

## SCPI コマンド便覧

---

- ・ 内容一覧表—それぞれの SCPI サブシステムを要約します。
- ・ SCPI サブシステム—それぞれの SCPI サブシステムにあるすべてのコマンドについての詳細な情報を提供します。

## 内容一覧表

表 18-1 から 18-11 は、それぞれの SCPI サブシステムで使うコマンドをまとめたものです。次のリストには、SCPI サブシステムと、各コマンドの要約が掲載されている表の番号が記載してあります。

表 18-1	CALCulate コマンドの要約
表 18-2	DISPlay コマンドの要約
表 18-3	FORMat コマンドの要約
表 18-4	OUTPut コマンドの要約
表 18-5	ROUTe コマンドの要約
表 18-6	SENSe コマンドの要約
表 18-7	SOURce コマンドの要約
表 18-8	STATus コマンドの要約
表 18-9	SYSTem コマンドの要約
表 18-10	TRACe コマンドの要約
表 18-11	TRIGger コマンドの要約

### 総合注記

- ・ 任意文字セットを表記するために、ブラケット ([]) を使用します。このような任意文字を、プログラムメッセージの中に含める必要はありません。プログラムメッセージの中では、ブラケットを使用しないでください。
- ・ アングルブラケット (<>) を使用して、パラメータタイプを表示します。プログラムメッセージの中では、アングルブラケットを使用しないでください。
- ・ ブールパラメータ (<b>) を使用して、計測器動作を使用可能または解除状態にします。1 または ON は動作を使用可能に、0 または OFF は解除状態にします。
- ・ 大文字は、コマンド語のショートフォームバージョンを示します。
- ・ デフォルトパラメーター記載したパラメータは、特に示す場合を除き、\*RST と:SYSTem:PRESet をデフォルトとします。パラメータに関する注記は、それぞれの表の末尾に記載してあります。
- ・ SCPI-チェックマーク (✓) は、そのコマンドとコマンドのパラメータが SCPI で確認されたことを示します。マークのないコマンドは、SCPI のコマンドですが SCPI 標準コマンドセットには準拠しません。これは SCPI コンソーシアムによる承認コマンドではありません。1 個以上の非 SCPI パラメータを使用するコマンドは、注記によって説明することを SCPI は確認しています。
- ・ ソースメモリーチェックマーク (✓) は、:SOURce[1]:MEMory:SAVE コマンドを使って、指定したコマンドに関連するパラメータを 100 カ所のメモリロケーションのどれにでも保管できることを示します。

表 18-1  
CALCulate コマンドの要約

コマンド	内容	デフォルト パラメータ	SCPI	ソース メモリ
:CALCulate[1]	CALC1 を制御するサブシステム		✓	
:MATH	数式を設定し制御するためのパス		✓	
[:EXPRession] <form>	標準数式演算子の記号を使用して数式を定義せよ。		✓	
[:EXPRession]?	数式を照会せよ。		✓	
:CATalog?	数式名のリストを照会せよ。		✓	
:NAME <name>	新しいユーザ定義式の名称を作れ。	Power	✓	✓
:NAME?	作った名称を照会せよ。		✓	
[:DEFine] <form>	:EXPRession <form> コマンドと同じ。		✓	
:DELete	ユーザ定義式を削除するためのパス		✓	
[:SElected] <name>	指定された式を削除せよ。		✓	
:ALL	すべてのユーザ定義式を削除せよ。		✓	
:UNITs <name>	数式の単位名称を定義せよ (3 個の ASCII 文字)。	"W "		✓
:UNITs?	数式の単位名称を照会せよ。			
:STATe <b>	数式を有効または解除状態にせよ。	OFF	✓	✓
:STATe?	数式の状態を照会せよ。		✓	
:DATA	CALC1 データへのパス。		✓	
:LATest?	最新の演算結果だけを戻せ。		✓	
:DATA?	数式の結果を読みとれ。		✓	
:CALCulate2	CALC2 を制御するためのサブシステム		✓	
:FEED <name>	入力パスを選択せよ;CALCulate[1], CURRent, VOLTage, or RESistance.	VOLT	✓	✓
:FEED?	CALC2 フィードを照会せよ。		✓	
:NULL	REL を設定、制御するためのパス			
:OFFSet <NRf>	REL の値を指定せよ;-9.9e37 から 9.9e37	0		✓
:OFFSet?	REL の値を照会せよ。			
:STATe <b>	REL を有効または解除状態にせよ。	OFF		✓
:STATe?	REL の状態を照会せよ。			
:ACquire	自動的に REL 値を取得せよ。			
:DATA	CALC2 データへのパス		✓	
:LATest?	最新の REL または LIMIT の結果だけを戻せ。			
:DATA?	CALC2 の演算結果を読み取れ。		✓	
:LIMit[1]	リミット 1 試験を制御するためのパス		✓	
:COMpliance	リミット 1 試験を設定せよ。			
:FAIL <name>	「不合格」条件を指定せよ;コンプライアンスの中 (IN) か、外 (OUT) か。	IN		✓
:FAIL?	「不合格」条件を照会せよ。			
:SOURce2 <NRf> I <NDN>	出力「不合格」パターンを指定せよ (0 から 7 は 3 ビット、0 から 15 は 4 ビット)	15		✓
:SOURce2?	「不合格」ビットパターンを照会せよ。			
:STATe <b>	リミット 1 試験を有効または解除状態にせよ。	OFF	✓	✓
:STATe?	リミット 1 試験の状態を照会せよ。		✓	
:FAIL?	リミット 1 試験の結果を戻せ;0 (合格) または 1 (不合格)		✓	

表 18-1  
CALCulate コマンドの要約(続き)

コマンド	内容	デフォルト パラメータ	SCPI	ソース メモリ
<b>:CALCulate2</b>				
<b>:LIMit2</b>	リミット2 試験を制御するためのパス		✓	
<b>:UPPer</b>	上限を設定せよ。		✓	
<b>[[:DATA] &lt;n&gt;</b>	上限を指定せよ; -9.999999e20 から 9.999999e20	1	✓	✓
<b>[[:DATA]?]</b>	上限を照会せよ。		✓	
<b>:SOURce2 &lt;NRf&gt;</b>	グレーディングモードの出力「不合格」パターンを指定せよ (0 から 7 は 3 ビット、0 から 15 は 4 ビット)。	15		✓
<b>  &lt;NDN&gt;</b>				
<b>:SOURce2?</b>	「不合格」ビットパターンを照会せよ。			
<b>:LOWer</b>	下限を設定せよ。		✓	
<b>[[:DATA] &lt;n&gt;</b>	下限を指定せよ; -9.999999e20 から 9.999999e20	-1	✓	✓
<b>[[:DATA]?]</b>	下限を照会せよ。		✓	
<b>:SOURce2 &lt;NRf&gt;</b>	グレーディングモードの出力「不合格」パターンを指定せよ (0 から 7 は 3 ビット、0 から 15 は 4 ビット)。	15		✓
<b>  &lt;NDN&gt;</b>				
<b>:SOURce2?</b>	「不合格」ビットパターンを照会せよ。			
<b>:PASS</b>	ソーティングモードの「合格」パターンを指定するパス			
<b>:SOURce2 &lt;NRf&gt;</b>	出力「合格」パターンを指定せよ (0 から 7 は 3 ビット、0 から 15 は 4 ビット)	15	✓	✓
<b>  &lt;NDN&gt;</b>				
<b>:SOURce2?</b>	「合格」ビットパターンを照会せよ。			
<b>:STATe &lt;b&gt;</b>	リミット2 試験を有効または解除状態にせよ。	OFF	✓	✓
<b>:STATe?</b>	リミット2 試験の状態を照会せよ。		✓	
<b>:FAIL?</b>	リミット2 試験の結果を戻せ; 0 (合格) または 1 (不合格)		✓	
<b>:LIMit3</b>	リミット3 試験を制御するためのパス		✓	
<b>:UPPer</b>	上限を設定せよ。		✓	
<b>[[:DATA] &lt;n&gt;</b>	上限を指定せよ; -9.999999e20 から 9.999999e20	1	✓	✓
<b>[[:DATA]?]</b>	上限を照会せよ。		✓	
<b>:SOURce2 &lt;NRf&gt;</b>	出力「不合格」パターンを指定せよ (0 から 7 は 3 ビット、0 から 15 は 4 ビット)	15		✓
<b>  &lt;NDN&gt;</b>				
<b>:SOURce2?</b>	「不合格」ビットパターンを照会せよ。			
<b>:LOWer</b>	下限を設定せよ。		✓	
<b>[[:DATA] &lt;n&gt;</b>	下限を指定せよ; -9.999999e20 から 9.999999e20	-1	✓	✓
<b>[[:DATA]?]</b>	下限を照会せよ。		✓	
<b>:SOURce2 &lt;NRf&gt;</b>	グレーディングモードの出力「不合格」パターンを指定せよ (0 から 7 は 3 ビット、0 から 15 は 4 ビット)。	15		✓
<b>  &lt;NDN&gt;</b>				
<b>:SOURce2?</b>	「不合格」ビットパターンを照会せよ。			
<b>:PASS</b>	ソーティングモードの「合格」パターンを指定せよ。			
<b>:SOURce2 &lt;NRf&gt;</b>	出力「合格」パターンを指定せよ (0 から 7 は 3 ビット、0 から 15 は 4 ビット)。	15	✓	✓
<b>  &lt;NDN&gt;</b>				
<b>:SOURce2?</b>	「合格」ビットパターンを照会せよ。			
<b>:STATe &lt;b&gt;</b>	リミット3 試験を有効または解除状態にせよ。	OFF	✓	✓
<b>:STATe?</b>	リミット3 試験の状態を照会せよ。		✓	
<b>:FAIL?</b>	リミット3 試験の結果を戻せ; 0 (合格) または 1 (不合格)		✓	

表 18-1

CALCulate コマンドの要約(続き)

コマンド	内容	デフォルト パラメータ	SCPI	ソース メモリ
:CALCulate2				
:LIMit4	リミット4試験のパス(コンタクトチェックオプションのみ)			
:SOURce2 <NRf>   <NDN>	出力「不合格」パターンを指定せよ (0 から 7 は 3 ビット、0 から 15 は 4 ビット)。	15		
:SOURce2?	「不合格」ビットパターンに照会せよ。			
:STATe <b>	リミット4試験を有効または無効にせよ。	OFF	✓	✓
:STATe?	リミット4試験の状態を照会せよ。		✓	
:FAIL?	リミット4試験の結果を戻せ。0 (合格) または 1 (不合格)		✓	
:LIMit5...12	リミット5からリミット12試験を制御するための パス (注記参照)		✓	
:UPPer [:DATA] <n>	上限を設定せよ。 上限を指定せよ (-9.999999e20 から 9.999999e20)	1	✓	✓
[:DATA]?	上限を照会せよ。		✓	
:SOURce2 <NRf>   <NDN>	グレーディングモードの出力「不合格」パターンを指定せよ (0 から 7 は 3 ビット、0 から 15 は 4 ビット)。	15		✓
:SOURce2?	「不合格」ビットパターンに照会せよ。		✓	
:LOWer [:DATA] <n>	下限を設定せよ。 下限を指定せよ (-9.999999e20 から 9.999999e20)	-1	✓	✓
[:DATA]?	下限を照会せよ。		✓	
:SOURce2 <NRf>   <NDN>	グレーディングモードの出力「不合格」パターンを指定せよ (0 から 7 は 3 ビット、0 から 15 は 4 ビット)。	15		✓
:SOURce2?	「不合格」ビットパターンに照会せよ。			
:PASS	ソーティングモードの「合格」パターンを指定するパス			
:SOURce2 <NRf>   <NDN>	出力「合格」パターンを指定せよ (0 から 7 は 3 ビット、0 から 15 は 4 ビット)。	15	✓	✓
:SOURce2?	「合格」ビットパターンに照会せよ。			
:STATe <b>	リミット5試験を有効または無効にせよ。	OFF	✓	✓
:STATe?	リミット5試験の結果を戻せ。0 (合格) または 1 (不合格)		✓	
:FAIL?			✓	

注記 LIMit5 から LIMit12 までを使い、リミット5からリミット12までの試験を制御してください。  
たとえば、LIM5:STAT ON を送りリミット5を有効にしてください。LIM10:FAIL?を送り、リ  
ミット10の結果を戻してください。

\* 戻された値のフォーマット (ASCII、16 進、8 進、または 2 進) は FORMat:SOURce2<name> を使って設定します。

表 18-2  
DISPlay コマンドの要約

コマンド	内容	デフォルト パラメータ	SCPI
<b>:DISPlay</b>	全面パネルディスプレイをオン状態に、またはオフ状態に入れよ。	(注記 1)	✓
<b>:ENABle &lt;b&gt;</b>	ディスプレイの状態を照会せよ。		✓
<b>:ENABle?</b>	ソース・メジャーディスプレイ状態に戻せ。		
<b>:CNDisplay</b>	上部ディスプレイへのメッセージを探すためのパス		✓
<b>[:WINDow[1]]</b>	ユーザテストメッセージを制御せよ。	(注記 2)	✓
<b>:TEXT</b>	ASCII メッセージ "a" を定義せよ (20 文字まで)。		✓
<b>:DATA &lt;a&gt;</b>	テキストメッセージを照会せよ。		✓
<b>:DATA?</b>	メッセージモードを有効、または解除状態にせよ。	(注記 3)	✓
<b>:STATe &lt;b&gt;</b>	テキストメッセージの状態を照会せよ。		✓
<b>:STATe?</b>	ディスプレイトップ部分のデータを照会せよ。		
<b>:DATA?</b>	メッセージ文字の属性を照会せよ;明滅 (1) または明滅なし (0)		
<b>:ATTRibutes?</b>	下部ディスプレイへのメッセージを探すためのパス		✓
<b>:WINDow2</b>	ユーザテストメッセージを制御せよ。	(注記 2)	✓
<b>:TEXT</b>	ASCII メッセージ "a" を定義せよ (32 文字まで)。		✓
<b>:DATA &lt;a&gt;</b>	テキストメッセージを照会せよ。		✓
<b>:DATA?</b>	メッセージモードを有効、または解除状態にせよ。	(注記 3)	✓
<b>:STATe &lt;b&gt;</b>	テキストメッセージの状態を照会せよ。		✓
<b>:STATe?</b>	ディスプレイ下部のデータを照会せよ。		✓
<b>:DATA?</b>	メッセージ文字の属性を照会せよ;明滅 (1) または明滅なし (0)		
<b>:ATTRibutes?</b>	ディスプレイの分解能を指定せよ;4 から 7	6	
<b>:DiGits &lt;n&gt;</b>	ディスプレイ分解能を照会せよ。		
<b>:DiGits?</b>			

#### 注記

1. \*RST と :SYSTem:PRESet は、ディスプレイ回路には影響を与えません。電力サイクリングを行うと、ディスプレイ回路は有効 (ON) になります。
2. \*RST と :SYSTem:PRESet は、ユーザ定義メッセージには影響を与えません。電力サイクリングを行うと、すべてのユーザ定義メッセージは取り消されます。
3. \*RST と :SYSTem:PRESet は、メッセージモードには影響を与えません。電力サイクリングを行うと、メッセージモードは解除状態 (OFF) になります。

表 18-3  
FORMat コマンドの要約

コマンド	内容	デフォルト パラメータ	SCPI
:FORMat			
:SREGister <name>	ステータスイベントレジスタを読み取るためのデータフォーマットを選択せよ (ASCIi、HEXadecimal、OCTal、または BINary)。	ASCIi	✓
:SREGister?	ステータスイベントレジスタを読み取るためのフォーマットを照会せよ。		
[[:DATA] <type>[<,length>]	データフォーマットを指定せよ;ASCIi、REAL、32 または SREAL	ASCIi	✓
[[:DATA]?	データフォーマットを照会せよ。		✓
:BORDER <name>	バイト順序を指定せよ;NORMAL または SWAPped	(注記参照)	✓
:BORDER?	バイト順序を照会せよ。		✓
:ELEMENTs			
[[:SENSe[1]] <item list>	データ要素を指定せよ (VOLTage、CURRent、RESistance、Time、STATus)。	ALL	
[[:SENSe[1]]?	データフォーマット要素を照会せよ。		
:CALCulate <item list>	CALC データ要素を指定せよ (CALC、TIME、または STATus)。	CALC	
:CALCulate?	CALC データ要素を照会せよ。		
:SOURce2 <name>	SOURce2 データフォーマットを指定せよ (ASCIi、HEXadecimal、OCTal、または BINary)。	ASCIi	
:SOURce2?	SOURce2 データフォーマットを照会せよ。		

注記 バイト順序—\*RST のデフォルトは NORMAL です。:SYSTEM:PRESet のデフォルトは SWAPped です。

表 18-4  
OUTPut コマンドの要約

コマンド	内容	デフォルト パラメータ	SCPI	ソース メモリ
:OUTPut[1]			✓	
:STATe <b>	[1]ソースをオン状態またはオフ状態にせよ。	OFF	✓	
:STATe?	ソースの状態を照会せよ。		✓	
:INTerlock	インターロックを制御するためのパス			
:STATe <b>	インターロックを有効または解除状態にせよ。	OFF		
:STATe?	インターロックの状態を照会せよ。			
:TRIPped?	インターロックトリップ?;1 (yes) または 0 (no)			
:SMODE <name>	出力モードを選択せよ;HIMPedance、NORMAL、または ZERO	NORMAL		✓
:SMODE?	出力オフモードを照会せよ。			

1.2430 型パルスモードの場合は、出力オフモードは常に NORMAL です。このコマンドは無効です (エラー +831)。







表18-6

SENSe コマンドの要約 (続き)

コマンド	内容	デフォルト パラメータ	SCPI	ソース メモリ
[[:SENSe[1]]]				
:RESistance	抵抗を設定するためのパス		√	
:MODE <name>	抵抗モードを選択せよ;MANual または AUTO	AUTO		√
:MODE?	抵抗モードを照会せよ。			
:OCOMpensated <b>	オフセット補償抵抗測定を有効または無効にせよ。	OFF	√	√
:OCOMpensated?	オフセット補償抵抗測定の状態を照会せよ。		√	
:RANGe	測定レンジを設定せよ。		√	
[:UPPer]	期待抵抗読み取り値を指定することにより、	2.1e5	√	√ <sup>7</sup>
<n> UPIDOWN	レンジを選択せよ。 <sup>2</sup>		√	
[:UPPer]?	レンジを照会せよ。			
:AUTO <b>	オートレンジを有効または解除状態にせよ。 <sup>3</sup>	ON	√	√
:AUTO?	オートレンジを照会せよ。		√	
ULIMit <n>	オートレンジ上限を設定せよ。 <sup>2</sup>	1.0	√	√
ULIMit?	オートレンジ上限を照会せよ。		√	
LLIMit <n>	オートレンジ下限を設定せよ。 <sup>2</sup>		√	
LLIMit?	オートレンジ下限を照会せよ。	2.1e8	√	
:NPLCycles <n>	積分速度 (単位:ラインサイクル) を指定せよ; 0.01 から 10。 <sup>4</sup>			
:NPLCycles?	積分速度を照会せよ。		√	
:AVERage	デジタルフィルタを設定、制御するためのパスフィ	REPeat	√	
:TCONtrol <name>	ルタの種類を指定せよ;MOVing または REPeat。 <sup>5</sup>		√	√
:TCONtrol?	フィルタの種類を照会せよ。		√	
:COUNT <n>	フィルタカウントを指定せよ;1 から 100 まで。 <sup>5</sup>	10	√	√
:COUNT?	フィルタカウントを照会せよ。		√	
[:STATe] <b>	フィルタを有効または解除状態にせよ。 <sup>6</sup>	OFF	√	√
[:STATe]?	フィルタの状態を照会せよ。		√	

1. 2430 型パルスモードの場合は、このコマンドを使ってオフセット補償抵抗測定を有効にすることはできません。このコマンドは無効です (エラー 831)。
2. このコマンドのパラメータ値には、詳細なコマンド参照情報が用意されています。SCPI コマンドの詳細な情報は、この部の最後のコマンド要約表の次に記載してあります。
3. 2430 型パルスモードの場合は、オートレンジは無効になります (エラー +831)。
4. 2430 型パルスモードの場合は、<n>=0.004 から 0.10
5. 2430 型パルスモードの場合は、フィルタリングは使用しません。このコマンドは無視されます。
6. 2430 型パルスモードの場合は、フィルタを有効にすることはできません。このコマンドは無効です (エラー +831)。
7. 自動抵抗測定が有効な場合

1. 2430 型パルスモードの場合は、オートクリアは常に有効になっています。このコマンドは無視されます。
2. 2430 型パルスモードの場合は、ソース遅延は使用しません。このコマンドは無視されます。
3. このコマンドのパラメータ値には、詳細なコマンド参照情報が用意されています。SCPI コマンドの詳細な情報は、この部の最後のコマンド要約表の次に記載してあります。
4. ソース V がアクティブである場合

表 18-7

SOURce コマンドの要約 (続き)

コマンド	内容	デフォルト パラメータ	SCPI	ソース メモリ
:SOURce[1] :CURRent				
:START <n>	I スイープのための開始レベルを指定せよ。 <sup>1</sup>	0	✓	
:START?	電流スイープのための開始レベルを照会せよ。		✓	
:STOP <n>	I スイープのための停止レベルを指定せよ。 <sup>1</sup>	0	✓	
:STOP?	電流スイープのための停止レベルを照会せよ。		✓	
:STEP <n>	I スイープのためのステップ値を指定せよ。 <sup>1</sup>	0		
:STEP?	電圧スイープのためのステップ値を照会せよ。			
:SPAN <n>	スパンを指定せよ。 <sup>1</sup>	0	✓	
:SPAN?	スパンを照会せよ。		✓	
:CENTer <n>	中心点を指定せよ。 <sup>1</sup>	0	✓	
:CENTer?	中心点を照会せよ。		✓	
:VOLTage	V ソースを設定するためのパス		✓	
:MODE <n>	V ソースモードを選択せよ;FIXed、SWEep、 または LIST	FIXed	✓	
:MODE?	V ソースモードを照会せよ。		✓	
:RANGe <n> UP DOWN  AUTO <b>	固定 V ソースレンジを選択せよ。 <sup>1</sup>	21	✓	√ <sup>3</sup>
:AUTO?	オートレンジ設定を有効または解除状態にせよ。	ON	✓	√ <sup>3</sup>
:RANGe?	オートレンジ設定の状態を照会せよ。		✓	
:LEVel	V ソースレンジ設定を照会せよ。		✓	
[:IMMediate]	V ソースレベル (単位 V) を設定せよ。		✓	
[:AMPLitude] <n>	指定されたレベルを直ちに(immediate)設定せよ。		✓	
[:AMPLitude]?	電圧レベルを指定せよ。 <sup>1</sup>	0	✓	√ <sup>3</sup>
:TRIGgered	電圧レベルを照会せよ。		✓	
[:AMPLitude] <n>	トリガするときのレベルを設定せよ。		✓	
[:AMPLitude]?	電圧レベルを指定せよ。 <sup>1</sup>	0	✓	
:SFACtor <n>	電圧レベルを照会せよ。		✓	
:STATE <b>	電圧スケール係数を設定せよ (-999.9999e+18 から +999.9999e+18)。	0		√ <sup>3</sup>
:STATE?	電圧スケール係数を有効/無効にせよ。	OFF		√ <sup>3</sup>
:SFACtor?	電圧スケール係数の状態を照会せよ。			
:PROTection	電圧スケール係数を照会せよ。		✓	
[:LEVel] <NRf>	出力電圧を制限するためのパス		✓	
[:LEVel]?	電圧リミットレベルを指定せよ。 <sup>1</sup>	(Note 2)	✓	
:TRIPped?	電圧リミットを照会せよ。		✓	
	電圧リミット検出;1 (yes) ,0 (no)		✓	

## 注記

- このコマンドのパラメータ値には、詳細なコマンド参照情報が用意されています。SCPI コマンドの詳細な情報は、この部の最後のコマンド要約表の次に記載してあります。
- SYSTEM:PRESet のデフォルトは 40 です。\*RST のデフォルトは NONE です。
- ソース V がアクティブである場合

表 18-7

SOURce コマンドの要約 (続き)

コマンド	内容	デフォルト パラメータ	SCPI	ソース メモリ
:SOURce{1}				
:VOLTage				
:STARt <n>	V スイープの開始レベルを指定せよ。 <sup>1</sup>	0	✓	
:STARt?	電圧スイープの開始レベルを照会せよ。		✓	
:STOP <n>	V スイープの停止レベルを指定せよ。 <sup>1</sup>	0	✓	
:STOP?	電圧スイープの停止レベルを照会せよ。		✓	
:STEP <n>	V スイープのステップ値を指定せよ。 <sup>1</sup>	0		
:STEP?	電圧スイープのステップ値を照会せよ。			
:SPAN <n>	スパンを指定せよ。 <sup>1</sup>	0	✓	
:SPAN?	スパンを照会せよ。		✓	
:CENTer <n>	中心点を指定せよ。 <sup>1</sup>	0	✓	
:CENTer?	中心点を照会せよ。		✓	
:SWEep	SWEep ソースモードを設定せよ。		✓	
:SOAK <NRf>	第1スイープ点ソーク時間を設定せよ (0.00000 から 9999.999s)。	0.00000	✓	
:SOAK?	ソーク時間を照会せよ。		✓	
:SPACing <name>	スイープ間隔の種類を選択せよ;LINear または LOGarithmic	LINear	✓	
:SPACing?	スイープ間隔を照会せよ。		✓	
:POINts <n>	スイープ点の数を指定せよ;2 から 2500 まで	2500		
:POINts?	スイープに含まれる点の数を照会せよ。		✓	
:DIRection <name>	開始点から終止点 (UP) へ、または終止点から 開始点 (DOWN) へスイープせよ。	UP		
:DIRection?	スイープ方向を照会せよ。			
:RANGing <name>	ソースレンジ設定モード (BEST、AUTO、 または FIXed) を選択せよ。	BEST	✓	
:RANGing?	ソースレンジ設定モードを照会せよ。		✓	
:LIST	LIST ソースモードを設定せよ。		✓	
:CURRent <NRf>	I ソース値のリストを作成せよ。 <sup>1</sup>			
:APPend <NRf>	I ソース値をリストの末尾に加えよ。		✓	
:POINts?	リストの中のソース値の数を照会せよ。			
:CURRent?	I ソースリストを照会せよ。		✓	
:VOLTage <NRf>	V ソース値のリストを作成せよ。 <sup>1</sup>			
:APPend <NRf>	V ソース値をリストの末尾に加えよ。			
:POINts?	リストの中のソース値の数を照会せよ。			
:VOLTage?	V ソースリストを照会せよ。			

1. このコマンドのパラメータ値には、詳細なコマンド参照情報が用意されています。SCPI コマンドの詳細な情報は、この部の最後のコマンド要約表の次に記載してあります。



表18-8  
STATus コマンドの要約

コマンド	内容	デフォルト パラメータ	SCPI
:STATus		(注記 1)	✓
:MEASurement	制御測定イベントレジスタ		
[:EVENT]?	イベントレジスタを読み取れ。 <sup>6</sup>	(注記 2)	✓
:ENABle <NDN> or <NRf>	イネーブルレジスタをプログラムせよ。	(注記 3)	✓
:ENABle?	イネーブルレジスタを読み取れ。 <sup>6</sup>		✓
:CONDition?	条件レジスタを読み取れ。 <sup>6</sup>		✓
:OPERation	制御動作ステータスレジスタ		✓
[:EVENT]?	イベントレジスタを読み取れ。 <sup>6</sup>	(注記 2)	✓
:ENABle <NDN> or <NRf>	イネーブルレジスタをプログラムせよ。	(注記 3)	✓
:ENABle?	イネーブルレジスタを読み取れ。 <sup>6</sup>		✓
:CONDition?	条件レジスタを読み取れ。 <sup>6</sup>		✓
:QUESTionable	制御疑問ステータスレジスタ		✓
[:EVENT]?	イベントレジスタを読み取れ。 <sup>6</sup>	(注記 2)	✓
:ENABle <NDN> or <NRf>	イネーブルレジスタをプログラムせよ。	(注記 3)	✓
:ENABle?	イネーブルレジスタを読み取れ。 <sup>6</sup>		✓
:CONDition?	条件レジスタを読み取れ。 <sup>6</sup>		✓
:PRESet	ステータスレジスタをデフォルト状態に戻せ。		✓
:QUEue	エラー待ち行列にアクセスするためのバス		
[:NEXT]?	最新のエラーメッセージを読み取れ。	(注記 4)	✓
:ENABle <list>	エラー待ち行列に対してエラーメッセージとステータスメッセージを指定せよ。	(注記 5)	✓
:ENABle?	有効になったメッセージを読み取れ。		✓
:DISable <list>	エラー待ち行列に置かないメッセージを指定せよ。	(注記 5)	
:DISable?	解除状態になったメッセージを読み取れ。		
:CLEar	すべてのメッセージをエラー待ち行列からクリアせよ。		

#### 注記

- このサブシステムのコマンドは、\*RST と:SYSTem:PRESet の影響を受けません。電源再投入、\*CLS、:STATus:PRESet の影響は、次の注記で説明します。
- イベントレジスタ電源投入と \*CLS は、すべてのビットをクリアします。:STATus:PRESet は影響を与えません。
- イネーブルレジスタ電源投入と:STATus:PRESet は、すべてのビットをクリアします。\*CLS は影響を与えません。SCPI 1995.0 (非 10 進数) フォーマット (または) を受け入れます。
- エラー待ち行列電源投入と \*CLS は、すべてのビットをクリアします。
- エラー待ち行列メッセージ電源投入は、メッセージリストをクリアします。\*CLS と:STATus:PRESet は影響を与えません。
- レジスタ照会コマンドー応答メッセージのフォーマット (ASCII、16 進、8 進、または 2 進) は、現在のデータフォーマットが選択されているかによります (:FORMat\*SREGister コマンド参照)。



表 18-9  
SYSTem コマンドの要約

コマンド	内容	デフォルト パラメータ	SCPI	ソース メモリ
:SYSTem	SYSTem:PRESet デフォルトに戻れ。		✓	
:PRESet	電源投入セットアップを選択せよ;RST、PRESet、またはSAV 0-4			
:POSetup <name>	電源投入セットアップを照会せよ。		✓	
:POSetup?	SCPI の改訂レベルを照会せよ。		✓	
:VERsion?	エラー待ち行列にあるメッセージを読み取るためのパス (注記 1)			
:ERRor	最も古いエラーを戻し、クリアせよ (コードとメッセージ)。		✓	
[:NEXT]?	すべてのエラーを戻し、クリアせよ (コードとメッセージ)。			
:ALL?	エラー数を戻せ。			
:COUNT?	エラーコード番号のみを戻すパス			
:CODE	最も古いエラーを戻し、クリアせよ (コードのみ)。			
[:NEXT]?	すべてのエラーを戻し、クリアせよ (コードのみ)。			
ALL?	メッセージをエラー待ち行列からクリアせよ。			
:CLear	リモートセンシングを有効または解除状態にせよ。	OFF		✓
:RSENse <b>	リモートセンシングの状態を照会せよ。			
:RSENse?	キー・プレスをシミュレートせよ (1から31まで)。図 18-3 を参照してください。		✓	
:KEY <n>	最後に「押された」キーを照会せよ。		✓	
:KEY?	ガードの種類を選択せよ (OHMS または CABLE)。	CABLE		
:GUARd <name>	ガードの種類を照会せよ。			
:GUARd?	ビーパを制御せよ。			
:BEEPer	指定周波数 (65 から 2e6Hz) で指定継続期間 (0 から 7.9 秒) にわたって鳴らせ。			
[:IMMediate]	ビーパを有効または解除状態にせよ。	ON	✓	
<freq, time>	ビーパの状態を照会せよ。		✓	
:STATe <b>	オートゼロを制御せよ。		✓	
:STATe?	オートゼロを有効または解除状態にせよ。	ON		✓
:AZERo	オートゼロの状態を照会せよ。		✓	
[:STATe] <b>	ライン周波数を選択せよ;50 または 60 (Hz)			
[:STATe]?	自動周波数を有効または解除状態にせよ。	(Note 3)		
:LFRequency <freq>	自動周波数の状態を照会せよ。			
:AUTO <b>	ライン周波数を照会せよ。			
:AUTO?	タイムスタンプ			
:LFRequency?	タイムスタンプをゼロ秒にリセットせよ。			
:TIME	アイドル状態から抜け出るときタイムスタンプを有効または解除状態にせよ	OFF		
:RESet	タイムスタンプを照会せよ。			
:AUTO <b>	メモリを初期化せよ。			
:TIME?	バッテリーバックアップ RAM を初期化せよ。		✓	
:MEMory	ユニットをリモート状態から外せ (RS-232 のみ)。		✓	
:INITialize	ユニットをリモート状態に入れよ (RS-232 のみ)。			
:LOCal	ローカルロックアウトを有効または解除状態にせよ (RS-232 のみ)。			
:REMote	オートレンジをコンプライアンスモードに設定せよ (SINGLE または MULTiple)。	SINGLE		
:RWLock <b>	コンプライアンスモードに設定されたオートレンジを照会せよ。			
:RCMode <name>				
:RCMode?				

#### 注記

- エラー待ち行列のクリアー電源投入と \*CLS はエラー待ち行列をクリアします。\*RST、:SYSTem:PRESet、:STATus:PRESet は影響を与えません。
- SCPI との適合性については、:ROUTe:TERMinals コマンドを使用し、入力/出力端子を選択してください。
- 自動ライン周波数設定値は、\*RST、:SYSTem:PRESet の影響を受けません。

表 18-10  
TRACe コマンドの要約

コマンド	内容	デフォルト パラメータ	SCPI
:TRACe:DATA	TRACe または:DATA をルートコマンドとして使用せよ。	(注記参照)	✓
:DATA?	バッファの内容を読み取れ (データストア)。		✓
:CLEar	読取り値をバッファからクリアせよ。		
:FREE?	利用可能なバイトと使用中のバイトを照会せよ。		✓
:POINts <NRf>	バッファサイズを指定せよ;1 から 2500		✓
:ACTual?	バッファに格納された読取り値の数を照会せよ。		
:POINts?	バッファサイズを照会せよ。		✓
:FEED <name>	読取り値のソースを選択せよ;SENSe[1]、CALCulate[1]、 または CALCulate2		✓
:CONTRol <name>	バッファ制御モードを指定せよ;NEVER または NEXT		✓
:CONTRol?	バッファ制御モードを照会せよ。		✓
:TSTamp	タイムスタンプフォーマットを設定するためのパス		
:FORMat <name>	フォーマットを選択せよ;ABSolute または Δ		
:FORMat?	タイムスタンプフォーマットを照会せよ。		

注記 :SYSTem:PRESet と \*RST は、このサブシステムのコマンドには影響を与えません。

表 18-11  
Trigger コマンドの要約

コマンド	内容	デフォルト パラメータ	SCPI
:INITiate[:IMMediate]	:INITiate[:IMMediate]		✓
:ABORt	1 ソース - メジャーサイクルを開始せよ。		✓
:ARM	トリガシステムをリセットせよ。アイドル状態に移行します。		✓
[:SEQuence[1]]	アームレイヤーをプログラムするためのパス		✓
[:LAYer[1]]			✓
:COUNt <n>	アームカウントを指定せよ;1 から 2500	1	✓
:COUNt?	アームカウントを照会せよ。(INFinite = +9.9e37)	IMMediate	✓
:SOURce <name>	制御ソースを指定せよ;IMMediate、TIMer、MANual、BUS、TLINK、NSTESTまたはPSTest)。		✓
:SOURce?	制御ソースを照会せよ。	0.1	✓
:TIMer <n>	タイマ間隔を秒で設定せよ;0.001 から 99999.99		✓
:TIMer?	タイマ間隔を照会せよ。		✓
[:TCONfigure]			✓
:DIRection <name>	バイパスを有効 (SOURce) または解除状態 (ACCeptor) にせよ。	ACCeptor	✓
:DIRection?	バイパスの状態を照会せよ。		✓
[:ASYNchronous]	出力トリガを設定せよ。		✓
:ILINe <n>	入力トリガラインを選択せよ;1、2、3、または4	1	
:ILINe?	入力トリガラインを照会せよ。		
:OLINe <n>	出力トリガラインを選択せよ;1、2、3、または4	2	
:OLINe?	出力トリガラインを照会せよ。		
:OUTPut <name>	SWEp のあとトリガを出力せよ。またはまったく出力するな。(NONE)	NONE	
:OUTPut?	アーム出力トリガのステータスを照会せよ。		
:TRIGger	トリガレイヤーをプログラムするためのパス		✓
:CLEar	保留中の入力トリガがあれば、直ちに(immediate)クリアせよ。		✓
[:SEQuence[1]]			✓
:COUNt <n>	トリガカウントを指定せよ;1 から 2500	1	✓
:COUNt?	トリガカウントを照会せよ。		✓
:DELay <n>	トリガディレイを指定せよ;0 から 999.9999 (sec) 1	0	✓
:DELay?	ソースディレイを照会せよ。		✓
:SOURce <name>	制御ソースを指定せよ;IMMediate または TLINK	IMMediate	✓
:SOURce?	制御ソースを照会せよ。		✓

1. 2430 型パルスモードの場合は、トリガディレイは使用しません。このコマンドは無視されます。

表 18-11

Trigger コマンドの要約 (続き)

コマンド	内容	デフォルト パラメータ	SCPI
<b>:TRIGger</b> [:SEQuence[1]] [:TCONfigure] :DIRection <name> :DIRection [:ASYNchronous] :ILINe <n> :ILINe? :INPut <event list>  :INPut? :OLINe <n> :OLINe? :OUTPut <event list>  :OUTPut?	バイパスを有効 (SOURce) または解除状態 (ACceptor) にせよ。 バイパスの状態を照会せよ。 出力トリガを設定せよ。 入力トリガラインを選択せよ;1、2、3、または4 入力トリガラインを照会せよ。 入力イベントディテクタを有効にせよ (SOURce、DELay、SENSe、または NONE) <sup>1</sup> 有効な入力イベントディテクタを照会せよ。 出力トリガラインを選択せよ;1、2、3、または4 出力トリガラインを照会せよ。 SOURce、DELay、または SENSe のあと、トリガを出力 (NONE) せよ。またはまったく出力するな。 <sup>2</sup> 出力トリガ発生時期を照会せよ。	1     NONE  2   NONE	   ✓ ✓ ✓ ✓

- 2430 型パルスモードの場合は、SOURce または NONE パラメータのみが有効です。DELay と SENSe パラメータは無視されます。
- 2430 型パルスモードの場合は、SENSe または NONE パラメータのみが有効です。DELay と SOURce パラメータは無視されます。

## Calculate サブシステム

Calculate サブシステムには3個のサブシステムがあります。CALC1 サブシステムは数式に、CALC2 はリミット試験に使用され、CALC3 はバッファに格納された読取り値に関する統計的データを提供します。これらのサブシステムのコマンドを、表 5-2 に要約します。

### CALCulate[1]

### 数式名称を選択（作成）せよ

**注記** 偏差率（%DEV）はカタログに内蔵数式として含まれていますが、フロントパネルからのみ使用可能です。ただし、偏差率はユーザー定義数式に加えて、リモートオペレーションで使用することができます（「プログラムの例」）。

### Select (create) math expression name（数式名を選択（作成）する）

:CATalog?

:CALCulate[1]:MATH[:EXpression]:CATalog

数式名称のリストを照会せよ

**説明** この照会コマンドを使用して、利用可能な数式名称のリストを作成します。このリストに含まれるのは、内蔵した数式名称とユーザが定義した数式名称です。内蔵数式の名称は次のとおりです。

"POWER","OFFCOMP Ω ","VOLTCOEFF","VARALPHA"

したがって、:CATalog? コマンドは、ユーザが定義した数式の名称だけでなく、上記の名称を戻します。ユーザ定義数式に名称を割り当てるための:NAME を参照してください。

:NAME<name>

:CALCulate[1]:MATH[:EXpression]<name>

数式を選択せよ

パラメータ	<name> =	"POWER"	瞬時電力数式
		"OFFCOMPOHM"	オフセット補償抵抗数式
		"VOLTCOEFF"	抵抗電圧係数数式
		"VARALPHA"	バリスタアルファ数式
		"ユーザ名称"	ユーザ定義数式に割り当てた名称
			ここでユーザ名称は、ASCII 文字で設定されます（10 文字まで）。

**照会** :NAME? 選択した数式を照会せよ。

**説明** このコマンドを使用して、すでに存在する数式（内蔵またはユーザ定義による数式）を選択することができます。すでに存在する数式名称をリストするには、:CATalog? コマンドを使用します。実際の数式を読むには、:MATH? コマンドを使用します。内蔵数式（POWER を除く）は、計算の実行に2点スワイプを必要とします。「プログラムフラグメント」は、これらの数式のためのソースメータを設定する方法を示します。

新しいユーザ定義数式を作成したいときには、次のステップを順番に行ってください。

1. 必要であれば、計算結果に単位を割り当ててください (:UNITS を参照)。単位は計算のために格納されます。
2. このコマンドを使用して、数式に名称 (10 個までの ASCII 文字を使用して) を割り当ててください。
3. \*DEFine または EXPRession コマンドを使用して、数式を定義してください。新しい数式は、現在選択されている数式となります。

#### 数式エラー

- +801「不十分なベクトルデータ」—ベクトルを完全に満杯にするのに十分なデータを取得する前にアイドル状態に戻った。CALC1 の結果が形成されていません。
- +804「数式リスト満杯」—リスト (カタログ) が満杯の時に新しい数式名称を作成しようとした。
- +805「未定義数式が存在」—前の数式の名称が未定義の状態に新しい数式名称を作成しようとした。
- +806「数式が見つかりません」—名称付の数式で見つけることのできないものを削除しようとした。
- +807「この定義は許されません」—これまでに名称を付与してない数式を定義しようとした。
- +808「この数式は削除できません」—内蔵数式の一つを削除しようとした (:DELete 参照)。
- +809「ソースメモリ位置が改訂されました」—:SOURce:MEMory スイープ位置が、もはや存在しない数式を参照したときに発生します。
- +811「演算子または数字ではありません」—有効な演算子または数字を使用しなかったので、無効の数式を定義した。
- +812「括弧の不整合」—開放括弧の数字は閉鎖括弧の数字と同じでなければなりません。たとえば、CALC:MATH:EXPR(2\*sin(VOLT))はこの種のエラーを発生します。
- +813「データハンドルの数字ではありません」—無効な浮動小数点数または VOLT、CURR、RES、または TIME 以外の記号が数式に現れる。
- +814「ブラケットの不整合」—ベクトル化された数式指標のためのブラケットの使い方が適切でない。たとえば、CALC1:MATH:EXPR(VOLT[0\*CURR[0]])はこの種のエラーを発生します。
- +815「括弧が多すぎる」—検出された閉鎖括弧の数が多すぎる。たとえば、CALC1:MATH:EXPR(In(VOLT))はこの種のエラーを発生します。
- +816「数式全体がパースされてない」—ソース・メータが計算する関数を、入力数式が生成しない場合に発生します。
- +817「トークンが未知」—無効な関数名を使って数式を定義しようとした。

- 注記

- ## プログラムの例

電力：

オフセットを補償した抵抗：

*RST	
:SENS:FUNC:OFF:ALL	
:SENS:FUNC	"VOLT","CURR"
:SOUR:FUNC	VOLT or CURR
if:SOUR:FUNC VOLT then	:SOUR:VOLT:STAR <n> STOP <n>; MODE SWE
if:SOUR:FUNC CURR then	:SOUR:VOLT:STAR <n> STOP <n>; MODE SWE
:SOUR:SWE:POIN	2

```

:TRIG: COUN 2
:CALC:MATH:EXPR:NAME "OFFCOMPOHM"
:CALC:STAT ON
:OUTPUT ON
:INIT
:CALC:DATA?

抵抗電圧係数
*RST
:SENS:FUNC:ON:ALL
:SENS:RES:MODE MAN
:SOUP:FUNC:ON VOLT or CURR
if:SOUP:FUNC VOLT then :SOUP:VOLT:STAR <n>
                        STOP <n>; MODE SWE
if:SOUP:FUNC CURR then :SOUP:VOLT:STAR <n>
                        STOP <n>; MODE SWE

:SOUP:SWE:POIN 2
:TRIG: COUN 2
:CALC:MATH:EXPR:NAME "VOLTCOEF"
:CALC:STAT ON
:OUTPUT ON
:INIT
:CALC:DATA?

```

## バリスタアルファ

```

*RST
:SENS:FUNC:OFF:ALL
:SENS:FUNC:ON "VOLT,"CURR"
:SOUP:FUNC:MODE CURR
:SOUP:CURR:STAR <n>;STOP <n>;MODE SWE
:SOUP:SWE:POIN 2
:TRIG: COUN 2
:CALC:MATH:EXPR:NAME "VARALPHA"
:CALC:STAT ON
:OUTPUT ON
:INIT
:CALC:DATA?

```

## 偏差率

注記 偏差率はリモート動作の内蔵数式ではありません。このプログラムの例では、下記の百分率 (PER\_DEV) 計算が生成され、10kΩ の抵抗体を試験します。このユーザ定義数式がカタログに追加されます。

$$\text{PER\_DEV} = (\text{RES} - 10\text{k}\Omega) / 10\text{k}\Omega \times 100$$

ここで RES は DUT の実測抵抗です。  
10kΩ は基準値です。

```

*RST
:SENS:FUNC:OFF:ALL
:SENS:FUNC:ON or "RES"

```



```
:CALC:MATH:UNIT "%"
:CALC:MATH:EXPR:NAME "PER_DEV"
:CALC:MATH:EXPR((RES - 10e3) / 10em) * 100
:CALC:MATH:EXPR:NAME "PER_DEV"
:CALC:STAT ON
:OUTPUT ON
:INIT
:CALC:DATA?
```

注記 上記の:SOUR:VOLT コマンドと:SOUR:CURR コマンドで参照するパラメータ <n> は、ユーザがプログラムする実際の演算を表します。ほかのすべてのコマンドは、表示されるとおりに入力する必要があります。

```
:DELeTe[:SELeCted] <n>
:CALCulate[1]:MATH[:EXPRession]:DELeTe[:SELeCted] <name> ユーザ定義数式を削除せよ。
```

パラメータ <name> = 「ユーザ - 名称」 ユーザ定義による数式の名称

説明 このコマンドを使用して、指定ユーザ定義数式をカタログから除去（削除）します。いったん除去すれば、それ以後は、その数式を選択することはできません。:CATalog?コマンドを使用して、その数式の除去を確認することができます。

たとえば、"math 1" という名称のユーザ定義数式を削除しようとする場合には、下記のコマンドを送ることになります。

```
:DELeTe"math 1"
```

内蔵数式を削除することはできません。削除しようすると、「この数式は削除できません」というエラーメッセージ 808 が現れます。

```
:DELeTe:ALL
:CALCulate[1]:MATH[:EXPRession] すべてのユーザ定義数式を削除せよ。
```

説明 このアクションコマンドは、すべてのユーザ定義数式をカタログから除去（削除）します。内蔵数式は影響を受けません。

## Assign unit suffix（単位サフィックスの割り付け）

```
:UNITs <name>
:CALCulate[1]:MATH:UNITs <name> ユーザ定義計算の単位を指定せよ。
```

パラメータ <name> = 単一または二重引用符で囲まれた 3 個の ASCII 文字

照会 :UNITs? ユーザ定義計算の単位を照会せよ。

説明 このコマンドを使用して、ユーザ定義数式計算の単位サフィックス名を指定します。単位サフィックス名としては、3 個の ASCII 文字を使用してください。3 文字にならないときは、文字列の中のユニット名の右にスペースを追加してください。たとえば、ユニット名が "Z" であれば、次のような形で送信してください。

```
:calc:math:unit "Z"
```

ユニットの名称は、次のように一重引用符の中で囲むこともできます。  
`:calc:math:unit 'Z'`

## Define math expression (数式の定義)

`[[:EXPRession] <form> または[:DEFine] <form>`  
`:CALCulate[1]:MATH[:EXPRession] <form>` 数式を定義せよ。  
`:CALCulate[1]:MATH[:EXPRession][:DEFine] <form>` 数式を定義せよ。

パラメータ <form> = 計測器の読取り値、数値、標準数学演算子記号を使用する数式。詳細については説明の項を参照してください。

照会 `:MATH?` ユーザ定義数式を照会せよ。

説明 これら2個のコマンドのどれかを使用して、メジャー読取り値とソース読取り値、数値定数、標準数学演算子記号を使用する数式を定義してください。数式を定義したあと、その数式は、`:NAME` コマンドを使用して作った名称に割り当てられ、選択数式になります。

メジャー読取り値とソース読取り値として有効なパラメータ名称には、次のものがあります。

<code>VOLTage</code>	V メジャーまたは V ソース読取り値を使用せよ。
<code>CURRent</code>	I メジャーまたは I ソース読取り値を使用せよ。
<code>RESistance</code>	抵抗読取り値を使用せよ。
<code>TIME</code>	タイムスタンプ読取り値を使用せよ。

有効な数学演算子とその演算の内容は、次にリストするとおりです。

<code>+</code>	加算
<code>-</code>	減算
<code>*</code>	乗算
<code>/</code>	除算
<code>^</code>	指数
<code>log</code>	対数、底 10
<code>ln</code>	自然対数
<code>sin</code>	正弦
<code>cos</code>	余弦
<code>tan</code>	正接
<code>exp</code>	$e^x$

注記 `log` と `ln` の演算は、指定した数の絶対値について行います。たとえば  
`log(100)=2` であり `log(-100)=2`

数式は、下記の優先順位ルールに従って値を求めます。

1. 括弧に囲まれた数式
2. 単項演算子 (+ と -)
3. (べき乗)
4. (乗算) と /(除算)
5. + (加算) と -(減算)
6. 左から右へ

計算に使用する読取り値は、ソースメータの設定によって変わります。ソースメータがVのソースとなりIをメジャーするように設定されているときは、計算用の電圧読取り値はソース値であり、電流読取り値は電流測定値となります。逆に、ソースIメジャーV用の設定であれば、電流読取り値はソース値、電圧読取り値は電圧測定値となります。

メジャー読取り値が、ソース読取り値に優先します。したがって、ソースVメジャーV用の設定の場合は、計算用の電圧読取り値は、電圧測定値となります（プログラムされたVソース値ではありません）。逆に、ソースIメジャーI用に設定されている場合は、計算用の電流読取り値は電流測定値となります。

ソースまたはメジャーによらない読取り値を使用する計算の結果は、+9.91e37 という無効 NAN（数ではない）値となります。たとえば、ソースVメジャーVの計算で電流読取り値を使用すると、結果はNANとなります。

ソースIメジャーVの設定を使用する例

`:calc:math (volt * curr)`      電圧測定値とIソース値を使用して電力を計算せよ。

計算のための設定を行い、計算が使用可能（:STATE参照）にされたあと、ソース-メジャー動作が行われると結果がディスプレイされます。:data?コマンドを使用して、結果をコンピュータに送ってください。

### ベクトル数式

ベクトルを取り入れることにより、数式計算にどの読取り値を使用するかを選択します。プログラムされたすべてのソース-メジャー動作が完了したあと、指定したベクトルが表示する読取り値を使用して、数式計算が行われます。

ベクトル数はブラケット（[]）の中に囲まれ、0から始まります。したがって、ベクトル0がアレイの中での最初の読取り値、ベクトル1は第2の読取り値、以下同様となります。数式の中の最大のベクトル数が、ベクトルアレイのサイズを決定します。

たとえば、ソースメータが10回のソース-メジャー動作を行うようにプログラムされており、次のベクトル数式計算が使用されると想定します。

`(volt[3]-volt[9])`

上記の式が定義するのは、10個の読取り値で設定されるベクトルアレイです。ソースメータは10回のソース-メジャー動作を行うように設定されているので、計算は10SDMサイクルごとに1つの結果を出すことになります。第4の電圧読取り値（ベクトル3）と第10の電圧読取り値（ベクトル9）を使って、計算を行います。

さて、ソースメータが20回のソース-メジャー動作を行うように設定されていると想定しましょう。ベクトルサイズは10のままですから、読取り値10のアレイ2個が生成されます。個の場合、計算はそれぞれのアレイごとに1つ、合計2つの結果を出します。

第1の結果は、前と同じように、第1アレイの第4と第10の読取り値に基づくものです。第2の結果は、第14と第20の結果に基づくものです。これらは、第2アレイの第4（ベクトル3）と第10（ベクトル9）の読取り値です。

有効な計算結果を取得するためには、完全なベクトルアレイが必要です。上の例で、ソースメータの設定を変更して25回のソース-メジャー動作を行わせようとする場合、第3アレイが不完全なアレイとなります（第1アレイは10個の読取り値、第2アレイは10個の読取り値、第3アレイは5個の読取り値）。ソースメータがアイドル状態に戻ったあと、「ベクトルデータ不十分」というエラーメッセージがディスプレイされ、第3の結果はNAN (+9.91e37) となります。

不完全ベクトルアレイを避けるために、プログラムしたソース-メジャー動作回数（アームカウント×トリガカウント）がベクトルアレイサイズの倍数であることを確認してください。前記の例では、ベクトルアレイサイズは10です。したがって「不完全ベクトルデータ」エラーを避けるためには、プログラムしたソース-メジャー動作回数は、10の倍数（10、20、30、40、以下同様）となる必要があります。

オフセット補償抵抗計算用の下記のベクトル数式は、正しいシンタクスを表します。

```
:calc:math ((volt[1]-volt[0]) / (curr[1]-curr[0]))
```

#### 注記

1. ネストされた括弧を使用して、計算に埋め込まれた数式計算を強行させてください（「ベクトル数式」の例を参照）。
2. 計算式の長さは、括弧とホワイトスペースを含めて256文字までです。
3. フィルタを使用する場合、計算に使用する測定読取り値は、フィルタ処理を受けており、計算の結果ではありません。
4. ベクトル計算の場合、REPEATフィルタだけを使用することを推奨します。リピートフィルタを使用する場合には、計算にはベクトル点のフィルタ処理読取り値だけが使用されます。代わりにMOVINGフィルタを使用すると、それぞれのベクトル点は、ベクトルアレイの中の、これまでのすべての読取り値のフィルタ処理平均値を反映します。
5. 計算結果のデータフォーマット（ASCIIまたは2進）を選択するには、:FORMat:DATA?コマンドを使用します（FORMatサブシステム参照）。\*RSTと\*SYSTem:PRESetデフォルトはASCIIです。
6. ブレース（[]）が数式の外に残されている場合は、その数式がアレイの最初のベクトル点を参照しているものと想定されます（すなわちVOLTは、VOLT[0]と同じです）。

## Enable and read math expression result (数式結果を使用可能にし、読み取る)

:STATe <b>

:CALCulate[1]:STATe <b>

CALC1 を制御せよ。

パラメータ <b>= 0 または OFF  
1 または ON

CALC1 の計算を解除状態にせよ。  
CALC1 計算を使用可能にせよ。

照会 :STATE? CALC1 の状態 (オンまたはオフ) を照会せよ。

説明 このコマンドを使用して、CALC1 の計算を使用可能または解除状態にします。使用状態にあるときには、ソースメータがトリガされてプログラムソース-メジャー動作を始めると、選択した数式の処理が実行されます。

ソースメータがアイドル状態に戻ったあと、:CALC1:DATA? コマンド (次のコマンド参照) を使用して、選択数式の計算結果を読み取ることができます。

解除状態にあるときには、:CALC1:DATA? コマンドが、+9.91e37 という NAN (数ではない) 値を返します。

:DATA?

:CALCulate[1]:DATA?

CALC1 の結果を読み取れ。

説明 この照会コマンドを使用して、CALC1 の計算の結果を読み取ります。有効な計算結果の最大値は、± 9.9e37 であり、これは (SCPI の定義によれば) 無限大です。

スカラー数式 (非ベクトル数式) の場合は、このコマンドを使用して、すべてのプログラムソース-メジャー点についての計算結果を返します。たとえば、20 回のソース-メジャー動作が行われた場合は、このコマンドによって 20 個の計算結果が返されます。

ベクトル数式の場合は、このコマンドは、指定したベクトル点の計算結果を返すだけです。

+9.91e37 という無効 NAN (数ではない) が結果として表示されれば、下記の状態が存在することを意味します。

- ・ 数式の中に誤りがある。
- ・ 必要な測定機能が解除状態になっている。
- ・ CALC1 が解除状態になっている (:STATe 参照)。

注記 ソースメータ内部の各ブロックをデータが流れる状況の詳細な説明については、付録 C の「データフロー」を参照してください。これを読めば、データを読み取るために各種のコマンドを使って取得した読み取り値の種類が、明らかになります。

:LATest?

:CALCulate[1]:DATA:LATest?

最新の CALC1 の結果を読み取れ。

説明 このコマンドは、最新の CALC1 の結果のみを返すという点を除いては、CALC1:DATA? とまったく同じように働きます。

## CALCulate2

## リミット試験を設定し制御せよ。

下記のコマンドを使用して、DUT についての3つのリミット試験を設定し、制御します。ピニング動作を行うためにハンドラとともに使用する場合は、ソースメータとハンドラとの通信は、デジタル I/O ポートを経由して行われます。デジタル出力ラインのための、多数の制御項目は、SOURce2 サブシステムを使って実施します。これらの制御項目に含まれるのは、デジタル出力ラインの設定とクリアリング、およびパルス幅の設定です。詳細については、「SOURce2 サブシステム」を参照してください。

## Select input path (入力パスを選択せよ)

:FEED <name>

:CALCulate2:FEED<name>                      リミット試験の入力パスを選択せよ。

パラメータ	<name> =	CALCulate[1]	CALC1 の結果を使用せよ。
		VOLTage	測定した電圧読取り値を使用せよ。
		CURRENT	測定した電流読取り値を使用せよ。
		RESistance	測定した抵抗読取り値を使用せよ。

照会            :EFFD?                      リミット試験の入力パスを照会せよ。

説明            このコマンドを使用して、リミット試験の入力パスを選択します。CALCulate[1]を選択すると、指定したリミットが、CALC1 の結果と比較されます。VOLTageを選択すると、リミットは電圧測定値と比較されます。CURRENT または RESistance を選択すると、リミットは、電流測定値、抵抗測定値と、それぞれ比較されます。

## Null feed reading (ヌルフィード読取り値)

:OFFSet <n>

:CALCulate2:NULL:OFFSet <n>                      フィードのヌルオフセット (REL) を指定せよ。

パラメータ    <n> = -9.999999e20 から 9.999999e20                      ヌルオフセット値を指定せよ。

照会            :OFFSet?                      ヌルオフセット値を照会せよ。

説明            このコマンドを使用すれば、選択したフィードに対するヌルオフセット (REL) を設定することができます。ヌルオフセットが使用可能な場合 (:NULL:STATe 参照) は、結果は、フィードの読取り値とオフセット値の代数差となります。  
CALC2 の読取り値 = フィードの読取り値 - ヌルオフセット

:ACQuire

:CALCulate2:NULL:ACQuire                      自動的に REL 値を取得せよ。

説明            このコマンドは、ヌルオフセット値を自動的に取得します。次に来る利用可能な読み取り値が、オフセット値となります。

:STATe <b>

:CALCulate2:NULL:STATe <b>

ヌルオフセットを制御せよ。

パラメータ <b>= 1 または ON

ヌルオフセットを使用許可にせよ。

0 または OFF

ヌルオフセットを解除状態にせよ。

照会

\*STATe?

ヌルオフセットの状態を照会せよ。

説明

このコマンドを使用して、ヌルオフセットを使用可能または解除状態にします。使用可能の場合は、CALC2の読取り値は、ヌルオフセット値を含みます (:OFFSet 参照)。解除状態の場合は、CALC2はヌルオフセットを含みません。

## Read CALC2 (CALC2 を読み取れ)

:DATA?

\*CALCulate2:DATA?

CALC2 を読み取れ。

説明

このコマンドを使用して、CALC2 リミット試験に使用した読取り値を取得します。ヌルオフセットが使用可能 (ヌルフィールド読取り値を参照) であれば、CALC2の読取り値はヌルオフセット値を含みます。リミット試験のうち、少なくとも1つは、リミット試験読取り値を取得するために、使用状態にする必要があります (「リミット試験の設定と制御 :STATe 参照」)。

注記

ソースメータの各動作ブロックを通過するデータの流れの状況の詳細な説明については、付録Dの「データフロー」を参照してください。

:LATest?

:CALCulate2:DATA:LATest?

最新のCAL2の結果を読み取れ。

説明

このコマンドは、最新のヌルオフセット値のみを戻すという点を除いては、CALC2:DATA とまったく同じように働きます。

## Configure and control limit tests (リミット試験の設定と制御)

:COMPLiance:FAIL <name>

:CALCulate2:LMit[1]:COMPLiance:FAIL <name>

パラメータ <name>=IN ユニットがコンプライアンスに入るとリミット1試験に不合格。

OUT ユニットがコンプライアンスから出るとリミット1試験に不合格。

照会

:FAIL?

リミット1試験が不合格になった時に照会せよ。

説明

このコマンドを使用して、リミット1試験不合格の原因となる条件を指定します。INを指定する場合は、ソースメータがコンプライアンスに入ると、試験は不合格となります。OUTを指定する場合は、ソースメータがコンプライアンスから出ると、試験は不合格となります。

[:DATA]&lt;n&gt;

:CALCulate2:LIMitx:LOWer[:DATA] &lt;n&gt; 下リミット x を指定せよ。(X = 2,3,2-12)

:CALCulate2:LIMitx:UPPer[:DATA] &lt;n&gt; 上リミット x を指定せよ。(X = 2,3,2-12)

パラメータ &lt;n&gt; = -9.999999e20 から +9.999999e20 リミット値を指定せよ。

DEFault

指定下リミットを -1 に設定せよ。

指定上リミットを 1 に設定せよ。

MINimum

指定リミットを -9.999999e20 に設定せよ。

MAXimum

指定リミットを +9.999999e20 に設定せよ。

照会

:UPPer?

指定上リミットを照会せよ。

:UPPer? DEFault

\*RST デフォルト値上リミットを照会せよ。

:UPPer? MINimum

最小許容上リミットを照会せよ。

:UPPer? MAXimum

最大許容上リミットを照会せよ。

:LOWer?

指定下リミットを照会せよ。

:LOWer? DEFault

\*RST デフォルト値下リミットを照会せよ。

:LOWer? MINimum

最小許容下リミットを照会せよ。

:LOWer? MAXimum

最大許容下リミットを照会せよ。

説明

これらのコマンドを使用して、LIMIT 2、LIMIT 3、LIMIT 5 の上下リミットを設定します。実際のリミットは、どの測定機能が現在選択されているか、によって決まります。たとえば、1  $\mu$  のリミット値は、電流機能に対しては 1  $\mu$  A、電圧機能に対しては 1  $\mu$  V です。リミット値は、レンジ依存ではありません。電圧の場合の 2 というリミットは、すべての測定レンジについて 2V です。

:SOURce2&lt;NRf&gt;|&lt;NDN&gt;

:CALCulate2:LIMit[1]:COMpliance:SOURce2 &lt;NRf&gt;|&lt;NDN&gt; リミット 1 不合格のパターンを指定せよ。

:CALCulate2:LIMit2:LOWer:SOURce2 &lt;NRf&gt;

グレーディングモードの場合の下リミット x 不合格のパターンを指定せよ (X=2,3,5-12)。

:CALCulate2:LIMit2:UPPer:SOURce2 &lt;NRf&gt;

グレーディングモードの場合の上リミット x 不合格のパターンを指定せよ (X=2,3,5-12)。

パラメータ <NRf> = 0 から 7 (3-bit) 10 進値  
 0 から 15 (4-bit) 10 進値  
 <NDN> = 0 から #b111 (3-bit) 2 進値  
 0 から #b1111 (4-bit) 2 進値  
 0 から #q7 (3-bit) 8 進値  
 0 から #q17 (4-bit) 8 進値  
 0 から #h7 (3-bit) 16 進値  
 0 から #h17 (4-bit) 16 進値

照会

:SOURce2? 指定リミットに対するソース値を照会せよ。

説明

これらのコマンドを使用して、指定した試験のデジタル出力「不合格」パターン (0 から 7 は 3 ビット、0 から 15 は 4 ビット) を定義します。リミット 2、3、5-12 の「不合格」パターンはグレーディングモードだけに適用されることに注意してください。

試験は次の順序で行われます。

1. リミット 4 試験 (コンタクトオプションのみ。付録 F 参照)



2. リミット試験1
3. リミット試験2
  - A. 下リミット2
  - B. 上リミット2
4. リミット試験x、ここでx=昇順で3、5-12
  - C. 下リミット
  - D. 上リミット

試験シーケンスでの最初の不合格によって、デジタル出力ポートのビットパターンが決められます。試験シーケンスでのそれ以後の不合格は、定義したデジタル出力パターンを変えません。出力値は、2進値、8進値または16進値で指定できることに留意してください。

下記の表を使用して、所要のデジタル出力パターンのパラメータ値を決めてください。非10進パラメータと合わせるために、10進値を等価な2進値、8進値または16進値に変換してください。

OUT 4*	OUT 3	OUT 2	OUT 1	0進値
L	L	L	L	0
L	L	L	H	1
L	L	H	L	2
L	L	H	H	3
L	H	L	L	4
L	H	L	H	5
L	H	H	L	6
L	H	H	H	7
H	L	L	L	8
H	L	L	H	9
H	L	H	L	10
H	L	H	H	11
H	H	L	L	12
H	H	L	H	13
H	H	H	L	14
H	H	H	H	15

L=Low (Gnd)

H=High (>+3V)

\*3ビットモードではOUT4は使用されません(値=0から7)

「不合格」条件が発生すれば定義した「不合格」ビットパターンを直ちにデジタル出力に出力するように、ソースメータを設定することができます。また、ソースメータはあるデバイスパッケージについてのすべての試験が完了するまで(動作はトリガレイヤーを離れます)待つこともできます。詳細については、「複合試験:BCONtrol」を参照してください。

**PASS:SOURce2<NRf>NDN**

:CALCulate2:LIMitx:PASS:SOURce2 <NRf>|<NDN> ソーティングモード「合格」パターンを設定せよ (X=2,3,5-12)。

パラメータ <NRf> =	0 から 7 (3-bit)	10 進値
	0 から 15 (4-bit)	10 進値
<NDN> =	0 から #b111 (3-bit)	2 進値
	0 から #b1111 (4-bit)	2 進値
	0 から #q7 (3-bit)	8 進値
	0 から #q17 (4-bit)	8 進値
	0 から #h7 (3-bit)	16 進値
	0 から #h17 (4-bit)	16 進値

**照会** :SOURce2 プログラムされたソース値を照会せよ。

**説明** このコマンドを使用して、ソーティングモードでの試験(リミット 2、3、5-12)に合格する場合の、ディジタル I/O ポートの 3 ビットまたは 4 ビット出力パターンを定義します。出力値は、2 進、8 進、または 16 進フォーマットで定義することができることに注意してください。:SOURce コマンドの「説明」の項に示す表を使用して、所要の 10 進ディジタル出力パターンのパラメータ値を決めて下さい。

**:STATe<b>**

:CALCulate2:LIMit[1]:STATe <b> リミット 1 試験を制御せよ。  
:CALCulate2:LIMitx:STATe <b> リミット x 試験を制御せよ (X=2,3,5-12)。

パラメータ <b>=1 または ON 指定されたりミット試験を使用可能にせよ。  
0 または OFF 指定されたりミット試験を使用禁止にせよ。

**照会** :STATe 指定されたりミット試験の状態を照会せよ。

**説明** これらのコマンドを使用して、リミット 1、リミット 2、リミット 3、およびリミット 5 からリミット 12 までの試験を使用可能または使用禁止にします。どのようなリミット試験でも、使用可能にされてなければ実行されません。リミット試験が使用可能になれば、ディジタル I/O ポートは、リミット試験の制御を受けます。すなわち、試験プロセスの結果が、I/O ポート上の出力パターンをアップデートします。

**:FAIL?**

:CALCulate2:LIMit[1]:FAIL? リミット 1 試験の結果を読み取れ。  
:CALCulate2:LIMitx:FAIL? リミット x 試験の結果を読み取れ(X=2,3,5-12)。

**説明** これらのコマンドを使用して、リミット 1、リミット 2、リミット 3、およびリミット 5 からリミット 12 までの試験結果を読み取ります。  
0 = リミット試験合格  
1 = リミット試験不合格

応答メッセージ (0 または 1) は、リミット試験の合否だけを知らせます。ミット 2、リミット 3、およびリミット 5 からリミット 12 までの試験については、メッセージはどちらのリミット (上リミットまたは下リミット) で不合格になったのかは、伝えません。どちらのリミットで不合格になったのかを確認するには、測定イベントレジスタを読む必要があります。([ステータスサブシステム] 参照)

リミット試験の結果を読み取っても、試験の不合格表示はクリアされません。不合格をクリアするには、:CLEar コマンドを使います。

## Composite testing (複合試験)

**PASS:SOURce2<NRf>NDN**

:CALCulate2:CLIMits:PASS:SOURce2 &lt;NRf&gt;|&lt;NDN&gt; 複合「合格」パターンを指定せよ。

パラメータ <NRf> =	0 から 7 (3-bit)	10 進値
	0 から 15 (4-bit)	10 進値
<NDN> =	0 から #b111 (3-bit)	2 進値
	0 から #b1111 (4-bit)	2 進値
	0 から #q7 (3-bit)	8 進値
	0 から #q17 (4-bit)	8 進値
	0 から #h7 (3-bit)	16 進値
	0 から #h17 (4-bit)	16 進値

照会 :SOURce2? プログラムされたソース値を照会せよ。

説明 このコマンドを使用して、不合格がない場合の、デジタル I/O ポートの 3 ビットまたは 4 ビット出力パターンを定義します。出力値は、2 進、8 進、または 16 進フォーマットで指定できることに注意してください。:SOURce コマンドの「説明」の項に示す表を使用して、所要の 10 進デジタル出力パターンのパラメータ値を決めて下さい。

ソース・メータは、「合格」条件が発生すれば直ちに、定義された「合格」ビットパターンをデジタル出力に送るように設定することもできますし、あるいはデバイスパッケージに関するすべての試験が完了するまで、待つこともできます(動作はトリガレイヤーを離れます)。詳細は「複合試験と BCONtrol」を参照してください。

ソーティングモードの場合は、このコマンドは、リミット 2、3、5-12 が使用禁止の場合の、リミット 1 試験(コンプライアンス)に対する 3 ビットまたは 4 ビット出力「合格」パターンを定義します。

**FAIL:SOURce2<NRf>NDN**

:CALCulate2:CLIMits:FAIL:SOURce2 &lt;NRf&gt;|&lt;NDN&gt; 「不合格」パターンを指定せよ。

パラメータ <NRf> =	0 から 7 (3-bit)	10 進値
	0 から 15 (4-bit)	10 進値
<NDN> =	0 から #b111 (3-bit)	2 進値
	0 から #b1111 (4-bit)	2 進値
	0 から #q7 (3-bit)	8 進値
	0 から #q17 (4-bit)	8 進値
	0 から #h7 (3-bit)	16 進値
	0 から #h17 (4-bit)	16 進値

照会 :SOURce2? プログラムされたソース値を照会せよ。

説明 ソーティングモードの場合には、このコマンドを使用して、不合格がある場合の、デジタル I/O ポートの 3 ビットまたは 4 ビット出力パターンを定義します。出力値は、2 進、8 進、または 16 進フォーマットで指定できることに注意してください。:SOURce コマンドの「説明」の項に示す表を使用して、所要のデジタル出力パターンの 10 進パラメータ値を決めて下さい。

**PASS:SMLocation<NRf>NEXT**

:CALCulate2:CLIMits:PASS:SMLocation <NRf>:Next 「合格」 ソースメモリロケーションを指定せよ。

パラメータ <NRf>=1 から 100      メモリロケーションを指定せよ。  
NEXT      リストに含まれる次のメモリロケーション  
(現在位置+1)

照会 :SMLocation? 「合格」 ソースメモリロケーションを照会せよ。

説明 リミット試験実施に際してソースメモリスイープを使っている間は、スイープは指定メモリロケーションに分岐するか、リストに含まれる次のメモリロケーションに進むことができます。

メモリロケーションが指定されている場合は、試験が順調に進めばスイープはそのメモリロケーションに分岐します (PASS 条件)。失敗した場合は (FAIL 条件)、スイープはリスト中の次のメモリロケーションに進みます。NEXT を選択すれば (デフォルト)、試験結果 (PASS または FAIL 条件) に関係なく、スイープは、リスト中の次のメモリロケーション (現在位置+1) に進みます。

詳細は、第 10 部の「ソースメモリスイープ」を参照してください。

**:BCONtrol<name>**

:CALCulate2:CLIMits:BCONtrol <name>      デジタル I/O ポートの合格/不合格アップデートを制御せよ。

パラメータ <name>=IMMediate      第 1 の不合格が発生したときにアップデート出力  
END      スイープ完了後にアップデート出力

照会 :BCONtrol      いつデジタル出力がアップデートするか照会せよ。

説明 このコマンドを使って、デジタル出力が「合格」または「不合格」パターンにアップデートされる時期を照会します。「合格」または「不合格」ビットパターンは、ハンドラに対して、試験プロセスを中止し、DUT を該当する区分容器に入れるように命令します。

IMMediate を選択したときは、デジタル出力は、試験プロセス中の、最初の不合格のビットパターンに直ちにアップデートされます。すべての試験に合格する場合は、出力は「合格」ビットパターンにアップデートされます。

END を選択したときは、デジタル出力は、ソース・メータがスイープまたはリスト動作を完了するまで、「合格」または「不合格」ビットパターンにアップデートしません。したがって、複数の試験サイクルを DUT について行うことが可能になります。スキャナカードを使用して、多素子デバイス (すなわち抵抗ネットワーク) を試験することができます。たとえば、仮にお客様が END を使わずに、しかもデバイスパッケージ中の第 1 素子が合格したとすると、「合格」ビットパターンが出力されます。試験プロセスは停止し、DUT は区分容器に置かれます。その結果、デバイスパッケージ中のほかの素子は、試験を受けないことになります。

:MODE<name>

:CALCulate2:CLIMits:MODE <name> デジタル I/O ポート合格/不合格出力を制御せよ。

パラメータ <name>=GRADing      グレードされた「合格/不合格」パターンを出力せよ。  
SORTing      ソートされた「合格/不合格」パターンを出力せよ。

照会 :MODE? デジタル I/O 合格/不合格パターンを照会せよ。

説明 このコマンドは、リミット計算がデジタル I/O ラインをドライブするモードを制御します。GRADing モードでは、読み取り値が、使用可能なすべての上/下リミット許容値の範囲内であれば、合格です。ただし、読み取り値がリミット 4 コンタクトチェック (コンタクトチェック点のみ) とリミット 1 コンプライアンス試験にまず合格していることが前提となります。デジタル I/O ラインは、第 1 のコンタクトチェック (オプション)、コンプライアンス、上リミットまたは下リミットの不合格のパターンでドライブされます。これ以外の場合は、CALC2:CLIM:PASS:SOUR2 のパターンが出力されます。

ソーティングモードでは、読み取り値が不合格となる条件は、コンタクトチェック試験とコンプライアンス試験に不合格となるか、それともデジタル I/O 帯域のどの範囲にも入らないことです。試験に合格し、リミット 1 または 4 だけが使用可能な場合は、CALC2:CLIM:PASS:SOUR2 のパターンが出力されます。これ以外の場合は、合格する第 1 のリミット試験帯域が、その LOW:SOUR2 パターンを出力します (UPP:SOUR2 パターンは無視されます)。リミット 1 またはリミット 4 に不合格になる場合は、それぞれの SOUR2 パターンが出力されます。リミット 2、3、5-12 リミット試験に合格しない場合は、CALC2:CLIM:FAIL:SOUR2 のパターンが出力されます。

## Clear test results (試験結果のクリアリング)

[:IMMediate]

:CALCulate2:CLIMits:CLear[:IMMediate]      試験結果をクリアし、デジタル I/O ポートをリセットせよ。

説明 このコマンドはリミット試験の試験結果 (合格または不合格) をクリアし、デジタル I/O ポートの出力ラインを:SOURce2:TTL の設定値にリセットします。  
([SOURce2 サブシステム] 参照)

:AUTO<b>

:CALCulate2:CLIMits:CLear:AUTO <b>試験結果を得るためにオートクリアを制御せよ。

パラメータ <b>=1 または ON      オートクリアを使用可能にせよ。  
0 または OFF      オートクリアを使用禁止にせよ。

照会 :AUTO?      オートクリアの状態を照会せよ。

説明 オートクリアを使用可能状態にすると、試験結果はクリアされます。そして:INITiate コマンドが送出されて新しい試験シーケンスが開始されると、デジタル I/O ポートの出力ラインはリセットします。

これが使用禁止になると、クリアアクションを実行するには、:IMMediate を使わなければなりません。

## CALCulate3

バッファ読取り値の統計データを提供せよ。

## Select statistic (統計の選択)

:FORMat &lt;name&gt;

:CALCulate3:FORMat &lt;name&gt; CALC3 フォーマットを指定せよ。

パラメータ <name> =    MEAN        バッファの中の読取り値の平均値  
                              SDEVIation   バッファの中の読取り値の標準偏差  
                              MAXimum    バッファの中の最大読取り値  
                              MINimum    バッファの中の最小読取り値

照会        :FORMat?    プログラムされた数式フォーマットを照会せよ

説明        このコマンドを使用して、バッファに格納された読取り値についての、所要の統計値を選択します。これらの統計値についての詳細については、第3部の「データストア」を参照してください。

バッファに格納された読取り値は、CALC1 の計算の「生の」測定読取り値であることも、また CALC2 の読取り値であることもあります。:TRACe サブシステムの:TRACe:FEED コマンドを使用して、格納すべき読取り値の種類を選択します。

## Acquire statistic (統計の取得)

:DATA?

:CALCulate3:DATA?    CALC3 の結果を読み取れ。

説明        この照会コマンドを使用して、選択した統計操作を行い、その結果を読み取ります。結果は常に ASCII フォーマットで戻されます。

バッファの設定が「生の」測定読取り値を格納する (:TRACe:FEED SENSE 1) ようになっており、複数の機能を測定する場合は、選択した動作はすべての測定読取り値に対して行われます。たとえば、電圧と電流の測定値がバッファに格納されると、どちらの読取り値に対しても、選択した統計操作が行われます。複数の測定機能に対する統計値は、次の順序で戻されます。  
 電圧に関する統計値、電流に関する統計値、抵抗に関する統計値

バッファに格納された TIME と STATus のデータ要素については、統計操作は行われません。

バッファが CALC1 または CALC2 の結果を格納するように設定されていれば (:TRACe:FEED CALC1 または CALC2)、1つの結果だけがこの照会コマンドによって戻されます。

## 注記：

1. バッファにデータがなければ、NAN（数ではない）+9.91e37 が戻されます。
2. 多数の読取り値がバッファの中にある場合は、一部の統計操作の実行時間が長すぎて、バスタイムアウトエラーが発生することがあります。これを避けるには、`:calc3:data?` コマンドを送り、ステータスバイトレジスタの中の MAV（メッセージ利用可能）ビットが設定されるまで待ち、そのあと、2400 型に対して talk するように呼びかけてください（第 4 部の「ステータス体系」を参照）。
3. ソースメータの各動作ブロックを通過するデータの流れの状況の詳細な説明については、付録 D の「データフロー」を参照してください。これを参照すれば、データを読み取るためにいろいろなコマンドが取得する読取り値の種類が明らかになります。