V
VOLT_MIN	2	V	
VOLT_CENT	1	V	V
VOLT_CENT	2	V	

※ 注: U6220A で未サポートの機能、指示、コマンドは以下では緑で表記されています。

# 2 概要

---

この章は U6200A/220A の基本的機能の概要を記載し、U6200A/220A の基本事項を紹介しています。この章を読めば基本的機能がよく把握できます。

---

## 2.1 U6200A/220A のセットアップ

このセクションはU6200A/220Aの使用法を説明しています。始めに、全部品が揃っているか確認して下さい。当社の全製品は出荷前に検査を受けていますが、損傷部品や部品の欠品があったり、機器の使用にかかわる疑問がある場合は、直ちに担当サービス代理店に連絡してください。損傷した機器は操作しないでください。

---

### 2.1.1 ハンドルの調節

目的に合わせて搬送ハンドルを調節することができます。次の図はその方法を示しています。

#### I. ユニバーサル・カウンターからハンドルを取り外す

【ステップ 1】(ハンドルを回し上げる)

ハンドル両側を若干外側に引き、**図 2-1** に示されるようにゆっくりと回して立てます。

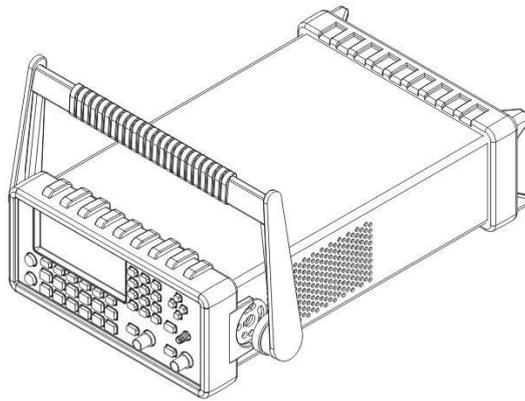


図 2-1

【ステップ 2】(ハンドルを引き出す)

ハンドルが筐体に対して 90° まで回ったら、図 2-2. に示うようにハンドルを筐体から引き抜いてください。

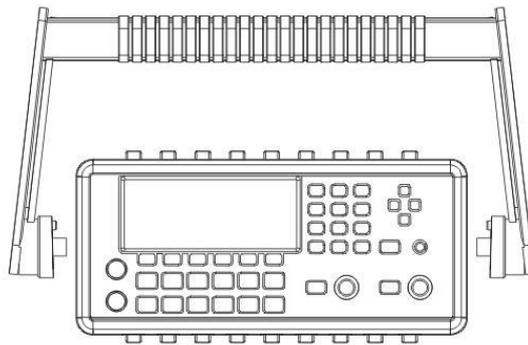


図 2-2

II. 便利な位置に調整

個別ニーズに合わせた位置の数例です：

【位置 1】

初期の位置は、パッキング用に、図 2-3 に示すような位置になっています。

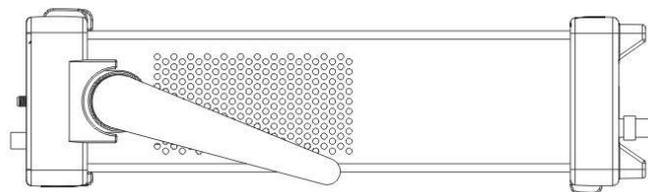
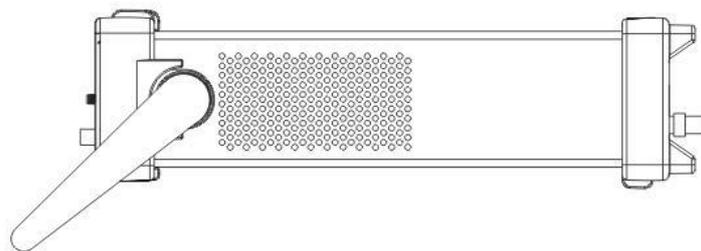


図 2-3

**【位置 2】**

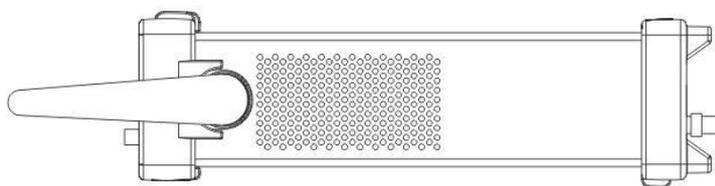
操作の位置は図 2-4 のようになります。



**図 2-4**

**【位置 3】**

持ち運ぶ際は図 2-5 のようになります。



**図 2-5**

## 2.2 U6200A/220A 使用開始

この章ではU6200A/220Aのフロントとリアパネルにある コネクタとボタンについて説明します。

### 2.2.1 フロントパネル

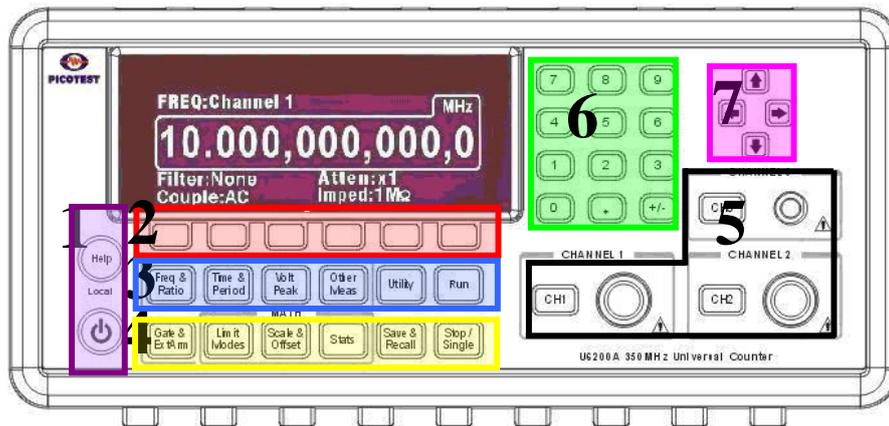


図 2-6

#### 1. 電源&ディスプレイ(紫部分):

- 電源: U6200A/220A の電源を ON/OFF します。
- ヘルプ: 指定の機能について説明します。

#### 2. 1列目ソフトキー (赤い部分):

U6200A/220A のディスプレイ表示メニューにアクセスできます。

#### 3. 2列目 (青い部分):

- Freq & Ratio: 周波数&周波数レシオ測定機能メニューキー
- Time & Period: 時間間隔 と周期測定機能メニューキー
- Volt Peak: ピーク電圧測定機能メニューキー
- Other Meas: フェーズ、デューティーサイクルとトータライズ測定メニューキー
- Utility: ユーティリティ設定メニューキー
- Run: 連続測定の開始キー

※ 注: U6220Aは「タイムドメイン」機能をサポートしていません。

#### 4. 第3列(黄色部分):

- Gate & Ext アーム: ゲート&外部アームメニューキー

- Limit Modes: リミット設定およびテストメニューキー
- SCALE & OFFSET: 倍率およびオフセット入力キー
- Stats: 統計機能メニューキー
- SAVE & RECALL: 測定設定の保存／読出しメニューキー
- Stop/Single: 単一測定を行うか現在の測定終了後に連続測定を停止するためのキー

5. **チャンネル状態ボタンとコネクタ (黒い部分):**

- CH1:チャンネル 1 状態制御ボタンおよびコネクタ
- CH2:チャンネル 2 状態制御ボタンおよびコネクタ
- CH3:チャンネル 3 状態制御ボタンおよびコネクタ

※ 注: U6220Aはチャンネル 2&3 はありません。

6. **12 個の数字キー (緑の部分):**

数値入力キーです。

7. **4 つの矢印キー(ピンクの部分):** 数値を増減するためカーソルを制御するキーで、英文字入力もサポートします。

注: フロントパネルにある全てのボタンとキーのなかでも、3と4の項目のボタンは機能が有効の場合点灯します。

## 2.2.2 リアパネル

U6200A/220Aのリアパネルは図 2-7 に示されています。

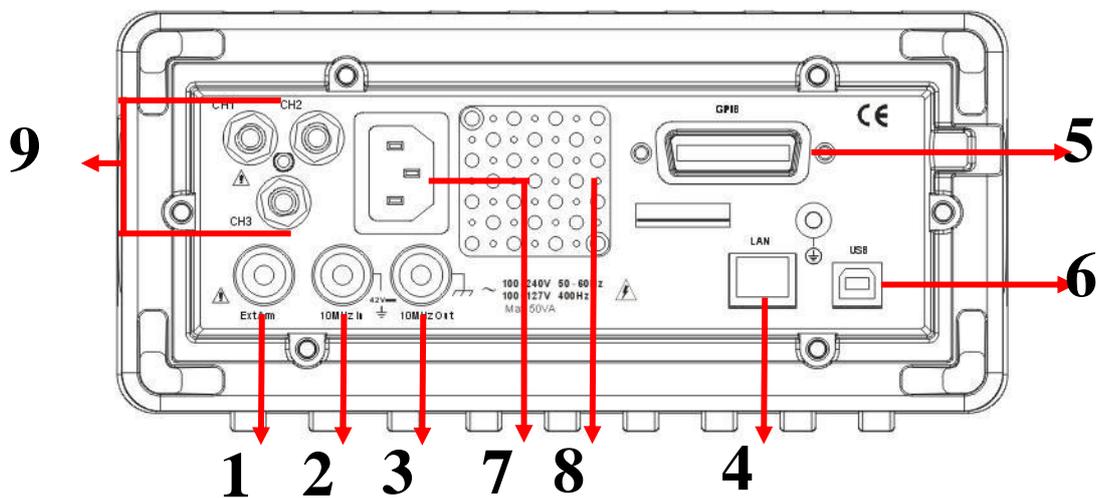


図 2-7

1. 外部アーム入力端子
2. 外部 10 MHz 入力端子
3. 外部 10 MHz 出力端子
4. LAN インターフェイスコネクタ
5. GPIB インターフェイスコネクタ
6. USB インターフェイスコネクタ
7. 電源コード コネクタ
8. ファン
9. CH1/2/3 リア入力

※ 注: U6220Aはリア入力 CH1/CH2/CH3 は提供していません。

# 3 測定の実施

---

この章はU6200A/220Aで基本的測定を行う方法と関連の初期設定パラメタについて説明します。

---

## 3.1 周波数測定

オートトリガーは周波数測定用に初期設定で有効化されており、トリガーレベルは信号の 50%ポイントに設定されています。手動でオートトリガー機能を無効化でき、トリガーレベルおよびスロープ設定を変更できます(4.8 章を参照)。

### 周波数の測り方

1. 信号源を使用したい入力チャンネルに接続します。
2. FREQ & RATIO ボタンを押します。
3. ソフトキーFREQ を押し、周波数測定機能を選択します。再度このソフトキーを押すと目的のチャンネルを選択できます。
4. オプション設定:
  - a. GATE & ExtArm ボタンを押してゲート時間、分解能または、必要ならアーミングを設定します(4.3 章を参照)。
  - b. SCALE & OFFSET ボタンを押して倍率と必要ならオフセット・パラメタを設定します(4.4 章を参照)。
  - c. Limit Modes ボタンを押して必要なら上限値と下限値を設定します(4.5 章を参照)。

※ 注: U6220A は“リミット判定”機能をサポートしていません。

- d. Stats ボタンを押して統計機能と必要ならリミット・フィルタリングを設定します(4.6 章を参照)。
- e. CH1 の選択肢から入力条件およびトリガー感度を設定します。
- f. Show Status ボタンを押して Input、Volt、Stats、およびリミット値の条件を確認します。
- g. SAVE & RECALL ボタンを押して設定を保存します(上記の a-d で行った変更)(4.9 章を参照)。

5. RUN ボタンを押して連続測定を開始します。
6. STOP/SINGL ボタンを押し、単一測定を行うか連続測定を停止します。

### リモートインターフェイスコマンド

次のコマンドで周波数測定を行うように U6200A に命令します:

```
[[:SENSe]:FUNction[:ON] "[[:XNOnE:]:FREQuency [1|2|3]"
```

または

```
:CONFigure[:SCALar][:VOLTage]:FREQuency [(@1)|(@2)|(@3)]
```

```
:READ[[:SCALar][:VOLTage]:FREQuency]?
```

または

```
:MEASure[:SCALar][:VOLTage]:FREQuency? [(@1)|(@2)|(@3)]
```

※ 注: U6220A はチャンネル 2&3 はサポートしていません。

---

## 3.2 周波数レシオ測定 (U6220Aにはこの機能はありません。)

周波数レシオ測定に対して、オートトリガーが初期設定で有効化されており、トリガーレベルは各信号の 50%ポイントに設定されています。手動で、オートトリガーを無効化しトリガーレベルとスロープ設定を変更することができます(4.8 章を参照)。

### 周波数レシオの測り方

1. 信号源を使用したい入力チャンネルに接続します。
  2. FREQ & RATIO ボタンを押します。
  3. RATIO の下のソフトキーを押して周波数レシオ測定機能を選びます。再度このソフトキーを押すと目的のチャンネルを選ぶことができます。
  4. オプション設定:
    - a. GATE & ExtArm ボタンを押してゲート時間、分解能または、必要ならアーミングを設定します(4.3 章を参照)。
    - b. SCALE & OFFSET ボタンを押して倍率と必要ならオフセット・パラメタを設定します(4.4 章を参照)。
    - c. Limit Modes ボタンを押して必要なら上限値と下限値を設定します(4.5 章を参照)
- ※ 注: U6220A は「リミット判定」機能をサポートしていません。
- d. Stats ボタンを押して統計機能および必要ならリミット・フィルタリングを設定します(4.6 章を参照)。

- e. CH1 の選択肢から入力条件およびトリガー感度を設定します。
  - f. Show Status ボタンを押して Input、Volt、Stats、およびリミット値の条件を確認します。
  - g. SAVE & RECALL ボタンを押して設定を保存します上記の a-d で行った変更) (4.9 章を参照)。
5. RUN ボタンを押して連続測定を開始します。
  6. STOP/SINGL ボタンを押し、単一測定を行うか連続測定を停止します。

### リモートインターフェイスコマンド

次のコマンドで周波数レシオを測定します:

```
[[:SENSe]:FUNction[:ON] "[[:XNOnE:]:FREQuency:RATio [1,2|2,1|3,1|1,3]"
```

または

```
:CONFigure[:SCALar][:VOLTage]:FREQuency:RATio
```

```
[(@1),(@2)](@2),(@1)](@1),(@3)](@3),(@1)]
```

```
:READ[:SCALar][:VOLTage]:FREQuency:RATio)?
```

または

```
:MEASure[:SCALar][:VOLTage]:FREQuency:RATio?
```

```
[(@1),(@2)](@2),(@1)](@1),(@3)](@3),(@1)]
```

※ 注: U6220A は「Frequency Ratio」コマンドをサポートしていません。

### 3.3 時間間隔の測定 (U6220Aにはこの機能はありません。)

Interval 1 to 2 はチャンネル 1 でのスタート・イベントとチャンネル 2 でのストップ・イベントの間の時間を測ります。オートトリガーは初期設定で有効化されており、トリガーレベルは信号の 50%ポイントに設定されています。手動でオートトリガー機能を無効化でき、トリガーレベルおよびスロープ設定を変更できます(4.8 章を参照)。

#### 時間間隔の測り方

1. 信号源を使用したい入力チャンネルに接続します。
2. TIME & PERIOD ボタンを押します。
3. Intval 1~2 の下のソフトキーを押して間隔測定機能を選びます。
4. オプション設定:
  - a. GATE & ExtArm ボタンを押してゲート時間、分解能または、必要ならアーミングを設定します(4.3 章を参照)。
  - b. SCALE & OFFSET ボタンを押して倍率と必要ならオフセット・パラメタを設定します(4.4 章を参照)。
  - c. Limit Modes ボタンを押して必要なら上限値と下限値を設定します(4.5 章を参照)。

参照)

- d. Stats ボタンを押して統計機能および必要ならリミット・フィルタリングを設定します(4.6 章を参照)。
  - e. CH1 の選択肢から入力条件およびトリガー感度を設定します。
  - f. Show Status ボタンを押し Input、Volt、Stats、およびリミット値の条件を確認します。
  - g. SAVE & RECALL ボタンを押して設定を保存します(上記の a-d で行った変更) (4.9 章を参照)。
5. RUN ボタンを押して連続測定を開始します。
  6. STOP/SINGL ボタンを押し、単一測定を行うか連続測定を停止します。
- スタート・イベントとストップ・イベントは 2 チャンネルのどちらかに接続された同じ信号からトリガーできます。詳しい操作については 4.8 章をご参照ください。

### リモートインターフェイス・コマンド

時間間隔測定を行うには次のコマンドを使用します:

```
[[:SENSE]:FUNCTION[:ON] "[:][XNONE:]TINterval [1,2]"
```

または

```
:CONFigure[:SCALar][:VOLTage]:TINterval [(@1),(@2)]
```

```
:READ[:SCALar][:VOLTage]:TINterval?
```

または

```
:MEASure[:SCALar][:VOLTage]:TINterval? [(@1),(@2)]
```

※ 注: U6220A は「時間間隔」測定をサポートしていません。

---

## 3.4 周期測定

周期測定用では、オート・トリガーが初期設定になっており、トリガー・レベルは各信号ごとに 50%ポイントに設定されています。尚、手動で、オート・トリガーを無効にし、トリガー・レベルとスロープ設定を変更することができます(4.8 章を参照)。

### 周期の測り方

1. 信号源をチャンネル 1 に接続します。
2. TIME & PERIOD ボタンを押します。
3. Period の下のソフトキーを押して周期測定機能を選びます。

4. オプション設定:
  - a. GATE & ExtArm ボタンを押してゲート時間、分解能または、必要ならアーミングを設定します(4.3 章を参照)。
  - b. Scale & Offset ボタンを押して倍率と必要ならオフセット・パラメタを設定します(4.4 章を参照)。
  - c. Limit Modes ボタンを押して必要なら上限値と下限値を設定します(4.5 章を参照)
  - d. Stats ボタンを押して統計機能および必要ならリミット・フィルタリングを設定します(4.6 章を参照)。
  - e. CH1 の選択肢から入力条件およびトリガー・センサを設定します。
  - f. Show Status ボタンを押して Input、Volt、Stats、および限界値の条件を確認します。
  - g. Save & Recall ボタンを押して設定を保存します(上の a-d で行った変更です。)(4.9 章を参照)。
5. RUN ボタンを押して連続測定を開始します。
6. STOP/SINGL ボタンを押し、単一測定を行うか連続測定を停止します。

#### リモートインターフェイス・コマンド

周期測定を行うには次のコマンドを使用します:

```
[[:SENSe]:Function[:ON] "[[:XNOnE:]:PERiod [1|2|3]"
```

または

```
:CONFigure[:SCALar][:VOLTage]:PERiod [(@1)|(@2)|(@3)]
```

```
:READ[[:SCALar][:VOLTage]:PERiod]?
```

または

```
:MEASure[:SCALar][:VOLTage]:PERiod? [(@1)|(@2)|(@3)]
```

※ 注: U6220A はチャンネル 2&3 はありません。

---

## 3.5 立ち上がり立ち下がり時間の測定

立ち上がり立ち下がり時間測定では、オート・トリガが初期設定になっており、信号の 10%と 90%ポイントを自動識別し、これに基づいてトリガ・レベルを設定します。手動で、オート・トリガを無効化し、設定を変更することができます (4.8 章を参照)。

### 立ち上がりまたは立ち下がり時間の測り方

1. 信号源をチャンネル 1 に接続します。
2. TIME & PERIOD ボタンを押します。
3. EDGE 下のソフトキーを押してエッジ測定機能を選びます。立ち上がりか立ち下がり時間かは再度押せばトグル式に変わります。
4. オプション設定：
  - a. GATE & ExtArm ボタンを押してゲート時間、分解能または、必要ならアーミングを設定します(4.3 章を参照)。
  - b. Scale & Offset ボタンを押して倍率と必要ならオフセット・パラメタを設定します(4.4 章を参照)。
  - c. Limit Modes ボタンを押して必要なら上限値と下限値を設定します(4.5 章を参照)
  - d. Stats ボタンを押して統計機能および必要ならリミット・フィルタリングを設定します(4.6 章を参照)。
  - e. CH1 の選択肢から入力条件およびトリガ感度を設定します。
  - f. Show Status ボタンを押して Input、Volt、Stats、およびリミット値の状態を確認します。
  - g. Save & Recall ボタンを押して必要なら設定を保存します(上の a-d で行った変更です。) (4.9 章を参照)。
5. RUN ボタンを押して連続測定を開始します。
6. STOP/SINGL ボタンを押し、単一測定を行うか連続測定を停止します。

### リモートインターフェイス・コマンド

立ち上がりか立ち下がり時間測定を行うには次のコマンドを使用します:

```
[:SENSe]:FUNction[:ON] “[:][XNONe:]RISE:TIME [1]”
[:SENSe]:FUNction[:ON] “[:][XNONe:]RTIME [1]”
[:SENSe]:FUNction[:ON] “[:][XNONe:]FALL:TIME [1]”
[:SENSe]:FUNction[:ON] “[:][XNONe:]FTIME [1]”
```

または

```
:CONFigure[:SCALar][:VOLTage]:RISE:TIME [(@1)]
:CONFigure[:SCALar][:VOLTage]:RTIME [(@1)]
:CONFigure[:SCALar][:VOLTage]:FALL:TIME [(@1)]
:CONFigure[:SCALar][:VOLTage]:FTIME [(@1)]
:READ[:SCALar][:VOLTage]:RISE:TIME?
:READ[:SCALar][:VOLTage]:RTIME?
:READ[:SCALar][:VOLTage]:FALL:TIME?
:READ[:SCALar][:VOLTage]:FTIME?
```

または

:MEASure[:SCALar][:VOLTage]:RISE:TIME? [(@1)]  
:MEASure[:SCALar][:VOLTage]:RTIME? [(@1)]  
:MEASure[:SCALar][:VOLTage]:FALL:TIME? [(@1)]  
:MEASure[:SCALar][:VOLTage]:FTIME? [(@1)]

※ 注: U6220A は「立ち上がり/立ち下がり時間測定」をサポートしていません。

---

## 3.6 パルス幅測定

パルス幅測定用に、オート・トリガが有効になっており、トリガ・レベルは信号ごとに50%ポイントに設定されています。手動で、オート・トリガを無効化し、設定を変更することができます (4.8 章を参照)。

### パルス幅の測り方

1. 信号源をチャンネル 1 に接続します。
2. TIME & PERIOD ボタンを押します。
3. WIDTH の下のソフトキーを押してパルス幅測定機能を選びます。再度押すとプラスかマイナスどちらかの幅を選ぶことができます。
4. オプション設定:
  - a. ソフトキーの PERIOD CH1/CH2 および GATE&ExtArm ボタンを押してゲート時間、分解能または、必要ならアーミングを設定します(4.3 章を参照)。
  - b. Scale & Offset ボタンを押して倍率と必要ならオフセット・パラメタを設定します(4.4 章を参照)。
  - c. Limit Modes ボタンを押して必要なら上限値と下限値を設定します(4.5 章を参照)
  - d. Stats ボタンを押して統計機能および必要ならリミット・フィルタリングを設定します(4.6 章を参照)。
  - e. CH1 の選択肢から入力条件およびトリガ感度を設定します。
  - f. Show Status ボタンを押して Input、Volt、Stats、およびリミット値の条件を確認します。
  - g. Save & Recall ボタンを押して必要なら行った設定を保存します(上の a-d で行った変更です。) (4.9 章を参照)。
5. RUN ボタンを押して連続測定を開始します。
6. STOP/SINGLE ボタンを押し、単一測定を行うか連続測定を停止します。

## リモートインターフェイス・コマンド

パルス幅を測定するには次のコマンドを使用します:

```
[[:SENSe]:FUNcTION[:ON] "[[:XNOnE:]:PWIDth [1]"
```

```
[[:SENSe]:FUNcTION[:ON] "[[:XNOnE:]:NWIDth [1]"
```

または

```
:CONFigure[:SCALar][:VOLTage]:PWIDth [(@1)]
```

```
:CONFigure[:SCALar][:VOLTage]:NWIDth [(@1)]
```

```
:READ[:SCALar][:VOLTage]:PWIDth?
```

```
:READ[:SCALar][:VOLTage]:NWIDth?
```

または

```
:MEASure[:SCALar][:VOLTage]:PWIDth? [(@1)]
```

```
:MEASure[:SCALar][:VOLTage]:NWIDth? [(@1)]
```

※ 注: U6220A は「パルス幅測定」をサポートしていません。

---

## 3.7 デューティーサイクルの測定

デューティーサイクルは次の公式で定義され、結果は 0 と 1 の間になります:

$$\text{デューティーサイクル} = (+\text{のパルス幅}) / \text{周期}$$

デューティーサイクルモードはチャンネル 1 でのみ適用可能です。さらに、オート・トリガが初期設定で有効化されており、トリガ・レベルは信号ごとに 50% に設定されています。手動で、オート・トリガを無効化し、設定を変更することができます (4.8 章を参照)。

### デューティーサイクルの測り方

1. 信号源をチャンネル 1 に接続します。
2. OTHER MEAS ボタンを押します。
3. DUTY CYC1 の下 1 列目のソフトキーを押し、デューティーサイクル測定機能を選びます。
4. オプション設定:
  - a. GATE & ExtArm ボタンを押して必要なら自動または外部ゲート時間を設定します(4.3 章を参照)。
  - b. Scale & Offset ボタンを押して倍率と必要ならオフセット・パラメタを設定します(4.4 章を参照)。
  - c. Limit Modes ボタンを押して必要なら OnFail、OUT\*、Upper と Lower リミットを設定します(4.5 章を参照)。

- d. Stats ボタンを押して統計機能および必要ならリミット・フィルタリングを設定します (4.6 章を参照)。
  - e. CH1 の選択肢から入力条件およびトリガ感度を設定します。
  - f. Show Status ボタンを押して Input、Volt、Stats、およびリミット値の条件を確認します。
  - g. Save 0& Recall ボタンを押して必要なら設定を保存します(上の a-d で行った変更です。) (4.9 章を参照)。
5. RUN ボタンを押して連続測定を開始します。
  6. STOP/SINGLE ボタンを押し、単一測定を行うか連続測定を停止します。

### リモートインターフェイス・コマンド

デューティーサイクルを測定するには次のコマンドを使用します:

```
[[:SENSe]:Function[:ON] “[[:XNOnE:]:DCYClE [1]”
```

```
[[:SENSe]:Function[:ON] “[[:XNOnE:]:PDUTyCyclE [1]”
```

または

```
:CONFigure[:SCALar][:VOLTagE]:DCYClE [[:@1]]
```

```
:CONFigure[:SCALar][:VOLTagE]:PDUTyCyclE [[:@1]]
```

```
:READ[:SCALar][:VOLTagE]:DCYClE?
```

```
:READ[:SCALar][:VOLTagE]:PDUTyCyclE?
```

または

```
:MEASure[:SCALar][:VOLTagE]:DCYClE? [[:@1]]
```

```
:MEASure[:SCALar][:VOLTagE]:PDUTyCyclE? [[:@1]]
```

※ 注: U6220A は「デューティーサイクル測定」をサポートしていません。

---

## 3.8 トータライズ測定

トータライズ測定は入力信号のトリガされたイベントの数を数えて表示します。チャンネル 1 のみ対応しています。入力された信号が測定に適合していないかもしれない為、オート・トリガが自動的にオフになります。GATE:AUTO が選択されている場合(4.3 章を参照)、カウンターはイベントカウントを連続表示します。それ以外の場合、トータライズ数はゲーティング&アーミング 設定により制約されます。測定値はトータライズ機能が最初に呼び出される時、RUN ボタンが押されたとき、または、STOP/SINGLE ボタンが押されたときゼロにリセットされます。

※ 注: トータライズ測定機能は Scale & Offset、Limit Modes、Stats (統計測定) では使用できません。

#### トータライズ測定を行う方法

1. 信号源をチャンネル 1 に接続します。
2. OTHER MEAS ボタンを押します。
3. TOTAL CH 1 の下のソフトキーを押し、トータライズ測定機能を選びます。
4. オプション設定:
  - a. GATE & ExtArm ボタンを使用して必要ならゲート時間または外部アーミングを設定します(4.3 章を参照)。
  - b. CH1 の選択肢から入力条件およびトリガ感度を設定します。
  - c. Show Status ボタンを押して Input、Volt、Stats、およびリミット値の条件を確認します。
  - d. Save & Recall ボタンを押して必要ならステップ a の変更を保存します(4.9 章を参照)。
5. RUN または STOP/SINGLE ボタンを押し、測定を開始します。

#### リモートインターフェイスコマンド

トータライズ測定を行うには次のコマンドを使用します:

```
[[:SENSe]:FUNction[:ON] "[:][XNOnE:]TOTalIze [1]"
```

または

```
:CONFigure[:SCALar][:VOLTage]:TOTalIze:CONTInuous [(@1)]
```

```
:CONFigure[:SCALar][:VOLTage]:TOTalIze:TIMed [(@1)]
```

```
:READ[[:SCALar][:VOLTage]:TOTalIze:TIMed]?
```

または

```
:MEASure[:SCALar][:VOLTage]:TOTalIze:CONTInuous? [(@1)]
```

```
:MEASure[:SCALar][:VOLTage]:TOTalIze:TIMed? [(@1)]
```

---

## 3.9 フェーズの測定

Phase 1 to 2 はチャンネル 1 とチャンネル 2 での信号の間の位相差を測定し、度数で示されます。さらに初期設定でオート・トリガが有効化されており、トリガ・レベルは信号ごとに 50%ポイントに設定されています。手動でオート・トリガ機能を無効化でき、トリガ・レベルおよびスロープ設定を変更できます(4.8 章を参照)。

この際チャンネル 2 が基準となります。チャンネル 2 の 1 周期を 360°とします。チャンネル 1 はチャンネル 2 より先行するので、結果は正になります。

### フェーズの測り方

1. 信号源をそれぞれチャンネル 1 とチャンネル 2 に接続します。
2. OTHER MEAS ボタンを押します。
3. (PHAS 1 to 2)の下のソフトキーを押し、フェーズの測定機能を選択します。
4. オプション設定：
  - a. ソフトキー「Phase 1 to 2」と GATE & ExtArm ボタンを押して必要なら自動または外部アームを設定します(4.3 章を参照)。
  - b. Scale&Offset ボタンを押して倍率と必要ならオフセット・パラメタを設定します(4.4 章を参照)。
  - c. Limit Modes ボタン を押して必要なら OnFail、OUT\*、Upper と Lower リミット値を設定します(4.5 章を参照)。
  - d. Stats ボタンを押して統計機能および必要ならリミット・フィルタリングを設定します(4.6 章を参照)。
  - e. CH1 & CH2 の選択肢から入力条件およびトリガー感度を設定します。
  - f. Show Status ボタンを押して Input、Volt、Stats、およびリミット値の条件を確認します。
  - g. Save & Recall ボタンを押して必要なら行った設定を保存します(上の a-d で行った変更です。) (4.9 章を参照)。
5. RUN ボタンを押して連続測定を開始します
6. STOP/SINGLE ボタンを押し、単一測定を行うか連続測定を停止します。

### リモートインターフェイスコマンド

フェーズを測定するには次のコマンドを使用します:

```
[[:SENSe]:FUNction[:ON] "[[:XNOnE:]:PHASe [1,2]"
```

または

```
:CONFigure[:SCALar][:VOLTage]:PHASe [(@1),(@2)]
```

```
:READ[[:SCALar][:VOLTage]:PHASe]?
```

または

```
:MEASure[:SCALar][:VOLTage]:PHASe? [(@1),(@2)]
```

※ 注: U6220A は「フェーズの測定」をサポートしていません。

---

## 3.10 ピーク電圧の測定

U6200A は入力信号のプラスまたはマイナスのピーク電圧も測ることができます。尚、入力信号は、入力設定（インピーダンス、カップリング、アッテネーションおよびフィルタリング、4.7 章を参照）により条件付けられた後で、測定される点に注意してください。測定結果は X10 減衰に対して補正されます。また、ゲーティング&アーミング、リミット及び演算機能は、ピーク電圧測定では利用できません。

### プラスまたはマイナスのピーク電圧の測り方

1. 信号源をチャンネル 1 または 2 に接続します。
2. VOLT PEAK ボタンを押してピーク電圧測定機能を選択します。
3. VOLT の下のソフトキーを押し、入力チャンネルを選択します。
4. オプション設定：
  - a. MODE の下のソフトキーを押し、操作モード Fast または Slow を選択します。
  - b. SHOW の下のソフトキーを押し、結果を Vmax (プラス・ピーク)、Vmin (マイナス・ピーク)、Vc (プラスとマイナスピークから求められる平均値)または Vpp (ピーク・ツー・ピーク)で表示するかを選びます。
  - c. CH1 又は CH2 の機能を使って入力条件とトリガ感度を設定します。
  - d. Show Status ボタンを押して Input、Volt、Stats、およびリミット値の条件を確認します。
  - e. SAVE&RECALL ボタンを押して必要なら設定を保存します(上の a-d で行った変更です。) (4.9 章を参照)。
5. RUN ボタンを押して連続測定を開始します
6. STOP/SINGLE ボタンを押し、単一測定を行うか連続測定を停止します。

※ 注: U6220A はチャンネル 2 は提供していません。

### リモートインターフェイスコマンド

次のコマンドを使用してピーク電圧測定を行います:

```
[[:SENSe]:]FUNction[:ON] “[:][XNOnE:]VOLTage:MAXimum [1|2]”
```

```
[[:SENSe]:]FUNction[:ON] “[:][XNOnE:]VOLTage:MINimum [1|2]”
```

```
[[:SENSe]:]FUNction[:ON] “[:][XNOnE:]VOLTage: PTPeak [1|2]”
```

```
[[:SENSe]:]FUNction[:ON] “[:][XNOnE:]VOLTage: CENTer [1|2]”
```

または

```
:CONFigure[:SCALar][:VOLTage]: MAXimum [(@1)|(@2)]
```

```
:CONFigure[:SCALar][:VOLTage]: MINimum [(@1)|(@2)]
```

```
:CONFigure[:SCALar][:VOLTage]:PTPeak [(@1)|(@2)]
```

:CONFigure[:SCALar][:VOLTage]: CENTer [(@1)|(@2)]

:READ[:SCALar][:VOLTage]:PTPeak?

または

:MEASure[:SCALar][:VOLTage]:MAXimum? [(@1)|(@2)]

:MEASure[:SCALar][:VOLTage]:MINimum? [(@1)|(@2)]

:MEASure[:SCALar][:VOLTage]:PTPeak? [(@1)|(@2)]

:MEASure[:SCALar][:VOLTage]:CENTer? [(@1)|(@2)]

※ 注: U6220A はチャンネル 2 は提供していません。

# 4 フロントパネルの操作

この章は測定機能を構成するいろいろな方法を説明します。この章には、測定用パラメータと設定の変更方法やシステム設定用のその他全てのフロントパネル操作についての情報が記載されています。リモートインターフェイスの設定は第 5 章に記載されています。

## 4.1 測定結果と状態の表示

U6200A/220A は、測定チャンネルの設定情報と共に測定結果を表示します。以下の説明はピーク電圧を除く全ての測定機能について有効です:

例えば、U6200A の TIME&PERIOD 測定機能の表示では、Show Status ソフトキーを押して図 4-1 に示されるようにディスプレイの右下コーナーのメッセージを確認します。次に、その項目の下の同じソフトキーを押すと次の情報に切り替わります。例えば、図 4-2 に示されるように、入力条件が現れます。



図 4-1



図 4-2

- Input: 図 4-3 に示されるように 入力チャンネルのパラメータを表示します。
- Stats: 図 4-4 に示されるように統計 (平均、標準偏差、 $\Delta P-P$  および N) と統計値を得る際の測定数を表示します。
- Limit: 図 4-6 に示されるように測定結果をグラフモードで表示します。
- Volt: 図 4-5 に示されるように 入力信号の最大値、最小値、Vc と Vpp 電圧を表示します。



図 4-3



図 4-4



図 4-5

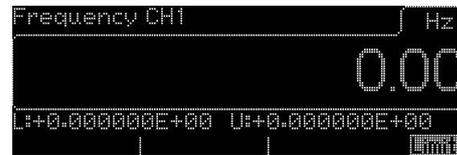


図 4-6

**Volt Peak 機能に関しては、状態は Input と Volt 情報を表示します。**

フロントパネル操作:

1. 測定キーのうち 1 つを押して測定機能を選択します。
2. STATUS の下のソフトキーを押します。
3. ソフトキーを押し続けオプション設定を順に切り替えます。
4. 選択した測定機能に関しては、3 章で説明されている手順を続けます。

---

## 4.2 キーの機能

U6200A/220A の各キーとボタンは用途に合わせて使用します。フロントパネルに 2 種類のキーを搭載しています。一つは、数字キーと矢印キーで、もう一つは、測定制御キーです。

---

### 4.2.1 数字キーパッドと矢印キー

数字キーパッド用の 12 個のボタンと 4 個の矢印キーがあり、数字入力に使用します。数値を入力するには、数字キーパッドで直接必要な数値を入力するか、左右矢印キーを使い目的の数字へカーソルを移動し、上下矢印キーで値の増減を行います。入力を完了するには、Done の下または目的の単位/指数の下のソフトキーを押して値を設定します。変更を取り消すには、CANCEL の下のソフトキーを押します。

---

## 4.2.2 測定制御キー (RunおよびStop/Single)

U6200A/220A には、RunとSingleの2つの測定モードがあります。Runキーを押すとカウンターはRunモードに、Stop/Singleキーはシングルモードになります。

### Runモードの操作

- U6200A/220A が、いずれかの測定モードであればRunキーを押してRunモードにし、連続測定を開始します。
- U6200A/220A が、いずれかの測定モードであれば再度Runキーを押して現在の測定を中断させ、再度押すと再びRunモードに入ります。この場合取得した統計データはすべて消去されます。
- U6200A/220A が、いずれかの測定モードであればStop/Singleキーを押すと、測定を停止します。
- U6200A/220A がRunモードであれば、他の測定機能を選択すると、実行中測定を中断します。

### Singleモードの操作

- U6200A/220A が、いずれかの測定モードで、測定実行中の場合、Stop/Singleキーを押すと、シングルモードに入り、STATS:SINGLE:Nが選択されているか、STATS:N CNTがNに設定されていれば、単一測定かN測定を行います。(4.6章を参照)。測定の完了後、カウンターは自動的にシングルモードを終了します。
  - U6200A/220A がRunモードであれば、Stop/Singleキーを押して、シングルモードにします。
  - U6200A/220A がシングルモードにあればStop/Singleキーを押してカウンターに現在の測定を中止させます。
- 

## 4.3 ゲート及び外部アームの操作

U6200A/220A は、自動、時間、桁、外部の4種類のアーミングモードがあります。その内容は次表に示されています。周波数/周期/レシオ、フェーズ(U6220Aには無い)、トータライズ、立ち上がり/立ち下がり時間(U6220Aには無い)、時間間隔(U6220Aには無い)などの各機能に対して、選択できる項目を記載しています:

周波数/周期/ レシオ	フェーズ	トータライズ	立上り時間,立下り時間 +/- 幅 デューティサイクル	時間間隔
自動アーミング:  ゲート: 自動	自動アーミング:  アーム: 自動	自動アーミング:  ゲート: 自動	自動アーミング:  ゲート: 自動	自動アーミング: *アーム: 自動 遅延*: 無し 時間 イベント 時間*: <時間> イベント*: <イベント>
桁数 アーミング: ゲート: 桁数 桁数: <桁数>				
時間 アーミング: ゲート: 時間 時間: <時間>		時間 アーミング: ゲート: 時間 時間: <時間>		
外部アーミング: ゲート: Ext 始動: POS NEG 自動 停止: NEG POS 時間 時間: <時間>	外部アーミング: アーム: Ext スロープ: POS NEG	外部アーミング: ゲート: Ext 始動: POS NEG 停止: NEG POS 時間 時間: <時間>	外部アーミング: ゲート: Ext 停止: POS NEG	外部アーミング: *アーム: Ext *停止: POS NEG *遅延: 無し 時間 イベント *時間: <時間> *イベント: <イベント> Stop*: 自動 Ext Slop*: POS NEG 遅延* 無し 時間 イベント 時間*: <時間> イベント*: <イベント>

### ● 自動アーミング

図 4-7 に示されるように、自動アーミングモードで利用可能な測定は周波数、周期、周波数レシオ、立上り時間、立下り時間、パルス幅、デューティサイクル、および、フェーズがあり、トータライズ測定以外は出来る限り早く測定を行います。トータライズ測定については、Run ボタンか Stop/Single ボタンが押されたときに測定を開始します。



図 4-7

- **時間アーミング**

図 4-8 に示されるように、時間アーミングモードは、周波数、周期、レシオ、トータライズ測定に利用でき、測定する時間（ゲート時間）の長さを設定することができます。



図 4-8

※ 注 1: 周波数と周期のゲート時間範囲は 1E-3～1000 秒。

※ 注 2: レシオのゲート時間範囲は 1E-3 ～200 秒。

- **外部アーミング**

図 4-9 に示されるように外部アーミングモードは、周期、周波数、レシオ、トータライズの測定に利用可能で、信号源をリアパネルにある Ext Arm コネクタに接続する必要があります。測定の開始と終了は外部信号で制御されます。

外部アーミングがトグル式の場合、さらに 2 つの項目を設定することができます。

一つは Start (Neg/Pos)、もう一つは Stop (Neg/Pos/時間/自動)です。



図 4-9

- **桁数アーミング**

図 4-10 に示されるように周波数、周期およびレシオの測定において、3～15 桁の範囲で桁数アーミングモードを利用でき、分解能を設定することができます。

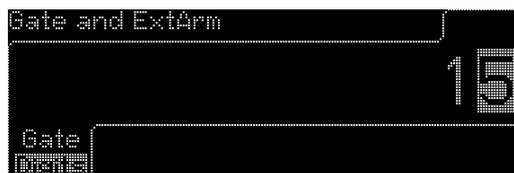


図 4-10

## ゲート&外部アーム操作

1. Stop/Single ボタンを押して、連続測定を停止します。
2. 測定機能キーと対応するソフトキーを押し、目的の測定機能を選びます(3 章を参照)。
3. 外部アーミングを適用したい場合は、リアパネル上の Ext Arm コネクタに外部信号源を接続します。
4. Gate & ExtArm ボタンを押してメニューからの選択を続けます。

## 周波数、期間、レシオ測定:

GATE の下のソフトキーを押し、自動、時間、桁、外部をトグルします。

- 自動アーミングが選択されている場合、カウンターは自動的に連続測定を開始します。
- 桁アーミングが選択されている場合、数字キーパッドか矢印キーで必要な桁数か分解能を入力します(4.2.1 章を参照)。
- 時間アーミングが選択されている場合、数字キーパッドか矢印キーで必要なゲート時間を選択します(4.2.1 章を参照)。単位は右上隅に表示されます。
- 外部アーミングが選択されている場合、START の下のソフトキーを押し、測定をプラスまたはマイナスエッジのどちらで開始するかを指定します。プラスとマイナスの間をトグルするために同じキーを押します。STOP の下のソフトキーを押し、プラス、マイナス、自動、時間の間をトグルします。これで測定の停止方法を指定します。プラスとマイナスは停止アーム信号のスロープを指定します。時間は測定継続時間を設定します。時間が選択されていれば、数字キーパッドか矢印キーで必要な継続時間の値を入力します(4.2.1 章を参照)。時間単位は入力完了後に右上隅に表示されます。

## リモートインターフェイスコマンド

### 自動アーミング:

```
[:SENSe]:FREQuency:ARM[:STARt]:SOURce IMMEDIATE
```

```
[:SENSe]:FREQuency:ARM:STOP:SOURce IMMEDIATE
```

### 桁アーミング:

```
[:SENSe]:FREQuency:ARM[:STARt]:SOURce IMMEDIATE
```

```
[:SENSe]:FREQuency:ARM:STOP:SOURce DIGits
```

```
[:SENSe]:FREQuency:ARM:STOP:DIGits <numeric value>
```

### 時間アーミング:

```
[:SENSe]:FREQuency:ARM[:STARt]:SOURce IMMEDIATE
```

```
[:SENSe]:FREQuency:ARM:STOP:SOURce TIMer
```

[:SENSe]:FREQuency:ARM:STOP:TIMer <numeric value>

**外部アーミング:**

[:SENSe]:FREQuency:ARM[:STARt]:SOURce EXTernal

[:SENSe]:FREQuency:ARM[:STARt]:SLOPe POSitive|NEGative

[:SENSe]:FREQuency:ARM:STOP:SOURce IMMEDIATE

[:SENSe]:FREQuency:ARM:STOP:SOURce EXTernal

[:SENSe]:FREQuency:ARM:STOP:SLOPe POSitive|NEGative

[:SENSe]:FREQuency:ARM:STOP:SOURce TIMer

[:SENSe]:FREQuency:ARM:STOP: TIMer <numeric value> [s]

※ 注: U6220A は緑色の「時間間隔、立上り／立り時間、デューティーサイクル、幅測定」をサポートしていません。

**立上り／立り時間、デューティーサイクル、幅測定:**

TIME & PERIOD と Gate & ExtArm ボタンを押し、Gate ソフトキー下の自動と外部の間をトグルします。

- 自動アーミングが選択されている場合、カウンターは自動的に連続測定を開始します。
- 外部アーミングが選択されている場合、SLOPE の下のソフトキーを押し、プラスとマイナスの間をトグルします。これで外部起動アーム信号の勾配を指定します。

**リモートインターフェイスコマンド**

**自動アーミング:**

[:SENSe]:TINTerval:ARM:ESTART:LAYer2:SOURce IMMEDIATE

**外部アーミング:**

[:SENSe]:TINTerval:ARM:ESTART:LAYer2:SOURce EXTernal

[:SENSe]:TINTerval:ARM:ESTART:LAYer2:SLOPe POSitive|NEGative

ESTART:LAY2:SOURce	フロントパネルアーミングモード
IMMEDIATE	自動
EXTernal	外部

**フェーズの測定:**

Other Meas と Gate & ExtArm ボタンを押し、ARM ソフトキー下にある自動と外部の間をトグルします。

- 自動アーミングが選択されている場合、カウンターは自動的に連続測定を開始しま

す。

- 外部アーミングが選択されている場合、SLOPE の下のソフトキーを押し、プラスとマイナスの間をトグルします。これで外部起動アーム信号のスロープを指定します。

#### リモートインターフェイスコマンド

##### 自動アーミング:

[[:SENSe]:PHASe:ARM[:STARt]:SOURce IMMEDIATE

##### 外部アーミング、U6200A 用のみ:

[[:SENSe]:PHASe:ARM[:STARt]:SOURce EXTERNAL

[[:SENSe]:PHASe:ARM[:STARt]:SLOPe POSitive|NEGative

START:SOURce	フロントパネルアーミングモード
IMMEDIATE	自動
EXTERNAL	外部

#### トータライズ測定:

GATE の下のソフトキーを押し、自動、時間、外部の間をトグルします。

- 自動アーミングが選択されている場合、カウンターは自動的に連続測定を開始します。
- 時間アーミングが選択されている場合、数字キーパッドか矢印キーで目的のゲート時間を選択します(4.2.1 章を参照)。時間単位は右上隅に表示されます。
- 外部アーミングが選択されている場合、START の下のソフトキーを押し、プラスとマイナスの間をトグルします。これで起動アーム信号のスロープを指定します。STOP の下のソフトキーを押し、プラス、マイナス、および、時間の間をトグルします。これで測定停止方法を指定します。プラスとマイナスは停止アーム信号のスロープを指定します。時間は測定継続時間に設定されています。時間が選択されている場合、数字キーパッドか矢印キーで必要な継続時間の値を入力します(4.2.1 章を参照)。時間単位は右上隅に表示されます。

#### リモートインターフェイスコマンド

##### 自動アーミング:

[[:SENSe]:TOTalize:ARM[:STARt]:SOURce IMMEDIATE

[[:SENSe]:TOTalize:ARM:STOP:SOURce IMMEDIATE

##### 時間アーミング:

[[:SENSe]:TOTalize:ARM[:STARt]:SOURce IMMEDIATE

[[:SENSe]:TOTalize:ARM:STOP:SOURce TIMER

[:SENSe]:TOTalize:ARM:STOP:TIMer <numeric value> [S]

**外部アーミング:**

[:SENSe]:TOTalize:ARM[:START]:SOURce EXTernal

[:SENSe]:TOTalize:ARM[:START]:SLOPe POSitive|NEGative

[:SENSe]:TOTalize:ARM:STOP:SOURce EXTernal

[:SENSe]:TOTalize:ARM:STOP:SLOPe POSitive|NEGative

[:SENSe]:TOTalize:ARM:STOP:SOURce TIMer

[:SENSe]:TOTalize:ARM:STOP:TIMer <numeric value> [S]

START:SOURce	STOP:SOURce	Front-Panel GATE	Settings STOP
IMMediate	IMMediate	Auto	----
IMMediate	TIMer	Time	----
EXTernal	EXTernal	EXTERNAL	POS, NEG
EXTernal	TIMer	EXTERNAL	Time

**時間間隔測定:**

GATE の下のソフトキーを押し、自動と外部の間をトグルします。DONE の下のソフトキーを押し、選択を完了します。

- 自動アーミングが選択されている場合、カウンターは自動的に連続測定を開始します。さらに Delay\* の下のソフトキーを押し、無し、時間、イベントの間をトグルします。時間とイベントの設定では目的に従いパラメタを入力することができます。
- 外部アーミングが選択されている場合、遅延時間その他の設定を\*Dely の下のソフトキーを押し、無し、時間、イベントの間をトグルし行うことができます。\*Slop の下のソフトキーを押し、Pos (プラス)と Neg (マイナス)の間をトグルし、Dely\* の下のソフトキーを押し、無し、時間、イベントの間をトグルし、Stop\* の下のソフトキーを押し、無し、自動と外部の間をトグルします。ソフトキーが Ext にトグルされるとはじめて、Pos(プラス)と Neg(マイナス)の間をトグルできる Slop\*項目が表示されることに注意して下さい。設定情報は次の通りです:

スタート遅延時間範囲: 100 ns ~999.9999 ms  
 スタート遅延時間分解能: 100 ns  
 スタート遅延イベント範囲: 1 ~ 99,999,999 チャンネル 2 イベント  
 スタート遅延イベント速度: 10 MHz 最大  
 スタート遅延イベントセットアップ時間: 30 ns 最小

ストップ遅延時間範囲: 100 ns ~10 s

ストップ遅延時間分解能: 100 ns : 100 ns ~ 1 s

1 ms : 1 s ~ 10 s

ストップ遅延イベント範囲: 1 ~ 99,999,999 チャンネル 2 イベント

ストップ遅延イベント速度: 10 MHz最大

ストップ遅延イベントセットアップ時間: 30 ns最小

※ 注: \*Slop のプラスとマイナスにより、起動アーム信号の勾配を指定します。Stop\* のプラスとマイナスにより、測定停止方法を指定します。また、時間に関して これは測定継続時間を設定します。時間が選択されている場合、数字キーパッドか矢印キーで目的の継続時間の値を入力します(4.2.1 章を参照)。時間単位は右上隅に表示されません。

## リモートインターフェイスコマンド

### 自動アーミング:

[ :SENSe]:TINTerval:ARM:ESTART:LAYer2:SOURce IMMEDIATE

[ :SENSe]:TINTerval:ARM:ESTOP:LAYer2:SOURce IMMEDIATE

[ :SENSe]:TINTerval:ARM:ESTOP[:LAYer[1]]:SOURce IMMEDIATE

[ :SENSe]:TINTerval:ARM:ESTOP[:LAYer[1]]:SOURce TIMer

[ :SENSe]:TINTerval:ARM:ESTOP[:LAYer[1]]:SOURce INTernal2

[ :SENSe]:TINTerval:ARM:ESTOP[:LAYer[1]]:TIMer <numeric\_value> [S]

[ :SENSe]:TINTerval:ARM:ESTOP[:LAYer[1]]:ECOunt <numeric\_value>

### U6200A の外部アーミング:

[ :SENSe]:TINTerval:ARM:ESTART:LAYer2:SOURce EXTERNAL

[ :SENSe]:TINTerval:ARM:ESTART:LAYer2:SLOPe POSitive | NEGative

[ :SENSe]:TINTerval:ARM:ESTART[:LAYer[1]]:SOURce IMMEDIATE

[ :SENSe]:TINTerval:ARM:ESTART[:LAYer[1]]:SOURce TIMer

[ :SENSe]:TINTerval:ARM:ESTART[:LAYer[1]]:SOURce INTernal1

[ :SENSe]:TINTerval:ARM:ESTART[:LAYer[1]]:TIMer <numeric\_value> [S]

[ :SENSe]:TINTerval:ARM:ESTART[:LAYer[1]]:ECOunt <numeric\_value>

[ :SENSe]:TINTerval:ARM:ESTOP:LAYer2:SOURce IMMEDIATE

[ :SENSe]:TINTerval:ARM:ESTOP:LAYer2:SOURce EXTERNAL

[ :SENSe]:TINTerval:ARM:ESTOP:LAYer2:SLOPe POSitive | NEGative

[ :SENSe]:TINTerval:ARM:ESTOP[:LAYer[1]]:SOURce IMMEDIATE

[ :SENSe]:TINTerval:ARM:ESTOP[:LAYer[1]]:SOURce TIMer

[ :SENSe]:TINTerval:ARM:ESTOP[:LAYer[1]]:SOURce INTernal2

[ :SENSe]:TINTerval:ARM:ESTOP[:LAYer[1]]:TIMer <numeric\_value> [S]

[ :SENSe]:TINTerval:ARM:ESTOP[:LAYer[1]]:ECOunt <numeric\_value>

ESTART:LAY2 :SOURCE	ESTART:LAY1 :SOURCE	ESTOP:LAY2 :SOURCE	ESTOP:LAY1 :SOURCE	Front-Panel Arming Setting			
				*START	*DELAY	STOP*	DELAY*
IMMediate	IMMediate	IMMediate	IMMediate	自動	---	自動	無し
IMMediate	IMMediate	IMMediate	TIMer	自動	---	自動	時間
IMMediate	IMMediate	IMMediate	INTernal2	自動	---	自動	イベント
EXTernal	IMMediate	IMMediate	IMMediate	Ext		自動	無し
EXTernal	IMMediate	IMMediate	TIMer	Ext	無し	自動	時間
EXTernal	IMMediate	IMMediate	INTernal2	Ext	無し	自動	イベント
EXTernal	IMMediate	EXTernal	IMMediate	Ext	無し	Ext	無し
EXTernal	IMMediate	EXTernal	TIMer	Ext	無し	Ext	時間
EXTernal	IMMediate	EXTernal	INTernal2	Ext	無し	Ext	イベント
EXTernal	TIMer	IMMediate	IMMediate	Ext	無し	自動	無し
EXTernal	TIMer	IMMediate	TIMer	Ext	時間	自動	時間
EXTernal	TIMer	IMMediate	INTernal2	Ext	時間	自動	イベント
EXTernal	TIMer	EXTernal	IMMediate	Ext	時間	Ext	無し
EXTernal	TIMer	EXTernal	TIMer	Ext	時間	Ext	時間
EXTernal	TIMer	EXTernal	INTernal2	Ext	時間	Ext	イベント
EXTernal	INTernal1	IMMediate	IMMediate	Ext	時間	自動	無し
EXTernal	INTernal1	IMMediate	TIMer	Ext	イベント	自動	時間
EXTernal	INTernal1	IMMediate	INTernal2	Ext	イベント	自動	イベント
EXTernal	INTernal1	EXTernal	IMMediate	Ext	イベント	Ext	無し
EXTernal	INTernal1	EXTernal	TIMer	Ext	イベント	Ext	時間
EXTernal	INTernal1	EXTernal	INTernal2	Ext	イベント	Ext	イベント

## 4.4 倍率(Scale)およびオフセットの設定

U6200A/220A は測定結果が表示される前に倍率とオフセットの計算を実行するオプションを提供しており、数式は次のようになります：

$$(\text{表示された結果}) = (\text{倍率} \times \text{測定結果}) + \text{オフセット}$$

図 4-10 に示すように、Scale と Offset はユーザー指定のパラメーターであり、体系的な誤差をオフセットしたり、信号間のパーセンテージの差を表示したりするための乗算および加算関数を提供します。このオプションは初期設定では無効になっています。ScaleまたはOffsetのいずれかを指定すると自動的にオンになります。



図 4-10

### 倍率およびオフセットでの計算を有効／無効にする

1. SCALE & OFFSET ボタンを押します。
2. 倍率かOffsの下のソフトキーを押し、個別設定を行います。

### リモートインターフェイスコマンド

このコマンドで数学的演算を有効化または無効化します:

:CALCulate:MATH:STATe OFF | ON

### 倍率の値指定

1. SCALE & OFFSETボタンを押します。
2. SCALEの下のソフトキーを押すと現在の倍率値が表示されます。
3. 数字キーパッドか矢印キーで目的の値を入力します (4.2.1 章を参照)。  
倍率 範囲:    -9.999,999E+12 ~ -1.000,000E-13  
                  0.000,000  
                  +1.000,000E-13 ~ +9.999,999E+12  
値の分解能:   7 桁
4. Doneの下のソフトキーを押し、入力を終えるか、Cancel下のソフトキーを押して変更を撤回します。

### リモートインターフェイスコマンド

次のコマンドで倍率の値を指定します:

:CALCulate:MATH:STATe OFF | ON

### オフセット値を指定する

1. SCALE&OFFSETボタンを押します。
2. Offsの下のソフトキーを押すと、オフセットの現在値が表示されます。
3. 数字キーパッドか矢印キーで目的の値を入力します(4.2.1 章を参照)。  
オフセット範囲:   -9.999,999,000,0E+12 ~ -1.000,000,000,0E-13  
                          0.000,000,000,0  
                          +1.000,000,000,0E-13 ~ +9.999,999,000,0E+12  
分解能値:   11 桁
4. Doneの下のソフトキーを押し、入力を終えるかCancel下のソフトキーを押して変更を撤回します。

### リモートインターフェイスコマンド

オフセット値を指定するには次のコマンドを使用します:

: TRACe[:DATA] OFFSET,<numeric value> [HZ|S|DEG]

### Set Offs?を指定する

Set Offsを有効化した場合、値は自動的にマイナスに設定されます。

1. SCALE & OFFSETボタンを押します。
2. Set Offs? の下のソフトキーを押します。

---

## 4.5 リミット機能

U6200A/220A では最大・最小測定リミット値を設定でき、測定値がリミット値を超えたときに測定を停止するか継続するかを指示することができます。

リミット・モードはトータライズ(Other Meas.にある)と Volt Peak 測定には利用できないことにご注意ください。

図 4-11 に示されるように、リミット・モードにある機能により、(OnFail: Stop)を設定すれば、ユーザー定義のリミット値を超える測定値を取得すると、測定を停止することができます。反対に、設定が(OnFail: Go on)であれば測定値がユーザー定義の値を超えても測定は継続可能です。また、(OUT\* USB)が有効なので、超過した測定データはUSB を介して別のユニットへ転送されそこで診断されます。

※ 注: U6220A は(OUT\* USB) 機能をサポートしていません。



図 4-11

また、単一測定モードでは(OnFail: Stop)は無効です。

### 上限値または下限値を設定する

1. Limit Modes ボタンを押します。
2. Upper の下のソフトキーを押し、上限値を、また、Lower の下のソフトキーを押し、下限値を設定します。数字キーパッドか矢印キーで必要な限界値を入力します(4.2.1 章を参照)。

上限範囲: -9.999,999,000,0E+12~-1.000,000,000,0E-13  
0.000,000,000,0  
+1.000,000,000,0E-13~+9.999,999,000,0E+12  
分解能: 11 桁

下限範囲: -9.999,999,000,0E+12~-1.000,000,000,0E-13  
0.000,000,000,0  
+1.000,000,000,0E-13~+9.999,999,000,0E+12  
分解能: 11 桁

3. DONE の下のソフトキーを押し、リミット値の設定を終了します。

#### リモートインターフェイスコマンド

次のコマンドを使用して上限値または下限値を設定します:

:CALCulate2:LIMit:UPPer[:DATA] <numeric value> [HZ|S|DEG]

:CALCulate2:LIMit:LOWer[:DATA] <numeric value> [HZ|S|DEG]

#### リミットテストを行う

1. Limit Modes ボタンを押します。
2. ONFAIL の下のソフトキーを押し、GO-ON および STOP の間をトグルします。  
GO-ON を選ぶと、測定値がリミット値を超えても測定を継続し、STOP を選ぶと測定値がリミット値を超えると測定を停止します。
3. DONE の下のソフトキーを押し設定を終了します。

#### リモートインターフェイスコマンド

次のコマンドを使用してリミット・テストを有効化し ONFAIL モードを設定します:

:CALCulate2:LIMit:STATe OFF|ON

:INITiate:AUTO OFF

:INITiate:AUTO ON

#### リミットテスト結果の図による表示

リミット・テスト結果を簡単な図で表示できます。ここでは 2 本の縦線以内にある○が 2 つのリミット値の範囲内の結果であることを示し、縦線の外にある×がリミット外であることを示しています。リミットテスト設定を完了し、DONE の下のソフトキーを押し、さらに Status ソフトキーを押し、Limit が表示されると、リミットテスト結果の図による表示がディスプレイの下側に現れます。

## リモートインターフェイスコマンド

次のコマンドでリミット・テスト結果の表示モードを選択します:

:CALCulate2:LIMit:DISPlay GRAPh | NUMBer

## OUT\*の指定方法

5章の説明を参照してください。

1. Limit Modes ボタンを押します。
2. OUT\*の下のソフトキーを押し、Off、USB...の間をトグルします。

※ 注: U6220A は(OUT\* USB) 機能をサポートしていません。

## リモートインターフェイスコマンド

:SYSTem:COMMunicate:SERial:CONTrol:DTR OFF|LIM

:SYSTem:COMMunicate:SERial:SOURce USB (for U6200A only)

---

## 4.6 統計演算

U6200A/220A は特定のリミット範囲内での測定値の統計演算が可能です(ピーク電圧とトータライズ測定は除きます)。STATS ボタンを押して統計演算のセットアップメニューに入ります:

- **統計演算の有効化/無効化:** フロントパネルにある MATH エリアの STATS を押して統計演算機能のオン/オフを行います。
- **表示する統計の選択:** SHOW の下のソフトキーを押し、Stdev、Max、Min、Mean、Delta、Meas、Allan の何れかを選択します。
- **測定値の表示:** SHOW の下のソフトキーを押し MEAS を選択します。
- **統計演算ごとの測定数の設定:** N Cnt の下のソフトキーを押し回数設定に入ります。数字キーパッドと矢印キーで必要な値を設定します (4.2.1 章を参照)。数値範囲は 2 ~1,000,000 です。
- **リミット内での測定に関する統計の表示(リミット値設定については 4.5 章を参照):** USE の下のソフトキーを押し、All (全測定値採用) か InLimit (リミット内の測定値採用)を選択します。
- **STOP/SINGLE ボタンが押されたときの単一または N 回測定の指定 (N Cnt により指定):** SINGLE の下のソフトキーを押し、Once か N を選択します。4.2.2 章のシングルモードオペレーションも参照してください。N とはリミット測定回数です。

※ 注：一般的に、測定分解能は $\sqrt{2}$ に比例して改善し、機器の数値処理限界まで改善することができます。

#### リモートインターフェイスコマンド

統計演算には次のコマンドを使用します：

:CALCulate3:AVERage[:STATe] OFF|ON

:CALCulate3:AVERage:COUNT <numeric value>

:CALCulate3:LFILter:STATe ON|OFF

:CALCulate3:AVERage:TYPE MAXimum | MINimum | SDEViation | MEAN or SCALar | DELTa | ALLAN

:DISPlay[:WINDow]:TEXT:FEED "CALC3"

:TRIGger:COUNT:AUTO ON | OFF

---

## 4.7 チャンネル入力条件の設定

チャンネル条件指定メニューに入るには CH1 または CH2 (U6200A のみ) ボタンを押します。このメニューでは対応チャンネルの入力条件を設定できます。入力条件設定手順は両方のチャンネルに対して同じです。

- **入力インピーダンスの設定:** IMPED の下のソフトキーを押し、1M $\Omega$  と 50 $\Omega$  の間を切り替えます。
- **入力カップリングの設定:** COUP の下のソフトキーを押し、AC と DC 間を切り替えます。
- **信号減衰の設定:** ATTEN の下のソフトキーを押し、X1 と X10 の間を切り替えます。
- **フィルターの設定:** Filter の下のソフトキーを押し、無しと 100k の間を切り替えます。
- **入力フロント/リア:** リア接続がされていれば入力 Front の下のソフトキーを押せばリアターミナルからも測定できます。但し、リア入力は、付属品 U6200-opt04 のご購入が必要になります。

※ 注：U6220A はリア入力 CH1/CH2/CH3 は提供していません。

#### リモートインターフェイスコマンド

次のコマンドを使用してチャンネル条件を割り当てます：

:INPut[1|2]:IMPedance 1E6|50 [OHM]

:INPut[1|2]:COUPling AC|DC  
:INPut[1|2]:ATTenuation 1|10  
:INPut[1|2]:FILTer ON|OFF  
:INPut[1|2]:PATH FRONt|REAR

※ 注:チャンネル 2 経由コマンドは U6200A 用のみです。

## 4.8 トリガー設定

全ての測定機能は、オート・トリガ・モードでほぼ実行されます。必要な際は、オート・トリガを手動でオフに変更することができます。

オート・トリガがオフのとき、対応する入力チャンネルが測定を開始する電圧レベルを指定することができます。オート・トリガがオンの場合、測定を開始する入力信号のパーセンテージを指定することができます。さらに、ノイズに起因するイベント・トリガの誤作動を防止するために、感度を設定することができます。

特定周波数での感度とは、測定可能な最低振幅信号のことです。アンプのゲインと、トリガ・ヒステリシス・レベル間の電圧差がカウンターの感度を決めます。感度はトリガ・レベルをセンターとする上限と下限により指定されます。入力信号は図 4-12 に示されるようにカウントをトリガするためには上限と下限両方と交わる必要があります。図 4-13 に示されるように信号ピーク間振幅が限界値を超えないとこの信号はカウントを開始しません。さらに、信号の中心より下に設定されたトリガ・レベルは図 4-14 に示されるようにリミット値を超えることがないので、カウントを行いません。

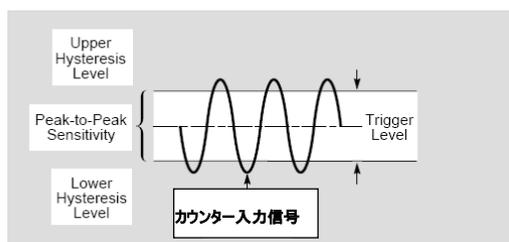


図 4-12

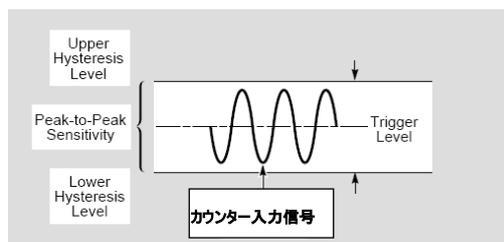


図 4-13

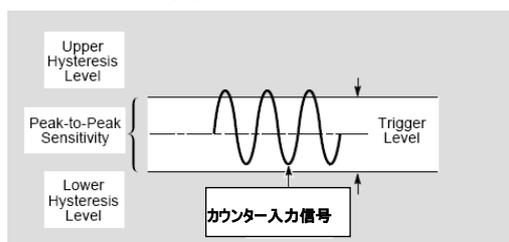


図 4-14

トリガ・メニューは入力チャンネルと関連付けられています。CH1 か CH2 (U6200A 用のみ) を押し、図 4-15 に示されるように TRIG SENS の下のソフトキーを押しメニューに入ります。次に 下記の操作ごとの指示に従ってください:



図 4-15

- オート・トリガのオン/オフ: AUTO の下のソフトキーを押し、図 4-16 に示されるように ON と OFF を切り替えます。



図 4-16

- トリガーレベルの設定: LEVEL の下のソフトキーを押し、図 4-17 に示されるように数字キーパッドか矢印キーで目的の値を入力します(4.2.1 章を参照)。

Trig Auto ON: 0 ~ 100 %  
 Trig Auto OFF: ATT X1: -5.125 ~ +5.125V  
 ATT X10: -51.25 ~ +51.25V  
 数値分解能: ATT X1: 0.0025V  
 ATT X10: 0.025V



図 4-17

- トリガ・スロープ/エッジの指定: 図 4-18 に示されるように SLOPE の下のソフトキーを押し、図 4-19 に示されるように POS(プラス)と図 4-20 に示されるように NEG(マイナス)を切替えます。



図 4-18

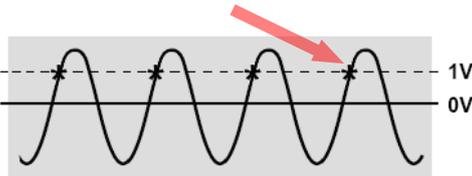


図 4-19

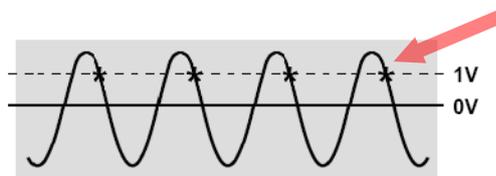


図 4-20

- **感度設定:** SENS の下のソフトキーを押し、図 4-21 に示されるように LOW、MED (medium)、HIGH を切替えます。
 

HIGH:	0 %
MED:	50 %
LOW:	100 %



図 4-21

- **Com.の設定:** 図 4.22 に示されるように、設定は時間間隔測定の場合しか有効化できません。ソフトキーCom. OFF の場合、時間間隔測定は、CH1 と CH2 に入力された信号が使われます。これに対して、ソフトキーCom. ON の場合は、時間間隔測定には、CH1 の信号のみが使われます。

## リモートインターフェイスコマンド

### オート・トリガのオン/オフ

[ :SENSE]:EVENT[1|2]:LEVel[:ABSolute]:AUTO ON|OFF

### トリガーレベルの設定

[ :SENSE]:EVENT[1|2]:LEVel[:ABSolute] <numeric value> [v]

[ :SENSE]:EVENT[1|2]:LEVel:RELative<numeric value> [PCT]

### 感度の設定

```
[:SENSe]:EVENT[1|2]:HYSTeresis:RELative 100|50|0
```

#### トリガ・スロープ/エッジの指定

```
[:SENSe]:EVENT[1|2]:SLOPe POSitive|NEGative
```

#### Common の設定

```
[:SENSe]:EVENT2:FEED “[:]INPut[1] | [:]INPut2” (U6200A only)
```

単一信号について、時間間隔 (TI) 測定を行うために Com. を使用します。

3.3 章で説明されているように、時間間隔 1 ~ 2 はチャンネル 1 でのスタート・イベントとチャンネル 2 でのストップ・イベントの間の時間を測ります。また、用途に合わせレベル、スロープまたは感度の設定を行うことができます。以下の手順で、スタートとストップ・イベントを 2 チャンネルのどちらか 1 チャンネルの単一信号で時間間隔の測定ができます。

1. 信号源をチャンネル 1 (または 2) に接続する。
2. TIME & PERIOD ボタンを押します。
3. (Intval 1 to 2) の下のソフトキーを押して時間間隔測定を選択します。
4. CH1 ボタンを押してチャンネル条件指定メニューに入ります。
5. TRIG SENS の下のソフトキーを押し、トリガー設定メニューに入ります。
6. Com. の下のソフトキーを押し、図 4-22 に示されるように ON を選択します。これで、チャンネル 1 (または 2) の時間間隔の測定ができます。



図 4-22

※ 注: U6220A は緑色の「時間間隔測定」をサポートしていません。

## 4.9 保存と再呼び出し機能

U6200A/220A は、測定設定の保存および呼び出し機能があります。

特に、この機能では測定設定、必要な精度、特定のオフセット/スケール、リミット範囲および統計を保存できます。設定は不揮発性メモリに保存され、電源を切っても消去されません。U6200A/220A は、図 4-23 に示すように、最大 20 個の異なる測定設定を保存できます。保存および呼び出し機能で対応する測定パラメータの詳細に

については、付録 B の表を参照してください。

### 測定設定の保存

1. SAVE&RECALL ボタンを押します。
2. SAVE の下のソフトキーで「保存」機能を選びます。
3. 同じソフトキーをトグルするか数字キーか矢印キーで現在の設定を保存したいメモリアドレス(1-20)を選択します。
4. Run Store の下のソフトキーを押し、図 4-23 に示されるように設定を保存します。



図 4-23

### リモートインターフェイスコマンド

\*SAV <Nrf>

### 測定設定の再呼び出し

1. SAVE & RECALL ボタンを押します。
2. Recall の下のソフトキーを押して「呼び出し」機能を選びます。
3. 同じ呼び出しソフトキーでトグルして、設定を再呼び出したかったメモリアドレスを選択します。
4. Run の下のソフトキーを押し、図 4-24 に示されるように設定を呼び出します。



図 4-24

### リモートインターフェイスコマンド

\*RCL<Nrf>

### 測定設定の消去

1. SAVE & RECALL ボタンを押します。
2. Erase の下のソフトキーで「消去」機能を選びます。
3. 同じ Erase ソフトキーでトグルし設定を消去したいメモリアドレスに入ります。
4. Run Erase の下のソフトキーを押し、図 4-25 に示されるように消去を実行しま

す。

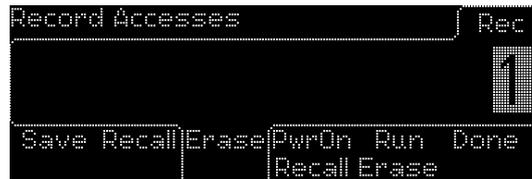


図 4-25

#### パワーオン再呼び出し方法

1. SAVE & RECALL ボタンを押します。
2. Save の下のソフトキーを押し、現行設定の保存用に 1～20 のうちからメモリ番号を選択します。矢印と数字キーも使用できます。(保存された番号ごとにメモしておくことをお勧めします。)
3. Done の下のソフトキーを押し、終了します。
4. PwrOn Recall の下のソフトキーを押し、ディスプレイに表示されている保存されたメモリ番号の間をトグルします。
5. Done の下のソフトキーを押し、パワーオンデフォルトとして使用する番号を選びます。

---

## 4.10 ユーティリティー操作

ユーティリティー操作で手動キャリブレーション、インターフェイスの構成、ビープ機能の有効化/無効化、自己診断の実行、タイムベースのセットアップおよび図 4-26 に示されるように全機能に対する最終データの表示ができます。



図 4-26

#### I/O 構成

インターフェイス構成については 5 章を参照してください。

#### キャリブレーション

U6200A では「6200」および U6220A では「6220」等セキュリティコードが必要です。正しいセキュリティコードを入力しないとキャリブレーションメニューに入れません。

1. Utility ボタンを押します。点灯信号で作動します。

2. CAL の下のソフトキーでキャリブレーション操作を選びます。
3. 図 4-27 に示されるように数字キーパッドか矢印キーで正しいセキュリティコードを入力します。
4. Cal Diag の下のソフトキーを押し、図 4-28 に示されるように Gain、Offs や T-F 等のキャリブレーション条件を手動設定します。(Save の下のソフトキーを押し、目的に応じて設定を保存することができます。)キャリブレーション情報について詳細はリスト 4-1 を参照してください。
5. すべて準備ができれば Run Cal の下のソフトキーを押します。

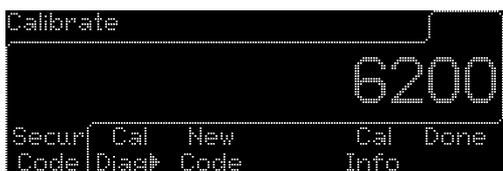


図 4-27



図 4-28

項目	ソフトキー	内容
1.	Gain Inp1 + Run Cal	DC +5V キャリブレーション@CH1
2.	Gain Inp2 + Run Cal	DC +5V キャリブレーション@CH2
3.	Gain Att1 + Run Cal	DC +9V ATT x 10 キャリブレーション@CH1
4.	Gain Att2 + Run Cal	DC +9V ATT x 10 キャリブレーション@CH2
5.	Offs Inp1 + Run Cal	DC +0V キャリブレーション@CH1
6.	Offs Inp2 + Run Cal	DC +0V キャリブレーション@CH2
7.	Offs Att1 + Run Cal	DC +0V ATT x 10 キャリブレーション@CH1
8.	Offs Att2 + Run Cal	DC +0V ATT x 10 キャリブレーション@CH2
9.	T-F クイック + Run Cal	タイムドメイン キャリブレーション (外部アームを含む)
10.	T-F Fine1 + Run Cal	時間間隔 キャリブレーション
11.	T-F Fine2 + Run Cal	時間間隔 キャリブレーション (外部アームを含む)
12.	T-F Freq + Run Cal	10 MHz キャリブレーション
13.	Save	設定をメモリに保存

リスト 4-1

※ 注: U6220A は緑で示すキャリブレーション項目をサポートしていません。

#### リモートインターフェイスコマンド

これらのコマンドでキャリブレーションを実行します:

:CALibration:SECurity:STATe <Boolean>,<NRf>

## システムのセットアップ

図 4-29 に示されるようにシステムのセットアップメニューから自己診断の実行、ビープ設定の調整、ドット表記の選択 (整数部分と少数部分の区画にカンマかピリオドを使用)および TOUT (タイムアウト「ON」が選択されていると、タイムアウトモードが有効化され、測定時間がタイムアウトのリミット値を超えたら、メッセージをコンピュータに送ります。この為、測定継続を停止できます。)を設定できます。さらに、タイムアウト「OFF」はこの機能を無効に設定します。



図 4-29

1. Utility ボタンを押します。
2. Syst の下のソフトキーでシステムのセットアップメニューに入ります。
3. ビープを有効化/無効化するには: BEEP の下のソフトキーを押し、OFF と ON を切替えます。
4. 自己診断の実行: SelfTest の下のソフトキーを押し、テストを実行します。  
(自己診断実行中、U6200A/U6220A にあるすべての接続されているソースのリードは開放されていなければなりません。次に参考までにエラーコードを掲載します。)  
0x0002:CHANNEL 1 Front End FAIL  
0x0004:CHANNEL 1 Front End FAIL  
0x0008:CHANNEL 2 Front End FAIL  
0x0010:CHANNEL 2 Front End FAIL  
0x0020:CHANNEL 1 AD FAIL  
0x0040:CHANNEL 1 AD FAIL  
0x0080:CHANNEL 2 AD FAIL  
0x0100:CHANNEL 2 AD FAIL  
0x0400:FPGA 1 FAIL
5. ドット表記の選択: Dot 9.0 の下のソフトキーを押し、ドットとカンマを切替えます。
6. オプションの表示: Show Opt の下のソフトキーを押し、内蔵オプション情報を表示します。
7. タイムアウトの有効化/無効化: TOUT の下のソフトキーを押し、OFF と ON を切替えます。

## リモートインターフェイスコマンド

\*TST?

:SYSTem:BEEPer:STATe ON|OFF

:DISPlay[:WINDow]:TEXT:RADix DPOint | COMMa

:SYSTem:TOUT ON|OFF

:SYSTem:TOUT:TIME <numeric\_value>

## タイムベースソース

タイムベースのソース選択は以下の手順で行います。

1. Utility ボタンを押します。
2. TimeBase 下のソフトキーでタイムベースソースを選択します。
3. 図 4-30 に示されるように TimeBase 下のソフトキーで AUTO、INT (内部)、EXT (外部)の間をトグルします。



図 4-30

## リモートインターフェイスコマンド

[:SENSe]:ROSCillator:SOURce INT | EXT

[:SENSe]:ROSCillator:SOURce:AUTO ON|OFF

## Data Temp

全測定の最後のデータを表示します。

## Style

Style 機能はディスプレイに値を表示させる方法です。Style 下のソフトキーで E-6 と  $\mu\text{Hz}$  表記を切替えます。 $\mu\text{Hz}$  は、エンジニアリング表記、E-6 は、指数表記を意味しています。

1. Utility ボタンを押します。
2. 図 4-31 と 図 4-32 に示されるように Style 下のソフトキーで  $\mu\text{Hz}$  と E-6 を切替えます。



図 4-31



図 4-32

---

## 4.11 表示のヒント

1. リモートコマンドエラーの表示、2. 任意のキーの HELP 表示、3. 電源 ON 時の初期状態リセット、4. 複数機器の同期化、5. PICOTEST テクニカルサポートを含むヒントを以下の手順で表示できます。

---

### 4.11.1 リモートコマンドエラーの表示

ディスプレイヒントにより、リモートコマンドエラーを容易に表示させることができます。操作手順は次の通りです。

1. フロントパネルの Help ボタンを押します。
2. ↓, ↑ と SELECT 下のソフトキーを押して項目 1 「View remote Command error」を選びます。エラーがある限りディスプレイにコマンドエラー情報が表示されます。
3. Done 下のソフトキーを押して、ディスプレイを終了して操作モードへ戻ります。エラーメッセージは消去され、コマンド SYST:ERR? が実行されるとディスプレイは+0, “No エラー”を表示します。

---

### 4.11.2 任意のキーで HELP 表示

ソフトキーを含むどのボタンを押してもサービスを提供している HELP を表示させることができます。操作手順は次の通りです。

1. フロントパネルの Help ボタンを押します。
2. ↓, ↑ と SELECT 下のソフトキーを押して項目 2 「Get HELP on any key」を選びます。HELP 情報がディスプレイに表示されます。:フロントパネルキーまたはメニューソフトキーについてのヘルプ情報を表示させるには、該当キーを数秒押し続けます。

3. Done 下のソフトキーを押し、設定を終了し操作モードに戻ります。

---

### 4.11.3 電源投入時の初期設定リセット

特定の初期設定でパワーオンできるように設定できます。操作手順は次の通りです。

1. フロントパネルの HELP ボタンを押します。
2. ↓,↑ と SELECT 下のソフトキーを押し項目 3 「Power up Default state resetting」を選びます。以下の指示が表示されるので、それに従い操作してください。  
1: [SAVE & RECALL].を押してください。2: 「PwrOn Recall」を押し、デフォルトレコード「0」を選んでください。3: 「Done」を押し設定を完了します。
3. Done 下のソフトキーを押し、ディスプレイを終了し操作モードへ戻ります。

---

### 4.11.4 複数機器の同期

複数機器の同期化ができます。操作手順は次の通りです。

1. フロントパネルの HELP ボタンを押します。  
↓,↑ と SELECT 下のソフトキーを押し項目 4 「Synchronize multiple instruments」を選びます。以下の指示が表示されるので、それに従い操作してください。例えば-1: [10MHz In] コネクタを機器間で接続してください。2: [Utility]を押します。3: 「TimeBase」を押し「自動」または「Ext」をタイムベースソースとして選択します。4: 「Done」を押します。
2. Done 下のソフトキーを押し、ディスプレイを終了し操作モードへ戻ります。

---

### 4.11.5 PICOTEST テクニカルサポート

PICOTEST の連絡先情報を表示できます。操作手順は次の通りです。

1. フロントパネルの HELP ボタンを押します。

2. ↓,↑ と SELECT 下のソフトキーを押し項目 5 「PICOTEST Technical Support」を選びます。PICOTES 連絡先情報 とウェブサイトの URL がディスプレイに表示されます。:テクニカルサポートを受けるには+886-7-815-7183 までお電話されるか、PICOTEST ウェブサイト: [Http://www.picotest.com.tw](http://www.picotest.com.tw) からテクニカルサポートへご連絡ください。
3. Done 下のソフトキーを押し、操作モードへ戻ります。

# 5 リモートインターフェイス操作

---

U6200A/220A はリモートアクセス用に 3 つのインターフェイスをサポートしています: USB、LAN、および、 GPIB (IEEE-488)。USB と LAN は U6200A/220A に搭載されています。 GPIB はオプションインターフェイスカードとしてご利用になれます。この章はインターフェイスのセットアップに加え、U6200A/220A の遠隔操作のために利用できる SCPI (プログラム可能機器用標準コマンド)コマンドリストを記載しています。SCPI コマンドを初めて使用される場合は 付録 C に詳細情報が記載されています。

※ 注: U6200A/220A は LAN ポート 5024 を Telnet セッションに使用しています。

Telnet セッションはホストコンピュータのシェルから下記の手順で開始します:

シェル:

```
telnet <IP_address> <port>
```

例:

```
telnet 192.168.0.100 5024
```

---

## 5.1 USB コネクタ一経由での Pass/Fail(合否)出力

U6200A のリアパネルにある USB コネクタはシリーズ「B」コネクタです。Limit Modes ボタンと OUT\*下のソフトキーを押し USB へ切り替えると、内部パス/フェイル TTL 出力信号 (リミットテスト)が USB ポートを通して伝送されます。

パス/フェイル信号は low true であり、次の読み取り値のパス/フェイル・リミット・テスト結果が GPIB インターフェイスに出力されることを示します。信号は読み取りデータごとに約 2ms ( $\pm 100 \mu\text{sec}$ )の間 active low 状態を保ちます。図 5-1 に USB コネクタ一(シリーズ「B」)を示します。

ピン番号	信号名	一般的な配線	内容
1	VBUS	赤	フローティング
2	D-	白	リミットテスト合格
3	D+	緑	リミットテスト不合格
4	GND	黒	GND

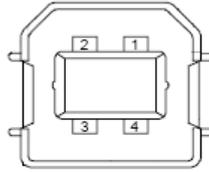


図 5-1

※ 注: 合否出力はピーク電圧 と トータライズ測定では使用できません。

※ 注: U6220A は緑の「OUT\*」機能をサポートしていません。

---

## 5.2 I/O のセットアップ

U6200A/220A アプリケーションを使用し USB、LAN または GBIB インターフェイス経由でリモート操作することができます。

### GPIB インターフェイスのセットアップ

GPIB インターフェイスカードをリアパネルのインターフェイススロットに差し込みます。インターフェイスにアドレスを割り当て、U6200A/220A アプリケーションを PC にインストールし、プログラムを実行します。コマンド コントロール用ツールタブをクリックして、コマンドを入力します。

### GPIB アドレスの割り当て方

1. Utility ボタンを押します。I/O 下のソフトキーを押して I/O メニューを選びます。
2. 「GPIB Addr」の下のソフトキーを押し図 5-2 に示されるように GPIB アドレスを指定します。
3. 数字キーパッドか矢印キーでアドレスを入力してください(4.2.1 章を参照)。



図 5-2

※ 注: GPIB アドレス範囲は 0 ~30 です。

### LAN 構成

LAN インターフェイスを構成するにはいくつかのパラメータを設定する必要があります。

LAN インターフェイスとの通信確立のための情報をネットワーク管理者から受け取ってください。

※ 注: U6200A/220AのLANはウェブサーバとジャバSEランタイム 6(アップデート 1)以降を含む環境をサポートしています。

### LAN インターフェイスの構成

1. Utility ボタンを押し I/O 下のソフトキーを押して I/O メニューを選びます。
2. LAN 下のソフトキーを押し、LAN メニューに入り、図 5-3 に示されるようにここで IP アドレスと DNS をセットアップします。

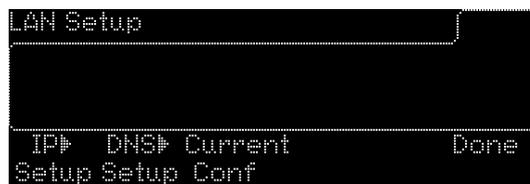


図 5-3

### IP アドレスの設定

1. IP SETUP の下のソフトキーを押し、IP アドレスメニューに入ります。
2. 図 5-4 に示されるように DHCP 下のソフトキーで ON と OFF を切り替えます。



図 5-4

※ 注: デフォルト設定アドレス 192.168.0.248 は参考までの例です。

DHCP を有効にしてあれば、IP アドレス、サブネットマスク、および、ゲートウェイは U6200A/220A をネットワークに接続すると DHCP サーバから自動的に割り当てられます。DHCP 機能を無効化してあると、手入力で IP アドレス、サブネットマスク、ゲートウェイアドレスを入力する必要があります。IP アドレスについて詳細情報はローカルネットワークの管理者にきいてください。

これらの値のセットアップ方法は同じで、全て IP アドレスフォーマット「n.n.n.n」形式によります。n は 0 と 255 の間の整数であり、手順を以下に説明します。

### IP アドレス、サブネットマスクまたはゲートウェイの設定

1. DHCP の下のソフトキーを押し、DHCP を OFF にします。
2. IP ADDR、NET MASK、または、GATEWAY の下のソフトキーを押しします。
3. 数字キーで値を入力します。
4. ENTER 下のソフトキーを押し値を設定するか、CANCEL 下のソフトキーで変更を撤回します。

### DNS のセットアップ方法

DNS (ドメイン・ネーム・サービス)はドメイン名を IP アドレスに変換します。DNS が使用されている場合、ローカルネットワーク管理者にホスト名、ドメイン名と DNS サーバのアドレスを聞いてください。図 5-5 に示されるように DNS サーバアドレスは「n.n.n.n」形式によります。n は 0 と 255 の間の整数です。



図 5-5

- **DNS サーバアドレスの設定:** LAN メニューで DNS Setup 下のソフトキーと DNS Server 下のソフトキーを押しします。数字キーか矢印キーで DNS サーバアドレスを入力します(4.2.1 章を参照)。
- **ホスト名の設定:** LAN メニューで Host name 下のソフトキーを押しします。数字キーか矢印キーでホスト名を入力します(4.2.1 章を参照)。
- **ドメイン名の設定方法:** LAN メニューで Domain name 下のソフトキーを押しします。数字キーか矢印キーでドメイン名を入力します(4.2.1 章を参照)。

### USB インターフェイスのセットアップ

U6200A/220A と PC を USB ケーブルで接続してください。U6200A/220A アプリケーションを PC にインストールし、プログラムを実行します。コマンドコントロール用ツールタブをクリックして、コマンドを入力します。

### USB ID の確認

1. Utility ボタンを押し I/O 下のソフトキーを押し I/O メニューを選びます。
2. SHOW USBID の下のソフトキーを押し、USB ID が表示されます。

### FSW のセットアップ

FSW のデフォルト設定は OFF です。OFF のときは各チャンネルが個別仕様で機能します。一方、ON にすると、CH1 と CH2 のパスが CH3 に切り替わり仕様は CH3 と同じ

になります。

### Mode のセットアップ

デフォルトモードは Native です。Native のとき、U6200A/220A はリモートで U6200A/220A の ソフトウェアに検出／認識されます。一方、Comp (互換モード)に変わると、カウンターは他のカウンターシステムと互換性を持ち、正しいコマンド文字列(コマンド:SYSTem:IDN 'string'を参照)を入力した後は、カウンターは他のソフトウェアシステムに認識され、Native のときの元の機器やソフトウェアとは機能しなくなります。

## 5.3 リモートインターフェイスコマンド

カウンター と PC が接続されて 5.1 章で説明されているようにインターフェイスが正しく構成された後ではリモート PC から SCPI コマンドで U6200A/220A に測定させることができます。U6200A/220A で利用可能な SCPI コマンドは下記クイックレファレンスリストに一覧されています。コマンドの詳細な構文や説明は付録 C に記載されています。SCPI コマンドを初めて使う場合は、付録 C.1 の SCPI についての初歩説明を参照してください。

※ 注: 次の表記規則が SCPI コマンド一覧では適用されています。山形括弧(<>)の意味はパラメタの値を自分で指定すること、角括弧([ ])はパラメタがオプションであり省略可能であることを意味しています。縦線(|) はパラメタのオプションの区切りを意味します。

<function>	<パラメタ>	[,source_list]
[[:VOLTage]:FREQuency	[<expected_value> [,<resolution>]]	[(@1) (@2) (@3)]
[[:VOLTage]:FREQuency:RATio	[<expected_value> [,<resolution>]]	[(@1),(@2) (@3)] [(@2) (@3),(@1)]
[[:VOLTage]:PERiod	[<expected_value> [,<resolution>]]	[(@1) (@2) (@3)]
[[:VOLTage]:RISE:TIME [:VOLTage]:RTIME	[<lower_reference> [,upper_reference]]	[(@1)]
[[:VOLTage]:FALL:TIME [:VOLTage]:FTIME	[<lower_reference> [,upper_reference]]	[(@1)]
[[:VOLTage]:PWIDth	[<reference>]	[(@1)]
[[:VOLTage]:NWIDth	[<reference>]	[(@1)]
[[:VOLTage]:TINterval		[(@1),(@2)]
[[:VOLTage]:PTPeak		[(@1),(@2)]
[[:VOLTage]:MAXimum		[(@1),(@2)]

[[:VOLTage]:MINimum		[(@1),(@2)]
[[:VOLTage]:CENTer		[(@1),(@2)]
[[:VOLTage]:PHASe		[(@1),(@2)]
[[:VOLTage]:DCYCLE [:VOLTage]:PDUtycycle	[<reference>]	[(@1)]
[[:VOLTage]:TOTalize:CONTInuous		[(@1)]
[[:VOLTage]:TOTalize:TIMed	[<gate_time>]	[(@1)]

※ 注: コマンド[:VOLTage]:TOTalize:CONTInuous は CONfigure でのみ可能です。

※ 注: U6220A は緑のコマンドとチャンネル 2&3 選択肢をサポートしていません。

### MEASure? コマンド

あまり柔軟性はありませんが、コマンド:MEASure?は測定のためにカウンターをプログラムするための最も簡単な方法です。コマンドで、カウンターは選択された測定機能に従い測定を実施、結果を出力バッファに送ります。

**:MEASure[:SCALar]:<function>? <parameters> [,<source\_list>]**

<function>、<parameters>、<source\_list>についての詳細は上の表を参照してください。

- **parameters** と **function** をスキップするか DEFault を使用すると、初期設定の (**parameters** と **function**)が使用されます。
- このコマンドを使用すると、自動的に統計、倍率およびオフセット、リミットテスト操作をオフにできます。
- トリガーレベルが< **parameters**>に指定されていないと、オート・トリガが有効化され、トリガ・レベル が測定チャンネル上で Volt Peak とトータライズ以外の全機能に対して自動的に設定されます。

**:MEASure**

**:MEASure[:SCALar]**

[[:VOLTage]:FREQuency? [<expected\_value> [,<resolution>]]

[,(@1)|(@2)|(@3)]

[[:VOLTage]:FREQuency:RATio? [<expected\_value> [,<resolution>]]

[,(@1),(@2)|(@3)]|[,(@2)|(@3),(@1)]

[[:VOLTage]:PERiod? [<expected\_value> [,<resolution>]]

[,(@1)|(@2)|(@3)]

[[:VOLTage]:RISE:TIME? [<lower\_reference> [,upper\_reference]][,(@1)]

[[:VOLTage]:RTIME? [<lower\_reference> [,upper\_reference]][,(@1)]

[[:VOLTage]:FALL:TIME? [<lower\_reference> [,upper\_reference]][,(@1)]

```

[:VOLTage]:FTIME? [<lower_reference> [,upper_reference]][,(@1)]
[:VOLTage]:PWIDth? [<reference>][,(@1)]
[:VOLTage]:NWIDth? [<reference>][,(@1)]
[:VOLTage]:TINterval? [(@1),(@2)]
[:VOLTage]:PTPeak? [(@1),(@2)]
[:VOLTage]:MAXimum? [(@1),(@2)]
[:VOLTage]:MINimum? [(@1),(@2)]
[:VOLTage]:CENTer? [(@1),(@2)]
[:VOLTage]:PHASe? [(@1),(@2)]
[:VOLTage]:DCYClE? [<reference>][,(@1)]
[:VOLTage]:PDUtycycle? [<reference>][,(@1)]
[:VOLTage]:TOTalize:TIMed? [<gate_time>][,(@1)]

```

※ 注: U6220A は緑のコマンドとチャンネル 2&3 選択肢をサポートしていません。

### CONfigure コマンド

CONfigure コマンド は MEASure? コマンドより若干柔軟性があり、通常は READ? コマンドといっしょに使用します。ユニバーサル カウンター は要求機能のパラメタを設定はしますが、測定は行いません。測定を開始させるには、READ? または INITiate コマンドを使用することができます。

**:CONFigure[:SCALar]:<function> <parameters> [,<source\_list>]**

<Function>、<parameters>、<source\_list> についての詳細は上表を参照してください。

- **Parameters**、**function** または **source\_list** をスキップするか、その代わりに DEFault を使用すると、初期設定の (**parameters**、**function**、**source**) が使用されます。
- このコマンドを使用すると統計演算、倍率およびオフセット操作、リミットテストは自動的にオフになります。
- < **parameters** > にトリガ・レベルが指定されていないと、オート・トリガが有効化され、トリガ・レベルは測定チャンネルに対して Volt Peak とトータライズ以外の全機能に自動的に設定されます。

**:CONFigure**

**:CONFigure[:SCALar]**

```

[:VOLTage]:FREQuency [<expected_value> [,<resolution>]]
[,(@1)|(@2)|(@3)]

```

```

[:VOLTage]:FREQuency:RATio [<expected_value> [,<resolution>]]
[,(@1)|(@2)|(@3)][,(@2)|(@3),(@1)]

```

```

[:VOLTage]:PERiod [<expected_value> [,<resolution>]]
    [,(@1)|(@2)|(@3)]
[:VOLTage]:RISE:TIME [<lower_reference> [,upper_reference]][,(@1)]
[:VOLTage]:RTIME [<lower_reference> [,upper_reference]][,(@1)]
[:VOLTage]:FALL:TIME [<lower_reference> [,upper_reference]][,(@1)]
[:VOLTage]:FTIME [<lower_reference> [,upper_reference]][,(@1)]
[:VOLTage]:PWIDth [<reference>][,(@1)]
[:VOLTage]:NWIDth [<reference>][,(@1)]
[:VOLTage]:TINterval [(@1),(@2)]
[:VOLTage]:PTPeak [(@1),(@2)]
[:VOLTage]:MAXimum[(@1),(@2)]
[:VOLTage]:MINimum [(@1),(@2)]
[:VOLTage]:CENTer [(@1),(@2)]
[:VOLTage]:PHASe [(@1),(@2)]
[:VOLTage]:DCYCLE [<reference>][,(@1)]
[:VOLTage]:PDUtYcycle [<reference>][,(@1)]
[:VOLTage]:TOTalize:CONTinuous [(@1)]
[:VOLTage]:TOTalize:TIMed [<gate_time>][,(@1)]

```

※ 注: U6220A は緑色のコマンドとチャンネル 2&3 選択肢をサポートしていません。

## READ? コマンド

READ?コマンドは頻繁に CONFIGure?コマンドといっしょに使用し、MEASure?コマンドと同じ効果を持ちます。ここでは必要なパラメタ入力、カウンター構成、測定実行はユーザーが行います。コマンド:CONFIgure? は測定の一部を設定、コマンド:READ?はデータ取得とポストプロセス(ある場合)を行い、結果を出力キューに出力します。このコマンドの組み合わせを使用してカウンターに測定セットアップからデータ取得まで実行させることができます。

### :READ[[:SCALar]:<function>]?

<Function>、<parameters>、<source\_list>についての詳細は上の表を参照してください。

READ?コマンドは、INITiate[:IMMediate]コマンドに:FETCh?コマンドを続けて使用すると同じ効果を持っています。

:READ

:READ[:SCALar]

[[:VOLTage]:FREQuency?  
[:VOLTage]:FREQuency:RATio?  
[:VOLTage]:PERiod?  
[:VOLTage]:RISE:TIME?  
[:VOLTage]:RTIME?  
[:VOLTage]:FALL:TIME?  
[:VOLTage]:FTIME?  
[:VOLTage]:PWIDth?  
[:VOLTage]:NWIDth?  
[:VOLTage]:TINterval?  
[:VOLTage]:PTPeak?  
[:VOLTage]:MAXimum?  
[:VOLTage]:MINimum?  
[:VOLTage]:CENTer?  
[:VOLTage]:PHASe?  
[:VOLTage]:DCYClE?  
[:VOLTage]:PDUtycycle?  
[:VOLTage]:TOTalize:TIMed?

※ 注: U6220A は緑のコマンドをサポートしていません。

### コマンド INITiate と FETCh?

これらのコマンドは測定トリガとデータ取得の最低レベルの制御を提供していますが、最も柔軟性があります。カウンターの構成を完了後、INITiate コマンドで測定を開始します。結果は内部メモリに送られ、読み出すまで保存されています。

### FETCh[[:SCALar]:<function>]?

<function>についての詳細は上の表を参照してください。

FETCh?コマンドは、ユニバーサルカウンターの内部メモリにあるデータを出力バッファへ送り、ここでバスコントローラに読み込むことができます。

### :INITiate

:AUTO<Boolean>  
:CONTinuous<Boolean>  
[:IMMediate]

### :FETCh

## **FETCH[:SCALAr]**

[[:VOLTage]:FREQuency?  
[:VOLTage]:FREQuency:RATio?  
[:VOLTage]:PERiod?  
[:VOLTage]:RISE:TIME?  
[:VOLTage]:RTIME?  
[:VOLTage]:FALL:TIME?  
[:VOLTage]:FTIME?  
[:VOLTage]:PWIDth?  
[:VOLTage]:NWIDth?  
[:VOLTage]:TINTerval?  
[:VOLTage]:PTPeak?  
[:VOLTage]:MAXimum?  
[:VOLTage]:MINimum?  
[:VOLTage]:CENTer?  
[:VOLTage]:PHASe?  
[:VOLTage]:DCYCLE?  
[:VOLTage]:PDUTYcycle?  
[:VOLTage]:TOTalize:TIMed?

※ 注: U6220A は緑色のコマンドをサポートしていません。

## **[:SENSe] コマンド**

[:SENSe]コマンドは、機器固有の設定を直接コントロールしますが、信号特性の設定には関与していません。

### **[:SENSe]**

:DATA? [[:SENSe[1]]]

### **[:SENSe]**

:EVENT2:FEED “[[:INPut[1]][[:INPut2]]”  
:EVENT[1|2]:HYSTEResis:RELative <numeric value> [PCT]  
:EVENT[1|2]:LEVel[:ABSolute] <numeric value> [V]  
:EVENT[1|2]:LEVel[:AUTO] <Boolean>  
:EVENT[1|2]:LEVel:RELative <numeric value> [PCT]  
:EVENT[1|2]:SLOPe POSitive|NEGative  
:EVENT3:LEVel[:ABSolute]?

:EVENT3:SLOPe?

**[:SENSe]**

:FREQuency:ARM[:START]:SLOPe POSitive|NEGative  
:FREQuency:ARM[:START]:SOURce IMMEDIATE|EXTernal  
:FREQuency:ARM:STOP:DIGits <numeric value>  
:FREQuency:ARM:STOP:SLOPe POSitive|NEGative  
:FREQuency:ARM:STOP:SOURce IMMEDIATE|EXTernal|TIMER|DIGits  
:FREQuency:ARM:STOP:TIMER <numeric value> [S]  
:FREQuency:EXPeCted[1|2|3] <numeric value> [HZ]  
:FREQuency:EXPeCted[1|2|3]:AUTO ON

**[:SENSe]**

:PHASe:ARM[:START]:SLOPe POSitive|NEGative  
:PHASe:ARM[:START]:SOURce IMMEDIATE|EXTernal

**[:SENSe]**

:TINTerval:ARM[:START]:SLOPe POSitive|NEGative  
:TINTerval:ARM[:START]:SOURce IMMEDIATE|EXTernal  
:TINTerval:ARM:STOP:SOURce IMMEDIATE|TIMER  
:TINTerval:ARM:STOP:TIMER <numeric value> [S]

**[:SENSe]**

:TOTaLize:ARM[:START]:SLOPe POSitive|NEGative  
:TOTaLize:ARM[:START]:SOURce IMMEDIATE|EXTernal  
:TOTaLize:ARM:STOP:SLOPe POSitive|NEGative  
:TOTaLize:ARM:STOP:SOURce IMMEDIATE|EXTernal|TIMER  
:TOTaLize:ARM:STOP:TIMER <numeric value> [S]

**[:SENSe]**

:VOLTage:MODE FAST | SLOW

※ 注: U6220A は緑色のコマンドとチャンネル 2&3 選択肢をサポートしていません。

**:TRACe コマンド**

:TRACe コマンドは倍率およびオフセット値のセットアップのために使用されます。

## **:TRACe**

:CATalog?  
[:DATA] OFFSET, <numeric value> [HZ|S|DEG]  
[:DATA] OFFSET, <arbitrary block>  
[:DATA]? OFFSET  
[:DATA] SCALE, <numeric value>  
[:DATA] SCALE, <arbitrary block>  
[:DATA]? SCALE

## **コマンド CALCulate**

全ての:CALCulate コマンドは、測定に対して数学的演算を実行しますが、このうち:CALCulate[1]コマンドは、倍率およびオフセット関連の演算を担当し、:CALCulate2 コマンドは、リミットテスト関連の演算を担当し、:CALCulate3 コマンドは、統計演算関連の全ての演算を担当します。

### **:CALCulate[1]**

:DATA?  
:FEED “[:]SENSe[1]”  
:IMMediate  
:IMMediate:AUTO <Boolean>  
:MATH[::EXPRession]:CATalog?  
:MATH[::EXPRession][:DEFine]?  
:MATH[::EXPRession]:NAME SCALE\_OFFSET

または

:MATH[::EXPRession]:SElect SCALE\_OFFSET  
:MATH:STATe <Boolean>

### **:CALCulate2**

:FEED “[:]CALCulate[1]”  
:IMMediate  
:IMMediate:AUTO <Boolean>  
:LIMit:CLEar:AUTO <Boolean>  
:LIMit:CLEar[:IMMediate]  
:LIMit:DISPlay GRAPH | NUMBer  
:LIMit:FAIL?  
:LIMit:FCOunt:LOWer?

:LIMit:FCOunt:UPPer?  
:LIMit:FCOunt[:TOTal]?  
:LIMit:LOWer[:DATA] <numeric\_value> [HZ|S|DEG]  
:LIMit:UPPer[:DATA] <numeric\_value> [HZ|S|DEG]  
:LIMit:PCOunt[:TOTal]?  
:LIMit:STATe <Boolean>

### **:CALCulate3**

:AVERage:ALL?  
:AVERage:CLEar  
:AVERage:COUNt <numeric\_value>  
:AVERage:COUNt:CURRent?  
:AVERage[:STATe] <Boolean>  
:AVERage:TYPE MAXimum | MINimum | SDEViation | SCALar | MEAN |  
DELTA or ALLAN  
:DATA?  
:FEED “[:]CALCulate[1]”  
:LFILter:LOWer[:DATA] <numeric\_value> [HZ|S|DEG]  
:LFILter:UPPer[:DATA] <numeric\_value> [HZ|S|DEG]  
:LFILter:STATe <Boolean>  
:PATH?

## **コマンド システム関連**

### **:ABORt**

### **:DISPlay**

:ENABle <Boolean>  
:MENU[:STATe] OFF  
[:WINDow]:TEXT:FEED “[:]CALCulate2” | “[:]CALCulate3”  
[:WINDow]:TEXT:RADix COMMa | DPOint

### **:FORMat**

[:DATA] ASCii | REAL

### **:SYSTem**

COMMunicate:SERial:CONTRol:DTR OFF | LIMit  
COMMunicate:SERial:SOURce USB

:COMMunicate:RLState REMote |LOCAl| RWLock

:ERRor?

:KEY <numeric value>

:KEY:LOG?

:VERSion?

:BEEPer:STATe ON | OFF

:IDNStr <QUOTED string>

Example: SYSTem:IDNStr "PICOTEST U6200A,0,0,0"

## CALIBRATION コマンド

U6200A/220A はカウンターを不慮の調整から保護するセキュリティコードがあります。例えば **U6200A** では「6200」、**U6220A** では「6220」などです。カウンターのhを行う前に正しいセキュリティコードを入力しなければなりません。

### :CALibration

:SECurity:STATe <Boolean>,<NRf>

:SECurity:CODE <new code>

:DATA <arbitrary block>

## その他

### :INPut[1|2]

:ATTenuation 1|10

:COUPling AC|DC

:FILTer[:LPASs][::STATe] <Boolean>

:FILTer[:LPASs]:FREQuency?

:IMPedance <numeric value> [OHM]

### :INPut3

:COUPling?

:IMPedance?

### :TRIGger

:COUNt:AUTO <Boolean>

### :MEMory

:NSTates?

## IEEE-488.2 共通コマンド

\*CLS  
\*DDT <arbitrary block>  
\*ESE <NRf>  
\*ESE?  
\*ESR?  
\*IDN?  
\*OPC  
\*OPC?  
\*OPT?  
\*RCL <NRf>  
\*RST  
\*SAV <NRf>  
\*SRE <NRf>  
\*SRE?  
\*STB?  
\*TRG  
\*TST?  
\*WAI

※ 注: U6220A は緑のコマンドとチャンネル 2&3 選択肢をサポートしていません。

# 6 手動キャリブレーション手順

---

手動キャリブレーションは次の手順で行います: UTILITY > CAL > Enter Security Code > Cal Diag > (Gain, Offs, T-F) > RunCal。 RunCal 下のソフトキーを押した後、カウンターのディスプレイは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。キャリブレーションは Save 下のソフトキーを押して保存することができます。

※ 注: オフセットキャリブレーションはゲインキャリブレーションより優先されます。

---

## 6.1 CH1 のオフセットキャリブレーションを行う方法

CH1 のオフセットキャリブレーションは DC +0V の確認のために行います。キャリブレーションを完了するには次の手順を行ってください。

1. UTILITY > CAL > Enter Security Code > Cal Diag > Offs Inp1 と押します。
  2. カウンターにつながる全てのソースを外します。
  3. RunCal 下のソフトキーを押し、キャリブレーションを実行します。次に、カウンターのディスプレイは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。
- 

## 6.2 CH2 のオフセットキャリブレーション方法

CH2 のオフセットキャリブレーションは DC +0V の確認のために行います。キャリブレーションを完了するには次の手順を行ってください。

1. UTILITY > CAL > Enter Security Code > Cal Diag > Offs Inp2 と押します。

2. カウンターにつながる全てのソースを外します。
3. RunCal 下のソフトキーを押し、キャリブレーションを実行します。次に、カウンターの ディスプレーは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。

※ 注: U6220A はチャンネル 2 のキャリブレーションをサポートしていません。

---

### 6.3 CH1 の ATT x 10 オフセットキャリブレーション方法

CH1 のオフセットキャリブレーションは ATT x 10 と DC +0V の確認のために行います。キャリブレーションを完了するには次の手順を行ってください。

1. UTILITY > CAL > Enter Security Code > Cal Diag > Offs Att1 と押します。
  2. カウンターにつながる全てのソースを外します。
  3. RunCal 下のソフトキーを押し、キャリブレーションを実行します。次に、カウンターの ディスプレーは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。
- 

### 6.4 CH2 の ATT x 10 オフセットキャリブレーション 方法

CH2 のオフセットキャリブレーションは ATT x 10 と DC +0V の確認のために行います。キャリブレーションを完了するには次の手順を行ってください。

1. UTILITY > CAL > Enter Security Code > Cal Diag > Offs Att2 と押します。
2. カウンターにつながる全てのソースを外します。
3. RunCal 下のソフトキーを押し、キャリブレーションを実行します。次に、カウンターの ディスプレーは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。

※ 注: U6220A はチャンネル 2 のキャリブレーションをサポートしていません。

---

## 6.5 CH1 のゲインキャリブレーション方法

CH1 のゲインキャリブレーションは DC +5V の確認のために行います。キャリブレーションを完了するには次の手順を行ってください。

1. UTILITY > CAL > Enter Security Code > Cal Diag > ゲイン Inp1 と押します。
2. BNC コード 2 本付き BNC T-コネクタを使用して Agilent 33220A と DMM (マルチメーター)をつなぎます。
3. Agilent 33220A の場合、操作は: Utility > DC ON > Output Setup > Hi-Z。
4. Agilent 33220A で電圧を調整し、DMM+5.000V ±0.001V を表示させます。
5. カウンターにつながる DMM から入力ソースを外します。
6. RunCal 下のソフトキーを押し、キャリブレーションを実行します。次に、カウンターのディスプレイは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。

---

## 6.6 CH2 のゲインキャリブレーション方法

CH2 のゲインキャリブレーションは DC +5V の確認のために行います。キャリブレーションを完了するには次の手順を行ってください。

1. UTILITY > CAL > Enter Security Code > Cal Diag > Gain Inp2 と押します。
2. BNC コード 2 本付き BNC T-コネクタを使用して Agilent 33220A と DMM (マルチメーター)をつなぎます。
3. Agilent 33220A の場合、操作は: Utility > DC ON > Output Setup > Hi-Z。
4. Agilent 33220A で電圧を調整し、DMM+5.000V ±0.001V を表示させます。
5. カウンターにつながる DMM から入力ソースを外します。
6. RunCal 下のソフトキーを押し、キャリブレーションを実行します。次に、カウンターのディスプレイは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。

※ 注: U6220A はチャンネル 2 のキャリブレーションをサポートしていません。

---

## 6.7 CH1 の ATT x 10 ゲインキャリブレーション方法

CH1 のゲインキャリブレーションは ATT x 10 と DC +9V の確認のために行います。

キャリブレーションを完了するには次の手順を行ってください。

1. UTILITY > CAL > Enter Security Code > Cal Diag > Gain Att1 と押します。
2. BNC コード 2 本付き BNC T-コネクタを使用して Agilent 33220A と DMM (マルチメーター)をつなぎます。
3. Agilent 33220A の場合、操作は:Utility > DC ON > Output Setup > Hi-Z。
4. Agilent 33220A で電圧を調整し、DMM+5.000V ±0.001V を表示させます。
5. カウンターにつながる DMM から入力ソースを外します。
6. RunCal 下のソフトキーを押し キャリブレーションを実行します。次に、カウンターのディスプレイは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。

---

## 6.8 CH2 の ATT x 10 ゲインキャリブレーション方法

CH2 のゲインキャリブレーションは ATT x 10 と DC +9V の確認のために行います。キャリブレーションを完了するには次の手順を行ってください。

1. UTILITY > CAL > Enter Security Code > Cal Diag > Gain Att2 と押します。
2. BNC コード 2 本付き BNC T-コネクタを使用して Agilent 33220A と DMM (マルチメーター)をつなぎます。
3. Agilent 33220A の場合、操作は:Utility > DC ON > Output Setup > Hi-Z。
4. Agilent 33220A で電圧を調整し、DMM+5.000V ±0.001V を表示させます。
5. カウンターにつながる DMM から入力ソースを外します。
6. RunCal 下のソフトキーを押し キャリブレーションを実行します。次に、カウンターのディスプレイは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。

※ 注: U6220A はチャンネル 2 のキャリブレーションをサポートしていません。

---

## 6.9 タイムドメイン・キャリブレーションの方法（外部アームを含む）

キャリブレーションは時間間隔以外のタイムドメインが対象です。キャリブレーションを完了するには次の手順を行ってください。

1. UTILITY > CAL > Enter Security Code > Cal Diag > T-F Quick と押します。
2. Agilent 33220A の場合、操作は:Utility > DC ON > Output Setup > Load。
3. 矩形波を 20MHz に調整し、Agilent 33220A で Ampl を 1.65Vpp、Offs を 0.75Vdc およびデューティーサイクルを 50%に設定します。
4. Agilent 33220A の Sync を カウンターの 外部アームに接続します。
5. Agilent 81110A の場合、操作は:MODE/TRG > Pulse-Period > internal PLL。
6. Agilent 81110A の場合、操作は:LEVELS の次に Output1 を押し Amplit を 1V 、Offset を 0V にします。
7. Agilent 81110A の場合、操作は: Timing を押し、Freq を ON に設定、1MHz に調整します。さらに output1 で Delay を 0s、DtyCyc を 50%、LeadE を 0.8ns に設定します。
8. Agilent 81110A の OUTPUT1 からカウンターの CH1 へ接続します。
9. RunCal 下のソフトキーを押し キャリブレーションを実行します。次に、カウンターのディスプレイは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。

---

## 6.10 時間間隔キャリブレーションの方法

キャリブレーションは外部アーム以外の時間間隔用です。キャリブレーションを完了するには次の手順を行ってください。

1. UTILITY > CAL > Enter Security Code > Cal Diag > T-F Fine1 を押します。
2. Agilent 81110A の場合、操作は:MODE/TRG > Pulse-Period > internal PLL。
3. Agilent 81110A の場合、操作は:LEVELS を押し、次に Output1 & Output2 ボタンと押して、Amplit を 1V、Offset を 0V に設定します。
4. Agilent 81110A の場合、操作は:Timing を押し Freq1 & Freq2 を ON、両方とも 1MHz に調整します。
5. Agilent 81110A の場合、操作は:Output1 と Output2 で Delay を 0s、DtyCyc 50 %、LeadE を 0.8ns に設定します。
6. Agilent 81110A の OUTPUT1 と OUTPUT2 からカウンターの CH1 と CH2

へそれぞれ接続します。

7. RunCal 下のソフトキーを押しキャリブレーションを実行します。次に、カウンターのディスプレイは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。

※ 注: U6220A は「時間間隔キャリブレーション」をサポートしていません。

---

## 6.11 時間間隔キャリブレーションの方法 (外部アームを含む)

キャリブレーションは外部アームを含む時間間隔を対象とします。キャリブレーションを完了するには次の手順を行ってください。

1. UTILITY > CAL > Enter Security Code > Cal Diag > T-F Fine2 と押します。
2. Agilent 33220A の場合、操作は: Utility > DC ON > Output Setup > Load。
3. 矩形波を 20MHz に調整し、Agilent 33220A で Ampl を 1.65Vpp、Offs を 0.75Vdc およびデューティーサイクルを 50%に設定します。
4. Agilent 33220A の Sync をカウンターの外部アームに接続します。
5. Agilent 81110A の場合、操作は: MODE/TRG > Pulse-Period > internal PLL。
6. Agilent 81110A の場合、操作は: LEVELS を押し、次に Output1 & Output2 ボタンを押して、Amplit を 1V、Offset を 0V に設定します。
7. Agilent 81110A の場合、操作は: Timing ボタンを押し Freq1 & Freq2 を ON、両方とも 1MHz に調整します。
8. Agilent 81110A の場合、操作は: Output1 で Delay を 0s、DtyCyc 50 %、LeadE を 0.8ns に設定します。Output2 で Delay を 125ns、DtyCyc 50%、LeadE 0.8ns に設定します。
9. Agilent 81110A の OUTPUT1 と OUTPUT2 からカウンターの CH1 と CH2 へそれぞれ接続します。
10. RunCal 下のソフトキーを押しキャリブレーションを実行します。次に、カウンターのディスプレイは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。

※ 注: U6220A は「時間間隔キャリブレーション」をサポートしていません。

---

## 6.12 標準タイムベースキャリブレーションの方法

キャリブレーション対象は 10MHz。キャリブレーションを完了するには次の手順を行ってください。

1. カウンターの電源を入れてから少なくとも 30 分ウォームアップします。
2. UTILITY > CAL > Enter Security Code > Cal Diag > T-F Freq と押します。
3. カウンターの CH1 に 10 MHz ソースを入力。
4. 10 MHz の標準仕様: 10.000,000,00 MHz  $\pm$ 1Hz。
5. RunCal 下のソフトキーを押しキャリブレーションを実行します。次に、カウンターのディスプレイは「Calibrating...」と表示します。キャリブレーションが完了すると完了したというメッセージがディスプレイから消えます。キャリブレーションに失敗すると NG メッセージが表示されます。

# 7 エラーメッセージ

---

エラーは先入れ先出し法(FIFO)で取得されます。最初に返されたエラーは最初に保存されたエラーです。使用法:SYST:ERR? でエラーを読み出します。キューから全てのエラーを読み出したら、エラーキューは空になり、エラークエリーは+0、「No エラー」を返します。エラーが発生するたびに U6200A は警告音を 1 回鳴らします。

30 個以上のエラーがある場合、キューに保存された最後のエラー(直前エラー)は-350、「キューオーバーフロー」で置き換えられます。キューからエラーを削除しない限りこれ以上エラーは保存されません。

エラーキューは電源を切るか\*CLS (ステータスクリア)コマンドを実行すると削除されません。\*RST (リセット)コマンドはエラーキューを削除しません。

---

## 7.1 コマンドエラー

- **+0 No エラー**  
エラーキューにエラーはありません。(:SYSTem:ERRor? query) またはキューは電源が入ってまたは\*CLSコマンドで削除されました。
- **-100 コマンドエラー**  
これはカウンターが特定エラーを検出できないときに使用される一般的構文エラーです。
- **-101 無効な文字**  
構文要素にその型には無効な文字が含まれます。例えば &文字を含むヘッダ  
— :INP:COUP& AC。
- **-102 構文エラー**  
認識できないコマンドまたはデータ型に行き当たりました。
- **-103 無効なセパレーター**  
パーサーがセパレーターを予期しているときに不正な文字に行き当たりました。
- **-104 データ型エラー**  
パーサーは許可されている型以外のデータ要素を認識しました。例えば、数値や文字列を予定していたところブロックデータが受信されました。

- **-105 GET 禁止**  
プログラムメッセージ内にグループ実行トリガーが受信されました。
- **-108 パラメタ禁止**  
ヘッダーに予定されているより多いパラメタが受信されました。
- **-109 パラメタ不足**  
ヘッダーに予定されているより少ないパラメタが受信されました。
- **-112 プログラムニーモニックが長すぎる**  
ヘッダーまたは文字データ要素に 12 文字以上含まれています。
- **-113 未定義ヘッダー**  
ヘッダーの構文は正しくてもカウンターに対しては未定義です。例えば、\*XYZはカウンターに対しては未定義です。
- **-120 数値データエラー**  
このエラーさらにエラー-121~-129 は非十進数値型を含む数値であると思われるデータ要素を構文解析する際生成されます。この特定のエラーメッセージはカウンターがこれより具体的なエラーを検出できないときに使用されます。
- **-121 数字に無効な文字**  
構文解析中にデータ型に対して無効な文字に行き当たりました。例えば、8 進データなのに「9」。
- **-123 指数が大きすぎ**  
数値オーバーフロー。
- **-124 桁が多すぎ**  
十進数値データ要素の仮数部に先頭の 0 も含めて 255 桁を超えて含まれています。
- **-128 数値データ禁止**  
正しい数値データ要素は受信されたがカウンターはヘッダーに対してはこの位置にこの要素を許可していません。
- **-131 無効なサフィックス**  
サフィックスはIEEE 488.2 規格に準拠していないか、カウンターに対しては不適切です。
- **-134 サフィックスが長すぎ**  
サフィックスに 12 文字を超えて含まれています。
- **-138 禁止サフィックス**  
サフィックスを許可していない数値要素の後にサフィックスに行き当たりました。
- **-141 無効な文字データ**  
無効な文字が文字データ要素に検出されました。
- **-148 文字データ禁止**  
カウンターにより禁止されていた正しい文字データ要素が検出されました。
- **-150 文字列データエラー**  
このエラーは文字列データ要素を構文解析する際生成されます。この特定のエラーメッセ

ージはカウンターがこれより具体的なエラーを検出できないときに使用されます。

- **-151 無効な文字列データ**  
文字列データ要素が予定されているのにある理由で無効でした。例えば、最後の引用符より前にENDメッセージが受信された場合。
- **-158 文字列データ禁止**  
構文解析で、文字列データ要素に行き当たりましたが、この場所では、カウンターが許可していない。
- **-160 ブロックデータエラー**  
このエラーはブロックデータ要素の構文解析時に生成されます。この特定のエラーメッセージはカウンターがこれより具体的なエラーを検出できないときに使用されます。
- **-161 無効なブロックデータ**  
構文解析で、ブロックデータ要素が期待されているのに、この場所ではカウンターが許可していません。
- **-168 ブロックデータ禁止**  
正しいブロックデータ要素に行き当たりましたがこの場所ではカウンターが許可していません。
- **-170 式エラー**  
このエラーは式データ要素の構文解析時に生成されます。このエラーはカウンターがより具体的なエラーを検出できないとき使用されます。
- **-171 無効な式**  
式データ要素は無効でした(IEEE 488.2 参照)。例えば、左右で異なる括弧や不正な文字。
- **-178 式データ禁止**  
構文解析で式データに行き当たりましたが現時点ではカウンターにより禁止されています。
- **-181 無効なマクロ外定義**  
これはマクロパラメタプレースホルダー(\$<number>)にマクロ定義の外で行き当たったことを示します。

---

## 7.2 実行エラー

- **-200 実行エラー**  
カウンターがより具体的なエラーを検出できないときの一般的構文エラーです。このコードは実行エラーの発生しか伝えていません。
- **-210 トリガーエラー**  
カウンターが:INIT、:TRIG、:ABOR サブシステムからより具体的なエラーを検出できないとき使用されます。

- **-211 トリガー無視**  
GET か\*TRG がカウンターに受信され認識されたが無視されたことを示します。
- **-213 Init 無視**  
測定開始要求が別の測定が進行中であるため無視されたことを示します。
- **-220 パラメタエラー**  
プログラムデータ要素関連エラーが発生したことを示します。このエラーはカウンターがより具体的なエラーを検出できないとき使用されます。
- **-221 設定の競合**  
正しいプログラムデータ要素は構文解析されたがカウンターの現状が原因で実行できなかったことを意味します。
- **-222 範囲外データ**  
正しいプログラムデータ要素は構文解析されたが、翻訳された値がカウンターで定義された正規範囲外であるために実行できなかったことを意味します。値は通常正規限界値までと規定されています。
- **-223 データ過多**  
ブロック、式、文字列タイプの正しいプログラムデータ要素が受信されたがメモリまたは関連カウンター固有の要件によりカウンターが処理できるよりも多いデータが含まれていたことを意味します。
- **-224 不正パラメタ値**  
可能な値のリストにもとづいて正確な値が予定されていたときに不正パラメタが使用された。
- **-230 データが破損か陳腐**  
有効なデータがありません。新規測定が始まりましたが完了しませんでした。
- **-240 ハードウェアエラー**  
正しいプログラムコマンドまたはクエリがカウンター内ハードウェアエラーが原因で実行できなかったことを意味します。
- **-241 ハードウェアがない**  
正しいプログラムコマンドまたはクエリが、ハードウェアが無いために実行できなかったことを意味します。例えば、チャンネル 3 オプションが未インストールでした。
- **-272 マクロ 実行エラー**  
正しいマクロプログラムデータシーケンスはマクロ定義に何らかのエラーがあるため実行できなかったことを意味します。

---

## 7.3 デバイス固有エラー

- **-300 デバイス固有エラー**  
これはデバイス依存一般的エラーです。
- **-310 システムエラー**  
システムエラーが発生したことを意味します。
- **-330 自己診断失敗**  
\*TST?コマンドが実行されたときに少なくとも 1 つのエラーが発生したことを意味します。
- **-350 キューオーバーフロー**  
エラーキューはいっぱいでエラーが発生しました。

---

## 7.4 クエリエラー

- **-400 クエリエラー**  
これは一般的クエリエラーです。
- **-410 クエリ中断**  
中断されたクエリエラーを発生させる事態が起きたことを意味します。例えば、前のクエリに応答する前に DAB または GET が出されました。
- **-420 クエリ未完**  
クエリエラー未完となる原因が発生したことを意味します。例えば、カウンターはトーカーになるよう指示され、不完全なプログラムメッセージが受信されました。
- **-430 クエリデッドロック**  
行き詰ったクエリエラー発生原因となる状態が発生したことを意味します。例えば、入力バッファと出力バッファともフルでありカウンターが続かなくなりました。
- **-440 不定応答後クエリ未完**  
ID を要求するクエリ(例えば、\*IDN? か \*OPT?)が実行された後、クエリが同じプログラムメッセージで受信されたことを意味します。

---

## 7.5 カウンター固有エラー

- **+2000 チャンネル 1 でオフセットキャリブレーションに失敗**
- **+2001 チャンネル 2 でオフセットキャリブレーションに失敗(U6200A 用のみ)**

- +2002 チャンネル 1 でゲインキャリブレーションに失敗
- +2003 チャンネル 2 でゲインキャリブレーションに失敗(U6200A 用のみ)
- +2004 補間器キャリブレーションに失敗
- +2005 振幅キャリブレーションに失敗
- +2006 時間間隔キャリブレーションに失敗(U6200A 用のみ)
- +2007 測定ハードウェアキャリブレーションに失敗

最後の測定で測定キャリブレーションに失敗

- +2009 無効な外部タイムベース  
外部参照が選択されたがリアパネル Ref In コネクタに外部参照は適用されていない、または外部信号は許可された周波数ではありません。
- +2010 外部タイムベースが測定中に失敗  
自動参照モードが選択されているがカウンターは外部参照が測定中に無効になり、このため現在の結果は無効であり、カウンターは内部参照使用に切り替わります。
- +2011 再呼び出しセットアップ失敗、ハードウェアエラー  
\*RCLは失敗しました。
- +2011 再呼び出しセットアップ失敗、レジスターは空  
\*RCLは再呼び出し用に指定されているレジスターが空であるため失敗しました。
- +2012 保存セットアップ失敗  
\*SAVが失敗、または\*RCLはレジスター0に保存できませんでした。
- +2013 EEPROM 失敗  
EEPROMに保存されている設定の更新要求  
:SYST:COMM:SER:CONT:DTR, :DISP[:WIND]:TEXT:RAD,  
:CAL[:DATA]  
がハードウェアエラーを起こしました。

# 付録

## A. 仕様リスト

この付録はU6200A/220Aの仕様を記載しています。AC、DC、抵抗、温度および周波数／周期特性を多様な条件の下でカバーしています。また、参考の為に、一般的特性と精度計算も記載しています。全仕様とも別途の記載無い限り、U6200A/220Aについてです。

※ 注: U6220A は緑色で示される仕様を提供していません。

チャンネル 1 & 2 (U6200A 用のみ) 入力仕様	
DC 接続時	1mHz~400 MHz
AC 接続時	200KHz~400 MHz (50 Ω) 30 Hz~400 MHz (1 MΩ)
FM 許容値	FM 許容値: 25%
電圧レンジと感度	
1mH~225 MHz	20 mVrms~±5 V ac + dc (Medium and High) 25 mVrms~±5 V ac + dc (Low)
225 MHz~400 MHz	30 mVrms~±5 V ac + dc
400 MHz~450 MHz	40 mVrms~±5 V ac + dc
チャンネル 1 & 2 (U6200A 用のみ) 入力特性	
インピーダンス (ATT X 1, 1 MΩ 静電容量) (ATT X 10, 1 MΩ 静電容量)	1 MΩ か 50 Ω 24 pF 15 pF
カップリング	AC か DC
ローパスフィルター	100 KHz (または無効) -20 dB at > 1 MHz
入力感度	Low, Medium (初期設定), High から選択 Medium は High の約 1.35 倍 Low は High の約 1.7 倍
内部ノイズ	200uVrms(代表値)
電圧レンジと感度 (シングルショットパルス)	
1.5ns~10ns パルス幅	80 mVpp~10 Vpp (150 mVpp; オプションリアコネクター)
>10 ns パルス幅	50 mVpp~10 Vpp (150 mVpp; オプションリアコネクター付き)
トリガーレベル(ATT x 1)	
レンジ	±5.125 V
精度	±(15 mV + 1% of トリガーレベル)
分解能	2.5mV

ATT x 10 レンジ	X 10	
トリガ・スロープ	プラスかマイナス	
オート・トリガ・レベル	レンジ	0~100%、1% ステップ
	周波数	ピーク電圧 fast モード>10 KHz ピーク電圧 slow モード> 100 Hz 振幅> 100 mVpp (振幅変調無し)
<b>ダメージレベル</b>		
DC~400MHz 50 Ω	12 Vrms	
0~3.5 kHz, 1 MΩ	350 V dc + ac pk	
3.5 kHz~100KHz, 1 MΩ	350 V dc + ac pk (12 Vrms までリニアに減衰)	
100KHz~400MHz, 1 MΩ	12 Vrms	
<b>アッテネーション</b>		
電圧レンジ	x10	
トリガ・レンジ	x10	

1. チャンネル 1 と 2 の仕様と特性は、コモン設定とセパレート設定ともに同じです。
2. 表記の値は減衰器 x 1 での設定の場合です。減衰器 x 10 での設定には全ての値を 10 (名目)倍してください。最大感度を得るには使用環境において(特に高温)入力オフセットを再度キャリブレーションすることが必要になる可能性があることにご注意ください。

<b>チャンネル 3 入力仕様 (U6200A 用のみ)</b>	
周波数範囲	375 MHz~6 GHz
<b>チャンネル 3 入力特性 (U6200A 用のみ)</b>	
インピーダンス	50 Ω
カップリング	AC
VSWR	< 2.5:1
<b>パワー・レンジと感度 (正弦波)</b>	
375 MHz~500 MHz	-16 dBm~+15 dBm
500 MHz~1 GHz	-20 dBm~+15 dBm
1 GHz~2 GHz	-23 dBm~+15 dBm
2 GHz~4 GHz	-25 dBm~+15 dBm
4 GHz~5 GHz	-21dBm~+15 dBm
5 GHz~5.5 GHz	-20 dBm~+15 dBm
5.5 GHz~6 GHz	-17 dBm~+15 dBm
<b>ダメージ・レベル</b>	
+25 dBm, DC ±12V	
<b>外部アーム入力仕様</b>	
信号入力レンジ	LVTTTL と TTL 互換
<b>時間的制約</b>	
パルス幅	> 50 ns
推移時間	< 250 ns
スタート~ストップ時間	> 50 ns
ダメージ・レベル	12 Vrms
<b>外部アーム入力特性</b>	
インピーダンス	1 kΩ

入力静電容量	17 pF	
スタート・スロープ	プラスかマイナス	
ストップ・スロープ	プラスかマイナス	
注	1. 外部アームは Peak Volt 以外全測定で利用できます。 2. 外部アームは一部の測定では外部ゲートとして参照されています。	
<b>内部タイムベース安定性</b>		
	標準 (0°~50°C)	
温度安定性 (基準参照~25°C)	< 1 x 10 <sup>-6</sup>	
エイジングスピード	1 日	< 0.2 x 10 <sup>-6</sup> < 1 x 10 <sup>-6</sup>
	1 月	
	1 年	
キャリブレーション	電子式	
<b>外部タイムベース入力仕様</b>		
電圧レンジ	200 mVrms~10 Vrms	
ダメージ・レベル	12 Vrms	
<b>外部タイムベース入力特性</b>		
しきい値	0 V	
インピーダンス	1 kΩ	
入力容量	25 pF	
入力周波数	10 MHz	
内部対外部タイムベース選択	手動	内部か外部かを選択
	自動	外部が無い場合は内部(初期設定)
<b>タイムベース出力仕様</b>		
出力周波数	10 MHz	
電圧	570 mVpp (0 dBm)、代表値	
インピーダンス	50 Ω(代表値)、AC 接続	
<b>測定仕様</b>		
周波数、周期;チャンネル 1&2	1mHz~400MHz (2.5ns~1000s)	
トリガ	初期設定は自動 トリガー@50 %	
“初期設定” ゲート時間	0.1 sec	
STD CH 3	375 MHz~6 GHz (0.166 ns~2.6 ns)	
周波数レシオ	CH 1/ CH 2, CH 1/ CH 3, CH 2/ CH 1, CH 3/ CH 1	
	(仕様は入力毎の全信号レンジの仕様に従います。)	
結果範囲	10 <sup>-10</sup> ~10 <sup>11</sup>	
「初期設定」ゲート時間	0.1 sec	
時間間隔	仕様は、チャンネル 1 と 2 の全信号レンジの仕様に従います。 パルス幅は 1 ns 以上。周波数範囲~300 MHz.	
トリガ	初期設定はオート・トリガ@ 50 %	
結果範囲	-0.5 ns~10 <sup>5</sup> s	
分解能	40 ps	
RMS 分解能	120 ps	
システム不確実性	±(TI x タイムベースエラー) ± トリガ・レベル タイミングエラー ± 500 ps 差分チャンネルエラー	
パルス幅時間	仕様は、チャンネル 1 の全信号レンジの仕様に従います。 パルス幅 は 1 ns 以上。周波数範囲~300 MHz。	
パルス選択	プラスかマイナス	

トリガ	初期設定はオート・トリガ@ 50 %		
結果範囲	1.5 ns~105 s		
分解能	40 ps		
RMS 分解能	120 ps		
システム不確か性	± (パルス幅 時間 x タイムベースエラー) ± トリガ・レベル タイミングエラー ±500 ps 差分チャンネルエラー		
立ち上がり/立ち下がり時間	仕様は、チャンネル 1 の全信号レンジの仕様に従います。 パルス幅 は 1 ns 以上。周波数範囲~300 MHz。		
エッジ 選択	プラスかマイナス		
トリガ	初期設定はオート・トリガ@ 10%と 90%		
結果範囲	2 ns~105 s		
分解能	40 ps		
RMS 分解能	120 ps		
システム不確か性	± (エッジ 時間 x タイムベースエラー) ± トリガ・レベル タイミングエラー ± 500 ps 差分チャンネルエラー		
フェーズ	仕様は入力毎の全信号レンジの仕様に従います。 パルス幅 は 1 ns 以上。周波数範囲~300 MHz		
結果範囲	-180°~+360°		
分解能	40 ps		
RMS 分解能	120 ps		
システム不確か性	± (トリガ・レベル タイミングエラー) × 周波数		
デューティーサイクル	仕様は、チャンネル 1 の全信号レンジの仕様に従います。 パルス幅 は 1 ns 以上。周波数範囲~300 MHz		
パルス選択	プラスかマイナス		
トリガ	初期設定はオート・トリガ@ 50 %		
結果範囲	0~1		
分解能	40 ps		
RMS 分解能	120 ps		
システム不確か性	± トリガ・レベル タイミングエラー ±500 ps 差分チャンネルエラー		
トータライズ	仕様は、チャンネル 1 の全信号レンジの仕様に従います。 パルス幅 は 1 ns 以上。周波数範囲~400 MHz		
パルス選択	プラスかマイナス		
トリガ	初期設定はトリガ@0 V		
結果範囲	0~10 <sup>15</sup>		
分解能	1 カウント		
システム不確か性	± 1 カウント		
ピーク電圧	結果範囲	-5.1 V~+ 5.1 V	
	分解能	2.5 mV	
DC 信号		15 mV + V の 2 %	ピーク間 振幅 > 200 mV
DC 信号 (ATT x 10)		150 mV + V の 2 %	ピーク間 振幅 > 1 V
1 Vp-p、50 Ω、ATT OFF	100 Hz ~ 10 KHz	15 mV + V の 2 %	ピーク間 振幅 > 200 mV
	10 KHz ~ 5 MHz	15 mV + V の 4 %	
	5 MHz ~ 80 MHz	15 mV + V の 7 %	
	80 MHz ~ 300 MHz	15 mV + V の 15 %	
<b>リア入力オプションチャンネル絶縁</b>			
周波数	フロントチャンネル(dB)	リアチャンネル(dB)	
100KHz	<-85	<-85	

1MHz	-85	-75
10MHz	-78	-55
50MHz	-67	-42
100MHz	-62	-37
200MHz	-55	-35
300MHz	-50	-36
400MHz	-47	-33

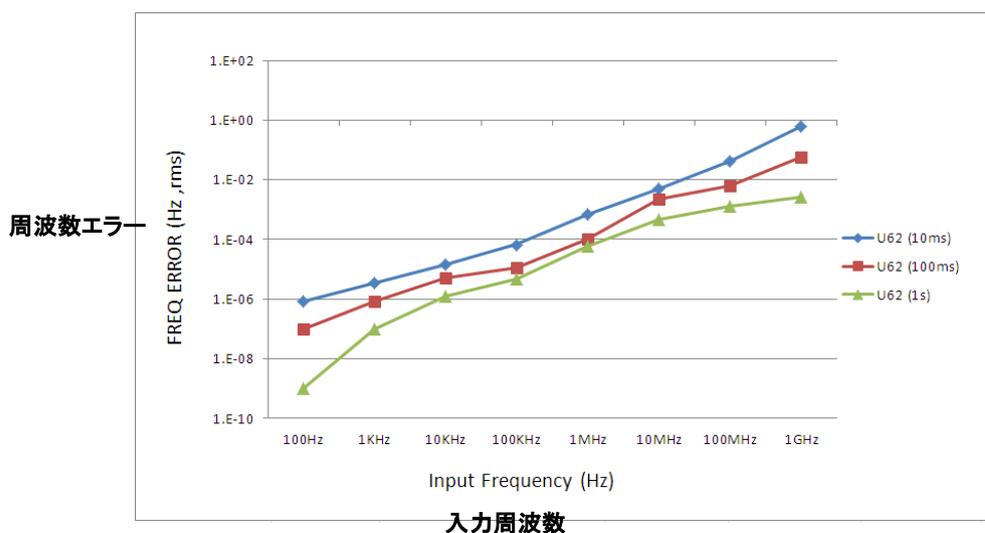
ピーク電圧測定は結果は参照のみとなっても、400 MHz まで継続します。

多くのシステム不確実性等式で見られる差分チャンネルエラー項の結果はチャンネル間の不一致と内部ノイズであり、これらの問題は温度がほどよく管理された環境での、TI キャリブレーションで改善されます。

## A.1 RMS 分解能

Rms	100Hz	1KHz	10KHz	100KHz
U62 (10ms)	0.000000842	0.000003438	0.000013896	0.000067275
U62 (100ms)	0.0000001	0.000000828	0.000005093	0.000011508
U62 (1s)	0.000000001	0.000000098	0.000001212	0.000004869

Rms	1MHz	10MHz	100MHz	1GHz
U62 (10ms)	0.000677504	0.004830870	0.042107484	0.611551072
U62 (100ms)	0.000101040	0.002272900	0.006344503	0.055991810
U62 (1s)	0.000058166	0.000469601	0.001275299	0.002602258



このグラフは トリガ・エラーの影響は無視しています。このエラー項の追加的影響に上限を設定するには、周波数エラー

—を正しいグラフから特定し、トリガ・エラー項を次の方程式から求めることが必要になります:

$$\text{周波数エラー} \quad \text{トリガーエラー} \quad \text{周波数 または 周期}$$

$$\text{Frequency Error} + \left( \frac{2 \times \sqrt{2} \times \text{Trigger Error}}{\text{Gate Time} \times \sqrt{\text{Number of Samples}}} \right) \times \text{Frequency or Period}$$

ゲート時間                      サンプル数

外部ソースと入力アンプノイズは測定の開始と終了を定義している トリガ点を進めるか遅らせ、その結果タイミング不確かさは 信号のスルーレートとスプリアスノイズスパイク(入力ヒステリシスバンドに対する)の関数となります。

シングルトリガ・ポイントに対する(rms) トリガ・エラー:

$$\text{トリガ・エラー} \quad \text{(秒)}$$

$$\text{Trigger Error} = \frac{\sqrt{(E_{input})^2 + (E_{signal})^2}}{\text{Input Signal Slew Rate at Trigger Point}} \quad \text{(in seconds)}$$

トリガ・ポイントでの入力信号スルーレート

## B. プリセット値とセーブ/リコール情報

※ 注: U6220A は緑色のチャンネル 2 経由機能は提供していません。

		*RST での値 ( GPIB リセット)	Save/Recal 内	不揮発性メモリ内
入力インピーダンス	CH1	1E+6 Ohms	Yes	no
	CH2	1E+6 Ohms	Yes	no
入力アッテネーション	CH1	x1	Yes	no
	CH2	x1	Yes	no
トリガ・レベル	CH1 (%)	50	Yes	no
	CH2 (%)	50	Yes	no
	CH1 (V)	0	Yes	no
	CH2 (V)	0	Yes	no
トリガ・スロープ	CH1	プラス	Yes	no
	CH2	プラス	Yes	no
感度	CH1	Medium	Yes	no
	CH2	Medium	Yes	no
倍率		1	Yes	no
オフセット		0	Yes	no
リミット・パラメタ	リミットテストオン/オフ	オフ	Yes	no
	エラー時停止/継続	継続	Yes	no
	下限リミット値	0	Yes	no
	上限リミット値	0	Yes	no

Stats パラメタ	Stats オン/オフ	オフ	Yes	no
	測定カウント	100	Yes	no
	ディスプレイ measurement/ stats	measurement	Yes	no
	全データ/リミット内	全データ	Yes	no
	オンシングル測定	1	Yes	no
タイムベース		auto	Yes	no
トリガ・オフセット Cal パラメータ	チャンネル 1 トリガ・オフセット Inp1 cal		no	yes
	チャンネル 2 トリガ・オフセット Inp2 cal		no	yes
	チャンネル 1 トリガ・オフセット Att1 cal		no	yes
	チャンネル 2 トリガ・オフセット Att2 cal		no	yes
トリガ・ゲイン Cal パラメータ	チャンネル 1 トリガ・ゲイン Inp1 cal		no	yes
	チャンネル 2 トリガ・ゲイン Inp2 cal		no	yes
	チャンネル 1 トリガ・ゲイン Att1 cal		no	yes
	チャンネル 2 トリガ・ゲイン Att2 cal		no	yes
時間間隔 オフセット Cal パラメータ	ファイン 1		no	yes
	ファイン 2		no	yes
	クイック		no	yes
タイムベース cal パラメータ			no	yes

## C. 一般仕様

※ 注: U6220A は緑色の付属品 1 ~ 2 はサポートしていません。

項目	制限と説明
電源 電圧	100V/240V ± 10% 、50Hz~60Hz ± 10%
	100V/120V ± 10% 、400Hz ± 10%
電源要件	最大 80VA (30W 代表値)
動作湿度	最大相対湿度 80%、(温度 31 °Cまで) 40°Cの相対湿度 50%まではリアに減衰
動作環境	0~55 °C
保管温度	- 40 °C~70 °C
動作高度	最大 2000m
ラックの寸法 (WxHxD)	214.6 x 88.6 x 346.9 mm
重量	3130 g / 2887 g
安全性	IEC61010-1:2010/EN61010-1:2010 (3 <sup>rd</sup> Edition) IEC61010-2-030:2010(1 <sup>st</sup> Edition)

項目	制限と説明
	EN61010-2-030:2010+
EMC	EN61326、IEC61000-3、IEC61000-4
ウォームアップ時間	1 時間
保証	1 年
付属品	1. U6200-opt04: リアパネル入力モジュール(CH1/CH2) 2. U6200-opt05: リアパネル入力モジュール(CH1/CH2/CH3) 3. M3500-opt04: GPIB カード

※ 注 1: 付属品は Picotest で組み立てるオプションです。

※ 注 2: デバイスを結線したり外したりし易い位置に機器を配置してください。

※ 注 3: 電源コードを不正定格コードに交換しないようにしてください。

## D. リモートインターフェイス・リファレンス

### D.1 SCPI 言語への導入

SCPI (プログラム可能機器用標準コマンド)はASCIIベース命令コマンド言語で、測定器を対象として構築されています。「プログラミング概要」に、リモートインターフェイスからのプログラミング用基本テクニックが記載されています。

SCPIコマンドはツリーシステムとしても周知の階層構造に基づいています。このシステムにおいては関連するコマンドは共通ノード (node) あるいはルート (root) のもとにグループ化され、サブシステムを成しています。SENSE サブシステムの一部がツリーシステムを説明する目的で下記に示されています。

**SENSe:**

TOTalize:ARM[:START]:SLOPe POSitive|NEGative

PHASe:ARM[:STOP]:SLOPe: POSitive|NEGative

SENSe はコマンドのルート(root)キーワードであり、TOTalizeとPHASeは 2 次レベルキーワードおよびARMは 3 次レベルキーワードです。コロン(:) はコマンドキーワードを低レベルキーワードと分離します。

### このマニュアルで使用されるコマンドのフォーマット

このマニュアルで コマンド表記用のフォーマットは次の通りです:

**[:SENSe]:TINterval:ARM[:STARt]:SLOPe Positive|Negative**

※ 注: U6220A は緑のコマンドをサポートしていません。

コマンド構文は大部分のコマンド(および一部のパラメタ)を大文字と小文字の混合で表記します。大文字 はコマンドの省略スペルを示します。プログラム行が短い場合は省略形を送ります。プログラムが読みやすいために長い形式で送ります。

例えば、上記の構文ではSENSとSENSe 両方とも許容され、大文字か小文字を使用することができます。従って、SENSE、sensおよびSens全て許容されます。SENとSEのような形式ではエラーになります。

縦線(|)は、コマンド文字列に対する複数のパラメタ選択肢を分離します。

山括弧(<>) は囲まれたパラメタに対する値を指定しなければならないことを示します。例えば、上記構文で、rangeパラメタが山括弧に囲まれています場合に、括弧はコマンド文字列とともに送れないので、パラメタの値を指定する必要があります。(「INPut:COUPling AC」等)。

一部のパラメタは角括弧 ([])で囲まれています。この括弧はパラメタがオプションであり省略可能であることを示します。括弧はコマンド文字列とともに送られません。オプションパラメタの値が未指定のままだと、カウンターは初期設定値を選びます。

### コマンドセパレーター

コロン(:)はコマンドキーワードを低レベルキーワードから分離するために使用されます。コマンドキーワードからパラメタを分離するためには、空白スペースを挿入します。コマンドが 2 つ以上のパラメタを要求する場合、下記のようにカンマで隣接パラメタを分離します:

:READ:SCALar:FALL:TIMe

セミコロン(;) は同じサブシステム内部でコマンドを分離するために使用され、タイピングを最小限ですますことができます。例えば、次のコマンド文字列を送ることは:

```
:DISP:MENU OFF; TEXT:FEED "CALC2"
```

次の 2 つのコマンドを送るのと同義です:

```
:DISP:MENU OFF
```

```
:DISP:TEXT:FEED "CALC2"
```

コロンとセミコロンでコマンドを異なるサブシステムからリンクします。例えば、次のコマンド 文字列では、コロンとセミコロンを使用しないとエラーが生成されます:

```
:DISP:MENU OFF;:CALC2:LIM:DISP NUMBER
```

### パラメタ設定のクエリ

コマンドに疑問符(?)を付けることにより大部分のパラメタの現在値を クエリすることができます。例えば、次のコマンドは入力減衰を 10 に設定します:

```
:INPut:ATTenuation 10
```

入力減衰を次のコマンドを実行してクエリできます:

```
:INPut:ATTenuation?
```

**※注意:** 最初から応答を読まずに 2 つのクエリコマンドを送り、2 つ目の応答を読み取ろうとすると、一部のデータは最初の応答から、続いて完全な応答を 2 番目の応答から受信することになる可能性があります。これを回避するため、クエリコマンドは応答を読まずに送らないでください。この状況を回避できない事情があれば、デバイスクリアを送ってから、2 番目のクエリコマンドを送ります。

## SCPI コマンド ターミネータ

カウンターへ送られるコマンド文字列は<new line>文字で終わっていることが必要です。IEEE-488 EOI (end-or-identify)メッセージは<new line>文字として解釈され、コマンド文字列を<new line>文字に代わって終了するために使用することができます。<carriage return>の次に<new line>とするのも許容されます。コマンド文字列終了は、現在のSCPI コマンドパスを常にrootレベルへリセットします。

## IEEE-488.2 共通コマンド

IEEE-488.2 規格は、リセット、自己診断および状態操作等の機能を実行する共通コマンドのセットを定義しています。共通コマンドは常に星マーク(\*)で始まり、4~5 文字長で、1 つまたはそれ以上のパラメータを持つことができます。コマンドキーワードは空白スペースにより最初のパラメータから分離されます。下記のようにセミコロン(;)で複数コマンドを分離してください :

\*RST; \*CLS; \*ESE 32; \*OPC?

## SCPI パラメータ タイプ

SCPI言語はプログラムメッセージと応答メッセージで使用されるいくつかの異なるデータフォーマットを定義しています。

数値パラメータを要求する **数値パラメータ・コマンド**は、全ての共通に使用される十進表記を受け入れます。これにはオプション符号、小数点、科学記号が含まれます。

**ディスクリートパラメータ** ディスクリートパラメータは、限られた数の値(IMMEDIATEやEXTERNAL等)を持つプログラム設定に使用されます。コマンドキーワードと同様に短縮形と長い形式とがあります。大文字と小文字を混在させることができます。クエリ応答は常に、全てが大文字の短縮形を返します。次のコマンドはディスクリートパラメータを使用しています:

:INITiate[:IMMEDIATE]

**Boolean パラメータ** ブーリアン・パラメータ は真(true)か、偽(false)のシングルバイナリ条件を表します。偽の場合、カウンターは「OFF」または「0」を受け入れます。真の場合、カウンターは「ON」または「1」を受け入れます。ブーリアン設定をクエリすると、

機器は常に、「0」か「1」を返します。次のコマンドは、ブーリアンパラメータを表しています:

```
INPut:IMPedance:AUTO OFF|ON
```

**String パラメータ** String(文字列)パラメータは事実上任意のASCII文字のセットでも含むことができます。文字列は同種の引用符で括られなければなりません: シングルクォートかダブルクォートです。その間に文字を入れずにクォートを二回タイプすれば、文字列の部分としてクォートによる区切りを含めることができます。次のコマンドは文字列パラメータを使用しています:

```
DISPlay:TEXT "CALC2"
```

## D.2 コマンドリファレンス

**:ABORt**

実行中の測定をできるかぎり迅速に中止します。

※ 注: U6220A は緑色で示される一部のコマンドと指示をサポートしていません。

**注:**

- 測定サイクルがアイドルの間に、このコマンドが送られると (:INIT:CONT OFF および保留演算フラッグは偽)、無視されます。
- このコマンドが単一測定進行中に(:TRIG:COUN:AUTO OFF または :CALC3:AVER OFF、:INIT:CONT OFF および保留中演算フラッグが真)送られると、測定は中止され、保留演算フラッグは偽に設定されます。
- このコマンドが繰返し測定中の送られると (:INIT:CONT ON)、現在進行中の測定が中止され、保留演算フラッグは偽に設定されます。次に新たな測定が自動的に開始され、実行中演算フラッグは真に設定されます。
- このコマンドが一連の測定の進行中に送られると(:TRIG:COUN:AUTO ON および :CALC3:AVER ON、:INIT:CONT OFF および保留演算フラッグが真)、測定が中止され、保留演算フラッグは偽に設定され 演算 Status Register にある測定ビットは偽に設定されます。
- 実行中測定を終了すると 結果は無効となりますが、自動的にアームされたトータライズ測定を終了するには適切な方法です。

## コマンド :CALCulate[1]

:SENSe コマンドにより取得された倍率およびオフセットデータの処理または転送を実行します。:CALCulate[1]:MATH:STATe ON を送り、:CALCulate[1]コマンド前に演算を開始させることを確実に行ってください。

### :CALCulate[1]:DATA?

スケールおよびオフセット演算による測定結果を問い合わせます。

### Return

- 結果は :FORMat[:DATA] ASCii|REAL 設定に従いフォーマットされます。
- ASCii フォーマットを使用する際、数値データは<NR3>フォーマットの ASCII バイトとして転送されます。有効数字は測定の分解能に依存して 1~15 の範囲になります。有効数字のみ返されます。
- 有効な結果が無いと、エラー -230 が生成され、9.91E37(Not A Number)が返されます。
- トータライズか Volt Peak が選ばれると、エラー 0221 が生成され、9.91E37(Not A Number)が返されます。

### 注:

- このコマンドが測定中に送られると、結果は測定完了までは利用できず、デバイスクリアか電源再投入が無い限り後続コマンドが機能しません。
- 最後に計算された結果は、関連機器の状態に対して新たな計算が修正されるまで有効であり続けます。

### :CALCulate[1]:FEED “[:]SENSe[1]”

CALCulate[1] ブロックにデータを代入します。

### Return

- 文字列「SENS」が返されます。

### :CALCulate[1]:IMMEDIATE

カウンターにデータを再取得せず現在のデータを再計算するように指示します。

このコマンドは次の内容に影響しません:

:CALC2:LIM:FCO

:CALC2:LIM:PCO

:CALC3:...

## Return

- 結果は:FORMat[:DATA] ASCii|REAL の設定に従いフォーマットされます。
- ASCii フォーマットを使用する際、数値データは<NR3>フォーマットで ASCII バイトとして転送されます。有効数字は測定分解能に依存して 1~15 の範囲になります。有効数字のみ返されます。
- 有効な結果が無いと、エラー -230 が生成され、9.91E37(Not A Number)が返されます。
- トータライズか Volt Peak が選択されると、エラー 0221 が生成され、9.91E37(Not A Number)が返されます。

## 注:

- :CALC:IMM:AUTO ON のとき:CALCulate[1|2]コマンドのいずれかに変更があると再計算が自動的に行われます。

## :CALCulate[1]:IMMediate?

新しい計算の結果をクエリします。

## 注:

コマンド INIT:CONT OFF と CALCulate[1]:IMMediate:AUTO OFF が入力されると、コマンド:CALCulate[1]:IMMediate?は有効になります。

## :CALCulate[1]:IMMediate:AUTO <Boolean>

:CALCulate[1|2]コマンドのいずれかに変更があれば、常に再計算が自動的に行われるかどうかを設定します。

OFF に設定されていれば、:CALCulate[1|2] コマンドは新しいデータが取得されなくても、コマンドが実行されると新しい結果を生成します。また、これで新しい構成が変更された直後、同じ現在のデータに関する新しい結果が得られます。

## Return

- ASCII エンコード型バイト。0 の意味は OFF、1 は ON です。

## 注:

- 再計算コマンドの全設定に影響します。
- これは、カウンターがオフに切り替えられたか、リモートインターフェイスがリセットされるた後で OFF に設定されます。

## :CALCulate[1]:MATH[:EXPRession]:CATalog?

定義済み方程式名をクエリします。

#### Return

- 文字列「SCALE\_OFFSET」を返します。

#### :CALCulate[1]:MATH[:EXPRession][:DEFine]?

計算で使用された方程式をクエリします。

#### Return

- 改行と EOI で終わる ASCII エンコード型バイトのシーケンス("SENS"\*SCALE+OFFSET)

#### 注

- 終了したプログラムメッセージにおける最後のクエリであることが必要です、そうでないとエラー-440 が生成されます。

#### :CALCulate[1]:MATH[:EXPRession]:NAME SCALE\_OFFSET

または

#### :CALCulate[1]:MATH[:EXPRession]:SElect SCALE\_OFFSET

数学的演算で使用される式の名前を定義します。

#### Return

- ASCII エンコード型バイトのシーケンス:SCALE\_OFFSET.

#### :CALCulate[1]:MATH:STATe <Boolean>

数学的演算を有効化または無効化します。

#### Return

- ASCII エンコード型バイト: 0 の意味は OFF、1 は ON です。

#### 注

- このコマンドはリミット数をクリアします(:CALC:LIM:FCO, :CALC:LIM:PCO)。

#### コマンド:CALCulate2

このグループはリミット・テストとデータ転送を実行します。

#### :CALCulate2:FEED "[[:]CALCulate[1]]"

CALCulate2 ブロックに代入されるべきデータフローを設定またはクエリします。

#### Return

- 文字列「CALC」が返されます。

### **:CALCulate2:IMMEDIATE**

データを再取得せず、現在のデータを再計算するように指示します。

このコマンドは次の内容に影響しません:

:CALC2:LIM:FCO

:CALC2:LIM:PCO

:CALC3:...

### **注**

- :CALC:IMM:AUTO が ON のときは:CALCulate[1|2]コマンドのいずれかに変更があると、再計算は自動的に行われます。

### **:CALCulate2:IMMEDIATE:AUTO <Boolean>**

:CALCulate[1|2]コマンドのいずれかに変更があれば、常に再計算が自動的に行われるかどうかを設定します。

OFF に設定されていれば、:CALCulate[1|2] コマンドは新しいデータが取得されなくてもコマンドが実行されると新しい結果を生成します。また、これで新しい構成が変更された直後に、同じ現在のデータに関する新しい結果が得られます。

### **Return**

- ASCII エンコード型バイト: 0 の意味は OFF、1 は ON です。

### **注**

- カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは OFF に設定されます。
- 再計算コマンドの全設定に影響します。
- ポストプロセス可能な唯一のリミット結果は:CALC2:LIM:FAIL?です。リミット数 (:CALC2:LIM:FCO と :CALC:LIM:PCO) は、データ取得時にリミットテストされた測定結果を反映します。

### **:CALCulate2:LIMit:CLEar:AUTO <Boolean>**

:INITiate[:IMMEDIATE]か :INITiate:CONTinuous ON コマンドに行き当たったときに、リミットテスト結果をクリアすべきかどうかを設定します。

これが OFF に設定されていると、:CALC2:LIM:CLE[:IMM] コマンドのみリミットテスト結果をクリアします。

### **Return**

- シングル ASCII エンコード型バイト: 0 の意味は OFF、1 は ON です。

#### 注

- カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは ON に設定されます。
- これが ON に設定されると、:INIT[:IMM] か:INIT:CONT ON が実行される場合は、常に次が実行されます:

リミットデータはクリアされます。

:CALC2:LIM:FAIL, :CALC2:LIM:FCO 及び :CALC2:LIM:PCO 情報はクリアされます。

RS-232 コネクタのリミット検出出力は、リミット内電圧レベルに設定されます。

#### :CALCulate2:LIMit:CLEar[:IMMediate]

直ちにリミットデータを無効化し、:CALC2:LIM:FAIL, :CALC2:LIM:FCO と :CALC2:LIM:PCO 情報をクリアし、RS-232C コネクタのリミット検出出力をリミット内電圧レベルにセットします。

#### 注

- リミット状態が OFF に設定されている場合は、エラー -221 が生成されます。

#### :CALCulate2:LIMit:DISPlay GRAPH|NUMBER

数値かグラフで表示するためのコマンドを実行するため、ディスプレイは、数値およびグラフの両方の結果を表示します。

#### Return

- ASCII エンコード型バイトのシーケンス。GRAP はグラフィックを意味し、NUMB は数値を意味します。

#### :CALCulate2:LIMit:FAIL?

リミットテストされた最終測定の状態をクエリします。

#### Return

- シングル ASCII エンコード型バイトが返されます。0 の意味はリミットテストに合格した最終測定、1 の意味は不合格です。

#### 注

- 有効な測定結果が無い場合、「0」が返され、エラー -230 が生成されます。
- このコマンドがリミットテストオン状態で、測定進行中に送られると、測定が完了するまで応答しません。さらにデバイスクリアまたは電源サイクルが無い限り、後続コマンドも実行されません。
- 現在の測定機能が、トータライズまたは Volt Peak の場合、「0」が返され、エラー -221 が生成されます。

**:CALCulate2:LIMit:FCOunt:LOWer?**

下限値テストに不合格であった測定数をクエリします。

**Return**

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

**注**

- 限界値テストがオフの場合、「0」が返され、エラー -221 が生成されます。
- 有効な測定結果が無い場合、「0」が返され、エラー -230 が生成されます。
- 現在の測定機能がトータライズまたは Volt Peak であれば、「0」が返され、エラー -221 が生成されます。

**:CALCulate2:LIMit:FCOunt[:TOTal]?**

リミットテストに不合格の測定合計数をクエリします。

**Return**

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

**注**

- リミットテストがオフの場合、「0」が返され、エラー -221 が生成されます。
- 有効な測定結果が無い場合、「0」が返され、エラー -230 が生成されます。
- 現在の測定機能がトータライズまたは Volt Peak の場合、「0」が返され、エラー -221 が生成されません。

**:CALCulate2:LIMit:FCOunt:UPPer?**

アッパーリミットテストに不合格であった測定数をクエリします。

**Return**

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

**注**

- リミットテストがオフの場合、「0」が返され、エラー -221 が生成されます。
- 有効な測定結果が無い場合、「0」が返され、エラー -230 が生成されます。
- 現在の測定機能がトータライズまたは Volt Peak の場合、「0」が返され、エラー -221 が生成されません。

#### **:CALCulate2:LIMit:PCOunt[:TOTal]?**

リミットテストに合格した測定合計数をクエリします。

#### **Return**

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

#### **注**

- リミットテストがオフの場合、「0」が返され、エラー -221 が生成されます。
- 有効な測定結果が無い場合、「0」が返され、エラー -230 が生成されます。
- 現在の測定機能がトータライズまたは Volt Peak の場合、「0」が返され、エラー -221 が生成されません。

#### **:CALCulate2:LIMit:STATe <Boolean>**

リミットテスト状態を定義またはクエリします。

#### **Return**

- シングル ASCII エンコード型バイトが返されます。0 の意味は OFF、1 は ON です。

#### **注**

- カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは OFF に設定されます。
- リミットデータと:CALC2:LIM:FAIL、:CALC2:LIM:FCO と :CALC2:LIM:PCO の情報は、クリアされます。
- リミット検出出力をリミット内電圧レベルに設定します。

#### **:CALCulate2:LIMit:LOWer[:DATA] <い numeric value> [HZ|S|DEG]**

リミットテストでの下限値を設定またはクエリします。

#### **Return**

- <NR3>フォーマットで ASCII バイトとして転送された 11 桁の有効数字。
- 結果が下限値未満であるとフェイルが返されますが、結果が下限値に等しい場合は、フェイルになりません。

#### **注**

- カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後、下限値は 0 に設定されます。
- 新たな下限値を設定すると、リミット値はクリアされます(:CALC2:LIM:FCO、:CALC2:LIM:PCO)。

- リミットテストがオンの場合(:CALC:MATH:STATe ON)、リミット値を指定する際は、倍率とオフセットの結果を考慮する必要があります。

#### **:CALCulate2:LIMit:UPPer[:DATA] <numeric value> [HZ|S|DEG]**

リミットテストでの上限値を設定またはクエリします。

#### **Return**

- <NR3>フォーマットで ASCII バイトとして転送された有効数字 11 桁からなる数値。
- 結果が上限値を超えているとフェイルが返されますが、結果が上限値に等しい場合は、フェイルとは報告しません。

#### **注**

- カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後、上限値 は 0 に設定されます。
- 新たな上限値を設定するとリミット値はクリアされます(:CALC2:LIM:FCO, :CALC2:LIM:PCO)。
- リミットテストがオンの場合(:CALC:MATH:STATe ON)、リミット値を指定する際は、倍率とオフセットの結果を考慮する必要があります。

### **コマンド:CALCulate3**

このグループは統計演算とデータ転送を実行します。

#### **:CALCulate3:AVERage:ALL?**

統計結果 4 つ全てをクエリします: 平均値、標準偏差、最大値、最小値。

#### **Return**

- <NR3>フォーマットで ASCII バイトとして転送される数値データ。分解能に依存する有効数字 1~15 桁の数。

#### **注**

- 統計演算はこのコマンドを使用する前に有効化されなければなりません(:CALC3:AVER[:STATe] ON)。もしそうでなければ、異常値 9.91E37 が返され、エラー -221 が生成されます。
- 結果はカンマで分離され、表示順は平均値、標準偏差、最小値、最大値です。
- 有効な結果が無いと、異常値 9.91E37 が返され、エラー -230 が生成されます。
- 現在の測定機能がトータライズか Volt Peak であれば、異常値 9.91E37 が返され、エラー -221 が生成されます。

#### **:CALCulate3:AVERage:CLEar**

統計データをクリアし、統計カウントを 0 に設定し、Operation Status Register ビット 8 にマイナス状態条

件をレポートします。

**注**

- 統計演算が有効化されていない場合(:CALC3:AVeR[:STATe] OFF)、エラー -221 が生成されます。

**:CALCulate3:AVeRage:COUNt <numeric value>**

統計演算で使用すべき測定数を設定またはクエリします。この数値の範囲 2 から 1,000,000 までです。

**Return**

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

**注**

- カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後、これはカウンターの初期値 100 に設定されます。
- 指定数の測定結果が収集された時、新たな統計計算を行う前に、もう一つの測定セットが収集される必要があります。

**:CALCulate3:AVeRage:COUNt:CURRent?**

統計計算のためにこれまでに収集された測定数をクエリします。範囲は 0 から 1,000,000 までです。

**Return**

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

**注**

- 統計演算が有効化されていないと(:CALC3:AVeR[:STATe] OFF)、エラー -221 が生成されます。
- 現在の測定機能がトータライズか Volt Peak の場合、0 が返され、エラー -221 が生成されます。

**:CALCulate3:AVeRage[:STATe] <Boolean>**

統計演算状態を設定またはクエリします。

**Return**

- シングル ASCII エンコード型バイト。0 の意味は OFF、1 は ON です。

**注**

- カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは OFF に設定されます。

**:CALCulate3:AVERage:TYPE MAXimum|MINimum|SDEviation|SCALar|MEAN|DELTA or ALLAN**

:CALC3:DATA?に対する統計的結果を設定し、:DISP[:WIND]:TEXT:FEED が「CALC3」に設定されているときに表示します。

#### Return

- ASCII エンコード型バイトのシーケンス。MAX、MIN、SDEV、または、MEAN があります。

#### 注

- ・ カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後、初期値である MEAN に設定されます。

**:CALCulate3:DATA?**

:CALC3:AVER:TYPE コマンドで得られる統計結果の選択をクエリします。

#### Return

- :FORMat[:DATA] ASCii|REAL コマンドでフォーマットされた結果。ASCII フォーマットを使用する場合、<NR3>フォーマットで転送される数値データを返します。有効桁数の範囲は分解能に依存し、1 から 15 までです。

#### 注

- 結果をクエリする前に統計演算を有効化してください。
- 統計演算が有効化されていない(:CALC3:AVER[:STATe] OFF)か、または現在の測定機能がトータライズまたは Volt Peak の場合、異常値 9.91E37 が返され、エラー-221 が生成されます。
- 有効な結果が無いと、異常値 9.91E37 が返され、エラー -230 が生成されます。
- 最後に計算された統計結果は、新しい計算が行われるか、関連の状態が変更されるまで、有効です。

**:CALCulate3:FEED “[:]CALCulate[1]”**

CALCulate3 ブロックに代入されるべきデータフローを、設定またはクエリします。

#### Return

- 文字列「CALC」が返されます。

**:CALCulate3:LFILter:LOWer[:DATA] <numeric value> [HZ|S|DEG]**

統計演算での下限値を設定またはクエリします。分解能 11 桁で範囲は-9.9999990000E+12 から -1.0000000000E-13、0.0000000000 および+1.0000000000E-13 から 9.9999990000E+12 までです。

## Return

- <NR3>フォーマットで ASCII バイトとして転送される有意な 11 桁数値データ。

## 注

- カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後、電源投入時の初期設定 0 に戻ります。
- このコマンドは:CALC:LIM:LOW コマンドと同じ値を共有しています。
- リミット・フィルタリングが有効化されている場合 (:CALC:LFIL:STAT ON)、この値より低いどの値も統計計算に使用されません。
- 数学的演算が有効化されている場合 (:CALC3:MATH:STATe ON)、リミット値を指定する際、倍率とオフセット測定を考慮する必要があります。
- 新たな下限値を設定するとリミット値はクリアされます (:CALC2:LIM:FCO, :CALC2:LIM:PCO)。
- ディスプレーがこの値の全ての有効桁を表示できない場合、表示される値は丸められている為、実測値とは異なります。しかし、数値がまだ表示されている間に ENTER を押すと、実測値に更新されます。

## :CALCulate3:LFILter:UPPer[:DATA] <numeric value> [HZ|S|DEG]

統計演算での上限値を設定またはクエリします。分解能 11 桁で範囲は-9.9999990000E+12 から -1.0000000000E-13、0.0000000000 および+1.0000000000E-13 から 9.9999990000E+12 までです。

## Return

- <NR3>フォーマットで ASCII バイトとして転送される有効数字 11 桁の数値データ。

## 注

- カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後、電源投入時の初期設定 0 に戻ります。
- このコマンドは:CALC:LIM:UPP コマンドと同じ値を共有しています。
- リミット・フィルタリングが有効化されている場合 (:CALC:LFIL:STAT ON)、この値より高いどの値も統計計算に使用されません。
- 数学的演算が有効化されていると (:CALC3:MATH:STATe ON)、リミット値を指定する際、倍率とオフセット測定を考慮する必要があります。
- 新たな上限値を設定するとリミット値はクリアされます (:CALC2:LIM:FCO, :CALC2:LIM:PCO)。
- ディスプレーがこの値の全ての有効桁を表示できない場合、表示される値は丸められている為、実測値とは異なります。しかし、数値がまだ表示されている間に ENTER を押すと、実測値に更新されます。

#### **:CALCulate3:LFILter:STATe<Boolean>**

統計フィルターリング状態を設定またはクエリします。

#### **Return**

- シングル ASCII エンコード型バイト。1 の意味は OFF、1 は ON です。

#### **注**

- カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後は 0 に戻されま  
す。
- これがオンであるときは、リミット値以内の測定値 (math がオンのときは倍率とオフセット)のみ統計  
計算に使用されます。

#### **:CALCulate3:PATH?**

CALCulate3 サブブロックが処理されるべき順序をクエリします。

#### **Return**

- ASCII エンコード型バイトのシーケンス: LFIL, AVER.

#### **注**

- この機器に対しては、シーケンスは LFILter 次に AVERage に固定されています。

### **コマンド:CALibration**

#### **:CALibration:SECurity:CODE<new code>**

キャリブレーション用セキュリティコードを設定します。範囲は 0 から 99,999,999 および分解能  
は 1 です。初期値のセキュリティ有効化コードは 6200 です。

#### **注**

- セキュリティコードは不揮発性メモリに保存されており、電源投入、保存/再呼び出しまたは  
\*RST により影響されません。
- 状態がセキュリティコードの要件であれば、正しいコードを入力します。その後コードを変  
更したいときは、新規コードを入力します。次に Enter ボタンを押して保存します。

#### **:CALibration:SECurity:STATe <Boolean>,<NRf>**

キャリブレーションのセキュリティステートを設定またはクエリします。

セキュリティ解除するには、有効なセキュリティコードで OFF と指定します。カウンターのセキュ  
リティ解除が完了すると、どのようなキャリブレーションでも行うことができます。

セキュリティを設定するには、有効なセキュリティコードで ON と指定します。キャリブレーション

に対してセキュリティ確定されていると、キャリブレーションは補間器 (interpolator) キャリブレーション以外行えなくなります。

#### Return

- シングル ASCII エンコード型バイト。0 は、セキュリティが掛かっていないこと、1 はセキュリティが掛かっていることを意味します。

#### 注

- セキュリティステートは不揮発性メモリに保存されており 電源投入、保存/再呼び出しまたは \*RST により影響されません。
- セキュリティコード は 6200 に設定されています (カウンターの出荷時設定) セキュリティコードを忘れた場合:CALibration:SECurity:CODE:reset コマンドでデフォルト状態に戻せます。

:CALibration:SECurity:CODE:RESet

ユーザー定義コードの忘れを防ぎ、迅速にキャリブレーションモードにアクセスできるように、コマンドを実行してデフォルト状態に戻せます。

#### コマンド:CONfigure

コマンド:CONfigure と:READ は、コマンド:MEASure より柔軟性を提供しています。測定構成部分を提供しますが、測定を行うこともクエリ結果も出力しません。コマンド:CONFigure を使用すると、倍率およびオフセット演算、統計、リミットテストは全て無効化されます。トリガーレベルが<parameter>に指定されていないときは、オート・トリガとオート・トリガレベルが有効化され、トータライズと Volt Peak 以外全ての測定機能に対して設定されます。このセクション後半にあるコマンド:MEASure?を参照すると機能ごとの<parameter>と<source list>について詳細が説明されています

※注: このコマンドを使用するときは、いくつかの初期設定が自動的に使用されます。設定を変えずに別の機能へ切り替えるには、[:SENS]:FUNC[:ON]を使用します。

:CONFigure[:SCALar]:<function> <parameter> [,<source\_list>]

指定された機能を測定を開始することなく実行するには、コマンド:INIT:FETC?か READ?を使用し、測定を行い、クエリします。<parameter>と[,<source\_list>]をスキップするとデフォルト状態に戻ります。

#### :CONFigure?

最後の:CONfigureor :MEASure コマンドにより構成された機能をクエリします。

## Return

- 「<function> <parameter> [,<source list>]」形態での文字列が返されます。
- このクエリより前に指定が無い場合、空の文字列が返され、エラー -221,「setting conflict」が生成されます。
- 応答は、\*RST、recall、または[:SENS]:FUNC に影響されません。

## 注

このセクション後半にある(コマンド:MEASure?)を参照すると機能ごとの<parameter>と<source list>について詳細が説明されています

## Device Clear

入力バッファと出力キューをクリアさせ、パーサー、実行コントロールおよびレスポンスフォーマッターをリセットします。コマンド \*RST などの処理を妨げるコマンドをすべてクリアします。直前に実行されたコマンド\*OPC の効果を無効化し、\*WAI, \*OPC? または、保留中の操作が終了するのを待っているデータクエリ (:MEASure クエリ、:READ クエリ、:FETCh クエリ、CALC:DATA?、:CALC2:LIM:FAIL?)の待機中アクションを終了します。

**注:** フロントパネルから開始された診断またはキャリブレーションは終了され、ユーザー入力を待ちます。

### :DISPlay:ENABle <Boolean>

全ディスプレイが有効化されるかどうかを設定またはクエリします。

## Return

- シングル ASCII エンコード型バイト。0 は無効化、1 は有効化を意味しています。

## 注

- \*RST コマンドの後、ディスプレイは有効に戻ります。
- この値は save&revall に影響されません。

### :DISPlay:MENU[:STATe] OFF

メニュー表示を無効化し 結果を標示します。応答は、メニュー又は結果が表示されているか否かです。

## Return

- シングル ASCII エンコード型バイト。0 は結果表示が有効化されていること、1 はメニュー表示が有効化されていることを意味します。

## 注

- 表示の状態は OFF に戻ります。これは、\*RST コマンドの後、メニュー表示が無効になり、結果表示が有効になることを意味します。
- :SYST:KEY コマンドを使用するか、またはフロントパネルキーからメニューを表示します。
- この値は SAVE&RECALL に影響されません。

## :DISPlay[:WINDow]:TEXT:FEED “[:]CALCulate2”|[:]CALCulate3”

どのデータフローがディスプレイに送られるかを設定またはクエリします。

“[:]CALCulate2”---は統計以外の結果を結果表示へ送るために使用されます。

結果は次のうち一つになります:

- 測定結果 : 数学的機能が有効化されていないとき
- 倍率/オフセット含む測定結果 : 演算機能が有効化されているとき
- リミットグラフ : リミットテストが有効化されグラフモードが選択されているとき。

“[:]CALCulate3”--- 統計結果を結果表示に送るために使用されます。特定の統計結果表示は::CALC3:AVER:TYPE により定義されます。

## Return

- 文字列「CALC2」または「CALC3」が返されます。

## 注

- これは\*RST コマンドの後で「CALC2」に戻されます。

## :DISPlay[:WINDow]:TEXT:RADix COMMa|DPOint

表示される数字の整数部分と少数部分を分離するために使用される記号のセットまたはクエリ。

COMMaを使用すると、1 千は 1.000,0 と表記され、DPOintを使用すると 1,000.0 と表記されます。

## Return

- ASCII エンコード型バイトのシーケンス:COMM または DPO。

## 注

- この値は不揮発性メモリに保存され、カウンターの電源が切られた後や\*RST コマンドまたは SAVE&RECALL 後に変更されません。

### :FETCh[:SCALar]:<function>?

このコマンドは、コマンド:INItiate (または :MEASure query または :READ?)により取得された測定機能をクエリします。

このコマンドはコマンド:FETCh?の一般的形式です。<function>と<parameter>をスキップすると、最後のコマンド :CONFigure, :MEASure, :READ、または:FETCh で指定された機能が指示されます。[:SCALar]:<function> が指定されている場合、現在の測定機能と一致する場合、または現在の測定機能から導き出せる場合に、指定された結果を返します。次の表は、どの機能が、どの機能から導き出せるかを示しています。

周波数	↔	期間
Volt Peak	↔	電圧最大
Volt Peak	↔	電圧最小
Volt Peak	↔	電圧 CENTER
電圧最大	↔	Volt Peak
電圧最小	↔	Volt Peak
電圧 CENTER	↔	Volt Peak

<function> オプションについて詳細は 5.4 章の表をご参照ください。

## Return

- 結果は :FORMat[:DATA] ASCii|REAL 仕様に従いフォーマットされます。
- ASCii フォーマットを選択すると、数値データは<NR3>フォーマットで ASCII バイトとして転送されます。有効桁数は測定分解能に依存して 1~5 の範囲になります。
- 有効な結果が無いと、異常値 9.91E37 が返され、エラー-230 が生成されます。
- 自動的にアームされたトータライズ測定が実行中のときは、:FETCh?コマンドを送るとエラー-221を発生させ、異常値 9.91E37 が返されます。このエラーを回避するには、:ABORt を使用して測定をまず終了します。

## 注

- 測定進行中にこの コマンドを送ると現在の測定が完了するまでその他のコマンドの実行は留保されます。しかし Device Clear を使用するか電源を入れてこの保留中の実行

予定をキャンセルすることができます。

#### **:FORMat[:DATA] ASCii|REAL**

データのフォーマットタイプを設定またはクエリします。

ASCii 型を選ぶと、数値データは <NR3>フォーマットで ASCII バイトとして送られます。数字は IEEE488.2 で指定されているようにカンマで区切られます。応答データが無いときは、異常値 9.91E37 が返されます。

REAL 型を選ぶと、応答データは 64 ビット IEEE754 浮動小数点数値として <definite length block> で送られます。応答データが無いときは、異常値 9.91E37 が <definite length block> に返されます。

#### **Return**

- ASCII エンコード型バイトのシーケンス:ASCii または REAL.

#### **注**

- \*RST コマンドの後は、データフォーマット型が ASCii に戻ります。
- このコマンドは次のコマンドの応答データタイプをコントロールします：

:CALCulate:DATA?

:CALCulate3:DATA?

:FETCh?

:MEASure query

:READ?

[:SENSe]:DATA?

:TRACe[:DATA] query

#### **:INITiate:AUTO <Boolean>**

測定値が指定されているリミット値を超えたときは、カウンターの測定停止か継続かを設定またはクエリします。

測定値が指定されているリミット値を超えたとき、AUTO OFF は測定を停止させます。測定がリミットテストにフェイルしたときでも、AUTO ON (set :INIT:CONT to OFF) は、測定を継続させます。

#### **Return**

- シングル ASCII エンコード型バイト。0 の意味は AUTO OFF、1 の意味は AUTO ON です。

## 注

- \*RST コマンドの後は、AUTO OFF に戻されます。

### **:INITiate:CONTinuous <Boolean>**

連続的測定開始の有効化を設定またはクエリします。

CONTinuous が ON に設定されているとき、測定は連続的に開始されます。CONTinuous が OFF に設定されていると、:INITiate[:IMMediate] コマンドを受けるまで、または CONTinuous が ON に設定されるまで、測定を開始しません。

### **Return**

- シングル ASCII エンコード型バイト。0 の意味は CONTinuous OFF、1 は CONTinuousON です。

## 注

- \*RST コマンドの後は、CONTinuous OFF に戻されます。
- :INIT:CONT ON コマンドが送られると:
  - 統計結果を無効化し
  - 現在の統計カウントをクリアして 0 にし、
  - 演算状態レジスタのビット 8 に、マイナスステータス状態 (統計演算しない。)を送ります。
- :CALC2:LIM:CLE:AUTO が ON で、:INIT:CONT ON コマンドを受けたときは、常に:
  - リミット値データを無効化し、
  - リミットテストのパス・フェイル数(:CLAC2:LIM:FCO と:CALC2:LIM:PCO)をクリアします。
  - リミット値検出出力をリミット値内電圧レベルに設定します。
- [:SENSe]:EVEN:LEV[:ABS]:AUTO が ON で:INIT:CONT ON コマンドが送られると、:INIT CONT が ON の間は、各測定サイクルの開始時に、ソースチャンネルをオートトリガします。
- :TRIG:COUN:AUTO の状態は:INIT:CONT ON 操作に影響しません。
- :TRIG:COUN が 1 であったかのように:INIT:CONT ON は動作します。
- :INIT:CONT ON と設定して最初の測定が開始されると、未完のオペレーション・フラグを真に設定します。測定を中止するか:INIT:CONT が OFF に設定後最後の測定の完了により、保留中のオペレーション・フラグは偽に設定されます。
- 測定が連続的に行われているときは、:ABORt コマンドは実行中測定を中断しますが、:INIT:CONT の値を変更しません。:INIT:CONT が:ABORt 以前に ON に設定されているときは、ON のままであり、新しい測定が開始されます。

- 単一測定進行中に(:INIT:CONT は OFF):
  - エラー -213 (Init ignored) が生成され、状態 INIT:CONT は:INIT:CONT ON に影響されません。
  - エラー -210 (トリガ・エラー)は:INIT:CONT OFF により生成されます。
- :INIT:CONT は、電源が切られた後には ON に設定され、\*RST コマンドの後に OFF に設定されます。

#### **:INITiate[:IMMEDIATE]**

シングル測定または特定数の測定を開始するように指示します。

次の場合は、シングル測定を開始します:

:TRIG:COUN:AUTO が OFF

または

:CALC3:AVER[:STAT]が OFF

または

[:SENS]:FUNC[:ON]が Totalize

または

[:SENS]:FUNC[:ON]が任意の Volt Peak。

次の場合は、測定ブロック全体を開始します:

:TRIG:COUN:AUTO が ON

および

:CALC3:AVER[:STAT]が ON

および

[:SENS]:FUNC[:ON]が、Totalize、Volt Peak のどちらでもない。

詳しくは:TRIG:COUN:AUTO を参照ください。

#### **注**

- :TRIG:COUN:AUTO と:CALC3:AVER[:STAT] がともに ON のとき、:INIT[:IMM]を受信すると、統計結果と現在のカウント数をクリアします。
- :CALC2:LIM:CLE:AUTO が ON で、:INIT:CONT ON コマンドを常に受信する場合:
  - リミットデータを無効化
  - リミットテストのパス/フェイル・カウントを削除 (:CALC2:LIM:FCO および :CALC2:LIM:PCO)
  - リミット検出出力をリミット内電圧レベルに設定します。
- [:SENSe]:EVEN:LEV[:ABS]:AUTO が ON で:INIT[:IMM]コマンドが送られると、ソースをオート・トリガします。
- 測定中または :INITiate:CONTinuous が ON に設定されていると、:IMMEDIATE コマンド

は何も効果を持たず、エラー -231 (Init ignored)が生成されます。

- 測定または測定ブロックを:INIT[:IMM]で開始すると、保留のオペレーショフラグを true に設定します。測定または測定ブロックを完了する(正常、または中止により)と、保留のオペレーショフラグを false に設定します。
- 自動的にアームされたトータライズ測定に対しては、:ABORt を使用して、:INIT[:IMM]で開始された測定を終了します。

#### **:INPut[1|2]:ATTenuation 1|10**

入力アッテネーションを設定またはクエリします。

#### **Return**

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

#### **注**

- 入力アッテネーションは、\*RST コマンドの後、X1 に設定されています

#### **:INPut[1|2]:COUPling AC|DC**

入力カップリングを設定またはクエリします。

#### **Return**

- ASCII エンコード型バイトのシーケンス: AC または DC。

#### **注**

- 入力カップリング は、\*RST コマンドの後、AC に設定されます。

#### **:INPut[1|2]:FILTer[:LPASs][:STATe] <Boolean>**

ローパスフィルターの状態を設定またはクエリします。

#### **Return**

- シングル ASCII エンコード型バイト。0 の意味は OFF、1 は ON です。

#### **注**

- \*RST コマンド後は off に設定されます。

#### **:INPut[1|2]:FILTer[:LPASs]:FREQuency?**

ローパスフィルターのカットオフ周波数をクエリします。

#### **Return**

- <NR3>フォーマットで ASCII バイトとして転送される有効数字 6 桁の数値データ。

- 値 100E+3 が返されます。

#### **:INPut[1|2]:IMPedance <numeric value> [OHM]**

入力インピーダンスを設定またはクエリします。50Ω か 1MΩ のいずれかです。

#### **Return**

- <NR3>フォーマットで ASCII バイトとして転送される有効数字 6 桁の数値データ。

#### **注**

- 入力インピーダンスは、\*RST コマンドの後 1E6 OHM に設定されます。

#### **:INPut3:COUPling ?**

チャンネル 3 入力カップリングをクエリします。

#### **Return**

- ASCII エンコード型バイトのシーケンス: AC

#### **:INPut3:IMPedance?**

チャンネル 3 入力インピーダンスをクエリします。

#### **Return**

- <NR3>フォーマットで ASCII バイトとして転送される有意な 6 桁の数値データ。
- 返される値は 50 です。

### **コマンド:MEASure?**

コマンド:MEASure は測定を行う最も簡単な方法です。1 つのコマンドだけでカウンターを構成し、測定を開始し、結果をクエリします。

**:MEASure[:SCALar]:<function>?<parameter> [,<source\_list>]**

上のコマンドは:MEASure?コマンドの一般的形態です。<function>と<parameter>をスキップすると、初期設定を使用するように指示します。各コマンド :MEASure? の説明は下記に一覧されています。

**:MEASure[:SCALar][:VOLTage]:DCYCle? [<reference>][,@1]**

測定アーミングを「auto」に設定し、デューティサイクルを指定パラメタで測定します。

<reference>はデューティサイクルが決定されるパルス上の点を指定し、%でも絶対電圧でも指定可能です。初期設定は%です。

<reference>パラメタはトリガ設定の構成に使用されます。これが初期設定の%に指定されていると、オート・トリガが有効化されトリガ・レベルは指定値です。電圧に指定されていると、オート・トリガが無効化され、トリガ・レベルは指定値です。

<reference>	初期設定	50 %
	%範囲	0~100 pct
	%分解能	10%
	電圧レンジ	-5.125V~5.125V アッテネーション×1 -51.25V~51,25V、アッテネーション×10
	電圧分解能	0.0025V、アッテネーション×1 0.025V、アッテネーション×10
<source list>	初期設定	(@1)
	範囲	(@1)

#### :MEASure[:SCALar][:VOLTage]:FALL:TIME?

[<lower\_reference> [<upper\_reference>]][,(@1)]

測定アーミングを「auto」に設定し、指定パラメタをもつ立ち下がり時間を測定します。

<lower\_reference>と<upper\_reference>は入力信号の下降エッジ上の上下ポイントを指定し、%または絶対電圧として指定することができます。

<lower\_reference>および<upper\_reference>はトリガ設定の構成に使用されます。これが初期値の%に指定されていると、オート・トリガが有効化されトリガ・レベルは指定値です。電圧に指定されていると、オート・トリガが無効化され、トリガ・レベルは指定値です。

<lower_reference>	初期設定	10 %
	%範囲	0~100 pct
	%分解能	10%
	電圧レンジ	-5.125V~5.125V、アッテネーション×1 -51.25V~51,25V、アッテネーション×10
	電圧分解能	0.0025V、アッテネーション×1 0.025V、アッテネーション×10
<upper_reference>	初期設定	90 %
	%範囲	0~100 pct
	%分解能	10%
	電圧レンジ	-5.125V~5.125V、減衰×1 -51.25V~51,25V、減衰×10
	電圧分解能	0.0025V、アッテネーション×1

		0.025V、アッテネーション×10
<source list>	初期設定	(@1)
	範囲	(@1)

**:MEASure[:SCALar][:VOLTage]:RISE:TIME?**

**[<lower\_reference> [<upper\_reference>]][(@1)]**

測定アーミングを「auto」に設定し、指定パラメタをもつ立ち上がり時間を測定します。

<lower\_reference>と<upper\_reference>は入力信号の立上りエッジ上の上下ポイントを指定し、%または絶対電圧として指定することができます。

<lower\_reference>および<upper\_reference>はトリガー設定の構成に使用されます。これが初期設定の%に指定されていると、オート・トリガが有効化されトリガ・レベルは指定値です。電圧に指定されていると、オート・トリガが無効化され、トリガ・レベルは指定値です。

<lower_reference>	初期設定	10 %
	%範囲	0~100 pct
	%分解能	10%
	電圧レンジ	-5.125V~5.125V、アッテネーション×1 -51.25V~51,25V、アッテネーション×10
	電圧分解能	0.0025V、アッテネーション×1 0.025V、アッテネーション×10
<upper_reference>	初期設定	90 %
	%範囲	0~100 pct
	%分解能	10%
	電圧範囲	-5.125V~5.125V、アッテネーション×1 -51.25V~51,25V、アッテネーション×10
	電圧分解能	0.0025V、アッテネーション×1 0.025V、アッテネーション×10
<source list>	初期設定	(@1)
	範囲	(@1)

**:MEASure[:SCALar][:VOLTage]:FREQUENCY?**

**[<expected\_value> [<resolution>]][(@1)]( @2)( @3)**

測定アーミングを「digits」に設定し、周波数を測定します。

<expected value>と<resolution>パラメタは分解能アーミング設定の桁数構成に使用されます。

チャンネル 1 とチャンネル 2 は同じトリガー設定を共有しており、測定チャンネルはオート・トリガが+スロープで50%に設定されています。

<expected value> Ch1, Ch2	初期設定	10 MHz
	レンジ	0.100 Hz~400 MHz
	分解能	<expected value>は最適なアーミング構成を得るには入力周波数の10%以内であるべきです。
<expected value> Ch3	初期設定	500 MHz
	レンジ	375 MHz~6GHz
	分解能	<expected value>は最適なアーミング構成を得るには入力周波数の10%以内であるべきです。
<resolution> Ch1, Ch2	初期設定	設定された<expected value>の分解能 4 桁を示す値
	レンジ	1E-16 Hz~1E6 Hz (15~3 桁分解能)
	分解能	<resolution>は仮数 1.0 を使用し、10 の偶数乗の必要があります。
<resolution> Ch3	初期設定	設定された<expected value>の分解能 4 桁を示す値
	範囲	1E-7 Hz~1E7 Hz (15 ~3 桁分解能)
	分解能	<resolution>は仮数 1.0 を使用し、10 の偶数乗の必要があります。
<source list>	デフォルト	(@1)
	範囲	(@1 @2 @3)

**:MEASure[:SCALar][:VOLTage]:FREQuency:RATio?**

**[<expected\_value> [,<resolution>]][,(@1),(@2)](@1),(@3)](@2),(@1)](@3),(@1)]**

測定アーミングを「digits」に設定し、2 入力間の周波数レシオを測定します。

<expected value>と<resolution>パラメタは分解能アーミング設定の桁数構成に使用されます。

<expected value>	初期設定	1
	レンジ Ch1/Ch2, Ch1/Ch3	1E-10~1E11
	レンジ Ch2/Ch1, Ch3/Ch1	1E-11~1E10
	分解能	<expected value>は最適なアーミング構成を得るにはレシオの 10%以内であるべきです。
<resolution>	初期設定	設定された<expected value>の分解能 4 桁を示す値
	レンジ Ch1/Ch2, Ch1/Ch3	1E-25~1E8 (15 ~3 桁分解能)
	レンジ Ch2/Ch1, Ch3/Ch1	1E-26~1E7 (15 ~3 桁分解能)

	分解能	<resolution>は仮数 1.0 を使用し、10 の偶数乗の必要があります。
<source list>	初期設定	(@1,@2)
	範囲	(@1,@2) (@1,@3) (@2,@1) (@3,@1)

**:MEASure[:SCALar][:VOLTage]:MAXimum?[(@1)](@2)**

最大電圧を測定します。

<source list>	初期設定	(@1)
	範囲	(@1) (@2)

**:MEASure[:SCALar][:VOLTage]:MINimum?[(@1)](@2)**

最小電圧を測定します。

<source list>	初期設定	(@1)
	範囲	(@1) (@2)

**:MEASure[:SCALar][:VOLTage]:NWIDth? [<reference>],(@1)**

測定アーミングを「auto」に設定し、マイナスパルス幅を測定します。

<reference>はマイナスパルス幅が決定されるパルス上のポイントを指定します。このポイントは%または電圧で指定できます。初期設定の単位は%です。

<reference>パラメータはトリガ設定の構成に使用されます。これが初期設定の%に指定されていると、オートトリガが有効化されトリガ・レベルは指定値です。電圧に指定されていると、オートトリガが無効化され、トリガ・レベルは指定値です。

<reference>	初期設定	50 %
	%範囲	0~100 pct
	%分解能	10%
	電圧レンジ	-5.125V~5.125V、アッテネーション×1 -51.25V~51.25V、アッテネーション×10
	電圧分解能	0.0025V、アッテネーション×1 0.025V、アッテネーション×10
<source list>	初期設定	(@1)
	範囲	(@1)

**:MEASure[:SCALar][:VOLTage]:PWIDth? [<reference>],(@1)**

測定アーミングを「auto」に設定しプラスパルス幅を測定します。

<reference> プラスパルス幅が決定されるパルス上のポイントを指定します。このポイントは%または電圧で指定できます。初期設定の単位は%です。

<reference>パラメタはトリガ設定の構成に使用されます。これが初期設定の%に指定されていると、オートトリガが有効化されトリガ・レベルは指定値です。電圧に指定されていると、オートトリガが無効化され、トリガ・レベルは指定値です。

<reference>	初期設定	50 %
	%範囲	0 ~ 100 pct
	%分解能	10%
	電圧レンジ	-5.125V~5.125V、アッテネーション×1 -51.25V~51.25V、アッテネーション×10
	電圧分解能	0.0025V、アッテネーション×1 0.025V、アッテネーション×10
<source list>	初期設定	(@1)
	範囲	(@1)

**:MEASure[:SCALar][:VOLTage]:PERiod? [<expected\_value> [,<resolution>]]**

**[,(@1)](,@2)](,@3)]**

測定 アーミングを「digits」に設定し、周期を測定します。

<expected value>と<resolution>パラメタは分解能アーミング設定の桁数構成に使用されます。

チャンネル 1 とチャンネル 2 は同じトリガ設定を共有しており、測定チャンネルはオートトリガが+スロープで 50%に設定されています。

<expected value> Ch1, Ch2	初期設定	100 ns
	レンジ	2 ns~10 sec
	分解能	<expected value>は最適なアーミング構成を得るには入力周期の 10%以内であるべきです。
<expected value> Ch3	初期設定	<b>2 ns</b>
	レンジ	0.17 ns~2.67 ns
	分解能	<expected value>は最適なアーミング構成を得るには入力周期の 10%以内であるべきです。
<resolution> Ch1, Ch2	初期設定	設定された<expected value>の分解能 4 桁を示す値
	レンジ	1E-23~1E-2 sec (15 ~3 桁分解能)
	分解能	<resolution>は仮数 1.0 を使用し、10 の偶数乗であるべきです
<resolution> Ch3	初期設定	指定された<expected value>の分解能 4 桁を示す値
	レンジ	1E-24~1E-11 sec (15 ~3 桁分解能)
	分解能	<resolution>は仮数 1.0 を使用し、10 の偶数乗であるべきです

<source list>	初期設定	(@1)
	範囲	(@1)(@2)(@3)

**:MEASure[:SCALar][:VOLTage]:PHASe?[(@1),(@2)]**

測定アーミングを「auto」に設定し、フェーズを測定します。

チャンネル 1 とチャンネル 2 は同じトリガ設定を共有しており、測定チャンネルはオート・トリガが+スロープで 50%に設定されています。

<source list>	初期設定	(@1),(@2)
	範囲	(@1),(@2)

**:MEASure[:SCALar][:VOLTage]:PTPeak?[(@1)(@2)]**

ピーク間測定を行います。

<source list>	初期設定	(@1)
	範囲	(@1)(@2)

**:MEASure[:SCALar][:VOLTage]:CENTer?[(@1)(@2)]**

中心電圧を測定します。

<source list>	初期設定	(@1)
	範囲	(@1)(@2)

**:MEASure[:SCALar][:VOLTage]:TINTerval?[(@1),(@2)]**

時間間隔を測定します。

第 1 チャンネルが開始チャンネル、第 2 チャンネルが停止チャンネルです。

チャンネル 1 とチャンネル 2 は同じトリガ設定を共有しており、測定チャンネルはオート・トリガが+スロープで 50%に設定されています。

測定 アーミング は遅延無し「auto」に設定されています。入力経路は個別入力と接続されています。

<source list>	初期設定	(@1),(@2)
	範囲	(@1),(@2)

**:MEASure[:SCALar][:VOLTage]:TOTalize:TIMed? [<gate\_time>],(@1)]**

指定された<gate\_time>間のトータライズ測定を行います。

トリガ設定は測定チャンネルがプラス・スロープで autoトリガが無効になります。

<gate_time>	初期設定	0.1 sec
	レンジ	1E-3 ~1,000 sec
	分解能	1E-5 sec
<source list>	初期設定	(@1)
	範囲	(@1)

### :MEMory:NSTates?

利用可能な\*SAV/\*RCL 状態数をクエリします。

#### Return

- <NR1>フォーマットで ASCII バイトとして転送された数値データ、ここでは 21。
- 応答は\*SAV と\*RCL コマンドにパラメタとして送信可能な最大数より 1 大きいです。

### :READ[[:SCALar]:<function>]?

コマンドは:CONfigure コマンドと通常いっしょに使用され、:MEASure? コマンドと同様に測定します。これらの間に、設定仕様コマンドを追加することにより、ユーザ固有の設定を作成することができます。

このコマンドは、:READ?コマンドの一般的形式です。<function>と<parameter>をスキップすると、前のコマンド :CONfigure, :MEASure, :READ、または:FETCh で指定された機能を使用します。[:SCALar]:<function>が指定されると、現在の測定機能と一致するか、あるいは現在の測定機能から導き出せる場合に限り、指定された結果を返します。機能の応用についてはこのセクションにあるコマンド:FETCh?の説明を参照ください。

#### Return

- 結果は:FORMat[:DATA] ASCii|REAL の設定に従いフォーマットされます。
- ASCii フォーマットを使用する際、数値データは<NR3>フォーマットで ASCII バイトとして転送されます。有効桁数は測定分解能に依存して 1~5 の範囲になります。

#### 注

- このコマンドを測定進行中に送ると現在の測定を終了し、望みの 測定を開始する前に測定サイクルをアイドル状態にし、測定の完了を待ちます。ここから先のコマンド実行は現在の測定が完了するまで保留されます。しかし Device Clear を使用するか電源を入れてこの保留されたコマンドの実行をキャンセルすることができます。
- 自動的にアームされたトータライズ測定の進行中に:READ?を送るとコマンドはエラー-221 を発生させ、異常値 9.91E37 が返されます。:INItiate で 測定を開始、:ABORT で中止します。:FETCh? は終了した測定結果の取得に使うべきです。

## コマンド:SENSe

`[[:SENSe]DATA?["[:]SENSe[1]"]]`

:SENSe コマンドの現在の測定結果をクエリします(倍率やオフセットは適用されません)。

### Return

- 結果は:FORMat[:DATA] ASCii|REAL の設定に従いフォーマットされます。

### 注

- ASCii フォーマットを使用する際、数値データは<NR3>フォーマットで ASCII バイトとして転送されます。有効桁数は測定分解能に依存して 1~10 の範囲になります。
- 有効な測定結果が無い場合、異常値 9.91E37 が返され、エラー -230 が生成されません。
- 測定進行中にこのコマンドを実行すると、直前の測定結果が返されます。
- このコマンドが自動的にアームされたトータライズ測定の進行中に実行されると、異常値 9.91E37 が返され、エラー -221 が生成されます。:ABORt コマンドで測定を中止し、エラー発生を防止してください。

`[[:SENSe]EVENT2:FEED ["[:]INPut[1]][:]INPut2"]`

コモン/セパレートの状態を設定またはクエリします。

### Return

- 文字列「INP」または「INP2」が返されます。

### 注

- カウンターがオフに切り替えられた後、またはリモートインターフェイスのリセット後、初期値の「INP2」に設定されます。
- 時間間隔測定以外の全機能は、入力経路の設定が可能です。

`[[:SENSe]EVENT[1|2]:HYSTeresis:RELative <numeric value> [PCT]`

ヒステリシス・ウインドウのサイズを、可能なヒステリシスの%として、設定またはクエリします。例えば、0%は最小ヒステリシス設定、100%は最大です。100%すなわち最大と指定すると、もっともノイズの影響を受けにくくなり(最低感度)、0%すなわち最小と指定すると最もノイズの影響を受けやすくなります。

### Return

- 「100」、「50」または「0」 PCT(%)が返されます。

## 注

- \*RST コマンドが実行されると、ノイズ耐性は 50 PCT(%) に設定されます。

### **[[:SENSe]EVENT[1|2]:LEVel[:ABSolute] <numeric value> [V]**

ヒステリシス・ウインドウのセンターレベルを設定またはクエリします。範囲は X1 アッテネーションでは -5.125~5.125V、X10 アッテネーションでは -51.25~51.25V です。分解能は X1 アッテネーションで 0.0025V、X10 アッテネーションで 0.025V です。

## Return

- 有効数字 6 桁の <NR3>フォーマットで ASCII バイトとして転送される数値データ。

## 注

- このコマンドを送ると[:SENS]:EVEN[1 | 2]:LEV:AUTO を無効化します

### **[[:SENSe]EVENT[1|2]:LEVel[:ABSolute]:AUTO <Boolean>**

オート・トリガの状態を設定またはクエリします。

オート・トリガが有効化されていると、トリガ・レベルをコマンド[:SENS]:EVEN[1 | 2]:LEV:REL により指定された%に従って自動的に測定および計算します。また、:INIT または:INIT:COUNT ON コマンドが送られるたびに、トリガ・レベルを測定および計算します。

オート・トリガがオンのとき、測定ごとに測定信号がトリガしているのかを確認し、トリガが見つからない場合は、新たにトリガ・レベルを測定し計算します。

## Return

- シングル ASCII エンコード型バイト。0 の意味は OFF、1 は ON です。

## 注

- オフに切り替えられた後ま、またはリモートインターフェイスのリセット後、これは ON に設定されます。
- トリガ・レベルをコマンド[:SENS]:EVEN[1 | 2]:LEV[:ABS]で指定すると オート・トリガを無効にします。

### **[[:SENSe]EVENT[1|2]:LEVel:RELative <numeric value> [PCT]**

機器が自動的に、トリガする信号のピーク間レンジの%を設定またはクエリします。レンジは 0 ~100 PCT(%)および分解能は 10 PCT(%)です。

このコマンドを受信すると、トリガ・レベルをコマンド[:SENS]:EVEN[1 | 2]:LEV:REL で指定された%に従い、自動的に測定および計算します。

#### Return

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

#### 注

- 電源オフに切り替えられた後、またはリモートインターフェイスのリセット後、50 PCT(%)にリセットされます。
- このコマンドは[:SENS]:EVENT[1|2]:LEV[:ABS]:AUTO が ON のときしか適用されません。

#### [:SENSe]EVENT[1|2]:SLOPe POSitive|NEGative

周波数、周期、周波数レシオ、時間間隔、トータライズまたはフェーズを測定する際、入力信号のどちらのエッジが適切か設定またはクエリします。

プラス・スロープを指定すると、低電圧から高電圧へ増加する信号でヒステリシス上限値をイベントと定義し、一方、マイナス・スロープを指定すると、高電圧から低電圧へ下がる信号でヒステリシス下限値をイベントと定義します。

#### Return

- ASCII エンコード型バイトのシーケンス、POS か NEG。

#### 注

- カウンターが電源オフの後、またはリモートインターフェイスのリセット後、これは Positive に設定されます。
- ここで指定されるうロープは立ち上がり/立ち下がり時間、デューティサイクル、およびパルス幅測定で使用されたスロープとは無関係です。

#### [:SENSe]EVENT3:LEVe[:ABSolute]?

チャンネル 3 トリガーレベルをクエリします。

#### Return

- <NR3>フォーマットで ASCII バイトとして転送される有効桁数 6 のボルト数値データ。

#### 注

- 返答値は 0、単位はボルトです。

#### [:SENSe]EVENT3:SLOPe?

どのチャンネル 3 エッジを使ってイベントを定義するかをクエリします。

#### Return

- ASCII エンコード型バイトのシーケンス、POS。

周期、周波数、周波数レシオの測定用ゲートを設定またはクエリします。フロントパネルゲート設定とインターフェイスコマンドパラメタの詳細情報については 4.3 章の **FREQ** 表をご参照ください。

**[[:SENSe]:FREQuency:ARM[:STARt]:SLOPe POSitive | NEGative**

周期、周数、周波数レシオの測定の外部アーミングで使用される外部スタートアーム信号のスロープを設定またはクエリします。

**Return**

- ASCII エンコード型バイトのシーケンス。POS は正、NEG は負です。

**注**

- 電源オフに切り替えられた後、またはリモートインターフェイスのリセット後、これは正に設定されます。
- このコマンドは[:SENSe]:FREQuency:ARM[:STARt]:SOUR EXT が使用されるときのみ適用されません。

**[[:SENSe]:FREQuency:ARM[:STARt]:SOURce IMMEDIATE | EXTERNAL**

周期、周波数、周波数レシオの測定用起動アームを設定またはクエリします。

**Return**

- ASCII エンコード型バイトのシーケンス。IMM は Immediate(即時)、EXT は External(外部)を意味します。

**注**

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、アームは Immediate に設定されます。

**[[:SENSe]:FREQuency:ARM:STOP:DIGits <Numeric value>**

周期、周波数、周波数レシオの測定のアーミングで使用される桁数での分解能を設定またはクエリします。

**Return**

- <NR1>フォーマットで ASCII バイトとして転送された数値データ、ここで分解能範囲は 3~15 です。

#### 注

- カウンターがオフに切り替えられた後またはリモートインターフェイスのリセット後、これは 4 に設定されます。
- このコマンドは[:SENSe]:FREQuency:ARM:STOP:SOUR DIG が使用されるときしか適用されません。

#### **[:SENSe]:FREQuency:ARM:STOP:SLOPe POSitive | NEGative**

周期、周波数、**周波数レシオ**の測定の外部アーミングで使用される外部ストップ信号のスロープを設定またはクエリします。

#### Return

- ASCII エンコード型バイトのシーケンス。POS は正、NEG は負です。

#### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは Negative (負)に設定されます。
- このコマンドは[:SENSe]:FREQuency:ARM:STOP:SOUR EXT が使用されるときしか適用されません。

#### **[:SENSe]:FREQuency:ARM:STOP:SOURce IMMEDIATE | EXTERNAL | TIMER | DIGits**

周期、周波数、**周波数レシオ**の測定用停止アームを設定またはクエリします。

#### Return

- ASCII エンコード型バイトのシーケンス。IMM は Immediate (即時)、EXT は External (外部)、TIM は TIMER (タイマー)、DIG は DIGit (デジット)を意味します。

#### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、TIMER に設定されます。

#### **[:SENSe]:FREQuency:ARM:STOP:TIMER <numeric value> [S]**

周期、周波数、**周波数レシオ** 測定のアーミングで使用されるゲート時間を設定またはクエリします。測定レンジは 1E-3 ~1,000 秒、分解能は 0.01E-3 秒です。**周波数レシオ範囲は 1E-3~200 秒であることにご注意ください。**

#### Return

- <NR3>フォーマットで ASCII バイトとして転送される有効数字 6 桁の数値データ。

#### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、0.1 秒に設定されます。
- このコマンドは[:SENSe]:FREq:ARM:STOP:SOUR TIM が使用されるときしか適用されません。

#### **[:SENSe]:FREQuency:EXPEcted[1 | 2 | 3] <numeric value> [HZ]**

周波数、周期、**周波数レシオ**、**フェーズ**、**デューティサイクル** 測定の際の入力信号周波数の概数値を設定またはクエリします。チャンネル 1 と 2 の周波数レンジは 0.001 ~400E6 Hz、**チャンネル 3** で 375E6~6E9 Hz です。

#### Return

- <NR3>フォーマットで ASCII バイトとして転送される有効数字 15 桁の数値データ。

#### 注

- 実際の結果は、この概数値の 10%以内でなければなりません。
- この値を提供することで準備時間を短縮し、測定を加速し、より正確な結果を得るのに役立ちます。
- [:SENS]:FREq:EXP[1 | 2 | 3]が ON のときは、異常値 9.91E37 が返され、エラー -221 が生成されます。
- 保存/呼び出し操作は、この設定に使用できません。

#### **[:SENSe]:FREQuency:EXPEcted[1 | 2 | 3]:AUTO ON**

周期、周波数、**フェーズ**、**デューティサイクル**と**周波数レシオ**測定用の測定前ステップとして入力信号周波数を決定するための構成です。また、この測定前の状態もクエリします。

#### Return

- シングル ASCII エンコード型バイト。0 の意味は OFF、1 は ON です。

#### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、ON に設定されません。

- 自動モードを有効化すると、直前に指定されたおよその周波数値をクリアします。
- これを無効化する唯一の方法は[:SENSe]:FREQuency:EXPeCted[1 | 2 | 3] <numeric value> [HZ]コマンドで周波数の概数を設定することです。
- 保存/呼び出し操作は、この設定に使用できません。

#### **[:SENSe]:Function[:ON] <sensor function>**

検知される<sensor function>を設定またはクエリします。

<sensor function>は次のものがあります:

“[:][XNOnE:]DCYCLe[1]” or “[:][XNOnE:]PDUtYcYclE[1]”

“[:][XNOnE:]FALL:TiMe[1]” or “[:][XNOnE:]FTiMe[1]”

“[:][XNOnE:]RISE:TiMe[1]” or “[:][XNOnE:]RTiMe[1]”

“[:][XNOnE:]FREQuency[1 | 2 | 3]”

“[:][XNOnE:]FREQuency:RATio[1,2 | 2,1 | 1,3 | 3,1]”

“[:][XNOnE:]PERiod[1 | 2 | 3]”

“[:][XNOnE:]PHASe[1,2]”

“[:][XNOnE:]NWiDth[1]”

“[:][XNOnE:]PWIDth[1]”

“[:][XNOnE:]TPERiod[1,2]”

“[:][XNOnE:]TOTalize[1]”

“[:][XNOnE:]VOLTage:PTPeak[1 | 2]”

“[:][XNOnE:]VOLTage:MAXimum[1 | 2]”

“[:][XNOnE:]VOLTage:MINimum[1 | 2]”

“[:][XNOnE:]VOLTage:CENTer[1 | 2]”

#### **Return**

- “<function> <channel> [,<channel>]” 文字列が返されます。

#### **注**

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、初期設定の「FREQuency 1」に設定されます。
- 指定チャンネルが初期設定チャンネルの場合、返される文字列はそれをスキップし、指定チャンネルが初期設定チャンネルではない場合のみ表示されます。例えば、sensor function がチャンネル 1 で周波数の場合、返される文字列は「FREQu」です。センサ機能がチャンネル 2 で周波数の場合、文字列「FREQu 2」が返されます。
- オプションのチャンネル仕様がスキップされるときは、自動的に初期設定チャンネルを使用するよう選択します。これは、周波数、期間、立ち上がり/立ち下がり時間、デューティサイクル、パルス幅、トータライズ、Volt Peak 測定についてはチャンネル 1、時間間隔、周波数レシオ、フェー

ズの測定についてはチャンネル 2 に対してチャンネル 1 となります。

- センサ機能がトータライズ、電圧最大、電圧最小または Volt Peak の場合、[:INIT]:IMM で単一測定を開始します。
- センサ機能が立ち上がり時間に設定されているとき、チャンネル 1 のトリガ・レベルは 10% に設定され、チャンネル 2 のトリガ・レベルは 90% に設定されます。センサ機能が立ち下がり時間に設定されているとき、チャンネル 1 のトリガ・レベルは 90% に設定され、チャンネル 2 のトリガ・レベルは 10% に設定されます。オート・トリガは両方のケースで有効です。
- センサ機能がパルス幅またはデューティサイクルに設定されているときは、チャンネル 1 とチャンネル 2 でのトリガ・レベルは 50% に設定されます。オート・トリガは有効です。
- センサ機能が[:][TPERiod[1,2]]に設定されているとき、入力経路は二つに分かれて([:SENSe]:EVEN2:FEED「INP2」)に設定されます。コモンモードの時間間隔では[:SENS]:EVEN2:FEED「INP1」を選びます。
- このコマンドは :FETCh?, READ?, または、CONFigure?に影響しません。

フェーズ測定用のゲートを設定またはクエリします。フロントパネルゲート設定とインターフェイスコマンドパラメタについて詳細は 4.3 章のフェーズ表をご参照ください。

#### **[:SENSe]:PHASe:ARM[:STARt]:SLOPe POSitive | NEGative**

フェーズの測定の外部アーマングで使用される外部スタートアーム信号のスロープを設定またはクエリします。

#### **Return**

- ASCII エンコード型バイトのシーケンス。POS は正、NEG は負です。

#### **注**

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、正に設定されます。
- このコマンドはコマンド[:SENS]:PHAS:ARM[:STARt]:SOUR EXT が使用されるときしか適用されません。

#### **[:SENSe]:PHASe:ARM[:STARt]:SOURce IMMEDIATE | EXTERNAL**

フェーズの測定用起動アームを設定またはクエリします。

#### **Return**

- ASCII エンコード型バイトのシーケンス。IMM は Immediate(即時)、EXT は External(外部)を意味します。

## 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは Immediate に設定されます。

### **[[:SENSe]:ROSCillator:EXTErnal:FREQUency?**

タイムベース測定において、入力周波数がなければカウンターには(Fno)が表示され、入力周波数が無効な値であればカウンターに(Fxx)、PC には「Not a number」が表示されます。一方、コマンド ROSCillator:SOURce は、内部条件では「Not a number」および外部条件では「1.00000E+007」を返します。

### **Return**

- <NR3>フォーマットで ASCII バイトとして転送される有効数字 6 桁の数値データ。

## 注

- 現在の基準タイムベースが内部、または現在の基準タイムベースが外部であるが周波数は未知の場合、異常値 9.91E37 が返されます。

### **[[:SENSe]:ROSCillator:SOURce INTernal|EXTErnal**

現在の基準タイムベースを設定またはクエリします。

### **Return**

- ASCII エンコード型バイトのシーケンス。INT はタイムベースが内部参照であることを意味します。EXT は外部を意味します。

## 注

- 基準タイムベースを指定すると、自動的に[:SENS]:ROSC:SOUR:AUTO (:[:SENS]:ROSC:SOUR:AUTO OFF)が無効化されます。
- [:SENS]:ROSC:SOUR:AUTO ON の場合、このクエリはどのタイムベース基準が使用されるかを出力します。
- 保存/呼び出し操作は、この設定に使用できません。

### **[[:SENSe]:RosCilator:SOURce:AUTO <Boolean>**

自動指定の基準タイムベースの状態を設定またはクエリします。これがオンの場合、有効な信号がリアパネルの Ref In コネクタに入力されている場合、自動的に外部基準信号を基準タイムベースとして選択します。無効な信号の場合は、内部タイムベースが使用されます。

### **Return**

- シングル ASCII エンコード型バイト。0 の意味は OFF、1 は ON です。

## 注

- 基準オシレーターを[:SENS]:ROSC:SOUR INT | EXT で指定すると、この自動指定は無効化されます。
- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、初期設定である「ON」に設定が戻されます。

立ち上がり時間、立下り時間、デューティーサイクルおよびパルス幅測定用ゲート時間を設定またはクエリします。フロントパネルゲート設定とインターフェイスコマンドパラメタについては 4.3 章立ち上がり/立下りを参照してください。

### **[:SENSe]:TPERiod:ARM:ESTARt:LAYer2:SOURce IMMEDIATE | EXTERNAL**

立ち上がり時間、立下り時間、デューティーサイクルおよびパルス幅機能の測定用起動アームを設定またはクエリします。

クエリ応答:ASCIIエンコード型バイトのシーケンス:IMM または EXT

Comments:\*RST:IMMEDIATE

Related:Gate & ExtArm

### **[:SENSe]:TPERiod:ARM:ESTARt:LAYer2:SLOPe POSitive | NEGative**

立ち上がり時間、立下り時間、デューティーサイクルおよびパルスの測定の外部アーミングで使用される外部スタートアーム信号のスロープを設定またはクエリします。

クエリ応答:ASCIIエンコード型バイトのシーケンス:POS または NEG

Comments:● \*RST:POSitive

- [:SENS]:TINT:ARM:ESTARt:LAY2:SOUR EXTが  
選択されているときのみ有効です。

Related:Gate & ExtArm

時間間隔測定用ゲート時間を設定またはクエリします。フロントパネルゲート設定とインターフェイスコマンドパラメタについては 5.4 章の TINT 表をご参照ください。

### **[:SENSe]:TPERiod:ARM:ESTARt:LAYer2:SLOPe POSitive | NEGative**

時間間隔測定の外部アーミングで使用される外部スタートアーム信号のスロープを設定またはクエリします。

## Return

- ASCII エンコード型バイトのシーケンス。POS は正、NEG は負です。

#### 注

電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは正に設定されます。

- このコマンドは[:SENSe]:TINT:ARM:ESTARt:LAY2:SOUR EXT が使用されるときしか適用されません。

#### **[:SENSe]:TPERiod:ARM:ESTARt:LAYer2:SOURce IMMEDIATE | EXTERNAL**

時間間隔測定用起動アームを設定またはクエリします。

#### Return

- ASCII エンコード型バイトのシーケンス。IMM は Immediate(即時)、EXT は External(外部)を意味します。

#### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは Immediate(即時)に設定されます。

#### **[:SENSe]:TPERiod:ARM:ESTARt[:LAYer[1]]:ECOunt <numeric value>**

時間間隔測定用遅延起動アームに使用されるイベント数を設定またはクエリします。レンジは 1 から 99,999,999 までです。分解能は 1 です。

#### Return

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

#### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは 1 に設定されます。
- このコマンドは[:SENSe]:TINT:ARM:ESTARt[LAY[1]]:SOUR INTERNAL1 が使用されるときしか適用されません。
- このコマンドは、立ち上がり/立ち下がり時間、デューティーサイクルまたはパルス幅測定用には適用されません。

#### **[:SENSe]:TPERiod:ARM:ESTARt[:LAYer[1]]:SOURce IMMEDIATE | TIMER | INTERNAL[1]**

時間間隔測定用起動アーム遅延を設定またはクエリします。

### Return

- ASCII エンコード型バイトのシーケンス。IMM は Immediate(即時)、TIM は Timer(タイマー)、INT は Internal(内部)を意味します。

### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、Immediate(即時)に設定されます。
- このコマンドは立ち上がり/立ち下がり時間、パルス幅またはデューティサイクル測定には適用されません。

### **[[:SENSe]:TPERiod:ARM:ESTART[:Layer[1]]:TIMer <numeric value> [S]**

時間間隔測定用スタートアームにおける遅延時間を設定またはクエリします。

### Return

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは 100E-9 秒に設定されます。
- このコマンドは[:SENSe]:TINT:ARM:ESTART[:LAYer[1]]:SOUR TIM が使用されるときしか適用されません。
- このコマンドは 立ち上がり/立ち下がり時間、パルス幅またはデューティサイクル測定には適用されません。

### **[[:SENSe]:TPERiod:ARM:ESTOP:LAYer2:SLOPe POSitive | NEGative**

時間間隔測定の外部アミングで使用される外部停止アーム信号のスロープを設定またはクエリします。

### Return

- ASCII エンコード型バイトのシーケンス。POS は立上りエッジ、NEG は立下りエッジを意味します。

### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは正に設定されます。
- このコマンドは[:SENSe]:TINT:ARM:ESTOP:LAY2:SOUR EXT が使用されるときしか適用されません。

### **[[:SENSe]:TPERiod:ARM:ESTOP:LAYer2:SLOPe IMMEDIATE | EXTERNAL**

時間間隔測定用アームの停止ポイントを設定またはクエリします。

#### **Return**

- ASCII エンコード型バイトのシーケンス。IMM は Immediate(即時)、EXT は External(外部)を意味します。

#### **注**

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは Immediate に設定されます。

### **[[:SENSe]:TPERiod:ARM:ESTOP[:LAYer[1]]:ECOunt <numeric value>**

時間間隔測定用停止アーム遅延に使用されるイベント数を設定またはクエリします。レンジは 1 から 99,999,999 までです。分解能は 1 です。

#### **Return**

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

#### **注**

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは 1 に戻されます。
- このコマンドは[:SENSe]:TINT:ARM:ESTOP[:LAYer[1]]:SOUR Internal2 が使用されるときしか適用されません。

### **[[:SENSe]:TPERiod:ARM:ESTOP[:LAYer[1]]:SOURce IMMEDIATE | TIMER | INTERNAL2**

時間間隔測定用アームの停止ポイントを設定またはクエリします。

#### **Return**

- ASCII エンコード型バイトのシーケンス。IMM は Immediate(即時)、INT は Internal2、(内部) TIM は Timer(タイマー)を意味します。

#### **注**

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは Immediate (即時)に設定されます。
- このコマンドは立ち上がり/立ち下がり時間、パルス幅またはデューティサイクル 測定には適用されません。

**[[:SENSe]:TPERiod:ARM:ESTop[:Layer[1]]:TIMer <numeric value> [S]**

時間間隔測定用停止アームで 사용되는遅延時間を設定またはクエリします。レンジは 100E-9~10.000 秒、分解能は 100E-9 秒です。

**Return**

- <NR1>フォーマットで ASCII バイトとして転送された数値データ

**注**

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは 100E-9 秒に設定されます。
- このコマンドは[:SENSe]:TINT:ARM:ESTOP[:LAY[1]]:SOUR TIM が使用されるときしか適用されません。
- このコマンドは立ち上がり/立ち下がり時間、パルス幅またはデューティーサイクル 測定には適用されません。

トータライズ測定用ゲートを設定またはクエリします。フロントパネルのゲート設定とインターフェイスコマンドパラメタの詳細情報については 5.4 章の TOT 表をご参照ください。

**[[:SENSe]:TOTalize:ARM[:START]:SLOPe POSitive | NEGative**

トータライズ測定の外部アーミングで 사용되는外部スタートアーム信号のスロープを設定またはクエリします。

**Return**

- ASCII エンコード型バイトのシーケンス。POS は正、NEG は負です。

**注**

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは正に設定されます。
- このコマンドは[:SENSe]:TOT:ARM[:STAR]:SOUR EXT が使用されるときしか適用されません。

**[[:SENSe]:TOTalize:ARM[:STARt]:SOURce IMMEDIATE | EXTERNAL**

トータライズ測定用アーム開始点を設定またはクエリします。

**Return**

- ASCII エンコード型バイトのシーケンス。IMM は Immediate(即時)、EXT は External(外部)を

意味します。

#### 注

- 電源がオフに切り替えられた後またはリモートインターフェイスのリセット後、これは Immediate に設定されます。
- 開始点と停止点ともに immediate に設定されているとき、コマンド:INIT[:IMM]でトータライズ測定を開始、コマンド:ABORtで中止することができます。「auto」トータライズ測定が終了するまでは、有効なトータライズ結果はありません。

#### **[[:SENSe]:TOTalize:ARM:STOP: SLOPe POSitive | NEGative**

トータライズ測定の外部アーミングで使用される外部停止アーム信号のスロープを設定またはクエリします。

#### Return

- ASCII エンコード型バイトのシーケンス。POS は正、NEG は負です。

#### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは Negative (負)に設定されます。
- このコマンドは[:SENSe]:TOT:ARM:STOP: SOUR EXT が使用されるときしか適用されません。

#### **[[:SENSe]:TOTalize:ARM:STOP: SOURce IMMEDIATE | TIMer | EXTernal**

トータライズ測定用アームの停止ポイントを設定またはクエリします。

#### Return

- ASCII エンコード型バイトのシーケンス。IMM は Immediate(即時)、TIM は Timer(タイマー)、EXT は External(外部)を意味します。

#### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、Timer に設定されます。
- 開始点と停止点ともに immediate に設定されているとき、コマンド:INIT[:IMM]でトータライズ測定を開始、コマンド:ABORtで中止することができます。「auto」トータライズ測定が終了するまでは、有効なトータライズ結果はありません。

#### **[[:SENSe]:TOTalize:ARM:STOP:TIMer <numeric value> [S]**

アーミングトータライズ測定での gate\_time を設定またはクエリします。レンジは 1E-3 ~1,000 秒、分解

能は 0.01E-3 秒です。

#### Return

- <NR3>フォーマットで ASCII バイトとして転送される有効桁 6 桁の数値データ。

#### 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスのリセット後、これは 100E-3 秒に設定されます。
- このコマンドは[:SENSe]:TOT:ARM:STOP:SOUR TIM が使用されるときしか適用されません。

#### [:SENSe]:VOLTage:MODE FAST | SLOW

Volt Peak測定モードで、「FAST」または「SLOW」を設定またはクエリします。入力ソースが > 10 KHzでは「FAST」、入力ソースが > 100 Hzでは「SLOW」が推奨されます。

#### Return

- ASCII エンコード型バイトのシーケンス、FAST か SLOW。

#### コマンド:SYSTem

これらのコマンドはカウンターの性能に直接影響しない機能を対象にしています。

#### :SYSTem:COMMunicate:SERial:SOURce USB

ハードウェアペーススキームを設定またはクエリします。LIMIT PASS/FALL は、USB に設定されています。出力ポートは USB 経由となります。

#### Return

- USB メッセージが返されます。

#### 注

- コマンド:SYSTem:COMMunicate:SERial:SOURce USB が実行されるので LAN または GPIB 除き USB 経由でのリモートコントロールは不可です。

#### :SYSTem:COMMunicate:SERial:CONTrol:DTR OFF | LIMit

ハードウェアペーススキームを設定またはクエリします。

パラメータの LIMit は、コマンド:SYSTem:COMMunicate:SERial:SOURce USB で設定できます。コマンドは OFF に設定されているので、リミット PASS/FALL の Output 機能を使用することはできません。

#### Return

- ASCII エンコードバイトのシーケンス:OFF か LIM。

## 注

- この値は不揮発性メモリに保存され、電源がオフに切り替えられた後、またはリモートインターフェイスがリセットされた後でも変更されません。
- 起動および停止のスレッシュホールドは、ユーザーによる構成はできません。

## :SYSTem:ERRor?

エラーキューから最初のエラーをクエリし、エラーキューから削除します。6章のエラー番号とメッセージ参照してください。

## Return

- エラー番号とエラーメッセージが返されます。エラー番号は-32767～32767の間の整数です。エラーキューにエラーがないとき、「0」が返されます。

## 注

- 電源がオフに切り替えられた後か、キューにある最後のエラーが読まれた後、エラーキューは空にされます。
- エラーキューオーバーフローの原因となる30個を超えるエラーがあるとき、最後のエラーはエラー番号-350「キューオーバーフロー」に置き換えられます。最新エラーは破棄されます。
- このコマンドはエラーキューから表示されたエラーを削除します。
- \*RST コマンドはエラーキューを変更しません。

## :SYSTem:KEY <numeric value>

このコマンドはフロントパネルキーを押した状態をシミュレートします。パラメタ <numeric value> はキーコード番号です。次の表はキーとそれらの対応番号コードを示しています。

このコマンドはキー・キューにエントリを設定します。これはフロントパネルで実際にキーを押したのと同じ効果を持ちます。キー・キューの最大容量は500です。

Key Code	Key	Key Code	Key
1	Freq & Ratio	20	F5
2	Time & Period	21	F6
3	Volt Peak	22	Number_0
4	Other Meas	23	Number_1
5	Utility	24	Number_2
6	Run	25	Number_3
7	Gate & ExtArm	26	Number_4
8	Limit Modes	27	Number_5
9	Scale & Offset	28	Number_6
10	Stats	29	Number_7
11	Save & Recall	30	Number_8
12	Stop/Single	31	Number_9
13	CH1	32	Number_.
14	CH2	33	Number_+/-
15	CH3(not support)	34	↑
16	F1	35	←
17	F2	36	→
18	F3	37	↓
19	F4	38	Preset

#### Return

- <NR1>フォーマットで ASCII バイトとして転送された数値データ。戻り値はフロントパネルで最後に押されたキーを表しています。このキー・キューが空のときは、「-1」が返されます。

#### 注

- 電源がオフに切り替えられた後か、\*RST コマンド後にキー・キューはクリアされます。
- 保存/呼び出し機能は、この設定に適用されません。
- キー コマンドは連続使用できますが、他のキー コマンドを処理するか、キー キューに入るという点においてのみです。キー コマンドによって実行される操作は、次の非キー コマンドの処理前に完了するという保証はありません。キー コマンドとキー以外のコマンドを混在させる場合は、この点に注意してください。

#### :SYSTem:KEY:LOG?

キー・キューにある全エントリをクエリします。

#### Return

- <NR1>フォーマットで ASCII バイトとして転送された数値データ。キューが空のとき「-1」が返されます。各 Key Code 番号はカンマで区切られ、エントリは後入れ先出しの順です。
- キーコードの最大番号は 500。各キーコードのレンジは 1~10。

## 注

- 電源がオフに切り替えられた後か、\*RST コマンド後にキー・キューは空にされます。
- 保存/再呼び出し機能は、この設定に適用されません。

## :SYSTem:VERsion?

カウンターが適合すべき SCPI バージョン番号をクエリします。

## Return

- <NR2>フォーマットで ASCII バイトとして転送された数値データ
- 応答形式は YYYY.V であり、ここに YYYY は年、V は当年に認可されたバージョンを示します。

## :SYSTem:COMMunicate:RLState LOCAL | REMote | RWLock

機器のリモート/ローカル状態を設定またはクエリします。

LOCAL:全パネルキーを解放。

REMOte://Oモードをリモートに設定、Preset/Local以外の全パネルキーをロックします。

RWLock:全パネルキーをロックします。(リモートインターフェイスを介してのみ機器制御が可能です)。

## SYSTem:BEEPer:STATe OFF | ON

ビープ有効化モードを設定またはクエリします。

エラーがフロントパネルからまたはリモートインターフェイスから生成されたときに限ります。

## :SYSTem:IDNStr <QUOTED STRING>

I/O 互換モード用 IDN 文字列を設定します。I/O モードが互換のとき、設定文字列は\*idn?でクエリされます。

## :SYSTem:UTILity:IO:MODE NATive | COMPatibility

IDN 文字列を設定およびクエリするために、NATiveは初期設定、およびCOMPatibilityはユーザー定義で選ぶことができます。

※ 注:COMP モードへ切り替える場合、不正な文字列を設定すると\*IDN クエリから異常が返され、リモートインターフェイスを介する通信を切断します。

## :SYSTem:IDN 'string'

デバイス ID、例えば、PICOTEST CORP.,U6200A,TW00000001,0.00-000-00-00-01 を互換装置に設定し書き込みます。概して、このコマンドは製造者による固有デバイスと互換であるために使用されます。

#### Return

- ASCIIエンコード型バイトのシーケンス:NAT かCOMP。

#### 注

- コマンド\*RST が実行されるにともない、NAT が設定されます。

### コマンド:TRACe

これらのコマンドは倍率およびオフセット値を指定しクエリするために使用されます。

#### :TRACe:CATalog?

使用する定数の名前をクエリします。

#### Return

- 文字列"SCALE"および"OFFSET"が返されます。

#### :TRACe[:DATA] OFFSET, <numeric value> [HZ|S|DEG]

または

#### :TRACe[:DATA] OFFSET, <arbitrary block>

オフセット値を設定します。オフセット値の範囲は  $-9.9999990000E+12$  ~  $-1.0000000000E-13$ 、 $0.0000000000$  および  $+1.0000000000E-13$  ~  $+9.9999990000E+12$ 。分解能は 11 桁。

#### 注

- オフセット値は、電源がオフに切り替えられた後、またはリモートインターフェイスがリセットされた後、0にリセットされます。
- 新しい オフセット値を指定するとリミットカウント数 (:CALC2:LIM:FCO, :CALC2:LIM:PCO)をクリアします。

#### :TRACe[:DATA]? OFFSET

オフセット値をクエリします。

#### Return

- 応答フォーマットは:FORMat[:DATA] ASCii|REAL コマンドによる指定に従います。
- ASCii フォーマットを使用すると、数値データは <NR3>フォーマットで有効桁数 11 桁の ASCII バイトとして送られます。

#### 注

- 表示画面に全ての有効桁が表示できないとき、表示される値は四捨五入されるので実測値と異

なります。しかし 11 桁表示の時に ENTER 下のソフトキーを押せば表示値を実測値に変更します。

**:TRACe[:DATA] SCALE, <numeric value>**

または

**:TRACe[:DATA] SCALE, <arbitrary block>**

倍率値を設定します。倍率値のレンジは  $-9.999999E+12 \sim -1.000000E-13$ 、 $0.0000000000$  および  $+1.000000E-13 \sim +9.999999E+12$ 。分解能は 7 桁。

**注**

- オフセット値は、電源がオフに切り替えられた後、またはリモートインターフェイスがリセットされた後、 $1.000000$  にリセットされます。
- 新しい オフセット値を指定するとリミットカウント数 (:CALC2:LIM:FCO, :CALC2:LIM:PCO)をクリアします。

**:TRACe[:DATA]? SCALE**

倍率の値をクエリします。

**Return**

- 応答フォーマットは:FORMat[:DATA] ASCii|REAL コマンドでの指定に従います。
- ASCii フォーマットを使用すると、数値データは有効桁数 11 桁を持つ<NR3>フォーマットの ASCII バイトとして転送されます。

**注**

- 表示画面が全ての有効桁数を表示できない場合、表示される値は丸められ、実測値とは異なりますが、表示中に ENTER の下のソフトキーを押せば、現在の表示値を実測値に変更します。

**:TRIGger:COUNt:AUTO <Boolean>**

:INITiate[:IMMEDIATE] コマンドが送られるときに取得される測定数の制御を設定またはクエリします。

これが OFF の場合、:INIT[:IMM]は 1 つの測定を初期化します。:TRIG:COUN:AUTO が ON で [:SENS]:FUNC[:ON]が周波数、周波数レシオ、Volt Peak、周期、パルス幅、デューティサイクル測定の 1 つであるときでも統計機能が無効(:CALC:AVER[:STAT] が OFF)であると、:INIT[:IMM]は 1 つの測定のみを開始しますが、統計機能が有効であれば(:CALC:AVER[:STAT] が ON)、:INIT[:IMM]は (:CALC3:AVER:COUN により)指定された測定数を開始します。

**Return**

- シングル ASCII エンコード型バイト。0 の意味は OFF、1 は ON です。

## 注

- 電源がオフに切り替えられた後、またはリモートインターフェイスがリセットされた後、これは OFF 設定へ戻ります。

## D.3 IEEE-488 共通コマンド

U6200A/220A で使用される SCPI コマンドは、SCPI 標準バージョン 1992.0 に準拠しています。一般的なコマンドは次のとおりです。

注: U6220A は、緑色の一部の IEEE-488 コマンドをサポートしていません。

以下は、カタカナ表記になる単語が多いため、原文のままです。

### \*CLS

Clears all event registers summarized in the status byte, including Standard Event Status Register, Operation Event Status Register and Questionable Data Event Status Register, and the error queue. This command also place the counter in "Operation Complete Idle State" and "Operation Complete Query Idle State" (IEEE488.2) and disables any prior \*OPC command.

If \*CLS follows immediately a program message terminator, the output queue and the MAV bit are cleared because "any" new program message after a program message terminator clears the output queue.

### Notes

- This command does not clear saved settings or any data memories.
- This command will clear any displayed error messages.

### \*DDT<arbitrary block>

This is a Define Device Trigger command. It sets or queries the command that the device will execute when it receives the IEEE 488.1 Group Execute Trigger (GET) interface message or a \*TRG common command.

The counter takes only the commands :INITiate[:IMMediate], :READ? and :FETCh? or otherwise error -224 will be generated. When the parameter is a zero-length <arbitrary block>, the counter does not do anything when it receives a GET or a \*TRG command.

### Return

- #14INIT, #15FETC?, #15READ? Or #0 terminated with a new line and EOI.

#### Notes

- It will be set back to #14INIT after the counter has been turned off or after a remoter interface reset.
- When defining the device trigger to :FETCh? Or :READ?, note that these definitions do not allow the specification of a particular function. This lack of a function specification results in each \*DDT using the function specified and used by the last command :CONFigure, :FETCh, :READ, or :MEASure, if possible.

#### \*ESE<NRf>

Sets the Standard Event Status Enable Register. The parameter's range is from 0 to 255 and the resolution is 1. The parameter represents the bit values of the Standard Event Status Enable Register when it is rounded to an integer value and expressed in base 2(binary). The value of unused bits is zero when queried and ignored when set.

#### Notes

- The Standard Event Status Enable Register is cleared and set to "0" after the counter has been turned off.
- This value is unaffected by save/recall and \*RST.

#### \*ESE?

Queries the Standard Event Status Enable Register.

#### Return

- Numeric data transferred as ASCII bytes in <NR1> format.

#### Notes

- The Standard Event Status Enable Register is cleared and set to "0" after the counter has been turned off.
- This value is unaffected by save/recall and \*RST.

#### \*ESR?

Queries the Standard Event Status Register.

This event register captures changes in conditions, by having each event bit correspond to a specific condition in the instrument.

An event becomes TRUE when the associated condition makes the defined transition. The event bits cannot be cleared until they are read, even if they do not reflect the current status of a related condition.

#### Return

- Numeric data transferred as ASCII bytes in <NR1> format.
- The query response is an integer formed by the binary-weighting of the bits. The value of unused bit is zero.

#### Notes

- This register is cleared after the counter has been turned off or after \*CLS or \*ESR?. Note that the instrument's power-on sequence initially clears the register, but then records any subsequent events during the power-on sequence including setting the Power on bit.

#### \*IDN?

Queries the counter identification.

#### \*OPC

Enables the OPC bit (bit 0) in the Standard Event Status Register to be set upon the transition if the measurement cycle from measuring to idle.

#### Note

- This event command is "disabled" by \*CLS, \*RST, Device Clear, after the counter has been turned off or upon the transition of the measurement cycle from measuring to idle.

#### \*OPC?

A response is generated when the counter is placed in an idle state from a measuring state. This allows synchronization between a controller and the instrument using the MAV bit in the Status Byte Register or a read of the Output Queue.

Since this query will not respond until the measurement cycle transition from measuring to idle, the only way to cancel the query "holdoff" is by Device Clear or power-on.

#### Return

- A single ASCII-encoded byte, 1.

#### Note

- This query does not affect the OPC bit in the Standard Event Status Register.

#### \*OPT?

Queries the instrument to identify any installed options.

Rear panel input module (CH1/CH2): U6200-opt04 (for U6200A only)  
Rear pane input module (CH1/CH2/CH3): U6200-opt05 (for U6200A only). GPIB  
Card: M3500-opt05

\*RCL<NRf>

Restores the state of the counter from a copy stored in local non-volatile memory. Before the recall takes place, the current state of the instrument is automatically saved to register 0. The parameter's range is from 0 to 20 and the resolution is 1.

Note

□ This command does not affect the following commands or statuses:

\*ESE

\*OPC

\*OPC?

\*SRE

\*WAI

:CALibration:DATA

:CALibration:SECurity:CODE

:CALibration:SECurity:STATe

:CONFigure?

:DISPlay:ENABle

136

:DISPlay:MENU[:STATe]

:DISPlay[:WINDow]:TEXT:RADix

[:SENSe]:FREQuency:EXPeCted[1|2|3]

[:SENSe]:FREQuency:EXPeCted[1|2|3]:AUTO

[:SENSe]:ROSCillator:SOURCe

[:SENSe]:ROSCillator:SOURCe:AUTO

:SYSTem:COMMunicate:SERial:CONTRol:DTR

:SYSTem:ERRor?

:SYSTem:KEY?

:SYSTem:KEY:LOG?

GP-IB address

\*RST

Resets the universal counter to its power-on configuration, disables macros, empties the Key queue and places the counter in "Operation Complete Idle State" and "Operation Complete Query Idle State".

This command does not change any calibration data, the Service Register Enable, the Standard Event Status Enable, the Output Queue, the IEEE488.1 address or the state of the IEEE488,1 interface.

**\*SAV<NRf>**

Saves the current state of the counter in local non-volatile memory.

The current state of the counter is saved in register 0 when \*RCL or front-panel recall is executed.

The parameter's range is from 1 to 20 and the resolution is 1.

**\*SRE<NRf>**

Sets the Service Request Enable Register.

The parameter value, when rounded to an integer value and expressed in base 2(binary), represents the bit values of the Service Request Enable Register.

This register is used to enable a single or inclusive OR group of Status Byte Register events to generate an SRQ. The parameter's range is to 255 and the resolution is 1.

**\*SRE?**

Queries the Service Request Enable Register.

Return

- Numeric data transferred as ASCII bytes in <NR1> format.
- The value of bit 6 and unused bits is zero when queries.

Notes

- This value is cleared and set back to zero after the counter has been turned off.
- This value is not affected by save/recall and \*RST command.

**\*STB?**

Queries the Status Byte Register. This query does not directly change the Status Byte Register (including the MSS/RQS bit) or anything related to the generation of SRQ. The response value when rounded to an integer value and expressed in base 2(binary), represents the bit values of the Status Byte Register.

Return

- Numeric data transferred as ASCII bytes in <NR1> format.
- Range is 0 to 255.

- The value of unused bits is zero when queried.
- The Master Summary Status, not the RQS message, is reported on bit 6. Master Summary Status indicates that the counter has at least one reason for requesting service. (The Master summary Status is not sent in response to a serial poll; the IEEE 488.1 RQS message is sent instead.) It is the inclusive OR of the bitwise combination (excluding bit6) of the Status Byte Register and the Service Request Enable Register,

#### \*TRG

This command is the device-specific analog of the IEEE 488.1 group Execute Trigger (GET) interface message, and has exactly the same effect.

This command will perform the action defined by the \*DDT command.

#### \*TST?

Starts an internal self-test and returns a response indicating if there is any error found in the self-test.

#### Return

- Numeric data transferred as ASCII bytes in <NR1> format.
- A response value of zero indicates a complete self-test without any errors and a response value of non-zero value indicates either a complete test with errors or an unfinished self-test.
- If the self-test fails, error -330 will be generated.

#### Note

- The self-test includes the followings:  
CPU, ROM, RAM, EEPROM, QSPI, FPGA, Front End, Measurement hardware and Interpolator hardware.

#### \*WAI

This command prevents the instrument from executing any further commands or queries until the measurement cycle transitions from measuring to idle. The only way to cancel this “holdoff” is by device memory clearance or power-on.

#### Note

- This command is not affected by \*RST and \*CLS commands.

## D.4 IEEE-488 コンプライアンス 情報

このセクションには、U6200A/220A に固有のコマンドのリストが含まれています。これらのコマンドは SCPI 形式を念頭に置いて設計されており、標準のすべての構文規則に従います。

※

注: U6220A は、緑色の IEEE-488 準拠の一部をサポートしていません。

:CALCulate[1]

:DATA?

:FEED “[:]SENSe[1]”

:IMMediate

:AUTO <Boolean>

:MATH

[:]EXPRession]

:CATalog?

[:]DEFine]?

:NAME | :SElect SCALE\_OFFSET

:STATe <Boolean>

:CALCulate2

:FEED “[:]CALCulate[1]”

:IMMediate

:AUTO <Boolean>

:LIMit

:STATe <Boolean>

:DISPlay GRAPH | NUMBer

:CLEar

:AUTO <Boolean>

[:]IMMediate]

:FAIL?

:PCOunt

[:]TOTal]?

:FCOunt

:LOWer?

:UPPer?

[:]TOTal]?

:LOWer

```

        [:DATA] <numeric_value> [HZ | S | DEG]
:UPPer
        [:DATA] <numeric_value> [HZ | S | DEG]

:CALCulate3
    :DATA?
    :FEED “[:]CALCulate[1]”
    :PATH?
    :AVERage
        [:STATe] <Boolean>
    :ALL?
    :CLEar
    :COUNT <numeric_value>
        :CURRent?
    :TYPE MAXimum | MINimum | SDEViation | SCALar or MEAN | DELTa|

ALLAN
    :LFILter
        :STATe <Boolean>
    :LOWer
        [:DATA] <numeric value> [HZ | S | DEG]
    :UPPer
        [:DATA] <numeric value> [HZ | S | DEG]

:CALibration
    :DATA <arbitrary block>
    :SECurity
        :STATe <Boolean>,<NRf>
        :CODE <Nrf>
        :RESet

:CONFigure[:SCALar]:<function>

:CONFigure?

:DISPlay
    :ENABle <Boolean>
    :MENU

```

```

[:STATe] OFF
[:WINDow]
:TEXT
    :FEED “[:]CALCulate2” | “[:]CALCulate3”
    :RADix COMMa | DPOint
:FETCh

:INITiate
    :AUTO <Boolean>
    :Continuous<Boolean>
    [:IMMEDIATE]

:INPut[1|2]
    :ATTenuation
    :COUPling AC | DC
    :LFILter
        [:LPASs]
            [:STATe] <Boolean>
            :FREQuency?
    :IMPedance <numeric_value> [OHM]

:INPut3
    :COUPling ?
    :IMPedance?

:MEASure[:SCALar]:<function>?

:READ[:SCALar]:<function>?

[:SENSe]
    :DATA?["[:]SENSe[1]"]
    :EVENt[1|2]
        :HYSTeresis
            :RELative <numeric value> [PCT]
        :LEVel
            [:ABSolute] <numeric value> [V]
            :AUTO <Boolean>

```

```

:RELative <numeric value> [PCT]
:SLOPe POSitive | NEGative
:EVENT2
:FEED "[:]INPut[1] | [:]INPut2"

:FREQuency
:ARM
[:START]
:SLOPe POSitive | NEGative
:SOURce IMMEDIATE | EXTERNAL
:STOP:DIGits<numeric value>
:SLOPe POSitive | NEGative
:SOURce IMMEDIATE | EXTERNAL | TIMER | DIGits
:TIMER <numeric value> [S]
:DIGits <numeric value>

:EXPEcted[1|2|3] <numeric value> [HZ]
:AUTO ON

:FUNCTion
[:ON] "[:][XNONe:]FREQuency[1|2|3]"
"[:][XNONe:]FREQuency:RATio [1,2|1,3|2,1|3,1]"
"[:][XNONe:]PERiod [1|2|3]"
"[:][XNONe:]RISE:TIME [1]"
"[:][XNONe:]RTIME [1]"
"[:][XNONe:]FALL:TIME [1]"
"[:][XNONe:]FTIME [1]"
"[:][XNONe:]PWIDth [1]"
"[:][XNONe:]NWIDth [1]"
"[:][XNONe:]TINTerval [1,2]"
"[:][XNONe:]VOLTage:PTPeak [1|2]"
"[:][XNONe:]VOLTage:MAXimum[1|2]"
"[:][XNONe:]VOLTage:MINimum[1|2]"
"[:][XNONe:]VOLTage:CENTer [1|2]"
"[:][XNONe:]PHASe [1,2]"
"[:][XNONe:]DCYCLE [1]"
"[:][XNONe:]PDUtycycle [1]"

```

”[:][XNONE:]TOTalize [1]”

:PHASe

:ARM

[:START]

:SLOPe POSitive | NEGative

:SOURce IMMEDIATE | EXTernal

:Rosclator

:SOURce INTernal | EXTernal

:AUTO <Boolean>

:EXTernal

:Frequency?

:TINTerval

:ARM

:ESTART

:LAYer2

:SLOPe POSitive | NEGative

:SOURce IMMEDIATE | EXTernal

[:LAYer[1]]

:ECOunt <numeric\_value>

:SOURce IMMEDIATE | TIMer | INTernal1

:TIMer <numeric value> [S]

:ESTOP

:LAYer2

:SLOPe POSitive | NEGative

:SOURce IMMEDIATE | EXTernal

[:LAYer[1]]

:ECOunt <numeric\_value>

:SOURce IMMEDIATE | TIMer | INTernal2

:TIMer <numeric\_value>

[:START]

:SLOPe POSitive | NEGative

:SOURce IMMEDIATE | EXTernal

:STOP

:SOURce IMMEDIATE | TIMer

:TImEr <numeric\_value>

:Totalize

:ARM

[:START]

:SLOPe POSitive | NEGative

:SOURce IMMEDIATE | EXTernal

:STOP

:SLOPe POSitive | NEGative

:SOURce IMMEDIATE | EXTernal | TImEr

:TImEr <numeric value> [S]

:SYSTem

:COMMunicate

:SERial

:SOURce USB

:CONTrol

:DTR OFF | LIMit

:ERRor?

:KEY <numeric\_value>

:LOG?

:VERSion?

:TRACe

:CATalog?

[:DATA] OFFSET, <numeric\_value> [HZ | S | DEG]

[:DATA] SCALE, <numeric\_value>

[:DATA]?OFFSET

[:DATA]?SCALE

:TRIGger

:COUNT

:AUTO <Boolean>

---

## E. アプリケーション・プログラムについて

このセクションは U6200A/220A のアプリケーション・サンプルごとに概説しています。

※ 注: U6220A は 緑で表記されている アプリケーション・プログラムをサポートしていません。

### CONfigure の測定への使用法

#### 例 1 Freq & Ratio 測定

```
// Freq_Ratio.cpp : Defines the entry point for the console application.
#include "stdafx.h"
#include "visa.h" //include visa32.lib first to the project.
#define MAX_COUNT 200
#include "stdio.h"

int main(int argc, char* argv[])
{
    ViStatus stat; /* For checking errors */
    ViSession defaultRM; /* Communication channels */
    ViSession sesn; /* Communication channel */
    ViChar rsrcName[VI_FIND_BUFLLEN]; /* Serial resource name */
    ViChar intfDesc[VI_FIND_BUFLLEN]; /* Port binding description */
    ViUInt32 ret; /* To hold number of resources */
    ViFindList flist; /* To hold list of resources */

    char readin[128]="";
    int tmo;

    printf("\n ##### Start C++ Example program. #####\n");
    printf(" We check the Counter on USB port and\n");
    printf(" identify the first connected device.\n\n");

    /* Begin by initializing the system */
    stat = viOpenDefaultRM(&defaultRM);

    if (stat < VI_SUCCESS) {
        /* Error Initializing VISA...exiting */
    }
}
```

```

        printf ("Error Initializing NI-VISA...");
        return 0;
    }
    stat = viFindRsrc (defaultRM, "USB[0-9]::0x164E::0x1838::?*INSTR",
&flist, &ret, rsrcName);

    stat = viOpen (defaultRM, rsrcName, VI_NULL, VI_NULL, &sesn);
    if (stat < VI_SUCCESS)
    {
        printf ("Could not open %s, status = 0x%08IX\n",rsrcName, stat);
        return 0;
    }
    else
    {
        stat = viGetAttribute (sesn, VI_ATTR_INTF_INST_NAME, intfDesc);
        printf ("Resource %s, Description %s\n", rsrcName, intfDesc);
    }
    stat = viFindNext (flist, rsrcName);

//Setting Process Visa Timeout
stat = viGetAttribute(sesn, VI_ATTR_TMO_VALUE, &tmo);
stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, 30000);

//send reset command '*RST' -- reset U6200A
stat = viWrite(sesn, (unsigned char *)("*RST"), 4, &ret);

//send Clear command '*CLS'-- Clear U6200A status register
stat = viWrite(sesn, (unsigned char *)("*CLS"), 4, &ret);
//send reset command 'INIT:CONT ON' -- Initial U6200A
stat = viWrite(sesn, (unsigned char *)("INIT:CONT ON"), 12, &ret);

//Send commands befor measure (required)

/*----- CH1 Input Condition -----*/
stat = viWrite(sesn, (unsigned char *)("INP1:IMP 1000000"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:COUP AC"), 12, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:ATT 1"), 10, &ret);

```

```

stat = viWrite(sesn, (unsigned char *)("INP1:FILT OFF"), 13, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:SLOP POS"), 14, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:HYST:REL 50"), 17, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:REL 50"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:ABS:AUTO ON"), 21,
&ret);
/*----- CH2 x Input Condition -----*/
stat = viWrite(sesn, (unsigned char *)("INP2:IMP 1000000"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:COUP AC"), 12, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:ATT 1"), 10, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:FILT OFF"), 13, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:SLOP POS"), 14, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:HYST:REL 50"), 17, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:REL 50"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:ABS:AUTO ON"), 21,
&ret);
/*-----*/

/*----- Freq & Ratio measure -----*/
stat = viWrite(sesn, (unsigned char *)("CONF:FREQ (@1)"), 14, &ret);
// stat = viWrite(sesn, "CONF:FREQ (@2)", 14, &ret);
// stat = viWrite(sesn, "CONF:FREQ (@3)", 14, &ret);
// stat = viWrite(sesn, "CONF:FREQ:RAT (@1),(@2)", 23, &ret);
// stat = viWrite(sesn, "CONF:FREQ:RAT (@1),(@3)", 23, &ret);
// stat = viWrite(sesn, "CONF:FREQ:RAT (@2),(@1)", 23, &ret);
// stat = viWrite(sesn, "CONF:FREQ:RAT (@3),(@1)", 23, &ret);
/*----- Gate Time -----*/
stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:SOUR IMM"), 17,
&ret);
stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:STOP:SOUR TIM"), 22,
&ret);
stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:STOP:TIM 1"), 19,
&ret);
/*----- Gate Ext -----*/
// stat = viWrite(sesn, "FREQ:ARM:SOUR Ext", 17, &ret);
// stat = viWrite(sesn, "FREQ:ARM:SLOP POS", 17, &ret);
// stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR Ext", 22, &ret);

```

```

// stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR TIM", 22, &ret);
// stat = viWrite(sesn, "FREQ:ARM:STOP:TIM 1", 19, &ret);
/*----- Gate Auto -----*/
// stat = viWrite(sesn, "FREQ:ARM:SOUR IMM", 17, &ret);
// stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR IMM", 22, &ret);
/*----- Gate Digits -----*/
// stat = viWrite(sesn, "FREQ:ARM:SOUR IMM", 17, &ret);
// stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR DIG", 22, &ret);
// stat = viWrite(sesn, "FREQ:ARM:STOP:DIG 5", 19, &ret);
/*-----*/

/*----- Read From Device -----*/
    stat = viWrite(sesn, (unsigned char *)("FETC?"), 5, &ret);
// stat = viWrite(sesn, "READ?", 5, &ret);
// stat = viWrite(sesn, "MEAS:FREQ?", 10, &ret);
// stat = viWrite(sesn, "SENS:DATA?", 10, &ret);
/*-----*/

    stat=viRead(sesn,(unsigned char *)readin,128,&ret);
    printf ("Read : %s",readin);
    stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, tmo);
    stat = viGpibControlREN(sesn, VI_GPIB_REN_ADDRESS_GTL);
    viClose (flist);
    viClose (defaultRM);
    return 0;
}

```

## 例 2 Time & Period 測定

```

// Time_Period.cpp : Defines the entry point for the console application.
#include "stdafx.h"
#include "visa.h"//include visa32.lib first to the probject.
#define MAX_COUNT 200
#include "stdio.h"

int main(int argc, char* argv[])
{
    ViStatus stat; /* For checking errors */

```

```

ViSession defaultRM; /* Communication channels */
ViSession sesn; /* Communication channel */
ViChar rsrcName[VI_FIND_BUFLen]; /* Serial resource name */
ViChar intfDesc[VI_FIND_BUFLen]; /* Port binding description */
ViUInt32 ret; /* To hold number of resources */
ViFindList flist; /* To hold list of resources */

char readin[128]="";
int tmo;

printf("\n ##### Start C++ Example program. #####\n");
printf(" We check the Counter on USB port and\n");
printf(" identify the first connected device.\n\n");

/* Begin by initializing the system */
stat = viOpenDefaultRM(&defaultRM);

if (stat < VI_SUCCESS) {
    /* Error Initializing VISA...exiting */
    printf ("Error Initializing NI-VISA...");
    return 0;
}
stat = viFindRsrc (defaultRM, "USB[0-9]::0x164E::0x1838::?*INSTR",
&flist, &ret, rsrcName);

stat = viOpen (defaultRM, rsrcName, VI_NULL, VI_NULL, &sesn);
if (stat < VI_SUCCESS)
{
    printf ("Could not open %s, status = 0x%08IX\n",rsrcName, stat);
    return 0;
}
else
{
    stat = viGetAttribute (sesn, VI_ATTR_INTF_INST_NAME, intfDesc);
    printf ("Resource %s, Description %s\n", rsrcName, intfDesc);
}
stat = viFindNext (flist, rsrcName);

```

```

//Setting Process Timeout
stat = viGetAttribute(sesn, VI_ATTR_TMO_VALUE, &tmo);
stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, 30000);

//send reset command '*RST' -- reset U6200A
stat = viWrite(sesn, (unsigned char *)("*RST"), 4, &ret);

//send Clear command '*CLS'-- Clear U6200A status register
stat = viWrite(sesn, (unsigned char *)("*CLS"), 4, &ret);

//send reset command 'INIT:CONT ON' -- Initial U6200A
stat = viWrite(sesn, (unsigned char *)("INIT:CONT ON"), 12, &ret);

//Send commands befor measure (required)

/*----- CH1 Input Condition -----*/
stat = viWrite(sesn, (unsigned char *)("INP1:IMP 1000000"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:COUP AC"), 12, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:ATT 1"), 10, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:FILT OFF"), 13, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:SLOP POS"), 14, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:HYST:REL 50"), 17, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:REL 50"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:ABS:AUTO ON"), 21,
&ret);
/*----- CH2 Input Condition -----*/
stat = viWrite(sesn, (unsigned char *)("INP2:IMP 1000000"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:COUP AC"), 12, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:ATT 1"), 10, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:FILT OFF"), 13, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:SLOP POS"), 14, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:HYST:REL 50"), 17, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:REL 50"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:ABS:AUTO ON"), 21,
&ret);
/*-----*/

```

```

/*----- Period measure -----*/
    stat = viWrite(sesn, (unsigned char *)("CONF:PER (@1)"), 13, &ret);
//  stat = viWrite(sesn, "CONF:PER (@2)", 13, &ret);
//  stat = viWrite(sesn, "CONF:PER (@3)", 13, &ret);
/*----- Gate Time -----*/
    stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:SOUR IMM"), 17,
&ret);
    stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:STOP:SOUR TIM"), 22,
&ret);
    stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:STOP:TIM 1"), 19,
&ret);
/*----- Gate Ext -----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR Ext", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:SLOP POS", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR Ext", 22, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR TIM", 22, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:TIM 1", 18, &ret);
/*----- Gate Auto -----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR IMM", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR IMM", 22, &ret);
/*----- Gate Digits -----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR IMM", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR DIG", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:DIG 5", 17, &ret);
/*-----*/

/*----- Rise/Fall Time, POS/NEG Width measure -----*/
//  stat = viWrite(sesn, "CONF:RISE:TIME", 14, &ret);
//  stat = viWrite(sesn, "CONF:FALL:TIME", 14, &ret);
//  stat = viWrite(sesn, "CONF:PWID", 9, &ret);
//  stat = viWrite(sesn, "CONF:NWID", 9, &ret);
/*----- Gate Ext -----*/
//  stat = viWrite(sesn, "TINT:ARM:ESTART:LAY2:SOUR IMM", 29, &ret);
//  stat = viWrite(sesn, "TINT:ARM:ESTART:LAY2:SOUR EXT", 29, &ret);
//  stat = viWrite(sesn, "TINT:ARM:ESTART:LAY2:SLOP POS", 29, &ret);
/*----- Gate Auto -----*/
//  stat = viWrite(sesn, "TINT:ARM:ESTART:LAY2:SOUR IMM", 29, &ret);

```

```

/*-----*/

/*----- Intval 1 to 2 measure -----*/
// stat = viWrite(sesn, "CONF:TINT (@1),(@2)", 19, &ret)
/*----- ARM Ext -----*/
// stat = viWrite(sesn, "TINT:ARM:ESTART:LAY2:SOUR EXT", 29, &ret);
// stat = viWrite(sesn, "TINT:ARM:ESTART:LAY2:SLOP POS", 29, &ret);
// stat = viWrite(sesn, "TINT:ARM:ESTART:LAY1:SOUR TIM", 29, &ret);
// stat = viWrite(sesn, "TINT:ARM:ESTART:LAY1:TIM 1Ms", 29, &ret);
// stat = viWrite(sesn, "TINT:ARM:ESTOP:LAY2:SOUR EXT", 29, &ret);
// stat = viWrite(sesn, "TINT:ARM:ESTOP:LAY2:SLOP POS", 29, &ret);
// stat = viWrite(sesn, "TINT:ARM:ESTOP:LAY1:SOUR TIM, ret);
//
// stat = viWrite(sesn,
"TINT:ARM:ESTOP:LAY1:TIM 1mS", 29, &ret);
/*----- ARM Auto -----*/
// stat = viWrite(sesn, "TINT:ARM:ESTART:LAY2:SOUR IMM", 29, &ret);
// stat = viWrite(sesn, "TINT:ARM:ESTOP:LAY2:SOUR IMM", 29, &ret);
// stat = viWrite(sesn, "TINT:ARM:ESTOP:LAY1:SOUR IMM", 29, &ret);
// stat = viWrite(sesn, "TINT:ARM:ESTOP:LAY1:SOUR TIM", 29, &ret);
// stat = viWrite(sesn, "TINT:ARM:ESTOP:LAY1:TIM 1mS", 29, &ret);
/*-----*/

/*----- Read From Device -----*/
stat = viWrite(sesn, (unsigned char *)("FETC?"), 5, &ret);
// stat = viWrite(sesn, "READ?", 5, &ret);
// stat = viWrite(sesn, "MEAS:FREQ?", 10, &ret);
// stat = viWrite(sesn, "SENS:DATA?", 10, &ret);
/*-----*/

stat=viRead(sesn,(unsigned char *)readin,128,&ret);
printf ("Read : %s",readin);
stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, tmo);
stat = viGpibControlREN(sesn, VI_GPIB_REN_ADDRESS_GTL);
viClose (flist);
viClose (defaultRM);
return 0;
}

```

### 例 3 Volt Peak 測定

```
// Volt_Peak.cpp : Defines the entry point for the console application.
#include "stdafx.h"
#include "visa.h"//include visa32.lib first to the probject.
#define MAX_COUNT 200
#include "stdio.h"

int main(int argc, char* argv[])
{
    ViStatus stat; /* For checking errors */
    ViSession defaultRM; /* Communication channels */
    ViSession sesn; /* Communication channel */
    ViChar rsrcName[VI_FIND_BUFLLEN]; /* Serial resource name */
    ViChar intfDesc[VI_FIND_BUFLLEN]; /* Port binding description */
    ViUInt32 ret; /* To hold number of resources */
    ViFindList flist; /* To hold list of resources */

    char readin[128]="";
    int tmo;

    printf("\n ##### Start C++ Example program. #####\n");
    printf(" We check the Counter on USB port and\n");
    printf(" identify the first connected device.\n\n");

    /* Begin by initializing the system */
    stat = viOpenDefaultRM(&defaultRM);

    if (stat < VI_SUCCESS) {
        /* Error Initializing VISA...exiting */
        printf ("Error Initializing NI-VISA...");
        return 0;
    }

    stat = viFindRsrc (defaultRM, "USB[0-9]::0x164E::0x1838::?*INSTR",
&flist, &ret, rsrcName);
    stat = viOpen (defaultRM, rsrcName, VI_NULL, VI_NULL, &sesn);
    if (stat < VI_SUCCESS)
```

```

{
    printf ("Could not open %s, status = 0x%08IX\n",rsrcName, stat);
    return 0;
}
else
{
    stat = viGetAttribute (sesn, VI_ATTR_INTF_INST_NAME, intfDesc);
    printf ("Resource %s, Description %s\n", rsrcName, intfDesc);
}
stat = viFindNext (flist, rsrcName);

//Setting Process Timeout
stat = viGetAttribute(sesn, VI_ATTR_TMO_VALUE, &tmo);
stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, 30000);

//send reset command '*RST' -- reset U6200A
stat = viWrite(sesn, (unsigned char *)("*RST"), 4, &ret);

//send Clear command '*CLS'-- Clear U6200A status register
stat = viWrite(sesn, (unsigned char *)("*CLS"), 4, &ret);

//send reset command 'INIT:CONT ON' -- Initial U6200A
stat = viWrite(sesn, (unsigned char *)("INIT:CONT ON"), 12, &ret);

//Send commands befor measure (required)

/*----- CH1 Input Condition -----*/
stat = viWrite(sesn, (unsigned char *)("INP1:IMP 1000000"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:COUP AC"), 12, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:ATT 1"), 10, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:FILT OFF"), 13, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:SLOP POS"), 14, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:HYST:REL 50"), 17, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:REL 50"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:ABS:AUTO ON"), 21,
&ret);
/*----- CH2 Input Condition -----*/

```

```

stat = viWrite(sesn, (unsigned char *)("INP2:IMP 1000000"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:COUP AC"), 12, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:ATT 1"), 10, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:FILT OFF"), 13, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:SLOP POS"), 14, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:HYST:REL 50"), 17, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:REL 50"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:ABS:AUTO ON"), 21,
&ret);
/*-----*/

/*----- Volt Peak measure -----*/
stat = viWrite(sesn, (unsigned char *)("CONF:VOLT:PTP (@1)"), 18, &ret);
// stat = viWrite(sesn, "CONF:VOLT:PTP (@2)", 18, &ret);
// stat = viWrite(sesn, "CONF:VOLT:MAX (@1)", 18, &ret);
// stat = viWrite(sesn, "CONF:VOLT:MAX (@2)", 18, &ret);
// stat = viWrite(sesn, "CONF:VOLT:MIN (@1)", 18, &ret);
// stat = viWrite(sesn, "CONF:VOLT:MIN (@2)", 18, &ret);
// stat = viWrite(sesn, "CONF:VOLT:CENT (@1)", 19, &ret);
// stat = viWrite(sesn, "CONF:VOLT:CENT (@2)", 19, &ret);
/*-----*/

/*----- Read From Device -----*/
stat = viWrite(sesn, (unsigned char *)("FETC?"), 5, &ret);
// stat = viWrite(sesn, "READ?", 5, &ret);
// stat = viWrite(sesn, "MEAS:FREQ?", 10, &ret);
// stat = viWrite(sesn, "SENS:DATA?", 10, &ret);
/*-----*/

stat=viRead(sesn,(unsigned char *)readin,128,&ret);
printf ("Read : %s",readin);
stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, tmo);
stat = viGpibControlREN(sesn, VI_GPIB_REN_ADDRESS_GTL);
viClose (flist);
viClose (defaultRM);
return 0;
}

```

#### 例 4 その他測定

```
// OtherMeas.cpp : Defines the entry point for the console application.
#include "stdafx.h"
#include "visa.h"//include visa32.lib first to the probject.
#define MAX_COUNT 200
#include "stdio.h"

int main(int argc, char* argv[])
{
    ViStatus stat; /* For checking errors */
    ViSession defaultRM; /* Communication channels */
    ViSession sesn; /* Communication channel */
    ViChar rsrcName[VI_FIND_BUFLLEN]; /* Serial resource name */
    ViChar intfDesc[VI_FIND_BUFLLEN]; /* Port binding description */
    ViUInt32 ret; /* To hold number of resources */
    ViFindList flist; /* To hold list of resources */

    char readin[128]="";
    int tmo;

    printf("\n ##### Start C++ Example program. #####\n");
    printf(" We check the Counter on USB port and\n");
    printf(" identify the first connected device.\n\n");

    /* Begin by initializing the system */
    stat = viOpenDefaultRM(&defaultRM);

    if (stat < VI_SUCCESS) {
        /* Error Initializing VISA...exiting */
        printf ("Error Initializing NI-VISA...");
        return 0;
    }

    stat = viFindRsrc (defaultRM, "USB[0-9]::0x164E::0x1838::?*INSTR",
&flist, &ret, rsrcName);

    stat = viOpen (defaultRM, rsrcName, VI_NULL, VI_NULL, &sesn);
```

```

if (stat < VI_SUCCESS)
{
    printf ("Could not open %s, status = 0x%08IX\n", rsrcName, stat);
    return 0;
}
else
{
    stat = viGetAttribute (sesn, VI_ATTR_INTF_INST_NAME, intfDesc);
    printf ("Resource %s, Description %s\n", rsrcName, intfDesc);
}
stat = viFindNext (flist, rsrcName);

//Setting Process Timeout
stat = viGetAttribute(sesn, VI_ATTR_TMO_VALUE, &tmo);
stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, 30000);

//send reset command '*RST' -- reset U6200A
stat = viWrite(sesn, (unsigned char *)("*RST"), 4, &ret);

//send Clear command '*CLS'-- Clear U6200A status register
stat = viWrite(sesn, (unsigned char *)("*CLS"), 4, &ret);

//send reset command 'INIT:CONT ON' -- Initial U6200A
stat = viWrite(sesn, (unsigned char *)("INIT:CONT ON"), 12, &ret);

//Send commands befor measure (required)

/*----- CH1 Input Condition -----*/
stat = viWrite(sesn, (unsigned char *)("INP1:IMP 1000000"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:COUP AC"), 12, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:ATT 1"), 10, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:FILT OFF"), 13, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:SLOP POS"), 14, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:HYST:REL 50"), 17, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:REL 50"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:ABS:AUTO ON"), 21,
&ret);

```

```

/*----- CH2 Input Condition -----*/
    stat = viWrite(sesn, (unsigned char *)("INP2:IMP 1000000"), 16, &ret);
    stat = viWrite(sesn, (unsigned char *)("INP2:COUP AC"), 12, &ret);
    stat = viWrite(sesn, (unsigned char *)("INP2:ATT 1"), 10, &ret);
    stat = viWrite(sesn, (unsigned char *)("INP2:FILT OFF"), 13, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN2:SLOP POS"), 14, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN2:HYST:REL 50"), 17, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:REL 50"), 16, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:ABS:AUTO ON"), 21,
&ret);
/*-----*/

/*----- Phase measure -----*/
    stat = viWrite(sesn, (unsigned char *)("CONF:PHAS (@1),(@2)"), 19,
&ret);
/*----- Gate Ext -----*/
    stat = viWrite(sesn, (unsigned char *)("PHAS:ARM:SOUR IMM"), 17,
&ret);
    stat = viWrite(sesn, (unsigned char *)("PHAS:ARM:SOUR EXT"), 17, &ret);
    stat = viWrite(sesn, (unsigned char *)("PHAS:ARM:SLOP POS"), 17, &ret);
/*----- Gate Auto-----*/
//    stat = viWrite(sesn, "PHAS:ARM:SOUR EXT", 17, &ret);
//    stat = viWrite(sesn, "PHAS:ARM:SLOP POS", 17, &ret);
/*-----*/

/*----- Duty Cycle measure -----*/
//    stat = viWrite(sesn, "CONF:DCYC (@1),(@2)", 19, &ret);
/*----- Gate Ext -----*/
//    stat = viWrite(sesn, "TINT:ARM:ESTART:LAY2:SOUR IMM", 29, &ret);
//    stat = viWrite(sesn, "TINT:ARM:ESTART:LAY2:SOUR EXT", 29, &ret);
//    stat = viWrite(sesn, "TINT:ARM:ESTART:LAY2:SLOP POS", 29, &ret);
/*----- Gate Auto -----*/
//    stat = viWrite(sesn, "TINT:ARM:ESTART:LAY2:SOUR IMM", 29, &ret);
/*-----*/

/*----- Total CH1 measure -----*/
//    stat = viWrite(sesn, "CONF:TOT:TIM (@1) ", 18, &ret);

```

```

/*----- Gate Ext -----*/
// stat = viWrite(sesn, "TOT:ARM:SOUR IMM", 16, &ret);
// stat = viWrite(sesn, "TOT:ARM:STOP:SOUR TIM", 21, &ret);
// stat = viWrite(sesn, "TOT:ARM:STOP:TIM 1", 18, &ret);
/*----- Gate Time -----*/
// stat = viWrite(sesn, "TOT:ARM:SOUR IMM", 16, &ret);
// stat = viWrite(sesn, "TOT:ARM:STOP:SOUR TIM", 21, &ret);
// stat = viWrite(sesn, "TOT:ARM:STOP:TIM 1", 18, &ret);
/*----- ARM Auto -----*/
// stat = viWrite(sesn, "TOT:ARM:SOUR IMM", 16, &ret);
// stat = viWrite(sesn, "TOT:ARM:STOP:SOUR IMM", 21, &ret);
/*-----*/

/*----- Read From Device -----*/
stat = viWrite(sesn, (unsigned char *)("FETC?"), 5, &ret);
// stat = viWrite(sesn, "READ?", 5, &ret);
// stat = viWrite(sesn, "MEAS:FREQ?", 10, &ret);
// stat = viWrite(sesn, "SENS:DATA?", 10, &ret);
/*-----*/

stat=viRead(sesn,(unsigned char *)readin,128,&ret);
printf ("Read : %s",readin);
stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, tmo);
stat = viGpibControlREN(sesn, VI_GPIB_REN_ADDRESS_GTL);
viClose (flist);
viClose (defaultRM);
return 0;
}

```

## 測定に Scale & Offset を使用

### 例 5 Scale & Offset

```

// Scal_Offset.cpp : Defines the entry point for the console application.
#include "stdafx.h"
#include "visa.h"//include visa32.lib first to the project.
#define MAX_COUNT 200
#include "stdio.h"

```

```

int main(int argc, char* argv[])
{
    ViStatus stat; /* For checking errors */
    ViSession defaultRM; /* Communication channels */
    ViSession sesn; /* Communication channel */
    ViChar rsrcName[VI_FIND_BUFLLEN]; /* Serial resource name */
    ViChar intfDesc[VI_FIND_BUFLLEN]; /* Port binding description */
    ViUInt32 ret; /* To hold number of resources */
    ViFindList flist; /* To hold list of resources */

    char readin[128]="";
    int tmo;

    printf("\n ##### Start C++ Example program. #####\n");
    printf(" We check the Counter on USB port and\n");
    printf(" identify the first connected device.\n\n");

    /* Begin by initializing the system */
    stat = viOpenDefaultRM(&defaultRM);

    if (stat < VI_SUCCESS) {
        /* Error Initializing VISA...exiting */
        printf ("Error Initializing NI-VISA...");
        return 0;
    }
    stat = viFindRsrc (defaultRM, "USB[0-9]::0x164E::0x1838::?*INSTR",
&flist, &ret, rsrcName);

    stat = viOpen (defaultRM, rsrcName, VI_NULL, VI_NULL, &sesn);
    if (stat < VI_SUCCESS)
    {
        printf ("Could not open %s, status = 0x%08IX\n",rsrcName, stat);
        return 0;
    }
    else
    {
        stat = viGetAttribute (sesn, VI_ATTR_INTF_INST_NAME, intfDesc);

```

```

        printf ("Resource %s, Description %s\n", rsrcName, intfDesc);
    }
    stat = viFindNext (flist, rsrcName);

    //Setting Process Timeout
    stat = viGetAttribute(sesn, VI_ATTR_TMO_VALUE, &tmo);
    stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, 30000);

    //send reset command '*RST' -- reset U6200A
    stat = viWrite(sesn, (unsigned char *)("*RST"), 4, &ret);

    //send Clear command '*CLS'-- Clear U6200A status register
    stat = viWrite(sesn, (unsigned char *)("*CLS"), 4, &ret);

    //send reset command 'INIT:CONT ON' -- Initial U6200A
    stat = viWrite(sesn, (unsigned char *)("INIT:CONT ON"), 12, &ret);

    //Send commands befor measure (required)
    /*----- CH1 Input Condition -----*/
    stat = viWrite(sesn, (unsigned char *)("INP1:IMP 1000000"), 16, &ret);
    stat = viWrite(sesn, (unsigned char *)("INP1:COUP AC"), 12, &ret);
    stat = viWrite(sesn, (unsigned char *)("INP1:ATT 1"), 10, &ret);
    stat = viWrite(sesn, (unsigned char *)("INP1:FILT OFF"), 13, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN1:SLOP POS"), 14, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN1:HYST:REL 50"), 17, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:REL 50"), 16, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:ABS:AUTO ON"), 21,
&ret);
    /*----- CH2 Input Condition -----*/
    stat = viWrite(sesn, (unsigned char *)("INP2:IMP 1000000"), 16, &ret);
    stat = viWrite(sesn, (unsigned char *)("INP2:COUP AC"), 12, &ret);
    stat = viWrite(sesn, (unsigned char *)("INP2:ATT 1"), 10, &ret);
    stat = viWrite(sesn, (unsigned char *)("INP2:FILT OFF"), 13, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN2:SLOP POS"), 14, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN2:HYST:REL 50"), 17, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:REL 50"), 16, &ret);
    stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:ABS:AUTO ON"), 21,

```

```

&ret);
/*-----*/

/*----- Freq & Ratio measure -----*/
    stat = viWrite(sesn, (unsigned char *)("CONF:FREQ (@1)", 14, &ret);
//  stat = viWrite(sesn, "CONF:FREQ (@2)", 14, &ret);
//  stat = viWrite(sesn, "CONF:FREQ (@3)", 14, &ret);
//  stat = viWrite(sesn, "CONF:FREQ:RAT (@1),(@2)", 23, &ret);
//  stat = viWrite(sesn, "CONF:FREQ:RAT (@1),(@3)", 23, &ret);
//  stat = viWrite(sesn, "CONF:FREQ:RAT (@2),(@1)", 23, &ret);
//  stat = viWrite(sesn, "CONF:FREQ:RAT (@3),(@1)", 23, &ret);
/*----- Gate Time-----*/
    stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:SOUR IMM", 17,
&ret);
    stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:STOP:SOUR TIM", 22,
&ret);
    stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:STOP:TIM 1", 19,
&ret);
/*----- Gate Ext-----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR Ext", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:SLOP POS", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR Ext", 22, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR TIM", 22, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:TIM 1", 19, &ret);
/*----- Gate Auto-----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR IMM", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR IMM", 22, &ret);
/*----- Gate Digits-----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR IMM", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR DIG", 22, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:DIG 5", 19, &ret);
/*-----*/

/*----- Scale & Offset -----*/
    stat = viWrite(sesn, (unsigned char *)("TRAC:DATA SCALE,3", 17, &ret);
    stat = viWrite(sesn, (unsigned char *)("TRAC:DATA OFFSET,5MHz", 21,
&ret);

```

```

/*-----*/

/*----- Read From Device-----*/
    stat = viWrite(sesn, (unsigned char *)("FETC?"), 5, &ret);
//  stat = viWrite(sesn, "READ?", 5, &ret);
//  stat = viWrite(sesn, "MEAS:FREQ?", 10, &ret);
//  stat = viWrite(sesn, "SENS:DATA?", 10, &ret);
/*-----*/

    stat=viRead(sesn,(unsigned char *)readin,128,&ret);
    printf ("Read : %s",readin);
    stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, tmo);
    stat = viGpibControlREN(sesn, VI_GPIB_REN_ADDRESS_GTL);
    viClose (flist);
    viClose (defaultRM);
    return 0;
}

```

## Limit を測定に使用する

### 例 6 Limit Modes

// Limit.cpp : Defines the entry point for the console application.

```

#include "stdafx.h"
#include "visa.h"//include visa32.lib first to the project.
#define MAX_COUNT 200
#include "stdio.h"

int main(int argc, char* argv[]){
    ViStatus stat; /* For checking errors */
    ViSession defaultRM; /* Communication channels */
    ViSession sesn; /* Communication channel */
    ViChar rsrcName[VI_FIND_BUFLen]; /* Serial resource name */
    ViChar intfDesc[VI_FIND_BUFLen]; /* Port binding description */
    ViUInt32 ret; /* To hold number of resources */
    ViFindList flist; /* To hold list of resources */

    char readin[128]="";

```

```

int tmo;

printf("\n ##### Start C++ Example program. #####\n");
printf(" We check the Counter on USB port and\n");
printf(" identify the first connected device.\n\n");

/* Begin by initializing the system */
stat = viOpenDefaultRM(&defaultRM);

if (stat < VI_SUCCESS) {
    /* Error Initializing VISA...exiting */
    printf ("Error Initializing NI-VISA...");
    return 0;
}
stat = viFindRsrc (defaultRM, "USB[0-9]::0x164E::0x1838::?*INSTR",
&flist, &ret, rsrcName);

stat = viOpen (defaultRM, rsrcName, VI_NULL, VI_NULL, &sesn);
if (stat < VI_SUCCESS)
{
    printf ("Could not open %s, status = 0x%08IX\n",rsrcName, stat);
    return 0;
}
else
{
    stat = viGetAttribute (sesn, VI_ATTR_INTF_INST_NAME, intfDesc);
    printf ("Resource %s, Description %s\n", rsrcName, intfDesc);
}
stat = viFindNext (flist, rsrcName);

//Setting Process Timeout
stat = viGetAttribute(sesn, VI_ATTR_TMO_VALUE, &tmo);
stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, 30000);

//send reset command '*RST' -- reset U6200A
stat = viWrite(sesn, (unsigned char *)("*RST"), 4, &ret);

```

```

//send Clear command '*CLS'-- Clear U6200A status register
stat = viWrite(sesn, (unsigned char *)("*CLS"), 4, &ret);

//send reset command 'INIT:CONT ON' -- Initial U6200A
stat = viWrite(sesn, (unsigned char *)("INIT:CONT ON"), 12, &ret);

//Send commands befor measure (required)
/*----- CH1 Input Condition -----*/
stat = viWrite(sesn, (unsigned char *)("INP1:IMP 1000000"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:COUP AC"), 12, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:ATT 1"), 10, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:FILT OFF"), 13, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:SLOP POS"), 14, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:HYST:REL 50"), 17, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:REL 50"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:ABS:AUTO ON"), 21,
&ret);
/*----- CH2 Input Condition -----*/
stat = viWrite(sesn, (unsigned char *)("INP2:IMP 1000000"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:COUP AC"), 12, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:ATT 1"), 10, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:FILT OFF"), 13, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:SLOP POS"), 14, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:HYST:REL 50"), 17, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:REL 50"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:ABS:AUTO ON"), 21,
&ret);
/*-----*/

/*----- Freq & Ratio measure -----*/
stat = viWrite(sesn, (unsigned char *)("CONF:FREQ (@1)", 14, &ret);
// stat = viWrite(sesn, "CONF:FREQ (@2)", 14, &ret);
// stat = viWrite(sesn, "CONF:FREQ (@3)", 14, &ret);
// stat = viWrite(sesn, "CONF:FREQ:RAT (@1),(@2)", 23, &ret);
// stat = viWrite(sesn, "CONF:FREQ:RAT (@1),(@3)", 23, &ret);
// stat = viWrite(sesn, "CONF:FREQ:RAT (@2),(@1)", 23, &ret);

```

```

//  stat = viWrite(sesn, "CONF:FREQ:RAT (@3),(@1)", 23, &ret);
/*-----*/

/*----- Gate Time -----*/
    stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:SOUR IMM"), 17,
&ret);
    stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:STOP:SOUR TIM"), 22,
&ret);
    stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:STOP:TIM 1"), 19,
&ret);
/*----- Gate Ext -----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR Ext", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:SLOP POS", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR Ext", 22, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR TIM", 22, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:TIM 1", 19, &ret);
/*----- Gate Auto-----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR IMM", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR IMM", 22, &ret);
/*----- Gate Digits-----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR IMM", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR DIG", 22, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:DIG 5", 19, &ret);
/*-----*/

/*----- Scale & Offset -----*/
    stat = viWrite(sesn, (unsigned char *)("CALC2:LIM:STAT ON"), 17, &ret);
    stat = viWrite(sesn, (unsigned char *)("CALC2:LIM:UPP 11MHz"), 19,
&ret);
    stat = viWrite(sesn, (unsigned char *)("CALC2:LIM:LOW 9MHz"), 18,
&ret);
    stat = viWrite(sesn, (unsigned char *)("CALC2:IMM:AUTO ON"), 17, &ret);
    stat = viWrite(sesn, (unsigned char *)("INIT:AUTO OFF"), 13, &ret);
    stat = viWrite(sesn, (unsigned char *)("INIT:CONT ON"), 12, &ret);
/*-----*/

/*----- Read From Device -----*/

```

```

    stat = viWrite(sesn, (unsigned char *)("FETC?"), 5, &ret);
//  stat = viWrite(sesn, "READ?", 5, &ret);
//  stat = viWrite(sesn, "MEAS:FREQ?", 10, &ret);
//  stat = viWrite(sesn, "SENS:DATA?", 10, &ret);
/*-----*/

    stat=viRead(sesn,(unsigned char *)readin,128,&ret);
    printf ("Read : %s",readin);
    stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, tmo);
    stat = viGpibControlREN(sesn, VI_GPIB_REN_ADDRESS_GTL);
    viClose (flist);
    viClose (defaultRM);
    return 0;
}

```

## Stats を測定に使用する

### 例 7 Stats モード

```

// Stats.cpp : Defines the entry point for the console application.
#include "stdafx.h"
#include "visa.h"//include visa32.lib first to the probject.
#define MAX_COUNT 200
#include "stdio.h"

int main(int argc, char* argv[])
{
    ViStatus stat; /* For checking errors */
    ViSession defaultRM; /* Communication channels */
    ViSession sesn; /* Communication channel */
    ViChar rsrcName[VI_FIND_BUFLen]; /* Serial resource name */
    ViChar intfDesc[VI_FIND_BUFLen]; /* Port binding description */
    ViUInt32 ret; /* To hold number of resources */
    ViFindList flist; /* To hold list of resources */

    char readin[128]="";
    int tmo;

    printf("\n ##### Start C++ Example program. #####\n");

```

```

printf(" We check the Counter on USB port and\n");
printf(" identify the first connected device.\n\n");

/* Begin by initializing the system */
stat = viOpenDefaultRM(&defaultRM);

if (stat < VI_SUCCESS) {
    /* Error Initializing VISA...exiting */
    printf ("Error Initializing NI-VISA...");
    return 0;
}
stat = viFindRsrc (defaultRM, "USB[0-9]::0x164E::0x1838::?*INSTR",
&flist, &ret, rsrcName);

stat = viOpen (defaultRM, rsrcName, VI_NULL, VI_NULL, &sesn);
if (stat < VI_SUCCESS)
{
    printf ("Could not open %s, status = 0x%08IX\n",rsrcName, stat);
    return 0;
}
else
{
    stat = viGetAttribute (sesn, VI_ATTR_INTF_INST_NAME, intfDesc);
    printf ("Resource %s, Description %s\n", rsrcName, intfDesc);
}
stat = viFindNext (flist, rsrcName);

//Setting Process Timeout
stat = viGetAttribute(sesn, VI_ATTR_TMO_VALUE, &tmo);
stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, 30000);

//send reset command '*RST' -- reset U6200A
stat = viWrite(sesn, (unsigned char *)("*RST"), 4, &ret);

//send Clear command '*CLS'-- Clear U6200A status register
stat = viWrite(sesn, (unsigned char *)("*CLS"), 4, &ret);

```

```

//send reset command 'INIT:CONT ON' -- Initial U6200A
stat = viWrite(sesn, (unsigned char *)("INIT:CONT ON"), 12, &ret);

//Send commands before measure (required)
/*----- CH1 Input Condition -----*/
stat = viWrite(sesn, (unsigned char *)("INP1:IMP 1000000"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:COUP AC"), 12, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:ATT 1"), 10, &ret);
stat = viWrite(sesn, (unsigned char *)("INP1:FILT OFF"), 13, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:SLOP POS"), 14, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:HYST:REL 50"), 17, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:REL 50"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN1:LEV:ABS:AUTO ON"), 21,
&ret);
/*----- CH2 Input Condition -----*/
stat = viWrite(sesn, (unsigned char *)("INP2:IMP 1000000"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:COUP AC"), 12, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:ATT 1"), 10, &ret);
stat = viWrite(sesn, (unsigned char *)("INP2:FILT OFF"), 13, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:SLOP POS"), 14, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:HYST:REL 50"), 17, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:REL 50"), 16, &ret);
stat = viWrite(sesn, (unsigned char *)("EVEN2:LEV:ABS:AUTO ON"), 21,
&ret);
/*-----*/

/*----- Freq & Ratio measure-----*/
stat = viWrite(sesn, (unsigned char *)("CONF:FREQ (@1)"), 14, &ret);
// stat = viWrite(sesn, "CONF:FREQ (@2)", 14, &ret);
// stat = viWrite(sesn, "CONF:FREQ (@3)", 14, &ret);
// stat = viWrite(sesn, "CONF:FREQ:RAT (@1),(@2)", 23, &ret);
// stat = viWrite(sesn, "CONF:FREQ:RAT (@1),(@3)", 23, &ret);
// stat = viWrite(sesn, "CONF:FREQ:RAT (@2),(@1)", 23, &ret);
// stat = viWrite(sesn, "CONF:FREQ:RAT (@3),(@1)", 23, &ret);
/*-----*/
/*----- Gate Time -----*/
stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:SOUR IMM"), 17,

```

```

&ret);
    stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:STOP:SOUR TIM"), 22,
&ret);
    stat = viWrite(sesn, (unsigned char *)("FREQ:ARM:STOP:TIM 1"), 19,
&ret);
/*----- Gate Ext -----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR Ext", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:SLOP POS", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR Ext", 22, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR TIM", 22, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:TIM 1", 19, &ret);
/*----- Gate Auto -----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR IMM", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR IMM", 22, &ret);
/*----- Gate Digits -----*/
//  stat = viWrite(sesn, "FREQ:ARM:SOUR IMM", 17, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:SOUR DIG", 22, &ret);
//  stat = viWrite(sesn, "FREQ:ARM:STOP:DIG 5", 19, &ret);
/*-----*/

/*----- Scale & Offset -----*/
    stat = viWrite(sesn, (unsigned char *)("CALC3:AVER:STAT ON"), 18,
&ret);
    stat = viWrite(sesn, (unsigned char *)("CALC3:AVER:COUN 10"), 18,
&ret);
    stat = viWrite(sesn, (unsigned char *)("CALC3:LFIL:stat OFF"), 19, &ret);
    stat = viWrite(sesn, (unsigned char *)("CALC3:AVER:Type SDEV"), 20,
&ret);
    stat = viWrite(sesn, (unsigned char *)("TRIG:COUN:AUTO OFF"), 18,
&ret);
    stat = viWrite(sesn, (unsigned char *)("INIT:CONT ON"), 12, &ret);
/*-----*/

/*----- Read From Device -----*/
    stat = viWrite(sesn, (unsigned char *)("FETC?"), 5, &ret);
//  stat = viWrite(sesn, "READ?", 5, &ret);
//  stat = viWrite(sesn, "MEAS:FREQ?", 10, &ret);

```

```
// stat = viWrite(sesn, "SENS:DATA?", 10, &ret);  
/*-----*/  
  
stat=viRead(sesn,(unsigned char *)readin,128,&ret);  
printf ("Read : %s",readin);  
stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, tmo);  
stat = viGpibControlREN(sesn, VI_GPIB_REN_ADDRESS_GTL);  
viClose (flist);  
viClose (defaultRM);  
return 0;  
}
```



**Conformity with the following European Directives:**

The product herein conforms with the requirements of the Low Voltage Directive 2006/95/EC and the EMC Directive 2004/108/EC and goes with the CE Marking accordingly.

**Conformity with the following product standards:**

**Manufacturer Name:** Picotest Corp.

**Manufacturer Address:** 5F-1, 286-9, Hsin-Ya Rd, 80673, Kaohsiung, Taiwan

**Declaration of Product**

**Product Name:** Universal Counter

**Model Number:** U6200A

**EMC:**

**EN61326-1:2006**

**EN61326-2-1:2006**

**EMI:**

**CISPR 11:2009+A1:2010+A2:2010 Class A**

**EN61000-3-2:2006+A1:2009+A2:2009**

**EN61000-3-3:2008**

**EMS:**

**IEC61000-4-2:2008**

**IEC61000-4-3:2010**

**IEC61000-4-4:2011**

**IEC61000-4-5:2005**

**IEC61000-4-6:2008**

**IEC61000-4-11:2004**

**Safety:**

**IEC61010-1:2010(3<sup>rd</sup> Edition)/EN61010-1:2010**

**IEC61010-2-030:2010(1<sup>st</sup> Edition)/EN61010-2-030:2010**

13 Aug. 2009

Date



Mr. Hawk Shang  
General Manager

For more information, please contact your local supplier, sales office or distributor.



**Conformity with the following European Directives:**

The product herein conforms with the requirements of the Low Voltage Directive 2006/95/EC and the EMC Directive 2004/108/EC and goes with the CE Marking accordingly.

**Conformity with the following product standards:**

**Manufacturer Name:** Picotest Corp.

**Manufacturer Address:** 5F-1, 286-9, Hsin-Ya Rd, 80673, Kaohsiung, Taiwan

**Declaration of Product**

**Product Name:** Universal Counter

**Model Number:** U6220A

**EMC:**

**EN61326-1:2006**

**EN61326-2-1:2006**

**EMI:**

**CISPR 11:2009+A1:2010+A2:2010 Class A**

**EN61000-3-2:2006+A1:2009+A2:2009**

**EN61000-3-3:2008**

**EMS:**

**IEC61000-4-2:2008**

**IEC61000-4-3:2010**

**IEC61000-4-4:2011**

**IEC61000-4-5:2005**

**IEC61000-4-6:2008**

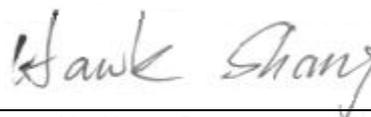
**IEC61000-4-11:2004**

**Safety:**

**IEC61010-1:2010(3rd Edition)/EN61010-1:2010**

**IEC61010-2-030:2010(1st Edition)/EN61010-2-030:2010**

13 Aug. 2009  
Date



Mr. Hawk Shang  
General Manager

For more information, please contact your local supplier, sales office or distributor.