

Micsig Oszilloskop SCPI- Befehlshandbuch



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

Inhalt

1.	Einführung	1
1.1	Zweck dieses Dokuments	1
1.2	Zielgruppe.....	1
1.3	Referenz	1
1.4	Begriffsbestimmungen	1
2.	Produktbesch	2
2.1	Hintergrundinformationen zum Produkt	2
2.2	Zielgruppen und Nachfragebeschreibung	2
2.3	Rollen im Produkt.....	2
2.4	Geschäftsablauf.....	2
2.5	Ziel-Betriebssoftware und -Hardwareumgebung.....	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	Befehl.....	4

3.1.2 Erläuterung der Symbole.....	4
3.1.3 Parametert	5
3.1.4 Befehlskürzel	5
3.2 Befehl	6
3.2.1 Allgemeine Befehle.....	6
3.2.2 :MENU Menüfunktionsbefehle	7
3.2.3 Abt-Subsystem.....	15
3.2.4 Kanalbefehl-Subsystem.....	24
3.2.5 Mathematik-Subsystem	33
3.2.6 Cursor-Befehlssystem	50
3.2.7 Anzeige-Befehlssystem	61
3.2.8 Messbe-Subsystem.....	66
3.2.9 Trigger-Befehlssystem.....	79
3.2.10 Zeitbasis-Befehlssystem.....	114
3.2.11 Speicherbefehls-Subsystem	117
3.2.12 Buskonfigurationsbe-Subsystem.....	122
3.2.13 Referenz-Befehlssystem.....	137

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

3. 2.15	Wellenform-Befehlssubs	147
---------	------------------------------	-----

1. Dokument „“ (Einführung)

1.1 Zweck dieses Dokumentes „”

Dieses Dokument dient dazu, die SCPI-Anforderungen für Oszilloskope zu definieren und Micsig-Oszilloskope auf die Unterstützung des SCPI-Protokolls und die Einhaltung des IEEE488.2-Standards vorzubereiten.

1.2 Zielgruppe der Dokumentation

Entwickler und Tester

1.3 Referenz sdokumentation

1.4 Terminologie

Abkürzungen und Begriffe	erklären
SCPI	Standardbefehle für programmierbare Instrumente – Handbuch
...	

2. Produkt beschreibung

2.1 Hintergrundinformationen zum Produkt

Das SCPI-Befehlsverarbeitungsmodul ist in unsere Produkte integriert, um den IEEE488.2-Standard zu erfüllen. Als SCPI-Gerät müssen wir es streng nach den Bestimmungen des IEEE488.2-Standards für Geräte entwickeln.

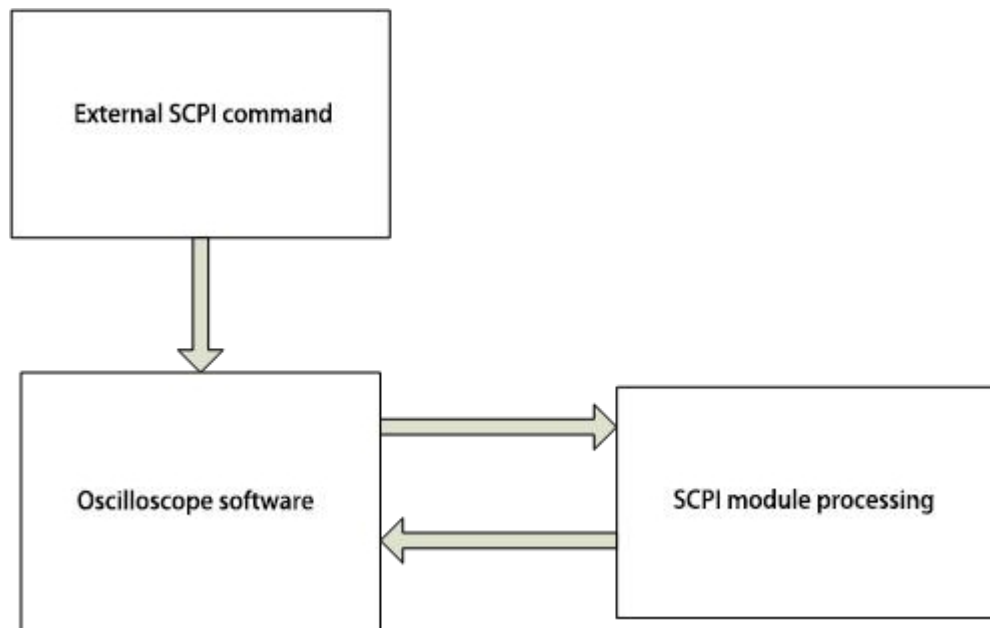
2.2 Zielgruppen und Nachfrage en zu den Produkten

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

2.3 Rollen im Produkt „ ”

Rolle Name	Verantwortlichkeiten
Gemeinsames Befehlssystem	Verarbeitet allgemeine Befehle für alle Instrumente
Grundlegendes Befehlssystem	Grundlegende Befehle für die Bedienung von Instrumenten und Geräten
andere	Optionale Befehle für die Handhabung von Instrumenten

2.4 Geschäfts -Workflow



2.5 Ziel-Betriebssoftware und Hardware- sumgebung

Betriebssystem: Android

Hardwareumgebung: Tablet-Oszilloskop

2.6 Einschränkungen und sbeschränkungen

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

Einschränkungen und Beschränkungen wie für das Produkt.

2.7 Verwendbare Schnittstellen en

USB, LAN, WLAN.

3. SCPI- sanforderungen

3.1 Einführung in die SCPI-

3.1.1 Befehls format

SCPI-Befehle sind baumartige hierarchische Strukturen, die mehrere Subsysteme umfassen, von denen jedes aus einem Stammschlüsselwort 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 nach 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 sind die Parameteroptionen. Parameterelemente werden in der Regel durch eine vertikale Linie „|“ getrennt. Bei der Verwendung eines Befehls müssen Sie einen der Parameter auswählen.

- 2、 Vertikale Linie |

Der senkrechte Strich wird verwendet, um mehrere Parameteroptionen voneinander zu trennen. Bei Verwendung eines Befehls müssen Sie einen der Parameter auswählen.

- 3、 Eckige Klammern []

Der Text in eckigen Klammern ist optional.

4、 Spitze Klammern < >

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

3.1.3 Parameter stypen

1、 Bool

Die Parameterwerte sind „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 Dezimalformat einstellen, da sonst eine Ausnahme auftritt.

4、 Real

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-Zeichenfolge

Der Parameterwert ist eine Kombination aus ASCII-Zeichen.

3.1.4 Befehls -Abkürzung

Bei allen Befehlen wird nicht zwischen Groß- und Kleinschreibung unterschieden, sie können also entweder komplett in Großbuchstaben oder komplett in Kleinbuchstaben eingegeben werden.

Wenn Sie jedoch Abkürzungen verwenden möchten, müssen Sie alle Buchstaben im Befehlsformat groß schreiben.

3.2 Befehls system

3.2.1 Gängige Befehle zur System

1*IDN

Funktion: Lesen Sie Informationen zum Oszilloskop. Dazu gehören die Versionsnummer, der Hersteller, das Produktmodell und die Seriennummer des Produkts.

Format: *IDN?

Rückgabeformat:

Micsig,< Modell>,<Seriennummer >,XXXXX

<Modell>: Gerätemodell.

<Seriennummer>: Seriennummer des Geräts.

XXXXX: Softwareversion des Geräts.

Beispiel:

Micsig,MDO5004,390000029,1.388.132

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

Format::SYS:SCR?

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

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 das nachfolgende FF\D8X\00\00\10JFIF\00\01\01\ steht für den Screenshot-Datenstrom.

3.2.2 :MENU Menüfunktions -Befehle

UTO

Die automatische Konfiguration kann das Oszilloskop schnell so konfigurieren, dass es das Eingangssignal optimal anzeigt. Die automatische Konfiguration umfasst: Anwendbar auf Einzelkanal und Mehrkanal; automatische Einstellung der horizontalen Skala, vertikalen Skala und des Triggerpegels des Signals; Oszilloskop-Wellenformumkehrung ist ausgeschaltet, Bandbreite ist auf volle Bandbreite eingestellt, Kopplungsmodus ist DC-Kopplung, Abtastmodus ist normale Abtastung; Trigger ist auf Flankentriggerung eingestellt, Triggermodus ist automatisch.

Funktion: Starten oder stoppen Sie die automatische Einstellung (Auto Range).

Die automatische Konfiguration kann das Oszilloskop schnell so konfigurieren, dass es das Eingangssignal optimal anzeigt. Die automatische Konfiguration umfasst: Anwendbar auf Einzelkanal und Mehrkanal; automatische Anpassung der horizontalen Skala, vertikalen Skala und Triggerpegel des Signals; Oszilloskop-Wellenformumkehrung ist ausgeschaltet, Bandbreite ist auf volle Bandbreite eingestellt, Kopplungsmodus ist DC-Kopplung, Abtastmodus ist normale Abtastung; Trigger ist auf Flankentrigger eingestellt, Triggermodus ist automatisch.

Format: :MENU:AUTO <bool>

:MENU:AUTO?

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

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?



Funktion: Starten Sie das Oszilloskop, erfüllen Sie die Triggerbedingungen und beginnen Sie mit der Datenerfassung.

Format: :MENÜ:RUN

TOP

Funktion: Stoppt das Oszilloskop und die Datenerfassung.

Format: :MENU:STOP

NGLE

Funktion: Stellen Sie das Oszilloskop auf Einzelsequenz ein. Das Oszilloskop erfasst und zeigt eine einzelne Erfassung an.

Format: :MENU:SINGLE

RESet

Funktion: Setzt die Oszilloskopeinstellungen auf die Werkseinstellungen zurück.

Format: :MENU:RESet

EDCK

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

Format: :MENU:LOCK <bool>

:MENU:LOCK?

Dabei ist bool ein boolescher Typ, {{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?

ALF

3.2.2.7.1 :MENU:HALF:CHANnel

Funktion: Setzt die Kanalposition auf die vertikale Nullposition (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: Setzt die Triggerposition auf die Mitte des Bildschirms.

Format: :MENU:HALF:TRIGpos <Quelle>

Wobei <Quelle> diskret ist, {CH1|CH2|CH3|CH4 }.

3.2.2.7.3 :MENU:HALF:XCURsor

Funktion: Setzt den vertikalen Cursor des Kanals auf 50 %.

Format: :MENU:HALF:XCURsor

3.2.2.7.4 :MENU:HALF:YCURsor

Funktion: Setzt den horizontalen Cursor des Kanals auf 50 %.

Format: :MENU:HALF:YCURSor

3.2.2.7.5 :MENÜ:HALB:LEVEL

Funktion: Stellt den Triggerpegel auf die mittlere Position der Triggersignalamplitude ein.

Format: :MENU:HALF:LEVel <Kanal>

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

~~2~~ :MENU:CHANnel

Funktion: Öffnen oder Schließen des Kanalmenüs

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

:MENU:CHANnel? <n>

Darunter <n>, diskreter Typ, {CH1|CH2|CH3|CH4 |MATH | REF|S1|S2 };

<bool>, Boolescher Typ, {{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

QUICK

Funktion: Öffnen oder Schließen des Kontextmenüs (unteres Menü)

Format: :MENU:QUICK <bool>

:MENU:QUICK?

Wobei <bool> ein Boolean-Typ ist, {{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?

MAIN

Funktion: Öffnen oder Schließen des Hauptmenüs (oberstes Menü)

Format: :MENU:MAIN <bool>

:MENU:MAIN?

Unter anderem <bool>, Boolescher Typ, {{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?

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?

Darunter <Typ> , diskreter Typ, {IN | OUT}.

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

Beispiel:

Der folgende Befehl wechselt zur Triggereingabe:

:MENU:AUX:TRIGger?

3.2.2.11.2 :MENU:AUX:INPutres

Funktion: Legt die Triggereingangsimpedanz fest.

Format: :MENU:AUX:INPutres <Typ>

:MENU:AUX:INPutres?

Darunter <Typ> — Diskreter Typ, {FIFTy | MEGA}.

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

Beispiel:

Der folgende Befehl setzt die Triggereingangsimpedanz auf 50 Ohm:

:MENU:AUX:INPutres FIFTy

LOCK

Funktion: Stellt den Takteingang/-ausgang ein.

Format: :MENU:AUX:CLOCK <Typ>

:MENU:AUX:CLOCK?

Unter ihnen <Typ>, diskreter Typ, {IN | OUT}.

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

Beispiel:

Der folgende Befehl wechselt zur Taktausgabe:

:MENU:AUX:CLOCK OUT

Die folgende Abfrage gibt „OUT“ zurück.

:MENU:AUX:CLOCK?

3.2.3 Befehl zum Abrufen von Abtastdaten -Subsystem

TYPE

Funktion: Legt die Abtastmethode fest.

Format: :ACQuire:TYPE <Typ>

:ACQuire:TYPE?

Wobei <type> diskret ist, { NORMAl | MEAN | ENVelop | PEAK }

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

Beispiel:

Der folgende Befehl wählt den Hüllkurven-Abtastmodus aus.

```
:ACquire:TYPE ENvelop
```

Die folgende Abfrage gibt „ENvelop“ zurück.

```
:ACquire:TYPE?
```

MEAN

Funktion: Legt die durchschnittliche Anzahl der Samples fest. Der festgelegte Wert ist ein ganzzahliges Vielfaches von 2.

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 setzt die durchschnittliche Anzahl der Samples auf „32“.

```
:ACquire:MEAN 32
```

Die folgende Abfrage gibt „32“ zurück.

```
:ACquire:MEAN?
```

ENvelop

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

Format: :ACQuire:ENVelop <Anzahl>

:ACQuire:ENVelop?

Wobei <count> ein diskreter Typ ist, {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 Envelope-Samples auf „32“.

:ACQuire:ENVelop 32

Die folgende Abfrage gibt „32“ zurück.

:ACQuire:ENVelop?

SEGmented

Funktion: Segment-Speicher einstellen.

3.2.3.4.1 :ACQuire:SEGmented <bool>

:ACQuire:SEGmented?

Festlegen und Abfragen, ob die Segment-Speicherung ein- oder ausgeschaltet ist;

Wert bool, boolescher Typ, {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 FPGA gespeichert wurden.

```
:ACQUIRE:SEGMENTED:NO?
```

3.2.3.4.3 :ACQUIRE:SEGMENTED:QTY <NO>

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

Legen Sie die Anzahl der in segmentierter Form gespeicherten Segmente fest und fragen Sie diese ab.

Wenn nein, ganzzahlige Zahl, siehe Datenhandbuch;

Beispiel

Der folgende Befehl legt die Anzahl der segmentierten Speichersegmente auf 4 fest.

```
: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 für segmentierten Speicher;

Wobei Typ , diskreter Typ, { SINGLE | FIT }; SINGLE ist eine Einzelbildanzeige, FIT ist eine angepasste Anzeige

Beispiel

Der folgende Befehl richtet 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 <Typ>

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

Legen Sie die Wiedergabereihenfolge für den Segment-Speicher fest und fragen Sie diese ab.

Wobei Typ, diskret, { ORD er | REOR der } ORD er die Reihenfolge ist, REOR der die umgekehrte Reihenfolge

Beispiel

Der folgende Befehl legt die sequenzielle Wiedergabe der Segment-Speicherung 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

Autoplay pausieren

3.2.3.4.7 :ACQuire:SEGmented:FRA1 <Wert>

:ACQuire:SEGmented:FRA1?

Legen Sie den aktuellen Frame fest und fragen Sie ihn ab, wenn Sie einen einzelnen Frame anzeigen.

Darunter Wert, ganzzahliger Typ, 1 bis Maximalwert beim Anhalten

Beispiel

Der folgende Befehl setzt den aktuellen Frame auf 546.

:ACQUIRE:SEGMmented:FRA1 546

Die folgende Abfrage gibt „546“ zurück.

:ACQUIRE:SEGMmented:FRA1?

3.2.3.4.8 :ACQUIRE:SEGMmented:FRA2 <Wert>

:ACQUIRE:SEGMmented:FRA2<Wert>

:ACQUIRE:SEGMmented:FRA2?

Legen Sie den Anfangsrahmen fest, wenn Sie die Abfrage anpassen möchten.

Wobei der Wert eine ganze Zahl zwischen 1 und dem Maximalwert beim Anhalten ist

Beispiel

Der folgende Befehl setzt den Anfangsrahmen der Anpassungsanzeige auf 100.

:ACQUIRE:SEGMmented:FRA2 100

Die folgende Abfrage gibt „100“ zurück.

:ACQUIRE:SEGMmented:FRA2?

3.2.3.4.9 :ACQUIRE:SEGMmented:FRA3 <Wert>

:ACQUIRE:SEGMmented:FRA3?

Legen Sie den Endrahmen der angepassten Anzeige fest und fragen Sie ihn ab.

Wobei Wert, Ganzzahl, FR2~der Maximalwert bei Stopp

Beispiel

Der folgende Befehl setzt den Endrahmen der angepassten 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?
```

SRATe

Funktion: Abfrage der Abtastrate des aktuellen Analogkanals.

Format: :ACQUIRE:SRATe? -

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} gesetzt werden.

Beispiel

Der folgende Befehl setzt die Speichertiefe auf AUTO.

:ACquire:DEPSelect 22000000

Die folgende Abfrage gibt „22000000 “ zurück.

:ACquire:DEPSelect?

DEPTH

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

Format: :ACquire:DEPTh?

Dabei gibt <type>, ein 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

~~CHANnel~~DISPlay

Funktion: Kanal öffnen oder schließen

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

:CHANnel <n>:DISPlay?

Dabei ist <n> ein diskreter Typ, { 1|2|3|4} ; <bool> ein boolescher Typ, {{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~~INVerse

Funktion: Schaltet die invertierte Anzeige des analogen Kanals ein oder aus.

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

:CHANnel <n>:INVerse?

Dabei ist <n> ein diskreter Typ, { 1|2|3|4} ; <bool> ein boolescher Typ, {{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~~BAND

Funktion: Legt die Bandbreitenbegrenzung des analogen Kanals auf „20M“ oder „Full Bandwidth“ fest.

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> , Real-Typ, {siehe Datenblatt}, nur gültig unter „HIGH“ und „LOW“.

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

Beispiel:

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

```
:CHANnel1:BAND HIGH , 10000000
```

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

```
:CHANnel1:BAND?
```

~~CHANnel~~PRTY

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

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

```
:CHANnel <n>:PRTY?
```

Darunter <n>, diskreter Typ, { 1|2|3|4} ; <Typ>, diskreter Typ, {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 Sondentyp Spannung auf.

```
:CHANnel1:PRTY VOL
```

Die folgende Abfrage gibt „VOL“ zurück.

```
:CHANnel1:PRTY?
```

~~CHANnel~~PROBe<atten>

Funktion: Einstellen des Dämpfungsverhältnisses der analogen Kanalsonde.

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}.

Rückgabeformat: 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?

~~KanalKopplungTyp~~

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?

~~KanalEingangswiderstand~~

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

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

:CHANnel <n>:INPutres?

Wobei <n> diskret ist, { 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?
```

~~CHANnelSCALe~~ ~~CHANnel~~ ~~EXETent~~



Funktion: Legt die vertikale Skalierung der Wellenformanzeige des angegebenen analogen Kanals fest.

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: Oszilloskop-Maximalstufe * aktueller Sondenmultiplikator

Minimalwert: Oszilloskop-Minimalstufe * aktuell eingestellter Messwandler-Multiplikator.

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

Beispiel:

Der folgende Befehl setzt die vertikale Skala von Kanal 1 auf 1 V/div.

```
: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?
```

Darunter: <n>, diskreter Typ, {1|2|3|4}; <pos>, reeller Typ.

Rückgabeformat: 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:REF~~

Funktion: Legt die vertikale Ausdehnungsbasis des analogen Kanals fest.

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

:CHANnel <n>:VREF?

Darunter: <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 Ausdehnungsbezugspunkt von Kanal 1 auf die Mitte fest.

:CHANnel1:VREF CENTER

Die folgende Abfrage gibt „CENT“ zurück.

:CHANnel1:VREF?

~~CHANnel~~LABel<Zeichenfolge>

Funktion: Legt die Kanalkennzeichnung des analogen Kanals fest.

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

:CHANnel <n>:LABel?

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

Rückgabeformat: Rückgabezeichenfolge der Abfrage.

Beispiel:

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

:CHANnel1:LABel DDR

Die folgende Abfrage gibt „ DDR “ zurück.

:CHANnel1:LABel?

~~CHANnel~~LEar

Funktion: Kanalbezeichnung löschen.

Format: :CHANnel <n>:LABel:CLEar

Darunter:<n> , diskreter Typ , { 1|2|3|4} .

Beispiel:

Der folgende Befehl löscht die Bezeichnung von Kanal 1 .

LEar

~~CHANnel~~<n>

Funktion: Legt den aktuellen Kanal fest.

Format: :Aktuell:Kanal <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 als Mittelpunkt fest.

:CURRENT:CHANnel CH1

Die folgende Abfrage gibt „ CH1 “ zurück.

:CURRENT:CHANnel?

3.2.5 Mathematikbefehl- -Subsystem

DISPlay

Funktion: Aktivieren oder deaktivieren Sie den mathematischen Operationstyp.

Format: :MATH:DISPlay <bool>

:MATH:DISPlay?

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

MODE

Funktion: Wählen Sie den Typ der mathematischen Operation aus.

Format: :MATH:MODE <Modus>

:MATH:MODE?

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

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

Beispiel:

Der folgende Befehl wählt die FFT-Operation aus.

:MATH:MODE FFT

Die folgende Abfrage gibt „FFT“ zurück.

:MATH:MODE?

VREF

Funktion: Legt die vertikale Ausdehnungsbasis der mathematischen Wellenform fest.

Format: :MATH:VREF <bool>

:MATH:VREF?

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

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

Beispiel:

Der folgende Befehl setzt die vertikale Ausdehnungsbasis auf die Mitte.

:MATH:VREF CENTer

Die folgende Abfrage gibt „CENT“ zurück.

:MATH:VREF?

BASE

3.2.5. 4 .1 :MATH:BASE:SOU1

Funktion: Quelle 1 für Dual-Wellenform-Betrieb auswählen

Format: :MATH:BASE:SOU1 <Quelle>

:MATH:BASE:SOU1?

Dabei ist <Quelle> 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 <Quelle>

:MATH:BASE:SOU2?

Dabei ist <Quelle> 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 die vertikale Skalierung des Ergebnisses der doppelten Wellenformoperation fest.

Format: :MATH:BASE:VSCale < Umfang >

:MATH:BASE:VSCale?

Dabei ist <Umfang> ein reeller Typ, <Umfang> ein reeller Typ, {1e-15~5e14, kann nur in Schritten von 1, 2, 5 sein}. .

Rückgabeformat: Die Abfrage gibt den Zahnradwert 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:VPosition

Funktion: Legt den vertikalen Versatz des Ergebnisses der Dualwellenformoperation fest.

Format: :MATH:BASE:VPosition <Position>

```
:MATH:BASE:VPosition?
```

Dabei ist <position> ein reeller Typ und wird in wissenschaftlicher Notation ausgedrückt.

```
:MATH:BASE:VPosition 8 /* Vertikalen Versatz auf 8 V festlegen*/
```

```
:MATH:BASE:VPosition? Die Abfrage gibt 8.000000E0* zurück
```

3.2.5.4.5 :MATH:BASE:OPERator

Funktion: Wählen Sie 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 FFT

3.2.5.5.1 :MATH:FFT:SOURce

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

Format: :MATH:FFT:SOURce <Quelle>

:MATH:FFT:SOURce?

Dabei ist <Quelle> 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:FFT:SOURce CH1

Die folgende Abfrage gibt „CH1“ zurück.

:MATH:FFT:SOURce?

3.2.5.5.2 :MATH:FFT:WINDow

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

Format: :MATH:FFT:WINDow <Quelle>

:MATH:FFT:WINDow?

Dabei ist <Quelle> ein diskreter Typ,

{ 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 „Logarithmic“ .

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 die vertikale Skalierung des FFT-Operationsergebnisses fest.

Format: :MATH:FFT:VScale <Ausdehnung>

```
:MATH:FFT:VScale?
```

Dabei ist <extent> ein reeller Typ, <extent> ein reeller Typ, in Zeilen, {1e-15~5e14, kann nur in Schritten von 1, 2, 5} oder in db {1~500, 1, 2, 5 Schritte} angegeben werden.

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

Beispiel:

Der folgende Befehl setzt die vertikale Skalierung des FFT-Operationsergebnisses auf 1.

```
:MATH:FFT:VScale 1
```

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

:MATH:FFT:VSCale?

3.2.5.5.5 :MATH:FFT:VPOsition

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

Format: :MATH:FFT:VPOsition <Position>

:MATH:FFT:VPOsition?

Dabei ist <positionoffset> ein reeller Typ und wird in wissenschaftlicher Notation ausgedrückt.

3.2.5.5.6 :MATH:FFT:HSCale

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

Format: :MATH:FFT:HSCale <hscale>

:MATH:FFT:HSCale?

Dabei ist < hscale > ein reeller Typ, {1 Hz bis 100 MHz, 1, 2, 5 Schritte}.

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

Beispiel:

Der folgende Befehl legt die horizontale Skala des FFT-Operationsergebnisses 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-Operationsergebnisses fest.

Format: :MATH:FFT:HPOSition <Position >

:MATH:FFT:HPOSition?

Dabei ist <position> ein reeller Typ.

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?

§ AX+B

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 <Quelle>

:MATH:AX+B:SOURce?

Dabei ist <Quelle> 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: Wählen Sie den Wert von A in AX+B aus.

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

:MATH:AX+B:A?

Unter diesen <a> realer Typ, den Bereich entnehmen Sie bitte dem Handbuch.

Rückgabeformat: Die Abfrage gibt einen Realwert 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: Wählen Sie den Wert von B in AX+B aus.

Format: :MATH:AX+B:B

:MATH:AX+B:B?

Dabei ist ein reeller Typ, dessen Bereich Sie dem Handbuch entnehmen können.

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: Wählen Sie die Einheit in AX+B aus.

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

:MATH:AX+B:UNIT?

Dabei ist <Einheit> eine Zeichenfolge, deren Bereich im Datenblatt angegeben ist.

Rückgabeformat: Die Abfrage gibt einen reellen Wert zurück.

Beispiel:

Der folgende Befehl math units.

: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 die vertikale Skalierung des Operationsergebnisses fest.

Format: :MATH:AX+B:VScale <Ausdehnung>

:MATH:AX+B:EXTent?

Unter diesen ist <extent> ein realer Typ, {1e-15~5e14, kann nur in Schritten von 1, 2, 5 }.

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

Beispiel: Der folgende Befehl setzt die vertikale Skalierung 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 > ein reeller Typ und wird in wissenschaftlicher Notation ausgedrückt.

ADVanced

3.2.5.7.1 :MATH:ADVanced:EXPRession

Funktion: Legt den Ausdruck für erweiterte Operationen fest.

Format: :MATH:ADVanced:EXPRession <Zeichenfolge>

:MATH:ADVanced:EXPRession?

Dabei ist <Zeichenfolge> eine ASCII-Zeichenfolge.

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:ADVanced:EXPRession?
```

3.2.5.7.2 :MATH:ADVanced:VAR1

Funktion: Legt die Variable im erweiterten Operationsausdruck 1 fest.

Format: :MATH:ADVanced:VAR1 <Wert>

Wobei <Wert> ein reeller Typ ist, der zwischen -9,9999E+9 und 9,9999E+9 liegt. 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: Variable 2 in erweiterter Operationsausdruck festlegen

Format: :MATH:ADVanced:VAR2 <Wert>

Dabei ist <Wert> ein reeller Typ zwischen -9,9999E+9 und 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 erweiterten Berechnungsergebnisse fest.

Format: :MATH:ADVanced:VSCale <Ausmaß>

```
:MATH:ADVanced:VSCale?
```

Dabei ist <Umfang> ein reeller Typ, {1e-15~5e14, kann nur um 1, 2, 5 erhöht werden}.

Rückgabeformat: Die Abfrage gibt den Zahnradwert 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 Operation fest.

Format: :MATH:ADVanced:VPOSiton <Position>

:MATH:ADVanced:VPOSiton?

Dabei ist <Position> ein reeller Typ, ausgedrückt in wissenschaftlicher Notation.

3.2.5.7.6 :MATH:ADVanced:UNIT <Einheit>

Funktion: Wählen Sie Einheiten in ADVanced aus.

Format: :MATH:ADVanced:UNIT <Einheit>

:MATH:ADVanced:UNIT?

Darunter <Einheit>, Zeichenfolge.

Rückgabeformat: Die Abfrage gibt einen reellen Wert zurück.

Beispiel:

Der folgende Befehl mathematische Einheiten.

:MATH:ADVanced:UNIT W

Die folgende Abfrage gibt „W“ zurück.

:MATH:ADVanced:UNIT?

3.2.5

Abfrage der Abtastrate mathematischer Wellenformen, wobei der Rückgabewert in wissenschaftlicher Notation ausgedrückt wird.

Beispiel:

Die folgende Abfrage gibt „2.500000e8“ zurück.

:MATH:SRATe?

3.2.6

Abfrage der Anzahl der Punkte der mathematischen Wellenform. Der zurückgegebene Wert wird in wissenschaftlicher Notation ausgedrückt.

Beispiel:

Die folgende Abfrage gibt „7.000000e2“ zurück.

:SAMPleACQuire:MATH:DEPth?

3.2.6 Cursor-Befehl „-Subsystem“

HORizontal

Funktion: Aktiviert oder deaktiviert die horizontale Cursorfunktion.

Format: :CURSor:HORizontal <bool>

:CURSor:HORizontal?

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

VERTical

Funktion: Schaltet die vertikale Cursorfunktion ein oder aus.

Format: :CURSor:VERTical <bool>

:CURSor:VERTical?

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

CX1

Funktion: Legt die Position des vertikalen Cursors X1 fest.

Format: :CURSor:CX1 <px>

:CURSor:CX1?

Wobei <px> eine ganze Zahl in Pixeln ist.

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?
```

CX2

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?
```

Y1

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?

Y2

Funktion: Position des horizontalen Cursors 2 festlegen.

Format: :CURSor:CY2<px>

:CURSor:CY2?

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 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 werden durch die aktuellen horizontalen Einheiten bestimmt.

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 wird durch die aktuelle horizontale Einheit bestimmt.

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?

Y1Value

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

Format: :CURSor:Y1Value?

Die Einheit des Abfragewerts wird durch die aktuelle vertikale Einheit bestimmt.

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?

Y2Value

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: Abfrage der Differenz zwischen den vertikalen Cursors X1 und X2 Δ . 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: Abfrage der Differenz zwischen dem horizontalen Cursor Y1 und Y2 Δ , die Einheit entspricht der vertikalen Einheit.

Format: :CURSor:YDELta?

Rückgabeformat: 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 1/ zwischen den vertikalen Cursorn 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?

RATIO

Funktion: Abfrage des Verhältnisses zwischen der Differenz zwischen den horizontalen Cursorn A und B und der Differenz zwischen den vertikalen Cursorn A und B.

Format: :CURSor:RATIo?

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

Beispiel:

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

:CURSor:RATio?

SOURce

Funktion: Legt die Kanalquelle für die Cursormessung fest.

Format: :CURSor:SOURce <Quelle>

: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

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?

Darunter: <bool> , <bool>, Boolescher Typ, {{0|OFF}}{1|ON}}; <src1> , Ganzzahl, steht für die Anzahl der einzustellenden Zylinder ; <src2> , Ganzzahl, steht für den eingestellten Winkel, der in der Regel ein Vielfaches von 1 80 ist ;

Rückgabeformat: Die Abfrage gibt die entsprechenden Parameter zurück.

Beispiel:

Der folgende Befehl schaltet den Phasencursor ein und setzt den Cursor auf 4 Zylinder, 720 Grad.

```
: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 Seite des Bildschirms als Referenz 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: Legt die Pixelposition der letzten Cursorlinie fest oder fragt sie ab.

Format: :PHCUrsor:YN <px>

:PHCUrsor:YN?

Dabei ist < px > eine ganze Zahl, die die Pixelposition der letzten Cursorzeile bezogen auf die linke Seite des Bildschirms 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: XN 200

Die folgende Abfrage gibt „200“ zurück.

3.2.7 Anzeigebefehl -Subsystem

WAVeform

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 Wellenform-Anzeigemodus auf „DOTS“ ein.

:DISPlay:WAVeform DOTS

Die folgende Abfrage gibt „DOTS“ zurück.

:DISPlay:WAVEform?

BRIGhtness

Funktion: Legt die Helligkeit der Wellenformanzeige auf dem Bildschirm fest.

Format: :DISPlay:BRIGhtness <Zeit>

: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?

GRATicule

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 <Zeit>

```
:DISPlay:INTensity?
```

Dabei ist <Zeit> 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: Persistenz-Anzeigemodus festlegen.

Format: :DISPlay:PERSist:MODE <Modus>

:DISPlay:PERSist:MODE?

Wobei <Modus> diskret ist, { AUTO|NORMAl|INFinite|none }.

3.2.7.5.2 :DISPlay:PERSist:ADJust

Funktion: Festlegen der Persistenzzeit im normalen Anzeigemodus

Format: :DISPlay:PERSist:ADJust <Zeit>

:DISPlay:PERSist:ADJust?

Wobei <Zeit> eine ganze Zahl in Millisekunden ist: 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: Nachleuchten der Anzeige löschen

Format: :DISPlay:PERSist:CLEar

IGH(@)Helligkeit

Funktion: Hohe Bildwiederholfrequenz ein- oder ausschalten

Format: :DISPlay:HIGH <bool>

:DISPlay:HIGH?

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

HORRef

Funktion: Legt den Modus für die horizontale Bildschirmvergrößerung fest: „Triggerpunkt“ oder „Bildschirmmitte“.

Format: :DISPlay:HORRef <Modus>

:DISPlay:HORRef?

Wobei <Modus> diskret ist und { CENTER|TRIGpos } ist.

ZOOM

Funktion: ZOOM öffnen oder schließen

Format: :DISPlay:ZOOM <bool>

:DISPlay:ZOOM?

Dabei ist <bool> vom Typ Boolean und kann entweder {{0|OFF}} oder {1|ON} sein.

CCT

Funktion: Farb-Temperaturanzeige ein- oder ausschalten

Format: :DISPlay:CCT <bool>

:DISPlay:CCT?

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

3.2.8 Messbefehl- -Subsystem

~~MEASURE:OPEN~~

Funktion: Hinzufügen von Messelementen für bestimmte Kanäle auf der Schnittstelle

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

Hinweis: <item> ist diskret und bezieht sich auf den Messpunkt, { 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 PHAS e , diskreter Typ,

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

<src1> bezieht sich auf die Parameter einiger Messgrößen, gültig bei DEL ay , diskreter

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

<src2> bezieht sich auf die Parameter einiger Messgrößen, gültig bei DEL ay , 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:<item>? [<n 1> , <n 2> ,<src1> ,<src2>]~~

Funktion: Abfrage des Werts des offenen Messelements

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

Hinweis: Der diskrete Typ <item> bezieht sich auf den Messwert { PERoid| FREq |

Anstiegszeit| Abfallzeit |VERZÖGERUNG| 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 PHAS e , und zu anderen Zeiten ungültig ist . Es muss nicht geschrieben werden, diskreter Typ,

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

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

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

Beispiel: Wenn die Periodenmessung von Kanal 1 eingeschaltet ist, fragen die folgenden Einstellungen den Periodenmesswert von Kanal 1 ab

:MEASure:PERiod? CH1

~~MEASure:PERiod?~~ ~~CH1~~

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

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

Hinweis: <item> ist diskret und bezieht sich auf den Messpunkt, { PEROid| 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, die gültig ist, wenn DELay und PHASe , diskreter Typ, {CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

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

< src2> bezieht sich auf die Parameter einiger Messgrößen, die gültig sind, wenn DELAY ,
diskreter Typ, [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 <Element>

Dabei ist <element> diskret, { element1| element 2| element 3| element 4| element 5| element 6|
element 7| element 8|element9| element 10| alle }.

1~10 entsprechen den 10 Messoptionen auf dem Bildschirm.

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: Löschen der historischen Statistiken und erneutes Zählen.

Format: :MEASure:STATistic:RESet

3.2.8.5.3 :MEASure:STATistic:MEAN <bool>

Funktion: Ein- oder Ausschalten der Anzeige des Durchschnittswerts in Statistiken

Format: :MEASure:STATistic:MEAN <bool>

:MEASure:STATistic:MEAN?

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

3.2.8.5.4 :MEASure:STATistic:MAX <bool>

Funktion: Ein- oder Ausschalten der Anzeige des Maximalwerts in Statistiken

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: Ein- oder Ausschalten der Anzeige des Minimalwerts in Statistiken

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: Ein- oder Ausschalten der Anzeige des mittleren quadratischen Fehlers in der Statistik

Format: :MEASure:STATistic:DEV <bool>

:MEASure:STATistic:DEV?

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

3.2.8.5.7 :MEASure:STATistic:COUNt <bool>

Funktion: Ein- oder Ausschalten der Zählanzeige 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: Abfrage aller Werte statistischer Elemente (gültig, wenn die Statistikfunktion aktiviert ist)

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

Darunter <item> , das geöffnete Messelement, <source>, diskreter Typ, Messquelle
{CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

Hinweis: <source> kann weggelassen werden, der Standardwert ist die aktuell vom Oszilloskop angegeben.

Rückgabeformat: Die von der Abfrage zurückgegebenen Werte werden in wissenschaftlicher Notation angezeigt, gefolgt von aktuellem Wert, Durchschnittswert, Maximalwert, Minimalwert, quadratischem Mittelwert und Anzahl.

Beispiel: Die folgende Abfrage gibt die statistischen Daten des Spitzen-Spitzen-Messwerts von Kanal 1 zurück, 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 aktuelle Messquelle, die vom Oszilloskop eingestellt wurde, 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 Durchschnittswerts statistischer Elemente (gültig, wenn die Statistikfunktion aktiviert ist)

Format: :MEASure:STATistic:MEAN:VIEW? <Element> , <Quelle>

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

Hinweis: <source> kann weggelassen werden, der Standardwert ist die aktuell vom Oszilloskop angegeben.

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

Beispiel: Die folgende Abfrage gibt die statistische Berechnung des Spitzen-Spitzen-Messwerts von Kanal 1 zurück. Durchschnittswert, z. B.: 1,00000 7 e-02

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

Wenn die vom Oszilloskop eingestellte aktuelle 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 Elemente (gültig, wenn die Statistikfunktion aktiviert ist)

Format: :MEASure:STATistic:MAX:VIEW? <Element> , <Quelle>

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

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

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

Notation.

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

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

Wenn die vom Oszilloskop eingestellte aktuelle 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 Elemente (gültig, wenn die Statistikfunktion aktiviert ist)

Format: :MEASure:STATistic:MIN:VIEW? <Element> , <Quelle>

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

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

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

Beispiel: Die folgende Abfrage gibt die statistische Berechnung des Spitzenwerts zurück.

Spitzenmesswert von Kanal 1. Minimalwert, z. B.: 1,000007e-02

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

Wenn die vom Oszilloskop eingestellte aktuelle 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? <Element>, <Quelle>

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

Hinweis: <source> kann weggelassen werden. Der Standardwert ist die derzeit 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-Messwert von Kanal 1 Mittlerer quadratischer Fehler, z. B. 1,000007 e-02

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

Wenn die Messquelle der aktuellen Oszilloskopeinstellung 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? <Element> , <Quelle>

Darunter <item> , das geöffnete Messelement, <source>, diskreter Typ, Messquelle {CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

Hinweis: <source> kann weggelassen werden. Der Standardwert ist die derzeit vom Oszilloskop angegebene Kanalquelle.

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

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

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

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

:MEASure:STATistic:COUNT:VIEW? PKPK,

3.2.8.5.14 :MEASure:STATistic:CURRENT:VIEW?

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, <source> der diskrete Typ, die Messquelle {CH1|CH2|CH3|CH4|R1|R2|R3|R4|MATH}.

Hinweis: <source> kann weggelassen werden, wobei der Standardwert die derzeit vom Oszilloskop angegebene Kanalquelle ist.

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

Beispiel: Die folgende Abfrage gibt die statistische Berechnung des Spitzen-Spitzen-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,

DISplay

Funktion: Alle Messungen ein- oder ausschalten.

Format: :MEASure:ADISplay <bool>

:MEASure:ADISplay?

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

COUNter

3.2.8.7.1 :MEASure:COUNter:SOURce

Funktion: Einstellen oder Abfragen der Messquelle des Zählers.

:MEASure:COUNter:SOURce <sour>

:MEASure:COUNter:SOURce?

Dabei ist <sour> diskret, {CLOS e| 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. Wenn die Frequenzzählerfunktion derzeit nicht aktiviert ist, wird 0,0000000e+00 zurückgegeben.

3.2.9 Triggerbefehl -Subsystem

TYPE

Funktion: Wählen Sie den Triggertyp aus.

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?

HOLDoff

Funktion: Trigger-Verzögerungszeit einstellen.

Format: :TRIGger:HOLDoff <Wert>

:TRIGger:HOLDoff?

Dabei ist <Wert> eine reelle Zahl zwischen 200 ns und 10 s.

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

Beispiel:

Der folgende Befehl setzt die Trigger-Holdoff-Zeit auf 200 ns.

:TRIGger:HOLDoff 0,0000002

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

:TRIGger:HOLDoff?

MODE

Funktion: Legt den Triggermodus fest: automatisch oder normal.

Format: :TRIGger:MODE <Modus>

:TRIGger:MODE?

Wobei <Modus> diskret ist und { AUTO|NORMal } ist.

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

Beispiel:

Der folgende Befehl wählt den automatischen Triggermodus aus.

:TRIGGER:MODE AUTO

Die folgende Abfrage gibt „AUTO“ zurück.

:TRIGger:MODE?

STATus

Funktion: Abfrage des aktuellen Triggerstatus.

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 Edge-Trigger.

Format: :TRIGger:EDGE:SOURce <Quelle>

:TRIGger:EDGE:SOURce?

Darunter <Quelle>, 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:EDGE:SOURce CH1

Die folgende Abfrage gibt „CH1“ zurück.

```
:TRIGger:EDGE:SOURce?
```

3.2.9.5.2 :TRIGger:EDGE:SLOPe

Funktion: Wählen Sie den Kantentyp des Kantentriggers 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 Triggerung bei steigender Flanke aus.

```
:TRIGger:EDGE:SLOPe RISE
```

Die folgende Abfrage gibt „RISE“ zurück.

```
:TRIGger:EDGE:SLOPe?
```

3.2.9.5.3 :TRIGger:EDGE:LEVel

Funktion: Triggerpegel bei Flankentriggerung einstellen

Format: :TRIGger:EDGE:LEVel <Pegel>

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

Dabei ist <level> ein reeller Typ.

Rückgabeformat: Die Abfrage gibt den Triggerpegelwert 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 Koppelmodus für den Flankentrigger fest.

Format: :TRIGger:EDGE:COUPle <Kopplung>

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

Dabei ist <couple> diskret, { DC|AC|HFRej|LFRej|Noiserej }.

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

Beispiel:

Der folgende Befehl wählt den DC-Kopplungsmodus 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 Impulsbreiten-Trigger fest.

Format: :TRIGger:PULSe:SOURce <Quelle>

:TRIGger:PULSe:SOURce

Dabei ist <Quelle> ein 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 <Polarität> ein diskreter Typ, { POSitiv|NEGativ }.

Rückgabeformat: Die Abfrage gibt „POSitiv “ oder „NEGativ “ zurück.

Beispiel:

Der folgende Befehl wählt die Triggerung 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 während der Impulsbreitenauslösung fest.

Format: :TRIGger:PULSe:WIDTh <Breite>

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

Dabei ist <Breite> ein reeller Typ zwischen 40 ns und 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.

UNGLEICH: Die Impulsbreite des Oszilloskop-Eingangssignals ist nicht gleich der angegebenen Impulsbreite.

3.2.9.6.5 :TRIGger:PULSe:LEVel

Funktion: Legen Sie den Triggerpegel für die Impulsbreitenauslösung fest.

Format: :TRIGger:PULSe:LEVel <Pegel>

:TRIGger:PULSe:LEVel?

Dabei ist <level> ein reeller Typ.

Rückgabeformat: Die Abfrage gibt den Triggerpegelwert 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?
```

LOGic

3.2.9.7.1 :TRIGger:LOGic:STATus

Funktion: Legt den Logikzustand jedes Kanals im Logiktrigger 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 Logik-Triggers fest.

Format: :TRIGger:LOGic:FUNcTion <Funktion>

: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?

Wobei <Bedingung> diskret ist, { GREat|LESS|EQUAL|UNEQUAL|TRUE|FALSE }.

GREat: Wird ausgelöst, wenn der logische Zustand länger als die Trigger-Logikzeit wahr ist.

LESS: Wird ausgelöst, wenn der logische Zustand kürzer als die Trigger-Logikzeit wahr ist;

EQUAL: Wird ausgelöst, wenn der logische Zustand wahr ist und die Haltezeit der Trigger-Logikzeit entspricht;

UNEQUAL: Wird ausgelöst, wenn die Haltezeit des logischen Zustands „wahr“ nicht mit der Auslösezeit übereinstimmt.

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 Trigger-Logikzeit fest.

Format: :TRIGger:LOGic:TIME <Zeit>

:TRIGger:LOGic:TIME?

Dabei ist <Zeit> ein reeller Wert zwischen 200 ns und 10 s.

3.2.9.7.5 :TRIGger:LOGic:LEVel

Funktion: Legt den Triggerpegel jedes Kanals während der Logikauslösung fest.

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

:TRIGger:LOGic:LEVel? <Kanal>

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

Runt

3.2.9.8.1 :TRIGger:Runt:SOURce

Funktion: Legt die Triggerquelle des Runt-Triggers fest.

Format: :TRIGger:Runt:SOURce <Quelle>

:TRIGger:Runt:SOURce?

Wobei <source> diskret ist, {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 <Polarität> 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?

Darunter ist <condition> 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 Obergrenze für die Runt-Triggerzeit fest.

Format: :TRIGger:Runt:HTIME <Zeit>

:TRIGger:Runt:HTIME?

Wobei <Zeit> ein Realwert zwischen 8 ns und 10 s ist.

3.2.9.8.5 :TRIGger:Runt:LTIME

Funktion: Legt die Untergrenze für die Runt-Triggerzeit fest.

Format: :TRIGger:Runt:LTIME <Zeit>

:TRIGger:Runt:LTIME?

Wobei <time> ein reeller Typ im Bereich von 8 ns bis 10 s ist.

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 >, < ltime > reelle Typen im Bereich von 8 ns bis 10 s. (hoch>niedrig)

< 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:HLEVEI?

Dabei ist <level> ein Real-Typ.

3.2.9.8.8 :TRIGger:Runt:LLEVEI

Funktion: Legt den niedrigen Pegel fest, wenn der Runt ausgelöst wird.

Format: :TRIGger:Runt:LLEVEI <Pegel>

:TRIGger:Runt:LLEVEI?

Dabei ist <Pegel> ein Realwert. (H LEV el > LLEV el)

SLOPe

3.2.9.9.1 :TRIGger:SLOPe:SOURce

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

Format: :TRIGger:SLOPe:SOURce <Quelle>

:TRIGger:SLOPe:SOURce?

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

3.2.9.9.2 :TRIGger:SLOPe:EDGE

Funktion: Legt die Flankentriggerung für den Slope-Trigger fest.

Format: :TRIGger:SLOPe:EDGE <Kante>

:TRIGger:SLOPe:EDGE?

Wobei <edge> diskret ist, { RISE|FALL|EITHer }.

3.2.9.9.3 :TRIGger:SLOPe:CONDition

Funktion: Legt die Grenzbedingungen für die Neigungsauslösung 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 die angegebene Zeiteinstellung.

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

3.2.9.9.4 :TRIGger:SLOPe:HTIME

Funktion: Legt die Obergrenze für die Steigungsauslösezeit fest.

Format: :TRIGger:SLOPe:HTIME <Zeit>

:TRIGger:SLOPe:HTIME?

Dabei ist <time> ein Realwert zwischen 8 ns und 10 s.

3.2.9.9.5 :TRIGger:SLOPe:LTIME

Funktion: Legt die Untergrenze für die Slope-Triggerzeit fest.

Format: :TRIGger:SLOPe:LTIMe <Zeit>

:TRIGger:SLOPe:LTIME?

Dabei ist <Zeit> ein Realwert zwischen 8 ns und 10 s.

3.2.9.9.6 :TRIGger:SLOPe:HLEVel

Funktion: Legt den hohen Pegel fest, wenn die Steigung ausgelöst wird.

Format: :TRIGger:SLOPe:HLEVel <Pegel>

:TRIGger:SLOPe:HLEVel?

Dabei ist <level> ein reeller Typ.

3.2.9.9.7 :TRIGger:SLOPe:LLEVel

Funktion: Legt den niedrigen Pegel fest, wenn die Steigung ausgelöst wird.

Format: :TRIGger:SLOPe:LLEVel <Pegel>

:TRIGger:SLOPe:LLEVel?

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

Timeout

3.2.9.10.1 :TRIGger:TIMeout:SOURce

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

Format: :TRIGger:TIMeout:SOURce <Quelle>

:TRIGger:TIMEout:SOURce?

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

3.2.9.10.2 :TRIGger:TIMEout:POLarity

Funktion: Legt die Polarität des Zeitüberschreitungs-Triggers fest.

Format: :TRIGger:TIMEout:POLarity <Polarität>

:TRIGger:TIMEout:POLarity?

Darunter < Polarität >, diskreter Typ, { POSitiv|NEGativ|WEDER POSitiv NOCH NEGativ }.

3.2.9.10.3 :TRIGger:TIMEout:TIME

Funktion: Legt die Zeitüberschreitungsdauer für den Zeitüberschreitungsauflöser fest.

Format: :TRIGger:TIMEout:TIME <Zeit>

:TRIGger:TIMEout:TIME?

Darunter <time>, Real-Typ, 8 ns bis 10 s.

3.2.9.10.4 :TRIGger:TIMEout:LEVel

Funktion: Legt den Übersteuerungswert für den Zeitüberschreitungstrigger fest.

Format: :TRIGger:TIMEout:LEVel <Level>

:TRIGger:TIMEout:LEVel?

Dabei beziehen sich <level>, real type und range auf das Datenblatt.

NEDGE

3.2.9.11.1 :TRIGger:NEDGE:SOURce

Funktion: Legt die Triggerquelle des N-ten Flankentriggers fest.

Format: :TRIGger:NEDGE:SOURce <Quelle>

:TRIGger:NEDGE:SOURce?

Dabei ist <Quelle> 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?

Darunter <slope>, diskreter Typ, {RISE|FALL}.

3.2.9.11.3 :TRIGger:NEDGE:IDLE

Funktion: Legt die Leerlaufzeit vor Beginn der Flankenzählung im N-ten Flankentrigger fest.

Format: :TRIGger:NEDGE:IDLE <Zeit>

:TRIGger:NEDGE:IDLE?

Wobei <time> ein reeller Typ im Bereich von 8 ns bis 10 s ist.

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 Auslöseschwellenwert fest, wenn die N-te Flanke ausgelöst wird.

Format: :TRIGger:NEDGE:LEVel <Pegel>

:TRIGger:NEDGE:LEVel?

Darunter ist <level> vom Typ Real.

3.2.9.12 Video

3.2.9.12.1 :TRIGger:VIDeo:SOURce

Funktion: Legt die Triggerquelle für den Videotrigger fest.

Format: :TRIGger:VIDeo:SOURce <Quelle>

:TRIGger:VIDeo:SOURce?

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

3.2.9.12.2 :TRIGger:VIDeo:POLarity

Funktion: Polarität der Videotriggerung einstellen

Format: :TRIGger:VIDeo:POLarity <Polarität>

:TRIGger:VIDeo:POLarity?

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

3.2.9.12.3 :TRIGger:VIDeo:STANdard

Funktion: Wählen Sie den Videostandard für die Videoauslösung aus.

Format: :TRIGger:VIDeo:STANdard <Standard>

:TRIGger:VIDeo:STANdard?

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

3.2.9.12.4 :TRIGger:VIDeo:MODE

Funktion: Wählen Sie den Synchronisationstyp für die Videoauslösung, wenn der Auslösestandard PAL, SECAm, NTSC oder 1080I ist.

Format: :TRIGger:VIDeo:MODE <Modus>

:TRIGger:VIDeo:MODE?

Darunter <Modus>, diskreter Typ, { ODDField|EVENfield|A LL Field|ALLLINE|LINE }.

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

3.2.9.12.5 :TRIGger:VIDEO:FREQuence

Funktion: Wählen Sie die Signalfrequenz des Videotriggers, wenn der Triggerstandard 1080P ist.

Format: :TRIGger:VIDeo:FREQuence <Frequenz>

:TRIGger:VIDeo:FR EQUence?

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

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?

Unter diesen ist <line> vom Typ real, 1~n, je nach Videotyp ist der Maximalwert von n unterschiedlich.

! bustriggercommand

Vor der Ausführung dieses Befehls ist Folgendes zu beachten

- A、 Öffnen Sie den Kanal von S1 oder S2.
- B、 Kanaleinstellungen für S1 oder S2 vornehmen
- C、 Auslösertyp ist S1 oder S2
- D、 Die folgenden Befehle sollten mit dem eingestellten Typ S1 oder S2 verglichen werden.

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

TYPE

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

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

: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 der Buskonfiguration auf 9 Bit eingestellt ist, kann der Triggertyp DATA nicht eingestellt werden.

Wenn die Buswortlänge in der Buskonfiguration auf 5 Bit, 6 Bit, 7 Bit oder 8 Bit eingestellt ist, können 0:DATA, 1:DATA oder X:DATA im Triggertyp nicht eingestellt werden.

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

Beispiel:

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

: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, X:DATA ausgewählt ist, legen Sie die UART-Bus-Triggerbeziehung fest.

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

:TRIGger:UART:RELAtion? < s >

Unter ihnen ist <s> ein diskreter Typ, { S1|S2 }; <RELATION> ist ein diskreter Typ, { GREAT|LESS|EQUAL|UNEQUAL }.

GREAT: Die Oszilloskop-Eingangsdaten sind größer als die angegebenen Triggerdaten ;

LESS: Die Oszilloskop-Eingangsdaten sind kleiner als die angegebenen Triggerdaten;

EQUAL: Die Oszilloskop-Eingangsdaten sind gleich den angegebenen Triggerdaten;

UNEQUAL: Die Oszilloskop-Eingangsdaten sind nicht gleich den angegebenen Triggerdaten.

Rückgabeformat: Die Abfrage gibt „GREAT“, „LESS“, „EQUAL“ und „UNEQUAL“ zurück.

Beispiel:

Der folgende Befehl setzt die Triggerbeziehung des Kanals S1 UART auf GREAT.

:TRIGger:UART:RELAtion S1, GREAT

Die folgende Abfrage gibt „GREAT“ zurück.

:TRIGger:UART:RELAtion? S1

DATA

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

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 Hexadezimalwerte von 0 bis FF zurück.

Beispiel:

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

:TRIGger:UART:DATA S1, AA

Die folgende Abfrage gibt „AA“ zurück.

:TRIGger:UART:DATA? S1

1 :TRIGger:LIN

TYPE

Funktion: Legt die Auslösebedingung 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“ „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, stellen Sie den Trigger-ID-Wert des LIN-Triggers ein.

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

:TRIGger:LIN:ID?

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

3F.

Rückgabeformat: Die Abfrage gibt Hexadezimalwerte 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

DATA

Funktion: Wenn die LIN-Bus-Triggerbedingung IDATa ist, stellen Sie die Triggerdaten des LIN-Triggers ein.

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

:TRIGger:LIN:DATA?

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

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

Beispiel:

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

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

Die folgende Abfrage gibt „0A“ zurück.

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

3 **:TRIGGER:CAN**

TYPE

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 , Frame-Start; RFID, Remote-Frame-ID; DFID, Daten-Frame-ID; RDID, Remote-Frame-/Daten-Frame-ID; IDATa , Daten-Frame-ID und Daten; WRFR, Fehler-Frame; AERRor , alle Fehler; ACKError , Bestätigungsfehler; OVERload , Überlastungs-Frame.



Funktion: Wenn die Auslösebedingung des CAN-Triggers RFID, DFID, IDATA oder RDID ist, legen Sie den Trigger-ID-Wert des CAN-Triggers fest.

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

:TRIGger:CAN:ID?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <data> ein ganzzahliger Typ, 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?

Darunter: <s>, diskreter Typ, {S1|S2}; <data>, ganzzahliger Typ, 0 bis 8 , 1 2 , 1 6 , 2 0 , 2 6 , 3 2 , 4 8 , 6 4 .



Funktion: Wenn die Auslösebedingung des CAN-Triggers IDATA ist, setzen Sie den Triggerdatenwert des CAN-Triggers.

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

:TRIGger:CAN:DATA?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <data> ist ein ganzzahliger Typ, hexadezimal, die Anzahl der Datenstellen wird durch DLC bestimmt.

3 :TRIGger:SPI

DATA

Funktion: Legt den Datenwert unter SPI-Trigger fest.

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

:TRIGger:SPI:DATA?

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

TYPE

Funktion: Legt den Datenwert unter SPI-Trigger fest.

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

:TRIGger:SPI:TYPE?

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

{CS|DATA|X:DATA}

3 :TRIGger:IIC

TYPE

Funktion: Legt den Triggertyp des IIC-Triggers fest.

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

:TRIGger:IIC:TYPE?

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

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

ADDRess

Funktion: Wenn die IIC-Triggerbedingung NACKaddress, FRAM1, FRAM2 oder WRITE10 ist, legen Sie die Triggeradresse des IIC-Bus-Triggers fest.

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).

RELation

Funktion: Wenn die IIC-Triggerbedingung RDATa ist, legen Sie die Triggerbeziehung des IIC-Bus-Triggers fest.

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

:TRIGger:IIC:RELation

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

GREAT: Die Oszilloskop-Eingangsdaten sind größer als die angegebenen Triggerdaten.

LESS: Die Oszilloskop-Eingangsdaten sind kleiner als die angegebenen Triggerdaten.

EQUAL: Die Oszilloskop-Eingangsdaten sind gleich den angegebenen Triggerdaten.

UNEQual: Die Oszilloskop-Eingangsdaten sind nicht gleich den angegebenen Triggerdaten.

DATA

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

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

:TRIGger:IIC:DATA?

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

DATA2

Funktion: Wenn die IIC-Triggerbedingung FRAM2 ist, werden die durch den IIC-Bus ausgelösten Triggerdaten gesetzt.

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

:TRIGger:IIC:DATA2?

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

TRIGger:1553B

TYPE

Funktion: Legen Sie die Auslösebedingung 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 , Befehls-/Statuswort-Synchronisationsheader; DWSYnc , Datenwort-Synchronisationsheader; CSWOrd , Befehls-/Statuswort; DWORD , Datenwort; RTAddress , Remote-Terminal-Adresse; OPERror , Fehler mit ungerader Parität; MERRor , Manchester-Code-Fehler; AERRor , alle Fehler.

CSWOrd

Funktion: Wenn die 1553B-Triggerbedingung CSWOrd ist, setzen Sie den Befehls-/Statuswortwert, der durch den 1553B-Bus ausgelöst wird.

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

:TRIGger:1553B:CSWOrd?

Darunter: <s>, diskreter Typ, {S1|S2}; <data>, ganzzahliger Typ, 0 bis FFFF.

DWORD

Funktion: Wenn die 1553B-Triggerbedingung DWORd ist, setzen Sie den Triggerdatenwert des 1553B-Bus-Triggers.

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

:TRIGger:1553B:DWORd?

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

RTAddress

Funktion: Wenn die Triggerbedingung für den 1553B-Bus RTAddress ist, stellen Sie die durch den 1553B-Bus ausgelöste Adresse des Remote-Terminals ein.

Format: :TRIGger:1553B:RTAddress < s >, <Adresse>

:TRIGger:1553B:RTAddress?

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

429 :TRIGger:429

TYPE

Funktion: Legt die Auslösebedingung 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, Wortbeginn; WEND, Wortende; LSDI, LABEL+SDI; LDATa , LABEL+DATA; LSSM, LABEL+SSM; WERRor , Wortfehler; WINTerval , Wortlückenfehler; VERRor , Prüfsummenfehler; AERRor , alle Fehler ; ALL0, alle 0-Bits; ALL1, alle 1-Bits.

Nachdem Sie LSDI, LDATa und LSSM ausgewählt haben, müssen Sie die zugehörigen Parameter einstellen. 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

WBEGin

Funktion: Wenn die Auslösebedingung für den 429-Bus „WORD start“ lautet, legen Sie den Auslösewortwert für den 429-Bus-Trigger fest.

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

:TRIGger:429:WBEG in?

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

WEND

Funktion: Wenn die Triggerbedingung für den 429-Bus „WORD ended“ lautet, legen Sie den Triggerwortwert des 429-Bus-Triggers fest.

Format: :TRIGger:429:WEND < s >

:TRIGger:429:WEND?

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

LABEL

Funktion: Wenn die Auslösebedingung für den 429-Bus LABEL, LSDI, LDATa oder LSSM ist, setzen Sie den Auslöse-LABEL-Wert des 429-Bus-Auslösers.

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

:TRIGger:429:LABEL?

Dabei ist <s> ein diskreter Typ, {S1|S2}; <data> ist ein ganzzahliger Typ, oktal, 0 bis 3 77 .

SDI

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

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

:TRIGger:429:SDI?

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

SDATA

Funktion: Wenn die 429-Bus-Triggerbedingung DATA oder LDATa ist, setzen Sie den Triggerdatenwert des 429-Bus-Triggers.

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

:TRIGger:429:DATA?

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

SSM

Funktion: Wenn die Auslösebedingung für den 429-Bus SSM oder LSSM ist, stellen Sie den Auslösedatenwert für den 429-Bus-Trigger ein.

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 Zeitbasisbefehl -Subsystem

EXTent

Funktion: Einstellen des horizontalen Zeitbasis-Getriebes.

Format: :TIMEbase:EXTent <extent>

:TIMEbase:EXTent?

Dabei ist <extent> ein Real-Typ. Einheit: S

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

Beispiel:

Der folgende Befehl setzt die horizontale Zeitbasis auf 2 us.

Format: :TIMEbase:EXTent 2.000000e-6

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

:TIMEbase:EXTent?

MODE

Funktion: Legt den Anzeigemodus für die Bildschirmzeitbasis fest. „YT“ oder „XY“.

Format: :TIMEbase:MODE <Modus>

:TIMEbase:MODE?

Wobei <Modus> diskret ist, „YT“ oder „XY“.

SPlay

Funktion: Schaltet den ROLL-Modus ein oder aus (Zeitbasis über 100 ms).

Format: :TIMebase:ROLL:DISPlay <bool>

:TIMebase:ROLL:DISPlay?

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

Hinweis: Wenn Sie im Rollmodus arbeiten müssen, müssen Sie nach dem Einschalten 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 > ein Real-Typ.

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

Beispiel:

Der folgende Befehl setzt den horizontalen Offset auf 2 us.

:TIMebase:POsition 0.000002

Die folgende Abfrage gibt „2.000000e-06“ zurück

:TIMebase:POsition?

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 Typ, {1e-9~1e3}

3.2.11 Speicherbefehl -Subsystem

SAVE

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

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

:STORage:SAVE <Kanal>

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

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

☐ **:STORage:SAVE:LOCAtion**

Format: :STORage:SAVE:LOCAtion <Speicherort>

:STORage:SAVE:LOCAtion?

Darunter < LOCAtion >, diskreter Typ, { LOCAl|UDISk };

☐ **:STORage:SAVE:TYPE**

Format: :STORage:SAVE:TYPE <Typ>

:STORage:SAVE:TYPE?

Darunter <TYPE>, diskreter Typ, {WAV|BIN|CSV};

☐ **:STORage:SAVE:FILEname**

Format: :STORage:SAVE:FILEname <Dateiname>

:STORage:SAVE:FILEname?

Wobei < Dateiname>:= ASCII-Zeichenkette in Anführungszeichen

☐ **ALLSegments**

Funktion: Bei segmentierter Speicherung alle Segmente speichern

Format: :STORage:SAVE:ALLSegments < bool >

:STORage:SAVE:ALLSegments?

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

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

Beispiel:

Der folgende Befehl aktiviert die Speicherung aller Segmente im Falle einer segmentierten Speicherung.

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

Die folgende Abfrage gibt „1“ zurück.

:STORage:SAVE:ALL elements?

Hinweis: Dieser Befehl ist nur gültig, wenn die segmentierte Speicherung aktiviert ist. Nach Aktivierung dieses Befehls kann der Speichertyp:STORage:SAVE:TYPE <Typ> nur BIN auswählen.

¶ **:STORage:SAVE:START**

Format: :STORage:SAVE:START

Speichern starten.

LOAD

Funktion: Referenz laden.

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

Darunter <Quelle>, diskreter Typ, {R1| R2| R3| R4 }; <Dateiname>, diskreter Typ, der geladene Name, <bool>, Boolescher Typ, {{0|OFF}}{1|ON}}.

3.2.11.3 CAPTURE

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?

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

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

Beispiel:

Der folgende Befehl aktiviert den Zeitstempel des Screenshots.

: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> ein boolescher Typ, {{0|OFF}}{1|ON}}.

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

ONSave

Funktion: Speichert die Oszilloskopeinstellungen.

Definiert den Namen der Oszilloskopeinstellung

Format: :STORage:CONSave:FILEname <Dateiname>

wobei <Dateiname>:= ASCII-Zeichenkette in Anführungszeichen

Format: :STORage:CONSave:START

Speichern starten.

3.2.12

Funktion: Ruft die Oszilloskopeinstellungen mit dem entsprechenden Namen auf.

3.2.12 Buskonfigurations- -Befehlssystem

1

3.2.12.1.1 :BUS <S>:DISPlay

Funktion: Dekodierungskanal umschalten.

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

:BUS <S>:DISPlay?

Dabei ist <n> ein diskreter Typ, { 1|2|3|4} ; <bool> ein boolescher Typ, {{0| OFF}|
{1|ON}}.

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

Beispiel:

Der folgende Befehl schaltet die Anzeige des Decodierungskanals 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> ist ein diskreter Typ, {UART|LIN|SPI|CAN|IIC|1553B| 429}.

3.2.12.1.3 :BUS <S>:MODE <Modus>

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>

Unter diesen ist <channel> ein diskreter Typ, {CH1|CH2|CH3|CH4}; <level> ist ein reeller 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, legen Sie den oberen Schwellenwert des Busses fest.

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}; <Pegel> ist ein reeller Typ.

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

1 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 Leerlaufpegelstatus der UART-Buskonfiguration fest.

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

:BUS <s>:UART:IDLElvl?

Darunter <Zustand> , diskreter Typ, { hoch|niedrig } .

3.2.12.2.3 :BUS <s>:UART:BAUDrate

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

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

:BUS <s>:UART:BAUDrate?

Darunter < baudrate > , diskreter Typ,
{1200|2400|4800|9600|19200|38400|43000|56000|57600|115200} .

3.2.12.2.4 :BUS <s>:UART:CHECK

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

Einheit: b/s

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

:BUS <s>:UART:USERbaud?

Dabei ist <baudrate> eine ganze Zahl zwischen 1200 und 8000000 .

3.2.12.2.6 :BUS <s>:UART:WIDTH

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

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

:BUS <s>:UART:WIDTH?

Wobei <width> eine diskrete Zahl ist, {5|6|7|8|9}.

3.2.12.2.7 :BUS <s>:UART:DISPlay

Funktion: Wählen Sie den Datenanzeigemodus bei der Konfiguration des UART-Busses.

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

:BUS <s>:UART:DISPlay?

Unter diesen ist < display > diskret, {HEX|BIN|ASCII } .

~~2~~BUS LIN

3.2.12.3.1 :BUS <S>:LIN:CHANnel

Funktion: Wählen Sie die Kanalquelle für die LIN-Buskonfiguration 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 Leerlaufpegelstatus der LIN-Buskonfiguration 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: Wählen Sie die Baudrate der LIN-Buskonfiguration aus. Einheit: b/s.

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

:BUS <S>:LIN:BAUDrate?

Wobei < Baudrate > diskret ist, {2400|9600|19200} .

3.2.12.3.4 :BUS <S>:LIN:USERbaud

Funktion: Wählen Sie die benutzerdefinierte Baudrate bei der Konfiguration des LIN-Busses.

Einheit: b/s

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

:BUS <S>:LIN:USERbaud?

Dabei ist <baudrate> eine ganze Zahl zwischen 2400 und 625000 .

~~BUS~~SPI

3.2.12.4.1 :BUS <S>:SPI:CLK

Funktion: Wählen Sie die Taktquelle für die SPI-Buskonfiguration aus.

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

:BUS <S>:SPI:CLK?

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

3.2.12.4.2 :BUS <S>:SPI:DATA

Funktion: Wählen Sie die Datenquelle für die SPI-Buskonfiguration aus.

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

:BUS <S>:SPI:DATA?

Wobei <Kanal> ein diskreter Typ ist, {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?

Wobei <Breite> diskret ist und {4|8|16|24|32} sein kann.

3.2.12.4.4 :BUS <S>:SPI:IDLElvl

Funktion: Wählen Sie den Leerlaufpegelzustand der SPI-Buskonfiguration aus.

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

:BUS <S>:SPI:IDLElvl?

Wobei <Zustand> diskret ist, { hoch|niedrig } .

3.2.12.4.5 :BUS <S>:SPI:SLOPe

Funktion: Wählen Sie den Taktflankentyp für die SPI-Buskonfiguration aus.

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

:BUS <S>:SPI:SLOPe?

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

3.2.12.4.6 :BUS <S>:SPI:CS

Funktion: CS-Freigabe in SPI festlegen und abfragen

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

:BUS <S>:SPI:CS?

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

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: Festlegen und Abfragen der Quelle von CS in SPI

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: Wählen Sie den Leerlaufzustand von CS in der SPI-Buskonfiguration aus.

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

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

Wobei <Zustand> diskret ist, {HIGH|LOW}.

Rückgabeformat: 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:IDLEI?

~~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:Kanal?

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

3.2.12.5.2 :BUS <S>:CAN:SIGNAL

Funktion: Legt den Leerlaufpegelzustand 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: Wählen Sie die Baudrate der CAN-Bus-Konfiguration aus. Einheit: b/s.

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

:BUS <S>:CAN:BAUDrate?

Wobei < baudrate > diskret ist,

{|50000|100000|250000|500000|800000|1000000} .

3.2.12.5.4 :BUS <S>:CAN:USERbaud

Funktion: Auswahl der 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 10000 und 1000000 .

3.2.12.5.5 :BUS <S>:CAN:SAMPlEpoint

Funktion: Auswahl des Abtastpunkts während der CAN-Bus-Konfiguration und des Abtastpunkts des Arbitrierungsfelds während CAN FD, Einheit %

Format: :BUS <S>:CAN:SAMPlEpoint < Prozent >

:BUS <S>:CAN:SAMPlEpoint?

Darunter <Prozent> , Ganzzahl , 1 bis 99 .

3.2.12.5.6 :BUS <S>:CAN:FDBAud rate

Funktion: Wählen Sie die Baudrate des CAN-FD-Buskonfigurationsdatenbits aus.

Einheit: b/s .

Format: :BUS <S>:CAN:FDBAud rate < Baudrate >

:BUS <S>:CAN:FDBAud rate?

Wobei < Baudrate > diskret ist, {NONE|2M|5M}

3.2.12.5.7 :BUS <S>:CAN:FDUSERbaud

Funktion: Wählen Sie bei der Konfiguration des CANFD-Busses die benutzerdefinierte Baudrate des Datenbits aus. Einheit: b/s

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

:BUS <S>:CAN:FDUSERbaud?

Wobei <baudrate> eine ganze Zahl zwischen 1.000.000 und 12.000.000 ist.

3.2.12.5.8 :BUS <S>:CAN:FDSAmappoint

Funktion: Auswahl des Abtastpunkts des CANF-Datenfelds, Einheit: %

Format: :BUS <S>:CAN:FDSA mplepoint < Prozent >

:BUS <S>:CAN:FDSAmplepoint?

Wobei < Prozent > eine ganze Zahl zwischen 1 und 99 ist.

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 der IIC-Buskonfiguration fest.

Format: :BUS <S>:IIC:SDA <Kanal>

:BUS <S>:IIC:SDA?

Darunter <Kanal> , 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?

Wobei <Kanal> ein diskreter Typ ist, {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

3.2.12.8.1 :BUS <S>:429:SOURce

Funktion: Legt die Kanalquelle der 429-Buskonfiguration fest.

Format: :BUS <S>:429:SOURce <Kanal>

:BUS <S>:429:SOURce?

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

3.2.12.8.2 :BUS <S>:429:FORMat

Funktion: Legt das Format der 429-Bus-Konfiguration fest.

Format: :BUS <S>:429:FORMat <Format>

:BUS <S>:429:FORMat?

Wenn <Format> diskret ist, { 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-Buskonfiguration 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?

Darunter < Bandrate > , diskreter Typ, {12500|100000} .

3.2.13 Referenzwellenformbefehl -Subsystem

DISPlay

Funktion: REF-Funktion ein- oder ausschalten.

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?
```

~~REFERENCE:ENABLE~~

Funktion: Schaltet den angegebenen Referenzkanal ein oder aus.

Format: :REfERENCE <n>:ENABle <bool>

```
:REfERENCE <n>:ENABle?
```

Dabei ist <n> ein diskreter Typ, { 1|2|3|4} ; <bool> ein boolescher Typ, {{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?
```

~~REFERENCE~~ HSCALE

Funktion: Legt die horizontale Skalierung des Referenzkanals fest.

Format: :REFERENCE <n>:HSCALE <Skala>

:REFERENCE <n>:HSCALE?

Dabei ist <n> ein diskreter Typ, { 1|2|3|4} ; <scale> ein reeller Typ, 1ns~1ks oder 1mHz~1GHz .

Rückgabeformat: Die Abfrage gibt die horizontale Skalierung in wissenschaftlicher Notation zurück.

~~REFERENCE~~ VSCALE

Funktion: Legt die vertikale Skala des Referenzkanals fest.

Format: :REFERENCE <n>:VSCALE <scale>

:REFERENCE <n>:VSCALE?

Darunter: <n>, diskreter Typ, { 1|2|3|4}; <scale>, reeller Typ, 5 mV bis 5 GV.

Rückgabeformat: Die Abfrage gibt den vertikalen Gang in wissenschaftlicher Notation zurück. (Wenn die aufgerufene Referenzwellenform fft ist, ändert sich die horizontale Skalierung des Kanals nicht.

Beispiel:

Der folgende Befehl setzt die vertikale Skala des Referenzkanals 1 auf 2 V.

:REFERENCE1:VSCALE 2

Der folgende Befehl gibt 2,000000e+00 zurück.

:REfERENCE1:VSCale?

REFERENCE

Funktion: Wählen Sie den aktuellen Referenzkanal aus.

Format: :CURRent:REFerence <n>

Dabei ist <n> ein diskreter Typ, { R1|R2|R3|R4} .

REFERENCEVPOSITION

Funktion: Legt den vertikalen Versatz der Wellenformanzeige des angegebenen Referenzkanals fest.

Format: :REFerence <n>:VPOSition <pos>

:REFerence <n>:VPOSition?

Darunter ist <n> ein diskreter Typ, { 1|2|3|4} ; <pos> ist ein reeller Typ.

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

Beispiel:

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

```
:REFerence 1:VPOSition 0,01
```

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

```
:REFerence1:VPOSition?
```

~~REF~~HPOSITION

Funktion: Legt den horizontalen Versatz der Wellenformanzeige fest.

Format: :REFerence:HPOSition <n>,<pos>

:REFerence:HPOSition? <n>

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

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

Beispiel:

Der folgende Befehl setzt den horizontalen Offset R1 auf 2 us.

:REFerence1:HPOSition 0,000002

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

:REFerence1:HPOSition?

~~REF~~SIZE

Abfrage der Abtastrate der Referenzwellenform

Wobei n ein reeller Typ ist {{ 1|2|3|4}}

~~REF~~DEPTH

Abfrage der Speichertiefe der Referenzwellenform

Wobei n ein reeller Typ { 1|2|3|4} ist

3.2.14 AUTO-Einstellung -Subsystem

CHANnel

Funktion: Aktiviert die automatische Einstellung des Kanals zum automatischen Öffnen und Schließen

Format: :AUTO:SET:CHANnel <bool>

:AUTO:SET:CHANnel?

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.

:AUTO:SET:CHANnel ON oder :AUTO:SET:CHANnel1

Die folgende Abfrage gibt „1“ zurück.

:AUTO:SET:CHANnel?

LEVel

Funktion: Effektiver Schwellenwert für das automatische Öffnen und Schließen von Kanälen im Automatikmodus

Format: :AUTO:SET:LEVel <Pegel>

:AUTO:SET:LEVel?

Dabei ist <level> ein Realwert zwischen 0,001 V und 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:LEVel 0.15

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

:AUTO:SET:LEVel?

SOURCE

Funktion: Bei Ausführung von „Autoset“ wird zuerst die Triggerquellenregel ausgewählt, aktuelle Priorität | maximale Wertpriorität

Format: :AUTO:SET:SOURce <Quelle>

:AUTO:SET:SOURce?

Wobei <Quelle> diskret ist und { CURrent|MAX } ist.

Rückgabeformat: Die Abfrage gibt „CURrent“ oder „MAX“ zurück.

Beispiel:

Wenn der folgende Befehl auf „automatisch“ gesetzt ist, hat der maximale Wert der Triggerquelle Vorrang

:AUTO:SET:SOURce MAX

Die folgende Abfrage gibt „MAX“ zurück.

:AUTO:SET:SOURce?

~~RANGE~~

Funktion: Der durch Drücken der Auto-Taste ausgeführte Modus, unterteilt in „autoset“ und „autorange“

Format: :AUTO:RANGE <bool>

:AUTO:RANge?

Wobei <bool> ein boolescher Typ ist, {{0|OFF}}{{1|ON}}, wenn er 0 ist, befindet er sich im AutoSet-Modus, und wenn er 1 ist, befindet er sich im AutoRange-Modus.

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

Beispiel:

Der folgende Befehl aktiviert die automatische Bereichswahl.

:AUTO:RANge ON oder :AUTO:RANge 1

Die folgende Abfrage gibt „1“ zurück.

:AUTO:RANge?

VERTical

Funktion: Gibt an, ob der vertikale Skalierungsfaktor während der automatischen Bereichsanpassung automatisch angepasst wird.

Format: :AUTO:RANge:VERTical <bool>

:AUTO:RANge:VERTical?

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

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

Beispiel:

Der folgende Befehl aktiviert automatisch die vertikale Ausrichtung.

:AUTO:RANge:VERTical ON oder :AUTO:RANge:VERTical 1

Die folgende Abfrage gibt „1“ zurück.

:AUTO:RANge:VERTical?

HORizontal

Funktion: Gibt an, ob die Zeitbasis während des AutoRange-Vorgangs automatisch eingestellt wird.

Format: :AUTO:RANge:HORizontal <bool>

:AUTO:RANge:HORizontal?

Wobei <bool> ein boolescher Typ ist, {{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?

LEVel

Funktion: Gibt an, ob der Auslösewert während des AutoRange-Prozesses automatisch festgelegt wird.

Format: :AUTO:RANge:LEVel <bool>

:AUTO:RANge:LEVel?

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

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 Wellenformbefehl -Subsystem

:WAVeform:SOURce

:WAVeform:MODE

:WAVeform:FORMat

:WAVeform:DATA?

:WAVeform:START

:WAVeform:STOP

:WAVeform:PREAmble?

:WAVeform:XINCrement?

:WAVeform:XORigin?

:WAVeform:XREFerence?

:WAVeform:YINCrement?

:WAVeform:YORigin?

:WAVeform:YREFerence?

1 SOURce

Funktion: Legt die Kanalquelle für das Lesen der Wellenform fest.

Format: :WAVeform:SOURce <Quelle>

:WAVeform:SOURce?

Wobei < Quelle > ein diskreter Typ ist, {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?

MODE

Diese Funktion legt den Modus zum Lesen der Wellenform fest oder fragt ihn ab.

Format: WAVeform:MODE <Modus>

:WAVeform:MODE?

Wobei < Modus > diskret, { NORMAl|MAXimum|RAW }

illustrieren

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, 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?
```

ORMat

Das Format der Daten, die zurückgegeben werden, wenn **die Funktion** die Daten festlegt oder abfragt.

Format :WAVeform:FORMat <Format>

:WAVeform:FORMat?

Unter diesen ist <format> ein diskreter Typ, { WORD| ASCii }

veranschaulicht

WORD: Die Daten eines Punktes belegen 1 6 Bits, zwei Bytes, was die vertikale Wertgröße angibt.

ASCII: Die Daten des Rückgabepunkts werden in wissenschaftlicher Notation angezeigt 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, .

Rückgabeformat

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?

START

Diese Funktion legt die Startposition der gelesenen Daten fest oder fragt sie ab.

Format :WAVEform:START < no >

:WAVEform:START?

Unter diesen ist <no> eine ganze Zahl, deren Wert sich auf den eingestellten FORMat-Typ bezieht

Im NORMAL-Modus: 1 bis zum maximalen horizontalen Pixelwert des
Bildschirmwellenformbereichs (Pixel pro Raster * horizontale Rasternummer)

MAX: 1 bis zur Anzahl der gültigen Punkte auf dem aktuellen Bildschirm

RAW: 1 bis zum Maximalwert der aktuellen Speichertiefe

illustrieren

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 der Start- und Endpunkt jedes Lesevorgangs festgelegt werden.

Rückgabeformat

Die Abfrage gibt einen ganzzahligen Wert zurück.

Beispiel

Der folgende Befehl setzt den Startpunkt auf 500 .

:WAVEform:START 500

Die folgende Abfrage gibt „500“ zurück.

:WAVEform:START?

STOP

Diese Funktion legt die Endposition für das Lesen von Daten fest oder fragt sie ab.

Format :WAVEform:STOP < no >

:WAVEform:STOP?

Dabei bezieht sich der Wert von <no> (Ganzzahl) auf den eingestellten FORMat-Typ

Im NORMAL-Modus: 1 bis zum maximalen horizontalen Pixelwert des
Bildschirmwellenformbereichs (Pixel pro Raster * horizontale Rasternummer)

MAX: 1 bis zur Anzahl der gültigen Punkte auf dem aktuellen Bildschirm

RAW: 1 bis zum Maximalwert der aktuellen Speichertiefe

illustrieren

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 der Startpunkt und der Endpunkt für jeden Lesevorgang 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 setzt den Startpunkt auf 1000 .

:WAVeform:STOP 1000

Die folgende Abfrage gibt „1000“ zurück.

:WAVeform:STOP?



Funktion: Liest schnell alle auf dem Bildschirm angezeigten Wellenformdaten.

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

Datenlängenbeschreiber, der Datenlänge und den Wellenformdaten.

#MdddddddXXXX

In diesem Format ist # die Kennung. M gibt an, dass die folgenden M Ziffern die Gesamtbyte-Länge 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.

Beispielverfahren für die schnelle Erfassung vollständiger Wellenformdaten:

:WAVeform:SOURce CH1

:WAVeform:MODE NORMAl

:WAVeform:DATA:BIN?

Hinweis: WAVeform:DATA:<type>? wird nur bei 12-Bit-Oszilloskopen unterstützt.



Die Funktion liest Wellenformdaten.

Format :WAVeform:DATA?

Dieser Befehl wird von :WAVeform:SOURce, :WAVeform:FORMat, :WAVeform:MODE und anderen Befehlseinstellungen beeinflusst.

illustrieren

Bildschirm Wellenformdaten-Lesevorgang:

S1. :WAV:SOURce CH1

Legt die Quelle für das Lesen fest

S2. :WAV:MODE NORM	Der Wellenformmodus ist NORM
S3. :WAV:Format BYTE	Datenrückgabeformat auf BYTE setzen
:WAV:DATA?	Daten auf dem Bildschirm anzeigen
Auslesen der gespeicherten Wellenformdaten:	
S1. :MENU:STOP	Die gespeicherte Wellenform kann nur im Stoppzustand gelesen werden
S2. :WAV:SOURce CH1	Quelle für das Auslesen einstellen
S3. :WAV:MODE	RAW-Wellenformmodus ist RAW
S4. :WAV:FORMat BYTE	Legt das Datenrückgabeformat auf WORD fest
S5.:WAVeform:START 1	Setzt die Startposition des Lesepunkts auf 1 S6.
:WAVeform:STOP 62500	Legt die Endposition des Lesepunkts auf 62500 fest S7.
:WAV:DATA?	Ruft die Daten im Puffer ab

Zurück

Wenn :WAV:FORM auf WORD gesetzt ist,

Die zurückgegebenen Daten bestehen aus 4 Teilen: Kennung, Beschreibung der Datenlänge,

Datenlänge und Wellenformdaten.

#M ddd ddddXXXX

einer davon # Kennung

M steht für die ersten M Bits der folgenden Daten, die die Gesamtzahl der Bytes der zurückgegebenen Wellenformdaten beschreiben, ausgedrückt als ddd dddd; Das folgende 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 eine Kennung. Die folgenden 9 Bits stehen für die Anzahl der Abtastpunkte der zurückgegebenen Daten. 0 00 00 10 24 sind insgesamt 9 Datenbits, was bedeutet, dass das Datenvolumen 1024 Abtastpunkte umfasst.

Hinweis:

Wenn die Datenmenge im Speicher groß ist, muss der Benutzer sie mehrmals lesen, jedes Mal einen Datenblock lesen und dann die jeweils gelesenen Daten verbinden. Die jeweils gelesene Datenmenge wird durch das festgelegte Datenrückgabeformat (:WAV:FORMat) bestimmt, wie in der

folgender Tabelle dargestellt:

Einstellungsformat	Maximale Datenmenge, die auf einmal gelesen werden kann	Anmerkung
WORD	62.5000 Abtastpunkte	
ASCII	15625	

Beispiel

Die Datenmenge beträgt 220 KB und das Datenrückgabeformat ist auf WORD eingestellt. Da im WORD-Modus jeweils 62.500 Daten gelesen werden und insgesamt 220.000 Daten gelesen werden müssen, sollte der Vorgang viermal wiederholt werden.

S6. :WAVeform:STOP 187500 Legen Sie die Endposition des Lesepunkts auf
187500 fest. S7. :WAV:DATA? Drittes Datensegment abrufen

Viertes Datensegment lesen

S5.:WAVeform:START 187501 Stellt die Startposition des Lesepunkts auf 187501 ein

S6.:WAVeform:STOP 220000 Setzt die Endposition des Lesepunkts auf 220000

S7. :WAV:DATA? Das vierte Datensegment

abrufen Alle Segmente verbinden, um die Daten des gesamten Speichers
zu erhalten.



Befehlsformat :WAVeform:PREamble?

Funktionsbeschreibung Abfrage und Rückgabe aller Wellenformparameter.

Rückgabeformat

Die Abfrage gibt 9 Wellenformparameter zurück, die durch „ , “ getrennt sind:

<Format>,<Typ>,<Anzahl>,<x-Inkrement>,<x-Ursprung>,<x-Referenz>,<y-Inkrement>,<y-Ursprung>,<y-Referenz>

<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 Wiederholungen im durchschnittlichen

Abtastmodus (siehe Befehl :ACQuire:AVERages) und 1 in anderen Modi.

<xincrement>: Die Zeitdifferenz zwischen zwei benachbarten Punkten in X-Richtung. Referenz:

Befehl WAVeform:XINCrement?

<xorigin>: Die Zeit vom Triggerpunkt bis zur „Referenzzeitbasis“ in X-Richtung. Referenz: Befehl

WAVEform:XORigin?

<xreference>: Die Referenzzeitbasis des Datenpunkts in X-Richtung. Referenz: Befehl

:WAVEform:XREFerence?

< yincrement >: Einheitsspannungswert in Y-Richtung. Referenz: Befehl WAVEform:YINCrement?

< yorigin >: Der vertikale Versatz in Y-Richtung relativ zur „vertikalen Referenzposition“ (Referenz: Befehl WAVEform:YREFerence?). Referenz: Befehl WAVEform:YORigin?.

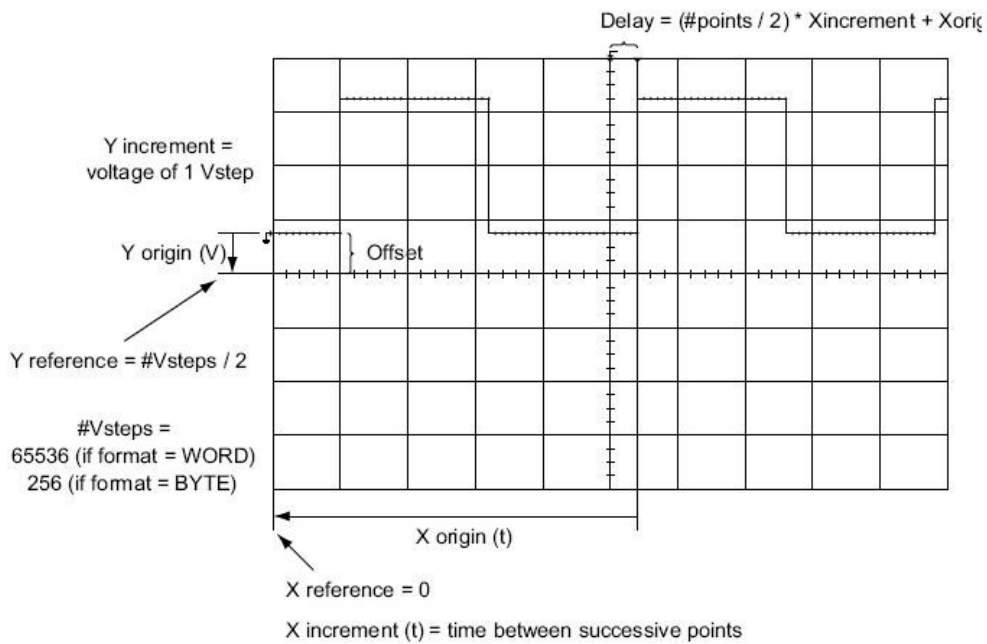
< yreference >: Die vertikale Referenzposition in Y-Richtung. Referenz:

Befehl WAVEform:YREFerence?

Beispiel

Die folgende Abfrage gibt „1,2,1,0,000000,-0,001488,0,0,062500,3,968750,127“ zurück.

:WAVEform:PREamble?



Gerät

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 ist XINCrement = 1/ mathematische Wellenform-Abtastrate. Im

RAW-Modus ist XINCrement = 1/ SampleRate .

Im MAX-Modus gilt bei laufendem Gerät XINCrement = 1/ mathematische Wellenform-

Abtastrate; bei angehaltenem Gerät gilt 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 (Referenz: Befehl 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:

NORMAL-Modus: Gibt die Zeit vom ersten Wellenformpunkt auf dem Bildschirm bis zur Triggerposition zurück.

RAW-Modus: Gibt die Zeit vom ersten Wellenformpunkt im Speicher bis zur Triggerposition zurück.

MAX-Modus: Wenn sich das Gerät im Betriebszustand befindet, gibt es die Zeit vom ersten Wellenformpunkt auf dem Bildschirm bis zur Triggerposition zurück; wenn sich das Gerät im Stillstand befindet, gibt es die Zeit vom ersten Wellenformpunkt im Speicher bis zur Triggerposition zurück.

Rückgabeformat

Die Abfrage gibt den Zeitwert in wissenschaftlicher Notation zurück.

Beispiel

Die folgende Abfrage gibt „-7.000000e-06“ zurück.

:WAVeform:XORigin?

! Wave

Befehlsformat :WAVeform:XREFerence?

Funktionsbeschreibung

Fragt die angegebene Quelle (Referenz: Befehl WAVeform:SOURce) nach der Referenzzeitbasis der Datenpunkte in X-Richtung ab. Die Einheit ist s , unter Verwendung der wissenschaftlichen Notation, wie oben beschrieben.

Rückgabeformat

Die Abfrage gibt die Zeitbasis als Ganzzahl zurück.

Beispiel

Die folgende Abfrage gibt „0“ zurück.

:WAVeform:XREFerence?

! Wave

Format: WAVeform:YINCrement?

Funktionsbeschreibung

Abfrage des Einheitenspannungswerts in Y-Richtung der angegebenen Quelle (Referenz: Befehl WAVeform:SOURce). Die Einheit entspricht der ausgewählten Quelleneinheit.

Rückgabeformat

Die Abfrage gibt den Spannungswert in wissenschaftlicher Notation

zurück. Beispiel

Die folgende Abfrage gibt „3,125000e-03 V“ zurück.

:WAVeform:YINCrement?



Format :WAVeform:YORigin?

Funktionsbeschreibung

Abfrage des vertikalen Versatzes der angegebenen Quelle (Referenz: Befehl WAVeform:SOURce) in Y-Richtung relativ zur „vertikalen Referenzposition“ (Referenz: Befehl 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?



Befehlsformat :WAVeform:YREFerence?

Funktionsbeschreibung

Abfrage der vertikalen Referenzposition der angegebenen Quelle (siehe :WAVeform:SOURce) in Y-Richtung abfragen. Die Einheiten stimmen mit den von der Quelle ausgewählten Einheiten überein.

Rückgabeformat

Die Abfrage gibt die Referenzposition als Ganzzahl zurück. Beispiel

Die folgende Abfrage gibt „127“ zurück.

:WAVEform:YREFerence?

Kontakt

E-Mailsales@micsig.com

Unternehmen: Shenzhen Micsig Technology Co., Ltd.

Adresse: 6F, Jinhuanu-Gebäude, Nr. 56, Tiezai Road, Bezirk Bao'an,
Shenzhen, Guangdong, China.

Dieses Handbuch kann ohne vorherige Ankündigung geändert werden.

Der Inhalt dieses Handbuchs ist nach bestem Wissen und Gewissen korrekt. Sollte der Benutzer Fehler, Auslassungen usw. feststellen, wenden Sie sich bitte an Micsig.

Das Unternehmen haftet nicht für Unfälle und Schäden, die durch unsachgemäße Bedienung durch den Benutzer verursacht werden.

Das Urheberrecht dieses Handbuchs liegt bei Micsig. Keine Einheit und keine Einzelperson darf es ohne die Genehmigung von Micsig reproduzieren, kopieren oder daraus Auszüge erstellen. Micsig behält sich das Recht vor, die oben genannten Handlungen strafrechtlich zu verfolgen.