

Table of Contents

Chapter 1	Bus Decode	1
Basic operation		1
Add a Bus Decode		1
Advance channel setting		2
Specially Bus Decode:		4
Bus Decode Settings Introduction		7
1-Wire		7
3-Wire		10
7-Segment		12
A/D Converter		15
Accelerometer		18
AD-Mux Flash		20
Advanced Platform Management Link (APML)		22
BiSS-C		24
BSD		26
BT1120		28
CAN 2.0B/ CAN FD		30
Closed Caption		32
Codec SSI		34
DALI		35
DMX512		37
Display Port Auxiliary Channel (DP Aux Ch)		39
Enhanced Serial Peripheral Interface (eSPI)		42
FlexRay		45
HD Audio		48

HDMI-CEC-----	50
HDMI-DDC (EDID)-----	52
HDLC-----	54
HDQ-----	56
HID Over I ² C-----	58
I ² C-----	60
I ² C User Defined-----	62
I ³ C-----	67
I ² S-----	71
I80-----	73
IDE-----	75
IrDA-----	79
ITU656 (CCIR656)-----	81
JTAG-----	83
LCD1602-----	88
LED_Ctrl-----	90
LIN-----	92
Line Decoding-----	94
Line Encoding-----	98
Low Pin Count (LPC)-----	104
Line Printer Terminal Port (LPT)-----	106
M-Bus-----	108
Math-----	110
Mini / Micro-LED-----	112
Mobile Display Digital Interface (MDDI)-----	114
MDIO-----	116

MHL-CBUS	118
MII/RMII	120
Microwire	123
MIPI DSI	125
MIPI RFFE	127
MIPI SPMI	128
MMC	130
ModBus	134
NAND Flash	136
NEC IR	149
PECI	151
PMBus	153
ProfiBus	155
PS/2	157
PWM	159
QI	163
QSPI	165
RC-5	167
RC-6	169
RGB Interface	171
S/PDIF	173
SDIO	176
SDR SDRAM	179
Serial Flash	182
Serial IRQ	188
Serial General Purpose Input Output (SGPIO)	192

Smart Card (ISO7816)-----	194
System Management Bus (SMBus) -----	196
Serial Microprocessor Interface (SMI)-----	199
SoundWire -----	201
Serial Peripheral Interface (SPI) -----	205
Serial Peripheral Interface NAND (SPI NAND)-----	212
SSI-----	214
ST7669 -----	216
Serial VID Interface 2.0 (SVI2)-----	218
Serial VID (SVID) -----	220
Serial Wire Debug (SWD) -----	222
SWIM-----	225
SWP-----	227
Universal Asynchronous Receiver/Transmitter (UART) -----	229
ULPI -----	232
UNI/O -----	234
USB1.1 -----	236
USB4/TBT3 SB-----	238
USB PD 2.0/3.0-----	240
Wiegand-----	243
Chapter 2 Bus Trigger-----	245
Bus Trigger -----	246
Parallel Clause Trigger-----	247
Clause Trigger-----	252
BiSS-C Trigger -----	256
CAN Trigger -----	257


DALI Trigger	259
eSPI Trigger	260
HIDoverI2C Trigger	262
I ² C Trigger	264
I ² S Trigger	266
LIN Trigger	268
MDIO Trigger	271
ModBus Logic Trigger	273
NAND Flash Trigger	275
ProfiBus Trigger	280
SD/eMMC Trigger	281
SD/eMMC Data Trigger	286
Serial Flash / SPI NAND Trigger	288
SMBus Trigger	289
SPI Trigger	290
SVI2 Trigger	292
SVID Trigger	293
UART Trigger	295
USB 1.1 Trigger	296
USB PD 3.0 Trigger	297

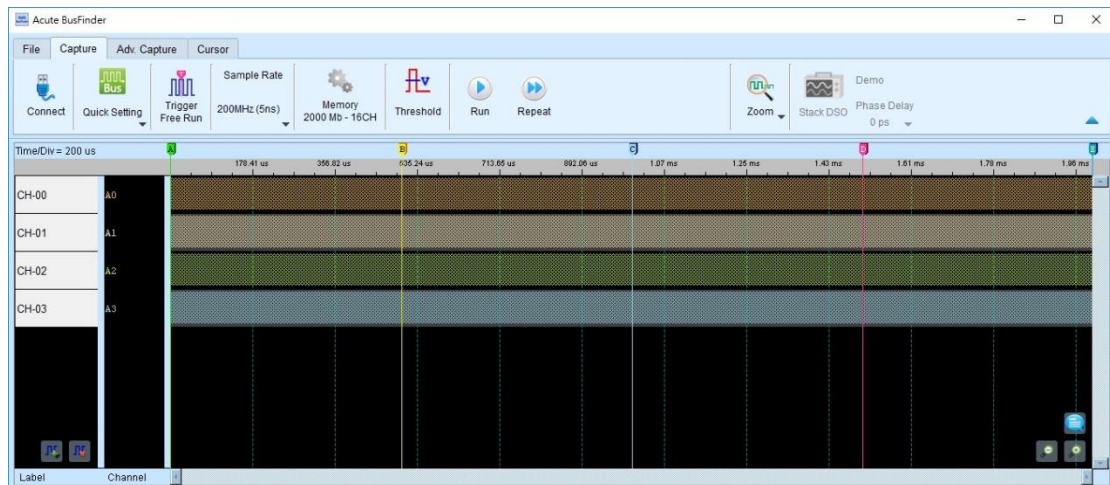
Chapter 1 Bus Decode

Basic operation

Add a Bus Decode

Method 1:

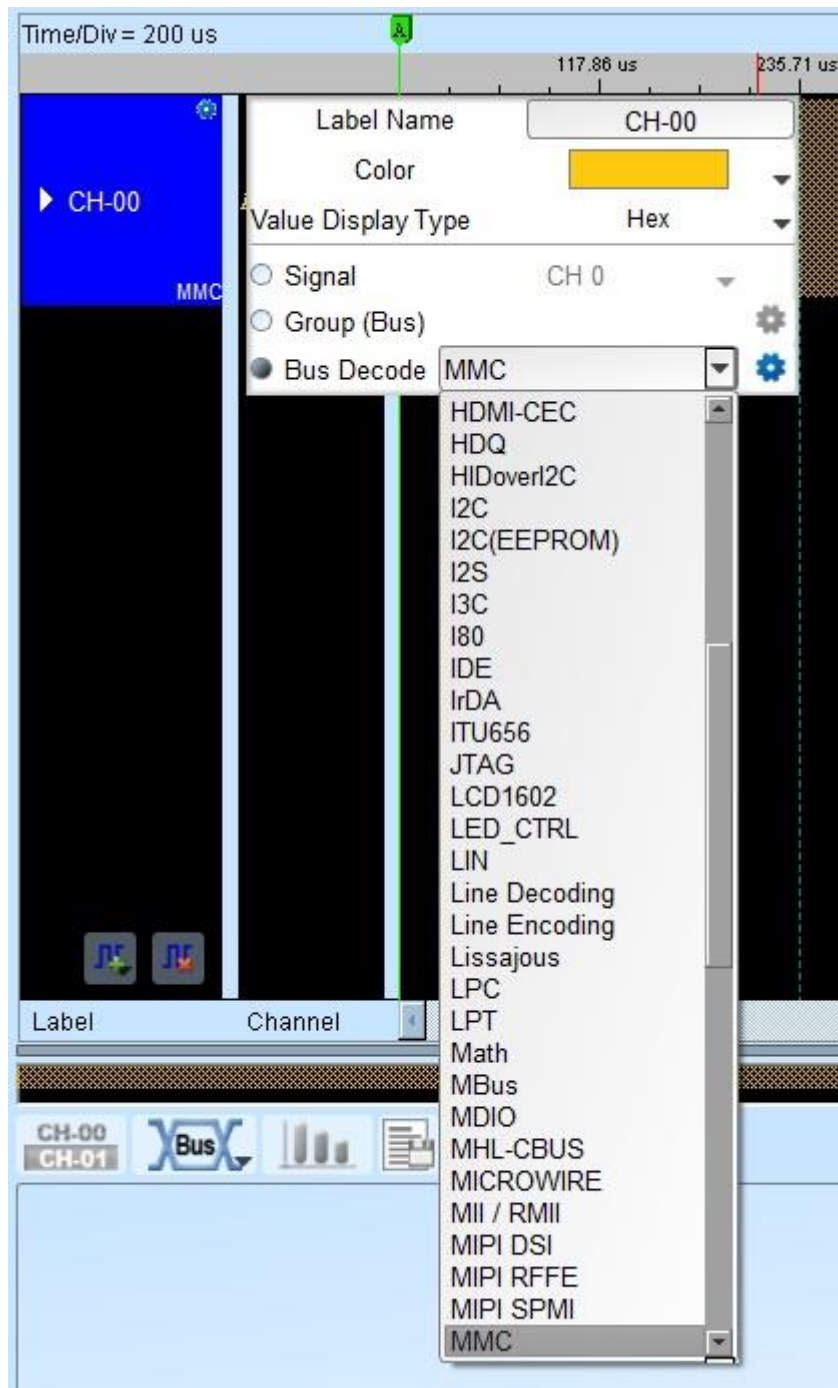
Click the Quick Setting  in the menu, and select the bus decode.



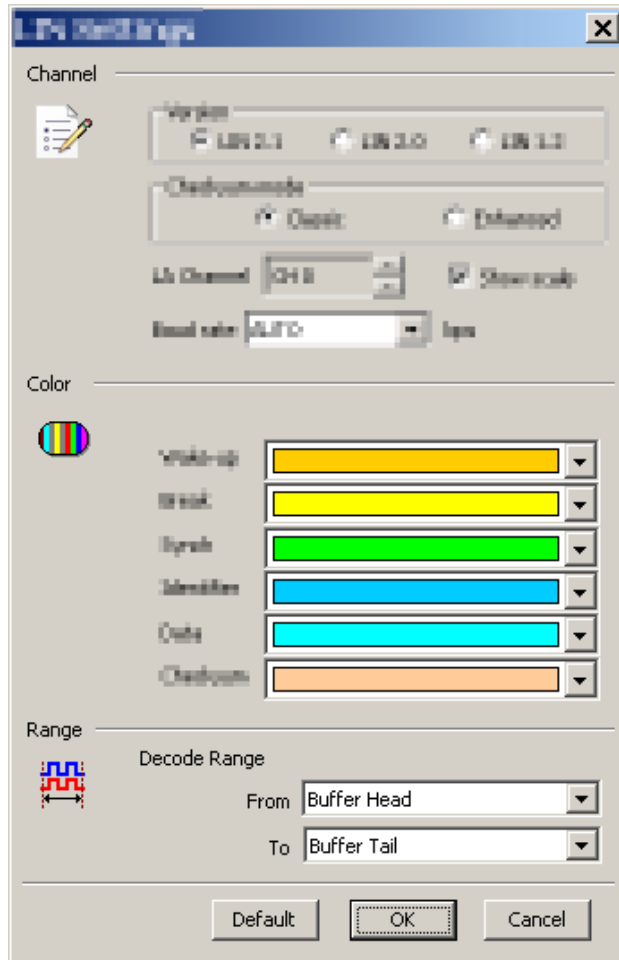
Method 2:

Click Add Bus Decode in the Label menu or right-click the label field to show the dialog box.

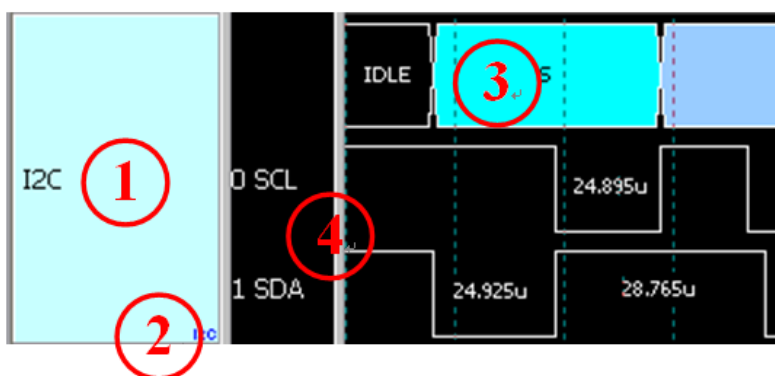
Advance channel setting



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”, “Color”, “Range” settings



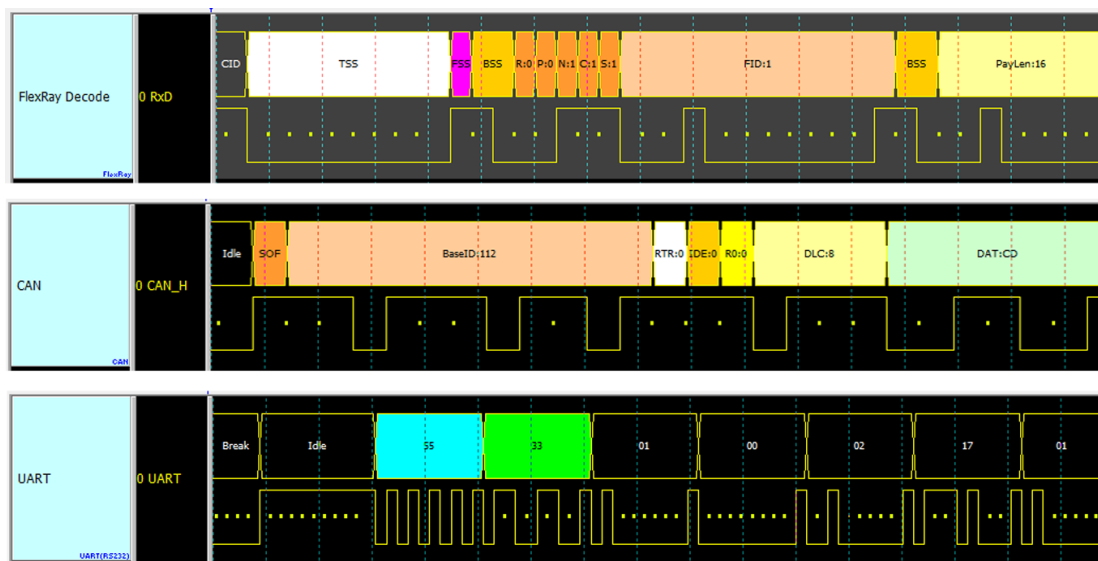
1. Bus name
2. Decode type
3. Result
4. Channel name & signal

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.

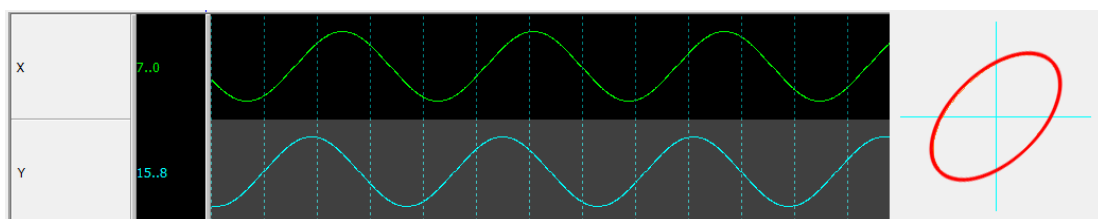
UART/CAN/FlexRay..bus decodes:

The data is displayed according to bit points in order to calculate the bit number ◦



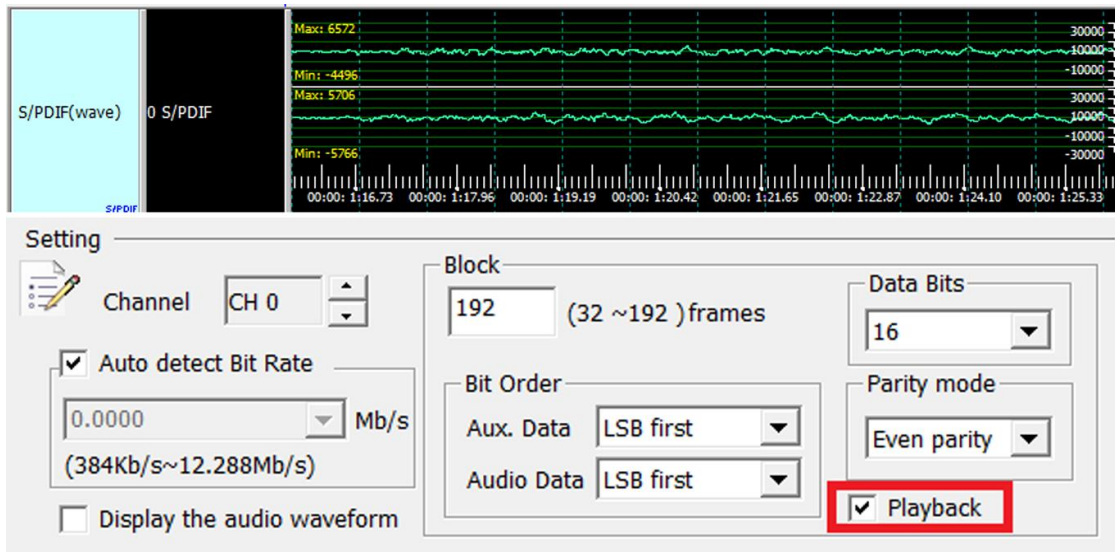
Lissajous analysis (released in 2009/09, LA Viewer Ver.2.0):

Display the signal in graph by X-Y or I-Q data. ◦



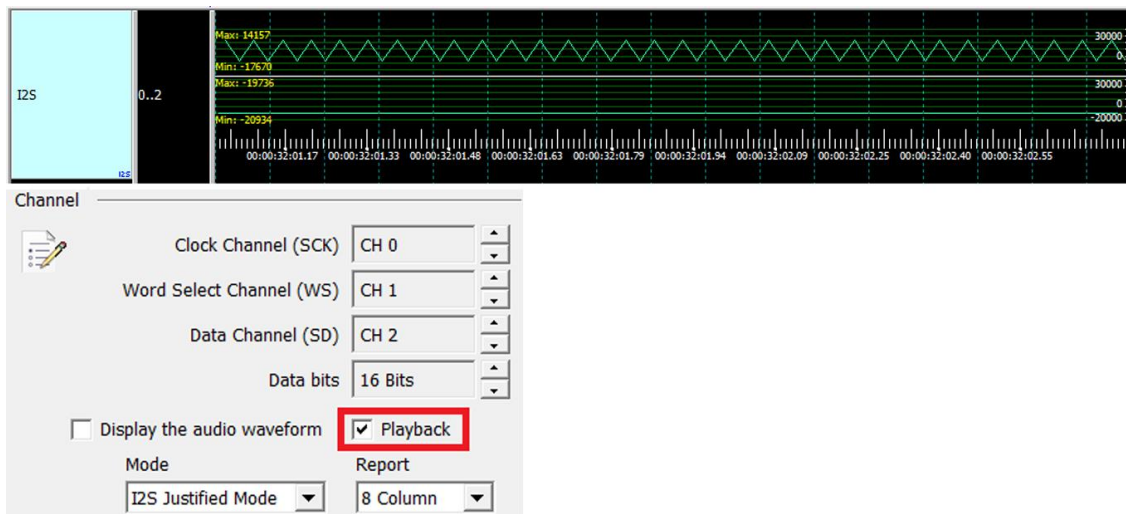
S/PDIF analysis:

Display the data in sound waveform ◦



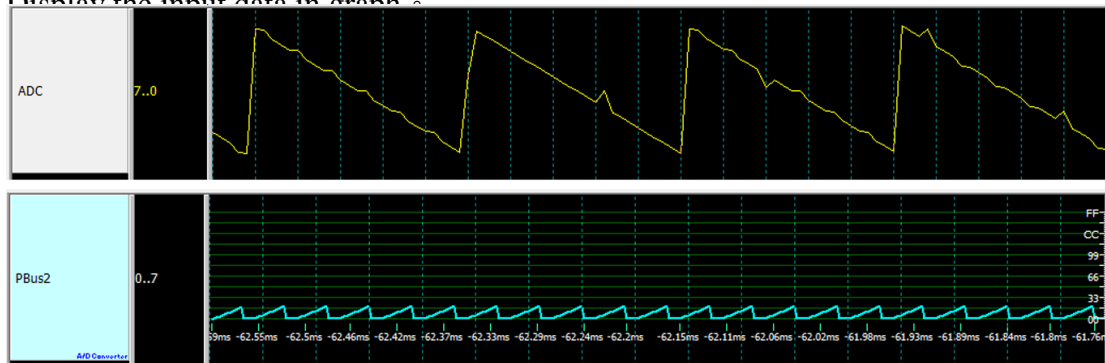
I2S analysis:

Display the data in sound waveform ◦



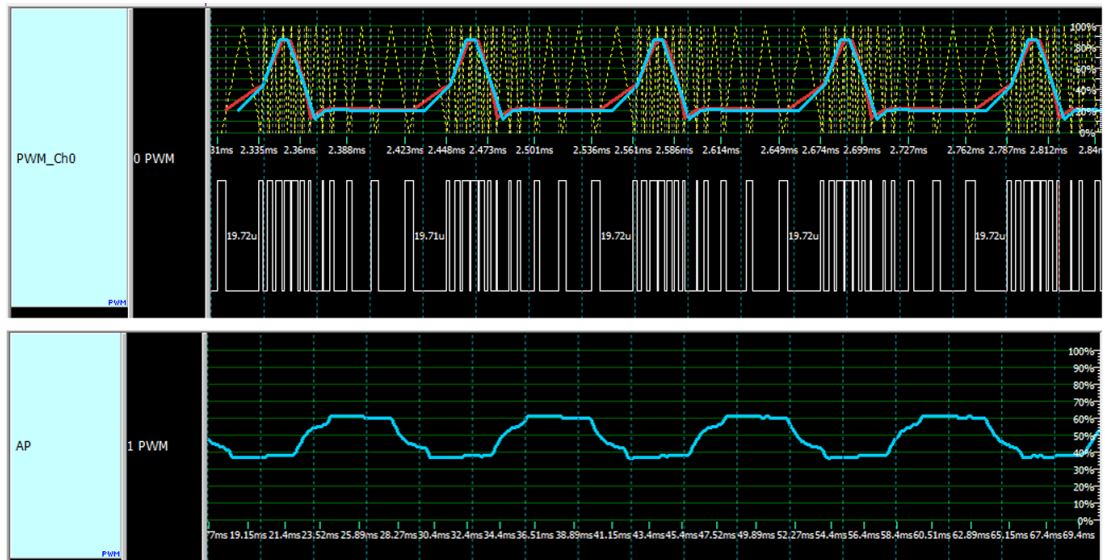
ADC bus decode:

Display the input data in graph



PWM analysis:

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

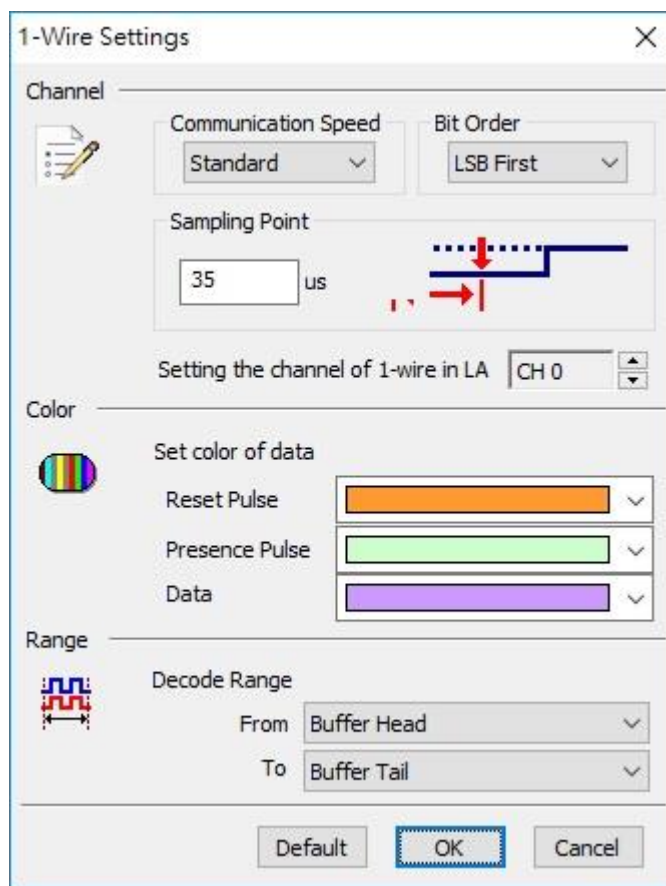


Bus Decode Settings Introduction

1-Wire

The 1-Wire bus has data bits (Reset Pulse, Presence Pulse, Write 1, Write 0, Read 1, Read 0) in standard or overdrive speed as the diagram below.

Settings



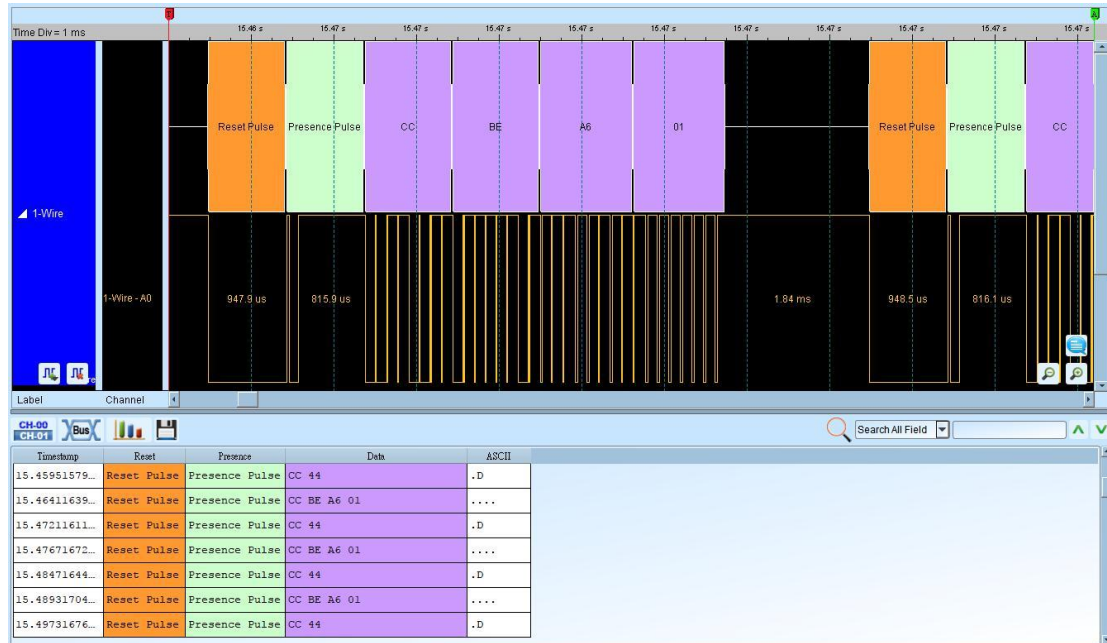
Communication Speed: Standard or overdrive.

Bit Order: LSB first or MSB first.

Sampling point: Set the sampling point N microseconds (us) after the beginning of each data bit.

Result

Click OK to run the 1-Wire decode and see the result on the Waveform Window below.



3-Wire

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

Settings

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.

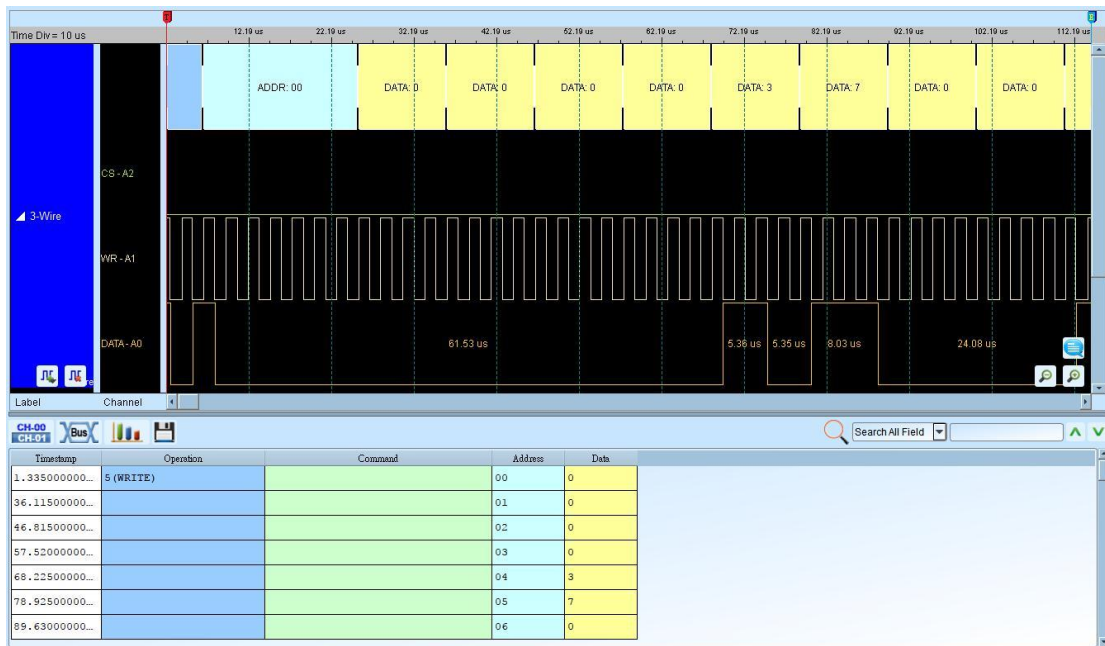
Active High: Select Active High.

Active Low: Select low chip select (CS).

Rising: Select Rising Data Edge.

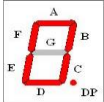
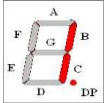
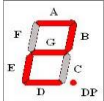
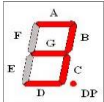
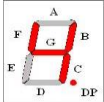
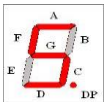
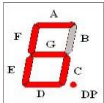
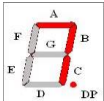
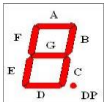
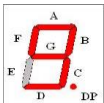
Falling: Select Falling Data Edge.

Result

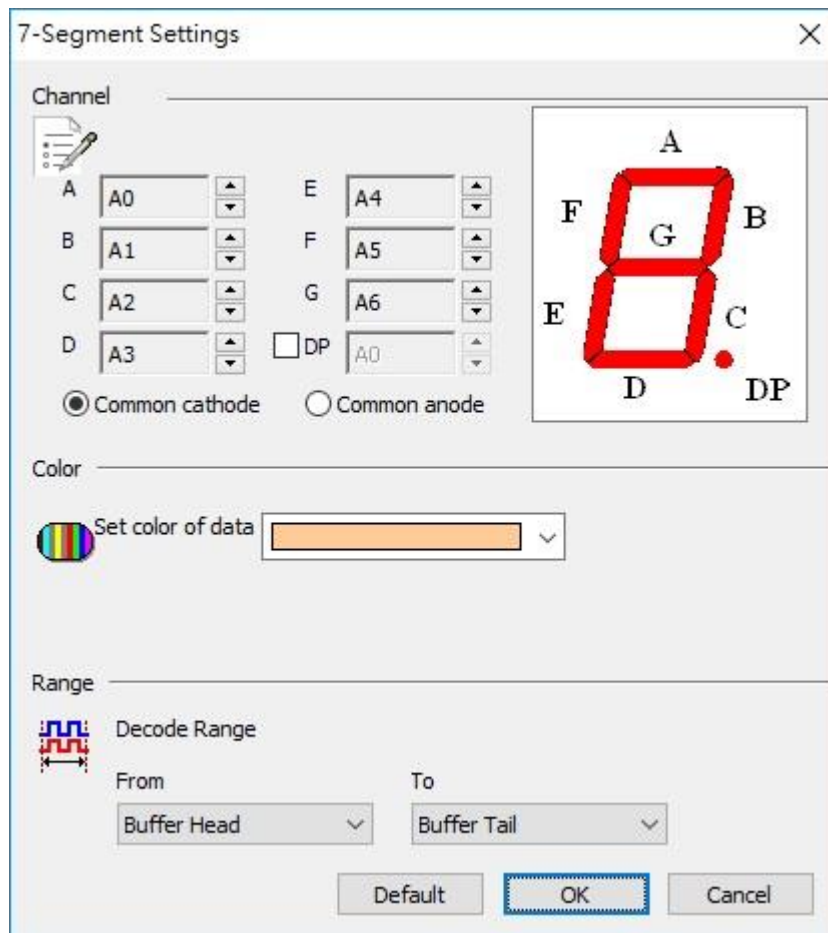


7-Segment

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



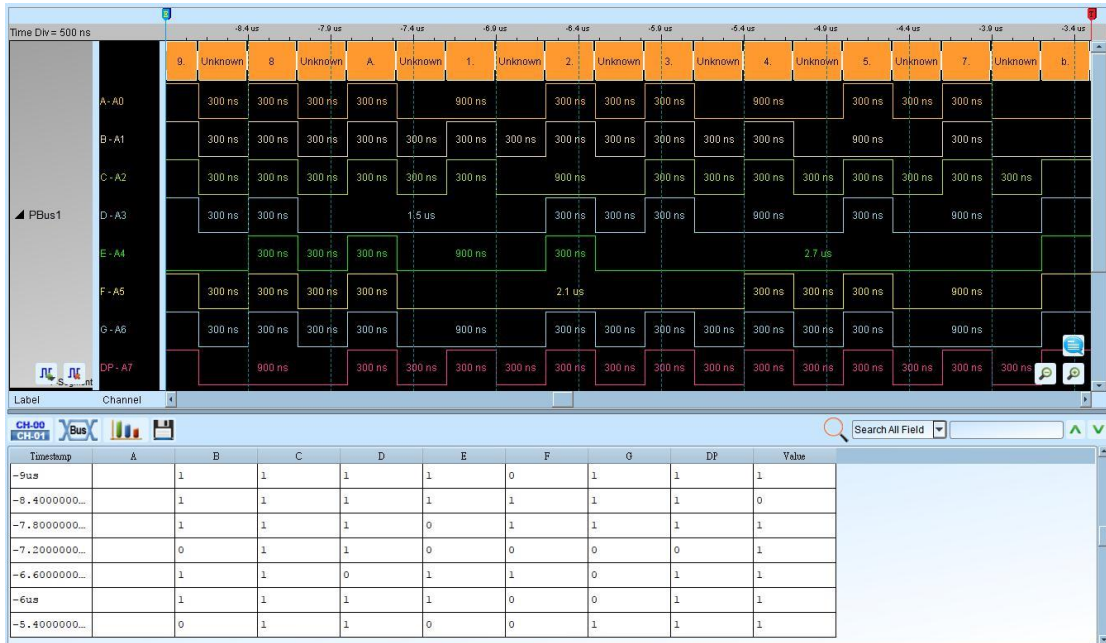
Channel: Show the selected channel (CH 0).

DP: to analysis decimal point.

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

Result

Click **OK** to run the 7 Segment decode and see the result on the Waveform Window below.



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

Data Channel Start From: ADC data channel start from

CLK Channel: ADC clock in channel

CS(OE) Channel: ADC chip select (output enable) channel

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

MSB First: Data bit starts form MSB; LSB defaulted

2's Complement: Show the 2's complement result.

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

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.

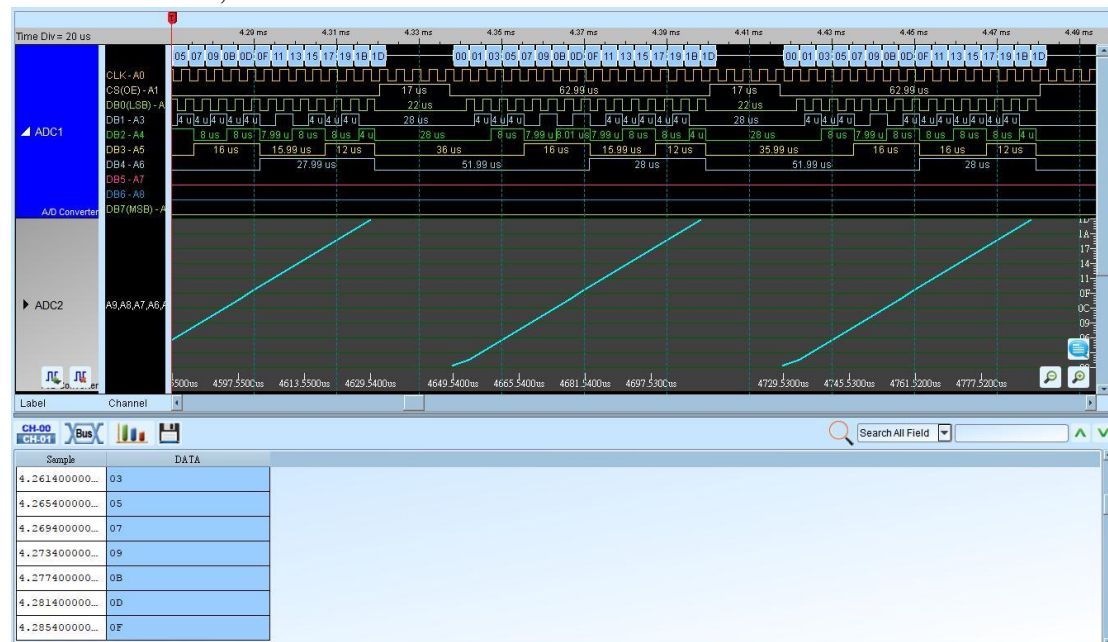
Insert Y axis bound: Set the maximum and minimum bounds of Y axis.

Note: When Insert Y axis boundaries is activated (Save it in the LA file directory), the Top and Bottom values will always be saved as an independent text file named as ADC.txt, different from the waveform file, at file work directory unless a different name is assigned. If you need the specific boundary settings, please save it, so you can reload the settings next time.

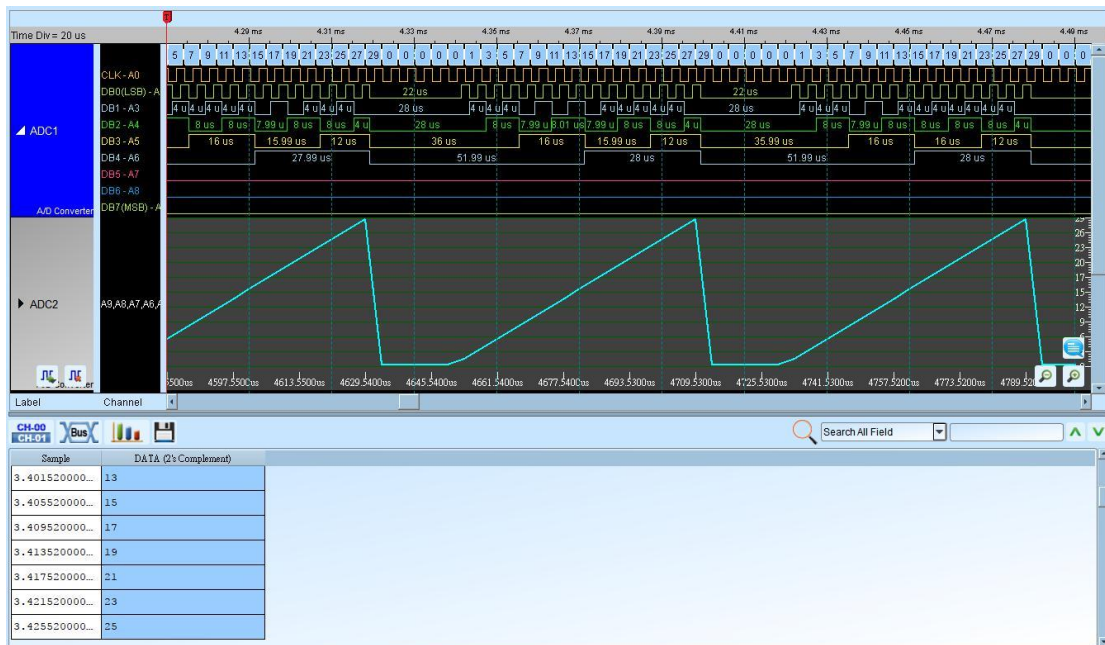
Result

Click **OK** to run the A/D Converter decode and see the result on the Waveform Window below.

Select 8-bit data, CLK/CS channels:



Select 8-bit data:



Accelerometer

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

Settings

The screenshot shows the 'AccMeter Settings' dialog box. It is divided into several sections:

- Parameter Settings:**
 - Channel Settings:** CS (A0), CLK (A1), SDI (A2), SDO (A3).
 - Edge Select:** CS (Active Low), SDI (Rising), SDO (Rising).
- Model:** AIS326DQ
- Initial Full-Scale:** 2 G
- Display Settings:**
 - Plot: Time (X) - Data (Y)
 - X Y Z
 - Advanced Decode
 - Calculate Average: (N - 0) to (N + 0)
- Color:**
 - R/W: Yellow
 - M/S: Green
 - Address: Cyan
 - Data: Blue
- Range:**
 - Decode Range: From Buffer Head to Buffer Tail

Buttons: Default, OK, Cancel

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.

Model: The IC model of the target accelerometer.

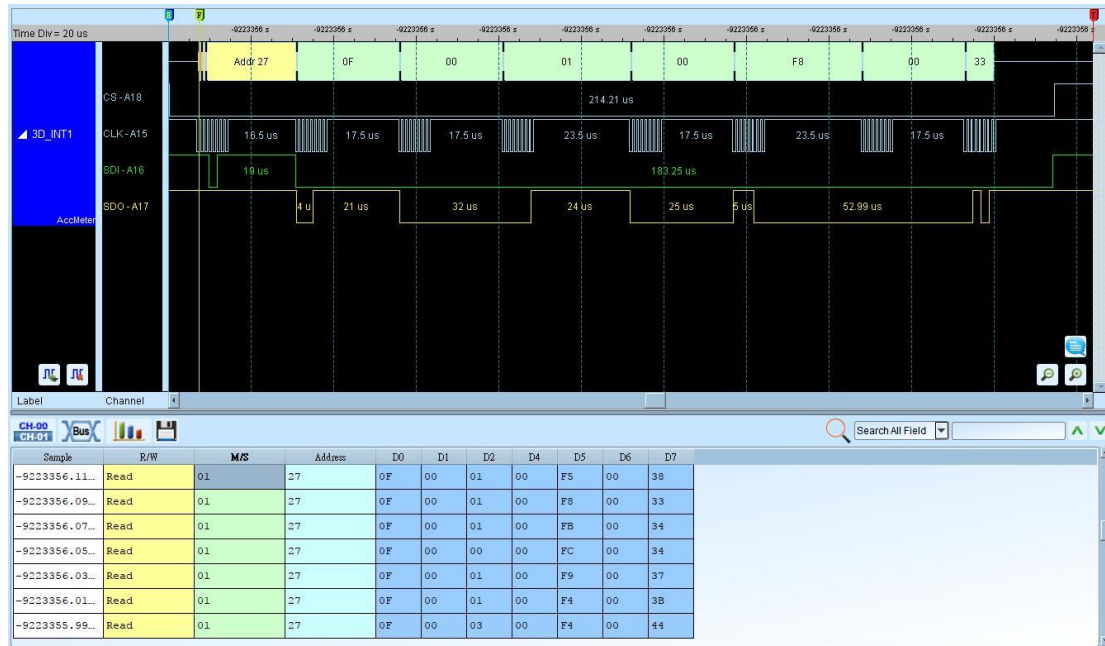
Initial Full-Scale: The default Full-Scale setting.

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

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

Result

Standard decoder result



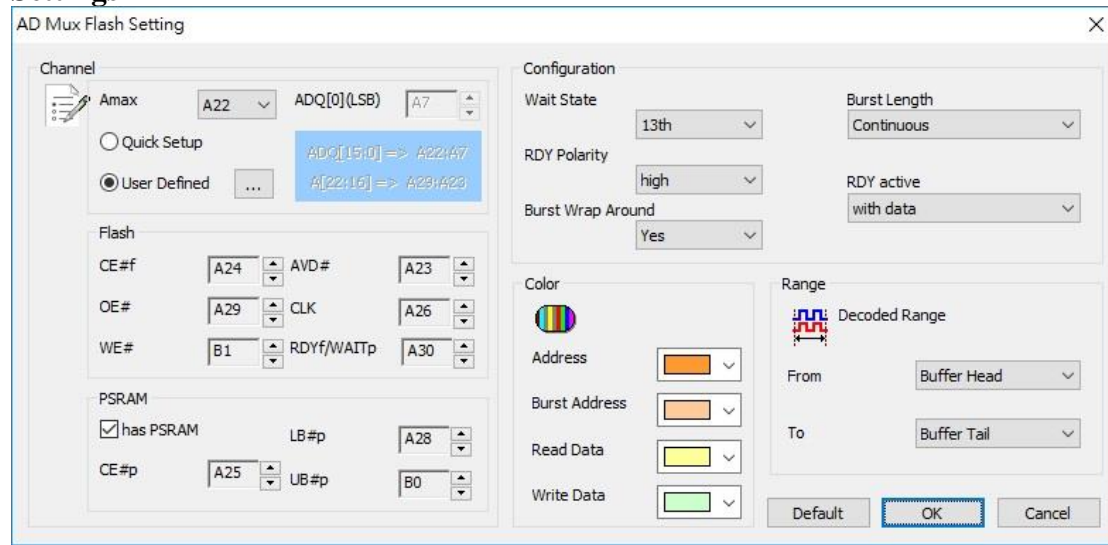
Advanced decode result + Time-Value curve display



AD-Mux Flash

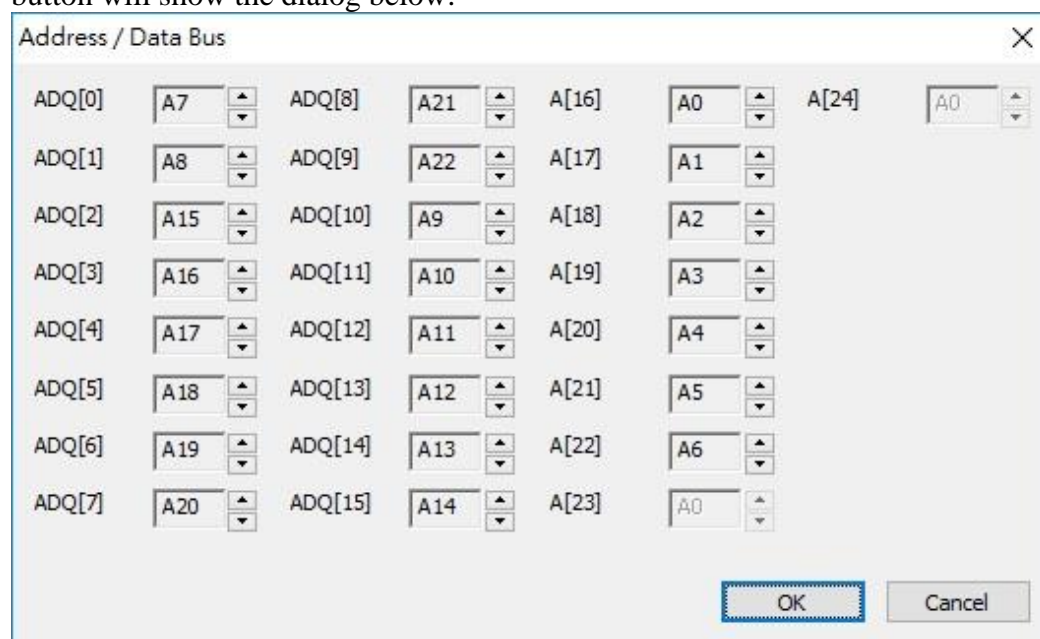
AD-Mux Flash is one kind of parallel flash that utilize an Address and Data multiplexed interface.

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:



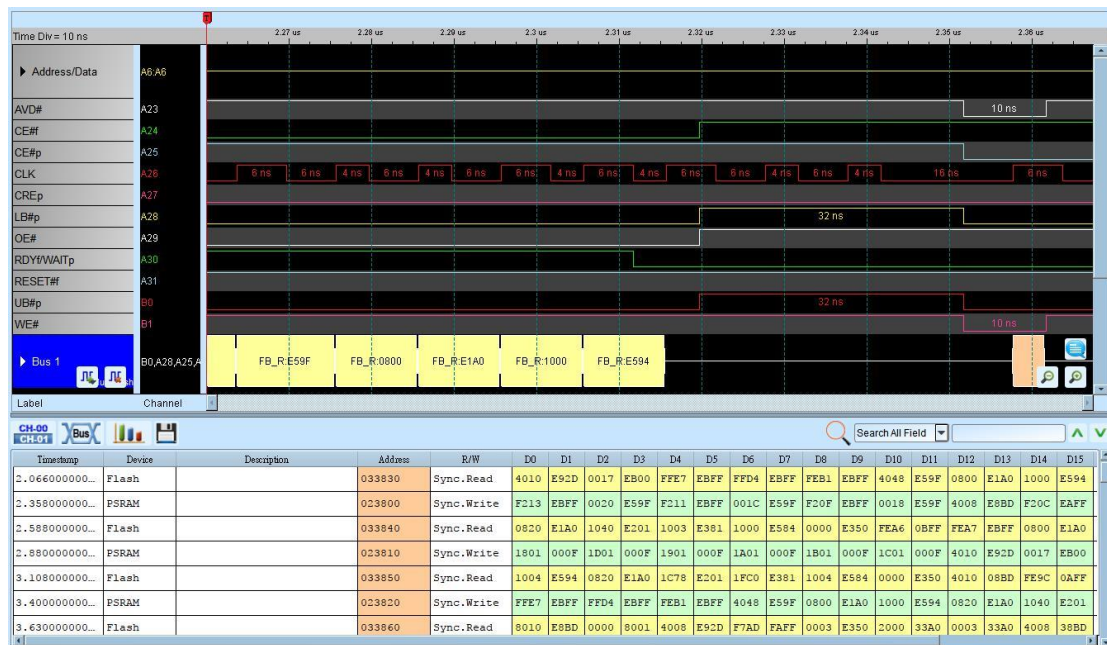
Flash: Control pins of flash.

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

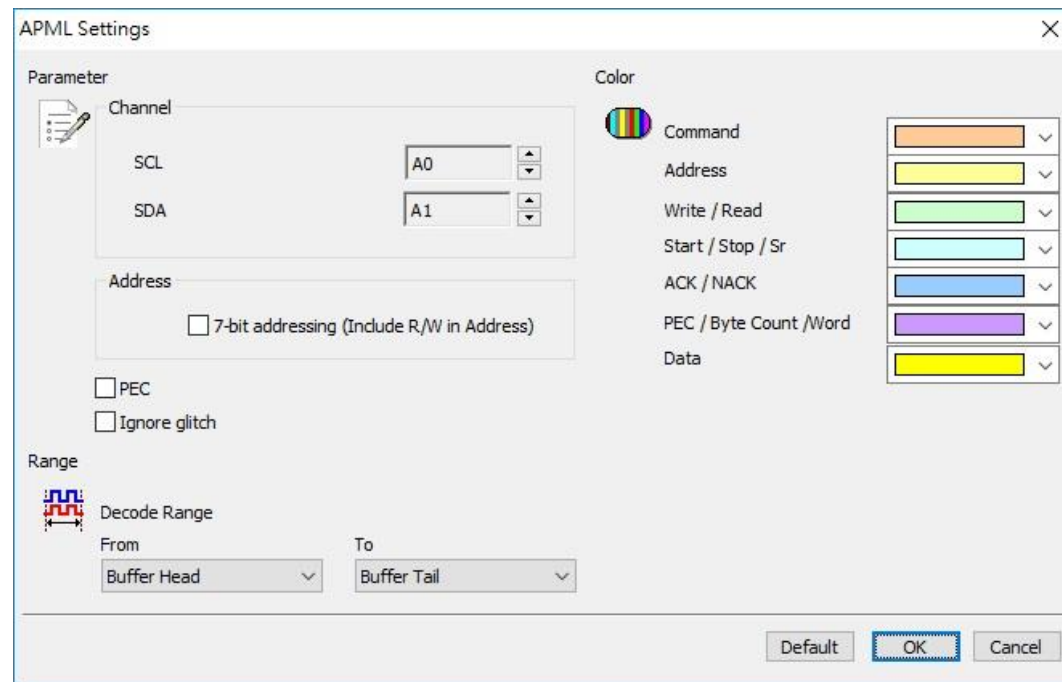
Click OK to run the AD-Mux Flash Decode and see result on the Waveform Windows below.



Advanced Platform Management Link (APML)

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

Settings



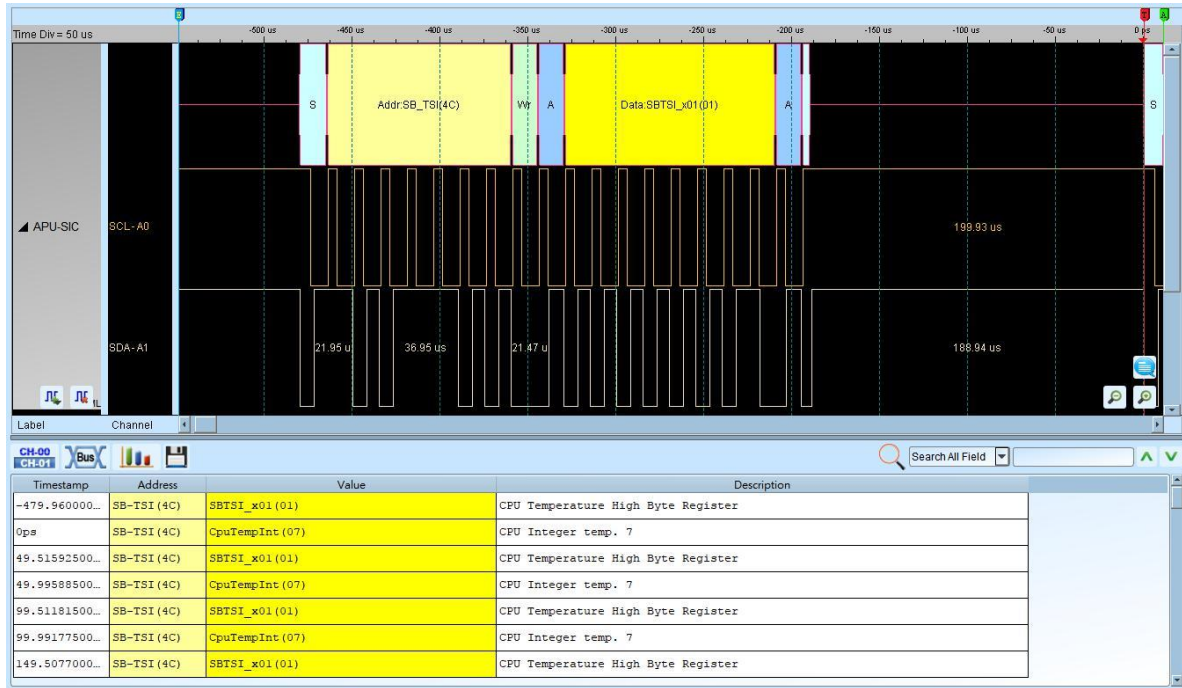
Channel: Show the selected channels (CS:CH0, WR:CH1, DATA:CH2, RD:CH3)

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

PEC: Packet Error Check.

Ignore glitch: Ignore the glitch when the slow transitions.

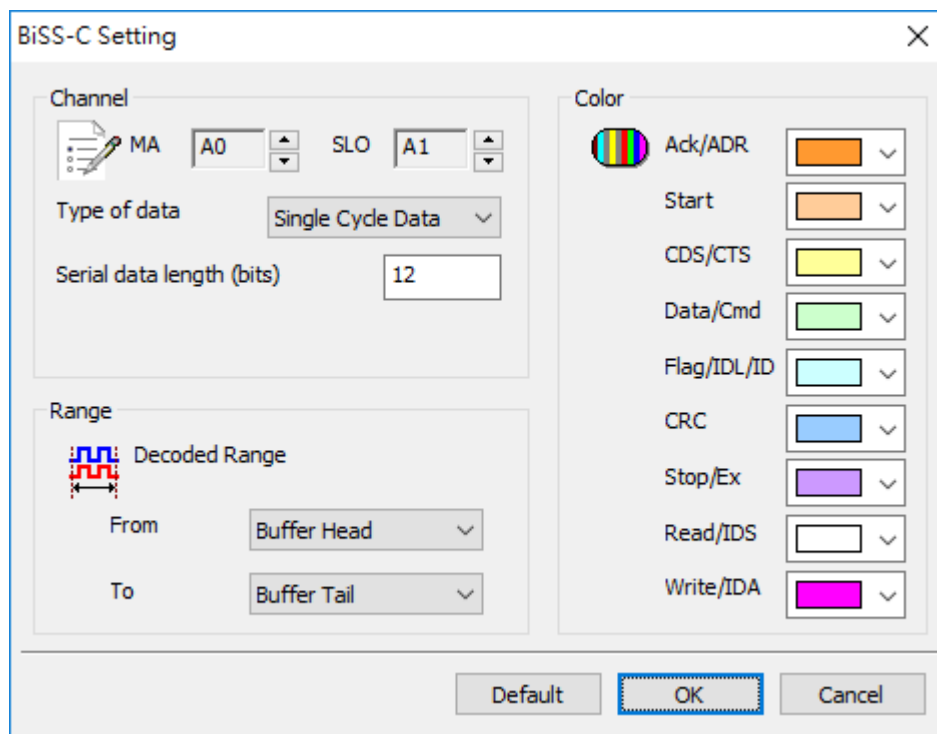
Result



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



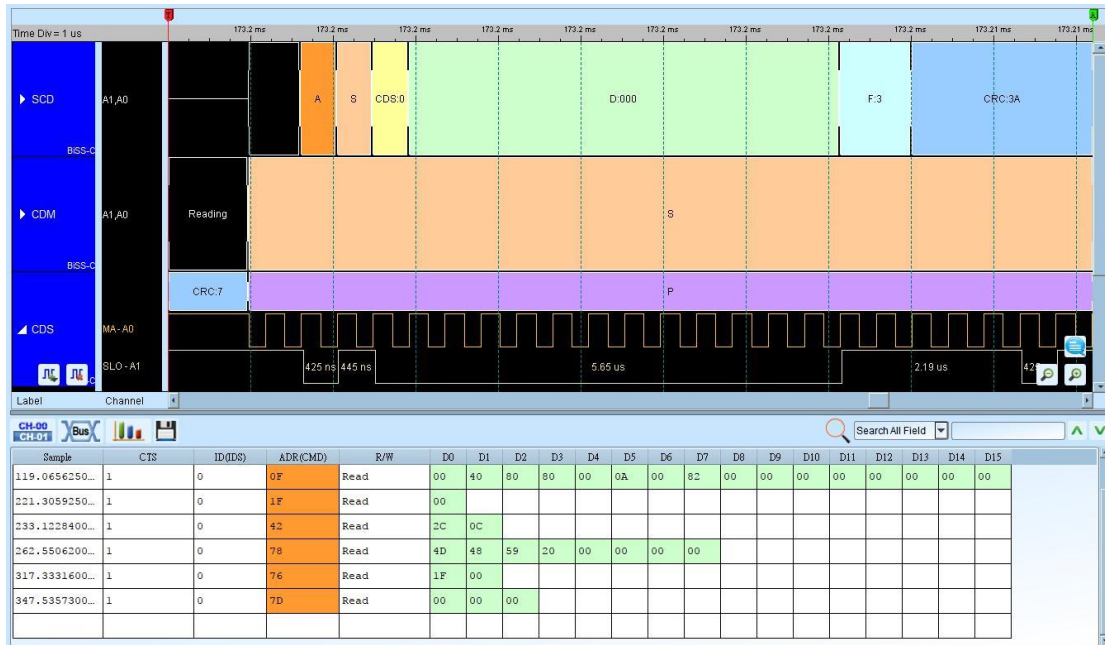
MA/SLO: Setting the channel of MA and SLO.

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.

Result

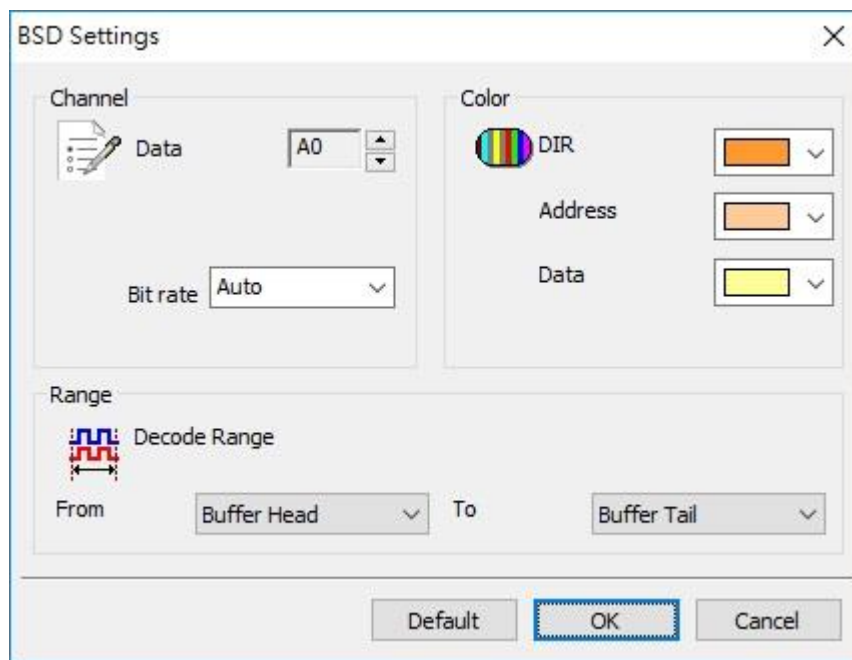
Click OK to run the BiSS-C Decode and see result on the Waveform Windows below.



BSD

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

Settings

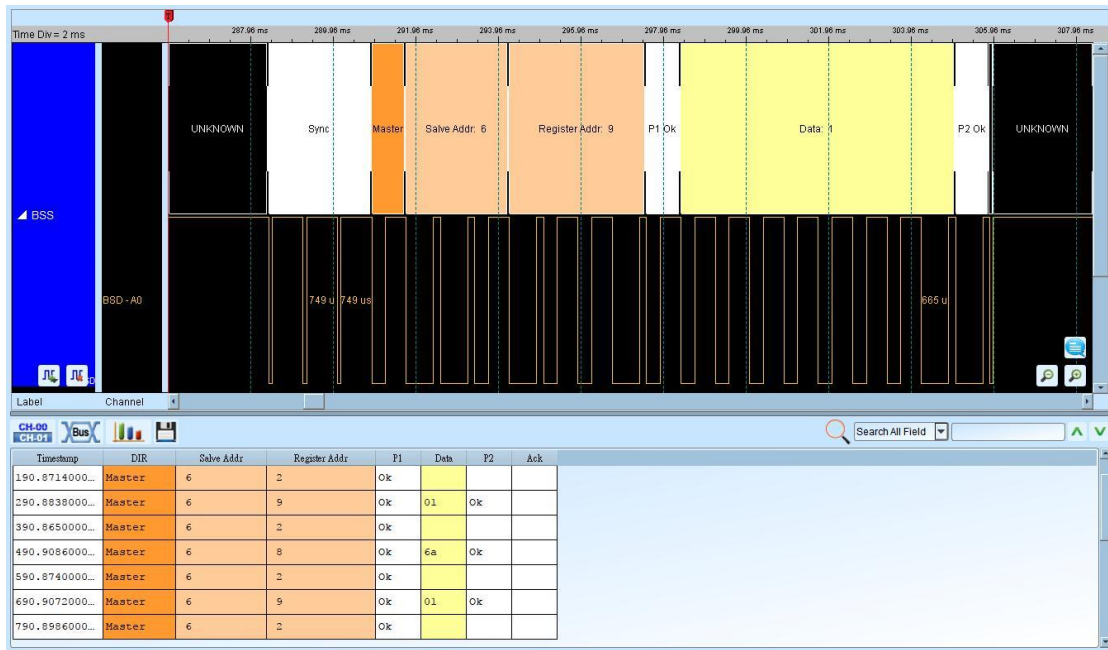


Data: The BSD data.

Bit rate: The bit rate of the BSD data

Result

Click OK to run the BSD Decode and see result on the Waveform Windows below.



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.

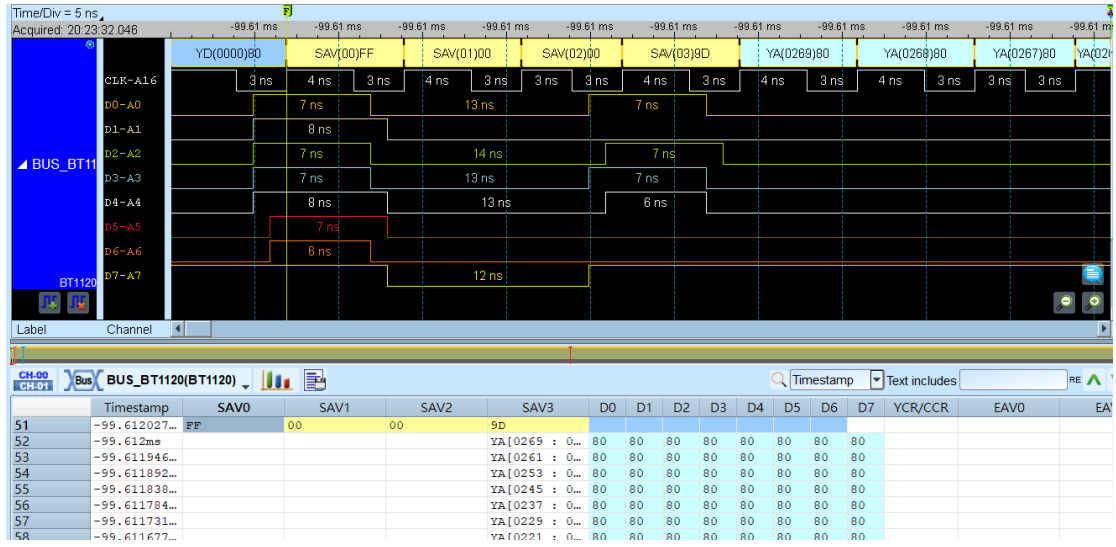
Settings

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

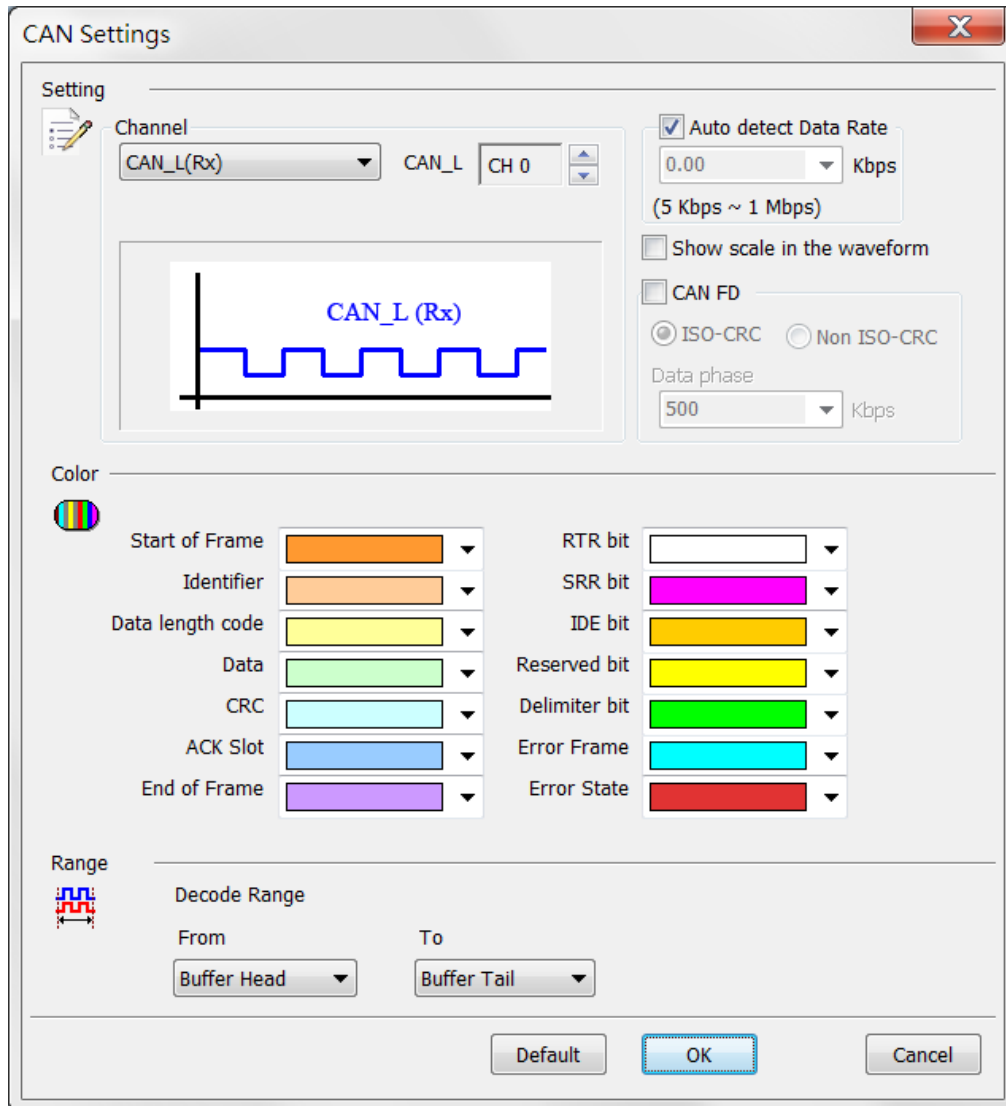


CAN 2.0B/ CAN FD

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



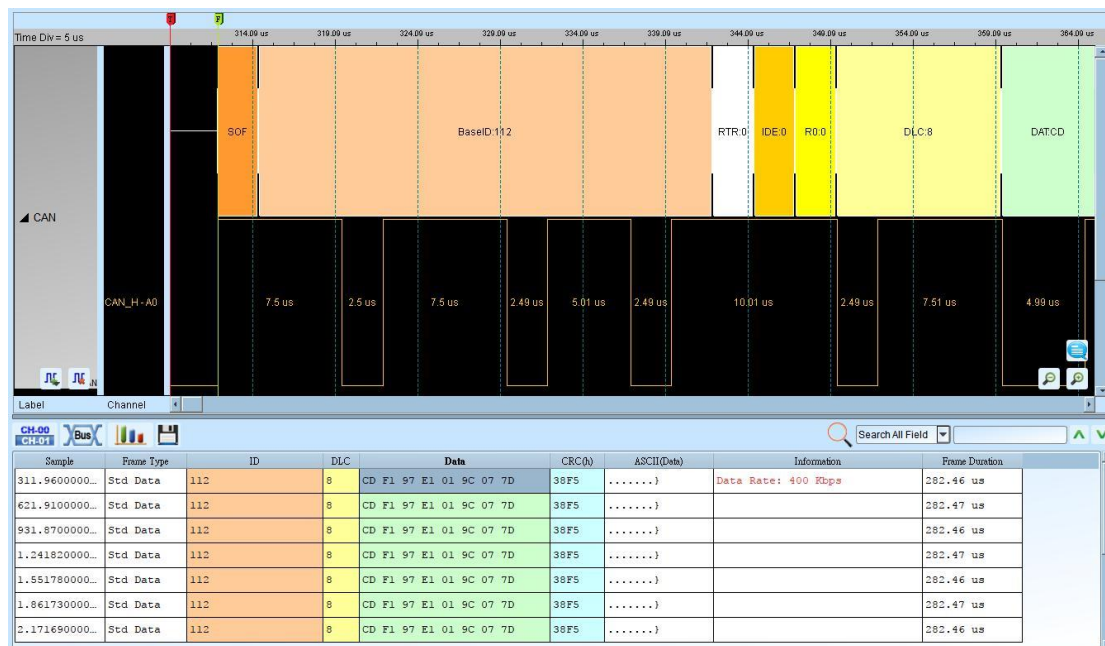
Channel: The differential data from the DSO channel (CAN_H or CAN_L) is shown by default.

Auto detect Data Rate: Check this option for auto-detecting the CAN bit rate by the LA Viewer; this option will be disabled when enabling CAN FD decode, the maximum input range of the Data rate is from 5Kbps-1Mbps.

Show scale in the waveform: Display the scale in the Waveform Window, this option will be disabled when enabling CAN FD decode.

Result

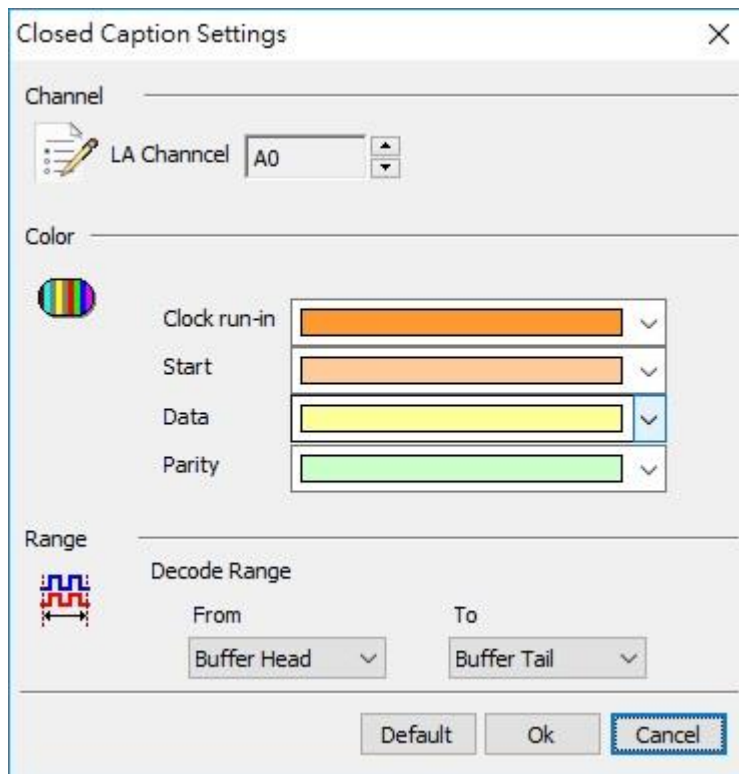
Click **OK** to run the CAN decode and see the result on the Waveform Window below.



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.

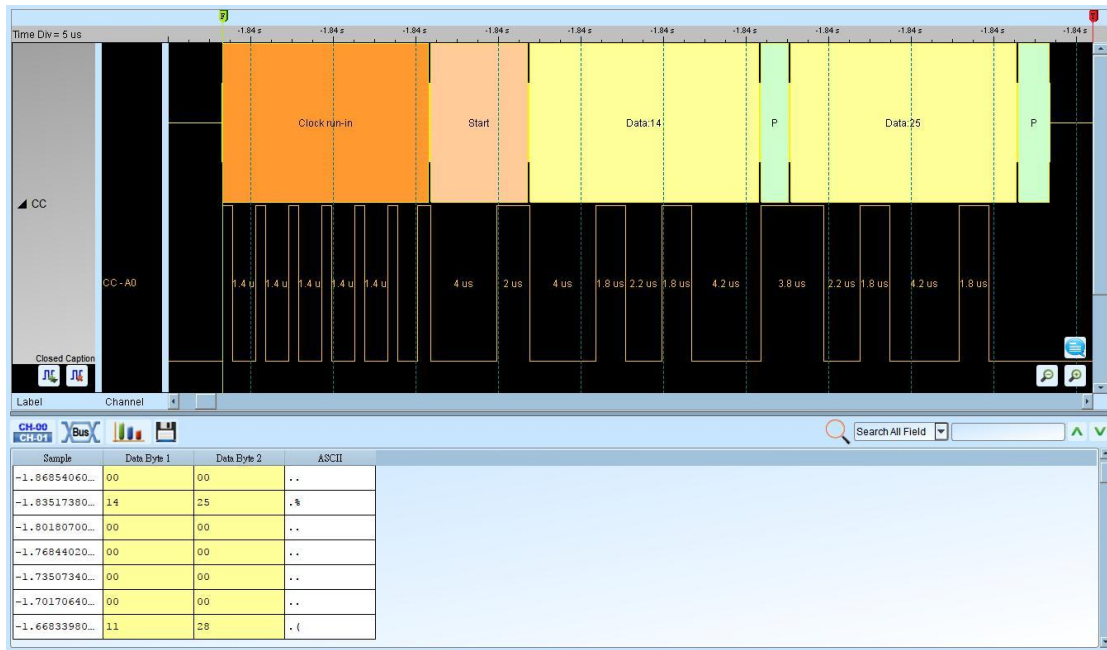
Settings



LA Channel: Show the selected channel (CH0).

Result

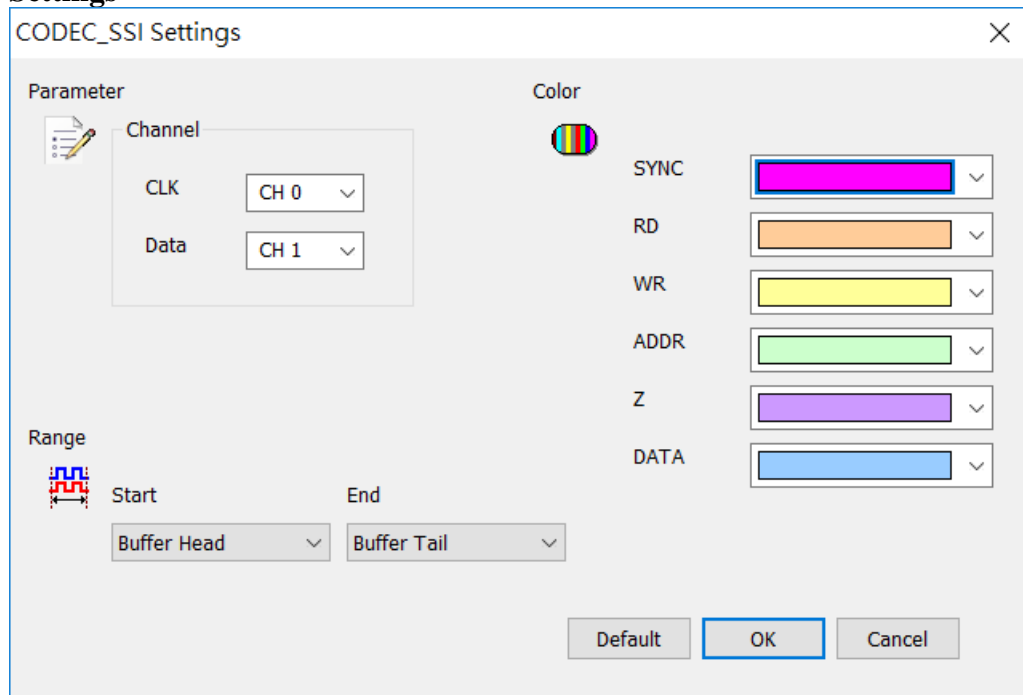
Click OK to run the Closed Caption Decode and see result on the Waveform Window below.



Codec SSI

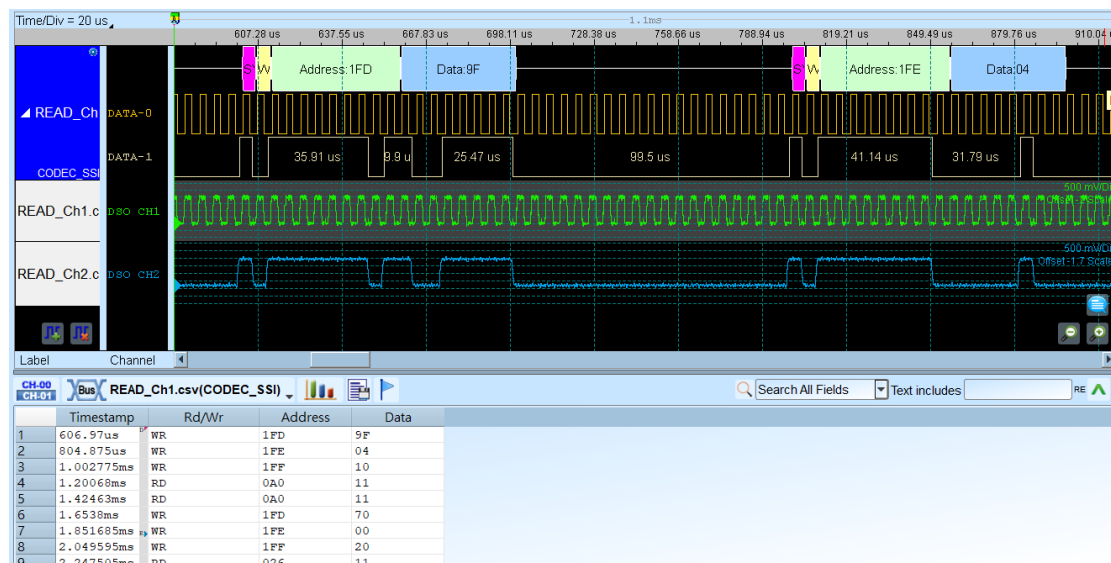
Serial Synchronous Interface (SSI) signal used by the codec (CODEC) inside the phone

Settings



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

Result



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

LA Channel: Show the selected channel (CH0).

Polarity:

D-: Access side of the signal polarity is D-.

D+: Access side of the signal polarity is D+.

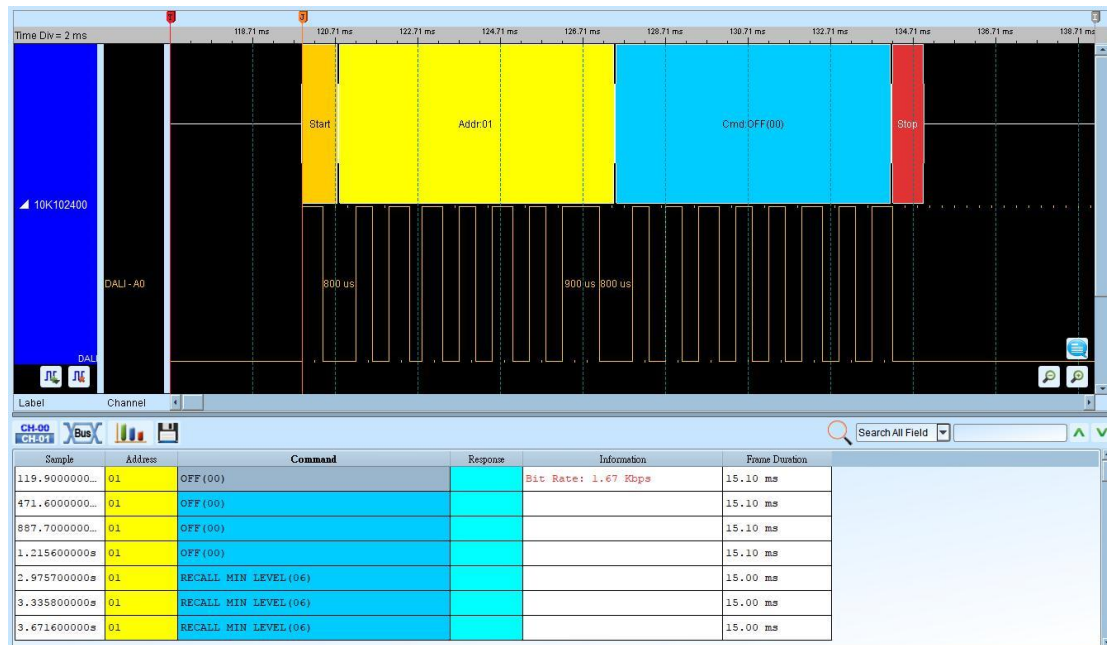
Auto: Automatically detect the polarity of the access end signal.

Show scale: Show Scale on the waveform.

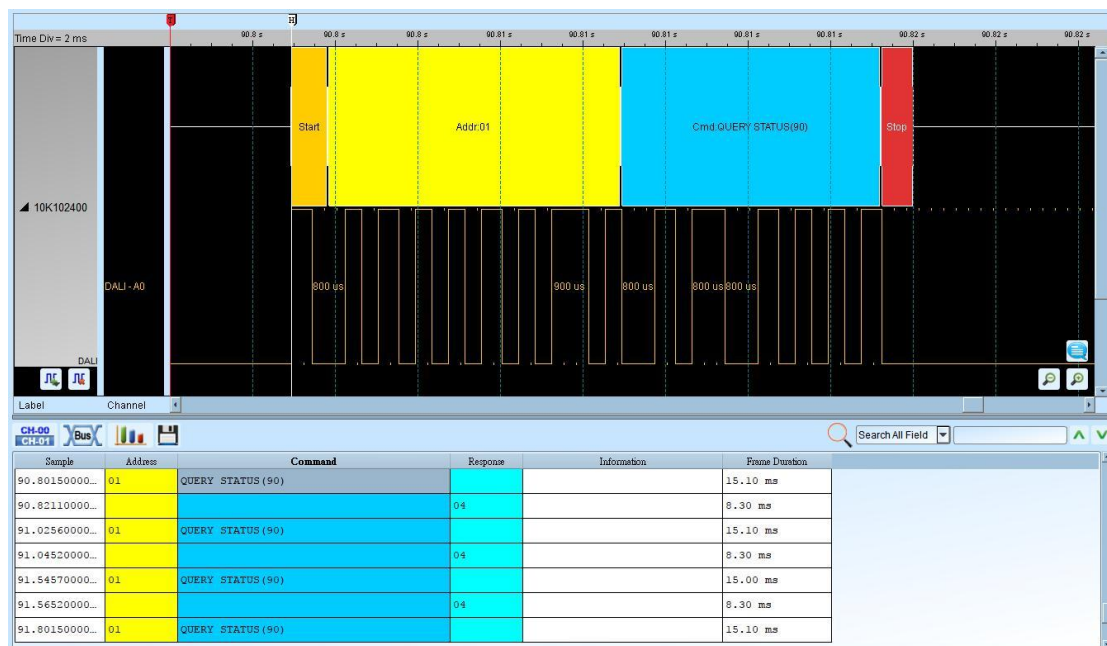
Result

Click OK to run DALI Decode and see result on the Waveform Window below.

Send data



Response data



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

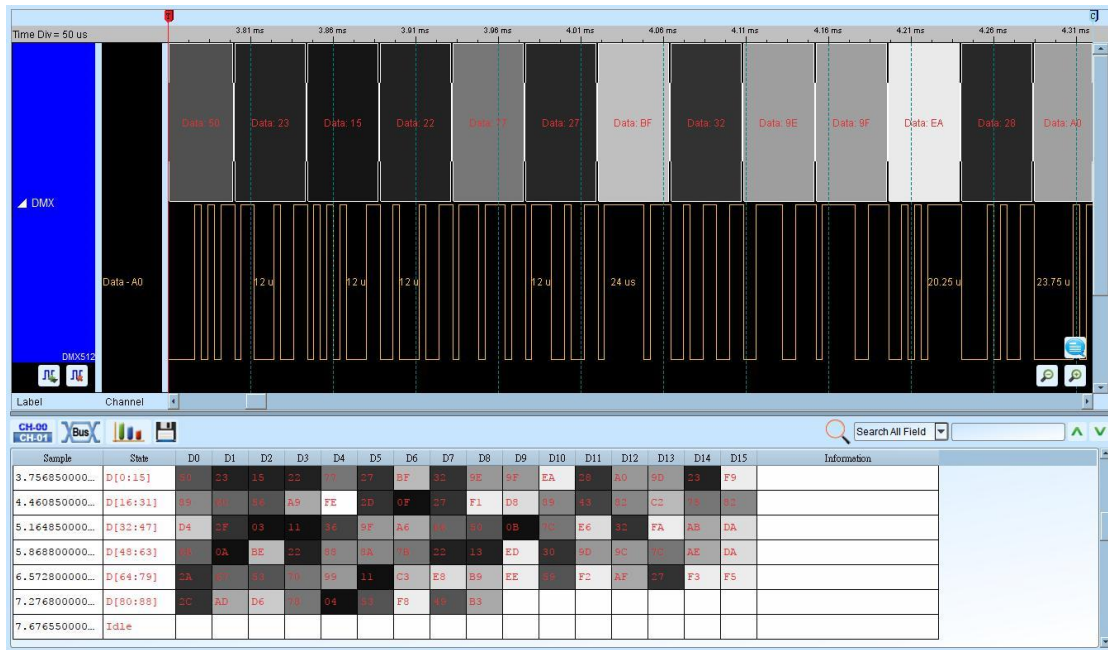
Default OK Cancel

Data: Show the selected channel (CH0).

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

Result

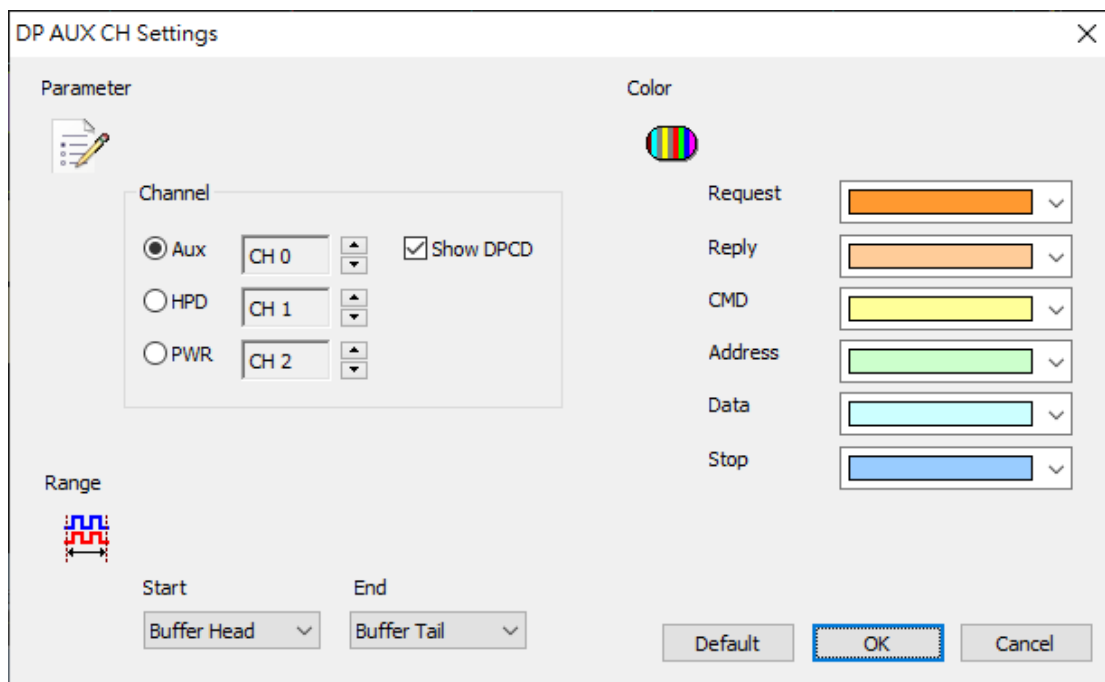
Use grayscale to display the decode results.



Display Port Auxiliary Channel (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 1.3, eDP 1.4)

Settings



Channel: Set the channel to decode

Show DPCD: Show the Display Port Configuration data.

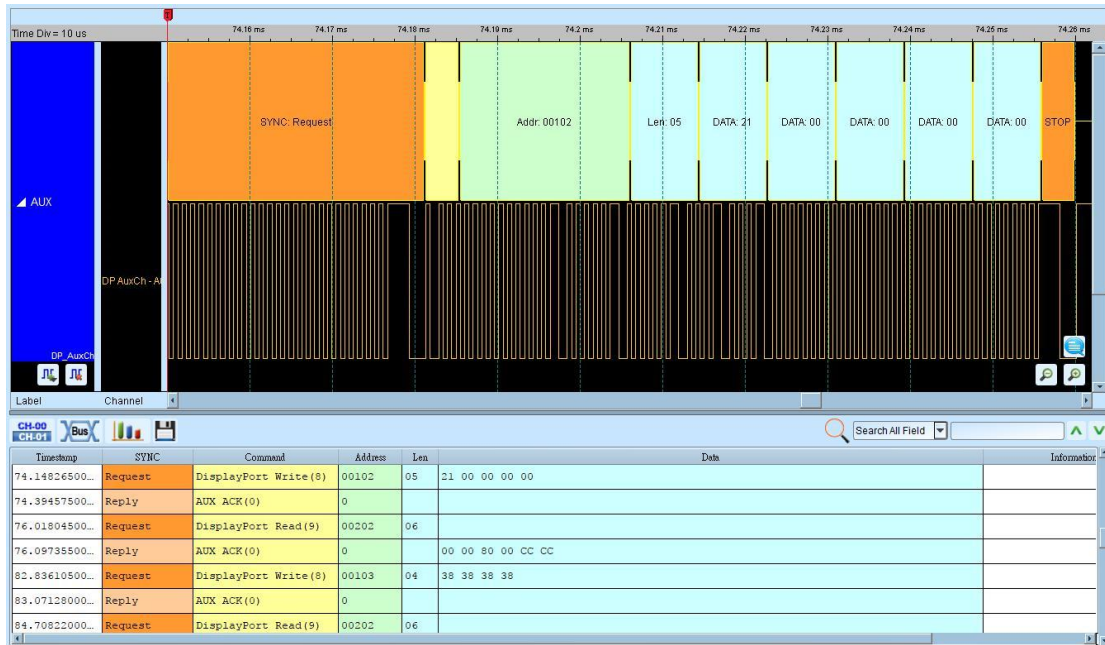
Mode: Choose the mode DP_Aux/HPD/PWR

Startup transaction: Set the transaction type of the first frame

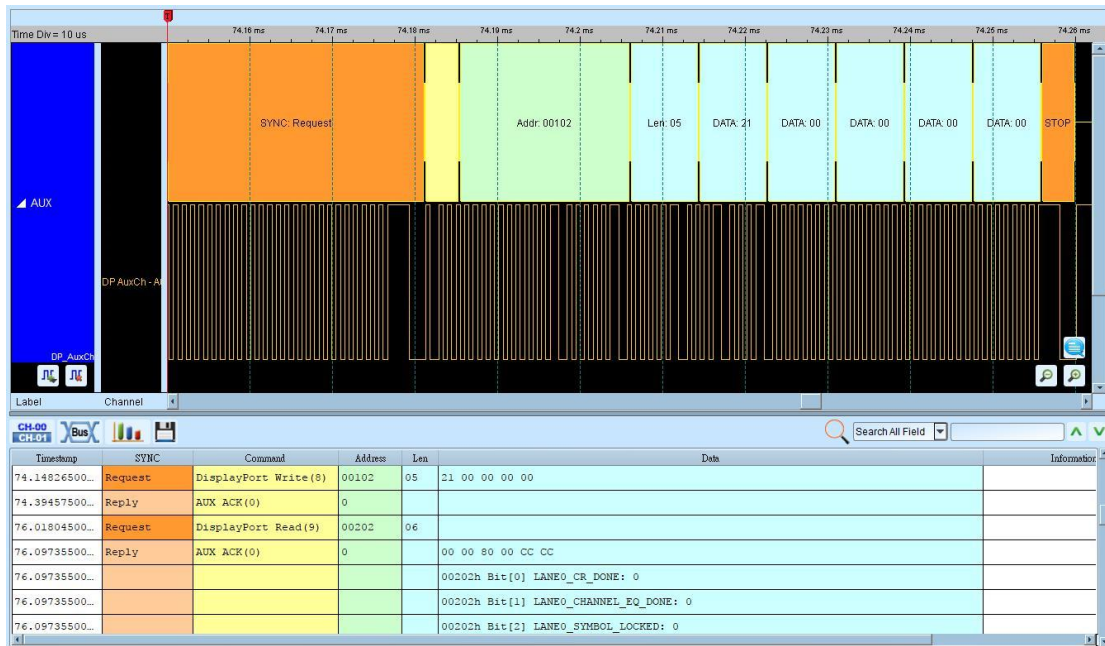
Reply Timeout: Set the value of timeout

Result

Without the DPCD information



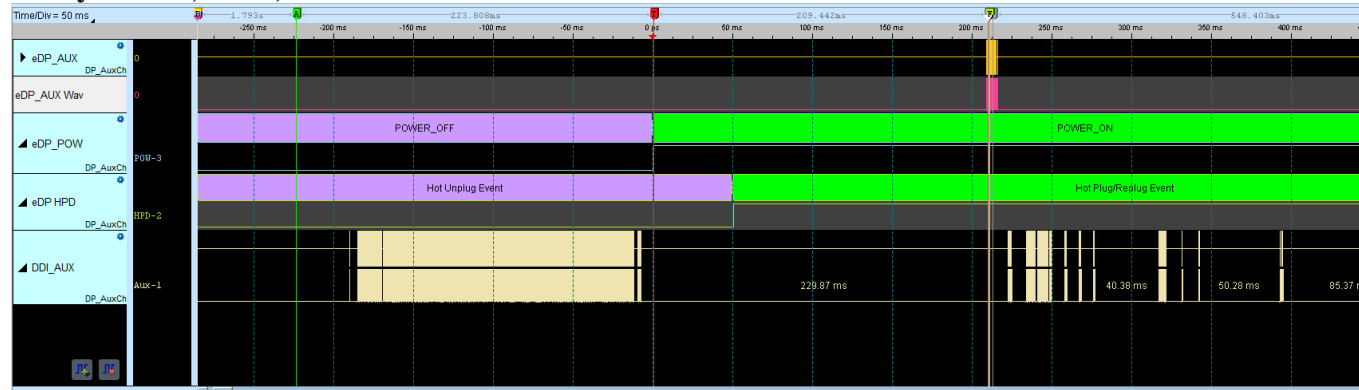
Show the DPCD information



Accessories:

http://www.acute.com.tw/eng/p3_product_detail.php?cid=13&tid=30

Analysis Aux, HPD, PWR



CH-00 Bus Customized Report

Timestamp	eDP_POW State	eDP HPD Event	DDI_AUX SVNC	DDI_AUX Command	DDI_AUX Address(h)	DDI_AUX Len	DDI_AUX Data(h)
1 -2.01855803s	Power Off						
2 -2.01655803s		Hot Unplug Event					
3 -189.897305ms			Request				
4 -189.850845ms			Reply				
5 -189.743675ms			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_DM_DEVICE_DP_FWR_5V: 0
9 -185.179985ms							[6]SET_DM_DEVICE_DP_FWR_12V: 0
10 -185.179985ms							[7]SET_DM_DEVICE_DP_FWR_18V: 0
11 -185.04335ms			Reply	AUX ACK (0)			
12 -184.942065ms			Request	Native Read (4)	00200	1	

Enhanced Serial Peripheral Interface (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 Single Mode
 Default Alert Mode From I/O[1]

Auto-select protocol timing by clock speed

Command deselect time 50ns
 Clock LOW to output valid 15ns

Advanced Decode Setting

Show RAW Byte Only
 Show Configuration Detail
 Show Status Bit Def.
 Show VWire Detail
 Show EC/KBC Command
 Show OOB Detail
 Reduced Report

Default Display
 PUT_PC
 PUT_PC
 PUT_PC
 PUT_PC

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.

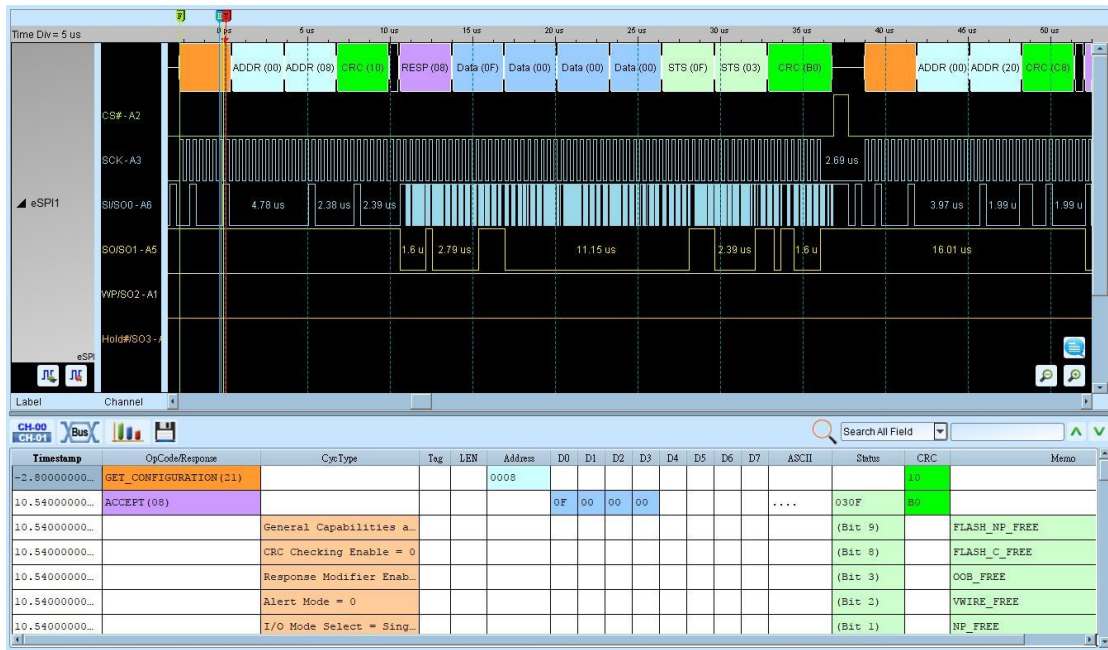
Reduced Report: Reducing the report is easy to check the Command Flow.

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

Click **OK** to run the eSPI decode and see the result on the Waveform Window below.



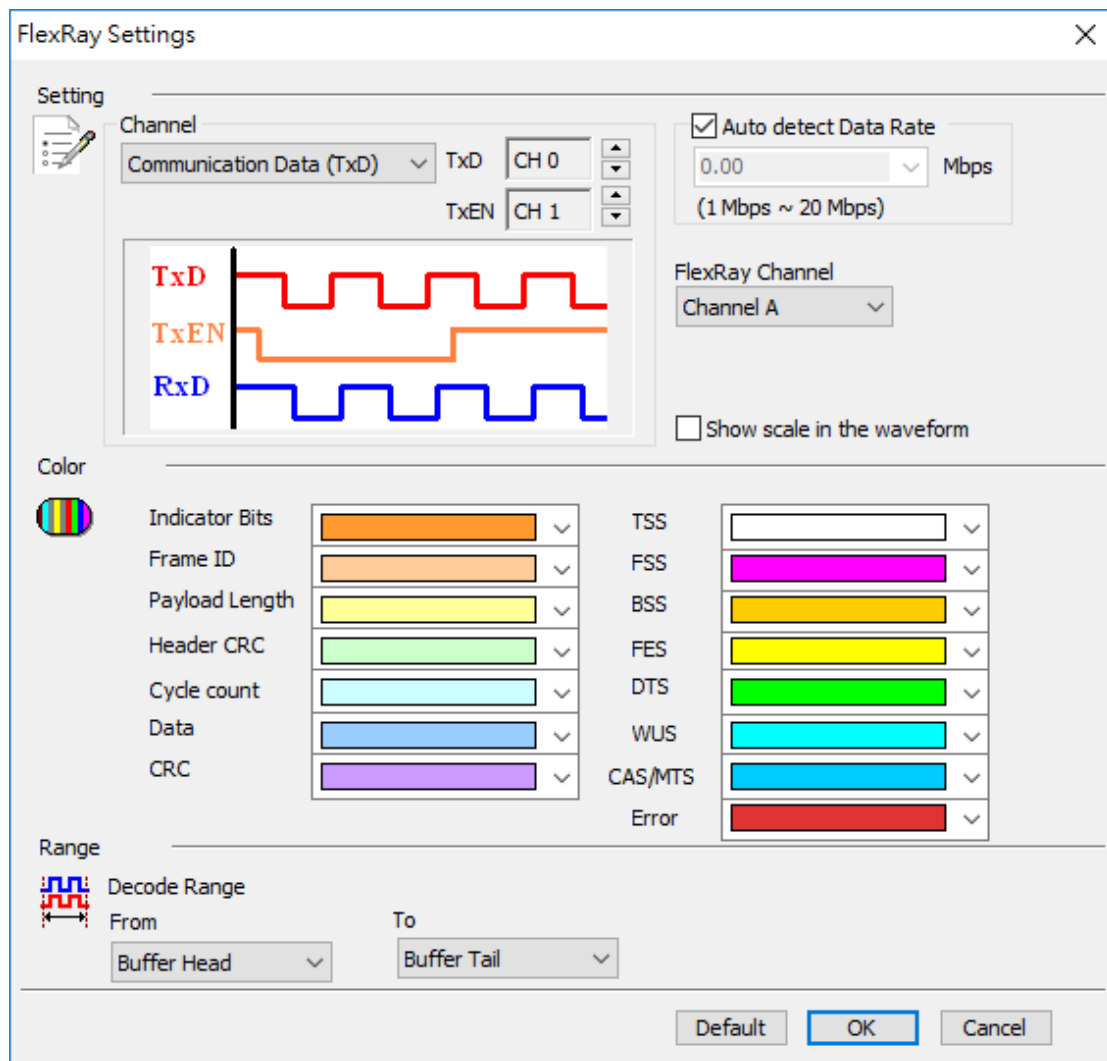
FlexRay

The FlexRay protocol has 2 bits with timing at 10Mbps.

FlexRay signal

Communication Data Layer: The FlexRay communication data is either at transmitter (TxD) or receiver (RxD) of the FlexRay transceiver. You may set the threshold according to the FlexRay transceiver voltage.

Settings



Channel: Display the channel, Physical Layer is the default.

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:

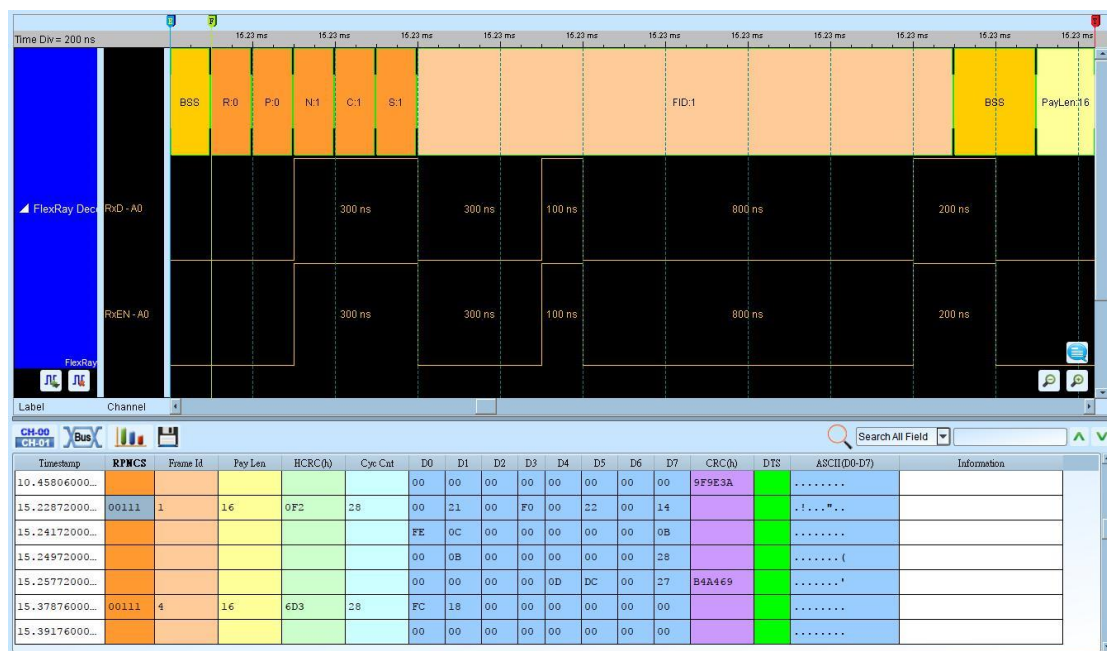
Abbreviation	Description
TSS	Transmission start sequence
FSS	Frame start sequence
BSS	Byte start sequence
FES	Frame end sequence
DTS	Dynamic trailing sequence
CAS	Collision Avoidance Symbol
MTS	Media Access Test Symbol

WUP	Wakeup Pattern
CID	Channel Idle Delimiter

Result

Click **OK** to run the FlexRay decode and see the result on the Waveform Window below.

10Mbps FlexRay Communication Data(RxD)



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

Channel

SYNC: A0 I/O 0: A2 Direction: SDI SDO

BCLK: A1

Color

Stream Data

Preamble: [Blue] Stream ID: [Orange]

Length: [Orange] Sample: [Orange]

Response (SDI)

Valid: [Orange] UnSol: [Orange]

Reserved: [Green] Response: [Orange]

Command (SDO)

Reserved: [Orange] CAd: [Orange]

NID: [Yellow] Verb ID: [Green]

Payload: [Cyan]

Range

Decode Range

From: Buffer Head To: Buffer Tail

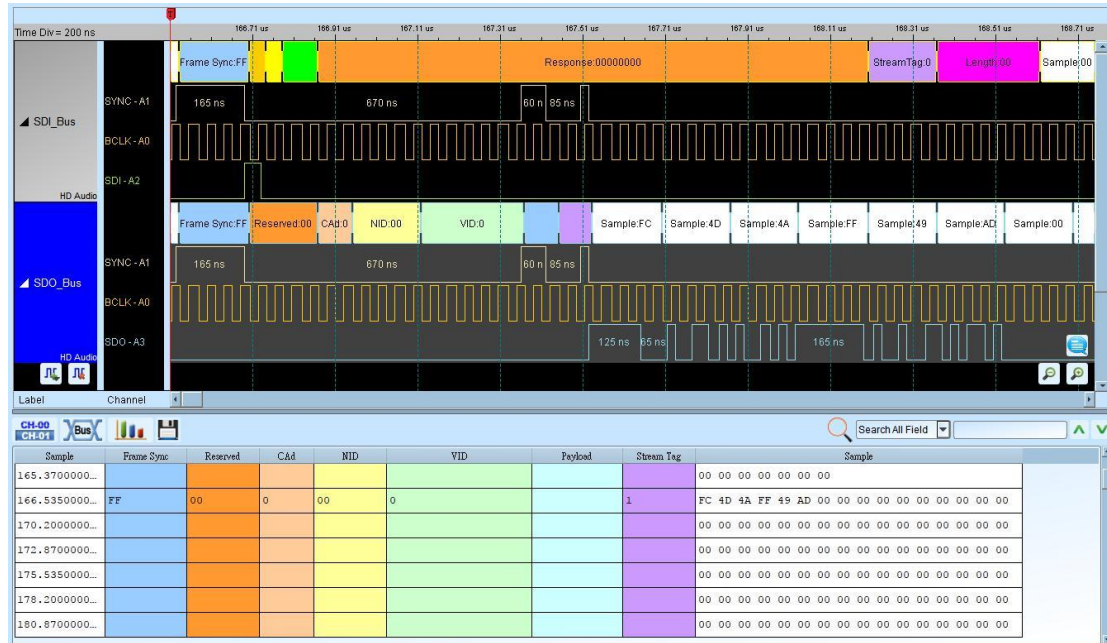
Buttons: Default, OK, Cancel

Channel: Show the selected channel (CH 0-CH3).

Direction: Show the data with SDI or SDO decoding.

Result

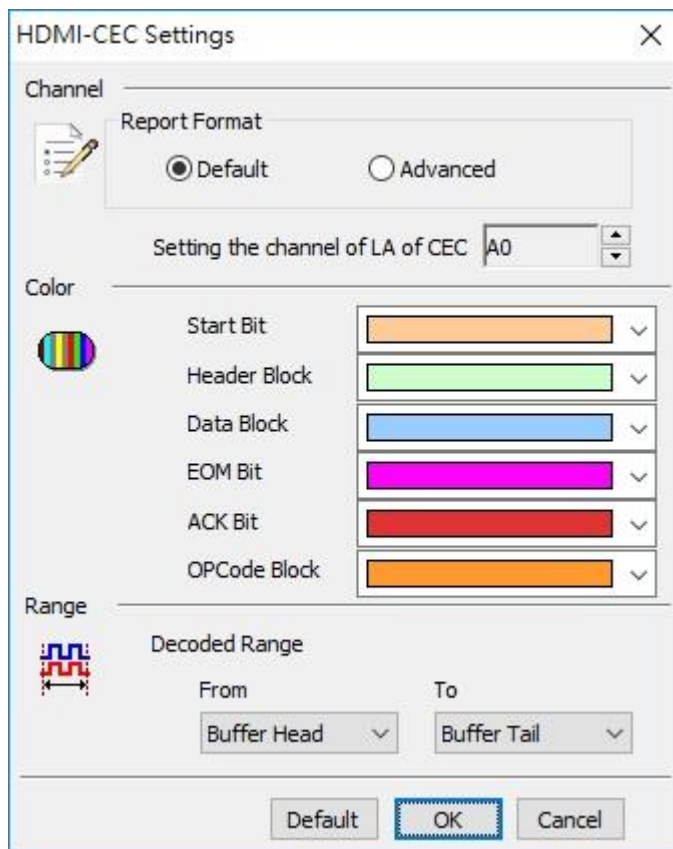
Click **OK** to run the HD Audio decode and see the result on the Waveform Window below.



HDMI-CEC

The HDMI-CEC bus is a one-wire, “party line” that connects up to ten (10) AV devices through standard HDMI cabling. The CEC protocol includes automatic mechanisms for physical address (topology) discovery, (product type based) logical addressing, arbitration, retransmission, broadcasting, and routing control. Message opcodes support both device specific (e.g. set-top-box, DTV, and player) and general features (e.g. for power, signal routing, remote control pass-through, and on-screen display).

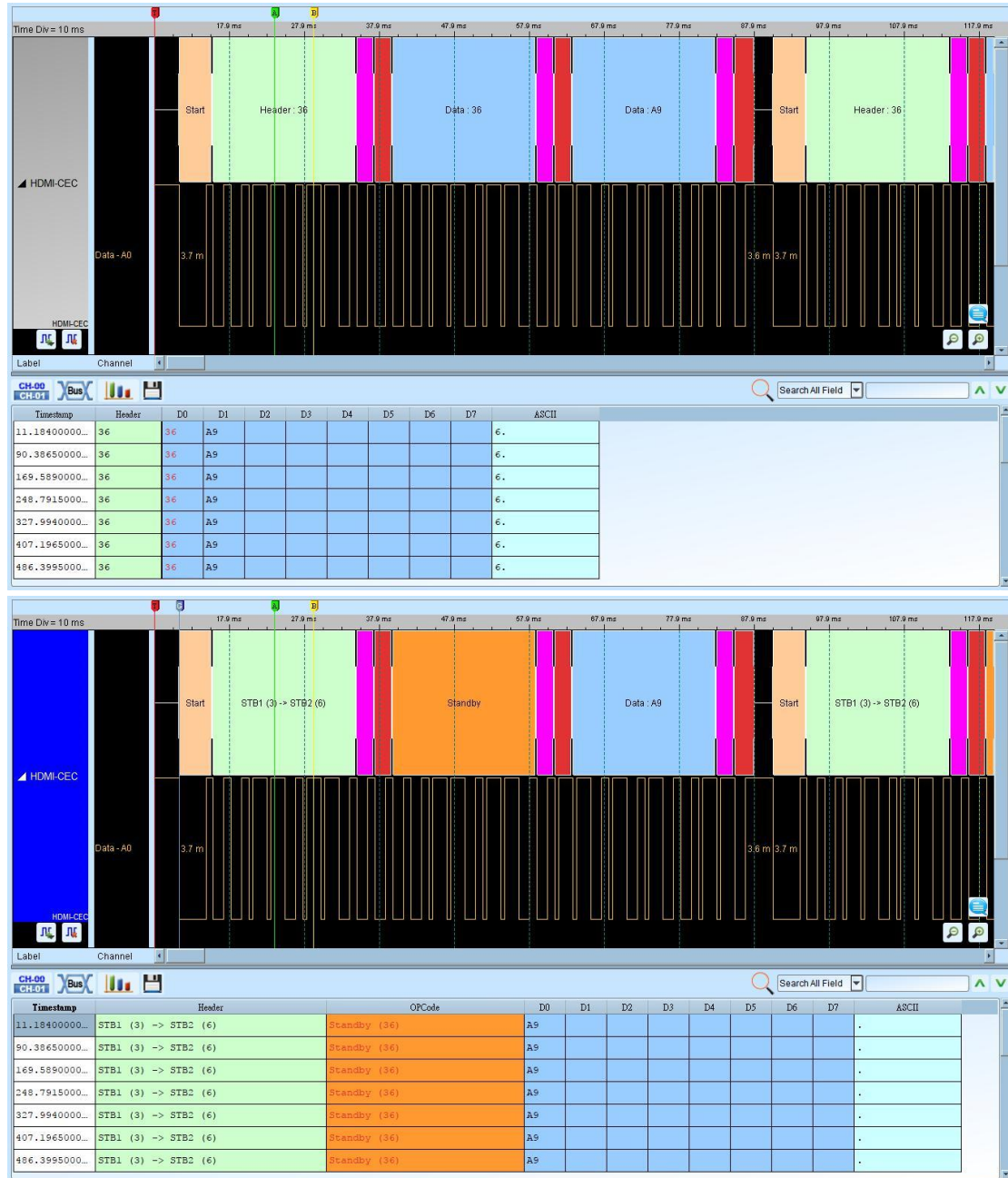
Settings



Latch data: Nominal (1.05 micro seconds) or User Define.

Result

Click **OK** to run the HDMI-CEC decode and see the result on the Waveform Window below.



HDMI-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

Channel: Display the channels (CH0 and CH1).

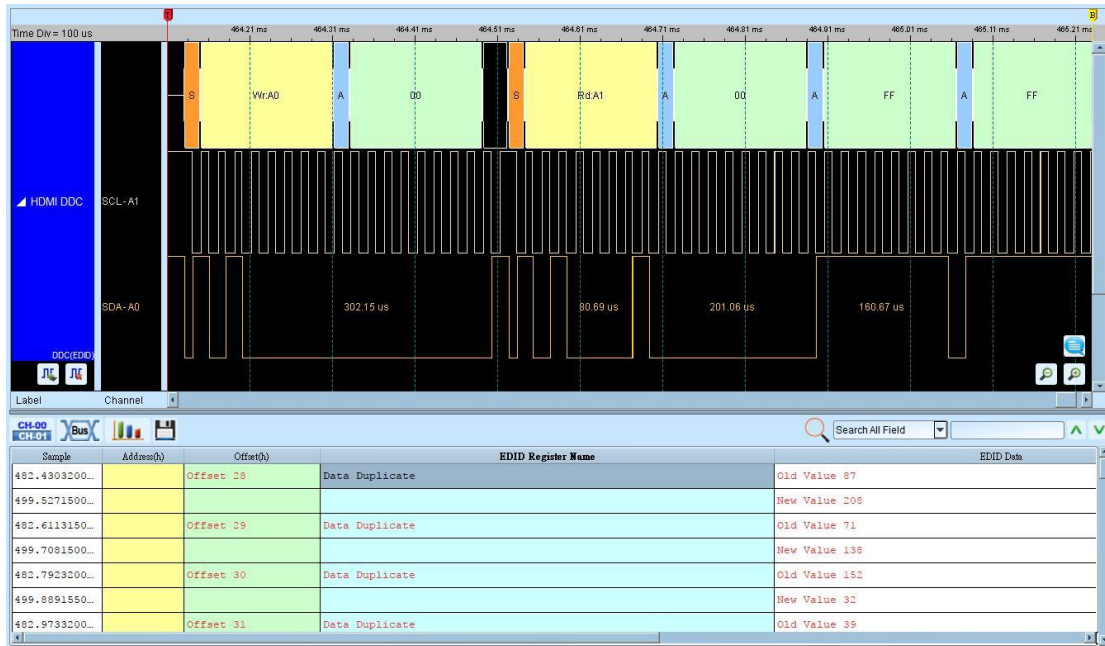
7-bit addressing: Show 7-bit addressing

7-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.

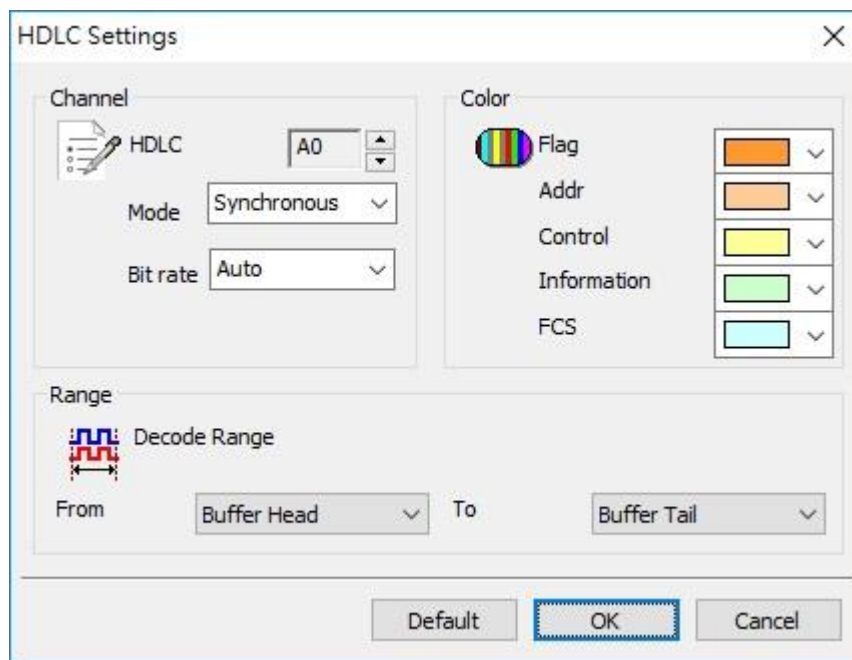
Result



HDLC

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

Settings

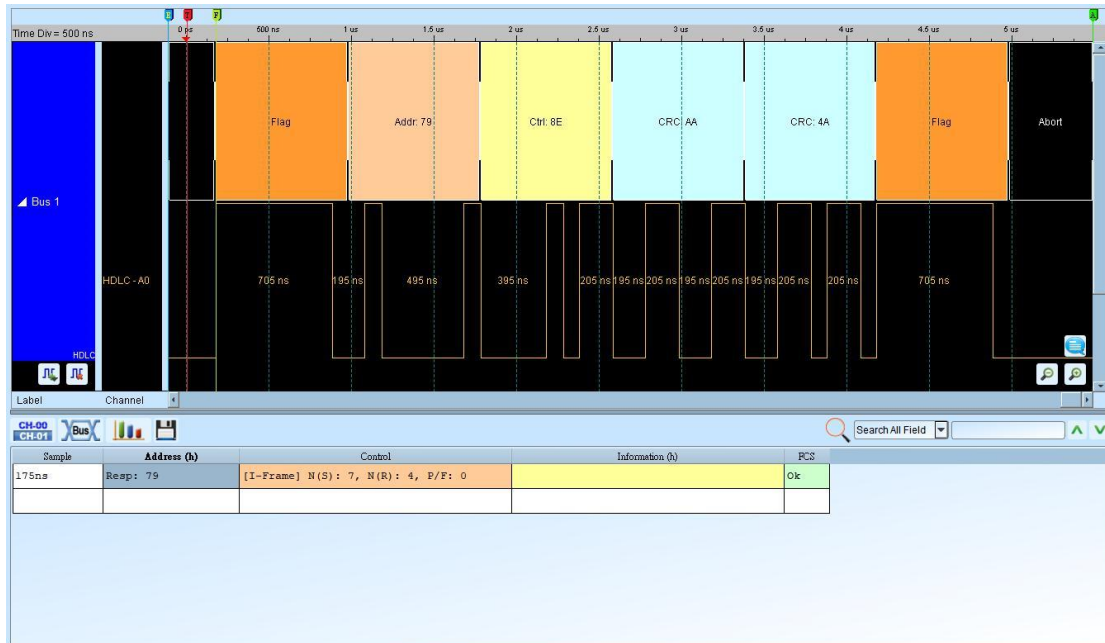


HDLC: Set the channel of the signal.

Mode: Synchronous or Asynchronous mode.

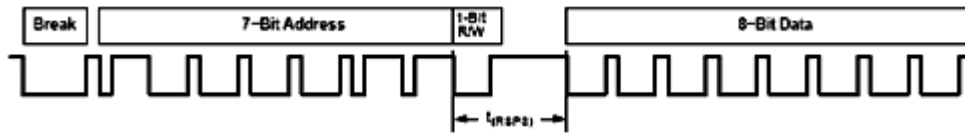
Bit rate: Set the specific data rate or auto detection.

Result

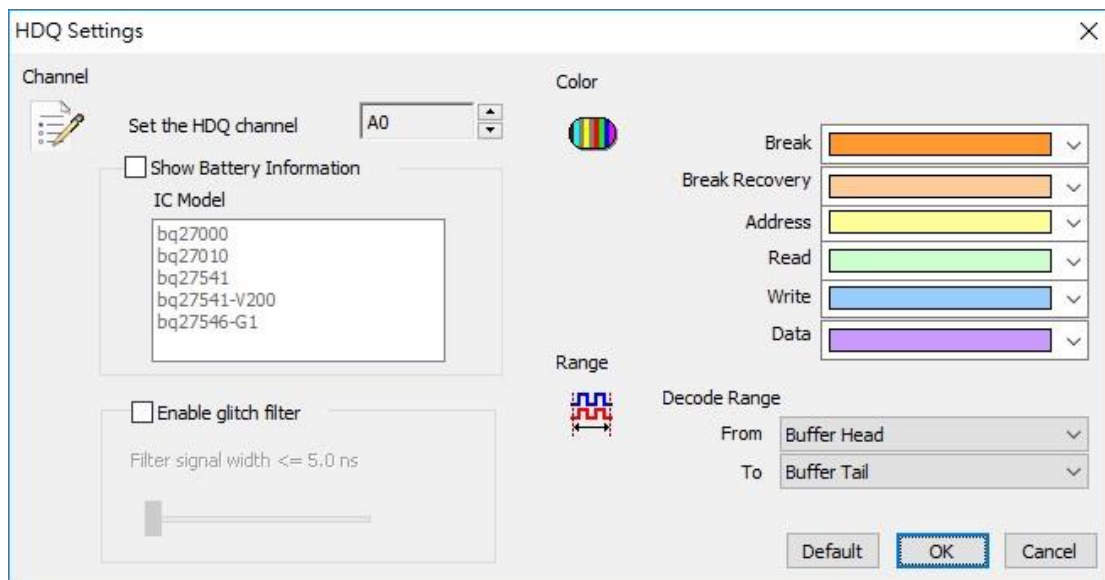


HDQ

The HDQ bus has two kinds of formats: 8 bits or 16 bits signals as the diagram below.



Settings

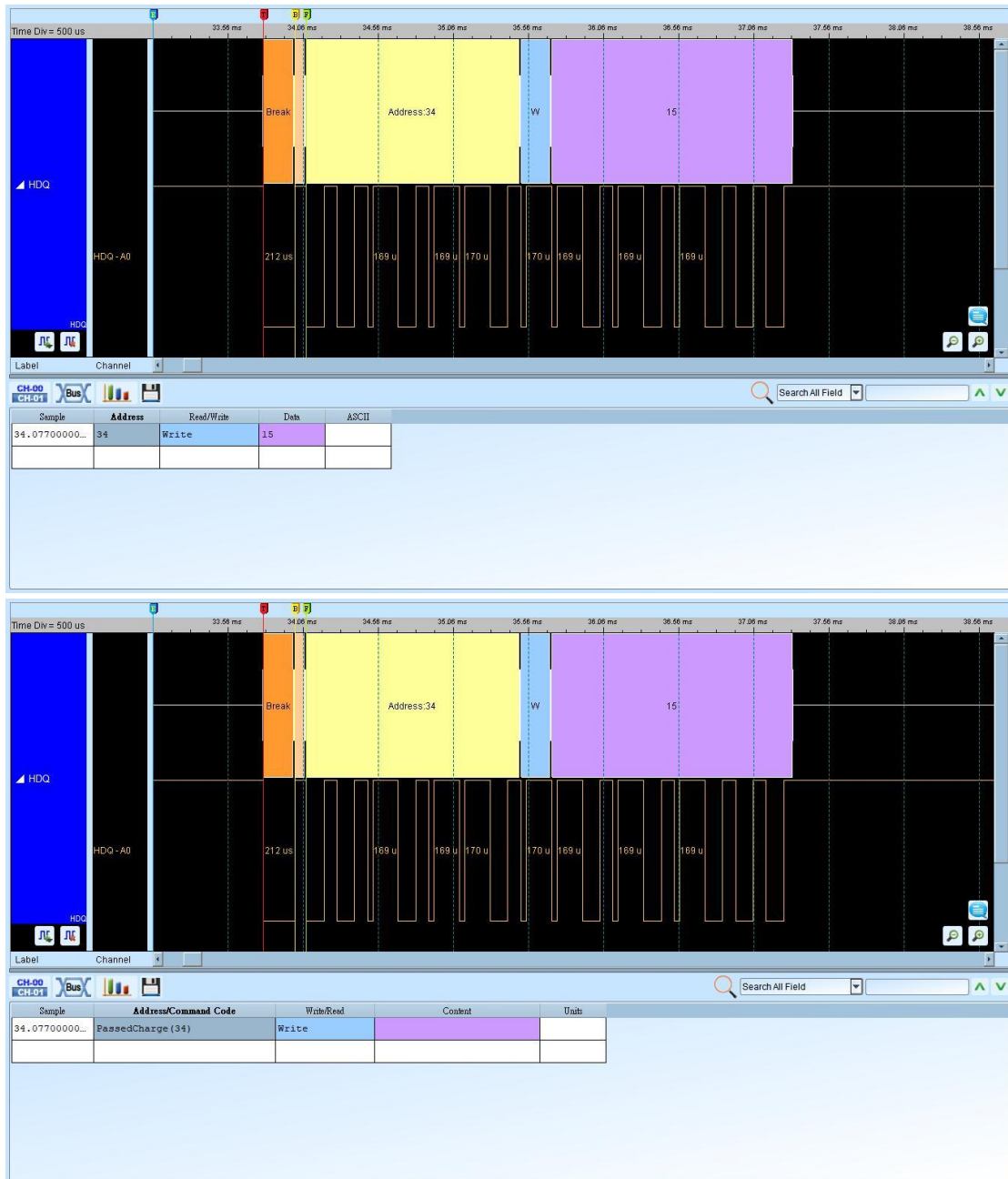


Channel: Set the HDQ channel: Show the selected channel (CH 0).

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

Result

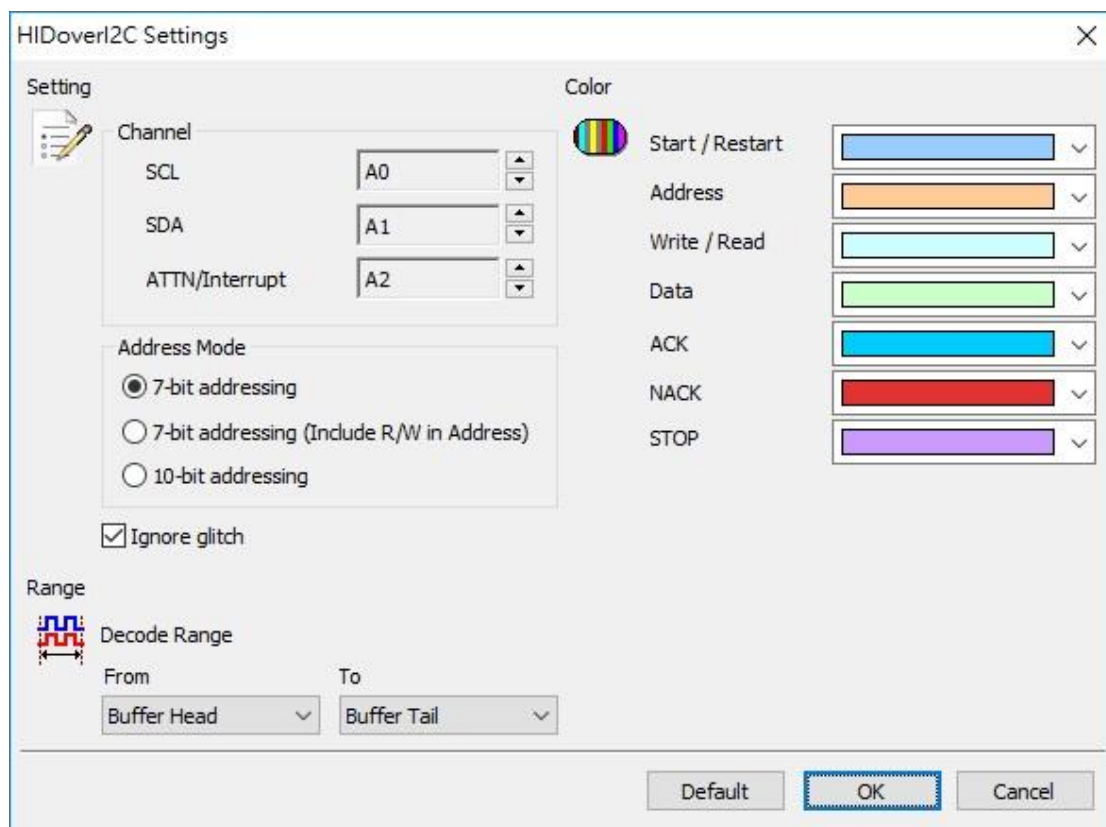
Click **OK** to run the HDQ decode and see the result on the Waveform Window below.



HID Over I²C

HID Over I2C (Human Interface Device Over I2C) protocol is established by Microsoft. It's applied for Windows 8 ARM platform.

Settings



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

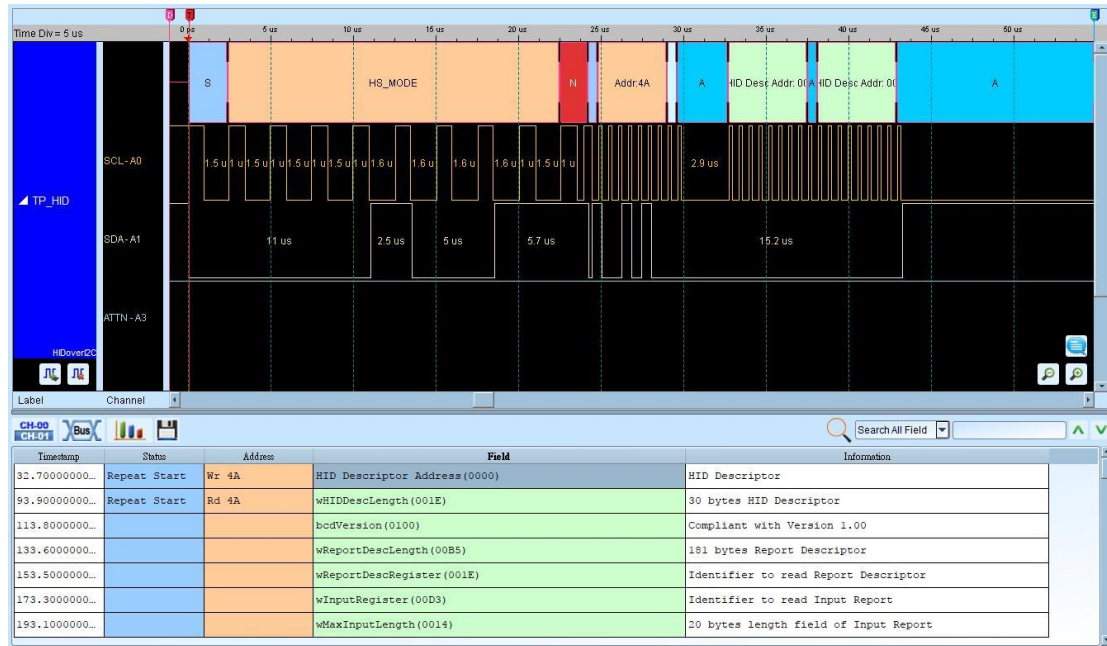
7-bit addressing: Show 7-bit addressing.

7-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.

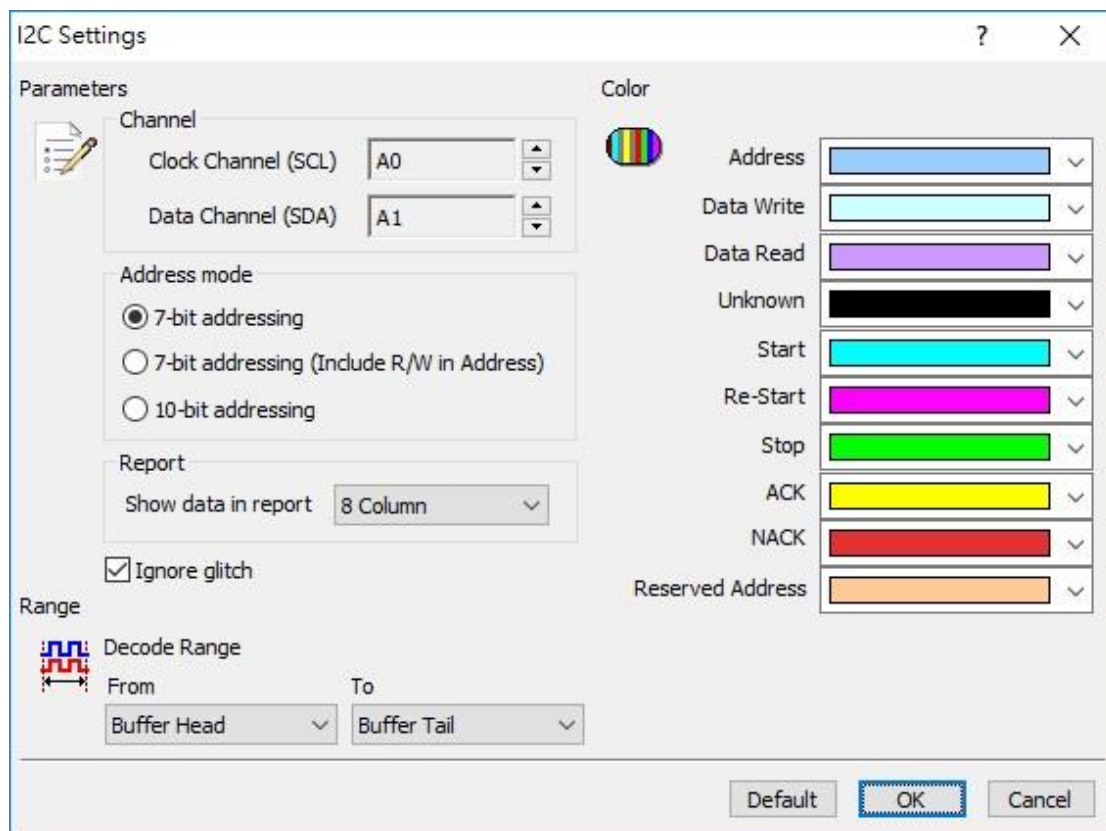
Result



I²C

The Inter-Integrated Circuit (I²C) bus has two data bits: Serial Data (SDA) and Serial Clock (SCL).

Settings



Channel: Display the channels (CH0 and CH1).

Address Mode: Select the 7-bit or 10-bit address.

7-bit addressing: Show 7-bit addressing.

7-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.

Report: Show either 8 or 16-columns data in the report window.

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

Result

Click **OK** to run the I²C decode and see the result on the Waveform Window below.

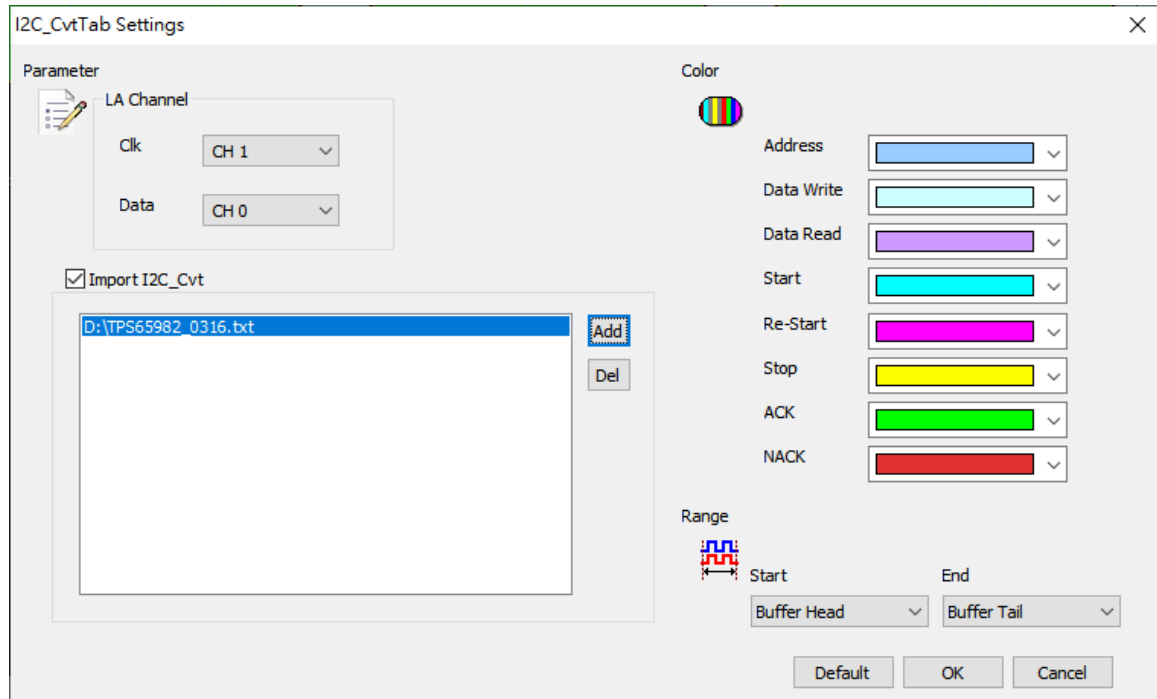
It shows frequency in the Information field.



I²C User Defined

Custom the command sets of I2C, and decode the data.

Parameter



CLK: I²C Clock °

Data: I²C Data °

Import I2C_CVT: Import I2C user defined table

Import Rules:

I. Basic syntax

A. ##: Notations

B. #: Add parsing according to the description content

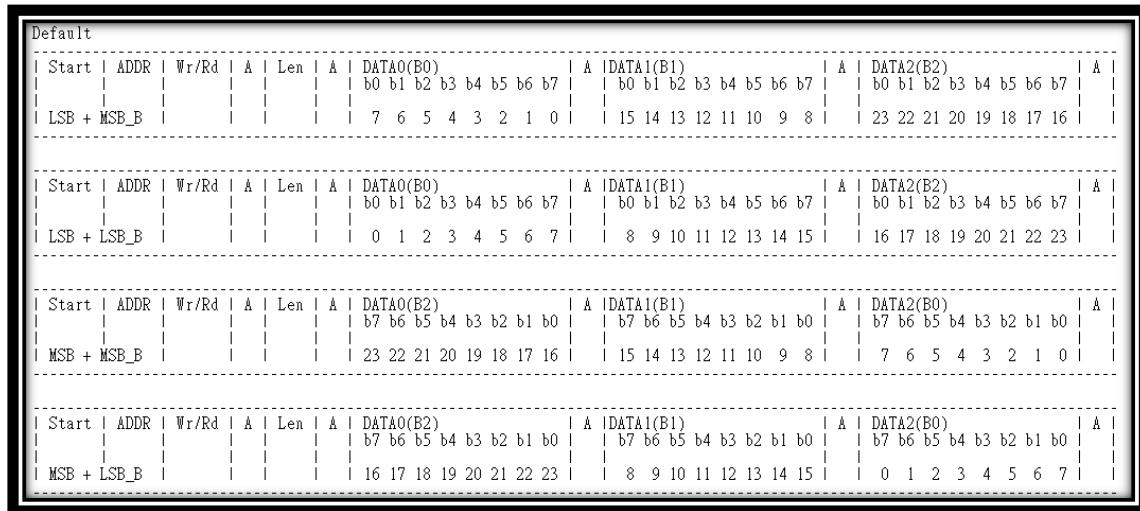


Figure 1. I2C frame structure

II. Detailed settings (take TI TPS65982 as an example)

A. #MSB, #LSB

Please refer to the structure of Figure 1.

Example: #MSB

B. #DAT, #DAT_C, #DAT_V

1st: Register Header(Hex)

2nd: Register Name

3rd: Access (W/R/RW)

Register Number	Register Name	Access	No. Data Bytes
0x00	VID	RO	4
0x01	DID	RO	4
0x02	ProtoVer	RO	4
0x03	Mode	RO	4CC
0x04	Type	RO	4CC

4th: Byte count

If `_C` is added behind, ASCII is displayed, and there is without bit detail. If `_T` is added to the back, Text is displayed without bit detail.

Example: `#DAT: 00h, VID, R, 4`

C. `#BIT_V`, `#BIT_T`

Table 3-46. 0x50 Data Control Register Bit Field Definitions

Bits	Name	Description
Bytes 1-4: Data Control (treated as 32-bit little endian value)		
31:8	Reserved	Reserved (write 0).
7:4	StatusNakReason	Reserved (Host may write any value to this field, no action to be taken).
3	StatusNak	Reserved (Host may write 0 or 1, no action to be taken).
2	InterruptAck	When set, causes IntMask1 value to be written to IntClear1 (clearing all interrupt events).
1	SoftReset	When set, causes a soft-reset of PD Controller. Equivalent to Gaid 4CC.
0	HostConnected	0b No TBT host connected. NOTE: The Thunderbolt Controller will also set bit 1 (SoftReset) when transitioning this bit from 1->0 to force a port disconnect/reconnect.
		1b TBT host connected.

1st: Byte position

2nd: Byte count (If Byte count = -4, registers use four character codes (4CC))

3rd: Bit position

4th: Description

5th: little/Big/None endian value (L/B/N)

If `_V` is added to the back, the value is displayed without bit detail.

If `_T` is added later, it will be parsed with the next `#BIT_T` definition.

Example1: `#BIT_V: 1, 3, 1, SoftReset, L`

Example2: `#BIT_T: 1, 3, 0, HostConnected, L`

#BIT_S: 0b , No TBT host connected.

#BIT_S: 1b , TBT host connected.

D. #BIT_T

Table 3-46. 0x50 Data Control Register Bit Field Definitions

Bits	Name	Description				
Bytes 1-4: Data Control (treated as 32-bit little endian value)						
31:8	Reserved	Reserved (write 0).				
7:4	StatusNakReason	Reserved (Host may write any value to this field, no action to be taken).				
3	StatusNak	Reserved (Host may write 0 or 1, no action to be taken).				
2	InterruptAck	When set, causes IntMask1 value to be written to IntClear1 (clearing all interrupt events).				
1	SoftReset	When set, causes a soft-reset of PD Controller. Equivalent to Gaid 4CC.				
0	HostConnected	<table border="1"> <tr> <td>0b</td> <td>No TBT host connected. NOTE: The Thunderbolt Controller will also set bit 1 (SoftReset) when transitioning this bit from 1->0 to force a port disconnect/reconnect.</td> </tr> <tr> <td>1b</td> <td>TBT host connected.</td> </tr> </table>	0b	No TBT host connected. NOTE: The Thunderbolt Controller will also set bit 1 (SoftReset) when transitioning this bit from 1->0 to force a port disconnect/reconnect.	1b	TBT host connected.
0b	No TBT host connected. NOTE: The Thunderbolt Controller will also set bit 1 (SoftReset) when transitioning this bit from 1->0 to force a port disconnect/reconnect.					
1b	TBT host connected.					

1st: Bit of value

2nd: Description

Analyze the value according to the bit position defined by #BIT_T, and

correspond to its description

Example: #BIT_S: 0b , No TBT host connected.

#BIT_S: 1b , TBT host connected.

III. Example

#LSB:

#DAT: 04h, Type, R, -4

#DAT: 50h, Data Control, RW, 4

#BIT_V: 1, 3, 31:8, Reserved, L

#BIT_V: 1, 3, 7:4, StatusNakReason, L

#BIT_V: 1, 3, 3, StatusNak, L

#BIT_V: 1, 3, 2, InterruptAck, L

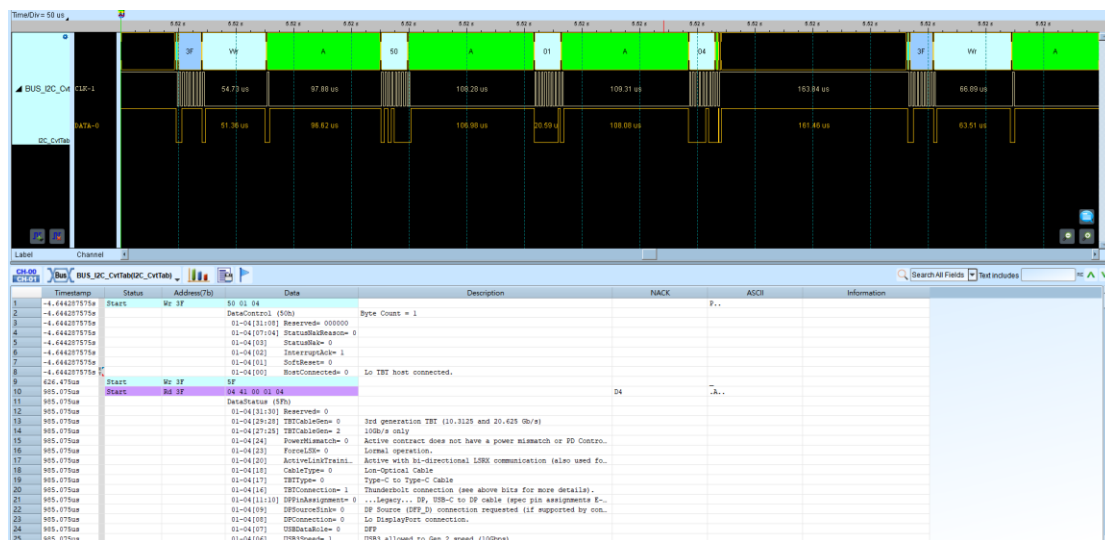
#BIT_V: 1, 3, 1, SoftReset, L

#BIT_T: 1, 3, 0, HostConnected, L

#BIT_S: 0b , No TBT host connected.

#BIT_S: 1b , TBT host connected.

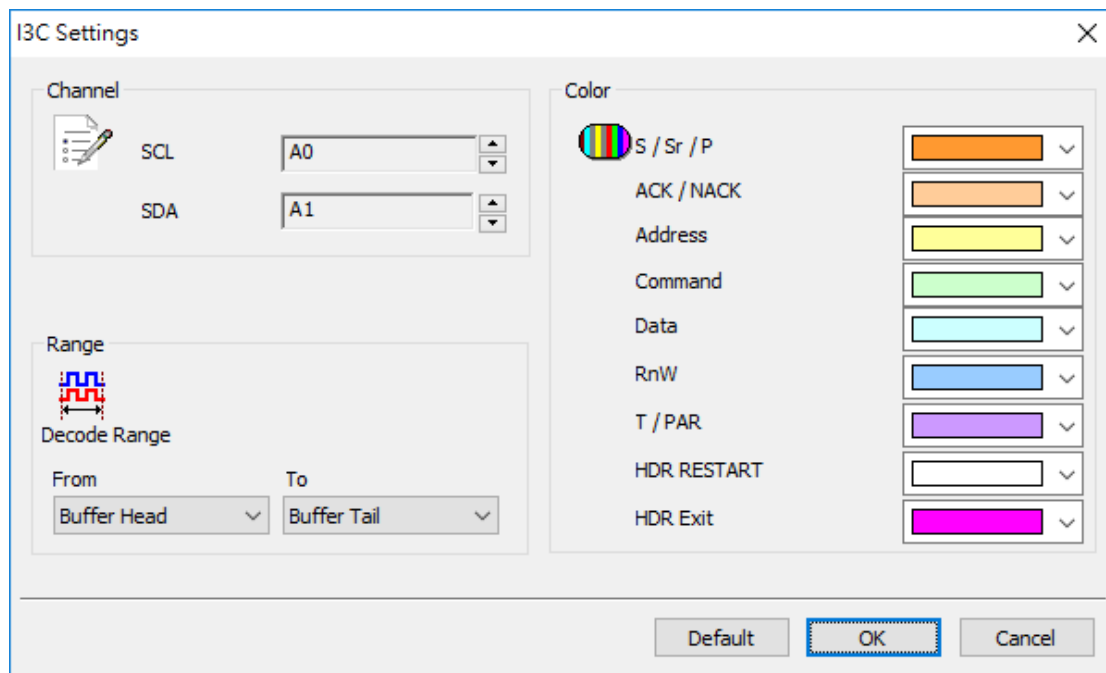
Result



I³C

I³C is a bus interface for connecting sensors to an application processor. It is a core sensor integration technology that can combine multiple sensors from different vendors in a device to streamline integration and improve cost efficiencies. It gives developers unprecedented opportunity to craft innovative designs for any mobile product, from smartphones, to wearables, to safety systems in automobiles.

Settings

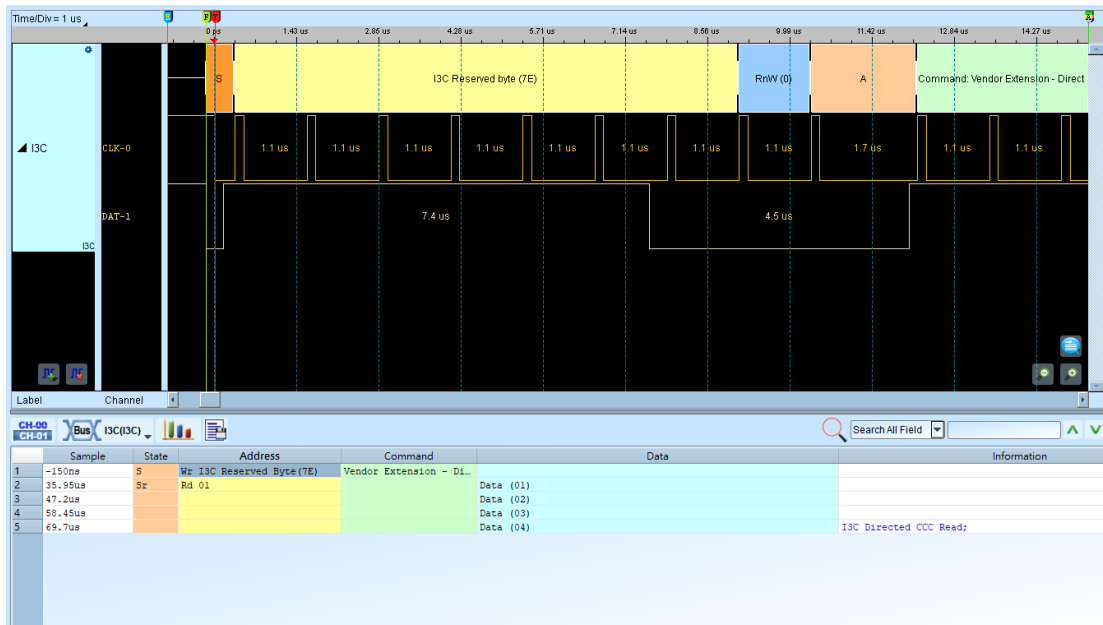


Channel: Display the channels (CH0 and CH1).

Result

Click **OK** to run the I³C decode and see the result on the Waveform Window below.

It shows frequency in the Information field.



I2C EEPROM

EEPROM can be erased and reprogrammed (written to) repeatedly through the application of higher than normal electrical voltage generated externally or internally in the case of modern EEPROMs.

Settings

Channel: Display the channels (CH0 and CH1)

Address: The default address width is 7.

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

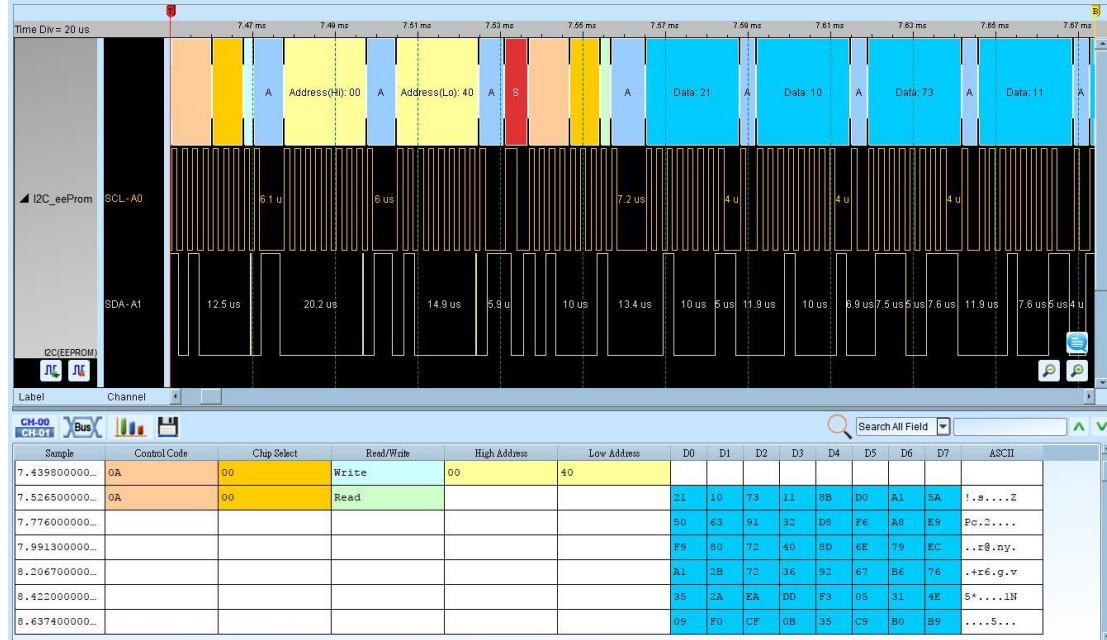
24LCS61/24LCS62: 24LCS61/24LCS62 EEPROM protocol.

Ignore glitch: Ignore the glitch when the slow transitions.

Result

Click OK to run the I²C EEPROM decode and see the result on the Waveform

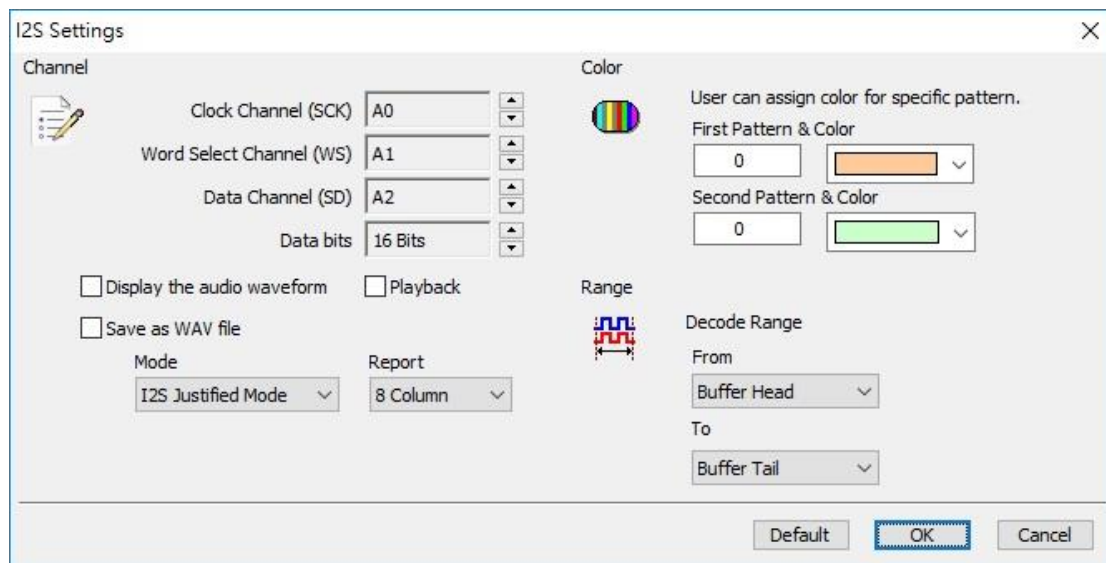
Window below.



I²S

The Inter-Integrated Circuit Sound (I²S) bus has three data bits: Serial Clock (SCK), Word Select Line (WS) and Multiplex Data Line (SD).

Settings



Channel: Show the selected channels (CH0, CH1 and CH2) and set Data bits (16 bits).

Display the audio waveform: Click to display the audio waveform in the Waveform Window.

Playback: Click to display the audio from the speaker..

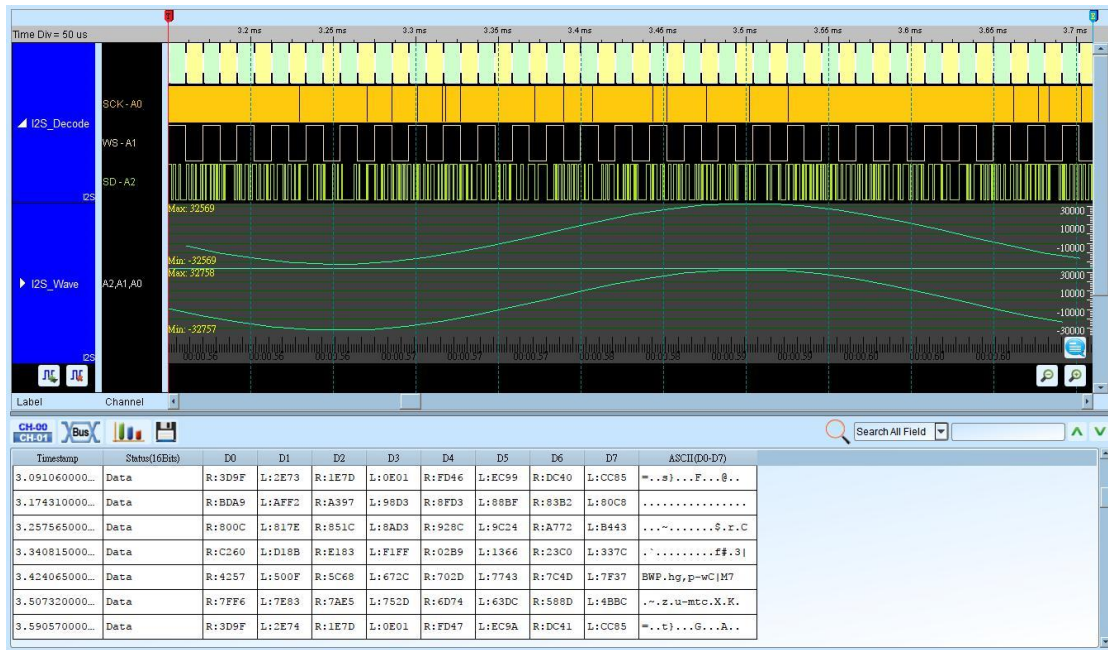
Save as WAV file: Click to save the data to a audio file(.wav) in the work directory.

Mode: I²S Justified/MSB Justified/LSB Justified/PCM/TDM.

Report: Select the column number.

Result

Click **OK** to run I²S decode and see the result on the Waveform Window below.



I80

The I80 controls 3 or 4-pins (WR, RD, CS, D/C) data.

Settings

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

Click **OK** to run the I80 decode and see the result on the Waveform Window below.



IDE

IDE (Integrated Device Electronics) is a computer hardware bus and has the following data bits:

General Channel (11 pins): DASP-, DIOR-:HDMARDY-:HSTROBE, DIOW-:STOP, DMACK-, DMARQ, INTRQ, IORDY:DDMARDY-:DSTROBE, PDIAG-:CBLID-, RESET-, CSEL, IOCS16-.

Register Channel (5 pins): CS(0:1)-, DA(2:0).

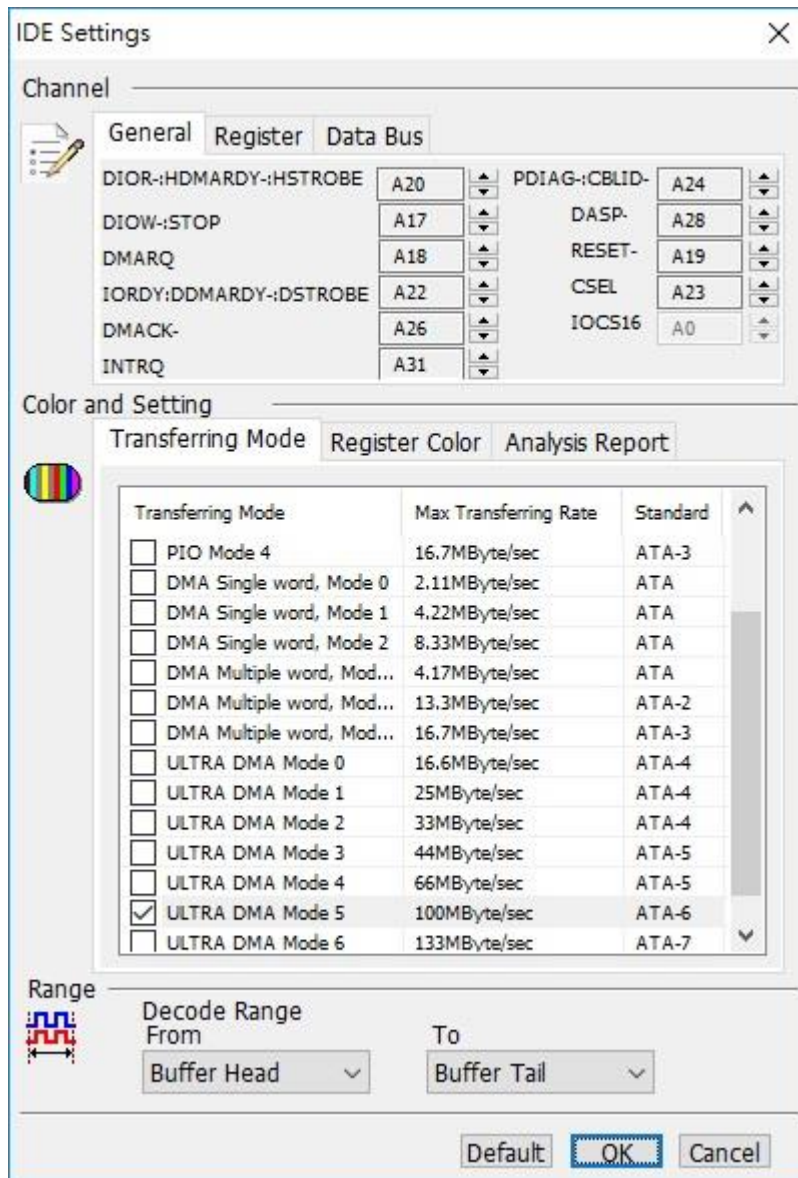
Data Bus (16 pins): DD(15:0)

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-:HDMAR DY- :HSTROBE	Device I/O read: Ultra DMA ready: Ultra DMA data strobe	Channel 19
Pin26	Ground		
Pin27	IORDY:DDMAR DY- :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



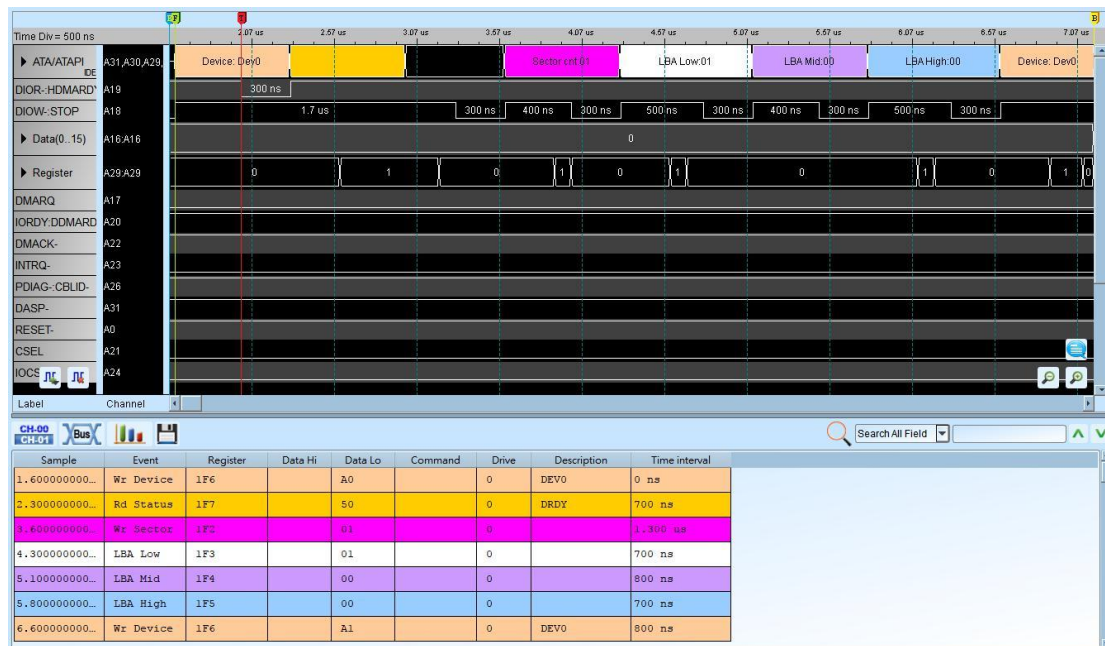
Channel: Set channel number for General, Register, and Data Bus.

Transferring Mode: Select the target system.

Analysis Report: Filter the data in the Report Window.

Result

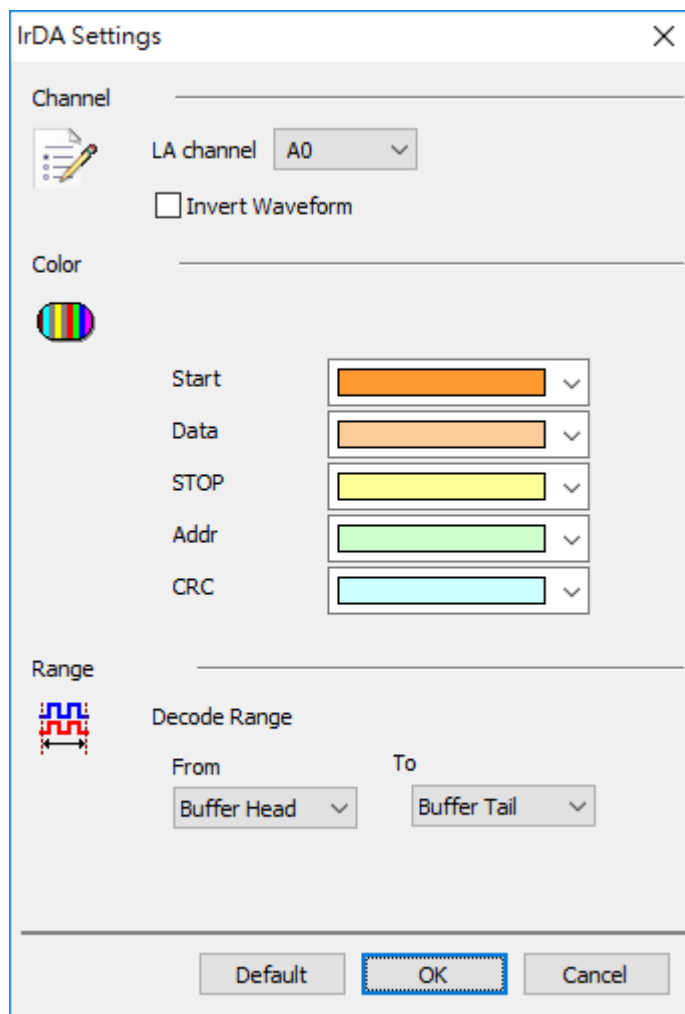
Click **OK** to run the IDE decode and see the result on the Waveform Window below.



IrDA

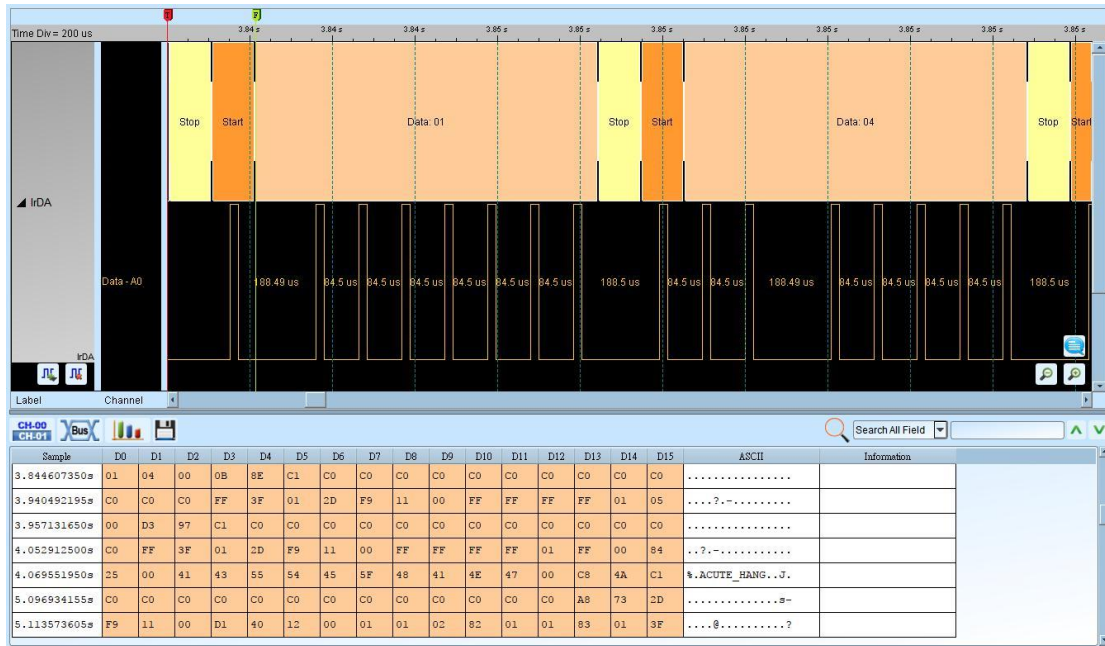
The Infrared Data Association (IrDA) was formed in 1993. The IrDA is point to point user model for a wide range of appliances and devices.

Settings



Channel: Show the selected channel.

Result



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

ITU656(CCIR656) Settings

Channel

Channel			Data Bits
Clk	A0	Data 5	8 Bits
Data 0	A1	Data 6	
Data 1	A2	Data 7	Channel number
Data 2	A3	Data 8	1
Data 3	A4	Data 9	
Data 4	A5	Data 10	

Color

SAV	CR
EAV	CB
Blanking	Y

Range

Decode Range

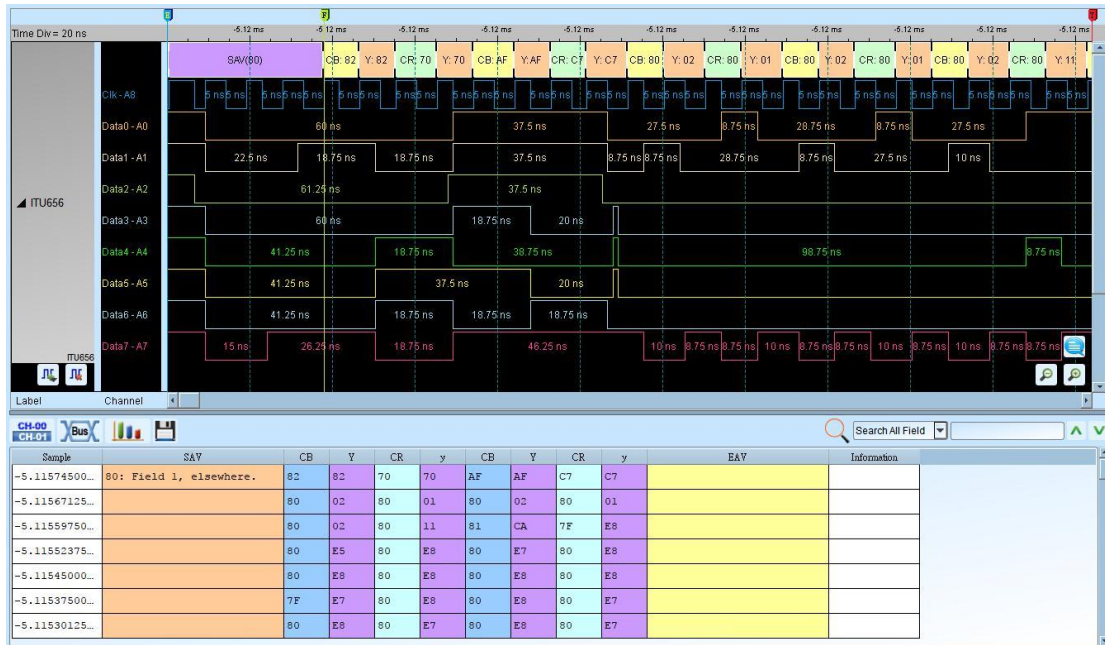
From	To
Buffer Head	Buffer Tail

Default OK Cancel

Channel: Show the selected channel (Clk, Data 0 – Data 9).

Data Bits: Show the number of data bits.

Result



JTAG

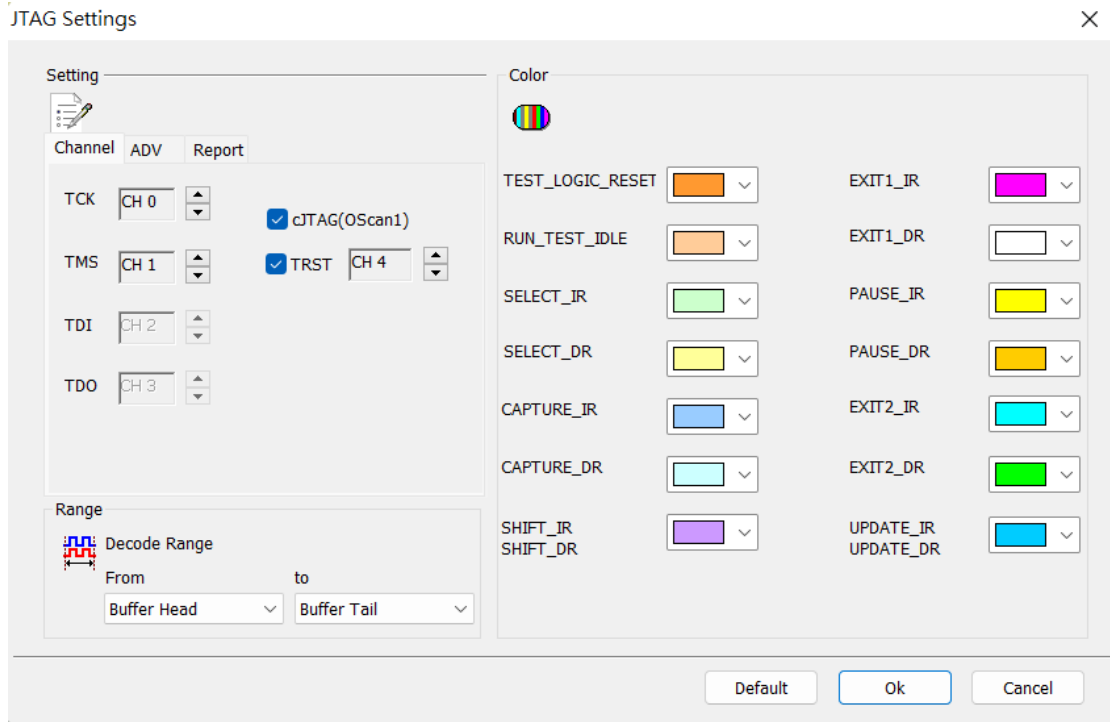
Joint Test Action Group (JTAG) is the common name used for the IEEE 1149.1 standard entitled Standard Test Access Port and Boundary-Scan Architecture for test access ports used for testing printed circuit boards using boundary scan.

A JTAG interface is a special four/five-pins interface added to a chip, designed so that multiple chips on a board can have their JTAG lines daisy-chained together if specific conditions are met, and a test probe need only connect to a single "JTAG port" to have access to all chips on a circuit board. When cJTAG(Oscan1) is active, the The protocol will transform to Advanced Protocol, The Advanced Protocol transfers information using only the TCK(C) and TMS(C) signals, serialized TDI, TMS, TDO, and control information for one TAPC state, causes the advance of the TAPC state.

Settings

The screenshot shows the 'JTAG Settings' dialog box. It is titled 'JTAG Settings' and has a close button (X) in the top right corner. The dialog is divided into several sections:

- Setting**: Contains a 'Channel' tab with 'ADV' and 'Report' sub-tabs. Under 'Channel', there are dropdown menus for TCK (CH 0), TMS (CH 1), TDI (CH 2), and TDO (CH 3). There is a checkbox for 'cJTAG(Oscan1)' which is currently unchecked, and a checked checkbox for 'TRST' with a dropdown menu set to 'CH 4'. Below this is a 'Range' section with a 'Decode Range' icon and 'From' and 'to' dropdown menus, both currently set to 'Buffer Head' and 'Buffer Tail' respectively.
- Color**: Contains a color selection icon and a list of JTAG signals with color pickers: TEST_LOGIC_RESET (orange), RUN_TEST_IDLE (light orange), SELECT_IR (light green), SELECT_DR (yellow), CAPTURE_IR (blue), CAPTURE_DR (light blue), SHIFT_IR (purple), SHIFT_DR (purple), EXIT1_IR (magenta), EXIT1_DR (white), PAUSE_IR (yellow), PAUSE_DR (orange), EXIT2_IR (cyan), EXIT2_DR (green), UPDATE_IR (cyan), and UPDATE_DR (cyan).
- Buttons**: 'Default', 'Ok', and 'Cancel' buttons are located at the bottom right of the dialog.

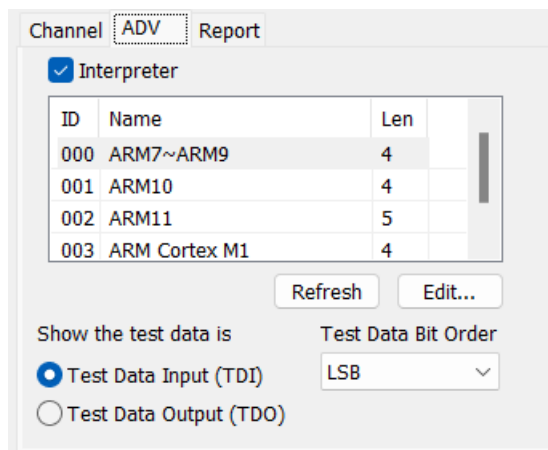


Settings: Includes Channel, Setting, and Report.

Channel: Set the channel number.

Show the test data is: The data is TDI or TDO.

Test Data Bit Order: LSB First or MSB First.

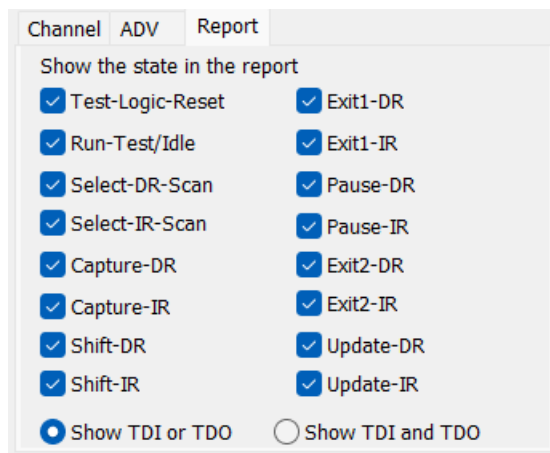


Interpreter instruction: Check the Interpreter Instruction and you will see a list of commands. The JTAG decode will display its updated commands in the Instruction Register, which is the temporary memory buffer for the commands. Click Edit to edit the commands in the Interpreter Instructions; then click Refresh to update the

commands in the Interpreter Instructions.

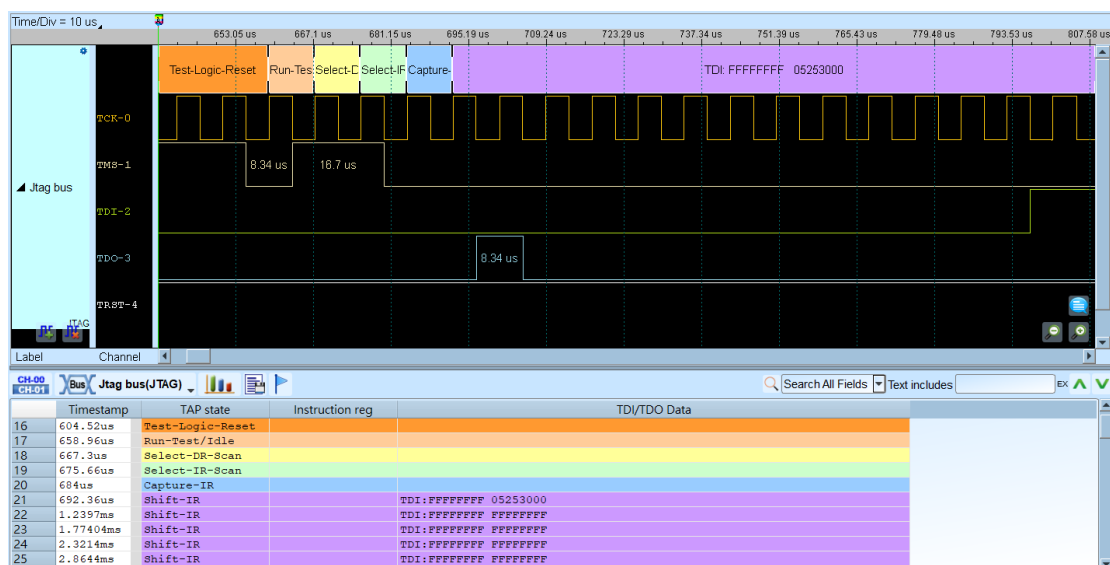
Acute JTAG Instruction table (JtagInst.txt): This file is supported by the JTAG DLL and can be modified if needed. The JTAG Decode also supports the BSDL format. You can load the BSDL file in order to save time on editing commands. If you are interested in more details, please refer to the Acute JTAG Instruction table Syntax Description in the end of the JTAG Decode chapter.

Report: You can filter the data you want to see on the Report Window.



Show TDI or/and TDO: Show either TDI or TDO or both.

Result



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

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

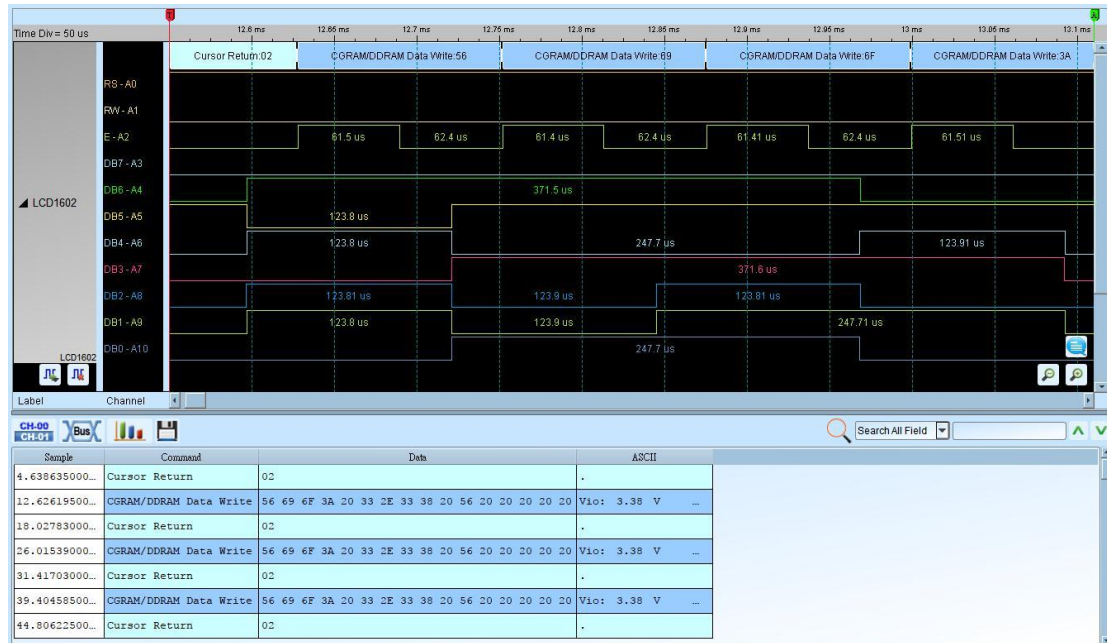
Channel: Show the selected channels (RS: CH0, RW: CH1, ..., DB0: CH10).

Data Mode: 8 lines or 4 lines.

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

Result

Click **OK** to run the LCD1602 decode and see the result on the Waveform Window below.



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

Channel: Show the selected channels.

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

Chip setting: Custom, TM1814, WS2811

Condition: Idle high

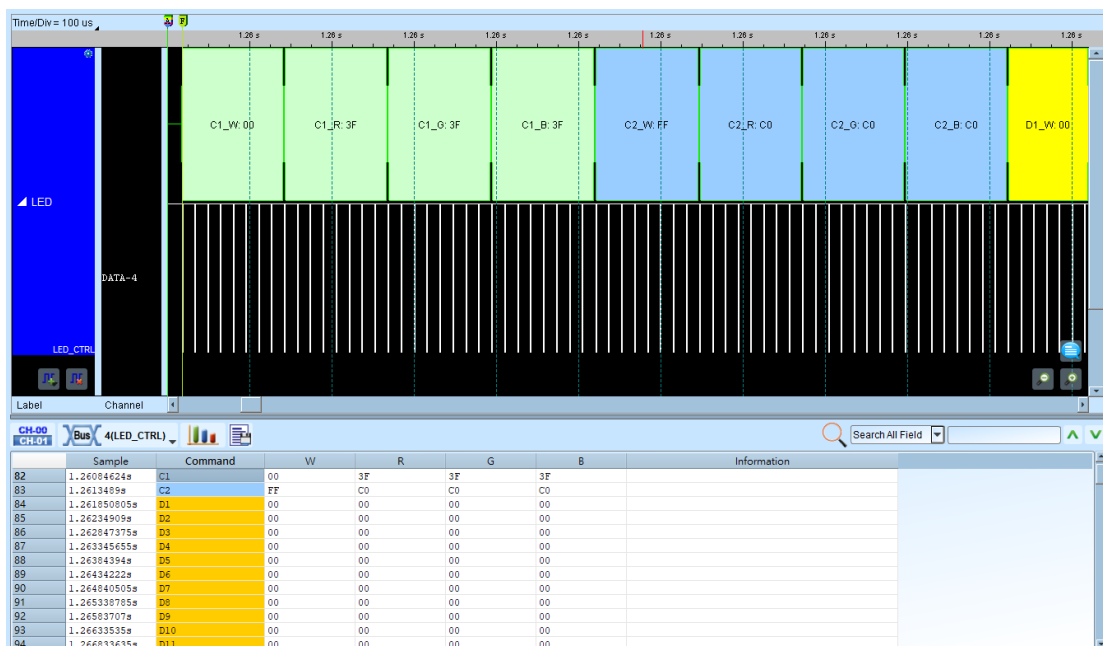
If the time that waveform goes to low between T0 min and T0 max, the bit will decode as zero. If it's between T1 min and T1 max, the bit will decode as one.

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

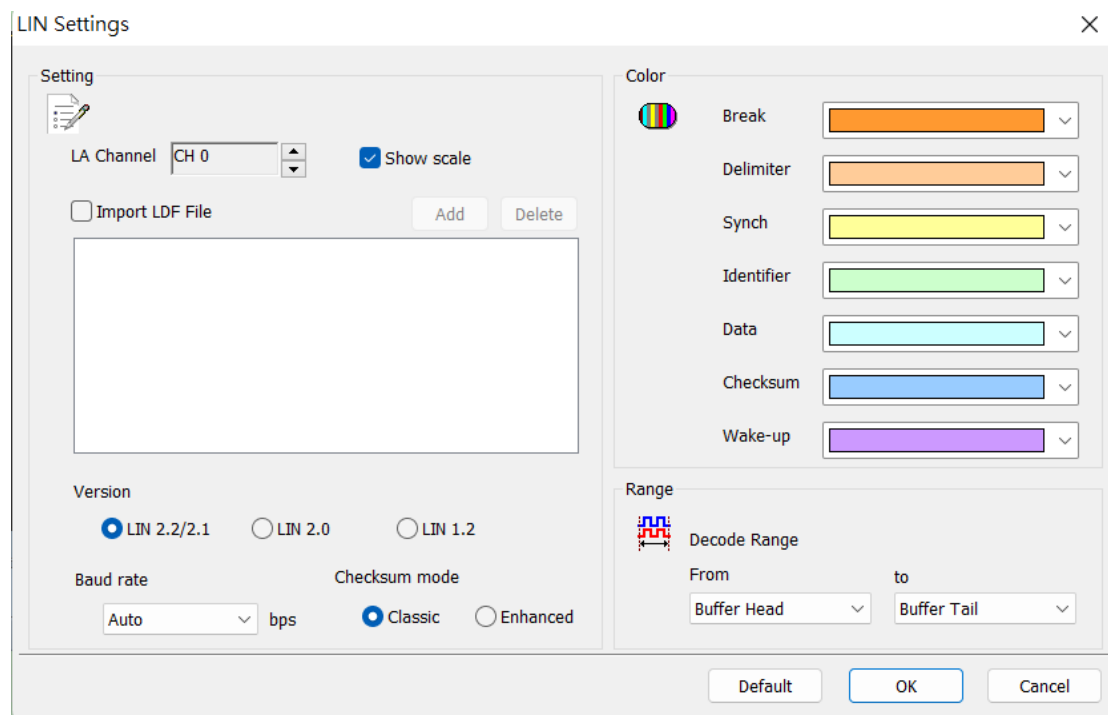
Click **OK** to run the LED_Ctrl decode and see the result on the Waveform Window below.



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



LA Channel: Show the selected channel (CH0 for LIN).

Show Scale: It will show the scales according to the bitwidth on the waveform if user selects Show Scale.

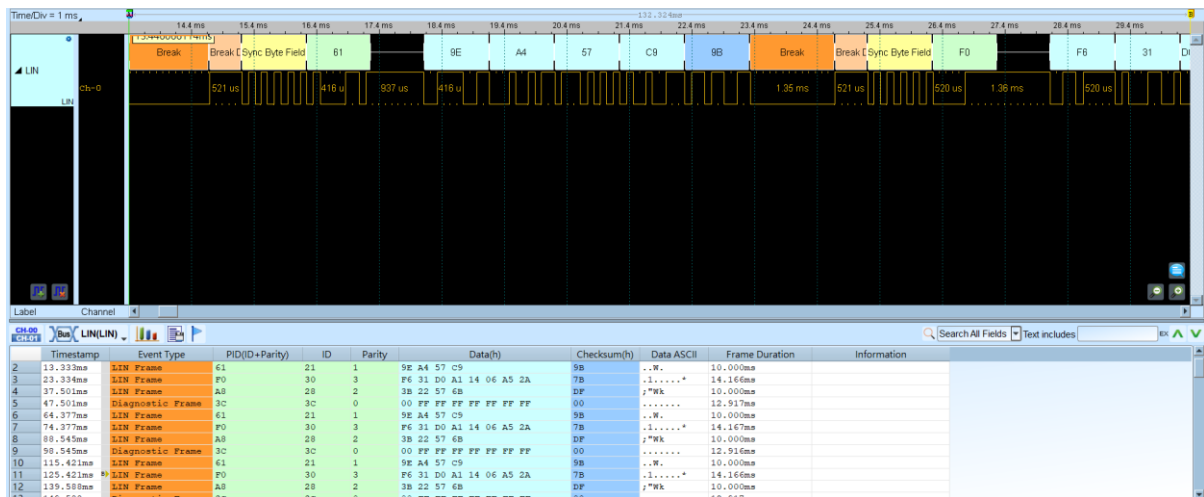
Import LDF File: Enable users to import the LIN Description File. Click on Add and select LDF file.

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



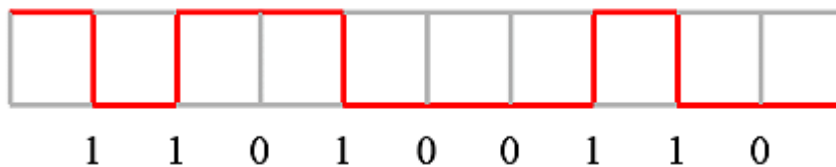
The decoder with LDF



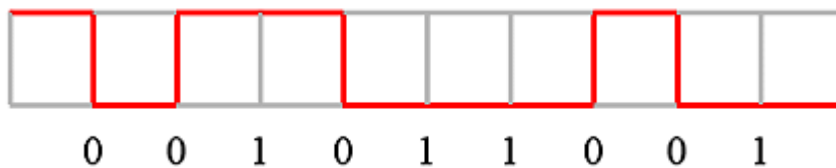
Line Decoding

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.

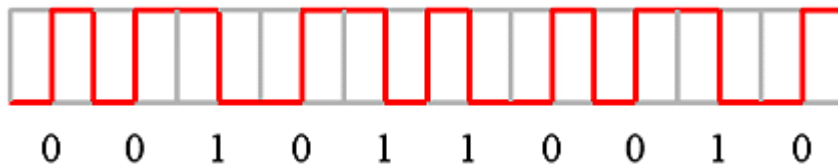


NRZI (Transition occurs for a zero): A 0 is represented by a transition of the physical level, a 1 has no transition.

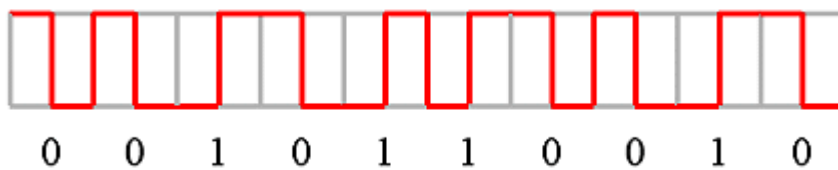


Manchester: In telecommunication, Manchester code is a line code in which the encoding of each data bit has at least one transition and occupies the same time. It is, therefore, self-clocking, which means that a clock signal can be recovered from the encoded data. There are three modes:

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.

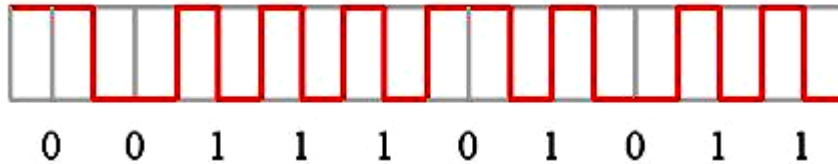


Differential Manchester: A 1 bit is indicated by making the first half of the signal equal to the last half of the previous bit's signal i.e. no transition at the start of the bit-time. A '0' bit is indicated by making the first half of the signal opposite to the last half of the previous bit's signal 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

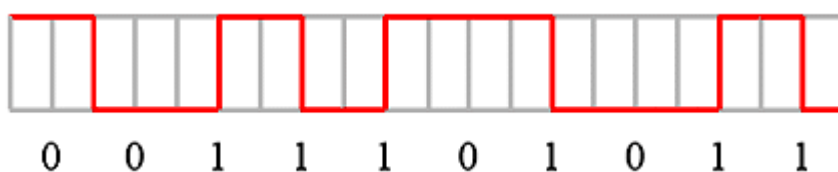
encoding, the symbol rate must be twice the bitrate of the original signal. Every bit of the original data is represented as two logical states that together form a bit.



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.



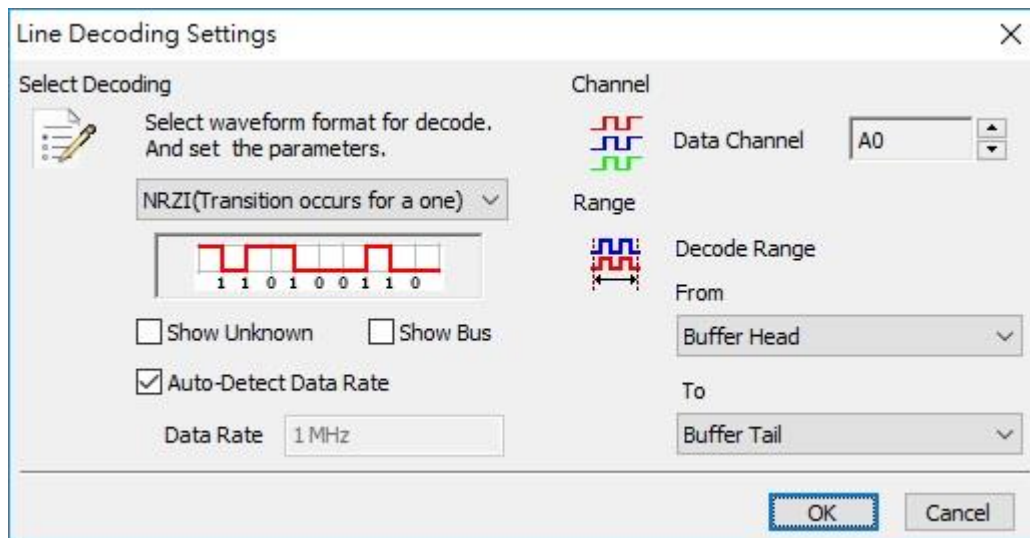
Modified Miller: The Modified Miller (M 2) demodulator facilitates demodulation of M 2 modulation data to NRZ-L (non-return-to-zero-level) data, composed of a data separation circuit for producing synchronizing clock pulses from the M 2 modulation data which is reproduced by a data recording device and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



Select Decoding: Select the line code you want to decode.

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.

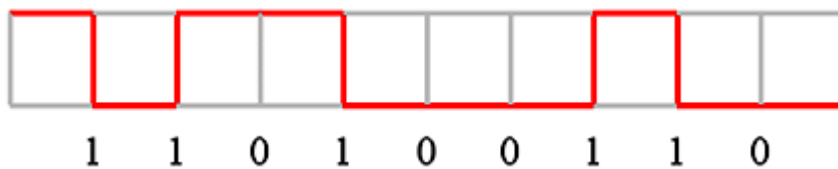
Channel: Show the selected channel (CH 0).

Range: Select the range within the waveform you want to decode.

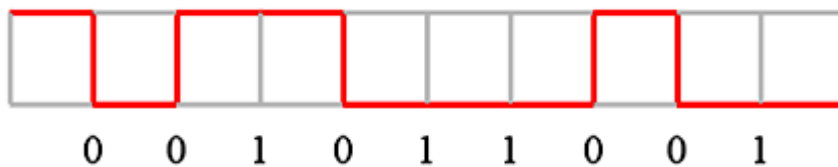
Line Encoding

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.

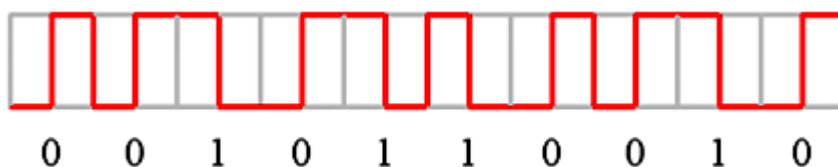


NRZI (Transition occurs for a zero): A 0 is represented by a transition of the physical level, a 1 has no transition.

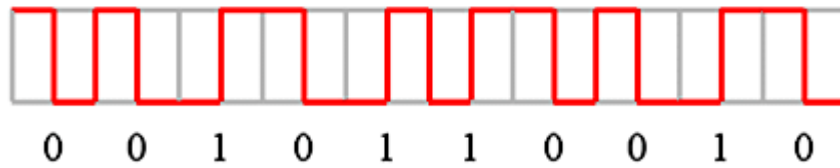


Manchester: In telecommunication, Manchester code is a line code in which the encoding of each data bit has at least one transition and occupies the same time. It is, therefore, self-clocking, which means that a clock signal can be recovered from the encoded data. There are three modes:

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



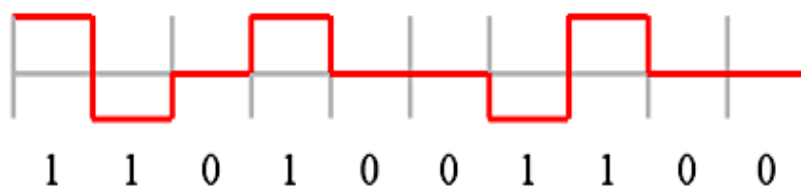
Manchester (IEEE802.3): A 1 is expressed by a low-to-high transition, a 0 by high-to-low transition.



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:

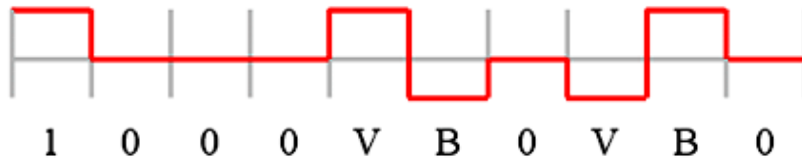
AMI (Standard): AMI (Alternate Mark Inversion) is a synchronous clock encoding technique that uses bipolar pulses to represent logical 1 value. It is therefore a three level system. A logical 0s is represented by no symbol, and a logical 1 is represented by alternating-polarity pulses.



AMI (B8ZS): Bipolar-8-Zero Substitution

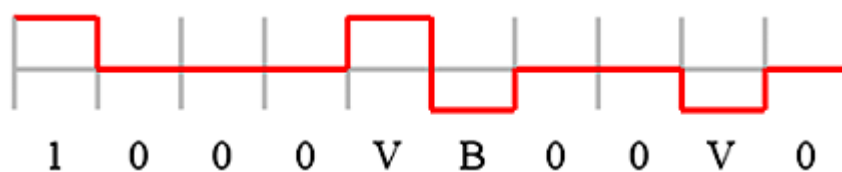
If 1 is +, 00000000 is represented to 000+-0-+

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.

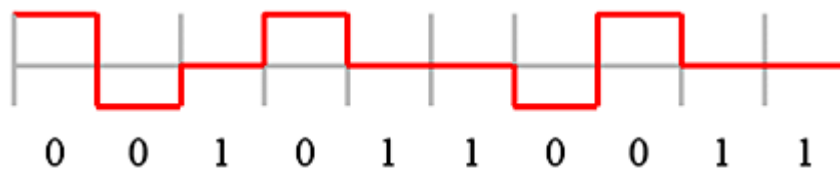


MLT-3: Multilevel Transmission 3: A 0 means no transition happens, a 1 is

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.

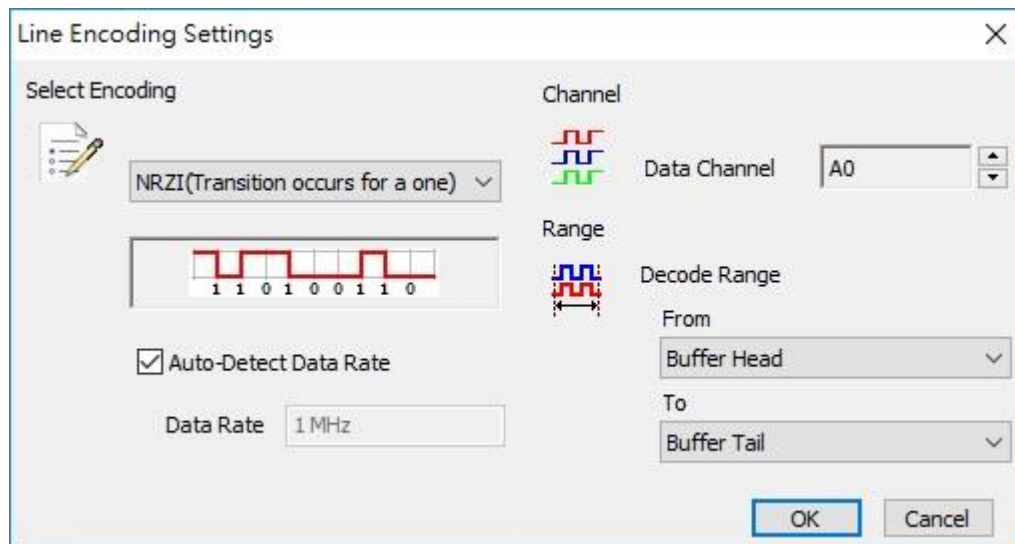
Delay encoding is used primarily for encoding radio signals because the frequency spectrum of the encoded signal contains less low-frequency energy than a conventional non-return-to-zero (NRZ) signal and less high-frequency energy than a bi-phase signal.



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



Select Encoding: Select the line code you want to encode.

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

Channel: Show the selected channel (CH 0).

Range: Select the range within the waveforms you want to encode.

Low Pin Count (LPC)

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

Settings

Channel: Show the selected channels.

LFRAME#: Frame indicator.

LAD[0-3]: Data bits.

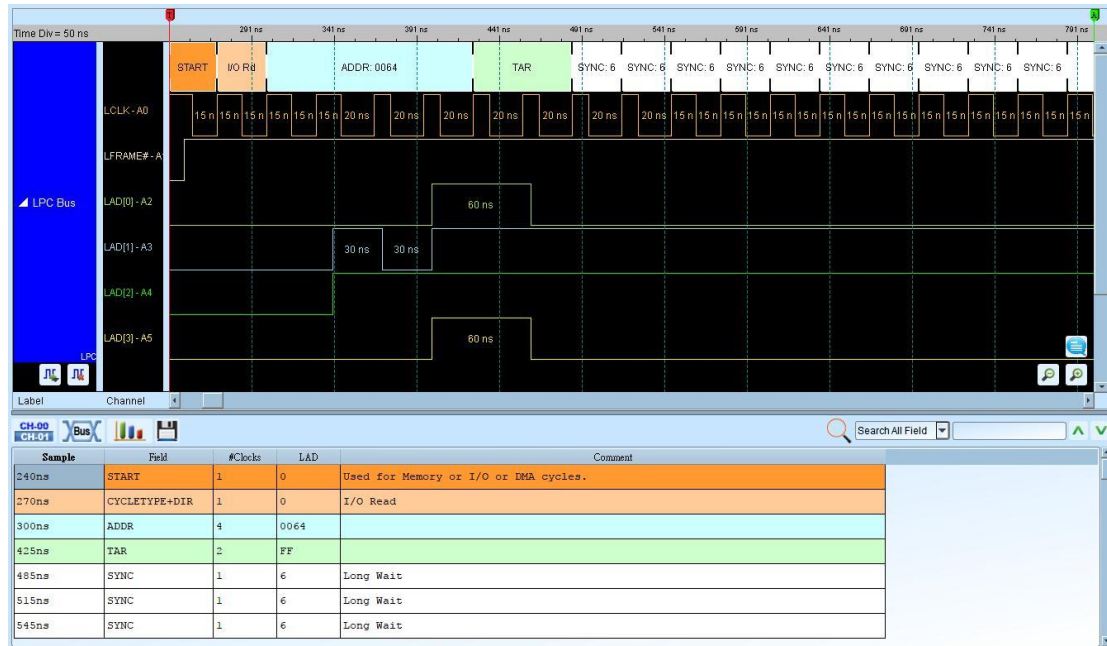
LCLK: Clock.

Filter the data in the report window: You can filter the data in the Report Window.

Result

Click **OK** to run the LPC decode and see the result on the Waveform Window below.

I/O Read Cycle.



Line Printer Terminal Port (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



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.

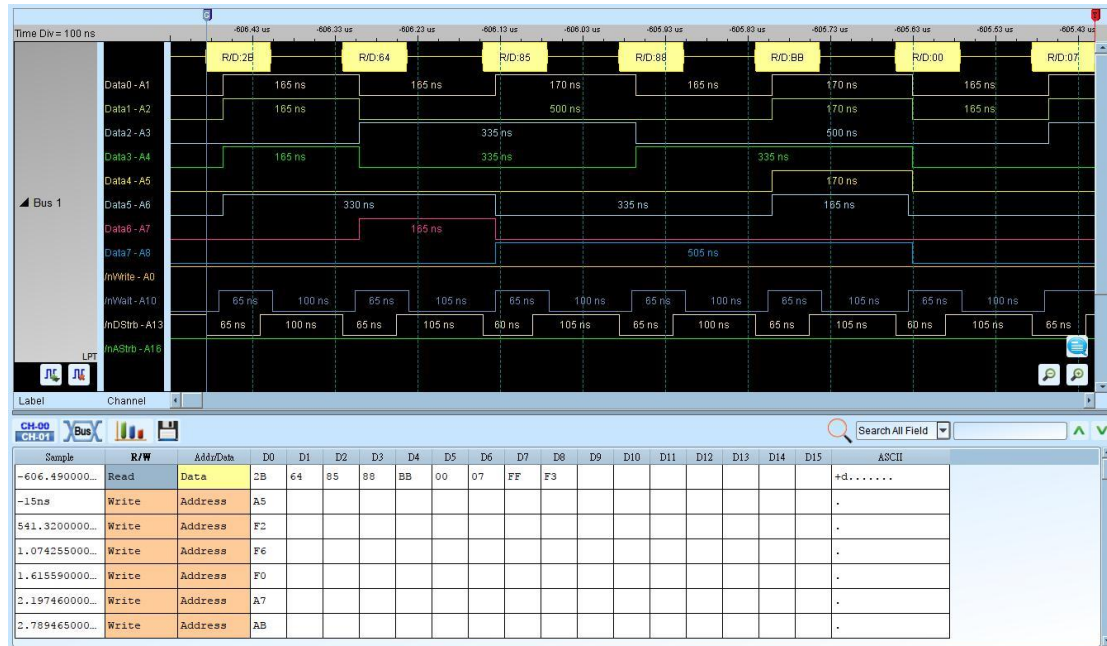
/nInit: Indicates a termination cycle in order to return the interface to the

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

Click OK to run the LPT Decode and see result on the Waveform Windows below.



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

Parity None

MSB first

Adv. report

Color

User can assign color for specific pattern.

Start / Stop

L Field

C Field

A Field

CI Field

Data

Check Sum

Range

From Buffer Head

To Buffer Tail

Default OK Cancel

Channel: Set the channel of the signal and Polarity.

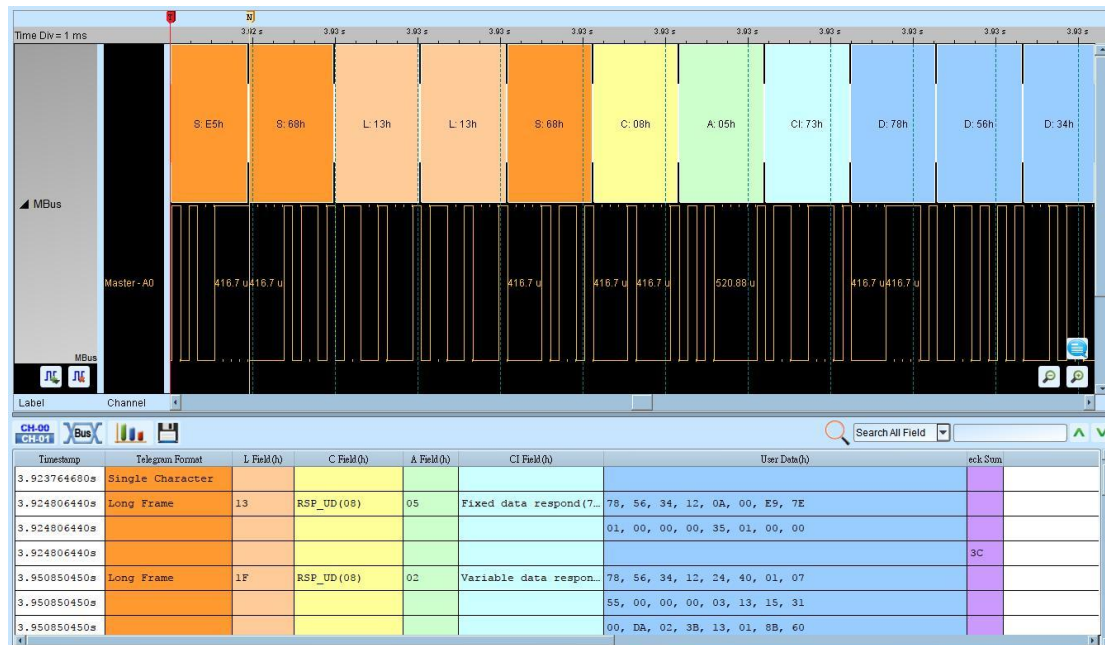
Baud rate: Set the specific data rate or auto detection.

Parity: Error detection.

MSB first: Set MSB format.

Adv. Report: Advanced report.

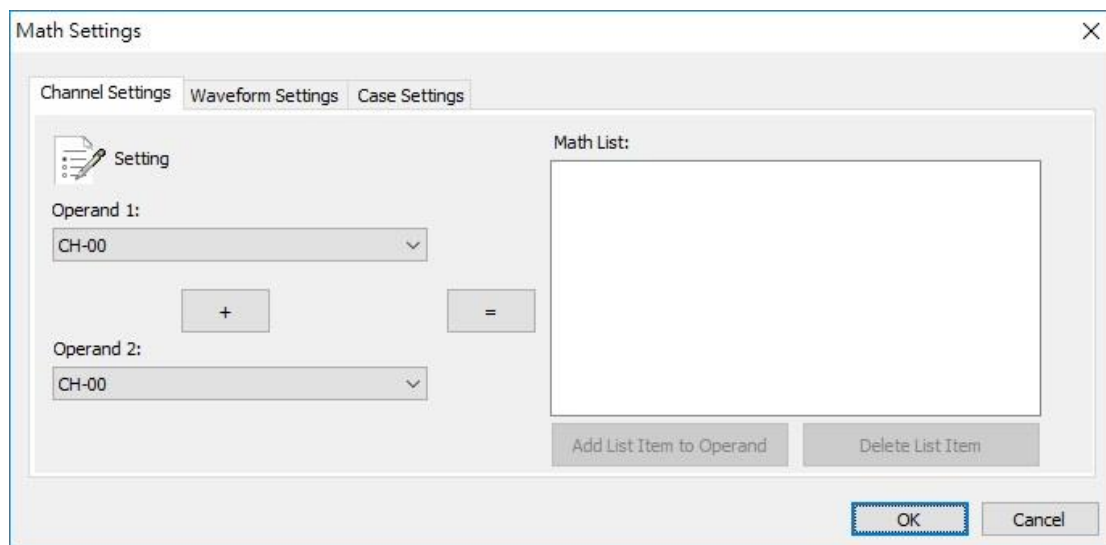
Result



Math

Math is used to conduct addition, subtraction, multiplication, division, AND, XOR, OR, NAND, NOR, XNOR operation for the channel or combined channels.

Settings



Operand: Select the channel(s) in the waveform window.

“+”: Select “+”, “-”, ”X”, ”/”, ”AND”, ”XOR”, ”OR”, ”NAND”, ”NOR”, ”XNOR” operator.

“=”: Add operation type.

Add List Item to Operand: Add operation type to operand.

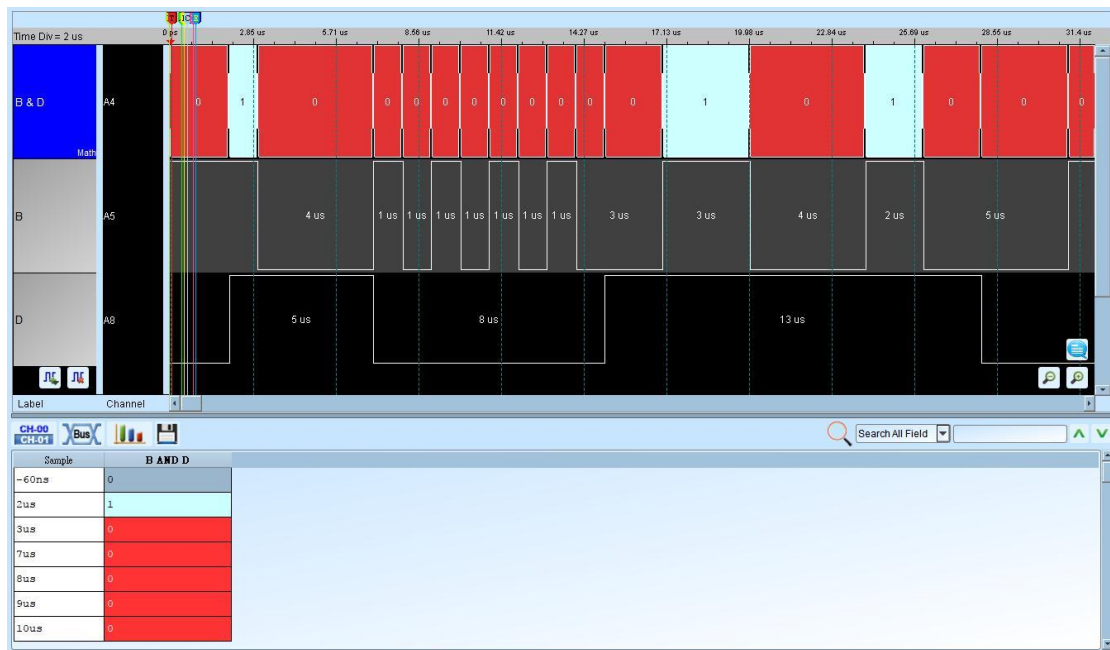
Delete List Item: Delete operation type from the list.

Color: Set the colors for the data bits.

Range: Select the range within the waveform you want to decode.

Case Settings: Select condition and frame color.

Result



Note: When the Math settings are finished, the settings will always be saved as an independent text file named as AqMath.txt, different from the waveform file, at work directory unless other name assigned. If you need the specific Math settings, please save it, so you can reload the settings next time you open the specific waveform file.

Mini / Micro-LED

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

Parameter Settings

DCLK: Default CLK channel

Data: Data channel

LE: The channel that can switch CMD or DATA

Word size: The number of the data bits

Bit order: MSB/LSB

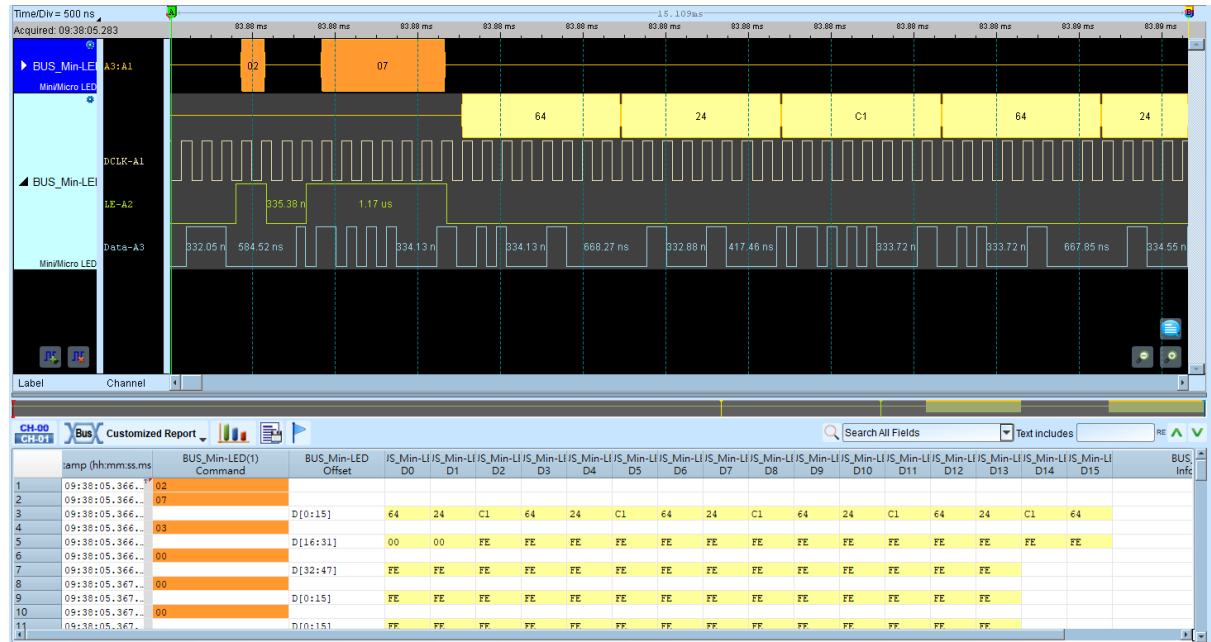
GCLK: The magnification of DCLK

Data edge: Rising/falling latch

Skip Data bit: Set the number of skip data bit after LE falling

Delay time: Set the time lead / delay in data line ◦

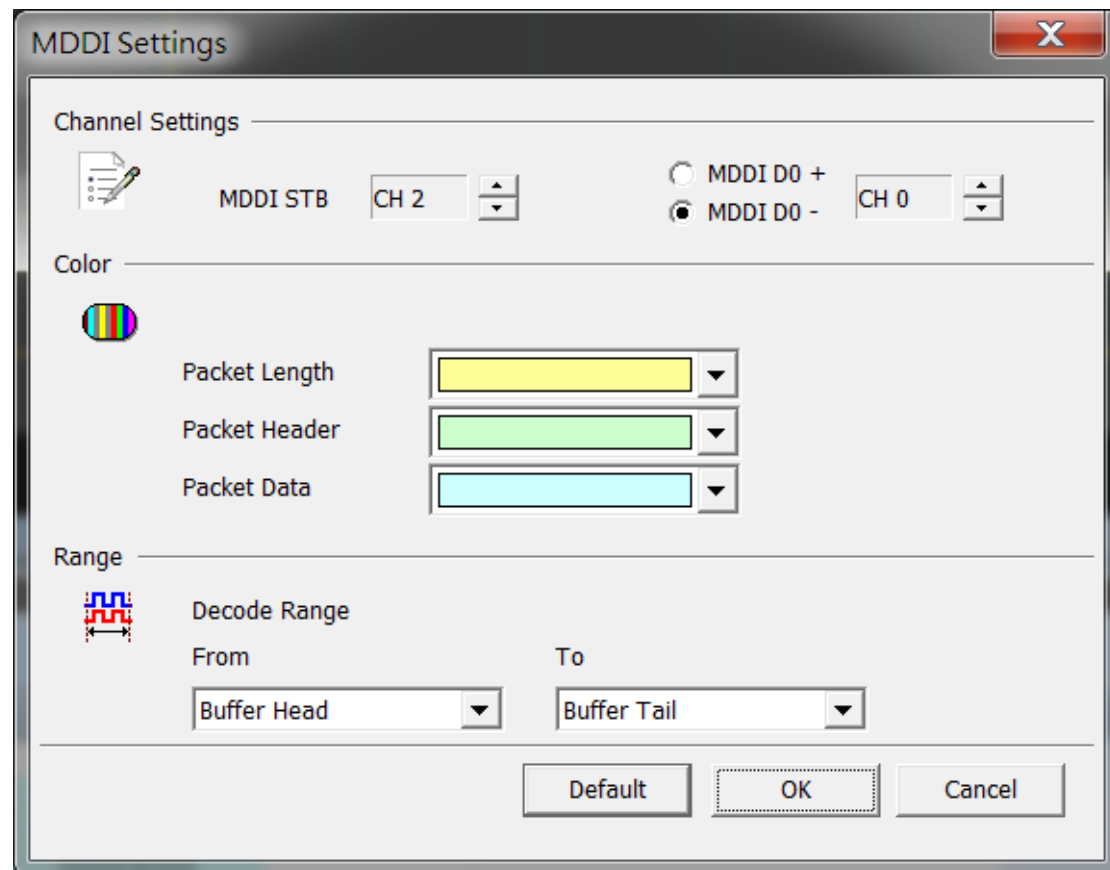
Result



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.

1. MDDI Parameter Settings



Channel Settings

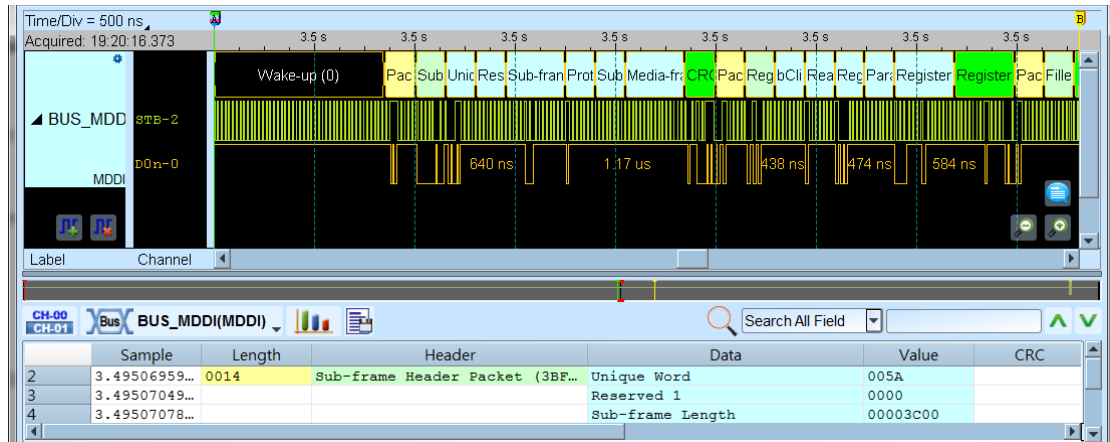
MDDI STB: MDDI Strobe

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

Configure the Channel setup for the decoder, and choose the data source

from Data 0+ or Data 0-.

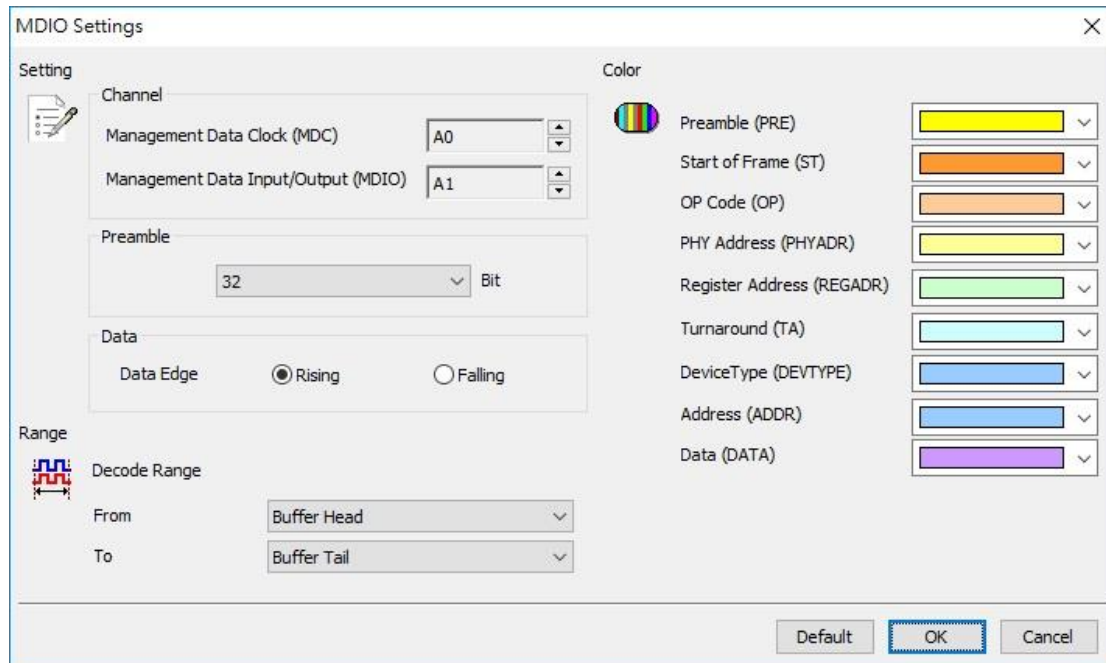
2. Result



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



MDC: Clock.

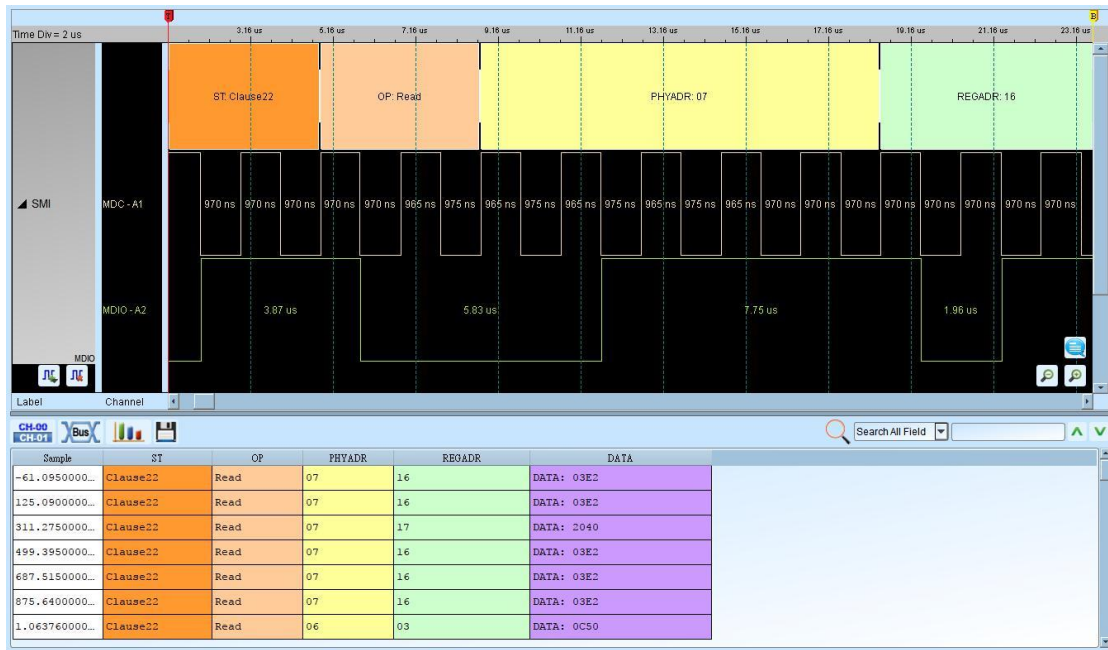
MDIO: Data Input / Output.

Preamble: Set the MDIO preamble width, 32 bit default.

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

Result

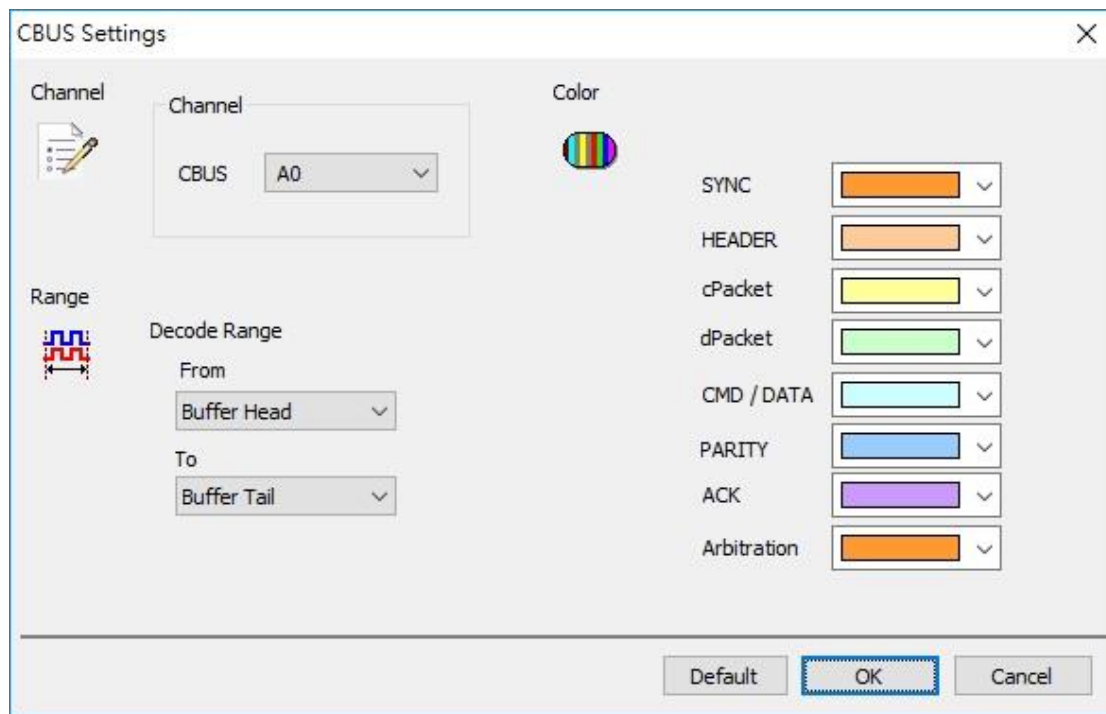
Click **OK** to run the MDIO decode and see the result on the Waveform Window below.



MHL-CBUS

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

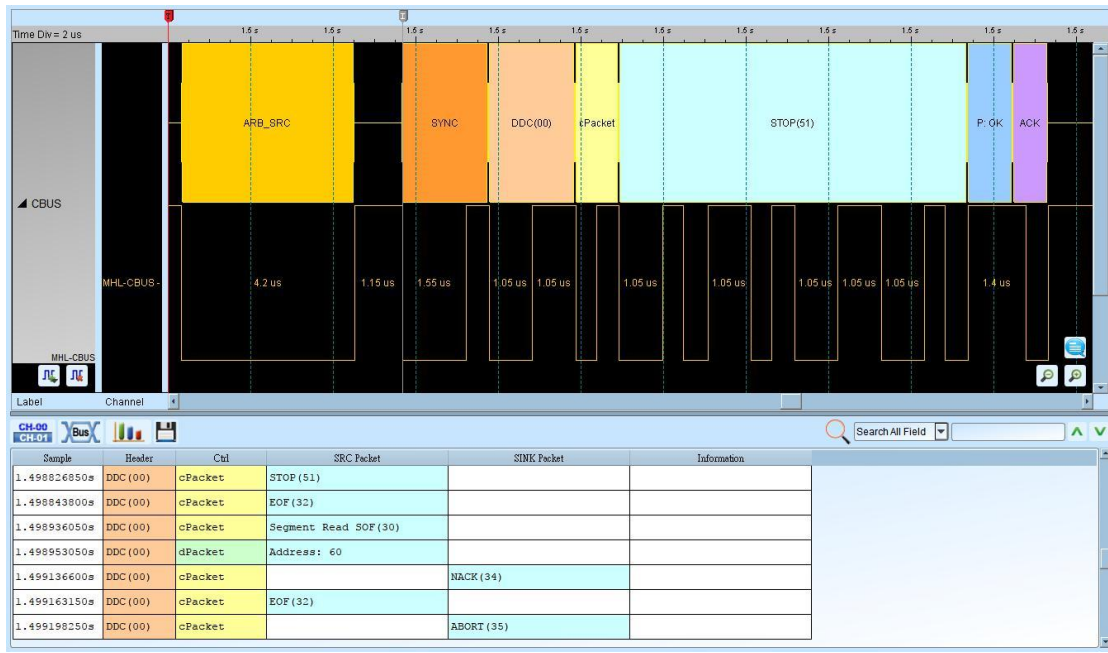
Settings



LA Channel: Show the selected channel (CH0).

Result

Click OK to run the MHL-CBUS Decode and see result on the Waveform Window below.



MII/RMII

MII/RMII (Media Independent Interface/Reduced Media Independent Interface) is a protocol 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

The screenshot shows the 'MII / RMII / GMII / RGMII Settings' dialog box. It is organized into several sections:

- Setting:** Includes a list of protocols: MII, RMII, Only CLK and DATA pins used (MII), GMII, RGMII, and Only CLK and DATA pins used (GMII).
- Mode:** Radio buttons for Transmit (Tx), Receive (Rx), and Duplex (Tx+Rx).
- Channel:** A red box highlights 'The data source from DSO - Differential'. Below are two columns of dropdown menus for Transmit (Tx) and Receive (Rx) signals, including TD+, TD-, TX_D1 through TX_D7, TX_EN, TX_ER, TX_COL, RD+, RD-, RX_D1 through RX_D7, RX_DV, and RX_ER.
- Data:** Radio buttons for Data Edge: Rising and Falling.
- Report:** Radio buttons for Data Columns: 8 columns and 16 columns.
- Decode Ethernet Packet (MAC):** A checked checkbox.
- Color:** A color selection icon and a list of color swatches for Data (yellow), Error (red), Collision (green), Idle (black), Carrier Extend / False Carrier indication / Status (cyan), and Preamble / SFD (light blue).
- Range:** A 'Decode Range' section with 'From' and 'To' dropdown menus, currently set to 'Buffer Head' and 'Buffer Tail'.

At the bottom of the dialog are three buttons: 'Default', 'OK', and 'Cancel'.

MII / RMII: Select the MII / RMII bus decode

GMII / RGMII: Select the GMII / RGMII bus decode

Only CLK and Data pins used(M/G): Select MII / GMII the CLK and Data pins only.

Transmit (Tx): Select TX mode

Receive (Rx): Select Rx mode

Duplex (Tx+Rx) : Select duplex mode

Channel: Set the channel number.

Rising: Select rising edge to latch data

Falling: Select falling edge to latch data

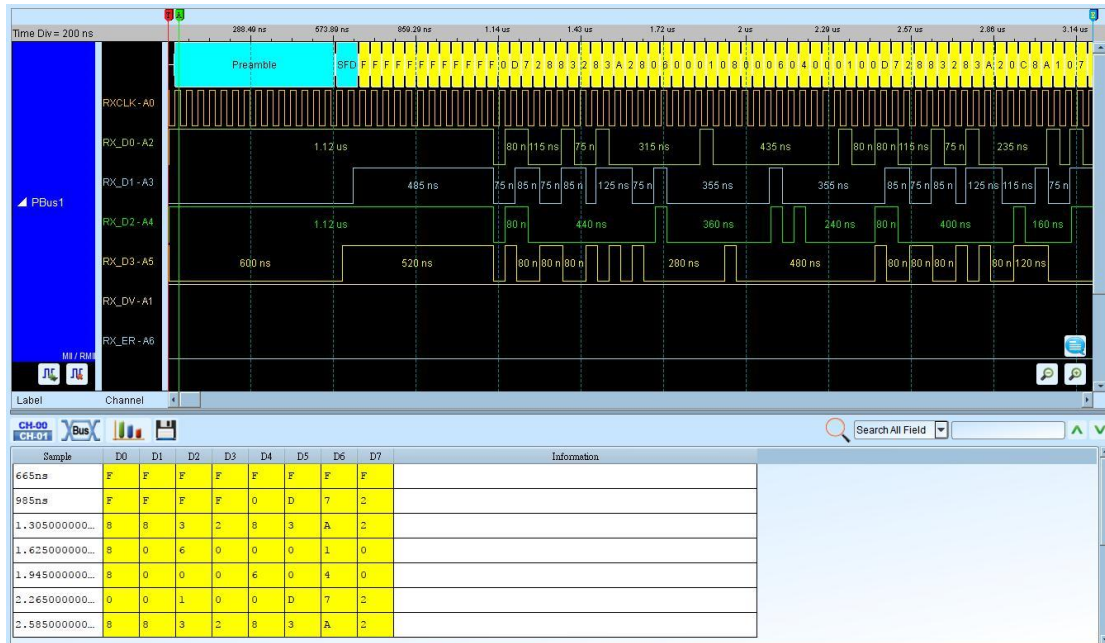
8 columns: show 8 columns data field in the report window

16 columns: show 16 columns data field in the report window

Result

Click **OK** to run the MII/RMII decode and see the result on the Waveform Window below.

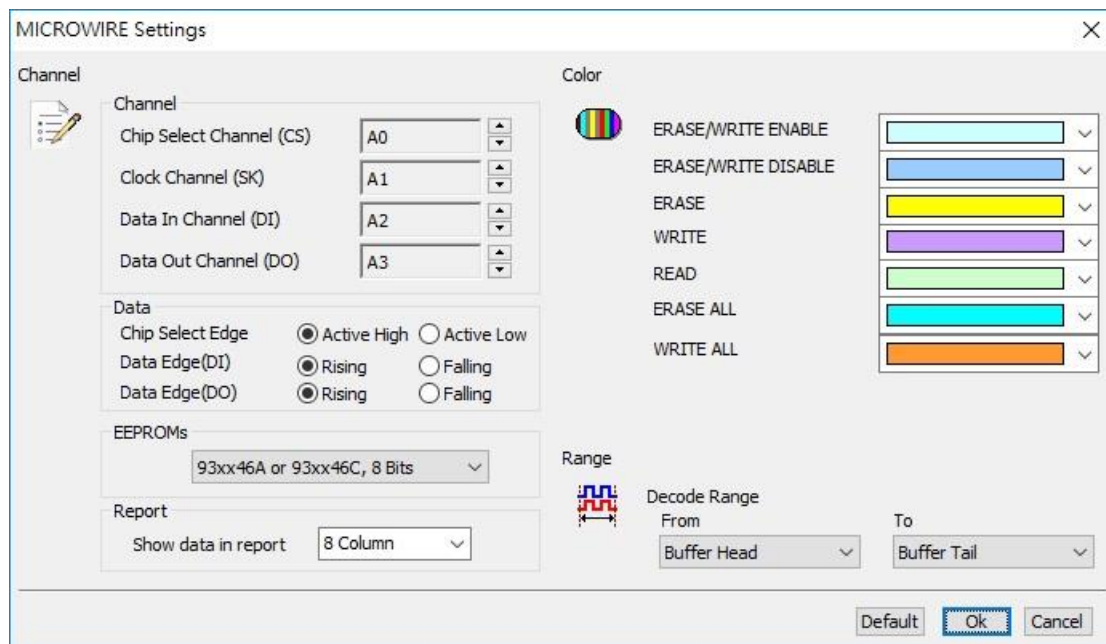
MII



Microwire

The Microwire bus has four data bits: Chip Select (CS), Serial Clock (SK), Data Input (DI), and Data Output (DO).

Settings



Channel: Show the selected channels.

Chip Select Edge: Active Low or Active High.

Data Edge: Rising or Falling.

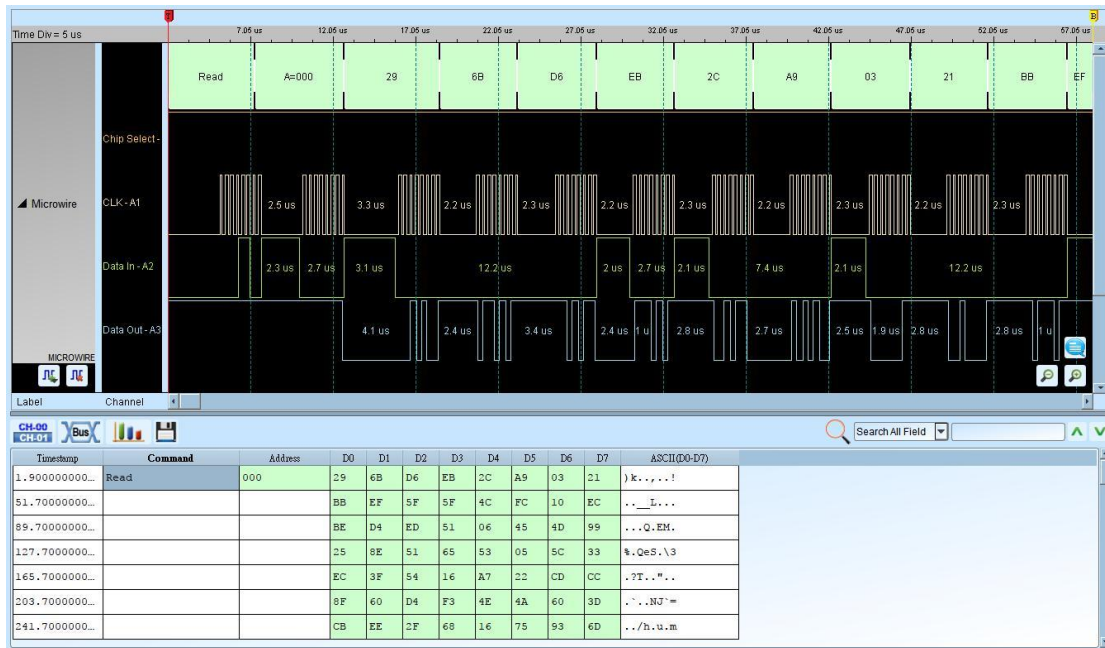
EEPROMs: Select EEPROMs.

Report: Show data in report.

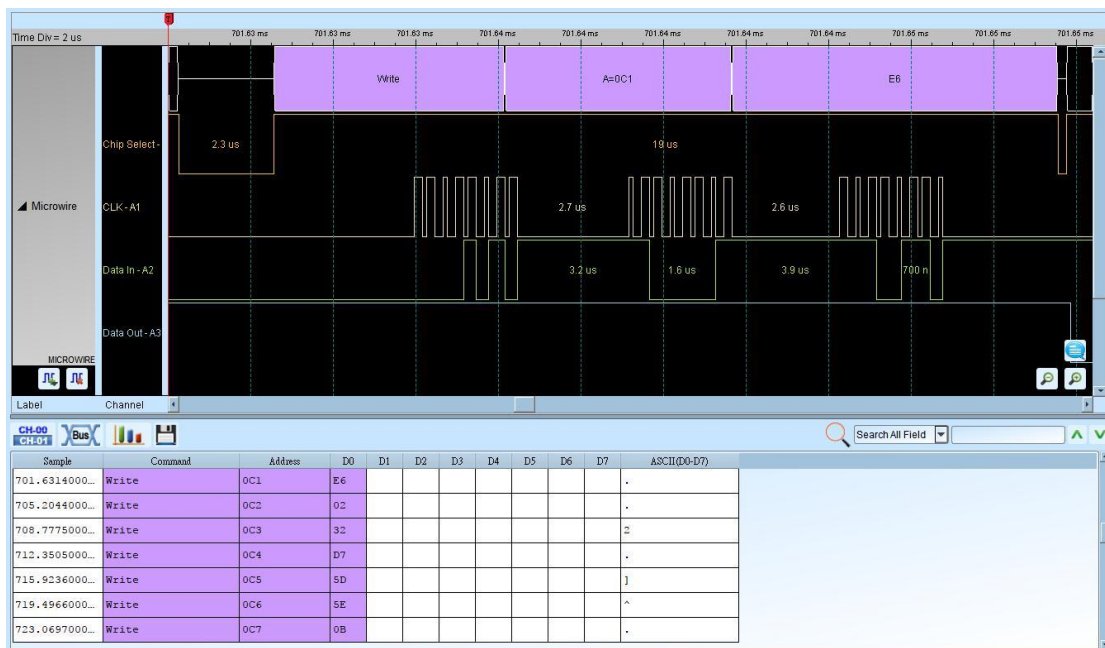
Result

Click **OK** to run the Microwire decode and see the result on the Waveform Window below.

Read



Write



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

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.

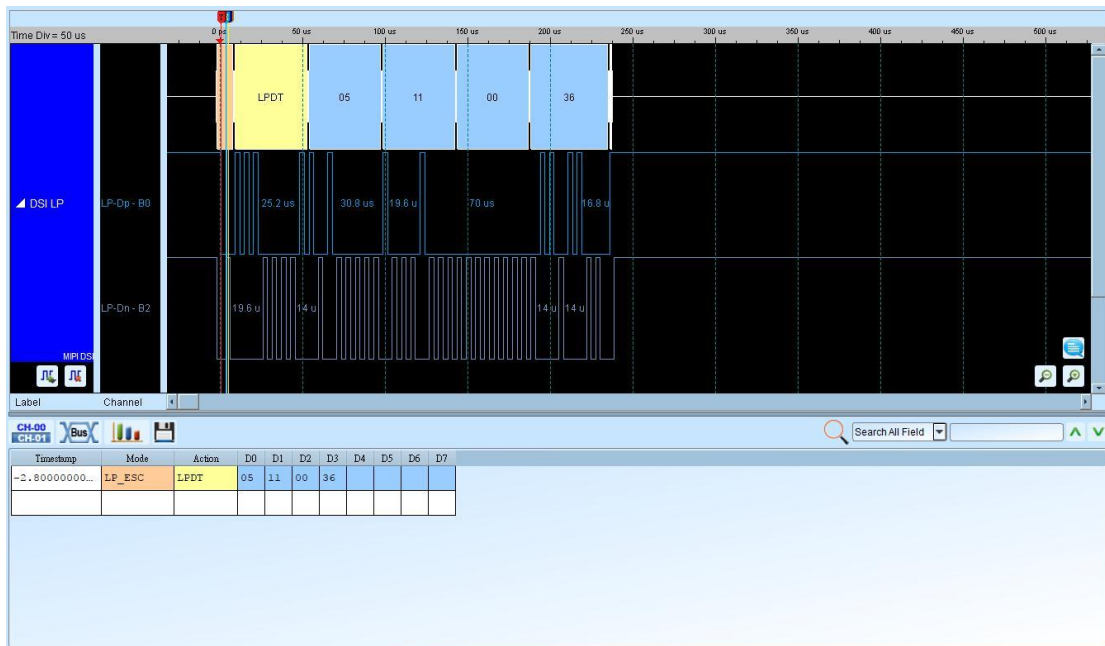
Show DCS Command: Enable DCS Command decode and display.

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

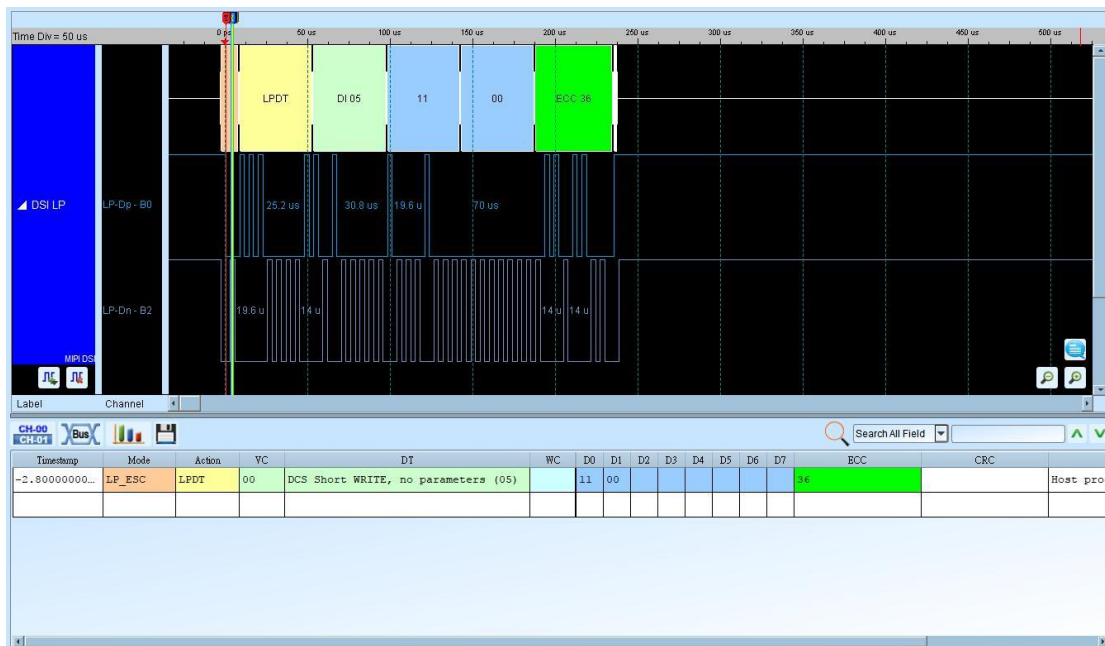
Initial Bus Direction: Select the Initial direction of the bus transmission.

Result

Advanced Decode Disabled:



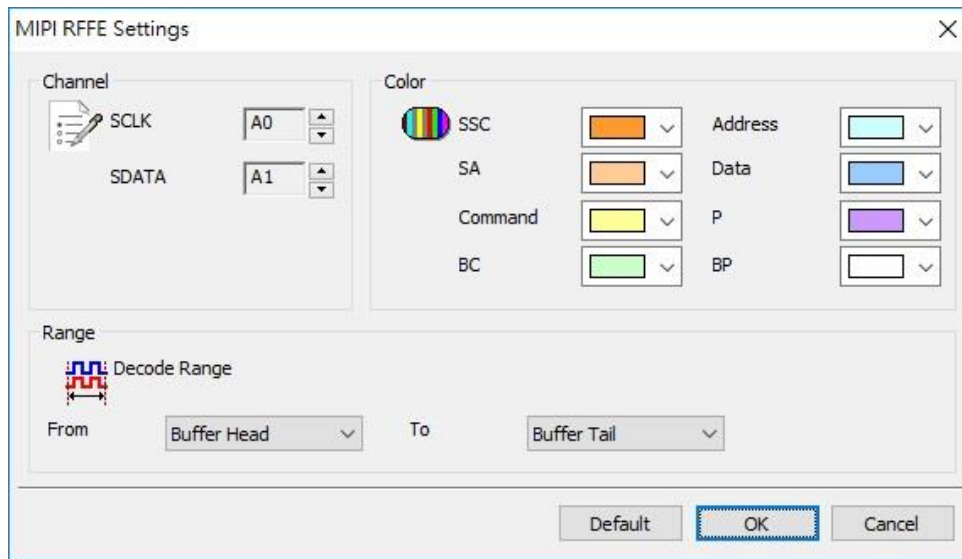
Advanced Decode Enabled:



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



Channel: Set the channels of SCLK and SDATA.

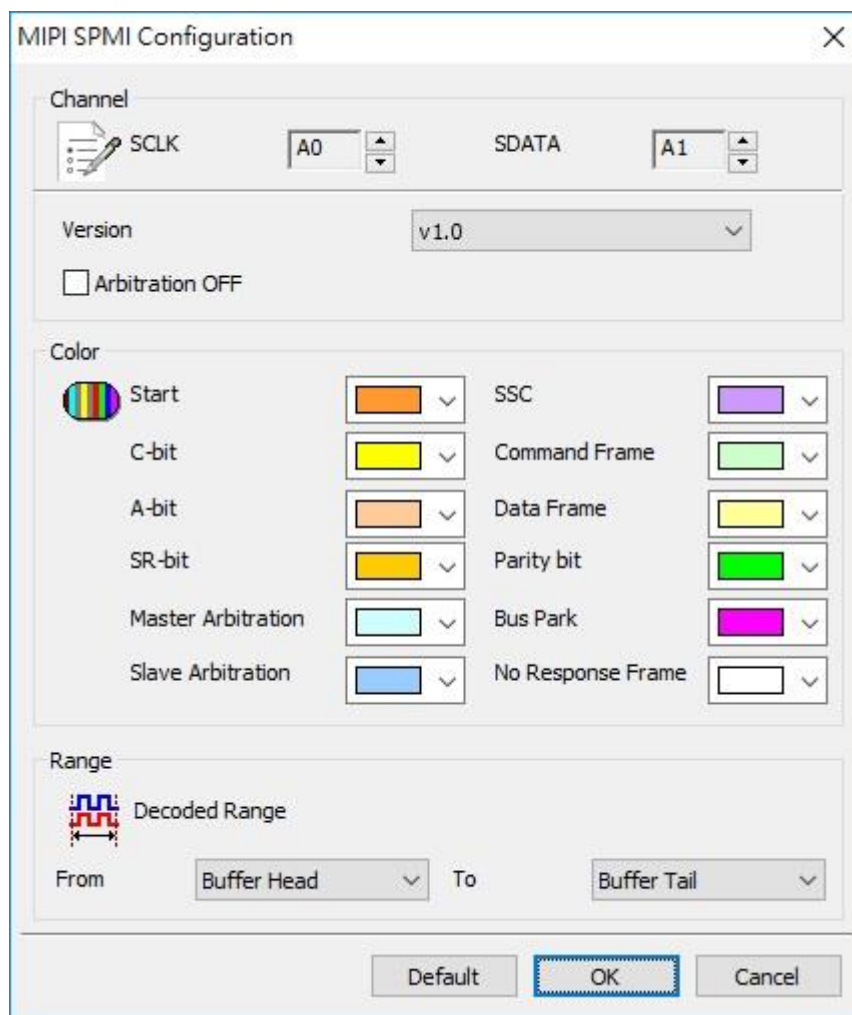
Result



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

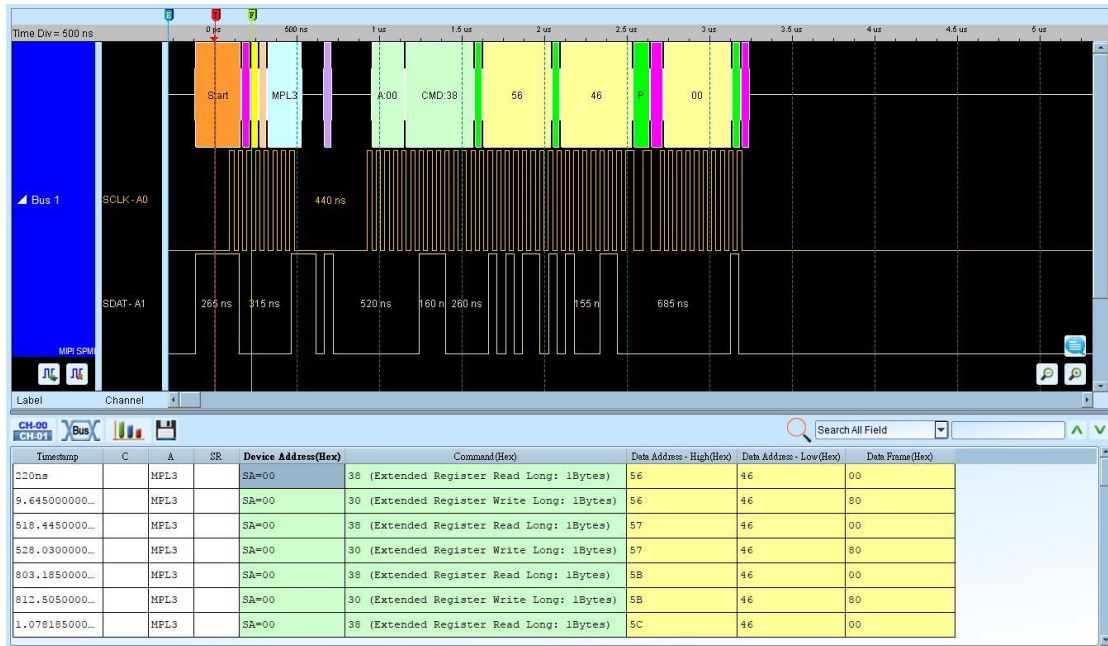


Channel: Set the channels of SCLK and SDATA.

Result

Click OK to run the MIPI SPMI Decode and see result on the Waveform Windows

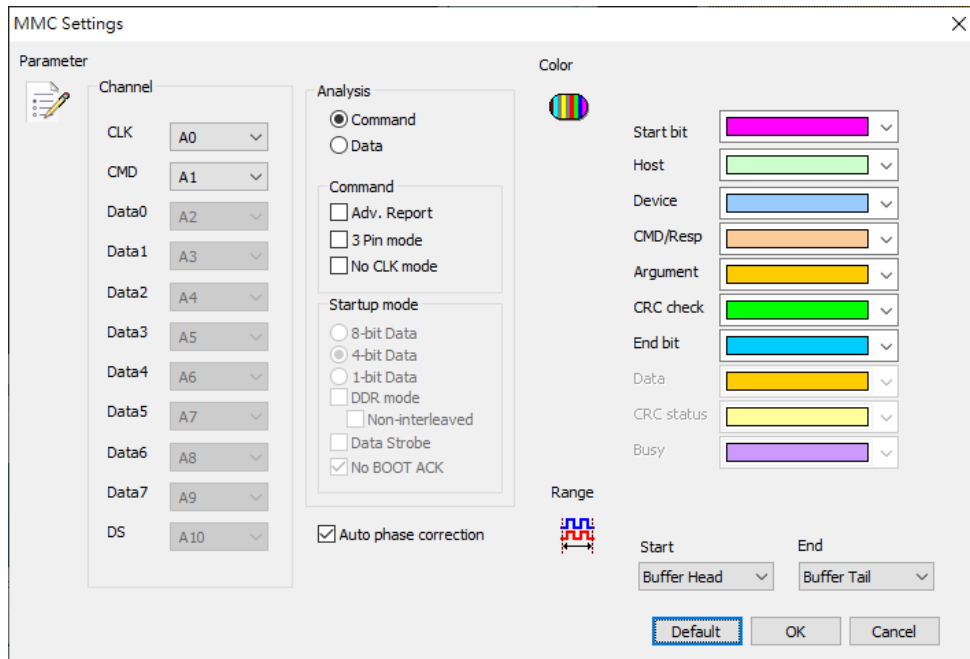
below.



MMC

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

Settings



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

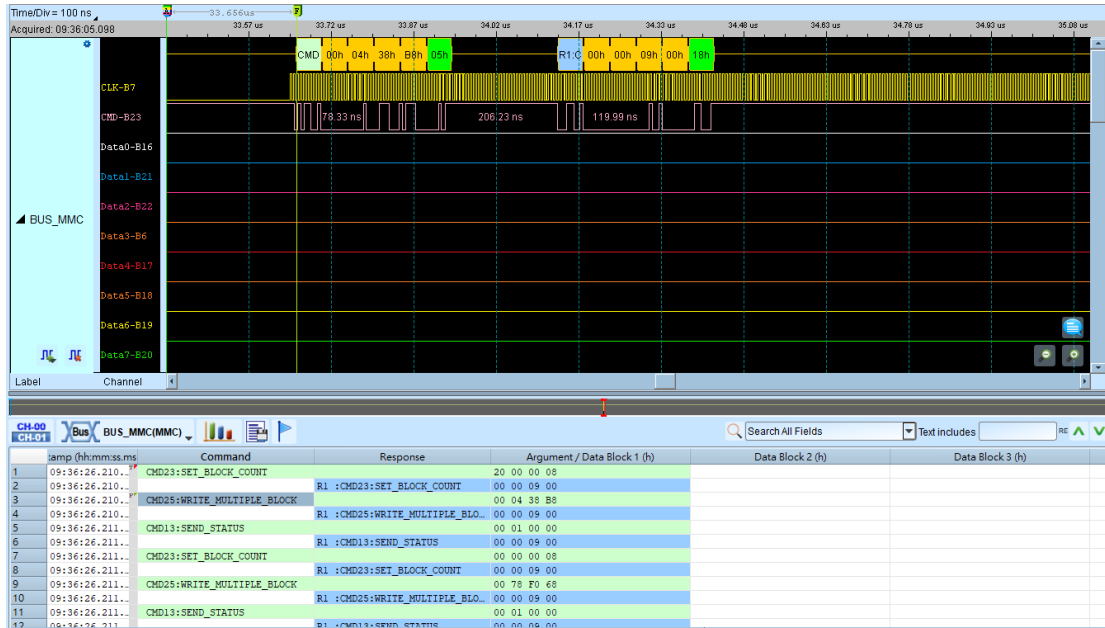
Startup mode: **When start to decode, set the DUT. status**

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

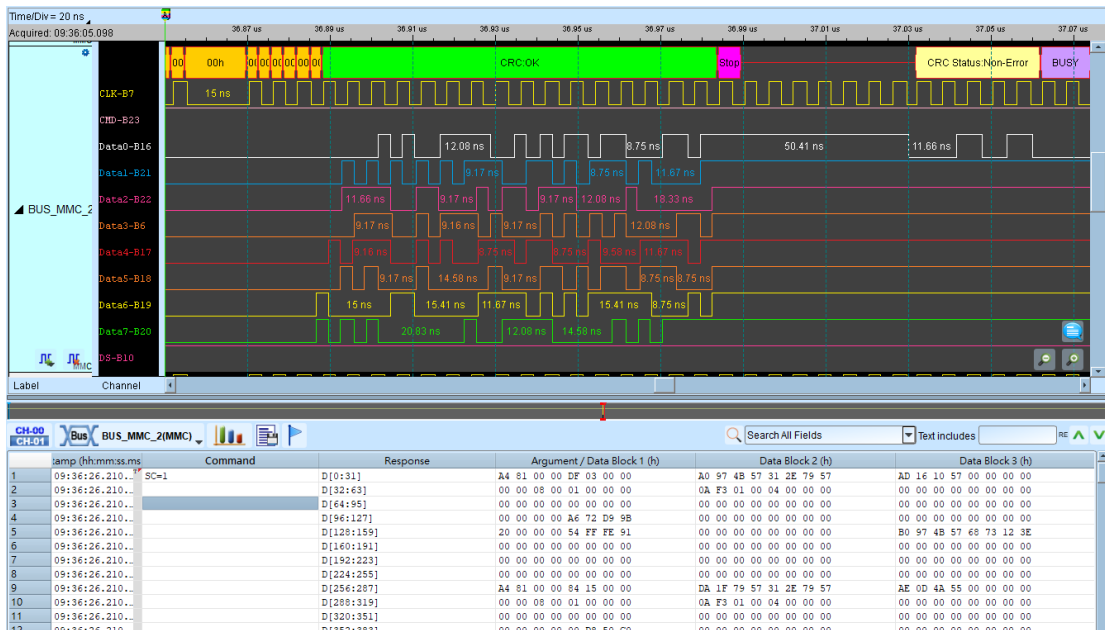
Auto Phase Correction: If it's checked, the decoder will auto adjust the phase of LA when measure the DUT..

Result

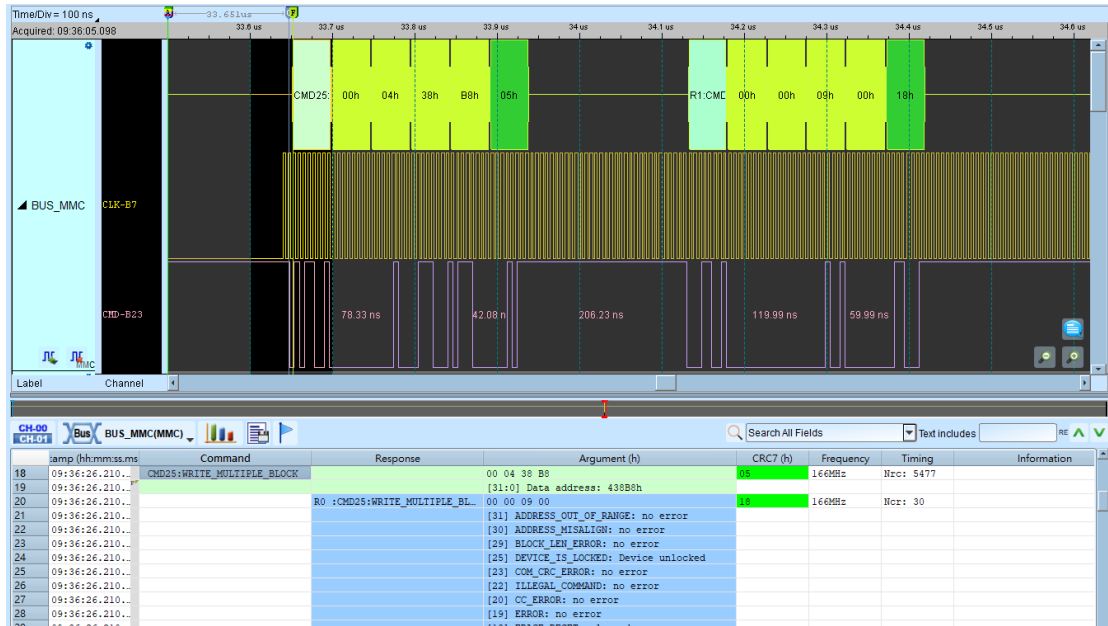
Command:



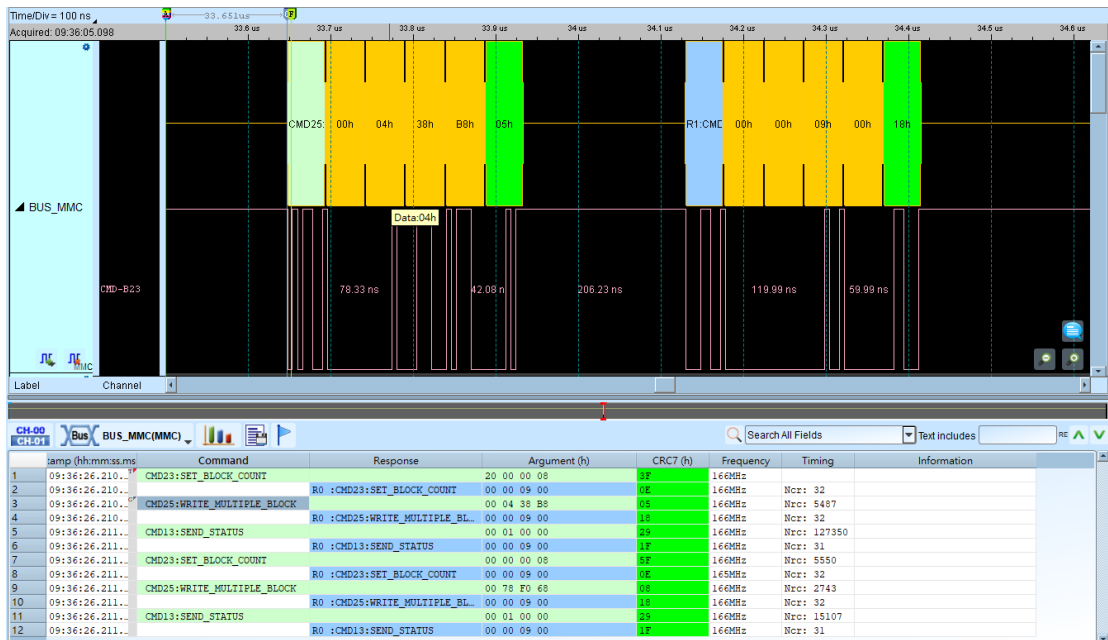
Data:



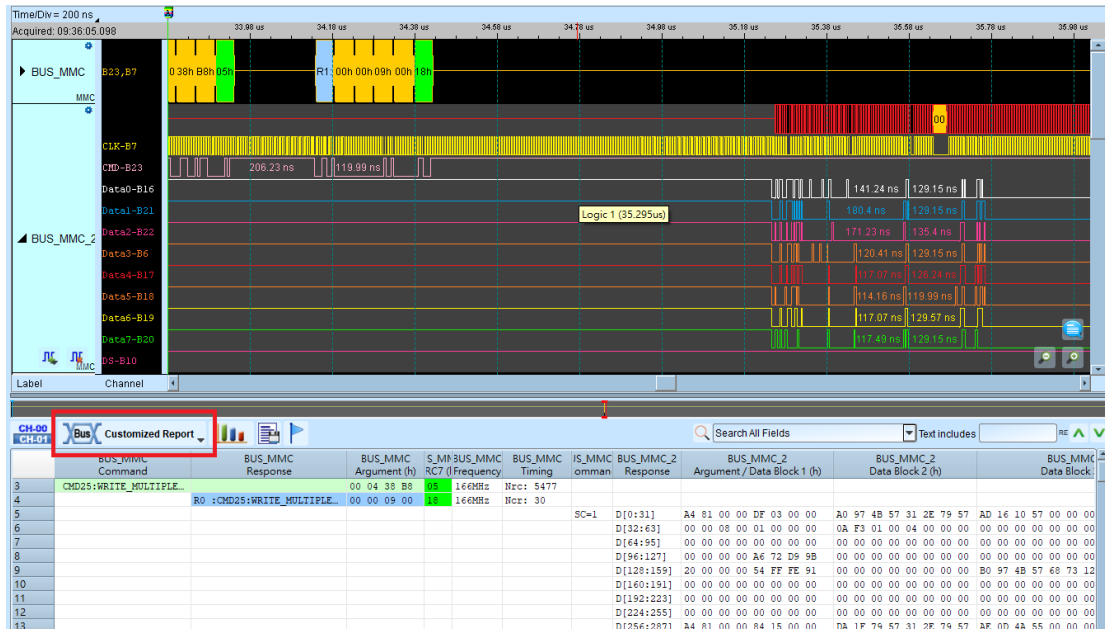
Adv. Report:



No CLK mode



Command + Data: (Customized Report)



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

Channel: Modbus (Tx) or Modbus (Rx).

Transmission Mode: ASCII and RTU mode.

Auto: Auto detection idle polarity.

Idle high: Idle condition shows High.

Idle low: Idle condition shows Low.

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

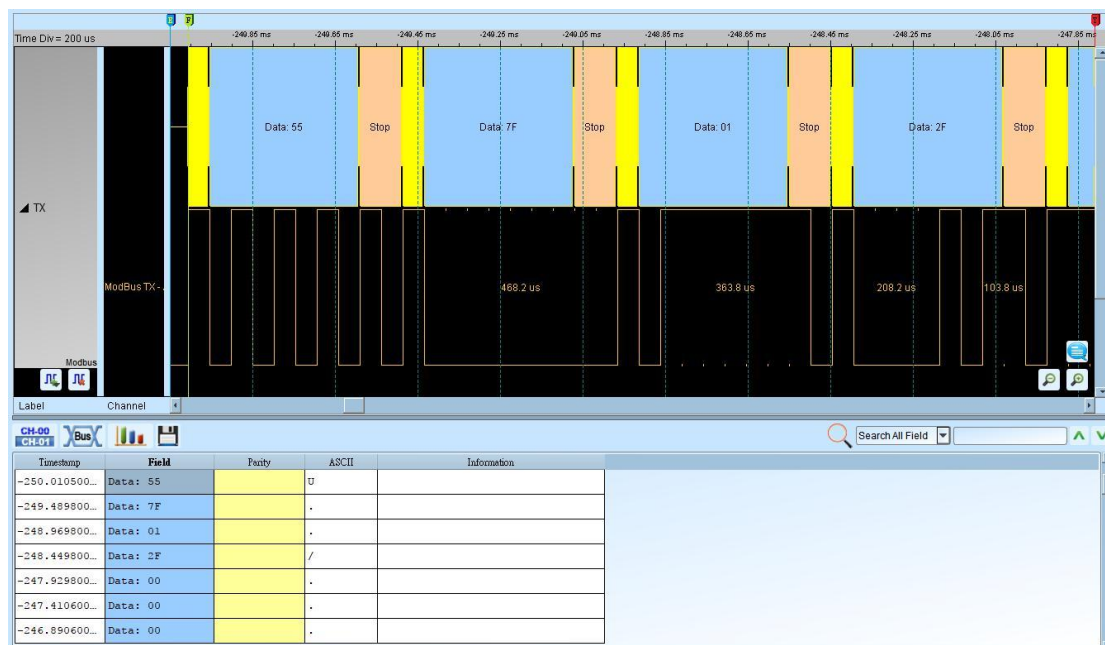
Baud Rate: Data rate (bits per second), and the range is 110 ~ 2M (bps).

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

MSB First: The default is LSB first; click it to change to MSB first.

Show scale in the waveform: Display the waveforms with scales.

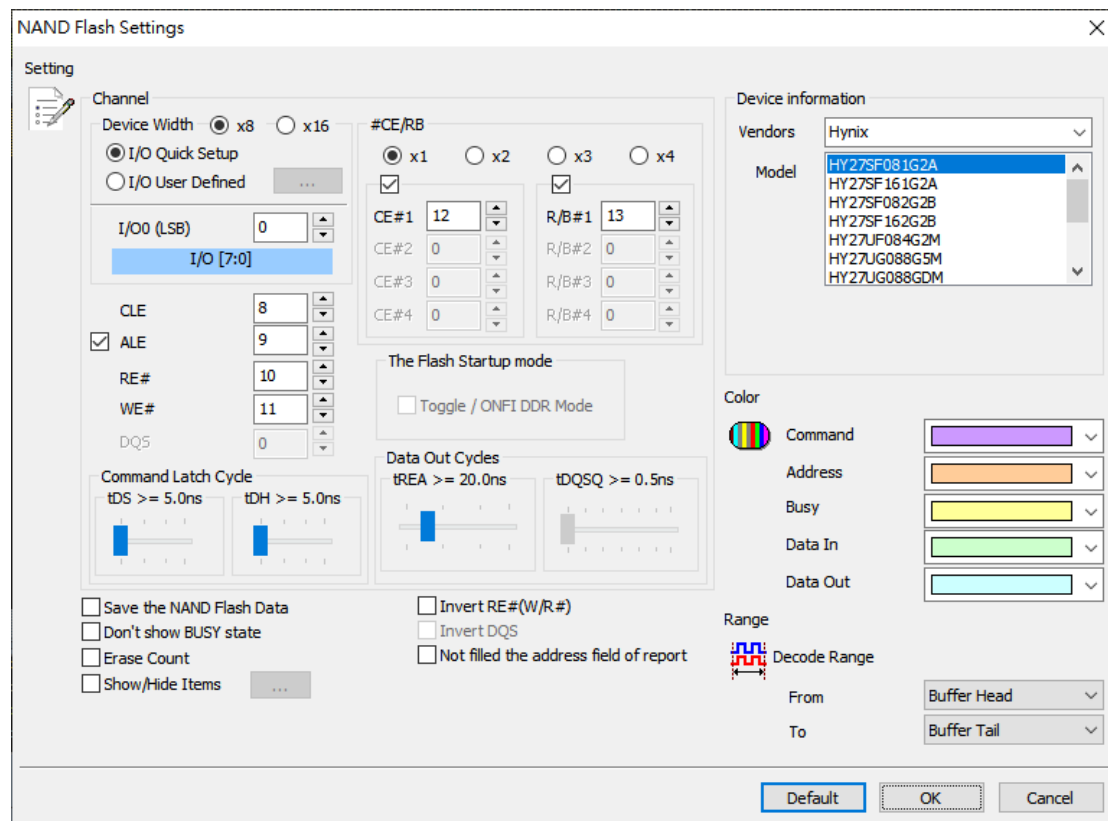
Result



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



Channel:

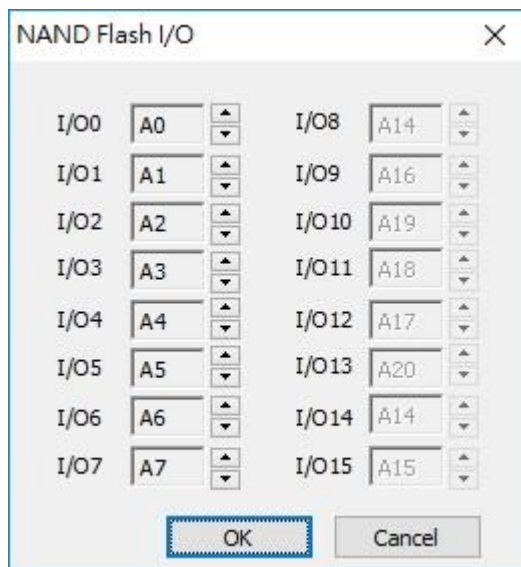
Async	Ssync	Description
I/Ox	DQx	NAND Flash data channels
CLE	CLE	Command Latch Enable channel
ALE	ALE	Address Latch Enable channel
RE	W/R	Read Enable and Write/Read channel

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:



User can set NAND I/O channel by channel.

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

t_{REA} / t_{DQSQ}: Set the delay time to access the NAND data under SDR / DDR.

To adjust the t_{REA}/t_{DQSQ} 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
_Rowxxxxxh	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. 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.

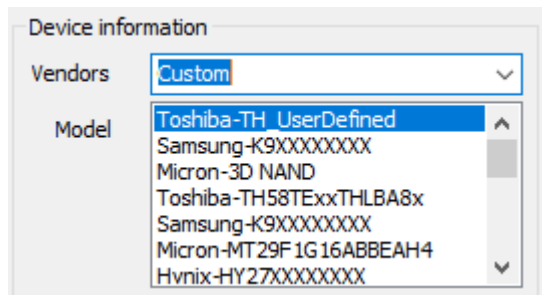
3 Flags: 1st flag of "Read Status" is Y means it's acceptable command

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.



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	tDQSK	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: $0.45 \times tDSC(\text{avg})$
2. tQSH/tQSL: $0.37 \times tRC(\text{avg})$
3. tREH/tRP: $0.43 \times tRC(\text{avg})$

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

```

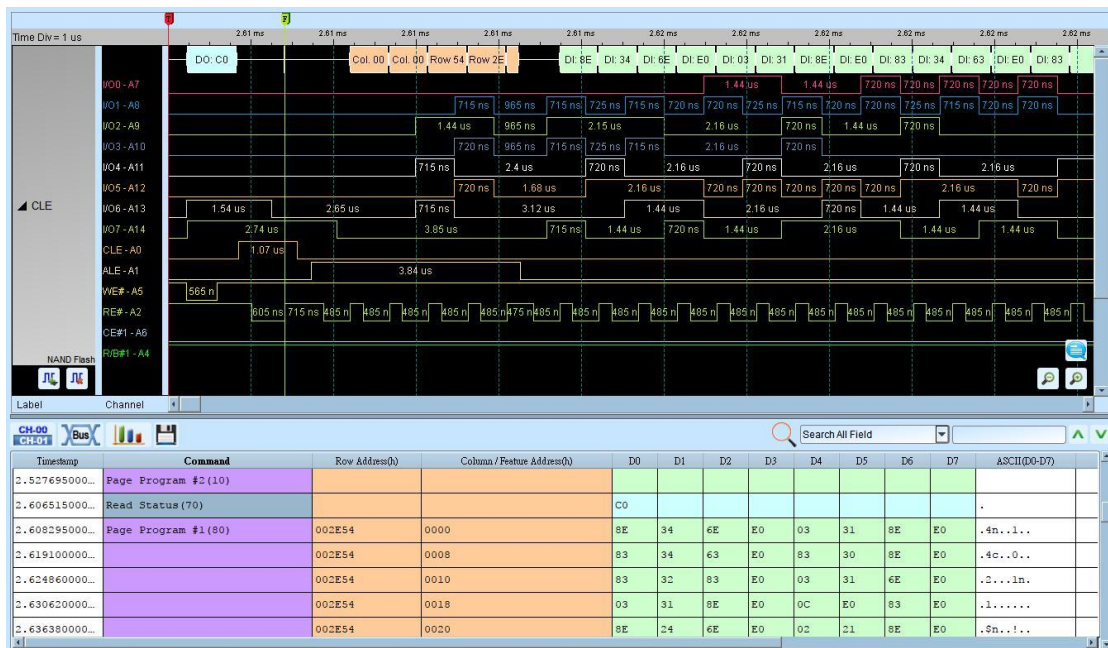
If you do not use the above Timing calculation method, you only need to enter
UsedtRCavg=N, UsedtDSCavg=N in the file header or remove this description
completely, and the value entered for the time check item will be the
minimum/maximum value of the time deal with.

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



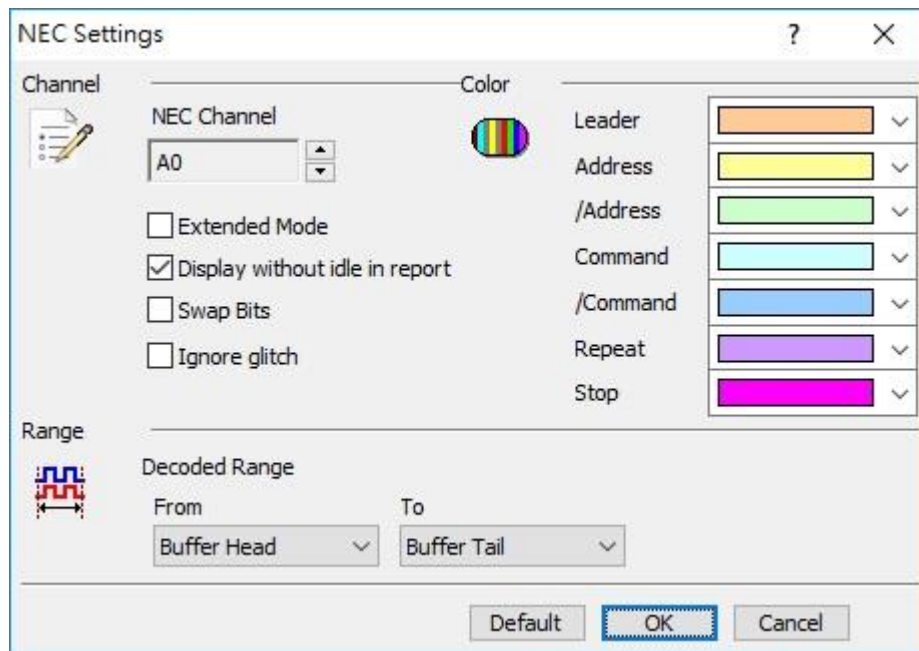
Timing Check

Timestamp	Command	Row Address(h)	Column / Feature Address(h)	D0	D1	D2	D3	D4	D5	D6	D7	ASCII(D0-D7)	Information
29													tWC (152.09 ns) :Min(25 ns)
30													tWC (152.50 ns) :Min(25 ns)
31													tAR (182.50 ns) :Min(10 ns)
32													tWHR (234.17 ns) :Min(80 ns)
33													tRPRE (100.83 ns) :Min(15 ns)
34													tRP (abs) (100.83 ns) :Min(0.43 ns)
35													tRC (103.75 ns) :Min(3.75 ns)
36													tWHR (337.92 ns) :Min(80 ns)
37													tWPRE (97.06 ns) :Min(15 ns)
38													tDQPRE (3.33 ns) :Min(8 ns) /Max(25 ns)
39													tDSL (97.06 ns) :Min(0.37 ns)

NEC IR

It needs only one channel to analysis NEC signals.

Settings



Channel: Display the channel (CH 0).

Extended Mode: It integrates /Address and Address into 16 Bits Address, /Command and Command into 16 Bits Command.

Display without idle in report: It will not idle on the Report Window for the user to observe and analyze data.

Swap Bits: Switch LSB First to MSB First.

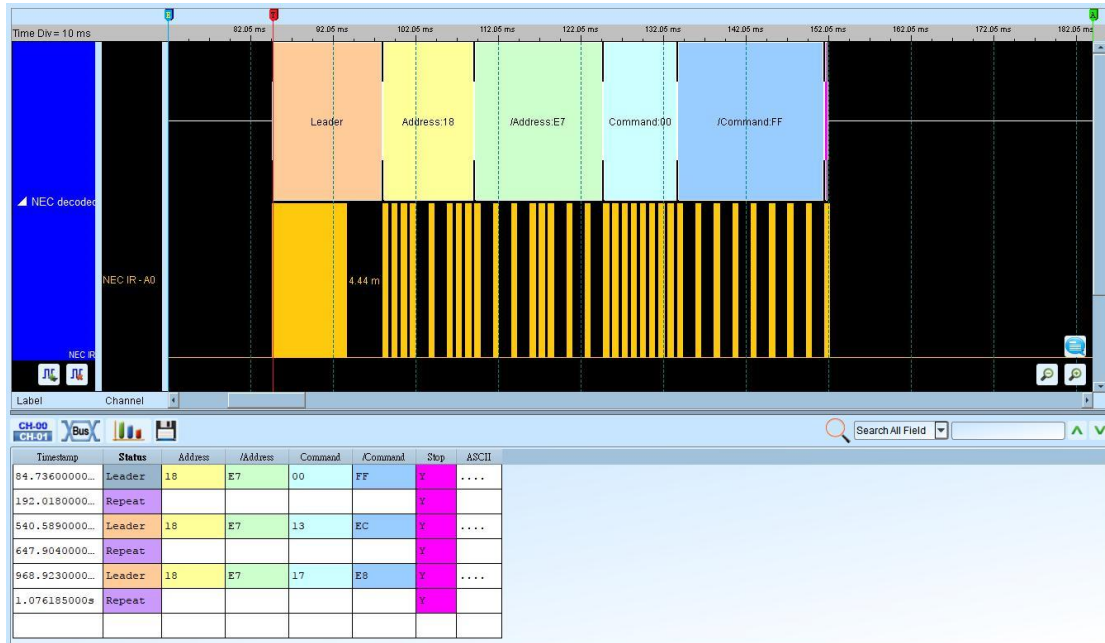
Auto: Shows High or Low when auto detection Idle.

Idle high: Idle condition shows High.

Idle low: Idle condition shows Low.

Result

Click **OK** to run the NEC IR decode and see the result on the Waveform Window below.



PECI

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

Settings

PECI Settings

Channel

Channel

Data A0

Report mode

Normal Advance

Color

Sync

Address

WL / RL

FCS

Data

Range

Decode range

From To

Buffer Head Buffer Tail

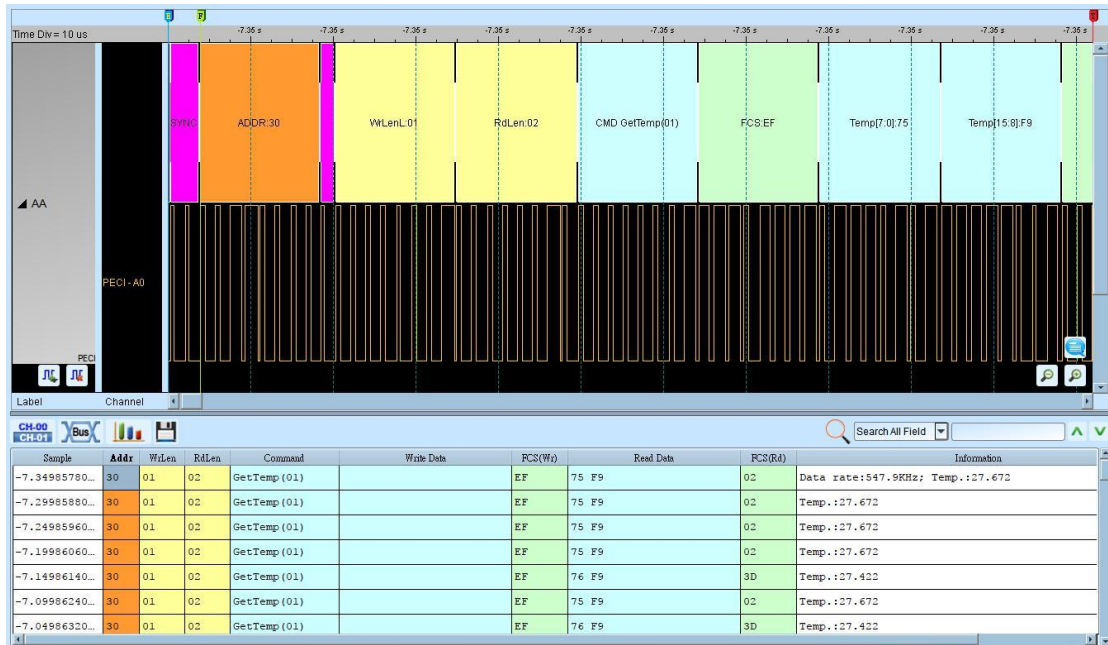
Default OK Cancel

Channel: Show the selected channels.

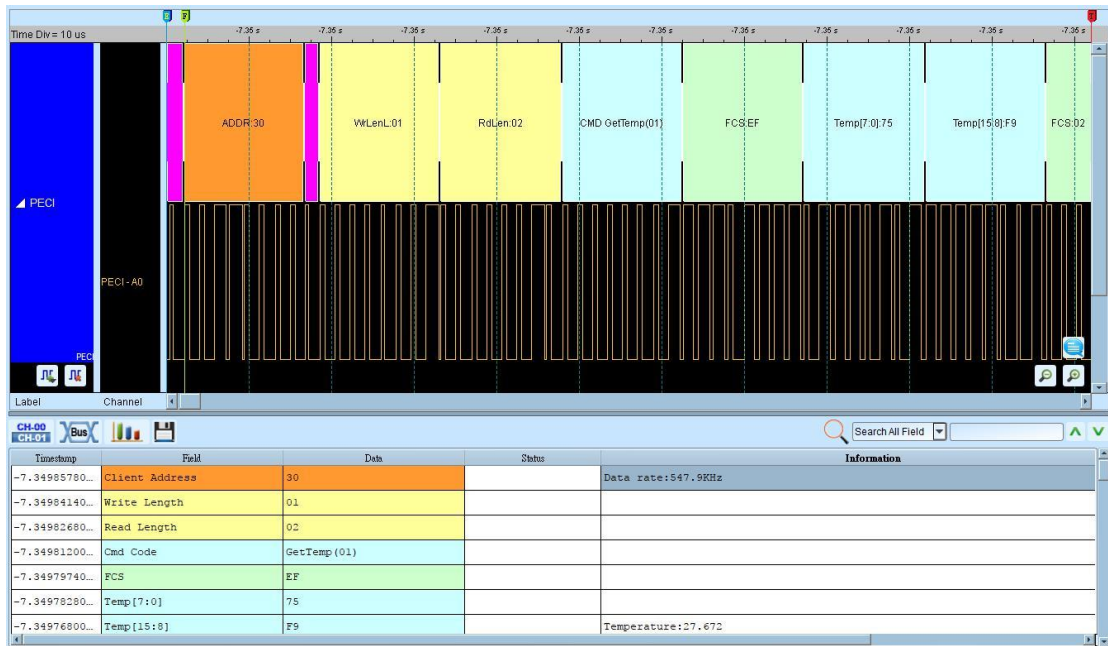
Report mode: Normal or Advance

Result

Normal mode



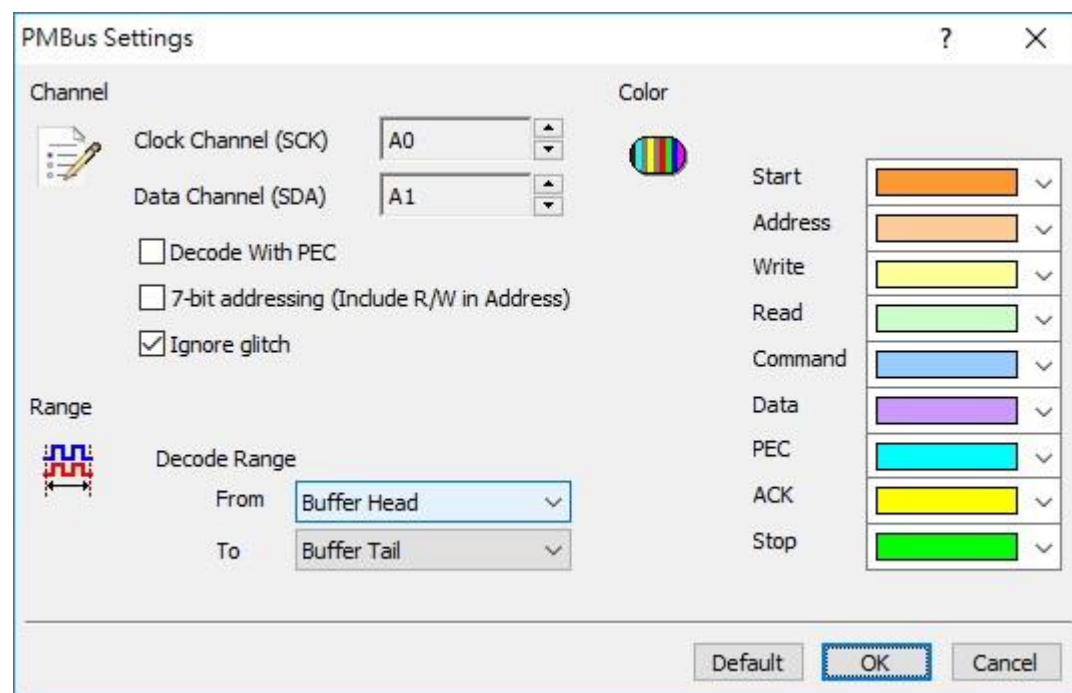
Advance mode



PMBus

The Power Management Bus (“PMBus”) is an open standard protocol that defines a means of communicating with power conversion and other devices.

Settings



Channel: Show the selected channels.

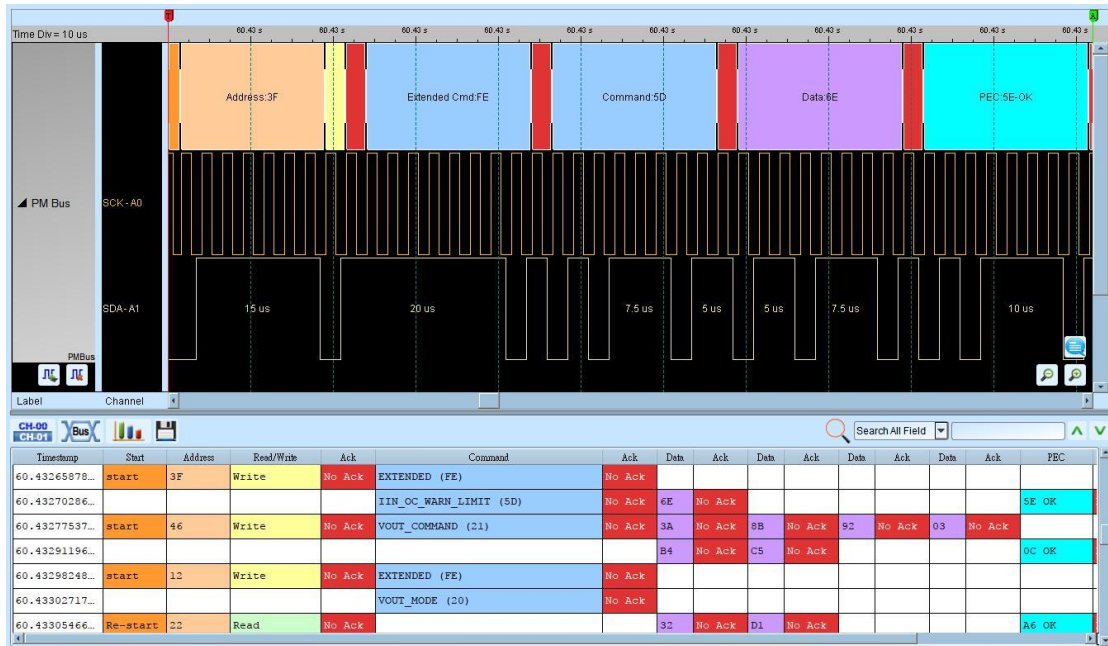
Decode With PEC: Group command protocol with PEC.

7-bit addressing (Include R/W in address): Show 8-bit addressing (include 7-bit addressing and 1-bit R/W).

Ignore glitch: Ignore the glitch when the slow transitions.

Result

Click **OK** to run the PMBus decode and see the result on the Waveform Window below.



ProfiBus

ProfiBus (PROcess Field Bus) is implemented by RS485. It includes PROFIBUS DP, PROFIBUS PA and PROFIBUS FMS.

Settings

Channel: Set the ProfiBus Channel

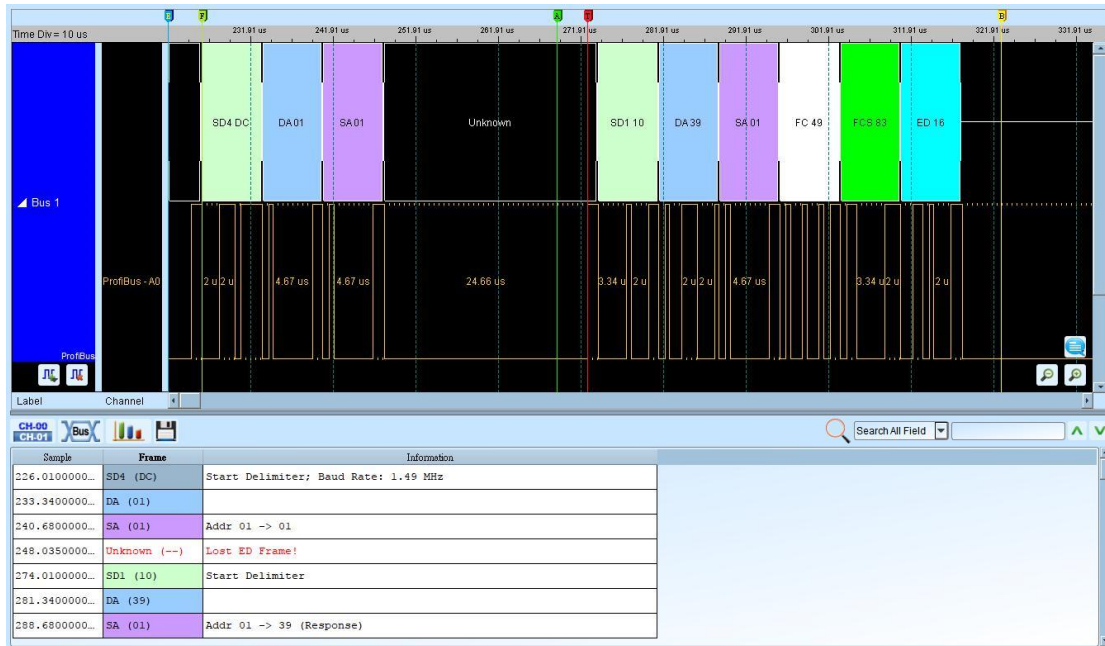
Polarity: Set the polarity Idle high / Idle low

Baud Rate/Auto Detect: Set the baud rate manually or auto detect

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

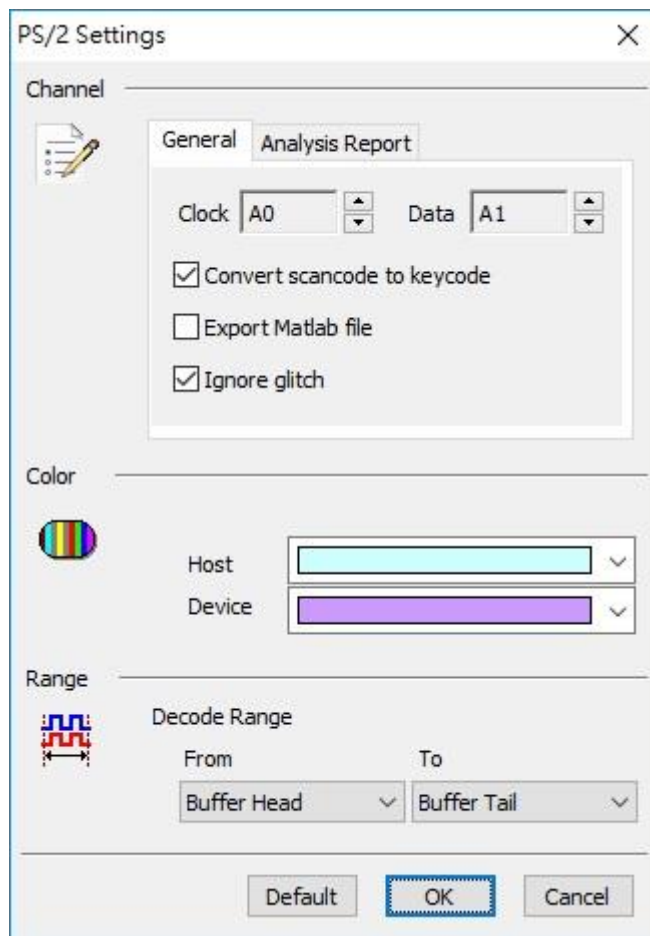
Result



PS/2

The Personal System/2 (PS/2) protocol has 6 data bits, but only the first bit (Data) and the fifth bit (Clock) need to be analyzed.

PS/2 Settings



Channel: Show the selected channels.

Convert scan code to key code: Transform data into keyboard characters.

Export MATLAB file: Export the data with MATLAB format as the following:

Time = [25.78484 25.785985 ...]

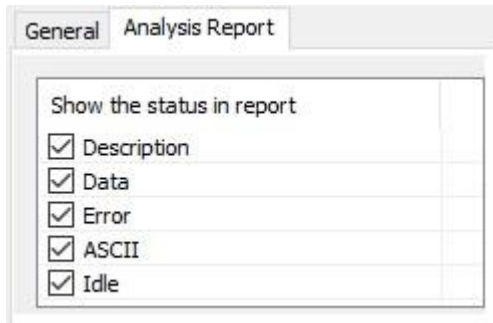
Description = [DH DH ...] DH = Device to Host, HD = Host to Device

Data = [58 FA 02 FA C4 ...]

The file (PS2_Matlab.m) will be saved at work directory

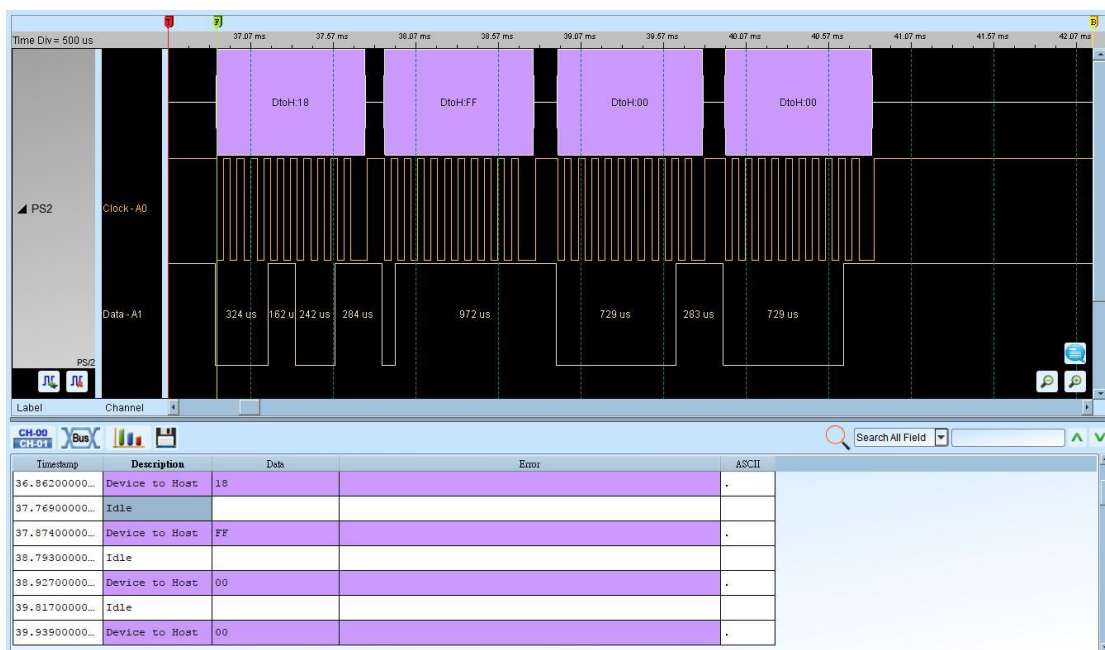
Ignore glitch: Ignore the glitch when the slow transitions.

Analysis Report: Show the selected status in report.



Result

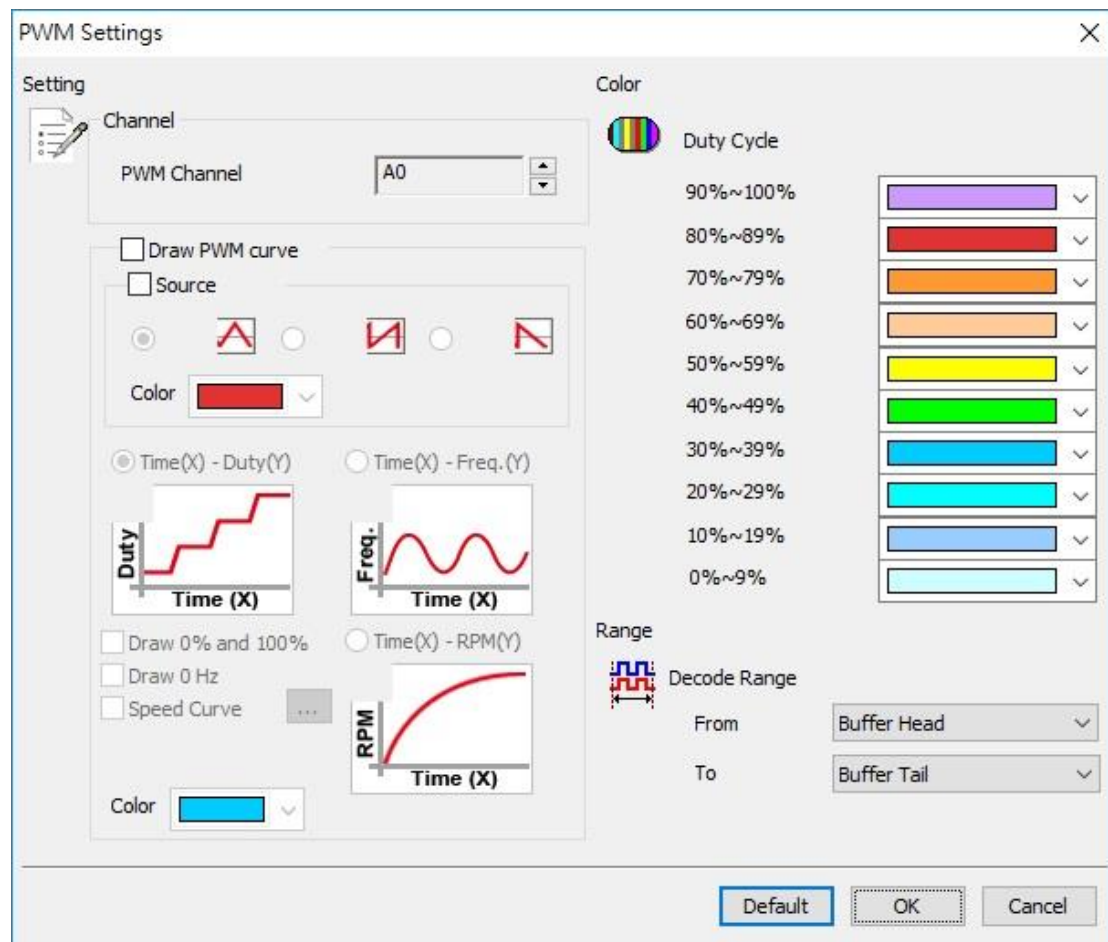
Click **OK** to run the PS/2 decode and see the result on the Waveform Window below.



PWM

Pulse-width modulation (PWM) is a commonly used technique for controlling power to inertial electrical devices, made practical by modern electronic power switches.

Settings



Channel: Show the selected channel.

Draw PWM curve:

Source: Show the source waveform of the PWM.

Time(X)-Duty(Y): Show the curve diagram with Time(X) and Duty(Y)

Time(X)-Freq.(Y): Show the curve diagram with Time(X) and Freq.(Y)

Time(X)-RPM(Y): Show the curve diagram with Time(X) and RPM(Y)

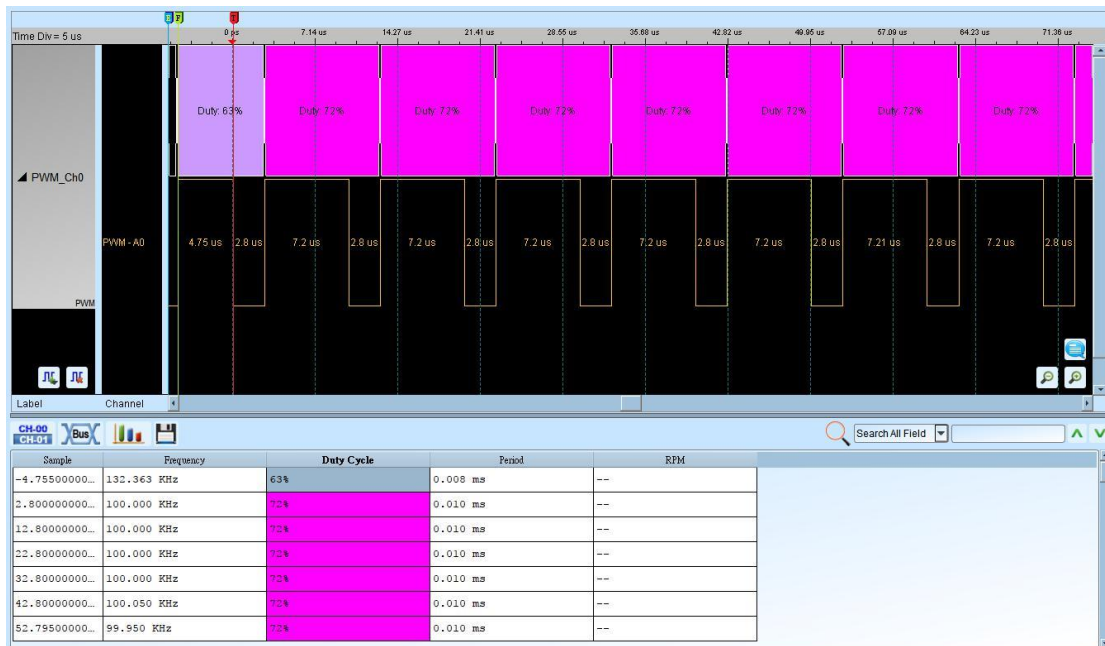
Draw 0% and 100%: When select the Time(X)-Duty(Y) drawing and check Draw 0% and 100%, the program will draw this duty curve of 0% or 100%; it will draw this duty curve of 0% or 100% when uncheck Draw 0% and 100%.

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.

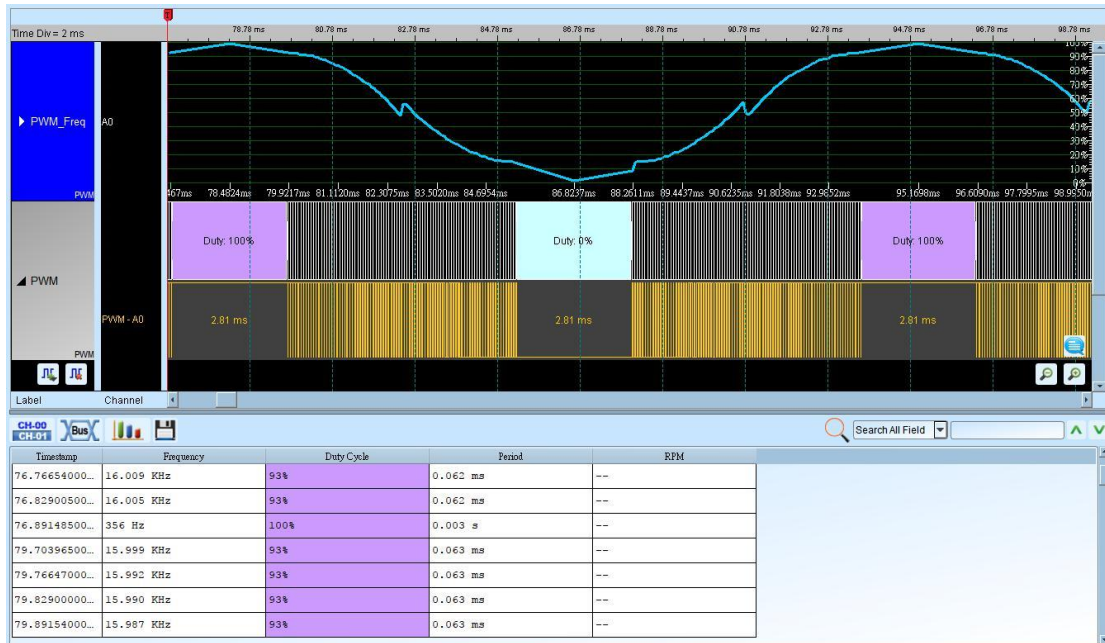
Result

Click **OK** to run the PWM decode and see the result on the Waveform Window below.

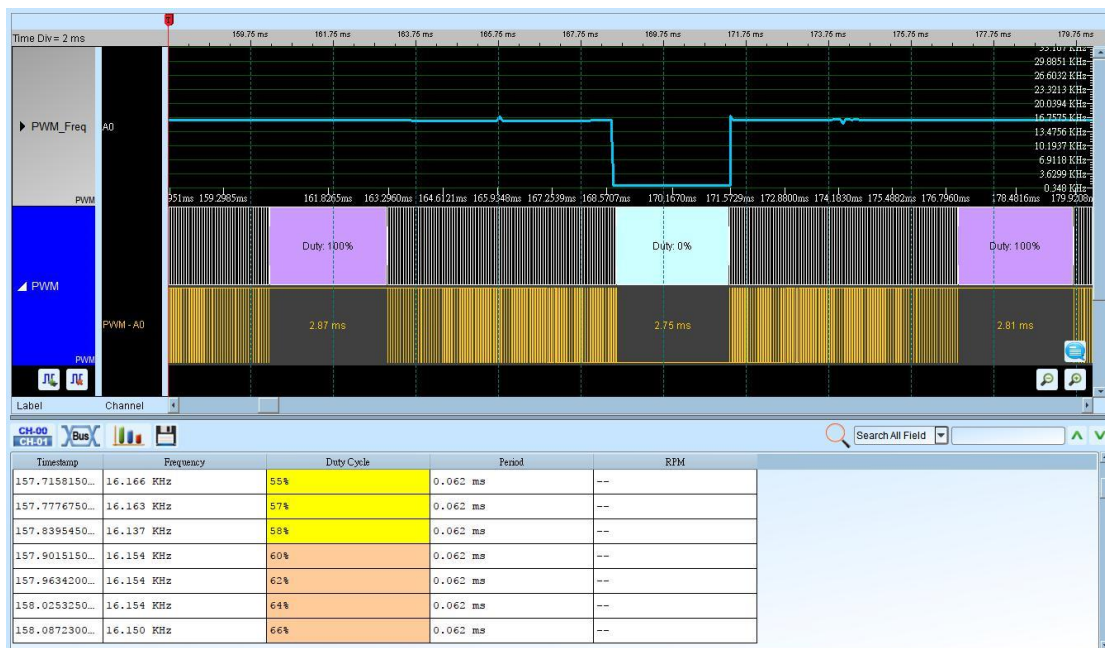
Select “Source”



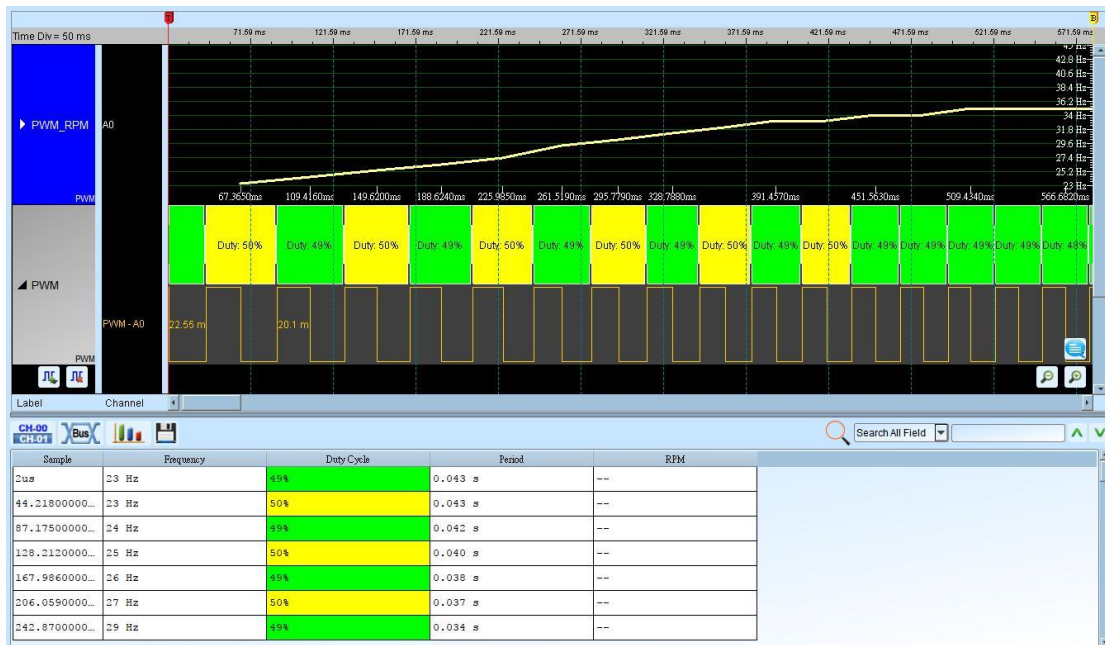
Select Time(X)-Duty(Y)



Select Time(X)-Freq.(Y)



Select Time(X)-RPM(Y)



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

Parameter Setting

QI Channel: A0 Advanced Decode

Color

Preamble: [Orange] Start: [Blue]

Head: [Light Orange] Parity: [Purple]

Message: [Yellow] Stop: [White]

CheckSum: [Green]

Range

Decode Range

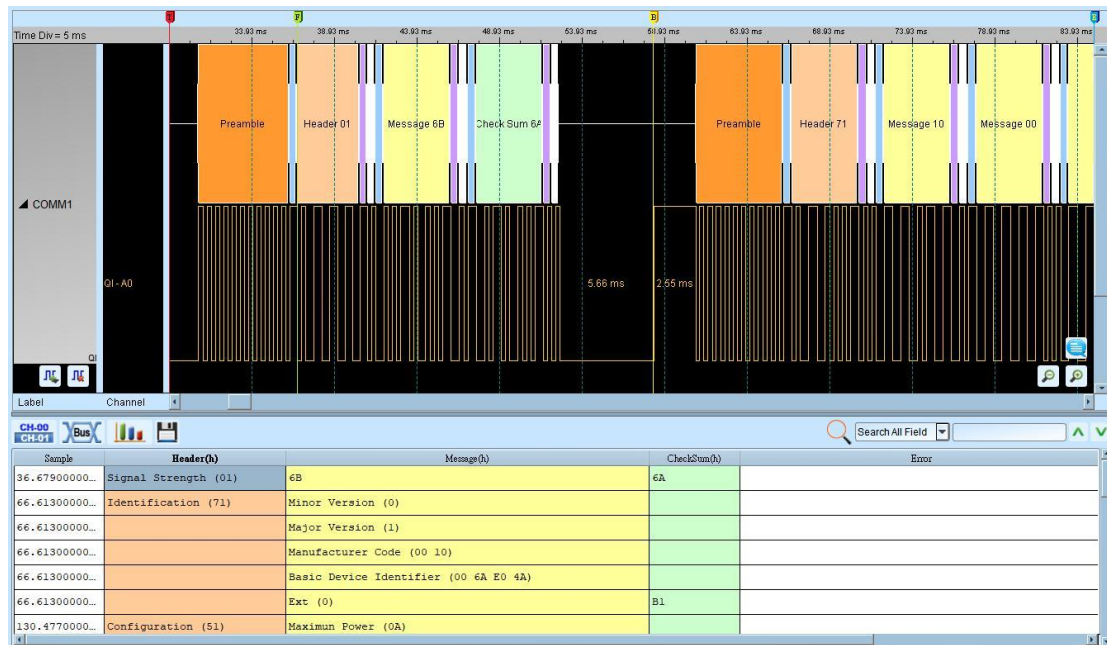
From: Buffer Head To: Buffer Head

Default OK Cancel

QI Channel: Show the selected channel.

Advance Decode: show detail message decode

Result



QSPI

QSPI is an enhanced version of SPI, which strengthens the transmission volume in DATA; The data line of QSPI is bidirectional and belongs to parallel transmission

Parameters

Channel:

CS: Capture data when CS edge is falling.

CLK: Clock channel.

D0-D7: Customizable data channel.

Significant Bit(D0): D0 is the MSB or LSB of the data arrangement,

1. Take Bus Width = 4 MSB as an example, the Byte combination is D0 D1 D2 D3

D0 D1 D2 D3

2. Take Bus Width = 4 LSB as an example, the Byte combination is D3 D2 D1 D0 D3

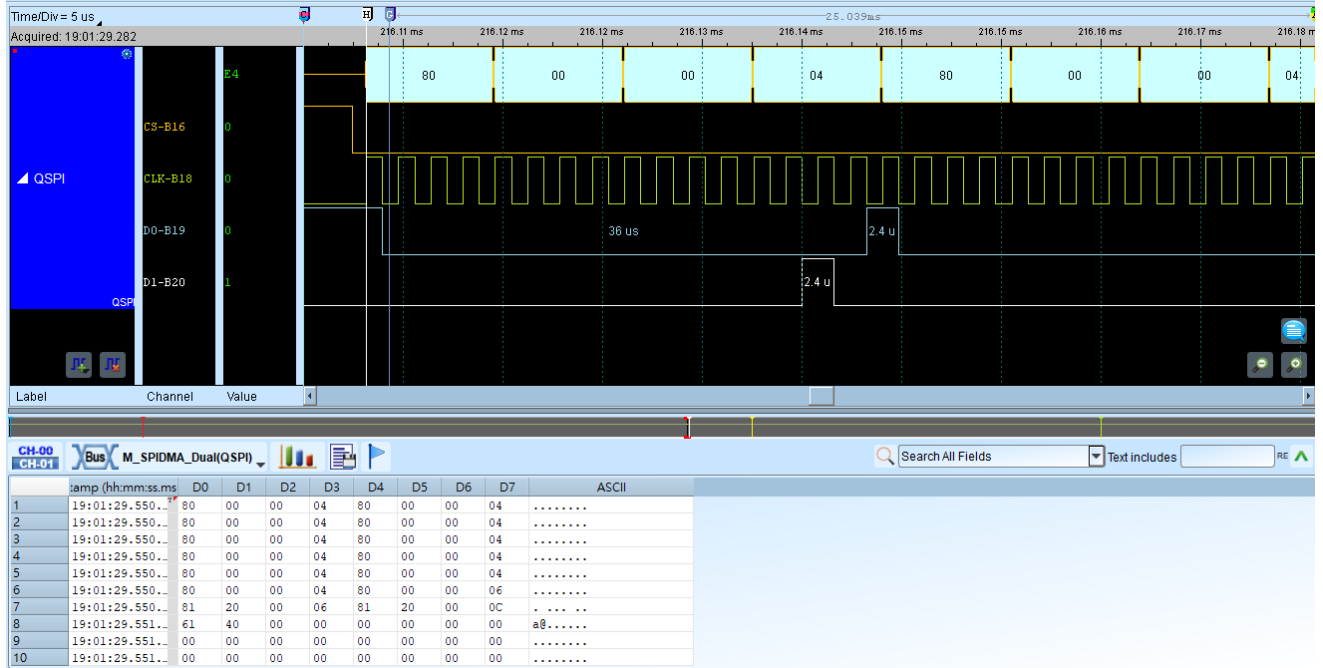
D2 D1 D0

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

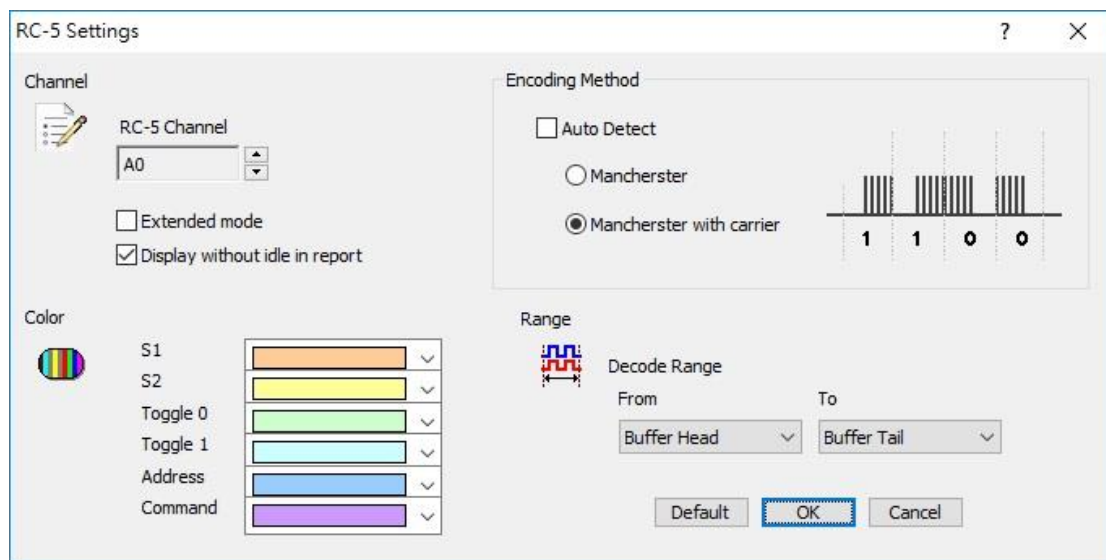
Result



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



Channel: Show the selected channel (CH 0).

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.

Display without idle in report: It will not idle on the Report Window for the user to observe and analyze data.

Encoding Method: Auto Detect mode, Manchester mode and Manchester with carrier mode.

S1/S2: Start bit.

Toggle 0/Toggle 1: The difference is that has been used while sending the message to

repeat, or send a new message.

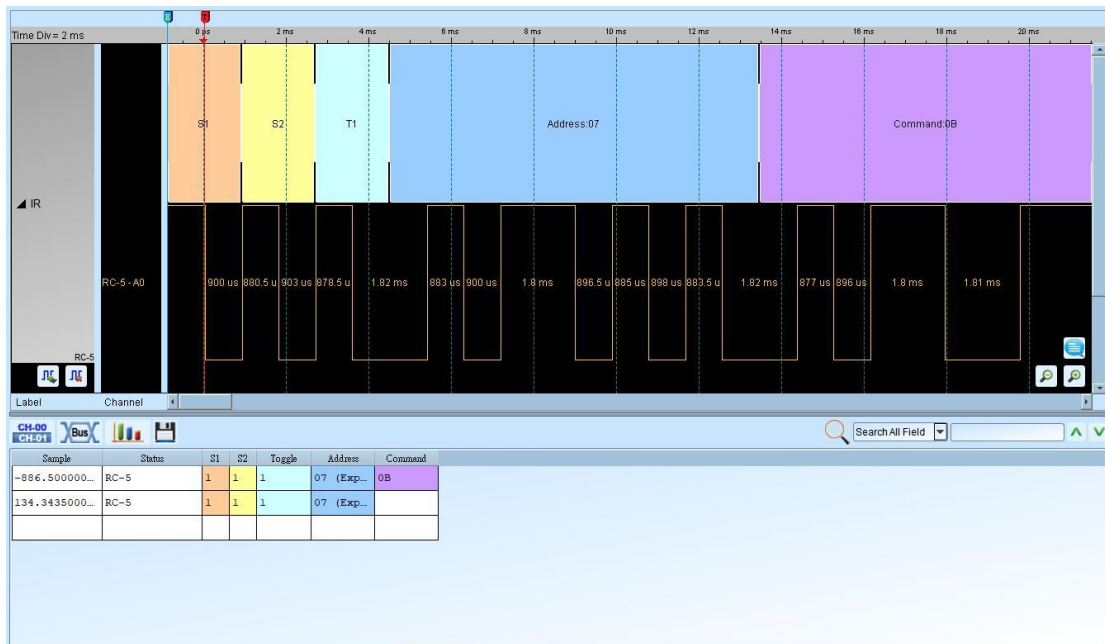
Address: To represent different device addresses.

Command: To represent the different button commands.

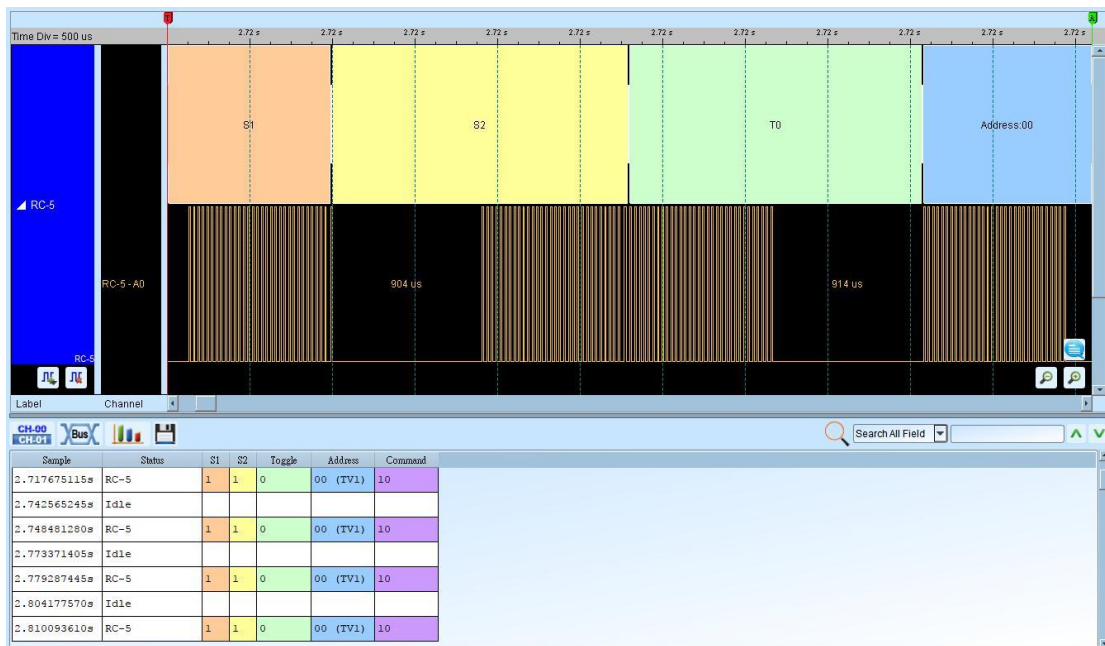
Result

Click **OK** to run the RC-5 decode and see the result on the Waveform Window below.

RC5 without carrier



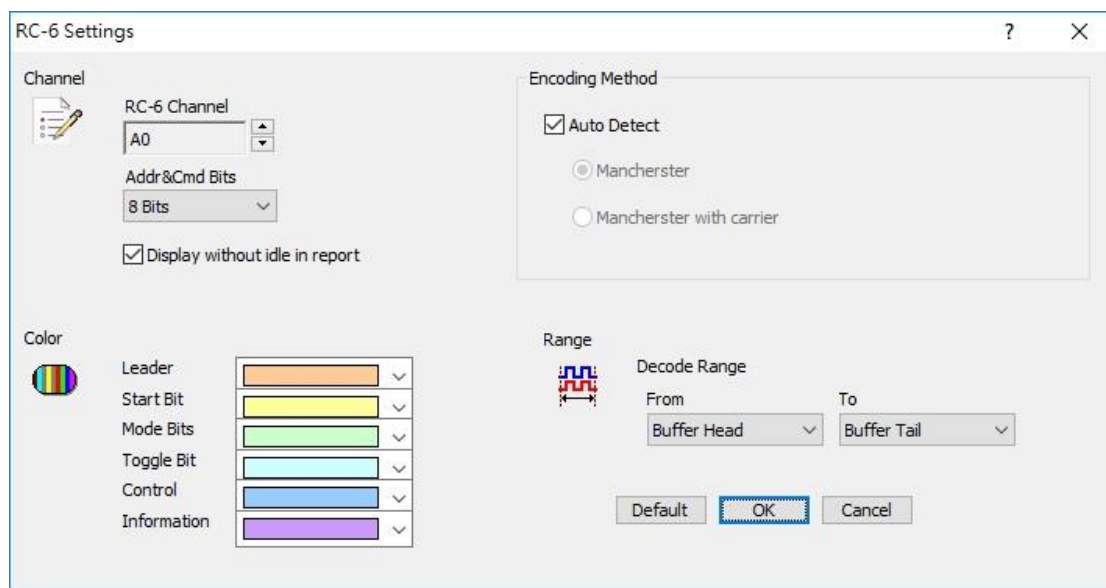
RC5 with carrier



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



Channel: Show the selected channel (CH 0).

Add & Cmd Bits: Show commands in 8 bits or 16 bits of address and information in the control label.

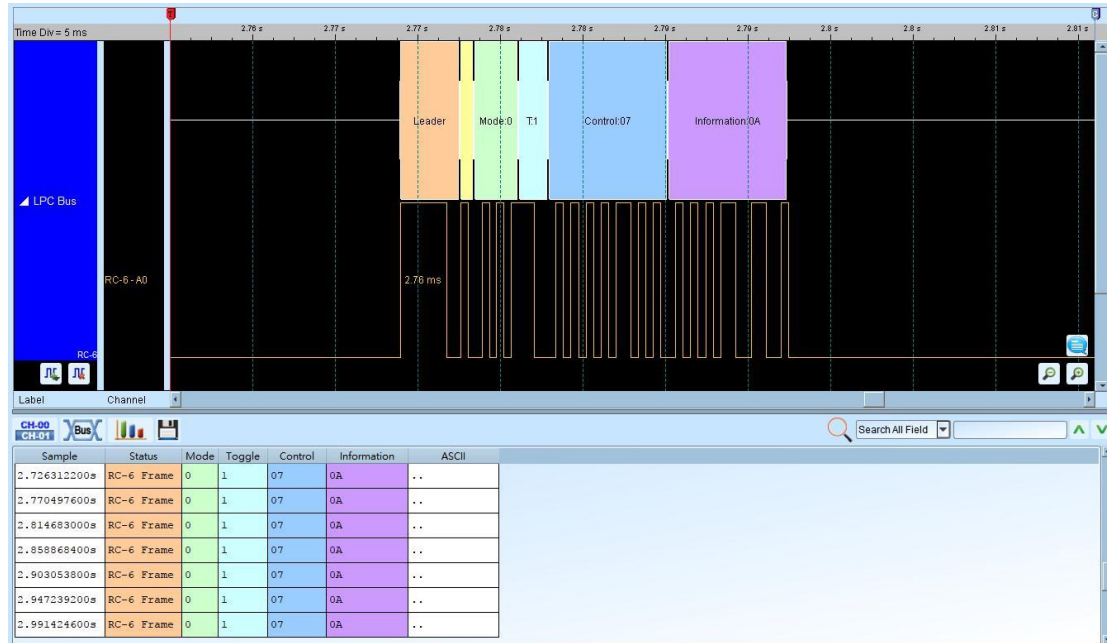
Display without idle in report: Do not display any idle in the Report Window.

Encoding Method: Auto detect mode, Manchester mode, Manchester with carrier mode.

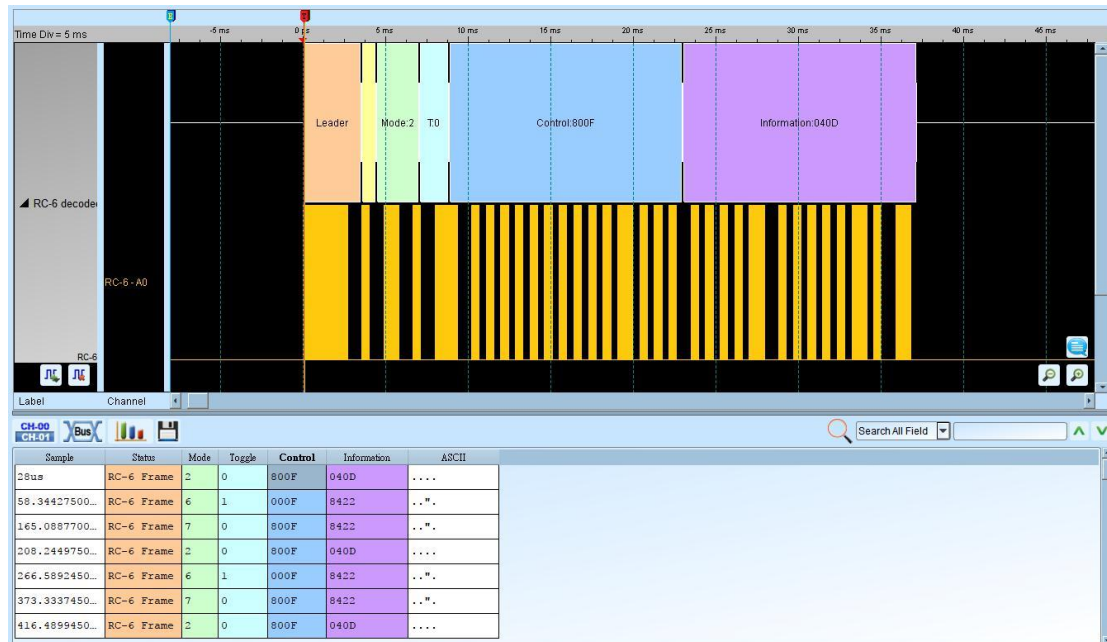
Result

Click **OK** to run the RC-6 decode and see the result on the Waveform Window below.

RC6 without carrier



RC6 with carrier



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

RGB8888 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

Decoded Range

From Buffer Head To Buffer Tail

Default OK Cancel

SCLK: The Clock pin.

DE: The Data Enable pin.

Hsync: The Horizontal synchronization pin.

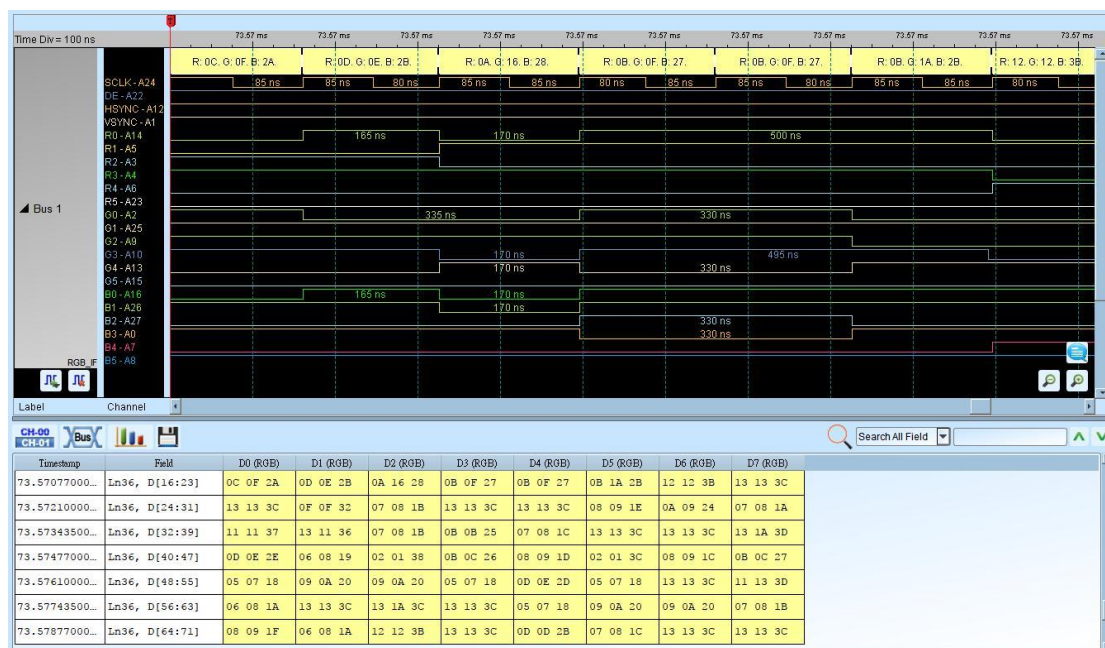
Vsync: The Vertical synchronization pin.

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



S/PDIF

The Sony/Philips Digital Interconnect Format (S/PDIF) is a digital audio transmission interface with the detailed specifications below:

Data format: Default is 16 bits, up to 24 bits.

Sampling frequency:

44.1Khz from CD → Bit Rate 2.8224 Mbit/s

48 Khz from DAT →Bit Rate 3.072 Mbit/s

32 Khz from DSR → Bit Rate 2.048 Mbit/s

Deliver method: One way.

V (Validity) bit: Audio samples to confirm effectiveness, if this bit is 0, the receiver should ignore this sub-frame.

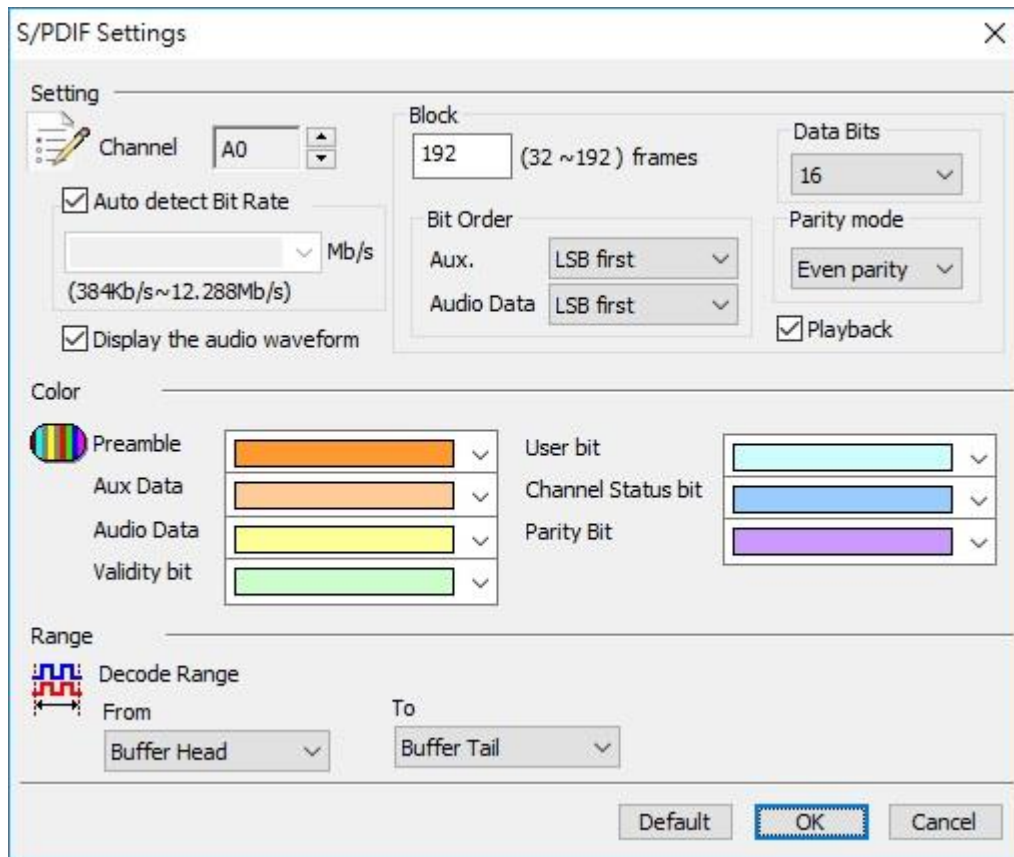
U (User) bit: User log information.

C (Channel status) bit: Channel state information.

P (Parity) bit: Parity bit check for the error.

The basic principle is to split the data bits into two parts. If the data is 1, split it into 01 or 10 or if the data is 0, split it into 00 or 11.

Settings



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.

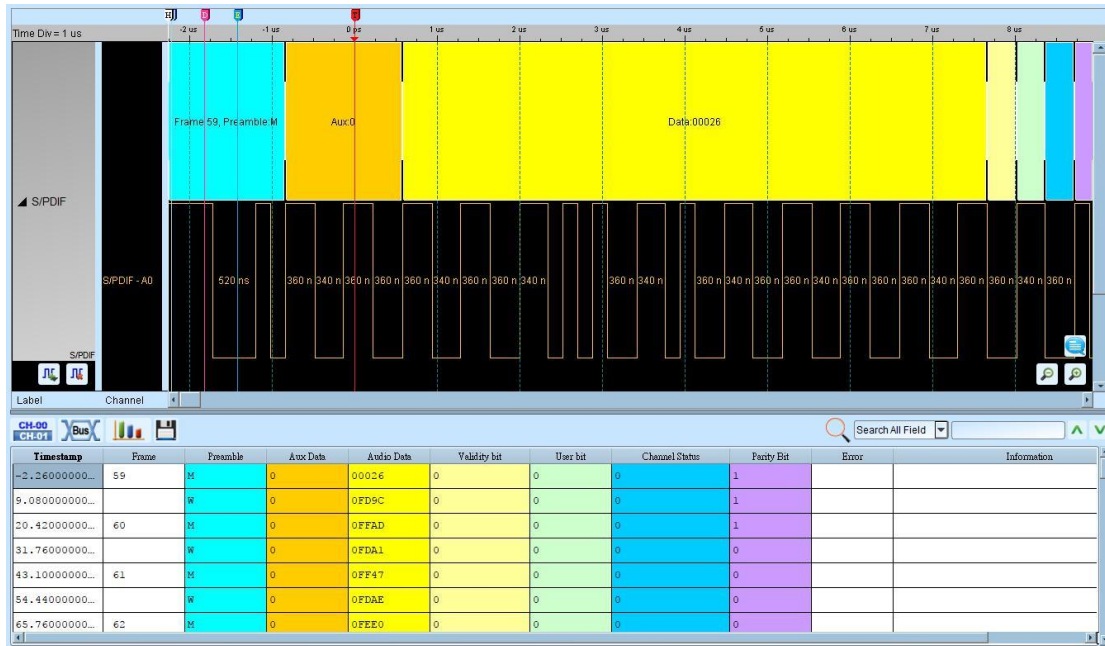
Parity mode: The default is even parity.

Display the audio waveform: Click to display the audio waveform in the Waveform Window.

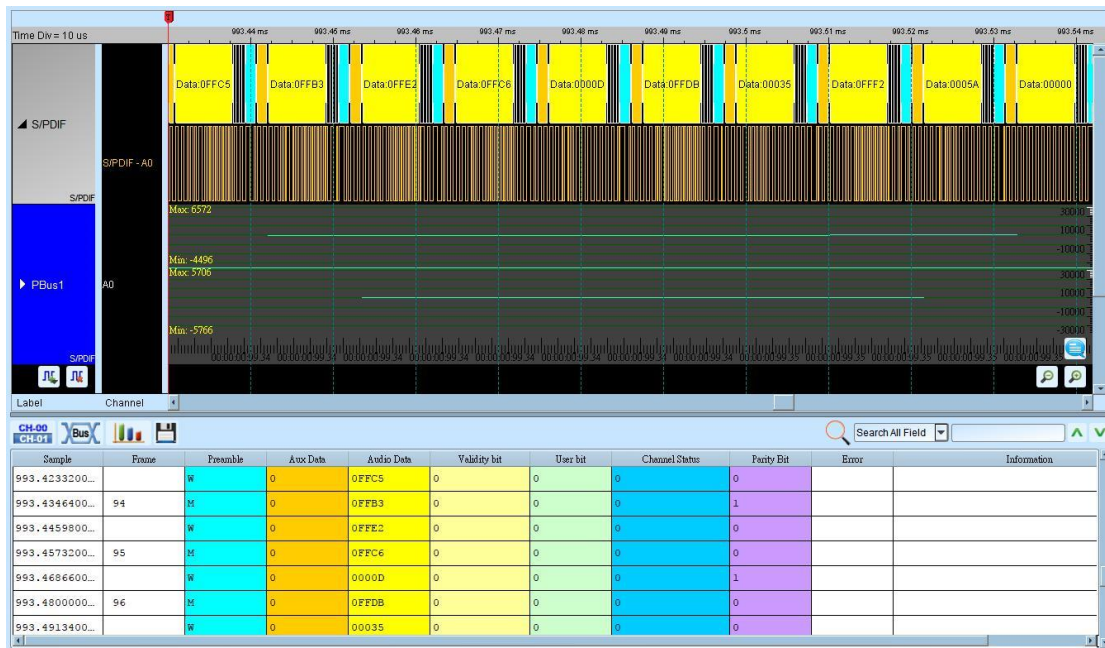
Result

Click **OK** to run the S/PDIF Decode and see the result on the Waveform Window

below.



Show wave:



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

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.

SPI mode: Analysis SD data with SPI mode ◦

Always Detect CRC Detect CRC in SPI mode

Command:

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.

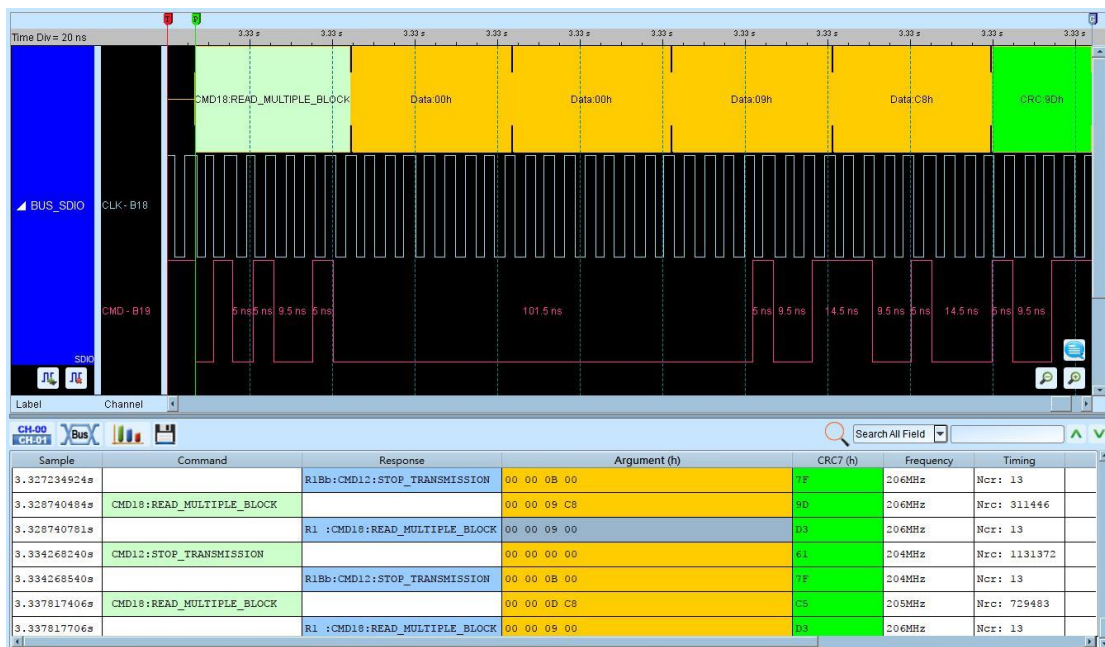
Auto Phase Correction:

If it's checked, the decoder will auto adjust the phase of LA when measure the DUT..

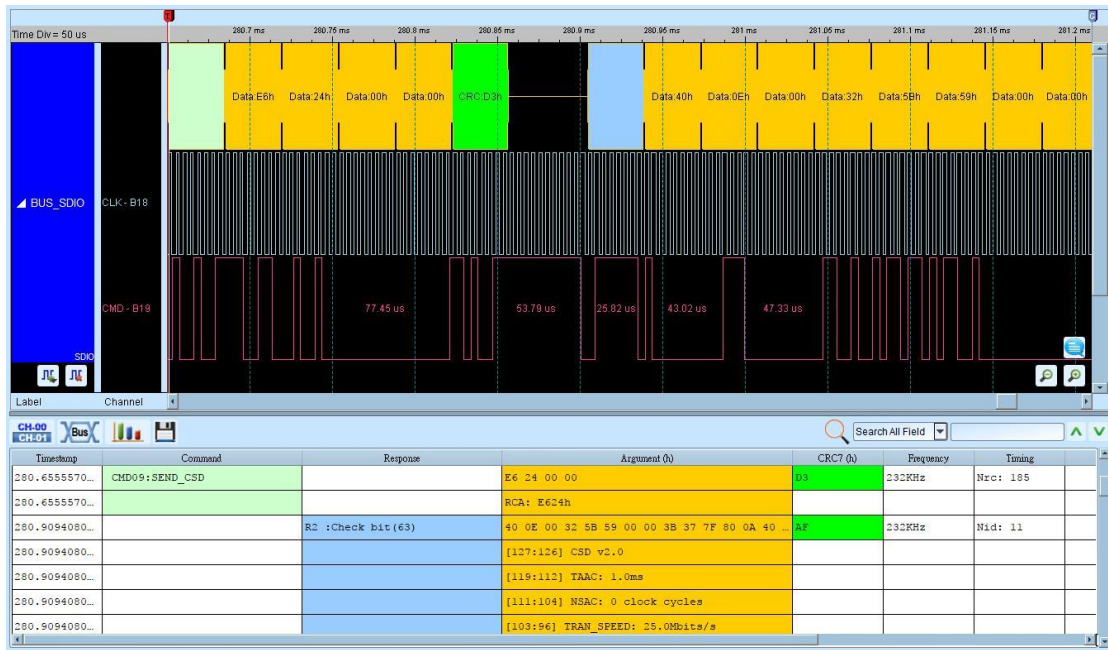
Result

Click **OK** to run the SDI/O decode and see the result on the Waveform Window below.

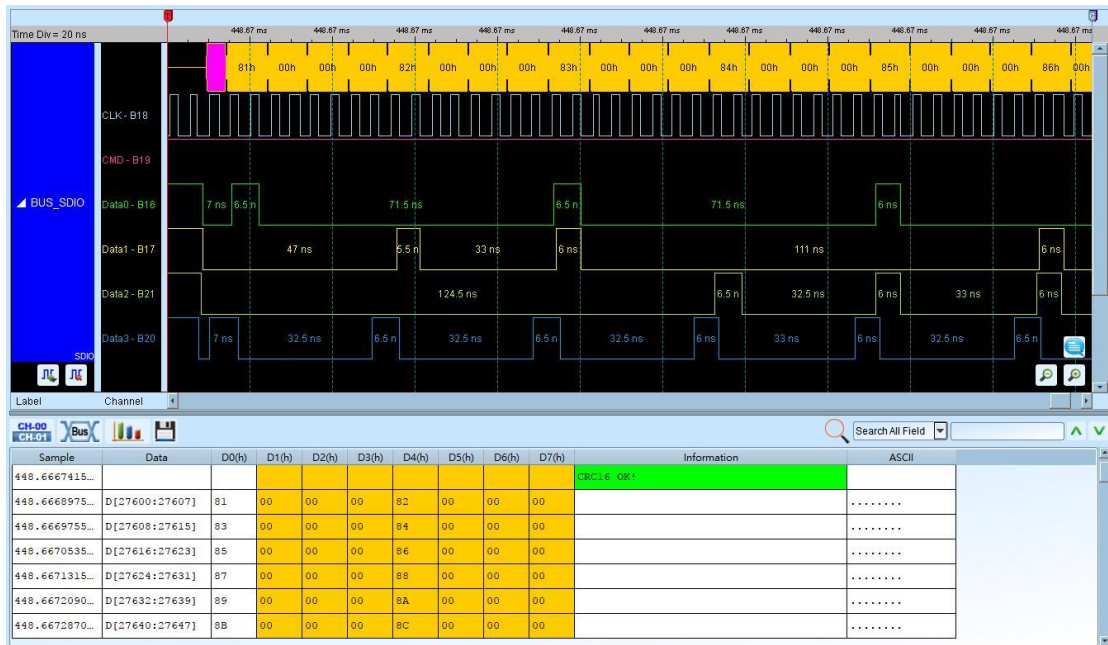
CMD mode



Adv. Report



Data mode



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+ 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

- 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

DQ0: A24, DQ8: B0, DQ16: B8, DQ24: B16, DQ32: B24

DQ1: A25, DQ9: B1, DQ17: B9, DQ25: B17, DQ33: B25

DQ2: A26, DQ10: B2, DQ18: B10, DQ26: B18, DQ34: B26

DQ3: A27, DQ11: B3, DQ19: B11, DQ27: B19, DQ35: B27

DQ4: A28, DQ12: B4, DQ20: B12, DQ28: B20, DQ36: B28

DQ5: A29, DQ13: B5, DQ21: B13, DQ29: B21, DQ37: B29

DQ6: A30, DQ14: B6, DQ22: B14, DQ30: B22, DQ38: B30

DQ7: A31, DQ15: B7, DQ23: B15, DQ31: B23, DQ39: B31

Parsing include Address and Data

Startup

- #CAS Latency
 - Non
 - 3 clocks
 - 2 clocks

Decode display in waveform area

- Command
- Address
- Data
- Bank Address
- A10

Color

- DESL: [Color]
- NOP: [Color]
- BST: [Color]
- READ/A: [Color]
- WRIT/A: [Color]
- ACT: [Color]
- PRE: [Color]
- CBR_AREF: [Color]
- MRS: [Color]
- PALL: [Color]
- SELF: [Color]
- Address: [Color]
- Data: [Color]

Range

Decode Range

From: Buffer Head To: Buffer Tail

Default OK Cancel

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

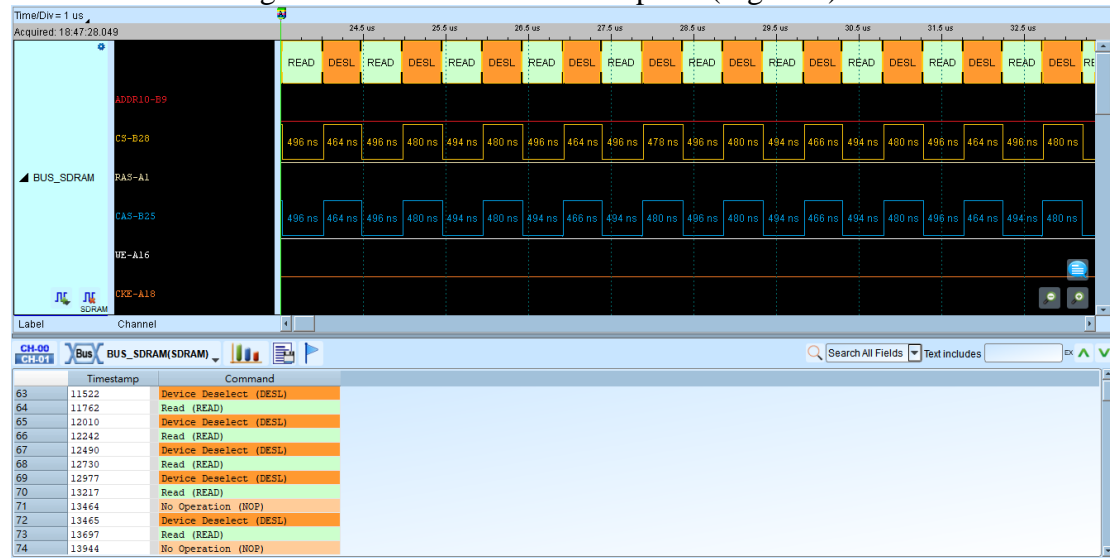
Because there are many states to be displayed by SDRAM, they cannot be displayed all at once in the waveform area.

Therefore, it is necessary to select the item to be viewed in the waveform area.

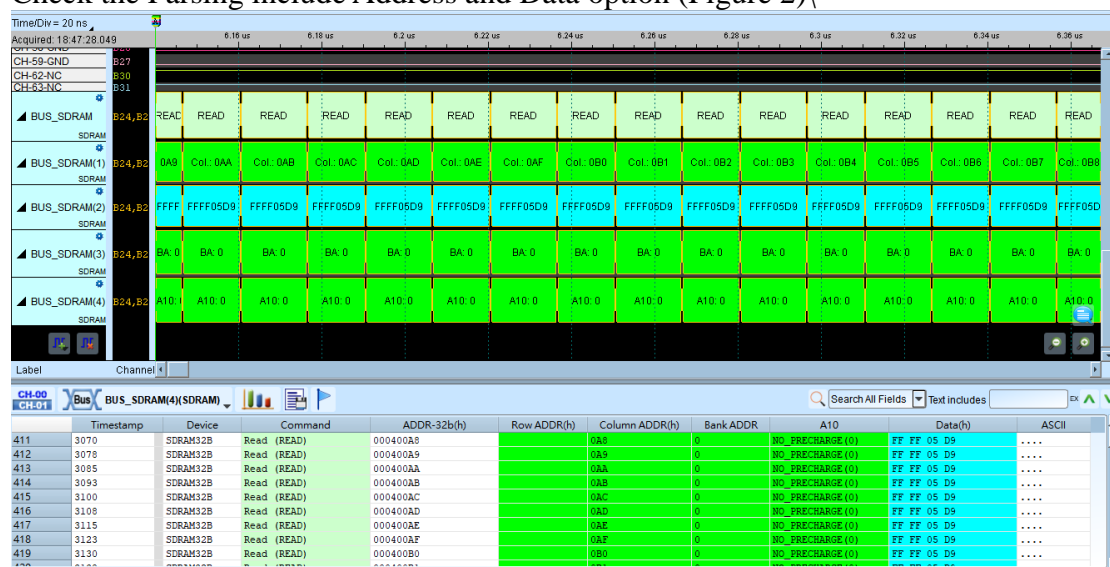
If there are many items to be viewed at the same time, you can add multiple groups of the same SDRAM decoding, then set different decoding and display modes of the waveform area respectively. As shown in Figure 2 below, the left side multiple groups of channels have been added to distinguish the display.

Result

Uncheck the Parsing include Address and Data option (Figure 1)



Check the Parsing include Address and Data option (Figure 2)



Serial Flash

SPI Serial Flash is small, low-power flash memory that features Serial Peripheral Interface (SPI) and pin-for-pin compatibility with industry-standard SPI EEPROM devices.

Settings

Channel: Show the selected channels (CH0 – CH5).

Manufacturer/Device: Select the Serial Flash device type, tCLQV and tSHSL.

QPI mode: Quad Peripheral Interface Mode/Quad SPI Mode

4-Byte mode: 4-Byte Address Mode

PEM mode: Performance Enhance Mode

Dummy Cycles: Clock buffers between read command and data.

Wrap Around: Wrap number

QE bit: Enable or disable the QPI mode.

Decode SI Only: Single mode, 3-wire → CS#, SCLK, SI.

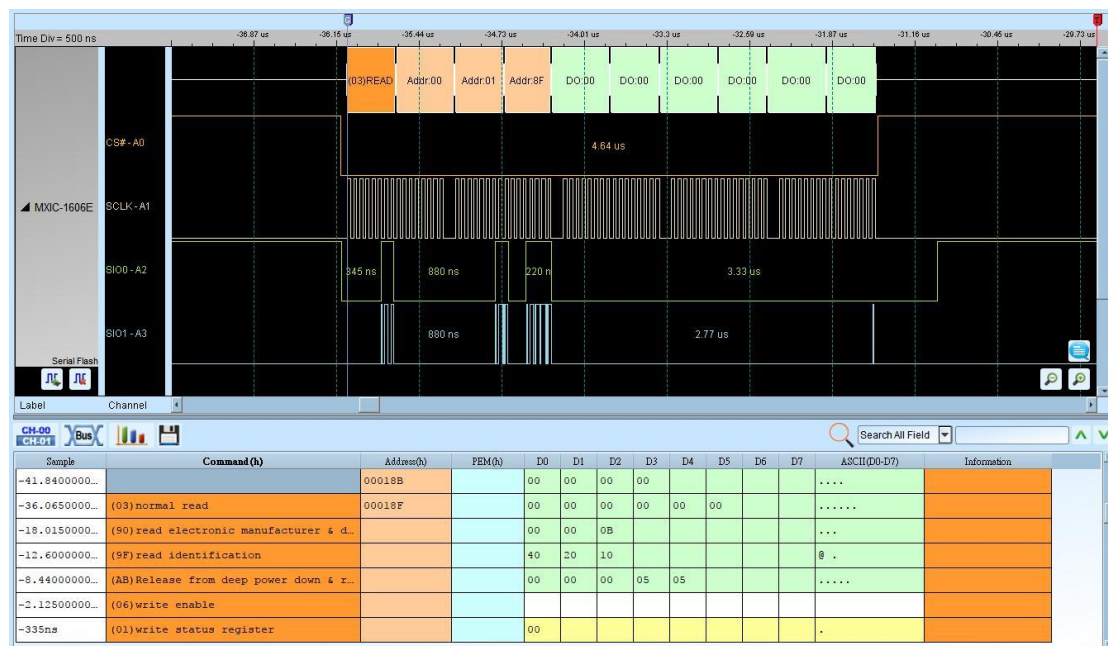
Decode Single Mode Only: Single mode, 4-wire → CS, Clock, SI, SO.

The LA viewer will choose 4-wire or 6-wire to analyze according to the Serial Flash device type.

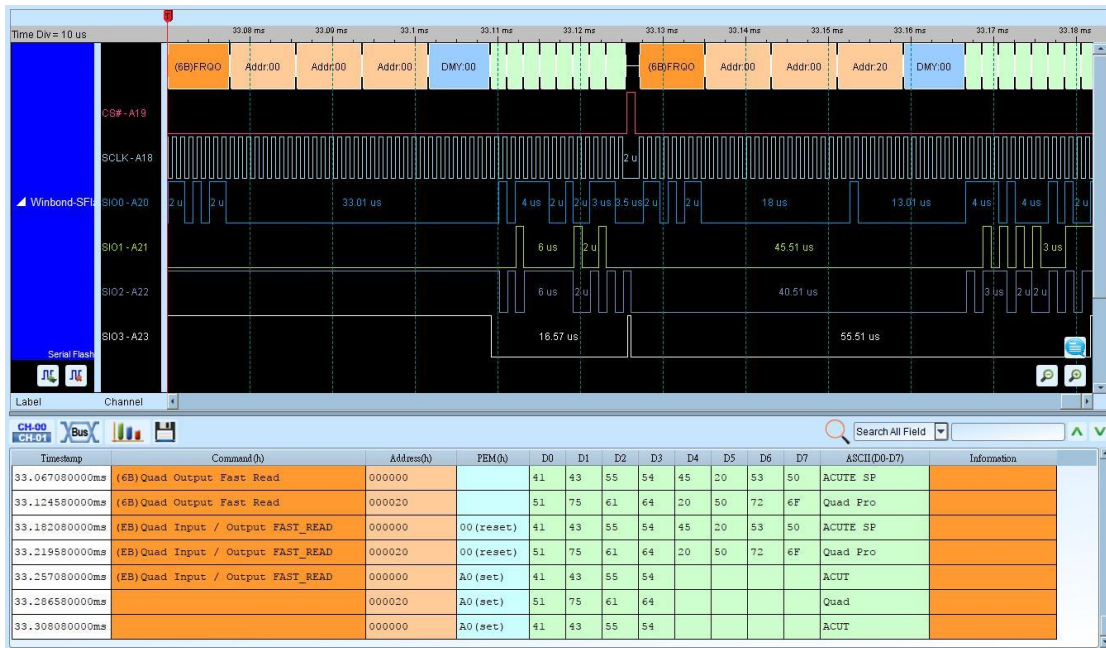
Result

Click **OK** to run the Serial Flash decode and see the result on the Waveform Window below.

Serial Flash decode within SPI mode.

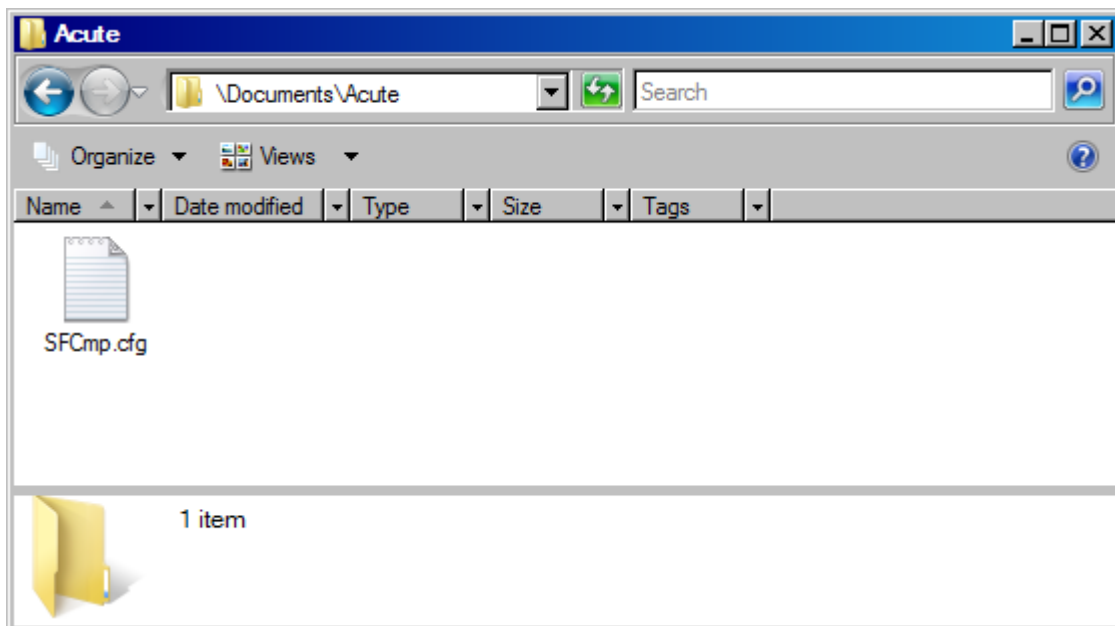


Serial Flash decode within QPI mode.

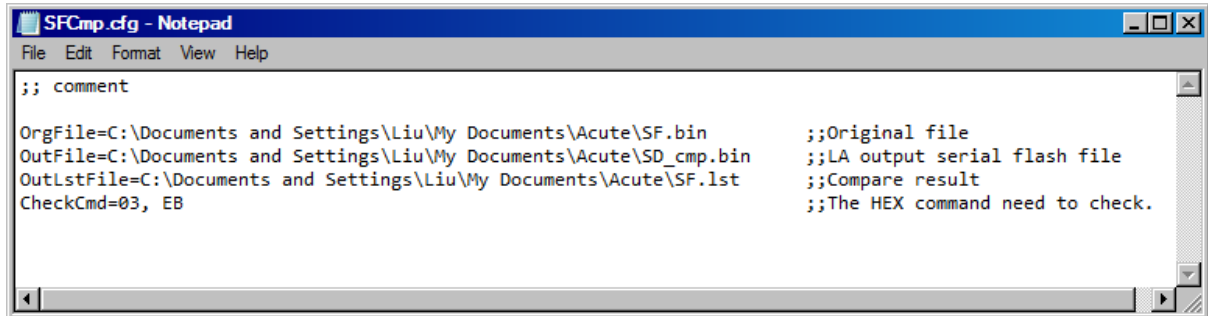


Serial Flash data Comparison : Compare the Serial Flash data by the waveform files.

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.
```

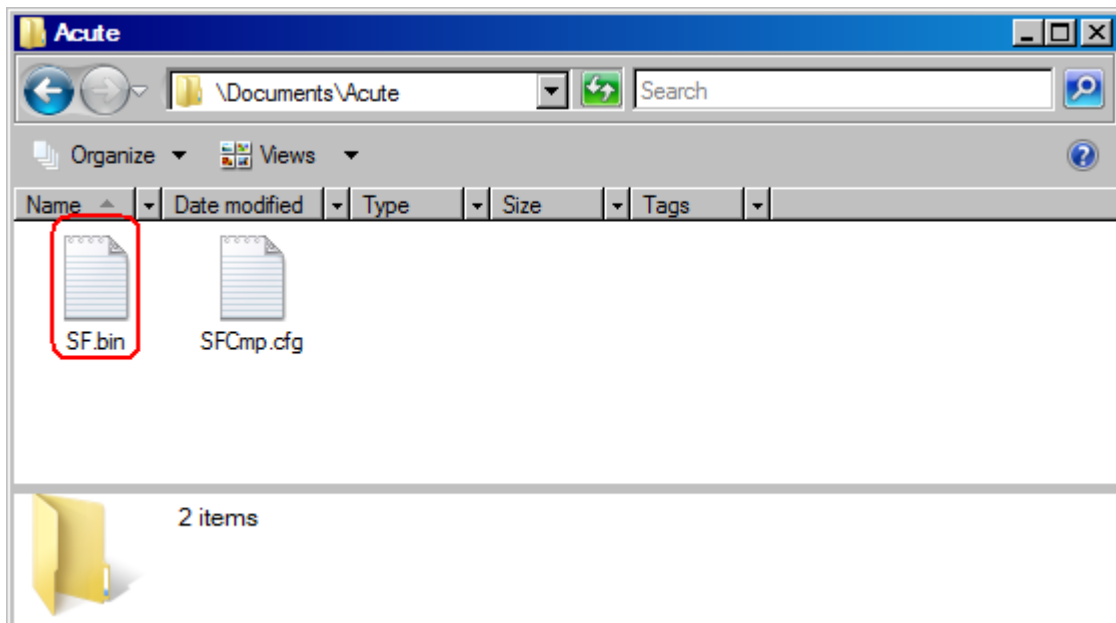
OrgFile=File_Path: Key in the file path of the original Serial Flash data file (.bin).

OutFile=File_Path: Key in the file path of the Serial Flash output file.

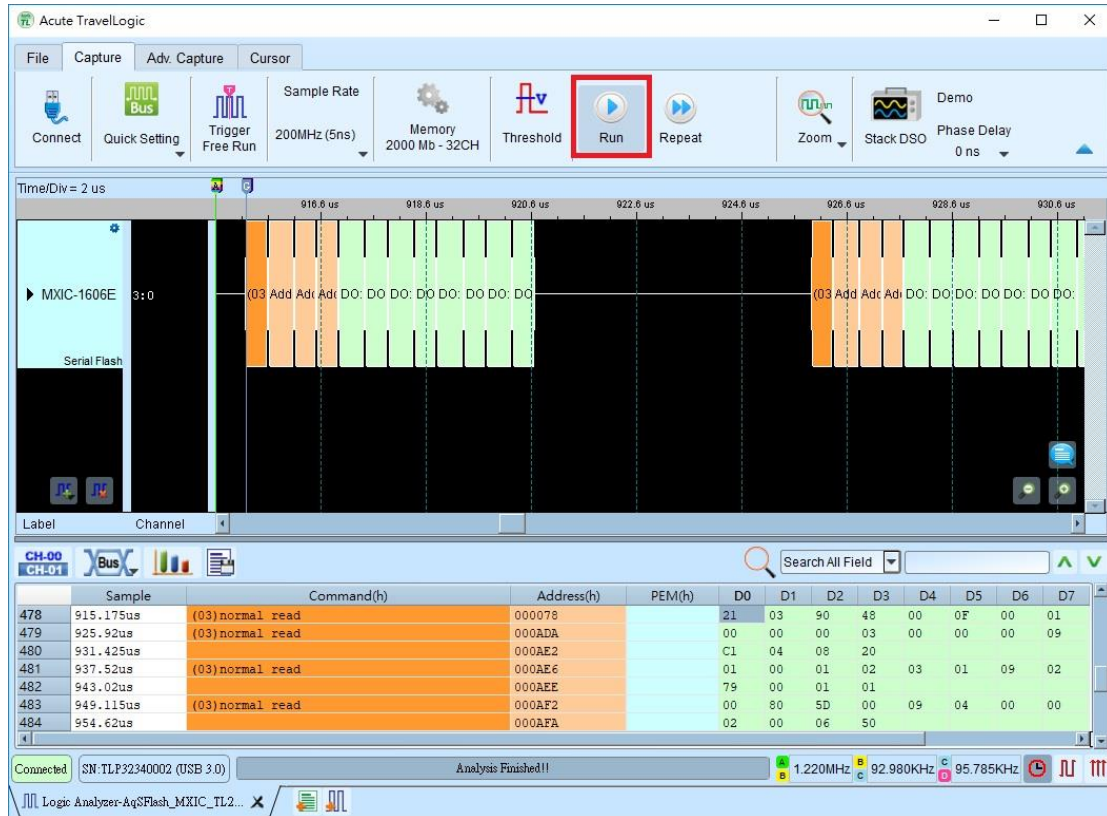
OutLstFile=File_Path: Key in the file path of the comparison result. The file name will has extension “.lst”.

CheckCmd=Serial Flash command: Key in the command in Hex that are separated by commas.

Save the OrgFile to the OrgFile file path.

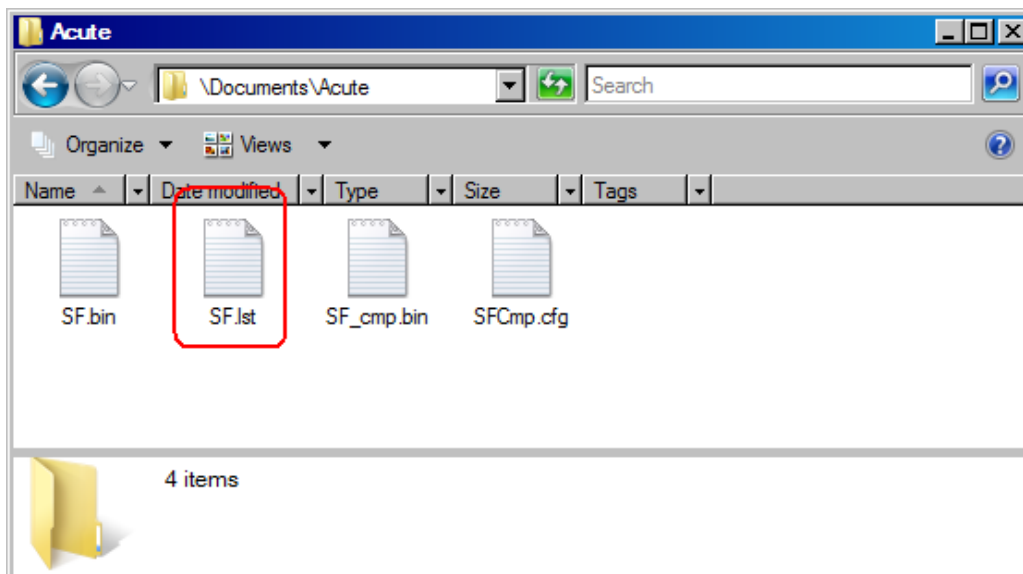


Run the Serial Flash Bus Decode to capture the Serial Flash signal.

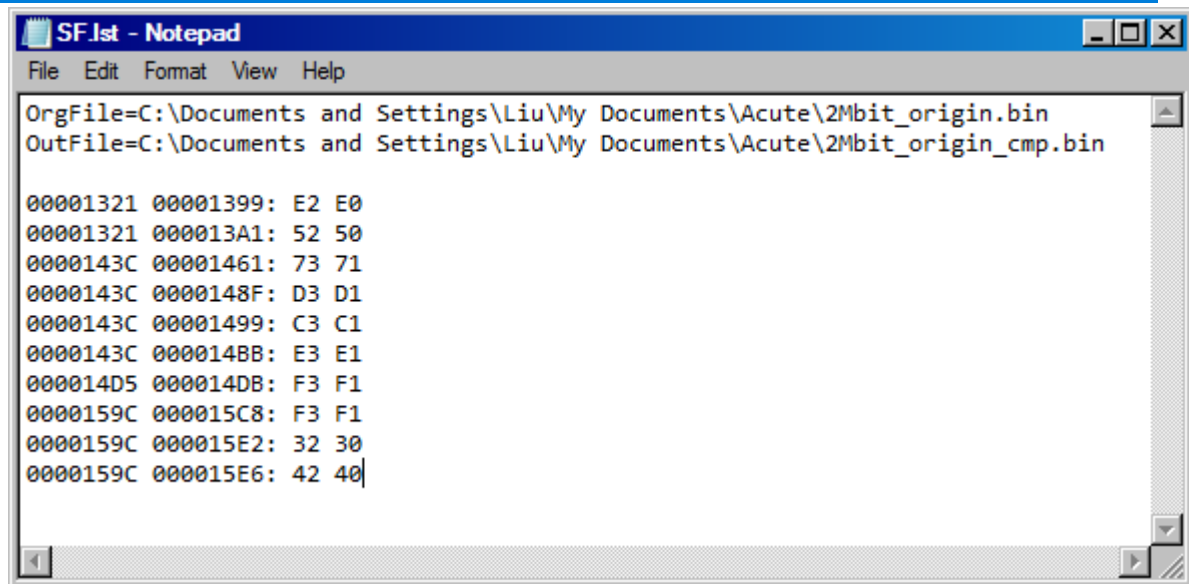


If the OutFile does not exist, it will copy the OrgFile to the OutFile and write the data to it according to the CheckCmd.

Compare result:



The OutLstFile:



```
SF.lst - Notepad
File Edit Format View Help
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

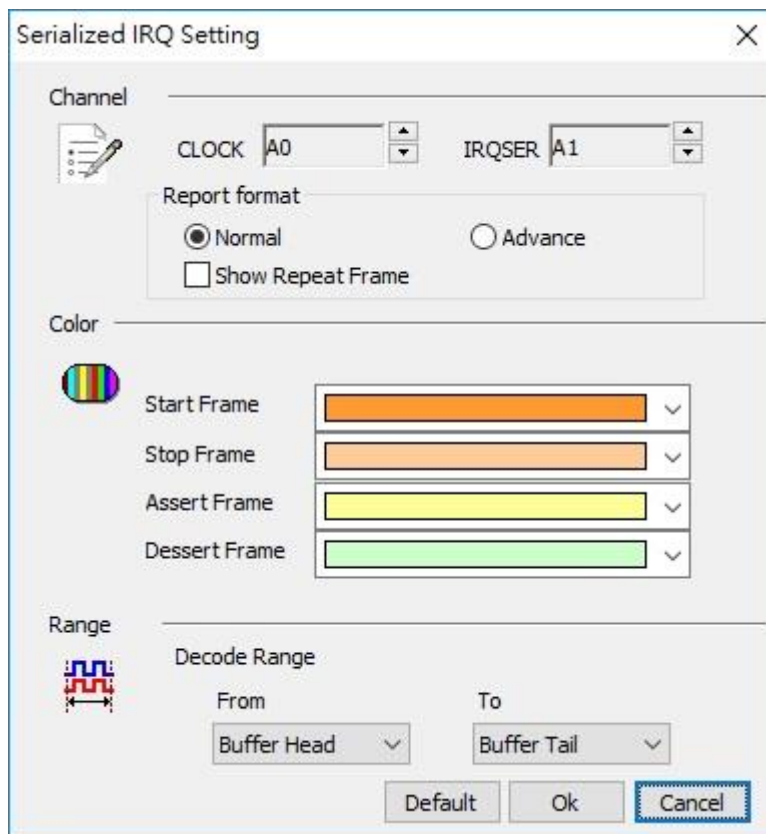
00001321 00001399: E2 E0
00001321 000013A1: 52 50
0000143C 00001461: 73 71
0000143C 0000148F: D3 D1
0000143C 00001499: C3 C1
0000143C 000014BB: E3 E1
000014D5 000014DB: F3 F1
0000159C 000015C8: F3 F1
0000159C 000015E2: 32 30
0000159C 000015E6: 42 40
```

The first column is the compared address from OrgFile, the second column is the different address from OutFile.

Serial IRQ

The IRQ/Data serializer is a Wired-OR structure that simply passes the state of one or more device's IRQ(s) and/or Data to the host controller. The transfer can be initiated by either a device or the host controller. A transfer, called an IRQSER Cycle, consists of three frame types: one Start Frame, several IRQ/Data Frames, and one Stop Frame. This protocol uses the PCI Clock as its clock source and conforms to the PCI bus electrical specification.

Settings



CLOCK: PCI Clock channel

IRQSER; IRQSER channel

Normal: Not show repeat frame

Clock	No.	Mode	0	1	SMI	3	4	5	6	7	8	9	10	11	12	13	14	15	IRQ
10286946	22830	Continue mode		A					A						A				
10291906	22841	Continue mode		A											A				
10404663	23091	Continue mode		A					A						A				
10415039	23114	Continue mode		A											A				
10459240	23212	Continue mode		A					A						A				
10461943	23218	Continue mode		A											A				
10580112	23480	Continue mode		A					A						A				
10590037	23502	Continue mode		A											A				
10634238	23600	Continue mode		A					A						A				
10636941	23606	Continue mode		A											A				

Show repeat frame

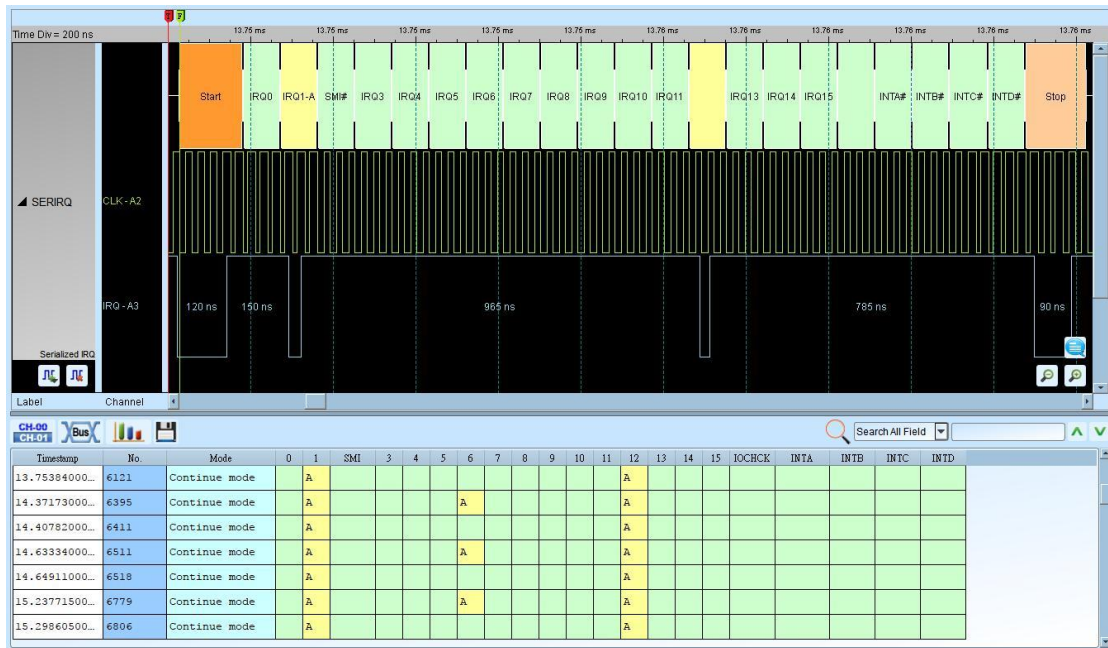
Clock	No.	Mode	0	1	SMI	3	4	5	6	7	8	9	10	11	12	13	14	15	IRQ
10457887	23209	Continue mode		A											A				
10458338	23210	Continue mode		A											A				
10458789	23211	Continue mode		A											A				
10459240	23212	Continue mode		A					A						A				
10459690	23213	Continue mode		A					A						A				
10460140	23214	Continue mode		A					A						A				
10460590	23215	Continue mode		A					A						A				
10461041	23216	Continue mode		A					A						A				
10461492	23217	Continue mode		A					A						A				
10461943	23218	Continue mode		A											A				
10462394	23219	Continue mode		A											A				
10462846	23220	Continue mode		A											A				
10463298	23221	Continue mode		A											A				

Advance:

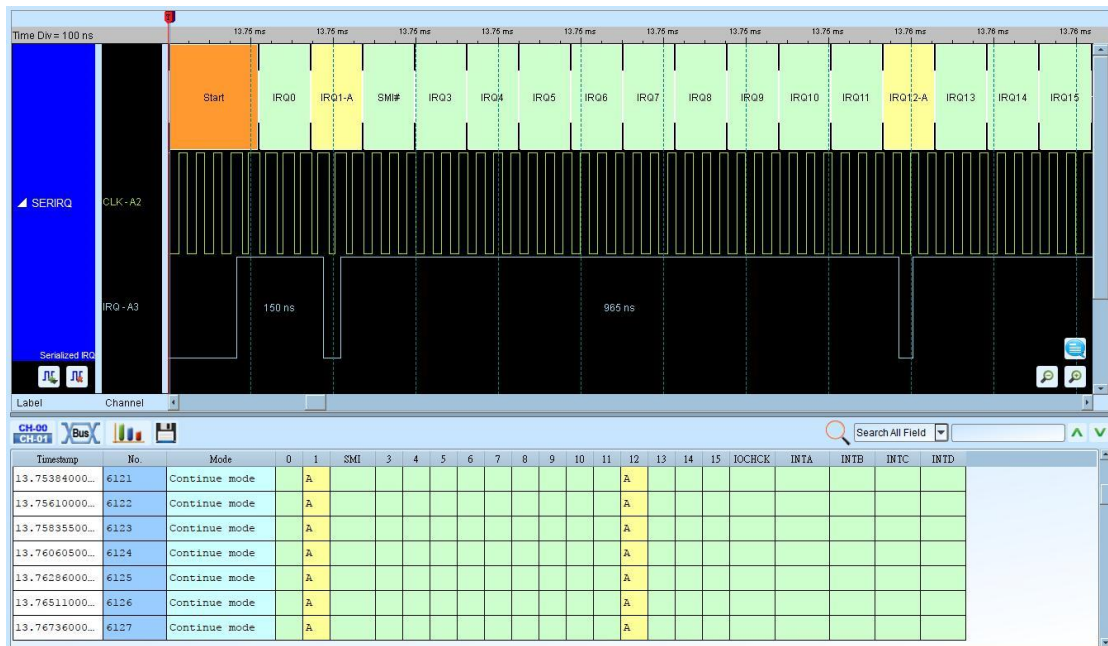
Clock	IRQ/Data Frame	Signal Sampled	# of clocks past Start
-9470	1	IRQ0	2
-9452	2	IRQ1	5
-9434	3	SMI#	8
-9416	4	IRQ3	11
-9398	5	IRQ4	14
-9380	6	IRQ5	17
-9362	7	IRQ6	20
-9344	8	IRQ7	23
-9326	9	IRQ8	26
-9308	10	IRQ9	29
-9290	11	IRQ10	32
-9272	12	IRQ11	35
-9254	13	IRQ12	38
-9236	14	IRQ13	41
-9217	15	IRQ14	44
-9199	16	IRQ15	47
-9181	17	IOCHCK#	50
-9163	18	INTA#	53
-9145	19	INTB#	56
-9127	20	INTC#	59
-9109	21	INTD#	62
-9019	1	IRQ0	2
-9001	2	IRQ1	5

Result

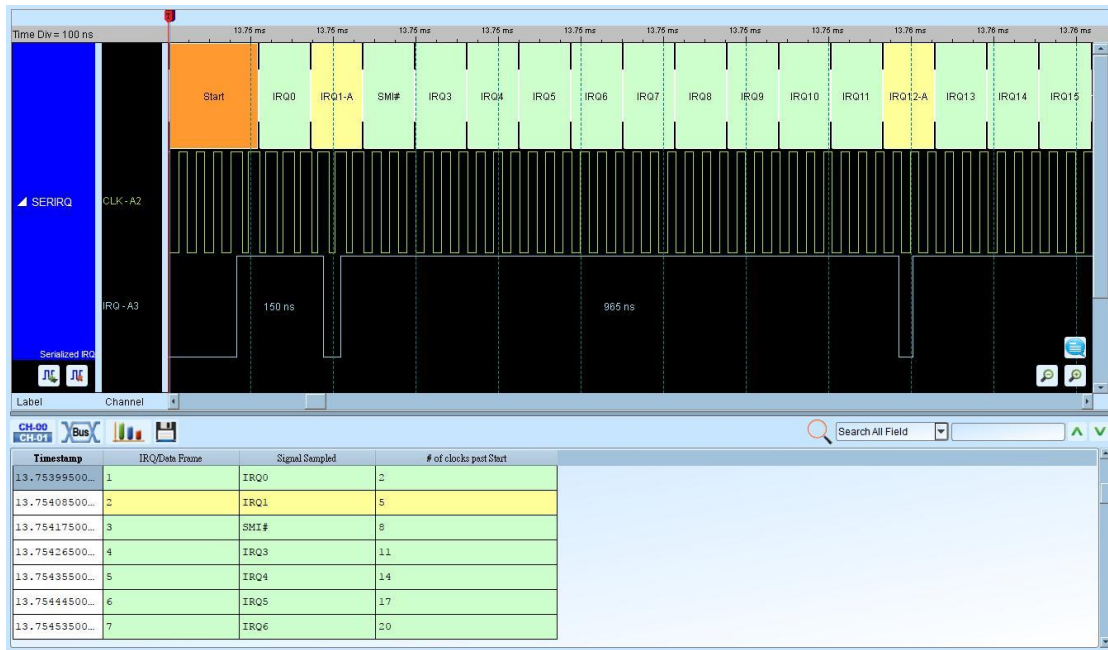
Normal mode



Normal mode(Show repeat frame)



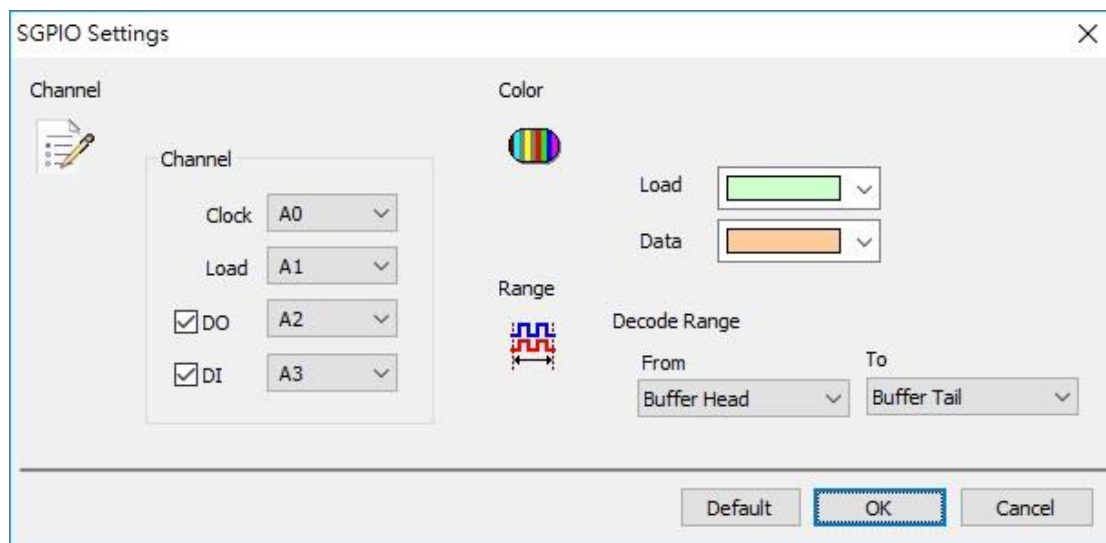
Advance mode



Serial General Purpose Input Output (SGPIO)

The SGPIO is a method to serialize general purpose IO signals. SGPIO defines the communication between an initiator and a target.

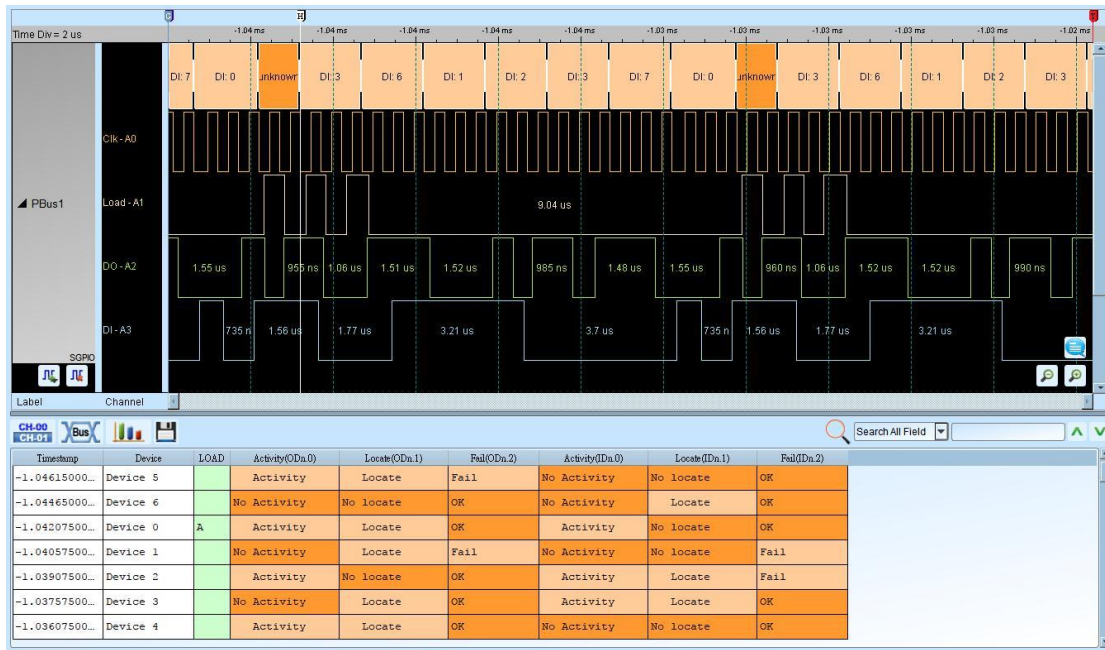
Settings



Channel: Show the selected channels (Clock, Load and Data), it can only use data out or data in or both.

Result:

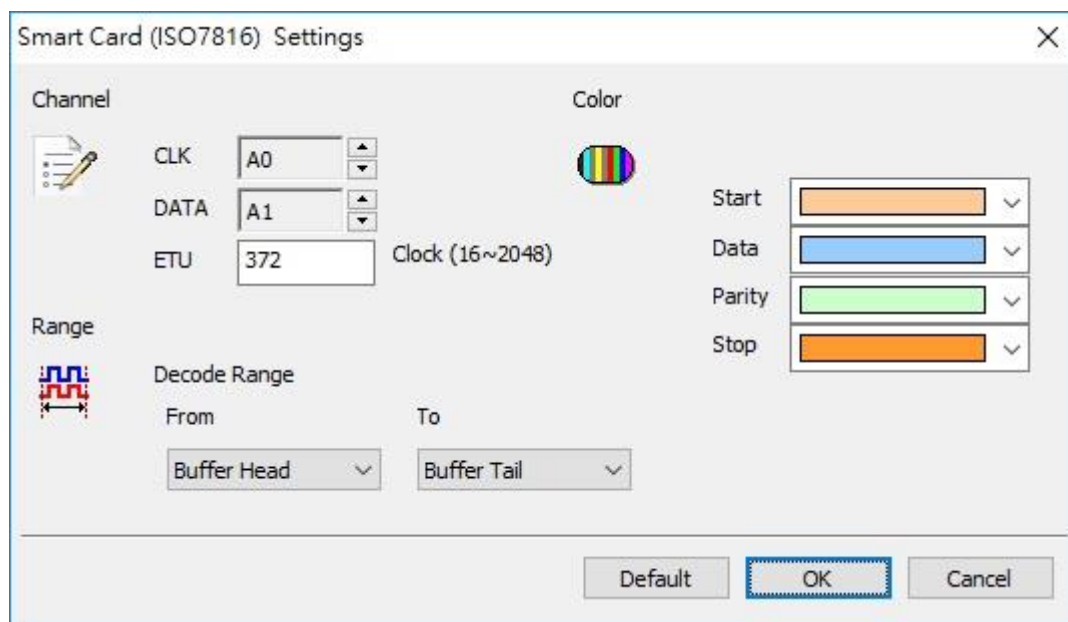
Click OK to run the Smart Card decode and see the result on the Waveform Window below.



Smart Card (ISO7816)

The card is made of plastic and provides strong security authentication for single sign-on within large organizations.

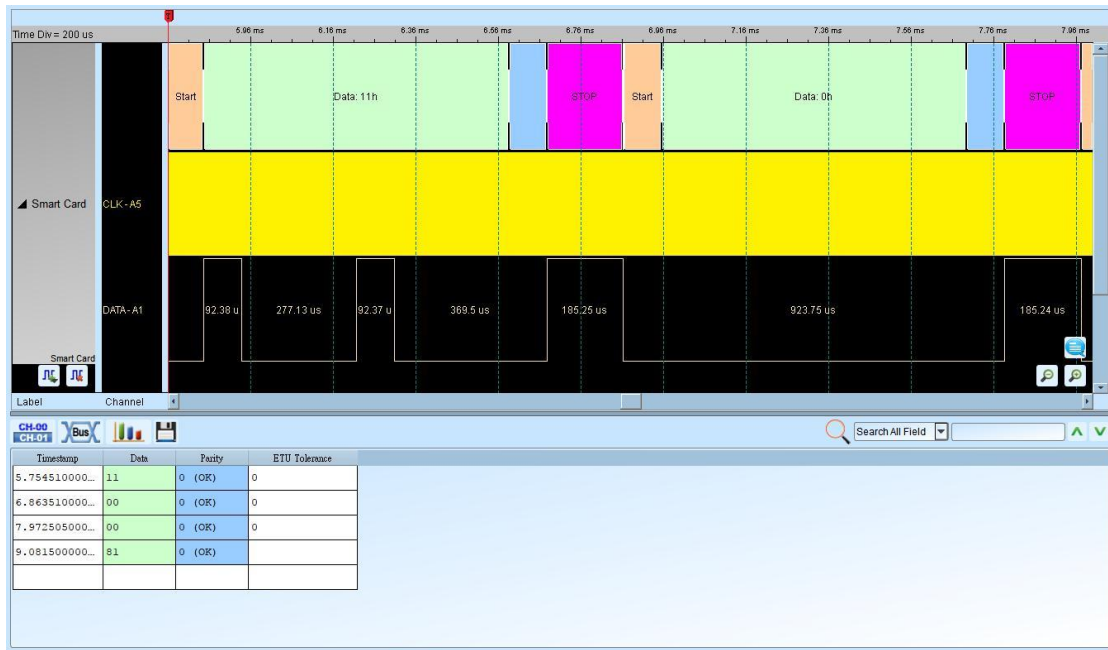
Settings



Channel: Show the selected channels (CLK and DATA) and the number of clocks within the bit (ETU).

Result

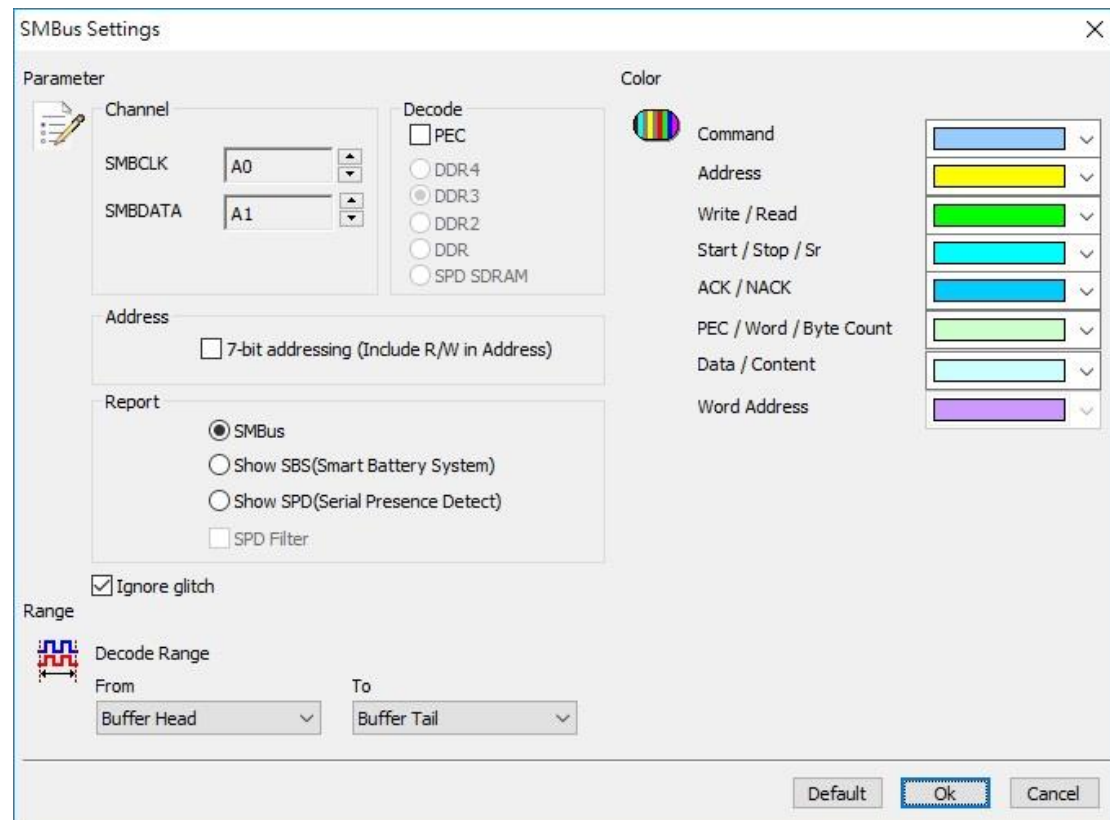
Click **OK** to run the Smart Card decode and see the result on the Waveform Window below.



System Management Bus (SMBus)

The SMBus (SMB) is a two-wire bus.

Settings



Clock: Show the selected channels (SMBCLK CH0 and SMBDATA CH1).

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

Show SBS (Smart Battery System): Show the Smart Battery System: voltage, electric current and the manufacturer.

Show SPD(Serial Presence Detect) : Report window show the configuration of memory module(DDR3, DDR2, DDR, SPD SDRAM) in EEPROM.

SBS/SPD Filter: Report window only show SBS/SPD packet.

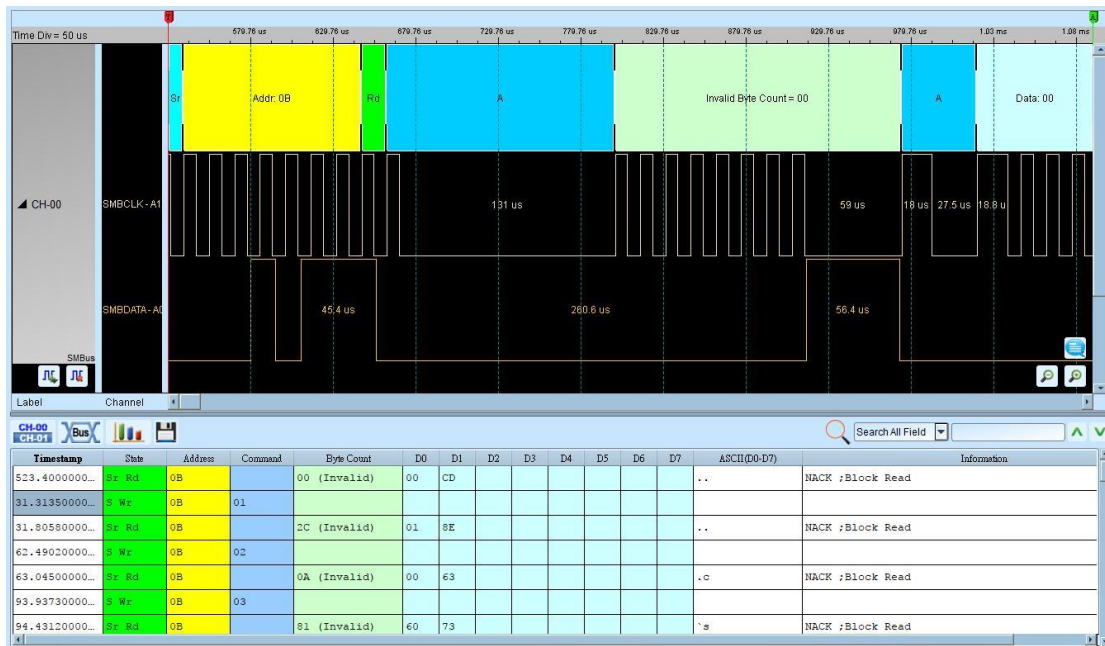
Ignore glitch: Ignore the glitch when the slow transitions.

Result

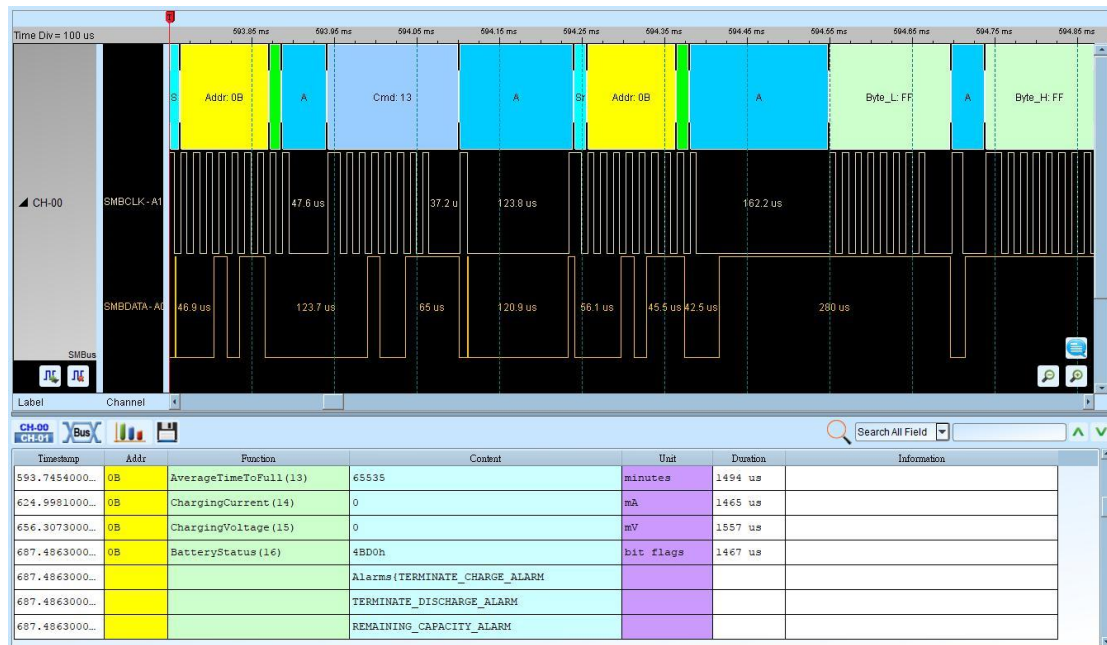
Click **OK** to run the SMBus Decode and see the result on the Waveform Window

below.

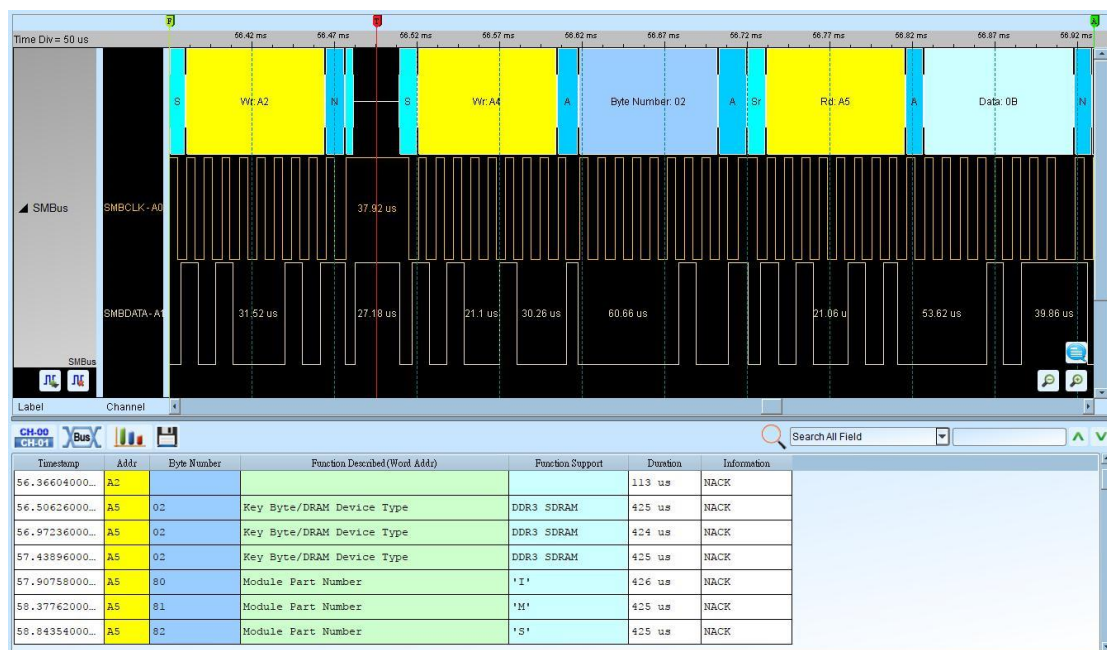
SMBus



Show SBS (Smart Battery System)



Show SPD (Serial Presence Detect)



Serial Microprocessor Interface (SMI)

The SMI is a two-wire bus.

Settings

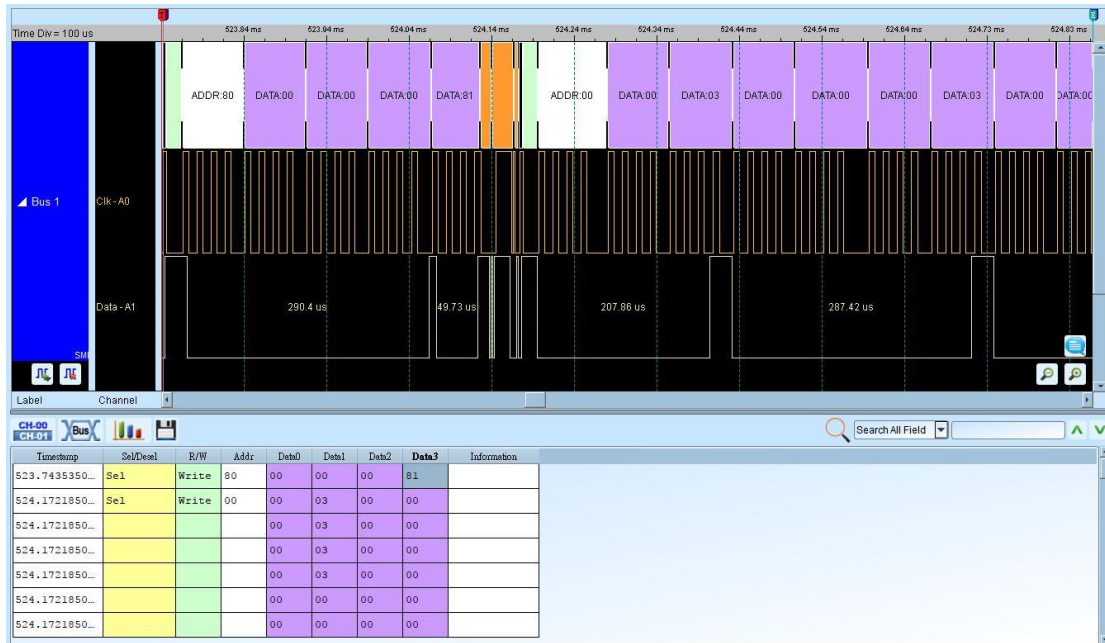
The SMI Settings dialog box is divided into three sections:

- Channel:** Contains two dropdown menus. 'Clk' is set to 'A0' and 'Data' is set to 'A1'.
- Color:** Contains six color selection dropdowns: 'Attn' (orange), 'Sel/Desel' (yellow), 'R/W' (light green), 'Address' (white), 'Data' (purple), and 'Attn desel' (orange).
- Range:** Contains a 'Decode Range' section with 'From' and 'To' dropdowns, both set to 'Buffer Head' and 'Buffer Tail' respectively.

At the bottom of the dialog are three buttons: 'Default', 'OK', and 'Cancel'.

Channel: Show the selected channels (Clk and Data)

Result



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

SoundWire Settings

SoundWire Settings

X

Parameter

Channel

CLK

Data

Bank

Bank0

Bank1

Color

PREQ

PING

Reserved

SSP

BREQ

BREL

TX

PayLoad Data

Dynamic SYNC

Device Address

Register Address

RegData

Parity

RX

Slave State

Static SYNC

PHY SYNC

RD

WR

ACK

NAK

Range

Start End

Frame Shape

Column Row

Payload Data

Data Port Settings

Number

	HStart		HStop	Word Length	Sample Interval	Offset 1	Offset 2	Channel								Block Package Mode	Port Flow Mode	
	1	2						3	4	5	6	7	8					
DP0	1	1	1	10	96	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP1	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP2	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP3	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP4	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP5	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP6	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP7	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP8	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP9	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP10	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP11	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP12	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP13	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP14	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous
DP15	1	1	1	0	96	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Block-per-Port	Isochronous

CLK: Clock signal

Data: Data signal

Frame Shape:

Column:2~16(Even)

Row:48~256

Bank: Bank0, Bank1

A Data Port is a subset of a Device that is either a source or sink of one Payload Stream on the SoundWire bus. Data Ports provide an abstraction of a Payload Stream being divided into one or more channels which, in the audio application space will correspond to individual audio channels.

Number of Data Port: 1~16(DP0~DP15)

1. **HStart:** HStart is a 4-bit number identifying the lowest numbered column in the SoundWire Frame that contains BitSlots for this Payload Stream.
2. **HStop:** HStop is a 4-bit number identifying the highest numbered column in the SoundWire Frame that might contain BitSlots for this Payload Stream.
3. **Word Length:** WordLength is a 6-bit number that controls the size of a Payload Channel Sample.
4. **Sample Interval:** Sample Interval is an integer in the range 2 to 65535 that controls the spacing from one Sample Event to the next, and the size of a Sample Window.

5. Block Package Mode:

Block per port: $\text{Block Offset} = \text{Offset1} + (256 * \text{Offset2})$

Block per channel: $\text{Block Offset} = \text{Offset1}$, $\text{Sub-Block Offset} = \text{Offset2}$.

*Block Offset is an integer in the range 0 ~ 65535

*Sub-Block Offset is an integer in the range 0 to 255

*Offset1 is an integer in the range 0~65535

*Offset2 is an integer in the range 0~255

6. Channel: Divide a Data Port into different parts, which are often used for data distribution of left and right audio channel etc. It can select up to 8 channels, and can be selected without order.

7. Port Flow Mode: Isochronous, Tx-Controlled, Rx-Controlled and Full-Asynchronous modes

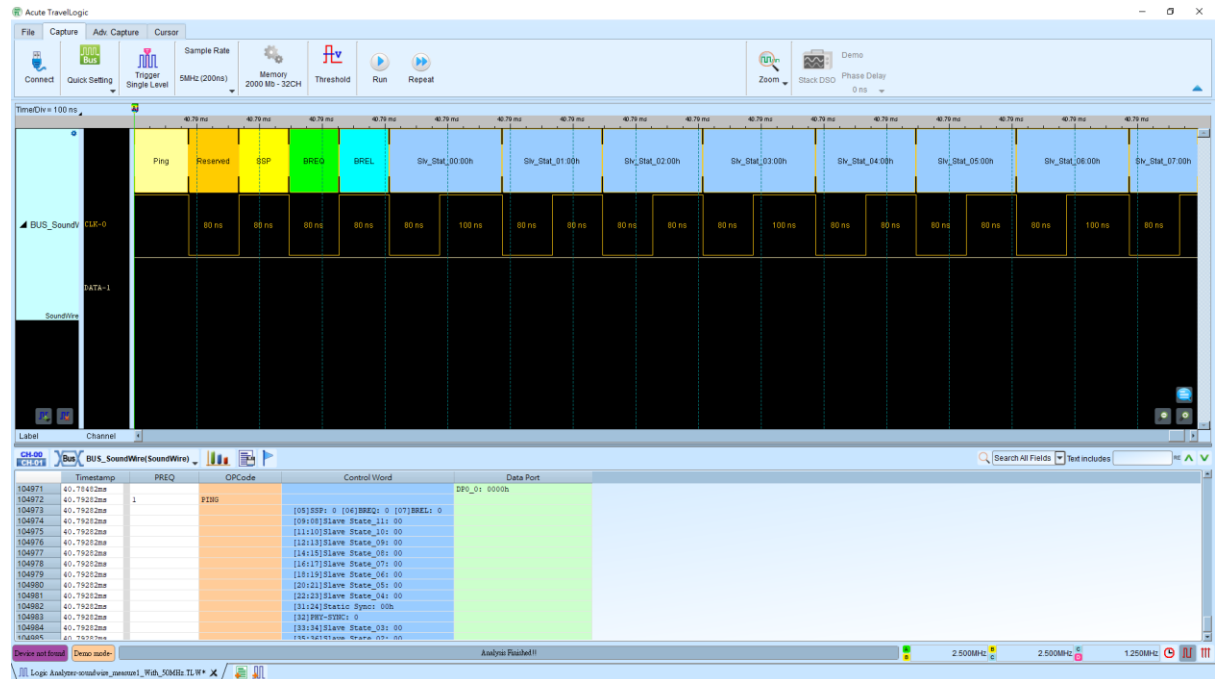
*Isochronous: 'Normal' mode No Valid data is transported in every
Payload Data Block.

*Tx-Controlled: 'Push' mode , A flow-control bit driven by the Source
Data Port controls whether or not valid data is transported
in the Payload Data Block.

*Rx-Controlled: 'Pull' mode , A flow-control bit driven by the Sink Data
Port controls whether or not valid data is transported in
the Payload Data Block.

*Full-Asynchronous: Flow-control bits driven by both the Source and
Sink Data Ports control whether or not valid data is
transported in the Payload Data Block.

Result

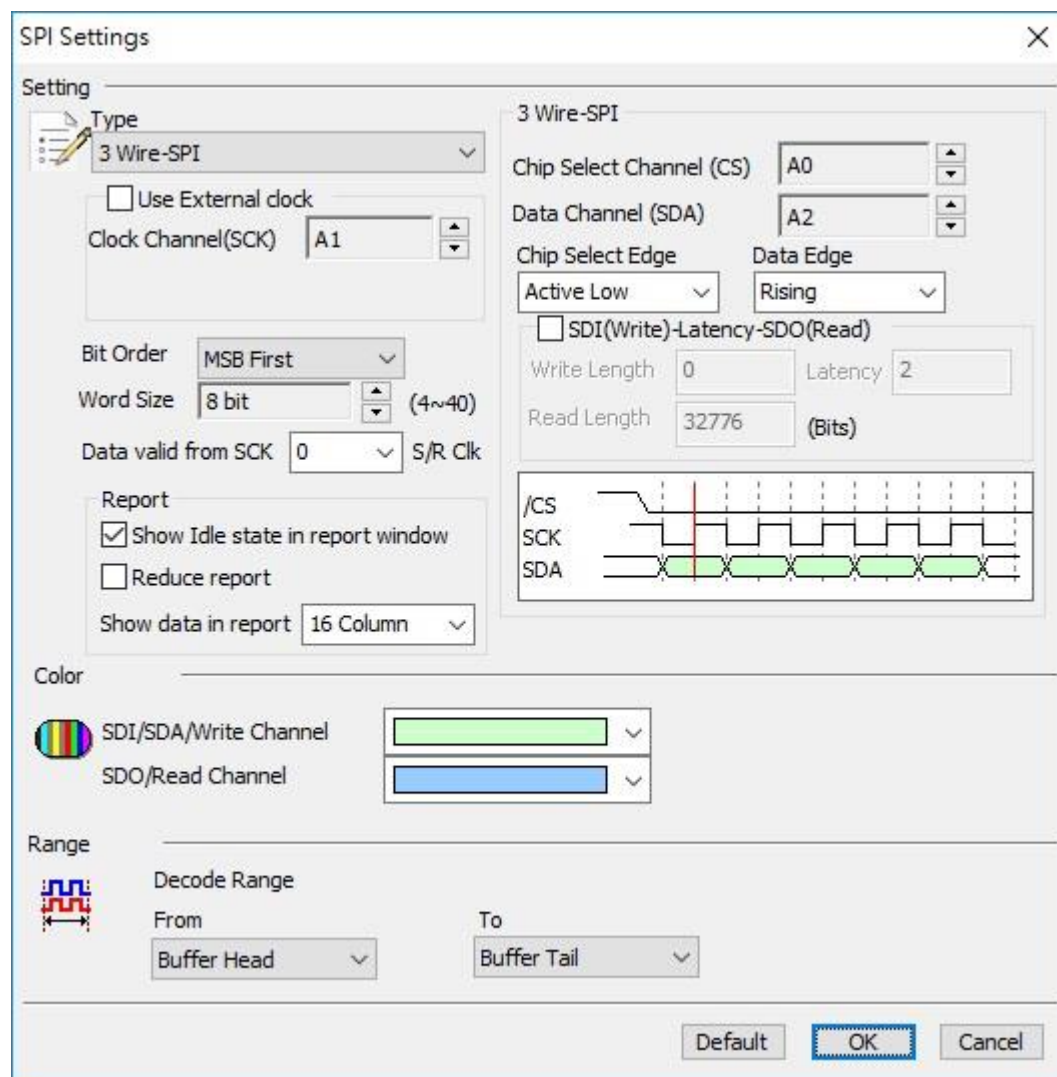


Serial Peripheral Interface (SPI)

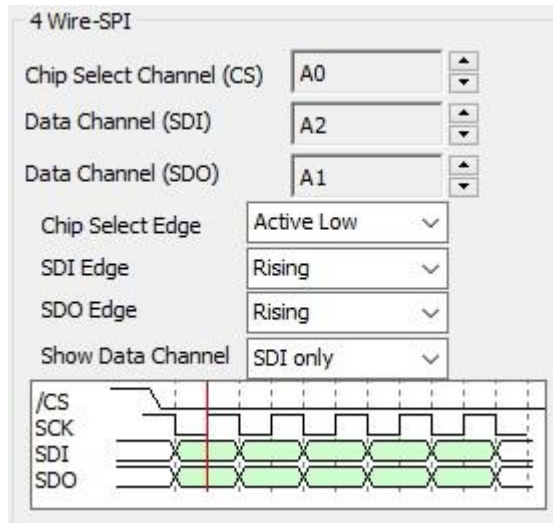
The SPI, is one kind of 4-wires synchronous serial data link. The SPI bus can be 4 wires, 3 wires, or 2 wires.

Settings

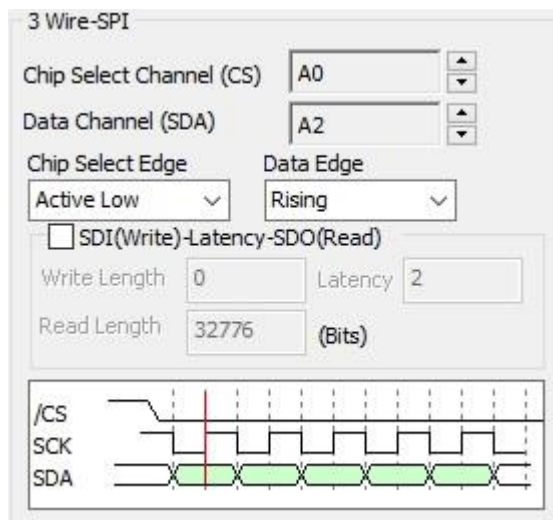
Type: The default is 3 Wire-SPI.



4 Wire-SPI dialog box → CS, SCK, SDI, SDO

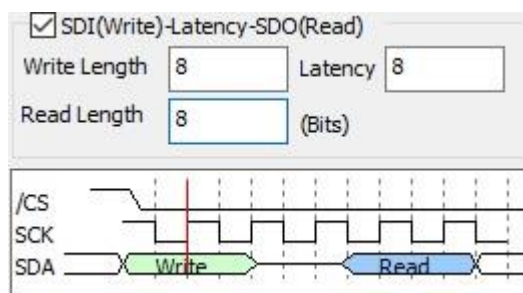


3 Wire-SPI dialog box → CS, SCK, SDA

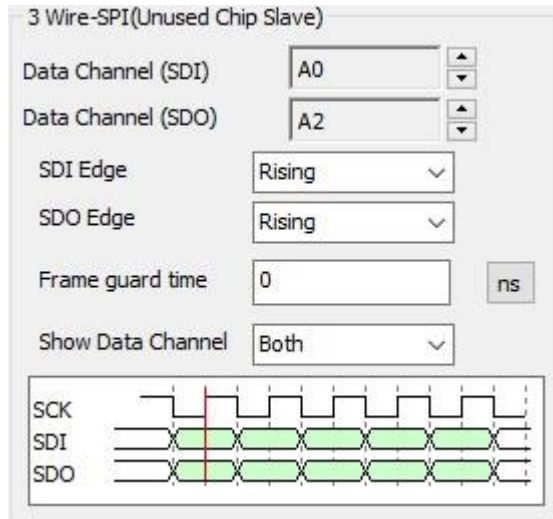


Click the SDI(Write)-Latency-SDO(Read) to show the dialog below.

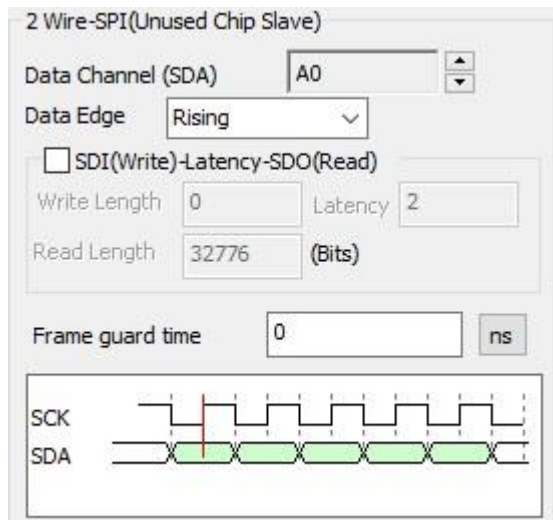
The maximum length is 65535 (Bits)



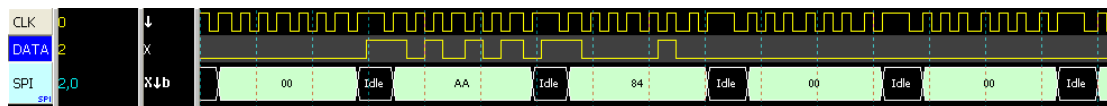
3 Wire-SPI (Unused Chip Slave) dialog box → SCK, SDI, SDO

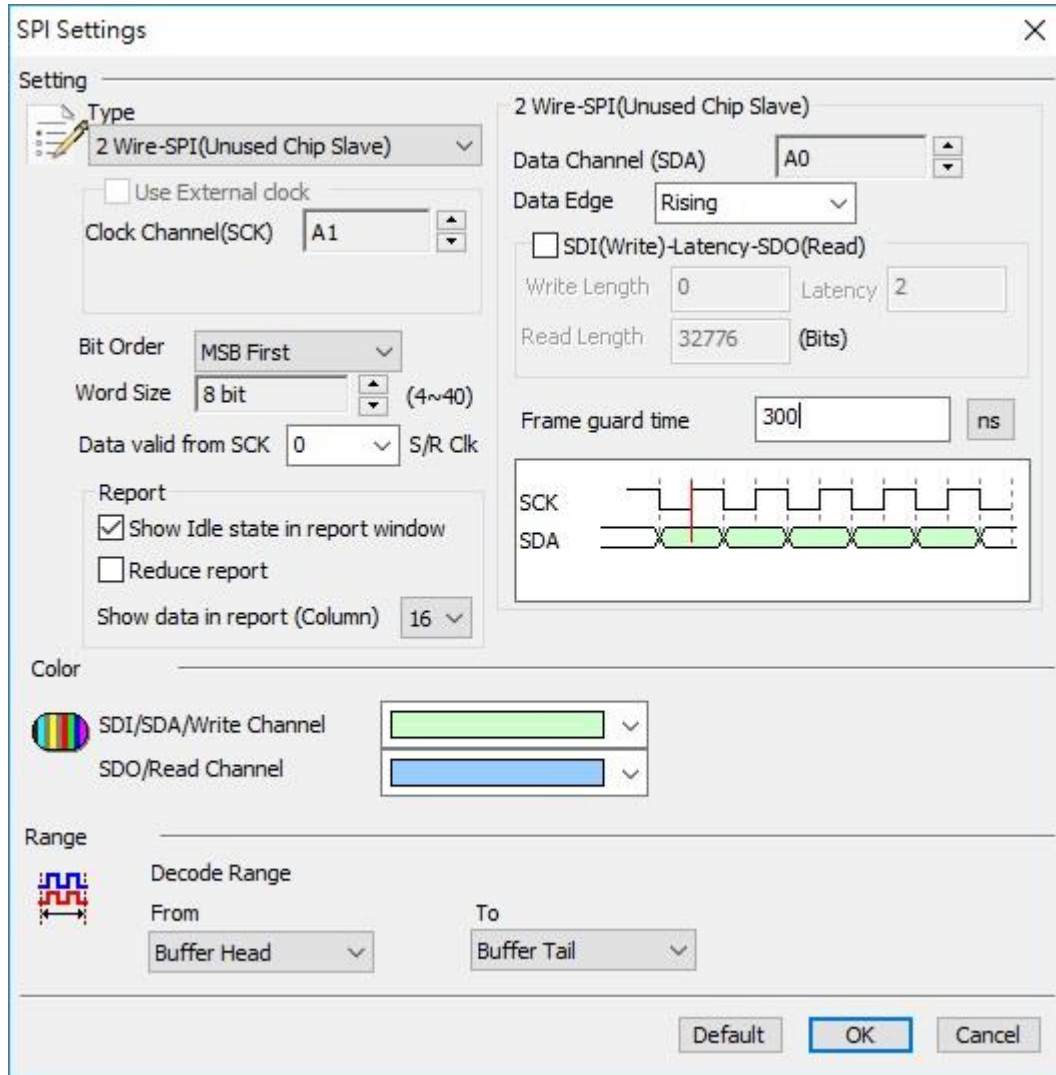


2 Wire-SPI (Unused Chip Slave) dialog box → SCK, SDA

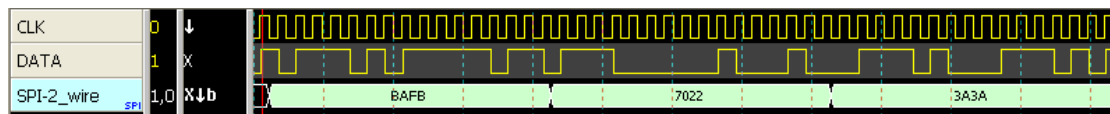


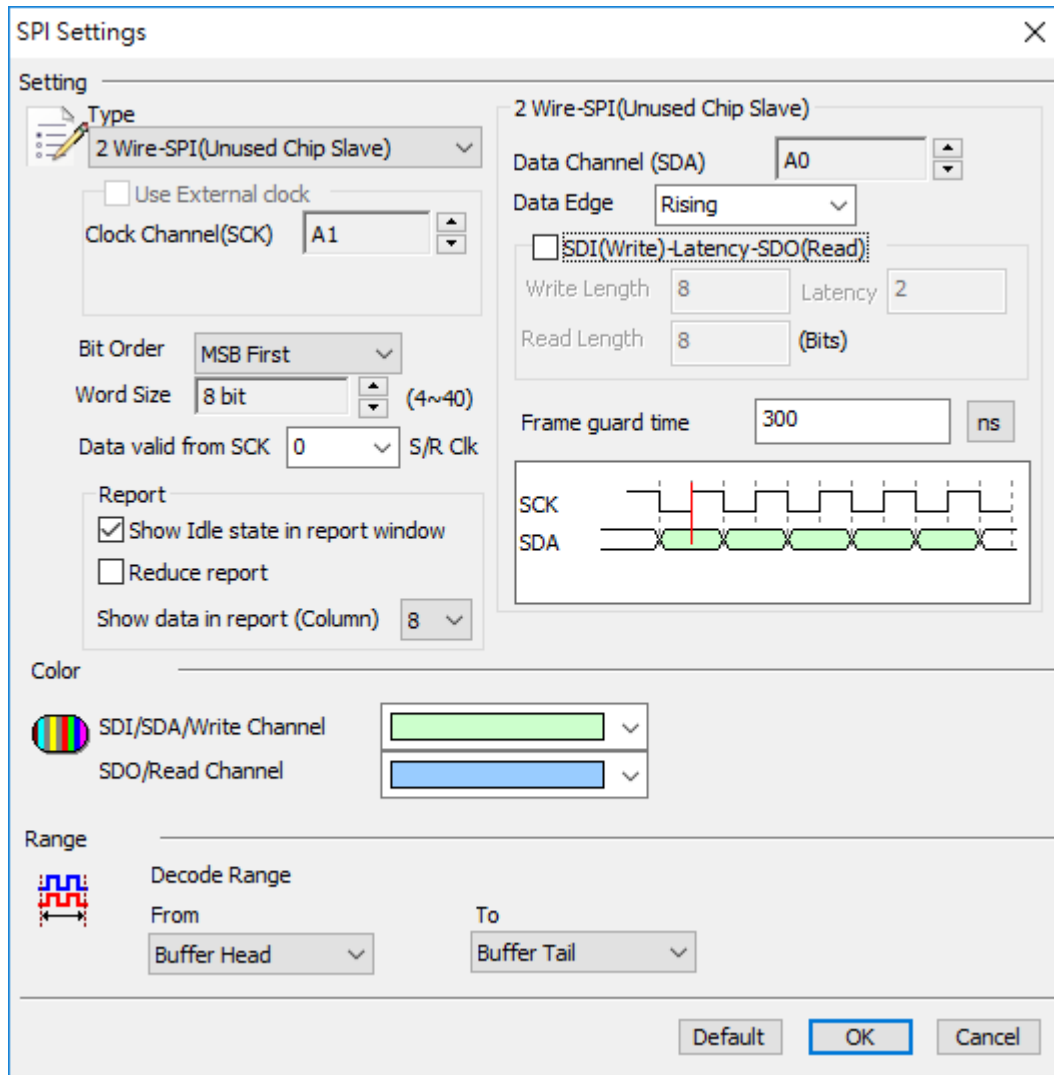
If the chip slave is not used and the interval between frames is not 0. You can set the interval as 6us, any data bit higher than 6us will be seen as Idle.



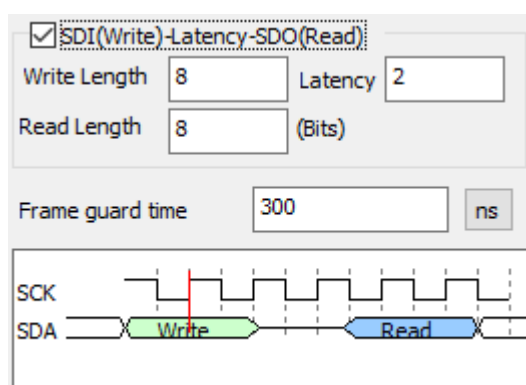


If the chip slave is not used and the interval between frames is 0. You can see the data continuous as the dialog below.





We also offer bi-direction mode as the dialog below.



Check the SDI(Write)-Latency-SDO(Read) to set the bit numbers for the 3 columns;

Write and Read (1~65535), Latency (0~65535).

Bit Order: MSB first or LSB first.

Word size: Default is 8 bits, minimum is 4 and maximum is 32.

Report: Show Idle state in report window, You can disable this default for not to display the idle state on the Report Window.

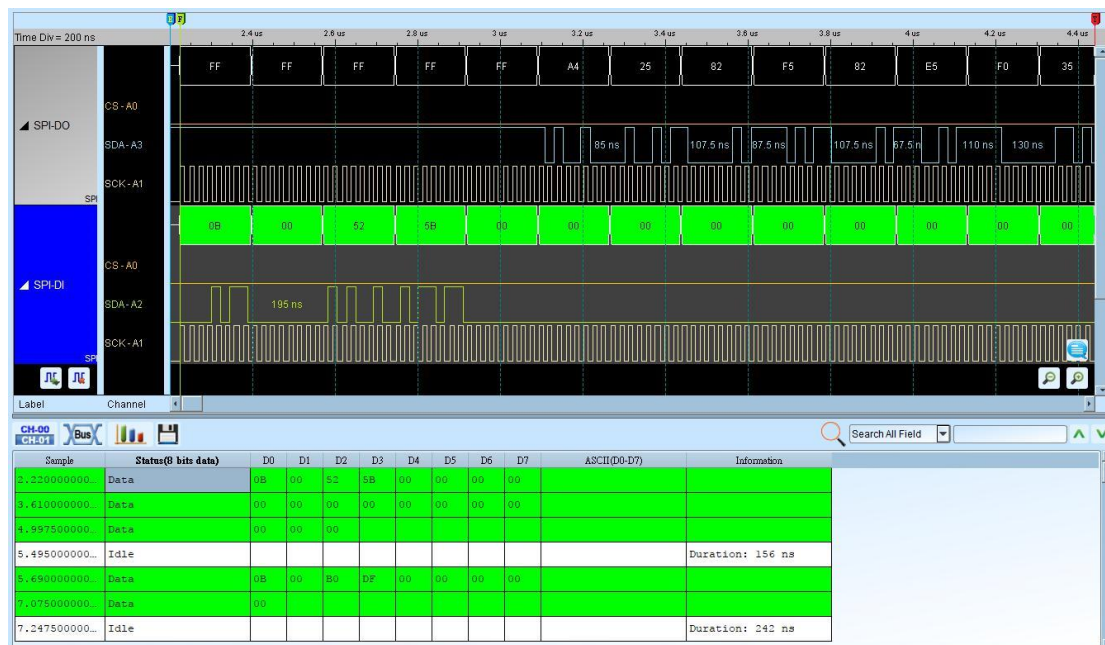
Show data in report: Display the ASCII code with default 16 columns on the Report Window.

Data Valid from SCK: In some SPI devices, the data is not valid right after the data output or the clock edge. You can set the data valid after 0-3 units of the sampling rate in Data valid from SCK; if the unit is 1 and the sampling rate is 200MHz, the delay time is 5 ns.

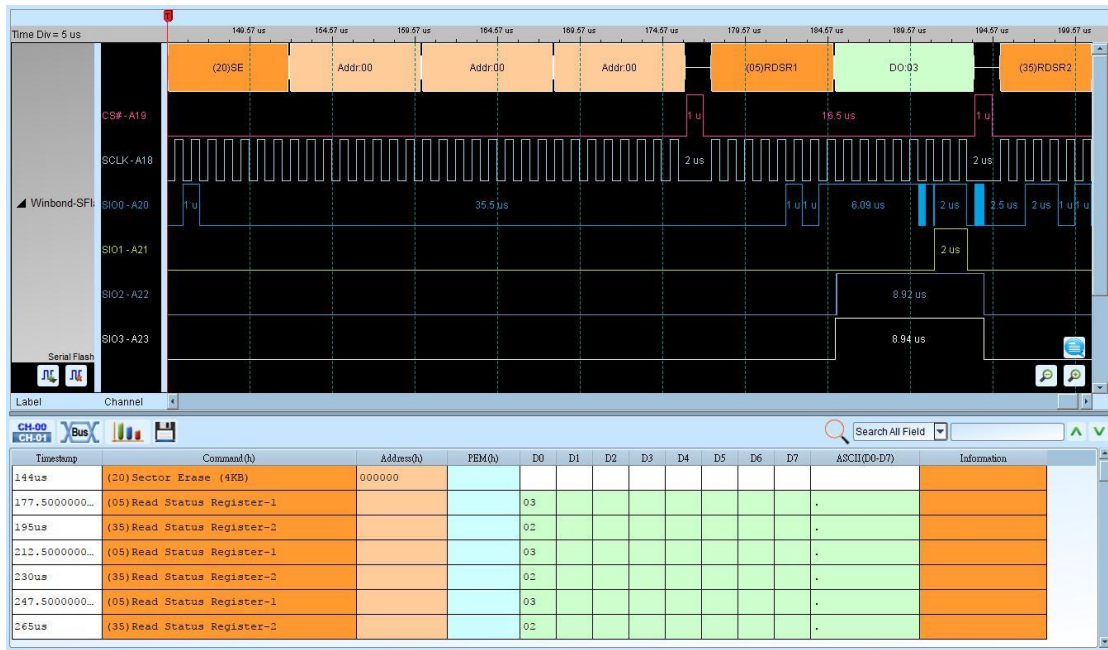
Result

Click **OK** to run the SPI decode and see the result on the Waveform Window below.

3-Wire SPI, Internal clock mode



3-Wire SPI, External clock mode



Serial Peripheral Interface NAND (SPI NAND)

SPI NAND Flash is a SPI/QPI interfaced NAND flash memory. The decoder translates the bus signal to command/address/data field to provide an easier way to exam the SPI NAND waveform.

Settings

CS#: Chip select

SCLK: Clock

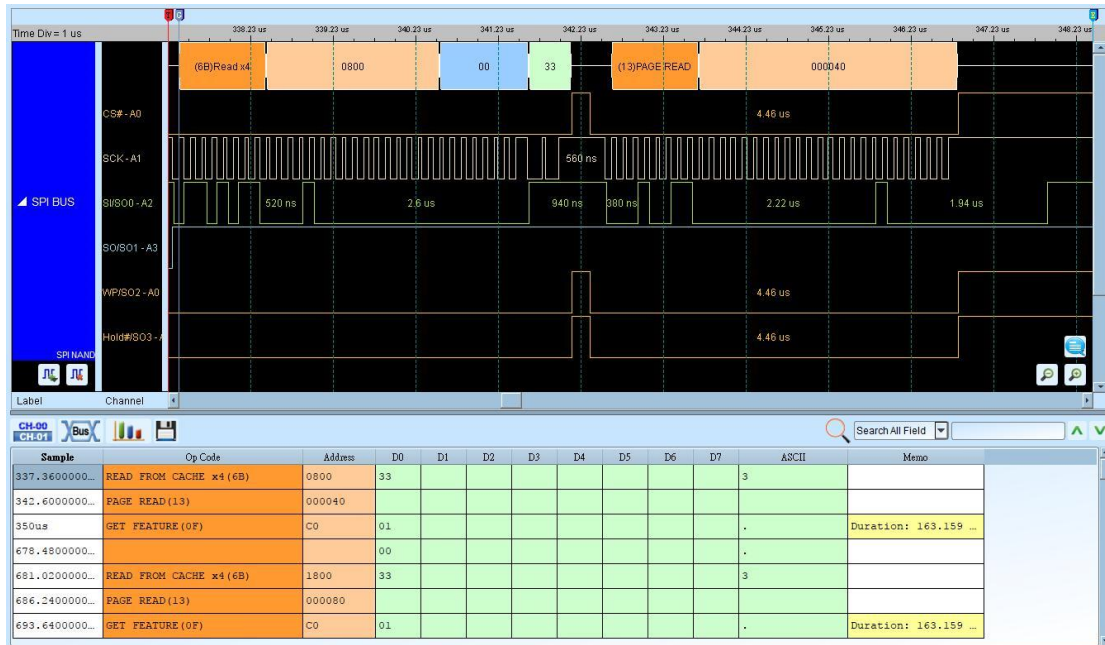
SIO0 – SIO3: Data

Start up reading mode: The initial setting of the reading mode for the decoder.

Command deselect time: The minimum required time for #CS deselecting. Clock

LOW to output valid: The data probe time after clock falling.

Result



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

Select Channel: Show the selected channels.

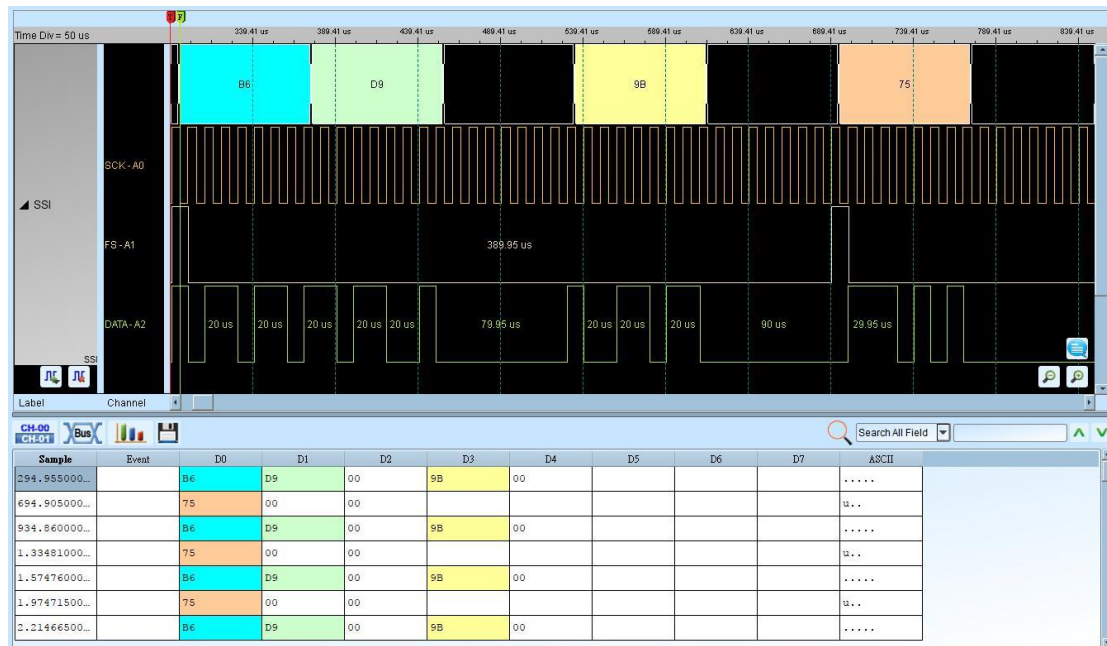
Mode: Normal or Network.

Line Of Data: Transmit or Receive.

Merge continuous unknown: Combine the unknown data only in Network mode.

Result

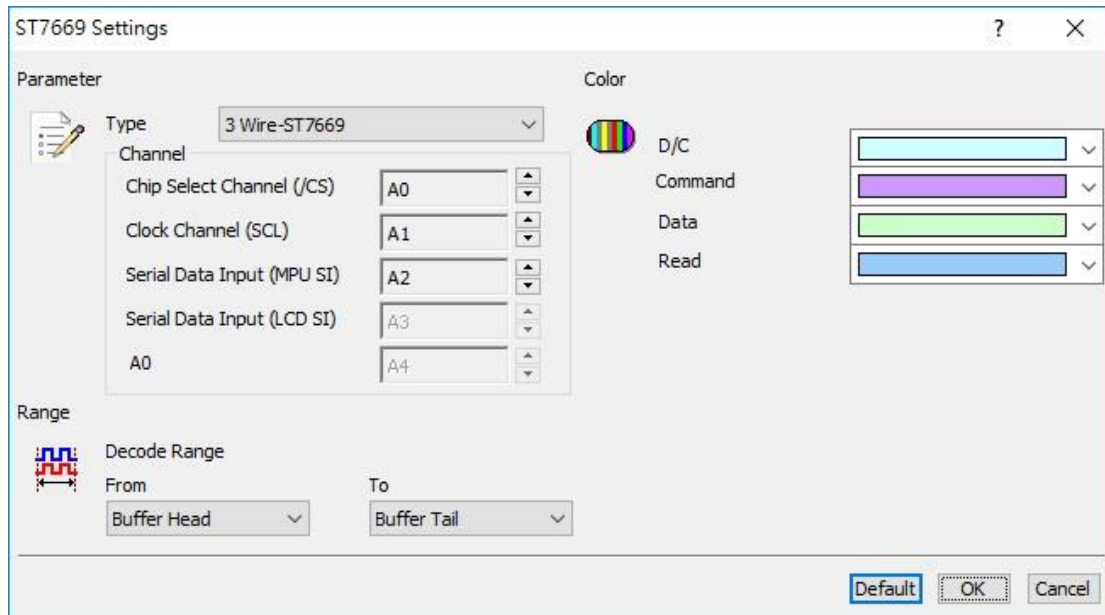
Click **OK** to run SSI decode and see the result on the Report Window below.



ST7669

ST76669 was developed by Sitronix to interface with the LCD module.

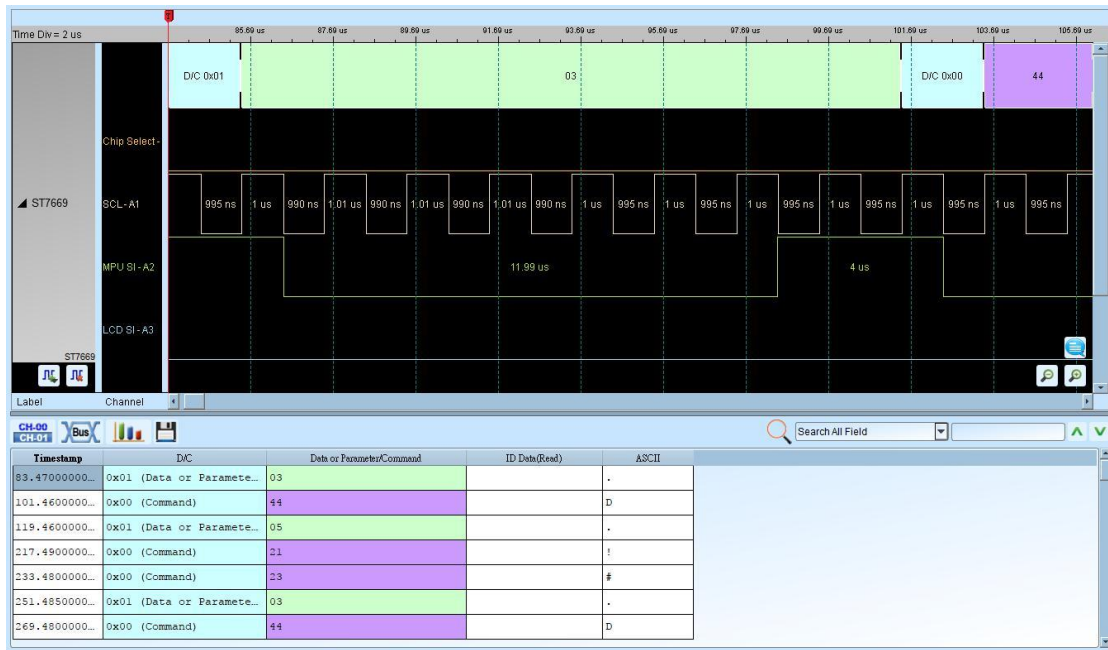
Settings



Channel: Show the selected channels.

Result

Click **OK** to run the LCD1602 decode and see the result on the Waveform Window below.



Serial VID Interface 2.0 (SVI2)

The SVI2 protocol is an interface for power management and developed by AMD.

The SVI2 working voltage is between 1V to 1.6V and its maximum frequency is at 20MHz with 3 bits : SVC/ SVD/ SVT.

Settings

SVC: SVI2 clock.

SVD: SVI2 data.

SVT: SVI2 telemetry data line. Telemetry will not be decoded if the SVT is not

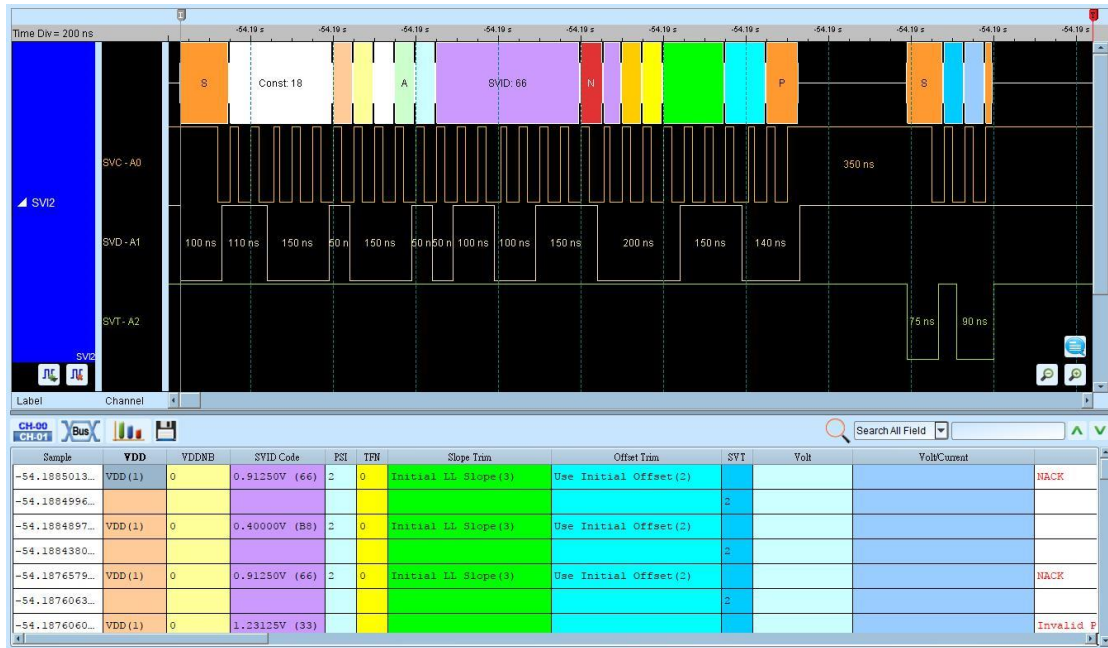
checked.

SVI2.x / SVI1.x: Select SVI2 / SVI protocol to decode.

Reduced Report: Only show SVD / VOTFC packet in the report window.

Result

Click **OK** to run the SVI2 decode and see the result on the Waveform Window below.



Serial VID (SVID)

Serial VID (SVID) protocol is for power management and developed by Intel. SVID voltage is between 1.0V to 1.1V, maximum frequency at 26.25MHz and is 3 wires : SCLK/ SDATA/ ALERT.

Supported version:

IMVP7/VR12, VR12.1, VR12.5, VR12.6

IMVP8/VR13

If you have any issues with SVID protocol features, please contact your Intel Field Representative.

Settings

SerialVID Settings

Setting

SCLK: A0 SDATA: A1 Alert: A2

VR Controller: Single Multiple

Startup: PWM Spec.: VR12.0

Fast slew rate: 10 mV/us

Slow slew rate: Fast /2

Not to decode the frame when its:

- Stop pattern is wrong
- Clock duty cycle is out of the Spec. range

Color

Start	[Orange]	End	[Black]
Address	[Light Orange]	Turn around	[Purple]
Command	[Yellow]	ACK	[White]
MA. Payload	[Light Green]	SL. Payload	[Light Blue]
Parity	[Cyan]	Frame fill	[Black]

Range

Decode Range: From [Buffer Head] To [Buffer Tail]

Buttons: Default, OK, Cancel

SClk: SVID clock.

SData: SVID data.

Alert: Alert, optional reminder.

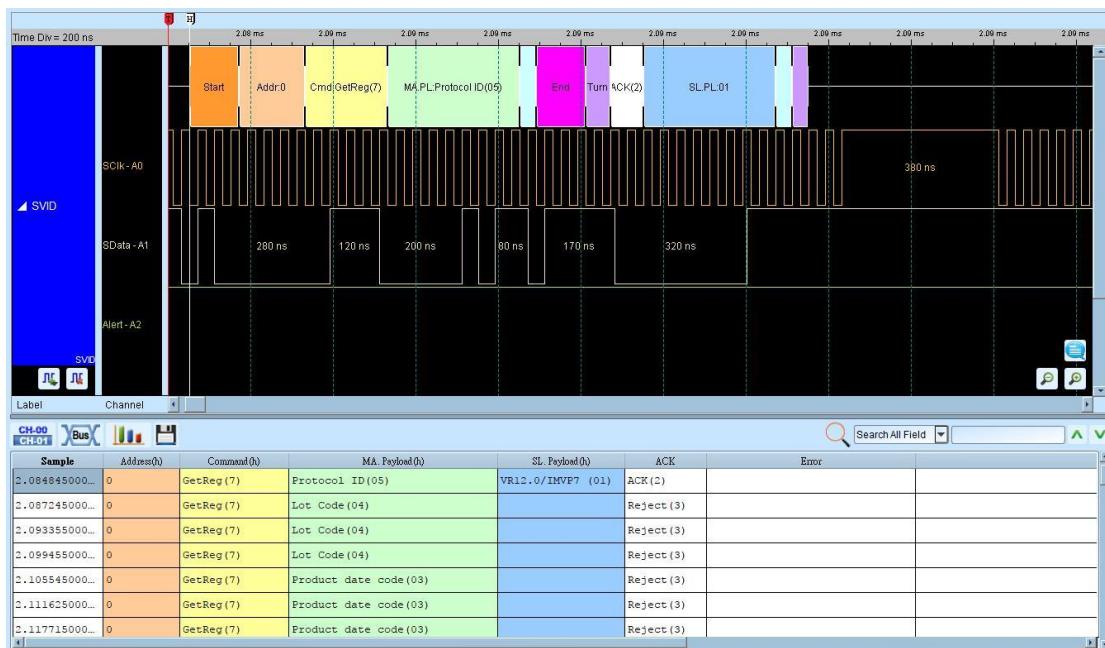
VR Controller: Set single or multiple controllers in current VR. Set different address.

Not to decode the frame when its stop pattern wrong or clock duty cycle is out of range.

Result

Click OK to run the SVID decode and see the result on the Waveform Window

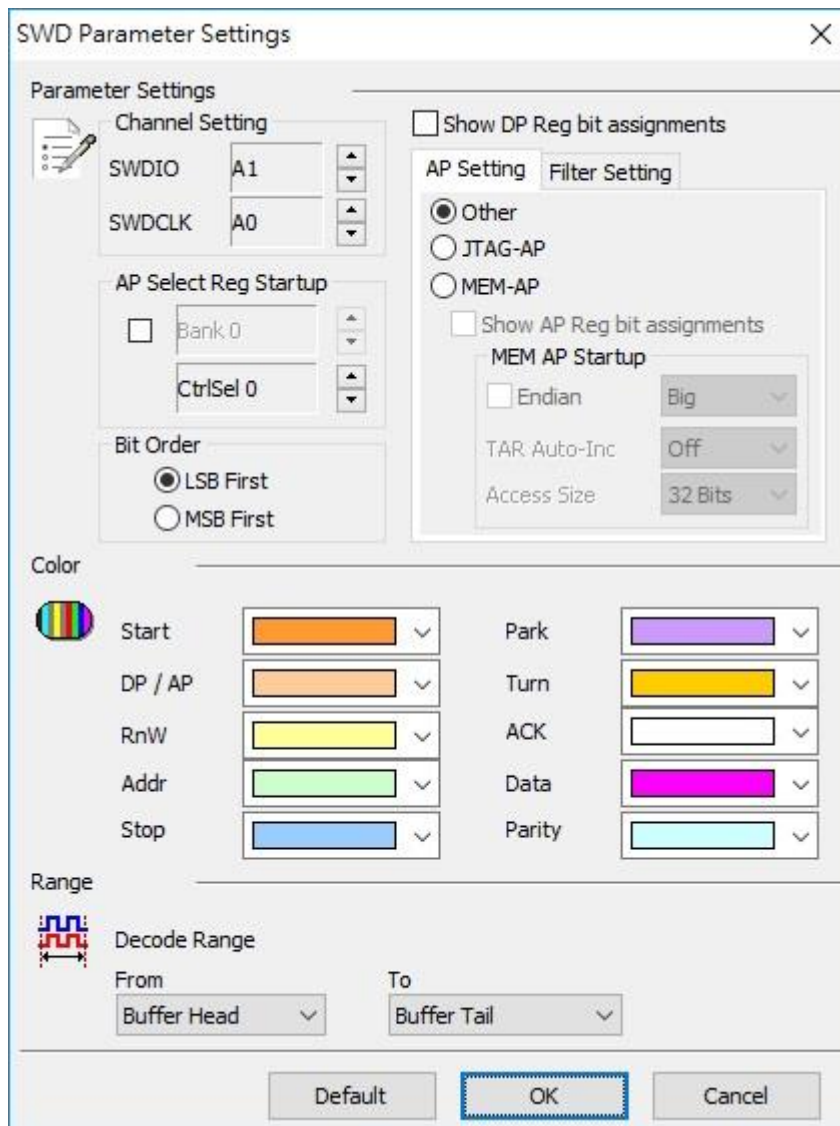
below.



Serial Wire Debug (SWD)

The SWD is a 2-pin electrical alternative JTAG interface that has the same JTAG protocol on top. It uses the existing GND connection. SWD uses an ARM CPU standard bi-directional wire protocol, defined in the ARM Debug Interface v5.

Settings



SWDIO: I/O data.

SWDCLK: Clock.

AP Select Reg Startup: When the AP Select Reg Startup is not assigned, only the

address information will be shown. You may input the Bank and Ctrl/Select initial values manually.

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:

JTAG-AP: Show the JTAG AP decode.

MEM-AP: Show the MEM AP decode.

Other: Show Bank X Register X.

<input checked="" type="radio"/> Other	Time	Select	RnW	Address (h)	ACK	Data
<input type="radio"/> JTAG-AP	-0.0003 ms	AP	Write	Bank 0 Register 0 (0)	OK	23 00 00 52
<input type="radio"/> MEM-AP	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
<input type="radio"/> Other	Time	Select	RnW	Address (h)	ACK	Data
<input checked="" type="radio"/> JTAG-AP	-0.0003 ms	AP	Write	CSW Register (0)	OK	23 00 00 52
<input type="radio"/> MEM-AP	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
<input type="radio"/> Other	Time	Select	RnW	Address (h)	ACK	Data
<input type="radio"/> JTAG-AP	-0.0003 ms	AP	Write	CSW Register (0)	OK	23 00 00 52
<input checked="" type="radio"/> MEM-AP	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

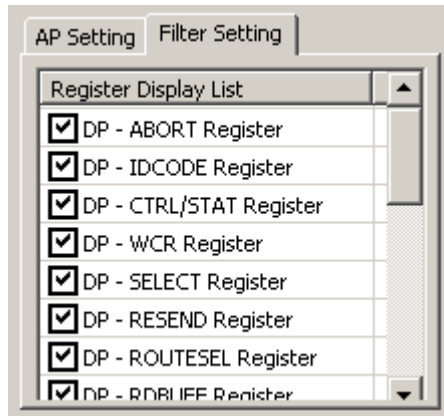
Show AP Reg bit assignments: Display the AP register information if JTAG-AP or MEM-AP checked.

Select	RnW	Address (h)	ACK	Data
<input checked="" type="checkbox"/> MEM-AP				
<input checked="" type="checkbox"/> Show AP Reg bit assignments				
AP	Read	BASE Register (8)	OK	00 00 00 00
				BASEADDR [31:12] E00FF
				Format [1] 1
				Entry present [0] 1

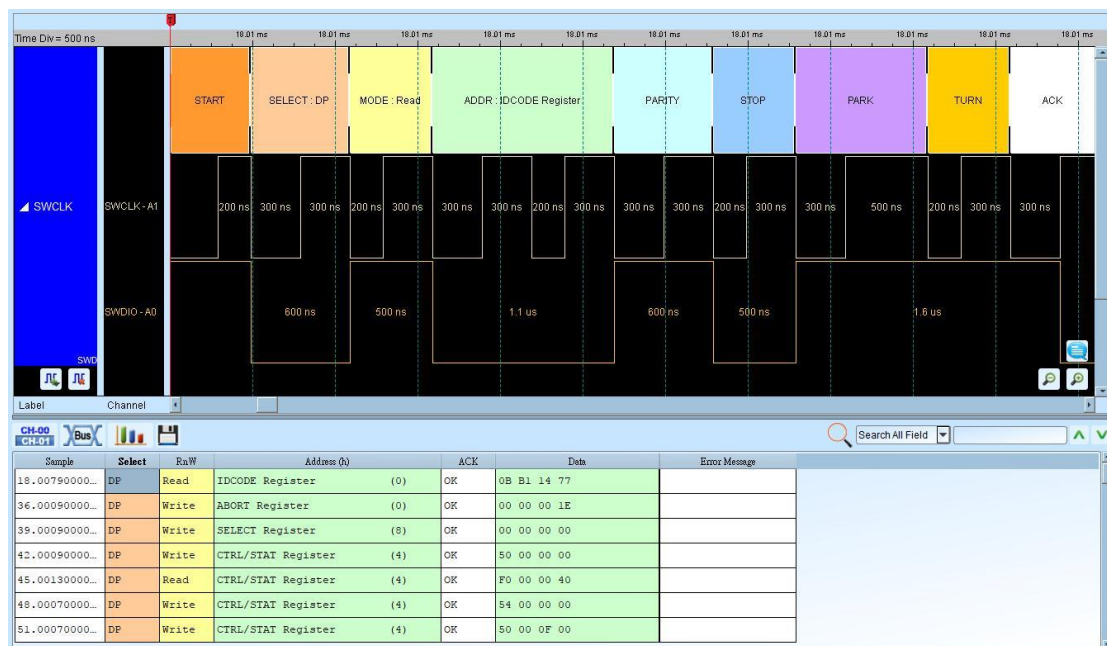
MEM AP Startup: Assign the initial MEM-AP value.

Select	RnW	Address (h)	ACK	Data	Notes
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.



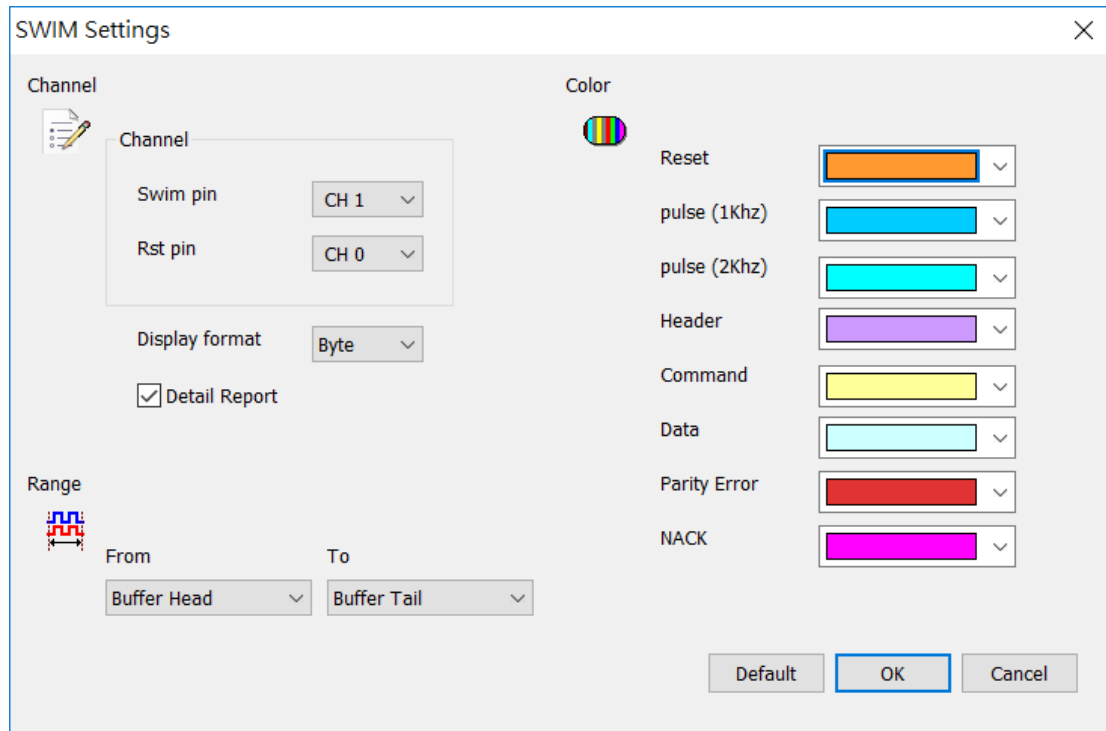
Result



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



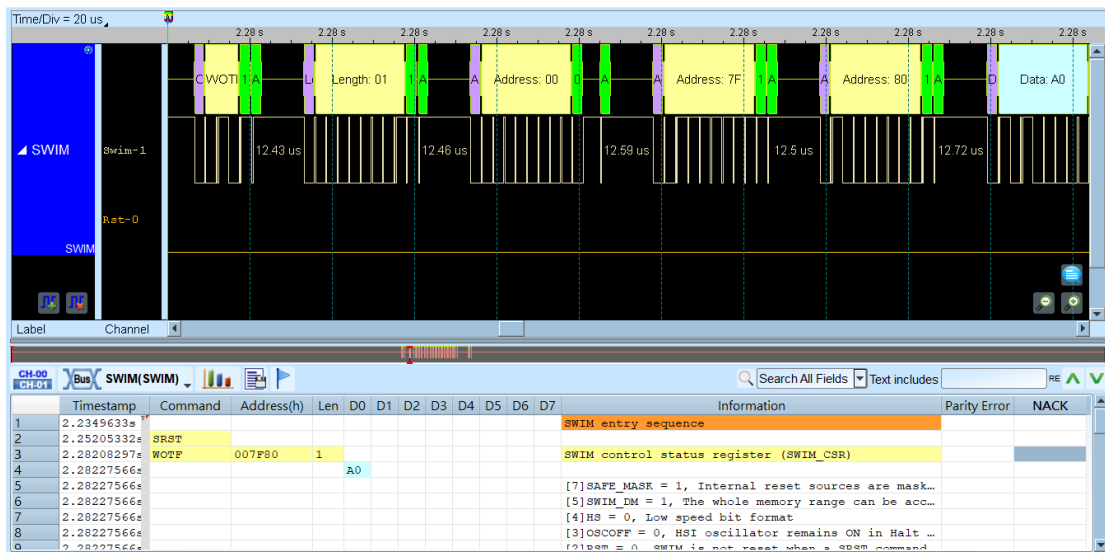
Swim pin: set the Swim Channel.

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.

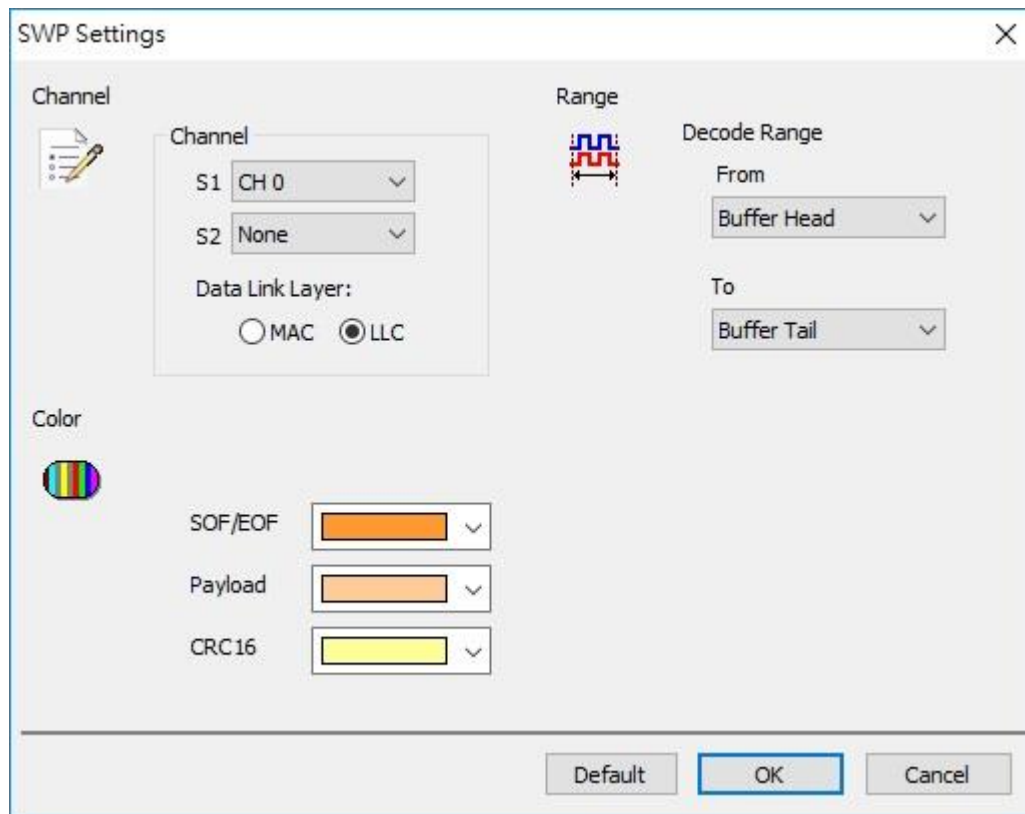
Result



SWP

The Single Wire Protocol (SWP) is a single-wire connection between the SIM card and a NFC chip in a cell phone.

Settings

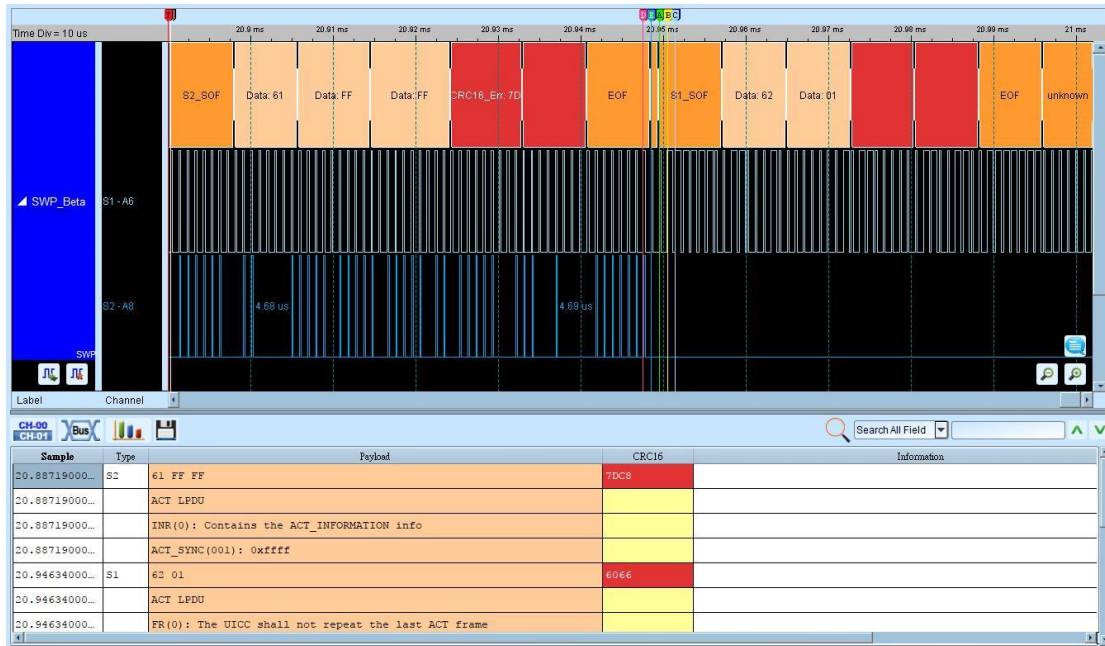


S1: I/O data.

S2: I/O data in current domain and it need convert to voltage domain.

Data Link Layer: Supported MAC and LLC layers

Result



Universal Asynchronous Receiver/Transmitter (UART)

The UART (RS-232 or RS-485) protocol has two message types: Transmitter (TX) and Receiver (RX); you can measure the UART protocol in one signal or two signals.

RS-485 need convert to logic signal, because LA can not capture differential signal.

Settings

Data: Show the selected channel (CH0).

Rx: Show Rx data in report window.

Auto: Shows High or Low when auto detection Idle.

Idle high: Idle condition shows High.

Idle low: Idle condition shows Low.

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

Baud Rate: Data rate (bits per second), and the range is 110 ~ 2M (bps).

Protocol: (Parity - Data Bits - Stop Bits)

Parity: N (None), O (Odd), or E (Even).

Data Bits: 5 to 10 bits.

Stop Bits: 1 to 2 bits.

MSB first: The default is LSB first; click it to change to MSB first.

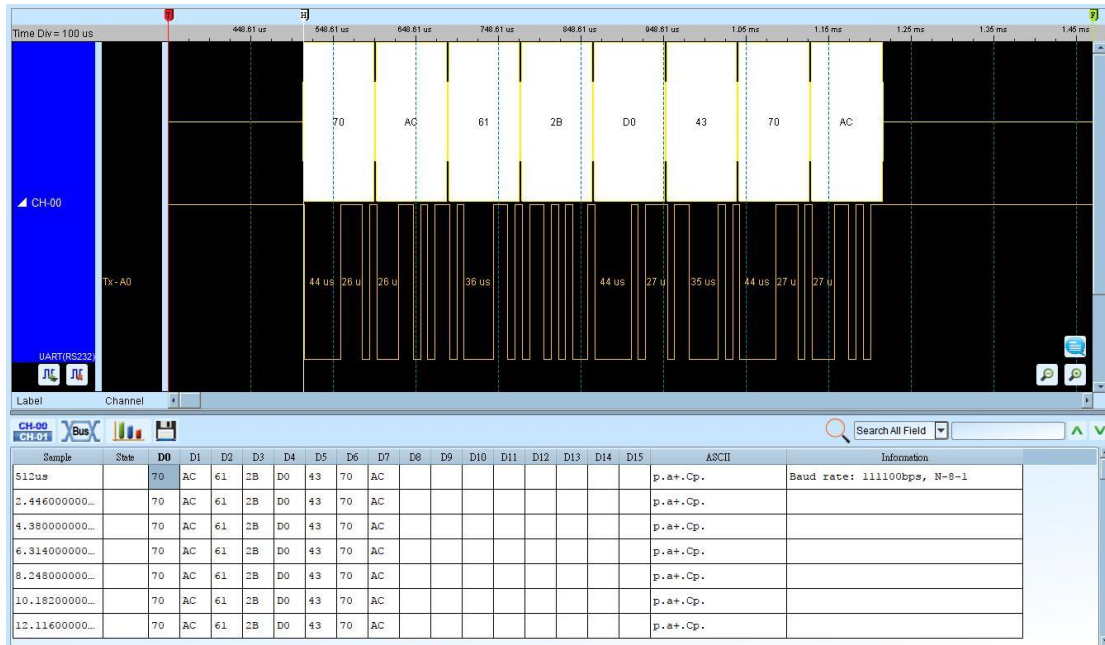
Report Unknown and Idle: Display the unknown and idle data in the Report Window.

Show scale in the waveform: Display the waveforms with scales.

Line Wrap Data: Use 1-2 value (hex) as header of data.

Result

Click **OK** to run UART Decode and see result on the Waveform Window below.



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:

Channel setting: Set the ULPI channel. And you can use Quick Setting to quickly set the data pin.

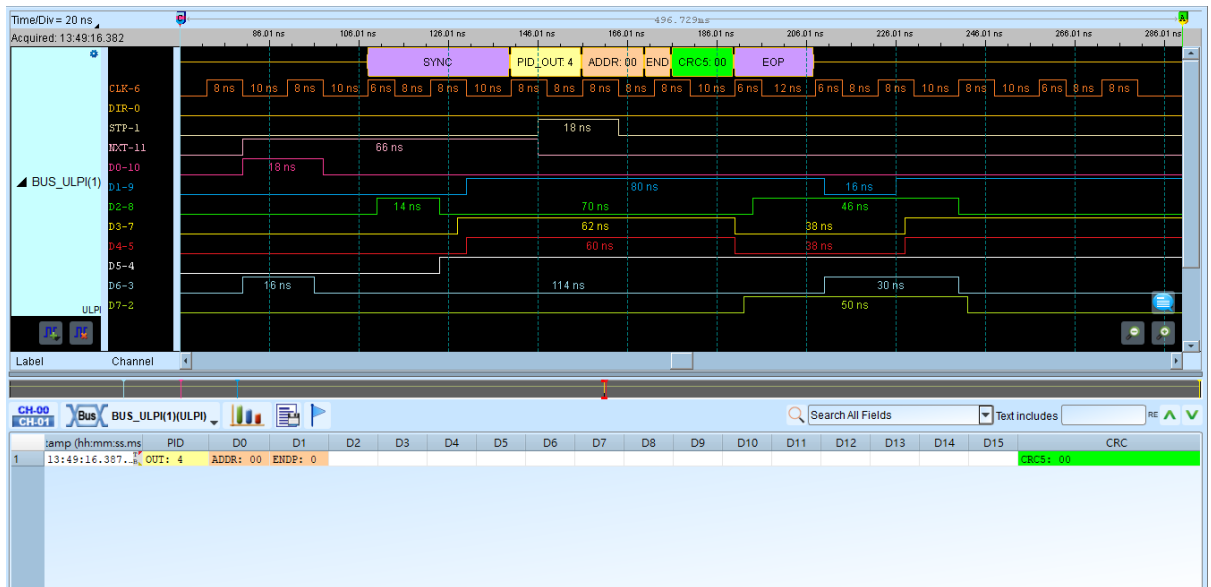
Decode: Set the decoding method of ULPI, optional ULPI / USB (analog timing).

Result

ULPI



USB



UNI/O

The UNI/O interface was developed by Microchip. The data transfer rate is 10Kbps to 100Kbps.

Settings

Channel: Show the selected channels (SCIO CH0).

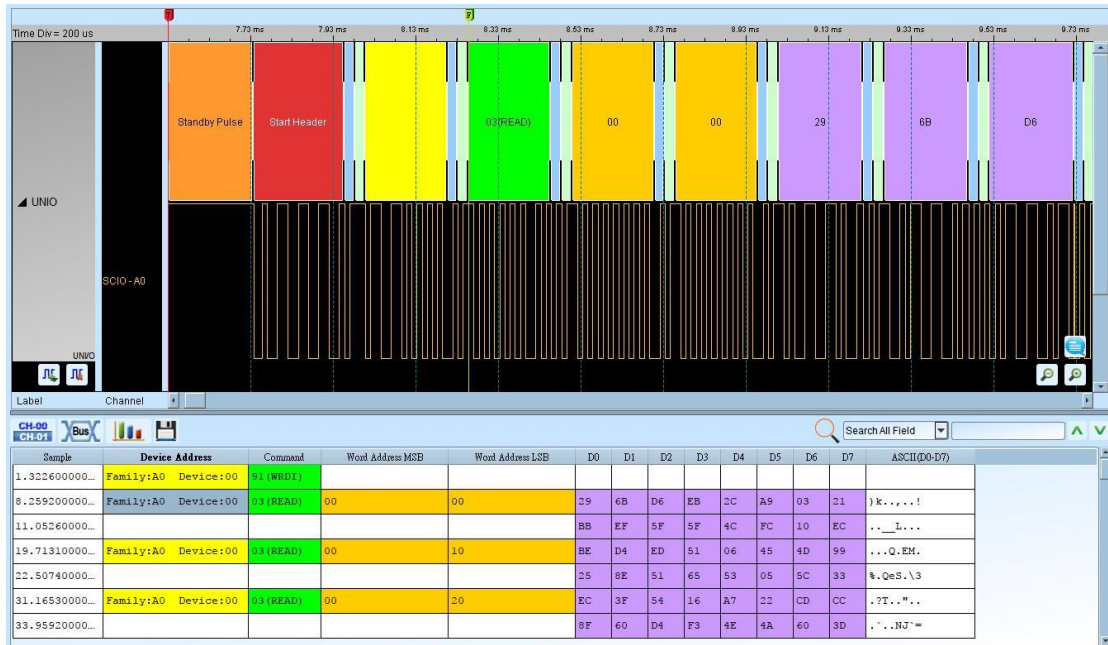
Device Address: Set data bits for the device address.

Tolerance: Set the Input / Output Edge Jitter Tolerance, $\pm 10\%$ / $\pm 25\%$ default.

Report Setting: To show the data by 8 or 16 columns in the Report Window.

Result

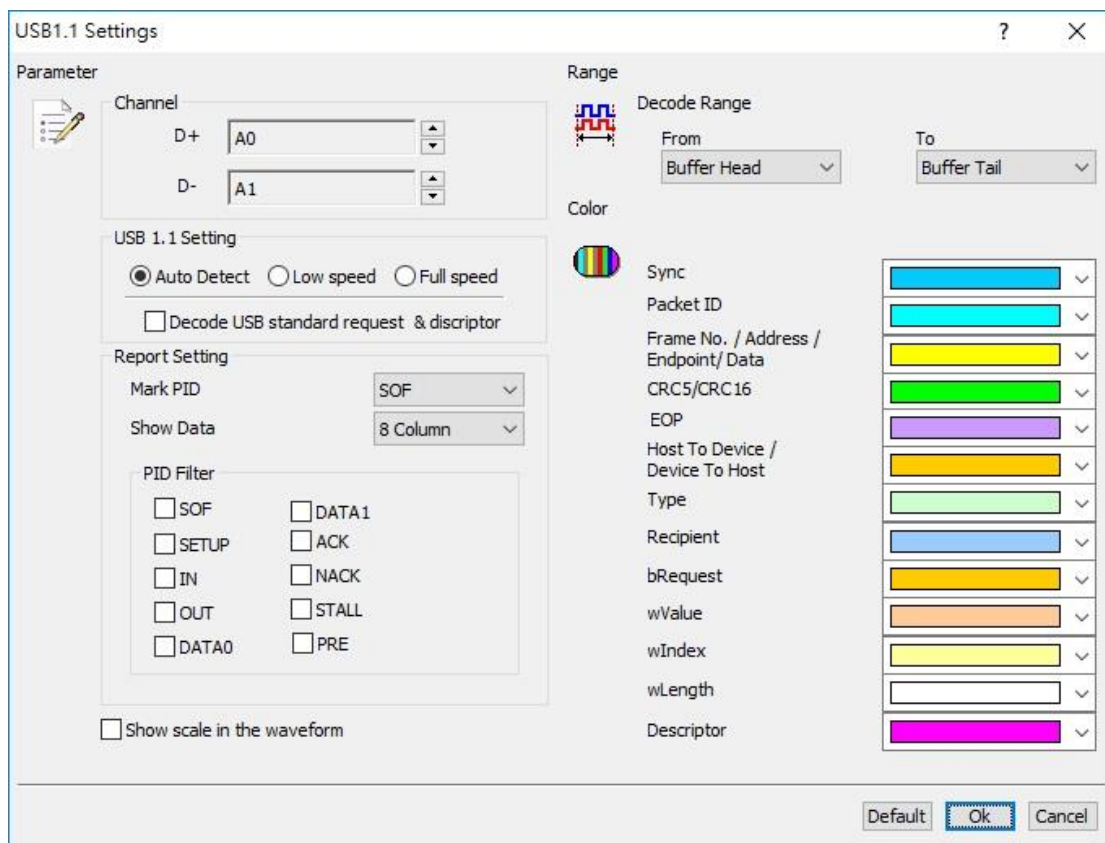
Click **OK** to run the UNI/O decode and see the result on the Waveform Window below.



USB1.1

The USB 1.0 specification was introduced in 1994. USB signals are transmitted on a twisted pair data cable with 90 Ohm $\pm 15\%$ impedance, labeled D+ and D-.

Settings

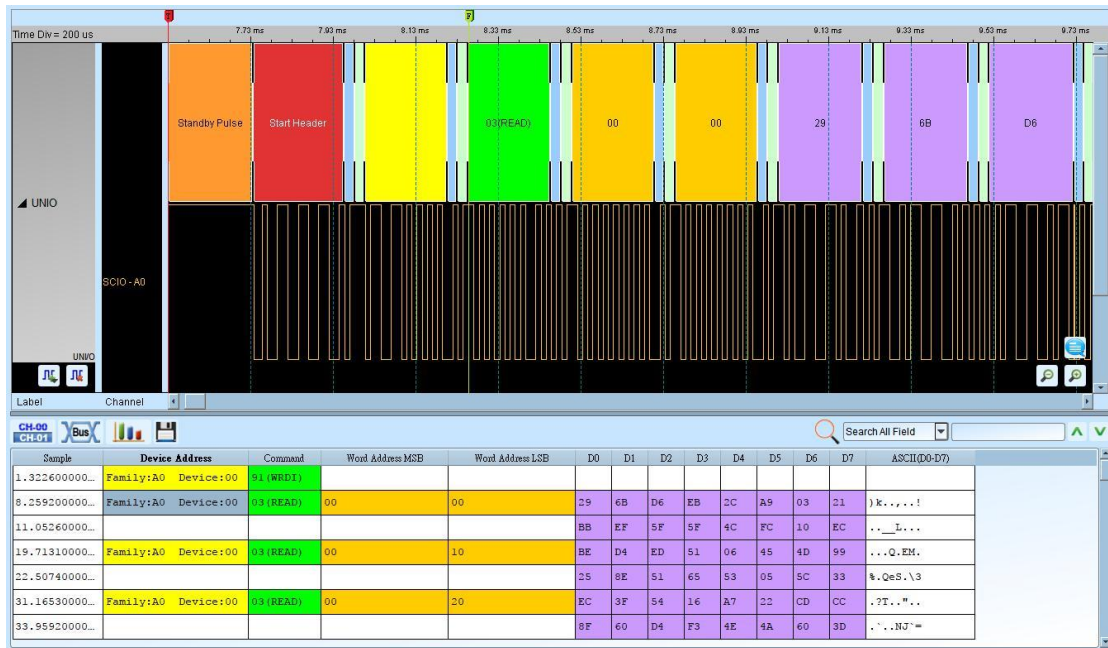


Channel: Show the selected channels (D+ CH0 and D- CH1).

USB1.1 Setting: Select USB state (low speed or full speed) and decode the USB standard request and descriptor.

Result

Click **OK** to run the USB1.1 decode and see the result on the Waveform Window.



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:

USB4/TBT3 SBChannel Settings

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:

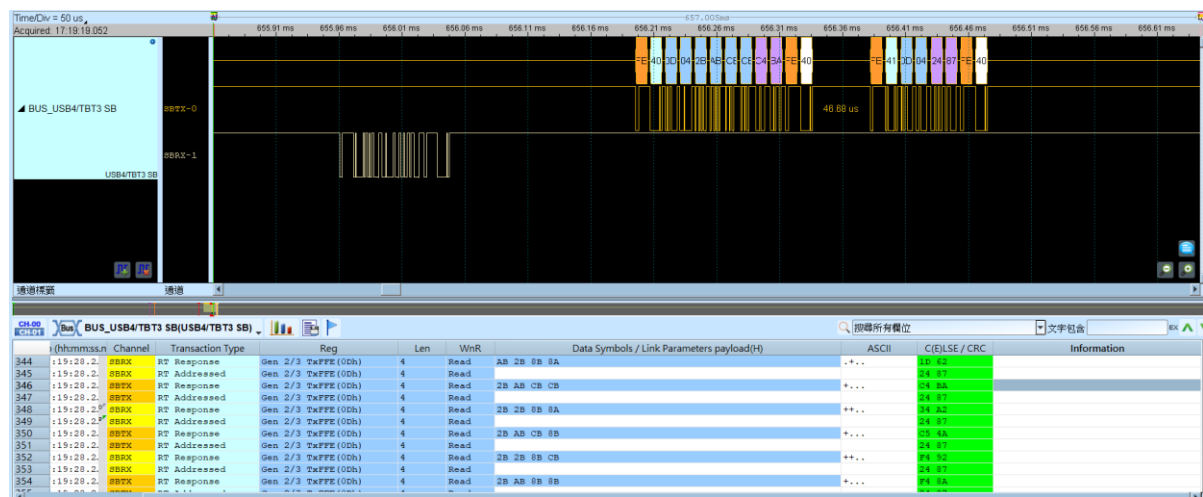
Version: USB4 2.0 is checked by default, and different versions can be selected for USB4/TBT3 SB signal analysis.

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

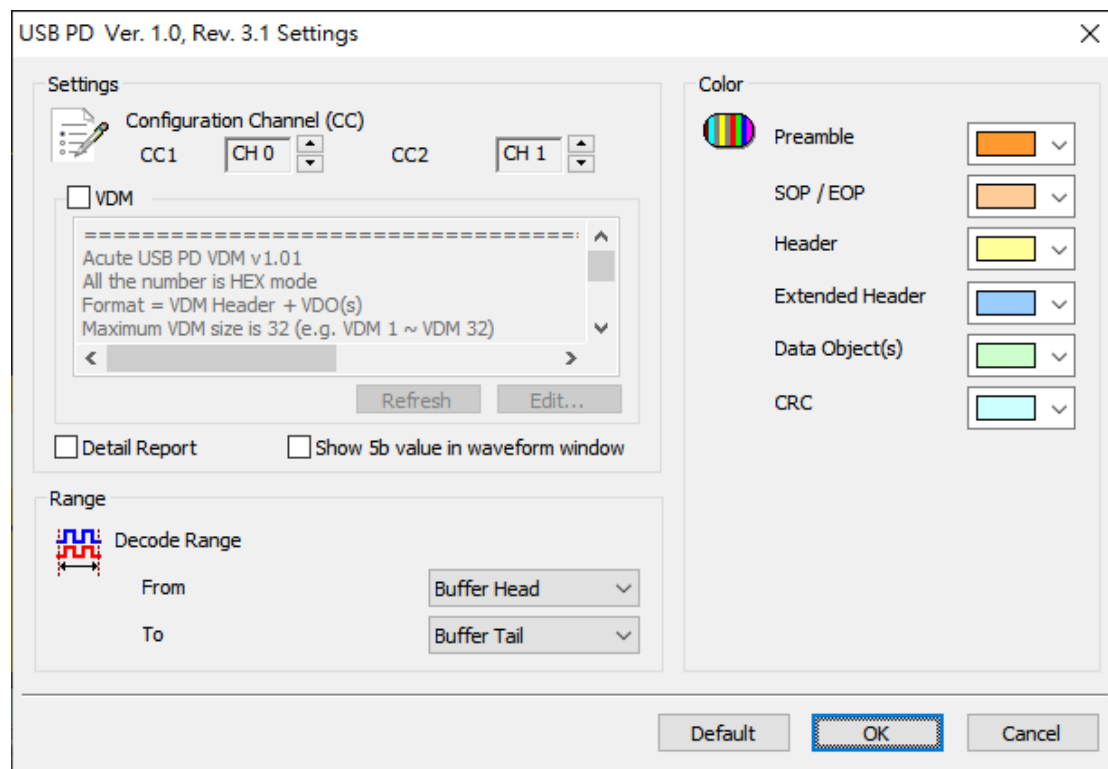
2. Result:



USB PD 2.0/3.0

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



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

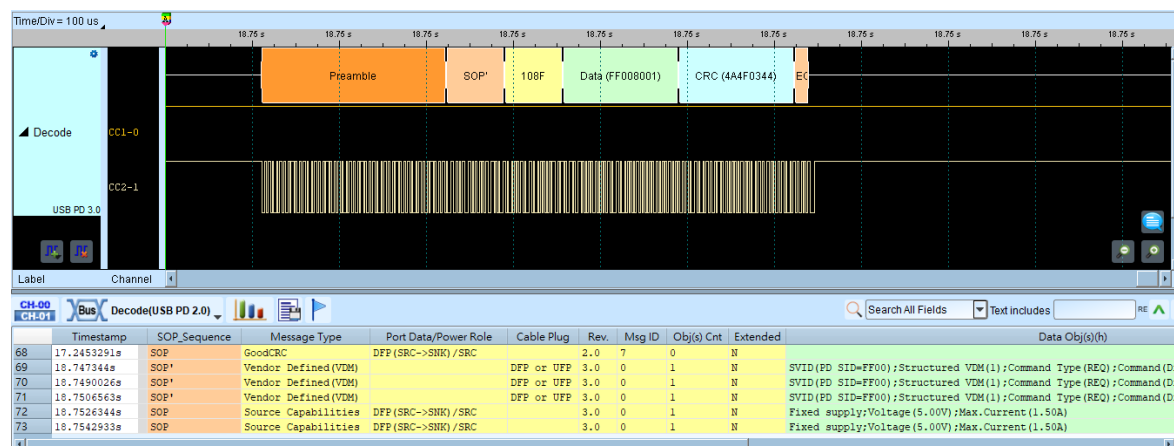
DISCOVER_BUTTONS.

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



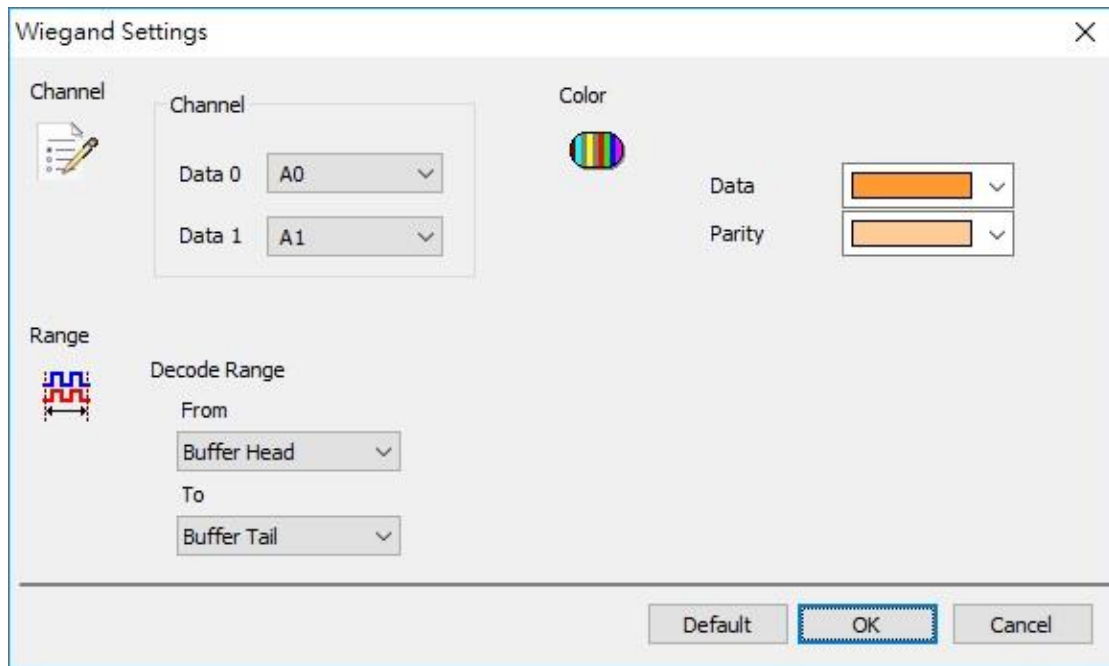
Accessories:

http://www.acute.com.tw/eng/p3_product_detail.php?cid=13&tid=29

Wiegand

It is commonly used to connect a card swipe mechanism to the rest of an electronic entry system..

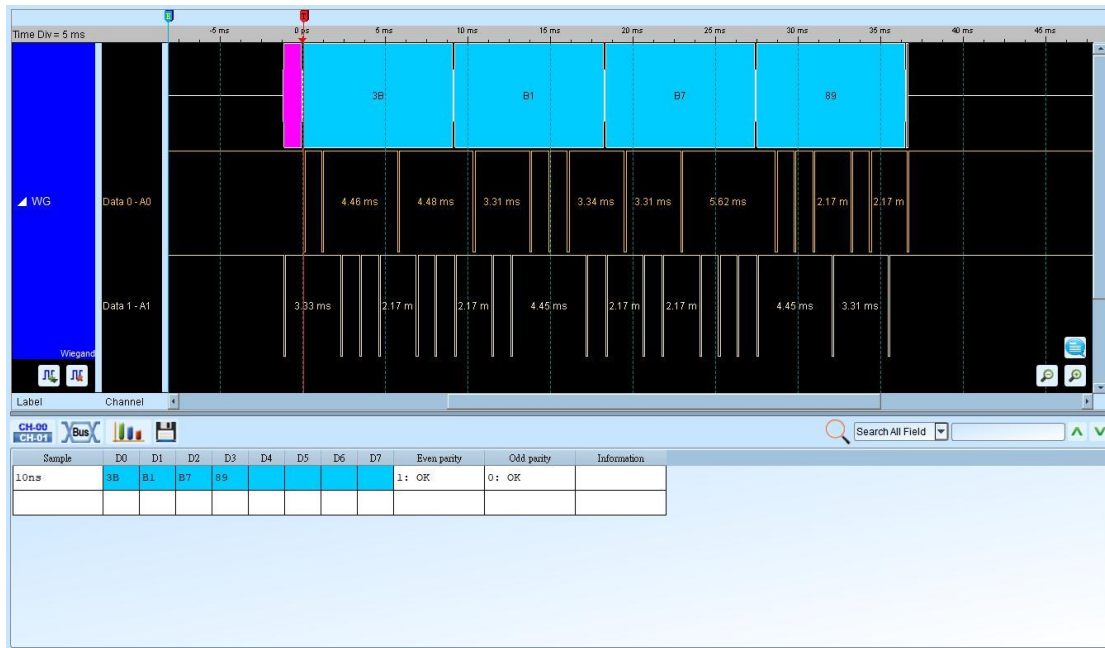
Settings



Channel: Show the selected channels (Data 0 and Data 1).

Result

Click **OK** to run the wiegand decode and see the result on the Waveform Window.



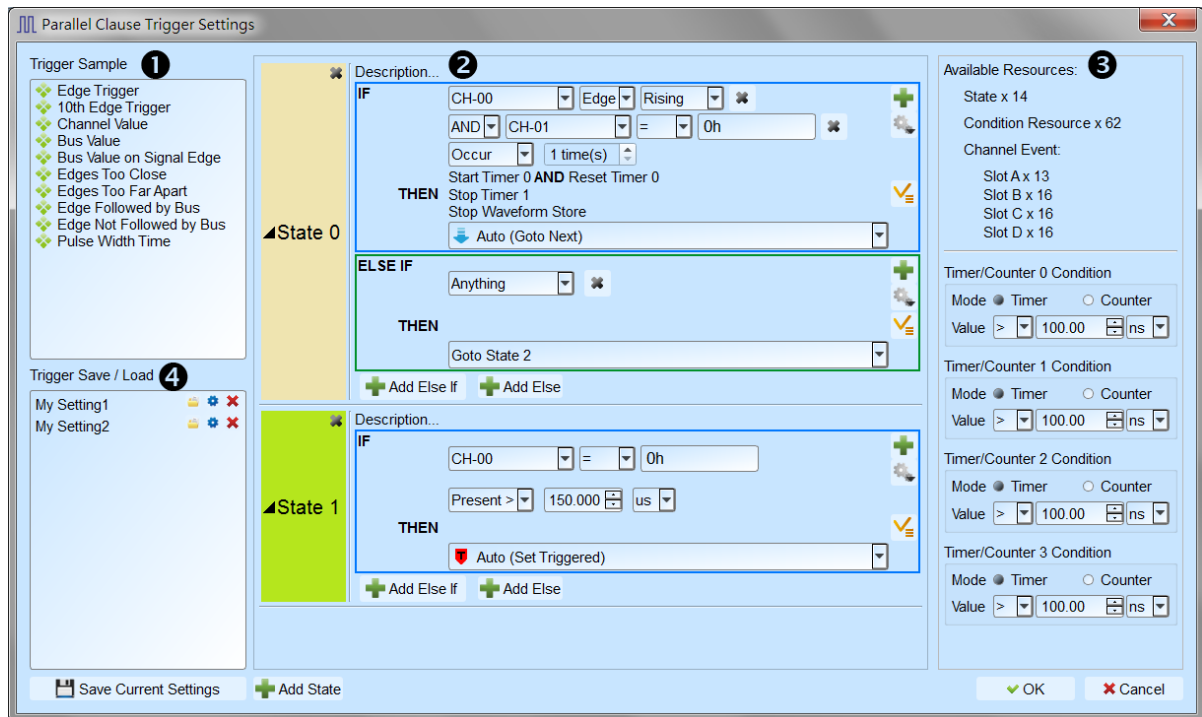
Chapter 2 Bus Trigger

Bus Trigger

The bus trigger are only available for the TravelLogic Series.

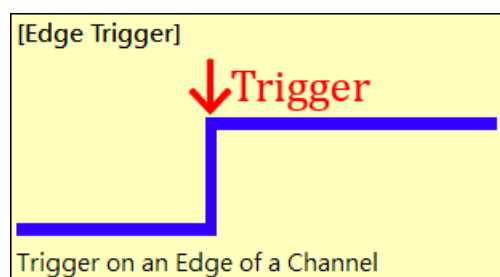
Bus Trigger	TravelLogic B+	TravelLogic B	TravelLogic E	TravelLogic
CAN	⊙	⊙		⊙
eSPI	⊙			
I2C	⊙	⊙		⊙
I2S	⊙	⊙	⊙	⊙
LIN	⊙	⊙		
LPC	⊙	⊙		
MIPI SPMI	⊙			
NAND Flash	⊙			
SD/eMMC	⊙			
Serial Flash	⊙			
SMBus/PMBus	⊙	⊙		
SPI	⊙	⊙		⊙
SVI2	⊙	⊙		
SVID	⊙	⊙		⊙
UART	⊙	⊙		⊙
USB 1.1	⊙	⊙		

Parallel Clause Trigger

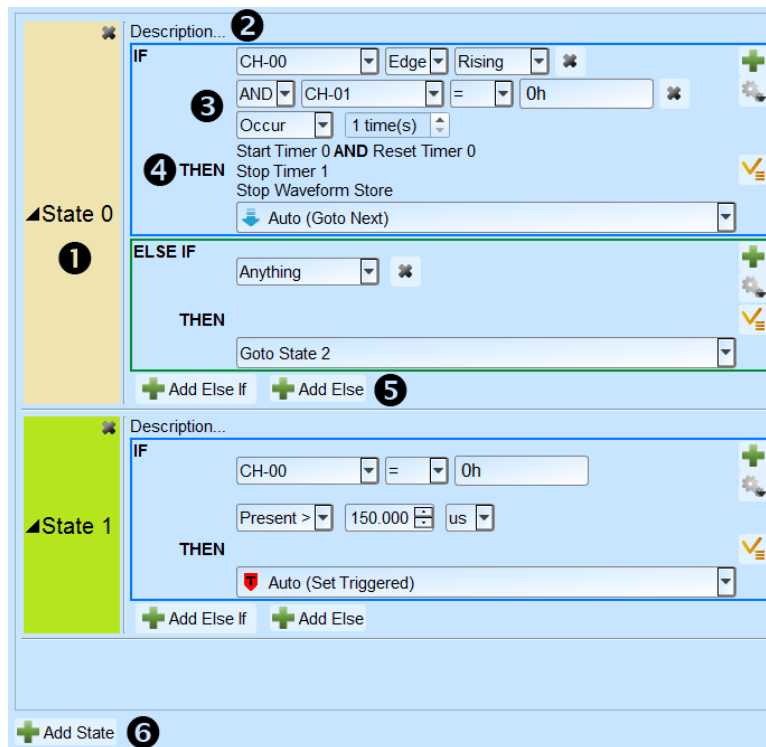


Parallel Clause Trigger feature provides 16 States, 64 Condition Resources and 4 Timer/Counter conditions to help user locate the target waveform, it is also available to set State branch control flow for each IF Clause and decide whether to store waveform or not.

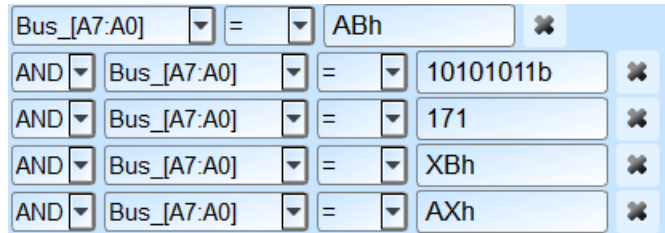
1. Trigger Sample: Provides several Trigger Samples for user's reference, users may also combine multiple trigger samples together for more complex trigger condition. Holding mouse cursor on each Sample for a while will pop up the sample description with schematic picture.




2. Trigger condition settings:




- 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.




- 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.

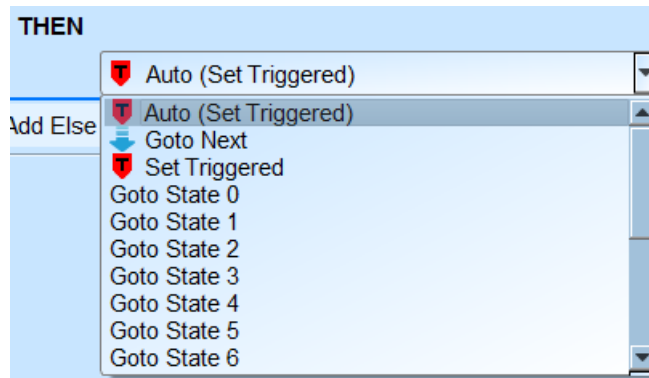
 Advanced control button, including:

 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.



✓ Additional Then action: Setup Timer/Counter Start/Stop/Reset actions or Waveform Store Suspend/Resume for each IF Clause when condition matched.

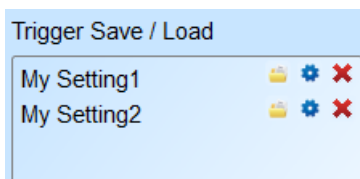
- ⑤ Add new ELSE IF / ELSE Clause: Click to add new ELSE IF / ELSE Clause, each clause condition will be checked from top to bottom, and perform the specified Action and Branch if the clause condition is matched. New clause conditions will consume Condition Resources. ELSE Clause condition will be set 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. Trigger 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.



 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.

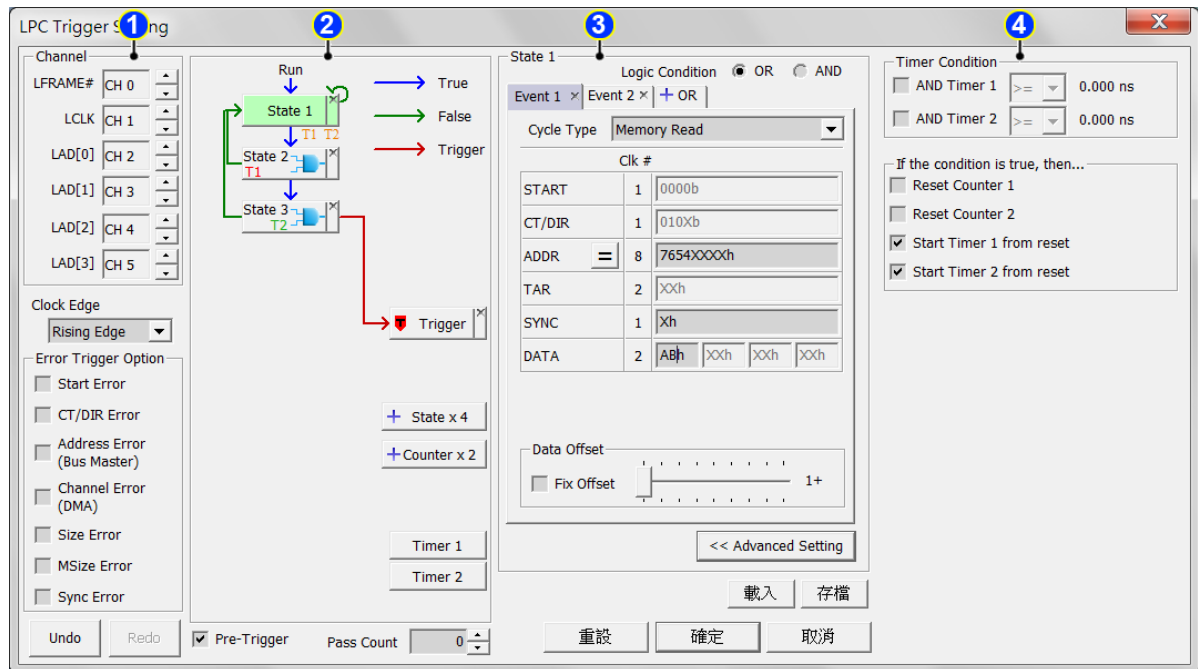
 Edit the display name for the selected Trigger settings.

 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

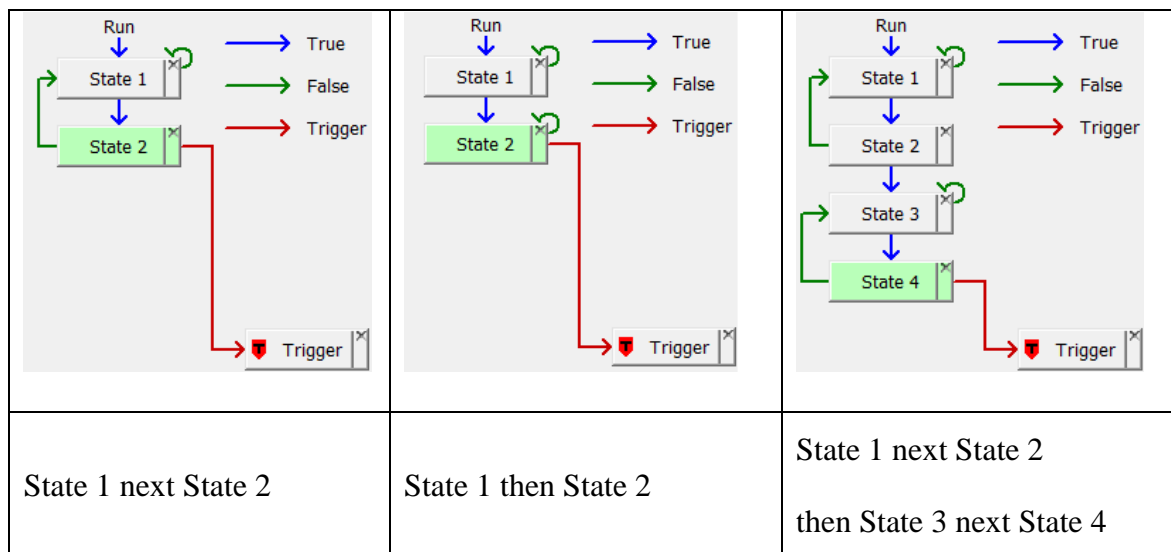
The Clause trigger settings dialog box is as below.

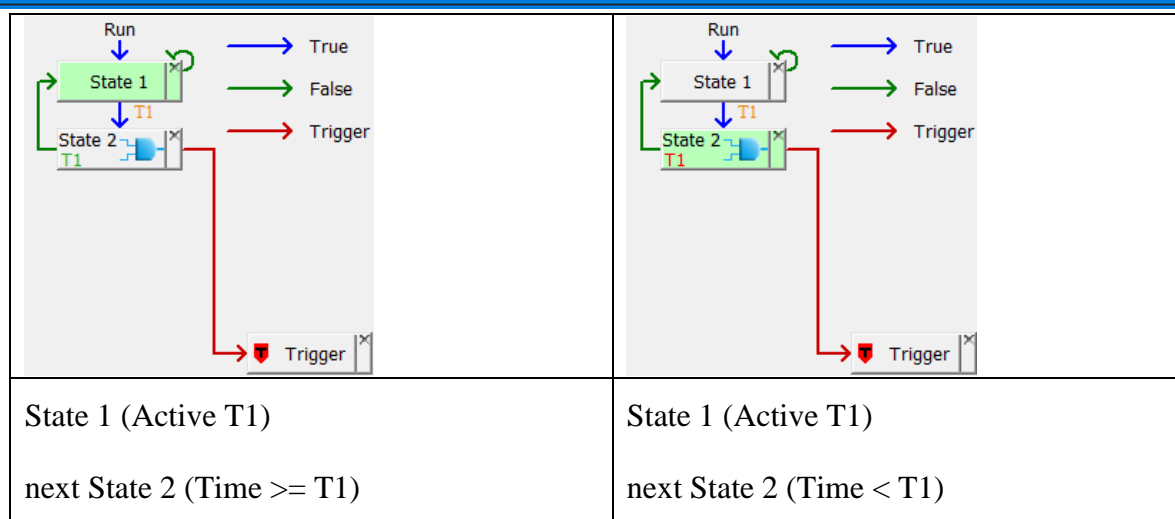


1. Channel and bus trigger settings:

Please refer to the descriptions in each bus trigger settings section.

2. Flow chart






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:

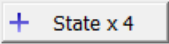
Click state 1 and state 2, then state 1 will branch to state 2 while state 1 is false.

Double click any state, the state will branch to itself if it is false.

Click the state button and trigger button (): trigger when the state conditions is satisfied.

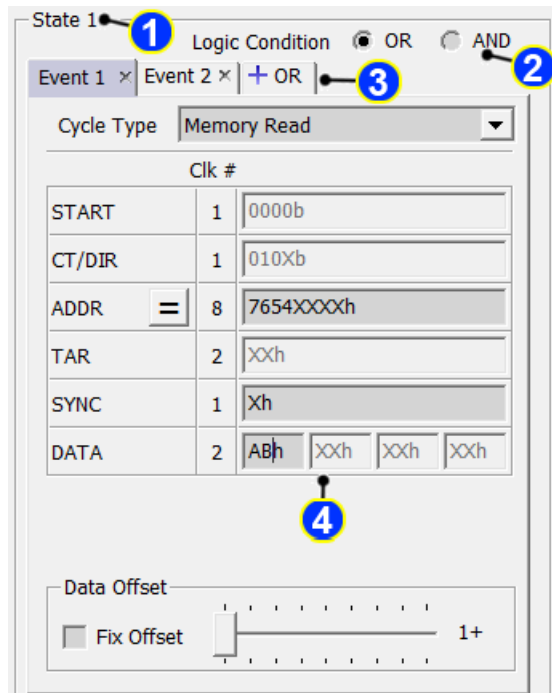
Delete any state by clicking the cross sign on the state.

Clear the links to the trigger button by clicking on the trigger button ()

Click  and  to add a new state.

 : The timer ranges from 5 ns to 8 days.

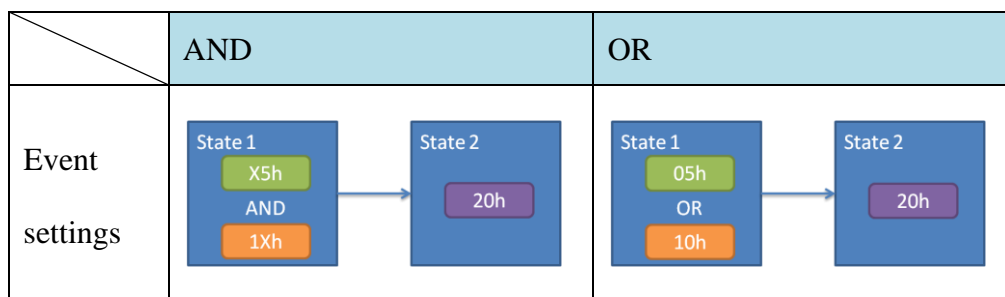
3. State settings

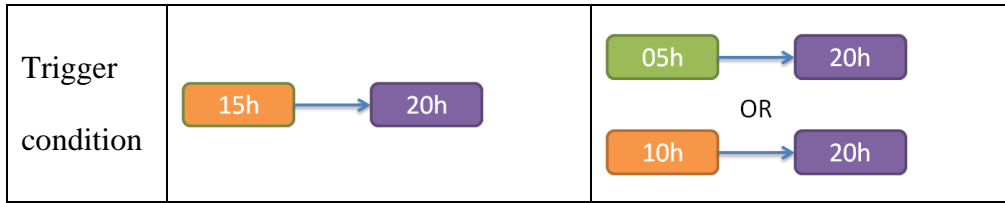


The trigger conditions of the state:

1. The current state index
2. Logic condition between every event in the state.
3. Switch between the tab windows to edit the trigger events . Click **+ OR** / **+AND** to add new events (up to 8 events). All inputs are supported binary(e.g. **01000001b**)/decimal(e.g. **65**) / hexadecimal(e.g. **41h**)
4. The bus trigger settings.

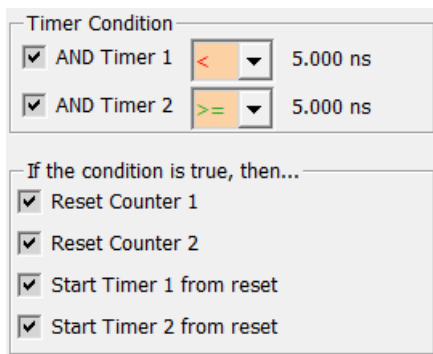
The relations between events and the trigger condition:





4. Timer and Counter settings

Press Advanced Setting >> button to edit the timer/counter reset settings of the state.



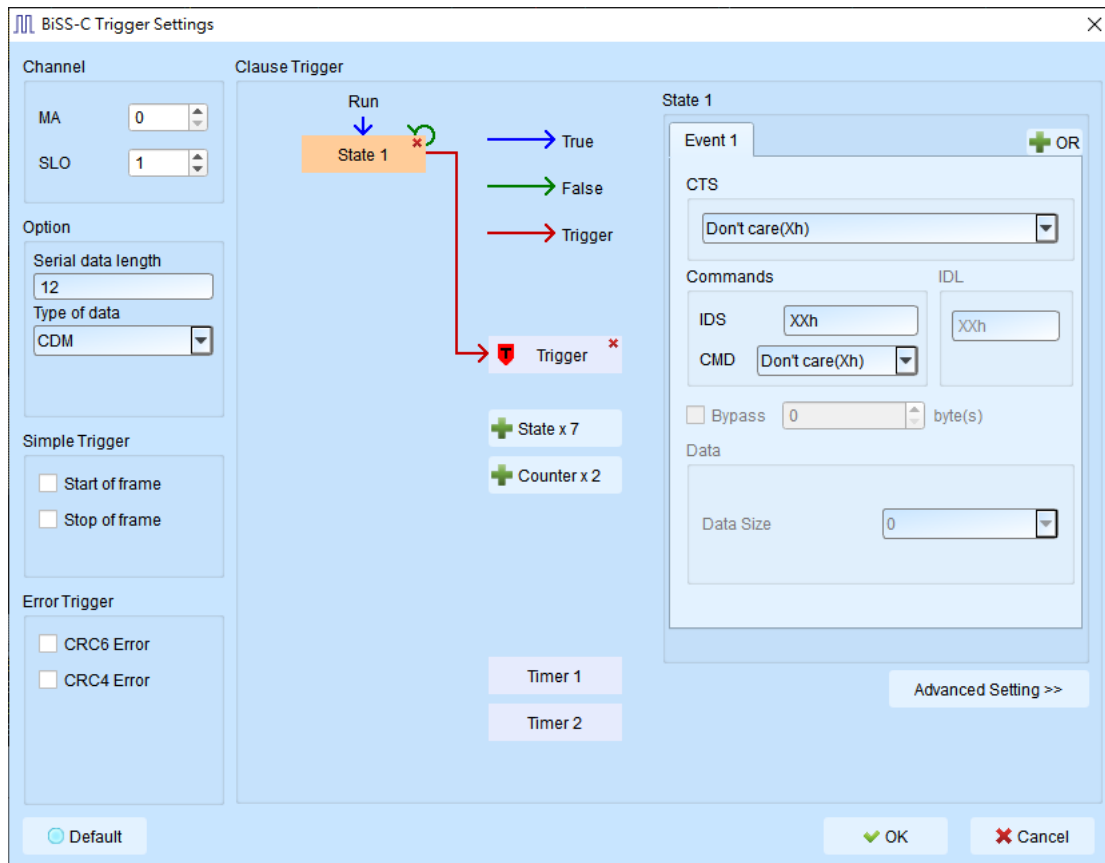
The following table describes the state button icons:

Conditions	State 1	State 1 And timer > T1	State 1 And timer < T1	State 1 And T2 < timer < T1
satisfied condition	Start T1	X	Start T2 Reset C2	Start T1 and T2 Reset C1 and C2

BiSS-C Trigger

BiSS-C Trigger Settings

Click BiSS-C Trigger in the toolbar and will show the dialog as the following.



1. **Channel:** Select channels.
2. **Option:** Serial data length: Set SCD packet length, the default is 12 bits, 255 bits is the maximum.

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”.

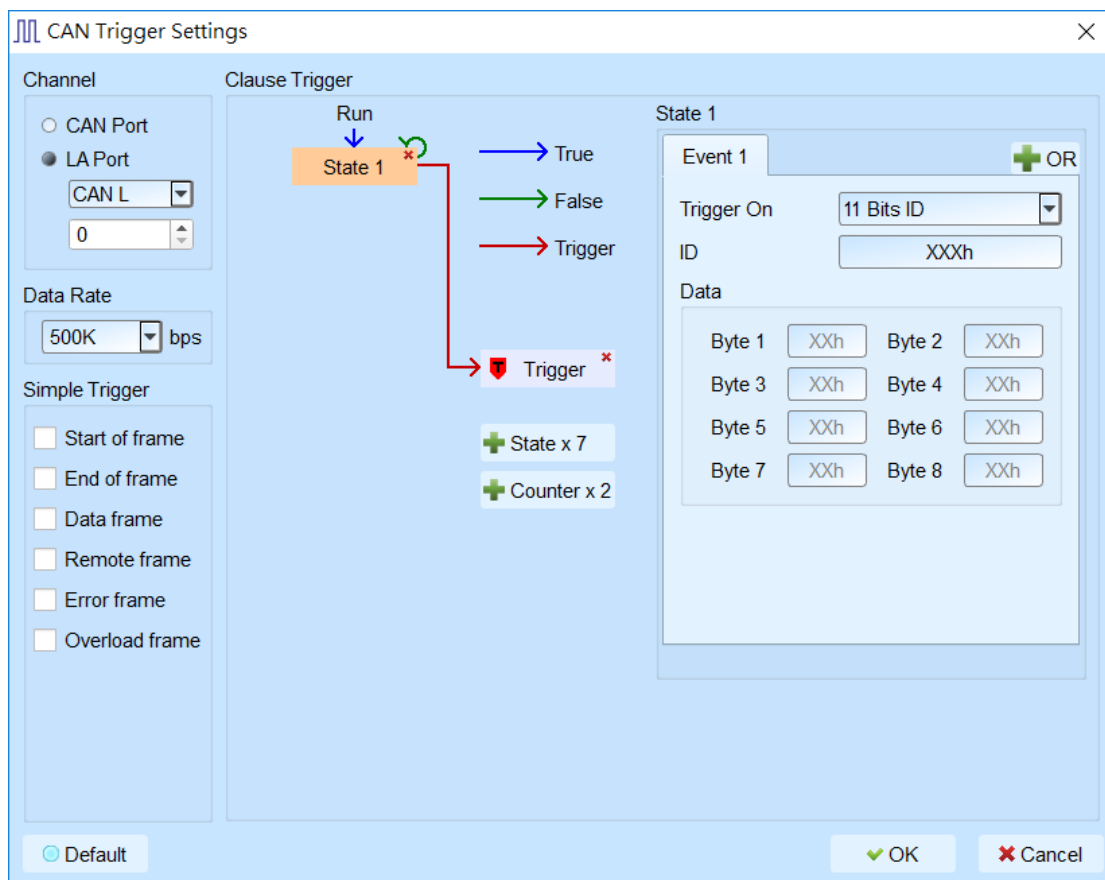
CAN Trigger

Supported Models : TB1016B. TB1016B+. TL3134B. TL3234B+. LA3068B.

LA3136B

CAN Trigger Settings

Click CAN Trigger in the toolbar and will show the dialog as the following °



1. **Channel:** Select CAN_H or CAN_L as the trigger channel.
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”.

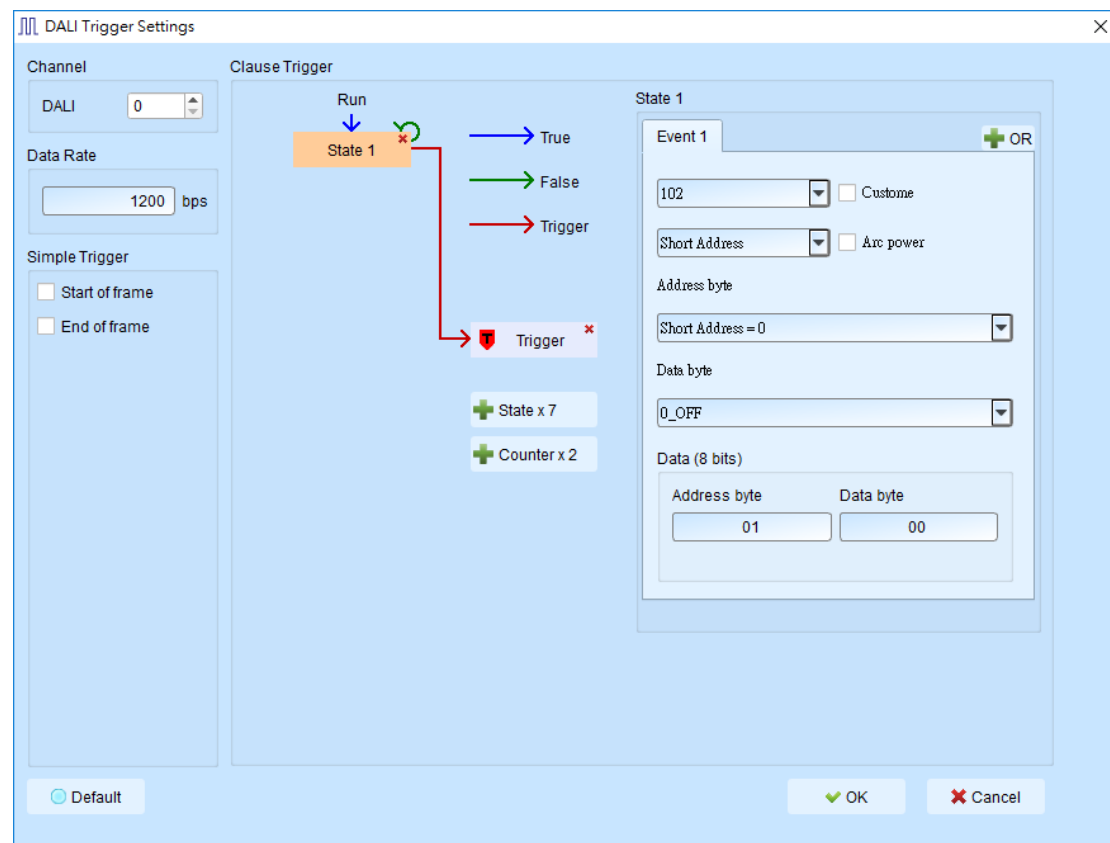
Note: Only TB1016B. TB1016B+ provide CAN port channel

DALI Trigger

Supported Models : TB1016E, TB1016B, TB1016B+, TL3134B, TL3234B+,
LA3068B, LA3136B

DALI Trigger Settings

Click DALI Trigger in the toolbar and will show the dialog as the following.



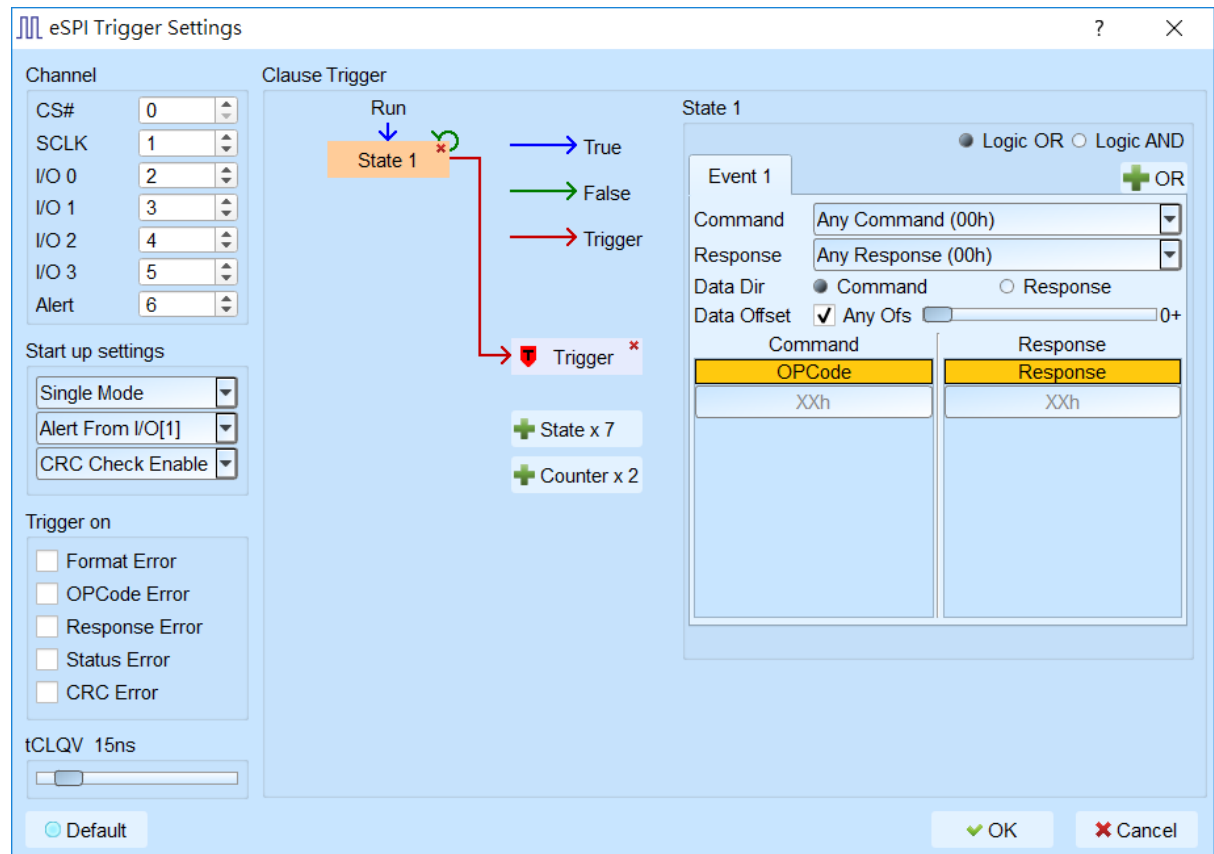
1. **Channel:** Select channels.
2. **Simple Trigger:** Trigger specific of DALI.
3. **Clause Trigger:** Please reference Clause Trigger chapter.
4. **State:** Show the details of trigger condition in every state as left side; selecting the trigger type of DALI (102, 103), command type (short address, group address, broadcast, response, special command), or custom cmd.

eSPI Trigger

Supported Models : TL3234B+. LA3068B. LA3136B

eSPI Trigger Settings

Click eSPI Trigger in the toolbar and will show the dialog as the following.



5. **Channel:** Select channels.
6. **Startup settings:** Set the initial parameters of eSPI.
7. **Trigger on:** Trigger specific error of eSPI.
8. **Clause Trigger:** Please reference Clause Trigger chapter.
9. **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.

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											55	
0.00000066 S	ACCEPT(08)					13	11	00	00					030F	95	
0.0000003 S	SET_CONFIGURATION(22)				0010	01	11	00	00							96	
0.00000935 S	ACCEPT(08)															97	
0.000008455 S	GET_STATUS(25)															98	
0.000009365 S	ACCEPT(08)															99	
0.001601195 S	GET_CONFIGURATION(21)				0010											55	
0.001602795 S	ACCEPT(08)					13	11	00	00					030F	95	
0.001604635 S	SET_CONFIGURATION(22)				0010	01	11	00	00							95	
0.001609575 S	ACCEPT(08)															96	

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 signal 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.

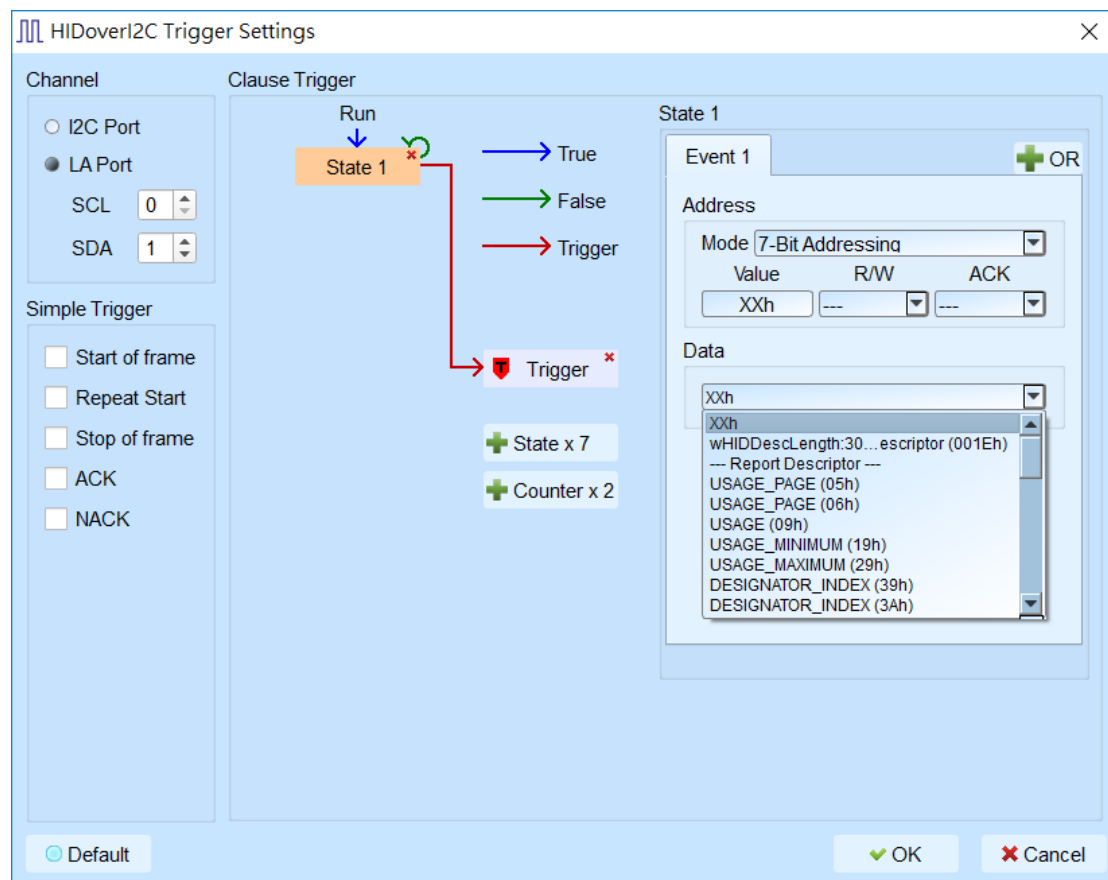
HIDoverI2C Trigger

Supported Models : TB1016E. TB1016B. TB1016B+. TL3134B. TL3234B+.

LA3068B. LA3136B

HIDoverI2C Trigger Settings

Click HIDoverI2C Trigger in the toolbar and will show the dialog as the following.



1. **Channel:** Select I2C / LA port.
2. **Simple Trigger:** Specific trigger function of I2C.
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 R/W, ACK and Data(HID Descriptor) fields, default value is XX means “don’t care”. When trigger the descriptor, R/W field

will be selected Read state automatically.

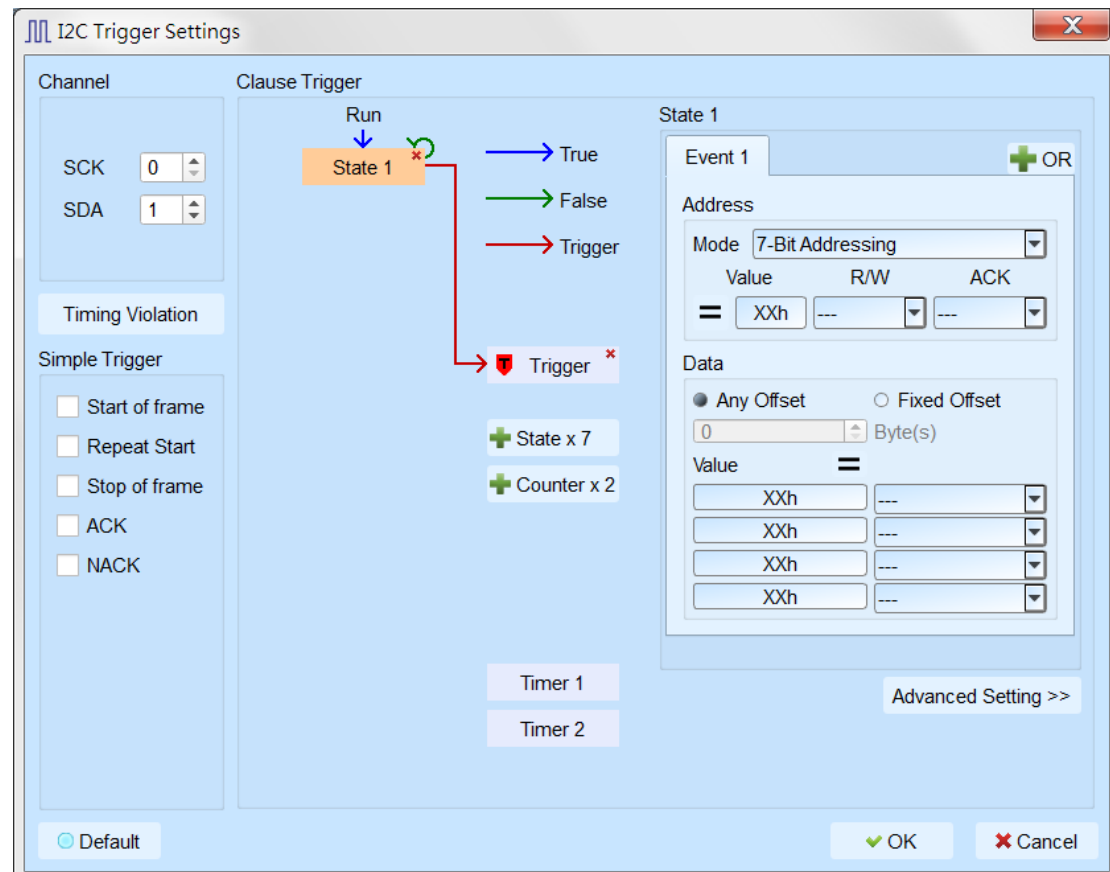
Note: Only TravelBus series provide I2C port channel

I²C Trigger

Supported Models : TravelBus series. TravelLogic series. Logic Analyzer series

I2C Trigger Settings

Click I2C Trigger in the toolbar and will show the dialog as the following.



1. **Channel:** Select I2C / LA port.
2. **Simple Trigger:** Specific trigger function of I2C.
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 R/W, ACK and Data fields, default value is XX means “don’t care”. Data field only provided 4 bytes data to trigger, checking the Fixed offset and selecting the offset value when want to trigger at the specific

position. It is also available to trigger on address or data not equal condition by clicking the “Equal” button.

Note: Only TravelBus series provide I2C port channel

I²S Trigger

Supported Models : TB1016E. TB1016B. TB1016B+. TL3134B. TL3234B+.

LA3068B. LA3136B

I2S Trigger Settings

Click I2S Trigger in the toolbar and will show the dialog as the following.

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 $-P < X \cup +P > X$.

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.

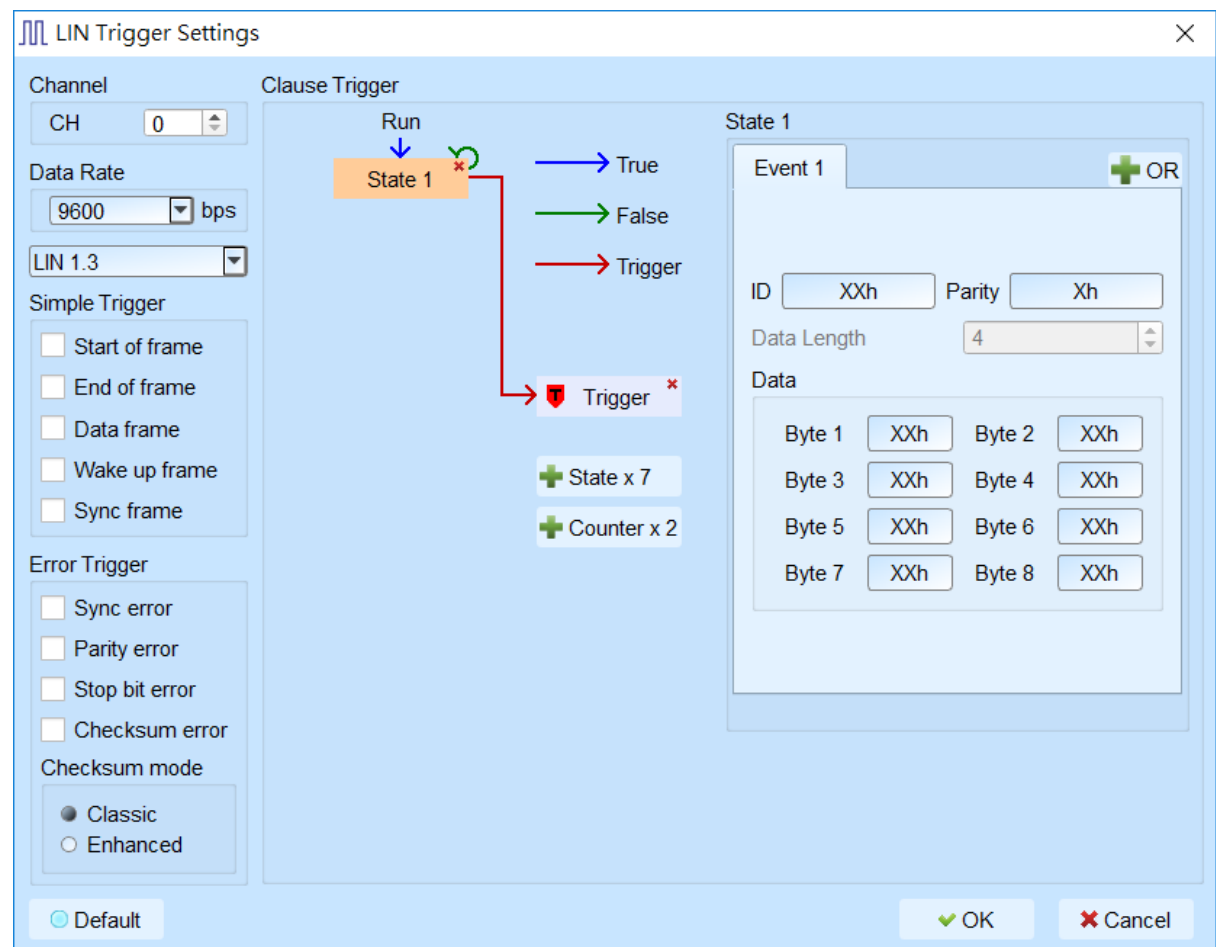
LIN Trigger

Supported Models : TB1016E. TB1016B. TB1016B+. TL3134B. TL3234B+.

LA3068B. LA3136B

LIN Trigger Settings

Click LIN Trigger in the toolbar and will show the dialog as the following.



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

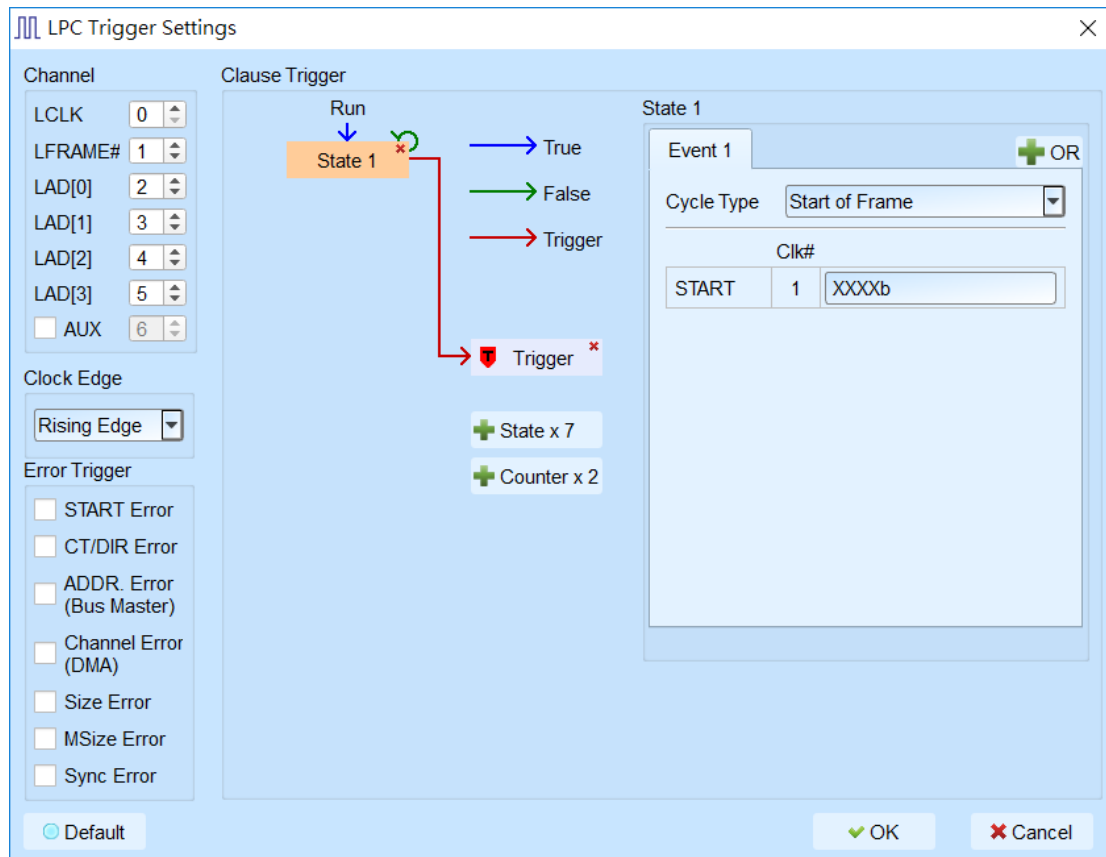
be enabled.

LPC Trigger

Supported Models : TL3134B. TL3234B+

LPC Trigger Settings

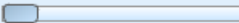
Click LPC Trigger in the toolbar and will show the dialog as the following.



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 $=$ / \neq / $>$ / \leq by click $=$ button.

Data offset setting:

Data Offset

Fix Offset  0+

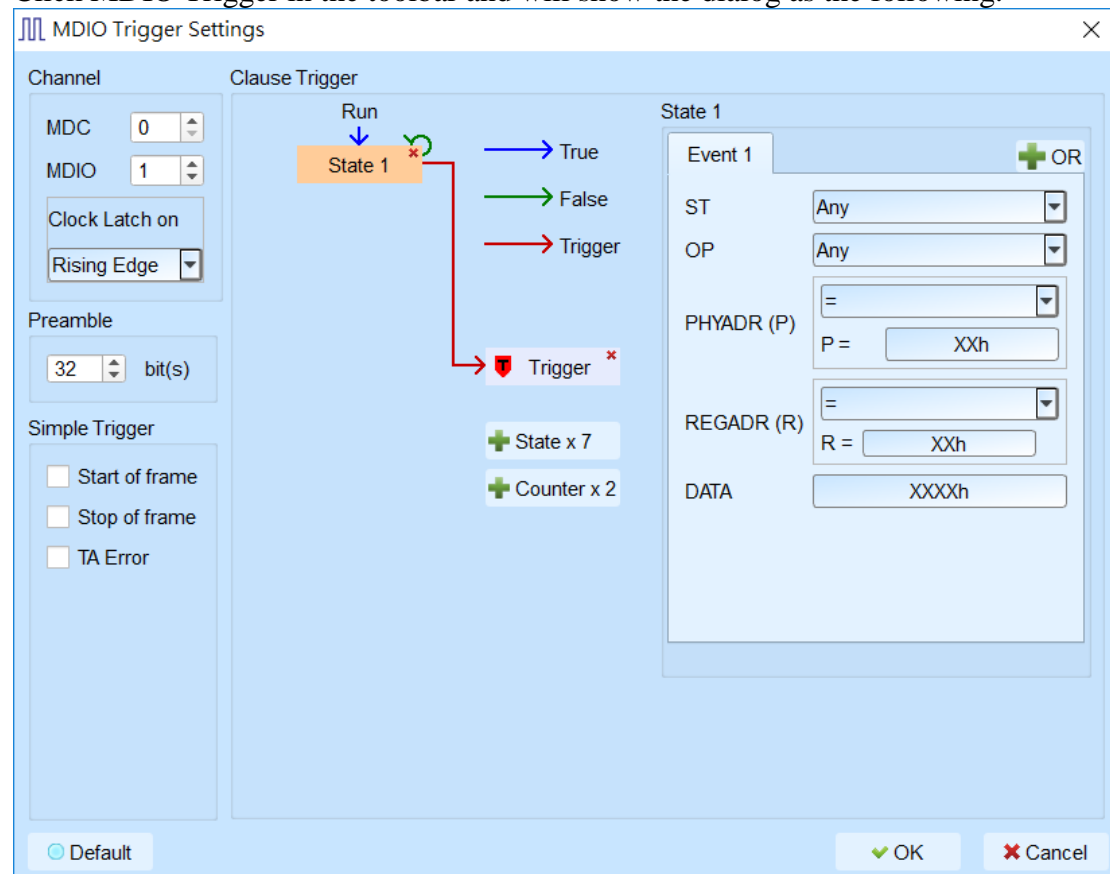
MDIO Trigger

Supported Models : TB1016E. TB1016B. TB1016B+. TL3134B. TL3234B+.

LA3068B. LA3136B

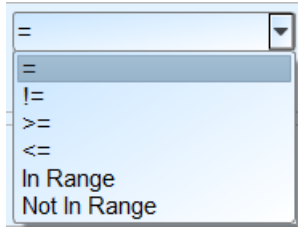
MDIO Trigger Settings

Click MDIO Trigger in the toolbar and will show the dialog as the following.



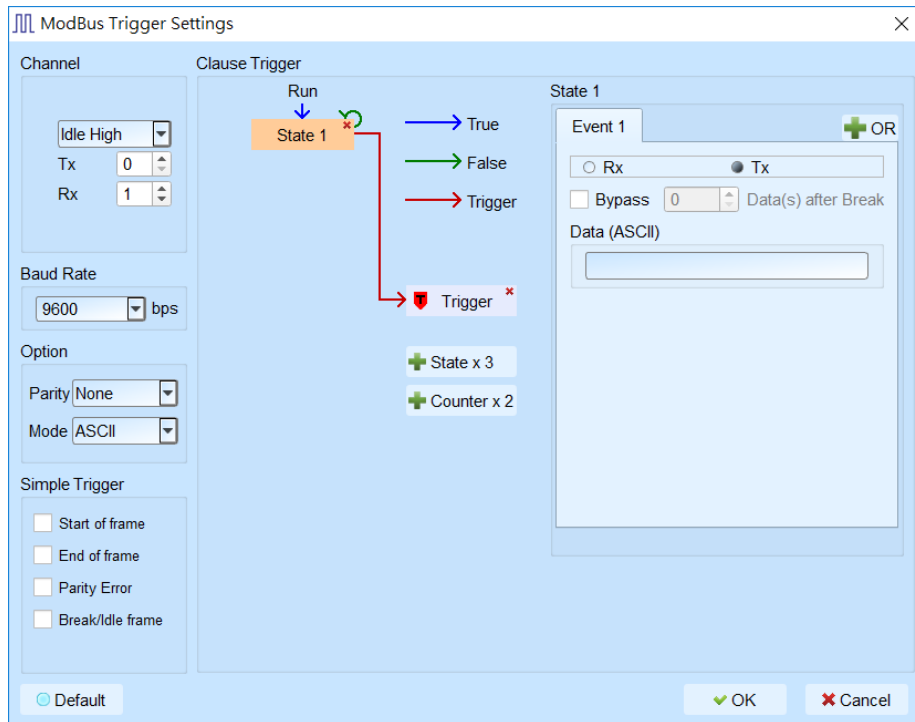
1. **Channel:** Select MDC / MDIO channels.
2. **Preamble:** Select the length of preamble.
3. **Simple Trigger:** Specific trigger function of MDIO.
4. **Clause Trigger:** Please reference Clause Trigger chapter.
5. **State:** Show the details of trigger condition in every state as left side; typing 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:

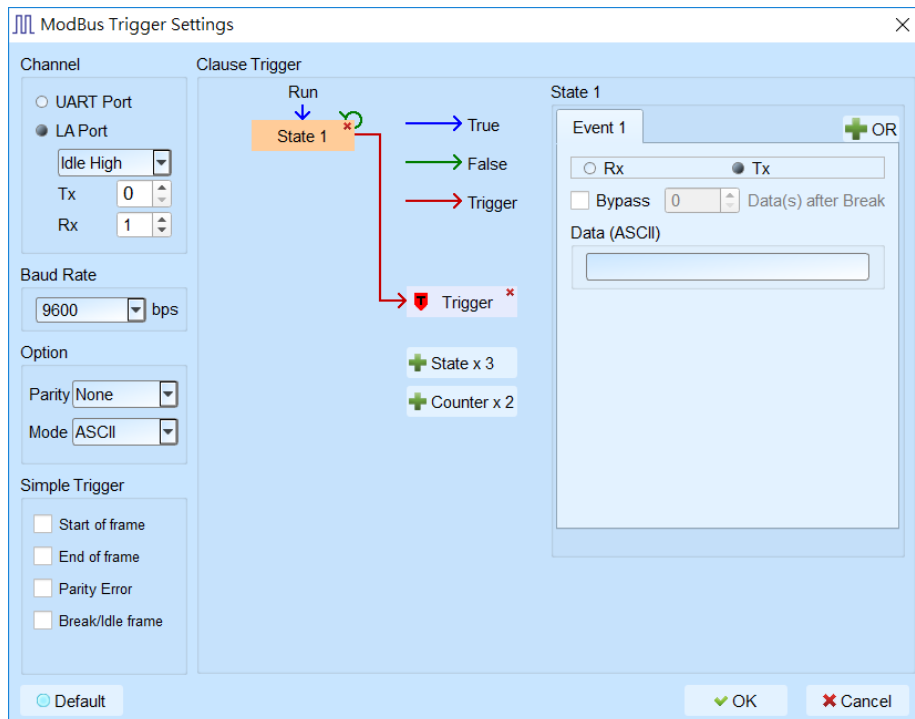


ModBus Logic Trigger

TL3K/LA3K Settings



TBA Settings



Channel

UART Port / LA Port Select ModBus channels.

Baud Rate

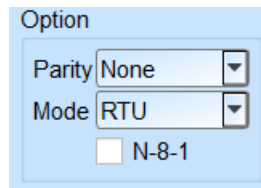
Select the baud rate and providing the manual input if no appropriate selection.

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.

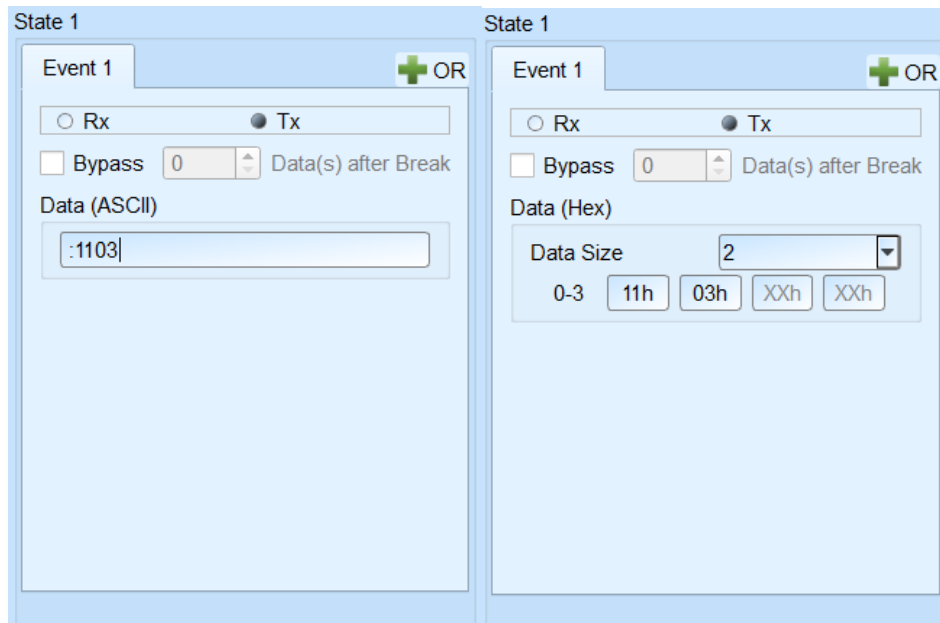


Simple Trigger Provide Start of frame, End of frame, Parity Error, Break/Idle frame triggers °

State Provide ASCII/HEX input mode by ASCII/RTU mode.

ASCII

RTU

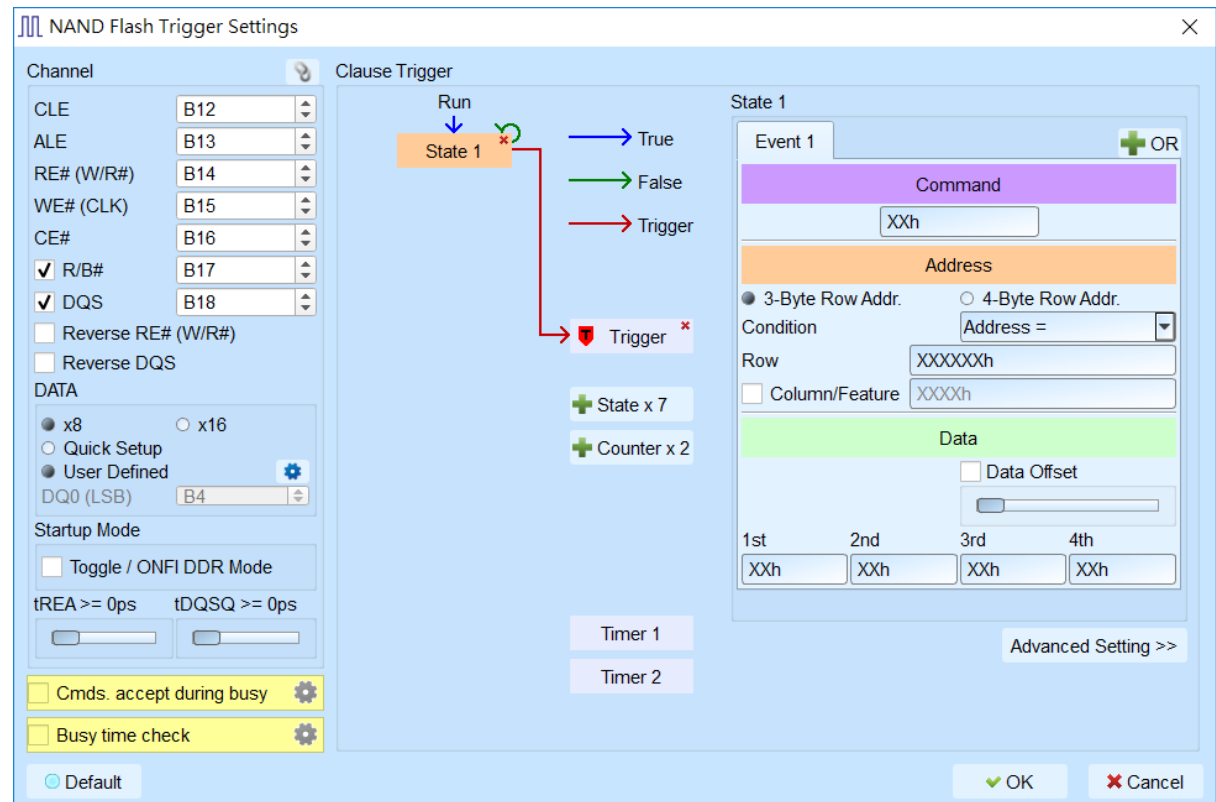


NAND Flash Trigger

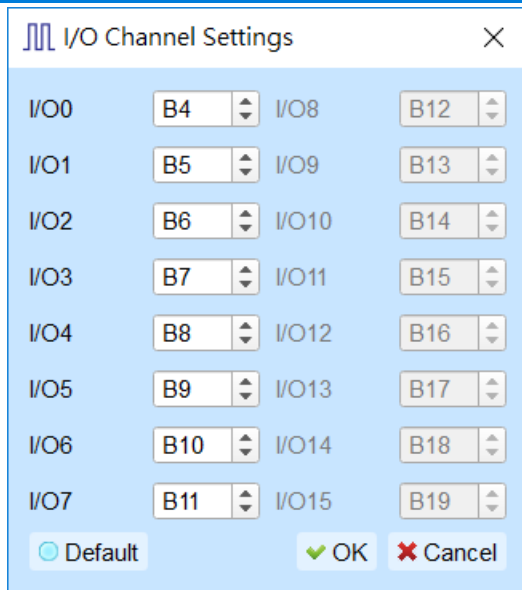
Supported Models : TL3234B+. LA3136B. BusFinder (NAND Flash solution)

NAND Flash Trigger Settings

Click NAND Flash Trigger in the toolbar and will show the dialog as the following.

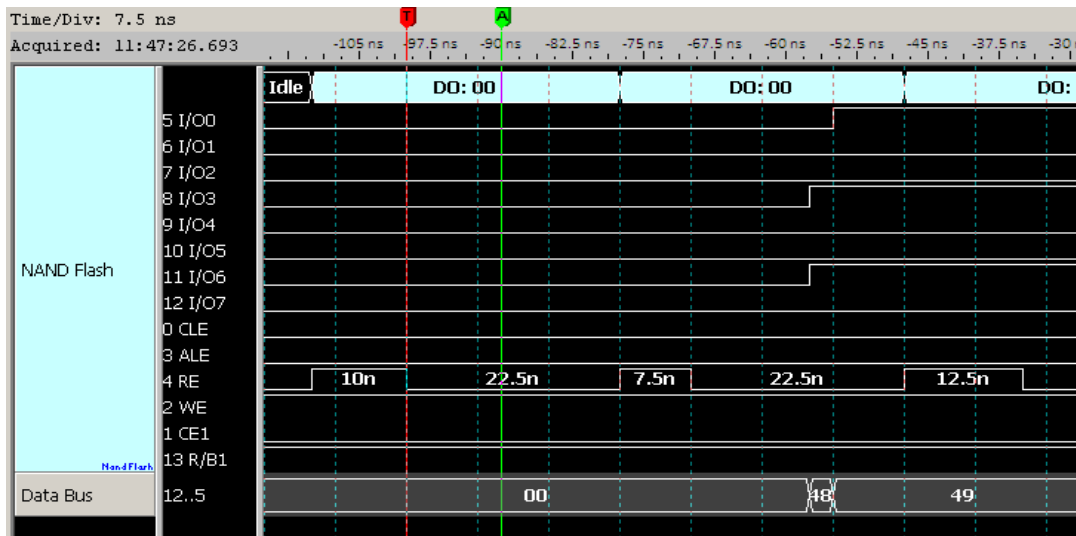


1. **Channel:** Select CLE, ALE, RE, WE, CE, R/B, DQS channels.
2. **DATA:** Select x8 or x16 bits
 - I. **Quick Setup:** Only set DQ0(LSB), others will be set automatically.
(e.g. LSB = B4, MSB = B11; LSB = B7, MSB = B14)
 - II. **User Defined:**

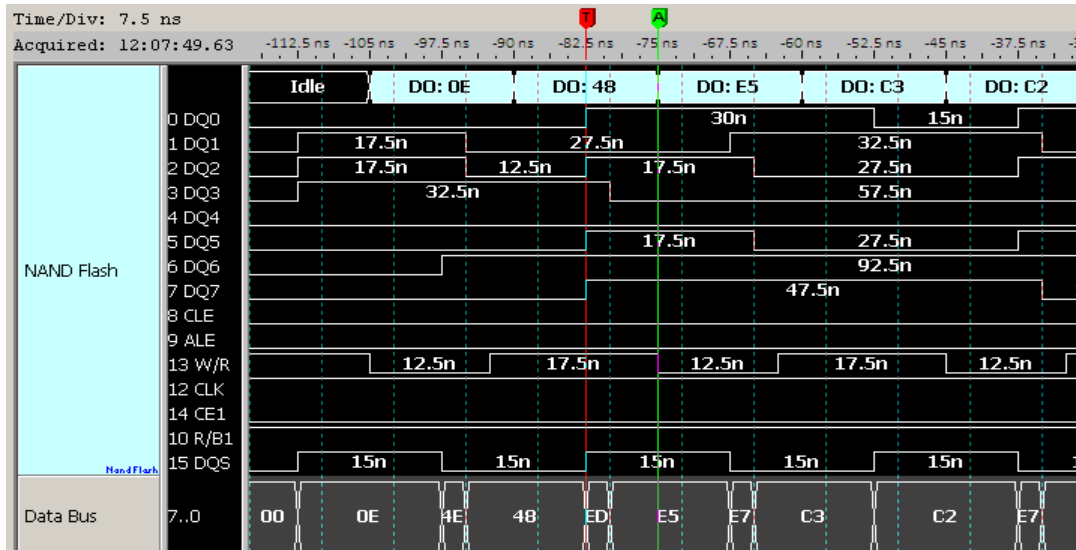


6. **Startup Mode:** Set DDR mode.
7. **tREA / tDQSQ:** Set tREA at SDR mode and tDQSQ at DDR mode.

tREA:



tDQSQ:



8. **Commands accepted during busy:** Set NAND commands still could be triggered during busy state, default is 70h/FFh/78h/7Bh.

Busy Command Settings ✕

Commands

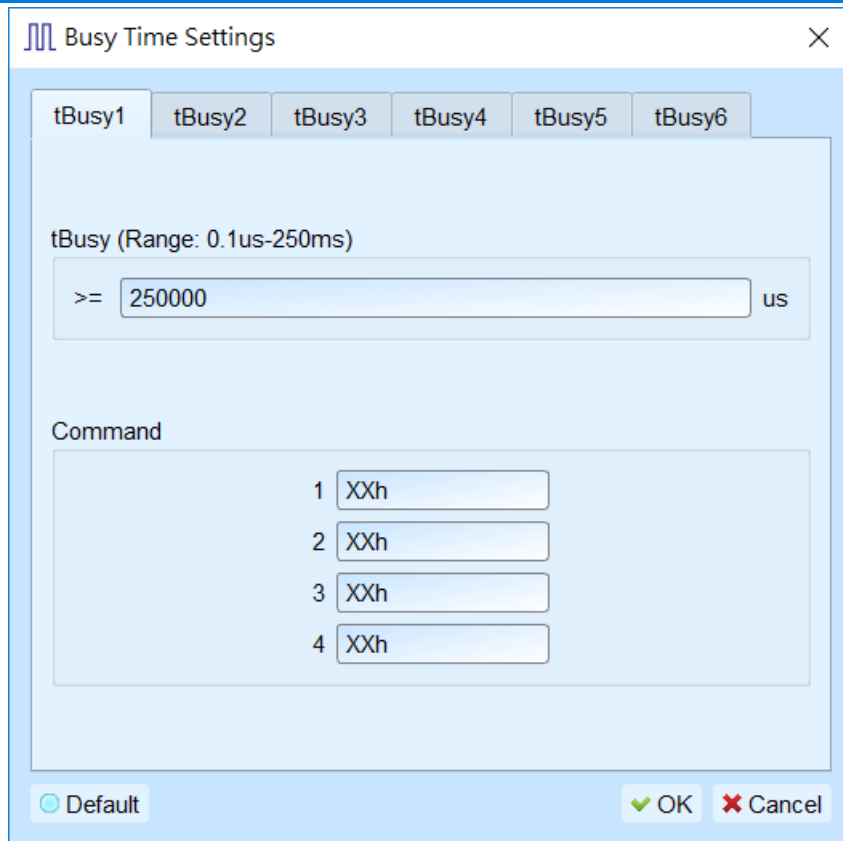
1 <input type="text" value="70h"/>	5 <input type="text" value="XXh"/>
2 <input type="text" value="FFh"/>	6 <input type="text" value="XXh"/>
3 <input type="text" value="78h"/>	7 <input type="text" value="XXh"/>
4 <input type="text" value="7Bh"/>	8 <input type="text" value="XXh"/>

Default

Trigger 70h during busy state:



9. **Busy time check:** Triggers when timing \geq tBusy, providing 6 tBusy to check.



Trigger the tBusy >= 25 us after NAND command (10h)



10. **Clause Trigger:** Please reference Clause Trigger chapter.
11. **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”.

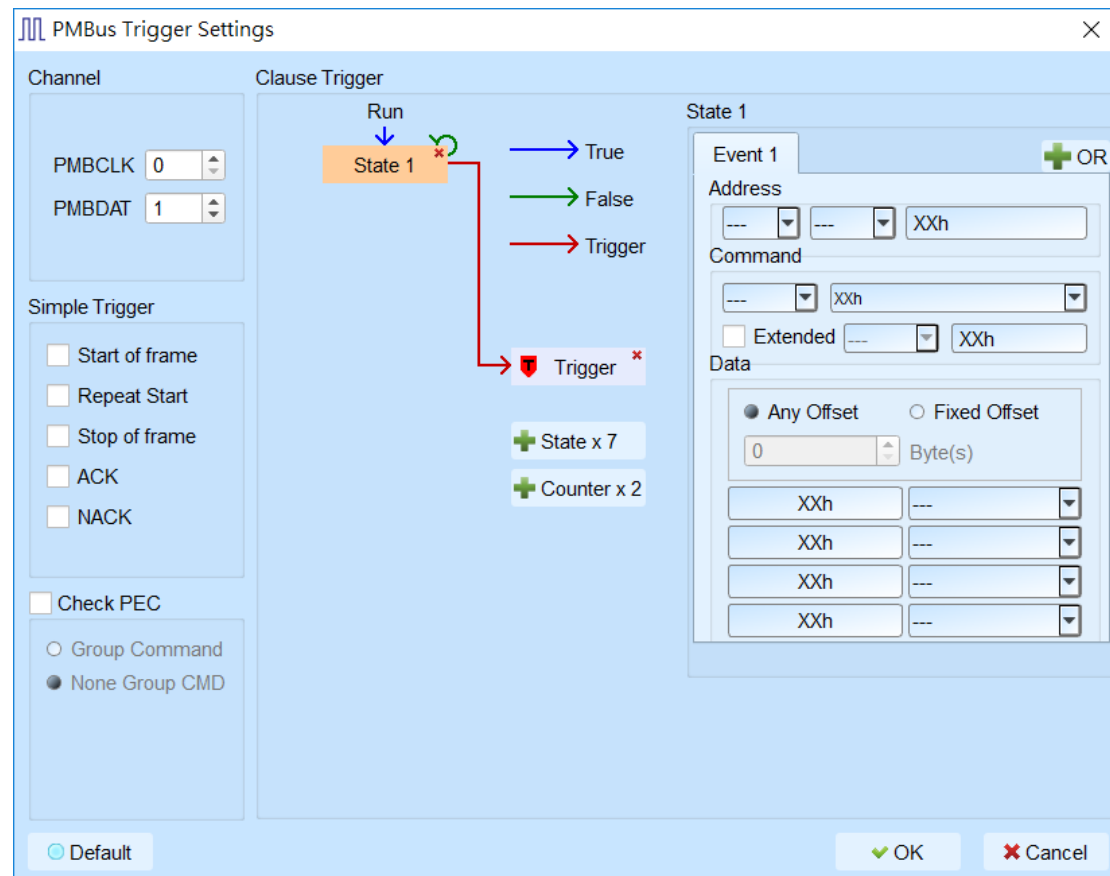
PMBus Trigger

Supported Models : TB1016E. TB1016B. TB1016B+. TL3134B. TL3234B+.

LA3068B. LA3136B

PMBus Trigger Settings

Click PMBus Trigger in the toolbar and will show the dialog as the following.



1. **Channel:** Select I2C / LA port.
2. **Simple Trigger:** Specific trigger function of I2C.
3. **Check PEC:** Trigger PEC.
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 Address, Command and Data fields, default value is XX means “don’t care”.

Data field only provided 4 bytes data to trigger, checking the Fixed offset and

selecting the offset value when want to trigger at the specific position.

Note: Only TravelBus series provide I2C port channel

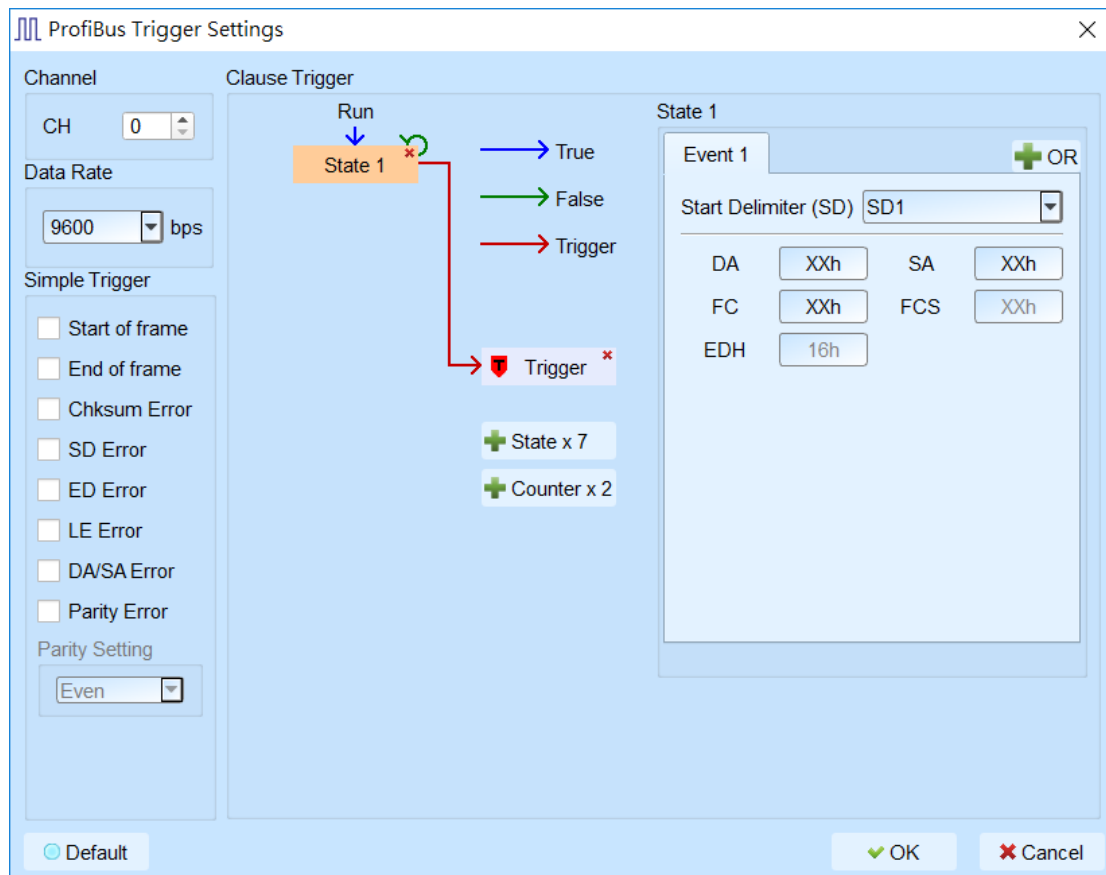
ProfiBus Trigger

Supported Models : TB1016E. TB1016B. TB1016B+. TL3134B. TL3234B+.

LA3068B. LA3136B

ProfiBus Trigger Settings

Click ProfiBus Trigger in the toolbar and will show the dialog as the following.



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

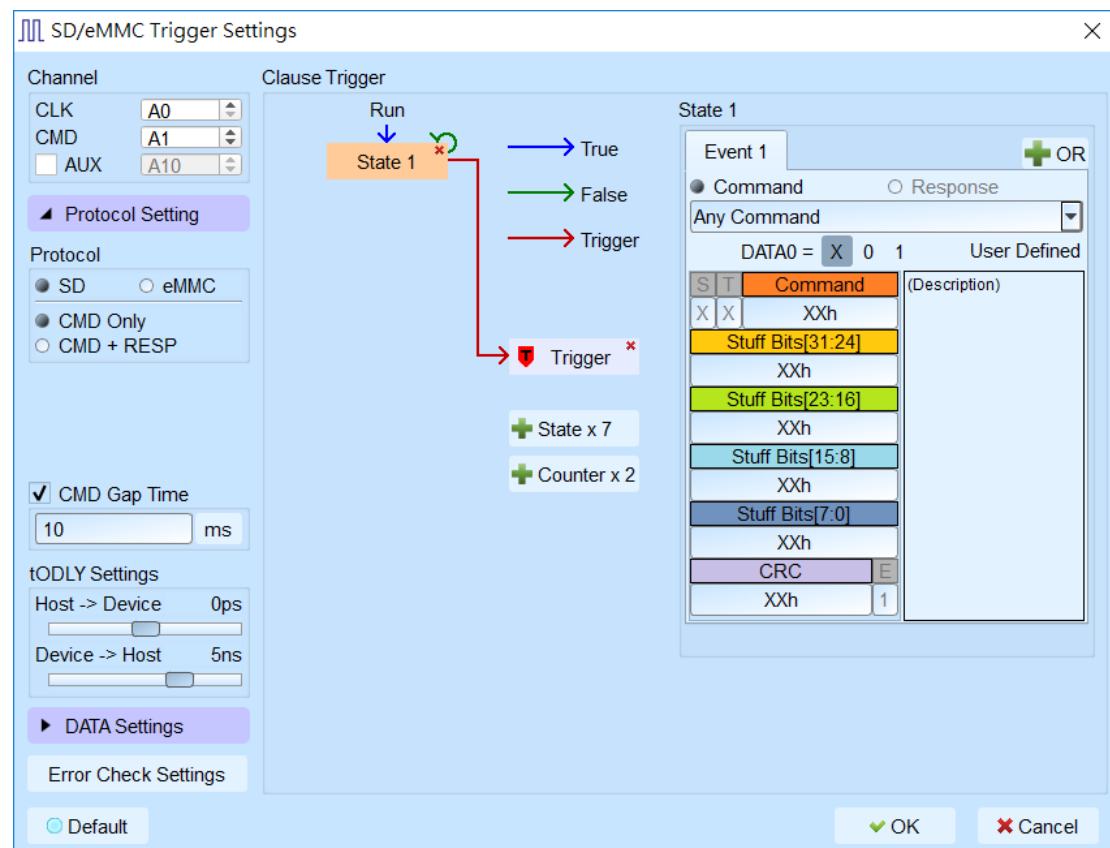
selecting the trigger values in the SD packet and its fields, default value is XX means “don’t care”.

SD/eMMC Trigger

Supported Models : TL3234B+. LA3068B. LA3136B. BusFinder (SD/eMMC solution)

SD/eMMC Trigger Settings

Click **SD/eMMC Trigger** in the toolbar and will show the dialog as the following ◦



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
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.

1. **Error Check Settings:** CRC7 error, CRC16 error, Timeout trigger are parallel structure with the Clause Trigger

I. CRC Error Trigger:

CRC Error Trigger Timeout Trigger

Trigger on CMD (CRC7) error

Trigger on DATA (CRC16) error

Read CMD list for CRC check

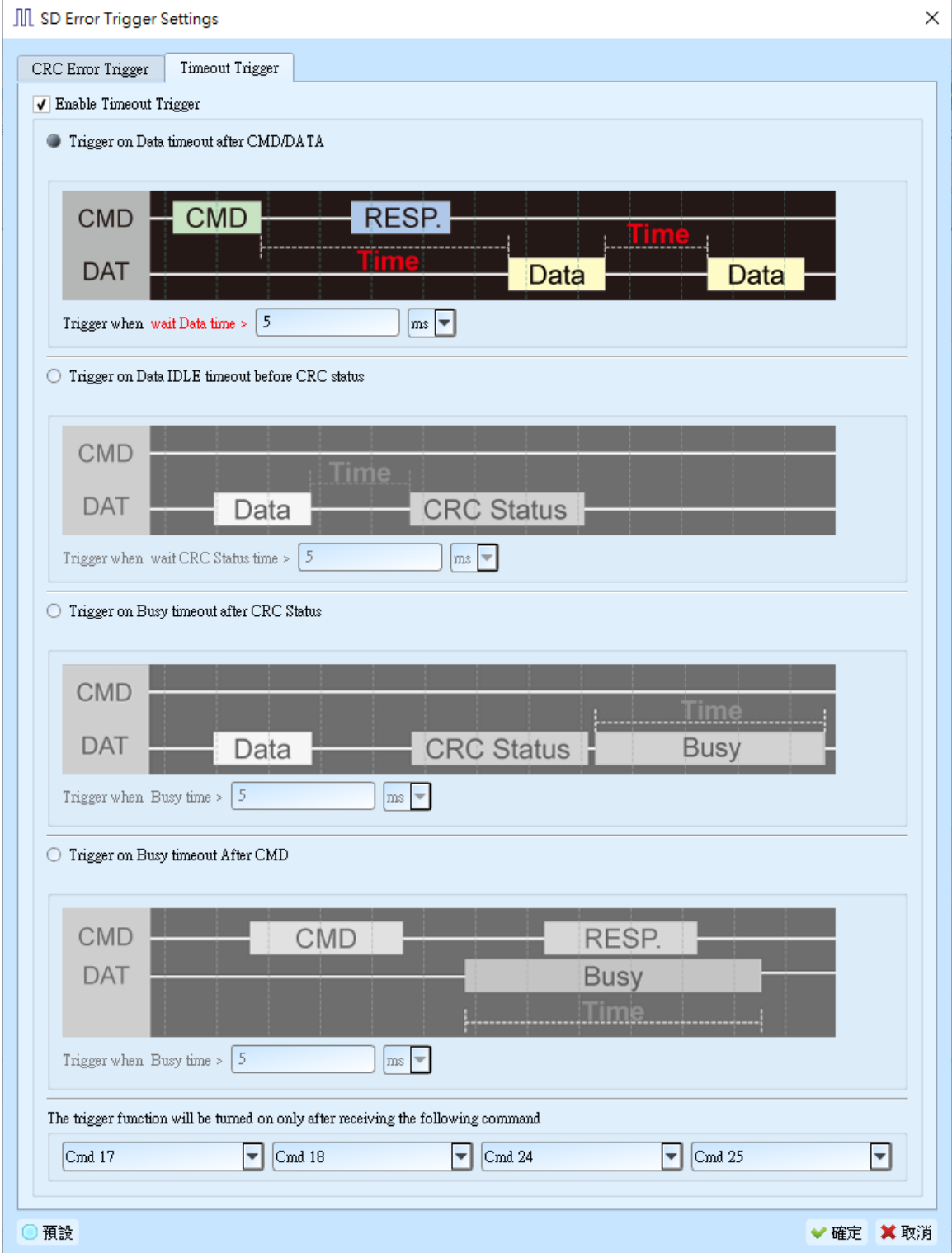
Cmd 17 Cmd 18 Cmd 46 Cmd 53

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

CMD CMD RESP.

DAT Time Data Data

Trigger when wait Data time > 5 ms

Trigger on Data IDLE timeout before CRC status

CMD Time

DAT Data CRC Status

Trigger when wait CRC Status time > 5 ms

Trigger on Busy timeout after CRC Status

CMD Time

DAT Data CRC Status Busy

Trigger when Busy time > 5 ms

Trigger on Busy timeout After CMD

CMD CMD RESP.

DAT Busy

Trigger when Busy time > 5 ms

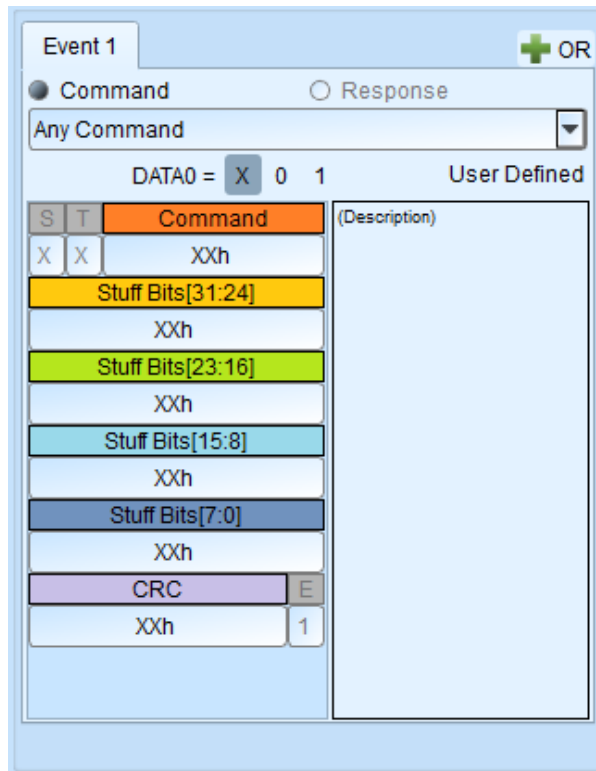
The trigger function will be turned on only after receiving the following command

Cmd 17 Cmd 18 Cmd 24 Cmd 25

預設 確定 取消

- ◆ **Trigger on Data timeout after CMD/DATA**
- ◆ **Trigger on Data IDLE timeout before CRC status**
- ◆ **Trigger on Busy timeout after CRC status**
- ◆ **Trigger on Busy timeout after CMD**

5. **Clause trigger settings:** Please reference Clause Trigger chapter
6. **Trigger settings**



Parameter :

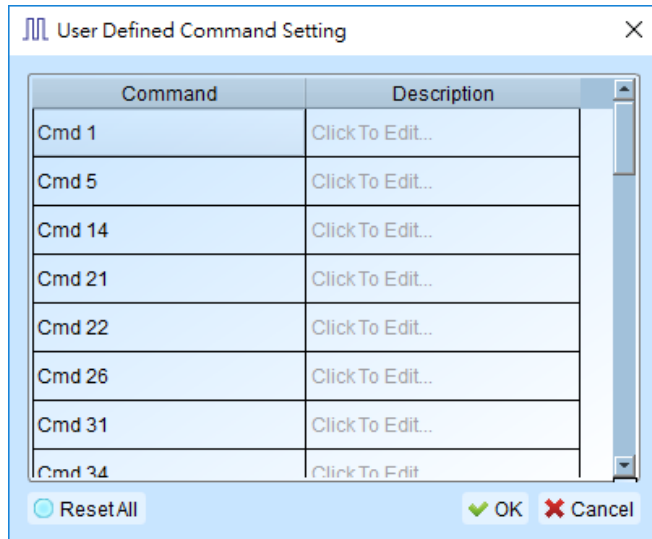
DATA0 = X 0 1 : use Data0 to judge the R1 and R1b.

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:

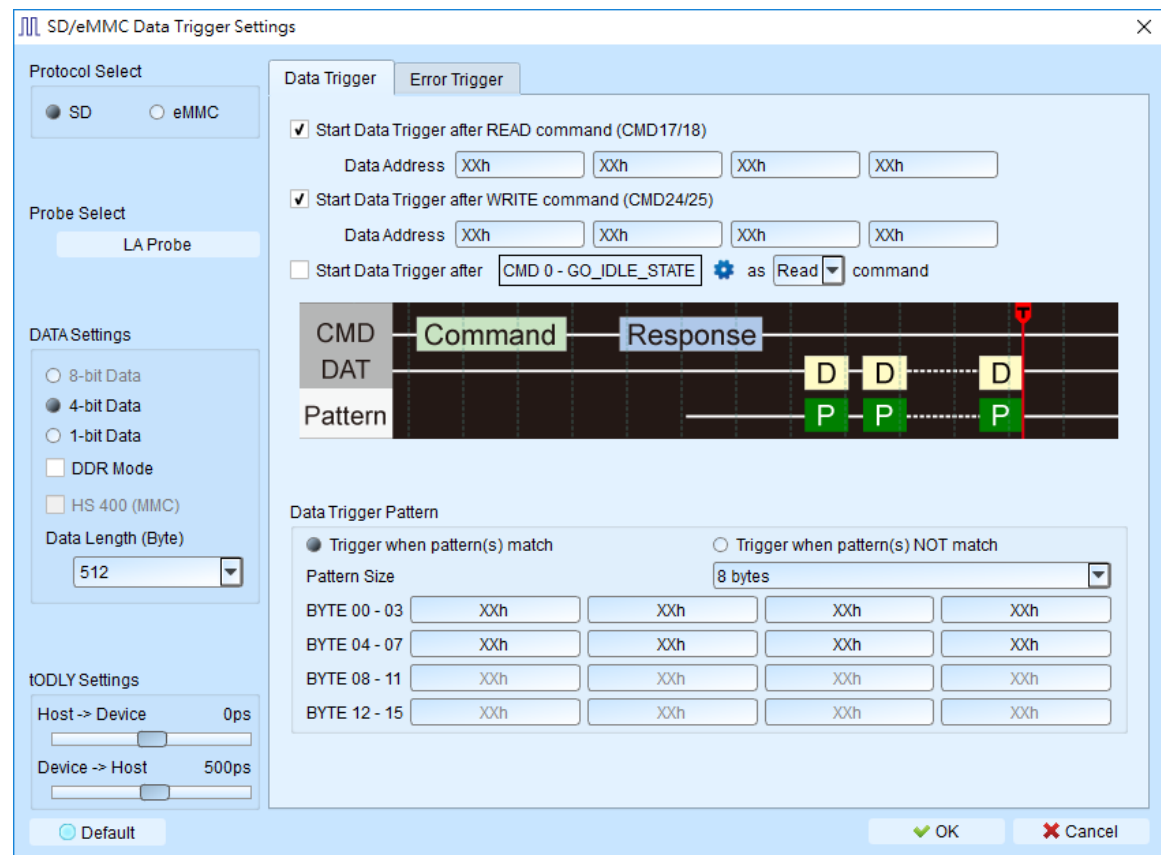


SD/eMMC Data Trigger

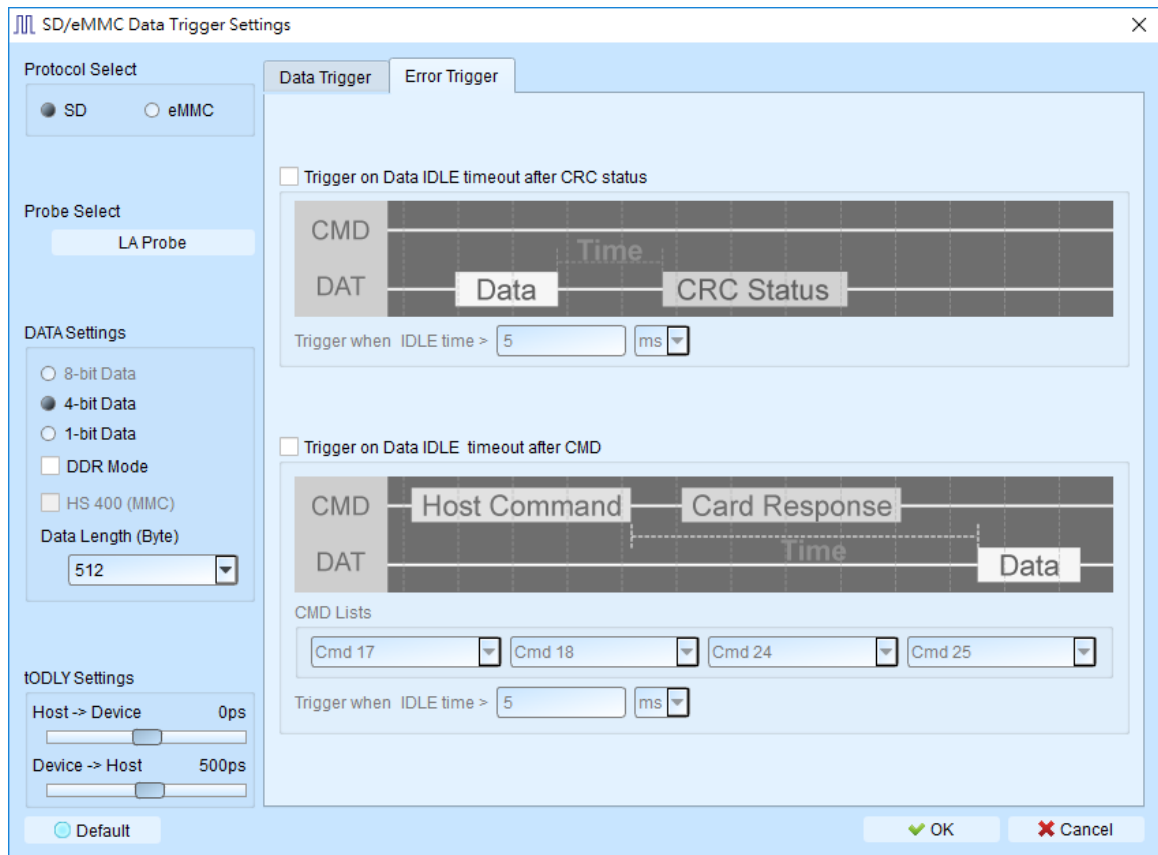
Supported Models : TL3234B+. LA3068B. LA3136B. BusFinder (SD/eMMC solution)

SD/eMMC Data Trigger Settings

Click **SD/eMMC Data Trigger** in the toolbar and will show the dialog as the following °



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.



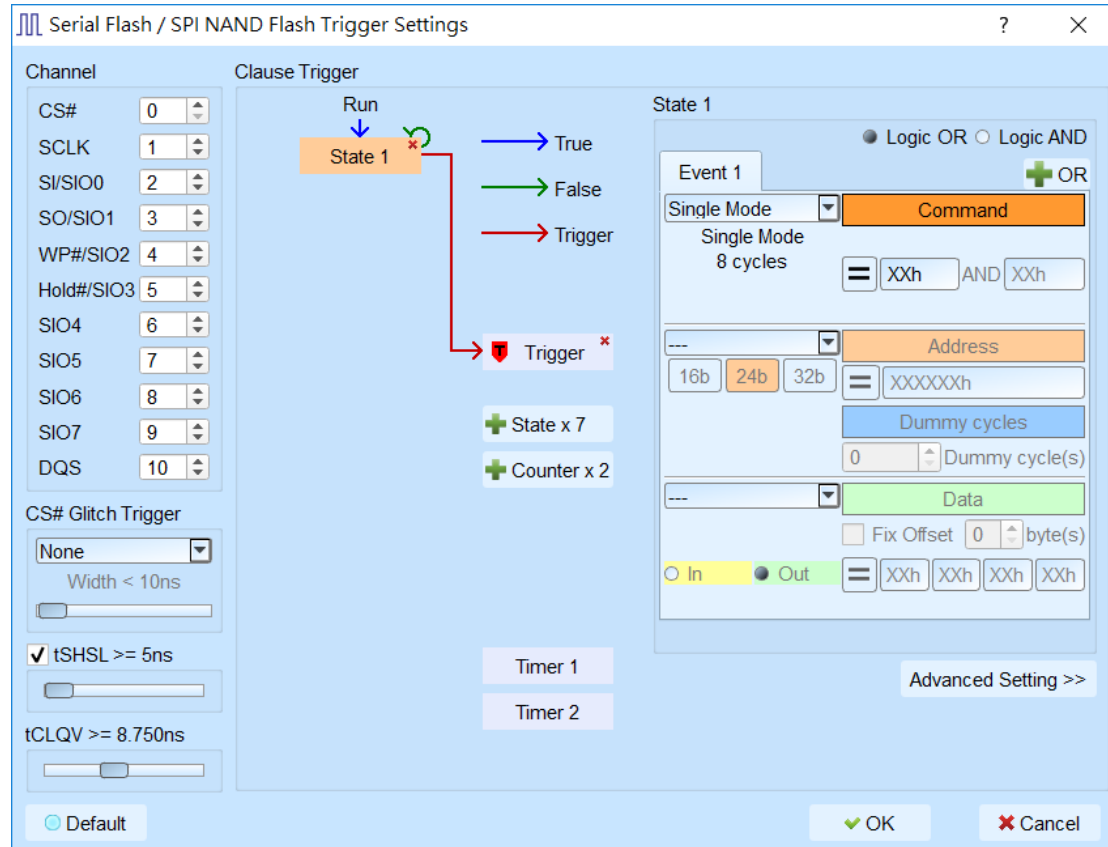
6. **Error Trigger:** set the condition of idle time to trigger.

Serial Flash / SPI NAND Trigger

Supported Models : TL3234B+. LA3136B. BusFinder series

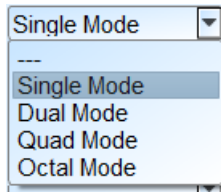
Serial Flash / SPI NAND Flash Trigger Settings

Click Serial Flash Trigger in the toolbar and will show the dialog as the following.



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.



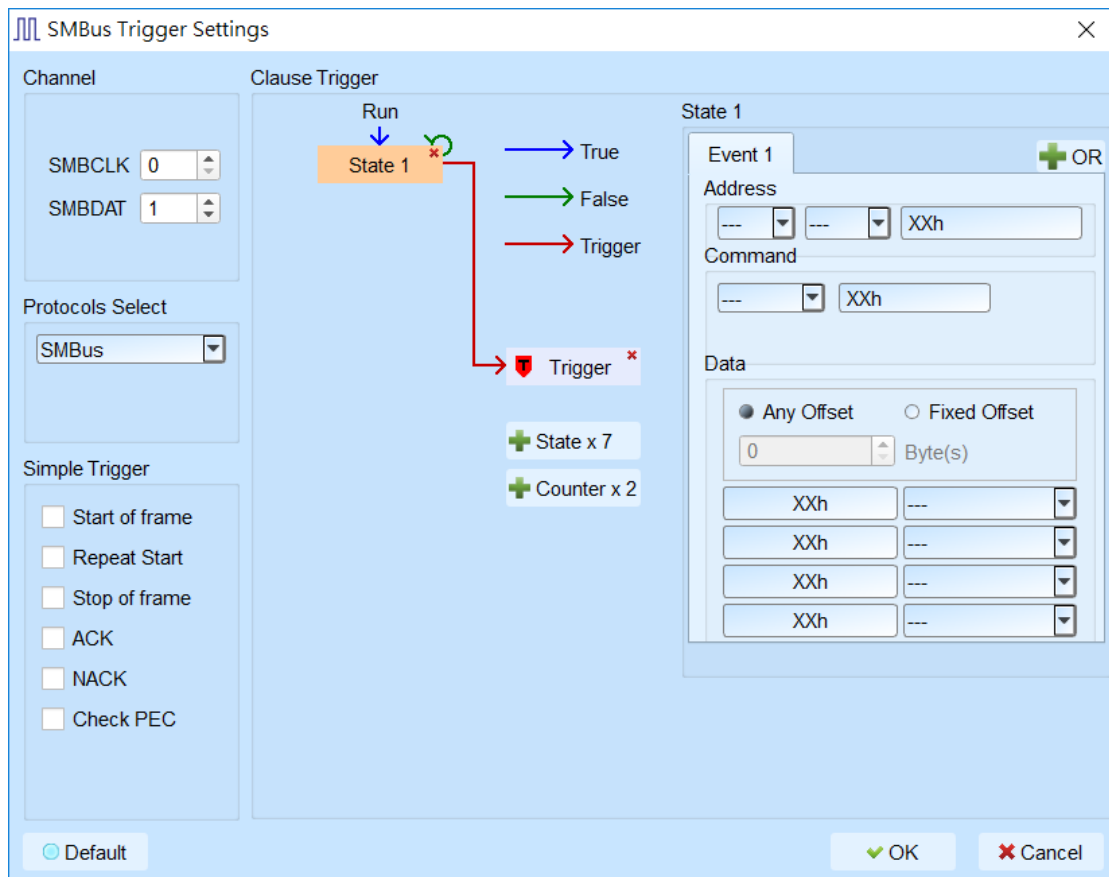
SMBus Trigger

Supported Models : TB1016E. TB1016B. TB1016B+. TL3134B. TL3234B+.

LA3068B. LA3136B

SMBus Trigger Settings

Click SMBus Trigger in the toolbar and will show the dialog as the following.



1. **Channel:** Select I2C / LA port.
2. **Protocols Select:** Select SMBus / SBS / SPD protocol.
3. **Simple Trigger:** Specific trigger function of SMBus.
4. **Clause Trigger:** Please reference Clause Trigger chapter.

- State:** Show the details of trigger condition in every state as left side; typing or selecting the trigger values in the Address, Command and Data fields, default value is XX means “don’t care”.

Data field only provided 4 bytes data to trigger, checking the Fixed offset and selecting the offset value when want to trigger at the specific position.

Note: Only TravelBus series provide I2C port channel

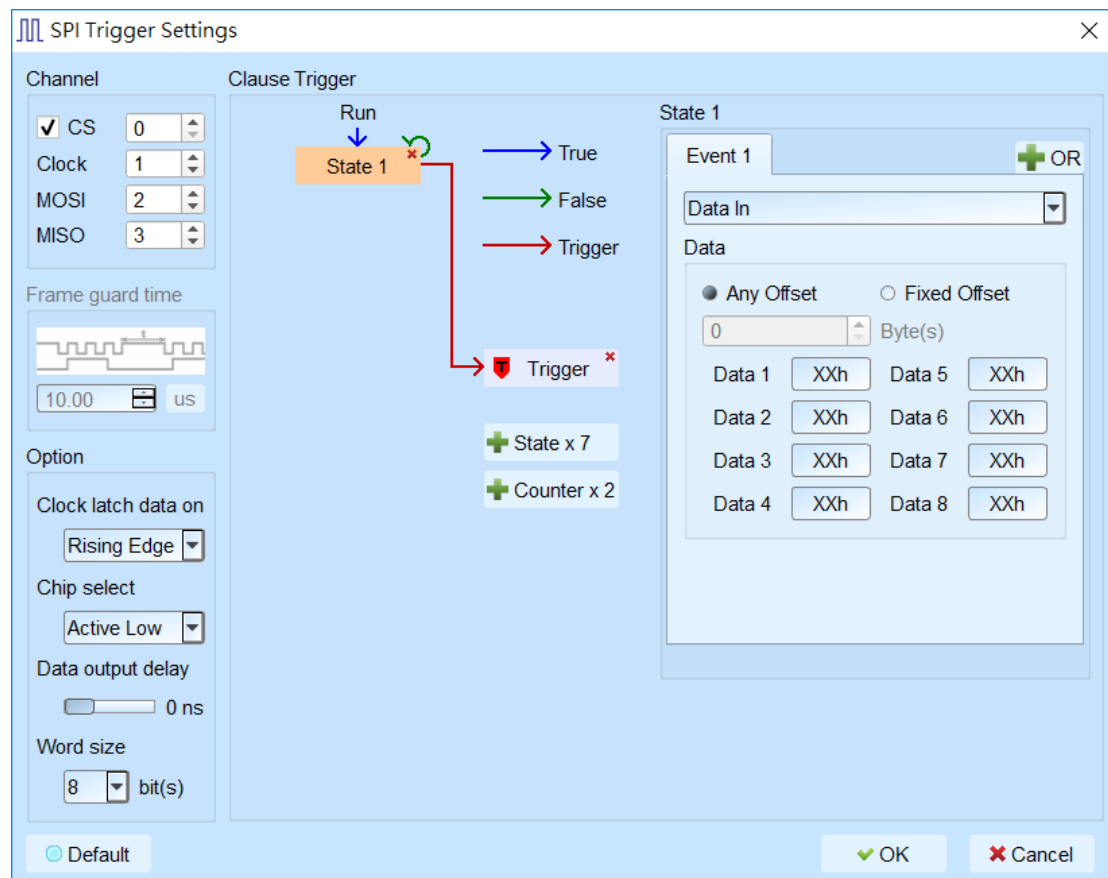
SPI Trigger

Supported Models : TravelBus series. TL3134E. TL3134B. TL3234B+.

Logic Analyzer series. BusFinder series

SPI Trigger Settings

Click SPI Trigger in the toolbar and will show the dialog as the following.



- Channel:** Select SPI channels.

2. **Frame guard time:** Only for unchecked CS state.
3. **Option:** Set SPI parameters.
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 Data fields, default value is XX means “don’t care”.

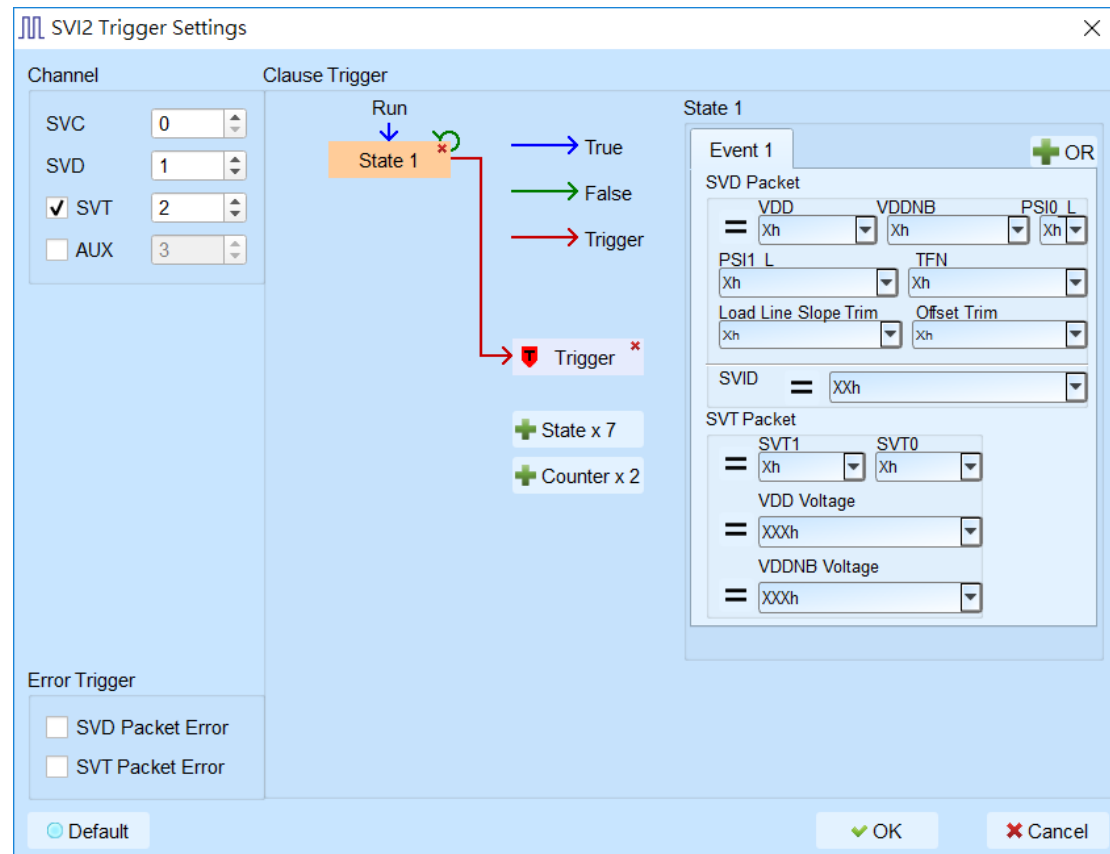
Data field provided 8 bytes data to trigger, checking the Fixed offset and selecting the offset value when want to trigger at the specific position.

SVI2 Trigger

Supported Models : TL3134B. TL3234B+

SVI2 Trigger Settings

Click SVI2 Trigger in the toolbar and will show the dialog as the following.



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).

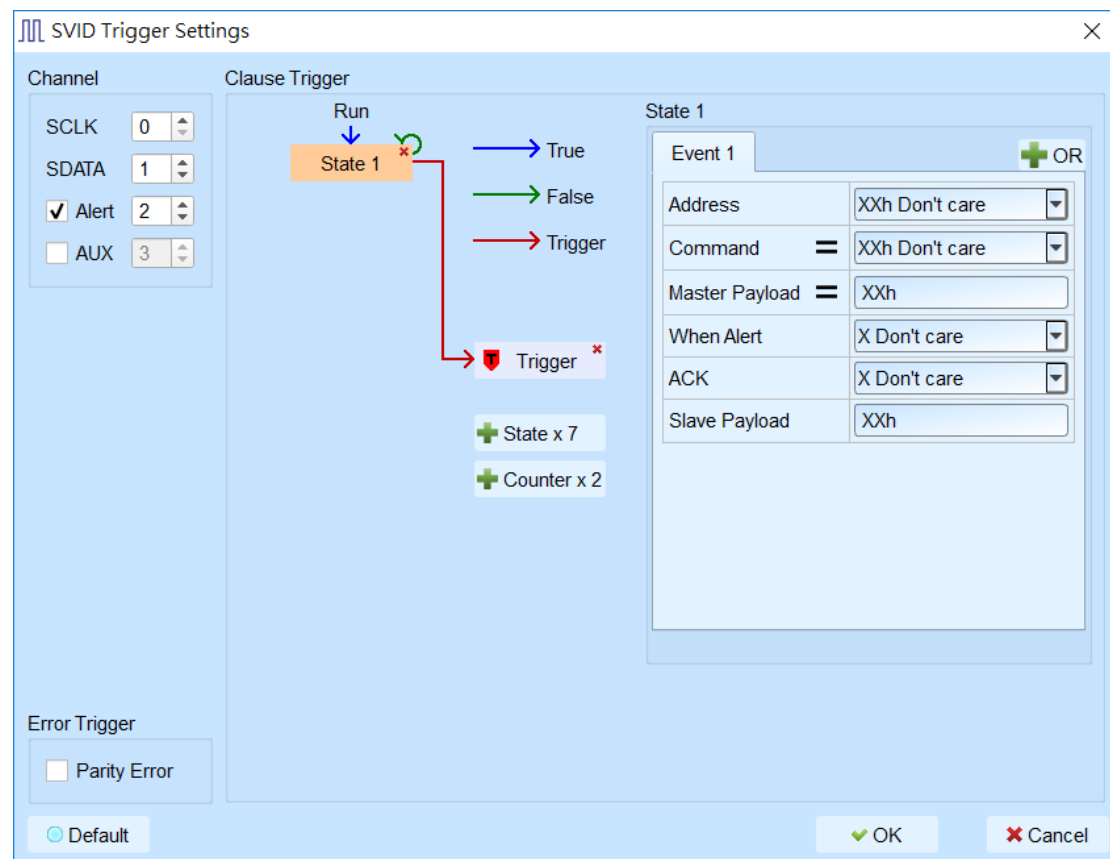
SVID Trigger

Supported Models : TL3134B. TL3234B+. LA3068B. LA3136B

If you have any issues with SVID protocol features, please contact your Intel Field Representative.

SVID Trigger Settings

Click SVID Trigger in the toolbar and will show the dialog as the following.



1. **Channel:** Select channels.
2. **Error Trigger:** Set SVID Parity Error trigger.
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 Address, Command, Master Payload, ACK and Slave Payload fields, default value is XX means “don’t care”.

When check the AUX, the trigger function will refer to the state of AUX
(High/Low).

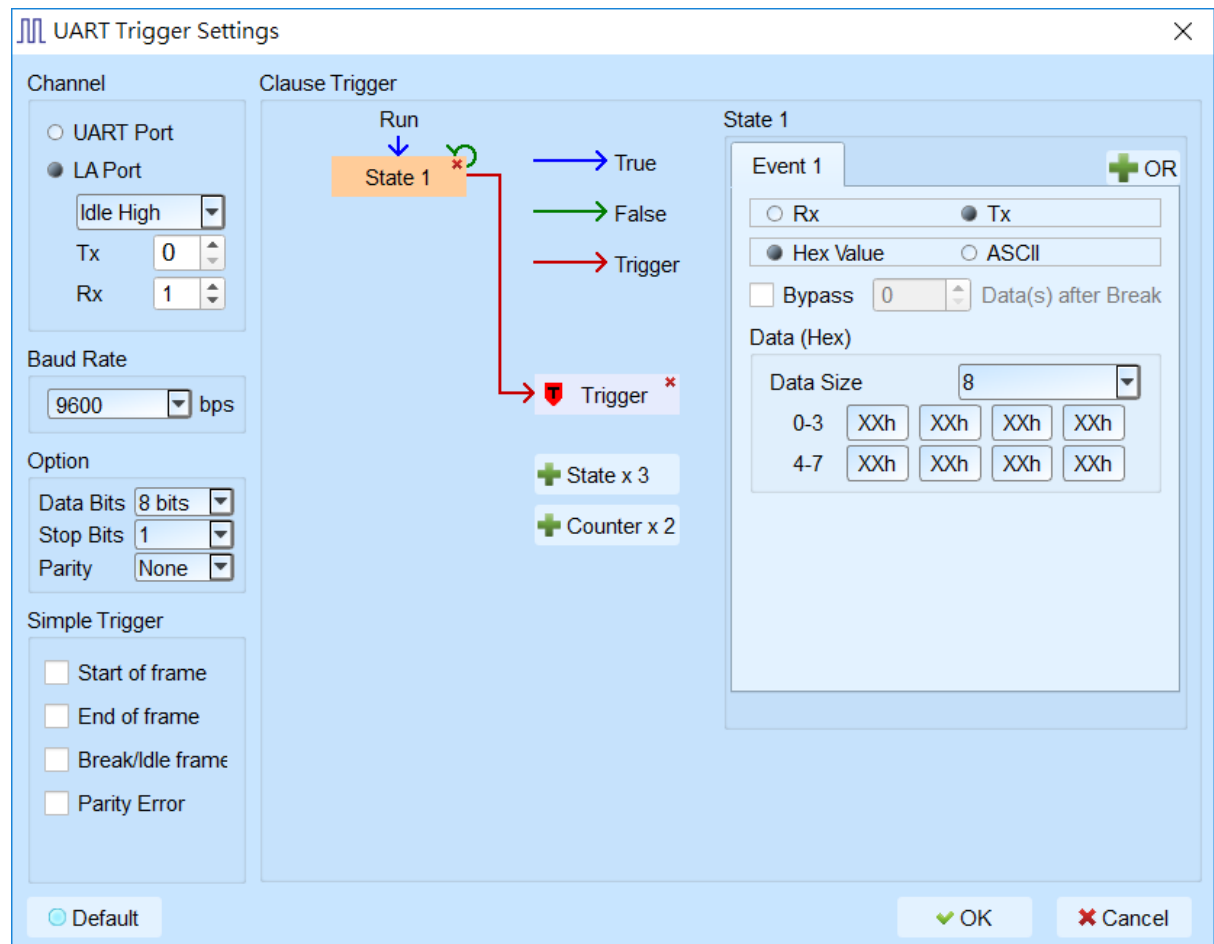
UART Trigger

Supported Models : TravelBus series. TL3134E. TL3134B. TL3234B+. Logic

Analyzer series

UART Trigger Settings

Click UART Trigger in the toolbar and will show the dialog as the following.



1. **Baud Rate:** Select UART baud rate.
2. **Simple Trigger:** Specific trigger function of UART.
3. **Clause Trigger:** Please reference Clause Trigger chapter.
4. **State:** Show the details of trigger condition in every state as left side; typing the trigger values in the Data fields, default value is XX means “don’t care”.

Data field provided 16 bytes data to trigger, checking the Bypass and selecting the offset value when want to trigger at the specific position.

Note: Only TB1016B. TB1016B+ provide UART port channel

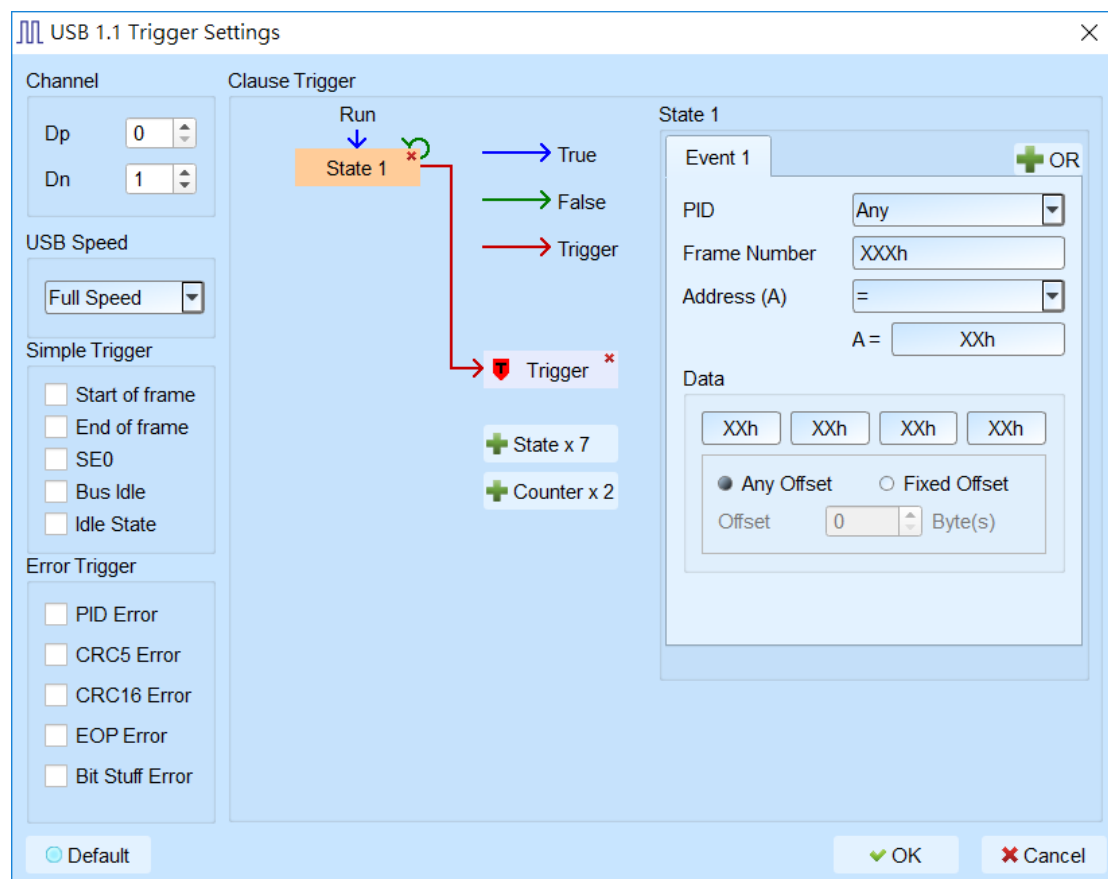
USB 1.1 Trigger

Supported Models : TB1016E. TB1016B. TB1016B+. TL3134B. TL3234B+.

LA3068B. LA3136B

USB 1.1 Trigger Settings

Click USB 1.1 Trigger in the toolbar and will show the dialog as the following.



- 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.

- 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.

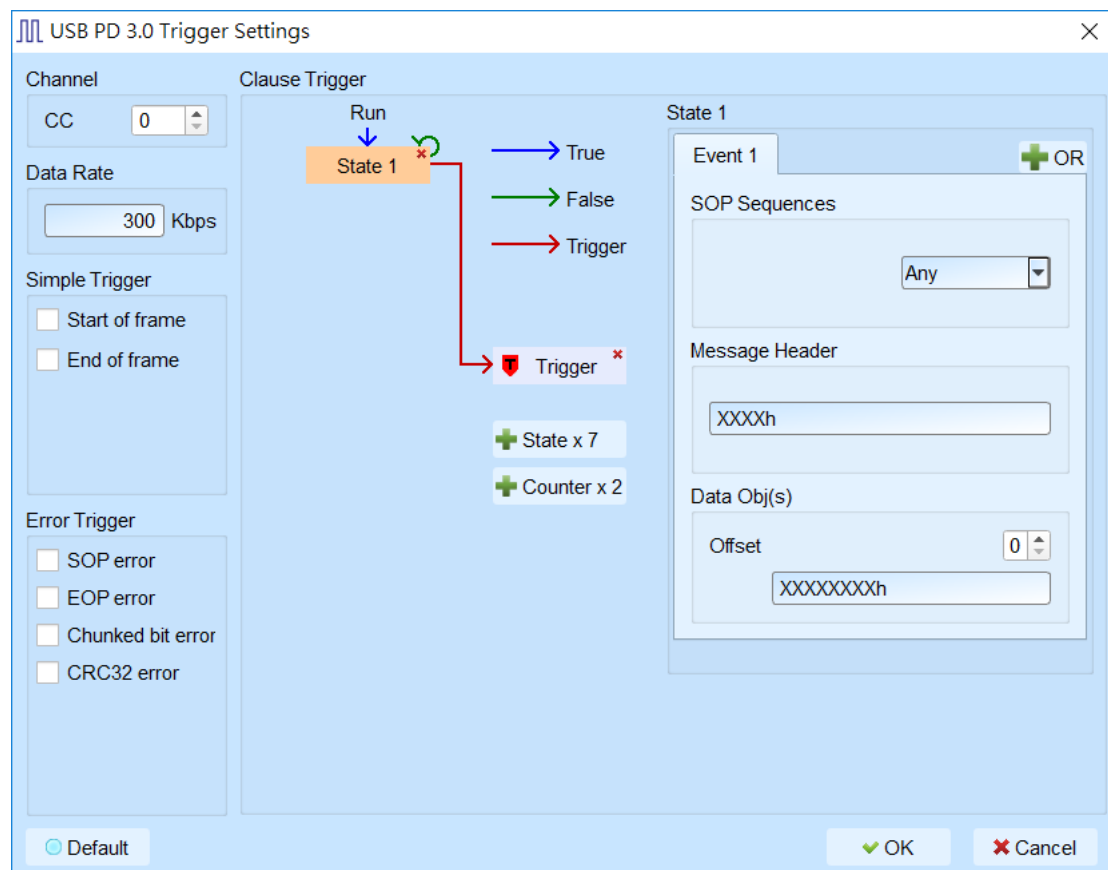
USB PD 3.0 Trigger

Supported Models : TB1016E. TB1016B. TB1016B+. TL3134E. TL3134B.

TL3234B+. Logic Analyzer Series

USB PD 3.0 Trigger Settings

Click USB PD 3.0 Trigger in the toolbar and will show the dialog as the following.



- Channel:** Select channels.
- Simple Trigger:** Specific trigger function of USB PD 3.0.
- 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.