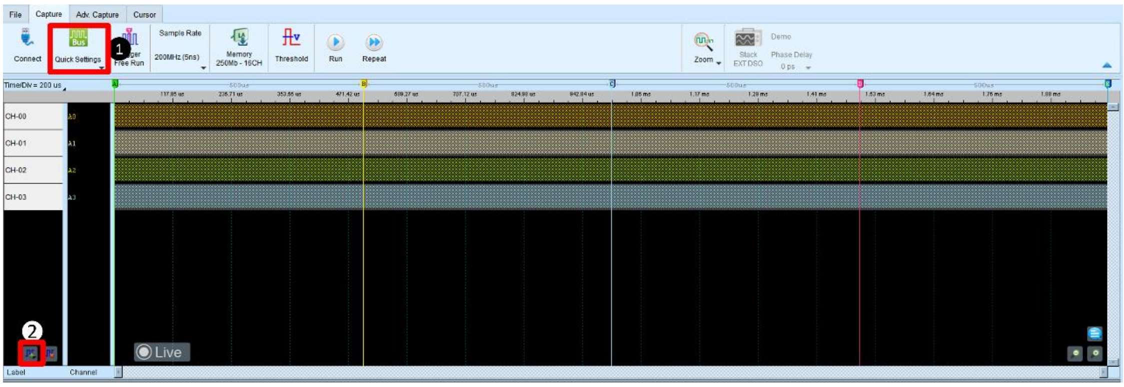
# Table of Contents

# Chapter 1 Bus Decode

## Basic operation

### Add a Bus Decode



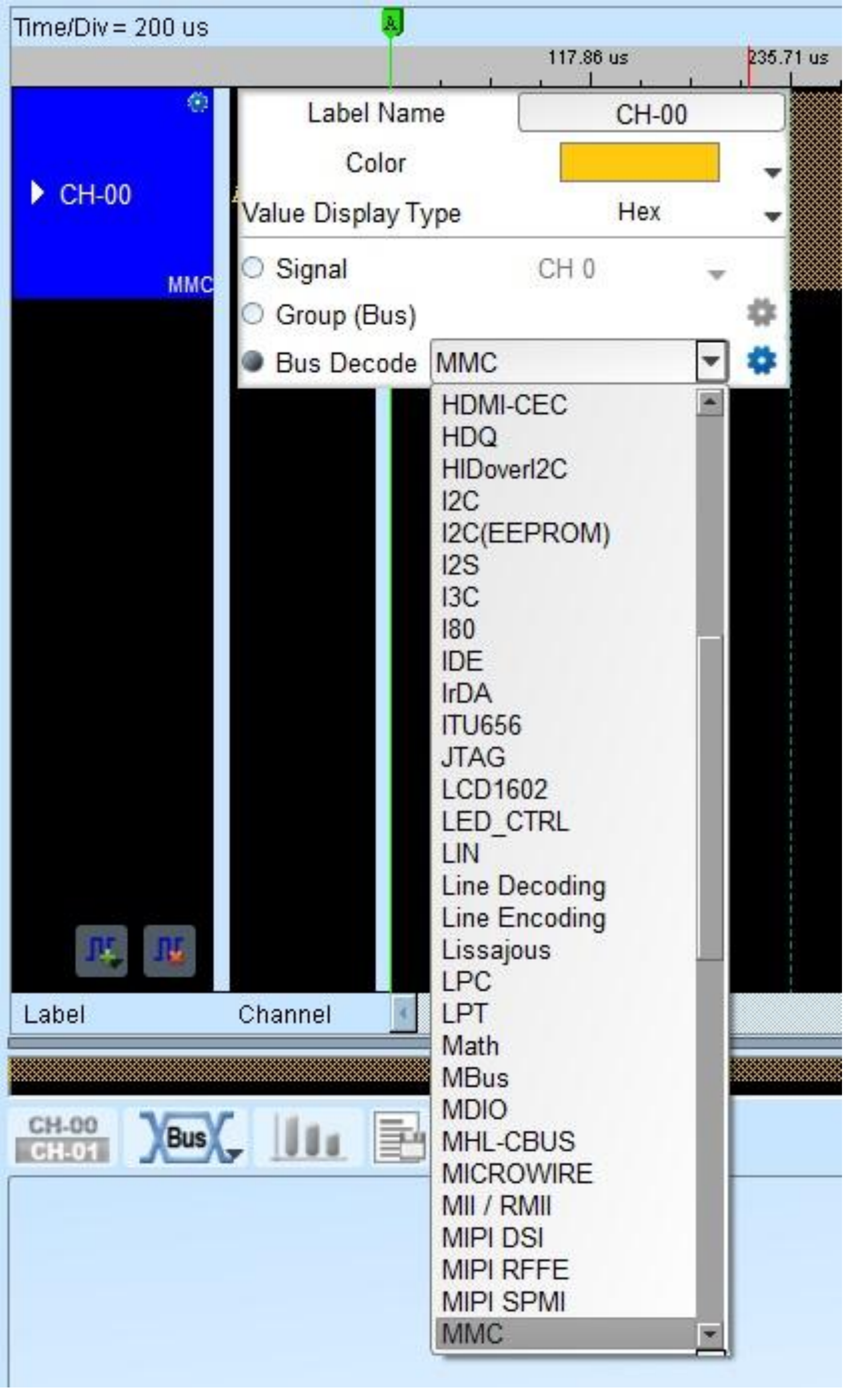
**Method 1:**

Click the Quick Setting in the menu (number 1 in the figure above), and select the bus decode.

**Method 2:**

Click Add Bus Decode (number 2 int the figure above) in the Label menu or right-click the label field to show the dialog box.

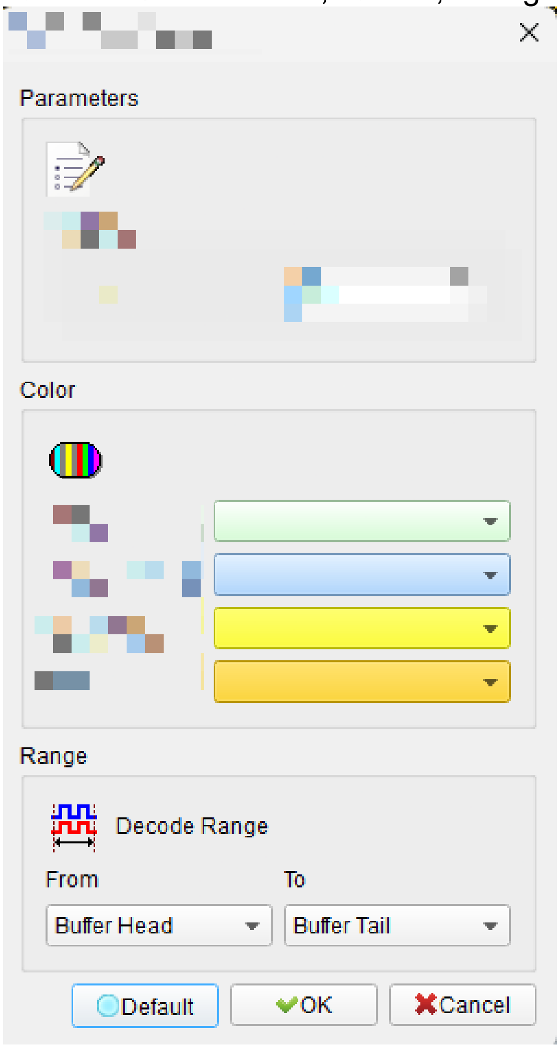
## Advance channel setting



Label Name: CH-00
Color
Value Display Type: Hex
○ Signal: CH 0
○ Group (Bus)
● Bus Decode: MMC

1. **Bus Name:** Enter the label name with 31 characters or less. (Chinese word expresses two characters.)
2. **Color:** Set the waveform color.
3. **Display the waveforms with decode**
4. **Display the waveforms with its decode together.**
5. **Advance:**

Set the decode parameters or press **OK** to use default settings. There are "Channel"/"Parameter', "Color", "Range" settings



Parameters

Color

Range

Decode Range

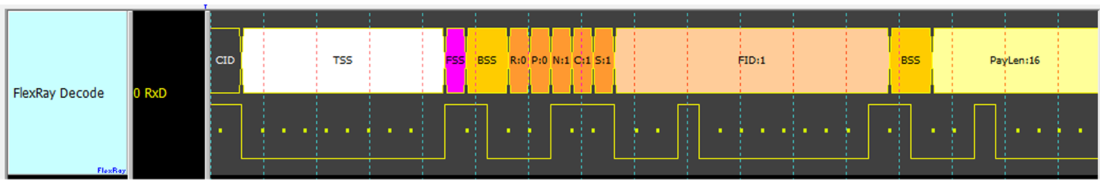
From: Buffer Head

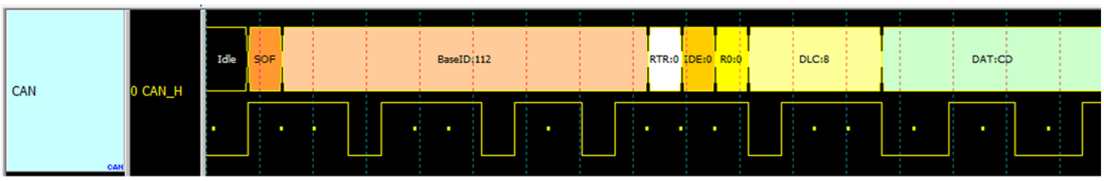
To: Buffer Tail

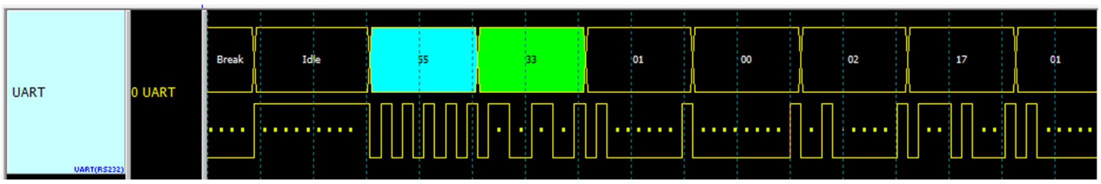
## Specially Bus Decode

The bus decodes display the data in text format, but some decodes are able to show the original form for the data such as voice (I2S decode…), image, analog waveform (ADC decode) etc.

**UART/CAN/FlexRay..bus decodes** (released since 2009/9, LA Viewer Ver2.0): **The data is displayed according to bit points in order to calculate the bit number。**

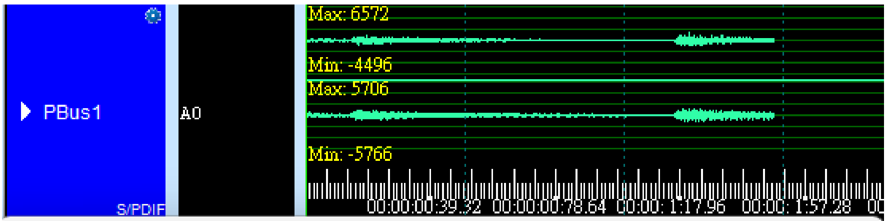


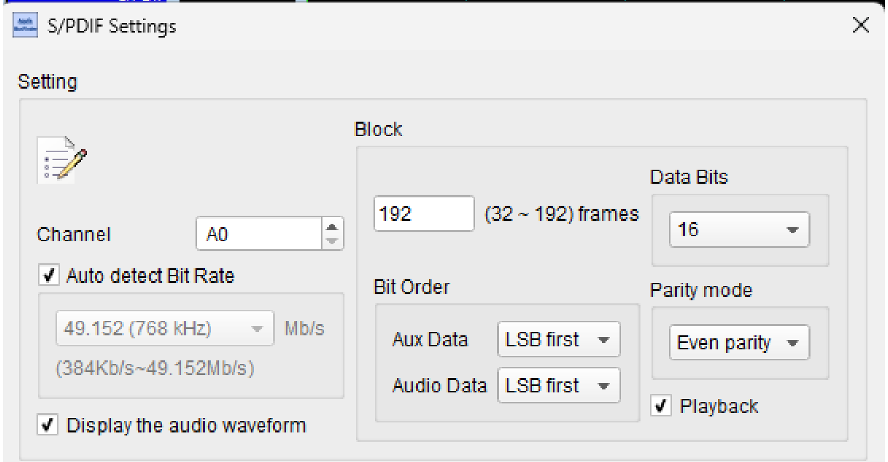




**S/PDIF analysis (released since 2010/11, LA Viewer Ver2.5):**

Display the data in sound waveform.




S/PDIF Settings

Setting

Channel A0

☑ Auto detect Bit Rate

49.152 (768 kHz) Mb/s

(384Kb/s~49.152Mb/s)

☑ Display the audio waveform

Block

192 (32 ~ 192) frames

Data Bits 16

Bit Order

Aux Data LSB first

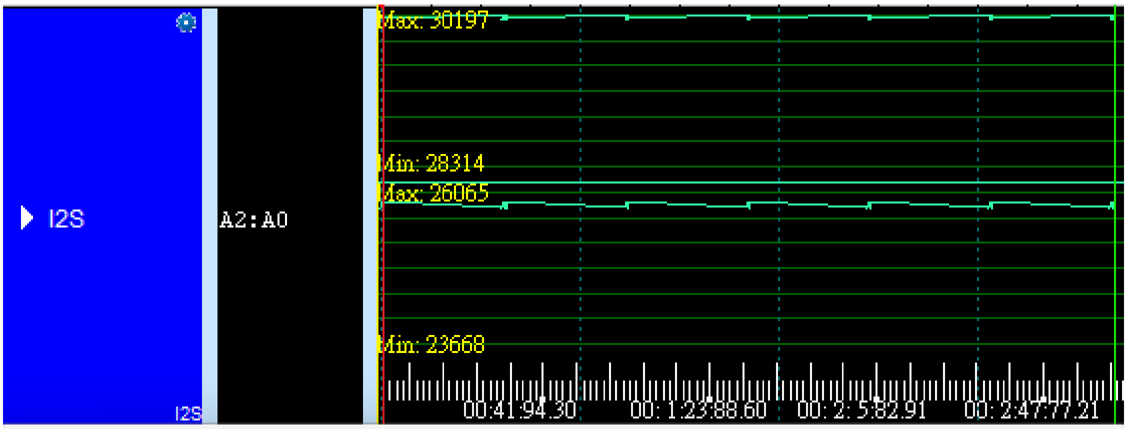
Audio Data LSB first

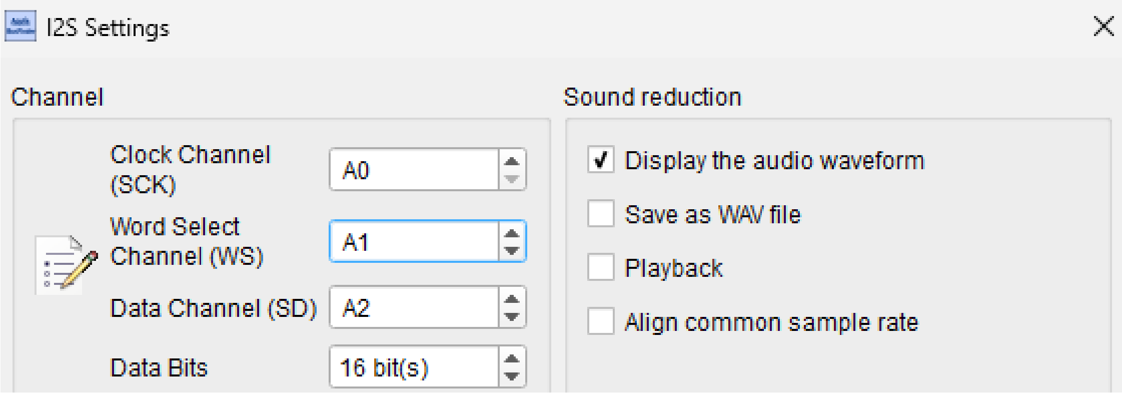
Parity mode Even parity

☑ Playback

## I$^2$S analysis (released since 2011/9, LA Viewer Ver2.6.3):

Display the data in sound waveform。





I2S Settings

Channel

Clock Channel (SCK): A0

Word Select Channel (WS): A1

Data Channel (SD): A2

Data Bits: 16 bit(s)

Sound reduction

☑ Display the audio waveform

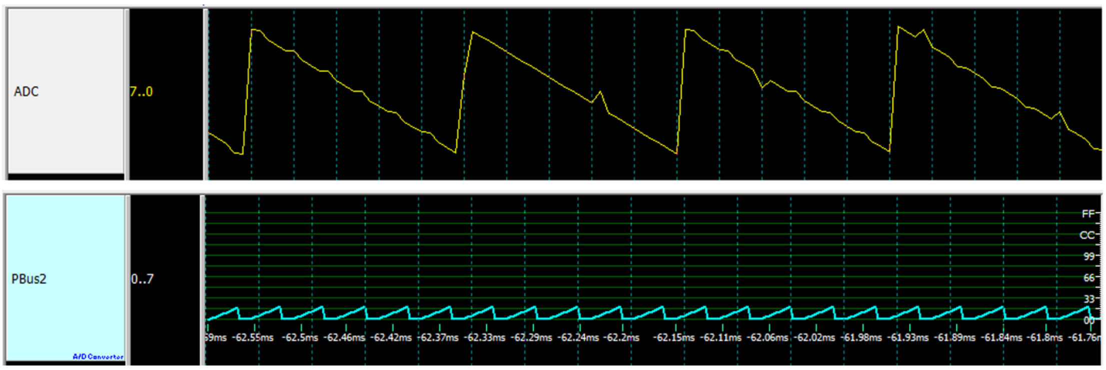
☐ Save as WAV file

☐ Playback

☐ Align common sample rate

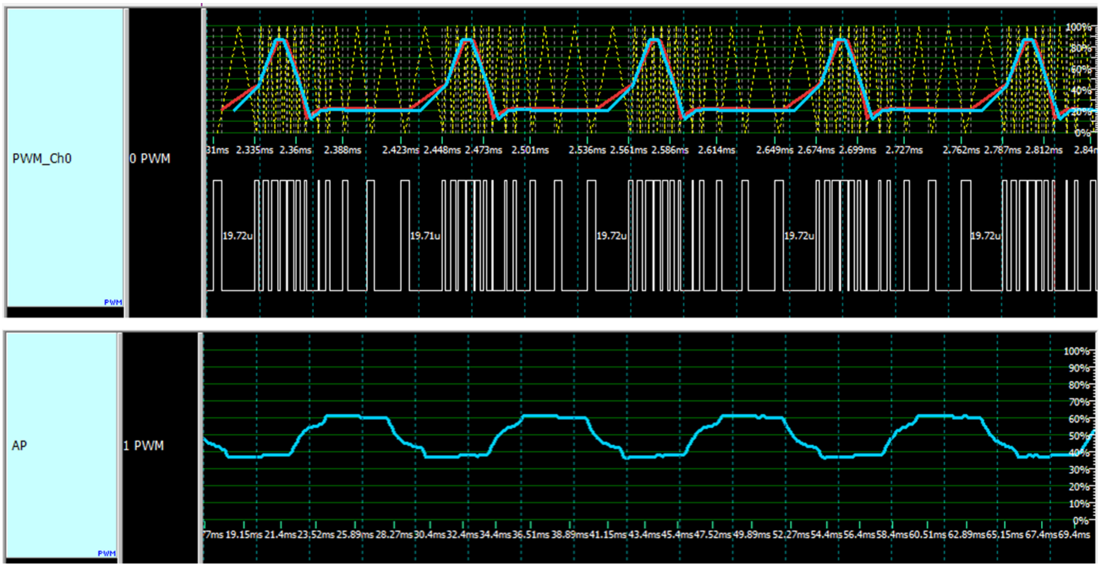
## ADC bus decode:

Display the input data in graph。



**PWM analysis (released since 2012/8, LA Viewer Ver2.7.3):**

Restore and display the data in the waveform window as percentile or frequency。


PWM_Ch0
PWM
0 PWM
100%
80%
60%
40%
20%
0%
31ms 2.335ms 2.36ms 2.388ms 2.423ms 2.448ms 2.473ms 2.501ms 2.536ms 2.561ms 2.586ms 2.614ms 2.649ms 2.674ms 2.699ms 2.727ms 2.762ms 2.787ms 2.812ms 2.84r
19.72u
19.71u
19.72u
19.72u
19.72u
AP
PWM
1 PWM
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
7ms 19.15ms 21.4ms 23.52ms 25.89ms 28.27ms 30.4ms 32.4ms 34.4ms 36.51ms 38.89ms 41.15ms 43.4ms 45.4ms 47.52ms 49.89ms 52.27ms 54.4ms 56.4ms 58.4ms 60.51ms 62.89ms 65.15ms 67.4ms 69.4ms

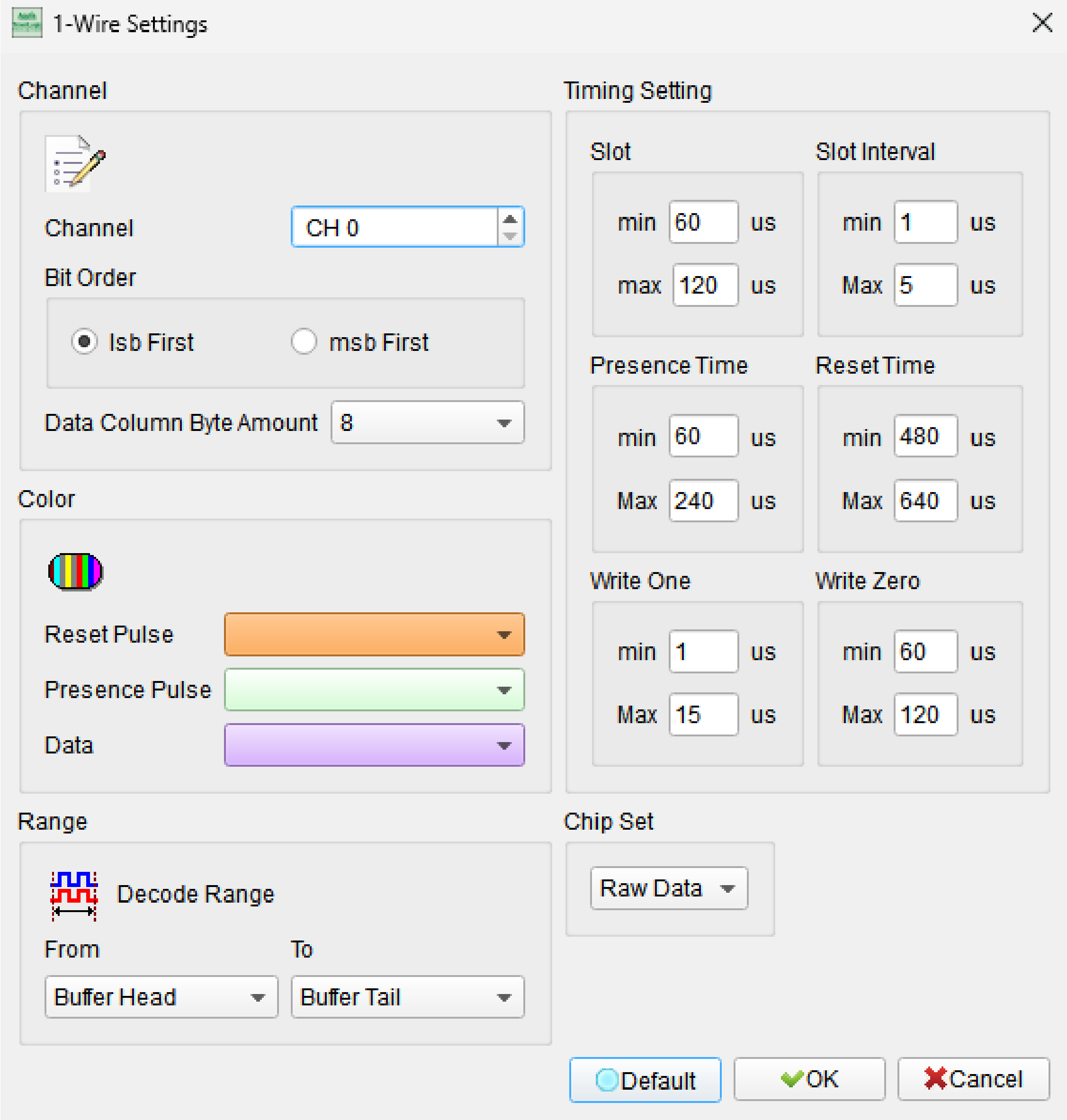
### Bus Decode Settings Introduction

Please note: All are prefixed with 'Ch' except BusFinder or LA in the channel, which are prefixed with 'A'.

## 1-Wire

Developed by Dallas Semiconductor, the 1-Wire protocol defines several signal types such as Reset Pulse, Presence Pulse, Write 1, Write 0, Read 1, and Read 0, and combines these signal types into a command sequence. The transmission mode is LSB (Least-significant bit) to MSB (Most-significant bit), and the transmission speed is divided into Overdrive speed and Standard speed.

## Settings



1-Wire Settings

Channel

Channel: CH 0

Bit Order: ● lsb First ○ msb First

Data Column Byte Amount: 8

Color

Reset Pulse: 🟧

Presence Pulse: 🟩

Data: 🟪

Range

Decode Range

From: Buffer Head To: Buffer Tail

Timing Setting

| | | | | | |
|---|---|---|---|---|---|
| **Slot** | | | **Slot Interval** | | |
| min | 60 | us | min | 1 | us |
| max | 120 | us | Max | 5 | us |
| **Presence Time** | | | **Reset Time** | | |
| min | 60 | us | min | 480 | us |
| Max | 240 | us | Max | 640 | us |
| **Write One** | | | **Write Zero** | | |
| min | 1 | us | min | 60 | us |
| Max | 15 | us | Max | 120 | us |

Chip Set: Raw Data

**Channel:** Setting the Data Channel Source

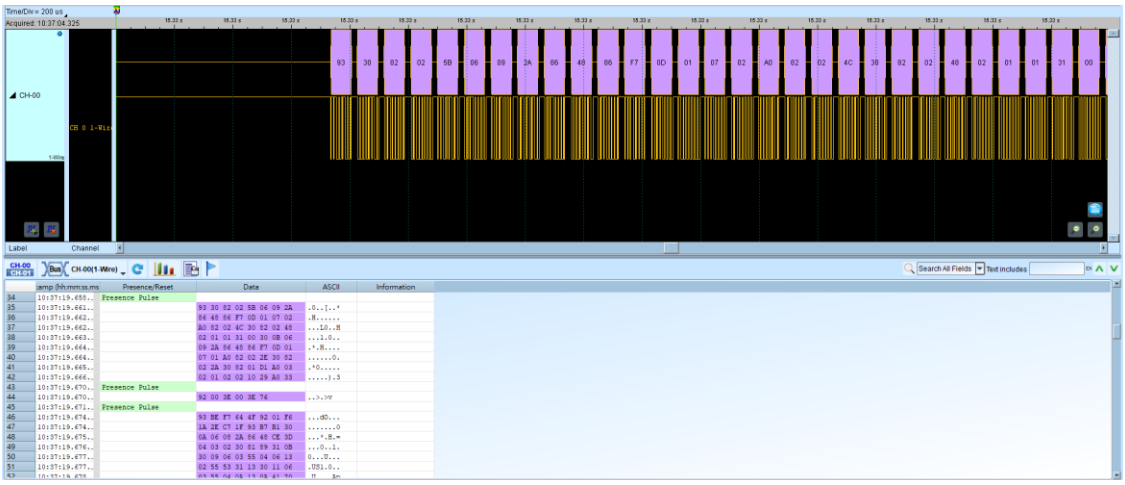
**Bit Order:** Set whether the analyzed data is LSB first or MSB first.

**Data Byte Column Amount:** Set how many bytes of data are displayed in a row of the Data field in the Report area; user can select 8, 16, or 32.

**Timing Setting: all unit is us.**

1. **Slot Time:**
   - I. **Min:** Set the minimum length of time of a slot.
   - II. **Max:** Set the maximum length of time of a slot.
2. **Slot Interval:**
   - I. **Min:** Set the minimum length of time between each slot.
   - II. **Max:** Set the maximum length of time between each slot.
3. **Reset Time:**
   - I. **Min:** Set the minimum length of time that the waveform will be pulled down when is in Reset.
   - II. **Max:** Set the maximum length of time that the waveform will be pulled down when is in Reset.
4. **Presence Time:**
   - I. **Min:** Set the minimum length of time that the waveform will be pulled down when is in Presence.
   - II. **Max:** Set the maximum length of time that the waveform will be pulled down when is in Presence.
5. **Write One:**
   - I. **Min:** Set the minimum length of time that the waveform will be pulled down when 1 is written.
   - II. **Max:** Set the maximum length of time that the waveform will be pulled down when 1 is written.
6. **Write Zero:**
   - I. **Min:** Set the minimum length of time that the waveform will be pulled down when 0 is written.
   - II. **Max:** Set the maximum length of time that the waveform will be pulled down when 0 is written.

**Result**

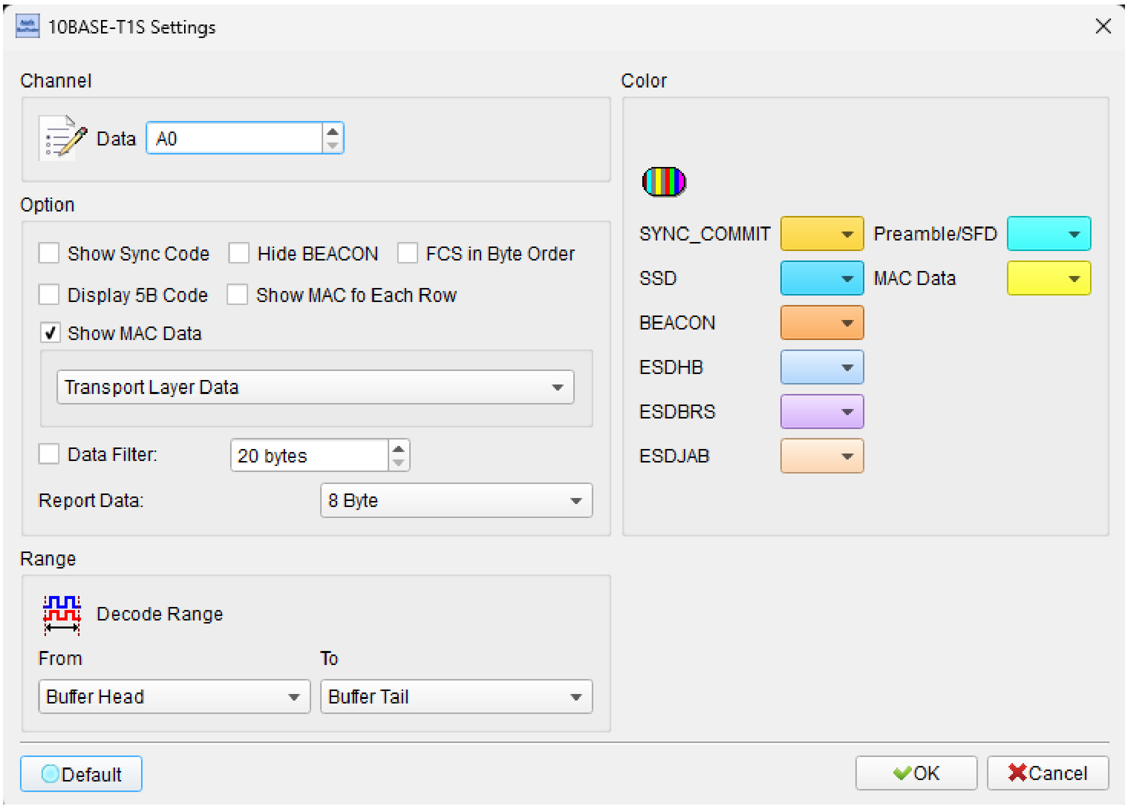


| | tamp (hh:mm:ss.ms | Presence/Reset | Data | ASCII | Information |
|---|---|---|---|---|---|
| 34 | 10:37:19.658.. | Presence Pulse | | | |
| 35 | 10:37:19.661.. | | 93 30 82 02 5B 06 09 2A | .0..[..* | |
| 36 | 10:37:19.662.. | | 86 48 86 F7 0D 01 07 02 | .H...... | |
| 37 | 10:37:19.662.. | | A0 82 02 4C 30 82 02 48 | ...L0..H | |
| 38 | 10:37:19.663.. | | 02 01 01 31 00 30 0B 06 | ...1.0.. | |
| 39 | 10:37:19.664.. | | 09 2A 86 48 86 F7 0D 01 | .*.H.... | |
| 40 | 10:37:19.664.. | | 07 01 A0 82 02 2E 30 82 | ......0. | |
| 41 | 10:37:19.665.. | | 02 2A 30 82 01 D1 A0 03 | .*0..... | |
| 42 | 10:37:19.666.. | | 02 01 02 02 10 29 A0 33 | .....).3 | |
| 43 | 10:37:19.670.. | Presence Pulse | | | |
| 44 | 10:37:19.670.. | | 92 00 3E 00 3E 76 | ..>.>v | |
| 45 | 10:37:19.671.. | Presence Pulse | | | |
| 46 | 10:37:19.674.. | | 93 BE F7 64 4F 92 01 F6 | ...dO... | |
| 47 | 10:37:19.674.. | | 1A 2E C7 1F 93 B7 B1 30 | .......0 | |
| 48 | 10:37:19.675.. | | 0A 06 08 2A 86 48 CE 3D | ...*.H.= | |
| 49 | 10:37:19.676.. | | 04 03 02 30 81 89 31 0B | ...0..1. | |
| 50 | 10:37:19.677.. | | 30 09 06 03 55 04 06 13 | 0...U... | |
| 51 | 10:37:19.677.. | | 02 55 53 31 13 30 11 06 | .US1.0.. | |
| 52 | 10:37:19.678 | | 03 55 04 08 13 0A 41 70 | .U....Ap | |

## 10BASE-T1S

10Base-T1S uses Differential Manchester Encoding (DME). DME encodes data based on whether there is a transition within the clock period, indicating the logical state of the signal. If there is no transition within the clock period, the data state is logical 0. If transitions (either positive or negative) occur within the clock period, the data state is logical 1.
10BASE-T1S operates in a half-duplex bus configuration with a maximum length of 25 meters. It supports multipoint connections with two to eight nodes. The "S" in the standard name denotes short-range implementation. The intended use of 10BASE-T1S is to replace existing bus architectures that often result in "communication islands," such as CAN, CAN FD, LIN, and RS-485.

### Settings:



10BASE-T1S Settings

Channel

Data: A0

Option

□ Show Sync Code □ Hide BEACON □ FCS in Byte Order

□ Display 5B Code □ Show MAC fo Each Row

☑ Show MAC Data

Transport Layer Data

□ Data Filter: 20 bytes

Report Data: 8 Byte

Range

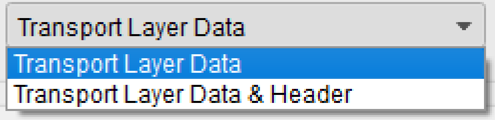
Decode Range

From: Buffer Head To: Buffer Tail

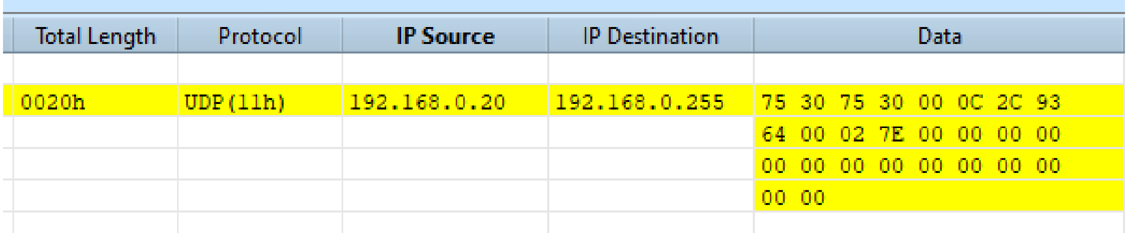
**Show Sync Code:** Show Commit, SSD in report; enabled when checked.
**Hide BEACON:** Hide Beacon data is not shown; enabled when checked.
**FCS in Byte order:** Present FCS in report in byte order; enabled when checked.
**Display 5B Code:** Show special code value

**Show MAC Data:** Show MAC Packet.

In IPv4(0800h), there will be 20bytes header. The Data column is able to show Transport Layer Data only or including the header.

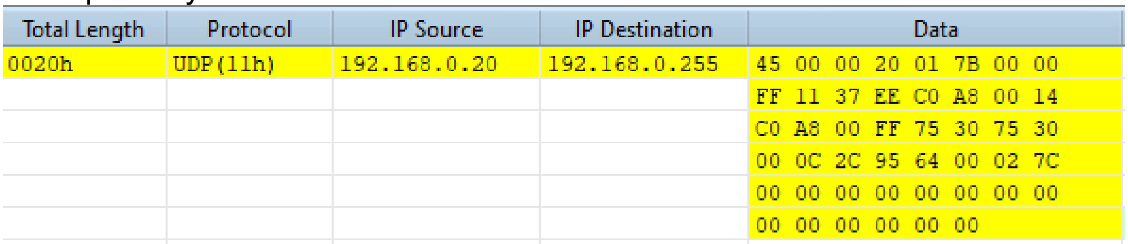


Transport Layer Data:



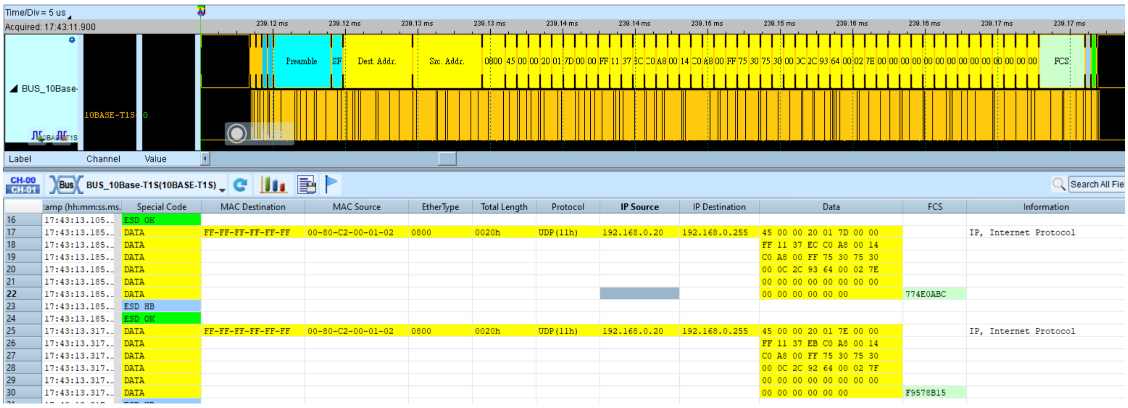
| Total Length | Protocol | IP Source | IP Destination | Data |
|---|---|---|---|---|
| | | | | |
| 0020h | UDP(11h) | 192.168.0.20 | 192.168.0.255 | 75 30 75 30 00 0C 2C 93 |
| | | | | 64 00 02 7E 00 00 00 00 |
| | | | | 00 00 00 00 00 00 00 00 |
| | | | | 00 00 |

Transport Layer Data & Header:



| Total Length | Protocol | IP Source | IP Destination | Data |
|---|---|---|---|---|
| 0020h | UDP(11h) | 192.168.0.20 | 192.168.0.255 | 45 00 00 20 01 7B 00 00 |
| | | | | FF 11 37 EE C0 A8 00 14 |
| | | | | C0 A8 00 FF 75 30 75 30 |
| | | | | 00 0C 2C 95 64 00 02 7C |
| | | | | 00 00 00 00 00 00 00 00 |
| | | | | 00 00 00 00 00 00 |

**Results:**

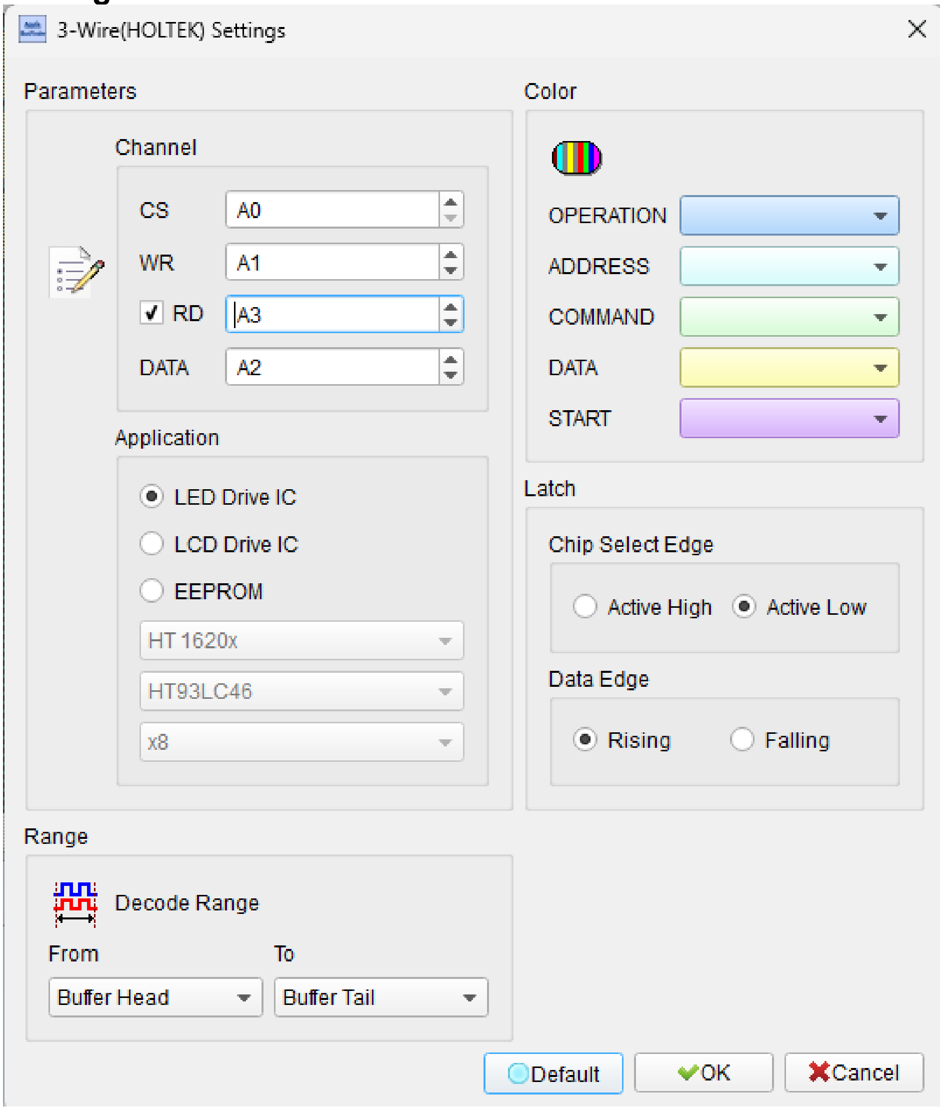


| | tamp (hh:mm:ss.ms. | Special Code | MAC Destination | MAC Source | EtherType | Total Length | Protocol | IP Source | IP Destination | Data | FCS | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 16 | 17:43:13.105... | ESD OK | | | | | | | | | | |
| 17 | 17:43:13.185... | DATA | FF-FF-FF-FF-FF-FF | 00-80-C2-00-01-02 | 0800 | 0020h | UDP(11h) | 192.168.0.20 | 192.168.0.255 | 45 00 00 20 01 7D 00 00 | | IP, Internet Protocol |
| 18 | 17:43:13.185... | DATA | | | | | | | | FF 11 37 EC C0 A8 00 14 | | |
| 19 | 17:43:13.185... | DATA | | | | | | | | C0 A8 00 FF 75 30 75 30 | | |
| 20 | 17:43:13.185... | DATA | | | | | | | | 00 0C 2C 93 64 00 02 7E | | |
| 21 | 17:43:13.185... | DATA | | | | | | | | 00 00 00 00 00 00 00 00 | | |
| 22 | 17:43:13.185... | DATA | | | | | | | | 00 00 00 00 00 00 | 774E0ABC | |
| 23 | 17:43:13.185... | ESD HB | | | | | | | | | | |
| 24 | 17:43:13.185... | ESD OK | | | | | | | | | | |
| 25 | 17:43:13.317... | DATA | FF-FF-FF-FF-FF-FF | 00-80-C2-00-01-02 | 0800 | 0020h | UDP(11h) | 192.168.0.20 | 192.168.0.255 | 45 00 00 20 01 7E 00 00 | | IP, Internet Protocol |
| 26 | 17:43:13.317... | DATA | | | | | | | | FF 11 37 EB C0 A8 00 14 | | |
| 27 | 17:43:13.317... | DATA | | | | | | | | C0 A8 00 FF 75 30 75 30 | | |
| 28 | 17:43:13.317... | DATA | | | | | | | | 00 0C 2C 92 64 00 02 7F | | |
| 29 | 17:43:13.317... | DATA | | | | | | | | 00 00 00 00 00 00 00 00 | | |
| 30 | 17:43:13.317... | DATA | | | | | | | | 00 00 00 00 00 00 | F9578B15 | |

## 3-Wire

3-Wire protocol is established by HOLTEK SEMICONDUCTOR INC. It’s Applied to control LED/LCD driver IC or EEPROM.

### Settings



3-Wire(HOLTEK) Settings

Parameters

Channel

| | |
|---|---|
| CS | A0 |
| WR | A1 |
| ☑ RD | A3 |
| DATA | A2 |

Application

● LED Drive IC
○ LCD Drive IC
○ EEPROM

HT 1620x
HT93LC46
x8

Color

OPERATION
ADDRESS
COMMAND
DATA
START

Latch

Chip Select Edge
○ Active High ● Active Low

Data Edge
● Rising ○ Falling

Range

Decode Range

From: Buffer Head
To: Buffer Tail

**Channel:** Show the selected channels (CS:CH0, WR:CH1, DATA:CH2, RD:CH3)

**LED Driver IC:** Select LED driver IC application.

**LCD Driver ID:** Select LCD driver IC application.

**EEPROM:** Select EEPROM application.

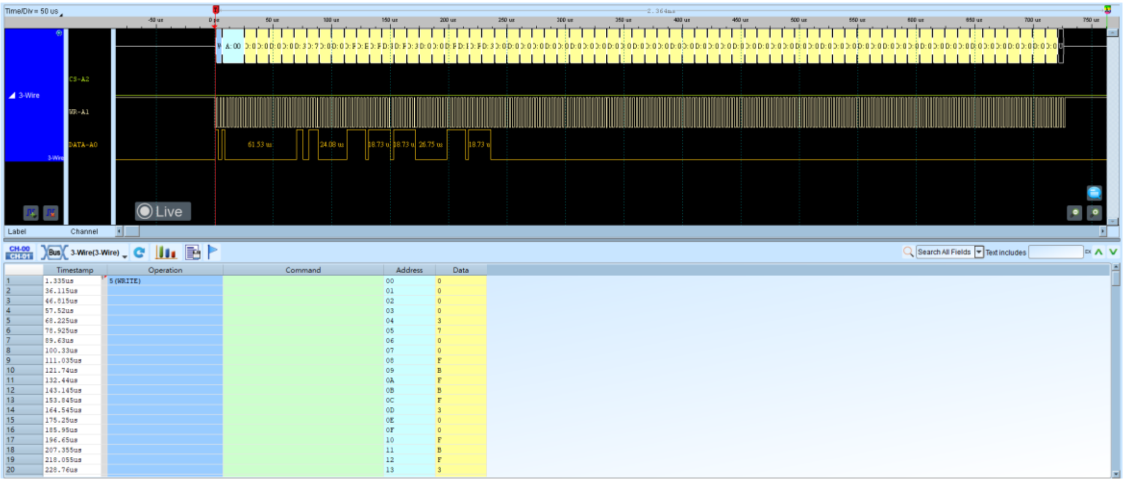
**Active High:** Select Active High.

**Active Low:** Select low chip select (CS).

**Rising:** Select Rising Data Edge.

**Falling:** Select Falling Data Edge.

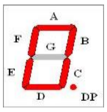
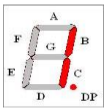
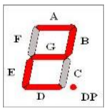
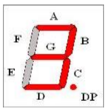
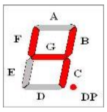
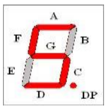
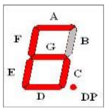
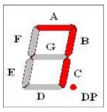
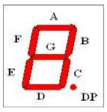
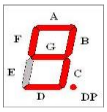
### Result



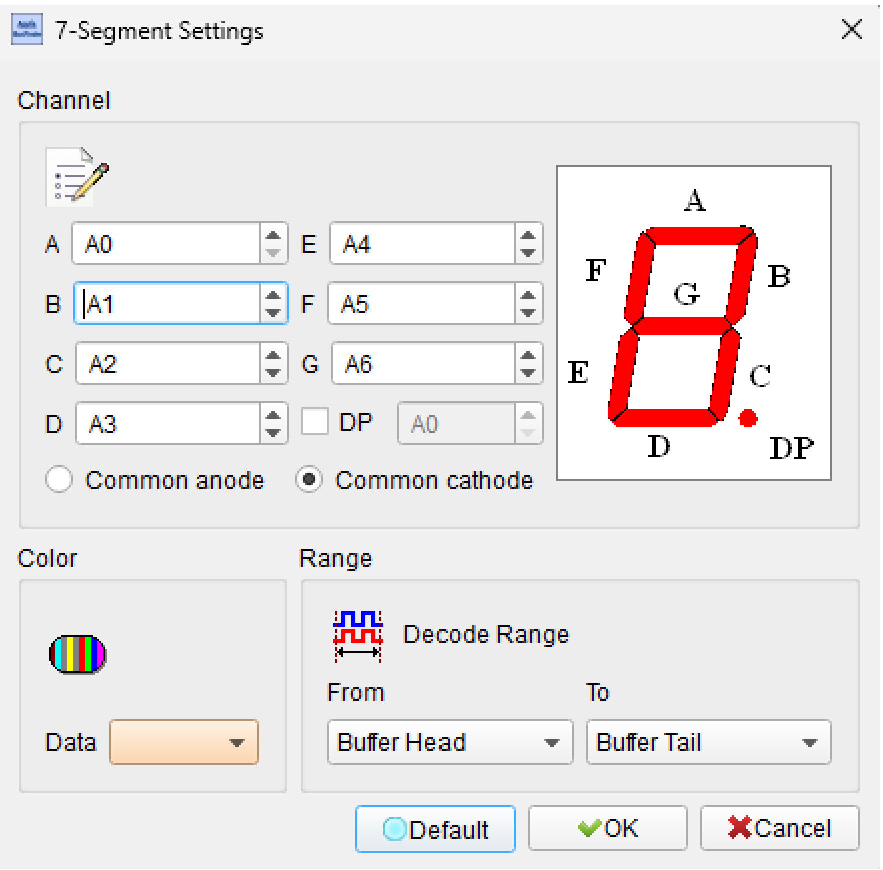
| | Timestamp | Operation | Command | Address | Data |
|---|---|---|---|---|---|
| 1 | 1.335us | 5(WRITE) | | 00 | 0 |
| 2 | 36.115us | | | 01 | 0 |
| 3 | 46.815us | | | 02 | 0 |
| 4 | 57.52us | | | 03 | 0 |
| 5 | 68.225us | | | 04 | 3 |
| 6 | 78.925us | | | 05 | 7 |
| 7 | 89.63us | | | 06 | 0 |
| 8 | 100.33us | | | 07 | 0 |
| 9 | 111.035us | | | 08 | F |
| 10 | 121.74us | | | 09 | B |
| 11 | 132.44us | | | 0A | F |
| 12 | 143.145us | | | 0B | B |
| 13 | 153.845us | | | 0C | F |
| 14 | 164.545us | | | 0D | 3 |
| 15 | 175.25us | | | 0E | 0 |
| 16 | 185.95us | | | 0F | 0 |
| 17 | 196.65us | | | 10 | F |
| 18 | 207.355us | | | 11 | B |
| 19 | 218.055us | | | 12 | F |
| 20 | 228.76us | | | 13 | 3 |

## 7-Segment

A seven-segment display, is a form of electronic display device for displaying decimal numerals that is an alternative to the more complex dot-matrix displays.

| Digit | LED | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|---|
| 0 |  | ON | ON | ON | ON | ON | ON | OFF |
| 1 |  | OFF | ON | ON | OFF | OFF | OFF | OFF |
| 2 |  | ON | ON | OFF | ON | ON | OFF | ON |
| 3 |  | ON | ON | ON | ON | OFF | OFF | ON |
| 4 |  | OFF | ON | ON | OFF | OFF | ON | ON |
| 5 |  | ON | OFF | ON | ON | OFF | ON | ON |
| 6 |  | ON | OFF | ON | ON | ON | ON | ON |
| 7 |  | ON | ON | ON | OFF | OFF | OFF | OFF |
| 8 |  | ON | ON | ON | ON | ON | ON | ON |
| 9 |  | ON | ON | ON | ON | OFF | ON | ON |

## Settings



7-Segment Settings

Channel

A: A0 E: A4

B: A1 F: A5

C: A2 G: A6

D: A3 □ DP A0

○ Common anode ● Common cathode

Color

Data

Range

Decode Range

From: Buffer Head

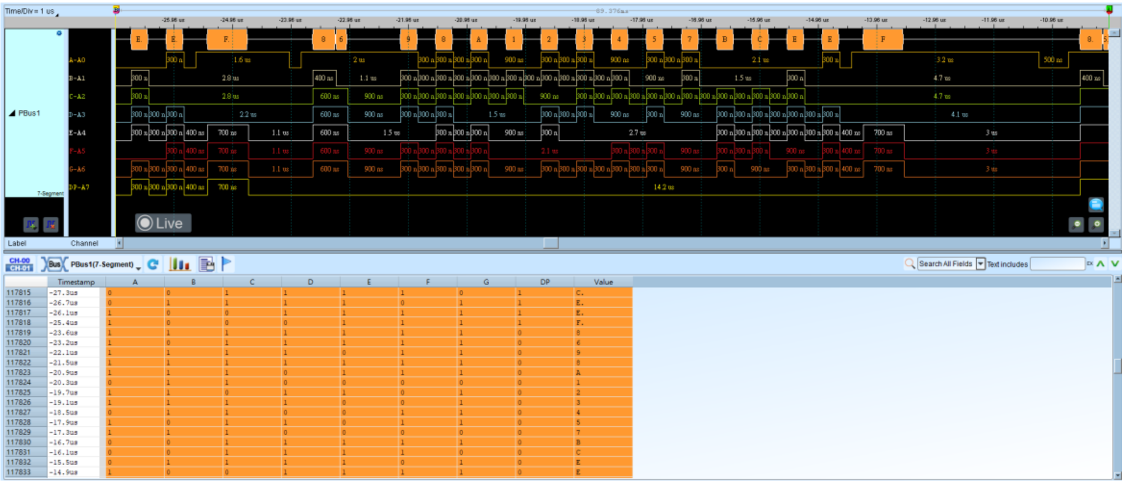
To: Buffer Tail

**Channel:** Show the selected channel (CH 0).

**DP**: to analysis decimal point.

**Common cathode/anode:** Show the same cathode or anode.

Result

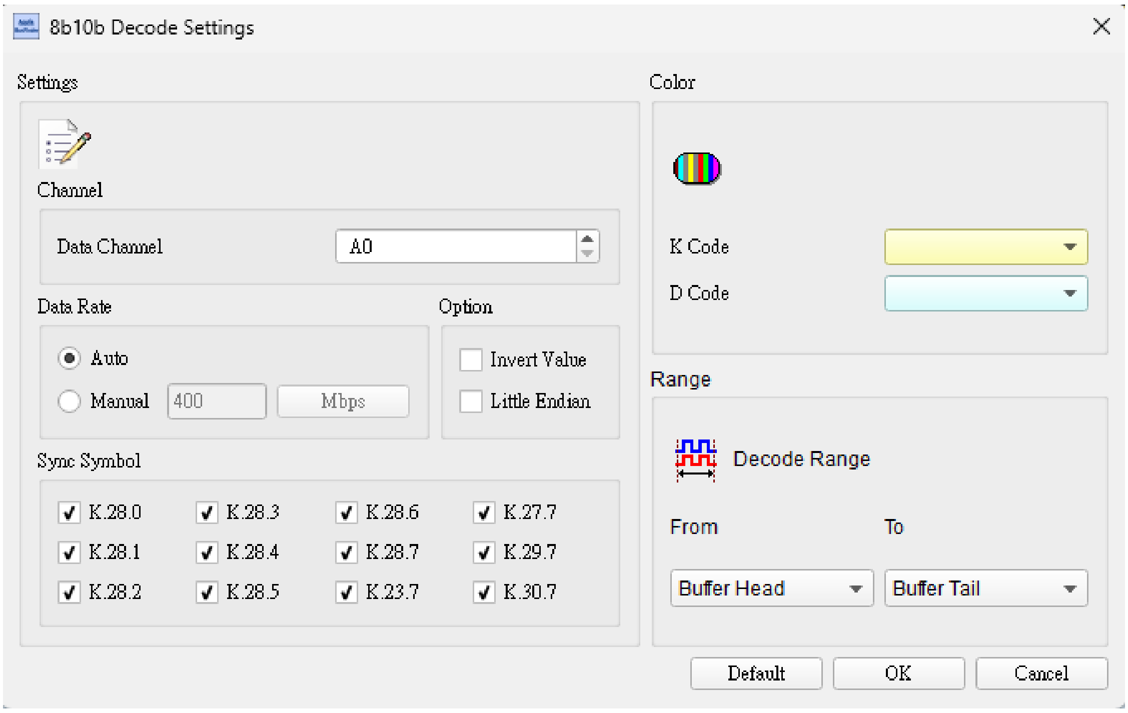


| | Timestamp | A | B | C | D | E | F | G | DP | Value |
|---|---|---|---|---|---|---|---|---|---|---|
| 117815 | -27.3us | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | C. |
| 117816 | -26.7us | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | E. |
| 117817 | -26.1us | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | E. |
| 117818 | -25.4us | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | F. |
| 117819 | -23.6us | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 8 |
| 117820 | -23.2us | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 6 |
| 117821 | -22.1us | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 9 |
| 117822 | -21.5us | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 8 |
| 117823 | -20.9us | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | A |
| 117824 | -20.3us | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 117825 | -19.7us | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 2 |
| 117826 | -19.1us | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 3 |
| 117827 | -18.5us | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 4 |
| 117828 | -17.9us | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 5 |
| 117829 | -17.3us | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 7 |
| 117830 | -16.7us | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | B |
| 117831 | -16.1us | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | C |
| 117832 | -15.5us | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | E |
| 117833 | -14.9us | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | E |

## 8b10b Decoding

8b/10b encoding is a coding technique commonly used in high-speed digital communications to convert every 8 bits of data into a 10 bits format. It was first invented by IBM in the 1980s to improve the reliability and stability of data transmission.

### Setting



8b10b Decode Settings

Settings

Channel

Data Channel: A0

Data Rate

● Auto

○ Manual 400 Mbps

Option

□ Invert Value

□ Little Endian

Sync Symbol

☑ K.28.0 ☑ K.28.3 ☑ K.28.6 ☑ K.27.7

☑ K.28.1 ☑ K.28.4 ☑ K.28.7 ☑ K.29.7

☑ K.28.2 ☑ K.28.5 ☑ K.23.7 ☑ K.30.7

Color

K Code

D Code

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Data Channel:** Set the signal on the DUT to the channel number of the logic analyzer.

(LVDS signals need to be converted into single-ended signals or use LVDS probes)

**Data Rate:** User can set the data rate manually, or choose the ‘Auto’, let software calculate the data rate automatically.

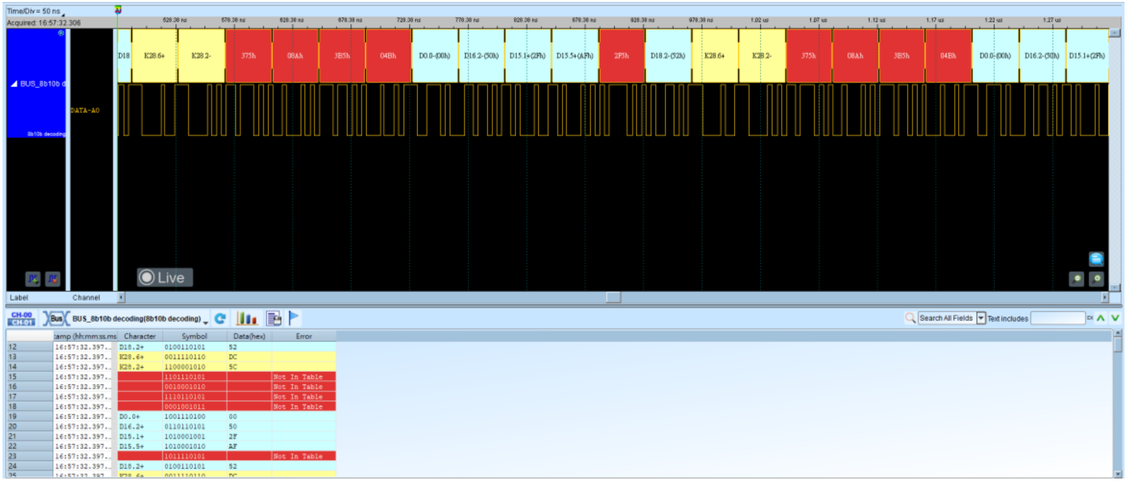
**Option:**

**Invert Value:** Invert the waveform.

**Little Endian:** The data would be sorted follow Little Endian when checked.

**Sync Symbol:** Choose which k-code for syncing.

### Result

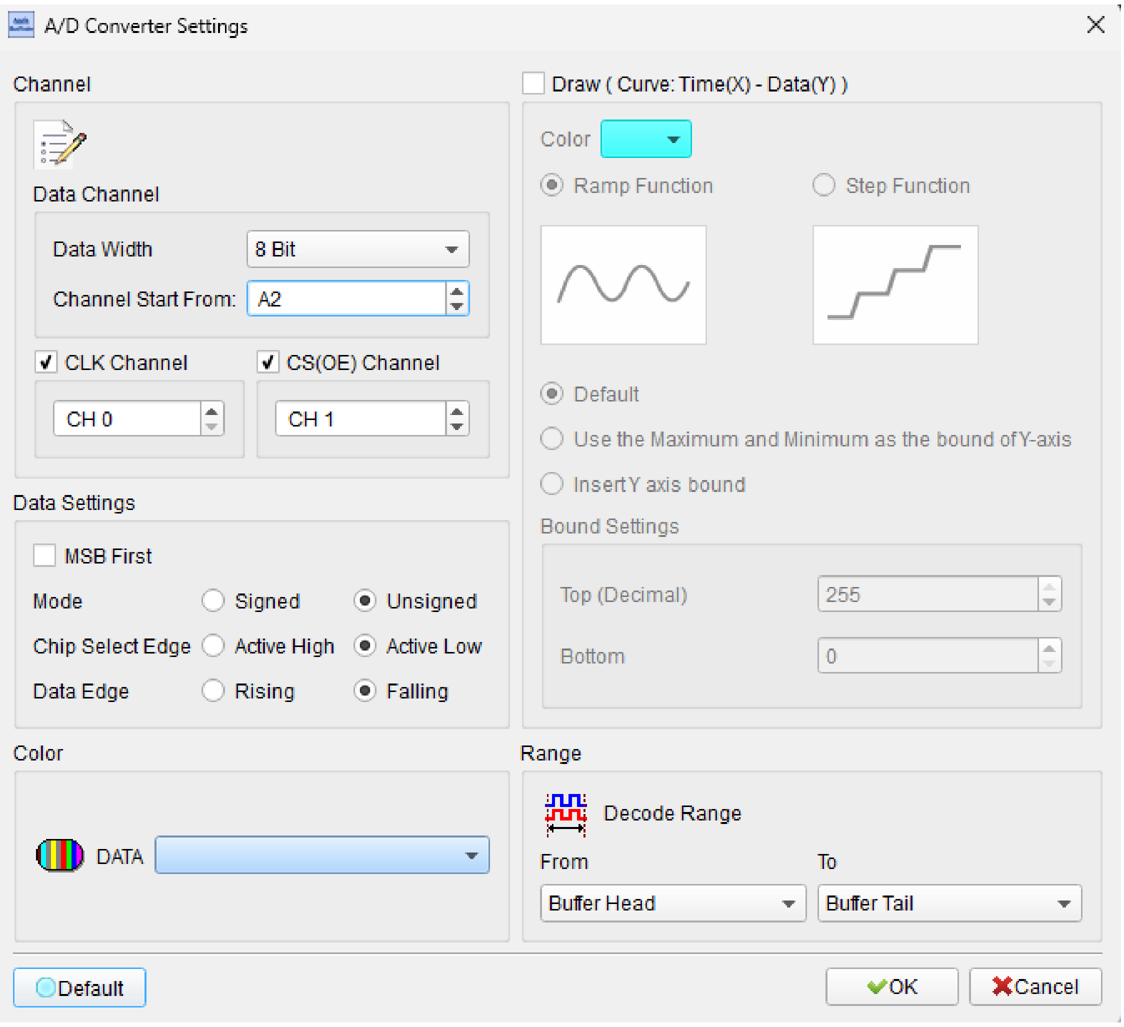


| | tamp (hh:mm:ss.ms | Character | Symbol | Data(hex) | Error |
|---|---|---|---|---|---|
| 12 | 16:57:32.397... | D18.2+ | 0100110101 | 52 | |
| 13 | 16:57:32.397... | K28.6+ | 0011110110 | DC | |
| 14 | 16:57:32.397... | K28.2+ | 1100001010 | 5C | |
| 15 | 16:57:32.397... | | 1101110101 | | Not In Table |
| 16 | 16:57:32.397... | | 0010001010 | | Not In Table |
| 17 | 16:57:32.397... | | 1110110101 | | Not In Table |
| 18 | 16:57:32.397... | | 0001001011 | | Not In Table |
| 19 | 16:57:32.397... | D0.0+ | 1001110100 | 00 | |
| 20 | 16:57:32.397... | D16.2+ | 0110110101 | 50 | |
| 21 | 16:57:32.397... | D15.1+ | 1010001001 | 2F | |
| 22 | 16:57:32.397... | D15.5+ | 1010001010 | AF | |
| 23 | 16:57:32.397... | | 1011110101 | | Not In Table |
| 24 | 16:57:32.397... | D18.2+ | 0100110101 | 52 | |
| 25 | 16:57:32.397 | K28.6+ | 0011110110 | DC | |

## A/D Converter

A/D Converter (Analog-To-Digital Converter), is a device that uses sampling to convert a continuous quantity to a discrete time representation in digital form.

## Settings



A/D Converter Settings

Channel

Data Channel

Data Width: 8 Bit

Channel Start From: A2

☑ CLK Channel: CH 0

☑ CS(OE) Channel: CH 1

Data Settings

□ MSB First

Mode: ○ Signed ● Unsigned

Chip Select Edge: ○ Active High ● Active Low

Data Edge: ○ Rising ● Falling

Color

DATA

□ Draw ( Curve: Time(X) - Data(Y) )

Color

● Ramp Function ○ Step Function

● Default

○ Use the Maximum and Minimum as the bound of Y-axis

○ Insert Y axis bound

Bound Settings

Top (Decimal): 255

Bottom: 0

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Data Channel Start From:** ADC data channel start from

**CLK Channel:** CLK IN channel of ADC, enable CLK channel and Data Edge option when checked.

**CS(OE) Channel:** Chip Select channel of ADC, enable CS channel and Chip Select Edge option when checked.

**Data Width:** ADC data width, range: 4Bit ~ 32Bit

**MSB First:** Data bit starts form MSB; LSB defaulted

**Chip Select Edge:** Set the chip select edge; Active Low defaulted

**Data Edge:** Set the Data Edge; Falling Edge defaulted

**Curve: Time(X)-Data(Y)** Show the diagram in form of time as X axis; data as Y axis.

**Ramp/Step Function:** Select Ramp/Step curve, Ramp Function defaulted

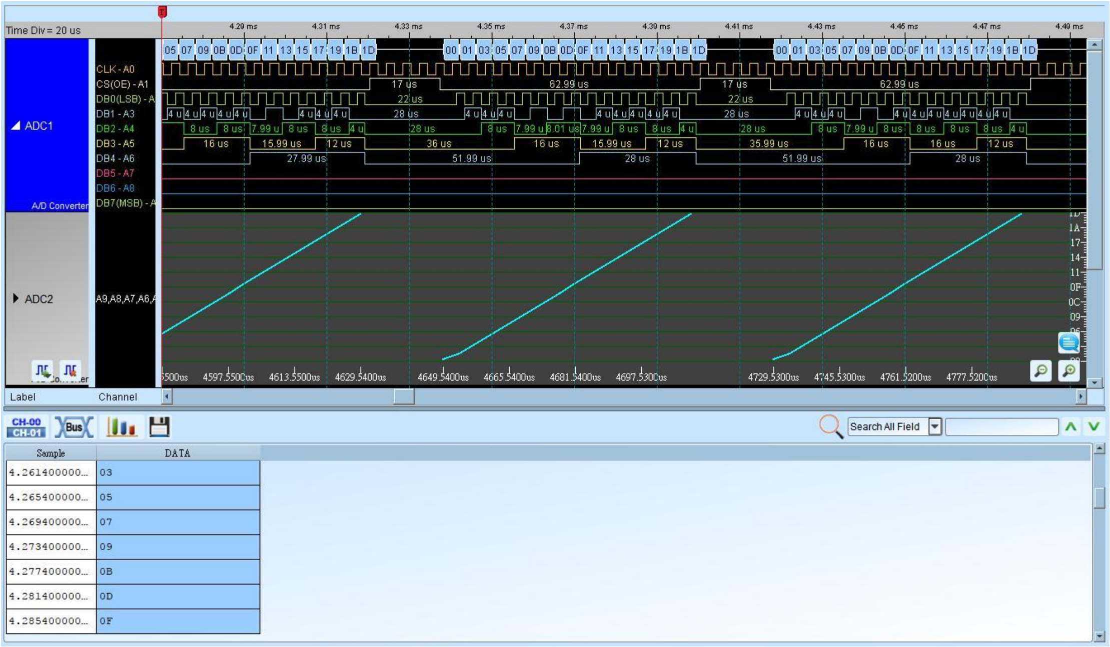
**Color:** Select the curve color

**Bound Value Range:**

> **Default:** The maximum value that can be represented using the data width is the upper bound.
>
> **Use the maximum and minimum as the bound of Y axis:** Use the maximum data as the top bound of Y axis and minimum data as the bottom bound of Y axis.
>
> **Insert Y axis bound:** Set the maximum and minimum bounds of Y axis.

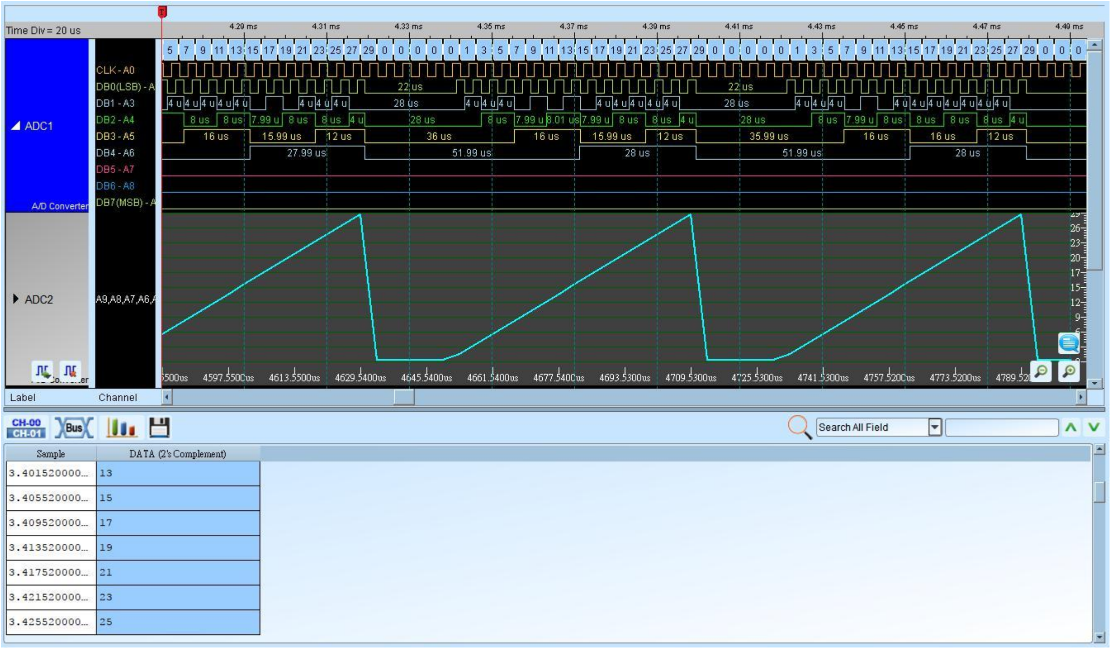
## Result

Select 8-bit data, CLK/CS channels:



| Sample | DATA |
|---|---|
| 4.261400000... | 03 |
| 4.265400000... | 05 |
| 4.269400000... | 07 |
| 4.273400000... | 09 |
| 4.277400000... | 0B |
| 4.281400000... | 0D |
| 4.285400000... | 0F |

Select 8-bit data, CLK:

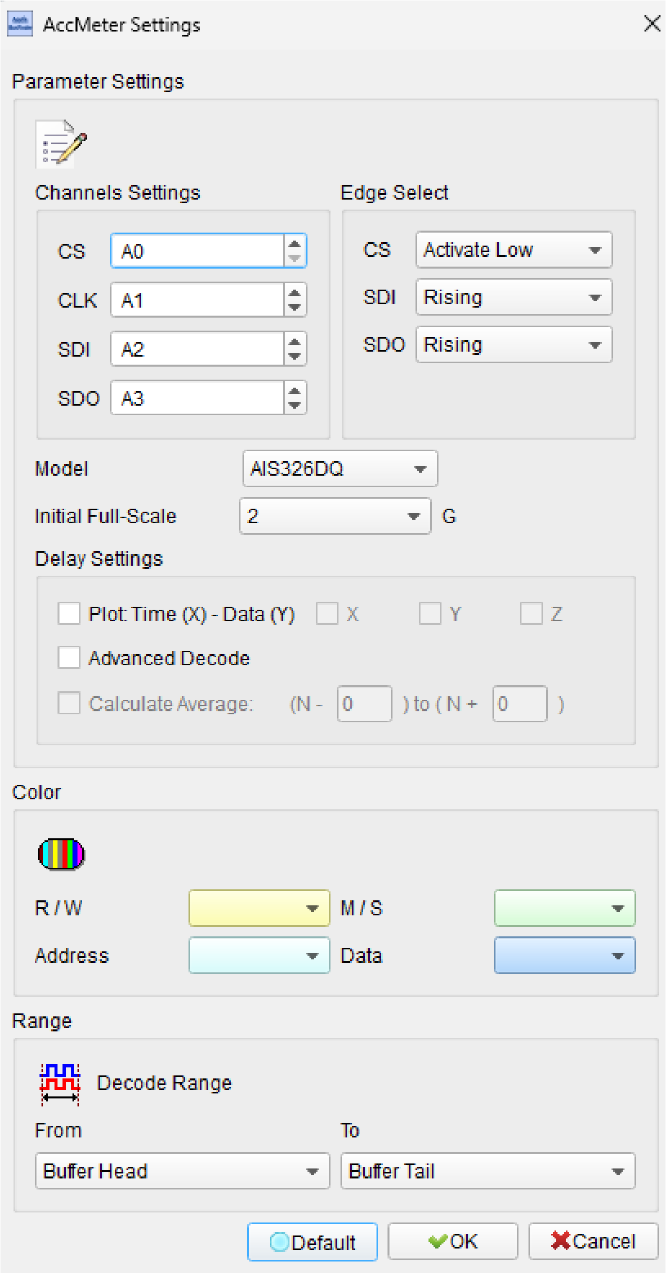


| Sample | DATA (2's Complement) |
|---|---|
| 3.401520000... | 13 |
| 3.405520000... | 15 |
| 3.409520000... | 17 |
| 3.413520000... | 19 |
| 3.417520000... | 21 |
| 3.421520000... | 23 |
| 3.425520000... | 25 |

## Accelerometer

Accelerometer (AccMeter) decoder is the SPI interfaced accelerometer data decoder, which provides bus value to acceleration value conversion and curve drawing function.

## Settings



AccMeter Settings

Parameter Settings

Channels Settings

| | |
|---|---|
| CS | A0 |
| CLK | A1 |
| SDI | A2 |
| SDO | A3 |

Edge Select

| | |
|---|---|
| CS | Activate Low |
| SDI | Rising |
| SDO | Rising |

Model AIS326DQ

Initial Full-Scale 2 G

Delay Settings

□ Plot: Time (X) - Data (Y) □ X □ Y □ Z

□ Advanced Decode

□ Calculate Average: (N - 0 ) to ( N + 0 )

Color

R / W 🟨 M / S 🟩

Address 🟦 Data 🟦

Range

Decode Range

From Buffer Head

To Buffer Tail

### Channel Setting:

**CS:** Chip Select, must specified the active state of the CS pin.

**CLK:** Clock

**SDI:** Data Input Pin, must specified the data sampling edge.

**SDO:** Data Output Pin, must specified the data sampling edge.

**Edge Setting:**

**CS:** Set the trigger edge of Chip Select to High or Low.

**SDI:** Set the trigger edge of SDI to Rising or Falling.

**SDO:** Set the trigger edge of SDO to Rising or Falling.

**Model:** The IC model of the target accelerometer.

**Initial Full-Scale:** The default Full-Scale setting.

**Display Setting, Enable when checked:**

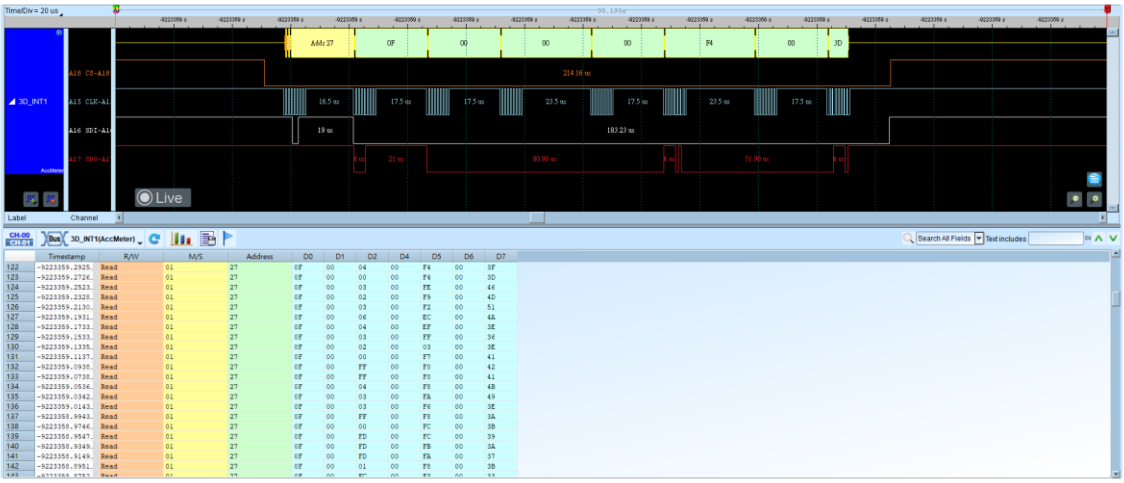
**Plot:** Enable/Disable to display the waveform in Time-Value curve.

**Advanced Decode:** Enable/Disable the address, value convert function.

**Calculate Average:** Enable/disable the average statistic function, the statistic range is ±255 data.

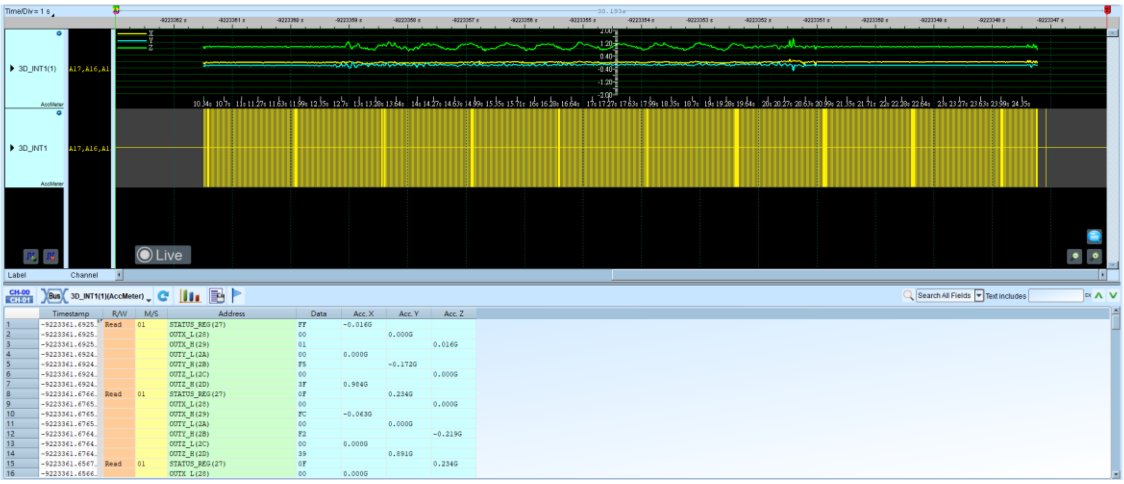
## Result

Standard decoder result



| | Timestamp | R/W | M/S | Address | D0 | D1 | D2 | D4 | D5 | D6 | D7 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 122 | -9223359.2925… | Read | 01 | 27 | 0F | 00 | 04 | 00 | F4 | 00 | 3F |
| 123 | -9223359.2726… | Read | 01 | 27 | 0F | 00 | 00 | 00 | F4 | 00 | 3D |
| 124 | -9223359.2523… | Read | 01 | 27 | 0F | 00 | 03 | 00 | FE | 00 | 46 |
| 125 | -9223359.2328… | Read | 01 | 27 | 0F | 00 | 02 | 00 | F9 | 00 | 4D |
| 126 | -9223359.2130… | Read | 01 | 27 | 0F | 00 | 03 | 00 | F2 | 00 | 51 |
| 127 | -9223359.1931… | Read | 01 | 27 | 0F | 00 | 06 | 00 | EC | 00 | 4A |
| 128 | -9223359.1733… | Read | 01 | 27 | 0F | 00 | 04 | 00 | EF | 00 | 3E |
| 129 | -9223359.1533… | Read | 01 | 27 | 0F | 00 | 03 | 00 | FF | 00 | 36 |
| 130 | -9223359.1335… | Read | 01 | 27 | 0F | 00 | 02 | 00 | 03 | 00 | 3E |
| 131 | -9223359.1137… | Read | 01 | 27 | 0F | 00 | 00 | 00 | F7 | 00 | 41 |
| 132 | -9223359.0938… | Read | 01 | 27 | 0F | 00 | FF | 00 | F0 | 00 | 42 |
| 133 | -9223359.0738… | Read | 01 | 27 | 0F | 00 | FF | 00 | F0 | 00 | 41 |
| 134 | -9223359.0536… | Read | 01 | 27 | 0F | 00 | 04 | 00 | F8 | 00 | 4B |
| 135 | -9223359.0342… | Read | 01 | 27 | 0F | 00 | 03 | 00 | FA | 00 | 49 |
| 136 | -9223359.0143… | Read | 01 | 27 | 0F | 00 | 03 | 00 | F6 | 00 | 3E |
| 137 | -9223358.9943… | Read | 01 | 27 | 0F | 00 | FF | 00 | F8 | 00 | 3A |
| 138 | -9223358.9746… | Read | 01 | 27 | 0F | 00 | 00 | 00 | FC | 00 | 3B |
| 139 | -9223358.9547… | Read | 01 | 27 | 0F | 00 | FD | 00 | FC | 00 | 39 |
| 140 | -9223358.9349… | Read | 01 | 27 | 0F | 00 | FD | 00 | FB | 00 | 3A |
| 141 | -9223358.9149… | Read | 01 | 27 | 0F | 00 | FD | 00 | FA | 00 | 37 |
| 142 | -9223358.8951… | Read | 01 | 27 | 0F | 00 | 01 | 00 | F8 | 00 | 3B |

Advanced decode result + Time-Value curve display



| | Timestamp | R/W | M/S | Address | Data | Acc. X | Acc. Y | Acc. Z |
|---|---|---|---|---|---|---|---|---|
| 1 | -9223361.6925… | Read | 01 | STATUS_REG(27) | FF | -0.016G | | |
| 2 | -9223361.6925… | | | OUTX_L(28) | 00 | | 0.000G | |
| 3 | -9223361.6925… | | | OUTX_H(29) | 01 | | | 0.016G |
| 4 | -9223361.6924… | | | OUTY_L(2A) | 00 | 0.000G | | |
| 5 | -9223361.6924… | | | OUTY_H(2B) | F5 | | -0.172G | |
| 6 | -9223361.6924… | | | OUTZ_L(2C) | 00 | | | 0.000G |
| 7 | -9223361.6924… | | | OUTZ_H(2D) | 3F | 0.984G | | |
| 8 | -9223361.6766… | Read | 01 | STATUS_REG(27) | 0F | | 0.234G | |
| 9 | -9223361.6765… | | | OUTX_L(28) | 00 | | | 0.000G |
| 10 | -9223361.6765… | | | OUTX_H(29) | FC | -0.063G | | |
| 11 | -9223361.6765… | | | OUTY_L(2A) | 00 | | 0.000G | |
| 12 | -9223361.6764… | | | OUTY_H(2B) | F2 | | | -0.219G |
| 13 | -9223361.6764… | | | OUTZ_L(2C) | 00 | 0.000G | | |
| 14 | -9223361.6764… | | | OUTZ_H(2D) | 39 | | 0.891G | |
| 15 | -9223361.6567… | Read | 01 | STATUS_REG(27) | 0F | | | 0.234G |
| 16 | -9223361.6566… | | | OUTX_L(28) | 00 | 0.000G | | |

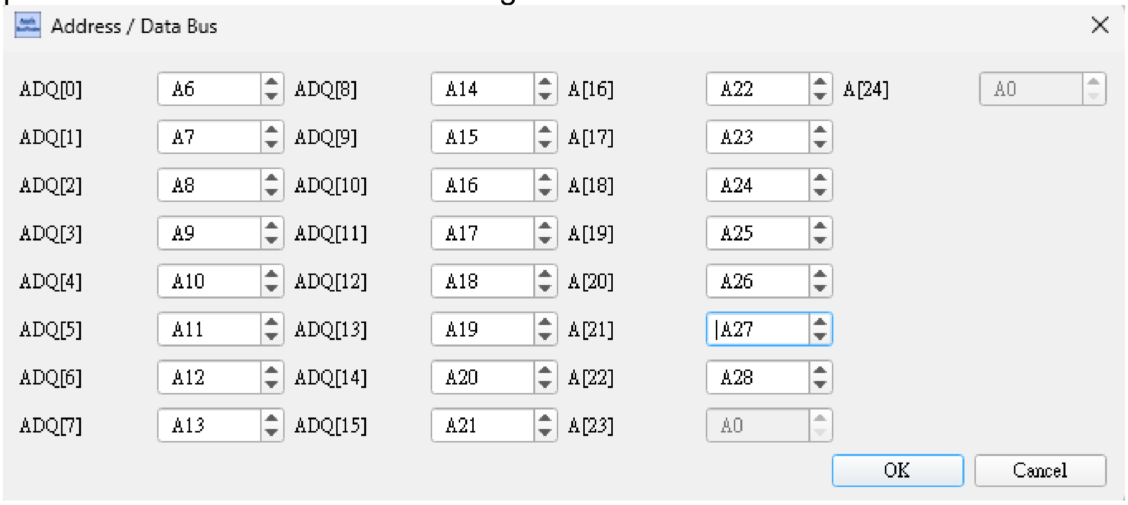
## AD-Mux Flash

AD-Mux Flash is one kind of parallel flash that utilize an Address and Data multiplexed interface.

### Settings

**Amax:** Setting the number of address pin.

**Quick Setup/User Defined:** Only set ADQ[0](LSB) when select the Quick Setup, other channels will be set automatically. When check User Defined and press the button will show the dialog below:



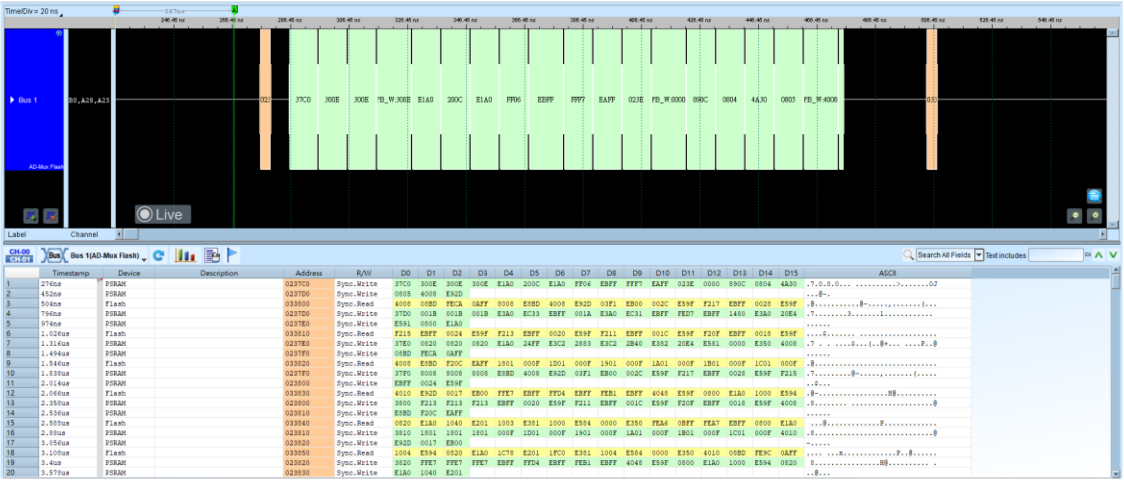
| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ADQ[0] | A6 | ADQ[8] | A14 | A[16] | A22 | A[24] | A0 |
| ADQ[1] | A7 | ADQ[9] | A15 | A[17] | A23 | | |
| ADQ[2] | A8 | ADQ[10] | A16 | A[18] | A24 | | |
| ADQ[3] | A9 | ADQ[11] | A17 | A[19] | A25 | | |
| ADQ[4] | A10 | ADQ[12] | A18 | A[20] | A26 | | |
| ADQ[5] | A11 | ADQ[13] | A19 | A[21] | A27 | | |
| ADQ[6] | A12 | ADQ[14] | A20 | A[22] | A28 | | |
| ADQ[7] | A13 | ADQ[15] | A21 | A[23] | A0 | | |

**Flash:** Control pins of flash.

**PSRAM:** Control pins of PSRAM. Some MCP include Flash and PSRAM in one package. It will decode PSRAM at the same time when "has PSRAM" is checked.

**Configuration:** The default setting of configuration register. User must set here to make a correct analysis.

### Result

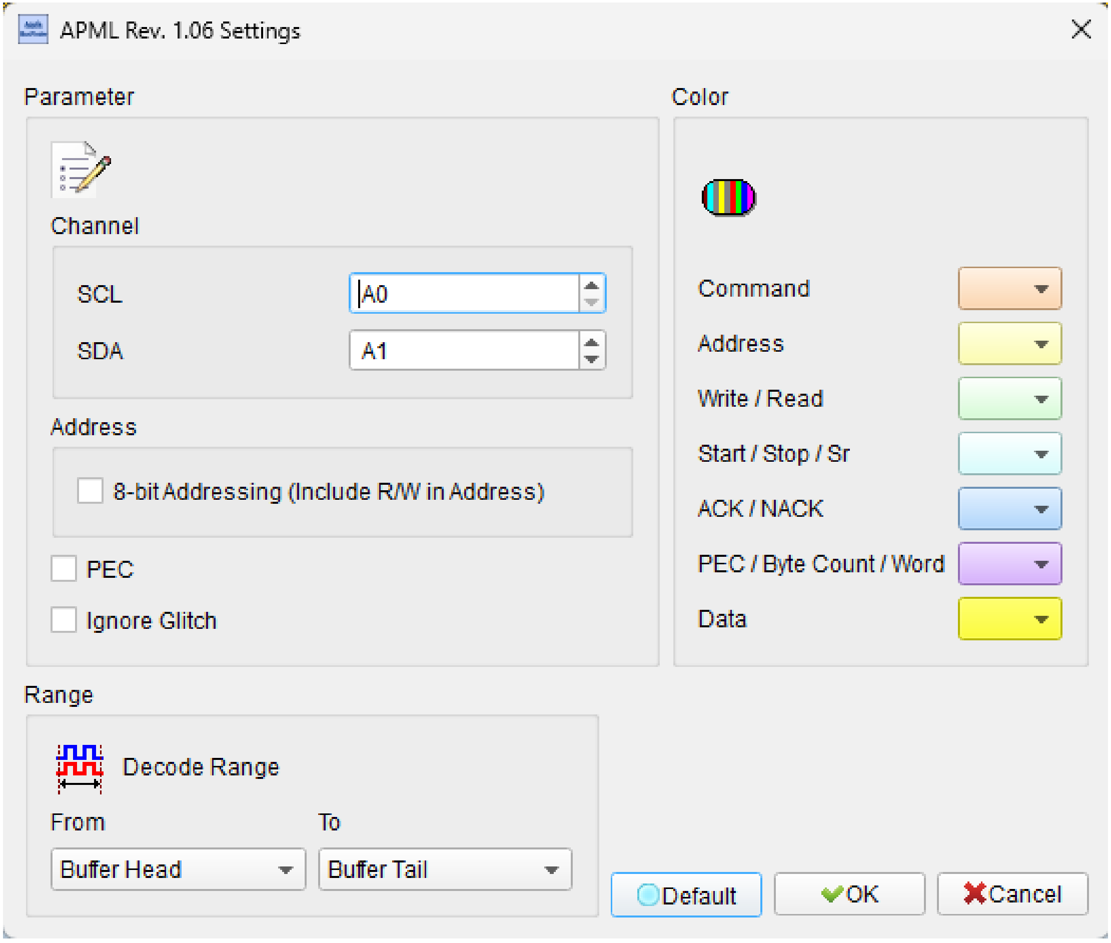


| | Timestamp | Device | Description | Address | R/W | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 276ns | PSRAM | | 0237C0 | Sync.Write | 37C0 | 300E | 300E | 300E | E1A0 | 200C | E1A0 | FF06 | EBFF | FFF7 | EAFF | 023E | 0000 | 890C | 0804 | 4A30 | .7.0.0.0... ..........>.......0J |
| 2 | 452ns | PSRAM | | 0237D0 | Sync.Write | 0805 | 4008 | E92D | | | | | | | | | | | | | | ...@-. |
| 3 | 504ns | Flash | | 033800 | Sync.Read | 4008 | 08BD | FECA | 0AFF | 8008 | E8BD | 4008 | E92D | 03F1 | EB00 | 002C | E59F | F217 | EBFF | 0028 | E59F | .@...........@-.....,.......(... |
| 4 | 796ns | PSRAM | | 0237D0 | Sync.Write | 37D0 | 001B | 001B | 001B | E3A0 | EC33 | EBFF | 001A | E3A0 | EC31 | EBFF | FED7 | EBFF | 1480 | E3A0 | 20E4 | .7........3.......1........... |
| 5 | 974ns | PSRAM | | 0237E0 | Sync.Write | E591 | 0800 | E1A0 | | | | | | | | | | | | | | ...... |
| 6 | 1.026us | Flash | | 033810 | Sync.Read | F215 | EBFF | 0024 | E59F | F213 | EBFF | 0020 | E59F | F211 | EBFF | 001C | E59F | F20F | EBFF | 0018 | E59F | ....$....... ................... |
| 7 | 1.316us | PSRAM | | 0237E0 | Sync.Write | 37E0 | 0820 | 0820 | 0820 | E1A0 | 24FF | E3C2 | 2883 | E3C2 | 2B40 | E382 | 20E4 | E581 | 0000 | E350 | 4008 | .7 . . ....$...(..@+... ....P..@ |
| 8 | 1.494us | PSRAM | | 0237F0 | Sync.Write | 08BD | FECA | 0AFF | | | | | | | | | | | | | | ...... |
| 9 | 1.546us | Flash | | 033820 | Sync.Read | 4008 | E8BD | F20C | EAFF | 1801 | 000F | 1D01 | 000F | 1901 | 000F | 1A01 | 000F | 1B01 | 000F | 1C01 | 000F | .@............................. |
| 10 | 1.838us | PSRAM | | 0237F0 | Sync.Write | 37F0 | 8008 | 8008 | 8008 | E8BD | 4008 | E92D | 03F1 | EB00 | 002C | E59F | F217 | EBFF | 0028 | E59F | F215 | .7.........@-.....,.......(..... |
| 11 | 2.014us | PSRAM | | 023800 | Sync.Write | EBFF | 0024 | E59F | | | | | | | | | | | | | | ..$... |
| 12 | 2.066us | Flash | | 033830 | Sync.Read | 4010 | E92D | 0017 | EB00 | FFE7 | EBFF | FFD4 | EBFF | FEB1 | EBFF | 4048 | E59F | 0800 | E1A0 | 1000 | E594 | .@-................H@......... |
| 13 | 2.358us | PSRAM | | 023800 | Sync.Write | 3800 | F213 | F213 | F213 | EBFF | 0020 | E59F | F211 | EBFF | 001C | E59F | F20F | EBFF | 0018 | E59F | 4008 | .8........ ....................@ |
| 14 | 2.536us | PSRAM | | 023810 | Sync.Write | E8BD | F20C | EAFF | | | | | | | | | | | | | | ...... |
| 15 | 2.588us | Flash | | 033840 | Sync.Read | 0820 | E1A0 | 1040 | E201 | 1003 | E381 | 1000 | E584 | 0000 | E350 | FEA6 | 0BFF | FEA7 | EBFF | 0800 | E1A0 | ...@.............P............. |
| 16 | 2.88us | PSRAM | | 023810 | Sync.Write | 3810 | 1801 | 1801 | 1801 | 000F | 1D01 | 000F | 1901 | 000F | 1A01 | 000F | 1B01 | 000F | 1C01 | 000F | 4010 | .8..............................@ |
| 17 | 3.056us | PSRAM | | 023820 | Sync.Write | E92D | 0017 | EB00 | | | | | | | | | | | | | | -..... |
| 18 | 3.108us | Flash | | 033850 | Sync.Read | 1004 | E594 | 0820 | E1A0 | 1C78 | E201 | 1FC0 | E381 | 1004 | E584 | 0000 | E350 | 4010 | 08BD | FE9C | 0AFF | .... ...x............P..@...... |
| 19 | 3.4us | PSRAM | | 023820 | Sync.Write | 3820 | FFE7 | FFE7 | FFE7 | EBFF | FFD4 | EBFF | FEB1 | EBFF | 4048 | E59F | 0800 | E1A0 | 1000 | E594 | 0820 | 8................H@.......... . |
| 20 | 3.578us | PSRAM | | 023830 | Sync.Write | E1A0 | 1040 | E201 | | | | | | | | | | | | | | ..@... |

## APML

APML protocol is established by AMD for it's Opteron CPU platform.

## Settings



SCL: A0

SDA: A1

□ 8-bit Addressing (Include R/W in Address)

□ PEC

□ Ignore Glitch

Command ▾
Address ▾
Write / Read ▾
Start / Stop / Sr ▾
ACK / NACK ▾
PEC / Byte Count / Word ▾
Data ▾

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

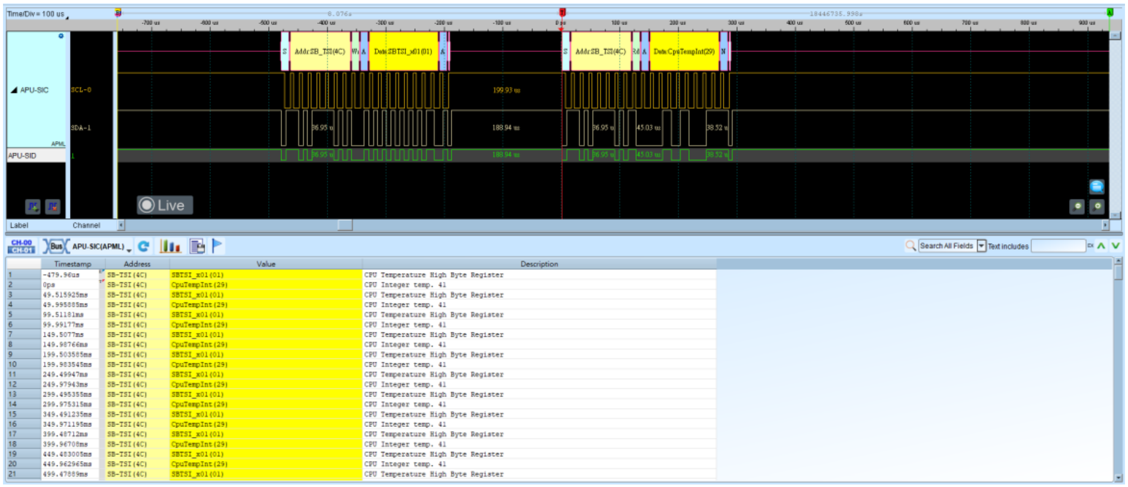
**8-bit addressing (Include R/W in Address):** Show 8-bit addressing (include

7-bit addressing and 1-bit R/W). Enable when checked.

**PEC:** Packet Error Check. Enable when checked.

**Ignore glitch:** Ignore the glitch when the slow transitions. Enable when checked.

## Result

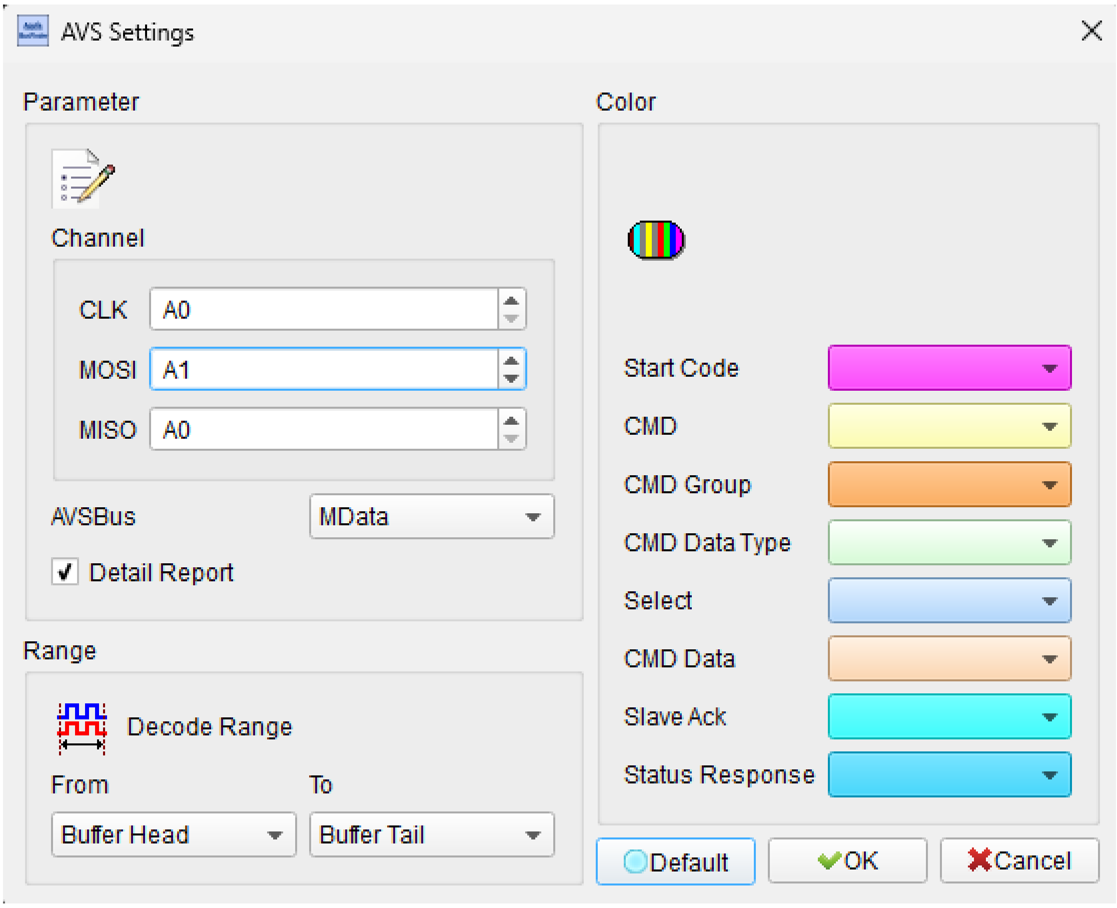


| | Timestamp | Address | Value | Description |
|---|---|---|---|---|
| 1 | -479.96us | SB-TSI(4C) | SBTSI_x01(01) | CPU Temperature High Byte Register |
| 2 | 0ps | SB-TSI(4C) | CpuTempInt(29) | CPU Integer temp. 41 |
| 3 | 49.515925ms | SB-TSI(4C) | SBTSI_x01(01) | CPU Temperature High Byte Register |
| 4 | 49.995885ms | SB-TSI(4C) | CpuTempInt(29) | CPU Integer temp. 41 |
| 5 | 99.51181ms | SB-TSI(4C) | SBTSI_x01(01) | CPU Temperature High Byte Register |
| 6 | 99.99177ms | SB-TSI(4C) | CpuTempInt(29) | CPU Integer temp. 41 |
| 7 | 149.5077ms | SB-TSI(4C) | SBTSI_x01(01) | CPU Temperature High Byte Register |
| 8 | 149.98766ms | SB-TSI(4C) | CpuTempInt(29) | CPU Integer temp. 41 |
| 9 | 199.503585ms | SB-TSI(4C) | SBTSI_x01(01) | CPU Temperature High Byte Register |
| 10 | 199.983545ms | SB-TSI(4C) | CpuTempInt(29) | CPU Integer temp. 41 |
| 11 | 249.49947ms | SB-TSI(4C) | SBTSI_x01(01) | CPU Temperature High Byte Register |
| 12 | 249.97943ms | SB-TSI(4C) | CpuTempInt(29) | CPU Integer temp. 41 |
| 13 | 299.495355ms | SB-TSI(4C) | SBTSI_x01(01) | CPU Temperature High Byte Register |
| 14 | 299.975315ms | SB-TSI(4C) | CpuTempInt(29) | CPU Integer temp. 41 |
| 15 | 349.491235ms | SB-TSI(4C) | SBTSI_x01(01) | CPU Temperature High Byte Register |
| 16 | 349.971195ms | SB-TSI(4C) | CpuTempInt(29) | CPU Integer temp. 41 |
| 17 | 399.48712ms | SB-TSI(4C) | SBTSI_x01(01) | CPU Temperature High Byte Register |
| 18 | 399.96708ms | SB-TSI(4C) | CpuTempInt(29) | CPU Integer temp. 41 |
| 19 | 449.483005ms | SB-TSI(4C) | SBTSI_x01(01) | CPU Temperature High Byte Register |
| 20 | 449.962965ms | SB-TSI(4C) | CpuTempInt(29) | CPU Integer temp. 41 |
| 21 | 499.47889ms | SB-TSI(4C) | SBTSI_x01(01) | CPU Temperature High Byte Register |

## AVSBus

AVS stands for Audio-Visual Coding Standard, which is a video and audio coding standard used to compress, transmit and decode digital video and audio data.

### Settings



AVS Settings

Parameter

Channel

CLK: A0
MOSI: A1
MISO: A0

AVSBus: MData

☑ Detail Report

Range

Decode Range

From: Buffer Head
To: Buffer Tail

Color

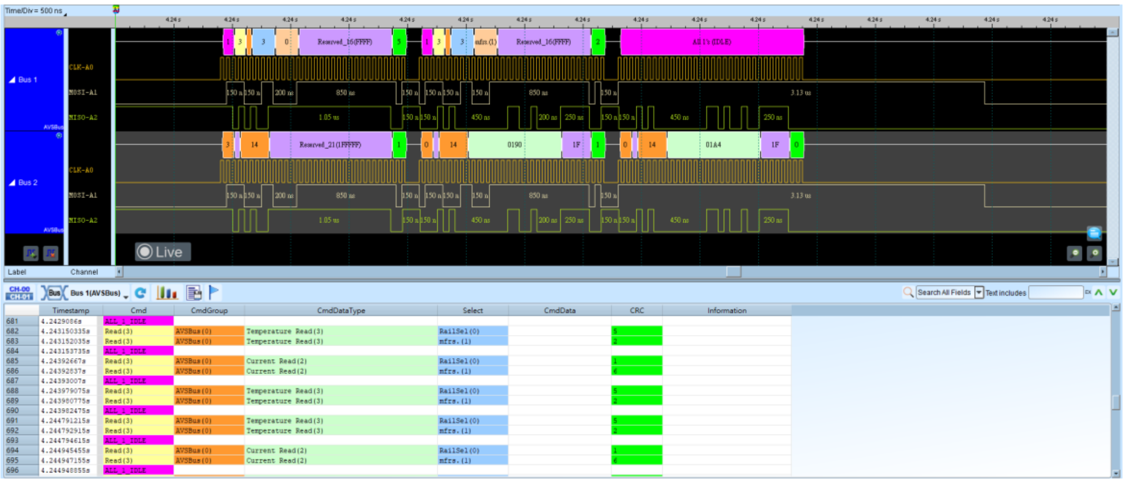
Start Code
CMD
CMD Group
CMD Data Type
Select
CMD Data
Slave Ack
Status Response

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**AVSBus:** Set the decoded data type to SData or MData.

**Detail Report:** Show details in report area, enabled when checked.

## Result

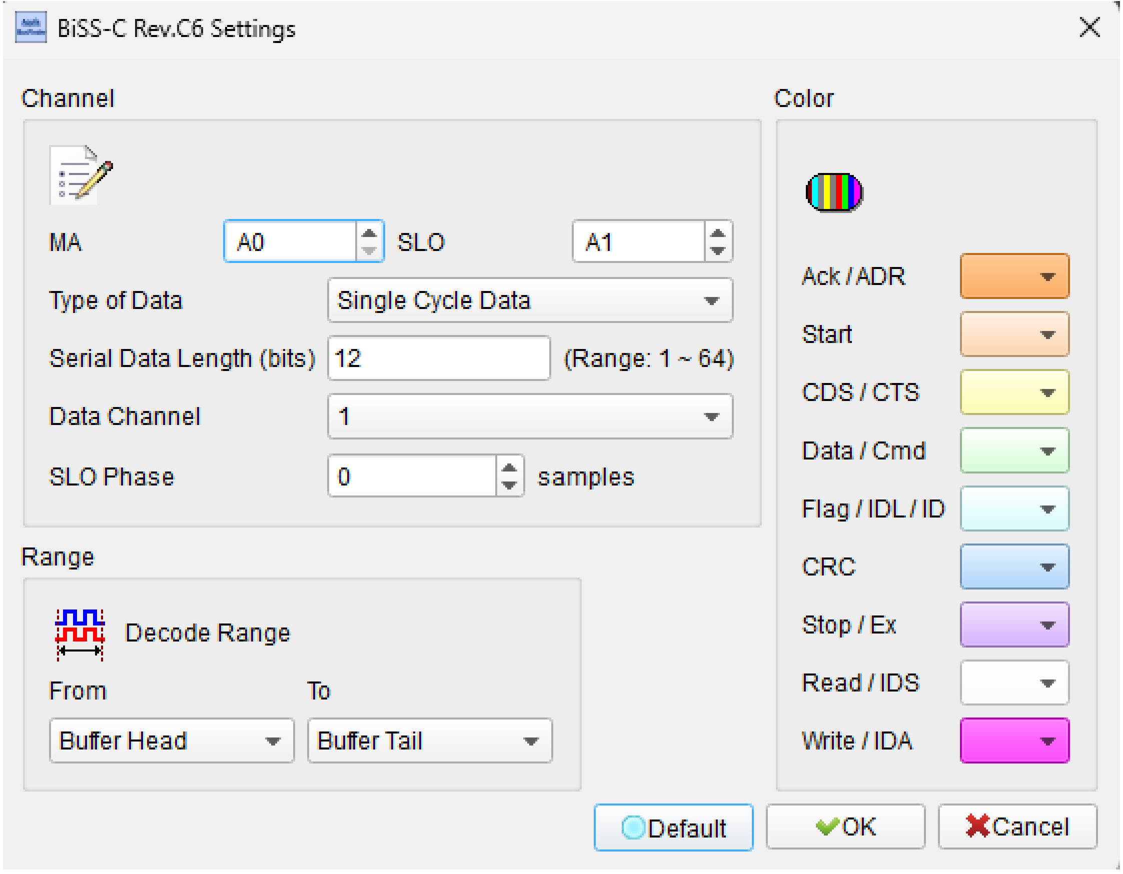


| | Timestamp | Cmd | CmdGroup | CmdDataType | Select | CmdData | CRC | Information |
|---|---|---|---|---|---|---|---|---|
| 681 | 4.2429086s | ALL_1_IDLE | | | | | | |
| 682 | 4.243150335s | Read(3) | AVSBus(0) | Temperature Read(3) | RailSel(0) | | 5 | |
| 683 | 4.243152035s | Read(3) | AVSBus(0) | Temperature Read(3) | mfrs.(1) | | 2 | |
| 684 | 4.243153735s | ALL_1_IDLE | | | | | | |
| 685 | 4.24392667s | Read(3) | AVSBus(0) | Current Read(2) | RailSel(0) | | 1 | |
| 686 | 4.24392837s | Read(3) | AVSBus(0) | Current Read(2) | mfrs.(1) | | 6 | |
| 687 | 4.24393007s | ALL_1_IDLE | | | | | | |
| 688 | 4.243979075s | Read(3) | AVSBus(0) | Temperature Read(3) | RailSel(0) | | 5 | |
| 689 | 4.243980775s | Read(3) | AVSBus(0) | Temperature Read(3) | mfrs.(1) | | 2 | |
| 690 | 4.243982475s | ALL_1_IDLE | | | | | | |
| 691 | 4.244791215s | Read(3) | AVSBus(0) | Temperature Read(3) | RailSel(0) | | 5 | |
| 692 | 4.244792915s | Read(3) | AVSBus(0) | Temperature Read(3) | mfrs.(1) | | 2 | |
| 693 | 4.244794615s | ALL_1_IDLE | | | | | | |
| 694 | 4.244945455s | Read(3) | AVSBus(0) | Current Read(2) | RailSel(0) | | 1 | |
| 695 | 4.244947155s | Read(3) | AVSBus(0) | Current Read(2) | mfrs.(1) | | 6 | |
| 696 | 4.244948855s | ALL_1_IDLE | | | | | | |

## BiSS-C

BiSS-C (Bidirectional Synchronous Serial C-mode) designed by Ic-Haus. The BiSS Interface is based on a protocol which implements a real time interface. It enables a digital, serial and secure communication between controller, sensor and actuator. It is used in industrial applications which require transfer rates, safety, flexibility and a minimized implementation effort.

## Settings



BiSS-C Rev.C6 Settings

Channel

MA: A0 SLO: A1

Type of Data: Single Cycle Data

Serial Data Length (bits): 12 (Range: 1 ~ 64)

Data Channel: 1

SLO Phase: 0 samples

Range

Decode Range

From: Buffer Head To: Buffer Tail

Color

Ack / ADR, Start, CDS / CTS, Data / Cmd, Flag / IDL / ID, CRC, Stop / Ex, Read / IDS, Write / IDA

**MA/SLO:** Setting the channel of MA and SLO.

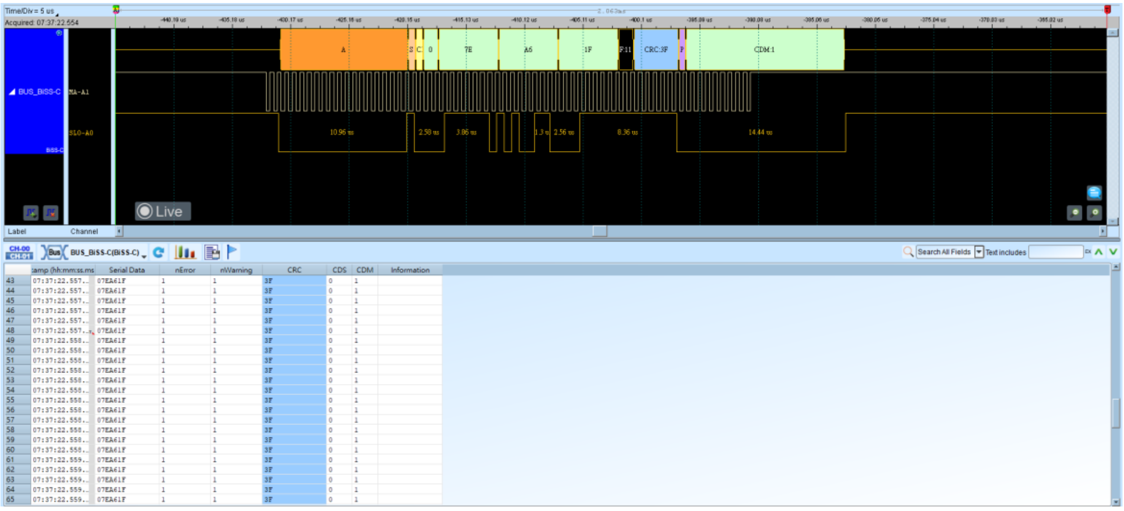
**Type of data:** Setting the type you want to decode. It include “Register Data-CDM”, “Register Data-CDS”, “Single Cycle Data”.

**Serial data length(bits):** Setting the data length when Single Cycle Data mode.

**Data Channel:** Startup setting, users need to provide the number of slaves as the base information for decoding.

**SLO Phase:** Sets the delay phase of the SLO.

## Result

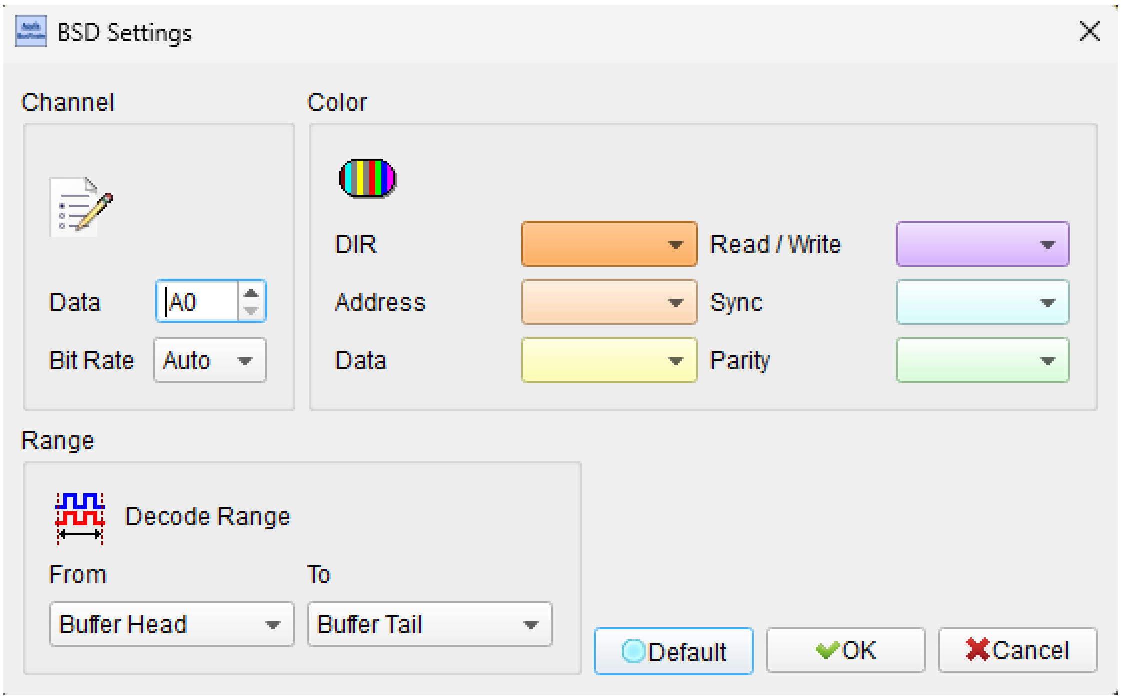


| | amp (hh:mm:ss.ms | Serial Data | nError | nWarning | CRC | CDS | CDM | Information |
|---|---|---|---|---|---|---|---|---|
| 43 | 07:37:22.557... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 44 | 07:37:22.557... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 45 | 07:37:22.557... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 46 | 07:37:22.557... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 47 | 07:37:22.557... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 48 | 07:37:22.557... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 49 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 50 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 51 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 52 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 53 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 54 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 55 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 56 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 57 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 58 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 59 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 60 | 07:37:22.558... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 61 | 07:37:22.559... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 62 | 07:37:22.559... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 63 | 07:37:22.559... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 64 | 07:37:22.559... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |
| 65 | 07:37:22.559... | 07EA61F | 1 | 1 | 3F | 0 | 1 | |

## BSD

BSD(Bit Serial Device) is a serial communications protocol for battery monitoring in automotive application.

### Settings



BSD Settings

Channel

Data: A0

Bit Rate: Auto

Color

DIR | Read / Write

Address | Sync

Data | Parity

Range

Decode Range

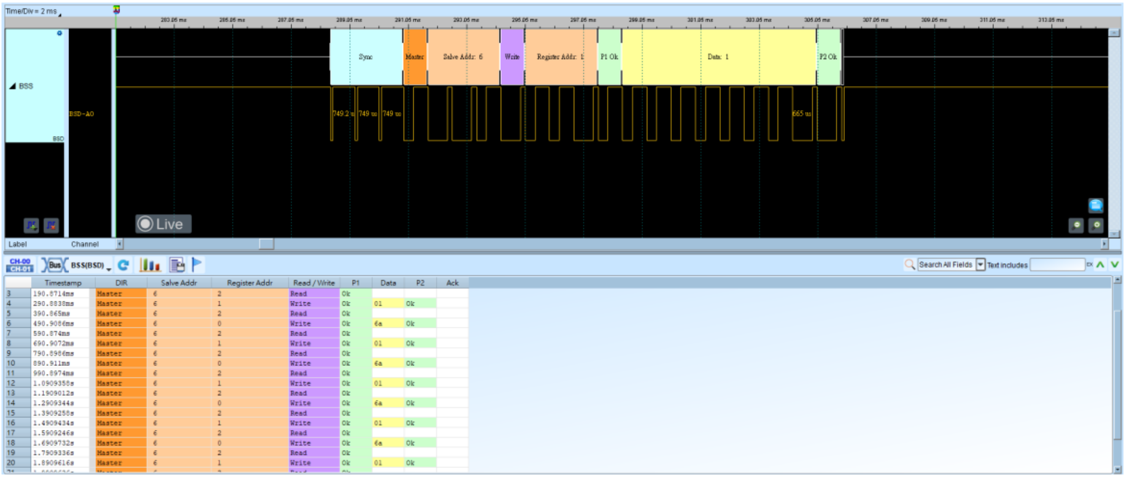
From: Buffer Head

To: Buffer Tail

**Data:** The BSD data.

**Bit rate:** The bit rate of the BSD data

### Result

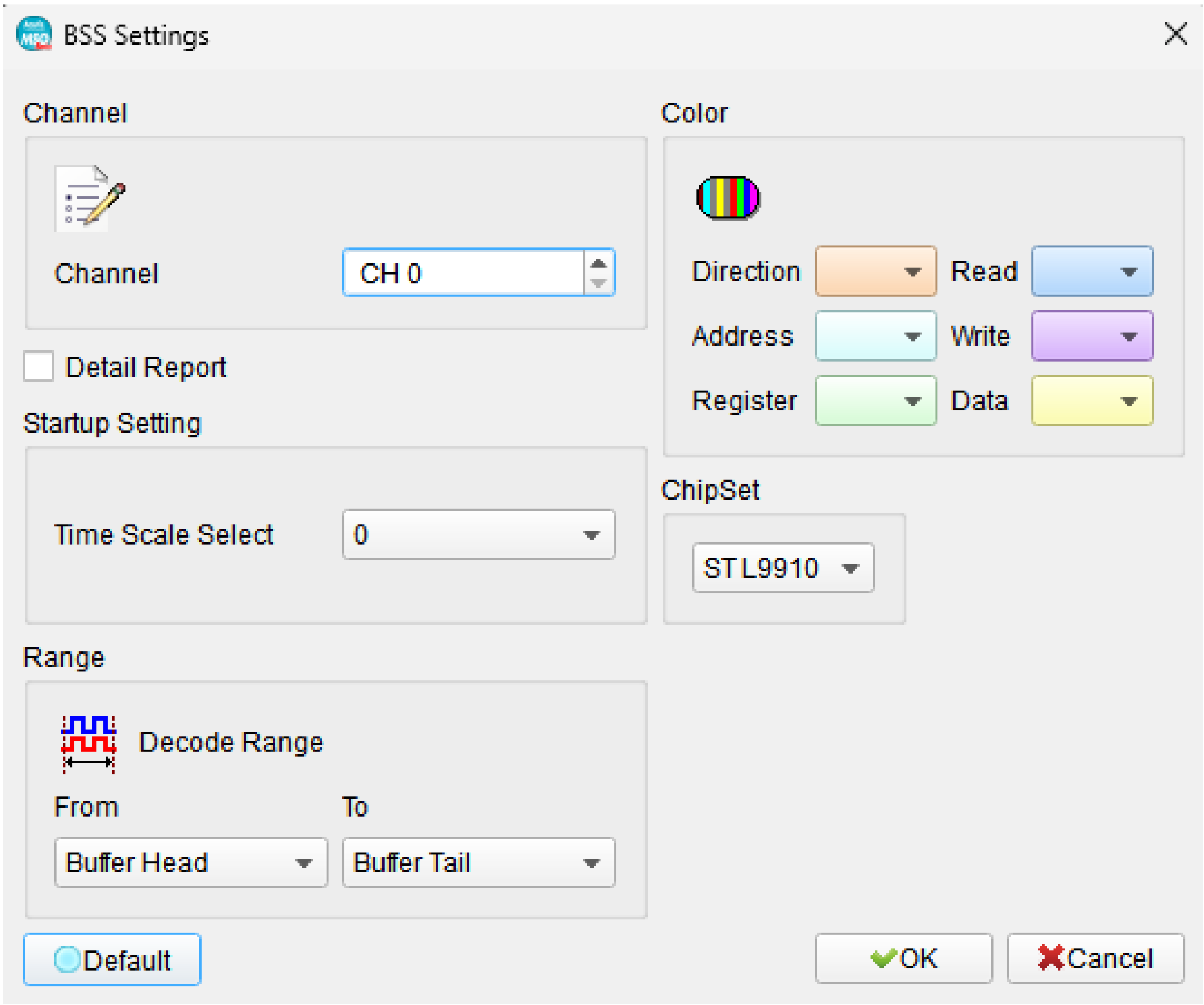


| | Timestamp | DIR | Salve Addr | Register Addr | Read / Write | P1 | Data | P2 | Ack |
|---|---|---|---|---|---|---|---|---|---|
| 3 | 190.8714ms | Master | 6 | 2 | Read | Ok | | | |
| 4 | 290.8838ms | Master | 6 | 1 | Write | Ok | 01 | Ok | |
| 5 | 390.865ms | Master | 6 | 2 | Read | Ok | | | |
| 6 | 490.9086ms | Master | 6 | 0 | Write | Ok | 6a | Ok | |
| 7 | 590.874ms | Master | 6 | 2 | Read | Ok | | | |
| 8 | 690.9072ms | Master | 6 | 1 | Write | Ok | 01 | Ok | |
| 9 | 790.8986ms | Master | 6 | 2 | Read | Ok | | | |
| 10 | 890.911ms | Master | 6 | 0 | Write | Ok | 6a | Ok | |
| 11 | 990.8974ms | Master | 6 | 2 | Read | Ok | | | |
| 12 | 1.0909358s | Master | 6 | 1 | Write | Ok | 01 | Ok | |
| 13 | 1.1909012s | Master | 6 | 2 | Read | Ok | | | |
| 14 | 1.2909344s | Master | 6 | 0 | Write | Ok | 6a | Ok | |
| 15 | 1.3909258s | Master | 6 | 2 | Read | Ok | | | |
| 16 | 1.4909434s | Master | 6 | 1 | Write | Ok | 01 | Ok | |
| 17 | 1.5909246s | Master | 6 | 2 | Read | Ok | | | |
| 18 | 1.6909732s | Master | 6 | 0 | Write | Ok | 6a | Ok | |
| 19 | 1.7909336s | Master | 6 | 2 | Read | Ok | | | |
| 20 | 1.8909616s | Master | 6 | 1 | Write | Ok | 01 | Ok | |

## BSS

Bit Synchronous Single-wire interface (BSS), direct link with the onboard Engine Control Unit (ECU). Aim of the BSS is real time control (through digital protocol) of main electrical parameters, such as voltage regulation and load response timing.

### Settings



BSS Settings

Channel

Channel: CH 0

□ Detail Report

Startup Setting

Time Scale Select: 0

Range

Decode Range

From: Buffer Head

To: Buffer Tail

Color

Direction, Read, Address, Write, Register, Data

ChipSet: STL9910

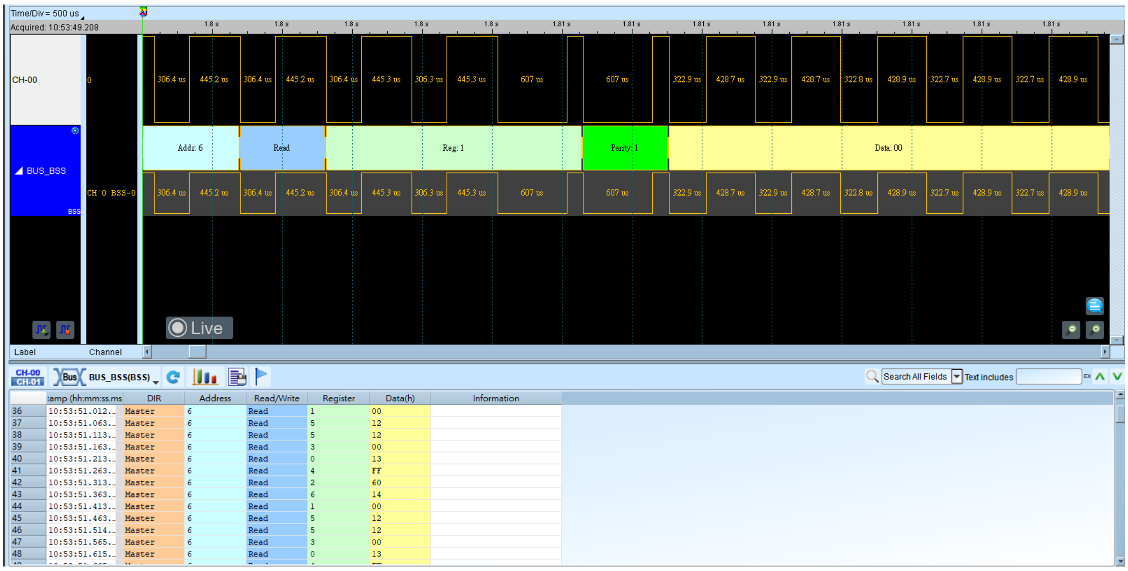
**Channel:** Set the signal channel of BSS.

**Detail Report:** Detailed analysis of the byte content. Enable when checked.

**Startup Setting:** Set the value of the Time Scale Select Bit

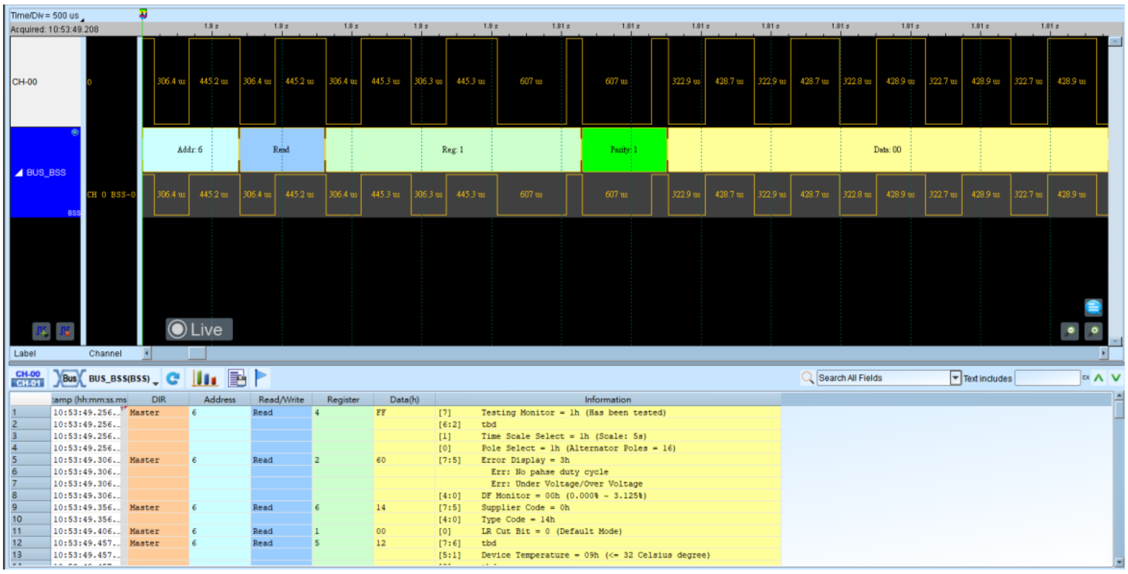
**ChipSet:** Set the Chipset that are going to be analyzed.

## Result



| | tamp (hh:mm:ss.ms | DIR | Address | Read/Write | Register | Data(h) | Information |
|---|---|---|---|---|---|---|---|
| 36 | 10:53:51.012... | Master | 6 | Read | 1 | 00 | |
| 37 | 10:53:51.063... | Master | 6 | Read | 5 | 12 | |
| 38 | 10:53:51.113... | Master | 6 | Read | 5 | 12 | |
| 39 | 10:53:51.163... | Master | 6 | Read | 3 | 00 | |
| 40 | 10:53:51.213... | Master | 6 | Read | 0 | 13 | |
| 41 | 10:53:51.263... | Master | 6 | Read | 4 | FF | |
| 42 | 10:53:51.313... | Master | 6 | Read | 2 | 60 | |
| 43 | 10:53:51.363... | Master | 6 | Read | 6 | 14 | |
| 44 | 10:53:51.413... | Master | 6 | Read | 1 | 00 | |
| 45 | 10:53:51.463... | Master | 6 | Read | 5 | 12 | |
| 46 | 10:53:51.514... | Master | 6 | Read | 5 | 12 | |
| 47 | 10:53:51.565... | Master | 6 | Read | 3 | 00 | |
| 48 | 10:53:51.615... | Master | 6 | Read | 0 | 13 | |

## Detail Report



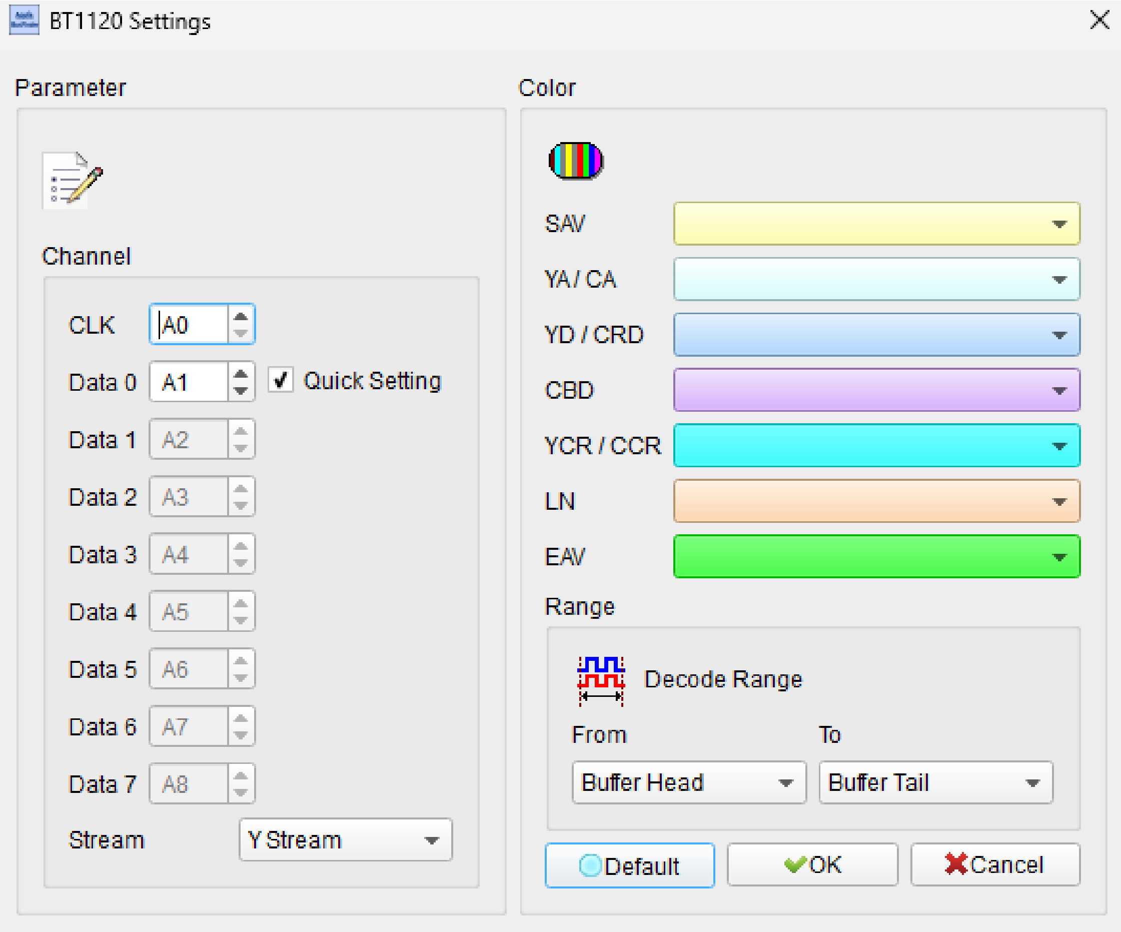
| | tamp (hh:mm:ss.ms | DIR | Address | Read/Write | Register | Data(h) | Information |
|---|---|---|---|---|---|---|---|
| 1 | 10:53:49.256... | Master | 6 | Read | 4 | FF | [7] Testing Monitor = 1h (Has been tested) |
| 2 | 10:53:49.256... | | | | | | [6:2] tbd |
| 3 | 10:53:49.256... | | | | | | [1] Time Scale Select = 1h (Scale: 5s) |
| 4 | 10:53:49.256... | | | | | | [0] Pole Select = 1h (Alternator Poles = 16) |
| 5 | 10:53:49.306... | Master | 6 | Read | 2 | 60 | [7:5] Error Display = 3h |
| 6 | 10:53:49.306... | | | | | | Err: No pahse duty cycle |
| 7 | 10:53:49.306... | | | | | | Err: Under Voltage/Over Voltage |
| 8 | 10:53:49.306... | | | | | | [4:0] DF Monitor = 00h (0.000% ~ 3.125%) |
| 9 | 10:53:49.356... | Master | 6 | Read | 6 | 14 | [7:5] Supplier Code = 0h |
| 10 | 10:53:49.356... | | | | | | [4:0] Type Code = 14h |
| 11 | 10:53:49.406... | Master | 6 | Read | 1 | 00 | [0] LR Cut Bit = 0 (Default Mode) |
| 12 | 10:53:49.457... | Master | 6 | Read | 5 | 12 | [7:6] tbd |
| 13 | 10:53:49.457... | | | | | | [5:1] Device Temperature = 09h (<= 32 Celsius degree) |

## BT1120

Digital interfaces for HDTV studio signals

The BT1120 is used for bit serial port transmission of HDTV signals. It mainly provides image format parameters and data transmission signals for HDTV production and international program exchange, and is backward compatible with old image frequencies of 60, 50, 30, 25 24Hz (progressive, interlaced, frame segmentation), total line number 1125, valid line number 1080 to cover both commercial and developing products. This interface will include all equipment necessary for broadcast and industrial applications.

### Settings



BT1120 Settings

Parameter

Channel

| | |
|---|---|
| CLK | A0 |
| Data 0 | A1 |
| Data 1 | A2 |
| Data 2 | A3 |
| Data 3 | A4 |
| Data 4 | A5 |
| Data 5 | A6 |
| Data 6 | A7 |
| Data 7 | A8 |
| Stream | Y Stream |

☑ Quick Setting

Color

SAV, YA / CA, YD / CRD, CBD, YCR / CCR, LN, EAV

Range

Decode Range

From: Buffer Head

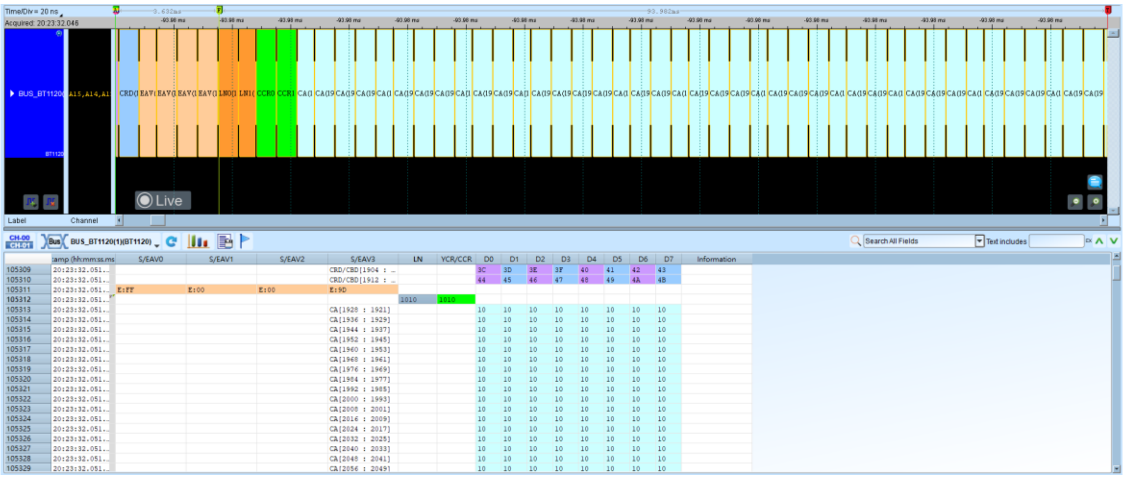
To: Buffer Tail

**Channel setting:** Set the object to be tested, CLK, each Data 0-7, the channel number connected to the logic analyzer,

**Quick Setting:** the Data channel setting will be automatically incremented.

**Stream:** Y, CB/CR stream

## Result



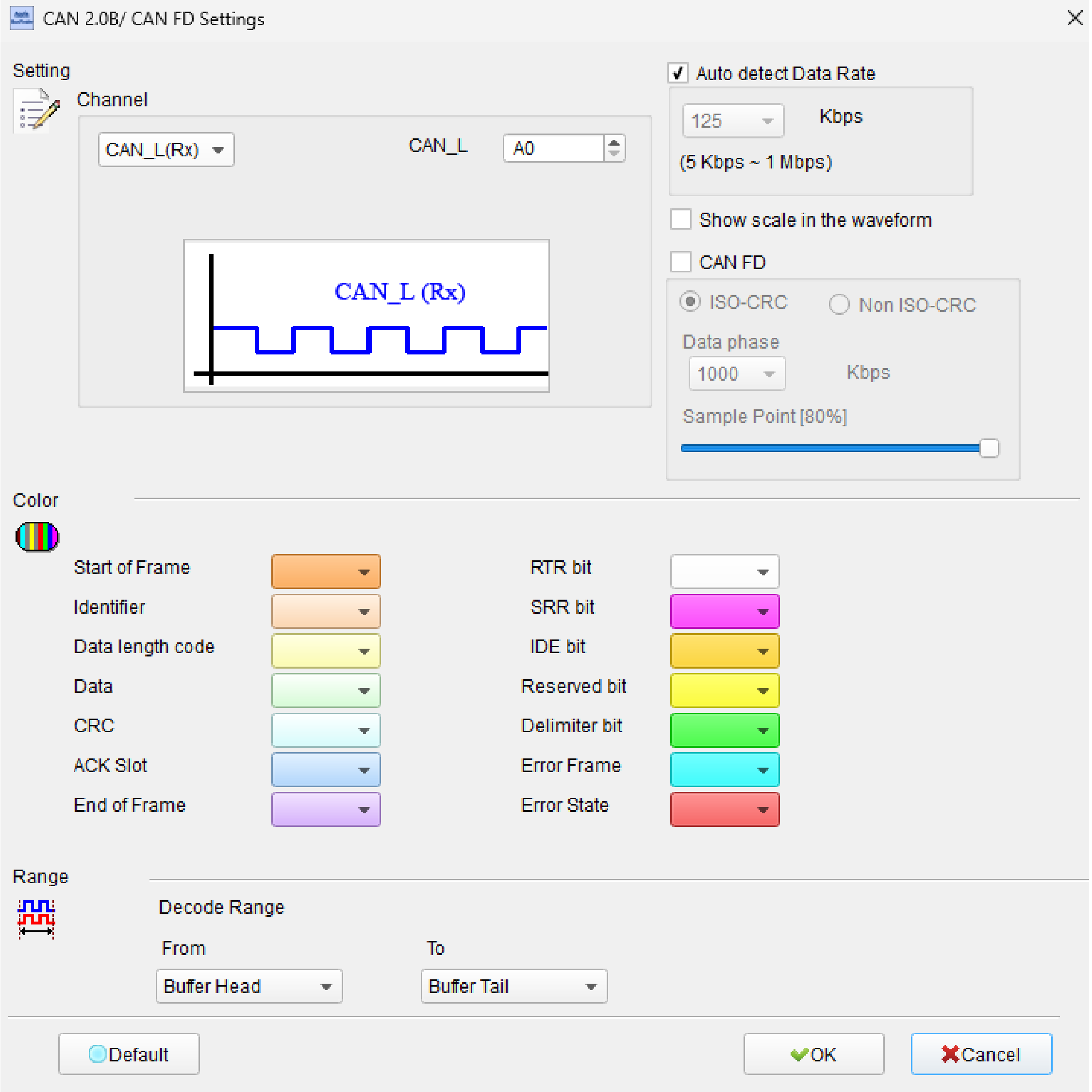
| | tamp (hh:mm:ss.ms | S/EAV0 | S/EAV1 | S/EAV2 | S/EAV3 | LN | YCR/CCR | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 105309 | 20:23:32.051... | | | | CRD/CBD[1904 : ... | | | 3C | 3D | 3E | 3F | 40 | 41 | 42 | 43 | |
| 105310 | 20:23:32.051... | | | | CRD/CBD[1912 : ... | | | 44 | 45 | 46 | 47 | 48 | 49 | 4A | 4B | |
| 105311 | 20:23:32.051... | E:FF | E:00 | E:00 | E:9D | | | | | | | | | | | |
| 105312 | 20:23:32.051... | | | | | 1010 | 1010 | | | | | | | | | |
| 105313 | 20:23:32.051... | | | | CA[1928 : 1921] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105314 | 20:23:32.051... | | | | CA[1936 : 1929] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105315 | 20:23:32.051... | | | | CA[1944 : 1937] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105316 | 20:23:32.051... | | | | CA[1952 : 1945] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105317 | 20:23:32.051... | | | | CA[1960 : 1953] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105318 | 20:23:32.051... | | | | CA[1968 : 1961] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105319 | 20:23:32.051... | | | | CA[1976 : 1969] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105320 | 20:23:32.051... | | | | CA[1984 : 1977] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105321 | 20:23:32.051... | | | | CA[1992 : 1985] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105322 | 20:23:32.051... | | | | CA[2000 : 1993] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105323 | 20:23:32.051... | | | | CA[2008 : 2001] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105324 | 20:23:32.051... | | | | CA[2016 : 2009] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105325 | 20:23:32.051... | | | | CA[2024 : 2017] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105326 | 20:23:32.051... | | | | CA[2032 : 2025] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105327 | 20:23:32.051... | | | | CA[2040 : 2033] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105328 | 20:23:32.051... | | | | CA[2048 : 2041] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |
| 105329 | 20:23:32.051... | | | | CA[2056 : 2049] | | | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | |

## CAN 2.0B/ CAN FD

The Controller Area Network (CAN) protocol has version 2.0A (Basic CAN, 11 bits) and version 2.0B (Extended CAN or Peli CAN, 29 bits); both versions have four message types: Data Frame, Remote Frame, Error Frame and Overload Frame as the diagrams below. The CAN Bus has two kinds of data output: CAN High (CAN_H) and CAN Low (CAN_L).

The data rate is flexible in CAN FD (CAN with Flexible Data-Rate). When CAN FD is transferring, it is 64 (bytes/per data) and including CRC17/CRC21.

### Settings



CAN 2.0B/ CAN FD Settings

Setting

Channel

CAN_L(Rx)

CAN_L A0

☑ Auto detect Data Rate

125 Kbps

(5 Kbps ~ 1 Mbps)

□ Show scale in the waveform

□ CAN FD

● ISO-CRC ○ Non ISO-CRC

Data phase

1000 Kbps

Sample Point [80%]

Color

| | | | |
|---|---|---|---|
| Start of Frame | 🟧 | RTR bit | ⬜ |
| Identifier | 🟧 | SRR bit | 🟪 |
| Data length code | 🟨 | IDE bit | 🟨 |
| Data | 🟩 | Reserved bit | 🟨 |
| CRC | 🟦 | Delimiter bit | 🟩 |
| ACK Slot | 🟦 | Error Frame | 🟦 |
| End of Frame | 🟪 | Error State | 🟥 |

Range

Decode Range

From Buffer Head

To Buffer Tail

**Channel Settings:**

**CAN_H/CAN_L(RX): C**an directly measure the stabilized physical layer or the logic signal converted by the transceiver, the best signal to measure is the logic signal Rx.

**Auto Detect Data Rate:**

I. When checked, the program assists in calculating the data rate.

II. When unchecked, user can choose the built-in Data Rate setting or input Data Rate by yourself.

III. The allowable input data rate range is 5Kbps-1Mbps.

IV. If CAN FD function is checked, this function will be disabled automatically because the Data Rate will be changed.

**Show Scale in Waveform:** When the box is checked, a scale point is displayed on the top of the waveform, which is convenient to check the bit cutting status. This function is not available if CAN FD is checked.

**CAN FD Setting:**

**ISO CRC/Non ISO CRC:** User can adjust the rules of ISO CRC analysis and calculation.

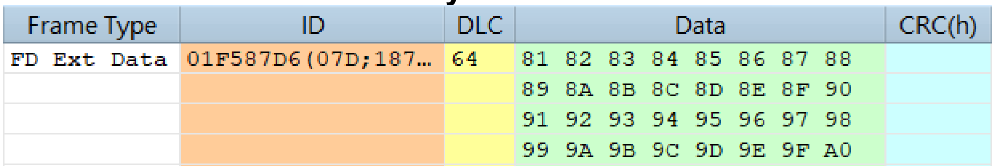
**Data Only:**

I. When checked, this function enables you to set the amount of data to be displayed in the analysis report, which can be set from 8 bytes to 60 bytes, and the data outside the set range will be deleted and not be displayed in the report, which is convenient for you to use when you view the report quickly.

II. When unchecked, all Data will be displayed.

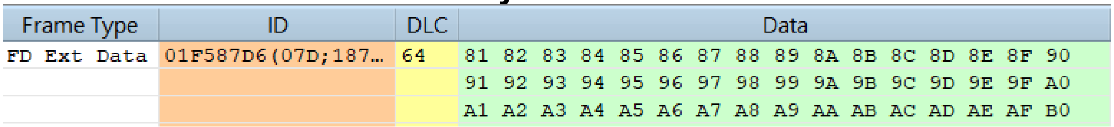
**Report Format:** Check this box to set the width of the data fields to be displayed in the analysis report, which can be set to 8, 16, or 32 bytes. 8 bytes will be displayed when the box is unchecked, and the following is an example of the application:

**Data field width set to 8 bytes.**



| Frame Type | ID | DLC | Data | CRC(h) |
|---|---|---|---|---|
| FD Ext Data | 01F587D6(07D;187… | 64 | 81 82 83 84 85 86 87 88 | |
| | | | 89 8A 8B 8C 8D 8E 8F 90 | |
| | | | 91 92 93 94 95 96 97 98 | |
| | | | 99 9A 9B 9C 9D 9E 9F A0 | |

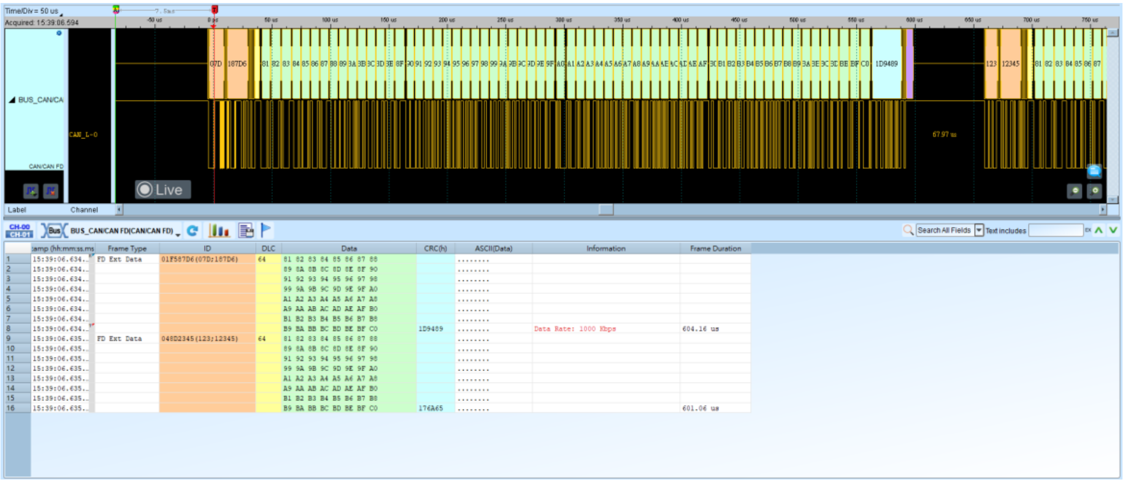
**Data field width is set to 16 bytes.**



| Frame Type | ID | DLC | Data |
|---|---|---|---|
| FD Ext Data | 01F587D6(07D;187… | 64 | 81 82 83 84 85 86 87 88 89 8A 8B 8C 8D 8E 8F 90 |
| | | | 91 92 93 94 95 96 97 98 99 9A 9B 9C 9D 9E 9F A0 |
| | | | A1 A2 A3 A4 A5 A6 A7 A8 A9 AA AB AC AD AE AF B0 |

## Result

## Use the CAN_L(Rx) signal for analysis.

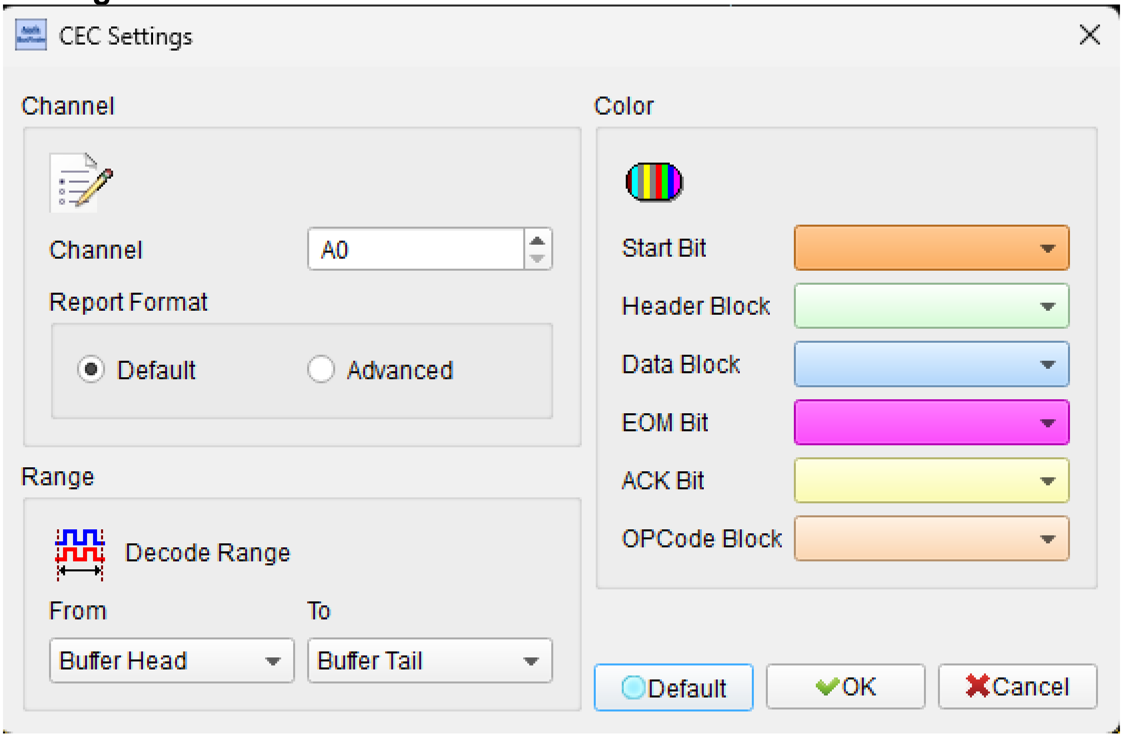


| | tamp (hh:mm:ss.ms | Frame Type | ID | DLC | Data | CRC(h) | ASCII(Data) | Information | Frame Duration |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 15:39:06.634.. | FD Ext Data | 01F587D6(07D;187D6) | 64 | 81 82 83 84 85 86 87 88 | | ........ | | |
| 2 | 15:39:06.634.. | | | | 89 8A 8B 8C 8D 8E 8F 90 | | ........ | | |
| 3 | 15:39:06.634.. | | | | 91 92 93 94 95 96 97 98 | | ........ | | |
| 4 | 15:39:06.634.. | | | | 99 9A 9B 9C 9D 9E 9F A0 | | ........ | | |
| 5 | 15:39:06.634.. | | | | A1 A2 A3 A4 A5 A6 A7 A8 | | ........ | | |
| 6 | 15:39:06.634.. | | | | A9 AA AB AC AD AE AF B0 | | ........ | | |
| 7 | 15:39:06.634.. | | | | B1 B2 B3 B4 B5 B6 B7 B8 | | ........ | | |
| 8 | 15:39:06.634.. | | | | B9 BA BB BC BD BE BF C0 | 1D9489 | ........ | Data Rate: 1000 Kbps | 604.16 us |
| 9 | 15:39:06.635.. | FD Ext Data | 048D2345(123;12345) | 64 | 81 82 83 84 85 86 87 88 | | ........ | | |
| 10 | 15:39:06.635.. | | | | 89 8A 8B 8C 8D 8E 8F 90 | | ........ | | |
| 11 | 15:39:06.635.. | | | | 91 92 93 94 95 96 97 98 | | ........ | | |
| 12 | 15:39:06.635.. | | | | 99 9A 9B 9C 9D 9E 9F A0 | | ........ | | |
| 13 | 15:39:06.635.. | | | | A1 A2 A3 A4 A5 A6 A7 A8 | | ........ | | |
| 14 | 15:39:06.635.. | | | | A9 AA AB AC AD AE AF B0 | | ........ | | |
| 15 | 15:39:06.635.. | | | | B1 B2 B3 B4 B5 B6 B7 B8 | | ........ | | |
| 16 | 15:39:06.635.. | | | | B9 BA BB BC BD BE BF C0 | 176A65 | ........ | | 601.06 us |

## CEC

Consumer Electronics Control, used for transmitting industrial specification AV Link protocol signals to support a single remote control to operate multiple AV machines, is a single-core, bi-directional serial bus.

### Settings



Channel: A0

Report Format: ● Default ○ Advanced

From: Buffer Head

To: Buffer Tail

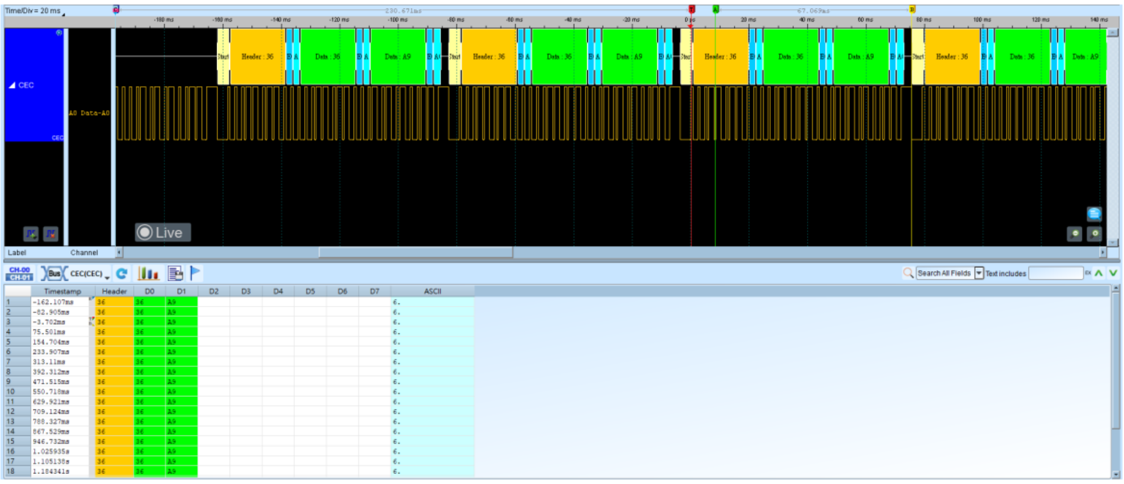
Start Bit
Header Block
Data Block
EOM Bit
ACK Bit
OPCode Block

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Report Format:** There are two types of settings, Advanced and Default. The Advanced mode explains the meaning of the Header and OPCode of the waveform.

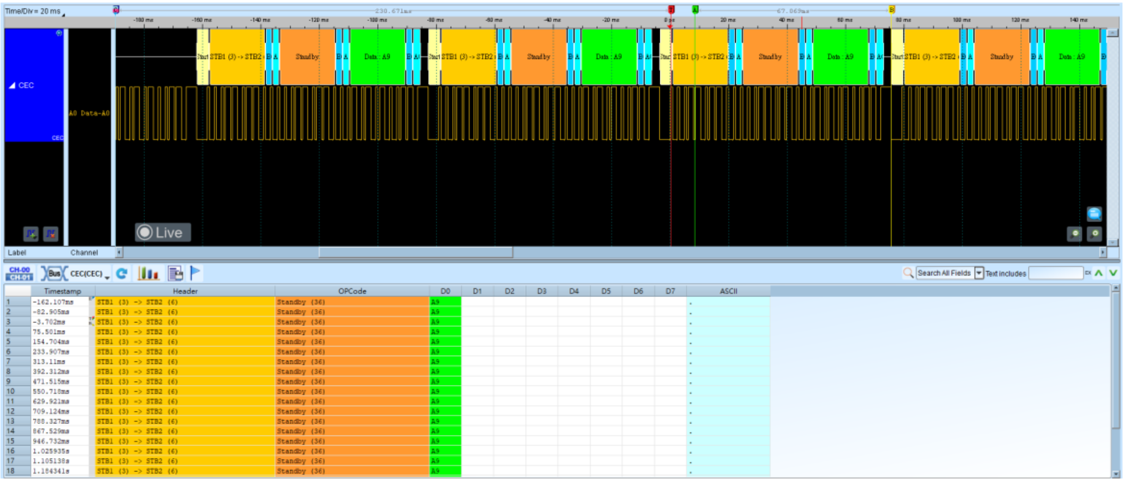
## Result

### Normal



| | Timestamp | Header | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -162.107ms | 36 | 36 | A9 | | | | | | | 6. |
| 2 | -82.905ms | 36 | 36 | A9 | | | | | | | 6. |
| 3 | -3.702ms | 36 | 36 | A9 | | | | | | | 6. |
| 4 | 75.501ms | 36 | 36 | A9 | | | | | | | 6. |
| 5 | 154.704ms | 36 | 36 | A9 | | | | | | | 6. |
| 6 | 233.907ms | 36 | 36 | A9 | | | | | | | 6. |
| 7 | 313.11ms | 36 | 36 | A9 | | | | | | | 6. |
| 8 | 392.312ms | 36 | 36 | A9 | | | | | | | 6. |
| 9 | 471.515ms | 36 | 36 | A9 | | | | | | | 6. |
| 10 | 550.718ms | 36 | 36 | A9 | | | | | | | 6. |
| 11 | 629.921ms | 36 | 36 | A9 | | | | | | | 6. |
| 12 | 709.124ms | 36 | 36 | A9 | | | | | | | 6. |
| 13 | 788.327ms | 36 | 36 | A9 | | | | | | | 6. |
| 14 | 867.529ms | 36 | 36 | A9 | | | | | | | 6. |
| 15 | 946.732ms | 36 | 36 | A9 | | | | | | | 6. |
| 16 | 1.025935s | 36 | 36 | A9 | | | | | | | 6. |
| 17 | 1.105138s | 36 | 36 | A9 | | | | | | | 6. |
| 18 | 1.184341s | 36 | 36 | A9 | | | | | | | 6. |

### Advanced

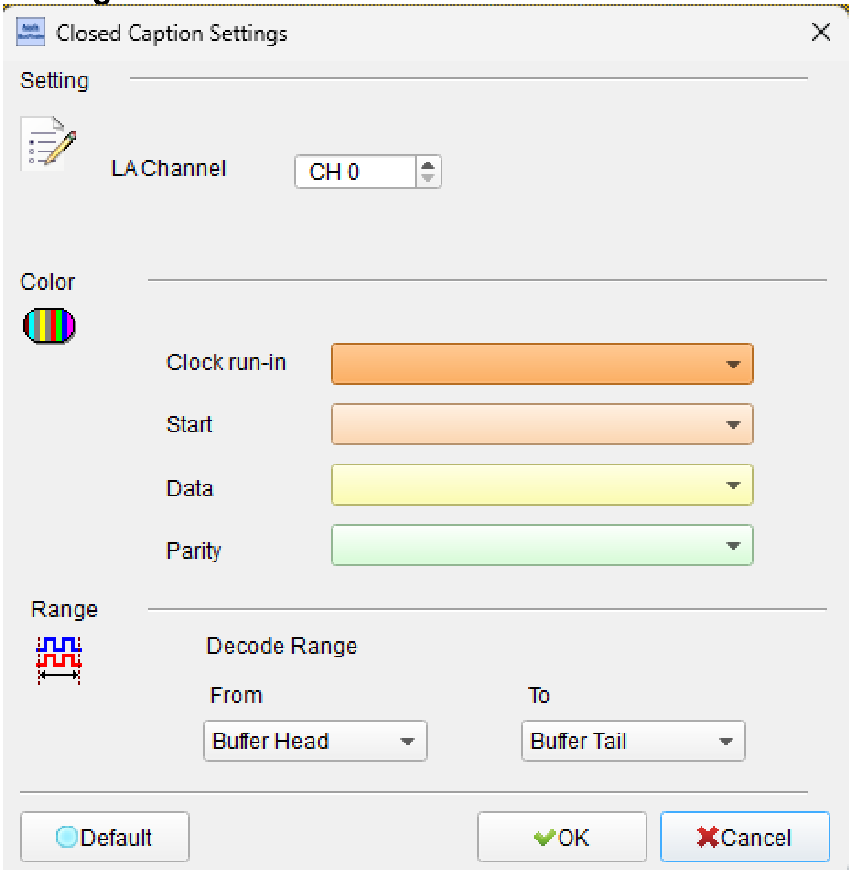


| | Timestamp | Header | OPCode | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -162.107ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 2 | -82.905ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 3 | -3.702ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 4 | 75.501ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 5 | 154.704ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 6 | 233.907ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 7 | 313.11ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 8 | 392.312ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 9 | 471.515ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 10 | 550.718ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 11 | 629.921ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 12 | 709.124ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 13 | 788.327ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 14 | 867.529ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 15 | 946.732ms | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 16 | 1.025935s | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 17 | 1.105138s | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |
| 18 | 1.184341s | STB1 (3) -> STB2 (6) | Standby (36) | A9 | | | | | | | | . |

## Closed Caption

Closed captioning is the process of displaying text on a TV or video screen. The text is encoded in the video data stream.

## Settings



Closed Caption Settings

Setting

LA Channel: CH 0

Color

Clock run-in: 🟧

Start: 🟧

Data: 🟨

Parity: 🟩

Range

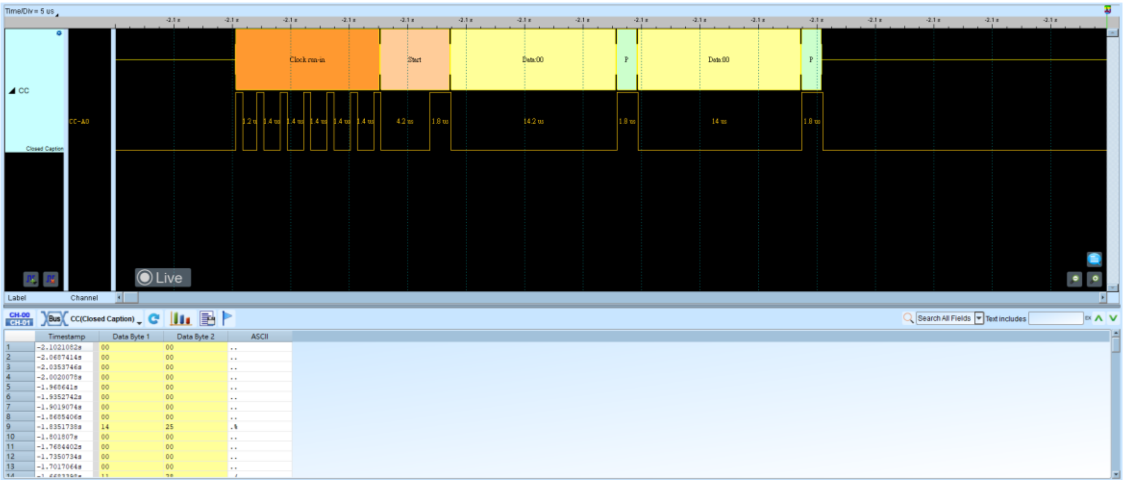
Decode Range

From: Buffer Head

To: Buffer Tail

**LA Channel:** Show the selected channel.

## Result

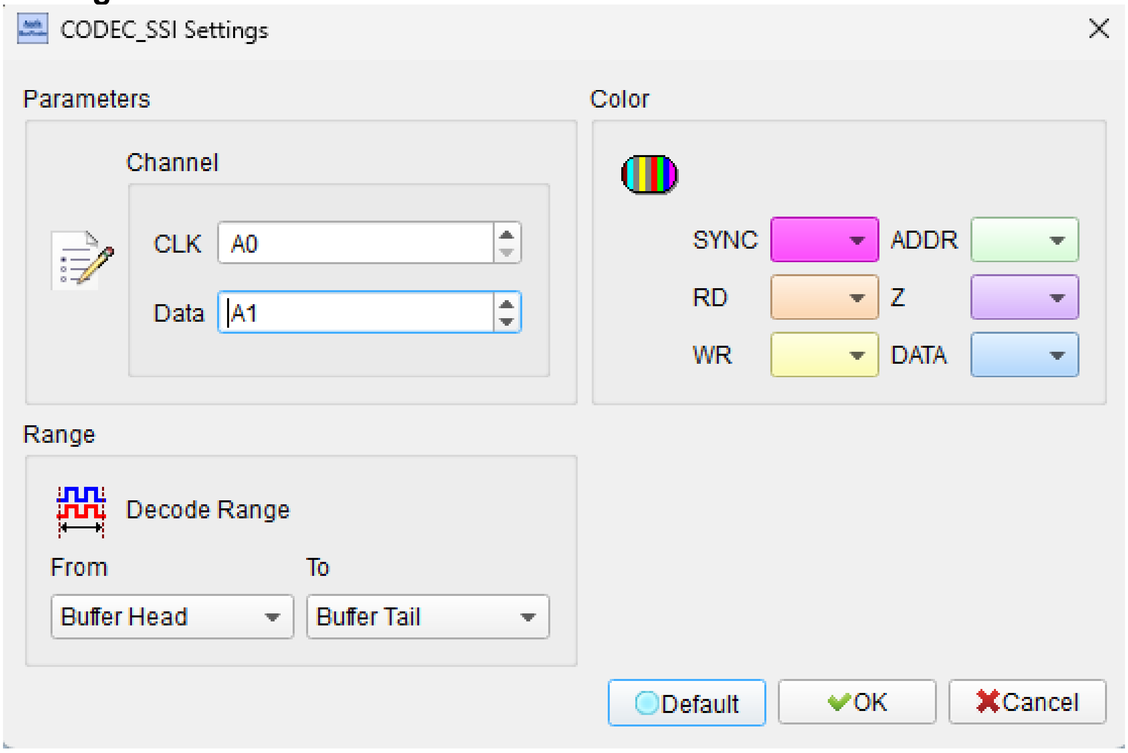


|  | Timestamp | Data Byte 1 | Data Byte 2 | ASCII |
|---|---|---|---|---|
| 1 | -2.1021062s | 00 | 00 | .. |
| 2 | -2.0687414s | 00 | 00 | .. |
| 3 | -2.0353746s | 00 | 00 | .. |
| 4 | -2.0020078s | 00 | 00 | .. |
| 5 | -1.968641s | 00 | 00 | .. |
| 6 | -1.9352742s | 00 | 00 | .. |
| 7 | -1.9019074s | 00 | 00 | .. |
| 8 | -1.8685406s | 00 | 00 | .. |
| 9 | -1.8351738s | 14 | 25 | .% |
| 10 | -1.801807s | 00 | 00 | .. |
| 11 | -1.7684402s | 00 | 00 | .. |
| 12 | -1.7350734s | 00 | 00 | .. |
| 13 | -1.7017064s | 00 | 00 | .. |

## Codec SSI

Serial Synchronous Interface (SSI) signal used by the codec (CODEC) inside the phone

### Settings



CODEC_SSI Settings

Parameters

Channel

CLK: A0

Data: A1

Color

SYNC, ADDR, RD, Z, WR, DATA

Range

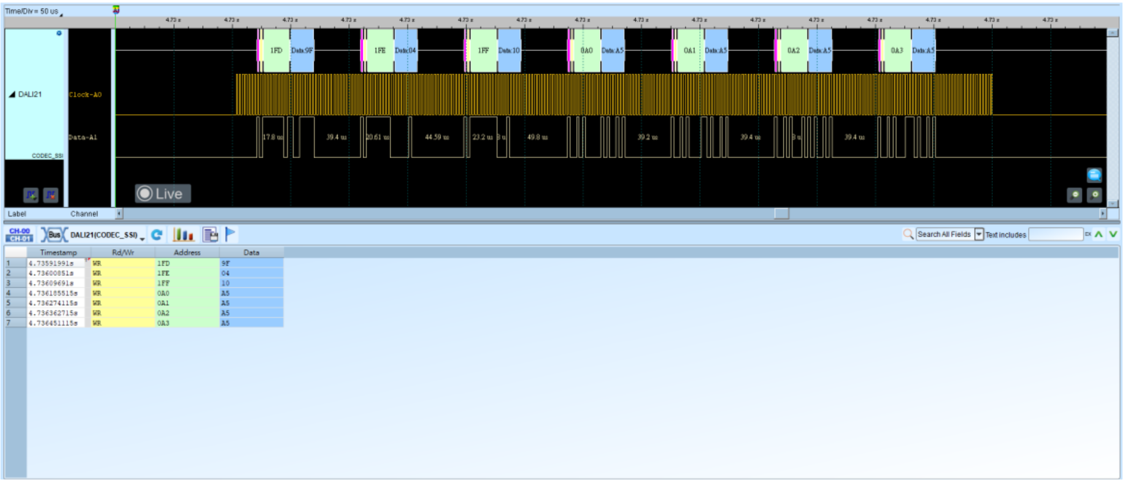
Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the signal on the DUT to the channel number of the logic analyzer.

### Result



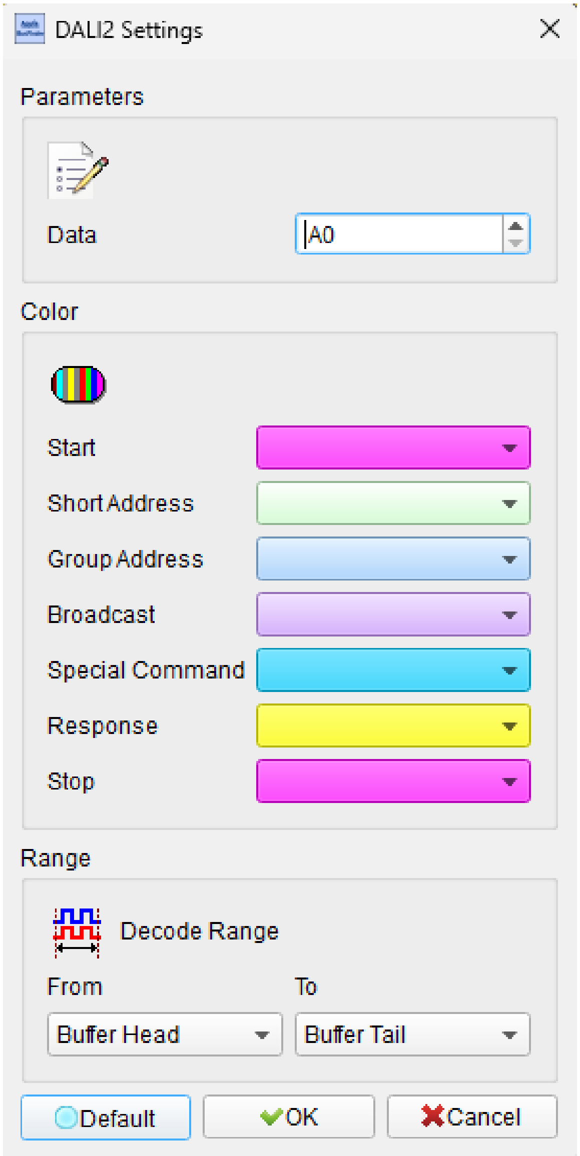
DALI21(CODEC_SSI)

|  | Timestamp | Rd/Wr | Address | Data |
|---|---|---|---|---|
| 1 | 4.73591991s | WR | 1FD | 9F |
| 2 | 4.73600851s | WR | 1FE | 04 |
| 3 | 4.73609691s | WR | 1FF | 10 |
| 4 | 4.736185515s | WR | 0A0 | A5 |
| 5 | 4.736274115s | WR | 0A1 | A5 |
| 6 | 4.736362715s | WR | 0A2 | A5 |
| 7 | 4.736451115s | WR | 0A3 | A5 |

## DALI

Digital Addressable Lighting Interface (DALI) is a technical standard for network-based systems that control lighting in buildings. The DALI standard, which is specified in the IEC 60929 standard for fluorescent lamp ballasts, encompasses the communications protocol and electrical interface for lighting control networks.

### Settings



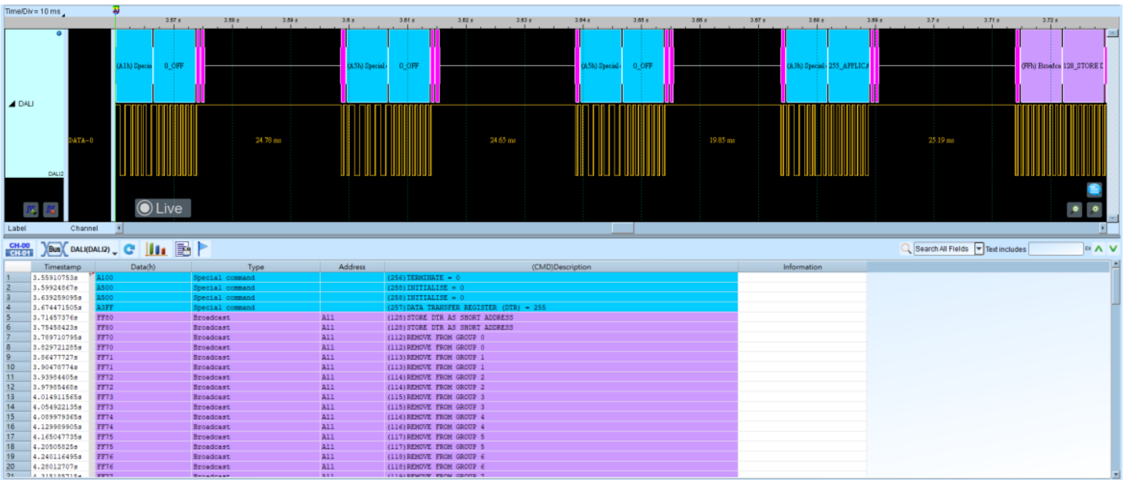
Data: A0

From: Buffer Head

To: Buffer Tail

**Data Channel:** Show the selected channel.

## Result

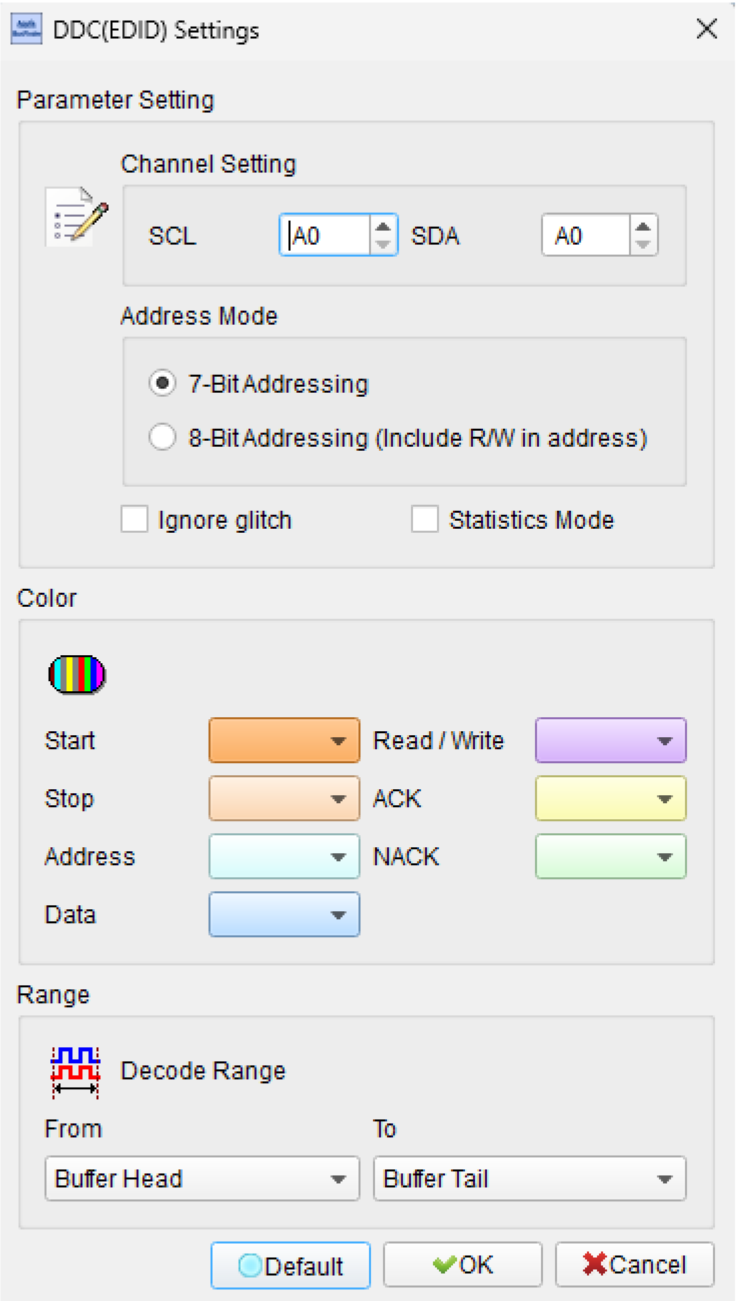


| | Timestamp | Data(h) | Type | Address | (CMD)Description | Information |
|---|---|---|---|---|---|---|
| 1 | 3.55910753s | A100 | Special command | | (256)TERMINATE = 0 | |
| 2 | 3.59924867s | A500 | Special command | | (258)INITIALISE = 0 | |
| 3 | 3.639259095s | A500 | Special command | | (258)INITIALISE = 0 | |
| 4 | 3.674471505s | A3FF | Special command | | (257)DATA TRANSFER REGISTER (DTR) = 255 | |
| 5 | 3.71457376s | FF80 | Broadcast | All | (128)STORE DTR AS SHORT ADDRESS | |
| 6 | 3.75458423s | FF80 | Broadcast | All | (128)STORE DTR AS SHORT ADDRESS | |
| 7 | 3.789710795s | FF70 | Broadcast | All | (112)REMOVE FROM GROUP 0 | |
| 8 | 3.829721285s | FF70 | Broadcast | All | (112)REMOVE FROM GROUP 0 | |
| 9 | 3.86477727s | FF71 | Broadcast | All | (113)REMOVE FROM GROUP 1 | |
| 10 | 3.90478774s | FF71 | Broadcast | All | (113)REMOVE FROM GROUP 1 | |
| 11 | 3.93984405s | FF72 | Broadcast | All | (114)REMOVE FROM GROUP 2 | |
| 12 | 3.97985468s | FF72 | Broadcast | All | (114)REMOVE FROM GROUP 2 | |
| 13 | 4.014911565s | FF73 | Broadcast | All | (115)REMOVE FROM GROUP 3 | |
| 14 | 4.054922135s | FF73 | Broadcast | All | (115)REMOVE FROM GROUP 3 | |
| 15 | 4.089979365s | FF74 | Broadcast | All | (116)REMOVE FROM GROUP 4 | |
| 16 | 4.129989905s | FF74 | Broadcast | All | (116)REMOVE FROM GROUP 4 | |
| 17 | 4.165047735s | FF75 | Broadcast | All | (117)REMOVE FROM GROUP 5 | |
| 18 | 4.20505825s | FF75 | Broadcast | All | (117)REMOVE FROM GROUP 5 | |
| 19 | 4.240116495s | FF76 | Broadcast | All | (118)REMOVE FROM GROUP 6 | |
| 20 | 4.28012707s | FF76 | Broadcast | All | (118)REMOVE FROM GROUP 6 | |
| 21 | 4.315185715s | FF77 | Broadcast | All | (119)REMOVE FROM GROUP 7 | |

## DDC (EDID)

EDID (Extended Display Identification Data) is I2C protocol base on DDC wire and transmitted monitor information. Now, HDMI, DVI and VGA are support this protocol.

### Settings



DDC(EDID) Settings

Parameter Setting

Channel Setting

SCL: A0 SDA: A0

Address Mode

● 7-Bit Addressing

○ 8-Bit Addressing (Include R/W in address)

□ Ignore glitch □ Statistics Mode

Color

| | | | |
|---|---|---|---|
| Start | ▼ | Read / Write | ▼ |
| Stop | ▼ | ACK | ▼ |
| Address | ▼ | NACK | ▼ |
| Data | ▼ | | |

Range

Decode Range

From: Buffer Head To: Buffer Tail

**SCL:** CLK of $I^2C$.

**SDA:** Data of $I^2C$.

**Address Mode:**

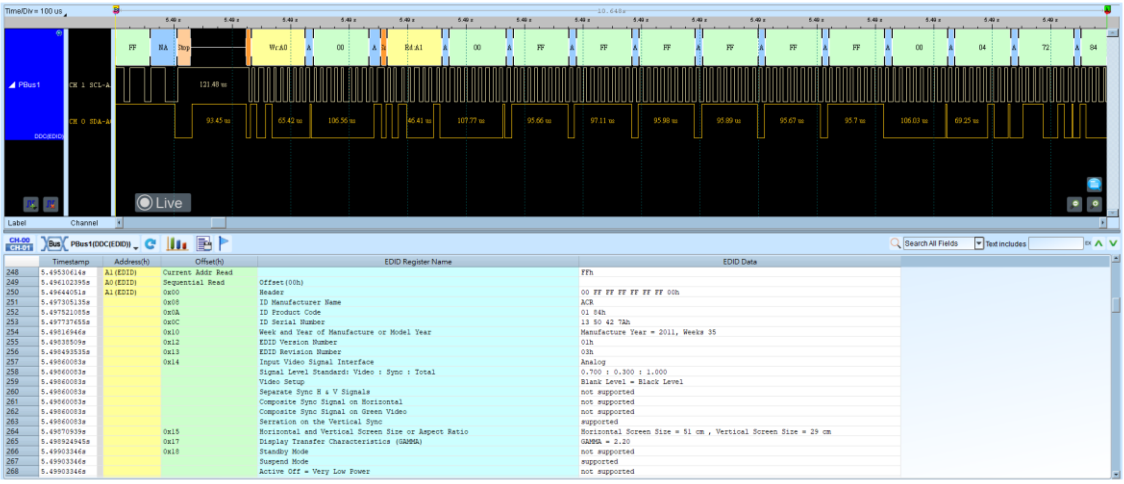
**7-bit addressing:** Show 7-bit addressing

**8-bit addressing(Include R/W in Address):** Show 8-bit addressing(include 7-bit addressing and 1-bit Rd/Wr).

**Ignore glitch:** Ignore the glitches occurred due to the slow transitions.

**Statistic Mode:** Collect all the data frames into one report by register address order.

**Result**

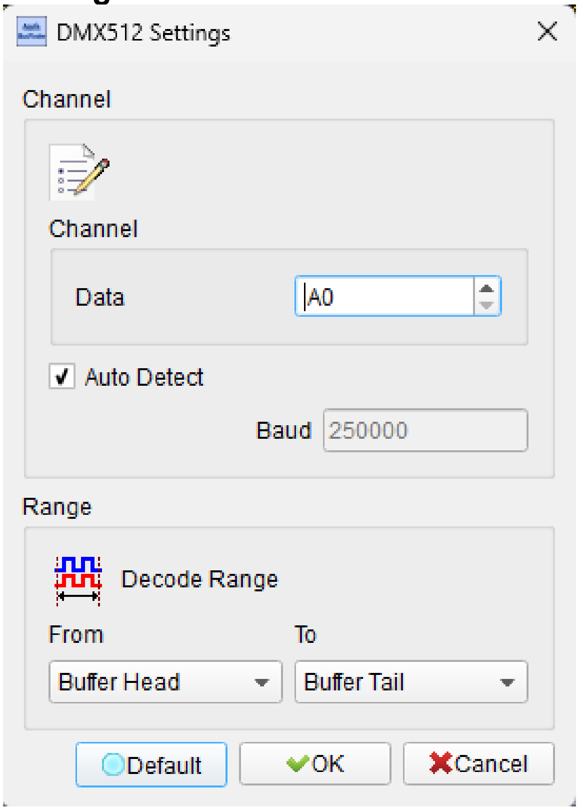


|  | Timestamp | Address(h) | Offset(h) | EDID Register Name | EDID Data |
|---|---|---|---|---|---|
| 248 | 5.49530614s | A1(EDID) | Current Addr Read |  | FFh |
| 249 | 5.496102395s | A0(EDID) | Sequential Read | Offset(00h) |  |
| 250 | 5.49644051s | A1(EDID) | 0x00 | Header | 00 FF FF FF FF FF FF 00h |
| 251 | 5.497305135s |  | 0x08 | ID Manufacturer Name | ACR |
| 252 | 5.497521085s |  | 0x0A | ID Product Code | 01 84h |
| 253 | 5.497737655s |  | 0x0C | ID Serial Number | 13 50 42 7Ah |
| 254 | 5.49816946s |  | 0x10 | Week and Year of Manufacture or Model Year | Manufacture Year = 2011, Weeks 35 |
| 255 | 5.49838509s |  | 0x12 | EDID Version Number | 01h |
| 256 | 5.498493535s |  | 0x13 | EDID Revision Number | 03h |
| 257 | 5.49860083s |  | 0x14 | Input Video Signal Interface | Analog |
| 258 | 5.49860083s |  |  | Signal Level Standard: Video : Sync : Total | 0.700 : 0.300 : 1.000 |
| 259 | 5.49860083s |  |  | Video Setup | Blank Level = Black Level |
| 260 | 5.49860083s |  |  | Separate Sync H & V Signals | not supported |
| 261 | 5.49860083s |  |  | Composite Sync Signal on Horizontal | not supported |
| 262 | 5.49860083s |  |  | Composite Sync Signal on Green Video | not supported |
| 263 | 5.49860083s |  |  | Serration on the Vertical Sync | supported |
| 264 | 5.49870939s |  | 0x15 | Horizontal and Vertical Screen Size or Aspect Ratio | Horizontal Screen Size = 51 cm , Vertical Screen Size = 29 cm |
| 265 | 5.498924945s |  | 0x17 | Display Transfer Characteristics (GAMMA) | GAMMA = 2.20 |
| 266 | 5.49903346s |  | 0x18 | Standby Mode | not supported |
| 267 | 5.49903346s |  |  | Suspend Mode | supported |
| 268 | 5.49903346s |  |  | Active Off = Very Low Power | not supported |

## DMX512

DMX512 is a standard for digital communication networks that are commonly used to control stage lighting and effects..

### Settings



DMX512 Settings

Channel

Channel

Data: A0

☑ Auto Detect

Baud: 250000

Range

Decode Range

From: Buffer Head

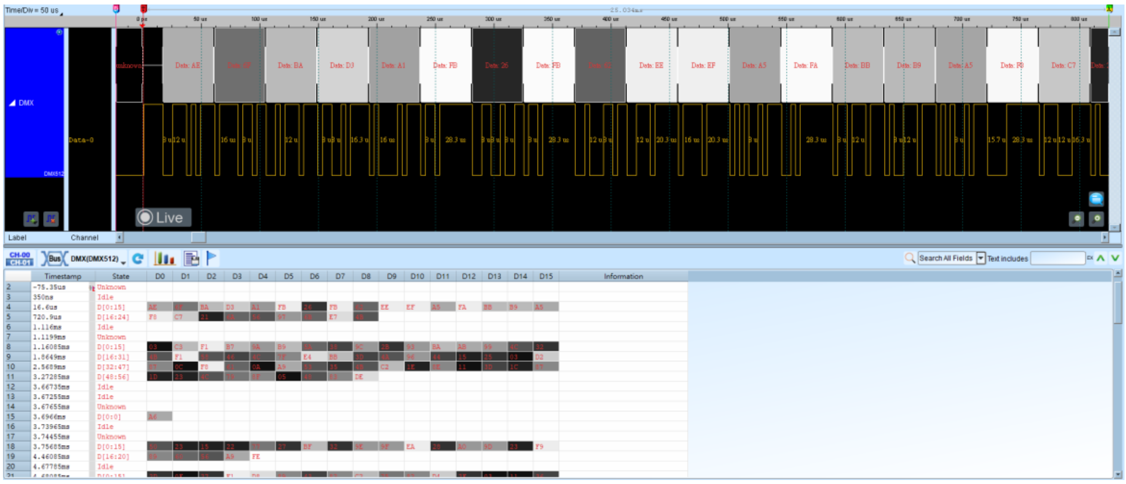
To: Buffer Tail

**Data**: Show the selected channel (CH0).

**Auto Detect:** Set the Baud Rate manually if not selected.

### Result

Use grayscale to display the decode results.

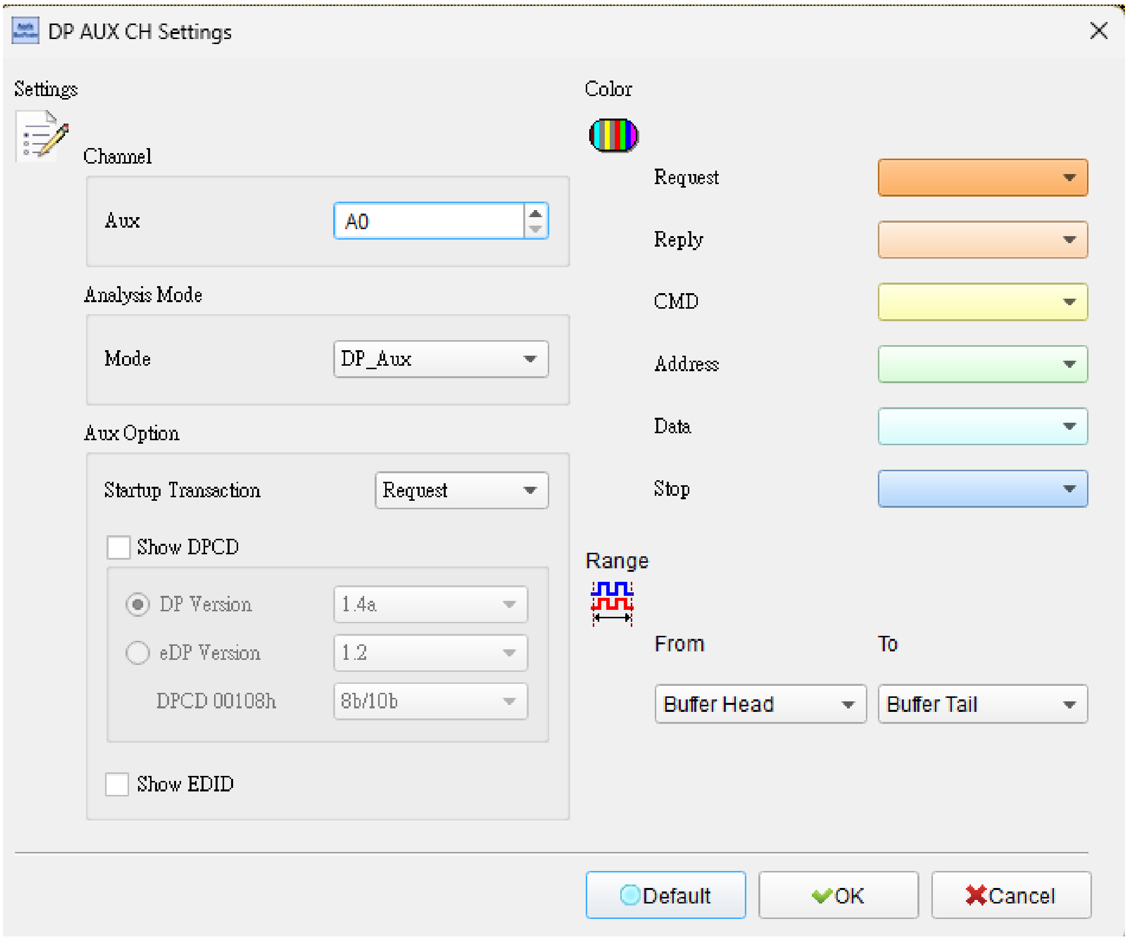


| | Timestamp | State | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | -75.35us | Unknown | | | | | | | | | | | | | | | | | |
| 3 | 350ns | Idle | | | | | | | | | | | | | | | | | |
| 4 | 16.6us | D[0:15] | AE | [illegible] | BA | D3 | A1 | FB | 26 | FB | [illegible] | EE | EF | A5 | FA | BB | B9 | A5 | |
| 5 | 720.9us | D[16:24] | F8 | C7 | 21 | [illegible] | 56 | 97 | [illegible] | E7 | 4B | | | | | | | | |
| 6 | 1.116ms | Idle | | | | | | | | | | | | | | | | | |
| 7 | 1.1199ms | Unknown | | | | | | | | | | | | | | | | | |
| 8 | 1.16085ms | D[0:15] | 03 | C3 | F1 | B7 | 9A | B9 | 5A | [illegible] | 9C | 2B | 93 | BA | AB | 99 | 4C | 32 | |
| 9 | 1.8649ms | D[16:31] | 4B | F1 | [illegible] | 46 | 4C | 7F | E4 | BB | 3D | 4A | 96 | 44 | 15 | 25 | 03 | D2 | |
| 10 | 2.5689ms | D[32:47] | 87 | 0C | F8 | [illegible] | 0A | A9 | 53 | 35 | 4B | C2 | 1E | 8E | 11 | 3D | 1C | 87 | |
| 11 | 3.27285ms | D[48:56] | 1D | 23 | 4C | 79 | 8F | 05 | 48 | 83 | DE | | | | | | | | |
| 12 | 3.66735ms | Idle | | | | | | | | | | | | | | | | | |
| 13 | 3.67255ms | Idle | | | | | | | | | | | | | | | | | |
| 14 | 3.67655ms | Unknown | | | | | | | | | | | | | | | | | |
| 15 | 3.6966ms | D[0:0] | A6 | | | | | | | | | | | | | | | | |
| 16 | 3.73965ms | Idle | | | | | | | | | | | | | | | | | |
| 17 | 3.74455ms | Unknown | | | | | | | | | | | | | | | | | |
| 18 | 3.75685ms | D[0:15] | 50 | 23 | 15 | 22 | [illegible] | 27 | BF | 32 | 9E | 9F | EA | 28 | A0 | 9D | 23 | F9 | |
| 19 | 4.46085ms | D[16:20] | 89 | 80 | 56 | A9 | FE | | | | | | | | | | | | |
| 20 | 4.67785ms | Idle | | | | | | | | | | | | | | | | | |
| 21 | [illegible] | D[0:15] | [illegible] | [illegible] | [illegible] | F1 | D8 | [illegible] | [illegible] | [illegible] | C7 | [illegible] | [illegible] | D4 | [illegible] | [illegible] | [illegible] | [illegible] | |

## DP Aux Ch

The DP Aux Ch is to detect the link, configuration and status of the Display Port source. The Display Port is the digital display interface that is specified in the VESA standard. (Support DP 2.1, eDP 1.5)

## Settings



DP AUX CH Settings

Settings

Channel

Aux: A0

Analysis Mode

Mode: DP_Aux

Aux Option

Startup Transaction: Request

□ Show DPCD

● DP Version: 1.4a

○ eDP Version: 1.2

DPCD 00108h: 8b/10b

□ Show EDID

Color

Request, Reply, CMD, Address, Data, Stop

Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel to decode

**Show DPCD:** Show the Display Port Configuration data. Enable when checked.

I. DP Version supported to 2.1

II. eDP Version supported to 1.5

III. DPCP 00108h: 8b/10b or 128b/132b encoding options are available.

**Mode:** Choose the mode DP_Aux / HPD / PWR

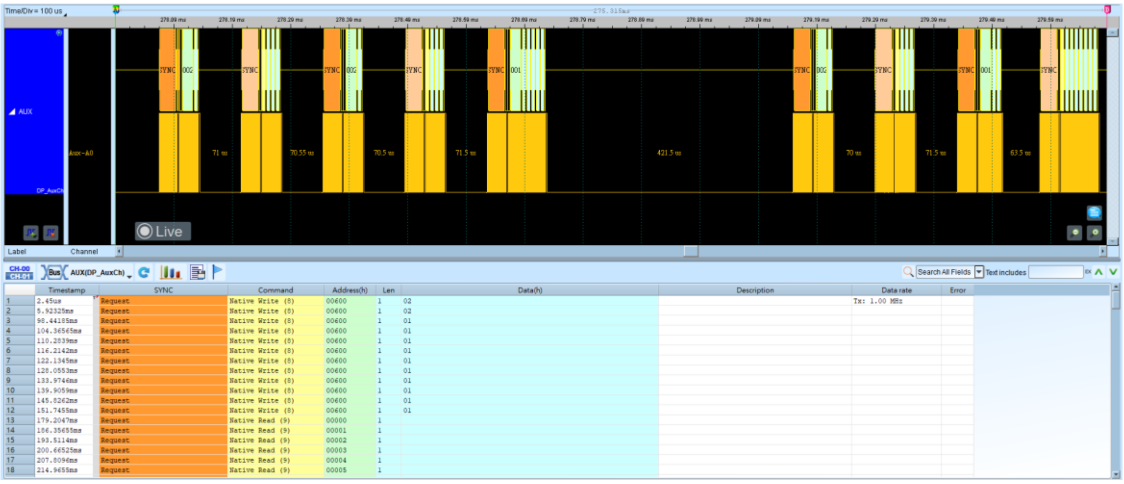
**Show EDID: Show the EDID information. Enable when checked.**

**Startup transaction:** Set the transaction type of the first frame

**Reply Timeout:** Set the value of timeout

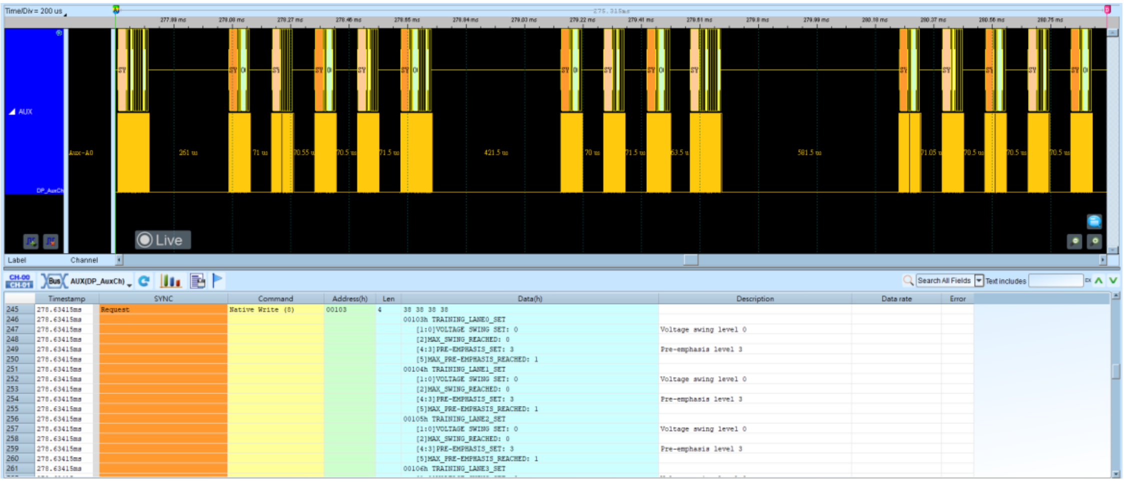
## Result

### Without the DPCD information



| | Timestamp | SYNC | Command | Address(h) | Len | Data(h) | Description | Data rate | Error |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 2.45us | Request | Native Write (8) | 00600 | 1 | 02 | | Tx: 1.00 MHz | |
| 2 | 5.92325ms | Request | Native Write (8) | 00600 | 1 | 02 | | | |
| 3 | 98.44185ms | Request | Native Write (8) | 00600 | 1 | 01 | | | |
| 4 | 104.36565ms | Request | Native Write (8) | 00600 | 1 | 01 | | | |
| 5 | 110.2839ms | Request | Native Write (8) | 00600 | 1 | 01 | | | |
| 6 | 116.2142ms | Request | Native Write (8) | 00600 | 1 | 01 | | | |
| 7 | 122.1345ms | Request | Native Write (8) | 00600 | 1 | 01 | | | |
| 8 | 128.0553ms | Request | Native Write (8) | 00600 | 1 | 01 | | | |
| 9 | 133.9746ms | Request | Native Write (8) | 00600 | 1 | 01 | | | |
| 10 | 139.9059ms | Request | Native Write (8) | 00600 | 1 | 01 | | | |
| 11 | 145.8262ms | Request | Native Write (8) | 00600 | 1 | 01 | | | |
| 12 | 151.7455ms | Request | Native Write (8) | 00600 | 1 | 01 | | | |
| 13 | 179.2047ms | Request | Native Read (9) | 00000 | 1 | | | | |
| 14 | 186.35655ms | Request | Native Read (9) | 00001 | 1 | | | | |
| 15 | 193.5114ms | Request | Native Read (9) | 00002 | 1 | | | | |
| 16 | 200.66525ms | Request | Native Read (9) | 00003 | 1 | | | | |
| 17 | 207.8096ms | Request | Native Read (9) | 00004 | 1 | | | | |
| 18 | 214.9655ms | Request | Native Read (9) | 00005 | 1 | | | | |

### Show the DPCD information

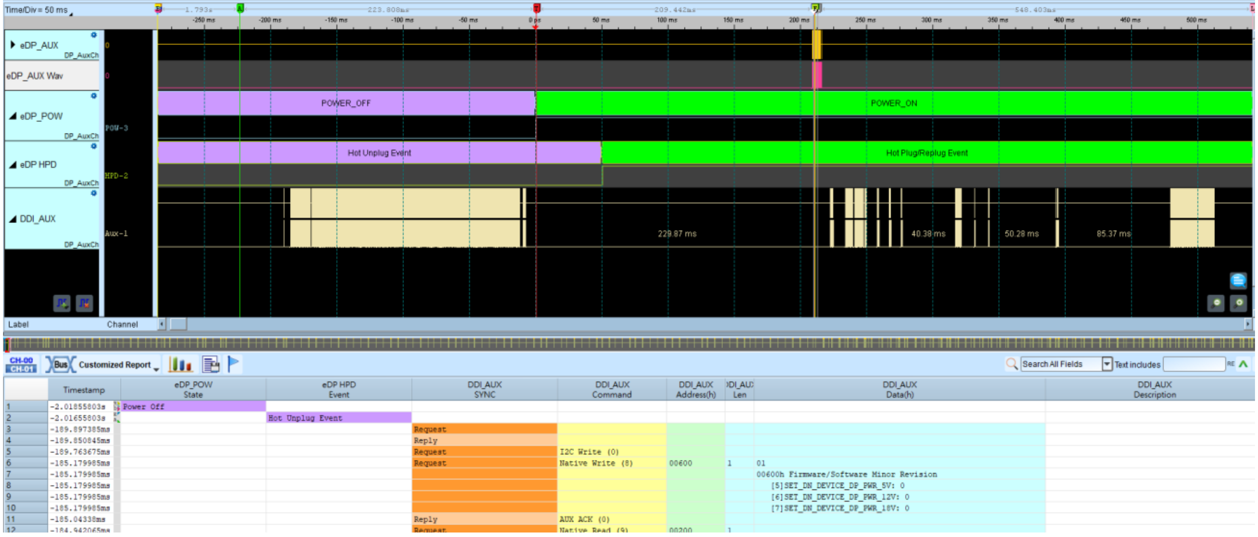


| | Timestamp | SYNC | Command | Address(h) | Len | Data(h) | Description | Data rate | Error |
|---|---|---|---|---|---|---|---|---|---|
| 245 | 278.63415ms | Request | Native Write (8) | 00103 | 4 | 38 38 38 38 | | | |
| 246 | 278.63415ms | | | | | 00103h TRAINING_LANE0_SET | | | |
| 247 | 278.63415ms | | | | | [1:0]VOLTAGE SWING SET: 0 | Voltage swing level 0 | | |
| 248 | 278.63415ms | | | | | [2]MAX_SWING_REACHED: 0 | | | |
| 249 | 278.63415ms | | | | | [4:3]PRE-EMPHASIS_SET: 3 | Pre-emphasis level 3 | | |
| 250 | 278.63415ms | | | | | [5]MAX_PRE-EMPHASIS_REACHED: 1 | | | |
| 251 | 278.63415ms | | | | | 00104h TRAINING_LANE1_SET | | | |
| 252 | 278.63415ms | | | | | [1:0]VOLTAGE SWING SET: 0 | Voltage swing level 0 | | |
| 253 | 278.63415ms | | | | | [2]MAX_SWING_REACHED: 0 | | | |
| 254 | 278.63415ms | | | | | [4:3]PRE-EMPHASIS_SET: 3 | Pre-emphasis level 3 | | |
| 255 | 278.63415ms | | | | | [5]MAX_PRE-EMPHASIS_REACHED: 1 | | | |
| 256 | 278.63415ms | | | | | 00105h TRAINING_LANE2_SET | | | |
| 257 | 278.63415ms | | | | | [1:0]VOLTAGE SWING SET: 0 | Voltage swing level 0 | | |
| 258 | 278.63415ms | | | | | [2]MAX_SWING_REACHED: 0 | | | |
| 259 | 278.63415ms | | | | | [4:3]PRE-EMPHASIS_SET: 3 | Pre-emphasis level 3 | | |
| 260 | 278.63415ms | | | | | [5]MAX_PRE-EMPHASIS_REACHED: 1 | | | |
| 261 | 278.63415ms | | | | | 00106h TRAINING_LANE3_SET | | | |

## Accessories:

[https://www.acute.com.tw/en/product/detail143](https://www.acute.com.tw/en/product/detail143)

## Analysis Aux, HPD, PWR

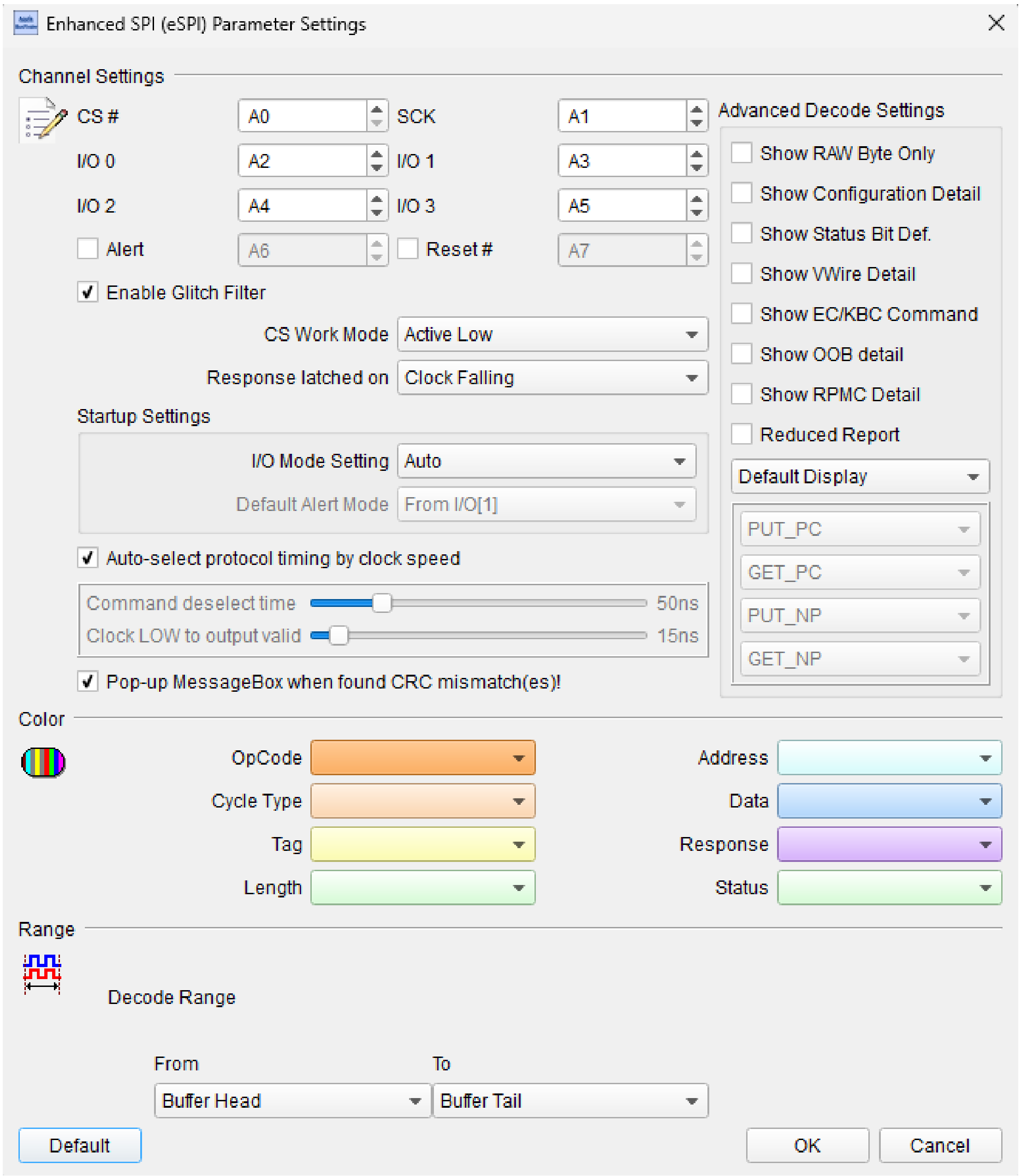


| | Timestamp | eDP_POW State | eDP HPD Event | DDI_AUX SYNC | DDI_AUX Command | DDI_AUX Address(h) | DDI_AUX Len | DDI_AUX Data(h) | DDI_AUX Description |
|---|---|---|---|---|---|---|---|---|---|
| 1 | -2.01855803s | Power Off | | | | | | | |
| 2 | -2.01655803s | | Hot Unplug Event | | | | | | |
| 3 | -189.897385ms | | | Request | | | | | |
| 4 | -189.850845ms | | | Reply | | | | | |
| 5 | -189.763675ms | | | Request | I2C Write (0) | | | | |
| 6 | -185.179985ms | | | Request | Native Write (8) | 00600 | 1 | 01 | |
| 7 | -185.179985ms | | | | | | | 00600h Firmware/Software Minor Revision | |
| 8 | -185.179985ms | | | | | | | [5]SET_DN_DEVICE_DP_PWR_5V: 0 | |
| 9 | -185.179985ms | | | | | | | [6]SET_DN_DEVICE_DP_PWR_12V: 0 | |
| 10 | -185.179985ms | | | | | | | [7]SET_DN_DEVICE_DP_PWR_18V: 0 | |
| 11 | -185.04338ms | | | Reply | AUX ACK (0) | | | | |
| 12 | -184.942065ms | | | Request | Native Read (9) | 00200 | 1 | | |

## eSPI

eSPI is the transmission protocol used in new generation baseboard of Intel, and its specification is to integrate SMBus / LPC / SPI Flash interface to simplify bus and increase transmission efficiency. Source of specification is based on Enhanced Serial Peripheral Interface (eSPI) Interface Base Specification (for Client and Server Platforms) June 2013, Revision 0.75.

## Settings



Enhanced SPI (eSPI) Parameter Settings

Channel Settings

CS # A0 SCK A1

I/O 0 A2 I/O 1 A3

I/O 2 A4 I/O 3 A5

□ Alert A6 □ Reset # A7

☑ Enable Glitch Filter

CS Work Mode: Active Low

Response latched on: Clock Falling

Startup Settings

I/O Mode Setting: Auto

Default Alert Mode: From I/O[1]

☑ Auto-select protocol timing by clock speed

Command deselect time 50ns

Clock LOW to output valid 15ns

☑ Pop-up MessageBox when found CRC mismatch(es)!

Advanced Decode Settings

□ Show RAW Byte Only

□ Show Configuration Detail

□ Show Status Bit Def.

□ Show VWire Detail

□ Show EC/KBC Command

□ Show OOB detail

□ Show RPMC Detail

□ Reduced Report

Default Display

PUT_PC

GET_PC

PUT_NP

GET_NP

Color

OpCode 🟧 Address 🟦

Cycle Type 🟧 Data 🟦

Tag 🟨 Response 🟪

Length 🟩 Status 🟩

Range

Decode Range

From: Buffer Head To: Buffer Tail

Default OK Cancel

### Channel:

CS#: Chip Select (Active Low)

SCK: Clock

I/O0 – I/O3: Data input / output

Alert: Alert signal (Optional)

Reset: Reset signal (Optional)

**Startup Settings:**

I/O Mode Setting: Set the initial I / O state to be Single / Dual / Quad, and I / O state would be switched automatically by the content of the waveform.

Default Alert Mode: Set the channel of Alert signal.

**Command deselect time:** Set tSHSL, Chip Select# Deassertion Time.

**Clock LOW to output valid:** Set tCLQV, Output Data Valid Time.

**Advanced Decode Setting:**

Show Configuration Detail: Show details of SET_CONFIG / GET_CONFIG.

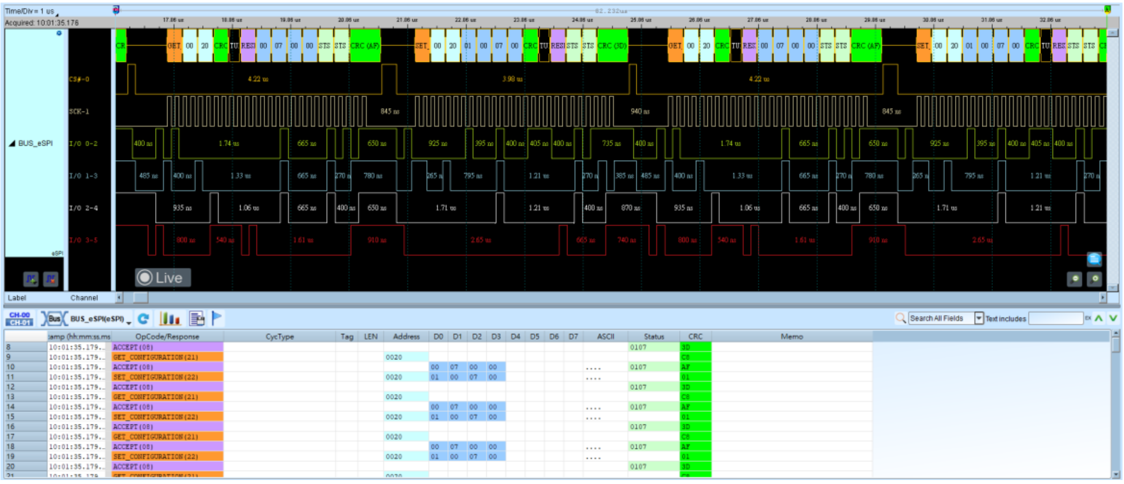
Show Status Bit Def.: Show details of Status.

Reduced Report: Reducing the report is easy to check the Command Flow.

Filter Setting: To show or hide the specific OP Code / Cycle Type or Address range in the report.

**Note:** The setting of Address Filter would be saved as LA\eSPI\eSPIFilterX.bin in the work directory.

## Result

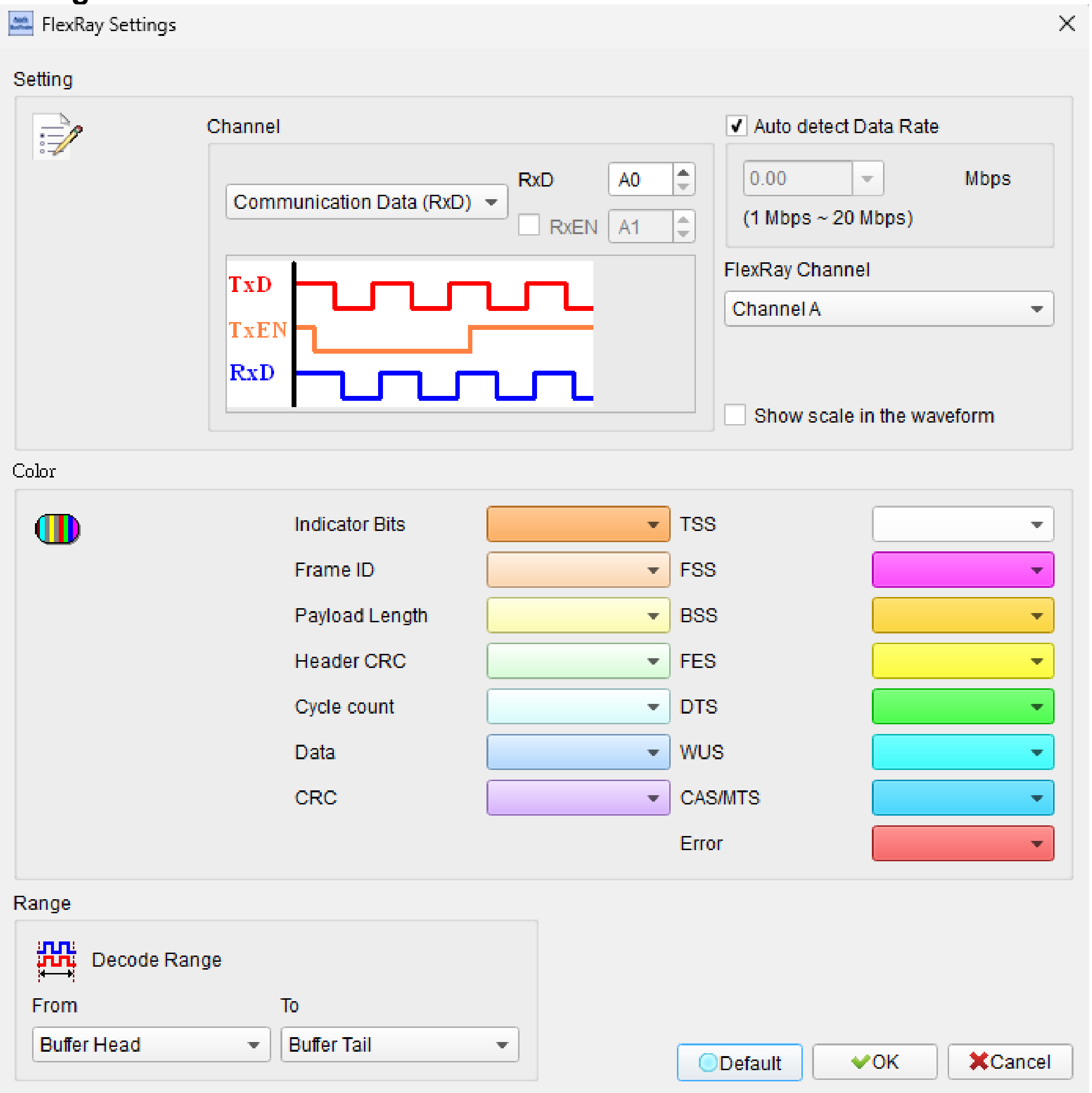


| | amp (hh:mm:ss.ms | OpCode/Response | CycType | Tag | LEN | Address | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII | Status | CRC | Memo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 8 | 10:01:35.179.. | ACCEPT(08) | | | | | | | | | | | | | | 0107 | 3D | |
| 9 | 10:01:35.179.. | GET_CONFIGURATION(21) | | | | 0020 | | | | | | | | | | | C8 | |
| 10 | 10:01:35.179.. | ACCEPT(08) | | | | | 00 | 07 | 00 | 00 | | | | | .... | 0107 | AF | |
| 11 | 10:01:35.179.. | SET_CONFIGURATION(22) | | | | 0020 | 01 | 00 | 07 | 00 | | | | | .... | | 01 | |
| 12 | 10:01:35.179.. | ACCEPT(08) | | | | | | | | | | | | | | 0107 | 3D | |
| 13 | 10:01:35.179.. | GET_CONFIGURATION(21) | | | | 0020 | | | | | | | | | | | C8 | |
| 14 | 10:01:35.179.. | ACCEPT(08) | | | | | 00 | 07 | 00 | 00 | | | | | .... | 0107 | AF | |
| 15 | 10:01:35.179.. | SET_CONFIGURATION(22) | | | | 0020 | 01 | 00 | 07 | 00 | | | | | .... | | 01 | |
| 16 | 10:01:35.179.. | ACCEPT(08) | | | | | | | | | | | | | | 0107 | 3D | |
| 17 | 10:01:35.179.. | GET_CONFIGURATION(21) | | | | 0020 | | | | | | | | | | | C8 | |
| 18 | 10:01:35.179.. | ACCEPT(08) | | | | | 00 | 07 | 00 | 00 | | | | | .... | 0107 | AF | |
| 19 | 10:01:35.179.. | SET_CONFIGURATION(22) | | | | 0020 | 01 | 00 | 07 | 00 | | | | | .... | | 01 | |
| 20 | 10:01:35.179.. | ACCEPT(08) | | | | | | | | | | | | | | 0107 | 3D | |
| 21 | 10:01:35.179 | GET_CONFIGURATION(21) | | | | 0020 | | | | | | | | | | | C8 | |

## FlexRay

FlexRay is an in-vehicle communication network standard that supports two communication channels, each with a speed of up to 10 Mbps.

## Settings



FlexRay Settings

Setting

Channel: Communication Data (RxD)

RxD: A0

□ RxEN: A1

☑ Auto detect Data Rate

0.00 Mbps

(1 Mbps ~ 20 Mbps)

FlexRay Channel: Channel A

□ Show scale in the waveform

Color

| | | | |
|---|---|---|---|
| Indicator Bits | 🟧 | TSS | ⬜ |
| Frame ID | 🟧 | FSS | 🟪 |
| Payload Length | 🟨 | BSS | 🟨 |
| Header CRC | 🟩 | FES | 🟨 |
| Cycle count | ⬜ | DTS | 🟩 |
| Data | 🟦 | WUS | 🟦 |
| CRC | 🟪 | CAS/MTS | 🟦 |
| | | Error | 🟥 |

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel, Communication Data (TxD)

**Communication Data (TxD):** The TxD data is from the TxD and TxEN of the FlexRay transceiver.

**Communication Data (RxD):** The RxD data is from the RxD and RxEN of the FlexRay transceiver.

**Auto detect Data Rate:** Default is Auto Bit Rate. If disabled, you may use built-in Bit Rate 10/5/2.5 Mbps or input manually, ranges from 1Mbps-20Mbps.

**FlexRay Channel:** Channel A or B, for Frame CRC checking.

**Errors are:**

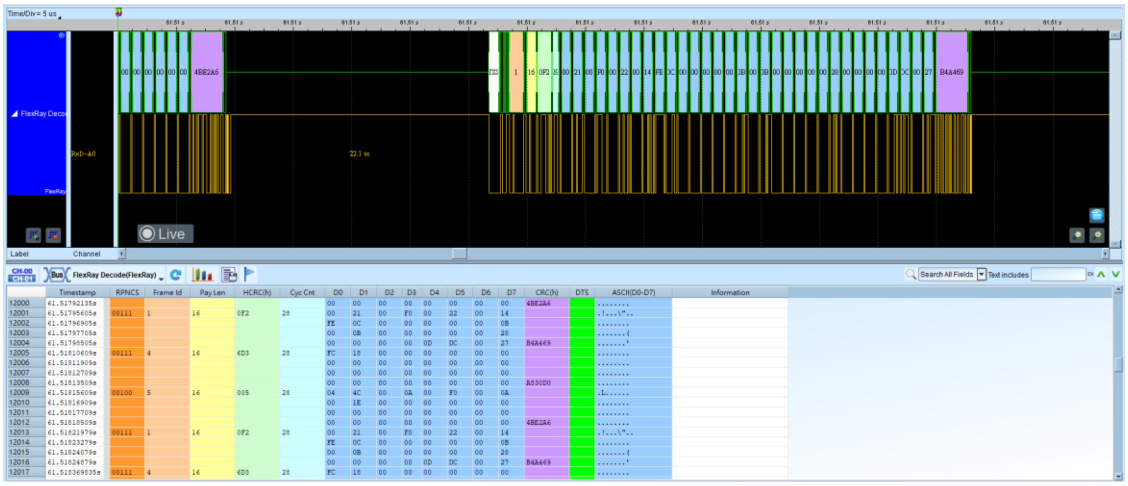
| Error | Description |
|---|---|
| TSS Error | Unable to detect TSS |
| FSS Error | Unable to detect FSS |
| BSS Error | Unable to detect BSS |
| FES Error | Unable to detect FES |
| Header CRC Error | The header CRC value is incorrect |
| Frame CRC Error | The frame CRC value is incorrect |

**Abbreviations are:**

| Abbreviation | Description |
|---|---|
| TSS | Transmission start sequence |
| FSS | Frame start sequence |
| BSS | Byte start sequence |
| FES | Frame end sequence |
| DTS | Dynamic trailing sequence |
| CAS | Collision Avoidance Symbol |
| MTS | Media Access Test Symbol |
| WUP | Wakeup Pattern |
| CID | Channel Idle Delimiter |

## Result

10Mbps FlexRay Communication Data(RxD)

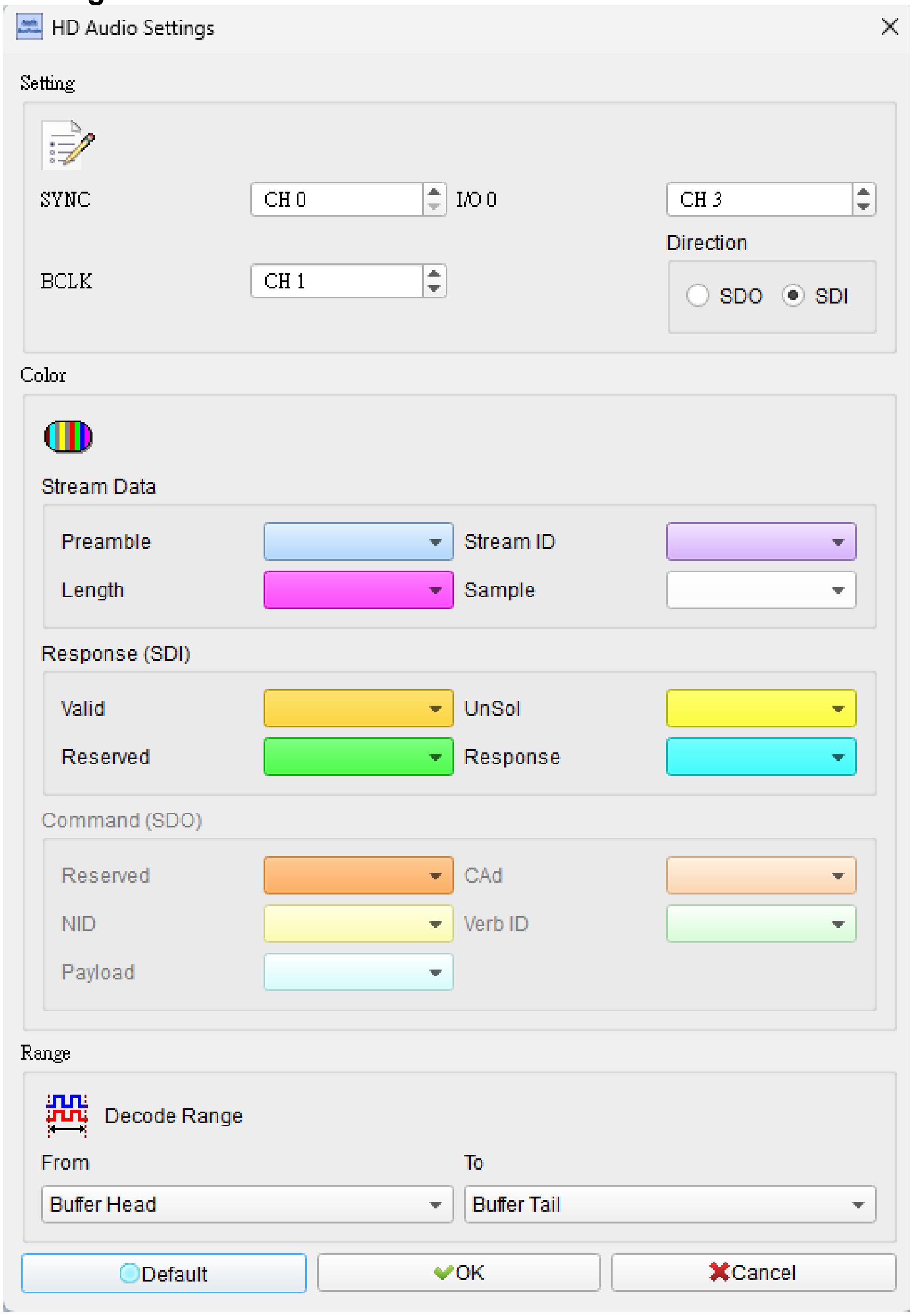


| | Timestamp | RPNCS | Frame Id | Pay Len | HCRC(h) | Cyc Cnt | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | CRC(h) | DTS | ASCII(D0-D7) | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 12000 | 61.51792135s | | | | | | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 4BE2A6 | | ........ | |
| 12001 | 61.51795605s | 00111 | 1 | 16 | 0F2 | 28 | 00 | 21 | 00 | F0 | 00 | 22 | 00 | 14 | | | .!...\".. | |
| 12002 | 61.51796905s | | | | | | FE | 0C | 00 | 00 | 00 | 00 | 00 | 0B | | | ........ | |
| 12003 | 61.51797705s | | | | | | 00 | 0B | 00 | 00 | 00 | 00 | 00 | 28 | | | .......( | |
| 12004 | 61.51798505s | | | | | | 00 | 00 | 00 | 00 | 0D | DC | 00 | 27 | B4A469 | | .......' | |
| 12005 | 61.51810609s | 00111 | 4 | 16 | 6D3 | 28 | FC | 18 | 00 | 00 | 00 | 00 | 00 | 00 | | | ........ | |
| 12006 | 61.51811909s | | | | | | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | | | ........ | |
| 12007 | 61.51812709s | | | | | | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | | | ........ | |
| 12008 | 61.51813509s | | | | | | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | A830D0 | | ........ | |
| 12009 | 61.51815609s | 00100 | 5 | 16 | 005 | 28 | 04 | 4C | 00 | 0A | 00 | F0 | 00 | 0A | | | .L...... | |
| 12010 | 61.51816909s | | | | | | 00 | 1E | 00 | 00 | 00 | 00 | 00 | 00 | | | ........ | |
| 12011 | 61.51817709s | | | | | | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | | | ........ | |
| 12012 | 61.51818509s | | | | | | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 4BE2A6 | | ........ | |
| 12013 | 61.51821979s | 00111 | 1 | 16 | 0F2 | 28 | 00 | 21 | 00 | F0 | 00 | 22 | 00 | 14 | | | .!...\".. | |
| 12014 | 61.51823279s | | | | | | FE | 0C | 00 | 00 | 00 | 00 | 00 | 0B | | | ........ | |
| 12015 | 61.51824079s | | | | | | 00 | 0B | 00 | 00 | 00 | 00 | 00 | 28 | | | .......( | |
| 12016 | 61.51824879s | | | | | | 00 | 00 | 00 | 00 | 0D | DC | 00 | 27 | B4A469 | | .......' | |
| 12017 | 61.518369835s | 00111 | 4 | 16 | 6D3 | 28 | FC | 18 | 00 | 00 | 00 | 00 | 00 | 00 | | | ........ | |

## HD Audio

High Definition Audio, also known as HD Audio or by its codename, Azalia, is an audio standard created by Intel to be used on their chipsets, i.e., it is a standard for high-quality on-board audio. In this tutorial we will explain more about this feature.

### Settings



HD Audio Settings

Setting

SYNC: CH 0
I/O 0: CH 3
BCLK: CH 1
Direction: ○ SDO ● SDI

Color

Stream Data
Preamble
Stream ID
Length
Sample

Response (SDI)
Valid
UnSol
Reserved
Response

Command (SDO)
Reserved
CAd
NID
Verb ID
Payload

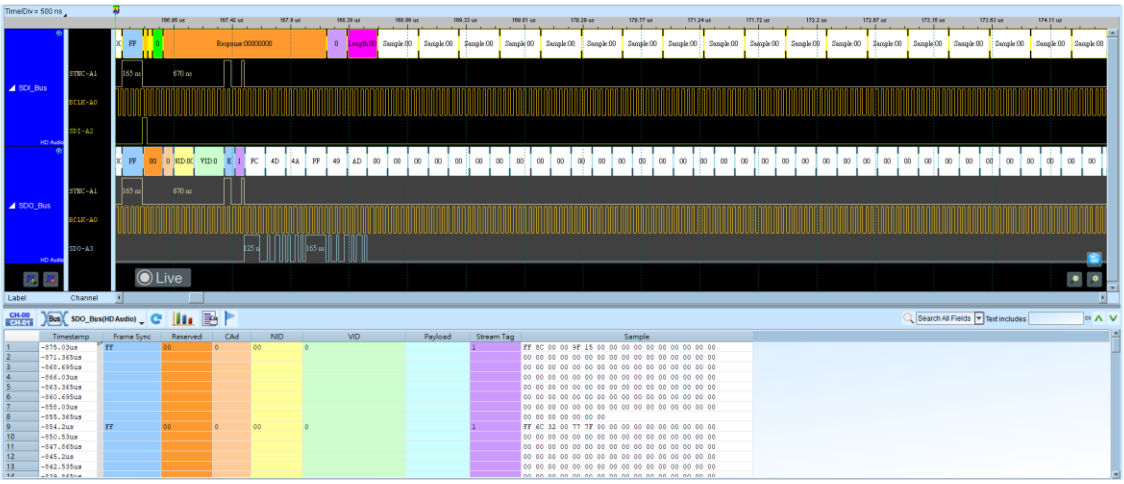
Range

Decode Range
From: Buffer Head
To: Buffer Tail

**Channel:** Show the selected channel.

**Direction:** Determines whether the I/O data decoding is SDI or SDO; this selection affects the rules of the analysis and causes the color-coded options in the lower field to change.

### Result

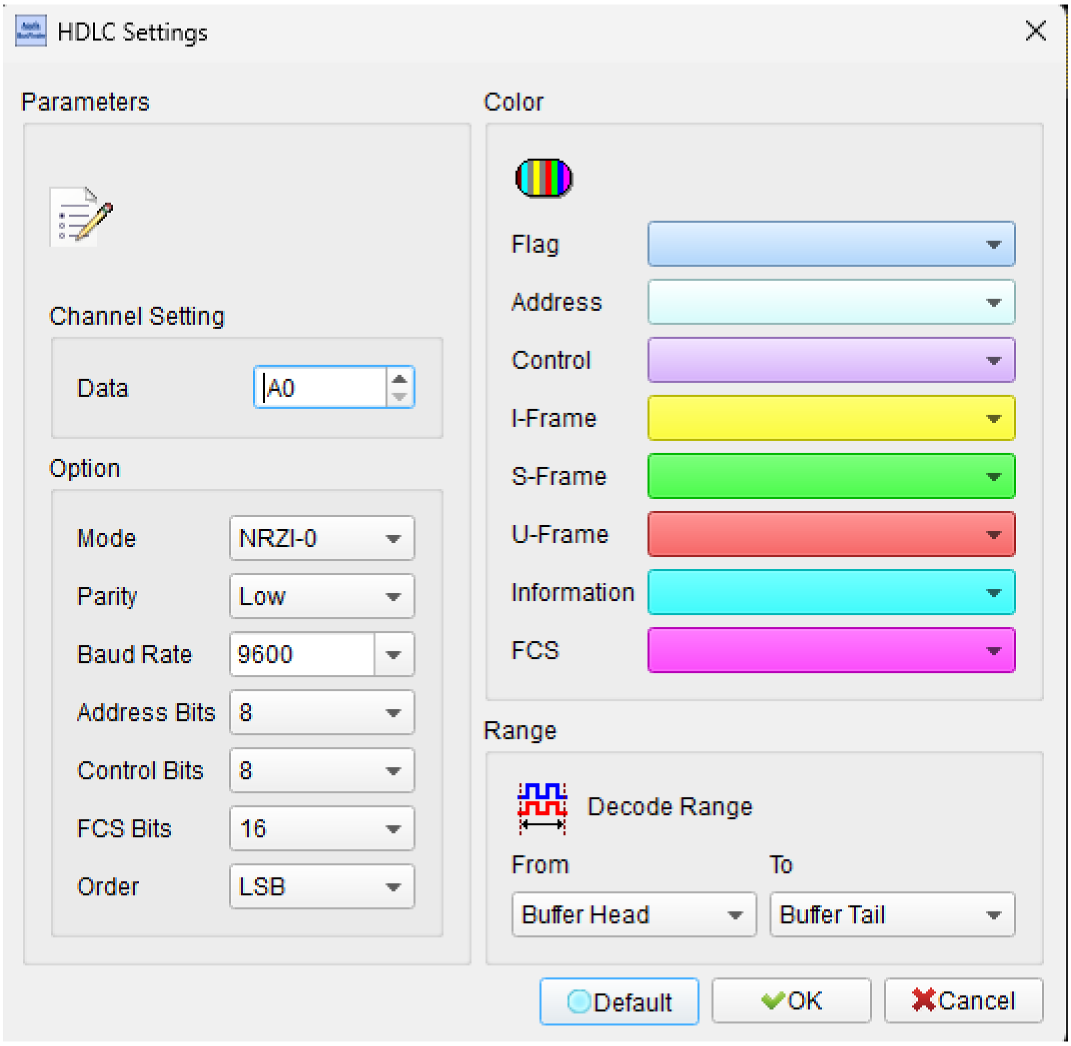


| | Timestamp | Frame Sync | Reserved | CAd | NID | VID | Payload | Stream Tag | Sample |
|---|---|---|---|---|---|---|---|---|---|
| 1 | -875.03us | FF | 00 | 0 | 00 | 0 | | 1 | FF 8C 00 00 9F 15 00 00 00 00 00 00 00 00 00 00 |
| 2 | -871.365us | | | | | | | | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 3 | -868.695us | | | | | | | | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 4 | -866.03us | | | | | | | | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 5 | -863.365us | | | | | | | | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 6 | -860.695us | | | | | | | | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 7 | -858.03us | | | | | | | | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 8 | -855.365us | | | | | | | | 00 00 00 00 00 00 00 |
| 9 | -854.2us | FF | 00 | 0 | 00 | 0 | | 1 | FF 6C 32 00 77 7F 00 00 00 00 00 00 00 00 00 00 |
| 10 | -850.53us | | | | | | | | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 11 | -847.865us | | | | | | | | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 12 | -845.2us | | | | | | | | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 13 | -842.535us | | | | | | | | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |

## HDLC

HDLC (High-level Data Link Control) is the default synchronous data link layer protocol used in the equipment of Cisco.

## Settings



HDLC Settings

Parameters

Channel Setting

Data: A0

Option

Mode: NRZI-0

Parity: Low

Baud Rate: 9600

Address Bits: 8

Control Bits: 8

FCS Bits: 16

Order: LSB

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**HDLC:** Set the channel of the signal.

**Option:** Start up Setting of signal analysis

I. **Mode:** Choose the analysis method.

II. **Parity:** Set Parity (High / Low)

III. **Baud Rate:** Set Baud rate

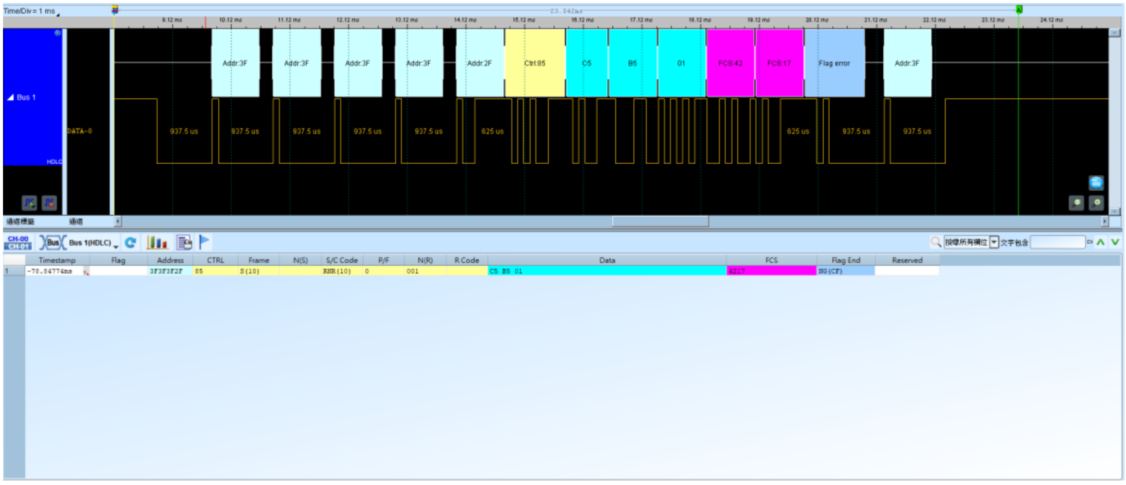
IV. **Address Bits:** Set the length of Address Bits.

V. **Control Bits:** Set the length of Control Bits.

**VI. FCS Bits:** Set the length of FCS (Frame Check Sequence) Bits**.**

VII. **Order:** Bit order of transmission.

**Result**

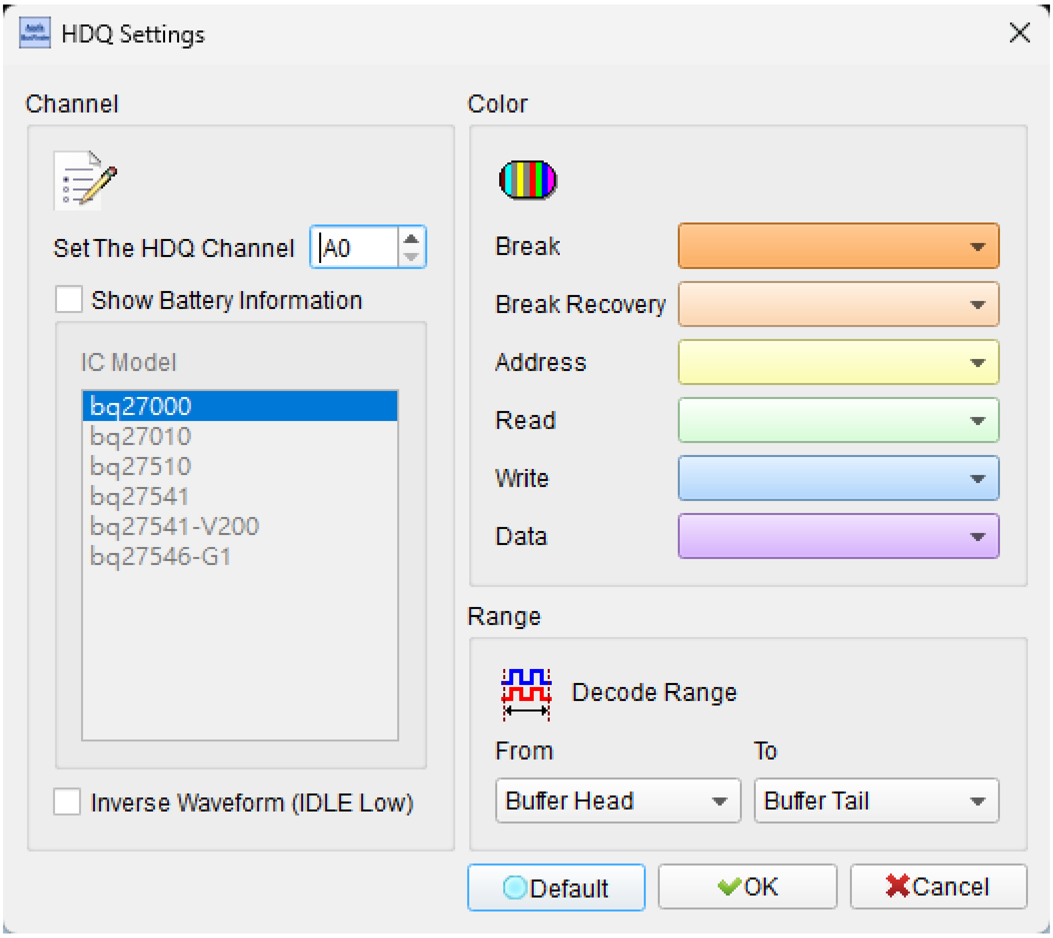


| | Timestamp | Flag | Address | CTRL | Frame | N(S) | S/C Code | P/F | N(R) | R Code | Data | FCS | Flag End | Reserved |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -78.84774ms | | 3F3F3F2F | 85 | S(10) | | RNR(10) | 0 | 001 | | C5 B5 01 | 4217 | NG(CF) | |

## HDQ

Developed by TEXAS INSTRUMENTS for battery management display applications, primarily in consumer electronics, HDQ is available in two data width formats, 8-bit and 16-bit, with a fixed 7-bit address. An HDQ packet mainly consists of Break, 7 bits Address, 1 bit R/W and 8 bits Data or 16 bits Data. The transmission method is LSB (Least-significant bit) to MSB (Most-significant bit), and the maximum transmission rate is 5 Kbit/s. The maximum transmission rate is 5Kbit/s.

### Settings



HDQ Settings

Channel

Set The HDQ Channel: A0

□ Show Battery Information

IC Model:
- bq27000 (selected)
- bq27010
- bq27510
- bq27541
- bq27541-V200
- bq27546-G1

□ Inverse Waveform (IDLE Low)

Color:
- Break 🟧
- Break Recovery 🟧
- Address 🟨
- Read 🟩
- Write 🟦
- Data 🟪

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the HDQ channel: Show the selected channel (CH 0).

**Show Battery Information:** monitor the command between battery and IC.

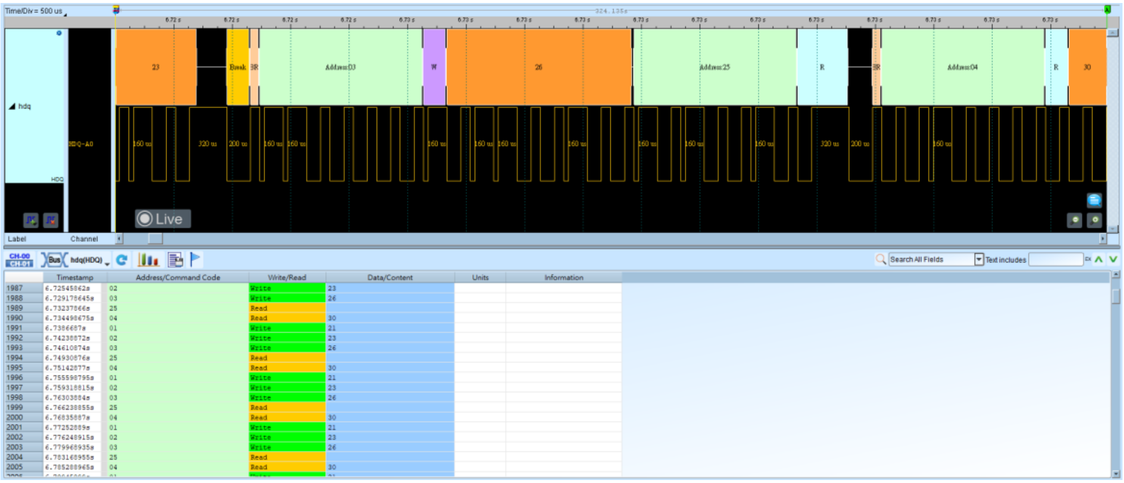
**Invese Waveform (IDLE Low): Invert the waveform. Enabled when**

**checked.**

**Result**

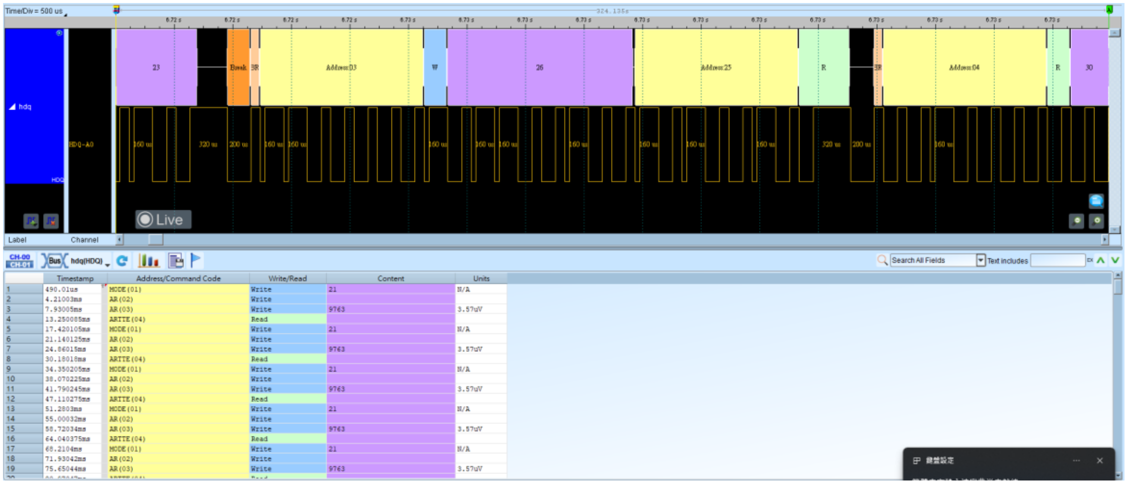
**Write:** Indicates that the data is written and followed by the data.

**Read:** Read Indicates that the data is read, followed by the data.



| | Timestamp | Address/Command Code | Write/Read | Data/Content | Units | Information |
|---|---|---|---|---|---|---|
| 1987 | 6.72545862s | 02 | Write | 23 | | |
| 1988 | 6.729178645s | 03 | Write | 26 | | |
| 1989 | 6.73237866s | 25 | Read | | | |
| 1990 | 6.734498675s | 04 | Read | 30 | | |
| 1991 | 6.7386687s | 01 | Write | 21 | | |
| 1992 | 6.74238872s | 02 | Write | 23 | | |
| 1993 | 6.74610874s | 03 | Write | 26 | | |
| 1994 | 6.74930876s | 25 | Read | | | |
| 1995 | 6.75142877s | 04 | Read | 30 | | |
| 1996 | 6.755598795s | 01 | Write | 21 | | |
| 1997 | 6.759318815s | 02 | Write | 23 | | |
| 1998 | 6.76303884s | 03 | Write | 26 | | |
| 1999 | 6.766238855s | 25 | Read | | | |
| 2000 | 6.76835887s | 04 | Read | 30 | | |
| 2001 | 6.77252889s | 01 | Write | 21 | | |
| 2002 | 6.776248915s | 02 | Write | 23 | | |
| 2003 | 6.779968935s | 03 | Write | 26 | | |
| 2004 | 6.783168955s | 25 | Read | | | |
| 2005 | 6.785288965s | 04 | Read | 30 | | |

## Show the battery information

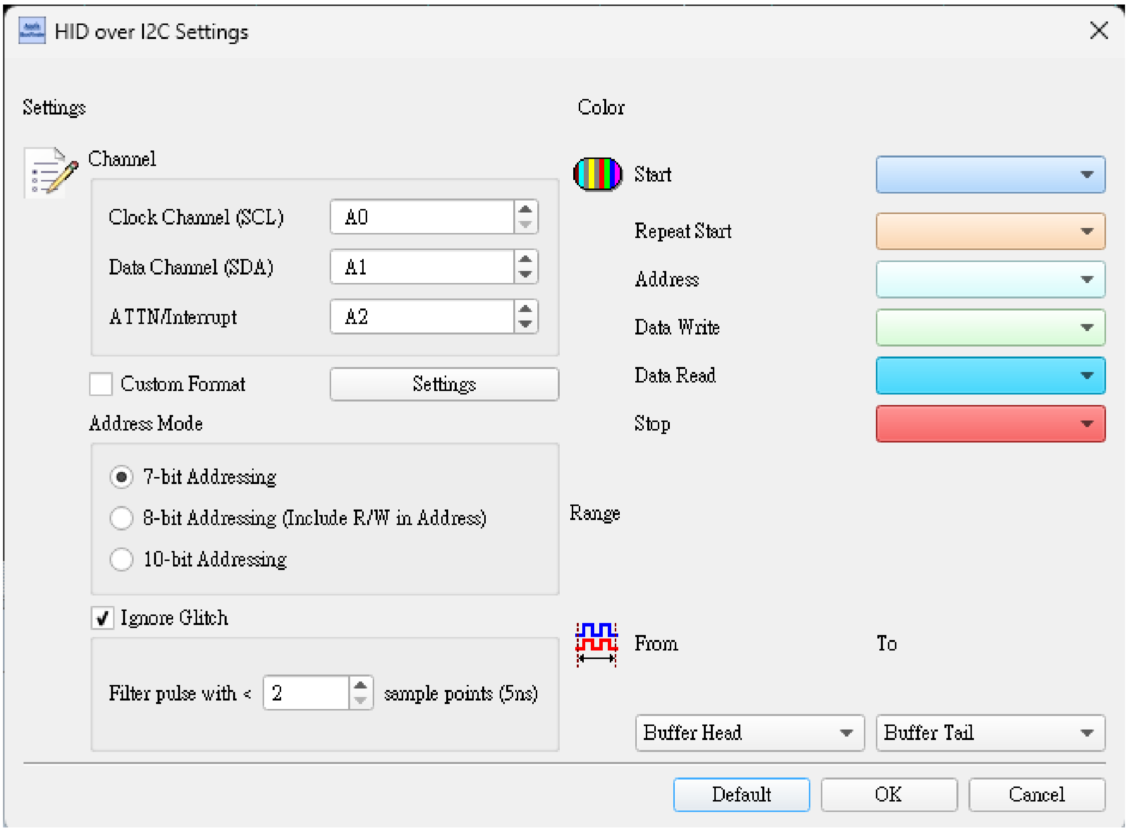


| | Timestamp | Address/Command Code | Write/Read | Content | Units |
|---|---|---|---|---|---|
| 1 | 490.01us | MODE(01) | Write | 21 | N/A |
| 2 | 4.21003ms | AR(02) | Write | | |
| 3 | 7.93005ms | AR(03) | Write | 9763 | 3.57uV |
| 4 | 13.250085ms | ARTTE(04) | Read | | |
| 5 | 17.420105ms | MODE(01) | Write | 21 | N/A |
| 6 | 21.140125ms | AR(02) | Write | | |
| 7 | 24.86015ms | AR(03) | Write | 9763 | 3.57uV |
| 8 | 30.18018ms | ARTTE(04) | Read | | |
| 9 | 34.350205ms | MODE(01) | Write | 21 | N/A |
| 10 | 38.070225ms | AR(02) | Write | | |
| 11 | 41.790245ms | AR(03) | Write | 9763 | 3.57uV |
| 12 | 47.110275ms | ARTTE(04) | Read | | |
| 13 | 51.2803ms | MODE(01) | Write | 21 | N/A |
| 14 | 55.00032ms | AR(02) | Write | | |
| 15 | 58.72034ms | AR(03) | Write | 9763 | 3.57uV |
| 16 | 64.040375ms | ARTTE(04) | Read | | |
| 17 | 68.2104ms | MODE(01) | Write | 21 | N/A |
| 18 | 71.93042ms | AR(02) | Write | | |
| 19 | 75.65044ms | AR(03) | Write | 9763 | 3.57uV |

### HID Over I$^2$C

HID Over I2C (Human Interface Device Over I2C) is mainly used in Windows 8, ARM platform architecture; the other is HID Over USB is used in x86 system, in Windows 8 common support for HID Over I2C bus protocol device is touch pad.

### Settings



HID over I2C Settings

Settings

Channel

Clock Channel (SCL) A0

Data Channel (SDA) A1

ATTN/Interrupt A2

□ Custom Format

Address Mode

● 7-bit Addressing

○ 8-bit Addressing (Include R/W in Address)

○ 10-bit Addressing

☑ Ignore Glitch

Filter pulse with < 2 sample points (5ns)

Color

Start

Repeat Start

Address

Data Write

Data Read

Stop

Range

From Buffer Head

To Buffer Tail

**Channel:** Show the selected channels (SCL:CH0, SDA:CH1, ATTN:CH2).

**Custom Format**: Customizable partial analysis content.

**Setting:** Click on it to bring up Notepad to edit the decoding format, the format is as follows: CMD, {Name of parse field 1, Number of bytes in parse field 1, Arrangement of parse field 1}, ......

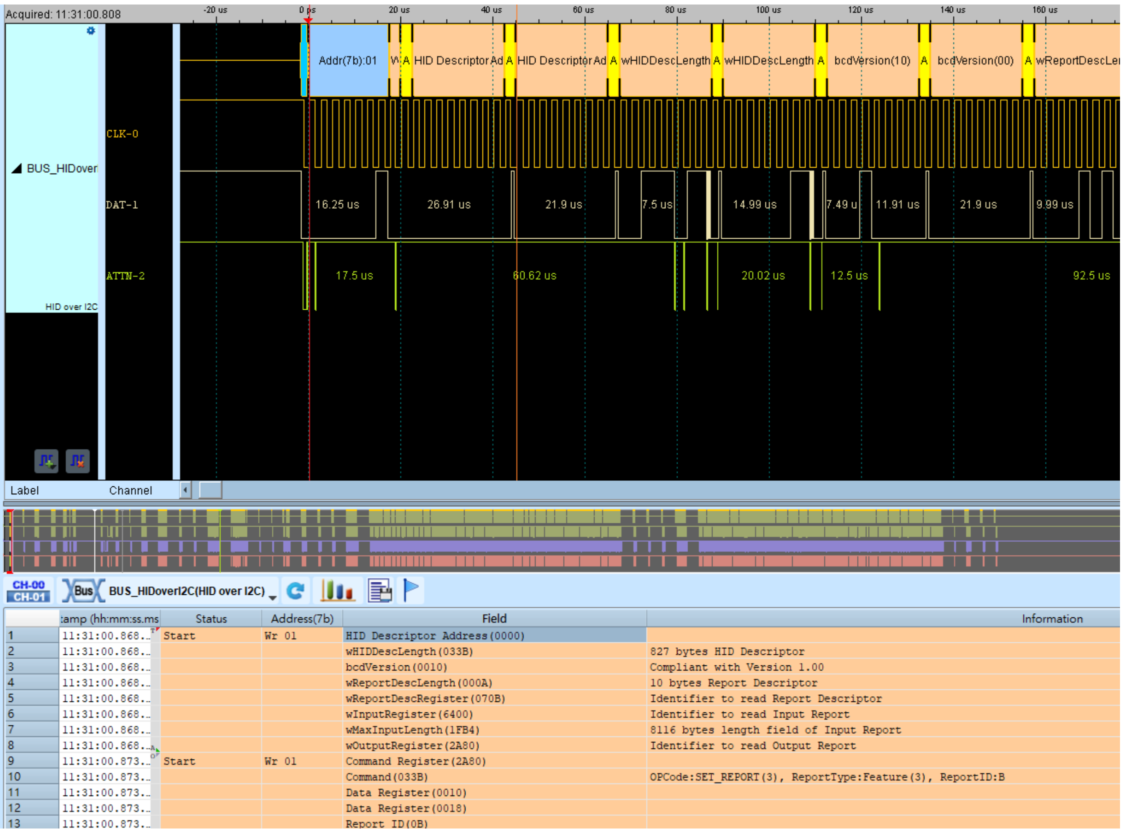
**7-bit addressing:** Show 7-bit addressing.

**8-bit addressing (Include R/W in Address):** Show 8-bit addressing (include 7-bit addressing and 1-bit Rd/Wr).

**10-bit addressing:** show 10-bit addressing.

**Ignore glitch:** Ignore the glitches occurred due to the slow transitions. Enabled when checked.

## Result

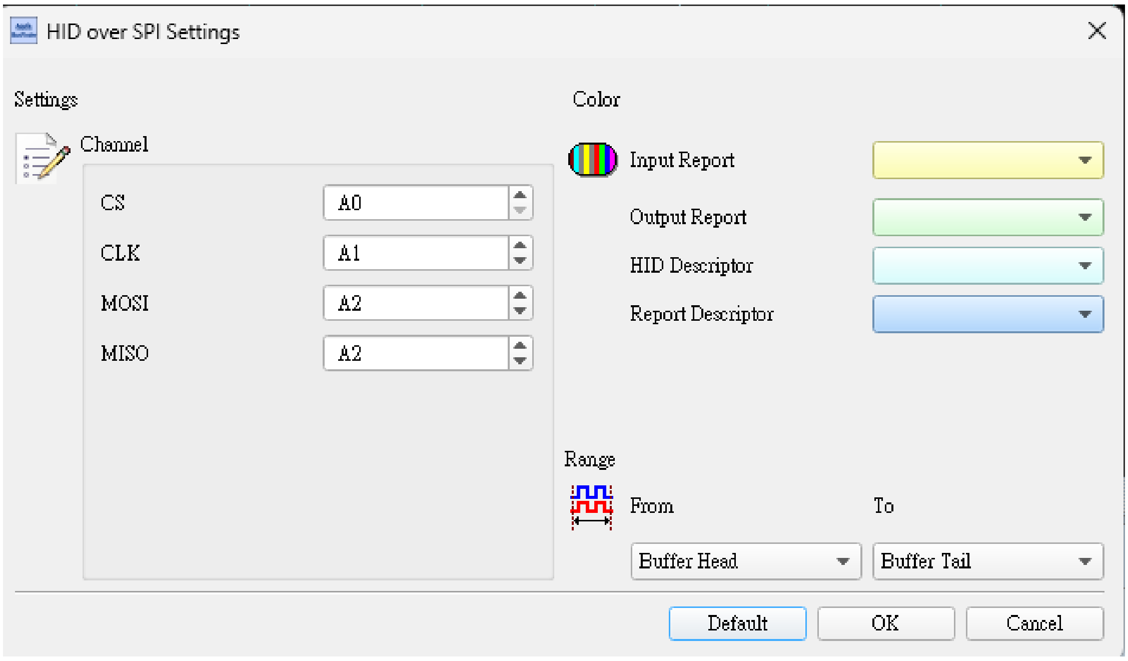


| | tamp (hh:mm:ss.ms | Status | Address(7b) | Field | Information |
|---|---|---|---|---|---|
| 1 | 11:31:00.868... | Start | Wr 01 | HID Descriptor Address(0000) | |
| 2 | 11:31:00.868... | | | wHIDDescLength(033B) | 827 bytes HID Descriptor |
| 3 | 11:31:00.868... | | | bcdVersion(0010) | Compliant with Version 1.00 |
| 4 | 11:31:00.868... | | | wReportDescLength(000A) | 10 bytes Report Descriptor |
| 5 | 11:31:00.868... | | | wReportDescRegister(070B) | Identifier to read Report Descriptor |
| 6 | 11:31:00.868... | | | wInputRegister(6400) | Identifier to read Input Report |
| 7 | 11:31:00.868... | | | wMaxInputLength(1FB4) | 8116 bytes length field of Input Report |
| 8 | 11:31:00.868... | | | wOutputRegister(2A80) | Identifier to read Output Report |
| 9 | 11:31:00.873... | Start | Wr 01 | Command Register(2A80) | |
| 10 | 11:31:00.873... | | | Command(033B) | OPCode:SET_REPORT(3), ReportType:Feature(3), ReportID:B |
| 11 | 11:31:00.873... | | | Data Register(0010) | |
| 12 | 11:31:00.873... | | | Data Register(0018) | |
| 13 | 11:31:00.873... | | | Report ID(0B) | |

## HID Over SPI

HID Over SPI (Human Interface Device Over SPI) protocol is established by Microsoft. It’s applied for Windows 8 ARM platform.

### Settings



HID over SPI Settings

Settings

Channel

| | |
|---|---|
| CS | A0 |
| CLK | A1 |
| MOSI | A2 |
| MISO | A2 |

Color

- Input Report
- Output Report
- HID Descriptor
- Report Descriptor

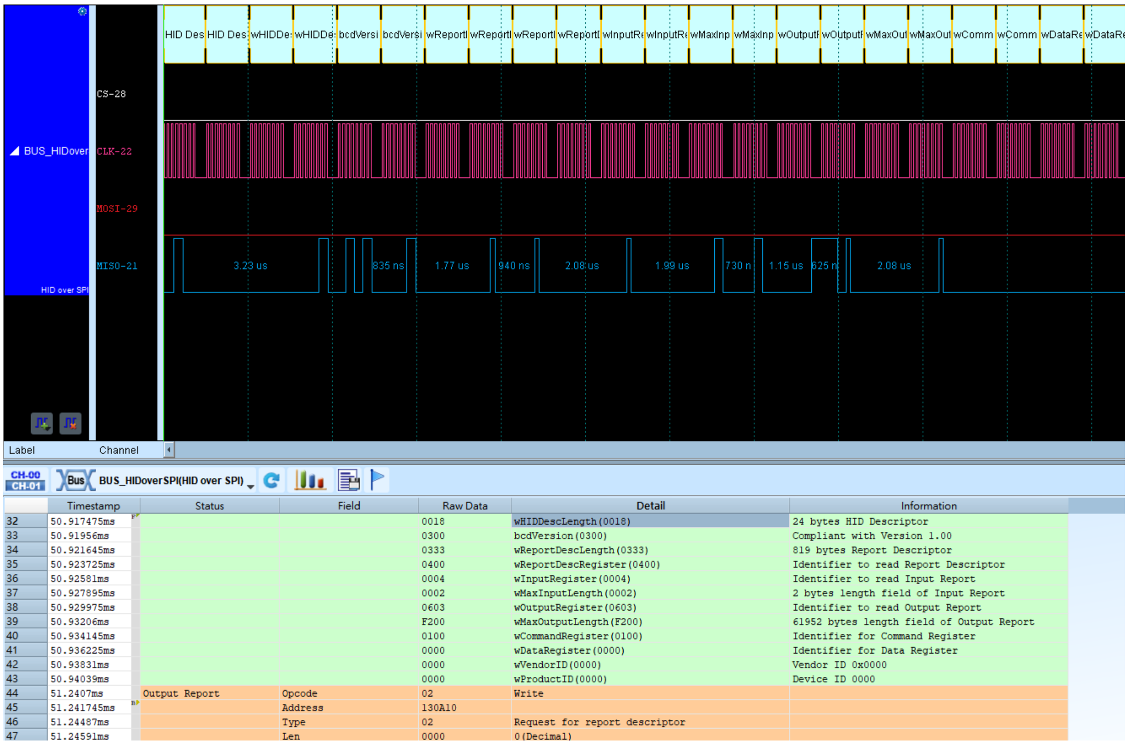
Range

From: Buffer Head

To: Buffer Tail

**Channel:** Show the selected channels (CS, CLK, MOSI, MISO).

### Result



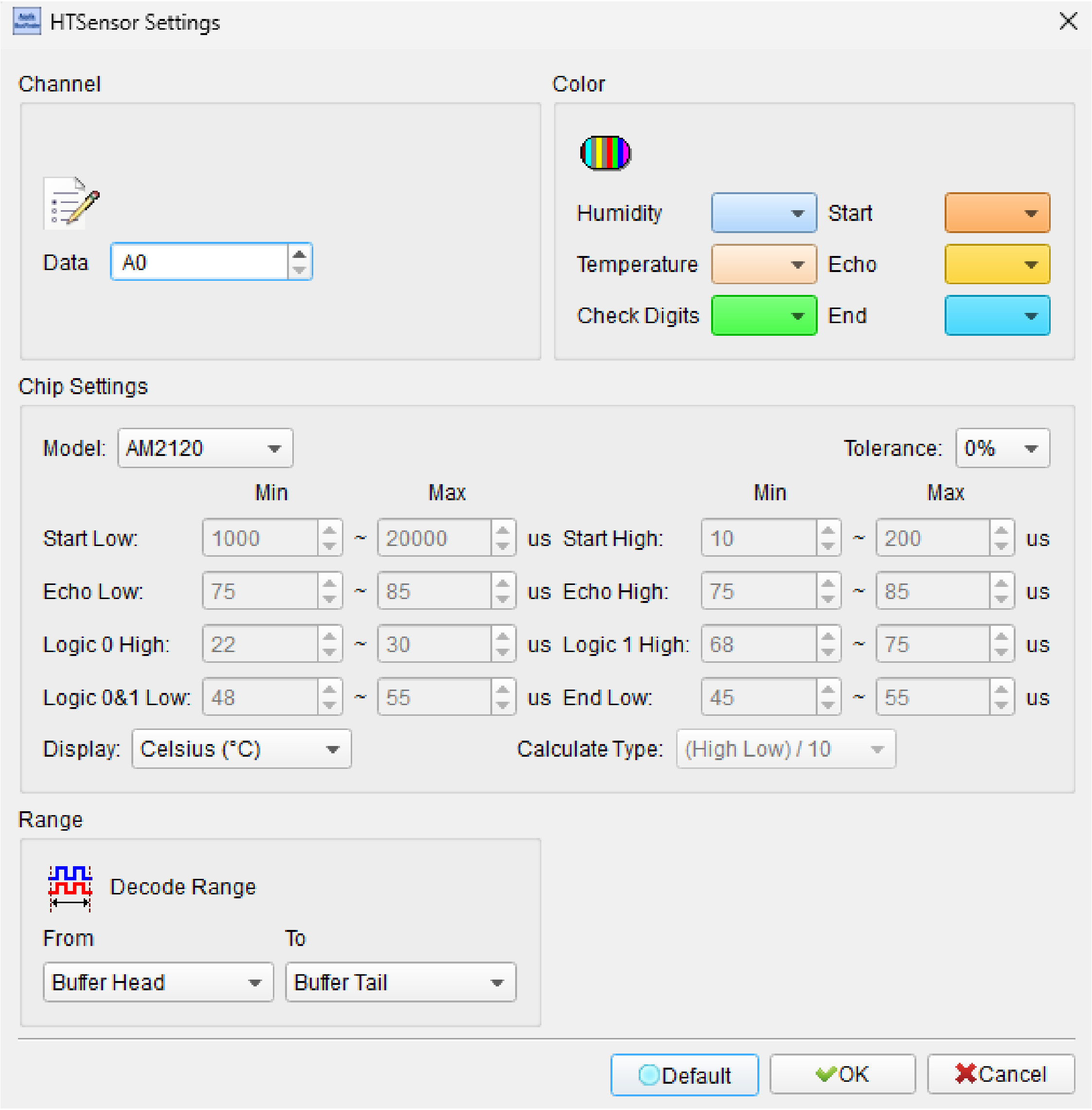
BUS_HIDoverSPI(HID over SPI)

| | Timestamp | Status | Field | Raw Data | Detail | Information |
|---|---|---|---|---|---|---|
| 32 | 50.917475ms | | | 0018 | wHIDDescLength(0018) | 24 bytes HID Descriptor |
| 33 | 50.91956ms | | | 0300 | bcdVersion(0300) | Compliant with Version 1.00 |
| 34 | 50.921645ms | | | 0333 | wReportDescLength(0333) | 819 bytes Report Descriptor |
| 35 | 50.923725ms | | | 0400 | wReportDescRegister(0400) | Identifier to read Report Descriptor |
| 36 | 50.92581ms | | | 0004 | wInputRegister(0004) | Identifier to read Input Report |
| 37 | 50.927895ms | | | 0002 | wMaxInputLength(0002) | 2 bytes length field of Input Report |
| 38 | 50.929975ms | | | 0603 | wOutputRegister(0603) | Identifier to read Output Report |
| 39 | 50.93206ms | | | F200 | wMaxOutputLength(F200) | 61952 bytes length field of Output Report |
| 40 | 50.934145ms | | | 0100 | wCommandRegister(0100) | Identifier for Command Register |
| 41 | 50.936225ms | | | 0000 | wDataRegister(0000) | Identifier for Data Register |
| 42 | 50.93831ms | | | 0000 | wVendorID(0000) | Vendor ID 0x0000 |
| 43 | 50.94039ms | | | 0000 | wProductID(0000) | Device ID 0000 |
| 44 | 51.2407ms | Output Report | Opcode | 02 | Write | |
| 45 | 51.241745ms | | Address | 130A10 | | |
| 46 | 51.24487ms | | Type | 02 | Request for report descriptor | |
| 47 | 51.24591ms | | Len | 0000 | 0(Decimal) | |

## HTSensor

An HTSensor is a sensor designed to measure and monitor environmental conditions. It typically measures environment-related parameters such as temperature and humidity and provides the corresponding data to a system or device for processing or control. These sensors are widely used in a variety of devices and systems, such as smart homes, automation systems, climate control systems, medical devices, and so on.

## Setting



HTSensor Settings

Channel

Data: A0

Color

| | | | |
|---|---|---|---|
| Humidity | 🟦 | Start | 🟧 |
| Temperature | 🟧 | Echo | 🟨 |
| Check Digits | 🟩 | End | 🟦 |

Chip Settings

Model: AM2120 Tolerance: 0%

| | Min | | Max | | | Min | | Max | |
|---|---|---|---|---|---|---|---|---|---|
| Start Low: | 1000 | ~ | 20000 | us | Start High: | 10 | ~ | 200 | us |
| Echo Low: | 75 | ~ | 85 | us | Echo High: | 75 | ~ | 85 | us |
| Logic 0 High: | 22 | ~ | 30 | us | Logic 1 High: | 68 | ~ | 75 | us |
| Logic 0&1 Low: | 48 | ~ | 55 | us | End Low: | 45 | ~ | 55 | us |

Display: Celsius (°C) Calculate Type: (High Low) / 10

Range

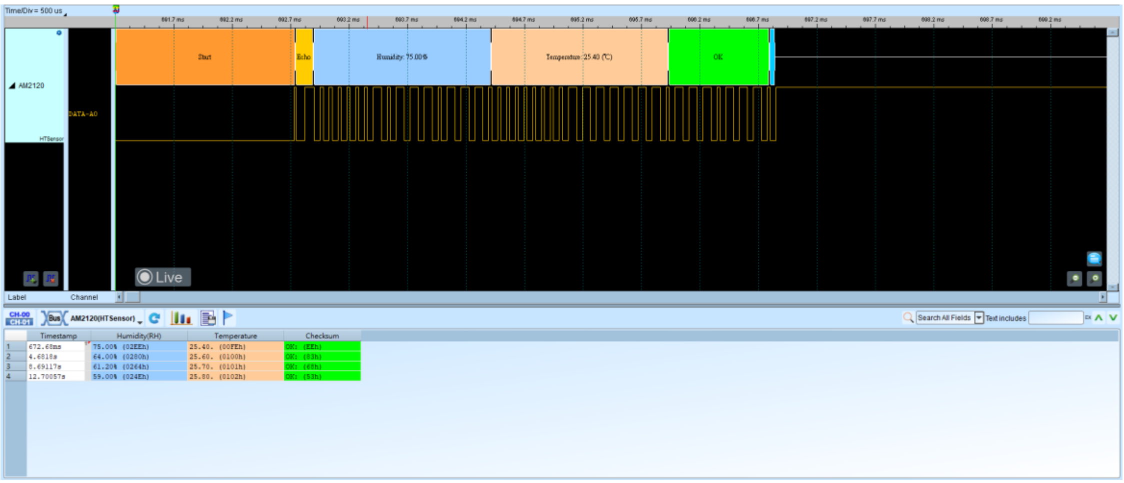
Decode Range

From: Buffer Head To: Buffer Tail

**Data:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Chip Setting:** Configure supported chip models or user-defined chip parameter details.

### Result

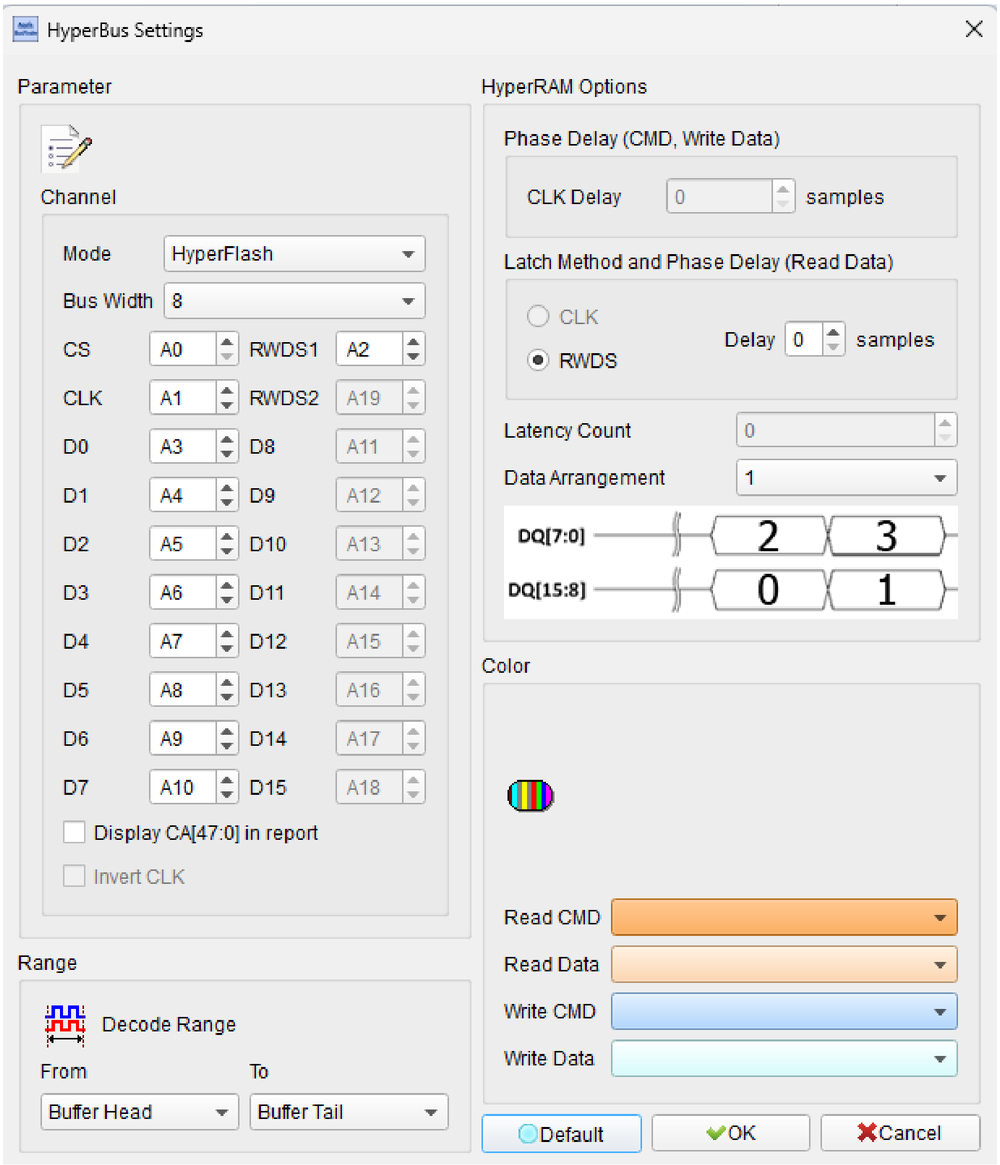


|  | Timestamp | Humidity(RH) | Temperature | Checksum |
|---|---|---|---|---|
| 1 | 672.68ms | 75.00% (02EEh) | 25.40. (00FEh) | OK: (EEh) |
| 2 | 4.6818s | 64.00% (0280h) | 25.60. (0100h) | OK: (83h) |
| 3 | 8.69117s | 61.20% (0264h) | 25.70. (0101h) | OK: (68h) |
| 4 | 12.70057s | 59.00% (024Eh) | 25.80. (0102h) | OK: (53h) |

## HyperBus

HyperBus is a high-performance memory interface technology designed to increase data transfer rates, especially in embedded systems such as smartphones, tablets, and IoT devices. Proposed by Micron Technology and adopted by several device manufacturers, HyperBus technology can support many different types of memory, most commonly used for connecting Flash memory and DRAM (Dynamic Random Access Memory), while providing faster data read and write speeds than traditional SPI and parallel memory interfaces.

## Settings



HyperBus Settings

Parameter

Channel

Mode: HyperFlash
Bus Width: 8

| | | | |
|---|---|---|---|
| CS | A0 | RWDS1 | A2 |
| CLK | A1 | RWDS2 | A19 |
| D0 | A3 | D8 | A11 |
| D1 | A4 | D9 | A12 |
| D2 | A5 | D10 | A13 |
| D3 | A6 | D11 | A14 |
| D4 | A7 | D12 | A15 |
| D5 | A8 | D13 | A16 |
| D6 | A9 | D14 | A17 |
| D7 | A10 | D15 | A18 |

□ Display CA[47:0] in report
□ Invert CLK

Range

Decode Range
From: Buffer Head
To: Buffer Tail

HyperRAM Options

Phase Delay (CMD, Write Data)
CLK Delay 0 samples

Latch Method and Phase Delay (Read Data)
○ CLK
● RWDS
Delay 0 samples

Latency Count: 0
Data Arrangement: 1

Color

Read CMD
Read Data
Write CMD
Write Data

**Mode:** Switching mode, can be set to Hyper Flash or Hyper Ram.

**Bus Width:** Setting the bus width, can be set to 8 or 16.

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Display CA[47:0] in report:** Displays CA[47:0] total 48bit information in the report. Available only when the mode is switched to HyperFlash. Enabled when checked.

**Invert CLK:** Invert CLK: Available only when mode is switched to HyperRam. Enabled when checked.

**HyperRAM Options:**

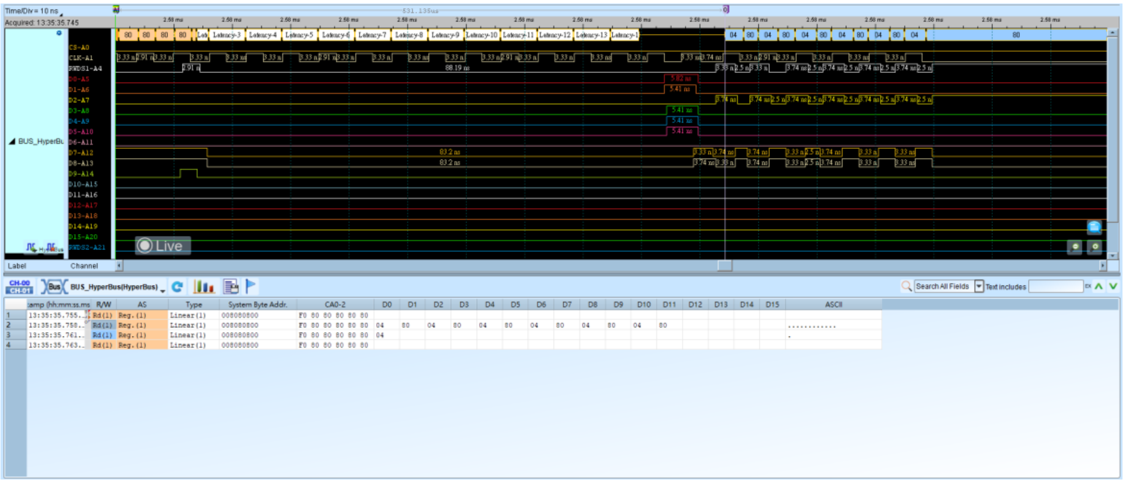
I. **Phase Delay (CMD, Write Data):** Sets the number of sample points for Delay, valid only when the mode is switched to HyperRam.

**II.** **Latch Method and Phase Delay (Read Data):** Set the number of sample points for CLK (valid only when mode is switched to HyperRam) or RWDS Delay.

**III.** **Latency Count:** Sets the number of Delay CLKs, valid only when the mode is switched to HyperRam.

**IV.** **Data Arrangement:** Sets the mode of the Data Arrangement to one of two modes.

## Result



Time/Div = 10 ns
Acquired: 13:35:35.745

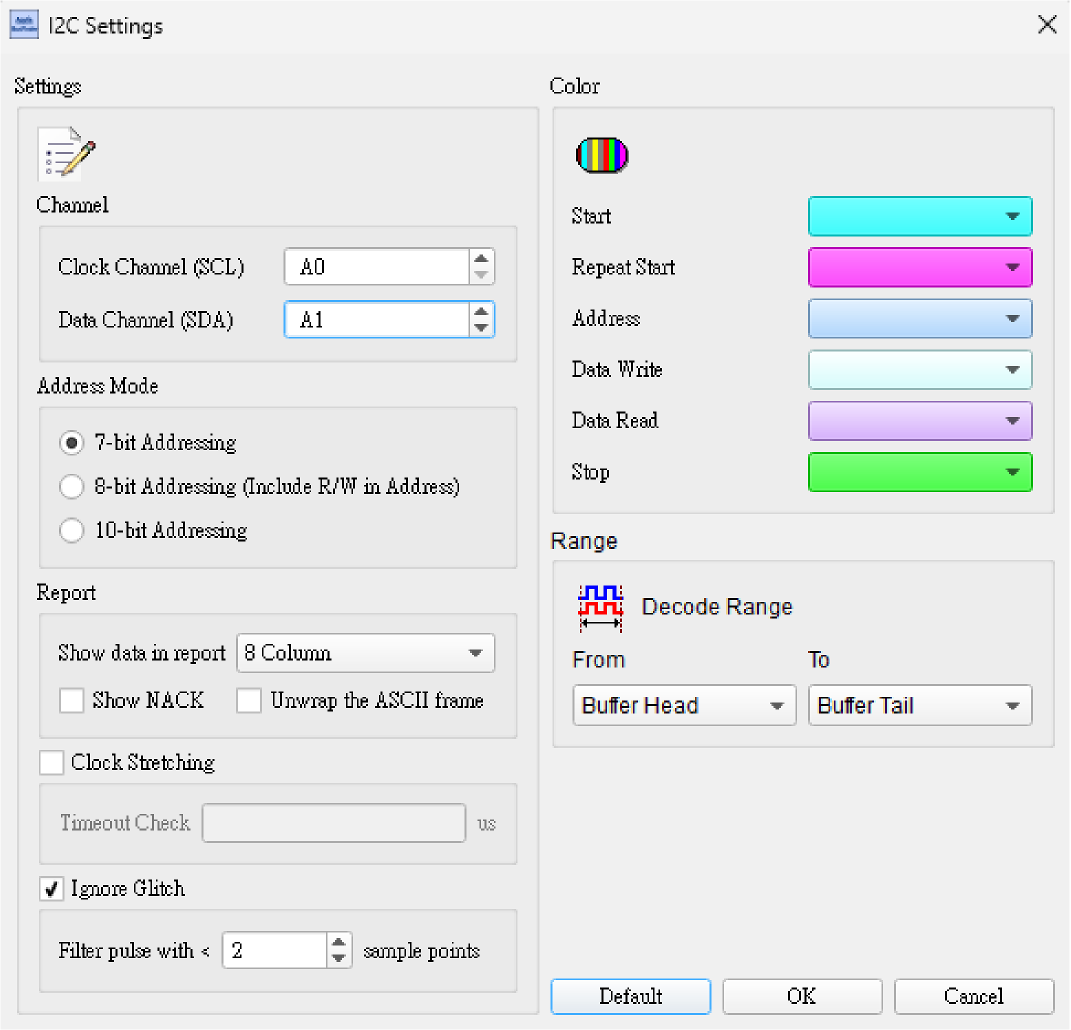
BUS_HyperBus(HyperBus)

| | amp (hh:mm:ss.ms | R/W | AS | Type | System Byte Addr. | CA0-2 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 13:35:35.755.. | Rd(1) | Reg.(1) | Linear(1) | 008080800 | F0 80 80 80 80 80 | | | | | | | | | | | | | | | | | |
| 2 | 13:35:35.758.. | Rd(1) | Reg.(1) | Linear(1) | 008080800 | F0 80 80 80 80 80 | 04 | 80 | 04 | 80 | 04 | 80 | 04 | 80 | 04 | 80 | 04 | 80 | | | | | ............ |
| 3 | 13:35:35.761.. | Rd(1) | Reg.(1) | Linear(1) | 008080800 | F0 80 80 80 80 80 | 04 | | | | | | | | | | | | | | | | . |
| 4 | 13:35:35.763.. | Rd(1) | Reg.(1) | Linear(1) | 008080800 | F0 80 80 80 80 80 | | | | | | | | | | | | | | | | | |

## I²C

I2C is a two-wire serial communications bus, using a multi-slave architecture, developed by Philips in the 1980s to enable motherboards, embedded systems, or cell phones to connect to low-speed peripheral devices, and is a commonly used type of electronic circuit system. I2C uses only two bi-directional signal lines, one clock line (SCL) and one data line (SDA). Signal content has a total of start (Start), address (Address), data (Data), read/write (Read/Write), etc., the transmission method is bidirectional, the data format is divided into two kinds of 8 bits and 10 bits. The transmission rate is 100kbit/s-3.4Mbit/s. The data format is divided into 8 bits and 10 bits.

### Settings



I2C Settings

Settings

Channel

Clock Channel (SCL): A0

Data Channel (SDA): A1

Address Mode

● 7-bit Addressing

○ 8-bit Addressing (Include R/W in Address)

○ 10-bit Addressing

Report

Show data in report: 8 Column

□ Show NACK □ Unwrap the ASCII frame

□ Clock Stretching

Timeout Check: us

☑ Ignore Glitch

Filter pulse with < 2 sample points

Color

Start

Repeat Start

Address

Data Write

Data Read

Stop

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Clock Channel (SCL):** Transfer clock of I$^2$C.

**Data Channel (SDA):** Transfer data of I$^2$C.

**Address Mode:**

I. **7-bit addressing:** Displays the address in 7-bit width and Rd/Wr in 1-bit width.

II. **8-bit addressing(Include R/W in Address):** Displays an 8-bit width address (7-bit width address plus 1-bit Rd/Wr).

III. **10-bit addressing:** Displays a 10-bit width address.

**Report:**

I. **Show data in report:** Displays the data in the report area with a choice of 8 or 16 fields.

II. **Show NACK:** Marks the Byte as NACK in the field. enabled when checked.

III. **Unwrap the ASCII frame:** Add “ASCII” field in the report area. Enabled when checked.

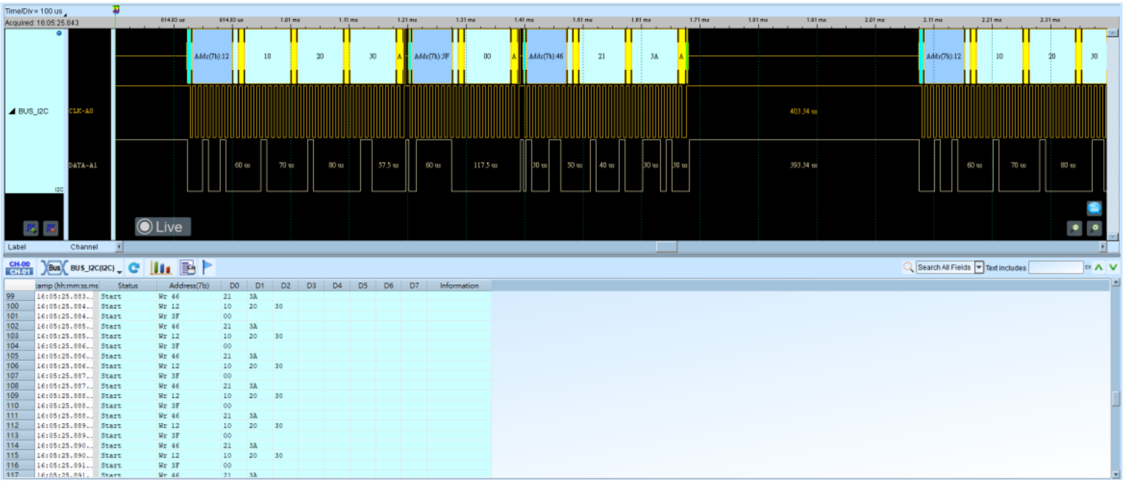
**Clock Stretching:** Set the length of time for Clock Streching. Enabled when checked.

**Ignore Glitch:** Ignore glitch caused by slow transitions when analyzing. Enabled when checked.

## Result

## Wr: Indicates that the data is written.

## Rd: Indicates reading data.

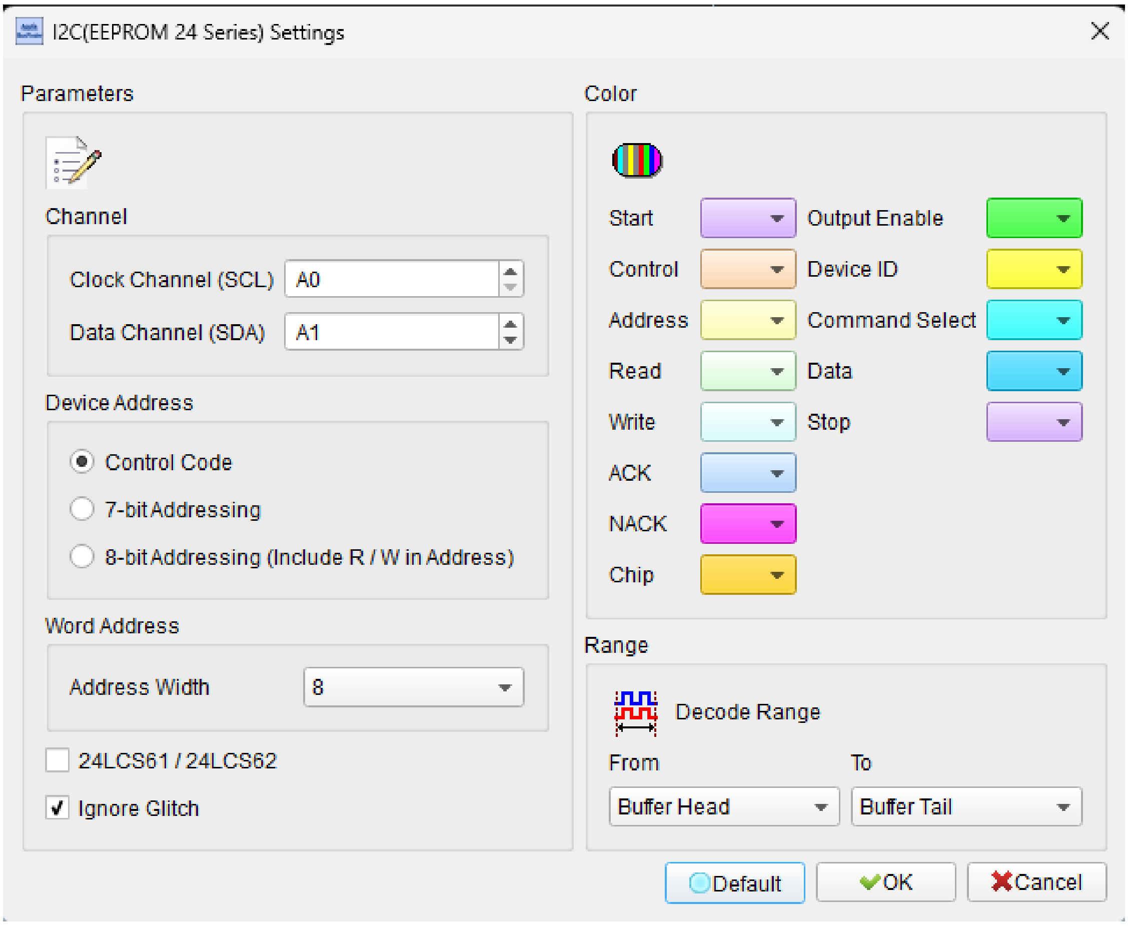


| | tamp (hh:mm:ss.ms | Status | Address(7b) | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 99 | 16:05:25.883... | Start | Wr 46 | 21 | 3A | | | | | | | |
| 100 | 16:05:25.884... | Start | Wr 12 | 10 | 20 | 30 | | | | | | |
| 101 | 16:05:25.884... | Start | Wr 3F | 00 | | | | | | | | |
| 102 | 16:05:25.885... | Start | Wr 46 | 21 | 3A | | | | | | | |
| 103 | 16:05:25.885... | Start | Wr 12 | 10 | 20 | 30 | | | | | | |
| 104 | 16:05:25.886... | Start | Wr 3F | 00 | | | | | | | | |
| 105 | 16:05:25.886... | Start | Wr 46 | 21 | 3A | | | | | | | |
| 106 | 16:05:25.886... | Start | Wr 12 | 10 | 20 | 30 | | | | | | |
| 107 | 16:05:25.887... | Start | Wr 3F | 00 | | | | | | | | |
| 108 | 16:05:25.887... | Start | Wr 46 | 21 | 3A | | | | | | | |
| 109 | 16:05:25.888... | Start | Wr 12 | 10 | 20 | 30 | | | | | | |
| 110 | 16:05:25.888... | Start | Wr 3F | 00 | | | | | | | | |
| 111 | 16:05:25.888... | Start | Wr 46 | 21 | 3A | | | | | | | |
| 112 | 16:05:25.889... | Start | Wr 12 | 10 | 20 | 30 | | | | | | |
| 113 | 16:05:25.889... | Start | Wr 3F | 00 | | | | | | | | |
| 114 | 16:05:25.890... | Start | Wr 46 | 21 | 3A | | | | | | | |
| 115 | 16:05:25.890... | Start | Wr 12 | 10 | 20 | 30 | | | | | | |
| 116 | 16:05:25.891... | Start | Wr 3F | 00 | | | | | | | | |
| 117 | 16:05:25.891... | Start | Wr 46 | 21 | 3A | | | | | | | |

## I$^2$C EEPROM

EEPROM, or E$^2$PROM, is known as Electrically-Erasable Programmable Read-Only Memory, and the interface of EEPROM components can be classified into serial and parallel, with I$^2$C EEPROM belonging to the 2-wire serial EEPROM, and its model is a series starting with 24. I$^2$C EEPROM belongs to the 2-wire serial EEPROM, and its model number is a series starting with 24.

### Settings



I2C(EEPROM 24 Series) Settings

Parameters

Channel

Clock Channel (SCL): A0

Data Channel (SDA): A1

Device Address

● Control Code

○ 7-bit Addressing

○ 8-bit Addressing (Include R / W in Address)

Word Address

Address Width: 8

□ 24LCS61 / 24LCS62

☑ Ignore Glitch

Color

Start, Control, Address, Read, Write, ACK, NACK, Chip, Output Enable, Device ID, Command Select, Data, Stop

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Clock Channel (SCL):** Transfer clock of I$^2$C.

**Data Channel (SDA):** Transfer data of I$^2$C.

**Device Address:**

1. **Control Code:** Display Control Code.

IV. **7-bit Addressing:** Displays the address in 7-bit width and Rd/Wr in 1-bit width.

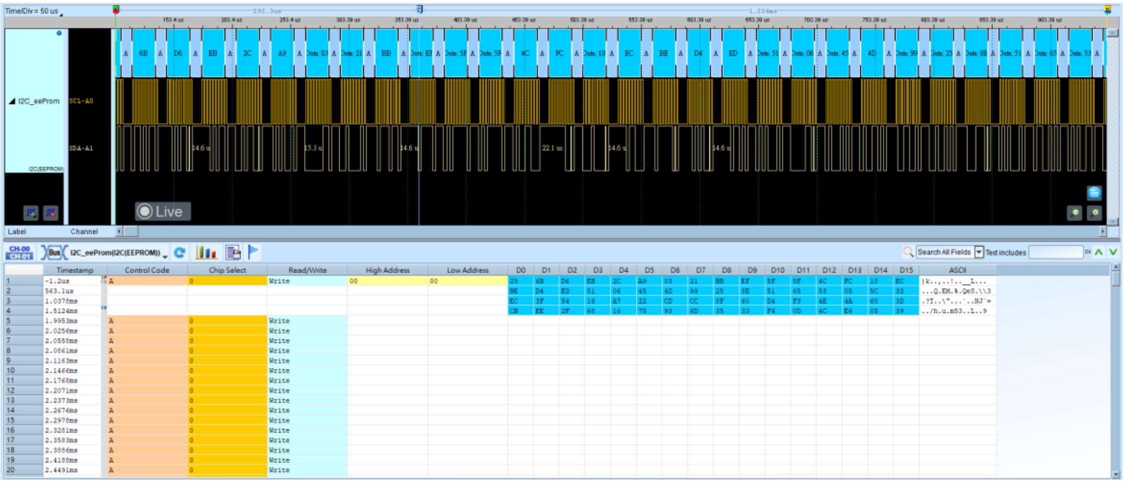
2. **8-bit Addressing (Include R/W in Address):** Displays an 8-bit width address (7-bit width address plus 1-bit Rd/Wr).

**Word Address:** Set the number of valid bits of I2C EEPROM address, the default value is 8.

**24LCS61 / 24LCS62:** Select whether to analyze 24LCS61 / 24LCS62. enabled when checked.

**Ignore Glitch:** Ignore glitch caused by slow transitions when analyzing. Enabled when checked.

**Result**

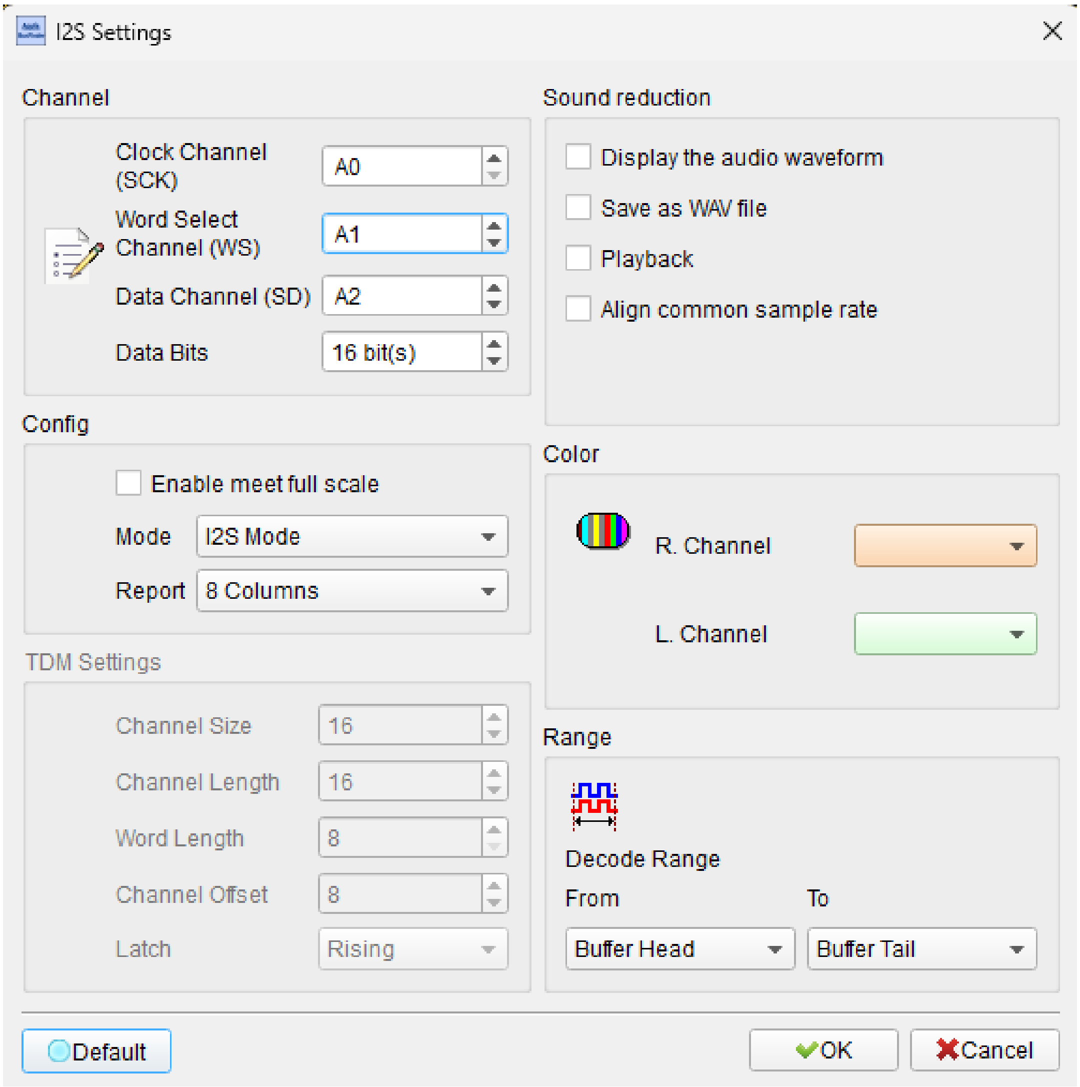


| | Timestamp | Control Code | Chip Select | Read/Write | High Address | Low Address | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -1.2us | A | 0 | Write | 00 | 00 | 29 | 6B | D6 | EB | 2C | A9 | 03 | 21 | BB | EF | 5F | 5F | 4C | FC | 10 | EC | )k..,..!..__L... |
| 2 | 563.1us | | | | | | BE | D4 | ED | 51 | 06 | 45 | 4D | 99 | 25 | 8E | 51 | 65 | 53 | 05 | 5C | 33 | ...Q.EM.%.QeS.\\3 |
| 3 | 1.0378ms | | | | | | EC | 3F | 54 | 16 | A7 | 22 | CD | CC | 8F | 60 | D4 | F3 | 4E | 4A | 60 | 3D | .?T..\"...`..NJ`= |
| 4 | 1.5124ms | | | | | | CB | EE | 2F | 68 | 16 | 75 | 93 | 6D | 35 | 33 | F4 | 0D | 4C | E6 | 05 | 39 | ../h.u.m53..L..9 |
| 5 | 1.9953ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 6 | 2.0256ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 7 | 2.0558ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 8 | 2.0861ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 9 | 2.1163ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 10 | 2.1466ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 11 | 2.1768ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 12 | 2.2071ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 13 | 2.2373ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 14 | 2.2676ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 15 | 2.2978ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 16 | 2.3281ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 17 | 2.3583ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 18 | 2.3886ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 19 | 2.4188ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |
| 20 | 2.4491ms | A | 0 | Write | | | | | | | | | | | | | | | | | | | |

## I$^2$S

I$^2$S is an interface standard for transferring digital audio data between ICs. It is a bus standard developed by Philips for transferring audio data between digital audio devices, and is often used for transferring PCM audio from a CD to a DAC in a CD player. The I$^2$S standard specifies the hardware interface specifications and the format of the digital audio data, and uses a sequential approach to transmit two groups (left and right channels) of data. It consists of three transmission lines, a clock line (SCK), a word selection line (WS), and a data line (SD). The data format is up to 32 bits.

### Settings



I2S Settings

Channel

Clock Channel (SCK): A0
Word Select Channel (WS): A1
Data Channel (SD): A2
Data Bits: 16 bit(s)

Config

□ Enable meet full scale
Mode: I2S Mode
Report: 8 Columns

TDM Settings

Channel Size: 16
Channel Length: 16
Word Length: 8
Channel Offset: 8
Latch: Rising

Sound reduction

□ Display the audio waveform
□ Save as WAV file
□ Playback
□ Align common sample rate

Color

R. Channel
L. Channel

Range

Decode Range
From: Buffer Head
To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Data Bits:** The number of bits of analyzed data ranges from 1 to 32 Bits; the default value is 16 Bits.

**Sound reduction:**

1. **Display the audio waveform:** User can draw the waveform of the sound in the Waveform area. Enabled when checked.
2. **Save as WAV file:** All Data can be saved as a sound file (.WAV) and saved in the working directory. Enabled when checked.
3. **Playback:** The default setting is off. This function collects all the data and plays it back after analyzing it. This is the fastest way to confirm that the sound is being transmitted correctly without having to go through the data one by one. Since the length of playback time depends on the depth of data that the Logic Analyzer can record, it is recommended that you increase the Logic Analyzer's data depth and reduce the number of channels used by the Logic Analyzer. Enabled when checked.
4. **Align common sample rate:** Automatically archives the sound waveforms with the sampling rate closest to the commonly used ones (44.1KHz, 48KHz ......). Enabled when checked.

**Config:**

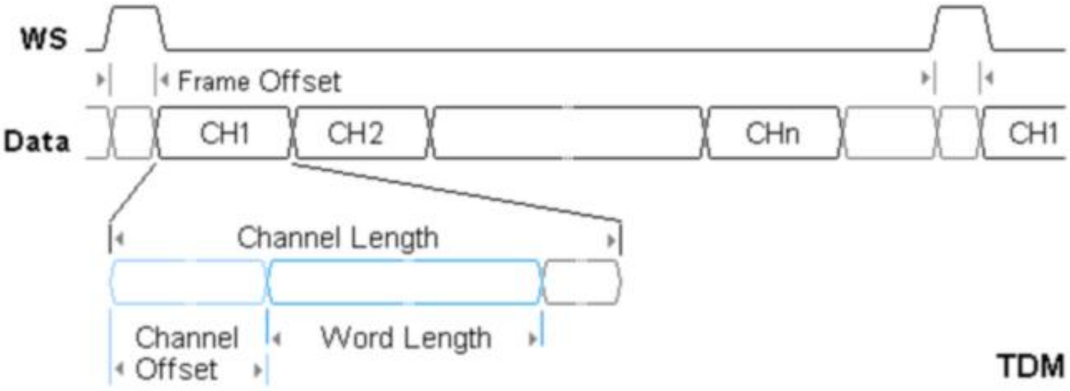
1. **Mode:** switch mode base on the needs.
   - i. I$^2$S Justified
   - ii. MSB Justified
   - iii. LSB Justified
   - iv. PCM

v. TDM

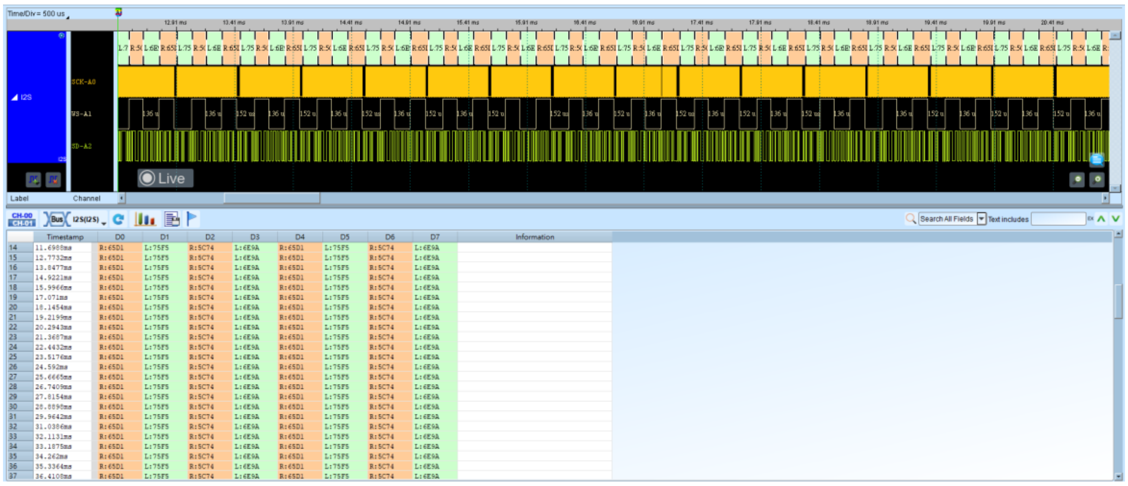
2. **Enable meet full scale:** Auto-completes missing bits in multiples of 8, e.g., 15 bits to 16 bits, 17 bits to 24 bits, enabled when checked.

3. **Report:** User can select the number of columns displayed in the report area.

**TDM Setting Parameter Definitions:**



**Result**



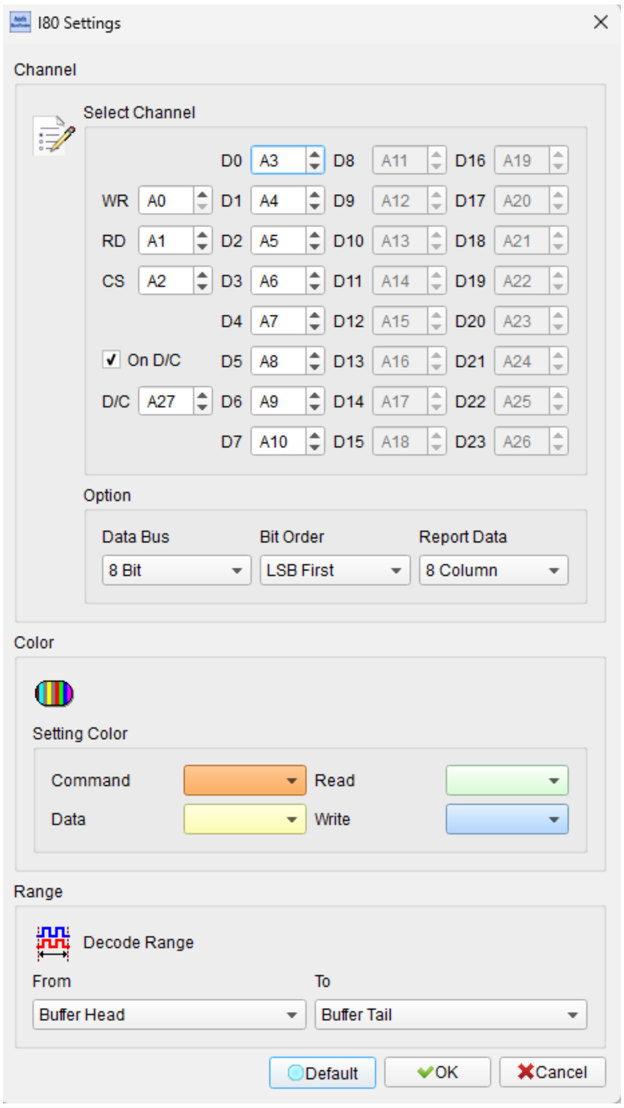
| | Timestamp | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | Information |
|---|---|---|---|---|---|---|---|---|---|---|
| 14 | 11.6988ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 15 | 12.7732ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 16 | 13.8477ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 17 | 14.9221ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 18 | 15.9966ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 19 | 17.071ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 20 | 18.1454ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 21 | 19.2199ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 22 | 20.2943ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 23 | 21.3687ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 24 | 22.4432ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 25 | 23.5176ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 26 | 24.592ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 27 | 25.6665ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 28 | 26.7409ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 29 | 27.8154ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 30 | 28.8898ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 31 | 29.9642ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 32 | 31.0386ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 33 | 32.1131ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 34 | 33.1875ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 35 | 34.262ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 36 | 35.3364ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |
| 37 | 36.4108ms | R:65D1 | L:75F5 | R:5C74 | L:6E9A | R:65D1 | L:75F5 | R:5C74 | L:6E9A | |

## I80

The Inter 8080-series interface is mainly used for LCM data transfer.

It is simply called I80 interface. To analyze the 8080-series, 3 or 4 Ctrl Buses (WR, RD, CS, and D/C) are required, and the Data Bus requires at least 4 bits depending on the user's definition, so at least 7 channels are required: WR, RD, CS, D0-D3, and 8 channels if there is a D/C Pin. The channel numbers of these signals can be adjusted. The channel numbers of these signals can be adjusted. The 8 bits Data bus requires 11 signals: WR, RD, CS, D0-D7, and so on... WR to CH0, and so on.

## Settings



I80 Settings

Channel

Select Channel

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | D0 | A3 | D8 | A11 | D16 | A19 |
| WR | A0 | D1 | A4 | D9 | A12 | D17 | A20 |
| RD | A1 | D2 | A5 | D10 | A13 | D18 | A21 |
| CS | A2 | D3 | A6 | D11 | A14 | D19 | A22 |
| | | D4 | A7 | D12 | A15 | D20 | A23 |
| ☑ On D/C | | D5 | A8 | D13 | A16 | D21 | A24 |
| D/C | A27 | D6 | A9 | D14 | A17 | D22 | A25 |
| | | D7 | A10 | D15 | A18 | D23 | A26 |

Option

| Data Bus | Bit Order | Report Data |
|---|---|---|
| 8 Bit | LSB First | 8 Column |

Color

Setting Color

| | | | |
|---|---|---|---|
| Command | 🟧 | Read | 🟩 |
| Data | 🟨 | Write | 🟦 |

Range

Decode Range

| From | To |
|---|---|
| Buffer Head | Buffer Tail |

**Select Channel:** Show the selected channels (WR, RD, CS, D0, D1, D2, D3, D4, D5, D6,…).

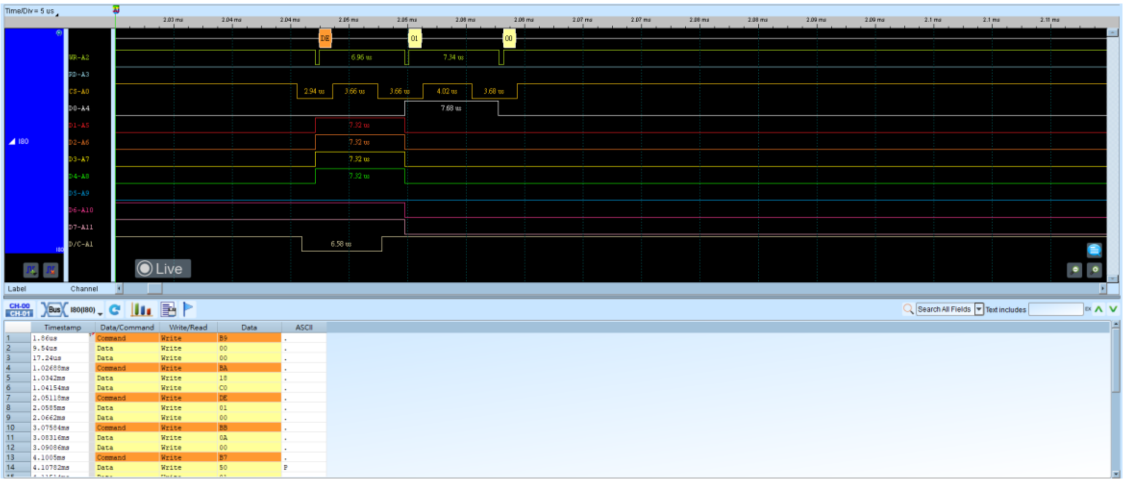
**On D/C:** Use the D/C pin as Command (Low) or Data (High).

**Data Bus:** Select 4 Bit, 8 Bit, 12 Bit, 16 Bit, 20 Bit, or 24 Bit.

**Bit Order:** Select LSB First or MSB First.

**Report Data:** Select 8 columns or 16 columns.

## Result



|  | Timestamp | Data/Command | Write/Read | Data | ASCII |
|---|---|---|---|---|---|
| 1 | 1.86us | Command | Write | B9 | . |
| 2 | 9.54us | Data | Write | 00 | . |
| 3 | 17.24us | Data | Write | 00 | . |
| 4 | 1.02686ms | Command | Write | BA | . |
| 5 | 1.0342ms | Data | Write | 18 | . |
| 6 | 1.04154ms | Data | Write | C0 | . |
| 7 | 2.05116ms | Command | Write | DE | . |
| 8 | 2.0585ms | Data | Write | 01 | . |
| 9 | 2.0662ms | Data | Write | 00 | . |
| 10 | 3.07584ms | Command | Write | BB | . |
| 11 | 3.08316ms | Data | Write | 0A | . |
| 12 | 3.09086ms | Data | Write | 00 | . |
| 13 | 4.1005ms | Command | Write | B7 | . |
| 14 | 4.10782ms | Data | Write | 50 | P |

## IDE

IDE (Integrated Device Electronics) is a standard interface for hard disks, solid-state drives, CD-ROMs, etc. IDE was first used by Western Digital in the U.S. for its hard disk sales business. The name IDE was first used by Western Digital in the United States for their hard disk sales business. The official specification name is ATA/ATAPI (Advanced Technology Attachment/AT Attachment Packet Interface). The ATA specification has continued to be added to due to the increased capacity of hard disks, the need for higher transfer speeds, and the constant evolution of storage devices. In 1998, the ATAPI specification was added to ATA-4, allowing ATA to connect to optical drives and other storage media. In 2003, the SATA (Serial ATA) specification was released, which retroactively renamed the original Parallel ATA to PATA (Parallel ATA) to differentiate it.

To analyze IDE, because it is a parallel transmission, it needs to use more channels, so we have to divide it into three types.

Normal channel (11 pin): Its signals are DASP-, DIOR-:HDMARDY-:HSTROBE, DIOW-:STOP, DMACK-, DMARQ, INTRQ, IORDY:DDMARDY-:DSTROBE, PDIAG-:CBLID-, RESET-, CSEL and IOCS16. -.

Register channel (5 pin): its signals are CS(0:1)- and DA(2:0).

Data channel (16 pin): its signal is DD(15:0).

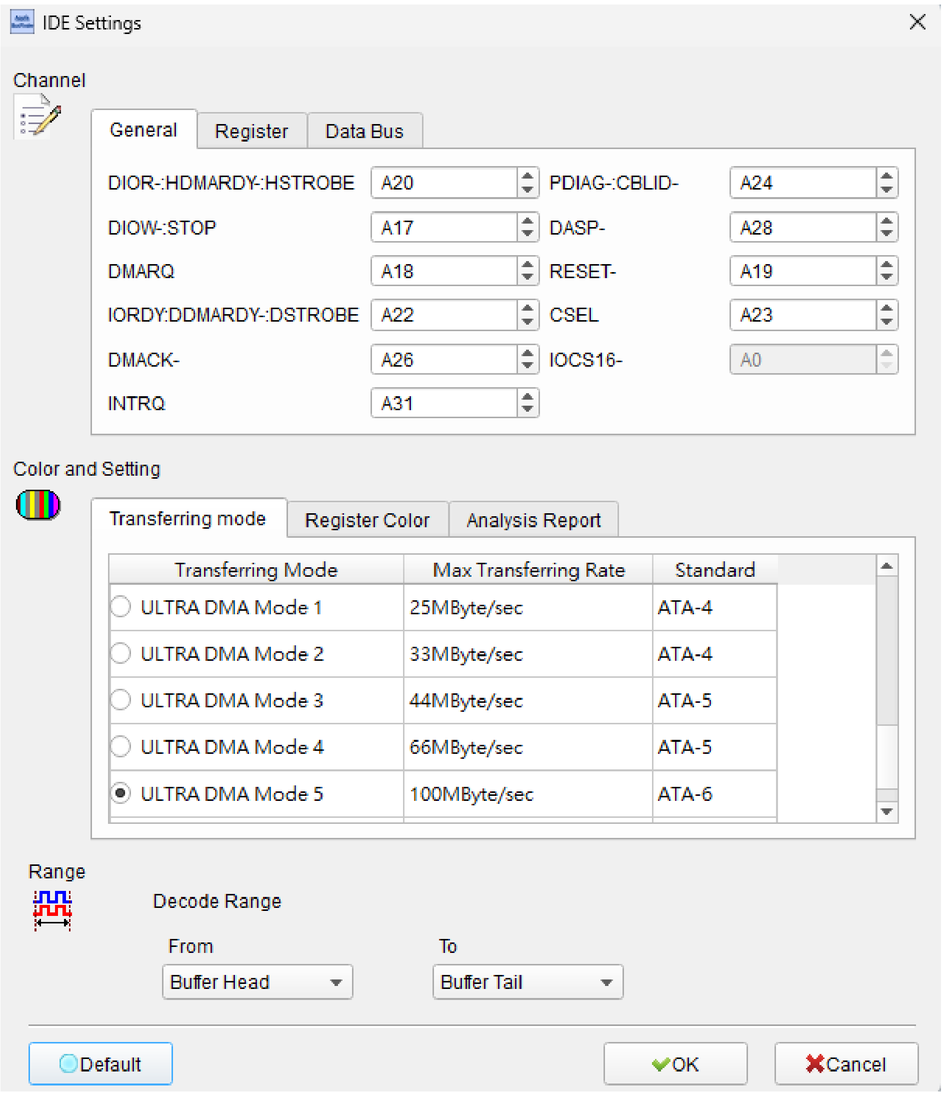
We recommend that IDE bus of the target system to be connected to the instrument as the following table:

| IDE Pin No. | IDE Pin name | IDE Pin Description | LA default Channel No. |
|---|---|---|---|
| Pin1 | Reset- | Hardware reset | Channel 0 |
| Pin2 | Ground | | |
| Pin3 | DD7 | Device data | Channel 1 |
| Pin4 | DD8 | Device data | Channel 2 |
| Pin5 | DD6 | Device data | Channel 3 |
| Pin6 | DD9 | Device data | Channel 4 |

| | | | |
|---|---|---|---|
| Pin7 | DD5 | Device data | Channel 5 |
| Pin8 | DD10 | Device data | Channel 6 |
| Pin9 | DD4 | Device data | Channel 7 |
| Pin10 | DD11 | Device data | Channel 8 |
| Pin11 | DD3 | Device data | Channel 9 |
| Pin12 | DD12 | Device data | Channel 10 |
| Pin13 | DD2 | Device data | Channel 11 |
| Pin14 | DD13 | Device data | Channel 12 |
| Pin15 | DD1 | Device data | Channel 13 |
| Pin16 | DD14 | Device data | Channel 14 |
| Pin17 | DD0 | Device data | Channel 15 |
| Pin18 | DD15 | Device data | Channel 16 |
| Pin19 | Ground | | |
| Pin20 | Key pin | | |
| Pin21 | DMARQ | DMA request | Channel 17 |
| Pin22 | Ground | | |
| Pin23 | DIOW-:STOP | Device I/O write: Stop Ultra DMA burst | Channel 18 |
| Pin24 | Ground | | |
| Pin25 | DIOR-:HDMARDY-:HSTROBE | Device I/O read: Ultra DMA ready: Ultra DMA data strobe | Channel 19 |
| Pin26 | Ground | | |
| Pin27 | IORDY:DDMARDY-:DSTROBE | I/O channel ready: Ultra DMA ready: Ultra DMA data strobe | Channel 20 |
| Pin28 | CSEL | Cable select | Channel 21 |
| Pin29 | DMACK- | DMA acknowledge | Channel 22 |
| Pin30 | Ground | | |
| Pin31 | INTRQ | Device interrupt | Channel 23 |
| Pin32 | Obsolete (see note) | Device 16-bit I/O in ATA-2 | Channel 24 |

| | | | |
|---|---|---|---|
| Pin33 | DA1 | Device address | Channel 25 |
| Pin34 | PDIAG-:CBLID- | Passed diagnostics: Cable assembly type identifier | Channel 26 |
| Pin35 | DA0 | Device address | Channel 27 |
| Pin36 | DA2 | Device address | Channel 28 |
| Pin37 | CS0- | Chip select | Channel 29 |
| Pin38 | CS1- | Chip select | Channel 30 |
| Pin39 | DASP- | Device active, device 1 present | Channel 31 |
| Pin40 | Ground | | |

### Settings



IDE Settings

Channel

| | | | |
|---|---|---|---|
| DIOR-:HDMARDY-:HSTROBE | A20 | PDIAG-:CBLID- | A24 |
| DIOW-:STOP | A17 | DASP- | A28 |
| DMARQ | A18 | RESET- | A19 |
| IORDY:DDMARDY-:DSTROBE | A22 | CSEL | A23 |
| DMACK- | A26 | IOCS16- | A0 |
| INTRQ | A31 | | |

Color and Setting

| Transferring Mode | Max Transferring Rate | Standard |
|---|---|---|
| ○ ULTRA DMA Mode 1 | 25MByte/sec | ATA-4 |
| ○ ULTRA DMA Mode 2 | 33MByte/sec | ATA-4 |
| ○ ULTRA DMA Mode 3 | 44MByte/sec | ATA-5 |
| ○ ULTRA DMA Mode 4 | 66MByte/sec | ATA-5 |
| ● ULTRA DMA Mode 5 | 100MByte/sec | ATA-6 |

Range

Decode Range

From: Buffer Head

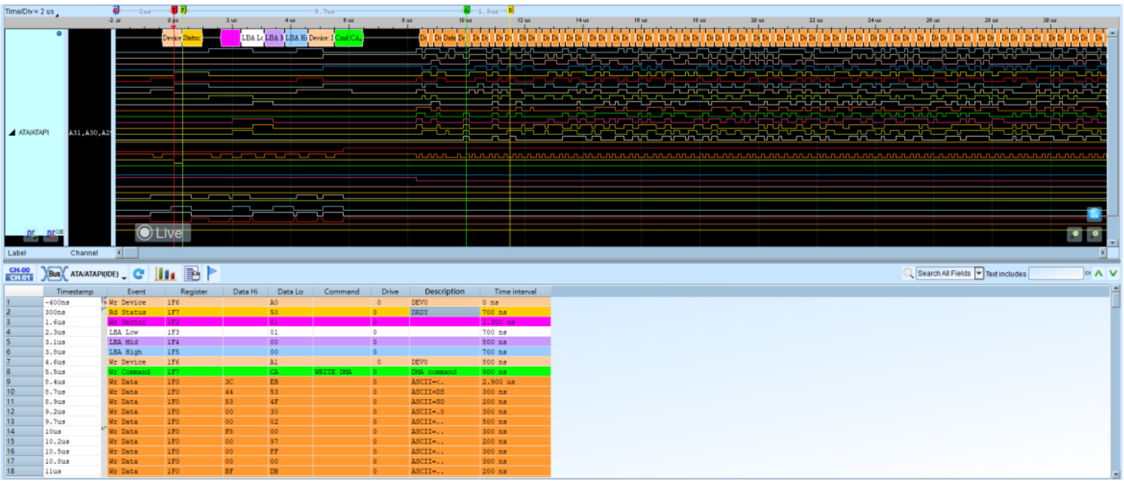
To: Buffer Tail

**Channel:** Set channel number for General, Register, and Data Bus.

**Transferring Mode:** User can specify which specification will be used by the device to be tested so that commands can be interpreted correctly during IDE analysis. If it is not specified correctly, the analysis can be performed.

**Analysis Report:** User can specify that only those registers are displayed in the Display Report window. For example, if the Data Register is unchecked, data about the data register will not appear in the Report window. In this way, the contents of the data registers are filtered out when viewing the analysis results.

## Result

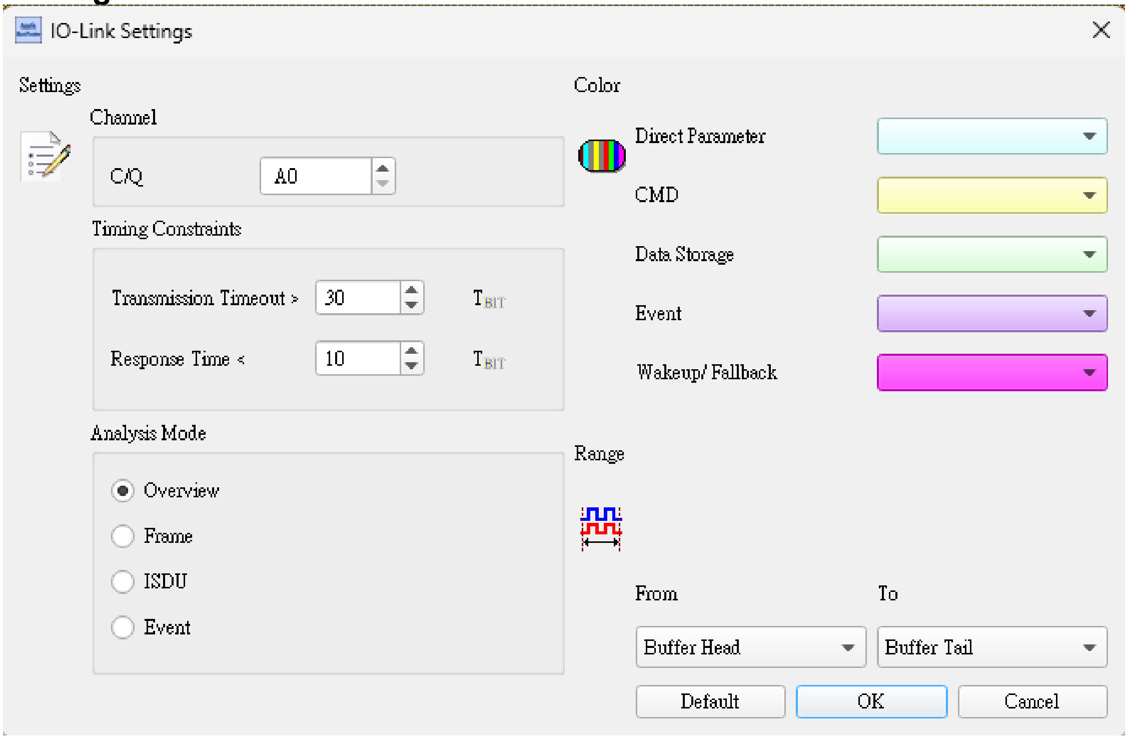


| | Timestamp | Event | Register | Data Hi | Data Lo | Command | Drive | Description | Time interval |
|---|---|---|---|---|---|---|---|---|---|
| 1 | -400ns | Wr Device | 1F6 | | A0 | | 0 | DEV0 | 0 ns |
| 2 | 300ns | Rd Status | 1F7 | | 50 | | 0 | DRDY | 700 ns |
| 3 | 1.6us | Wr Sector | 1F2 | | 01 | | 0 | | 1.300 us |
| 4 | 2.3us | LBA Low | 1F3 | | 01 | | 0 | | 700 ns |
| 5 | 3.1us | LBA Mid | 1F4 | | 00 | | 0 | | 800 ns |
| 6 | 3.8us | LBA High | 1F5 | | 00 | | 0 | | 700 ns |
| 7 | 4.6us | Wr Device | 1F6 | | A1 | | 0 | DEV0 | 800 ns |
| 8 | 5.5us | Wr Command | 1F7 | | CA | WRITE DMA | 0 | DMA command | 900 ns |
| 9 | 8.4us | Wr Data | 1F0 | 3C | EB | | 0 | ASCII=<. | 2.900 us |
| 10 | 8.7us | Wr Data | 1F0 | 44 | 53 | | 0 | ASCII=DS | 300 ns |
| 11 | 8.9us | Wr Data | 1F0 | 53 | 4F | | 0 | ASCII=SO | 200 ns |
| 12 | 9.2us | Wr Data | 1F0 | 00 | 30 | | 0 | ASCII=.0 | 300 ns |
| 13 | 9.7us | Wr Data | 1F0 | 00 | 02 | | 0 | ASCII=.. | 500 ns |
| 14 | 10us | Wr Data | 1F0 | F8 | 00 | | 0 | ASCII=.. | 300 ns |
| 15 | 10.2us | Wr Data | 1F0 | 00 | 97 | | 0 | ASCII=.. | 200 ns |
| 16 | 10.5us | Wr Data | 1F0 | 00 | FF | | 0 | ASCII=.. | 300 ns |
| 17 | 10.8us | Wr Data | 1F0 | 00 | 00 | | 0 | ASCII=.. | 300 ns |
| 18 | 11us | Wr Data | 1F0 | BF | DB | | 0 | ASCII=.. | 200 ns |

## IO-Link

IO-Link is a communication system that connects smart sensors and actuators to automation systems, in accordance with the Single-drop digital communication interface for small sensors and actuators (SDCI) in the IEC 61131-9 standard. This specification includes electrical connections and digital communication protocols through which smart sensors and actuators can interact with automation systems.

### Settings



IO-Link Settings

Settings

Channel

C/Q A0

Timing Constraints

Transmission Timeout > 30 $T_{BIT}$

Response Time < 10 $T_{BIT}$

Analysis Mode

● Overview
○ Frame
○ ISDU
○ Event

Color

Direct Parameter
CMD
Data Storage
Event
Wakeup/ Fallback

Range

From: Buffer Head
To: Buffer Tail

**Channel:** Set the C/Q signal terminal on the object under test to be connected to the channel number of the logic analyzer.

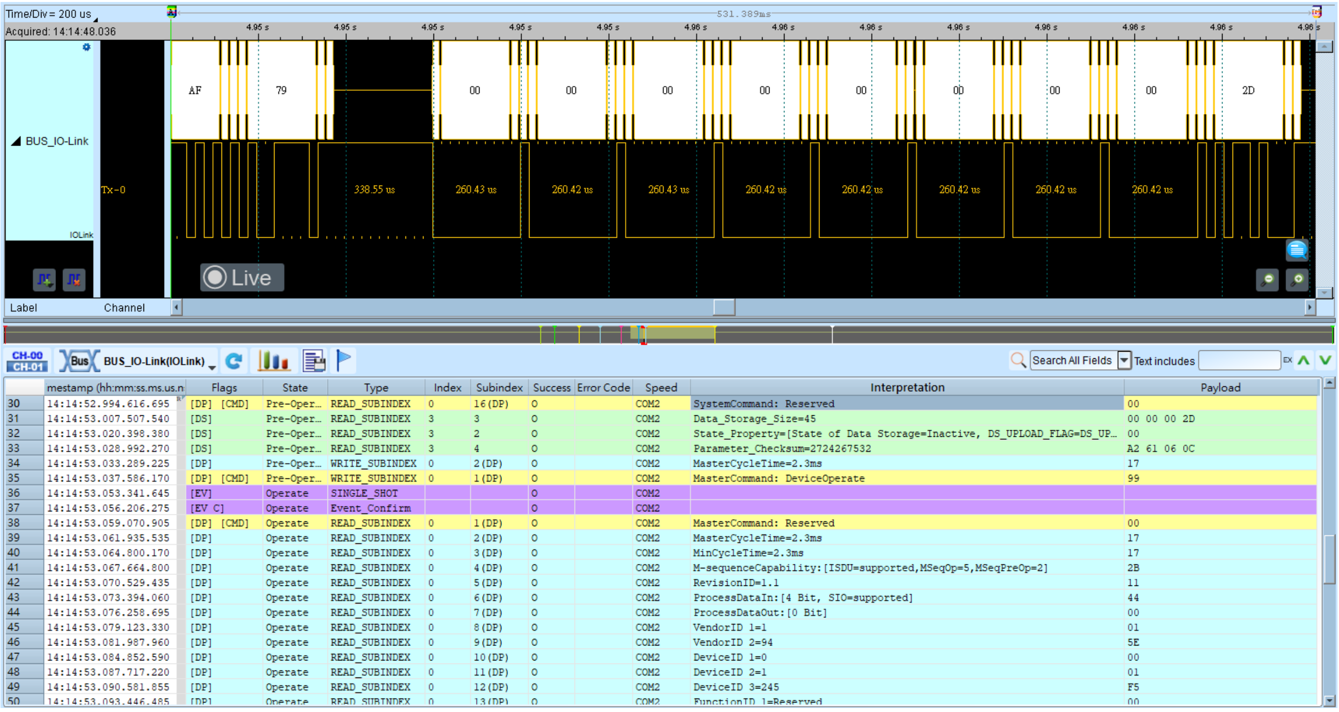
**Timing Constraints:**

- **Transmission Timeout:** The default is 30 TBIT. If IDLE is larger than this value, the next data will be decoded and analyzed by the Master.
- **Response Time:** The default is 10 TBIT. If IDLE is less than this value and greater than 3 TBIT, the next data will be decoded and analyzed by Device.

**Analysis Mode:** The report will be analyzed in the selected mode (Overview, Frame, ISDU, Event).

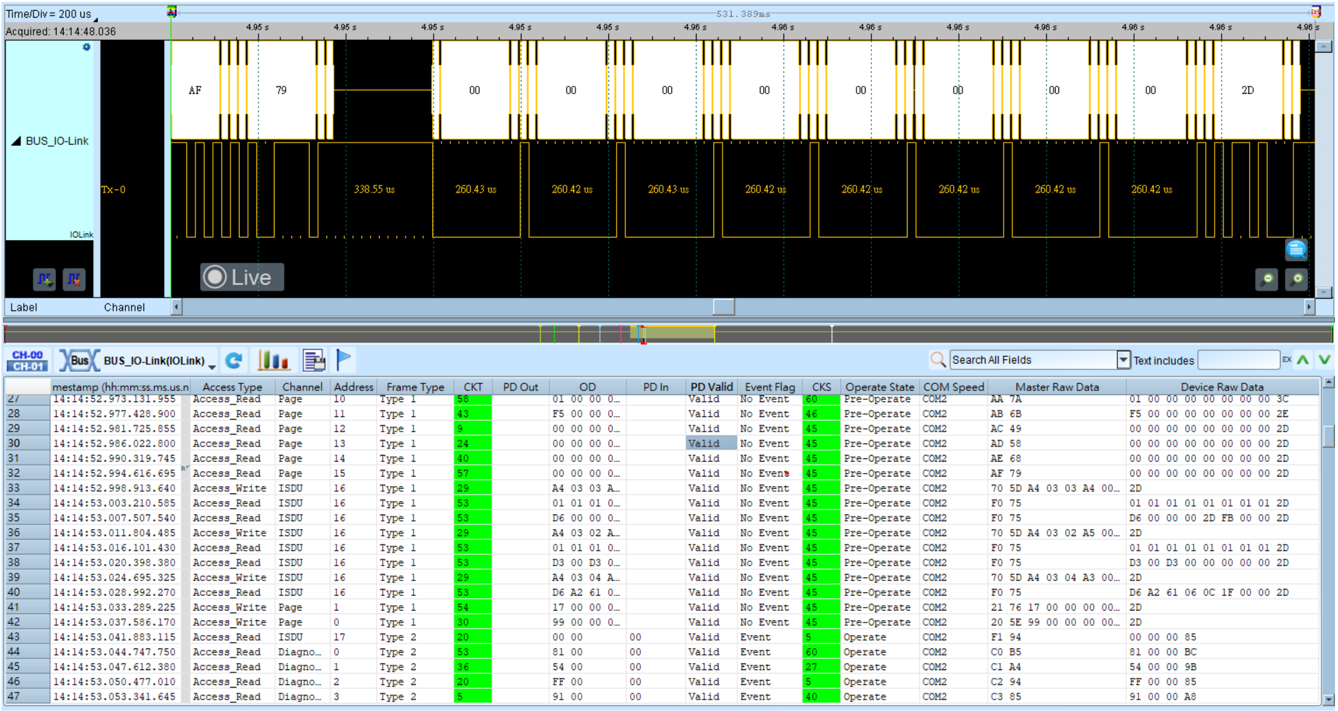
# Result

## Overview



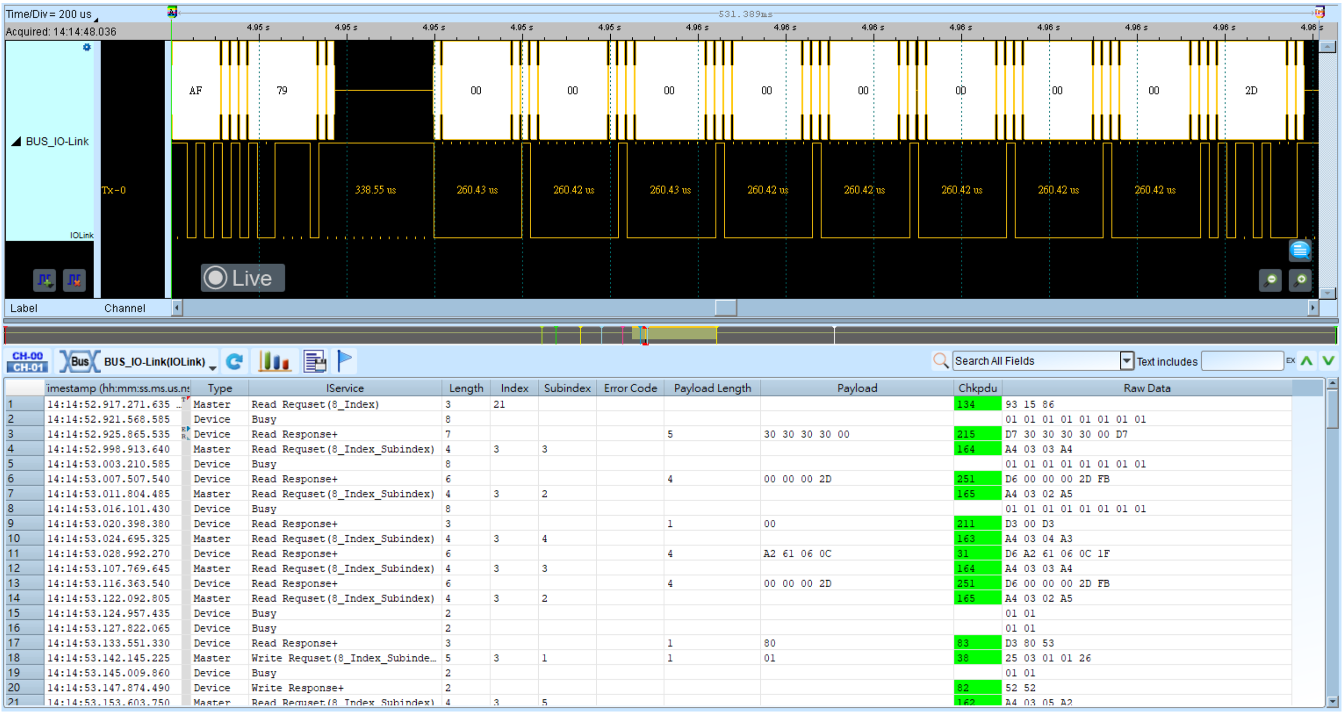
| | mestamp (hh:mm:ss.ms.us.n | Flags | State | Type | Index | Subindex | Success | Error Code | Speed | Interpretation | Payload |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 30 | 14:14:52.994.616.695 | [DP] [CMD] | Pre-Oper… | READ_SUBINDEX | 0 | 16 (DP) | O | | COM2 | SystemCommand: Reserved | 00 |
| 31 | 14:14:53.007.507.540 | [DS] | Pre-Oper… | READ_SUBINDEX | 3 | 3 | O | | COM2 | Data_Storage_Size=45 | 00 00 00 2D |
| 32 | 14:14:53.020.398.380 | [DS] | Pre-Oper… | READ_SUBINDEX | 3 | 2 | O | | COM2 | State_Property=[State of Data Storage=Inactive, DS_UPLOAD_FLAG=DS_UP… | 00 |
| 33 | 14:14:53.028.992.270 | [DS] | Pre-Oper… | READ_SUBINDEX | 3 | 4 | O | | COM2 | Parameter_Checksum=2724267532 | A2 61 06 0C |
| 34 | 14:14:53.033.289.225 | [DP] | Pre-Oper… | WRITE_SUBINDEX | 0 | 2 (DP) | O | | COM2 | MasterCycleTime=2.3ms | 17 |
| 35 | 14:14:53.037.586.170 | [DP] [CMD] | Pre-Oper… | WRITE_SUBINDEX | 0 | 1 (DP) | O | | COM2 | MasterCommand: DeviceOperate | 99 |
| 36 | 14:14:53.053.341.645 | [EV] | Operate | SINGLE_SHOT | | | O | | COM2 | | |
| 37 | 14:14:53.056.206.275 | [EV C] | Operate | Event_Confirm | | | O | | COM2 | | |
| 38 | 14:14:53.059.070.905 | [DP] [CMD] | Operate | READ_SUBINDEX | 0 | 1 (DP) | O | | COM2 | MasterCommand: Reserved | 00 |
| 39 | 14:14:53.061.935.535 | [DP] | Operate | READ_SUBINDEX | 0 | 2 (DP) | O | | COM2 | MasterCycleTime=2.3ms | 17 |
| 40 | 14:14:53.064.800.170 | [DP] | Operate | READ_SUBINDEX | 0 | 3 (DP) | O | | COM2 | MinCycleTime=2.3ms | 17 |
| 41 | 14:14:53.067.664.800 | [DP] | Operate | READ_SUBINDEX | 0 | 4 (DP) | O | | COM2 | M-sequenceCapability:[ISDU=supported,MSeqOp=5,MSeqPreOp=2] | 2B |
| 42 | 14:14:53.070.529.435 | [DP] | Operate | READ_SUBINDEX | 0 | 5 (DP) | O | | COM2 | RevisionID=1.1 | 11 |
| 43 | 14:14:53.073.394.060 | [DP] | Operate | READ_SUBINDEX | 0 | 6 (DP) | O | | COM2 | ProcessDataIn:[4 Bit, SIO=supported] | 44 |
| 44 | 14:14:53.076.258.695 | [DP] | Operate | READ_SUBINDEX | 0 | 7 (DP) | O | | COM2 | ProcessDataOut:[0 Bit] | 00 |
| 45 | 14:14:53.079.123.330 | [DP] | Operate | READ_SUBINDEX | 0 | 8 (DP) | O | | COM2 | VendorID 1=1 | 01 |
| 46 | 14:14:53.081.987.960 | [DP] | Operate | READ_SUBINDEX | 0 | 9 (DP) | O | | COM2 | VendorID 2=94 | 5E |
| 47 | 14:14:53.084.852.590 | [DP] | Operate | READ_SUBINDEX | 0 | 10 (DP) | O | | COM2 | DeviceID 1=0 | 00 |
| 48 | 14:14:53.087.717.220 | [DP] | Operate | READ_SUBINDEX | 0 | 11 (DP) | O | | COM2 | DeviceID 2=1 | 01 |
| 49 | 14:14:53.090.581.855 | [DP] | Operate | READ_SUBINDEX | 0 | 12 (DP) | O | | COM2 | DeviceID 3=245 | F5 |
| 50 | 14:14:53.093.446.485 | [DP] | Operate | READ_SUBINDEX | 0 | 13 (DP) | O | | COM2 | FunctionID 1=Reserved | 00 |

## Frame



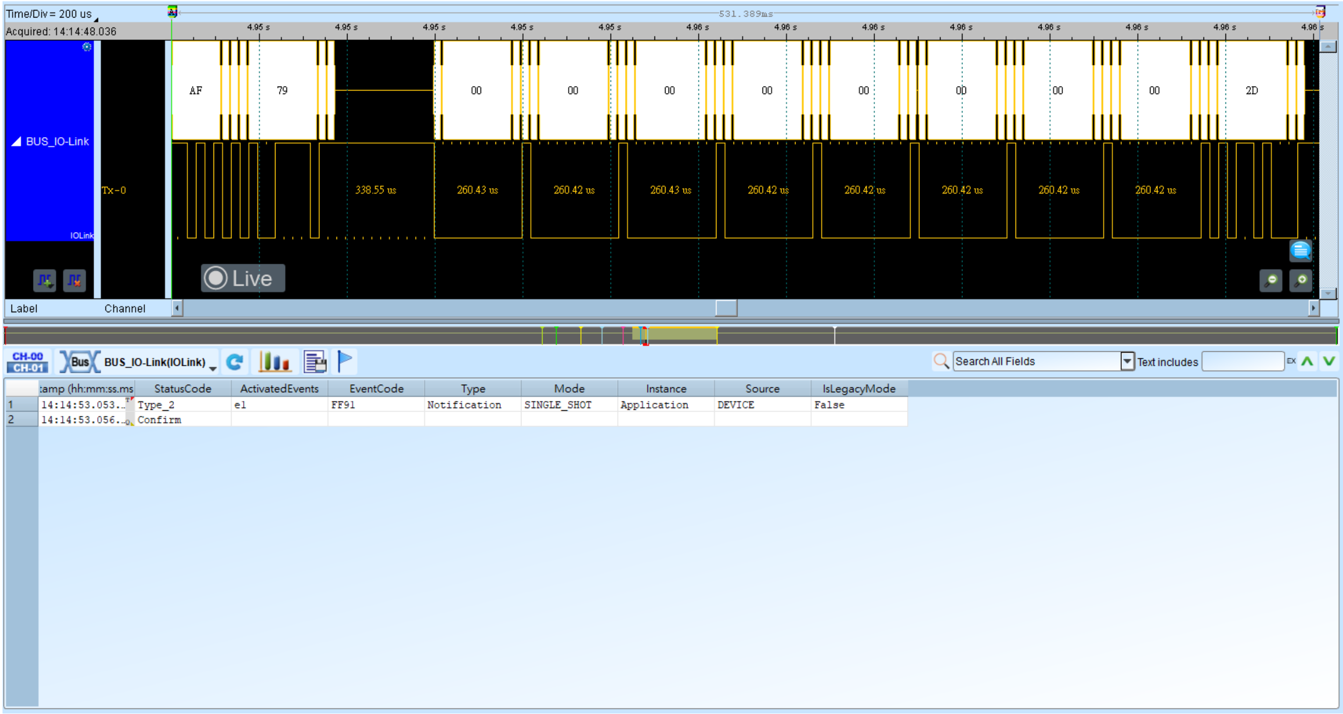
| | mestamp (hh:mm:ss.ms.us.n | Access Type | Channel | Address | Frame Type | CKT | PD Out | OD | PD In | PD Valid | Event Flag | CKS | Operate State | COM Speed | Master Raw Data | Device Raw Data |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 27 | 14:14:52.973.131.955 | Access_Read | Page | 10 | Type 1 | 58 | | 01 00 00 0… | | Valid | No Event | 60 | Pre-Operate | COM2 | AA 7A | 01 00 00 00 00 00 00 00 3C |
| 28 | 14:14:52.977.428.900 | Access_Read | Page | 11 | Type 1 | 43 | | F5 00 00 0… | | Valid | No Event | 46 | Pre-Operate | COM2 | AB 6B | F5 00 00 00 00 00 00 00 2E |
| 29 | 14:14:52.981.725.855 | Access_Read | Page | 12 | Type 1 | 9 | | 00 00 00 0… | | Valid | No Event | 45 | Pre-Operate | COM2 | AC 49 | 00 00 00 00 00 00 00 00 2D |
| 30 | 14:14:52.986.022.800 | Access_Read | Page | 13 | Type 1 | 24 | | 00 00 00 0… | | Valid | No Event | 45 | Pre-Operate | COM2 | AD 58 | 00 00 00 00 00 00 00 00 2D |
| 31 | 14:14:52.990.319.745 | Access_Read | Page | 14 | Type 1 | 40 | | 00 00 00 0… | | Valid | No Event | 45 | Pre-Operate | COM2 | AE 68 | 00 00 00 00 00 00 00 00 2D |
| 32 | 14:14:52.994.616.695 | Access_Read | Page | 15 | Type 1 | 57 | | 00 00 00 0… | | Valid | No Event | 45 | Pre-Operate | COM2 | AF 79 | 00 00 00 00 00 00 00 00 2D |
| 33 | 14:14:52.998.913.640 | Access_Write | ISDU | 16 | Type 1 | 29 | | A4 03 03 A… | | Valid | No Event | 45 | Pre-Operate | COM2 | 70 5D A4 03 03 A4 00… | 2D |
| 34 | 14:14:53.003.210.585 | Access_Read | ISDU | 16 | Type 1 | 53 | | 01 01 01 0… | | Valid | No Event | 45 | Pre-Operate | COM2 | F0 75 | 01 01 01 01 01 01 01 01 2D |
| 35 | 14:14:53.007.507.540 | Access_Read | ISDU | 16 | Type 1 | 53 | | D6 00 00 0… | | Valid | No Event | 45 | Pre-Operate | COM2 | F0 75 | D6 00 00 00 2D FB 00 00 2D |
| 36 | 14:14:53.011.804.485 | Access_Write | ISDU | 16 | Type 1 | 29 | | A4 03 02 A… | | Valid | No Event | 45 | Pre-Operate | COM2 | 70 5D A4 03 02 A5 00… | 2D |
| 37 | 14:14:53.016.101.430 | Access_Read | ISDU | 16 | Type 1 | 53 | | 01 01 01 0… | | Valid | No Event | 45 | Pre-Operate | COM2 | F0 75 | 01 01 01 01 01 01 01 01 2D |
| 38 | 14:14:53.020.398.380 | Access_Read | ISDU | 16 | Type 1 | 53 | | D3 00 D3 0… | | Valid | No Event | 45 | Pre-Operate | COM2 | F0 75 | D3 00 D3 00 00 00 00 00 2D |
| 39 | 14:14:53.024.695.325 | Access_Write | ISDU | 16 | Type 1 | 29 | | A4 03 04 A… | | Valid | No Event | 45 | Pre-Operate | COM2 | 70 5D A4 03 04 A3 00… | 2D |
| 40 | 14:14:53.028.992.270 | Access_Read | ISDU | 16 | Type 1 | 53 | | D6 A2 61 0… | | Valid | No Event | 45 | Pre-Operate | COM2 | F0 75 | D6 A2 61 06 0C 1F 00 00 2D |
| 41 | 14:14:53.033.289.225 | Access_Write | Page | 1 | Type 1 | 54 | | 17 00 00 0… | | Valid | No Event | 45 | Pre-Operate | COM2 | 21 76 17 00 00 00 00… | 2D |
| 42 | 14:14:53.037.586.170 | Access_Write | Page | 0 | Type 1 | 30 | | 99 00 00 0… | | Valid | No Event | 45 | Pre-Operate | COM2 | 20 5E 99 00 00 00 00… | 2D |
| 43 | 14:14:53.041.883.115 | Access_Read | ISDU | 17 | Type 2 | 20 | | 00 00 | 00 | Valid | Event | 5 | Operate | COM2 | F1 94 | 00 00 00 85 |
| 44 | 14:14:53.044.747.750 | Access_Read | Diagno… | 0 | Type 2 | 53 | | 81 00 | 00 | Valid | Event | 60 | Operate | COM2 | C0 B5 | 81 00 00 BC |
| 45 | 14:14:53.047.612.380 | Access_Read | Diagno… | 1 | Type 2 | 36 | | 54 00 | 00 | Valid | Event | 27 | Operate | COM2 | C1 A4 | 54 00 00 9B |
| 46 | 14:14:53.050.477.010 | Access_Read | Diagno… | 2 | Type 2 | 20 | | FF 00 | 00 | Valid | Event | 5 | Operate | COM2 | C2 94 | FF 00 00 85 |
| 47 | 14:14:53.053.341.645 | Access_Read | Diagno… | 3 | Type 2 | 5 | | 91 00 | 00 | Valid | Event | 40 | Operate | COM2 | C3 85 | 91 00 00 A8 |

## ISDU



| | imestamp (hh:mm:ss.ms.us.ns | Type | IService | Length | Index | Subindex | Error Code | Payload Length | Payload | Chkpdu | Raw Data |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 14:14:52.917.271.635 ... | Master | Read Requset(8_Index) | 3 | 21 | | | | | 134 | 93 15 86 |
| 2 | 14:14:52.921.568.585 | Device | Busy | 8 | | | | | | | 01 01 01 01 01 01 01 01 |
| 3 | 14:14:52.925.865.535 | Device | Read Response+ | 7 | | | | 5 | 30 30 30 30 00 | 215 | D7 30 30 30 30 00 D7 |
| 4 | 14:14:52.998.913.640 | Master | Read Requset(8_Index_Subindex) | 4 | 3 | 3 | | | | 164 | A4 03 03 A4 |
| 5 | 14:14:53.003.210.585 | Device | Busy | 8 | | | | | | | 01 01 01 01 01 01 01 01 |
| 6 | 14:14:53.007.507.540 | Device | Read Response+ | 6 | | | | 4 | 00 00 00 2D | 251 | D6 00 00 00 2D FB |
| 7 | 14:14:53.011.804.485 | Master | Read Requset(8_Index_Subindex) | 4 | 3 | 2 | | | | 165 | A4 03 02 A5 |
| 8 | 14:14:53.016.101.430 | Device | Busy | 8 | | | | | | | 01 01 01 01 01 01 01 01 |
| 9 | 14:14:53.020.398.380 | Device | Read Response+ | 3 | | | | 1 | 00 | 211 | D3 00 D3 |
| 10 | 14:14:53.024.695.325 | Master | Read Requset(8_Index_Subindex) | 4 | 3 | 4 | | | | 163 | A4 03 04 A3 |
| 11 | 14:14:53.028.992.270 | Device | Read Response+ | 6 | | | | 4 | A2 61 06 0C | 31 | D6 A2 61 06 0C 1F |
| 12 | 14:14:53.107.769.645 | Master | Read Requset(8_Index_Subindex) | 4 | 3 | 3 | | | | 164 | A4 03 03 A4 |
| 13 | 14:14:53.116.363.540 | Device | Read Response+ | 6 | | | | 4 | 00 00 00 2D | 251 | D6 00 00 00 2D FB |
| 14 | 14:14:53.122.092.805 | Master | Read Requset(8_Index_Subindex) | 4 | 3 | 2 | | | | 165 | A4 03 02 A5 |
| 15 | 14:14:53.124.957.435 | Device | Busy | 2 | | | | | | | 01 01 |
| 16 | 14:14:53.127.822.065 | Device | Busy | 2 | | | | | | | 01 01 |
| 17 | 14:14:53.133.551.330 | Device | Read Response+ | 3 | | | | 1 | 80 | 83 | D3 80 53 |
| 18 | 14:14:53.142.145.225 | Master | Write Requset(8_Index_Subinde... | 5 | 3 | 1 | | 1 | 01 | 38 | 25 03 01 01 26 |
| 19 | 14:14:53.145.009.860 | Device | Busy | 2 | | | | | | | 01 01 |
| 20 | 14:14:53.147.874.490 | Device | Write Response+ | 2 | | | | | | 82 | 52 52 |
| 21 | 14:14:53.153.603.750 | Master | Read Requset(8_Index_Subindex) | 4 | 3 | 5 | | | | 162 | A4 03 05 A2 |

## Event

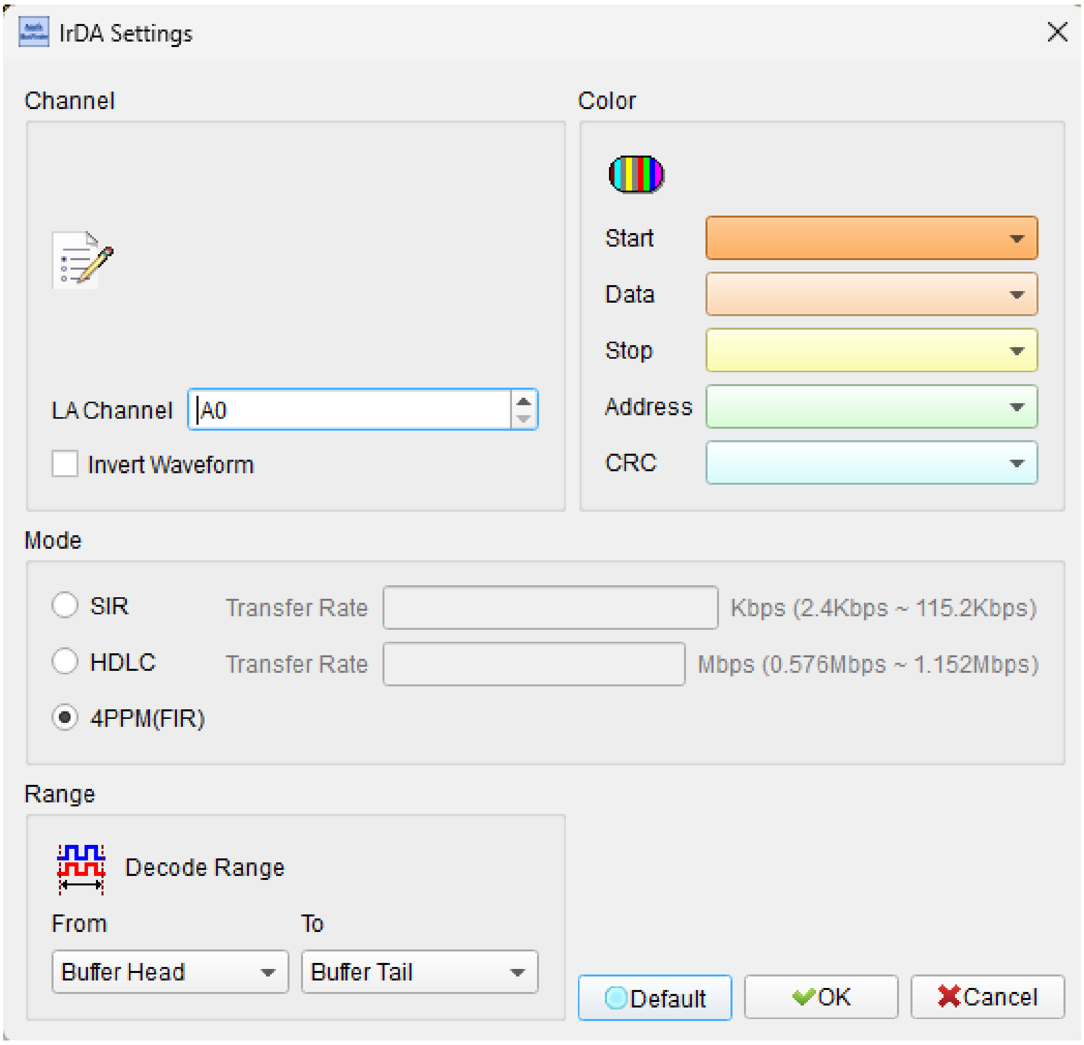


| | tamp (hh:mm:ss.ms | StatusCode | ActivatedEvents | EventCode | Type | Mode | Instance | Source | IsLegacyMode |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 14:14:53.053... | Type_2 | e1 | FF91 | Notification | SINGLE_SHOT | Application | DEVICE | False |
| 2 | 14:14:53.056... | Confirm | | | | | | | |

## IrDA

The Infrared Data Association (IrDA) was formed in 1993. The IrDA is point to point user model for a wide range of appliances and devices.

## Settings



IrDA Settings

Channel

LA Channel: A0

□ Invert Waveform

Color

Start
Data
Stop
Address
CRC

Mode

○ SIR Transfer Rate Kbps (2.4Kbps ~ 115.2Kbps)

○ HDLC Transfer Rate Mbps (0.576Mbps ~ 1.152Mbps)

● 4PPM(FIR)

Range

Decode Range

From: Buffer Head

To: Buffer Tail

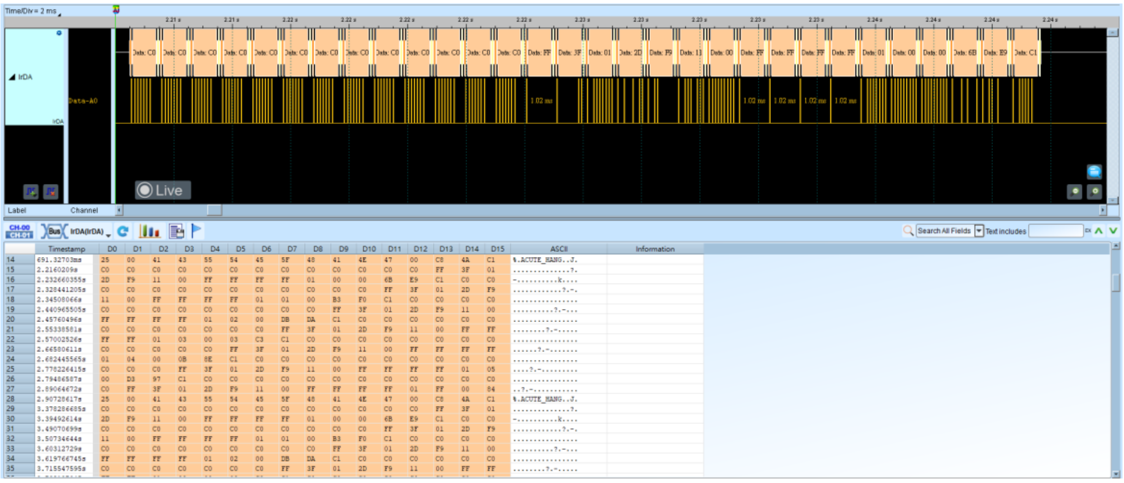
**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Invert Waveform:** Invert the waveform before analysis. Enabled when checked.

**Mode:**

1. **SIR:** After switching to SIR mode, enter the transfer rate to complete the setting.
2. **HDLC:** After switching to HDLC mode, enter the input rate to complete the setting.
3. **4PPM (FIR):**

## Result

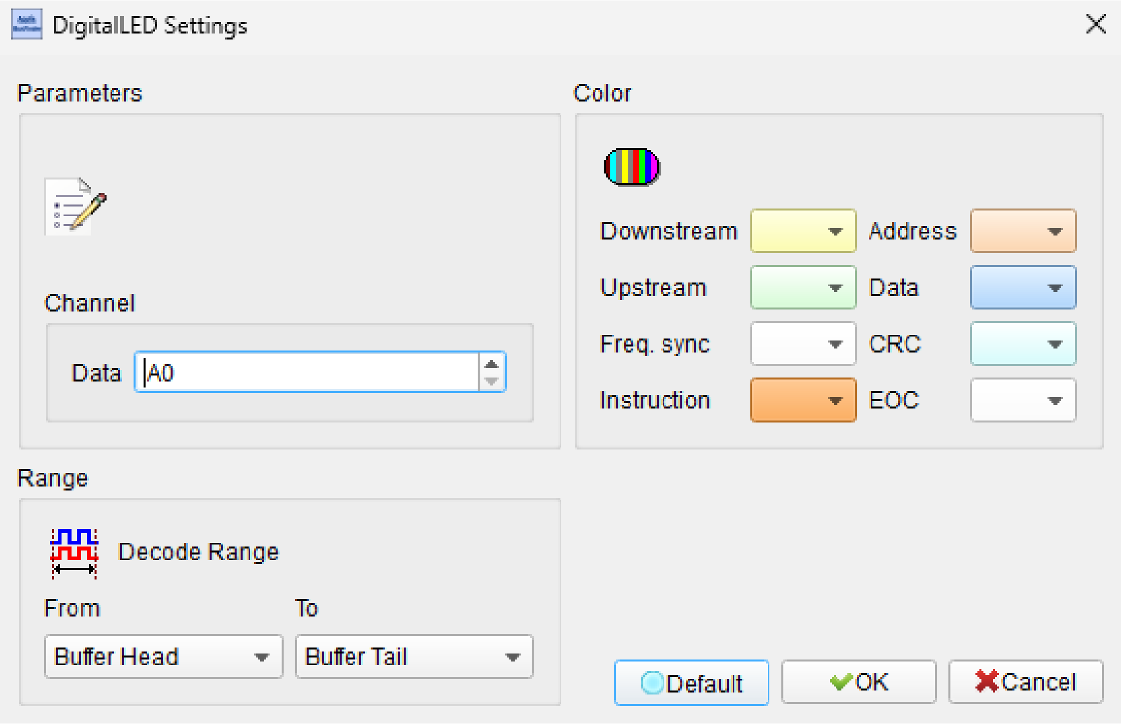


| | Timestamp | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ASCII | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 14 | 691.32703ms | 25 | 00 | 41 | 43 | 55 | 54 | 45 | 5F | 48 | 41 | 4E | 47 | 00 | C8 | 4A | C1 | %.ACUTE_HANG..J. | |
| 15 | 2.2160209s | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | FF | 3F | 01 | ..............?. | |
| 16 | 2.232660355s | 2D | F9 | 11 | 00 | FF | FF | FF | FF | 01 | 00 | 00 | 6B | E9 | C1 | C0 | C0 | -..........k.... | |
| 17 | 2.328441205s | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | FF | 3F | 01 | 2D | F9 | ............?.-. | |
| 18 | 2.34508066s | 11 | 00 | FF | FF | FF | FF | 01 | 01 | 00 | B3 | F0 | C1 | C0 | C0 | C0 | C0 | ................ | |
| 19 | 2.440965505s | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | FF | 3F | 01 | 2D | F9 | 11 | 00 | ..........?.-... | |
| 20 | 2.45760496s | FF | FF | FF | FF | 01 | 02 | 00 | DB | DA | C1 | C0 | C0 | C0 | C0 | C0 | C0 | ................ | |
| 21 | 2.55338581s | C0 | C0 | C0 | C0 | C0 | C0 | C0 | FF | 3F | 01 | 2D | F9 | 11 | 00 | FF | FF | ........?.-..... | |
| 22 | 2.57002526s | FF | FF | 01 | 03 | 00 | 03 | C3 | C1 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | ................ | |
| 23 | 2.66580611s | C0 | C0 | C0 | C0 | C0 | FF | 3F | 01 | 2D | F9 | 11 | 00 | FF | FF | FF | FF | ......?.-....... | |
| 24 | 2.682445565s | 01 | 04 | 00 | 0B | 8E | C1 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | ................ | |
| 25 | 2.778226415s | C0 | C0 | C0 | FF | 3F | 01 | 2D | F9 | 11 | 00 | FF | FF | FF | FF | 01 | 05 | ....?.-......... | |
| 26 | 2.79486587s | 00 | D3 | 97 | C1 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | ................ | |
| 27 | 2.89064672s | C0 | FF | 3F | 01 | 2D | F9 | 11 | 00 | FF | FF | FF | FF | 01 | FF | 00 | 84 | ..?.-........... | |
| 28 | 2.90728617s | 25 | 00 | 41 | 43 | 55 | 54 | 45 | 5F | 48 | 41 | 4E | 47 | 00 | C8 | 4A | C1 | %.ACUTE_HANG..J. | |
| 29 | 3.378286685s | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | FF | 3F | 01 | ..............?. | |
| 30 | 3.39492614s | 2D | F9 | 11 | 00 | FF | FF | FF | FF | 01 | 00 | 00 | 6B | E9 | C1 | C0 | C0 | -..........k.... | |
| 31 | 3.49070699s | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | FF | 3F | 01 | 2D | F9 | ............?.-. | |
| 32 | 3.50734644s | 11 | 00 | FF | FF | FF | FF | 01 | 01 | 00 | B3 | F0 | C1 | C0 | C0 | C0 | C0 | ................ | |
| 33 | 3.60312729s | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | C0 | FF | 3F | 01 | 2D | F9 | 11 | 00 | ..........?.-... | |
| 34 | 3.619766745s | FF | FF | FF | FF | 01 | 02 | 00 | DB | DA | C1 | C0 | C0 | C0 | C0 | C0 | C0 | ................ | |
| 35 | 3.715547595s | C0 | C0 | C0 | C0 | C0 | C0 | C0 | FF | 3F | 01 | 2D | F9 | 11 | 00 | FF | FF | ........?.-..... | |

## ISELED

ISELED (Integrated Smart Embedded LED) is a new smart embedded LED technology designed to provide more efficient and intelligent control of LED lighting systems. It combines the luminous performance of LEDs with integrated circuit (IC) technology to enable more precise and flexible control for a wide range of application scenarios, especially in automotive, architectural and consumer electronics.

### Settings



DigitalLED Settings

Parameters

Channel

Data: A0

Color

Range

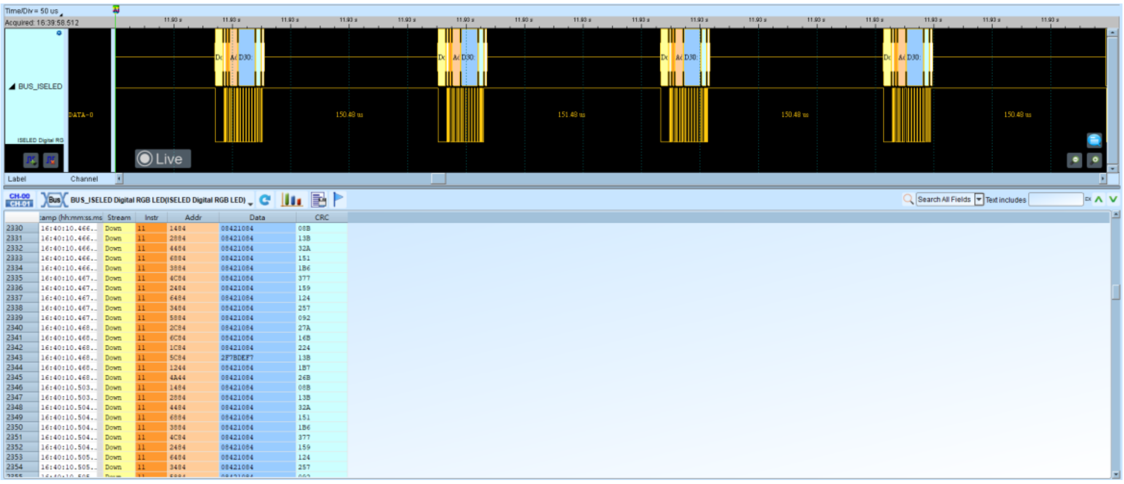
Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

## Result

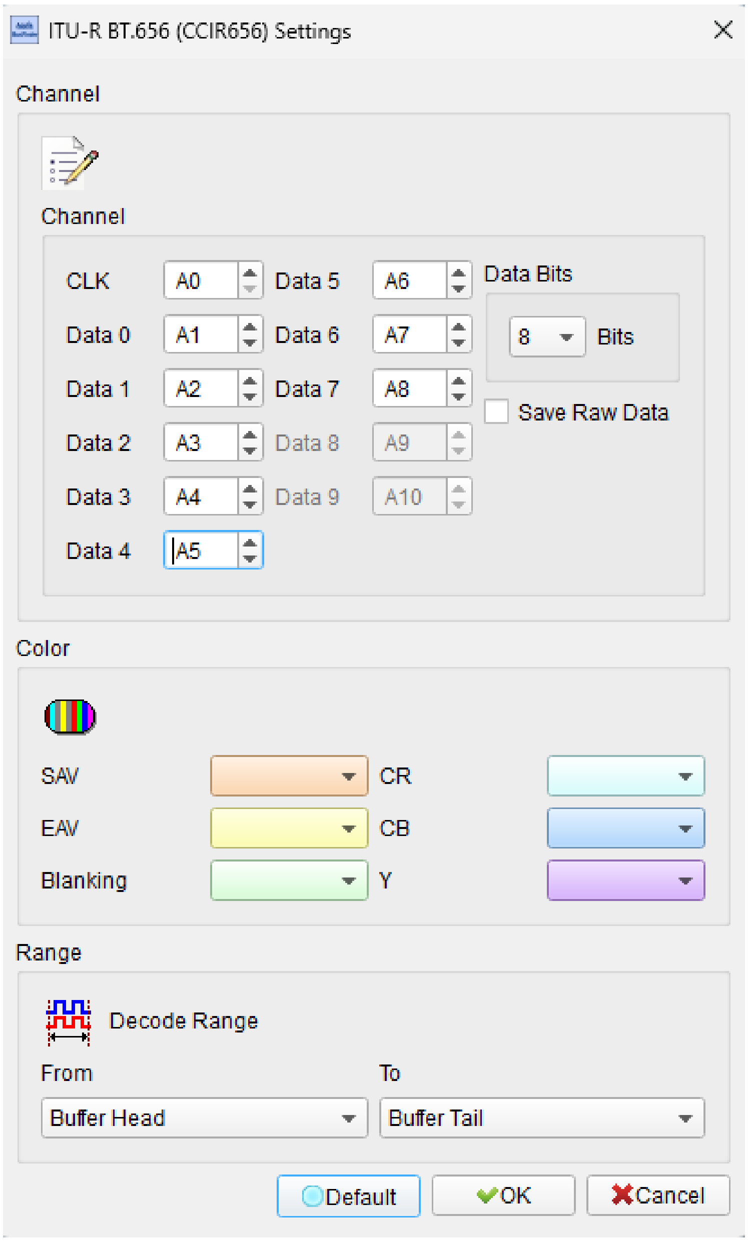


| | :amp (hh:mm:ss.ms | Stream | Instr | Addr | Data | CRC |
|---|---|---|---|---|---|---|
| 2330 | 16:40:10.466.. | Down | 11 | 1484 | 08421084 | 08B |
| 2331 | 16:40:10.466.. | Down | 11 | 2884 | 08421084 | 13B |
| 2332 | 16:40:10.466.. | Down | 11 | 4484 | 08421084 | 32A |
| 2333 | 16:40:10.466.. | Down | 11 | 6884 | 08421084 | 151 |
| 2334 | 16:40:10.466.. | Down | 11 | 3884 | 08421084 | 1B6 |
| 2335 | 16:40:10.467.. | Down | 11 | 4C84 | 08421084 | 377 |
| 2336 | 16:40:10.467.. | Down | 11 | 2484 | 08421084 | 159 |
| 2337 | 16:40:10.467.. | Down | 11 | 6484 | 08421084 | 124 |
| 2338 | 16:40:10.467.. | Down | 11 | 3484 | 08421084 | 257 |
| 2339 | 16:40:10.467.. | Down | 11 | 5884 | 08421084 | 092 |
| 2340 | 16:40:10.468.. | Down | 11 | 2C84 | 08421084 | 27A |
| 2341 | 16:40:10.468.. | Down | 11 | 6C84 | 08421084 | 16B |
| 2342 | 16:40:10.468.. | Down | 11 | 1C84 | 08421084 | 224 |
| 2343 | 16:40:10.468.. | Down | 11 | 5C84 | 2F7BDEF7 | 13B |
| 2344 | 16:40:10.468.. | Down | 11 | 1244 | 08421084 | 1B7 |
| 2345 | 16:40:10.468.. | Down | 11 | 4A44 | 08421084 | 26B |
| 2346 | 16:40:10.503.. | Down | 11 | 1484 | 08421084 | 08B |
| 2347 | 16:40:10.503.. | Down | 11 | 2884 | 08421084 | 13B |
| 2348 | 16:40:10.504.. | Down | 11 | 4484 | 08421084 | 32A |
| 2349 | 16:40:10.504.. | Down | 11 | 6884 | 08421084 | 151 |
| 2350 | 16:40:10.504.. | Down | 11 | 3884 | 08421084 | 1B6 |
| 2351 | 16:40:10.504.. | Down | 11 | 4C84 | 08421084 | 377 |
| 2352 | 16:40:10.504.. | Down | 11 | 2484 | 08421084 | 159 |
| 2353 | 16:40:10.505.. | Down | 11 | 6484 | 08421084 | 124 |
| 2354 | 16:40:10.505.. | Down | 11 | 3484 | 08421084 | 257 |
| 2355 | 16:40:10.505.. | Down | 11 | 5884 | 08421084 | 092 |

## ITU656 (CCIR656)

ITU656 describes a simple digital video protocol for streaming uncompressed PAL or NTSC Standard Definition TV (525 or 625 lines) signals. The protocol builds upon the 4:2:2 digital video encoding parameters defined in ITU-R Recommendation BT.601, which provides interlaced video data, streaming each field separately, and uses the YCbCr color space and a 13.5 MHz sampling frequency for pixels.

### Settings



ITU-R BT.656 (CCIR656) Settings

Channel

Channel

| | | | |
|---|---|---|---|
| CLK | A0 | Data 5 | A6 |
| Data 0 | A1 | Data 6 | A7 |
| Data 1 | A2 | Data 7 | A8 |
| Data 2 | A3 | Data 8 | A9 |
| Data 3 | A4 | Data 9 | A10 |
| Data 4 | A5 | | |

Data Bits: 8 Bits

□ Save Raw Data

Color

SAV, EAV, Blanking, CR, CB, Y

Range

Decode Range

From: Buffer Head

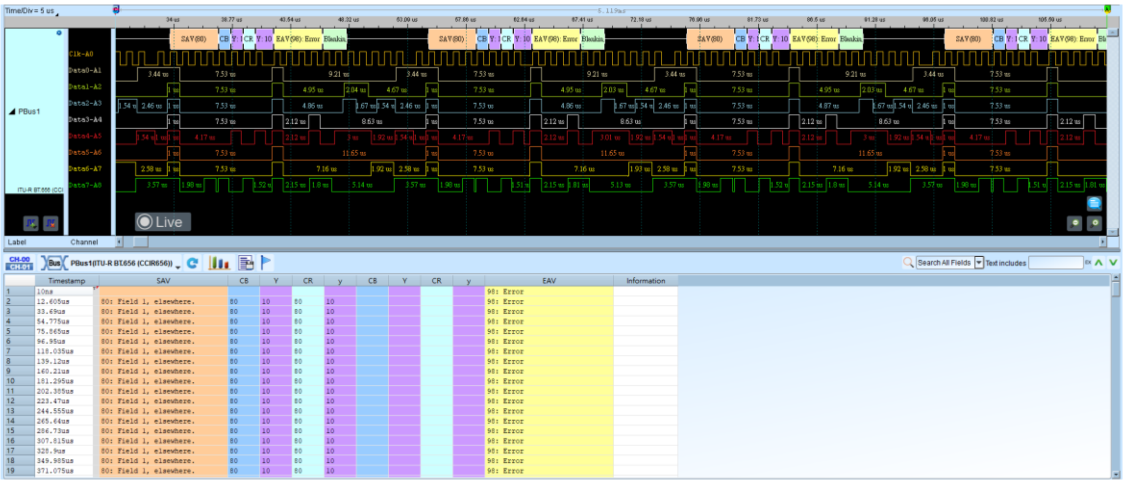
To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Data Bits:** Show the number of data bits.

**Save Raw Data:** Save the result as .bin file. Enabled when checked.

**Result**



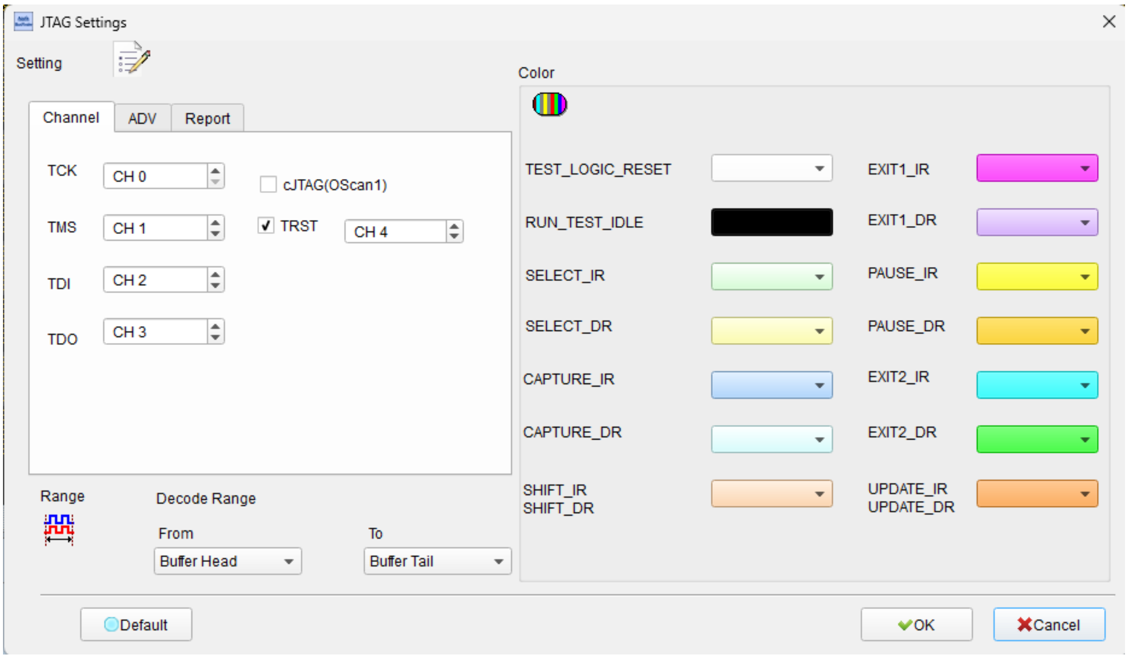
PBus1(ITU-R BT.656 (CCIR656))

| | Timestamp | SAV | CB | Y | CR | y | CB | Y | CR | y | EAV | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 10ns | | | | | | | | | | 98: Error | |
| 2 | 12.605us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 3 | 33.69us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 4 | 54.775us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 5 | 75.865us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 6 | 96.95us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 7 | 118.035us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 8 | 139.12us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 9 | 160.21us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 10 | 181.295us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 11 | 202.385us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 12 | 223.47us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 13 | 244.555us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 14 | 265.64us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 15 | 286.73us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 16 | 307.815us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 17 | 328.9us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 18 | 349.985us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |
| 19 | 371.075us | 80: Field 1, elsewhere. | 80 | 10 | 80 | 10 | | | | | 98: Error | |

## JTAG

JTAG (Joint Test Action Group) is an international standard test protocol (IEEE 1149.1), mainly used for on-chip internal testing, and nowadays most advanced components support JTAG protocol, such as DSPs, FPGAs, etc. The standard JTAG consists of five signal interfaces: TCK, TMS, TDI, TDO and TRST, four of which are input signal interfaces and the other one is output signal interface. The standard JTAG consists of five signal interfaces: TCK, TMS, TDI, TDO and TRST, four of which are input signal interfaces and the other one is an output signal interface. The basic principle is to define a TAP (Test Access Port) inside the device and test the internal nodes with a dedicated JTAG test tool.

## Settings



JTAG Settings

Setting

TCK: CH 0
TMS: CH 1
TDI: CH 2
TDO: CH 3
□ cJTAG(OScan1)
☑ TRST: CH 4

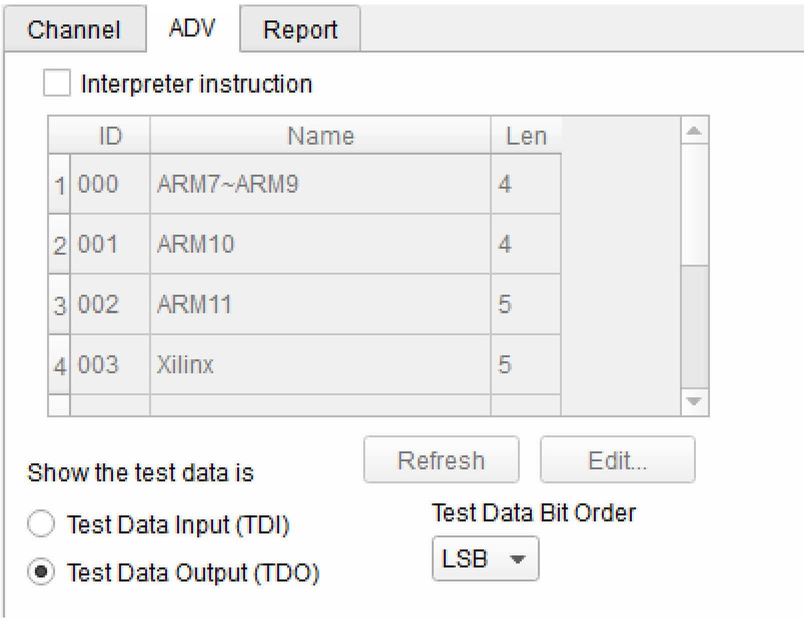
Range
Decode Range
From: Buffer Head
To: Buffer Tail

Color

| | | | |
|---|---|---|---|
| TEST_LOGIC_RESET | ⬜ | EXIT1_IR | 🟪 |
| RUN_TEST_IDLE | ⬛ | EXIT1_DR | 🟪 |
| SELECT_IR | 🟩 | PAUSE_IR | 🟨 |
| SELECT_DR | 🟨 | PAUSE_DR | 🟨 |
| CAPTURE_IR | 🟦 | EXIT2_IR | 🟦 |
| CAPTURE_DR | ⬜ | EXIT2_DR | 🟩 |
| SHIFT_IR<br>SHIFT_DR | 🟧 | UPDATE_IR<br>UPDATE_DR | 🟧 |

**Channel:** Specify the channel number of the Logic Analyzer to be connected with the object to be tested, the TRST pin can be decided by the user, if you are going to use the function of interpreting commands, then the system will decide whether to use the TRST pin or not according to the data of the commands that you have selected, and the user can decide whether to turn on the cJTAG or not, if you turn on the cJTAG option, the TDI/TDO channels are displayed in If cJTAG is enabled, the TDI/TDO channels will be grayed out, and the TCK/TMS channels will be treated as TCKC/TMSC channels in cJTAG OScan1 mode.

**Advanced:**



□ Interpreter instruction

|  | ID | Name | Len |
|---|---|---|---|
| 1 | 000 | ARM7~ARM9 | 4 |
| 2 | 001 | ARM10 | 4 |
| 3 | 002 | ARM11 | 5 |
| 4 | 003 | Xilinx | 5 |

Show the test data is

○ Test Data Input (TDI)

● Test Data Output (TDO)

Test Data Bit Order: LSB

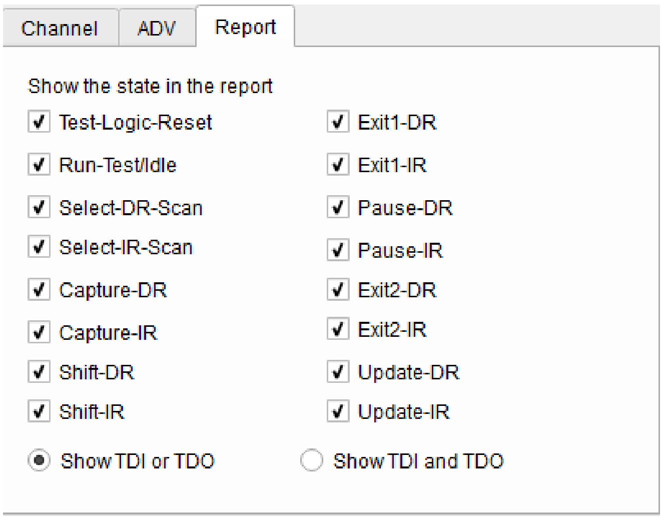
I. **Show the test data is:** User can select the state of TAP state as Shift-IR, Shift-DR. TDI or TDO data will be displayed in hexadecimal.

II. **Test Data Bit Order:** The length of data may vary during data transmission by JTAG. Therefore, the user can specify whether the data is LSB First or MSB First when interpreting TDI/TDO.

III. **Interpreter Instruction:** If you open the Explain Command function, you will see a list of commands, and the JTAG protocol analyzer will display the commands in the Instruction register during Update-IR. Users can select “Edit...” function to add and modify the instruction list file (JtagInst.txt) by themselves using the editor. After finishing the modification, click “Refresh” again to update the instruction list.

IV. **Acute Jtag Instruction table(JtagInst.txt):** This file is provided by Jtag DLL, users can re-edit this file according to their own needs. We also supports BSDL format, user can directly add the BSDL file, user can save the time of editing instruction data, please see the last appendix of this unit Acute Jtag Instruction table syntax description.

**Report:** To enable report filtering, simply check the items to be displayed in the report window.



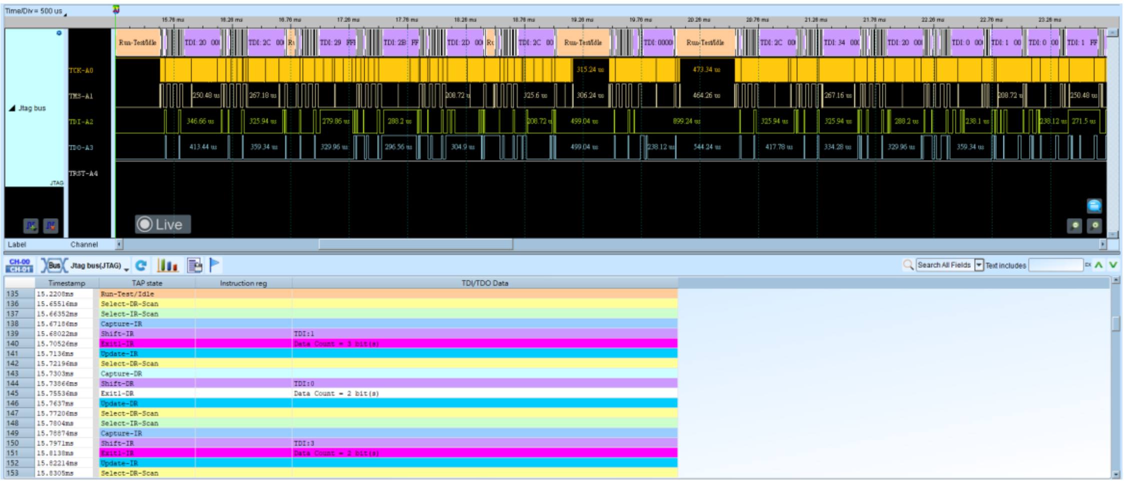
Show the state in the report

- ☑ Test-Logic-Reset
- ☑ Run-Test/Idle
- ☑ Select-DR-Scan
- ☑ Select-IR-Scan
- ☑ Capture-DR
- ☑ Capture-IR
- ☑ Shift-DR
- ☑ Shift-IR
- ☑ Exit1-DR
- ☑ Exit1-IR
- ☑ Pause-DR
- ☑ Pause-IR
- ☑ Exit2-DR
- ☑ Exit2-IR
- ☑ Update-DR
- ☑ Update-IR

● Show TDI or TDO ○ Show TDI and TDO

**I. Show TDI or/and TDO: If choose "Show TDI and TDO", the report area will show the TDI and TDO at the same time.**

**Result**



| | Timestamp | TAP state | Instruction reg | TDI/TDO Data |
|---|---|---|---|---|
| 135 | 15.2208ms | Run-Test/Idle | | |
| 136 | 15.65516ms | Select-DR-Scan | | |
| 137 | 15.66352ms | Select-IR-Scan | | |
| 138 | 15.67186ms | Capture-IR | | |
| 139 | 15.68022ms | Shift-IR | | TDI:1 |
| 140 | 15.70526ms | Exit1-IR | | Data Count = 3 bit(s) |
| 141 | 15.7136ms | Update-IR | | |
| 142 | 15.72196ms | Select-DR-Scan | | |
| 143 | 15.7303ms | Capture-DR | | |
| 144 | 15.73866ms | Shift-DR | | TDI:0 |
| 145 | 15.75536ms | Exit1-DR | | Data Count = 2 bit(s) |
| 146 | 15.7637ms | Update-DR | | |
| 147 | 15.77206ms | Select-DR-Scan | | |
| 148 | 15.7804ms | Select-IR-Scan | | |
| 149 | 15.78874ms | Capture-IR | | |
| 150 | 15.7971ms | Shift-IR | | TDI:3 |
| 151 | 15.8138ms | Exit1-IR | | Data Count = 2 bit(s) |
| 152 | 15.82214ms | Update-IR | | |
| 153 | 15.8305ms | Select-DR-Scan | | |

**Acute JTAG Instruction table Syntax Description (JtagInst.txt):**

The numbers used in this file are hexadecimal.

**##:** is comment.

**#ID:** Command list number; the range is 00 - FF and , MUST be entered in order or will be seen as the end of commands.

**#NAME:** Command Name, 32 bytes most will be shown in the command list.

**#LENGTH:** Command length, unit in bits.

**#CAPTURE:** Command Capture Code, is stored in Instruction Register..

**#INST:** Command List, listed by Command Code and Command Name or will be seen as the end of commands..

**#TRST:** Enter 1 if TREST is needed or enter 0 or nothing if TREST is not needed.

**#BSDL:** Load the BSDL file. Use the BSDL file as step 1-6.

**Example:**#ID:00

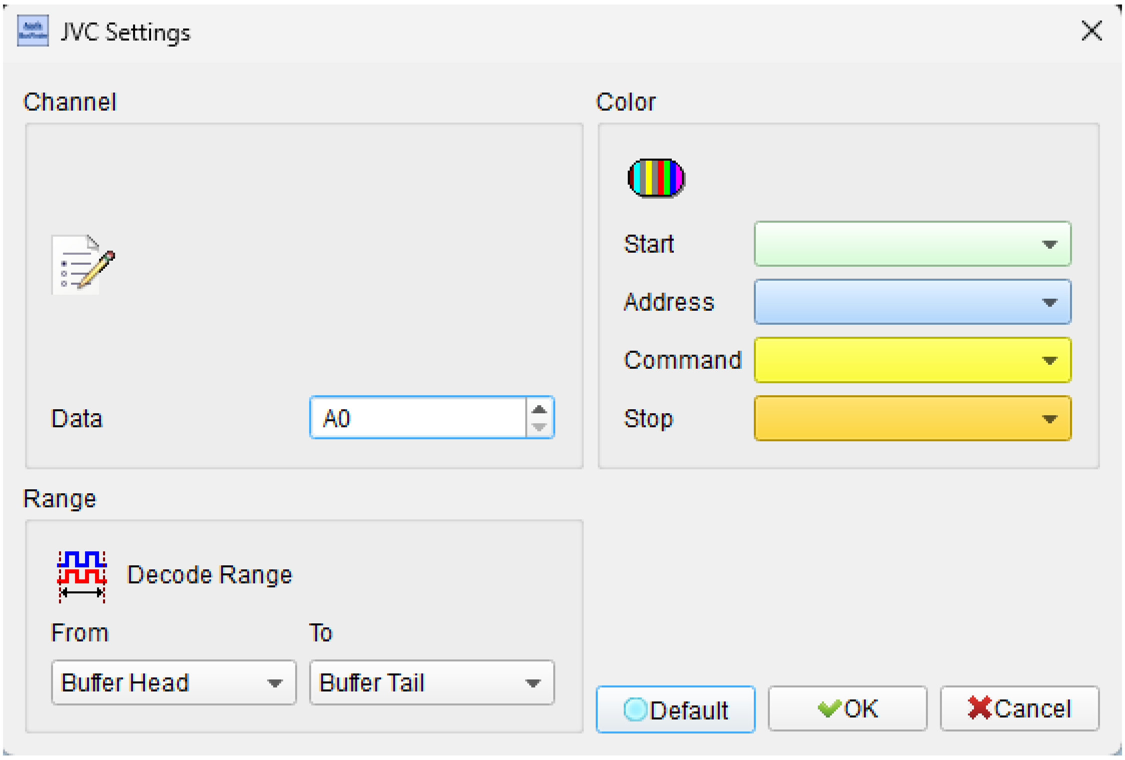
```
#NAME:ARM7-ARM9
#LENGTH:4
#CAPTURE:1
#INST:0, EXTEST
#INST:2, SCAN_N
#INST:3, SAMPLE/PRELOAD
#INST:4, RESTART
#INST:5, CLAMP
#INST:7, HIGHZ
#INST:9, CLAMPZ
#INST:C, INTEST
```

```
#INST:E, IDCODE

#INST:F, BYPASS

#INST:

#ID:01

#BSDL:C:\3256at144_1532.bsd
```

## JVC IR

JVC IR refers to products or systems related to infrared (IR) technology introduced by JVC (Japan Victor Company).

## Settings



JVC Settings

Channel

Data: A0

Color

Start

Address

Command

Stop

Range

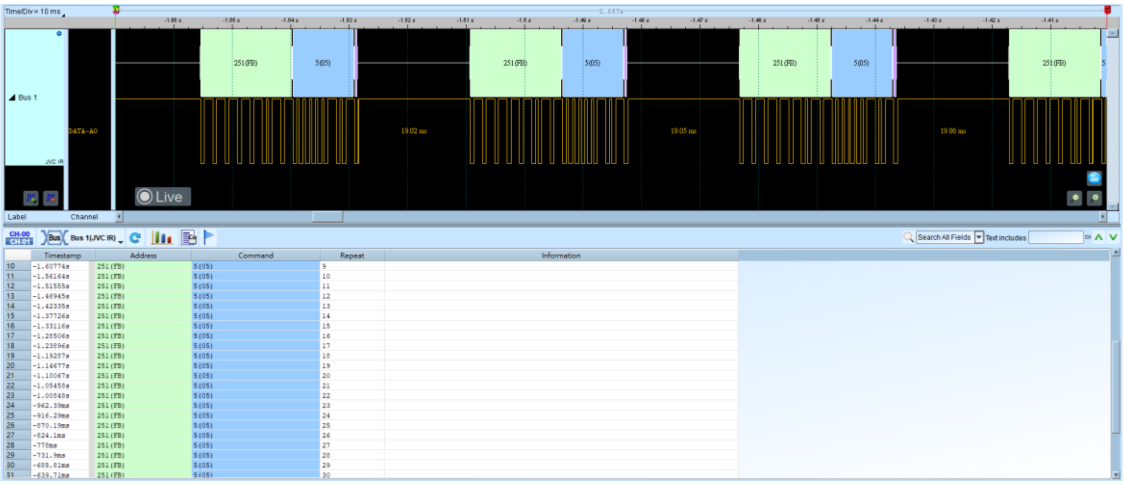
Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

## Result

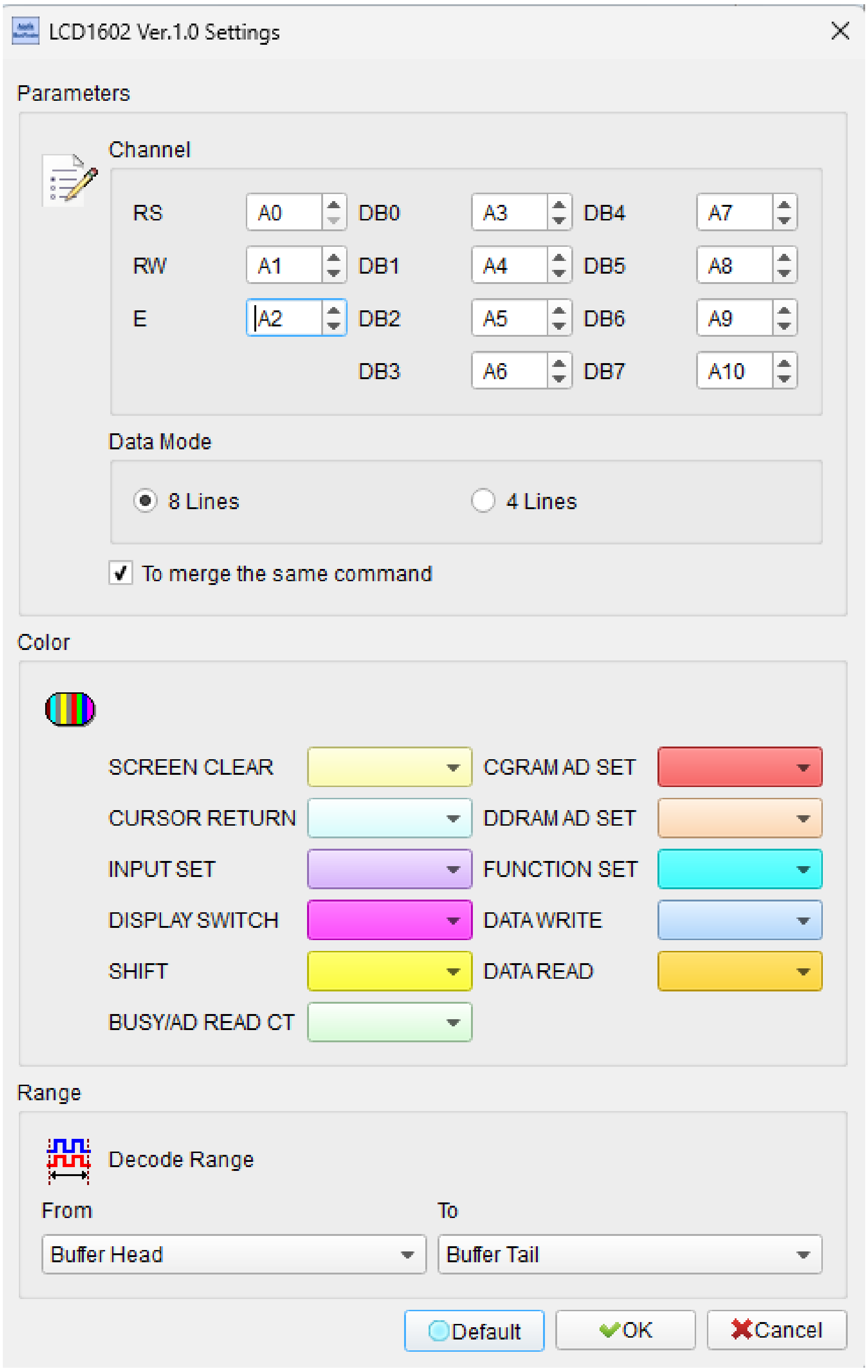


| | Timestamp | Address | Command | Repeat | Information |
|---|---|---|---|---|---|
| 10 | -1.60774s | 251 (FB) | 5 (05) | 9 | |
| 11 | -1.56164s | 251 (FB) | 5 (05) | 10 | |
| 12 | -1.51555s | 251 (FB) | 5 (05) | 11 | |
| 13 | -1.46945s | 251 (FB) | 5 (05) | 12 | |
| 14 | -1.42335s | 251 (FB) | 5 (05) | 13 | |
| 15 | -1.37726s | 251 (FB) | 5 (05) | 14 | |
| 16 | -1.33116s | 251 (FB) | 5 (05) | 15 | |
| 17 | -1.28506s | 251 (FB) | 5 (05) | 16 | |
| 18 | -1.23896s | 251 (FB) | 5 (05) | 17 | |
| 19 | -1.19287s | 251 (FB) | 5 (05) | 18 | |
| 20 | -1.14677s | 251 (FB) | 5 (05) | 19 | |
| 21 | -1.10067s | 251 (FB) | 5 (05) | 20 | |
| 22 | -1.05458s | 251 (FB) | 5 (05) | 21 | |
| 23 | -1.00848s | 251 (FB) | 5 (05) | 22 | |
| 24 | -962.39ms | 251 (FB) | 5 (05) | 23 | |
| 25 | -916.29ms | 251 (FB) | 5 (05) | 24 | |
| 26 | -870.19ms | 251 (FB) | 5 (05) | 25 | |
| 27 | -824.1ms | 251 (FB) | 5 (05) | 26 | |
| 28 | -778ms | 251 (FB) | 5 (05) | 27 | |
| 29 | -731.9ms | 251 (FB) | 5 (05) | 28 | |
| 30 | -685.81ms | 251 (FB) | 5 (05) | 29 | |
| 31 | -639.71ms | 251 (FB) | 5 (05) | 30 | |

## LCD1602

The Liquid Crystal Display 1602 (LCD1602) bus has 11 data bits: Instruction/Data Register Select (RS), Read/Write Select (RW), Enable Select (E) and 8 bits or 4 bits Data Input/Output lines (DB0~DB7/DB0~DB3).

### Settings



LCD1602 Ver.1.0 Settings

Parameters

Channel

|  |  |  |  |  |  |
|---|---|---|---|---|---|
| RS | A0 | DB0 | A3 | DB4 | A7 |
| RW | A1 | DB1 | A4 | DB5 | A8 |
| E | A2 | DB2 | A5 | DB6 | A9 |
|  |  | DB3 | A6 | DB7 | A10 |

Data Mode

● 8 Lines ○ 4 Lines

☑ To merge the same command

Color

|  |  |  |  |
|---|---|---|---|
| SCREEN CLEAR | 🟨 | CGRAM AD SET | 🟥 |
| CURSOR RETURN | 🟦 | DDRAM AD SET | 🟧 |
| INPUT SET | 🟪 | FUNCTION SET | 🟦 |
| DISPLAY SWITCH | 🟪 | DATA WRITE | 🟦 |
| SHIFT | 🟨 | DATA READ | 🟧 |
| BUSY/AD READ CT | 🟩 |  |  |

Range

Decode Range

From: Buffer Head

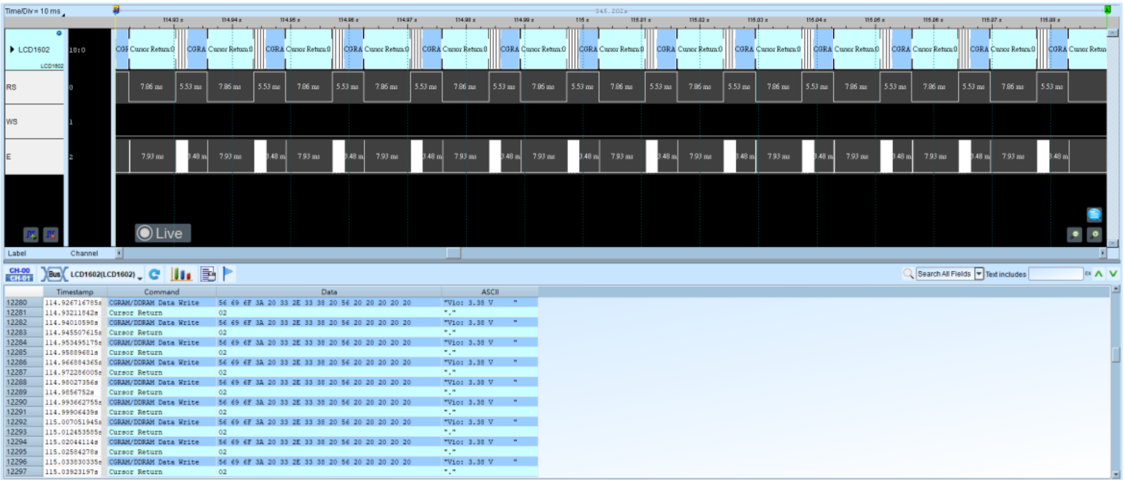
To: Buffer Tail

**Channel:** Show the selected channels (RS: CH0, RW: CH1,..., DB0: CH10).

**Data Mode:** 8 lines or 4 lines.

**To merge the same command:** Merge data with its command.

### Result

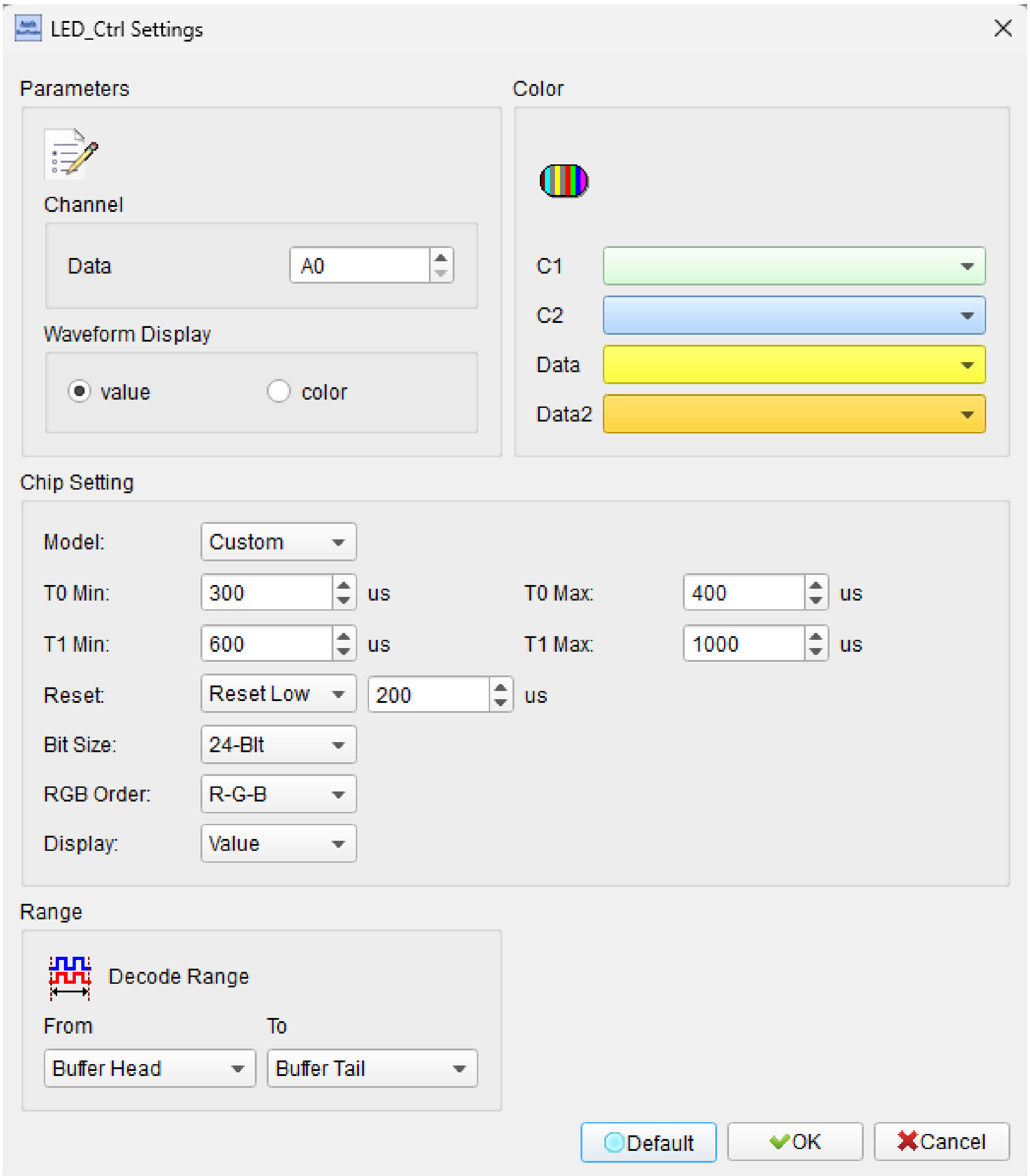


| | Timestamp | Command | Data | ASCII |
|---|---|---|---|---|
| 12280 | 114.926716785s | CGRAM/DDRAM Data Write | 56 69 6F 3A 20 33 2E 33 38 20 56 20 20 20 20 20 | "Vio: 3.38 V      " |
| 12281 | 114.93211842s | Cursor Return | 02 | "." |
| 12282 | 114.94010598s | CGRAM/DDRAM Data Write | 56 69 6F 3A 20 33 2E 33 38 20 56 20 20 20 20 20 | "Vio: 3.38 V      " |
| 12283 | 114.945507615s | Cursor Return | 02 | "." |
| 12284 | 114.953495175s | CGRAM/DDRAM Data Write | 56 69 6F 3A 20 33 2E 33 38 20 56 20 20 20 20 20 | "Vio: 3.38 V      " |
| 12285 | 114.95889681s | Cursor Return | 02 | "." |
| 12286 | 114.966884365s | CGRAM/DDRAM Data Write | 56 69 6F 3A 20 33 2E 33 38 20 56 20 20 20 20 20 | "Vio: 3.38 V      " |
| 12287 | 114.972286005s | Cursor Return | 02 | "." |
| 12288 | 114.98027356s | CGRAM/DDRAM Data Write | 56 69 6F 3A 20 33 2E 33 38 20 56 20 20 20 20 20 | "Vio: 3.38 V      " |
| 12289 | 114.9856752s | Cursor Return | 02 | "." |
| 12290 | 114.993662755s | CGRAM/DDRAM Data Write | 56 69 6F 3A 20 33 2E 33 38 20 56 20 20 20 20 20 | "Vio: 3.38 V      " |
| 12291 | 114.99906439s | Cursor Return | 02 | "." |
| 12292 | 115.007051945s | CGRAM/DDRAM Data Write | 56 69 6F 3A 20 33 2E 33 38 20 56 20 20 20 20 20 | "Vio: 3.38 V      " |
| 12293 | 115.012453585s | Cursor Return | 02 | "." |
| 12294 | 115.02044114s | CGRAM/DDRAM Data Write | 56 69 6F 3A 20 33 2E 33 38 20 56 20 20 20 20 20 | "Vio: 3.38 V      " |
| 12295 | 115.02584278s | Cursor Return | 02 | "." |
| 12296 | 115.033830335s | CGRAM/DDRAM Data Write | 56 69 6F 3A 20 33 2E 33 38 20 56 20 20 20 20 20 | "Vio: 3.38 V      " |
| 12297 | 115.03923197s | Cursor Return | 02 | "." |

## LED_Ctrl

LED Controls are made specifically for digital type LEDs. These LEDs contain a special IC chips that allow the user to control each LED or section of LEDs.

## Settings



LED_Ctrl Settings

Parameters

Channel

Data: A0

Waveform Display

● value ○ color

Color

C1: 🟩

C2: 🟦

Data: 🟨

Data2: 🟧

Chip Setting

| | | | | | |
|---|---|---|---|---|---|
| Model: | Custom | | | | |
| T0 Min: | 300 | us | T0 Max: | 400 | us |
| T1 Min: | 600 | us | T1 Max: | 1000 | us |
| Reset: | Reset Low | 200 us | | | |
| Bit Size: | 24-BIt | | | | |
| RGB Order: | R-G-B | | | | |
| Display: | Value | | | | |

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Show the selected channels.

**Waveform display:** show value or RGB color in waveform area.

**Chip setting:**

**Model: User can select the IC model. We now support** TM1814、WS2811、WS2812、RT7905、HZ0028 and Custom。

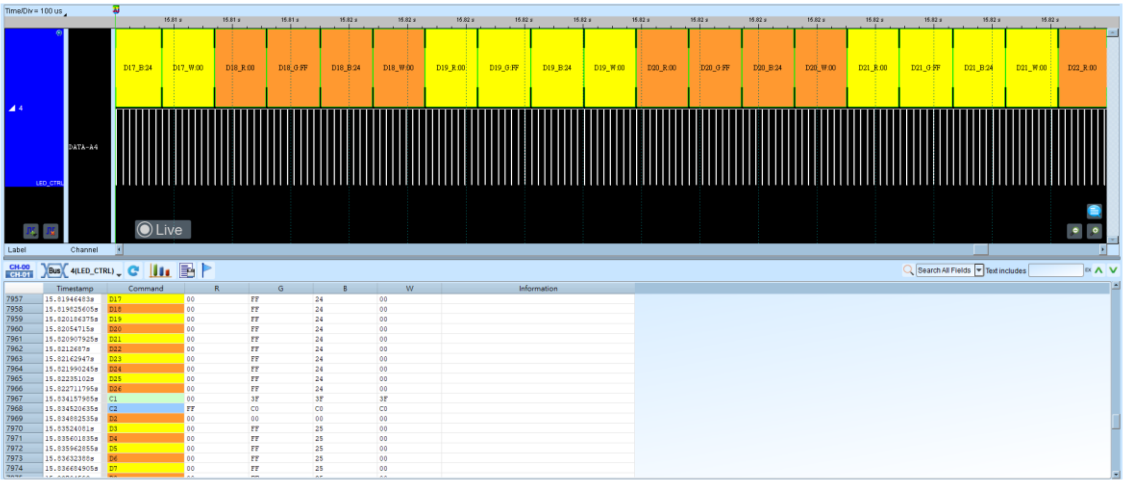
**Condition:** Idle high

If the time that waveform goes to low between T0 min and T0 max, the bit will decode as zero. If it's between T1 min and T1 max, the bit will decode as one.

**Reset:** if the waveform keeps at high potential over the time, the decoder will reset the start bit.

**Bit size:** choose 32-bit (WRGB) or 24-bit (RGB).

**Result**

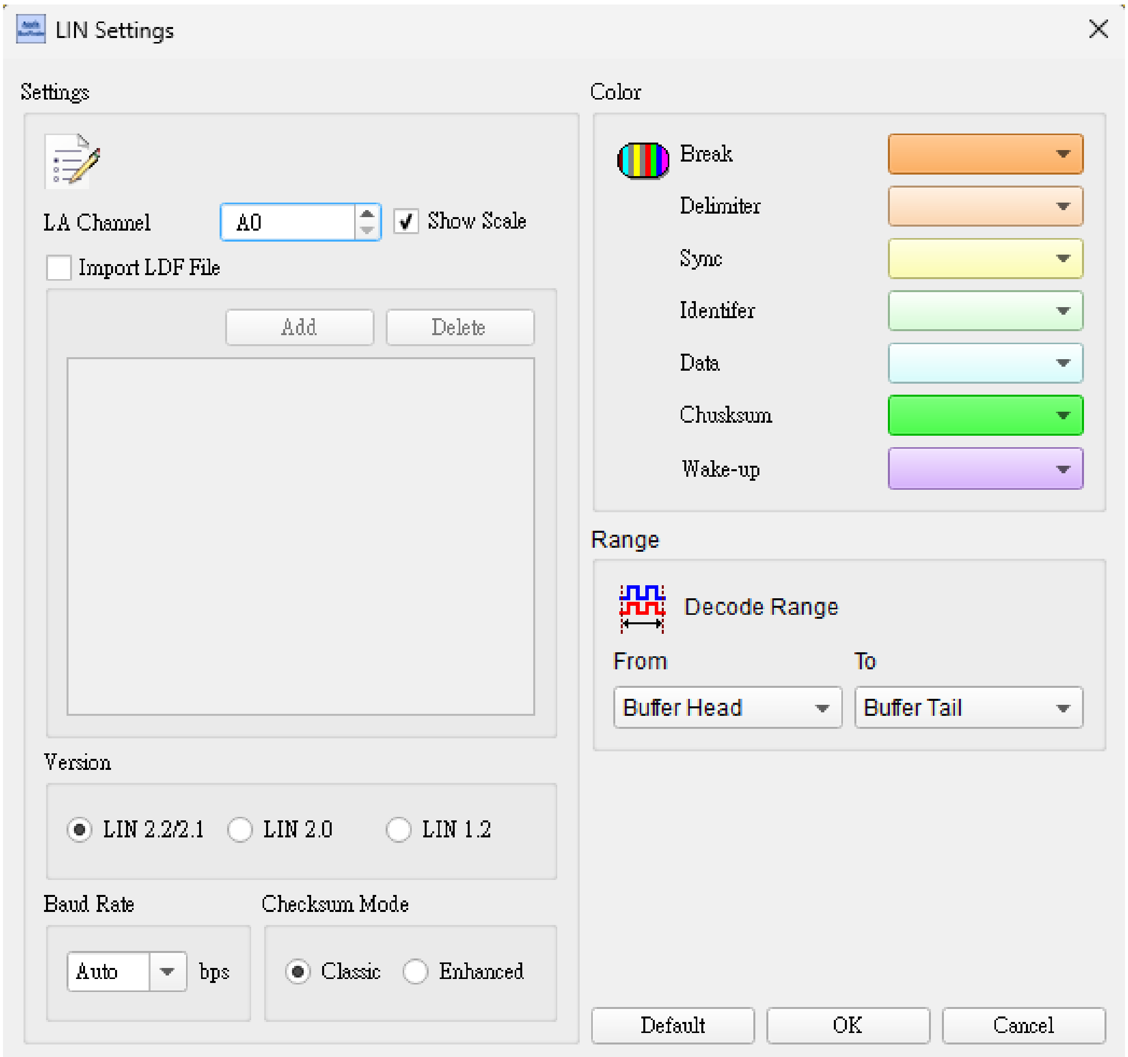


| | Timestamp | Command | R | G | B | W | Information |
|---|---|---|---|---|---|---|---|
| 7957 | 15.81946483s | D17 | 00 | FF | 24 | 00 | |
| 7958 | 15.819625605s | D18 | 00 | FF | 24 | 00 | |
| 7959 | 15.820186375s | D19 | 00 | FF | 24 | 00 | |
| 7960 | 15.82054715s | D20 | 00 | FF | 24 | 00 | |
| 7961 | 15.820907925s | D21 | 00 | FF | 24 | 00 | |
| 7962 | 15.8212687s | D22 | 00 | FF | 24 | 00 | |
| 7963 | 15.82162947s | D23 | 00 | FF | 24 | 00 | |
| 7964 | 15.821990245s | D24 | 00 | FF | 24 | 00 | |
| 7965 | 15.82235102s | D25 | 00 | FF | 24 | 00 | |
| 7966 | 15.822711795s | D26 | 00 | FF | 24 | 00 | |
| 7967 | 15.834157965s | C1 | 00 | 3F | 3F | 3F | |
| 7968 | 15.834520635s | C2 | FF | C0 | C0 | C0 | |
| 7969 | 15.834882535s | D2 | 00 | 00 | 00 | 00 | |
| 7970 | 15.83524081s | D3 | 00 | FF | 25 | 00 | |
| 7971 | 15.835601835s | D4 | 00 | FF | 25 | 00 | |
| 7972 | 15.835962855s | D5 | 00 | FF | 25 | 00 | |
| 7973 | 15.83632388s | D6 | 00 | FF | 25 | 00 | |
| 7974 | 15.836684905s | D7 | 00 | FF | 25 | 00 | |

## LIN

LIN (Local Interconnect Network) is a serial network protocol used for communication between components in vehicles. LIN may be used also over the vehicle's battery power-line with a special LIN over DC powerline (DC-LIN) transceiver.

### Settings



LIN Settings

Settings

LA Channel: A0 ☑ Show Scale

□ Import LDF File

Version: ● LIN 2.2/2.1 ○ LIN 2.0 ○ LIN 1.2

Baud Rate: Auto bps

Checksum Mode: ● Classic ○ Enhanced

Color

Break
Delimiter
Sync
Identifer
Data
Chusksum
Wake-up

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**LA Channel:** Show the selected channel.

**Show Scale:** It will show the scales according to the bit width on the waveform. Enabled when checked.

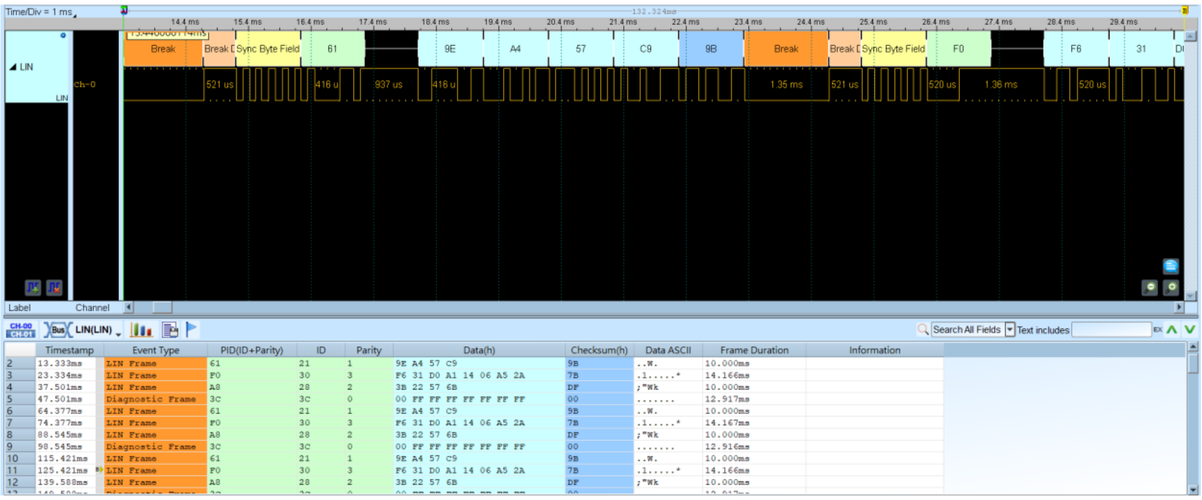
**Import LDF File:** Enable users to import the LIN Description File. Click on Add and select LDF file.

**Version:** Different versions can be selected for LIN analysis. The Checksum verification after Lin version 2.0 has changed to two modes. If user need to use the enhanced verification below, user should select the version after 2.0 to use it.

**Baud rate:** Show the selected baud rate.

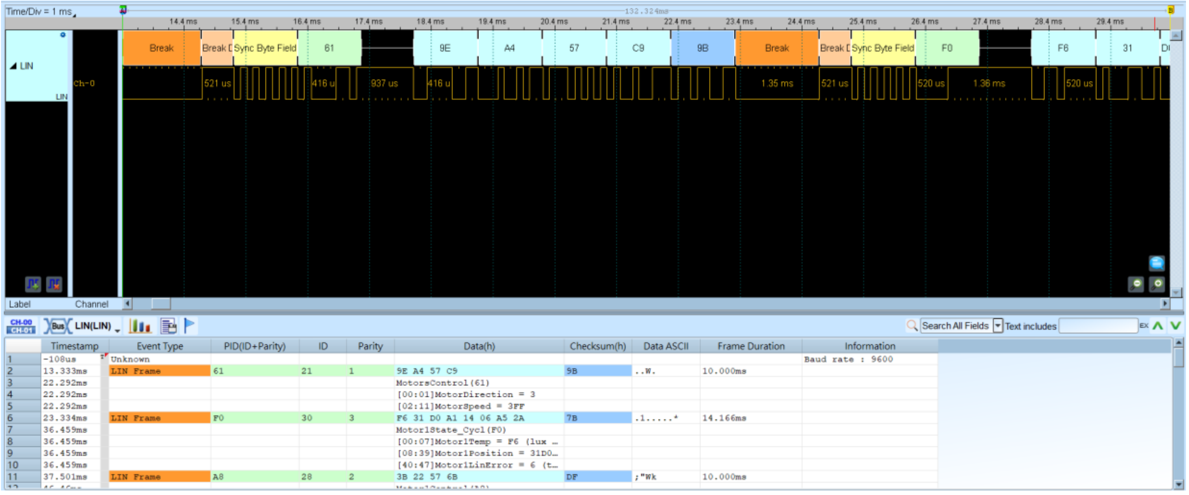
**Checksum Mode:** Select the proper baud rate of the signal before the analysis. When set to auto, it will automatically detect the baud rate that matches the signal to be analyzed.

## Result



| | Timestamp | Event Type | PID(ID+Parity) | ID | Parity | Data(h) | Checksum(h) | Data ASCII | Frame Duration | Information |
|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 13.333ms | LIN Frame | 61 | 21 | 1 | 9E A4 57 C9 | 9B | ..W. | 10.000ms | |
| 3 | 23.334ms | LIN Frame | F0 | 30 | 3 | F6 31 D0 A1 14 06 A5 2A | 7B | .1.....* | 14.166ms | |
| 4 | 37.501ms | LIN Frame | A8 | 28 | 2 | 3B 22 57 6B | DF | ;"Wk | 10.000ms | |
| 5 | 47.501ms | Diagnostic Frame | 3C | 3C | 0 | 00 FF FF FF FF FF FF FF | 00 | ....... | 12.917ms | |
| 6 | 64.377ms | LIN Frame | 61 | 21 | 1 | 9E A4 57 C9 | 9B | ..W. | 10.000ms | |
| 7 | 74.377ms | LIN Frame | F0 | 30 | 3 | F6 31 D0 A1 14 06 A5 2A | 7B | .1.....* | 14.167ms | |
| 8 | 88.545ms | LIN Frame | A8 | 28 | 2 | 3B 22 57 6B | DF | ;"Wk | 10.000ms | |
| 9 | 98.545ms | Diagnostic Frame | 3C | 3C | 0 | 00 FF FF FF FF FF FF FF | 00 | ....... | 12.916ms | |
| 10 | 115.421ms | LIN Frame | 61 | 21 | 1 | 9E A4 57 C9 | 9B | ..W. | 10.000ms | |
| 11 | 125.421ms | LIN Frame | F0 | 30 | 3 | F6 31 D0 A1 14 06 A5 2A | 7B | .1.....* | 14.166ms | |
| 12 | 139.588ms | LIN Frame | A8 | 28 | 2 | 3B 22 57 6B | DF | ;"Wk | 10.000ms | |

## The decoder with LDF

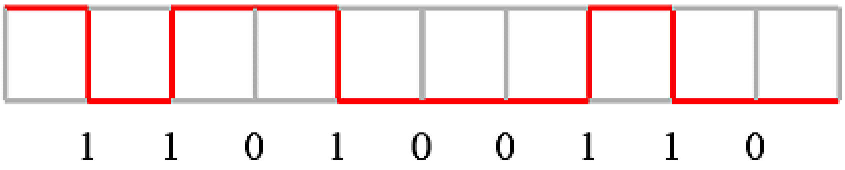


| | Timestamp | Event Type | PID(ID+Parity) | ID | Parity | Data(h) | Checksum(h) | Data ASCII | Frame Duration | Information |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -108us | Unknown | | | | | | | | Baud rate : 9600 |
| 2 | 13.333ms | LIN Frame | 61 | 21 | 1 | 9E A4 57 C9 | 9B | ..W. | 10.000ms | |
| 3 | 22.292ms | | | | | MotorsControl(61) | | | | |
| 4 | 22.292ms | | | | | [00:01]MotorDirection = 3 | | | | |
| 5 | 22.292ms | | | | | [02:11]MotorSpeed = 3FF | | | | |
| 6 | 23.334ms | LIN Frame | F0 | 30 | 3 | F6 31 D0 A1 14 06 A5 2A | 7B | .1.....* | 14.166ms | |
| 7 | 36.459ms | | | | | Motor1State_Cycl(F0) | | | | |
| 8 | 36.459ms | | | | | [00:07]Motor1Temp = F6 (lux … | | | | |
| 9 | 36.459ms | | | | | [08:39]Motor1Position = 31D0… | | | | |
| 10 | 36.459ms | | | | | [40:47]Motor1LinError = 6 (t… | | | | |
| 11 | 37.501ms | LIN Frame | A8 | 28 | 2 | 3B 22 57 6B | DF | ;"Wk | 10.000ms | |

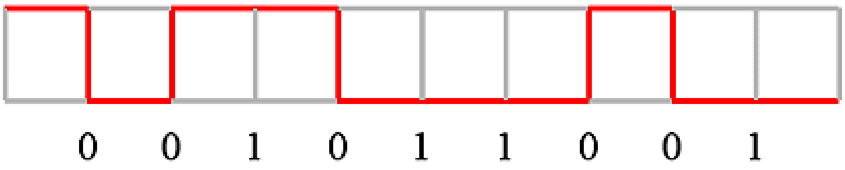
### Line Decoding

**NRZI (Non return to zero, inverted):** Non return to zero, inverted (NRZI) is a method of mapping a binary signal to a physical signal for transmission over some transmission media. The two level NRZI signal has a transition at a clock boundary if the bit being transmitted is a logical one, and does not have a transition if the bit being transmitted is a logical zero. There are two modes:

**NRZI (Transition occurs for a one):** A 1 is represented by a transition of the physical level, a 0 has no transition.

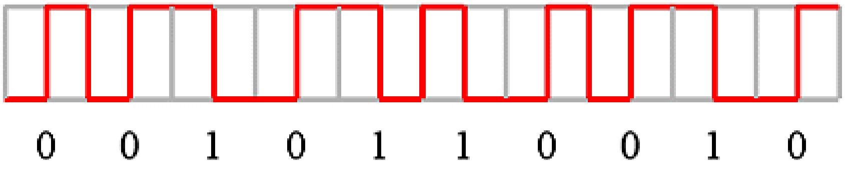


**NRZI (Transition occurs for a zero):** A 0 is represented by a transition of the physical level, a 1 has no transition.

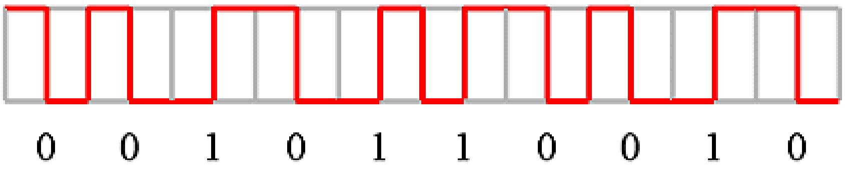


**Manchester:** In telecommunication, Manchester code is a line code in which the encoding of each data bit has at least one transition and occupies the same time. It is, therefore, self-clocking, which means that a clock signal can be recovered from the encoded data. There are three modes:

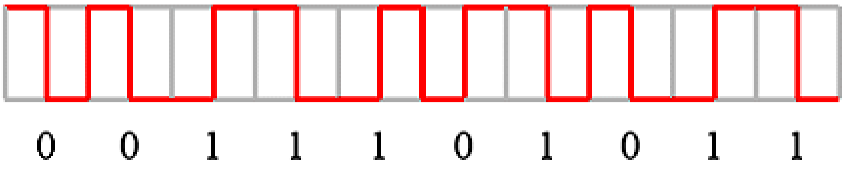
**Manchester (Thomas):** A 0 is expressed by a low-to-high transition, a 1 by high-to-low transition.



**Manchester (IEEE802.3):** A 1 is expressed by a low-to-high transition, a 0 by high-to-low transition.

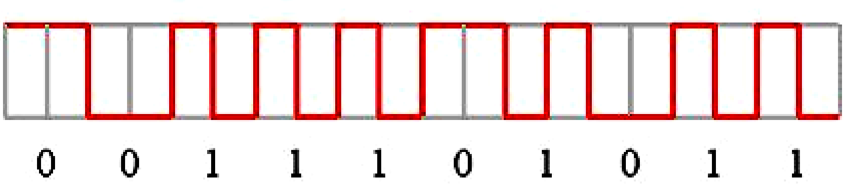


**Differential Manchester:** A 1 bit is indicated by making the first half of the signal equal to the last half of the previous bit's signal i.e. no transition at the start of the bit-time. A '0' bit is indicated by making the first half of the signal opposite to the last half of the previous bit's signal i.e. a zero bit is indicated by a transition at the beginning of the bit-time. In the middle of the bit-time there is always a transition, whether from high to low, or low to high. A reversed scheme is possible, and no advantage is given by using either scheme.



**Bi-phase Mark:** The bi-phase mark code (also called FM1 code) is a type of encoding for binary data streams. When a binary data stream is sent without modification via a channel, there can be long series of logical ones or zeros without any transitions which makes clock recovery and synchronization difficult.

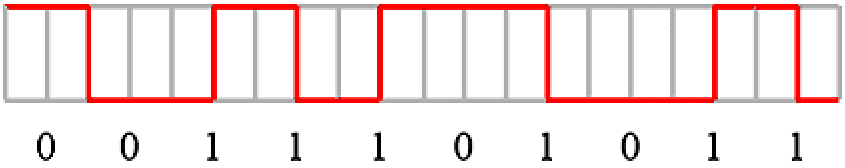
When encoding, the symbol rate must be twice the bitrate of the original signal. Every bit of the original data is represented as two logical states that together form a bit.



**Miller:** Delay encoding is also known as Miller encoding.

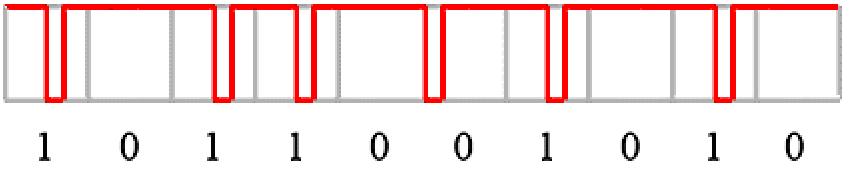
In telecommunications, delay encoding is the encoding of binary data to form a two-level signal such that (a) a "0" causes no change of signal level unless it is followed by another "0" in which case a transition to the other level takes place at the end of the first bit period; and (b) a "1" causes a transition from one level to the other in the middle of the bit period.

Delay encoding is used primarily for encoding radio signals because the frequency spectrum of the encoded signal contains less low-frequency energy than a conventional non-return-to-zero (NRZ) signal and less high-frequency energy than a bi-phase signal.

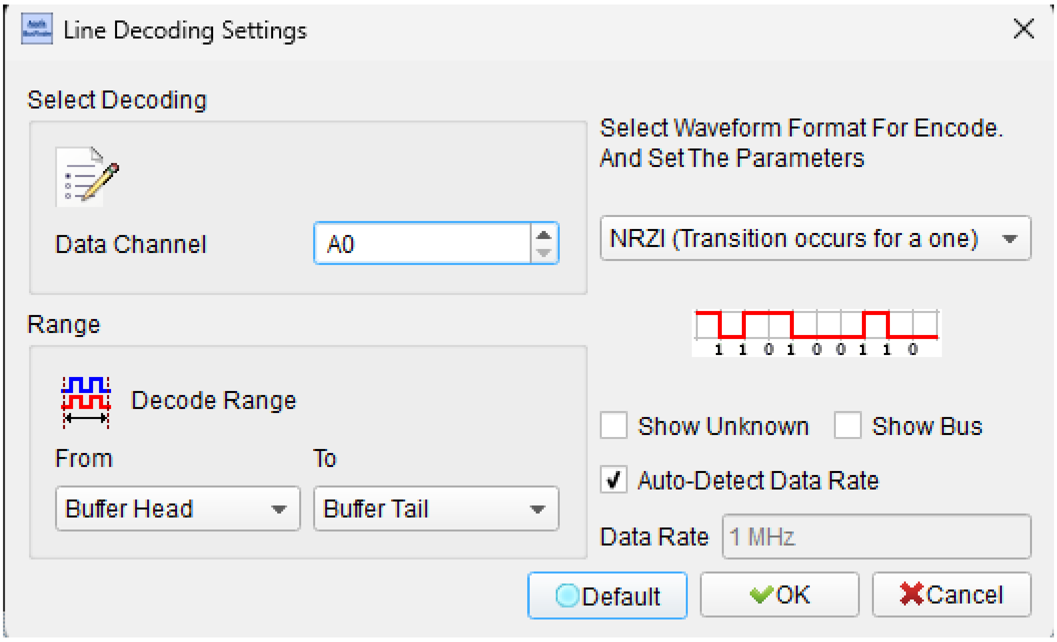


**Modified Miller:** The Modified Miller (M 2) demodulator facilitates demodulation of M 2 modulation data to NRZ-L (non-return-to-zero-level) data, composed of a data separation circuit for producing synchronizing clock pulses from the M 2 modulation data which is reproduced by a data recording device and separating the M 2 modulation data into clock bits and data bits, and an M 2 modulation data demodulation circuit for producing NRZ - L data by utilizing the clock bits, data bits and synchronizing clock pulses which are output from the data separation circuit.

This structure enables the M 2 modulation data which is input to the M 2 demodulation circuit to be easily demodulated to an NRZ - L type data signal by means of a very simple circuit structure.　An example is as below:



**Settings**



Line Decoding Settings

Select Decoding

Data Channel: A0

Select Waveform Format For Encode. And Set The Parameters

NRZI (Transition occurs for a one)

Range

Decode Range

From: Buffer Head

To: Buffer Tail

☐ Show Unknown ☐ Show Bus

☑ Auto-Detect Data Rate

Data Rate: 1 MHz

**Select Decoding:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

Select the line code that user want to decode:

I. NRZI (Transition occurs for a one)

II. NRZI (Transition occurs for a zero)

III. Manchester (Thomas)

IV. Manchester (IEEE802.3)

V. Differential Manchester

VI. Biphase Mark Decode

VII. Miller

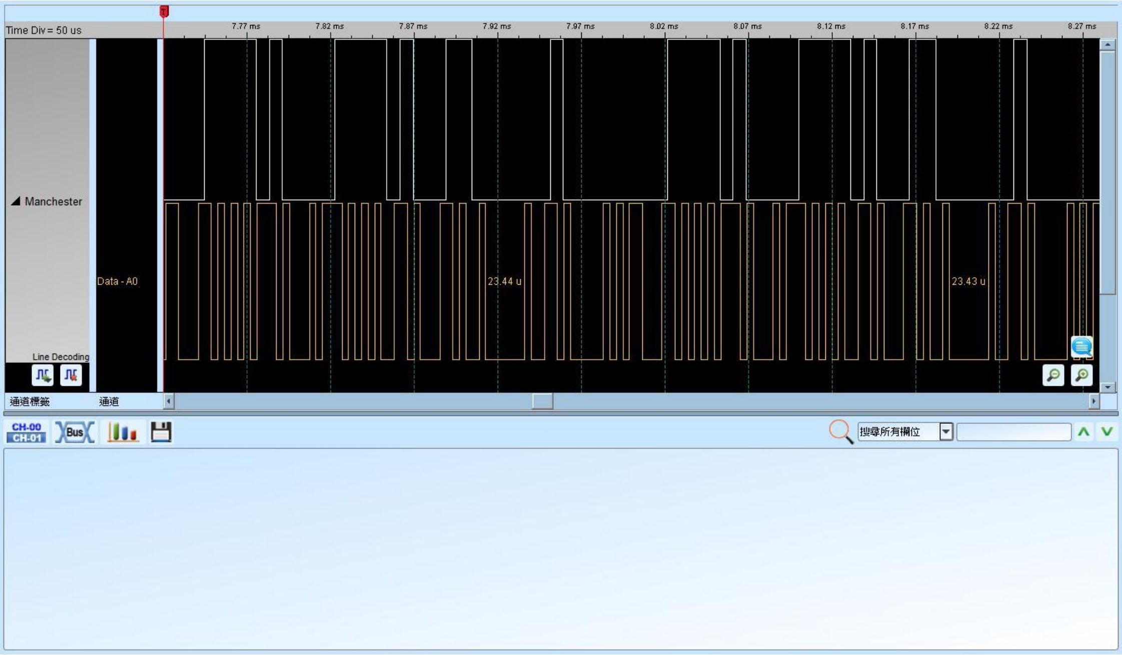
VIII. Modified Miller

**Show Unknown:** Display unknown data.

**Show Bus:** Display bus data.

**Auto-Detect Data Rate:** Enter the Data Rate manually if the Auto-Detect Date Rate is not selected.

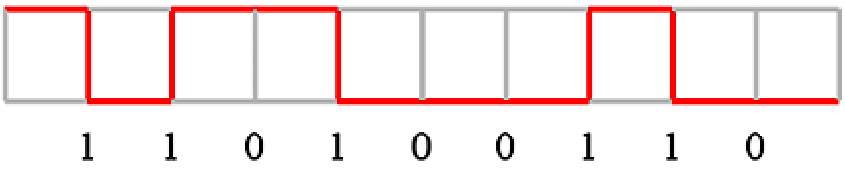
**Result**



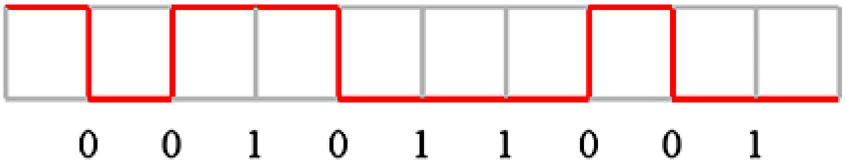
## Line Encoding

**NRZI (Non return to zero, inverted):** Non return to zero, inverted (NRZI) is a method of mapping a binary signal to a physical signal for transmission over some transmission media. The two level NRZI signal has a transition at a clock boundary if the bit being transmitted is a logical one, and does not have a transition if the bit being transmitted is a logical zero. There are two modes:

**NRZI (Transition occurs for a one):** A 1 is represented by a transition of the physical level, a 0 has no transition.

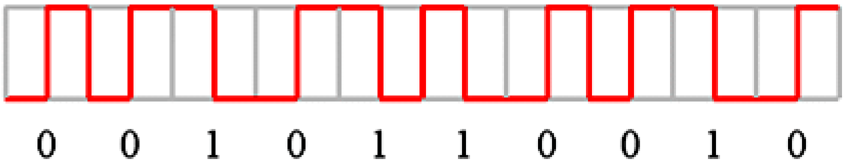


**NRZI (Transition occurs for a zero):** A 0 is represented by a transition of the physical level, a 1 has no transition.

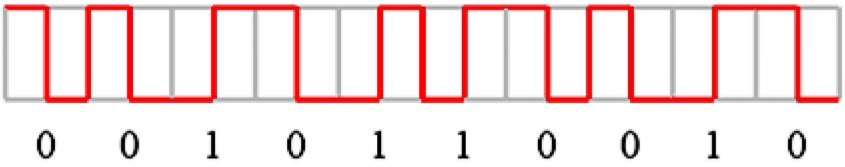


**Manchester:** In telecommunication, Manchester code is a line code in which the encoding of each data bit has at least one transition and occupies the same time. It is, therefore, self-clocking, which means that a clock signal can be recovered from the encoded data. There are three modes:

**Manchester (Thomas):** A 0 is expressed by a low-to-high transition, a 1 by high-to-low transition.



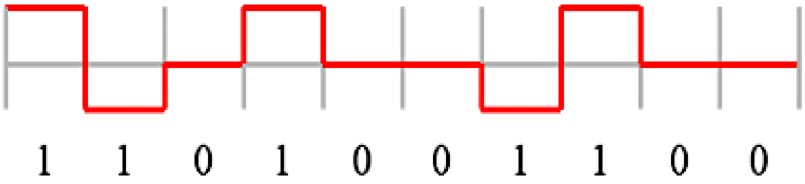
**Manchester (IEEE802.3):** A 1 is expressed by a low-to-high transition, a 0 by high-to-low transition.



**Differential Manchester:** A 1 bit is indicated by making the first half of the signal equal to the last half of the previous bit, i.e. no transition at the start of the bit-time. A '0' bit is indicated by making the first half of the signal opposite to the last half of the previous bit's signal i.e. a zero bit is indicated by a transition at the beginning of the bit-time. In the middle of the bit-time there is always a transition, whether from high to low, or low to high. A reversed scheme is possible, and no advantage is given by using either scheme.

**AMI (Alternate Mark Inversion):** There are four modes:

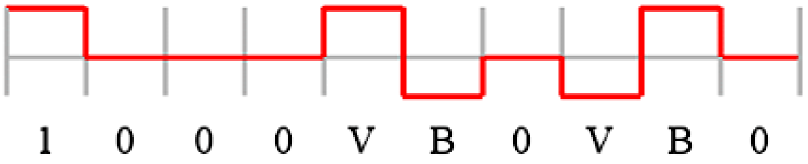
**AMI (Standard):** AMI (Alternate Mark Inversion) is a synchronous clock encoding technique that uses bipolar pulses to represent logical 1 value. It is therefore a three level system. A logical 0s is represented by no symbol, and a logical 1 is represented by alternating-polarity pulses.



**AMI (B8ZS):** Bipolar-8-Zero Substitution

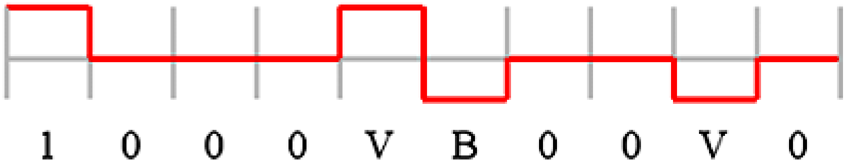
If 1 is +, 00000000 is represented to 000+-0-+

1 is -, 00000000 is represented to 000-+0+-

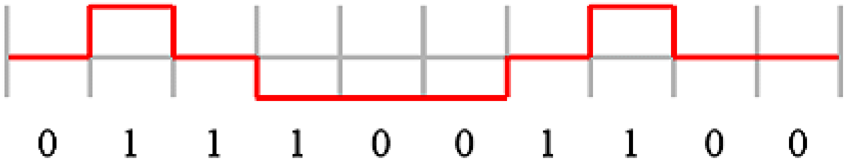


**AMI (HDB3): High Density Bipolar 3**

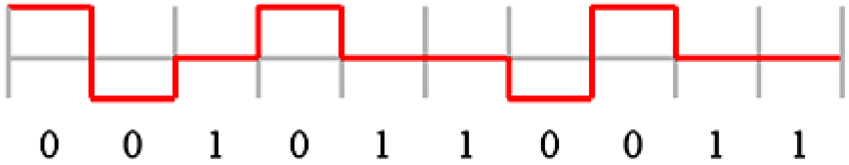
The HDB3 code is a bipolar signaling technique (i.e. relies on the transmission of both positive and negative pulses). It is based on Alternate Mark Inversion (AMI), but extends this by inserting violation codes whenever there is a run of 4 or more 0's. This and similar (more complex) codes have replaced AMI in modern distribution networks. The encoding rules follow those for AMI, except that sequences of four consecutive 0’s are encoding using a special "violation" bit. This bit has the same polarity as the last 1-bit which was sent using the AMI encoding rule. The purpose of this is to prevent long runs of 0's in the data stream that may otherwise prevent a DPLL from tracking the center of each bit. Such a code is sometimes called a "run length limited" code, since it limits the runs of 0’s that would otherwise be produced by AMI. One refinement is necessary, to prevent a dc voltage being introduced by excessive runs of zeros. This refinement is to encode any pattern of more than four bits as B00V, where B is a balancing pulse. The value of B is assigned as + or -, so as to make alternate "V"s of opposite polarity. The receiver removes all Violation pulses, but in addition a violation preceded by two zeros and a pulse is treated as the "B00V" pattern and both the violation and balancing pulse are removed from the received bit stream. This restores the original bit stream.



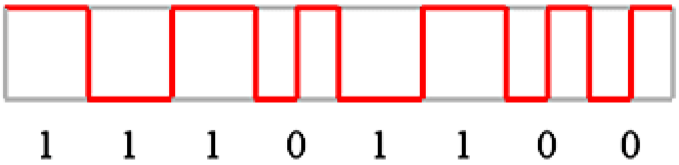
**MLT-3: Multilevel Transmission 3:** A 0 means no transition happens, a 1 is represented by a transition (0, +, 0, -).



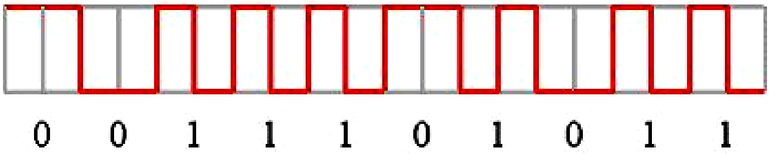
**Pseudoternary:** A 1 is always zero, a 0 is represented by a transition (+, -).



**CMI (Coded Mark Inversion):** A zero is sent a low to high [01] transition, while a one is sent as either a one [1] or zero [0] depending on the previous state. If the previous state was high the one is sent as a zero [0], if it was low the one is sent as a one [1].



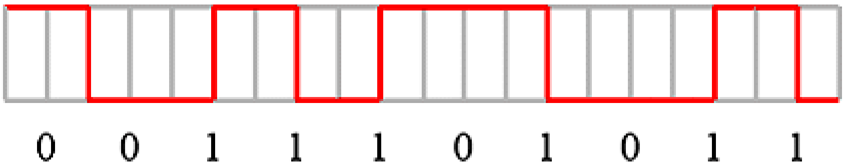
**Bi-phase Mark:** The bi-phase mark code (also called FM1 code) is a type of encoding for binary data streams. When a binary data stream is sent without modification via a channel, there can be long series of logical ones or zeros without any transitions that make clock recovery and synchronization difficult. When encoding, the symbol rate must be twice the bitrate of the original signal. Every bit of the original data is represented as two logical states that form a bit.



**Miller:** Delay encoding is also known as Miller encoding.

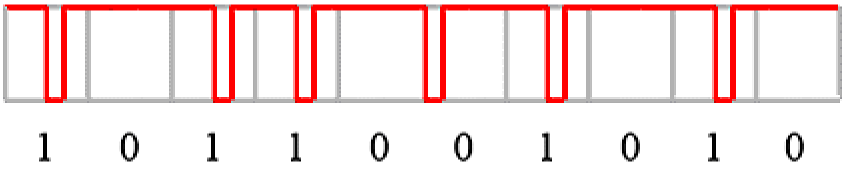
In telecommunications, delay encoding is the encoding of binary data to form a two-level signal such that (a) a "0" causes no change of signal level unless it is followed by another "0" in which case a transition to the other level takes place at the end of the first bit period; and (b) a "1" causes a transition from one level to the other in the middle of the bit period.

Delay encoding is used primarily for encoding radio signals because the frequency spectrum of the encoded signal contains less low-frequency energy than a conventional non-return-to-zero (NRZ) signal and less high-frequency energy than a bi-phase signal.

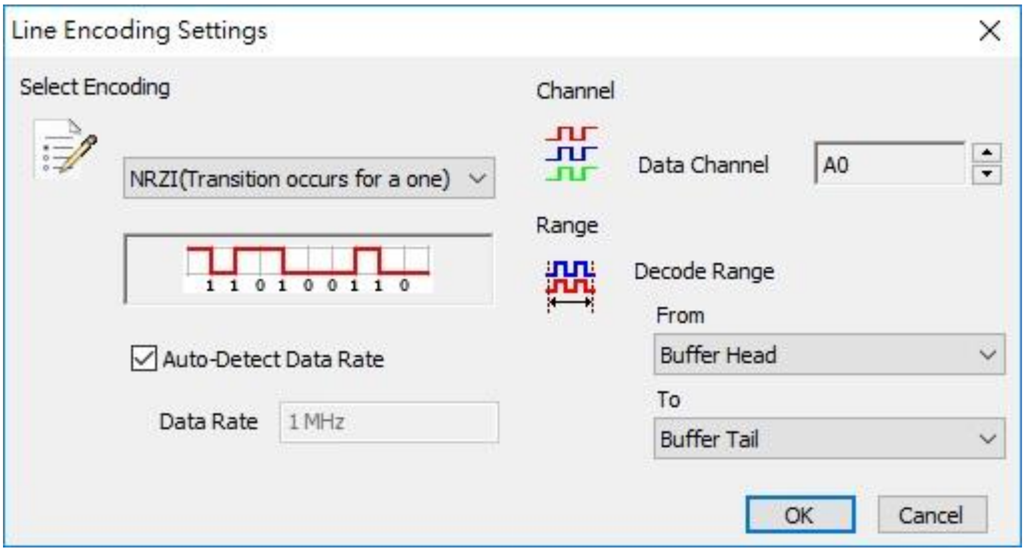


**Modified Miller:** The Modified Miller (M 2) demodulator facilitates demodulation of M 2 modulation data to NRZ-L (non-return-to-zero-level) data, composed of a data separation circuit for producing synchronizing clock pulses from the M 2 modulation data which is reproduced by a data recording device and separating the M 2 modulation data into clock bits and data bits, and an M 2 modulation data demodulation circuit for producing NRZ - L data by utilizing the clock bits, data bits and synchronizing clock pulses which are output from the data separation circuit.

This structure enables the M 2 modulation data which is input to the M 2 demodulation circuit to be easily demodulated to an NRZ - L type data signal by means of a very simple circuit structure.　An example is as below:



**Settings**


Line Encoding Settings

Select Encoding: NRZI(Transition occurs for a one)

☑ Auto-Detect Data Rate

Data Rate: 1 MHz

Channel

Data Channel: A0

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Select Encoding:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

Select the line code you want to encode:

I. NRZI (Transition occurs for a one)
II. NRZI (Transition occurs for a zero)
III. Manchester (Thomas)
IV. Manchester (IEEE802.3)
V. Differential Manchester
VI. AMI (Standard)
VII. AMI (B8ZS)

VIII. AMI (HDB3)

IX. Pseudoternary

X. MLT-3

XI. CMI

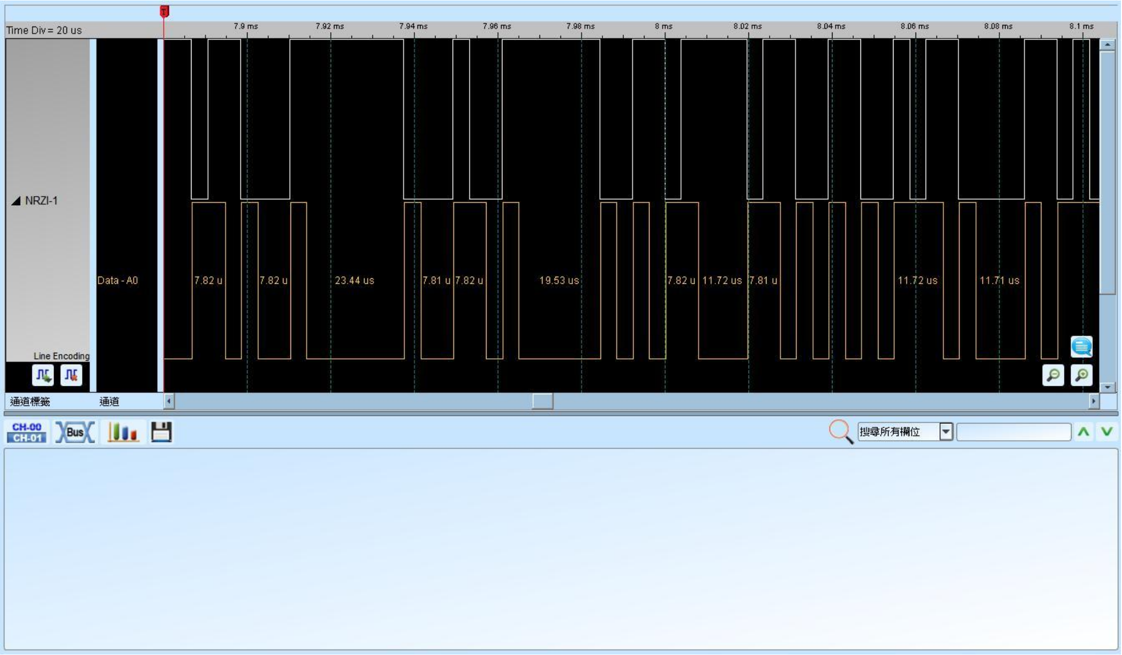
XII. Biphase Mark Encode

XIII. Miller

XIV. Modified Miller

**Auto-Detect Data Rate:** Enter the Data Rate manually if the Auto-Detect Date Rate is not selected.

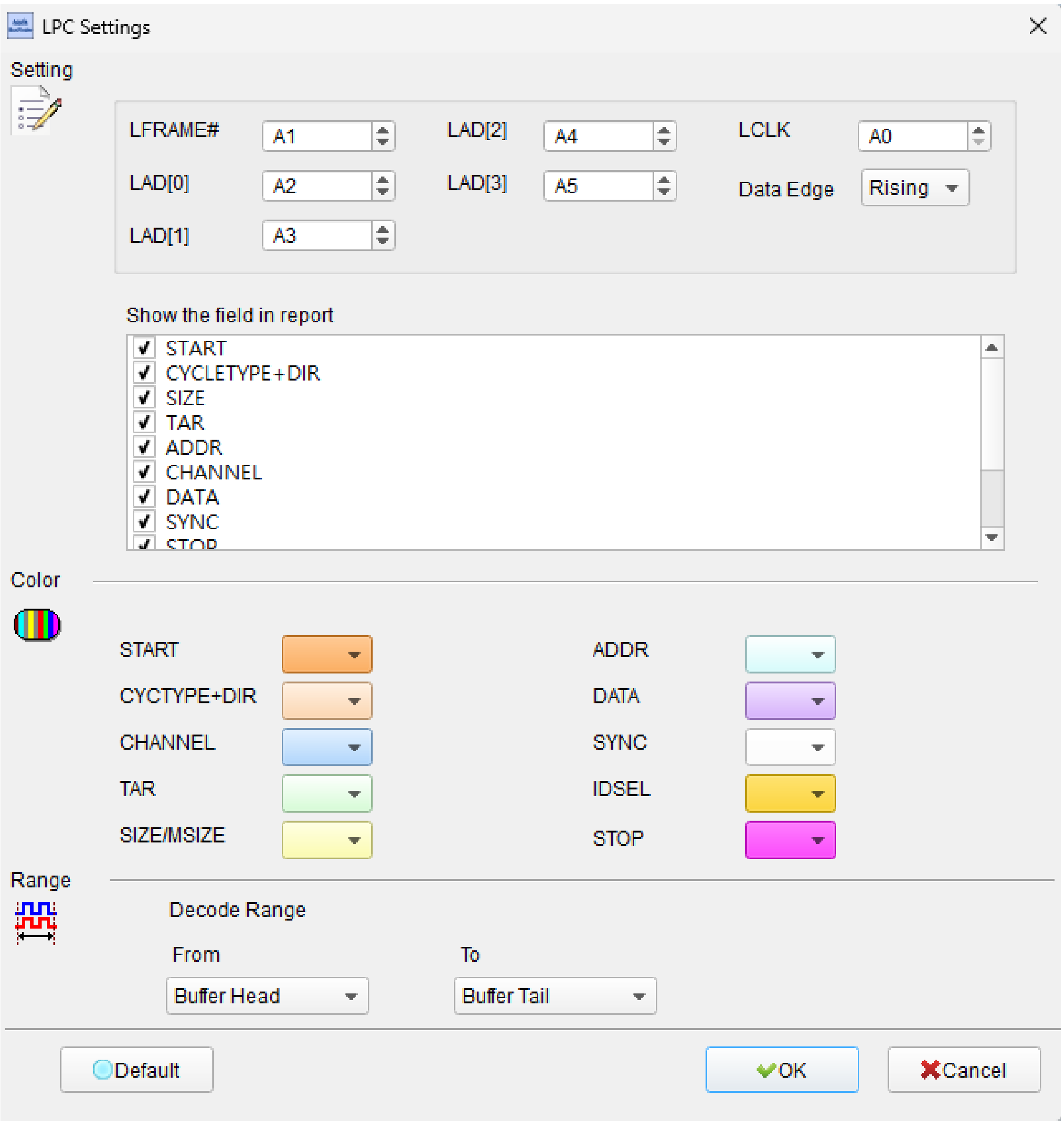
**Result**



## LPC

The LPC bus, for the data transmissions, was developed by Intel to replace the ISA bus.

## Settings



LFRAME# A1
LAD[0] A2
LAD[1] A3
LAD[2] A4
LAD[3] A5
LCLK A0
Data Edge Rising

Show the field in report

- ☑ START
- ☑ CYCLETYPE+DIR
- ☑ SIZE
- ☑ TAR
- ☑ ADDR
- ☑ CHANNEL
- ☑ DATA
- ☑ SYNC
- ☑ STOP

START 🟧
CYCTYPE+DIR 🟧
CHANNEL 🟦
TAR 🟩
SIZE/MSIZE 🟨
ADDR 🟦
DATA 🟪
SYNC ⬜
IDSEL 🟨
STOP 🟪

Decode Range
From Buffer Head
To Buffer Tail

**LCLK:** Transfer clock of LPC.

**Data Edge:** Latch the data to be analyzed when the LCLK is rising or falling.

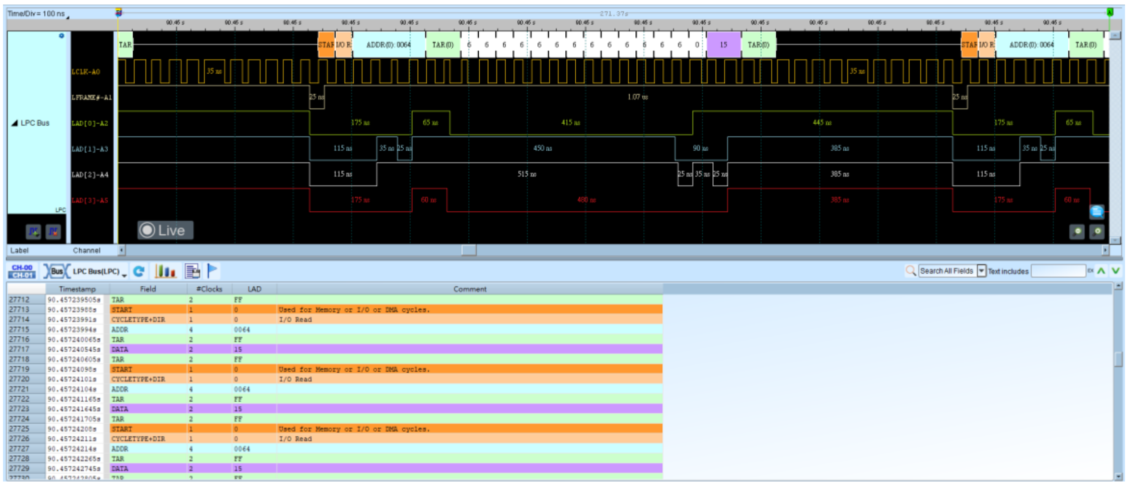
**LFRAME#:** Marks the start of each Frame transfer cycle or is used to interrupt

Frame transfer.

**LAD[0-3]:** The data bus is used to transfer commands, addresses, and data.

**Show the field in report:** To enable report filtering, simply check the items to be displayed in the report window.

**Result**

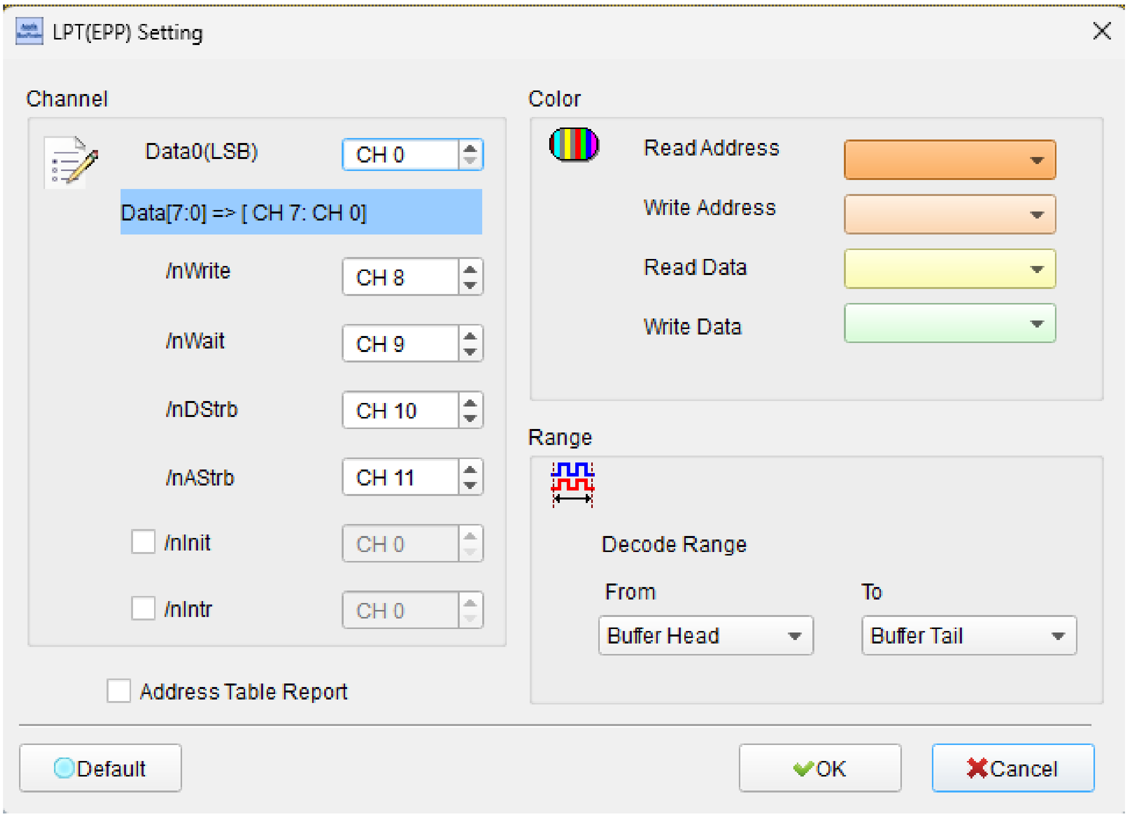


| | Timestamp | Field | #Clocks | LAD | Comment |
|---|---|---|---|---|---|
| 27712 | 90.457239505s | TAR | 2 | FF | |
| 27713 | 90.45723988s | START | 1 | 0 | Used for Memory or I/O or DMA cycles. |
| 27714 | 90.45723991s | CYCLETYPE+DIR | 1 | 0 | I/O Read |
| 27715 | 90.45723994s | ADDR | 4 | 0064 | |
| 27716 | 90.457240065s | TAR | 2 | FF | |
| 27717 | 90.457240545s | DATA | 2 | 15 | |
| 27718 | 90.457240605s | TAR | 2 | FF | |
| 27719 | 90.45724098s | START | 1 | 0 | Used for Memory or I/O or DMA cycles. |
| 27720 | 90.45724101s | CYCLETYPE+DIR | 1 | 0 | I/O Read |
| 27721 | 90.45724104s | ADDR | 4 | 0064 | |
| 27722 | 90.457241165s | TAR | 2 | FF | |
| 27723 | 90.457241645s | DATA | 2 | 15 | |
| 27724 | 90.457241705s | TAR | 2 | FF | |
| 27725 | 90.45724208s | START | 1 | 0 | Used for Memory or I/O or DMA cycles. |
| 27726 | 90.45724211s | CYCLETYPE+DIR | 1 | 0 | I/O Read |
| 27727 | 90.45724214s | ADDR | 4 | 0064 | |
| 27728 | 90.457242265s | TAR | 2 | FF | |
| 27729 | 90.457242745s | DATA | 2 | 15 | |
| 27730 | 90.457242805s | TAR | 2 | FF | |

### LPT

LPT is a universal parallel interface that use in PC since 1980's. It was primarily designed to operate a line printer, but could also be used to adapt other peripherals. This decode only support EPP Mode.

### Settings



LPT(EPP) Setting

Channel

Data0(LSB) CH 0

Data[7:0] => [ CH 7: CH 0]

/nWrite CH 8

/nWait CH 9

/nDStrb CH 10

/nAStrb CH 11

□ /nInit CH 0

□ /nIntr CH 0

□ Address Table Report

Color

Read Address

Write Address

Read Data

Write Data

Range

Decode Range

From Buffer Head

To Buffer Tail

**Data0(LSB):** There are 8 data channel. Only set Data0(LSB) here, other channel will be set automatically.

**/nWrite:** Indicates the direction of transfer.

**/nWait:** To acknowledge that a transfer has finished.

**/nDStrb:** Indicates the data cycle.

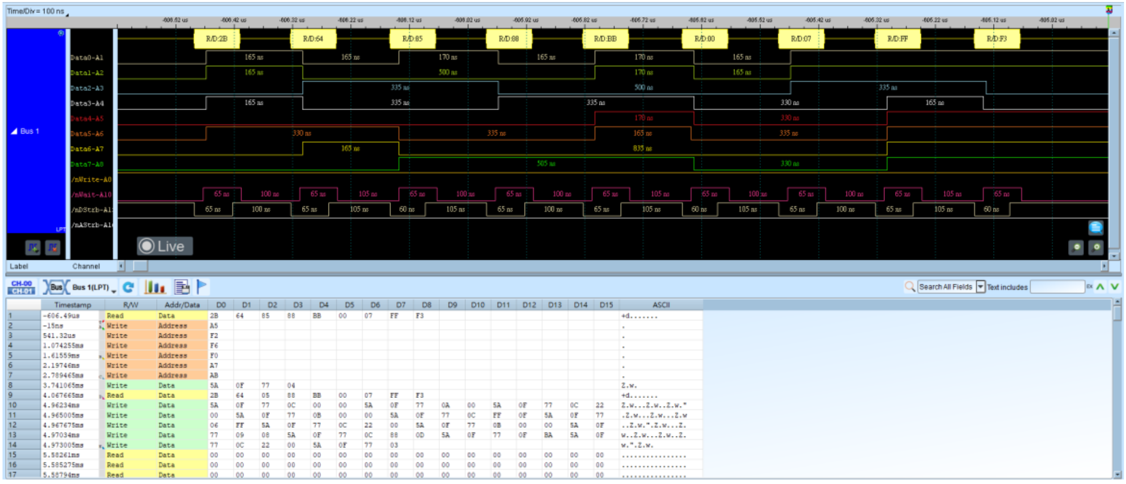
**/nAStrb:** Indicates the address cycle.

**/nInit:** Indicates a termination cycle in order to return the interface to the

Compatibility mode. User can option to use this channel or not.

**/nIntr:** This is an interrupt signal. User can option to use this channel or not.

### Result

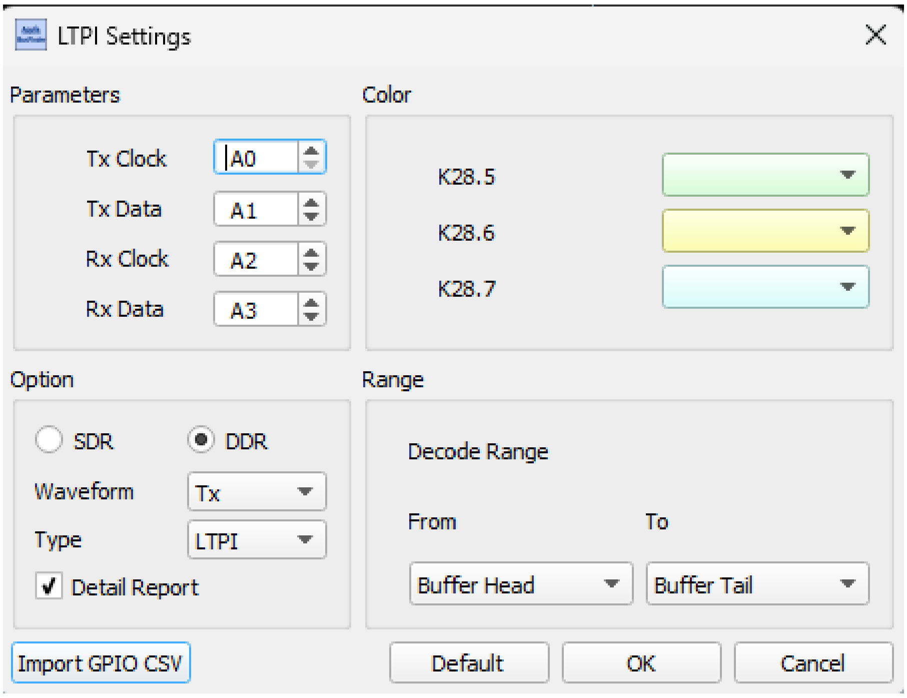


| | Timestamp | R/W | Addr/Data | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -606.49us | Read | Data | 2B | 64 | 85 | 88 | BB | 00 | 07 | FF | F3 | | | | | | | | +d....... |
| 2 | -15ns | Write | Address | A5 | | | | | | | | | | | | | | | | . |
| 3 | 541.32us | Write | Address | F2 | | | | | | | | | | | | | | | | . |
| 4 | 1.074255ms | Write | Address | F6 | | | | | | | | | | | | | | | | . |
| 5 | 1.61559ms | Write | Address | F0 | | | | | | | | | | | | | | | | . |
| 6 | 2.19746ms | Write | Address | A7 | | | | | | | | | | | | | | | | . |
| 7 | 2.789465ms | Write | Address | AB | | | | | | | | | | | | | | | | . |
| 8 | 3.741065ms | Write | Data | 5A | 0F | 77 | 04 | | | | | | | | | | | | | Z.w. |
| 9 | 4.067665ms | Read | Data | 2B | 64 | 05 | 88 | BB | 00 | 07 | FF | F3 | | | | | | | | +d....... |
| 10 | 4.96234ms | Write | Data | 5A | 0F | 77 | 0C | 00 | 00 | 5A | 0F | 77 | 0A | 00 | 5A | 0F | 77 | 0C | 22 | Z.w...Z.w..Z.w." |
| 11 | 4.965005ms | Write | Data | 00 | 5A | 0F | 77 | 0B | 00 | 00 | 5A | 0F | 77 | 0C | FF | 0F | 5A | 0F | 77 | .Z.w...Z.w...Z.w |
| 12 | 4.967675ms | Write | Data | 06 | FF | 5A | 0F | 77 | 0C | 22 | 00 | 5A | 0F | 77 | 0B | 00 | 00 | 5A | 0F | ..Z.w.".Z.w...Z. |
| 13 | 4.97034ms | Write | Data | 77 | 09 | 08 | 5A | 0F | 77 | 0C | 88 | 0D | 5A | 0F | 77 | 0F | BA | 5A | 0F | w..Z.w...Z.w..Z. |
| 14 | 4.973005ms | Write | Data | 77 | 0C | 22 | 00 | 5A | 0F | 77 | 03 | | | | | | | | | w.".Z.w. |
| 15 | 5.58261ms | Read | Data | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ................ |
| 16 | 5.585275ms | Read | Data | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ................ |
| 17 | 5.58794ms | Read | Data | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ................ |

## LTPI

LTPI (LVDS Tunneling Protocol & Interface) is an interface protocol designed for transmitting low-speed signals between the HPM (Host Processor Module) and SCM (Server Control Module). It supports multiple channels such as GPIO, I2C/SMBus, and UART, utilizing time-division multiplexing (TDM) over LVDS links to achieve multi-channel transmission. The GPIO channel includes both low-latency and normal-latency types. The I2C/SMBus channel handles single-controller communication links. The UART channel supports flow control. Additionally, an OEM channel is provided for proprietary interfaces, and a data channel supports memory image data transfers

### Parameters



LTPI Settings

Parameters

Tx Clock: A0
Tx Data: A1
Rx Clock: A2
Rx Data: A3

Color

K28.5: 🟩
K28.6: 🟨
K28.7: 🟦

Option

○ SDR ● DDR
Waveform: Tx
Type: LTPI
☑ Detail Report

Range

Decode Range
From: Buffer Head
To: Buffer Tail

**Tx Clock:** Transfer Clock of Tx.

**Tx Data:** Transfer Data of Tx

**Rx Clock:** Transfer Clock of Rx

**Rx Data:** Transfer Data of Tx

**SDR:** The transmission method is SDR。

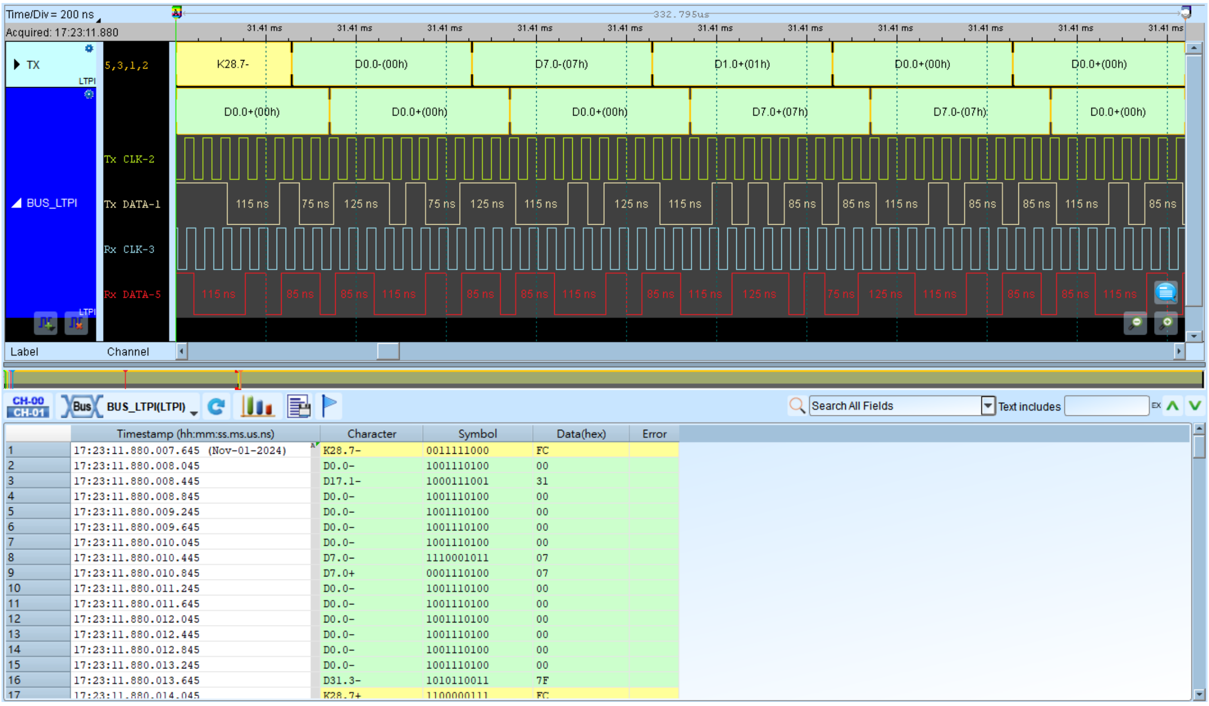
**DDR:** The transmission method is DDR。

**Waveform:** The direction (Transmit / Receive) and decoded content displayed in the waveform area (8b10b / LTPI / GPIO / UART / I2C)

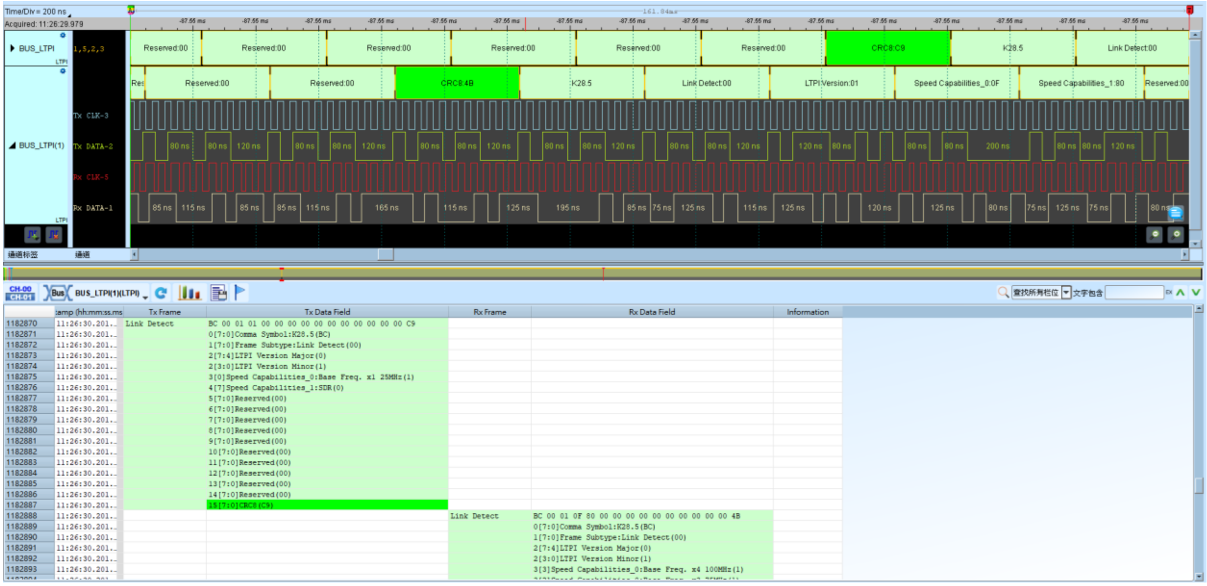
**Detail Report:** Whether to enable detailed reporting

**Import GPIO CSV:** Whether to import the GPIO bit name

## Result



| | Timestamp (hh:mm:ss.ms.us.ns) | Character | Symbol | Data(hex) | Error |
|---|---|---|---|---|---|
| 1 | 17:23:11.880.007.645 (Nov-01-2024) | K28.7- | 0011111000 | FC | |
| 2 | 17:23:11.880.008.045 | D0.0- | 1001110100 | 00 | |
| 3 | 17:23:11.880.008.445 | D17.1- | 1000111001 | 31 | |
| 4 | 17:23:11.880.008.845 | D0.0- | 1001110100 | 00 | |
| 5 | 17:23:11.880.009.245 | D0.0- | 1001110100 | 00 | |
| 6 | 17:23:11.880.009.645 | D0.0- | 1001110100 | 00 | |
| 7 | 17:23:11.880.010.045 | D0.0- | 1001110100 | 00 | |
| 8 | 17:23:11.880.010.445 | D7.0- | 1110001011 | 07 | |
| 9 | 17:23:11.880.010.845 | D7.0+ | 0001110100 | 07 | |
| 10 | 17:23:11.880.011.245 | D0.0- | 1001110100 | 00 | |
| 11 | 17:23:11.880.011.645 | D0.0- | 1001110100 | 00 | |
| 12 | 17:23:11.880.012.045 | D0.0- | 1001110100 | 00 | |
| 13 | 17:23:11.880.012.445 | D0.0- | 1001110100 | 00 | |
| 14 | 17:23:11.880.012.845 | D0.0- | 1001110100 | 00 | |
| 15 | 17:23:11.880.013.245 | D0.0- | 1001110100 | 00 | |
| 16 | 17:23:11.880.013.645 | D31.3- | 1010110011 | 7F | |
| 17 | 17:23:11.880.014.045 | K28.7+ | 1100000111 | FC | |

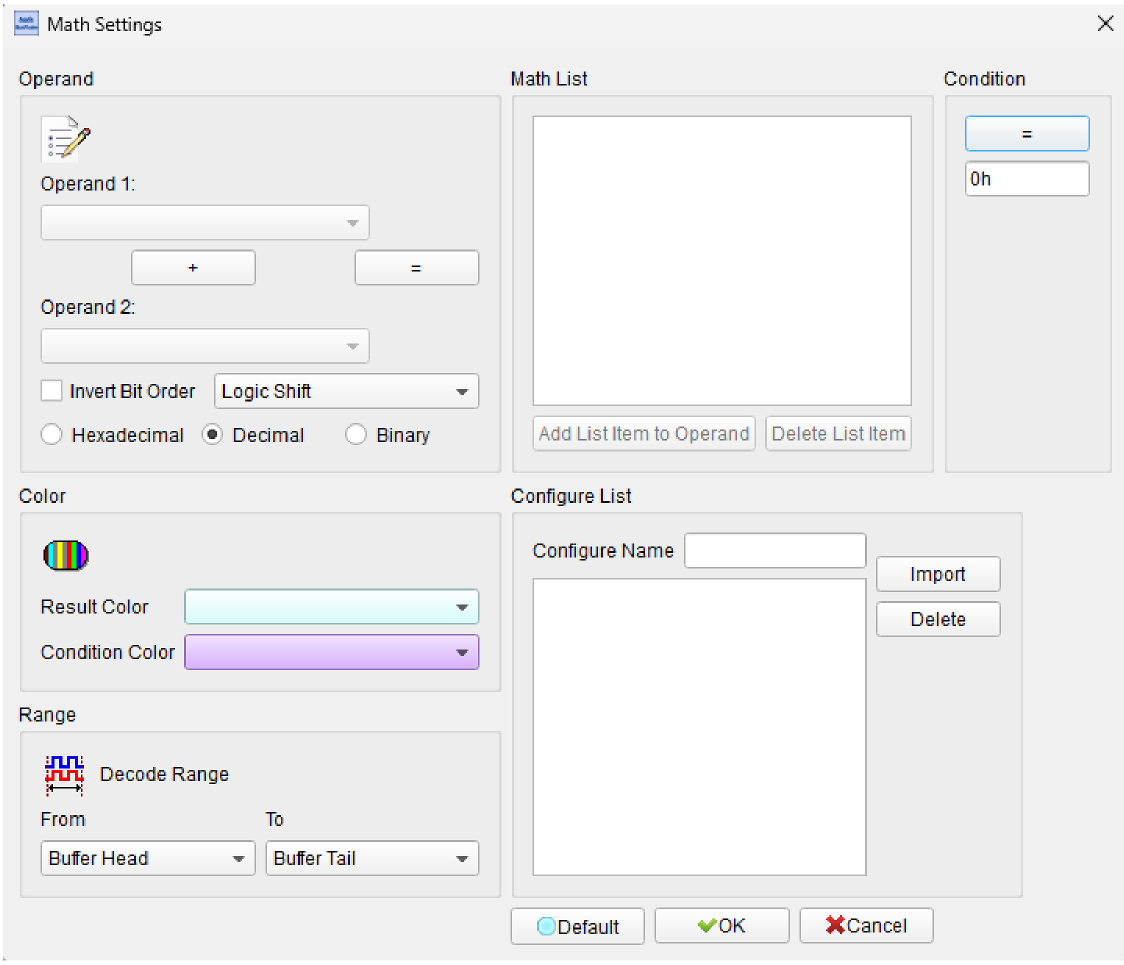


| | :amp (hh:mm:ss.ms | Tx Frame | Tx Data Field | Rx Frame | Rx Data Field | Information |
|---|---|---|---|---|---|---|
| 1182870 | 11:26:30.201... | Link Detect | BC 00 01 01 00 00 00 00 00 00 00 00 00 00 00 C9 | | | |
| 1182871 | 11:26:30.201... | | 0[7:0]Comma Symbol:K28.5(BC) | | | |
| 1182872 | 11:26:30.201... | | 1[7:0]Frame Subtype:Link Detect(00) | | | |
| 1182873 | 11:26:30.201... | | 2[7:4]LTPI Version Major(0) | | | |
| 1182874 | 11:26:30.201... | | 2[3:0]LTPI Version Minor(1) | | | |
| 1182875 | 11:26:30.201... | | 3[0]Speed Capabilities_0:Base Freq. x1 25MHz(1) | | | |
| 1182876 | 11:26:30.201... | | 4[7]Speed Capabilities_1:SDR(0) | | | |
| 1182877 | 11:26:30.201... | | 5[7:0]Reserved(00) | | | |
| 1182878 | 11:26:30.201... | | 6[7:0]Reserved(00) | | | |
| 1182879 | 11:26:30.201... | | 7[7:0]Reserved(00) | | | |
| 1182880 | 11:26:30.201... | | 8[7:0]Reserved(00) | | | |
| 1182881 | 11:26:30.201... | | 9[7:0]Reserved(00) | | | |
| 1182882 | 11:26:30.201... | | 10[7:0]Reserved(00) | | | |
| 1182883 | 11:26:30.201... | | 11[7:0]Reserved(00) | | | |
| 1182884 | 11:26:30.201... | | 12[7:0]Reserved(00) | | | |
| 1182885 | 11:26:30.201... | | 13[7:0]Reserved(00) | | | |
| 1182886 | 11:26:30.201... | | 14[7:0]Reserved(00) | | | |
| 1182887 | 11:26:30.201... | | 15[7:0]CRC8(C9) | | | |
| 1182888 | 11:26:30.201... | | | Link Detect | BC 00 01 0F 80 00 00 00 00 00 00 00 00 00 00 4B | |
| 1182889 | 11:26:30.201... | | | | 0[7:0]Comma Symbol:K28.5(BC) | |
| 1182890 | 11:26:30.201... | | | | 1[7:0]Frame Subtype:Link Detect(00) | |
| 1182891 | 11:26:30.201... | | | | 2[7:4]LTPI Version Major(0) | |
| 1182892 | 11:26:30.201... | | | | 2[3:0]LTPI Version Minor(1) | |
| 1182893 | 11:26:30.201... | | | | 3[3]Speed Capabilities_0:Base Freq. x4 100MHz(1) | |

## Math

Math is used to conduct addition, subtraction, multiplication, division, AND, XOR, OR,NAND, NOR, XNOR, Bit Shift operation for the combined value of channel or bus.

## Parameters Settings



Math Settings

☐ Invert Bit Order  Logic Shift

○ Hexadecimal  ● Decimal  ○ Binary

Condition: =  0h

Decode Range  From: Buffer Head  To: Buffer Tail

### 1. Channels Settings

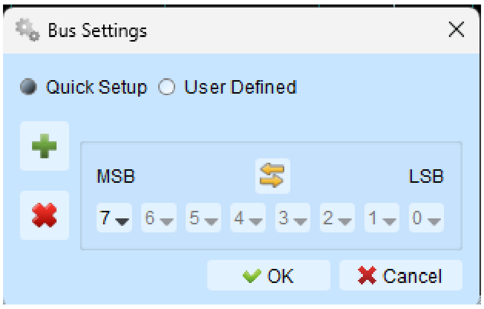
i. **Operand: The channel or bus for which the calculation is to be performed is automatically listed with the labeled name of the channel or bus in the current waveform window.**

ii. Operator: We support

1. Arithmetic operator: + 、 - 、 × 、 /
2. Logic operator: AND、XOR、OR、NAND、NOR、XNOR、>>、<<

iii. 「=」button: Add the equation to the Math List

iv. Invert Bit Order: By default, when adding a new bus, the smaller channel number is lsb and the larger channel number is msb. Users can reverse the order of lsb and msb by setting this option.



v. Bit Shift Method: We support (i)Arithmetic Shift、(ii)Logic Shift、(iii) Rotate Circular Shift and (iv)Rotate through Carry Shift. Use with >> and << in logical operators.

vi. Numeric Display: We support Hexadecimal、Decimal、Binary.

vii. Math List: Displaying the equations added by the user, up to a limit of 8.

viii. 「Add List Item to Operand」button: Adding the selected item from *Math List* to operand for user to select for further operating with other channel or bus. Those item that was added to operand is enclosed in a set of parentheses when it operates with other operand. Up 2 levels of parentheses are supported.

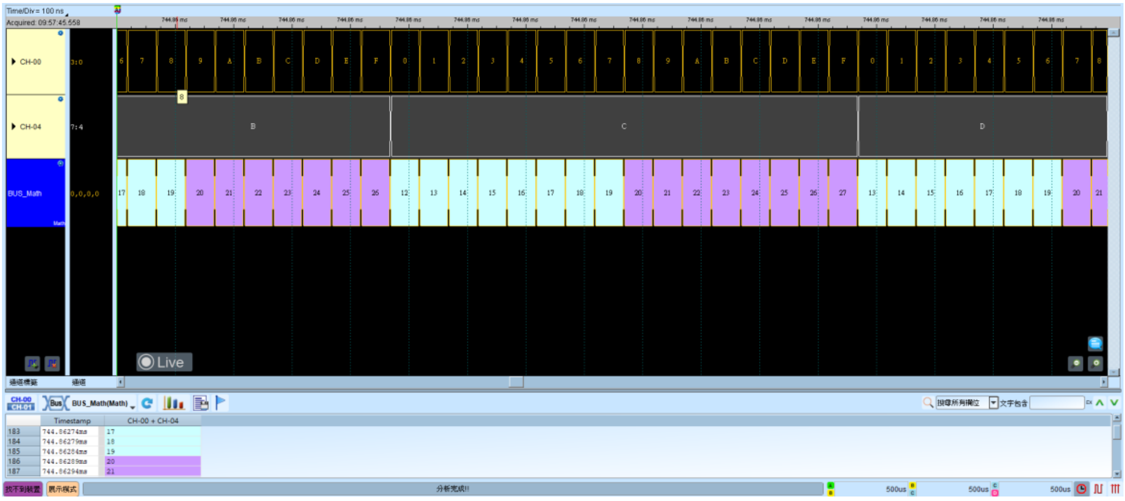
ix. 「Delete List Item」 button: Deleting the selected item from *Math List*.

2. **Waveform Settings:**
   - i. **Setting the color of frame that represents the calculated result.**
   - ii. **Setting the comparison condition  and the frame color that matches the condition.**
3. **Case Settings:**
   - i. **Comparison Condition: We support >= 、 > 、 = 、 <= 、 <。**
   - ii. **Comparison Value: Entering the value of the condition to be compared. Supports hexadecimal and decimal representation.**
4. **Configure: Lists all the configure names so that users can quickly switch between different configures. The configure records a list of different equations.**

**Operating Methods**

1. **After selecting the operand and operator, click "=" to add the equation to the *Math List*.**
2. **Choosing the equations to be operated on.**
3. **Setting the color of frame that represents the calculated result.**
4. **Setting the comparison condition and entering the comparison value, also setting the color for displaying the matches frame.**

## Result



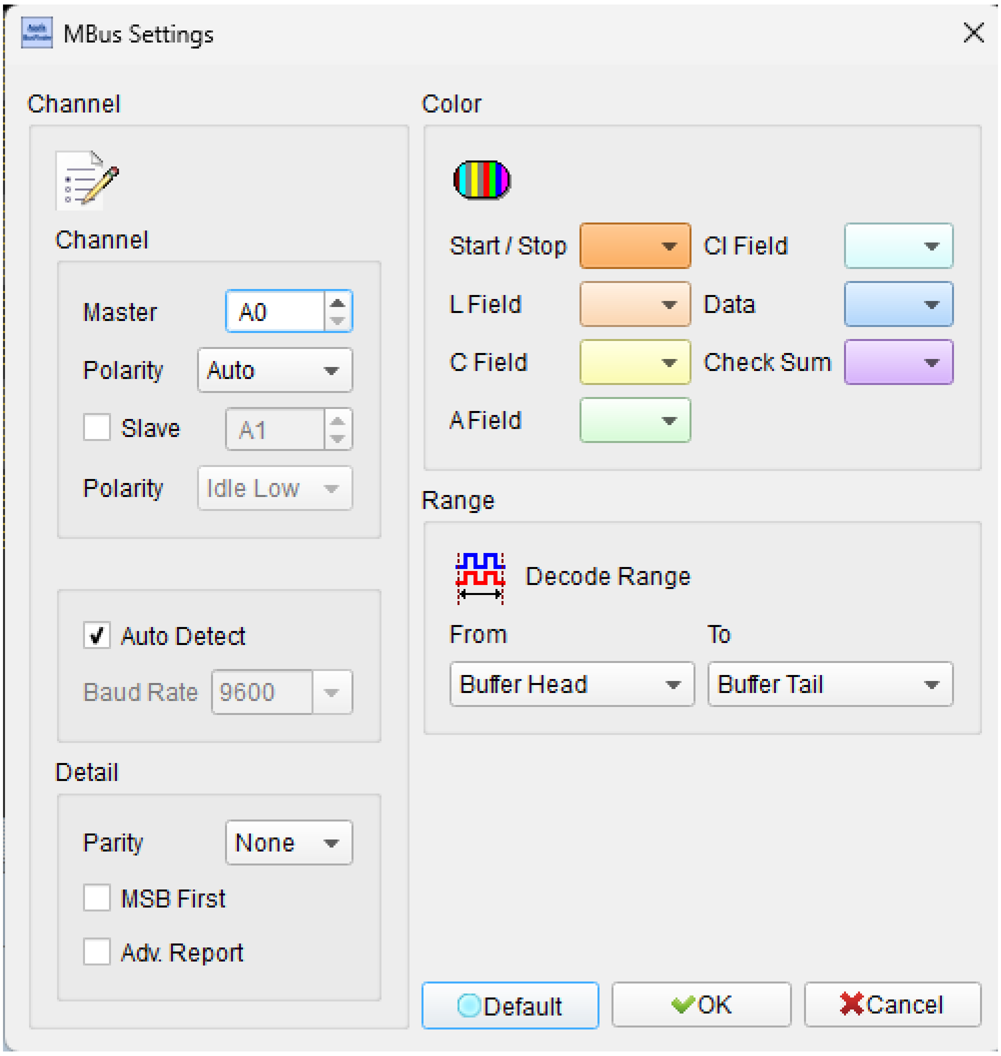
|  | Timestamp | CH-00 + CH-04 |
|---|---|---|
| 183 | 744.86274ms | 17 |
| 184 | 744.86279ms | 18 |
| 185 | 744.86284ms | 19 |
| 186 | 744.86289ms | 20 |
| 187 | 744.86294ms | 21 |

**Note:** After setting, press OK to write all the settings to a file and save it to the working directory (AqMath.bin). The file will be overwritten every time user press OK, so when user save the file, user need to save a copy of AqMath.bin in addition to the waveform file. When user open the waveform file, user need to place AqMath.bin in the working directory before opening the waveform file.

## M-Bus

M-Bus (Meter-Bus) is for remote reading of heat meters and other types of consumption meters.

## Settings



MBus Settings

Channel

Channel

Master A0

Polarity Auto

□ Slave A1

Polarity Idle Low

☑ Auto Detect

Baud Rate 9600

Detail

Parity None

□ MSB First

□ Adv. Report

Color

Start / Stop 🟧

CI Field

L Field

Data

C Field

Check Sum

A Field

Range

Decode Range

From Buffer Head

To Buffer Tail

**Channel:** Set the signal channel and polarity. If there is a Slave on the bus, user can set up additional Slave channels. Enable when checked.

**Baud Rate:** The transmission speed of the signal. Checking Auto Detection will detect the baud rate by itself.

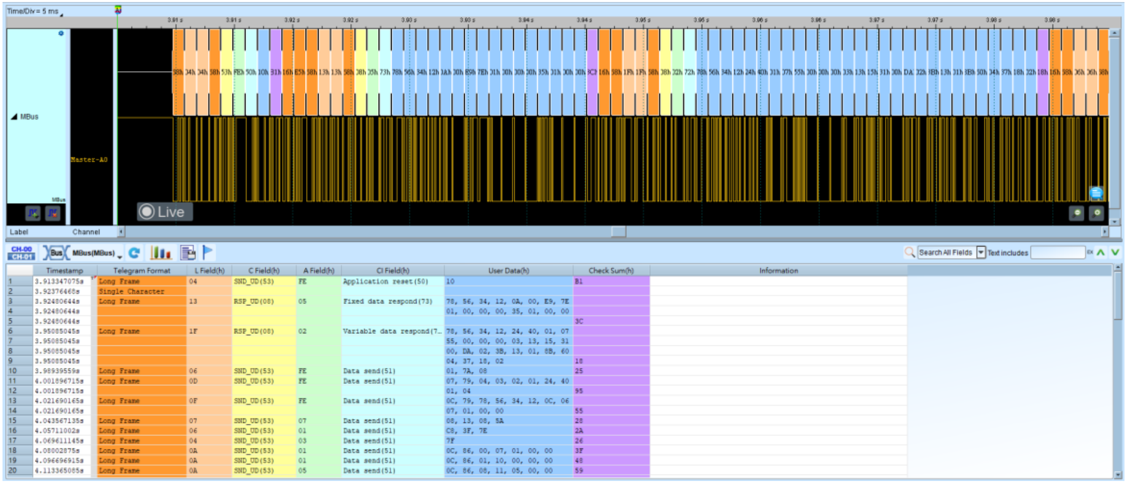
**Detail:**

I. **Parity:** Parity error detect.

II. **MSB First:** Displayed in MSB first format. Enabled when checked.

III. **Adv. Report:** Advanced report. Enabled when checked.

## Result



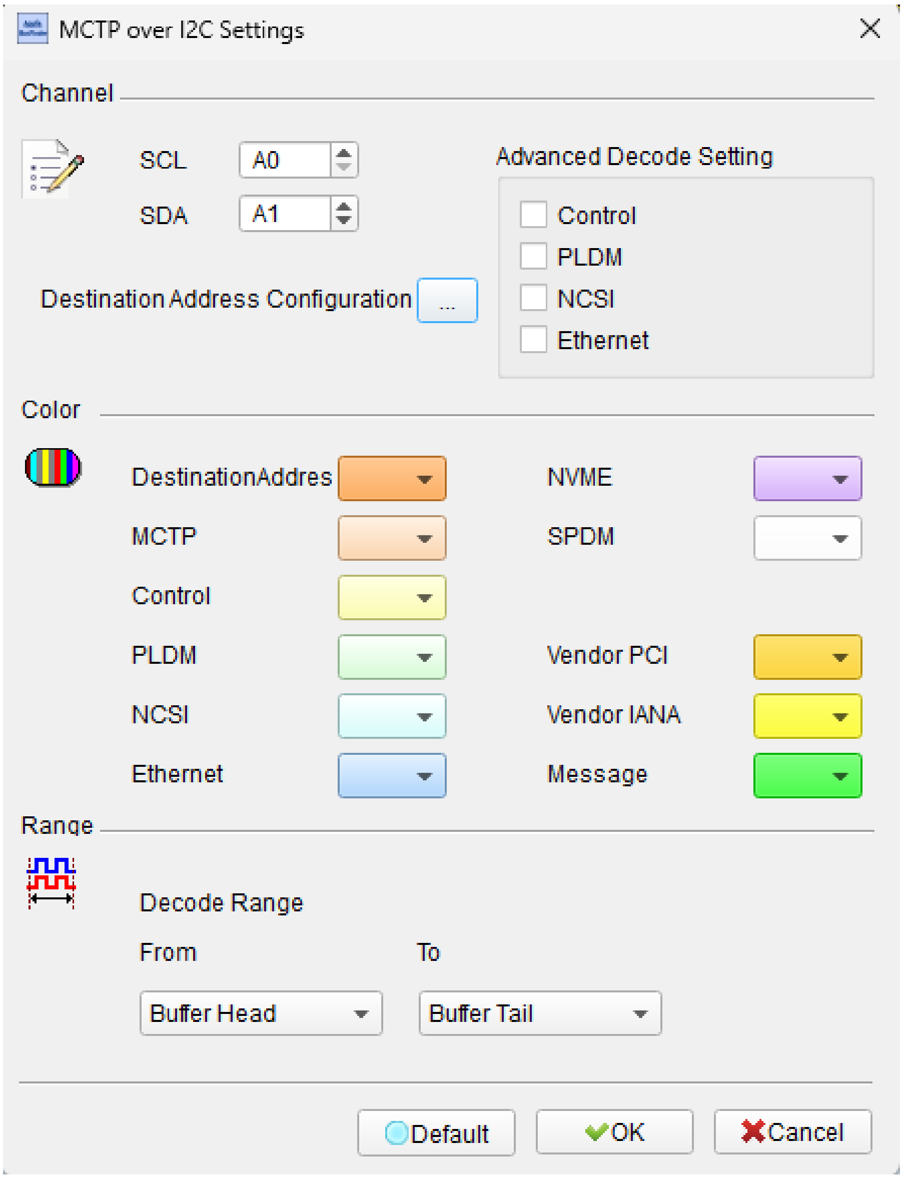
| | Timestamp | Telegram Format | L Field(h) | C Field(h) | A Field(h) | CI Field(h) | User Data(h) | Check Sum(h) | Information |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 3.913347075s | Long Frame | 04 | SND_UD(53) | FE | Application reset(50) | 10 | B1 | |
| 2 | 3.92376468s | Single Character | | | | | | | |
| 3 | 3.92480644s | Long Frame | 13 | RSP_UD(08) | 05 | Fixed data respond(73) | 78, 56, 34, 12, 0A, 00, E9, 7E | | |
| 4 | 3.92480644s | | | | | | 01, 00, 00, 00, 35, 01, 00, 00 | | |
| 5 | 3.92480644s | | | | | | | 3C | |
| 6 | 3.95085045s | Long Frame | 1F | RSP_UD(08) | 02 | Variable data respond(7… | 78, 56, 34, 12, 24, 40, 01, 07 | | |
| 7 | 3.95085045s | | | | | | 55, 00, 00, 00, 03, 13, 15, 31 | | |
| 8 | 3.95085045s | | | | | | 00, DA, 02, 3B, 13, 01, 8B, 60 | | |
| 9 | 3.95085045s | | | | | | 04, 37, 18, 02 | 18 | |
| 10 | 3.98939559s | Long Frame | 06 | SND_UD(53) | FE | Data send(51) | 01, 7A, 08 | 25 | |
| 11 | 4.001896715s | Long Frame | 0D | SND_UD(53) | FE | Data send(51) | 07, 79, 04, 03, 02, 01, 24, 40 | | |
| 12 | 4.001896715s | | | | | | 01, 04 | 95 | |
| 13 | 4.021690165s | Long Frame | 0F | SND_UD(53) | FE | Data send(51) | 0C, 79, 78, 56, 34, 12, 0C, 06 | | |
| 14 | 4.021690165s | | | | | | 07, 01, 00, 00 | 55 | |
| 15 | 4.043567135s | Long Frame | 07 | SND_UD(53) | 07 | Data send(51) | 08, 13, 08, 5A | 28 | |
| 16 | 4.05711002s | Long Frame | 06 | SND_UD(53) | 01 | Data send(51) | C8, 3F, 7E | 2A | |
| 17 | 4.069611145s | Long Frame | 04 | SND_UD(53) | 03 | Data send(51) | 7F | 26 | |
| 18 | 4.08002875s | Long Frame | 0A | SND_UD(53) | 01 | Data send(51) | 0C, 86, 00, 07, 01, 00, 00 | 3F | |
| 19 | 4.096696915s | Long Frame | 0A | SND_UD(53) | 01 | Data send(51) | 0C, 86, 01, 10, 00, 00, 00 | 48 | |
| 20 | 4.113365085s | Long Frame | 0A | SND_UD(53) | 05 | Data send(51) | 0C, 86, 08, 11, 05, 00, 00 | 59 | |

### MCTP over I²C

MCTP (Management Component Transport Protocol) over I²C is a transport binding that enables communication between management controllers and other components in a system using I²C as the physical layer.

MCTP is a transport-independent protocol, meaning it can work over various underlying buses like PCIe, SMBus/I²C, UART, and Ethernet. When MCTP operates over I²C/SMBus, it follows the I²C/SMBus protocol for message passing between MCTP endpoints.

**Settings:**



MCTP over I2C Settings

Channel

SCL: A0

SDA: A1

Destination Address Configuration ...

Advanced Decode Setting

□ Control
□ PLDM
□ NCSI
□ Ethernet

Color

| | | | |
|---|---|---|---|
| DestinationAddres | ▼ | NVME | ▼ |
| MCTP | ▼ | SPDM | ▼ |
| Control | ▼ | | |
| PLDM | ▼ | Vendor PCI | ▼ |
| NCSI | ▼ | Vendor IANA | ▼ |
| Ethernet | ▼ | Message | ▼ |

Range

Decode Range

From: Buffer Head

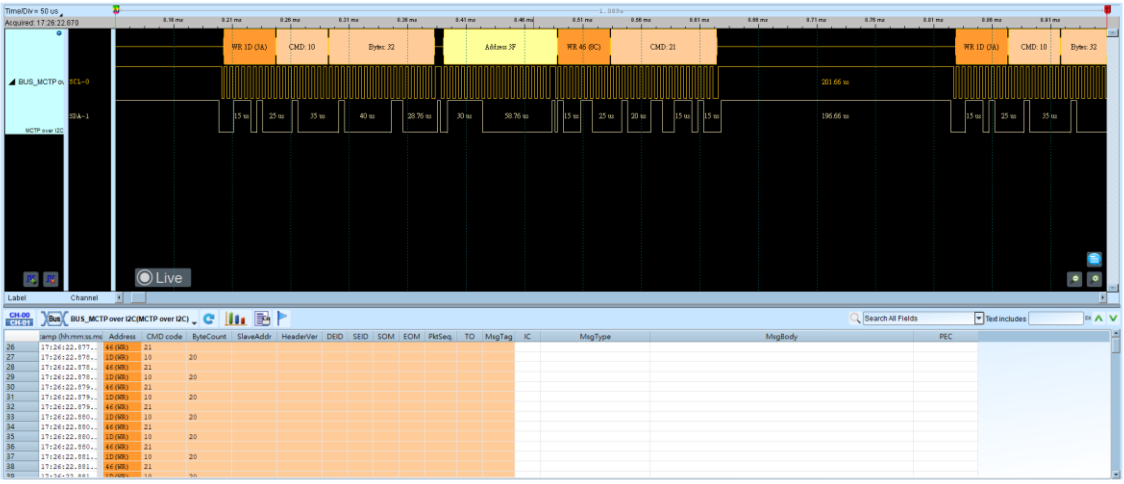
To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Destination Address Configuration:** Set the device address and its corresponding protocol.

**Advanced Decode Setting:** Show the detail meaning of the raw data.

**Result:**



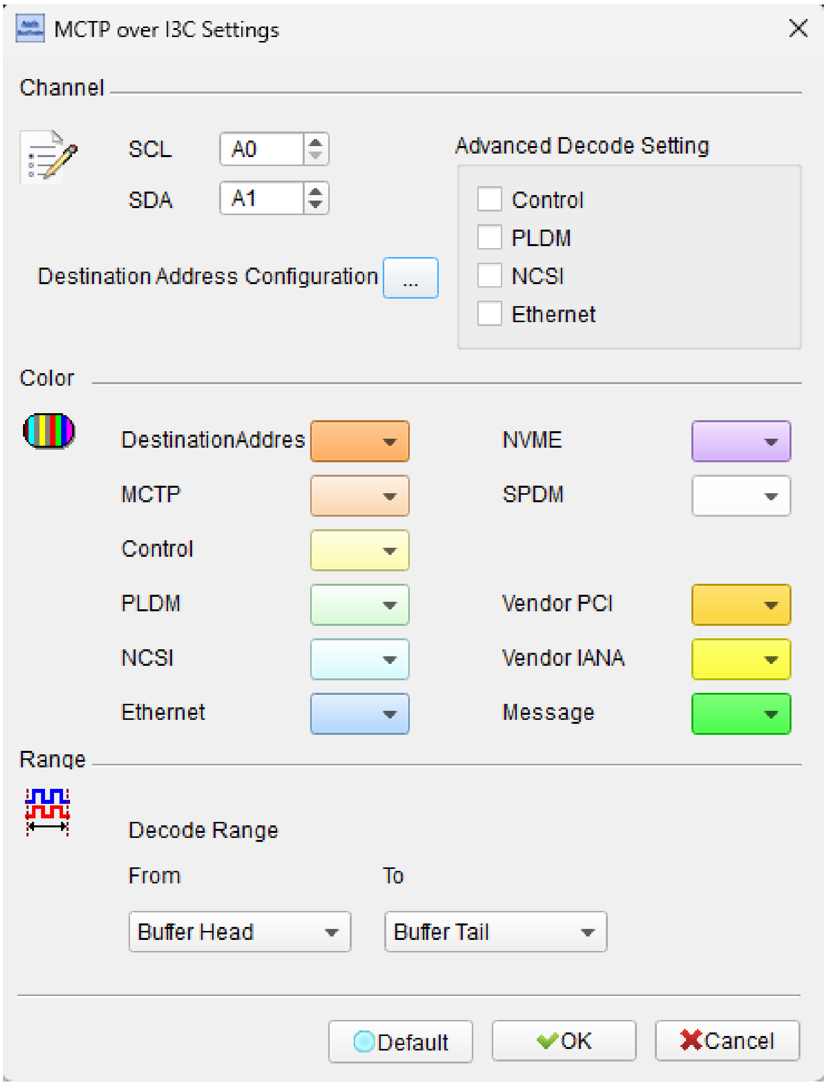
| | tamp (hh:mm:ss.ms | Address | CMD code | ByteCount | SlaveAddr | HeaderVer | DEID | SEID | SOM | EOM | PktSeq. | TO | MsgTag | IC | MsgType | MsgBody | PEC |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 26 | 17:26:22.877... | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 27 | 17:26:22.878... | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 28 | 17:26:22.878... | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 29 | 17:26:22.878... | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 30 | 17:26:22.879... | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 31 | 17:26:22.879... | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 32 | 17:26:22.879... | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 33 | 17:26:22.880... | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 34 | 17:26:22.880... | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 35 | 17:26:22.880... | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 36 | 17:26:22.880... | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 37 | 17:26:22.881... | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 38 | 17:26:22.881... | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 39 | 17:26:22.881... | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |

## MCTP over I3C

MCTP (Management Component Transport Protocol) over I3C is a transport binding that enables standardized communication between system management controllers and peripherals using the MIPI I3C (Improved Inter-Integrated Circuit) bus.

MCTP is a transport-agnostic protocol, meaning it can operate over various physical layers like PCIe, SMBus/I²C, UART, and Ethernet. When using I3C, MCTP benefits from its higher speed, dynamic addressing, and in-band interrupt support, making it a more efficient alternative to I²C/SMBus.

### Settings:



MCTP over I3C Settings

Channel

SCL A0

SDA A1

Destination Address Configuration ...

Advanced Decode Setting

□ Control

□ PLDM

□ NCSI

□ Ethernet

Color

| | | | |
|---|---|---|---|
| DestinationAddres | 🟧 | NVME | 🟪 |
| MCTP | 🟧 | SPDM | ⬜ |
| Control | 🟨 | | |
| PLDM | 🟩 | Vendor PCI | 🟨 |
| NCSI | 🟦 | Vendor IANA | 🟨 |
| Ethernet | 🟦 | Message | 🟩 |

Range

Decode Range

From Buffer Head

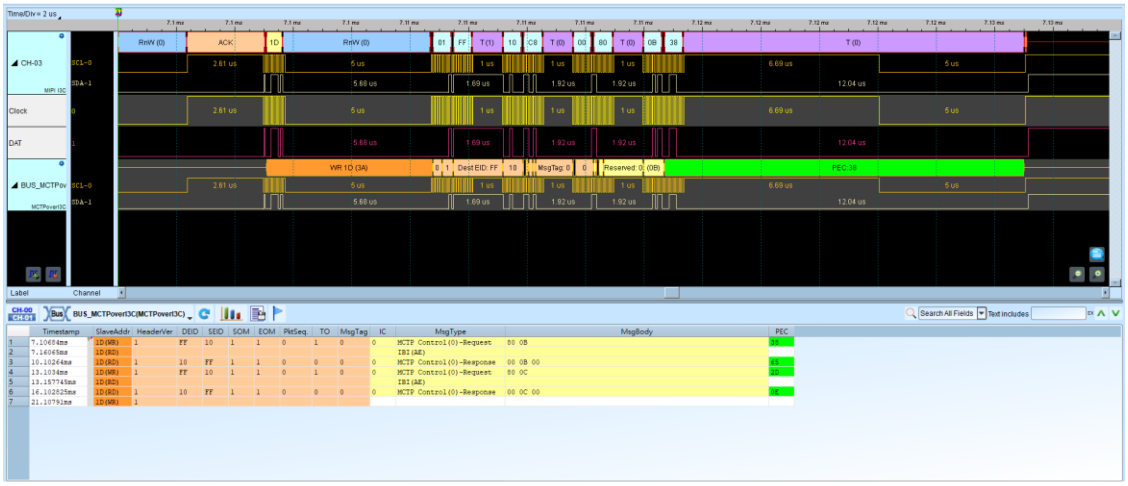
To Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Destination Address Configuration:** Set the device address and its corresponding protocol.

**Advanced Decode Setting:** Show the detail meaning of the raw rata.

**Result:**



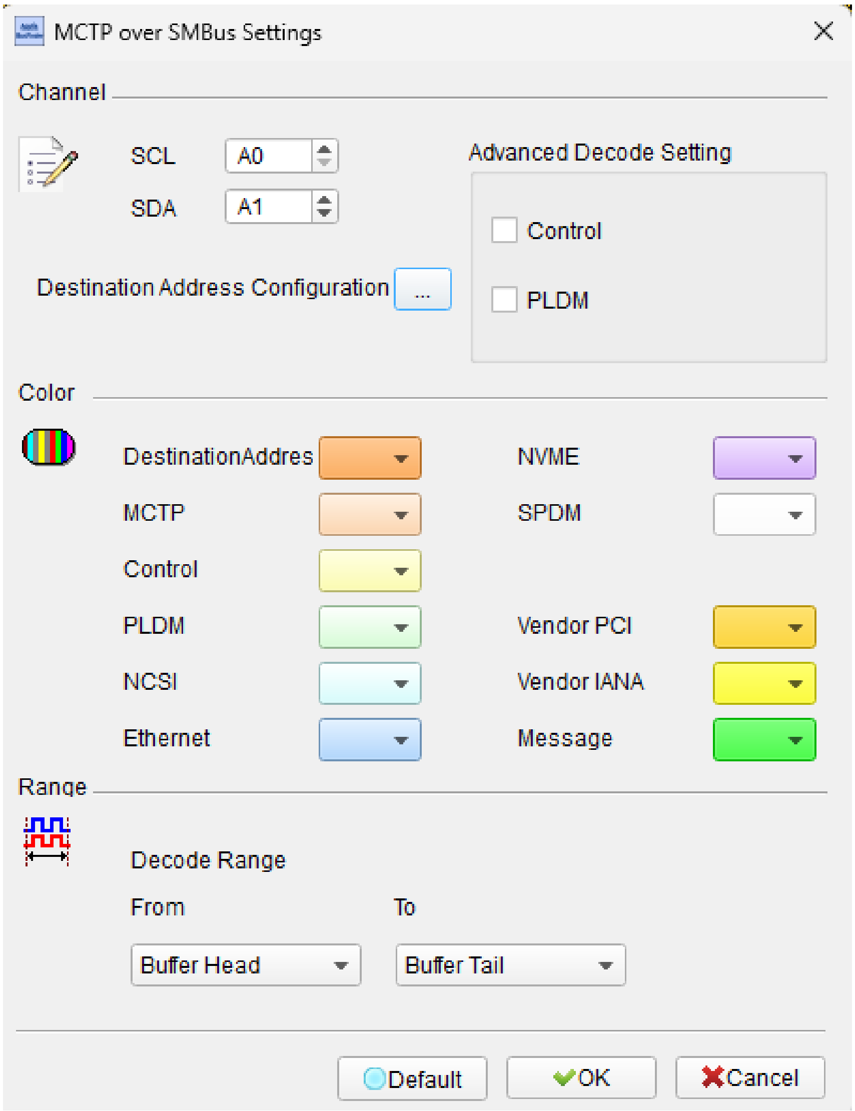
| | Timestamp | SlaveAddr | HeaderVer | DEID | SEID | SOM | EOM | PktSeq. | TO | MsgTag | IC | MsgType | MsgBody | PEC |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 7.10684ms | 1D(WR) | 1 | FF | 10 | 1 | 1 | 0 | 1 | 0 | 0 | MCTP Control(0)-Request | 80 0B | 38 |
| 2 | 7.16065ms | 1D(RD) | | | | | | | | | | IBI(AE) | | |
| 3 | 10.10264ms | 1D(RD) | 1 | 10 | FF | 1 | 1 | 0 | 0 | 0 | 0 | MCTP Control(0)-Response | 00 0B 00 | 65 |
| 4 | 13.1034ms | 1D(WR) | 1 | FF | 10 | 1 | 1 | 0 | 1 | 0 | 0 | MCTP Control(0)-Request | 80 0C | 2D |
| 5 | 13.157745ms | 1D(RD) | | | | | | | | | | IBI(AE) | | |
| 6 | 16.102825ms | 1D(RD) | 1 | 10 | FF | 1 | 1 | 0 | 0 | 0 | 0 | MCTP Control(0)-Response | 00 0C 00 | 0E |
| 7 | 21.10791ms | 1D(WR) | 1 | | | | | | | | | | | |

### MCTP over SMBus

MCTP (Management Component Transport Protocol) over SMBus is a transport binding that enables communication between management controllers and other components in a system using SMBus as the physical layer.

MCTP is a transport-independent protocol, meaning it can work over various underlying buses like PCIe, SMBus/I²C, UART, and Ethernet. When MCTP operates over I²C/SMBus, it follows the I²C/SMBus protocol for message passing between MCTP endpoints.

**Settings:**



MCTP over SMBus Settings

Channel

SCL A0

SDA A1

Destination Address Configuration ...

Advanced Decode Setting

□ Control

□ PLDM

Color

| | | | |
|---|---|---|---|
| DestinationAddres | 🟧 | NVME | 🟪 |
| MCTP | 🟧 | SPDM | ⬜ |
| Control | 🟨 | | |
| PLDM | 🟩 | Vendor PCI | 🟨 |
| NCSI | 🟦 | Vendor IANA | 🟨 |
| Ethernet | 🟦 | Message | 🟩 |

Range

Decode Range

From Buffer Head

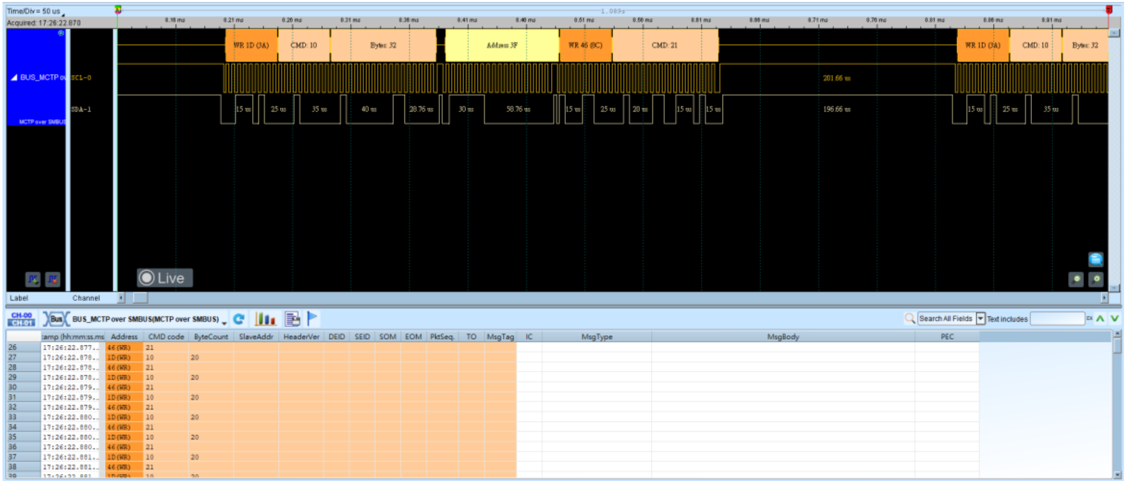
To Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Destination Address Configuration:** Set the device address and its corresponding protocol.

**Advanced Decode Setting:** Show the detail meaning of the raw rata.

**Result:**

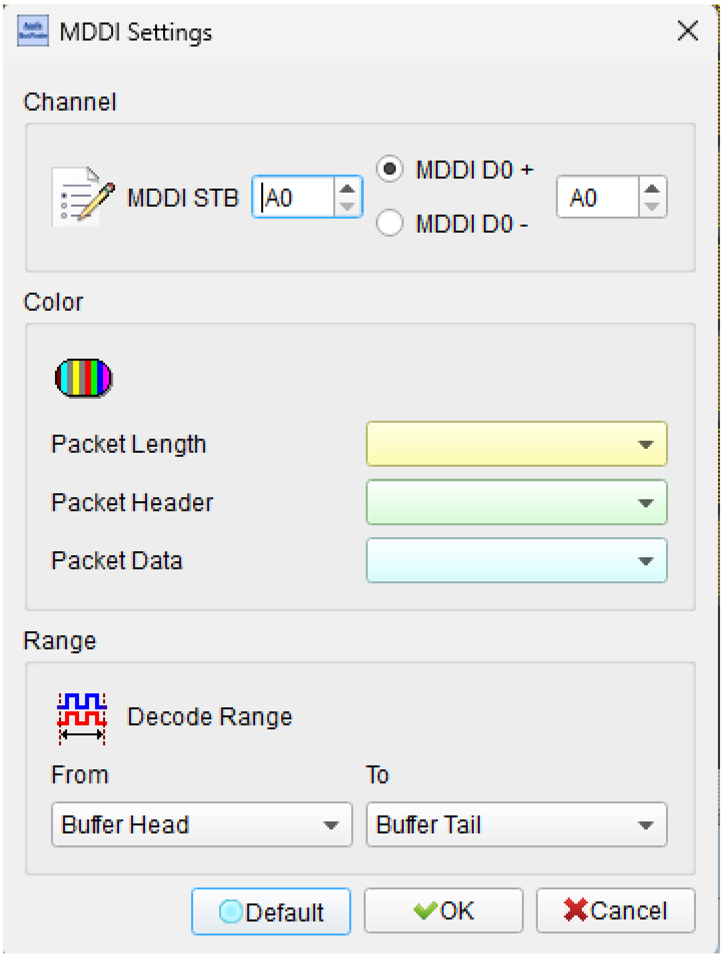


| | tamp (hh:mm:ss.ms | Address | CMD code | ByteCount | SlaveAddr | HeaderVer | DEID | SEID | SOM | EOM | PktSeq. | TO | MsgTag | IC | MsgType | MsgBody | PEC |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 26 | 17:26:22.877.. | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 27 | 17:26:22.878.. | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 28 | 17:26:22.878.. | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 29 | 17:26:22.878.. | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 30 | 17:26:22.879.. | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 31 | 17:26:22.879.. | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 32 | 17:26:22.879.. | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 33 | 17:26:22.880.. | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 34 | 17:26:22.880.. | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 35 | 17:26:22.880.. | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 36 | 17:26:22.880.. | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 37 | 17:26:22.881.. | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |
| 38 | 17:26:22.881.. | 46 (WR) | 21 | | | | | | | | | | | | | | |
| 39 | 17:26:22.881.. | 1D (WR) | 10 | 20 | | | | | | | | | | | | | |

### Mobile Display Digital Interface (MDDI)

The Mobile Display Digital Interface (MDDI) is a cost-effective low-power solution that enables high-speed short-range communication with a display device using a digital packet data link for connecting portable computing, communication, and entertainment devices to wearable micro displays. This decoder is based on VESA Mobile Display Digital Interface Standard Version 1.2, only Type I communication is supported in this decoder.

### Settings



MDDI Settings

Channel

MDDI STB: A0

● MDDI D0 +
○ MDDI D0 -
A0

Color

Packet Length
Packet Header
Packet Data

Range

Decode Range

From: Buffer Head
To: Buffer Tail

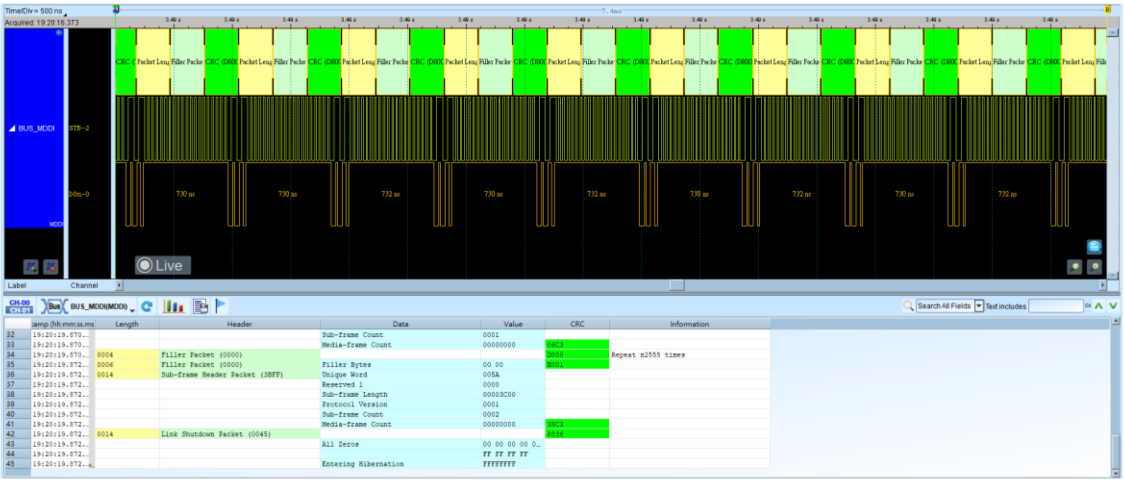
Channel Settings

MDDI STB: MDDI Strobe

MDDI D0+/-:  MDDI Data 0 +/-

Configure the Channel setup for the decoder, and choose the data source from Data 0+ or Data 0-.

Result

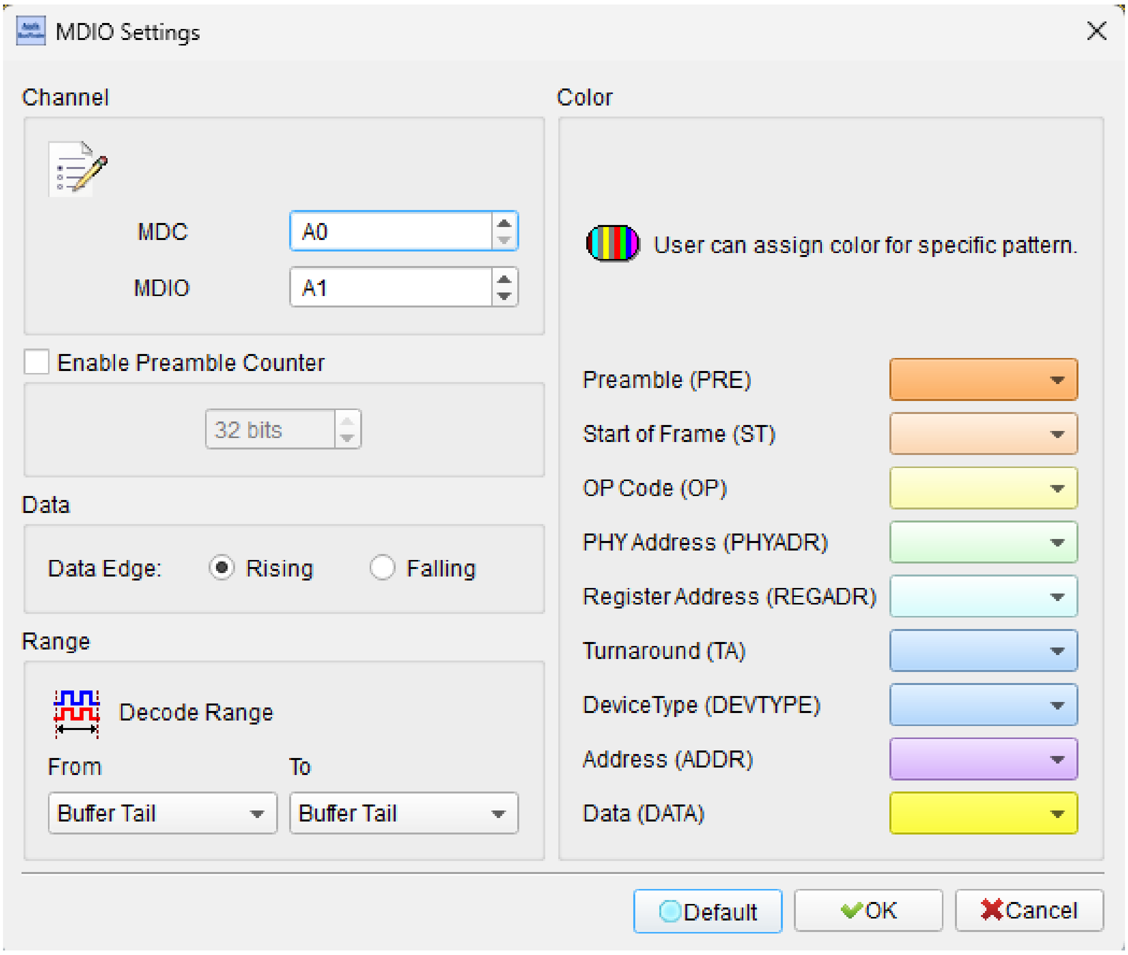


| | amp (hh:mm:ss.ms | Length | Header | Data | Value | CRC | Information |
|---|---|---|---|---|---|---|---|
| 32 | 19:20:19.870... | | | Sub-frame Count | 0001 | | |
| 33 | 19:20:19.870... | | | Media-frame Count | 00000000 | 06C3 | |
| 34 | 19:20:19.870... | 0004 | Filler Packet (0000) | | | D800 | Repeat x2555 times |
| 35 | 19:20:19.872... | 0006 | Filler Packet (0000) | Filler Bytes | 00 00 | B801 | |
| 36 | 19:20:19.872... | 0014 | Sub-frame Header Packet (3BFF) | Unique Word | 005A | | |
| 37 | 19:20:19.872... | | | Reserved 1 | 0000 | | |
| 38 | 19:20:19.872... | | | Sub-frame Length | 00003C00 | | |
| 39 | 19:20:19.872... | | | Protocol Version | 0001 | | |
| 40 | 19:20:19.872... | | | Sub-frame Count | 0002 | | |
| 41 | 19:20:19.872... | | | Media-frame Count | 00000000 | 35C3 | |
| 42 | 19:20:19.872... | 0014 | Link Shutdown Packet (0045) | | | 8836 | |
| 43 | 19:20:19.872... | | | All Zeros | 00 00 00 00 0... | | |
| 44 | 19:20:19.872... | | | | FF FF FF FF | | |
| 45 | 19:20:19.872... | | | Entering Hibernation | FFFFFFFF | | |

## MDIO

MDIO, also known as Serial Management Interface (SMI), is a serial bus defined for the Ethernet IEEE 802.3 specification for Media Independent Interface, or MII.

## Settings



MDIO Settings

Channel

MDC: A0

MDIO: A1

☐ Enable Preamble Counter

32 bits

Data

Data Edge: ● Rising ○ Falling

Range

Decode Range

From: Buffer Tail

To: Buffer Tail

Color

User can assign color for specific pattern.

| | |
|---|---|
| Preamble (PRE) | 🟧 |
| Start of Frame (ST) | 🟧 |
| OP Code (OP) | 🟨 |
| PHY Address (PHYADR) | 🟩 |
| Register Address (REGADR) | 🟦 |
| Turnaround (TA) | 🟦 |
| DeviceType (DEVTYPE) | 🟦 |
| Address (ADDR) | 🟪 |
| Data (DATA) | 🟨 |

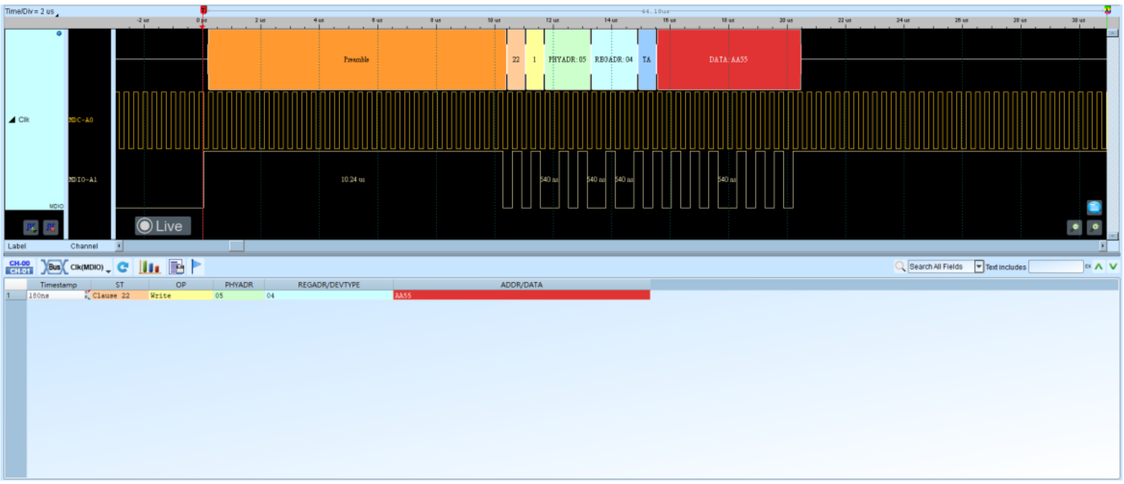
**MDC:** Transfer clock of MDIO.

**MDIO:** Transfer data input/output of MIDO.

**Data Edge:** Set the MDC Rising/Falling edge to latch the data field, Rising Edge default.

**Enable Preamble Counter:** Configurable MDIO Preamble width, 4 - 32 Bit, default is 32 Bit. Enabled when checked.

### Result

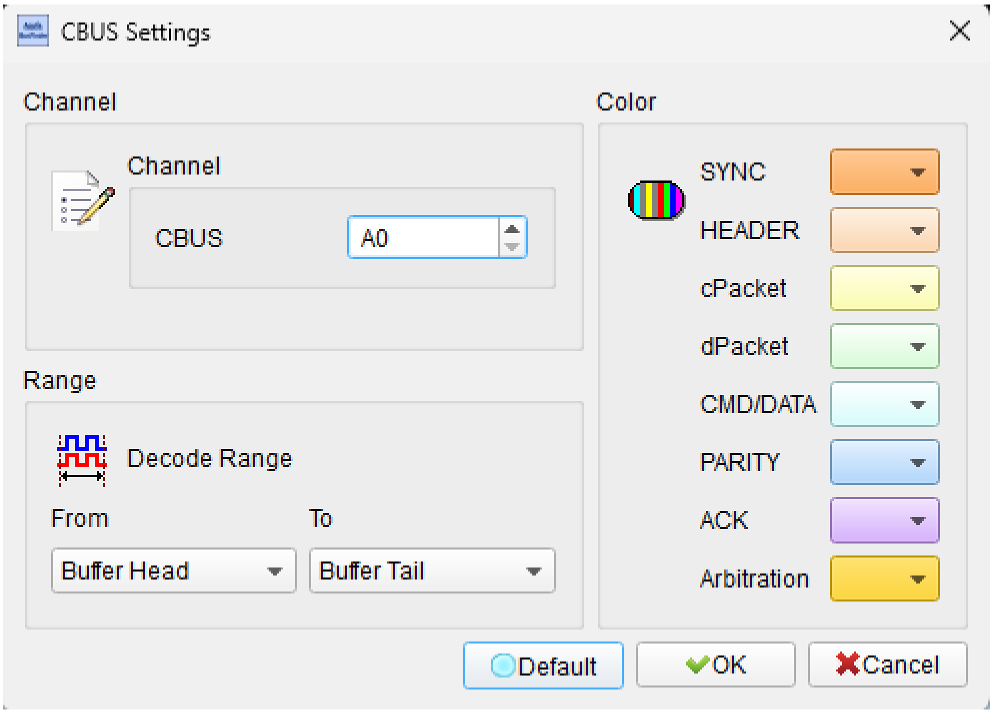


| | Timestamp | ST | OP | PHYADR | REGADR/DEVTYPE | ADDR/DATA |
|---|---|---|---|---|---|---|
| 1 | 180ns | Clause 22 | Write | 05 | 04 | AA55 |

## MHL-CBUS

Mobile High-definition Link (MHL) is an HD audio and video interface, Control Bus (CBUS) is used to control it.

### Settings



CBUS Settings

Channel

Channel

CBUS: A0

Range

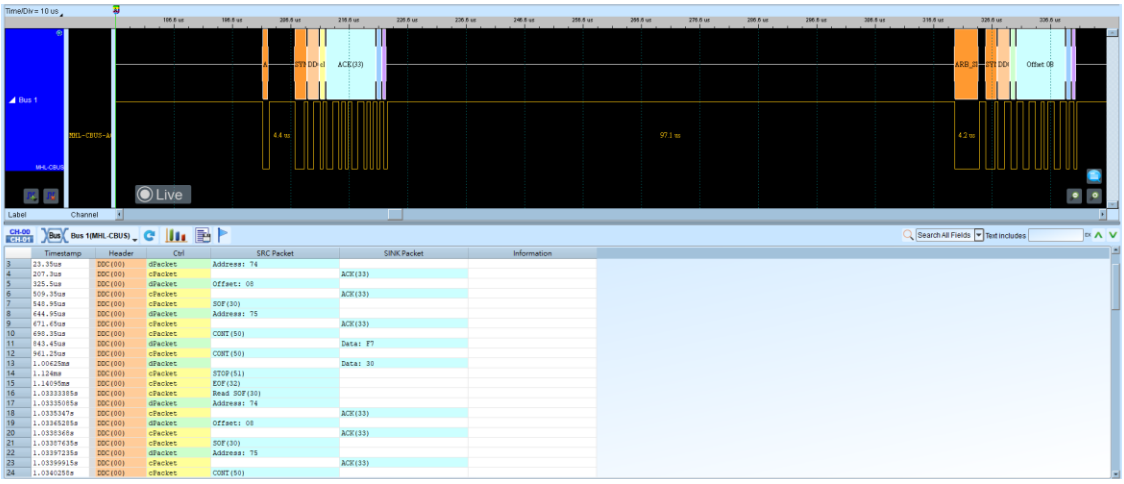
Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

### Result

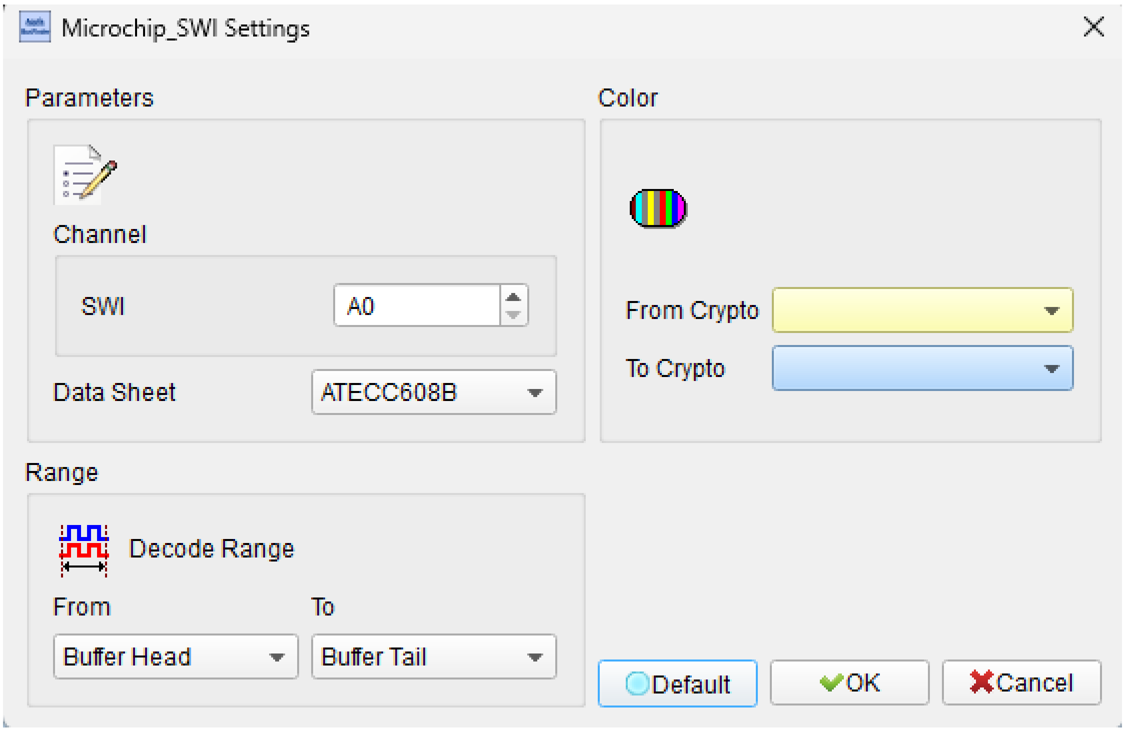


| | Timestamp | Header | Ctrl | SRC Packet | SINK Packet | Information |
|---|---|---|---|---|---|---|
| 3 | 23.35us | DDC(00) | dPacket | Address: 74 | | |
| 4 | 207.3us | DDC(00) | cPacket | | ACK(33) | |
| 5 | 325.5us | DDC(00) | dPacket | Offset: 08 | | |
| 6 | 509.35us | DDC(00) | cPacket | | ACK(33) | |
| 7 | 548.95us | DDC(00) | cPacket | SOF(30) | | |
| 8 | 644.95us | DDC(00) | dPacket | Address: 75 | | |
| 9 | 671.65us | DDC(00) | cPacket | | ACK(33) | |
| 10 | 698.35us | DDC(00) | cPacket | CONT(50) | | |
| 11 | 843.45us | DDC(00) | dPacket | | Data: F7 | |
| 12 | 961.25us | DDC(00) | cPacket | CONT(50) | | |
| 13 | 1.00625ms | DDC(00) | dPacket | | Data: 30 | |
| 14 | 1.124ms | DDC(00) | cPacket | STOP(51) | | |
| 15 | 1.14095ms | DDC(00) | cPacket | EOF(32) | | |
| 16 | 1.03333385s | DDC(00) | cPacket | Read SOF(30) | | |
| 17 | 1.03335085s | DDC(00) | dPacket | Address: 74 | | |
| 18 | 1.0335347s | DDC(00) | cPacket | | ACK(33) | |
| 19 | 1.03365285s | DDC(00) | dPacket | Offset: 08 | | |
| 20 | 1.0338368s | DDC(00) | cPacket | | ACK(33) | |
| 21 | 1.03387635s | DDC(00) | cPacket | SOF(30) | | |
| 22 | 1.03397235s | DDC(00) | dPacket | Address: 75 | | |
| 23 | 1.03399915s | DDC(00) | cPacket | | ACK(33) | |
| 24 | 1.0340258s | DDC(00) | cPacket | CONT(50) | | |

## Microchip SWI

SWI usually refers to Single Wire Interface, a communication protocol provided by Microchip Technology to simplify communication between devices.

## Setting



Microchip_SWI Settings

Parameters

Channel

SWI: A0

Data Sheet: ATECC608B

Color

From Crypto

To Crypto

Range

Decode Range

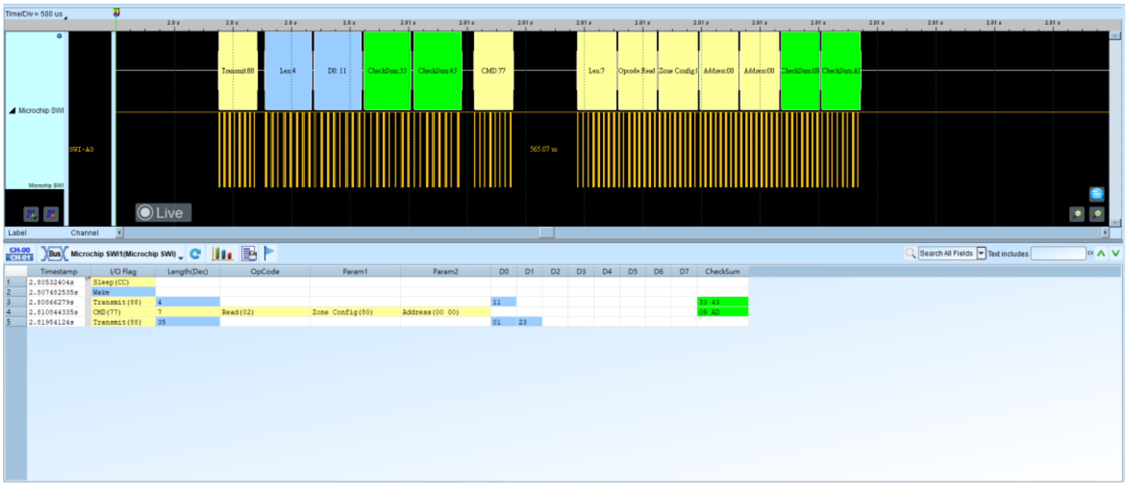
From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Data Sheet:** Choose the supported IC model. We now support ATECC608B.

## Result

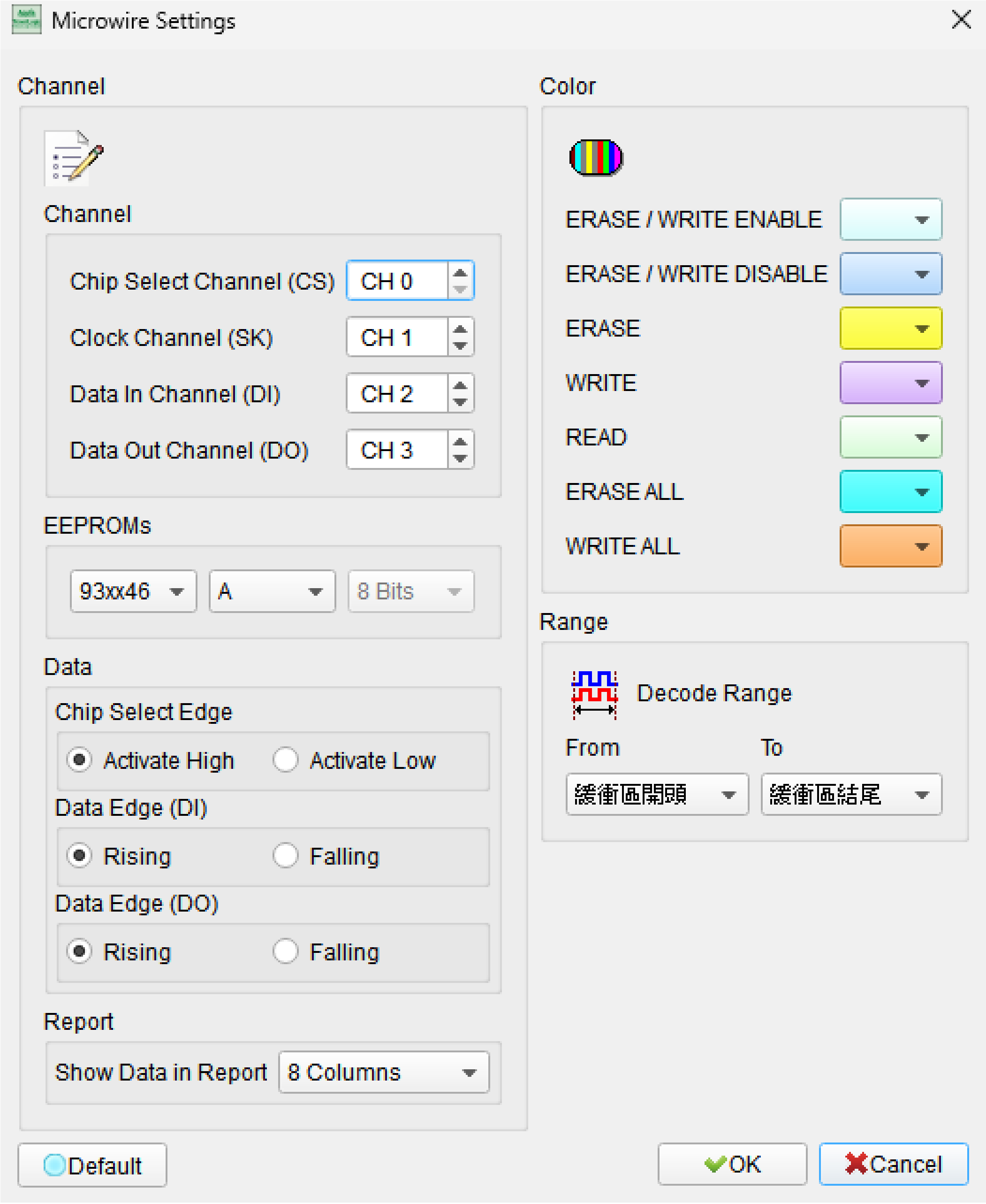


| | Timestamp | I/O Flag | Length(Dec) | OpCode | Param1 | Param2 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | CheckSum |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2.80532404s | Sleep(CC) | | | | | | | | | | | | | |
| 2 | 2.807482535s | Wake | | | | | | | | | | | | | |
| 3 | 2.80866279s | Transmit(88) | 4 | | | | 11 | | | | | | | | 33 43 |
| 4 | 2.810844335s | CMD(77) | 7 | Read(02) | Zone Config(80) | Address(00 00) | | | | | | | | | 09 AD |
| 5 | 2.81954124s | Transmit(88) | 35 | | | | 01 | 23 | | | | | | | |

### Microwire

A serial signal format developed by National Semiconductor, the hardware structure and signal operation are the same as SPI (Serial Peripheral Interface). In the line structure, there are device selection line (CS: Chip Select), clock line (SK: Serial Clock) and data input/output line (DI: Data Input/DO: Data Output).

### Settings



Microwire Settings

Channel

Channel

Chip Select Channel (CS) CH 0

Clock Channel (SK) CH 1

Data In Channel (DI) CH 2

Data Out Channel (DO) CH 3

EEPROMs

93xx46 A 8 Bits

Data

Chip Select Edge

● Activate High ○ Activate Low

Data Edge (DI)

● Rising ○ Falling

Data Edge (DO)

● Rising ○ Falling

Report

Show Data in Report 8 Columns

Color

ERASE / WRITE ENABLE

ERASE / WRITE DISABLE

ERASE

WRITE

READ

ERASE ALL

WRITE ALL

Range

Decode Range

From 緩衝區開頭

To 緩衝區結尾

**Channel**: Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Chip Select Edge:** Active Low or Active High.

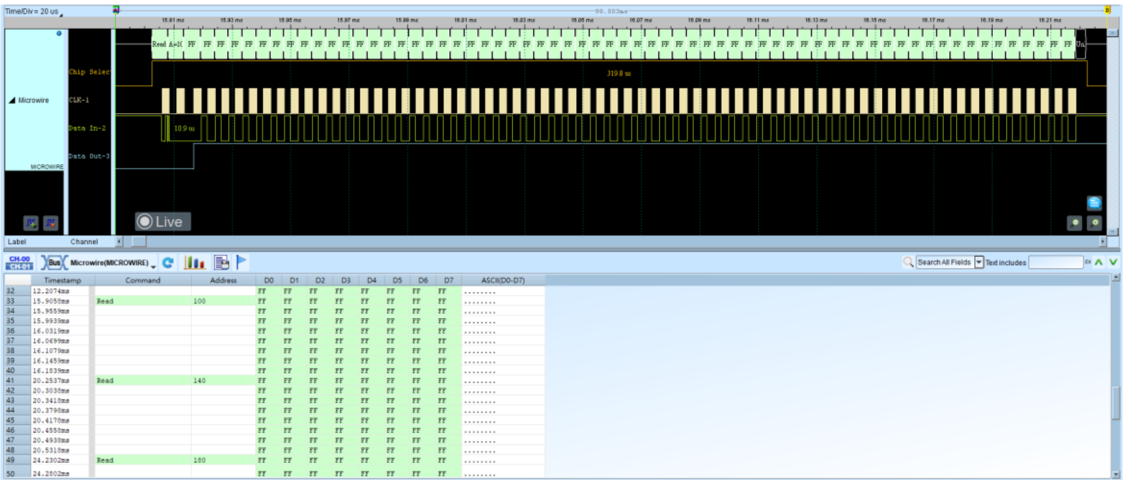
**Data Edge:** Rising or Falling.

**EEPROMs**: Select EEPROMs.

**Report**: Show data in report.

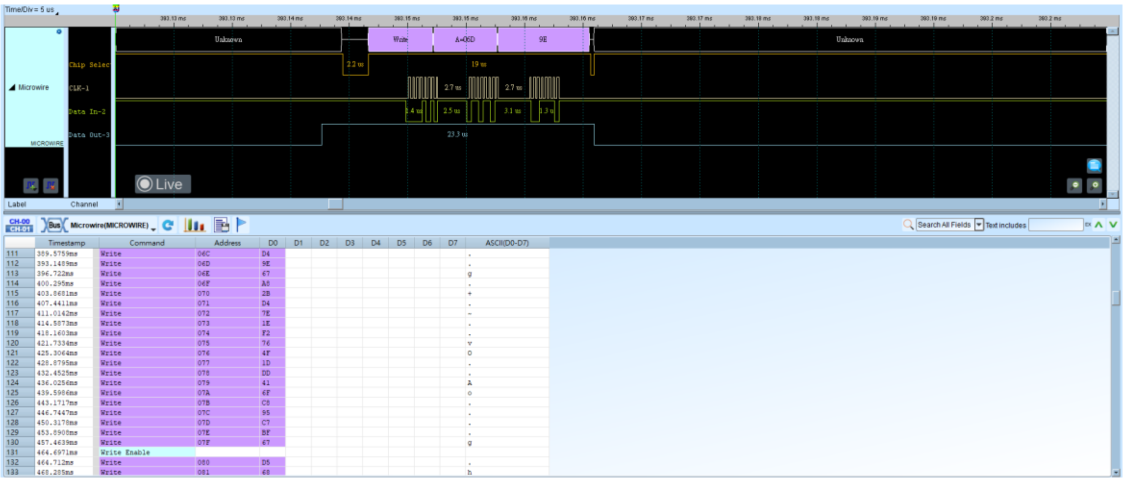
**Result**

Read



| | Timestamp | Command | Address | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII(D0-D7) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 32 | 12.2074ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 33 | 15.9058ms | Read | 100 | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 34 | 15.9559ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 35 | 15.9939ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 36 | 16.0319ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 37 | 16.0699ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 38 | 16.1079ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 39 | 16.1459ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 40 | 16.1839ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 41 | 20.2537ms | Read | 140 | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 42 | 20.3038ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 43 | 20.3418ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 44 | 20.3798ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 45 | 20.4178ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 46 | 20.4558ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 47 | 20.4938ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 48 | 20.5318ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 49 | 24.2302ms | Read | 180 | FF | FF | FF | FF | FF | FF | FF | FF | ........ |
| 50 | 24.2802ms | | | FF | FF | FF | FF | FF | FF | FF | FF | ........ |

Write



| | Timestamp | Command | Address | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII(D0-D7) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 111 | 389.5759ms | Write | 06C | D4 | | | | | | | | . |
| 112 | 393.1489ms | Write | 06D | 9E | | | | | | | | . |
| 113 | 396.722ms | Write | 06E | 67 | | | | | | | | g |
| 114 | 400.295ms | Write | 06F | A8 | | | | | | | | . |
| 115 | 403.8681ms | Write | 070 | 2B | | | | | | | | + |
| 116 | 407.4411ms | Write | 071 | D4 | | | | | | | | . |
| 117 | 411.0142ms | Write | 072 | 7E | | | | | | | | ~ |
| 118 | 414.5873ms | Write | 073 | 1E | | | | | | | | . |
| 119 | 418.1603ms | Write | 074 | F2 | | | | | | | | . |
| 120 | 421.7334ms | Write | 075 | 76 | | | | | | | | v |
| 121 | 425.3064ms | Write | 076 | 4F | | | | | | | | O |
| 122 | 428.8795ms | Write | 077 | 1D | | | | | | | | . |
| 123 | 432.4525ms | Write | 078 | DD | | | | | | | | . |
| 124 | 436.0256ms | Write | 079 | 41 | | | | | | | | A |
| 125 | 439.5986ms | Write | 07A | 6F | | | | | | | | o |
| 126 | 443.1717ms | Write | 07B | C8 | | | | | | | | . |
| 127 | 446.7447ms | Write | 07C | 95 | | | | | | | | . |
| 128 | 450.3178ms | Write | 07D | C7 | | | | | | | | . |
| 129 | 453.8908ms | Write | 07E | BF | | | | | | | | . |
| 130 | 457.4639ms | Write | 07F | 67 | | | | | | | | g |
| 131 | 464.6971ms | Write Enable | | | | | | | | | | |
| 132 | 464.712ms | Write | 080 | D5 | | | | | | | | . |
| 133 | 468.285ms | Write | 081 | 68 | | | | | | | | h |

## MII / RMII / RGMII / GMII

MII: Media Independent Interface

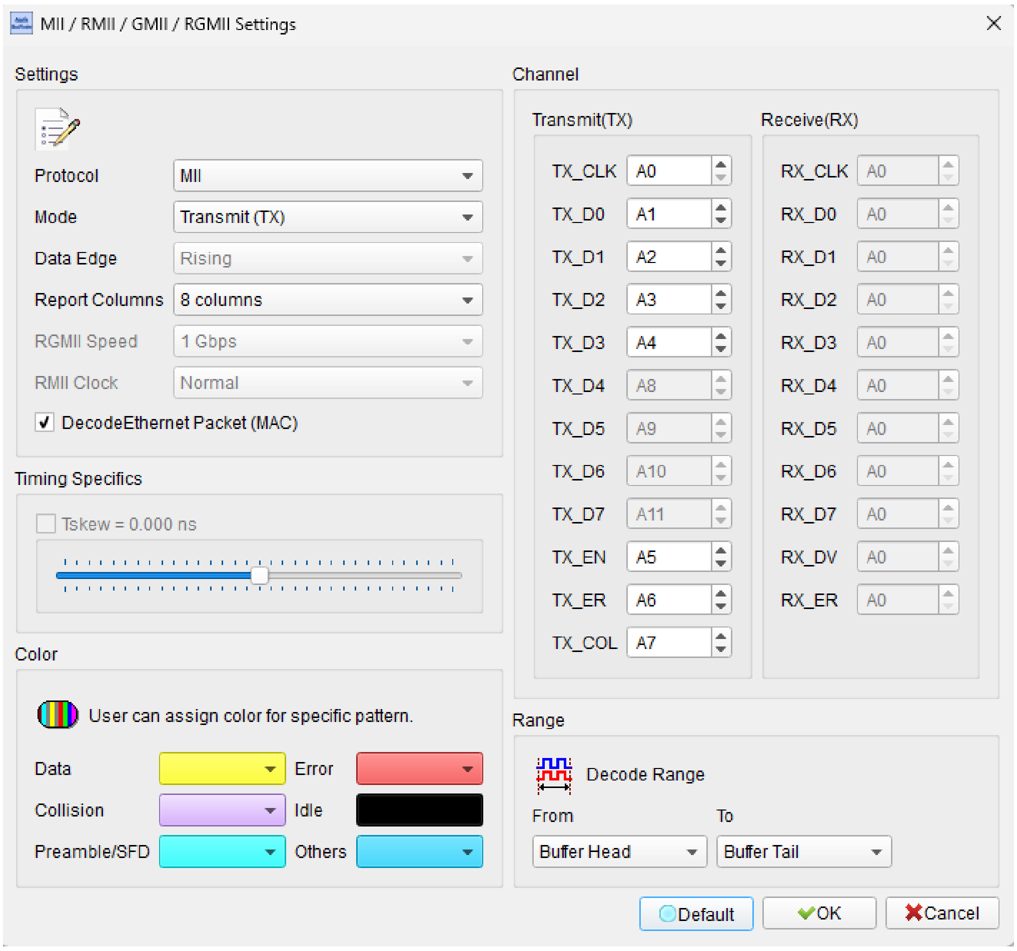
RMII: Reduced Media Independent Interface

RGMII: Reduced Gigabit Media Independent Interface

GMII: Gigabit Media Independent Interface

Formulated by 802.3u that applied to Fast Ethernet, connecting MAC of Data Link Layer and PHY layer. Its clock frequency is either 25MHz or 2.5MHz (Ethernet); they are TX_CLK and RX_CLK. TX [0:3], RX [0:3] are 4-bit-width bus and TX_EN, RX_EN enable the IN/OUT; TX_ER, RX_ER can detect the errors on the bus; RX_DV inform bus the data received is valid or not; COL can detect the collision on the bus. **Serial Management Interface (SMI), also known as MDIO, is also an important part of MII.**

### Settings



MII / RMII / GMII / RGMII Settings

Settings

Protocol: MII

Mode: Transmit (TX)

Data Edge: Rising

Report Columns: 8 columns

RGMII Speed: 1 Gbps

RMII Clock: Normal

☑ DecodeEthernet Packet (MAC)

Timing Specifics

□ Tskew = 0.000 ns

Color

User can assign color for specific pattern.

| | | | |
|---|---|---|---|
| Data | 🟨 | Error | 🟥 |
| Collision | 🟪 | Idle | ⬛ |
| Preamble/SFD | 🟦 | Others | 🟦 |

Channel

| Transmit(TX) | | Receive(RX) | |
|---|---|---|---|
| TX_CLK | A0 | RX_CLK | A0 |
| TX_D0 | A1 | RX_D0 | A0 |
| TX_D1 | A2 | RX_D1 | A0 |
| TX_D2 | A3 | RX_D2 | A0 |
| TX_D3 | A4 | RX_D3 | A0 |
| TX_D4 | A8 | RX_D4 | A0 |
| TX_D5 | A9 | RX_D5 | A0 |
| TX_D6 | A10 | RX_D6 | A0 |
| TX_D7 | A11 | RX_D7 | A0 |
| TX_EN | A5 | RX_DV | A0 |
| TX_ER | A6 | RX_ER | A0 |
| TX_COL | A7 | | |

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Protocol:** MII / RMII / GMII / RGMII can be selected. except BusFinder / LA, TravelLogic and MSO do not support GMII.

**Mode:** Transmit (Tx) or Receive (Rx) mode can be selected.

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Data Edge:**

**Rising:** Latch data when Clock edge is rising.

**Falling:** Latch data when Clock edge is falling.

**Report Column:**

**8 Columns:** Set the data fields of the report window to be displayed in 8 columns.

**16 Columns:** Set the data fields of the report window to be displayed in 16 columns.

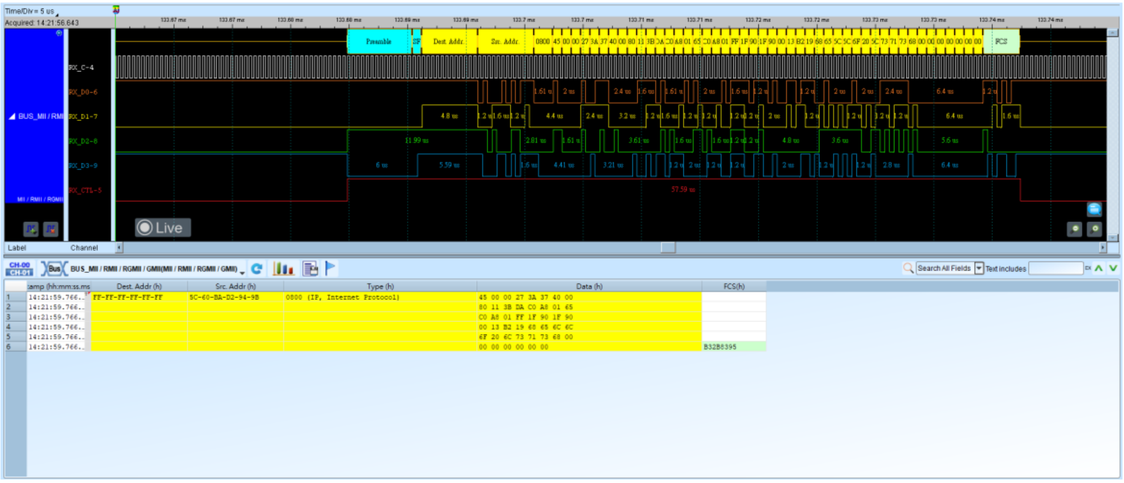
**RGMII Speed:** Set the speed of RGMII to 1Gbps or 100/10Mbps, valid only when MII is set to RGMII.

**RMII Clock:** Set the RMII Clock to Normal or Decrease Latch Frequency (x10). Only available when MII is set to RMII.

**Decode Ethernet Packet (MAC):** Decode MAC packets. Enabled when checked.

**Timing Specifics:** Sets the delay or advance of the Data Latch. Available only when MII is set to GMII. Enabled when the check box is selected.

## Result

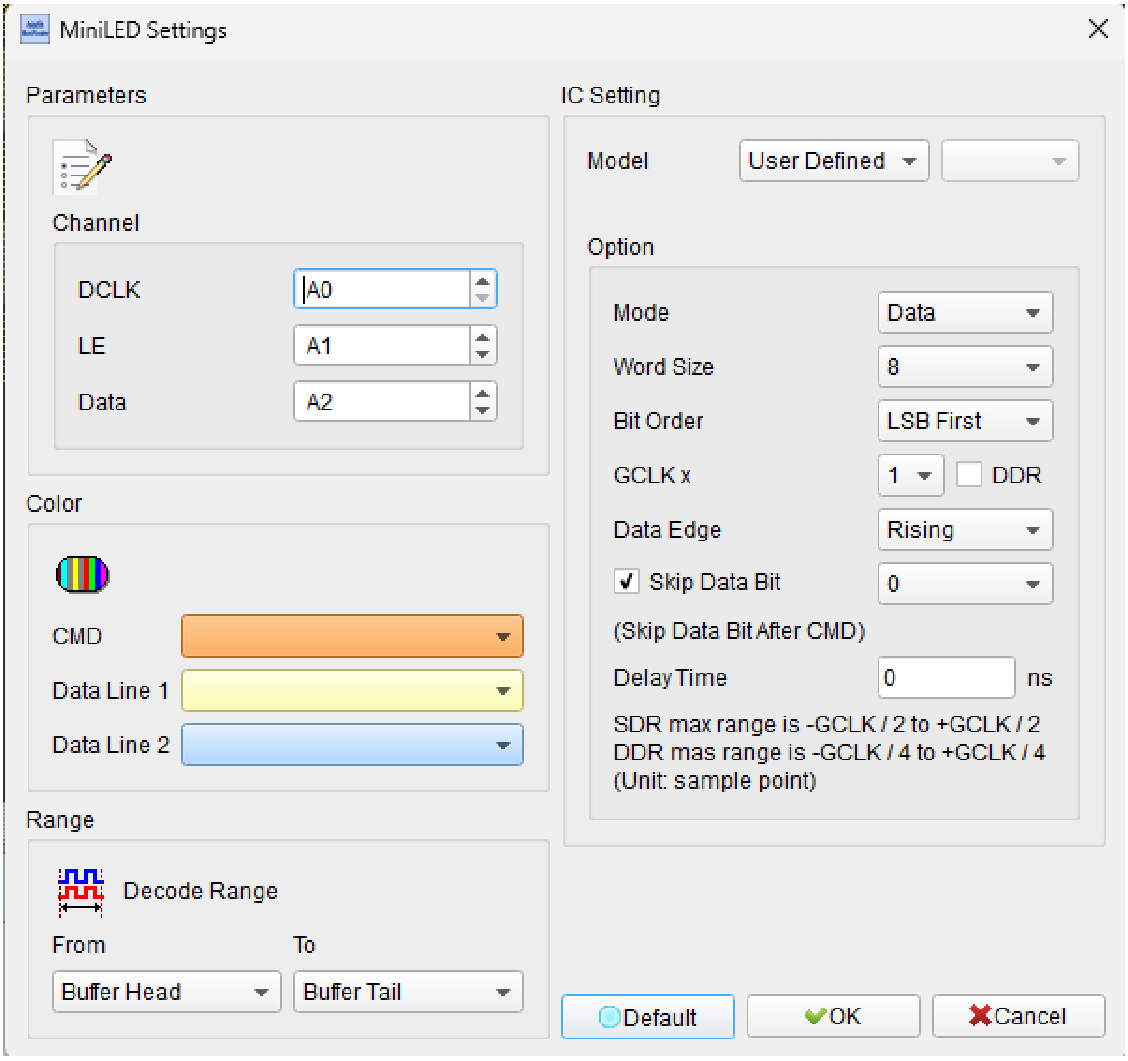


| | amp (hh:mm:ss.ms | Dest. Addr (h) | Src. Addr (h) | Type (h) | Data (h) | FCS(h) |
|---|---|---|---|---|---|---|
| 1 | 14:21:59.766.. | FF-FF-FF-FF-FF-FF | 5C-60-BA-D2-94-9B | 0800 (IP, Internet Protocol) | 45 00 00 27 3A 37 40 00 | |
| 2 | 14:21:59.766.. | | | | 80 11 3B DA C0 A8 01 65 | |
| 3 | 14:21:59.766.. | | | | C0 A8 01 FF 1F 90 1F 90 | |
| 4 | 14:21:59.766.. | | | | 00 13 B2 19 68 65 6C 6C | |
| 5 | 14:21:59.766.. | | | | 6F 20 6C 73 71 73 68 00 | |
| 6 | 14:21:59.766.. | | | | 00 00 00 00 00 00 | B32B8395 |

## Mini / Micro-LED

The die positioning of display panel in Mini LED is 100 ~ 200μm, and Micro LED is below 50μm.

Parameter Settings



Parameters

Channel

| | |
|---|---|
| DCLK | A0 |
| LE | A1 |
| Data | A2 |

Color

CMD

Data Line 1

Data Line 2

Range

Decode Range

From: Buffer Head

To: Buffer Tail

IC Setting

Model: User Defined

Option

| | |
|---|---|
| Mode | Data |
| Word Size | 8 |
| Bit Order | LSB First |
| GCLK x | 1 □ DDR |
| Data Edge | Rising |
| ☑ Skip Data Bit | 0 |

(Skip Data Bit After CMD)

Delay Time 0 ns

SDR max range is -GCLK / 2 to +GCLK / 2
DDR mas range is -GCLK / 4 to +GCLK / 4
(Unit: sample point)

### Channel

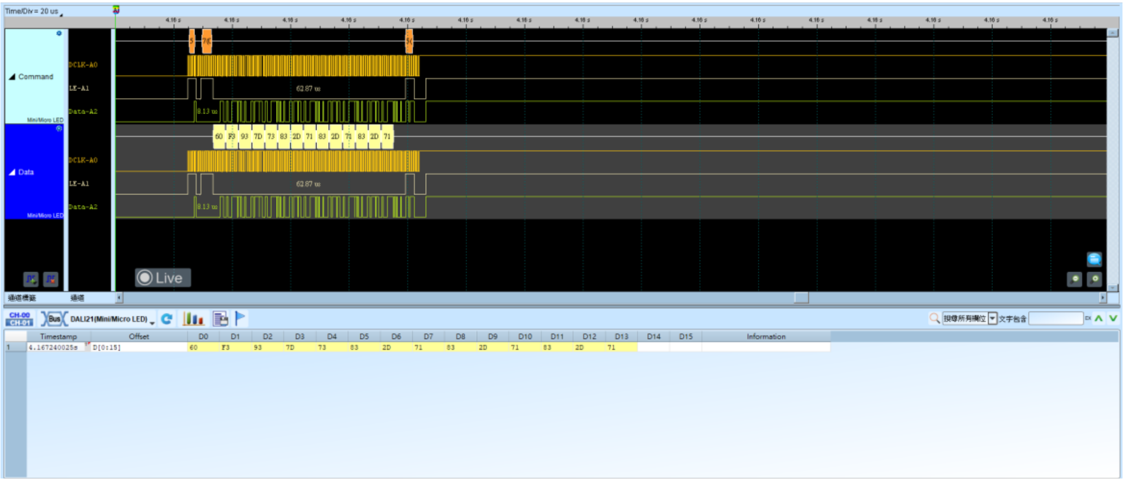
1. **DCLK:** Default CLK channel
2. **Data:** Data channel
3. **LE:** The channel that can switch CMD or DATA

### IC Setting:

1. **Model:** Setting the IC model number, currently supports ICND, MBI and User Defined; after selecting the model number, user can select the IC number again.

2. **Mode:** Set the Mode of the IC to Data or Command.
3. **Word size:** The number of the data bits.
4. **Bit order:** MSB/LSB first.
5. **GCLK:** The magnification of DCLK.
6. **DDR:** DDR mode. Enabled when checked.
7. **Data Edge:** Rising/falling latch.
8. **Skip Data Bit:** Set the number of skip data bit after LE falling
9. **Delay Time:** Set the time lead / delay in data line.

## Result

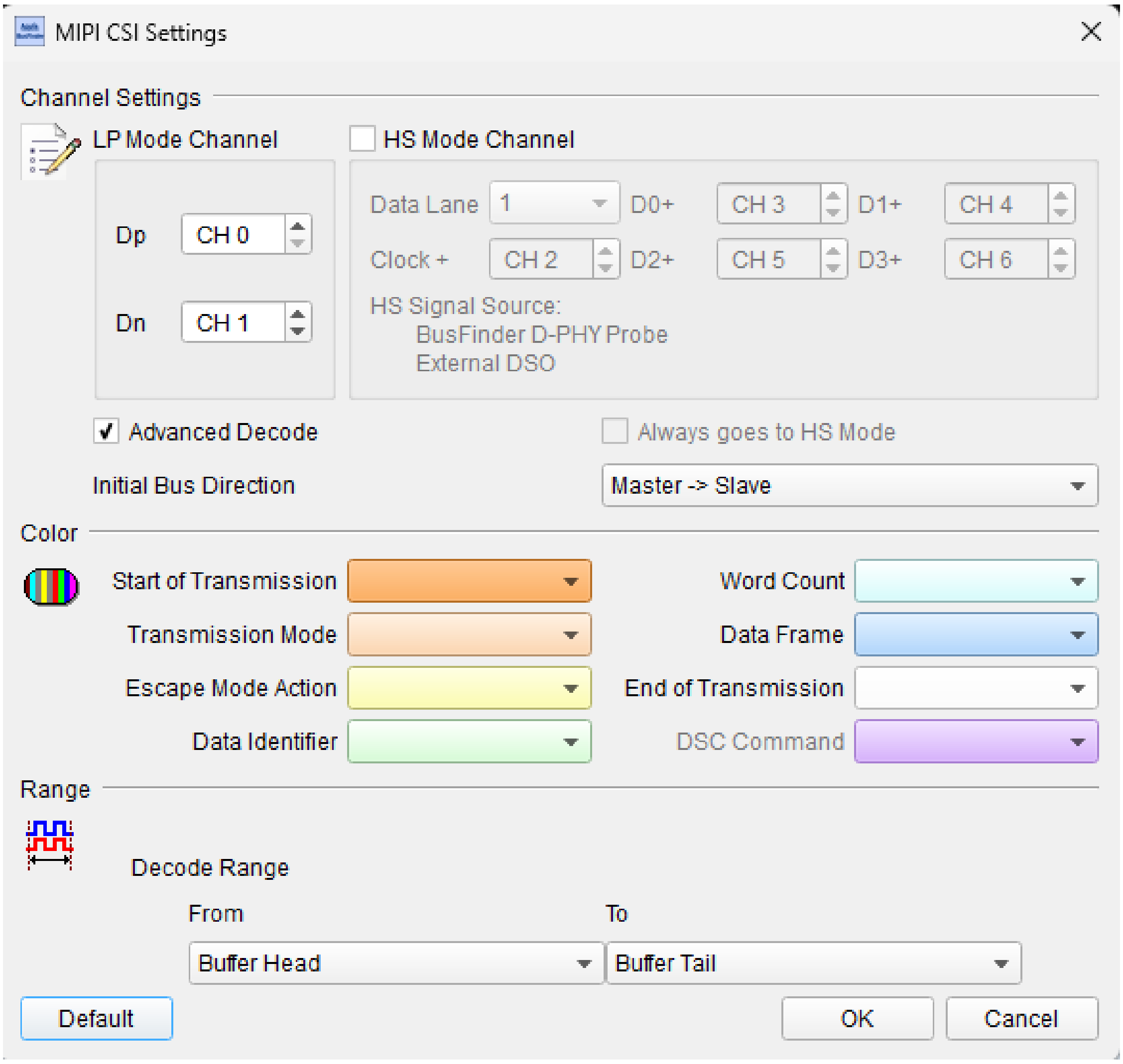


| | Timestamp | Offset | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4.167240025s | D[0:15] | 60 | F3 | 93 | 7D | 73 | 83 | 2D | 71 | 83 | 2D | 71 | 83 | 2D | 71 | | | |

## MIPI CSI

MIPI CSI (Mobile Industry Processor Interface Camera Serial Interface) is a standardized data transfer protocol specifically designed to connect image sensors (such as camera modules) to processors (such as mobile phones, tablets or embedded devices). It was developed by the MIPI Alliance (Mobile Industry Processor Interface Alliance) to provide an efficient, low-power and high-speed data transfer channel.

## Settings



MIPI CSI Settings

Channel Settings

LP Mode Channel
Dp: CH 0
Dn: CH 1

□ HS Mode Channel
Data Lane: 1
D0+: CH 3
D1+: CH 4
Clock +: CH 2
D2+: CH 5
D3+: CH 6
HS Signal Source:
BusFinder D-PHY Probe
External DSO

☑ Advanced Decode
□ Always goes to HS Mode
Initial Bus Direction: Master -> Slave

Color
Start of Transmission: 🟧
Transmission Mode: 🟧
Escape Mode Action: 🟨
Data Identifier: 🟩
Word Count: 🟦
Data Frame: 🟦
End of Transmission: ⬜
DSC Command: 🟪

Range
Decode Range
From: Buffer Head
To: Buffer Tail

**Dp, Dn:** The signal channel for DSI-LP mode.

**Data Lane:** The lane amount under DSI-HS mode.

**Clock+, D0+, D1+, D2+, D3+:** The signal channel under DSI-HS mode. Enabled when checked.

**Advanced Decode:** Decodes the data in CSI format. Enabled when checked.

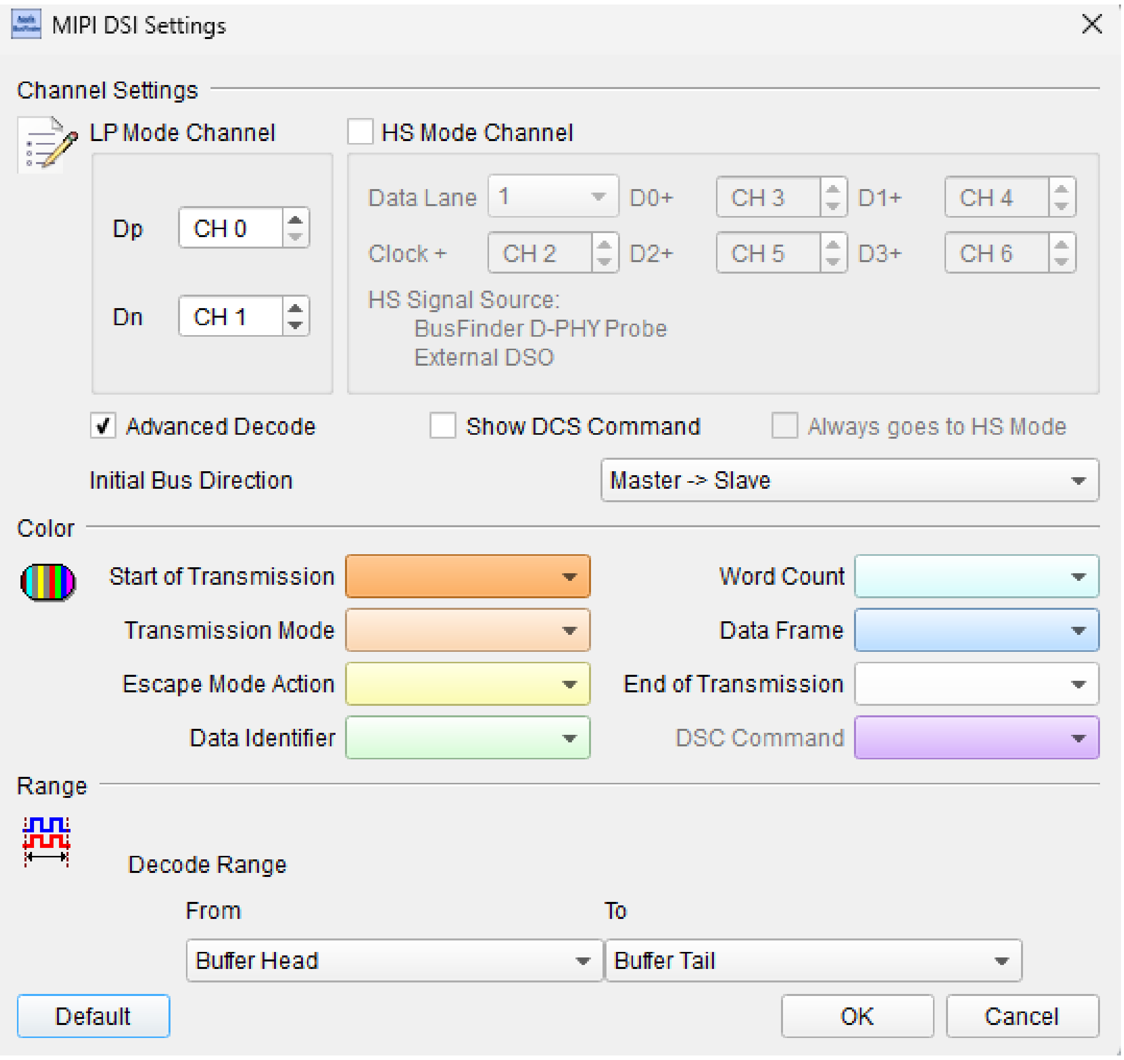
**Always goes to HS Mode:** Ignore the status of Dp and Dn in DSI-LP mode, and always read the data as HS-Mode, enabled when checked.

**Initial Bus Direction:** Select the data transmission direction of the bus in the initial state.

## MIPI DSI

MIPI Display Serial Interface (DSI) designed by MIPI alliance for the protocols between a host processor and peripheral devices using a D-PHY physical interface. The operation mode includes High Speed Mode and Low Power Mode (LPM).

## Settings



MIPI DSI Settings

Channel Settings

LP Mode Channel

Dp: CH 0

Dn: CH 1

□ HS Mode Channel

Data Lane: 1

D0+: CH 3

D1+: CH 4

Clock +: CH 2

D2+: CH 5

D3+: CH 6

HS Signal Source:
BusFinder D-PHY Probe
External DSO

☑ Advanced Decode

□ Show DCS Command

□ Always goes to HS Mode

Initial Bus Direction: Master -> Slave

Color

Start of Transmission

Transmission Mode

Escape Mode Action

Data Identifier

Word Count

Data Frame

End of Transmission

DSC Command

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Dp, Dn:** DSI-LP signal lines

**Data Lane:** DSI-HS mode Data Lane number

**Clock+, D0+, D1+, D2+, D3+:** DSI-HS signal lines

**Advanced Decode:** Enable DSI format decode and display.

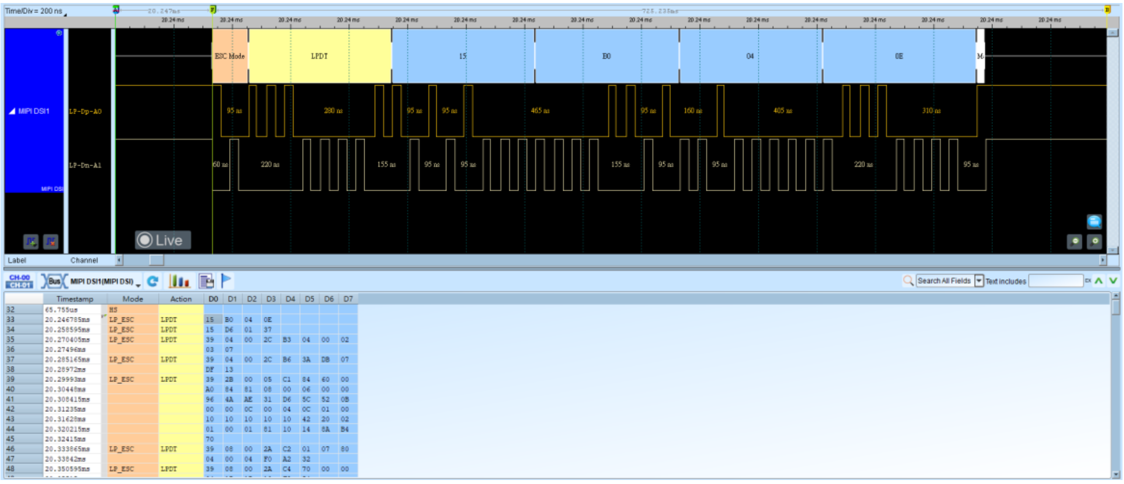
**Show DCS Command:** Enable DCS Command decode and display.

**Always goes to HS Mode:** Ignore the Dp and Dn status and decode all the data frame in HS mode

**Initial Bus Direction:** Select the Initial direction of the bus transmission.

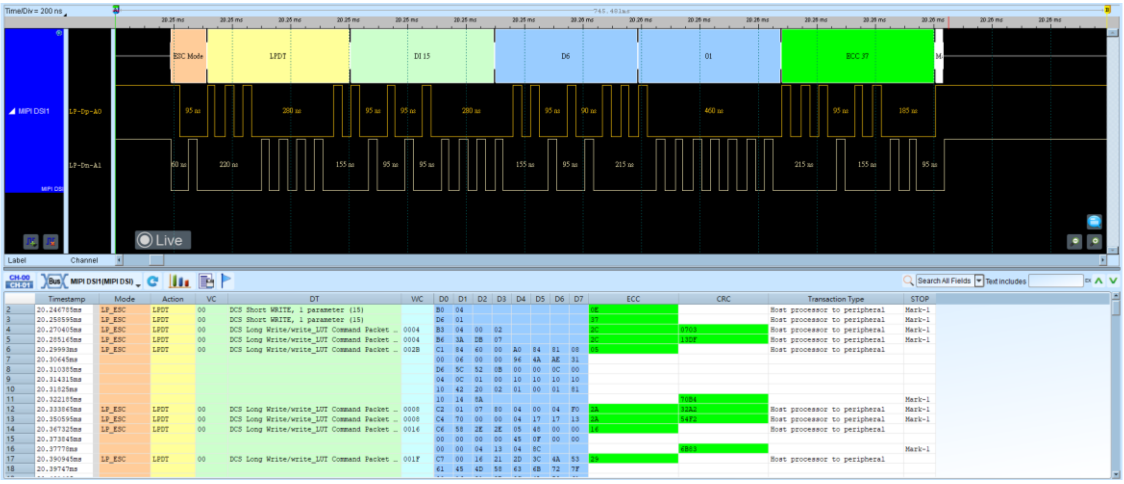
## Result

Advanced Decode Disabled:



| | Timestamp | Mode | Action | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 32 | 65.755us | HS | | | | | | | | | |
| 33 | 20.246785ms | LP_ESC | LPDT | 15 | B0 | 04 | 0E | | | | |
| 34 | 20.258595ms | LP_ESC | LPDT | 15 | D6 | 01 | 37 | | | | |
| 35 | 20.270405ms | LP_ESC | LPDT | 39 | 04 | 00 | 2C | B3 | 04 | 00 | 02 |
| 36 | 20.27496ms | | | 03 | 07 | | | | | | |
| 37 | 20.285165ms | LP_ESC | LPDT | 39 | 04 | 00 | 2C | B6 | 3A | DB | 07 |
| 38 | 20.28972ms | | | DF | 13 | | | | | | |
| 39 | 20.29993ms | LP_ESC | LPDT | 39 | 2B | 00 | 05 | C1 | 84 | 60 | 00 |
| 40 | 20.30448ms | | | A0 | 84 | 81 | 08 | 00 | 06 | 00 | 00 |
| 41 | 20.308415ms | | | 96 | 4A | AE | 31 | D6 | 5C | 52 | 0B |
| 42 | 20.31235ms | | | 00 | 00 | 0C | 00 | 04 | 0C | 01 | 00 |
| 43 | 20.31628ms | | | 10 | 10 | 10 | 10 | 10 | 42 | 20 | 02 |
| 44 | 20.320215ms | | | 01 | 00 | 01 | 81 | 10 | 14 | 8A | B4 |
| 45 | 20.32415ms | | | 70 | | | | | | | |
| 46 | 20.333865ms | LP_ESC | LPDT | 39 | 08 | 00 | 2A | C2 | 01 | 07 | 80 |
| 47 | 20.33842ms | | | 04 | 00 | 04 | F0 | A2 | 32 | | |
| 48 | 20.350595ms | LP_ESC | LPDT | 39 | 08 | 00 | 2A | C4 | 70 | 00 | 00 |

Advanced Decode Enabled:



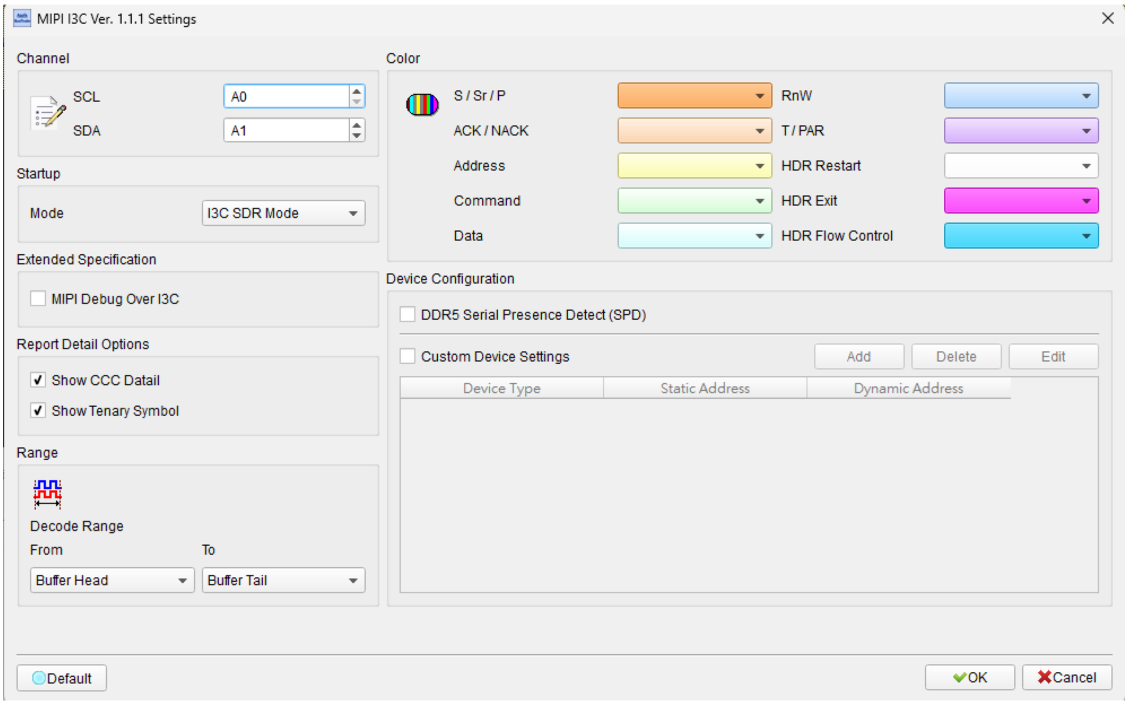
| | Timestamp | Mode | Action | VC | DT | WC | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ECC | CRC | Transaction Type | STOP |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 20.246785ms | LP_ESC | LPDT | 00 | DCS Short WRITE, 1 parameter (15) | | B0 | 04 | | | | | | | 0E | | Host processor to peripheral | Mark-1 |
| 3 | 20.258595ms | LP_ESC | LPDT | 00 | DCS Short WRITE, 1 parameter (15) | | D6 | 01 | | | | | | | 37 | | Host processor to peripheral | Mark-1 |
| 4 | 20.270405ms | LP_ESC | LPDT | 00 | DCS Long Write/write_LUT Command Packet … | 0004 | B3 | 04 | 00 | 02 | | | | | 2C | 0703 | Host processor to peripheral | Mark-1 |
| 5 | 20.285165ms | LP_ESC | LPDT | 00 | DCS Long Write/write_LUT Command Packet … | 0004 | B6 | 3A | DB | 07 | | | | | 2C | 13DF | Host processor to peripheral | Mark-1 |
| 6 | 20.29993ms | LP_ESC | LPDT | 00 | DCS Long Write/write_LUT Command Packet … | 002B | C1 | 84 | 60 | 00 | A0 | 84 | 81 | 08 | 05 | | Host processor to peripheral | |
| 7 | 20.30645ms | | | | | | 00 | 06 | 00 | 00 | 96 | 4A | AE | 31 | | | | |
| 8 | 20.310385ms | | | | | | D6 | 5C | 52 | 0B | 00 | 00 | 0C | 00 | | | | |
| 9 | 20.314315ms | | | | | | 04 | 0C | 01 | 00 | 10 | 10 | 10 | 10 | | | | |
| 10 | 20.31825ms | | | | | | 10 | 42 | 20 | 02 | 01 | 00 | 01 | 81 | | | | |
| 11 | 20.322185ms | | | | | | 10 | 14 | 8A | | | | | | | 70B4 | | Mark-1 |
| 12 | 20.333865ms | LP_ESC | LPDT | 00 | DCS Long Write/write_LUT Command Packet … | 0008 | C2 | 01 | 07 | 80 | 04 | 00 | 04 | F0 | 2A | 32A2 | Host processor to peripheral | Mark-1 |
| 13 | 20.350595ms | LP_ESC | LPDT | 00 | DCS Long Write/write_LUT Command Packet … | 0008 | C4 | 70 | 00 | 00 | 04 | 17 | 17 | 13 | 2A | 54F2 | Host processor to peripheral | Mark-1 |
| 14 | 20.367325ms | LP_ESC | LPDT | 00 | DCS Long Write/write_LUT Command Packet … | 0016 | C6 | 58 | 2E | 2E | 05 | 48 | 00 | 00 | 16 | | Host processor to peripheral | |
| 15 | 20.373845ms | | | | | | 00 | 00 | 00 | 00 | 45 | 0F | 00 | 00 | | | | |
| 16 | 20.37778ms | | | | | | 00 | 00 | 04 | 13 | 04 | 8C | | | | 6B83 | | Mark-1 |
| 17 | 20.390945ms | LP_ESC | LPDT | 00 | DCS Long Write/write_LUT Command Packet … | 001F | C7 | 00 | 16 | 21 | 2D | 3C | 4A | 53 | 29 | | Host processor to peripheral | |
| 18 | 20.39747ms | | | | | | 61 | 45 | 4D | 58 | 63 | 6B | 72 | 7F | | | | |

## MIPI I3C

MIPI I3C is an extension of I2C interface, so it still maintains the two-wire SCL (clock), SDA (data) as I2C. The frequency of MIPI I3C SCL clock is defined in spec. up to 12.9 MHz, and generally it is 12.5 MHz. It supports three operating voltages, 1.2 V / 1.8 V / 3.3 V. The MIPI I3C is a new generation of sensor interface specification, which integrates multiple sensor interfaces in one unified specification, mainly for simplifying the application of smart phone, smart phone, and sensor interface.

MIPI I3C is a new generation sensor interface specification that integrates multiple sensor interfaces in a unified specification, with the main application being to simplify sensor integration in smart phones, IoT devices, and automotive systems.

## Settings



MIPI I3C Ver. 1.1.1 Settings

Channel
SCL: A0
SDA: A1

Startup
Mode: I3C SDR Mode

Extended Specification
□ MIPI Debug Over I3C

Report Detail Options
☑ Show CCC Datail
☑ Show Tenary Symbol

Range
Decode Range
From: Buffer Head
To: Buffer Tail

Color

| | | | |
|---|---|---|---|
| S / Sr / P | 🟧 | RnW | 🟦 |
| ACK / NACK | 🟧 | T / PAR | 🟪 |
| Address | 🟨 | HDR Restart | ⬜ |
| Command | 🟩 | HDR Exit | 🟪 |
| Data | ⬜ | HDR Flow Control | 🟦 |

Device Configuration
□ DDR5 Serial Presence Detect (SPD)
□ Custom Device Settings

| Device Type | Static Address | Dynamic Address |
|---|---|---|

**Channel:**

1. **Clock Channel (SCL):** Transfer clock of I3C.
2. **Data Channel (SDA):** Transfer data of I3C.

**Startup:** Specifies the mode that is currently running on the bus. User can set the mode to:

1. I3C SDR Mode
2. I2C Mode
3. I3C HDR-DDR Mode
4. I3C HDR-TSP Mode
5. I3C HDR-TSL Mode

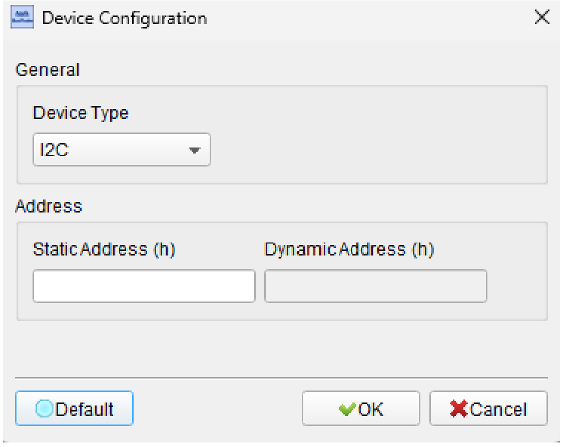
**Extended Specification—MIPI Debug Over I3C:** The command for debugging I3C. Enabled when checked

**Report Detail Options:**

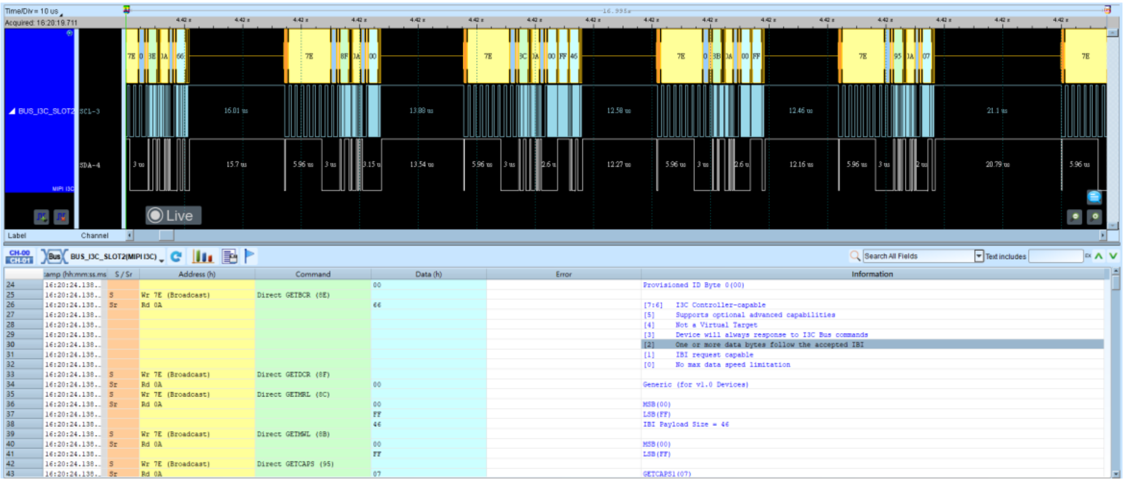
1. **Show CCC Detail:** Show the CCC(Common Command Code) information in report. Enabled when checked.
2. **Show Ternary Symbol:** Show the Ternary Symbol in report. Enabled when checked.

**Device Configuration:**

1. **DDR5 Serial Presence Detect (SPD):** SPD function. Enabled when checked.
2. **Custom Device Settings:** Add user-defined device. User can add I$^2$C or MCTP device. Enabled when checked.



## Result

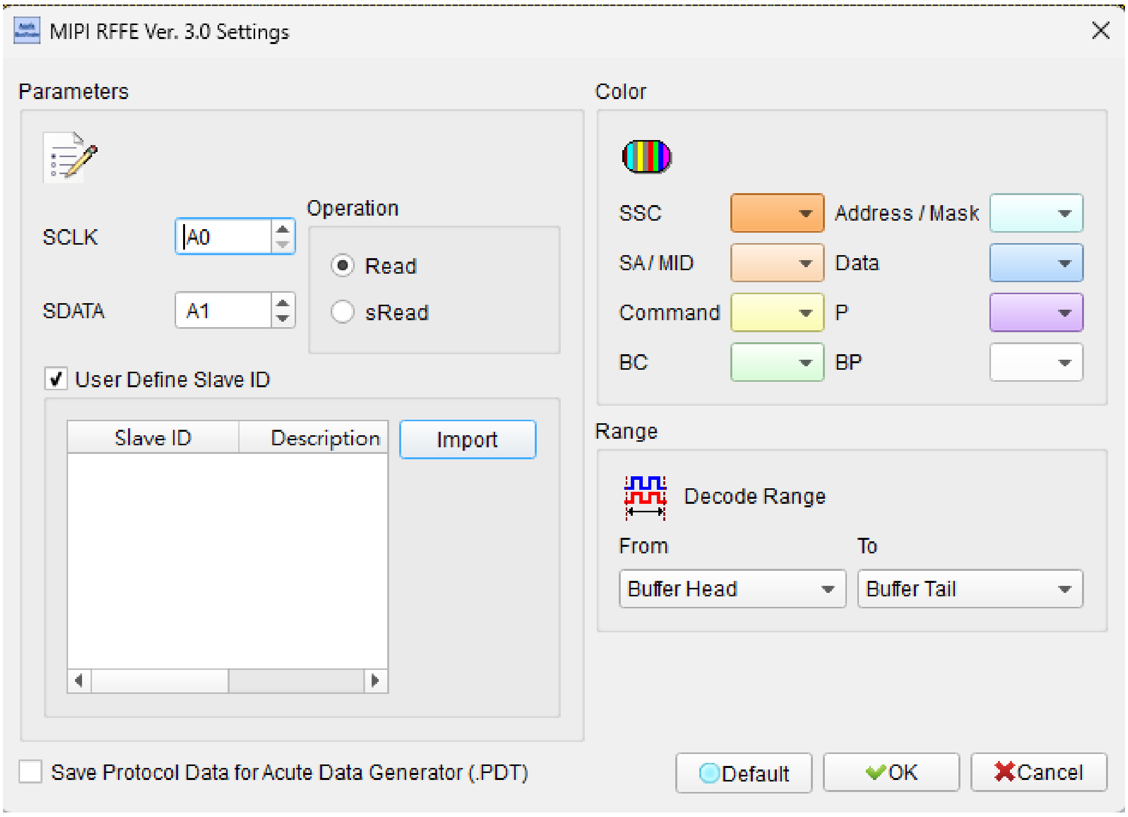


| | amp (hh:mm:ss.ms | S / Sr | Address (h) | Command | Data (h) | Error | Information |
|---|---|---|---|---|---|---|---|
| 24 | 16:20:24.138.. | | | | 00 | | Provisioned ID Byte 0(00) |
| 25 | 16:20:24.138.. | S | Wr 7E (Broadcast) | Direct GETBCR (8E) | | | |
| 26 | 16:20:24.138.. | Sr | Rd 0A | | 66 | | [7:6] I3C Controller-capable |
| 27 | 16:20:24.138.. | | | | | | [5] Supports optional advanced capabilities |
| 28 | 16:20:24.138.. | | | | | | [4] Not a Virtual Target |
| 29 | 16:20:24.138.. | | | | | | [3] Device will always response to I3C Bus commands |
| 30 | 16:20:24.138.. | | | | | | [2] One or more data bytes follow the accepted IBI |
| 31 | 16:20:24.138.. | | | | | | [1] IBI request capable |
| 32 | 16:20:24.138.. | | | | | | [0] No max data speed limitation |
| 33 | 16:20:24.138.. | S | Wr 7E (Broadcast) | Direct GETDCR (8F) | | | |
| 34 | 16:20:24.138.. | Sr | Rd 0A | | 00 | | Generic (for v1.0 Devices) |
| 35 | 16:20:24.138.. | S | Wr 7E (Broadcast) | Direct GETMRL (8C) | | | |
| 36 | 16:20:24.138.. | Sr | Rd 0A | | 00 | | MSB(00) |
| 37 | 16:20:24.138.. | | | | FF | | LSB(FF) |
| 38 | 16:20:24.138.. | | | | 46 | | IBI Payload Size = 46 |
| 39 | 16:20:24.138.. | S | Wr 7E (Broadcast) | Direct GETMWL (8B) | | | |
| 40 | 16:20:24.138.. | Sr | Rd 0A | | 00 | | MSB(00) |
| 41 | 16:20:24.138.. | | | | FF | | LSB(FF) |
| 42 | 16:20:24.138.. | S | Wr 7E (Broadcast) | Direct GETCAPS (95) | | | |
| 43 | 16:20:24.138.. | Sr | Rd 0A | | 07 | | GETCAPS1(07) |

## MIPI RFFE

MIPI RFFE (RF Front-End Control Interface) designed by MIPI alliance is for controlling RF front-end devices including Power Amplifiers, Low-Noise Amplifiers, filters, switches, power management modules, antenna tuners and sensors.

### Settings



MIPI RFFE Ver. 3.0 Settings

Parameters

SCLK: A0

SDATA: A1

Operation: ● Read ○ sRead

☑ User Define Slave ID

| Slave ID | Description |
|---|---|

Color

| | | | |
|---|---|---|---|
| SSC | 🟧 | Address / Mask | 🟦 |
| SA / MID | 🟧 | Data | 🟦 |
| Command | 🟨 | P | 🟪 |
| BC | 🟩 | BP | ⬜ |

Range

Decode Range

From: Buffer Head

To: Buffer Tail

□ Save Protocol Data for Acute Data Generator (.PDT)

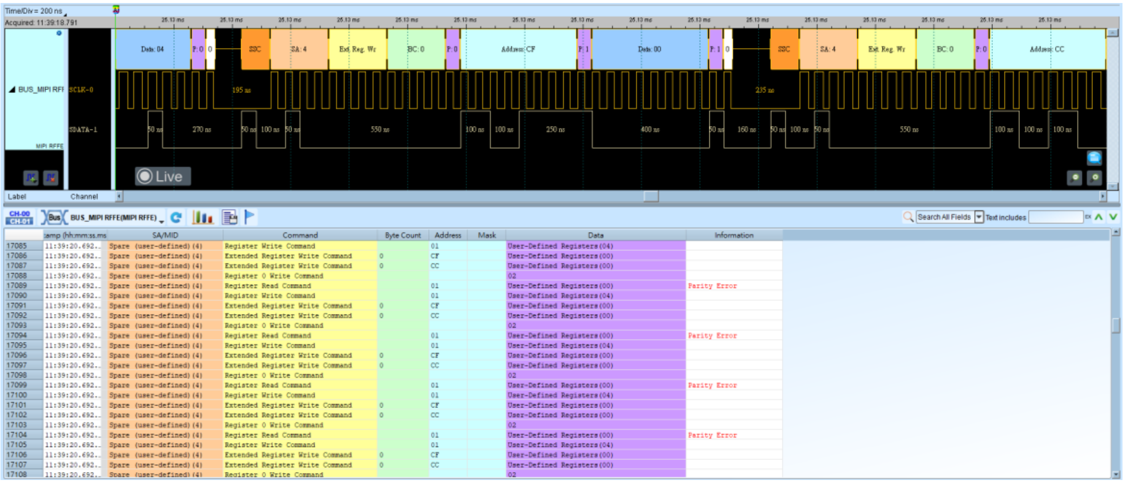
**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Operation:** Set to Read or sRead.

**User Define Slave ID:** Add used-defined Slave ID. Enabled when checked.

**Save Protocol Data for Acute Data Generator (.PDT):** Save the analyzed result as .PDT file for the usage of Acute Data Generator. Enabled when checked.

## Result



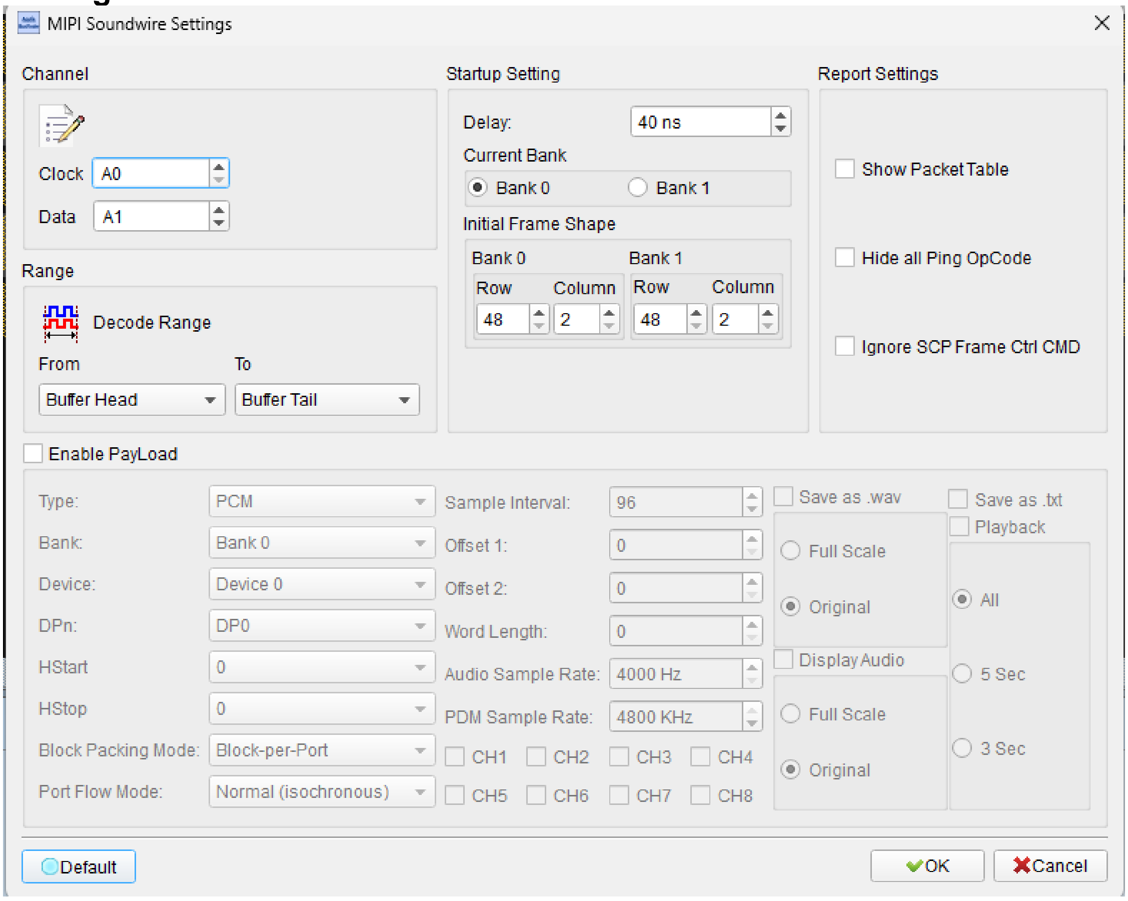
| | amp (hh:mm:ss.ms | SA/MID | Command | Byte Count | Address | Mask | Data | Information |
|---|---|---|---|---|---|---|---|---|
| 17085 | 11:39:20.692... | Spare (user-defined) (4) | Register Write Command | | 01 | | User-Defined Registers(04) | |
| 17086 | 11:39:20.692... | Spare (user-defined) (4) | Extended Register Write Command | 0 | CF | | User-Defined Registers(00) | |
| 17087 | 11:39:20.692... | Spare (user-defined) (4) | Extended Register Write Command | 0 | CC | | User-Defined Registers(00) | |
| 17088 | 11:39:20.692... | Spare (user-defined) (4) | Register 0 Write Command | | | | 02 | |
| 17089 | 11:39:20.692... | Spare (user-defined) (4) | Register Read Command | | 01 | | User-Defined Registers(00) | Parity Error |
| 17090 | 11:39:20.692... | Spare (user-defined) (4) | Register Write Command | | 01 | | User-Defined Registers(04) | |
| 17091 | 11:39:20.692... | Spare (user-defined) (4) | Extended Register Write Command | 0 | CF | | User-Defined Registers(00) | |
| 17092 | 11:39:20.692... | Spare (user-defined) (4) | Extended Register Write Command | 0 | CC | | User-Defined Registers(00) | |
| 17093 | 11:39:20.692... | Spare (user-defined) (4) | Register 0 Write Command | | | | 02 | |
| 17094 | 11:39:20.692... | Spare (user-defined) (4) | Register Read Command | | 01 | | User-Defined Registers(00) | Parity Error |
| 17095 | 11:39:20.692... | Spare (user-defined) (4) | Register Write Command | | 01 | | User-Defined Registers(04) | |
| 17096 | 11:39:20.692... | Spare (user-defined) (4) | Extended Register Write Command | 0 | CF | | User-Defined Registers(00) | |
| 17097 | 11:39:20.692... | Spare (user-defined) (4) | Extended Register Write Command | 0 | CC | | User-Defined Registers(00) | |
| 17098 | 11:39:20.692... | Spare (user-defined) (4) | Register 0 Write Command | | | | 02 | |
| 17099 | 11:39:20.692... | Spare (user-defined) (4) | Register Read Command | | 01 | | User-Defined Registers(00) | Parity Error |
| 17100 | 11:39:20.692... | Spare (user-defined) (4) | Register Write Command | | 01 | | User-Defined Registers(04) | |
| 17101 | 11:39:20.692... | Spare (user-defined) (4) | Extended Register Write Command | 0 | CF | | User-Defined Registers(00) | |
| 17102 | 11:39:20.692... | Spare (user-defined) (4) | Extended Register Write Command | 0 | CC | | User-Defined Registers(00) | |
| 17103 | 11:39:20.692... | Spare (user-defined) (4) | Register 0 Write Command | | | | 02 | |
| 17104 | 11:39:20.692... | Spare (user-defined) (4) | Register Read Command | | 01 | | User-Defined Registers(00) | Parity Error |
| 17105 | 11:39:20.692... | Spare (user-defined) (4) | Register Write Command | | 01 | | User-Defined Registers(04) | |
| 17106 | 11:39:20.692... | Spare (user-defined) (4) | Extended Register Write Command | 0 | CF | | User-Defined Registers(00) | |
| 17107 | 11:39:20.692... | Spare (user-defined) (4) | Extended Register Write Command | 0 | CC | | User-Defined Registers(00) | |
| 17108 | 11:39:20.692... | Spare (user-defined) (4) | Register 0 Write Command | | | | 02 | |

## MIPI SoundWire

Soundwire is a hardware interface and transport protocol developed by MIPI. It provides an expandable, simple, low-power, low-latency, dual-lead (clock and data) bus that can be used to transfer multiple audio streams such as amplifiers and microphones and embedded control commands.

Max Clock Rate:12.288MHz

## Settings



MIPI Soundwire Settings

Channel
Clock: A0
Data: A1

Range
Decode Range
From: Buffer Head
To: Buffer Tail

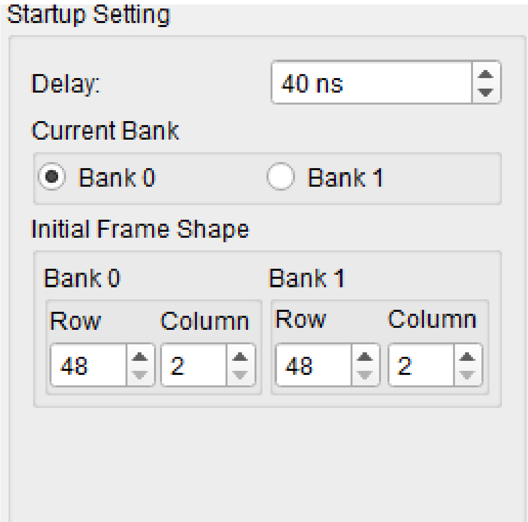
Startup Setting
Delay: 40 ns
Current Bank: ● Bank 0 ○ Bank 1
Initial Frame Shape
Bank 0: Row 48, Column 2
Bank 1: Row 48, Column 2

Report Settings
□ Show Packet Table
□ Hide all Ping OpCode
□ Ignore SCP Frame Ctrl CMD

□ Enable PayLoad
Type: PCM
Bank: Bank 0
Device: Device 0
DPn: DP0
HStart: 0
HStop: 0
Block Packing Mode: Block-per-Port
Port Flow Mode: Normal (isochronous)
Sample Interval: 96
Offset 1: 0
Offset 2: 0
Word Length: 0
Audio Sample Rate: 4000 Hz
PDM Sample Rate: 4800 KHz
□ CH1 □ CH2 □ CH3 □ CH4
□ CH5 □ CH6 □ CH7 □ CH8
□ Save as .wav
○ Full Scale
● Original
□ Display Audio
○ Full Scale
● Original
□ Save as .txt
□ Playback
● All
○ 5 Sec
○ 3 Sec

**CLK:** Transfer clock of SoundWire.

**Data:** Transfer data of SoundWire.



Startup Setting

Delay: 40 ns

Current Bank

● Bank 0 ○ Bank 1

Initial Frame Shape

| Bank 0 | | Bank 1 | |
|---|---|---|---|
| Row | Column | Row | Column |
| 48 | 2 | 48 | 2 |

**Delay:** Fix the position of latching data.

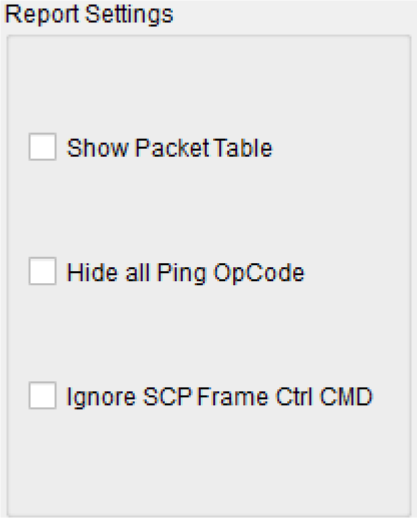
**Bank:** There are two types of parameters setting files, Bank0 and Bank1.

The Data port (DP) is the source or sink of the Payload Stream on the SoundWire bus, and the DP also divides the Payload Stream into one or more channels for each audio channel.

**Initial Frame Shape:** set the row and column of each setting file.

Column: 2~16(Only even)

Row: 48~256



Report Settings

□ Show Packet Table

□ Hide all Ping OpCode

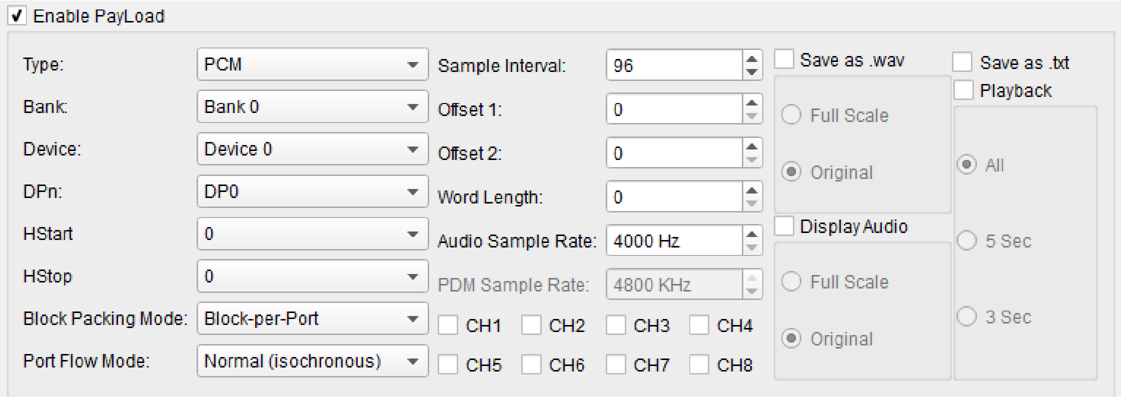
□ Ignore SCP Frame Ctrl CMD

**Show Packet Table:** Show Packet Table (in 2-Dimension). Enabled when checked.

**Hide all Ping OPCode:** Hide all the Ping OPCode in report. Enabled when checked.

**Ignore SCP Frame Ctrl CMD:** Hide all SCP CMD in report for easy reading.

Enabled when checked.



☑ Enable PayLoad

Type: PCM

Bank: Bank 0

Device: Device 0

DPn: DP0

HStart 0

HStop 0

Block Packing Mode: Block-per-Port

Port Flow Mode: Normal (isochronous)

Sample Interval: 96

Offset 1: 0

Offset 2: 0

Word Length: 0

Audio Sample Rate: 4000 Hz

PDM Sample Rate: 4800 KHz

☐ CH1 ☐ CH2 ☐ CH3 ☐ CH4

☐ CH5 ☐ CH6 ☐ CH7 ☐ CH8

☐ Save as .wav

○ Full Scale

● Original

☐ Display Audio

○ Full Scale

● Original

☐ Save as .txt

☐ Playback

● All

○ 5 Sec

○ 3 Sec

**DPn amount:** 1~16, No. DP0~DP15

1. **HStart:** The start position of data in Frame Shape.
2. **HStop:** The stop position of data in Frame Shape.
3. **Word Length:** Data length of each channel in DPn.
4. **Sample Interval:** DPn sample interval.
5. **Block Package Mode:**
   i. **Block per port:** Block Offset = Offset1 + (256 * Offset2)
   ii. **Block per channel:** Block Offset = Offset1，Sub-Block Offset = Offset2.
   iii. ***Block Offset range :** 0 ~ 65535
   iv. ***Sub-Block Offset range:** 0 to 255
   v. ***Offset1 range:** 0~65535
   vi. ***Offset2 range:** 0~255
6. **Channel:** Divide a Data Port into different parts, commonly used in the left channel, right channel and so on for data distribution, user can choose up to 8 channels to use, and do not need to follow the order of selection.
7. **Port Flow Mode:** it has four mode, Isochronous, Tx-Controlled, Rx-Controlled & Full-Asynchronous
   i. *Isochronous: ‘Normal’ mode, there is no valid data in each Payload Data Block.
   ii. *Tx-Controlled: ‘Push’ mode, whether the flow-control bit driven by the

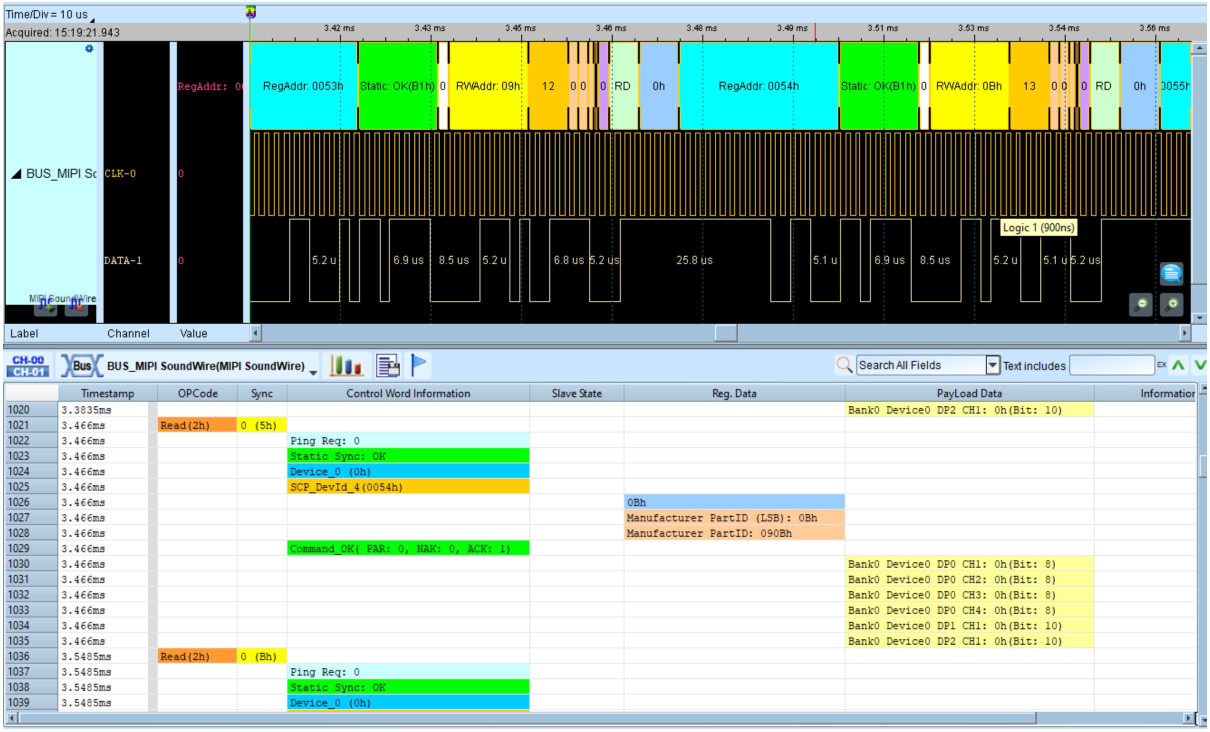
Source Data Port transmits valid data in the Payload Data Block.

iii. *Rx-Controlled: ‘Pull’ mode, whether the flow-control bit driven by the Sink Data Port transmits valid data in the Payload Data Block.

iv. *Full-Asynchronous: Whether the flow-control bit driven by the Sink and Source Data Port transmits valid data in the Payload Data Block.

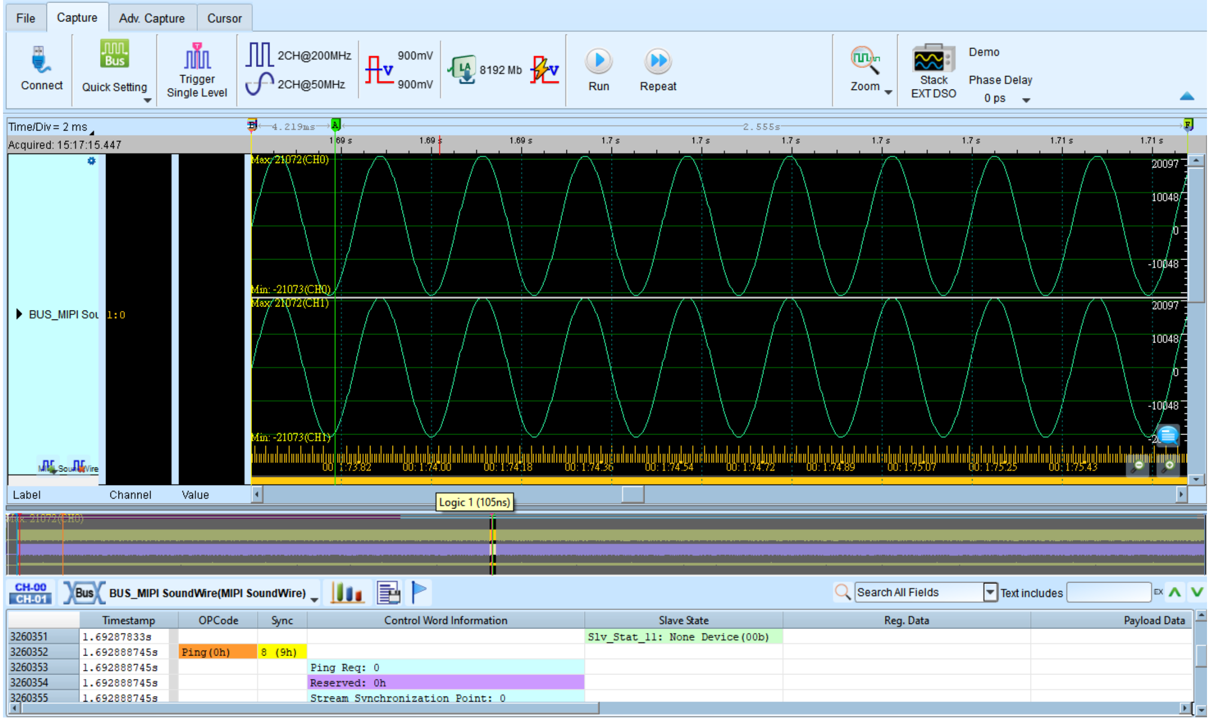
**Result:**

**Control Word + Report (Control Word & Payload):**



| | Timestamp | OPCode | Sync | Control Word Information | Slave State | Reg. Data | PayLoad Data | Information |
|---|---|---|---|---|---|---|---|---|
| 1020 | 3.3835ms | | | | | | Bank0 Device0 DP2 CH1: 0h(Bit: 10) | |
| 1021 | 3.466ms | Read(2h) | 0 (5h) | | | | | |
| 1022 | 3.466ms | | | Ping Req: 0 | | | | |
| 1023 | 3.466ms | | | Static Sync: OK | | | | |
| 1024 | 3.466ms | | | Device_0 (0h) | | | | |
| 1025 | 3.466ms | | | SCP_DevId_4(0054h) | | | | |
| 1026 | 3.466ms | | | | | 0Bh | | |
| 1027 | 3.466ms | | | | | Manufacturer PartID (LSB): 0Bh | | |
| 1028 | 3.466ms | | | | | Manufacturer PartID: 090Bh | | |
| 1029 | 3.466ms | | | Command_OK( PAR: 0, NAK: 0, ACK: 1) | | | | |
| 1030 | 3.466ms | | | | | | Bank0 Device0 DP0 CH1: 0h(Bit: 8) | |
| 1031 | 3.466ms | | | | | | Bank0 Device0 DP0 CH2: 0h(Bit: 8) | |
| 1032 | 3.466ms | | | | | | Bank0 Device0 DP0 CH3: 0h(Bit: 8) | |
| 1033 | 3.466ms | | | | | | Bank0 Device0 DP0 CH4: 0h(Bit: 8) | |
| 1034 | 3.466ms | | | | | | Bank0 Device0 DP1 CH1: 0h(Bit: 10) | |
| 1035 | 3.466ms | | | | | | Bank0 Device0 DP2 CH1: 0h(Bit: 10) | |
| 1036 | 3.5485ms | Read(2h) | 0 (Bh) | | | | | |
| 1037 | 3.5485ms | | | Ping Req: 0 | | | | |
| 1038 | 3.5485ms | | | Static Sync: OK | | | | |
| 1039 | 3.5485ms | | | Device_0 (0h) | | | | |

## Audio Display + Report (Control Word & Payload)

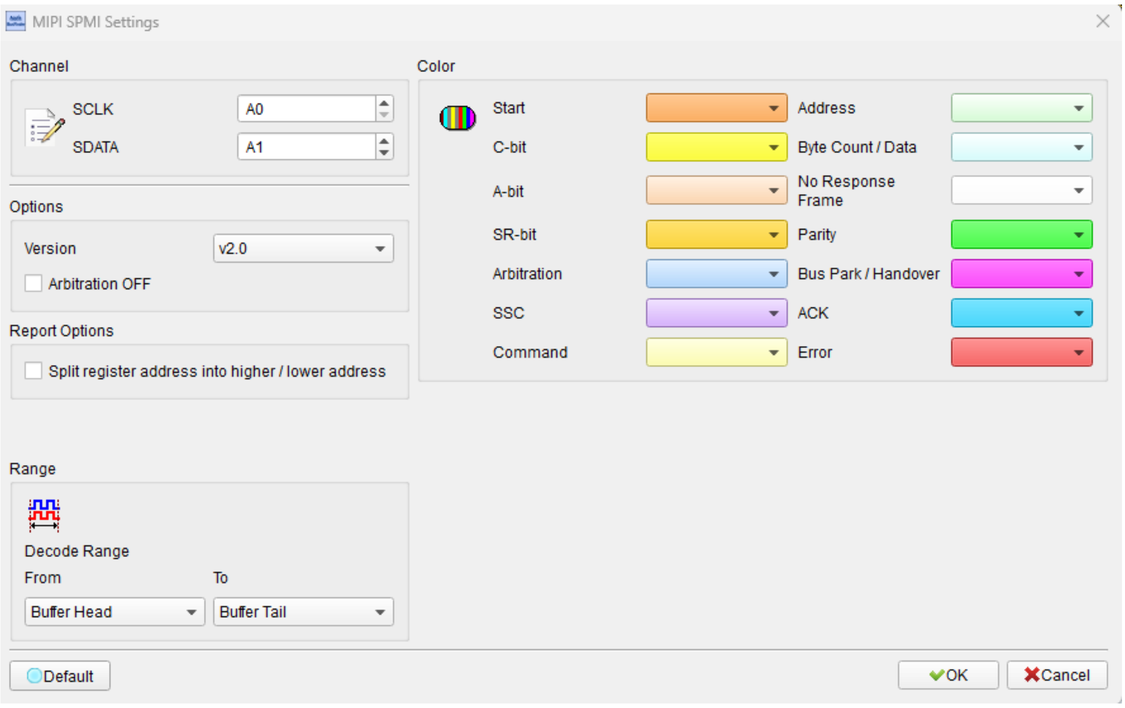


| | Timestamp | OPCode | Sync | Control Word Information | Slave State | Reg. Data | Payload Data |
|---|---|---|---|---|---|---|---|
| 3260351 | 1.69287833s | | | | Slv_Stat_11: None Device(00b) | | |
| 3260352 | 1.692888745s | Ping(0h) | 8 (9h) | | | | |
| 3260353 | 1.692888745s | | | Ping Req: 0 | | | |
| 3260354 | 1.692888745s | | | Reserved: 0h | | | |
| 3260355 | 1.692888745s | | | Stream Synchronization Point: 0 | | | |

## MIPI SPMI

MIPI SPMI(System Power Management Interface) designed by MIPI alliance. SPMI is a serial interface that connects the integrated Power Controller(PC) with Power management Integrated Circuits(PMIC).

### Settings



MIPI SPMI Settings

Channel

| | |
|---|---|
| SCLK | A0 |
| SDATA | A1 |

Options

Version: v2.0

□ Arbitration OFF

Report Options

□ Split register address into higher / lower address

Color

| | | | |
|---|---|---|---|
| Start | 🟧 | Address | 🟩 |
| C-bit | 🟨 | Byte Count / Data | 🟦 |
| A-bit | 🟧 | No Response Frame | ⬜ |
| SR-bit | 🟨 | Parity | 🟩 |
| Arbitration | 🟦 | Bus Park / Handover | 🟪 |
| SSC | 🟪 | ACK | 🟦 |
| Command | 🟨 | Error | 🟥 |

Range

Decode Range

From: Buffer Head
To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

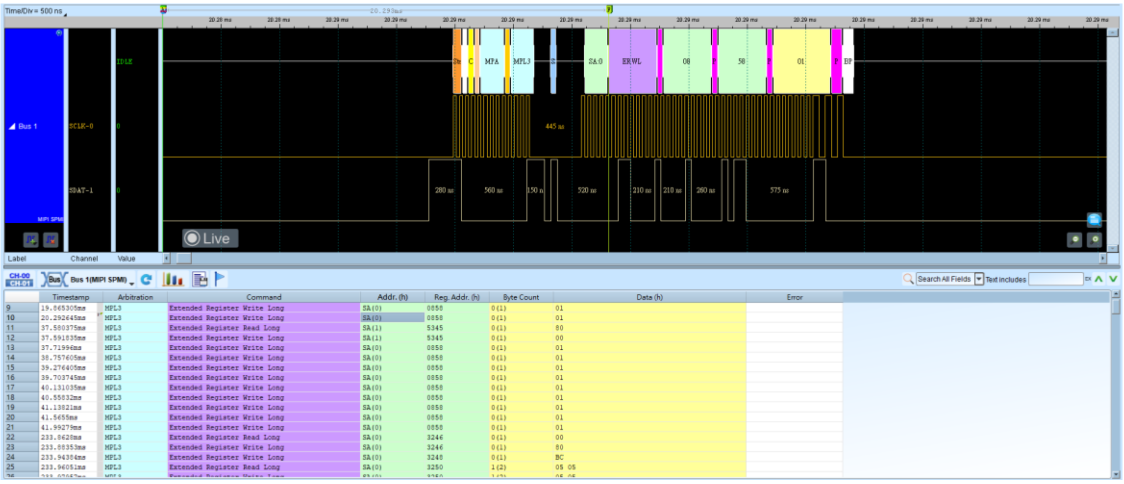
**Options:**

**Version:** Choose the version of MIPI SPMI. V2.0 and V1.0 are allowed to be chosen.

**Arbitration OFF:** Means there is no arbitration on the bus. Enabled when checked.

**Report Options:** Split register address into higher and lower address in report. Enabled when checked.

## Result

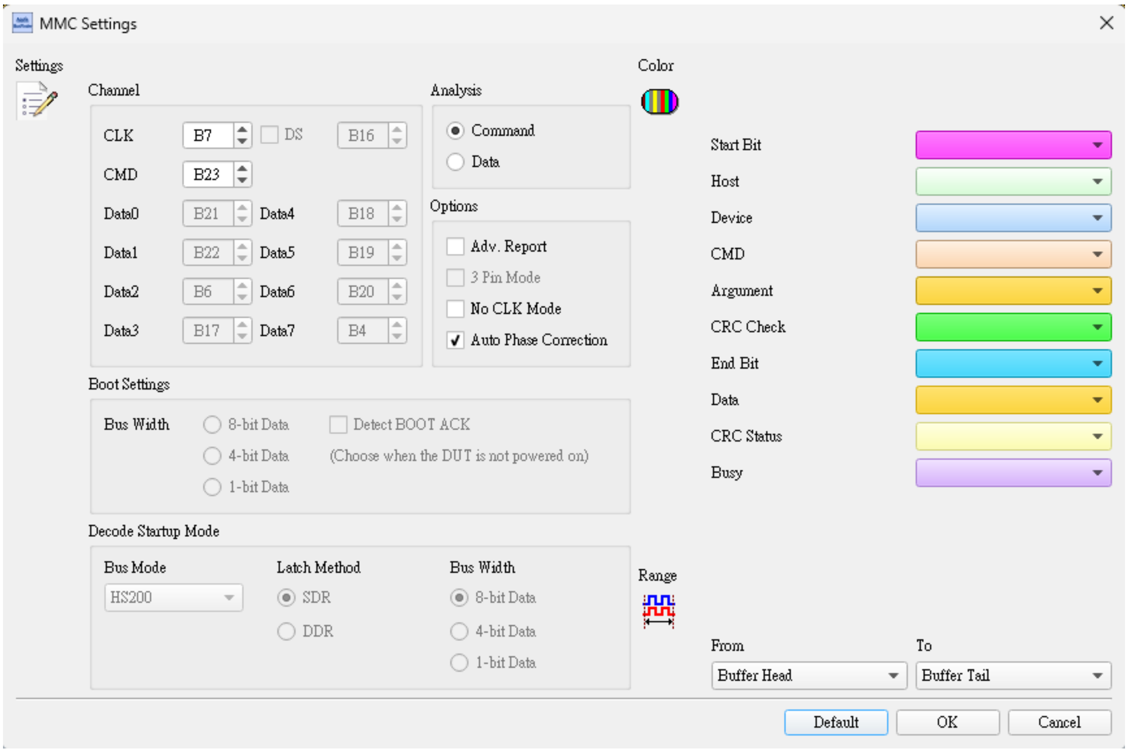


| | Timestamp | Arbitration | Command | Addr. (h) | Reg. Addr. (h) | Byte Count | Data (h) | Error |
|---|---|---|---|---|---|---|---|---|
| 9 | 19.865305ms | MPL3 | Extended Register Write Long | SA(0) | 0858 | 0(1) | 01 | |
| 10 | 20.292645ms | MPL3 | Extended Register Write Long | SA(0) | 0858 | 0(1) | 01 | |
| 11 | 37.580375ms | MPL3 | Extended Register Read Long | SA(1) | 5345 | 0(1) | 80 | |
| 12 | 37.591835ms | MPL3 | Extended Register Write Long | SA(1) | 5345 | 0(1) | 00 | |
| 13 | 37.71996ms | MPL3 | Extended Register Write Long | SA(0) | 0858 | 0(1) | 01 | |
| 14 | 38.757605ms | MPL3 | Extended Register Write Long | SA(0) | 0858 | 0(1) | 01 | |
| 15 | 39.276405ms | MPL3 | Extended Register Write Long | SA(0) | 0858 | 0(1) | 01 | |
| 16 | 39.703745ms | MPL3 | Extended Register Write Long | SA(0) | 0858 | 0(1) | 01 | |
| 17 | 40.131035ms | MPL3 | Extended Register Write Long | SA(0) | 0858 | 0(1) | 01 | |
| 18 | 40.55832ms | MPL3 | Extended Register Write Long | SA(0) | 0858 | 0(1) | 01 | |
| 19 | 41.13821ms | MPL3 | Extended Register Write Long | SA(0) | 0858 | 0(1) | 01 | |
| 20 | 41.5655ms | MPL3 | Extended Register Write Long | SA(0) | 0858 | 0(1) | 01 | |
| 21 | 41.99279ms | MPL3 | Extended Register Write Long | SA(0) | 0858 | 0(1) | 01 | |
| 22 | 233.8628ms | MPL3 | Extended Register Read Long | SA(0) | 3246 | 0(1) | 00 | |
| 23 | 233.88353ms | MPL3 | Extended Register Write Long | SA(0) | 3246 | 0(1) | 80 | |
| 24 | 233.94384ms | MPL3 | Extended Register Write Long | SA(0) | 3248 | 0(1) | BC | |
| 25 | 233.96051ms | MPL3 | Extended Register Read Long | SA(0) | 3250 | 1(2) | 05 05 | |

## MMC

The Multi Media Card (MMC) or the Embedded Multi Media Card (eMMC) version 5.1 is a flash memory card standard.

### Settings



MMC Settings

Settings

Channel

| | | | | |
|---|---|---|---|---|
| CLK | B7 | □ DS | B16 | |
| CMD | B23 | | | |
| Data0 | B21 | Data4 | B18 | |
| Data1 | B22 | Data5 | B19 | |
| Data2 | B6 | Data6 | B20 | |
| Data3 | B17 | Data7 | B4 | |

Analysis
● Command
○ Data

Options
□ Adv. Report
□ 3 Pin Mode
□ No CLK Mode
☑ Auto Phase Correction

Boot Settings
Bus Width: ○ 8-bit Data ○ 4-bit Data ○ 1-bit Data
□ Detect BOOT ACK
(Choose when the DUT is not powered on)

Decode Startup Mode
Bus Mode: HS200
Latch Method: ● SDR ○ DDR
Bus Width: ● 8-bit Data ○ 4-bit Data ○ 1-bit Data

Color

| | |
|---|---|
| Start Bit | 🟪 |
| Host | 🟩 |
| Device | 🟦 |
| CMD | 🟧 |
| Argument | 🟨 |
| CRC Check | 🟩 |
| End Bit | 🟦 |
| Data | 🟨 |
| CRC Status | 🟨 |
| Busy | 🟪 |

Range
From: Buffer Head
To: Buffer Tail

**Channel:** Set the used LA Channels which are connected with DUT.

**Analysis:** **(Combine with the feature – Customized Report)**

Command: Analysis Command only.

Data: Analysis Data only.

**Command:** **Adv. feature**

Adv. Report: Advance report on CMD, DATA argument.

3Pin mode: Decode with CLK, CMD, D0

No CLK mode: Only use the CLK to decode

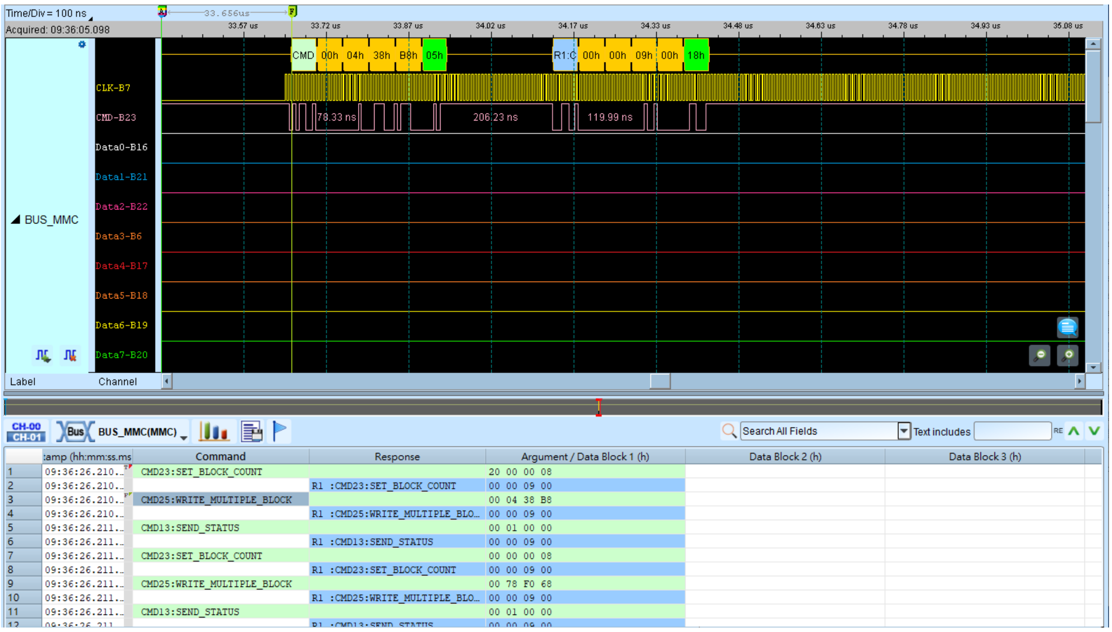
**Startup mode:** **When start to decode, set the DUT. status**

Need to set the DUT. Status. Including 1-8 bit data, DDR mode, Data Strobe and no BOOT ACK.

**Auto Phase Correction:** If it’s checked, the decoder will auto adjust the phase of LA when measure the DUT..

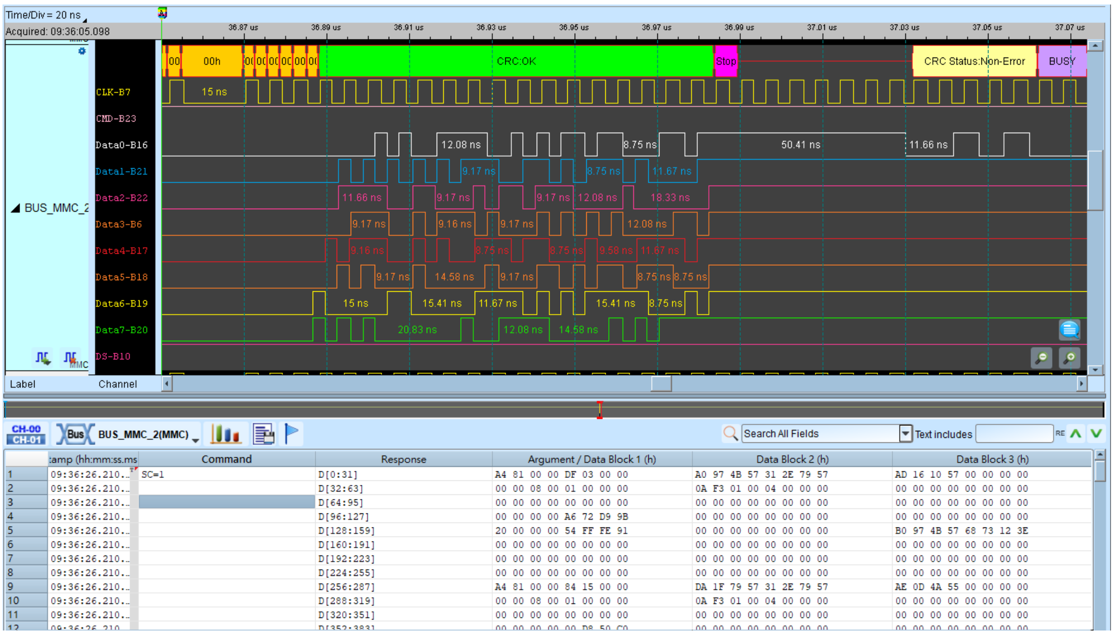
**Result**

Command:



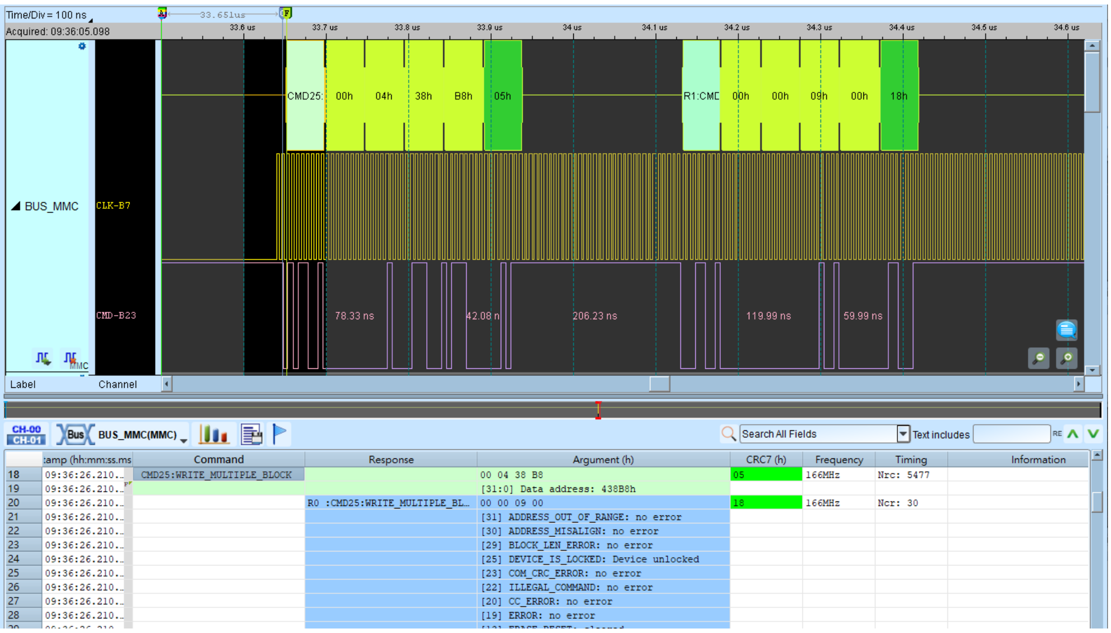
| | tamp (hh:mm:ss.ms | Command | Response | Argument / Data Block 1 (h) | Data Block 2 (h) | Data Block 3 (h) |
|---|---|---|---|---|---|---|
| 1 | 09:36:26.210... | CMD23:SET_BLOCK_COUNT | | 20 00 00 08 | | |
| 2 | 09:36:26.210... | | R1 :CMD23:SET_BLOCK_COUNT | 00 00 09 00 | | |
| 3 | 09:36:26.210... | CMD25:WRITE_MULTIPLE_BLOCK | | 00 04 38 B8 | | |
| 4 | 09:36:26.210... | | R1 :CMD25:WRITE_MULTIPLE_BLO... | 00 00 09 00 | | |
| 5 | 09:36:26.211... | CMD13:SEND_STATUS | | 00 01 00 00 | | |
| 6 | 09:36:26.211... | | R1 :CMD13:SEND_STATUS | 00 00 09 00 | | |
| 7 | 09:36:26.211... | CMD23:SET_BLOCK_COUNT | | 00 00 00 08 | | |
| 8 | 09:36:26.211... | | R1 :CMD23:SET_BLOCK_COUNT | 00 00 09 00 | | |
| 9 | 09:36:26.211... | CMD25:WRITE_MULTIPLE_BLOCK | | 00 78 F0 68 | | |
| 10 | 09:36:26.211... | | R1 :CMD25:WRITE_MULTIPLE_BLO... | 00 00 09 00 | | |
| 11 | 09:36:26.211... | CMD13:SEND_STATUS | | 00 01 00 00 | | |
| 12 | 09:36:26.211 | | R1 :CMD13:SEND_STATUS | 00 00 09 00 | | |

Data:



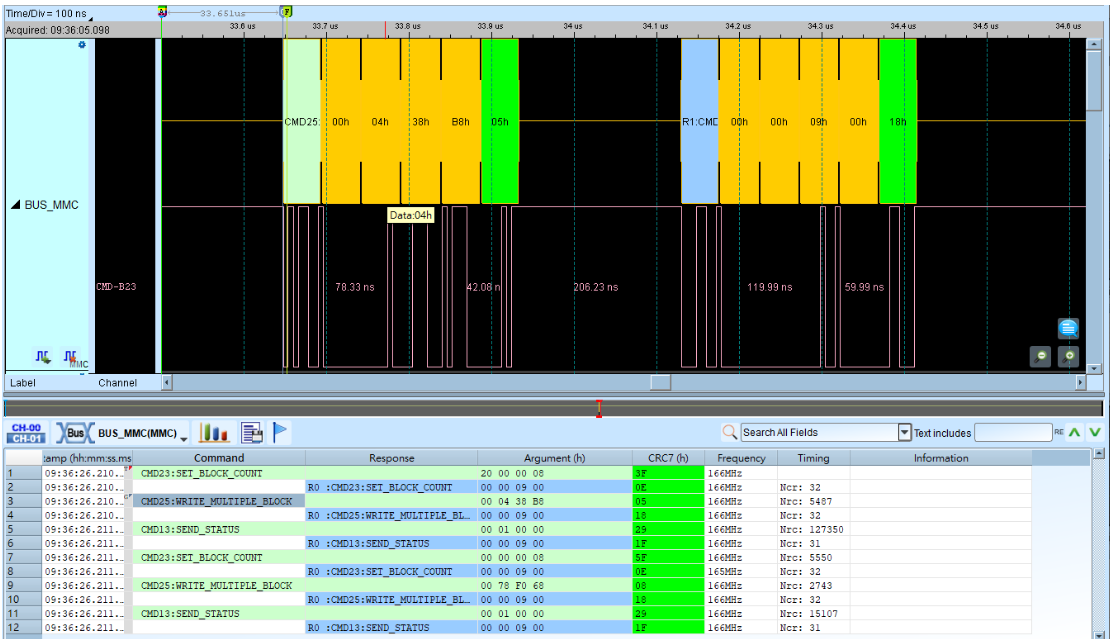
| | tamp (hh:mm:ss.ms | Command | Response | Argument / Data Block 1 (h) | Data Block 2 (h) | Data Block 3 (h) |
|---|---|---|---|---|---|---|
| 1 | 09:36:26.210... | SC=1 | D[0:31] | A4 81 00 00 DF 03 00 00 | A0 97 4B 57 31 2E 79 57 | AD 16 10 57 00 00 00 00 |
| 2 | 09:36:26.210... | | D[32:63] | 00 00 08 00 01 00 00 00 | 0A F3 01 00 04 00 00 00 | 00 00 00 00 00 00 00 00 |
| 3 | 09:36:26.210... | | D[64:95] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |
| 4 | 09:36:26.210... | | D[96:127] | 00 00 00 00 A6 72 D9 9B | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |
| 5 | 09:36:26.210... | | D[128:159] | 20 00 00 00 54 FF FE 91 | 00 00 00 00 00 00 00 00 | B0 97 4B 57 68 73 12 3E |
| 6 | 09:36:26.210... | | D[160:191] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |
| 7 | 09:36:26.210... | | D[192:223] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |
| 8 | 09:36:26.210... | | D[224:255] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |
| 9 | 09:36:26.210... | | D[256:287] | A4 81 00 00 84 15 00 00 | DA 1F 79 57 31 2E 79 57 | AE 0D 4A 55 00 00 00 00 |
| 10 | 09:36:26.210... | | D[288:319] | 00 00 08 00 01 00 00 00 | 0A F3 01 00 04 00 00 00 | 00 00 00 00 00 00 00 00 |
| 11 | 09:36:26.210... | | D[320:351] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |
| 12 | 09:36:26.210 | | D[352:383] | 00 00 00 00 00 D8 50 C0 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |

Adv. Report:



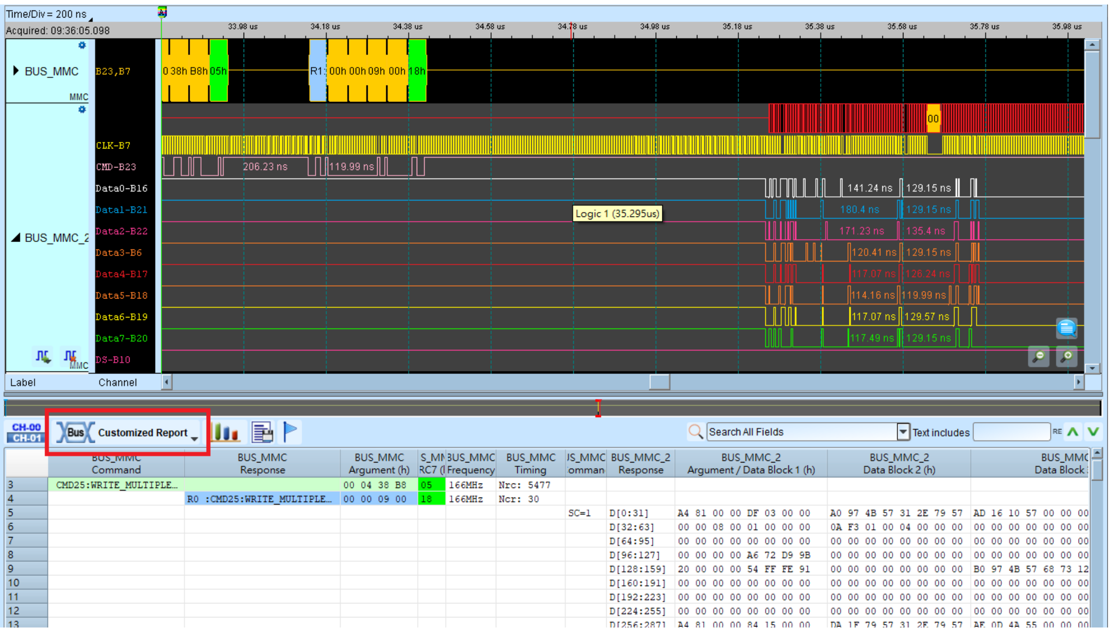
| | tamp (hh:mm:ss.ms | Command | Response | Argument (h) | CRC7 (h) | Frequency | Timing | Information |
|---|---|---|---|---|---|---|---|---|
| 18 | 09:36:26.210... | CMD25:WRITE_MULTIPLE_BLOCK | | 00 04 38 B8 | 05 | 166MHz | Nrc: 5477 | |
| 19 | 09:36:26.210... | | | [31:0] Data address: 438B8h | | | | |
| 20 | 09:36:26.210... | | R0 :CMD25:WRITE_MULTIPLE_BL... | 00 00 09 00 | 18 | 166MHz | Ncr: 30 | |
| 21 | 09:36:26.210... | | | [31] ADDRESS_OUT_OF_RANGE: no error | | | | |
| 22 | 09:36:26.210... | | | [30] ADDRESS_MISALIGN: no error | | | | |
| 23 | 09:36:26.210... | | | [29] BLOCK_LEN_ERROR: no error | | | | |
| 24 | 09:36:26.210... | | | [25] DEVICE_IS_LOCKED: Device unlocked | | | | |
| 25 | 09:36:26.210... | | | [23] COM_CRC_ERROR: no error | | | | |
| 26 | 09:36:26.210... | | | [22] ILLEGAL_COMMAND: no error | | | | |
| 27 | 09:36:26.210... | | | [20] CC_ERROR: no error | | | | |
| 28 | 09:36:26.210... | | | [19] ERROR: no error | | | | |

No CLK mode



| | tamp (hh:mm:ss.ms | Command | Response | Argument (h) | CRC7 (h) | Frequency | Timing | Information |
|---|---|---|---|---|---|---|---|---|
| 1 | 09:36:26.210... | CMD23:SET_BLOCK_COUNT | | 20 00 00 08 | 3F | 166MHz | | |
| 2 | 09:36:26.210... | | R0 :CMD23:SET_BLOCK_COUNT | 00 00 09 00 | 0E | 166MHz | Ncr: 32 | |
| 3 | 09:36:26.210... | CMD25:WRITE_MULTIPLE_BLOCK | | 00 04 38 B8 | 05 | 166MHz | Nrc: 5487 | |
| 4 | 09:36:26.210... | | R0 :CMD25:WRITE_MULTIPLE_BL... | 00 00 09 00 | 18 | 166MHz | Ncr: 32 | |
| 5 | 09:36:26.211... | CMD13:SEND_STATUS | | 00 01 00 00 | 29 | 166MHz | Nrc: 127350 | |
| 6 | 09:36:26.211... | | R0 :CMD13:SEND_STATUS | 00 00 09 00 | 1F | 166MHz | Ncr: 31 | |
| 7 | 09:36:26.211... | CMD23:SET_BLOCK_COUNT | | 00 00 00 08 | 5F | 166MHz | Nrc: 5550 | |
| 8 | 09:36:26.211... | | R0 :CMD23:SET_BLOCK_COUNT | 00 00 09 00 | 0E | 165MHz | Ncr: 32 | |
| 9 | 09:36:26.211... | CMD25:WRITE_MULTIPLE_BLOCK | | 00 78 F0 68 | 08 | 166MHz | Nrc: 2743 | |
| 10 | 09:36:26.211... | | R0 :CMD25:WRITE_MULTIPLE_BL... | 00 00 09 00 | 18 | 166MHz | Ncr: 32 | |
| 11 | 09:36:26.211... | CMD13:SEND_STATUS | | 00 01 00 00 | 29 | 166MHz | Nrc: 15107 | |
| 12 | 09:36:26.211... | | R0 :CMD13:SEND_STATUS | 00 00 09 00 | 1F | 166MHz | Ncr: 31 | |

## Command + Data: (Customized Report)



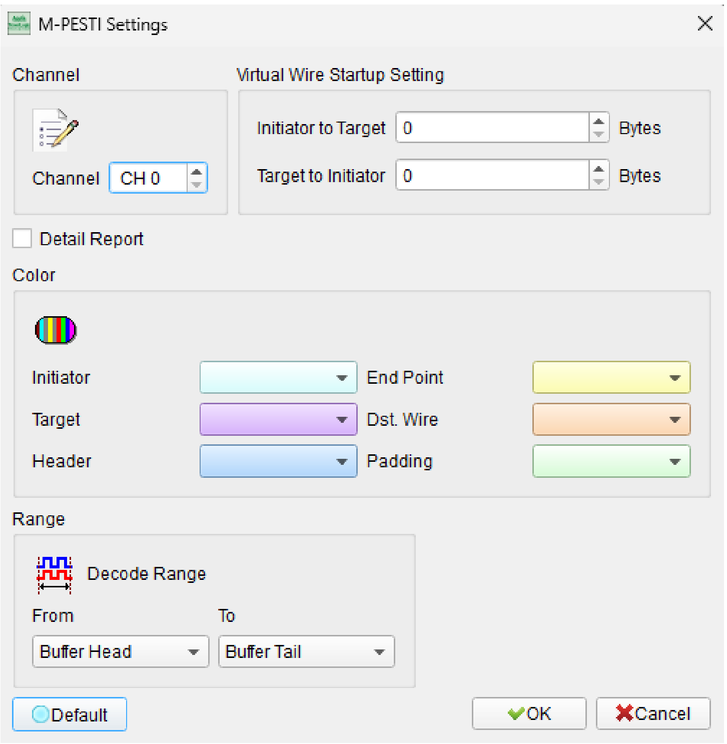
| | BUS_MMC<br>Command | BUS_MMC<br>Response | BUS_MMC<br>Argument (h) | S_MI<br>RC7 (I | BUS_MMC<br>Frequency | BUS_MMC<br>Timing | JS_MMC<br>omman | BUS_MMC_2<br>Response | BUS_MMC_2<br>Argument / Data Block 1 (h) | BUS_MMC_2<br>Data Block 2 (h) | BUS_MMC<br>Data Block : |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | CMD25:WRITE_MULTIPLE… | | 00 04 38 B8 | 05 | 166MHz | Nrc: 5477 | | | | | |
| 4 | | R0 :CMD25:WRITE_MULTIPLE… | 00 00 09 00 | 18 | 166MHz | Ncr: 30 | | | | | |
| 5 | | | | | | | SC=1 | D[0:31] | A4 81 00 00 DF 03 00 00 | A0 97 4B 57 31 2E 79 57 | AD 16 10 57 00 00 00 |
| 6 | | | | | | | | D[32:63] | 00 00 08 00 01 00 00 00 | 0A F3 01 00 04 00 00 00 | 00 00 00 00 00 00 00 |
| 7 | | | | | | | | D[64:95] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 |
| 8 | | | | | | | | D[96:127] | 00 00 00 00 A6 72 D9 9B | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 |
| 9 | | | | | | | | D[128:159] | 20 00 00 00 54 FF FE 91 | 00 00 00 00 00 00 00 00 | B0 97 4B 57 68 73 12 |
| 10 | | | | | | | | D[160:191] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 |
| 11 | | | | | | | | D[192:223] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 |
| 12 | | | | | | | | D[224:255] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 |
| 13 | | | | | | | | D[256:287] | A4 81 00 00 84 15 00 00 | DA 1F 79 57 31 2E 79 57 | AE 0D 4A 55 00 00 00 |

## M-PESTI

M-PESTI stands for "Modular Peripheral Sideband Tunneling Interface". It is an IP (Intellectual Property) core designed for early detection and attribute collection of peripheral devices before system boot.

M-PESTI supports bidirectional communication based on an initiator-command and target-response structure.

### Settings



M-PESTI Settings

Channel

Channel CH 0

Virtual Wire Startup Setting

Initiator to Target 0 Bytes

Target to Initiator 0 Bytes

□ Detail Report

Color

| | | | |
|---|---|---|---|
| Initiator | | End Point | |
| Target | | Dst. Wire | |
| Header | | Padding | |

Range

Decode Range

From: Buffer Head

To: Buffer Tail

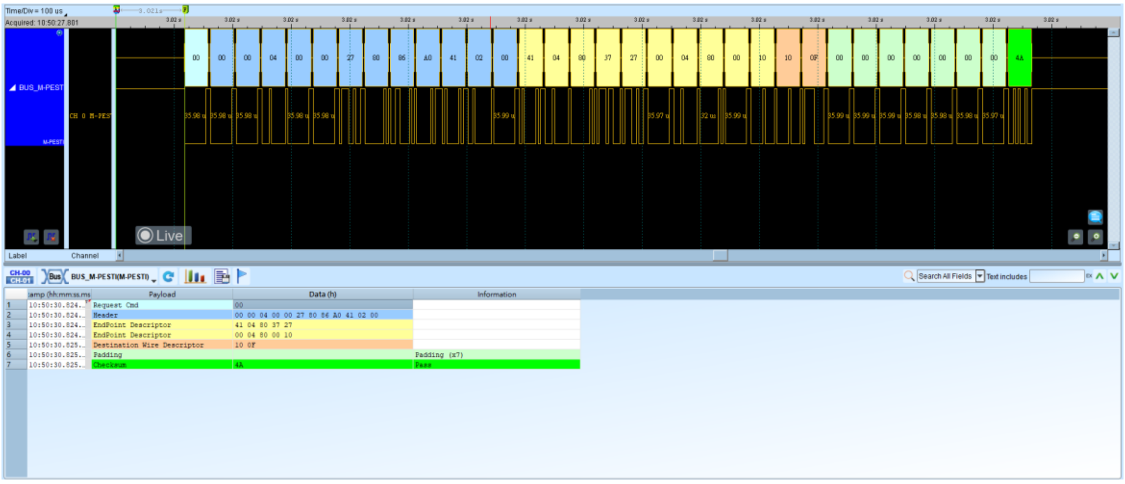
**Channel:** Set the signal channel of M-PESTI.

**Virtual Wire Startup Setting:**

1. **Initiator to Target:** Configure the number of bytes to be sent from the Initiator to the Target, excluding the Virtual Wire Command.
2. **Target to Initiator:** Configure the number of bytes to be returned from the Target to the Initiator.

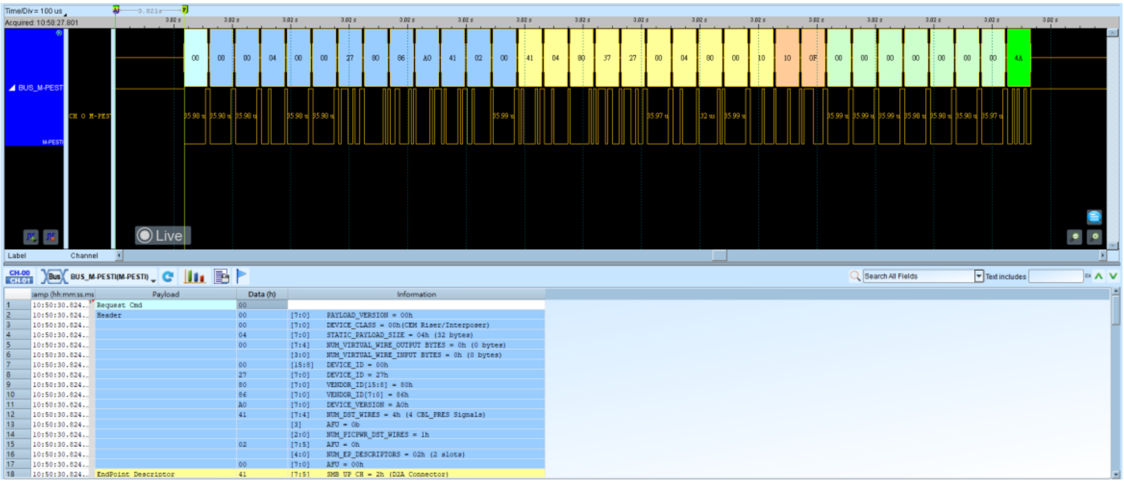
**Detail Report:** Detailed analysis of the byte content. Enable when checked.

## Result



| | amp (hh:mm:ss.ms | Payload | Data (h) | Information |
|---|---|---|---|---|
| 1 | 10:50:30.824... | Request Cmd | 00 | |
| 2 | 10:50:30.824... | Header | 00 00 04 00 00 27 80 86 A0 41 02 00 | |
| 3 | 10:50:30.824... | EndPoint Descriptor | 41 04 80 37 27 | |
| 4 | 10:50:30.824... | EndPoint Descriptor | 00 04 80 00 10 | |
| 5 | 10:50:30.825... | Destination Wire Descriptor | 10 0F | |
| 6 | 10:50:30.825... | Padding | | Padding (x7) |
| 7 | 10:50:30.825... | Checksum | 4A | Pass |

## Detail Report

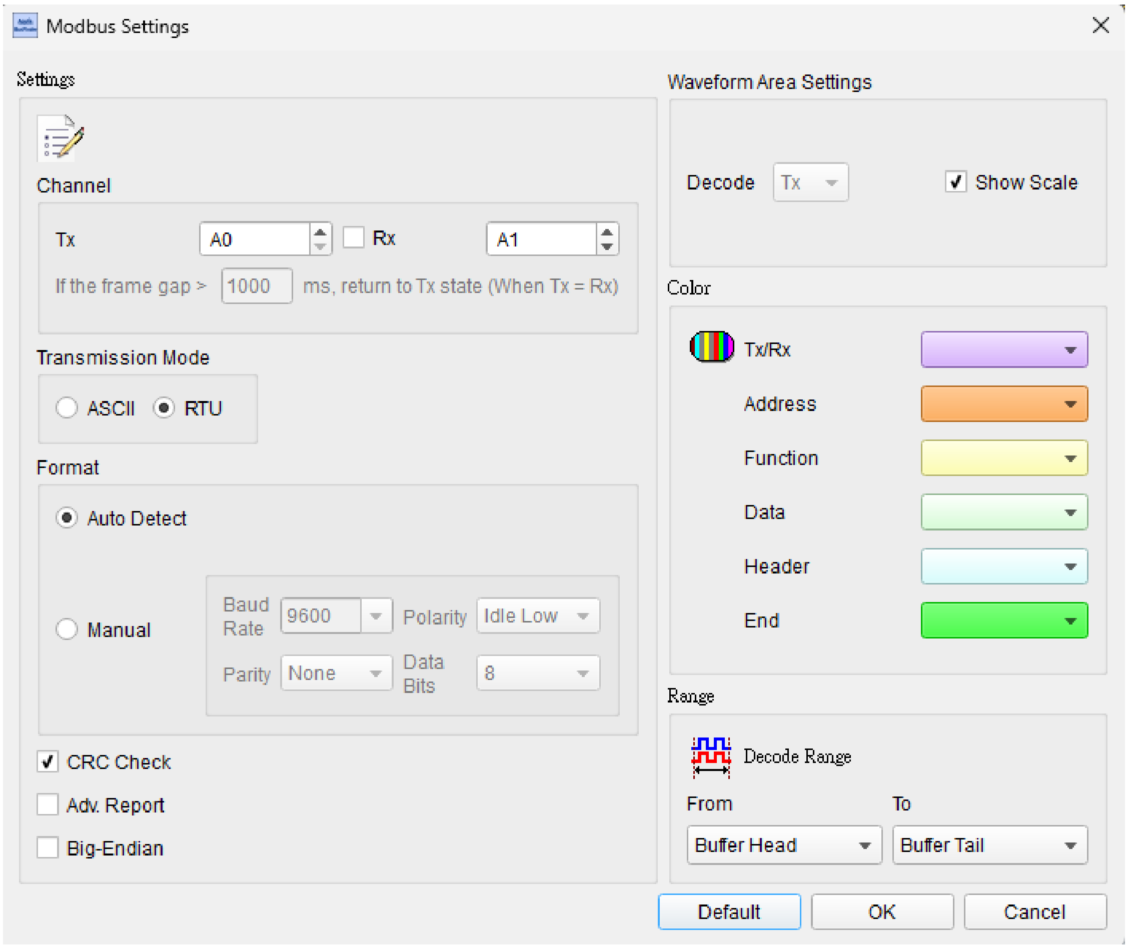


| | amp (hh:mm:ss.ms | Payload | Data (h) | Information |
|---|---|---|---|---|
| 1 | 10:50:30.824... | Request Cmd | 00 | |
| 2 | 10:50:30.824... | Header | 00 | [7:0] PAYLOAD_VERSION = 00h |
| 3 | 10:50:30.824... | | 00 | [7:0] DEVICE_CLASS = 00h(CEM Riser/Interposer) |
| 4 | 10:50:30.824... | | 04 | [7:0] STATIC_PAYLOAD_SIZE = 04h (32 bytes) |
| 5 | 10:50:30.824... | | 00 | [7:4] NUM_VIRTUAL_WIRE_OUTPUT BYTES = 0h (0 bytes) |
| 6 | 10:50:30.824... | | | [3:0] NUM_VIRTUAL_WIRE_INPUT BYTES = 0h (0 bytes) |
| 7 | 10:50:30.824... | | 00 | [15:8] DEVICE_ID = 00h |
| 8 | 10:50:30.824... | | 27 | [7:0] DEVICE_ID = 27h |
| 9 | 10:50:30.824... | | 80 | [7:0] VENDOR_ID[15:8] = 80h |
| 10 | 10:50:30.824... | | 86 | [7:0] VENDOR_ID[7:0] = 86h |
| 11 | 10:50:30.824... | | A0 | [7:0] DEVICE_VERSION = A0h |
| 12 | 10:50:30.824... | | 41 | [7:4] NUM_DST_WIRES = 4h (4 CBL_PRES Signals) |
| 13 | 10:50:30.824... | | | [3] AFU = 0b |
| 14 | 10:50:30.824... | | | [2:0] NUM_PICPWR_DST_WIRES = 1h |
| 15 | 10:50:30.824... | | 02 | [7:5] AFU = 0h |
| 16 | 10:50:30.824... | | | [4:0] NUM_EP_DESCRIPTORS = 02h (2 slots) |
| 17 | 10:50:30.824... | | 00 | [7:0] AFU = 00h |
| 18 | 10:50:30.824... | EndPoint Descriptor | 41 | [7:5] SMB UP CH = 2h (D2A Connector) |

## ModBus

Modbus is a serial communications protocol published by Modicon in 1979 for use with its programmable logic controllers (PLCs). Simple and robust, it has since become one of the standard communications protocols in the industry, and it is now amongst the most commonly available means of connecting industrial electronic devices.

## Settings



Modbus Settings

Settings

Channel

Tx A0 □ Rx A1

If the frame gap > 1000 ms, return to Tx state (When Tx = Rx)

Transmission Mode

○ ASCII ● RTU

Format

● Auto Detect

○ Manual

Baud Rate 9600 Polarity Idle Low

Parity None Data Bits 8

☑ CRC Check

□ Adv. Report

□ Big-Endian

Waveform Area Settings

Decode Tx ☑ Show Scale

Color

Tx/Rx 🟪

Address 🟧

Function 🟨

Data 🟩

Header 🟦

End 🟩

Range

Decode Range

From Buffer Head To Buffer Tail

### Channel:

1. **Tx:** Signal channel of Modbus Tx.
2. **Rx:** Signal channel of Modbus Rx.

**Transmission Mode:** It has ASCII and RTU mode.

**Format:**

1. **Auto Detect:** Auto detect the value of all the options that can adjust by user if it was checked.
2. **Manual:** User can manually adjust the options below:
   I. **Baud Rate:** Data rate (bits per second), and the range is 110 ~ 2M (bps).

   II. **Polarity:**
      - **Idle high:** Idle condition shows High.
      - **Idle low:** Idle condition shows Low.

   III. **Parity:** N-None Parity, O-Odd Parity, E-Even Parity.

   IV. **Data Bits:** Set the data bits. User can set to 7, 8 or 9.

**CRC Check:** Do CRC check. Enabled when checked.

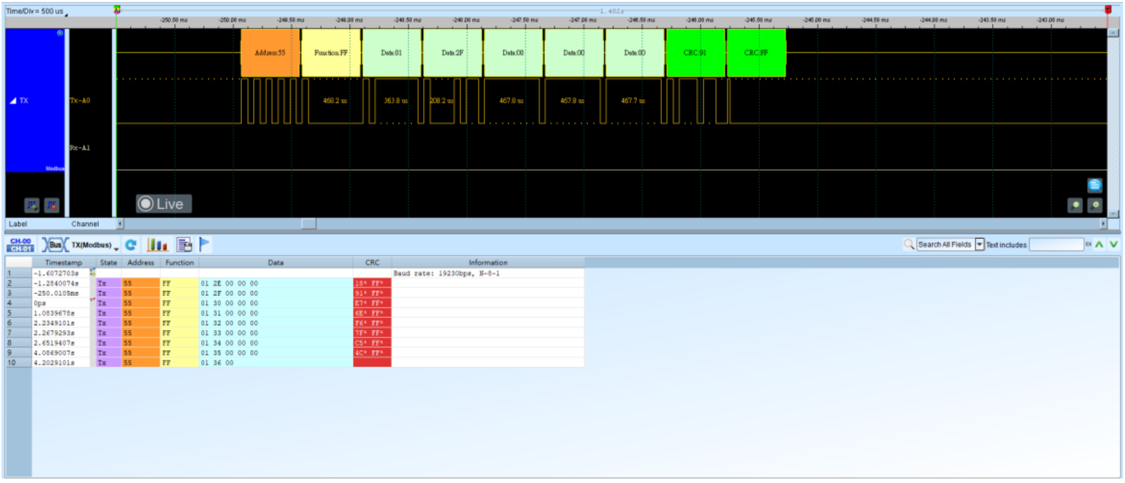
**Adv. Report:** Show the detail report in the report area. Enabled when checked.

**Big-Endian:** Data is arranged in Big-Endian style. Enabled when checked.

**Waveform Area Settings:**

1. **Decode:** Display the analyzed result of Tx or Rx in waveform area. Rx option only available when Rx is activated.
2. **Show Scale:** Display the waveforms with scales.

**Result**

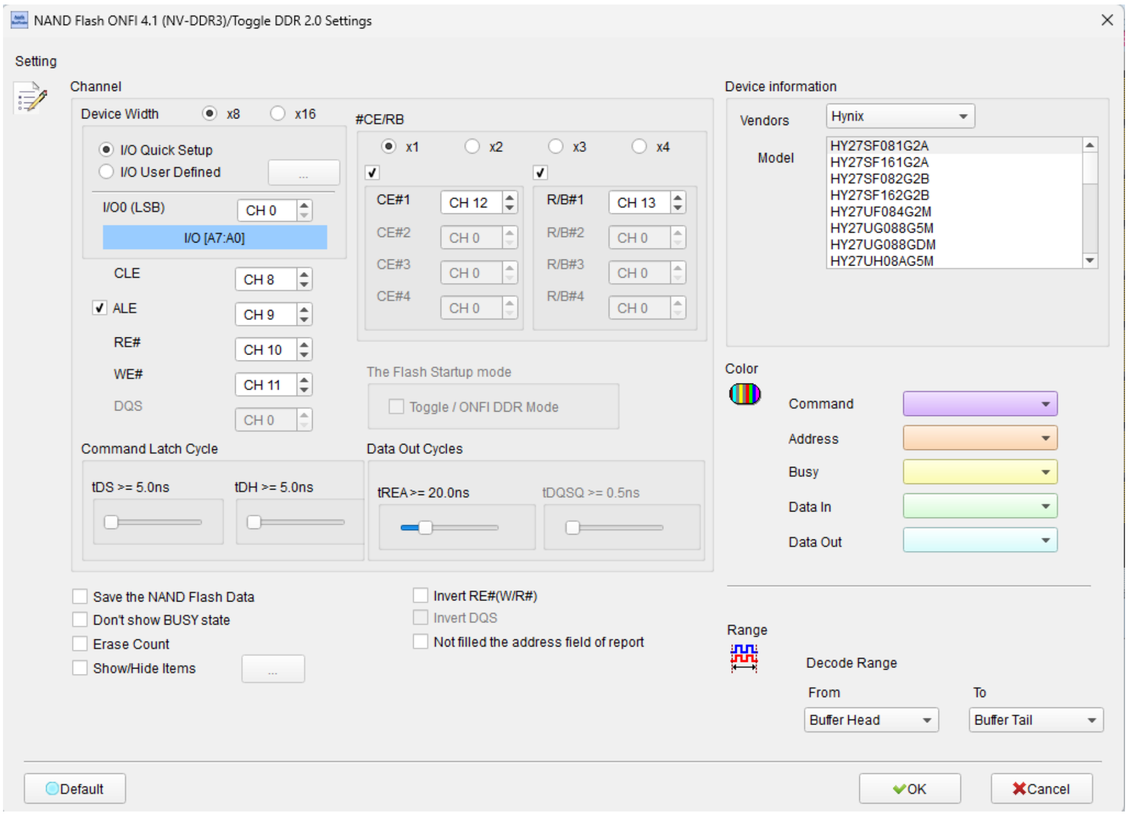


| | Timestamp | State | Address | Function | Data | CRC | Information |
|---|---|---|---|---|---|---|---|
| 1 | -1.6072703s | | | | | | Baud rate: 19230bps, N-8-1 |
| 2 | -1.2840074s | Tx | 55 | FF | 01 2E 00 00 00 | 18* FF* | |
| 3 | -250.0105ms | Tx | 55 | FF | 01 2F 00 00 00 | 91* FF* | |
| 4 | 0ps | Tx | 55 | FF | 01 30 00 00 00 | E7* FF* | |
| 5 | 1.0839678s | Tx | 55 | FF | 01 31 00 00 00 | 6E* FF* | |
| 6 | 2.2349101s | Tx | 55 | FF | 01 32 00 00 00 | F6* FF* | |
| 7 | 2.2679293s | Tx | 55 | FF | 01 33 00 00 00 | 7F* FF* | |
| 8 | 2.6519407s | Tx | 55 | FF | 01 34 00 00 00 | C5* FF* | |
| 9 | 4.0869007s | Tx | 55 | FF | 01 35 00 00 00 | 4C* FF* | |
| 10 | 4.2029101s | Tx | 55 | FF | 01 36 00 | | |

## NAND Flash

NAND flash uses tunnel injection for writing and tunnel release for erasing. NAND flash memory forms the core of the removable USB storage devices known as USB flash drives, as well as most memory card formats and solid-state drives available today.

### Settings



NAND Flash ONFI 4.1 (NV-DDR3)/Toggle DDR 2.0 Settings

Setting

Channel

Device Width ● x8 ○ x16

● I/O Quick Setup
○ I/O User Defined

I/O0 (LSB) CH 0
I/O [A7:A0]

CLE CH 8
☑ ALE CH 9
RE# CH 10
WE# CH 11
DQS CH 0

#CE/RB

● x1 ○ x2 ○ x3 ○ x4

☑ CE#1 CH 12
CE#2 CH 0
CE#3 CH 0
CE#4 CH 0

☑ R/B#1 CH 13
R/B#2 CH 0
R/B#3 CH 0
R/B#4 CH 0

The Flash Startup mode
□ Toggle / ONFI DDR Mode

Command Latch Cycle
tDS >= 5.0ns tDH >= 5.0ns

Data Out Cycles
tREA >= 20.0ns tDQSQ >= 0.5ns

□ Save the NAND Flash Data
□ Don't show BUSY state
□ Erase Count
□ Show/Hide Items

□ Invert RE#(W/R#)
□ Invert DQS
□ Not filled the address field of report

Device information

Vendors Hynix

Model
HY27SF081G2A
HY27SF161G2A
HY27SF082G2B
HY27SF162G2B
HY27UF084G2M
HY27UG088G5M
HY27UG088GDM
HY27UH08AG5M

Color
Command
Address
Busy
Data In
Data Out

Range
Decode Range
From Buffer Head
To Buffer Tail

### Channel:

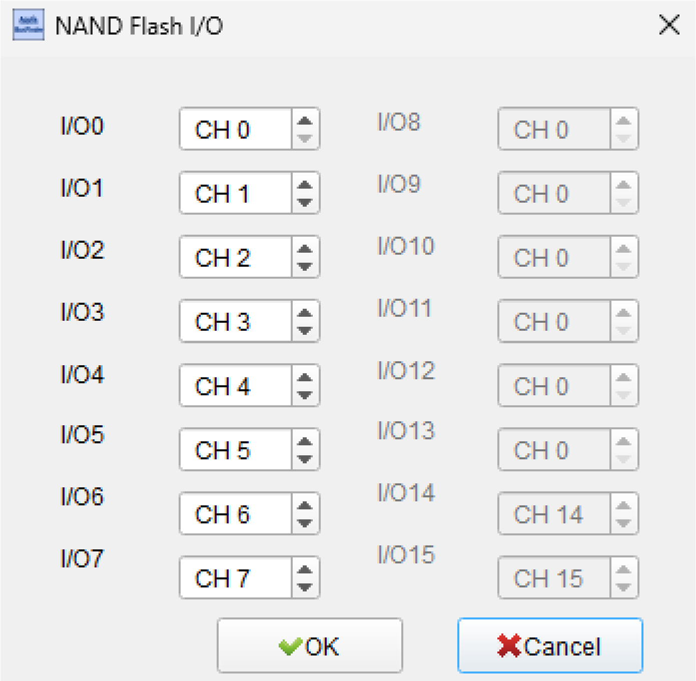
| Async | Ssync | Description |
|---|---|---|
| I/Ox | DQx | NAND Flash data channels |
| CLE | CLE | Command Latch Enable channel |
| ALE | ALE | Address Latch Enable channel |
| RE | W/R | Read Enable and Write/Read channel |
| WE | CLK | Write Enable and Clock channel |
| RB# | RB# | Ready/Busy channel |

| | | |
|---|---|---|
| CE# | CE# | Chip Enable channel |
| --- | DQS | Data Strobe channel |

**Device Width:** Select 8/16 bits device width.

**The Flash Startup mode**: Check Toggle /ONFI DDR Mode to run synchronous data interface.

**I/O Quick Setup / I/O User Defined**: Only set I/O0 (LSB) when select the I/O Quick Setup, other channels will be set automatically. when check the I/O User Defined and press the button will show the dialog below:



NAND Flash I/O

| | | | |
|---|---|---|---|
| I/O0 | CH 0 | I/O8 | CH 0 |
| I/O1 | CH 1 | I/O9 | CH 0 |
| I/O2 | CH 2 | I/O10 | CH 0 |
| I/O3 | CH 3 | I/O11 | CH 0 |
| I/O4 | CH 4 | I/O12 | CH 0 |
| I/O5 | CH 5 | I/O13 | CH 0 |
| I/O6 | CH 6 | I/O14 | CH 14 |
| I/O7 | CH 7 | I/O15 | CH 15 |

User can set NAND I/O channel by channel.

**The Flash Startup mode:** Check Toggle /ONFI DDR Mode to run synchronous data interface.

**tREA / tDQSQ:** Set the delay time to access the NAND data under SDR / DDR. To adjust the tREA/tDQSQ when the data out value of NAND Flash is invalid.

**Save the NAND Flash Data:** Save the read/write data. Program will save the NAND Flash read/write data as a file when check Save the NAND Flash Data. It will be saved into the LA work directory.

mode.

**Don’t show BUSY State**: Show/hide the BUSY state information (e.g. BUSY START / BUSY END) in the report window.

**Erase Count**: Show/ hide NAND Erase command/address statistics.

**Show/Hide Items**: Show/hide the items in the report window.

**Invert RE# (W/R#) / Invert DQS:** check this item when connect the RE/DQS# pin under DDR mode.

**Not Filled the address field of report:** Filled the NAND write/Read address column in the report window or not.

**Invert RE# (W/R#) / Invert DQS:** Check this when connect the RE / DQS# under DDR mode.

**Don’t care ALE/RB#/CE# signal:** Ignore the signal selected when decode.

Description of file name as following **(Save the NAND Flash Data function)**:

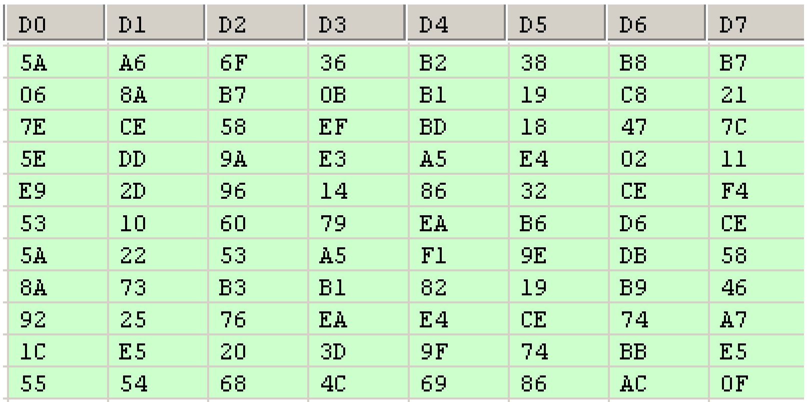
| File Name | Description |
|---|---|
| NF_DI/NF_DO | NAND Flash Data In / Data Out |
| _Rowxxxxxxh | Row Address |
| _Colxxxxh | Column Address |
| CEx | Active CEx |
| _1, _2, _3 ....... | File Order |

Ex:NF_DI_Row017821h_Col0000h_CE1_1.bin

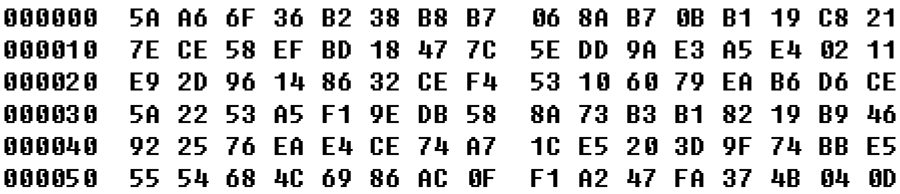
NF_DO_Row017821h_Col0000h_CE1_2.bin

NF_DO_Row_Col_CE1_3.bin

Compare the content of file with the one of report.



| D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 |
|---|---|---|---|---|---|---|---|
| 5A | A6 | 6F | 36 | B2 | 38 | B8 | B7 |
| 06 | 8A | B7 | 0B | B1 | 19 | C8 | 21 |
| 7E | CE | 58 | EF | BD | 18 | 47 | 7C |
| 5E | DD | 9A | E3 | A5 | E4 | 02 | 11 |
| E9 | 2D | 96 | 14 | 86 | 32 | CE | F4 |
| 53 | 10 | 60 | 79 | EA | B6 | D6 | CE |
| 5A | 22 | 53 | A5 | F1 | 9E | DB | 58 |
| 8A | 73 | B3 | B1 | 82 | 19 | B9 | 46 |
| 92 | 25 | 76 | EA | E4 | CE | 74 | A7 |
| 1C | E5 | 20 | 3D | 9F | 74 | BB | E5 |
| 55 | 54 | 68 | 4C | 69 | 86 | AC | 0F |



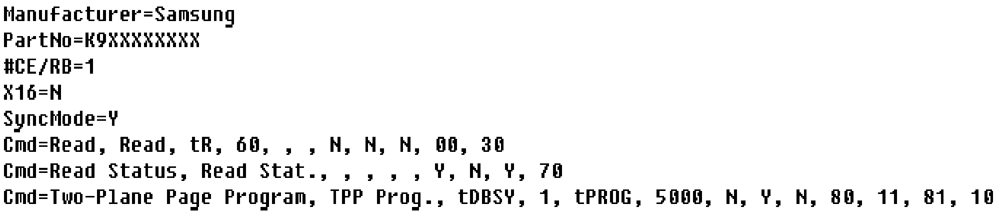
```
000000  5A A6 6F 36 B2 38 B8 B7  06 8A B7 0B B1 19 C8 21
000010  7E CE 58 EF BD 18 47 7C  5E DD 9A E3 A5 E4 02 11
000020  E9 2D 96 14 86 32 CE F4  53 10 60 79 EA B6 D6 CE
000030  5A 22 53 A5 F1 9E DB 58  8A 73 B3 B1 82 19 B9 46
000040  92 25 76 EA E4 CE 74 A7  1C E5 20 3D 9F 74 BB E5
000050  55 54 68 4C 69 86 AC 0F  F1 A2 47 FA 37 4B 04 0D
```

**Device information**

**Vendors**: Select the NAND Flash Vendor. Please refer to the following details when select the **Custom** item.

**Model:** Select the NAND Flash device type.

**Custom:** Users create a **AqNFCustom.txt** file into the LA work directory when select the **Custom** vendor item and edit NAND Flash Command set.



```
Manufacturer=Samsung
PartNo=K9XXXXXXXX
#CE/RB=1
X16=N
SyncMode=Y
Cmd=Read, Read, tR, 60, , , N, N, N, 00, 30
Cmd=Read Status, Read Stat., , , , , Y, N, Y, 70
Cmd=Two-Plane Page Program, TPP Prog., tDBSY, 1, tPROG, 5000, N, Y, N, 80, 11, 81, 10
```

Manufacturer, PartNo, #CE/RB, X16, SyncMode, Cmd are keywords.

| Keyword | Description |
|---|---|
| Manufacturer | NAND Flash Vendor. |
| PartNo | NAND Flash IC Model. |
| #CE/RB | Number of targets, only 1/2/4 acceptable. |
| X16 | 8/16 bits device width, only Y/N acceptable. |
| SyncMode | Only Y/N acceptable, Y: Synchronous data interface |

| | |
|---|---|
| | supported; N: Not supported. |
| Cmd | Cmd is composed of several parts, it's divided with comma. |
| | 1. Complete command name. |
| | 2. Abbreviation of command. |
| | 3. Name of first busy time check. Put a space and add a comma if unused. |
| | 4. Value of first busy time check. Its unit is micro seconds. Put a space and add a comma if unused. |
| | 5. Name of second busy time check. Put a space and add a comma if unused. |
| | 6. Value of second busy time check. Its unit is micro mseconds.<br>Put a space and add a comma if unused. |
| | 7. First flag. It's acceptable command during busy. |
| | 8. Second flag. It can be inserted by some command or not. |
| | 9. Third flag. It can insert into some multi plane command or not. |
| | 10. Command. |

Ex: Cmd=Read, Read, tR, 60, , , N, N, N, 00, 30

Cmd=Read Status, Read Stat., , , , , Y, N, Y ,70

Cmd=Two-Plane Page Program, TPP Prog., tDBSY, 1, tPROG, 5000,

N, Y, N, 80, 11, 81, 10

Read Status / Two-Plane Page Program : complete command.

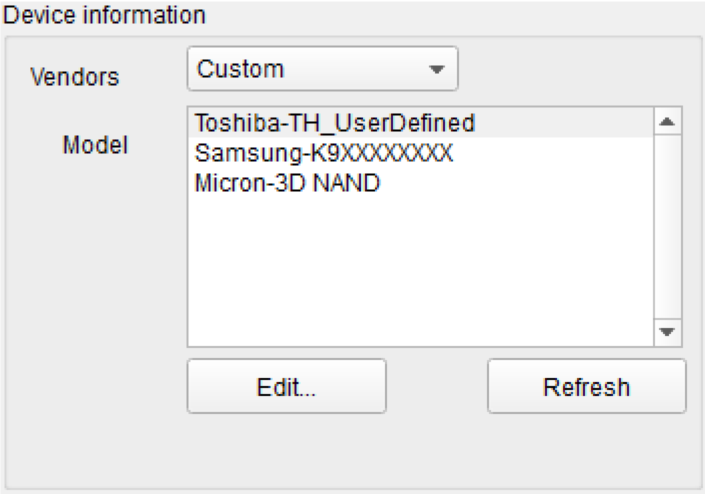
Read Stat. / TPP Prog. : abbreviation of command.

Busy Time Check(tDBSY, 1, tPROG, 5000) : tDBSY is 1us; tPROG is 5000 us. It will show some information when violation of busy time.

3 Flags: 1st flag of "Read Status" is Y means it's acceptable command During busy; 2nd flag of "Two-Plane Page Program" and 3rd flag of "Read

Status" means any command between 11h and 81h is prohibited except "Read Status (70h)".

Perform NAND Flash bus analysis, select Custom, the manufacturer name and model entered in the AqNFCustom.txt file will be displayed in the model; the analysis result will also be displayed according to the input command.



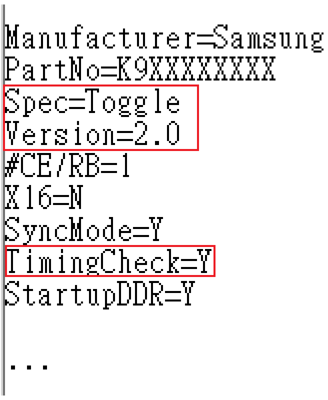
Device information

Vendors: Custom

Model:
Toshiba-TH_UserDefined
Samsung-K9XXXXXXXX
Micron-3D NAND

NAND Bus Decode Timing Check function description

When using this function, please add the following description in the first paragraph of the content of the custom file AqNFCustom.txt.



```
Manufacturer=Samsung
PartNo=K9XXXXXXXX
Spec=Toggle
Version=2.0
#CE/RB=1
X16=N
SyncMode=Y
TimingCheck=Y
StartupDDR=Y

...
```

1. Spec=Toggle

Only ONFI or Toggle can be filled in.

2. Version=2.0

ONFI fill in SDR/NV-DDR/NV-DDR2-3; Toggle fill in Legacy/1.0/2.0.

3. TimingCheck=Y

Only Y/N can be filled in, Y means the Timing Check function is enabled.

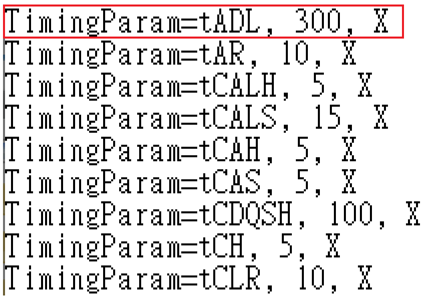
If you turn on the Timing Check function, please fill in the items that need Timing Check, the format:

**Item Name, Minimum Time, Maximum Time**

The unit of time value is ns, and for the Timing Check item provided by the software, please fill in the item name specified in the appendix below according to the Spec/Version you set. These are the standard time check items listed in ONFI and Toggle NAND Flash. All other project names will be ignored.

If the time value check item of a certain item is not needed, please fill in X. If the two time values are both X, the item will also be ignored.

The time values listed in the appendix can be adjusted according to the actual NAND Flash specifications during use.



```
TimingParam=tADL, 300, X
TimingParam=tAR, 10, X
TimingParam=tCALH, 5, X
TimingParam=tCALS, 15, X
TimingParam=tCAH, 5, X
TimingParam=tCAS, 5, X
TimingParam=tCDQSH, 100, X
TimingParam=tCH, 5, X
TimingParam=tCLR, 10, X
```

The results will be displayed in the Information field of the report window, and those that violate the set time range will be displayed in red font.

Normally, it will be displayed in black font.

The displayed information will include:

1. Time test item 2. Measurement time 3. Set time range.

Appendix

| ONFI | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| SDR | | | NV-DDR | | | NV-DDR2-3 | | |
| tADL | 400 | X | tAC | 3 | 25 | tAR | 10 | X |
| tALH | 20 | X | tADL | 400 | X | tCAH | 5 | X |
| tALS | 50 | X | tCADf | 25 | X | tCAS | 5 | X |
| tAR | 25 | X | tCADs | 45 | X | tCALH | 5 | X |
| tCEA | X | 100 | tCAH | 10 | X | tCALS | 15 | X |
| tCEH | 20 | X | tCALH | 10 | X | tCEH | 20 | X |
| tCH | 20 | X | tCALS | 10 | X | tCH | 5 | X |
| tCLH | 20 | X | tCAS | 10 | X | tCS | 20 | X |
| tCLR | 20 | X | tCEH | 20 | X | tCSD | 10 | X |
| tCLS | 50 | X | tCH | 10 | X | tCLR | 10 | X |
| tCOH | 0 | X | tCK | 50 | X | tCR | 10 | X |
| tCR | 10 | X | tCKH(abs) | 0.43 | 0.57 | tDBS | 5 | X |
| tCS | 10 | X | tCKL(abs) | 0.43 | 0.57 | tRHW | 100 | X |
| tDH | 20 | X | tCKWR | 0.43 | X | tWC | 25 | X |
| tDS | 40 | X | tCS | 35 | X | tWH | 11 | X |
| tITC | X | 1000 | tDH | 5 | X | tWHR | 80 | X |

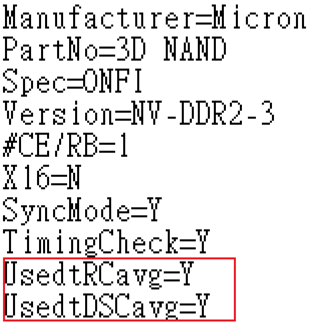
| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| tRC | 100 | X | tDQSCK | 3 | 25 | tITC | X | 1000 |
| tREH | 30 | X | tDQSH | 0.4 | 0.6 | tRR | 20 | X |
| tRHOH | 0 | X | tDQSL | 0.4 | 0.6 | tWB | X | 100 |
| tRHW | X | 200 | tDQSQ | X | 5 | tADL | 400 | X |
| tRLOH | 0 | X | tDSC | 50 | X | tDQSH | 0.43 | X |
| tRP | 50 | X | tDSH | 0.2 | X | tDQSL | 0.43 | X |
| tRR | 40 | X | tDSS | 0.2 | X | tWPRE | 15 | X |
| tWB | X | 100 | tHP | 0.43 | X | tWPST | 6.5 | X |
| tWC | 100 | X | tWPRE | 1.5 | X | tWPSTH | 15 | X |
| tWH | 30 | X | tWPST | 1.5 | X | tDH | 0.3 | X |
| tWHR | 120 | X | tWHR | 80 | X | tDS | 0.3 | X |
| tWP | 50 | X | tFEAT | X | 1000 | tDSC | 3.75 | X |
| tFEAT | X | 1000 | tRST | X | 500000 | tAC | 3 | 25 |
| tRST | X | 500000 | | | | tDQSRE | 3 | 25 |
| | | | | | | tQSH | 0.37 | X |
| | | | | | | tQSL | 0.37 | X |
| | | | | | | tREH(abs) | 0.43 | X |
| | | | | | | tRP(abs) | 0.43 | X |
| | | | | | | tWP | 11 | X |
| | | | | | | tRPRE | 15 | X |
| | | | | | | tRPST | 4.875 | X |
| | | | | | | tRPSTH | 15 | X |
| | | | | | | tDQSRH | 5 | X |
| | | | | | | tRC | 3.75 | X |
| | | | | | | tCD | 3.75 | X |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | tFEAT | X | 1000 |
| | | | | | | tRST | X | 500000 |

Note: Some Timing Check items will be multiplied by the average value of certain time items, including:

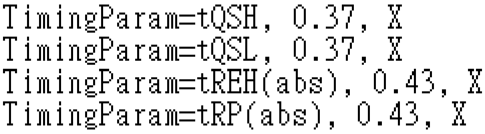
1. tDQSH/tDQSL: 0.45 x tDSC(avg)
2. tQSH/tQSL: 0.37 x tRC(avg)
3. tREH/tRP: 0.43 x tRC(avg)

If the above Timing calculation method is used, the following description must be added to the front end of the file:



```
Manufacturer=Micron
PartNo=3D NAND
Spec=ONFI
Version=NV-DDR2-3
#CE/RB=1
X16=N
SyncMode=Y
TimingCheck=Y
UsedtRCavg=Y
UsedtDSCavg=Y
```

Then check the item input value at that time:



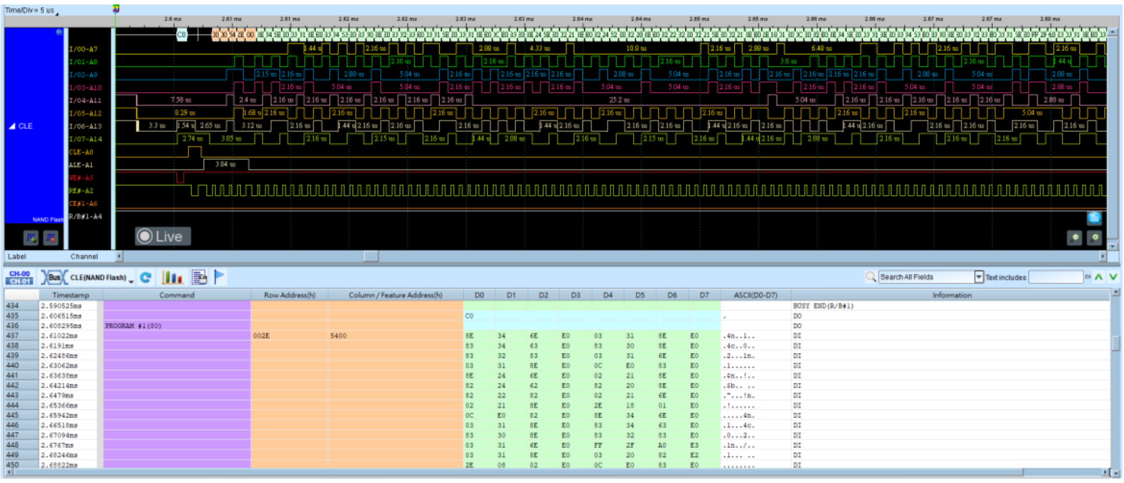
```
TimingParam=tQSH, 0.37, X
TimingParam=tQSL, 0.37, X
TimingParam=tREH(abs), 0.43, X
TimingParam=tRP(abs), 0.43, X
```

If you do not use the above Timing calculation method, you only need to enter UsedtRCavg=N, UsedtDSCavg=N in the file header or remove this description completely, and the value entered for the time check item will be the minimum/maximum value of the time deal with.

| Toggle | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Legacy | | | 1.0 | | | 2.0 | | |
| tCLS | 10 | X | tADL | 300 | X | tADL | 300 | X |
| tCLS2 | 40 | X | tAR | 10 | X | tAR | 10 | X |
| tCLH | 5 | X | tCALH | 5 | X | tCALH | 5 | X |
| tCS | 15 | X | tCALS | 15 | X | tCALS | 15 | X |
| tCH | 5 | X | tCAH | 5 | X | tCAH | 5 | X |
| tWP | 10 | X | tCAS | 5 | X | tCAS | 5 | X |
| tALS | 10 | X | tCDQSH | 100 | X | tCDQSH | 100 | X |
| tALH | 5 | X | tCH | 5 | X | tCH | 5 | X |
| tDS | 5 | X | tCLR | 10 | X | tCLR | 10 | X |
| tDH | 5 | X | tCOH | 5 | X | tCOH | 5 | X |
| tWC | 10 | X | tCR | 10 | X | tCR | 10 | X |
| tWH | 10 | X | tCRES | 10 | X | tCRES | 10 | X |
| tADL | 300 | X | tCS | 20 | X | tCS | 20 | X |
| tRR | 10 | X | tDH | 0.9 | X | tDH | 0.4 | X |
| tRP | 10 | X | tDQSH | 4 | X | tDQSH | 2 | X |
| tRC | 20 | X | tDQSL | 4 | X | tDQSL | 2 | X |
| tCR | 9 | X | tDQSRE | X | 25 | tDQSRE | X | 25 |
| tCLR | 10 | X | tDSC | 10 | X | tRC | 5 | X |
| tAR | 10 | X | tDS | 0.9 | X | tREH | 2 | X |
| tRHOH | 25 | X | tRC | 10 | X | tRP | 2 | X |
| tRLOH | 5 | X | tREH | 4 | X | tRPP | 30 | X |
| tREH | 7 | X | tRP | 4 | X | tRPRE | 15 | X |
| tWHR | 30 | X | tRPP | 30 | X | tRPST | 27.5 | X |

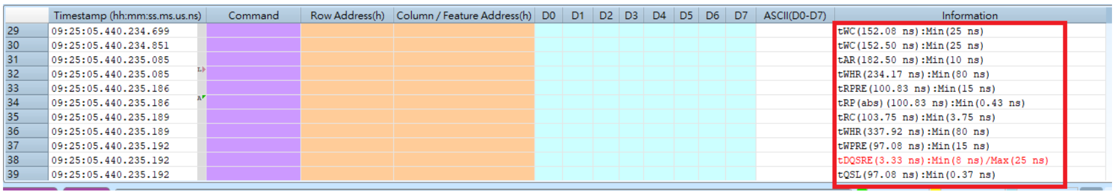
| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| tWHC | 30 | X | tRPRE | 15 | X | tRPSTH | 25 | X |
| tWHR1 | 180 | X | tRPST | 27.5 | X | tRR | 5 | X |
| tWHR2 | 300 | X | tRPSTH | 25 | X | tWB | X | 100 |
| tWB | X | 100 | tRR | 20 | X | tWC | 25 | X |
| tFEAT | X | 1000 | tWB | X | 100 | tWH | 11 | X |
| tRST | X | 100000 | tWC | 25 | X | tWHR | 120 | X |
| | | | tWH | 11 | X | tWHR2 | 300 | X |
| | | | tWHR | 120 | X | tWP | 11 | X |
| | | | tWHR2 | 300 | X | tWPRE | 15 | X |
| | | | tWP | 11 | X | tWPST | 6.5 | X |
| | | | tWPRE | 15 | X | tWPSTH | 25 | X |
| | | | tWPST | 6.5 | X | tFEAT | X | 1000 |
| | | | tWPSTH | 25 | X | tRST | X | 500000 |
| | | | tFEAT | 1000 | X | | | |
| | | | tRST | 500000 | X | | | |

## Result



| | Timestamp | Command | Row Address(h) | Column / Feature Address(h) | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII(D0-D7) | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 434 | 2.590525ms | | | | | | | | | | | | | BUSY END(R/B#1) |
| 435 | 2.606515ms | | | | C0 | | | | | | | | . | DO |
| 436 | 2.608295ms | PROGRAM #1(80) | | | | | | | | | | | | DO |
| 437 | 2.61022ms | | 002E | 5400 | 8E | 34 | 6E | E0 | 03 | 31 | 8E | E0 | .4n..1.. | DI |
| 438 | 2.6191ms | | | | 83 | 34 | 63 | E0 | 83 | 30 | 8E | E0 | .4c..0.. | DI |
| 439 | 2.62486ms | | | | 83 | 32 | 83 | E0 | 03 | 31 | 6E | E0 | .2...1n. | DI |
| 440 | 2.63062ms | | | | 03 | 31 | 8E | E0 | 0C | E0 | 83 | E0 | .1...... | DI |
| 441 | 2.63638ms | | | | 8E | 24 | 6E | E0 | 02 | 21 | 8E | E0 | .$n..!.. | DI |
| 442 | 2.64214ms | | | | 82 | 24 | 62 | E0 | 82 | 20 | 8E | E0 | .$b.. .. | DI |
| 443 | 2.6479ms | | | | 82 | 22 | 82 | E0 | 02 | 21 | 6E | E0 | ."...!n. | DI |
| 444 | 2.65366ms | | | | 02 | 21 | 8E | E0 | 2E | 18 | 01 | E0 | .!...... | DI |
| 445 | 2.65942ms | | | | 0C | E0 | 82 | E0 | 8E | 34 | 6E | E0 | .....4n. | DI |
| 446 | 2.66518ms | | | | 03 | 31 | 8E | E0 | 83 | 34 | 63 | E0 | .1...4c. | DI |
| 447 | 2.67094ms | | | | 83 | 30 | 8E | E0 | 83 | 32 | 83 | E0 | .0...2.. | DI |
| 448 | 2.6767ms | | | | 03 | 31 | 6E | E0 | FF | 2F | A0 | E3 | .1n../.. | DI |
| 449 | 2.68246ms | | | | 03 | 31 | 8E | E0 | 03 | 20 | 82 | E2 | .1... .. | DI |
| 450 | 2.68822ms | | | | 2E | 08 | 02 | E0 | 0C | E0 | 83 | E0 | ........ | DI |

## Timing Check

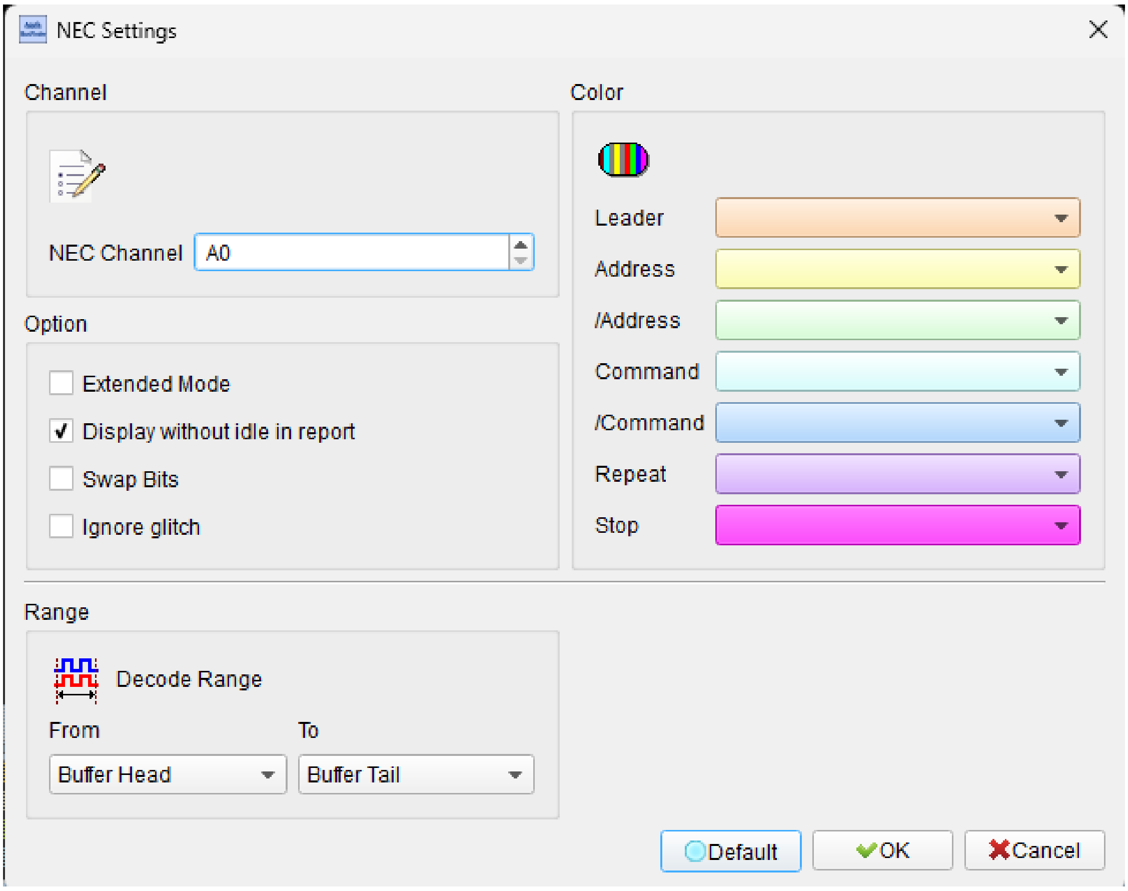


| | Timestamp (hh:mm:ss.ms.us.ns) | Command | Row Address(h) | Column / Feature Address(h) | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII(D0-D7) | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 29 | 09:25:05.440.234.699 | | | | | | | | | | | | | tWC(152.08 ns):Min(25 ns) |
| 30 | 09:25:05.440.234.851 | | | | | | | | | | | | | tWC(152.50 ns):Min(25 ns) |
| 31 | 09:25:05.440.235.085 | | | | | | | | | | | | | tAR(182.50 ns):Min(10 ns) |
| 32 | 09:25:05.440.235.085 | | | | | | | | | | | | | tWHR(234.17 ns):Min(80 ns) |
| 33 | 09:25:05.440.235.186 | | | | | | | | | | | | | tRPRE(100.83 ns):Min(15 ns) |
| 34 | 09:25:05.440.235.186 | | | | | | | | | | | | | tRP(abs)(100.83 ns):Min(0.43 ns) |
| 35 | 09:25:05.440.235.189 | | | | | | | | | | | | | tRC(103.75 ns):Min(3.75 ns) |
| 36 | 09:25:05.440.235.189 | | | | | | | | | | | | | tWHR(337.92 ns):Min(80 ns) |
| 37 | 09:25:05.440.235.192 | | | | | | | | | | | | | tWPRE(97.08 ns):Min(15 ns) |
| 38 | 09:25:05.440.235.192 | | | | | | | | | | | | | tDQSRE(3.33 ns):Min(8 ns)/Max(25 ns) |
| 39 | 09:25:05.440.235.192 | | | | | | | | | | | | | tQSL(97.08 ns):Min(0.37 ns) |

## NEC IR

NEC IR (NEC Infrared) is a company that specializes in infrared technology and solutions, often referred to as NEC (Nippon Electric Company) products or services in the field of infrared technology.

### Settings



NEC Settings

Channel

NEC Channel: A0

Option

☐ Extended Mode
☑ Display without idle in report
☐ Swap Bits
☐ Ignore glitch

Range

Decode Range
From: Buffer Head
To: Buffer Tail

Color

Leader, Address, /Address, Command, /Command, Repeat, Stop

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

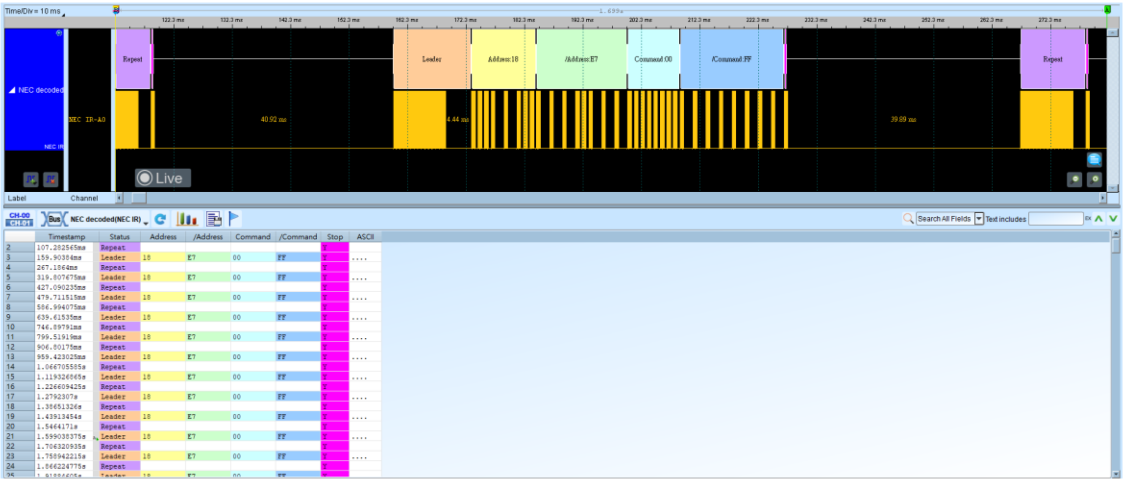
**Options:** Enabled when checked.

1. **Extended Mode:** It integrates /Address and Address into 16 Bits Address, /Command and Command into 16 Bits Command.
2. **Display without idle in report:** It will not idle on the Report Window for the user to observe and analyze data.

3. **Swap Bits:** Switch LSB First to MSB First.

4. **Ignore glitch:** Ignore glitch while analysis. Enabled when checked.

## Result

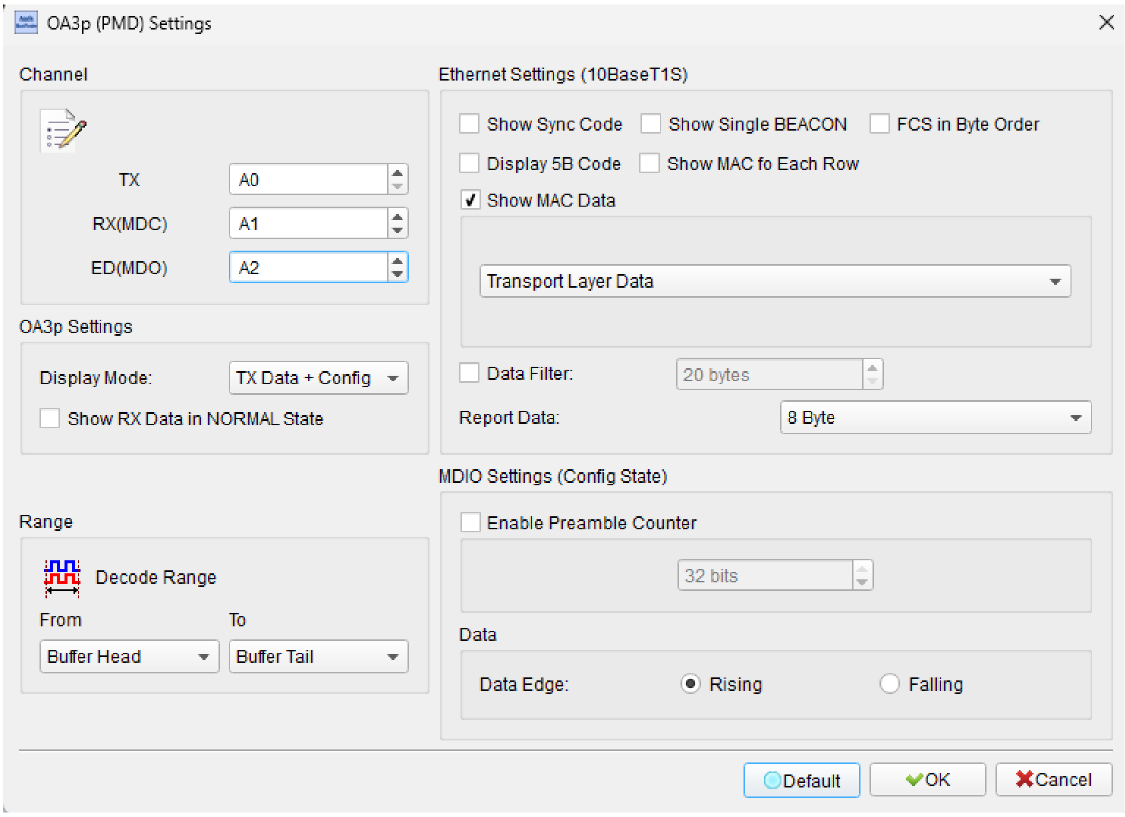


| | Timestamp | Status | Address | /Address | Command | /Command | Stop | ASCII |
|---|---|---|---|---|---|---|---|---|
| 2 | 107.282565ms | Repeat | | | | | Y | |
| 3 | 159.90384ms | Leader | 18 | E7 | 00 | FF | Y | .... |
| 4 | 267.1864ms | Repeat | | | | | Y | |
| 5 | 319.807675ms | Leader | 18 | E7 | 00 | FF | Y | .... |
| 6 | 427.090235ms | Repeat | | | | | Y | |
| 7 | 479.711515ms | Leader | 18 | E7 | 00 | FF | Y | .... |
| 8 | 586.994075ms | Repeat | | | | | Y | |
| 9 | 639.61535ms | Leader | 18 | E7 | 00 | FF | Y | .... |
| 10 | 746.89791ms | Repeat | | | | | Y | |
| 11 | 799.51919ms | Leader | 18 | E7 | 00 | FF | Y | .... |
| 12 | 906.80175ms | Repeat | | | | | Y | |
| 13 | 959.423025ms | Leader | 18 | E7 | 00 | FF | Y | .... |
| 14 | 1.066705585s | Repeat | | | | | Y | |
| 15 | 1.119326865s | Leader | 18 | E7 | 00 | FF | Y | .... |
| 16 | 1.226609425s | Repeat | | | | | Y | |
| 17 | 1.2792307s | Leader | 18 | E7 | 00 | FF | Y | .... |
| 18 | 1.38651326s | Repeat | | | | | Y | |
| 19 | 1.43913454s | Leader | 18 | E7 | 00 | FF | Y | .... |
| 20 | 1.5464171s | Repeat | | | | | Y | |
| 21 | 1.599038375s | Leader | 18 | E7 | 00 | FF | Y | .... |
| 22 | 1.706320935s | Repeat | | | | | Y | |
| 23 | 1.758942215s | Leader | 18 | E7 | 00 | FF | Y | .... |
| 24 | 1.866224775s | Repeat | | | | | Y | |
| 25 | 1.91884605s | Leader | 18 | E7 | 00 | FF | Y | |

## OA3p(PMD)

OA3p (OPEN Alliance 3-pin) is an important protocol in automotive Ethernet testing that defines the requirements for Physical Layer (PHY) testing in automotive environments to ensure the consistency, reliability and stability of the equipment under different operating conditions.

### Settings



OA3p (PMD) Settings

Channel

TX: A0
RX(MDC): A1
ED(MDO): A2

OA3p Settings

Display Mode: TX Data + Config
□ Show RX Data in NORMAL State

Range

Decode Range
From: Buffer Head
To: Buffer Tail

Ethernet Settings (10BaseT1S)

□ Show Sync Code □ Show Single BEACON □ FCS in Byte Order
□ Display 5B Code □ Show MAC fo Each Row
☑ Show MAC Data
Transport Layer Data
□ Data Filter: 20 bytes
Report Data: 8 Byte

MDIO Settings (Config State)

□ Enable Preamble Counter
32 bits

Data

Data Edge: ● Rising ○ Falling

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**OA3p settings:**

1. **Display Mode:** Display TX Data / RX Data + Config result in waveform area.
2. **Show Rx Data in NORMAL State:** Show RX analyzed result even the Bus goes into Normal State. Enabled when checked.

**Ethernet Settings (10BaseT1S):** Set the parsing result of Ethernet in report area. Enabled when checked.

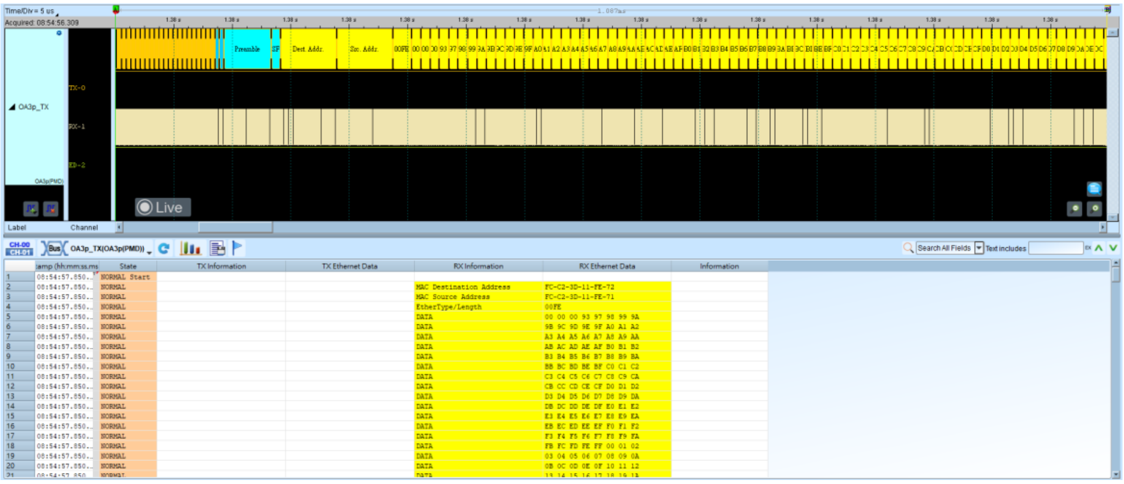
1. **Show Sync Code**
2. **Show 5B Code**
3. **Show MAC Data:** Enabling this option allows user to additionally set whether the Transport Layer Data or the Transport Layer Data & Header is displayed.
4. **Show Single BEACON**
5. **Show MAC of Each Row**
6. **FCS in Byte Order**
7. **Data Filter:** Displays only the data of the set number of bytes.
8. **Report Data:** Set the number of bytes displayed in the data field in report area, and the excess bytes are displayed in a new line.

**MDIO Settings (Config State):**

1. **Enable Preamble Counter:** Set the number of preamble bit. Enabled when checked.

**Data Edge:** Latch data at the rising or falling edge.

**Result**

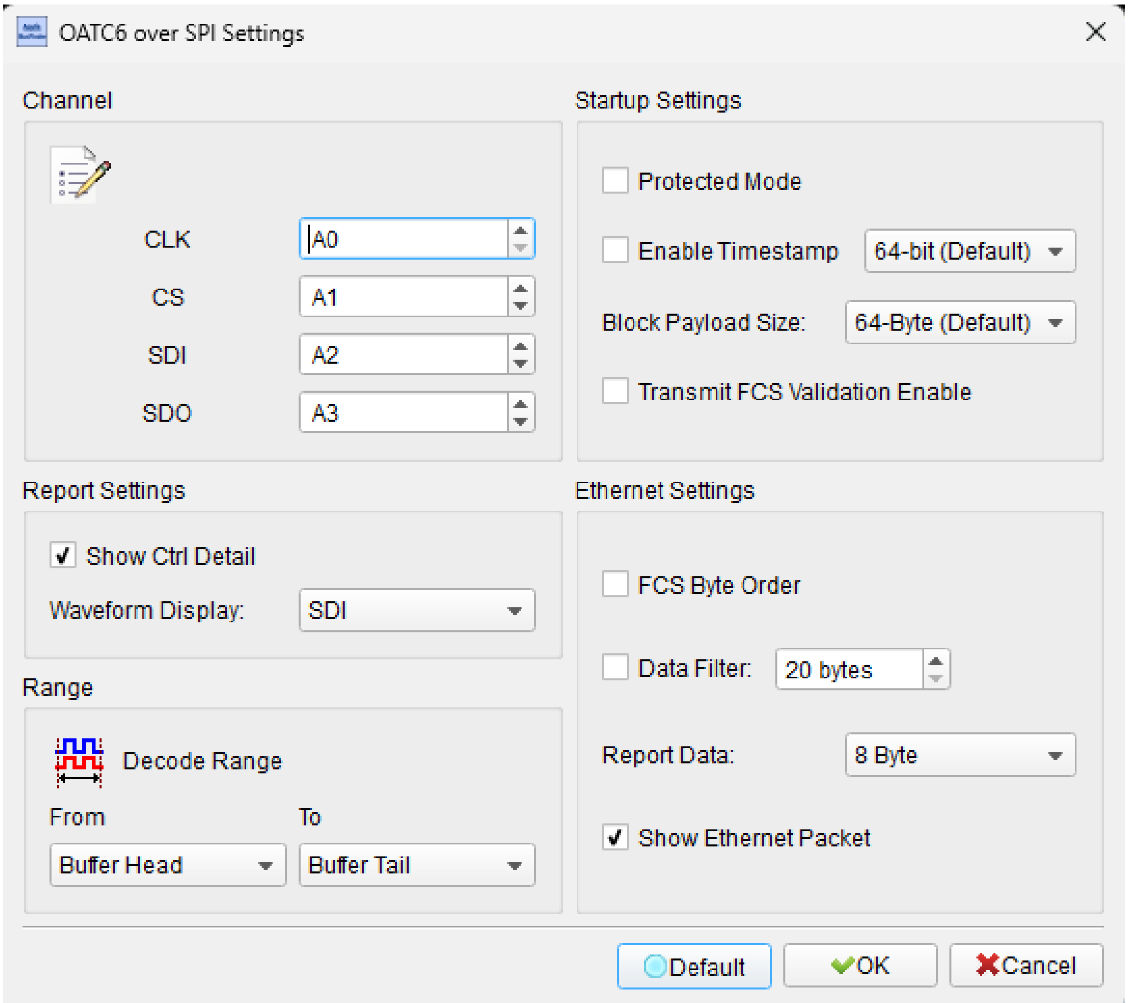


| | tamp (hh:mm:ss.ms | State | TX Information | TX Ethernet Data | RX Information | RX Ethernet Data | Information |
|---|---|---|---|---|---|---|---|
| 1 | 08:54:57.850... | NORMAL Start | | | | | |
| 2 | 08:54:57.850... | NORMAL | | | MAC Destination Address | FC-C2-3D-11-FE-72 | |
| 3 | 08:54:57.850... | NORMAL | | | MAC Source Address | FC-C2-3D-11-FE-71 | |
| 4 | 08:54:57.850... | NORMAL | | | EtherType/Length | 00FE | |
| 5 | 08:54:57.850... | NORMAL | | | DATA | 00 00 00 93 97 98 99 9A | |
| 6 | 08:54:57.850... | NORMAL | | | DATA | 9B 9C 9D 9E 9F A0 A1 A2 | |
| 7 | 08:54:57.850... | NORMAL | | | DATA | A3 A4 A5 A6 A7 A8 A9 AA | |
| 8 | 08:54:57.850... | NORMAL | | | DATA | AB AC AD AE AF B0 B1 B2 | |
| 9 | 08:54:57.850... | NORMAL | | | DATA | B3 B4 B5 B6 B7 B8 B9 BA | |
| 10 | 08:54:57.850... | NORMAL | | | DATA | BB BC BD BE BF C0 C1 C2 | |
| 11 | 08:54:57.850... | NORMAL | | | DATA | C3 C4 C5 C6 C7 C8 C9 CA | |
| 12 | 08:54:57.850... | NORMAL | | | DATA | CB CC CD CE CF D0 D1 D2 | |
| 13 | 08:54:57.850... | NORMAL | | | DATA | D3 D4 D5 D6 D7 D8 D9 DA | |
| 14 | 08:54:57.850... | NORMAL | | | DATA | DB DC DD DE DF E0 E1 E2 | |
| 15 | 08:54:57.850... | NORMAL | | | DATA | E3 E4 E5 E6 E7 E8 E9 EA | |
| 16 | 08:54:57.850... | NORMAL | | | DATA | EB EC ED EE EF F0 F1 F2 | |
| 17 | 08:54:57.850... | NORMAL | | | DATA | F3 F4 F5 F6 F7 F8 F9 FA | |
| 18 | 08:54:57.850... | NORMAL | | | DATA | FB FC FD FE FF 00 01 02 | |
| 19 | 08:54:57.850... | NORMAL | | | DATA | 03 04 05 06 07 08 09 0A | |
| 20 | 08:54:57.850... | NORMAL | | | DATA | 0B 0C 0D 0E 0F 10 11 12 | |
| 21 | 08:54:57.850... | NORMAL | | | DATA | 13 14 15 16 17 18 19 1A | |

### OATC6 over SPI

OPEN Alliance Technical Committee 6 (TC6) focuses on improving the media-independent communication interface (xMII) to enhance its use in automotive networks.TC6 develops recommendations for automotive xMII standards and defines related improvements.

### Settings



OATC6 over SPI Settings

Channel

| | |
|---|---|
| CLK | A0 |
| CS | A1 |
| SDI | A2 |
| SDO | A3 |

Startup Settings

□ Protected Mode

□ Enable Timestamp 64-bit (Default)

Block Payload Size: 64-Byte (Default)

□ Transmit FCS Validation Enable

Report Settings

☑ Show Ctrl Detail

Waveform Display: SDI

Ethernet Settings

□ FCS Byte Order

□ Data Filter: 20 bytes

Report Data: 8 Byte

☑ Show Ethernet Packet

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Report Settings:**

1. **Show Ctrl Detail:** Show the Ctrl detail information in report area. Enabled when checked.

2. **Waveform Display:** Display the SDI or SDO result in waveform area.

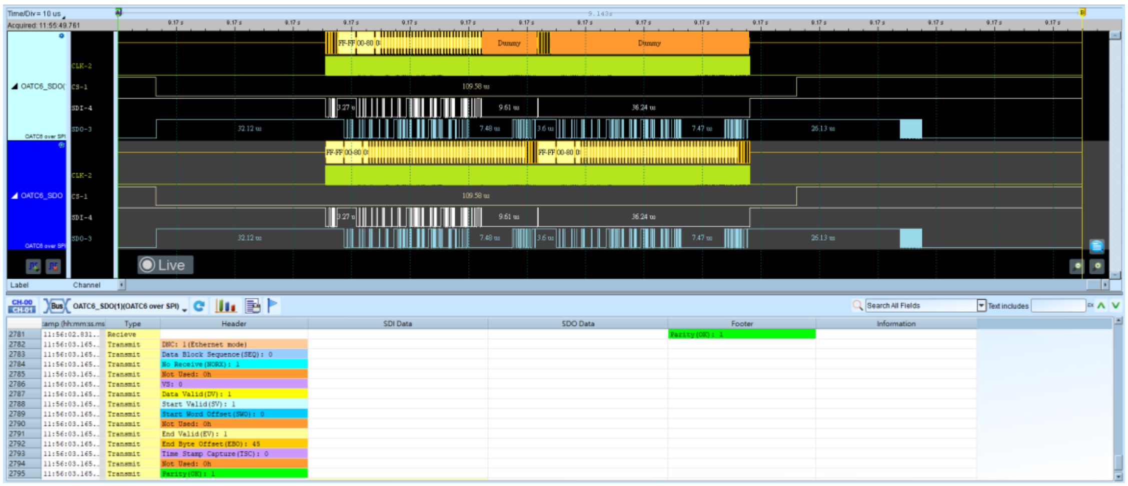
**Startup Settings, enabled when checked:**

1. **Protected Mode:** Enable or disable the protected format int the control frame.
2. **Enable Timestamp:** Set the output timestamp format. User can choose 64 bit (default value) or 32 bit format.
3. **Block Payload Size:** Assign the Block Payload Size. User can choose 64 bit (default value) or 32 bit format.
4. **Transmit FCS Validation Enable:** Enable Frame Check Sequence (FCS) validation during transmission to ensure data integrity.

**Ethernet Settings:**

1. **FCS Byte Order:** Present the FCS in Byte order in the report.
2. **Data Filter:** Only how many bytes of data are displayed (at least 20 bytes).
3. **Report Data:** Limits the maximum number of bytes of data that can be displayed in the Data field; the portion that exceeds the setting continues to be displayed on a new line.
4. **Show Ethernet Packet:** Displays Ethernet data such as Address, Data, and FCS.

## Result

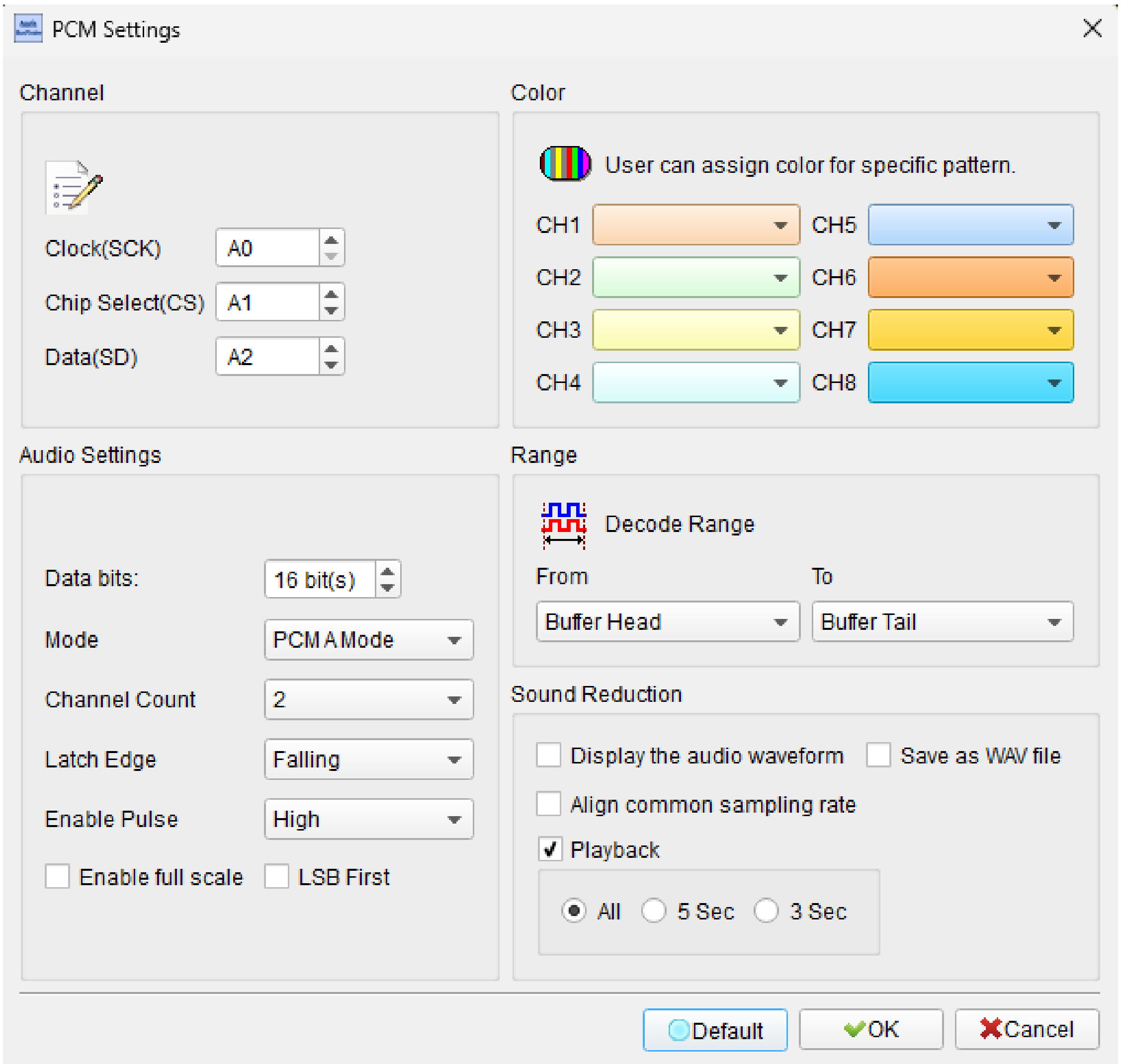


| | amp (hh:mm:ss.ms | Type | Header | SDI Data | SDO Data | Footer | Information |
|---|---|---|---|---|---|---|---|
| 2781 | 11:56:02.831.. | Recieve | | | | Parity(OK): 1 | |
| 2782 | 11:56:03.165.. | Transmit | DNC: 1(Ethernet mode) | | | | |
| 2783 | 11:56:03.165.. | Transmit | Data Block Sequence(SEQ): 0 | | | | |
| 2784 | 11:56:03.165.. | Transmit | No Receive(NORX): 1 | | | | |
| 2785 | 11:56:03.165.. | Transmit | Not Used: 0h | | | | |
| 2786 | 11:56:03.165.. | Transmit | VS: 0 | | | | |
| 2787 | 11:56:03.165.. | Transmit | Data Valid(DV): 1 | | | | |
| 2788 | 11:56:03.165.. | Transmit | Start Valid(SV): 1 | | | | |
| 2789 | 11:56:03.165.. | Transmit | Start Word Offset(SWO): 0 | | | | |
| 2790 | 11:56:03.165.. | Transmit | Not Used: 0h | | | | |
| 2791 | 11:56:03.165.. | Transmit | End Valid(EV): 1 | | | | |
| 2792 | 11:56:03.165.. | Transmit | End Byte Offset(EBO): 45 | | | | |
| 2793 | 11:56:03.165.. | Transmit | Time Stamp Capture(TSC): 0 | | | | |
| 2794 | 11:56:03.165.. | Transmit | Not Used: 0h | | | | |
| 2795 | 11:56:03.165.. | Transmit | Parity(OK): 1 | | | | |

## PCM

The format that describes sound as a series of numbers is often referred to as a raw audio file, and the term for it is PCM (Pulse-code modulation). Examples of this format include WAV files on Microsoft Windows (sound files recorded using the "Sound Recorder" software) and AIFF files on Apple platforms. These files belong to this format, typically with a header added at the beginning.

## Settings



PCM Settings

Channel

Clock(SCK): A0
Chip Select(CS): A1
Data(SD): A2

Color

User can assign color for specific pattern.

CH1, CH2, CH3, CH4, CH5, CH6, CH7, CH8

Audio Settings

Data bits: 16 bit(s)
Mode: PCM A Mode
Channel Count: 2
Latch Edge: Falling
Enable Pulse: High
☐ Enable full scale ☐ LSB First

Range

Decode Range
From: Buffer Head
To: Buffer Tail

Sound Reduction

☐ Display the audio waveform ☐ Save as WAV file
☐ Align common sampling rate
☑ Playback
● All ○ 5 Sec ○ 3 Sec

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

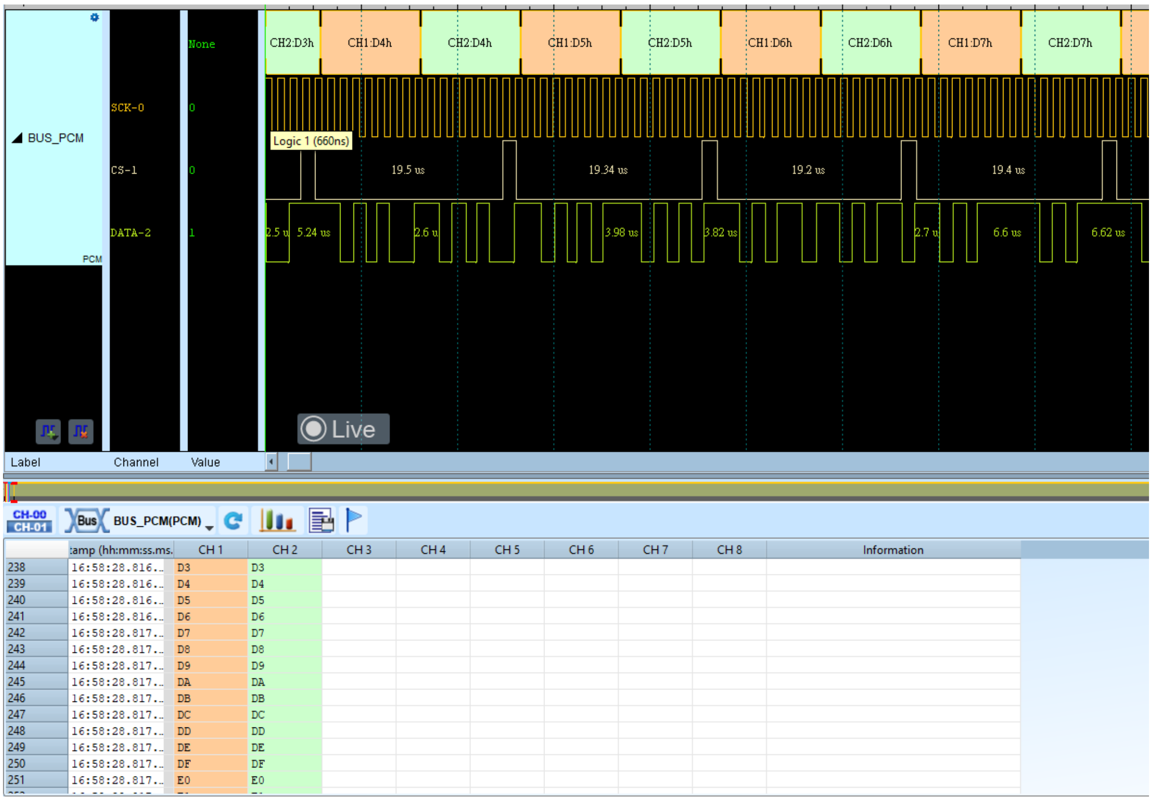
**Audio Settings:**

1. **Data bits:** Set the PCM Data bit amount.
2. **Mode:** PCM mode settings. User can set to PCM A Mode, PCM B Mode, PCM Multi Mode.
3. **Channel Count:** PCM Channel Number Adjustment
4. **Latch Edge:** Latch data at the clock rising or falling edge.
5. **Enable Pulse:** CS operation settings, can be set to High or Low.
6. **Enable full scale:** Full scale function. Enabled when checked.
7. **LSB First:** Data is arranged in LSB first. Enabled when checked.

**Sound Reduction:** Display, playback or save the sound waveform. Enabled when checked.

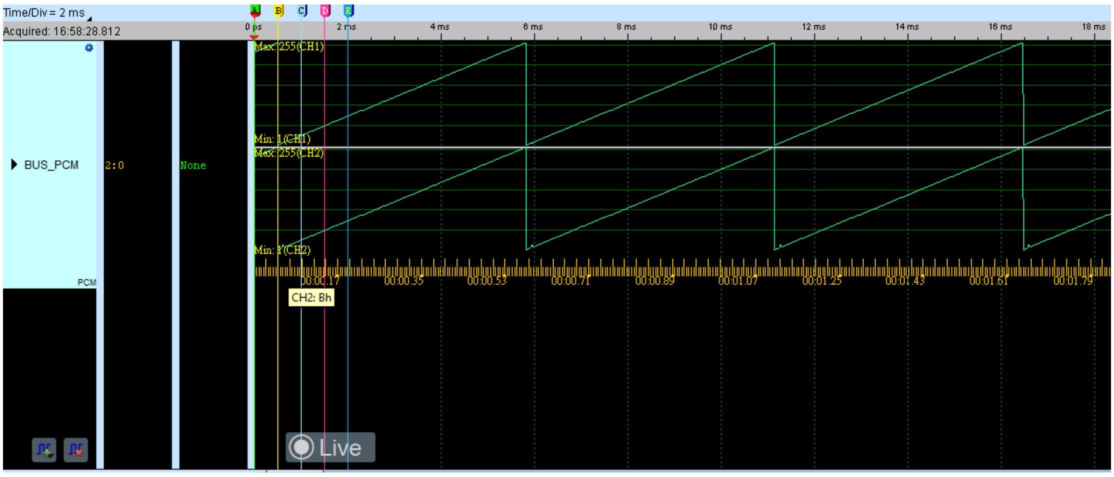
**Result**

**Packet**



| | tamp (hh:mm:ss.ms. | CH 1 | CH 2 | CH 3 | CH 4 | CH 5 | CH 6 | CH 7 | CH 8 | Information |
|---|---|---|---|---|---|---|---|---|---|---|
| 238 | 16:58:28.816... | D3 | D3 | | | | | | | |
| 239 | 16:58:28.816... | D4 | D4 | | | | | | | |
| 240 | 16:58:28.816... | D5 | D5 | | | | | | | |
| 241 | 16:58:28.816... | D6 | D6 | | | | | | | |
| 242 | 16:58:28.817... | D7 | D7 | | | | | | | |
| 243 | 16:58:28.817... | D8 | D8 | | | | | | | |
| 244 | 16:58:28.817... | D9 | D9 | | | | | | | |
| 245 | 16:58:28.817... | DA | DA | | | | | | | |
| 246 | 16:58:28.817... | DB | DB | | | | | | | |
| 247 | 16:58:28.817... | DC | DC | | | | | | | |
| 248 | 16:58:28.817... | DD | DD | | | | | | | |
| 249 | 16:58:28.817... | DE | DE | | | | | | | |
| 250 | 16:58:28.817... | DF | DF | | | | | | | |
| 251 | 16:58:28.817... | E0 | E0 | | | | | | | |

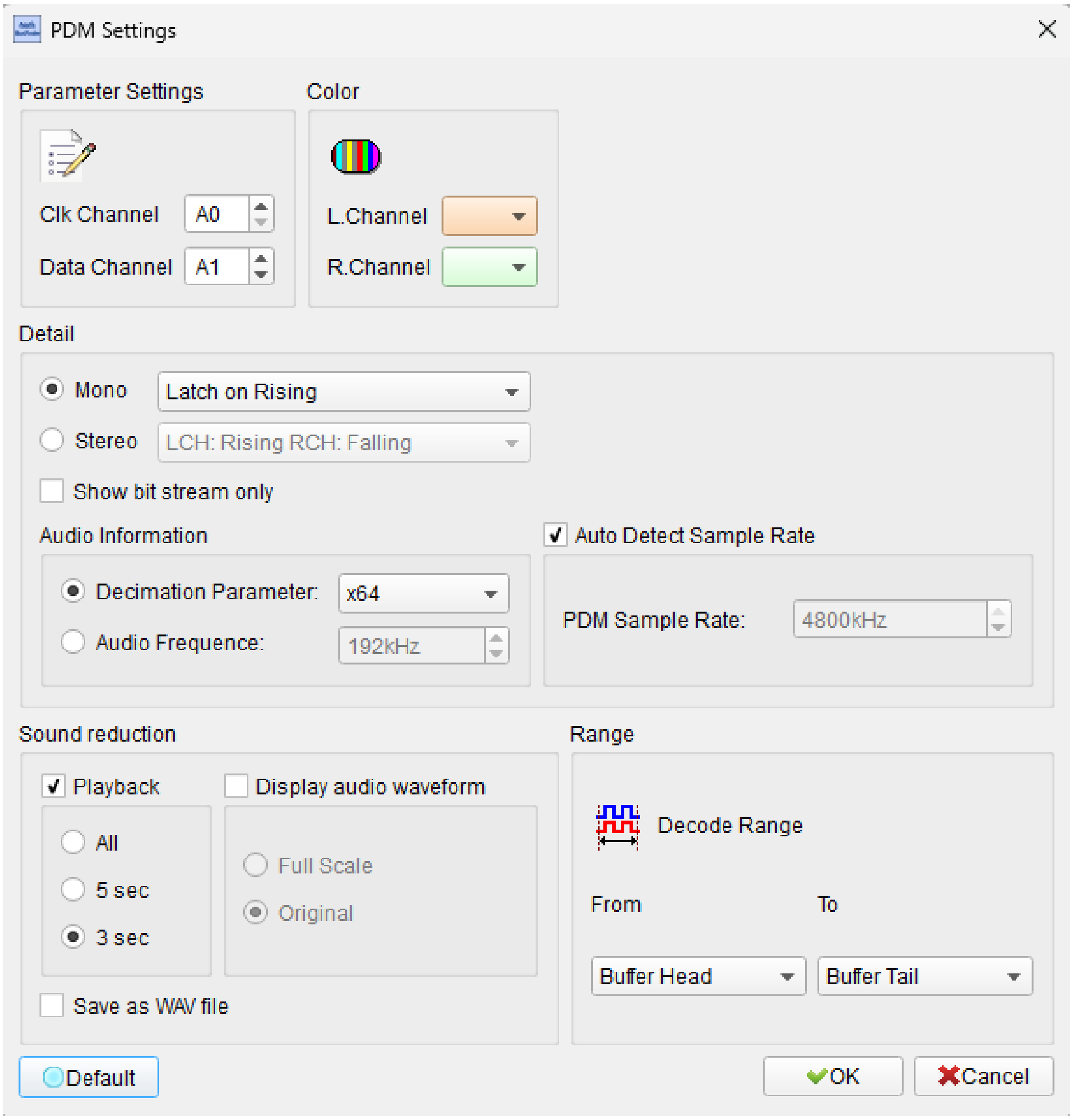
## Audio


Time/Div = 2 ms
Acquired: 16:58:28.812
A
B
C
D
E
0 ps
2 ms
4 ms
6 ms
8 ms
10 ms
12 ms
14 ms
16 ms
18 ms
BUS_PCM
2:0
None
PCM
Max: 255(CH1)
Min: 1(CH1)
Max: 255(CH2)
Min: 1(CH2)
00:00.17
00:00.35
00:00.53
00:00.71
00:00.89
00:01.07
00:01.25
00:01.43
00:01.61
00:01.79
CH2: Bh
Live

## PDM

PDM (Pulse Density Modulation) is a digital signal modulation technique primarily applied in the digitization and transmission of audio signals.

### Settings



PDM Settings

Parameter Settings

Clk Channel: A0

Data Channel: A1

Color

L.Channel: 🟧

R.Channel: 🟩

Detail

● Mono: Latch on Rising

○ Stereo: LCH: Rising RCH: Falling

□ Show bit stream only

Audio Information

● Decimation Parameter: x64

○ Audio Frequence: 192kHz

☑ Auto Detect Sample Rate

PDM Sample Rate: 4800kHz

Sound reduction

☑ Playback

○ All

○ 5 sec

● 3 sec

□ Display audio waveform

○ Full Scale

● Original

□ Save as WAV file

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Detail:**

1. **Mono & Stereo:** Audio mono or stereo selection.

2. **PDM Sample Rate: P**DM Clock Speed
3. **Decimation Parameter:** Calculation parameter for PDM to PCM
4. **Audio Frequency:** PCM audio frequency

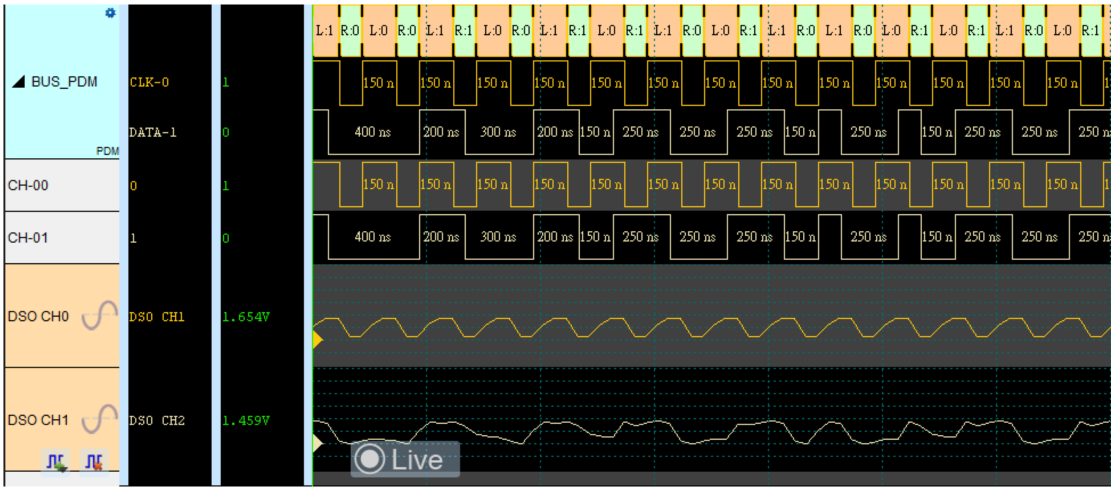
**Sound reduction:**

1. **Playback:** Set the playback time duration.
2. **Display audio waveform:** Use Full Scale or Original to draw sound waveform in waveform area.

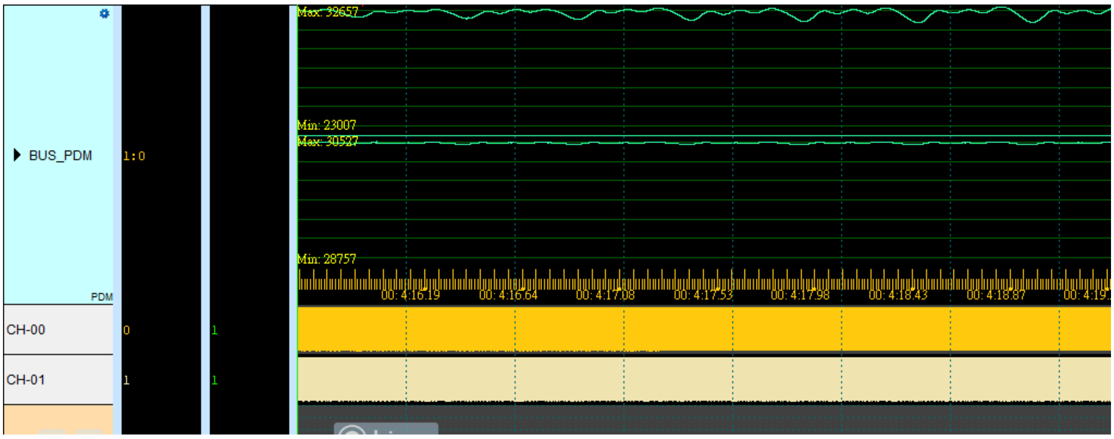
**Save as WAV file:** Saves the restored sound waveform as a .wav. Enabled when checked.

**Result**

**Normal mode**


BUS_PDM
PDM
CLK-0
DATA-1
CH-00
CH-01
DSO CH0
DSO CH1
DSO CH1
DSO CH2
1.654V
1.459V
L:1 R:0 L:0 R:0 L:1 R:1 L:0 R:0 L:1 R:1 L:0 R:1 L:1 R:0 L:0 R:1 L:1 R:0 L:1 R:0 L:0 R:1 L:0 R:1 L:1 R:0 L:0 R:1
150 n
400 ns
200 ns
300 ns
250 ns
Live

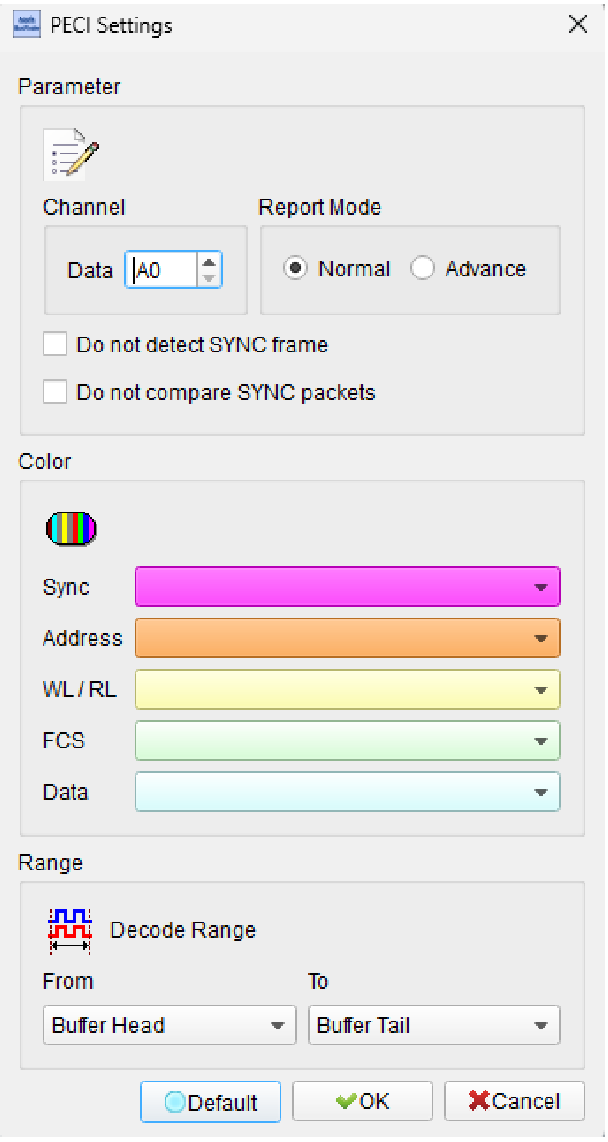
Audio mode



## PECI

Platform Environment Control Interface, Platform management include thermal, power and electrical error monitoring.

### Settings



PECI Settings

Parameter

Channel

Data A0

Report Mode

● Normal ○ Advance

□ Do not detect SYNC frame

□ Do not compare SYNC packets

Color

Sync

Address

WL / RL

FCS

Data

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

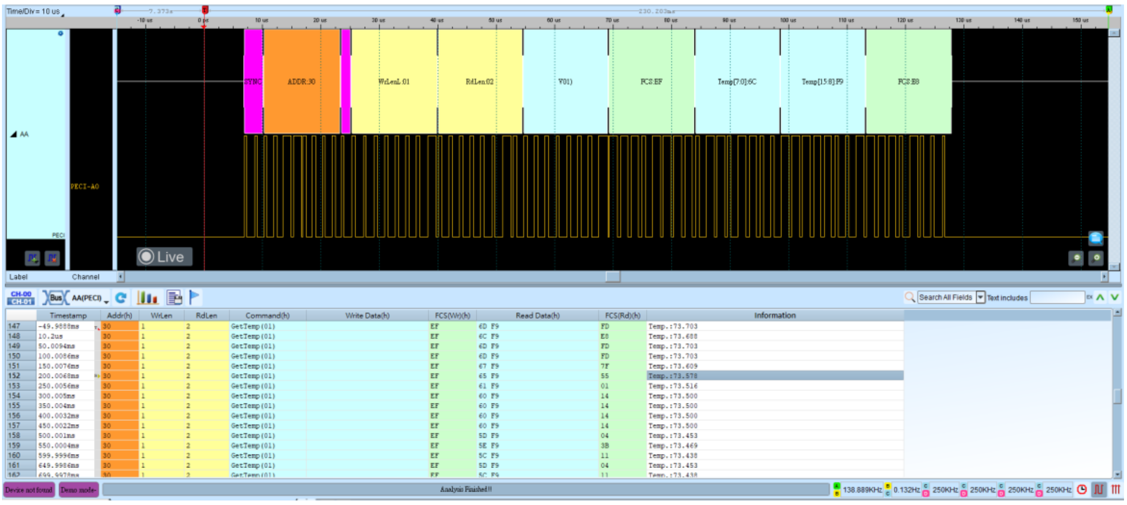
**Report Mode:** User can choose between Normal and Advanced modes, and the Advanced mode displays more detailed information.

**Do not detect SYNC frame:** Do not detect SYNC frame. Enabled when checked.

**Do not compare SYNC packets:** Do not compare SYNC packets. Enabled when checked.

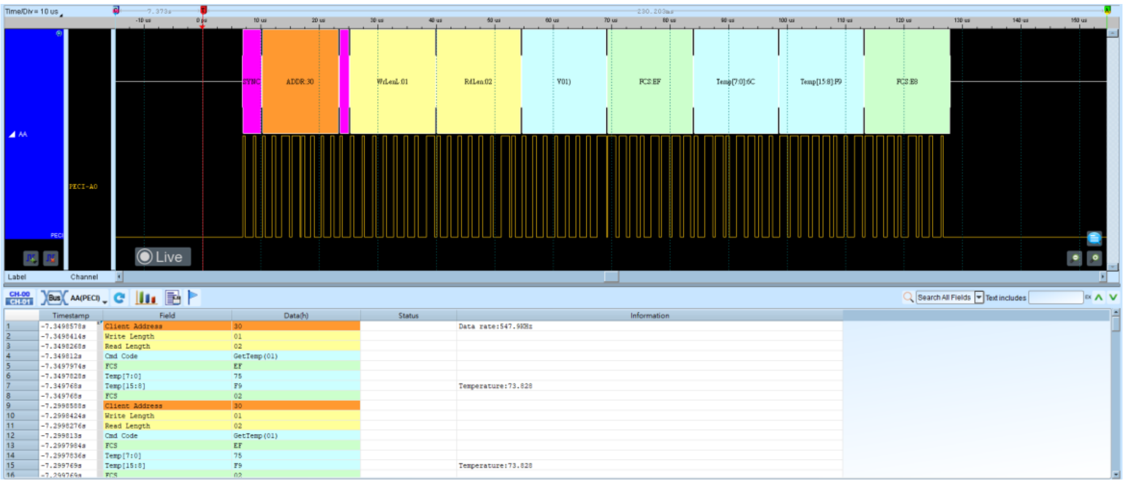
### Result

### Normal mode



| | Timestamp | Addr(h) | WrLen | RdLen | Command(h) | Write Data(h) | FCS(Wr)(h) | Read Data(h) | FCS(Rd)(h) | Information |
|---|---|---|---|---|---|---|---|---|---|---|
| 147 | -49.9888ms | 30 | 1 | 2 | GetTemp(01) | | EF | 6D F9 | FD | Temp.:73.703 |
| 148 | 10.2us | 30 | 1 | 2 | GetTemp(01) | | EF | 6C F9 | E8 | Temp.:73.688 |
| 149 | 50.0094ms | 30 | 1 | 2 | GetTemp(01) | | EF | 6D F9 | FD | Temp.:73.703 |
| 150 | 100.0086ms | 30 | 1 | 2 | GetTemp(01) | | EF | 6D F9 | FD | Temp.:73.703 |
| 151 | 150.0076ms | 30 | 1 | 2 | GetTemp(01) | | EF | 67 F9 | 7F | Temp.:73.609 |
| 152 | 200.0068ms | 30 | 1 | 2 | GetTemp(01) | | EF | 65 F9 | 55 | Temp.:73.578 |
| 153 | 250.0056ms | 30 | 1 | 2 | GetTemp(01) | | EF | 61 F9 | 01 | Temp.:73.516 |
| 154 | 300.005ms | 30 | 1 | 2 | GetTemp(01) | | EF | 60 F9 | 14 | Temp.:73.500 |
| 155 | 350.004ms | 30 | 1 | 2 | GetTemp(01) | | EF | 60 F9 | 14 | Temp.:73.500 |
| 156 | 400.0032ms | 30 | 1 | 2 | GetTemp(01) | | EF | 60 F9 | 14 | Temp.:73.500 |
| 157 | 450.0022ms | 30 | 1 | 2 | GetTemp(01) | | EF | 60 F9 | 14 | Temp.:73.500 |
| 158 | 500.001ms | 30 | 1 | 2 | GetTemp(01) | | EF | 5D F9 | 04 | Temp.:73.453 |
| 159 | 550.0004ms | 30 | 1 | 2 | GetTemp(01) | | EF | 5E F9 | 3B | Temp.:73.469 |
| 160 | 599.9996ms | 30 | 1 | 2 | GetTemp(01) | | EF | 5C F9 | 11 | Temp.:73.438 |
| 161 | 649.9986ms | 30 | 1 | 2 | GetTemp(01) | | EF | 5D F9 | 04 | Temp.:73.453 |
| 162 | 699.9978ms | 30 | 1 | 2 | GetTemp(01) | | EF | 5C F9 | 11 | Temp.:73.438 |

### Advance mode

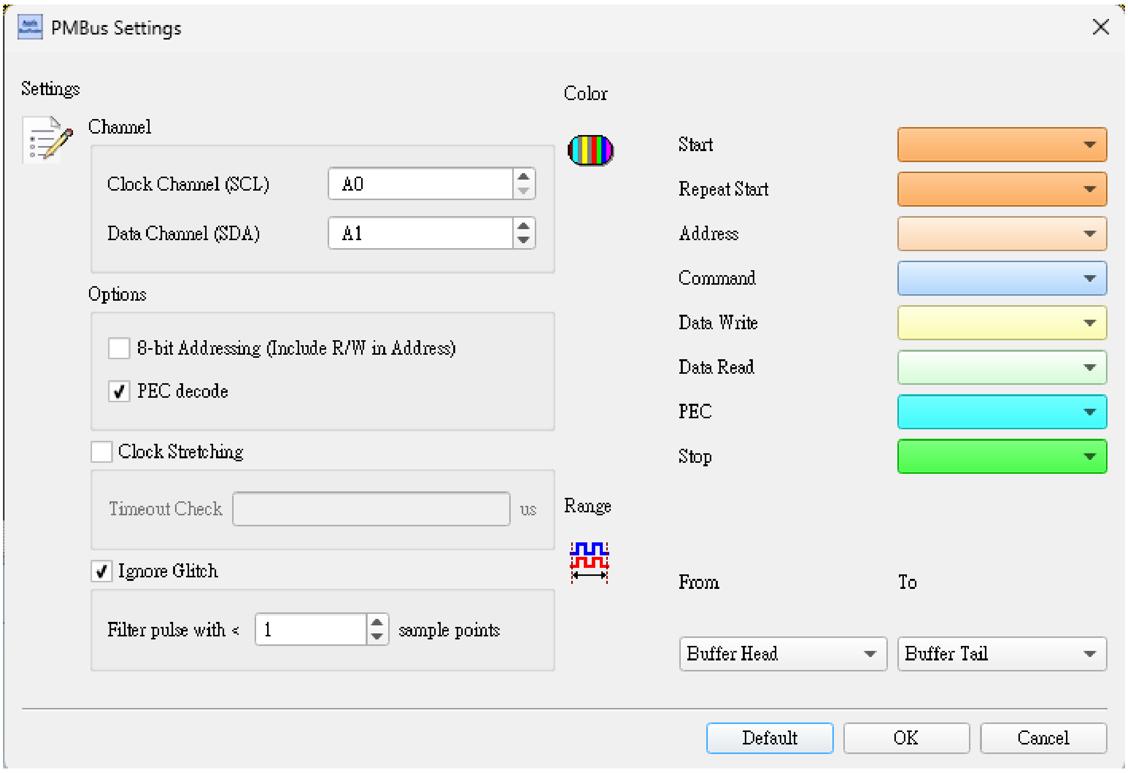


| | Timestamp | Field | Data(h) | Status | Information |
|---|---|---|---|---|---|
| 1 | -7.3498578s | Client Address | 30 | | Data rate:547.9KHz |
| 2 | -7.3498414s | Write Length | 01 | | |
| 3 | -7.3498268s | Read Length | 02 | | |
| 4 | -7.349812s | Cmd Code | GetTemp(01) | | |
| 5 | -7.3497974s | FCS | EF | | |
| 6 | -7.3497828s | Temp[7:0] | 75 | | |
| 7 | -7.349768s | Temp[15:8] | F9 | | Temperature:73.828 |
| 8 | -7.349768s | FCS | 02 | | |
| 9 | -7.2998588s | Client Address | 30 | | |
| 10 | -7.2998424s | Write Length | 01 | | |
| 11 | -7.2998276s | Read Length | 02 | | |
| 12 | -7.299813s | Cmd Code | GetTemp(01) | | |
| 13 | -7.2997984s | FCS | EF | | |
| 14 | -7.2997836s | Temp[7:0] | 75 | | |
| 15 | -7.299769s | Temp[15:8] | F9 | | Temperature:73.828 |
| 16 | -7.299769s | FCS | 02 | | |

## PMBus

The Power Management Bus ("PMBus") is an open standard protocol that defines a means of communicating with power conversion and other devices.

## Settings



PMBus Settings

Settings

Channel

Clock Channel (SCL): A0

Data Channel (SDA): A1

Options

☐ 8-bit Addressing (Include R/W in Address)

☑ PEC decode

☐ Clock Stretching

Timeout Check: us

☑ Ignore Glitch

Filter pulse with < 1 sample points

Color

| | |
|---|---|
| Start | 🟧 |
| Repeat Start | 🟧 |
| Address | 🟧 |
| Command | 🟦 |
| Data Write | 🟨 |
| Data Read | 🟩 |
| PEC | 🟦 |
| Stop | 🟩 |

Range

From: Buffer Head

To: Buffer Tail

### Channel:

1. **Clock Channel (SCK):** Transfer clock of PMBus.
2. **Data Channel (SDA):** Transfer data of PMBus.

### Options:

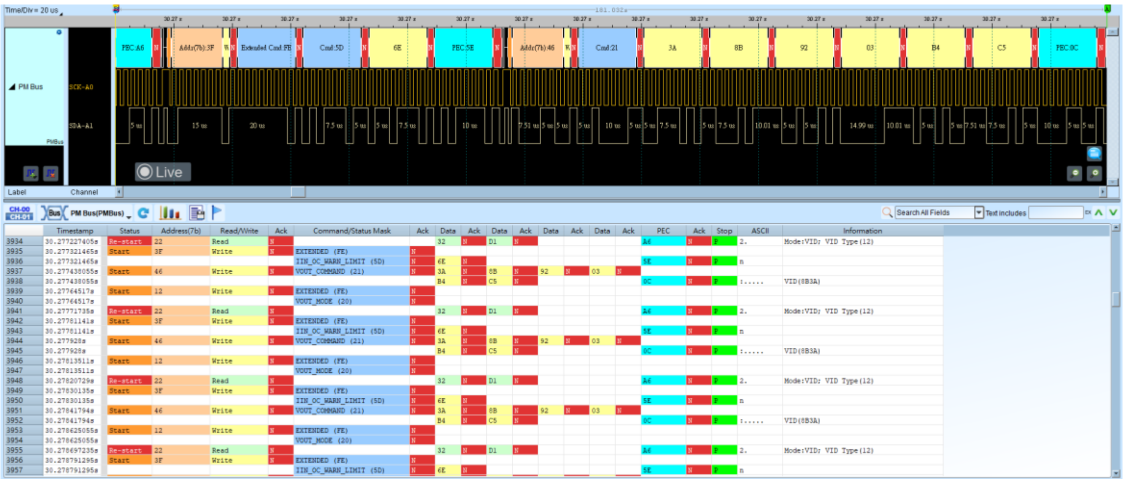
1. **8-bit addressing (Including R/W in Address):** Displays an 8-bit width address (7-bit width address plus 1-bit Rd/Wr).
2. **PEC decode:** Set whether the analyzed data contains PEC.

**Clock Stretching:** Set the time of Clock Stretching. Enabled when checked.

**Ignore Glitch:** Ignore noise caused by slow transitions when analyzing.

Enabled when checked.

**Result**

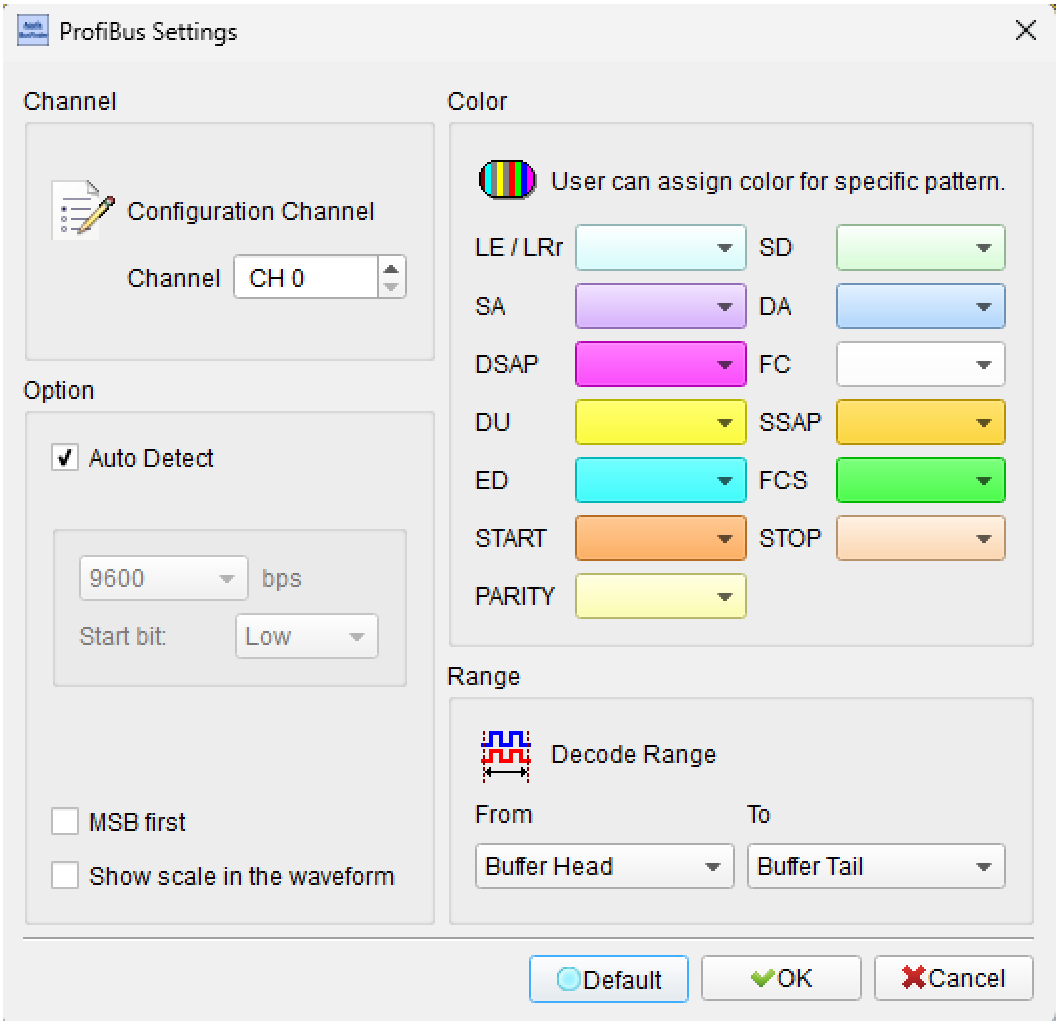


| | Timestamp | Status | Address(7b) | Read/Write | Ack | Command/Status Mask | Ack | Data | Ack | Data | Ack | Data | Ack | Data | Ack | PEC | Ack | Stop | ASCII | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3934 | 30.277227405s | Re-start | 22 | Read | N | | | 32 | N | D1 | N | | | | | A6 | N | P | 2. | Mode:VID; VID Type(12) |
| 3935 | 30.277321465s | Start | 3F | Write | N | EXTENDED (FE) | N | | | | | | | | | | | | | |
| 3936 | 30.277321465s | | | | | IIN_OC_WARN_LIMIT (5D) | N | 6E | N | | | | | | | 5E | N | P | n | |
| 3937 | 30.277438055s | Start | 46 | Write | N | VOUT_COMMAND (21) | N | 3A | N | 8B | N | 92 | N | 03 | N | | | | | |
| 3938 | 30.277438055s | | | | | | | B4 | N | C5 | N | | | | | 0C | N | P | :..... | VID(8B3A) |
| 3939 | 30.27764517s | Start | 12 | Write | N | EXTENDED (FE) | N | | | | | | | | | | | | | |
| 3940 | 30.27764517s | | | | | VOUT_MODE (20) | N | | | | | | | | | | | | | |
| 3941 | 30.27771735s | Re-start | 22 | Read | N | | | 32 | N | D1 | N | | | | | A6 | N | P | 2. | Mode:VID; VID Type(12) |
| 3942 | 30.27781141s | Start | 3F | Write | N | EXTENDED (FE) | N | | | | | | | | | | | | | |
| 3943 | 30.27781141s | | | | | IIN_OC_WARN_LIMIT (5D) | N | 6E | N | | | | | | | 5E | N | P | n | |
| 3944 | 30.277928s | Start | 46 | Write | N | VOUT_COMMAND (21) | N | 3A | N | 8B | N | 92 | N | 03 | N | | | | | |
| 3945 | 30.277928s | | | | | | | B4 | N | C5 | N | | | | | 0C | N | P | :..... | VID(8B3A) |
| 3946 | 30.27813511s | Start | 12 | Write | N | EXTENDED (FE) | N | | | | | | | | | | | | | |
| 3947 | 30.27813511s | | | | | VOUT_MODE (20) | N | | | | | | | | | | | | | |
| 3948 | 30.27820729s | Re-start | 22 | Read | N | | | 32 | N | D1 | N | | | | | A6 | N | P | 2. | Mode:VID; VID Type(12) |
| 3949 | 30.27830135s | Start | 3F | Write | N | EXTENDED (FE) | N | | | | | | | | | | | | | |
| 3950 | 30.27830135s | | | | | IIN_OC_WARN_LIMIT (5D) | N | 6E | N | | | | | | | 5E | N | P | n | |
| 3951 | 30.27841794s | Start | 46 | Write | N | VOUT_COMMAND (21) | N | 3A | N | 8B | N | 92 | N | 03 | N | | | | | |
| 3952 | 30.27841794s | | | | | | | B4 | N | C5 | N | | | | | 0C | N | P | :..... | VID(8B3A) |
| 3953 | 30.278625055s | Start | 12 | Write | N | EXTENDED (FE) | N | | | | | | | | | | | | | |
| 3954 | 30.278625055s | | | | | VOUT_MODE (20) | N | | | | | | | | | | | | | |
| 3955 | 30.278697235s | Re-start | 22 | Read | N | | | 32 | N | D1 | N | | | | | A6 | N | P | 2. | Mode:VID; VID Type(12) |
| 3956 | 30.278791295s | Start | 3F | Write | N | EXTENDED (FE) | N | | | | | | | | | | | | | |
| 3957 | 30.278791295s | | | | | IIN_OC_WARN_LIMIT (5D) | N | 6E | N | | | | | | | 5E | N | P | n | |

### ProfiBus

ProfiBus (PROcess Field Bus) was developed in 1987 by Siemens and other 14 companies and 5 research institutes in Germany, and is widely used in industrial control automation, transportation and power automation, etc. ProfiBus consists of 3 parts, PROFIBUS FMS (Fieldbus Message Specification), PROFIBUS DP (Decentralized Peripherals), and PROFIBUS PA (Process Automation). Currently, PROFIBUS DP and PROFIBUS PA are the most commonly used.

### Settings



ProfiBus Settings

Channel

Configuration Channel

Channel: CH 0

Option

☑ Auto Detect

9600 bps

Start bit: Low

□ MSB first

□ Show scale in the waveform

Color

User can assign color for specific pattern.

LE / LRr, SD, SA, DA, DSAP, FC, DU, SSAP, ED, FCS, START, STOP, PARITY

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

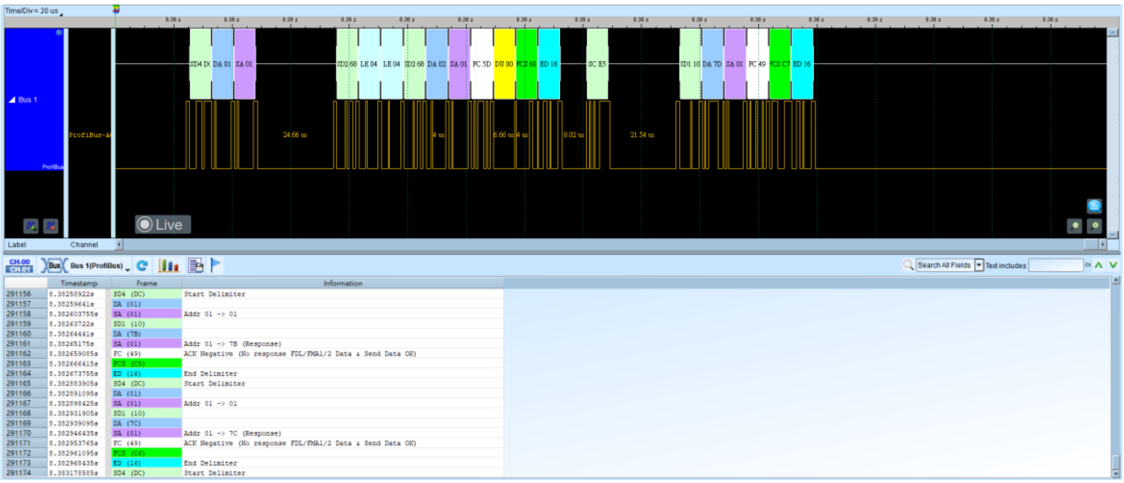
**Baud Rate/Auto Detect:** Set the baud rate manually or auto detect

**Start bit:** Set the Start bit to be High or Low.

**MSB First:** The default is LSB first; click it to change to MSB first.

**Show scale in the waveform:** Show the scale in the waveform section

### Result

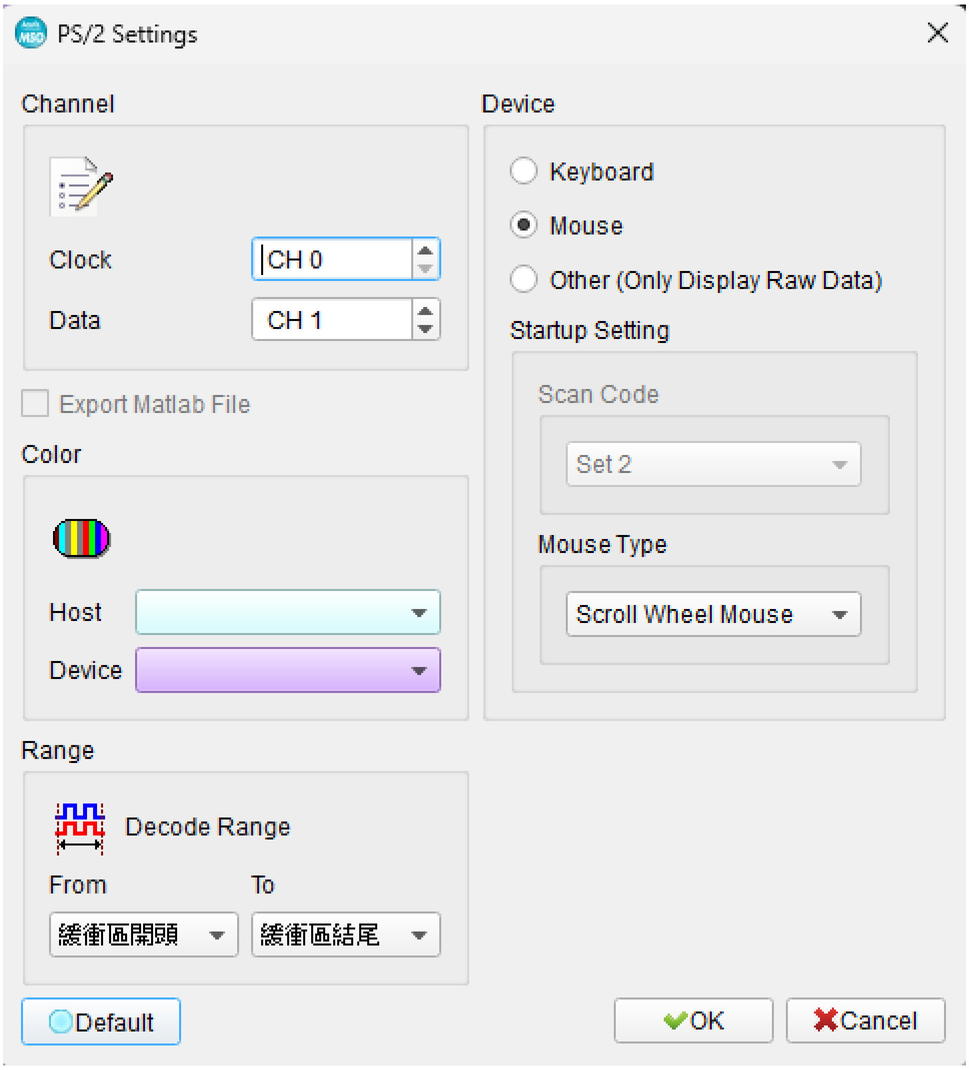


| | Timestamp | Frame | Information |
|---|---|---|---|
| 291156 | 8.38258922s | SD4 (DC) | Start Delimiter |
| 291157 | 8.38259641s | DA (01) | |
| 291158 | 8.382603755s | SA (01) | Addr 01 -> 01 |
| 291159 | 8.38263722s | SD1 (10) | |
| 291160 | 8.38264441s | DA (7B) | |
| 291161 | 8.38265175s | SA (01) | Addr 01 -> 7B (Response) |
| 291162 | 8.382659085s | FC (49) | ACK Negative (No response FDL/FMA1/2 Data & Send Data OK) |
| 291163 | 8.382666415s | FCS (C5) | |
| 291164 | 8.382673755s | ED (16) | End Delimiter |
| 291165 | 8.382883905s | SD4 (DC) | Start Delimiter |
| 291166 | 8.382891095s | DA (01) | |
| 291167 | 8.382898425s | SA (01) | Addr 01 -> 01 |
| 291168 | 8.382931905s | SD1 (10) | |
| 291169 | 8.382939095s | DA (7C) | |
| 291170 | 8.382946435s | SA (01) | Addr 01 -> 7C (Response) |
| 291171 | 8.382953765s | FC (49) | ACK Negative (No response FDL/FMA1/2 Data & Send Data OK) |
| 291172 | 8.382961095s | FCS (C6) | |
| 291173 | 8.382968435s | ED (16) | End Delimiter |
| 291174 | 8.383178585s | SD4 (DC) | Start Delimiter |

## PS/2

PS/2 is a bi-directional synchronous serial protocol for communication between a keyboard or mouse and a PC, developed by IBM, it consists of six pins, namely Clock, Data, +5v, Ground and two blank pins. PS/2 uses a bi-directional synchronous transmission method, whereby data is exchanged between the two ends of the communication via Clock and Data.

### Settings



PS/2 Settings

Channel

Clock: CH 0

Data: CH 1

☐ Export Matlab File

Color

Host

Device

Range

Decode Range

From: 緩衝區開頭

To: 緩衝區結尾

Device

○ Keyboard

● Mouse

○ Other (Only Display Raw Data)

Startup Setting

Scan Code: Set 2

Mouse Type: Scroll Wheel Mouse

**Channel**: Show the selected channels.

**Export MATLAB file:** Export the data with MATLAB format as the following:

Time = [25.78484 25.785985 … ]

Description = [DH DH … ]     DH = Device to Host, HD = Host to Device

Data = [ 58 FA 02 FA C4 … ]

The file (PS2_Matlab.m) will be saved at work directory

**Device:**

**Keyboard:** Assign the current device is PS/2 keyboard.

**Mouse:** Assign the current device is PS/2 mouse.

**Other (Only Display Raw Data):** Other PS/2 device, only display raw data in this mode.

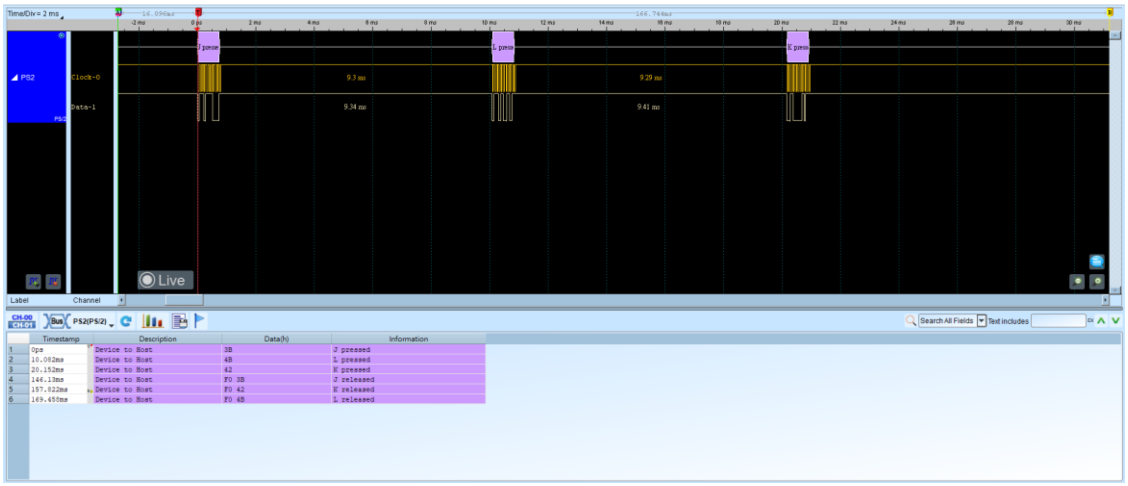
**Startup Setting:**

**Scan Code:** Set the Scan Code Set of PS/2 keyboard.

**Mouse Type:** Set the Type of the PS/2 mouse.

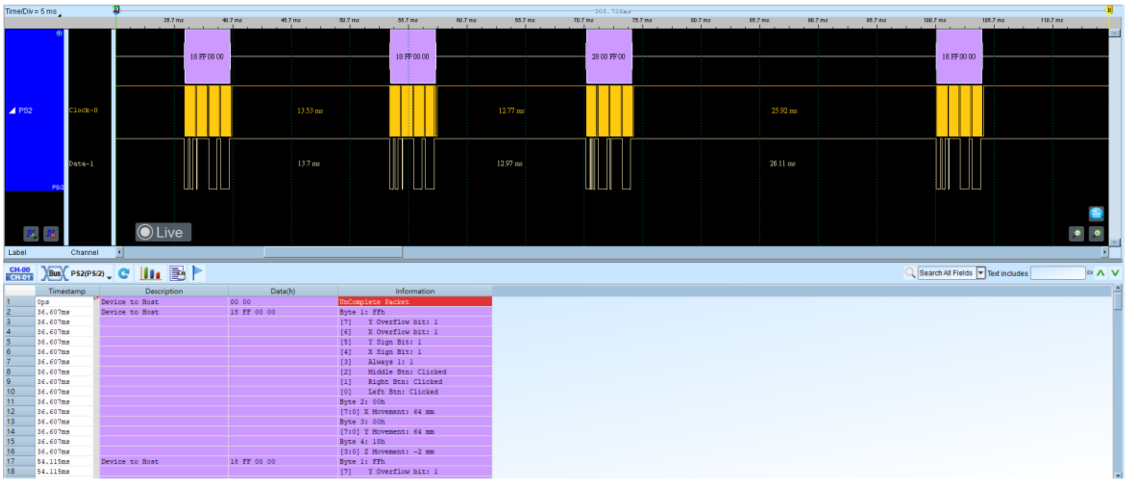
**Result**

**Keyboard:**



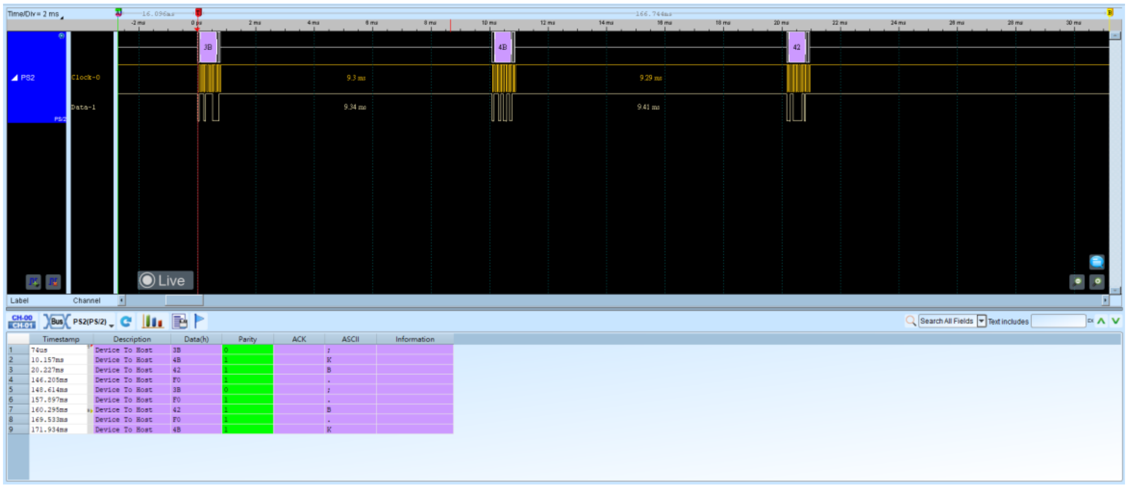
|  | Timestamp | Description | Data(h) | Information |
|---|---|---|---|---|
| 1 | 0ps | Device to Host | 3B | J pressed |
| 2 | 10.082ms | Device to Host | 4B | L pressed |
| 3 | 20.152ms | Device to Host | 42 | K pressed |
| 4 | 146.13ms | Device to Host | F0 3B | J released |
| 5 | 157.822ms | Device to Host | F0 42 | K released |
| 6 | 169.458ms | Device to Host | F0 4B | L released |

**Mouse:**



| | Timestamp | Description | Data(h) | Information |
|---|---|---|---|---|
| 1 | 0ps | Device to Host | 00 00 | UnComplete Packet |
| 2 | 36.607ms | Device to Host | 18 FF 00 00 | Byte 1: FFh |
| 3 | 36.607ms | | | [7] Y Overflow bit: 1 |
| 4 | 36.607ms | | | [6] X Overflow bit: 1 |
| 5 | 36.607ms | | | [5] Y Sign Bit: 1 |
| 6 | 36.607ms | | | [4] X Sign Bit: 1 |
| 7 | 36.607ms | | | [3] Always 1: 1 |
| 8 | 36.607ms | | | [2] Middle Btn: Clicked |
| 9 | 36.607ms | | | [1] Right Btn: Clicked |
| 10 | 36.607ms | | | [0] Left Btn: Clicked |
| 11 | 36.607ms | | | Byte 2: 00h |
| 12 | 36.607ms | | | [7:0] X Movement: 64 mm |
| 13 | 36.607ms | | | Byte 3: 00h |
| 14 | 36.607ms | | | [7:0] Y Movement: 64 mm |
| 15 | 36.607ms | | | Byte 4: 18h |
| 16 | 36.607ms | | | [3:0] Z Movement: -2 mm |
| 17 | 54.115ms | Device to Host | 18 FF 00 00 | Byte 1: FFh |
| 18 | 54.115ms | | | [7] Y Overflow bit: 1 |

## Other (Only Display Raw Data):

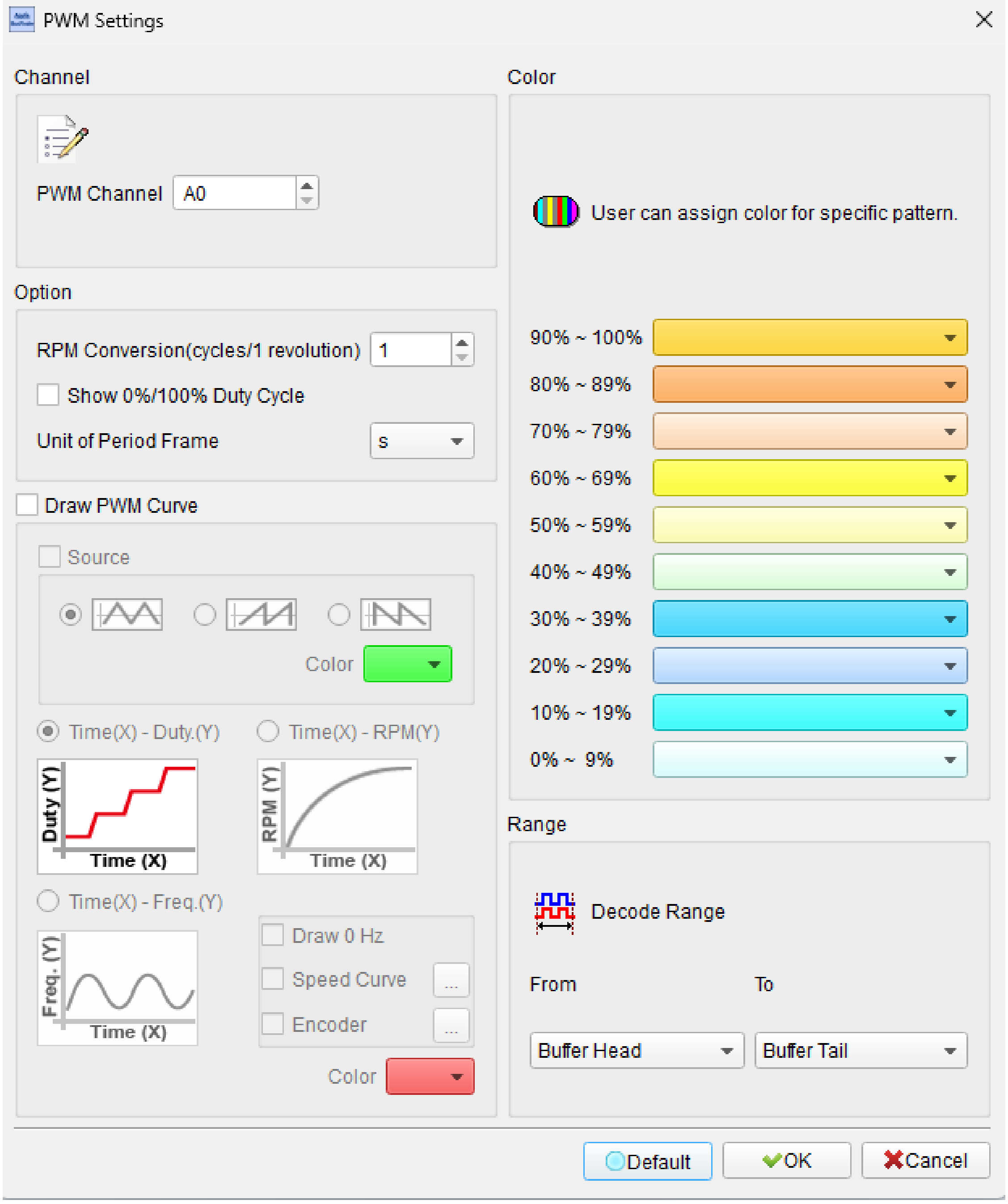


| | Timestamp | Description | Data(h) | Parity | ACK | ASCII | Information |
|---|---|---|---|---|---|---|---|
| 1 | 74us | Device To Host | 3B | 0 | | ; | |
| 2 | 10.157ms | Device To Host | 4B | 1 | | K | |
| 3 | 20.227ms | Device To Host | 42 | 1 | | B | |
| 4 | 146.205ms | Device To Host | F0 | 1 | | . | |
| 5 | 148.614ms | Device To Host | 3B | 0 | | ; | |
| 6 | 157.897ms | Device To Host | F0 | 1 | | . | |
| 7 | 160.295ms | Device To Host | 42 | 1 | | B | |
| 8 | 169.533ms | Device To Host | F0 | 1 | | . | |
| 9 | 171.934ms | Device To Host | 4B | 1 | | K | |

## PWM

PWM (Pulse Width Modulation), called pulse width modulation, it is not a bus analysis protocol. It is a very effective technique to control analog circuits by utilizing the pulse width cycle, which is widely used in some rotational speed control, brightness control and temperature control.

## Settings



PWM Settings

Channel

PWM Channel A0

Option

RPM Conversion(cycles/1 revolution) 1

□ Show 0%/100% Duty Cycle

Unit of Period Frame s

□ Draw PWM Curve

□ Source

● ○ ○

Color 🟩

● Time(X) - Duty.(Y) ○ Time(X) - RPM(Y)

○ Time(X) - Freq.(Y)

□ Draw 0 Hz

□ Speed Curve ...

□ Encoder ...

Color 🟥

Color

User can assign color for specific pattern.

| | |
|---|---|
| 90% ~ 100% | 🟨 |
| 80% ~ 89% | 🟧 |
| 70% ~ 79% | 🟧 |
| 60% ~ 69% | 🟨 |
| 50% ~ 59% | 🟨 |
| 40% ~ 49% | 🟩 |
| 30% ~ 39% | 🟦 |
| 20% ~ 29% | 🟦 |
| 10% ~ 19% | 🟦 |
| 0% ~ 9% | ⬜ |

Range

Decode Range

From Buffer Head

To Buffer Tail

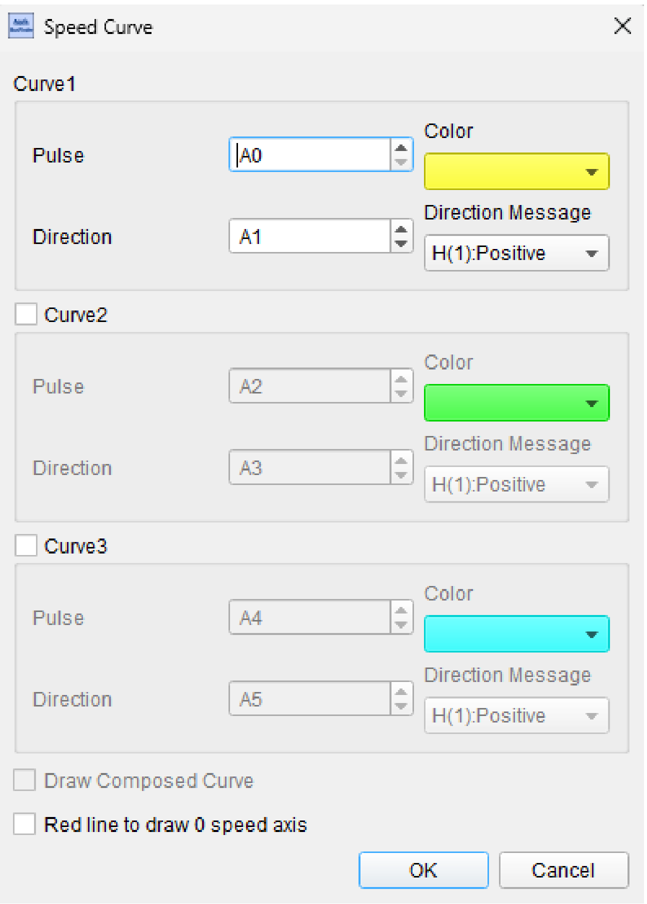
**PWM Channel**: Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Option:**

1. **RPM Conversion (cycles/1 revolution):** Set the number of PWM cycles.
2. **Show 0% and 100% Duty Cycle:** When Time (X)-Period (Y) is selected for plotting, check the box to draw 0% and 100% periods, then the curve will be drawn; otherwise, the curve will not be drawn. If 0% follows 100% or 100% follows 0%, the curve of the two periods will not be drawn.
3. **Unit of Period Frame:** Set the unit of time, user can set s, ms, us.

**Draw PWM curve:**

1. **Source:** Show the source waveform of the PWM.
2. **Time(X)-Duty(Y):** Show the curve diagram with Time(X) and Duty(Y)
3. **Time(X)-Freq.(Y):** Show the curve diagram with Time(X) and Freq.(Y)
4. **Time(X)-RPM(Y):** Show the curve diagram with Time(X) and RPM(Y)
5. **Draw 0 Hz:** When select the Time(X)-Freq.(Y) drawing and check the item Draw 0 Hz, will show the Frequency from 0 Hz at Y axis.
6. **Speed Curve:** Supports up to three Curves for overlay drawing.



Speed Curve

Curve1

Pulse: A0 — Color: 🟨

Direction: A1 — Direction Message: H(1):Positive

□ Curve2

Pulse: A2 — Color: 🟩

Direction: A3 — Direction Message: H(1):Positive

□ Curve3

Pulse: A4 — Color: 🟦

Direction: A5 — Direction Message: H(1):Positive

□ Draw Composed Curve

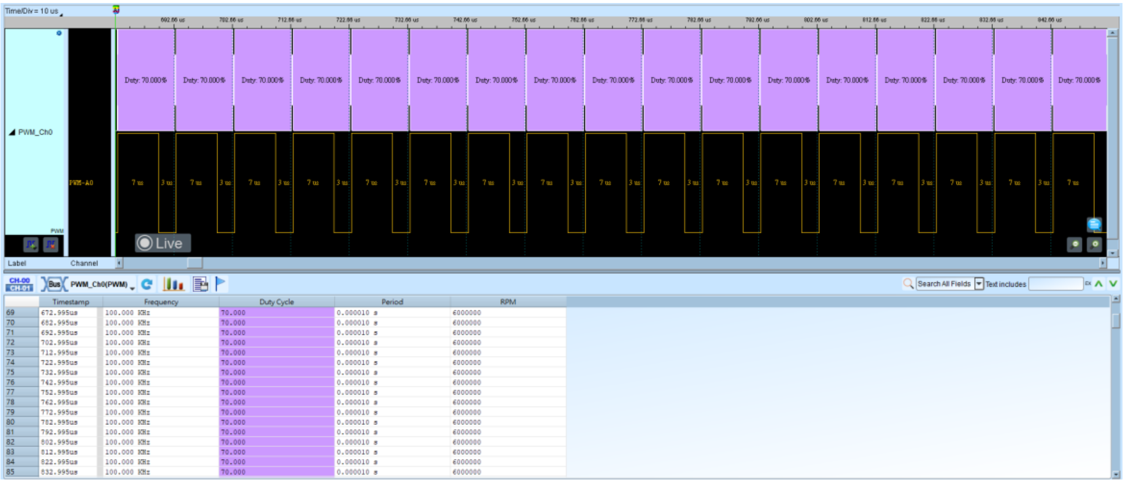
□ Red line to draw 0 speed axis

**Pulse/Direction:** Setting the pulse and direction of the signal channel on the instrument.

**Direction Message:** Set the direction source to H(1): Positive or L(0): Positive.

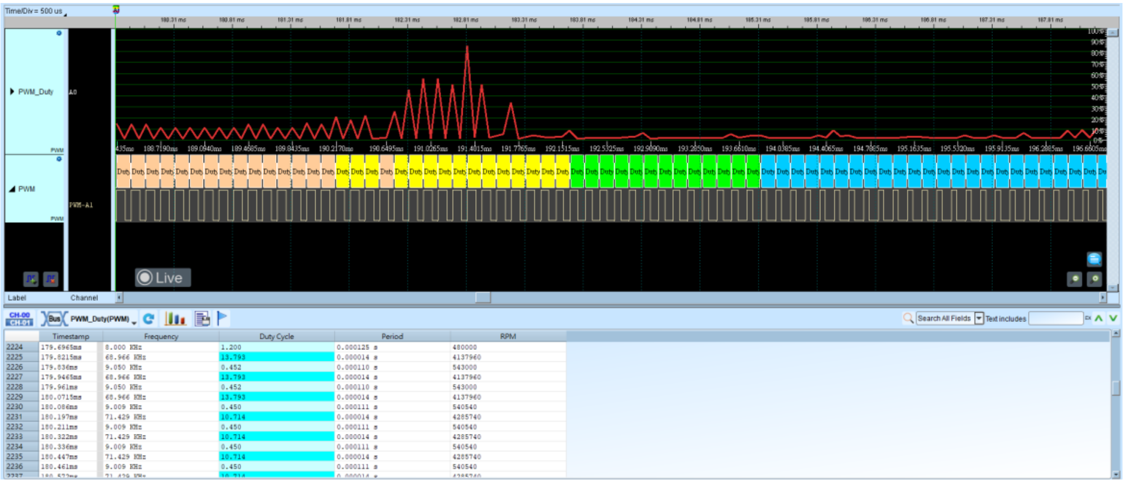
7. **Encoder:** Encode the PWM parsing result, enabled when checked.

## Result



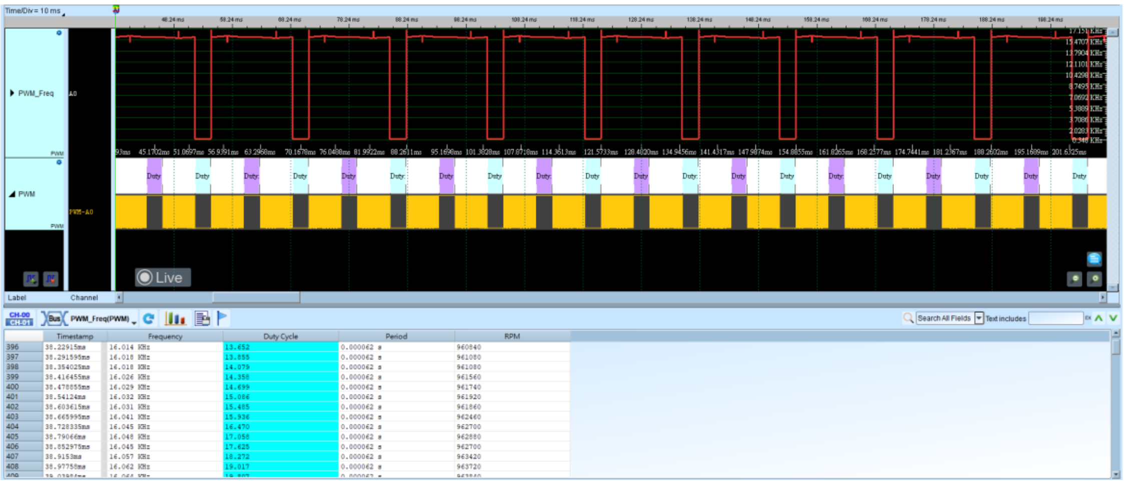
| | Timestamp | Frequency | Duty Cycle | Period | RPM |
|---|---|---|---|---|---|
| 69 | 672.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 70 | 682.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 71 | 692.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 72 | 702.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 73 | 712.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 74 | 722.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 75 | 732.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 76 | 742.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 77 | 752.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 78 | 762.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 79 | 772.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 80 | 782.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 81 | 792.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 82 | 802.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 83 | 812.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 84 | 822.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |
| 85 | 832.995us | 100.000 KHz | 70.000 | 0.000010 s | 6000000 |

Select Time(X)-Duty(Y)



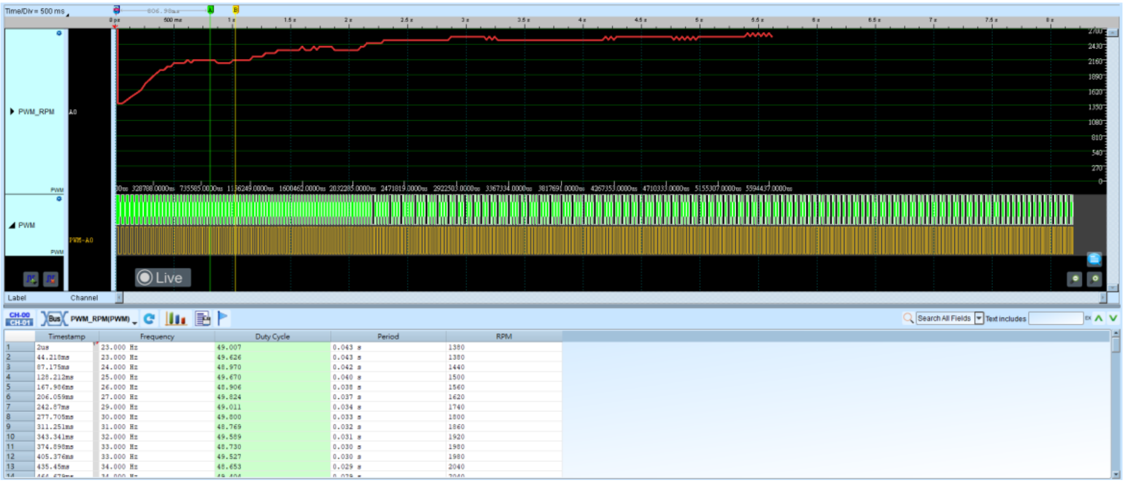
| | Timestamp | Frequency | Duty Cycle | Period | RPM |
|---|---|---|---|---|---|
| 2224 | 179.6965ms | 8.000 KHz | 1.200 | 0.000125 s | 480000 |
| 2225 | 179.8215ms | 68.966 KHz | 13.793 | 0.000014 s | 4137960 |
| 2226 | 179.836ms | 9.050 KHz | 0.452 | 0.000110 s | 543000 |
| 2227 | 179.9465ms | 68.966 KHz | 13.793 | 0.000014 s | 4137960 |
| 2228 | 179.961ms | 9.050 KHz | 0.452 | 0.000110 s | 543000 |
| 2229 | 180.0715ms | 68.966 KHz | 13.793 | 0.000014 s | 4137960 |
| 2230 | 180.086ms | 9.009 KHz | 0.450 | 0.000111 s | 540540 |
| 2231 | 180.197ms | 71.429 KHz | 10.714 | 0.000014 s | 4285740 |
| 2232 | 180.211ms | 9.009 KHz | 0.450 | 0.000111 s | 540540 |
| 2233 | 180.322ms | 71.429 KHz | 10.714 | 0.000014 s | 4285740 |
| 2234 | 180.336ms | 9.009 KHz | 0.450 | 0.000111 s | 540540 |
| 2235 | 180.447ms | 71.429 KHz | 10.714 | 0.000014 s | 4285740 |
| 2236 | 180.461ms | 9.009 KHz | 0.450 | 0.000111 s | 540540 |
| 2237 | 180.572ms | 71.429 KHz | 10.714 | 0.000014 s | 4285740 |

## Select Time(X)-Freq.(Y)



| | Timestamp | Frequency | Duty Cycle | Period | RPM |
|---|---|---|---|---|---|
| 396 | 38.22915ms | 16.014 KHz | 13.652 | 0.000062 s | 960840 |
| 397 | 38.291595ms | 16.018 KHz | 13.855 | 0.000062 s | 961080 |
| 398 | 38.354025ms | 16.018 KHz | 14.079 | 0.000062 s | 961080 |
| 399 | 38.416455ms | 16.026 KHz | 14.358 | 0.000062 s | 961560 |
| 400 | 38.478855ms | 16.029 KHz | 14.699 | 0.000062 s | 961740 |
| 401 | 38.54124ms | 16.032 KHz | 15.086 | 0.000062 s | 961920 |
| 402 | 38.603615ms | 16.031 KHz | 15.485 | 0.000062 s | 961860 |
| 403 | 38.665995ms | 16.041 KHz | 15.936 | 0.000062 s | 962460 |
| 404 | 38.728335ms | 16.045 KHz | 16.470 | 0.000062 s | 962700 |
| 405 | 38.79066ms | 16.048 KHz | 17.058 | 0.000062 s | 962880 |
| 406 | 38.852975ms | 16.045 KHz | 17.625 | 0.000062 s | 962700 |
| 407 | 38.9153ms | 16.057 KHz | 18.272 | 0.000062 s | 963420 |
| 408 | 38.97758ms | 16.062 KHz | 19.017 | 0.000062 s | 963720 |

## Select Time(X)-RPM(Y)

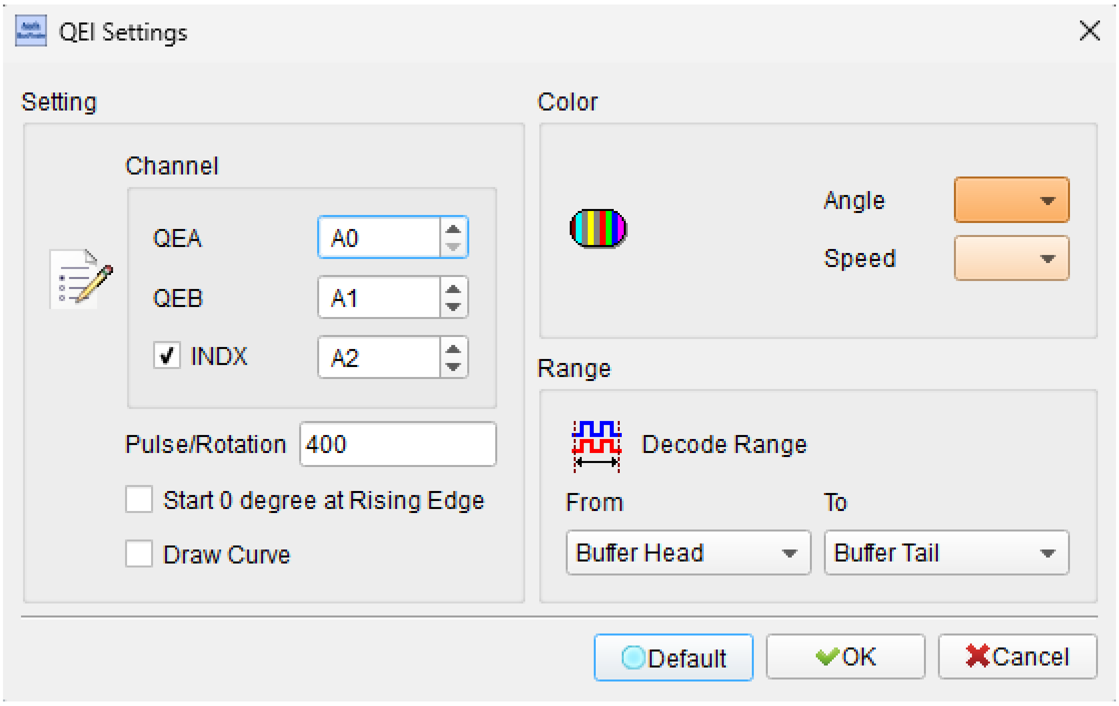


| | Timestamp | Frequency | Duty Cycle | Period | RPM |
|---|---|---|---|---|---|
| 1 | 2us | 23.000 Hz | 49.007 | 0.043 s | 1380 |
| 2 | 44.218ms | 23.000 Hz | 49.626 | 0.043 s | 1380 |
| 3 | 87.175ms | 24.000 Hz | 48.970 | 0.042 s | 1440 |
| 4 | 128.212ms | 25.000 Hz | 49.670 | 0.040 s | 1500 |
| 5 | 167.986ms | 26.000 Hz | 48.906 | 0.038 s | 1560 |
| 6 | 206.059ms | 27.000 Hz | 49.824 | 0.037 s | 1620 |
| 7 | 242.87ms | 29.000 Hz | 49.011 | 0.034 s | 1740 |
| 8 | 277.705ms | 30.000 Hz | 49.800 | 0.033 s | 1800 |
| 9 | 311.251ms | 31.000 Hz | 48.769 | 0.032 s | 1860 |
| 10 | 343.341ms | 32.000 Hz | 49.589 | 0.031 s | 1920 |
| 11 | 374.898ms | 33.000 Hz | 48.730 | 0.030 s | 1980 |
| 12 | 405.376ms | 33.000 Hz | 49.527 | 0.030 s | 1980 |
| 13 | 435.45ms | 34.000 Hz | 48.653 | 0.029 s | 2040 |

## QEI

QEI (Quadrature Encoder Interface) is a feedback signal (encoder) for small motor control. The QEA/QEB signal is used to obtain the RPM of the motor, and the INDX signal can be used to obtain the angle of rotation of the motor.

### Settings



QEI Settings

Setting

Channel

QEA: A0

QEB: A1

☑ INDX: A2

Pulse/Rotation: 400

□ Start 0 degree at Rising Edge

□ Draw Curve

Color

Angle

Speed

Range

Decode Range

From: Buffer Head

To: Buffer Tail

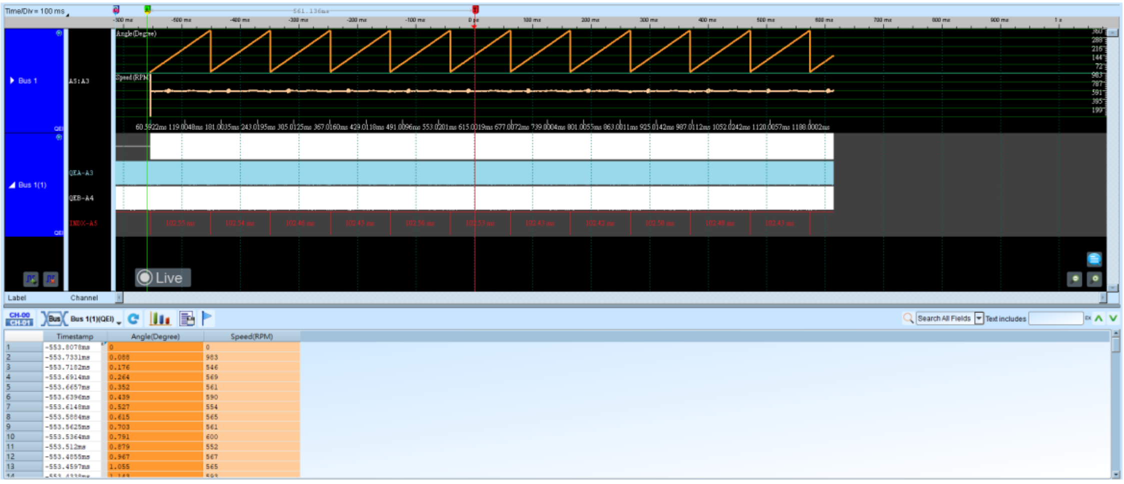
**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Pulse/Rotation:** Set the number of lines of the encoder, the default is 400 lines.

**Start 0 degree at Rising Edge:** Sets the rising edge of the INDX signal to an angle of 0 degrees; the default is for the falling edge to be at an angle of 0 degrees. Enabled when checked.

**Draw Curve:** Whether draw the angle/speed curve. Enabled when checked.

## Result

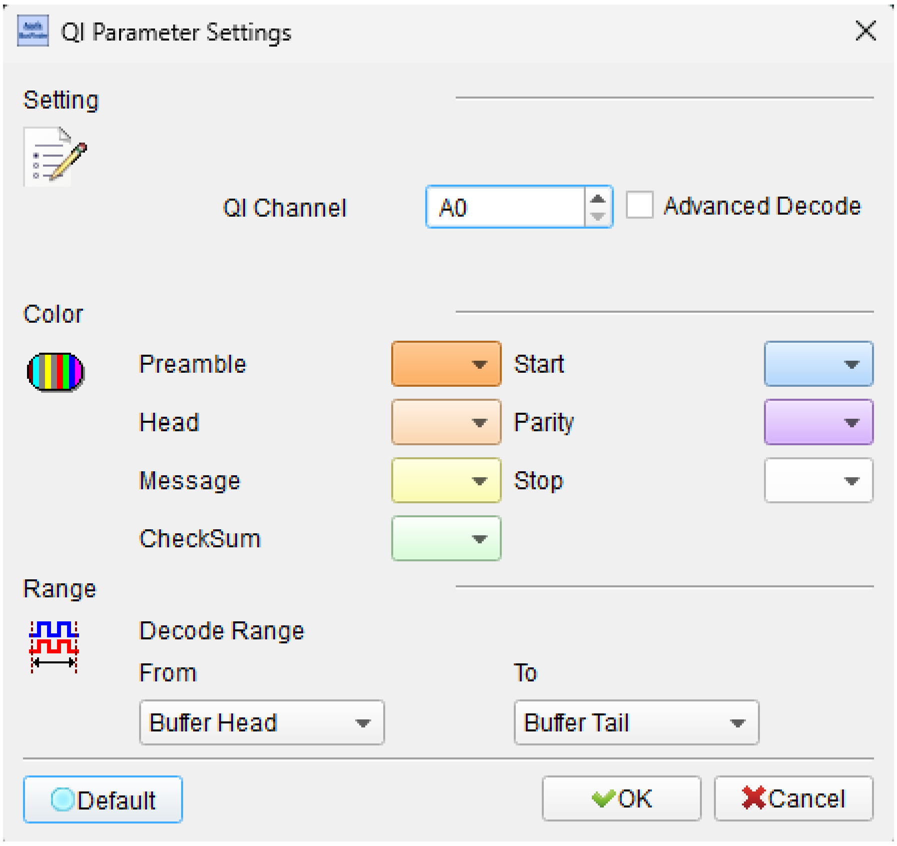


|  | Timestamp | Angle(Degree) | Speed(RPM) |
|---|---|---|---|
| 1 | -553.8078ms | 0 | 0 |
| 2 | -553.7331ms | 0.088 | 983 |
| 3 | -553.7182ms | 0.176 | 546 |
| 4 | -553.6914ms | 0.264 | 569 |
| 5 | -553.6657ms | 0.352 | 561 |
| 6 | -553.6396ms | 0.439 | 590 |
| 7 | -553.6148ms | 0.527 | 554 |
| 8 | -553.5884ms | 0.615 | 565 |
| 9 | -553.5625ms | 0.703 | 561 |
| 10 | -553.5364ms | 0.791 | 600 |
| 11 | -553.512ms | 0.879 | 552 |
| 12 | -553.4855ms | 0.967 | 567 |
| 13 | -553.4597ms | 1.055 | 565 |
| 14 | -553.4338ms | 1.143 | 593 |

## QI

QI is a contactless power transfer protocol published by Wireless Power Consortium (WPC). It is a method of contactless power transfer from a Base Station to a Mobile Device, which is based on near field magnetic induction between coils.

### Settings



QI Parameter Settings

Setting

QI Channel: A0

□ Advanced Decode

Color

| | | | |
|---|---|---|---|
| Preamble | 🟧 | Start | 🟦 |
| Head | 🟧 | Parity | 🟪 |
| Message | 🟨 | Stop | □ |
| CheckSum | 🟩 | | |

Range

Decode Range

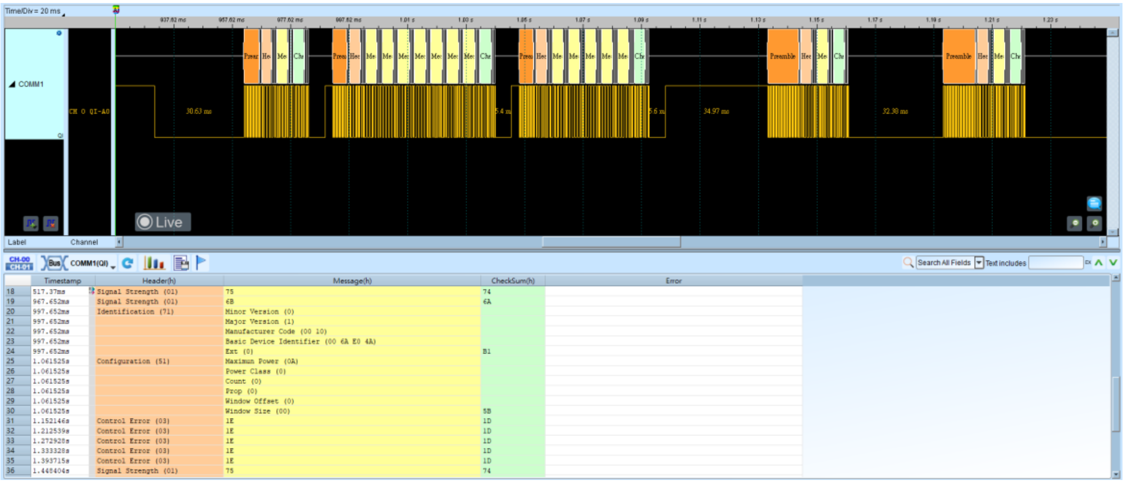
From: Buffer Head

To: Buffer Tail

**QI Channel:** Show the selected channel.

**Advance Decode:** show detail message decode

## Result

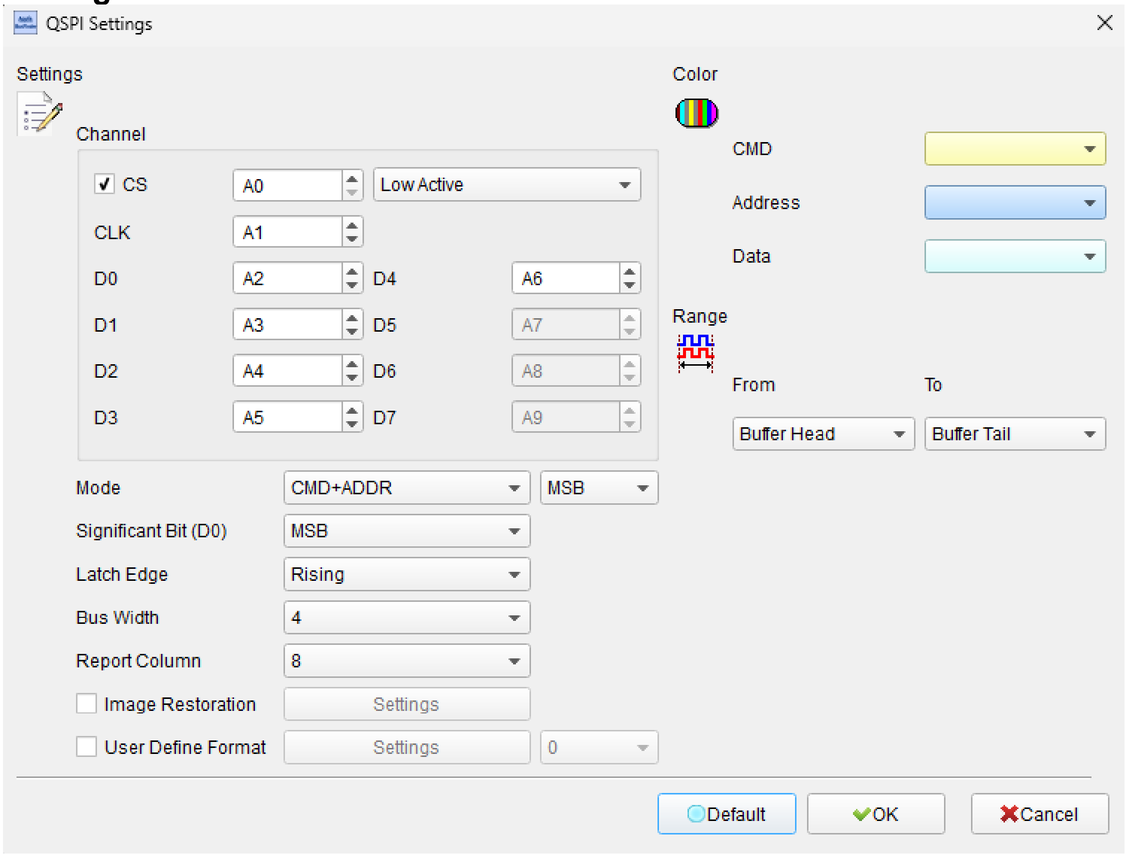


| | Timestamp | Header(h) | Message(h) | CheckSum(h) | Error |
|---|---|---|---|---|---|
| 18 | 517.37ms | Signal Strength (01) | 75 | 74 | |
| 19 | 967.652ms | Signal Strength (01) | 6B | 6A | |
| 20 | 997.652ms | Identification (71) | Minor Version (0) | | |
| 21 | 997.652ms | | Major Version (1) | | |
| 22 | 997.652ms | | Manufacturer Code (00 10) | | |
| 23 | 997.652ms | | Basic Device Identifier (00 6A E0 4A) | | |
| 24 | 997.652ms | | Ext (0) | B1 | |
| 25 | 1.061525s | Configuration (51) | Maximun Power (0A) | | |
| 26 | 1.061525s | | Power Class (0) | | |
| 27 | 1.061525s | | Count (0) | | |
| 28 | 1.061525s | | Prop (0) | | |
| 29 | 1.061525s | | Window Offset (0) | | |
| 30 | 1.061525s | | Window Size (00) | 5B | |
| 31 | 1.152146s | Control Error (03) | 1E | 1D | |
| 32 | 1.212539s | Control Error (03) | 1E | 1D | |
| 33 | 1.272928s | Control Error (03) | 1E | 1D | |
| 34 | 1.333328s | Control Error (03) | 1E | 1D | |
| 35 | 1.393715s | Control Error (03) | 1E | 1D | |
| 36 | 1.448404s | Signal Strength (01) | 75 | 74 | |

## QSPI

QSPI is an enhanced version of SPI, which uses additional data lines to increase throughput in DATA; The data line of QSPI is bidirectional and belongs to parallel transmission

### Settings



☑ CS: A0, Low Active

CLK: A1

D0: A2 D4: A6

D1: A3 D5: A7

D2: A4 D6: A8

D3: A5 D7: A9

Mode: CMD+ADDR, MSB

Significant Bit (D0): MSB

Latch Edge: Rising

Bus Width: 4

Report Column: 8

□ Image Restoration: Settings

□ User Define Format: Settings, 0

CMD

Address

Data

From: Buffer Head

To: Buffer Tail

**Channel:**

**CS: Capture data w**hen CS edge is falling.

**CLK:** Clock channel.

**D0-D7:** Customizable data channel.

**Mode**: Set the mode of QSPI. It can be set to CMD+ADDR, CMD or DATA, and it can be set to MSB first or LSB first.

**Significant Bit(D0):** D0 is the MSB or LSB of the data arrangement,

3. Take Bus Width = 4 MSB as an example, the Byte combination is D0 D1 D2 D3 D0 D1 D2 D3
4. Take Bus Width = 4 LSB as an example, the Byte combination is D3

D2 D1 D0 D3 D2 D1 D0

**Latch Edge:** Rising/Falling/Both can be selected as the data collection location

**Bus Width:** Optional data 1, 2, 4, 8 lines

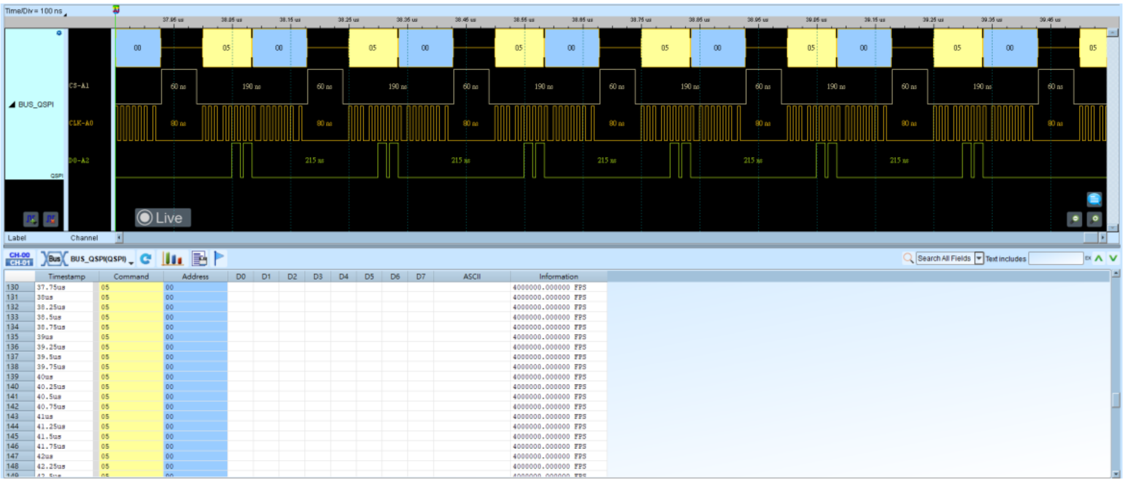
**Report Column:** Report presentation mode, 8/16 fields can be selected

**Image Restoration:** Save the parsed waveform as a picture for additional reading of the configuration file. Enabled when checked.

**User Defined Format:** User-defined parsing format, which can read additional configure files. Enabled when checked.

**** Please note that Image Restoration and User Define Format cannot be enabled at the same time **.**

**Result**

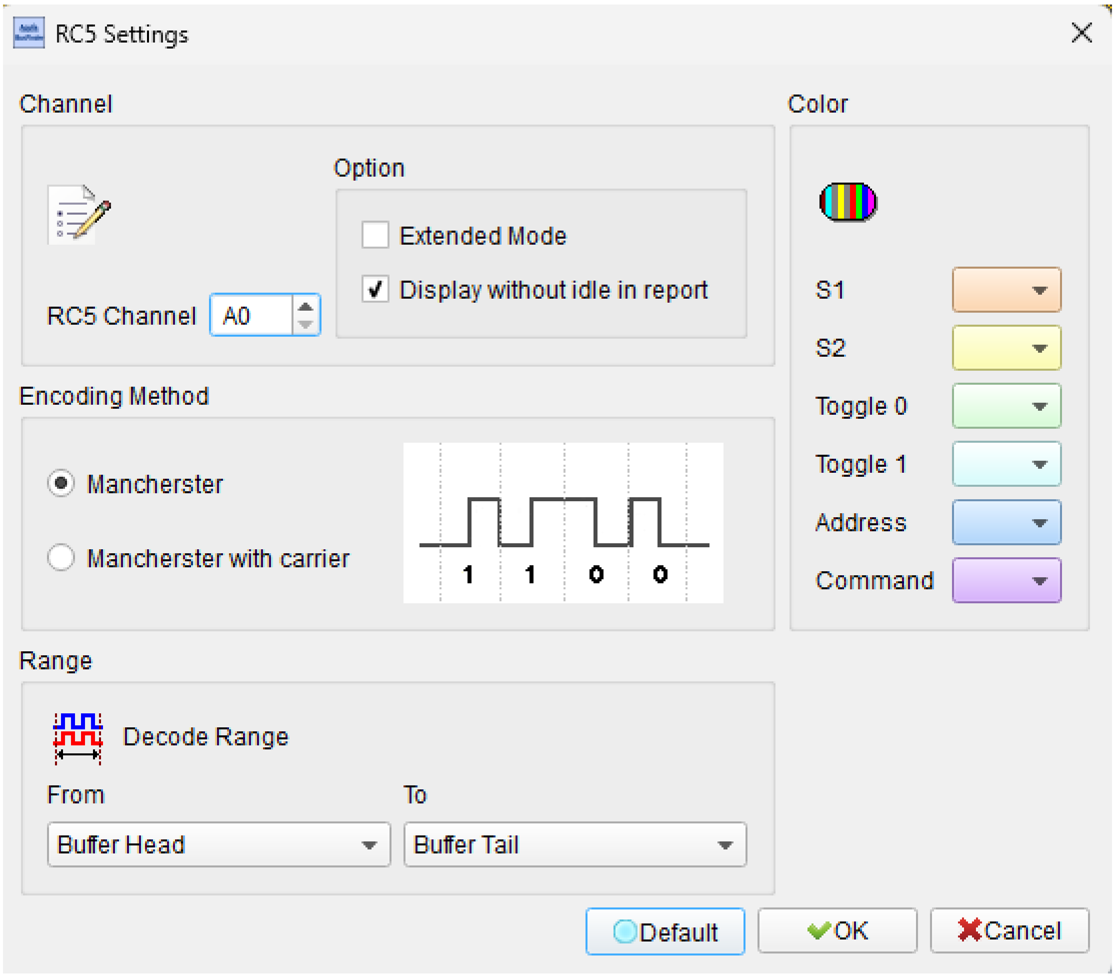


| | Timestamp | Command | Address | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 130 | 37.75us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 131 | 38us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 132 | 38.25us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 133 | 38.5us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 134 | 38.75us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 135 | 39us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 136 | 39.25us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 137 | 39.5us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 138 | 39.75us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 139 | 40us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 140 | 40.25us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 141 | 40.5us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 142 | 40.75us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 143 | 41us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 144 | 41.25us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 145 | 41.5us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 146 | 41.75us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 147 | 42us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 148 | 42.25us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |
| 149 | 42.5us | 05 | 00 | | | | | | | | | | 4000000.000000 FPS |

## RC-5

The RC-5 code from Philips is possibly the most used protocol by hobbyists, probably because of the wide availability of cheap remote controls. The protocol is well defined for different device types ensuring compatibility with your whole entertainment system.

### Settings



RC5 Settings

Channel

RC5 Channel: A0

Option

□ Extended Mode

☑ Display without idle in report

Encoding Method

● Mancherster

○ Mancherster with carrier

Range

Decode Range

From: Buffer Head

To: Buffer Tail

Color

**Channel**: Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Option:**

1. **Extended mode:** When the Extended enabled, the S2 will be converted

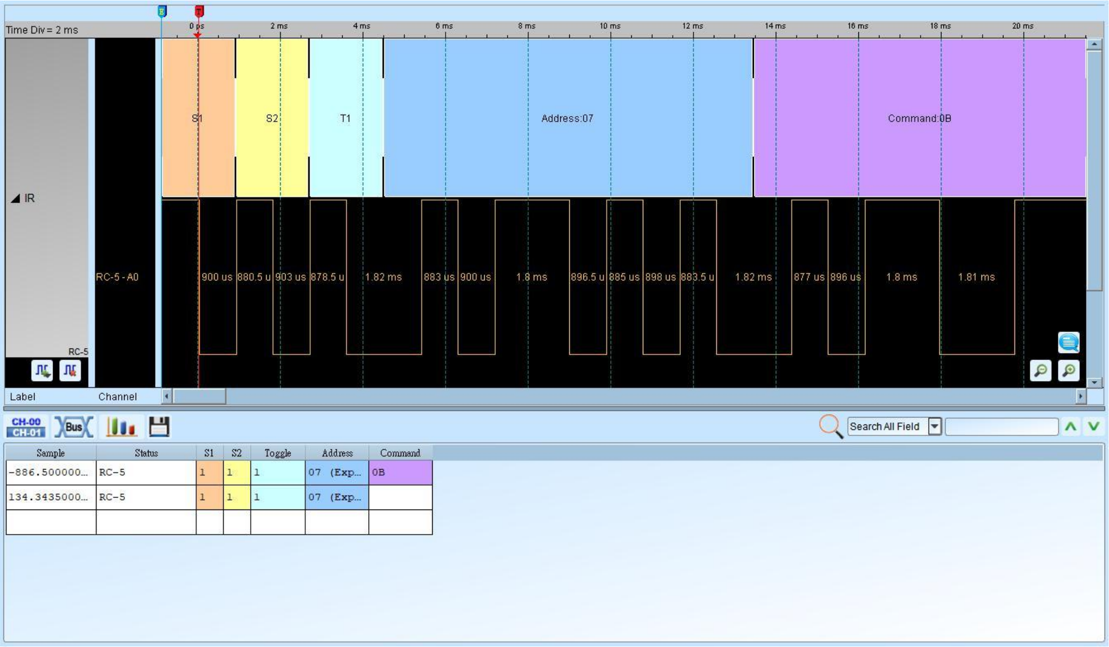
into seventh bit of the Command. There is an Extend Command on the Waveform Window.

2. **Display without idle in report:** It will not idle on the Report Window for the user to observe and analyze data.

**Encoding Method:** Mancherster mode and Mancherster with carrier mode.

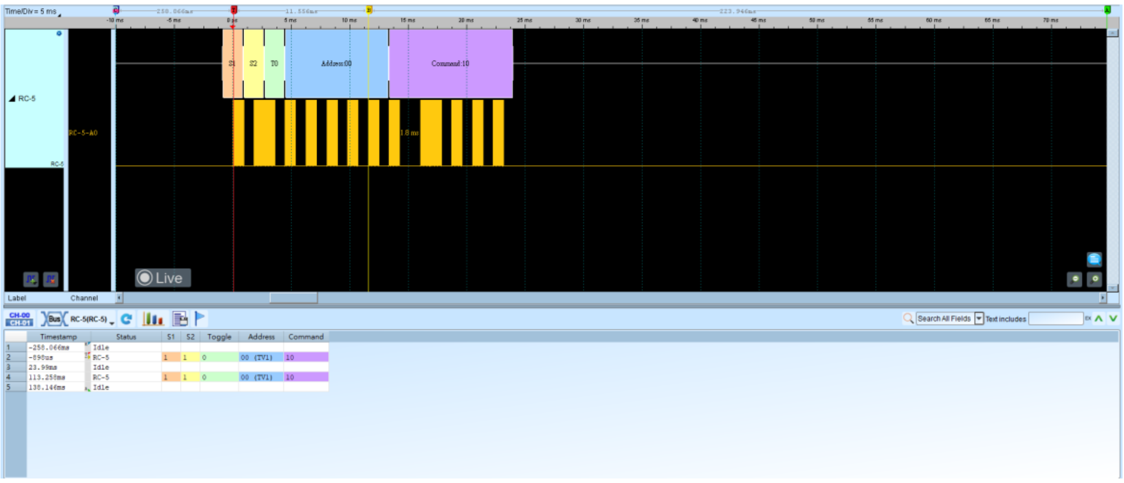
Result

RC5 without carrier



| Sample | Status | S1 | S2 | Toggle | Address | Command |
|---|---|---|---|---|---|---|
| -886.500000... | RC-5 | 1 | 1 | 1 | 07 (Exp... | 0B |
| 134.3435000... | RC-5 | 1 | 1 | 1 | 07 (Exp... | |
| | | | | | | |

RC5 with carrier

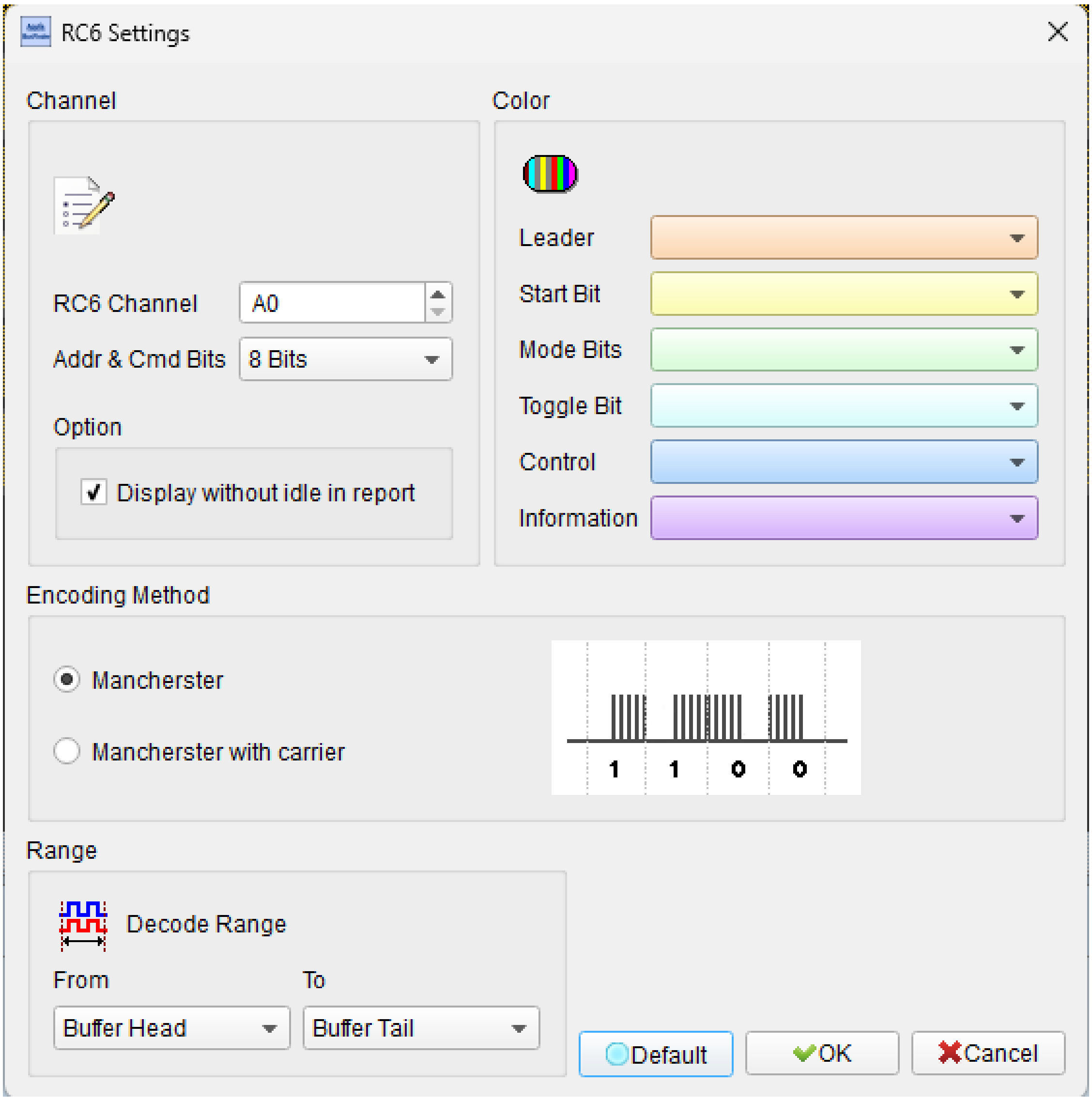


| | Timestamp | Status | S1 | S2 | Toggle | Address | Command |
|---|---|---|---|---|---|---|---|
| 1 | -258.066ms | Idle | | | | | |
| 2 | -898us | RC-5 | 1 | 1 | 0 | 00 (TV1) | 10 |
| 3 | 23.99ms | Idle | | | | | |
| 4 | 113.258ms | RC-5 | 1 | 1 | 0 | 00 (TV1) | 10 |
| 5 | 138.146ms | Idle | | | | | |

### RC-6

RC-6, like RC-5, is also developed by Philips. But, RC-6 has more features of remote controls than RC-5.

### Settings



RC6 Settings

Channel

RC6 Channel: A0

Addr & Cmd Bits: 8 Bits

Option

☑ Display without idle in report

Color

Leader
Start Bit
Mode Bits
Toggle Bit
Control
Information

Encoding Method

● Mancherster

○ Mancherster with carrier

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

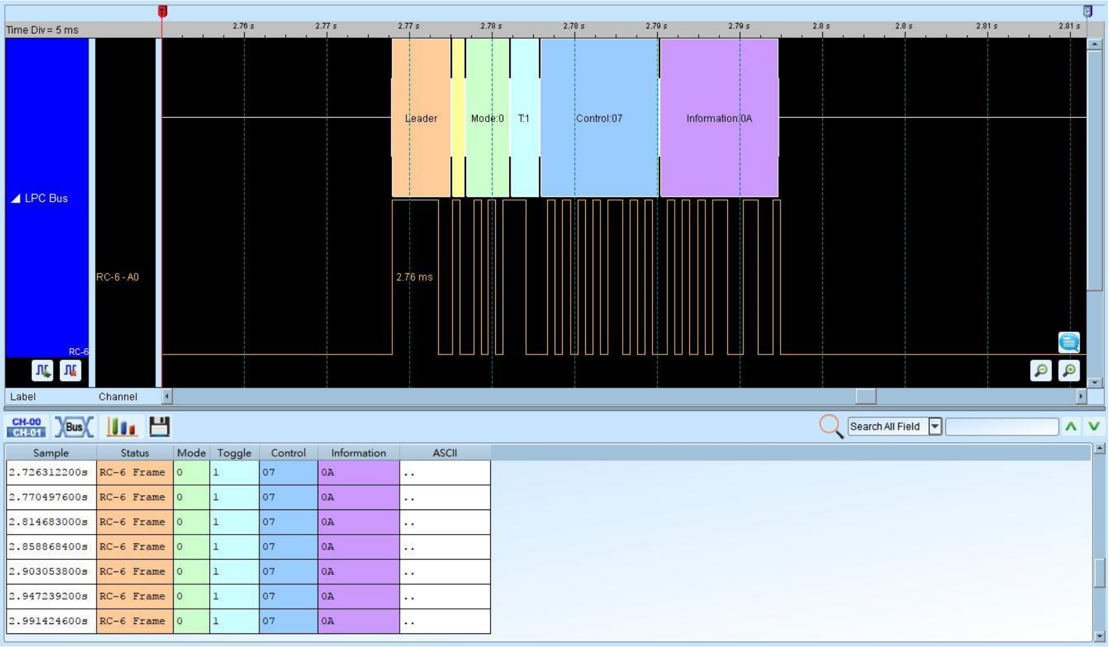
**Add & Cmd Bits:** Show commands in 8 bits or 16 bits of address and information in the control label.

**Display without idle in report:** Do not display any idle in the Report Window.

**Encoding Method:** Mancherster mode, Mancherster with carrier mode.

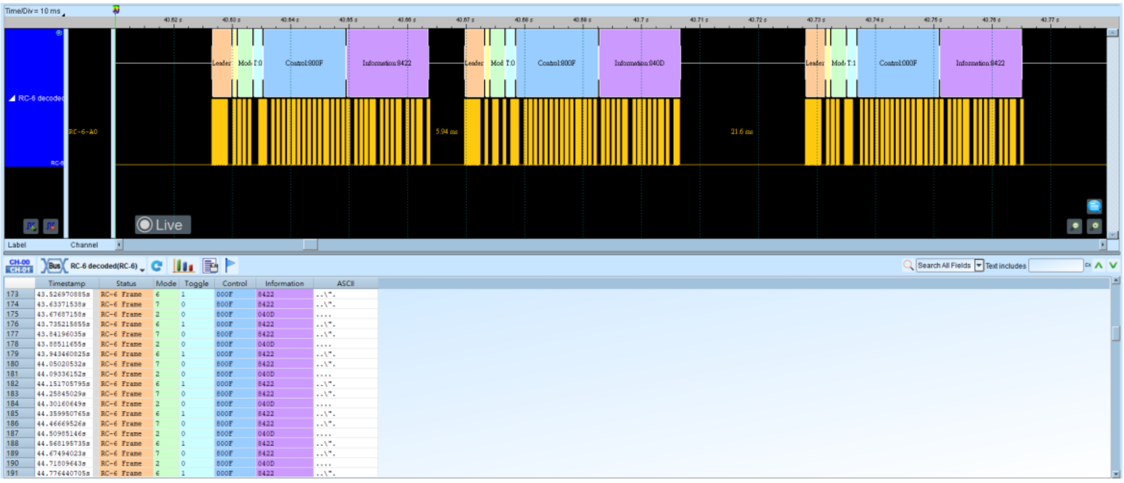
## Result

## RC6 without carrier



| Sample | Status | Mode | Toggle | Control | Information | ASCII |
|---|---|---|---|---|---|---|
| 2.726312200s | RC-6 Frame | 0 | 1 | 07 | 0A | .. |
| 2.770497600s | RC-6 Frame | 0 | 1 | 07 | 0A | .. |
| 2.814683000s | RC-6 Frame | 0 | 1 | 07 | 0A | .. |
| 2.858868400s | RC-6 Frame | 0 | 1 | 07 | 0A | .. |
| 2.903053800s | RC-6 Frame | 0 | 1 | 07 | 0A | .. |
| 2.947239200s | RC-6 Frame | 0 | 1 | 07 | 0A | .. |
| 2.991424600s | RC-6 Frame | 0 | 1 | 07 | 0A | .. |

## RC6 with carrier

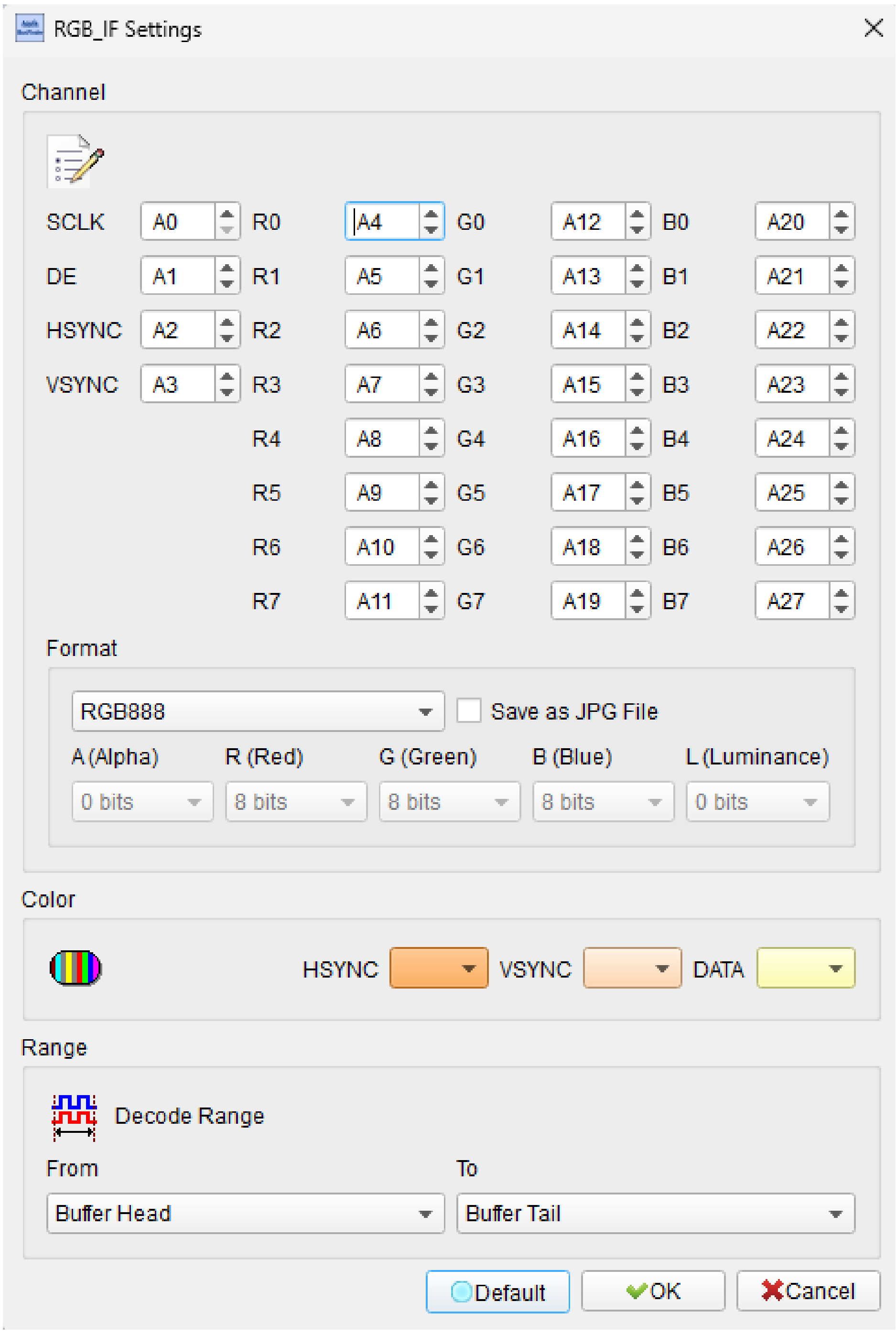


| | Timestamp | Status | Mode | Toggle | Control | Information | ASCII |
|---|---|---|---|---|---|---|---|
| 173 | 43.526970885s | RC-6 Frame | 6 | 1 | 000F | 8422 | ..\". |
| 174 | 43.63371538s | RC-6 Frame | 7 | 0 | 800F | 8422 | ..\". |
| 175 | 43.67687158s | RC-6 Frame | 2 | 0 | 800F | 040D | .... |
| 176 | 43.735215855s | RC-6 Frame | 6 | 1 | 000F | 8422 | ..\". |
| 177 | 43.84196035s | RC-6 Frame | 7 | 0 | 800F | 8422 | ..\". |
| 178 | 43.88511655s | RC-6 Frame | 2 | 0 | 800F | 040D | .... |
| 179 | 43.943460825s | RC-6 Frame | 6 | 1 | 000F | 8422 | ..\". |
| 180 | 44.05020532s | RC-6 Frame | 7 | 0 | 800F | 8422 | ..\". |
| 181 | 44.09336152s | RC-6 Frame | 2 | 0 | 800F | 040D | .... |
| 182 | 44.151705795s | RC-6 Frame | 6 | 1 | 000F | 8422 | ..\". |
| 183 | 44.25845029s | RC-6 Frame | 7 | 0 | 800F | 8422 | ..\". |
| 184 | 44.30160649s | RC-6 Frame | 2 | 0 | 800F | 040D | .... |
| 185 | 44.359950765s | RC-6 Frame | 6 | 1 | 000F | 8422 | ..\". |
| 186 | 44.46669526s | RC-6 Frame | 7 | 0 | 800F | 8422 | ..\". |
| 187 | 44.50985146s | RC-6 Frame | 2 | 0 | 800F | 040D | .... |
| 188 | 44.568195735s | RC-6 Frame | 6 | 1 | 000F | 8422 | ..\". |
| 189 | 44.67494023s | RC-6 Frame | 7 | 0 | 800F | 8422 | ..\". |
| 190 | 44.71809643s | RC-6 Frame | 2 | 0 | 800F | 040D | .... |
| 191 | 44.776440705s | RC-6 Frame | 6 | 1 | 000F | 8422 | ..\". |

## RGB Interface

RGB Interface is for data transmission between MCU and LCD. LCD Panel can be driven by LCD controller. RGB data would be written in memory and can be transmitted to LCD controller. It is able to show the picture of LCD Panel by reading the data from the interface.

## Settings



RGB_IF Settings

Channel

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| SCLK | A0 | R0 | A4 | G0 | A12 | B0 | A20 |
| DE | A1 | R1 | A5 | G1 | A13 | B1 | A21 |
| HSYNC | A2 | R2 | A6 | G2 | A14 | B2 | A22 |
| VSYNC | A3 | R3 | A7 | G3 | A15 | B3 | A23 |
| | | R4 | A8 | G4 | A16 | B4 | A24 |
| | | R5 | A9 | G5 | A17 | B5 | A25 |
| | | R6 | A10 | G6 | A18 | B6 | A26 |
| | | R7 | A11 | G7 | A19 | B7 | A27 |

Format

RGB888 □ Save as JPG File

| A (Alpha) | R (Red) | G (Green) | B (Blue) | L (Luminance) |
|---|---|---|---|---|
| 0 bits | 8 bits | 8 bits | 8 bits | 0 bits |

Color

HSYNC VSYNC DATA

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Channel**

**SCLK:** The Clock pin.

**DE:** The Data Enable pin.

**Hsync:** The Horizontal synchronization pin.

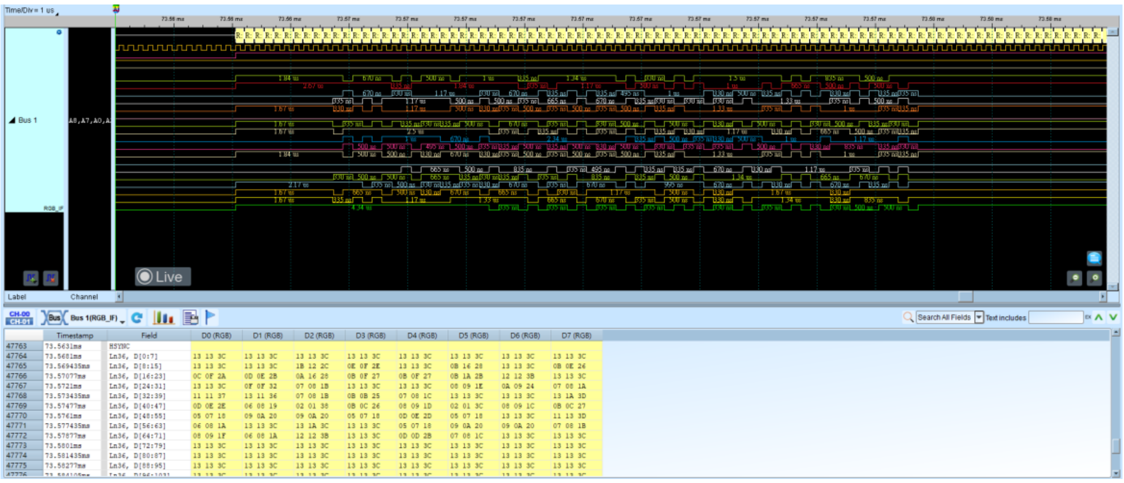
**Vsync:** The Vertical synchronization pin.

**R0 – 7, G0 – 7, B0 – 7:** RGB data pins.

**Format:** Select one of RGB formats or User defined.

**Save as JPG file:** Generate the JPG file with RGB data in the work directory of LA.

**Result**

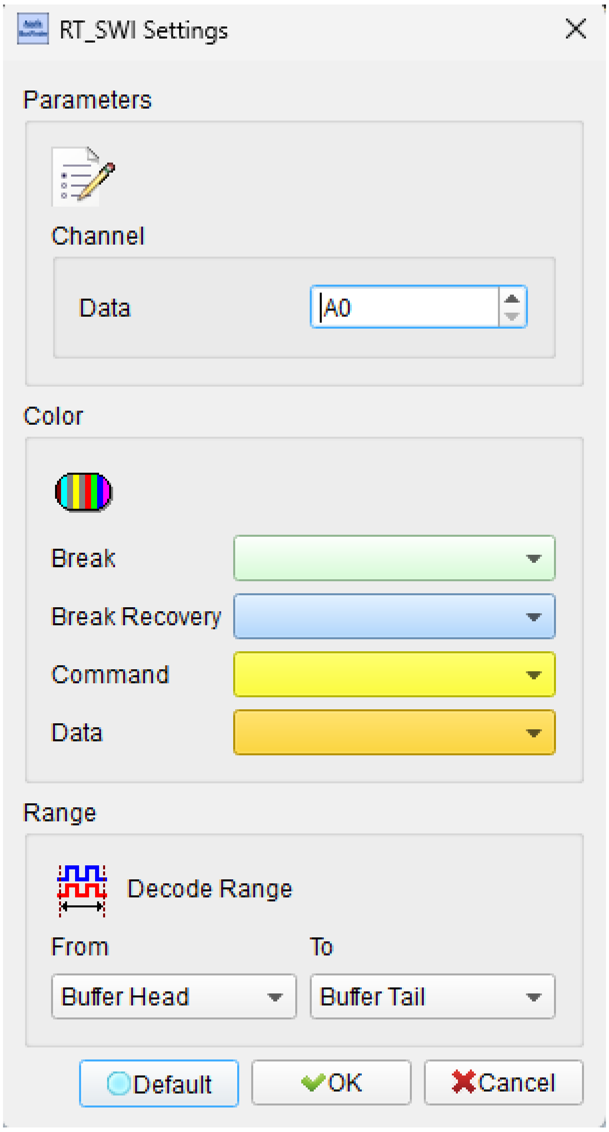


| | Timestamp | Field | D0 (RGB) | D1 (RGB) | D2 (RGB) | D3 (RGB) | D4 (RGB) | D5 (RGB) | D6 (RGB) | D7 (RGB) |
|---|---|---|---|---|---|---|---|---|---|---|
| 47763 | 73.5631ms | HSYNC | | | | | | | | |
| 47764 | 73.5681ms | Ln36, D[0:7] | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C |
| 47765 | 73.569435ms | Ln36, D[8:15] | 13 13 3C | 13 13 3C | 1B 12 2C | 0E 0F 2E | 13 13 3C | 0B 16 28 | 13 13 3C | 0B 0E 26 |
| 47766 | 73.57077ms | Ln36, D[16:23] | 0C 0F 2A | 0D 0E 2B | 0A 16 28 | 0B 0F 27 | 0B 0F 27 | 0B 1A 2B | 12 12 3B | 13 13 3C |
| 47767 | 73.5721ms | Ln36, D[24:31] | 13 13 3C | 0F 0F 32 | 07 08 1B | 13 13 3C | 13 13 3C | 08 09 1E | 0A 09 24 | 07 08 1A |
| 47768 | 73.573435ms | Ln36, D[32:39] | 11 11 37 | 13 11 36 | 07 08 1B | 0B 0B 25 | 07 08 1C | 13 13 3C | 13 13 3C | 13 1A 3D |
| 47769 | 73.57477ms | Ln36, D[40:47] | 0D 0E 2E | 06 08 19 | 02 01 38 | 0B 0C 26 | 08 09 1D | 02 01 3C | 08 09 1C | 0B 0C 27 |
| 47770 | 73.5761ms | Ln36, D[48:55] | 05 07 18 | 09 0A 20 | 09 0A 20 | 05 07 18 | 0D 0E 2D | 05 07 18 | 13 13 3C | 11 13 3D |
| 47771 | 73.577435ms | Ln36, D[56:63] | 06 08 1A | 13 13 3C | 13 1A 3C | 13 13 3C | 05 07 18 | 09 0A 20 | 09 0A 20 | 07 08 1B |
| 47772 | 73.57877ms | Ln36, D[64:71] | 08 09 1F | 06 08 1A | 12 12 3B | 13 13 3C | 0D 0D 2B | 07 08 1C | 13 13 3C | 13 13 3C |
| 47773 | 73.5801ms | Ln36, D[72:79] | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C |
| 47774 | 73.581435ms | Ln36, D[80:87] | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C |
| 47775 | 73.58277ms | Ln36, D[88:95] | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C |
| 47776 | 73.584105ms | Ln36, D[96:103] | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C | 13 13 3C |

## RT_SWI

Realtek Single Wire Interface (SWI) is a communication protocol provided by Realtek. It is an interface designed for data transfer over a single data wire. This interface can help simplify hardware design, reduce wiring complexity, and effectively save space and cost.

### Settings



RT_SWI Settings

Parameters

Channel

Data: A0

Color

Break: 🟩

Break Recovery: 🟦

Command: 🟨

Data: 🟧

Range

Decode Range

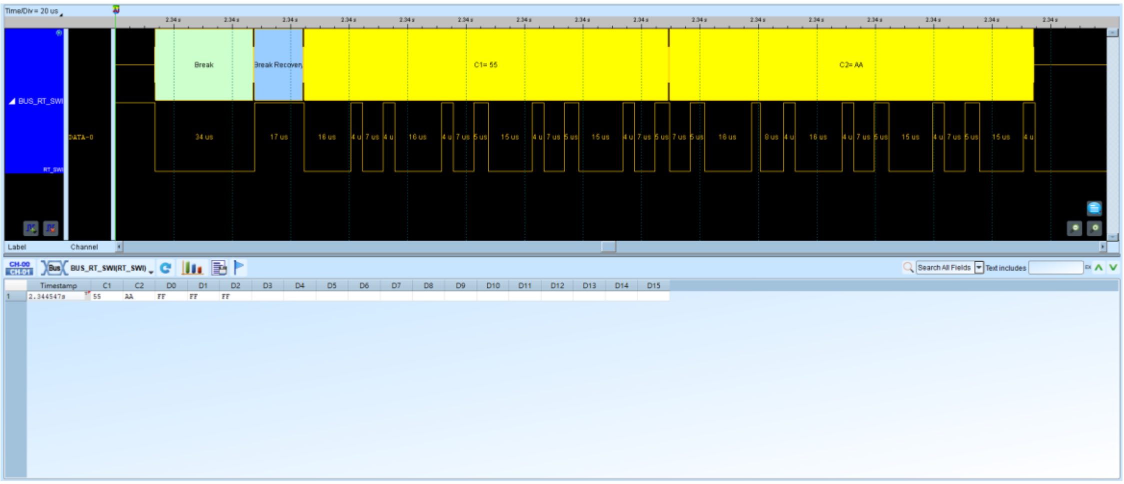
From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

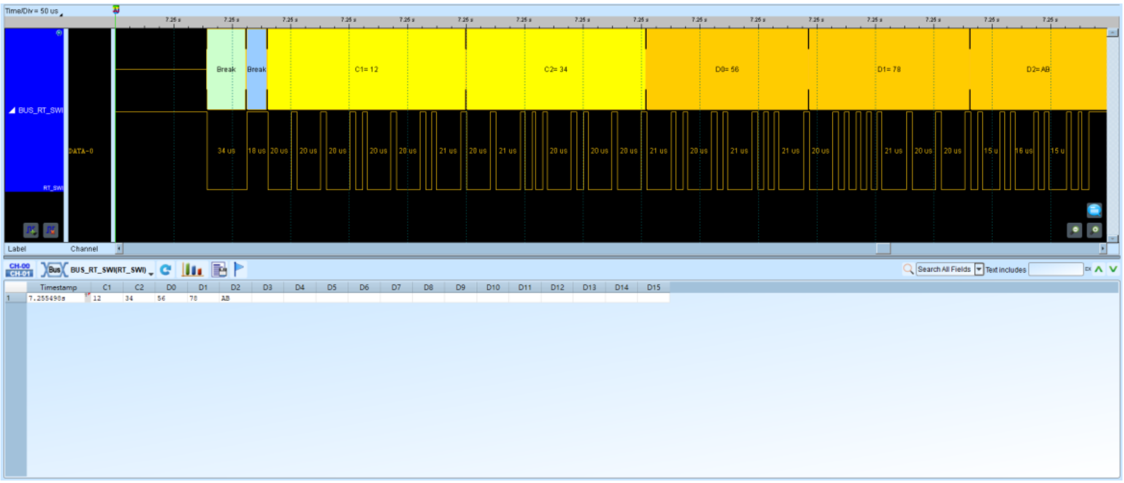
## Result

### Read



| | Timestamp | C1 | C2 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2.344547s | 55 | AA | FF | FF | FF | | | | | | | | | | | | | |

### Write

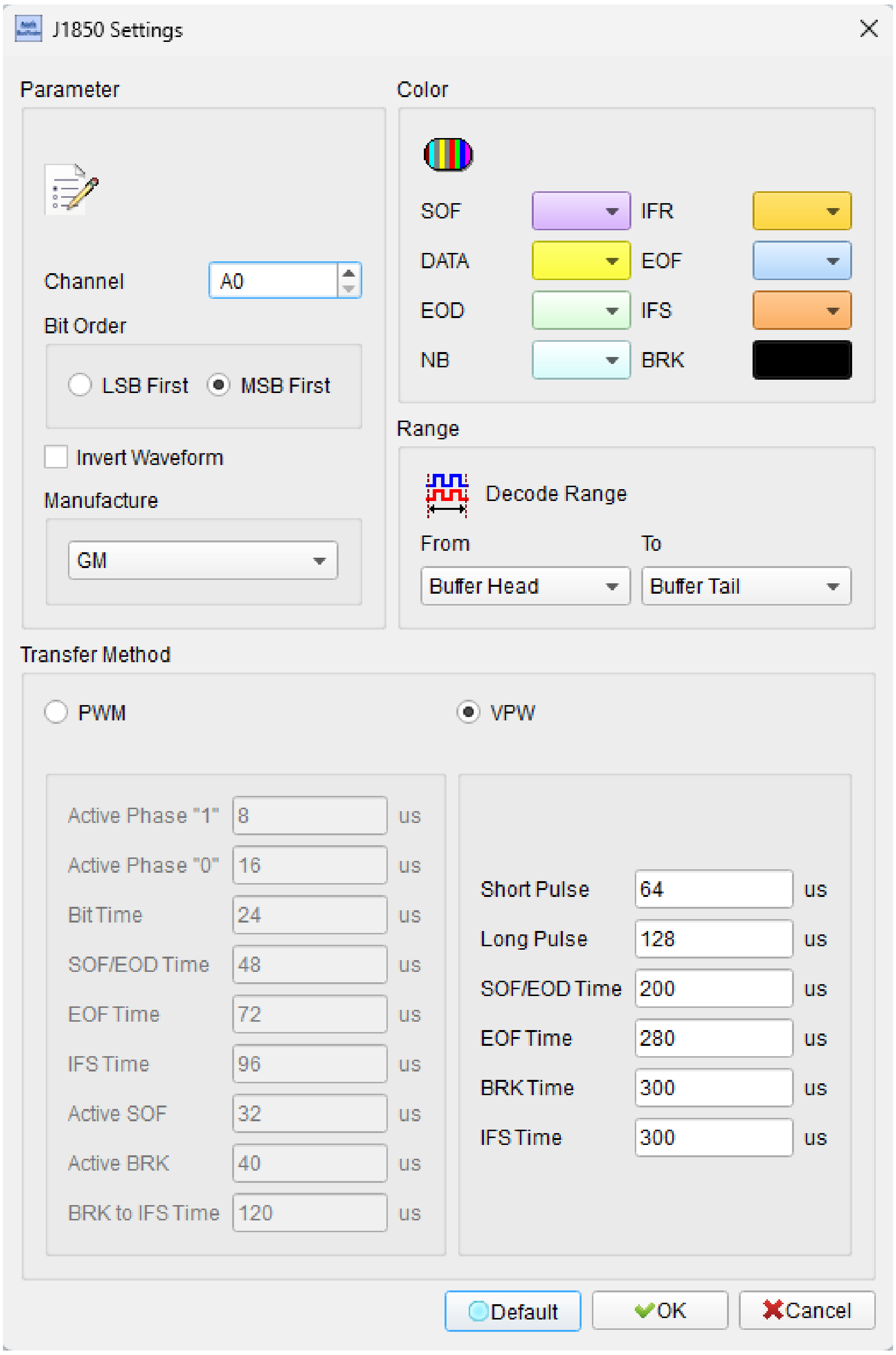


| | Timestamp | C1 | C2 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 7.255498s | 12 | 34 | 56 | 78 | AB | | | | | | | | | | | | | |

## SAE J1850

SAE J1850 is an automotive communication protocol used primarily for on-board diagnostics (OBD) and in-vehicle networking. It was widely used in vehicles before CAN (Controller Area Network) became the standard. J1850 allows different electronic control units (ECUs) to communicate within a vehicle.

## Settings



J1850 Settings

Parameter

Channel: A0

Bit Order: ○ LSB First ● MSB First

□ Invert Waveform

Manufacture: GM

Color

| | | | |
|---|---|---|---|
| SOF | | IFR | |
| DATA | | EOF | |
| EOD | | IFS | |
| NB | | BRK | |

Range

Decode Range

From: Buffer Head To: Buffer Tail

Transfer Method

○ PWM ● VPW

| | | |
|---|---|---|
| Active Phase "1" | 8 | us |
| Active Phase "0" | 16 | us |
| Bit Time | 24 | us |
| SOF/EOD Time | 48 | us |
| EOF Time | 72 | us |
| IFS Time | 96 | us |
| Active SOF | 32 | us |
| Active BRK | 40 | us |
| BRK to IFS Time | 120 | us |

| | | |
|---|---|---|
| Short Pulse | 64 | us |
| Long Pulse | 128 | us |
| SOF/EOD Time | 200 | us |
| EOF Time | 280 | us |
| BRK Time | 300 | us |
| IFS Time | 300 | us |

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Bit Order:** Select the transferred bit order. Default is MSB first.

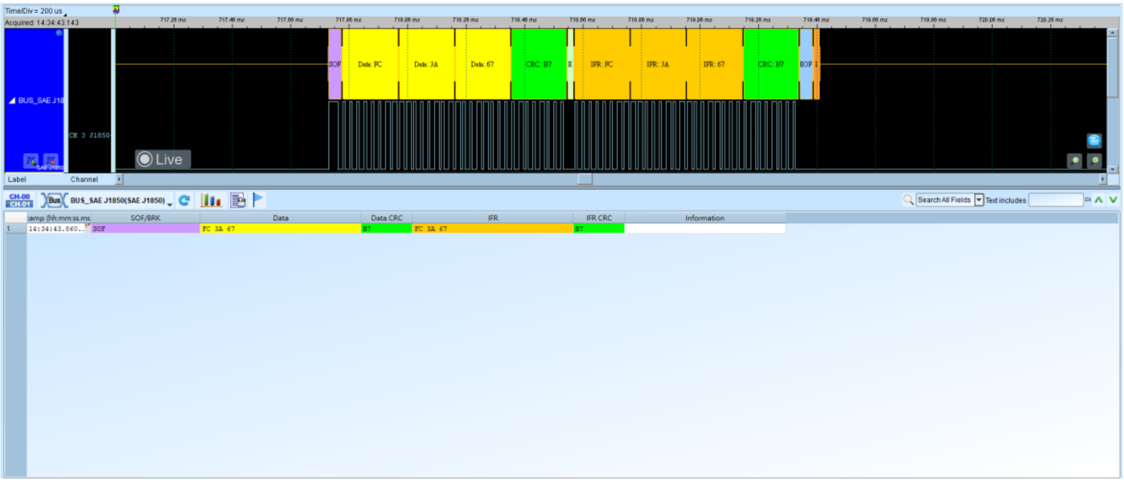
**Invert Waveform:** Invert the waveform for analyzing. Enable while checking.

**Manufacture:** Select the Manufacture. Due to different manufacture defined different NB value to indicate enable CRC or not. Please note that, this combo box can only be accessed when the transferred method had been adjusted to VPW. We only support GM and Chrysler now, if user need more manufacture candidate, please contact us.

**Transfer Method:** Select the transferred method. Default is VPW. PWM and VPW has different timing parameters. User can adjust the value manually.

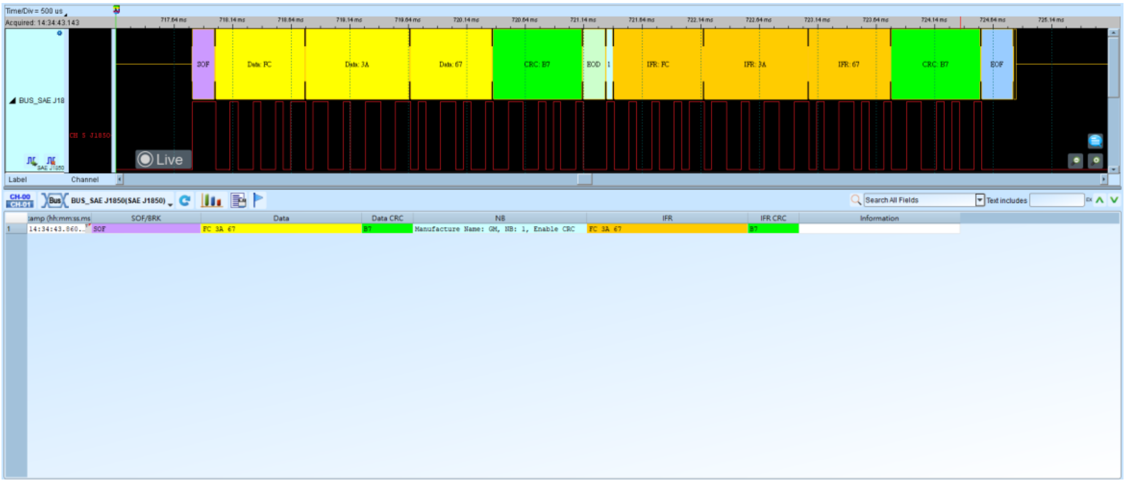
**Result:**

**PWM:**



| | tamp (hh:mm:ss.ms | SOF/BRK | Data | Data CRC | IFR | IFR CRC | Information |
|---|---|---|---|---|---|---|---|
| 1 | 14:34:43.860... | SOF | FC 3A 67 | B7 | FC 3A 67 | B7 | |

**VPW:**

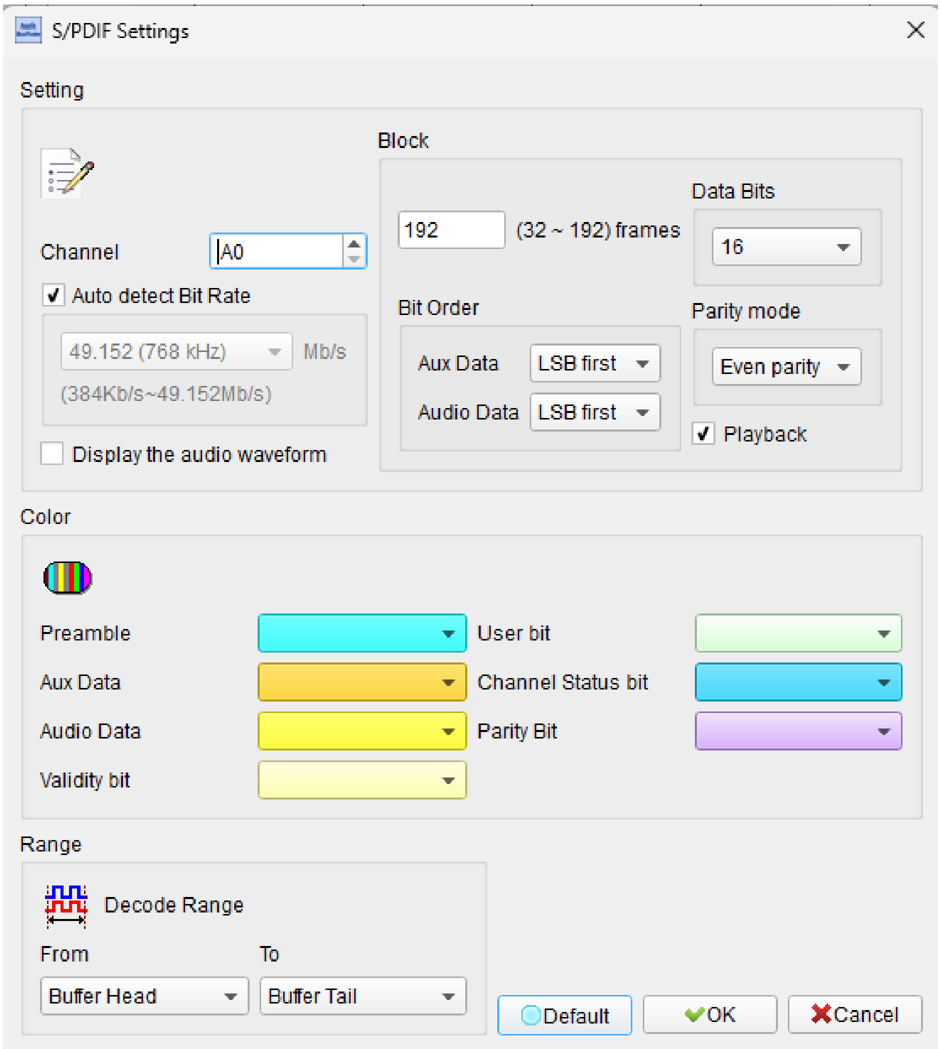


|  | tamp (hh:mm:ss.ms | SOF/BRK | Data | Data CRC | NB | IFR | IFR CRC | Information |
|---|---|---|---|---|---|---|---|---|
| 1 | 14:34:43.860... | SOF | FC 3A 67 | B7 | Manufacture Name: GM, NB: 1, Enable CRC | FC 3A 67 | B7 |  |

## S/PDIF

It is a digital audio interface that can be transmitted using either wire or fiber optics. It is called the Sony/Philips Digital Interconnect Format (also known as Sony Philips Digital InterFace). These two companies are the primary specification developers, and the specification is derived from the AES/EBU professional digital audio interface, with some modifications for use in lower-cost hardware.

## Settings



S/PDIF Settings

Setting

Channel: A0

☑ Auto detect Bit Rate

49.152 (768 kHz) Mb/s

(384Kb/s~49.152Mb/s)

□ Display the audio waveform

Block: 192 (32 ~ 192) frames

Data Bits: 16

Bit Order: Aux Data LSB first; Audio Data LSB first

Parity mode: Even parity

☑ Playback

Color

Preamble, Aux Data, Audio Data, Validity bit, User bit, Channel Status bit, Parity Bit

Range

Decode Range

From: Buffer Head  To: Buffer Tail

**Channel:** The default is Channel 0.

**Auto detect Bit Rate:** Turned on by default.

**Num of frame:** 192 frames within each block by default, used to analyze each sub-frame order User bit and Channel status bit.

**Bit Order (Aux. Data):** The default is the LSB first for the Aux. data.

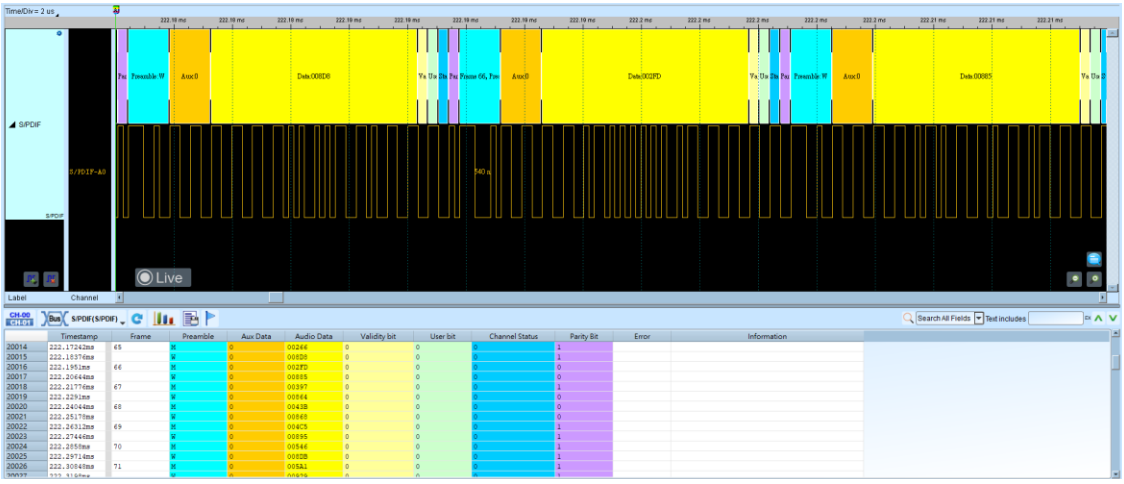
**Bit Order (Audio Data):** The default is the LSB first for Audio data.

**Data format:** The default is 16 bits.

**Parity mode:** The default is even parity.

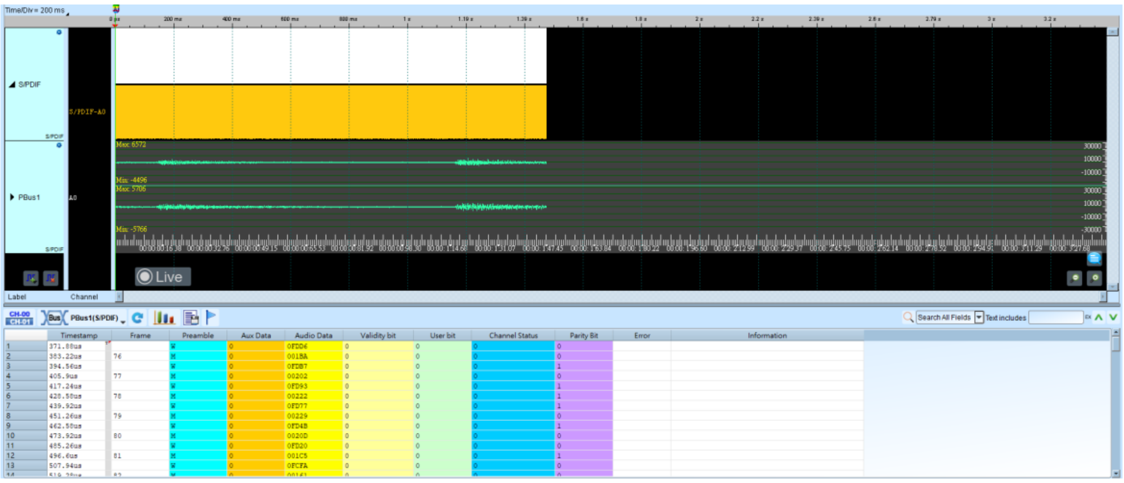
**Display the audio waveform:** Click to display the audio waveform in the Waveform Window.

**Result**



| | Timestamp | Frame | Preamble | Aux Data | Audio Data | Validity bit | User bit | Channel Status | Parity Bit | Error | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 20014 | 222.17242ms | 65 | M | 0 | 00266 | 0 | 0 | 0 | 1 | | |
| 20015 | 222.18376ms | | W | 0 | 008D8 | 0 | 0 | 0 | 1 | | |
| 20016 | 222.1951ms | 66 | M | 0 | 002FD | 0 | 0 | 0 | 0 | | |
| 20017 | 222.20644ms | | W | 0 | 00885 | 0 | 0 | 0 | 0 | | |
| 20018 | 222.21776ms | 67 | M | 0 | 00397 | 0 | 0 | 0 | 1 | | |
| 20019 | 222.2291ms | | W | 0 | 00864 | 0 | 0 | 0 | 0 | | |
| 20020 | 222.24044ms | 68 | M | 0 | 0043B | 0 | 0 | 0 | 0 | | |
| 20021 | 222.25178ms | | W | 0 | 00868 | 0 | 0 | 0 | 0 | | |
| 20022 | 222.26312ms | 69 | M | 0 | 004C5 | 0 | 0 | 0 | 1 | | |
| 20023 | 222.27446ms | | W | 0 | 00895 | 0 | 0 | 0 | 1 | | |
| 20024 | 222.2858ms | 70 | M | 0 | 00546 | 0 | 0 | 0 | 1 | | |
| 20025 | 222.29714ms | | W | 0 | 008DB | 0 | 0 | 0 | 1 | | |
| 20026 | 222.30848ms | 71 | M | 0 | 005A1 | 0 | 0 | 0 | 1 | | |
| 20027 | 222.3198ms | | W | 0 | 00929 | 0 | 0 | 0 | 1 | | |

Show wave:

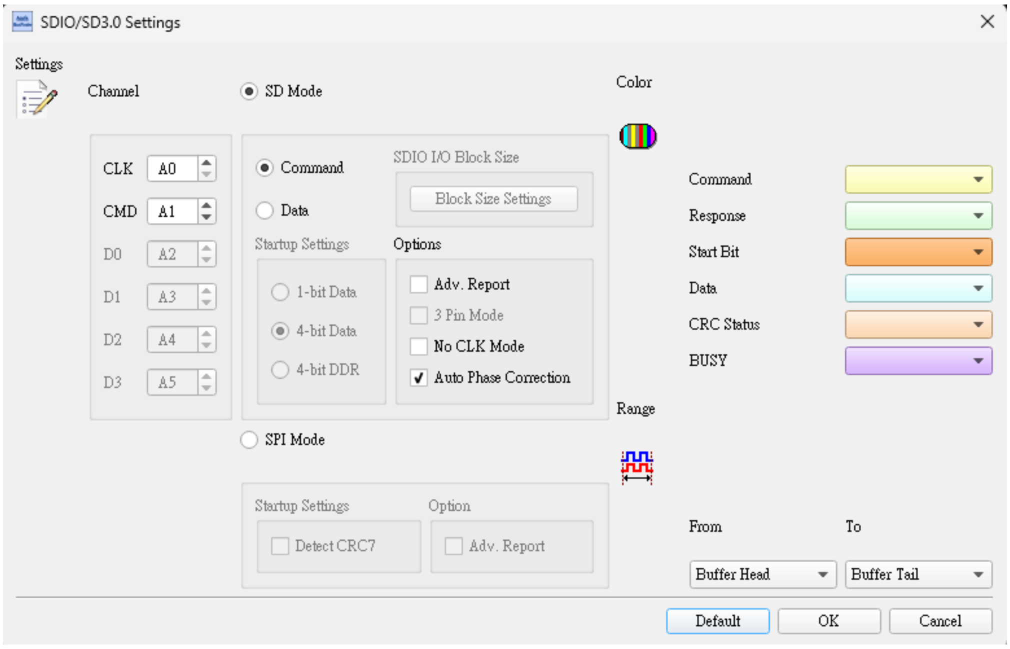


| | Timestamp | Frame | Preamble | Aux Data | Audio Data | Validity bit | User bit | Channel Status | Parity Bit | Error | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 371.88us | | W | 0 | 0FDD6 | 0 | 0 | 0 | 0 | | |
| 2 | 383.22us | 76 | M | 0 | 001BA | 0 | 0 | 0 | 0 | | |
| 3 | 394.56us | | W | 0 | 0FDB7 | 0 | 0 | 0 | 1 | | |
| 4 | 405.9us | 77 | M | 0 | 00202 | 0 | 0 | 0 | 0 | | |
| 5 | 417.24us | | W | 0 | 0FD93 | 0 | 0 | 0 | 1 | | |
| 6 | 428.58us | 78 | M | 0 | 00222 | 0 | 0 | 0 | 1 | | |
| 7 | 439.92us | | W | 0 | 0FD77 | 0 | 0 | 0 | 1 | | |
| 8 | 451.26us | 79 | M | 0 | 00229 | 0 | 0 | 0 | 0 | | |
| 9 | 462.58us | | W | 0 | 0FD4B | 0 | 0 | 0 | 1 | | |
| 10 | 473.92us | 80 | M | 0 | 0020D | 0 | 0 | 0 | 0 | | |
| 11 | 485.26us | | W | 0 | 0FD20 | 0 | 0 | 0 | 0 | | |
| 12 | 496.6us | 81 | M | 0 | 001C5 | 0 | 0 | 0 | 1 | | |
| 13 | 507.94us | | W | 0 | 0FCFA | 0 | 0 | 0 | 0 | | |
| 14 | 519.28us | 82 | M | 0 | 00161 | 0 | 0 | 0 | 0 | | |

## SDIO

The SD3.0/SDIO3.0 Protocol is a high speed serial protocol used primarily for interfacing with SD (Secure Digital) Flash memory cards.

### Settings



SDIO/SD3.0 Settings

Settings

Channel

| | |
|---|---|
| CLK | A0 |
| CMD | A1 |
| D0 | A2 |
| D1 | A3 |
| D2 | A4 |
| D3 | A5 |

● SD Mode

● Command
○ Data

SDIO I/O Block Size: Block Size Settings

Startup Settings
○ 1-bit Data
● 4-bit Data
○ 4-bit DDR

Options
□ Adv. Report
□ 3 Pin Mode
□ No CLK Mode
☑ Auto Phase Correction

○ SPI Mode

Startup Settings: □ Detect CRC7

Option: □ Adv. Report

Color

| | |
|---|---|
| Command | 🟨 |
| Response | 🟩 |
| Start Bit | 🟧 |
| Data | 🟦 |
| CRC Status | 🟧 |
| BUSY | 🟪 |

Range

From: Buffer Head
To: Buffer Tail

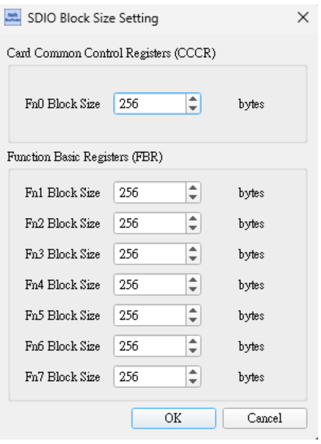
**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**SD Mode, analyzed in SPI mode:**

**Command:** Analyze Command.

**Data:** Analyze Data.

**SDIO I/O Block Size:** Set the block size of CCCR and FBR of SDIO.



SDIO Block Size Setting

Card Common Control Registers (CCCR)

Fn0 Block Size 256 bytes

Function Basic Registers (FBR)

Fn1 Block Size 256 bytes
Fn2 Block Size 256 bytes
Fn3 Block Size 256 bytes
Fn4 Block Size 256 bytes
Fn5 Block Size 256 bytes
Fn6 Block Size 256 bytes
Fn7 Block Size 256 bytes

**Options:**

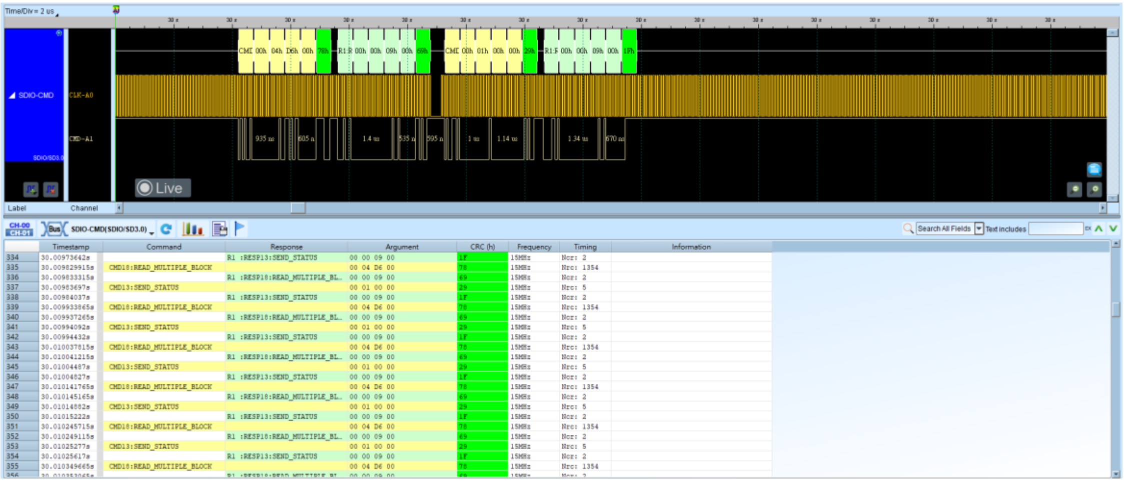
1. **Adv. Report:** Show details. Enabled when checked.
2. **3 Pin Mode:** Analyze the data using CLK, CMD, and D0.
3. **No CLK Mode:** Analyze the data using CMD line only.
4. **Auto Phase Correction:** Automatically adjusts the phase of measurement.

**SPI Mode, analyzed in SPI mode:**

1. **Detect CRC7:** Detect CRC 7 or not. Enabled when checked.
2. **Adv. Report:** Show details. Enabled when checked.

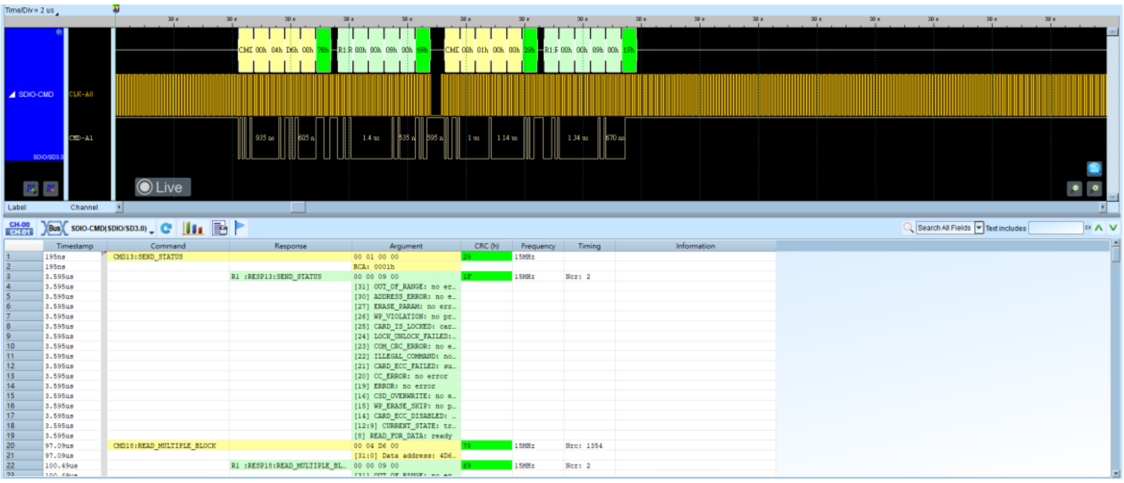
**Result**

CMD mode



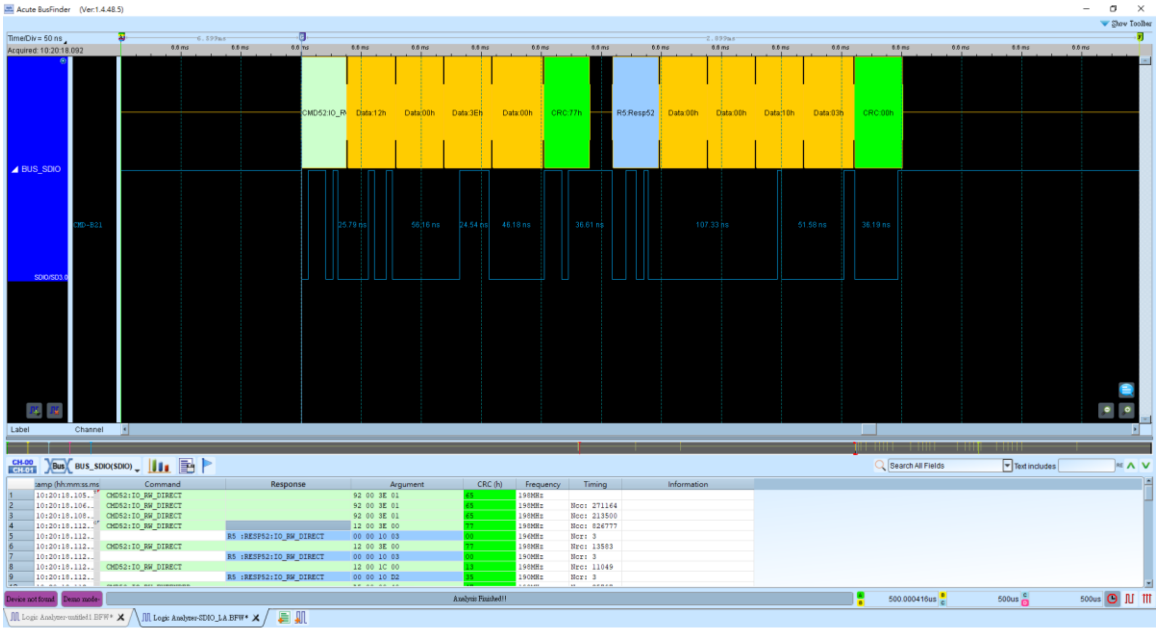
| | Timestamp | Command | Response | Argument | CRC (h) | Frequency | Timing | Information |
|---|---|---|---|---|---|---|---|---|
| 334 | 30.00973642s | | R1 :RESP13:SEND_STATUS | 00 00 09 00 | 1F | 15MHz | Ncr: 2 | |
| 335 | 30.009829915s | CMD18:READ_MULTIPLE_BLOCK | | 00 04 D6 00 | 78 | 15MHz | Nrc: 1354 | |
| 336 | 30.009833315s | | R1 :RESP18:READ_MULTIPLE_BL.. | 00 00 09 00 | 69 | 15MHz | Ncr: 2 | |
| 337 | 30.00983697s | CMD13:SEND_STATUS | | 00 01 00 00 | 29 | 15MHz | Nrc: 5 | |
| 338 | 30.00984037s | | R1 :RESP13:SEND_STATUS | 00 00 09 00 | 1F | 15MHz | Ncr: 2 | |
| 339 | 30.009933865s | CMD18:READ_MULTIPLE_BLOCK | | 00 04 D6 00 | 78 | 15MHz | Nrc: 1354 | |
| 340 | 30.009937265s | | R1 :RESP18:READ_MULTIPLE_BL.. | 00 00 09 00 | 69 | 15MHz | Ncr: 2 | |
| 341 | 30.00994092s | CMD13:SEND_STATUS | | 00 01 00 00 | 29 | 15MHz | Nrc: 5 | |
| 342 | 30.00994432s | | R1 :RESP13:SEND_STATUS | 00 00 09 00 | 1F | 15MHz | Ncr: 2 | |
| 343 | 30.010037815s | CMD18:READ_MULTIPLE_BLOCK | | 00 04 D6 00 | 78 | 15MHz | Nrc: 1354 | |
| 344 | 30.010041215s | | R1 :RESP18:READ_MULTIPLE_BL.. | 00 00 09 00 | 69 | 15MHz | Ncr: 2 | |
| 345 | 30.01004487s | CMD13:SEND_STATUS | | 00 01 00 00 | 29 | 15MHz | Nrc: 5 | |
| 346 | 30.01004827s | | R1 :RESP13:SEND_STATUS | 00 00 09 00 | 1F | 15MHz | Ncr: 2 | |
| 347 | 30.010141765s | CMD18:READ_MULTIPLE_BLOCK | | 00 04 D6 00 | 78 | 15MHz | Nrc: 1354 | |
| 348 | 30.010145165s | | R1 :RESP18:READ_MULTIPLE_BL.. | 00 00 09 00 | 69 | 15MHz | Ncr: 2 | |
| 349 | 30.01014882s | CMD13:SEND_STATUS | | 00 01 00 00 | 29 | 15MHz | Nrc: 5 | |
| 350 | 30.01015222s | | R1 :RESP13:SEND_STATUS | 00 00 09 00 | 1F | 15MHz | Ncr: 2 | |
| 351 | 30.010245715s | CMD18:READ_MULTIPLE_BLOCK | | 00 04 D6 00 | 78 | 15MHz | Nrc: 1354 | |
| 352 | 30.010249115s | | R1 :RESP18:READ_MULTIPLE_BL.. | 00 00 09 00 | 69 | 15MHz | Ncr: 2 | |
| 353 | 30.01025277s | CMD13:SEND_STATUS | | 00 01 00 00 | 29 | 15MHz | Nrc: 5 | |
| 354 | 30.01025617s | | R1 :RESP13:SEND_STATUS | 00 00 09 00 | 1F | 15MHz | Ncr: 2 | |
| 355 | 30.010349665s | CMD18:READ_MULTIPLE_BLOCK | | 00 04 D6 00 | 78 | 15MHz | Nrc: 1354 | |

Adv. Report



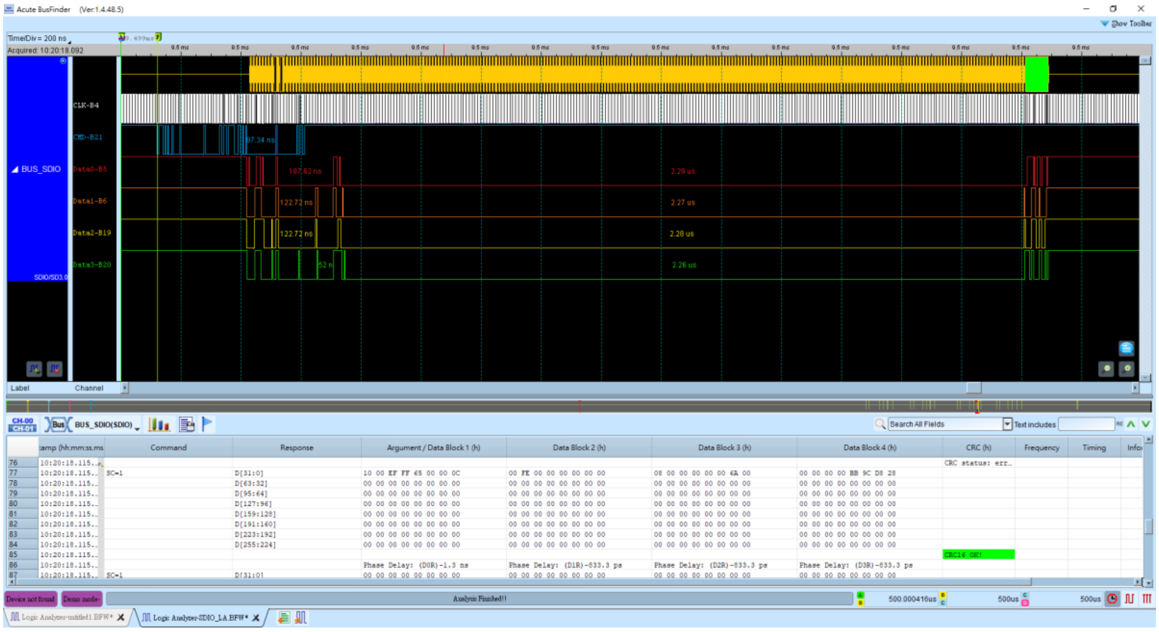
| | Timestamp | Command | Response | Argument | CRC (h) | Frequency | Timing | Information |
|---|---|---|---|---|---|---|---|---|
| 1 | 195ns | CMD13:SEND_STATUS | | 00 01 00 00 | 29 | 15MHz | | |
| 2 | 195ns | | | RCA: 0001h | | | | |
| 3 | 3.595us | | R1 :RESP13:SEND_STATUS | 00 00 09 00 | 1F | 15MHz | Ncr: 2 | |
| 4 | 3.595us | | | [31] OUT_OF_RANGE: no er… | | | | |
| 5 | 3.595us | | | [30] ADDRESS_ERROR: no e… | | | | |
| 6 | 3.595us | | | [27] ERASE_PARAM: no err… | | | | |
| 7 | 3.595us | | | [26] WP_VIOLATION: no pr… | | | | |
| 8 | 3.595us | | | [25] CARD_IS_LOCKED: car… | | | | |
| 9 | 3.595us | | | [24] LOCK_UNLOCK_FAILED:… | | | | |
| 10 | 3.595us | | | [23] COM_CRC_ERROR: no e… | | | | |
| 11 | 3.595us | | | [22] ILLEGAL_COMMAND: no… | | | | |
| 12 | 3.595us | | | [21] CARD_ECC_FAILED: su… | | | | |
| 13 | 3.595us | | | [20] CC_ERROR: no error | | | | |
| 14 | 3.595us | | | [19] ERROR: no error | | | | |
| 15 | 3.595us | | | [16] CSD_OVERWRITE: no e… | | | | |
| 16 | 3.595us | | | [15] WP_ERASE_SKIP: no p… | | | | |
| 17 | 3.595us | | | [14] CARD_ECC_DISABLED: … | | | | |
| 18 | 3.595us | | | [12:9] CURRENT_STATE: tr… | | | | |
| 19 | 3.595us | | | [8] READ_FOR_DATA: ready | | | | |
| 20 | 97.09us | CMD18:READ_MULTIPLE_BLOCK | | 00 04 D6 00 | 78 | 15MHz | Nrc: 1354 | |
| 21 | 97.09us | | | [31:0] Data address: 4D6… | | | | |
| 22 | 100.49us | | R1 :RESP18:READ_MULTIPLE_BL… | 00 00 09 00 | 69 | 15MHz | Ncr: 2 | |

No CLK mode:



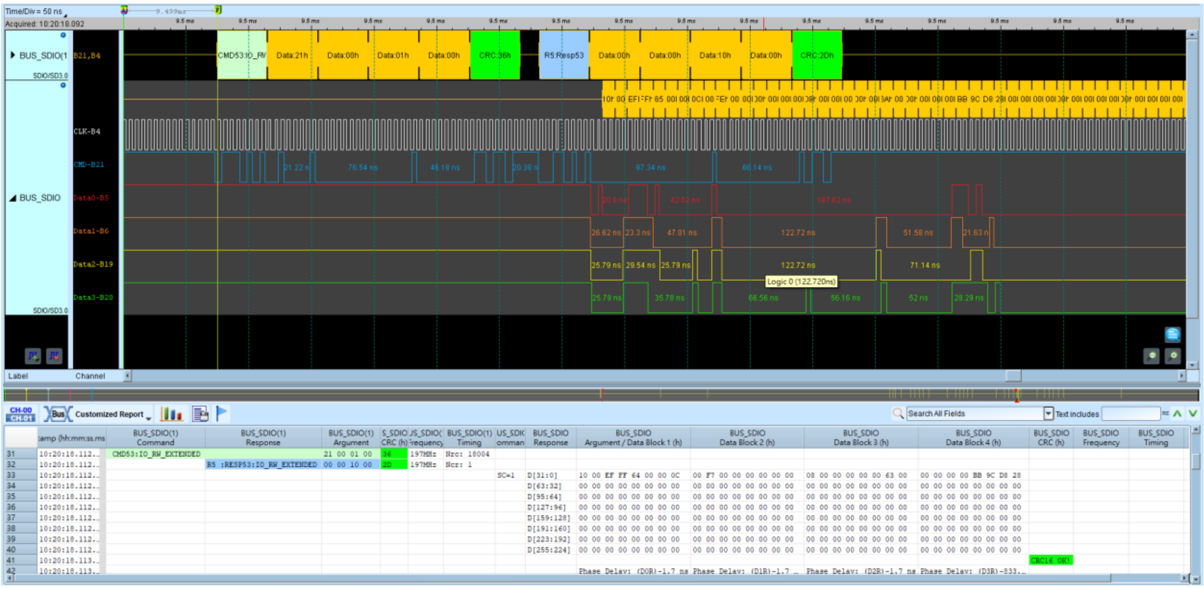
| | amp (hh:mm:ss.ms | Command | Response | Argument | CRC (h) | Frequency | Timing | Information |
|---|---|---|---|---|---|---|---|---|
| 1 | 10:20:18.105… | CMD52:IO_RW_DIRECT | | 92 00 3E 01 | 65 | 198MHz | | |
| 2 | 10:20:18.106… | CMD52:IO_RW_DIRECT | | 92 00 3E 01 | 65 | 198MHz | Ncc: 271164 | |
| 3 | 10:20:18.108… | CMD52:IO_RW_DIRECT | | 92 00 3E 01 | 65 | 198MHz | Ncc: 213500 | |
| 4 | 10:20:18.112… | CMD52:IO_RW_DIRECT | | 12 00 3E 00 | 77 | 198MHz | Ncc: 826777 | |
| 5 | 10:20:18.112… | | R5 :RESP52:IO_RW_DIRECT | 00 00 10 03 | 00 | 196MHz | Ncr: 3 | |
| 6 | 10:20:18.112… | CMD52:IO_RW_DIRECT | | 12 00 3E 00 | 77 | 198MHz | Nrc: 13583 | |
| 7 | 10:20:18.112… | | R5 :RESP52:IO_RW_DIRECT | 00 00 10 03 | 00 | 190MHz | Ncr: 3 | |
| 8 | 10:20:18.112… | CMD52:IO_RW_DIRECT | | 12 00 1C 00 | 13 | 198MHz | Nrc: 11049 | |
| 9 | 10:20:18.112… | | R5 :RESP52:IO_RW_DIRECT | 00 00 10 D2 | 35 | 190MHz | Ncr: 3 | |

## Data:



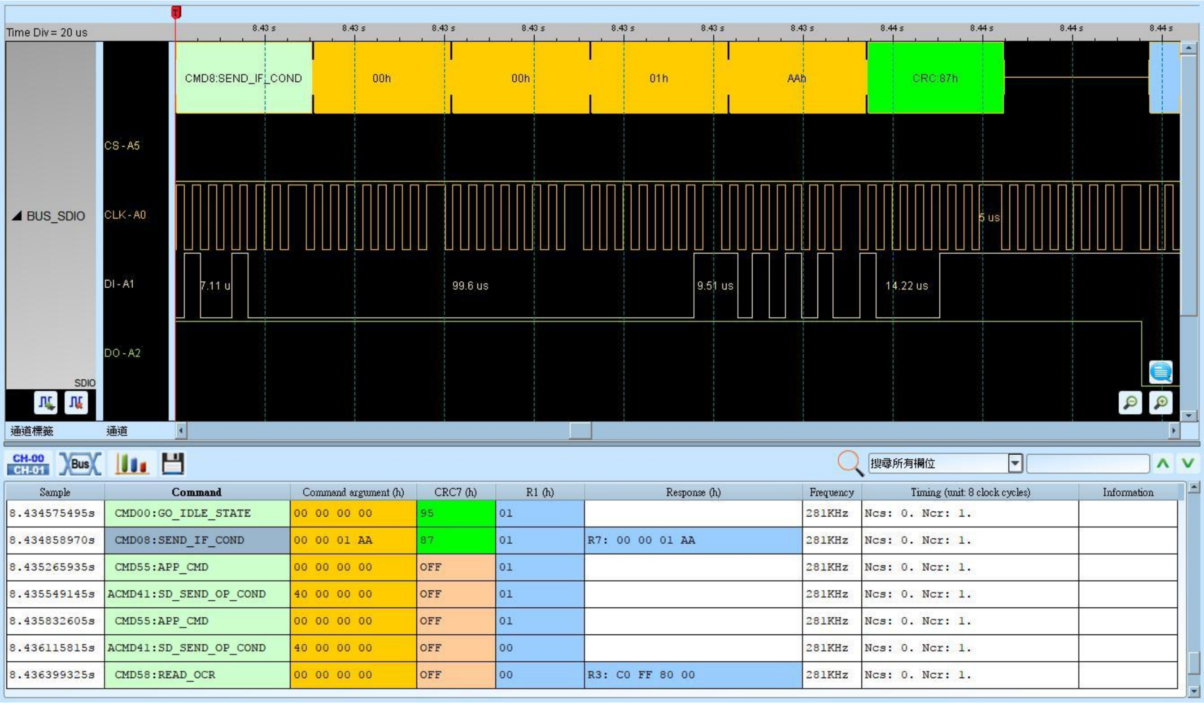
|  | tamp (hh:mm:ss.ms | Command | Response | Argument / Data Block 1 (h) | Data Block 2 (h) | Data Block 3 (h) | Data Block 4 (h) | CRC (h) | Frequency | Timing | Infor |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 76 | 10:20:18.115.. |  |  |  |  |  |  | CRC status: err.. |  |  |  |
| 77 | 10:20:18.115.. | SC=1 | D[31:0] | 10 00 EF FF 65 00 00 0C | 00 FE 00 00 00 00 00 00 | 08 00 00 00 00 00 6A 00 | 00 00 00 00 BB 9C D8 28 |  |  |  |  |
| 78 | 10:20:18.115.. |  | D[63:32] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |  |  |  |  |
| 79 | 10:20:18.115.. |  | D[95:64] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |  |  |  |  |
| 80 | 10:20:18.115.. |  | D[127:96] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |  |  |  |  |
| 81 | 10:20:18.115.. |  | D[159:128] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |  |  |  |  |
| 82 | 10:20:18.115.. |  | D[191:160] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |  |  |  |  |
| 83 | 10:20:18.115.. |  | D[223:192] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |  |  |  |  |
| 84 | 10:20:18.115.. |  | D[255:224] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |  |  |  |  |
| 85 | 10:20:18.115.. |  |  |  |  |  |  | CRC16 OK! |  |  |  |
| 86 | 10:20:18.115.. |  |  | Phase Delay: (D0R)-1.3 ns | Phase Delay: (D1R)-833.3 ps | Phase Delay: (D2R)-833.3 ps | Phase Delay: (D3R)-833.3 ps |  |  |  |  |
| 87 | 10:20:18.115.. | SC=1 | D[31:0] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 |  |  |  |  |

## Command + Data mode:



| | tamp (hh:mm:ss.ms | BUS_SDIO(1) Command | BUS_SDIO(1) Response | BUS_SDIO(1) Argument | S_SDIO CRC (h) | JS_SDIO( Frequency | BUS_SDIO(1) Timing | US_SDIO omman | BUS_SDIO Response | BUS_SDIO Argument / Data Block 1 (h) | BUS_SDIO Data Block 2 (h) | BUS_SDIO Data Block 3 (h) | BUS_SDIO Data Block 4 (h) | BUS_SDIO CRC (h) | BUS_SDIO Frequency | BUS_SDIO Timing |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 31 | 10:20:18.112... | CMD53:IO_RW_EXTENDED | | 21 00 01 00 | 36 | 197MHz | Nrc: 18004 | | | | | | | | | |
| 32 | 10:20:18.112... | | R5 :RESP53:IO_RW_EXTENDED | 00 00 10 00 | 2D | 197MHz | Ncr: 1 | | | | | | | | | |
| 33 | 10:20:18.112... | | | | | | | SC=1 | D[31:0] | 10 00 EF FF 64 00 00 0C | 00 F7 00 00 00 00 00 00 | 08 00 00 00 00 00 63 00 | 00 00 00 00 BB 9C D8 28 | | | |
| 34 | 10:20:18.112... | | | | | | | | D[63:32] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | | | |
| 35 | 10:20:18.112... | | | | | | | | D[95:64] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | | | |
| 36 | 10:20:18.112... | | | | | | | | D[127:96] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | | | |
| 37 | 10:20:18.112... | | | | | | | | D[159:128] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | | | |
| 38 | 10:20:18.112... | | | | | | | | D[191:160] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | | | |
| 39 | 10:20:18.112... | | | | | | | | D[223:192] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | | | |
| 40 | 10:20:18.112... | | | | | | | | D[255:224] | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00 | | | |
| 41 | 10:20:18.113... | | | | | | | | | | | | | CRC16 OK! | | |
| 42 | 10:20:18.113... | | | | | | | | | Phase Delay: (D0R)-1.7 ns | Phase Delay: (D1R)-1.7 ... | Phase Delay: (D2R)-1.7 ns | Phase Delay: (D3R)-833... | | | |

## SPI mode:

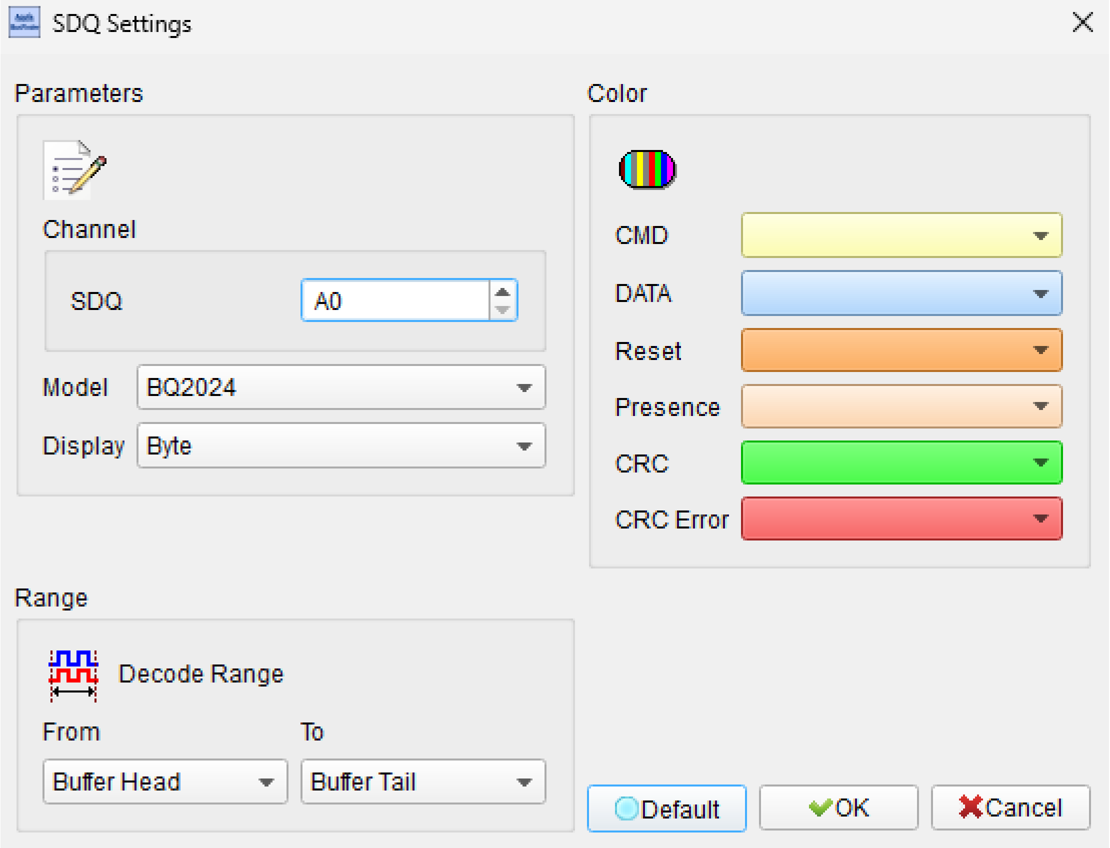


| Sample | Command | Command argument (h) | CRC7 (h) | R1 (h) | Response (h) | Frequency | Timing (unit: 8 clock cycles) | Information |
|---|---|---|---|---|---|---|---|---|
| 8.4345754955s | CMD00:GO_IDLE_STATE | 00 00 00 00 | 95 | 01 | | 281KHz | Ncs: 0. Ncr: 1. | |
| 8.4348589705s | CMD08:SEND_IF_COND | 00 00 01 AA | 87 | 01 | R7: 00 00 01 AA | 281KHz | Ncs: 0. Ncr: 1. | |
| 8.4352659355s | CMD55:APP_CMD | 00 00 00 00 | OFF | 01 | | 281KHz | Ncs: 0. Ncr: 1. | |
| 8.4355491455s | ACMD41:SD_SEND_OP_COND | 40 00 00 00 | OFF | 01 | | 281KHz | Ncs: 0. Ncr: 1. | |
| 8.4358326055s | CMD55:APP_CMD | 00 00 00 00 | OFF | 01 | | 281KHz | Ncs: 0. Ncr: 1. | |
| 8.4361158155s | ACMD41:SD_SEND_OP_COND | 40 00 00 00 | OFF | 00 | | 281KHz | Ncs: 0. Ncr: 1. | |
| 8.4363993255s | CMD58:READ_OCR | 00 00 00 00 | OFF | 00 | R3: C0 FF 80 00 | 281KHz | Ncs: 0. Ncr: 1. | |

## SDQ

SDQ (Serial Data Quality) interface is an interface standard designed to improve the quality of data transmission, primarily in digital signal processing and communication systems. The SDQ interface is designed to improve data accuracy, reduce miscoding and interference, and ensure signal quality during high-speed data transmission.

### Settings



SDQ Settings

Parameters

Channel

SDQ: A0

Model: BQ2024

Display: Byte

Range

Decode Range

From: Buffer Head

To: Buffer Tail

Color

CMD: 🟨

DATA: 🟦

Reset: 🟧

Presence: 🟧

CRC: 🟩

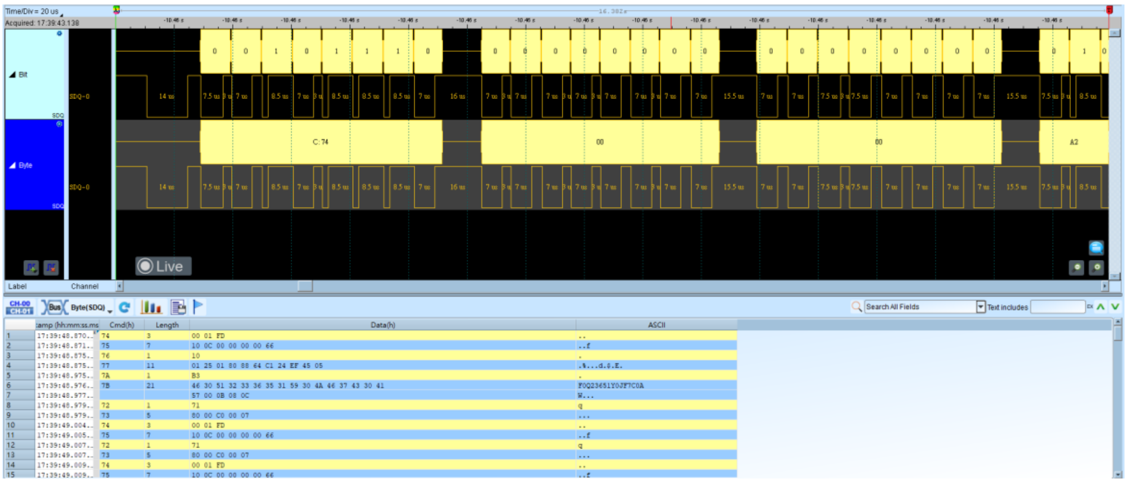
CRC Error: 🟥

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Model:** Set the IC model number. Currently supports BQ2024, BQ2025, BQ2026.

**Display:** The result of waveform resolution is displayed in Bit or Byte format.

## Result



|  | tamp (hh:mm:ss.ms | Cmd(h) | Length | Data(h) | ASCII |
|---|---|---|---|---|---|
| 1 | 17:39:48.870... | 74 | 3 | 00 01 FD | .. |
| 2 | 17:39:48.871... | 75 | 7 | 10 0C 00 00 00 00 66 | ..f |
| 3 | 17:39:48.875... | 76 | 1 | 10 | . |
| 4 | 17:39:48.875... | 77 | 11 | 01 25 01 80 88 64 C1 24 EF 45 05 | .%...d.$.E. |
| 5 | 17:39:48.975... | 7A | 1 | B3 | . |
| 6 | 17:39:48.976... | 7B | 21 | 46 30 51 32 33 36 35 31 59 30 4A 46 37 43 30 41 | F0Q23651Y0JF7C0A |
| 7 | 17:39:48.977... |  |  | 57 00 0B 08 0C | W... |
| 8 | 17:39:48.979... | 72 | 1 | 71 | q |
| 9 | 17:39:48.979... | 73 | 5 | 80 00 C0 00 07 | ... |
| 10 | 17:39:49.004... | 74 | 3 | 00 01 FD | .. |
| 11 | 17:39:49.005... | 75 | 7 | 10 0C 00 00 00 00 66 | ..f |
| 12 | 17:39:49.007... | 72 | 1 | 71 | q |
| 13 | 17:39:49.007... | 73 | 5 | 80 00 C0 00 07 | ... |
| 14 | 17:39:49.009... | 74 | 3 | 00 01 FD | .. |
| 15 | 17:39:49.009... | 75 | 7 | 10 0C 00 00 00 00 66 | ..f |

## SDR SDRAM

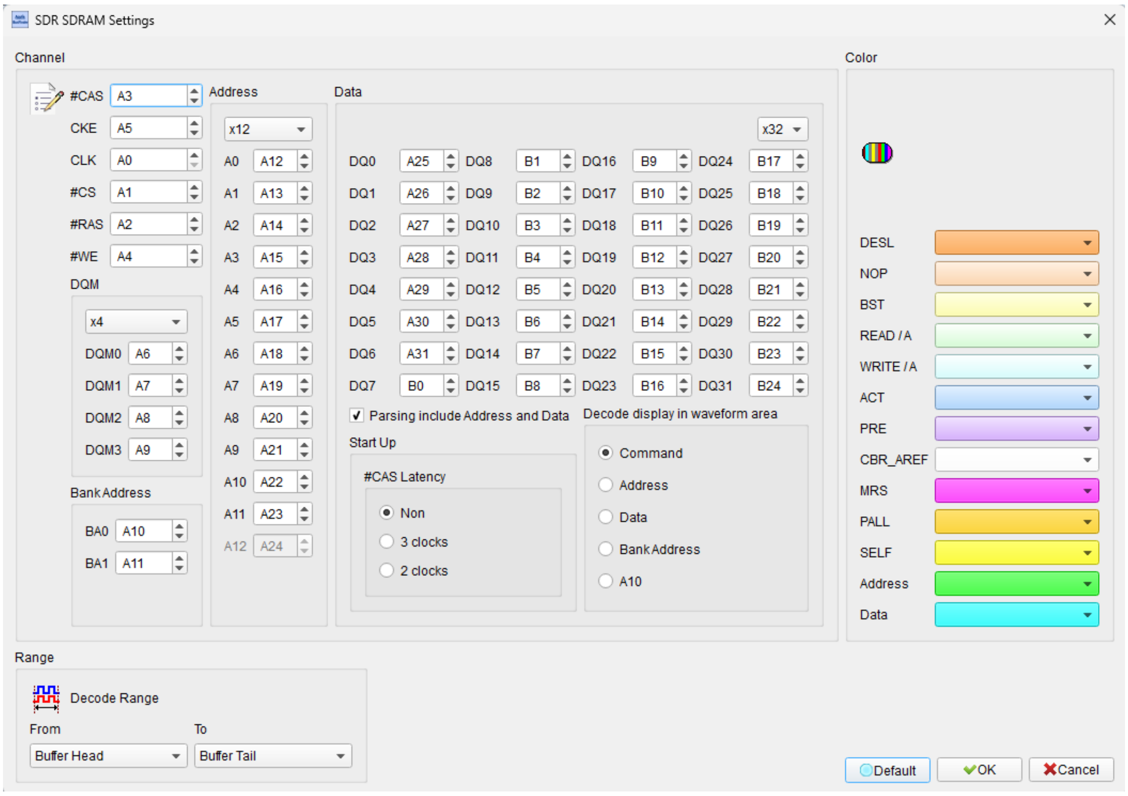
SDRAM (Synchronous Dynamic Random Access Memory) is a synchronous dynamic random access memory. Its characteristic is that it can synchronize the clock pulse of the memory with the host. Because it can only transmit data at the Rising edge, SDRAM can also be called SDR SDRAM (Single Data Rate SDRAM) .

SDRAM is different from the DDR SDRAM structure used in current computers. DDR (Double Data Rate) actually refers to DDR SDRAM (Double Data Rate SDRAM), which means that Rising/Falling edge can transmit data.

Due to the large number of channels required and the high signal speed, this decode is only available on LA3000+, LA4000+ or BusFinder models.

In addition, this decode only supports SDR SDRAM analysis, not DDR SDRAM.

### Settings



SDR SDRAM Settings

Channel

| | |
|---|---|
| #CAS | A3 |
| CKE | A5 |
| CLK | A0 |
| #CS | A1 |
| #RAS | A2 |
| #WE | A4 |

DQM: x4

| | |
|---|---|
| DQM0 | A6 |
| DQM1 | A7 |
| DQM2 | A8 |
| DQM3 | A9 |

Bank Address

| | |
|---|---|
| BA0 | A10 |
| BA1 | A11 |

Address: x12

| | |
|---|---|
| A0 | A12 |
| A1 | A13 |
| A2 | A14 |
| A3 | A15 |
| A4 | A16 |
| A5 | A17 |
| A6 | A18 |
| A7 | A19 |
| A8 | A20 |
| A9 | A21 |
| A10 | A22 |
| A11 | A23 |
| A12 | A24 |

Data: x32

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| DQ0 | A25 | DQ8 | B1 | DQ16 | B9 | DQ24 | B17 |
| DQ1 | A26 | DQ9 | B2 | DQ17 | B10 | DQ25 | B18 |
| DQ2 | A27 | DQ10 | B3 | DQ18 | B11 | DQ26 | B19 |
| DQ3 | A28 | DQ11 | B4 | DQ19 | B12 | DQ27 | B20 |
| DQ4 | A29 | DQ12 | B5 | DQ20 | B13 | DQ28 | B21 |
| DQ5 | A30 | DQ13 | B6 | DQ21 | B14 | DQ29 | B22 |
| DQ6 | A31 | DQ14 | B7 | DQ22 | B15 | DQ30 | B23 |
| DQ7 | B0 | DQ15 | B8 | DQ23 | B16 | DQ31 | B24 |

☑ Parsing include Address and Data

Start Up

#CAS Latency
- ● Non
- ○ 3 clocks
- ○ 2 clocks

Decode display in waveform area
- ● Command
- ○ Address
- ○ Data
- ○ Bank Address
- ○ A10

Color

DESL, NOP, BST, READ / A, WRITE / A, ACT, PRE, CBR_AREF, MRS, PALL, SELF, Address, Data

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of logic analyzer

Parsing include Address and Data:

When this item is not checked:

Only for simple analysis of SDRAM Command, just connect #CAS, CKE, #CS,

#RAS, the 6 channels of #WE, A10 can be analyzed. In this way, the number of

wiring can be reduced, but because it cannot knowing the status of Address, Data, etc., is only suitable for primary analysis. Please see Figure 1 below for the analysis results**.**

When checked:

Including all SDRAM pins for a complete analysis, the analysis results are shown in Figure 2 below

**Startup:**

#CAS Latency:

Set the delay time for SDRAM read operation

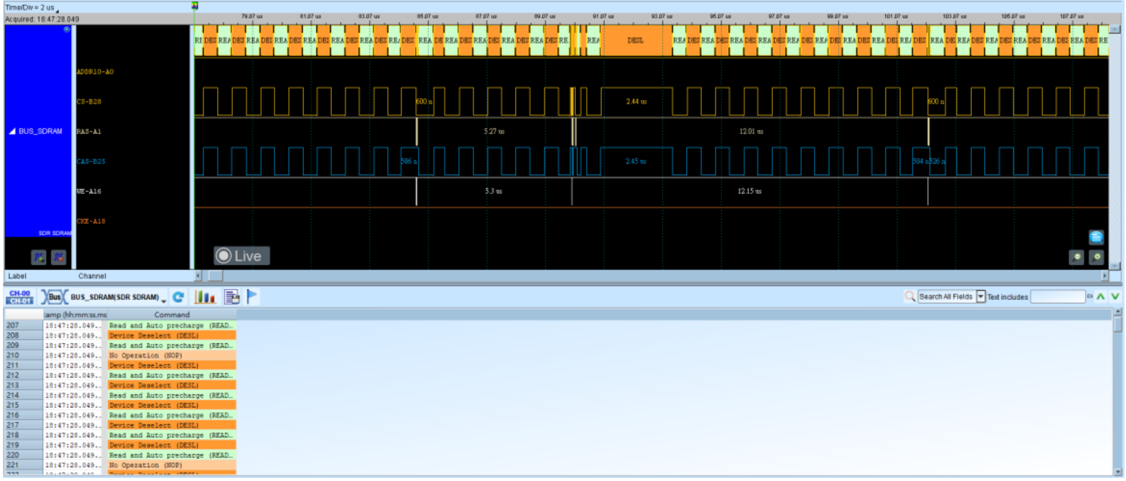
**Decode display in waveform area**

Because there are many states to be displayed by SDRAM, they cannot be displayed all at once in the waveform area.

Therefore, it is necessary to select the item to be viewed in the waveform area. If there are many items to be viewed at the same time, you can add multiple groups of the same SDRAM decoding, then set different decoding and display modes of the waveform area respectively. As shown in Figure 2 below, the left side multiple groups of channels have been added to distinguish the display.

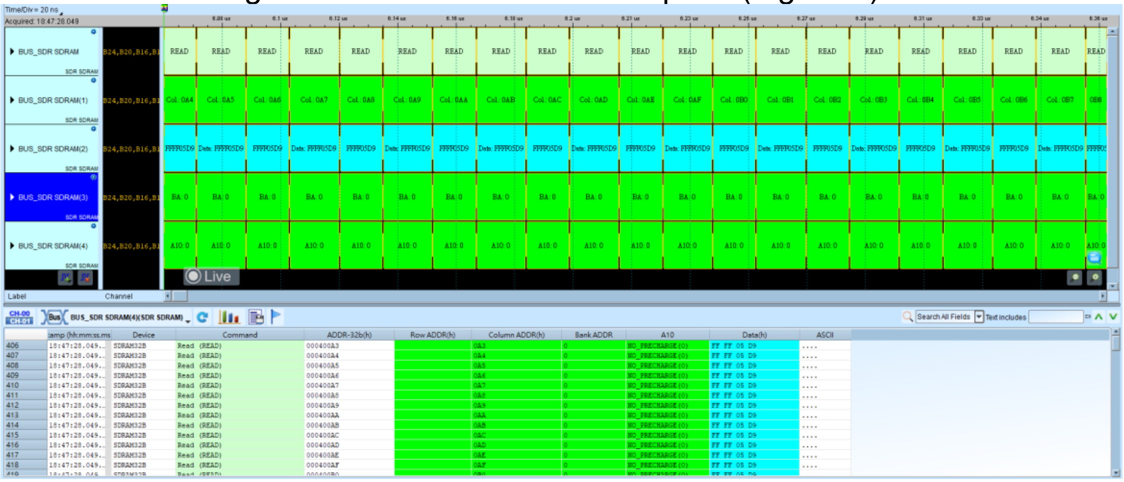
## Result

Uncheck the Parsing include Address and Data option (Figure 1)



|  | tamp (hh:mm:ss.ms | Command |
|---|---|---|
| 207 | 18:47:28.049... | Read and Auto precharge (READ... |
| 208 | 18:47:28.049... | Device Deselect (DESL) |
| 209 | 18:47:28.049... | Read and Auto precharge (READ... |
| 210 | 18:47:28.049... | No Operation (NOP) |
| 211 | 18:47:28.049... | Device Deselect (DESL) |
| 212 | 18:47:28.049... | Read and Auto precharge (READ... |
| 213 | 18:47:28.049... | Device Deselect (DESL) |
| 214 | 18:47:28.049... | Read and Auto precharge (READ... |
| 215 | 18:47:28.049... | Device Deselect (DESL) |
| 216 | 18:47:28.049... | Read and Auto precharge (READ... |
| 217 | 18:47:28.049... | Device Deselect (DESL) |
| 218 | 18:47:28.049... | Read and Auto precharge (READ... |
| 219 | 18:47:28.049... | Device Deselect (DESL) |
| 220 | 18:47:28.049... | Read and Auto precharge (READ... |
| 221 | 18:47:28.049... | No Operation (NOP) |

Check the Parsing include Address and Data option (Figure 2)

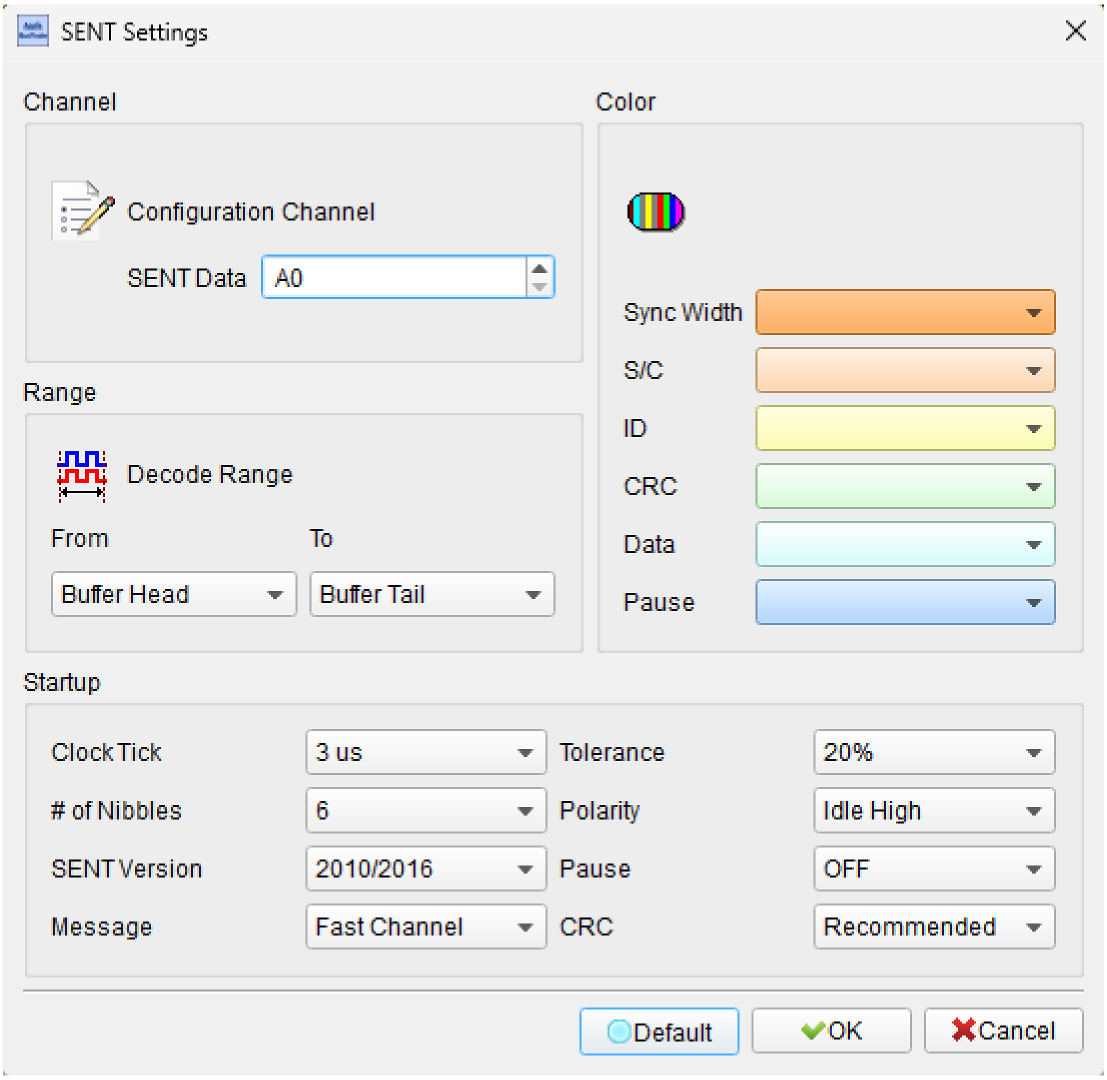


|  | tamp (hh:mm:ss.ms | Device | Command | ADDR-32b(h) | Row ADDR(h) | Column ADDR(h) | Bank ADDR | A10 | Data(h) | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|
| 406 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400A3 |  | 0A3 | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 407 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400A4 |  | 0A4 | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 408 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400A5 |  | 0A5 | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 409 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400A6 |  | 0A6 | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 410 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400A7 |  | 0A7 | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 411 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400A8 |  | 0A8 | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 412 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400A9 |  | 0A9 | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 413 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400AA |  | 0AA | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 414 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400AB |  | 0AB | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 415 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400AC |  | 0AC | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 416 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400AD |  | 0AD | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 417 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400AE |  | 0AE | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |
| 418 | 18:47:28.049... | SDRAM32B | Read (READ) | 000400AF |  | 0AF | 0 | NO_PRECHARGE (0) | FF FF 05 D9 | .... |

### SENT

The SENT (Single Edge Nibble Transmission) protocol is a communication protocol used in the field of automotive electronics, especially for data transmission between sensors and control units (ECUs) in the vehicle's internal electronic system. The SENT protocol is widely used in high-performance in-vehicle sensors such as wheel speedometers, position sensors, etc. It enables efficient data transfer with limited bandwidth and guarantees high-precision data transfer.

### Settings



SENT Settings

Channel

Configuration Channel

SENT Data: A0

Range

Decode Range

From: Buffer Head

To: Buffer Tail

Color

Sync Width

S/C

ID

CRC

Data

Pause

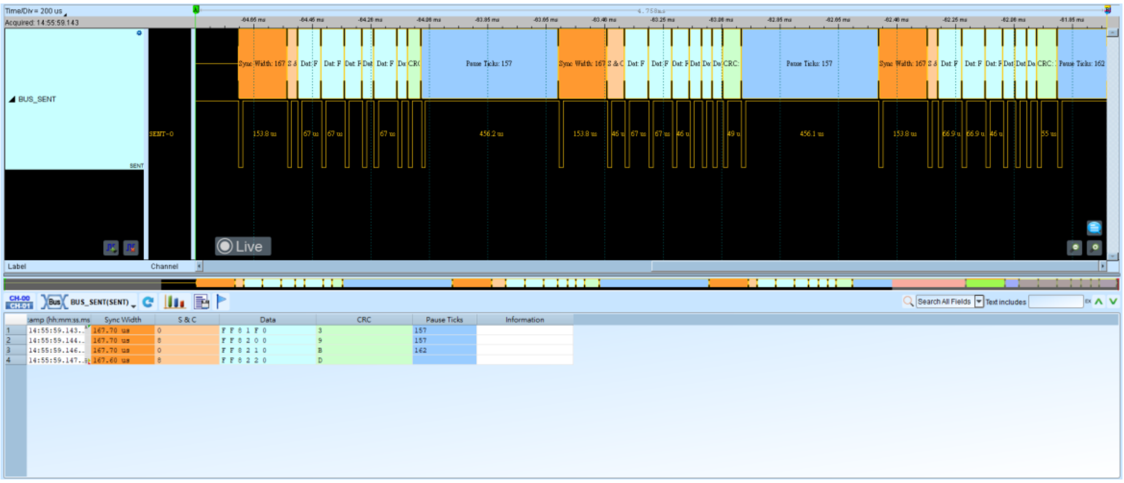
Startup

| | | | |
|---|---|---|---|
| Clock Tick | 3 us | Tolerance | 20% |
| # of Nibbles | 6 | Polarity | Idle High |
| SENT Version | 2010/2016 | Pause | OFF |
| Message | Fast Channel | CRC | Recommended |

**Channel:** Set the channel number of the Logic Analyzer to which each signal end is connected on the object to be tested.

**Startup:** Sets the startup setting for SENT before analysis.

### Result

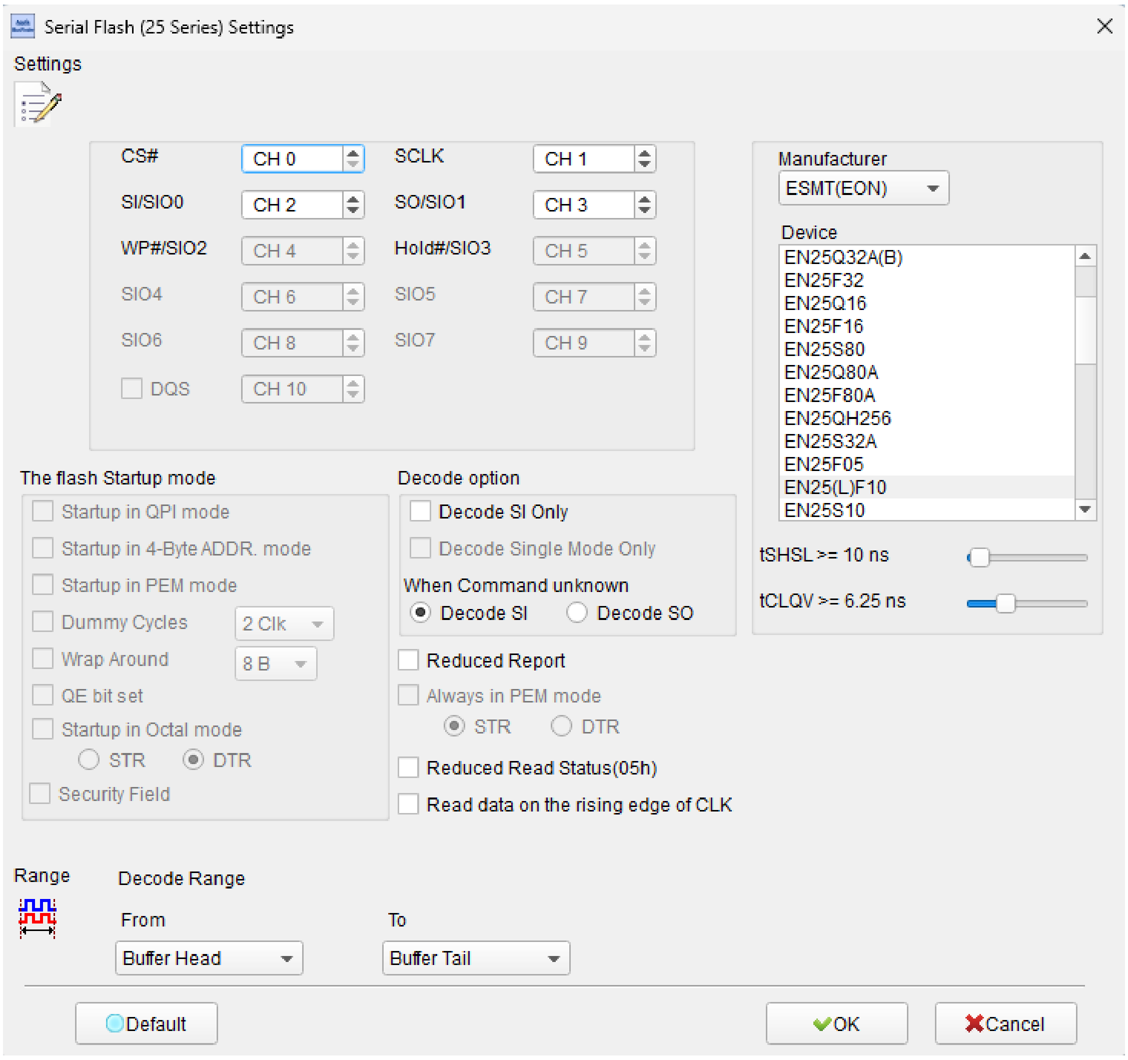


| | :amp (hh:mm:ss.ms | Sync Width | S & C | Data | CRC | Pause Ticks | Information |
|---|---|---|---|---|---|---|---|
| 1 | 14:55:59.143.. | 167.70 us | 0 | F F 8 1 F 0 | 3 | 157 | |
| 2 | 14:55:59.144.. | 167.70 us | 8 | F F 8 2 0 0 | 9 | 157 | |
| 3 | 14:55:59.146.. | 167.70 us | 0 | F F 8 2 1 0 | B | 162 | |
| 4 | 14:55:59.147.. | 167.60 us | 8 | F F 8 2 2 0 | D | | |

## Serial Flash

Serial flash (SPI Flash) 25, 35, etc. series use SPI/QPI/OPI protocol as their data transfer communication method. Serial flash bus analysis provides users with the ability to view commands and input/output bus messages at the same time when viewing signals, saving users the time of analyzing waveforms using the SPI bus.

## Settings



Serial Flash (25 Series) Settings

Settings

CS# CH 0　SCLK CH 1
SI/SIO0 CH 2　SO/SIO1 CH 3
WP#/SIO2 CH 4　Hold#/SIO3 CH 5
SIO4 CH 6　SIO5 CH 7
SIO6 CH 8　SIO7 CH 9
□ DQS CH 10

Manufacturer: ESMT(EON)

Device:
EN25Q32A(B)
EN25F32
EN25Q16
EN25F16
EN25S80
EN25Q80A
EN25F80A
EN25QH256
EN25S32A
EN25F05
EN25(L)F10
EN25S10

tSHSL >= 10 ns
tCLQV >= 6.25 ns

The flash Startup mode
□ Startup in QPI mode
□ Startup in 4-Byte ADDR. mode
□ Startup in PEM mode
□ Dummy Cycles 2 Clk
□ Wrap Around 8 B
□ QE bit set
□ Startup in Octal mode
○ STR ● DTR
□ Security Field

Decode option
□ Decode SI Only
□ Decode Single Mode Only
When Command unknown
● Decode SI ○ Decode SO
□ Reduced Report
□ Always in PEM mode
● STR ○ DTR
□ Reduced Read Status(05h)
□ Read data on the rising edge of CLK

Range
Decode Range
From: Buffer Head
To: Buffer Tail

**CS#:** Chip Select of transfer signal.

**SCLK:** Clock of transfer signal.

**SIO0-SIO7:** Data pin of transfer signal.

**Manufacturer:** The main purpose of this function is to select the correct Flash model, tCLQV and tSHSL for command parsing. If no exact model is found, the user can also select a model that is compatible with the command format.

**The Flash Startup Mode:** Since the Serial Flash can switch the operation mode with commands, the Logic Analyzer does not know the current operation mode of the actual Serial Flash when it captures the waveforms. Therefore, the user should be informed if necessary. When the user selects a Flash model that does not support mode switching, the relevant option will be disabled and cannot be set.

**QPI Mode:** Refers to Quad Peripheral Interface Mode or Quad SPI Mode.

**4-Byte Mode:** Refers to the 4-Byte Address Mode.

**PEM Mode:** Refers to the Performance Enhancement Mode.

**Dummy Cycles:** Some Read instructions wait for Dummy cycles, and the number of cycles they wait for can be preset.

**Wrap Around:** The value of Wrap around can be preset.

**QE bit:** QE bit in the status register, which can be used for QPI mode enable/disable control.

**Startup in Octal Mode:** Refers to the OPI mode.

**Security Field:** We provide Flash with AES encryption and decryption function, please contact us if user need to use it.

**Decode SI Only:** When checked, the program will use Single mode to analyze the waveforms in 3-line mode. These 3 lines are CS# / SCLK / SI.

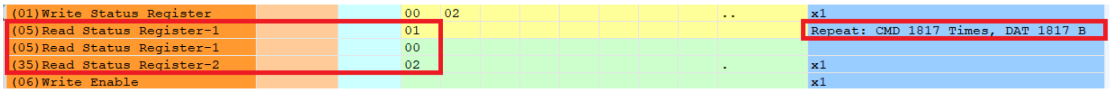
**Decode Single Mode Only:** When this option is selected, the program will use Single mode to analyze the waveform in 4-line mode. These 4 lines are CS / Clock / SI / SO. In this case, the program will ignore the command to switch to multi-line mode. If unchecked, the program will analyze the waveform in 4-wire or 6-wire mode according to the selected Flash model.

**When Command Unknown:** Decoded for SO or SI only.

**Always in PEM Mode:** When checked, the analyzer will maintain the PEM mode regardless of the command setting, and you can choose to maintain in STR or DTR mode.

**Reduce Read Status(05h):** When checked, the Read Status (05h) command that repeats the unchanged data will be combined into a single command when generating an analysis report, and the number of repetitions will be displayed, thus reducing the number of reports and making it easier to view.

As you can see from the example below, the Read status data = 01 is repeated 1817 times.

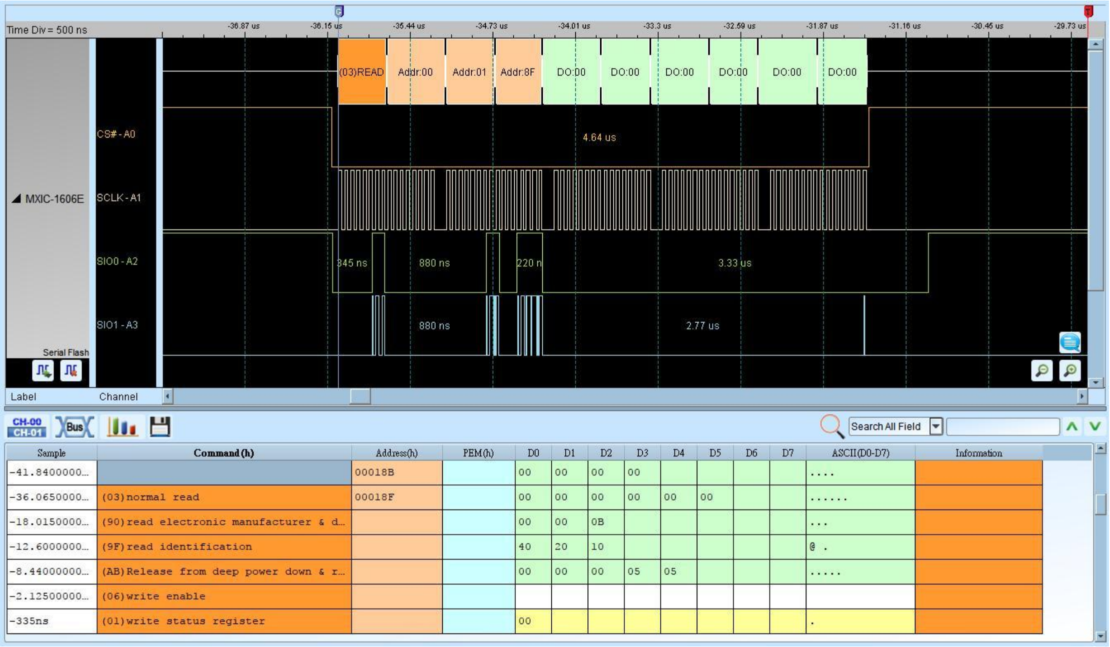


| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (01)Write Status Register | | | 00 | 02 | | | | | | | .. | x1 |
| (05)Read Status Register-1 | | | 01 | | | | | | | | | Repeat: CMD 1817 Times, DAT 1817 B |
| (05)Read Status Register-1 | | | 00 | | | | | | | | | |
| (35)Read Status Register-2 | | | 02 | | | | | | | | . | x1 |
| (06)Write Enable | | | | | | | | | | | | x1 |

**Read data on the rising edge of CLK:** For SDR Read Data mode, if the Clock signal Duty is not stable enough and the Latch Data out cannot be correct after setting tCLQV value, the Latch bit can be set to Next rising edge.

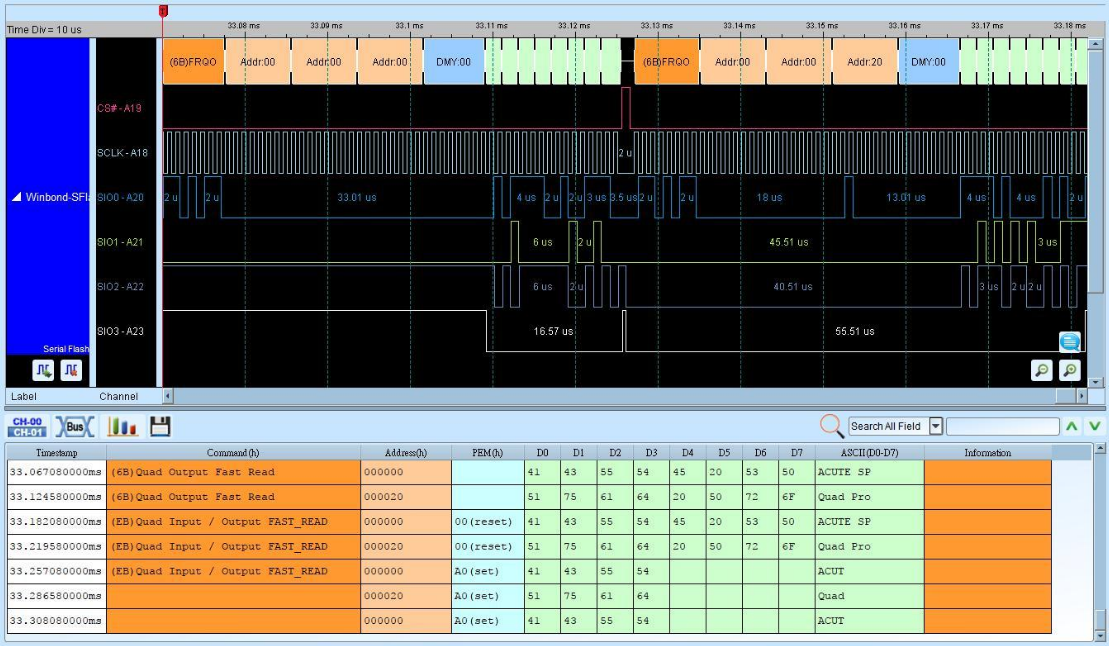
**Result**

**Using SPI Mode Serial Flash Decoding Situation**



| Sample | Command (h) | Address(h) | PEM(h) | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII(D0-D7) | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| -41.8400000... |  | 00018B |  | 00 | 00 | 00 | 00 |  |  |  |  | .... |  |
| -36.0650000... | (03)normal read | 00018F |  | 00 | 00 | 00 | 00 | 00 | 00 |  |  | ...... |  |
| -18.0150000... | (90)read electronic manufacturer & d... |  |  | 00 | 00 | 0B |  |  |  |  |  | ... |  |
| -12.6000000... | (9F)read identification |  |  | 40 | 20 | 10 |  |  |  |  |  | @ . |  |
| -8.44000000... | (AB)Release from deep power down & r... |  |  | 00 | 00 | 00 | 05 | 05 |  |  |  | ..... |  |
| -2.12500000... | (06)write enable |  |  |  |  |  |  |  |  |  |  |  |  |
| -335ns | (01)write status register |  |  | 00 |  |  |  |  |  |  |  | . |  |

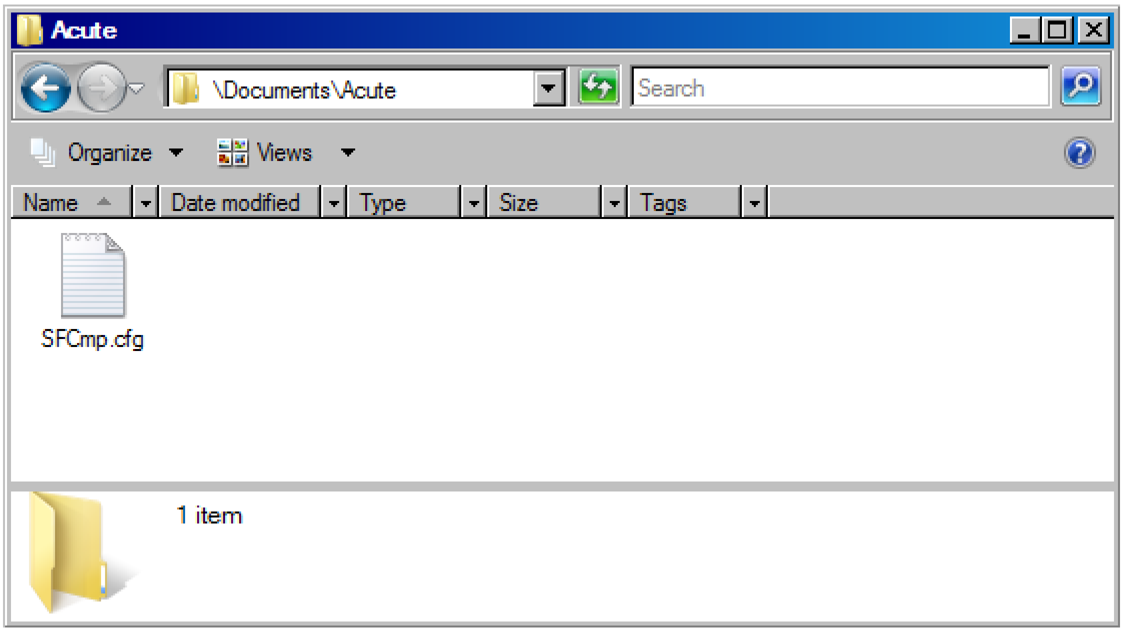
Using QPI Mode Serial Flash Decoding Scenario



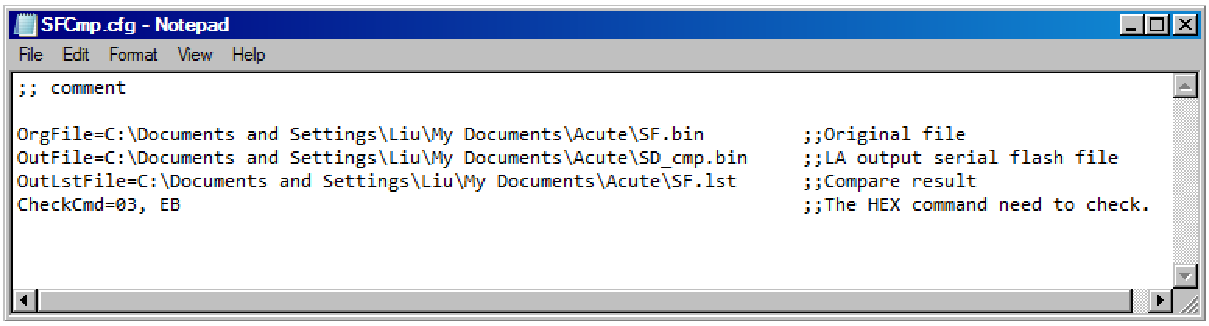
| Timestamp | Command (h) | Address(h) | PEM(h) | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII(D0-D7) | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 33.067080000ms | (6B)Quad Output Fast Read | 000000 |  | 41 | 43 | 55 | 54 | 45 | 20 | 53 | 50 | ACUTE SP |  |
| 33.124580000ms | (6B)Quad Output Fast Read | 000020 |  | 51 | 75 | 61 | 64 | 20 | 50 | 72 | 6F | Quad Pro |  |
| 33.182080000ms | (EB)Quad Input / Output FAST_READ | 000000 | 00(reset) | 41 | 43 | 55 | 54 | 45 | 20 | 53 | 50 | ACUTE SP |  |
| 33.219580000ms | (EB)Quad Input / Output FAST_READ | 000020 | 00(reset) | 51 | 75 | 61 | 64 | 20 | 50 | 72 | 6F | Quad Pro |  |
| 33.257080000ms | (EB)Quad Input / Output FAST_READ | 000000 | A0(set) | 41 | 43 | 55 | 54 |  |  |  |  | ACUT |  |
| 33.286580000ms |  | 000020 | A0(set) | 51 | 75 | 61 | 64 |  |  |  |  | Quad |  |
| 33.308080000ms |  | 000000 | A0(set) | 41 | 43 | 55 | 54 |  |  |  |  | ACUT |  |

**Serial Flash data Comparison :** Compare the Serial Flash data by the waveform files.

**Method:** Create a file by text editor and save it as SFCmp.cfg in order to compare with the real Serial Flash waveform to find the bug, the default path is “My Documents\Acute”



SFCmp.cfg information:



```
;; comment

OrgFile=C:\Documents and Settings\Liu\My Documents\Acute\SF.bin        ;;Original file
OutFile=C:\Documents and Settings\Liu\My Documents\Acute\SD_cmp.bin    ;;LA output serial flash file
OutLstFile=C:\Documents and Settings\Liu\My Documents\Acute\SF.lst     ;;Compare result
CheckCmd=03, EB                                                        ;;The HEX command need to check.
```

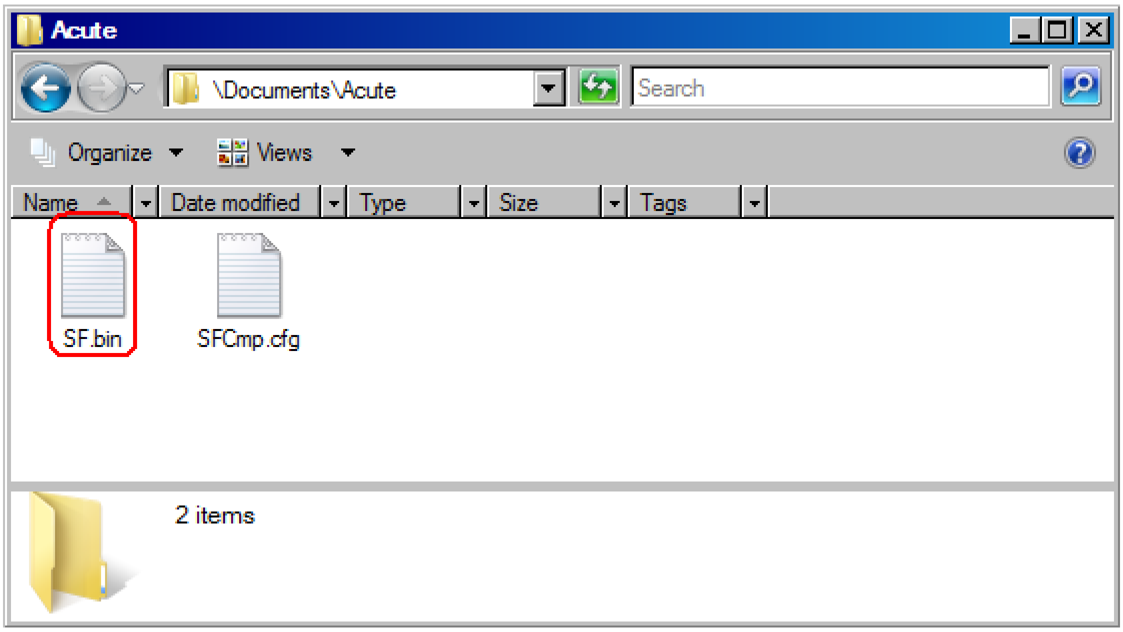
**OrgFile=File_Path:** Key in the file path of the original Serial Flash data file (.bin).

**OutFile=File_Path:** Key in the file path of the Serial Flash output file.

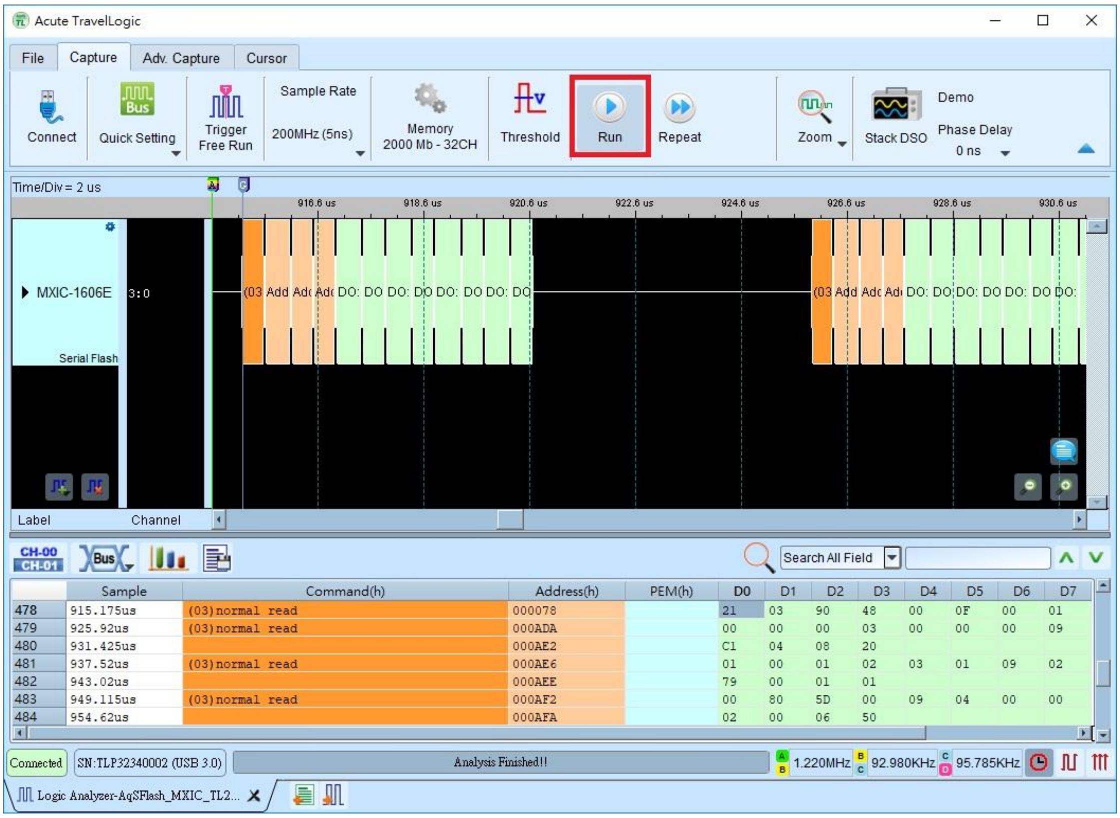
**OutLstFile=File_Path:** Key in the file path of the comparison result. The file name will has extension ".lst".

**CheckCmd=Serial Flash command:** Key in the command in Hex that are separated by commas.

Save the OrgFile to the OrgFile file path.



Run the Serial Flash Bus Decode to capture the Serial Flash signal.



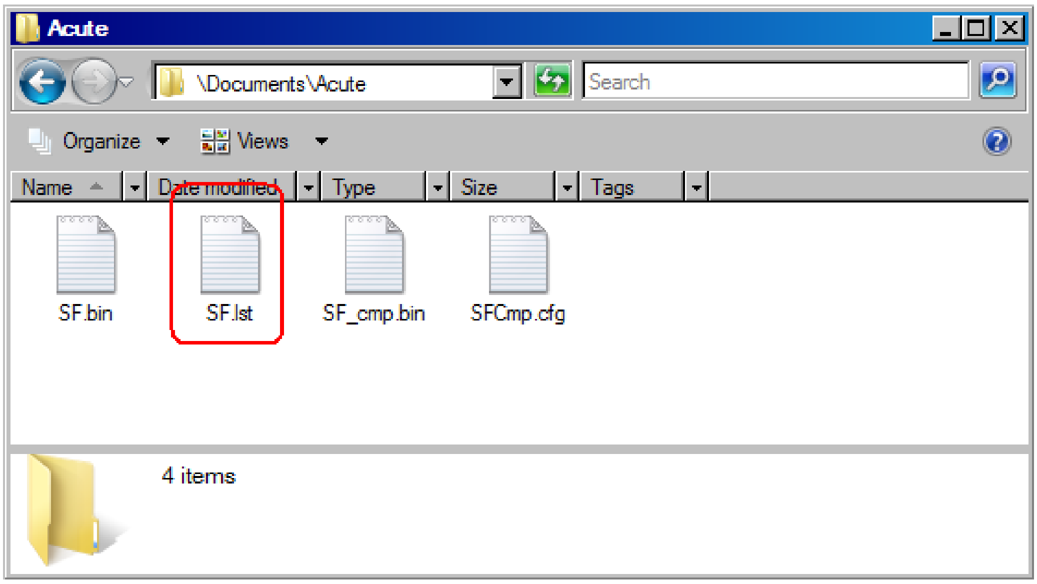
|  | Sample | Command(h) | Address(h) | PEM(h) | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 478 | 915.175us | (03)normal read | 000078 |  | 21 | 03 | 90 | 48 | 00 | 0F | 00 | 01 |
| 479 | 925.92us | (03)normal read | 000ADA |  | 00 | 00 | 00 | 03 | 00 | 00 | 00 | 09 |
| 480 | 931.425us |  | 000AE2 |  | C1 | 04 | 08 | 20 |  |  |  |  |
| 481 | 937.52us | (03)normal read | 000AE6 |  | 01 | 00 | 01 | 02 | 03 | 01 | 09 | 02 |
| 482 | 943.02us |  | 000AEE |  | 79 | 00 | 01 | 01 |  |  |  |  |
| 483 | 949.115us | (03)normal read | 000AF2 |  | 00 | 80 | 5D | 00 | 09 | 04 | 00 | 00 |
| 484 | 954.62us |  | 000AFA |  | 02 | 00 | 06 | 50 |  |  |  |  |

Connected | SN:TLP32340002 (USB 3.0) | Analysis Finished!! | 1.220MHz | 92.980KHz | 95.785KHz

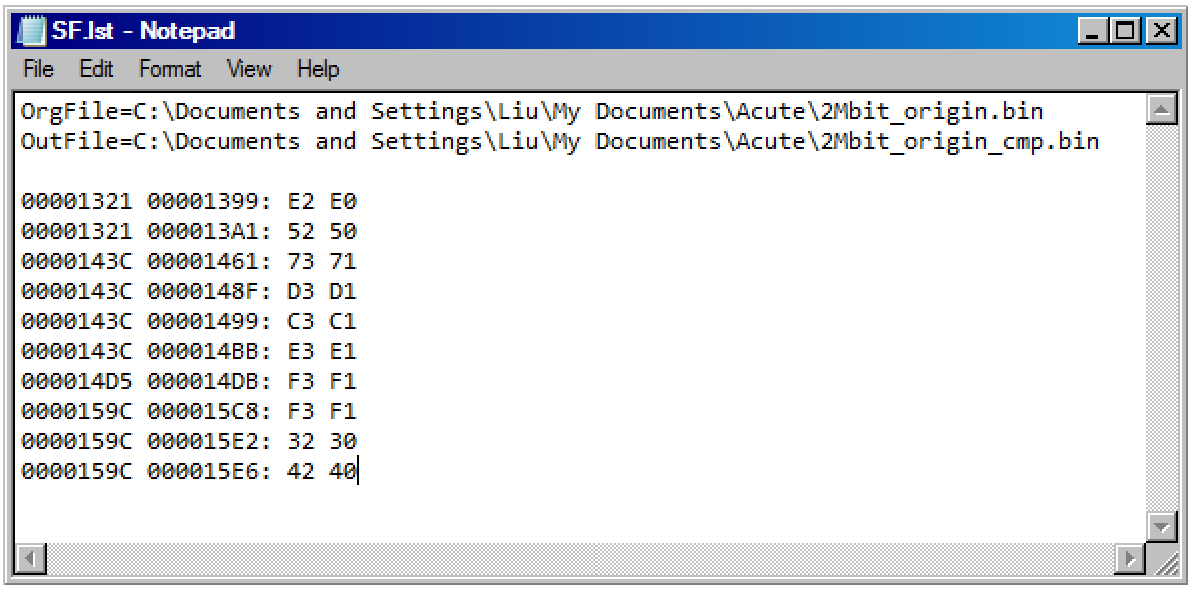
Logic Analyzer-AqSFlash_MXIC_TL2...

If the OutFile does not exit, it will copy the OrgFile to the OutFile and write the data to it according to the CheckCmd.

Compare result:



The OutLstFile:



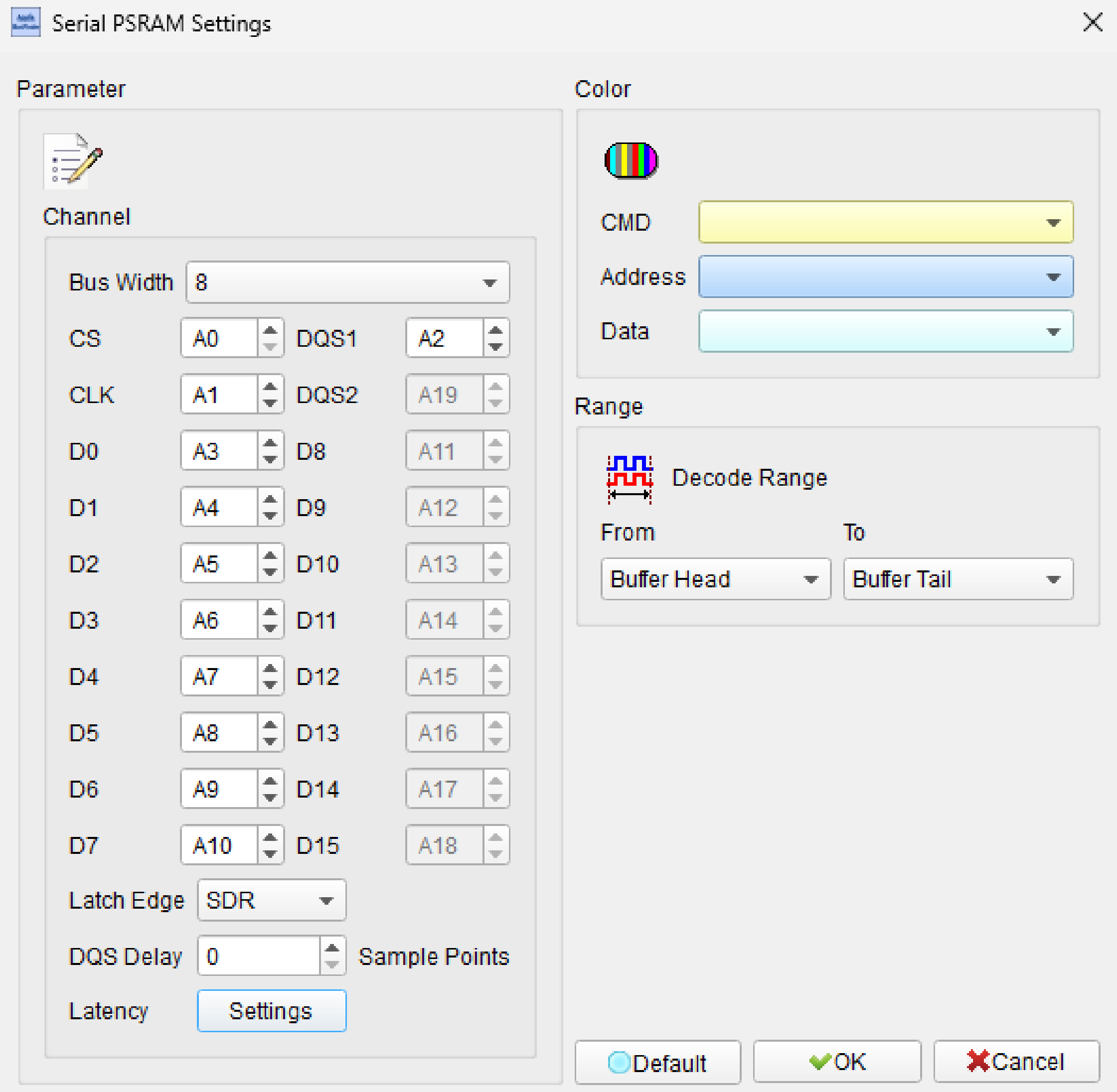
```
OrgFile=C:\Documents and Settings\Liu\My Documents\Acute\2Mbit_origin.bin
OutFile=C:\Documents and Settings\Liu\My Documents\Acute\2Mbit_origin_cmp.bin

00001321 00001399: E2 E0
00001321 000013A1: 52 50
0000143C 00001461: 73 71
0000143C 0000148F: D3 D1
0000143C 00001499: C3 C1
0000143C 000014BB: E3 E1
000014D5 000014DB: F3 F1
0000159C 000015C8: F3 F1
0000159C 000015E2: 32 30
0000159C 000015E6: 42 40
```

The first column is the compared address from OrgFile, the second column is the different address from OutFile.

## Serial PSRAM

Serial PSRAM is a special form of PSRAM (Pseudostatic RAM) that uses a serial interface to communicate with a microprocessor or other system.PSRAM is a type of memory that is intermediate between Dynamic Random Access Memory (DRAM) and Static Random Access Memory (SRAM). It has the advantages of high density of DRAM, but similar to SRAM, it offers relatively simple operation and simplifies memory control to some extent.

## Settings



Serial PSRAM Settings

Parameter

Channel

Bus Width: 8

| | | | |
|---|---|---|---|
| CS | A0 | DQS1 | A2 |
| CLK | A1 | DQS2 | A19 |
| D0 | A3 | D8 | A11 |
| D1 | A4 | D9 | A12 |
| D2 | A5 | D10 | A13 |
| D3 | A6 | D11 | A14 |
| D4 | A7 | D12 | A15 |
| D5 | A8 | D13 | A16 |
| D6 | A9 | D14 | A17 |
| D7 | A10 | D15 | A18 |

Latch Edge: SDR

DQS Delay: 0 Sample Points

Latency: Settings

Color

CMD

Address

Data

Range

Decode Range

From: Buffer Head

To: Buffer Tail

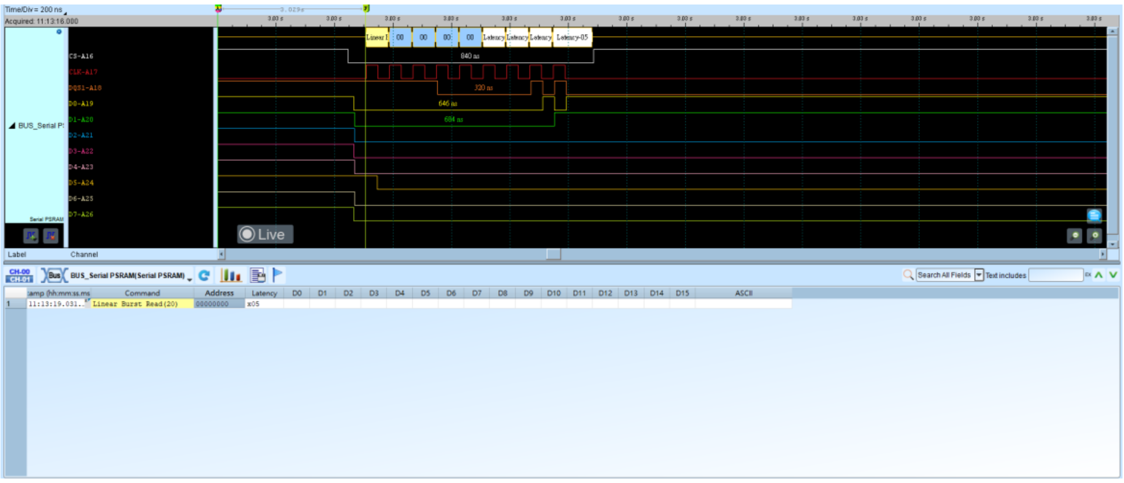
**Bus Width:** Set the width of the bus to 8 or 16.

**Latch Edge:** Choose using SDR or DDR to latch data.

**DQS Delay:** Set the DQS delay time in units of sampling points.

**Latency:** Details Specifies the amount of time that a particular piece of data will be delayed.

## Result

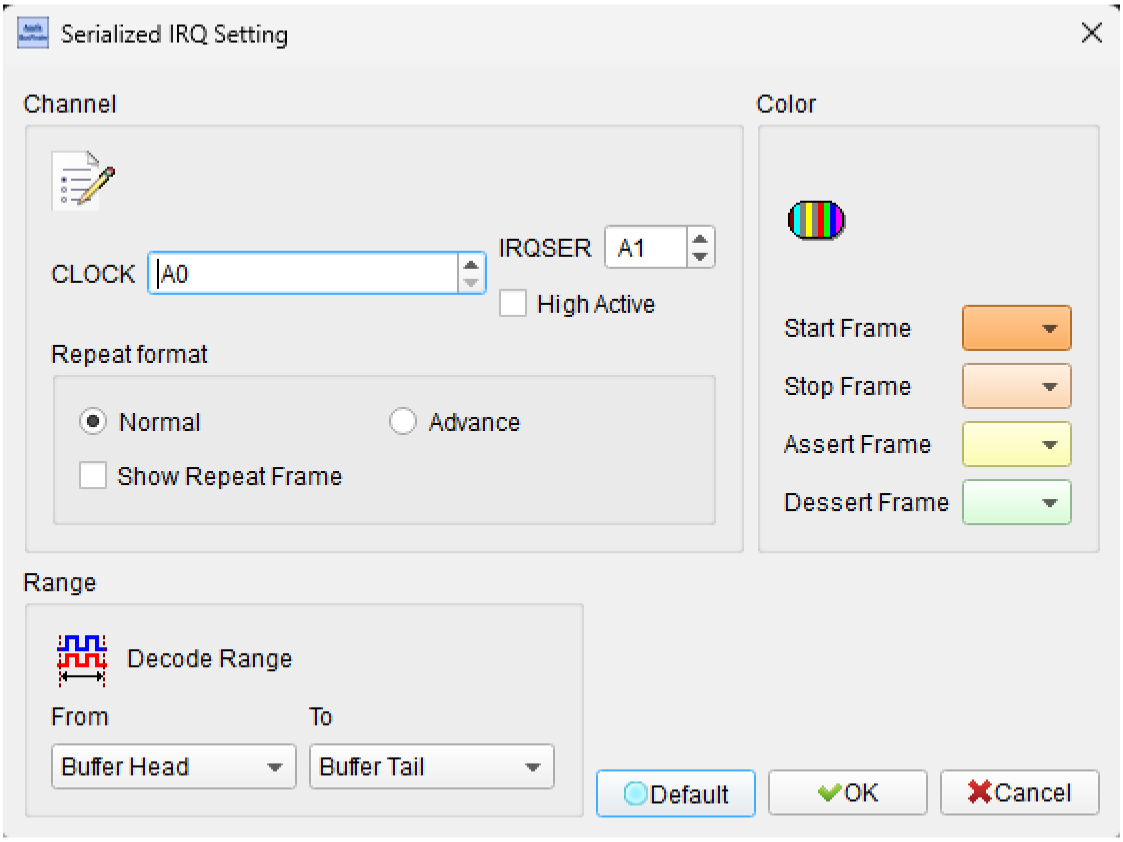


| | amp (hh:mm:ss.ms | Command | Address | Latency | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 11:13:19.031.. | Linear Burst Read(20) | 00000000 | x05 | | | | | | | | | | | | | | | | | |

### Serialized IRQ

Serialized IRQ/Data is a communication protocol composed of two lines, PCI-Clock and IRQSER, used to transmit interrupt status. An IRQSER cycle essentially consists of three parts: Start, IRQ/Data, and Stop Frame. The operating modes are divided into Continuous mode and Quiet mode. In Continuous mode, the source of the Start Frame is not restricted, but in Quiet mode, only the Host can generate the Start Frame signal.

### Settings



Serialized IRQ Setting

Channel

CLOCK A0

IRQSER A1

□ High Active

Repeat format

● Normal ○ Advance

□ Show Repeat Frame

Color

Start Frame

Stop Frame

Assert Frame

Dessert Frame

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**CLOCK:** PCI Clock channel

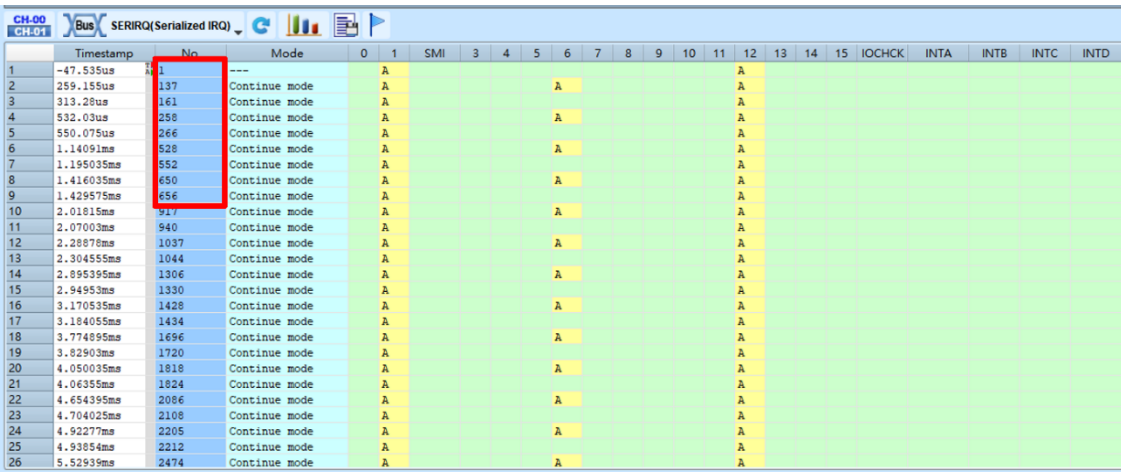
**IRQSER;** IRQSER channel

**Normal:** Not show repeat frame

**High Active:** Enables the user to adjust the numeric judgment conditions.

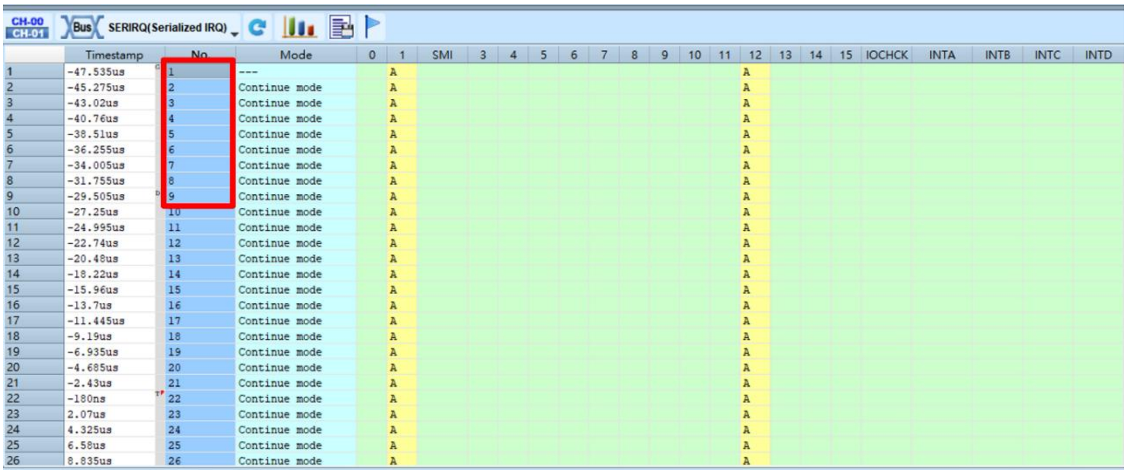
Enabled when checked.

Hide Repeat Frame (default)



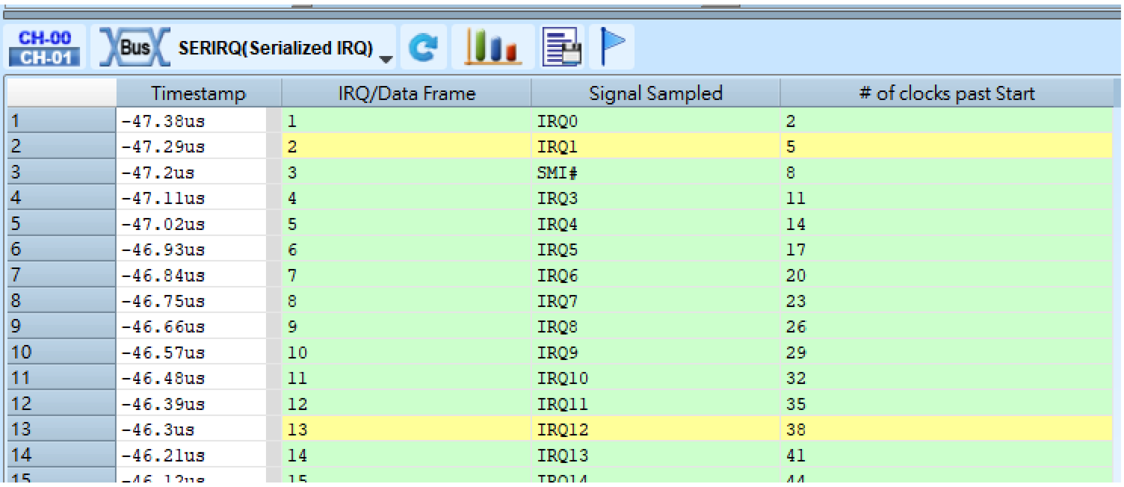
|  | Timestamp | No | Mode | 0 | 1 | SMI | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | IOCHCK | INTA | INTB | INTC | INTD |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -47.535us | 1 | --- |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 2 | 259.155us | 137 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 3 | 313.28us | 161 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 4 | 532.03us | 258 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 5 | 550.075us | 266 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 6 | 1.14091ms | 528 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 7 | 1.195035ms | 552 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 8 | 1.416035ms | 650 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 9 | 1.429575ms | 656 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 10 | 2.01815ms | 917 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 11 | 2.07003ms | 940 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 12 | 2.28878ms | 1037 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 13 | 2.304555ms | 1044 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 14 | 2.895395ms | 1306 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 15 | 2.94953ms | 1330 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 16 | 3.170535ms | 1428 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 17 | 3.184055ms | 1434 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 18 | 3.774895ms | 1696 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 19 | 3.82903ms | 1720 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 20 | 4.050035ms | 1818 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 21 | 4.06355ms | 1824 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 22 | 4.654395ms | 2086 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 23 | 4.704025ms | 2108 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 24 | 4.92277ms | 2205 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 25 | 4.93854ms | 2212 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 26 | 5.52939ms | 2474 | Continue mode |  | A |  |  |  |  | A |  |  |  |  |  | A |  |  |  |  |  |  |  |  |

Show repeat frame



|  | Timestamp | No | Mode | 0 | 1 | SMI | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | IOCHCK | INTA | INTB | INTC | INTD |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -47.535us | 1 | --- |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 2 | -45.275us | 2 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 3 | -43.02us | 3 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 4 | -40.76us | 4 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 5 | -38.51us | 5 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 6 | -36.255us | 6 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 7 | -34.005us | 7 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 8 | -31.755us | 8 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 9 | -29.505us | 9 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 10 | -27.25us | 10 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 11 | -24.995us | 11 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 12 | -22.74us | 12 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 13 | -20.48us | 13 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 14 | -18.22us | 14 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 15 | -15.96us | 15 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 16 | -13.7us | 16 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 17 | -11.445us | 17 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 18 | -9.19us | 18 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 19 | -6.935us | 19 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 20 | -4.685us | 20 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 21 | -2.43us | 21 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 22 | -180ns | 22 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 23 | 2.07us | 23 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 24 | 4.325us | 24 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 25 | 6.58us | 25 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |
| 26 | 8.835us | 26 | Continue mode |  | A |  |  |  |  |  |  |  |  |  |  | A |  |  |  |  |  |  |  |  |

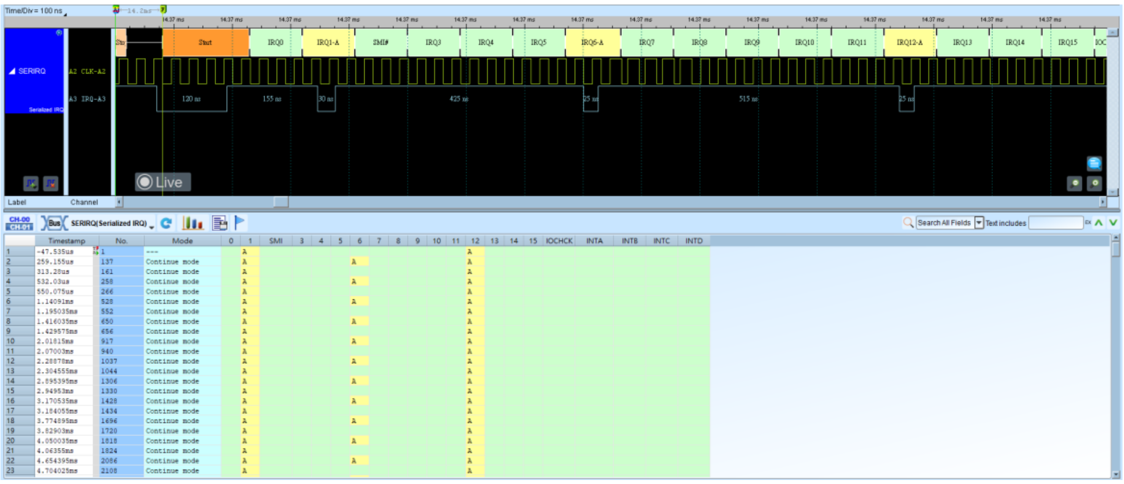
**Advance:** Spread all IRQ/Data in a frame on different lines.



| | Timestamp | IRQ/Data Frame | Signal Sampled | # of clocks past Start |
|---|---|---|---|---|
| 1 | -47.38us | 1 | IRQ0 | 2 |
| 2 | -47.29us | 2 | IRQ1 | 5 |
| 3 | -47.2us | 3 | SMI# | 8 |
| 4 | -47.11us | 4 | IRQ3 | 11 |
| 5 | -47.02us | 5 | IRQ4 | 14 |
| 6 | -46.93us | 6 | IRQ5 | 17 |
| 7 | -46.84us | 7 | IRQ6 | 20 |
| 8 | -46.75us | 8 | IRQ7 | 23 |
| 9 | -46.66us | 9 | IRQ8 | 26 |
| 10 | -46.57us | 10 | IRQ9 | 29 |
| 11 | -46.48us | 11 | IRQ10 | 32 |
| 12 | -46.39us | 12 | IRQ11 | 35 |
| 13 | -46.3us | 13 | IRQ12 | 38 |
| 14 | -46.21us | 14 | IRQ13 | 41 |
| 15 | -46.12us | 15 | IRQ14 | 44 |

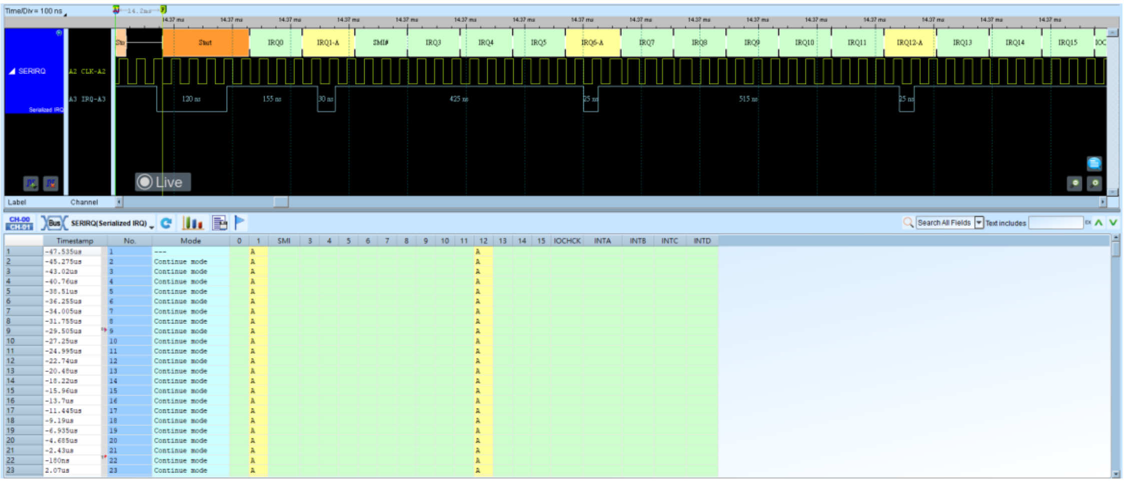
## Result

### Normal mode (Hide Repeat Frame)



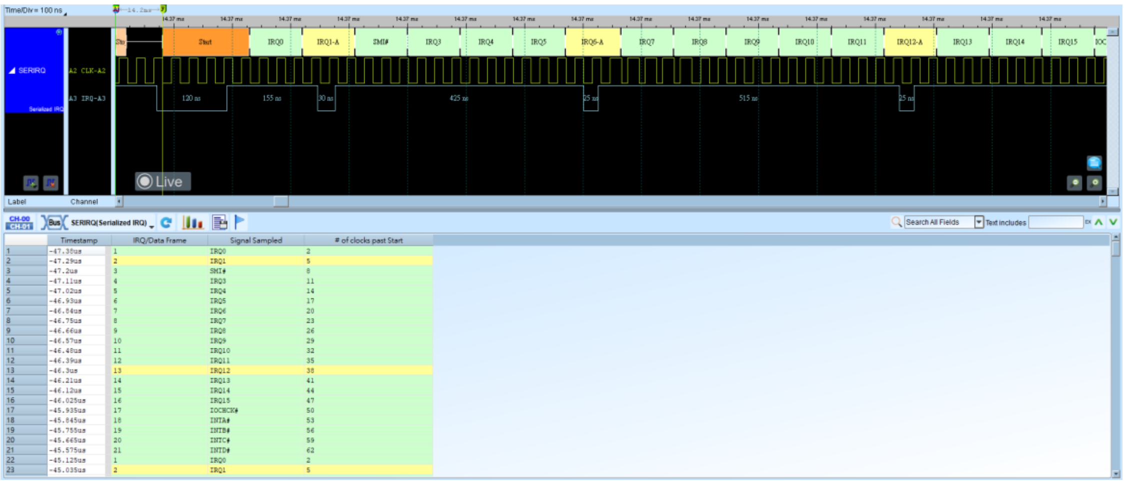
| | Timestamp | No. | Mode | 0 | 1 | SMI | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | IOCHCK | INTA | INTB | INTC | INTD |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -47.535us | 1 | --- | | A | | | | | | | | | | | A | | | | | | | | |
| 2 | 259.155us | 137 | Continue mode | | A | | | | | A | | | | | | A | | | | | | | | |
| 3 | 313.28us | 161 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 4 | 532.03us | 258 | Continue mode | | A | | | | | A | | | | | | A | | | | | | | | |
| 5 | 550.075us | 266 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 6 | 1.14091ms | 528 | Continue mode | | A | | | | | A | | | | | | A | | | | | | | | |
| 7 | 1.195035ms | 552 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 8 | 1.416035ms | 650 | Continue mode | | A | | | | | A | | | | | | A | | | | | | | | |
| 9 | 1.429575ms | 656 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 10 | 2.01815ms | 917 | Continue mode | | A | | | | | A | | | | | | A | | | | | | | | |
| 11 | 2.07003ms | 940 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 12 | 2.28878ms | 1037 | Continue mode | | A | | | | | A | | | | | | A | | | | | | | | |
| 13 | 2.304555ms | 1044 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 14 | 2.895395ms | 1306 | Continue mode | | A | | | | | A | | | | | | A | | | | | | | | |
| 15 | 2.94953ms | 1330 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 16 | 3.170535ms | 1428 | Continue mode | | A | | | | | A | | | | | | A | | | | | | | | |
| 17 | 3.184055ms | 1434 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 18 | 3.774895ms | 1696 | Continue mode | | A | | | | | A | | | | | | A | | | | | | | | |
| 19 | 3.82903ms | 1720 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 20 | 4.050035ms | 1818 | Continue mode | | A | | | | | A | | | | | | A | | | | | | | | |
| 21 | 4.06355ms | 1824 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 22 | 4.654395ms | 2086 | Continue mode | | A | | | | | A | | | | | | A | | | | | | | | |
| 23 | 4.704025ms | 2108 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |

### Normal mode(Show repeat frame)



| | Timestamp | No. | Mode | 0 | 1 | SMI | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | IOCHCK | INTA | INTB | INTC | INTD |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -47.535us | 1 | --- | | A | | | | | | | | | | | A | | | | | | | | |
| 2 | -45.275us | 2 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 3 | -43.02us | 3 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 4 | -40.76us | 4 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 5 | -38.51us | 5 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 6 | -36.255us | 6 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 7 | -34.005us | 7 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 8 | -31.755us | 8 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 9 | -29.505us | 9 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 10 | -27.25us | 10 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 11 | -24.995us | 11 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 12 | -22.74us | 12 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 13 | -20.48us | 13 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 14 | -18.22us | 14 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 15 | -15.96us | 15 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 16 | -13.7us | 16 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 17 | -11.445us | 17 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 18 | -9.19us | 18 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 19 | -6.935us | 19 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 20 | -4.685us | 20 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 21 | -2.43us | 21 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 22 | -180ns | 22 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |
| 23 | 2.07us | 23 | Continue mode | | A | | | | | | | | | | | A | | | | | | | | |

### Advance mode

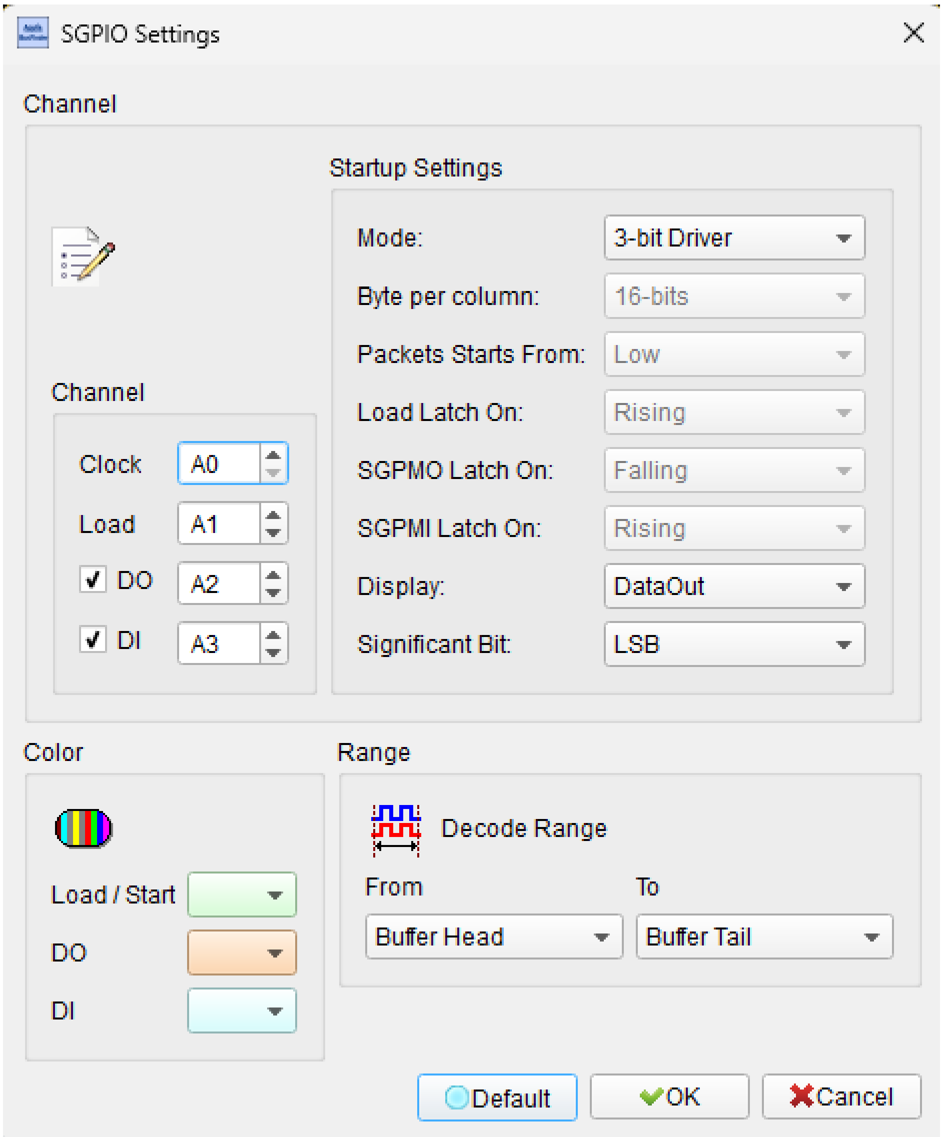


| | Timestamp | IRQ/Data Frame | Signal Sampled | # of clocks past Start |
|---|---|---|---|---|
| 1 | -47.38us | 1 | IRQ0 | 2 |
| 2 | -47.29us | 2 | IRQ1 | 5 |
| 3 | -47.2us | 3 | SMI# | 8 |
| 4 | -47.11us | 4 | IRQ3 | 11 |
| 5 | -47.02us | 5 | IRQ4 | 14 |
| 6 | -46.93us | 6 | IRQ5 | 17 |
| 7 | -46.84us | 7 | IRQ6 | 20 |
| 8 | -46.75us | 8 | IRQ7 | 23 |
| 9 | -46.66us | 9 | IRQ8 | 26 |
| 10 | -46.57us | 10 | IRQ9 | 29 |
| 11 | -46.48us | 11 | IRQ10 | 32 |
| 12 | -46.39us | 12 | IRQ11 | 35 |
| 13 | -46.3us | 13 | IRQ12 | 38 |
| 14 | -46.21us | 14 | IRQ13 | 41 |
| 15 | -46.12us | 15 | IRQ14 | 44 |
| 16 | -46.025us | 16 | IRQ15 | 47 |
| 17 | -45.935us | 17 | IOCHCK# | 50 |
| 18 | -45.845us | 18 | INTA# | 53 |
| 19 | -45.755us | 19 | INTB# | 56 |
| 20 | -45.665us | 20 | INTC# | 59 |
| 21 | -45.575us | 21 | INTD# | 62 |
| 22 | -45.125us | 1 | IRQ0 | 2 |
| 23 | -45.035us | 2 | IRQ1 | 5 |

## SGPIO

SGPIO (Serial General Purpose Input Output Serial) is a general-purpose input/output that can be controlled by the user.

### Settings



SGPIO Settings

Channel

Startup Settings

| | |
|---|---|
| Mode: | 3-bit Driver |
| Byte per column: | 16-bits |
| Packets Starts From: | Low |
| Load Latch On: | Rising |
| SGPMO Latch On: | Falling |
| SGPMI Latch On: | Rising |
| Display: | DataOut |
| Significant Bit: | LSB |

Channel

| | |
|---|---|
| Clock | A0 |
| Load | A1 |
| ☑ DO | A2 |
| ☑ DI | A3 |

Color

- Load / Start
- DO
- DI

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Channel:** Set the channel number of each signal on the object to be tested that is connected to the Logic Analyzer. These are Clock, Load, Data Out, and Data In. User can choose to have only Data Out, Data In, or both.

**Startup Settings:**

**Mode:**

1. **3-Bit Driver:**

2. **User Defined Data, t**he following settings are valid only when the mode is switched to this item**:**

   **Byte per Column:** Sets the Data-Size.

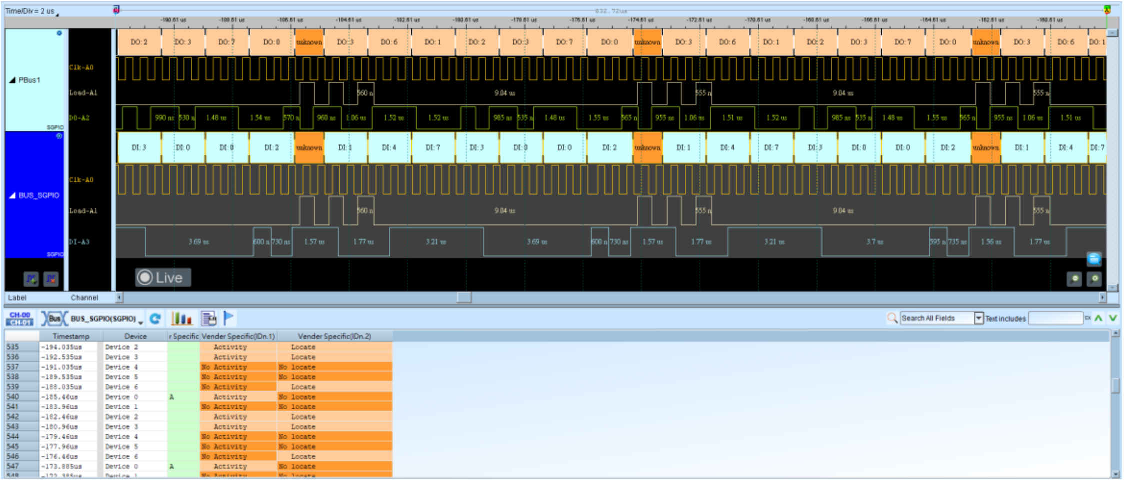
   **Packets Starts From:** Set the Packet to start from Low or High.

   **Load Latch On:** Latch data on rising or falling edge of Load Pin.

   **SGPMO Latch On:** Latch data on rising or falling edge of SGPMO Pin.

   **SGPMI Latch On:** Latch data on rising or falling edge of SGPMI Pin.

**Result:**

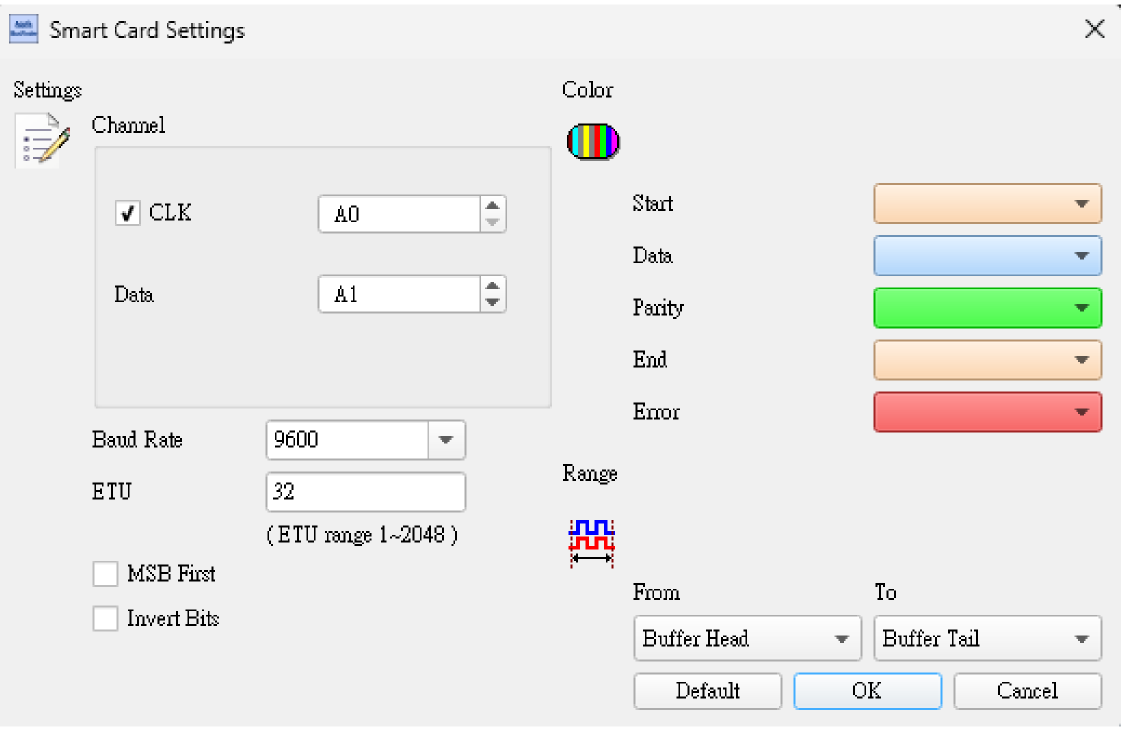


|  | Timestamp | Device | r Specific | Vender Specific(IDn.1) | Vender Specific(IDn.2) |
|---|---|---|---|---|---|
| 535 | -194.035us | Device 2 |  | Activity | Locate |
| 536 | -192.535us | Device 3 |  | Activity | Locate |
| 537 | -191.035us | Device 4 |  | No Activity | No locate |
| 538 | -189.535us | Device 5 |  | No Activity | No locate |
| 539 | -188.035us | Device 6 |  | No Activity | Locate |
| 540 | -185.46us | Device 0 | A | Activity | No locate |
| 541 | -183.96us | Device 1 |  | No Activity | No locate |
| 542 | -182.46us | Device 2 |  | Activity | Locate |
| 543 | -180.96us | Device 3 |  | Activity | Locate |
| 544 | -179.46us | Device 4 |  | No Activity | No locate |
| 545 | -177.96us | Device 5 |  | No Activity | No locate |
| 546 | -176.46us | Device 6 |  | No Activity | Locate |
| 547 | -173.885us | Device 0 | A | Activity | No locate |
| 548 | -172.385us | Device 1 |  | No Activity | No locate |

## Smart Card (ISO7816)

Smart Card is a communication protocol based on ISO 7816 specification, generally used in IC card or IC chip card, different IC chips have different functions and applications. It is mainly used for identification, recording and encoding/decoding.

### Settings



Smart Card Settings

Settings

Channel

☑ CLK A0

Data A1

Baud Rate 9600

ETU 32

( ETU range 1~2048 )

□ MSB First

□ Invert Bits

Color

Start 🟧

Data 🟦

Parity 🟩

End 🟧

Error 🟥

Range

From Buffer Head

To Buffer Tail

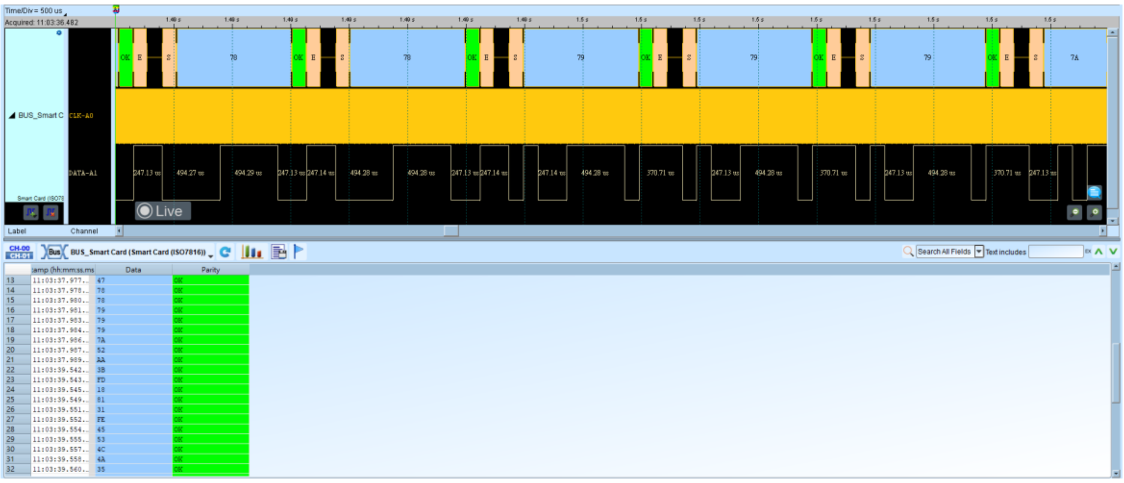
### Channel

**CLK:** Clock of transfer signal.

**Data:** Data of transfer signal.

**ETU(Elementary Time Unit):** The number of Clocks in each Bit.

## Result

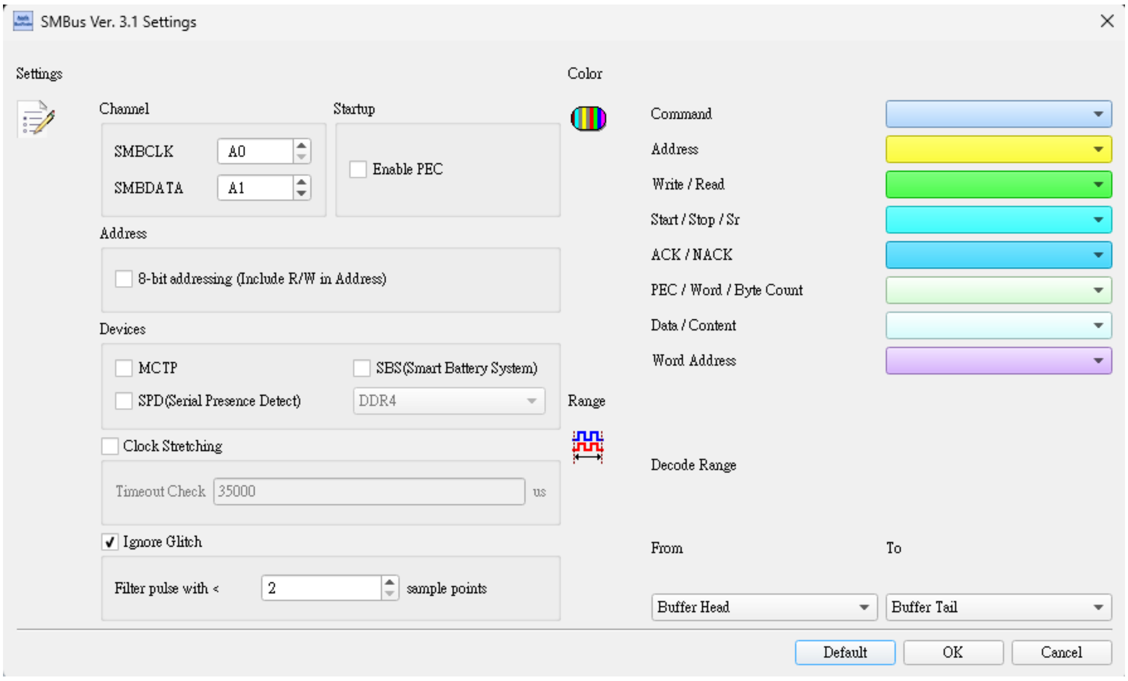


| | amp (hh:mm:ss.ms | Data | Parity |
|---|---|---|---|
| 13 | 11:03:37.977... | 47 | OK |
| 14 | 11:03:37.978... | 78 | OK |
| 15 | 11:03:37.980... | 78 | OK |
| 16 | 11:03:37.981... | 79 | OK |
| 17 | 11:03:37.983... | 79 | OK |
| 18 | 11:03:37.984... | 79 | OK |
| 19 | 11:03:37.986... | 7A | OK |
| 20 | 11:03:37.987... | 52 | OK |
| 21 | 11:03:37.989... | AA | OK |
| 22 | 11:03:39.542... | 3B | OK |
| 23 | 11:03:39.543... | FD | OK |
| 24 | 11:03:39.545... | 18 | OK |
| 25 | 11:03:39.549... | 81 | OK |
| 26 | 11:03:39.551... | 31 | OK |
| 27 | 11:03:39.552... | FE | OK |
| 28 | 11:03:39.554... | 45 | OK |
| 29 | 11:03:39.555... | 53 | OK |
| 30 | 11:03:39.557... | 4C | OK |
| 31 | 11:03:39.558... | 4A | OK |
| 32 | 11:03:39.560... | 35 | OK |

## SMBus

The full name System Management Bus (SMBus) is derived from I$^2$C bus, which is a bus consisting of two signals. SMBus was defined by Intel in 1995, and contains Clock, Data, and commands based on Philips' I²C serial bus protocol. The clock frequency ranges from 10KHz to 100KHz.

## Settings



SMBus Ver. 3.1 Settings

Settings

Channel

SMBCLK A0

SMBDATA A1

Startup

□ Enable PEC

Address

□ 8-bit addressing (Include R/W in Address)

Devices

□ MCTP

□ SBS(Smart Battery System)

□ SPD(Serial Presence Detect) DDR4

□ Clock Stretching

Timeout Check 35000 us

☑ Ignore Glitch

Filter pulse with < 2 sample points

Color

Command

Address

Write / Read

Start / Stop / Sr

ACK / NACK

PEC / Word / Byte Count

Data / Content

Word Address

Range

Decode Range

From Buffer Head

To Buffer Tail

### Channel:

**SMBCLK:** Clock of transfer signal.

**SMBDATA:** Data of transfer signal.

**Startup:** PEC analysis. Enabled when checked.

**8-bit addressing (Include R/W in Address):** Displays the 8-bit width address (7-bit width address plus 1-bit Rd/Wr). Enabled when checked.

**Device**, by default, the report window displays the SMBus analysis result. Enabled when checked**:**

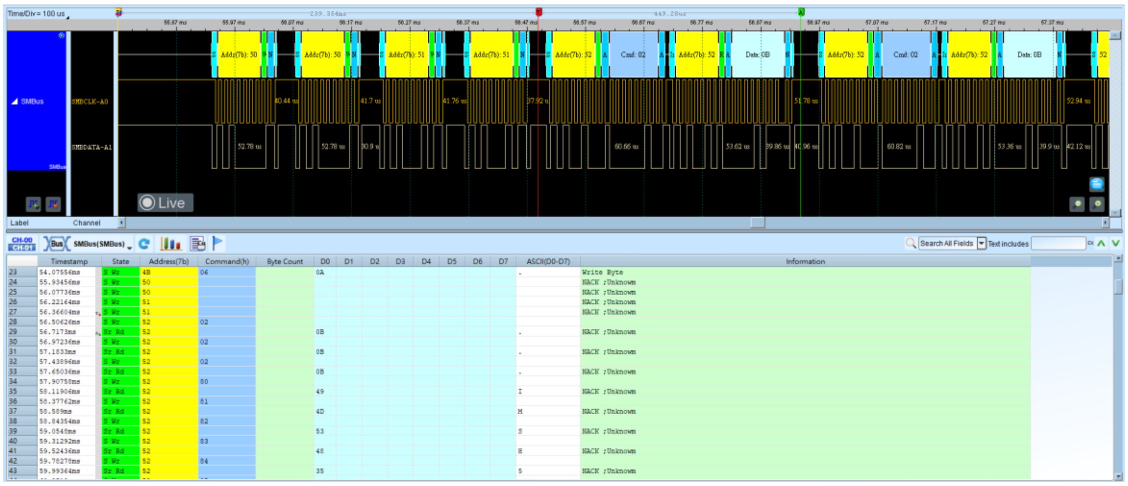
1. **MCTP:** Display MCTP analysis result in report area.
2. **Show SBS:** The report window displays the Smart Battery System analysis, which shows the status of the battery and information such as voltage, current, or manufacturer information.
3. **Show SPD(Serial Presence Detect):** The report window displays the EEPROM analysis content, which shows the configuration information of memory modules (DDR3, DDR2, DDR, SPD SDRAM), such as the number of P-Banks, voltage, the number of row addresses/column addresses, the bit-widths, and the timing of each major operation (e.g., CL, tRCD, tRP, tRAS, etc.).

**Clock Stretching:** Set the time for Clock Stretching. Enabled when checked.

**Ignore Glitch:** Ignore noise caused by slow transitions when analyzing. Enabled when checked.

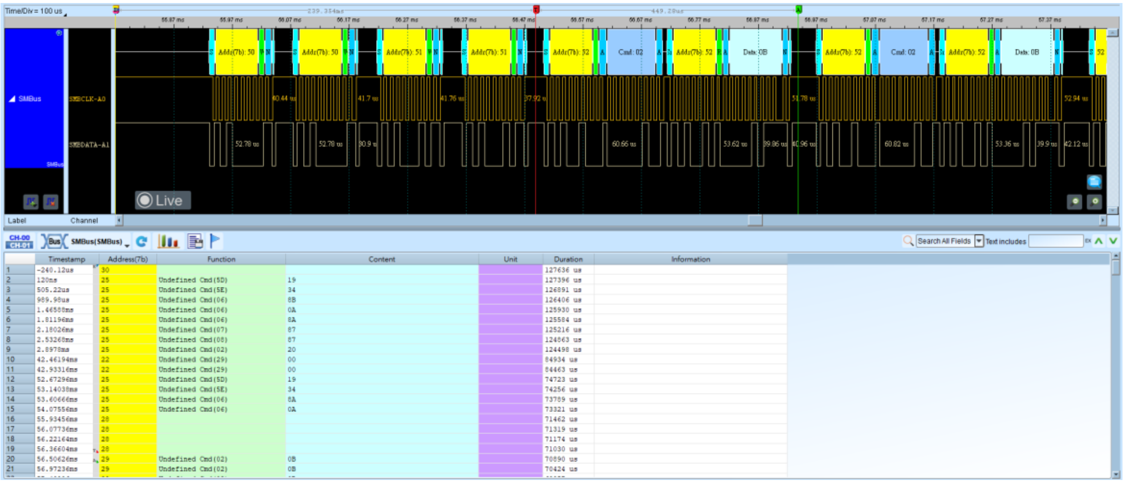
## Result

SMBus



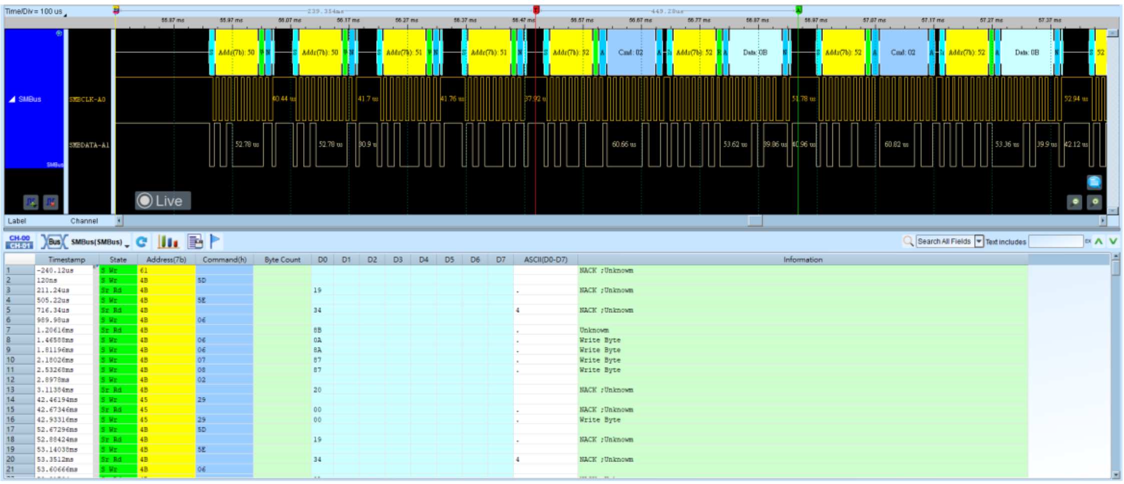
| | Timestamp | State | Address(7b) | Command(h) | Byte Count | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII(D0-D7) | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 23 | 54.0755óms | S Wr | 4B | 06 | | 0A | | | | | | | | . | Write Byte |
| 24 | 55.9345óms | S Wr | 50 | | | | | | | | | | | | NACK ;Unknown |
| 25 | 56.0773óms | S Wr | 50 | | | | | | | | | | | | NACK ;Unknown |
| 26 | 56.22164ms | S Wr | 51 | | | | | | | | | | | | NACK ;Unknown |
| 27 | 56.36604ms | S Wr | 51 | | | | | | | | | | | | NACK ;Unknown |
| 28 | 56.5062óms | S Wr | 52 | 02 | | | | | | | | | | | |
| 29 | 56.7173ms | Sr Rd | 52 | | | 0B | | | | | | | | . | NACK ;Unknown |
| 30 | 56.9723óms | S Wr | 52 | 02 | | | | | | | | | | | |
| 31 | 57.1833ms | Sr Rd | 52 | | | 0B | | | | | | | | . | NACK ;Unknown |
| 32 | 57.4389óms | S Wr | 52 | 02 | | | | | | | | | | | |
| 33 | 57.6503óms | Sr Rd | 52 | | | 0B | | | | | | | | . | NACK ;Unknown |
| 34 | 57.90758ms | S Wr | 52 | 80 | | | | | | | | | | | |
| 35 | 58.1190óms | Sr Rd | 52 | | | 49 | | | | | | | | I | NACK ;Unknown |
| 36 | 58.37762ms | S Wr | 52 | 81 | | | | | | | | | | | |
| 37 | 58.589ms | Sr Rd | 52 | | | 4D | | | | | | | | M | NACK ;Unknown |
| 38 | 58.84354ms | S Wr | 52 | 82 | | | | | | | | | | | |
| 39 | 59.0548ms | Sr Rd | 52 | | | 53 | | | | | | | | S | NACK ;Unknown |
| 40 | 59.31292ms | S Wr | 52 | 83 | | | | | | | | | | | |
| 41 | 59.5243óms | Sr Rd | 52 | | | 48 | | | | | | | | H | NACK ;Unknown |
| 42 | 59.78278ms | S Wr | 52 | 84 | | | | | | | | | | | |
| 43 | 59.99364ms | Sr Rd | 52 | | | 35 | | | | | | | | 5 | NACK ;Unknown |

## Show SBS (Smart Battery System)



| | Timestamp | Address(7b) | Function | Content | Unit | Duration | Information |
|---|---|---|---|---|---|---|---|
| 1 | -240.12us | 30 | | | | 127636 us | |
| 2 | 120ns | 25 | Undefined Cmd(5D) | 19 | | 127396 us | |
| 3 | 505.22us | 25 | Undefined Cmd(5E) | 34 | | 126891 us | |
| 4 | 989.98us | 25 | Undefined Cmd(06) | 8B | | 126406 us | |
| 5 | 1.46588ms | 25 | Undefined Cmd(06) | 0A | | 125930 us | |
| 6 | 1.81196ms | 25 | Undefined Cmd(06) | 8A | | 125584 us | |
| 7 | 2.18026ms | 25 | Undefined Cmd(07) | 87 | | 125216 us | |
| 8 | 2.53268ms | 25 | Undefined Cmd(08) | 87 | | 124863 us | |
| 9 | 2.8978ms | 25 | Undefined Cmd(02) | 20 | | 124498 us | |
| 10 | 42.46194ms | 22 | Undefined Cmd(29) | 00 | | 84934 us | |
| 11 | 42.93316ms | 22 | Undefined Cmd(29) | 00 | | 84463 us | |
| 12 | 52.67296ms | 25 | Undefined Cmd(5D) | 19 | | 74723 us | |
| 13 | 53.14038ms | 25 | Undefined Cmd(5E) | 34 | | 74256 us | |
| 14 | 53.60666ms | 25 | Undefined Cmd(06) | 8A | | 73789 us | |
| 15 | 54.07556ms | 25 | Undefined Cmd(06) | 0A | | 73321 us | |
| 16 | 55.93456ms | 28 | | | | 71462 us | |
| 17 | 56.07736ms | 28 | | | | 71319 us | |
| 18 | 56.22164ms | 28 | | | | 71174 us | |
| 19 | 56.36604ms | 28 | | | | 71030 us | |
| 20 | 56.50626ms | 29 | Undefined Cmd(02) | 0B | | 70890 us | |
| 21 | 56.97236ms | 29 | Undefined Cmd(02) | 0B | | 70424 us | |

## Show SPD (Serial Presence Detect)

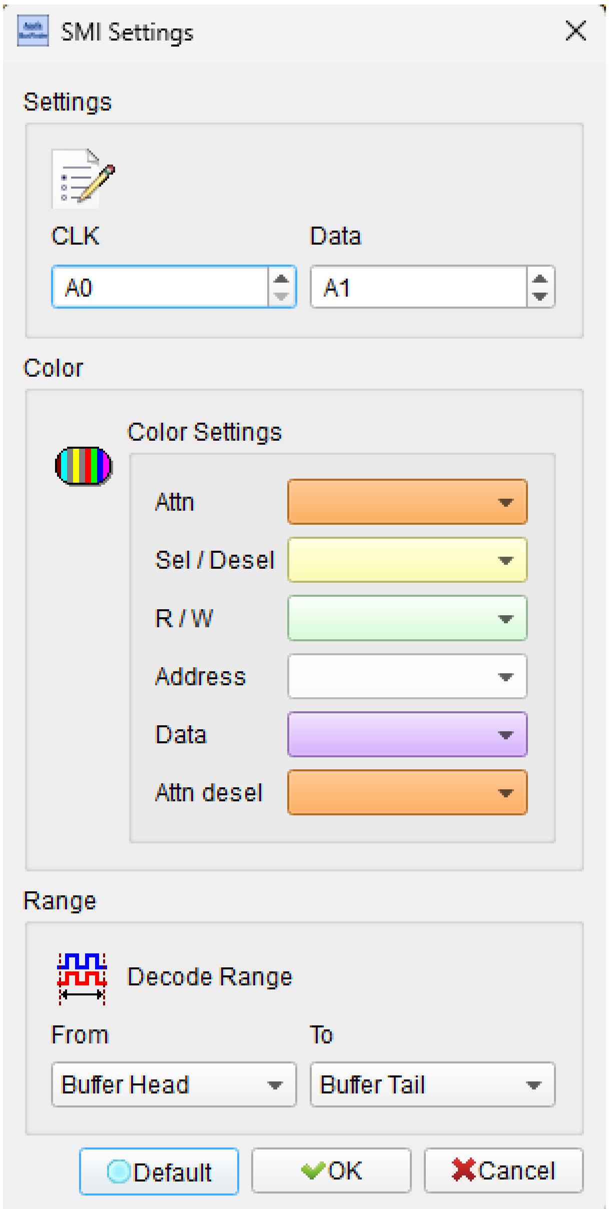


| | Timestamp | State | Address(7b) | Command(h) | Byte Count | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII(D0-D7) | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -240.12us | S Wr | 61 | | | | | | | | | | | | NACK ;Unknown |
| 2 | 120ns | S Wr | 4B | 5D | | | | | | | | | | | |
| 3 | 211.24us | Sr Rd | 4B | | | 19 | | | | | | | | . | NACK ;Unknown |
| 4 | 505.22us | S Wr | 4B | 5E | | | | | | | | | | | |
| 5 | 716.34us | Sr Rd | 4B | | | 34 | | | | | | | | 4 | NACK ;Unknown |
| 6 | 989.98us | S Wr | 4B | 06 | | | | | | | | | | | |
| 7 | 1.20616ms | Sr Rd | 4B | | | 8B | | | | | | | | . | Unknown |
| 8 | 1.46588ms | S Wr | 4B | 06 | | 0A | | | | | | | | . | Write Byte |
| 9 | 1.81196ms | S Wr | 4B | 06 | | 8A | | | | | | | | . | Write Byte |
| 10 | 2.18026ms | S Wr | 4B | 07 | | 87 | | | | | | | | . | Write Byte |
| 11 | 2.53268ms | S Wr | 4B | 08 | | 87 | | | | | | | | . | Write Byte |
| 12 | 2.8978ms | S Wr | 4B | 02 | | | | | | | | | | | |
| 13 | 3.11384ms | Sr Rd | 4B | | | 20 | | | | | | | | | NACK ;Unknown |
| 14 | 42.46194ms | S Wr | 45 | 29 | | | | | | | | | | | |
| 15 | 42.67346ms | Sr Rd | 45 | | | 00 | | | | | | | | . | NACK ;Unknown |
| 16 | 42.93316ms | S Wr | 45 | 29 | | 00 | | | | | | | | . | Write Byte |
| 17 | 52.67296ms | S Wr | 4B | 5D | | | | | | | | | | | |
| 18 | 52.88424ms | Sr Rd | 4B | | | 19 | | | | | | | | . | NACK ;Unknown |
| 19 | 53.14038ms | S Wr | 4B | 5E | | | | | | | | | | | |
| 20 | 53.3512ms | Sr Rd | 4B | | | 34 | | | | | | | | 4 | NACK ;Unknown |
| 21 | 53.60666ms | S Wr | 4B | 06 | | | | | | | | | | | |

## SMI

SMI (Serial Microprocessor Interface) was developed by BDNC and consists of a Clock and Data.

### Settings



SMI Settings

Settings

CLK: A0

Data: A1

Color

Color Settings

Attn: 🟧

Sel / Desel: 🟨

R / W: 🟩

Address: ⬜

Data: 🟪

Attn desel: 🟧

Range

Decode Range

From: Buffer Head

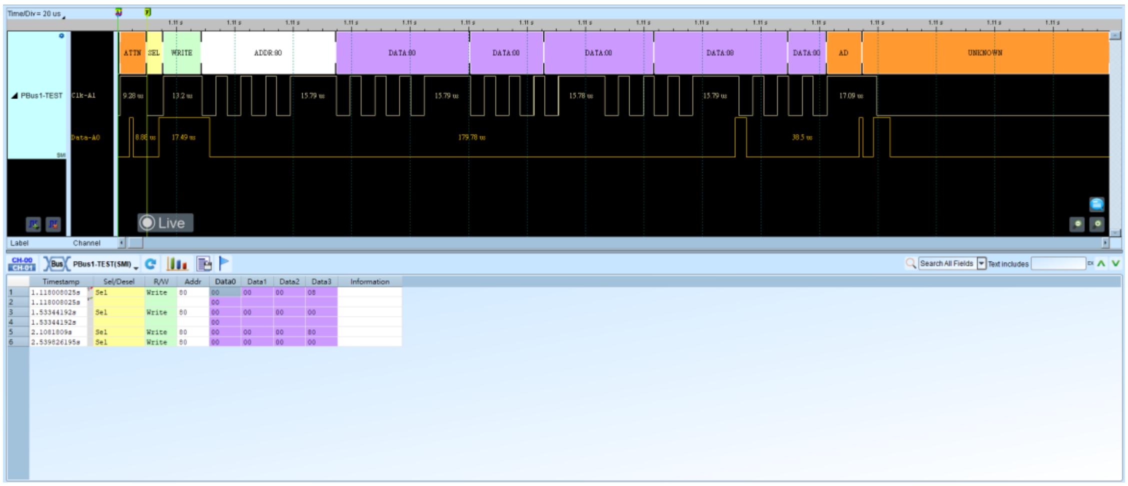
To: Buffer Tail

**Channel:**

**CLK:** Clock of transfer signal.

**Data:** Data of transfer signal.

## Result

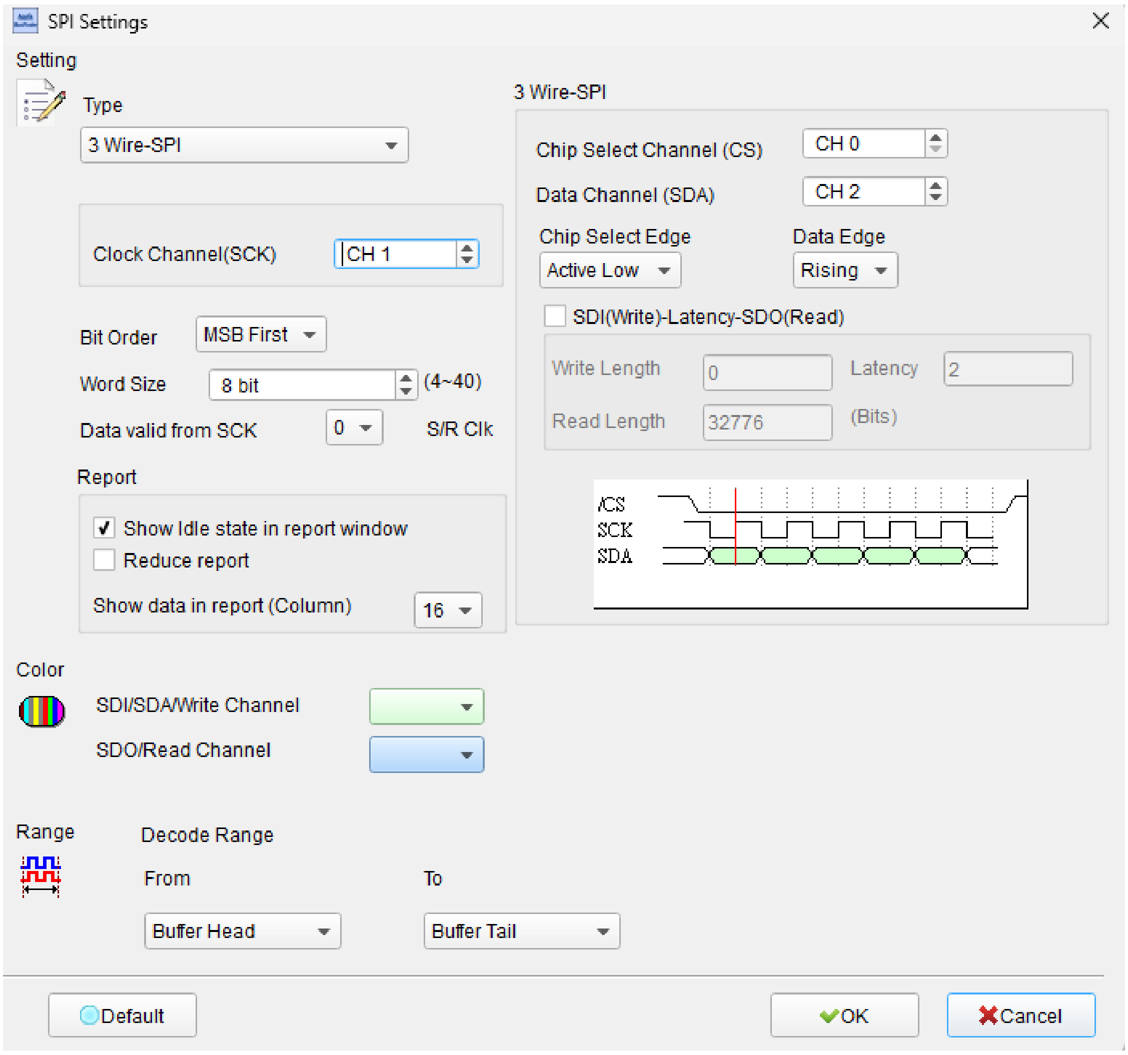


| | Timestamp | Sel/Desel | R/W | Addr | Data0 | Data1 | Data2 | Data3 | Information |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1.118008025s | Sel | Write | 80 | 00 | 00 | 00 | 08 | |
| 2 | 1.118008025s | | | | 00 | | | | |
| 3 | 1.53344192s | Sel | Write | 80 | 00 | 00 | 00 | 00 | |
| 4 | 1.53344192s | | | | 00 | | | | |
| 5 | 2.1081809s | Sel | Write | 80 | 00 | 00 | 00 | 80 | |
| 6 | 2.539826195s | Sel | Write | 80 | 00 | 00 | 00 | 00 | |

## SPI

The Serial Peripheral Interface Bus (SPI) is a 4-wire synchronous sequential data protocol for portable device platform systems. Serial peripheral interfaces are generally 4-wire, and can sometimes be 3-wire or 2-wire.

## Settings



SPI Settings

Setting

Type: 3 Wire-SPI

Clock Channel(SCK): CH 1

Bit Order: MSB First

Word Size: 8 bit (4~40)

Data valid from SCK: 0 S/R Clk

Report

☑ Show Idle state in report window

□ Reduce report

Show data in report (Column): 16

3 Wire-SPI

Chip Select Channel (CS): CH 0

Data Channel (SDA): CH 2

Chip Select Edge: Active Low

Data Edge: Rising

□ SDI(Write)-Latency-SDO(Read)

Write Length: 0 Latency: 2

Read Length: 32776 (Bits)

Color

SDI/SDA/Write Channel: 🟩

SDO/Read Channel: 🟦

Range

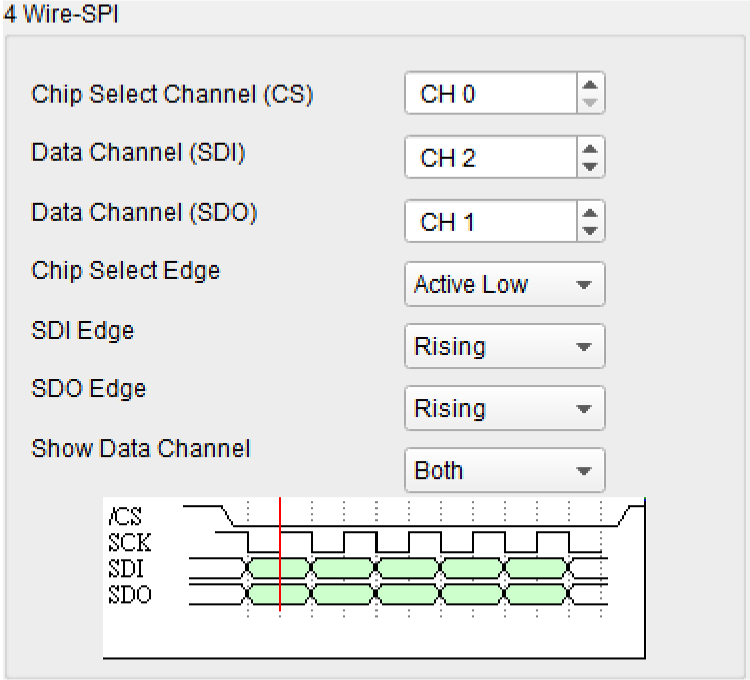
Decode Range

From: Buffer Head To: Buffer Tail

**Type:** Select the SPI type, the default is 3-wire-SPI, which is included:

**4 Wire-SPI dialog box ➔CS, SCK, SDI, SDO**

User can set the trigger edge of CS, SDI and SDO respectively, the default setting of CS is Active Low, and the default setting of SDI/SDO is Active High, because the SDI and SDO data will be displayed at the same time. User can select the last data to be displayed in the display data channel as SDI only, SDO only or both, and the default is Both.



4 Wire-SPI

Chip Select Channel (CS): CH 0

Data Channel (SDI): CH 2

Data Channel (SDO): CH 1

Chip Select Edge: Active Low

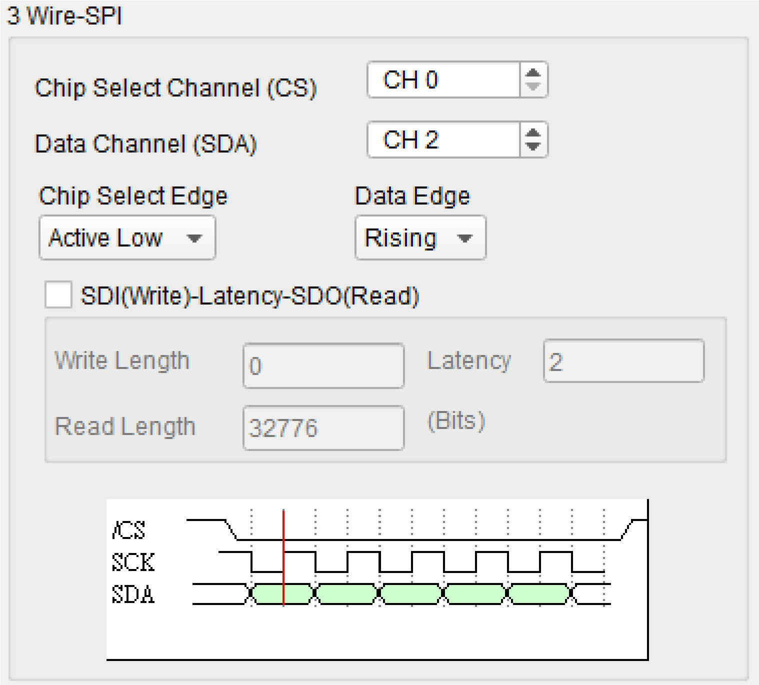
SDI Edge: Rising

SDO Edge: Rising

Show Data Channel: Both

**3 Wire-SPI dialog box ➔ CS, SCK, SDA**

In 3-wire Slave select mode, only 1 data channel (either SDI or SDO) is required. User can set the triggering edge of CS and Data respectively, the default setting of CS is Active Low, and the default setting of Data is Active High, for general application, the data channel is a single line and unidirectional way to transfer data.



3 Wire-SPI

Chip Select Channel (CS): CH 0

Data Channel (SDA): CH 2

Chip Select Edge: Active Low

Data Edge: Rising

□ SDI(Write)-Latency-SDO(Read)

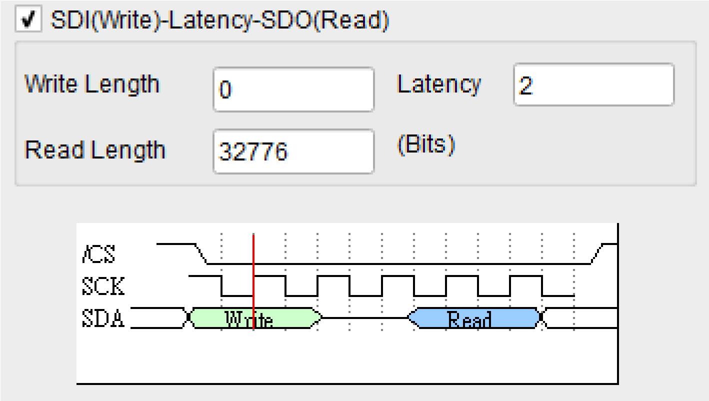
Write Length: 0

Latency: 2

Read Length: 32776

(Bits)

We also provide a one-line bidirectional transmission mode. As shown in the figure below



☑ SDI(Write)-Latency-SDO(Read)

Write Length: 0

Latency: 2

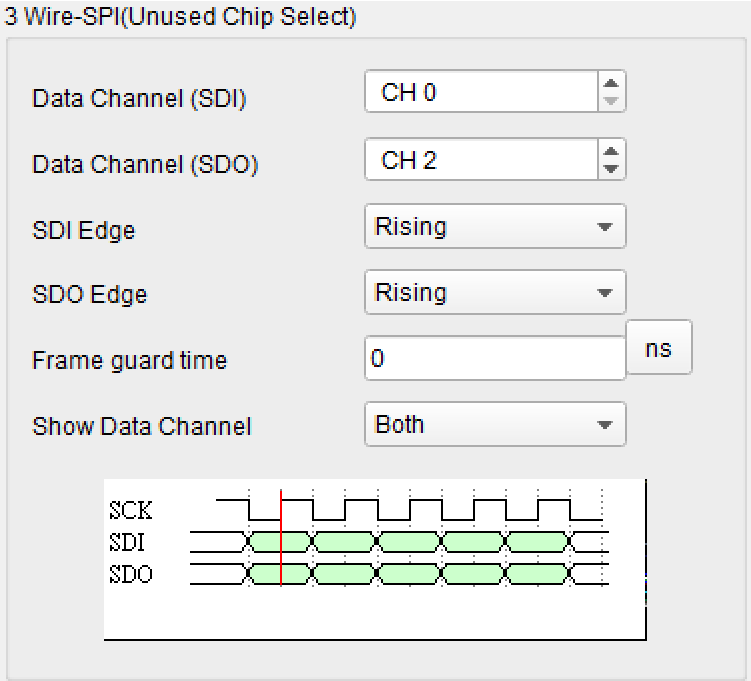
Read Length: 32776

(Bits)

User only need to check “SDI(Write)-Waiting-SDO(Read)”, then user can set the bit number of bidirectional transmission. Let's take Master as our point of view, the write length is the number of bits that Master puts into the data channel, the minimum is 1. The number of bits that wait for Slave to process is the minimum is 0. Then the data is collected according to the read length, the minimum is 1. The maximum value of these 3 parameters is 65535.

**3 Wire-SPI (Unused Chip Slave) dialog box ➔ SCK, SDI, SDO**

Since CS is not used, user has to set the Idle time of SCK as the frame separating time. In 3-wire without Slave select mode, user need to set the channel where SDI/SDO is located and its trigger edge to Active High. In the 3-wire non-slave select mode, user need to set the channel where SDI/SDO is located, and set the trigger edge of SDI/SDO to Active High, and set the time to wait for Clock Idle to separate the frames. The SDI and SDO data will appear at the same time. User can select the last data to be displayed in SDI only, SDO only or both in the display data channel, and the default is Both.

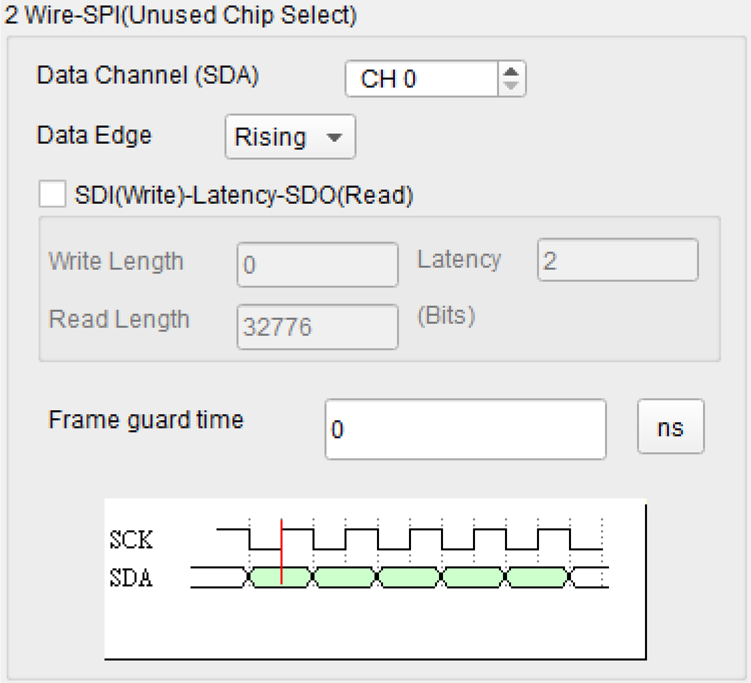


3 Wire-SPI(Unused Chip Select)

Data Channel (SDI): CH 0
Data Channel (SDO): CH 2
SDI Edge: Rising
SDO Edge: Rising
Frame guard time: 0 ns
Show Data Channel: Both

**2 Wire-SPI (Unused Chip Slave) dialog box → SCK, SDA**

Since CS is not used, user has to set the Idle time of SCK as the frame separating time. In 2-wire without Slave select mode, user need to set the channel where the data is located and its trigger edge to Active High. The trigger edge of the data channel and the trigger edge of the data channel should be set to Active High, and set the time to wait for the Clock Idle as the Frame Separation. In general application, the data channel is a single line and unidirectional way to transfer the data.



2 Wire-SPI(Unused Chip Select)

Data Channel (SDA) CH 0

Data Edge Rising

□ SDI(Write)-Latency-SDO(Read)

Write Length 0

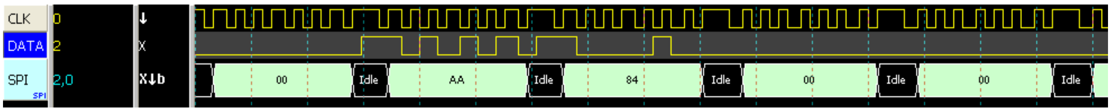
Latency 2

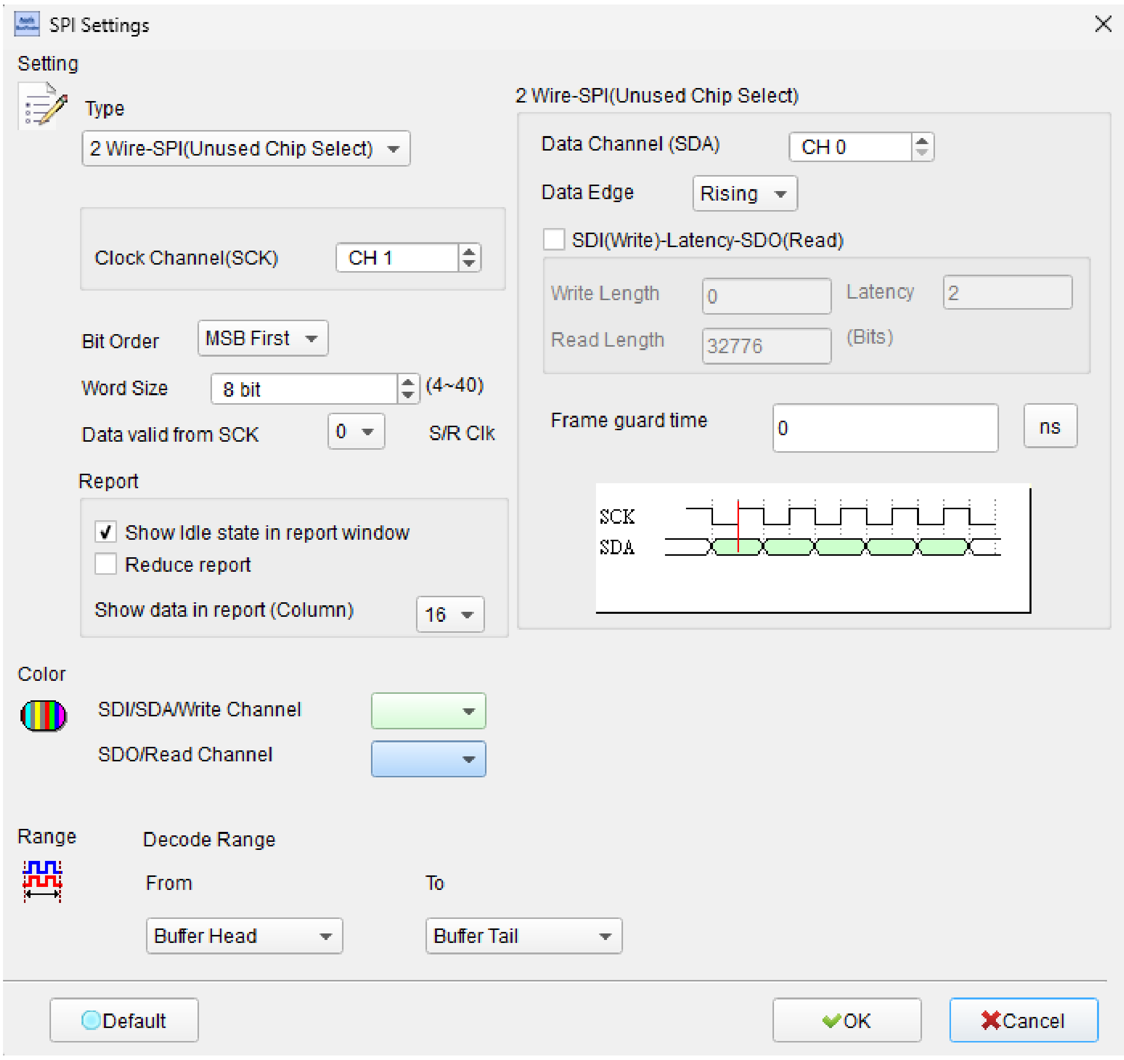
Read Length 32776

(Bits)

Frame guard time 0 ns

If Slave select is not used, and the time between frames is not 0, the application example is as follows. The signals are only CLK, DATA, Frame is separated by 6 us, and the data trigger edge is Rising. it can be seen that if the Clock pause interval is more than 6 us, it will be recognized as Idle.





SPI Settings

Setting

Type: 2 Wire-SPI(Unused Chip Select)

Clock Channel(SCK): CH 1

Bit Order: MSB First

Word Size: 8 bit (4~40)

Data valid from SCK: 0 S/R Clk

Report

☑ Show Idle state in report window

□ Reduce report

Show data in report (Column): 16

2 Wire-SPI(Unused Chip Select)

Data Channel (SDA): CH 0

Data Edge: Rising

□ SDI(Write)-Latency-SDO(Read)

Write Length: 0 Latency: 2

Read Length: 32776 (Bits)

Frame guard time: 0 ns

Color

SDI/SDA/Write Channel: 🟩

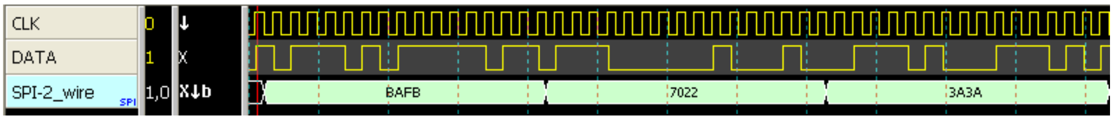
SDO/Read Channel: 🟦

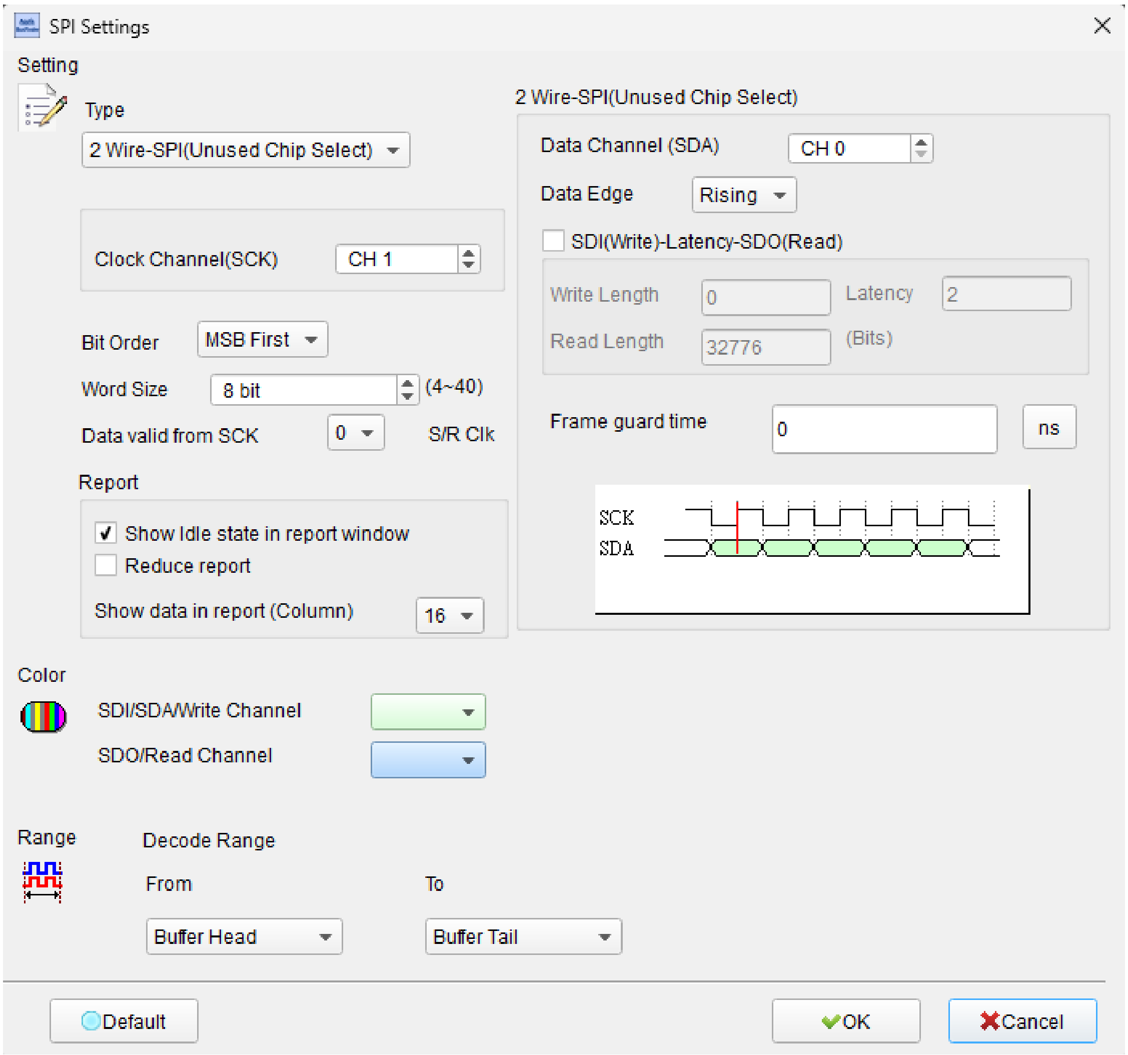
Range

Decode Range

From: Buffer Head To: Buffer Tail

When Slave select is not used, and the Frame separation time is 0, it can be another kind of continuous data analysis, as shown in the following figure. The signal is only CLK, DATA, and the time between frames is 0, the data trigger is in Falling.





SPI Settings

Setting

Type: 2 Wire-SPI(Unused Chip Select)

Clock Channel(SCK): CH 1

Bit Order: MSB First

Word Size: 8 bit (4~40)

Data valid from SCK: 0 S/R Clk

Report

☑ Show Idle state in report window

□ Reduce report

Show data in report (Column): 16

2 Wire-SPI(Unused Chip Select)

Data Channel (SDA): CH 0

Data Edge: Rising

□ SDI(Write)-Latency-SDO(Read)

Write Length: 0 Latency: 2

Read Length: 32776 (Bits)

Frame guard time: 0 ns

Color

SDI/SDA/Write Channel: 🟩

SDO/Read Channel: 🟦

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Bit Order:** You can configure the SPI data interpretation as either MSB first or LSB first. The default setting is LSB first.

**Word Size:** You can set the data word size in bits. During SPI analysis, this value will define the number of bits for each data word. The minimum value is 4, and the maximum value is 40. The default value is 8.

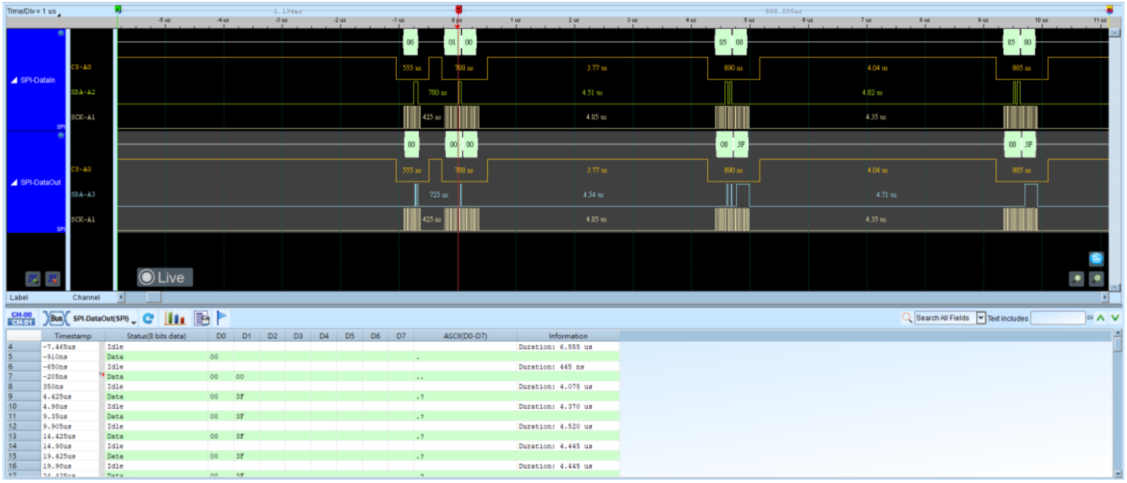
**Report:**

**Show Idle state in report window:** In SPI applications, there may be intervals with the Idle State between each data capture. To make data review easier, you can configure the report window to hide the Idle State. By default, the Idle State is shown.

**Reduce report:** You can configure consecutive SPI data to be displayed in the report window in a format ranging from 1 to 16 columns. The default setting is 16 columns, and the ASCII-encoded results can be viewed on the far right side of the report window.

**Data Valid from SCK:** In certain devices using SPI transmission, there is a delay between the data output and the valid data, which does not align with the clock's edge. To accommodate such devices, you can configure the "Data Valid from SCK" to delay this time. You can input the delay time in terms of the sampling rate, with a range of 0-3. By default, there is no delay. If set to 1, with a sampling rate of 200 MHz, the actual delay time will be 5 ns.

**Result**

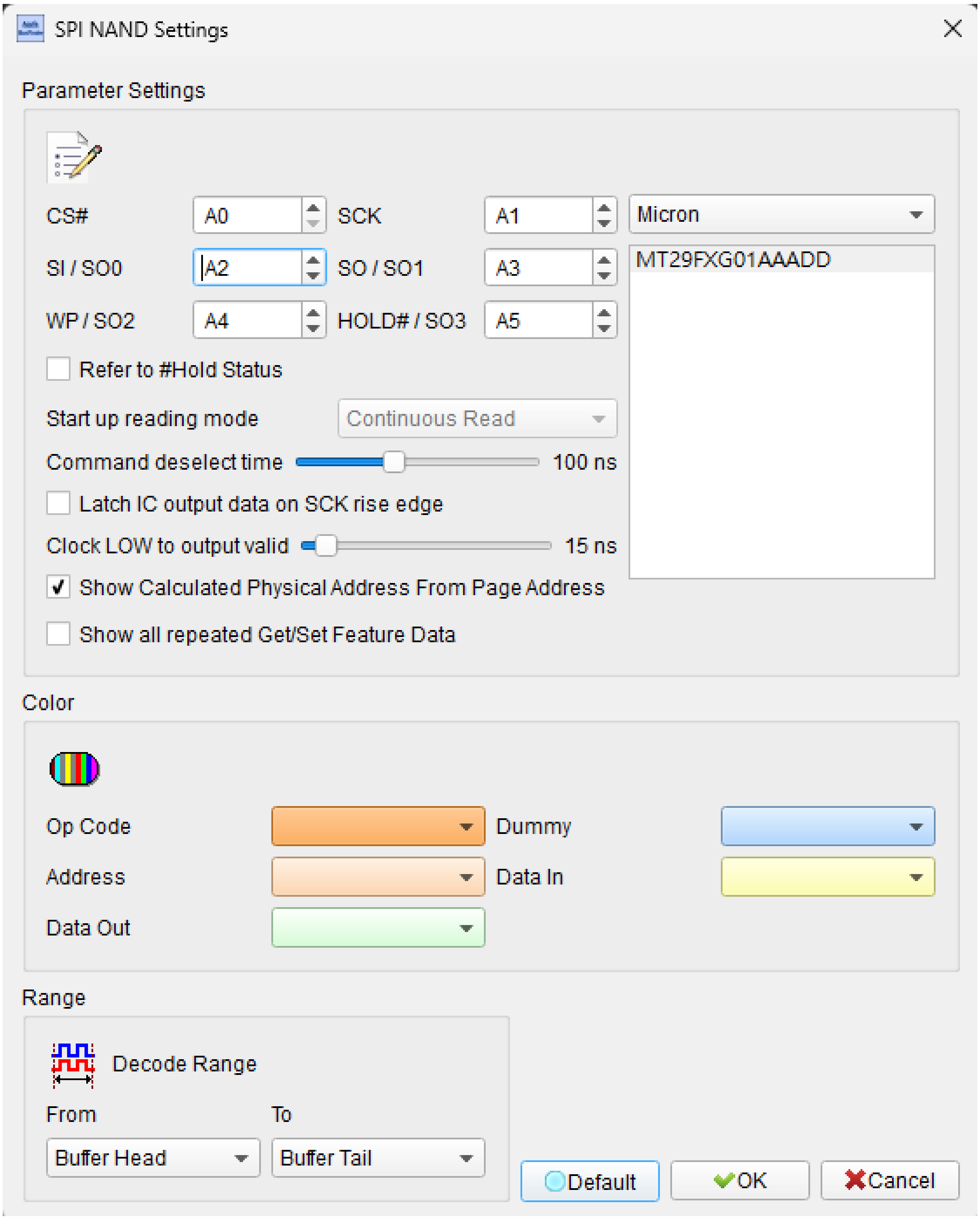


| | Timestamp | Status(8 bits data) | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII(D0-D7) | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | -7.465us | Idle | | | | | | | | | | Duration: 6.555 us |
| 5 | -910ns | Data | 00 | | | | | | | | . | |
| 6 | -650ns | Idle | | | | | | | | | | Duration: 445 ns |
| 7 | -205ns | Data | 00 | 00 | | | | | | | .. | |
| 8 | 350ns | Idle | | | | | | | | | | Duration: 4.075 us |
| 9 | 4.425us | Data | 00 | 3F | | | | | | | .? | |
| 10 | 4.98us | Idle | | | | | | | | | | Duration: 4.370 us |
| 11 | 9.35us | Data | 00 | 3F | | | | | | | .? | |
| 12 | 9.905us | Idle | | | | | | | | | | Duration: 4.520 us |
| 13 | 14.425us | Data | 00 | 3F | | | | | | | .? | |
| 14 | 14.98us | Idle | | | | | | | | | | Duration: 4.445 us |
| 15 | 19.425us | Data | 00 | 3F | | | | | | | .? | |
| 16 | 19.98us | Idle | | | | | | | | | | Duration: 4.445 us |
| 17 | 24.425us | Data | 00 | 3F | | | | | | | .? | |

## SPI NAND

The SPI NAND Flash Memory series uses the SPI/QPI transmission protocol for data communication. The SPI NAND bus analysis allows users to view both command and input/output bus information simultaneously, saving time when using the SPI bus analysis waveform.

## Settings



SPI NAND Settings

Parameter Settings

CS# A0 SCK A1 Micron

SI / SO0 A2 SO / SO1 A3 MT29FXG01AAADD

WP / SO2 A4 HOLD# / SO3 A5

□ Refer to #Hold Status

Start up reading mode: Continuous Read

Command deselect time: 100 ns

□ Latch IC output data on SCK rise edge

Clock LOW to output valid: 15 ns

☑ Show Calculated Physical Address From Page Address

□ Show all repeated Get/Set Feature Data

Color

Op Code, Dummy, Address, Data In, Data Out

Range

Decode Range

From: Buffer Head To: Buffer Tail

**CS#:** Chip Select of transfered singal.

**SCLK:** Clock of transferred signal.

**SIO0 – SIO3:** Data pins for data transfer.

**Start up reading mode:** Enables selection of the read state during initial analysis.

**Command deselect time:** Adjustable hold time for determining CS# inactivity during analysis.

**Clock LOW to output valid：** Adjustable setting for determining the actual data position during analysis.

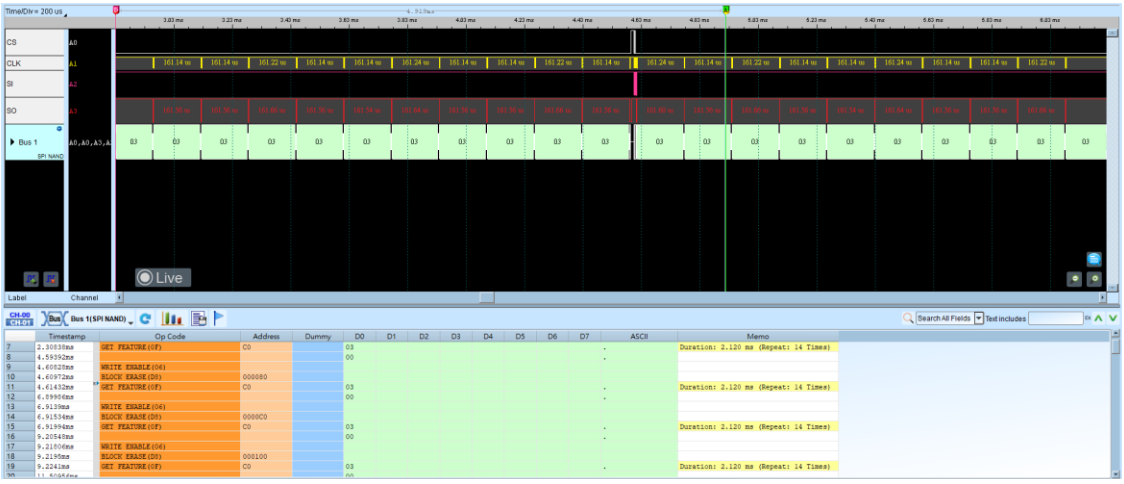
**Refer to #Hold Status:** Decode based on the state of the #Hold pin. Enabled when checked.

**Latch IC output data on SCK rise edge:** Latch Data on the rising edge of SCK for analysis. Enabled when checked.

**Show Calculated Physical Address From Page Address:** Display the full address. Enabled when checked.

**Show all repeated Get/Set Feature Data:** Fully display repeated Get or Set Feature data. Enabled when checked.

## Result



| | Timestamp | Op Code | Address | Dummy | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII | Memo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7 | 2.30838ms | GET FEATURE(0F) | C0 | | 03 | | | | | | | | . | Duration: 2.120 ms (Repeat: 14 Times) |
| 8 | 4.59392ms | | | | 00 | | | | | | | | . | |
| 9 | 4.60828ms | WRITE ENABLE(06) | | | | | | | | | | | | |
| 10 | 4.60972ms | BLOCK ERASE(D8) | 000080 | | | | | | | | | | | |
| 11 | 4.61432ms | GET FEATURE(0F) | C0 | | 03 | | | | | | | | . | Duration: 2.120 ms (Repeat: 14 Times) |
| 12 | 6.89986ms | | | | 00 | | | | | | | | . | |
| 13 | 6.9139ms | WRITE ENABLE(06) | | | | | | | | | | | | |
| 14 | 6.91534ms | BLOCK ERASE(D8) | 0000C0 | | | | | | | | | | | |
| 15 | 6.91994ms | GET FEATURE(0F) | C0 | | 03 | | | | | | | | . | Duration: 2.120 ms (Repeat: 14 Times) |
| 16 | 9.20548ms | | | | 00 | | | | | | | | . | |
| 17 | 9.21806ms | WRITE ENABLE(06) | | | | | | | | | | | | |
| 18 | 9.2195ms | BLOCK ERASE(D8) | 000100 | | | | | | | | | | | |
| 19 | 9.2241ms | GET FEATURE(0F) | C0 | | 03 | | | | | | | | . | Duration: 2.120 ms (Repeat: 14 Times) |
| 20 | 11.50956ms | | | | 00 | | | | | | | | | |

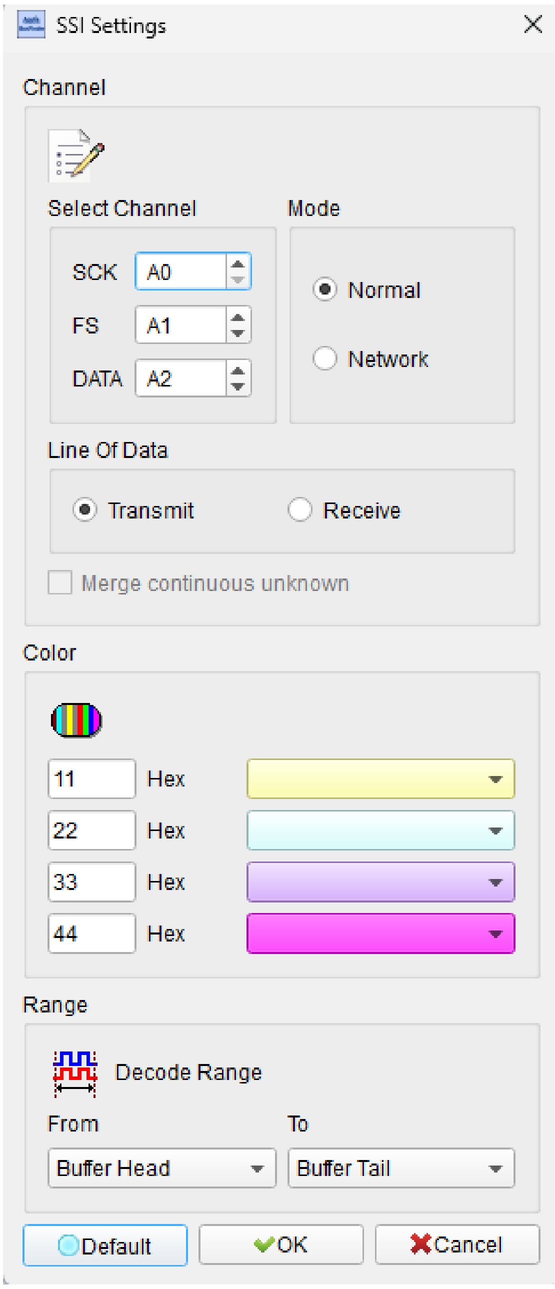
The report will delete the repeated value while Op Code = Get Feature (0F).

And mark its duration time。

## SSI

The Synchronous Serial Interface (SSI) protocol has four kinds of signals: Serial Clock (SCK), Transmit Data, Receive Data and Transmit/Receive Frame Synchronous (FS). The SSI protocol supports either the Normal or Network mode that is independent of whether the transmitter and the receiver are synchronous or asynchronous.

### Settings



SSI Settings

Channel

Select Channel

SCK A0

FS A1

DATA A2

Mode

● Normal

○ Network

Line Of Data

● Transmit ○ Receive

□ Merge continuous unknown

Color

| | | |
|---|---|---|
| 11 | Hex | 🟨 |
| 22 | Hex | 🟦 |
| 33 | Hex | 🟪 |
| 44 | Hex | 🟪 |

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Select Channel:** Assign the signal pins of the Device Under Test (DUT) to the corresponding channel numbers of the logic analyzer

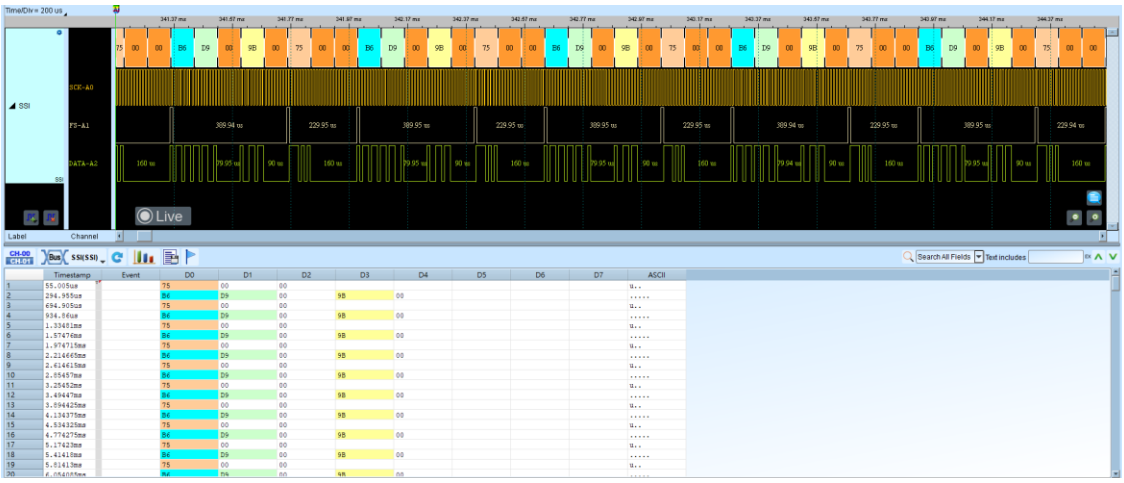
**Mode:** Normal or Network.

**Line Of Data:** Transmit or Receive.

**Merge continuous unknown:** Combine the unknown data only in Network mode.

## Result

Click **OK** to run SSI decode and see the result on the Report Window below.

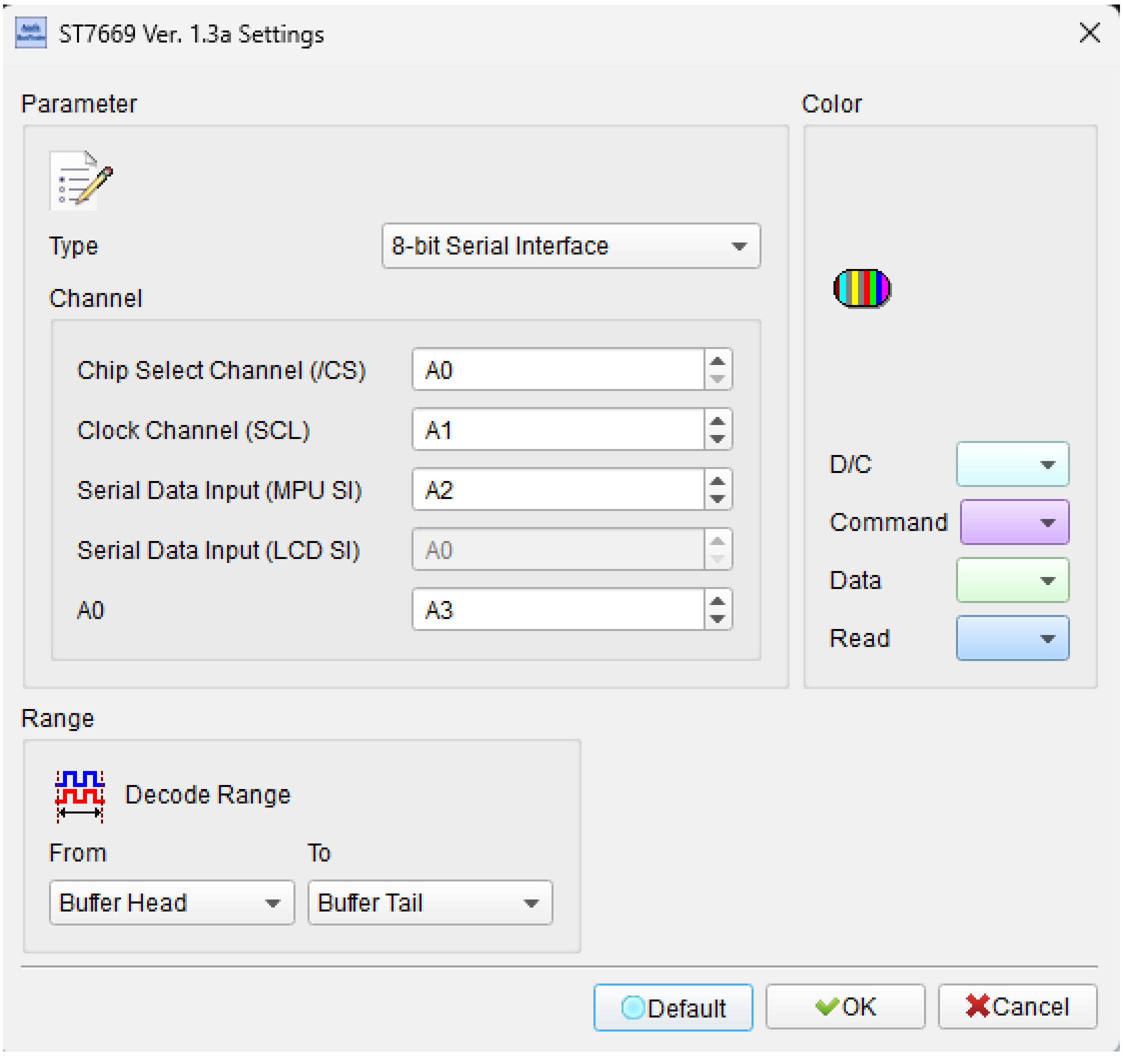


| | Timestamp | Event | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 55.005us | | 75 | 00 | 00 | | | | | | u.. |
| 2 | 294.955us | | B6 | D9 | 00 | 9B | 00 | | | | ..... |
| 3 | 694.905us | | 75 | 00 | 00 | | | | | | u.. |
| 4 | 934.86us | | B6 | D9 | 00 | 9B | 00 | | | | ..... |
| 5 | 1.33481ms | | 75 | 00 | 00 | | | | | | u.. |
| 6 | 1.57476ms | | B6 | D9 | 00 | 9B | 00 | | | | ..... |
| 7 | 1.974715ms | | 75 | 00 | 00 | | | | | | u.. |
| 8 | 2.214665ms | | B6 | D9 | 00 | 9B | 00 | | | | ..... |
| 9 | 2.614615ms | | 75 | 00 | 00 | | | | | | u.. |
| 10 | 2.85457ms | | B6 | D9 | 00 | 9B | 00 | | | | ..... |
| 11 | 3.25452ms | | 75 | 00 | 00 | | | | | | u.. |
| 12 | 3.49447ms | | B6 | D9 | 00 | 9B | 00 | | | | ..... |
| 13 | 3.894425ms | | 75 | 00 | 00 | | | | | | u.. |
| 14 | 4.134375ms | | B6 | D9 | 00 | 9B | 00 | | | | ..... |
| 15 | 4.534325ms | | 75 | 00 | 00 | | | | | | u.. |
| 16 | 4.774275ms | | B6 | D9 | 00 | 9B | 00 | | | | ..... |
| 17 | 5.17423ms | | 75 | 00 | 00 | | | | | | u.. |
| 18 | 5.41418ms | | B6 | D9 | 00 | 9B | 00 | | | | ..... |
| 19 | 5.81413ms | | 75 | 00 | 00 | | | | | | u.. |
| 20 | 6.054085ms | | B6 | D9 | 00 | 9B | 00 | | | | ..... |

## ST7669

ST76669 was developed by Sitronix to interface with the LCD module.

## Settings



ST7669 Ver. 1.3a Settings

Parameter

Type: 8-bit Serial Interface

Channel

Chip Select Channel (/CS): A0

Clock Channel (SCL): A1

Serial Data Input (MPU SI): A2

Serial Data Input (LCD SI): A0

A0: A3

Color

D/C

Command

Data

Read

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**Type, selectable options:**

1. 8-bit Serial Interface
2. 8-bit Serial Interface + LCD SI
3. 9-bit Serial Interface
4. 9-bit Serial Interface + LCD SI

**Chip Select Channel (/CS):** CS for ST7669 data transmission.

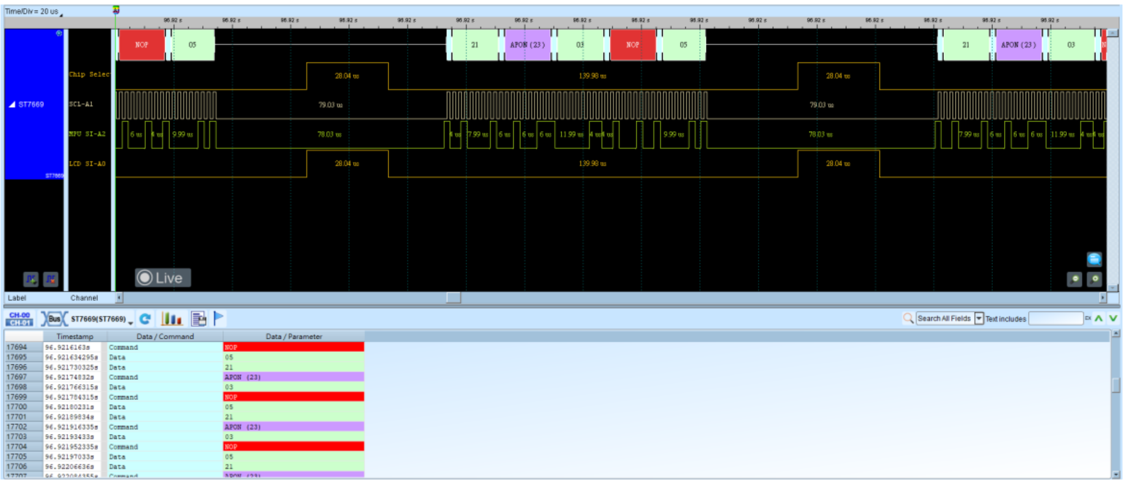
**Clock Channel (SCL)**: Clock for ST7669 data transmission.

**Serial Data Input (MPU SI)**: MPU Data Input for ST7669 data transmission.

**Serial Data Input (LCD SI)**: LCD Data Input for ST7669 data transmission.

**A0**: A0 for ST7669 data transmission.

## Result

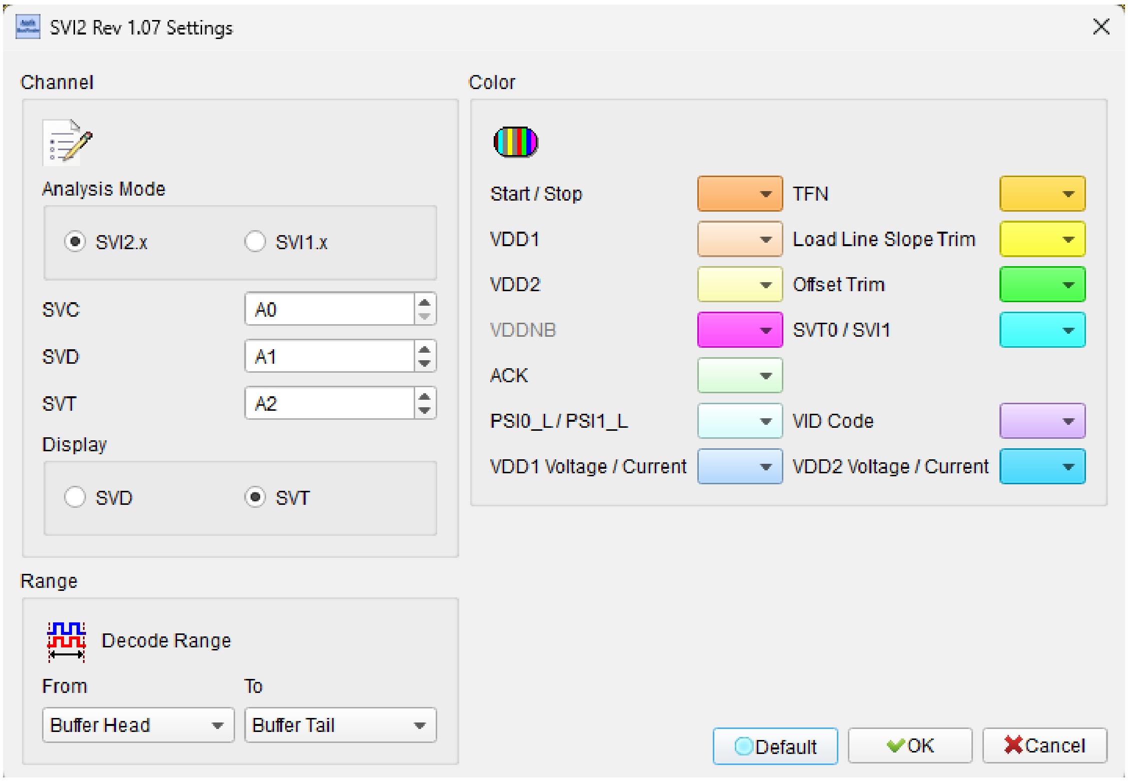


| | Timestamp | Data / Command | Data / Parameter |
|---|---|---|---|
| 17694 | 96.9216163s | Command | NOP |
| 17695 | 96.921634295s | Data | 05 |
| 17696 | 96.921730325s | Data | 21 |
| 17697 | 96.92174832s | Command | APON (23) |
| 17698 | 96.921766315s | Data | 03 |
| 17699 | 96.921784315s | Command | NOP |
| 17700 | 96.92180231s | Data | 05 |
| 17701 | 96.92189834s | Data | 21 |
| 17702 | 96.921916335s | Command | APON (23) |
| 17703 | 96.92193433s | Data | 03 |
| 17704 | 96.921952335s | Command | NOP |
| 17705 | 96.92197033s | Data | 05 |
| 17706 | 96.92206636s | Data | 21 |
| 17707 | 96.922084355s | Command | APON (23) |

## SVI2

The SVI2 (Serial VID Interface 2.0) bus is a communication protocol used by AMD for power management control data transmission, typically applied in voltage control. SVI2 bus analysis allows users to view transmitted signal packets, reducing the time needed for waveform interpretation. It operates with a voltage range of 1V to 1.8V and a maximum frequency of 20MHz. The interface consists of three signal channels: SVC, SVD, and SVT. When measuring signals, it is important to set the trigger level between 0.6V and 0.9V to ensure stable signal triggering.

## Settings



SVI2 Rev 1.07 Settings

Channel

Analysis Mode

● SVI2.x ○ SVI1.x

SVC: A0

SVD: A1

SVT: A2

Display

○ SVD ● SVT

Range

Decode Range

From: Buffer Head

To: Buffer Tail

Color

Start / Stop

TFN

VDD1

Load Line Slope Trim

VDD2

Offset Trim

VDDNB

SVT0 / SVI1

ACK

PSI0_L / PSI1_L

VID Code

VDD1 Voltage / Current

VDD2 Voltage / Current

**Analysis Mode:**

**SVI2.x / SVI1.x**: Select SVI2 / SVI decoding.

**Channel:**

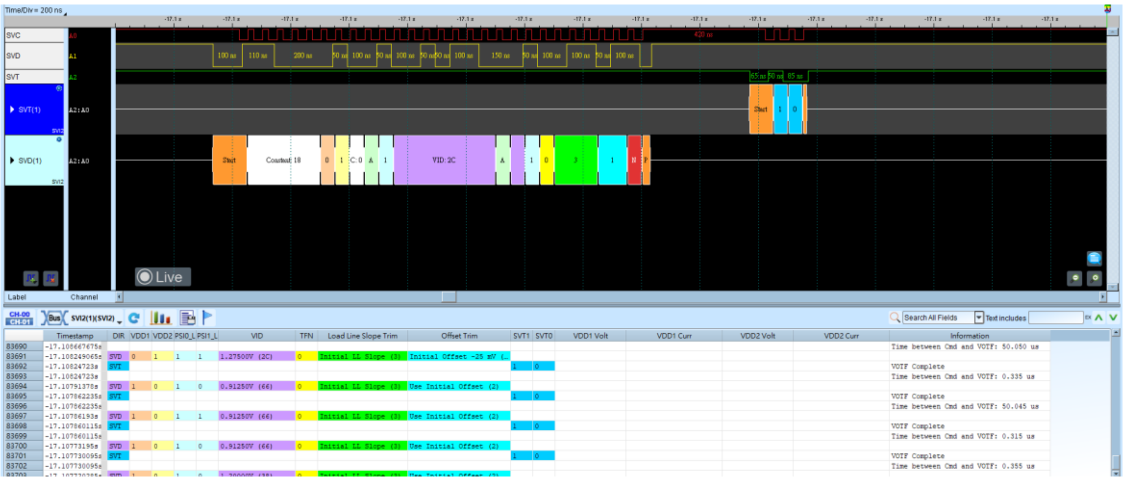
**SVC:** Clock for transferred signal.

**SVD:** Data for transferred signal.

**SVT:** Telemetry Data Line for SVI2 data transmission. Only effective when the analysis mode is set to SVI2.x.

**Display:** Display SVD or SVT decoding results in the waveform area. The SVT option is only effective when the analysis mode is set to SVI2.x.

### Result

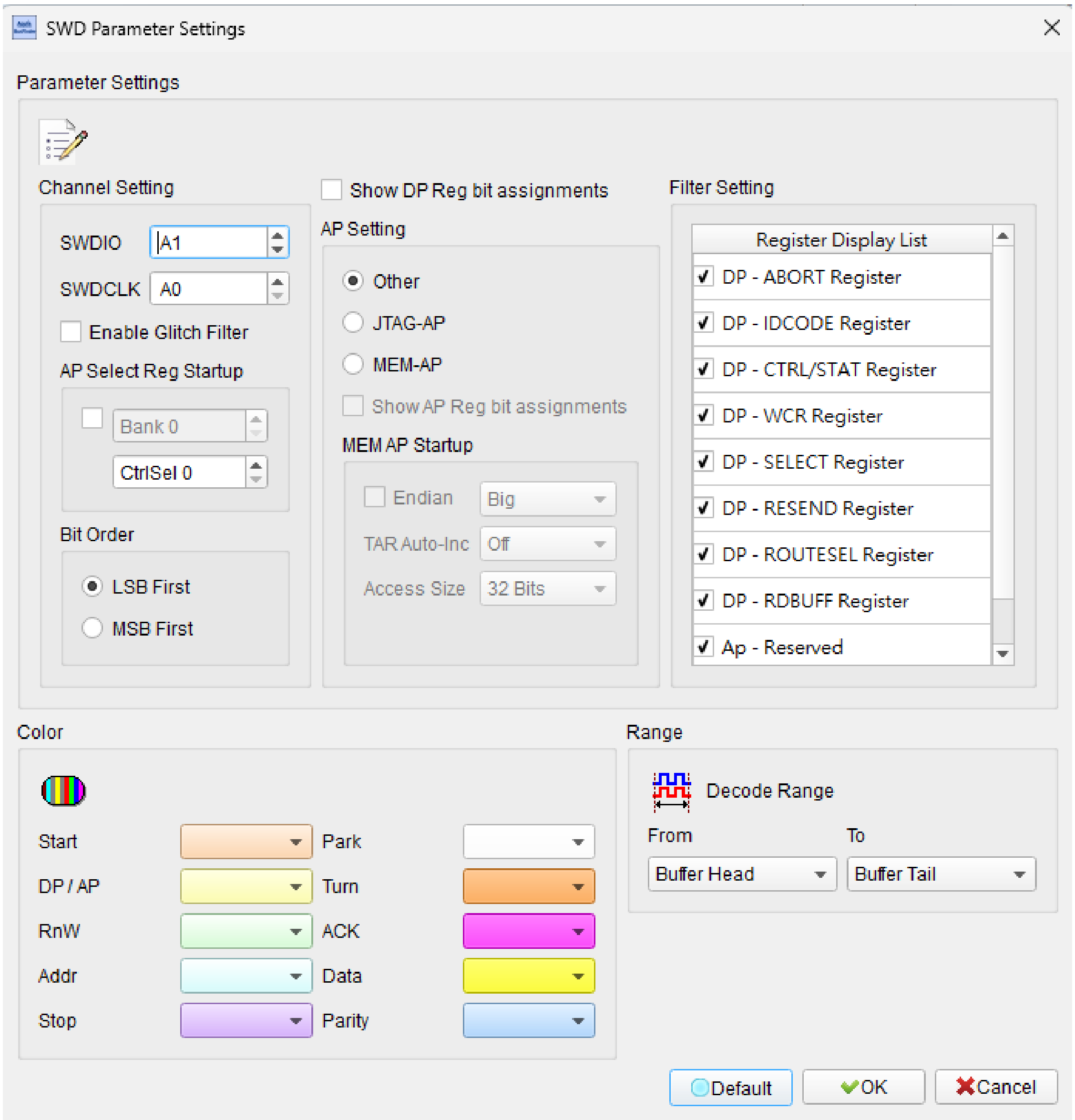


| | Timestamp | DIR | VDD1 | VDD2 | PSI0_L | PSI1_L | VID | TFN | Load Line Slope Trim | Offset Trim | SVT1 | SVT0 | VDD1 Volt | VDD1 Curr | VDD2 Volt | VDD2 Curr | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 83690 | -17.108667675s | | | | | | | | | | | | | | | | Time between Cmd and VOTF: 50.050 us |
| 83691 | -17.108249065s | SVD | 0 | 1 | 1 | 1 | 1.27500V (2C) | 0 | Initial LL Slope (3) | Initial Offset -25 mV (... | | | | | | | |
| 83692 | -17.10824723s | SVT | | | | | | | | | 1 | 0 | | | | | VOTF Complete |
| 83693 | -17.10824723s | | | | | | | | | | | | | | | | Time between Cmd and VOTF: 0.335 us |
| 83694 | -17.10791378s | SVD | 1 | 0 | 1 | 0 | 0.91250V (66) | 0 | Initial LL Slope (3) | Use Initial Offset (2) | | | | | | | |
| 83695 | -17.107862235s | SVT | | | | | | | | | 1 | 0 | | | | | VOTF Complete |
| 83696 | -17.107862235s | | | | | | | | | | | | | | | | Time between Cmd and VOTF: 50.045 us |
| 83697 | -17.10786193s | SVD | 1 | 0 | 1 | 1 | 0.91250V (66) | 0 | Initial LL Slope (3) | Use Initial Offset (2) | | | | | | | |
| 83698 | -17.107860115s | SVT | | | | | | | | | 1 | 0 | | | | | VOTF Complete |
| 83699 | -17.107860115s | | | | | | | | | | | | | | | | Time between Cmd and VOTF: 0.315 us |
| 83700 | -17.10773195s | SVD | 1 | 0 | 1 | 0 | 0.91250V (66) | 0 | Initial LL Slope (3) | Use Initial Offset (2) | | | | | | | |
| 83701 | -17.107730095s | SVT | | | | | | | | | 1 | 0 | | | | | VOTF Complete |
| 83702 | -17.107730095s | | | | | | | | | | | | | | | | Time between Cmd and VOTF: 0.355 us |
| 83703 | -17.10772038[illegible]s | SVD | 1 | 0 | 1 | 0 | 1.20000V (38) | 0 | Initial LL Slope (3) | Use Initial Offset (2) | | | | | | | |

## SWD

SWD (Serial Wire Debug) is a test protocol defined by ARM, consisting of two signal lines: SWDIO and SWDCLK. It serves as a debug access protocol for CoreSight™ Debug Access Port and is an alternative to JTAG when there are pin count limitations.

## Settings



SWD Parameter Settings

Parameter Settings

Channel Setting

SWDIO: A1

SWDCLK: A0

□ Enable Glitch Filter

AP Select Reg Startup

□ Bank 0

CtrlSel 0

Bit Order

● LSB First

○ MSB First

□ Show DP Reg bit assignments

AP Setting

● Other

○ JTAG-AP

○ MEM-AP

□ Show AP Reg bit assignments

MEM AP Startup

□ Endian: Big

TAR Auto-Inc: Off

Access Size: 32 Bits

Filter Setting

Register Display List

☑ DP - ABORT Register

☑ DP - IDCODE Register

☑ DP - CTRL/STAT Register

☑ DP - WCR Register

☑ DP - SELECT Register

☑ DP - RESEND Register

☑ DP - ROUTESEL Register

☑ DP - RDBUFF Register

☑ Ap - Reserved

Color

| | | | |
|---|---|---|---|
| Start | ▾ | Park | ▾ |
| DP / AP | ▾ | Turn | ▾ |
| RnW | ▾ | ACK | ▾ |
| Addr | ▾ | Data | ▾ |
| Stop | ▾ | Parity | ▾ |

Range

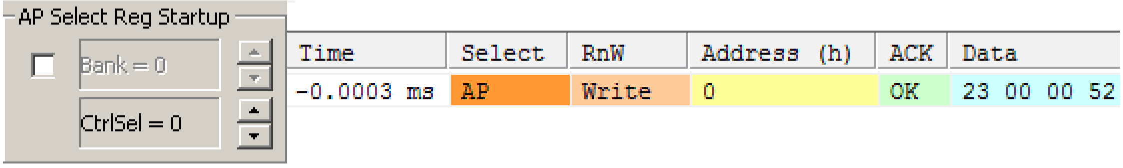
Decode Range

From: Buffer Head

To: Buffer Tail

**SWDIO**: I/O data.

**SWDCLK**: Clock.

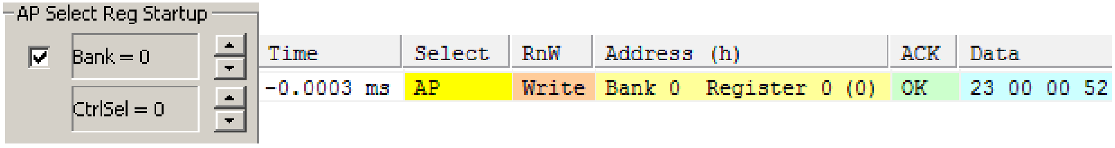


AP Select Reg Startup

☐ Bank = 0

CtrlSel = 0

| Time | Select | RnW | Address (h) | ACK | Data |
|---|---|---|---|---|---|
| -0.0003 ms | AP | Write | 0 | OK | 23 00 00 52 |



AP Select Reg Startup

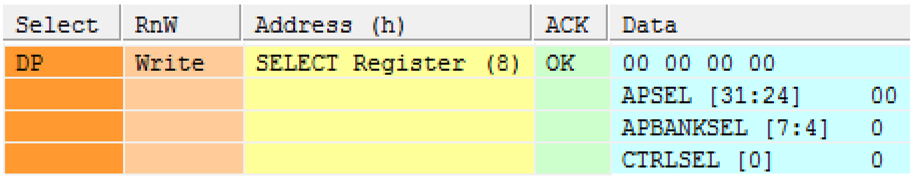
☑ Bank = 0

CtrlSel = 0

| Time | Select | RnW | Address (h) | ACK | Data |
|---|---|---|---|---|---|
| -0.0003 ms | AP | Write | Bank 0 Register 0 (0) | OK | 23 00 00 52 |

**Bit Order:** LSB or MSB.

**Show DP Reg bit assignments:** Show the DP register information.



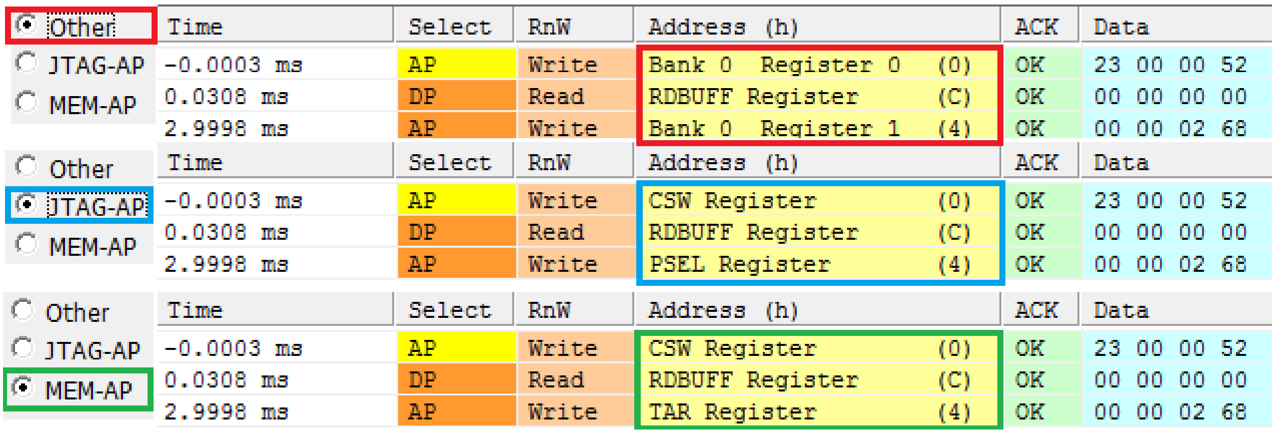
| Select | RnW | Address (h) | ACK | Data |
|---|---|---|---|---|
| DP | Write | SELECT Register (8) | OK | 00 00 00 00 |
| | | | | APSEL [31:24] 00 |
| | | | | APBANKSEL [7:4] 0 |
| | | | | CTRLSEL [0] 0 |

**AP Setting:** You can select between MEM-AP and JTAG-AP for AP Register decoding. If the user selects Other, the AP data will only display as Bank X Register X without further interpretation.

**JTAG-AP:** Show the JTAG AP decode.

**MEM-AP:** Show the MEM AP decode.

**Other:** Show Bank X Register X.



● Other
○ JTAG-AP
○ MEM-AP

| Time | Select | RnW | Address (h) | ACK | Data |
|---|---|---|---|---|---|
| -0.0003 ms | AP | Write | Bank 0 Register 0 (0) | OK | 23 00 00 52 |
| 0.0308 ms | DP | Read | RDBUFF Register (C) | OK | 00 00 00 00 |
| 2.9998 ms | AP | Write | Bank 0 Register 1 (4) | OK | 00 00 02 68 |

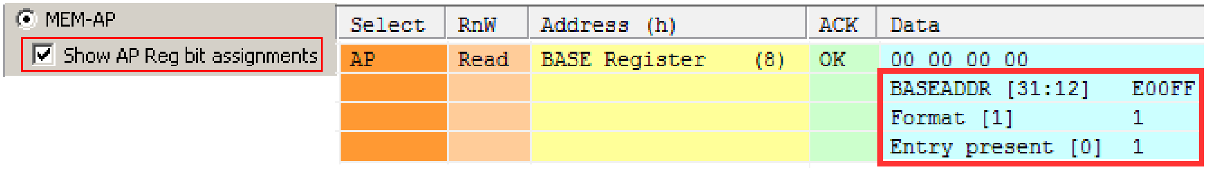
○ Other
● JTAG-AP
○ MEM-AP

| Time | Select | RnW | Address (h) | ACK | Data |
|---|---|---|---|---|---|
| -0.0003 ms | AP | Write | CSW Register (0) | OK | 23 00 00 52 |
| 0.0308 ms | DP | Read | RDBUFF Register (C) | OK | 00 00 00 00 |
| 2.9998 ms | AP | Write | PSEL Register (4) | OK | 00 00 02 68 |

○ Other
○ JTAG-AP
● MEM-AP

| Time | Select | RnW | Address (h) | ACK | Data |
|---|---|---|---|---|---|
| -0.0003 ms | AP | Write | CSW Register (0) | OK | 23 00 00 52 |
| 0.0308 ms | DP | Read | RDBUFF Register (C) | OK | 00 00 00 00 |
| 2.9998 ms | AP | Write | TAR Register (4) | OK | 00 00 02 68 |

**Show AP Reg bit assignments:** Display the AP register information if JTAG-AP or MEM-AP checked.

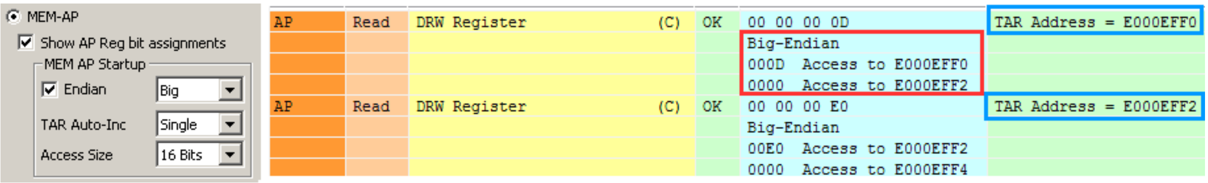


● MEM-AP

☑ Show AP Reg bit assignments

| Select | RnW | Address (h) | ACK | Data |
|---|---|---|---|---|
| AP | Read | BASE Register (8) | OK | 00 00 00 00 |
| | | | | BASEADDR [31:12] E00FF |
| | | | | Format [1] 1 |
| | | | | Entry present [0] 1 |

**MEM AP Startup:** When selecting MEM-AP, the contents of MEM-AP can be initialized. During data capture, if a corresponding register address is encountered, the data will be updated according to the bus content. Enabling the Endian checkbox activates the display of data along with the corresponding read/write addresses.



● MEM-AP

☑ Show AP Reg bit assignments

MEM AP Startup

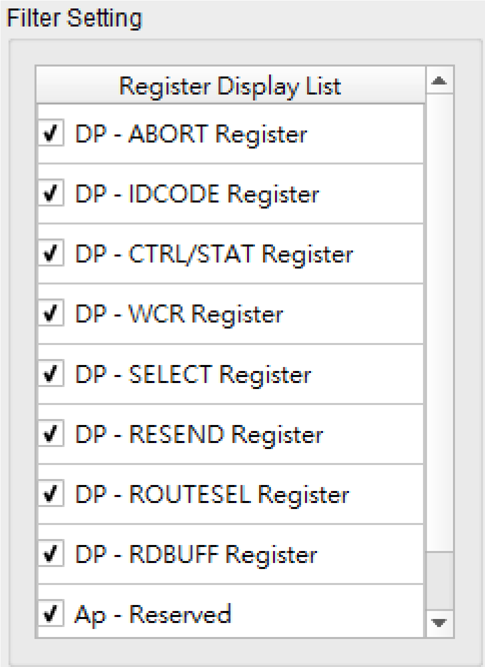
☑ Endian: Big

TAR Auto-Inc: Single

Access Size: 16 Bits

| | | | | | |
|---|---|---|---|---|---|
| AP | Read | DRW Register (C) | OK | 00 00 00 0D | TAR Address = E000EFF0 |
| | | | | Big-Endian | |
| | | | | 000D Access to E000EFF0 | |
| | | | | 0000 Access to E000EFF2 | |
| AP | Read | DRW Register (C) | OK | 00 00 00 E0 | TAR Address = E000EFF2 |
| | | | | Big-Endian | |
| | | | | 00E0 Access to E000EFF2 | |
| | | | | 0000 Access to E000EFF4 | |

**Filter Setting:** Filter the unwanted Registers.

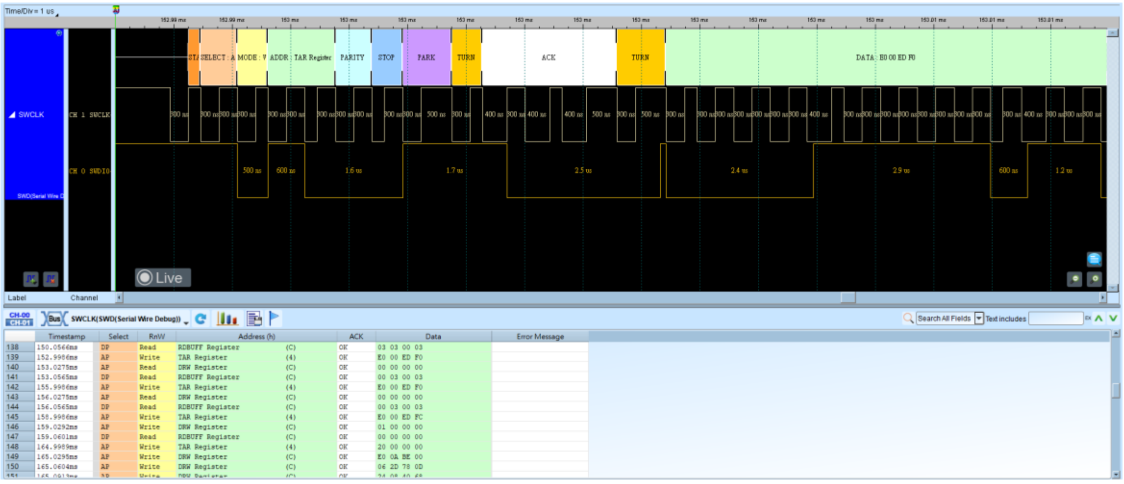


Filter Setting

Register Display List

- ✓ DP - ABORT Register
- ✓ DP - IDCODE Register
- ✓ DP - CTRL/STAT Register
- ✓ DP - WCR Register
- ✓ DP - SELECT Register
- ✓ DP - RESEND Register
- ✓ DP - ROUTESEL Register
- ✓ DP - RDBUFF Register
- ✓ Ap - Reserved

## Result

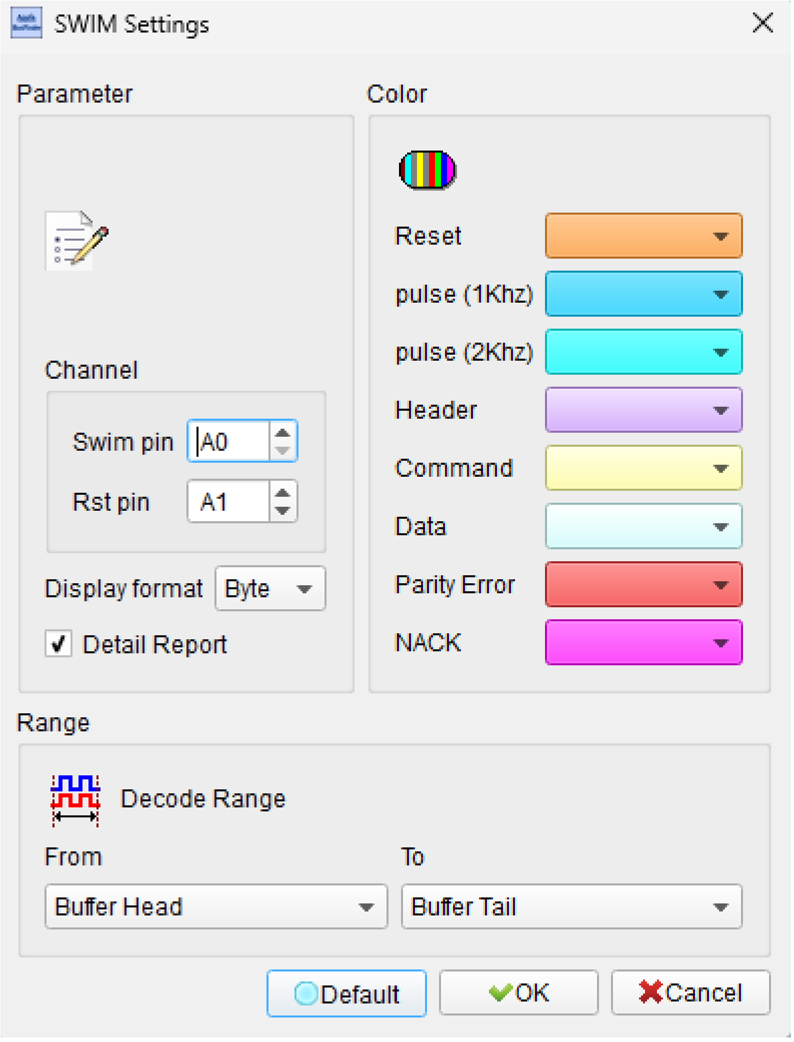


| | Timestamp | Select | RnW | Address (h) | | ACK | Data | Error Message |
|---|---|---|---|---|---|---|---|---|
| 138 | 150.0566ms | DP | Read | RDBUFF Register | (C) | OK | 03 03 00 03 | |
| 139 | 152.9986ms | AP | Write | TAR Register | (4) | OK | E0 00 ED F0 | |
| 140 | 153.0275ms | AP | Read | DRW Register | (C) | OK | 00 00 00 00 | |
| 141 | 153.0565ms | DP | Read | RDBUFF Register | (C) | OK | 00 03 00 03 | |
| 142 | 155.9986ms | AP | Write | TAR Register | (4) | OK | E0 00 ED F0 | |
| 143 | 156.0275ms | AP | Read | DRW Register | (C) | OK | 00 00 00 00 | |
| 144 | 156.0565ms | DP | Read | RDBUFF Register | (C) | OK | 00 03 00 03 | |
| 145 | 158.9986ms | AP | Write | TAR Register | (4) | OK | E0 00 ED FC | |
| 146 | 159.0292ms | AP | Write | DRW Register | (C) | OK | 01 00 00 00 | |
| 147 | 159.0601ms | DP | Read | RDBUFF Register | (C) | OK | 00 00 00 00 | |
| 148 | 164.9989ms | AP | Write | TAR Register | (4) | OK | 20 00 00 00 | |
| 149 | 165.0295ms | AP | Write | DRW Register | (C) | OK | E0 0A BE 00 | |
| 150 | 165.0604ms | AP | Write | DRW Register | (C) | OK | 06 2D 78 0D | |
| 151 | 165.0913ms | AP | Write | DRW Register | (C) | OK | 24 08 40 68 | |

### SWIM

SWIM is the single wire interface module of STM8 8-bit MCUs family. While the CPU is running, the SWIM allows a non-intrusive read/write accesses to be performed on-the-fly to the RAM and peripheral registers, for debug purposes.

### Settings



SWIM Settings

Parameter

Channel

Swim pin: A0

Rst pin: A1

Display format: Byte

☑ Detail Report

Color

Reset, pulse (1Khz), pulse (2Khz), Header, Command, Data, Parity Error, NACK

Range

Decode Range

From: Buffer Head

To: Buffer Tail

Swim pin: set the Swim Channel.

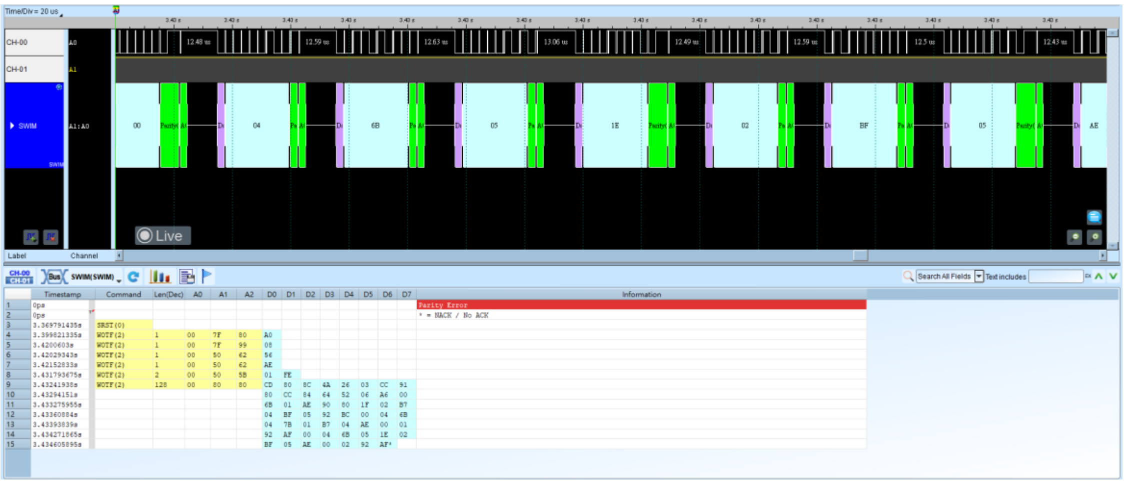
Rst pin: set the Reset Channel.

Display format: the display method of waveform decode (Byte, Bit).

Detail Report: show the detail information of SWIM decode.

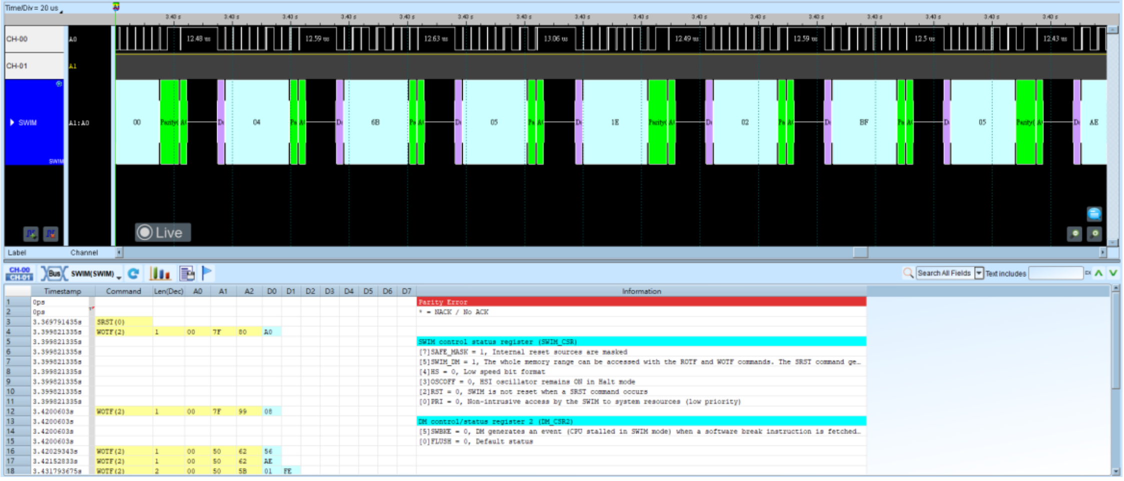
## Result

## Normal



| | Timestamp | Command | Len(Dec) | A0 | A1 | A2 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0ps | | | | | | | | | | | | | | Parity Error |
| 2 | 0ps | | | | | | | | | | | | | | * = NACK / No ACK |
| 3 | 3.369791435s | SRST(0) | | | | | | | | | | | | | |
| 4 | 3.399821335s | WOTF(2) | 1 | 00 | 7F | 80 | A0 | | | | | | | | |
| 5 | 3.4200603s | WOTF(2) | 1 | 00 | 7F | 99 | 08 | | | | | | | | |
| 6 | 3.42029343s | WOTF(2) | 1 | 00 | 50 | 62 | 56 | | | | | | | | |
| 7 | 3.42152833s | WOTF(2) | 1 | 00 | 50 | 62 | AE | | | | | | | | |
| 8 | 3.431793675s | WOTF(2) | 2 | 00 | 50 | 5B | 01 | FE | | | | | | | |
| 9 | 3.43241938s | WOTF(2) | 128 | 00 | 80 | 80 | CD | 80 | 8C | 4A | 26 | 03 | CC | 91 | |
| 10 | 3.43294151s | | | | | | 80 | CC | 84 | 64 | 52 | 06 | A6 | 00 | |
| 11 | 3.433275955s | | | | | | 6B | 01 | AE | 90 | 80 | 1F | 02 | B7 | |
| 12 | 3.43360884s | | | | | | 04 | BF | 05 | 92 | BC | 00 | 04 | 6B | |
| 13 | 3.43393839s | | | | | | 04 | 7B | 01 | B7 | 04 | AE | 00 | 01 | |
| 14 | 3.434271865s | | | | | | 92 | AF | 00 | 04 | 6B | 05 | 1E | 02 | |
| 15 | 3.434605895s | | | | | | BF | 05 | AE | 00 | 02 | 92 | AF* | | |

## Detail Report

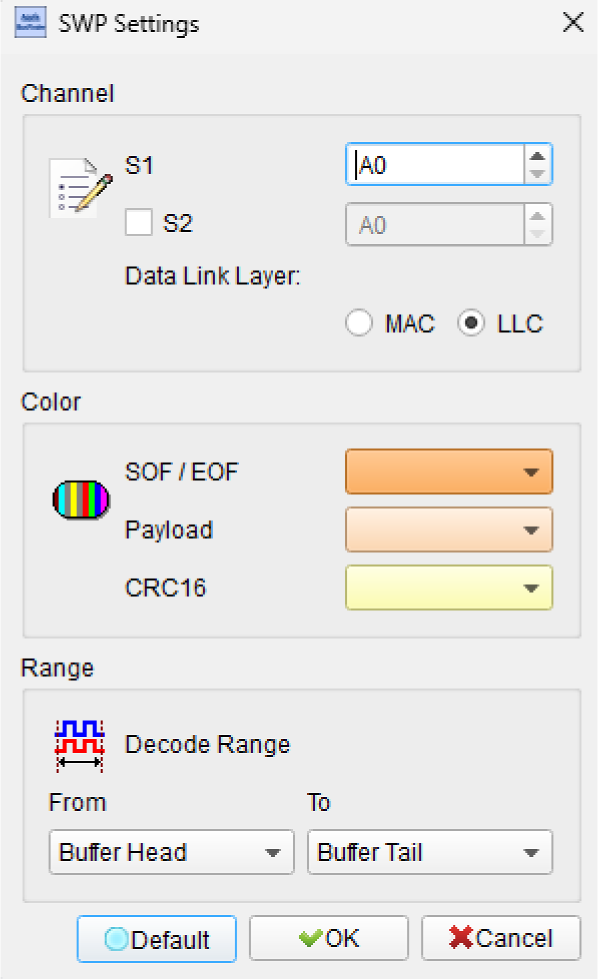


| | Timestamp | Command | Len(Dec) | A0 | A1 | A2 | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0ps | | | | | | | | | | | | | | Parity Error |
| 2 | 0ps | | | | | | | | | | | | | | * = NACK / No ACK |
| 3 | 3.369791435s | SRST(0) | | | | | | | | | | | | | |
| 4 | 3.399821335s | WOTF(2) | 1 | 00 | 7F | 80 | A0 | | | | | | | | |
| 5 | 3.399821335s | | | | | | | | | | | | | | SWIM control status register (SWIM_CSR) |
| 6 | 3.399821335s | | | | | | | | | | | | | | [7]SAFE_MASK = 1, Internal reset sources are masked |
| 7 | 3.399821335s | | | | | | | | | | | | | | [5]SWIM_DM = 1, The whole memory range can be accessed with the ROTF and WOTF commands. The SRST command ge… |
| 8 | 3.399821335s | | | | | | | | | | | | | | [4]HS = 0, Low speed bit format |
| 9 | 3.399821335s | | | | | | | | | | | | | | [3]OSCOFF = 0, HSI oscillator remains ON in Halt mode |
| 10 | 3.399821335s | | | | | | | | | | | | | | [2]RST = 0, SWIM is not reset when a SRST command occurs |
| 11 | 3.399821335s | | | | | | | | | | | | | | [0]PRI = 0, Non-intrusive access by the SWIM to system resources (low priority) |
| 12 | 3.4200603s | WOTF(2) | 1 | 00 | 7F | 99 | 08 | | | | | | | | |
| 13 | 3.4200603s | | | | | | | | | | | | | | DM control/status register 2 (DM_CSR2) |
| 14 | 3.4200603s | | | | | | | | | | | | | | [5]SWBKE = 0, DM generates an event (CPU stalled in SWIM mode) when a software break instruction is fetched… |
| 15 | 3.4200603s | | | | | | | | | | | | | | [0]FLUSH = 0, Default status |
| 16 | 3.42029343s | WOTF(2) | 1 | 00 | 50 | 62 | 56 | | | | | | | | |
| 17 | 3.42152833s | WOTF(2) | 1 | 00 | 50 | 62 | AE | | | | | | | | |
| 18 | 3.431793675s | WOTF(2) | 2 | 00 | 50 | 5B | 01 | FE | | | | | | | |

## SWP

The Single Wire Protocol (SWP) is a single-wire connection between the SIM card and a NFC chip in a cell phone.

### Settings


SWP Settings

Channel

S1: A0

□ S2: A0

Data Link Layer: ○ MAC ● LLC

Color

SOF / EOF

Payload

CRC16

Range

Decode Range

From: Buffer Head

To: Buffer Tail

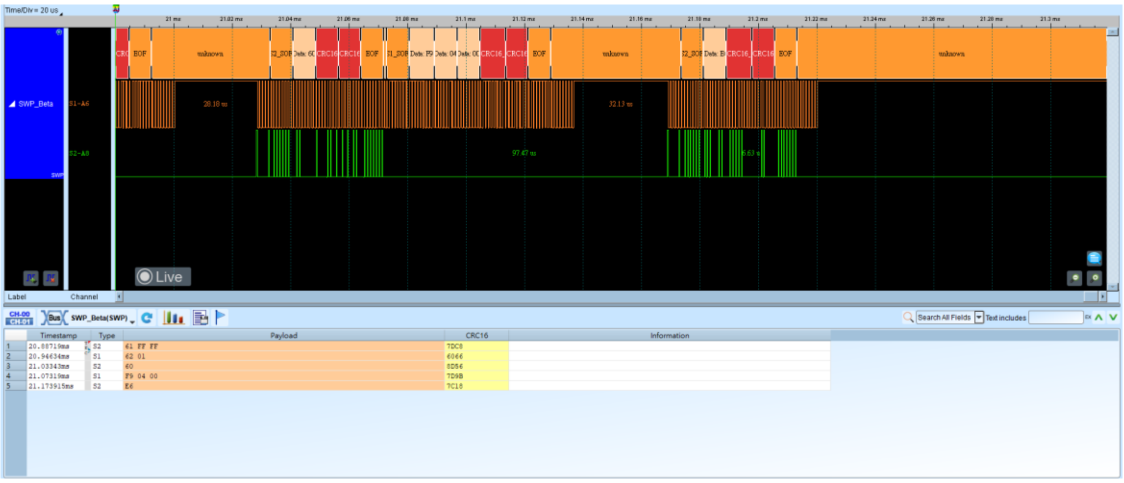
**S1**: I/O data.

**S2**: I/O data in current domain and it need convert to voltage domain.

**Data Link Layer**: Supported MAC and LLC layers

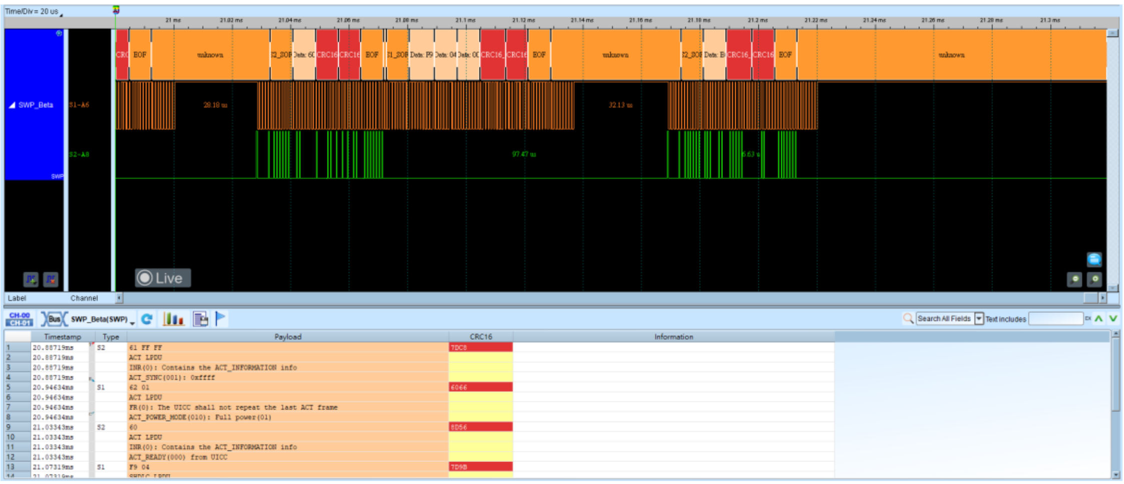
## Result

## MAC



| | Timestamp | Type | Payload | CRC16 | Information |
|---|---|---|---|---|---|
| 1 | 20.88719ms | S2 | 61 FF FF | 7DC8 | |
| 2 | 20.94634ms | S1 | 62 01 | 6066 | |
| 3 | 21.03343ms | S2 | 60 | 8D56 | |
| 4 | 21.07319ms | S1 | F9 04 00 | 7D9B | |
| 5 | 21.173915ms | S2 | E6 | 7C18 | |

## LLC

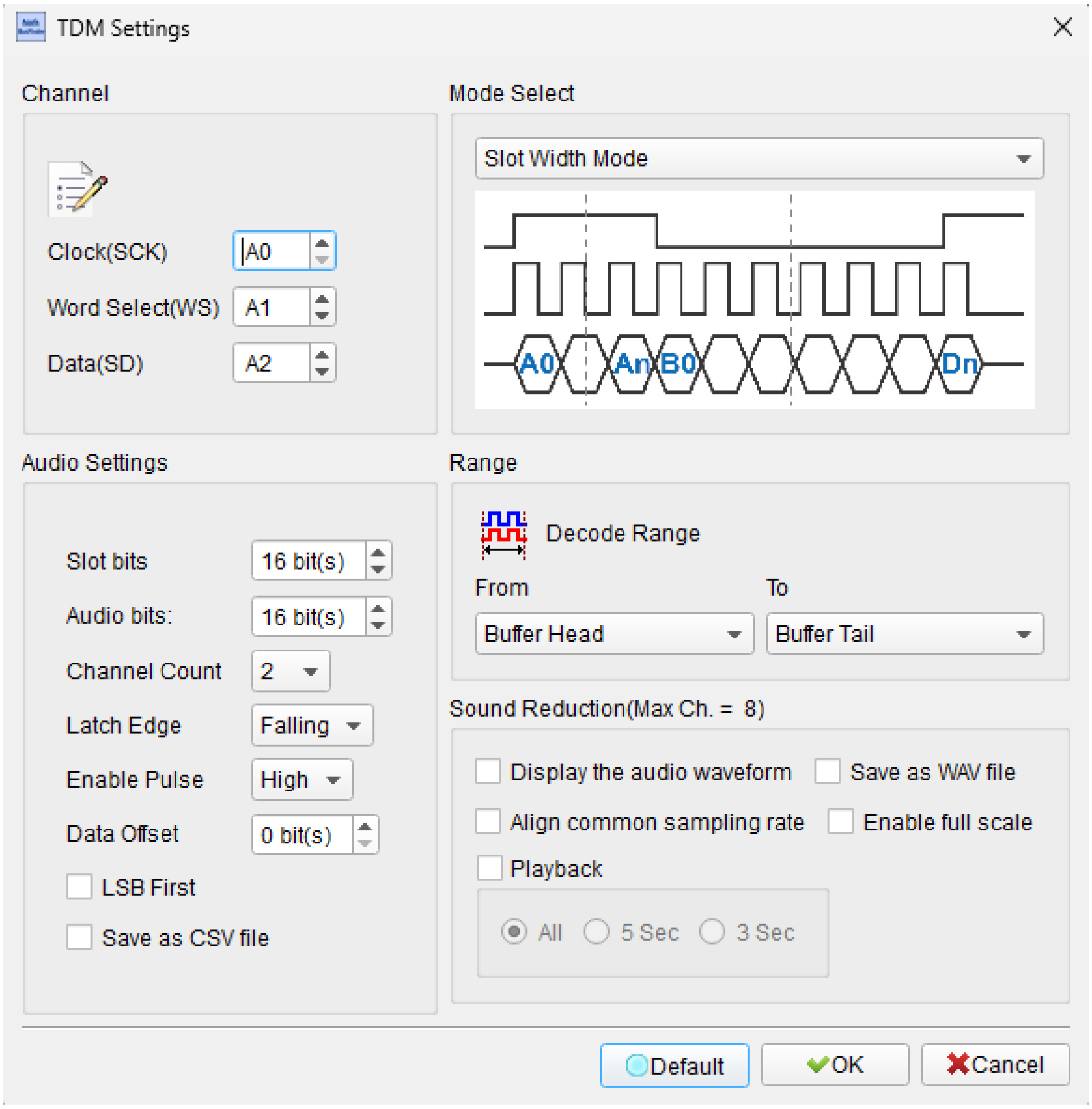


| | Timestamp | Type | Payload | CRC16 | Information |
|---|---|---|---|---|---|
| 1 | 20.88719ms | S2 | 61 FF FF | 7DC8 | |
| 2 | 20.88719ms | | ACT LPDU | | |
| 3 | 20.88719ms | | INR(0): Contains the ACT_INFORMATION info | | |
| 4 | 20.88719ms | | ACT_SYNC(001): 0xffff | | |
| 5 | 20.94634ms | S1 | 62 01 | 6066 | |
| 6 | 20.94634ms | | ACT LPDU | | |
| 7 | 20.94634ms | | FR(0): The UICC shall not repeat the last ACT frame | | |
| 8 | 20.94634ms | | ACT_POWER_MODE(010): Full power(01) | | |
| 9 | 21.03343ms | S2 | 60 | 8D56 | |
| 10 | 21.03343ms | | ACT LPDU | | |
| 11 | 21.03343ms | | INR(0): Contains the ACT_INFORMATION info | | |
| 12 | 21.03343ms | | ACT_READY(000) from UICC | | |
| 13 | 21.07319ms | S1 | F9 04 | 7D9B | |
| 14 | 21.07319ms | | SHDLC LPDU | | |

## TDM

TDM Audio (Time Division Multiplexing Audio) is a method for transmitting multiple audio channels over a single data line (or a pair of data lines) by dividing the transmission content into multiple time slots. This technique is widely used in digital audio systems to efficiently manage and transmit multiple audio streams, particularly in professional audio, consumer electronics, and automotive applications.

## Settings



TDM Settings

Channel

Clock(SCK): A0

Word Select(WS): A1

Data(SD): A2

Mode Select: Slot Width Mode

Audio Settings

Slot bits: 16 bit(s)

Audio bits: 16 bit(s)

Channel Count: 2

Latch Edge: Falling

Enable Pulse: High

Data Offset: 0 bit(s)

□ LSB First

□ Save as CSV file

Range

Decode Range

From: Buffer Head

To: Buffer Tail

Sound Reduction(Max Ch. = 8)

□ Display the audio waveform □ Save as WAV file

□ Align common sampling rate □ Enable full scale

□ Playback

● All ○ 5 Sec ○ 3 Sec

**Channel:** The signal pins of the Device Under Test (DUT) are assigned to the channel numbers of the logic analyzer.

**Option:** Adjust the detailed settings of the audio signal.

**Slot bits:** Data width of an audio channel.

**Audio bits:** Valid bits in the slot bits.

**Channel Count:** Number of audio channels (e.g., 2, 4, 8)

**Latch Edge:** Choose which clock edge to latch data (Rising or Falling)

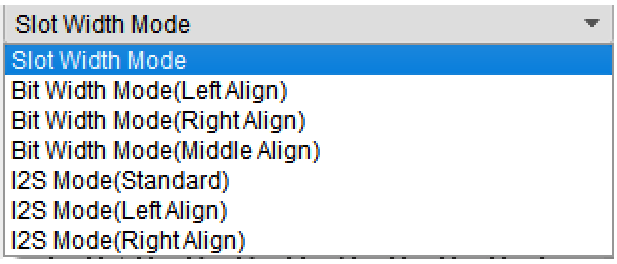
**Start Pulse:** Set the polarity of the active pulse (High or Low)

**Data Offset:** Bit offset before starting data capture

**LSB First:** Check this if the data is transmitted with LSB first

**Save as CSV file:** Save the decoded result as a .CSV file

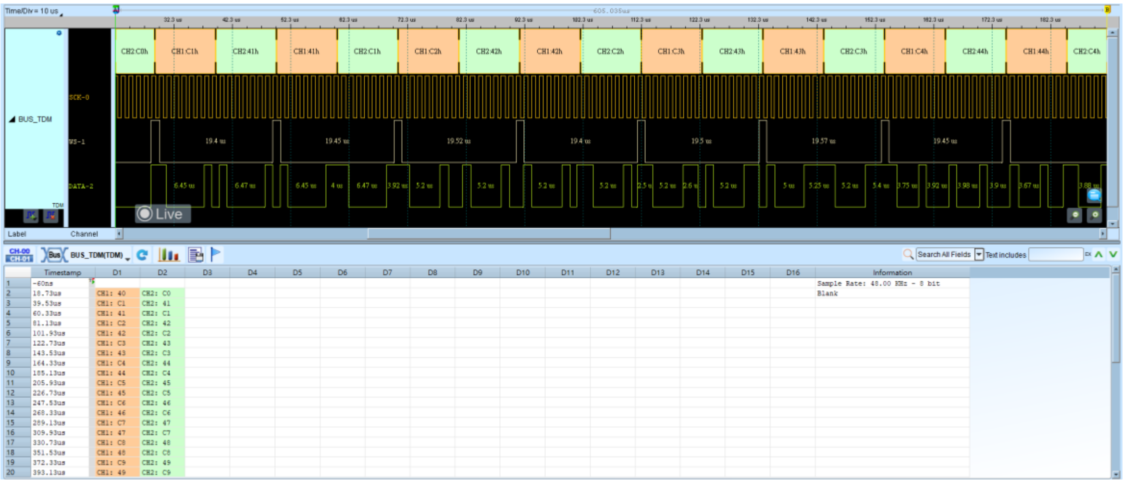
**Mode:** Set the data arrangement format. The following formats are currently supported:



**Sound Reduction:** Set whether to save, replay, or visualize the audio waveform based on the analysis results after decoding. Enabled when selected.

- **Display the audio waveform**
- **Enable meet full scale**
- **Save as WAV file**
- **Align common sampling rate**
- **Playback: Choose to play all, 5 seconds, or 3 seconds**

## Result

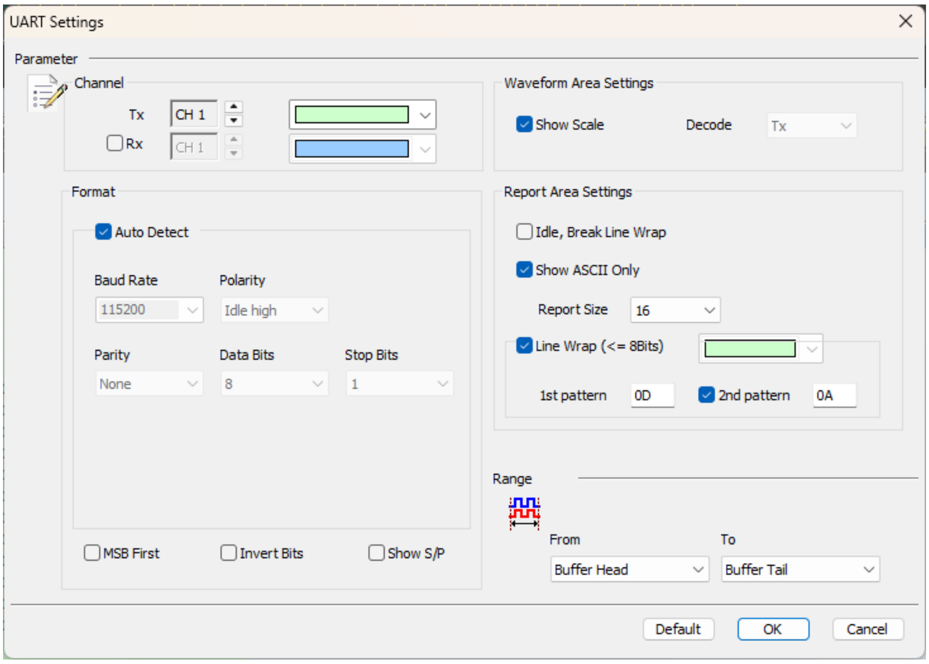


| | Timestamp | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | D16 | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -60ns | | | | | | | | | | | | | | | | | Sample Rate: 48.00 KHz - 8 bit |
| 2 | 18.73us | CH1: 40 | CH2: C0 | | | | | | | | | | | | | | | Blank |
| 3 | 39.53us | CH1: C1 | CH2: 41 | | | | | | | | | | | | | | | |
| 4 | 60.33us | CH1: 41 | CH2: C1 | | | | | | | | | | | | | | | |
| 5 | 81.13us | CH1: C2 | CH2: 42 | | | | | | | | | | | | | | | |
| 6 | 101.93us | CH1: 42 | CH2: C2 | | | | | | | | | | | | | | | |
| 7 | 122.73us | CH1: C3 | CH2: 43 | | | | | | | | | | | | | | | |
| 8 | 143.53us | CH1: 43 | CH2: C3 | | | | | | | | | | | | | | | |
| 9 | 164.33us | CH1: C4 | CH2: 44 | | | | | | | | | | | | | | | |
| 10 | 185.13us | CH1: 44 | CH2: C4 | | | | | | | | | | | | | | | |
| 11 | 205.93us | CH1: C5 | CH2: 45 | | | | | | | | | | | | | | | |
| 12 | 226.73us | CH1: 45 | CH2: C5 | | | | | | | | | | | | | | | |
| 13 | 247.53us | CH1: C6 | CH2: 46 | | | | | | | | | | | | | | | |
| 14 | 268.33us | CH1: 46 | CH2: C6 | | | | | | | | | | | | | | | |
| 15 | 289.13us | CH1: C7 | CH2: 47 | | | | | | | | | | | | | | | |
| 16 | 309.93us | CH1: 47 | CH2: C7 | | | | | | | | | | | | | | | |
| 17 | 330.73us | CH1: C8 | CH2: 48 | | | | | | | | | | | | | | | |
| 18 | 351.53us | CH1: 48 | CH2: C8 | | | | | | | | | | | | | | | |
| 19 | 372.33us | CH1: C9 | CH2: 49 | | | | | | | | | | | | | | | |
| 20 | 393.13us | CH1: 49 | CH2: C9 | | | | | | | | | | | | | | | |

## UART(RS-232, RS485)

It is a serial data communication interface standard established by the Electronic Industries Alliance (EIA) in the United States. In the RS-232 and RS-485 standards, characters are transmitted sequentially as a series of bits in a serial manner. The advantages of this method include fewer transmission lines, simpler wiring, and longer transmission distances. Since RS-485 uses differential signaling, the signal must be converted into a logic signal before measurement. The logic analyzer (LA) cannot directly measure differential signals.

### Settings



UART Settings

Parameter

Channel
Tx: CH 1 🟩
□ Rx: CH 1 🟦

Waveform Area Settings
☑ Show Scale
Decode: Tx

Format
☑ Auto Detect
Baud Rate: 115200
Polarity: Idle high
Parity: None
Data Bits: 8
Stop Bits: 1
□ MSB First □ Invert Bits □ Show S/P

Report Area Settings
□ Idle, Break Line Wrap
☑ Show ASCII Only
Report Size: 16
☑ Line Wrap (<= 8Bits) 🟩
1st pattern: 0D
☑ 2nd pattern: 0A

Range
From: Buffer Head
To: Buffer Tail

### Channel:

**Tx:** Assign the Tx signal pin to the corresponding channel number of the logic analyzer.

**Rx:** Assign the Rx signal pin to the corresponding channel number of the logic analyzer. Enabled when selected.

**Auto:** Automatically detect the settings for the following options. Enabled when selected.

**Baud Rate:** The data transmission speed, measured in bits per second (bps). The supported range is 110 to 2M bps.

**Polarity:** Includes Idle High and Idle Low formats.

**Parity:** Includes N - None Parity (No parity bit), O - Odd Parity, and E - Even Parity.

**Data Bits:** Can be set between 4 to 16 bits.

**Stop Bits:** Can be set to 1, 1.5, 2, 2.5, 3, 3.5, 4, or 4.5 bits.

**MSB First:** When selected, the Most Significant Bit (MSB) comes after the Start Bit. When not selected, the Least Significant Bit (LSB) is used.

**Invert Bits:** Reverses High and Low levels. Enabled when selected.

**Show S/P:** Displays Start and Stop in the waveform area. Enabled when selected.

**Waveform Area Settings:**

**Decode:** Select whether to display the Rx or Tx decoding results in the waveform area. The Rx option is only effective when the Rx channel is enabled.

**Show Scale:** Displays the scale in the waveform area. Enabled when selected.

**Report Area Settings, enabled when selected:**

**Idle, Break Line Wrap:** When the bus Idle/Break, the report is displayed on a new line.

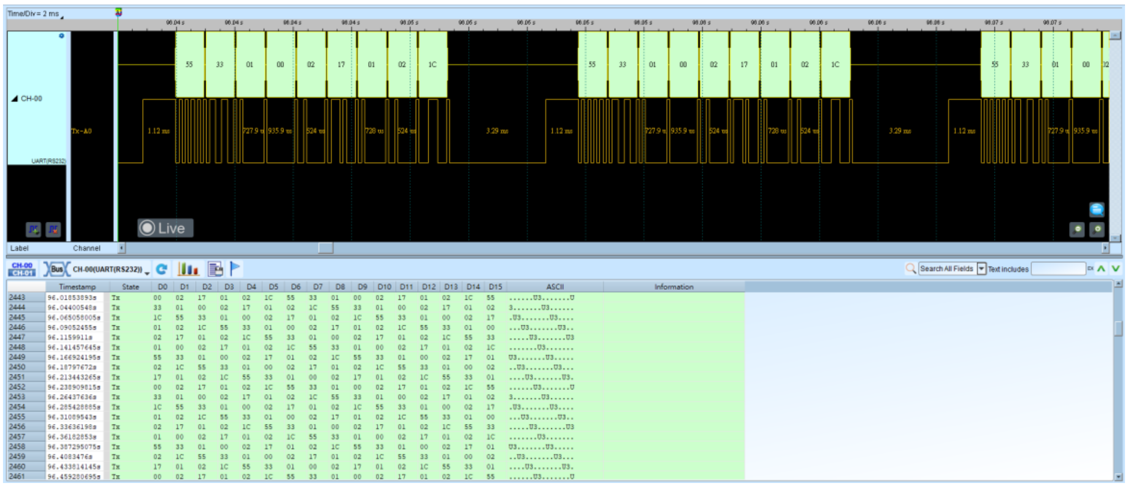
**Show ASCII Only:** Show only ASCII reports.

**Report Size:** Sets the number of Data fields in the report area. Can be set to 16 or 32.

**Line Wrap Data:** Allows setting two values as the primary order for decoding, making it easier to view analysis results.

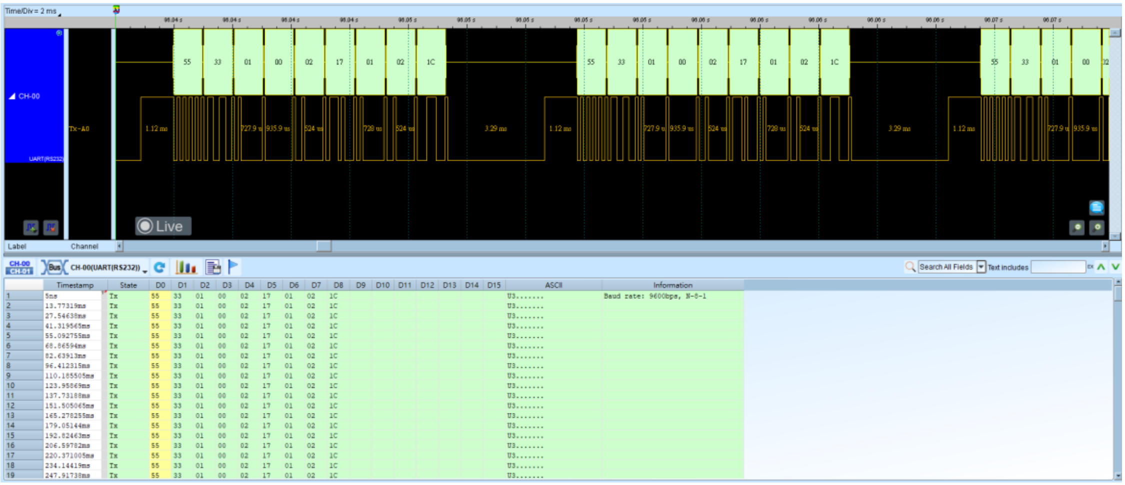
## Result

### Normal Data Analysis Displaying



| | Timestamp | State | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ASCII | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2443 | 96.01853893s | Tx | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | ......U3.......U | |
| 2444 | 96.04400548s | Tx | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 3.......U3...... | |
| 2445 | 96.065058005s | Tx | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | .U3.......U3.... | |
| 2446 | 96.09052455s | Tx | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | ...U3.......U3.. | |
| 2447 | 96.1159911s | Tx | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | .....U3.......U3 | |
| 2448 | 96.141457645s | Tx | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | .......U3....... | |
| 2449 | 96.166924195s | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | U3.......U3..... | |
| 2450 | 96.18797672s | Tx | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | ..U3.......U3... | |
| 2451 | 96.213443265s | Tx | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | ....U3.......U3. | |
| 2452 | 96.238909815s | Tx | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | ......U3.......U | |
| 2453 | 96.26437636s | Tx | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 3.......U3...... | |
| 2454 | 96.285428885s | Tx | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | .U3.......U3.... | |
| 2455 | 96.31089543s | Tx | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | ...U3.......U3.. | |
| 2456 | 96.33636198s | Tx | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | .....U3.......U3 | |
| 2457 | 96.36182853s | Tx | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | .......U3....... | |
| 2458 | 96.387295075s | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | U3.......U3..... | |
| 2459 | 96.4083476s | Tx | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | ..U3.......U3... | |
| 2460 | 96.433814145s | Tx | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | ....U3.......U3. | |
| 2461 | 96.459280695s | Tx | 00 | 02 | 17 | 01 | 02 | 1C | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | 55 | ......U3.......U | |

### Enable Line Wrap Data Analysis Displaying

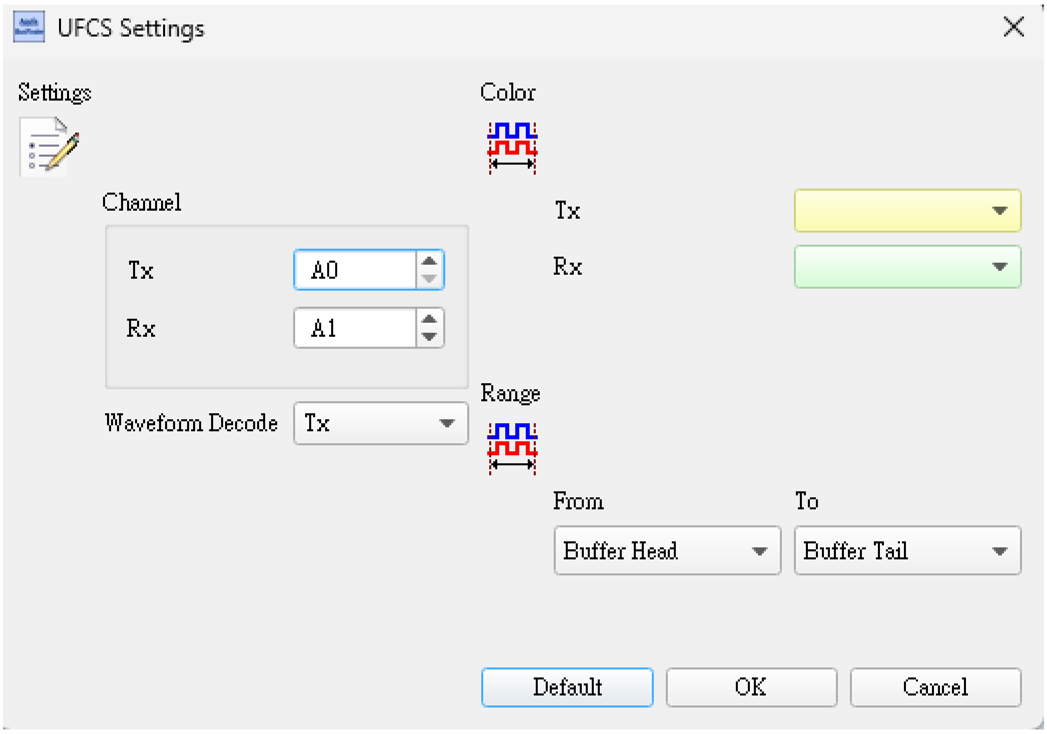


| | Timestamp | State | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | ASCII | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 5ns | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | Baud rate: 9600bps, N-8-1 |
| 2 | 13.77319ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 3 | 27.54638ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 4 | 41.319565ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 5 | 55.092755ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 6 | 68.86594ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 7 | 82.63913ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 8 | 96.412315ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 9 | 110.185505ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 10 | 123.95869ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 11 | 137.73188ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 12 | 151.505065ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 13 | 165.278255ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 14 | 179.05144ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 15 | 192.82463ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 16 | 206.59782ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 17 | 220.371005ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 18 | 234.14419ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |
| 19 | 247.91738ms | Tx | 55 | 33 | 01 | 00 | 02 | 17 | 01 | 02 | 1C | | | | | | | | U3....... | |

## UFCS

UFCS(Universal Fast Charging Specification) formulate integrated fast charging standards for mobile terminals to solve the problem of incompatibility between fast charging and create a fast, safe and compatible charging environment for end users. The power supply port of UFCS adopts USB Type-A charging port, and the signal transmission is based on USB D+/D-.

### Settings



UFCS Settings

Settings

Channel

Tx: A0

Rx: A1

Waveform Decode: Tx

Color

Tx: 🟨

Rx: 🟩

Range

From: Buffer Head

To: Buffer Tail

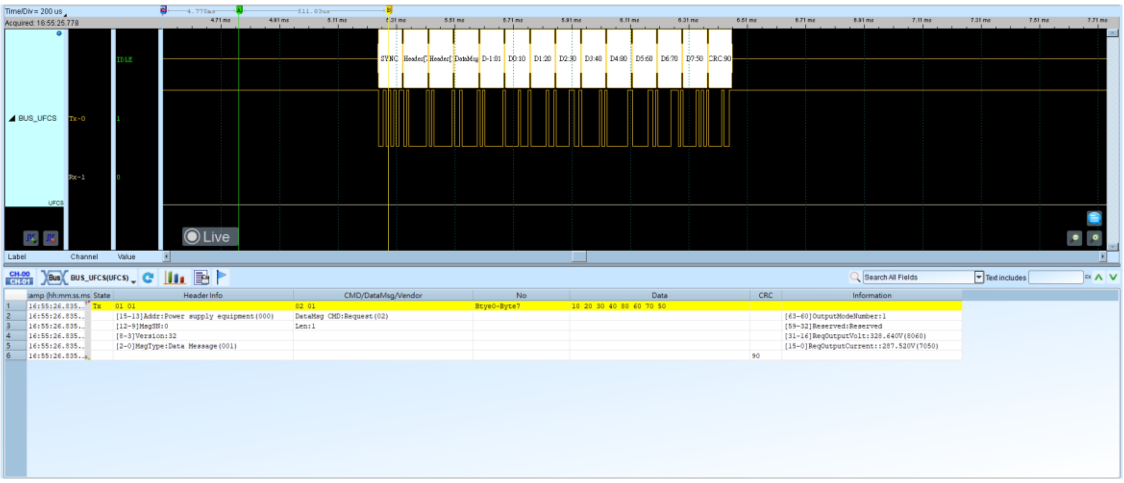
**Channel:** Set the Tx/Rx channel of UFCS.

**Waveform Decode:** Tx/ Rx.

(Multiple sets of waveform area decoding cannot be displayed on a single decoder. If you need to view multiple sets at the same time, please add an additional set of decoders.)

## Result

## UFCS

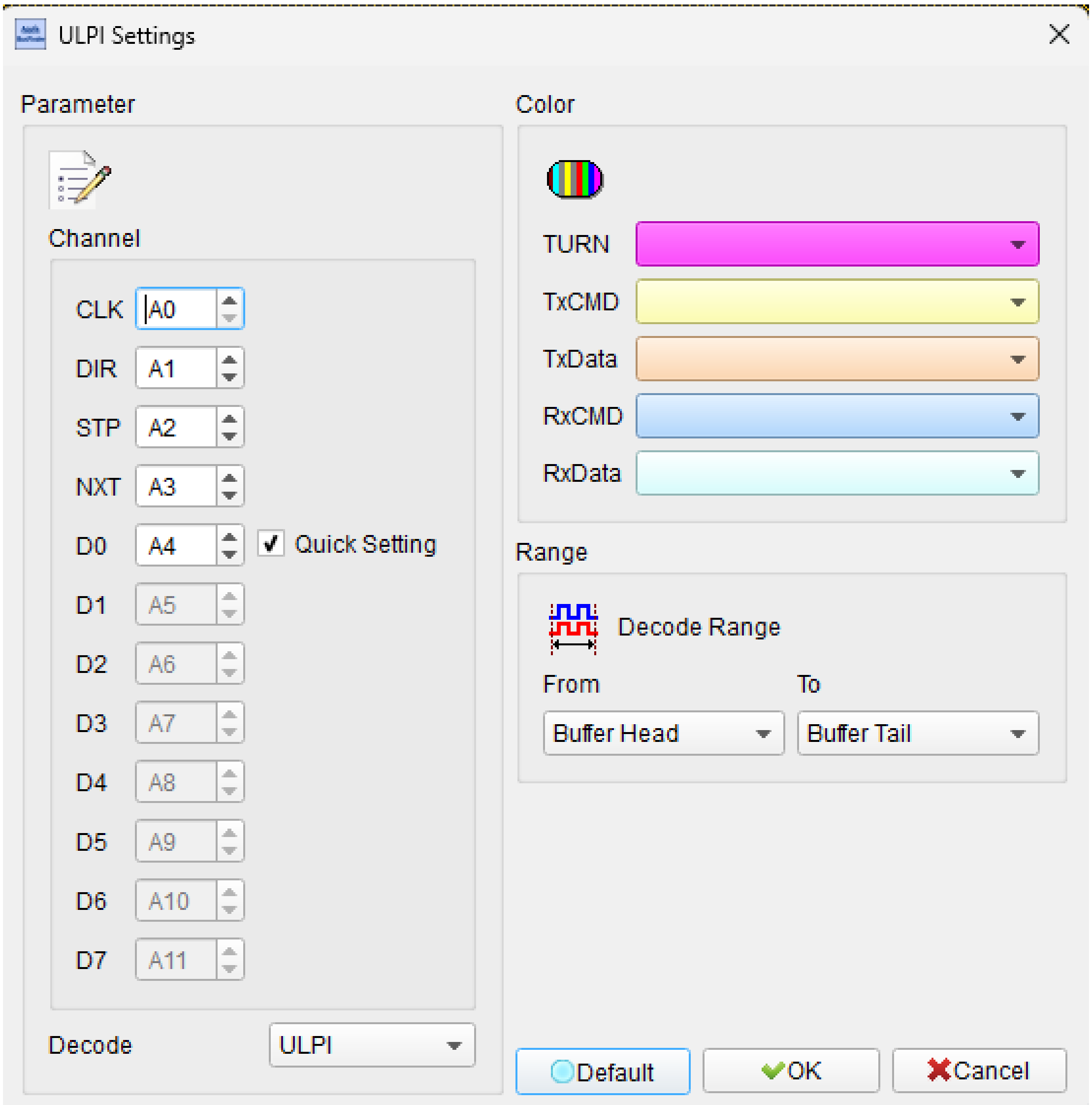


| | tamp (hh:mm:ss.ms | State | Header Info | CMD/DataMsg/Vendor | No | Data | CRC | Information |
|---|---|---|---|---|---|---|---|---|
| 1 | 16:55:26.835.. | Tx | 01 01 | 02 01 | Btye0-Byte7 | 10 20 30 40 80 60 70 50 | | |
| 2 | 16:55:26.835.. | | [15-13]Addr:Power supply equipment(000) | DataMsg CMD:Request(02) | | | | [63-60]OutputModeNumber:1 |
| 3 | 16:55:26.835.. | | [12-9]MsgSN:0 | Len:1 | | | | [59-32]Reserved:Reserved |
| 4 | 16:55:26.835.. | | [8-3]Version:32 | | | | | [31-16]ReqOutputVolt:328.640V(8060) |
| 5 | 16:55:26.835.. | | [2-0]MsgType:Data Message(001) | | | | | [15-0]ReqOutputCurrent::287.520V(7050) |
| 6 | 16:55:26.835.. | | | | | | 90 | |

## ULPI

UTMI + Low Pin Interface. ULPI is the Low Pin version of UTMI. UTMI (USB2.0 Transceiver Macrocell Interface) is a protocol for USB controller and USB PHY communication. Compared with ULPI, UTMI has more control signals and supports 8bit / 16bit data interface.

**Parameter:**



ULPI Settings

Parameter

Channel

| | |
|---|---|
| CLK | A0 |
| DIR | A1 |
| STP | A2 |
| NXT | A3 |
| D0 | A4 |
| D1 | A5 |
| D2 | A6 |
| D3 | A7 |
| D4 | A8 |
| D5 | A9 |
| D6 | A10 |
| D7 | A11 |

☑ Quick Setting

Decode: ULPI

Color

TURN, TxCMD, TxData, RxCMD, RxData

Range

Decode Range

From: Buffer Head

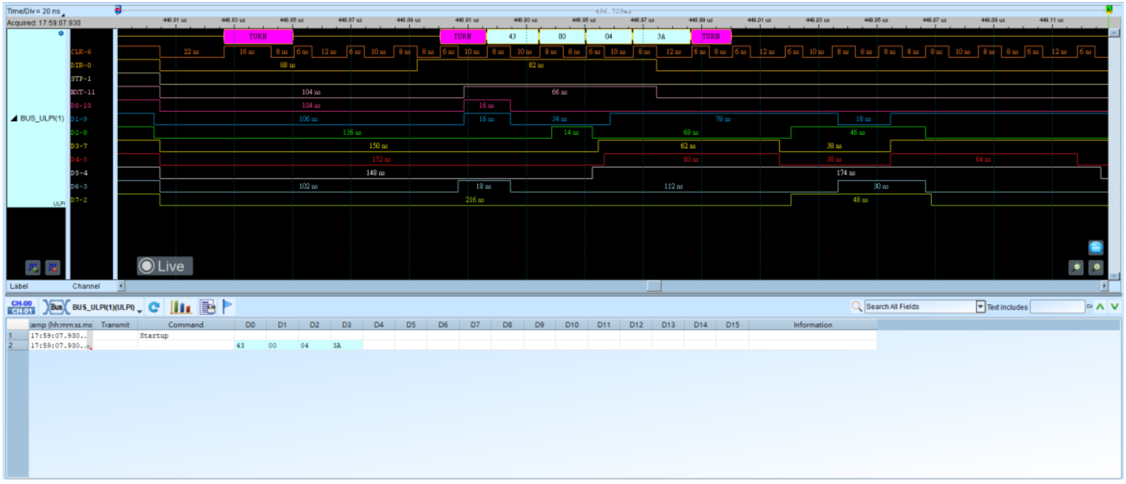
To: Buffer Tail

**Channel setting:** Set the ULPI channel. And you can use Quick Setting to quickly set the data pin.

**Decode:** Set the decoding method of ULPI, optional ULPI / USB (analog timing).

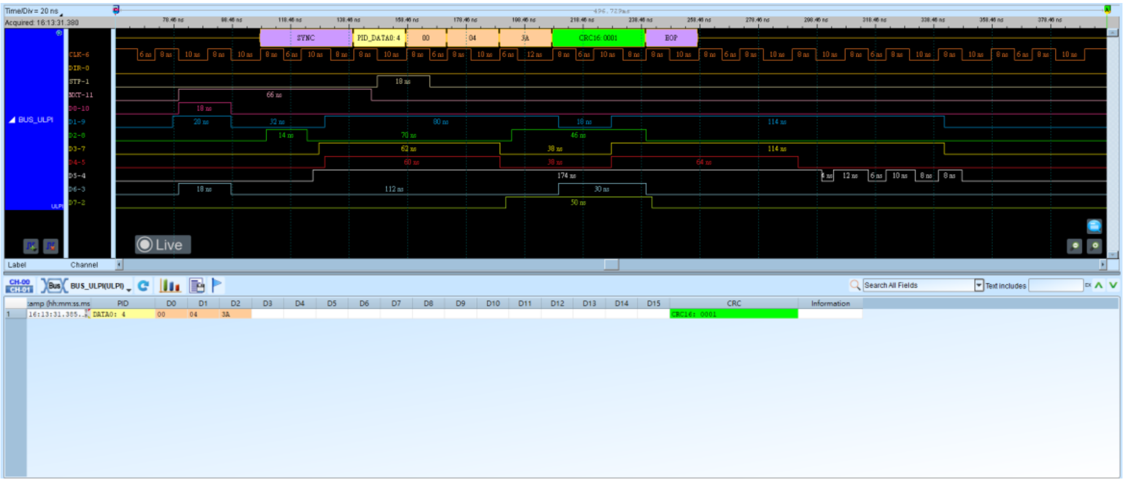
## Result

### ULPI



| amp (hh:mm:ss.ms | Transmit | Command | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 17:59:07.930.. |  | Startup |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 17:59:07.930.. |  |  | 43 | 00 | 04 | 3A |  |  |  |  |  |  |  |  |  |  |  |  |  |

### USB

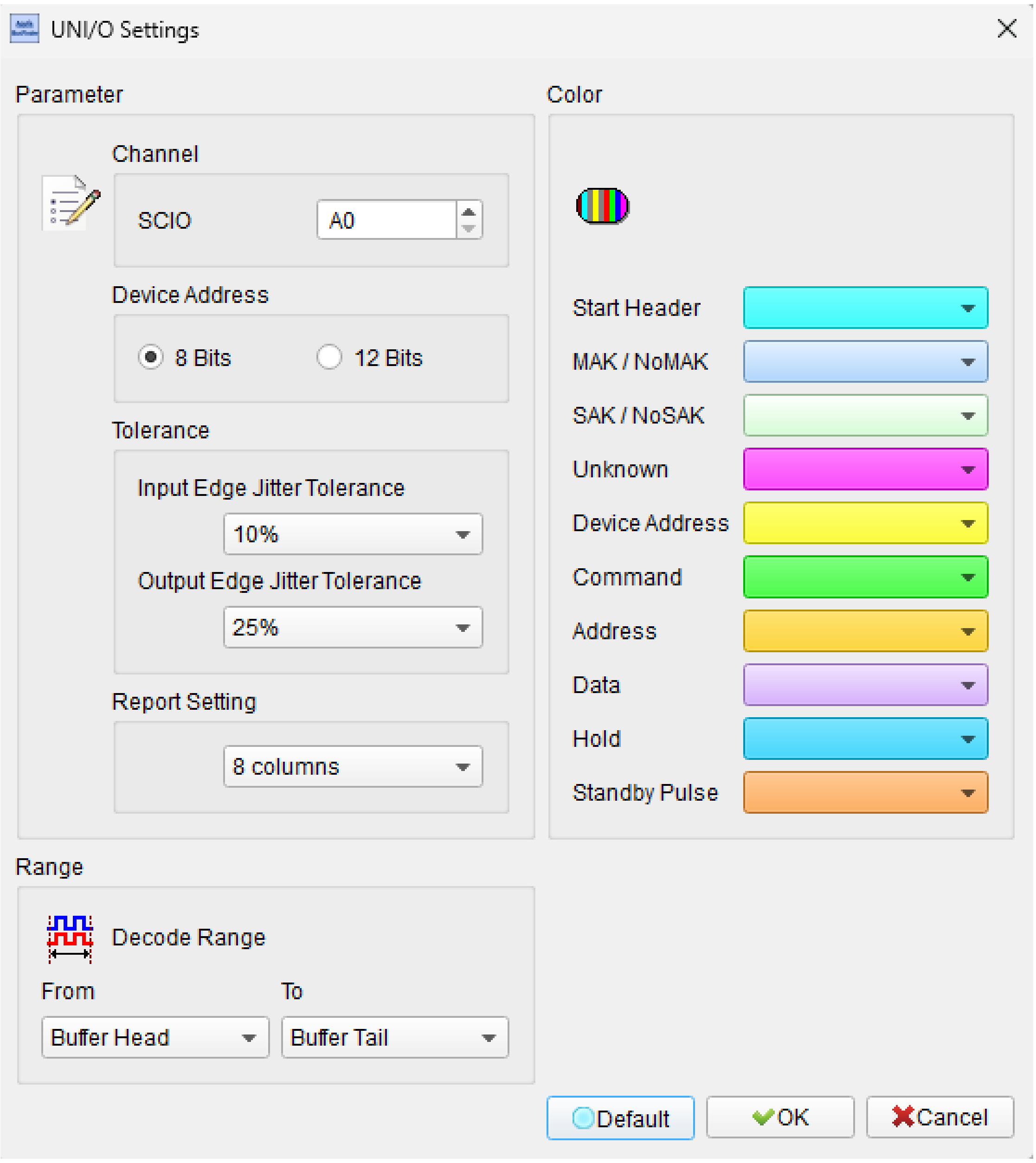


| amp (hh:mm:ss.ms | PID | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 | D15 | CRC | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 16:13:31.385.. | DATA0: 4 | 00 | 04 | 3A |  |  |  |  |  |  |  |  |  |  |  |  |  | CRC16: 0001 |  |

## UNI/O

UNI/O is a communication protocol developed by Microchip, primarily used in EEPROM applications. It was designed to meet the growing demand for fewer I/O pins in miniaturized embedded systems, while also providing a low-cost and easy-to-use single-wire bus solution. UNI/O utilizes Manchester Encoding and supports data transmission rates ranging from 10Kbps to 100Kbps

## Settings



UNI/O Settings

Parameter

Channel

SCIO: A0

Device Address

● 8 Bits ○ 12 Bits

Tolerance

Input Edge Jitter Tolerance: 10%

Output Edge Jitter Tolerance: 25%

Report Setting: 8 columns

Color

Start Header
MAK / NoMAK
SAK / NoSAK
Unknown
Device Address
Command
Address
Data
Hold
Standby Pulse

Range

Decode Range

From: Buffer Head

To: Buffer Tail

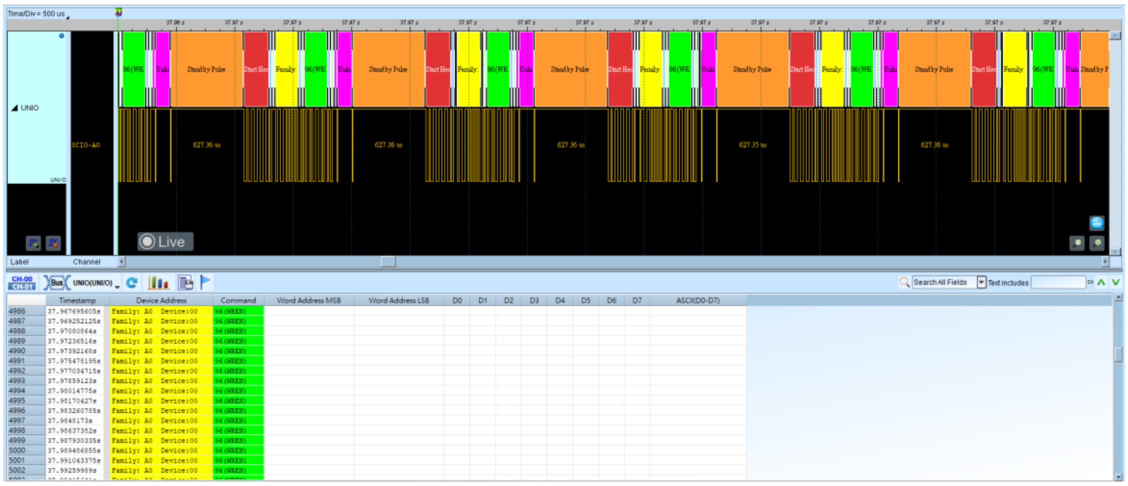
**Channel:** Show the selected channels (SCIO CH0).

**Device Address:** Set data bits for the device address.

**Tolerance:** Set the Input / Output Edge Jitter Tolerance, ±10% / ±25% default.

**Report Setting:** To show the data by 8 or 16 columns in the Report Window.

## Result

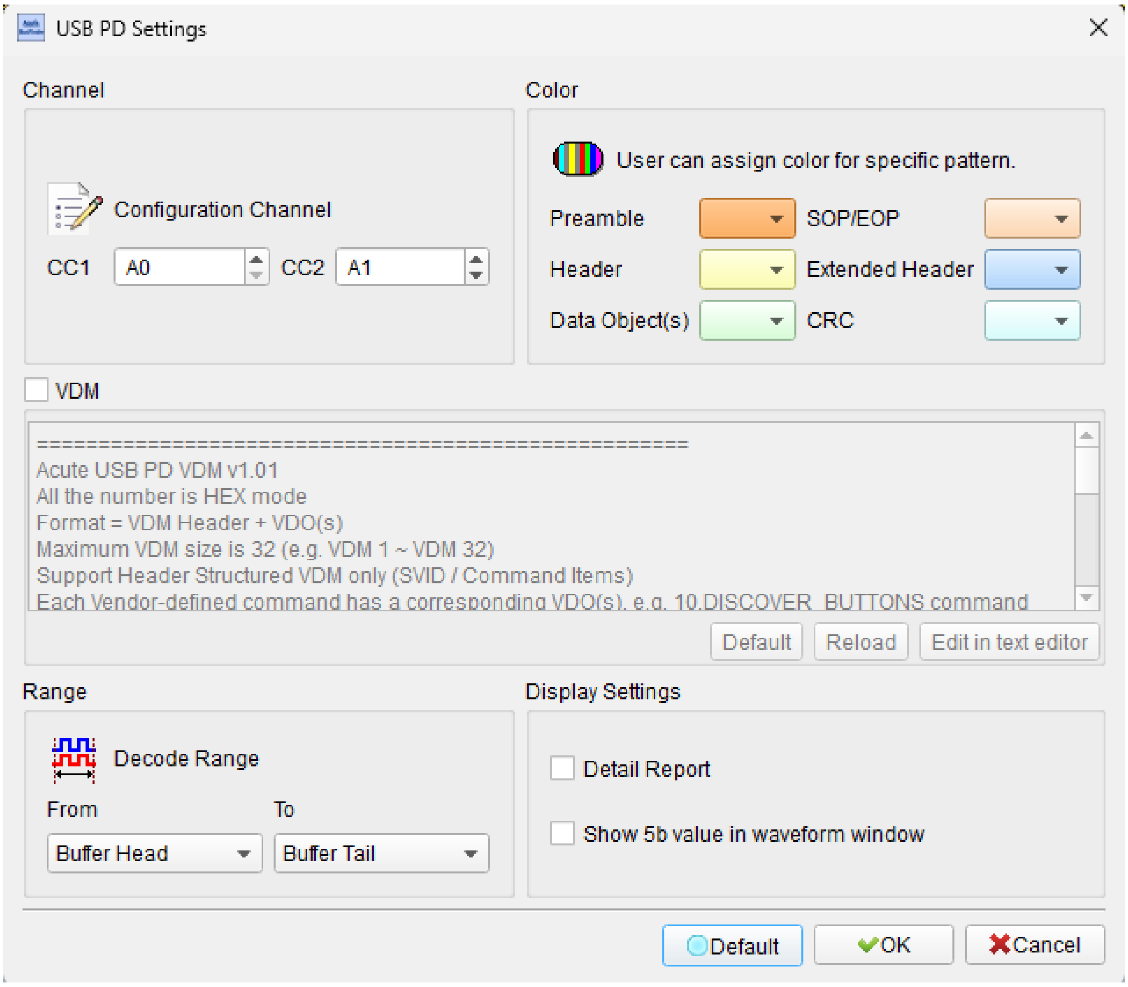


| | Timestamp | Device Address | Command | Word Address MSB | Word Address LSB | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII(D0-D7) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4986 | 37.967695605s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4987 | 37.969252125s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4988 | 37.97080864s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4989 | 37.97236516s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4990 | 37.97392168s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4991 | 37.975478195s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4992 | 37.977034715s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4993 | 37.97859123s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4994 | 37.98014775s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4995 | 37.98170427s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4996 | 37.983260785s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4997 | 37.9848173s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4998 | 37.98637382s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 4999 | 37.987930335s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 5000 | 37.989486855s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 5001 | 37.991043375s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |
| 5002 | 37.99259989s | Family: A0 Device:00 | 96(WREN) | | | | | | | | | | | |

## USB PD

USB PD (Power Delivery) 2.0/3.0 is the protocol based on BMC (Biphase Mark Coding) and can be applied to Laptops / Tablets / Mobile phones / Power banks or other devices with USB Type-C for power supplying or charging. The maximum power offered by USB Type-C can be 240W, and the users can charge the device by supporting USB Type-C connector, current version is Revision 3.1, Version 1.0.

## Settings



USB PD Settings

Channel

Configuration Channel

CC1 A0 CC2 A1

Color

User can assign color for specific pattern.

Preamble, SOP/EOP, Header, Extended Header, Data Object(s), CRC

□ VDM

```
====================================================
Acute USB PD VDM v1.01
All the number is HEX mode
Format = VDM Header + VDO(s)
Maximum VDM size is 32 (e.g. VDM 1 ~ VDM 32)
Support Header Structured VDM only (SVID / Command Items)
Each Vendor-defined command has a corresponding VDO(s), e.g. 10.DISCOVER_BUTTONS command
```

Range

Decode Range

From: Buffer Head To: Buffer Tail

Display Settings

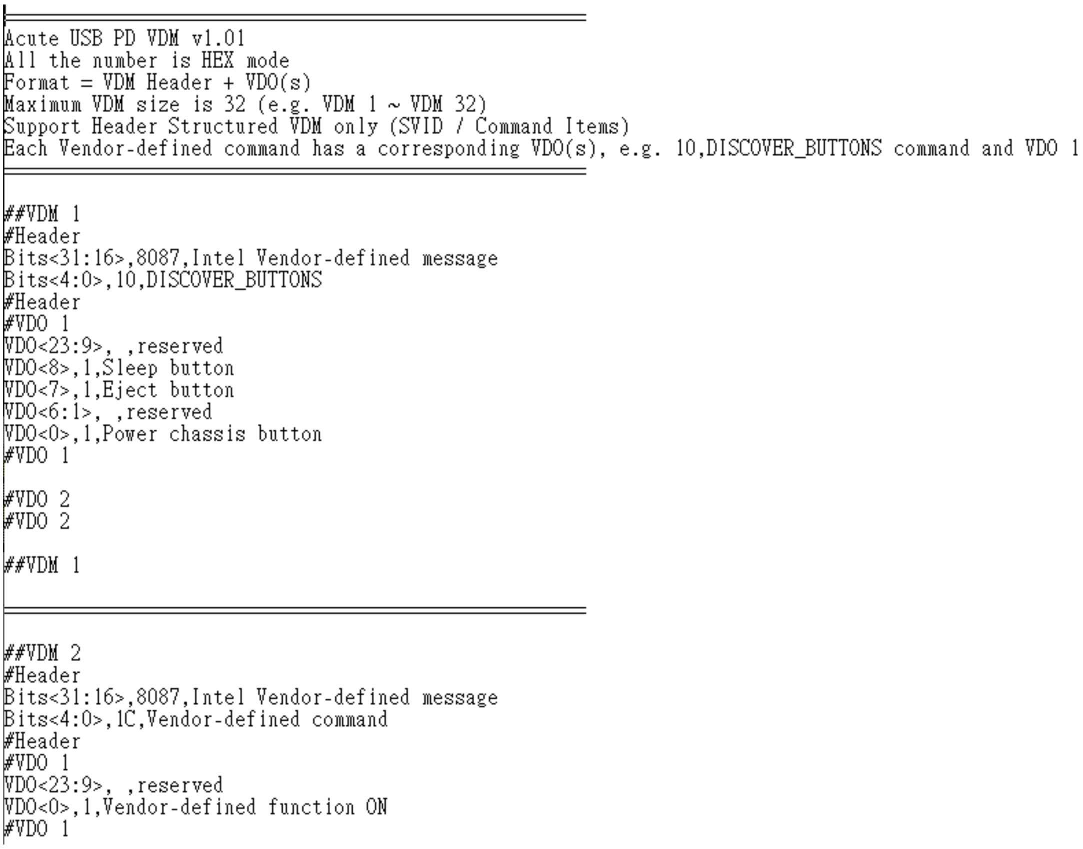
□ Detail Report

□ Show 5b value in waveform window

**Channel:** Set Configuration Channel (CC) CC1 & CC2.

**VDM:** Use the vendor defined message function when checked; users could define the SVID/Command of the Structured VDM by Edit/Refresh.

Content of the configure file as the following:



```
Acute USB PD VDM v1.01
All the number is HEX mode
Format = VDM Header + VDO(s)
Maximum VDM size is 32 (e.g. VDM 1 ~ VDM 32)
Support Header Structured VDM only (SVID / Command Items)
Each Vendor-defined command has a corresponding VDO(s), e.g. 10,DISCOVER_BUTTONS command and VDO 1

##VDM 1
#Header
Bits<31:16>,8087,Intel Vendor-defined message
Bits<4:0>,10,DISCOVER_BUTTONS
#Header
#VDO 1
VDO<23:9>, ,reserved
VDO<8>,1,Sleep button
VDO<7>,1,Eject button
VDO<6:1>, ,reserved
VDO<0>,1,Power chassis button
#VDO 1

#VDO 2
#VDO 2

##VDM 1

##VDM 2
#Header
Bits<31:16>,8087,Intel Vendor-defined message
Bits<4:0>,1C,Vendor-defined command
#Header
#VDO 1
VDO<23:9>, ,reserved
VDO<0>,1,Vendor-defined function ON
#VDO 1
```

VDM (Vendor defined message)

Maximum quantity: 32 (##VDM1 ~ ##VDM32)

The ##XXX at the beginning of each column is a keyword and don’t use it at other position in the file. Each VDM is composed of a header and its corresponding VDO (VDM Object), and the header is the first Data Object of VDM; the rest is VDO(s), which is the response message according to the Command.

The header part uses the keyword #Header to include the definition of SVID / Command:

SVID (Standard/Vendor ID):

Bits<31:16>,8087,Intel Vendor-defined message

When the value of Header bit 31 ~ bit 16 is 8087h, the Intel Vendor-defined message is displayed.

Command:

Bits<4:0>,10,DISCOVER_BUTTONS

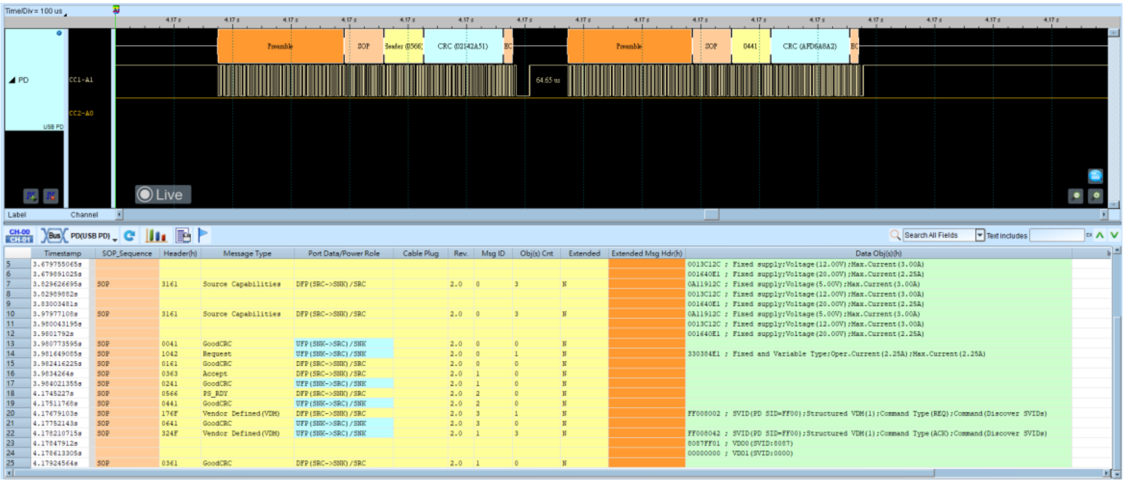
When the value of Header bit 4 ~ bit 0 is 10h, it’s command is DISCOVER_BUTTONS.

**Show 5b value in waveform window:** Show 5b value in the waveform.

**Details Report:** Show the details of bit parsing for the Data Obj(s) frame.

**Show 5b value in waveform window:** Show 4b or 5b value in the waveform.

**Result**



| | Timestamp | SOP_Sequence | Header(h) | Message Type | Port Data/Power Role | Cable Plug | Rev. | Msg ID | Obj(s) Cnt | Extended | Extended Msg Hdr(h) | Data Obj(s)(h) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5 | 3.679755065s | | | | | | | | | | | 0013C12C ; Fixed supply;Voltage(12.00V);Max.Current(3.00A) |
| 6 | 3.679891025s | | | | | | | | | | | 001640E1 ; Fixed supply;Voltage(20.00V);Max.Current(2.25A) |
| 7 | 3.829626695s | SOP | 3161 | Source Capabilities | DFP(SRC->SNK)/SRC | | 2.0 | 0 | 3 | N | | 0A11912C ; Fixed supply;Voltage(5.00V);Max.Current(3.00A) |
| 8 | 3.82989882s | | | | | | | | | | | 0013C12C ; Fixed supply;Voltage(12.00V);Max.Current(3.00A) |
| 9 | 3.83003481s | | | | | | | | | | | 001640E1 ; Fixed supply;Voltage(20.00V);Max.Current(2.25A) |
| 10 | 3.97977108s | SOP | 3161 | Source Capabilities | DFP(SRC->SNK)/SRC | | 2.0 | 0 | 3 | N | | 0A11912C ; Fixed supply;Voltage(5.00V);Max.Current(3.00A) |
| 11 | 3.980043195s | | | | | | | | | | | 0013C12C ; Fixed supply;Voltage(12.00V);Max.Current(3.00A) |
| 12 | 3.9801792s | | | | | | | | | | | 001640E1 ; Fixed supply;Voltage(20.00V);Max.Current(2.25A) |
| 13 | 3.980773595s | SOP | 0041 | GoodCRC | UFP(SNK->SRC)/SNK | | 2.0 | 0 | 0 | N | | |
| 14 | 3.981649085s | SOP | 1042 | Request | UFP(SNK->SRC)/SNK | | 2.0 | 0 | 1 | N | | 330384E1 ; Fixed and Variable Type;Oper.Current(2.25A);Max.Current(2.25A) |
| 15 | 3.982416225s | SOP | 0161 | GoodCRC | DFP(SRC->SNK)/SRC | | 2.0 | 0 | 0 | N | | |
| 16 | 3.9834264s | SOP | 0363 | Accept | DFP(SRC->SNK)/SRC | | 2.0 | 1 | 0 | N | | |
| 17 | 3.984021355s | SOP | 0241 | GoodCRC | UFP(SNK->SRC)/SNK | | 2.0 | 1 | 0 | N | | |
| 18 | 4.1745227s | SOP | 0566 | PS_RDY | DFP(SRC->SNK)/SRC | | 2.0 | 2 | 0 | N | | |
| 19 | 4.17511768s | SOP | 0441 | GoodCRC | UFP(SNK->SRC)/SNK | | 2.0 | 2 | 0 | N | | |
| 20 | 4.17679103s | SOP | 176F | Vendor Defined(VDM) | DFP(SRC->SNK)/SRC | | 2.0 | 3 | 1 | N | | FF008002 ; SVID(PD SID=FF00);Structured VDM(1);Command Type(REQ);Command(Discover SVIDs) |
| 21 | 4.17752143s | SOP | 0641 | GoodCRC | UFP(SNK->SRC)/SNK | | 2.0 | 3 | 0 | N | | |
| 22 | 4.178210715s | SOP | 324F | Vendor Defined(VDM) | UFP(SNK->SRC)/SNK | | 2.0 | 1 | 3 | N | | FF008042 ; SVID(PD SID=FF00);Structured VDM(1);Command Type(ACK);Command(Discover SVIDs) |
| 23 | 4.17847912s | | | | | | | | | | | 8087FF01 ; VDO0(SVID:8087) |
| 24 | 4.178613305s | | | | | | | | | | | 00000000 ; VDO1(SVID:0000) |
| 25 | 4.17924564s | SOP | 0361 | GoodCRC | DFP(SRC->SNK)/SRC | | 2.0 | 1 | 0 | N | | |

**Accessories:**

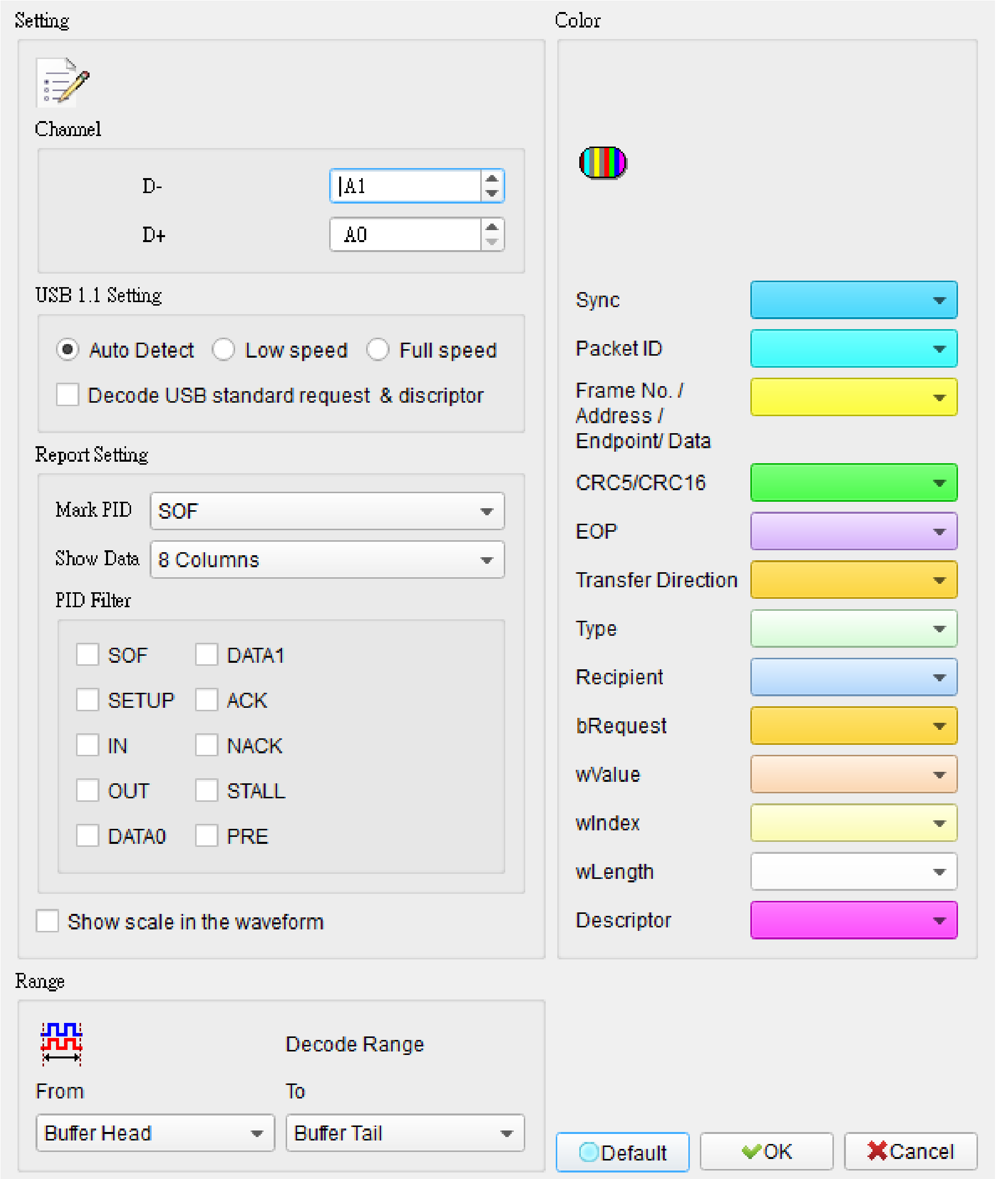
https://www.acute.com.tw/en/product/detail142

## USB1.1

USB (Universal Serial Bus), known as the "Universal Serial Bus", was initially developed by seven companies: Intel, Microsoft, National Semiconductor, Compaq, Northern Telecom, NEC, and AT&T. The development of USB began in 1994, with version 1.0, followed by version 1.1 in 1998, and later USB 2.0 in 2000. The data transfer speed evolved from 12 Mbps in USB 1.1 to 480 Mbps in USB 2.0.

In the USB protocol, communication between the host and device is primarily established through two differential signal lines, D+ and D-.

## Settings



Setting

Channel

D- A1

D+ A0

USB 1.1 Setting

● Auto Detect ○ Low speed ○ Full speed

□ Decode USB standard request & discriptor

Report Setting

Mark PID: SOF

Show Data: 8 Columns

PID Filter

□ SOF □ DATA1
□ SETUP □ ACK
□ IN □ NACK
□ OUT □ STALL
□ DATA0 □ PRE

□ Show scale in the waveform

Color

| | |
|---|---|
| Sync | 🟦 |
| Packet ID | 🟦 |
| Frame No. / Address / Endpoint/ Data | 🟨 |
| CRC5/CRC16 | 🟩 |
| EOP | 🟪 |
| Transfer Direction | 🟧 |
| Type | 🟩 |
| Recipient | 🟦 |
| bRequest | 🟧 |
| wValue | 🟧 |
| wIndex | 🟨 |
| wLength | ⬜ |
| Descriptor | 🟪 |

Range

Decode Range

From: Buffer Head

To: Buffer Tail

**D+:** D+ line for USB 1.1 data transmission.

**D-:** D- line for USB 1.1 data transmission.

**USB1.1 Settings:** Configure the USB 1.1 signal as Low-Speed or Full-Speed, and determine whether to decode USB standard requests and descriptors.

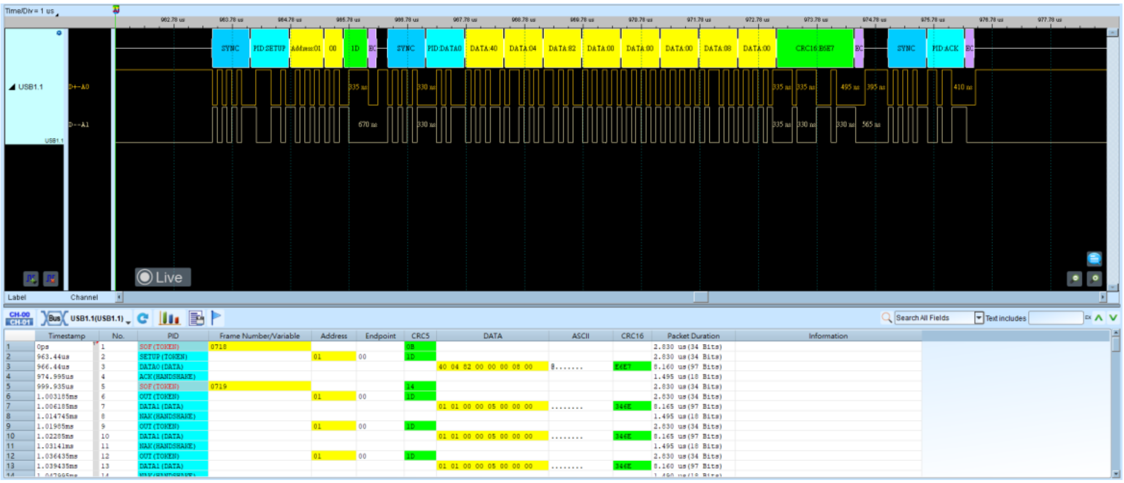
**Mark PID**: Allows marking selected PID types with specific colors in the report window.

**Show Data:** Sets the data display in the report window to 8 columns or 16 columns.

**PID Filter:** Allows hiding specific data from display.

**Show Scale in Waveform:** Displays scale markings on the waveform.

Result.



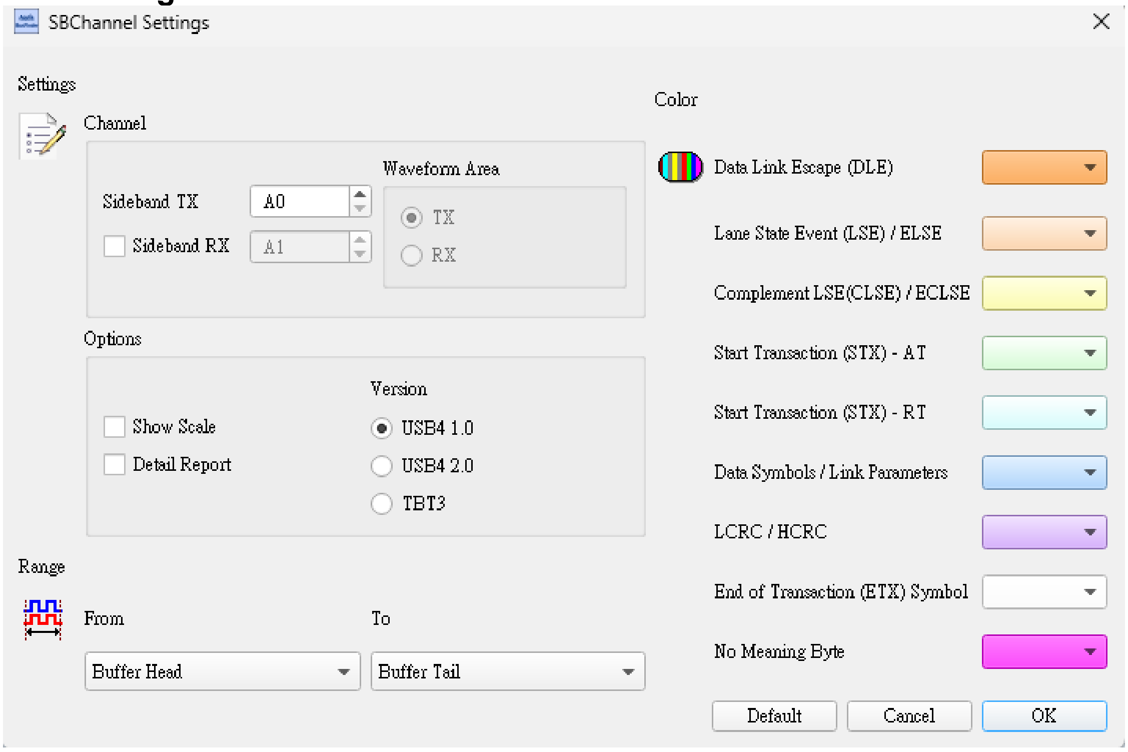
| | Timestamp | No. | PID | Frame Number/Variable | Address | Endpoint | CRC5 | DATA | ASCII | CRC16 | Packet Duration | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0ps | 1 | SOF(TOKEN) | 0718 | | | 0B | | | | 2.830 us(34 Bits) | |
| 2 | 963.44us | 2 | SETUP(TOKEN) | | 01 | 00 | 1D | | | | 2.830 us(34 Bits) | |
| 3 | 966.44us | 3 | DATA0(DATA) | | | | | 40 04 82 00 00 00 08 00 | @....... | E6E7 | 8.160 us(97 Bits) | |
| 4 | 974.995us | 4 | ACK(HANDSHAKE) | | | | | | | | 1.495 us(18 Bits) | |
| 5 | 999.935us | 5 | SOF(TOKEN) | 0719 | | | 14 | | | | 2.830 us(34 Bits) | |
| 6 | 1.003185ms | 6 | OUT(TOKEN) | | 01 | 00 | 1D | | | | 2.830 us(34 Bits) | |
| 7 | 1.006185ms | 7 | DATA1(DATA) | | | | | 01 01 00 00 05 00 00 00 | ........ | 346E | 8.165 us(97 Bits) | |
| 8 | 1.014745ms | 8 | NAK(HANDSHAKE) | | | | | | | | 1.495 us(18 Bits) | |
| 9 | 1.01985ms | 9 | OUT(TOKEN) | | 01 | 00 | 1D | | | | 2.830 us(34 Bits) | |
| 10 | 1.02285ms | 10 | DATA1(DATA) | | | | | 01 01 00 00 05 00 00 00 | ........ | 346E | 8.165 us(97 Bits) | |
| 11 | 1.03141ms | 11 | NAK(HANDSHAKE) | | | | | | | | 1.495 us(18 Bits) | |
| 12 | 1.036435ms | 12 | OUT(TOKEN) | | 01 | 00 | 1D | | | | 2.830 us(34 Bits) | |
| 13 | 1.039435ms | 13 | DATA1(DATA) | | | | | 01 01 00 00 05 00 00 00 | ........ | 346E | 8.160 us(97 Bits) | |
| 14 | 1.047995ms | 14 | NAK(HANDSHAKE) | | | | | | | | 1.490 us(18 Bits) | |

## USB4/TBT3 SB

USB4 can simultaneously transmit DisplayPort video and PCIE signals through Intel Tunneled technology, and also support PD (Power Delivery) fast charging technology.

USB4 is backward compatible with USB 2.0 and USB 3.2 Gen1/Gen2 and supports Thunderbolt 3/4. The sideband channel (SB) was originally defined as a channel for video protocol communication after entering the Alt-Mode (Alternate Mode) in USB 3.2 (for example: DP Alt-Mode transmits AUX signals through the Sideband channel...etc.). In the USB4 specification, new features of Sideband channel are added to confirm whether the USB4 interface is connected, start and close the channel, initialize the channel, and enter or leave the sleep mode. Thunderbolt Alt-Mode will be enabled when connecting to the device via Thunderbolt 3.

### 1. Settings:



SBChannel Settings

Settings

Channel

Sideband TX: A0

□ Sideband RX: A1

Waveform Area: ● TX ○ RX

Options

□ Show Scale

□ Detail Report

Version: ● USB4 1.0 ○ USB4 2.0 ○ TBT3

Range

From: Buffer Head

To: Buffer Tail

Color

Data Link Escape (DLE)

Lane State Event (LSE) / ELSE

Complement LSE(CLSE) / ECLSE

Start Transaction (STX) - AT

Start Transaction (STX) - RT

Data Symbols / Link Parameters

LCRC / HCRC

End of Transaction (ETX) Symbol

No Meaning Byte

### a. Channel:

**Sideband TX:** Show the selected channel.

**Sideband RX:** It is checked by default, and the dual-channel analysis mode can be enabled by checking Sideband RX.

**Data Convert:** TX is checked by default, and the TX/RX channel can be selected for analysis in the data convert field.

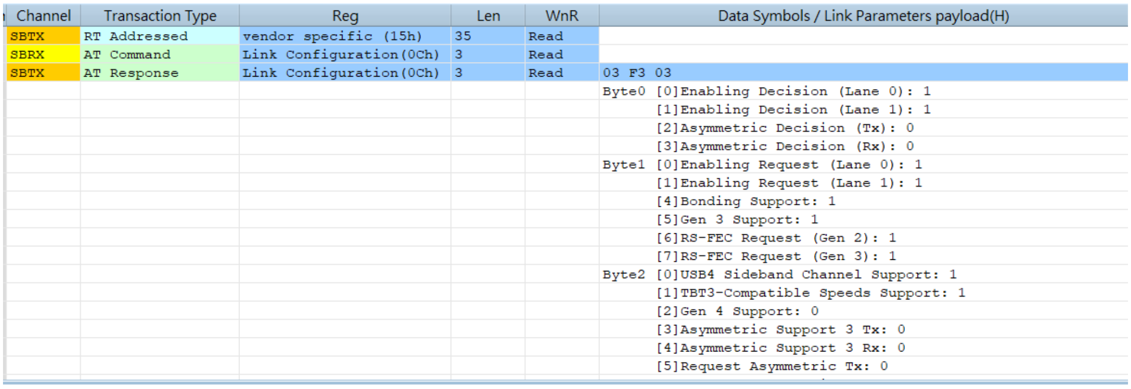
**Show on Report:** TX and RX is checked by default. Choose which channel to display in the report area, and both channels can be displayed at the same time.

### b. Others:

**Version:** USB4 2.0 is checked by default, and different versions can be selected for USB4/TBT3 SB signal analysis.

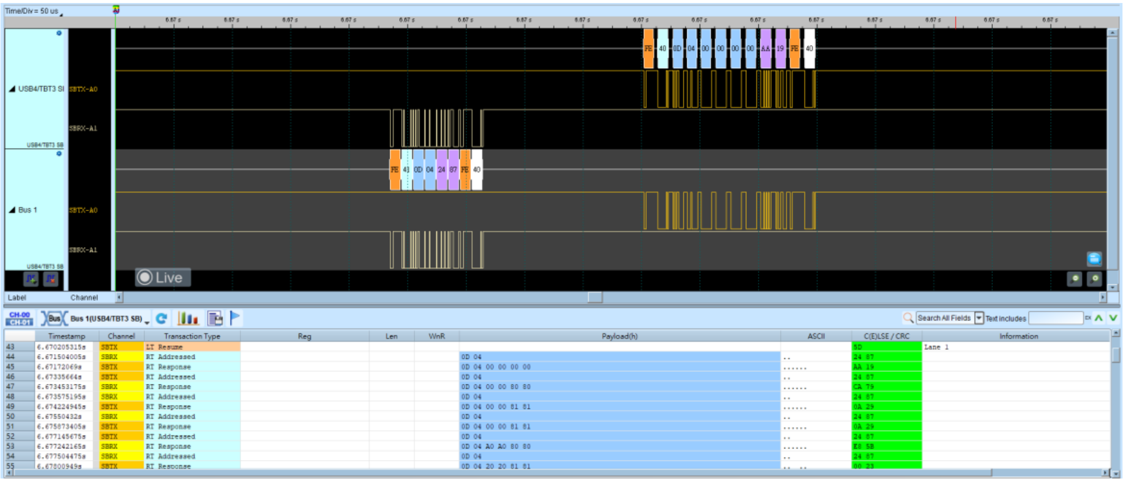
**Show scale on data convert:** The default is off. Show the scale on data convert field.

**Show data detail on report:** The default is off. If checked, additional data information will show on report as shown below.



| Channel | Transaction Type | Reg | Len | WnR | Data Symbols / Link Parameters payload(H) |
|---|---|---|---|---|---|
| SBTX | RT Addressed | vendor specific (15h) | 35 | Read | |
| SBRX | AT Command | Link Configuration(0Ch) | 3 | Read | |
| SBTX | AT Response | Link Configuration(0Ch) | 3 | Read | 03 F3 03 |
| | | | | | Byte0 [0]Enabling Decision (Lane 0): 1 |
| | | | | | [1]Enabling Decision (Lane 1): 1 |
| | | | | | [2]Asymmetric Decision (Tx): 0 |
| | | | | | [3]Asymmetric Decision (Rx): 0 |
| | | | | | Byte1 [0]Enabling Request (Lane 0): 1 |
| | | | | | [1]Enabling Request (Lane 1): 1 |
| | | | | | [4]Bonding Support: 1 |
| | | | | | [5]Gen 3 Support: 1 |
| | | | | | [6]RS-FEC Request (Gen 2): 1 |
| | | | | | [7]RS-FEC Request (Gen 3): 1 |
| | | | | | Byte2 [0]USB4 Sideband Channel Support: 1 |
| | | | | | [1]TBT3-Compatible Speeds Support: 1 |
| | | | | | [2]Gen 4 Support: 0 |
| | | | | | [3]Asymmetric Support 3 Tx: 0 |
| | | | | | [4]Asymmetric Support 3 Rx: 0 |
| | | | | | [5]Request Asymmetric Tx: 0 |

## 2. Result:

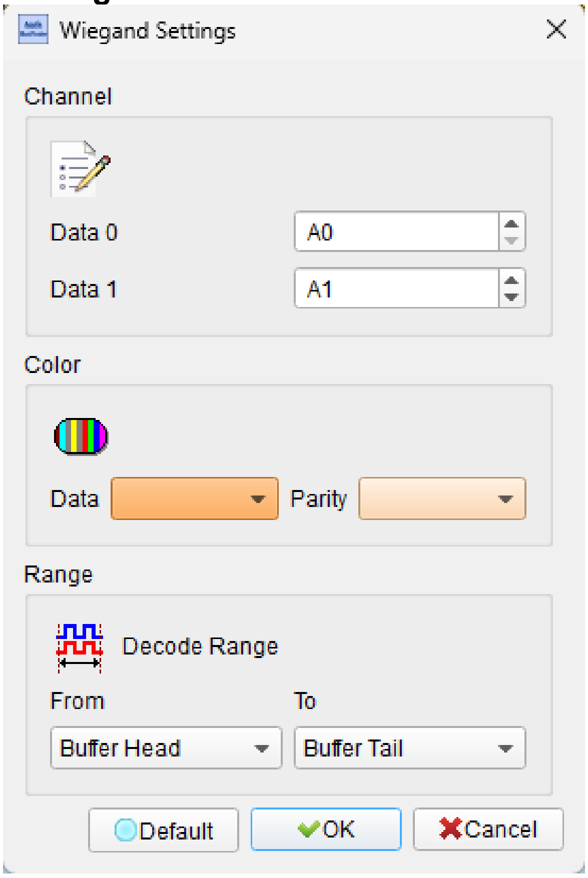


|  | Timestamp | Channel | Transaction Type | Reg | Len | WnR | Payload(h) | ASCII | C(E)LSE / CRC | Information |
|---|---|---|---|---|---|---|---|---|---|---|
| 43 | 6.670205315s | SBTX | LT Resume |  |  |  |  |  | 5D | Lane 1 |
| 44 | 6.671504005s | SBRX | RT Addressed |  |  |  | 0D 04 | .. | 24 87 |  |
| 45 | 6.67172069s | SBTX | RT Response |  |  |  | 0D 04 00 00 00 00 | ...... | AA 19 |  |
| 46 | 6.67335664s | SBTX | RT Addressed |  |  |  | 0D 04 | .. | 24 87 |  |
| 47 | 6.673453175s | SBRX | RT Response |  |  |  | 0D 04 00 00 80 80 | ...... | CA 79 |  |
| 48 | 6.673575195s | SBRX | RT Addressed |  |  |  | 0D 04 | .. | 24 87 |  |
| 49 | 6.674224945s | SBTX | RT Response |  |  |  | 0D 04 00 00 81 81 | ...... | 0A 29 |  |
| 50 | 6.67550432s | SBRX | RT Addressed |  |  |  | 0D 04 | .. | 24 87 |  |
| 51 | 6.675873405s | SBTX | RT Response |  |  |  | 0D 04 00 00 81 81 | ...... | 0A 29 |  |
| 52 | 6.677145675s | SBTX | RT Addressed |  |  |  | 0D 04 | .. | 24 87 |  |
| 53 | 6.677242165s | SBRX | RT Response |  |  |  | 0D 04 A0 A0 80 80 | ...... | E8 5B |  |
| 54 | 6.677504475s | SBRX | RT Addressed |  |  |  | 0D 04 | .. | 24 87 |  |
| 55 | 6.67800949s | SBTX | RT Response |  |  |  | 0D 04 20 20 81 81 | .. .. | 00 23 |  |

## Wiegand

It is commonly used to connect a card swipe mechanism to the rest of an electronic entry system.

### Settings



Wiegand Settings

Channel

Data 0: A0

Data 1: A1

Color

Data

Parity

Range

Decode Range

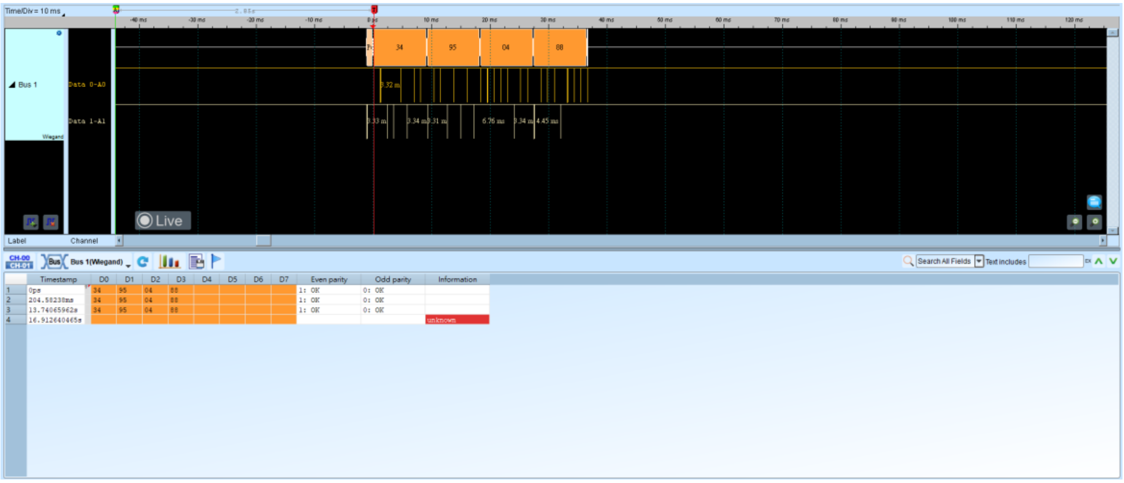
From: Buffer Head

To: Buffer Tail

**Data 0:** Wiegand data 0。

**Data 1:** Wiegand data 1。

### Result



| | Timestamp | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | Even parity | Odd parity | Information |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0ps | 34 | 95 | 04 | 88 | | | | | 1: OK | 0: OK | |
| 2 | 204.58238ms | 34 | 95 | 04 | 88 | | | | | 1: OK | 0: OK | |
| 3 | 13.74065962s | 34 | 95 | 04 | 88 | | | | | 1: OK | 0: OK | |
| 4 | 16.912640465s | | | | | | | | | | | unknown |

# Chapter 2 Bus Trigger

## Bus Trigger

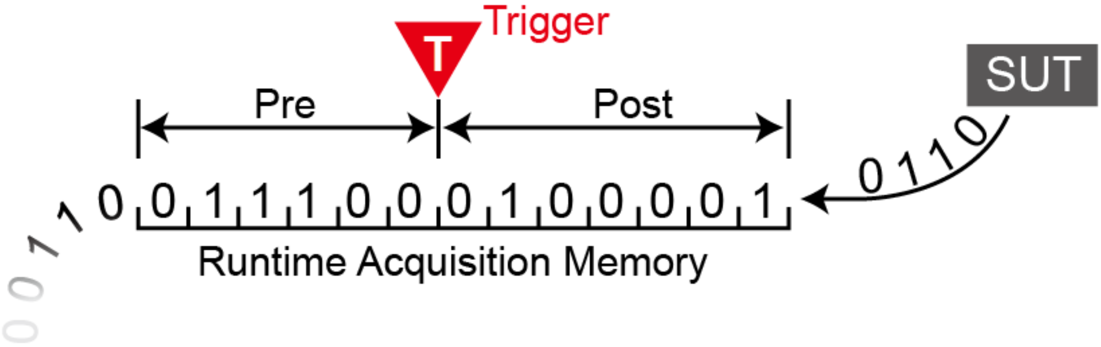
### Introduction of trigger

Trigger function is to utilize the logic analyzer's hardware circuitry to check whether the signal to be tested meets the trigger conditions within a limited period of time by using parallel processing techniques, and then carry out the signal acquisition work. Ideal logic analyzer trigger function, in addition to the basic must be accurate, but also as much as possible can be varied. The hardware of logic analyzer takes the signal acquisition function and then performs the signal acquisition work.
The hardware of logic analyzer take the parallel processing technology to check the signal.

### Trigger Mode

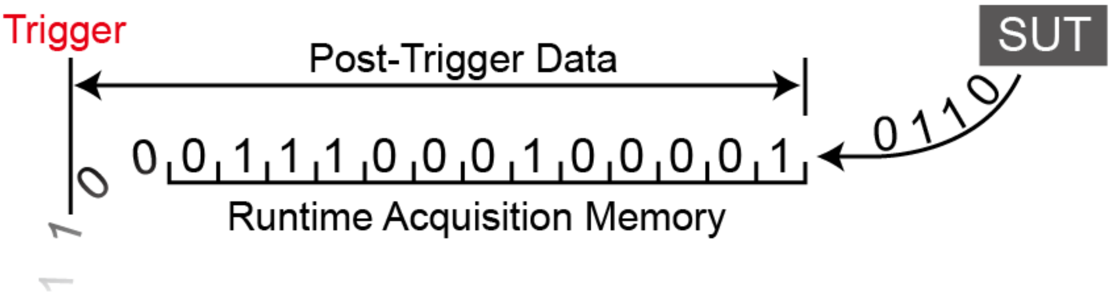
1. Pre-Trigger

   In certain applications, when the user wants to capture signals before the trigger point, the Pre-Trigger function must be enabled. After pressing the 'Start Capture' button, the logic analyzer will wait until the data fills the memory from the beginning of the buffer to the trigger cursor before allowing the trigger circuit to start operating (it starts operating, not issuing the trigger signal). Therefore, before the logic analyzer has filled the data between the buffer and the trigger cursor, any signal that meets the trigger condition will not cause the trigger circuit to issue a trigger signal.



2. Post-Trigger

This is the most basic triggering method. After pressing the 'Start Capture' button, the logic analyzer will wait for the trigger to occur and then start capturing data from the position specified by the trigger cursor. Once the memory is fully filled with data, the capture will stop.



3. Delay-Trigger

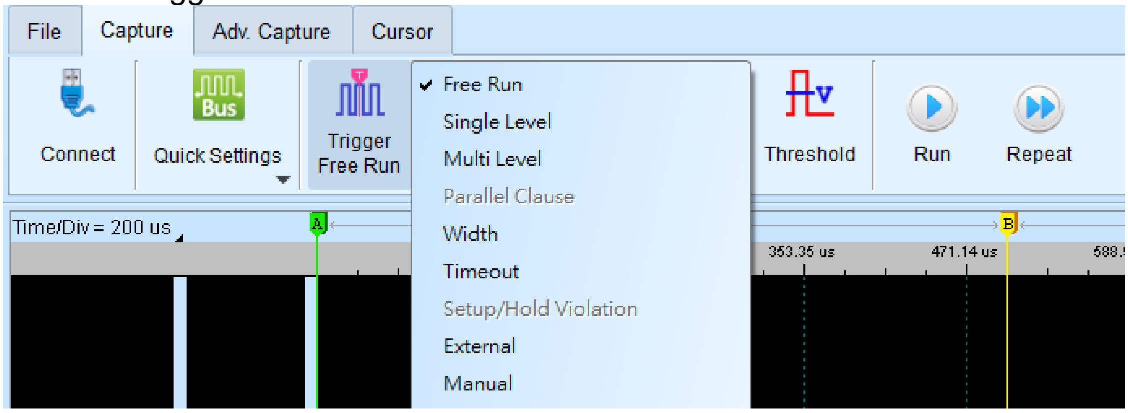
   In certain applications, when the user wants to capture signals after the trigger point and with a delay, the trigger delay function can be used to set the desired delay time. After the signal capture is successful, the trigger cursor will stop at the position where the data capture started.

4. Pass Count

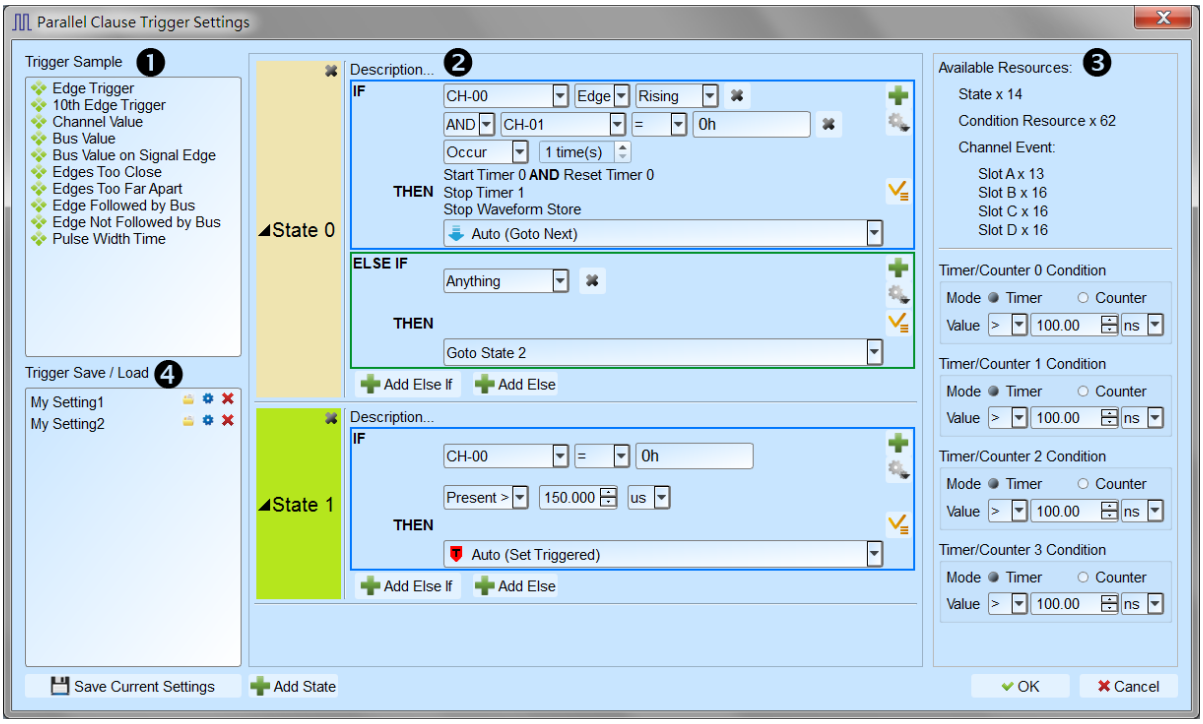
   Pass Count represents the number of times the set trigger parameters should be ignored. Under normal conditions, Pass Count is set to 0, which means data capture will start as soon as the trigger parameters are met. If set to N times, it means the trigger parameters must be met N+1 times before data capture begins. The maximum value of Pass Count will be automatically adjusted based on the specific model.

5. Choose to Trigger

   Click on 'Trigger' in the toolbar and select the desired condition to use.



## Parallel Clause Trigger



State 0

IF CH-00 Edge Rising
AND CH-01 = 0h
Occur 1 time(s)
THEN Start Timer 0 AND Reset Timer 0
Stop Timer 1
Stop Waveform Store
Auto (Goto Next)

ELSE IF Anything
THEN Goto State 2

State 1

IF CH-00 = 0h
Present > 150.000 us
THEN Auto (Set Triggered)

Available Resources:
State x 14
Condition Resource x 62
Channel Event:
Slot A x 13
Slot B x 16
Slot C x 16
Slot D x 16

Timer/Counter 0 Condition
Mode ● Timer ○ Counter
Value > 100.00 ns

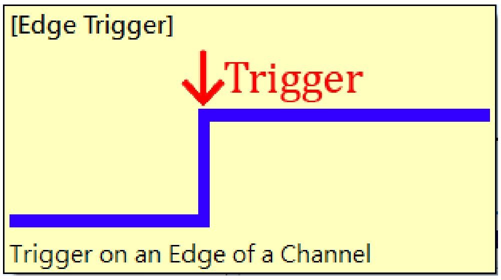
Timer/Counter 1 Condition
Mode ● Timer ○ Counter
Value > 100.00 ns

Timer/Counter 2 Condition
Mode ● Timer ○ Counter
Value > 100.00 ns

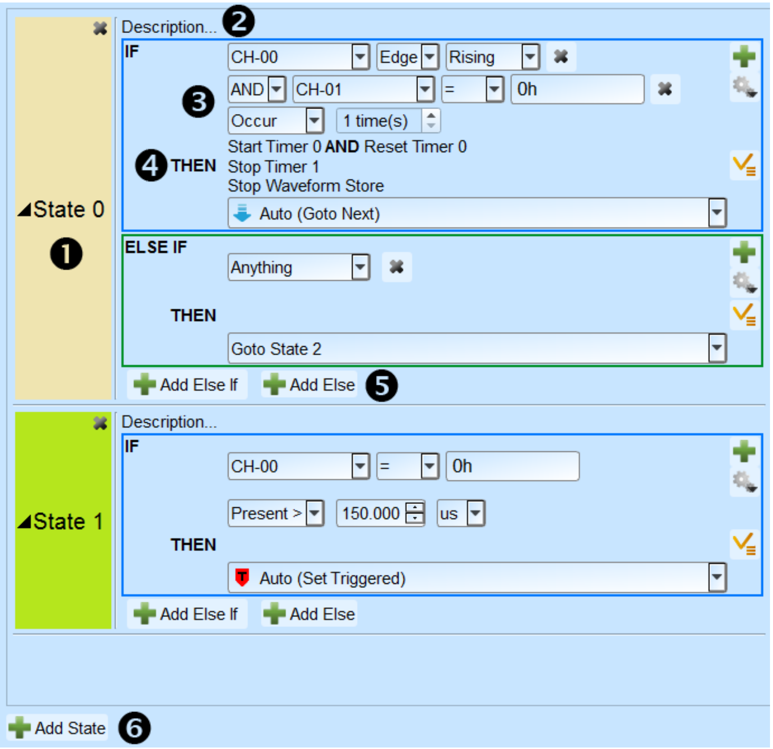
Timer/Counter 3 Condition
Mode ● Timer ○ Counter
Value > 100.00 ns

Parallel Clause Trigger feature provides 16 States, 64 Condition Resources and 4 Timer/Counter conditions(1) to help user locate the target waveform, it is also available to set State branch control flow for each IF Clause and decide whether to store waveform or not.

1. Trigger Sample: Provides several Trigger Samples for user’s reference, users may also combine multiple trigger samples together for more complex trigger condition. Holding mouse cursor on each Sample for a while will pop up the sample description with schematic picture.



2. Trigger condition settings:



State 0
Description...
IF CH-00 Edge Rising
AND CH-01 = 0h
Occur 1 time(s)
THEN Start Timer 0 AND Reset Timer 0
Stop Timer 1
Stop Waveform Store
Auto (Goto Next)
ELSE IF Anything
THEN Goto State 2

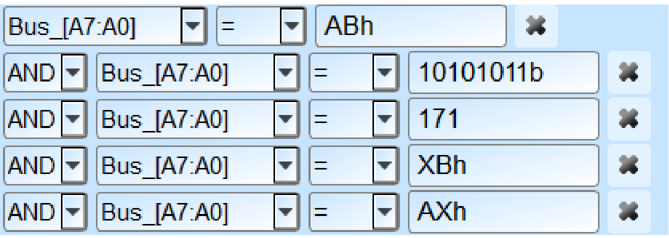
State 1
Description...
IF CH-00 = 0h
Present > 150.000 us
THEN Auto (Set Triggered)

① State button: Click to switch between Text Read Only mode and Edit mode.

② State description: Click to edit a short user description for the State, maximum input text length is 80 characters.

③ IF Clause: Setup trigger conditions for Label selected in waveform area, and it is also available to setup AND/OR logic combination for multiple IF conditions.

i. Channel Logic/Edge/Pattern compare: Specify value or edge condition for each channel label, it is also available to input X as don’t care value for the comparison. Input value ended with h for hexadecimal value; input value ended with b for binary value; input value not ended with b or h for decimal value.



| | | | | |
|---|---|---|---|---|
| | Bus_[A7:A0] | = | ABh | ✖ |
| AND | Bus_[A7:A0] | = | 10101011b | ✖ |
| AND | Bus_[A7:A0] | = | 171 | ✖ |
| AND | Bus_[A7:A0] | = | XBh | ✖ |
| AND | Bus_[A7:A0] | = | AXh | ✖ |

ii. Timer/Counter condition check: Check Timer/Counter conditions, condition will be set as True if the Timer/Counter value is matched, otherwise the condition is be set as False.

iii. Occur and Present time check: Additional Occur times check or Present time check when the all conditions are set to True in the IF Clause.

iv. Control buttons

➕ Add new condition: Click to add new AND/OR IF condition, the new IF condition will consume 1 Condition Resource.

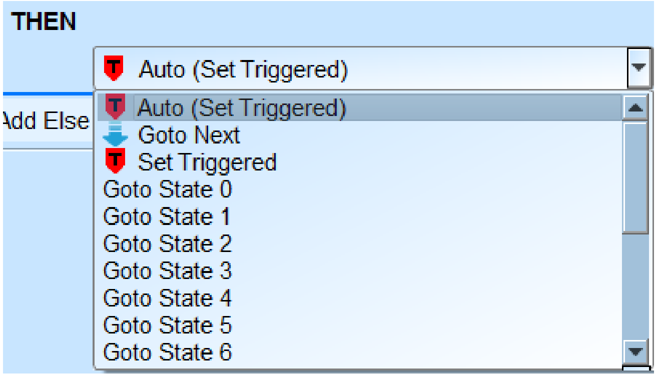
⚙ Advanced control button, including:

⚑ Get Value From: Set IF condition value by all Label value from selected Cursor position in the waveform area.

⎘ Copy: Copy all IF conditions of current IF Clause.

📋 Paste: Paste the copied IF conditions into current IF Clause.

④ THEN Branch: Select the Branch or Trigger control flow when the IF Clause is matched (2). When select Auto, the Branch or Trigger control will be determined by current State order in the setting, the Action will be set as Trigger if current State is the last State in the setting; the Action will be set as Goto Next State if current State is NOT the last State in the setting.



THEN

Auto (Set Triggered)

\dd Else

- Auto (Set Triggered)
- Goto Next
- Set Triggered
- Goto State 0
- Goto State 1
- Goto State 2
- Goto State 3
- Goto State 4
- Goto State 5
- Goto State 6

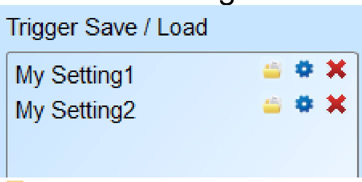
✓ Additional Then action: Setup Timer/Counter Start/Stop/Reset actions or Waveform Store Suspend/Resume for each IF Clause when condition matched.

⑤ Add new ELSE IF / ELSE Clause: Click to add new ELSE IF / ELSE Clause, each clause condition will be checked from top to bottom, and perform the specified Action and Branch if the clause condition is matched. New clause conditions will consume Condition Resources. ELSE Clause condition will be set as "Anything" and Branch to current State by default if the user didn't assign ELSE Clause.

⑥ Add new State: Click to add new State Clause, the new State Clause will consume both State and Condition Resources.

3. Available Resources and Timer/Counter settings: Display remain available resource number and Timer/Counter settings.
   Timer / Counter: Parallel Clause trigger provides 4 independent Timer/Counter resources, each resource could be selected to work in Timer or Counter mode. The minimum Timer interval is 12 Sample clock intervals (i.e. 60ns when working in 200MHz sampling mode), and the maximum Timer interval is 0x3FFFFFFF (i.e. 5s when working in 200MHz sampling mode). The minimum Counter value is 1, and the maximum Counter value is 0x3FFFFFFF.

4. Triger Save/Load: Users may save their current trigger settings with maximum 20 characters description, or load previous saved trigger settings from the list, the saved settings will be stored into Working Directory with file name PClauseUserSetting.aqr. Copy and share the .aqr file in order to share the settings to other user.



Trigger Save / Load

My Setting1
My Setting2

📁 Load selected Trigger settings, it's also available to select 🔄 to overwrite current Trigger settings, or select ✚ to append the selected settings to the end of current Trigger settings.

⚙ Edit the display name for the selected Trigger settings.

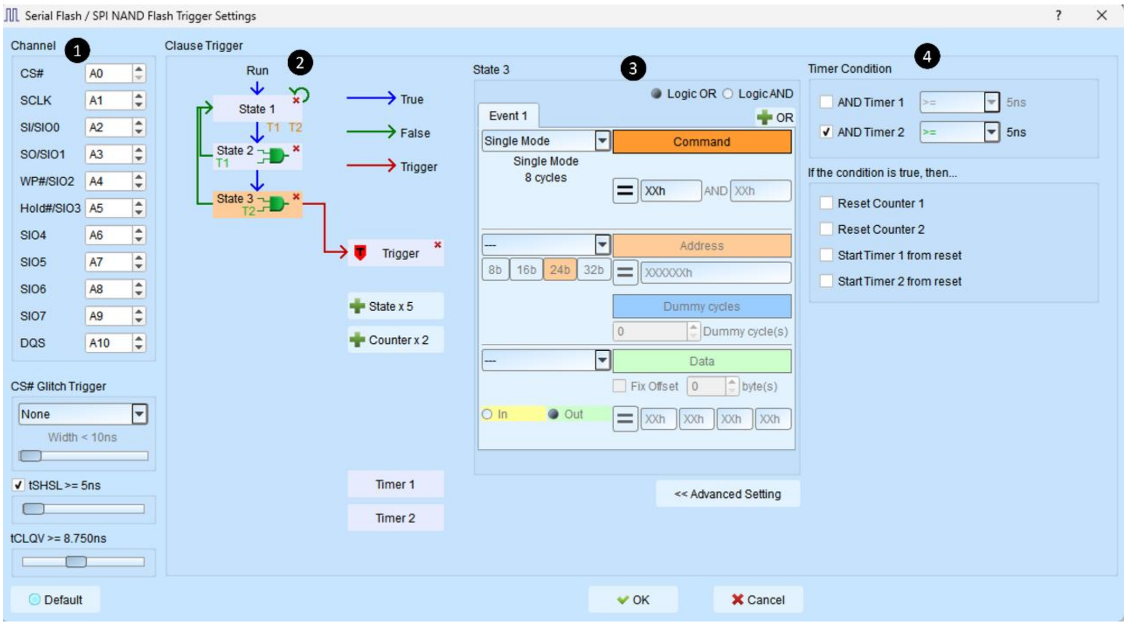
✖ Delete the selected Trigger settings.

(1) Timer / Counter features are provided in 300MHz, 250MHz, 200MHz and below sampling rate mode only.

(2) If these not valid trigger setting, system will be hold and the status bar will display “Wait for Trigger”, and must press Stop Capture manually in order to retrieve the waveform.

## Clause Trigger

The Clause trigger settings dialog box is as below.



Serial Flash / SPI NAND Flash Trigger Settings

Channel 1

| | |
|---|---|
| CS# | A0 |
| SCLK | A1 |
| SI/SIO0 | A2 |
| SO/SIO1 | A3 |
| WP#/SIO2 | A4 |
| Hold#/SIO3 | A5 |
| SIO4 | A6 |
| SIO5 | A7 |
| SIO6 | A8 |
| SIO7 | A9 |
| DQS | A10 |

CS# Glitch Trigger: None

Width < 10ns

☑ tSHSL >= 5ns

tCLQV >= 8.750ns

Clause Trigger 2

State 3 3

● Logic OR ○ Logic AND

Event 1

Single Mode — Command

Single Mode
8 cycles

= XXh AND XXh

--- — Address

8b 16b 24b 32b = XXXXXXh

Dummy cycles

0 Dummy cycle(s)

--- — Data

□ Fix Offset 0 byte(s)

○ In ● Out = XXh XXh XXh XXh

Timer Condition 4

□ AND Timer 1 >= 5ns

☑ AND Timer 2 >= 5ns

If the condition is true, then...

□ Reset Counter 1

□ Reset Counter 2

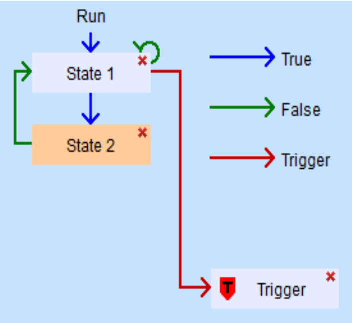
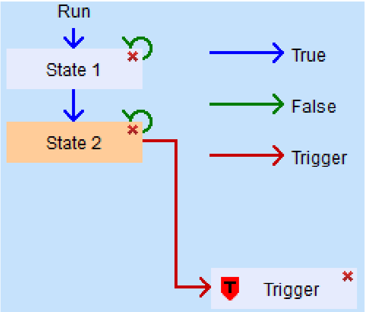
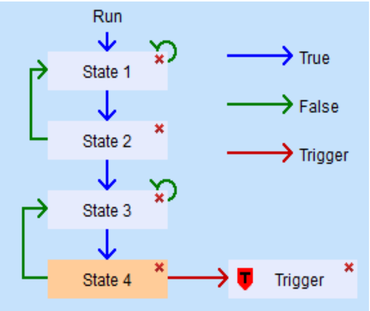
□ Start Timer 1 from reset

□ Start Timer 2 from reset

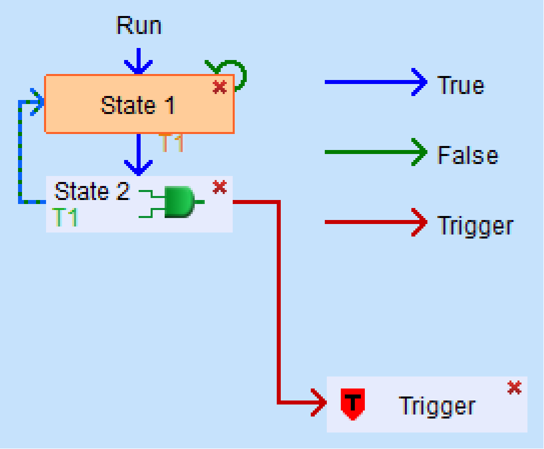
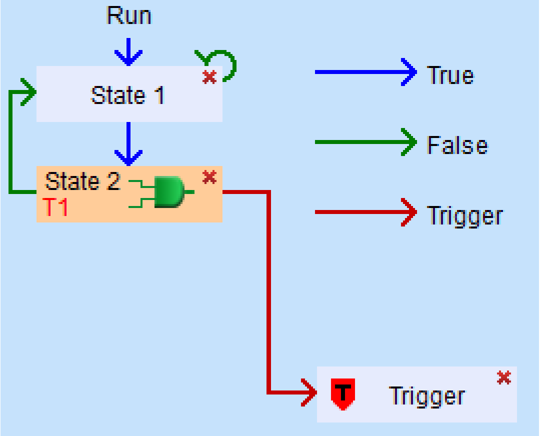
## 1. Channel and bus trigger settings:

Please refer to the descriptions in each bus trigger settings section.

## 2. Flow chart

| | | |
|---|---|---|
|  |  |  |

|  |  |  |
|---|---|---|
| State 1 next State 2 | State 1 then State 2 | State 1 next State 2<br>then State 3 next State 4 |

|  |  |
|---|---|
|  Run<br>State 1<br>T1<br>State 2<br>T1<br>True<br>False<br>Trigger<br>Trigger |  Run<br>State 1<br>State 2<br>T1<br>True<br>False<br>Trigger<br>Trigger |
| State 1 (Active T1)<br>next State 2 (Time >= T1) | State 1 (Active T1)<br>next State 2 (Time < T1) |

A state button represents a trigger state. The true state condition will only lead the branch to the next state, on the other hand, the false state condition could lead the branch to any state as below:

Click state 1 and state 2, then state 1 will branch to state 2 while state 1 is false.

Double click any state, the state will branch to itself if it is false.

Click the state button and trigger button ( T Trigger ✖ ): triggere when the state conditions is satisfied.

Delete any state by clicking the cross sign on the state.

Clear the links to the trigger button by clicking on the trigger button ( T Trigger ✖ )

Click ✚ State x 6 and ✚ Counter x 2 to add a new state.

Timer 1 : The timer ranges from 5 ns to 8 days.

### 3. State settings

This area displays the detailed trigger conditions for each level in the triggering process on the left:

The State 1 text in the upper left indicates the currently displayed state number.

● Logic OR ○ Logic AND User can set the logic operation rules between events.

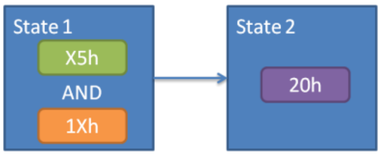
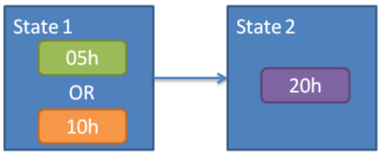
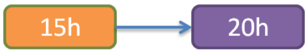
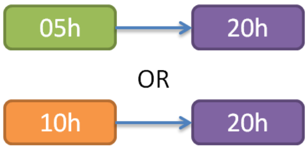
Event 1 ✖ | Event 2 ✖ | Event 3 ✖ | ✚ OR User can switch/view the number of combinations of OR/AND conditions currently in this hierarchy.

Choose ✚ OR / ✚ AND tab to add up to 8 OR/AND triggers.

The center parameter setting area varies according to the selected trigger type.

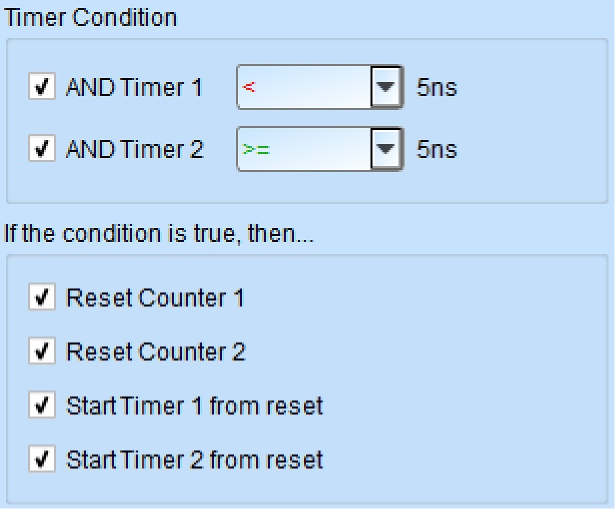
The input values support 2 /10/16 progress, binary code (followed by b, such as 01000001b), decimal code (followed by no b, such as 65), hexadecimal code (followed by h, such as 41h).

The relationship between events and triggering signals within each level can be seen in the following table:

| | AND | OR |
|---|---|---|
| Event settings | <br>State 1: X5h AND 1Xh → State 2: 20h | <br>State 1: 05h OR 10h → State 2: 20h |
| Trigger condition | <br>15h → 20h | <br>05h → 20h<br>OR<br>10h → 20h |

### 4. Timer and Counter settings

Press Advanced Setting >> button to edit the timer/counter reset settings of the state.



Timer Condition

☑ AND Timer 1 < 5ns

☑ AND Timer 2 >= 5ns

If the condition is true, then...

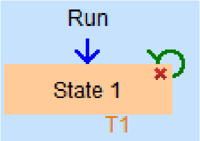
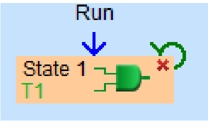
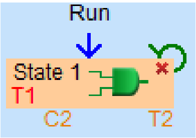
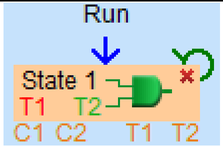
☑ Reset Counter 1

☑ Reset Counter 2

☑ Start Timer 1 from reset

☑ Start Timer 2 from reset

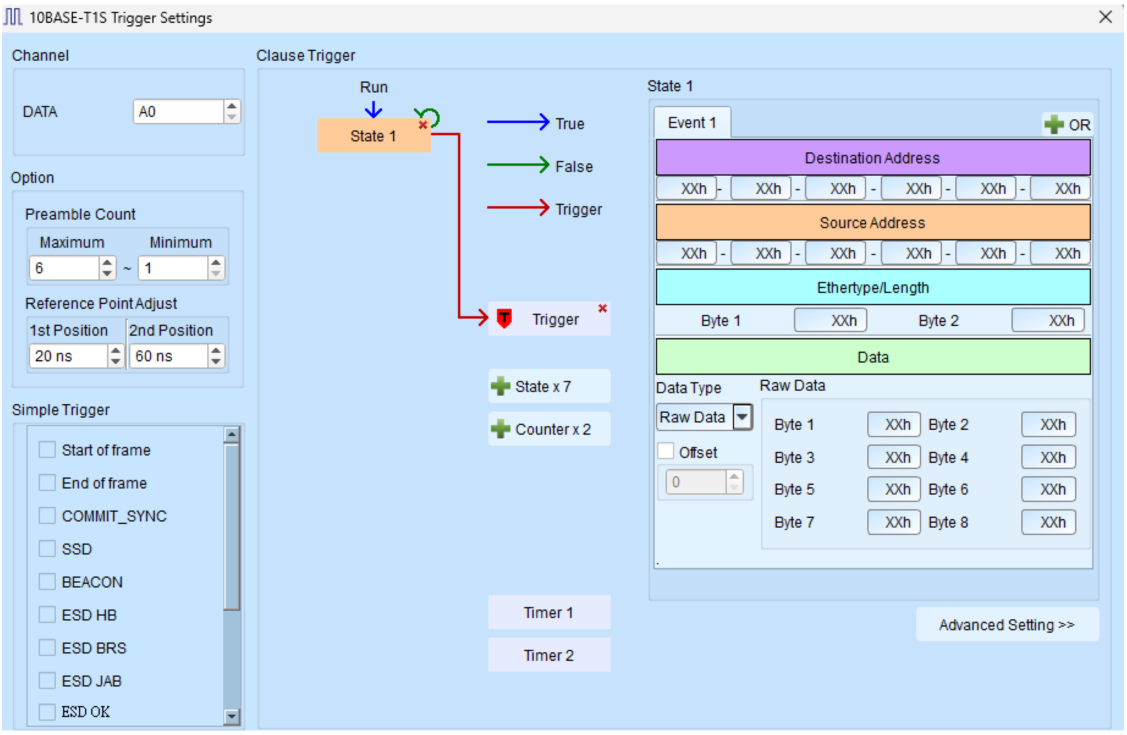
The following table describes the state button icons:

| |  |  |  |  |
|---|---|---|---|---|
| Conditions | State 1 | State 1<br>And<br>timer > T1 | State 1<br>And<br>timer < T1 | State 1<br>And<br>T2 < timer <T1 |
| satisfied condition | Start T1 | X | Start T2<br>Reset C2 | Start T1 and T2<br>Reset C1 and C2 |

## 10BASE-T1S Trigger

### Trigger Parameter Settings

Click on the "10BASE-T1S Trigger Settings" in the toolbar, and the following window will appear.



10BASE-T1S Trigger Settings

Channel
DATA: A0

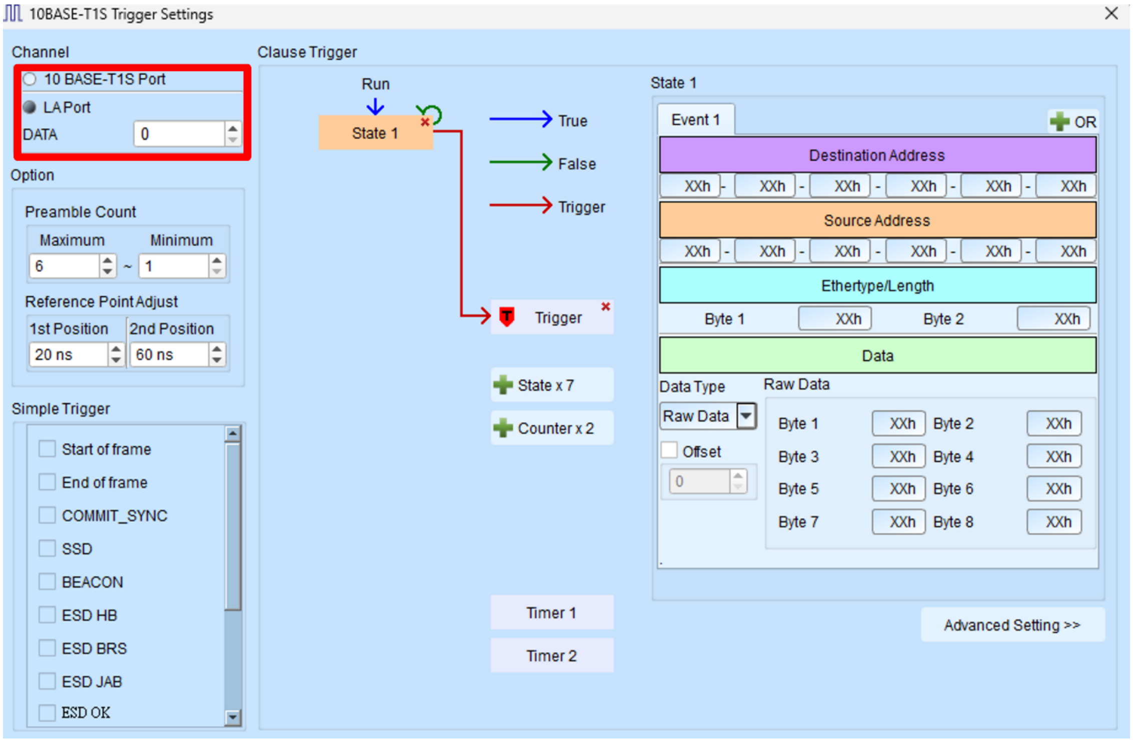
Option
Preamble Count
Maximum: 6 ~ Minimum: 1
Reference Point Adjust
1st Position: 20 ns
2nd Position: 60 ns

Simple Trigger
□ Start of frame
□ End of frame
□ COMMIT_SYNC
□ SSD
□ BEACON
□ ESD HB
□ ESD BRS
□ ESD JAB
□ ESD OK

Clause Trigger

State 1
Event 1
Destination Address: XXh - XXh - XXh - XXh - XXh - XXh
Source Address: XXh - XXh - XXh - XXh - XXh - XXh
Ethertype/Length: Byte 1 XXh, Byte 2 XXh
Data
Data Type: Raw Data
□ Offset: 0
Raw Data: Byte 1 XXh, Byte 2 XXh, Byte 3 XXh, Byte 4 XXh, Byte 5 XXh, Byte 6 XXh, Byte 7 XXh, Byte 8 XXh

Travel Bus series



10BASE-T1S Trigger Settings

Channel
○ 10 BASE-T1S Port
● LA Port
DATA: 0

Option
Preamble Count
Maximum: 6 ~ Minimum: 1
Reference Point Adjust
1st Position: 20 ns
2nd Position: 60 ns

Simple Trigger
□ Start of frame
□ End of frame
□ COMMIT_SYNC
□ SSD
□ BEACON
□ ESD HB
□ ESD BRS
□ ESD JAB
□ ESD OK

Clause Trigger

State 1
Event 1
Destination Address: XXh - XXh - XXh - XXh - XXh - XXh
Source Address: XXh - XXh - XXh - XXh - XXh - XXh
Ethertype/Length: Byte 1 XXh, Byte 2 XXh
Data
Data Type: Raw Data
□ Offset: 0
Raw Data: Byte 1 XXh, Byte 2 XXh, Byte 3 XXh, Byte 4 XXh, Byte 5 XXh, Byte 6 XXh, Byte 7 XXh, Byte 8 XXh

***Since TravelBus has specially designed channels for 10BASE-T1, the 10BASE-T1S trigger settings in the TravelBus software will include additional channel source options.***

1. **Channel:** Configure the Data channel.
2. **Option**

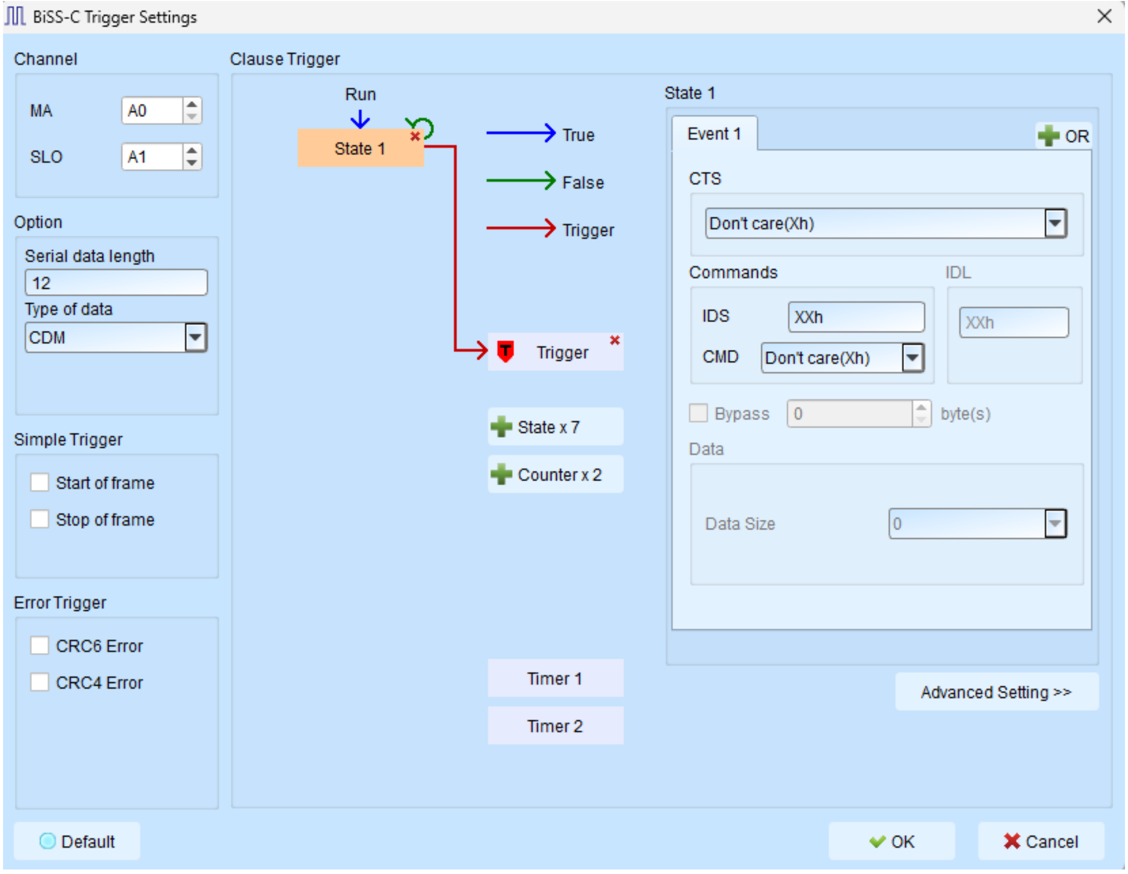
   **Preamble Count:** Set the number of Preamble bits.

   **Reference Point Adjust:** Adjusts the reference point for logic level determination.
3. **Simple Trigger:** Configure the trigger packet.
4. **Clause Trigger:** Please refer to the Parallel Clause Trigger description for details.
5. **State:** This section displays the detailed trigger conditions within each level of the trigger process on the left. Users can specify values for Destination Address, Source Address, Ethertype/Length, or Data, or leave 'X' as a wildcard for any value.

## BiSS-C Trigger

### BiSS-C Trigger Parameter Settings

Click on the " BiSS-C Trigger Settings" in the toolbar, and the following window will appear.



BiSS-C Trigger Settings

Channel
MA: A0
SLO: A1

Option
Serial data length: 12
Type of data: CDM

Simple Trigger
□ Start of frame
□ Stop of frame

Error Trigger
□ CRC6 Error
□ CRC4 Error

State 1 – Event 1
CTS: Don't care(Xh)
Commands – IDS: XXh
CMD: Don't care(Xh)
IDL: XXh
□ Bypass 0 byte(s)
Data – Data Size: 0

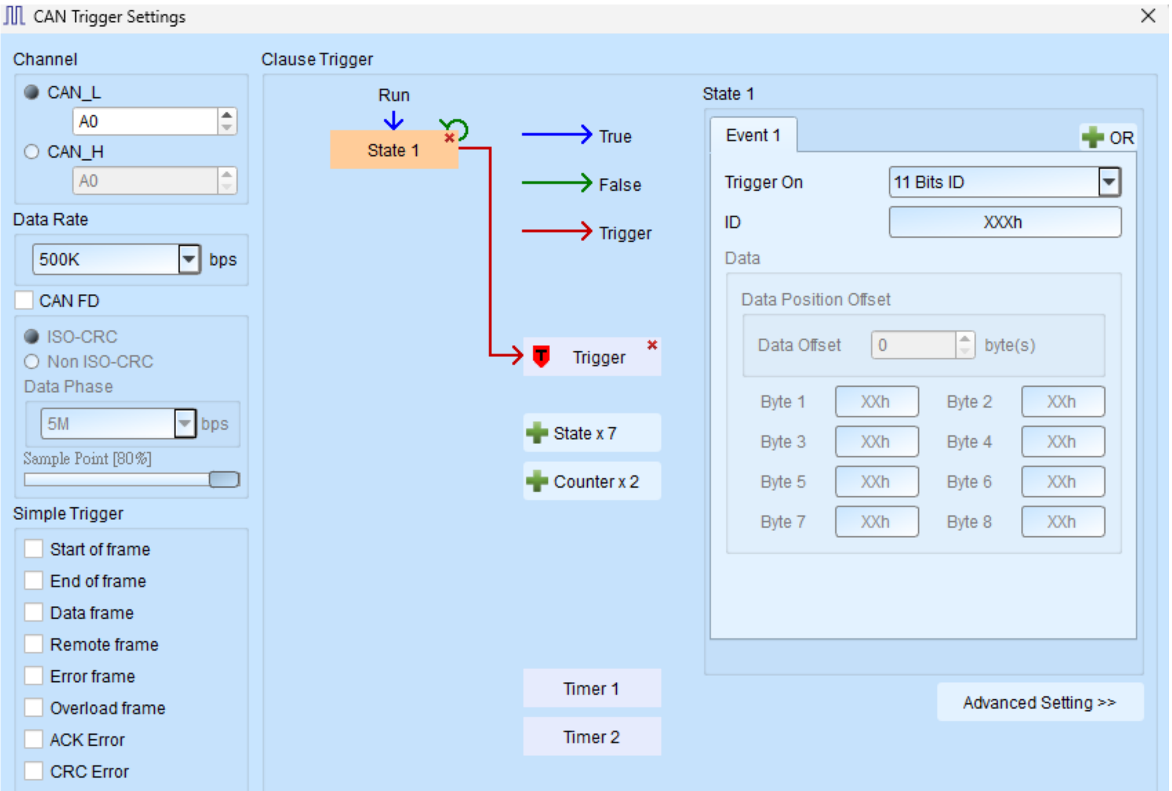
1. **Channel:** Select channels.
2. **Option:**
   - I. Serial data length: Set SCD packet length, the default is 12 bits, 255 bits is the maximum.
   - II. Type of data: Select the CDM or CDS packet.
3. **Simple Trigger:** Specific trigger function of BiSS-C.
4. **Error Trigger:**

   CRC6 Error/CRC4 Error: Trigger CRC6 or CRC4 error of BiSS-C.
5. **Clause Trigger:** Please reference Clause Trigger chapter.
6. **State:** Show the details of trigger condition in every state as left side; typing or selecting the trigger values in the CTS, Commands, IDL and Data fields, default value is XX means “don’t care”.

# CAN Trigger

## CAN Trigger Settings

Click CAN Trigger in the toolbar and will show the dialog as the following。



CAN Trigger Settings

Channel

- ● CAN_L: A0
- ○ CAN_H: A0

Data Rate: 500K bps

□ CAN FD

- ● ISO-CRC
- ○ Non ISO-CRC

Data Phase: 5M bps

Sample Point [80%]

Simple Trigger

- □ Start of frame
- □ End of frame
- □ Data frame
- □ Remote frame
- □ Error frame
- □ Overload frame
- □ ACK Error
- □ CRC Error

Clause Trigger

State 1

Event 1 | OR

Trigger On: 11 Bits ID

ID: XXXh

Data

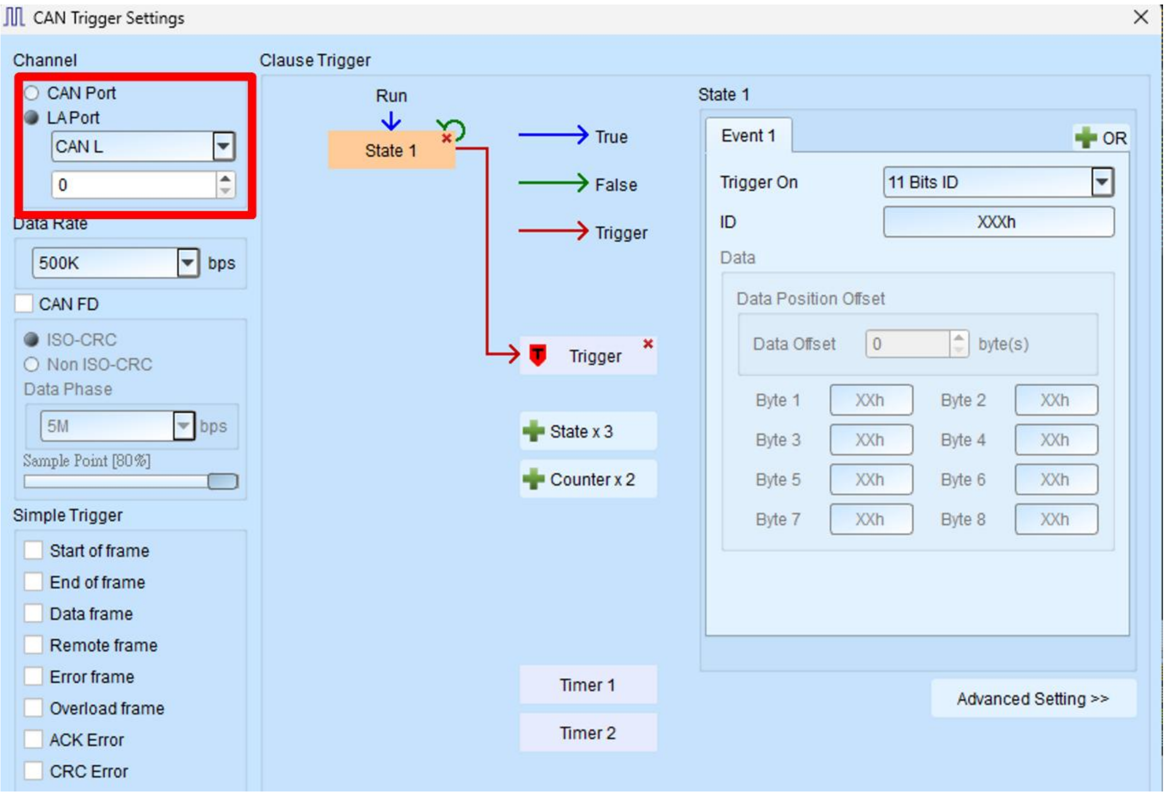
Data Position Offset

Data Offset: 0 byte(s)

| | | | |
|---|---|---|---|
| Byte 1 | XXh | Byte 2 | XXh |
| Byte 3 | XXh | Byte 4 | XXh |
| Byte 5 | XXh | Byte 6 | XXh |
| Byte 7 | XXh | Byte 8 | XXh |

Advanced Setting >>

Travel Bus series



CAN Trigger Settings

Channel

- ○ CAN Port
- ● LA Port: CAN L, 0

Data Rate: 500K bps

□ CAN FD

- ● ISO-CRC
- ○ Non ISO-CRC

Data Phase: 5M bps

Sample Point [80%]

Simple Trigger

- □ Start of frame
- □ End of frame
- □ Data frame
- □ Remote frame
- □ Error frame
- □ Overload frame
- □ ACK Error
- □ CRC Error

Clause Trigger

State 1

Event 1 | OR

Trigger On: 11 Bits ID

ID: XXXh

Data

Data Position Offset

Data Offset: 0 byte(s)

| | | | |
|---|---|---|---|
| Byte 1 | XXh | Byte 2 | XXh |
| Byte 3 | XXh | Byte 4 | XXh |
| Byte 5 | XXh | Byte 6 | XXh |
| Byte 7 | XXh | Byte 8 | XXh |

Advanced Setting >>

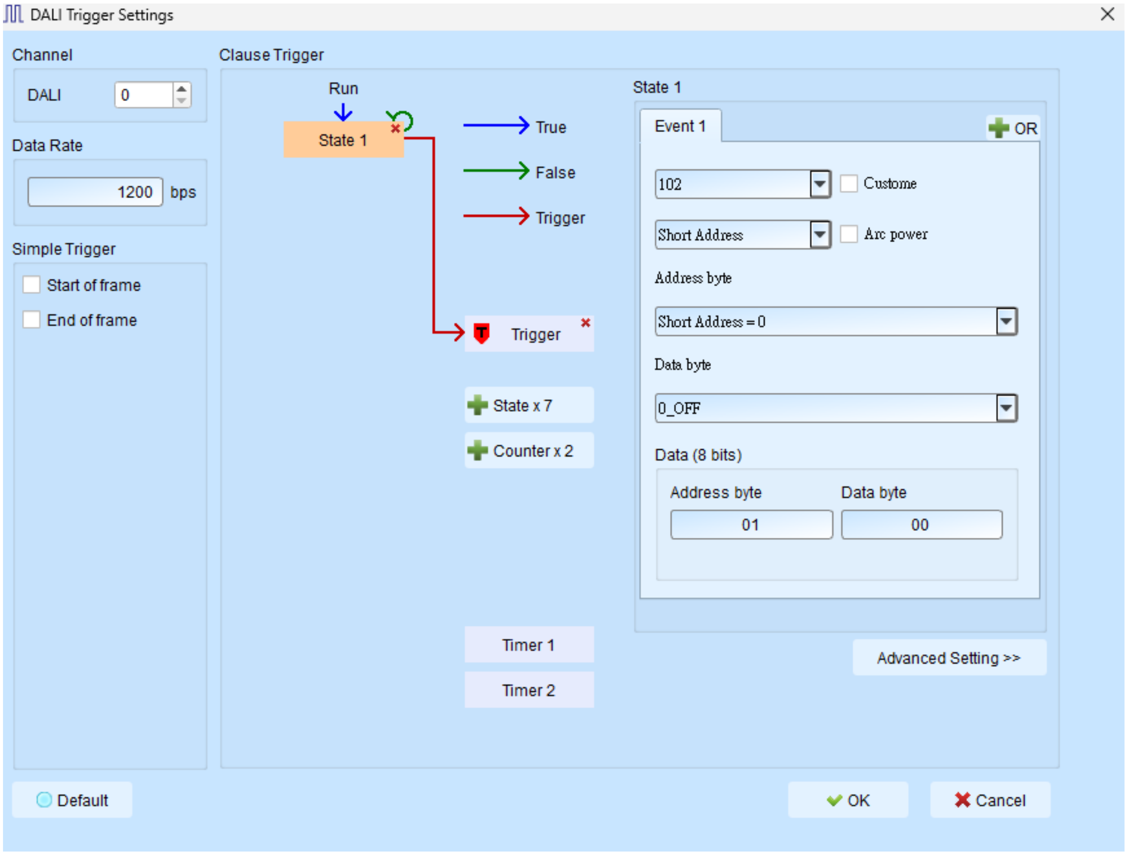
***Since TravelBus has specially designed channels for CAN, the CAN trigger settings in the TravelBus software will include additional channel source options.***

1. **Channel:** Configure the CAN interface (supported only on TravelBus B series models) or set to LA channels.
2. **Data Rate:** CAN data rate. Users could type the data rate if no data rate is available.
3. **Simple Trigger:** Specific trigger function of CAN.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Show the details of trigger condition in every state as left side; there are 11 Bits ID, 29 Bits ID, Data, 11 Bits ID + Data and 29 Bits ID + Data selections in the Trigger On; selecting one of them and typing the trigger values in the ID or Data fields, default value is XX means “don’t care”.

## DALI2 Trigger

### DALI Trigger Settings

Click DALI Trigger in the toolbar and will show the dialog as the following.



DALI Trigger Settings

Channel: DALI 0

Data Rate: 1200 bps

Simple Trigger:
□ Start of frame
□ End of frame

State 1, Event 1: 102 □ Custome; Short Address □ Arc power

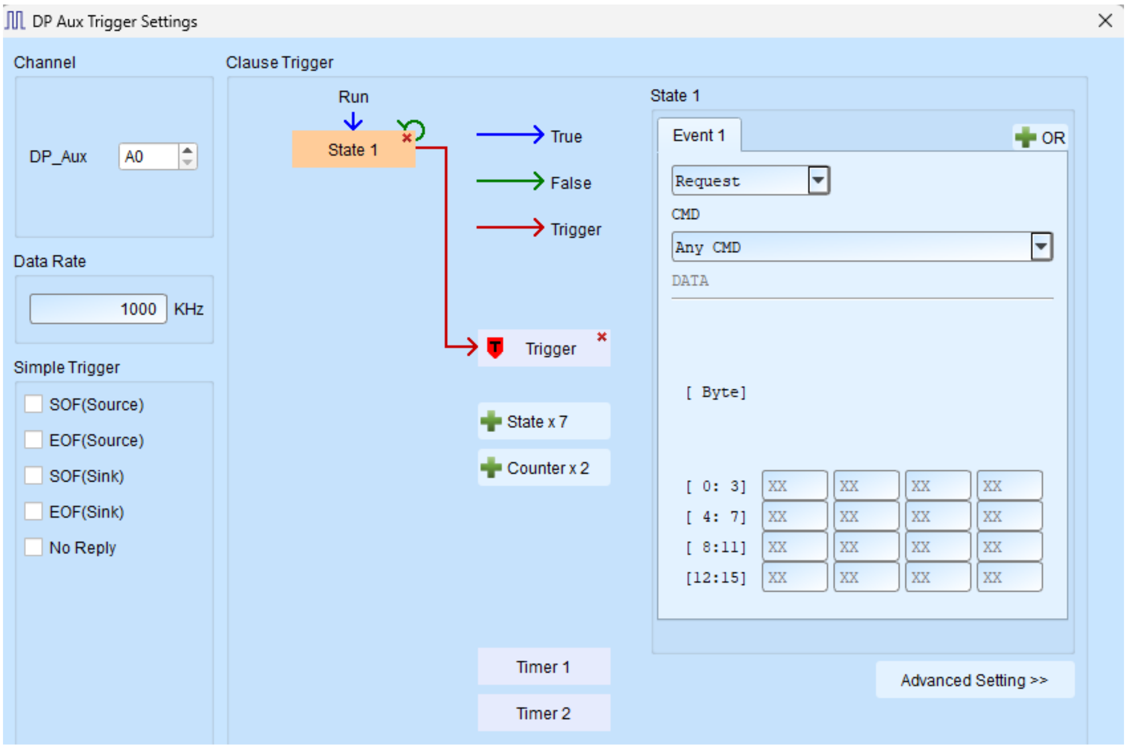
Address byte: Short Address = 0

Data byte: 0_OFF

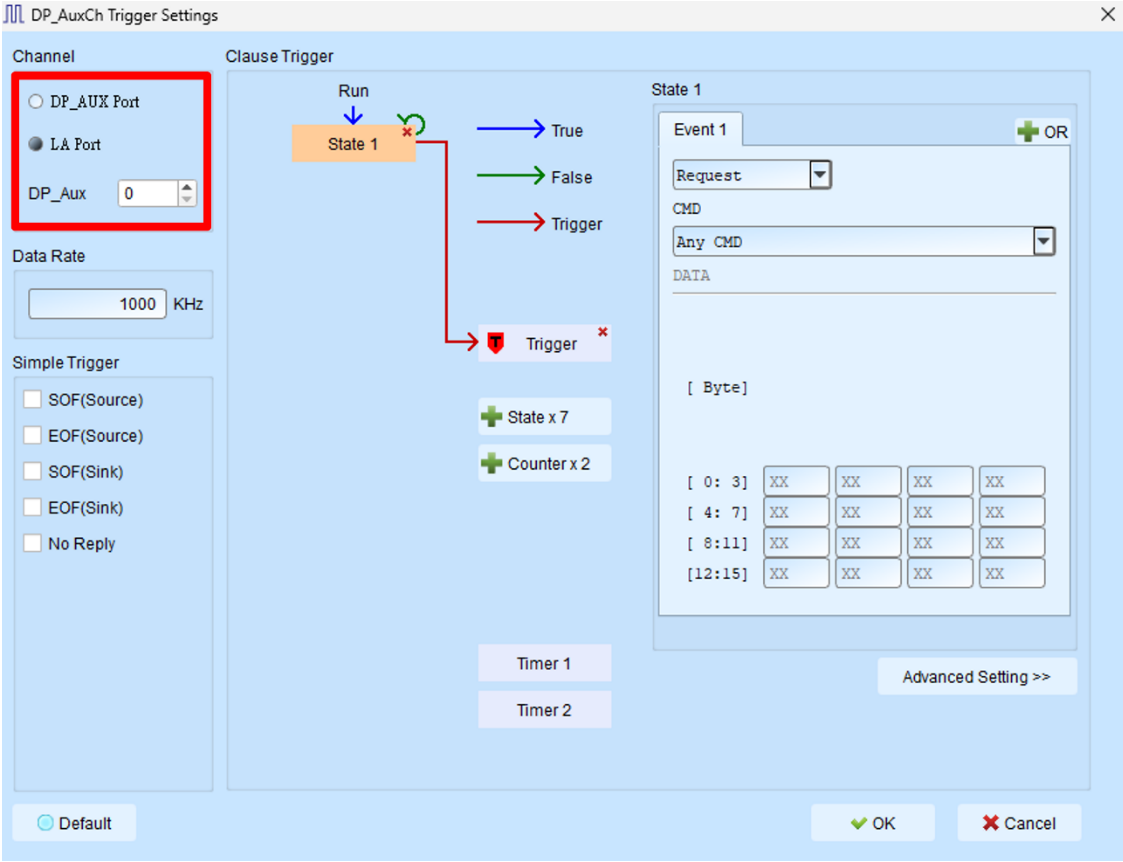
Data (8 bits): Address byte 01, Data byte 00

1. **Channel:** Select channels.
2. **Simple Trigger:** Trigger specific of DALI.
3. **Clause Trigger:** Please reference Clause Trigger chapter.
4. **State:** Show the details of trigger condition in every state as left side; selecting the trigger type of DALI (102, 103), command type (short address, group address, broadcast, response, special command), or custom cmd.

## DPAux Ch Trigger



## Travel Bus series



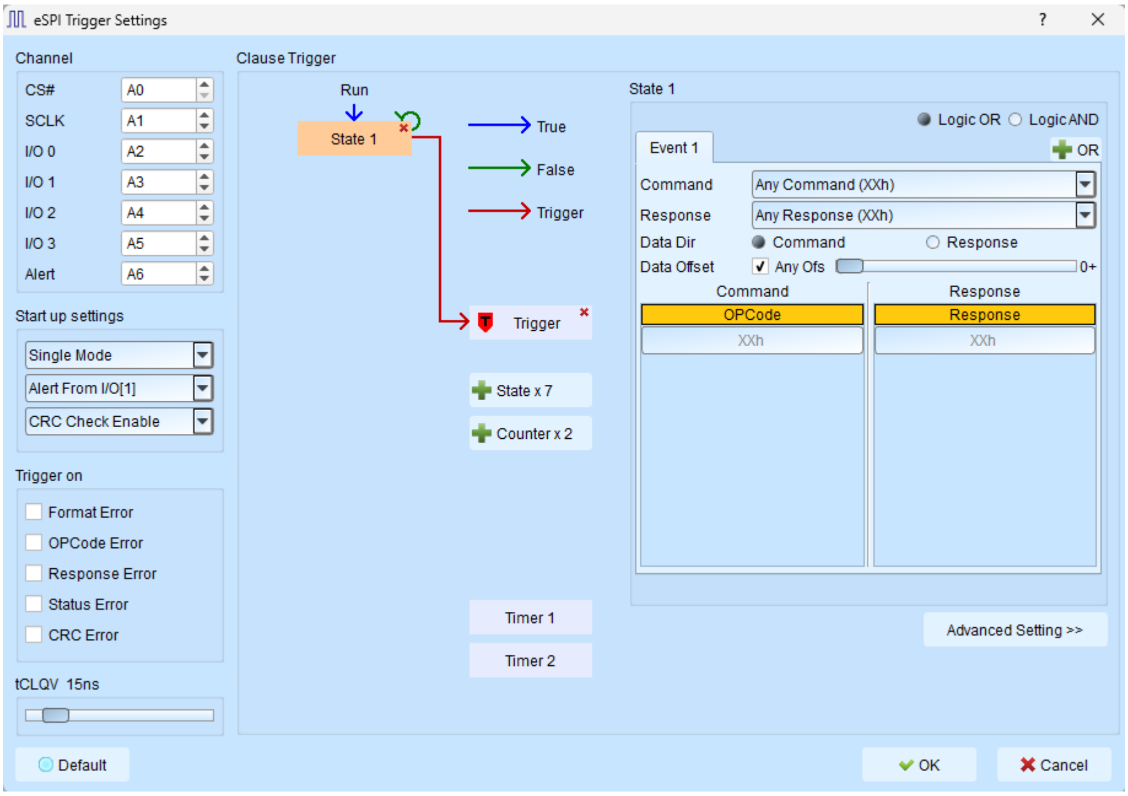
***Since TravelBus has specially designed channels for DP Aux, the DP Aux trigger settings in the TravelBus software will include additional channel source options.***

1. **Channel:** Configure the DP Aux channel
2. **Data Rate: Configure the DP Aux data rate**
3. **Simple Trigger:** Triggers on common basic packets. Enabled when selected.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** This section displays the detailed trigger conditions within each level of the trigger process on the left. Users can configure the packet as Request or Reply, fine-tune Command settings, or specify trigger values for the Data packet content. Alternatively, 'X' can be used as a wildcard to represent any value.

## eSPI Trigger

### eSPI Trigger Settings

Click eSPI Trigger in the toolbar and will show the dialog as the following.



eSPI Trigger Settings

Channel

| | |
|---|---|
| CS# | A0 |
| SCLK | A1 |
| I/O 0 | A2 |
| I/O 1 | A3 |
| I/O 2 | A4 |
| I/O 3 | A5 |
| Alert | A6 |

Start up settings

Single Mode
Alert From I/O[1]
CRC Check Enable

Trigger on

□ Format Error
□ OPCode Error
□ Response Error
□ Status Error
□ CRC Error

tCLQV 15ns

Clause Trigger

State 1

● Logic OR ○ Logic AND

Event 1

Command: Any Command (XXh)
Response: Any Response (XXh)
Data Dir: ● Command ○ Response
Data Offset: ☑ Any Ofs 0+

| Command | Response |
|---|---|
| OPCode | Response |
| XXh | XXh |

1. **Channel:** Select channels.

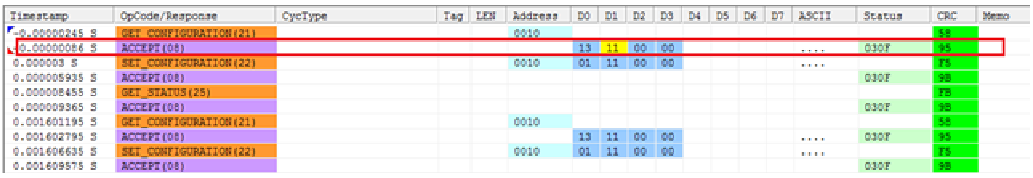
2. **Startup settings:** Set the initial parameters of eSPI.

3. **Trigger on:** Trigger specific error of eSPI.

4. **Clause Trigger:** Please reference Clause Trigger chapter.

5. **State:** Show the details of trigger condition in every state as left side; selecting the trigger values in the Command, Response fields, default value is 00h.

   I. **Data Dir:** Trigger the data in the Command or Response.



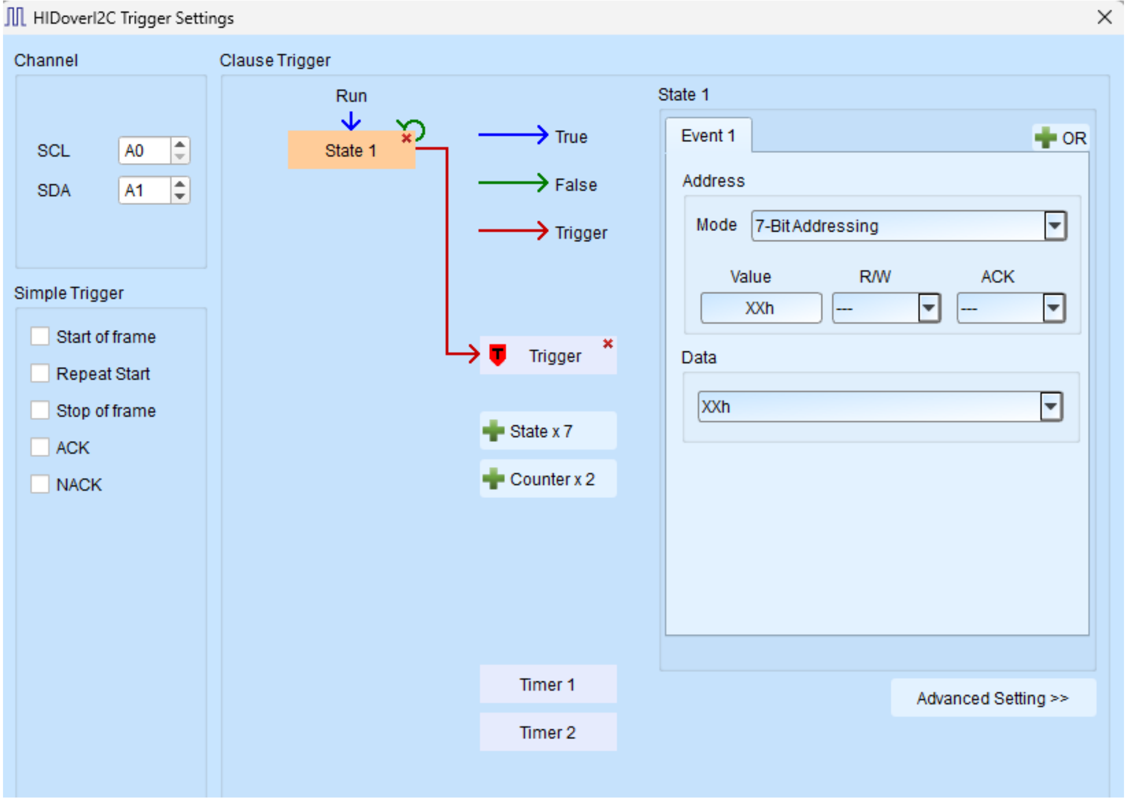
| Timestamp | OpCode/Response | CycType | Tag | LEN | Address | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | ASCII | Status | CRC | Memo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| -0.00000245 S | GET_CONFIGURATION(21) | | | | 0010 | | | | | | | | | | | 58 | |
| -0.00000086 S | ACCEPT(08) | | | | | 13 | 11 | 00 | 00 | | | | | .... | 030F | 95 | |
| 0.000003 S | SET_CONFIGURATION(22) | | | | 0010 | 01 | 11 | 00 | 00 | | | | | .... | | F5 | |
| 0.000005935 S | ACCEPT(08) | | | | | | | | | | | | | | 030F | 9B | |
| 0.000008455 S | GET_STATUS(25) | | | | | | | | | | | | | | | FB | |
| 0.000009365 S | ACCEPT(08) | | | | | | | | | | | | | | 030F | 9B | |
| 0.001601195 S | GET_CONFIGURATION(21) | | | | 0010 | | | | | | | | | | | 58 | |
| 0.001602795 S | ACCEPT(08) | | | | | 13 | 11 | 00 | 00 | | | | | .... | 030F | 95 | |
| 0.001606635 S | SET_CONFIGURATION(22) | | | | 0010 | 01 | 11 | 00 | 00 | | | | | .... | | F5 | |
| 0.001609575 S | ACCEPT(08) | | | | | | | | | | | | | | 030F | 9B | |

II. **Data Offset: Trigger the data from start of data frame without any offset. For example, setting D0 13h will check the first byte of data frame. With any offset, the siganl would be triggered by the byte pattern. For instance, setting D0+ XXh D1+ 11h, and the byte pattern of XXh and 11h will be triggered regardless of the position of the data.**

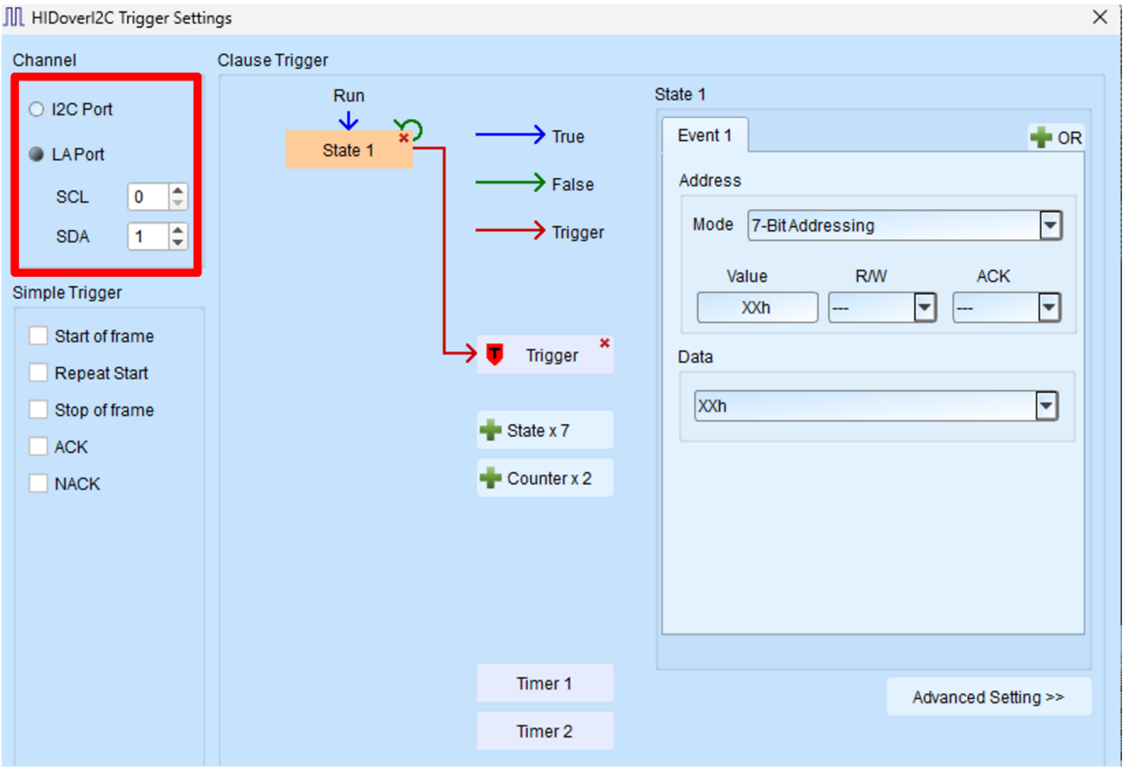
## HIDoverI2C Trigger

### HIDoverI2C Trigger Settings

Click HIDoverI2C Trigger in the toolbar and will show the dialog as the following.



## Travel Bus series



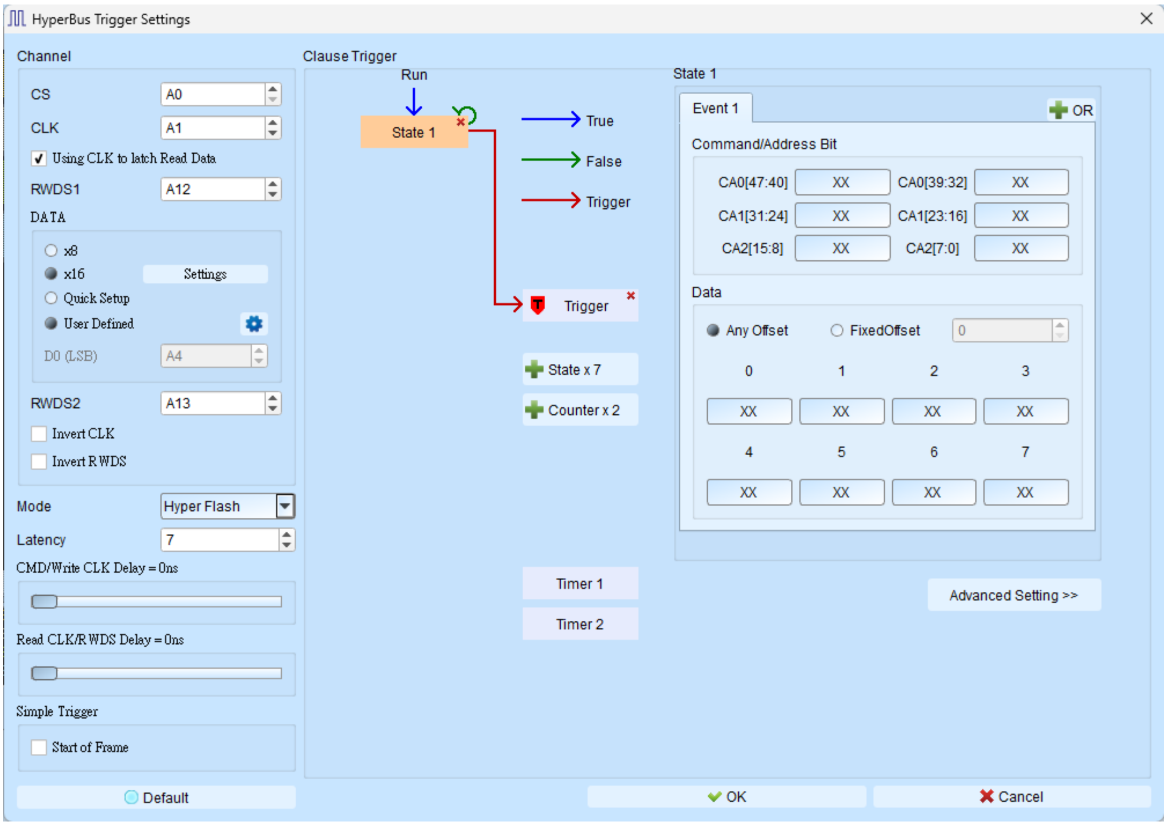
***Since TravelBus has specially designed channels for I2C, the HID Over I2C settings in the TravelBus software will include additional channel source options.***

1. **Channel:** Available only on TravelBus B series models, or set to LA channels.
2. **Simple Trigger:** Configure I2C Specific Frame Trigger.
3. **Clause Trigger:** Please reference Clause Trigger chapter.
4. **State:** This section displays the detailed trigger conditions within each level of the trigger process on the left. Users can enter specific trigger values in the Value, R/W, ACK, and Data fields, or use 'X' as a wildcard for any value.
   The Data field can be set to HID descriptor as a trigger condition. When the trigger item is set to descriptor, the R/W field will automatically switch to Read mode.

## HyperBus Trigger

### HyperBus Trigger Settings

Click HyperBus Trigger in the toolbar and will show the dialog as the following.



HyperBus Trigger Settings

Channel

CS: A0

CLK: A1

☑ Using CLK to latch Read Data

RWDS1: A12

DATA

○ x8

● x16 Settings

○ Quick Setup

● User Defined

D0 (LSB): A4

RWDS2: A13

☐ Invert CLK

☐ Invert RWDS

Mode: Hyper Flash

Latency: 7

CMD/Write CLK Delay = 0ns

Read CLK/RWDS Delay = 0ns

Simple Trigger

☐ Start of Frame

State 1

Event 1

Command/Address Bit

| | | | |
|---|---|---|---|
| CA0[47:40] | XX | CA0[39:32] | XX |
| CA1[31:24] | XX | CA1[23:16] | XX |
| CA2[15:8] | XX | CA2[7:0] | XX |

Data

● Any Offset ○ FixedOffset 0

| 0 | 1 | 2 | 3 |
|---|---|---|---|
| XX | XX | XX | XX |
| 4 | 5 | 6 | 7 |
| XX | XX | XX | XX |

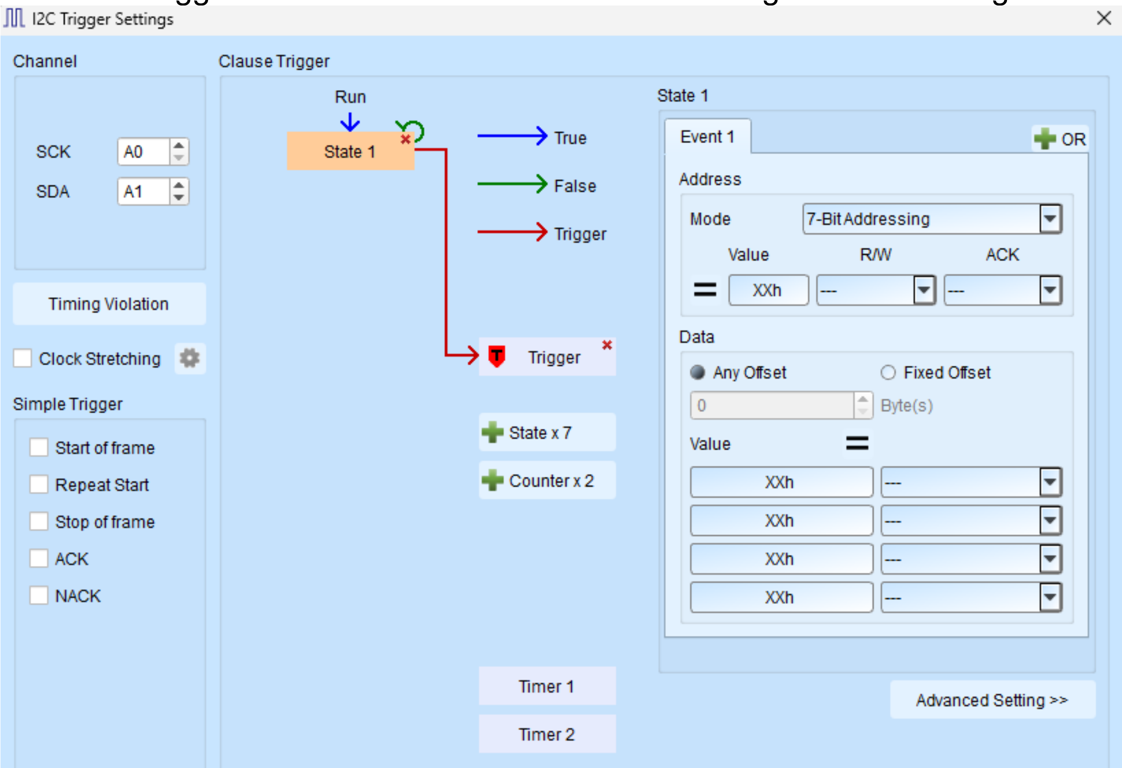
1. **Channel:**
   - I. **CS、CLK、RWDS1、RWDS2:** Configure HyperBus Channel.
   - II. **Using CLK to latch Read Data:** Uses CLK to latch Read Data. Enabled when selected.
   - III. **DATA:**
     - ♦ **x8、x16:** Configure the Data Bus Width.
     - ♦ **Quick Setup、User Defined:** Users can either use the Quick Setup function to batch-configure DATA channels or manually define each DATA bit channel.
     - ♦ **Settings:** Configure the Data Arrangement format.
   - IV. **Invert CLK:** Reverses the CLK signal. Enabled when selected.
   - V. **Invert RWDS respectively:** Independently inverts the RWDS1 & RWDS2 signals. Enabled when selected.
2. **Mode:** Can be set to HyperFlash or HyperRAM.

3. **Latency:** Sets the number of Sample Points for latency.
4. **CMD/Write CLK Delay、Read CLK/RWDS Delay:** Configures the delay time for commands, write clock, read clock, and RWDS.
5. **Clause Trigger:** Please reference Parallel Clause Trigger chapter.
6. **State:** Displays the detailed trigger conditions within each level of the trigger process on the left. Users can enter specific trigger values in the Command/Address Bit and Data fields, or use 'X' as a wildcard for any value.

# I$^2$C Trigger

## I2C Trigger Settings

Click I2C Trigger in the toolbar and will show the dialog as the following.



I2C Trigger Settings

Channel

SCK A0

SDA A1

Timing Violation

□ Clock Stretching

Simple Trigger

□ Start of frame

□ Repeat Start

□ Stop of frame

□ ACK

□ NACK

Clause Trigger

State 1

Event 1 OR

Address

Mode 7-Bit Addressing

Value = XXh, R/W ---, ACK ---

Data

● Any Offset ○ Fixed Offset

0 Byte(s)

Value =

XXh ---

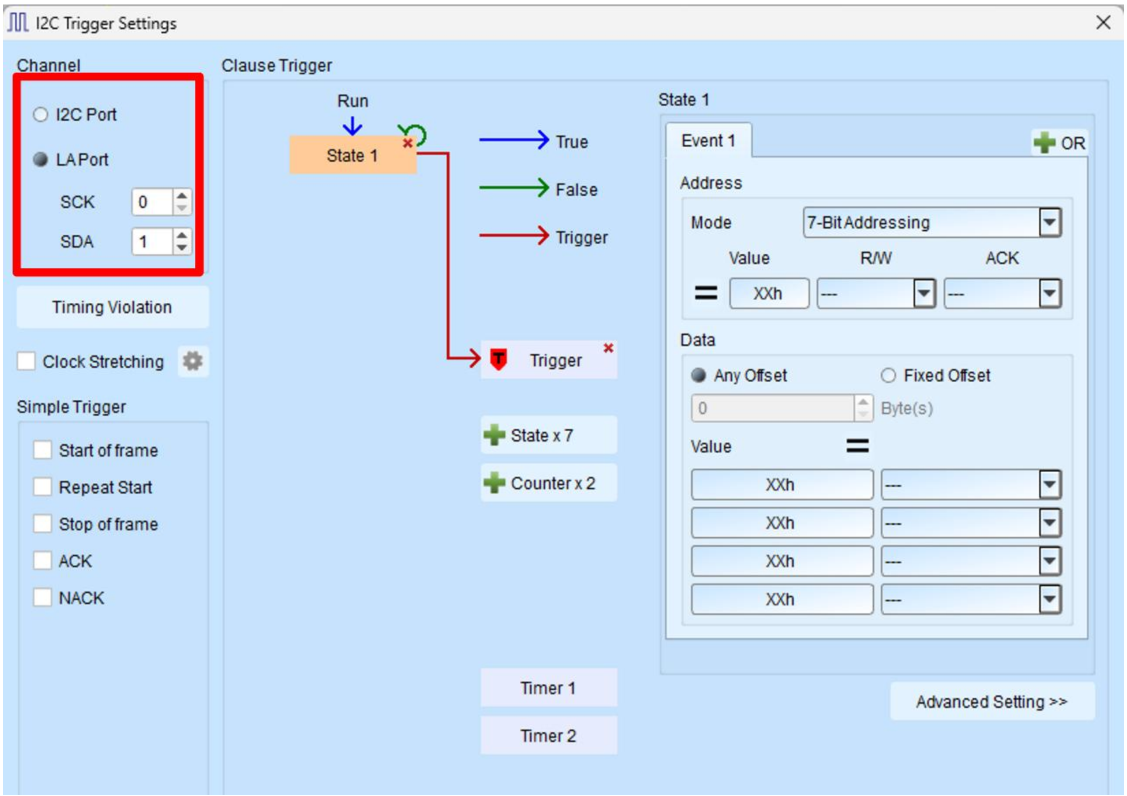
XXh ---

XXh ---

XXh ---

Advanced Setting >>

## Travel Bus series



I2C Trigger Settings

Channel

○ I2C Port

● LA Port

SCK 0

SDA 1

Timing Violation

□ Clock Stretching

Simple Trigger

□ Start of frame

□ Repeat Start

□ Stop of frame

□ ACK

□ NACK

Clause Trigger

State 1

Event 1 OR

Address

Mode 7-Bit Addressing

Value = XXh, R/W ---, ACK ---

Data

● Any Offset ○ Fixed Offset

0 Byte(s)

Value =

XXh ---

XXh ---

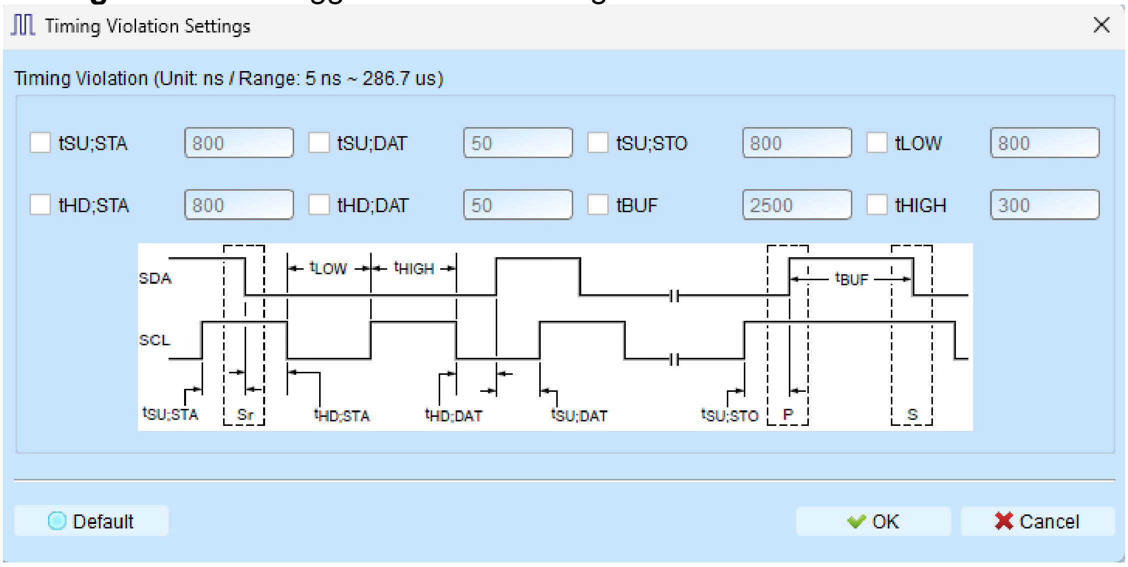
XXh ---

XXh ---

Advanced Setting >>

***Since TravelBus has specially designed channels for I2C, the I2C settings in the TravelBus software will include additional channel source options.***

1. **Channel:** Configure the I2C interface (supported only on TravelBus B series models) or set to LA channels.
2. **Simple Trigger:** Configure I2C specific frame trigger.
3. **Timing Violation:** Trigger based on timing condition violations.

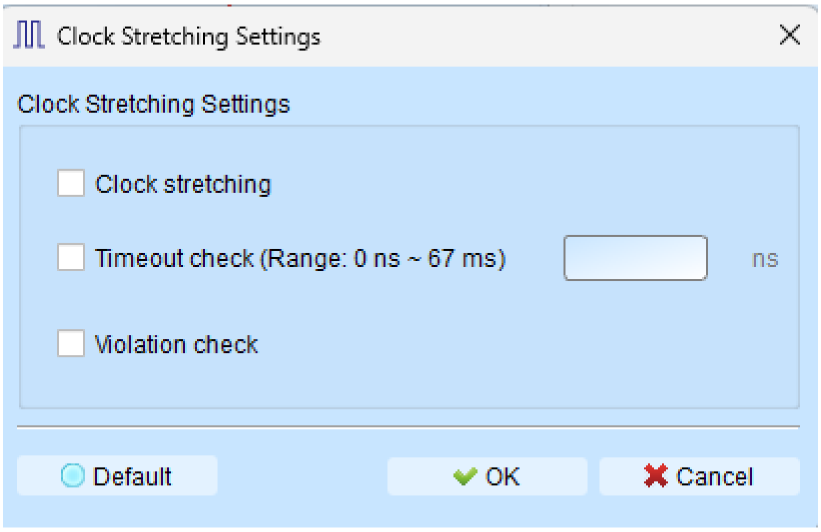


Timing Violation Settings

Timing Violation (Unit: ns / Range: 5 ns ~ 286.7 us)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| □ tSU;STA | 800 | □ tSU;DAT | 50 | □ tSU;STO | 800 | □ tLOW | 800 |
| □ tHD;STA | 800 | □ tHD;DAT | 50 | □ tBUF | 2500 | □ tHIGH | 300 |

4. **Clock Stretching:** Triggers on Clock Stretching events. Enabled when selected.



Clock Stretching Settings

Clock Stretching Settings

□ Clock stretching

□ Timeout check (Range: 0 ns ~ 67 ms) ns

□ Violation check

I. **Clock Stretching:** Triggered when slave stretching clock.

II. **Timeout Check:** Set the amount of time after which the timeout is determined and triggered.

III. **Violation Check:** In HS mode, triggered at the illegal SCL width.

5. **Clause Trigger:** Please reference Parallel Clause Trigger chapter.

6. **State:** Displays the detailed trigger conditions within each level of the trigger process on the left. Users can specify trigger values for Value, R/W, ACK, and Data fields or use 'X' as a wildcard for any value.

   **I.** The Data field allows up to 4 Bytes to be set. Unused fields should be filled with XXh to indicate any value. Additionally, users can click the equal sign ("=") next to the set value to modify the trigger condition to "not equal to" the specified value.

   **II.** The input field can contain the required trigger Data or 'X' to represent any value. When entering values:

   - Hexadecimal values should end with 'h'.
   - Binary values should end with 'b'.
   - Decimal values do not require a suffix.

   **III.** Trigger Data Offset (Offset):

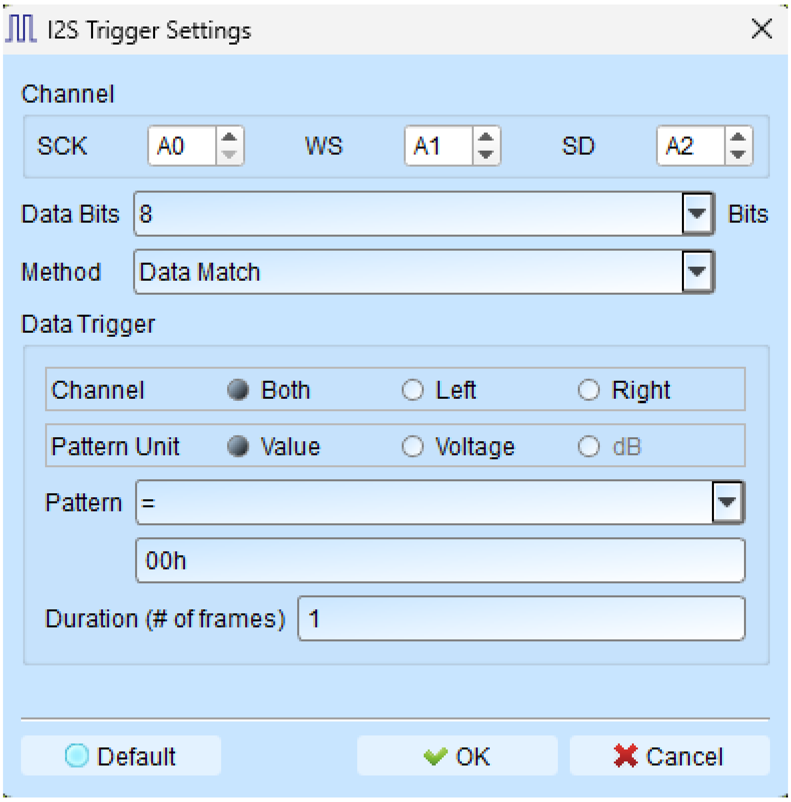
   i. Any Offset: Triggers when valid **Data** that meets the set conditions appears anywhere in the **Data** field, regardless of offset.

   ii. Fixed Offset: Triggers only when valid Data that meets the set conditions appears at the specified offset.

## I$^2$S Trigger

### I2S Trigger Settings

Click I2S Trigger in the toolbar and will show the dialog as the following.



I2S Trigger Settings

Channel

SCK: A0 WS: A1 SD: A2

Data Bits: 8 Bits

Method: Data Match

Data Trigger

Channel: ● Both ○ Left ○ Right

Pattern Unit: ● Value ○ Voltage ○ dB

Pattern: =

00h

Duration (# of frames): 1

1. **Channel:** Select Serial clock (SCK), word select (WS) and serial data (SD).
2. **Data Bits:** 1-32 bits, normally it is 8, 12, 16, 24 or 32.
3. **Method:**

   Data Match：Trigger when the conditions matched.

   Rising Edge：Trigger when it is a rising edge between two patterns.

   Falling Edge：Trigger when it is a falling edge between two patterns.

   Glitch：Trigger when there is a glitch.

   Mute：When the duration (or number of frames) is p, the instrument will trigger when the signal is within the range of -P < X < +P.

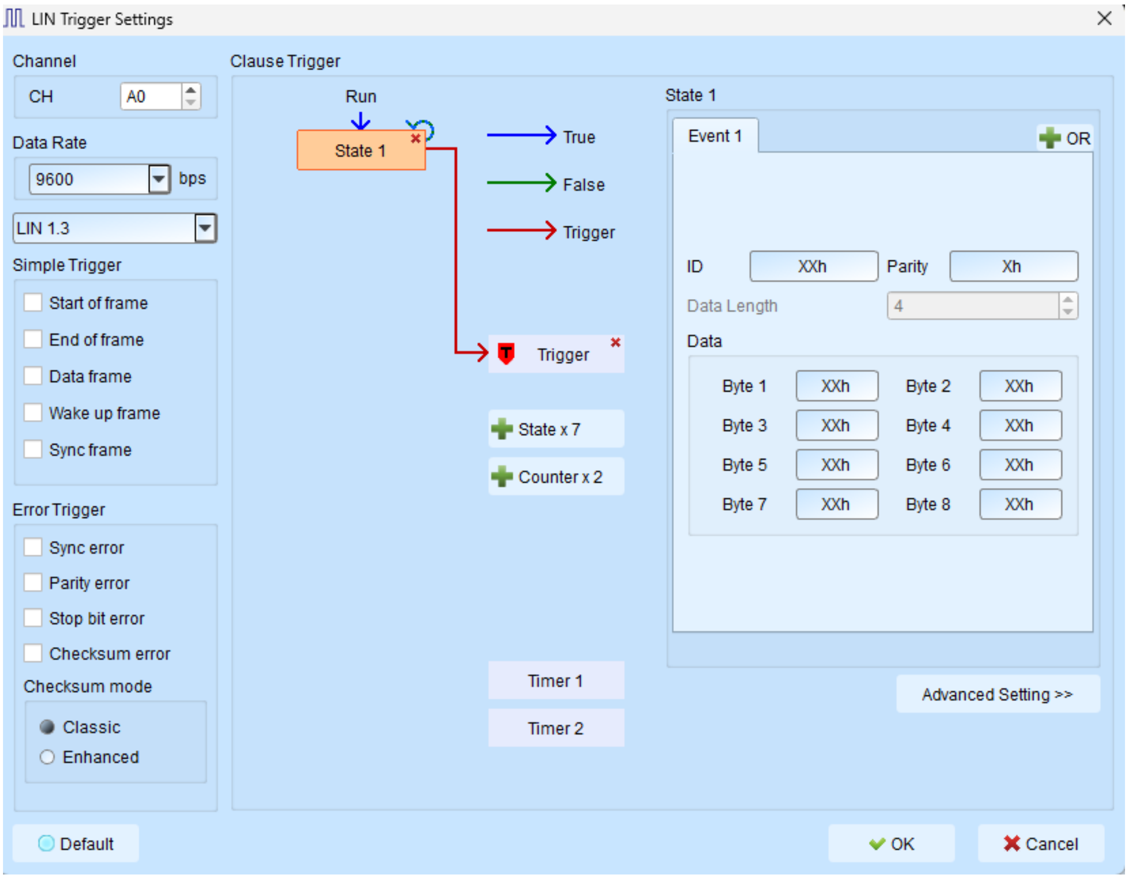
   Clip：When the duration (or number of frames) is p, the instrument will trigger when the signal is within the range of -P < X ∪ +P > X.

   Timing Violation： Provide 6 trigger conditions and this function will be enabled at 200 MHz sampling rate.
4. **Data Trigger:** Select Both, Left, Right channel and pattern unit.

## LIN Trigger

### LIN Trigger Settings

Click LIN Trigger in the toolbar and will show the dialog as the following.


LIN Trigger Settings

Channel: CH A0

Data Rate: 9600 bps

LIN 1.3

Simple Trigger
- □ Start of frame
- □ End of frame
- □ Data frame
- □ Wake up frame
- □ Sync frame

Error Trigger
- □ Sync error
- □ Parity error
- □ Stop bit error
- □ Checksum error

Checksum mode
- ● Classic
- ○ Enhanced

Clause Trigger

State 1 — Event 1

ID: XXh Parity: Xh

Data Length: 4

Data

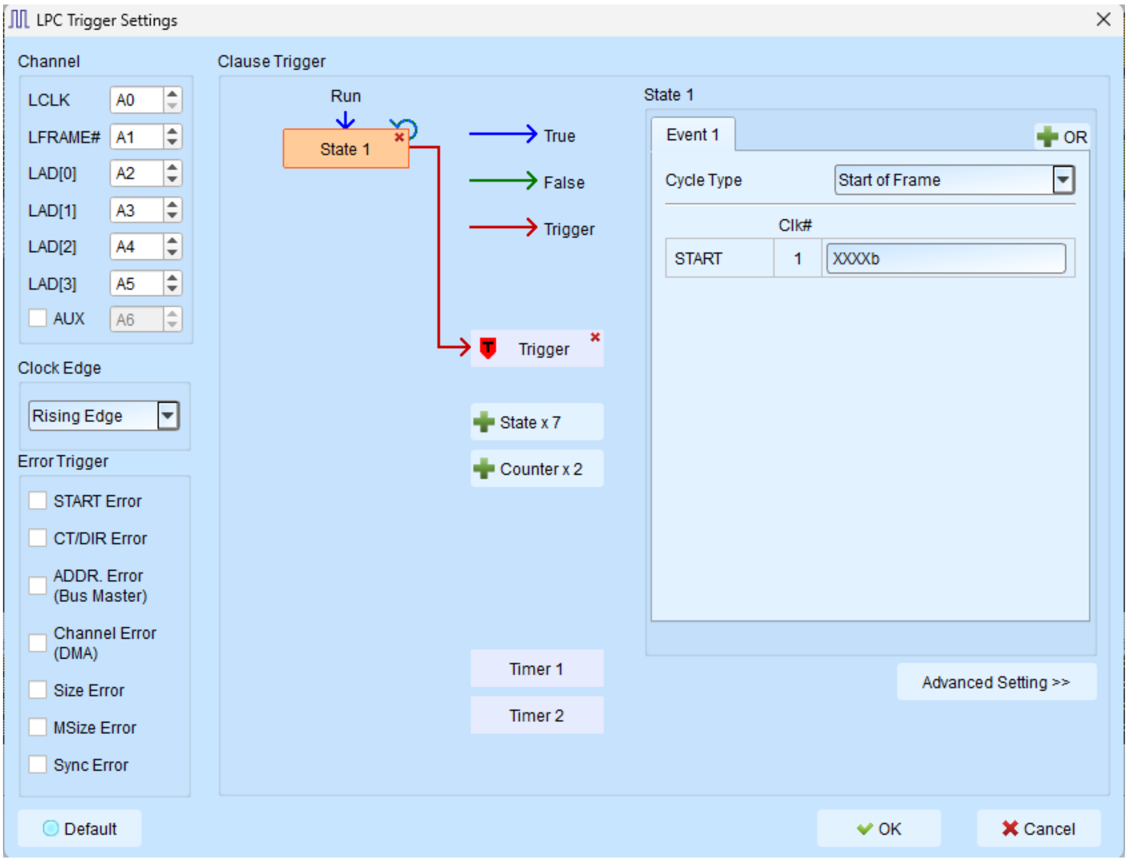
| | | | |
|---|---|---|---|
| Byte 1 | XXh | Byte 2 | XXh |
| Byte 3 | XXh | Byte 4 | XXh |
| Byte 5 | XXh | Byte 6 | XXh |
| Byte 7 | XXh | Byte 8 | XXh |

1. **Channel:** Select channels.
2. **Simple Trigger:** Specific trigger function of LIN.
3. **Error Trigger:** Trigger specific error of LIN.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Show the details of trigger condition in every state as left side; typing or selecting the trigger values in the ID, Parity, Data Length and Data fields, default value is XX means “don’t care”. When select the LIN 2.2, Data Length field will be enabled.

## LPC Trigger

### LPC Trigger Settings

Click LPC Trigger in the toolbar and will show the dialog as the following.



LPC Trigger Settings

Channel

| | |
|---|---|
| LCLK | A0 |
| LFRAME# | A1 |
| LAD[0] | A2 |
| LAD[1] | A3 |
| LAD[2] | A4 |
| LAD[3] | A5 |
| □ AUX | A6 |

Clock Edge: Rising Edge

Error Trigger
- □ START Error
- □ CT/DIR Error
- □ ADDR. Error (Bus Master)
- □ Channel Error (DMA)
- □ Size Error
- □ MSize Error
- □ Sync Error

Clause Trigger

State 1

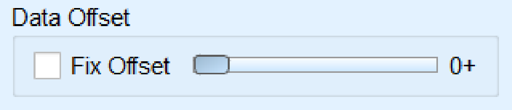
Event 1 | OR

Cycle Type: Start of Frame

| | Clk# | |
|---|---|---|
| START | 1 | XXXXb |

1. **Channel:** Select LPC channels.
2. **Clock Edge:** Select Rising/Falling clock edge.
3. **Error Trigger:** Trigger specific error of LPC.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Show the details of trigger condition in every state as left side; typing or selecting the trigger values in the Cycle Type and its parameters, default value is XX means "don't care". Switch = / ≠ / > / ≤ by click = button.

   Data offset setting:



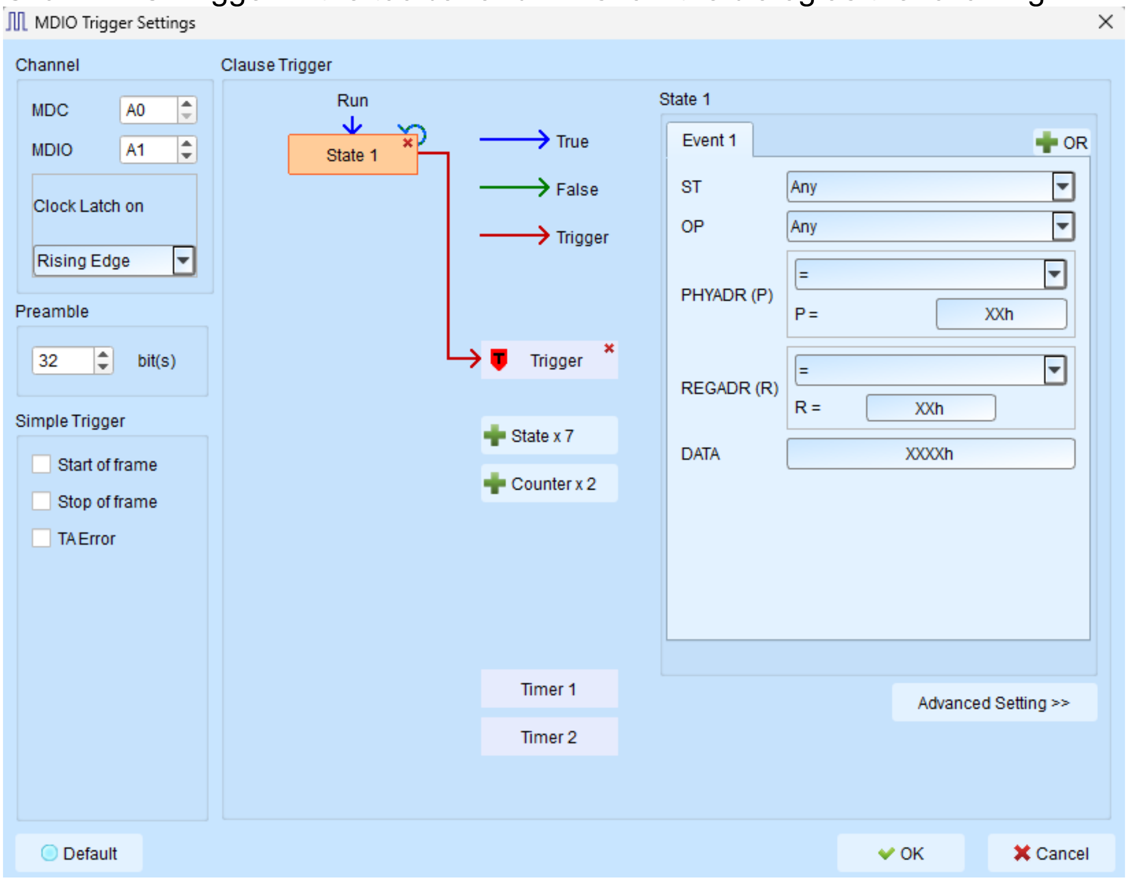
Data Offset

□ Fix Offset 0+

## MDIO Trigger

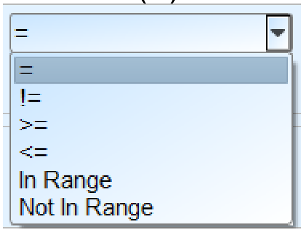
### MDIO Trigger Settings

Click MDIO Trigger in the toolbar and will show the dialog as the following.



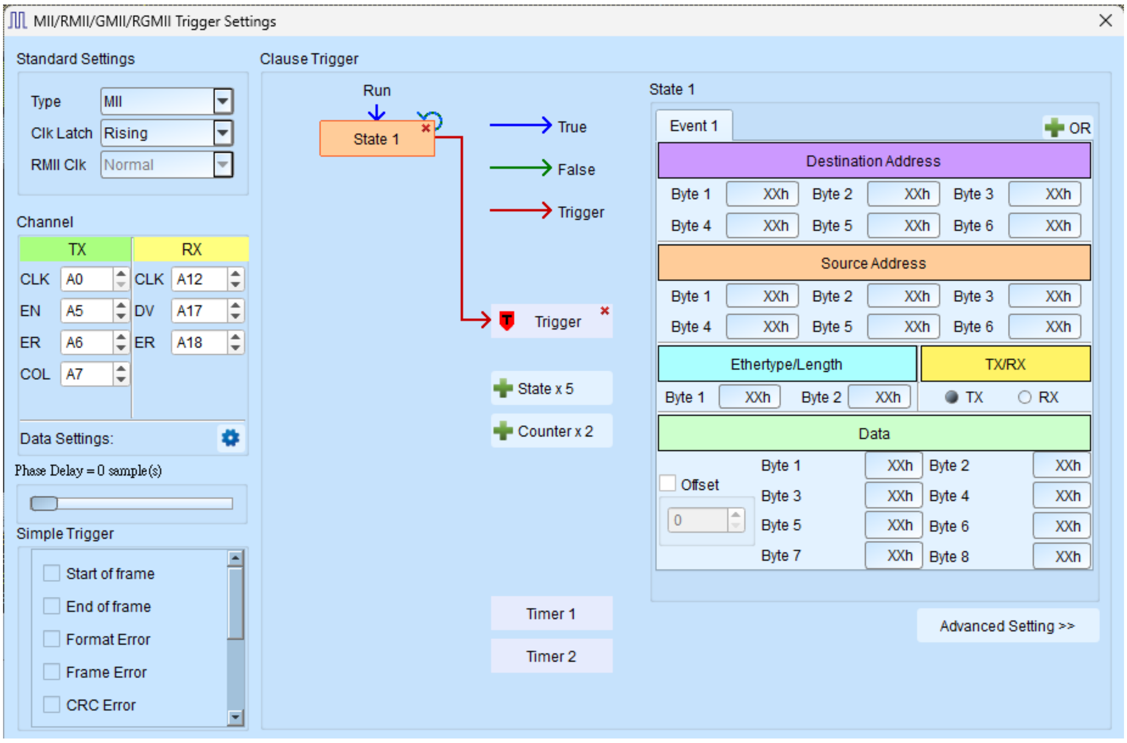
1. **Channel:** Select MDC / MDIO channels.
2. **Preamble:** Select the length of preamble.
3. **Simple Trigger:** Specific trigger function of MDIO.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Show the details of trigger condition in every state as left side; typing or selecting the trigger values in the ST, OP, PHYADR, REGADR and DATA fields, default value is XX means “don’t care”.

   PHYADR(P) / REGADR(R) fields provided the range function:



## MII / RMII / GMII / RGMII Trigger

**BusFinder/LA:** Click MII / RMII / GMII / RGMII Trigger in the toolbar and will show the dialog as the following.



MII/RMII/GMII/RGMII Trigger Settings

Standard Settings

Type: MII
Clk Latch: Rising
RMII Clk: Normal

Channel

| TX | | RX | |
|---|---|---|---|
| CLK | A0 | CLK | A12 |
| EN | A5 | DV | A17 |
| ER | A6 | ER | A18 |
| COL | A7 | | |

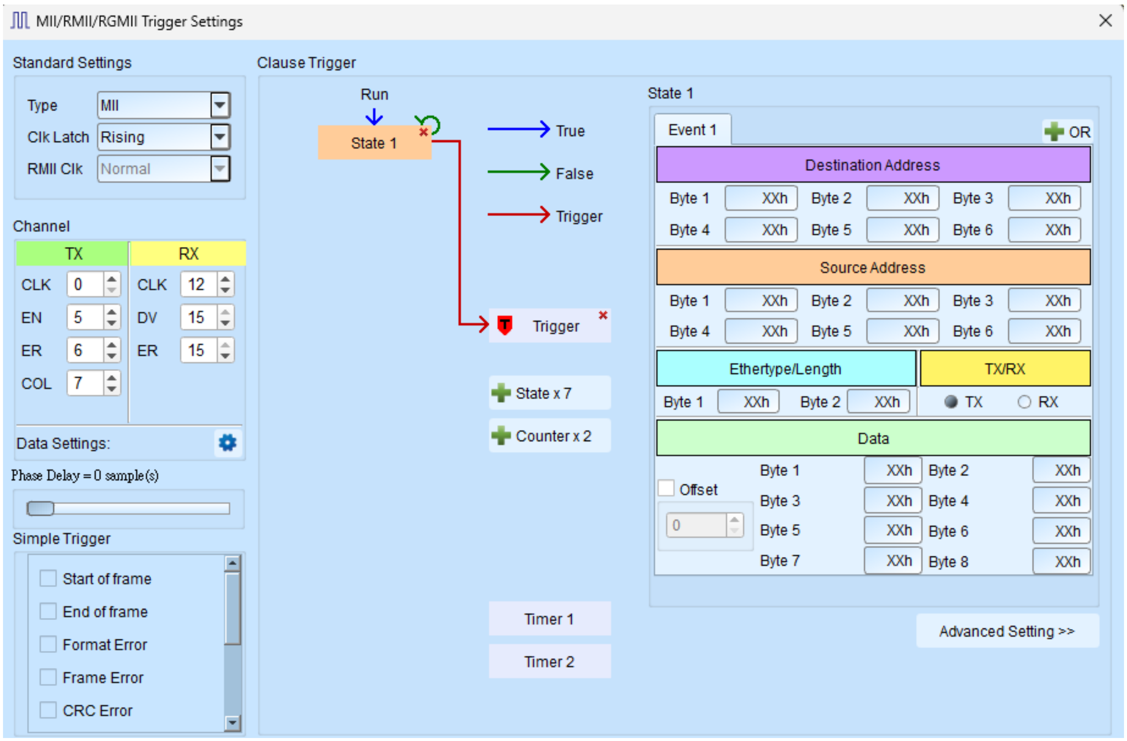
Data Settings:

Phase Delay = 0 sample(s)

Simple Trigger

□ Start of frame
□ End of frame
□ Format Error
□ Frame Error
□ CRC Error

**Travel Logic/MSO:** Click MII / RMII / RGMII Trigger in the toolbar and will show the dialog as the following.



MII/RMII/RGMII Trigger Settings

Standard Settings

Type: MII
Clk Latch: Rising
RMII Clk: Normal

Channel

| TX | | RX | |
|---|---|---|---|
| CLK | 0 | CLK | 12 |
| EN | 5 | DV | 15 |
| ER | 6 | ER | 15 |
| COL | 7 | | |

Data Settings:

Phase Delay = 0 sample(s)

Simple Trigger

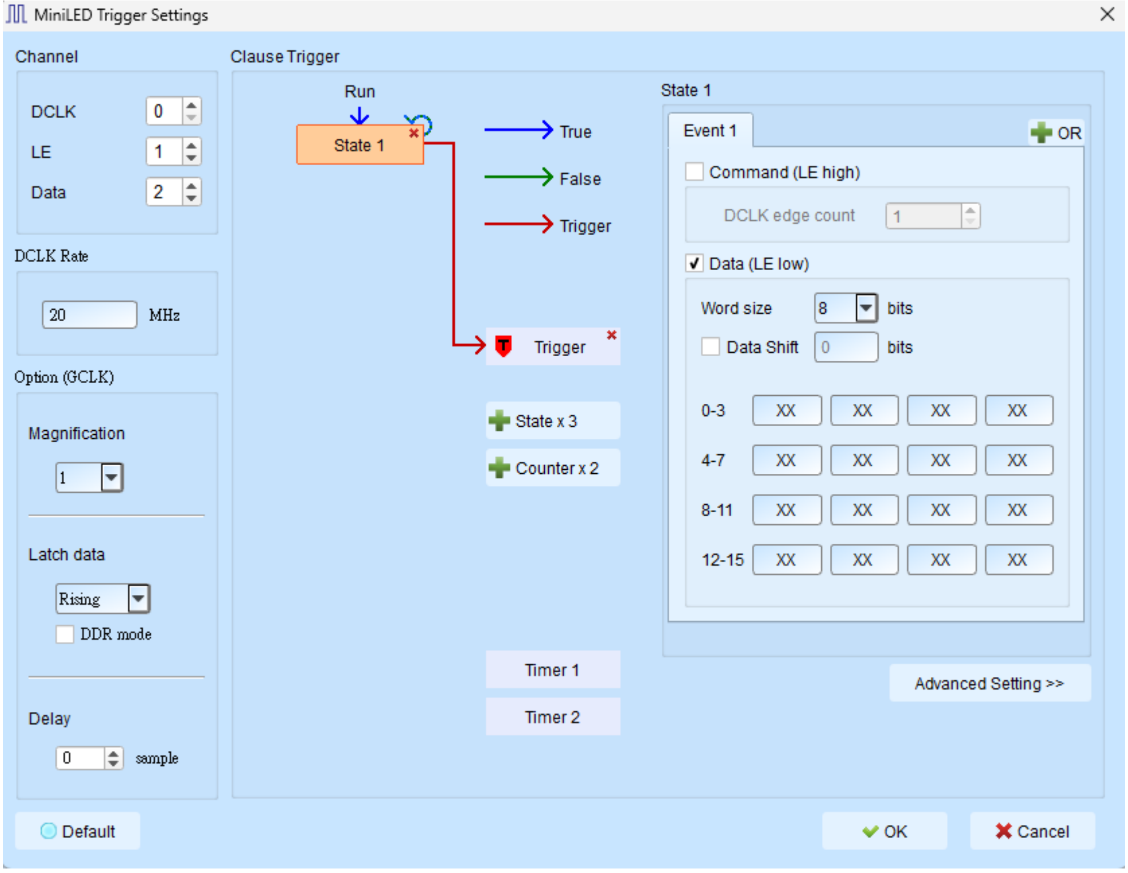
□ Start of frame
□ End of frame
□ Format Error
□ Frame Error
□ CRC Error

1. **Standard Settings:**
   - **I. Type:** Sets the decoding category. GMII measurement is not supported on TravelLogic and MSO, except for BusFinder/LA models. Additionally, to enable RGMII, the sampling rate must be set to 1 GHz or higher.
   - II. **Clk Latch:** Configures whether data is latched on the rising or falling edge of the CLK signal.
   - III. **RMII Clk:** Sets the CLK mode for RMII, applicable only when the measurement category is set to RMII.
2. **Channel:**
   - I. **TX:** Configures the TX channel, excluding the Data channel.
   - **II. RX:** Configures the RX channel, excluding the Data channel.
   - **III. Data Setting:** Assigns channels for TX Data and RX Data.
3. **Phases Delay:** Configures the Phase Delay, measured in sample points.
4. **Simple Trigger:** Provides basic trigger conditions, such as Start of Frame and End of Frame. Enabled when selected.
5. **Clause Trigger:** Please reference Parallel Clause Trigger chapter.
6. **State:** Displays the detailed trigger conditions within each level of the trigger process on the left. Users can enter specific trigger values for Destination Address, Source Address, or use "X" as a wildcard for any value.

## MiniLED Trigger

### MiniLED Trigger Settings

Click MiniLED Trigger in the toolbar and will show the dialog as the following.



MiniLED Trigger Settings

Channel
DCLK 0
LE 1
Data 2

DCLK Rate
20 MHz

Option (GCLK)
Magnification 1
Latch data Rising
□ DDR mode
Delay 0 sample

Clause Trigger

State 1
Event 1
□ Command (LE high)
DCLK edge count 1
☑ Data (LE low)
Word size 8 bits
□ Data Shift 0 bits

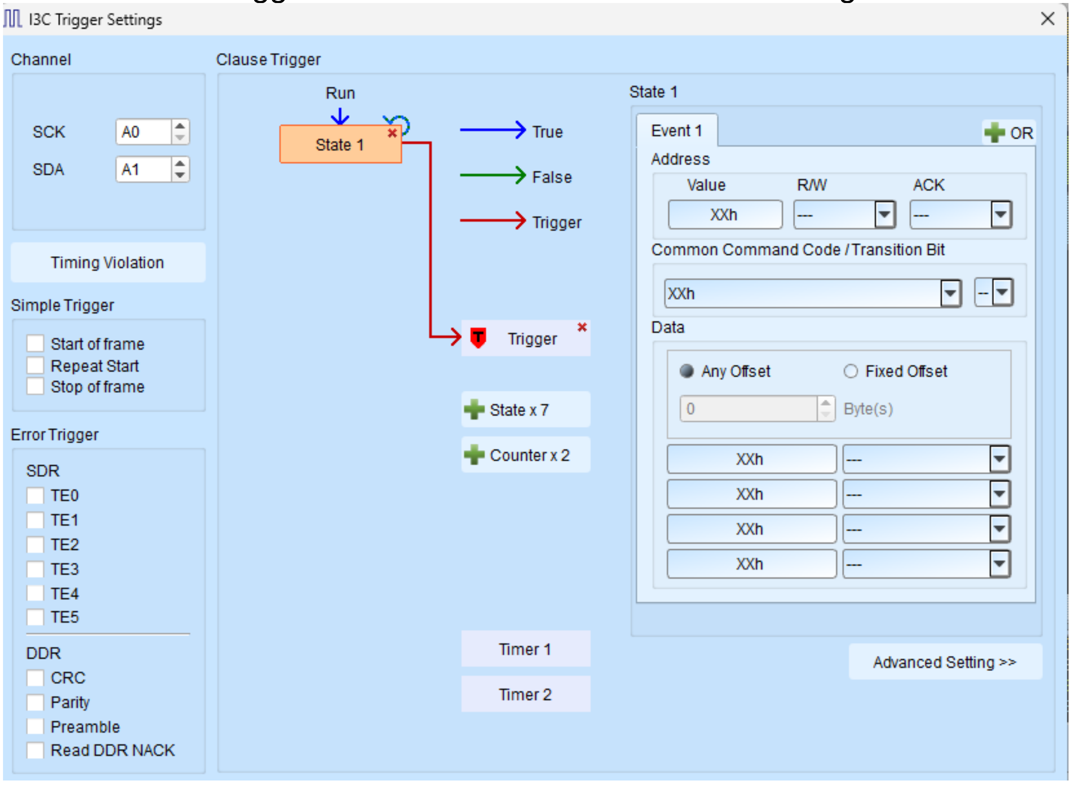
| | | | | |
|---|---|---|---|---|
| 0-3 | XX | XX | XX | XX |
| 4-7 | XX | XX | XX | XX |
| 8-11 | XX | XX | XX | XX |
| 12-15 | XX | XX | XX | XX |

1. **Channel:** Configures the channels for DCLK, LE, and Data.
2. **DCLK Rate:** Sets the speed of DCLK.
3. **Option (GCLK):**
   - I. **Magnification:** Sets the simulation CLK multiplier relative to DCLK.
   - II. **Latch Data:** Determines whether Data is latched on the rising or falling edge.
     - **DDR mode:** Enables DDR mode when selected.
   - III. **Delay:** Configures the delay time, measured in sample points.
4. **Clause Trigger:** Please reference Parallel Clause Trigger chapter.
5. **State:** Displays the detailed trigger conditions within each level of the trigger process on the left. In the Command field, enter the DCLK edge count value, while in the Data field, enter a specific trigger value or use 'X' as a wildcard for any value.

## MIPI I3C Trigger

### MIPI I3C Trigger Settings

Click MIPI I3C Trigger in the toolbar and will show the dialog as the following.



I3C Trigger Settings

Channel

SCK: A0
SDA: A1

Simple Trigger

- □ Start of frame
- □ Repeat Start
- □ Stop of frame

Error Trigger

SDR

- □ TE0
- □ TE1
- □ TE2
- □ TE3
- □ TE4
- □ TE5

DDR

- □ CRC
- □ Parity
- □ Preamble
- □ Read DDR NACK

State 1 – Event 1

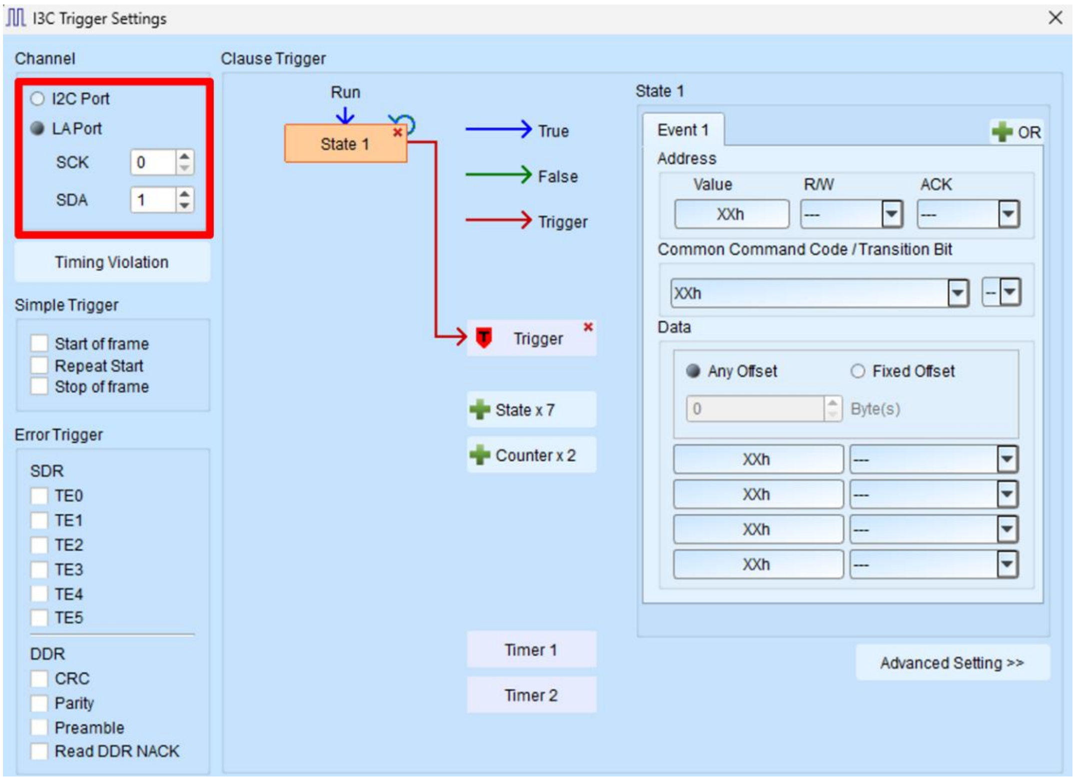
Address: Value XXh, R/W ---, ACK ---

Common Command Code / Transition Bit: XXh, --

Data: ● Any Offset ○ Fixed Offset 0 Byte(s)

| | |
|---|---|
| XXh | --- |
| XXh | --- |
| XXh | --- |
| XXh | --- |

## Travel Bus series



I3C Trigger Settings

Channel

○ I2C Port
● LA Port
SCK: 0
SDA: 1

Simple Trigger

- □ Start of frame
- □ Repeat Start
- □ Stop of frame

Error Trigger

SDR

- □ TE0
- □ TE1
- □ TE2
- □ TE3
- □ TE4
- □ TE5

DDR

- □ CRC
- □ Parity
- □ Preamble
- □ Read DDR NACK

State 1 – Event 1

Address: Value XXh, R/W ---, ACK ---

Common Command Code / Transition Bit: XXh, --

Data: ● Any Offset ○ Fixed Offset 0 Byte(s)

| | |
|---|---|
| XXh | --- |
| XXh | --- |
| XXh | --- |
| XXh | --- |

**Channel:** Configure the I3C channel using either I2C Port or LA Port.

**Simple Trigger:** Set triggers for I3C Start, Repeat Start, or Stop events.

**State:** Categorized into Address, Common Command Code (CCC), and Data.

For the Data parameter, two offset options are available:

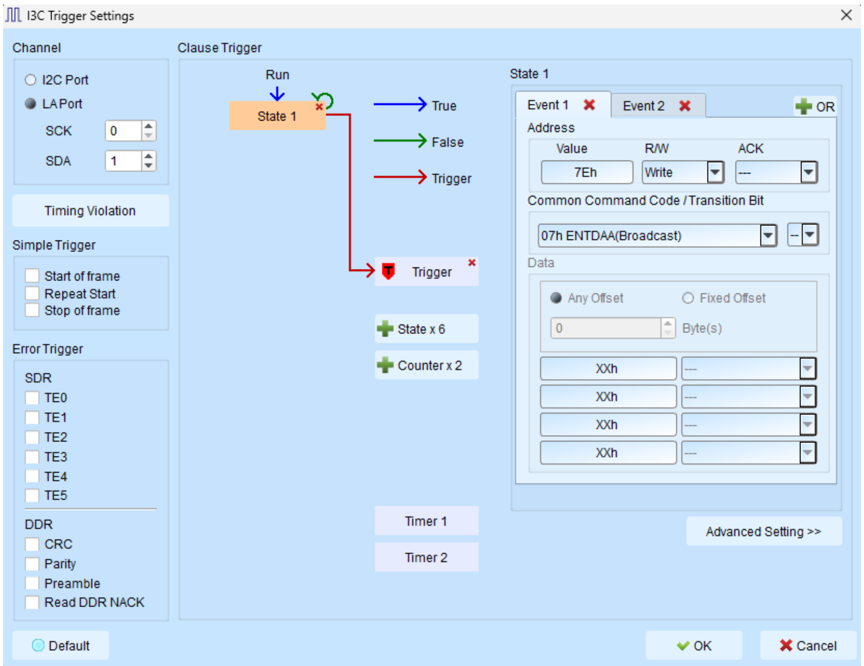
Any Offset (default): Triggers when the first detected I3C data matches the specified value.

**Fixed Offset:** Specifies both the I3C data value and its exact position (offset). An offset of 0 triggers on the first data byte, with the offset unit in bytes.

Since some I3C CCCs contain specially formatted data, the following describes their specific trigger settings:

1. Triggering on RSTDAA (06h) / ENTDAA (07h) data

   a. Event 1 CCC is set to RSTDAA (06h) / ENTDAA (07h).



I3C Trigger Settings

Channel: ○ I2C Port ● LA Port; SCK 0; SDA 1

Simple Trigger: □ Start of frame □ Repeat Start □ Stop of frame

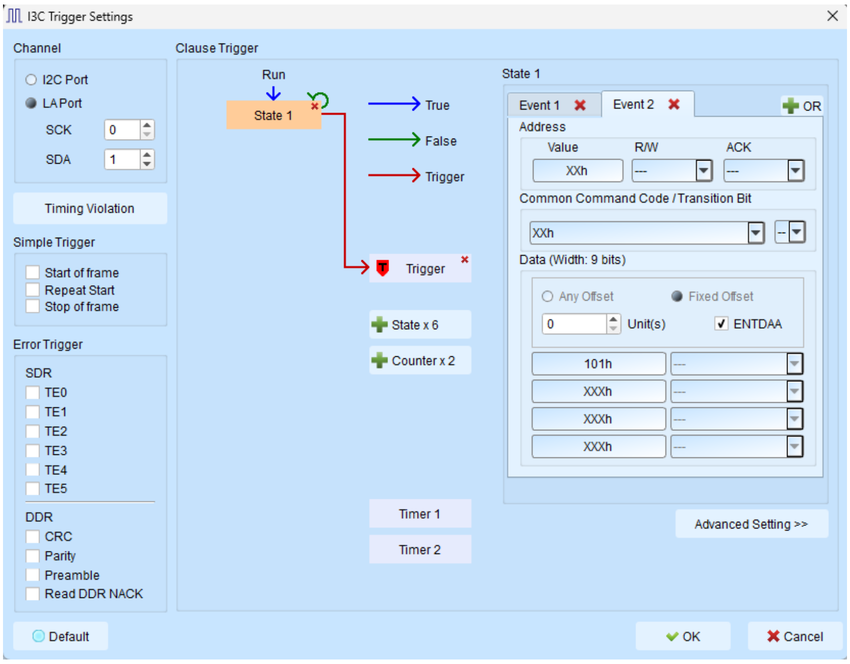
Error Trigger: SDR □ TE0 □ TE1 □ TE2 □ TE3 □ TE4 □ TE5; DDR □ CRC □ Parity □ Preamble □ Read DDR NACK

State 1 – Event 1: Address Value 7Eh, R/W Write, ACK ---

Common Command Code / Transition Bit: 07h ENTDAA(Broadcast), --

Data: ● Any Offset ○ Fixed Offset; 0 Byte(s); XXh ---; XXh ---; XXh ---; XXh ---

   b. Event 2: When selecting ENTDAA, the entered data width must be 9 bits and is fixed to Fixed Offset.



I3C Trigger Settings

Channel: ○ I2C Port ● LA Port; SCK 0; SDA 1

Simple Trigger: □ Start of frame □ Repeat Start □ Stop of frame

Error Trigger: SDR □ TE0 □ TE1 □ TE2 □ TE3 □ TE4 □ TE5; DDR □ CRC □ Parity □ Preamble □ Read DDR NACK

State 1 – Event 1, Event 2

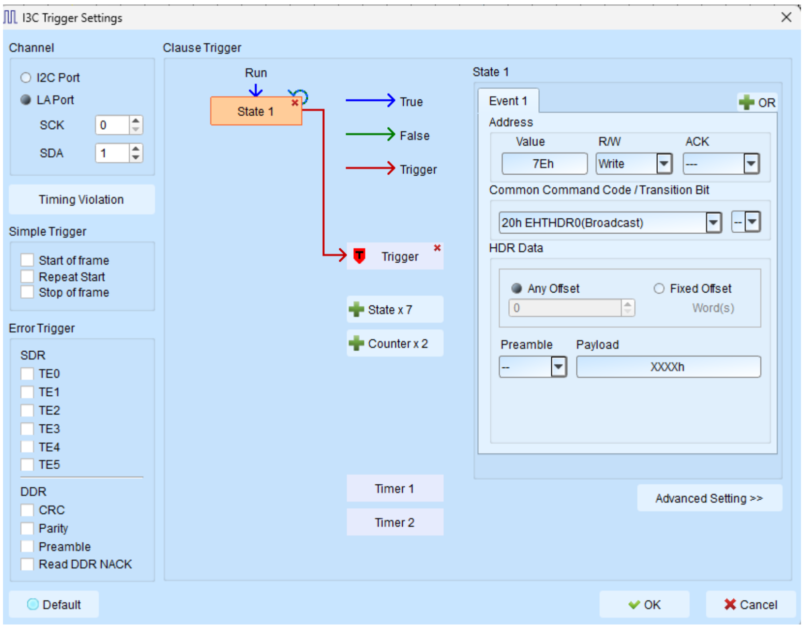
Address: Value XXh; R/W ---; ACK ---

Common Command Code / Transition Bit: XXh; --

Data (Width: 9 bits): ○ Any Offset ● Fixed Offset; 0 Unit(s); ☑ ENTDAA

| | |
|---|---|
| 101h | --- |
| XXXh | --- |
| XXXh | --- |
| XXXh | --- |

2. Triggering on EHTHDR0 (20h) / EHTHDR1 (21h) / EHTHDR2 (22h) HDR Data: The data format consists of a 16-bit payload.



I3C Trigger Settings

Channel: ○ I2C Port ● LA Port; SCK 0; SDA 1

Simple Trigger: □ Start of frame □ Repeat Start □ Stop of frame

Error Trigger: SDR □ TE0 □ TE1 □ TE2 □ TE3 □ TE4 □ TE5; DDR □ CRC □ Parity □ Preamble □ Read DDR NACK

State 1 – Event 1

Address: Value 7Eh; R/W Write; ACK ---

Common Command Code / Transition Bit: 20h EHTHDR0(Broadcast); --

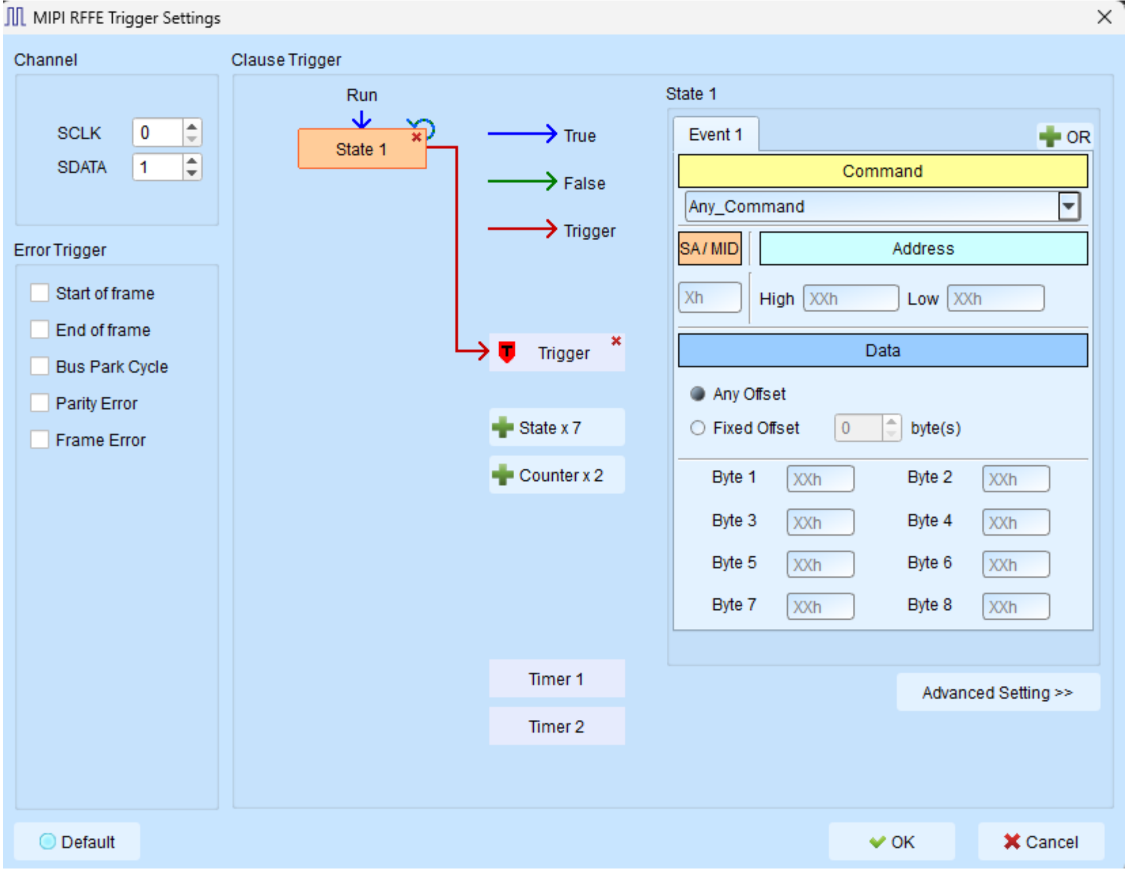
HDR Data: ● Any Offset ○ Fixed Offset; 0 Word(s)

Preamble --; Payload XXXXh

## MIPI RFFE Trigger

### MIPI RFFE Trigger Settings

Click MIPI RFFE Trigger in the toolbar and will show the dialog as the following.



MIPI RFFE Trigger Settings

Channel

SCLK 0

SDATA 1

Error Trigger

□ Start of frame

□ End of frame

□ Bus Park Cycle

□ Parity Error

□ Frame Error

State 1

Event 1

Command: Any_Command

SA / MID: Xh

Address: High XXh Low XXh

Data

● Any Offset

○ Fixed Offset 0 byte(s)

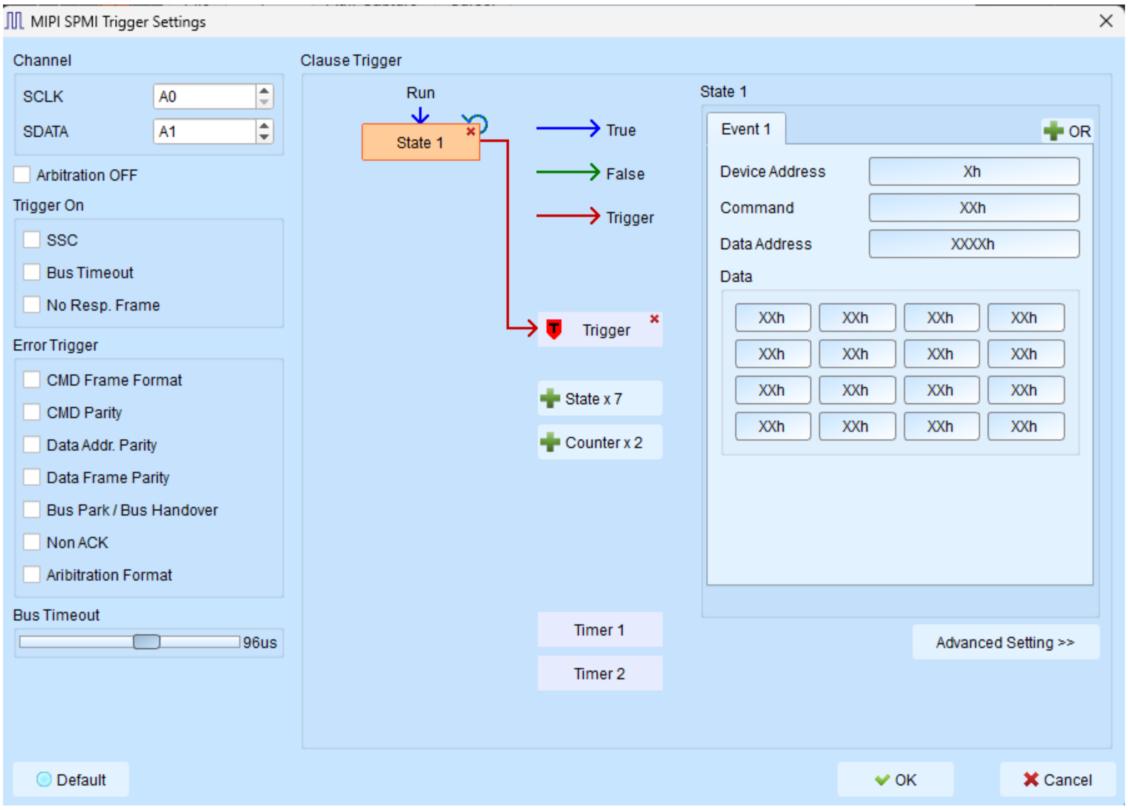
| | | | |
|---|---|---|---|
| Byte 1 | XXh | Byte 2 | XXh |
| Byte 3 | XXh | Byte 4 | XXh |
| Byte 5 | XXh | Byte 6 | XXh |
| Byte 7 | XXh | Byte 8 | XXh |

1. **Channel:** Configure the channels for SCLK and SDATA.
2. **Error Trigger:** Provides basic trigger conditions, such as Start of Frame and End of Frame. Enabled when selected.
3. **Clause Trigger:** Please reference Parallel Clause Trigger chapter.
4. **State:** Displays the detailed trigger conditions within each level of the trigger process on the left. Users can configure the Command type, enter a specific Address trigger value, or specify a Data trigger value, with the option to use 'X' as a wildcard for any value.

## MIPI SPMI Trigger

### MIPI SPMI Trigger Settings

Click MIPI SPMI Trigger in the toolbar and will show the dialog as the following.



MIPI SPMI Trigger Settings

Channel

SCLK: A0

SDATA: A1

□ Arbitration OFF

Trigger On

□ SSC

□ Bus Timeout

□ No Resp. Frame

Error Trigger

□ CMD Frame Format

□ CMD Parity

□ Data Addr. Parity

□ Data Frame Parity

□ Bus Park / Bus Handover

□ Non ACK

□ Aribitration Format

Bus Timeout: 96us

State 1 — Event 1

Device Address: Xh

Command: XXh

Data Address: XXXXh

Data:

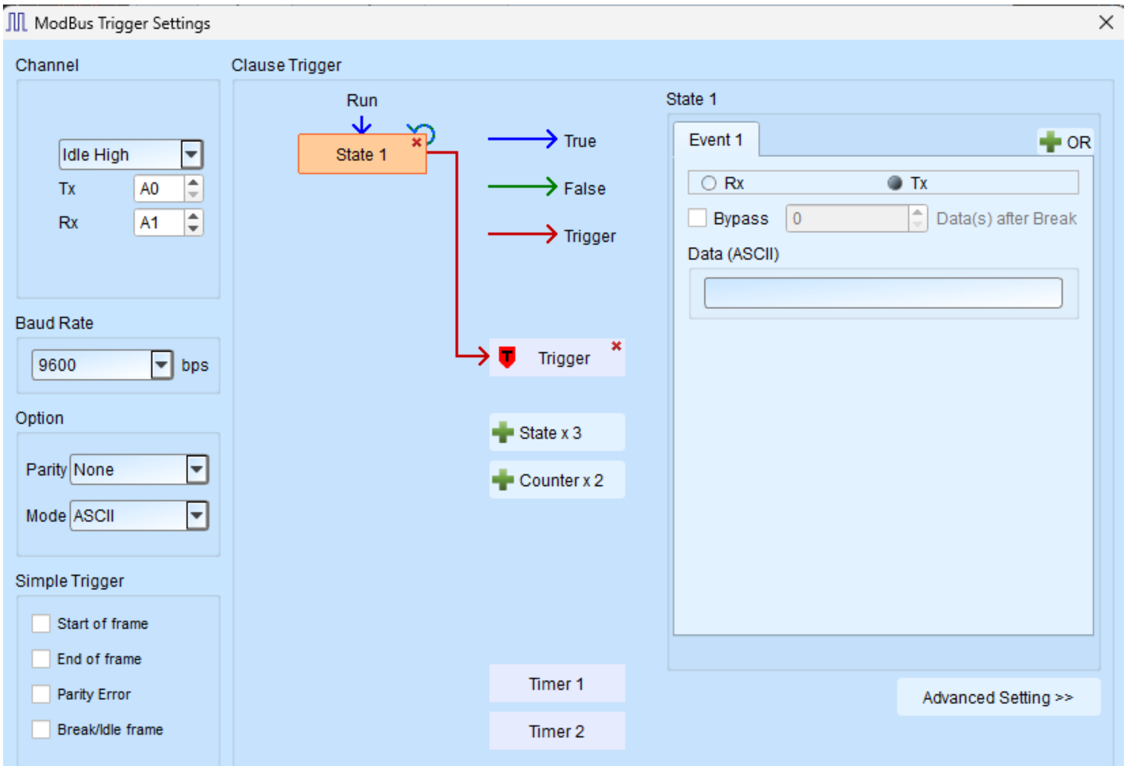
|  |  |  |  |
|---|---|---|---|
| XXh | XXh | XXh | XXh |
| XXh | XXh | XXh | XXh |
| XXh | XXh | XXh | XXh |
| XXh | XXh | XXh | XXh |

1. **Channel:** Configure the channels for SCLK and SDATA.
2. **Arbitration OFF:** Disables Arbitration. Enabled when selected.
3. **Trigger On:** Configure specific special trigger conditions. Enabled when selected.
4. **Error Trigger:** Provides basic trigger conditions, such as CMD Frame Format and CMD Parity. Enabled when selected.
5. **Bus Timeout:** Set the timeout duration.
6. **Clause Trigger:** Please reference Parallel Clause Trigger chapter.
7. **State:** Displays the detailed trigger conditions within each level of the trigger process on the left. Users can configure the Command type, enter a specific Address trigger value, or specify a Data trigger value, with the option to use 'X' as a wildcard for any value.

## ModBus Trigger

### ModBus Trigger Settings

Click ModBus Trigger in the toolbar and will show the dialog as the following.



ModBus Trigger Settings

Channel: Idle High; Tx A0; Rx A1

Baud Rate: 9600 bps

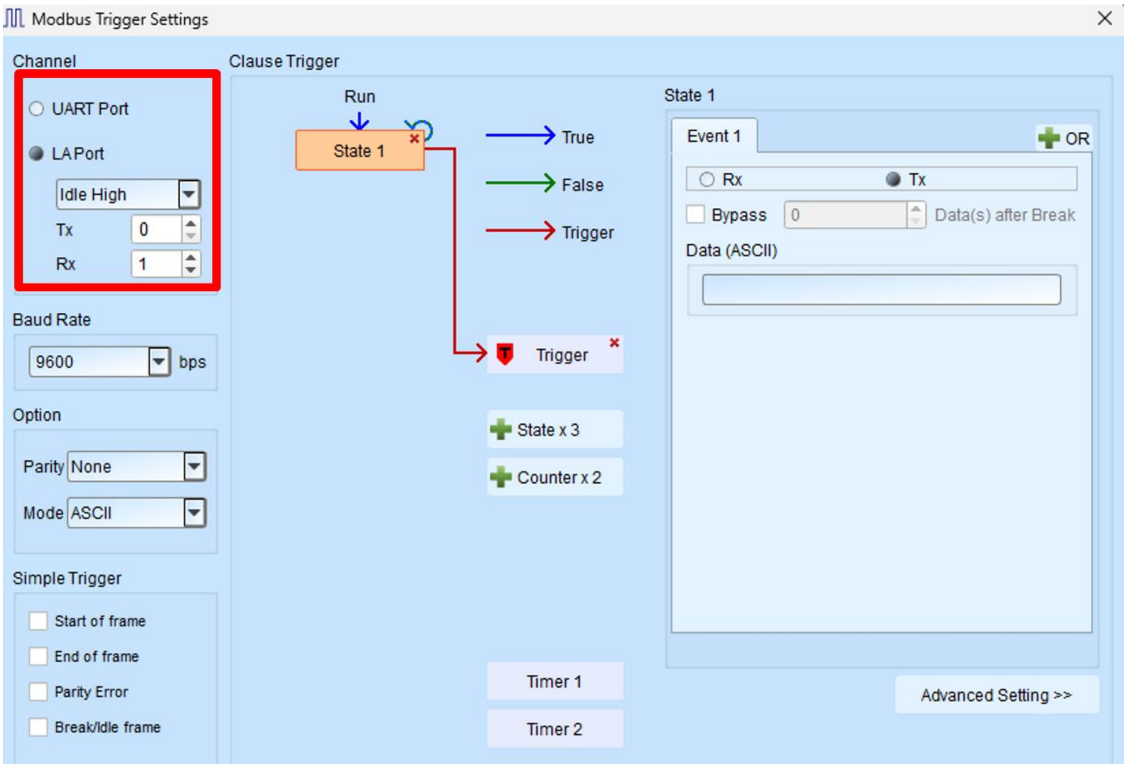
Option: Parity None; Mode ASCII

Simple Trigger: □ Start of frame □ End of frame □ Parity Error □ Break/Idle frame

Clause Trigger

State 1: Event 1; ○ Rx ● Tx; □ Bypass 0 Data(s) after Break; Data (ASCII)

Travel Bus series



Modbus Trigger Settings

Channel: ○ UART Port ● LA Port; Idle High; Tx 0; Rx 1

Baud Rate: 9600 bps

Option: Parity None; Mode ASCII

Simple Trigger: □ Start of frame □ End of frame □ Parity Error □ Break/Idle frame

Clause Trigger

State 1: Event 1; ○ Rx ● Tx; □ Bypass 0 Data(s) after Break; Data (ASCII)

***Since TravelBus has specially designed channels for UART, the ModBus settings in the TravelBus software will include additional channel source options.***

**Channel**

UART Port / LA Port : Select ModBus channels.

**Baud Rate**

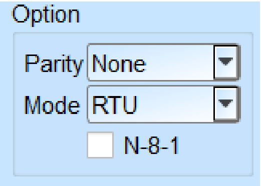
Select the baud rate and providing the manual input if no appropriate selection.

**Option**

Parity: Provide None/Odd/Even selections, none parity defaulted.

Mode: Provide ASCII/RTU mode, ASCII mode defaulted.

Support the 8-N-1 protocol under RTU mode, 8-N-2 protocol defaulted.



Option

Parity None

Mode RTU

□ N-8-1

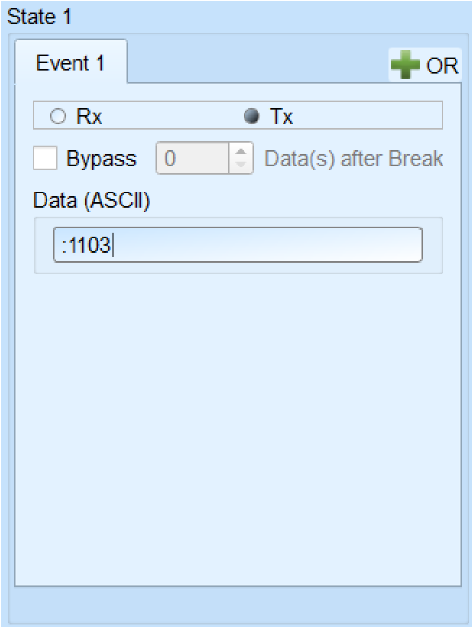
**Simple Trigger:**

Provide Start of frame, End of frame, Parity Error, Break/Idle frame triggers。

**State:**

Provide ASCII/HEX input mode by ASCII/RTU mode.

ASCII



State 1

Event 1

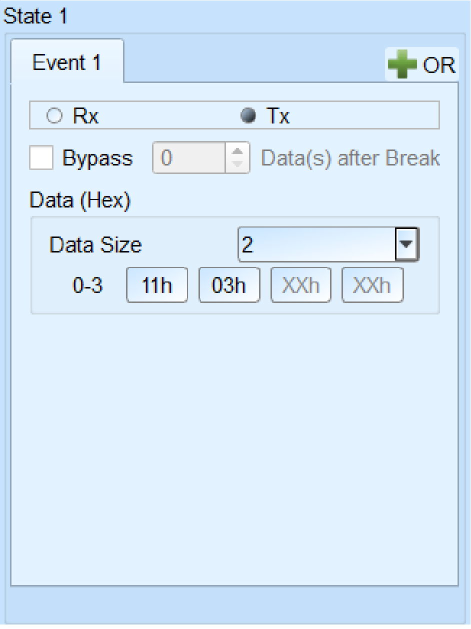
○ Rx ● Tx

□ Bypass 0 Data(s) after Break

Data (ASCII)

:1103

RTU



State 1

Event 1

○ Rx ● Tx

□ Bypass 0 Data(s) after Break

Data (Hex)

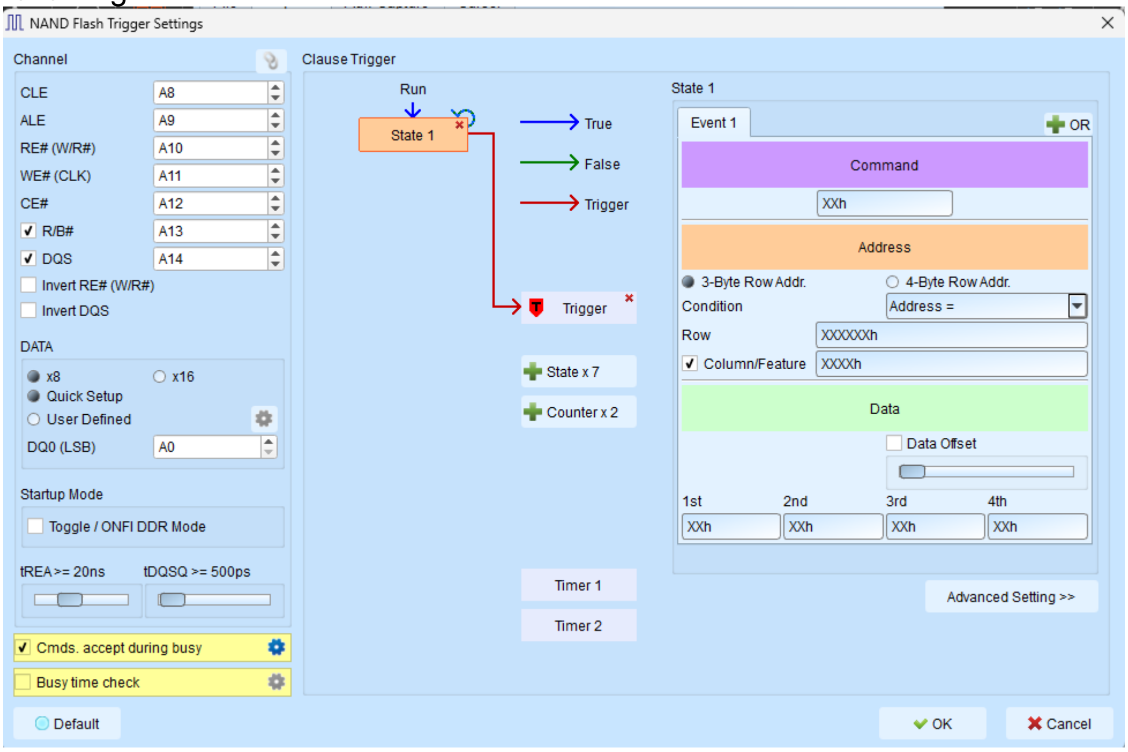
Data Size 2

0-3 11h 03h XXh XXh

## NAND Flash Trigger

### NAND Flash Trigger Settings

Click NAND Flash Trigger in the toolbar and will show the dialog as the following.



NAND Flash Trigger Settings

Channel

| | |
|---|---|
| CLE | A8 |
| ALE | A9 |
| RE# (W/R#) | A10 |
| WE# (CLK) | A11 |
| CE# | A12 |
| ☑ R/B# | A13 |
| ☑ DQS | A14 |

□ Invert RE# (W/R#)
□ Invert DQS

DATA
● x8 ○ x16
● Quick Setup
○ User Defined
DQ0 (LSB): A0

Startup Mode
□ Toggle / ONFI DDR Mode

tREA>= 20ns tDQSQ >= 500ps

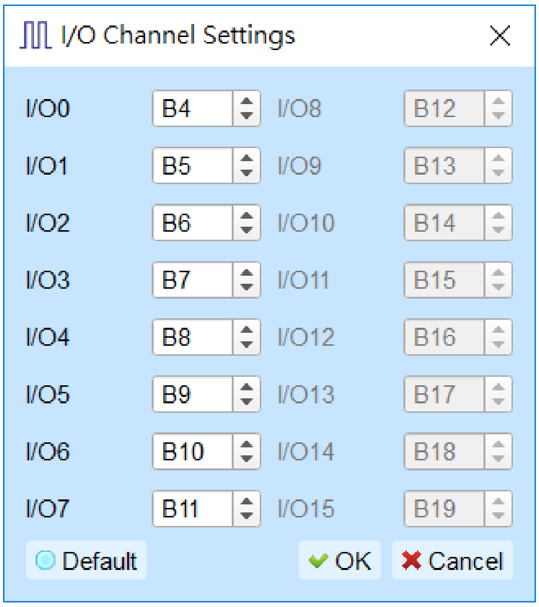
☑ Cmds. accept during busy
□ Busy time check

Clause Trigger

State 1
Event 1 OR
Command: XXh
Address
● 3-Byte Row Addr. ○ 4-Byte Row Addr.
Condition: Address =
Row: XXXXXXh
☑ Column/Feature: XXXXh
Data
□ Data Offset

| 1st | 2nd | 3rd | 4th |
|---|---|---|---|
| XXh | XXh | XXh | XXh |

1. **Channel:** Select CLE, ALE, RE, WE, CE, R/B, DQS channels.
2. **DATA:** Select x8 or x16 bits
   - I. **Quick Setup:** Only set DQ0(LSB), others will be set automatically. (e.g. LSB = B4, MSB = B11; LSB = B7, MSB = B14)
   - II. **User Defined:**

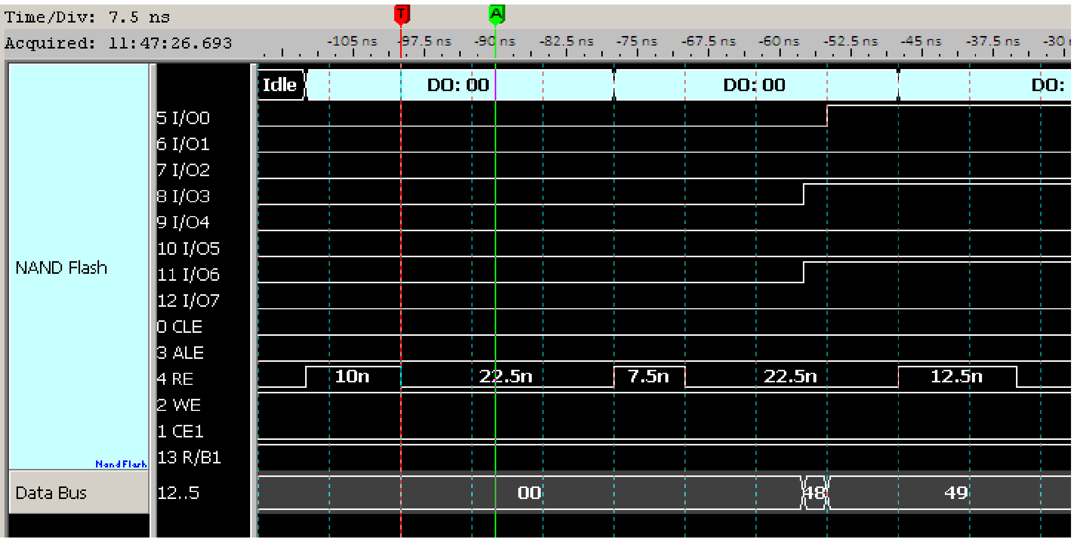


I/O Channel Settings

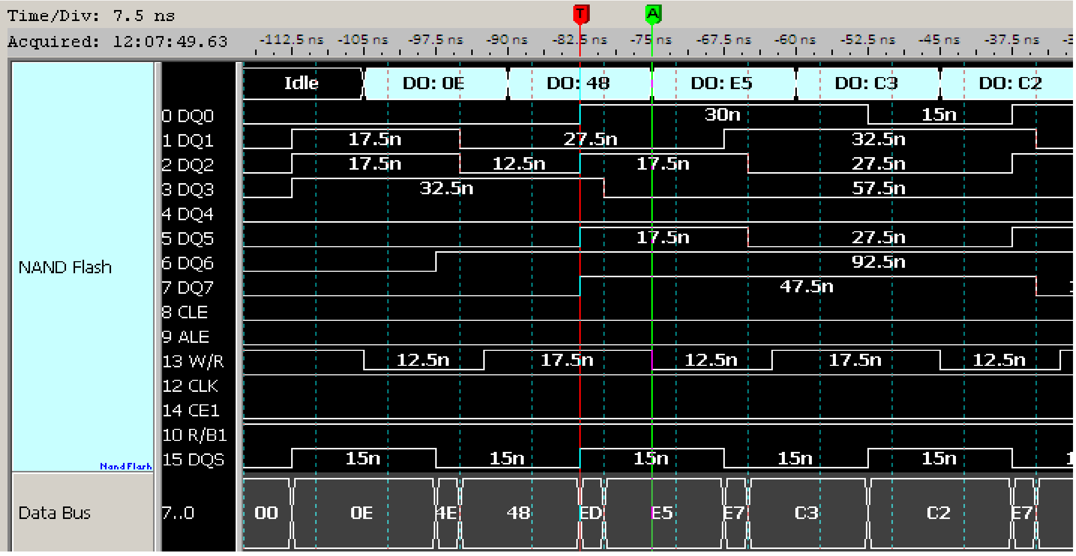
| | | | |
|---|---|---|---|
| I/O0 | B4 | I/O8 | B12 |
| I/O1 | B5 | I/O9 | B13 |
| I/O2 | B6 | I/O10 | B14 |
| I/O3 | B7 | I/O11 | B15 |
| I/O4 | B8 | I/O12 | B16 |
| I/O5 | B9 | I/O13 | B17 |
| I/O6 | B10 | I/O14 | B18 |
| I/O7 | B11 | I/O15 | B19 |

3. **Startup Mode:** Set DDR mode.
4. **tREA / tDQSQ:** Set tREA at SDR mode and tDQSQ at DDR mode.

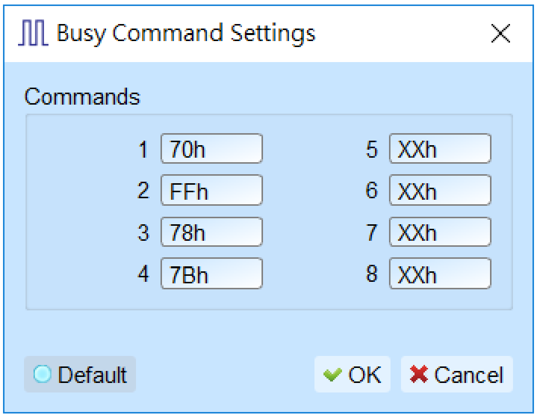
tREA:



tDQSQ:



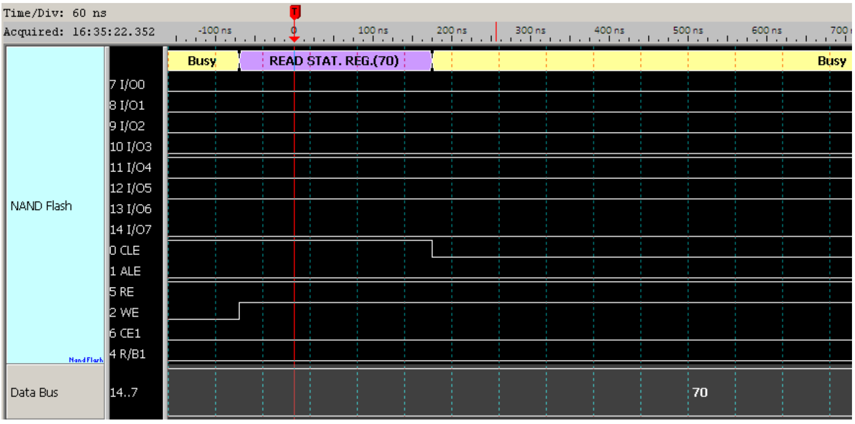
5. **Commands accepted during busy:** Set NAND commands still could be triggered during busy state, default is 70h/FFh/78h/7Bh.



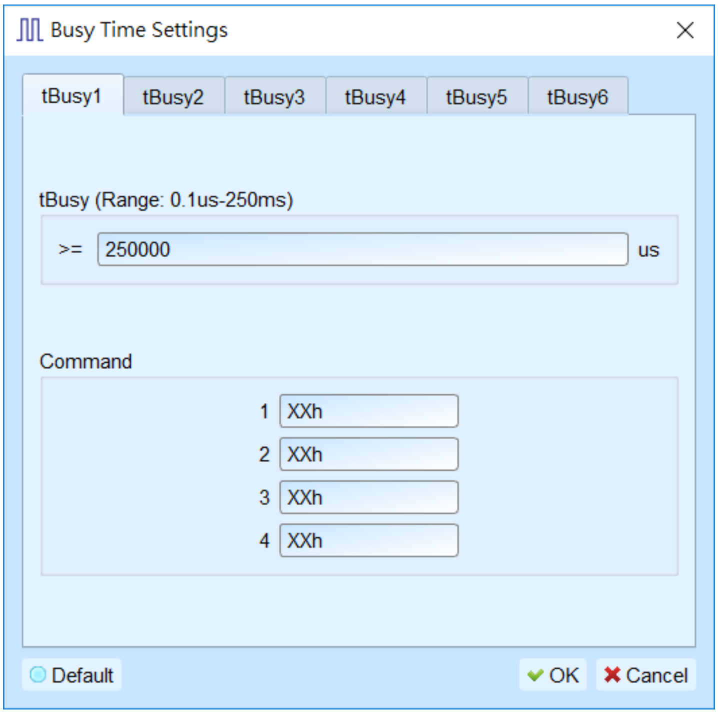
Commands

| | | | |
|---|---|---|---|
| 1 | 70h | 5 | XXh |
| 2 | FFh | 6 | XXh |
| 3 | 78h | 7 | XXh |
| 4 | 7Bh | 8 | XXh |

**Trigger 70h during busy state:**



6. **Busy time check:** Triggers when timing >= tBusy, providing 6 tBusy to check.



tBusy (Range: 0.1us-250ms)

>= 250000 us

Command

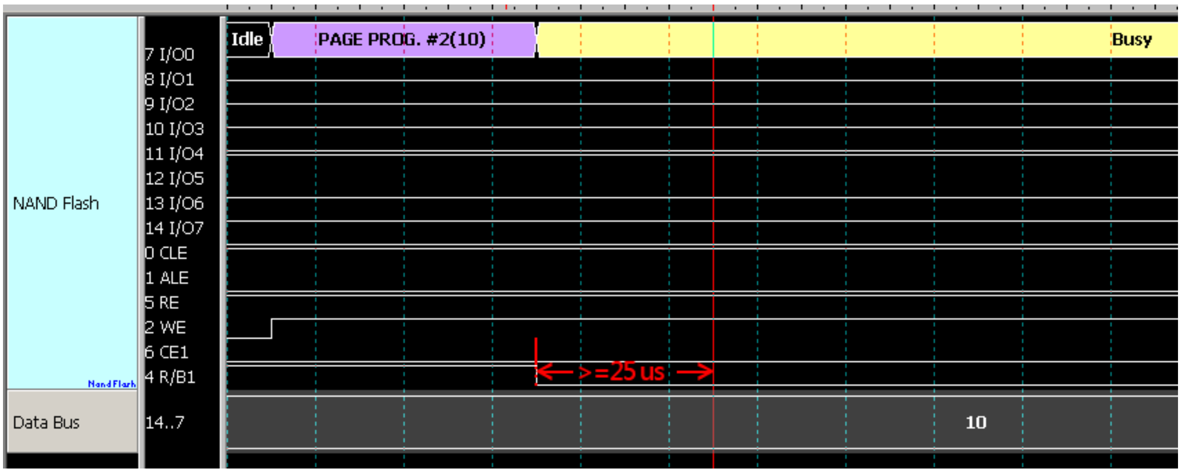
1 XXh

2 XXh

3 XXh

4 XXh

**Trigger the tBusy >= 25 us after NAND command (10h)**

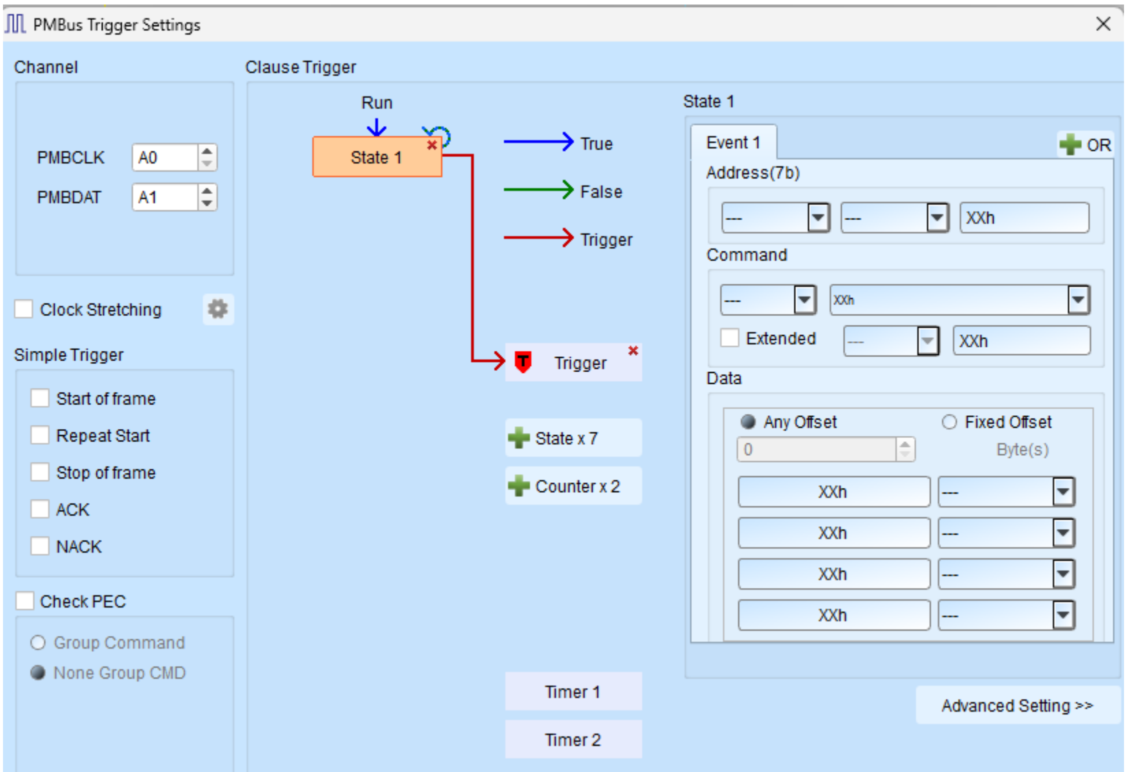


7. **Clause Trigger:** Please reference Clause Trigger chapter.
8. **State:** Show the details of trigger condition in every state as left side; selecting the trigger values in the Command, Address and Data fields, default value is XX means “don’t care”.

## PMBus Trigger

### PMBus Trigger Settings

Click PMBus Trigger in the toolbar and will show the dialog as the following.



PMBus Trigger Settings

Channel

PMBCLK A0

PMBDAT A1

□ Clock Stretching

Simple Trigger

□ Start of frame

□ Repeat Start

□ Stop of frame

□ ACK

□ NACK

□ Check PEC

○ Group Command

● None Group CMD

Clause Trigger

State 1

Event 1 OR

Address(7b): --- --- XXh

Command: --- XXh

□ Extended --- XXh

Data

● Any Offset ○ Fixed Offset

0 Byte(s)

XXh ---

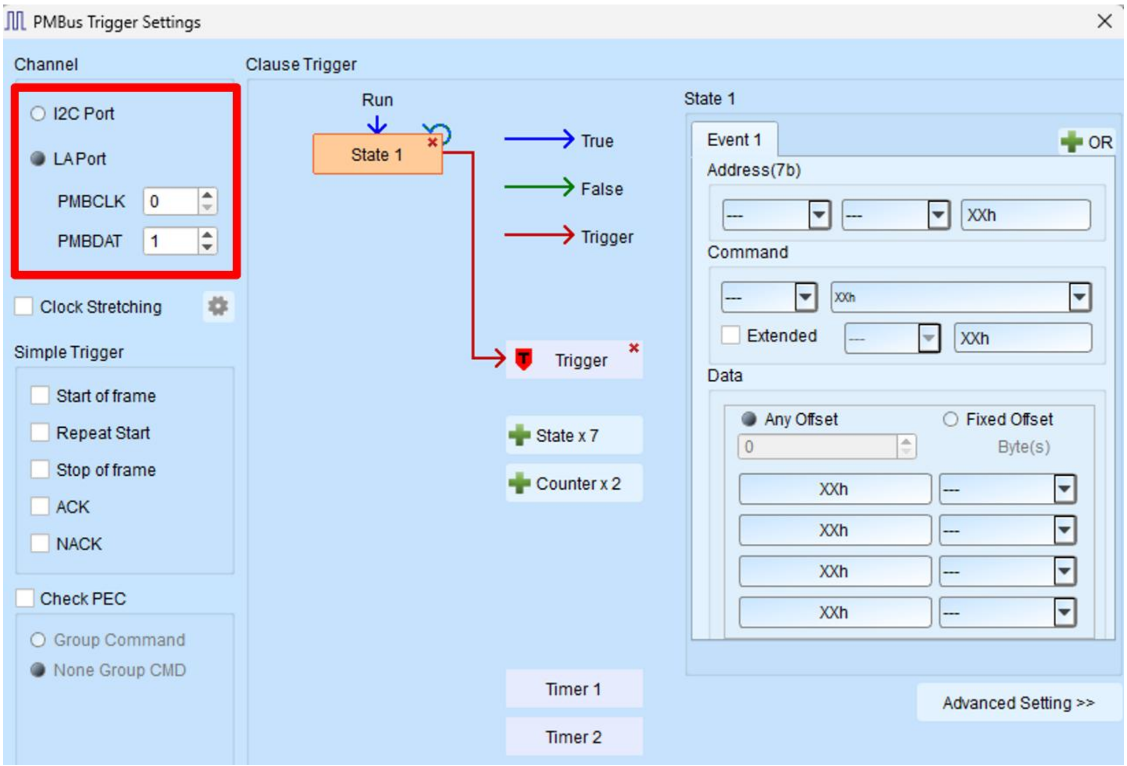
XXh ---

XXh ---

XXh ---

Advanced Setting >>

Travel Bus series



PMBus Trigger Settings

Channel

○ I2C Port

● LA Port

PMBCLK 0

PMBDAT 1

□ Clock Stretching

Simple Trigger

□ Start of frame

□ Repeat Start

□ Stop of frame

□ ACK

□ NACK

□ Check PEC

○ Group Command

● None Group CMD

Clause Trigger

State 1

Event 1 OR

Address(7b): --- --- XXh

Command: --- XXh

□ Extended --- XXh

Data

● Any Offset ○ Fixed Offset

0 Byte(s)

XXh ---

XXh ---

XXh ---

XXh ---

Advanced Setting >>

***Since TravelBus has specially designed channels for I2C, the PMBus settings in the TravelBus software will include additional channel source options.***

1. **Channel:** Configure the I2C interface or LA channel.
2. **Simple Trigger:** Set PMBus-specific frame trigger.
3. **Check PEC:** Configure the trigger for Packet Error Checking (PEC).
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Displays the detailed trigger conditions within each level of the trigger process on the left. Users can enter specific trigger values for Address, Command, and Data, or use 'X' as a wildcard for any value.

   **I.** The Data field allows up to 4 Bytes to be set. Unused fields should be filled with XXh to indicate any value.

   **II.** The input field can contain the required trigger Data or 'X' as a wildcard.

   - **Hexadecimal** values should end with **'h'**.
   - **Binary** values should end with **'b'**.
   - **Decimal** values do not require a suffix.

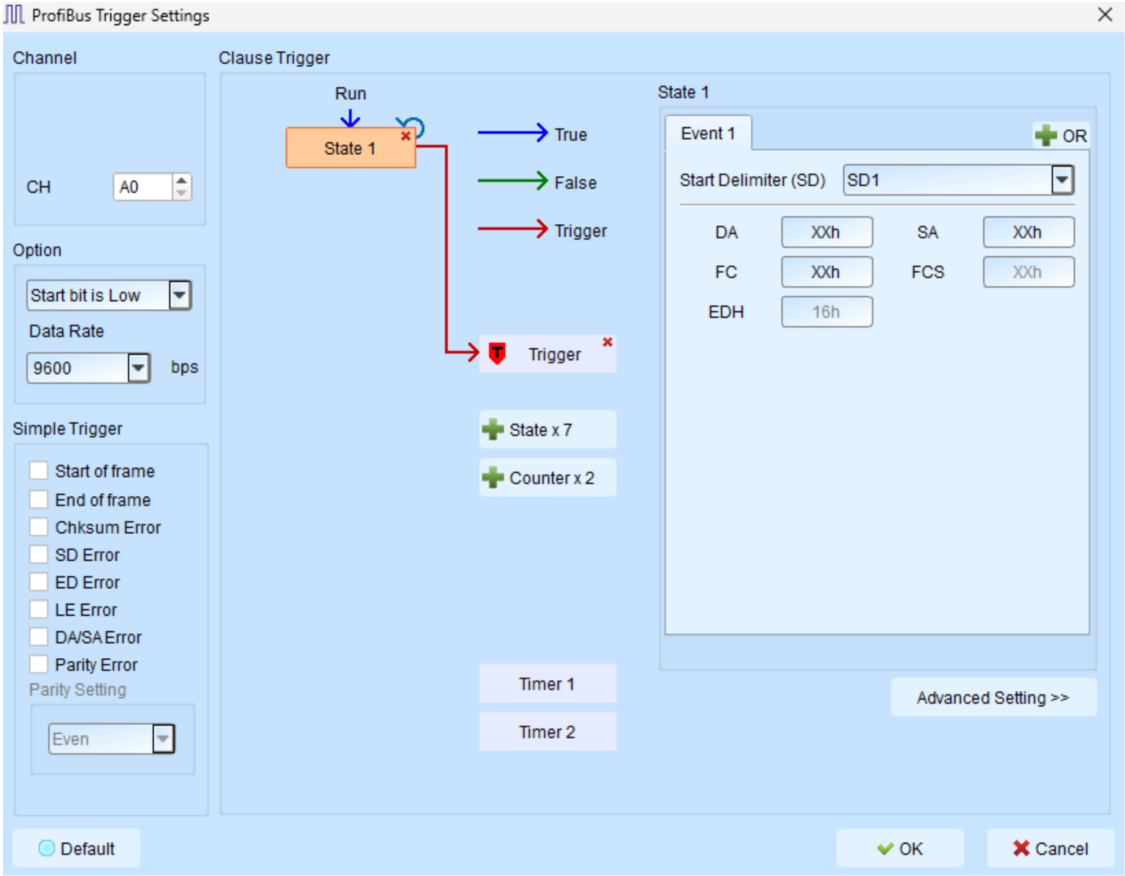
   **III. Data Offset Triggering:**

   - **Any Offset:** Triggers when valid Data that meets the set conditions appears anywhere in the Data field, regardless of offset.
   - **Fixed Offset:** Triggers only when valid Data that meets the set conditions appears at the specified offset.

## ProfiBus Trigger

### ProfiBus Trigger Settings

Click ProfiBus Trigger in the toolbar and will show the dialog as the following.



ProfiBus Trigger Settings

Channel

CH A0

Option

Start bit is Low

Data Rate 9600 bps

Simple Trigger

□ Start of frame
□ End of frame
□ Chksum Error
□ SD Error
□ ED Error
□ LE Error
□ DA/SA Error
□ Parity Error

Parity Setting Even

Clause Trigger

State 1

Event 1

Start Delimiter (SD) SD1

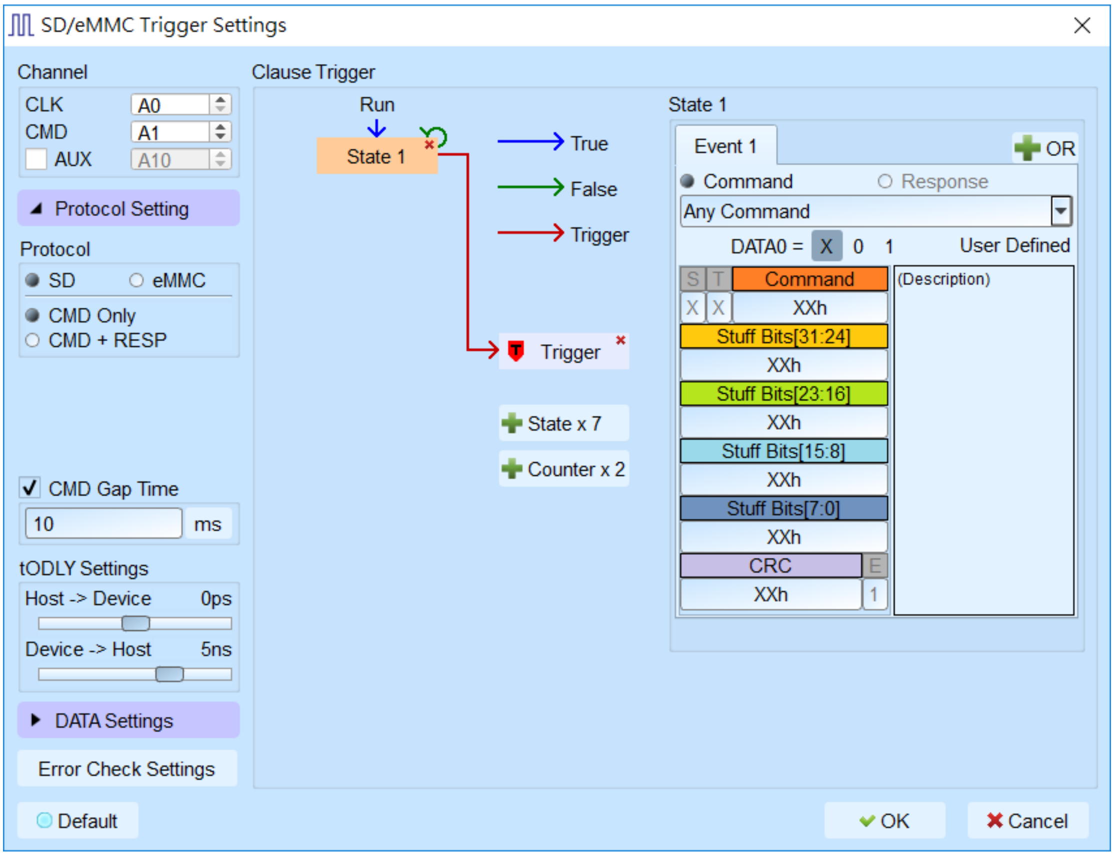
| | | | |
|---|---|---|---|
| DA | XXh | SA | XXh |
| FC | XXh | FCS | XXh |
| EDH | 16h | | |

1. **Channel:** Select channels.
2. **Simple Trigger:** Specific trigger function of ProfiBus.
3. **Clause Trigger:** Please reference Clause Trigger chapter.
4. **State:** Show the details of trigger condition in every state as left side; typing or selecting the trigger values in the SD packet and its fields, default value is XX means “don’t care”.

## SD/eMMC Trigger

### SD/eMMC Trigger Settings

Click **SD/eMMC** Trigger in the toolbar and will show the dialog as the following。



SD/eMMC Trigger Settings

Channel
CLK A0
CMD A1
□ AUX A10

Protocol Setting

Protocol
● SD ○ eMMC
● CMD Only
○ CMD + RESP

☑ CMD Gap Time
10 ms

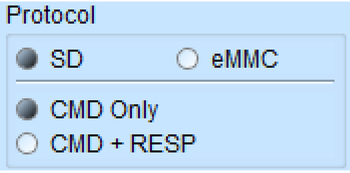
tODLY Settings
Host -> Device 0ps
Device -> Host 5ns

● Command ○ Response
Any Command
DATA0 = X 0 1 User Defined

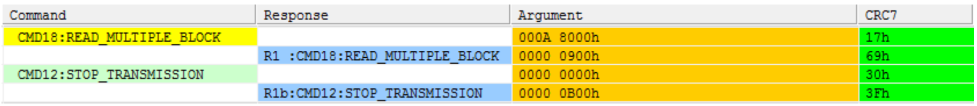
1. **Channel:** select CLK, CMD as the trigger channel

   AUX can be used for checking power state before running CRC error check, disabled by default.

2. **Protocol:** select SD or eMMC trigger



Protocol
● SD ○ eMMC
● CMD Only
○ CMD + RESP

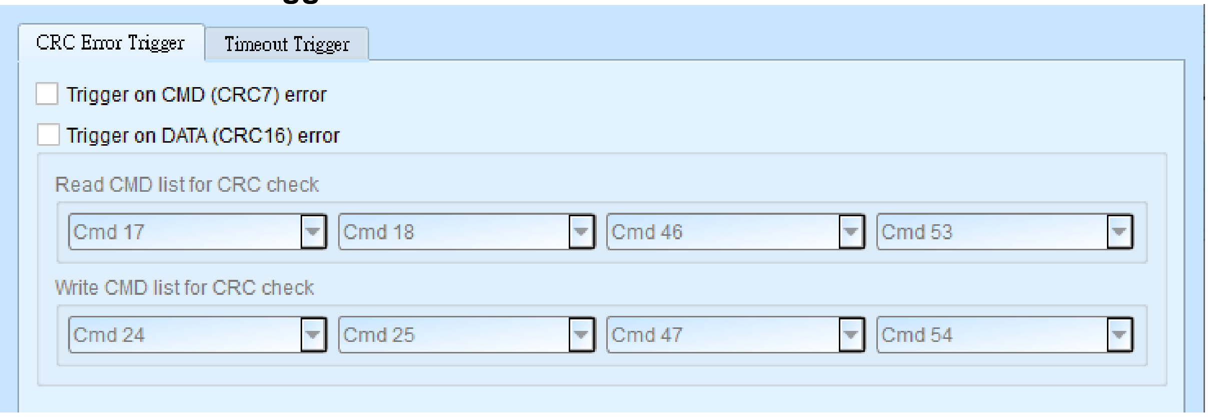


| Command | Response | Argument | CRC7 |
|---|---|---|---|
| CMD18:READ_MULTIPLE_BLOCK |  | 000A 8000h | 17h |
|  | R1 :CMD18:READ_MULTIPLE_BLOCK | 0000 0900h | 69h |
| CMD12:STOP_TRANSMISSION |  | 0000 0000h | 30h |
|  | R1b:CMD12:STOP_TRANSMISSION | 0000 0B00h | 3Fh |

3. **CMD Gap time:** delay trigger between the two CMDs.
4. **tODLY Settings:** set the phase delay time such that the device can latch valid command and response.

5. **Error Check Settings:** CRC7 error, CRC16 error, Timeout trigger are parallel structure with the Clause Trigger

### I. CRC Error Trigger:



□ Trigger on CMD (CRC7) error

□ Trigger on DATA (CRC16) error

Read CMD list for CRC check

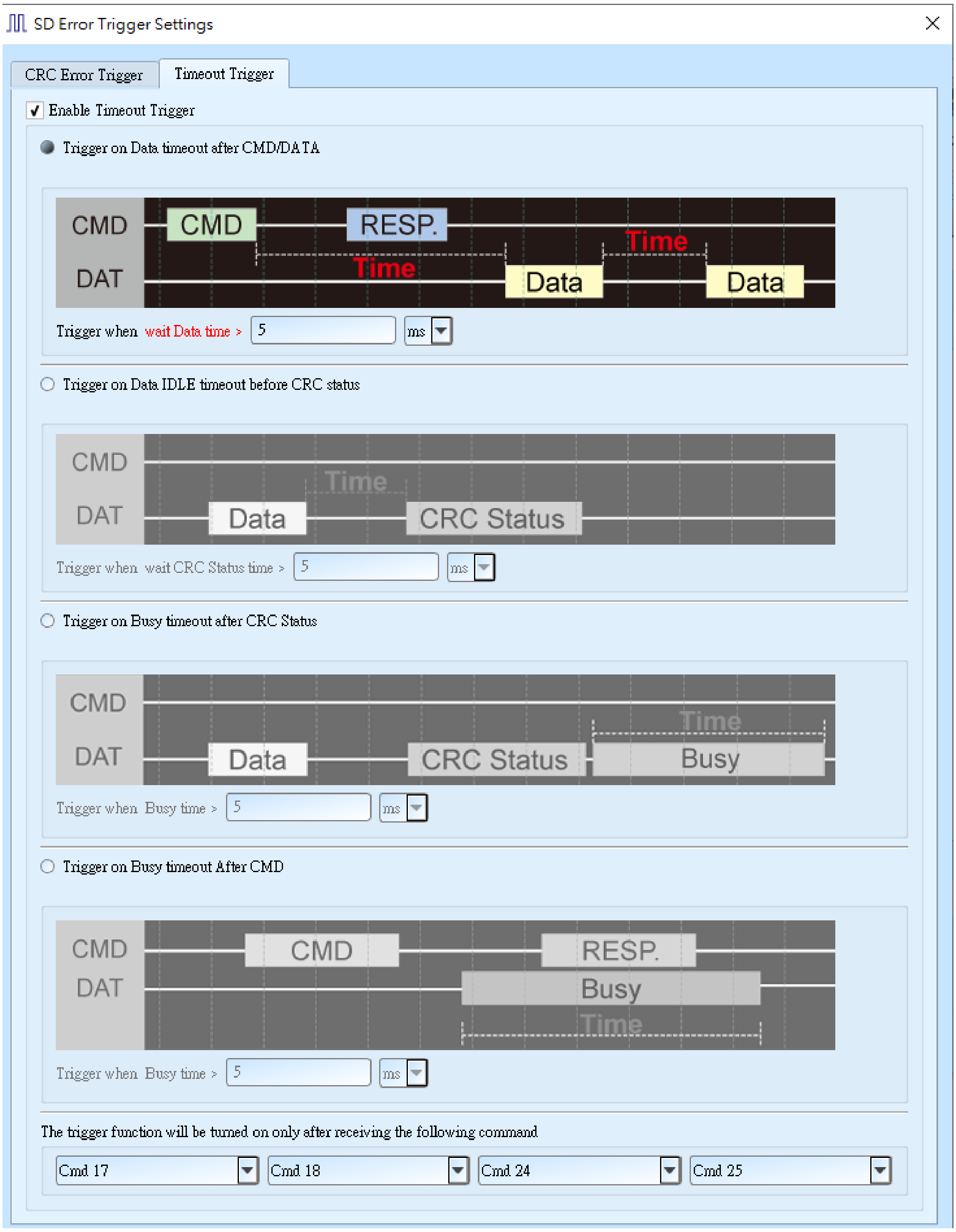
Cmd 17 | Cmd 18 | Cmd 46 | Cmd 53

Write CMD list for CRC check

Cmd 24 | Cmd 25 | Cmd 47 | Cmd 54

- **Trigger on CMD (CRC7) error:** Check the CRC7 error on Command line.
- **Trigger on DATA (CRC16) error:** Check the CRC16 error on DATA lines.

### II. Timeout Trigger: provide 4 statements as following



SD Error Trigger Settings

CRC Error Trigger | Timeout Trigger

☑ Enable Timeout Trigger

● Trigger on Data timeout after CMD/DATA

Trigger when wait Data time > 5 ms

○ Trigger on Data IDLE timeout before CRC status

Trigger when wait CRC Status time > 5 ms

○ Trigger on Busy timeout after CRC Status

Trigger when Busy time > 5 ms

○ Trigger on Busy timeout After CMD

Trigger when Busy time > 5 ms

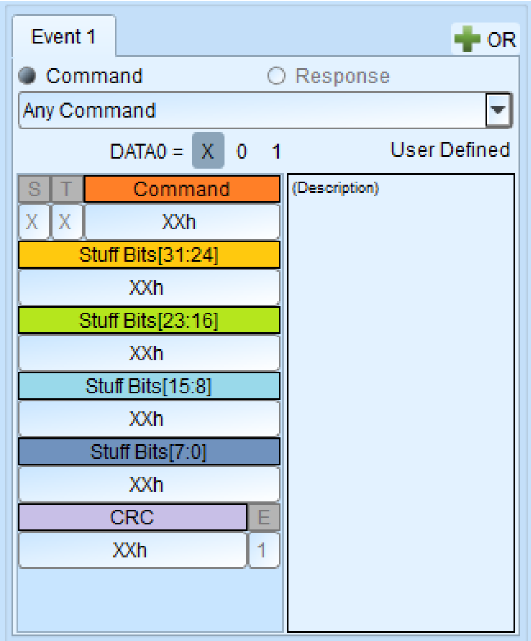
The trigger function will be turned on only after receiving the following command

Cmd 17 | Cmd 18 | Cmd 24 | Cmd 25

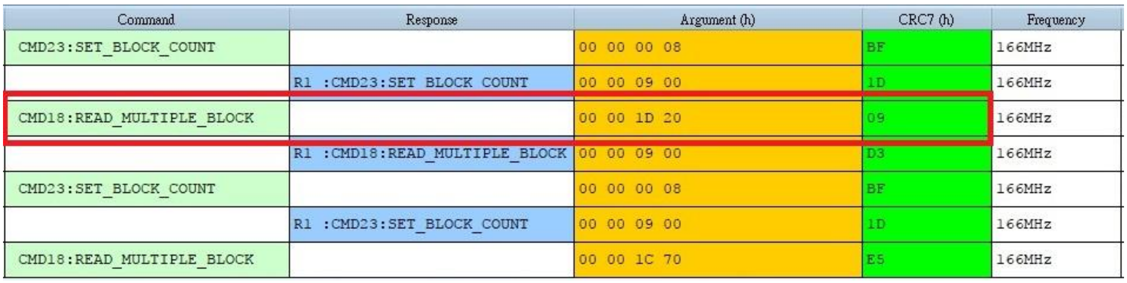
- ♦ **Trigger on Data timeout after CMD/DATA**
- ♦ **Trigger on Data IDLE timeout before CRC ststus**
- ♦ **Trigger on Busy timeout after CRC ststus**
- ♦ **Trigger on Busy timeout after CMD**

6. **Clause trigger settings:** Please reference Clause Trigger chapter

7. **Trigger settings**



SD/eMMC analysis result



| Command | Response | Argument (h) | CRC7 (h) | Frequency |
|---|---|---|---|---|
| CMD23:SET_BLOCK_COUNT | | 00 00 00 08 | BF | 166MHz |
| | R1 :CMD23:SET_BLOCK_COUNT | 00 00 09 00 | 1D | 166MHz |
| CMD18:READ_MULTIPLE_BLOCK | | 00 00 1D 20 | 09 | 166MHz |
| | R1 :CMD18:READ_MULTIPLE_BLOCK | 00 00 09 00 | D3 | 166MHz |
| CMD23:SET_BLOCK_COUNT | | 00 00 00 08 | BF | 166MHz |
| | R1 :CMD23:SET_BLOCK_COUNT | 00 00 09 00 | 1D | 166MHz |
| CMD18:READ_MULTIPLE_BLOCK | | 00 00 1C 70 | E5 | 166MHz |

Parameter：

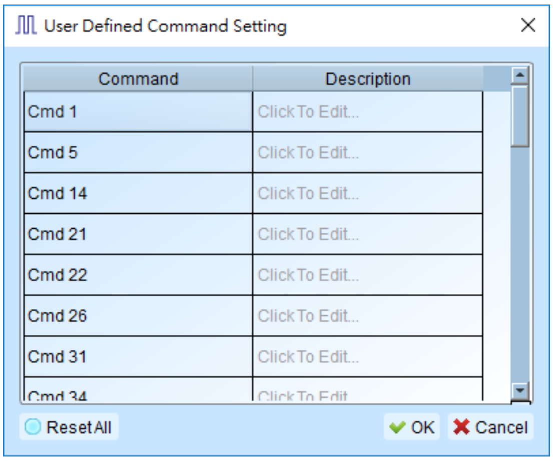
DATA0 = X 0 1 : use Data0 to judge the R1 and R1b.

Input trigger value, input ‘X’ means don’t care.

Append ‘h’ for HEX, ‘b’ for binary, no append for DEC.

When cursor on the input table, the information showed in the right side field.

User Defined add the command reserved:

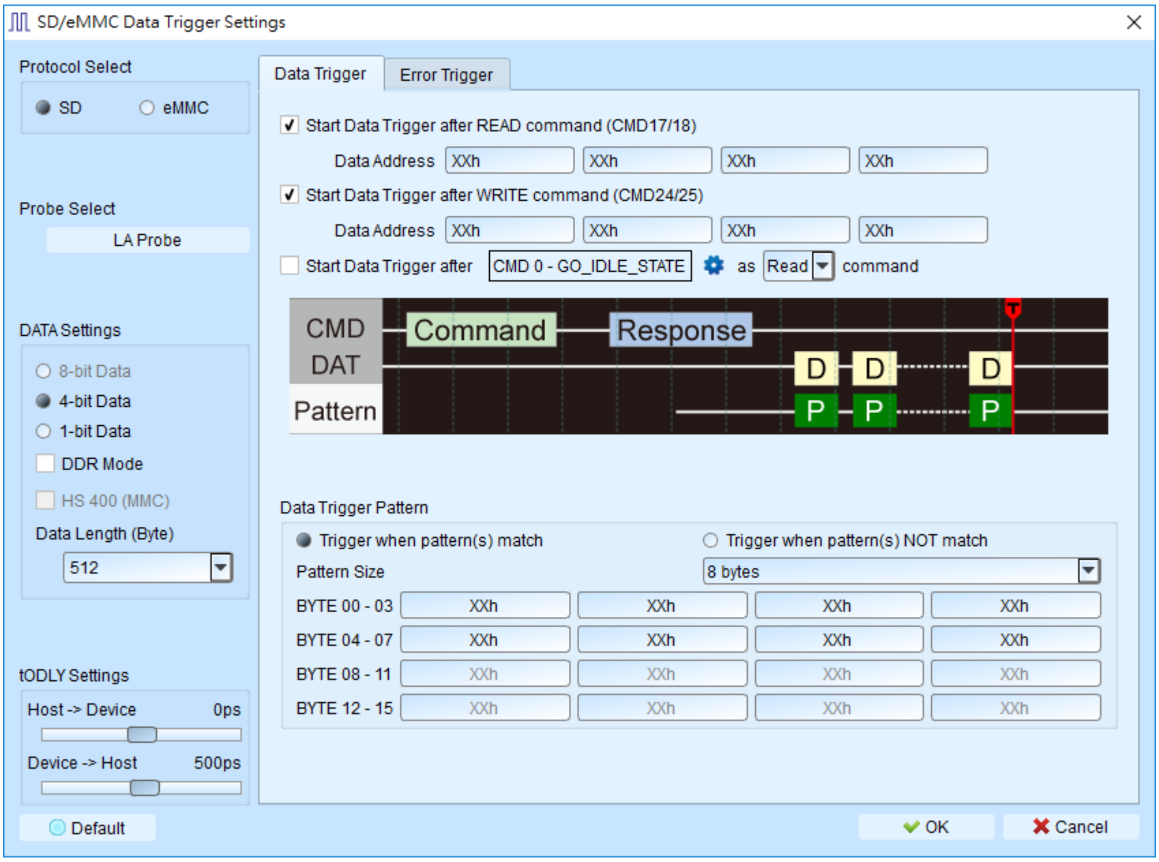


| Command | Description |
|---|---|
| Cmd 1 | Click To Edit... |
| Cmd 5 | Click To Edit... |
| Cmd 14 | Click To Edit... |
| Cmd 21 | Click To Edit... |
| Cmd 22 | Click To Edit... |
| Cmd 26 | Click To Edit... |
| Cmd 31 | Click To Edit... |
| Cmd 34 | Click To Edit... |

## SD/eMMC Data Trigger

### SD/eMMC Data Trigger Settings

Click **SD/eMMC Data** Trigger in the toolbar and will show the dialog as the following。



SD/eMMC Data Trigger Settings

Protocol Select

● SD ○ eMMC

Probe Select

LA Probe

DATA Settings

○ 8-bit Data
● 4-bit Data
○ 1-bit Data
□ DDR Mode
□ HS 400 (MMC)

Data Length (Byte): 512

tODLY Settings

Host -> Device 0ps

Device -> Host 500ps

Data Trigger | Error Trigger

☑ Start Data Trigger after READ command (CMD17/18)

Data Address XXh XXh XXh XXh

☑ Start Data Trigger after WRITE command (CMD24/25)

Data Address XXh XXh XXh XXh

□ Start Data Trigger after CMD 0 - GO_IDLE_STATE as Read command

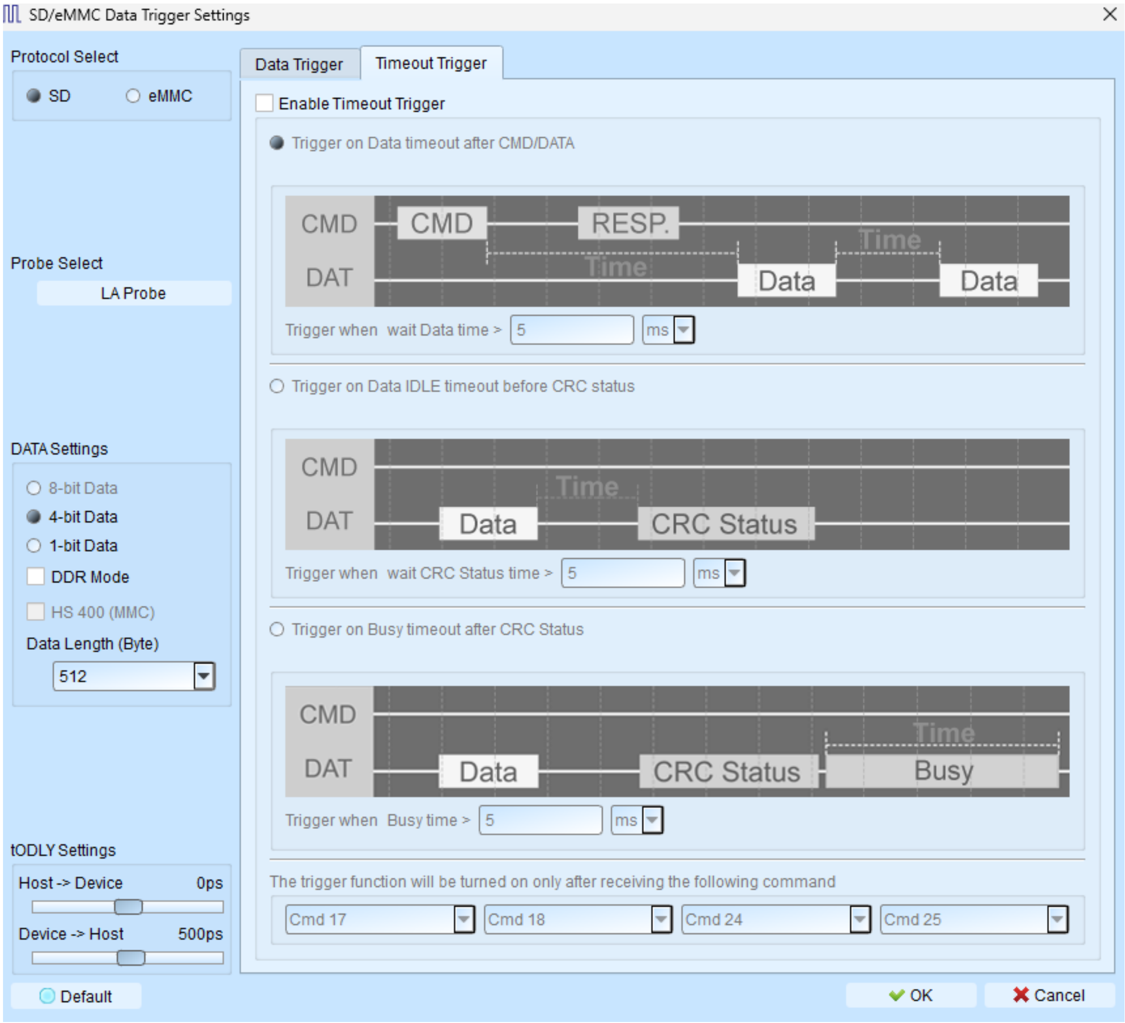
Data Trigger Pattern

● Trigger when pattern(s) match ○ Trigger when pattern(s) NOT match

Pattern Size: 8 bytes

| | | | | |
|---|---|---|---|---|
| BYTE 00 - 03 | XXh | XXh | XXh | XXh |
| BYTE 04 - 07 | XXh | XXh | XXh | XXh |
| BYTE 08 - 11 | XXh | XXh | XXh | XXh |
| BYTE 12 - 15 | XXh | XXh | XXh | XXh |

1. **Protocol:** select SD or eMMC trigger
2. **Probe Select:** select CLK, CMD and AUX as the trigger channel.
3. **Data Settings:** select current bus width, byte, SDR, DDR of the test object
4. **tODLY Settings:** set the phase delay time such that the device can latch valid command and response.
5. **Data Trigger:** set the patterns of data trigger.
6. **Timeout Trigger:** set the condition of idle time to trigger.



SD/eMMC Data Trigger Settings

Protocol Select

● SD ○ eMMC

Probe Select

LA Probe

DATA Settings

○ 8-bit Data
● 4-bit Data
○ 1-bit Data
□ DDR Mode
□ HS 400 (MMC)

Data Length (Byte)

512

tODLY Settings

Host -> Device 0ps

Device -> Host 500ps

Default

Data Trigger | Timeout Trigger

□ Enable Timeout Trigger

● Trigger on Data timeout after CMD/DATA

Trigger when wait Data time > 5 ms

○ Trigger on Data IDLE timeout before CRC status

Trigger when wait CRC Status time > 5 ms

○ Trigger on Busy timeout after CRC Status

Trigger when Busy time > 5 ms

The trigger function will be turned on only after receiving the following command

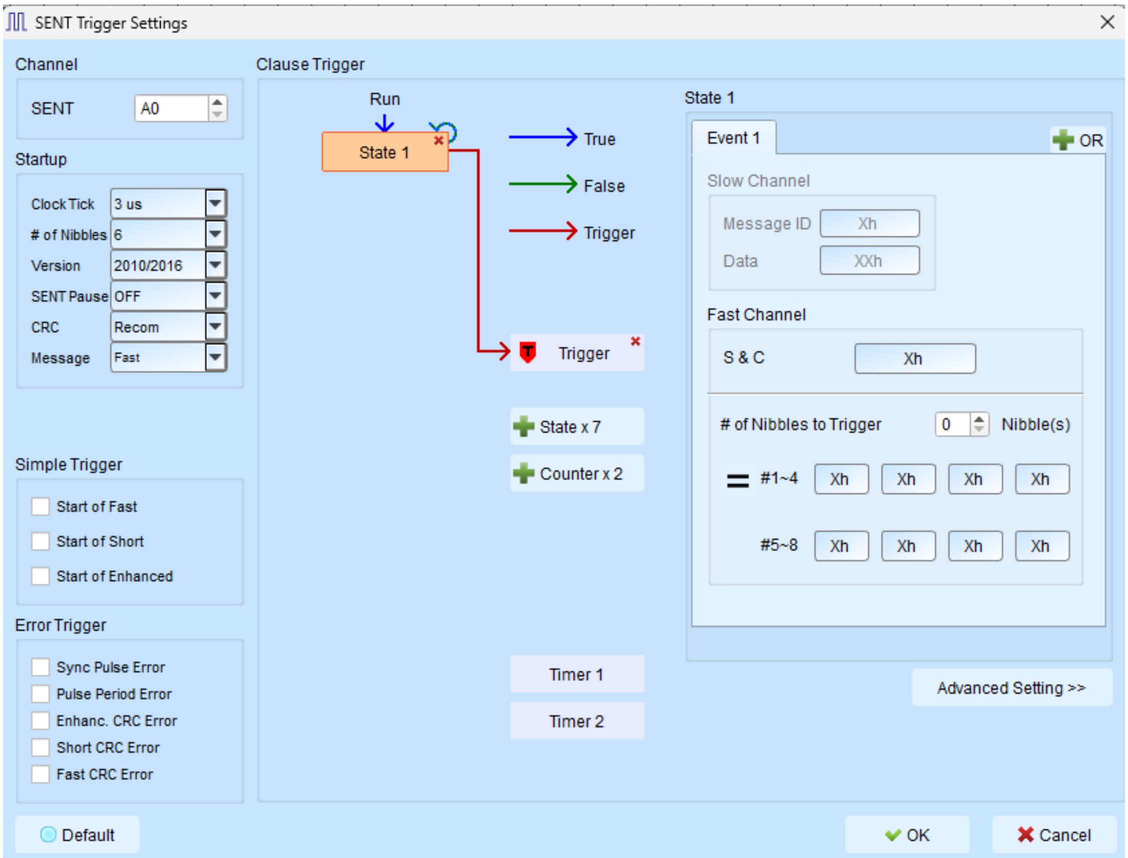
Cmd 17 | Cmd 18 | Cmd 24 | Cmd 25

✔ OK ✖ Cancel

## SENT Trigger

### SENT Trigger Settings

Click SENT Trigger in the toolbar and will show the dialog as the following.



SENT Trigger Settings

Channel
SENT: A0

Startup
Clock Tick: 3 us
# of Nibbles: 6
Version: 2010/2016
SENT Pause: OFF
CRC: Recom
Message: Fast

Simple Trigger
□ Start of Fast
□ Start of Short
□ Start of Enhanced

Error Trigger
□ Sync Pulse Error
□ Pulse Period Error
□ Enhanc. CRC Error
□ Short CRC Error
□ Fast CRC Error

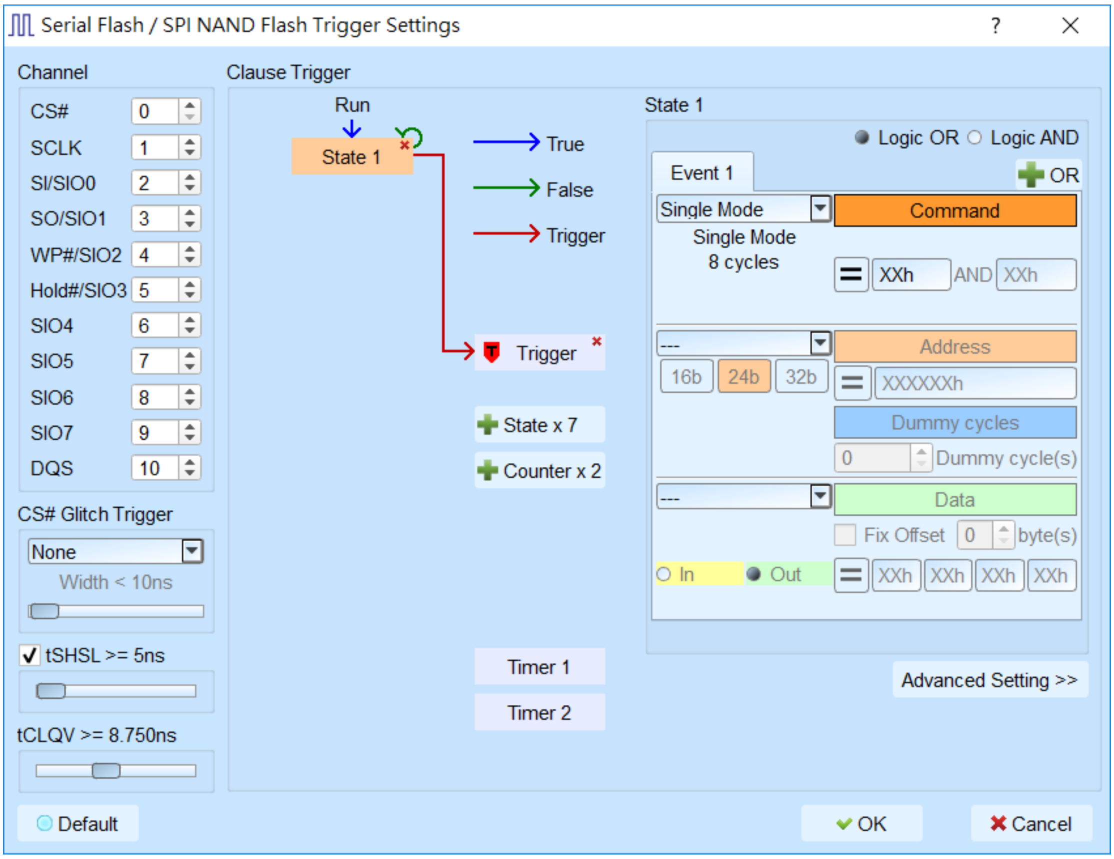
State 1
Event 1
Slow Channel
Message ID: Xh
Data: XXh
Fast Channel
S & C: Xh
# of Nibbles to Trigger: 0 Nibble(s)
#1~4: Xh Xh Xh Xh
#5~8: Xh Xh Xh Xh

1. **Channel:** Configure the SENT channel.
2. **Startup:** Set the Startup conditions.
3. **Simple Trigger:** Configure SENT-specific frame trigger. Enabled when selected.
4. **Error Trigger:** Configure SENT-specific error trigger. Enabled when selected.
5. **Clause Trigger:** Please reference Clause Trigger chapter.
6. **State:** Displays the detailed trigger conditions within each level of the trigger process on the left. Users can configure the Slow Channel Message ID or Data (available only when the Message setting in Startup is set to Short or Enhanced) or specify details in the Fast Channel (available only when the Message setting in Startup is set to Fast).

## Serial Flash / SPI NAND Trigger

### Serial Flash / SPI NAND Flash Trigger Settings

Click Serial Flash Trigger in the toolbar and will show the dialog as the following.



Serial Flash / SPI NAND Flash Trigger Settings

Channel

| | |
|---|---|
| CS# | 0 |
| SCLK | 1 |
| SI/SIO0 | 2 |
| SO/SIO1 | 3 |
| WP#/SIO2 | 4 |
| Hold#/SIO3 | 5 |
| SIO4 | 6 |
| SIO5 | 7 |
| SIO6 | 8 |
| SIO7 | 9 |
| DQS | 10 |

CS# Glitch Trigger: None, Width < 10ns

☑ tSHSL >= 5ns

tCLQV >= 8.750ns

State 1: ● Logic OR ○ Logic AND

Event 1: Single Mode, Single Mode 8 cycles, Command = XXh AND XXh

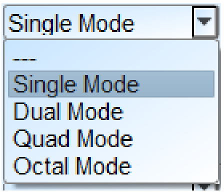
Address: ---, 16b / 24b / 32b, = XXXXXXh

Dummy cycles: 0 Dummy cycle(s)

Data: ---, □ Fix Offset 0 byte(s), ○ In ● Out, = XXh XXh XXh XXh

1. **Channel:** Select channels.
2. **CS# Glitch Trigger:** Trigger the glitch on CS# channel.
3. **tSHSL / tCLQV:** Set tSHSL / tCLQV.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Show the details of trigger condition in every state as left side; typing or selecting the trigger values in the Command, Address and Data fields, default value is XX means “don’t care”.

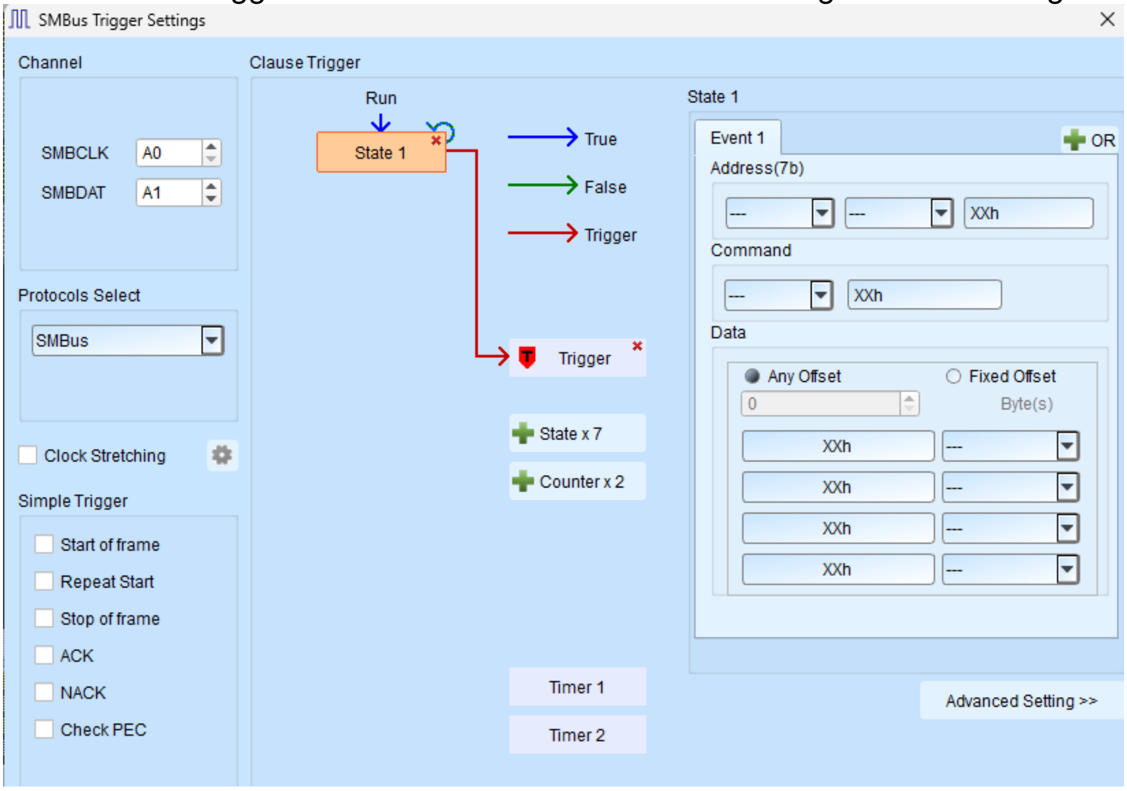
   Provided the Single / Dual / Quad / Octal mode.



## SMBus Trigger

### SMBus Trigger Settings

Click SMBus Trigger in the toolbar and will show the dialog as the following.



SMBus Trigger Settings

Channel: SMBCLK A0, SMBDAT A1

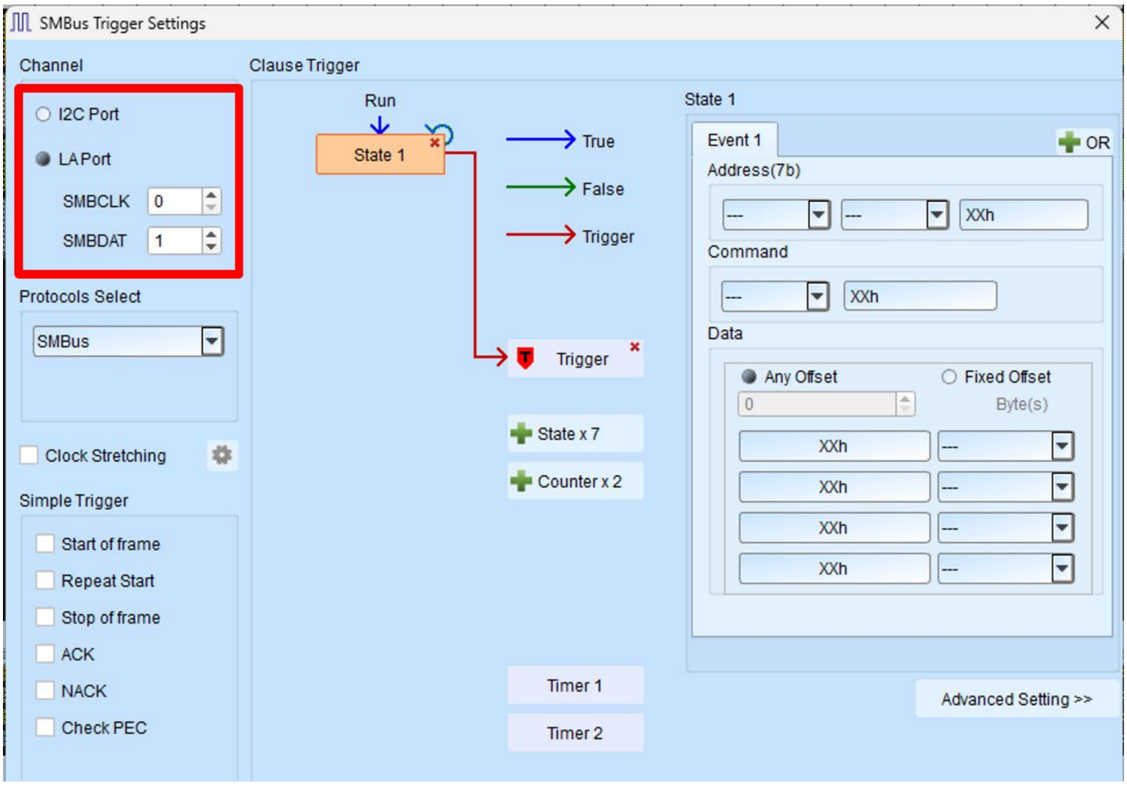
Protocols Select: SMBus

☐ Clock Stretching

Simple Trigger: ☐ Start of frame ☐ Repeat Start ☐ Stop of frame ☐ ACK ☐ NACK ☐ Check PEC

State 1 – Event 1: Address(7b) --- --- XXh; Command --- XXh; Data ● Any Offset ○ Fixed Offset, 0 Byte(s); XXh --- ×4

Travel Bus series



SMBus Trigger Settings

Channel: ○ I2C Port ● LA Port, SMBCLK 0, SMBDAT 1

Protocols Select: SMBus

☐ Clock Stretching

Simple Trigger: ☐ Start of frame ☐ Repeat Start ☐ Stop of frame ☐ ACK ☐ NACK ☐ Check PEC

State 1 – Event 1: Address(7b) --- --- XXh; Command --- XXh; Data ● Any Offset ○ Fixed Offset, 0 Byte(s); XXh --- ×4

***Since TravelBus has specially designed channels for I2C, the SMBus settings in the TravelBus software will include additional channel source options.***

1. **Channel:** Configure the I2C interface (supported only on TravelBus B series models) or set to LA channels.
2. **Protocols Select:** Set the trigger for SMBus, SBS, or SPD protocols.
3. **Simple Trigger:** Configure SMBus-specific frame trigger.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Displays the detailed trigger conditions within each level of the trigger process on the left. Users can enter specific trigger values for Address, Command, and Data, or use 'X' as a wildcard for any value. The Command field will display different command formats based on the Protocols Select setting.

   **I.** The Data field allows up to 4 Bytes to be set. Unused fields should be filled with XXh to indicate any value.

   **II.** The input field can contain the required trigger Data or 'X' as a wildcard.

   - **Hexadecimal** values should end with **'h'**.
   - **Binary** values should end with **'b'**.
   - **Decimal** values do not require a suffix.

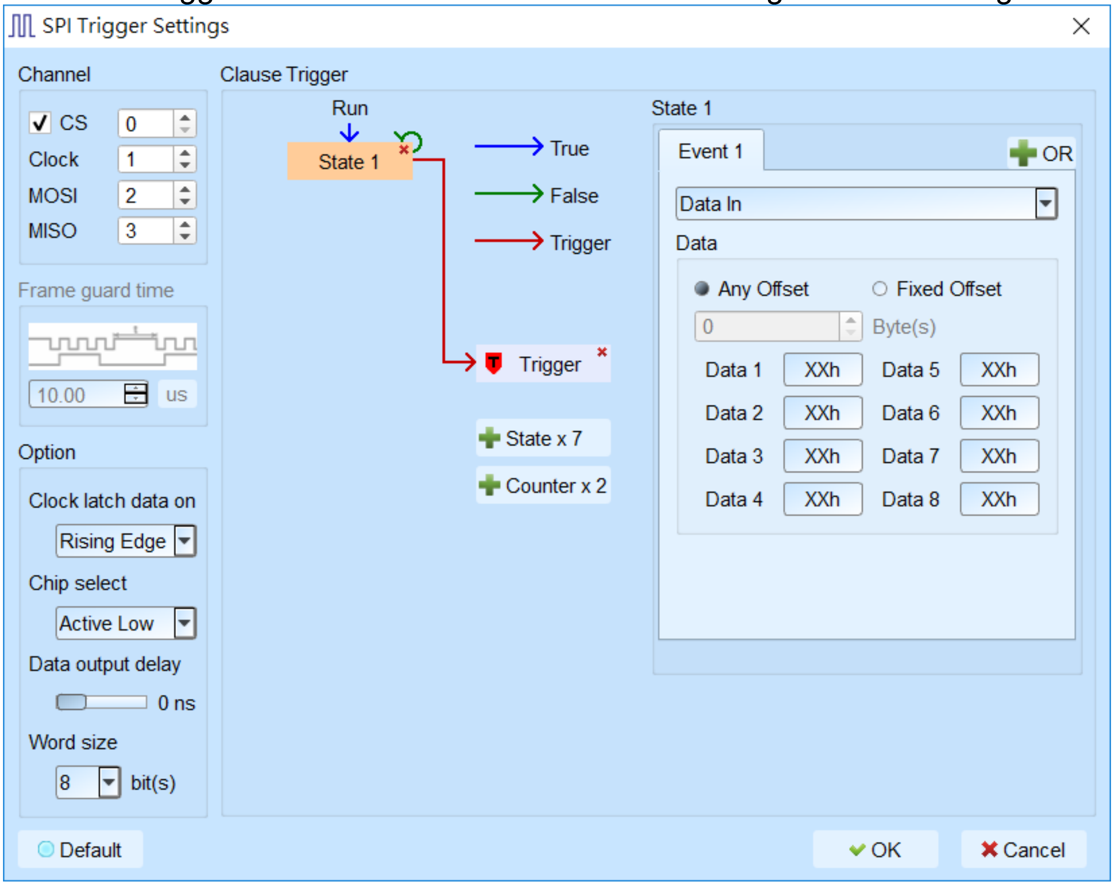
   **III. Data Offset Triggering:**

   - **Any Offset:** Triggers when valid Data that meets the set conditions appears anywhere in the Data field, regardless of offset.
   - **Fixed Offset:** Triggers only when valid Data that meets the set conditions appears at the specified offset.

## SPI Trigger

### SPI Trigger Settings

Click SPI Trigger in the toolbar and will show the dialog as the following.



Channel

☑ CS 0

Clock 1

MOSI 2

MISO 3

Frame guard time 10.00 us

Option

Clock latch data on Rising Edge

Chip select Active Low

Data output delay 0 ns

Word size 8 bit(s)

State 1

Event 1

Data In

Data

● Any Offset ○ Fixed Offset

0 Byte(s)

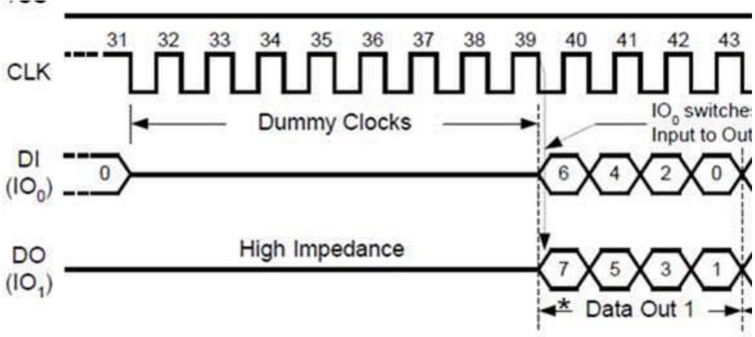
| | | | |
|---|---|---|---|
| Data 1 | XXh | Data 5 | XXh |
| Data 2 | XXh | Data 6 | XXh |
| Data 3 | XXh | Data 7 | XXh |
| Data 4 | XXh | Data 8 | XXh |

1. **Channel:** Select the channel. Depending on the mode, two to four channels can be used.
2. **Frame guard time:** When CS is not selected, this value can be set to define the time threshold for recognizing the next Frame.
3. **Option:** The Data Output Delay setting specifies how long the Latch Data should be delayed after the transition edge. The default value is 0, with a maximum delay of 75 ns.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Displays the detailed trigger conditions within each level of the trigger process.

   I. Data In: Triggers based only on the Data In channel.

   II. Data Out (Ref. Output Delay): Triggers based only on the Data Out channel, applying the Data Output Delay parameter.

III. Dual Data: Treats Data In/Out as a 2-bit dual-channel mode. For example, if the Word Size is set to 8, only 4 clock cycles are required for transmission. The 1st bit of the Data Out pin represents the MSB.



IV. Dual Data(Ref. Output Delay): Functions the same as Dual Data, but applies the Data Output Delay parameter.

Data Field Input Rules:

I. The Data field allows up to 8 Bytes. Unused fields should be filled with XXh to indicate any value.

II. The input field can contain the required trigger Data or 'X' as a wildcard.

- Hexadecimal values should end with 'h'.
- Binary values should end with 'b'.
- Decimal values do not require a suffix.

III. Data Offset Triggering

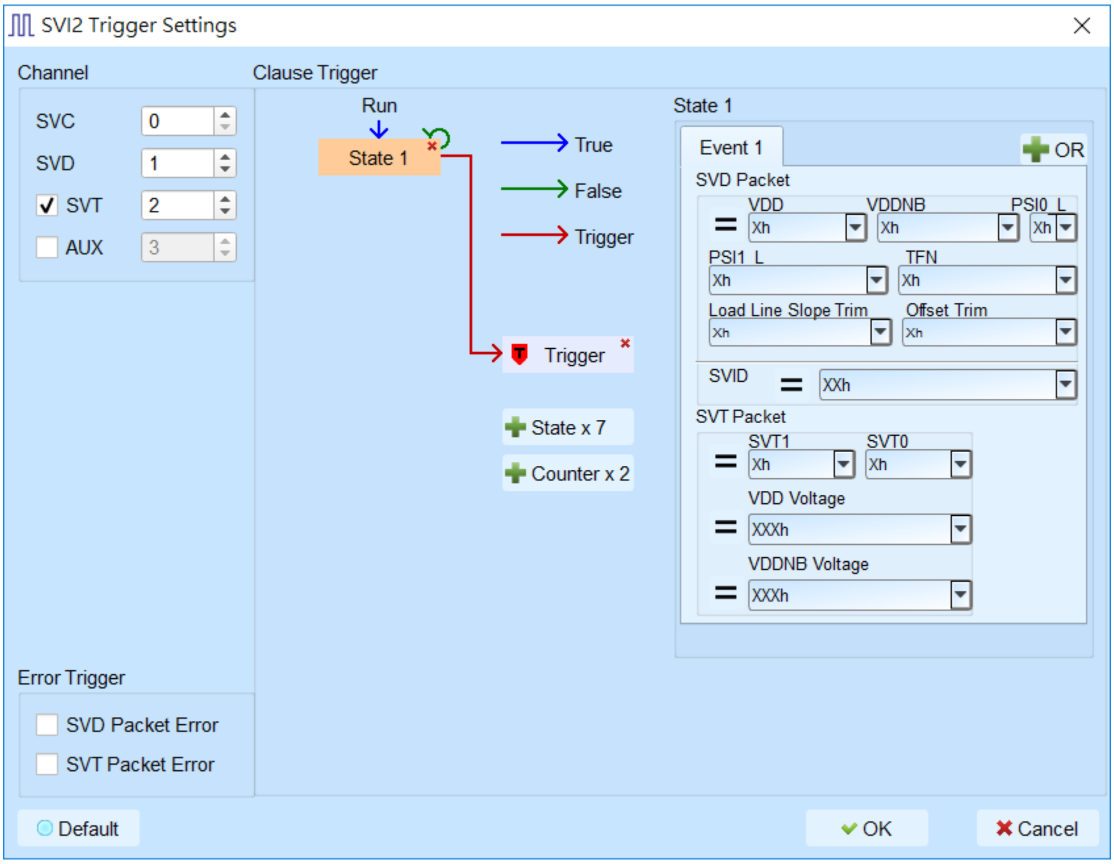
Any Offset: Triggers when valid Data that meets the set conditions appears anywhere in the Data field, regardless of offset.

Fixed Offset: Triggers only when valid Data that meets the set conditions appears at the specified offset.

## SVI2 Trigger

### SVI2 Trigger Settings

Click SVI2 Trigger in the toolbar and will show the dialog as the following.



SVI2 Trigger Settings

Channel

SVC: 0

SVD: 1

☑ SVT: 2

□ AUX: 3

Error Trigger

□ SVD Packet Error

□ SVT Packet Error

State 1 — Event 1

SVD Packet: VDD = Xh, VDDNB = Xh, PSI0_L = Xh, PSI1_L = Xh, TFN = Xh, Load Line Slope Trim = Xh, Offset Trim = Xh

SVID = XXh

SVT Packet: SVT1 = Xh, SVT0 = Xh

VDD Voltage = XXXh

VDDNB Voltage = XXXh

1. **Channel:** Select channels.

2. **Error Trigger:** Trigger specific error of SVI2.

3. **Clause Trigger:** Please reference Clause Trigger chapter.

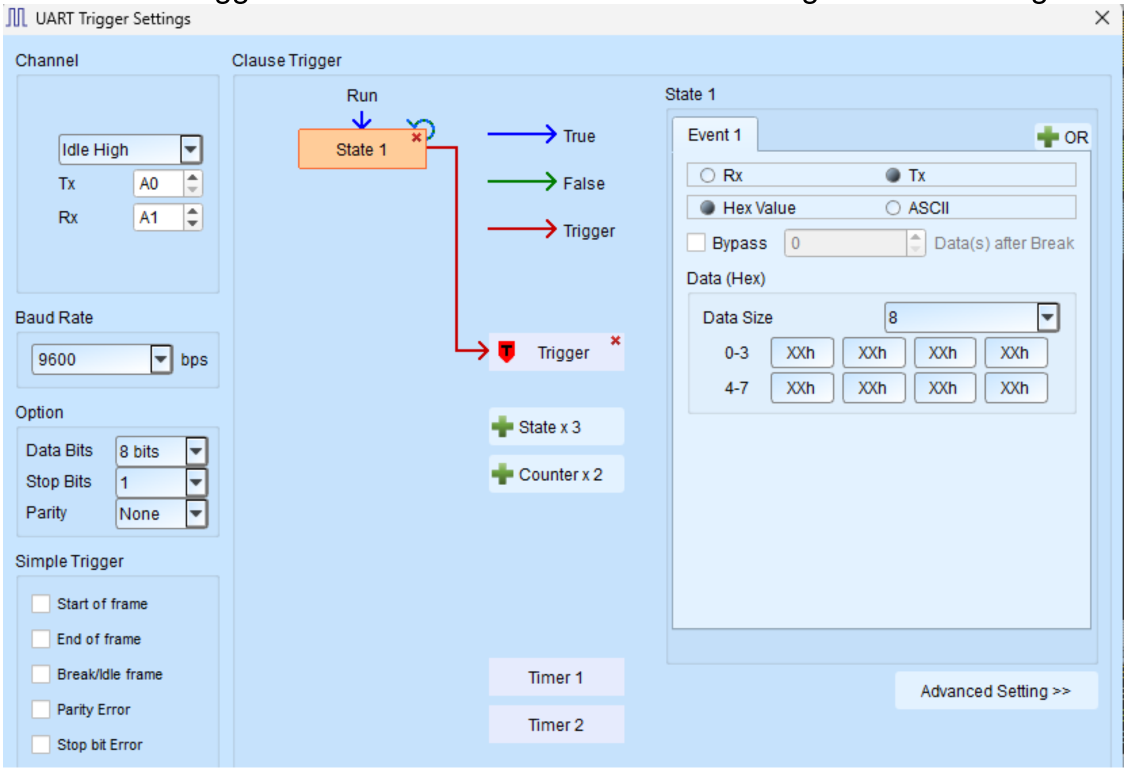
4. **State:** Show the details of trigger condition in every state as left side; typing or selecting the trigger values in the SVD and SVT packet, default value is XX means “don’t care”.

   When check the AUX, the trigger function will refer to the state of AUX (High/Low).

## UART Trigger

### UART Trigger Settings

Click UART Trigger in the toolbar and will show the dialog as the following.



UART Trigger Settings

Channel: Idle High; Tx: A0; Rx: A1

Baud Rate: 9600 bps

Option: Data Bits: 8 bits; Stop Bits: 1; Parity: None

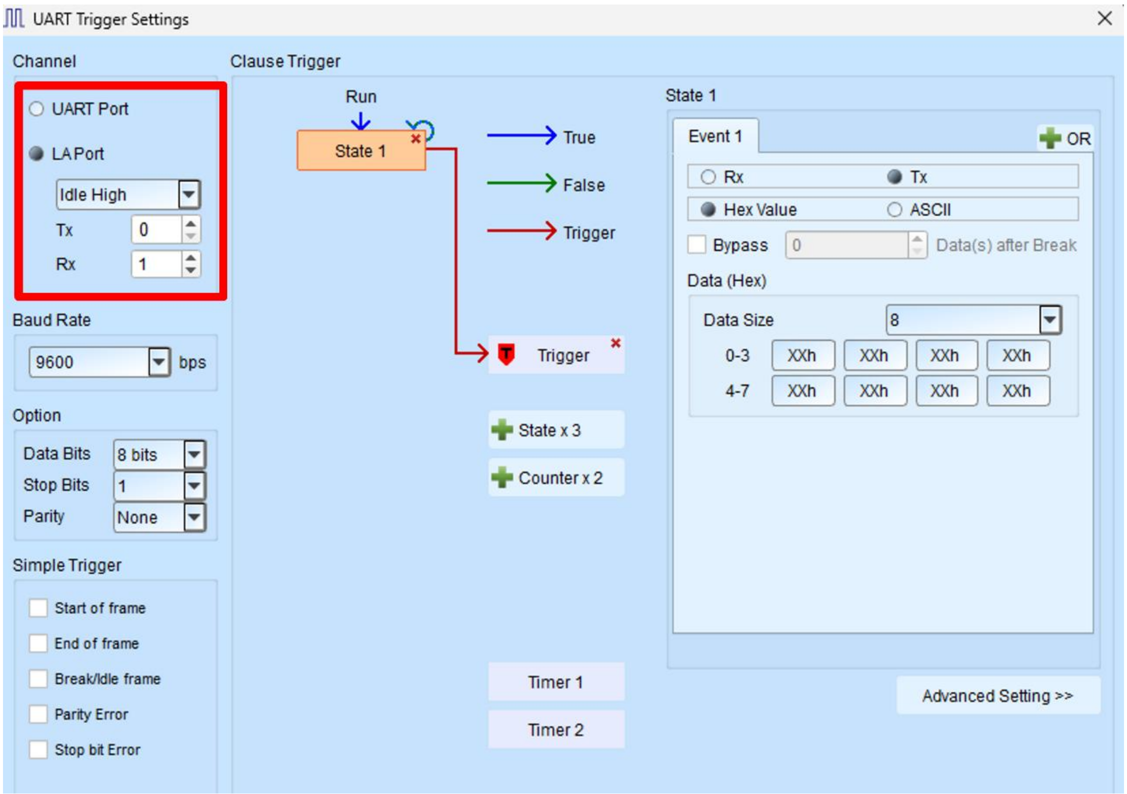
Simple Trigger: □ Start of frame; □ End of frame; □ Break/Idle frame; □ Parity Error; □ Stop bit Error

State 1 – Event 1: ○ Rx ● Tx; ● Hex Value ○ ASCII; □ Bypass 0 Data(s) after Break

Data (Hex): Data Size 8

| | | | | |
|---|---|---|---|---|
| 0-3 | XXh | XXh | XXh | XXh |
| 4-7 | XXh | XXh | XXh | XXh |

Travel Bus series



UART Trigger Settings

Channel: ○ UART Port ● LA Port; Idle High; Tx: 0; Rx: 1

Baud Rate: 9600 bps

Option: Data Bits: 8 bits; Stop Bits: 1; Parity: None

Simple Trigger: □ Start of frame; □ End of frame; □ Break/Idle frame; □ Parity Error; □ Stop bit Error

State 1 – Event 1: ○ Rx ● Tx; ● Hex Value ○ ASCII; □ Bypass 0 Data(s) after Break

Data (Hex): Data Size 8

| | | | | |
|---|---|---|---|---|
| 0-3 | XXh | XXh | XXh | XXh |
| 4-7 | XXh | XXh | XXh | XXh |

***Since TravelBus has specially designed channels for UART, the UART***

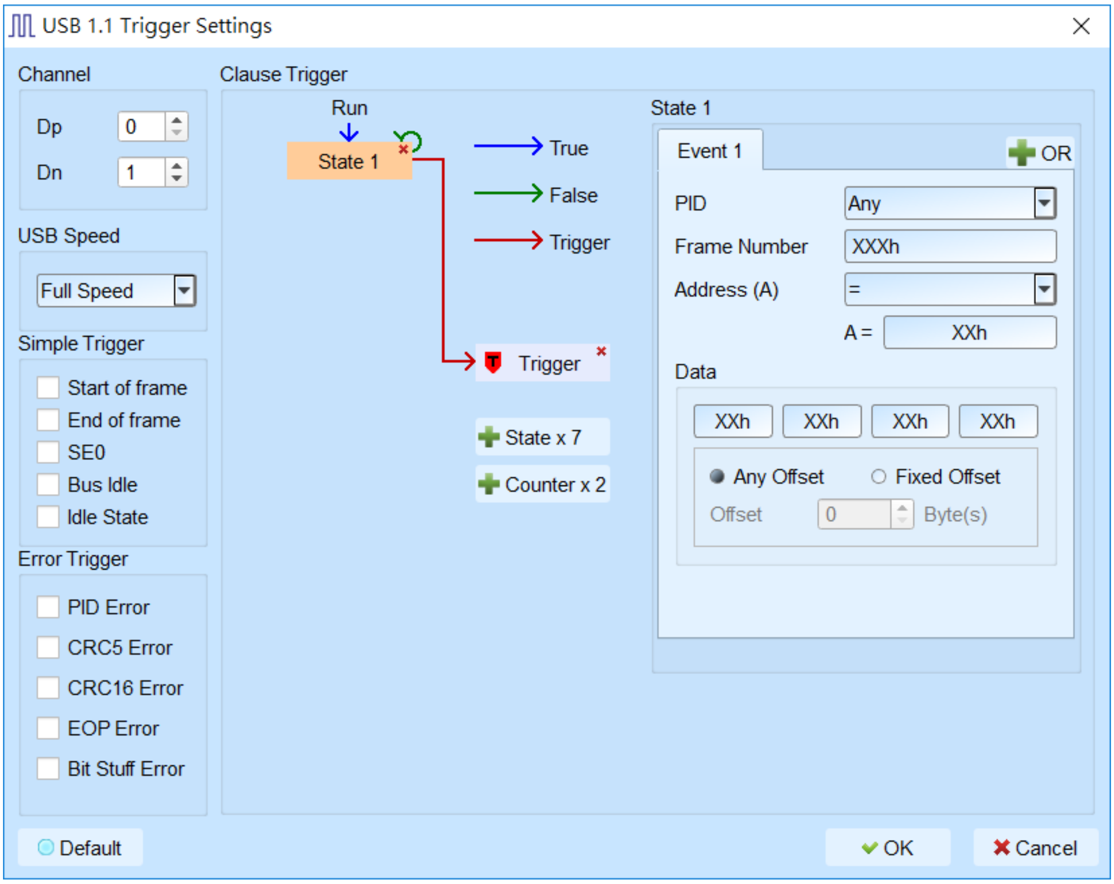
***settings in the TravelBus software will include additional channel source options.***

1. **Channel:** Configure the UART interface.
2. **Baud Rate:** Set the UART Baud Rate.
3. **Simple Trigger:** Configure UART-specific frame trigger.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Displays the detailed trigger conditions within each level of the trigger process. Users can enter specific trigger values in the Data field or use 'X' as a wildcard for any value. The Data trigger field supports up to 16 Bytes; to specify a trigger data position, Bypass must be selected, and an offset value must be entered.

## USB 1.1 Trigger

### USB 1.1 Trigger Settings

Click USB 1.1 Trigger in the toolbar and will show the dialog as the following.



USB 1.1 Trigger Settings

Channel
Dp: 0
Dn: 1

USB Speed: Full Speed

Simple Trigger
□ Start of frame
□ End of frame
□ SE0
□ Bus Idle
□ Idle State

Error Trigger
□ PID Error
□ CRC5 Error
□ CRC16 Error
□ EOP Error
□ Bit Stuff Error

State 1 – Event 1
PID: Any
Frame Number: XXXh
Address (A): =
A = XXh
Data: XXh XXh XXh XXh
● Any Offset ○ Fixed Offset
Offset: 0 Byte(s)

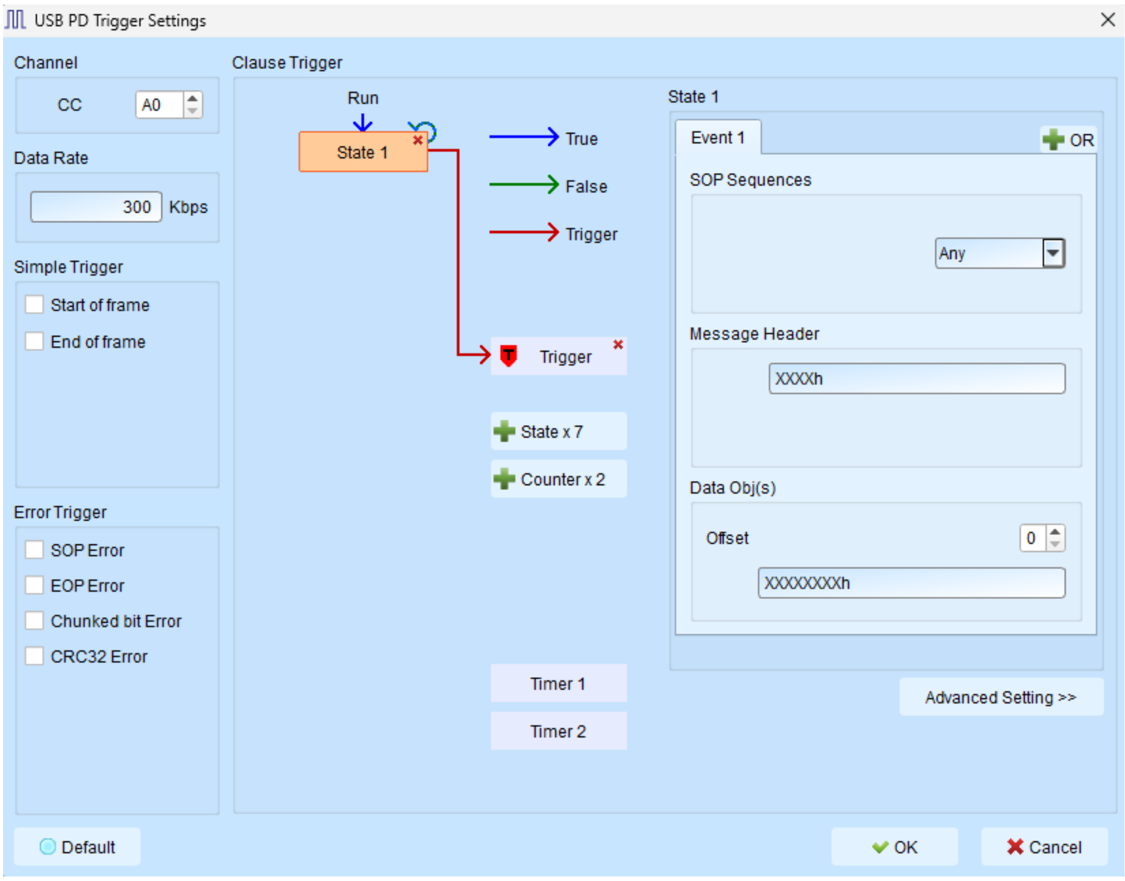
1. **Channel:** Select channels.
2. **Simple Trigger:** Specific trigger function of USB 1.1.
3. **Error Trigger:** Trigger specific error of USB 1.1.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Show the details of trigger condition in every state as left side; typing or selecting the trigger values in the PID, Frame Number, Address and Data Field, default value is XX means “don’t care”.

   Data field provided 4 bytes data to trigger, checking the Fixed Offset and selecting the offset value when want to trigger at the specific position.

## USB PD Trigger

### USB PD 3.0 Trigger Settings

Click USB PD 3.0 Trigger in the toolbar and will show the dialog as the following.



1. **Channel:** Select channels.
2. **Simple Trigger:** Specific trigger function of USB PD 3.0.
3. **Error Trigger:** Trigger specific error of USB PD 3.0.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Show the details of trigger condition in every state as left side; typing or selecting the trigger values in the SOP Sequences, Message Header and Data Obj(s), default value is XX means “don’t care”. Provide 0 ~ 7 offset value for Data Obj(s) field.