

---

# アレイ 372X シリーズ

3720A(250W)/3721A(400W)

## SCPI プログラミング説明書



Ver.1.01

作成 2011年8月8日

**株式会社ティ・アンド・シー・テクニカル**

本社：東京都足立区 千住仲町40-12

☎: 03(3870)7101 FAX: 03(3870)7102

藤代事業所：茨城県取手市宮和田 448-1

☎: 0297(83)0721 FAX: 0297(82)7127

©2011 株式会社ティ・アンド・シー・テクニカル

日本語版文書番号：ARR372X\_TAC\_J\_SCPI, 2011年6月8日



## 目 次

1. はじめに .....	6
1.1 SCPI について .....	6
1.2 SCPI コマンド構文 .....	7
1.2.1 SCPI キーワード .....	7
1.2.2 セパレーター .....	8
1.2.3 変数、項目 .....	9
1.2.4 ターミネーターと通信設定 .....	10
1.3 コマンドの例 .....	12
1.3.1 シンプルコマンド文 .....	12
1.3.2 複合コマンド文 .....	12
1.3.4 複合コマンドクエリー .....	12
2. SCPI コマンドの説明 .....	13
2.1 IEEE488.2 コモンコマンド .....	13
2.2 SCPI ルートコマンド .....	19
2.2.1 CURRent サブシステム .....	23
[SOURce:]CURRent[:LEVel][IMMediate] .....	23
[SOURce:]CURRent[:LEVel]:TRIGgered .....	23
[SOURce:]CURRent:PROTection[:LEVel] .....	23
[SOURce:]CURRent:PROTection:DELaY .....	24
[SOURce:]CURRent:PROTection:STATe .....	24
[SOURce:]CURRent:LLEVel .....	25
[SOURce:]CURRent:HLEVel .....	25
[SOURce:]CURRent:RISE:RATE .....	25
[SOURce:]CURRent:FALL:RATE .....	26
2.2.2 VOLTage サブシステム .....	27
[SOURce:]VOLTage[:LEVel][IMMediate] .....	27
[SOURce:]VOLTage[:LEVel]:TRIGgered .....	27
[SOURce:]VOLTage:LLEVel .....	27
[SOURce:]VOLTage:HLEVel .....	28
2.2.3 RESistance サブシステム .....	28
[SOURce:]RESistance[:LEVel][IMMediate] .....	28
[SOURce:]RESistance[:LEVel]:TRIGgered .....	29
[SOURce:]RESistance:LLEVel .....	29
[SOURce:]RESistance:HLEVel .....	29

2.2.4	POWer サブシステム	30
	[SOURce:]POWer[:LEVel][IMMEDIATE]	30
	[SOURce:]POWer[:LEVel]:TRIGgered	30
2.2.5	TRANsient サブシステム	31
	[SOURce:]TRANsient[:STATe]	31
	[SOURce:]TRANsient:MODE	31
	[SOURce:]TRANsient:HTIME	31
	[SOURce:]TRANsient:LTIME	32
	[SOURce:]TRANsient:RTIME	32
	[SOURce:]TRANsient:LTIME	32
2.2.7	MEASure	33
	MEASure:VOLTage[:DC]?	33
	MEASure:CURRent[:DC]?	33
	MEASure:POWer[:DC]?	33
	MEASure:RESistance[:DC]?	33
2.2.8	INPut サブシステム	34
	INPut[:STATe]	34
	INPut:SHORt[:STATe]	34
	INPut:PROTection:CLEar	34
	INPut:VOLTage:ON	35
	INPut:VOLTage:ON:LATCh	35
2.2.9	LIST サブシステム	36
	LIST[:STATe]	36
	LIST:CLEar	36
	LIST:NUMber	36
	LIST:MEMO	37
	LIST:ADD	37
	LIST:INSert	37
	LIST:EDIT	37
	LIST:DELeTe	38
	LIST:DELeTe:ALL	38
	LIST:COUNT	38
	LIST:CHAln	38
	LIST:SAVe	38
2.2.10	TRIGger サブシステム	39
	TRIGger[:IMMEDIATE]	39

TRIGger:SOURce.....	39
TRIGger:FUNcTion.....	39
2.2.11 CV:CURRent:LIMIt 電圧モードにおける入力電流制限.....	40
2.2.12 BATTery サブシステム.....	40
BATTery[:STATe].....	40
BATTery:VOLTage:OFF.....	40
BATTery:DIScharge:CURRent.....	41
BATTery:CAPAcity?.....	41
BATTery:TIME?.....	41
2.2.13 ABORt.....	42
2.2.14 STATus サブシステム.....	42
STATus:QUEStionable[:EVENT]?.....	42
STATus:QUEStionable:CONDition?.....	42
STATus:QUEStionable:ENABLE.....	43
STATus:OPERation[:EVENT]?.....	43
STATus:OPERation:CONDition?.....	43
STATus:OPERation:ENABLE.....	44
2.2.15 SYSTem.....	44
SYSTem:ERRor?.....	44
SYSTem:VERSion?.....	45
SYSTem:REMote.....	45
SYSTem:LOCal.....	45
2.2.16 MODE.....	46
MODE.....	46
3. エラーメッセージ.....	47
4. ステータス・レジスター・リポーティング.....	48
SCPI ステータス・レジスター.....	48
ステータス・レジスターの体系.....	50

## 1. はじめに

### 1.1 SCPI について

アレイエレクトロニクス社 モデル 3720A(250W)、3721A(400W)はインターフェースオプションで IEEE488.1、IEEE488.2、また IEEE754 および ISO646 7 ビットコード文字セット等の標準を採用しています。これにより SCPI による外部制御を通じ、装置の持つすべての機能を行えるようにしています。

本装置の SCPI コマンドセットは IEEE488.2 に定められた標準コマンドセット、そして独自のコマンドセットがあります。標準コマンドは\*で始まり 3 文字で構成された表記となっています(例：\*IDN?、\*SAV、\*WAI など)。

独自のコマンドは主に設定、試験そして測定を実行する等、負荷装置の機能を用いるためのものです。これらのコマンドはツリー構造となっており、それぞれ分類がなされています。枝分かれのところは“:”で表されています。ツリーのトップコマンドはルートと呼ばれ、枝、そして葉に至る経路がすべて記述されることでプログラムとなります。

## 1.2 SCPI コマンド構文

372XA 負荷装置用 SCPI コマンドは IEEE488.2 の標準コマンドを拡張したものを uses。プログラムされるコマンドはキーワード、セパレーター、変数(項目)そしてターミネーター等で構成されます。いかに例を示します。

```
CURRent:PROTection:STATe ON
```

CURRent、PROTection、STATe はキーワードにあたり、コロン「:」はセパレーターです。ON は項目となります。

備考:いくつかのコマンドは複数の項目(変数)を持ちます。この項目(変数)のセパレーターはカンマ「,」を uses。

構文の説明を容易にするため、数種類のフォローアップシンボルを uses 表記を簡略化しています。

[ ] : 囲まれているところは、オプション的なキーワードあるいは項目で、省略が可能です。

{ } : 囲まれているところは、コマンド構文のなかで項目のオプションを表記します。

<> : 囲まれているところは、数値項目です。数字の入力を行います。

| : は複数の選択項目があるとき、その項目の区切りを表します。

### 1.2.1 SCPI キーワード

#### 1.2.1.1 キーワードのフォーム

キーワードのフォームは短いものと長いものの 2 種類があります。

**ロングフォーム**: 一つの単語あるいは 1 つの節で構成されます。単語である場合、ロングフォームキーワードは単語としての表記となり、節の場合は最初の単語の頭文字と次に続く単語を接続した構成となります。いかに例を示します。

STATUS はロングフォームでは STATus となります。

LOW LEVEL は LLEVEL となります。

**ショートフォーム**: 一つの単語あるいは 1 つの節で構成されます。単語である場合、ショートフォームキーワードは単語の最初の 3 文字もしくは 4 文字の表記となり、節の場合は最初の単語の頭文字と次に続く単語を接続し 3 文字もしくは 4 文字の表記となります。

STATUS はショートフォームでは STAT となります。

LOW LEVEL は LLEV となります。

ショートフォームは以下のルールで作成されています。

キーワードが 4 文字もしくはそれ以下の文字の単語の場合、それらはすべて使用されません。

キーワードが 4 文字を超える場合、また 4 番目の文字が子音の場合、最初の 4 文字が使用されます。また 4 番目の文字が母音(A,E,I,O,U)の場合、最初の 3 文字が使用されます。

#### 1.2.1.2 キーワード変換

ショートフォームのそれぞれのキーワードは大文字で表現し、そのキーワードを覚えやすくしています。

TRIGger  
IMMediate  
RESistance  
SHORT

SCPI の構文解析機は大文字小文字に敏感ではありません。そのため Trig, trig、trigger、TRIGGER など、大文字小文字が今までのルールに当てはまらなくても、キーワードがロングフォーム、ショートフォームで綴られ、それらが正確であればプログラムとして成立します。例として RESI あるいは TRI は正しいコマンドとして認識されません。

#### 1.2.2 セパレーター

コロン「:」はコマンドツリーに複数のキーワードがある場合、個々の階層のキーワードの分岐に使用します。例を以下に示します。

INPut:PROTection:CLEar コロンは、コマンドを示す上で、コマンドの最初の文字とみることができ、つまりルートコマンドの役割となります。

スペースはコマンドと項目を離すのに使用します。

セミコロン「;」は複数のコマンドを現在のコマンドレベル内で変更すること無く使用する場合に用います。例として CURR:PROT:LEV 3;DEL 10 を示します。このコマンドは以下の二つのコマンドと同等になります。

CURR:PROT:LEV 3  
CURR:PROT:DEL 10

セミコロンとコロンを適切に用いることでコマンドレベルの記述を単純化することができます。例えば CURR 25;VOLT 50 などの表記が行えます。注意点としてコマンドの長さは 100 バイトを超えないようにしてください。コマンドがこの長さを超えた場合、-521 入力バッファオーバーエラー

ーが発生します。コンマ「,」は複数の項目を分けるのに使用します。

### 1.2.3 変数、項目

項目、変数はデータの値あるいは文字列となり、それらはある特定のキーワードの後に伴っていません。そしてスペースでキーワードから離して表記されます。

#### 1.2.3.1 変数、項目の様式

項目の様式は数値データの様式、文字データ様式、そしてブール様式などがあります。文字データ様式とブーリーン様式はキーワードの様式と似ています。例えば ON、OFF などの表記がそうです。すべての項目は ASCII で表記されます。項目の様式についてはリスト 1-1 を参照してください。

表 1-1 項目の様式

シンボル	内容	例
<NR1>	小数点なしの数字。小数点が最下位桁の右側にあるとみなされます。	2730、02730
<NR2>	小数点を伴った数字	27.30、.02730
<NR3>	小数点、指数を伴った数値	2.730 E+2、2.730 E-2
<NRf>	NR1 や NR2、NR3 を含んだ浮動小数点で表記される数字	2730、27.30、2.730E-2、
<NRf+>	NRf の数字形式に MIN と MAX が含まれるように拡張したものです。MIN と MAX は、パラメータの最小値と最大制限値です。	2730、27.30、2.730E-2、MIN、MAX
<Bool>	ブールデータです。	ON OFF
<crd>	文字データ	CV
<aard>	戻り ASCII データです。これは区切られていない7ビットの ASCII を返信します。このデータタイプはターミネーターを含んでいません。	

### 1.2.3.2 データの単位

単位を伴う形の数字のデータがあります。データに単位が付属しないとき、データの単位はコマンドで定義される標準単位が用いられます。表 1-2 はデータの単位、そして表 1-3 が用いられる単位の乗数です。

表 1-2 データの単位

クラス	単位	セカンダリーの単位	参照されている単位
Current	A		アンペア
Resistance	OHM		オーム
		MOHM	メガオーム
Time	S		秒
Amplitude	V		電圧
Power	W		ワット

表 1-3 乗数の表記と意味

指数の表記	ニーモニック	定義
1E6	MA	メガ
1E3	K	キロ
1E-3	M	ミリ
1E-6	U	マイクロ
1E-9	N	ナノ

### 1.2.4 ターミネーターと通信設定

コマンド文を負荷に送信するときは必ず LF を行い終端とします。IEEE-488 EOI は、コマンドの終端を設定するため LF をそのシンボルとみなしています。また LF シンボルが CR の後にあってもそれも終端として受け付けます。

#### ハード設定

RS-232C 接続についてはクロスケーブルをご使用してください。

#### ピンアサイン

Pin	Input/Output	Description
1	-	Not used
2	Input	RXD Receive data
3	Output	TXD Transmit data
4	Output	DTR Data terminal ready
5	Common	GND Ground
6	Input	DSR Data set ready
7	-	Not used
8	-	Not used
9	-	Not used

<通信テスト例>

\* RS232C での通信テストの場合

1. 通信の設定を行う。  
ボーレート、パリティチェック、データビット、ストップビット、フローコントロールの設定を PC、電子負荷装置共に同じにします。
2. コマンドを送信する。  
\*IDN?(CR+LF) (CR=0x0D, LF=0x0A)
3. データを受信する。  
下記のようなデータが返ってきたら通信が正しく確立されています。  
ARRAY,3721A,0,1.10-0.0-0.0(CR+LF)

### 1.3 コマンドの例

#### 1.3.1 シンプルコマンド文

シンプルコマンド文は1つのコマンド、あるいはキーワードで構成され、通常以下の変数がついてきます。

```
VOLT 25  
CURR 50  
TRIG
```

#### 1.3.2 複合コマンド文

2つあるいはそれ以上のキーワードがコロンの連続される場合、それは複合コマンド文となります。通常最後のキーワードは変数が付属します。そしてスペースで区切られます。

```
VOLT:SLEW 1000  
CURR:RANG 6  
TRIG:SOUR BUS
```

複合コマンド文が[ ]を含んでいる場合、そのキーワードの部分は省略できます。

```
CURRent[:LEVel]:TRIGgered 0.5
```

↓

```
CURRent:TRIGgered 0.5
```

上記のコマンドはともに同じ動作です。

#### 1.3.4 複合コマンドクエリー

2つもしくはそれ以上のキーワードがコロンにより接続されていて、クエスチョンマークがついている場合、それは複合クエリー文となります。

```
VOLT:TRG?  
CURR:PROT?  
MEAS:POW?
```

## 2. SCPI コマンドの説明

### 2.1 IEEE488.2 コモンコマンド

コモンコマンドとは IEEE488.2 で標準のコマンドとして定義されるものです。これらは機器の基本的な機能の動作のために使用します。例として保存する、呼び出す、リセットする等がそれに当てはまります。また IEEE488.2 コモンコマンドは階層構造を持っていません。

#### \*CLS

以下のレジスターの内容を消去します。

標準イベントレジスター  
クエスチョナブル・ステイタス・イベント・レジスター  
オペレーション・ステイタス・イベント・レジスター  
ステイタス・バイト・イベント・レジスター  
エラーキューのクリアー

コマンド構文：\*CLS

項目・変数：無し

#### \*ESE

本コマンドはスタンダード・イベント・ステイタス・レジスターの状態を設定します。ビットのポジションを 1 とすると、対応するイベントを使用可能にし、ステイタス・バイト・レジスターへレポートします。詳細はステイタス・レジスター・レポート内スタンダード・イベント・ステイタス・イネイブル・レジスターを参照してください。

コマンド構文 : \*ESE<NRf>  
変数 : 0 ~ 255  
パワーオン時値 : \*PSC コマンドで参照  
例 : \*ESE 100  
クエリー構文 : \*ESE?  
戻り変数 : <NR1> 0 ~ 255  
関連コマンド : \*PSC \*STB?

\*ESR?

本クエリーはスタンダード・イベント・レジスターの値を戻します。レジスターを読み消去します。以下の表にビットの設定を示します。

表 2-1 スタンダード・イベント・レジスター ビット定義

ビット	7	6	5	4	3	2	1	0
ビット名	PON	N.U	CME	EXE	DDE	QYE	N.U	OPC
PON : パワーオン このビットは負荷装置の電源が入ると設定されます。 N.U : Null です。 CME : コマンドエラー EXE : 実行エラー					DDE : 機器に由来するエラー QYE : クエリーエラークエリーにエラーが発生すると QYE ビットが設定されます。 N.U : Null です。 OPC : 動作が終了しました。			

クエリー構文 : \*ESR?  
 変数 : 無し  
 戻り変数 : <NR1>  
 関連コマンド : \*CLS \*OPC

\*IDN?

負荷に関する情報をクエリーします。

クエリーの構文 : \*IDN?  
 変数 : 無し  
 戻り変数 : <aard>

返信例

ARRAY,3721A,0,1.10-0.0-0.0

戻り変数はコンマとハイフンで区切られた 6 つの連分となっています。最初の分は製造会社をあらわします。2 つ目は製品の型式、3 つ目は予備(空き)となっており常にゼロを戻します。4 つ目は負荷装置のファームウェアのバージョン番号、5 つ目は GPIB カードのファームウェアバージョン番号、6 つ目は USB のファームウェアバージョン番号となっています。

#### \*OPC

本コマンドは、負荷装置がすべての待機中の動作を実行し終わったときにスタンダード・イベント・ステータス・レジスターのビット 0 を 0 から 1 とするように設定します。\*OPC はその後続くコマンドの実行を妨げることはありません。すべての待機している動作、すべての待機しているトリガーレベルがトリガーされることを含め、それらが完了される状態を意味します。

コマンドの構文 : \*OPC  
変数 : 無し  
関連コマンド : \*WAI、\*OPC

#### \*OPC?

現在待機中の動作が完了したとき、出力のキューに ASCII で「1」の状態に設定します。\*OPC と異なり \*OPC? は以降に続くコマンドすべてのプロセスを妨げます。もし以降のプロセスにトリガーがある場合、そしてそのトリガーソースがそのとき EXT に設定されていない状態の場合、動作をリストアするためのプログラム可能な方法は DCL(デバイスクリア)コマンドを送ることのみとなります。

クエリー構文 : \*OPC?  
戻り変数 : <NR1>  
関連コマンド : \*OPC、TRIG:SOUR、\*WAI

#### \*PSC

本コマンドは負荷装置の電源が入るときサービス要求を制御します。

1: 負荷装置の電源が入り、ステータス・バイト・イネイブル・イネイブル・レジスターとスタンダード・イベント・イネイブル・レジスターが 0 に設定されます。

0: ステータス・バイト・イネイブル・レジスターとスタンダード・イベント・イネイブル・レジスターの現在の設定が不揮発性メモリーに保存されます。次に負荷装置に電源が入ったとき、スタンダード・イベント・イネイブル・レジスターは保存された設定でプログラムされます。

コマンド構文 : \*PSC <bool>  
変数 : 0|1  
クエリー構文 : \*PSC?  
戻り変数 : 0|1  
関連コマンド : \*SRE、\*ESE

#### \*RCL

本コマンドは負荷装置に保存されている項目を呼び出すのに使用します。そのため現在メモリーに保

存されている項目が必要となります。保存されている項目が呼び出されるとき、これらの変数の保存場所が指定される必要があります。\*RCL コマンドはまた以下のことを実行します。

1. 項目をリセットする前に ABORt コマンドを実行します。(これはすべての待機中トリガーレベルをキャンセルします)
2. \*RCL ですべての項目が呼び出された後、INP:PROT:CLE を実行し負荷装置の保護状態の消去を実行します。電源を入れると、負荷装置は\*RCL 0 の設定が実行されます。保存していない場所の呼び出しを行うと自動的に 0 番の設定が設定されます。

コマンド構文 : \*RCL <NR1>  
項目 : 0~9  
例 : \*RCL 5  
関連コマンド : \*RST、\*SAV

#### \*RST

本コマンドは負荷装置を工場出荷時の状態に設定します。このコマンドを使用する前に ABORt コマンドを実行してください。\*RST ですべての項目がリセットされました、INP:PROT:CLE を実行し負荷装置の保護状態の消去を実行します。

コマンドの構文 : \*RST  
変数 : 無し  
関連するコマンド : \*RCL、\*SAV

#### \*SAV

このコマンドは現在の設定を不揮発性メモリーにすべて保存します。保存は最大 10 個の設定が可能です(0 から 9 番)。ユーザーマニュアルを参照してください。

コマンドの構文 : \*SAV <NR1>  
変数 : 0~9  
例 : \*SAV 5  
関連コマンド : \*RCL、\*RST

#### \*SRE

本コマンドはステータス・バイト・イネイブル・レジスターの状態を設定します。プログラムされる変数は、ステータス・バイト・レジスターにおいて RQS ビットのステータス・バイト・レジスターに

0 から 1 への変化を決定することです。

ステータス・バイト・レジスターの詳細はステータス・リポーティングを参照してください。

コマンドの構文 : \*SRE <NR1>  
変数 : 0 ~ 255  
クエリーの構文 : \*SRE?  
戻り変数 : <NR1>  
関連コマンド : \*PSC

#### \*STB?

本クエリーはステータス・バイト・レジスターを戻します。ステータス・バイト・レジスターは本クエリーが実行された後消去されます。\*STB?クエリーとシリアルポールは、同じ戻り変数を使用します。しかしサービス・リクエスト・ビット(ビット 6)のステータス・バイト・レジスターはシリアルポールが送信された場合消去されません。

ステータス・バイト・レジスターの詳細はステータス・リポーティングを参照してください。

クエリーの構文 : \*STB?  
変数 : 無し  
戻り変数 : <NR1>

#### \*TRG

本コマンドは、実際<GET>コマンド(グループ実行コマンド)と同様のものですが、バスに TRIG:SOUR が設定された場合負荷装置に対しトリガーを発生します。

コマンドの構文 : \*TRG  
変数 : 無し  
関連コマンド : TRIG、TRIG:SOUR

#### \*TST?

本クエリーは負荷装置に対し制限された範囲で自己診断をジッパこうします。診断はモード、項目の設定等には影響しません。

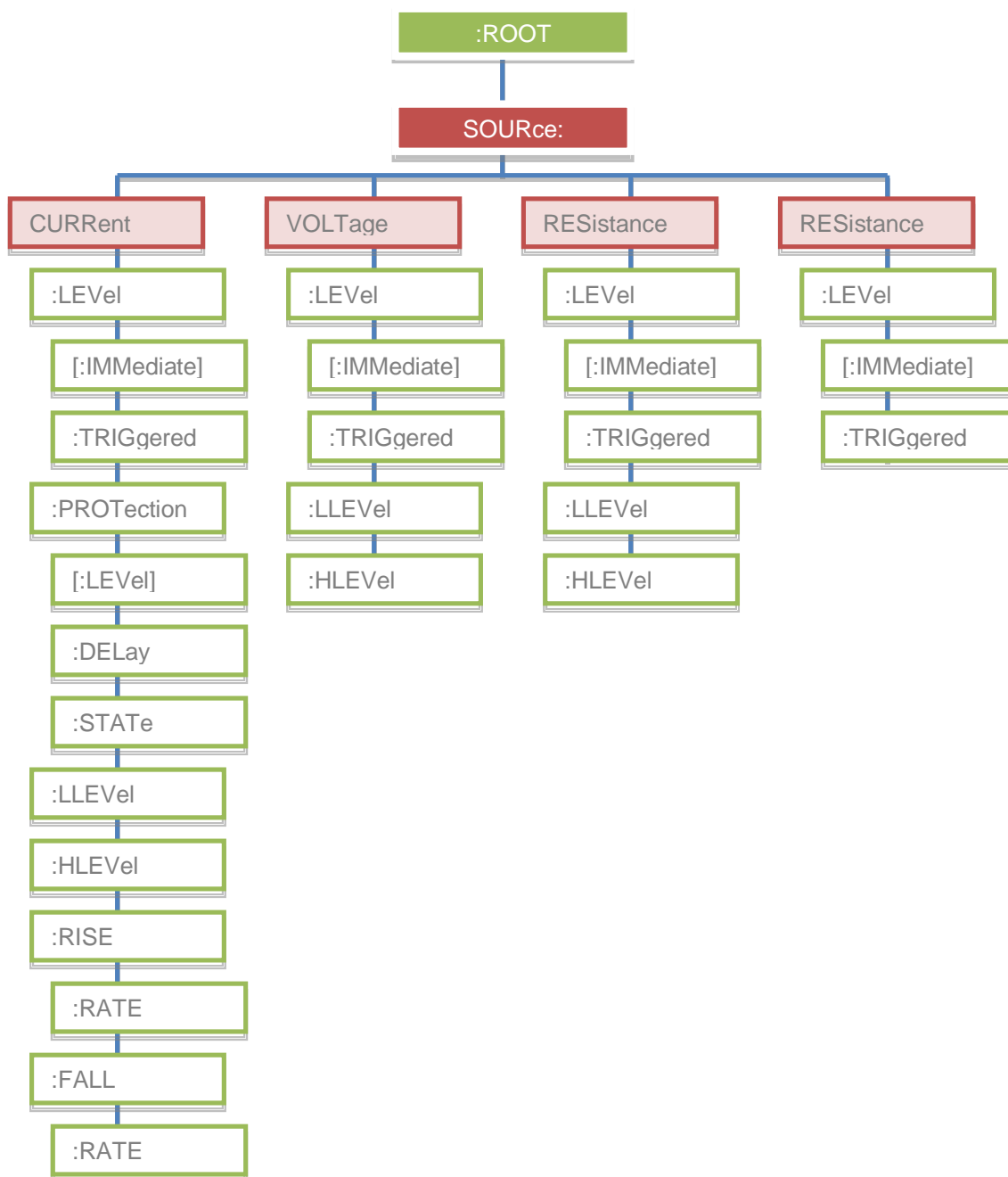
コマンドの構文 : \*TST?  
戻り変数 : <NR1> 0 = 師団は正常に終了 0 以外= 診断は失敗しました。

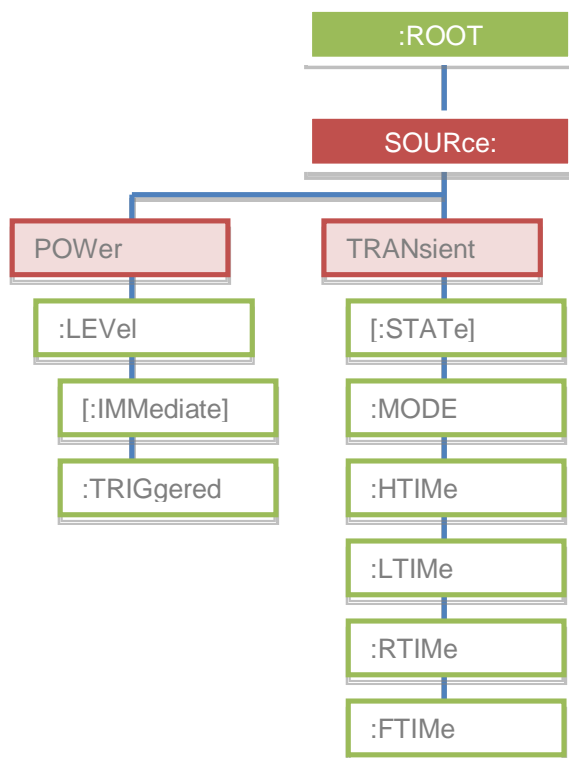
#### \*WAI

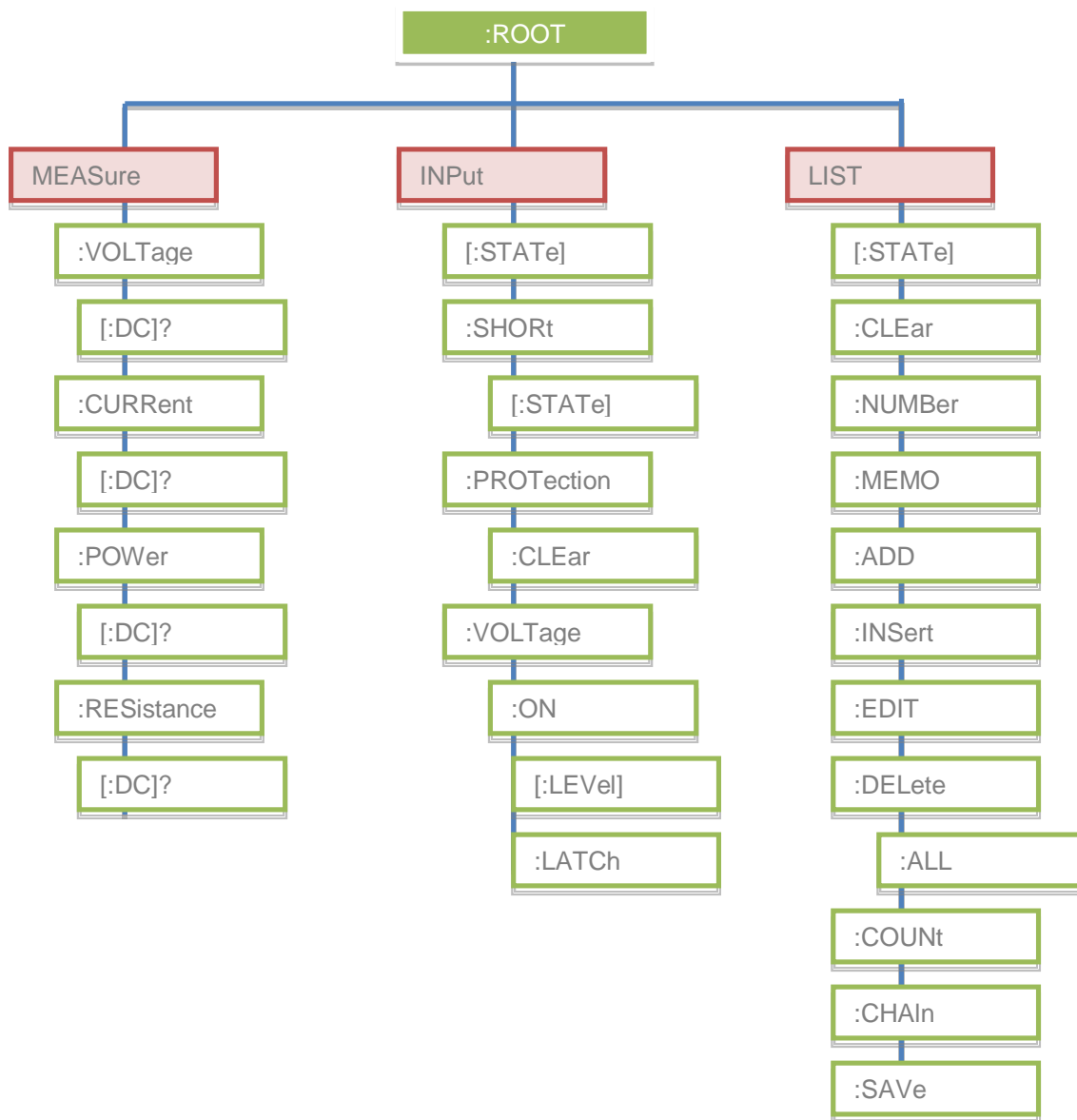
本コマンドは現在待機中の動作が終了するまで、現在送ったコマンドのプロセスを行わないようにします。すべての待機中の動作は、待機中のトリガー動作の終了も含まれます。\*WAI の実行を停止する場合は DCL(デバイスクリアー)のコマンドを送信します。

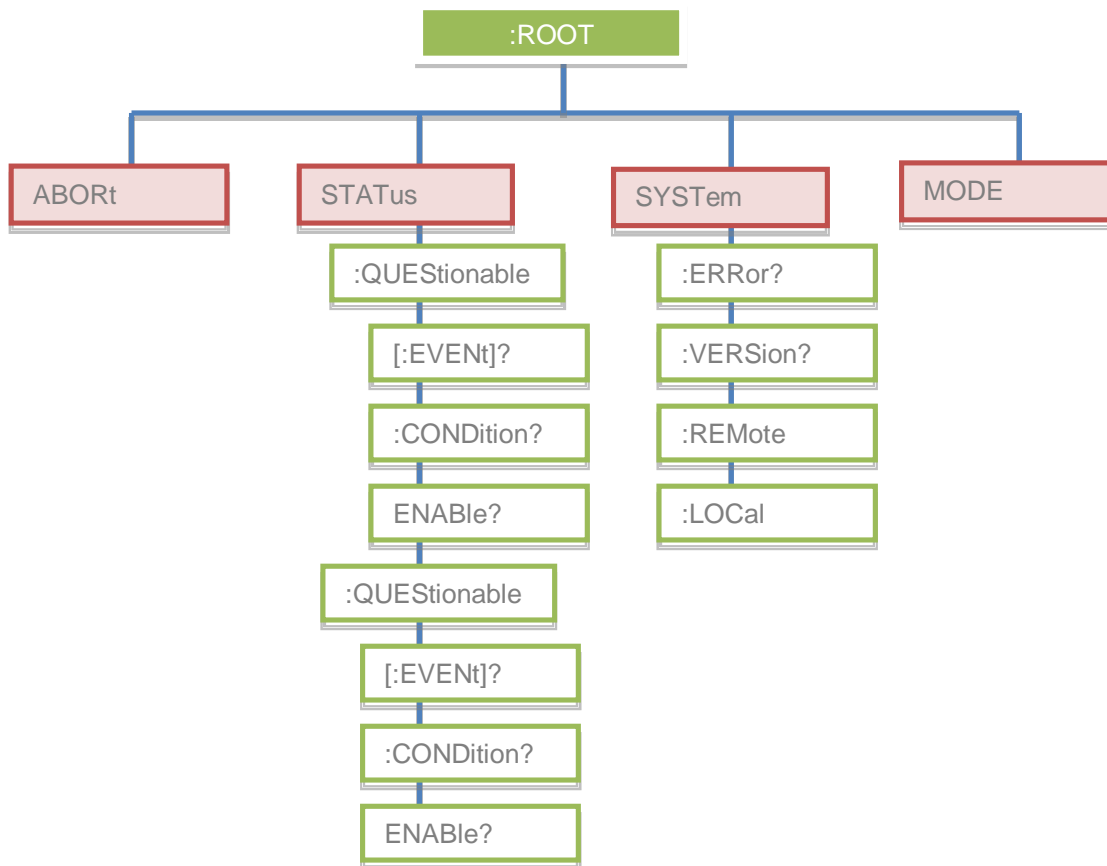
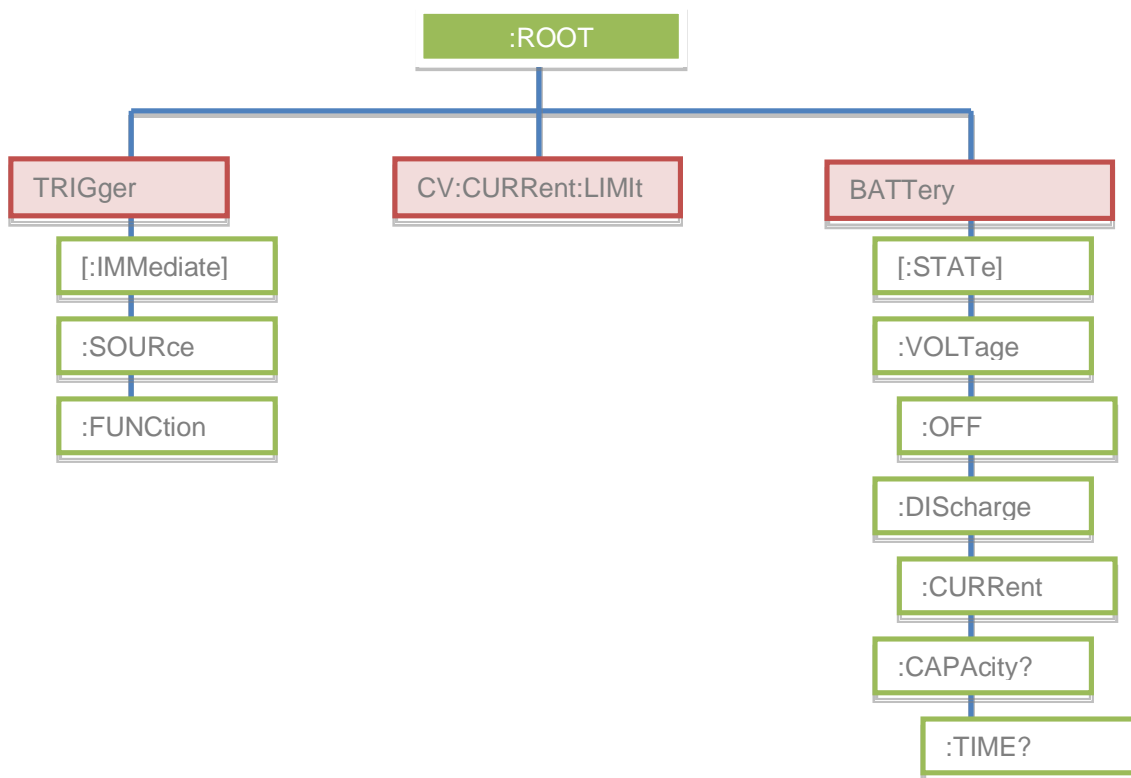
コマンドの構文 : \*WAI  
変数 : 無し  
関連コマンド : \*OPC、\*OPC?

2.2 SCPI ルートコマンド









## 2.2.1 CURRent サブシステム

### [SOURce:]CURRent[:LEVel][IMMediate]

負荷装置に対しすぐに実行する電流入力レベルを定義します。CURR? MIN と CURR? MAX は現在のレンジにおける最小及び最大入力電流レベルを戻します。設定レベルがレンジを超えている場合エラーが発生します。

コマンド構文	: [SOURce:]CURRent[:LEVel][IMMediate] <NRf+>
変数	: 数値 MIN MAX
単位	: A
例	: CURR 5、CURR 50mA
クエリーの構文	: [SOURce:]CURRent[:LEVel][IMMediate]?
変数	: 無し MIN MAX
戻り変数	: <NR3>
例	: CURR?、CURR? MIN、CURR? MAX
関連コマンド	: MODE

### [SOURce:]CURRent[:LEVel]:TRIGgered

電流のトリガーレベルを定義します。現在の電流入力で動作している電流レンジに同じように設定されます。電流トリガーレベルはトリガー信号が受け取られたときのみ、設定電流レベルになります。

コマンド構文	: [SOURce:]CURRent[:LEVel]:TRIGgered <NRf+>
変数	: 数値 MIN MAX
単位	: A
例	: CURR:TRIG 5、CURR:TRIG 50mA
クエリーの構文	: [SOURce:]CURRent[:LEVel]:TRIGgered?
変数	: 無し MIN MAX
戻り変数	: <NR3>
例	: CURR:TRIG?、CURR:TRIG? MIN、CURR:TRIG? MAX
関連コマンド	: MODE、TRIG

### [SOURce:]CURRent:PROTection[:LEVel]

負荷装置に対しソフトウェアによる電流保護制限の設定値を定義します。電流保護制限のレンジは 0-40A です。電流保護が使用可能な状態のとき、実際の電流が電流保護制限の設定点に達すると、負荷装置は保護動作を開始します。入力電流が制限値に達すると、その瞬間から保護動作までである一定時間そのままの状態を維持します。そしてその後負荷は停止します。

コマンド構文	: [SOURce:]CURRent:PROTection[:LEVel] <NRf+>
変数	: 数値 MIN MAX
単位	: A
例	: CURR:PROT 15
クエリーの構文	: [SOURce:]CURRent:PROTection[:LEVel]?
変数	: 無し MIN MAX
戻り変数	: <NR3>
例	: CURR:PROT?, CURR:PROT? MIN, CURR:PROT? MAX
関連コマンド	: INP:PROT:CLE

#### [SOURce:]CURRent:PROTection:DELaY

負荷装置に対しソフトウェアによる電流保護制限の起動までの時間を設定します。入力電流が制限値に達すると、その瞬間から保護動作までである一定時間そのままの状態を維持します。そしてその後負荷は停止します。

コマンド構文	: [SOURce:]CURRent:PROTection:DELaY <NRf+>
変数	: 数値 MIN MAX
単位	: s
例	: CURR:PROT:DEL 0.5
クエリーの構文	: [SOURce:]CURRent:PROTection:DELaY?
変数	: 無し MIN MAX
例	: CURR:PROT:DEL?;CURR:PROT:DEL? MIN;CURR:PROT:DEL? MAX
戻り変数	: <NR3>
関連コマンド	: INP:PROT:CLE

#### [SOURce:]CURRent:PROTection:STATe

電流保護制限の機能を使用するかしないかを設定します。

コマンド構文	: [SOURce:]CURRent:PROTection:STATe <bool>
項目	: ON OFF
例	: CURR:PROT:STAT ON
クエリーの構文	: [SOURce:]CURRent:PROTection:STATe?
変数	: 無し
例	: CURR:PROT:STAT?
戻り変数	: <NR1> 値 : 0 は OFF、1 は ON です。

関連コマンド : INP:PROT:CLE

#### [SOURce:]CURRent:LLEVel

過渡電流のローレベルでの電流を設定します。現在の電流設定にすぐに反映されます、

コマンド構文 : [SOURce:]CURRentc:LLEVel <NRf+>  
項目 : 数値|MIN|MAX  
単位 : A  
例 : CURR:LLEVel 3  
クエリーの構文 : [SOURce:]CURRent:LLEVel?  
変数 : 無し|MIN|MAX  
例 : CURR:LLEV?, CURR:LLEV? MIN、CURR:LLEV:? MAX  
戻り変数 : <NR3>  
関連コマンド : TRANSient 過渡応答の項を参照してください。  
[SOUR:]CURR:HLEVel

#### [SOURce:]CURRent:HLEVel

過渡電流のハイレベル電流を設定します。現在の電流設定にすぐに反映されます。

コマンド構文 : [SOURce:]CURRentc:HLEVel <NRf+>  
項目 : 数値|MIN|MAX  
単位 : A  
例 : CURR:HLEVel 3  
クエリーの構文 : [SOURce:]CURRent:HLEVel?  
変数 : 無し|MIN|MAX  
例 : CURR:HLEV?, CURR:HLEV? MIN、CURR:HLEV:? MAX  
戻り変数 : <NR3>  
関連コマンド : TRANSient 過渡応答の項を参照してください。

#### [SOURce:]CURRent:RISE:RATE

電流モードの上昇過渡時間を設定します。CURR:RISE:RATE? MIN および CURR:RISE:RATE? MAX は設定可能な最小値と最大値を戻します。

コマンド構文 : [SOURce:]CURRentc:RISE:RATE <NRf+>  
項目 : 数値|MIN|MAX  
単位 : A

例 : CURR:RISE:RATE 3  
クエリーの構文 : [SOURce:]CURRent:RISE:RATE?  
変数 : 無し|MIN|MAX  
例 : CURR:RISE:RATE?、CURR:RISE:RATE? MIN、CURR:RISE:RATE? MAX  
戻り変数 : <NR3>  
関連コマンド : MODE、CURR

#### **[SOURce:]CURRent:FALL:RATE**

電流モードの下降過渡時間を設定します。CURR:FALL:RATE? MIN および CURR:FALL:RATE? MAX は設定可能な最小値と最大値を戻します。

コマンド構文 : [SOURce:]CURRent:FALL:RATE <NRf+>  
項目 : 数値|MIN|MAX  
単位 : A  
例 : CURR:FALL:RATE 3  
クエリーの構文 : [SOURce:]CURRent:FALL:RATE?  
変数 : 無し|MIN|MAX  
例 : CURR:FALL:RATE?、CURR:FALL:RATE? MIN、CURR:FALL:RATE? MAX  
戻り変数 : <NR3>  
関連コマンド : MODE、CURR

## 2.2.2 VOLTage サブシステム

### [SOURce:]VOLTage[:LEVel][IMMEDIATE]

負荷装置に対しすぐに実行する電圧入力レベルを定義します。VOLT? MIN と VOLT? MAX は現在のレンジにおける最小及び最大入力電圧レベルを戻します。設定レベルがレンジを超えている場合エラーが発生します。

コマンド構文	: [SOURce:]VOLTage[:LEVel][IMMEDIATE] <NRf+>
変数	: 数値 MIN MAX
単位	: V
例	: VOLT 5
クエリーの構文	: [SOURce:]VOLTage[:LEVel][IMMEDIATE]?
変数	: 無し MIN MAX
戻り変数	: <NR3>
例	: VOLT?, VOLT? MIN, VOLT? MAX
関連コマンド	: MODE

### [SOURce:]VOLTage[:LEVel]:TRIGgered

電圧のトリガーレベルを定義します。現在の電圧入力で動作している電圧レンジに同じように設定されます。電圧トリガーレベルはトリガー信号が受け取られたときのみ、設定電圧レベルになります。

コマンド構文	: [SOURce:]VOLTage[:LEVel]:TRIGgered <NRf+>
変数	: 数値 MIN MAX
単位	: V
例	: VOLT:TRIG 5
クエリーの構文	: [SOURce:]VOLTage[:LEVel]:TRIGgered?
変数	: 無し MIN MAX
戻り変数	: <NR3>
例	: VOLT:TRIG?, VOLT:TRIG? MIN, VOLT:TRIG? MAX
関連コマンド	: MODE、TRIG

### [SOURce:]VOLTage:LLEVel

過渡電圧のローレベルでの電圧を設定します。現在の電圧設定にすぐに反映されます、

コマンド構文	: [SOURce:]VOLTage:LLEVel <NRf+>
項目	: 数値 MIN MAX
単位	: V

例 : VOLT:LLEVel 3  
 クエリーの構文 : [SOURce:]VOLTage:LLEVel?  
 変数 : 無し|MIN|MAX  
 例 : VOLT:LLEV?, VOLT:LLEV? MIN、VOLT:LLEV:? MAX  
 戻り変数 : <NR3>  
 関連コマンド : TRANsient 過渡応答の項を参照してください。  
                   [SOUR:]VOLT:HLEVel

### [SOURce:]VOLTage:HLEVel

過渡電圧のハイレベル電圧を設定します。現在の電圧設定にすぐに反映されます。

コマンド構文 : [SOURce:]VOLTage:HLEVel <NRf+>  
 項目 : 数値|MIN|MAX  
 単位 : V  
 例 : VOLT:HLEVel 3  
 クエリーの構文 : [SOURce:]VOLTage:HLEVel?  
 変数 : 無し|MIN|MAX  
 例 : VOLT:HLEV?, VOLT:HLEV? MIN、VOLT:HLEV:? MAX  
 戻り変数 : <NR3>  
 関連コマンド : TRANsient 過渡応答の項を参照してください。

### 2.2.3 RESistance サブシステム

#### [SOURce:]RESistance[:LEVel][:IMMediate]

負荷装置に対しすぐに実行する抵抗レベルを定義します。RES? MIN と RES? MAX は現在のレンジにおける最小及び最大入力レベルを戻します。設定レベルがレンジを超えている場合エラーが発生します。

コマンド構文 : [SOURce:]RESistance[:LEVel][:IMMediate] <NRf+>  
 変数 : 数値|MIN|MAX  
 単位 : OHM  
 例 : RES 5  
 クエリーの構文 : [SOURce:]RESistance[:LEVel][:IMMediate]?  
 変数 : 無し|MIN|MAX  
 戻り変数 : <NR3>  
 例 : RES?, RES? MIN、RES? MAX  
 関連コマンド : MODE

### **[SOURce:]RESistance[:LEVel]:TRIGgered**

抵抗のトリガーレベルを定義します。現在の抵抗で動作している抵抗レンジに同じように設定されます。抵抗トリガーレベルはトリガー信号が受け取られたときのみ、設定抵抗レベルになります。

コマンド構文	: [SOURce:]RESistance[:LEVel]:TRIGgered <NRf+>
変数	: 数値 MIN MAX
単位	: OHM
例	: RES:TRIG 5
クエリーの構文	: [SOURce:]RESistance[:LEVel]:TRIGgered?
変数	: 無し MIN MAX
戻り変数	: <NR3>
例	: RES:TRIG?, RES:TRIG? MIN, RES:TRIG? MAX
関連コマンド	: MODE、TRIG

### **[SOURce:]RESistance:LLEVel**

過渡抵抗のローレベルでの抵抗を設定します。現在の抵抗設定にすぐに反映されます、

コマンド構文	: [SOURce:]RESistancec:LLEVel <NRf+>
項目	: 数値 MIN MAX
単位	: OHM
例	: RES:LLEVel 3
クエリーの構文	: [SOURce:]RESistance:LLEVel?
変数	: 無し MIN MAX
例	: RES:LLEV?, RES:LLEV? MIN, RES:LLEV:? MAX
戻り変数	: <NR3>
関連コマンド	: TRANsient 過渡応答の項を参照してください。 [SOUR:]RES:HLEVel

### **[SOURce:]RESistance:HLEVel**

過渡抵抗のハイレベル抵抗を設定します。現在の抵抗設定にすぐに反映されます。

コマンド構文	: [SOURce:]RESistancec:HLEVel <NRf+>
項目	: 数値 MIN MAX
単位	: OHM
例	: RES:HLEVel 3

クエリーの構文 : [SOURce:]RESistance:HLEVel?  
変数 : 無し|MIN|MAX  
例 : RES:HLEV?、RES:HLEV? MIN、RES:HLEV:? MAX  
戻り変数 : <NR3>  
関連コマンド : TRANsient 過渡応答の項を参照してください。

#### 2.2.4 POWer サブシステム

##### [SOURce:]POWer[:LEVel][IMMediate]

負荷装置に対しすぐに実行する電力入力レベルを定義します。POW? MIN と POW? MAX は現在のレンジにおける最小及び最大入力電力レベルを戻します。設定レベルがレンジを超えている場合エラーが発生します。

コマンド構文 : [SOURce:]POWer[:LEVel][:IMMediate] <NRf+>  
変数 : 数値|MIN|MAX  
単位 : W  
例 : POW 5、POW 10mW  
クエリーの構文 : [SOURce:]POWer[:LEVel][:IMMediate]?  
変数 : 無し|MIN|MAX  
戻り変数 : <NR3>  
例 : POW?、POW? MIN、POW? MAX  
関連コマンド : MODE

##### [SOURce:]POWer[:LEVel]:TRIGgered

電力のトリガーレベルを定義します。現在の電力入力で動作している電力レンジに同じように設定されます。電力トリガーレベルはトリガー信号が受け取られたときのみ、設定電力レベルになります。

コマンド構文 : [SOURce:]POWer[:LEVel]:TRIGgered <NRf+>  
変数 : 数値|MIN|MAX  
単位 : W  
例 : POW:TRIG 5  
クエリーの構文 : [SOURce:]POWer[:LEVel]:TRIGgered?  
変数 : 無し|MIN|MAX  
戻り変数 : <NR3>  
例 : POW:TRIG?、POW:TRIG? MIN、POW:TRIG? MAX  
関連コマンド : MODE、TRIG

## 2.2.5 TRANsient サブシステム

### [SOURce:]TRANsient[:STATe]

過渡試験モードを ON あるいは OFF します。

コマンド構文	: TRANsient[:STATe] <bool>
変数	: ON OFF
例	: TRANsient ON
クエリーの構文	: TRANsient[:STATe]?
戻り変数	: <NR1>, 0= OFF, 1= ON

### [SOURce:]TRANsient:MODE

過渡試験モードを設定します。モードは3つあります。

CONTInuous 連続モード : トリガーで過渡試験を開始し、再度トリガーを送信するまで連続で動作

PULSe パルスモード(単発) : 一回のトリガーで一回のみ動作

TOGGle トグルモード : これはトグルスイッチのような動きを行うモードです。トリガーで上昇 : スイッチオン状態、再度トリガーを送信すると下降 : スイッチオフ状態となります。

コマンド構文	: [SOURce:]TRANsient:MODE <aard>
変数	: CONTInuous PULSe TOGGle
例	: TRANsient:MODE PULSe
クエリーの構文	: TRANsient:MODE?
戻り変数	: <aard>CONT,PULS,TOGG
関連コマンド	: TRIG

### [SOURce:]TRANsient:HTIME

過渡試験モードでハイレベルの持続時間を設定します。

コマンド構文	: [SOURce:]TRANsient:HTIME <NRf+>
変数	: digit MIN MAX
単位	: s
例	: TRANsient:HTIME 500ms
クエリーの構文	: TRANsient:HTIME?
変数	: 無し MIN MAX
例	: TRAN:HTIME?, TRAN:HTIME? MIN, TRAN:HTIME? MAX
戻り変数	: <NR3>

### [SOURce:]TRANsient:LTIMe

過渡試験モードでローレベルの持続時間を設定します。

コマンド構文	: [SOURce:]TRANsient:LTIMe <NRf+>
変数	: digit MIN MAX
単位	: s
例	: TRANsient:LTIMe 500ms
クエリーの構文	: TRANsient:LTIMe?
変数	: 無し MIN MAX
例	: TRAN:LTIMe?, TRAN:LTIMe? MIN, TRAN:LTIMe? MAX
戻り変数	: <NR3>

### [SOURce:]TRANsient:RTIMe

過渡試験モードでローレベルのエッジからハイレベルのエッジへの上昇時間を設定します。

コマンド構文	: [SOURce:]TRANsient:RTIMe <NRf+>
変数	: digit MIN MAX
単位	: s
例	: TRANsient:RTIMe 500ms
クエリーの構文	: TRANsient:RTIMe?
変数	: 無し MIN MAX
例	: TRAN:RTIMe?, TRAN:RTIMe? MIN, TRAN:RTIMe? MAX
戻り変数	: <NR3>

### [SOURce:]TRANsient:FTIMe

過渡試験モードでハイレベルのエッジからローレベルのエッジへの下降時間を設定します。

コマンド構文	: [SOURce:]TRANsient:FTIMe <NRf+>
変数	: digit MIN MAX
単位	: s
例	: TRANsient:FTIMe 500ms
クエリーの構文	: TRANsient:FTIMe?
変数	: 無し MIN MAX
例	: TRAN:FTIMe?, TRAN:FTIMe? MIN, TRAN:FTIMe? MAX
戻り変数	: <NR3>

## 2.2.7 MEASure

### MEASure:VOLTage[:DC]?

本クエリーは負荷装置に入力されている電圧値を戻します。

コマンド構文	: MEASure:VOLTage[:DC]?
変数	: 無し
戻り変数	: <NR3>
例	: MEAS:VOLT?

### MEASure:CURRent[:DC]?

本クエリーは負荷装置に入力されている電流値を戻します。

コマンド構文	: MEASure:CURRent[:DC]?
変数	: 無し
戻り変数	: <NR3>
例	: MEAS:CURR?

### MEASure:POWer[:DC]?

本クエリーは負荷装置に入力されている電力値を戻します。

コマンド構文	: MEASure:POWer[:DC]?
変数	: 無し
戻り変数	: <NR3>
例	: MEAS:POW?

### MEASure:RESistance[:DC]?

本クエリーは負荷装置に設定されている抵抗値を戻します。

コマンド構文	: MEASure:RESistance[:DC]?
変数	: 無し
戻り変数	: <NR3>
例	: MEAS:RES?

## 2.2.8 INPut サブシステム

### INPut[:STATe]

負荷装置への入力の ON/OFF を実行します。

コマンド構文	: INPut[:STATe] <bool>
変数	: ON OFF
例	: INP ON
クエリーの構文	: INPut[:STATe]?
変数	: 無し
例	: INP?
戻り変数	: <NR1> 0 = OFF、1 = ON
関連コマンド	: INP:SHOR、CURR:PROT

### INPut:SHORt[:STATe]

負荷装置を短絡モードに設定します。

コマンド構文	: INPut:SHORt[:STATe] <bool>
変数	: ON OFF
例	: INP:SHOR ON
クエリーの構文	: INPut:SHORt[:STATe]?
変数	: 無し
例	: INP:SHOR?
戻り変数	: <NR1> 0 = OFF、1 = ON
関連コマンド	: INP:STAT

### INPut:PROTection:CLEAr

負荷装置の保護状態(OC,OV,OP,OT,RV) を解除します。

コマンド構文	: INPut:PROTection:CLEAr
変数	: 無し
例	: INP:PROT:CLE
クエリーの構文	: 無し
関連コマンド	: CURR:PROT[:LEV]、CURR:PROT:DEL、CURR:PROT:STAT

### INPut:VOLTage:ON

負荷装置の Von 設定を行います。

コマンド構文	: INPut:VOLTage:ON <NRf+>
変数	: 数値 MIN MAX
単位	: V
例	: INPut:VOLTage:ON 10
クエリーの構文	: INPut:VOLTage:ON?
例	: INP:VOLT:ON?
戻り変数	: <NR3>
関連コマンド	: INP:VOLT:ON:LATC

### INPut:VOLTage:ON:LATCh

負荷装置の Von 設定のラッチを設定します。ラッチを掛けますと、Von 設定値を下回っても負荷動作を継続しますが、ラッチが掛かってない場合は、負荷装置は自動的に入力を切ります。また Von 設定値に入力電圧が戻れば自動的に負荷は ON します。

コマンド構文	: INPut:VOLTage:ON:LATCh <bool>
変数	: ON OFF
単位	: 無し
例	: INP:VOLT:ON:LATC ON
クエリーの構文	: INPut:VOLTage:ON:LATCh?
変数	: 無し
例	: INP:VOLT:ON:LATC?
戻り変数	: <NR1> 0 = OFF、1 = ON
関連コマンド	: INP:VOLT:ON

## 2.2.9 LIST サブシステム

### LIST[:STATe]

リストモードの ON / OFF を設定します。

コマンド構文	: LIST[:STATe] <bool>
変数	: on off
例	: LIST[:STATe] on、LIST[:STATe] off
クエリーの構文	: LIST[:STATe]?
変数	: 無し
例	: LIST?
戻り変数	: <NR1> 0 = OFF、1 = ON
関連コマンド	: 無し

### LIST:CLEAr

リストデータを初期化します。

コマンド構文	: LIST:CLEAr
変数	: 無し
例	: LIST:CLE
クエリーの構文	: 無し

### LIST:NUMber

リスト動作をさせたいリストの番号を設定します。

コマンド構文	: LIST:NUMber <NR1>
変数	: 0 ~ 6
例	: LIST:NUMber 0
クエリーの構文	: 無し

## LIST:MEMO

リストにメモをつけます。

コマンド構文 : LIST:MEMO "<aard>"  
変数 : 0x20 ~ 0x7f  
例 : LIST:MEMO "ARRAY"  
クエリーの構文 : LIST:MEMO?  
戻り変数 : <aard>

## LIST:ADD

新しいステップを追加します。それぞれのステップの値には必ず動作する負荷モード、負荷の値、動作時間を入力します。

コマンド構文 : LIST:ADD <aard>, <NRf>, <NRf>  
変数 : CCL|CCH|CRL|CRM|CRH|CV, 数値|MIN|MAX, 数値|MIN|MAX  
単位 : 無し|A|V|Ω|s(秒)  
例 : LIST:ADD cch, 1A, 1s  
クエリーの構文 : 無し

## LIST:INSert

リストのあるステップの場所に新しいステップを挿入します。

コマンド構文 : LIST:INSert <step>, <aard>, <NRf>, <NRf>  
変数 : 1 ~ 50, CCL|CCH|CRL|CRM|CRH|CV, 数値|MIN|MAX, 数値|MIN|MAX  
単位 : 無し|A|V|Ω|s(秒)  
例 : LIST:INSert 2, cch, 2A, 1s  
クエリーの構文 : 無し

## LIST:EDIT

リストのあるステップを編集します。

コマンド構文 : LIST:EDIT <step>, <aard>, <NRf>, <NRf>  
変数 : 1 ~ 50, CCL|CCH|CRL|CRM|CRH|CV, 数値|MIN|MAX, 数値|MIN|MAX  
単位 : 無し|A|V|Ω|s(秒)  
例 : LIST:EDIT 2, cch, 3A, 1s  
クエリーの構文 : 無し

### LIST:DELeTe

リストのあるステップを削除します。

コマンド構文	: LIST:DELeTe <step>
変数	: 1 ~ 50
単位	: 無し
例	: LIST:DELeTe 2

### LIST:DELeTe:ALL

リスト動作のためのすべてのステップを削除します。

コマンド構文	: LIST:DELeTe:ALL
変数	: 無し
例	: LIST:DELeTe:ALL

### LIST:COUNt

リスト動作の繰り返し回数を設定します。

コマンド構文	: LIST:COUNt <NR1>
変数	: 0 ~ 65535
単位	: 無し
例	: LIST:COUNt 10
クエリー構文	: LIST:COUNt?
戻り変数	: <NR1>
関連コマンド	: 無し

### LIST:CHAIn

現在のリストに対し結合設定：ステップをステップ番号に従って連続動作するようにします。

コマンド構文	: LIST:CHAIn <NR1> <aard>
変数	: 0 ~ 6 OFF
単位	: 無し
例	: LIST:CHAIn 1

### LIST:SAVe

現在のリストを不揮発性メモリーに保存します。

コマンド構文 : LIST:SAVe  
変数 : 無し

## 2.2.10 TRIGger サブシステム

### TRIGger[:IMMediate]

このコマンドは負荷装置に対し、トリガースourceがどのような設定であれトリガー信号を送信します。この機能は\*TRG コマンドと同じです。

コマンド構文 : TRIGger[:IMMediate]  
変数 : 無し  
関連コマンド : TRIG:SOURCE

### TRIGger:SOURCE

負荷のトリガースourceを設定します。負荷装置にはバス、外部、ホールドの三つのトリガースourceがあります。

BUS : GPIB の<GET>信号あるいは\*TRG コマンドをトリガースourceとして使用します。  
EXTernal : 外部トリガー入力端子を選択する、あるいはフロントパネルの 2nd ボタン +Trigger ボタンをトリガースourceとして使用します。外部信号入力端子への入力信号は TTL レベルで、フォールエッジトリガーとなっています。  
HOLD : TRIGger:IMMediate コマンドのみがトリガーとして受け付けられます。それ以外は\*TRG を含め受け付けません。

コマンド構文 : TRIGger:SOURCE <aard>  
変数 : BUS|EXTernal|HOLD  
クエリー構文 : TRIGger:SOURCE?  
戻り変数 : <aard> BUS|EXTernal|HOLD

### TRIGger:FUNCTION

トリガー制御を行いたいモードを選択します。選択対象は過渡試験モードあるいはリスト機能となります。

コマンド構文 : TRIGger:FUNCTION <aard>  
変数 : Tran|List  
クエリー構文 : TRIGger:FUNCTION?  
戻り変数 : <aard>Tran|List

2.2.11 CV:CURRent:LIMIt 電圧モードにおける入力電流制限  
電圧モードにおいて入力する電流を定義します。

コマンドの構文 : CV:CURRent:LIMIt <NRf+>  
変数 : 数値:MIN:MAX  
単位 : A  
例 : CV:CURRent:LIMIt 20A  
クエリー構文 : CV:CURRent:LIMIt?  
変数 : 無し|MIN:MAX  
例 : CV:CURRent:LIMIt? CV:CURRent:LIMIt? MIN CV:CURRent:LIMIt? MAX  
戻り変数 : <NR3>

2.2.12 BATTery サブシステム

#### **BATTery[:STATe]**

本コマンドはバッテリー放電動作モードの可 / 不可を設定します。

コマンド構文 : BATTery[:STATe] <bool>  
変数 : on|off  
例 : BATTery[:STATe] on、 BATTery[:STATe] off  
クエリーの構文 : BATTery[:STATe]?  
変数 : 無し  
例 : BATT?  
戻り変数 : <NR1> 0 = OFF、 1 = ON  
関連コマンド : 無し

#### **BATTery:VOLTage:OFF**

放電動作において、終止電圧を設定します。

コマンド構文 : BATTery:VOLTage:OFF <NRf+>  
変数 : 数値|MIN|MAX  
単位 : V  
クエリーの構文 : BATTery:VOLTage:OFF?  
変数 : 無し|MIN|MAX  
例 : BATT:VOLT:OFF? BATT:VOLT:OFF? MAX BATT:VOLT:OFF? MIN?  
戻り変数 : <NR3>  
関連コマンド : 無し

### BATTery:DISCharge:CURRent

放電動作における放電電流値を設定します。

コマンド構文	: BATTery:DISCharge:CURRent <NRf+>
変数	: 数値 MIN MAX
単位	: A
クエリーの構文	: BATTery:DISCharge:CURRent?
変数	: 無し MIN MAX
例	: BATT:DIS:CURR? BATT:DIS:CURR? MAX BATT:DIS:CURR? MIN?
戻り変数	: <NR3>
関連コマンド	: 無し

### BATTery:CAPAcity?

本クエリーは放電電流量を戻します。

コマンド構文	: BATTery:CAPAcity?
変数	: 無し
クエリーの構文	: BATT:CAPA?
戻り変数	: <NR3>
関連コマンド	: 無し

### BATTery:TIME?

本クエリーは放電時間を戻します。

コマンド構文	: BATTery:TIME?
変数	: 無し
クエリーの構文	: BATT:TIME?
戻り変数	: <NR3>
関連コマンド	: 無し

### 2.2.13 ABORt

本コマンドはすべての待機中の[:LEVel]:TRIG(例として CURR:TRIG 構文に含まれます)の動作を取り消します。その結果、後続のトリガーは入力レベルでは全く影響を受けません。ABORt は過渡試験そしてリスト動作には影響を与えません。

コマンドの構文 : ABORt  
変数 : 無し  
例 : ABOR  
クエリー構文 : 無し  
関連コマンド : CURR[:LEV]:TRIG、VOLT[:LEV]:TRIG、RES[:LEV]:TRIG、  
STAT:OPER:COND?

### 2.2.14 STATus サブシステム

#### **STATus:QUEStionable[:EVENT]?**

本クエリーはクエスチョナブル・ステイタス・イベント・レジスターを戻します。戻り値は 10 進数で表現され、値はすべての 2 進数の値の総和となっています。本レジスターへのクエリーを実行しますと、レジスターは 0 にリセットされます。数値の詳細については 4 章のステイタス・リポーティングを参照します。

コマンドの構文 : STATus:QUEStionable[:EVENT]?  
変数 : 無し  
クエリー構文 : STAT:QUES?  
戻り変数 : <NR1>

#### **STATus:QUEStionable:CONDition?**

本クエリーはクエスチョナブル・ステイタス・コンディション・レジスターの値を戻します。それぞれの 0 と 1 の値はクエスチョナブル・ステイタス・イベント・レジスターのビットに対応しています。

コマンドの構文 : STATus:QUEStionable:CONDition?  
変数 : 無し  
クエリー構文 : STAT:QUES:COND?  
戻り変数 : <NR1>

### STATus:QUEStionable:ENABle

クエスチョナブル・ステイタス・イネイブル・レジスターのビットを設定します。クエスチョナブル・イベントレジスターにおける 0 もしくは 1 への変更に関わらず、プログラムされた変数はステイタス・バイト・レジスターの QUES のビットを設定します。詳細はステイタス・リポーティングを参照します。

コマンドの構文	: STATus:QUEStionable:ENABle <NRf>
変数	: 0 ~ 65535
電源投入時の値	: *PSC で参照します。
例	: STAT:QUES:ENAB 64
クエリー構文	: STAT:QUES:ENAB?
戻り変数	: <NR1>
関連コマンド	: *PSC

### STATus:OPERation[:EVENT]?

本クエリーはオペレーション・イベント・レジスターの値を戻します。戻した後レジスターは 0 にリセットされます。

コマンドの構文	: STATus:OPERation[:EVENT]?
変数	: 無し
戻り変数	: <NR1>

### STATus:OPERation:CONDition?

本クエリーはオペレーション・コンディション・レジスターの値を戻します。それぞれの 0 と 1 の状態値はオペレーション・イベント・レジスターのビットに対応しています。

コマンドの構文	: STATus:OPERation:CONDition?
変数	: 無し
戻り変数	: <NR1>

### STATus:OPERation:ENABLE

オペレーション・イネイブル・レジスターのビットを設定します。オペレーション・イベント・レジスターにおける 0 もしくは 1 への変更に関わらず、プログラムされた変数はステータス・バイト・レジスターの OPER のビットを設定します。詳細はステータス・リポーティングを参照します。

コマンドの構文	: STATus:OPERation:ENABLE <NRf>
変数	: 0 ~ 255
電源投入時の値	: *PSC で参照します。
例	: STAT:OPER:ENAB 128
クエリー構文	: STAT:OPER:ENAB?
戻り変数	: <NR1>
関連コマンド	: *PSC

### 2.2.15 SYStem

#### SYSTem:ERRor?

本負荷装置のエラーコードとエラーメッセージを戻します。フロントパネルのディスプレイに ERR が表示されたとき、一つあるいは複数のエラーが発生しています。最大 20 個のエラーが負荷装置に記録されます。ただし 20 を超えるエラーが発生した場合それは記録されません。

1. エラーキューは FIFO モードで保存されまた呼び出されます。最初に戻されるエラーメッセージは最初に発生したエラーとなります。すべてのエラーが戻されるとディスプレイから ERR の表示は消灯します。
2. エラーの記録が 20 を超えた場合、エラーキュー内の最後のエラーは 350 に置き換わります。これはエラーが 20 個を超えて発生していることを意味します。負荷装置は、エラーの消去あるいはキューからエラーを戻す操作をしない限り 20 を超えるエラーを記憶しません。
3. エラーが発生していない場合、No error が戻されます。\*CLS はエラーキュー内のエラーを消去しますが\*RST は消去できません。すべての記録されたエラーは負荷装置の電源が切られると消去されます。

コマンドの構文	: SYSTem:ERRor?
変数	: 無し
戻り変数	: <NR1>, <AARD>

### SYSTem:VERsion?

ソフトウェアのバージョン番号を戻します。例 V1.00

コマンドの構文 : SYSTem:VERsion?  
変数 : 無し  
戻り変数 : <AARD>, <NR2>

### SYSTem:REMOte

負荷装置をリモート制御に設定します。 GPIB ポートを経由して通信している場合はエラーメッセージが発生します。

コマンドの構文 : SYSTem:REMOte  
変数 : 無し  
戻り変数 : 無し  
関連コマンド : SYSTem:LOCAl

### SYSTem:LOCAl

負荷装置をローカル制御に設定します。 GPIB ポートを経由して通信している場合はエラーメッセージが発生します。

コマンドの構文 : SYSTem:LOCAl  
変数 : 無し  
戻り変数 : 無し  
関連コマンド : SYSTem:REMOte

## 2.2.16 MODE

### MODE

負荷装置の負荷モードを設定します。

CC モード : CCL CCH

CR モード : CRL CRM CRH

CV モード : CV

CP モード : CPC CPV

コマンドの構文 : MODE <AARD>

変数 : CCL|CCH|CRL|CRM|CRH|CV|CPC|CPV

例 : MODE CCL

クエリー構文 : MODE?

戻り変数 : <AARD> CCL|CCH|CRL|CRM|CRH|CV|CPC|CPV

### 3. エラーメッセージ

エラー番号	エラーの内容
-103	無効な分割です。 コマンド文の中に無効な分割の表記があります。コロン、セミコロンあるいはスペースの代わりにカンマを使用している、コロンのかわりにスペースを用いたりしている場合発生します。
-104	文字列が必要なところに数値が書き込まれている。あるいは数値が必要なところに文字が書き込まれています。
-108	複数の変数が受信されました。コマンドで用いることができる変数の数に直してください。変数を必要としないコマンドで変数が用いられています。
-113	定義されていないヘッダーです。コマンドの表記が間違えています。あるいはコマンドが無効です。
-123	指数が大きすぎます。
-131	無効な接尾辞です。接尾辞の表記が間違えています。
-170	表記が間違えています。 シングルクォーテーションで表記を開始しています。そしてダブルクォーテーションで表記が終わっています。
-210	GPIB では使用できないコマンドです。
-221	LIST がレンジの範囲を超えています。
-222	データがレンジを超えています。
-223	データが多すぎます。 負荷装置で文字列が受信されたとき、文字数が規定を超えているため実行不能です。
-330	自己診断試験が失敗しました。 一つあるいは複数のエラーがセルフテスト中に発生しています。-6XX の項目を参照してください。
-350	エラーが多すぎます。 20 を超えるエラーが発生しています。これはエラーキューが満杯の状態のため新しいエラーは記録できない状態です。負荷装置の電源を切り消去するか、*CLS を送信しエラーキュー内の記録を消去します。
-410	割り込みのクエリー 出力バッファにデータを送信するコマンドが受信されました。しかしデータは以前のコマンドにより送信されるデータが出力バッファにあります。 電源を切るか*RST コマンドを実行し出力バッファの中を消去します。
-440	クエリーが完了しない データをインターフェース経由で送信しているが、出力バッファにデータ送信コマンドが受信されていない。

#### 4. ステータス・レジスター・リポーティング

##### SCPI ステータス・レジスター

負荷装置には状態を記録するための4つのステータス・レジスターが備わっています。それらはステータス・バイト・レジスター、スタンダード・イベント・レジスター、クエスチョナブル・ステータス・レジスターそしてオペレーション・ステータス・レジスターです。

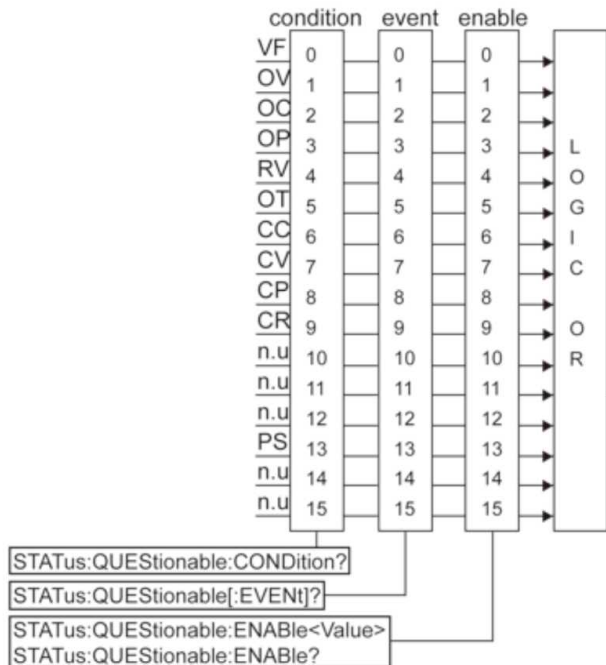
ステータス・バイト・レジスターはすべてのステータス・レジスターからすべてのステータス・イベントを要約します。すべてのステータス・レジスターを下記に示します。

ビット	信号	意味
		オペレーション・ステータス・レジスター
0	CAL	校正が実行されています。
1	WTG	トリガーを待っています。
		クエスチョナブル・ステータス・レジスター
0	VF	電圧異常 極性が反転しているあるいはか電圧状態が発生しているときに発生します。VF は INT:PROT:CLE が受信されるまで保持されます。
1	OV	過電圧 過電圧が発生すると、負荷装置は停止し、OV/OF が同時に設定されます。OV/OF は過電圧状態が無くなり、INT:PROT:CLE が受信されるまで保持されます。
2	OC	過電流 過電流が発生すると、OC の設定値に抑制されます。過電流状態が解除されるまでその状態は維持されます。過電流状態が定められた時間を経過すると保護状態(PS:入力遮断)へ移行し負荷装置の入力が遮断します。OC/PS は過電流状態が解消されるまで維持されます。また INT:PROT:CLE が負荷装置で受信されるまで維持されます。
3	OP	過電力 過電力が発生すると、負荷装置は停止し、OP/PS が同時に設定されます。OP/PS は過電力状態が無くなるまで維持され、INT:PROT:CLE が受信されるまで保持されます。
4	RV	入力の極性が反転しています。 極性が反転している状態が検知されると、RV/VF が設定されます。RV は極性が反転している状態が解消されるまで保持されます。VF は INT:PROT:CLE が受信されるまで保持されます。
5	OT	過熱保護 過熱状態が発生すると負荷装置は入力を遮断します。そして OT/PS は過熱時温度を下まわるまで保持されます。PS は INT:PROT:CLE が受信されるまで保持されます。
6	CC	定電流モード
7	CV	定電圧モード
8	CP	定電力モード
9	CR	定抵抗モード

13	PS	保護動作による入力遮断 負荷装置は過電流、過電力、過熱の状態が発生すると入力を遮断します。PS はは INT:PROT:CLE が受信されるまで保持されます。
		スタンダード・イベント・レジスター
0	OPC	動作完了 負荷装置はすべての待機中の動作を完了しています。*OPC を使用しますと負荷装置がすべての待機中動作を終了したときに、このビットが設定されます。
2	QYE	クエリーが不正 出力キューにデータが無い状態のとき発生します。エラーのレンジが-499 から-400 の範囲の場合 QYE を設定します。
3	DDE	装置に起因するエラー メモリーが消去された状態で発生します。エラーのレンジが-399 から-300 の範囲の場合 DDE を設定します。
4	EXE	実行エラー コマンドの変数が定められた範囲を逸脱している場合、矛盾している場合に設定します。あるいはコマンドが現在の状態では実行できない場合に設定します。エラーのレンジは-299 から-200 の範囲のとき EXE が設定されます。
5	CME	コマンドエラー 構文あるいは意味が通じていない場合に設定されます。エラーのレンジは-199 から-100 の範囲のとき CME が設定されます。
7	PON	装置電源が ON 負荷装置の電源が入っているとき設定されます。
		ステイタス・バイト・レジスター
3	QUES	クエスチョナブル イネイブル・クエスチョナブル・レジスターが有効な場合、QUES が設定されます。
4	MAV	メッセージの提供可 出力キューバッファがデータを含んでいる場合設定されます。
5	ESB	イベント・ステイタス・ビット スタンダード・イベント・イネイブル・レジスターが有効な場合、ESB が設定されます。
6	MSS RQS	シリアルポールを行っているとき、RQS(サービス要求)が戻され、消去されます。*STB?コマンドでは MSS(マスター・サマリー・ステイタス)は消去されずに戻されます。
7	OPER	オペレーション オペレーション・ステイタス・レジスターが有効にされている場合、OPER が設定されます。

ステータス・レジスタの体系

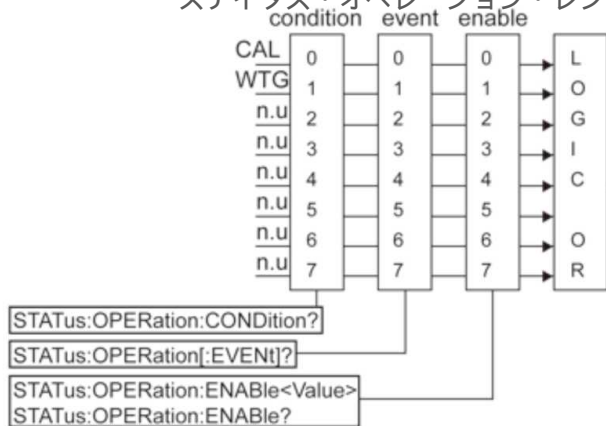
ステータス・クエシヨナブル・レジスタ



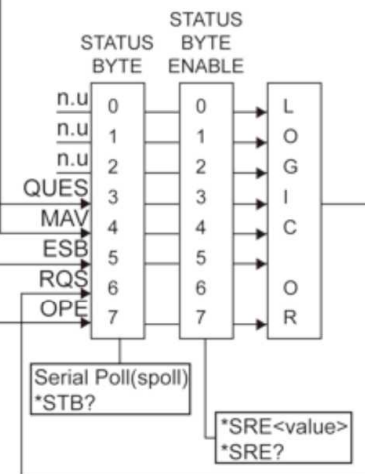
出力キューバッファ



ステータス・オペレーション・レジスタ



ステータス・バイト・レジスタ



スタンダード・イベント・レジスタ

