

Return Format

The query returns the maximum number of frames in integer.

Example

```
:ACQuire:ULTRa:MAXFrame 100      /*Sets the maximum number of frames  
to 100.*/  
:ACQuire:ULTRa:MAXFrame?          /*The query returns 100.*/
```

3.4 :BUS<n> Commands

The :BUS<n> commands are used to execute the decoding-related settings and operations.

3.4.1 :BUS<n>:MODE

Syntax

```
:BUS<n>:MODE <mode>  
:BUS<n>:MODE?
```

Description

Sets or queries the decoding type of the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<mode>	Discrete	{PARallel RS232 SPI IIC LIN CAN}	PARallel

Remarks

Only the DHO900 series supports LIN and CAN decodings.

Return Format

The query returns PAR, RS232, SPI, IIC, LIN, or CAN.

Example

```
:BUS1:MODE SPI      /*Sets the decoding type to SPI.*/  
:BUS1:MODE?          /*The query returns SPI.*/
```

3.4.2 :BUS<n>:DISPLAY

Syntax

```
:BUS<n>:DISPLAY <bool>  
:BUS<n>:DISPLAY?
```

Description

Enables or disables the specified decoding bus; or queries the on/off display status of the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<bool>	Bool	{ {1 ON} {0 OFF} }	0 OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:BUS1:DISPLAY ON      /*Enables the decoding bus.*/
:BUS1:DISPLAY?        /*The query returns 1.*/
```

3.4.3 :BUS<n>:FORMAT

Syntax

```
:BUS<n>:FORMAT <format>
:BUS<n>:FORMAT?
```

Description

Sets or queries the format of decoding data on the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<format>	Discrete	{HEX ASCII DEC BIN}	HEX

Remarks

- **Hex:** indicates Hexadecimal;
- **ASCII:** indicates ASCII;
- **DEC:** indicates Decimal;
- **BIN:** indicates Binary.

Return Format

The query returns HEX, ASC, DEC, or BIN.

Example

```
:BUS1:FORMAT HEX      /*Sets the display format of the bus to HEX.*/
:BUS1:FORMAT?          /*The query returns HEX.*/
```

3.4.4 :BUS<n>:EVENT

Syntax

```
:BUS<n>:EVENT <bool>
```

```
:BUS<n>:EVENT?
```

Description

Enables or disables the event table of the specified decoding bus; or queries the on/off status of the specified decoding bus event table.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<bool>	Bool	{1 ON}{0 OFF}	0 OFF

Remarks

Before using the command, enable the specified decoding bus.

Return Format

The query returns 1 or 0.

Example

```
:BUS1:EVENT ON      /*Enables the event table of the specified
decoding bus.*/
:BUS1:EVENT?          /*The query returns 1.*/
```

3.4.5 :BUS<n>:LABEL

Syntax

```
:BUS<n>:LABEL <bool>
```

```
:BUS<n>:LABEL?
```

Description

Enables or disables the label of the specified decoding bus; or queries the on/off display status of the label of the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<bool>	Bool	{1 ON}{0 OFF}	1 ON

Remarks

Before using the command, enable the specified decoding bus.

Return Format

The query returns 1 or 0.

Example

```
:BUS1:LABEL ON /*Enables the label of the specified decoding bus.*/
:BUS1:LABEL? /*The query returns 1.*/
```

3.4.6 :BUS<n>:DATA?

Syntax

```
:BUS<n> :DATA?
```

Description

Reads the data from the decoding event table.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-

Remarks

N/A

Return Format

The query returns the data in the decoding event table with the following formats.

```
#9000000086PARALLEL
Time,Data,
-2.47us,0,
-2.444us,1,
-1.448us,0,
-446ns,1,
551.6ns,0,
1.554us,1,
```

Wherein, "#9000000086" is the TMC data block header, which is followed by the data in the event table. The 9-digit data following #9 in the data block header indicates the number of bytes of the effective data. "PARALLEL" indicates the decoding type. The

available decoding type can also be RS232, I2C, SPI, LIN, and etc. The data are separated by a comma, and will automatically switch to the next line according to the data information in the decoding list. The data value is related to the numeral system that you have set.



CAUTION

You can save all the data (except TMC data block header and decoding type, e.g. #9000000086PARALLEL) as the ".csv" file and view the data in the form of a list.

Example

N/A

3.4.7 :BUS< n >:EEXPORT

Syntax

:BUS< n >:EEXPORT <path>

Description

Exports the decoding information from the specified decoding bus event table in CSV form.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<path>	ASCII String	Refer to <i>Remarks</i>	-

Remarks

- <path> includes the file storage location and the filename with a suffix. If the specified storage location already contains a file with the same filename, the original file will be overwritten.
- When the operating status of the instrument is STOP (set it by sending the :STOP command), you can export the time and corresponding decoding data from the current event table.
- This command is valid when the display of the event table is enabled. You can enable the display of the event table by sending the :BUS< n >:EVENT command.
- The stored ".csv" file can be opened and edited in Excel.

Return Format

N/A

Example

```
:BUS1:EEXPORT C:/123.csv /*Exports the decoding information from
the bus event table and saves it to the local Disk C, with the
filename 123.csv.*/
```

3.4.8 :BUS<n>:POSITION**Syntax**

```
:BUS<n>:POSITION <pos>
:BUS<n>:POSITION?
```

Description

Sets or queries the vertical position of the bus on the screen.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<pos>	Integer	-250 to 250	0

Remarks

N/A

Return Format

The query returns an integer ranging from -250 to 250.

Example

```
:BUS1:POSITION 200 /*Sets the vertical position of the bus to
200.*/
:BUS1:POSITION? /*The query returns 200.*/
```

3.4.9 :BUS<n>:THRESHOLD**Syntax**

```
:BUS<n>:THRESHOLD <value>,<type>
:BUS<n>:THRESHOLD? <type>
```

Description

Sets or queries the threshold of the specified decoding source on the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<value>	Real	(-5 x VerticalScale - OFFSet) to (5 x VerticalScale - OFFSet)	0
<type>	Discrete	{PAL TX RX SCL SDA CS CLK MISO MOSI LIN CAN PALCLK CH1 CH2 CH3 CH4}	-

Remarks

For VerticalScale, refer to [:CHANnel<n>:SCALe](#). For OFFSet, refer to [:CHANnel<n>:OFFSet](#).

- **PAL:** indicates the bus source of Parallel decoding.
- **PALCLK:** indicates the clock source of Parallel decoding. Only when the source is enabled, can you set the threshold.
- **TX:** indicates the TX channel source of RS232 decoding.
- **RX:** indicates the RX channel source of RS232 decoding. Only when the RX source is enabled, can you set the threshold.
- **SCL:** indicates the clock source of I2C decoding.
- **SDA:** indicates the clock source of I2C decoding.
- **CS:** indicates the source of the CS line of SPI decoding.
- **CLK:** indicates the clock source of SPI decoding.
- **MISO:** indicates the MISO data source of SPI decoding.
- **MOSI:** indicates the MOSI data source of SPI decoding.
- **LIN:** indicates the bus source of LIN decoding.
- **CAN:** indicates the channel source of CAN decoding.
- **CH1|CH2|CH3|CH4:** CH1-CH4, available for all the decoding types.

Only DHO900 series supports LIN and CAN decodings.

For DHO800 series, only 4-channel models support CS setting for SPI decoding.

Return Format

The query returns the threshold of the specified decoding source in scientific notation.

Example

```
:BUS1:THreshold 2.4, PAL      /*Sets the threshold of the Parallel
decoding source to 2.4 v.*/
:BUS1:THreshold? PAL         /*The query returns 2.400000E0.*/
```

3.4.10 :BUS<n>:PARallel

:BUS<n>:PARallel commands are used to set relevant parameters for parallel decoding.

Parallel bus consists of clock line and data line. As shown in the figure below, CLK is the clock line, whereas Bit0 and Bit1 are the 0 bit and 1st bit on the data line respectively. The oscilloscope will sample the channel data on the rising edge, falling edge, or the rising/falling edge of the clock and judge each data point (logic "1" or logic "0") according to the preset threshold level.

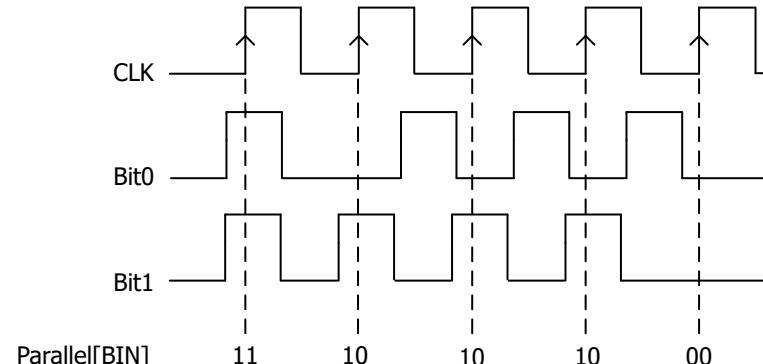


Figure 3.1 Schematic Diagram of Parallel Decoding

3.4.10.1 :BUS<n>:PARallel:BUS

Syntax

```
:BUS</> :PARallel:BUS <source>
:BUS</> :PARallel:BUS?
```

Description

Sets or queries the current source of the Parallel decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D7D0 D15D8 D15D0 D0D7 D8D15 D0D15 CHANnel1 CHANnel2 CHANnel3 CHANnel4 USER}	CHANnel1

Remarks

Only DHO900 series supports the digital channels D0-D15.

Return Format

The query returns D7D0, D15D8 D15D0, D0D7, D8D15, D0D15, CHAN1, CHAN2, CHAN3, CHAN4, or USER.

Example

```
:BUS1:PARallel:BUS CHANnel1 /*Sets the current source of the
Parallel decoding bus to CHANnel1.*/
:BUS1:PARallel:BUS?           /*The query returns CHAN1.*/
```

3.4.10.2 :BUS<n>:PARallel:CLK**Syntax**

```
:BUS<n>:PARallel:CLK <source>
:BUS<n>:PARallel:CLK?
```

Description

Sets or queries the clock source of the Parallel decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANnel1 CHANnel2 CHANnel3 CHANnel4 OFF}	OFF

Remarks

Only DHO900 series supports the digital channels D0-D15.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:BUS1:PARallel:CLK CHANne12 /*Sets the clock source of the
Parallel decoding to CHANne12.*/
:BUS1:PARallel:CLK?           /*The query returns CHAN2.*/
```

3.4.10.3 :BUS<n>:PARallel:SLOPe

Syntax

```
:BUS</n> :PARallel:SLOPe <slope>
:BUS</n> :PARallel:SLOPe?
```

Description

Sets or queries the edge type of the clock channel when being sampled by Parallel decoding on the data channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<slope>	Discrete	{POSitive NEGative BOTH}	POSitive

Remarks

If no clock channel is selected, the instrument will sample when the channel data hopping occurs during the decoding.

Return Format

The query returns POS, NEG, or BOTH.

Example

```
:BUS1:PARallel:SLOPe BOTH    /*Sets the Parallel decoding to
sample on any edge of the clock channel.*/
:BUS1:PARallel:SLOPe?        /*The query returns BOTH.*/
```

3.4.10.4 :BUS<n>:PARallel:WIDTh

Syntax

```
:BUS</n> :PARallel:WIDTh <wid>
:BUS</n> :PARallel:WIDTh?
```

Description

Sets or queries the data width of the Parallel decoding, that is, the number of bits per frame.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<wid>	Integer	1 to 4	1

Remarks

- Only when the bus source is set to User (*:BUS<n>:PARallel:BUS USER*), can this command be valid.
- Send this command to set the data width of the bus first, then send the *:BUS<n>:PARallel:BITX* and *:BUS<n>:PARallel:SOURce* command to select the bit and set the channel source for the bit.

Return Format

The query returns an integer ranging from 1 to 4.

Example

```
:BUS1:PARallel:WIDTh 4      /*Sets the data width of Parallel
decoding to 4.*/
:BUS1:PARallel:WIDTh?      /*The query returns 4.*/
```

3.4.10.5 :BUS<n>:PARallel:BITX

Syntax

```
:BUS<n>:PARallel:BITX <bit>
:BUS<n>:PARallel:BITX?
```

Description

Sets or queries the data bit that the parallel bus requires to set for the channel source.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<bit>	Integer	0 to (data width - 1)	0

Remarks

- Only when the bus source is set to User (*:BUS<n>:PARallel:BUS USER*), can this command be valid.
- The data width is set by the *:BUS<n>:PARallel:WIDTH* command.
- After selecting the desired bit, send the *:BUS<n>:PARallel:SOURce* command to set the channel source for the bit.

Return Format

The query returns the current data bits in integer. Its unit is Hz.

Example

```
:BUS1:PARallel:BITX 2      /*Sets the current bit to 2.*/
:BUS1:PARallel:BITX?      /*The query returns 2.*/
```

3.4.10.6 :BUS<n>:PARallel:SOURce

Syntax

```
:BUS</n> :PARallel:SOURce <src>
:BUS</n> :PARallel:SOURce?
```

Description

Sets or queries the channel source of the currently selected data bit.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<src>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANnel1 CHANnel2 CHANnel3 CHANnel4}	Related to the selected bit

Remarks

- Only when the bus source is set to User (*:BUS<n>:PARallel:BUS USER*), can this command be valid.
- >Before sending this command, send the *:BUS<n>:PARallel:BITX* command to select the desired data bit.
- D0-D15 are only available for the DHO900 series.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:PARallel:SOURce CHANnel2      /*Sets the channel source of the
current bit to CHANnel2.*/
:BUS1:PARallel:SOURce?             /*The query returns CHAN2.*/
```

3.4.10.7 :BUS<n>:PARallel:ENDian

Syntax

```
:BUS <n> :PARallel:ENDian <endian>
:BUS <n> :PARallel:ENDian?
```

Description

Sets or queries the endian of Parallel decoding on the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<endian>	Discrete	{MSB LSB}	MSB

Remarks

- MSB:** indicates Most Significant Bit transmission sequence, i.g. the highest bit of the data is transmitted first.
- LSB:** indicates Least Significant Bit transmission sequence, i.g. the lowest bit of the data is transmitted first.

Return Format

The query returns MSB or LSB.

Example

```
:BUS1:PARallel:ENDian LSB /*Sets the endian of Parallel decoding
to LSB.*/
:BUS1:PARallel:ENDian? /*The query returns LSB.*/
```

3.4.10.8 :BUS<n>:PARallel:POLarity

Syntax

```
:BUS <n> :PARallel:POLarity <pol>
```

:BUS<n>:PARallel:POLarity?

Description

Sets or queries the data polarity of Parallel decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<pol>	Discrete	{NEGative POSitive}	POSitive

Remarks

- NEGative:** indicates negative polarity.
- POSitive:** indicates positive polarity.

Return Format

The query returns NEG or POS.

Example

```
:BUS1:PARallel:POLarity NEGative      /*Sets the data polarity of
Parallel decoding to Negative.*/
:BUS1:PARallel:POLarity?              /*The query returns NEG.*/
```

3.4.11 :BUS<n>:RS232

:BUS<n>:RS232 commands are used to set relevant parameters for RS232 decoding.

RS232 serial bus consists of the transmitting data line (TX) and the receiving data line (RX).

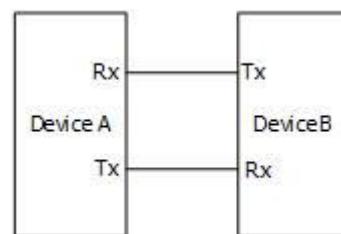
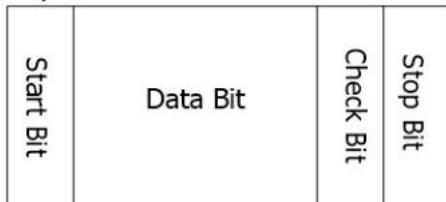


Figure 3.2 Schematic Diagram of RS232 Serial Bus

In RS232, baud rate is used to represent the transmission rate (namely bits per second) of the data. You need to set the start bit, data bits, check bit (optional), and stop bits for each frame of data.



- **Start Bit:** indicates when to output data.
- **Data Bit:** indicates the number of data bits actually contained in each frame of data.
- **Check Bit:** used to check whether the data are properly transmitted.
- **Stop Bit:** indicates when to stop outputting data.

3.4.11.1 :BUS<n>:RS232:TX

Syntax

`:BUS <n> :RS232 :TX <source>`

`:BUS <n> :RS232 :TX?`

Description

Sets or queries the Tx source of RS232 decoding on the specific bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANnel1 CHANnel2 CHANnel3 CHANnel4 OFF}	CHANnel1

Remarks

The Tx and Rx sources cannot be set to OFF at the same time. The Rx source can be set by using the `:BUS <n>:RS232:RX` command.

D0-D15 are only available for the DHO900 series.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:BUS1:RS232:TX CHANnel2    /*Sets the Tx source of RS232 decoding
to CHANel2.*/
:BUS1:RS232:TX?           /*The query returns CHAN2.*/
```

3.4.11.2 :BUS<n>:RS232:RX**Syntax**

```
:BUS <n> :RS232 :RX <source>
:BUS <n> :RS232 :RX?
```

Description

Sets or queries the Rx source of RS232 decoding on the specific bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANnel1 CHANnel2 CHANnel3 CHANnel4 OFF}	OFF

Remarks

The Tx and Rx sources cannot be set to OFF at the same time. The Tx source can be set by using the **:BUS<n>:RS232:TX** command.

D0-D15 are only available for the DHO900 series.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:BUS1:RS232:RX CHANnel2    /*Sets the Rx source of RS232 decoding
to CHANel2.*/
:BUS1:RS232:RX?           /*The query returns CHAN2.*/
```

3.4.11.3 :BUS<n>:RS232:POLarity**Syntax**

```
:BUS <n> :RS232 :POLarity <pol>
:BUS <n> :RS232 :POLarity?
```

Description

Sets or queries the polarity of RS232 decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<pol>	Discrete	{POSitive NEGative}	NEGative

Remarks

- **POSitive:** indicates positive polarity. High level is logic "1" and low level is logic "0".
- **NEGative:** indicates negative polarity. High level is logic "0" and low level is logic "1".

Return Format

The query returns POS or NEG.

Example

```
:BUS1:RS232:POLarity POSitive      /*Sets the polarity of RS232
decoding to Positive.*/
:BUS1:RS232:POLarity?            /*The query returns POS.*/
```

3.4.11.4 :BUS<n>:RS232:PARity

Syntax

```
:BUS<n>:RS232:PARity <parity>
:BUS<n>:RS232:PARity?
```

Description

Sets or queries the odd-even check mode of data transmission in RS232 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<parity>	Discrete	{NONE ODD EVEN}	NONE

Remarks

- **None:** indicates that there is no parity bit in data transmission.

- **ODD:** indicates the odd parity bit. The total count of occurrences of 1 in the data bit and check bit is an odd number. For example, if 0x55 (01010101) is transmitted, 1 shall be added to the check bit.
- **Even:** indicates the even parity bit. The total count of occurrences of 1 in the data bit and check bit is an even number. For example, if 0x55 (01010101) is transmitted, 0 shall be added to the check bit.

Return Format

The query returns NONE, ODD, or EVEN.

Example

```
:BUS1:RS232:PARity ODD      /*Sets the odd-even check mode of
data transmission in RS232 decoding to ODD.*/
:BUS1:RS232:PARity?          /*The query returns ODD.*/
```

3.4.11.5 :BUS<n>:RS232:ENDian

Syntax

```
:BUS<n>:RS232:ENDian < endian>
:BUS<n>:RS232:ENDian?
```

Description

Sets or queries the endian of data transmission in RS232 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<endian>	Discrete	{MSB LSB}	LSB

Remarks

- **LSB:** indicates Least Significant Bit transmission sequence, that is, the lowest bit of the data is transmitted first.
- **MSB:** indicates Most Significant Bit transmission sequence, that is, the highest bit of the data is transmitted first.

Return Format

The query returns LSB or MSB.

Example

```
:BUS1:RS232:ENDian MSB      /*Sets the transmission order of  
RS232 decoding to MSB.*/  
:BUS1:RS232:ENDian?          /*The query returns MSB.*/
```

3.4.11.6 :BUS<n>:RS232:BAUD

Syntax

```
:BUS <n> :RS232 :BAUD <baud>  
:BUS <n> :RS232 :BAUD?
```

Description

Sets or queries the baud rate of data transmission in RS232 decoding. The default unit is bps.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<baud>	Integer	1bps to 20Mbps	9600 bps

Remarks

If the baud rate is set to a value with "M", then "A" should be added at the end of the value. For example, if you send 5M, you need to send 5MA.

Return Format

The query returns an integer ranging from 1 to 20M.

Example

```
:BUS1:RS232:BAUD 4800      /*Sets the baud rate of data  
transmission in RS232 decoding to 4800 bps.*/  
:BUS1:RS232:BAUD?          /*The query returns 4800.*/
```

3.4.11.7 :BUS<n>:RS232:DBITS

Syntax

```
:BUS <n> :RS232 :DBITS <bits>  
:BUS <n> :RS232 :DBITS?
```

Description

Sets or queries the data width of RS232 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<bits>	Discrete	{5 6 7 8 9}	8

Remarks

N/A

Return Format

The query returns 5, 6, 7, 8, or 9.

Example

```
:BUS1:RS232:DBITS 7    /*Sets the data width of RS232 decoding to  
7.*/  
:BUS1:RS232:DBITS?      /*The query returns 7.*/
```

3.4.11.8 :BUS<n>:RS232:SBITS**Syntax**

```
:BUS<n>:RS232:SBITS <stop bits>  
:BUS<n>:RS232:SBITS?
```

Description

Sets or queries the stop bits of each frame of data in RS232 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<stop bits>	Discrete	{1 1.5 2}	1

Remarks

N/A

Return Format

The query returns 1, 1.5, or 2.

Example

```
:BUS1:RS232:SBITS 2    /*Sets the stop bits of RS232 decoding to  
2.*/  
:BUS1:RS232:SBITS?    /*The query returns 2.*/
```

3.4.12 :BUS<n>:IIC

The :BUS<n>:IIC commands are used to set relevant parameters for I2C decoding.

I2C serial bus consists of the clock line (SCL) and the data line (SDA).

- **SCL:** samples SDA on the rising or falling edge of the clock.
- **SDA:** indicates the data channel.

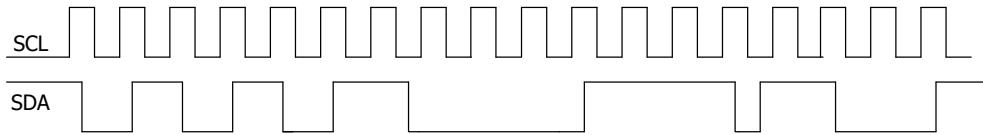


Figure 3.3 I2C Serial Bus

3.4.12.1 :BUS<n>:IIC:SCLK:SOURce

Syntax

```
:BUS<n> :IIC:SCLK:SOURce <source>
```

```
:BUS<n> :IIC:SCLK:SOURce?
```

Description

Sets or queries the clock source of I2C decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Remarks

D0-D15 are only available for the DHO900 series.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:IIC:SCLK:SOURce CHANnel2      /*Sets the clock source of I2C
decoding to CHANnel2.*/
:BUS1:IIC:SCLK:SOURce?            /*The query returns CHAN2.*/
```

3.4.12.2 :BUS<n>:IIC:SDA:SOURce

Syntax

```
:BUS<n> : IIC:SDA:SOURce <source>
:BUS<n> : IIC:SDA:SOURce?
```

Description

Sets or queries the data source of the I2C decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANne1 CHANne2 CHANne3 CHANne4}	CHANne1

Remarks

D0-D15 are only available for the DHO900 series.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:IIC:SDA:SOURce CHANne12 /*Sets the data source of I2C
decoding to CHANne12.*/
:BUS1:IIC:SDA:SOURce? /*The query returns CHAN2.*/
```

3.4.12.3 :BUS<n>:IIC:EXCHange

Syntax

```
:BUS<n> : IIC:EXCHange <bool>
:BUS<n> : IIC:EXCHange?
```

Description

Sets to exchange the clock channel source and the data channel source of I2C decoding on the specified bus; queries whether the clock channel source and the data channel source of I2C decoding on the specified bus have been exchanged.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<bool>	Bool	{1 ON} {0 OFF}	0 OFF

Remarks

N/A

Return Format

The query returns 0 or 1.

Example

```
:BUS1:IIC:EXCHange ON /*Sets to exchange the clock channel source  
and the data channel source.*/
:BUS1:IIC:EXCHange? /*The query returns 1.*/
```

3.4.12.4 :BUS<n>:IIC:ADDReSS**Syntax**

```
:BUS<n>:IIC:ADDReSS <addr>
:BUS<n>:IIC:ADDReSS?
```

Description

Sets or queries the address mode of I2C decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<addr>	Discrete	{NORMAl RW}	NORMAl

Remarks

- NORMAl:** indicates that the address width does not include the R/W bit.
- RW:** indicates that the address width includes the R/W bit.

Return Format

The query returns NORM or RW.

Example

```
:BUS1:IIC:ADDReSS RW /*Sets the address of I2C decoding to  
include the R/W bit.*/
:BUS1:IIC:ADDReSS? /*The query returns RW.*/
```

3.4.13 :BUS<n>:SPI

:BUS<n>:SPI commands are used to set relevant parameters for SPI decoding.

SPI bus is based on the master — slave configuration and usually consists of chip select line (CS), clock line (CLK), and data line (SDA). Wherein, the data lines include the master input/slave output (MISO) data line and master output/slave input (MOSI) data line. The oscilloscope samples the channel data on the rising or falling edge of the clock signal. For analog channels, it also judges each data point (logic "1" or logic "0") according to the preset threshold level.

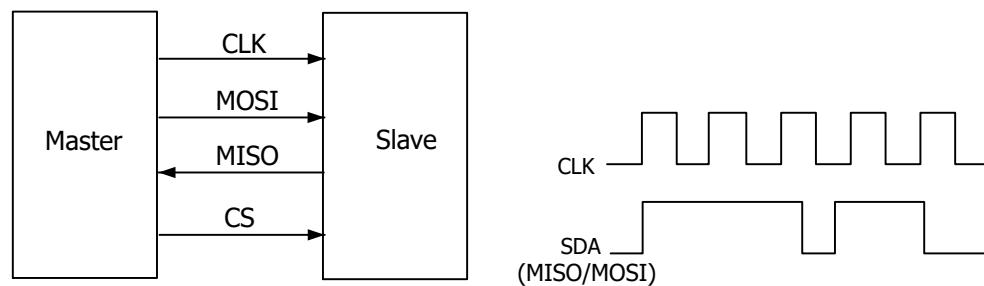


Figure 3.4 SPI Serial Bus

3.4.13.1 :BUS<n>:SPI:SCLK:SOURce

Syntax

```
:BUS<n> :SPI:SCLK:SOURce <source>
:BUS<n> :SPI:SCLK:SOURce?
```

Description

Sets or queries the clock source of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Remarks

D0-D15 are only available for the DHO900 series.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:SPI:SCLK:SOURce CHANnel2      /*Sets the clock source of SPI
decoding to CHANel2.*/
:BUS1:SPI:SCLK:SOURce?            /*The query returns CHAN2.*/
```

3.4.13.2 :BUS<n>:SPI:SCLK:SLOPe

Syntax

```
:BUS<n>:SPI:SCLK:SLOPe <slope>
:BUS<n>:SPI:SCLK:SLOPe?
```

Description

Sets or queries the clock edge type of the SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<slope>	Discrete	{POSitive NEGative}	POSitive

Remarks

N/A

Return Format

The query returns POS or NEG.

Example

```
:BUS1:SPI:SCLK:SLOPe NEGative      /*Sets the clock edge type of SPI
decoding to Negative.*/
:BUS1:SPI:SCLK:SLOPe?            /*The query returns NEG.*/
```

3.4.13.3 :BUS<n>:SPI:MISO:SOURce

Syntax

```
:BUS<n>:SPI:MISO:SOURce <SOURCE>
:BUS<n>:SPI:MISO:SOURce?
```

Description

Sets or queries the MISO data source of SPI decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANnel1 CHANnel2 CHANnel3 CHANnel4 OFF}	CHANnel2

Remarks

The source specified in this command and the `:BUS<n>:SPI:MOSI:SOURce` command cannot be set to OFF at the same time.

D0-D15 are only available for the DHO900 series.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:BUS1:SPI:MISO:SOURce CHANnel2      /*Sets the MISO data source of
SPI decoding to CHANnel2.*/
:BUS1:SPI:MISO:SOURce?            /*The query returns CHAN2.*/
```

3.4.13.4 :BUS<n>:SPI:MOSI:SOURce**Syntax**

```
:BUS<n>:SPI:MOSI:SOURce <source>
:BUS<n>:SPI:MOSI:SOURce?
```

Description

Sets or queries the MOSI data source of SPI decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANnel1 CHANnel2 CHANnel3 CHANnel4 OFF}	OFF

Remarks

The source specified in this command and the `:BUS<n>:SPI:MISO:SOURce` command cannot be set to OFF at the same time.

D0-D15 are only available for the DHO900 series.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:BUS1:SPI:MOStI:SOURce CHANnel2      /*Sets the MOSI data source of
SPI decoding to CHANNEL2.*/
:BUS1:SPI:MOStI:SOURce?            /*The query returns CHANNEL2.*/
```

3.4.13.5 :BUS<n>:SPI:POLarity

Syntax

```
:BUS<n>:SPI:POLarity <polarity>
:BUS<n>:SPI:POLarity?
```

Description

Sets or queries the polarity of the SPI decoding on the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<polarity>	Discrete	{HIGH LOW}	HIGH

Remarks

- **HIGH:** indicates positive polarity. The high level is 1, and low level is 0.
- **LOW:** indicates negative polarity. The high level is 0, and low level is 1.

Return Format

The query returns HIGH or LOW.

Example

```
:BUS1:SPI:POLarity HIGH      /*Sets the polarity of the SPI
decoding to Positive.*/
:BUS1:SPI:POLarity?        /*The query returns HIGH.*/
```

3.4.13.6 :BUS<n>:SPI:MISO:POLarity

Syntax

```
:BUS<n> :SPI:MISO:POLarity <polarity>
:BUS<n> :SPI:MISO:POLarity?
```

Description

Sets or queries the polarity of MISO data line of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<polarity>	Discrete	{HIGH LOW}	HIGH

Remarks

- **HIGH:** positive polarity. It indicates that high level is 1, and low level is 0.
- **LOW:** negative polarity. It indicates that low level is 1, and high level is 0.

This command exists for backwards compatibility. Use the command [:BUS<n>:SPI:POLarity](#).

Return Format

The query returns HIGH or LOW.

Example

```
:BUS1:SPI:MISO:POLarity HIGH      /*Sets the polarity of MISO data
line to Positive.*/
:BUS1:SPI:MISO:POLarity?          /*The query returns HIGH.*/
```

3.4.13.7 :BUS<n>:SPI:MOSt:POLarity

Syntax

```
:BUS<n> :SPI:MOSt:POLarity <polarity>
:BUS<n> :SPI:MOSt:POLarity?
```

Description

Sets or queries the polarity of MOSI data line of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-

Name	Type	Range	Default
<polarity>	Discrete	{HIGH LOW}	HIGH

Remarks

- **HIGH:** positive polarity. It indicates that high level is 1, and low level is 0.
- **LOW:** negative polarity. It indicates that low level is 1, and high level is 0.

Return Format

The query returns HIGH or LOW.

Example

```
:BUS1:SPI:莫斯I:POLarity HIGH      /*Sets the polarity of MOSI data
line to Positive.*/
:BUS1:SPI:莫斯I:POLarity?          /*The query returns HIGH.*/
```

3.4.13.8 :BUS<n>:SPI:DBITs

Syntax

```
:BUS<n>:SPI:DBITs <width>
:BUS<n>:SPI:DBITs?
```

Description

Sets or queries the data width of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<width>	Integer	4 to 32	8

Remarks

N/A

Return Format

The query returns an integer ranging from 4 to 32.

Example

```
:BUS1:SPI:DBITs 10      /*Sets the data width of SPI decoding to
10.*/
:BUS1:SPI:DBITs?        /*The query returns 10.*/
```

3.4.13.9 :BUS<n>:SPI:ENDian

Syntax

```
:BUS<n> :SPI:ENDian <endian>
:BUS<n> :SPI:ENDian?
```

Description

Sets or queries the endian of data transmission in SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<endian>	Discrete	{MSB LSB}	MSB

Remarks

- MSB:** indicates Most Significant Bit transmission sequence, that is, the highest bit of the data is transmitted first.
- LSB:** indicates Least Significant Bit transmission sequence, that is, the lowest bit of the data is transmitted first.

Return Format

The query returns MSB or LSB.

Example

```
:BUS1:SPI:ENDian LSB      /*Sets the endian of data transmission
in SPI decoding to LSB.*/
:BUS1:SPI:ENDian?          /*The query returns LSB.*/
```

3.4.13.10 :BUS<n>:SPI:MODE

Syntax

```
:BUS<n> :SPI:MODE <mode>
:BUS<n> :SPI:MODE?
```

Description

Sets or queries the decode mode of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<mode>	Discrete	{CS TImeout}	TImeout

Remarks

- **CS:** indicates chip select. It contains a chip select line (CS). You can perform frame synchronization according to CS.
- **TImeout:** indicates timed out. You can perform frame synchronization according to the timeout.

For DHO800 series, only 4-channel models support CS setting for SPI decoding.

Return Format

The query returns CS or TIM.

Example

```
:BUS1:SPI:MODE CS /*Sets the decode mode of SPI decoding to CS.*/
:BUS1:SPI:MODE? /*The query returns CS.*/
```

3.4.13.11 :BUS<n>:SPI:TImeout:TIME**Syntax**

```
:BUS<n>:SPI:TImeout:TIME <time>
:BUS<n>:SPI:TImeout:TIME?
```

Description

Sets or queries the timeout value of SPI decoding on the specified bus. The unit is s.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<time>	Real	8 ns to 10 s	1μs

Remarks

- This setting command is only valid in timeout mode. You can send **:BUS<n>:SPI:MODE** to set or query the decode mode of SPI decoding.

- The timeout must be greater than the maximum clock pulse width and less than the idle time between frames.

Return Format

The query returns the timeout value in scientific notation.

Example

```
:BUS1:SPI:TIMEout:TIME 0.000005      /*Sets the timeout value to 5
μs.*/
:BUS1:SPI:TIMEout:TIME?                  /*The query returns
5.000000E-6.*/
```

3.4.13.12 :BUS<n>:SPI:SS:SOURce

Syntax

```
:BUS<n>:SPI:SS:SOURCE <source>
:BUS<n>:SPI:SS:SOURCE?
```

Description

Sets or queries the source of the CS line of SPI decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANne1 CHANne2 CHANne3 CHANne4}	CHANne3

Remarks

This setting command is only valid in CS mode. You can send [:BUS<n>:SPI:MODE](#) to set or query the decode mode of the SPI decoding.

Only DHO900 series supports the digital channels D0-D15.

For DHO800 series, only 4-channel models support this command.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:SPI:SS:SOURce CHANnel2    /*Sets the source of CS line of
SPI decoding to CHANnel2.*/
:BUS1:SPI:SS:SOURce?          /*The query returns CHAN2.*/
```

3.4.13.13 :BUS<n>:SPI:SS:POLarity**Syntax**

```
:BUS<n>:SPI:SS:POLarity <polarity>
:BUS<n>:SPI:SS:POLarity?
```

Description

Sets or queries the polarity of CS line of SPI decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<polarity>	Discrete	{HIGH LOW}	LOW

Remarks

- **HIGH:** indicates that the oscilloscope samples data of the source channel of data line on the specified edge of the clock signal when the CS signal is high level.
- **LOW:** indicates that the oscilloscope samples data of the source channel of data line on the specified edge of the clock signal when the CS signal is low level.

This setting command is only valid in CS mode. You can send [:BUS<n>:SPI:MODE](#) to set or query the decode mode of the SPI decoding.

For DHO800 series, only 4-channel models support this command.

Return Format

The query returns HIGH or LOW.

Example

```
:BUS1:SPI:SS:POLarity HIGH /*Sets the polarity of CS line of SPI
decoding to HIGH.*/
:BUS1:SPI:SS:POLarity? /*The query returns HIGH.*/
```

3.4.14 :BUS<n>:CAN

The :BUS<n>:CAN commands are used to set relevant parameters for CAN decoding.

The oscilloscope samples the CAN at the specified sample position, and judges each data point to be logic "1" or logic "0" according to the preset threshold level. You need to specify the CAN type and sample position for CAN decoding.

Sample Position

Sample position is a point within a bit's time. The oscilloscope samples the bit level at this point. The sample point position is expressed as the ratio of "time from the bit start to the sample point" to "bit time", in %.

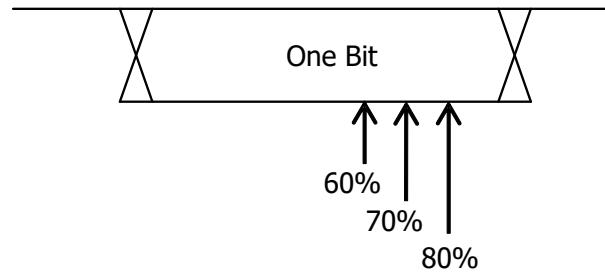


Figure 3.5 Sample Position

NOTE



Only the DHO900 series supports :BUS<n>:CAN commands.

3.4.14.1 :BUS<n>:CAN:SOURce

Syntax

:BUS</n> :CAN :SOURce <source>

:BUS</n> :CAN :SOURce?

Description

Sets or queries the source of CAN decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Remarks

D0-D15 are only available for the DHO900 series.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:CAN:SOURce CHANnel2    /*Sets the source of CAN decoding to  
CHANne12.*/  
:BUS1:CAN:SOURce?          /*The query returns CHAN2.*/
```

3.4.14.2 :BUS<n>:CAN:STYPe

Syntax

```
:BUS </n> :CAN:STYPe <stype>  
:BUS </n> :CAN:STYPe?
```

Description

Sets or queries the signal type of CAN decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<stype>	Discrete	{TX RX CANH CANL DIFFerential}	CANL

Remarks

- TX:** indicates the Transmit signal from the CAN bus transceiver.
- RX:** indicates the Receive signal from the CAN bus transceiver.
- CANH:** indicates the actual CAN_H differential bus signal.
- CANL:** indicates the actual CAN_L differential bus signal.
- DIFFerential:** indicates the CAN differential bus signal connected to an analog channel by using a differential probe. Connect the differential probe's positive lead to the CAN_H bus signal and connect the negative lead to the CAN_L bus signal.

Return Format

The query returns TX, RX, CANH, CANL, or DIFF.

Example

```
:BUS1:CAN:STYPe TX      /*Sets the signal type of CAN decoding
to TX.*/
:BUS1:CAN:STYPe?        /*The query returns TX.*/
```

3.4.14.3 :BUS<n>:CAN:BAUD**Syntax**

```
:BUS <n> :CAN :BAUD <baud>
:BUS <n> :CAN :BAUD?
```

Description

Sets or queries the signal rate of CAN decoding on the specified decoding bus. The unit is bps.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<baud>	Integer	10 kbps to 5 Mbps	1 Mbps

Remarks

If the baud rate is set to a value with "M", then "A" should be added at the end of the value. For example, if you send 5M, you need to send 5MA.

Return Format

The query returns an integer ranging from 10k to 5M.

Example

```
:BUS1:CAN:BAUD 120000    /*Sets the signal rate of CAN decoding to
120000 bps*/
:BUS1:CAN:BAUD?          /*The query returns 120000.*/
```

3.4.14.4 :BUS<n>:CAN:SPOint**Syntax**

```
:BUS <n> :CAN :SPOint <spoint>
:BUS <n> :CAN :SPOint?
```

Description

Sets or queries the sample point position of the CAN decoding on the specified bus (expressed in %).

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<spoint>	Integer	10 to 90	50

Remarks

For detailed sample position, refer to [Sample Position](#).

Return Format

The query returns an integer ranging from 10 to 90.

Example

```
:BUS1:CAN:SPOint 70          /*Sets the sample point position of CAN
decoding to 70%.*/
:BUS1:CAN:SPOint?           /*The query returns 70.*/
```

3.4.15 :BUS<n>:LIN

The :BUS<n>:LIN commands are used to set relevant parameters for LIN decoding.

The oscilloscope samples the LIN signal, and judges each data point to be logic "1" or logic "0" according to the preset threshold level. The LIN decoding is required to specify the LIN signal protocol version.

NOTE

Only the DHO900 series supports :BUS<n>:LIN commands.

3.4.15.1 :BUS<n>:LIN:PARity

Syntax

```
:BUS<n>:LIN:PARity <bool>
:BUS<n>:LIN:PARity?
```

Description

Sets or queries whether the LIN decoding on the specified decoding bus includes the parity bit.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<bool>	Bool	{1 ON}{0 OFF}	0 OFF

Remarks

- **1|ON:** includes the parity bit.
- **0|OFF:** does not include the parity bit.

Return Format

The query returns 0 or 1.

Example

```
:BUS1:LIN:PARITY ON          /*Sets the parity bit to be
included in LIN decoding.*/
:BUS1:LIN:PARITY?           /*The query returns 1.*/
```

3.4.15.2 :BUS<n>:LIN:SOURce**Syntax**

```
:BUS<n> :LIN:SOURce <source>
:BUS<n> :LIN:SOURce?
```

Description

Sets or queries the source of LIN bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<source>	Discrete	{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15 CHANnel1 CHANnel2 CHANnel3 CHANnel4}	CHANnel1

Remarks

D0-D15 are only available for the DHO900 series.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:LIN:SOURce CHANnel2    /*Sets the source of LIN bus to
CHANnel2.*/
:BUS1:LIN:SOURce?           /*The query returns CHAN2.*/
```

3.4.15.3 :BUS<n>:LIN:STANDARD

Syntax

:BUS <n> :LIN:STANDARD <value>

:BUS <n> :LIN:STANDARD?

Description

Sets or queries the version of LIN bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1 2 3 4}	-
<value>	Discrete	{V1X V2X MIXed}	MIXed

Remarks

N/A

Return Format

The query returns V1X, V2X, or MIX.

Example

```
:BUS1:LIN:STANDARD V2X          /*Sets the LIN bus version to
V2X.*/
:BUS1:LIN:STANDARD?            /*The query returns V2X.*/
```

3.5 :BODEplot Commands

The **:BODEplot** commands are used to execute the bode-related settings and operations.

Bode plot is a way of graphically displaying the frequency response of a system. Through the analysis on the system's gain and phase margins, you can determine the stability of the system.

With the built-in signal generator module, the oscilloscope generates the sweep signal of a specified frequency range and outputs to the switch power supply circuit under test. Then, the oscilloscope draws a Bode plot displaying the variation of phase and gain with different frequencies.

NOTE

The commands are only available for DHO914S and DHO924S.