

---

---

# ADVANTEST®

株式会社アドバンテスト

---

D3371

トランスミッション・アナライザ

リモート・プログラミング・マニュアル

MANUAL NUMBER FFJ-8370628B00

---



## 本器を安全に取り扱うための注意事項

本器の機能を十分にご理解いただき、より効果的にご利用いただくために、必ずご使用前に取扱説明書をお読み下さい。また、本器の誤った使用、不適切な使用等に起因する運用結果につきましては、当社は責任を負いかねますのでご了承下さい。

本器の操作・保守等の作業を行う場合、誤った方法で使用すると本器の保護機能がそこなわれることがあります。常に安全に心がけてご使用頂くようお願い致します。

### ■危険警告ラベル

アドバンテストの製品には、特有の危険が存在する場所に危険警告ラベルが貼られています。取り扱いには十分注意して下さい。また、これらのラベルを破いたり、傷つけたりしないで下さい。また、日本国内で製品を購入し海外で使用する場合は、必要に応じて英語版の危険警告ラベルをお貼り下さい。危険警告ラベルについてのお問い合わせは、当社の最寄りの営業所までお願いします。所在地および電話番号は巻末に記載してあります。

危険警告ラベルのシグナル・ワードとその定義は、以下のとおりです。

- 危険： 死または重度の障害が差し迫っている。
- 警告： 死または重度の障害が起こる可能性がある。
- 注意： 軽度の人身障害あるいは物損が起こる可能性がある。

### ■基本的注意事項

火災、火傷、感電、怪我などの防止のため、以下の注意事項をお守り下さい。

- 電源電圧に応じた電源ケーブルを使用して下さい。ただし、海外で使用する場合は、それぞれの国の安全規格に適合した電源ケーブルを使用して下さい。また、電源ケーブルの上には重いものをのせないで下さい。
- 電源プラグをコンセントに差し込むときは、電源スイッチを OFF にしてから奥までしっかり差し込んで下さい。
- 電源プラグをコンセントから抜くときは、電源スイッチを OFF にしてから、電源ケーブルを引っぱらずにプラグを持って抜いて下さい。このとき、濡れた手で抜かないで下さい。
- 電源投入前に、本器の電源電圧が供給電源電圧と一致していることを確認して下さい。
- 電源ケーブルは、保護導体端子を備えた電源コンセントに接続して下さい。保護導体端子を備えていない延長コードを使用すると、保護接地が無効になります。
- 3ピン-2ピン変換アダプタ（弊社の製品には添付していません）を使用する場合は、アダプタから出ている接地ピンをコンセントのアース端子に接続し、大地接地して下さい。また、アダプタの接地ピンの短絡に注意して下さい。
- 電源電圧に適合した規格のヒューズを使用して下さい。
- ケースを開けたままで本器を使用しないで下さい。

## 本器を安全に取り扱うための注意事項

- 規定の周囲環境で本器を使用して下さい。
- 製品の上に物をのせたり、製品の上から力を加えたりしないで下さい。また、花瓶や薬品などの液体の入った容器を製品のそばに置かないで下さい。
- 通気孔のある製品については、通気孔に金属類や燃えやすい物などを差し込んだり、落としたりしないで下さい。
- 台車に載せて使用する場合は、ベルト等によって落下防止を行って下さい。
- 周辺機器を接続する場合は、本器の電源を切ってから接続して下さい。





### ■取扱説明書中の注意表記

取扱説明書中で使用している注意事項に関するシグナル・ワードとその定義は以下のとおりです。

- 危険： 重度の人身障害（死亡や重傷）の恐れがある注意事項  
警告： 人身の安全／健康に関する注意事項  
注意： 製品／設備の損傷に関する注意事項または使用上の制限事項

### ■製品上の安全マーク

アドバンテストの製品には、以下の安全マークが付いています。

- ： 取扱い注意を示しています。人体および製品を保護するため、取扱説明書を参照する必要がある場所に付いています。
- ： アース記号を示しています。感電防止のため機器を使用する前に、接地が必要なフィールド・ワイヤリング端子を示しています。
- ： 高電圧危険を示しています。1000V 以上の電圧が人力または出力される場所に付いています。
- ： 感電注意を示しています。

### ■寿命部品の交換について

計測器に使用されている主な寿命部品は以下のとおりです。  
製品の性能、機能を維持するために、寿命を目安に早めに交換して下さい。  
ただし、製品の使用環境、使用頻度および保存環境により記載の寿命より交換時期が早くなる場合がありますので、ご了承下さい。  
なお、ユーザによる交換はできません。交換が必要な場合は、当社または代理店へご連絡下さい。

製品ごとに個別の寿命部品を使用している場合があります。  
本書、寿命部品に関する記載項を参照して下さい。

主な寿命部品と寿命

部品名称	寿命
ユニット電源	5年
ファン・モータ	5年
電解コンデンサ	5年
液晶ディスプレイ	6年
液晶ディスプレイ用バックライト	2.5年
フロッピー・ディスク・ドライブ	5年
メモリ・バックアップ用電池	5年

■ハード・ディスク搭載製品について

使用上の留意事項を以下に示します。

- 本器は、電源が入った状態で持ち運んだり、衝撃や振動を与えないで下さい。  
ハード・ディスクの内部は、情報を記録するディスクが高速に回転しながら、情報の読み書きを行っているため、非常にデリケートです。
- 本器は、以下の条件に合う場所で使用および保管をして下さい。  
極端な温度変化のない場所  
衝撃や振動のない場所  
湿気や埃・粉塵の少ない場所  
磁石や強い磁界の発生する装置から離れた場所
- 重要なデータは、必ずバックアップを取っておいて下さい。  
取扱方法によっては、ディスク内のデータが破壊される場合があります。また、使用条件によりますが、ハード・ディスクには、その構造上、寿命があります。  
なお、消失したデータ等の保証は、いたしかねますのでご了承下さい。

■本器の廃棄時の注意

製品を廃棄する場合、有害物質は、その国の法律に従って適正に処理して下さい。

- 有害物質： (1) PCB (ポリ塩化ビフェニール)  
(2) 水銀  
(3) Ni-Cd (ニッケル-カドミウム)  
(4) その他

シアン、有機リン、六価クロムを有する物およびカドミウム、鉛、砒素を溶出する恐れのある物（半田付けの鉛は除く）

例： 蛍光管、バッテリー

■使用環境

本器は、以下の条件に合う場所に設置して下さい。

- 腐食性ガスの発生しない場所
- 直射日光の当たらない場所
- 埃の少ない場所
- 振動のない場所
- 最大高度 2000 m

本器を安全に取り扱うための注意事項

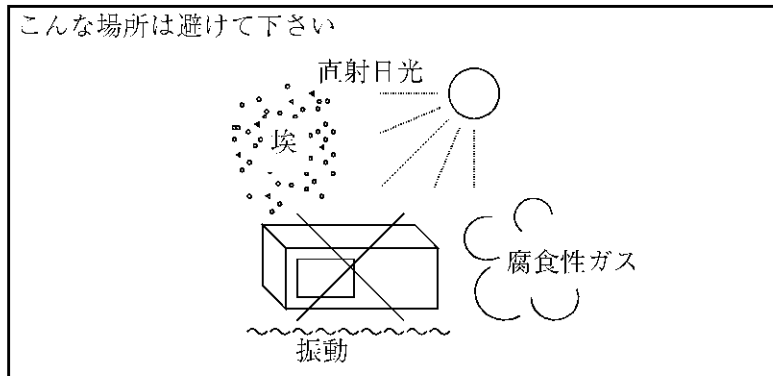


図-1 使用環境

●設置姿勢

本器は、必ず水平状態で使用して下さい。  
本器は内部温度上昇をおさえるため、強制空冷用のファンを搭載しております。  
ファンの吐き出し口、通気孔をふさがらないで下さい。

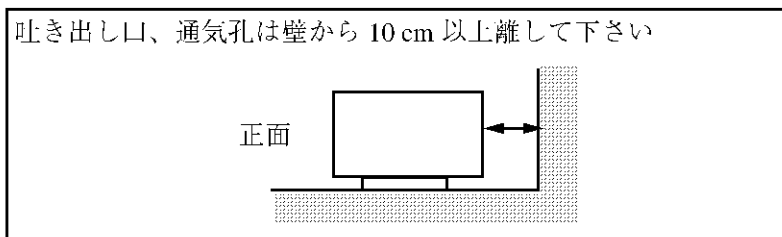


図-2 設置

●保管姿勢

本器は、なるべく水平状態で保管して下さい。  
本器を立てた状態で保管する場合、または運搬時、一時的に立てた状態で置く場合、  
転倒しないよう注意して下さい。衝撃・振動により転倒する恐れがあります。

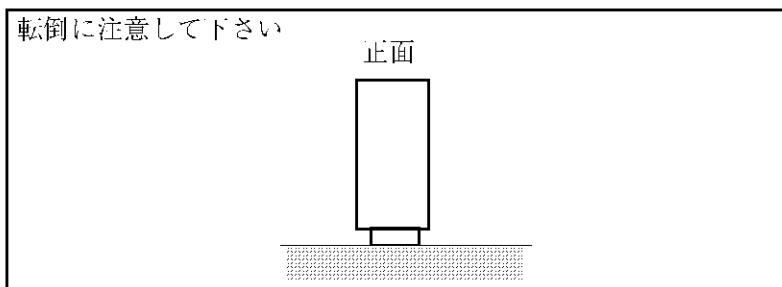
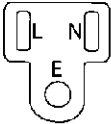
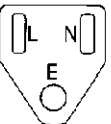
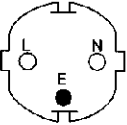
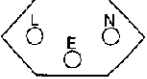

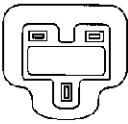
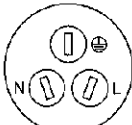


図-3 保管

- IEC61010-1 で定義される、主電源に典型的に存在する過渡過電圧および汚染度の分類は、以下のとおりです。  
IEC60364-4-443 の耐インパルス（過電圧）カテゴリ II  
汚染度 2

■電源ケーブルの種類

「電源ケーブルの種類」の記述が本文中にある場合には、以下の表に置き替えてお読み下さい。

プラグ	適用規格	定格・色・長さ	型名 (オプション No.)
	PSE: 日本 電気用品安全法	125V/7A 黒、2m	ストレート・タイプ A01402 アングル・タイプ A01412
	UL: アメリカ CSA: カナダ	125V/7A 黒、2m	ストレート・タイプ A01403 (オプション 95) アングル・タイプ A01413
	CEE: ヨーロッパ DEMKO: デンマーク NEMKO: ノルウェー VDE: ドイツ KEMA: オランダ CEBEC: ベルギー OVE: オーストリア FIMKO: フィンランド SEMKO: スウェーデン	250V/6A 灰、2m	ストレート・タイプ A01404 (オプション 96) アングル・タイプ A01414
	SEV: スイス	250V/6A 灰、2m	ストレート・タイプ A01405 (オプション 97) アングル・タイプ A01415
	SAA: オーストラリア ニュージーランド	250V/6A 灰、2m	ストレート・タイプ A01406 (オプション 98) アングル・タイプ ---
	BS: イギリス	250V/6A 黒、2m	ストレート・タイプ A01407 (オプション 99) アングル・タイプ A01417
	CCC: 中国	250V/10A 黒、2m	ストレート・タイプ A114009 (オプション 94) アングル・タイプ A114109





## 緒言

本書は、D3371 トランスミッション・アナライザのリモート・プログラミングについて説明してあります。本器は、本書と以下の2冊で構成されています。使用目的に合わせてご使用下さい。

### D3371 トランスミッション・アナライザ取扱説明書

本器の操作方法、各部の名称、機能説明などを説明しています。本器を安全に使用するため、使用開始の前に必ずお読み下さい。(別冊)

#### 1. 商標について

Microsoft Windows98 は、米国マイクロソフト株式会社の登録商標です。

#### 2. 本書の構成

本書の章構成は、以下のとおりです。

本器を安全に取り扱うための注意事項	本器を安全に使用するため、使用開始の前に必ずお読み下さい。
1. はじめに	GPIB インタフェースの概要、接続方法を示します。
2. GPIB バスの機能	本器の GPIB バス機能について説明します。
3. コマンド文法	コマンド文法について説明します。
4. ステータス・バイト	ステータス・バイトの動作モデルとイベントの割り当てを説明します。
5. コマンド・リファレンス	はじめに、サブシステムの構成やコマンドの記述上の注意を説明します。そして、コマンドの詳細を説明します。
6. サンプル・プログラム	本器の GPIB を使用してリモート・コントロールをする例を記述します。
付録 A.1 コマンド一覧	コマンド一覧を示します。
付録 A.2 設定メニューに対応するコマンド一覧	設定メニューに対応するコマンド一覧を示します。
付録 A.3 初期設定	初期設定を示します。



# 目次

<b>1.</b>	<b>はじめに</b> .....	1-1
1.1	GPIB とは .....	1-1
1.2	コマンド・モード .....	1-2
1.3	GPIB のセット・アップ .....	1-3
1.3.1	GPIB の接続 .....	1-3
1.3.2	GPIB アドレスの設定 .....	1-4
<b>2.</b>	<b>GPIB バスの機能</b> .....	2-1
2.1	GPIB インタフェース機能 .....	2-1
2.2	インタフェース・メッセージに対する応答 .....	2-2
2.2.1	インタフェース・クリア (IFC) .....	2-2
2.2.2	リモート・イネーブル (REN) .....	2-2
2.2.3	シリアル・ポール・イネーブル (SPE) .....	2-2
2.2.4	デバイス・クリア (DCL) .....	2-3
2.2.5	セレクトッド・デバイス・クリア (SDC) .....	2-3
2.2.6	ゴー・トゥ・ローカル (GTL) .....	2-3
2.2.7	ローカル・ロック・アウト (LLO) .....	2-3
2.3	メッセージ交換プロトコル .....	2-4
2.3.1	GPIB 各種バッファ .....	2-4
2.3.2	IEEE488.2-1987 コマンド・モード .....	2-5
<b>3.</b>	<b>コマンド文法</b> .....	3-1
3.1	コマンド文法 .....	3-1
3.2	データ・フォーマット .....	3-3
3.3	デリミタ (ターミネータ) .....	3-5
<b>4.</b>	<b>ステータス・バイト</b> .....	4-1
4.1	ステータス・レジスタ .....	4-1
4.1.1	ステータス・レジスタの構造 .....	4-1
4.1.2	ステータス・レジスタの種類 .....	4-2
4.2	ステータス・バイト・レジスタ .....	4-5
4.3	標準イベント・ステータス・レジスタ .....	4-7
4.4	オペレーション・ステータス・レジスタ .....	4-8
4.5	クエスチョナブル・ステータス・レジスタ .....	4-9
4.6	セトリング・ステータス・レジスタ .....	4-9
4.7	クロックロス・ステータス・レジスタ .....	4-9
4.8	デバイス・ステータス・レジスタ .....	4-10
4.9	ステータス・レジスタのクリア、リセット方法 .....	4-11
<b>5.</b>	<b>コマンド・リファレンス</b> .....	5-1
5.1	GPIB コマンド・インデックス .....	5-1
5.2	コマンド解説の前に .....	5-10
5.2.1	サブシステムの構成 .....	5-10
5.2.2	コマンド記述上の注意 .....	5-12
5.2.3	D3371 とコントローラの同期 .....	5-13
5.2.4	システム・バージョンによるコマンド差異について .....	5-13

## 目次

5.3	共通コマンド .....	5-14
5.4	PPG モジュール .....	5-19
5.4.1	SOURce[1] サブシステム .....	5-19
5.4.2	OUTPut[1] サブシステム .....	5-37
5.4.3	OUTPut3 サブシステム .....	5-40
5.4.4	SOURce2 サブシステム .....	5-44
5.4.5	OUTPut2 サブシステム .....	5-45
5.4.6	OUTPut4 サブシステム .....	5-49
5.4.7	SOURce5 サブシステム .....	5-53
5.4.8	SOURce6 サブシステム .....	5-55
5.4.9	SOURce7 サブシステム .....	5-59
5.5	ED モジュール .....	5-60
5.5.1	SENSe[1] サブシステム .....	5-60
5.5.2	INPut[1] サブシステム .....	5-97
5.5.3	INPut2 サブシステム .....	5-100
5.5.4	SOURce8 サブシステム .....	5-103
5.5.5	SOURce9 サブシステム .....	5-104
5.6	結果、システム・コマンド .....	5-105
5.6.1	FETCh/CFETch サブシステム .....	5-105
5.6.2	DISPlay サブシステム .....	5-128
5.6.3	HCOPy サブシステム .....	5-132
5.6.4	MMEMory サブシステム .....	5-133
5.6.5	SYSTem サブシステム .....	5-143
5.6.6	STATus サブシステム .....	5-146
5.7	SSG モジュール .....	5-155
5.7.1	SOURce10 サブシステム .....	5-155
5.8	ジッタ耐力測定 .....	5-159
5.8.1	JTOLerance サブシステム .....	5-159
<b>6.</b>	<b>サンプル・プログラム .....</b>	<b>6-1</b>
6.1	測定条件の設定および読み込みのプログラム例 .....	6-1
6.2	測定プログラム例 .....	6-4
<b>付録</b>	<b>.....</b>	<b>A-1</b>
A.1	コマンド一覧 .....	A-1
A.1.1	サブシステムと入出力インタフェース .....	A-1
A.1.1.1	PPG モジュールの入出力とサブシステム .....	A-1
A.1.1.2	ED モジュールの入出力とサブシステム .....	A-2
A.1.1.3	SSG モジュールの入出力とサブシステム .....	A-2
A.1.2	共通コマンド .....	A-3
A.1.3	PPG モジュール .....	A-4
A.1.3.1	SOURce[1] サブシステム .....	A-4
A.1.3.2	OUTPut[1] サブシステム .....	A-5
A.1.3.3	OUTPut3 サブシステム .....	A-6
A.1.3.4	SOURce2 サブシステム .....	A-7
A.1.3.5	OUTPut2 サブシステム .....	A-7
A.1.3.6	OUTPut4 サブシステム .....	A-7
A.1.3.7	SOURce5 サブシステム .....	A-8
A.1.3.8	SOURce6 サブシステム .....	A-8
A.1.3.9	SOURce7 サブシステム .....	A-9

A.1.4	ED モジュール .....	A-10
A.1.4.1	SENSe[1] サブシステム .....	A-10
A.1.4.2	INPut[1] サブシステム .....	A-14
A.1.4.3	INPut2 サブシステム .....	A-15
A.1.4.4	SOURce8 サブシステム .....	A-15
A.1.4.5	SOURce9 サブシステム .....	A-15
A.1.5	結果、システム・コマンド .....	A-16
A.1.5.1	FETCh/CFETch サブシステム .....	A-16
A.1.5.2	DISPlay サブシステム .....	A-19
A.1.5.3	HCOPy サブシステム .....	A-19
A.1.5.4	MMEMory サブシステム .....	A-20
A.1.5.5	SYSTem サブシステム .....	A-21
A.1.5.6	STATus サブシステム .....	A-22
A.1.6	SSG モジュール .....	A-23
A.1.6.1	SOURce10 サブシステム .....	A-23
A.1.7	ジッタ耐力測定 .....	A-24
A.1.7.1	JTOLerance サブシステム .....	A-24
A.2	設定メニューに対応するコマンド一覧 .....	A-25
A.2.1	メニュー .....	A-25
A.2.2	[Settings] ダイアログ・ボックス .....	A-27
A.2.2.1	SSG モジュール .....	A-27
A.2.2.2	PPG モジュール .....	A-27
A.2.2.3	ED モジュール .....	A-31
A.2.2.4	System モジュール .....	A-37
A.3	初期設定 .....	A-38
A.3.1	PPG モジュール .....	A-38
A.3.2	ED モジュール .....	A-42
A.3.3	結果、システム .....	A-46
A.3.4	SSG モジュール .....	A-46
A.3.5	STM パターン .....	A-47
A.3.5.1	オーバヘッド .....	A-47
A.3.5.2	ペイロード .....	A-47
A.3.6	FLEX パターン .....	A-48
A.3.6.1	パターン・シーケンス・テーブル .....	A-48
A.3.6.2	PROG パターン .....	A-48
A.3.7	ジッタ耐力パラメータ・テーブル .....	A-48
索引	.....	I-1



## 図一覧

図番号	名 称	ページ
1-1	GPIB ケーブルの接続 .....	1-3
4-1	ステータス・レジスタの構造 .....	4-1
4-2	ステータス・レジスタの種類と配置 .....	4-3
4-3	ステータス・レジスタの配置概要 .....	4-4
4-4	ステータス・バイト・レジスタの構造 .....	4-5





## 表一覧

表番号	名 称	ページ
5-1	システム・バージョンによるコマンド差異 .....	5-13



## 1. はじめに

本器は、IEEE 規格 488.1-1987 および 488.2-1987 に準拠した GPIB(General Purpose Interface Bus) インタフェースを標準装備し、外部コントローラによるリモート・コントロールが可能です。

この章では、GPIB インタフェースの概要、接続方法を示します。

### 1.1 GPIB とは

GPIB は、コンピュータと測定器を統合する高性能のバスを提供します。

この GPIB の動作は IEEE 規格 488.1-1987 によって定義されています。GPIB はバス構造のインタフェースのため、各機器が固有の互いに異なる機器アドレスを持つことによって、特定の機器を指定します。これらの機器は 1 つのバスに 15 台まで並列に接続できます。GPIB 機器は、以下の機能のうち 1 つ以上を備えています。

- トーカ： バスにデータを送信するために指定された機器を「トーカ」と呼びます。GPIB バス上では、一台の機器のみがアクティブ・トーカとして動作します。
- リスナ： バスのデータを受信するために指定された機器を「リスナ」と呼びます。アクティブなリスナ機器は GPIB バス上に複数存在することができます。
- コントローラ： トーカ、リスナを指定する機器を「コントローラ」と呼びます。GPIB バス上では一台の機器のみがアクティブ・コントローラとして動作します。これらのコントローラのうち、IFC、および REN のメッセージをコントロールできる機器を特に「システム・コントローラ」と呼びます。  
システム・コントローラは、GPIB バス上に一台だけ許されます。バス上に複数のコントローラがある場合、システム起動時にはシステム・コントローラがアクティブ・コントローラとなり、その他のコントローラ能力を持つ機器はアドレスサブル機器として動作します。  
その他のコントローラをアクティブ・コントローラにするには Take Control(TCT) インタフェース・メッセージを用います。そのとき自分はノンアクティブ・コントローラとなります。  
コントローラはインタフェース・メッセージ、またはデバイス・メッセージを各測定器に送ってシステム全体をコントロールします。  
それぞれ以下の役目を果たします。
  - インタフェース・メッセージ： GPIB バスをコントロールします。
  - デバイス・メッセージ： 測定器をコントロールします。

## 1.2 コマンド・モード

本器は、IEEE 規格 488.2-1987 コマンド・モードで動作します。  
この 488.2-1987 は、以下に示すものを 488.1-1987 に拡張して定義しています。

- 測定器をプログラミングする際の文法
- コマンドとデータの通信プロトコル (手順)
- 共通コマンド\*
- ステータスデータ構造
- システムの同期プロトコル

\*: 共通コマンドとは、すべての測定器で同じ動作をするコマンドのことです。

## 1.3 GPIB のセット・アップ

GPIB インタフェースを使用するには、以下のセットアップが必要です。

- GPIB の接続
- GPIB アドレスの設定

### 1.3.1 GPIB の接続

以下に標準的な GPIB の接続を示します。GPIB コネクタは 2 本のねじでしっかり固定して、使用中にゆるむことがないように注意して下さい。

---

注意 本器へ GPIB ケーブルを接続する場合、必ず電源を切った状態で行って下さい。

---

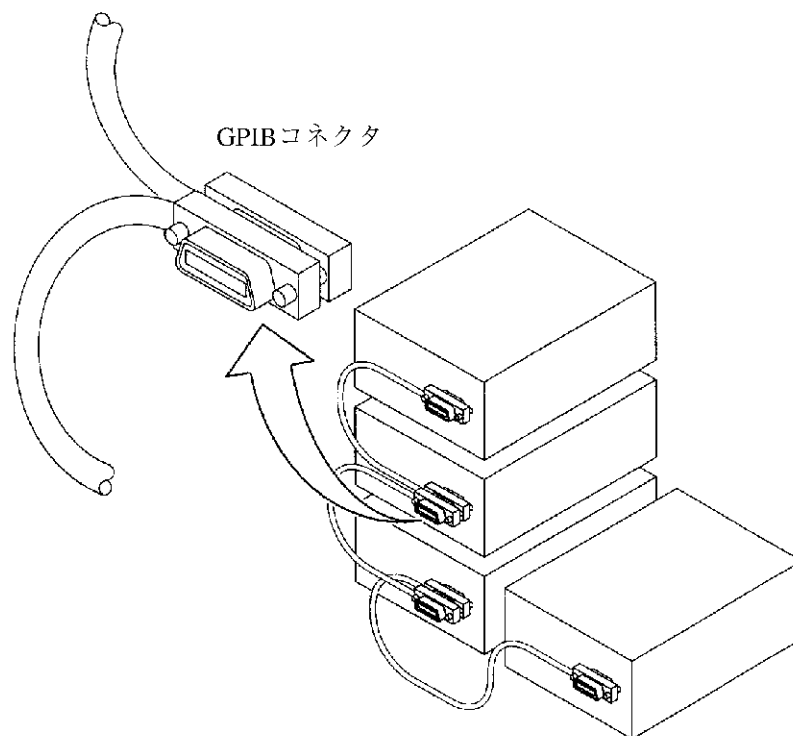


図 1-1 GPIB ケーブルの接続

### 1.3.2 GPIB アドレスの設定

GPIB インタフェースの使用時には、以下のことに注意して下さい。

- 1つのバス・システムで使われる GPIB ケーブルの全ケーブル長は、 $2\text{m} \times$  {接続される機器の数 (GPIB コントローラも1つの機器をして数える)} 以下です。また、ケーブルの全ケーブル長は 20m 以下とします。
- 1つのバス・システムに接続できる機器の数は、最高 15 台です。
- ケーブル間の接続方法には制限はありません。ただし、1 台の機器上に 4 個以上の GPIB コネクタを重ねないで下さい。4 個以上重ねるとコネクタの取り付け部に過度の力が加わり、破損することがあります。

(例) 5 台の機器から構成されるシステムで使用できる全ケーブル長は、10m 以下 (5 台  $\times$  2m / 台 = 10m) です。全ケーブル長が許容最大長を超えない範囲で、自由に分配することができます。

ただし、10 台以上の機器を接続する場合は、何台かの機器を 2m 以下のケーブルで接続して、全ケーブル長が 20m を超えないようにする必要があります。

### 1.3.2 GPIB アドレスの設定

本器の GPIB アドレスの設定方法を示します。操作の詳細は、D3371 トランスミッション・アナライザ取扱説明書を参照して下さい。

1. 電源を投入します。  
本器の測定アプリケーションが起動されます。
2. ローカル状態であることを確認します。  
ローカル状態は、正面パネル左側にある **REMOTE** ランプの消灯により確認します。  
ローカル状態でない場合は、正面パネル左側にある **LOCAL** キーを押し、ローカル状態にします。
3. 標準ツール・バーの [**Settings**] ボタンをクリックします。  
Settings ウィンドウが表示されます。
4. モジュール選択リスト・バーの [**System**] をクリックします。
5. [**Remote Interface**] ドロップ・ダウン・リスト・ボックスの [**GPIB**] を選択します。
6. [**Address**] テキスト・ボックスに、設定する GPIB アドレスを入力します。
7. [**OK**] ボタンを押します。  
Settings ウィンドウが閉じ、GPIB アドレスが設定されます。

## 2. GPIB バスの機能

この章では、本器の GPIB バス機能について説明します。

### 2.1 GPIB インタフェース機能

コード	説明
SH1	ソース・ハンドシェーク機能あり
AH1	アクセプタ・ハンドシェーク機能あり
T6	基本的トーカー機能、シリアル・ポール機能、リスナ指定によるトーカー解除機能
TE0	拡張トーカー機能なし
L4	基本的リスナ機能、トーカー指定によるリスナ解除機能
LE0	拡張リスナ機能なし
SR1	サービス・リクエスト機能あり
RL1	リモート機能、ローカル機能、ローカル・ロック・アウト機能
PP0	パラレル・ポール機能なし
DC1	デバイス・クリア機能
DT0	デバイス・トリガ機能なし
C0	コントローラ機能なし
E1	オープン・コレクタ・バス・ドライバを使用

---

## 2.2 インタフェース・メッセージに対する応答

### 2.2 インタフェース・メッセージに対する応答

本器のインタフェース・メッセージに対する応答を示します。インタフェース・メッセージに対する応答は、IEEE 規格 488.1-1987 および 488.2-1987 で定義されています。

インタフェース・メッセージの本器への送り方は、使用するコントローラの取扱説明書を参照して下さい。

#### 2.2.1 インタフェース・クリア (IFC)

このメッセージは、本器へ直接信号線で送られてきます。

このメッセージによって本器は GPIB バスの動作を停止します。すべての入／出力を停止しますが、入出力バッファはクリアされません (クリアは DCL により実行されます)。

#### 2.2.2 リモート・イネーブル (REN)

このメッセージは、本器へ直接信号線で送られてきます。このメッセージが真のとき、本器がリスナに指定されるとリモート状態になります。この状態は GTL を受けとるか、REN が偽になるか、または LOCAL キーを押すまで続きます。

本器は、ローカル状態のとき、すべての受信データを無視します。

リモート状態のとき、D3371 Transmission Analyzer アプリケーションは LOCAL キーを除くすべてのキー入力を無視します。

ローカル・ロック・アウト状態 (LLO:「2.2.7 項」を参照) のとき、すべてのキー入力を無視します。

#### 2.2.3 シリアル・ポール・イネーブル (SPE)

本器はこのメッセージを外部から受信すると、シリアル・ポール・モードになります。

このモードでは、トーカーに指定されると通常メッセージではなくステータス・バイトを送信します。このモードはシリアル・ポール・ディセーブル (SPD) メッセージを受信するか、IFC メッセージを受信するまで続きます。

本器がサービス・リクエスト (SRQ) メッセージをコントローラに送信しているときには、応答データの bit6 (RQS bit) が 1 (TRUE) になります。送信が終了後、RQS bit は 0 (FALSE) になります。サービス・リクエスト (SRQ) メッセージは、直接信号線で送ります。



## 2.2.4 デバイス・クリア (DCL)

本器は DCL を受け取ったときに、以下のことを実行します。

- 入力バッファと出力バッファのクリア
- 構文解析部、実行コントロール部、応答データ生成部のリセット
- 次に実行するリモート・コマンドを妨げる全コマンドのキャンセル
- 他のパラメータを待つため一時停止されているコマンドのキャンセル
- OPC と OPC? のキャンセル

以下のことは実行しません。

- 本器に設定または格納されているデータの変更
- 正面パネル操作の中断
- 実行中の本器の動作への影響や中断
- MAV を除くステータス・バイトの変更 (MAV は出力バッファのクリアの結果として 0 になります)

## 2.2.5 セレクトッド・デバイス・クリア (SDC)

DCL と同一の動作を行います。ただし、SDC は本器がリスナのときだけ実行されます。その他の場合は無視されます。

## 2.2.6 ゴー・トゥ・ローカル (GTL)

このメッセージは、本器をローカル状態にします。ローカル状態になると、正面パネル操作がすべて有効になります。

## 2.2.7 ローカル・ロック・アウト (LLO)

このメッセージは、本器をローカル・ロック・アウト状態にします。この状態で本器がリモート状態になると、正面パネル操作はすべて禁止されます (通常のリモート状態では、LOCAL キーで正面パネル操作ができます)。

このとき本器をローカル状態にする方法は、以下の 2 とおりあります。

- GTL メッセージを本器に送る
- REN メッセージを偽にする (このときローカル・ロック・アウト状態も解除します)

## 2.3 メッセージ交換プロトコル

本器は、コントローラやその他の機器から GPIB バスを通じてプログラム・メッセージを受け取り、応答データを発生します。プログラム・メッセージには、コマンド、クエリ（応答データを問い合わせるコマンドのことを、特に「クエリ」と呼ぶ）、データが含まれています。それらのデータのやり取りには手順があります。ここではその手順について説明します。

### 2.3.1 GPIB 各種バッファ

本器には、以下の3つのバッファがあります。

- 入力バッファ  
コマンド解析のために一時的にデータを貯めておくバッファです。  
(1024 バイトの長さをもちます)  
入力バッファのクリア方法は、2 とおりあります。
  - 電源投入
  - DCL または SDC の実行
- 出力バッファ  
コントローラからデータを読まれるまでデータを貯めておくバッファです。  
(1024 バイトの長さをもちます)  
出力バッファのクリア方法は、2 とおりあります。
  - 電源投入
  - DCL または SDC の実行
- エラー・キュー  
これはリモート・コマンドのエラー・メッセージを蓄えておくキューです。  
(10 の深さをもちます)  
リモート・コマンドの解析／実行でエラーが発生する毎に、メッセージをキューにつまみす。  
エラー・メッセージは、SYST:ERR? クエリで読み出すことができます。1 つ読み出すと、キューから1つメッセージを削除します。  
エラー・キューのクリア方法は、2 とおりあります。
  - 電源投入
  - \*CLS の実行

### 2.3.2 IEEE488.2-1987 コマンド・モード

IEEE488.2-1987 コマンド・モードは、IEEE 規格 488.2-1987 に適合したメッセージ交換プロトコルに従ってメッセージの送受信を行います。  
他のコントローラや機器がメッセージを本器から受信するときに、以下の点に注意して下さい。

- クエリの受信によって応答データが生成される
- クエリを実行した順にデータが生成される

パーサーでは、入力バッファから受信した順序通りにコマンド・メッセージを受け取り、構文解析を実行し、受け取ったコマンドがどんな内容で実行するのかを決定します。

コマンドの構文解析時にコマンドの木構造の追跡も行っています。  
木構造のどの部分から解析すべきなのかを次のコマンド解析のために覚えています。  
この情報はパーサーがクリアされると木構造の頭まで戻ります。

パーサーのクリア方法は、4 とおりあります。

- 電源投入
- DCL または SDC の受信
- ‘;’ の次の ‘?’ の受信
- ターミネータまたは EOI の受信

パーサーがクエリを実行すると、その応答としてデータを出力バッファ上に生成します（つまり、本器がデータを出力するにはその直前に必ずクエリを送る必要があります）。  
これはクエリで生成されるデータは、コントローラがリードしなければデータがクリアされないことを意味します。

コントローラのリード以外でデータがクリアされる条件は 2 とおりあり、これらの状態では Query Error を発生します。

- **Unterminated condition:**  
クエリをターミネート（ASCII の LF コードまたは GPIB の END メッセージ）せずにコントローラが応答データをリードしたか、クエリを送らずにコントローラが応答データをリードした場合
- **Interrupted condition:**  
コントローラが応答データをリードする前に、本器が次のプログラム・メッセージを受け取った場合



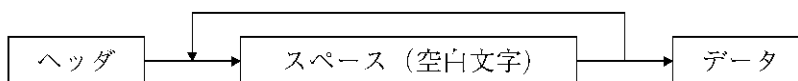
### 3. コマンド文法

コマンドを入力する文字は、文字列データとブロック・データを除き、英字の大文字・小文字の区別はありません。

この章では、コマンド文法について説明します。

#### 3.1 コマンド文法

コマンド文法は、以下のフォーマットで定義されています。



##### 1. ヘッダ

ヘッダは、コロン (;) で区切られた複数のニーモニックからなる階層構造を持ちます。4文字以上からなるニーモニックは、4文字 (または3文字) の「ショート・フォーム」を持ちます (省略しないニーモニックを「ロング・フォーム」と呼びます)。

どちらのフォームをどのように組み合わせても構いません。

ヘッダの直後に ? を付けるとクエリ・コマンドになります。

##### 2. スペース (空白文字)

1文字分以上のスペースが可能です。スペース以外ではエラーとなります。

##### 3. データ

コマンドが複数のデータを必要とするときは、データをカンマ (,) で区切って複数並べます。カンマ (,) の前後にスペース (空白文字) を入れても構いません。

データ・タイプの詳細については、「3.2 データ・フォーマット」を参照して下さい。

##### 4. 複数のコマンドの記述

複数のコマンドをセミコロン (;) で区切って1行で記述することが可能です。

このようにコマンドを記述した場合には、ヘッダの持つ階層構造の中でカレント・パスを移動しながらコマンドを実行していきます。

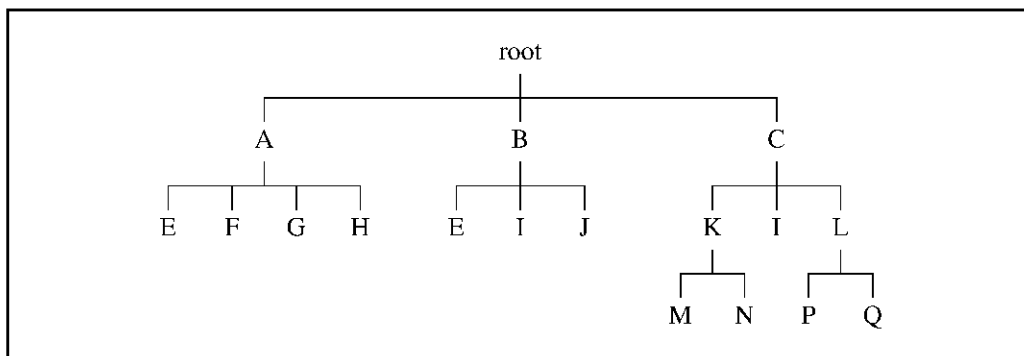
## 3.1 コマンド文法

## 5. カレント・パスの移動

以下の規則にしたがって、カレント・パスは移動します。

- 電源投入時: カレント・パスは root にセットされる
- ターミネータ: カレント・パスは root にセットされる
- コロン (:): カレント・パスをコマンド・ツリーの中で1階層下に移動する  
コロン (:) がコマンドの先頭文字の場合、コロン (:) はカレント・パスを root にする
- セミコロン (;): カレント・パスを変更しない
- 共通コマンド: カレント・パスに関係なく実行できる  
\*RST コマンドを実行するとカレント・パスは root にセットされる

(例) 以下のヘッダ構造とします。



このとき、以下のカレント・パス動作になります。

- :A:E;;B:E  
2つ目のコマンドの: はカレント・パスを root に移動するので、A:E と B:E はどちらも正しいコマンドです。
- : A:E<END>B:E  
<END> (ターミネータ) はカレント・パスを root に移動するので、A:E と B:E はどちらも正しいコマンドです。
- :A:E;F;G;H  
: はカレント・パスを移動しないので、:A:E;F;G;H は結果的に A:E、A:F、A:G、A:H の4つのコマンドと等しくなります。
- :C:I;K:N;M  
: がカレント・パスを移動するので、K:N は :C: の階層から見ることになります。したがって K:N は C:K:N となります。また同時に、K:N は : を含むためカレント・パスを :C:K に変更し、最後の M は C:K:M と解釈されます。
- :A:E;\*ESE 16  
共通コマンドはカレント・パスに関係ないので、\*ESE 16 は正しく実行されます。

- :A:E;\*ESE 16;F;G;H

共通コマンドはカレント・パスを変更しないので、3つ目の F は 1つ目の :A:E で設定されたカレント・パスの :A: で探されます。したがって、F は A:F、G は A:G、H は A:H になります。

以下の例では、文法エラーとなります。

- :A:E;B:E

A:E はカレント・パスを :A: に変更しています。

したがって、B:E は :A: の階層で探されるが、B というニーモニックが見つからないのでエラーとなります。

- :C:K:M;L:P

:C:K:M はカレント・パスを :C:K: に変更しています。

したがって、L:P は :C:K: で探されるが、L というニーモニックが見つからないのでエラーとなります。

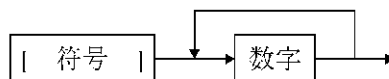
## 3.2 データ・フォーマット

本器は、以下に示すデータ・タイプをデータの入出力で使用します。

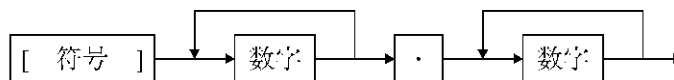
### 1. 数値データ

数値データには以下の3つのフォーマットがあります。本器に対する数値の入力では、どれを用いても構いません（入力するデータの型に応じて四捨五入されます）。

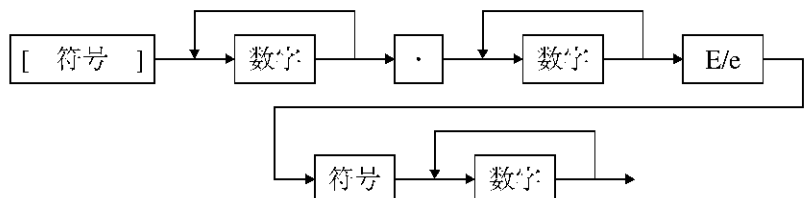
- 整数型： NR1 フォーマット (注) [] で囲んだ符号は省略可能です。



- 固定小数点型： NR2 フォーマット



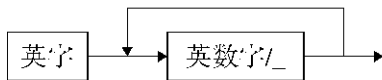
- 浮動小数点型： NR3 フォーマット



3.2 データ・フォーマット

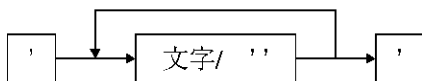
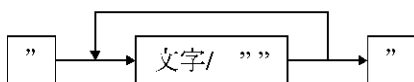
2. 文字データ

文字データのフォーマットを以下に示します。



3. 文字列データ

文字列データには、2つのフォーマットがあります。



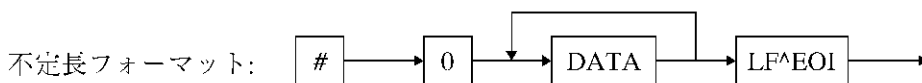
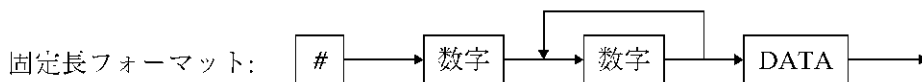
文字列データ中では、ASCII 7bit コード文字として使用できます。

注 ”で始まる文字列データ中では”を””で表現しなければなりません。  
 ’で始まる文字列データ中では’を’’で表現しなければなりません。

応答データが文字列データの場合、”で始まる文字列データを必ず出力します。

4. ブロック・データ

ブロック・データには、2つのフォーマットがあります。本器への入力時には、どちらのフォーマットを用いても構いません。



固定長のフォーマットでは、#の後の1文字の数字でその後に続くデータのバイト数の桁数を表します。0は使えません(不定長になります)。

(例) #3128 <data byte> というブロック・データの場合

#の後の3がその後に続く文字列(128)の桁数を表し、128はその後に続く<data byte>のバイト数を表します。



### 3.3 デリミタ (ターミネータ)

1. コマンド入力時  
GPIB コマンドを読み込む際には、常に LF または EOI によって終端されます。
2. クエリ応答時  
クエリの結果を出力する際には、常に LF 出力と同時に EOI が出力されます。

---

注 LF = ライン・フィード (16 進 0A)  
EOI = 単線信号 (End Or Identify)

---



## 4. ステータス・バイト

本器は、IEEE 規格 488.2-1987 に適合した階層化されたステータス・レジスタ構造をもち、機器の様々な状態をコントローラへ送信できます。

この章では、ステータス・バイトの動作モデルと、イベントの割り当てを説明します。

### 4.1 ステータス・レジスタ

#### 4.1.1 ステータス・レジスタの構造

本器は、IEEE 規格 488.2-1987 で定義されたステータス・レジスタのモデルを採用しており、コンディション・レジスタ、トランジション・フィルタ、イベント・レジスタ、イネーブル・レジスタから構成されています。

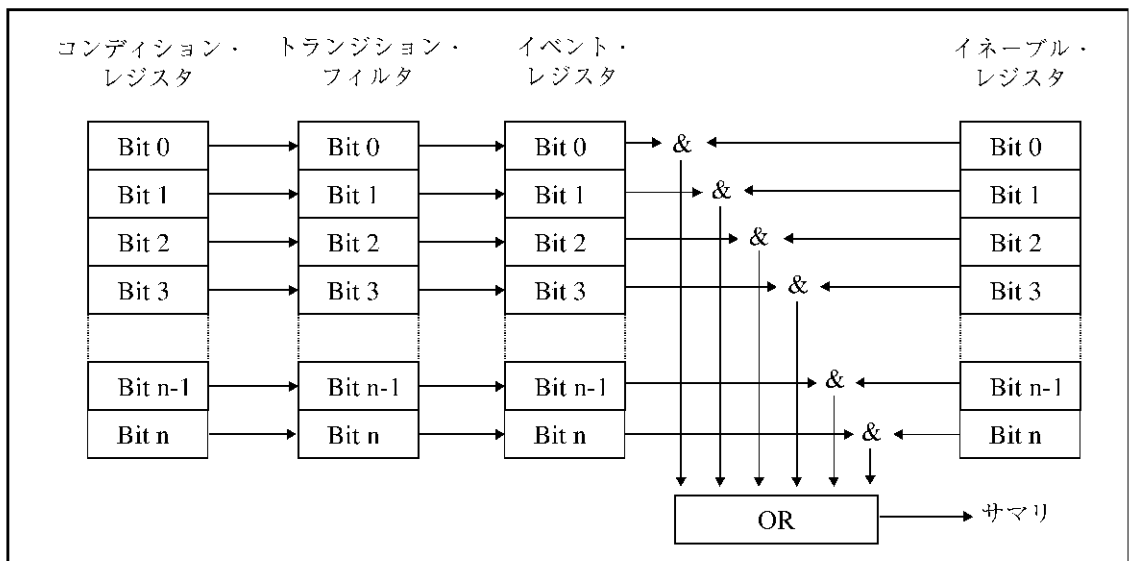


図 4-1 ステータス・レジスタの構造

#### 1. コンディション・レジスタ

コンディション・レジスタは、コンディション・ビットごとに機器の最新の状態をモニタしています。このレジスタは状態を記憶せず、リアルタイムで変化します。bit に対応する事象が発生しているときに、その bit が 1 にセットされます。コンディション・レジスタにデータを書き込むことはできません。

#### 2. トランジション・フィルタ

トランジション・フィルタは、コンディション・ビットが正方向、負方向のどちらの変化があった場合に、イベント・ビットを 1 にセットするかを指定します。イベント・ビットを 1 にセットしたい事象に対応する bit を 1 にセットします。トランジション・フィルタはデータを書き込めます。

#### 4.1.2 ステータス・レジスタの種類

##### 3. イベント・レジスタ

イベント・レジスタは、トランジション・フィルタで指定した変化がコンディション・レジスタで発生した場合、対応するイベント・ビットが1にセットされます。コンディション・レジスタ、トランジション・フィルタが存在しないか無効の場合は、対応する事象が発生したときにイベント・ビットが1にセットされます。このレジスタは、クエリで読み出すか \*CLS を実行するとクリアされます。イベント・レジスタにデータを書き込むことはできません。

##### 4. イネーブル・レジスタ

イネーブル・レジスタは、イベント・レジスタのどのイベント・ビットを有効なステータスとしてサマリを生成するのか指定します。イネーブル・レジスタはイベント・レジスタと bit ごとに AND をとられ、それらの OR がサマリとして生成されます。サマリは次のステータス・レジスタに書き込まれます。サマリを生成したいイベント・ビットに対応する bit を1にセットします。イネーブル・レジスタはデータを書き込めます。

#### 4.1.2 ステータス・レジスタの種類

本器のステータス・レジスタは、以下の7種類があります。

- |                         |          |
|-------------------------|----------|
| 1. ステータス・バイト・レジスタ:      | 4.2 節を参照 |
| 2. 標準イベント・ステータス・レジスタ:   | 4.3 節を参照 |
| 3. オペレーション・ステータス・レジスタ:  | 4.4 節を参照 |
| 4. クエスチョナブル・ステータス・レジスタ: | 4.5 節を参照 |
| 5. セトリング・ステータス・レジスタ:    | 4.6 節を参照 |
| 6. クロックロス・ステータス・レジスタ:   | 4.7 節を参照 |
| 7. デバイス・ステータス・レジスタ:     | 4.8 節を参照 |

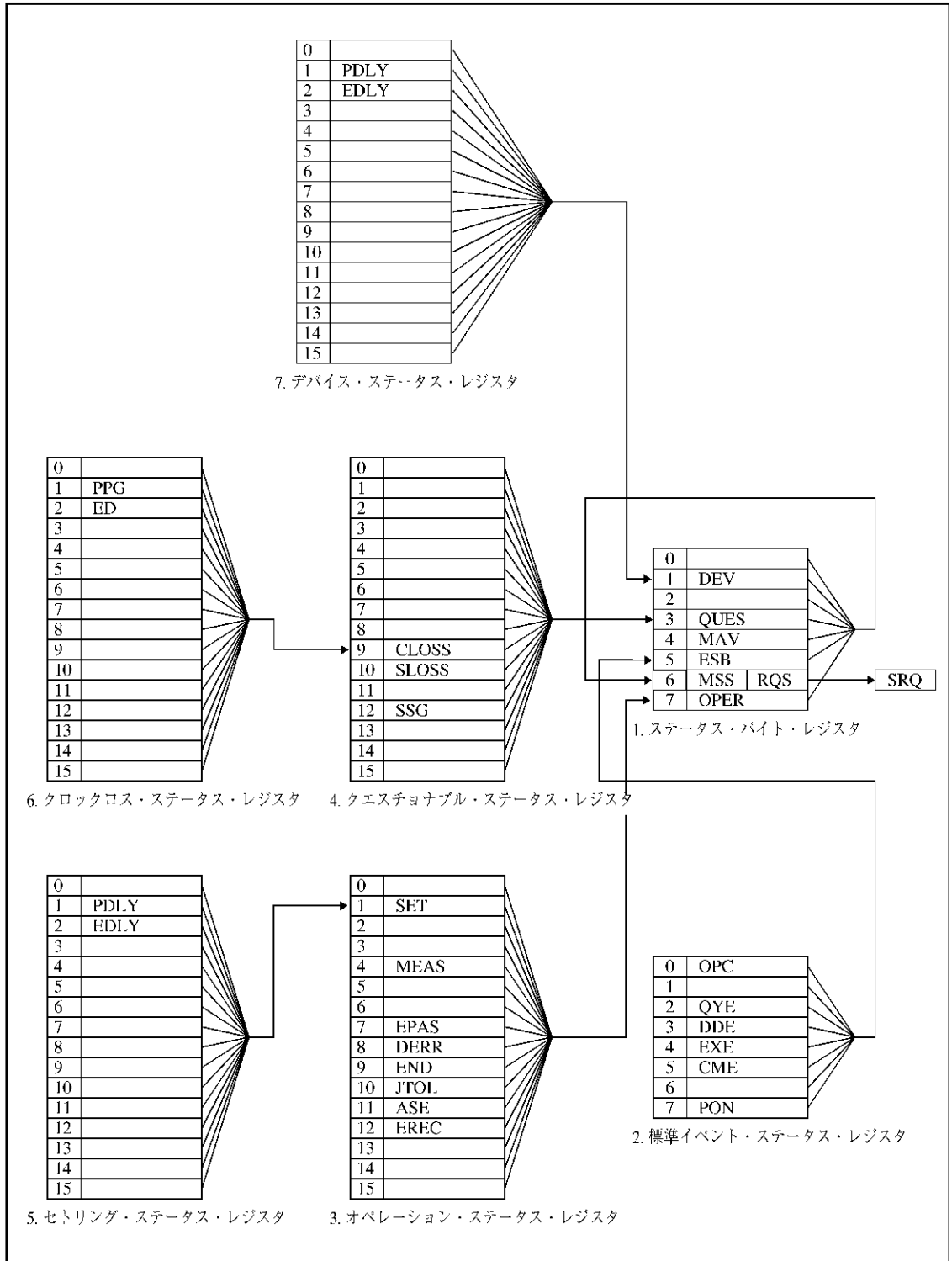


図 4-2 ステータス・レジスタの種類と配置

4.1.2 ステータス・レジスタの種類

本器のステータス・レジスタの配置概要を、以下に示します。

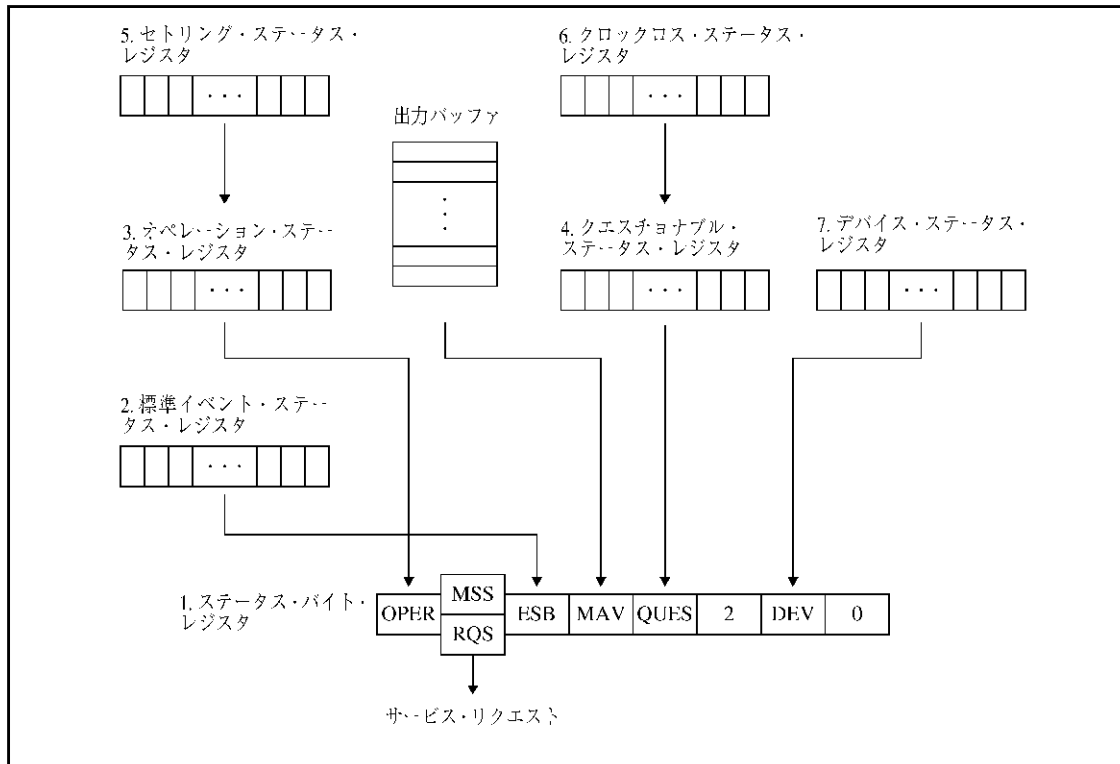


図 4-3 ステータス・レジスタの配置概要

## 4.2 ステータス・バイト・レジスタ

ステータス・バイト・レジスタは、ステータス・レジスタ（4.1.1 項を参照）からの情報を要約しています。また、このステータス・バイト・レジスタのサマリがサービス・リクエストとしてコントローラに送信されます。そのため、ステータス・バイト・レジスタは、ステータス・レジスタ構造とは若干違った動作を行います。この節ではステータス・バイト・レジスタに関して説明します。

ステータス・バイト・レジスタの構造を、以下に示します。

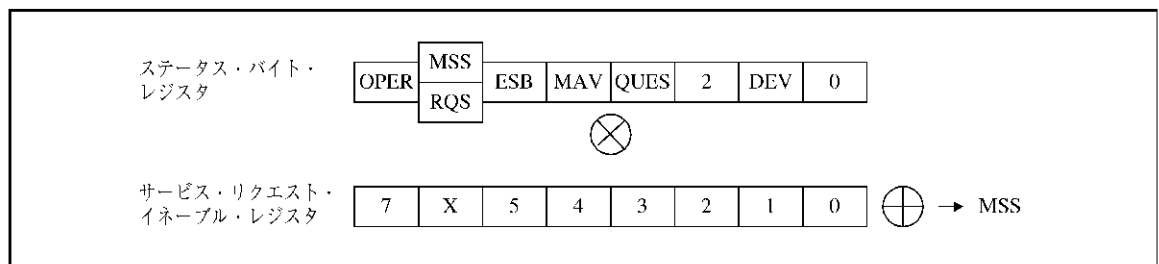


図 4-4 ステータス・バイト・レジスタの構造

このステータス・バイト・レジスタは、以下の 3 点を除くと「4.1.1 項」のステータス・レジスタ構造にしたがいます。

1. ステータス・バイト・レジスタのサマリが、ステータス・バイト・レジスタの bit6 に書き込まれます。
2. イネーブル・レジスタの bit6 は常に有効で変更できません。
3. ステータス・バイト・レジスタの bit6(MSS) が、サービス・リクエスト要求の RQS を書き込みます。

このレジスタがコントローラからのシリアル・ポールに対して応答します。シリアル・ポールに対して応答するときには、ステータス・バイト・レジスタの bit0～5、bit7 および RQS が読み出され、その後に RQS は 0 にリセットされます。その他の bit はそれぞれの要因が 0 になるまでクリアされません。

ステータス・バイト・レジスタ、RQS、MSS は、\*CLS を実行するとクリアできます。

## 4.2 ステータス・バイト・レジスタ

ステータス・バイト・レジスタの各 bit の意味を、以下に示します。  
このレジスタは、他のレジスタのサマリが表示されます。

bit	ニーモニック	説明
7	OPER	OPER は、オペレーション・ステータス・レジスタのサマリです。
6	RQS/MSS	RQS は、ステータス・バイト・レジスタの MSS が 1 になったとき TRUE になるが、その MSS はすべてのステータス・データ構造のサマリ・bit になっています。
		MSS は、シリアル・ポールでは読めません（ただし、RQS が 1 のときは MSS が 1 です）。
		MSS を読むには、共通コマンド *STB? を用います。 *STB? ではステータス・バイト・レジスタの bit0~5、bit7 および MSS が読み出されます。 この場合ステータス・バイト・レジスタと MSS はクリアされません。
		MSS は、ステータス・レジスタ構造のすべてのマスクされていない要因がクリアされるまで 0 になりません。
5	ESB	ESB は標準イベント・ステータス・レジスタのサマリです。
4	MAV	MAV は、出力バッファの要約 bit です。 出力バッファに出力データがある間 1 になり、データが読み出されると 0 になります。
3	QUES	QUES は、クエシヨナブル・ステータス・レジスタのサマリです。
2		常に 0。
1	DEV	DEV は、デバイス・ステータス・レジスタのサマリです。
0		常に 0。



### 4.3 標準イベント・ステータス・レジスタ

標準イベント・ステータス・レジスタの割り当てを、以下に示します。  
このレジスタは、測定器の標準的な情報を提供します。

bit	ニーモニック	説明
7	PON	電源投入で 1 になります。
6		常に 0。
5	CME	パーサーが文法エラーを見つけたときに 1 にセットされます。
4	EXE	GPIB コマンドとして受け取った命令の実行を何らかの理由（パラメータが範囲外など）で失敗すると 1 にセットされます。
3	DDE	Command Error、Execution Error、Query Error 以外のエラーが発生したとき 1 にセットされます。
2	QYE	コントローラが本器からデータを読み出そうとしたときに、データが存在しない、またはデータが消失していると 1 にセットされます。
1		常に 0。
0	OPC	*OPC コマンドを受け取った後、かつ本器が実行しているコマンドがなくなると 1 にセットされます。

#### 4.4 オペレーション・ステータス・レジスタ

オペレーション・ステータス・レジスタの割り当てを、以下に示します。  
 このレジスタは、16ビットのレジスタで、測定器の処理状況を提供します。  
 DERRとENDはイベント・ビットであり、コンディション・レジスタ、トランジション・フィルタは、無効です。

ビット	ニーモニック	説明
15~13		常に0。
12	EREC	エラー記録動作中であることを示します。シンクロス、クロッククロスが発生している間もエラー記録動作中に含めています。 (エラー位相解析オプション)
11	ASE	オートサーチ実行中であることを示します。
10	JTOL	ジッタ耐力測定中であることを示します。 (ジッタ耐力オプション)
9	END	測定期間が経過したことを示します。
8	DERR	基本測定でエラーが検出されたことを示します。
7	EPAS	エラー記録結果をファイルに保存中であることを示します。 (エラー位相解析オプション)
6~5		常に0。
4	MEAS	測定中であることを示します。
3~2		常に0。
1	SET	SETはセトリング・ステータス・レジスタのサマリです。
0		常に0。

#### 4.5 クエスチョナブル・ステータス・レジスタ

クエスチョナブル・ステータス・レジスタの割り当てを、以下に示します。  
このレジスタは、16 bit のレジスタで、測定器のエラー状態を提供します。

bit	ニーモニック	説明
15~13		常に 0。
12	SSG	SSG でエラーが発生していることを示します。
11		常に 0。
10	SLOSS	シンクロスが発生していることを示します。
9	CLOSS	CLOSS はクロックロス・ステータス・レジスタのサマリです。
8~0		常に 0。

#### 4.6 セtring・ステータス・レジスタ

セtring・ステータス・レジスタの割り当てを、以下に示します。  
このレジスタは、16 bit のレジスタで、測定器の設定処理状態を提供します。

bit	ニーモニック	説明
15~13		常に 0。
2	EDLY	ED モジュールのディレイを設定中であることを示します。
1	PDLY	PPG モジュールのディレイを設定中であることを示します。
0		常に 0。

#### 4.7 クロックロス・ステータス・レジスタ

クロックロス・ステータス・レジスタの割り当てを、以下に示します。  
このレジスタは、16 bit のレジスタで、測定器のクロックロス状態を提供します。

bit	ニーモニック	説明
15~13		常に 0。
2	ED	ED モジュールでクロックロスが発生していることを示します。
1	PPG	PPG モジュールでクロックロスが発生していることを示します。
0		常に 0。

## 4.8 デバイス・ステータス・レジスタ

デバイス・ステータス・レジスタの割り当てを、以下に示します。  
このレジスタは、16 bit のレジスタで、測定器の障害情報を提供します。  
コンディション・レジスタ、トランジション・フィルタは存在しません。また、イネーブル・レジスタは常にセットされていて、設定することはできません。

bit	ニーモニック	説明
15 ~ 13		常に 0。
2	EDLY	ED モジュールのダイレイ回路に障害があったことを示します。
1	PDLY	PPG モジュールのダイレイ回路に障害があったことを示します。
0		常に 0。

## 4.9 ステータス・レジスタのクリア、リセット方法

ステータス・レジスタのクリア、リセット方法を以下に示します。

レジスタ	*CLS	起動後	STATUS: PREset
ステータス・バイト・レジスタ	クリア	クリア	変化なし
サービス・リクエスト・イネーブル・レジスタ	変化なし	クリア	変化なし
標準イベント・ステータス・レジスタ	クリア	クリア	変化なし
標準イベント・ステータス・イネーブル・レジスタ	変化なし	クリア	変化なし
オペレーション・ステータス・レジスタ	イベント・レジスタ	クリア	変化なし
	イネーブル・レジスタ	変化なし	オール0
	正トランジション・フィルタ	変化なし	オール1
	負トランジション・フィルタ	変化なし	オール0
クエスチオナブル・ステータス・レジスタ	イベント・レジスタ	クリア	変化なし
	イネーブル・レジスタ	変化なし	オール0
	正トランジション・フィルタ	変化なし	オール1
	負トランジション・フィルタ	変化なし	オール0
クロックロス・ステータス・レジスタ	イベント・レジスタ	クリア	変化なし
	イネーブル・レジスタ	変化なし	オール1
	正トランジション・フィルタ	変化なし	オール1
	負トランジション・フィルタ	変化なし	オール0
セトリング・ステータス・レジスタ	イベント・レジスタ	クリア	変化なし
	イネーブル・レジスタ	変化なし	オール1
	正トランジション・フィルタ	変化なし	オール1
	負トランジション・フィルタ	変化なし	オール0
デバイス・ステータス・レジスタ	イベント・レジスタ	クリア	変化なし



## 5. コマンド・リファレンス

この章は、はじめにサブシステムの構成やコマンドの記述上の注意を説明します。そして、コマンドの詳細の説明をします。

### 注意

- コマンドを参照する場合、コマンド・ニックの一部が省略可能なことを考慮に入れて下さい。  
(例) 以下の2つのコマンドは、表記が異なりますが同じものです。  
SOURCE1:PATTERN:SELECT PROGRAM  
SOURCE1:PATTERN PROGRAM
- SOUR:PATT という記述からコマンドのリファレンスを参照できなかった場合、「付録 A.1 コマンド一覧」からコマンドの完全な記述を探し、そこからリファレンスを参照して下さい。コマンドの完全な記述がわかっている場合、索引からの検索が可能です。

### 5.1 GPIB コマンド・インデックス

この GPIB コマンド・インデックスは、5 章の GPIB コマンド索引として活用して下さい。

GPIB コマンド	参照ページ
*CLS .....	5-14
*ESE .....	5-14
*ESR? .....	5-15
*IDN? .....	5-15
*OPC .....	5-16
*OPT? .....	5-18
*RST .....	5-16
*SRE .....	5-17
*STB? .....	5-17
*TST? .....	5-17
*WAI .....	5-18
CFETch[:SENSe[1]]:BMEasurement:BIERror:BCOunt? .....	5-121
CFETch[:SENSe[1]]:BMEasurement:BIERror:ECOunt? .....	5-122
CFETch[:SENSe[1]]:BMEasurement:BIERror:ERATe? .....	5-121
CFETch[:SENSe[1]]:BMEasurement:BCOunt:INSerting? .....	5-105
CFETch[:SENSe[1]]:BMEasurement:BCOunt:OMITting? .....	5-105
CFETch[:SENSe[1]]:BMEasurement:BCOunt:OTHer? .....	5-105
CFETch[:SENSe[1]]:BMEasurement:BCOunt:OVERhead? .....	5-105
CFETch[:SENSe[1]]:BMEasurement:BCOunt:PAYLoad? .....	5-105
CFETch[:SENSe[1]]:BMEasurement:BCOunt:SPECific? .....	5-105
CFETch[:SENSe[1]]:BMEasurement:BCOunt[:TOTal]? .....	5-105
CFETch[:SENSe[1]]:BMEasurement:CLINtervals? .....	5-117
CFETch[:SENSe[1]]:BMEasurement:ECOunt:INSerting? .....	5-109
CFETch[:SENSe[1]]:BMEasurement:ECOunt:OMITting? .....	5-109

## 5.1 GPIB コマンド・インデックス

CFETch[:SENSe[1]]:BMEasurement:ECOUNT:OTHER?	5-109
CFETch[:SENSe[1]]:BMEasurement:ECOUNT:OVERhead?	5-109
CFETch[:SENSe[1]]:BMEasurement:ECOUNT:PAYLoad?	5-109
CFETch[:SENSe[1]]:BMEasurement:ECOUNT:SPECific?	5-109
CFETch[:SENSe[1]]:BMEasurement:ECOUNT[:TOTAl]?	5-109
CFETch[:SENSe[1]]:BMEasurement:EFINtervals:INSerting?	5-115
CFETch[:SENSe[1]]:BMEasurement:EFINtervals:OMITting?	5-115
CFETch[:SENSe[1]]:BMEasurement:EFINtervals:OTHER?	5-115
CFETch[:SENSe[1]]:BMEasurement:EFINtervals:OVERhead?	5-115
CFETch[:SENSe[1]]:BMEasurement:EFINtervals:PAYLoad?	5-115
CFETch[:SENSe[1]]:BMEasurement:EFINtervals:SPECific?	5-115
CFETch[:SENSe[1]]:BMEasurement:EFINtervals[:TOTAl]?	5-115
CFETch[:SENSe[1]]:BMEasurement:EINtervals:INSerting?	5-113
CFETch[:SENSe[1]]:BMEasurement:EINtervals:OMITting?	5-113
CFETch[:SENSe[1]]:BMEasurement:EINtervals:OTHER?	5-113
CFETch[:SENSe[1]]:BMEasurement:EINtervals:OVERhead?	5-113
CFETch[:SENSe[1]]:BMEasurement:EINtervals:PAYLoad?	5-113
CFETch[:SENSe[1]]:BMEasurement:EINtervals:SPECific?	5-113
CFETch[:SENSe[1]]:BMEasurement:EINtervals[:TOTAl]?	5-113
CFETch[:SENSe[1]]:BMEasurement:EPERformance:DMINutes[:TOTAl]?	5-120
CFETch[:SENSe[1]]:BMEasurement:EPERformance:EFSeconds[:TOTAl]?	5-120
CFETch[:SENSe[1]]:BMEasurement:EPERformance:ESECONDS[:TOTAl]?	5-120
CFETch[:SENSe[1]]:BMEasurement:EPERformance:SESeconds[:TOTAl]?	5-120
CFETch[:SENSe[1]]:BMEasurement:EPERformance:USECONDS[:TOTAl]?	5-120
CFETch[:SENSe[1]]:BMEasurement:ERATE:INSerting?	5-107
CFETch[:SENSe[1]]:BMEasurement:ERATE:OMITting?	5-107
CFETch[:SENSe[1]]:BMEasurement:ERATE:OTHER?	5-107
CFETch[:SENSe[1]]:BMEasurement:ERATE:OVERhead?	5-107
CFETch[:SENSe[1]]:BMEasurement:ERATE:PAYLoad?	5-107
CFETch[:SENSe[1]]:BMEasurement:ERATE:SPECific?	5-107
CFETch[:SENSe[1]]:BMEasurement:ERATE[:TOTAl]?	5-107
CFETch[:SENSe[1]]:BMEasurement:FREQuency?	5-117
CFETch[:SENSe[1]]:BMEasurement:IECOUNT:INSerting?	5-112
CFETch[:SENSe[1]]:BMEasurement:IECOUNT:OMITting?	5-112
CFETch[:SENSe[1]]:BMEasurement:IECOUNT:OTHER?	5-112
CFETch[:SENSe[1]]:BMEasurement:IECOUNT:OVERhead?	5-112
CFETch[:SENSe[1]]:BMEasurement:IECOUNT:PAYLoad?	5-112
CFETch[:SENSe[1]]:BMEasurement:IECOUNT:SPECific?	5-112
CFETch[:SENSe[1]]:BMEasurement:IECOUNT[:TOTAl]?	5-112
CFETch[:SENSe[1]]:BMEasurement:IERate:INSerting?	5-111
CFETch[:SENSe[1]]:BMEasurement:IERate:OMITting?	5-111
CFETch[:SENSe[1]]:BMEasurement:IERate:OTHER?	5-111
CFETch[:SENSe[1]]:BMEasurement:IERate:OVERhead?	5-111
CFETch[:SENSe[1]]:BMEasurement:IERate:PAYLoad?	5-111
CFETch[:SENSe[1]]:BMEasurement:IERate:SPECific?	5-111
CFETch[:SENSe[1]]:BMEasurement:IERate[:TOTAl]?	5-111
CFETch[:SENSe[1]]:BMEasurement:SLINtervals?	5-117
CFETch[:SENSe[1]]:BMEasurement:TEFINtervals:EM[:TOTAl]?	5-119
CFETch[:SENSe[1]]:BMEasurement:TEFINtervals:LEM8[:TOTAl]?	5-119
CFETch[:SENSe[1]]:BMEasurement:TEINtervals:EM[:TOTAl]?	5-118
CFETch[:SENSe[1]]:BMEasurement:TEINtervals:LEM8[:TOTAl]?	5-118



DISPlay:WINDow:BCOunt:DFORmat.....	5-128
DISPlay:WINDow:ECOunt:DFORmat.....	5-128
DISPlay:WINDow:EFINtervals:DFORmat.....	5-130
DISPlay:WINDow:EINtervals:DFORmat.....	5-129
DISPlay:WINDow:EPERformance:DFORmat.....	5-131
DISPlay:WINDow:IECOunt:DFORmat.....	5-129
DISPlay:WINDow:TEFIntervals:DFORmat.....	5-131
DISPlay:WINDow:TEINtervals:DFORmat.....	5-130
FETCh:JTOLerance:MDATa:JAMplitude:COUnT?	5-126
FETCh:JTOLerance:MDATa:JFRequency:COUnT?	5-125
FETCh:JTOLerance:MDATa?	5-126
FETCh:JTOLerance:MTIME:ETIME?	5-127
FETCh:JTOLerance:MTIME:STIME?	5-127
FETCh[:SENSe[1]]:BMEasurement:B1ERror:BCOunt?	5-121
FETCh[:SENSe[1]]:BMEasurement:B1ERror:ECOunt?	5-122
FETCh[:SENSe[1]]:BMEasurement:B1ERror:ERATe?	5-121
FETCh[:SENSe[1]]:BMEasurement:BCOunt:INSerting?	5-105
FETCh[:SENSe[1]]:BMEasurement:BCOunt:OMITting?	5-105
FETCh[:SENSe[1]]:BMEasurement:BCOunt:OTHer?	5-105
FETCh[:SENSe[1]]:BMEasurement:BCOunt:OVERhead?	5-105
FETCh[:SENSe[1]]:BMEasurement:BCOunt:PAYLoad?	5-105
FETCh[:SENSe[1]]:BMEasurement:BCOunt:SPECific?	5-105
FETCh[:SENSe[1]]:BMEasurement:BCOunt[:TOTal]?	5-105
FETCh[:SENSe[1]]:BMEasurement:CLINtervals?.....	5-117
FETCh[:SENSe[1]]:BMEasurement:ECOunt:INSerting?	5-109
FETCh[:SENSe[1]]:BMEasurement:ECOunt:OMITting?	5-109
FETCh[:SENSe[1]]:BMEasurement:ECOunt:OTHer?	5-109
FETCh[:SENSe[1]]:BMEasurement:ECOunt:OVERhead?	5-109
FETCh[:SENSe[1]]:BMEasurement:ECOunt:PAYLoad?	5-109
FETCh[:SENSe[1]]:BMEasurement:ECOunt:SPECific?	5-109
FETCh[:SENSe[1]]:BMEasurement:ECOunt[:TOTal]?	5-109
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:INSerting?	5-115
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:OMITting?	5-115
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:OTHer?	5-115
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:OVERhead?	5-115
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:PAYLoad?	5-115
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:SPECific?	5-115
FETCh[:SENSe[1]]:BMEasurement:EFINtervals[:TOTal]?	5-115
FETCh[:SENSe[1]]:BMEasurement:EINtervals:INSerting?	5-113
FETCh[:SENSe[1]]:BMEasurement:EINtervals:OMITting?	5-113
FETCh[:SENSe[1]]:BMEasurement:EINtervals:OTHer?	5-113
FETCh[:SENSe[1]]:BMEasurement:EINtervals:OVERhead?	5-113
FETCh[:SENSe[1]]:BMEasurement:EINtervals:PAYLoad?	5-113
FETCh[:SENSe[1]]:BMEasurement:EINtervals:SPECific?	5-113
FETCh[:SENSe[1]]:BMEasurement:EINtervals[:TOTal]?	5-113
FETCh[:SENSe[1]]:BMEasurement:EPERformance:DMINutes[:TOTal]?	5-120
FETCh[:SENSe[1]]:BMEasurement:EPERformance:EFSeconds[:TOTal]?	5-120
FETCh[:SENSe[1]]:BMEasurement:EPERformance:ESEConds[:TOTal]?	5-120
FETCh[:SENSe[1]]:BMEasurement:EPERformance:SESeconds[:TOTal]?	5-120
FETCh[:SENSe[1]]:BMEasurement:EPERformance:USEConds[:TOTal]?	5-120
FETCh[:SENSe[1]]:BMEasurement:ERATe:INSerting?	5-107

## 5.1 GPIB コマンド・インデックス

FETCh[:SENSe[1]]:BMEasurement:ERATe:OMITting?	5-107
FETCh[:SENSe[1]]:BMEasurement:ERATe:OTHer?	5-107
FETCh[:SENSe[1]]:BMEasurement:ERATe:OVERhead?	5-107
FETCh[:SENSe[1]]:BMEasurement:ERATe:PAYLoad?	5-107
FETCh[:SENSe[1]]:BMEasurement:ERATe:SPECific?	5-107
FETCh[:SENSe[1]]:BMEasurement:ERATe[:TOTAl]?	5-107
FETCh[:SENSe[1]]:BMEasurement:FREQuency?	5-117
FETCh[:SENSe[1]]:BMEasurement:SLINtervals?	5-117
FETCh[:SENSe[1]]:BMEasurement:TEFintervals:EM[:TOTAl]?	5-119
FETCh[:SENSe[1]]:BMEasurement:TEFintervals:LEM8[:TOTAl]?	5-119
FETCh[:SENSe[1]]:BMEasurement:TEINtervals:EM[:TOTAl]?	5-118
FETCh[:SENSe[1]]:BMEasurement:TEINtervals:LEM8[:TOTAl]?	5-118
FETCh[:SENSe[1]]:EPANalysis:MDATa:TSErIes:CYCLE:COUNt?	5-125
FETCh[:SENSe[1]]:EPANalysis:MDATa:TSErIes:ERRor:COUNt?	5-125
FETCh[:SENSe[1]]:MONitor:HISTory:CLOSs?	5-124
FETCh[:SENSe[1]]:MONitor:HISTory:DERRor?	5-124
FETCh[:SENSe[1]]:MONitor:HISTory:SLOSs?	5-124
FETCh[:SENSe[1]]:MONitor:RTIME:CLOSs?	5-124
FETCh[:SENSe[1]]:MONitor:RTIME:DERRor?	5-124
FETCh[:SENSe[1]]:MONitor:RTIME:SLOSs?	5-124
FETCh[:SENSe[1]]:MTIME:ETIME?	5-122
FETCh[:SENSe[1]]:MTIME:RTIME?	5-123
FETCh[:SENSe[1]]:MTIME:STIME?	5-123
FETCh[:SENSe[1]]:MTIME:TIMed?	5-122
HCOPy[:ITEM]:ED:BMEasurement	5-132
HCOPy[:ITEM]:JTOLerance	5-132
INPut[1]:POLarity	5-97
INPut[1]:TERMination	5-98
INPut[1]:TERMination:VARiable	5-98
INPut[1]:TERMination:VOLTagE	5-99
INPut[1]:TVOLtagE	5-97
INPut2:DELay	5-100
INPut2:POLarity	5-100
INPut2:TERMination	5-101
INPut2:TERMination:VARiable	5-101
INPut2:TERMination:VOLTagE	5-102
JTOLerance:ASEarch	5-166
JTOLerance:CFRequency	5-165
JTOLerance:ETHReshold	5-166
JTOLerance:MDATa	5-166
JTOLerance:MDATa:ALL:PASS?	5-167
JTOLerance:MODE	5-164
JTOLerance:PERiod	5-165
JTOLerance:STIME	5-165
JTOLerance:TABLE	5-159
JTOLerance:TABLE:COUNt?	5-161
JTOLerance:TABLE:NEW	5-162
JTOLerance:TABLE:REMOve	5-164
JTOLerance:TEMPlate:SElect	5-164
JTOLerance[:STATe]	5-159
MMEMory:LOAD:ED:FLEXible	5-135

MMEMory:LOAD:ED:FLEXible:PROGrama	5-136
MMEMory:LOAD:ED:PROGrama	5-135
MMEMory:LOAD:ED:SETUp	5-135
MMEMory:LOAD:ED:STM	5-135
MMEMory:LOAD:JTOLerance	5-136
MMEMory:LOAD:PPG:FLEXible	5-134
MMEMory:LOAD:PPG:FLEXible:PROGrama	5-134
MMEMory:LOAD:PPG:PROGrama	5-134
MMEMory:LOAD:PPG:SETUp	5-133
MMEMory:LOAD:PPG:STM	5-134
MMEMory:LOAD:SETUp	5-133
MMEMory:LOAD:SSG:SETUp	5-133
MMEMory:STORE:ED:BMEasurement:LOG	5-139
MMEMory:STORE:ED:EPANalysis:CANCel	5-141
MMEMory:STORE:ED:EPANalysis:FAIL?	5-142
MMEMory:STORE:ED:EPANalysis:STATistics:TEXT	5-141
MMEMory:STORE:ED:EPANalysis:TSErIES:TEXT	5-140
MMEMory:STORE:ED:EPANalysis?	5-141
MMEMory:STORE:ED:FLEXible	5-139
MMEMory:STORE:ED:FLEXible:PROGrama	5-139
MMEMory:STORE:ED:PROGrama	5-138
MMEMory:STORE:ED:SETUp	5-138
MMEMory:STORE:ED:STM	5-138
MMEMory:STORE:JTOLerance	5-142
MMEMory:STORE:JTOLerance:TEXT	5-142
MMEMory:STORE:PPG:FLEXible	5-137
MMEMory:STORE:PPG:FLEXible:PROGrama	5-138
MMEMory:STORE:PPG:PROGrama	5-137
MMEMory:STORE:PPG:SETUp	5-137
MMEMory:STORE:PPG:STM	5-137
MMEMory:STORE:SETUp	5-136
MMEMory:STORE:SSG:SETUp	5-136
OUTPut[1]:CROSS	5-39
OUTPut[1]:TERMination	5-39
OUTPut[1]:TERMination:VOLTagE	5-40
OUTPut[1]:VOLTagE[:LEVel][:IMMediate]:HIGH	5-38
OUTPut[1]:VOLTagE[:LEVel][:IMMediate]:LOW	5-39
OUTPut[1]:VOLTagE[:LEVel][:IMMediate]:MIDDLE	5-38
OUTPut[1]:VOLTagE[:LEVel][:IMMediate][:AMPLitude]	5-37
OUTPut[1][:STATe]	5-37
OUTPut2:TERMination	5-47
OUTPut2:TERMination:VOLTagE	5-48
OUTPut2:VOLTagE[:LEVel][:IMMediate]:HIGH	5-46
OUTPut2:VOLTagE[:LEVel][:IMMediate]:LOW	5-47
OUTPut2:VOLTagE[:LEVel][:IMMediate]:MIDDLE	5-46
OUTPut2:VOLTagE[:LEVel][:IMMediate][:AMPLitude]	5-45
OUTPut2[:STATe]	5-45
OUTPut3:CROSS	5-42
OUTPut3:TERMination	5-43
OUTPut3:TERMination:VOLTagE	5-43
OUTPut3:TRACK	5-43

## 5.1 GPIB コマンド・インデックス

OUTPut3:VOLTage[:LEVel][:IMMEDIATE]:HIGH .....	5-41
OUTPut3:VOLTage[:LEVel][:IMMEDIATE]:LOW .....	5-42
OUTPut3:VOLTage[:LEVel][:IMMEDIATE]:MIDDLE.....	5-42
OUTPut3:VOLTage[:LEVel][:IMMEDIATE][:AMPLitude] .....	5-41
OUTPut3[:STATE] .....	5-40
OUTPut4:TERMination.....	5-51
OUTPut4:TERMination:VOLTage .....	5-52
OUTPut4:TRACk .....	5-52
OUTPut4:VOLTage[:LEVel][:IMMEDIATE]:HIGH .....	5-50
OUTPut4:VOLTage[:LEVel][:IMMEDIATE]:LOW .....	5-51
OUTPut4:VOLTage[:LEVel][:IMMEDIATE]:MIDDLE.....	5-50
OUTPut4:VOLTage[:LEVel][:IMMEDIATE][:AMPLitude] .....	5-49
OUTPut4[:STATE] .....	5-49
SENSe[1]:ASEarch:CLOCK .....	5-82
SENSe[1]:ASEarch:FAIL? .....	5-81
SENSe[1]:ASEarch:PATTERn .....	5-82
SENSe[1]:ASEarch:TVOLTage .....	5-82
SENSe[1]:ASEarch[:STATE] .....	5-81
SENSe[1]:BMEasurement:BURSt .....	5-87
SENSe[1]:BMEasurement:CDATa .....	5-89
SENSe[1]:BMEasurement:DMODE .....	5-87
SENSe[1]:BMEasurement:EPERformance:THRESHold .....	5-89
SENSe[1]:BMEasurement:EVALuate:CLOSs .....	5-87
SENSe[1]:BMEasurement:EVALuate:SLOSs .....	5-88
SENSe[1]:BMEasurement:MTIME:INTERval .....	5-86
SENSe[1]:BMEasurement:MTIME:MODE .....	5-85
SENSe[1]:BMEasurement:MTIME:PERiod.....	5-86
SENSe[1]:BMEasurement[:STATE] .....	5-85
SENSe[1]:EPANalysis:ERECord:AREa .....	5-96
SENSe[1]:EPANalysis:ERECord[:STATE] .....	5-95
SENSe[1]:LOG:BMEasurement.....	5-83
SENSe[1]:LOG:HISTory:MDATa:EFINTERvals .....	5-84
SENSe[1]:LOG:HISTory:MDATa:EINTERvals .....	5-84
SENSe[1]:LOG:HISTory:MDATa:FREQuency .....	5-85
SENSe[1]:LOG:HISTory:MDATa:IECount .....	5-84
SENSe[1]:LOG:HISTory:MDATa:IERate.....	5-83
SENSe[1]:LOG:HISTory:SCLoss .....	5-83
SENSe[1]:MASK:ROUte.....	5-81
SENSe[1]:MONitor:HISTory:CLEar .....	5-83
SENSe[1]:PATTERn:FLEXible:PRBS:MRATio .....	5-75
SENSe[1]:PATTERn:FLEXible:PRBS:SElect .....	5-75
SENSe[1]:PATTERn:FLEXible:PROGram:BDATa .....	5-74
SENSe[1]:PATTERn:FLEXible:PROGram:DATA.....	5-73
SENSe[1]:PATTERn:FLEXible:PROGram[:LENGth] .....	5-72
SENSe[1]:PATTERn:FLEXible:STABle.....	5-69
SENSe[1]:PATTERn:FLEXible:STABle:COUNt? .....	5-70
SENSe[1]:PATTERn:FLEXible:STABle:DEFault .....	5-72
SENSe[1]:PATTERn:FLEXible:STABle:NEW .....	5-71
SENSe[1]:PATTERn:FLEXible:STABle:REMOve .....	5-72
SENSe[1]:PATTERn:POLarity .....	5-76
SENSe[1]:PATTERn:PRBS:MRATio .....	5-61

SENSe[1]:PATtern:PROGram:BDATa .....	5-64
SENSe[1]:PATtern:PROGram:DATA .....	5-63
SENSe[1]:PATtern:PROGram[:LENGth] .....	5-62
SENSe[1]:PATtern:STM:B1 .....	5-69
SENSe[1]:PATtern:STM:BDATa .....	5-66
SENSe[1]:PATtern:STM:DATA .....	5-65
SENSe[1]:PATtern:STM:NFRame .....	5-64
SENSe[1]:PATtern:STM:PAYLoad:PRBS .....	5-67
SENSe[1]:PATtern:STM:PAYLoad:PRBS:MRATio .....	5-68
SENSe[1]:PATtern:STM:PAYLoad:PRBS:SElect .....	5-67
SENSe[1]:PATtern:STM:SCRamble .....	5-68
SENSe[1]:PATtern:TRACK .....	5-76
SENSe[1]:PATtern:ZSUBstitut[:ZLEngth] .....	5-61
SENSe[1]:PATtern[:SElect] .....	5-60
SENSe[1]:SPECific:FLEXible:END:ADDRESS .....	5-95
SENSe[1]:SPECific:FLEXible:END:INDEX .....	5-94
SENSe[1]:SPECific:FLEXible:START:ADDRESS .....	5-94
SENSe[1]:SPECific:FLEXible:START:INDEX .....	5-94
SENSe[1]:SPECific:PROGram:END:ADDRESS .....	5-90
SENSe[1]:SPECific:PROGram:START:ADDRESS .....	5-90
SENSe[1]:SPECific:STM:END:COLumn .....	5-93
SENSe[1]:SPECific:STM:END:FNUMber .....	5-92
SENSe[1]:SPECific:STM:END:ROW .....	5-92
SENSe[1]:SPECific:STM:MODE .....	5-93
SENSe[1]:SPECific:STM:START:COLumn .....	5-92
SENSe[1]:SPECific:STM:START:FNUMber .....	5-91
SENSe[1]:SPECific:STM:START:ROW .....	5-91
SENSe[1]:SPECific:ZSUBstitut:END:ADDRESS .....	5-91
SENSe[1]:SPECific:ZSUBstitut:START:ADDRESS .....	5-90
SENSe[1]:SYNChronisat:PATtern:FLEXible:ADDRESS .....	5-78
SENSe[1]:SYNChronisat:PATtern:FLEXible:INDEX .....	5-77
SENSe[1]:SYNChronisat:PATtern:LENGth .....	5-78
SENSe[1]:SYNChronisat:PATtern:PROGram:ADDRESS .....	5-77
SENSe[1]:SYNChronisat:THReshold:AUTO .....	5-78
SENSe[1]:SYNChronisat:THReshold:MEMory:GAIN .....	5-80
SENSe[1]:SYNChronisat:THReshold:MEMory:LOSS .....	5-80
SENSe[1]:SYNChronisat:THReshold:PRBS:GAIN .....	5-79
SENSe[1]:SYNChronisat:THReshold:PRBS:LOSS .....	5-79
SENSe[1]:SYNChronisat[:AUTO] .....	5-76
SOURce[1]:EADdition .....	5-35
SOURce[1]:EADdition:MODE .....	5-36
SOURce[1]:EADdition:RATE .....	5-36
SOURce[1]:EADdition:ROUte .....	5-36
SOURce[1]:EADdition:SINGLE .....	5-35
SOURce[1]:PATtern:FLEXible:PRBS:MRATio .....	5-34
SOURce[1]:PATtern:FLEXible:PRBS:SElect .....	5-34
SOURce[1]:PATtern:FLEXible:PROGram:BDATa .....	5-33
SOURce[1]:PATtern:FLEXible:PROGram:DATA .....	5-32
SOURce[1]:PATtern:FLEXible:PROGram[:LENGth] .....	5-31
SOURce[1]:PATtern:FLEXible:STABLE .....	5-28
SOURce[1]:PATtern:FLEXible:STABLE:COUNT? .....	5-29

## 5.1 GPIB コマンド・インデックス

SOURce[1]:PATTerN:FLEXible:STABle:DEFault .....	5-31
SOURce[1]:PATTerN:FLEXible:STABle:NEW .....	5-30
SOURce[1]:PATTerN:FLEXible:STABle:REMOve .....	5-31
SOURce[1]:PATTerN:POLarity .....	5-35
SOURce[1]:PATTerN:PRBS:MRATio .....	5-20
SOURce[1]:PATTerN:PROGrama:BDATA .....	5-23
SOURce[1]:PATTerN:PROGrama:DATA .....	5-22
SOURce[1]:PATTerN:PROGrama[:LENGth] .....	5-21
SOURce[1]:PATTerN:STM:B1 .....	5-28
SOURce[1]:PATTerN:STM:BDATA .....	5-25
SOURce[1]:PATTerN:STM:DATA .....	5-24
SOURce[1]:PATTerN:STM:NFRame .....	5-23
SOURce[1]:PATTerN:STM:PAYLoad:PRBS .....	5-26
SOURce[1]:PATTerN:STM:PAYLoad:PRBS:MRATio .....	5-27
SOURce[1]:PATTerN:STM:PAYLoad:PRBS:SElect .....	5-26
SOURce[1]:PATTerN:STM:SCRamble .....	5-27
SOURce[1]:PATTerN:ZSUBstitut[:ZLENGth] .....	5-20
SOURce[1]:PATTerN[:SElect] .....	5-19
SOURce10:FREQuency[:CW] .....	5-155
SOURce10:JMODulation:AMPLitude .....	5-157
SOURce10:JMODulation:FREQuency .....	5-156
SOURce10:JMODulation[:STATe] .....	5-156
SOURce10:OUTPut[:STATe] .....	5-155
SOURce10:REFerence .....	5-156
SOURce2:DELay .....	5-44
SOURce5:BOUtpuT:CYLE .....	5-53
SOURce5:BOUtpuT:OTIME .....	5-54
SOURce5:BOUtpuT:SOURce .....	5-53
SOURce5:BOUtpuT[:STATe] .....	5-53
SOURce6:TRIGger:FLEXible:ADDReSS .....	5-58
SOURce6:TRIGger:FLEXible:INDex .....	5-58
SOURce6:TRIGger:PRBS:POSition .....	5-55
SOURce6:TRIGger:PROGrama:POSition .....	5-56
SOURce6:TRIGger:STM:COLumn .....	5-57
SOURce6:TRIGger:STM:FNUMber .....	5-56
SOURce6:TRIGger:STM:MODE .....	5-58
SOURce6:TRIGger:STM:ROW .....	5-57
SOURce6:TRIGger:ZSUBstitut:POSition .....	5-56
SOURce6:TRIGger[:MODE] .....	5-55
SOURce8:TRIGger:STM:MODE .....	5-103
SOURce8:TRIGger[:MODE] .....	5-103
SOURce9:AUX[:MODE] .....	5-104
STATus:DEVIce[:EVENT]? .....	5-153
STATus:OPERation:CONDition? .....	5-146
STATus:OPERation:ENABLE .....	5-147
STATus:OPERation:NTRansition .....	5-147
STATus:OPERation:PTRansition .....	5-147
STATus:OPERation:SETTling:CONDition? .....	5-148
STATus:OPERation:SETTling:ENABLE .....	5-148
STATus:OPERation:SETTling:NTRansition .....	5-149
STATus:OPERation:SETTling:PTRansition .....	5-149

STATus:OPERation:SETTLing[:EVENT]? .....	5-148
STATus:OPERation[:EVENT]? .....	5-146
STATus:PRESet .....	5-154
STATus:QUEStionable:CLOSs:CONDition? .....	5-152
STATus:QUEStionable:CLOSs:ENABle.....	5-152
STATus:QUEStionable:CLOSs:NTRansition.....	5-153
STATus:QUEStionable:CLOSs:PTRansition .....	5-153
STATus:QUEStionable:CLOSs[:EVENT]? .....	5-151
STATus:QUEStionable:CONDition? .....	5-150
STATus:QUEStionable:ENABle.....	5-150
STATus:QUEStionable:NTRansition.....	5-151
STATus:QUEStionable:PTRansition .....	5-151
STATus:QUEStionable[:EVENT]? .....	5-149
SYSTem:BEEPer:ALARm[:STATe] .....	5-143
SYSTem:BEEPer:DERRor[:STATe] .....	5-143
SYSTem:DATE .....	5-145
SYSTem:ERRor? .....	5-144
SYSTem:TIME.....	5-145

## 5.2 コマンド解説の前に

## 5.2 コマンド解説の前に

ここでは、サブシステムの構成やコマンドの記述上の注意を説明します。

## 5.2.1 サブシステムの構成

本器のコマンドは、以下のサブシステムから構成されています。  
同名のサブシステムに対する対応モジュールの指定は、サフィックスにより行います。

サブシステム	説明	対応モジュール
共通コマンド	すべての測定器で同じ動作をするコマンドです。	
SOURce[1]	PPG の発生信号に関するコマンドです。 パターン、エラー付加の設定などが含まれます。	PPG
OUTPut[1]	DATA 出力に関するコマンドです。 DATA 出力信号、終端電圧の設定などが含まれます。	PPG
OUTPut3	DATA 出力に関するコマンドです。 DATA 出力信号、終端電圧の設定などが含まれます。	PPG
SOURce2	クロック信号に関するコマンドです。 クロック遅延量の設定などが含まれます。	PPG
OUTPut2	CLOCK 出力に関するコマンドです。 CLOCK 出力信号、終端電圧の設定などが含まれます。	PPG
OUTPut4	CLOCK 出力に関するコマンドです。 CLOCK 出力信号、終端電圧の設定などが含まれます。	PPG
SOURce5	BURST OUTPUT に関するコマンドです。 信号源、バースト信号の設定などが含まれます。	PPG
SOURce6	トリガ出力に関するコマンドです。 トリガ出力の種類、位置の設定などが含まれます。	PPG
SOURce7	補助出力に関するコマンドです。 補助出力の種類の設定などが含まれます。	PPG
SENSe[1]	測定に関するコマンドです。 パターン、同期、オート・サーチ、基本測定の設定などが含まれます。	ED
INPut[1]	DATA 入力に関するコマンドです。 DATA 入力極性、スレッシュホールド電圧、終端電圧の設定などが含まれます。	ED
INPut2	CLOCK 入力に関するコマンドです。 CLOCK 入力極性、遅延量、終端電圧の設定などが含まれます。	ED
SOURce8	トリガ出力に関するコマンドです。 トリガ出力の種類の設定などが含まれます。	ED



サブシステム	説明	対応モジュール
SOURce9	補助出力に関するコマンドです。 補助出力の種類の設定などが含まれます。	ED
FETCh	エラー測定結果、測定時間、エラー・モニタ情報を問い合わせます。	ED
CFETch	エラー測定の途中経過を問い合わせます。	ED
DISPlay	各種表示に関するコマンドです。 表示フォーマットの設定などが含まれます。	
HCOPy	測定結果の印刷を行うコマンドです。	
MMEMory	測定データや設定情報の読み込み／保存に関するコマンドです。	
SYSTem	システムに関するコマンドです。 ブザー、日付、時刻の設定などが含まれます。	
STATus	ステータス・レジスタに関するコマンドです。 オペレーション・ステータス・レジスタ、クエスチョナブル・ステータス・レジスタの設定／問い合わせなどが含まれます。	
SOURce10	SSGに関するコマンドです。 周波数の設定などが含まれます。	SSG
JTOLerance	ジッタ耐力測定の設定に関するコマンドです。ジッタ耐力パラメータ・テーブル、クロック周波数、測定モード、判定条件の設定などが含まれます。	SSG, ED

## 5.2.2 コマンド記述上の注意

1. コマンドと応答データ・フォーマットは、以下の記号を用いて記述します。
  - 記号 <>: 文法の構成要素を示す  
その内容は、その後に記述される
  - 記号 |: 複数の中から1つを選択することを示す  
(例) A|B|C           これは A、B または C という意味
  - 記号 []: 囲まれた項目は、オプション (省略可能) であることを示す
  - 記号 {}: 囲まれた項目は、グループを表し、{} の中で | で区切られた複数の項目の1つを選択することを示す
2. コマンドとクエリの存在を以下のように記述します。
  - Command/Query:       コマンドとクエリがどちらも存在する
  - Command:               コマンドだけが存在する
  - Query:                  クエリだけが存在する
3. 4 文字以上のニックネームはショート・フォームをもちます。本文中では大文字で記述した部分がショート・フォームになります。
  - (例) SOURce1:PATtern:SELEct PROGRAM  
      ショート・フォーム   SOUR1:PATT:SEL PROG  
      ロング・フォーム     SOURCE1:PATTERN:SELECT PROGRAM
4. クエリは、コマンドのヘッダに ? をつけます。パラメータを必要とするクエリは、クエリのフォーマットも記述します。
5. コマンド詳細で共通に用いているパラメータの書式を、以下に示します。
  - <int>:       数値データで NR1、NR2、NR3 の各フォーマットで入力できる  
              本器が受け取った後、整数に丸め込まれる
  - <real>:      数値データで NR1、NR2、NR3 の各フォーマットで入力できる  
              本器が受け取った後、有効な桁数の実数に丸め込まれる
  - <bool>:      0|1 のスイッチ  
              0 と OFF、1 と ON がそれぞれ対応する
  - <str>:       文字列  
              “または” で囲まれた英数記号を示す
  - <block>:     ブロック・データ型  
              データの内容は 8 bits のバイナリ・データ列  
              フォーマットは、「3.2 データ・フォーマット」を参照して下さい。

### 5.2.3 D3371 とコントローラの同期

以下に示すコマンドは、IEEE488.2 で定義されるオーバーラップ・コマンドです。

SOURce2:DElay <int> (PPG モジュールのクロック遅延量設定)

INPut2:DElay <int> (ED モジュールのクロック遅延量設定)

SENSe[1]:ASEarch[:STATe] ON (オート・サーチの実行)

JTOlerance[:STATe] ON (ジッタ耐力測定の実行)

MMEMory:STORe:ED:EPANalysis:TSEries:TEXT <cn>,<sp1>,<sp2>,<sp3>,<nr>,<str>  
(エラー記録結果を時系列フォーマットでファイルに保存)

MMEMory:STORe:ED:EPANalysis:STATistics:TEXT <sp1>,<sp2>,<sp3>,<range>,<str>  
(エラー記録結果を統計フォーマットでファイルに保存)

\*OPC, \*OPC?, \*WAI を使用して、すべての動作完了のタイミングで同期を取ることができます。

上記以外のコマンドは、すべてシーケンシャル・コマンドです。

### 5.2.4 システム・バージョンによるコマンド差異について

表 5-1 システム・バージョンによるコマンド差異

GPIB コマンド	A00/A02	B00
SENS[1]:SYNC:THR:MEM:GAIN <real>	<real> に 1E-11, 1E-12 使用可能	<real> に 1E-11,1E-12 使用不可
SENS[1]:SYNC:THR:MEM:LOSS <real>	<real> に 1E-11, 1E-12 使用可能	<real> に 1E-11,1E-12 使用不可
SOUR<n>:TRIG[:MODE] SYNC	SOUR8:TRIG[:MODE] SYNC フロント端子から SYNC トリガ出力	SOUR9:AUX[:MODE] SYNC リア端子から SYNC トリガ出力

## 5.3 共通コマンド

## 5.3 共通コマンド

## 1. \*CLS

- 機能                                    ステータス・バイトと関連データのクリア
- コマンドとクエリの存在    Command
- コマンド                                \*CLS
- 説明                                    \*CLS はステータス・データ構造をクリアし、強制的に \*OPC と \*OPC? をキャンセルします。また、エラー・キューもクリアします。しかし、このコマンド自身は出力バッファをクリアしないので、出力データがある場合 MAV ビットはクリアされません。ただし、行の最初にこのコマンドを実行するとデータがクリアされるので、MAV を含めてすべてのステータスがクリアされます。

## 2. \*ESE

- 機能                                    標準イベント・ステータス・イネーブル・レジスタの設定
- コマンドとクエリの存在    Command/Query
- コマンド                                \*ESE <int>
- パラメータ                             <int>
- 応答形式                               NR1 (整数型)
- 説明                                    標準イベント・ステータス・レジスタのイネーブル・レジスタを設定します。このレジスタの 1 に設定されたビットに対応する標準イベント・ステータス・レジスタが、有効ビットとしてステータス・バイト・レジスタに反映されます。  
詳細は「4. ステータス・バイト」を参照して下さい。
- 例                                        DDE (ビット 3) と OPC (ビット 0) をイネーブルに設定する場合、 $2^3 + 2^0 = 8 + 1 = 9$  を計算し、\*ESE 9 と設定します。

## 3. \*ESR?

- 機能 標準イベント・ステータス・レジスタの読み出し
- コマンドとクエリの存在 Query
- クエリ \*ESR?
- 応答形式 NR1 (整数型)
- 説明 標準イベント・ステータス・レジスタの内容を読み出します。  
標準イベント・ステータス・レジスタは読み出すとクリアされ、対応するステータス・バイトのビット (ビット 5) もクリアされます。  
詳細は「4. ステータス・バイト」を参照して下さい。

## 4. \*IDN?

- 機能 機器の問い合わせ
- コマンドとクエリの存在 Query
- クエリ \*IDN?
- 応答形式 <manufacturer>, <model>, <serial number>, <firmware level>  
<manufacturer> = ADVANTEST  
<model> = 機種名  
<serial number> = シリアル番号  
<firmware level> = システム・バージョン
- 説明 本器の識別情報を問い合わせます。上記の応答形式の項目で記述している 4 項目を文字列形式で出力します。

## 5.3 共通コマンド

## 5. \*OPC

- 機能 実行中のすべての動作の終了通知
- コマンドとクエリの存在 Command/Query
- コマンド \*OPC
- 応答形式 1
- 説明
 

\*OPCは現在実行中のすべてのコマンドが終了したときに標準イベント・ステータス・レジスタの OPC ビットを 1 に設定します。  
 ”現在実行中のすべてのコマンド”が終了する前に次のコマンドを受け取ると、そのコマンド実行の終了も待ちます。つまり、\*OPCを受け取った後に本器が何も実行していない状態になったときにステータス・レジスタの設定をします。  
 \*OPC? は上記の \*OPC で設定する OPC ビットの代わりに出力バッファに 1 を書き込みます。つまり、コントローラが本器からの応答を受け取るタイミングでコマンド終了のタイミングをとれます。  
 \*OPC、\*OPC? とともに DCL インタフェース・メッセージ、\*CLS、および \*RST で解除されます。  
 \*WAI も参照して下さい。

## 6. \*RST

- 機能 機器のリセット
- コマンドとクエリの存在 Command
- クエリ \*RST
- 説明
 

\*RST は本器のリセットを実行します。実際には以下のことを実行します。

  1. 本器の設定を初期状態にする（「A.3 初期設定」を参照）。
  2. \*OPC、\*OPC? を無効にする。

以下への影響はありません。

  1. GPIB バスの状態
  2. GPIB アドレス
  3. 出力バッファ
  4. ステータス・データ構造
  5. デバイスの校正データ

7. \*SRE
- 機能 サービス・リクエスト・イネーブル・レジスタの設定
  - コマンドとクエリの存在 Command/Query
  - コマンド \*SRE <int>
  - 応答形式 NR1 (整数型)
  - 説明 サービス・リクエスト・イネーブル・レジスタを設定します。  
このレジスタの1に設定されたビットに対応するステータス・バイト・レジスタが有効ビットとしてMSSに反映します。  
クエリ時の応答データビット6は、常に0となります。  
詳細は「4. ステータス・バイト」を参照して下さい。
  - 例 OPER (ビット7), ESB (ビット5) および MAV (ビット4) をイネーブルに設定する場合、 $2^7 + 2^5 + 2^4 = 128 + 32 + 16 = 176$  を計算し、\*SRE 176 と設定します。
8. \*STB?
- 機能 ステータス・バイト・レジスタの読み出し
  - コマンドとクエリの存在 Query
  - クエリ \*STB?
  - 応答形式 NR1 (整数型)
  - 説明 ステータス・バイト・レジスタの内容を読み出します。  
ここで読み出されるリクエストの要約ビットはMSSです。  
このレジスタとMSSは読み出されてもクリアされません。  
詳細は「4. ステータス・バイト」を参照して下さい。
9. \*TST?
- 機能 セルフテストの結果の問い合わせ
  - コマンドとクエリの存在 Query
  - クエリ \*TST?
  - 応答形式 0 | エラー・コード
  - 説明 \*TST? は本器にセルフテストを実行させ、その結果を応答します。  
0の応答はセルフテストの成功を意味し、それ以外の応答はエラー・コードを意味します。本器の場合、\*TST? に対して0以外の応答をしません。

5.3 共通コマンド

10. \*WAI

- 機能 実行しているすべての動作の終了を待つ
- コマンドとクエリの存在 Command
- コマンド \*WAI
- 説明 \*WAIは現在実行しているすべてのコマンドが終了するのを待ちます。  
このコマンドを実行すると、これ以降のすべてのコマンドは現在実行中のコマンドの終了まで遅延されます。  
\*WAIはDCLインタフェース・メッセージでキャンセルされます。

11. \*OPT?

- 機能 オプションの問い合わせ
- コマンドとクエリの存在 Query
- クエリ \*OPT?
- 応答形式 <opt>,<opt>,...  
<opt> = オプション番号

<opt>	内容
10	PPG(2V) モジュール
11	PPG(3V) モジュール
12	ED モジュール
13	SSG モジュール
70	ジッタ耐力オプション
71	パターンオプション
72	エラー位相解析オプション

- 説明 \*OPT?は本器に実装されているオプションを問い合わせます。



## 5.4 PPG モジュール

### 5.4.1 SOURce[1] サブシステム

#### 1. SOURce[1]:PATtern[:SElect]

- 機能 パターン・タイプの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:PATtern[:SElect] <type>
- パラメータ <type> = {PRBS<n> | ZSUBstitut<n> | PROGram | STM<n> | FLEXible}
- クエリ応答形式 PRBS<n> | ZSUB<n> | PROG | STM<n> | FLEX
- 説明 PPG モジュールの発生パターンのタイプを設定します。

コマンド・パラメータ	パターン・タイプ
PRBS<n>	パターン長 (2 <sup>n</sup> -1) bits の疑似ランダム・パターン (PRBS) <n> = 7, 9, 10, 11, 15, 23, 31
ZSUB<n>	パターン長 2 <sup>n</sup> bits のゼロ置換パターン (ZSUB) <n> = 7, 9, 10, 11, 15
PROG	プログラマブル・パターン (PROG)
STM<n>	STM パターン (STM) <n> = 4, 16 パターンオプションが搭載されている場合に設定可能です。
FLEX	フレキシブル・パターン (FLEX) パターンオプションが搭載されている場合に設定可能です。

5.4.1 SOURce[1] サブシステム

2. SOURce[1]:PATtern:PRBS:MRATio

- 機能 PRBS パターンのマーク率の設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:PATtern:PRBS:MRATio <ratio>
- パラメータ <ratio> = {MRATio<n> | MINVerted4}
- クエリ応答形式 MRAT<n> | MINV4
- 説明 PRBS パターン時のマーク率を設定します。

コマンド・パラメータ	マーク率
MRAT<n>	<n> = 0 マーク率 0/8 <n> = 1 マーク率 1/8 <n> = 2 マーク率 1/4 <n> = 4 マーク率 1/2 <n> = 6 マーク率 3/4 <n> = 7 マーク率 7/8 <n> = 8 マーク率 8/8
MINV4	マーク率 1/2B

3. SOURce[1]:PATtern:ZSUBstitut[:ZLENgth]

- 機能 ゼロ連続ビット長の設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:PATtern:ZSUBstitut[:ZLENgth] <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 ZSUB パターン時のゼロ連続ビット長を設定します。  
 設定範囲  $N \sim (2^N - 1)$  bits (N= 7,9,10,11,15)  
 設定分解能 1 bit

## 4. SOURce[1]:PATtern:PROGram[:LENGth]

- 機能                    PROG パターン長の設定
- コマンドとクエリの存在    Command/Query
- コマンド                SOURce[1]:PATtern:PROGram[:LENGth] <int>
- パラメータ              <int>
- クエリ応答形式        NR1 (整数型)
- 説明                    PROG パターン時のパターン長を設定します。

設定範囲	設定分解能
1 bit ~ 262,144 bits	1 bit
262,146 bits ~ 524,288 bits	2 bits
524,292 bits ~ 1,048,576 bits	4 bits
1,048,584 bits ~ 2,097,152 bits	8 bits
2,097,168 bits ~ 4,194,304 bits	16 bits
4,194,336 bits ~ 8,388,608 bits	32 bits

## 5.4.1 SOURce[1] サブシステム

5.	SOURce[1]:PATtern:PROG:DATA	
機能	文字列データによる PROG パターンの設定	
コマンドとクエリの存在	Command/Query	
コマンド	SOURce[1]:PATtern:PROG:DATA <start_bit>, <length>, <str>	
クエリ	SOURce[1]:PATtern:PROG:DATA? <start_bit>, <length>	
パラメータ	<start_bit> = パターン設定を開始するアドレス 設定範囲      0 bit~ (パターン長 -1) bit 設定分解能   1 bit <length> = 設定するビット数 <str> = 設定するパターン・データ "B-----" パターン・データを 2 進で指定 -----:0, 1 (最大 512 文字) "H-----" パターン・データを 16 進で指定 -----:0~9, a~f, A~F (最大 512 文字)	
クエリ応答形式	<str> = 応答するパターン "H-----" パターン・データを 16 進で応答 -----:0~9, A~F (最大 512 文字)	
説明	<start_bit> から <length> で示されるビット数の PROG パターンを <str> の内容で上書きします。 PROG パターンの範囲を超えるデータは無視されます。 <length> より <str> で示されるデータのビット数が小さい場合は <str> で示されるデータのビット数だけ設定します。 <length> が <str> で示されるデータのビット数より小さい場合は <length> だけ設定します。 クエリの場合、PROG パターンの範囲を超えるデータと、512 文 字で示されるビット数 (152×4=2048 ビット) を超えるデータは 読み出せません。	
使用例	アドレス 0 bit から 16 ビット・データ 4142 (16 進数) を設定す る場合 SOUR1:PATT:PROG:DATA 0, 16, "H4142"	

6. SOURce[1]:PATtern:PROGram:BDATa

- 機能                   ブロック・データによる PROG パターンの設定
- コマンドとクエリの存在   Command/Query
- コマンド               SOURce[1]:PATtern:PROGram:BDATa <start\_bit>, <length>, <block>
- クエリ                 SOURce[1]:PATtern:PROGram:BDATa? <start\_bit>, <length>
- パラメータ           <start\_bit> = パターン設定を開始するアドレス  
                          設定範囲        0 bit~ (パターン長 -1) bit  
                          設定分解能    1 bit  
                          <length> = 設定するビット数  
                          <block> = 設定するパターン・データ
- クエリ応答形式       <block> = 応答するパターン
- 説明                   <start\_bit> から <length> で示されるビット数の PROG パターンを <block> の内容で上書きします。  
                          PROG パターンの範囲を超えるデータは無視されます。  
                          <length> より <block> で示されるデータのビット数が小さい場合は <block> で示されるデータのビット数だけ設定します。  
                          <length> が <block> で示されるデータのビット数より小さい場合は <length> だけ設定します。  
                          クエリの場合、PROG パターンの範囲を超えるデータは読み出せません。
- 使用例                 アドレス 0 bit から 16 ビット・データ 4142 (16 進数) を設定する場合  
                          SOUR1:PATT:PROG:BDAT 0, 16, #12AB

7. SOURce[1]:PATtern:STM:NFRame (パターンオプション)

- 機能                   STM パターンのフレーム数の設定
- コマンドとクエリの存在   Command/Query
- コマンド               SOURce[1]:PATtern:STM:NFRame <int>
- パラメータ           <int>
- クエリ応答形式       NR1 (整数型)
- 説明                   STM パターン時のフレーム数を設定します。

STM パターン	設定範囲
STM4	1 ~ 107
STM16	1 ~ 26

設定分解能 1

5.4.1 SOURce[1] サブシステム

8. SOURce[1]:PATtern:STM:DATA (パターンオプション)

- 機能 文字列データによる STM パターンの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:PATtern:STM:DATA <start\_frm>,<start\_row>,<start\_col>,<length>,<str>
- クエリ SOURce[1]:PATtern:STM:DATA? <start\_frm>,<start\_row>,<start\_col>,<length>
- パラメータ
  - <start\_frm> = パターン設定を開始するフレーム番号  
設定範囲 1 ~ フレーム数  
設定分解能 1
  - <start\_row> = パターン設定を開始する行  
設定範囲 1 ~ 9  
設定分解能 1
  - <start\_col> = パターン設定を開始する列

STM パターン	設定範囲
STM4	1 byte ~ 1080 byte
STM16	1 byte ~ 4320 byte

設定分解能 1 byte

<length> = 設定するビット数

<str> = 設定するパターン・データ

”B-----”パターンデータを 2 進で指定

-----: 0,1 (最大 512 文字)

”H-----”パターン・データを 16 進で指定

-----: 0 ~ 9, a ~ f, A ~ F (最大 512 文字)

- クエリ応答形式 <str> = 応答するパターン  
”H-----”パターン・データを 16 進で応答  
-----: 0 ~ 9, a ~ f, A ~ F (最大 512 文字)

- 説明 <start\_frm>,<start\_row>,<start\_col> から <length> で示されるビット数の STM パターンを <str> の内容で上書きします。STM パターンの範囲を超えるデータは無視されます。<length> より <str> で示されるデータのビット数が小さい場合は <str> で示されるデータのビット数だけ設定します。<length> が <str> で示されるデータのビット数より小さい場合は <length> だけ設定します。  
ペイロードに PRBS パターンを挿入している時は、ペイロード部のビットはパターン・メモリに保存されるだけとなります。クエリの場合、STM パターンの範囲を超えるデータと、512 文字で示されるビット数 (512×4=2048 ビット) を超えるデータは読み出せません。常にパターン・メモリの内容を読み出します。

- 使用例 フレーム番号 1、1 行、1 列から 16 ビット・データ 4142 (16 進数) を設定する場合  
SOUR1:PATT:STM:DATA 1,1,1,16,"H4142"

9. SOURce[1]:PATTern:STM:BDATa (パターンオプション)

- 機能 ブロック・データによる STM パターンの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:PATTern:STM:BDATa <start\_frm>,<start\_row>,<start\_col>,<length>,<block>
- クエリ SOURce[1]:PATTern:STM:BDATa? <start\_frm>,<start\_row>,<start\_col>,<length>
- パラメータ
  - <start\_frm> = パターン設定を開始するフレーム番号  
設定範囲 1 ~ フレーム数  
設定分解能 1
  - <start\_row> = パターン設定を開始する行  
設定範囲 1 ~ 9  
設定分解能 1
  - <start\_col> = パターン設定を開始する列

STM パターン	設定範囲
STM4	1 byte ~ 1080 bytes
STM16	1 byte ~ 4320 bytes

設定分解能 1 byte

<length> = 設定するビット数

<block> = 設定するパターン・データ

- クエリ応答形式 <block> = 応答するパターン
- 説明
  - <start\_frm>,<start\_row>,<start\_col> から <length> で示されるビット数の STM パターンを <block> の内容で上書きします。STM パターンの範囲を超えるデータは無視されます。
  - <length> より <block> で示されるデータのビット数が小さい場合は <block> で示されるデータのビット数だけ設定します。
  - <length> が <block> で示されるデータのビット数より小さい場合は <length> だけ設定します。
  - パイロードに PRBS パターンを挿入しているときは、パイロード部のビットはパターン・メモリに保存されるだけとなります。クエリの場合、STM パターンの範囲を超えるデータは読み出せません。常にパターン・メモリの内容を読み出します。
- 使用例 フレーム番号 1、1 行、1 列から 16 ビット・データ 4142 (16 進数) を設定する場合  
SOUR1:PATT:STM:BDAT 1,1,1,16,#12AB

5.4.1 SOURce[1] サブシステム

10. SOURce[1]:PATtern:STM:PAYLoad:PRBS (パターンオプション)
- 機能 STM パターンのペイロードへの PRBS パターン挿入機能の ON/OFF
  - コマンドとクエリの存在 Command/Query
  - コマンド SOURce[1]:PATtern:STM:PAYLoad:PRBS <bool>
  - パラメータ <bool>
  - クエリ応答形式 0 | 1
  - 説明 STM パターンのペイロードに PRBS パターンを挿入するか (ON)、しないか (OFF) を設定します。
11. SOURce[1]:PATtern:STM:PAYLoad:PRBS:SElect (パターンオプション)
- 機能 STM パターンのペイロードへ挿入する PRBS パターンのパターン長の設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SOURce[1]:PATtern:STM:PAYLoad:PRBS:SElect <type>
  - パラメータ <type> = PRBS<n>  
<n> = 7,9,10,11,15,23,31
  - クエリ応答形式 PRBS<n>  
<n> = 7,9,10,11,15,23,31
  - 説明 STM パターンのペイロードへ挿入する PRBS パターンのパターン長を設定します。



## 12. SOURce[1]:PATtern:STM:PAYLoad:PRBS:MRATio (パターンオプション)

- 機能 STM パターンのペイロードへ挿入する PRBS パターンのマーク率の設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:PATtern:STM:PAYLoad:PRBS:MRATio <ratio>
- パラメータ <ratio> = {MRATio<n> | MINVerted4}
- クエリ応答形式 MRAT<n> | MINV4
- 説明 STM パターンのペイロードへ挿入する PRBS パターンのマーク率を設定します。

コマンド・パラメータ	マーク率
MRAT<n>	<n> = 0   マーク率 0/8
	<n> = 1   マーク率 1/8
	<n> = 2   マーク率 1/4
	<n> = 4   マーク率 1/2
	<n> = 6   マーク率 3/4
	<n> = 7   マーク率 7/8
	<n> = 8   マーク率 8/8
MINV4	マーク率 1/2B

## 13. SOURce[1]:PATtern:STM:SCRamble (パターンオプション)

- 機能 STM パターンのスクランブル機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:PATtern:STM:SCRamble <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 STM パターンに対してスクランブルを実行するか (ON)、しないか (OFF) を設定します。

5.4.1 SOURce[1] サブシステム

14. SOURce[1]:PATtern:STM:B1 (パターンオプション)
- 機能 STM パターンの B1 バイト挿入機能の ON/OFF
  - コマンドとクエリの存在 Command/Query
  - コマンド SOURce[1]:PATtern:STM:B1 <bool>
  - パラメータ <bool>
  - クエリ応答形式 0 | 1
  - 説明 STM パターンに対して B1 バイトを挿入するか (ON)、しないか (OFF) を設定します。
15. SOURce[1]:PATtern:FLEXible:STABLE (パターンオプション)
- 機能 FLEX パターン・シーケンス・テーブルのパターン編集
  - コマンドとクエリの存在 Command/Query
  - コマンド SOURce[1]:PATtern:FLEXible:STABLE <index>,<patt>,<length>,<trig\_signal>
  - クエリ SOURce[1]:PATtern:FLEXible:STABLE? <index>
  - パラメータ
    - <index> = FLEX パターン・シーケンス・テーブル内のパターンを指定するインデックス
      - 設定範囲 1~パターン数
      - 設定分解能 1
    - <patt> = 選択パターン {<prog\_no> | PRBS}
    - <length> = <patt> が PRBS のときのパターン長
      - 設定範囲 128 bits~ 2,097,152 bits
      - 設定分解能 64 bits
    - <trig\_signal> = FLEX Trigger の出力レベル {HIGH | LOW}
  - クエリ応答形式 <patt>,<length>,<trig\_signal>

- 説明 FLEX パターン・シーケンス・テーブルのパターンを編集します。  
<length>の設定は選択パターンがPRBSパターンのときに有効になります。PROG パターンの場合は任意の値を設定して下さい。  
FLEX パターンで使用される PROG パターン長は  
SOUR1:PATT:FLEX:PROG[:LENG]  
で設定されます。  
クエリの場合、選択パターンが PROG パターンのときは <length>  
で PROG パターン長が読み出せます。

<patt>	選択パターン
<prog_no>	PROG パターン PROG パターンの番号を指定します。 設定範囲 1 ~ 127 設定分解能 1
PRBS	PRBS パターン

<trig_signal>	出力レベル
HIGH	HIGH レベル FLEX Trigger において、HIGH レベルの信号 を出力します。
LOW	LOW レベル FLEX Trigger において、LOW レベルの信号 を出力します。

16. SOURce[1]:PATTern:FLEXible:STABLE:COUNT? (パターンオプション)

- 機能 FLEXパターン・シーケンス・テーブルのパターン数の問い合わせ
- コマンドとクエリの存在 Query
- クエリ SOURce[1]:PATTern:FLEXible:STABLE:COUNT?
- クエリ応答形式 NR1 (整数型)
- 説明 FLEX パターン・シーケンス・テーブルのパターン数を問い合わせます。

5.4.1 SOURce[1] サブシステム

17. SOURce[1]:PATtern:FLEXible:STABLE:NEW (パターンオプション)

- 機能 FLEX パターン・シーケンス・テーブルにパターン追加
- コマンドとクエリの存在 Command
- コマンド SOURce[1]:PATtern:FLEXible:STABLE:NEW <index>,<patt>,<length>,<trig\_signal>
- パラメータ
  - <index> = FLEX パターン・シーケンス・テーブル内のパターンを指定するインデックス
    - 設定範囲 1 ~ (パターン数 + 1)
    - 設定分解能 1
  - <patt> = 選択パターン {<prog\_no> | PRBS}
  - <length> = <patt> が PRBS のときのパターン長
    - 設定範囲 128 bits~ 2,097,152 bits
    - 設定分解能 64 bits
  - <trig\_signal> = FLEX Trigger の出力レベル {HIGH | LOW}
- 説明 FLEX パターン・シーケンス・テーブルにパターンを追加します。<index> で指定された位置にパターンを追加します。パターン・シーケンス・テーブルのパターン数は最大 1024 です。パターン数が 1024 の場合、パターンを追加できません。<length> の設定は選択パターンが PRBS パターンのときに有効になります。PROG パターンの場合は任意の値を設定して下さい。FLEX パターンで使用される PROG パターン長は SOUR1:PATT:FLEX:PROG[:LENG] で設定されます。

<patt>	選択パターン
<prog_no>	PROG パターン PROG パターンの番号を指定します。 設定範囲 1 ~ 127 設定分解能 1
PRBS	PRBS パターン

<trig_signal>	出力レベル
HIGH	HIGH レベル FLEX Trigger において、HIGH レベルの信号を出力します。
LOW	LOW レベル FLEX Trigger において、LOW レベルの信号を出力します。

18. SOURce[1]:PATtern:FLEXible:STABle:REMove (パターンオプション)
- 機能 FLEX パターン・シーケンス・テーブルのパターン削除
  - コマンドとクエリの存在 Command
  - コマンド SOURce[1]:PATtern:FLEXible:STABle:REMove <index>
  - パラメータ
    - <index> = FLEX パターン・シーケンス・テーブル内のパターンを指定するインデックス
    - 設定範囲 1 ~ パターン数
    - 設定分解能 1
  - 説明 FLEX パターン・シーケンス・テーブルのパターンを削除します。パターンをすべて削除することはできません。インデックス 2 のパターンが PRBS パターンの場合、インデックス 1 のパターンは削除できません。
19. SOURce[1]:PATtern:FLEXible:STABle:DEFault (パターンオプション)
- 機能 FLEX パターン・シーケンス・テーブルのデフォルト設定
  - コマンドとクエリの存在 Command
  - コマンド SOURce[1]:PATtern:FLEXible:STABle:DEFault
  - 説明 FLEX パターン・シーケンス・テーブルをデフォルトの設定に戻します。
20. SOURce[1]:PATtern:FLEXible:PROGram[:LENGth] (パターンオプション)
- 機能 FLEX パターンで使用される PROG パターン長の設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SOURce[1]:PATtern:FLEXible:PROGram[:LENGth] <prog\_no>, <length>
  - クエリ SOURce[1]:PATtern:FLEXible:PROGram[:LENGth]? <prog\_no>
  - パラメータ
    - <prog\_no> = FLEX パターンで使用される PROG パターンの番号
    - 設定範囲 1 ~ 127
    - 設定分解能 1
    - <length> = PROG パターン長
    - 設定範囲 128 bits ~ 65,536 bits
    - 設定分解能 64 bits
  - クエリ応答形式 <length> = NR1 (整数型)
  - 説明 FLEX パターンで使用される PROG パターン長を設定します。

## 5.4.1 SOURce[1] サブシステム

21.	SOURce[1]:PATtern:FLEXible:PROGram:DATA	(パターンオプション)
機能	文字列データによる FLEX パターンで使用される PROG パターンの設定	
コマンドとクエリの存在	Command/Query	
コマンド	SOURce[1]:PATtern:FLEXible:PROGram:DATA <prog_no>, <start_bit>,<length>,<str>	
クエリ	SOURce[1]:PATtern:FLEXible:PROGram:DATA? <prog_no>, <start_bit>,<length>	
パラメータ	<prog_no> = FLEX パターンで使用される PROG パターンの番号 設定範囲      1 ~ 127 設定分解能    1 <start_bit> = パターン設定を開始するアドレス 設定範囲      0 bit ~ (パターン長 - 1) bit 設定分解能    1 bit <length> = 設定するビット数 <str> = 設定するパターン・データ "B-----" パターン・データを 2 進で指定 -----: 0,1 (最大 512 文字) "H-----" パターン・データを 16 進で指定 -----: 0 ~ 9, a ~ f, A ~ F (最大 512 文字)	
クエリ応答形式	<str> = 応答するパターン "H-----" パターン・データを 16 進で応答 -----: 0 ~ 9, a ~ f, A ~ F (最大 512 文字)	
説明	<prog_no> で指定された PROG パターンを、<start_bit> から <length> で示されるビット数だけ <str> の内容で上書きします。 <prog_no> で指定された PROG パターンの範囲を超えるデータは無視されます。 <length> より <str> で示されるデータのビット数が小さい場合は <str> で示されるデータのビット数だけ設定します。 <length> が <str> で示されるデータのビット数より小さい場合は <length> だけ設定します。 クエリの場合、<prog_no> で指定された PROG パターンの範囲を超えるデータと、512 文字で示されるビット数(512×4=2048 ビット)を超えるデータは読み出せません。	
使用例	PROG パターン番号 1 のアドレス 0 bit から 16 ビット・データ 4142 (16 進数) を設定する場合 SOUR1:PATT:FLEX:PROG:DATA 1,0,16,"H4142"	

22.	SOURce[1]:PATTern:FLEXible:PROGram:BDATa	(パターンオプション)
• 機能	ブロック・データによる FLEX パターンで使用される PROG パターンの設定	
• コマンドとクエリの存在	Command/Query	
• コマンド	SOURce[1]:PATTern:FLEXible:PROGram:BDATa <prog_no>, <start_bit>,<length>,<block>	
• クエリ	SOURce[1]:PATTern:FLEXible:PROGram:BDATa? <prog_no>, <start_bit>,<length>	
• パラメータ	<prog_no> = FLEX パターンで使用される PROG パターンの番号 設定範囲 1 ~ 127 設定分解能 1 <start_bit> = パターン設定を開始するアドレス 設定範囲 0 bit ~ (パターン長 - 1) bit 設定分解能 1 bit <length> = 設定するビット数 <block> = 設定するパターン・データ	
• クエリ応答形式	<block> = 応答するパターン	
• 説明	<prog_no> で指定された PROG パターンを、<start_bit> から <length>で示されるビット数だけ<block>の内容で上書きします。 <prog_no> で指定されたPROGパターンの範囲を超えるデータは無視されます。 <length> より <block> で示されるデータのビット数が小さい場合は <block> で示されるデータのビット数だけ設定します。 <length> が <block> で示されるデータのビット数より小さい場合は <length> だけ設定します。 クエリの場合、<prog_no> で指定された PROG パターンの範囲を超えるデータは読み出せません。	
• 使用例	PROG パターン番号 1 のアドレス 0 bit から 16 ビット・データ 4142 (16 進数) を設定する場合 SOUR1:PATT:FLEX:PROG:BDAT 1,0,16,#12AB	

5.4.1 SOURce[1] サブシステム

23. SOURce[1]:PATtern:FLEXible:PRBS:SElect (パターンオプション)
- 機能 FLEX パターンで使用される PRBS パターンのパターン長の設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SOURce[1]:PATtern:FLEXible:PRBS:SElect <type>
  - パラメータ <type> = PRBS<n>  
<n> = 7,9,10,11,15,23,31
  - クエリ応答形式 PRBS<n>  
<n> = 7,9,10,11,15,23,31
  - 説明 FLEX パターンで使用される PRBS パターンのパターン長を設定します。

24. SOURce[1]:PATtern:FLEXible:PRBS:MRATio (パターンオプション)
- 機能 FLEX パターンで使用される PRBS パターンのマーク率の設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SOURce[1]:PATtern:FLEXible:PRBS:MRATio <ratio>
  - パラメータ <ratio> = {MRATio<n> | MINVerted4}
  - クエリ応答形式 MRAT<n> | MINV4
  - 説明 FLEX パターンで使用される PRBS パターンのマーク率を設定します。

コマンド・パラメータ	マーク率
MRAT<n>	<n> = 0    マーク率 0/8
	<n> = 1    マーク率 1/8
	<n> = 2    マーク率 1/4
	<n> = 4    マーク率 1/2
	<n> = 6    マーク率 3/4
	<n> = 7    マーク率 7/8
	<n> = 8    マーク率 8/8
MINV4	マーク率 1/2B



## 25. SOURce[1]:PATtern:POLarity

- 機能 パターン極性の設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:PATtern:POLarity <type>
- パラメータ <type> = {NORMal | INVerted}
- クエリ応答形式 NORM | INV
- 説明 パターン極性を設定します。

コマンド・パラメータ	パターン極性
NORM	正論理
INV	負論理

## 26. SOURce[1]:EADDITION

- 機能 エラー付加の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:EADDITION <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 エラー付加を行うか (ON)、行わないか (OFF) を設定します。

## 27. SOURce[1]:EADDITION:SINGLE

- 機能 1 ビット・エラーの付加
- コマンドとクエリの存在 Command
- コマンド SOURce[1]:EADDITION:SINGLE
- 説明 1 ビットのエラーを付加します。  
エラー付加が ON で、エラー付加モードがシングル・モードの場合のみ有効です。

5.4.1 SOURce[1] サブシステム

28. SOURce[1]:EADDITION:MODE

- 機能 エラー付加モードの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:EADDITION:MODE <type>
- パラメータ <type> = {SINGle | REPeat | EXTernal}
- クエリ応答形式 SING | REP | EXT
- 説明 エラー付加モードを設定します。

コマンド・パラメータ	エラー付加モード
SING	シングル・モード。 手動で1ビットのエラーを付加します。
REP	リピート・モード。 エラー付加レートのエラーを付加します。
EXT	外部制御モード。 外部制御によりエラーを付加します。

29. SOURce[1]:EADDITION:RATE

- 機能 エラー付加レートの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:EADDITION:RATE <real>
- パラメータ <real> = 1E-2, 1E-3, 1E-4, 1E-5, 1E-6, 1E-7, 1E-8, 1E-9
- クエリ応答形式 NR3 (浮動小数点型)
- 説明 エラー付加するレートを設定します。  
エラー付加モードがリピート・モードのときのみ有効です。

30. SOURce[1]:EADDITION:ROUTE

- 機能 エラー付加ルートの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce[1]:EADDITION:ROUTE <int>
- パラメータ <int> = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16
- クエリ応答形式 NR1 (整数型)
- 説明 エラーを付加するルートを設定します。

## 5.4.2 OUTPut[1] サブシステム

### 1. OUTPut[1][:STATe]

- 機能 DATA 出力の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut[1][:STATe] <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 DATA 出力を有効にするか (ON)、無効にするか (OFF) を設定します。

### 2. OUTPut[1]:VOLTagE[:LEVel][:IMMediate][:AMPLitude]

- 機能 DATA 出力振幅の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut[1]:VOLTagE[:LEVel][:IMMediate][:AMPLitude] <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 出力の振幅を設定します。

終端タイプ	設定範囲
LVPECL (+1.3V)	0.60V <sub>P-P</sub> ~ 1.00V <sub>P-P</sub>
to GND (0V)	0.30V <sub>P-P</sub> ~ 2.00V <sub>P-P</sub> (2V 出力モジュール) 0.30V <sub>P-P</sub> ~ 3.00V <sub>P-P</sub> (3V 出力モジュール)
ECL (-2V)	0.60V <sub>P-P</sub> ~ 1.00V <sub>P-P</sub>
CML	0.30V <sub>P-P</sub> ~ 1.00V <sub>P-P</sub>

設定分解能 0.01V<sub>P-P</sub>

設定単位 V<sub>P-P</sub>

5.4.2 OUTPut[1] サブシステム

3. OUTPut[1]:VOLTage[:LEVel][:IMMediate]:HIGH

- 機能 DATA 出力の HIGH レベル・オフセット値の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut[1]:VOLTage[:LEVel][:IMMediate]:HIGH <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 出力のオフセット値を HIGH レベルとして設定します。

終端タイプ	設定範囲
LVPECL (+1.3V)	+2.30V ~ +2.70V
to GND (0V)	-2.00V ~ +2.00V*1
ECL (-2V)	-1.00V ~ -0.60V
CML	(Vcc-0.20)V ~ (Vcc+0.20)V*2

\*1: 3V 出力モジュールにて 2V<sub>p,p</sub> を超える振幅設定時には、オフセット設定範囲は -1.00V ~ +1.00V に制限されます。

\*2: Vcc は DATA 出力終端電圧です。

設定分解能 0.01V

設定単位 V

4. OUTPut[1]:VOLTage[:LEVel][:IMMediate]:MIDDLE

- 機能 DATA 出力の中間レベル・オフセット値の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut[1]:VOLTage[:LEVel][:IMMediate]:MIDDLE <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 出力のオフセット値を HIGH と LOW レベルの中間値として設定します。  
HIGH レベルのオフセット値が範囲内に入る値を設定して下さい。

設定分解能 0.01V

設定単位 V

5. OUTPut[1]:VOLTage[:LEVel][:IMMediate]:LOW

- 機能 DATA 出力の LOW レベル・オフセット値の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut[1]:VOLTage[:LEVel][:IMMediate]:LOW <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 出力のオフセット値を LOW レベルとして設定します。  
HIGH レベルのオフセット値が範囲内に入る値を設定して下さい。  
設定分解能 0.01V  
設定単位 V

6. OUTPut[1]:CROSSs

- 機能 DATA 出力のクロスポイントの設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut[1]:CROSSs <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 DATA 出力のクロスポイントを設定します。  
設定範囲 20% ~ 80%  
設定分解能 1%

7. OUTPut[1]:TERMination

- 機能 DATA 出力終端タイプの設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut[1]:TERMination <type>
- パラメータ <type> = {P1R3V | GND | M2V | CML}
- クエリ応答形式 P1R3V | GND | M2V | CML
- 説明 DATA 出力の終端タイプを設定します。

コマンド・パラメータ	終端タイプ
P1R3V	LVPECL (+1.3V)
GND	to GND (0V)
M2V	ECL (-2V)
CML	CML

5.4.3 OUTPut3 サブシステム

8. OUTPut[1]:TERMination:VOLTagE

- 機能 DATA 出力終端電圧の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut[1]:TERMination:VOLTagE <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 出力の終端電圧を設定します。  
終端タイプが CML の場合のみ有効です。  
設定範囲 0.00V ~ 3.50V  
設定分解能 0.05V  
設定単位 V

5.4.3 OUTPut3 サブシステム

1. OUTPut3[:STATe]

- 機能 DATA 出力の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut3[:STATe] <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 DATA 出力を有効にするか (ON)、無効にするか (OFF) を設定します。

2. `OUTPut3:VOLtage[:LEVel][:IMMediate][:AMPLitude]`

- 機能 DATA 出力振幅の設定
- コマンドとクエリの存在 Command/Query
- コマンド `OUTPut3:VOLtage[:LEVel][:IMMediate][:AMPLitude] <real>`
- パラメータ `<real>`
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 出力の振幅を設定します。

終端タイプ	設定範囲
LVPECL (+1.3V)	0.60V <sub>p-p</sub> ~ 1.00V <sub>p-p</sub>
to GND (0V)	0.30V <sub>p-p</sub> ~ 2.00V <sub>p-p</sub> (2V 出力モジュール) 0.30V <sub>p-p</sub> ~ 3.00V <sub>p-p</sub> (3V 出力モジュール)
ECL (-2V)	0.60V <sub>p-p</sub> ~ 1.00V <sub>p-p</sub>
CML	0.30V <sub>p-p</sub> ~ 1.00V <sub>p-p</sub>

設定分解能 0.01V<sub>p-p</sub>設定単位 V<sub>p-p</sub>3. `OUTPut3:VOLtage[:LEVel][:IMMediate]:HIGH`

- 機能 DATA 出力の HIGH レベル・オフセット値の設定
- コマンドとクエリの存在 Command/Query
- コマンド `OUTPut3:VOLtage[:LEVel][:IMMediate]:HIGH <real>`
- パラメータ `<real>`
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 出力のオフセット値を HIGH レベルとして設定します。

終端タイプ	設定範囲
LVPECL (+1.3V)	+2.30V ~ +2.70V
to GND (0V)	-2.00V ~ +2.00V <sup>*1</sup>
ECL (-2V)	-1.00V ~ -0.60V
CML	(V <sub>cc</sub> -0.20)V ~ (V <sub>cc</sub> +0.20)V <sup>*2</sup>

\*1: 3V 出力モジュールにて 2V<sub>p-p</sub> を超える振幅設定時には、オフセット設定範囲は -1.00V ~ +1.00V に制限されます。

\*2: V<sub>cc</sub> は DATA 出力終端電圧です。

設定分解能 0.01V

設定単位 V

## 5.4.3 OUTPut3 サブシステム

## 4. OUTPut3:VOLTage[:LEVel][:IMMediate]:MIDDLE

- 機能 DATA 出力の中間レベル・オフセット値の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut3:VOLTage[:LEVel][:IMMediate]:MIDDLE <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 出力のオフセット値を HIGH と LOW レベルの中間値として設定します。  
HIGH レベルのオフセット値が範囲内に入る値を設定して下さい。  
設定分解能 0.01V  
設定単位 V

## 5. OUTPut3:VOLTage[:LEVel][:IMMediate]:LOW

- 機能 DATA 出力の LOW レベル・オフセット値の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut3:VOLTage[:LEVel][:IMMediate]:LOW <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 出力のオフセット値を LOW レベルとして設定します。  
HIGH レベルのオフセット値が範囲内に入る値を設定して下さい。  
設定分解能 0.01V  
設定単位 V

## 6. OUTPut3:CROSSs

- 機能 DATA 出力のクロスポイントの設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut3:CROSSs <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 DATA 出力のクロスポイントを設定します。  
設定範囲 20% ~ 80%  
設定分解能 1%



## 7. OUTPut3:TERMination

- 機能 DATA 出力終端タイプの設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut3:TERMination <type>
- パラメータ <type> = {P1R3V | GND | M2V | CML}
- クエリ応答形式 P1R3V | GND | M2V | CML
- 説明 DATA 出力の終端タイプを設定します。

コマンド・パラメータ	終端タイプ
P1R3V	LVPECL (+1.3V)
GND	to GND (0V)
M2V	ECL (-2V)
CML	CML

## 8. OUTPut3:TERMination:VOLTagE

- 機能 DATA 出力終端電圧の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut3:TERMination:VOLTagE <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 出力の終端電圧を設定します。  
終端タイプが CML の場合のみ有効です。  
設定範囲 0.00V ~ 3.50V  
設定分解能 0.05V  
設定単位 V

## 9. OUTPut3:TRACk

- 機能 データ出力のトラッキング機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut3:TRACk <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 データ出力のトラッキングを行うか (ON)、行わないか (OFF) を設定します。  
ON にすると、DATA 出力は DATA 出力と同じ設定になります。

## 5.4.4 SOURce2 サブシステム

1. SOURce2:DElay

- 機能 クロック遅延量を設定します。
- コマンドとクエリの存在 Command/Query
- コマンド SOURce2:DElay <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 クロック遅延量を設定します。  
設定範囲 -1000ps ~ 1000ps  
設定分解能 1ps  
設定単位 ps

## 5.4.5 OUTPut2 サブシステム

### 1. OUTPut2[:STATe]

- 機能                   CLOCK 出力の ON/OFF
- コマンドとクエリの存在   Command/Query
- コマンド                OUTPut2[:STATe] <bool>
- パラメータ             <bool>
- クエリ応答形式         0 | 1
- 説明                    CLOCK 出力を有効にするか (ON)、無効にするか (OFF) を設定します。

### 2. OUTPut2:VOLTagE[:LEVel][:IMMediate][:AMPLitude]

- 機能                    CLOCK 出力振幅の設定
- コマンドとクエリの存在   Command/Query
- コマンド                OUTPut2:VOLTagE[:LEVel][:IMMediate][:AMPLitude] <real>
- パラメータ             <real>
- クエリ応答形式         NR2 (固定小数点型)
- 説明                    CLOCK 出力の振幅を設定します。

終端タイプ	設定範囲
LVPECL (+1.3V)	0.60V <sub>P-P</sub> ~ 1.00V <sub>P-P</sub>
to GND (0V)	0.30V <sub>P-P</sub> ~ 2.00V <sub>P-P</sub>
ECL (-2V)	0.60V <sub>P-P</sub> ~ 1.00V <sub>P-P</sub>
CML	0.30V <sub>P-P</sub> ~ 1.00V <sub>P-P</sub>

設定分解能 0.01V<sub>P-P</sub>

設定単位 V<sub>P-P</sub>

## 5.4.5 OUTPut2 サブシステム

## 3. OUTPut2:VOLtage[:LEVel][:IMMediate]:HIGH

- 機能                   CLOCK 出力の HIGH レベル・オフセット値の設定
- コマンドとクエリの存在   Command/Query
- コマンド                OUTPut2:VOLtage[:LEVel][:IMMediate]:HIGH <real>
- パラメータ             <real>
- クエリ応答形式         NR2 (固定小数点型)
- 説明                    CLOCK 出力のオフセット値を HIGH レベルとして設定します。

終端タイプ	設定範囲
LVPECL (+1.3V)	+2.30V ~ +2.70V
to GND (0V)	-2.00V ~ +2.00V
ECL (-2V)	-1.00V ~ -0.60V
CML	(Vcc-0.20)V ~ (Vcc+0.20)V*1

\*1: Vcc は CLOCK 出力終端電圧です。

設定分解能 0.01V

設定単位 V

## 4. OUTPut2:VOLtage[:LEVel][:IMMediate]:MIDDLE

- 機能                   CLOCK 出力の中間レベル・オフセット値の設定
- コマンドとクエリの存在   Command/Query
- コマンド                OUTPut2:VOLtage[:LEVel][:IMMediate]:MIDDLE <real>
- パラメータ             <real>
- クエリ応答形式         NR2 (固定小数点型)
- 説明                    CLOCK 出力のオフセット値を HIGH と LOW レベルの中間値として設定します。  
HIGH レベルのオフセット値が範囲内に入る値を設定して下さい。  
設定分解能 0.01V  
設定単位 V

## 5. OUTPut2:VOLtage[:LEVel][:IMMediate]:LOW

- 機能 CLOCK 出力の LOW レベル・オフセット値の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut2:VOLtage[:LEVel][:IMMediate]:LOW <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 CLOCK 出力のオフセット値を LOW レベルとして設定します。  
HIGH レベルのオフセット値が範囲内に入る値を設定して下さい。  
設定分解能 0.01V  
設定単位 V

## 6. OUTPut2:TERMination

- 機能 CLOCK 出力終端タイプの設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut2:TERMination <type>
- パラメータ <type> = {P1R3V | GND | M2V | CML}
- クエリ応答形式 P1R3V | GND | M2V | CML
- 説明 CLOCK 出力の終端タイプを設定します。

コマンド・パラメータ	終端タイプ
P1R3V	LVPECL (+1.3V)
GND	to GND (0V)
M2V	ECL (-2V)
CML	CML

5.4.5 OUTPut2 サブシステム

7. OUTPut2:TERMination:VOLTag

- 機能                   CLOCK 出力終端電圧の設定
- コマンドとクエリの存在   Command/Query
- コマンド                OUTPut2:TERMination:VOLTag <real>
- パラメータ             <real>
- クエリ応答形式         NR2 (固定小数点型)
- 説明                    CLOCK 出力の終端電圧を設定します。  
                          終端タイプが CML の場合のみ有効です。  
                          設定範囲     0.00V ~ 3.50V  
                          設定分解能   0.05V  
                          設定単位     V

## 5.4.6 OUTPut4 サブシステム

### 1. OUTPut4[:STATe]

- 機能  $\overline{\text{CLOCK}}$  出力の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut4[:STATe] <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明  $\overline{\text{CLOCK}}$  出力を有効にするか (ON)、無効にするか (OFF) を設定します。

### 2. OUTPut4:VOLTagE[:LEVel][:IMMediate][:AMPLitude]

- 機能  $\overline{\text{CLOCK}}$  出力振幅の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut4:VOLTagE[:LEVel][:IMMediate][:AMPLitude] <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明  $\overline{\text{CLOCK}}$  出力の振幅を設定します。

終端タイプ	設定範囲
LVPECL (+1.3V)	0.60V <sub>P-P</sub> ~ 1.00V <sub>P-P</sub>
to GND (0V)	0.30V <sub>P-P</sub> ~ 2.00V <sub>P-P</sub>
ECL (-2V)	0.60V <sub>P-P</sub> ~ 1.00V <sub>P-P</sub>
CML	0.30V <sub>P-P</sub> ~ 1.00V <sub>P-P</sub>

設定分解能 0.01V<sub>P-P</sub>

設定単位 V<sub>P-P</sub>

5.4.6 OUTPut4 サブシステム

3. OUTPut4:VOLtage[:LEVel][:IMMediate]:HIGH

- 機能 CLOCK 出力の HIGH レベル・オフセット値の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut4:VOLtage[:LEVel][:IMMediate]:HIGH <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 CLOCK 出力のオフセット値を HIGH レベルとして設定します。

終端タイプ	設定範囲
LVPECL (+1.3V)	+2.30V ~ +2.70V
to GND (0V)	-2.00V ~ +2.00V
ECL (-2V)	-1.00V ~ -0.60V
CML	(Vcc-0.20)V ~ (Vcc+0.20)V*1

\*1: Vcc は CLOCK 出力終端電圧です。

設定分解能 0.01V

設定単位 V

4. OUTPut4:VOLtage[:LEVel][:IMMediate]:MIDDLE

- 機能 CLOCK 出力の中間レベル・オフセット値の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut4:VOLtage[:LEVel][:IMMediate]:MIDDLE <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 CLOCK 出力のオフセット値を HIGH と LOW レベルの中間値として設定します。  
HIGH レベルのオフセット値が範囲内に入る値を設定して下さい。  
設定分解能 0.01V  
設定単位 V



5. OUTPut4:VOLtage[:LEVel][:IMMediate]:LOW

- 機能 CLOCK 出力の LOW レベル・オフセット値の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut4:VOLtage[:LEVel][:IMMediate]:LOW <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 CLOCK 出力のオフセット値を LOW レベルとして設定します。  
HIGH レベルのオフセット値が範囲内に入る値を設定して下さい。  
設定分解能 0.01V  
設定単位 V

6. OUTPut4:TERMination

- 機能 CLOCK 出力終端タイプの設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut4:TERMination <type>
- パラメータ <type> = {P1R3V | GND | M2V | CML}
- クエリ応答形式 P1R3V | GND | M2V | CML
- 説明 CLOCK 出力の終端タイプを設定します。

コマンド・パラメータ	終端タイプ
P1R3V	LVPECL (+1.3V)
GND	to GND (0V)
M2V	ECL (-2V)
CML	CML

## 5.4.6 OUTPut4 サブシステム

## 7. OUTPut4:TERMination:VOLTagE

- 機能 CLOCK 出力終端電圧の設定
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut4:TERMination:VOLTagE <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 CLOCK 出力の終端電圧を設定します。  
終端タイプが CML の場合のみ有効です。  
設定範囲 0.00V ~ 3.50V  
設定分解能 0.05V  
設定単位 V

## 8. OUTPut4:TRACk

- 機能 クロック出力のトラッキング機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド OUTPut4:TRACk <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 クロック出力のトラッキングを行うか (ON)、行わないか (OFF) を設定します。  
ONにすると、CLOCK出力はCLOCK出力と同じ設定になります。

## 5.4.7 SOURce5 サブシステム

### 1. SOURce5:BOUtput[:STATe]

- 機能 バースト・モードの ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SOURce5:BOUtput[:STATe] <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 バースト・モードを有効にするか (ON)、無効にするか (OFF) を設定します。

### 2. SOURce5:BOUtput:SOURce

- 機能 BURST OUTPUT 信号源の設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce5:BOUtput:SOURce <type>
- パラメータ <type> = {INTernal | EXTernal}
- クエリ応答形式 INT | EXT
- 説明 BURST OUTPUT の信号源を設定します。

コマンド・パラメータ	信号源
INT	内部信号
EXT	外部入力

### 3. SOURce5:BOUtput:CYCLe

- 機能 バースト周期の設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce5:BOUtput:CYCLe <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明
  - バースト周期を設定します。
  - バースト周期には、OFF 時間より大きい値を設定して下さい。
  - 設定範囲 2 $\mu$ s ~ 50000 $\mu$ s
  - 設定分解能 1 $\mu$ s
  - 設定単位  $\mu$ s

5.4.7 SOURce5 サブシステム

4. SOURce5:BOUtpuT:OTIME

- 機能                                      バースト周期中の OFF 時間の設定
- コマンドとクエリの存在      Command/Query
- コマンド                                SOURce5:BOUtpuT:OTIME <int>
- パラメータ                              <int>
- クエリ応答形式                      NR1 (整数型)
- 説明                                      バースト周期中の OFF 時間を設定します。  
OFF 時間には、バースト周期より小さい値を設定して下さい。  
設定範囲      1 $\mu$ s ~ 49999 $\mu$ s  
設定分解能    1 $\mu$ s  
設定単位       $\mu$ s

## 5.4.8 SOURce6 サブシステム

### 1. SOURce6:TRIGger[:MODE]

- 機能 トリガ出力ソースの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce6:TRIGger[:MODE] <type>
- パラメータ <type> = {CLOCk8 | CLOCk32 | PATTem | FLEXible}
- クエリ応答形式 CLOC8 | CLOC32 | PATT | FLEX
- 説明 トリガ出力のソースを設定します。

コマンド・パラメータ	トリガ出力のソース
CLOC8	出力クロックの 1/8 周波数のクロック信号
CLOC32	出力クロックの 1/32 周波数のクロック信号
PATT	パターンに同期したトリガ出力
FLEX	パターン・シーケンス・テーブルの FLEX Trigger で設定したレベル パターンオプションが搭載され、パターンタイプが FLEX の場合に設定可能です。

### 2. SOURce6:TRIGger:PRBS:POSition

- 機能 PRBS パターン時のトリガ信号出力ポジションの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce6:TRIGger:PRBS:POSition <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 PRBS パターン時のトリガ信号出力ポジションを設定します。  
設定範囲 0 bit~ (パターン長 -1) bit  
設定分解能 1 bit

## 5.4.8 SOURce6 サブシステム

## 3. SOURce6:TRIGger:ZSUBstitut:POSition

- 機能 ZSUB パターン時のトリガ信号出力ポジションの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce6:TRIGger:ZSUBstitut:POSition <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 ZSUB パターン時のトリガ信号出力ポジションを設定します。  
設定範囲 0 bit~ (パターン長 -1) bit  
設定分解能 16 bits

## 4. SOURce6:TRIGger:PROGram:POSition

- 機能 PROG パターン時のトリガ信号出力ポジションの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce6:TRIGger:PROGram:POSition <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 PROG パターン時のトリガ信号出力ポジションを設定します。  
設定範囲 0 bit~ (パターン長 -1) bit  
設定分解能 16 bits

## 5. SOURce6:TRIGger:STM:FNUMBER (パターンオプション)

- 機能 STM パターン時のトリガ出力フレーム番号の設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce6:TRIGger:STM:FNUMBER <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 STM パターン時のトリガ出力フレーム番号を設定します。  
設定範囲 1 ~ フレーム数  
設定分解能 1

## 6. SOURce6:TRIGger:STM:ROW (パターンオプション)

- 機能 STM パターン時のトリガ出力行の設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce6:TRIGger:STM:ROW <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 STM パターン時のトリガ出力行を設定します。  
設定範囲 1 ~ 9  
設定分解能 1

## 7. SOURce6:TRIGger:STM:COLumn (パターンオプション)

- 機能 STM パターン時のトリガ出力列の設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce6:TRIGger:STM:COLumn <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 STM パターン時のトリガ出力列を設定します。

STM パターン	設定範囲
STM4	1 byte ~ 1080 byte
STM16	1 byte ~ 4320 byte

設定分解能 2 bytes

5.4.8 SOURce6 サブシステム

8. SOURce6:TRIGger:STM:MODE (パターンオプション)

- 機能 STM パターン時のトリガ出力モードの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce6:TRIGger:STM:MODE <type>
- パラメータ <type> = {PATTern | FRAMe}
- クエリ応答形式 PATT | FRAM
- 説明 STM パターン時のトリガ出力モードを設定します。

コマンド・パラメータ	STM パターントリガ出力モード
PATT	パターン・モード パターン毎にトリガ出力を行います。
FRAM	フレーム・モード フレーム毎にトリガ出力を行います。

9. SOURce6:TRIGger:FLEXible:INDEX (パターンオプション)

- 機能 FLEX パターン時のトリガ出力インデックスの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce6:TRIGger:FLEXible:INDEX <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 FLEX パターン時のトリガ出力インデックスを設定します。  
設定範囲 1 ~ パターン数  
設定分解能 1

10. SOURce6:TRIGger:FLEXible:ADDRESS (パターンオプション)

- 機能 FLEX パターン時のトリガ出力アドレスの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce6:TRIGger:FLEXible:ADDRESS <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 FLEX パターン時のトリガ出力アドレスを設定します。  
設定範囲 0 bit ~ (トリガ出力インデックスの  
パターン長 - 1) bit  
設定分解能 16 bits



#### 5.4.9 SOURce7 サブシステム

現バージョンでは、本サブシステム用のコマンドとクエリはありません。

## 5.5 ED モジュール

## 5.5 ED モジュール

## 5.5.1 SENSE[1] サブシステム

## 1. SENSE[1]:PATtern[:SElect]

- 機能 パターン・タイプの設定
- コマンドとクエリが存在 Command/Query
- コマンド SENSE[1]:PATtern[:SElect] <type>
- パラメータ <type> = {PRBS<n> | ZSUBstitut<n> | PROGram | STM<n> | FLEXible}
- クエリ応答形式 PRBS<n> | ZSUB<n> | PROG | STM<n> | FLEX
- 説明 ED モジュールの測定パターンのタイプを設定します。

コマンド・パラメータ	パターン・タイプ
PRBS<n>	パターン長 (2 <sup>n</sup> -1) bits の疑似ランダム・パターン (PRBS) <n> = 7, 9, 10, 11, 15, 23, 31
ZSUB<n>	パターン長 2 <sup>n</sup> bits のゼロ置換パターン (ZSUB) <n> = 7, 9, 10, 11, 15
PROG	プログラマブル・パターン (PROG)
STM<n>	STM パターン (STM) <n> = 4, 16 パターンオプションが搭載されている場合に設定可能です。
FLEX	フレキシブル・パターン (FLEX) パターンオプションが搭載されている場合に設定可能です。

## 2. SENSE[1]:PATTERN:PRBS:MRATIO

- 機能 PRBS パターンのマーク率の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:PATTERN:PRBS:MRATIO <ratio>
- パラメータ <ratio> = {MRATIO<n> | MINVERTED4}
- クエリ応答形式 MRAT<n> | MINV4
- 説明 PRBS パターン時のマーク率を設定します。

コマンド・パラメータ	マーク率
MRAT<n>	<n> = 0 マーク率 0/8
	<n> = 1 マーク率 1/8
	<n> = 2 マーク率 1/4
	<n> = 4 マーク率 1/2
	<n> = 6 マーク率 3/4
	<n> = 7 マーク率 7/8
	<n> = 8 マーク率 8/8
MINV4	マーク率 1/2B

## 3. SENSE[1]:PATTERN:ZSUBSTITUT[:ZLENGTH]

- 機能 ゼロ連続ビット長の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:PATTERN:ZSUBSTITUT[:ZLENGTH] <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 ZSUB パターン時のゼロ連続ビット長を設定します。  
設定範囲  $N \sim (2^N - 1)$  bits (N=7,9,10,11,15)  
設定分解能 1 bit

5.5.1 SENSE[1] サブシステム

4. SENSE[1]:PATtern:PROGram[:LENGth]

- 機能                                   PROG パターン長の設定
- コマンドとクエリの存在   Command/Query
- コマンド                            SENSe[1]:PATtern:PROGram[:LENGth] <int>
- パラメータ                        <int>
- クエリ応答形式                NR1 (整数型)
- 説明                                 PROG パターン時のパターン長を設定します。

設定範囲	設定分解能
1 bit ~ 262,144 bits	1 bit
262,146 bits ~ 524,288 bits	2 bits
524,292 bits ~ 1,048,576 bits	4 bits
1,048,584 bits ~ 2,097,152 bits	8 bits
2,097,168 bits ~ 4,194,304 bits	16 bits
4,194,336 bits ~ 8,388,608 bits	32 bits

## 5. SENSE[1]:PATTERN:PROGRAM:DATA

- 機能 文字列データによる PROG パターンの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:PATTERN:PROGRAM:DATA <start\_bit>, <length>, <str>
- クエリ SENSE[1]:PATTERN:PROGRAM:DATA? <start\_bit>, <length>
- パラメータ
  - <start\_bit> = パターン設定を開始するアドレス
  - 設定範囲 0 bit~ (パターン長-1) bit
  - 設定分解能 1 bit
  - <length> = 設定するビット数
  - <str> = 設定するパターン・データ
    - "B-----" パターン・データを 2 進で指定
    - :0, 1 (最大 512 文字)
    - "H-----" パターン・データを 16 進で指定
    - :0~9, a~f, A~F (最大 512 文字)
- クエリ応答形式 <str> = 応答するパターン
  - "H-----" パターン・データを 16 進で応答
  - :0~9, A~F (最大 512 文字)
- 説明
  - <start\_bit> から <length> で示されるビット数の PROG パターンを <str> の内容で上書きします。
  - PROG パターンの範囲を超えるデータは無視されます。
  - <length> より <str> で示されるデータのビット数が小さい場合は <str> で示されるデータのビット数だけ設定します。
  - <length> が <str> で示されるデータのビット数より小さい場合は <length> だけ設定します。
  - クエリの場合、PROG パターンの範囲を超えるデータと、512 文字で示されるビット数 (512×4=2048 ビット) を超えるデータは読み出せません。
- 使用例 アドレス 0 bit から 16 ビット・データ 4142 (16 進数) を設定する場合  
SENS1:PATT:PROG:DATA 0, 16, "H4142"

5.5.1 SENSE[1] サブシステム

6. SENSE[1]:PATtern:PROGram:BDATa
- ・ 機能 ブロック・データによる PROG パターンの設定
  - ・ コマンドとクエリの存在 Command/Query
  - ・ コマンド SENSE[1]:PATtern:PROGram:BDATa <start\_bit>, <length>, <block>
  - ・ クエリ SENSE[1]:PATtern:PROGram:BDATa? <start\_bit>, <length>
  - ・ パラメータ
    - <start\_bit> = パターン設定を開始するアドレス
    - 設定範囲 0 bit~ (パターン長 -1) bit
    - 設定分解能 1 bit
    - <length> = 設定するビット数
    - <block> = 設定するパターン・データ
  - ・ クエリ応答形式 <block> = 応答するパターン
  - ・ 説明
    - <start\_bit> から <length> で示されるビット数の PROG パターンを <block> の内容で上書きします。
    - PROG パターンの範囲を超えるデータは無視されます。
    - <length> より <block> で示されるデータのビット数が小さい場合は <block> で示されるデータのビット数だけ設定します。
    - <length> が <block> で示されるデータのビット数より小さい場合は <length> だけ設定します。
    - クエリの場合、PROG パターンの範囲を超えるデータは読み出せません。
  - ・ 使用例 アドレス 0 bit から 16 ビット・データ 4142 (16 進数) を設定する場合  
SENS1:PATT:PROG:BDAT 0, 16, #12AB
7. SENSE[1]:PATtern:STM:NFRame (パターンオプション)
- ・ 機能 STM パターンのフレーム数の設定
  - ・ コマンドとクエリの存在 Command/Query
  - ・ コマンド SENSE[1]:PATtern:STM:NFRame <int>
  - ・ パラメータ <int>
  - ・ クエリ応答形式 NR1 (整数型)
  - ・ 説明 STM パターン時のフレーム数を設定します。
- | STM パターン | 設定範囲    |
|----------|---------|
| STM4     | 1 ~ 107 |
| STM16    | 1 ~ 26  |
- 設定分解能 1

8. SENSE[1]:PATTERN:STM:DATA (パターンオプション)

- 機能 文字列データによる STM パターンの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:PATTERN:STM:DATA <start\_frm>,<start\_row>,<start\_col>,<length>,<str>
- クエリ SENSE[1]:PATTERN:STM:DATA? <start\_frm>,<start\_row>,<start\_col>,<length>
- パラメータ
  - <start\_frm> = パターン設定を開始するフレーム番号  
 設定範囲 1 ~ フレーム数  
 設定分解能 1
  - <start\_row> = パターン設定を開始する行  
 設定範囲 1 ~ 9  
 設定分解能 1
  - <start\_col> = パターン設定を開始する列

STM パターン	設定範囲
STM4	1 byte ~ 1080 byte
STM16	1 byte ~ 4320 byte

- 設定分解能 1 byte
  - <length> = 設定するビット数
  - <str> = 設定するパターン・データ
    - "B-----" パターンデータを 2 進で指定  
 -----: 0,1 (最大 512 文字)
    - "H-----" パターン・データを 16 進で指定  
 -----: 0~9, a~f, A~F (最大 512 文字)
- クエリ応答形式 <str> = 応答するパターン  
 "H-----" パターン・データを 16 進で応答  
 -----: 0~9, a~f, A~F (最大 512 文字)
- 説明 <start\_frm>,<start\_row>,<start\_col> から <length> で示されるビット数の STM パターンを <str> の内容で上書きします。STM パターンの範囲を超えるデータは無視されます。<length> より <str> で示されるデータのビット数が小さい場合は <str> で示されるデータのビット数だけ設定します。<length> が <str> で示されるデータのビット数より小さい場合は <length> だけ設定します。  
 ペイロードに PRBS パターンを挿入しているときは、ペイロード部のビットはパターン・メモリに保存されるだけとなります。クエリの場合、STM パターンの範囲を超えるデータと、512 文字で示されるビット数 (512 × 4=2048 ビット) を超えるデータは読み出せません。常にパターン・メモリの内容を読み出します。

5.5.1 SENSE[1] サブシステム

- 使用例 フレーム番号 1、1 行、1 列から 16 ビット・データ 4142 (16 進数) を設定する場合  
SENS1:PATT:STM:DATA 1,1,1,16,"H4142"

9. SENSE[1]:PATTern:STM:BDATa (パターンオプション)

- 機能 ブロック・データによる STM パターンの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:PATTern:STM:BDATa <start\_frm>,<start\_row>,<start\_col>,<length>,<block>
- クエリ SENSE[1]:PATTern:STM:BDATa? <start\_frm>,<start\_row>,<start\_col>,<length>
- パラメータ  
 <start\_frm> = パターン設定を開始するフレーム番号  
 設定範囲 1 ~ フレーム数  
 設定分解能 1  
 <start\_row> = パターン設定を開始する行  
 設定範囲 1 ~ 9  
 設定分解能 1  
 <start\_col> = パターン設定を開始する列

STM パターン	設定範囲
STM4	1 byte ~ 1080 byte
STM16	1 byte ~ 4320 byte

設定分解能 1 byte

<length> = 設定するビット数

<block> = 設定するパターン・データ

- クエリ応答形式 <block> = 応答するパターン
- 説明  
 <start\_frm>,<start\_row>,<start\_col> から <length> で示されるビット数の STM パターンを <block> の内容で上書きします。  
 STM パターンの範囲を超えるデータは無視されます。  
 <length> より <block> で示されるデータのビット数が小さい場合は <block> で示されるデータのビット数だけ設定します。  
 <length> が <block> で示されるデータのビット数より小さい場合は <length> だけ設定します。  
 ペイロードに PRBS パターンを挿入しているときは、ペイロード部のビットはパターン・メモリに保存されるだけとなります。  
 クエリの場合、STM パターンの範囲を超えるデータは読み出せません。常にパターン・メモリの内容を読み出します。
- 使用例 フレーム番号 1、1 行、1 列から 16 ビット・データ 4142 (16 進数) を設定する場合  
SENS1:PATT:STM:BDAT 1,1,1,16,#12AB



10. SENSE[1]:PATTERN:STM:PAYLOAD:PRBS (パターンオプション)
- 機能 STM パターンのペイロードへの PRBS パターン挿入機能の ON/OFF
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:PATTERN:STM:PAYLOAD:PRBS <bool>
  - パラメータ <bool>
  - クエリ応答形式 0 | 1
  - 説明 STM パターンのペイロードに PRBS パターンを挿入するか (ON)、しないか (OFF) を設定します。
11. SENSE[1]:PATTERN:STM:PAYLOAD:PRBS:SELECT (パターンオプション)
- 機能 STM パターンのペイロードへ挿入する PRBS パターンのパターン長の設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:PATTERN:STM:PAYLOAD:PRBS:SELECT <type>
  - パラメータ <type> = PRBS<n>  
<n> = 7,9,10,11,15,23,31
  - クエリ応答形式 PRBS<n>  
<n> = 7,9,10,11,15,23,31
  - 説明 STM パターンのペイロードへ挿入する PRBS パターンのパターン長を設定します。

5.5.1 SENSE[1] サブシステム

12. SENSE[1]:PATtern:STM:PAYLoad:PRBS:MRATio (パターンオプション)

- 機能 STM パターンのペイロードへ挿入する PRBS パターンのマーク率の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:PATtern:STM:PAYLoad:PRBS:MRATio <ratio>
- パラメータ <ratio> = {MRATio<n> | MINVerted4}
- クエリ応答形式 MRAT<n> | MINV4
- 説明 STM パターンのペイロードへ挿入する PRBS パターンのマーク率を設定します。

コマンド・パラメータ	マーク率
MRAT<n>	<n> = 0 マーク率 0/8
	<n> = 1 マーク率 1/8
	<n> = 2 マーク率 1/4
	<n> = 4 マーク率 1/2
	<n> = 6 マーク率 3/4
	<n> = 7 マーク率 7/8
	<n> = 8 マーク率 8/8
MINV4	マーク率 1/2B

13. SENSE[1]:PATtern:STM:SCRamble (パターンオプション)

- 機能 STM パターンのスクランブル機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:PATtern:STM:SCRamble <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 STM パターンに対してスクランブルを実行するか (ON)、しないか (OFF) を設定します。

14. SENSE[1]:PATTERN:STM:B1 (パターンオプション)
- 機能 STM パターンの B1 バイト挿入機能の ON/OFF
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:PATTERN:STM:B1 <bool>
  - パラメータ <bool>
  - クエリ応答形式 0 | 1
  - 説明 STM パターンに対して B1 バイトを挿入するか (ON)、しないか (OFF) を設定します。
15. SENSE[1]:PATTERN:FLEXible:STABLE (パターンオプション)
- 機能 FLEX パターン・シーケンス・テーブルのパターン編集
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:PATTERN:FLEXible:STABLE <index>,<patt>,<length>,<trig\_signal>
  - クエリ SENSE[1]:PATTERN:FLEXible:STABLE? <index>
  - パラメータ
    - <index> = FLEX パターン・シーケンス・テーブル内のパターンを指定するインデックス
      - 設定範囲 1 ~ パターン数
      - 設定分解能 1
    - <patt> = 選択パターン {<prog\_no> | PRBS}
    - <length> = <patt> が PRBS のときのパターン長
      - 設定範囲 128 bits ~ 2,097,152 bits
      - 設定分解能 64 bits
    - <trig\_signal> = FLEX Trigger の出力レベル {HIGH | LOW}
  - クエリ応答形式 <patt>,<length>,<trig\_signal>
  - 説明 FLEX パターン・シーケンス・テーブルのパターンを編集します。  
<length> の設定は選択パターンが PRBS パターンのときに有効になります。PROG パターンの場合は任意の値を設定して下さい。  
FLEX パターンで使用される PROG パターン長は  
SENS1:PATT:FLEX:PROG[LENG]  
で設定されます。

5.5.1 SENSE[1] サブシステム

クエリの場合、選択パターンが PROG パターンのときは <length> で PROG パターン長が読み出せます。

<patt>	選択パターン
<prog_no>	PROG パターン PROG パターンの番号を指定します。 設定範囲 1 ~ 127 設定分解能 1
PRBS	PRBS パターン

<trig_signal>	出力レベル
HIGH	HIGH レベル FLEX Trigger において、HIGH レベルの信号を出力します。
LOW	LOW レベル FLEX Trigger において、LOW レベルの信号を出力します。

16. SENSE[1]:PATTERN:FLEXible:STABLE:COUNT? (パターンオプション)

- 機能 FLEXパターン・シーケンス・テーブルのパターン数の問い合わせ
- コマンドとクエリの存在 Query
- クエリ SENSE[1]:PATTERN:FLEXible:STABLE:COUNT?
- クエリ応答形式 NR1 (整数型)
- 説明 FLEXパターン・シーケンス・テーブルのパターン数を問い合わせます。

17. SENSE[1]:PATTERN:FLEXible:STABLE:NEW (パターンオプション)

- 機能 FLEX パターン・シーケンス・テーブルにパターン追加
- コマンドとクエリの存在 Command
- コマンド SENSE[1]:PATTERN:FLEXible:STABLE:NEW <index>,<patt>,<length>,<trig\_signal>
- パラメータ
  - <index> = FLEX パターン・シーケンス・テーブル内のパターンを指定するインデックス
    - 設定範囲 1 ~ (パターン数 + 1)
    - 設定分解能 1
  - <patt> = 選択パターン {<prog\_no> | PRBS}
  - <length> = <patt> が PRBS のときのパターン長
    - 設定範囲 128 bits ~ 2,097,152 bits
    - 設定分解能 64 bits
  - <trig\_signal> = FLEX Trigger の出力レベル {HIGH | LOW}
- 説明 FLEX パターン・シーケンス・テーブルにパターンを追加します。  
 <index> で指定された位置にパターンを追加します。  
 パターン・シーケンス・テーブルのパターン数は最大 1024 です。  
 パターン数が 1024 の場合、パターンを追加できません。  
 <length> の設定は選択パターンが PRBS パターンのときに有効になります。  
 PROG パターンの場合は任意の値を設定して下さい。  
 FLEX パターンで使用される PROG パターン長は  
 SENSE1:PATT:FLEX:PROG[:LENG]  
 で設定されます。

<patt>	選択パターン
<prog_no>	PROG パターン PROG パターンの番号を指定します。 設定範囲 1 ~ 127 設定分解能 1
PRBS	PRBS パターン

<trig_signal>	出力レベル
HIGH	HIGH レベル FLEX Trigger において、HIGH レベルの信号を出力します。
LOW	LOW レベル FLEX Trigger において、LOW レベルの信号を出力します。

## 5.5.1 SENSE[1] サブシステム

18. SENSE[1]:PATtern:FLEXible:STABLE:REMOve (パターンオプション)
- 機能 FLEX パターン・シーケンス・テーブルのパターン削除
  - コマンドとクエリの存在 Command
  - コマンド SENSE[1]:PATtern:FLEXible:STABLE:REMOve <index>
  - パラメータ <index> = FLEX パターン・シーケンス・テーブル内のパターンを指定するインデックス  
設定範囲 1 ~ パターン数  
設定分解能 1
  - 説明 FLEX パターン・シーケンス・テーブルのパターンを削除します。パターンをすべて削除することはできません。インデックス 2 のパターンが PRBS パターンの場合、インデックス 1 のパターンは削除できません。
19. SENSE[1]:PATtern:FLEXible:STABLE:DEFault (パターンオプション)
- 機能 FLEX パターン・シーケンス・テーブルのデフォルト設定
  - コマンドとクエリの存在 Command
  - コマンド SENSE[1]:PATtern:FLEXible:STABLE:DEFault
  - 説明 FLEX パターン・シーケンス・テーブルをデフォルトの設定に戻します。
20. SENSE[1]:PATtern:FLEXible:PROGram[:LENGth] (パターンオプション)
- 機能 FLEX パターンで使用される PROG パターン長の設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:PATtern:FLEXible:PROGram[:LENGth] <prog\_no>, <length>
  - クエリ SENSE[1]:PATtern:FLEXible:PROGram[:LENGth]? <prog\_no>
  - パラメータ <prog\_no> = FLEX パターンで使用される PROG パターンの番号  
設定範囲 1 ~ 127  
設定分解能 1  
<length> = PROG パターン長  
設定範囲 128 bits ~ 65,536 bits  
設定分解能 64 bits
  - クエリ応答形式 <length> = NR1 (整数型)
  - 説明 FLEX パターンで使用される PROG パターン長を設定します。

21.	SENSE[1]:PATTERN:FLEXible:PROGrama:DATA	(パターンオプション)
機能	文字列データによる FLEX パターンで使用される PROG パターンの設定	
コマンドとクエリの存在	Command/Query	
コマンド	SENSE[1]:PATTERN:FLEXible:PROGrama:DATA <prog_no>, <start_bit>,<length>,<str>	
クエリ	SENSE[1]:PATTERN:FLEXible:PROGrama:DATA? <prog_no>, <start_bit>,<length>	
パラメータ	<prog_no> = FLEX パターンで使用される PROG パターンの番号 設定範囲 1 ~ 127 設定分解能 1 <start_bit> = パターン設定を開始するアドレス 設定範囲 0 bit ~ (パターン長 - 1) bit 設定分解能 1 bit <length> = 設定するビット数 <str> = 設定するパターン・データ "B-----" パターン・データを 2 進で指定 -----: 0,1 (最大 512 文字) "H-----" パターン・データを 16 進で指定 -----: 0 ~ 9, a ~ f, A ~ F (最大 512 文字)	
クエリ応答形式	<str> = 応答するパターン "H-----" パターン・データを 16 進で応答 -----: 0 ~ 9, a ~ f, A ~ F (最大 512 文字)	
説明	<prog_no> で指定された PROG パターンを、<start_bit> から <length> で示されるビット数だけ <str> の内容で上書きします。 <prog_no> で指定された PROG パターンの範囲を超えるデータは無視されます。 <length> より <str> で示されるデータのビット数が小さい場合は <str> で示されるデータのビット数だけ設定します。 <length> が <str> で示されるデータのビット数より小さい場合は <length> だけ設定します。 クエリの場合、<prog_no> で指定された PROG パターンの範囲を超えるデータと、512 文字で示されるビット数 (512×4=2048 ビット) を超えるデータは読み出せません。	
使用例	PROG パターン番号 1 のアドレス 0 bit から 16 ビット・データ 4142 (16 進数) を設定する場合 SENSE1:PATT:FLEX:PROG:DATA 1,0,16,"H4142"	

## 5.5.1 SENSE[1] サブシステム

22.	SENSe[1]:PATtern:FLExiBle:PROGram:BDATa	(パターンオプション)
機能	ブロック・データによる FLEX パターンで使用される PROG パターンの設定	
コマンドとクエリの存在	Command/Query	
コマンド	SENSe[1]:PATtern:FLExiBle:PROGram:BDATa <prog_no>, <start_bit>,<length>,<block>	
クエリ	SENSe[1]:PATtern:FLExiBle:PROGram:BDATa? <prog_no>, <start_bit>,<length>	
パラメータ	<prog_no> = FLEX パターンで使用される PROG パターンの番号 設定範囲      1 ~ 127 設定分解能    1 <start_bit> = パターン設定を開始するアドレス 設定範囲      0 bit~ (パターン長 - 1) bit 設定分解能    1 bit <length> = 設定するビット数 <block> = 設定するパターン・データ	
クエリ応答形式	<block> = 応答するパターン	
説明	<prog_no> で指定された PROG パターンを、<start_bit> から <length>で示されるビット数だけ<block>の内容で上書きします。 <prog_no>で指定されたPROGパターンの範囲を超えるデータは無視されます。 <length>より<block>で示されるデータのビット数が小さい場合は<block>で示されるデータのビット数だけ設定します。 <length>が<block>で示されるデータのビット数より小さい場合は<length>だけ設定します。 クエリの場合、<prog_no>で指定されたPROGパターンの範囲を超えるデータは読み出せません。	
使用例	PROG パターン番号 1 のアドレス 0 bit から 16 ビット・データ 4142 (16 進数) を設定する場合 SENS1:PATT:FLEX:PROG:BDAT 1,0,16,#12AB	



23. SENSE[1]:PATTERN:FLEXible:PRBS:SElect (パターンオプション)
- 機能 FLEX パターンで使用される PRBS パターンのパターン長の設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:PATTERN:FLEXible:PRBS:SElect <type>
  - パラメータ <type> = PRBS<n>  
<n> = 7,9,10,11,15,23,31
  - クエリ応答形式 PRBS<n>  
<n> = 7,9,10,11,15,23,31
  - 説明 FLEX パターンで使用される PRBS パターンのパターン長を設定します。

24. SENSE[1]:PATTERN:FLEXible:PRBS:MRATio (パターンオプション)
- 機能 FLEX パターンで使用される PRBS パターンのマーク率の設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:PATTERN:FLEXible:PRBS:MRATio <ratio>
  - パラメータ <ratio> = {MRATio<n> | MINVerted4}
  - クエリ応答形式 MRAT<n> | MINV4
  - 説明 FLEX パターンで使用される PRBS パターンのマーク率を設定します。

コマンド・パラメータ	マーク率
MRAT<n>	<n> = 0    マーク率 0/8
	<n> = 1    マーク率 1/8
	<n> = 2    マーク率 1/4
	<n> = 4    マーク率 1/2
	<n> = 6    マーク率 3/4
	<n> = 7    マーク率 7/8
	<n> = 8    マーク率 8/8
MINV4	マーク率 1/2B

5.5.1 SENSE[1] サブシステム

25. SENSE[1]:PATTERN:POLARITY

- 機能                      パターン極性の設定
- コマンドとクエリの存在    Command/Query
- コマンド                    SENSE[1]:PATTERN:POLARITY <type>
- パラメータ                 <type> = {NORMAL | INVERTED}
- クエリ応答形式            NORM | INV
- 説明                        パターン極性を設定します。

コマンド・パラメータ	パターン極性
NORM	正論理
INV	負論理

26. SENSE[1]:PATTERN:TRACK

- 機能                      PPG モジュールのパターンと共通設定の ON/OFF
- コマンドとクエリの存在    Command/Query
- コマンド                    SENSE[1]:PATTERN:TRACK <bool>
- パラメータ                 <bool>
- クエリ応答形式            0 | 1
- 説明                        PPG モジュールのパターンと同じパターンを使用するか (ON)、使用しないか (OFF) を設定します。

27. SENSE[1]:SYNCHRONISAT[:AUTO]

- 機能                      自動パターン同期の設定
- コマンドとクエリの存在    Command/Query
- コマンド                    SENSE[1]:SYNCHRONISAT[:AUTO] ONCE | <bool>
- パラメータ                 ONCE  
<bool>
- クエリ応答形式            0 | 1
- 説明                        1 回だけパターン同期を行うか (ONCE)、パターン同期を自動で行うか (ON)、行わないか (OFF) を設定します。

## 28. SENSE[1]:SYNChronisat:PATtern:PROGram:ADDRess

- 機能                    PROG パターン時の同期パターン・アドレスの設定
- コマンドとクエリの存在    Command/Query
- コマンド                SENSE[1]:SYNChronisat:PATtern:PROGram:ADDRess <int>
- パラメータ              <int>
- クエリ応答形式        NR1 (整数型)
- 説明                    PROG パターン時の同期パターン・アドレスを設定します。  
                           設定範囲    0 bit~ (パターン長 -1) bit  
                           設定分解能 1 bit

## 29. SENSE[1]:SYNChronisat:PATtern:FLEXible:INDEX (パターンオプション)

- 機能                    FLEX パターン時の同期パターン・インデックスの設定
- コマンドとクエリの存在    Command/Query
- コマンド                SENSE[1]:SYNChronisat:PATtern:FLEXible:INDEX <int>
- パラメータ              <int>
- クエリ応答形式        NR1 (整数型)
- 説明                    FLEX パターン時の同期パターン・インデックスを設定します。  
                           選択されているパターンがPRBSであるインデックスは設定できません。  
                           設定範囲    1 ~ パターン数  
                           設定分解能 1

5.5.1 SENSE[1] サブシステム

30. SENSE[1]:SYNChronisat:PATtern:FLEXible:ADDRESS(パターンオプション)

- 機能 FLEX パターン時の同期パターン・アドレスの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SYNChronisat:PATtern:FLEXible:ADDRESS <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 FLEX パターン時の同期パターン・アドレスを設定します。

同期パターン・インデックスの 次のインデックスのパターン	設定範囲
PROG	0 bit~ (同期パターン・インデックスの パターン長-1) bit
PRBS	0 bit~ (同期パターン・インデックスの パターン長-32) bit

設定分解能 1 bit

31. SENSE[1]:SYNChronisat:PATtern:LENGth

- 機能 同期パターン長の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SYNChronisat:PATtern:LENGth <int>
- パラメータ <int> = 8, 16, 24, 32
- クエリ応答形式 NR1 (整数型)
- 説明 PROG パターンの同期引き込みに使用する、同期パターンの長さ (ビット数) を設定します。

32. SENSE[1]:SYNChronisat:THReshold:AUTO

- 機能 同期判定に使用するエラー・レートのスレッシュホールド自動設定の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SYNChronisat:THReshold:AUTO <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 同期引き込みおよび同期外れ判定のエラー・レートのスレッシュホールドを、自動的にセットするか (ON)、しないか (OFF) を設定します。

## 33. SENSE[1]:SYNChronisat:THReshold:PRBS:GAIN

- 機能 PRBS パターン時 (STM, FLEX パターンの PRBS パターンも含む) の同期引き込み判定のエラー・レートのスレッシュホールドの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SYNChronisat:THReshold:PRBS:GAIN <real>
- パラメータ <real> = 1E-2, 1E-3, 1E-4, 1E-5, 1E-6, 1E-7
- クエリ応答形式 NR3 (浮動小数点型)
- 説明 PRBS パターン時 (STM, FLEX パターンの PRBS パターンも含む) の同期引き込み判定のエラー・レートのスレッシュホールドを設定します。  
同期引き込み判定のスレッシュホールドと同期外れ判定のスレッシュホールドの関係が、[Sync Gain Threshold] ≤ [Sync Loss Threshold] の条件を満たすように設定して下さい。

## 34. SENSE[1]:SYNChronisat:THReshold:PRBS:LOSS

- 機能 PRBS パターン時 (STM, FLEX パターンの PRBS パターンも含む) の同期外れ判定のエラー・レートのスレッシュホールドの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SYNChronisat:THReshold:PRBS:LOSS <real>
- パラメータ <real> = 1E-2, 1E-3, 1E-4, 1E-5, 1E-6, 1E-7
- クエリ応答形式 NR3 (浮動小数点型)
- 説明 PRBS パターン時 (STM, FLEX パターンの PRBS パターンも含む) の同期外れ判定のエラー・レートのスレッシュホールドを設定します。  
同期引き込み判定のスレッシュホールドと同期外れ判定のスレッシュホールドの関係が、[Sync Gain Threshold] ≤ [Sync Loss Threshold] の条件を満たすように設定して下さい。

## 5.5.1 SENSE[1] サブシステム

## 35. SENSE[1]:SYNChronisat:THReshold:MEMory:GAIN

- 機能 ZSUB, PROG, STM, FLEX パターン時の同期引き込み判定のエラー・レートのスレッシュホールドの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SYNChronisat:THReshold:MEMory:GAIN <real>
- パラメータ <real> = 1E-2, 1E-3, 1E-4, 1E-5, 1E-6, 1E-7, 1E-8, 1E-9, 1E-10
- クエリ応答形式 NR3 (浮動小数点型)
- 説明 ZSUB, PROG, STM, FLEX パターン時の同期引き込み判定のエラー・レートのスレッシュホールドを設定します。  
STM, FLEX パターンで使用される PRBS パターンについては SENSE[1]:SYNC:THR:PRBS:GAIN コマンドで別にスレッシュホールドを設定します。  
同期引き込み判定のスレッシュホールドと同期外れ判定のスレッシュホールドの関係が、 $[\text{Sync Gain Threshold}] \leq [\text{Sync Loss Threshold}]$  の条件を満たすように設定して下さい。

## 36. SENSE[1]:SYNChronisat:THReshold:MEMory:LOSS

- 機能 ZSUB, PROG, STM, FLEX パターン時の同期外れ判定のエラー・レートのスレッシュホールドの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SYNChronisat:THReshold:MEMory:LOSS <real>
- パラメータ <real> = 1E-2, 1E-3, 1E-4, 1E-5, 1E-6, 1E-7, 1E-8, 1E-9, 1E-10
- クエリ応答形式 NR3 (浮動小数点型)
- 説明 ZSUB, PROG, STM, FLEX パターン時の同期外れ判定のエラー・レートのスレッシュホールドを設定します。  
STM, FLEX パターンで使用される PRBS パターンについては SENSE[1]:SYNC:THR:PRBS:LOSS コマンドで別にスレッシュホールドを設定します。  
同期引き込み判定のスレッシュホールドと同期外れ判定のスレッシュホールドの関係が、 $[\text{Sync Gain Threshold}] \leq [\text{Sync Loss Threshold}]$  の条件を満たすように設定して下さい。

## 37. SENSE[1]:MASK:ROUTE&lt;n&gt;

- 機能 測定ルート・マスクの ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:MASK:ROUTE<n> <bool>  
<n> = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 <n> で指定された測定ルートをマスクするか (ON)、しないか (OFF) を設定します。

## 38. SENSE[1]:ASEarch[:STATe]

- 機能 オート・サーチの開始/停止
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:ASEarch[:STATe] <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 オート・サーチの開始 (ON)、停止 (OFF) を設定します。

## 39. SENSE[1]:ASEarch:FAIL?

- 機能 オート・サーチ実行結果の問い合わせ
- コマンドとクエリの存在 Query
- クエリ SENSE[1]:ASEarch:FAIL?
- クエリ応答形式 0 | 1
- 説明 オート・サーチの実行結果を問い合わせます。  
オート・サーチが失敗した場合と一度も実行されていない場合は 1 を返します。オート・サーチが成功した場合は 0 を返します。  
オート・サーチの終了を確認してから実行して下さい。

## 5.5.1 SENSE[1] サブシステム

## 40. SENSE[1]:ASEarch:CLOCK

- 機能 クロック入力の遅延量と入力極性のオート・サーチ機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:ASEarch:CLOCK <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 クロック入力の遅延量と入力極性をオート・サーチするか (ON)、しないか (OFF) を設定します。

## 41. SENSE[1]:ASEarch:TVOLTage

- 機能 データ入力のスレッショルド電圧のオート・サーチ機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:ASEarch:TVOLTage <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 データ入力のスレッショルド電圧をオート・サーチするか (ON)、しないか (OFF) を設定します。

## 42. SENSE[1]:ASEarch:PATtern

- 機能 パターン関連のオート・サーチ機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:ASEarch:PATtern <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 PRBS パターンの場合はパターン長、マーク率、パターン極性を、PRBS パターン以外の場合は、パターン極性をオート・サーチするか (ON)、しないか (OFF) を設定します。



## 43. SENSE[1]:MONitor:HISTory:CLEar

- 機能 モニタ・ツール・バーの履歴情報のクリア
- コマンドとクエリの存在 Command
- コマンド SENSE[1]:MONitor:HISTory:CLEar
- 説明 モニタ・ツール・バーの履歴情報をクリアします。

## 44. SENSE[1]:LOG:BMEasurement

- 機能 基本測定結果のログ記録機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:LOG:BMEasurement <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 基本測定の結果をログに記録するか (ON)、しないか (OFF) を設定します。ログには最後に測定した結果のみを記録します。

## 45. SENSE[1]:LOG:HISTory:SCLoss

- 機能 シンクロスとクロックロスの履歴記録機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:LOG:HISTory:SCLoss <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 シンクロスとクロックロスの発生と復旧した時間をログに履歴記録するか (ON)、しないか (OFF) を設定します。

## 46. SENSE[1]:LOG:HISTory:MDATa:IERate

- 機能 区間エラー・レート値の履歴記録機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:LOG:HISTory:MDATa:IERate <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 区間エラー・レート値の測定データをログに履歴記録するか (ON)、しないか (OFF) を設定します。

## 5.5.1 SENSE[1] サブシステム

## 47. SENSE[1]:LOG:HISTORY:MDATA:IECount

- 機能 区間エラー・カウント値の履歴記録機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:LOG:HISTORY:MDATA:IECount <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 区間エラー・カウント値の測定データをログに履歴記録するか (ON)、しないか (OFF) を設定します。

## 48. SENSE[1]:LOG:HISTORY:MDATA:EINTervals

- 機能 エラー・インターバル値の履歴記録機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:LOG:HISTORY:MDATA:EINTervals <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 エラー・インターバル値の測定データをログに履歴記録するか (ON)、しないか (OFF) を設定します。

## 49. SENSE[1]:LOG:HISTORY:MDATA:EFINTervals

- 機能 エラー・フリー・インターバル値の履歴記録機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:LOG:HISTORY:MDATA:EFINTervals <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 エラー・フリー・インターバル値の測定データをログに履歴記録するか (ON)、しないか (OFF) を設定します。

## 50. SENSE[1]:LOG:HISTORY:MDATA:FREQUENCY

- 機能 周波数値のヒストリ記録機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:LOG:HISTORY:MDATA:FREQUENCY <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 周波数値の測定データをログにヒストリ記録するか (ON)、しないか (OFF) を設定します。

## 51. SENSE[1]:BMEASUREMENT[:STATE]

- 機能 基本測定の開始/停止
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:BMEASUREMENT[:STATE] <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 基本測定の開始 (ON)、停止 (OFF) を設定します。  
測定モードがシングルの場合は、測定が終了すると自動的に OFF に設定されます。

## 52. SENSE[1]:BMEASUREMENT:MTIME:MODE

- 機能 測定タイム・モードの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:BMEASUREMENT:MTIME:MODE <type>
- パラメータ <type> = {SINGLE | REPEAT | UNTIMED}
- クエリ応答形式 SING | REP | UNT
- 説明 測定モードを設定します。

コマンド・パラメータ	測定モード
SING	シングル・モード。 測定期間を 1 回だけ測定します。
REP	リピート・モード。 測定期間を繰り返し測定します。
UNT	アンタイムド・モード。 測定期間に関係なく測定します。

## 5.5.1 SENSE[1] サブシステム

## 53. SENSE[1]:BMEasurement:MTIME:INTERval

- 機能 測定インターバルの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:BMEasurement:MTIME:INTERval <real>
- パラメータ <real> = 0.1, 1.0
- クエリ応答形式 NR2 (固定小数点型)
- 説明 測定インターバルを設定します。  
測定インターバルには、0.1 秒か 1 秒のどちらかを設定します。

## 54. SENSE[1]:BMEasurement:MTIME:PERiod

- 機能 測定期間の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:BMEasurement:MTIME:PERiod <day>, <hour>, <minute>, <second>
- パラメータ <day> = 0 ~ 99 の数値データ (日)  
<hour> = 0 ~ 23 の数値データ (時)  
<minute> = 0 ~ 59 の数値データ (分)  
<second> = 0 ~ 59 の数値データ (秒)
- クエリ応答形式 <day>, <hour>, <minute>, <second>  
<day> = <hour> = <minute> = <second> = NR1 (整数型)
- 説明 測定期間を設定します。  
設定範囲 1 秒 ~ 99 日 23 時 59 分 59 秒

## 55. SENSE[1]:BMEasurement:DMODE

- 機能 エラー検出モードの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:BMEasurement:DMODE <type>
- パラメータ <type> = {OIT | OPT | SOT}
- クエリ応答形式 OIT | OPT | SOT
- 説明 エラー検出モードを設定します。

コマンド・パラメータ	内容
OIT	欠落エラー (Omitting)、挿入エラー (Inserting)、Total エラーを検出します。
OPT	オーバーヘッド領域 (Overhead)、ペイロード領域 (Payload)、すべての領域 (Total) のエラーを検出します。
SOT	特定領域 (Specific)、特定領域以外 (Other)、すべての領域 (Total) のエラーを検出します。

## 56. SENSE[1]:BMEasurement:BURSt

- 機能 バースト・モードの ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:BMEasurement:BURSt <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 バースト・モードを有効にするか (ON)、無効にするか (OFF) を設定します。

## 57. SENSE[1]:BMEasurement:EVALuate:CLOsS

- 機能 クロックロス・インターバルを測定に含めるか否かの選択
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:BMEasurement:EVALuate:CLOsS <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 クロックロス・インターバルを測定に含めるか (ON)、含めないか (OFF) を設定します。

5.5.1 SENSE[1] サブシステム

58. SENSE[1]:BMEasurement:EVALuate:SLOSs

- 機能 シンクロス・インターバルを測定に含めるか否かの選択
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:BMEasurement:EVALuate:SLOSs <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 シンクロス・インターバルを測定に含めるか (ON)、含めないか (OFF) を設定します。

59. SENSE[1]:BMEasurement:EPERformance:THReshold

- 機能 US/SES, DM の計算に含める期間のエラー・レート・スレッシュホールドの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:BMEasurement:EPERformance:THReshold <us\_ses>, <dm>
- パラメータ <us\_ses> = 1E-3, 1E-4  
<dm> = 1E-6, 1E-8
- クエリ応答形式 <us\_ses>, <dm>  
<us\_ses> = <dm> = NR3 (浮動小数点型)
- 説明 エラー・パフォーマンス測定における不稼動秒 (US:Unavailable Seconds)、異常誤り秒 (SES:Severely Errored Seconds)、劣化分 (DM:Degraded Minutes) の計算に含める期間のエラー・レートのスレッシュホールドを設定します。

コマンド・パラメータ	内容
<us_ses>	US/SES の計算に含める期間のエラー・レートのスレッシュホールド
<dm>	DM の計算に含める期間のエラー・レートのスレッシュホールド

<us\_ses> と <dm> は以下に示すペアで使用します。

<us_ses>	<dm>
1E-3	1E-6
1E-4	1E-8

60. SENSE[1]:BMEasurement:CDAa

- 機能 基本測定の途中結果表示 ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:BMEasurement:CDAa <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 基本測定の途中結果を表示するか (ON)、しないか (OFF) を設定します。

## 5.5.1 SENSE[1] サブシステム

## 61. SENSE[1]:SPECific:PROGram:STARt:ADDRess

- 機能                    PROG パターン時の特定領域開始アドレスの設定
- コマンドとクエリの存在    Command/Query
- コマンド                SENSE[1]:SPECific:PROGram:STARt:ADDRess <int>
- パラメータ              <int>
- クエリ応答形式        NR1 (整数型)
- 説明                    PROG パターン時の特定領域開始アドレスを設定します。  
設定範囲    0 bit~ (パターン長 - 1) bit  
設定分解能 1 bit

## 62. SENSE[1]:SPECific:PROGram:END:ADDRess

- 機能                    PROG パターン時の特定領域終了アドレスの設定
- コマンドとクエリの存在    Command/Query
- コマンド                SENSE[1]:SPECific:PROGram:END:ADDRess <int>
- パラメータ              <int>
- クエリ応答形式        NR1 (整数型)
- 説明                    PROG パターン時の特定領域終了アドレスを設定します。  
設定範囲    0 bit~ (パターン長 - 1) bit  
設定分解能 1 bit

## 63. SENSE[1]:SPECific:ZSUBstitut:STARt:ADDRess

- 機能                    ZSUB パターン時の特定領域開始アドレスの設定
- コマンドとクエリの存在    Command/Query
- コマンド                SENSE[1]:SPECific:ZSUBstitut:STARt:ADDRess <int>
- パラメータ              <int>
- クエリ応答形式        NR1 (整数型)
- 説明                    ZSUB パターン時の特定領域開始アドレスを設定します。  
設定範囲    0 bit~ (パターン長 - 1) bit  
設定分解能 1 bit



## 64. SENSE[1]:SPECific:ZSUBstitut:END:ADDRESS

- 機能 ZSUB パターン時の特定領域終了アドレスの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SPECific:ZSUBstitut:END:ADDRESS <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 ZSUB パターン時の特定領域終了アドレスを設定します。  
設定範囲 0 bit~ (パターン長 - 1) bit  
設定分解能 1 bit

## 65. SENSE[1]:SPECific:STM:START:FNUMBER (パターンオプション)

- 機能 STM パターン時の特定領域開始フレーム番号の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SPECific:STM:START:FNUMBER <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 STM パターン時の特定領域開始フレーム番号を設定します。  
設定範囲 1 ~ フレーム数  
設定分解能 1

## 66. SENSE[1]:SPECific:STM:START:ROW (パターンオプション)

- 機能 STM パターン時の特定領域開始行の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SPECific:STM:START:ROW <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 STM パターン時の特定領域開始行を設定します。  
設定範囲 1 ~ 9  
設定分解能 1

5.5.1 SENSE[1] サブシステム

67. SENSE[1]:SPECific:STM:STARt:COLumn (パターンオプション)

- 機能 STM パターン時の特定領域開始列の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SPECific:STM:STARt:COLumn <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 STM パターン時の特定領域開始列を設定します。

STM パターン	設定範囲
STM4	1 byte ~ 1080 byte
STM16	1 byte ~ 4320 byte

設定分解能 1 byte

68. SENSE[1]:SPECific:STM:END:FNUMBER (パターンオプション)

- 機能 STM パターン時の特定領域終了フレーム番号の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SPECific:STM:END:FNUMBER <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 STM パターン時の特定領域終了フレーム番号を設定します。  
設定範囲 1 ~ フレーム数  
設定分解能 1

69. SENSE[1]:SPECific:STM:END:ROW (パターンオプション)

- 機能 STM パターン時の特定領域終了行の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SPECific:STM:END:ROW <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 STM パターン時の特定領域終了行を設定します。  
設定範囲 1 ~ 9  
設定分解能 1

70. SENSE[1]:SPECIFIC:STM:END:COLumn (パターンオプション)

- 機能 STM パターン時の特定領域終了列の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SPECIFIC:STM:END:COLumn <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 STM パターン時の特定領域終了列を設定します。

STM パターン	設定範囲
STM4	1 byte ~ 1080 byte
STM16	1 byte ~ 4320 byte

設定分解能 1 byte

71. SENSE[1]:SPECIFIC:STM:MODE (パターンオプション)

- 機能 STM パターン時の特定領域測定モードの設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:SPECIFIC:STM:MODE <type>
- パラメータ <type> = {PATTERN | FRAME}
- クエリ応答形式 PATT | FRAM
- 説明 STM パターン時の特定領域測定モードを設定します。

コマンド・パラメータ	特定領域測定モード
PATT	パターン・モード。 パターン毎に特定領域測定を行います。
FRAM	フレーム・モード。 フレーム毎に特定領域測定を行います。

## 5.5.1 SENSE[1] サブシステム

72. SENSE[1]:SPECific:FLEXible:STARt:INdex (パターンオプション)
- 機能 FLEX パターン時の特定領域開始インデックスの設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:SPECific:FLEXible:STARt:INdex <int>
  - パラメータ <int>
  - クエリ応答形式 NR1 (整数型)
  - 説明 FLEX パターン時の特定領域開始インデックスを設定します。  
設定範囲 1 ~ パターン数  
設定分解能 1
73. SENSE[1]:SPECific:FLEXible:STARt:ADDRESS (パターンオプション)
- 機能 FLEX パターン時の特定領域開始アドレスの設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:SPECific:FLEXible:STARt:ADDRESS <int>
  - パラメータ <int>
  - クエリ応答形式 NR1 (整数型)
  - 説明 FLEX パターン時の特定領域開始アドレスを設定します。  
設定範囲 0 bit~(特定領域開始インデックスのパターン長-1) bit  
設定分解能 1 bit
74. SENSE[1]:SPECific:FLEXible:END:INdex (パターンオプション)
- 機能 FLEX パターン時の特定領域終了インデックスの設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:SPECific:FLEXible:END:INdex <int>
  - パラメータ <int>
  - クエリ応答形式 NR1 (整数型)
  - 説明 FLEX パターン時の特定領域終了インデックスを設定します。  
設定範囲 1 ~ パターン数  
設定分解能 1

75. SENSE[1]:SPECific:FLEXible:END:ADDRESS (パターンオプション)
- 機能 FLEX パターン時の特定領域終了アドレスの設定
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:SPECific:FLEXible:END:ADDRESS <int>
  - パラメータ <int>
  - クエリ応答形式 NR1 (整数型)
  - 説明 FLEX パターン時の特定領域終了アドレスを設定します。  
 設定範囲 0 bit~(特定領域終了インデックスのパターン長-1) bit  
 設定分解能 1 bit

76. SENSE[1]:EPANalysis:ERECord[:STATe] (エラー位相解析オプション)
- 機能 エラー記録機能の ON/OFF
  - コマンドとクエリの存在 Command/Query
  - コマンド SENSE[1]:EPANalysis:ERECord[:STATe] <bool>
  - パラメータ <bool>
  - クエリ応答形式 0 | 1
  - 説明 基本測定と同時にエラー記録を行うか (ON)、行わないか (OFF) を設定します。実際にエラー記録が実行されるには設定パターンが PRBS パターン以外で、パターン長が 256 ビット以上かつ 32 ビットの整数倍の長さである必要があります。

パターンタイプ	説明
PRBS	エラー記録不可
PROG	パターン長が、256 ビット以上で 32 ビットの整数倍の長さのときエラー記録可能
ZSUB	ZSUB7 以外はエラー記録可能
STM	エラー記録可能
FLEX	トータルのパターン長が 256 ビット以上の長さのときエラー記録可能

5.5.1 SENSE[1] サブシステム

77. SENSE[1]:EPANalysis:ERECord:AREa (エラー位相解析オプション)

- 機能 エラー記録領域の設定
- コマンドとクエリの存在 Command/Query
- コマンド SENSE[1]:EPANalysis:ERECord:AREa <type>
- パラメータ <type> = { SPECific | TOTal }
- クエリ応答形式 SPEC | TOT
- 説明 エラー記録の対象となる領域を設定します。

コマンド・パラメータ	エラー記録領域
SPEC	特定領域 この設定が有効になるにはエラー検出モードを Specific/Other/Total にする必要があります。 特定領域の設定は SENS[1]:SPEC 階層のコマンドを使用します。
TOT	パターン全体

## 5.5.2 INPut[1] サブシステム

### 1. INPut[1]:POLarity

- 機能 DATA 入力極性の設定
- コマンドとクエリの存在 Command/Query
- コマンド INPut[1]:POLarity <type>
- パラメータ <type> = {NORMal | INVerted}
- クエリ応答形式 NORM | INV
- 説明 DATA 入力の極性を設定します。

コマンド・パラメータ	DATA 入力極性
NORM	非反転
INV	反転

### 2. INPut[1]:TVOLtage

- 機能 DATA 入力スレッショルド電圧の設定
- コマンドとクエリの存在 Command/Query
- コマンド INPut[1]:TVOLtage <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 入力スレッショルド電圧を設定します。

終端タイプ	設定範囲
PECL (+3V)	+3.150V ~ +4.250V
LVPECL (+1.3V)	+1.450V ~ +2.550V
to GND (0V)	-2.040V ~ +2.040V
ECL (-2V)	-1.850V ~ -0.750V
CML	$(V_{cc}-1.100)V \sim (V_{cc}+0.100)V^{*1}$

\*1: Vcc は DATA 入力終端電圧です。

設定分解能 0.001V  
設定単位 V

5.5.2 INPut[1] サブシステム

3. INPut[1]:TERMination

- 機能 DATA 入力終端タイプの設定
- コマンドとクエリの存在 Command/Query
- コマンド INPut[1]:TERMination <type>
- パラメータ <type> = {P3V | P1R3V | GND | M2V | CML}
- クエリ応答形式 P3V | P1R3V | GND | M2V | CML
- 説明 DATA 入力の終端タイプを設定します。

コマンド・パラメータ	終端タイプ
P3V	PECL (+3V)
P1R3V	LVPECL (+1.3V)
GND	to GND (0V)
M2V	ECL (-2V)
CML	CML

4. INPut[1]:TERMination:VARiable

- 機能 DATA 入力の終端電圧値可変機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド INPut[1]:TERMination:VARiable <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 DATA 入力終端電圧値を可変するか (ON)、しないか (OFF) を設定します。ON に設定した場合は、設定された終端電圧値で終端され、OFF に設定した場合、初期値で終端されます。終端タイプが GND の場合は設定できません。



## 5. INPut[1]:TERMination:VOLTagE

- 機能 DATA 入力終端電圧の設定
- コマンドとクエリの存在 Command/Query
- コマンド INPut[1]:TERMination:VOLTagE <real>
- パラメータ <real>
- クエリ応答形式 NR2 (固定小数点型)
- 説明 DATA 入力終端電圧を設定します。  
この設定を有効にするためには、DATA 入力の終端電圧可変機能を ON に設定する必要があります。  
終端タイプが GND の場合は設定できません。

終端タイプ	設定範囲
PECL (+3V)	+2.70V ~ +3.30V
LVPECL (+1.3V)	+1.00V ~ +1.60V
ECL (-2V)	-2.30V ~ -1.70V
CML	0.00V ~ +3.50V

設定分解能 0.05V

設定単位 V

5.5.3 INPut2 サブシステム

5.5.3 INPut2 サブシステム

1. INPut2:POLarity

- 機能                      CLOCK 入力極性の設定
- コマンドとクエリの存在    Command/Query
- コマンド                    INPut2:POLarity <type>
- パラメータ                 <type> = {NORMal | INVerted}
- クエリ応答形式            NORM | INV
- 説明                        CLOCK 入力の極性を設定します。

コマンド・パラメータ	CLOCK 入力極性
NORM	非反転
INV	反転

2. INPut2:DELAy

- 機能                      CLOCK 遅延量の設定
- コマンドとクエリの存在    Command/Query
- コマンド                    INPut2:DELAy <int>
- パラメータ                 <int>
- クエリ応答形式            NR1 (整数型)
- 説明                        CLOCK 遅延量を設定します。  
                                  設定範囲    -1000ps ~ +1000ps  
                                  設定分解能   1ps  
                                  設定単位     ps

## 3. INPut2:TERMination

- 機能 CLOCK 入力終端タイプの設定
- コマンドとクエリの存在 Command/Query
- コマンド INPut2:TERMination <type>
- パラメータ <type> = {P3V | P1R3V | GND | M2V | CML}
- クエリ応答形式 P3V | P1R3V | GND | M2V | CML
- 説明 CLOCK 入力の終端タイプを設定します。

コマンド・パラメータ	終端タイプ
P3V	PECL (+3V)
P1R3V	LVPECL (+1.3V)
GND	to GND (0V)
M2V	ECL (-2V)
CML	CML

## 4. INPut2:TERMination:VARiable

- 機能 CLOCK 入力の終端電圧値可変機能の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド INPut2:TERMination:VARiable <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 CLOCK 入力終端電圧値を可変するか (ON)、しないか (OFF) を設定します。ON に設定した場合は、設定された終端電圧値で終端され、OFF に設定した場合、初期値で終端されます。終端タイプが GND の場合は設定できません。

5.5.3 INPut2 サブシステム

5. INPut2:TERMination:VOLTagE

- 機能                      CLOCK 入力終端電圧の設定
- コマンドとクエリの存在    Command/Query
- コマンド                    INPut2:TERMination:VOLTagE <real>
- パラメータ                 <real>
- クエリ応答形式            NR2 (固定小数点型)
- 説明                         CLOCK 入力終端電圧を設定します。  
この設定を有効にするためには、CLOCK 入力の終端電圧可変機能を ON に設定する必要があります。  
終端タイプが GND の場合は設定できません。

終端タイプ	設定範囲
PECL (+3V)	+2.70V ~ +3.30V
LVPECL (+1.3V)	+1.00V ~ +1.60V
ECL (-2V)	-2.30V ~ -1.70V
CML	0.00V ~ +3.50V

設定分解能 0.05V  
設定単位 V

## 5.5.4 SOURce8 サブシステム

### 1. SOURce8:TRIGger[:MODE]

- 機能 トリガ出力ソースの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce8:TRIGger[:MODE] <type>
- パラメータ <type> = {CLOCK16 | PATtern | FLEXible}
- クエリ応答形式 CLOC16 | PATT | FLEX
- 説明 トリガ出力のソースを設定します。

コマンド・パラメータ	トリガ出力のソース
CLOC16	入力クロックの 1/16 周波数のクロック信号
PATT	パターンに同期したトリガ
FLEX	パターン・シーケンス・テーブルの FLEX Trigger で設定したレベル パターンオプションが搭載され、パターンタイプが FLEX の場合に設定可能です。

### 2. SOURce8:TRIGger:STM:MODE (パターンオプション)

- 機能 STM パターン時のトリガ出力モードの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce8:TRIGger:STM:MODE <type>
- パラメータ <type> = {PATtern | FRAMe}
- クエリ応答形式 PATT | FRAM
- 説明 STM パターン時のトリガ出力モードを設定します。

コマンド・パラメータ	STM パターントリガ出力モード
PATT	パターン・モード パターンごとにトリガ出力を行います。
FRAM	フレーム・モード フレームごとにトリガ出力を行います。

5.5.5 SOURce9 サブシステム

5.5.5 SOURce9 サブシステム

1. SOURce9:AUX[:MODE]

- 機能 補助出力ソースの設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce9:AUX[:MODE] <type>
- パラメータ <type> = {DTYPe | SYNChronisat}
- クエリ応答形式 DTYP | SYNC
- 説明 補助出力のソースを設定します。

コマンド・パラメータ	補助出力のソース
DTYP	パターンのデータ・タイプ
SYNC	同期状態

## 5.6 結果、システム・コマンド

### 5.6.1 FETCh/CFETch サブシステム

FETCh/CFETch クエリは、ED モジュールから測定データを取得するために使用されます。FETCh クエリは測定結果を問い合わせ、CFETch クエリは測定途中経過の値を問い合わせます。例えば、リピート測定を行っている場合、FETCh:BME:ECO? クエリでは、前の測定期間に測定したエラー・カウント値を問い合わせ、CFETCh:BME:ECO? クエリでは、測定途中のエラー・カウント値を問い合わせます。

1. FETCh[:SENSe[1]]:BMEasurement:BCOunt[:TOTal]?  
 FETCh[:SENSe[1]]:BMEasurement:BCOunt:OMITting?  
 FETCh[:SENSe[1]]:BMEasurement:BCOunt:INSerting?  
 FETCh[:SENSe[1]]:BMEasurement:BCOunt:OVERhead? (パターンオプション)  
 FETCh[:SENSe[1]]:BMEasurement:BCOunt:PAYLoad? (パターンオプション)  
 FETCh[:SENSe[1]]:BMEasurement:BCOunt:SPECific?  
 FETCh[:SENSe[1]]:BMEasurement:BCOunt:OTHer?  
 CFETCh[:SENSe[1]]:BMEasurement:BCOunt[:TOTal]?  
 CFETCh[:SENSe[1]]:BMEasurement:BCOunt:OMITting?  
 CFETCh[:SENSe[1]]:BMEasurement:BCOunt:INSerting?  
 CFETCh[:SENSe[1]]:BMEasurement:BCOunt:OVERhead? (パターンオプション)  
 CFETCh[:SENSe[1]]:BMEasurement:BCOunt:PAYLoad? (パターンオプション)  
 CFETCh[:SENSe[1]]:BMEasurement:BCOunt:SPECific?  
 CFETCh[:SENSe[1]]:BMEasurement:BCOunt:OTHer?

- 機能 ビット・カウント値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ
  - FETCh[:SENSe[1]]:BMEasurement:BCOunt[:TOTal]?
  - FETCh[:SENSe[1]]:BMEasurement:BCOunt:OMITting?
  - FETCh[:SENSe[1]]:BMEasurement:BCOunt:INSerting?
  - FETCh[:SENSe[1]]:BMEasurement:BCOunt:OVERhead?
  - FETCh[:SENSe[1]]:BMEasurement:BCOunt:PAYLoad?
  - FETCh[:SENSe[1]]:BMEasurement:BCOunt:SPECific?
  - FETCh[:SENSe[1]]:BMEasurement:BCOunt:OTHer?
  - CFETCh[:SENSe[1]]:BMEasurement:BCOunt[:TOTal]?
  - CFETCh[:SENSe[1]]:BMEasurement:BCOunt:OMITting?
  - CFETCh[:SENSe[1]]:BMEasurement:BCOunt:INSerting?
  - CFETCh[:SENSe[1]]:BMEasurement:BCOunt:OVERhead?
  - CFETCh[:SENSe[1]]:BMEasurement:BCOunt:PAYLoad?
  - CFETCh[:SENSe[1]]:BMEasurement:BCOunt:SPECific?
  - CFETCh[:SENSe[1]]:BMEasurement:BCOunt:OTHer?

5.6.1 FETCh/CFETch サブシステム

- クエリ応答形式      表示フォーマットが整数形式の場合： NR1（整数型）  
表示フォーマットが指数形式の場合： NR3（浮動小数点型）
- 説明                    測定したビット・カウント値を問い合わせます。  
ヘッダ末尾のニーモニックにより、以下に示すビット・カウント値を問い合わせます。取得できるビット・カウント値はエラー検出モードの設定により異なります。

末尾のニーモニック	ビット・カウントの種類
[TOTal]	受信したビット・カウント値
OMITting	欠落エラーの測定対象となるビット・カウント値
INSerting	挿入エラーの測定対象となるビット・カウント値
OVERhead	オーバーヘッド領域のビット・カウント値
PAYLoad	ペイロード領域のビット・カウント値
SPECific	特定領域のビット・カウント値
OTHer	特定領域以外のビット・カウント値

データがオーバフロー、不定、無効時には、以下の値を返します。

表示フォーマット	オーバフロー	不定	無効
整数形式	9999999998	9999999999	9999999997
指数形式	9.9998E+98	9.9999E+99	9.9997E+97



```

2.  FETCh[:SENSe[1]]:BMEasurement:ERATe[:TOTal]?
    FETCh[:SENSe[1]]:BMEasurement:ERATe:OMITting?
    FETCh[:SENSe[1]]:BMEasurement:ERATe:INSerting?
    FETCh[:SENSe[1]]:BMEasurement:ERATe:OVERhead?      (パターンオプション)
    FETCh[:SENSe[1]]:BMEasurement:ERATe:PAYLoad?      (パターンオプション)
    FETCh[:SENSe[1]]:BMEasurement:ERATe:SPECific?
    FETCh[:SENSe[1]]:BMEasurement:ERATe:OTHer?
    CFETCh[:SENSe[1]]:BMEasurement:ERATe[:TOTal]?
    CFETCh[:SENSe[1]]:BMEasurement:ERATe:OMITting?
    CFETCh[:SENSe[1]]:BMEasurement:ERATe:INSerting?
    CFETCh[:SENSe[1]]:BMEasurement:ERATe:OVERhead?      (パターンオプション)
    CFETCh[:SENSe[1]]:BMEasurement:ERATe:PAYLoad?      (パターンオプション)
    CFETCh[:SENSe[1]]:BMEasurement:ERATe:SPECific?
    CFETCh[:SENSe[1]]:BMEasurement:ERATe:OTHer?

```

- 機能 エラー・レート値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ
 

```

FETCh[:SENSe[1]]:BMEasurement:ERATe[:TOTal]?
FETCh[:SENSe[1]]:BMEasurement:ERATe:OMITting?
FETCh[:SENSe[1]]:BMEasurement:ERATe:INSerting?
FETCh[:SENSe[1]]:BMEasurement:ERATe:OVERhead?
FETCh[:SENSe[1]]:BMEasurement:ERATe:PAYLoad?
FETCh[:SENSe[1]]:BMEasurement:ERATe:SPECific?
FETCh[:SENSe[1]]:BMEasurement:ERATe:OTHer?
CFETCh[:SENSe[1]]:BMEasurement:ERATe[:TOTal]?
CFETCh[:SENSe[1]]:BMEasurement:ERATe:OMITting?
CFETCh[:SENSe[1]]:BMEasurement:ERATe:INSerting?
CFETCh[:SENSe[1]]:BMEasurement:ERATe:OVERhead?
CFETCh[:SENSe[1]]:BMEasurement:ERATe:PAYLoad?
CFETCh[:SENSe[1]]:BMEasurement:ERATe:SPECific?
CFETCh[:SENSe[1]]:BMEasurement:ERATe:OTHer?

```
- クエリ応答形式 NR3 (浮動小数点型)

## 5.6.1 FETCH/CFETCH サブシステム

- 説明

測定したエラー・レート値を問い合わせます。  
ヘッダ末尾のモニターにより、以下に示すエラー・レート値を問い合わせます。取得できるエラー・レート値はエラー検出モードの設定により異なります。

末尾のモニター	エラー・レートの種類
[TOTAl]	Total エラー
OMITting	欠落エラー
INSerting	挿入エラー
OVERhead	オーバーヘッド領域のエラー
PAYLoad	ペイロード領域のエラー
SPECific	特定領域のエラー
OTHer	特定領域以外のエラー

データがオーバフローの場合は 9.9998E+98 を、不定の場合は 9.9999E+99 を、無効の場合は 9.9997E+97 を返します。

```

3.  FETCh[:SENSe[1]]:BMEasurement:ECOUNT[:TOTal]?
    FETCh[:SENSe[1]]:BMEasurement:ECOUNT:OMITting?
    FETCh[:SENSe[1]]:BMEasurement:ECOUNT:INSerting?
    FETCh[:SENSe[1]]:BMEasurement:ECOUNT:OVERhead? (パターンオプション)
    FETCh[:SENSe[1]]:BMEasurement:ECOUNT:PAYLoad? (パターンオプション)
    FETCh[:SENSe[1]]:BMEasurement:ECOUNT:SPECific?
    FETCh[:SENSe[1]]:BMEasurement:ECOUNT:OTHer?
    CFETCh[:SENSe[1]]:BMEasurement:ECOUNT[:TOTal]?
    CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:OMITting?
    CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:INSerting?
    CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:OVERhead? (パターンオプション)
    CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:PAYLoad? (パターンオプション)
    CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:SPECific?
    CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:OTHer?

```

- 機能 エラー・カウント値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ
 

```

FETCh[:SENSe[1]]:BMEasurement:ECOUNT[:TOTal]?
FETCh[:SENSe[1]]:BMEasurement:ECOUNT:OMITting?
FETCh[:SENSe[1]]:BMEasurement:ECOUNT:INSerting?
FETCh[:SENSe[1]]:BMEasurement:ECOUNT:OVERhead?
FETCh[:SENSe[1]]:BMEasurement:ECOUNT:PAYLoad?
FETCh[:SENSe[1]]:BMEasurement:ECOUNT:SPECific?
FETCh[:SENSe[1]]:BMEasurement:ECOUNT:OTHer?
CFETCh[:SENSe[1]]:BMEasurement:ECOUNT[:TOTal]?
CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:OMITting?
CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:INSerting?
CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:OVERhead?
CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:PAYLoad?
CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:SPECific?
CFETCh[:SENSe[1]]:BMEasurement:ECOUNT:OTHer?

```
- クエリ応答形式
 

表示フォーマットが整数形式の場合：	NR1 (整数型)
表示フォーマットが指数形式の場合：	NR3 (浮動小数点型)

## 5.6.1 FETCh/CFETch サブシステム

- 説明

測定したエラー・カウント値を問い合わせます。  
ヘッダ末尾のニーモニックにより、以下に示すエラー・カウント値を問い合わせます。取得できるエラー・カウント値はエラー検出モードの設定により異なります。

末尾のニーモニック	エラー・カウントの種類
[TOTal]	Total エラー
OMITting	欠落エラー
INSerting	挿入エラー
OVERhead	オーバーヘッド領域のエラー
PAYLoad	ペイロード領域のエラー
SPECific	特定領域のエラー
OTHer	特定領域以外のエラー

データがオーバフロー、不定、無効時には、以下の値を返します。

表示フォーマット	オーバフロー	不定	無効
整数形式	9999999998	9999999999	9999999997
指数形式	9.9998E+98	9.9999E+99	9.9997E+97

```

4. CFETch[:SENSe[1]]:BMEasurement:IERate[:TOTal]?
   CFETch[:SENSe[1]]:BMEasurement:IERate:OMITting?
   CFETch[:SENSe[1]]:BMEasurement:IERate:INSerting?
   CFETch[:SENSe[1]]:BMEasurement:IERate:OVERhead? (パターンオプション)
   CFETch[:SENSe[1]]:BMEasurement:IERate:PAYLoad? (パターンオプション)
   CFETch[:SENSe[1]]:BMEasurement:IERate:SPECific?
   CFETch[:SENSe[1]]:BMEasurement:IERate:OTHer?

```

- 機能 区間エラー・レート値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ CFETch[:SENSe[1]]:BMEasurement:IERate[:TOTal]?  
CFETch[:SENSe[1]]:BMEasurement:IERate:OMITting?  
CFETch[:SENSe[1]]:BMEasurement:IERate:INSerting?  
CFETch[:SENSe[1]]:BMEasurement:IERate:OVERhead?  
CFETch[:SENSe[1]]:BMEasurement:IERate:PAYLoad?  
CFETch[:SENSe[1]]:BMEasurement:IERate:SPECific?  
CFETch[:SENSe[1]]:BMEasurement:IERate:OTHer?
- クエリ応答形式 NR3 (浮動小数点型)
- 説明 測定した区間エラー・レート値を問い合わせます。  
ヘッダ末尾のニーマニックにより、以下に示す区間エラー・レート値を問い合わせます。取得できる区間エラー・レート値はエラー検出モードの設定により異なります。

末尾のニーマニック	区間エラー・レートの種類
[TOTal]	Total エラー
OMITting	欠落エラー
INSerting	挿入エラー
OVERhead	オーバーヘッド領域のエラー
PAYLoad	ペイロード領域のエラー
SPECific	特定領域のエラー
OTHer	特定領域以外のエラー

データがオーバフローの場合は 9.9998E+98 を、不定の場合は 9.9999E+99 を、無効の場合は 9.9997E+97 を返します。

5.6.1 FETCh/CFETch サブシステム

5. CFETch[:SENSe[1]]:BMEasurement:IECount[:TOTal]?  
 CFETch[:SENSe[1]]:BMEasurement:IECount:OMITting?  
 CFETch[:SENSe[1]]:BMEasurement:IECount:INSerting?  
 CFETch[:SENSe[1]]:BMEasurement:IECount:OVERhead? (パターンオプション)  
 CFETch[:SENSe[1]]:BMEasurement:IECount:PAYLoad? (パターンオプション)  
 CFETch[:SENSe[1]]:BMEasurement:IECount:SPECific?  
 CFETch[:SENSe[1]]:BMEasurement:IECount:OTHer?

- 機能 区間エラー・カウント値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ CFETch[:SENSe[1]]:BMEasurement:IECount[:TOTal]?  
 CFETch[:SENSe[1]]:BMEasurement:IECount:OMITting?  
 CFETch[:SENSe[1]]:BMEasurement:IECount:INSerting?  
 CFETch[:SENSe[1]]:BMEasurement:IECount:OVERhead?  
 CFETch[:SENSe[1]]:BMEasurement:IECount:PAYLoad?  
 CFETch[:SENSe[1]]:BMEasurement:IECount:SPECific?  
 CFETch[:SENSe[1]]:BMEasurement:IECount:OTHer?
- クエリ応答形式 表示フォーマットが整数形式の場合: NR1 (整数型)  
 表示フォーマットが指数形式の場合: NR3 (浮動小数点型)
- 説明 測定した区間エラー・カウント値を問い合わせます。  
 ヘッダ末尾のニーマニックにより、以下に示す区間エラー・カウント値を問い合わせます。取得できる区間エラー・カウント値はエラー検出モードの設定により異なります。

末尾のニーマニック	区間エラー・カウントの種類
[TOTal]	Total エラー
OMITting	欠落エラー
INSerting	挿入エラー
OVERhead	オーバーヘッド領域のエラー
PAYLoad	ペイロード領域のエラー
SPECific	特定領域のエラー
OTHer	特定領域以外のエラー

データがオーバフロー、不定、無効時には、以下の値を返します。

表示フォーマット	オーバフロー	不定	無効
整数形式	9999999998	9999999999	9999999997
指数形式	9.9998E+98	9.9999E+99	9.9997E+97

```

6.  FETCh[:SENSe[1]]:BMEasurement:EINTervals[:TOTal]?
    FETCh[:SENSe[1]]:BMEasurement:EINTervals:OMITting?
    FETCh[:SENSe[1]]:BMEasurement:EINTervals:INSerting?
    FETCh[:SENSe[1]]:BMEasurement:EINTervals:OVERhead? (パターンオプション)
    FETCh[:SENSe[1]]:BMEasurement:EINTervals:PAYLoad? (パターンオプション)
    FETCh[:SENSe[1]]:BMEasurement:EINTervals:SPECific?
    FETCh[:SENSe[1]]:BMEasurement:EINTervals:OTHer?
    CFETCh[:SENSe[1]]:BMEasurement:EINTervals[:TOTal]?
    CFETCh[:SENSe[1]]:BMEasurement:EINTervals:OMITting?
    CFETCh[:SENSe[1]]:BMEasurement:EINTervals:INSerting?
    CFETCh[:SENSe[1]]:BMEasurement:EINTervals:OVERhead? (パターンオプション)
    CFETCh[:SENSe[1]]:BMEasurement:EINTervals:PAYLoad? (パターンオプション)
    CFETCh[:SENSe[1]]:BMEasurement:EINTervals:SPECific?
    CFETCh[:SENSe[1]]:BMEasurement:EINTervals:OTHer?

```

- 機能 エラー・インターバル値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ
 

```

FETCh[:SENSe[1]]:BMEasurement:EINTervals[:TOTal]?
FETCh[:SENSe[1]]:BMEasurement:EINTervals:OMITting?
FETCh[:SENSe[1]]:BMEasurement:EINTervals:INSerting?
FETCh[:SENSe[1]]:BMEasurement:EINTervals:OVERhead?
FETCh[:SENSe[1]]:BMEasurement:EINTervals:PAYLoad?
FETCh[:SENSe[1]]:BMEasurement:EINTervals:SPECific?
FETCh[:SENSe[1]]:BMEasurement:EINTervals:OTHer?
CFETCh[:SENSe[1]]:BMEasurement:EINTervals[:TOTal]?
CFETCh[:SENSe[1]]:BMEasurement:EINTervals:OMITting?
CFETCh[:SENSe[1]]:BMEasurement:EINTervals:INSerting?
CFETCh[:SENSe[1]]:BMEasurement:EINTervals:OVERhead?
CFETCh[:SENSe[1]]:BMEasurement:EINTervals:PAYLoad?
CFETCh[:SENSe[1]]:BMEasurement:EINTervals:SPECific?
CFETCh[:SENSe[1]]:BMEasurement:EINTervals:OTHer?

```
- クエリ応答形式 表示フォーマットが整数形式の場合： NR1 (整数型)  
表示フォーマットが百分率形式の場合： NR2 (固定小数点型)

## 5.6.1 FETCh/CFETch サブシステム

- 説明

測定したエラー・インターバル値を問い合わせます。  
ヘッダ末尾のニーモニックにより、以下に示すエラー・インターバル値を問い合わせます。取得できるエラー・インターバル値はエラー検出モードの設定により異なります。

末尾のニーモニック	エラー・インターバルの種類
[TOTal]	Total エラー
OMITting	欠落エラー
INSerting	挿入エラー
OVERhead	オーバーヘッド領域のエラー
PAYLoad	ペイロード領域のエラー
SPECific	特定領域のエラー
OTHer	特定領域以外のエラー

データがオーバフロー、不定、無効時には、以下の値を返します。

表示フォーマット	オーバフロー	不定	無効
整数形式	9999999998	9999999999	9999999997
百分率形式	999.9998	999.9999	999.9997



```

7.  FETCh[:SENSe[1]]:BMEasurement:EFINtervals[:TOTal]?
    FETCh[:SENSe[1]]:BMEasurement:EFINtervals:OMITting?
    FETCh[:SENSe[1]]:BMEasurement:EFINtervals:INSerting?
    FETCh[:SENSe[1]]:BMEasurement:EFINtervals:OVERhead? (パターンオプション)
    FETCh[:SENSe[1]]:BMEasurement:EFINtervals:PAYLoad? (パターンオプション)
    FETCh[:SENSe[1]]:BMEasurement:EFINtervals:SPECific?
    FETCh[:SENSe[1]]:BMEasurement:EFINtervals:OTHer?
    CFETCh[:SENSe[1]]:BMEasurement:EFINtervals[:TOTal]?
    CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:OMITting?
    CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:INSerting?
    CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:OVERhead? (パターンオプション)
    CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:PAYLoad? (パターンオプション)
    CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:SPECific?
    CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:OTHer?

```

- 機能 エラー・フリー・インターバル値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ
 

```

FETCh[:SENSe[1]]:BMEasurement:EFINtervals[:TOTal]?
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:OMITting?
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:INSerting?
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:OVERhead?
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:PAYLoad?
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:SPECific?
FETCh[:SENSe[1]]:BMEasurement:EFINtervals:OTHer?
CFETCh[:SENSe[1]]:BMEasurement:EFINtervals[:TOTal]?
CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:OMITting?
CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:INSerting?
CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:OVERhead?
CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:PAYLoad?
CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:SPECific?
CFETCh[:SENSe[1]]:BMEasurement:EFINtervals:OTHer?

```
- クエリ応答形式 表示フォーマットが整数形式の場合： NR1 (整数型)  
表示フォーマットが百分率形式の場合： NR2 (固定小数点型)

5.6.1 FETCh/CFETch サブシステム

- 説明

測定したエラー・フリー・インターバル値を問い合わせます。ヘッダ末尾のニーモニックにより、以下に示すエラー・フリー・インターバル値を問い合わせます。取得できるエラー・フリー・インターバル値はエラー検出モードの設定により異なります。

末尾のニーモニック	エラー・フリー・インターバルの種類
[TOTAl]	Total エラー
OMITting	欠落エラー
INSerting	挿入エラー
OVERhead	オーバーヘッド領域のエラー
PAYLoad	ペイロード領域のエラー
SPECific	特定領域のエラー
OTHer	特定領域以外のエラー

データがオーバフロー、不定、無効時には、以下の値を返します。

表示フォーマット	オーバフロー	不定	無効
整数形式	9999999998	9999999999	9999999997
百分率形式	999.9998	999.9999	999.9997

8. FETCh[:SENSe[1]]:BMEasurement:FREQuency?  
CFETCh[:SENSe[1]]:BMEasurement:FREQuency?

- 機能 周波数値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSe[1]]:BMEasurement:FREQuency?  
CFETCh[:SENSe[1]]:BMEasurement:FREQuency?
- クエリ応答形式 NR1 (整数型)
- 説明 測定した周波数値を Hz 単位で返します。  
不定の場合は 999999999 を返します。

9. FETCh[:SENSe[1]]:BMEasurement:CLINtervals?  
CFETCh[:SENSe[1]]:BMEasurement:CLINtervals?

- 機能 クロックロス・インターバル値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSe[1]]:BMEasurement:CLINtervals?  
CFETCh[:SENSe[1]]:BMEasurement:CLINtervals?
- クエリ応答形式 NR1 (整数型)
- 説明 測定したクロックロス・インターバル値を問い合わせます。  
データがオーバフローの場合は 999999998 を、不定の場合は  
999999999 を、無効の場合は 999999997 を返します。

10. FETCh[:SENSe[1]]:BMEasurement:SLINtervals?  
CFETCh[:SENSe[1]]:BMEasurement:SLINtervals?

- 機能 シンクロス・インターバル値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSe[1]]:BMEasurement:SLINtervals?  
CFETCh[:SENSe[1]]:BMEasurement:SLINtervals?
- クエリ応答形式 NR1 (整数型)
- 説明 測定したシンクロス・インターバル値を問い合わせます。  
データがオーバフローの場合は 999999998 を、不定の場合は  
999999999 を、無効の場合は 999999997 を返します。

5.6.1 FETCh/CFETch サブシステム

11. FETCh[:SENSe[1]]:BMEasurement:TEINtervals:EM<n>[:TOTal]?  
 FETCh[:SENSe[1]]:BMEasurement:TEINtervals:LEM8[:TOTal]?  
 CFETch[:SENSe[1]]:BMEasurement:TEINtervals:EM<n>[:TOTal]?  
 CFETch[:SENSe[1]]:BMEasurement:TEINtervals:LEM8[:TOTal]?

- 機能 スレッシュヨルド EI 値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSe[1]]:BMEasurement:TEINtervals:EM<n>[:TOTal]?  
 FETCh[:SENSe[1]]:BMEasurement:TEINtervals:LEM8[:TOTal]?  
 CFETch[:SENSe[1]]:BMEasurement:TEINtervals:EM<n>[:TOTal]?  
 CFETch[:SENSe[1]]:BMEasurement:TEINtervals:LEM8[:TOTal]?  
 <n> = 3, 4, 5, 6, 7, 8
- クエリ応答形式 表示フォーマットが整数形式の場合： NR1（整数型）  
 表示フォーマットが百分率形式の場合： NR2（固定小数点型）
- 説明 測定したスレッシュヨルド EI 値を問い合わせます。  
 スレッシュヨルド EI 値は、1 インターバルのエラー・レートがスレッシュヨルド条件を満足するインターバル数のことです。  
 ヘッダ末尾のニーモニックにより、以下に示すスレッシュヨルド条件を指定します。

末尾のニーモニック	スレッシュヨルド条件
EM3[:TOTal]	> 1E-3
EM4[:TOTal]	> 1E-4
EM5[:TOTal]	> 1E-5
EM6[:TOTal]	> 1E-6
EM7[:TOTal]	> 1E-7
EM8[:TOTal]	> 1E-8
LEM8[:TOTal]	≤ 1E-8

データがオーバフロー、不定、無効時には、以下の値を返します。

表示フォーマット	オーバフロー	不定	無効
整数形式	999999998	999999999	999999997
百分率形式	999.9998	999.9999	999.9997

```

12. FETCh[:SENSe[1]]:BMEasurement:TEFintervals:EM<n>[:TOTal]?
    FETCh[:SENSe[1]]:BMEasurement:TEFintervals:LEM8[:TOTal]?
    CFETCh[:SENSe[1]]:BMEasurement:TEFintervals:EM<n>[:TOTal]?
    CFETCh[:SENSe[1]]:BMEasurement:TEFintervals:LEM8[:TOTal]?

```

- 機能 スレッシュールド EFI 値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSe[1]]:BMEasurement:TEFintervals:EM<n>[:TOTal]?  
FETCh[:SENSe[1]]:BMEasurement:TEFintervals:LEM8[:TOTal]?  
CFETCh[:SENSe[1]]:BMEasurement:TEFintervals:EM<n>[:TOTal]?  
CFETCh[:SENSe[1]]:BMEasurement:TEFintervals:LEM8[:TOTal]?
- パラメータ <n> = 3, 4, 5, 6, 7, 8
- クエリ応答形式 表示フォーマットが整数形式の場合: NR1 (整数型)  
表示フォーマットが百分率形式の場合: NR2 (固定小数点型)
- 説明 測定したスレッシュールド EFI 値を問い合わせます。  
スレッシュールド EFI 値は、1 インターバルのエラー・レートがスレッシュールド条件を満足しないインターバル数のことです。  
ヘッダ末尾のニーマニックにより、以下に示すスレッシュールド条件を指定します。

末尾のニーマニック	スレッシュールド条件
EM3[:TOTal]	> 1E-3
EM4[:TOTal]	> 1E-4
EM5[:TOTal]	> 1E-5
EM6[:TOTal]	> 1E-6
EM7[:TOTal]	> 1E-7
EM8[:TOTal]	> 1E-8
LEM8[:TOTal]	≤ 1E-8

データがオーバフロー、不定、無効時には、以下の値を返します。

表示フォーマット	オーバフロー	不定	無効
整数形式	9999999998	9999999999	9999999997
百分率形式	999.9998	999.9999	999.9997

5.6.1 FETCh/CFETCh サブシステム

```

13. FETCh[:SENSe[1]]:BMEasurement:EPERformance:ESECONDS[:TOTal]?
    FETCh[:SENSe[1]]:BMEasurement:EPERformance:EFSeconds[:TOTal]?
    FETCh[:SENSe[1]]:BMEasurement:EPERformance:SESeconds[:TOTal]?
    FETCh[:SENSe[1]]:BMEasurement:EPERformance:USECONDS[:TOTal]?
    FETCh[:SENSe[1]]:BMEasurement:EPERformance:DMINutes[:TOTal]?
    CFETCh[:SENSe[1]]:BMEasurement:EPERformance:ESECONDS[:TOTal]?
    CFETCh[:SENSe[1]]:BMEasurement:EPERformance:EFSeconds[:TOTal]?
    CFETCh[:SENSe[1]]:BMEasurement:EPERformance:SESeconds[:TOTal]?
    CFETCh[:SENSe[1]]:BMEasurement:EPERformance:USECONDS[:TOTal]?
    CFETCh[:SENSe[1]]:BMEasurement:EPERformance:DMINutes[:TOTal]?
    
```

- 機能 エラー・パフォーマンス値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ
  - FETCh[:SENSe[1]]:BMEasurement:EPERformance:ESECONDS[:TOTal]?
  - FETCh[:SENSe[1]]:BMEasurement:EPERformance:EFSeconds[:TOTal]?
  - FETCh[:SENSe[1]]:BMEasurement:EPERformance:SESeconds[:TOTal]?
  - FETCh[:SENSe[1]]:BMEasurement:EPERformance:USECONDS[:TOTal]?
  - FETCh[:SENSe[1]]:BMEasurement:EPERformance:DMINutes[:TOTal]?
  - CFETCh[:SENSe[1]]:BMEasurement:EPERformance:ESECONDS[:TOTal]?
  - CFETCh[:SENSe[1]]:BMEasurement:EPERformance:EFSeconds[:TOTal]?
  - CFETCh[:SENSe[1]]:BMEasurement:EPERformance:SESeconds[:TOTal]?
  - CFETCh[:SENSe[1]]:BMEasurement:EPERformance:USECONDS[:TOTal]?
  - CFETCh[:SENSe[1]]:BMEasurement:EPERformance:DMINutes[:TOTal]?
- クエリ応答形式 表示フォーマットが整数形式の場合： NR1（整数型）  
表示フォーマットが百分率形式の場合： NR2（固定小数点型）
- 説明 測定したエラー・パフォーマンス値を問い合わせます。  
ヘッダ末尾のニーモニックにより、以下に示すエラー・パフォーマンス値を問い合わせます。

末尾のニーモニック	エラー・パフォーマンス項目
ESECONDS[:TOTal]	誤り秒 (Error Seconds)
EFSeconds[:TOTal]	誤りなし秒 (Error Free Seconds)
SESeconds[:TOTal]	異常誤り秒 (Severely Errored Seconds)
USECONDS[:TOTal]	不稼働時間 (Unavailable Seconds)
DMINutes[:TOTal]	劣化分 (Degraded Minutes)

データがオーバーフロー、不定、無効時には、以下の値を返します。

表示フォーマット	オーバーフロー	不定	無効
整数形式	999999998	999999999	999999997
百分率形式	999.9998	999.9999	999.9997

14. FETCh[:SENSe[1]]:BMEasurement:B1ERror:BCOunt? (パターンオプション)  
 CFETCh[:SENSe[1]]:BMEasurement:B1ERror:BCOunt? (パターンオプション)

- 機能 総フレーム・ビット数の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSe[1]]:BMEasurement:B1ERror:BCOunt?  
CFETCh[:SENSe[1]]:BMEasurement:B1ERror:BCOunt?
- クエリ応答形式 表示フォーマットが整数形式の場合: NR1 (整数型)  
表示フォーマットが指数形式の場合: NR3 (浮動小数点型)
- 説明 総フレーム・ビット数を問い合わせます。  
総フレーム・ビット数は下式により算出されます。  
総フレーム・ビット数 = B1 バイト測定したフレーム数 × 1 フレームのビット数  
データがオーバーフロー、不定、無効時には、以下の値を返します。

表示フォーマット	オーバーフロー	不定	無効
整数形式	9999999998	9999999999	9999999997
指数形式	9.9998E+98	9.9999E+99	9.9997E+97

15. FETCh[:SENSe[1]]:BMEasurement:B1ERror:ERATe? (パターンオプション)  
 CFETCh[:SENSe[1]]:BMEasurement:B1ERror:ERATe? (パターンオプション)

- 機能 B1 バイト測定のエラー・レート値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSe[1]]:BMEasurement:B1ERror:ERATe?  
CFETCh[:SENSe[1]]:BMEasurement:B1ERror:ERATe?
- クエリ応答形式 NR3 (浮動小数点型)
- 説明 測定した B1 バイトのエラー・レート値を問い合わせます。  
データがオーバーフローの場合は 9.9998E+98 を、不定の場合は 9.9999E+99 を、無効の場合は 9.9997E+97 を返します。

5.6.1 FETCh/CFETch サブシステム

16. FETCh[:SENSE[1]]:BMEasurement:B1ERror:ECOut? (パターンオプション)  
 CFETch[:SENSE[1]]:BMEasurement:B1ERror:ECOut? (パターンオプション)

- 機能 B1 バイト測定のエラー・カウント値の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSE[1]]:BMEasurement:B1ERror:ECOut?  
CFETch[:SENSE[1]]:BMEasurement:B1ERror:ECOut?
- クエリ応答形式 表示フォーマットが整数形式の場合: NR1 (整数型)  
表示フォーマットが指数形式の場合: NR3 (浮動小数点型)
- 説明 測定した B1 バイトのエラー・カウント値を問い合わせます。  
データがオーバーフロー、不定、無効時には、以下の値を返します。

表示フォーマット	オーバーフロー	不定	無効
整数形式	999999998	999999999	999999997
指数形式	9.9998E+98	9.9999E+99	9.9997E+97

17. FETCh[:SENSE[1]]:MTIME:ETIME?

- 機能 測定経過時間の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSE[1]]:MTIME:ETIME?
- クエリ応答形式 <day>, <hour>, <minute>, <second>  
<day> = <hour> = <minute> = <second> = NR1 (整数型)
- 説明 測定経過時間を問い合わせます。  
測定経過時間がオーバーフローの場合、98, 98, 98, 98 を返します。

18. FETCh[:SENSE[1]]:MTIME:TIMEd?

- 機能 測定残り時間の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSE[1]]:MTIME:TIMEd?
- クエリ応答形式 <day>, <hour>, <minute>, <second>  
<day> = <hour> = <minute> = <second> = NR1 (整数型)
- 説明 測定残り時間を問い合わせます。  
測定残り時間が不定の場合は 99, 99, 99, 99 を返します。



## 19. FETCh[:SENSe[1]]:MTIME:STIME?

- 機能 測定開始時間の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSe[1]]:MTIME:STIME?
- クエリ応答形式 <year>, <month>, <day>, <hour>, <minute>, <second>  
<year> = <month> = <day> = <hour> = <minute> = <second> =  
NR1 (整数型)
- 説明 測定開始時間を問い合わせます。  
測定開始時間が不定の場合は 9999, 99, 99, 99, 99, 99 を返します。

## 20. FETCh[:SENSe[1]]:MTIME:RTIME?

- 機能 現在の日付と時刻の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSe[1]]:MTIME:RTIME?
- クエリ応答形式 <year>, <month>, <day>, <hour>, <minute>, <second>  
<year> = <month> = <day> = <hour> = <minute> = <second> =  
NR1 (整数型)
- 説明 現在の日付と時刻を問い合わせます。

5.6.1 FETCh/CFETCh サブシステム

21. FETCh[:SENSe[1]]:MONitor:RTIME:DERRor?  
 FETCh[:SENSe[1]]:MONitor:RTIME:CLOSs?  
 FETCh[:SENSe[1]]:MONitor:RTIME:SLOSs?  
 FETCh[:SENSe[1]]:MONitor:HISTory:DERRor?  
 FETCh[:SENSe[1]]:MONitor:HISTory:CLOSs?  
 FETCh[:SENSe[1]]:MONitor:HISTory:SLOSs?

- 機能 モニタ・ステータスの問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh[:SENSe[1]]:MONitor:RTIME:DERRor?  
 FETCh[:SENSe[1]]:MONitor:RTIME:CLOSs?  
 FETCh[:SENSe[1]]:MONitor:RTIME:SLOSs?  
 FETCh[:SENSe[1]]:MONitor:HISTory:DERRor?  
 FETCh[:SENSe[1]]:MONitor:HISTory:CLOSs?  
 FETCh[:SENSe[1]]:MONitor:HISTory:SLOSs?
- クエリ応答形式 0 | 1
- 説明 モニタ・ステータスを問い合わせます。  
 モニタ・ステータスにはリアルタイムと履歴の2種類があります。リアルタイムでは、現在の状態を問い合わせ、履歴では、測定中にその事象が発生したかどうかについて問い合わせます。ヘッダ末尾のニーモニックにより、以下に示すモニタ・ステータスを問い合わせます。

末尾のニーモニック	モニタ・ステータス
RTIME:DERRor	リアルタイムのデータ・エラー発生
RTIME:CLOSs	リアルタイムのクロックロス発生
RTIME:SLOSs	リアルタイムのシンクロス発生
HISTory:DERRor	履歴のデータ・エラー発生
HISTory:CLOSs	履歴のクロックロス発生
HISTory:SLOSs	履歴のシンクロス発生

リアルタイムでは、事象が発生しているときに1を返し、履歴では、事象が発生したときに1を返します。

22. FETCh[:SENSe[1]]:EPANalysis:MDATa:TSERies:ERRor:COUNT? (エラー位相解析オプション)
- 機能 エラー記録の時系列フォーマットにおけるエラー行数の問い合わせ
  - コマンドとクエリの存在 Query
  - クエリ FETCh[:SENSe[1]]:EPANalysis:MDATa:TSERies:ERRor:COUNT?
  - クエリ応答形式 NR1 (整数型)
  - 説明 エラー記録の時系列フォーマットにおけるエラー行数を問い合わせます。
23. FETCh[:SENSe[1]]:EPANalysis:MDATa:TSERies:CYCLe:COUNT? (エラー位相解析オプション)
- 機能 エラー記録の時系列フォーマットにおけるパターン周期数の問い合わせ
  - コマンドとクエリの存在 Query
  - クエリ FETCh[:SENSe[1]]:EPANalysis:MDATa:TSERies:CYCLe:COUNT?
  - クエリ応答形式 NR1 (整数型)
  - 説明 エラー記録の時系列フォーマットにおけるパターン周期数を問い合わせます。
24. FETCh:JTOLerance:MDATa:JFRequency:COUNT? (ジッタ耐力オプション)
- 機能 ジッタ耐力測定の対象となったジッタ周波数の数の問い合わせ
  - コマンドとクエリの存在 Query
  - クエリ FETCh:JTOLerance:MDATa:JFRequency:COUNT?
  - クエリ応答形式 NR1 (整数型)
  - 説明 ジッタ耐力測定の対象となったジッタ周波数の数を問い合わせます。

5.6.1 FETCh/CFETCh サブシステム

25. FETCh:JTOLerance:MDATa:JAMplitude:COUNT? (ジッタ耐力オプション)

- 機能 ジッタ耐力測定の対象となったジッタ振幅の数の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh:JTOLerance:MDATa:JAMplitude:COUNT? <index\_freq>
- パラメータ <index\_freq> = 測定ジッタ周波数を指定するインデックス値  
 設定範囲 1 ~ (ジッタ耐力測定の対象となったジッタ周波数の数)  
 設定分解能 1
- クエリ応答形式 NR1 (整数型)
- 説明 1つのジッタ周波数において、ジッタ耐力測定の対象となったジッタ振幅の数を問い合わせます。

26. FETCh:JTOLerance:MDATa? (ジッタ耐力オプション)

- 機能 ジッタ耐力測定データの問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh:JTOLerance:MDATa? <index\_freq>,<index\_ampl>
- パラメータ <index\_freq> = 測定ジッタ周波数を指定するインデックス値  
 設定範囲 1 ~ (ジッタ耐力測定の対象となったジッタ周波数の数)  
 設定分解能 1  
 <index\_ampl> = <index\_freq>における測定ジッタ振幅を指定するインデックス値  
 設定範囲 1 ~ (<index\_freq>において対象となったジッタ振幅の数)  
 ただし、サーチモードのときは常に1を設定します。  
 設定分解能 1
- クエリ応答形式 <str> = 1ポイントの測定結果  
 "FREQ,M,AMPL,RES"  
 FREQ ジッタ周波数 [Hz] 10 ~ 20000000  
 M 修飾子 (必要がないときは1スペース)  
 > 測定限界 (上限)  
 < 測定限界 (下限) または測定不可  
 スweep・モードのときは常に1スペースとなります。  
 AMPL ジッタ振幅 [UIp-p] 0.01 ~ 800.00

RES 結果

測定モード	RES (結果)	意味
サーチ・モード	OK	ジッタ耐力点のジッタ振幅がテンプレートより大きい
	NG	ジッタ耐力点のジッタ振幅がテンプレート以下
	FAIL	測定不可
	----	テンプレートなし、または範囲外
スイープ・モード	O (アルファベット大文字)	指定したスレッショルド未満
	X (アルファベット大文字)	指定したスレッショルド以上
	FAIL	測定不可

- 説明 1 ポイントあたりのジッタ耐力測定データを問い合わせます。

27. FETCh:JTOLerance:MTIME:ETIME? (ジッタ耐力オプション)

- 機能 ジッタ耐力測定の間隔時間の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh:JTOLerance:MTIME:ETIME?
- クエリ応答形式 <day>,<hour>,<minute>,<second>  
<day> = <hour> = <minute> = <second> = NR1 (整数型)
- 説明 ジッタ耐力測定の間隔時間を問い合わせます。

28. FETCh:JTOLerance:MTIME:STIME? (ジッタ耐力オプション)

- 機能 ジッタ耐力測定の開始時間の問い合わせ
- コマンドとクエリの存在 Query
- クエリ FETCh:JTOLerance:MTIME:STIME?
- クエリ応答形式 <year>,<month>,<day>,<hour>,<minute>,<second>  
<year>=<month>=<day>=<hour>=<minute>=<second>= NR1 (整数型)
- 説明 ジッタ耐力測定の開始時間を問い合わせます。  
測定開始時間が不定の場合は 9999,99,99,99,99,99 を返します。

### 5.6.2 DISPlay サブシステム

1. DISPlay:WINDow:BCOunt:DFORmat

- 機能                                      ビット・カウント値の表示フォーマットの設定
- コマンドとクエリの存在      Command/Query
- コマンド                                DISPlay:WINDow:BCOunt:DFORmat <type>
- パラメータ                            <type> = {INTegral | EXPonential}
- クエリ応答形式                    INT | EXP
- 説明                                    測定したビット・カウント値の表示フォーマットを設定します。表示フォーマットを変更すると、結果画面の表示フォーマットとFETCh/CFETch サブシステムのクエリで返すビット・カウント値の応答形式が変わります。B1 測定のビット・カウント値の表示フォーマットも同時に変わります。

コマンド・パラメータ	表示フォーマット
INT	整数形式
EXP	指数形式

2. DISPlay:WINDow:ECOunt:DFORmat

- 機能                                      エラー・カウント値の表示フォーマットの設定
- コマンドとクエリの存在      Command/Query
- コマンド                                DISPlay:WINDow:ECOunt:DFORmat <type>
- パラメータ                            <type> = {INTegral | EXPonential}
- クエリ応答形式                    INT | EXP
- 説明                                    測定したエラー・カウント値の表示フォーマットを設定します。表示フォーマットを変更すると、結果画面の表示フォーマットとFETCh/CFETch サブシステムのクエリで返すエラー・カウント値の応答形式が変わります。B1 測定のエラー・カウント値の表示フォーマットも同時に変わります。

コマンド・パラメータ	表示フォーマット
INT	整数形式
EXP	指数形式

## 3. DISPlay:WINDow:IECount:DFORmat

- 機能 区間エラー・カウント値の表示フォーマットの設定
- コマンドとクエリの存在 Command/Query
- コマンド DISPlay:WINDow:IECount:DFORmat <type>
- パラメータ <type> = {INTegral | EXPonential}
- クエリ応答形式 INT | EXP
- 説明 測定した区間エラー・カウント値の表示フォーマットを設定します。  
表示フォーマットを変更すると、結果画面の表示フォーマットとCFETch サブシステムのクエリで返す区間エラー・カウント値の応答形式が変わります。

コマンド・パラメータ	表示フォーマット
INT	整数形式
EXP	指数形式

## 4. DISPlay:WINDow:EINTervals:DFORmat

- 機能 エラー・インターバル値の表示フォーマットの設定
- コマンドとクエリの存在 Command/Query
- コマンド DISPlay:WINDow:EINTervals:DFORmat <type>
- パラメータ <type> = {INTegral | PERCent}
- クエリ応答形式 INT | PERC
- 説明 測定したエラー・インターバル値の表示フォーマットを設定します。  
表示フォーマットを変更すると、結果画面の表示フォーマットとFETCh/CFETch サブシステムのクエリで返すエラー・インターバル値の応答形式が変わります。

コマンド・パラメータ	表示フォーマット
INT	整数形式
PERC	百分率形式

5.6.2 DISPlay サブシステム

5. DISPlay:WINDow:EFINtervals:DFORmat

- 機能 エラー・フリー・インターバル値の表示フォーマットの設定
- コマンドとクエリの存在 Command/Query
- コマンド DISPlay:WINDow:EFINtervals:DFORmat <type>
- パラメータ <type> = {INTegral | PERCent}
- クエリ応答形式 INT | PERC
- 説明 測定したエラー・フリー・インターバル値の表示フォーマットを設定します。  
表示フォーマットを変更すると、結果画面の表示フォーマットと FETCh/CFETch サブシステムのクエリで返すエラー・フリー・インターバル値の応答形式が変わります。

コマンド・パラメータ	表示フォーマット
INT	整数形式
PERC	百分率形式

6. DISPlay:WINDow:TEINtervals:DFORmat

- 機能 スレッシュヨルド EI 値の表示フォーマットの設定
- コマンドとクエリの存在 Command/Query
- コマンド DISPlay:WINDow:TEINtervals:DFORmat <type>
- パラメータ <type> = {INTegral | PERCent}
- クエリ応答形式 INT | PERC
- 説明 測定したスレッシュヨルドEI値の表示フォーマットを設定します。  
表示フォーマットを変更すると、結果画面の表示フォーマットと FETCh/CFETch サブシステムのクエリで返すスレッシュヨルド EI 値の応答形式が変わります。

コマンド・パラメータ	表示フォーマット
INT	整数形式
PERC	百分率形式



## 7. DISPlay:WINDow:TEFintervals:DFORmat

- 機能 スレッシュホールド EFI 値の表示フォーマットの設定
- コマンドとクエリの存在 Command/Query
- コマンド DISPlay:WINDow:TEFintervals:DFORmat <type>
- パラメータ <type> = {INTegral | PERCent}
- クエリ応答形式 INT | PERC
- 説明 測定したスレッシュホールド EFI 値の表示フォーマットを設定します。  
表示フォーマットを変更すると、結果画面の表示フォーマットと FETCh/CFETch サブシステムのクエリで返すスレッシュホールド EFI 値の応答形式が変わります。

コマンド・パラメータ	表示フォーマット
INT	整数形式
PERC	百分率形式

## 8. DISPlay:WINDow:EPERformance:DFORmat

- 機能 エラー・パフォーマンス値の表示フォーマットの設定
- コマンドとクエリの存在 Command/Query
- コマンド DISPlay:WINDow:EPERformance:DFORmat <type>
- パラメータ <type> = {INTegral | PERCent}
- クエリ応答形式 INT | PERC
- 説明 測定したエラー・パフォーマンス値の表示フォーマットを設定します。  
表示フォーマットを変更すると、結果画面の表示フォーマットと FETCh/CFETch サブシステムのクエリで返すエラー・パフォーマンス値の応答形式が変わります。

コマンド・パラメータ	表示フォーマット
INT	整数形式
PERC	百分率形式

### 5.6.3 HCOPy サブシステム

1. HCOPy[:ITEM]:ED:BMEasurement

- 機能 基本測定結果の印刷
- コマンドとクエリの存在 **Command**
- コマンド HCOPy[:ITEM]:ED:BMEasurement
- 説明 基本測定結果を印刷します。

2. HCOPy[:ITEM]:JTOLerance (ジッタ耐力オプション)

- 機能 ジッタ耐力測定結果の印刷
- コマンドとクエリの存在 **Command**
- コマンド HCOPy[:ITEM]:JTOLerance
- 説明 ジッタ耐力測定結果を印刷します。

## 5.6.4 MMEMemory サブシステム

### 1. MMEMemory:LOAD:SETup

- 機能 D3371 設定ファイルの読み込み
- コマンドとクエリの存在 Command
- コマンド MMEMemory:LOAD:SETup <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 D3371 設定ファイルを読み込みます。

### 2. MMEMemory:LOAD:SSG:SETup

- 機能 SSG モジュールの設定ファイルの読み込み
- コマンドとクエリの存在 Command
- コマンド MMEMemory:LOAD:SSG:SETup <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 SSG モジュールの設定ファイルを読み込みます。

### 3. MMEMemory:LOAD:PPG:SETup

- 機能 PPG モジュールの設定ファイルの読み込み
- コマンドとクエリの存在 Command
- コマンド MMEMemory:LOAD:PPG:SETup <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 PPG モジュールの設定ファイルを読み込みます。

## 5.6.4 MMEMemory サブシステム

## 4. MMEMemory:LOAD:PPG:PROGram

- 機能 PROG パターン・ファイルの PPG モジュールへの読み込み
- コマンドとクエリの存在 Command
- コマンド MMEMemory:LOAD:PPG:PROGram <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 PROG パターン・ファイルを PPG モジュールに読み込みます。

## 5. MMEMemory:LOAD:PPG:STM (パターンオプション)

- 機能 STM パターン・ファイルの PPG モジュールへの読み込み
- コマンドとクエリの存在 Command
- コマンド MMEMemory:LOAD:PPG:STM <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 STM パターン・ファイルを PPG モジュールに読み込みます。

## 6. MMEMemory:LOAD:PPG:FLEXible (パターンオプション)

- 機能 FLEX パターン・ファイルの PPG モジュールへの読み込み
- コマンドとクエリの存在 Command
- コマンド MMEMemory:LOAD:PPG:FLEXible <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 FLEX パターン・ファイルを PPG モジュールに読み込みます。

## 7. MMEMemory:LOAD:PPG:FLEXible:PROGram (パターンオプション)

- 機能 FLEX パターンで使用する PROG パターンの PPG モジュールへの読み込み
- コマンドとクエリの存在 Command
- コマンド MMEMemory:LOAD:PPG:FLEXible:PROGram <prog\_no>,<str>
- パラメータ <prog\_no> = FLEX パターンで使用する PROG パターンの番号  
設定範囲 1 ~ 127  
設定分解能 1  
<str> = ファイル名 (絶対パスで指定)
- 説明 FLEX パターンで使用する PROG パターンを PPG モジュールに読み込みます。

## 8. MMEemory:LOAD:ED:SETup

- 機能 ED モジュールの設定ファイルの読み込み
- コマンドとクエリの存在 Command
- コマンド MMEemory:LOAD:ED:SETup <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 ED モジュールの設定ファイルを読み込みます。

## 9. MMEemory:LOAD:ED:PROGram

- 機能 PROG パターン・ファイルの ED モジュールへの読み込み
- コマンドとクエリの存在 Command
- コマンド MMEemory:LOAD:ED:PROGram <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 PROG パターン・ファイルを ED モジュールに読み込みます。

## 10. MMEemory:LOAD:ED:STM (パターンオプション)

- 機能 STM パターン・ファイルの ED モジュールへの読み込み
- コマンドとクエリの存在 Command
- コマンド MMEemory:LOAD:ED:STM <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 STM パターン・ファイルを ED モジュールに読み込みます。

## 11. MMEemory:LOAD:ED:FLEXible (パターンオプション)

- 機能 FLEX パターン・ファイルの ED モジュールへの読み込み
- コマンドとクエリの存在 Command
- コマンド MMEemory:LOAD:ED:FLEXible <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 FLEX パターン・ファイルを ED モジュールに読み込みます。

## 5.6.4 MMEMemory サブシステム

12. MMEMemory:LOAD:ED:FLEXible:PROGram (パターンオプション)
- 機能 FLEX パターンで使用される PROG パターンの ED モジュールへの読み込み
  - コマンドとクエリの存在 Command
  - コマンド MMEMemory:LOAD:ED:FLEXible:PROGram <prog\_no>,<str>
  - パラメータ <prog\_no> = FLEX パターンで使用される PROG パターンの番号  
設定範囲 1 ~ 127  
設定分解能 1  
<str> = ファイル名 (絶対パスで指定)
  - 説明 FLEX パターンで使用される PROG パターンを ED モジュールに読み込みます。
13. MMEMemory:LOAD:JTOLerance (ジッタ耐力オプション)
- 機能 ジッタ耐力測定結果ファイル (バイナリ形式) の読み込み
  - コマンドとクエリの存在 Command
  - コマンド MMEMemory:LOAD:JTOLerance <str>
  - パラメータ <str> = ファイル名 (絶対パスで指定)
  - 説明 バイナリ形式で保存されたジッタ耐力測定結果ファイルを読み込みます (JID ファイル)。
14. MMEMemory:STORe:SETup
- 機能 D3371 の設定をファイルに保存
  - コマンドとクエリの存在 Command
  - コマンド MMEMemory:STORe:SETup <str>
  - パラメータ <str> = ファイル名 (絶対パスで指定)
  - 説明 D3371 の設定をファイルに保存します。
15. MMEMemory:STORe:SSG:SETup
- 機能 SSG モジュールの設定をファイルに保存
  - コマンドとクエリの存在 Command
  - コマンド MMEMemory:STORe:SSG:SETup <str>
  - パラメータ <str> = ファイル名 (絶対パスで指定)
  - 説明 SSG モジュールの設定をファイルに保存します。

## 16. MMEemory:STORe:PPG:SETup

- 機能 PPG モジュールの設定をファイルに保存
- コマンドとクエリの存在 Command
- コマンド MMEemory:STORe:PPG:SETup <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 PPG モジュールの設定をファイルに保存します。

## 17. MMEemory:STORe:PPG:PROGram

- 機能 PPG モジュールの PROG パターンをファイルに保存
- コマンドとクエリの存在 Command
- コマンド MMEemory:STORe:PPG:PROGram <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 PPG モジュールの PROG パターンをファイルに保存します。

## 18. MMEemory:STORe:PPG:STM (パターンオプション)

- 機能 PPG モジュールの STM パターンをファイルに保存
- コマンドとクエリの存在 Command
- コマンド MMEemory:STORe:PPG:STM <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 PPG モジュールの STM パターンをファイルに保存します。

## 19. MMEemory:STORe:PPG:FLEXible (パターンオプション)

- 機能 PPG モジュールの FLEX パターンをファイルに保存
- コマンドとクエリの存在 Command
- コマンド MMEemory:STORe:PPG:FLEXible <str>
- パラメータ <str> = ファイル名 (絶対パスで指定)
- 説明 PPG モジュールの FLEX パターンをファイルに保存します。

## 5.6.4 MMEemory サブシステム

20. MMEemory:STORE:PPG:FLEXible:PROGram (パターンオプション)
- 機能 PPG モジュールの FLEX パターンで使用される PROG パターンをファイルに保存
  - コマンドとクエリの存在 Command
  - コマンド MMEemory:STORE:PPG:FLEXible:PROGram <prog\_no>,<str>
  - パラメータ <prog\_no> = FLEX パターンで使用される PROG パターンの番号  
設定範囲 1 ~ 127  
設定分解能 1  
<str> = ファイル名 (絶対パスで指定)
  - 説明 PPG モジュールの FLEX パターンで使用される PROG パターンをファイルに保存します。
21. MMEemory:STORE:ED:SETup
- 機能 ED モジュールの設定をファイルに保存
  - コマンドとクエリの存在 Command
  - コマンド MMEemory:STORE:ED:SETup <str>
  - パラメータ <str> = ファイル名 (絶対パスで指定)
  - 説明 ED モジュールの設定をファイルに保存します。
22. MMEemory:STORE:ED:PROGram
- 機能 ED モジュールの PROG パターンをファイルに保存
  - コマンドとクエリの存在 Command
  - コマンド MMEemory:STORE:ED:PROGram <str>
  - パラメータ <str> = ファイル名 (絶対パスで指定)
  - 説明 ED モジュールの PROG パターンをファイルに保存します。
23. MMEemory:STORE:ED:STM (パターンオプション)
- 機能 ED モジュールの STM パターンをファイルに保存
  - コマンドとクエリの存在 Command
  - コマンド MMEemory:STORE:ED:STM <str>
  - パラメータ <str> = ファイル名 (絶対パスで指定)
  - 説明 ED モジュールの STM パターンをファイルに保存します。



24. MMEemory:STORe:ED:FLEXible (パターンオプション)
- 機能 ED モジュールの FLEX パターンをファイルに保存
  - コマンドとクエリの存在 Command
  - コマンド MMEemory:STORe:ED:FLEXible <str>
  - パラメータ <str> = ファイル名 (絶対パスで指定)
  - 説明 ED モジュールの FLEX パターンをファイルに保存します。
25. MMEemory:STORe:ED:FLEXible:PROGram (パターンオプション)
- 機能 ED モジュールの FLEX パターンで使用される PROG パターンをファイルに保存
  - コマンドとクエリの存在 Command
  - コマンド MMEemory:STORe:ED:FLEXible:PROGram <prog\_no>,<str>
  - パラメータ <prog\_no> = FLEX パターンで使用される PROG パターンの番号  
設定範囲 1 ~ 127  
設定分解能 1  
<str> = ファイル名 (絶対パスで指定)
  - 説明 ED モジュールの FLEX パターンで使用される PROG パターンをファイルに保存します。
26. MMEemory:STORe:ED:BMEasurement:LOG
- 機能 測定したログ情報をファイルに保存
  - コマンドとクエリの存在 Command
  - コマンド MMEemory:STORe:ED:BMEasurement:LOG <str>
  - パラメータ <str> = ファイル名 (絶対パスで指定)
  - 説明 測定したログ情報をファイルに保存します。

5.6.4 MMEemory サブシステム

27. MMEemory:STORe:ED:EPANalysis:TSErIES:TEXT (エラー位相解析オプション)

- 機能 エラー記録結果を時系列フォーマットでファイルに保存
- コマンドとクエリの存在 **Command**
- コマンド MMEemory:STORe:ED:EPANalysis:TSErIES:TEXT <cn>,<sp1>,<sp2>,<sp3>,<nr>,<str>
- パラメータ
  - <cn> = 保存開始パターンの周期番号  
 設定範囲 1 ~ パターン周期数  
 設定分解能 1
  - <sp1> = <sp2> = <sp3> = 保存開始位置

パターン・タイプ	開始位置	設定範囲	分解能
PROG, ZSUB	<sp1> アドレス	0 bit ~ (パターン長 - 1) bit	32 bits
	<sp2> ---	任意の値	---
	<sp3> ---	任意の値	---
STM	<sp1> フレーム番号	1 ~ フレーム数	1
	<sp2> 行	1 ~ 9	1
	<sp3> 列	1 byte ~ 1080 byte (STM4) 1 byte ~ 4320 byte (STM16)	4 bytes
FLEX	<sp1> インデックス	1 ~ パターン数	1
	<sp2> アドレス	0 bit ~ (パターン長 - 1) bit	32 bits
	<sp3> ---	任意の値	---

<nr> = 保存するエラー行数  
 設定範囲 1 ~ エラー行数  
 設定分解能 1

<str> = ファイル名 (絶対パスで指定)

- 説明 エラー記録結果を時系列フォーマットでファイルに保存します。テキスト形式の保存になります。<cn>,<sp1>,<sp2>,<sp3> から、<nr> で示される数のエラー行を保存します。

28. MMEemory:STORE:ED:EPANalysis:STATistics:TEXT (エラー位相解析オプション)

- 機能 エラー記録結果を統計フォーマットでファイルに保存
- コマンドとクエリの存在 Command
- コマンド MMEemory:STORE:ED:EPANalysis:STATistics:TEXT <sp1>,<sp2>,<sp3>,<range>,<str>  
パラメータ<sp1> = <sp2> = <sp3> = 保存開始位置

パターン・タイプ	開始位置	設定範囲	分解能	
PROG, ZSUB	<sp1>	アドレス	0 bit~ (パターン長 -1) bit	1 bit
	<sp2>	---	任意の値	---
	<sp3>	---	任意の値	---
STM	<sp1>	フレーム番号	1~フレーム数	1
	<sp2>	行	1~9	1
	<sp3>	列	1 byte~ 1080 byte (STM4) 1 byte ~ 4320 byte (STM16)	1 byte
FLEX	<sp1>	インデックス	1~パターン数	1
	<sp2>	アドレス	0 bit~ (パターン長 -1) bit	1 bit
	<sp3>	---	任意の値	---

<range> = 保存対象の範囲 (ビット数)

<str> = ファイル名 (絶対パスで指定)

- 説明 エラー記録結果を統計フォーマットでファイルに保存します。テキスト形式の保存になります。  
<sp1>,<sp2>,<sp3> から <range> の範囲におけるエラービットを保存します。

29. MMEemory:STORE:ED:EPANalysis:CANCel (エラー位相解析オプション)

- 機能 エラー記録結果のファイル保存キャンセル
- コマンドとクエリの存在 Command
- コマンド MMEemory:STORE:ED:EPANalysis:CANCel
- 説明 エラー記録結果のファイル保存処理をキャンセルします。

30. MMEemory:STORE:ED:EPANalysis? (エラー位相解析オプション)

- 機能 エラー記録結果のファイル保存処理状態の問い合わせ
- コマンドとクエリの存在 Query
- クエリ MMEemory:STORE:ED:EPANalysis?
- クエリ応答形式 0 | 1

## 5.6.4 MMEemory サブシステム

- 説明 エラー記録結果のファイル保存処理状態を問い合わせます。  
エラー記録結果のファイル保存中に 1 を返します。保存処理を行っていない場合は 0 を返します。
31. MMEemory:STORe:ED:EPANalysis:FAIL? (エラー位相解析オプション)
- 機能 エラー記録結果のファイル保存結果の問い合わせ
  - コマンドとクエリの存在 Query
  - クエリ MMEemory:STORe:ED:EPANalysis:FAIL?
  - クエリ応答形式 0 | 1
  - 説明 エラー記録結果のファイル保存結果を問い合わせます。  
保存に失敗した場合、データが存在しない場合、一度も保存が実行されていない場合は 1 を返します。ファイル保存に成功した場合は 0 を返します。  
エラー記録結果のファイル保存終了を確認してから実行して下さい。
32. MMEemory:STORe:JTOLerance (ジッタ耐力オプション)
- 機能 ジッタ耐力測定結果をバイナリ形式でファイルに保存
  - コマンドとクエリの存在 Command
  - コマンド MMEemory:STORe:JTOLerance <str>
  - パラメータ <str> = ファイル名 (絶対パスで指定)
  - 説明 ジッタ耐力測定結果をバイナリ形式でファイルに保存します (JID ファイル)。
33. MMEemory:STORe:JTOLerance:TEXT (ジッタ耐力オプション)
- 機能 ジッタ耐力測定結果をテキスト形式でファイルに保存
  - コマンドとクエリの存在 Command
  - コマンド MMEemory:STORe:JTOLerance:TEXT <str>
  - パラメータ <str> = ファイル名 (絶対パスで指定)
  - 説明 ジッタ耐力測定結果をテキスト形式でファイルに保存します。

## 5.6.5 SYSTEM サブシステム

### 1. SYSTEM:BEEPer:DERRor[:STATe]

- 機能 データ・エラー発生時に鳴らすブザーの ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SYSTEM:BEEPer:DERRor[:STATe] <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 データ・エラー発生時にブザーを鳴らすか (ON)、鳴らさないか (OFF) を設定します。

### 2. SYSTEM:BEEPer:ALARm[:STATe]

- 機能 アラーム発生時に鳴らすブザーの ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SYSTEM:BEEPer:ALARm[:STATe] <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 アラーム発生時にブザーを鳴らすか (ON)、鳴らさないか (OFF) を設定します。

## 5.6.5 SYSTem サブシステム

## 3. SYSTem:ERRor?

- 機能 エラーの問い合わせ
- コマンドとクエリの存在 Query
- クエリ SYSTem:ERRor?
- クエリ応答形式 <errno>, <errmsg>  
<errno> = NRI (整数型)  
<errmsg> = エラー・メッセージ (文字列)
- 説明
 

本器は最大 10 個まで発生したエラーをエラー・キューに記憶します。10 個以上のエラーが発生したときは、10 個目のエラーが以下のように置き換わります。

-350, "Queue overflow"

この 10 個日以降のエラーは保持されません。

SYSTem:ERRor? は、このエラー・キューからエラーを取り出します。

エラー・キューは、FIFO(First in First Out) 方式でエラーをストアするのでエラーは発生した順序通りに取り出されます。

エラー・キューからエラーが取り出されると、エラーはエラー・キューから消え、次に発生するエラーを記憶できるようになります。

エラーがないときは、以下のように応答します。

0, "No error"

エラー・キューは \*CLS でクリアされます。

## 4. SYSTEM:DATE

- 機能 日付の設定
- コマンドとクエリの存在 Command/Query
- コマンド SYSTEM:DATE <year>, <month>, <day>
- パラメータ <year> = 1980 ~ 2099 の数値データ  
<month> = 1 ~ 12 の数値データ  
<day> = 1 ~ 31 の数値データ
- クエリ応答形式 <year>, <month>, <day>  
<year> = <month> = <day> = NR1 (整数型)
- 説明 本器に内蔵の時計の日付を設定します。  
年の設定は西暦 (4 桁) で表して下さい (例えば 2000, 2001)。

## 5. SYSTEM:TIME

- 機能 時刻の設定
- コマンドとクエリの存在 Command/Query
- コマンド SYSTEM:TIME <hour>, <minute>, <second>
- パラメータ <hour> = 0 ~ 23 の数値データ  
<minute> = 0 ~ 59 の数値データ  
<second> = 0 ~ 59 の数値データ
- クエリ応答形式 <hour>, <minute>, <second>  
<hour> = <minute> = <second> = NR1 (整数型)
- 説明 本器に内蔵の時計の時刻を設定します。時間は 24 時間制です。

## 5.6.6 STATus サブシステム

## 5.6.6 STATus サブシステム

## 1. STATus:OPERation[:EVENT]?

- 機能                                    オペレーション・ステータス・イベント・レジスタの読み出し
- コマンドとクエリの存在    Query
- クエリ                                STATus:OPERation[:EVENT]?
- クエリ応答形式            NR1（整数型）
- 説明                                    オペレーション・ステータス・レジスタのイベント・レジスタの内容を読み出します。このレジスタは読み出されるとクリアされ、対応するステータス・バイト・レジスタのビット7もクリアされます。  
詳細は「4. ステータス・バイト」を参照して下さい。

## 2. STATus:OPERation:CONDition?

- 機能                                    オペレーション・ステータス・コンディション・レジスタの読み出し
- コマンドとクエリの存在    Query
- クエリ                                STATus:OPERation:CONDition?
- クエリ応答形式            NR1（整数型）
- 説明                                    オペレーション・ステータス・レジスタのコンディション・レジスタの内容を読み出します。このレジスタは読み出してもクリアされません。  
詳細は「4. ステータス・バイト」を参照して下さい。



## 3. STATus:OPERation:ENABle

- 機能 オペレーション・ステータス・イネーブル・レジスタの設定
- コマンドとクエリの存在 Command/Query
- コマンド STATus:OPERation:ENABle <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 オペレーション・ステータス・レジスタのイネーブル・レジスタの内容を設定します。このレジスタの 1 に設定されたビットに対応するイベント・ビットが、有効ビットとしてステータス・バイト・レジスタのビット 7 に反映されます。  
詳細は「4. ステータス・バイト」を参照して下さい。
- 例 SET (ビット 1) と END (ビット 9) をイネーブルに設定する場合、 $2^1 + 2^9 = 2 + 512 = 514$  を計算し、STAT:OPER:ENAB 514 と設定します。

4. STATus:OPERation:PTRansition  
STATus:OPERation:NTRansition

- 機能 オペレーション・ステータスのトランジション・フィルタの設定
- コマンドとクエリの存在 Command/Query
- コマンド STATus:OPERation:PTRansition <int>  
STATus:OPERation:NTRansition <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 オペレーション・ステータス・レジスタのトランジション・フィルタの内容を設定します。コンディション・ビットに変化があった場合に、イベント・ビットを 1 にセットするかを指定します。ヘッダ末尾のニーモニックにより、フィルタをかける変化方向を指定します。

末尾のニーモニック	変化方向
PTRansition	正方向
NTRansition	負方向

詳細は「4. ステータス・バイト」を参照して下さい。

## 5.6.6 STATus サブシステム

## 5. STATus:OPERation:SETTling[:EVENT]?

- 機能 セトリング・ステータス・イベント・レジスタの読み出し
- コマンドとクエリの存在 Query
- クエリ STATus:OPERation:SETTling[:EVENT]?
- クエリ応答形式 NR1 (整数型)
- 説明 セトリング・ステータス・レジスタのイベント・レジスタの内容を読み出します。このレジスタは読み出されるとクリアされず。  
詳細は「4. ステータス・バイト」を参照して下さい。

## 6. STATus:OPERation:SETTling:CONDition?

- 機能 セトリング・ステータス・コンディション・レジスタの読み出し
- コマンドとクエリの存在 Query
- クエリ STATus:OPERation:SETTling:CONDition?
- クエリ応答形式 NR1 (整数型)
- 説明 セトリング・ステータス・レジスタのコンディション・レジスタの内容を読み出します。このレジスタは読み出してもクリアされません。  
詳細は「4. ステータス・バイト」を参照して下さい。

## 7. STATus:OPERation:SETTling:ENABle

- 機能 セトリング・ステータス・イネーブル・レジスタの設定
- コマンドとクエリの存在 Command/Query
- コマンド STATus:OPERation:SETTling:ENABle <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 セトリング・ステータス・レジスタのイネーブル・レジスタの内容を設定します。このレジスタの 1 に設定されたビットに対応するイベント・ビットが、有効ビットとしてオペレーション・ステータス・コンディション・レジスタのビット 1 に反映されます。  
詳細は「4. ステータス・バイト」を参照して下さい。
- 例 PDLY (ビット 1) と EDLY (ビット 2) をイネーブルに設定する場合、 $2^1 + 2^2 = 2 + 4 = 6$  を計算し、STAT:OPER:SETT:ENAB 6 と設定します。

8. STATus:OPERation:SETTling:PTRansition  
STATus:OPERation:SETTling:NTRansition

- 機能 セトリング・ステータス・トランジション・フィルタの設定
- コマンドとクエリの存在 Command/Query
- コマンド STATus:OPERation:SETTling:PTRansition <int>  
STATus:OPERation:SETTling:NTRansition <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 セトリング・ステータス・レジスタのトランジション・フィルタの内容を設定します。コンディション・ビットに変化があった場合に、イベント・ビットを1にセットするかを指定します。ヘッダ末尾のニーモニックにより、フィルタをかける変化方向を指定します。

末尾のニーモニック	変化方向
PTRansition	正方向
NTRansition	負方向

詳細は「4. ステータス・バイト」を参照して下さい。

9. STATus:QUEStionable[:EVENT]?

- 機能 クエスチョナブル・ステータス・イベント・レジスタの読み出し
- コマンドとクエリの存在 Query
- クエリ STATus:QUEStionable[:EVENT]?
- クエリ応答形式 NR1 (整数型)
- 説明 クエスチョナブル・ステータス・レジスタのイベント・レジスタの内容を読み出します。このレジスタは読み出されるとクリアされ、対応するステータス・バイト・レジスタのビット3もクリアされます。

詳細は「4. ステータス・バイト」を参照して下さい。

## 5.6.6 STATus サブシステム

## 10. STATus:QUEStionable:CONDition?

- 機能                      クエスチョナブル・ステータス・コンディション・レジスタの読み出し
- コマンドとクエリの存在    Query
- クエリ                      STATus:QUEStionable:CONDition?
- クエリ応答形式            NR1 (整数型)
- 説明                      クエスチョナブル・ステータス・レジスタのコンディション・レジスタの内容を読み出します。このレジスタは読み出してもクリアされません。  
詳細は「4. ステータス・バイト」を参照して下さい。

## 11. STATus:QUEStionable:ENABle

- 機能                      クエスチョナブル・ステータス・イネーブル・レジスタの設定
- コマンドとクエリの存在    Command/Query
- コマンド                    STATus:QUEStionable:ENABle <int>
- パラメータ                 <int>
- クエリ応答形式            NR1 (整数型)
- 説明                      クエスチョナブル・ステータス・レジスタのイネーブル・レジスタの内容を設定します。このレジスタの1に設定されたビットに対応するイベント・ビットが、有効ビットとしてステータス・バイト・レジスタのビット3に反映されます。  
詳細は「4. ステータス・バイト」を参照して下さい。
- 例                          CLOSS (ビット9)と SLOSS (ビット10)をイネーブルに設定する場合、 $2^9 + 2^{10} = 512 + 1024 = 1536$ を計算し、STAT:QUES:ENAB 1536と設定します。

12. STATus:QUEStionable:PTRansition  
STATus:QUEStionable:NTRansition

- 機能 クエスチョナブル・ステータス・トランジション・フィルタの設定
- コマンドとクエリの存在 Command/Query
- コマンド STATus:QUEStionable:PTRansition <int>  
STATus:QUEStionable:NTRansition <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 クエスチョナブル・ステータス・レジスタのトランジション・フィルタの内容を設定します。コンディション・ビットに変化があった場合に、イベント・ビットを1にセットするかを指定します。  
ヘッダ末尾のニーモニックにより、フィルタをかける変化方向を指定します。

末尾のニーモニック	変化方向
PTRansition	正方向
NTRansition	負方向

詳細は「4. ステータス・バイト」を参照して下さい。

13. STATus:QUEStionable:CLOsS[:EVENT]?

- 機能 クロックロス・ステータス・イベント・レジスタの読み出し
- コマンドとクエリの存在 Query
- クエリ STATus:QUEStionable:CLOsS[:EVENT]?
- クエリ応答形式 NR1 (整数型)
- 説明 クロックロス・ステータス・レジスタのイベント・レジスタの内容を読み出します。このレジスタは読み出されるとクリアされます。

詳細は「4. ステータス・バイト」を参照して下さい。

## 5.6.6 STATus サブシステム

## 14. STATus:QUEStionable:CLOsS:CONDition?

- 機能 クロックロス・ステータス・コンディション・レジスタ読み出し
- コマンドとクエリの存在 Query
- クエリ STATus:QUEStionable:CLOsS:CONDition?
- クエリ応答形式 NR1 (整数型)
- 説明 クロックロス・ステータス・レジスタのコンディション・レジスタの内容を読み出します。このレジスタは読み出してもクリアされません。  
詳細は「4. ステータス・バイト」を参照して下さい。

## 15. STATus:QUEStionable:CLOsS:ENABle

- 機能 クロックロス・ステータス・イネーブル・レジスタの設定
- コマンドとクエリの存在 Command/Query
- コマンド STATus:QUEStionable:CLOsS:ENABle <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 クロックロス・ステータス・レジスタのイネーブル・レジスタの内容を設定します。このレジスタの1に設定されたビットに対応するイベント・ビットが、有効ビットとしてクエスチョナブル・ステータス・コンディション・レジスタのビット9に反映されます。  
詳細は「4. ステータス・バイト」を参照して下さい。
- 例 PPG (ビット1) と ED (ビット2) をイネーブルに設定する場合、 $2^1 + 2^2 = 2 + 4 = 6$  を計算し、STAT:QUES:CLOS:ENAB 6 と設定します。

16. STATus:QUEStionable:CLOsS:PTRansition  
STATus:QUEStionable:CLOsS:NTRansition

- 機能 クロックロス・ステータス・トランジション・フィルタの設定
- コマンドとクエリの存在 Command/Query
- コマンド STATus:QUEStionable:CLOsS:PTRansition <int>  
STATus:QUEStionable:CLOsS:NTRansition <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 クロックロス・ステータス・レジスタのトランジション・フィルタの内容を設定します。コンディション・ビットに変化があった場合に、イベント・ビットを1にセットするかを指定します。ヘッダ末尾のニーモニックにより、フィルタをかける変化方向を指定します。

末尾のニーモニック	変化方向
PTRansition	正方向
NTRansition	負方向

詳細は「4. ステータス・バイト」を参照して下さい。

17. STATus:DEVIce[:EVENT]?

- 機能 デバイス・ステータスのイベント・レジスタの読み出し
- コマンドとクエリの存在 Query
- クエリ STATus:DEVIce[:EVENT]?
- クエリ応答形式 NR1 (整数型)
- 説明 デバイス・ステータス・レジスタのイベント・レジスタの内容を読み出します。このレジスタは読み出されるとクリアされ、対応するステータス・バイト・レジスタのビット1もクリアされます。詳細は「4. ステータス・バイト」を参照して下さい。

5.6.6 STATus サブシステム

18. STATus:PRESet

- 機能                                 ステータス・レジスタの初期化
- コマンドとクエリの存在     **Command**
- コマンド                         STATus:PRESet
- 説明                                 ステータス・レジスタの初期化を行います。  
このコマンドは、オペレーション・ステータス・レジスタ、クエスチョナブル・ステータス・レジスタ、クロックロス・ステータス・レジスタ、セトリング・ステータス・レジスタのイネーブル・レジスタとトランジション・フィルタのみ初期化します。  
詳細は「4. ステータス・バイト」を参照して下さい。



## 5.7 SSG モジュール

### 5.7.1 SOURce10 サブシステム

#### 1. SOURce10:FREQuency[:CW]

- 機能 周波数の設定
- コマンドとクエリの存在 Command/Query
- コマンド SOURce10:FREQuency[:CW] <int>
- パラメータ <int>
- クエリ応答形式 NR1 (整数型)
- 説明 SSG モジュール背面パネルの CLOCK OUTPUT から発生するクロック信号の周波数を設定します。  
ジッタ変調機能が ON のとき、設定されているジッタ周波数、ジッタ振幅によっては、クロック周波数を 800 MHz 未満に設定できない場合があります。  
設定範囲 10,000 kHz ~ 3,600,000 kHz (ジッタ変調機能が OFF のとき)  
10,000 kHz ~ 3,200,000 kHz (ジッタ変調機能が ON のとき)  
設定分解能 1 kHz  
設定単位 kHz

#### 2. SOURce10:OUTPut[:STATe]

- 機能 クロック出力の ON/OFF
- コマンドとクエリの存在 Command/Query
- コマンド SOURce10:OUTPut[:STATe] <bool>
- パラメータ <bool>
- クエリ応答形式 0 | 1
- 説明 SSG モジュール背面パネルの CLOCK OUTPUT にクロック信号を出力するか (ON)、しないか (OFF) を設定します。

5.7.1 SOURce10 サブシステム

3. SOURce10:REFerence

- 機能                                      クロック基準信号の入力ソースの設定
- コマンドとクエリの存在      Command/Query
- コマンド                                SOURce10:REFerence <type>
- パラメータ                            <type> = {INTernal | EXTernal}
- クエリ応答形式                    INT | EXT
- 説明                                    SSG モジュールの信号発生に使用するクロック基準信号の入力ソースを選択します。

コマンド・パラメータ	基準周期信号
INT	内部信号
EXT	外部信号

4. SOURce10:JMODulation[:STATe] (ジッタ耐力オプション)

- 機能                                      ジッタ変調機能の ON/OFF
- コマンドとクエリの存在      Command/Query
- コマンド                                SOURce10:JMODulation[:STATe] <bool>
- パラメータ                            <bool>
- クエリ応答形式                    0 | 1
- 説明                                    ジッタ変調機能を有効にするか (ON)、無効にするか (OFF) を設定します。

5. SOURce10:JMODulation:FREQuency (ジッタ耐力オプション)

- 機能                                      ジッタ周波数の設定
- コマンドとクエリの存在      Command/Query
- コマンド                                SOURce10:JMODulation:FREQuency <int>
- パラメータ                            <int>
- クエリ応答形式                    NR1 (整数型)
- 説明                                    ジッタ周波数を設定します。

クロック周波数 (f)	ジッタ周波数設定範囲
10 MHz ≤ f < 175 MHz	10 Hz ~ 2,000,000 Hz
175 MHz ≤ f < 800 MHz	10 Hz ~ 5,000,000 Hz
800 MHz ≤ f ≤ 3.2 GHz	10 Hz ~ 20,000,000 Hz

設定分解能    10 Hz

設定単位        Hz

## 6. SOURce10:JMODulation:AMPLitude (ジッタ耐力オプション)

- 機能                      ジッタ振幅の設定
- コマンドとクエリの存在    Command/Query
- コマンド                    SOURce10:JMODulation:AMPLitude <real>
- パラメータ                 <real>
- クエリ応答形式            NR2 (固定小数点型)
- 説明                         ジッタ振幅を設定します。

[クロック周波数が 10 MHz 以上、175 MHz 未満の場合]

ジッタ周波数 (fm)	ジッタ振幅設定範囲
10 Hz	0 UIp-p ~ 200 UIp-p
20 Hz	0 UIp-p ~ 120 UIp-p
20 Hz < fm < 200 Hz	0 UIp-p ~ (2,400/fm)UIp-p
200 Hz ≤ fm < 5,000 Hz	0 UIp-p ~ 12 UIp-p
5,000 Hz ≤ fm < 200,000 Hz	0 UIp-p ~ 5 UIp-p
200,000 Hz < fm < 2,000,000 Hz	0 UIp-p ~ (1,000,000/fm)UIp-p
2,000,000 Hz	0 UIp-p ~ 0.5 UIp-p

ジッタ振幅設定範囲	ジッタ振幅設定分解能
0 UIp-p ~ 1 UIp-p	0.01 UIp-p
1 UIp-p ~ 10 UIp-p	0.1 UIp-p
10 UIp-p ~ 100 UIp-p	1 UIp-p
100 UIp-p ~ 200 UIp-p	2 UIp-p

## 5.7.1 SOURce10 サブシステム

[クロック周波数が 175 MHz 以上、800 MHz 未満の場合]

ジッタ周波数 (fm)	ジッタ振幅設定範囲
10 Hz	0 UIp-p ~ 800 UIp-p
20 Hz	0 UIp-p ~ 500 UIp-p
20 Hz < fm < 200 Hz	0 UIp-p ~ (10,000/fm)UIp-p
200 Hz ≤ fm < 5,000 Hz	0 UIp-p ~ 50 UIp-p
5,000 Hz ≤ fm < 125,000 Hz	0 UIp-p ~ 20 UIp-p
125,000 Hz ≤ fm < 5,000,000 Hz	0 UIp-p ~ (2,500,000/fm)UIp-p
5,000,000 Hz	0 UIp-p ~ 0.5 UIp-p

ジッタ振幅設定範囲	ジッタ振幅設定分解能
0 UIp-p ~ 5 UIp-p	0.01 UIp-p
5 UIp-p ~ 50 UIp-p	0.1 UIp-p
50 UIp-p ~ 500 UIp-p	1 UIp-p
500 UIp-p ~ 800 UIp-p	2 UIp-p

[クロック周波数が 800 MHz 以上、3.2 GHz 以下の場合]

ジッタ周波数 (fm)	ジッタ振幅設定範囲
10 Hz	0 UIp-p ~ 800 UIp-p
20 Hz	0 UIp-p ~ 500 UIp-p
20 Hz < fm < 200 Hz	0 UIp-p ~ (10,000/fm)UIp-p
200 Hz ≤ fm < 5,000 Hz	0 UIp-p ~ 50 UIp-p
5,000 Hz ≤ fm < 300,000 Hz	0 UIp-p ~ 20 UIp-p
300,000 Hz ≤ fm < 20,000,000 Hz	0 UIp-p ~ (6,000,000/fm)UIp-p
20,000,000 Hz	0 UIp-p ~ 0.3 UIp-p

ジッタ振幅設定範囲	ジッタ振幅設定分解能
0 UIp-p ~ 5 UIp-p	0.01 UIp-p
5 UIp-p ~ 50 UIp-p	0.1 UIp-p
50 UIp-p ~ 500 UIp-p	1 UIp-p
500 UIp-p ~ 800 UIp-p	2 UIp-p

設定単位 UIp-p

## 5.8 ジッタ耐力測定

### 5.8.1 JTOLerance サブシステム

ジッタ耐力測定の設定に関するコマンドです。ジッタ耐力パラメータ・テーブル、クロック周波数、測定モード、判定条件の設定などが含まれます。(これらのコマンドはジッタ耐力オプションが搭載されている場合のみ使用できます。)

テンプレートを編集するコマンドは存在しません。あらかじめテンプレートの編集を行ってからリモート制御を実行して下さい。

1. JTOLerance[:STATE] (ジッタ耐力オプション)
- 機能 ジッタ耐力測定の開始/停止
  - コマンドとクエリの存在 Command/Query
  - コマンド JTOLerance[:STATE] <bool>
  - パラメータ <bool>
  - クエリ応答形式 0 | 1
  - 説明 ジッタ耐力測定の開始 (ON)、停止 (OFF) を設定します。両面が Jitter Tolerance に切り換わります。

2. JTOLerance:TABLE (ジッタ耐力オプション)
- 機能 ジッタ耐力パラメータ・テーブルのリスト編集
  - コマンドとクエリの存在 Command/Query
  - コマンド JTOLerance:TABLE <index>,<freq>,<min\_ampl>,<max\_ampl>,<points>
  - クエリ JTOLerance:TABLE? <index>
  - パラメータ <index> = ジッタ耐力パラメータ・テーブル内のリストを指定するインデックス  
設定範囲 1~リスト数  
設定分解能 1  
<freq> = ジッタ周波数

クロック周波数 (f)	ジッタ周波数設定範囲
10 MHz ≤ f < 175 MHz	10 Hz ~ 2,000,000 Hz
175 MHz ≤ f < 800 MHz	10 Hz ~ 5,000,000 Hz
800 MHz ≤ f ≤ 3.2 GHz	10 Hz ~ 20,000,000 Hz

設定分解能 10 Hz

設定単位 Hz

<min\_ampl> = 最小ジッタ振幅

<max\_ampl> = 最大ジッタ振幅

## 5.8.1 JTOLerance サブシステム

[クロック周波数が 10 MHz 以上、175 MHz 未満の場合]

ジッタ周波数 (fm)	ジッタ振幅設定範囲
10 Hz	0.01 UIp-p ~ 200 UIp-p
20 Hz	0.01 UIp-p ~ 120 UIp-p
20 Hz < fm < 200 Hz	0.01 UIp-p ~ (2,400/fm) UIp-p
200 Hz ≤ fm < 5,000 Hz	0.01 UIp-p ~ 12 UIp-p
5,000 Hz ≤ fm < 200,000 Hz	0.01 UIp-p ~ 5 UIp-p
200,000 Hz ≤ fm < 2,000,000 Hz	0.01 UIp-p ~ (1,000,000/fm) UIp-p
2,000,000 Hz	0.01 UIp-p ~ 0.5 UIp-p

ジッタ振幅設定範囲	ジッタ振幅設定分解能
0.01 UIp-p ~ 1 UIp-p	0.01 UIp-p
1 UIp-p ~ 10 UIp-p	0.1 UIp-p
10 UIp-p ~ 100 UIp-p	1 UIp-p
100 UIp-p ~ 200 UIp-p	2 UIp-p

[クロック周波数が 175 MHz 以上、800 MHz 未満の場合]

ジッタ周波数 (fm)	ジッタ振幅設定範囲
10 Hz	0.01 UIp-p ~ 800 UIp-p
20 Hz	0.01 UIp-p ~ 500 UIp-p
20 Hz < fm < 200 Hz	0.01 UIp-p ~ (10,000/fm) UIp-p
200 Hz ≤ fm < 5,000 Hz	0.01 UIp-p ~ 50 UIp-p
5,000 Hz ≤ fm < 125,000 Hz	0.01 UIp-p ~ 20 UIp-p
125,000 Hz ≤ fm < 5,000,000 Hz	0.01 UIp-p ~ (2,500,000/fm) UIp-p
5,000,000 Hz	0.01 UIp-p ~ 0.5 UIp-p

ジッタ振幅設定範囲	ジッタ振幅設定分解能
0.01 UIp-p ~ 5 UIp-p	0.01 UIp-p
5 UIp-p ~ 50 UIp-p	0.1 UIp-p
50 UIp-p ~ 500 UIp-p	1 UIp-p
500 UIp-p ~ 800 UIp-p	2 UIp-p

[クロック周波数が 800 MHz 以上、3.2 GHz 以下の場合]

ジッタ周波数 (fm)	ジッタ振幅設定範囲
10 Hz	0.01 UIp-p ~ 800 UIp-p
20 Hz	0.01 UIp-p ~ 500 UIp-p
20 Hz < fm < 200 Hz	0.01 UIp-p ~ (10,000/fm) UIp-p
200 Hz ≤ fm < 5,000 Hz	0.01 UIp-p ~ 50 UIp-p
5,000 Hz ≤ fm < 300,000 Hz	0.01 UIp-p ~ 20 UIp-p
300,000 Hz ≤ fm < 20,000,000 Hz	0.01 UIp-p ~ (6,000,000/fm) UIp-p
20,000,000 Hz	0.01 UIp-p ~ 0.3 UIp-p

ジッタ振幅設定範囲	ジッタ振幅設定分解能
0.01 UIp-p ~ 5 UIp-p	0.01 UIp-p
5 UIp-p ~ 50 UIp-p	0.1 UIp-p
50 UIp-p ~ 500 UIp-p	1 UIp-p
500 UIp-p ~ 800 UIp-p	2 UIp-p

設定単位 UIp-p

<points> = ジッタ振幅の測定ポイント数

設定範囲 1 ~ 40

設定分解能 1

- クエリ応答形式 <freq>,<min\_ampl>,<max\_ampl>,<points>  
<freq> = NR1 (整数型)  
<min\_ampl> = NR2 (固定小数点型)  
<max\_ampl> = NR2 (固定小数点型)  
<points> = NR1 (整数型)
- 説明 ジッタ耐力パラメータ・テーブルに登録されているリストを編集します。<min\_ampl> ≤ <max\_ampl> の条件を満たすように設定して下さい。

3. JTOLerance:TABLE:COUNT? (ジッタ耐力オプション)

- 機能 ジッタ耐力パラメータ・テーブルのリスト数の問い合わせ
- コマンドとクエリの存在 Query
- クエリ JTOLerance:TABLE:COUNT?
- クエリ応答形式 NR1 (整数型)
- 説明 ジッタ耐力パラメータ・テーブルのリスト数を問い合わせます。

5.8.1 JTOLerance サブシステム

4. JTOLerance:TABLE:NEW (ジッタ耐力オプション)

- 機能                                   ジッタ耐力パラメータ・テーブルにリスト追加
- コマンドとクエリの存在   Command
- コマンド                           JTOLerance:TABLE:NEW <index>,<freq>,<min\_ampl>,<max\_ampl>,<points>
- パラメータ                       <index> = ジッタ耐力パラメータ・テーブル内のリストを指定するインデックス  
   設定範囲       1 ~ (リスト数 + 1)  
   設定分解能   1  
   <freq> = ジッタ周波数

クロック周波数 (f)	ジッタ周波数設定範囲
10 MHz ≤ f < 175 MHz	10 Hz ~ 2,000,000 Hz
175 MHz ≤ f < 800 MHz	10 Hz ~ 5,000,000 Hz
800 MHz ≤ f ≤ 3.2 GHz	10 Hz ~ 20,000,000 Hz

設定分解能   10 Hz

設定単位       Hz

<min\_ampl> = 最小ジッタ振幅

<max\_ampl> = 最大ジッタ振幅

[クロック周波数が 10 MHz 以上、175 MHz 未満の場合]

ジッタ周波数 (fm)	ジッタ振幅設定範囲
10 Hz	0.01 UIp-p ~ 200 UIp-p
20 Hz	0.01 UIp-p ~ 120 UIp-p
20 Hz < fm < 200 Hz	0.01 UIp-p ~ (2,400/fm) UIp-p
200 Hz ≤ fm < 5,000 Hz	0.01 UIp-p ~ 12 UIp-p
5,000 Hz ≤ fm < 200,000 Hz	0.01 UIp-p ~ 5 UIp-p
200,000 Hz ≤ fm < 2,000,000 Hz	0.01 UIp-p ~ (1,000,000/fm) UIp-p
2,000,000 Hz	0.01 UIp-p ~ 0.5 UIp-p

ジッタ振幅設定範囲	ジッタ振幅設定分解能
0.01 UIp-p ~ 1 UIp-p	0.01 UIp-p
1 UIp-p ~ 10 UIp-p	0.1 UIp-p
10 UIp-p ~ 100 UIp-p	1 UIp-p
100 UIp-p ~ 200 UIp-p	2 UIp-p



[クロック周波数が 175 MHz 以上、800 MHz 未満の場合]

ジッタ周波数 (fm)	ジッタ振幅設定範囲
10 Hz	0.01 UIp-p ~ 800 UIp-p
20 Hz	0.01 UIp-p ~ 500 UIp-p
20 Hz < fm < 200 Hz	0.01 UIp-p ~ (10,000/fm) UIp-p
200 Hz ≤ fm < 5,000 Hz	0.01 UIp-p ~ 50 UIp-p
5,000 Hz ≤ fm < 125,000 Hz	0.01 UIp-p ~ 20 UIp-p
125,000 Hz ≤ fm < 5,000,000 Hz	0.01 UIp-p ~ (2,500,000/fm) UIp-p
5,000,000 Hz	0.01 UIp-p ~ 0.5 UIp-p

ジッタ振幅設定範囲	ジッタ振幅設定分解能
0.01 UIp-p ~ 5 UIp-p	0.01 UIp-p
5 UIp-p ~ 50 UIp-p	0.1 UIp-p
50 UIp-p ~ 500 UIp-p	1 UIp-p
500 UIp-p ~ 800 UIp-p	2 UIp-p

[クロック周波数が 800 MHz 以上、3.2 GHz 以下の場合]

ジッタ周波数 (fm)	ジッタ振幅設定範囲
10 Hz	0.01 UIp-p ~ 800 UIp-p
20 Hz	0.01 UIp-p ~ 500 UIp-p
20 Hz < fm < 200 Hz	0.01 UIp-p ~ (10,000/fm) UIp-p
200 Hz ≤ fm < 5,000 Hz	0.01 UIp-p ~ 50 UIp-p
5,000 Hz ≤ fm < 300,000 Hz	0.01 UIp-p ~ 20 UIp-p
300,000 Hz ≤ fm < 20,000,000 Hz	0.01 UIp-p ~ (6,000,000/fm) UIp-p
20,000,000 Hz	0.01 UIp-p ~ 0.3 UIp-p

ジッタ振幅設定範囲	ジッタ振幅設定分解能
0.01 UIp-p ~ 5 UIp-p	0.01 UIp-p
5 UIp-p ~ 50 UIp-p	0.1 UIp-p
50 UIp-p ~ 500 UIp-p	1 UIp-p
500 UIp-p ~ 800 UIp-p	2 UIp-p

設定単位 UIp-p

<points> = ジッタ振幅の測定ポイント数

設定範囲 1 ~ 40

設定分解能 1

• 説明

ジッタ耐力パラメータ・テーブルにリストを追加します。<index>で指定された位置にリストを追加します。<min\_ampl> ≤ <max\_ampl> の条件を満たすように設定して下さい。ジッタ耐力パラメータ・テーブルのリスト数は最大 20 です。リスト数が 20 の場合、リストを追加できません。

5.8.1 JTOLerance サブシステム

5. JTOLerance:TABLE:REMOve (ジッタ耐力オプション)

- 機能                      ジッタ耐力パラメータ・テーブルのリスト削除
- コマンドとクエリの存在    Command
- コマンド                    JTOLerance:TABLE:REMOve <index>
- パラメータ                 <index> = ジッタ耐力パラメータ・テーブル内のリストを指定するインデックス  
設定範囲     1~リスト数  
設定分解能   1
- 説明                         ジッタ耐力パラメータ・テーブルのリストを削除します。リストをすべて削除することはできません。

6. JTOLerance:TEMPlate:SELEct (ジッタ耐力オプション)

- 機能                         テンプレートの設定
- コマンドとクエリの存在    Command/Query
- コマンド                    JTOLerance:TEMPlate:SELEct <str>
- パラメータ                 <str> = テンプレート・テーブルに登録されているテンプレート名
- クエリ応答形式             <str> = 設定されているテンプレート名
- 説明                         サーチ・モードで使用されるテンプレートを設定します。“NONE”を設定すると、テンプレートは使用されません。

7. JTOLerance:MODE (ジッタ耐力オプション)

- 機能                         ジッタ耐力測定モードの設定
- コマンドとクエリの存在    Command/Query
- コマンド                    JTOLerance:MODE <type>
- パラメータ                 <type> = {SEARch | SWEep}
- クエリ応答形式             SEAR | SWE
- 説明                         ジッタ耐力測定モードを設定します。

コマンド・パラメータ	測定モード
SEAR	サーチ・モード
SWE	スイープ・モード

8. JTOLerance:CFrequency (ジッタ耐力オプション)
- 機能 ジッタ耐力測定のカロック周波数の設定
  - コマンドとクエリの存在 Command/Query
  - コマンド JTOLerance:CFrequency <int>
  - パラメータ <int>
  - クエリ応答形式 NR1 (整数型)
  - 説明 ジッタ耐力測定のカロック周波数を設定します。  
既に設定されているジッタ耐力パラメータ・テーブル内のジッタ周波数、ジッタ振幅によっては、カロック周波数を 800 MHz 未満に設定できない場合があります。  
設定範囲 10,000 kHz ~ 3,200,000 kHz  
設定分解能 1 kHz  
設定単位 kHz
9. JTOLerance:STIME (ジッタ耐力オプション)
- 機能 ジッタ耐力測定のカセットリング時間の設定
  - コマンドとクエリの存在 Command/Query
  - コマンド JTOLerance:STIME <real>
  - パラメータ <real>
  - クエリ応答形式 NR2 (固定小数点型)
  - 説明 ジッタ耐力測定のカセットリング時間を設定します。  
設定範囲 0.0 s ~ 100.0 s  
設定分解能 0.1 s  
設定単位 s
10. JTOLerance:PERiod (ジッタ耐力オプション)
- 機能 ジッタ耐力測定のカ1ポイントあたりの測定時間の設定
  - コマンドとクエリの存在 Command/Query
  - コマンド JTOLerance:PERiod <int>
  - パラメータ <int>
  - クエリ応答形式 NR1 (整数型)
  - 説明 ジッタ耐力測定のカ1ポイントあたりの測定時間を設定します。  
設定範囲 1 s ~ 1000 s  
設定分解能 1 s  
設定単位 s

## 5.8.1 JTOLerance サブシステム

- |               |  |              |
|---------------|--|--------------|
| 11.           | JTOLerance:ETHReshold  | (ジッタ耐力オプション) |
| • 機能          | ジッタ耐力測定の評定基準となるエラー・カウンツの設定   |              |
| • コマンドとクエリの存在 | Command/Query  |              |
| • コマンド        | JTOLerance:ETHReshold <int>  |              |
| • パラメータ       | <int>  |              |
| • クエリ応答形式     | NR1 (整数型)  |              |
| • 説明          | ジッタ耐力測定の評定基準となるエラー・カウンツを設定します。<br>設定範囲 1 bit ~ 1,000,000 bits<br>設定分解能 1 bit |              |
| 12.           | JTOLerance:ASEarch   | (ジッタ耐力オプション) |
| • 機能          | ジッタ耐力測定のアート・サーチ機能の ON/OFF  |              |
| • コマンドとクエリの存在 | Command/Query  |              |
| • コマンド        | JTOLerance:ASEarch <bool>  |              |
| • パラメータ       | <bool>   |              |
| • クエリ応答形式     | 0   1  |              |
| • 説明          | ジッタ耐力測定開始時にアート・サーチを実行するか (ON)、実行しないか (OFF) を設定します。                           |              |
| 13.           | JTOLerance:MDATa   | (ジッタ耐力オプション) |
| • 機能          | ジッタ耐力測定データ表示の ON/OFF   |              |
| • コマンドとクエリの存在 | Command/Query  |              |
| • コマンド        | JTOLerance:MDATa <bool>  |              |
| • パラメータ       | <bool>   |              |
| • クエリ応答形式     | 0   1  |              |
| • 説明          | ジッタ耐力測定データを表示するか (ON)、しないか (OFF) を設定します。                                     |              |

14. JTOLerance:MDATa:ALL:PASS? (ジッタ耐力オプション)

- 機能 ジッタ耐力測定データの総合判定の問い合わせ
- コマンドとクエリの存在 Query
- クエリ JTOLerance:MDATa:ALL:PASS?
- クエリ応答形式 0 | 1
- 説明 ジッタ耐力測定データの総合判定を問い合わせます。すべての測定ポイントにおいて判定 OK の場合は 1 を返します。それ以外では 0 を返します。ジッタ耐力測定の終了を確認してから実行して下さい。

測定モード	判定 OK の条件
サーチ・モード	ジッタ耐力点のジッタ振幅がテンプレートより大きい
スイープ・モード	エラー・カウントがジッタ耐力測定の判定基準となるエラー・カウント未満



## 6. サンプル・プログラム

この章では、本器の GPIB ポートを使用してリモート・コントロールをする例を記述します。

**注意** 記述したサンプル・プログラムは、言語として Visual Basic 6.0（以降 VB と記述）を使用しています。また、GPIB 用コントロール・ボードとして National Instruments 社（以下 NI 社と記述）製 GPIB ボードを、コントロール・ドライバとして NI 社のドライバを使用しています。

### 6.1 測定条件の設定および読み込みのプログラム例

例 1 本器を初期状態にする

```
Call ioclr(ud) 'デバイス・クリア
Call iowrt(ud, "*RST") '初期状態にセット
```

例 2 SSG の周波数を 3.6GHz に設定

```
Call ioclr(uc) 'デバイス・クリア
Call iowrt(uc, "SOUR10:PRFQ 3600000") '周波数を 3.6GHz に設定
```

例 3 オートサーチを実行し、結果を表示する

```
Call ioclr(uc) 'デバイス・クリア
Call iowrt(ud, "SENS1:ASE ON") 'オート・サーチ開始
Call iowrt(ud, "**WAI") 'オート・サーチの終了を待つ
Call iowrt(uc, "SENS1:ASR:PA1L?") 'オート・サーチ結果のクエリ・コマンド
ReadBuff = Space(30) '30 バイトのスペース文字列を取得
Call iobrd(ud, ReadBuff) 'オート・サーチの結果を読み込む
Text1.Text = Left(ReadBuff, InStr(ReadBuff, Chr(10)) - 1)
'オート・サーチの結果を表示
```

例 4 D3371 の設定をファイルに保存

```
Call ioclr(ud) 'デバイス・クリア
Call iowrt(ud, "MMEM:STOR:SET ""D:\D3371\SETUP1""")
' SETUP1.EDS, SETUP1.PGS, SETUP1.SGS, SETUP1.TAS が作成される
```

例 5 パターンタイプを PROG パターンに設定する。

```
Call ioclr(uc) 'デバイス・クリア
Call iowrt(ud, "SOUR1:PA1L PROG") 'PPG のパターン・タイプを PROG に設定
Call iowrt(ud, "SENS1:PA1L PROG") 'ED のパターン・タイプを PROG に設定
```

## 6.1 測定条件の設定および読み込みのプログラム例

## 例 6 PROG パターン長を設定する。

```

Call ioclr(uc) * デバイス・クリア
Call iowrt(ud, "SOUR1:PATT:PROG:128") * PPG の PROG パターン長を 128 bit に設定
Call iowrt(ud, "SENS1:PATT:PROG:128") * ED の PROG パターン長を 128 bit に設定

```

## 例 7 文字列データによる PROG パターン転送 (先頭から 16 bit を 0x1234 にする)

```

Call ioclr(ud) * デバイス・クリア
Call iowrt(uc, "SOUR1:PATT:PROG:DATA 0,16,\"#1234\"") * PPG へ PROG パターン転送 (16 進フォーマット)
Call iowrt(ud, "SOUR1:PATT:PROG:DATA 0,16,\"#0001001000110100\"") * PPG へ PROG パターン転送 (2 進フォーマット)

```

## 例 8 ブロックデータによる PROG パターン転送 (先頭から 16 bit を 0x1234 にする)

```

Dir PattBuff(50) As Byte * PattBuff の領域を定義
Call ioclr(ud) * デバイス・クリア
PattBuff(0) = &H12 * PattBuff(0) に 0x12 をセット
PattBuff(1) = &H34 * PattBuff(1) に 0x34 をセット
Call ioeot(uc, 0) * EOI ラインの自動アサートを無効
Call iowrt(ud, "SOUR1:PATT:PROG:DATA 0,16,#12") * ブロックデータのヘッダまで送信
Call ioeot(ud, 1) * EOI ラインの自動アサートを有効
Call iowrt32(ud, PattBuff(0), 2) * 16bit データをバイナリで送信
Call copy_iovars * NI 社製ドライバのグローバル変数の設定

```

## 例 9 文字列データによる PROG パターンの読み込み (先頭から 16 bit)

```

Call ioclr(ud) * デバイス・クリア
Call iowrt(ud, "SOUR1:PATT:PROG:DATA? 0,16") * PPG から PROG パターンの読み込み
ReadBuff = Space(30) * 30 バイトのスペース文字列を取得
Call iorc(uc, ReadBuff) * 文字列データでパターン受信
Text1.Text = "PROG Pattern-" & Left(ReadBuff, Len(ReadBuff) - 1) * 読み込んだパターンを表示

```

## 結果例

```
PROG Pattern-"#1234"
```



## 例 10 ブロックデータによる PROG パターンの読み込み（先頭から 16 bit）

```

Dir PattBuff(50) As Byte          ' PattBuff の領域を定義
Dir DigitNum As Integer          ' ブロック・データ長の桁数
Dir BlockLen As String          ' ブロック・データ長
Dir DataByte As String          ' バイナリ・データの文字列変換用

Call ibclr(ud)                   ' デバイス・クリア
Call iowrt(ud, "SOUR1:PATT:PROG:0,16") ' PPG から PROG パターンの読み込み
Call iwr32(ud, PattBuff(0), 20)  ' ブロック・データの読み込み
Call copy_iowars                  ' NI 社製ドライバのグローバル変数の設定
DigitNum = PattBuff(1) - &H30    ' ブロック・データ長の桁数を取得

For i = 0 To DigitNum - 1        ' ブロック・データ長の桁数だけループ
    BlockLen = BlockLen & Chr(PattBuff(i + 2)) ' ブロック・データ長の組み立て
Next i

For i = DigitNum + 2 To Val(BlockLen) + DigitNum + 2 - 1
    DataByte = DataByte & Hex(PattBuff(i))    ' バイナリ・データ取得のためのループ
                                                ' バイナリ・データを文字列に変換
Next i

Text1.Text = "PROG Pattern=" & DataByte      ' Text1 にパターンを文字列表示

```

結果例  
PROG Pattern=1234

## 例 11 トラッキングの指定

```

Call ibclr(ud)                   ' デバイス・クリア
Call iowrt(ud, "SENS1:PATT:TRAC ON") ' パターン・トラッキング・オン
Call iowrt(ud, "OUTP3:TRAC ON")    ' データ出力トラッキング・オン

```

## 6.2 測定プログラム例

## 6.2 測定プログラム例

## 例1 SRQ を使用しない例

```

Dim Result As Integer          ' 結果文字列の数値変換用
Dim State As Integer          ' 状態管理用

Call iocv(0, GpioAddress, 0, 3000, 1, 0, ud) ' タイム・アウト値 300 秒
Call ioclr(ud)                 ' デバイス・クリア

Call iowrt(ud, "SENS1:BYE:MODE SING")      ' 測定タイマ・モードをシングルに設定
Call iowrt(uc, "SENS1:BYE:MODE:PER 0,0,0,10") ' 測定期間を 10 秒に設定
Call iowrt(ud, "SENS1:PAFF:PRAC ON")       ' パターン・トラッキングをオン
Call iowrt(ud, "SOUR1:PAFF:PRBS23")       ' パターンを PRBS23 に設定
Call iowrt(ud, "SOUR1:PAFF:PRBS:VRAT MRAT4") ' マーク率 1/2 に設定

Call iowrt(ud, "SENS1:ASE ON")             ' オート・サーチの開始
Call iowrt(ud, "*WAI")                     ' オート・サーチの終了を待つ
Call iowrt(ud, "SENS1:ASE:FAIL?")         ' オート・サーチの結果を取得
ReadBuff = Space(30)                       ' 30 バイトのスペース文字列を取得
Call iobrd(ud, ReadBuff)                   ' オート・サーチの結果を読み込む

Result = Val(ReadBuff)                     ' 結果を値に変換
If Result = 0 Then                          ' 結果が 0 だった場合
    Call iowrt(ud, "SENS1:BYE ON")         ' 測定開始

    State = 1                               ' State 変数の初期化
    While State = 1                          ' 測定終了をチェックするループ
        Call iowrt(ud, "SENS1:BYE?")       ' 測定状態取得のクエリ・コマンド
        ReadBuff = Space(30)               ' 30 バイトのスペース文字列を取得
        Call iorc(uc, ReadBuff)            ' 測定状態を読み込む
        State = Val(ReadBuff)              ' 測定状態を State 変数にセット
        DoEvents
    Wend

    Call iowrt(uc, "DFRC:BYE:ERRAT?")      ' エラー・レートを取得
    ReadBuff = Space(30)                   ' 30 バイトのスペース文字列を取得
    Call iobrd(ud, ReadBuff)               ' エラー・レートの読み込み

    Text1.Text = "Error Rate=" & Left(ReadBuff, InStr(ReadBuff, Chr(10)) - 1)
    ' Text1 にエラー・レートを表示
End If

```

## 結果例

```

Error Rate=0.0000E-09

```

## 例 2 SRQ を使用する例

```

Dir Result As Integer          ' 結果文字列の数値変換用
Dir SIO As Integer            ' シリアル・ボール心答バイト格納用

Call iobdev(0, GpioAddress, 0, 0300s, 1, 0, u6) ' タイム・アウト値 300 秒
Call ioclr(u6)                 ' デバイス・クリア
Call iowrt(u6, "**CLS")         ' ステータス・バイトクリア
Call iowrt(u6, "**SRE L28")    ' ステータス・バイトの OPER をセット
Call iowrt(u6, "STAT:OPER:ENAB 512") ' END bit をインペブルにセット

Call iowrt(u6, "SENS1:BYE:YLLX:MODE SING") ' 測定タイマ・モードをシングルに設定
Call iowrt(u6, "SENS1:BYE:YDLY:PER 0,0,0,10") ' 測定期間を 10 秒に設定
Call iowrt(u6, "SENS1:PACT:TRAC ON")       ' パターン・トラッキングをオン
Call iowrt(u6, "SOUR1:PACT:PRBS23")       ' パターンを PRBS23 段に設定
Call iowrt(u6, "SOUR1:PAUL:PRBS:XRAT MRAT4") ' マーク率 1/2 に設定

Call iowrt(u6, "SENS1:ASE ON")            ' オート・サーチの開始
Call iowrt(u6, "**WAT")                   ' オート・サーチの終了を待つ
Call iowrt(u6, "SENS1:ASE:FALL?")        ' オート・サーチの結果を取得
ReadBuff = Space(30)                     ' 30 バイトのスペース文字列を取得
Call iorc(u6, ReadBuff)                   ' オート・サーチの結果を読み込む

Result = Val(ReadBuff)                    ' 結果を値に変換
If Result = 0 Then                          ' 結果が 0 だった場合
    Call iowrt(u6, "SRMS":BYE ON")        ' 測定開始

    Call waitSRQ(BOARD_L, S.b)            ' 測定終了待ち
    Call iorsp(u6, S.b)                   ' シリアル・ボール実行
    Call iowrt(u6, "PFTC:BYE:ERR?")      ' エラー・レートを取得
    ReadBuff = Space(30)                  ' 30 バイトのスペース文字列を取得
    Call iord(u6, ReadBuff)               ' エラー・レートの読み込み

    Text1.Text = "Error Rate=" & Left(ReadBuff, InStr(ReadBuff, Chr(0)) - 1)
    ' Text1 にエラー・レートを表示
End If

結果例
Error Rate=0.0000E-09

```



## 付録

この章では、コマンド一覧および初期設定を示します。

### A.1 コマンド一覧

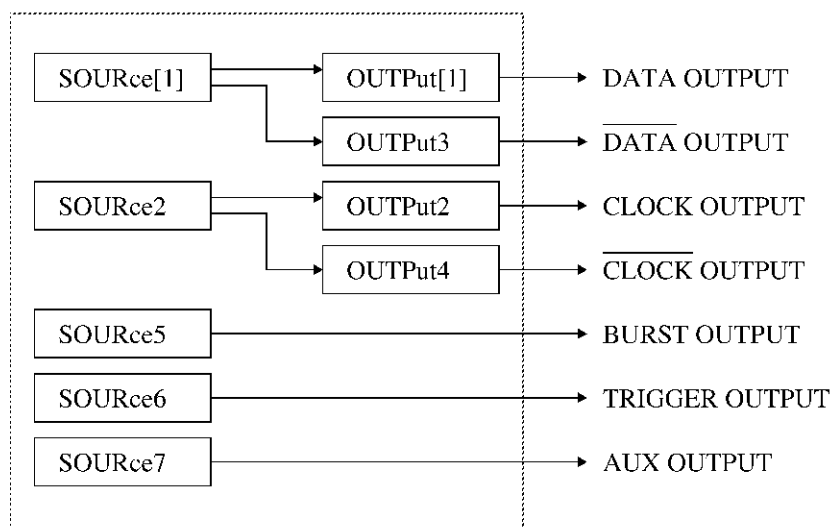
本器のコマンドをサブシステムごとに示します。  
各コマンドの詳細については、「5. コマンド・リファレンス」を参照して下さい。

#### A.1.1 サブシステムと入出力インタフェース

サブシステムと各測定モジュールの入出力インタフェースの関係を示します。

##### A.1.1.1 PPG モジュールの入出力とサブシステム

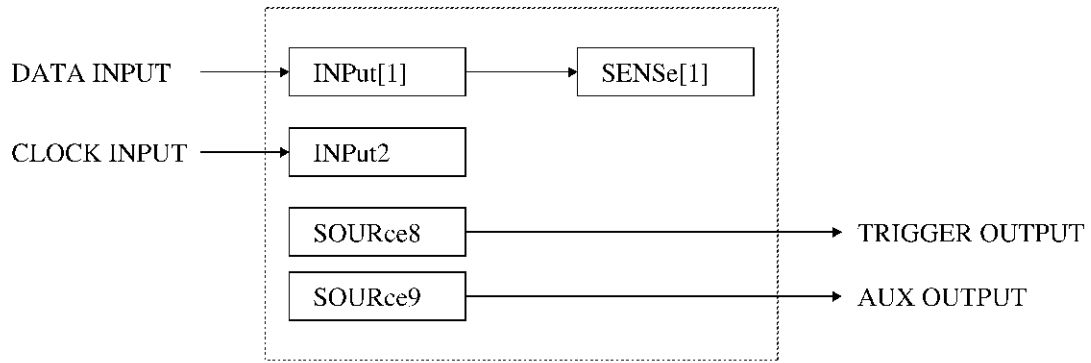
本器の PPG モジュールの入出力インタフェースとサブシステムの対応を示します。



A.1.1 サブシステムと入出力インターフェース

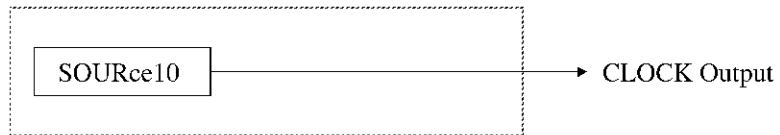
**A.1.1.2 ED モジュールの入出力とサブシステム**

本器の ED モジュールの入出力インターフェースとサブシステムの対応を示します。



**A.1.1.3 SSG モジュールの入出力とサブシステム**

本器の SSG モジュールの入出力インターフェースとサブシステムの対応を示します。



**A.1.2 共通コマンド**

KEYWORD	PARAMETER FORM	COMMENTS
*CLS	---	no query
*ESE	<int>	
*ESE?	<NR1>	
*ESR?	<NR1>	query only
*IDN?	<manufacturer>,<model>, <serial number>,<firmware level>	query only
*OPC	---	
*OPC?	1	
*RST	---	no query
*SRE	<int>	
*SRE?	<NR1>	
*STB?	<NR1>	query only
*TST?	0	query only
*WAI	---	
*OPT?	<opt>,<opt>,...	query only

## A.1.3 PPG モジュール

## A.1.3 PPG モジュール

## A.1.3.1 SOURce[1] サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
SOURce[1]		
:PATTern		
[:SElect]	PRBS<n> ZSUBstitut<n> PROGram STM<n> FLEXible	
[:SElect]?	PRBS<n> ZSUB<n> PROG STM<n> FLEX	
:PRBS		
:MRATio	MRATio<n> MINVerted4	
:MRATio?	MRAT<n> MINV4	
:ZSUBstitut		
[:ZLEngth]	<int>	
[:ZLEngth]?	<NR1>	
:PROGram		
[:LEngth]	<int>	
[:LEngth]?	<NR1>	
:DATA	<start_bit>,<length>,<str>	
:DATA?	<start_bit>,<length>/<str>	
:BDATa	<start_bit>,<length>,<block>	
:BDATa?	<start_bit>,<length>/<block>	
:STM		
:NFRame	<int>	
:NFRame?	<NR1>	
:DATA	<start_frm>,<start_row>,<start_col>,<length>,<str>	
:DATA?	<start_frm>,<start_row>,<start_col>,<length>/<str>	
:BDATa	<start_frm>,<start_row>,<start_col>,<length>,<block>	
:BDATa?	<start_frm>,<start_row>,<start_col>,<length>/<block>	
:PAYLoad		
:PRBS	<bool>	
:PRBS?	0 1	
:SElect	PRBS<n>	
:SElect?	PRBS<n>	
:MRATio	MRATio<n> MINVerted4	
:MRATio?	MRAT<n> MINV4	
:SCRamble	<bool>	
:SCRamble?	0 1	
:B1	<bool>	
:B1?	0 1	
:FLEXible		
:STable	<index>,<prog_no> PRBS,<length>,HIGH LOW	



:STABLE?	<index>/<prog_no> PRBS,<length>,HIGH LOW	
:COUNT?	<NR1>	query only
:NEW	<index>,<prog_no> PRBS,<length>,HIGH LOW	no query
:REMOve	<index>	no query
:DEFault		no query
:PROGram		
[:LENGth]	<prog_no>,<length>	
[:LENGth]?	<prog_no>/<NR1>	
:DATA	<prog_no>,<start_bit>,<length>,<str>	
:DATA?	<prog_no>,<start_bit>,<length>/<str>	
:BDATA	<prog_no>,<start_bit>,<length>,<block>	
:BDATA?	<prog_no>,<start_bit>,<length>/<block>	
:PRBS		
:SElect	PRBS<n>	
:SElect?	PRBS<n>	
:MRATio	MRATio<n> MINVerted4	
:MRATio?	MRAT<n> MINV4	
:POLarity	NORMal INVerted	
:POLarity?	NORM INV	
:EADDition	<bool>	
:EADDition?	0 1	
:SINGle		no query
:MODE	SINGle REPeat EXTernal	
:MODE?	SING REP EXT	
:RATE	<real>	
:RATE?	<NR3>	
:ROUTE	<int>	
:ROUTE?	<NR1>	

### A.1.3.2 OUTPUT[1] サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
OUTPut[1]		
[:STATe]	<bool>	
[:STATe]?	0 1	
:VOLTagE		
[:LEVel]		
[:IMMediate]		
[:AMPLitude]	<real>	
[:AMPLitude]?	<NR2>	
:HIGH	<real>	

A.1.3 PPG モジュール

	:HIGH?	<NR2>
	:MIDDLE	<real>
	:MIDDLE?	<NR2>
	:LOW	<real>
	:LOW?	<NR2>
:CROSSs		<int>
:CROSSs?		<NR1>
:TERMination		P1R3V GND M2V CML
:TERMination?		P1R3V GND M2V CML
	:VOLTage	<real>
	:VOLTage?	<NR2>

**A.1.3.3**    **OUTPut3** サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
OUTPut3		
[:STATe]	<bool>	
[:STATe]?	0 1 1	
:VOLTage		
[:LEVel]		
[:IMMediate]		
[:AMPLitude]	<real>	
[:AMPLitude]?	<NR2>	
:HIGH	<real>	
:HIGH?	<NR2>	
:MIDDLE	<real>	
:MIDDLE?	<NR2>	
:LOW	<real>	
:LOW?	<NR2>	
:CROSSs	<int>	
:CROSSs?	<NR1>	
:TERMination	P1R3V GND M2V CML	
:TERMination?	P1R3V GND M2V CML	
:VOLTage	<real>	
:VOLTage?	<NR2>	
:TRACk	<bool>	
:TRACk?	0 1 1	

**A.1.3.4 SOURce2 サブシステム**

KEYWORD	PARAMETER FORM	COMMENTS
SOURce2		
:DElay	<int>	
:DElay?	<NR1>	

**A.1.3.5 OUTPut2 サブシステム**

KEYWORD	PARAMETER FORM	COMMENTS
OUTPut2		
[:STATe]	<bool>	
[:STATe]?	0 1	
:VOLTagE		
[:LEVel]		
[:IMMediate]		
[:AMPLitude]	<real>	
[:AMPLitude]?	<NR2>	
:HIGH	<real>	
:HIGH?	<NR2>	
:MIDDLE	<real>	
:MIDDLE?	<NR2>	
:LOW	<real>	
:LOW?	<NR2>	
:TERMination	P1R3V GND M2V CML	
:TERMination?	P1R3V GND M2V CML	
:VOLTagE	<real>	
:VOLTagE?	<NR2>	

**A.1.3.6 OUTPut4 サブシステム**

KEYWORD	PARAMETER FORM	COMMENTS
OUTPut4		
[:STATe]	<bool>	
[:STATe]?	0 1	
:VOLTagE		
[:LEVel]		
[:IMMediate]		
[:AMPLitude]	<real>	
[:AMPLitude]?	<NR2>	
:HIGH	<real>	

A.1.3 PPG モジュール

:HIGH?	<NR2>
:MIDDLE	<real>
:MIDDLE?	<NR2>
:LOW	<real>
:LOW?	<NR2>
:TERMination	P1R3V GND M2V CML
:TERMination?	P1R3V GND M2V CML
:VOLTage	<real>
:VOLTage?	<NR2>
:TRACk	<bool>
:TRACk?	0 1

**A.1.3.7 SOURce5** サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
SOURce5		
:BOUtput		
[:STATe]	<bool>	
[:STATe]?	0 1	
:SOURce	INTernal EXTernal	
:SOURce?	INT EXT	
:CYCLe	<int>	
:CYCLe?	<NR1>	
:OTIME	<int>	
:OTIME?	<NR1>	

**A.1.3.8 SOURce6** サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
SOURce6		
:TRIGger		
[:MODE]	CLOCk8 CLOCk32 PATTErn FLEXible	
[:MODE]?	CLOC8 CLOC32 PATT FLEX	
:PRBS		
:POSition	<int>	
:POSition?	<NR1>	
:ZSUBstitut		
:POSition	<int>	
:POSition?	<NR1>	
:PROGram		
:POSition	<int>	

:POSition?	<NR1>
:STM	
:FNuMber	<int>
:FNuMber?	<NR1>
:ROW	<int>
:ROW?	<NR1>
:COLuMn	<int>
:COLuMn?	<NR1>
:MODE	PATTeRn FRAMe
:MODE?	PATT FRAM
:FLEXible	
:INDeX	<int>
:INDeX?	<NR1>
:ADDeSS	<int>
:ADDeSS?	<NR1>

#### A.1.3.9 SOURce7 サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
SOURce7		

## A.1.4 ED モジュール

## A.1.4.1 SENSE[1] サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
SENSE[1]		
:PATTERN		
[:SElect]	PRBS<n> ZSUBstitut<n> PROGrama STM<n> FLEXible	
[:SElect]?	PRBS<n> ZSUB<n> PROG STM<n> FLEX	
:PRBS		
:MRATio	MRATio<n> MINVerted4	
:MRATio?	MRAT<n> MINV4	
:ZSUBstitut		
[:ZLEngth]	<int>	
[:ZLEngth]?	<NR1>	
:PROGrama		
[:LEngth]	<int>	
[:LEngth]?	<NR1>	
:DATA	<start_bit>,<length>,<str>	
:DATA?	<start_bit>,<length>/<str>	
:BDATA	<start_bit>,<length>,<block>	
:BDATA?	<start_bit>,<length>/<block>	
:STM		
:NFRame	<int>	
:NFRame?	<NR1>	
:DATA	<start_frm>,<start_row>,<start_col>,<length>,<str>	
:DATA?	<start_frm>,<start_row>,<start_col>,<length>/<str>	
:BDATA	<start_frm>,<start_row>,<start_col>,<length>,<block>	
:BDATA?	<start_frm>,<start_row>,<start_col>,<length>/<block>	
:PAYLoad		
:PRBS	<bool>	
:PRBS?	011	
:SElect	PRBS<n>	
:SElect?	PRBS<n>	
:MRATio	MRATio<n> MINVerted4	
:MRATio?	MRAT<n> MINV4	
:SCRamble	<bool>	
:SCRamble?	011	
:B1	<bool>	
:B1?	011	
:FLEXible		

:STABLE	<index>, <prog_no> PRBS, <length>, HIGH LOW	
:STABLE?	<index>/<prog_no> PRBS, <length>, HIGH LOW	
:COUNT?	<NR1>	query only
:NEW	<index>, <prog_no> PRBS, <length>, HIGH LOW	
		no query
:REMOVe	<index>	no query
:DEFault		no query
:PROGrama		
[:LENGTh]	<prog_no>, <length>	
[:LENGTh]?	<prog_no>/<NR1>	
:DATA	<prog_no>, <start_bit>, <length>, <str>	
:DATA?	<prog_no>, <start_bit>, <length>/<str>	
:BDATA	<prog_no>, <start_bit>, <length>, <block>	
:BDATA?	<prog_no>, <start_bit>, <length>/<block>	
:PRBS		
:SElect	PRBS<n>	
:SElect?	PRBS<n>	
:MRATio	MRATio<n> MINVerted4	
:MRATio?	MRAT<n> MINV4	
:POLarity	NORMAl INVerted	
:POLarity?	NORM INV	
:TRACk	<bool>	
:TRACk?	0 1	
:SYNChronisat		
[:AUTO]	ONCE <bool>	
[:AUTO]?	0 1	
:PATtern		
:PROGrama		
:ADDReSS	<int>	
:ADDReSS?	<NR1>	
:FLEXible		
:INDeX	<int>	
:INDeX?	<NR1>	
:ADDReSS	<int>	
:ADDReSS?	<NR1>	
:LENGTh	<int>	
:LENGTh?	<NR1>	
:THReShold		
:AUTO	<bool>	
:AUTO?	0 1	
:PRBS		
:GAIN	<real>	
:GAIN?	<NR3>	
:LOSS	<real>	

A.1.4 ED モジュール

:LOSS?	<NR3>	
:MEMory		
:GAIN	<real>	
:GAIN?	<NR3>	
:LOSS	<real>	
:LOSS?	<NR3>	
:MASK		
:ROUte<n>	<bool>	
:ROUte<n>?	011	
:ASearch		
[:STATe]	<bool>	
[:STAEe]?	011	
:FAIL?	011	query only
:CLOCK	<bool>	
:CLOCK?	011	
:TVOLtage	<bool>	
:TVOLtage?	011	
:PATtern	<bool>	
:PATtern?	011	
:MONitor		
:HISTory		
:CLEar		no query
:LOG		
:BMEasurement	<bool>	
:BMEasurement?	011	
:HISTory		
:SCLoss	<bool>	
:SCLoss?	011	
:MDATa		
:IERate	<bool>	
:IERate?	011	
:IECount	<bool>	
:IECount?	011	
:EINTervals	<bool>	
:EINTervals?	011	
:EFINTervals	<bool>	
:EFINTervals?	011	
:FREQuency	<bool>	
:FREQuency?	011	
:BMEasurement		
[:STATe]	<bool>	
[:STATe]?	011	
:MTIME		
:MODE	SINGLE REPeat UNTimed	



```

:MODE?          SING|REP|UNT
:INterval       <real>
:INterval?     <NR2>
:PERiod        <day>,<hour>,<minute>,<second>
:PERiod?       <day>,<hour>,<minute>,<second>
:DMODE         OIT|OPT|SOT
:DMODE?        OIT|OPT|SOT
:BURSt         <bool>
:BURSt?        0|1
:EVAluate
  :CLOsS       <bool>
  :CLOsS?     0|1
  :SLOsS       <bool>
  :SLOsS?     0|1
:EPERformance
  :THReshold   <us_ses>,<dm>
  :THReshold? <us_ses>,<dm>
:CDATa         <bool>
:CDATa?        0|1
:SPECific
:PROgram
  :START
    :ADDRESS   <int>
    :ADDRESS?  <NR1>
  :END
    :ADDRESS   <int>
    :ADDRESS?  <NR1>
:ZSUBstitut
  :START
    :ADDRESS   <int>
    :ADDRESS?  <NR1>
  :END
    :ADDRESS   <int>
    :ADDRESS?  <NR1>
:STM
  :START
    :FNUMber   <int>
    :FNUMber?  <NR1>
    :ROW       <int>
    :ROW?      <NR1>
    :COLumn    <int>
    :COLumn?   <NR1>
  :END
    :FNUMber   <int>

```

A.1.4 ED モジュール

```

        :FNUMber?      <NR1>
        :ROW          <int>
        :ROW?         <NR1>
        :COLumn       <int>
        :COLumn?     <NR1>
        :MODE         PATTern|FRAMe
        :MODE?       <NR1>
:FLExible
    :START
        :INDEX       <int>
        :INDEX?     <NR1>
        :ADDRESS     <int>
        :ADDRESS?   <NR1>
    :END
        :INDEX       <int>
        :INDEX?     <NR1>
        :ADDRESS     <int>
        :ADDRESS?   <NR1>
:EPANalysis
    :ERECord
        [:STATe]     <bool>
        [:STATe]?   0|1
        :AREa       SPECific|TOTal
        :AREa?     SPEC|TOT

```

**A.1.4.2 INPut[1] サブシステム**

KEYWORD	PARAMETER FORM	COMMENTS
INPut[1]		
:POLarity	NORMal INVerted	
:POLarity?	NORM INV	
:TVOLTage	<real>	
:TVOLTage?	<NR2>	
:TERMination	P3V P1R3V GND M2V CML	
:TERMination?	P3V P1R3V GND M2V CML	
:VARIable	<bool>	
:VARIable?	0 1	
:VOLTage	<real>	
:VOLTage?	<NR2>	

**A.1.4.3 INPut2 サブシステム**

KEYWORD	PARAMETER FORM	COMMENTS
INPut2		
:POLarity	NORMal INVerted	
:POLarity?	NORM INV	
:DELay	<int>	
:DELay?	<NR1>	
:TERMination	P3V P1R3V GND M2V CML	
:TERMination?	P3V P1R3V GND M2V CML	
:VARiable	<bool>	
:VARiable?	0 1	
:VOLTagE	<real>	
:VOLTagE?	<NR2>	

**A.1.4.4 SOURce8 サブシステム**

KEYWORD	PARAMETER FORM	COMMENTS
SOURce8		
:TRIGger		
[:MODE]	CLOCK16 PATTERN FLEXible	
[:MODE]?	CLOC16 PATT FLEX	
STM		
:MODE	PATTERN FRAME	
:MODE?	PATT FRAM	

**A.1.4.5 SOURce9 サブシステム**

KEYWORD	PARAMETER FORM	COMMENTS
SOURce9		
:AUX		
[:MODE]	DTYPE SYNChronisat	
[:MODE]?	DTYPE SYNC	

## A.1.5 結果、システム・コマンド

## A.1.5 結果、システム・コマンド

## A.1.5.1 FETCH/CFETCH サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
FETCH CFETCH		
[:SENSE[1]]		
:BMEasurement		
:BCount		
[:TOTal]?	<NR1> <NR3>	query only
:OMITting?	<NR1> <NR3>	query only
:INSerting?	<NR1> <NR3>	query only
:OVERhead?	<NR1> <NR3>	query only
:PAYLoad?	<NR1> <NR3>	query only
:SPECific?	<NR1> <NR3>	query only
:OTHer?	<NR1> <NR3>	query only
:ERATe		
[:TOTal]?	<NR3>	query only
:OMITting?	<NR3>	query only
:INSerting?	<NR3>	query only
:OVERhead?	<NR3>	query only
:PAYLoad?	<NR3>	query only
:SPECific?	<NR3>	query only
:OTHer?	<NR3>	query only
:ECCount		
[:TOTal]?	<NR1> <NR3>	query only
:OMITting?	<NR1> <NR3>	query only
:INSerting?	<NR1> <NR3>	query only
:OVERhead?	<NR1> <NR3>	query only
:PAYLoad?	<NR1> <NR3>	query only
:SPECific?	<NR1> <NR3>	query only
:OTHer?	<NR1> <NR3>	query only
:IERate		
[:TOTal]?	<NR3>	query, CFETCH only
:OMITting?	<NR3>	query, CFETCH only
:INSerting?	<NR3>	query, CFETCH only
:OVERhead?	<NR3>	query, CFETCH only
:PAYLoad?	<NR3>	query, CFETCH only
:SPECific?	<NR3>	query, CFETCH only
:OTHer?	<NR3>	query, CFETCH only
:IECount		
[:TOTal]?	<NR1> <NR3>	query, CFETCH only

:OMITting?	<NR1> <NR3>	query, CFETch only
:INSerting?	<NR1> <NR3>	query, CFETch only
:OVERhead?	<NR1> <NR3>	query, CFETch only
:PAYLoad?	<NR1> <NR3>	query, CFETch only
:SPECific?	<NR1> <NR3>	query, CFETch only
:OTHER?	<NR1> <NR3>	query, CFETch only
:EINTervals		
[:TOTal]?	<NR1> <NR2>	query only
:OMITting?	<NR1> <NR2>	query only
:INSerting?	<NR1> <NR2>	query only
:OVERhead?	<NR1> <NR2>	query only
:PAYLoad?	<NR1> <NR2>	query only
:SPECific?	<NR1> <NR2>	query only
:OTHER?	<NR1> <NR2>	query only
:EFINTervals		
[:TOTal]?	<NR1> <NR2>	query only
:OMITting?	<NR1> <NR2>	query only
:INSerting?	<NR1> <NR2>	query only
:OVERhead?	<NR1> <NR2>	query only
:PAYLoad?	<NR1> <NR2>	query only
:SPECific?	<NR1> <NR2>	query only
:OTHER?	<NR1> <NR2>	query only
:FREQuency?	<NR1>	query only
:CLINTervals?	<NR1>	query only
:SLINTervals?	<NR1>	query only
:TEINTervals		
:EM<n>		
[:TOTal]?	<NR1> <NR2>	query only
:LEM8		
[:TOTal]?	<NR1> <NR2>	query only
:TEFintervals		
:EM<n>		
[:TOTal]?	<NR1> <NR2>	query only
:LEM8		
[:TOTal]?	<NR1> <NR2>	query only
:EPERformance		
:ESEConds		
[:TOTal]?	<NR1> <NR2>	query only
:EFSeconds		
[:TOTal]?	<NR1> <NR2>	query only
:SESeconds		
[:TOTal]?	<NR1> <NR2>	query only
:USEConds		
[:TOTal]?	<NR1> <NR2>	query only

## A.1.5 結果、システム・コマンド

:DMINutes			
[:TOTal]?	<NR1> <NR2>		query only
:BlError			
:BCOunt?	<NR1> <NR3>		query only
:ERATe?	<NR3>		query only
:ECOunt?	<NR1> <NR3>		query only
:MTIME			
:ETIME?	<day>,<hour>,<minute>,<second>		query, FETCh only
:TIMed?	<day>,<hour>,<minute>,<second>		query, FETCh only
:STIME?	<year>,<month>,<day>,<hour>,<minute>,<second>		query, FETCh only
:RTIME?	<year>,<month>,<day>,<hour>,<minute>,<second>		query, FETCh only
:MONitor			
:RTIME			
:DERRor?	011		query, FETCh only
:CLOSs?	011		query, FETCh only
:SLOSs?	011		query, FETCh only
:HISTory			
:DERRor?	011		query, FETCh only
:CLOSs?	011		query, FETCh only
:SLOSs?	011		query, FETCh only
:EPANalysis			
:MDATa			
:TSERIes			
:ERRor			
:COUNT?	<NR1>		query, FETCh only
:CYCLe			
:COUNT?	<NR1>		query, FETCh only
:JTOLerance:			
:MDATa			
:JFRequency			
:COUNT?	<NR1>		query, FETCh only
:JAMPLitude			
:COUNT?	<NR1>		query, FETCh only
:MDATa?	<index_freq>,<index_ampl>/<str>		query, FETCh only
:MTIME			
:ETIME?	<day>,<hour>,<minute>,<second>		query, FETCh only
:STIME?	<year>,<month>,<day>,<hour>,<minute>,<second>		query, FETCh only

**A.1.5.2 DISPlay** サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
DISPlay		
:WINDow		
:BCount		
:DFormat	INTEgral EXPOntial	
:DFormat?	INT EXP	
:ECount		
:DFormat	INTEgral EXPOntial	
:DFormat?	INT EXP	
:IECount		
:DFormat	INTEgral EXPOntial	
:DFormat?	INT EXP	
:EINTervals		
:DFormat	INTEgral PERCent	
:DFormat?	INT PERC	
:EFINTervals		
:DFormat	INTEgral PERCent	
:DFormat?	INT PERC	
:TEINTervals		
:DFormat	INTEgral PERCent	
:DFormat?	INT PERC	
:TEFintervals		
:DFormat	INTEgral PERCent	
:DFormat?	INT PERC	
:EPERformance		
:DFormat	INTEgral PERCent	
:DFormat?	INT PERC	

**A.1.5.3 HCOpy** サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
HCOpy		
[:ITEM]		
:ED		
:BMEasurement		no query
:JTOLerance		no query

## A.1.5.4 MMEMemory サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
MMEMemory		
:LOAD		
:SETup	<str>	no query
:SSG		
:SETup	<str>	no query
:PPG		
:SETup	<str>	no query
:PROGram	<str>	no query
:STM	<str>	no query
:FLEXible	<str>	no query
:PROGram	<prog_no>,<str>	no query
:ED		
:SETup	<str>	no query
:PROGram	<str>	no query
:STM	<str>	no query
:FLEXible	<str>	no query
:PROGram	<prog_no>,<str>	no query
:JTOLerance	<str>	no query
:STORE		
:SETup	<str>	no query
:SSG		
:SETup	<str>	no query
:PPG		
:SETup	<str>	no query
:PROGram	<str>	no query
:STM	<str>	no query
:FLEXible	<str>	no query
:PROGram	<prog_no>,<str>	no query
:ED		
:SETup	<str>	no query
:PROGram	<str>	no query
:STM	<str>	no query
:FLEXible	<str>	no query
:PROGram	<prog_no>,<str>	no query
:BMEasurement		
:LOG	<str>	no query
:EPANalysis		
:EPANalysis?	011	query only
:TSErIES		
:TEXT	<cn>,<sp1>,<sp2>,<sp3>,<nr>,<str>	no query



:STATistics			
:TEXT	<sp1>,<sp2>,<sp3>,<range>,<str>		no query
:CANCel			no query
:FAIL?	0 1		query only
:JTOLerance	<str>		no query
:TEXT	<str>		no query

### A.1.5.5 SYSTEM サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
SYSTEM		
:BEEPer		
:DERRor		
[:STATe]	<bool>	
[:STATe]?	0 1	
:ALARm		
[:STATe]	<bool>	
[:STATe]?	<0 1>	
:ERRor?	<NR1>,<str>	query only
:DATE	<year>,<month>,<day>	
:DATE?	<year>,<month>,<day>	
:TIME	<hour>,<minute>,<second>	
:TIME?	<hour>,<minute>,<second>	

## A.1.5.6 STATus サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
STATus		
:OPERation		
[:EVENT]?	<NR1>	query only
:CONDition?	<NR1>	query only
:ENABle	<int>	
:ENABle?	<NR1>	
:PTRansition	<int>	
:PTRansition?	<NR1>	
:NTRansition	<int>	
:NTRansition?	<NR1>	
:SETTling		
[:EVENT]?	<NR1>	query only
:CONDition?	<NR1>	query only
:ENABle	<int>	
:ENABle?	<NR1>	
:PTRansition	<int>	
:PTRansition?	<NR1>	
:NTRansition	<int>	
:NTRansition?	<NR1>	
:QUEStionable		
[:EVENT]?	<NR1>	query only
:CONDition?	<NR1>	query only
:ENABle	<int>	
:ENABle?	<NR1>	
:PTRansition	<int>	
:PTRansition?	<NR1>	
:NTRansition	<int>	
:NTRansition?	<NR1>	
:CLOSSs		
[:EVENT]?	<NR1>	query only
:CONDition?	<NR1>	query only
:ENABle	<int>	
:ENABle?	<NR1>	
:PTRansition	<int>	
:PTRansition?	<NR1>	
:NTRansition	<int>	
:NTRansition?	<NR1>	
:DEVIce		
[:EVENT]?	<NR1>	query only
:PRESet	---	no query

## A.1.6 SSG モジュール

### A.1.6.1 SOURce10 サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
SOURce10		
:FREQuency		
[:CW]	<int>	
[:CW]?	<NR1>	
:OUTPut		
[:STATe]	<bool>	
[:STATe]?	0 1	
:REFeRence	INTernal EXTernal	
:REFeRence?	INT EXT	
:JMODulation		
[:STATe]	<bool>	
[:STATe]?	0 1	
:FREQuency	<int>	
:FREQuency?	<NR1>	
:AMPLitude	<real>	
:AMPLitude?	<NR2>	

## A.1.7 ジッタ耐力測定

## A.1.7 ジッタ耐力測定

## A.1.7.1 JTOLerance サブシステム

KEYWORD	PARAMETER FORM	COMMENTS
JTOLerance		
[:STATe]	<bool>	
[:STATe]?	011	
:TABLE	<index>,<freq>,<min_ampl>,<max_ampl>,<points>	
:TABLE?	<index>/<freq>,<min_ampl>,<max_ampl>,<points>	
:COUNT?	<NR1>	query only
:NEW	<index>,<freq>,<min_ampl>,<max_ampl>,<points>	no query
:REMOve	<index>	no query
:TEMPlate		
:SElect	<str>	
:SElect?	<str>	
:MODE	SEARCh SWEep	
:MODE?	SEAR SWE	
:CFRequency	<int>	
:CFRequency?	<NR1>	
:STIME	<real>	
:STIME?	<NR2>	
:PERiod	<int>	
:PERiod?	<NR1>	
:ETHReshold	<int>	
:ETHReshold?	<NR1>	
:ASEarch	<bool>	
:ASEarch?	011	
:MDATa	<bool>	
:MDATa?	011	
:ALL		
:PASS?	011	query only

## A.2 設定メニューに対応するコマンド一覧

本器の設定メニューに対応する GPIB コマンドを示します。

### A.2.1 メニュー

メニュー	サブ・メニュー	コマンド
File	Open	MMEemory:LOAD:SETup <str> MMEemory:LOAD:SSG:SETup <str> MMEemory:LOAD:PPG:SETup <str> MMEemory:LOAD:PPG:PROGram <str> MMEemory:LOAD:PPG:STM <str> MMEemory:LOAD:PPG:FLEXible <str> MMEemory:LOAD:PPG:FLEXible:PROGram <prog_no>,<str> MMEemory:LOAD:ED:SETup <str> MMEemory:LOAD:ED:PROGram <str> MMEemory:LOAD:ED:STM <str> MMEemory:LOAD:ED:FLEXible <str> MMEemory:LOAD:ED:FLEXible:PROGram <prog_no>,<str> MMEemory:LOAD:JTOLerance <str>
	Save As	MMEemory:STORe:SETup <str> MMEemory:STORe:SSG:SETup <str> MMEemory:STORe:PPG:SETup <str> MMEemory:STORe:PPG:PROGram <str> MMEemory:STORe:PPG:STM <str> MMEemory:STORe:PPG:FLEXible <str> MMEemory:STORe:PPG:FLEXible:PROGram <prog_no>,<str> MMEemory:STORe:ED:SETup <str> MMEemory:STORe:ED:PROGram <str> MMEemory:STORe:ED:STM <str> MMEemory:STORe:ED:FLEXible <str> MMEemory:STORe:ED:FLEXible:PROGram <prog_no>,<str> MMEemory:STORe:ED:MEasurement:LOG <str> MMEemory:STORe:ED:EPANalysis:TSERies:TEXT <cn>,<sp1>,<sp2>,<sp3>,<rr>,<str> MMEemory:STORe:ED:EPANalysis:STATistics:TEXT <sp1>,<sp2>,<sp3>,<range>,<str> MMEemory:STORe:JTOLerance <str> MMEemory:STORe:JTOLerance:TEXT <str>
	Print	ECPY[:ITEM]:ED:MEasurement ECPY[:ITEM]:JTOLerance

A.2.1 メニュー

メニュー	サブ・メニュー	コマンド
Measurement	Start	SENSe[1]:3MEasurement[:STATe] ON JTOlerance[:STATe] ON
	Stop	SENSe[1]:3MEasurement[:STATe] OFF JTOlerance[:STATe] OFF
	Auto Search...	SENSe[1]:ASEarch[:STATe] ON
	Output Clock & Data	OUTPut[1][:STATe] <bool> OUTPut2[:STATe] <bool> OUTPut3[:STATe] <bool> OUTPut4[:STATe] <bool>
	Single Error Addition	SOURCE[1]:EADdition:SINgle
	ReSync	SENSe[1]:SYNChronisat[:AUTO] ONCE
	Buzzer	SYSTem:BEePer:DERRor[:STATe] <bool> SYSTem:BEePer:ALARm[:STATe] <bool>
	Clear History	SENSe[1]:MONitor:HIStory:CLear
	Set Installation Defaults	*RST

## A.2.2 [Settings] ダイアログ・ボックス

### A.2.2.1 SSG モジュール

- Frequency

設定項目	コマンド
Output	SOURCE[0]:OUTPUT[:STATE] <bool>
Frequency	SOURCE[0]:FREQUENCY[:CW] <int>
Reference	SOURCE[0]:REFERENCE INTERNAL EXTERNAL
Jitter Modulation	SOURCE[0]:JMODULATION[:STATE] <bool>
Frequency	SOURCE[0]:JMODULATION:FREQUENCY <int>
Amplitude	SOURCE[0]:JMODULATION:AMPLITUDE <real>

### A.2.2.2 PPG モジュール

- Data

設定項目	コマンド						
Track Data	OUTPUT3:TRACK <bool>						
Data	Enable DATA Output	OUTPUT[1][:STATE] <bool>					
	Cross Point	OUTPUT[1]:CROSS <int>					
	Termination	OUTPUT[1]:TERMINATION P1R3V GND M2V CML					
	Termination Voltage	OUTPUT[1]:TERMINATION:VOLTAGE <real>					
	Amplitude	OUTPUT[1]:VOLTAGE[:LEVEL][:IMMEDIATE][:AMPLITUDE] <real>					
	Offset	<table border="1"> <tr> <td>High</td> <td>OUTPUT[1]:VOLTAGE[:LEVEL][:IMMEDIATE]:HIGH &lt;real&gt;</td> </tr> <tr> <td>Middle</td> <td>OUTPUT[1]:VOLTAGE[:LEVEL][:IMMEDIATE]:MIDDLE &lt;real&gt;</td> </tr> <tr> <td>Low</td> <td>OUTPUT[1]:VOLTAGE[:LEVEL][:IMMEDIATE]:LOW &lt;real&gt;</td> </tr> </table>	High	OUTPUT[1]:VOLTAGE[:LEVEL][:IMMEDIATE]:HIGH <real>	Middle	OUTPUT[1]:VOLTAGE[:LEVEL][:IMMEDIATE]:MIDDLE <real>	Low
High	OUTPUT[1]:VOLTAGE[:LEVEL][:IMMEDIATE]:HIGH <real>						
Middle	OUTPUT[1]:VOLTAGE[:LEVEL][:IMMEDIATE]:MIDDLE <real>						
Low	OUTPUT[1]:VOLTAGE[:LEVEL][:IMMEDIATE]:LOW <real>						
XData	Enable XDATA Output	OUTPUT3[:STATE] <bool>					
	Cross Point	OUTPUT3:CROSS <int>					
	Termination	OUTPUT3:TERMINATION P1R3V GND M2V CML					
	Termination Voltage	OUTPUT3:TERMINATION:VOLTAGE <real>					
	Amplitude	OUTPUT3:VOLTAGE[:LEVEL][:IMMEDIATE][:AMPLITUDE] <real>					
	Offset	<table border="1"> <tr> <td>High</td> <td>OUTPUT3:VOLTAGE[:LEVEL][:IMMEDIATE]:HIGH &lt;real&gt;</td> </tr> <tr> <td>Middle</td> <td>OUTPUT3:VOLTAGE[:LEVEL][:IMMEDIATE]:MIDDLE &lt;real&gt;</td> </tr> <tr> <td>Low</td> <td>OUTPUT3:VOLTAGE[:LEVEL][:IMMEDIATE]:LOW &lt;real&gt;</td> </tr> </table>	High	OUTPUT3:VOLTAGE[:LEVEL][:IMMEDIATE]:HIGH <real>	Middle	OUTPUT3:VOLTAGE[:LEVEL][:IMMEDIATE]:MIDDLE <real>	Low
High	OUTPUT3:VOLTAGE[:LEVEL][:IMMEDIATE]:HIGH <real>						
Middle	OUTPUT3:VOLTAGE[:LEVEL][:IMMEDIATE]:MIDDLE <real>						
Low	OUTPUT3:VOLTAGE[:LEVEL][:IMMEDIATE]:LOW <real>						

A.2.2 [Settings] ダイアログ・ボックス

• Clock

設定項目		コマンド	
Clock Delay		SOURCE2:DELAY <int>	
Track Clock		OUTPUT4:TRACK <bool>	
Clock	Enable CLOCK Output	OUTPUT2[:STATE] <bool>	
	Termination	OUTPUT2:TERMINATION P1R3V GND M2VICML	
	Termination Voltage	OUTPUT2:TERMINATION:VOLTAGE <real>	
	Amplitude	OUTPUT2:VOLTAGE[:LEVEL][:IMMEDIATE][:AMPLITUDE] <real>	
	Offset	High	OUTPUT2:VOLTAGE[:LEVEL][:IMMEDIATE]:HIGH <real>
		Middle	OUTPUT2:VOLTAGE[:LEVEL][:IMMEDIATE]:MIDDLE <real>
Low		OUTPUT2:VOLTAGE[:LEVEL][:IMMEDIATE]:LOW <real>	
XCLOCK	Enable XCLOCK Output	OUTPUT4[:STATE] <bool>	
	Termination	OUTPUT4:TERMINATION P1R3V GND M2VICML	
	Termination Voltage	OUTPUT4:TERMINATION:VOLTAGE <real>	
	Amplitude	OUTPUT4:VOLTAGE[:LEVEL][:IMMEDIATE][:AMPLITUDE] <real>	
	Offset	High	OUTPUT4:VOLTAGE[:LEVEL][:IMMEDIATE]:HIGH <real>
		Middle	OUTPUT4:VOLTAGE[:LEVEL][:IMMEDIATE]:MIDDLE <real>
Low		OUTPUT4:VOLTAGE[:LEVEL][:IMMEDIATE]:LOW <real>	
Track Clock		OUTPUT4:TRACK <bool>	

• Pattern

設定項目		コマンド
Pattern Polarity		SOURCE[1]:PATTERN:POLARITY NORMAL INVERTED
Pattern Type		SOURCE[1]:PATTERN[:SELECT] PRBS<n> ZSUBstitut<n> PROGRAM STM<n> FLEXible
PRBS	Pattern Length	SOURCE[1]:PATTERN[:SELECT] PRBS<n>
	Mark Ratio	SOURCE[1]:PATTERN:PRBS:MRATIO MRATIO<n> MINVERTED4
ZSUB	Pattern Length	SOURCE[1]:PATTERN[:SELECT] ZSUBstitut<n>
	Zero Length	SOURCE[1]:PATTERN:ZSUBstitut[:ZLENGTH] <int>
PROG	Pattern Length	SOURCE[1]:PATTERN:PROGRAM[:LENGTH] <int>
	Pattern	SOURCE[1]:PATTERN:PROGRAM:DATA <start_bit>,<length>,<str> SOURCE[1]:PATTERN:PROGRAM:BDATA <start_bit>,<length>,<block>



設定項目		コマンド	
STM	Insert PRBS into Payload	SOURCE[1]:PATTERN:STM:PAYLOAD:PRBS <bool>	
	Pattern Length	SOURCE[1]:PATTERN:STM:PAYLOAD:PRBS:SELECT PRBS<n>	
	Mark Ratio	SOURCE[1]:PATTERN:STM:PAYLOAD:PRBS:MRATIO MRATIO<n> MINVERTED4	
	Scramble	SOURCE[1]:PATTERN:STM:SCRAMBLE <bool>	
	Insert B1	SOURCE[1]:PATTERN:STM:B1 <bool>	
	Number of Frame	SOURCE[1]:PATTERN:STM:NFRAME <int>	
	Pattern	SOURCE[1]:PATTERN:STM:DATA <start_frm>,<start_row>,<start_col>,<length>,<str> SOURCE[1]:PATTERN:STM:BDATA <start_frm>,<start_row>,<start_col>,<length>,<block>	
FLEX	Pattern Sequence Table	New	SOURCE[1]:PATTERN:FLEXIBLE:STABLE:NEW <index>,<prog_no> PRBS,<length>,HIGH LOW
		Edit	SOURCE[1]:PATTERN:FLEXIBLE:STABLE <index>,<prog_no> PRBS,<length>,HIGH LOW
		Remove	SOURCE[1]:PATTERN:FLEXIBLE:STABLE:REMOVE <index>
		Default	SOURCE[1]:PATTERN:FLEXIBLE:STABLE:DEFAULT
	PROG	Pattern Length	SOURCE[1]:PATTERN:FLEXIBLE:PROGRAM[:LENGTH] <prog_no>,<length>
		Pattern	SOURCE[1]:PATTERN:FLEXIBLE:PROGRAM:DATA <prog_no>,<start_bit>,<length>,<str> SOURCE[1]:PATTERN:FLEXIBLE:PROGRAM:BDATA <prog_no>,<start_bit>,<length>,<block>
	PRBS	Pattern Length	SOURCE[1]:PATTERN:FLEXIBLE:PRBS:SELECT PRBS<n>
		Mark Ratio	SOURCE[1]:PATTERN:FLEXIBLE:PRBS:MRATIO MRATIO<n> MINVERTED4

• Burst

設定項目		コマンド
Burst Mode		SOURCE5:BOUtput[:STATE] <bool>
Source		SOURCE5:BOUtput:SOURCE INTERNAL EXTERNAL
Internal	Cycle	SOURCE5:BOUtput:CYCLE <int>
	OFF Time	SOURCE5:BOUtput:OTIME <int>

A.2.2 [Settings] ダイアログ・ボックス

• Error Addition

設定項目		コマンド
Addition Route		SOURCE[1]:EADDITION:ROUTE <int>
Add Errors		SOURCE[1]:EADDITION <bool>
Single		SOURCE[1]:EADDITION:MODE SINGLE
Repeat		SOURCE[1]:EADDITION:MODE REPEAT
Rate		SOURCE[1]:EADDITION:RATE <real>
External		SOURCE[1]:EADDITION:MODE EXTERNAL

• Trigger/AUX

設定項目		コマンド
Trigger Output		SOURCE6:TRIGGER[:MODE] CLOCK<8 CLOCK32 PATTERN FLEXIBLE
Position	PRBS	SOURCE6:TRIGGER:PRBS:POSITION <int>
	PROG	SOURCE6:TRIGGER:PROGRAM:POSITION <int>
ZSUB		SOURCE6:TRIGGER:ZSUBSTITUT:POSITION <int>
STM	Pattern/Frame	SOURCE6:TRIGGER:STM:MODE PATTERN FRAME
	Frame No.	SOURCE6:TRIGGER:STM:FNUMBER <int>
	Row	SOURCE6:TRIGGER:STM:ROW <int>
	Column	SOURCE6:TRIGGER:STM:COLUMN <int>
FLEX	Index	SOURCE6:TRIGGER:FLEXIBLE:INDEX <int>
	Address	SOURCE6:TRIGGER:FLEXIBLE:ADDRESS <int>

### A.2.2.3 ED モジュール

- Data

設定項目	コマンド
Data Polarity	INPut[1]:POLarity NORMal INVerted
Termination	INPut[1]:TERMination: P3V P1R3V GND M2V CML
Variable	INPut[1]:TERMination:VARIable <bool>
Termination Voltage	INPut[1]:TERMination:VOLTagE <real>
Threshold Voltage	INPut[1]:TVOLtagE <real>

- Clock

設定項目	コマンド
Clock Delay	INPut2:DElay <int>
Clock Polarity	INPut2:POLarity NORMal INVerted
Termination	INPut2:TERMination: P3V P1R3V GND M2V CML
Variable	INPut2:TERMination:VARIable <bool>
Termination Voltage	INPut2:TERMination:VOLTagE <real>

A.2.2 [Settings] ダイアログ・ボックス

• Pattern

設定項目		コマンド	
Use the same Pattern as PPG		SENSe[1]:PATtern:TRACk <bool>	
Pattern Polarity		SENSe[1]:PATtern:POLarity NORMal INVerted	
Pattern Type		SENSe[1]:PATtern[:SElect] PRBS<n> ZSUBstitut<n> PROGram STM<r> FLEXible	
PRBS	Pattern Length	SENSe[1]:PATtern[:SElect] PRBS<n>	
	Mark Ratio	SENSe[1]:PATtern:PRBS:MRATio MRATio<n>  MINVerted4	
ZSUB	Pattern Length	SENSe[1]:PATtern[:SElect] ZSUBstitut<n>	
	Zero Length	SENSe[1]:PATtern:ZSUBstitut[:ZLEngth] <int>	
PROG	Pattern Length	SENSe[1]:PATtern:PROGram[:LENGth] <int>	
	Pattern	SENSe[1]:PATtern:PROGram:DATA <start_bit>,<length>,<str> SENSe[1]:PATtern:PROGram:BDATA <start_bit>,<length>,<block>	
STM	Insert PRBS into Payload	SENSe[1]:PATtern:STM:PAYLoaD:PRBS <bool>	
	Pattern Length	SENSe[1]:PATtern:STM:PAYLoaD:PRBS:SElect PRBS<n>	
	Mark Ratio	SENSe[1]:PATtern:STM:PAYLoaD:PRBS:MRATio MRATio<r> MINVerted4	
	Scramble	SENSe[1]:PATtern:STM:SCRamble <bool>	
	Insert B1	SENSe[1]:PATtern:STM:B1 <bool>	
	Number of Frame	SENSe[1]:PATtern:STM:NFRame <int>	
	Pattern	SENSe[1]:PATtern:STM:DATA <start_frm>,<start_row>,<start_col>,<length>,<str> SENSe[1]:PATtern:STM:BDATA <start_frm>,<start_row>,<start_col>,<length>,<block>	
FLEX	Pattern Sequence Table	New	SENSe[1]:PATtern:FLEXible:STABLE:NEW <index>,<prog_no> PRBS,<length>,HIGH LOW
		Edit	SENSe[1]:PATtern:FLEXible:STABLE <index>,<prog_no> PRBS,<length>,HIGH LOW
		Remove	SENSe[1]:PATtern:FLEXible:STABLE:REMOve <index>
		Default	SENSe[1]:PATtern:FLEXible:STABLE:DEFault
	PROG	Pattern Length	SENSe[1]:PATtern:FLEXible:PROGram[:LENGth] <prog_no>,<length>
		Pattern	SENSe[1]:PATtern:FLEXible:PROGram:DATA <prog_no>,<start_bit>,<length>,<str> SENSe[1]:PATtern:FLEXible:PROGram:BDATA <prog_no>,<start_bit>,<length>,<block>
	PRBS	Pattern Length	SENSe[1]:PATtern:FLEXible:PRBS:SElect PRBS<n>
		Mark Ratio	SENSe[1]:PATtern:FLEXible:PRBS:MRATio MRATio<r> MINVerted4

- Sync

設定項目	コマンド
Auto Sync	SENSE[1]:SYNChronisat[:AUTO] <bool>
Sync Pattern Length	SENSE[1]:SYNChronisat:PATtern:LENGth <int>
Sync PROG Address	SENSE[1]:SYNChronisat:PATtern:PROGraw:ADDRes <int>
Sync FLEX Index	SENSE[1]:SYNChronisat:PATtern:FLEXible:INDEx <int>
Sync FLEX Address	SENSE[1]:SYNChronisat:PATtern:FLEXible:ADDRes <int>
Auto Threshold	SENSE[1]:SYNChronisat:THResholD:AUTO <bool>
Sync Gain Threshold	PRBS SENSE[1]:SYNChronisat:THResholD:PRBS:GAIN <real>
	Memory SENSE[1]:SYNChronisat:THResholD:MEMory:GAIN <real>
Sync Loss Threshold	PRBS SENSE[1]:SYNChronisat:THResholD:PRBS:LOSS <real>
	Memory SENSE[1]:SYNChronisat:THResholD:MEMory:LOSS <real>

- Auto Search

設定項目	コマンド
Clock Polarity & Delay	SENSE[1]:ASEarch:ClOCK <bool>
Threshold Voltage	SENSE[1]:ASEarch:TVOltage <bool>
Pattern	SENSE[1]:ASEarch:PATtern <bool>

- Mask

設定項目	コマンド
Mask Route	SENSE[1]:MASK:ROUTE<n> <bool>

- Trigger/AUX

設定項目	コマンド
Trigger Output	SOURCE8:TRIGger[:MODE] ClOCK16 PATtern FLEXible
STM	SOURCE8:TRIGger:STM:MODE PATtern FRAMe
AUX	SOURCE9:AUX[:MODE] DTYPe SYNChronisat

A.2.2 [Settings] ダイアログ・ボックス

- Log

設定項目	コマンド
Basic Measurement Results	SENSE[1]:LOG:BMMeasurement <bool>
Sync & Clock Loss History	SENSE[1]:LOG:HISTORY:SCLoss <bool>
Measurement Data History	
Frequency	SENSE[1]:LOG:HISTORY:MDATA:FREQUENCY <bool>
Immediate Error Rate	SENSE[1]:LOG:HISTORY:MDATA:ERate <bool>
Immediate Error Count	SENSE[1]:LOG:HISTORY:MDATA:ECOUNT <bool>
Error Intervals	SENSE[1]:LOG:HISTORY:MDATA:ERRORINTERVALS <bool>
Error Free Intervals	SENSE[1]:LOG:HISTORY:MDATA:ERRORFREEINTERVALS <bool>

- Condition

設定項目	コマンド
Timer Mode	SENSE[1]:BMEasurement:MTIME:MODE SINGLE REPEAT UNTIMED
Interval	SENSE[1]:BMEasurement:MTIME:INTERVAL <real>
Error Performance Threshold	SENSE[1]:BMEasurement:EPERFORMANCE:THRESHOLD <us_ses>,<cm>
Period	SENSE[1]:BMEasurement:MTIME:PERIOD <day>,<hour>,<minute>,<second>
Detection Mode	SENSE[1]:BMEasurement:DMODE OFF OPTIMIZE
Burst Mode	SENSE[1]:BMEasurement:BURST <bool>
Current Data	SENSE[1]:BMEasurement:CDAData <bool>
Evaluate Clock Loss Intervals	SENSE[1]:BMEasurement:EVALUATE:CLOSS <bool>
Evaluate Sync Loss Intervals	SENSE[1]:BMEasurement:EVALUATE:SLLOSS <bool>
Error Record	SENSE[1]:EPANalysis:ERECORD[:STATE] <bool>
Specific/Total	SENSE[1]:EPANalysis:ERECORD:AREA SPECIFIC TOTAL

- Display Format

設定項目	コマンド
Bit Count	DISPlay:WINDow:BCOunt:DFORmat INTEGRal EXPonential
Error Count	DISPlay:WINDow:ECOunt:DFORmat INTEGRal EXPonential
Immediate Error Count	DISPlay:WINDow:IECOunt:DFORmat INTEGRal EXPonential
Error Intervals	DISPlay:WINDow:EINtervals:DFORmat INTEGRal PERCent
Error Free Intervals	DISPlay:WINDow:EFINtervals:DFORmat INTEGRal PERCent
Threshold EI	DISPlay:WINDow:TEINtervals:DFORmat INTEGRal PERCent
Threshold EFI	DISPlay:WINDow:TEFINtervals:DFORmat INTEGRal PERCent
Error Performance	DISPlay:WINDow:EPERformance:DFORmat INTEGRal PERCent

- Buzzer

設定項目	コマンド
Error	SYSTem:BEERPer:DERror[:STATe] <bool>
Alarm	SYSTem:BEERPer:ALARn[:STATe] <bool>

- Jitter Tolerance

設定項目	コマンド
New	JTOLerance:TABLE:NEW <index>,<freq>,<min_ampl>,<max_ampl>,<points>
Edit	JTOLerance:TABLE <index>,<freq>,<min_ampl>,<max_ampl>,<points>
Remove	JTOLerance:TABLE:REMOve <index>
Measurement Mode	JTOLerance:MODE SEARCH SWEEg
Display Measurement Data	JTOLerance:MDATA <bool>
Template Type	JTOLerance:TEMPlate:SELEct <str>
Auto Search	JTOLerance:ASEArch <bool>
Clock Frequency	JTOLerance:CFrequency <int>
Settling Time	JTOLerance:STIME <real>
Period	JTOLerance:PERiod <int>
Error Threshold	JTOLerance:ETHReshold <int>

- Specific Field

設定項目		コマンド
PROG	Start Address	SENSe[1]:SPECific:PROG:START:ADDRESS <int>
	End Address	SENSe[1]:SPECific:PROG:END:ADDRESS <int>
ZSUB	Start Address	SENSe[1]:SPECific:ZSUBstitut:START:ADDRESS <int>
	End Address	SENSe[1]:SPECific:ZSUBstitut:END:ADDRESS <int>
STM	Pattern/Frame	SENSe[1]:SPECific:STM:MODE PATtern FRAME
	Start Frame No.	SENSe[1]:SPECific:STM:START:FNUMBER <int>
	Start Row	SENSe[1]:SPECific:STM:START:ROW <int>
	Start Column	SENSe[1]:SPECific:STM:START:COLUm <int>
	End Frame No.	SENSe[1]:SPECific:STM:END:FNUMBER <int>
	End Row	SENSe[1]:SPECific:STM:END:ROW <int>
	End Column	SENSe[1]:SPECific:STM:END:COLUm <int>
FLEX	Start Index	SENSe[1]:SPECific:FLEXible:START:INDEX <int>
	Start Address	SENSe[1]:SPECific:FLEXible:START:ADDRESS <int>
	End Index	SENSe[1]:SPECific:FLEXible:END:INDEX <int>
	End Address	SENSe[1]:SPECific:FLEXible:END:ADDRESS <int>



#### A.2.2.4 System モジュール

- Utility

設定項目	コマンド
Date	SYSTEM:DATE <year>,<month>,<day>
Time	SYSTEM:TIME <hour>,<minute>,<second>

## A.3 初期設定

## A.3 初期設定

## A.3.1 PPG モジュール

設定項目	初期値
パターン・タイプ	PRBS15
PRBS パターンのマーク率	1/2
ZSUB パターン長	2 <sup>7</sup> bits
ゼロ連続 bit 長	7 bits
PROG パターン長	16 bits
PROG パターン	1010 1010 1010 1010
STM パターンのペイロードへの PRBS パターン挿入機能の ON/OFF	OFF
STM パターンのペイロードへ挿入する PRBS パターンのパターン長	PRBS15
STM パターンのペイロードへ挿入する PRBS パターンのマーク率	1/2
STM パターンのスクランブル機能の ON/OFF	OFF
STM パターンの B1 バイト挿入機能の ON/OFF	OFF
STM パターン長	STM-16, 1 フレーム
STM パターン	「A.3.5 STM パターン」を参照して下さい。」
FLEX パターン・シーケンス・テーブルのパターン数	1
FLEX パターン・シーケンス・テーブル	「A.3.6 FLEX パターン」を参照して下さい。」
FLEX パターンで使用される PROG パターン長	すべて 128 bits
FLEX パターンで使用される PROG パターン	「A.3.6 FLEX パターン」を参照して下さい。」
FLEX パターンで使用される PRBS パターンのパターン長	PRBS15
FLEX パターンで使用される PRBS パターンのマーク率	1/2
パターン極性	正論理 (NORMAL)
エラー付加の ON/OFF	ON
エラー付加モード	シングル・モード (SINGLE)
エラー付加レート	1E-8
エラー付加ルート	1

設定項目	初期値
DATA 出力の ON/OFF	OFF
DATA 出力振幅 (LVPECL 終端)	0.80 V <sub>P-P</sub>
DATA 出力振幅 (GND 終端)	1.00 V <sub>P-P</sub>
DATA 出力振幅 (ECL 終端)	0.80 V <sub>P-P</sub>
DATA 出力振幅 (CML 終端)	0.40 V <sub>P-P</sub>
DATA 出力の HIGH レベルオフセット値 (LVPECL 終端)	2.40 V
DATA 出力の中間レベルオフセット値 (LVPECL 終端)	2.00 V
DATA 出力の LOW レベルオフセット値 (LVPECL 終端)	1.60 V
DATA 出力の HIGH レベルオフセット値 (GND 終端)	0.00 V
DATA 出力の中間レベルオフセット値 (GND 終端)	-0.50
DATA 出力の LOW レベルオフセット値 (GND 終端)	-1.00
DATA 出力の HIGH レベルオフセット値 (ECL 終端)	-0.90 V
DATA 出力の中間レベルオフセット値 (ECL 終端)	-1.30 V
DATA 出力の LOW レベルオフセット値 (ECL 終端)	-1.70 V
DATA 出力の HIGH レベルオフセット値 (CML 終端)	0.00 V
DATA 出力の中間レベルオフセット値 (CML 終端)	-0.20 V
DATA 出力の LOW レベルオフセット値 (CML 終端)	-0.40 V
DATA 出力のクロスポイント	50%
DATA 出力終端タイプ	GND (0 V)
DATA 出力終端電圧 (CML 終端)	0.00 V
$\overline{\text{DATA}}$ 出力の ON/OFF	OFF
$\overline{\text{DATA}}$ 出力振幅 (LVPECL 終端)	0.80 V <sub>P-P</sub>
$\overline{\text{DATA}}$ 出力振幅 (GND 終端)	1.00 V <sub>P-P</sub>
$\overline{\text{DATA}}$ 出力振幅 (ECL 終端)	0.80 V <sub>P-P</sub>
$\overline{\text{DATA}}$ 出力振幅 (CML 終端)	0.40 V <sub>P-P</sub>
$\overline{\text{DATA}}$ 出力の HIGH レベルオフセット値 (LVPECL 終端)	2.40 V
$\overline{\text{DATA}}$ 出力の中間レベルオフセット値 (LVPECL 終端)	2.00 V
$\overline{\text{DATA}}$ 出力の LOW レベルオフセット値 (LVPECL 終端)	1.60 V
$\overline{\text{DATA}}$ 出力の HIGH レベルオフセット値 (GND 終端)	0.00 V
$\overline{\text{DATA}}$ 出力の中間レベルオフセット値 (GND 終端)	-0.50
$\overline{\text{DATA}}$ 出力の LOW レベルオフセット値 (GND 終端)	-1.00
$\overline{\text{DATA}}$ 出力の HIGH レベルオフセット値 (ECL 終端)	-0.90 V

## A.3.1 PPG モジュール

設定項目	初期値
DATA 出力の中間レベルオフセット値 (ECL 終端)	-1.30 V
DATA 出力の LOW レベルオフセット値 (ECL 終端)	-1.70 V
DATA 出力の HIGH レベルオフセット値 (CML 終端)	0.00 V
DATA 出力の中間レベルオフセット値 (CML 終端)	-0.20 V
DATA 出力の LOW レベルオフセット値 (CML 終端)	-0.40 V
DATA 出力のクロスポイント	50%
DATA 出力終端タイプ	GND (0 V)
DATA 出力終端電圧 (CML 終端)	0.00 V
データ出力のトラッキング機能の ON/OFF	OFF
クロック遅延量	初期化されない
CLOCK 出力の ON/OFF	OFF
CLOCK 出力振幅 (LVPECL 終端)	0.80 V <sub>P-P</sub>
CLOCK 出力振幅 (GND 終端)	1.00 V <sub>P-P</sub>
CLOCK 出力振幅 (ECL 終端)	0.80 V <sub>P-P</sub>
CLOCK 出力振幅 (CML 終端)	0.40 V <sub>P-P</sub>
CLOCK 出力の HIGH レベルオフセット値 (LVPECL 終端)	2.40 V
CLOCK 出力の中間レベルオフセット値 (LVPECL 終端)	2.00 V
CLOCK 出力の LOW レベルオフセット値 (LVPECL 終端)	1.60 V
CLOCK 出力の HIGH レベルオフセット値 (GND 終端)	0.00 V
CLOCK 出力の中間レベルオフセット値 (GND 終端)	-0.50
CLOCK 出力の LOW レベルオフセット値 (GND 終端)	-1.00
CLOCK 出力の HIGH レベルオフセット値 (ECL 終端)	-0.90 V
CLOCK 出力の中間レベルオフセット値 (ECL 終端)	-1.30 V
CLOCK 出力の LOW レベルオフセット値 (ECL 終端)	-1.70 V
CLOCK 出力の HIGH レベルオフセット値 (CML 終端)	0.00 V
CLOCK 出力の中間レベルオフセット値 (CML 終端)	-0.20 V
CLOCK 出力の LOW レベルオフセット値 (CML 終端)	-0.40 V
CLOCK 出力終端タイプ	GND (0 V)
CLOCK 出力終端電圧 (CML 終端)	0.00 V
CLOCK 出力の ON/OFF	OFF
CLOCK 出力振幅 (LVPECL 終端)	0.80 V <sub>P-P</sub>
CLOCK 出力振幅 (GND 終端)	1.00 V <sub>P-P</sub>

設定項目	初期値
CLOCK 出力振幅 (ECL 終端)	0.80 V <sub>P-P</sub>
CLOCK 出力振幅 (CML 終端)	0.40 V <sub>P-P</sub>
CLOCK 出力の HIGH レベルオフセット値 (LVPECL 終端)	2.40 V
CLOCK 出力の中間レベルオフセット値 (LVPECL 終端)	2.00 V
CLOCK 出力の LOW レベルオフセット値 (LVPEC 終端)	1.60 V
CLOCK 出力の HIGH レベルオフセット値 (GND 終端)	0.00 V
CLOCK 出力の中間レベルオフセット値 (GND 終端)	-0.50
CLOCK 出力の LOW レベルオフセット値 (GND 終端)	-1.00
CLOCK 出力の HIGH レベルオフセット値 (ECL 終端)	-0.90 V
CLOCK 出力の中間レベルオフセット値 (ECL 終端)	-1.30 V
CLOCK 出力の LOW レベルオフセット値 (ECL 終端)	-1.70 V
CLOCK 出力の HIGH レベルオフセット値 (CML 終端)	0.00 V
CLOCK 出力の中間レベルオフセット値 (CML 終端)	-0.20 V
CLOCK 出力の LOW レベルオフセット値 (CML 終端)	-0.40 V
CLOCK 出力終端タイプ	GND (0 V)
CLOCK 出力終端電圧 (CML 終端)	0.00 V
クロック出力のトラッキング機能の ON/OFF	OFF
バースト・モードの ON/OFF	OFF
BURST OUTPUT 信号源	内部信号 (INTernal)
バースト周期	1000 μs
バースト周期中の OFF 時間	500 μs
トリガ出力ソース	パターン周波数の 1/8 周波数 (CLOCK/8)
PRBS パターン時のトリガ出力ポジション	0 bit
ZSUB パターン時のトリガ出力ポジション	0 bit
PROG パターン時のトリガ出力ポジション	0 bit
STM パターン時のトリガ出力モード	パターン・モード
STM パターン時のトリガ出力フレーム番号	1
STM パターン時のトリガ出力行	1
STM パターン時のトリガ出力列	1 byte
FLEX パターン時のトリガ出力インデックス	1
FLEX パターン時のトリガ出力アドレス	0 bit

## A.3.2 ED モジュール

設定項目	初期値
パターン・タイプ	PRBS15
PRBS パターンのマーク率	1/2
ZSUB パターン長	2 <sup>7</sup> bits
ゼロ連続 bit 長	7 bits
PROG パターン長	16 bits
PROG パターン	1010 1010 1010 1010
STM パターンのペイロードへの PRBS パターン挿入機能の ON/OFF	OFF
STM パターンのペイロードへ挿入する PRBS パターンのパターン長	PRBS15
STM パターンのペイロードへ挿入する PRBS パターンのマーク率	1/2
STM パターンのスクランブル機能の ON/OFF	OFF
STM パターンの B1 バイト挿入機能の ON/OFF	OFF
STM パターン長	STM-16, 1 フレーム
STM パターン	「A.3.5 STM パターン」を参照して下さい。
FLEX パターン・シーケンス・テーブルのパターン数	1
FLEX パターン・シーケンス・テーブル	「A.3.6 FLEX パターン」を参照して下さい。
FLEX パターンで使用される PROG パターン長	すべて 128 bits
FLEX パターンで使用される PROG パターン	「A.3.6 FLEX パターン」を参照して下さい。
FLEX パターンで使用される PRBS パターンのパターン長	PRBS15
FLEX パターンで使用される PRBS パターンのマーク率	1/2
パターン極性	正論理 (NORMal)
PPG パターンと共通設定の ON/OFF	OFF
自動パターン同期	ON
PROG パターン時の同期パターンアドレス	0 bit
同期パターン長	32 bits
FLEX パターン時の同期パターンインデックス	1 bit
FLEX パターン時の同期パターンアドレス	0 bit

設定項目	初期値
同期スレッシュホールド自動設定の ON/OFF	ON
PRBS パターン時の Sync Gain Threshold	1E-2
PRBS パターン時の Sync Loss Threshold	1E-2
ZSUB, PROG パターン時の Sync Gain Threshold	1E-4
ZSUB, PROG パターン時の Sync Loss Threshold	1E-4
ルートマスクの ON/OFF	すべてのルートが OFF
クロック遅延量とクロック入力極性をオートサーチする機能の ON/OFF	ON
データ入力スレッシュホールド電圧をオートサーチする機能の ON/OFF	ON
PRBS の段数とパターン極性をオートサーチする機能の ON/OFF	ON
基本測定結果のログ記録機能の ON/OFF	ON
シンクロスとクロックロスのヒストリ記録機能の ON/OFF	ON
区間エラーレート値のヒストリ記録機能の ON/OFF	ON
区間エラーカウント値のヒストリ記録機能の ON/OFF	ON
エラーインターバル値のヒストリ記録機能の ON/OFF	ON
エラーフリーインターバル値のヒストリ記録機能の ON/OFF	ON
周波数値のヒストリ記録機能の ON/OFF	ON
測定モード	アンタイムドモード (UNTimed)
測定インターバル	0.1 s
測定期間	10 s
エラー検出モード	欠落エラー (Omitting) / 挿入エラー (Inserting) / Total エラー
BURST モードの ON/OFF	OFF
クロックロスインターバルを測定に含めるか否かの選択	ON
シンクロスインターバルを測定に含めるか否かの選択	ON
US/SES, DM の計算に含める期間のエラーレートのスレッシュホールド	US/SES:1E-3, DM:1E-6
途中結果表示 ON/OFF	ON
エラー記録機能の ON/OFF	ON
エラー記録領域	パターン全体
ジッタ耐力パラメータ・テーブルのリスト数	15

## A.3.2 ED モジュール

設定項目	初期値
ジッタ耐力パラメータ・テーブル	「A.3.7 ジッタ耐力パラメータ・テーブル」を参照して下さい。
ジッタ耐力測定モード	サーチ・モード
ジッタ耐力測定データ表示の ON/OFF	ON
ジッタ耐力測定のテンプレート	STM-16 (A)
ジッタ耐力測定のオートサーチ機能の ON/OFF	ON
ジッタ耐力測定のクロック周波数	2,488,320 kHz
ジッタ耐力測定のセットリング時間	0.0 s
ジッタ耐力測定の1ポイントあたりの測定時間	1 s
ジッタ耐力測定の判定基準となるエラー・カウント	1 bit
PROG パターン時の特定領域開始アドレス	0 bit
PROG パターン時の特定領域終了アドレス	0 bit
ZSUB パターン時の特定領域開始アドレス	0 bit
ZSUB パターン時の特定領域終了アドレス	0 bit
STM パターン時の特定領域測定モード	パターン・モード
STM パターン時の特定領域開始フレーム番号	1
STM パターン時の特定領域開始行	1
STM パターン時の特定領域開始列	1 byte
STM パターン時の特定領域終了フレーム番号	1
STM パターン時の特定領域終了行	1
STM パターン時の特定領域終了列	1 byte
FLEX パターン時の特定領域開始インデックス	1
FLEX パターン時の特定領域開始アドレス	0 bit
FLEX パターン時の特定領域終了インデックス	1
FLEX パターン時の特定領域終了アドレス	0 bit
DATA 入力極性	非反転 (NORMAL)
DATA 入力スレッシュホールド電圧 (PECL 終端)	3.700 V
DATA 入力スレッシュホールド電圧 (LVPECL 終端)	2.000 V
DATA 入力スレッシュホールド電圧 (GND 終端)	-0.500 V
DATA 入力スレッシュホールド電圧 (ECL 終端)	-1.300 V
DATA 入力スレッシュホールド電圧 (CML 終端)	-0.200 V
DATA 入力終端タイプ	GNG (0 V)



設定項目	初期値
DATA 入力の終端電圧可変機能の ON/OFF (PECL 終端)	OFF
DATA 入力の終端電圧可変機能の ON/OFF (LVPECL 終端)	OFF
DATA 入力の終端電圧可変機能の ON/OFF (ECL 終端)	OFF
DATA 入力の終端電圧可変機能の ON/OFF (CML 終端)	OFF
DATA 入力の終端電圧 (PECL 終端)	+3.00 V
DATA 入力の終端電圧 (LVPECL 終端)	+1.30 V
DATA 入力の終端電圧 (ECL 終端)	-2.00 V
DATA 入力の終端電圧 (CML 終端)	0.00 V
CLOCK 入力の極性	非反転 (NORMAL)
CLOCK 入力遅延量	初期値なし
CLOCK 入力終端タイプ	GND (0 V)
CLOCK 入力の終端電圧可変機能の ON/OFF (PECL 終端)	OFF
CLOCK 入力の終端電圧可変機能の ON/OFF (LVPECL 終端)	OFF
CLOCK 入力の終端電圧可変機能の ON/OFF (ECL 終端)	OFF
CLOCK 入力の終端電圧可変機能の ON/OFF (CML 終端)	OFF
CLOCK 入力の終端電圧 (PECL 終端)	+3.00 V
CLOCK 入力の終端電圧 (LVPECL 終端)	+1.30 V
CLOCK 入力の終端電圧 (ECL 終端)	-2.00 V
CLOCK 入力の終端電圧 (CML 終端)	0.00 V
トリガ出力ソース	パターン周波数の 1/16 周波数 (CLOCK16)
STM パターン時のトリガ出力モード	パターン・モード
補助出力モード	データ・タイプ

## A.3.3 結果、システム

## A.3.3 結果、システム

設定項目	初期値
ビット・カウント値の表示フォーマット	指数形式 (EXponential)
エラー・カウント値の表示フォーマット	指数形式 (EXponential)
区間エラー・カウント値の表示フォーマット	指数形式 (EXponential)
エラー・インターバル値の表示フォーマット	百分率形式 (PERCent)
エラー・フリー・インターバル値の表示フォーマット	百分率形式 (PERCent)
スレッシュホールド EI 値の表示フォーマット	百分率形式 (PERCent)
スレッシュホールド EFI 値の表示フォーマット	百分率形式 (PERCent)
エラー・パフォーマンス値の表示フォーマット	百分率形式 (PERCent)
データ・エラー発生時のブザーの ON/OFF	OFF
アラーム発生時のブザーの ON/OFF	OFF
リモート・インタフェース	初期化されない
GPIO アドレス	初期化されない

## A.3.4 SSG モジュール

設定項目	初期値
クロック出力の ON/OFF	ON
周波数	2,500,000 kHz
クロック基準信号の入力ソース	内部信号 (INTERNAL)
ジッタ変調機能の ON/OFF	OFF
ジッタ周波数	10 Hz
ジッタ振幅	0.00 UI <sub>P-P</sub>



## A.3.6 FLEX パターン

## A.3.6 FLEX パターン

## A.3.6.1 パターン・シーケンス・テーブル

Index	Pattern	Length	FLEX Trigger
1	PROG001	128 bits	Low

## A.3.6.2 PROG パターン

FLEX で使用される PROG パターンは、PROG No.1 ~ 127 まですべて以下のパターンで初期化されます。

Pattern No.	Pattern Length	Pattern
1 ~ 127	128 bits	1010 1010 (2 進) の繰り返しパターン

## A.3.7 ジッタ耐力パラメータ・テーブル

インデックス	ジッタ周波数	ジッタ振幅		
		最小値	最大値	測定ポイント数
1	10 Hz	1.50 UI <sub>p-p</sub>	800.00 UI <sub>p-p</sub>	5
2	100 Hz	1.50 UI <sub>p-p</sub>	100.00 UI <sub>p-p</sub>	5
3	1,000 Hz	1.50 UI <sub>p-p</sub>	50.00 UI <sub>p-p</sub>	5
4	5,000 Hz	1.50 UI <sub>p-p</sub>	50.00 UI <sub>p-p</sub>	5
5	10,000 Hz	1.50 UI <sub>p-p</sub>	20.00 UI <sub>p-p</sub>	5
6	20,000 Hz	1.50 UI <sub>p-p</sub>	20.00 UI <sub>p-p</sub>	5
7	50,000 Hz	1.50 UI <sub>p-p</sub>	20.00 UI <sub>p-p</sub>	5
8	100,000 Hz	1.50 UI <sub>p-p</sub>	20.00 UI <sub>p-p</sub>	10
9	200,000 Hz	0.75 UI <sub>p-p</sub>	20.00 UI <sub>p-p</sub>	10
10	500,000 Hz	0.30 UI <sub>p-p</sub>	12.00 UI <sub>p-p</sub>	10
11	1,000,000 Hz	0.15 UI <sub>p-p</sub>	6.00 UI <sub>p-p</sub>	10
12	2,000,000 Hz	0.15 UI <sub>p-p</sub>	3.00 UI <sub>p-p</sub>	10
13	5,000,000 Hz	0.15 UI <sub>p-p</sub>	1.20 UI <sub>p-p</sub>	10
14	10,000,000 Hz	0.15 UI <sub>p-p</sub>	0.60 UI <sub>p-p</sub>	5
15	20,000,000 Hz	0.15 UI <sub>p-p</sub>	0.30 UI <sub>p-p</sub>	5

## 索引

## [ シンボル ]

*CLS .....	5-14
*ESE .....	5-14
*ESR? .....	5-15
*IDN? .....	5-15
*OPC .....	5-16
*OPT? .....	5-18
*RST .....	5-16
*SRE .....	5-17
*STB? .....	5-17
*TST? .....	5-17
*WAI .....	5-18
[Settings] ダイアログ・ボックス .....	A-27

## [C]

CFETch[:SENSe[1]]:	
BMEasurement .....	5-105, 5-107, 5-109, 5-111, 5-112, 5-113, 5-115, 5-117, 5-118, 5-119, 5-120, 5-121, 5-122

## [D]

DISPlay:	
WINDow .....	5-128, 5-129, 5-130, 5-131
DISPlay サブシステム .....	5-128, A-19

## [E]

ED モジュール .....	5-60, A-10, A-31, A-42
ED モジュールの入出力と サブシステム .....	A-2

## [F]

FETCh/CFETch サブシステム .....	5-105, A-16
FETCh:	
JTOLerance .....	5-125, 5-126, 5-127
FETCh[:SENSe[1]]:	

BMEasurement .....	5-105, 5-107, 5-109, 5-113, 5-115, 5-117, 5-118, 5-119, 5-120, 5-121, 5-122
MONitor .....	5-124
MTIME .....	5-122, 5-123
EPANalysis .....	5-125

## [G]

GPIB アドレスの設定 .....	1-4
GPIB インタフェース機能 .....	2-1
GPIB 各種バッファ .....	2-4
GPIB とは .....	1-1
GPIB の接続 .....	1-3
GPIB のセット・アップ .....	1-3
GPIB バスの機能 .....	2-1

## [H]

HCOpy[:ITEM]:	
ED .....	5-132
JTOLerance .....	5-132
HCOpy サブシステム .....	5-132, A-19

## [I]

INPut[1]:	
POLarity .....	5-97
TERMination .....	5-98, 5-99
TVOLTage .....	5-97
INPut[1] サブシステム .....	5-97, A-14
INPut2:	
DELay .....	5-100
POLarity .....	5-100
TERMination .....	5-101, 5-102
INPut2 サブシステム .....	5-100, A-15

## [J]

JTOLerance:	
ASEarch .....	5-166
CFRequency .....	5-165
ETHReshold .....	5-166
MDATa .....	5-166, 5-167
MODE .....	5-164

索引

PERiod ..... 5-165  
 STIME ..... 5-165  
 TABLE ..... 5-159,  
 5-161,  
 5-162, 5-164  
 TEMPlate ..... 5-164  
 JTOLerance  
 [:STATe] ..... 5-159

**[M]**

MMEMory:  
 LOAD ..... 5-133,  
 5-134,  
 5-135, 5-136  
 STORE ..... 5-136,  
 5-137,  
 5-138,  
 5-139,  
 5-140,  
 5-141, 5-142  
 MMEMory サブシステム ..... 5-133, A-20

**[O]**

OUTPut[1]  
 [:STATe] ..... 5-37  
 OUTPut[1]:  
 CROSS ..... 5-39  
 TERMination ..... 5-39, 5-40  
 VOLTage ..... 5-37, 5-38,  
 5-39  
 OUTPut[1] サブシステム ..... 5-37, A-5  
 OUTPut2  
 [:STATe] ..... 5-45  
 OUTPut2:  
 TERMination ..... 5-47, 5-48  
 VOLTage ..... 5-45, 5-46,  
 5-47  
 OUTPut2 サブシステム ..... 5-45, A-7  
 OUTPut3  
 [:STATe] ..... 5-40  
 OUTPut3:  
 CROSS ..... 5-42  
 TERMination ..... 5-43  
 TRACK ..... 5-43  
 VOLTage ..... 5-41, 5-42  
 OUTPut3 サブシステム ..... 5-40, A-6  
 OUTPut4  
 [:STATe] ..... 5-49  
 OUTPut4:  
 TRACK ..... 5-52  
 TERMination ..... 5-51, 5-52

VOLTage ..... 5-49, 5-50,  
 5-51  
 OUTPut4 サブシステム ..... 5-49, A-7

**[P]**

PPG モジュール ..... 5-19, A-4,  
 A-27, A-38  
 PPG モジュールの入出力と  
 サブシステム ..... A-1

**[S]**

SENSe[1]:  
 ASEarch ..... 5-81, 5-82  
 BMEasurement ..... 5-85, 5-86,  
 5-87, 5-88,  
 5-89  
 EPANalysis ..... 5-95, 5-96  
 LOG ..... 5-83, 5-84,  
 5-85  
 MASK ..... 5-81  
 MONitor ..... 5-83  
 PATtern ..... 5-60, 5-61,  
 5-62, 5-63,  
 5-64, 5-65,  
 5-66, 5-67,  
 5-68, 5-69,  
 5-70, 5-71,  
 5-72, 5-73,  
 5-74, 5-75,  
 5-76  
 SPECific ..... 5-90, 5-91,  
 5-92, 5-93,  
 5-94, 5-95  
 SYNChronisat ..... 5-76, 5-77,  
 5-78, 5-79,  
 5-80  
 SENSe[1] サブシステム ..... 5-60, A-10  
 SOURce[1]:  
 EADdition ..... 5-35, 5-36  
 PATtern ..... 5-19, 5-20,  
 5-21, 5-22,  
 5-23, 5-24,  
 5-25, 5-26,  
 5-27, 5-28,  
 5-29, 5-30,  
 5-31, 5-32,  
 5-33, 5-34,  
 5-35  
 SOURce[1] サブシステム ..... 5-19, A-4  
 SOURce10:  
 FREQuency[:CW] ..... 5-155

- JMODulation ..... 5-156  
 JMODulation[:STATe] ..... 5-156, 5-157  
 OUTPut[:STATe] ..... 5-155  
 REFerence ..... 5-156  
 SOURce10 サブシステム ..... 5-155, A-23  
 SOURce2:  
 DELay ..... 5-44  
 SOURce2 サブシステム ..... 5-44, A-7  
 SOURce5:  
 BOUtpuT ..... 5-53, 5-54  
 SOURce5 サブシステム ..... 5-53, A-8  
 SOURce6:  
 TRIGger ..... 5-56, 5-57,  
 5-58, 5-55,  
 5-56  
 SOURce6 サブシステム ..... 5-55, A-8  
 SOURce7 サブシステム ..... 5-59, A-9  
 SOURce8:  
 TRIGger ..... 5-103  
 SOURce8 サブシステム ..... 5-103, A-15  
 SOURce9:  
 AUX[:MODE] ..... 5-104  
 SOURce9 サブシステム ..... 5-104, A-15  
 SSG モジュール ..... 5-155, A-23,  
 A-24, A-27,  
 A-46  
 SSG モジュールの入出力と  
 サブシステム ..... A-2  
 STATus:  
 DEvIce ..... 5-153  
 OPErAtion ..... 5-146,  
 5-147,  
 5-148, 5-149  
 PRESet ..... 5-154  
 QUEStionable ..... 5-149,  
 5-150,  
 5-151,  
 5-152, 5-153  
 STATus サブシステム ..... 5-146, A-22  
 SYSTem:  
 BEEPer ..... 5-143  
 DATE ..... 5-145  
 ERRor? ..... 5-144  
 TIME ..... 5-145  
 SYSTem サブシステム ..... 5-143, A-21  
 System モジュール ..... A-37
- 【あ】**
- インタフェース・クリア (IFC) ..... 2-2  
 インタフェース・メッセージに  
 対する応答 ..... 2-2  
 オペレーション・ステータス・  
 レジスタ ..... 4-8
- 【か】**
- 共通コマンド ..... 5-14, A-3  
 クエスチョナブル・ステータス・  
 レジスタ ..... 4-9  
 クエリの受信と応答データの生成 ..... 2-5  
 クロックロス・ステータス・  
 レジスタ ..... 4-9  
 結果、システム ..... A-46  
 結果、システム・コマンド ..... 5-105, A-16  
 ゴー・トゥー・ローカル (GTL) ..... 2-3  
 コマンド一覧 ..... A-1  
 コマンド記述上の注意 ..... 5-12  
 コマンド文法 ..... 3-1  
 コマンド・モード ..... 1-2  
 コマンド・リファレンス ..... 5-1
- 【さ】**
- サブシステムと  
 入出力インタフェース ..... A-1  
 サブシステムの構成 ..... 5-10  
 サンプル・プログラム ..... 6-1  
 初期設定 ..... A-38  
 シリアル・ポール・イネーブル  
 (SPE) ..... 2-2  
 ステータス・バイト ..... 4-1  
 ステータス・バイト・レジスタ ..... 4-5  
 ステータス・レジスタ ..... 4-1  
 ステータス・レジスタのクリア、  
 リセット方法 ..... 4-11  
 ステータス・レジスタの構造 ..... 4-1  
 ステータス・レジスタの種類 ..... 4-2  
 設定メニューに対応する  
 コマンド一覧 ..... A-25  
 セトリング・ステータス・レジスタ ..... 4-9  
 セレクトッド・デバイス・クリア  
 (SDC) ..... 2-3  
 測定条件の設定および読み込みの  
 プログラム例 ..... 6-1  
 測定プログラム例 ..... 6-4
- 【た】**
- データ・フォーマット ..... 3-3  
 デバイス・クリア (DCL) ..... 2-3  
 デバイス・ステータス・レジスタ ..... 4-10

索引

デリミタ (ターミネータ) ..... 3-5

**【は】**

はじめに ..... 1-1  
標準イベント・ステータス・  
レジスタ ..... 4-7  
付録 ..... A-1

**【ま】**

メッセージ交換プロトコル ..... 2-4  
メニュー ..... A-25

**【ら】**

リモート・イネーブル (REN) ..... 2-2  
ローカル・ロック・アウト (LLO) ..... 2-3



## 本製品に含まれるソフトウェアのご使用について

本製品に含まれるソフトウェア（以下本ソフトウェア）のご使用について以下のことにご注意下さい。

ここでいうソフトウェアには、本製品に含まれる又は共に使用されるコンピュータ・プログラム、将来弊社よりお客様に提供されることのある追加、変更、修正プログラムおよびアップデート版のコンピュータ・プログラム、ならびに本製品に関する取扱説明書等の付随資料を含みます。

### 使用許諾

本ソフトウェアの著作権を含む一切の権利は弊社に帰属いたします。

弊社は、本ソフトウェアを本製品上または本製品とともに使用する限りにおいて、お客様に使用を許諾するものといたします。

### 禁止事項

お客様は、本ソフトウェアのご使用に際し以下の事項は行わないで下さい。

- 本製品使用目的以外で使用する事
- 許可なく複製、修正、改変を行う事
- リバース・エンジニアリング、逆コンパイル、逆アセンブルなどを行う事

### 免責

お客様が、本製品を通常の用法以外の用法で使用したことにより本製品に不具合が発生した場合、およびお客様と第三者との間で著作権等に関する紛争が発生した場合、弊社は一切の責任を負いかねますのでご了承下さい。

# 保証について

製品の保証期間は、お客様と別段の取り決めがある場合または当社が特に指定した場合を除き、製品の納入日(システム機器については検取日)から1年間といたします。保証期間中に、当社の責めに帰する製造上の欠陥により製品が故障した場合、無償で修理いたします。ただし、下記に該当する場合は、保証期間中であっても保証の対象から除外させていただきます。

- 当社が認めていない改造または修理を行った場合
- 支給品等当社指定品以外の部品を使用した場合
- 取扱説明書に記載する使用条件を超えて製品を使用した場合(定められた許容範囲を超える物理的ストレスまたは電流電圧がかかった場合など)
- 通常想定される使用環境以外で製品を使用した場合(腐食性の強いガス、塵埃の多い環境等による電気回路の腐食、部品の劣化が早められた場合など)
- 取扱説明書または各種製品マニュアルの指示事項に従わずに使用された場合
- 不注意または不当な取扱により不具合が生じた場合
- お客様のご指示に起因する場合
- 消耗品や消耗材料に基づく場合
- 火災、天変地異等の不可抗力による場合
- 日本国外に持出された場合
- 製品を使用できなかったことによる損失および逸失利益

当社の製品の保証は、本取扱説明書に記載する内容に限られるものとします。

## 保守に関するお問い合わせについて

長期間にわたる信頼性の保証、国家標準とのトレーサビリティを実現するためにアドバンテスでは、工場から出荷された製品の保守に対し、カスタム・エンジニアを配置しています。

カスタム・エンジニアは、故障などの不慮の事故は元より、製品の長期間にわたる性能の保証活動にフィールド・エンジニアとしても活動しています。

万一、動作不良などの故障が発生した場合には、当社のMS(計測器)コールセンターにご連絡下さい。

## 製品修理サービス

- 製品修理期間  
製品の修理サービス期間は、製品の納入後10年間とさせていただきます。
- 製品修理活動  
当社の製品に故障が発生した場合、当社に送っていただく引取り修理、または当社技術員が現地に出張しての出張修理にて対応いたします。

## 製品校正サービス

- 校正サービス  
ご使用中の製品に対し、品質および信頼性の維持を図ることを目的に行うもので、校正後の製品には校正ラベルを貼付けし、品質を保証いたします。
- 校正サービス活動  
校正サービス活動は、株式会社アドバンテス カスタマサポートに送っていただく引取り校正、または当社技術員が現地に出張しての出張校正にて対応いたします。

## 予防保守のおすすめ

製品にはエレクトロニクス部品およびメカニカル部品の一部に寿命を考慮すべき部品を使用しているため、定期的な交換を必要とします。適正な交換期間を過ぎて使用し発生した障害に対しては、修理および性能の保証ができません場合があります。

アドバンテスでは、このようなトラブルを未然に防ぐため、予防保守が有効な手段と考え、予防保守作業を実施する体制を整えています。

各種の予防保守を定期的実施することで、製品の安定稼働を図り、不意の費用発生を防ぐため、年間保守契約による予防保守の実施をお勧めいたします。

なお、年間保守契約は、製品、使用状況および使用環境により内容が変わりますので、最寄りの弊社営業支店にお問い合わせ下さい。

# ADVANTEST

<http://www.advantest.co.jp>

## 株式会社アドバンテス

本社事務所  
〒100-0005 千代田区丸の内1-6-2 新丸の内センタービルディング  
TEL: 03-3214-7500 (代)

第4アカウント販売部(東日本)  
〒100-0005 千代田区丸の内1-6-2 新丸の内センタービルディング  
TEL: 0120-988-971  
FAX: 0120-988-973

第4アカウント販売部(西日本)  
〒564-0062 吹田市垂水町3-34-1  
TEL: 0120-638-557  
FAX: 0120-638-568

### ★計測器に関するお問い合わせ先

(製品の仕様、取扱い、修理・校正等計測器関連全般)

MS(計測器)コールセンタ ☎ TEL 0120-919-570  
FAX 0120-057-508

E-mail: [icc@acs.advantest.co.jp](mailto:icc@acs.advantest.co.jp)