

Micsig-Oszilloskop:

Handbuch zu den SCPI-

Befehlen



*Gilt für hochauflösende Oszilloskope der Serien MHO6, MHO3, MHO1, MO3, MDO, ETO, STO, SATO, TO und ATO

Inhaltsverzeichnis

1.	Einführung in das Dokument	1
1.1	Zweck dieses Dokuments	1
1.2	Zielgruppe	1
1.3	Referenz.....	1
1.4	Begriffserläuterungen	1
2.	Produktbeschreibung.....	2
2.1	Hintergrund des Produkts.....	2
2.2	Zielgruppen des Produkts und Beschreibung des Bedarfs.....	2
2.3	Rollen im Produkt.....	2
2.4	Geschäftsablauf	2
2.5	Zielumgebung für Software und Hardware	3
2.6	Einschränkungen und Beschränkungen	3
2.7	Anwendbare Schnittstelle	3
3.	SCPI-Anforderungen.....	4
3.1	Einführung in SCPI	4
3.1.1	Befehlsformat	4

3.1.2 Erläuterung der Symbole.....	4
3.1.3 Parametertypen	5
3.1.4 Befehlsabkürzungen.....	5
3.2 Befehlssystem.....	6
3.2.1 Allgemeine Befehle	6
3.2.2 :MENU Menü-Funktionsbe	7
3.2.3 Befehlssystem für die Abtastung	15
3.2.4 Kanal-Befehlssystem.....	24
3.2.5 Mathematik-Befehlssystem	33
3.2.6 Cursor-Befehlssystem.....	50
3.2.7 Anzeigubsystem	61
3.2.8 Messbefehl-Subsystem	66
3.2.9 Trigger-Befehlssystem	79
3.2.10 Zeitbasis-Befehlssystem	114
3.2.11 Speicher-Befehlssystem.....	117
3.2.12 Befehlssystem für die Buskonfiguration	122
3.2.13 Befehlssystem für Referenzwellenformen	137

3.2.14	AUTO-Einstellungs-Subsystem	142
--------	-----------------------------------	-----

3. 2.15	Wellenform-Befehlssystem	147
---------	--------------------------------	-----

1. Einleitung

1.1 Zweck dieses Dokuments

Dieses Dokument dient dazu, die SCPI-Anforderungen für Oszilloskope zu definieren und die Voraussetzungen dafür zu schaffen, dass Micsig-Oszilloskope das SCPI-Protokoll unterstützen und dem Standard IEEE 488.2 entsprechen.

1.2 Zielgruppe

Entwickler und Tester

1.3 Referenzdokumentation

1.4 Begriffe

Abkürzungen und Begriffe	Erläuterung
SCPI	Handbuch zu Standardbefehlen für programmierbare Messgeräte
...	

2. Produktbeschreibung

2.1 Produkt-Hintergrund

Das SCPI-Befehlsverarbeitungsmodul ist in unsere Produkte integriert, um den Anforderungen des IEEE488.2-Standards zu entsprechen. Als SCPI-Gerät müssen wir es streng nach den Vorgaben des IEEE488.2-Standards für Messgeräte entwickeln.

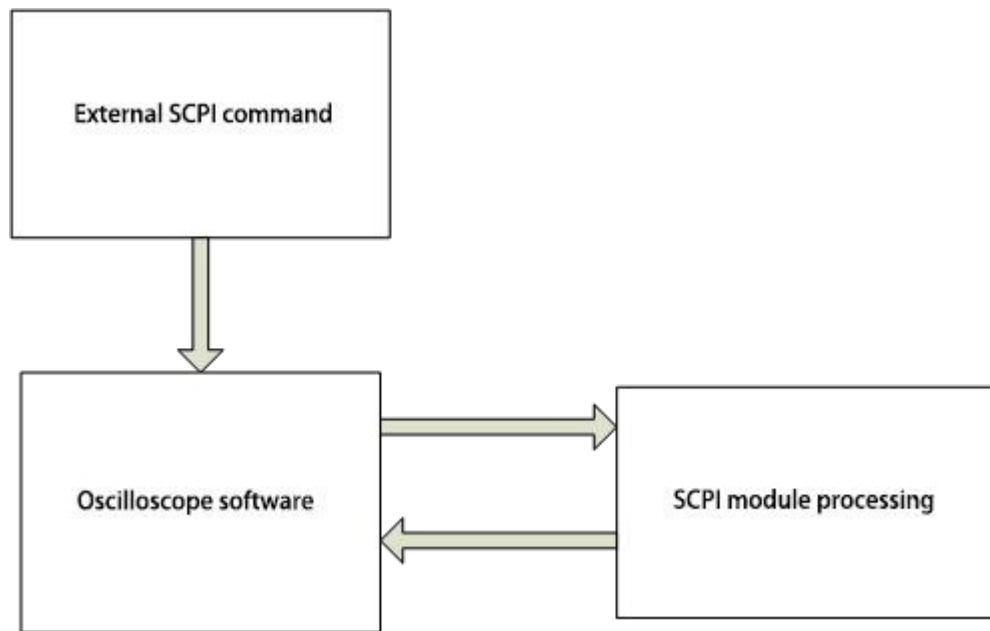
2.2 Zielgruppen und Anforderungsprofil

Das SCPI-Befehlsverarbeitungsmodul ist ausschließlich für die Software selbst bestimmt und dient der Verarbeitung aller SCPI-Befehle, die von außerhalb des Geräts an die Software gesendet werden.

2.3 Rollen im Produkt

Rollenbezeichnung	Aufgaben
Gemeinsames Befehlssystem	Verarbeitet allgemeine Befehle für alle Messgeräte
Grundlegendes Befehlssystem	Wesentliche Befehle zur Bedienung von Instrumenten und Geräten
Sonstiges	Optionale Befehle zur Bedienung der Messgeräte

2.4 Geschäftsablauf



2.5 Zielumgebung für Betriebssoftware und Hardware

Betriebssystem: Android

Hardwareumgebung: Tablet-Oszilloskop

2.6 Einschränkungen und Beschränkungen

Da es sich bei diesem Modul um ein Softwaremodul handelt, gelten dieselben

Einschränkungen und Beschränkungen wie für das Produkt.

2.7 Unterstützte Schnittstellen

USB, LAN, WLAN.

3. SCPI-Anforderungen

3.1 Einführung in SCPI

3.1.1 Befehlsformat

SCPI-Befehle sind baumartige hierarchische Strukturen, die mehrere Teilsysteme umfassen, von denen jedes aus einem Stamm-Schlüsselwort und einem oder mehreren hierarchischen Schlüsselwörtern besteht.

Die Befehlszeile beginnt in der Regel mit einem Doppelpunkt „:“; **Schlüsselwörter werden durch Doppelpunkte „:“ getrennt**, gefolgt von optionalen Parametereinstellungen; ein Fragezeichen „?“ wird hinter der Befehlszeile hinzugefügt, um eine Abfrage für diese Funktion anzuzeigen; Befehle und Parameter werden durch „Leerzeichen“ getrennt.

3.1.2 Erläuterung der Symbole

- 1、 große Klammern { }

Der Inhalt in geschweiften Klammern stellt die Parameteroptionen dar. Die einzelnen Parameter werden in der Regel durch einen senkrechten Strich „|“ voneinander getrennt. Bei der Verwendung eines Befehls muss einer der Parameter ausgewählt werden.

- 2、 Senkrechter Strich |

Der senkrechte Strich dient zur Trennung mehrerer Parameteroptionen. Bei der Verwendung eines Befehls müssen Sie einen der Parameter auswählen.

- 3、 Eckige Klammern []

Der Text in eckigen Klammern ist optional.

4、 Dreieckige Klammern < >

Die Parameter in den spitzen Klammern müssen durch einen gültigen Wert ersetzt werden.

3.1.3 Parametertypen

1、 Bool

Die Parameterwerte lauten „OFF“, „ON“, „0“ oder „1“.

2、 Diskret

Die Parameterwerte sind die aufgeführten Optionen.

3、 Ganzzahl

Sofern nicht anders angegeben, kann der Parameter eine beliebige ganze Zahl (NR1-Format) innerhalb des gültigen Wertebereichs sein. Bitte beachten Sie, dass Sie den Parameter zu diesem Zeitpunkt nicht auf das Dezimalformat setzen dürfen, da sonst eine Ausnahme auftritt.

4、 Reelle Zahl

Der Parameter kann eine beliebige reelle Zahl innerhalb des gültigen Wertebereichs sein.

Der Befehl akzeptiert Parametereingaben im Dezimalformat (NR2-Format) und in wissenschaftlicher Notation (NR3-Format).

5、 ASCII-Zeichenkette

Der Parameterwert besteht aus einer Kombination von ASCII-Zeichen.

3.1.4 Befehlsabkürzung

Bei allen Befehlen wird die Groß-/Kleinschreibung nicht berücksichtigt; sie können sowohl in Groß- als auch in Kleinbuchstaben geschrieben werden.

Wenn Sie jedoch eine Abkürzung verwenden möchten, müssen Sie den Befehl vollständig in Großbuchstaben eingeben.

3.2 Befehlssystem

3.2.1 Gängige Befehle

IDN

Funktion: Liest Informationen zum Oszilloskop aus. Enthält Versionsnummer, Hersteller, Produktmodell und Produktseriennummer.

Format: *IDN?

Rückgabeformat:

Micsig,<Modell>,<Seriennummer>,XXXXX

<Modell>: Gerätemodell.

<Seriennummer>: Seriennummer des Geräts.

XXXXX: Softwareversion des Instruments.

Beispiel:

Micsig,MDO5004,390000029,1.388.132

Funktion: Fragt den Datenstrom des aktuellen Oszilloskop-Bildschirminhalts ab.

Format::SYS:SCR?

Rückgabeformat: Die Abfrage gibt einen binären Datenstrom des Bildschirmaufnahmes im PNG-Format zurück. Die zurückgegebenen Daten bestehen aus drei Teilen: einer Kennung, der Datenlänge und dem Datenstrom der Bildschirmaufnahme.

Beispiel für Rückgabedaten:

#9000358370\FF\D8X\00\00\10JFIF\00\01\01\

In diesem Datenstrom ist # die Kennung, 9 gibt an, dass 9 Bytes zur Beschreibung der Datenlänge verwendet werden, 000358370 gibt die Länge des zurückgegebenen Datenstroms an, und die nachfolgenden Zeichen FF\D8X\00\00\10JFIF\00\01\01\ stellen den Screenshot-Datenstrom dar.

3.2.2 :MENU Menü-Funktionsbefehle

MENU

Die automatische Konfiguration kann das Oszilloskop schnell so einstellen, dass das Eingangssignal optimal dargestellt wird. Die automatische Konfiguration umfasst: anwendbar auf einen einzelnen Kanal und mehrere Kanäle; automatische Anpassung der horizontalen und vertikalen Signalskala sowie des Triggerpegels; die Wellenformumkehr des Oszilloskops ist deaktiviert, die Bandbreite ist auf die volle Bandbreite eingestellt, der Kopplungsmodus ist DC-Kopplung, der Abtastmodus ist normale Abtastung; der Trigger ist auf Flankentrigger eingestellt, der Triggermodus ist

automatisch.

Funktion: Starten oder Beenden der automatischen Einstellung (Auto-Range).

Die automatische Konfiguration ermöglicht eine schnelle Einstellung des Oszilloskops, um das Eingangssignal optimal darzustellen. Die automatische Konfiguration umfasst: anwendbar auf Einzel- und Mehrkanalbetrieb; automatische Anpassung der horizontalen und vertikalen Skalierung sowie des Triggerpegels; die Wellenformumkehr des Oszilloskops ist deaktiviert, die Bandbreite ist auf die volle Bandbreite eingestellt, der Kopplungsmodus ist Gleichstromkopplung, der Abtastmodus ist normale Abtastung; der Trigger ist auf Flankentrigger eingestellt, der Triggermodus ist automatisch.

Format: :MENU:AUTO <bool>

:MENU:AUTO?

Dabei ist <bool> ein boolescher Wert, der die Werte {{0|OFF}}|{{1|ON}} annehmen kann.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl schaltet die Anzeige von Kanal 1 ein.

:MENU:AUTO ON oder :MENU:AUTO 1

Die folgende Abfrage gibt „1“ zurück.

:MENU:AUTO?

MERN

Funktion: Das Oszilloskop wird in Betrieb gesetzt, die Triggerbedingungen werden erfüllt und die Datenerfassung beginnt.

Format: :MENU:RUN

~~MENSTOP~~

Funktion: Stoppt das Oszilloskop und die Datenerfassung.

Format: :MENU:STOP

~~MENSING~~

Funktion: Stellt das Oszilloskop auf Einzelmessung ein. Das Oszilloskop erfasst und zeigt eine einzelne Messung an.

Format: :MENU:SINGLE

~~MENRESET~~

Funktion: Setzt die Einstellungen des Oszilloskops auf die Werkseinstellungen zurück.

Format: :MENU:RESet

~~MENLOCK~~

Funktion: Schließt den Touchscreen des Oszilloskops bzw. bricht den Schließvorgang ab.

Format: :MENU:LOCK <bool>

:MENU:LOCK?

Dabei ist „bool“ ein boolescher Typ mit den Werten {{0|OFF}}{{1|ON}}.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl deaktiviert den Touchscreen.

:MENU:LOCK ON oder MENU:LOCK 1

Die folgende Abfrage gibt „1“ zurück.

:MENU:LOCK?

MME

3.2.2.7.1 :MENU:HALF:CHANnel

Funktion: Stellt die Kanalposition auf die vertikale Nullposition ein (vertikale Mitte des Wellenform-Anzeigebereichs).

Format: :MENU:HALF:CHANnel <Kanal>

Dabei ist <Kanal> ein diskreter Typ: {CH1|CH2|CH3|CH4}.

3.2.2.7.2 :MENU:HALF:TRIGpos

Funktion: Stellt die Triggerposition auf die Mitte des Bildschirms ein.

Format: :MENU:HALF:TRIGpos <source>

Dabei ist <source> ein diskreter Wert, {CH1|CH2|CH3|CH4}.

3.2.2.7.3 :MENU:HALF:XCURsor

Funktion: Stellt den vertikalen Cursor des Kanals auf 50 % ein.

Format: :MENU:HALF:XCURsor

3.2.2.7.4 :MENU:HALF:YCURsor

Funktion: Stellt den horizontalen Cursor des Kanals auf 50 % ein.

Format: :MENU:HALF:YCURSor

3.2.2.7.5 :MENU:HALF:LEVel

Funktion: Stellt den Triggerpegel auf die mittlere Position der Trigger-Signalamplitude ein.

Format: :MENU:HALF:LEVel <Kanal>

Dabei ist <channel> ein diskreter Typ {CH1|CH2|CH3|CH4}, standardmäßig ist der aktuelle Kanal eingestellt.

~~3.2.2.7.6~~ :MENU:CHANnel

Funktion: Öffnet oder schließt das Kanalmenü

Format: :MENU:CHANnel <n>,<bool>

:MENU:CHANnel? <n>

Dabei ist <n> vom Typ „diskret“ und kann folgende Werte annehmen: {CH1|CH2|CH3|CH4 |MATH | REF|S1|S2};

<bool>, Typ „Boolean“, {{0|OFF}| {1|ON}}.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl schaltet die Anzeige von Kanal 1 ein.

:MENU:CHANnel CH1, ON oder:MENU:CHANnel CH1, 1

Die folgende Abfrage gibt „1“ zurück.

:MENU:CHANnel? CH1

~~MENU:QUICK~~

Funktion: Öffnet oder schließt das Kontextmenü (unteres Menü)

Format: :MENU:QUICK <bool>

:MENU:QUICK?

Dabei ist <bool> ein boolescher Wert, {{0|OFF}}{1|ON}}.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl schaltet die Anzeige von Kanal 1 ein.

:MENU:QUICK ON oder :MENU:QUICK 1

Die folgende Abfrage gibt „1“ zurück.

:MENU:QUICK?

1MENU

Funktion: Hauptmenü (oberes Menü) öffnen oder schließen

Format: :MENU:MAIN <bool>

:MENU:MAIN?

Dabei ist <bool> ein boolescher Wert, {{0|OFF}}{{1|ON}}.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl schaltet die Anzeige von Kanal 1 ein.

:MENU:MAIN ON oder :MENU:MAIN 1

Die folgende Abfrage gibt „1“ zurück.

:MENU:MAIN?

***1MENU*TRIGger**

3.2.2.11.1 :MENU:AUX:TRIGger <Typ>

Funktion: Legt den externen Trigger-Ein-/Ausgang fest.

Format: :MENU:AUX:TRIGger <Typ>

:MENU:AUX:TRIGger?

Dabei ist <Typ> ein diskreter Typ, {IN | OUT}.

Rückgabeformat: Die Abfrage gibt „IN“ oder „OUT“ zurück.

Beispiel:

Der folgende Befehl schaltet auf Trigger-Eingang um:

:MENU:AUX:TRIGger?

3.2.2.11.2 :MENU:AUX:INPutres

Funktion: Stellt die Eingangsimpedanz des Triggers ein.

Format: :MENU:AUX:INPutres <Typ>

:MENU:AUX:INPutres?

Dabei ist <Typ> ein diskreter Typ, {FIFTy | MEGA}.

Rückgabeformat: Die Abfrage gibt „FIFTy“ oder „MEGA“ zurück.

Beispiel:

Der folgende Befehl setzt die Trigger-Eingangsimpedanz auf 50 Ohm:

:MENU:AUX:INPutres FIFTy

~~2MENU:CLK~~

Funktion: Legt den Takt-Ein- und -Ausgang fest.

Format: :MENU:AUX:CLOCK <Typ>

:MENU:AUX:CLOCK?

Dabei ist <Typ> ein diskreter Typ, {IN | OUT}.

Rückgabeformat: Die Abfrage gibt „IN“ oder „OUT“ zurück.

Beispiel:

Der folgende Befehl schaltet auf Takt-Ausgabe um:

:MENU:AUX:CLOCK OUT

Die folgende Abfrage gibt „OUT“ zurück.

:MENU:AUX:CLOCK?

3.2.3 Befehlssystem für die Abtastung

ACQUIRE

Funktion: Legt die Abtastmethode fest.

Format: :ACQUIRE:TYPE <Typ>

:ACQUIRE:TYPE?

Wobei <Typ> diskret ist: { NORMAl | MEAN | ENVelop | PEAK }

Rückgabeformat: Die Abfrage gibt „NORMAl“, „MEAN“, „PEAK“ oder „ENVelop“ zurück.

Beispiel:

Der folgende Befehl wählt den Hüllkurven-Abtastmodus aus.

```
:ACQuire:TYPE ENVelop
```

Die folgende Abfrage gibt „ENVelop“ zurück.

```
:ACQuire:TYPE?
```

~~AC~~MEAN

Funktion: Legt die durchschnittliche Anzahl der Abtastwerte fest. Der festgelegte Wert muss ein ganzzahliges Vielfaches von 2 sein.

Format: :ACQuire:MEAN <Anzahl>

```
:ACQuire:MEAN?
```

Dabei ist <count> ein diskreter Typ, {2|4|8|16|32|64|128|256}

Rückgabeformat: Die Abfrage gibt eine ganze Zahl zurück.

Beispiel:

Der folgende Befehl legt die durchschnittliche Anzahl der Samples auf „32“ fest.

```
:ACQuire:MEAN 32
```

Die folgende Abfrage gibt „32“ zurück.

```
:ACQuire:MEAN?
```

~~AC~~ENVelop

Funktion: Legt die Abtastzeiten der Hüllkurve fest. Der einzustellende Wert ist ein ganzzahliges Vielfaches von 2 oder unendlich.

Format: :ACQuire:ENVelop <Anzahl>

:ACQuire:ENVelop?

Dabei ist <count> ein diskreter Typ, {2|4|8|16|32|64|128|256|inf}.

Rückgabeformat: Die Abfrage gibt eine ganze Zahl zurück.

Beispiel:

Der folgende Befehl setzt die Anzahl der Hüllkurven-Abtastungen auf „32“.

:ACQuire:ENVelop 32

Die folgende Abfrage gibt „32“ zurück.

:ACQuire:ENVelop?

~~ACQUIRE~~

Funktion: Segment-Speicher festlegen.

3.2.3.4.1 :ACQuire:SEGMENTed <bool>

:ACQuire:SEGMENTed?

Legt fest und fragt ab, ob die Segment-Speicherung aktiviert oder deaktiviert ist;

Wobei „bool“ der Typ „Boolean“ ist, {0|OFF}{1|ON};

Beispiel

Der folgende Befehl aktiviert die segmentierte Speicherung.

:ACQuire:SEGMented ON

Die folgende Abfrage gibt „1“ zurück.

:ACQuire:SEGMented?

3.2.3.4.2 :ACQuire:SEGMented:NO?

:ACQuire:SEGMented:NO?

Abfrage der Anzahl der derzeit ausgelösten Segmente;

Beispiel

Die folgende Abfrage gibt „1003“ zurück, was bedeutet, dass derzeit 1003 Segmente ausgelöst und im Speicher des FPGAs gespeichert wurden.

:ACQuire:SEGMented:NO?

3.2.3.4.3 :ACQuire:SEGMented:QTY <NO>

:ACQuire:SEGMented:QTY?

Festlegen und Abfragen der Anzahl der in segmentierter Form gespeicherten Segmente;

Wobei „no“ eine Ganzzahl ist; siehe Datenhandbuch;

Beispiel

Der folgende Befehl setzt die Anzahl der segmentierten Speichersegmente auf 4.

```
:ACQUIRE:SEGMENTED:QTY 4
```

Die folgende Abfrage gibt „4“ zurück.

```
:ACQUIRE:SEGMENTED:QTY?
```

3.2.3.4.4 :ACQUIRE:SEGMENTED:DISType < Typ >

```
:ACQUIRE:SEGMENTED:DISType?
```

Festlegen und Abfragen des Anzeigemodus des segmentierten Speichers;

wobei „type“ ein diskreter Typ ist: { SINGLE | FIT }; „SINGLE“ steht für die Einzelbildanzeige, „FIT“ für die Anpassungsanzeige

Beispiel

Der folgende Befehl richtet den segmentierten Speicher für die Einzelbildanzeige ein.

```
:ACQUIRE:SEGMENTED:DISType SINGLE
```

Die folgende Abfrage gibt „SINGLE“ zurück.

```
:ACQUIRE:SEGMENTED:DISType?
```

3.2.3.4.5 :ACQUIRE:SEGMENTED:ORDER <type>

```
:ACQUIRE:SEGMENTED:ORDER?
```

Festlegen und Abfragen der Wiedergabereihenfolge des Segment-Speichers;

wobei „type“ „discrete“ ist, { ORD er | REOR der }; „ORD er“ ist die Reihenfolge, „REOR der“ ist die umgekehrte Reihenfolge

Beispiel

Der folgende Befehl legt die sequenzielle Wiedergabe des Segment-Speichers fest.

:ACQuire:SEGMENTed:ORDER ORDER

Die folgende Abfrage gibt „ ORDER “ zurück.

:ACQuire:SEGMENTed:ORDER?

3.2.3.4.6 :ACQuire:SEGMENTed:PLAY

Automatische Wiedergabe starten

:ACQuire:SEGMENTed:STOP

Automatische Wiedergabe anhalten

3.2.3.4.7 :ACQuire:SEGMENTed:FRA1 <Wert>

:ACQuire:SEGMENTed:FRA1?

Festlegen und Abfragen des aktuellen Frames bei der Anzeige eines einzelnen Frames;

Dabei ist „value“ ein ganzzahliger Wert zwischen 1 und dem Maximalwert beim Anhalten

Beispiel

Der folgende Befehl setzt den aktuellen Frame auf 546.

:ACQUIRE:SEGMENTED:FRA1 546

Die folgende Abfrage gibt „546“ zurück.

:ACQUIRE:SEGMENTED:FRA1?

3.2.3.4.8 :ACQUIRE:SEGMENTED:FRA2 <Wert>

:ACQUIRE:SEGMENTED:FRA2<Wert>

:ACQUIRE:SEGMENTED:FRA2?

Legen Sie den Start-Frame fest, wenn die Abfrage-Anpassung angezeigt wird;

wobei „value“ eine ganze Zahl ist, die beim Anhalten den Wert 1 bis zum Maximalwert annimmt

Beispiel

Der folgende Befehl legt den Startframe der Anpassungsanzeige auf 100 fest.

:ACQUIRE:SEGMENTED:FRA2 100

Die folgende Abfrage gibt „100“ zurück.

:ACQUIRE:SEGMENTED:FRA2?

3.2.3.4.9 :ACQUIRE:SEGMENTED:FRA3 <Wert>

:ACQUIRE:SEGMENTED:FRA3?

Festlegen und Abfragen des Endframes der Fitting-Anzeige;

wobei „value“ eine ganze Zahl ist, FR2~der Maximalwert bei Stopp

Beispiel

Der folgende Befehl setzt den Endframe der Fitting-Anzeige auf 150.

:ACQuire:SEGMented:FRA3 150

Die folgende Abfrage gibt „150“ zurück.

:ACQuire:SEGMented:FRA3?

3.2.3.4.10 :ACQuire:SEGMented:PLAY:SPED <sped>

:ACQuire:SEGMented:SPED?

Geschwindigkeit der automatischen Wiedergabe festlegen und abfragen;

darunter „sped“, diskreter Typ, {1|2|4|8}

Beispiel

Der folgende Befehl stellt die Wiedergabegeschwindigkeit auf das 4-Fache ein.

:ACQuire:SEGMented:PLAY:SPED 4

Die folgende Abfrage gibt „4“ zurück.

:ACQuire:SEGMented:PLAY:SPED?

~~3.2.3.4.11~~

Funktion: Abfrage der Abtastrate des aktuellen analogen Kanals.

Format: :ACQuire:SRATe? -

~~ACQ~~DEPSelect

Funktion: Einstellen und Abfragen der aktuellen Speichertiefe des Oszilloskops.

Format: :ACQuire:DEPSelect <Typ>

:ACQuire:DEPSelect?

Dabei unterstützt <type> (diskreter Typ) je nach Modell unterschiedliche Werte und kann auf { AUTO|110000000|11000000|1100000|110000 | 11000} eingestellt werden

Beispiel

Der folgende Befehl stellt die Speichertiefe auf „AUTO“ ein.

:ACQuire:DEPSelect 22000000

Die folgende Abfrage gibt „22000000 “ zurück.

:ACQuire:DEPSelect?

~~ACQ~~DEPTH

Funktion: Abfrage des aktuellen Werts der aktuellen Speichertiefe des Oszilloskops.

Format: :ACQuire:DEPTh?

Davon gibt <type> (diskreter Typ) den tatsächlichen Wert der Speichertiefe je nach Modell zurück.

Beispiel

Die folgende Abfrage gibt „22000000 “ zurück.

:ACQuire:DEPTh?

3.2.4 Kanalbefehl-Subsystem

~~3.2.4 Kanalbefehl-Subsystem~~

Funktion: Kanal öffnen oder schließen

Format: :CHANnel <n>:DISPlay <bool>

:CHANnel <n>:DISPlay?

Dabei ist <n> vom Typ „diskret“ ({ 1|2|3|4}); <bool> vom Typ „Boolescher Wert“ ({0| OFF| 1|ON}).

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl schaltet die Anzeige von Kanal 1 ein.

:CHANnel1:DISPlay ON oder :CHANnel1:DISPlay 1

Die folgende Abfrage gibt „1“ zurück.

:CHANnel1:DISPlay?

~~:CHANnel <n>:INVerse ~~

Funktion: Aktiviert oder deaktiviert die invertierte Anzeige des analogen Kanals.

Format: :CHANnel <n>:INVerse <bool>

:CHANnel <n>:INVerse?

Dabei ist <n> vom Typ „diskret“ { 1|2|3|4} ; <bool> vom Typ „bool“ {{0| OFF}|
{1|ON}}.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl aktiviert die invertierte Anzeige von Kanal 1.

:CHANnel1:INVerse ON oder :CHANnel1:INVerse 1

Die folgende Abfrage gibt „1“ zurück.

:CHANnel1:INVerse?

~~:CHANnel <n>:BAND <Typ>, <Frequenz>~~

Funktion: Stellt die Bandbreitenbegrenzung des analogen Kanals auf „20M“
oder „Volle Bandbreite“ ein.

Format: :CHANnel <n>:BAND <Typ>, <Frequenz>

:CHANnel <n>:BAND?

Parameter: <n>, diskreter Typ, { 1|2|3| 4}; <type>, diskreter Typ, {20M|FULL|HIGH|LOW}; <freq>, reeller Typ, {siehe Datenblatt}, nur gültig bei „HIGH“ und „LOW“.

Rückgabeformat: Die Abfrage gibt „20M“, „FULL“, „HIGH“ und „LOW“ zurück.

Beispiel:

Der folgende Befehl setzt die Bandbreitenbegrenzung von Kanal 1 auf „High“, 10000000.

:CHANnel1:BAND HIGH, 10000000

[Hinweis] „10000000“ kann ein beliebiger Wert sein. Dieser Wert ist unter den Einstellungen „20M“ und „FULL“ ungültig.

:CHANnel1:BAND?

~~CHANnelPRty~~

Funktion: Legt den Messtyp des analogen Kanals auf „Spannung“ oder „Strom“ fest.

Format: :CHANnel <n>:PRTY <Typ>

:CHANnel <n>:PRTY?

Dabei ist <n> ein diskreter Wert aus { 1|2|3|4 }; <type> ein diskreter Wert aus {VOL|CUR | BAR|MPA| PSI }. (Die drei markierten Parameter sind für Kfz-Oszilloskope geeignet)

Rückgabeformat: Die Abfrage gibt „VOL“, „CUR“, „BAR“, „MPA“ oder „PSI“ zurück.

Beispiel:

Der folgende Befehl zeichnet Kanal 1 mit dem Messspannungsmodus auf.

```
:CHANnel1:PRTY VOL
```

Die folgende Abfrage gibt „VOL“ zurück.

```
:CHANnel1:PRTY?
```

~~CHANnel1:PROBe~~

Funktion: Stellt das Dämpfungsverhältnis der analogen Kanal-Messspitze ein.

Format: :CHANnel <n>:PROBe <atten>

```
:CHANnel <n>:PROBe?
```

Dabei ist <n> diskret, { 1|2|3|4 }; <atten> ist diskret,

{ 0,001|0,002|0,005|0,01|0,02|0,05|0,1|0,2|0,5|1|2|5|10|20|50|100|200|500|1000 }.

Ausgabeformat: Die Abfrage gibt „0,001“, „0,002“, „0,005“, „0,01“, „0,02“, „0,05“, „0,1“, „0,2“, „0,2“, „1“, „2“, „5“, „10“, „20“, „50“, „100“, „200“, „500“, „1000“.

Beispiel:

Der folgende Befehl setzt das Dämpfungsverhältnis der an Kanal 1 angeschlossenen Sonde auf 10.

```
:CHANnel1:PROBe 10
```

Die folgende Abfrage gibt „10“ zurück.

:CHANnel1:PROBe?

~~:CHANnel<n>COUPle~~

Funktion: Stellt den Kopplungsmodus des analogen Kanaleingangs auf „AC“, „DC“ oder „GND“ ein.

Format: :CHANnel <n>:COUPle <couple>

:CHANnel <n>:COUPle?

Dabei ist <n> ein diskreter Typ, { 1|2|3|4} ; <couple> ein diskreter Typ, {AC|DC| GND}

Rückgabeformat: Die Abfrage gibt „AC“, „DC“ oder „GND“ zurück.

Beispiel:

Der folgende Befehl setzt den Eingangskopplungsmodus von Kanal 1 auf „AC“.

:CHANnel1:COUPle AC

Die folgende Abfrage gibt „AC“ zurück.

:CHANnel1:COUPle?

~~:CHANnel<n>INPutres~~

Funktion: Stellt die Eingangsimpedanz des analogen Kanals auf „MEGA (1 M Ω)“ oder „FIFTy (50 Ω)“ ein.

Format: :CHANnel <n>:INPutres <input>

:CHANnel <n>:INPutres?

Dabei ist <n> diskret, { 1|2|3|4 }; <input> ist diskret, { MEGA| FIFTy }.

Rückgabeformat: Die Abfrage gibt „MEGA“ oder „FIFTy“ zurück.

Beispiel:

Der folgende Befehl setzt die Eingangsimpedanz von Kanal 1 auf 1 MΩ.

```
:CHANnel1:INPutres MEGA
```

Die folgende Abfrage gibt „MEGA“ zurück.

```
:CHANnel1:INPutres?
```

~~CHAN<n>SCALe~~ **CHAN<n>EXETent**



Funktion: Stellt den vertikalen Maßstab der Wellenformanzeige des angegebenen analogen Kanals ein.

Format: :CHANnel <n>:SCALe <extent>

```
:CHANnel <n>:SCALe?
```

Dabei ist <n> ein diskreter Typ, {1|2|3|4}; <extent> ist ein reeller Typ, der die Maximal- und Minimalwerte nicht überschreitet

Maximalwert: maximale Verstärkung des Oszilloskops * aktuell eingestellter Sondeneinstellfaktor

Mindestwert: Minimaler Messbereich des Oszilloskops * aktuell eingestellter Messbereichsfaktor der Sonde.

Rückgabeformat: Die Abfrage gibt den Wert der vertikalen Skalierung in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl stellt die vertikale Skalierung von Kanal 1 auf 1 V/div ein.

```
:CHANnel1:SCALe 1
```

Die folgende Abfrage gibt „1,000000e+00“ zurück.

```
:CHANnel1:SCALe?
```

~~CHANnel~~ POSition

Funktion: Legt die vertikale Position der angezeigten Wellenform des angegebenen Kanals fest.

Format: :CHANnel <n>:POSition <pos>

```
:CHANnel <n>:POSition?
```

Dabei ist <n> vom Typ „diskret“ {1|2|3|4} und <pos> vom Typ „reell“.

Ausgabeformat: Die Abfrage gibt den Offset-Wert in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt den vertikalen Offset von Kanal 1 auf 0,01 V.

```
:CHANnel1:POSition 0,01
```

Die folgende Abfrage gibt „1,000000e-02“ zurück

```
:CHANnel1:POSition?
```

~~CHANnel~~ VB

Funktion: Legt die vertikale Ausdehnungsbasis des analogen Kanals fest.

Format: :CHANnel <n>:VREF <bool>

:CHANnel <n>:VREF?

Dabei gilt: <n>, diskreter Typ, { 1|2|3| 4}; <bool>, diskreter Typ, { CENTER|ZERO }.

Rückgabeformat: Die Abfrage gibt „CENT“ oder „ZERO“ zurück.

Beispiel:

Der folgende Befehl legt den vertikalen Referenzpunkt von Kanal 1 auf die Mitte fest.

:CHANnel1:VREF CENTER

Die folgende Abfrage gibt „CENT“ zurück.

:CHANnel1:VREF?

~~**Funktion:** Legt die Kanalbezeichnung des analogen Kanals fest.~~

Funktion: Legt die Kanalbezeichnung des analogen Kanals fest.

Format: :CHANnel <n>:LABel < string >

:CHANnel <n>:LABel?

Wobei: <n>, diskreter Typ, { 1|2|3| 4}; <string>, Zeichenkette.

Rückgabeformat: Abfrage-Rückgabebeihenfolge.

Beispiel:

Der folgende Befehl setzt die Bezeichnung von Kanal 1 auf DDR.

:CHANnel1:LABel DDR

Die folgende Abfrage gibt „DDR“ zurück.

:CHANnel1:LABel?

~~CHANnel~~

Funktion: Kanalbezeichnung löschen.

Format: :CHANnel <n>:LABel:CLEar

Dabei ist <n> ein diskreter Typ, { 1|2|3|4 }.

Beispiel:

Der folgende Befehl löscht die Bezeichnung von Kanal 1.



~~CHANnel~~<n>

Funktion: Den aktuellen Kanal festlegen.

Format: :CURrent:CHANnel <n>

:CURrent:CHANnel?

Darunter: <n>, diskreter Typ, {CH1|CH2|CH3|CH4|MATH|R1|R2|R3|R4|S1|S2}.

Rückgabeformat: Die Abfrage gibt

{CH1|CH2|CH3|CH4|MATH|R1|R2|R3|R4|S1|S2} zurück.

Beispiel:

Der folgende Befehl legt den vertikalen Ausdehnungsbezugspunkt von Kanal 1 auf die Mitte fest.

:CURRent:CHANnel CH1

Die folgende Abfrage gibt „CH1“ zurück.

:CURRent:CHANnel?

3.2.5 Mathematik-Befehlssystem

MATH

Funktion: Schaltet den mathematischen Operationstyp ein oder aus.

Format: :MATH:DISPlay <bool>

:MATH:DISPlay?

Darunter <bool>, Typ „Boolean“, {{0|OFF}}{1|ON}}.

MATHMODE

Funktion: Wählt die Art der mathematischen Operation aus.

Format: :MATH:MODE <Modus>

:MATH:MODE?

Dazu gehört <mode>, diskreter Typ, {BASE | FFT| AX+B|ADVAnced}.

Rückgabeformat: Die Abfrage gibt „BASE“, „FFT“, „AX+B“ und „ADVAnced“ zurück.

Beispiel:

Der folgende Befehl wählt die FFT-Operation aus.

:MATH:MODE FFT

Die folgende Abfrage gibt „FFT“ zurück.

:MATH:MODE?

~~3.4.7.3~~

Funktion: Legt die vertikale Erweiterungsbasis der mathematischen Wellenform fest.

Format: :MATH:VREF <bool>

:MATH:VREF?

Wobei: <bool>, diskreter Typ, { CENTer | ZERO }.

Rückgabeformat: Die Abfrage gibt „CENT“ oder „ZERO“ zurück.

Beispiel:

Der folgende Befehl setzt die Basis der vertikalen Erweiterung auf die Mitte.

:MATH:VREF CENTER

Die folgende Abfrage gibt „CENT“ zurück.

:MATH:VREF?

MATH

3.2.5. 4.1 :MATH:BASE:SOU1

Funktion: Auswahl von Quelle 1 für den Dual-Wellenform-Betrieb

Format: :MATH:BASE:SOU1 <source>

:MATH:BASE:SOU1?

Dabei ist <source> ein diskreter Typ, {CH1|CH2|CH3|CH4}.

Rückgabeformat: Die Abfrage gibt „CH1“, „CH2“, „CH3“ oder „CH4“ zurück.

Beispiel:

Der folgende Befehl wählt Kanal 1 als Quelle 1 aus.

:MATH:BASE:SOU1 CH1

Die folgende Abfrage gibt „CH1“ zurück.

:MATH:BASE:SOU1?

3.2.5. 4.2 :MATH:BASE:SOU2

Funktion: Wählt Quelle 2 für den Dual-Wellenform-Betrieb aus.

Format: :MATH:BASE:SOU2 <source>

:MATH:BASE:SOU2?

Dabei ist <source> ein diskreter Typ, {CH1|CH2|CH3|CH4}.

Rückgabeformat: Die Abfrage gibt „CH1“, „CH2“, „CH3“ oder „CH4“ zurück.

Beispiel:

Der folgende Befehl wählt Kanal 1 als Quelle 2 aus.

:MATH:BASE:SOU2 CH1

Die folgende Abfrage gibt „CH1“ zurück.

:MATH:BASE:SOU2?

3.2.5. 4.3 :MATH:BASE:VSCale

Funktion: Legt den vertikalen Maßstab des Ergebnisses der Doppelwellenform-Operation fest.

Format: :MATH:BASE:VSCale < Bereich >

:MATH:BASE:VSCale?

Dabei ist <extent> vom Typ „real“ und muss im Bereich {1e-15 bis 5e14} liegen, wobei nur die Schritte 1, 2 und 5 zulässig sind.

Rückgabeformat: Die Abfrage gibt den Wert in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt die vertikale Skalierung des Additionsergebnisses auf 1.

:MATH:BASE:VSCale 1

Die folgende Abfrage gibt „1,000000e+00“ zurück.

:MATH:BASE:VSCale?

3.2.5.4.4 :MATH:BASE:VPOStion

Funktion: Legt den vertikalen Versatz des Ergebnisses der Dual-Wellenform-Operation fest.

Format: :MATH:BASE:VPOStion <position>

:MATH:BASE:VPOStion?

Dabei ist <position> vom Typ „real“ und wird in wissenschaftlicher Notation angegeben.

:MATH:BASE:VPOStion 8 /* Vertikalen Versatz auf 8 V festlegen*/

:MATH:BASE:VPOStion? Die Abfrage gibt 8,000000E0* zurück

3.2.5.4.5 :MATH:BASE:OPERator

Funktion: Wählt den Operator für die Additionsoperation aus

Format: :MATH:BASE:OPERator <operator>

:MATH:BASE:OPERator?

Dabei ist <operator> ein diskreter Typ: {ADD|SUB|MUL|DIV}.

Rückgabeformat: Die Abfrage gibt „ADD“, „SUB“, „MUL“ oder „DIV“ zurück.

Beispiel:

Der folgende Befehl setzt den Operator auf „Plus“.

:MATH:BASE:OPERator ADD

Die folgende Abfrage gibt „ADD“ zurück.

:MATH:BASE:OPERator?

**3.2.5.5.1 :MATH:FFT:SOURce**

Funktion: Wählen Sie die Quelle für die FFT-Operation aus.

Format: :MATH:FFT:SOURce <source>

:MATH:FFT:SOURce?

Dabei gilt: <source>, diskreter Typ, {CH1|CH2|CH3|CH4}.

Rückgabeformat: Die Abfrage gibt „CH1“, „CH2“, „CH3“ oder „CH4“ zurück.

Beispiel:

Der folgende Befehl wählt Kanal 1 als Quelle aus.

:MATH:FFT:SOURce CH1

Die folgende Abfrage gibt „CH1“ zurück.

:MATH:FFT:SOURce?

3.2.5.5.2 :MATH:FFT:WINDow

Funktion: Wählt die Fensterfunktion für die FFT-Operation aus.

Format: :MATH:FFT:WINDow <source>

:MATH:FFT:WINDow?

Darin sind <source>, diskrete Typen,
{ RECTangle|HAMMing|BLACKman|HANNing }.

Rückgabeformat: Die Abfrage gibt „RECTangle“, „HAMMing“, „BLACKman“ oder „HANNing“ zurück.

Beispiel:

Der folgende Befehl wählt die HANNing-Fensterfunktion aus.

:MATH:FFT:WINDow HANNing

Die folgende Abfrage gibt „HANNing“ zurück.

:MATH:FFT:WINDow?

3.2.5.5.3 :MATH:FFT:TYPE

Funktion: Wählen Sie den Anzeigemodus der FFT-Wellenform als „Linear“ oder „Logarithmisch“ .

Format: :MATH:FFT:TYPE <Typ>

:MATH:FFT:TYPE?

Dabei ist <type> ein diskreter Typ, {LINE|DB}.

Rückgabeformat: Die Abfrage gibt „LINE“ oder „DB“ zurück.

Beispiel:

Der folgende Befehl wählt den logarithmischen Anzeigemodus aus.

```
:MATH:FFT:TYPE DB
```

Die folgende Abfrage gibt „DB“ zurück.

```
:MATH:FFT:TYPE?
```

3.2.5.5.4 :MATH:FFT:VSCale

Funktion: Legt den vertikalen Maßstab des FFT-Ergebnisses fest.

Format: :MATH:FFT:VSCale <Ausmaß>

```
:MATH:FFT:VSCale?
```

Dabei ist <extent> vom Typ „real“, <extent> vom Typ „real“, in Zeilenangabe, {1e-15 bis 5e14, nur in Schritten von 1, 2, 5} oder in dB {1 bis 500, Schritte von 1, 2, 5}.

Rückgabeformat: Die Abfrage gibt den Skalierungsfaktor in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt den vertikalen Maßstab des FFT-Ergebnisses auf 1.

```
:MATH:FFT:VSCale 1
```

Die folgende Abfrage gibt „1.000000e+00“ zurück.

:MATH:FFT:VScale?

3.2.5.5.5 :MATH:FFT:VPOStion

Funktion: Legt den vertikalen Versatz des FFT-Ergebnisses fest.

Format: :MATH:FFT:VPOStion <Position>

:MATH:FFT:VPOStion?

Dabei ist <positionoffset> vom Typ „real“ und wird in wissenschaftlicher Notation angegeben.

3.2.5.5.6 :MATH:FFT:HSCale

Funktion: Legt die horizontale Skalierung der FFT-Ergebnisse fest.

Format: :MATH:FFT:HSCale <hscale>

:MATH:FFT:HSCale?

Dabei ist <hscale> vom Typ „real“ und kann einen Wert zwischen 1 Hz und 100 MHz in Schritten von 1, 2 oder 5 annehmen.

Rückgabeformat: Die Abfrage gibt den Skalierungswert in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl legt den horizontalen Maßstab des FFT-Ergebnisses auf

1.

:MATH:FFT:HSCale 1

Die folgende Abfrage gibt „1.000000e+00“ zurück.

:MATH:FFT:HSCale?

3.2.5.5.7 :MATH:FFT:HPOSITION

Funktion: Legt den horizontalen Versatz des FFT-Ergebnisses fest.

Format: :MATH:FFT:HPOSition <position >

:MATH:FFT:HPOSition?

Dabei ist <position> vom Typ „real“,

Rückgabeformat: Die Abfrage gibt den Versatzwert in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt den horizontalen Offset auf 2 Hz.

:MATH:FFT:HPOSition 2

Die folgende Abfrage gibt „2,000000e0“ zurück

:MATH:FFT:HPOSition?

MATH

3.2.5. 6.1 :MATH:AX+B:SOURce

Funktion: Wählen Sie die Quelle für die AX+B-Operation aus.

Format: :MATH:AX+B:SOURce <source>

:MATH:AX+B:SOURce?

Dabei ist <source> ein diskreter Typ: {CH1|CH2|CH3|CH4}.

Rückgabeformat: Die Abfrage gibt „CH1“, „CH2“, „CH3“ oder „CH4“ zurück.

Beispiel:

Der folgende Befehl wählt Kanal 1 als Quelle aus.

:MATH:AX+B:SOURce CH1

Die folgende Abfrage gibt „CH1 “ zurück.

:MATH:AX+B:SOURce?

3.2.5. 6.2 :MATH:AX+B:A

Funktion: Wählt den Wert von A in AX+B aus.

Format: :MATH:AX+B:A <a>

:MATH:AX+B:A?

Dabei ist <a> vom Typ „real“; den Wertebereich entnehmen Sie bitte dem Handbuch.

Rückgabeformat: Die Abfrage gibt einen reellen Wert zurück.

Beispiel:

Der folgende Befehl legt den numerischen Wert von A fest.

:MATH:AX+B:A 2

Die folgende Abfrage gibt „2“ zurück.

:MATH:AX+B:A?

3.2.5.6.3 :MATH:AX+B:B

Funktion: Wählt den Wert von B in AX+B aus.

Format: :MATH:AX+B:B

:MATH:AX+B:B?

Dabei ist vom Typ „real“; den Wertebereich entnehmen Sie bitte dem Handbuch.

Rückgabeformat: Die Abfrage gibt einen reellen Wert zurück.

Beispiel:

Der folgende Befehl legt den Wert von B fest.

:MATH:AX+B:B 100

Die folgende Abfrage gibt „100“ zurück.

:MATH:AX+B:B?

3.2.5.6.4 :MATH:AX+B:UNIT <Einheit>

Funktion: Wählt die Einheit in AX+B aus.

Format: :MATH:AX+B:UNIT <Einheit>

:MATH:AX+B:UNIT?

Dabei ist <unit> eine Zeichenfolge, deren Wertebereich im Datenblatt angegeben ist.

Rückgabeformat: Die Abfrage gibt einen reellen Wert zurück.

Beispiel:

Der folgende Befehl berechnet Einheiten.

:MATH:AX+B:UNIT W

Die folgende Abfrage gibt „W“ zurück.

:MATH:AX+B:UNIT?

3.2.5.6.5 :MATH:AX+B:VSCale

Funktion: Legt den vertikalen Maßstab des Rechenergebnisses fest.

Format: :MATH:AX+B:VSCale <Ausmaß>

:MATH:AX+B:EXTent?

Dabei ist <extent> vom Typ „real“ ($\{1e-15 \sim 5e14\}$, kann nur in Schritten von 1, 2, 5 } erhöht werden.

Rückgabeformat: Die Abfrage gibt den Wert in wissenschaftlicher Notation zurück.

Beispiel: Der folgende Befehl setzt den vertikalen Maßstab des Ergebnisses der logischen Operation auf 1.

:MATH:AX+B:VScale 1

Die folgende Abfrage gibt „1,000000e+00“ zurück.

:MATH:AX+B:VScale?

3.2.5.6.6 :MATH:AX+B:VPOSiton

Funktion: Legt den vertikalen Versatz des Berechnungsergebnisses fest.

Format: :MATH:AX+B:VPOSiton < Position >

:MATH:AX+B:VPOSiton?

Dabei ist < position > vom Typ „real“ und wird in wissenschaftlicher Notation ausgedrückt.

3.2.5.7

3.2.5.7.1 :MATH:Erweitert:Ausdruck

Funktion: Legen Sie den Ausdruck für erweiterte Operationen fest.

Format: :MATH:ADVanced:EXPRession <Zeichenkette>

:MATH:ADVanced:EXPRession?

Dabei ist <Zeichenkette> eine ASCII-Zeichenkette.

Rückgabeformat: Die Abfrage gibt den aktuellen Ausdruck in Zeichenfolgenform zurück.

Beispiel:

Der folgende Befehl setzt den Ausdruck auf „CH1+CH2“.

:MATH:ADVanced:EXPRession CH1+CH2

Die folgende Abfrage gibt „CH1+CH2“ zurück.

:MATH:Erweitert:Ausdruck?

3.2.5.7.2 :MATH:Erweitert:VAR1

Funktion: Legt die Variable im Ausdruck der erweiterten Operation 1 fest.

Format: :MATH:ADVanced:VAR1 <Wert>

Dabei ist <Wert> ein reeller Wert im Bereich von -9,9999E+9 bis 9,9999E+9. Den genauen Bereich entnehmen Sie bitte dem Datenblatt.

Rückgabeformat: Die Abfrage gibt den Wert der aktuellen Variablen 1 in wissenschaftlicher Notation zurück.

Beispiel: Der folgende Befehl setzt den Wert der Variablen 1 auf 100.

:MATH:ADVanced:VAR1 100

Die folgende Abfrage gibt „1,000000e+02“ zurück.

:MATH:ADVanced:VARiable1?

3.2.5. 7.3 :MATH:ADVanced:VAR2

Funktion: Variablen 2 in einem Ausdruck für erweiterte Operationen **setzen**

Format: :MATH:ADVanced:VAR2 <Wert>

Dabei ist <Wert> vom Typ „real“ und liegt im Bereich von -9,9999E+9 bis 9,9999E+9;
den genauen Bereich entnehmen Sie bitte dem Datenblatt.

Rückgabeformat: Die Abfrage gibt den Wert der aktuellen Variablen 2 in
wissenschaftlicher Notation zurück.

Beispiel: Der folgende Befehl setzt den Wert der Variablen 2 auf 100.

:MATH:ADVanced:VAR2 100

Die folgende Abfrage gibt „1,000000e+02“ zurück.

:MATH:ADVanced:VAR2?

3.2.5.7.4 :MATH:ADVanced:VSCale

Funktion: Legt die vertikale Skalierung der Ergebnisse erweiterter Berechnungen fest.

Format: :MATH:ADVanced:VSCale <Ausdehnung>

:MATH:ADVanced:VSCale?

Dabei ist <extent> vom Typ „real“ ({1e-15~5e14, kann nur um 1, 2 oder 5 erhöht
werden}).

Rückgabeformat: Die Abfrage gibt den Wert in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt die vertikale Skalierung des Ergebnisses der erweiterten Operation auf 1.

:MATH:ADVanced:VScale 1

Die folgende Abfrage gibt „1,000000e+00“ zurück.

:MATH:ADVanced:VScale?

3.2.5.7.5 :MATH:ADVanced:VPOSiton

Funktion: Legt den vertikalen Versatz der Ergebnisse der erweiterten Berechnung fest.

Format: :MATH:ADVanced:VPOSiton <Position>

:MATH:ADVanced:VPOSiton?

Dabei ist <position> ein reeller Wert, der in wissenschaftlicher Notation angegeben wird.

3.2.5.7.6 :MATH:ADVanced:UNIT <Einheit>

Funktion: Wählt Einheiten in „ADVanced“ aus.

Format: :MATH:ADVanced:UNIT <unit>

:MATH:ADVanced:UNIT?

Darunter <unit>, Zeichenfolge.

Rückgabeformat: Die Abfrage gibt einen reellen Wert zurück.

Beispiel:

Der folgende Befehl berechnet Einheiten.

:MATH:ADVanced:UNIT W

Die folgende Abfrage gibt „W“ zurück.

:MATH:ADVanced:UNIT?

3.2.5.2

Fragt die Abtastrate mathematischer Wellenformen ab; der Rückgabewert wird in wissenschaftlicher Notation ausgedrückt.

Beispiel:

Die folgende Abfrage gibt „2,500000e8“ zurück.

:MATH:SRATe?

3.2.5.3

Fragt die Anzahl der Punkte der mathematischen Wellenform ab. Der zurückgegebene Wert wird in wissenschaftlicher Notation ausgedrückt.

Beispiel:

Die folgende Abfrage gibt „7,000000e2“ zurück.

:SAMPleACQuire:MATH:DEPth?

3.2.6 Cursor-Befehlssystem

SET

Funktion: Schaltet die horizontale Cursor-Funktion ein oder aus.

Format: :CURSor:HORizontal <bool>

:CURSor:HORizontal?

Dabei ist <bool> vom Typ „Boolean“ und kann die Werte {{0|OFF}}{{1|ON}} annehmen.

SET

Funktion: Die vertikale Cursor-Funktion ein- oder ausschalten.

Format: :CURSor:VERTical <bool>

:CURSor:VERTical?

Dazu gehören <bool>, ein boolescher Typ, {{0|OFF}}{{1|ON}}.

SET

Funktion: Legt die Position des vertikalen Cursors X1 fest.

Format: :CURSor:CX1 <px>

:CURSor:CX1?

Dabei ist <px> eine ganze Zahl und wird in Pixeln angegeben.

Rückgabeformat: Die Abfrage gibt eine ganze Zahl zurück.

Beispiel:

Der folgende Befehl setzt die horizontale Position des vertikalen Cursors X1 auf „100“.

:CURSor:CX1 100

Die folgende Abfrage übersetzt „100“.

:CURSor:CX1?

52

Funktion: Position des vertikalen Cursors X2 festlegen.

Format: :CURSor:CX2<px>

:CURSor:CX2?

Dabei ist <px> eine ganze Zahl in Pixeln.

Rückgabeformat: Die Abfrage gibt eine ganze Zahl zurück.

Beispiel:

Der folgende Befehl setzt die horizontale Position des vertikalen Cursors X2 auf „100“.

:CURSor:CX2 100

Die folgende Abfrage übersetzt „100“.

:CURSor:CX2?

51

Funktion: Position des horizontalen Cursors 1 festlegen.

Format: :CURSor:CY1<px>

:CURSor:CY1?

Dabei ist <px> eine ganze Zahl in Pixeln.

Rückgabeformat: Die Abfrage gibt eine ganze Zahl zurück.

Beispiel:

Der folgende Befehl setzt die vertikale Position des horizontalen Cursors Y1 auf „100“.

:CURSor:CY1100

Die folgende Abfrage übersetzt „100“.

:CURSor:CY1?

52

Funktion: Position des horizontalen Cursors 2 festlegen.

Format: :CURSor:CY2<px>

:CURSor:CY2?

Darin steht <px>, eine ganze Zahl in Pixeln.

Rückgabeformat: Die Abfrage gibt eine ganze Zahl zurück.

Beispiel:

Der folgende Befehl setzt die vertikale Position des horizontalen Cursors Y2 auf „100“.

:CURSor:CY2 100

Die folgende Abfrage übersetzt „100“.

:CURSor:CY2?

1Value

Funktion: Abfrage des x-Werts des vertikalen Cursors X1.

Format: :CURSor:X1Value?

Die Einheiten des Abfragewerts richten sich nach den aktuellen horizontalen Einheiten.

Rückgabeformat: Die Abfrage gibt den X-Wert am Cursor X1 in wissenschaftlicher Notation zurück.

Beispiel:

Die folgende Abfrage gibt „-0,000000e-02“ zurück

:CURSor:X1Value?

2Value

Funktion: Abfrage des X-Werts des vertikalen Cursors X2.

Format: :CURSor:X2Value?

Die Einheit des Abfragewerts richtet sich nach der aktuellen horizontalen Einheit.

Rückgabeformat: Die Abfrage gibt den X-Wert am Cursor X2 in wissenschaftlicher Notation zurück.

Beispiel:

Die folgende Abfrage gibt „-0,000000e-02“ zurück

:CURSor:X2Value?



Funktion: Abfrage des y-Werts des horizontalen Cursors Y1.

Format: :CURSor:Y1Value?

Die Einheit des Abfragewerts richtet sich nach der aktuellen vertikalen Einheit.

Rückgabeformat: Die Abfrage gibt den Y-Wert am Cursor A in wissenschaftlicher Notation zurück.

Beispiel:

Die folgende Abfrage gibt „-0,000000e-02“ zurück

:CURSor:YAValue?



Funktion: Abfrage des Y-Werts des horizontalen Cursors Y2.

Format: :CURSor:Y2Value?

Die Einheit des Abfragewerts wird durch die vertikale Einheit bestimmt.

Rückgabeformat: Die Abfrage gibt den Y-Wert am Cursor B in wissenschaftlicher Notation zurück.

Beispiel:

Die folgende Abfrage gibt „-0,000000e-02“ zurück

:CURSor:Y2Value?

DELta

Funktion: Ermittelt die Differenz Δ zwischen den vertikalen Cursors X1 und X2. Die Einheit entspricht der horizontalen Einheit.

Format: :CURSor:XDELta?

Rückgabeformat: Die Abfrage gibt den aktuellen Differenzwert X in wissenschaftlicher Notation **zurück**. Δ

Beispiel:

Die folgende Abfrage gibt „1,000000e-03“ zurück.

:CURSor:XDELta?

DELta

Funktion: Ermittelt die Differenz zwischen den horizontalen Cursorpositionen Y1 und Y2 (Δ); die Einheit entspricht der vertikalen Einheit.

Format: :CURSor:YDELta?

Ausgabeformat: Die Abfrage gibt den aktuellen Differenzwert X in wissenschaftlicher Notation zurück: Δ

Beispiel:

Die folgende Abfrage gibt „1,000000e-03“ zurück.

:CURSor:YDELta?



Funktion: Abfrage des Verhältnisses 1/ zwischen den vertikalen Cursors X1 und X2 Δ in Hz.

Format: :CURSor:FREQ?

Rückgabeformat: Die Abfrage gibt den aktuellen Wert in wissenschaftlicher Notation zurück.

Beispiel:

Die folgende Abfrage gibt „1,000000e 03“ zurück.

:CURSor:FREQ?



Funktion: Ermittelt das Verhältnis zwischen der Differenz der horizontalen Cursor A und B und der Differenz der vertikalen Cursor A und B.

Format: :CURSor:RATIo?

Ausgabeformat: Die Abfrage gibt den Wert in wissenschaftlicher Notation zurück.

Beispiel:

Die folgende Abfrage gibt „3,200000e-02“ zurück.

:CURSor:RATIo?



Funktion: Legt die Kanalquelle für die Cursor-Messung fest.

Format: :CURSor:SOURce <source>

:CURSor:SOURce?

Dabei ist <source> ein diskreter Typ: {CH1|CH2|CH3|CH4| R1| R2| R3| R4|MATH
|AUTO}.

Rückgabeformat: Die Abfrage gibt „CH1“, „CH2“, „CH3“, „CH4“, „R1“, „R2“, „R3“, „R4“
oder „MATH“ zurück.

Beispiel:

Der folgende Befehl legt Kanal 1 als Messquelle fest.

:CURSor:SOURce CH1

Die folgende Abfrage gibt „CH1“ zurück.

:CURSor:SOURce?



3.2.6.16.1 :PHCUrsor [< bool >, < src1> , < src2>]

Funktion: Aktivieren/Deaktivieren/Abfragen des Phasencursors und Festlegen der zugehörigen Parameter.

Format: :PHCUrsor [< bool >, < src1> , < src2>]

:PHCUrsor?

Dabei sind <bool>, <bool> vom Typ Boolesche Werte, {{0|OFF}}{1|ON}}; <src1>, Ganzzahl, steht für die Anzahl der einzustellenden Zylinder; <src2>, Ganzzahl, steht für den Einstellwinkel, der in der Regel ein Vielfaches von 180 ist;

Rückgabeformat: Die Abfrage gibt die entsprechenden Parameter zurück.

Beispiel:

Der folgende Befehl schaltet den Phasen-Cursor ein und stellt den Cursor auf 4 Zylinder und 720 Grad ein.

```
:PHCUrsor 1, 4, 720
```

Die folgende Abfrage gibt „1, 4,720“ zurück.

```
:PHCUrsor?
```

3.2.6.16.2 :PHCUrsor:X0 <px> ,

Funktion: Festlegen oder Abfragen der Pixelposition der 0-Grad-Cursorlinie.

Format: :PHCUrsor:X0 <px>

:PHCUrsor:X0?

Dabei ist <px> eine ganze Zahl, die die Pixelposition der 0-Grad-Cursorlinie auf dem Bildschirm angibt, wobei die linke Bildschirmkante als Bezugspunkt dient;

Rückgabeformat: Die Abfrage gibt die Pixelposition der 0-Grad-Cursorlinie auf dem Bildschirm zurück.

Beispiel:

Der folgende Befehl setzt die Position der 0-Grad-Cursorlinie auf 100 Pixel.

:PHCUrsor:X0 100

Die folgende Abfrage gibt „100“ zurück.

:PHCUrsor:X0?

3.2.6.16.3 :PHCUrsor:YN <px>,

Funktion: Festlegen oder Abfragen der Pixelposition der letzten Cursorzeile.

Format: :PHCUrsor:YN <px>

:PHCUrsor:YN?

Dabei ist <px> eine ganze Zahl, die die Pixelposition der letzten Cursorzeile, gemessen vom linken Bildschirmrand, angibt;

Rückgabeformat: Die Abfrage gibt die Pixelposition der letzten Cursorzeile auf dem Bildschirm zurück.

Beispiel:

Der folgende Befehl setzt die Position der letzten Cursorzeile auf 200 Pixel.

:PHCursor:ZN 200

Die folgende Abfrage gibt „200“ zurück.

3.2.7 Anzeigebefehl-Subsystem

~~DISPlay~~

Funktion: Legt den Anzeigemodus der Wellenform auf dem Bildschirm fest: „Punktanzeige“ oder „Linienanzeige“.

Format: :DISPlay:WAVEform <Typ>

:DISPlay:WAVEform?

Dabei ist <Typ> ein diskreter Typ, { VECTors|DOTS }.

Rückgabeformat: Die Abfrage gibt „VECTors“ oder „DOTS“ zurück.

Beispiel:

Der folgende Befehl stellt den Anzeigemodus für die Wellenform auf „DOTS“ ein.

:DISPlay:WAVEform DOTS

Die folgende Abfrage gibt „DOTS“ zurück.

:DISPlay:WAVEform?

~~DISPLAY~~

Funktion: Stellt die Helligkeit der Wellenformanzeige auf dem Bildschirm ein.

Format: :DISPlay:BRIGhtness <time>

:DISPlay:BRIGhtness?

Dabei ist <time> eine ganze Zahl zwischen 0 und 100.

Rückgabeformat: Die Abfrage gibt eine ganze Zahl zurück.

Beispiel:

Der folgende Befehl stellt die Helligkeit der Wellenformanzeige auf 80 ein.

:DISPlay:BRIGhtness 80

Die folgende Abfrage gibt „80“ zurück.

:DISPlay:BRIGhtness?

~~DISPLAY~~

Funktion: Legt den auf dem Bildschirm angezeigten Gittertyp fest.

Format: :DISPlay:GRATicule <Typ>

:DISPlay:GRATicule?

Dabei ist <Typ> ein diskreter Typ: { FULL|GRID|RETical|FRAMe }.

Rückgabeformat: Die Abfrage gibt „FULL“, „GRID“, „RETical“ oder „FRAMe“ zurück.

Beispiel:

Der folgende Befehl setzt den Bildschirmrastertyp auf „FULL“.

```
:DISPlay:GRATicule FULL
```

Die folgende Abfrage gibt „FULL“ zurück.

```
:DISPlay:GRATicule?
```

Intensity

Funktion: Stellt die Helligkeit der Gitteranzeige auf dem Bildschirm ein.

Format: :DISPlay:INTensity <time>

```
:DISPlay:INTensity?
```

Dabei ist <time> eine ganze Zahl zwischen 0 und 100.

Rückgabeformat: Die Abfrage gibt eine ganze Zahl zurück.

Beispiel:

Der folgende Befehl stellt die Helligkeit des Bildschirmrasters auf 80 ein.

```
:DISPlay:INTensity 80
```

Die folgende Abfrage gibt „80“ zurück.

:DISPlay:INTensity?

PERSist

3.2.7.5.1 :DISPlay:PERSist:MODE

Funktion: Legt den Anzeigemodus für die Persistenz fest.

Format: :DISPlay:PERSist:MODE <Modus>

:DISPlay:PERSist:MODE?

Dabei ist <mode> ein diskreter Wert: { AUTO|NORMAl|INFinite|none }.

3.2.7.5.2 :DISPlay:PERSist:ADJust

Funktion: Festlegen der Persistenzzeit im normalen Anzeigemodus

Format: :DISPlay:PERSist:ADJust <time>

:DISPlay:PERSist:ADJust?

Dabei ist <time> eine ganze Zahl in Millisekunden: 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000

3.2.7.5.3 :DISPlay:PERSist:CLEar

Funktion: Nachleuchtanzeige löschen

Format: :DISPlay:PERSist:CLEar

HIGH (gilt für Geräte mit unabhängigem Hochaktualisierungsmodus)

Funktion: Hohe Bildwiederholrate ein- oder ausschalten

Format: :DISPlay:HIGH <bool>

:DISPlay:HIGH?

Dabei ist <bool> vom Typ „Boolean“ und kann die Werte {{0|OFF}}{{1|ON}} annehmen.

~~18~~ HORRef

Funktion: Legt den Modus für die horizontale Bildschirmerweiterung fest:

„Triggerpunkt“ oder „Bildschirmmitte“.

Format: :DISPlay:HORRef <Modus>

:DISPlay:HORRef?

Dabei ist <Modus> ein diskreter Wert und kann { CENTER|TRIGpos } annehmen.

~~19~~ ZOOM

Funktion: ZOOM öffnen oder schließen

Format: :DISPlay:ZOOM <bool>

:DISPlay:ZOOM?

Dabei ist <bool> vom Typ „Boolean“ und kann die Werte {{0|OFF}}{{1|ON}} annehmen.

~~20~~ CCT

Funktion: Farbtemperaturanzeige ein- oder ausschalten

Format: :DISPlay:CCT <bool>

:DISPlay:CCT?

Dabei ist <bool> ein boolescher Wert, der die Werte {{0|OFF}}{{1|ON}} annehmen kann.

3.2.8 Subsystem für Messbefehle

~~MEASURE, OPEN~~

Funktion: Hinzufügen von Messelementen der angegebenen Kanäle auf der Schnittstelle

Format: :MEASure:OPEN [<item>,< n1> , < n2> ,< src1> ,< src2>]

Hinweis: <item> ist diskret und bezieht sich auf die Messgröße: { PERiod | FREQ | RISE

time | FALL time | DELAy | PDUTy | NDUTy | PWIDth | NWIDth

|BURStw|ROV|FOV|PHASe|PKPK|AMP|HIGH|LOW|MAX|MIN|RMS|CRMS|MEAN|CM EAn

|ACRMS |+RATE|-RATE}.

<n1> bezieht sich auf die Quelle, diskreter Typ,

{CH1|CH2|CH3|CH4|R1|R2|R3|R4| MATH }.

<n2> bezieht sich auf die Quelle, gültig bei DELay und PHASE, diskreter Typ,

{CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

<src1> bezieht sich auf die Parameter bestimmter Messgrößen, gültig bei DELay,

diskreter Typ, [FRISe|FFALL|LRISe|LFALL].

<src2> bezieht sich auf die Parameter bestimmter Messgrößen, gültig bei DELay,

diskreter Typ, [FRISe|FFALL|LRISe|LFALL].

Beispiel:

Die folgenden Einstellungen aktivieren die Periodenmessung für Kanal 1 auf dem Bildschirm

:MEASure:OPEN PERiod, CH1

die erste steigende Flanke von Kanal 2 bis zur ersten steigenden Flanke von Kanal 3 auf dem Bildschirm.

:MEASure:OPEN DELay CH2,CH3,FRISe,FRISe

2 ~~MEASure:<n1>,<n2>,<src1>,<src2>~~

Funktion: Abfrage des Werts des Open-Messelements

Format: :MEASure:<item>? [< n 1 > , < n 2> ,< src1> ,< src2>]

Hinweis: Der diskrete Typ <item> bezieht sich auf das Messelement: { PEROID | FREQ |

RISE time | FALL time | DELAY | PDUTy | NDUTy | PWIDth | NWIDth

|BURStw|ROV|FOV|PHASe|PKPK

|AMP|HIGH|LOW|MAX|MIN|RMS|CRMS|MEAN|CMEAN| ACRMS |+RATE|-RATE-}.

< n1> bezieht sich auf die Quelle, diskreter Typ, {CH1|CH2|CH3|CH4|R1|R2|R3|R4|

MATH}.

<n2> bezieht sich auf die Quelle, die gültig ist, wenn DELay und PHASe gelten, und zu anderen Zeiten ungültig ist. Sie muss nicht angegeben werden, diskreter Typ,

{CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

<src1> bezieht sich auf die Parameter bestimmter Messgrößen, gültig bei DELay, diskreter Typ, [FRISe|FFALL|LRISe|LFALL].

<src2> bezieht sich auf die Parameter bestimmter Messgrößen, die bei DELAY gültig sind, diskreter Typ, [FRISe|FFALL|LRISe|LFALL].

Beispiel: Wenn die Periodenmessung von Kanal 1 aktiviert ist, werden mit den folgenden Einstellungen die Periodenmesswerte von Kanal 1 abgefragt

:MEASure:PERiod? CH1

2 ~~MEASure:CLOSE~~

Funktion: Löscht die Messelemente des angegebenen Kanals im geöffneten Zustand auf der Schnittstelle

Format: :MEASure:CLOSE [<item>,< n1> , < n2> ,< src1> ,< src2>]

Hinweis: <item> ist diskret und bezieht sich auf die Messgröße: { PEROID | FREQ | RISE time | FALL time | DELAY | PDUTy | NDUTy | PWIDth | NWIDth |BURStw|ROV|FOV|PHASe|PKPK|AMP|HIGH|LOW|MAX|MIN|RMS|CRMS|MEAN|CMEAN|ACRMS|+RATE|-RATE}.

<n1> bezieht sich auf die Quelle, diskreter Typ, {CH1|CH2|CH3|CH4|R1|R2|R3|R4| MATH }.

<n2> bezieht sich auf die Quelle, die gültig ist, wenn DELay und PHASe (diskreter Typ) {CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH} gültig sind.

<src1> bezieht sich auf die Parameter bestimmter Messgrößen und ist gültig, wenn DELAY, diskreter Typ, [FRISe|FFALL|LRISe| LFALL] ist.

< src2> bezieht sich auf die Parameter bestimmter Messgrößen und ist gültig, wenn DELAY vom diskreten Typ ist, [FRISe|FFALL|LRISe| LFALL].

Beispiel:

Die folgenden Einstellungen deaktivieren die Periodenmessung von Kanal 1 auf dem Bildschirm.

:MEASure:CLOSe PERiod, CH1

Funktion: Alle Elemente in den geöffneten Messelementen löschen.

Format: :MEASure:CLEar <item>

Dabei ist <item> eine diskrete Variable, { item1| item 2| item 3| item 4| item 5| item 6| item 7| item 8|item9| item 10| all }.

1 bis 10 entsprechen den 10 Messoptionen auf dem Bildschirm.

~~MEAS~~STATistic

3.2.8.5.1 :MEASure:STATistic:DISPlay

Funktion: Schaltet die Statistikfunktion ein oder aus.

Format: :MEASure:STATistic:DISPlay <bool>

:MEASure:STATistic:DISPlay?

Dabei ist <bool> ein boolescher Typ, {{0|OFF}}{{1|ON}}.

3.2.8.5.2 :MEASure:STATistic:RESet

Funktion: Historische Statistiken löschen und neu zählen.

Format: :MEASure:STATistic:RESet

3.2.8.5.3 :MEASure:STATistic:MEAN <bool>

Funktion: Anzeige des Durchschnittswerts in den Statistiken ein- oder ausschalten

Format: :MEASure:STATistic:MEAN <bool>

:MEASure:STATistic:MEAN?

Dabei ist <bool> vom Typ „Boolean“ und kann die Werte {{0|OFF}}{{1|ON}} annehmen.

3.2.8.5.4 :MEASure:STATistic:MAX <bool>

Funktion: Aktiviert oder deaktiviert die Anzeige des Maximalwerts in der Statistik

Format: :MEASure:STATistic:MAX <bool>

:MEASure:STATistic:MAX?

Darunter <bool>, Boolescher Typ, {{0|OFF}}{{1|ON}}.

3.2.8.5.5 :MEASure:STATistic:MIN <bool>

Funktion: Aktiviert oder deaktiviert die Anzeige des Minimalwerts in der Statistik

Format: :MEASure:STATistic:MIN <bool>

:MEASure:STATistic:MIN?

Darunter <bool>, Boolescher Typ, {{0|OFF}}{{1|ON}}.

3.2.8.5.6 :MEASure:STATistic:DEV <bool>

Funktion: Aktiviert oder deaktiviert die Anzeige des mittleren quadratischen Fehlers in der Statistik

Format: :MEASure:STATistic:DEV <bool>

:MEASure:STATistic:DEV?

Dabei ist <bool> ein boolescher Wert, der die Werte {{0|OFF}}{{1|ON}} annehmen kann.

3.2.8.5.7 :MEASure:STATistic:COUNt <bool>

Funktion: Aktiviert oder deaktiviert die Anzeige der Zählung in der Statistik

Format: :MEASure:STATistic:COUNt <bool>

:MEASure:STATistic:COUNt?

Darunter <bool>, Boolescher Typ, {{0|OFF}}{{1|ON}}.

3.2.8.5.8 :MEASure:STATistic:VIEW?

Funktion: Alle Werte statistischer Elemente abfragen (gültig, wenn die Statistikfunktion aktiviert ist)

Format: :MEASure:STATistic:VIEW? <item>, <source>

Dabei ist <item> das geöffnete Messelement und <source> ein diskreter Typ, der die Messquelle angibt: {CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

Hinweis: <source> kann weggelassen werden; standardmäßig wird die aktuell aktive Kanalquelle verwendet

vom Oszilloskop festgelegt wurde.

Ausgabeformat: Die von der Abfrage zurückgegebenen Werte werden in wissenschaftlicher Notation angezeigt, gefolgt vom aktuellen Wert, dem Durchschnittswert, dem Maximalwert, dem Minimalwert, dem quadratischen Mittelwert und der Anzahl.

Beispiel: Die folgende Abfrage gibt die statistischen Daten des Spitzen-Spitzen-Messwerts von Kanal 1 zurück, wie z. B.: 1,00000 7 e-02, 1,00000 5 e-02, 1,00000 9 e-02, 1,00000 1 e-02, 1,000000e-02, 1,75e02,

:MEASure:STATistic:VIEW? PKPK, CH1

Wenn die vom Oszilloskop aktuell eingestellte Messquelle ebenfalls Kanal 1 ist, verwenden Sie direkt den folgenden Befehl:

:MEASure:STATistic:VIEW? PKPK

3.2.8.5.9 :MEASure:STATistic:MEAN:VIEW?

Funktion: Abfrage des Mittelwerts statistischer Größen (gültig, wenn die Statistikfunktion aktiviert ist)

Format: :MEASure:STATistic:MEAN:VIEW? <item> , <source>

Dabei ist <item> das geöffnete Messelement und <source> der diskrete Typ der Messquelle {CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

Hinweis: <source> kann weggelassen werden; standardmäßig wird die aktuell aktive Kanalquelle verwendet

vom Oszilloskop festgelegt wurde.

Ausgabeformat: Der von der Abfrage zurückgegebene Wert wird in wissenschaftlicher Notation angezeigt.

Beispiel: Die folgende Abfrage liefert die statistische Auswertung des Spitzen-Spitzen-Messwerts von Kanal 1. Mittelwert, z. B.: 1,00000 7 e-02

:MEASure:STATistic:MEAN:VIEW? PKPK, CH1

Wenn die vom Oszilloskop aktuell eingestellte Messquelle ebenfalls Kanal 1 ist, verwenden Sie direkt den folgenden Befehl:

:MEASure:STATistic:MEAN:VIEW? PKPK,

3.2.8.5.10 :MEASure:STATistic:MAX:VIEW?

Funktion: Abfrage des Maximalwerts statistischer Größen (gültig, wenn die Statistikfunktion aktiviert ist)

Format: :MEASure:STATistic:MAX:VIEW? <item> , <source>

Dabei ist <item> das geöffnete Messelement und <source> der diskrete Typ der Messquelle {CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

Hinweis: <source> kann weggelassen werden; standardmäßig wird die aktuell vom Oszilloskop angegebene Kanalquelle verwendet.

Ausgabeformat: Der von der Abfrage zurückgegebene Wert wird im wissenschaftlichen

Darstellung.

Beispiel: Die folgende Abfrage gibt die statistische Auswertung des Spitze-Spitze-Messwerts von Kanal 1 zurück. Maximalwert, z. B.: 1,000007 e-02

:MEASure:STATistic:MAX:VIEW? PKPK, CH1

Wenn die aktuell vom Oszilloskop eingestellte Messquelle ebenfalls Kanal 1 ist, verwenden Sie direkt den folgenden Befehl:

:MEASure:STATistic:MAX:VIEW? PKPK,

3.2.8.5.11 :MEASure:STATistic:MIN:VIEW?

Funktion: Abfrage des Minimalwerts statistischer Größen (gültig, wenn die Statistikfunktion aktiviert ist)

Format: :MEASure:STATistic:MIN:VIEW? <item> , <source>

Dabei ist <item> das geöffnete Messelement und <source> der diskrete Typ der Messquelle {CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

Hinweis: <source> kann weggelassen werden; als Standardwert gilt die aktuell vom Oszilloskop angegebene Kanalquelle.

Rückgabeformat: Der von der Abfrage zurückgegebene Wert wird in wissenschaftlicher Notation angezeigt.

Beispiel: Die folgende Abfrage liefert die statistische Berechnung des Verhältnisses von Spitzenwert zu

Spitzen-zu-Spitzen-Messwert von Kanal 1. Minimalwert, z. B.: 1,000007e-02

:MEASure:STATistic:MIN:VIEW? PKPK, CH1

Wenn die aktuell vom Oszilloskop eingestellte Messquelle ebenfalls Kanal 1 ist, verwenden Sie direkt den folgenden Befehl:

:MEASure:STATistic:MIN:VIEW? PKPK,

3.2.8.5.12 :MEASure:STATistic:DAV:VIEW?

Funktion: Abfrage des mittleren quadratischen Fehlers statistischer Elemente (gültig, wenn die Statistikfunktion aktiviert ist)

Format: :MEASure:STATistic:DAV:VIEW? <item>, <source>

Dabei ist <item> das aktivierte Messelement,
<source> der diskrete Typ und die Messquelle lautet
{CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

Hinweis: <source> kann weggelassen werden; der Standardwert ist die aktuell vom Oszilloskop angegebene Kanalquelle.

Rückgabeformat: Der Rückgabewert der Abfrage wird in wissenschaftlicher Notation angezeigt.

Beispiel: Die folgende Abfrage gibt das statistische Ergebnis der Spitze-Spitze-Messwert von Kanal 1 als mittlerer quadratischer Fehler, z. B. 1,000007 e-02

:MEASure:STATistic:DAV:VIEW? PKPK, CH1

Wenn die Messquelle der aktuellen Oszilloskop-Einstellung ebenfalls Kanal 1 ist, verwenden Sie direkt den folgenden Befehl:

:MEASure:STATistic:DAV:VIEW? PKPK,

3.2.8.5.13 :MEASure:STATistic:COUNt:VIEW?

Funktion: Abfrage der statistischen Größe von statistischen Elementen (gültig, wenn die Statistikfunktion aktiviert ist)

Format: :MEASure:STATistic:COUNt:VIEW? <item> , <source>

Dabei ist <item> das geöffnete Messelement und <source> der diskrete Typ sowie die Messquelle {CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

Hinweis: <source> kann weggelassen werden; als Standardwert gilt die aktuell vom Oszilloskop angegebene Kanalquelle.

Rückgabeformat: Der Rückgabewert der Abfrage wird in wissenschaftlicher Notation angezeigt.

Beispiel: Die folgende Abfrage gibt die statistische Berechnung des Spitze-Spitze-Messwerts von Kanal 1 zurück. Statistische Größe, z. B.: 1,000007e-02

:MEASure:STATistic:COUNt:VIEW? PKPK, CH1

Wenn die Messquelle der aktuellen Oszilloskop-Einstellung ebenfalls Kanal 1 ist, verwenden Sie direkt den folgenden Befehl:

:MEASure:STATistic:COUNT:VIEW? PKPK,

3.2.8.5.14 :MESSUNG:STATISTIK:AKTUELL:ANSICHT?

Funktion: Abfrage des aktuellen Werts statistischer Elemente (gültig, wenn die Statistikfunktion aktiviert ist)

Format: :MEASure:STATistic:CURRent:VIEW? <item> , <source>

Dabei ist <item> das geöffnete Messelement und <source> der diskrete Typ sowie die Messquelle {CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

Hinweis: <source> kann weggelassen werden; als Standardwert gilt die aktuell vom Oszilloskop angegebene Kanalquelle.

Rückgabeformat: Der von der Abfrage zurückgegebene Wert wird in wissenschaftlicher Notation angezeigt.

Beispiel: Die folgende Abfrage gibt die statistische Auswertung des Spitze-Spitze-Messwerts von Kanal 1 zurück. Durchschnittswert, z. B.: 1,000007e-02

:MEASure:STATistic:CURRent:VIEW? PKPK, CH1

Wenn die aktuelle Messquelle des Oszilloskops ebenfalls Kanal 1 ist, verwenden Sie direkt den folgenden Befehl:

:MEASure:STATistic:CURRent:VIEW? PKPK,

~~MEAS~~ADISplay

Funktion: Alle Messungen ein- oder ausschalten.

Format: :MEASure:ADISplay <bool>

:MEASure:ADISplay?

Dabei ist <bool> vom Typ „Boolean“ und kann die Werte {{0|OFF}}|{{1|ON}} annehmen.

~~MEAS~~COUNT

3.2.8.7.1 :MEASure:COUNter:SOURce

Funktion: Die Messquelle des Zählers festlegen oder abfragen.

:MEASure:COUNter:SOURce <sour>

:MEASure:COUNter:SOURce?

Dabei ist <sour> ein diskretes Wert, {CLOSE|CH1|CH2|CH3|CH4}.

3.2.8.7.2 :MEASure:COUNter:MODE <mode>

3.2.8.7.3 :MEASure:COUNter:VALue?

Funktion: Abfrage des Messergebnisses des Zählers.

:MEASure:COUNter:VALue?

Die Abfrage gibt den aktuellen Messwert in wissenschaftlicher Notation zurück. Ist die Frequenzzählerfunktion derzeit nicht aktiviert, wird 0,0000000e+00 zurückgegeben.

3.2.9 Trigger-Befehlssystem



Funktion: Triggertyp auswählen.

Format: :TRIGger:TYPE <Typ>

:TRIGger:TYPE?

Dabei ist <Typ> ein diskreter Typ:

{EDGE|PULSe|LOGic|NEDGE|DWARt|SLOPe|TIMEout|VIDeo|S1|S2}

Rückgabeformat: Die Abfrage gibt den aktuell verwendeten Triggertyp zurück.

Beispiel:

Der folgende Befehl wählt die Flankentriggerung aus.

:TRIGger:TYPE EDGE

Die folgende Abfrage gibt „EDGE“ zurück.

:TRIGger:TYPE?



Funktion: Trigger-Verzögerungszeit festlegen.

Format: :TRIGger:HOLDoff <Wert>

:TRIGger:HOLDoff?

Dabei ist <Wert> eine reelle Zahl im Bereich von 200 ns bis 10 s.

Rückgabeformat: Die **Abfrage** gibt die Trigger-Verzögerungszeit in wissenschaftlicher Notation **zurück**.

Beispiel:

Der folgende Befehl setzt die Trigger-Verzögerungszeit auf 200 ns.

:TRIGger:HOLDoff 0,0000002

Die folgende Abfrage gibt „2,000000e-07“ zurück.

:TRIGger:HOLDoff?

MODE

Funktion: Legt den Auslösemodus fest: „automatisch“ oder „normal“.

Format: :TRIGger:MODE <Modus>

:TRIGger:MODE?

Dabei ist <Modus> ein diskreter Wert und kann { AUTO|NORMal } annehmen.

Rückgabeformat: Die Abfrage gibt „AUTO“ oder „NORMal“ zurück.

Beispiel:

Der folgende Befehl wählt den automatischen Trigger-Modus aus.

:TRIGGER:MODE AUTO

Die folgende Abfrage gibt „AUTO“ zurück.

:TRIGger:MODE?

STATus

Funktion: Abfrage des aktuellen Trigger-Status.

Format: :TRIGger:STATus?

Rückgabeformat: Die Abfrage gibt „RUN“, „WAIT“, „AUTO“ und „STOP“ zurück.

EDGE

3.2.9.5.1 :TRIGger:EDGE:SOURce

Funktion: Auswahl der Triggerquelle für den Flankentrigger.

Format: :TRIGger:EDGE:SOURce <source>

:TRIGger:EDGE:SOURce?

Dabei ist <source> ein diskreter Typ mit den Werten {CH1|CH2|CH3|CH4}.

Rückgabeformat: Die Abfrage gibt „CH1“, „CH2“, „CH3“ oder „CH4“ zurück.

Beispiel:

Der folgende Befehl legt Kanal 1 als Triggerquelle fest.

:TRIGger:EDGE:SOURce CH1

Die folgende Abfrage gibt „CH1“ zurück.

:TRIGger:EDGE:SOURce?

3.2.9.5.2 :TRIGger:EDGE:SLOPe

Funktion: Wählt den Flankentyp des Flankentriggers aus.

Format: :TRIGger:EDGE:SLOPe <Kante>

:TRIGger:EDGE:SLOPe?

Dabei ist <edge> ein diskreter Typ: {RISE|FALL|DUAL}.

Rückgabeformat: Die Abfrage gibt „RISE“, „FALL“ oder „DUAL“ zurück.

Beispiel:

Der folgende Befehl wählt die Auslösung durch eine steigende Flanke aus.

:TRIGger:EDGE:SLOPe RISE

Die folgende Abfrage gibt „RISE“ zurück.

:TRIGger:EDGE:SLOPe?

3.2.9.5.3 :TRIGger:EDGE:LEVeL

Funktion: Festlegen des Triggerpegels bei Flankentriggerung

Format: :TRIGger:EDGE:LEVeL <Pegel>

:TRIGger:EDGE:LEVeL?

Dabei ist <level> ein reeller Wert.

Rückgabeformat: Die Abfrage gibt den Wert des Triggerpegels in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt den Triggerpegel auf 150 mV.

```
:TRIGger:EDGE:LEVel 0,15
```

Die folgende Abfrage gibt „1,500000e-01“ zurück.

```
:TRIGger:EDGE:LEVel?
```

3.2.9.5.4 :TRIGger:EDGE:COUPle

Funktion: Legt den Kopplungsmodus für den Flankentrigger fest.

Format: :TRIGger:EDGE:COUPle <Kopplung>

```
:TRIGger:EDGE:COUPle?
```

Dabei ist <Kopplung> ein diskreter Wert: { DC|AC|HFRej|LFRej|Noiserej }.

Rückgabeformat: Die Abfrage gibt „DC“, „AC“, „HFRej“, „LFRej“ oder „Noiserej“ zurück.

Beispiel:

Der folgende Befehl wählt den Gleichstromkopplungsmodus aus.

```
:TRIGger:EDGE:COUPle DC
```

Die folgende Abfrage gibt „DC“ zurück.

:TRIGger:EDGE:COUPle?

PULSe

3.2.9.6.1 :TRIGger:PULSe:SOURce

Funktion: Legt die Triggerquelle für den Pulsbreiten-Trigger fest.

Format: :TRIGger:PULSe:SOURce <source>

:TRIGger:PULSe:SOURce

Darunter <source>, diskreter Typ, {CH1|CH2|CH3|CH4}.

Rückgabeformat: Die Abfrage gibt „CH1“, „CH2“, „CH3“ oder „CH4“ zurück.

Beispiel:

Der folgende Befehl legt Kanal 1 als Triggerquelle fest.

:TRIGger:PULSe:SOURce CH1

Die folgende Abfrage gibt „CH1“ zurück.

:TRIGger:PULSe:SOURce?

3.2.9.6.2 :TRIGger:PULSe:POLarity

Funktion: Legt die Polarität der Impulsbreitenauslösung fest.

Format: :TRIGger:PULSe:POLarity <Polarität>

:TRIGger:PULSe:POLarity?

Dabei ist <polarity> ein diskreter Typ, { POSitive|NEGative }.

Rückgabeformat: Die Abfrage gibt „POSitive“ oder „NEGative“ zurück.

Beispiel:

Der folgende Befehl wählt die Auslösung bei steigender Flanke aus.

```
:TRIGger:PULSe:POLarity POSitive
```

Die folgende Abfrage gibt „POSitive“ zurück.

```
:TRIGger:PULSe:POLarity?
```

3.2.9.6.3 :TRIGger:PULSe:WIDTh

Funktion: Legt den Wert für die Impulsbreite bei der Impulsbreitenauslösung fest.

Format: :TRIGger:PULSe:WIDTh <Breite>

```
:TRIGger:PULSe:WIDTh?
```

Dabei ist <Breite> ein Realwert im Bereich von 40 ns bis 10 s.

Rückgabeformat: Die Abfrage gibt reelle Zahlen zurück.

Beispiel:

Der folgende Befehl setzt den Wert für die Impulsbreite auf 4 ns.

```
:TRIGger:PULSe:WIDTh 4.000000e-08
```

Die folgende Abfrage gibt „4,000000e-08“ zurück.

:TRIGger:PULSe:WIDTh?

3.2.9.6.4 :TRIGger:PULSe:CONDition

Funktion: Triggerbedingungen für die Impulsbreite festlegen.

Format: :TRIGger:PULSe:CONDition <Bedingung>

:TRIGger:PULSe:CONDition?

Dabei ist <Bedingung> ein diskreter Typ: { GREat|LESS|EQUAL|UNEQual }.

GREat: Die Impulsbreite des Oszilloskop-Eingangssignals ist größer als die angegebene Impulsbreite;

LESS: Die Impulsbreite des Oszilloskop-Eingangssignals ist kleiner als die angegebene Impulsbreite;

EQUAL: Die Impulsbreite des Oszilloskop-Eingangssignals entspricht der angegebenen Impulsbreite;

UNEQual: Die Impulsbreite des Oszilloskop-Eingangssignals ist nicht gleich der angegebenen Impulsbreite;

3.2.9.6.5 :TRIGger:PULSe:LEVel

Funktion: Festlegen des Triggerpegels für die Pulsbreitenauslösung

Format: :TRIGger:PULSe:LEVel <Pegel>

:TRIGger:PULSe:LEVel?

Dabei ist <level> ein reeller Wert.

Rückgabeformat: Die Abfrage gibt den Wert des Triggerpegels in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt den Triggerpegel auf 150 mV.

```
:TRIGger:PULSe:LEVel 0,15
```

Die folgende Abfrage gibt „1,500000e-01“ zurück.

```
:TRIGger:PULSe:LEVel?
```



3.2.9.7.1 :TRIGger:LOGic:STATus

Funktion: Legt den Logikstatus jedes Kanals im Logik-Trigger fest

Format: :TRIGger:LOGic:STATus <Kanal>,<Status>

```
:TRIGger:LOGic:STATus? <Kanal>
```

Dabei ist <Kanal> ein diskreter Typ, {CH1|CH2|CH3|CH4}. <Status> ist ein diskreter Typ, {HIGH|LOW|NONE}.

3.2.9.7.2 :TRIGger:LOGic:FUNCTion

Funktion: Legt die Vergleichsfunktion des logischen Triggers fest.

Format: :TRIGger:LOGic:FUNCTion <function>

:TRIGger:LOGic:FUNctIon?

Dabei ist <function> ein diskreter Typ und kann „AND“, „OR“, „NAND“ oder „NOR“ sein kann.

3.2.9.7.3 :TRIGger:LOGic:CONDition Funktion:

Legt die logische Auslösebedingung fest.

Format: :TRIGger:LOGic:CONDition <Bedingung>

:TRIGger:LOGic:CONDition?

Dabei ist <Bedingung> ein diskreter Wert: { GREat|LESS|EQUal|UNEQual|TRUE|FALSe }.

GREat: Wird ausgelöst, wenn der logische Zustand länger als die Auslösezeit wahr ist;

LESS: Wird ausgelöst, wenn der logische Zustand kürzer als die Triggerlogikzeit wahr ist;

EQUal: Ausgelöst, wenn der logische Zustand wahr ist und die Haltezeit der Auslösezeit entspricht;

UNEQual: Wird ausgelöst, wenn die Haltezeit des logischen Zustands „wahr“ nicht gleich der Triggerlogikzeit ist;

TRUL: Wird ausgelöst, wenn der logische Zustand „wahr“ ist;

FALSe: Wird ausgelöst, wenn der logische Zustand „falsch“ ist.

3.2.9.7.4 :TRIGger:LOGic:TIME

Funktion: Legt die Triggerlogikzeit fest.

Format: :TRIGger:LOGic:TIME <time>

:TRIGger:LOGic:TIME?

Dabei ist <time> ein reeller Wert im Bereich von 200 ns bis 10 s.

3.2.9.7.5 :TRIGger:LOGic:LEVel

Funktion: Legt den Triggerpegel jedes Kanals bei logischem Trigger fest

Format: :TRIGger:LOGic:LEVel <Kanal>,<Pegel>

:TRIGger:LOGic:LEVel? <Kanal>

Dabei ist <Kanal> ein diskreter Typ, {CH1|CH2|CH3|CH4}; <Pegel> ist ein reeller Wert.

Runt

3.2.9.8.1 :TRIGger:Runt:SOURce

Funktion: Legt die Triggerquelle des Runt-Triggers fest.

Format: :TRIGger:Runt:SOURce <source>

:TRIGger:Runt:SOURce?

Dabei ist <source> ein diskretes Wert, {CH1|CH2|CH3|CH4}.

3.2.9.8.2 :TRIGger:Runt:POLArity

Funktion: Legt die Impulspolarität des Runt-Triggers fest.

Format: :TRIGger:Runt:POLArity <Polarität>

:TRIGger:Runt:POLArity?

Dabei ist <polarity> ein diskreter Typ, { POSItive|NEGAtive|EITHer }.

3.2.9.8.3 :TRIGger:Runt:CONDition

Funktion: Legt die Bedingung für die Impulsbreitenbegrenzung fest.

Format: :TRIGger:Runt:CONDition <Bedingung>

:TRIGger:Runt:CONDition?

Dabei ist <Bedingung> vom diskreten Typ { GREAt|LESS|BETWeen|NONE }.

GREAt: Die Impulsbreite des Oszilloskop-Eingangssignals ist größer als die angegebene Impulsbreite;

LESS: Die Impulsbreite des Oszilloskop-Eingangssignals ist kleiner als die angegebene Impulsbreite;

BETWeen: Die Impulsbreite des Oszilloskop-Eingangssignals liegt zwischen den angegebenen Impulsbreiten;

NONE: irrelevant;

3.2.9.8.4 :TRIGger:Runt:HTIME

Funktion: Legt die obere Zeitgrenze für den Runt-Trigger fest.

Format: :TRIGger:Runt:HTIME <time>

:TRIGger:Runt:HTIME?

Dabei ist <time> ein reeller Wert im Bereich von 8 ns bis 10 s.

3.2.9.8.5 :TRIGger:Runt:LTIME

Funktion: Legt die untere Grenze der Runt-Triggerzeit fest.

Format: :TRIGger:Runt:LTIME <time>

:TRIGger:Runt:LTIME?

Dabei ist <time> ein reeller Wert im Bereich von 8 ns bis 10 s.

3.2.9.8.6 :TRIGger:Runt:BTIME

Funktion: Legt das Zeitintervall für den Runt-Trigger fest.

Format: :TRIGger:Runt:BTIME < htime >, < ltime >

:TRIGger:Runt:BTIME? <Typ>

Dabei sind < htime > und < ltime > reelle Werte im Bereich von 8 ns bis 10 s. (high > low)

< type >, diskreter Typ, { HIGH|LOW }

3.2.9.8.7 :TRIGger:Runt:HLEVEL

Funktion: Legt den hohen Pegel während des Runt-Triggers fest.

Format: :TRIGger:Runt:HLEVEL <Pegel>

:TRIGger:Runt:HLEVEl?

Dabei ist <level> ein Realwert.

3.2.9.8.8 :TRIGger:Runt:LLEVEl

Funktion: Legt den Low-Pegel fest, wenn der Runt-Trigger ausgelöst wird.

Format: :TRIGger:Runt:LLEVEl <level>

:TRIGger:Runt:LLEVEl?

Dabei ist <level> ein reeller Wert. (H LEV el > LLEV el)



3.2.9.9.1 :TRIGger:SLOPe:SOURce

Funktion: Legt die Auslösequelle des Steigungsauslösers fest.

Format: :TRIGger:SLOPe:SOURce <source>

:TRIGger:SLOPe:SOURce?

Dabei ist <source> ein diskreter Typ: {CH1|CH2|CH3|CH4}.

3.2.9.9.2 :TRIGger:SLOPe:EDGE

Funktion: Legen Sie die Flankenform des Steigungstriggers fest.

Format: :TRIGger:SLOPe:EDGE <Kante>

:TRIGger:SLOPe:EDGE?

Dabei ist <Kante> diskret, { RISE|FALL|EITHer }.

3.2.9.9.3 :TRIGger:SLOPe:CONDition

Funktion: Legt die Grenzbedingungen für die Steigungsauslösung fest.

Format: :TRIGger:SLOPe:CONDition <Bedingung>

:TRIGger:SLOPe:CONDition?

Dabei ist <Bedingung> diskret: { GREat|LESS|BETWeen }.

GREat: Die Steigung des Oszilloskop-Eingangssignals ist größer als die angegebene Zeiteinstellung;

LESS: Die Steigung des Oszilloskop-Eingangssignals ist kleiner als der festgelegte Zeitwert;

BETWeen: Die Steigung des Oszilloskop-Eingangssignals ist größer als die festgelegte obere Zeitgrenze und kleiner als die festgelegte untere Zeitgrenze.

3.2.9.9.4 :TRIGger:SLOPe:HTIME

Funktion: Festlegen der oberen Grenze der Steigungs-Triggerzeit.

Format: :TRIGger:SLOPe:HTIME <time>

:TRIGger:SLOPe:HTIME?

Dabei ist <time> ein reeller Wert im Bereich von 8 ns bis 10 s.

3.2.9.9.5 :TRIGger:SLOPe:LTIME

Funktion: Legt die untere Grenze der Triggerzeit für die Flankensteilheit fest.

Format: :TRIGger:SLOPe:LTIMe <time>

:TRIGger:SLOPe:LTIMe?

Dabei ist <time> ein reeller Wert im Bereich von 8 ns bis 10 s.

3.2.9.9.6 :TRIGger:SLOPe:HLEVel

Funktion: Legt den hohen Pegel fest, bei dem der Steigungsauslöser ausgelöst wird.

Format: :TRIGger:SLOPe:HLEVel <level>

:TRIGger:SLOPe:HLEVel?

Dabei ist <level> ein reeller Wert.

3.2.9.9.7 :TRIGger:SLOPe:LLEVel

Funktion: Legt den unteren Schwellenwert fest, bei dem die Steigung ausgelöst wird.

Format: :TRIGger:SLOPe:LLEVel <level>

:TRIGger:SLOPe:LLEVel?

Dabei ist <level> ein Realwert. (HLEV el > LLEV el)

3.2.9.10 :TIMeout

3.2.9.10.1 :TRIGger:TIMeout:SOURce

Funktion: Legt die Auslösequelle des Timeout-Triggers fest.

Format: :TRIGger:TIMeout:SOURce <source>

:TRIGger:TIMEout:SOURce?

Dabei ist <source> ein diskreter Typ: {CH1|CH2|CH3|CH4}

3.2.9.10.2 :TRIGger:TIMEout:POLarity

Funktion: Legt die Polarität des Timeout-Triggers fest.

Format: :TRIGger:TIMEout:POLarity <polarity>

:TRIGger:TIMEout:POLarity?

Davon ist <Polarität> vom diskreten Typ { POSitiv|NEGativ|EITHer}.

3.2.9.10.3 :TRIGger:TIMEout:TIME

Funktion: Festlegen der Timeout-Dauer für den Timeout-Trigger.

Format: :TRIGger:TIMEout:TIME <time>

:TRIGger:TIMEout:TIME?

Dabei ist <time> vom Typ „real“ und liegt im Bereich von 8 ns bis 10 s.

3.2.9.10.4 :TRIGger:TIMEout:LEVel

Funktion: Legt den Übertrigger-Pegel für den Timeout-Trigger fest.

Format: :TRIGger:TIMEout:LEVel <Pegel>

:TRIGger:TIMEout:LEVel?

Dabei beziehen sich <Pegel>, vom Typ „real“, und der Bereich auf das Datenblatt.



3.2.9.11.1 :TRIGger:NEDGE:SOURce

Funktion: Legt die Triggerquelle für den N-te-Flanke-Trigger fest.

Format: :TRIGger:NEDGE:SOURce <source>

:TRIGger:NEDGE:SOURce?

Dabei ist <source> ein diskreter Typ: {CH1|CH2|CH3|CH4}.

3.2.9.11.2 :TRIGger:NEDGE:SLOPe

Funktion: Legt den Flankentyp des N-ten Flankentriggers fest.

Format: :TRIGger:NEDGE:SLOPe <slope>

:TRIGger:NEDGE:SLOPe?

Dabei ist <slope> vom diskreten Typ {RISE|FALL}.

3.2.9.11.3 :TRIGger:NEDGE:IDLE

Funktion: Legt die Wartezeit fest, bevor die Flankenanzählung im N-ten Flankentrigger beginnt.

Format: :TRIGger:NEDGE:IDLE <time>

:TRIGger:NEDGE:IDLE?

Dabei ist <time> ein reeller Wert im Bereich von 8 ns bis 10 s.

3.2.9.11.4 :TRIGger:NEDGE:EDGE

Funktion: Legt den Wert von N fest, der durch die N-te Flanke ausgelöst wird.

Format: :TRIGger:NEDGE:EDGE <Zahl>

:TRIGger:NEDGE:EDGE?

Dabei ist <Zahl> ein reeller Wert zwischen 1 und 65535.

3.2.9.11.5 :TRIGger:NEDGE:LEVel

Funktion: Legt den Auslösepegel fest, wenn bei der N-ten Flanke ausgelöst wird

Format: :TRIGger:NEDGE:LEVel <Pegel>

:TRIGger:NEDGE:LEVel?

Dabei ist <level> vom Typ „real“.

Video

3.2.9.12.1 :TRIGger:VIDeo:SOURce

Funktion: Legt die Triggerquelle für den Video-Trigger fest.

Format: :TRIGger:VIDeo:SOURce <source>

:TRIGger:VIDeo:SOURce?

Dabei ist <source> ein diskreter Typ mit den Werten {CH1|CH2|CH3|CH4}.

3.2.9.12.2 :TRIGger:VIDeo:POLarity

Funktion: Legt die Polarität des Video-Triggers fest

Format: :TRIGger:VIDeo:POLarity <Polarität>

:TRIGger:VIDeo:POLarity?

Darin ist <polarity> ein diskreter Typ vom Typ { POSItive|NEGAtive }.

3.2.9.12.3 :TRIGger:VIDeo:STANdard

Funktion: Wählt den Videostandard für die Videoauslösung aus.

Format: :TRIGger:VIDeo:STANdard <standard>

:TRIGger:VIDeo:STANdard?

Dazu gehören <standard>, diskreter Typ, {PAL|SECAm|NTSC|720P|1080I|1080P}.

3.2.9.12.4 :TRIGger:VIDeo:MODE

Funktion: Wählen Sie den Synchronisationstyp für den Video-Trigger aus, wenn der Trigger-Standard PAL, SECAm, NTSC oder 1080I ist.

Format: :TRIGger:VIDeo:MODE <mode>

:TRIGger:VIDeo:MODE?

Dabei ist <mode> ein diskreter Typ:

{ODDField|EVENfield|ALLField|ALLLine|LINE}.

(Der gelb markierte ausführbare Parameter ist EVENfield|ALLField|ALLLine)

3.2.9.12.5 :TRIGger:VIDEO:FREQuence

Funktion: Wählt die Signalfrequenz des Video-Triggers aus, wenn der Trigger-Standard 1080P ist.

Format: :TRIGger:VIDeo:FREQuence <Frequenz>

:TRIGger:VIDeo:FR EQuence?

Dabei ist <Frequenz> diskret, {60 Hz|50 Hz|30 Hz|25 Hz|24 Hz}.

3.2.9.12.6 :TRIGger:VIDeo:LINE

Funktion: Wählen Sie die angegebene Triggerzeile für die Auslösung aus.

Format: :TRIGger:VIDeo:LINE <Zeile>

:TRIGger:VIDeo:LINE?

Dabei ist <line> vom Typ „real“ und nimmt Werte von 1 bis n an; je nach Videotyp ist der Maximalwert von n unterschiedlich.

3 folgenma!

Vor der Ausführung dieses Befehls ist Folgendes zu beachten

- A、 Öffnen Sie den Kanal von S1 oder S2
- B、 Nehmen Sie die Kanaleinstellungen für S1 oder S2 vor.
- C、 Der Triggertyp ist S1 oder S2
- D. Die folgenden Befehle sollten mit dem eingestellten Typ S1 oder S2 abgeglichen werden; zum Beispiel:

Öffnen Sie S1 und S2,

S1 auf CAN und S2 auf LIN

einstellen Den Triggertyp auf S1

einstellen

Dann sollte der folgende Trigger auf den Typ S1 CAN eingestellt werden

3.2.9. 13.1: TRIGger:UART

WRITE

Funktion: Legen Sie die Triggerbedingungen für den UART-Trigger fest.

Format: :TRIGger:UART:TYPE < s >, <Typ>

:TRIGger:UART:TYPE? < s >

Dabei ist <s> ein diskreter Typ, { S1|S2 }; <type> ist ein diskreter Typ,
{START|STOP|DATA|0:DATA|1:DATA|X:DATA| PARity}.

Wenn die Buswortlänge in den Bus-Einstellungen auf 9 Bit eingestellt ist, kann der
Triggertyp „DATA“ nicht eingestellt werden;

Wenn die Buswortlänge in den Bus-Einstellungen auf 5 Bit, 6 Bit, 7 Bit oder 8 Bit
eingestellt ist, können 0:DATA, 1:DATA oder X:DATA als Triggertyp nicht eingestellt
werden.

Rückgabeformat: Die Abfrage gibt „START“, „STOP“, „DATA“, „0:DATA“, „1:DATA“, „X:DATA“,
„PARity }“ zurück.

Beispiel:

Der folgende Befehl legt den START-Trigger des UART-Kanals S1 fest.

:TRIGger:UART:TYPE S1, START

Die folgende Abfrage gibt „START“ zurück.

:TRIGger:UART:TYPE? S1

RELAtion

Funktion: Wenn die UART-Bus-Triggerbedingung als DATA, 0:DATA, 1:DATA oder X:DATA ausgewählt ist, legen Sie die UART-Bus-Triggerbeziehung fest.

Format: :TRIGger:UART:RELAtion < s >, <RELATION>

:TRIGger:UART:RELAtion? < s >

Dabei ist <s> ein diskreter Typ, { S1|S2 }; <RELATION> ist ein diskreter Typ, { GREAt|LESS|EQUAL| UNEQual }.

GREAt: Die Eingangsdaten des Oszilloskops sind größer als die angegebenen Triggerdaten;

LESS: Die Eingangsdaten des Oszilloskops sind kleiner als die angegebenen Triggerdaten;

EQUAL: Die Eingangsdaten des Oszilloskops sind gleich den angegebenen Triggerdaten;

UNEQual: Die Eingangsdaten des Oszilloskops sind nicht gleich den angegebenen Triggerdaten;

Rückgabeformat: Die Abfrage gibt „GREAt“, „LESS“, „EQUAL“ und „UNEQual“ zurück.

Beispiel:

Der folgende Befehl legt die Triggerbeziehung des S1-Kanals UART auf „GREAt“ fest.

:TRIGger:UART:RELAtion S1, GREAt

Die folgende Abfrage gibt „GREAt “ zurück.

:TRIGger:UART:RELAtion? S1

UART:DATA

Funktion: Wenn die UART-Bus-Triggerbedingung als „DATA“, „0:DATA“, „1:DATA“ oder „X:DATA“ ausgewählt ist, werden die UART-Bus-Triggerdaten festgelegt.

Format: :TRIGger:UART:DATA <s>, <data>

:TRIGger:UART:DATA?

Dabei ist <s> ein diskreter Typ, { S1|S2 }; <data> ist ein ganzzahliger Typ, hexadezimal, 0 bis FF.

Rückgabeformat: Die Abfrage gibt hexadezimale Werte von 0 bis FF zurück.

Beispiel:

Der folgende Befehl setzt den DATA-Wert des UART-Kanals S1 auf AA.

:TRIGger:UART:DATA S1, AA

Die folgende Abfrage gibt „AA“ zurück.

:TRIGger:UART:DATA? S1



:TRIGger:LIN



Funktion: Legen Sie die Triggerbedingung für den LIN-Trigger fest.

Format: :TRIGger:LIN:TYPE < s >, <type>

:TRIGger:LIN:TYPE?

Darunter: <s>, diskreter Typ, {S1|S2}; <type>, diskreter Typ, { SRISe|FID| IDATa }.

SRISe, synchrone steigende Flanke; FID, Frame-ID; IDATa, Frame-ID und Daten.

Rückgabeformat: Die Abfrage gibt „SRISe“, „FID“ und „IDATa“ zurück.

Beispiel:

Der folgende Befehl legt den Kanal S1 fest.

:TRIGger:LIN:TYPE S1, SRISe

Die folgende Abfrage gibt „SRISe“ zurück.

:TRIGger:LIN:TYPE? S1



Funktion: Wenn die LIN-Bus-Triggerbedingung FID oder IDATa ist, wird der Trigger-ID-Wert des LIN-Triggers festgelegt.

Format: :TRIGger:LIN:ID <s> , <data>

:TRIGger:LIN:ID?

Dabei ist <s> vom Typ „diskret“, {S1|S2}; <data> vom Typ „Ganzzahl“, hexadezimal, 0 bis 3F.

Rückgabeformat: Die Abfrage gibt hexadezimale Werte von 0 bis 3F zurück.

Beispiel:

Der folgende Befehl setzt den DATA-Wert des Kanals S1 LIN auf 0A.

:TRIGger:LIN:ID S1, 0A

Die folgende Abfrage gibt „0A“ zurück.

:TRIGger:LIN:ID? S1



Funktion: Wenn die Triggerbedingung des LIN-Busses IDATa lautet, werden die Triggerdaten des LIN-Triggers gesetzt.

Format: :TRIGger:LIN:DATA <s>, <data>

:TRIGger:LIN:DATA?

Dabei ist <s> vom Typ „diskret“, {S1|S2}; <data> ist vom Typ „Ganzzahl“, hexadezimal, 0 bis FFFF,FFFF,FFFF,FFFF.

Rückgabeformat: Die Abfrage gibt einen hexadezimalen Wert von 0 bis FFFF,FFFF,FFFF,FFFF zurück.

Beispiel:

Der folgende Befehl setzt den DATA-Wert des S1-Kanals LIN auf 0A.

```
:TRIGger:LIN:DATA S1, 0A
```

Die folgende Abfrage gibt „0A“ zurück.

```
:TRIGger:LIN:DATA? S1
```

35 :TRIGGER:CAN

Syntax

Funktion: Festlegen der Auslösebedingung für den CAN-Trigger

Format: :TRIGger:CAN:TYPE < s >, <type>

```
:TRIGger:CAN:TYPE?
```

Dabei ist <s> ein diskreter Typ, {S1|S2}; <type> ist ein diskreter Typ, { FSTArt|RFID|DFID|RDID|IDATa|WRFR|AERRor|ACKError|OVERload }.

FSTArt: Rahmenanfang; RFID: Remote-Rahmen-ID; DFID: Datenrahmen-ID; RDID: Remote-Rahmen-/Datenrahmen-ID; IDATa: Datenrahmen-ID und Daten; WRFR: Fehlerrahmen; AERRor: alle Fehler; ACKError: Bestätigungsfehler; OVERload: Überlastungsrahmen.



Funktion: Wenn die Triggerbedingung des CAN-Triggers RFID, DFID, IDATa oder RDID ist, wird der Trigger-ID-Wert des CAN-Triggers festgelegt.

Format: :TRIGger:CAN:ID <s>, <data>

:TRIGger:CAN:ID?

Dabei gilt: <s>, diskreter Typ, {S1|S2}; <data>, Ganzzahltyp, hexadezimal, 0 bis 7 FFF,FFFF.



Funktion: Wenn die Auslösebedingung des CAN-Triggers „IDATa“ ist, legen Sie den DLC-Wert des CAN-Triggers fest.

Format: :TRIGger:CAN:DLC <s>, <data>

:TRIGger:CAN:DLC?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <data> ein ganzzahliger Typ, 0 bis 8, 12, 16, 20, 26, 32, 48, 64.



Funktion: Wenn die Triggerbedingung des CAN-Triggers IDATa ist, wird der Triggerdatenwert des CAN-Triggers festgelegt.

Format: :TRIGger:CAN:DATA <s>, <data>

:TRIGger:CAN:DATA?

Dabei ist <s> vom Typ „diskret“, {S1|S2}; <data> vom Typ „Ganzzahl“, hexadezimal, wobei die Anzahl der Datenstellen durch DLC festgelegt wird.

:TRIGger:SPI

DATA

Funktion: Legt den Datenwert unter dem SPI-Trigger fest.

Format: :TRIGger:SPI:DATA <s>,<data>

:TRIGger:SPI:DATA?

Dabei ist <s> vom Typ „diskret“, {S1|S2}; <data> ist vom Typ „Ganzzahl“, binär.

TYPE

Funktion: Legt den Datenwert unter dem SPI-Trigger fest.

Format: :TRIGger:SPI:TYPE <s>,< dtype >

:TRIGger:SPI:TYPE?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <type> ist ein diskreter Typ,

{ CS|DATA|X:DATA }

:TRIGger:IIC

TYPE

Funktion: Legt den Triggertyp des IIC-Triggers fest.

Format: :TRIGger:IIC:TYPE <s>,<Typ>

:TRIGger:IIC:TYPE?

Dabei gilt: <s>, diskreter Typ, {S1|S2}; <type>, diskreter Typ, {START|STOP|ACKLost|NACKaddress|REStart|RDATa|FRAM1|FRAM2 |WRITe10 }.

START, Startbedingung; STOP, Stoppbedingung; ACKLost, Bestätigung verloren; NACKaddress, keine Bestätigung im Adressfeld; REStart, Neustart; RDATa, EEPROM-Datenauslesung; FRAM1, Frame-Typ 1; FRAM2, Frame-Typ 2; WRITe10, 10-Bit-Schreib-Frame.

ADDRess

Funktion: Wenn die IIC-Triggerbedingung NACKaddress, FRAM1, FRAM2 oder WRITe10 ist, wird die Triggeradresse des IIC-Bus-Triggers festgelegt.

Format: :TRIGger:IIC:ADDRess <s>, <data>

:TRIGger:IIC:ADDRess?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <data> ist ein ganzzahliger Typ, hexadezimal, 0 bis 7F (7 Bit) oder 0 bis 3FF (10 Bit).

ELation

Funktion: Wenn die IIC-Triggerbedingung RDATa ist, wird die Triggerbeziehung des IIC-Bus-Triggers festgelegt.

Format: :TRIGger:IIC:RELation <s>, <relation>

:TRIGger:IIC:RELation

Dabei ist <s> ein diskreter Typ, {S1|S2}; <RELATION> ein diskreter Typ,
{ GREAT|LESS|EQUAL| UNEQual }.

GREAT: Die Eingangsdaten des Oszilloskops sind größer als die angegebenen Triggerdaten;

LESS: Die Eingangsdaten des Oszilloskops sind kleiner als die angegebenen Triggerdaten;

EQUAL: Die Eingangsdaten des Oszilloskops sind gleich den angegebenen Triggerdaten;

UNEQual: Die Eingangsdaten des Oszilloskops sind nicht gleich den angegebenen Triggerdaten;

DATA

Funktion: Wenn die IIC-Triggerbedingung RDATA, FRAM1, FRAM2 oder WRITe 10 ist, werden die Triggerdaten für den IIC-Bus-Trigger festgelegt.

Format: :TRIGger:IIC:DATA <s>, <data>

:TRIGger:IIC:DATA?

Dabei ist <s> vom Typ „diskret“, {S1|S2}; <data> ist vom Typ „Ganzzahl“, hexadezimal, 0–FF.

DATA2

Funktion: Wenn die IIC-Triggerbedingung FRAM2 ist, legen Sie die durch den IIC-Bus ausgelösten Triggerdaten fest.

Format: :TRIGger:IIC:DATA2 < s >, <data>

:TRIGger:IIC:DATA2?

Dabei ist <s> vom Typ „diskret“, {S1|S2}; <data> ist vom Typ „Ganzzahl“, hexadezimal, 0–FF.

3 :TRIGger:1553B

3 TYPE

Funktion: Legt die Triggerbedingung für den 1553B-Bus-Trigger fest.

Format: :TRIGger:1553B:TYPE < s >, <type>

:TRIGger:1553B:TYPE?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <type> ist ein diskreter Typ, { CSSYnc|DWSYnc|CSWOrd|DWORD|RTADdress|OPERror|MERRor| AERRor }.

CSSYnc, Synchronisations-Header für Befehls-/Statusworte; DWSYnc, Synchronisations-Header für Datenworte; CSWOrd, Befehls-/Statuswort; DWORD, Datenwort; RTADdress, Adresse des entfernten Terminals; OPERror, Fehler bei ungerader Parität; MERRor, Manchester-Code-Fehler; AERRor, alle Fehler.

3 CSWOrd

Funktion: Wenn die 1553B-Triggerbedingung CSWOrd lautet, wird der Wert des Befehls-/Statusworts gesetzt, der durch den 1553B-Bus ausgelöst wurde.

Format: :TRIGger:1553B:CSWOrd <s>, <data>

:TRIGger:1553B:CSWOrd?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <data> ein Ganzzahltyp, 0 bis FFFF.

1553B

Funktion: Wenn die 1553B-Triggerbedingung „DWORD“ lautet, wird der Wert der Triggerdaten des 1553B-Bus-Triggers festgelegt.

Format: :TRIGger:1553B:DWORD <s>, <data>

:TRIGger:1553B:DWORD?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <data> ein ganzzahliger Typ, 0 bis FFFF.

1553B

Funktion: Wenn die 1553B-Bus-Triggerbedingung RTADdress lautet, wird die durch den 1553B-Bus ausgelöste Fernterminaladresse festgelegt.

Format: :TRIGger:1553B:RTADdress < s >, <address>

:TRIGger:1553B:RTADdress?

Dabei gilt: <s>, diskreter Typ, {S1|S2}; <address>, Ganzzahltyp, 0 bis FF.

429 :TRIGger:429

429

Funktion: Legt die Triggerbedingung für den 429-Bus-Trigger fest.

Format: :TRIGger:429:TYPE < s >, <Typ>

:TRIGger:429:TYPE?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <type> ist ein diskreter Typ,
 { WBEGin|WEND |
 LABEL|SDI|DATA|SSM|LSDI|LDATa|LSSM|WERRor|WINTerval|VERRor|AERRor|ALL0|AL L1}.

WBEGin, Wortanfang; WEND, Wortende; LSDI, LABEL+SDI; LDATa, LABEL+DATA;
 LSSM, LABEL+SSM; WERRor, Wortfehler; WINTerval, Wortlückenfehler; VERRor,
 Prüfsummenfehler; AERRor, alle Fehler; ALL0, alle Bits auf 0; ALL1, alle Bits auf 1.

Nach der Auswahl von LSDI, LDATa und LSSM müssen Sie die zugehörigen Parameter festlegen. Verwenden Sie dazu LABEL, SDI, DATA und SSM, um die Parameter einzustellen;

Beispiel:

:TRIGger:429:TYPE S1, LSDI

:TRIGger:429:LABEL S1,377

:TRIGger:429:SDI S1,11



Funktion: Wenn die Triggerbedingung des 429-Busses „WORD-Start“ lautet, wird der Wert des Triggerworts des 429-Bus-Triggers gesetzt.

Format: :TRIGger:429:WBEG in < s > >

:TRIGger:429:WBEG in?

Dabei ist <s> diskret, {S1|S2}.

SDI

Funktion: Wenn die Triggerbedingung des 429-Busses „WORD-Ende“ lautet, legen Sie den Wert des Triggerworts für den 429-Bus-Trigger fest.

Format: :TRIGger:429:WEND < s >

:TRIGger:429:WEND?

Dabei ist <s> diskret, {S1|S2}.

SDI

Funktion: Wenn die Triggerbedingung des 429-Busses „LABEL“, „LSDI“, „LDATa“ oder „LSSM“ lautet, wird der LABEL-Wert des 429-Bus-Triggers festgelegt.

Format: :TRIGger:429:LABEL <s> , <data>

:TRIGger:429:LABEL?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <data> ist ein ganzzahliger Typ, okt, 0 bis 377.

SDI

Funktion: Wenn die 429-Bus-Triggerbedingung SDI oder LSDI ist, legen Sie den SDI-Wert des 429-Bus-Triggers fest.

Format: :TRIGger:429:SDI <s> , <data>

:TRIGger:429:SDI?

Dabei ist <s> vom Typ „diskret“, {S1|S2}; <data> ist vom Typ „Ganzzahl“, binär, 0 0 bis

11.

DATA

Funktion: Wenn die Triggerbedingung des 429-Busses DATA oder LDATa ist, wird der Triggerdatenwert des 429-Bus-Triggers gesetzt.

Format: :TRIGger:429:DATA <s>, <data>

:TRIGger:429:DATA?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <data> ist ein ganzzahliger Typ im Hexadezimalformat im Bereich von 0 bis FFFFFFFF.

SSM

Funktion: Wenn die 429-Bus-Triggerbedingung „SSM“ oder „LSSM“ lautet, legen Sie den Triggerdatenwert des 429-Bus-Triggers fest.

Format: :TRIGger:429:SSM < s >, <data>

:TRIGger:429:SSM?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <data> ein ganzzahliger Typ, binär, 0 bis 11.

3.2.10 Befehlssystem für die Zeitbasis

EXTent

Funktion: Stellt die horizontale Zeitbasis ein.

Format: :TIMebase:EXTent <extent>

:TIMebase:EXTent?

Dabei ist <extent> vom Typ „real“. Einheit: S

Rückgabeformat: Die Abfrage gibt den Offset-Wert in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt die horizontale Zeitbasis auf 2 µs.

Format: :TIMebase:EXTent 2.000000e-6

Die folgende Abfrage gibt „2,000000e-06“ zurück

:TIMebase:EXTent?

MODE

Funktion: Legt den Anzeigemodus für die Bildschirm-Zeitbasis fest. „YT“ oder „XY“.

Format: :TIMebase:MODE <Modus>

:TIMebase:MODE?

Dabei ist <Modus> ein diskreter Wert: „YT“ oder „XY“.

Delay

Funktion: ROLL-Modus ein- oder ausschalten (Zeitbasis über 100 ms).

Format: :TIMebase:ROLL:DISPlay <bool>

:TIMebase:ROLL:DISPlay?

Dabei ist <bool> vom Typ „Boolean“ und kann die Werte {{0|OFF}}{1|ON}} annehmen.

Hinweis: Wenn Sie im ROLL-Modus arbeiten möchten, müssen Sie nach dem Aktivieren dieses Schalters auch die Zeitbasis auf über 100 ms/div einstellen.

POSITION

Funktion: Legt den horizontalen Versatz der Wellenformanzeige fest.

Format: :TIMebase:POSition <position>

:TIMebase:POSition?

Dabei ist < POSition > vom Typ „real“.

Rückgabeformat: Die Abfrage gibt den Versatzwert in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt den horizontalen Offset auf 2 µs.

:TIMebase:POSition 0,000002

Die folgende Abfrage gibt „2,000000e-06“ zurück

:TIMebase:POSition?

~~TIMEbase~~SCALE

Funktion: Festlegen und Abfragen der Zeitbasis des großen Fensters nach dem Öffnen des Zooms.

:TIMEbase:ZOOM:SCALE <Wert>

:TIMEbase:ZOOM:SCALE?

Dabei ist <Wert> ein reeller Wert vom Typ {1e-9~1e3}

3.2.11 Befehlssystem „Speicher“

~~STORAGE~~SAVE

Funktion: Speichert die Wellenform des angegebenen Kanals an der angegebenen Stelle.

Format: :STORage:SAVE <Kanal>,<Speicherort>

:STORage:SAVE <Kanal>

Dabei ist <Kanal> ein diskreter Typ, {CH1|CH2|CH3|CH4|MATH}; <Speicherort> ein diskreter Typ, {LOCAL|UDISK}, Standardwert LOCAL

Hinweis: Bei der segmentierten Speicherung wird der aktuelle Frame gespeichert.

3.2.11.1.1 :STORage:SAVE:SOURce

Format: :STORage:SAVE:SOURce <Kanal>

:STORage:SAVE:SOURce?

Dabei ist <Kanal>, diskreter Typ, {CH1|CH2|CH3|CH4|MATH};

:STORage:SAVE:LOCAtion

Format: :STORage:SAVE:LOCAtion <location>

:STORage:SAVE:LOCAtion?

Davon < LOCAtion >, diskreter Typ, { LOCal|UDISk };

:STORage:SAVE:TYPE

Format: :STORage:SAVE:TYPE <Typ>

:STORage:SAVE:TYPE?

Davon ist <TYPE> ein diskreter Typ, {WAV|BIN|CSV};

:STORage:SAVE:FILEname

Format: :STORage:SAVE:FILEname <Dateiname>

:STORage:SAVE:FILEname?

Dabei gilt: <Dateiname> := ASCII-Zeichenkette in Anführungszeichen

~~:STORage:SAVE:ALLSegments~~

Funktion: Bei segmentierter Speicherung festlegen, dass alle Segmente gespeichert werden

Format: :STORage:SAVE:ALLSegments < bool >

:STORage:SAVE:ALLSegments?

Dabei ist <bool> vom Typ Boolesche Variable, {{0|OFF}}{1|ON}}.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl aktiviert bei segmentierter Speicherung die Speicherung aller Segmente.

```
:STORage:SAVE:ALLSegments ON oder :STORage:SAVE:ALLSegments 1
```

Die folgende Abfrage gibt „1“ zurück.

```
:STORage:SAVE:ALLS elements?
```

Hinweis: Dieser Befehl ist nur gültig, wenn die segmentierte Speicherung aktiviert ist.

Nachdem dieser Befehl aktiviert wurde, kann der Speichertyp :STORage:SAVE:TYPE <type> nur noch „BIN“ auswählen.

:STORage:SAVE:START

Format: :STORage:SAVE:START

Speichern starten.

~~STORage~~MOD

Funktion: Ref. laden

Format: :STORage:LOAD <ref><bool>, <filename>

Davon sind <source> vom Typ „diskret“, {R1| R2| R3| R4 }; < filename> vom Typ „diskret“, der Name der geladenen Datei; <bool> vom Typ „Boolescher Wert“, {{0|OFF}}{1|ON}}.

~~STORAGE~~

Funktion: Einstellungen für Screenshots.

3.2.11.3.1 :STORage:CAPTure:TIME <bool>

Funktionen: Einstellungen und Abfragen, Zeitstempel von Screenshots

Format: :STORage:CAPTure:TIMEstamp <bool>

:STORage:CAPTure:TIMEstamp?

Dabei ist <bool> ein boolescher Wert, {{0|OFF}}{{1|ON}}.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl aktiviert den Zeitstempel für den Screenshot.

:STORage:CAPTure:TIMEstamp ON oder :STORage:CAPTure:TIMEstamp 1

Die folgende Abfrage gibt „1“ zurück.

:STORage:CAPTure:TIMEstamp?

3.2.11.3.2 :STORage:CAPTure:INCOlor <bool>

Funktion: Festlegen und Abfragen, ob der Screenshot invertiert ist

Format: :STORage:CAPTure:INCOlor < bool >

:STORage:CAPTure:INCOlor?

Dabei ist <bool> vom Typ „Boolean“ und kann die Werte {{0|OFF}}{{1|ON}} annehmen.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl aktiviert Zeitstempel für Screenshots.

:STORage:CAPTure:INCOlor ON oder :STORage:CAPTure:INCOlor 1

Die folgende Abfrage gibt „1“ zurück.

:STORage:CAPTure:INCOlor?

3.2.11.3.3 :STORage:CAPTure:STARt

Funktion: Screenshot starten

Format: :STORage:CAPTure:STARt

Beispiel:

Der folgende Befehl startet die Erstellung von Screenshots.

:STORage:CAPTure:STARt

3.2.11.3.4 :STORage:CONSave

Funktion: Speichert die Oszilloskop-Einstellungen.

Legt den Namen der Oszilloskop-Einstellung fest

Format: :STORage:CONSave:FILEname <Dateiname>

wobei <Dateiname>:= ASCII-Zeichenkette in Anführungszeichen

Format: :STORage:CONSave:START

Speichern starten.

3.2.12

Funktion: Ruft die Oszilloskop-Einstellungen mit dem entsprechenden Namen auf.

3.2.12 Subsystem für Buskonfigurationsbefehle

1 BUS

3.2.12.1.1 :BUS <S>:DISPlay

Funktion: Dekodierungskanal umschalten.

Format: :BUS <S>:DISPlay <bool>

:BUS <S>:DISPlay?

Dabei ist <n> vom Typ „diskret“ ({ 1|2|3|4}); <bool> vom Typ „Boolescher Wert“ ({0| OFF}| {1|ON}).

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl schaltet die Anzeige des Dekodierkanals 1 ein.

:BUS1:DIAPlay ON oder :BUS1:DIAPlay 1

Die folgende Abfrage gibt „1“ zurück.

:BUS1:display?

3.2. 12 .1.2 :BUS <S>:TYPE

Funktion: Legt den Bustyp von Bus S1 oder S2 fest.

Format: :BUS <S>:TYPE <Typ>

:BUS <S>:TYPE?

Dabei ist <s> diskret, {S1|S2};

<Typ>, diskreter Typ, {UART|LIN|SPI|CAN|IIC|1553B|429}.

3.2.12.1.3 :BUS <S>:MODE <mode>

Funktion: Legt den Anzeigemodus des Busses fest, einschließlich Grafik- und Textmodus.

Format: :BUS <S>:MODE < Modus >

:BUS <S>:MODE?

Dabei ist < mode > ein diskreter Typ, {GRAP|TXT}.

3.2. 12 .1.4 :BUS <S>:LEVel < Kanal >,<Pegel>

Funktion: Legt den Schwellenwert des Busses fest.

Format: :BUS <S>:LEVel <Kanal>,<Pegel>

:BUS <S>:LEVel? <Kanal>

Dabei ist <channel> vom diskreten Typ {CH1|CH2|CH3|CH4}; <level> vom reellen Typ.

Hinweis: Dieser Befehl muss im Grafikmodus (:BUS<S>:MODE GRAP) nach Abschluss anderer Buskonfigurationen eingestellt werden.

3.2.12.1.5 :BUS <S>:HLEVel <Kanal>,<Pegel>

Funktion: Wenn der Bus über zwei Schwellenwerte verfügt, wird der obere Schwellenwert des Busses festgelegt.

Format: :BUS <S>:HLEVel <Kanal>,<Pegel>

:BUS <S>:HLEVel? <Kanal>

Dabei ist <Kanal> ein diskreter Typ, {CH1|CH2|CH3|CH4}; <Pegel> ist ein reeller Typ.

Hinweis: Dieser Befehl muss im Grafikmodus (:BUS<S>:MODE GRAP) nach Abschluss anderer Buskonfigurationen eingestellt werden.

3.2.12.1.6 :BUS <S>:LLEVel <Kanal>,<Pegel>

Funktion: Wenn der Bus über zwei Schwellenwerte verfügt, wird der untere Schwellenwert des Busses festgelegt.

Format: :BUS <S>:LLEVel <Kanal>,<Pegel>

:BUS <S>:LLEVel? <Kanal>

Dabei ist <Kanal> ein diskreter Typ ({CH1|CH2|CH3|CH4}) und <Pegel> ein reeller Typ.

Hinweis: Dieser Befehl muss im Grafikmodus (:BUS<S>:MODE GRAP) nach Abschluss anderer Buskonfigurationen eingestellt werden.

2 ~~BUS~~ ~~UART~~

3.2.12.2.1 :BUS <s>:UART:RX

Funktion: Legt die RX-Kanalquelle der UART-Buskonfiguration fest.

Format: :BUS <s>:UART:RX <Kanal>

:BUS <s>:UART:RX?

Dabei ist <s> diskret, {S1|S2}, und <Kanal> ist diskret, {CH1|CH2|CH3|CH4}.

3.2.12.2.2 :BUS <s>:UART:IDLElvl

Funktion: Legt den Ruhepegel-Zustand der UART-Buskonfiguration fest.

Format: :BUS <s>:UART:IDLElvl <Zustand>

:BUS <s>:UART:IDLElvl?

Dabei ist <Zustand> ein diskreter Wert vom Typ { high|low }.

3.2.12.2.3 :BUS <s>:UART:BAUDrate

Funktion: Wählt die Baudrate der UART-Buskonfiguration aus. Einheit: b/s

Format: :BUS <s>:UART:BAUDrate < Baudrate >

:BUS <s>:UART:BAUDrate?

Dabei ist < Baudrate > ein diskreter Typ,
{1200|2400|4800|9600|19200|38400|43000|56000|57600|115200} .

3.2.12.2.4 :BUS <s>:UART:CHECK

Funktion: Wählt die Überprüfungsmethode für die UART-Buskonfiguration aus.

Format: :BUS <s>:UART:CHECK <check>

:BUS <s>:UART:CHECK?

Dabei ist <check> ein diskreter Typ, {NONE|ODD|EVEN}.

3.2.12.2.5 :BUS <s>:UART:USERbaud

Funktion: Wählt die benutzerdefinierte Baudrate bei der Konfiguration des UART-Busses aus.

Einheit: b/s

Format: :BUS <s>:UART:USERbaud < Baudrate >

:BUS <s>:UART:USERbaud?

Dabei ist <baudrate> eine ganze Zahl im Bereich von 1200 bis 8000000.

3.2.12.2.6 :BUS <s>:UART:WIDTH

Funktion: Auswahl der Datenbreite bei der Konfiguration des UART-Busses.

Format: :BUS <s>:UART:WIDTH <Breite>

:BUS <s>:UART:WIDTH?

Dabei ist <width> ein diskretes Wert aus {5|6|7|8|9}.

3.2.12.2.7 :BUS <s>:UART:DISPlay

Funktion: Wählt den Datenanzeigemodus aus, wenn der UART-Bus konfiguriert ist.

Format: :BUS <s>:UART:DISPlay < Anzeige >

:BUS <s>:UART:DISPlay?

Dabei ist <Anzeige> diskret, {HEX|BIN|ASCII}.

~~2~~BUS <S>:LIN

3.2.12.3.1 :BUS <S>:LIN:CHANnel

Funktion: Wählt die Kanalquelle für die LIN-Bus-Konfiguration aus.

Format: :BUS <S>:LIN:CHANnel <Kanal>

:BUS <S>:LIN:CHANnel?

Wobei <Kanal> ein diskreter Typ ist, {CH1|CH2|CH3|CH4}.

3.2.12.3.2 :BUS <S>:LIN:IDLElvl

Funktion: Legt den Ruhezustands-Level der LIN-Bus-Konfiguration fest.

Format: :BUS <S>:LIN:IDLElvl <Zustand>

:BUS <S>:LIN:IDLElvl?

Dabei ist <Zustand> ein diskreter Typ, { high|low }.

3.2.12.3.3 :BUS <S>:LIN:BAUDrate

Funktion: Wählt die Baudrate der LIN-Bus-Konfiguration aus. Einheit: b/s.

Format: :BUS <S>:LIN:BAUDrate < Baudrate >

:BUS <S>:LIN:BAUDrate?

Dabei ist <baudrate> ein diskretes Wert, {2400|9600|19200}.

3.2.12.3.4 :BUS <S>:LIN:USERbaud

Funktion: Wählt die benutzerdefinierte Baudrate bei der Konfiguration des LIN-Busses aus.

Einheit: b/s

Format: :BUS <S>:LIN:USERbaud <baudrate>

:BUS <S>:LIN:USERbaud?

Dabei ist <baudrate> eine ganze Zahl im Bereich von 2400 bis 625000.

~~BUS~~SPI

3.2.12.4.1 :BUS <S>:SPI:CLK

Funktion: Wählt die Taktquelle für die SPI-Bus-Konfiguration aus.

Format: :BUS <S>:SPI:CLK <Kanal>

:BUS <S>:SPI:CLK?

Dabei ist <Kanal> ein diskreter Typ, {CH1|CH2|CH3|CH4}.

3.2.12.4.2 :BUS <S>:SPI:DATA

Funktion: Wählt die Datenquelle für die SPI-Bus-Konfiguration aus.

Format: :BUS <S>:SPI:DATA <Kanal>

:BUS <S>:SPI:DATA?

Dabei ist <channel> ein diskreter Typ, {CH1|CH2|CH3|CH4}.

3.2.12.4.3 :BUS <S>:SPI:WIDTh

Funktion: Wählen Sie die Datenbreite bei der Konfiguration des SPI-Busses aus.

Format: :BUS <S>:SPI:WIDTh <Breite>

:BUS <S>:SPI:WIDTh?

Dabei ist <width> ein diskreter Wert und kann {4|8|16|24|32} annehmen.

3.2.12.4.4 :BUS <S>:SPI:IDLElvl

Funktion: Wählt den Ruhepegelzustand der SPI-Bus-Konfiguration aus.

Format: :BUS <S>:SPI:IDLElvl <Zustand>

:BUS <S>:SPI:IDLElvl?

Dabei ist <Zustand> diskret und kann { high|low } annehmen.

3.2.12.4.5 :BUS <S>:SPI:SLOPe

Funktion: Wählt den Taktflankentyp für die SPI-Bus-Konfiguration aus.

Format: :BUS <S>:SPI:SLOPe <slope>

:BUS <S>:SPI:SLOPe?

Dabei ist <slope> diskret, {RISE|FALL}.

3.2.12.4.6 :BUS <S>:SPI:CS

Funktion: Setzen und Abfragen der CS-Freigabe im SPI

Format: :BUS <S>:SPI:CS <bool>

:BUS <S>:SPI:CS?

Dabei ist <bool> vom Typ „Boolean“ und kann die Werte {{0|OFF}}{1|ON}} annehmen.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl aktiviert CS.

:BUS <S>:SPI:CS <bool> ON oder :BUS<S>:SPI:CS <bool> 1

Die folgende Abfrage gibt „1“ zurück.

:BUS <S>:SPI:CS?

3.2.12.4.7 :BUS <S>:SPI:CS:SOURce

Funktion: Quelle von CS im SPI setzen und abfragen

Format: :BUS <S>:SPI:CS:SOURce <Kanal>

:BUS <S>:SPI:CS:SOURce?

Dabei ist <Kanal> ein diskreter Typ, {CH1|CH2|CH3|CH4}.

Rückgabeformat: Die Abfrage gibt „CH1“, „CH2“, „CH3“ und „CH4“ zurück.

Beispiel:

Der folgende Befehl legt die Quelle von CS fest.

:BUS <S>:SPI:CS:SOURce CH1

Die folgende Abfrage gibt „CH1“ zurück.

:BUS <S>:SPI:CS:SOURce?

3.2.12.4.8 :BUS <S>:SPI:CS:IDLEm

Funktion: Wählt den Leerlaufpegelzustand von CS in der SPI-Buskonfiguration aus.

Format: :BUS <S>:SPI:CS:IDLEm <Zustand>

:BUS <S>:SPI:CS:IDLEm?

Dabei ist <Zustand> ein diskreter Wert, {HIGH|LOW}.

Ausgabeformat: Die Abfrage gibt „HIGH“ und „LOW“ zurück.

Beispiel:

Der folgende Befehl legt die Quelle von CS fest.

:BUS <S>:SPI:CS:IDLEm HIGH

Die folgende Abfrage gibt „HIGH“ zurück.

:BUS <S>:SPI:CS:IDLElvl?

~~3.2.12.5~~

3.2.12.5.1 :BUS <S>:CAN:CHANnel

Funktion: Wählen Sie die Kanalquelle für die CAN-Bus-Konfiguration aus.

Format: :BUS <S>:CAN:CHANnel <Kanal>

:BUS <S>:CAN:CHANnel?

Dabei ist <Kanal> ein diskreter Typ, {CH1|CH2|CH3|CH4}.

3.2.12.5.2 :BUS <S>:CAN:SIGNAL

Funktion: Legt den Ruhepegel der CAN-Bus-Konfiguration fest.

Format: :BUS <S>:CAN:SIGNAL <signal>

:BUS <S>:CAN:SIGNAL?

Dabei ist <signal> ein diskreter Typ, {CAN_H|CAN_L|H_L|L_H|RX|TX}.

3.2.12.5.3 :BUS <S>:CAN:BAUDRate

Funktion: Wählt die Baudrate der CAN-Bus-Konfiguration aus. Einheit: b/s.

Format: :BUS <S>:CAN:BAUDRate < Baudrate >

:BUS <S>:CAN:BAUDRate?

Dabei ist < baudrate > ein diskreter Wert,
{50000|100000|250000|500000|800000|1000000} ist.

3.2.12.5.4 :BUS <S>:CAN:USERbaud

Funktion: Auswahl einer benutzerdefinierten Baudrate während der CAN-Bus-Konfiguration.
Einheit: b/s

Format: :BUS <S>:CAN:USERbaud < Baudrate >

:BUS <S>:CAN:USERbaud?

Dabei ist < Baudrate > eine ganze Zahl zwischen 10.000 und 1.000.000.

3.2.12.5.5 :BUS <S>:CAN:SAMPlpoint

Funktion: Auswahl des Abtastpunkts bei der CAN-Bus-Konfiguration und des
Abtastpunkts des Arbitrierungsfelds bei CAN FD, Einheit %

Format: :BUS <S>:CAN:SAMPlpoint < Prozent >

:BUS <S>:CAN:SAMPlpoint?

Dabei ist <Prozent> eine ganze Zahl von 1 bis 99.

3.2.12.5.6 :BUS <S>:CAN:FDBAudrate

Funktion: Auswahl der Baudrate des Datenbits für die CAN-FD-Buskonfiguration.
Einheit: b/s.

Format: :BUS <S>:CAN:FDBAudrate < Baudrate >

:BUS <S>:CAN:FDBAudrate?

Dabei ist < baudrate > ein diskretes Wert, {NONE|2M|5M}

3.2.12.5.7 :BUS <S>:CAN:FDUSERbaud

Funktion: Wählen Sie bei der Konfiguration des CAN-FD-Busses die benutzerdefinierte Baudrate des Datenbits aus. Einheit: b/s

Format: :BUS <S>:CAN:FDUSERbaud < Baudrate >

:BUS <S>:CAN:FDUSERbaud?

Dabei ist <baudrate> eine ganze Zahl im Bereich von 1.000.000 bis 12.000.000.

3.2.12.5.8 :BUS <S>:CAN:FDSA mappoint

Funktion: Auswahl des Abtastpunkts des CANF-Datenfelds, Einheit: %

Format: :BUS <S>:CAN:FDSA mappoint < Prozent >

:BUS <S>:CAN:FDSA mappoint?

Dabei ist < Prozent > eine ganze Zahl im Bereich von 1 bis 99.

3.2.12.5.9 :BUS <S>:CAN:ISO

Funktion: Legt den Standard für die CAN-Bus-Konfiguration fest: ISO oder Nicht-ISO.

Format: :BUS <S>:CAN:ISO < iso >

:BUS <S>:CAN:ISO?

Dabei ist <iso> ein diskreter Typ, {ISO|NON}.

3.2.12.6.1 :BUS <S>:IIC:SDA

Funktion: Legt die Quelle des seriellen Datenkanals für die IIC-Buskonfiguration fest.

Format: :BUS <S>:IIC:SDA <Kanal>

:BUS <S>:IIC:SDA?

Dabei ist <Kanal> ein diskreter Typ, {CH1|CH2|CH3|CH4}.

3.2.12.6.2 :BUS <S>:IIC:SCL

Funktion: Legt die Kanalquelle des vom IIC-Bus konfigurierten seriellen Takts fest.

Format: :BUS <S>:IIC:SCL <Kanal>

:BUS <S>:IIC:SCL?

Dabei ist <channel> ein diskreter Typ, {CH1|CH2|CH3|CH4}.

~~1553B~~ 1553B

3.2.12.7.1 :BUS <S>:1553B:SOURce

Funktion: Legt die Kanalquelle der 1553B-Buskonfiguration fest.

Format: :BUS <S>:1553B:SOURce <Kanal>

:BUS <S>:1553B:SOURce?

Dabei ist <Kanal> ein diskreter Typ, {CH1|CH2|CH3|CH4}.

3.2.12.7.2 :BUS <S>:1553B:DISPlay

Funktion: Legt den Anzeigemodus der 1553B-Buskonfiguration fest.

Format: :BUS <S>1553B:DISPlay <Anzeige>

:BUS <S>:1553B:DISPlay?

Dabei ist <Anzeige> ein diskreter Typ, { BINArY| HEX }.



3.2.12.8.1 :BUS <S>:429:SOURce

Funktion: Legt die Kanalquelle der 429-Bus-Konfiguration fest.

Format: :BUS <S>:429:SOURce <Kanal>

:BUS <S>:429:SOURce?

Dabei ist <Kanal> ein diskreter Typ, {CH1|CH2|CH3|CH4}.

3.2.12.8.2 :BUS <S>:429:FORMat

Funktion: Festlegen des Formats der 429-Bus-Konfiguration

Format: :BUS <S>:429:FORMat <format>

:BUS <S>:429:FORMat?

Dabei ist <format> ein diskreter Typ, { LDAT | LDSS | LSDS }.

LSDS: LABEL+SDI+DATA+SSM; LDSS: LABEL+DATA+SSM; LDAT: LABEL+DATA.

3.2.12.8.3 :BUS <S>:429:DISPlay

Funktion: Legt den Anzeigemodus der 429-Bus-Konfiguration fest.

Format: :BUS <S>:429:DISPlay <Anzeige>

:BUS <S>:429:DISPlay?

Dabei ist <Anzeige> ein diskreter Typ, { BINARY| HEX }.

3.2.12.8.4 :BUS <S>:429:BANDRate

Funktion: Legt die Baudrate der 429-Bus-Konfiguration fest.

Format: :BUS <S>:429:BANDRate < Bandrate >

:BUS <S>:429:BANDRate?

Dabei ist < bandrate > vom Typ „diskret“, {12500|100000}.

3.2.13 Referenz-Wellenform-Befehlssystem

REFereNce:DISPlay

Funktion: Schaltet die REF-Funktion ein oder aus.

Format: :REFereNce:DISPlay <bool>

:REFereNce:DISPlay?

Dabei ist <bool> ein boolescher Typ, {{0|OFF}}{1|ON}}.

Rückgabeformat: Die Abfrage gibt „1“ oder „0“ zurück.

Beispiel:

Der folgende Befehl aktiviert die REF-Funktion.

:REFErence:DISPlay ON

Der folgende Befehl gibt 1 zurück.

:REFErence:DISPlay?

~~3.1.1~~ :REFErence:ENABle <n>

Funktion: Den angegebenen Referenzkanal aktivieren oder deaktivieren.

Format: :REFErence <n>:ENABle <bool>

:REFErence <n>:ENABle?

Dabei ist <n> vom Typ „diskret“ ({ 1|2|3|4}); <bool> vom Typ „Boolescher Wert“ ({{0| OFF}| {1|ON}}).

Rückgabeformat: Die Abfrage gibt „1“ oder „0“ zurück.

Beispiel:

Der folgende Befehl schaltet R1 ein.

:REFErence1:ENABle ON

Der folgende Befehl gibt 1 zurück.

:REFErence1:ENABle?

~~FFRefHScale~~

Funktion: Legt den horizontalen Maßstab des Referenzkanals fest.

Format: :REFErence <n>:HSCale <scale>

:REFErence <n>:HSCale?

Dabei ist <n> vom Typ „diskret“ und kann die Werte { 1|2|3|4} annehmen; <scale> ist vom Typ „real“ und kann Werte im Bereich von 1 ns bis 1 ks oder von 1 mHz bis 1 GHz annehmen.

Rückgabeformat: Die Abfrage gibt den horizontalen Skalierungsfaktor in wissenschaftlicher Notation zurück.

~~FFRefVScale~~

Funktion: Legt den vertikalen Maßstab des Referenzkanals fest.

Format: :REFErence <n>:VSCale <scale>

:REFErence <n>:VSCale?

Dabei ist <n> vom Typ „diskret“ und kann die Werte { 1|2|3|4} annehmen; <scale> ist vom Typ „real“ und kann Werte zwischen 5 mV und 5 GV annehmen.

Rückgabeformat: Die Abfrage gibt den vertikalen Maßstab in wissenschaftlicher Notation zurück. (Wenn die aufgerufene Referenzwellenform „fft“ ist, ändert sich der horizontale Maßstab des Kanals nicht.)

Beispiel:

Der folgende Befehl setzt den vertikalen Maßstab von Referenzkanal 1 auf 2 V.

:REFerence1:VSCale 2

Der folgende Befehl gibt 2,000000e+00 zurück.

:REfERENCE1:VSCale?



Funktion: Wählt den aktuellen Referenzkanal aus.

Format: :CURRent:REfERENCE <n>

Dabei ist <n> ein diskreter Typ, { R1|R2|R3|R4} .



Funktion: Legt den vertikalen Versatz der Wellenformanzeige des angegebenen Referenzkanals fest.

Format: :REfERENCE <n>:VPOSition <pos>

:REfERENCE <n>:VPOSition?

Dabei ist <n> vom diskreten Typ { 1|2|3|4} und <pos> vom realen Typ.

Rückgabeformat: Die Abfrage gibt den Versatzwert in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt den vertikalen Offset von Kanal 1 auf 0,01 V.

:REfERENCE 1:VPOSition 0,01

Die folgende Abfrage gibt „1,000000e-02“ zurück

:REfERENCE1:VPOSition?

~~REF<n>HPOSition~~

Funktion: Legt den horizontalen Versatz der Wellenformanzeige fest.

Format: :REFerence:HPOSition <n>,<pos>

:REFerence:HPOSition? <n>

Dabei ist <n> vom Typ „diskret“ { 1|2|3|4} und <pos> vom Typ „reell“.

Rückgabeformat: Die Abfrage gibt den Versatzwert in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt den horizontalen Offset von R1 auf 2 µs.

:REFerence1:HPOSition 0,000002

Die folgende Abfrage gibt „2,000000e-06“ zurück

:REFerence1:HPOSition?

~~REF<n>SPR~~

Abfrage der Abtastrate der Referenzwellenform

Dabei ist n ein reeller Wert {{ 1|2|3|4}}

~~REF<n>MB~~

Abfrage der Speichertiefe der Referenzwellenform

Dabei ist n ein reeller Wert { 1|2|3|4}

3.2.14 AUTO-Einstellungssystem

~~AUTO:SET:CHANnel~~

Funktion: Aktiviert die automatische Einstellung des Kanals zum automatischen Öffnen und Schließen

Format: :AUTO:SET:CHANnel <bool>

:AUTO:SET:CHANnel?

Dabei ist <bool> vom Typ „Boolean“ und kann die Werte {{0|OFF}}|{{1||ON}} annehmen.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl schaltet die Anzeige von Kanal 1 ein.

:AUTO:SET:CHANnel ON oder :AUTO:SET:CHANnel1

Die folgende Abfrage gibt „1“ zurück.

:AUTO:SET:CHANnel?

~~AUTO:SET:LEVel~~

Funktion: Schwellenwert für das automatische Öffnen und Schließen von Kanälen im Automatikmodus

Format: :AUTO:SET:LEVel <level>

:AUTO:SET:LEVel?

Dabei ist <level> ein reeller Wert im Bereich von 0,001 V bis 99 V.

Rückgabeformat: Die Abfrage gibt den gültigen Schwellenwert in wissenschaftlicher Notation zurück.

Beispiel:

Der folgende Befehl setzt den effektiven Schwellenwert auf 150 mV.

:AUTO:SET:LEVnel 0,15

Die folgende Abfrage gibt „1,500000e-01“ zurück.

:AUTO:SET:LEVel?

AUTOSET

Funktion: Bei Ausführung von „autoset“ wird zunächst die Regel für die Triggerquelle ausgewählt: aktuelle Priorität | Priorität des Maximalwerts

Format: :AUTO:SET:SOURce <source>

:AUTO:SET:SOURce?

Dabei ist <source> diskret und kann den Wert { CURrent | MAX } annehmen.

Rückgabeformat: Die Abfrage gibt „CURrent“ oder „MAX“ zurück.

Beispiel:

Wenn der folgende Befehl auf „automatisch“ gesetzt ist, hat der Maximalwert der Triggerquelle Vorrang

:AUTO:SET:SOURce MAX

Die folgende Abfrage gibt „MAX“ zurück.

:AUTO:SET:SOURCE?

~~AUTO:RANGE~~

Funktion: Der durch Drücken der Auto-Taste ausgeführte Modus, unterteilt in „autoset“ und „autorange“

Format: :AUTO:RANGE <bool>

:AUTO:RANge?

Dabei ist <bool> ein boolescher Wert, {{0|OFF}}{{1||ON}}; bei 0 befindet sich das System im „AutoSet“-Modus, bei 1 im „AutoRange“-Modus.

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl aktiviert „Autorange“.

:AUTO:RANge ON oder :AUTO:RANge 1

Die folgende Abfrage gibt „1“ zurück.

:AUTO:RANge?

~~AUTO:Range~~ **AUTO:VERTical**

Funktion: Gibt an, ob der vertikale Skalierungsfaktor während der automatischen Bereichsanpassung automatisch angepasst wird

Format: :AUTO:RANge:VERtical <bool>

:AUTO:RANge:VERtical?

Dabei ist <bool> ein boolescher Wert, {{0|OFF}}{{1||ON}}

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl aktiviert VERtical automatisch.

:AUTO:RANge:VERtical ON oder :AUTO:RANge:VERtical 1

Die folgende Abfrage gibt „1“ zurück.

:AUTO:RANge:VERtical?

~~AUTO:Range~~ **AUTO:HORiz**

Funktion: Gibt an, ob die Zeitbasis während des AutoRange-Vorgangs automatisch eingestellt wird

Format: :AUTO:RANge:HORizontal <bool>

:AUTO:RANge:HORizontal?

Dabei ist <bool> ein boolescher Wert, {{0|OFF}}{{1||ON}}

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl aktiviert HORizontal automatisch.

:AUTO:RANge:HORizontal ON oder :AUTO:RANge:HORizontal 1

Die folgende Abfrage gibt „1“ zurück.

:AUTO:RANge:HORizontal?

~~AUTO:RANge:LEVel~~

Funktion: Gibt an, ob der Auslösepegel während des AutoRange-Vorgangs automatisch festgelegt wird

Format: :AUTO:RANge:LEVel <bool>

:AUTO:RANge:LEVel?

Dabei ist <bool> vom Typ „Boolean“ und kann die Werte {{0|OFF}}|{1|ON}} annehmen

Rückgabeformat: Die Abfrage gibt „0“ oder „1“ zurück.

Beispiel:

Der folgende Befehl schaltet LEVel automatisch ein.

:AUTO:RANge:LEVel ON oder:AUTO:RANge:LEVel 1

Die folgende Abfrage gibt „1“ zurück.

:AUTO:RANge:LEVel?

3. 2. 15 Befehlssystem „Waveform“

:WAVEform:SOURce

:WAVEform:MODE

:WAVEform:FORMat

:Wellenform:DATEN?

:WAVEform:START

:WAVEform:STOP

:WAVEform:PREamble?

:Wellenform:Inkrement?

:WAVEform:XORigin?

:WAVEform:XREFERENCE?

:WAVEform:Y-Inkrement?

:WAVEform:YORigin?

:WAVEform:YREFERENCE?

~~WAVEform~~

Funktion: Legt die Kanalquelle für das Einlesen der Wellenform fest.

Format: :WAVEform:SOURce <source>

:WAVeform:SOURce?

Dabei ist < source > ein diskreter Typ, {CH1|CH2|CH3|CH4}

Rückgabeformat

Die Abfrage gibt „CH1“, „CH2“, „CH3“ oder „CH4“ zurück.

Beispiel

Der folgende Befehl wählt Kanal 2 als Kanalquelle aus.

:WAVeform:SOURce CH2

Die folgende Abfrage gibt „CH2“ zurück.

:WAVeform:SOURce?

~~WAVE~~MODE

Diese Funktion legt den Modus zum Einlesen von Wellenformen fest oder fragt ihn ab.

Format :WAVeform:MODE < Modus >

:WAVeform:MODE?

Dabei ist < Modus > diskret und kann einen der folgenden Werte annehmen: { NORMAl | MAXimum | RAW }

Beispiel

NORMAl: Gibt die Anzahl der Wellenformpunkte nach der Abtastung zurück.

MAXimum: Gibt die maximale Anzahl gültiger Datenpunkte im aktuellen Zustand zurück. Im laufenden Zustand gibt es die Anzahl der auf dem Bildschirm angezeigten Datenpunkte zurück, und im angehaltenen Zustand die Anzahl der Datenpunkte im Speicher.

RAW: Gibt die Anzahl der Datenpunkte im aktuellen Systemspeicher zurück. Nur im angehaltenen Zustand gültig.

Rückgabeformat

Die Abfrage gibt „NORMal“, „MAXimum“ oder „RAW“ zurück.

Beispiel

Der folgende Befehl wählt den RAW-Modus aus.

```
:WAVeform:MODE RAW
```

Die folgende Abfrage gibt „RAW“ zurück.

```
:WAVeform:MODE?
```

~~WAV~~FORMAT

Das Format der Daten, die zurückgegeben werden, wenn **die Funktion** die Daten festlegt oder abfragt.

Format :WAVEform:FORMat <Format>

:WAVEform:FORMat?

Dabei ist <Format> ein diskreter Typ, { WORD| ASCii }

veranschaulicht

WORD: Die Daten eines Punktes belegen 16 Bit (zwei Bytes), was die vertikale Wertgröße angibt.

ASCII: Die Daten des Messpunkts werden in wissenschaftlicher Notation dargestellt und durch Kommas getrennt, zum Beispiel +3,590104E- 02, - 7,180208E-02, -7,180208E-02, +0,000000E+00, -3,590104E-02, -3,590104E-02, -7,180208E-02.

Ausgabeformat

Die Abfrage gibt „WORD“ oder „ASCII“ zurück.

Beispiel

Der folgende Befehl wählt den WORD-Modus aus.

:WAVEform:FORMat WORD

Die folgende Abfrage gibt „WORD“ zurück.

:WAVEform:FORMat?

WAVEform:START

Diese Funktion legt die Startposition der gelesenen Daten fest oder fragt sie ab.

Format :WAVEform:START < no >

:WAVEform:START?

Dabei ist <no> eine ganze Zahl, deren Wert vom eingestellten FORMat-Typ abhängt

Im NORMAL-Modus: 1 bis zum maximalen horizontalen Pixelwert des Bildschirm-Wellenformbereichs (Pixel pro Raster * Anzahl der horizontalen Raster)

MAX: 1 bis zur Anzahl der gültigen Punkte auf dem aktuellen Bildschirm

RAW: 1 bis zum Maximalwert der aktuellen Speichertiefe

Veranschaulichung

Wenn die Datenmenge auf dem Bildschirm groß ist, kann sie in der Regel nicht auf einmal gelesen werden und muss mehrmals gelesen werden. In diesem Fall müssen Start- und Endpunkt jedes Lesevorgangs festgelegt werden.

Rückgabeformat

Die Abfrage gibt einen ganzzahligen Wert zurück.

Beispiel

Der folgende Befehl legt den Startpunkt auf 500 fest.

:WAVEform:START 500

Die folgende Abfrage gibt „500“ zurück.

:WAVEform:START?

~~WAVEform:STOP~~

Diese Funktion legt die Endposition für das Auslesen von Daten fest oder fragt sie ab.

Format :WAVEform:STOP < no >

:WAVEform:STOP?

Dabei ist <no> eine ganze Zahl, deren Wert vom festgelegten FORMat-Typ abhängt

Im NORMAL-Modus: 1 bis zum maximalen horizontalen Pixelwert des Bildschirm-

Wellenformbereichs (Pixel pro Raster * Anzahl der horizontalen Raster)

MAX: 1 bis zur Anzahl der gültigen Punkte auf dem aktuellen Bildschirm

RAW: 1 bis zum Maximalwert der aktuellen Speichertiefe

Veranschaulichung

Wenn die Datenmenge auf dem Bildschirm relativ groß ist, kann sie in der Regel nicht auf einmal gelesen werden und muss mehrmals gelesen werden. In diesem Fall müssen Start- und Endpunkt jedes Lesevorgangs festgelegt werden. Der Wert des Endpunkts muss größer oder gleich dem Wert des Startpunkts sein.

Rückgabeformat

Die Abfrage gibt einen ganzzahligen Wert zurück.

Beispiel

Der folgende Befehl legt den Startpunkt auf 1000 fest.

:WAVEform:STOP 1000

Die folgende Abfrage gibt „1000“ zurück.

:WAVEform:STOP?

~~WAVEform~~

Funktion: Liest schnell alle auf dem Bildschirm angezeigten Wellenformdaten ein.

Format:

:WAVEform:DATA:HEX? Gibt Hexadezimalzeichen zurück

:WAVEform:DATA:BIN? Gibt Binärdaten zurück

:WAVEform:DATA:ASCii? Gibt Wellenformdaten zurück

Rückgabeformat:

Die zurückgegebenen Daten bestehen aus vier Teilen: einer Kennung, einem

Datenlängen-Deskriptor, der Datenlänge und den Wellenformdaten.

#MdddddddddXXXX

In diesem Format ist # die Kennung. M gibt an, dass die folgenden M Ziffern die Gesamtbytelänge der zurückgegebenen Wellenformdaten angeben, dargestellt durch dddddddd. Die nachfolgenden XXXX enthalten die Wellenformdaten.

Hinweis: Die Menge der zurückgegebenen Daten ist groß; stellen Sie sicher, dass der Empfangspuffer mit ausreichender Größe konfiguriert ist.

Beispiel für eine schnelle Erfassung vollständiger Wellenformdaten:

:WAVeform:SOURce CH1

:WAVeform:MODE NORMAl

:WAVeform:DATA:BIN?

Hinweis: WAVeform:DATA:<type>? wird nur bei 12-Bit-Oszilloskopen unterstützt.

~~MA6DP~~

Die Funktion liest Wellenformdaten ein.

Format :WAVeform:DATA?

Dieser Befehl wird von den Befehlseinstellungen :WAVeform:SOURce, :WAVeform:FORMAt,

:WAVeform:MODE und anderen beeinflusst.

Veranschaulichung

Bildschirmdarstellung des Lesevorgangs der Wellenformdaten:

S1. :WAV:SOURce CH1

Legt die Quelle für das Einlesen fest

S2. :WAV:MODE NORM	Der Wellenformmodus ist NORM
S3. :WAV:FORMat BYTE	Legt das Datenrückgabeformat auf BYTE fest
:WAV:DATA?	Daten auf dem Bildschirm anzeigen

Auslesen der Wellenformdaten aus dem Speicher:

S1. :MENU:STOP	Die gespeicherte Wellenform kann nur im Stoppzustand ausgelesen werden
S2. :WAV:SOURce CH1	Quelle für das Auslesen festlegen
S3. :WAV:MODE	Der RAW-Wellenformmodus ist RAW
S4. :WAV:FORMat BYTE	Legt das Datenrückgabeformat auf WORD fest
S5.:WAVEform:START 1	Stellt die Startposition des Lesepunkts auf 1
ein S6. :WAVEform:STOP 62500	Setzt die Endposition des Lesepunkts auf 62500
S7. :WAV:DATA?	Ruft die Daten aus dem Puffer ab

Rückgabe

Wenn :WAV:FORM auf WORD gesetzt ist,

bestehen die zurückgegebenen Daten aus 4 Teilen: Kennung, Beschreibung der Datenlänge,

Datenlänge und Wellenformdaten.

#M ddd ddddXXXX

eines davon #

Kennung

M steht für die ersten M Bits der folgenden Daten, die die Gesamtanzahl der Bytes der zurückgegebenen Wellenformdaten beschreiben, ausgedrückt als ddd dddd; die folgenden X XXX sind die Wellenformdaten

Beispiel:

Beim Senden des Befehls :WAV:DATA? werden die Daten zurückgegeben. # 90 00 00 10 24 80 81 82 83 89

ist ein Bezeichner 9. Die darauf folgenden 9 Bits stellen die Anzahl der Abtastpunkte der zurückgegebenen Daten dar. 0 00 00 10 24 sind insgesamt 9 Datenbits, was angibt, dass das Datenvolumen 1024 Abtastpunkte beträgt.

Hinweis:

Ist die Datenmenge im Speicher groß, muss der Benutzer sie in mehreren Durchgängen einlesen, wobei jedes Mal ein Datenblock gelesen und anschließend die jeweils gelesenen Daten miteinander verknüpft werden. Die bei jedem Durchgang gelesene Datenmenge wird durch das festgelegte Datenrückgabeformat (:WAV:FORMat) bestimmt, wie in folgende Tabelle dargestellt:

Einstellungsformat	Maximale Datenmenge, die auf einmal gelesen werden kann	Anmerkung
WORD	62.5000 Abtastpunkte	
ASCII	15.625	

Beispielsweise

Die Datenmenge im Speicher beträgt 220K, und das Datenrückgabeformat ist auf WORD eingestellt; da im WORD-Modus pro Lesevorgang 62.500 Daten gelesen werden und insgesamt 220.000 Daten gelesen werden müssen, sollte der Lesevorgang viermal durchgeführt werden;

S6.:WAVEform:STOP 187500 Setze die Endposition des Lesepunkts auf
187500 S7.:WAV:DATA? Das dritte Datensegment abrufen

Lese den vierten Datenwert

S5.:WAVEform:START 187501 Setzt die Startposition des Lesepunkts auf 187501

S6.:WAVEform:STOP 220000 Setzt die Endposition des Lesepunkts auf 220000

S7.:WAV:DATA? Das vierte Datensegment

abrufen Verbinden Sie alle Segmente, um die Daten des gesamten
Speichers abzurufen.

WAVEform

Befehlsformat :WAVEform:PREamble?

Funktionsbeschreibung Alle Wellenformparameter abfragen und

zurückgeben. Rückgabeformat

Die Abfrage gibt 9 Wellenformparameter zurück, die durch „ , “ getrennt sind:

<format>,<type>,<count>,<xincrement>,<xorigin>,<xreference>,<yincrement>,<yorigin>,<yreference
>

<format>: 0 (WORD) oder 2 (ASCII). Siehe Befehl :WAVEform:FORMat.

<type>: 0 (NORMal), 1 (MAXimum) oder 2 (RAW). Siehe Befehl :WAVEform:MODE.

<count>: Die durchschnittliche Anzahl der Abtastungen im

Durchschnittsabtastmodus (siehe Befehl :ACQuire:AVERages) und 1 in
anderen Modi.

<xincrement>: Die Zeitdifferenz zwischen zwei benachbarten Punkten in X-Richtung.

Siehe Befehl :WAVEform:XINCrement?.

<xorigin>: Die Zeit vom Triggerpunkt bis zur „Referenzzeitbasis“ in X-Richtung. Siehe Befehl

:WAVeform:XORigin?.

<xreference>: Die Referenzzeitbasis des Datenpunkts in X-Richtung. Siehe Befehl

:WAVeform:XREference?.

< yincrement >: Spannungswert in der Y-Richtung. Siehe Befehl :WAVeform:YINCrement?.

< yorigin >: Der vertikale Versatz in Y-Richtung relativ zur „vertikalen Referenzposition“

(Befehl „reference :WAVeform:YREference?“). Siehe Befehl „:WAVeform:YORigin?“.

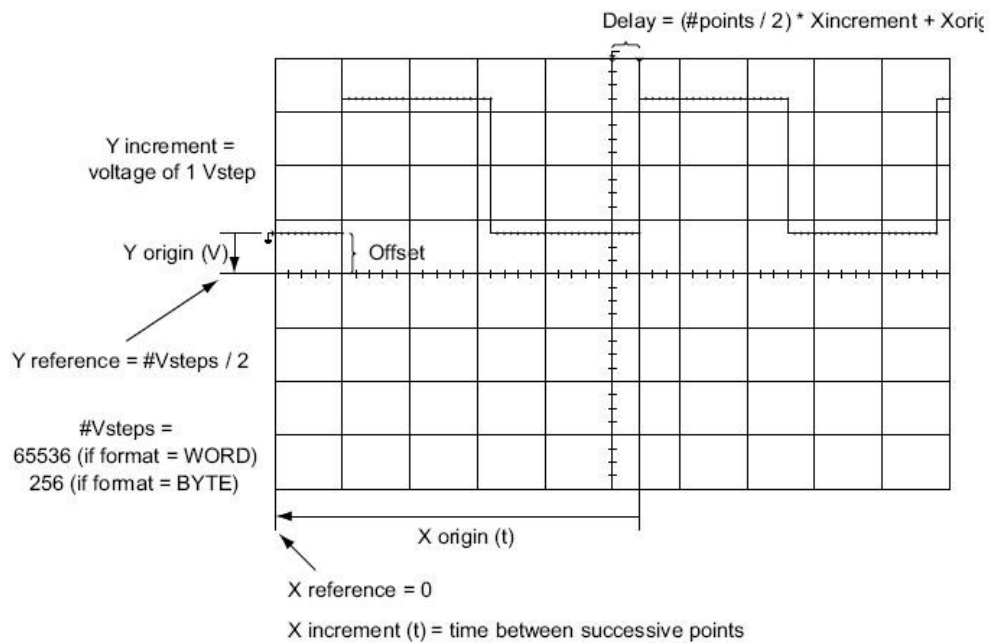
< yreference >: Die vertikale Referenzposition in Y-Richtung. Siehe

Befehl :WAVeform:YREference?.

Beispiel

Die folgende Abfrage gibt „1,2,1,0,000000,-0,001488,0,0,062500,3,968750,127“ zurück.

:WAVeform:PREamble?



WAVEform:XINCrement

Format :WAVEform:XINCrement?

Funktionsbeschreibung Abfrage des Zeitintervalls zwischen zwei benachbarten Punkten in X-Richtung der aktuell ausgewählten Kanalquelle.

Der Rückgabewert hängt vom aktuellen Datenlesemodus ab:

Im NORMAl-Modus gilt: XINCrement = 1/mathematische Abtastrate der

Wellenform. Im RAW-Modus gilt: XINCrement = 1/SampleRate.

Im MAX-Modus gilt: Wenn das Gerät läuft, ist XINCrement = 1/mathematische Abtastrate der Wellenform; wenn das Gerät angehalten ist, ist XINCrement = 1/SampleRate.

Die Einheiten beziehen sich auf die aktuelle

Kanalquelle. Rückgabeformat

Die Abfrage gibt das Zeitintervall in wissenschaftlicher Notation

zurück. Beispiel

Die folgende Abfrage gibt „2,000000e-08“ zurück.

:WAVEform:XINCrement?



Befehlsformat :WAVEform:XORigin?

Funktionsbeschreibung

Abfrage der angegebenen Quelle (Befehl „Reference:WAVEform:SOURce“). Der erste

Wellenformpunkt in X-Richtung, die Zeit bis zur Triggerposition (berechnet mit der Triggerposition

als Basis 0), die Einheit ist s.

Der Rückgabewert hängt vom aktuellen Datenlesemodus ab:

Im NORMAl-Modus wird die Zeit vom ersten Wellenformpunkt auf dem Bildschirm bis zur

Triggerposition zurückgegeben.

Im RAW-Modus wird die Zeit vom ersten Wellenformpunkt im Speicher bis zur Triggerposition

zurückgegeben.

Im MAX-Modus gibt das Gerät im laufenden Zustand die Zeit vom ersten Wellenformpunkt auf

dem Bildschirm bis zur Triggerposition zurück; im angehaltenen Zustand gibt es die Zeit vom

ersten Wellenformpunkt im Speicher bis zur Triggerposition zurück.

Ausgabeformat

Die Abfrage gibt den Zeitwert in wissenschaftlicher Notation

zurück. Beispiel

Die folgende Abfrage gibt „-7,000000e-06“ zurück.

:WAVEform:XORigin?

~~1 WAVEFORM~~

Befehlsformat :WAVEform:XREFerence?

Funktionsbeschreibung

Fragt bei der angegebenen Quelle (Befehl „reference:WAVEform:SOURce“) die Referenzzeitbasis der Datenpunkte in X-Richtung ab. Die Einheit ist s, in wissenschaftlicher Notation, wie oben beschrieben.

Rückgabeformat

Die Abfrage gibt die Zeitbasis als Ganzzahl zurück.

Beispiel

Die folgende Abfrage gibt „0“ zurück.

:WAVEform:XREFerence?

~~1 WAVEFORM~~

Format :WAVEform:YINCrement?

Funktionsbeschreibung

Abfrage des Spannungswerts in der Y-Richtung der angegebenen Quelle (Befehl „reference:WAVEform:SOURce“). Die Einheit entspricht der für die Quelle ausgewählten Einheit.

Rückgabeformat

Die Abfrage gibt den Spannungswert in wissenschaftlicher

Notation zurück. Beispiel

Die folgende Abfrage gibt „3,125000e-03 V“ zurück.

:WAVEform:YINCrement?

~~WAVEform:YORigin?~~

Format :WAVEform:YORigin?

Funktionsbeschreibung

Abfrage des vertikalen Versatzes der angegebenen Quelle (Befehl reference:WAVEform:SOURce) in Y-Richtung relativ zur „vertikalen Referenzposition“ (Befehl reference:WAVEform:YREference?). Die Einheit entspricht der für die Quelle ausgewählten Einheit.

Rückgabeformat

Die Abfrage gibt den Versatzwert in wissenschaftlicher

Notation zurück. Beispiel

Die folgende Abfrage gibt „3,968750e+00 V“ zurück.

:WAVEform:YORigin? ~~WAVEform:YORigin?~~ **WAVEform:YREference?**

Befehlsformat :WAVEform:YREference?

Funktionsbeschreibung

Abfrage der vertikalen Referenzposition der angegebenen Quelle (siehe :WAVEform:SOURce) in der Y-Richtung ab. Die Einheiten entsprechen den für die Quelle ausgewählten Einheiten.

Rückgabeformat

Die Abfrage gibt die Referenzposition als Ganzzahl zurück.

Beispiel

Die folgende Abfrage gibt „127“ zurück.

:WAVeform:YREFerence?

Kontakt

E-Mail:sales@micsig.com

Unternehmen: Shenzhen Micsig Technology Co., Ltd.

Adresse: 6. Stock, Jinhuanu-Gebäude, Nr. 56, Tiezai-Straße, Bezirk Bao'an,
Shenzhen, Guangdong, China.

Änderungen an dieser Anleitung sind ohne vorherige Ankündigung vorbehalten.

Der Inhalt dieser Anleitung wird als korrekt angesehen. Sollte der Benutzer Fehler, Auslassungen usw. feststellen, wenden Sie sich bitte an Micsig.

Das Unternehmen haftet nicht für Unfälle und Schäden, die durch eine unsachgemäße Bedienung durch den Benutzer verursacht werden.

Das Urheberrecht an diesem Handbuch liegt bei Micsig. Keine Organisation und keine Einzelperson darf es ohne die Genehmigung von Micsig vervielfältigen, kopieren oder Auszüge daraus erstellen. Micsig behält sich das Recht vor, die oben genannten Handlungen strafrechtlich zu verfolgen.