

## Table of Contents

<b>Chapter 1</b>	<b>Bus Decode</b>	<b>1</b>
1-Wire		9
3-Wire		11
7-Segment		13
A/D Converter		16
Accelerometer		20
AD-Mux Flash		22
Advanced Platform Management Link (APML)		24
BiSS-C		26
BSD		28
CAN 2.0B/ CAN FD		29
Closed Caption		32
DALI		34
DMX512		36
Display Port Auxiliary Channel (DP Aux Ch)		38
Enhanced Serial Peripheral Interface (eSPI)		40
FlexRay		42
HD Audio		46
HDLC		48
HDQ		49
HID Over I <sup>2</sup> C		51
I <sup>2</sup> C		53
I3C		55
I <sup>2</sup> C EEPROM		58
I <sup>2</sup> S		60

I80	62
IDE	64
Indicator	68
IrDA	69
ITU656 (CCIR656)	71
JTAG	73
LCD1602	78
LIN	80
Line Decoding	82
Line Encoding	86
Lissajous	92
Low Pin Count (LPC)	96
Line Printer Terminal Port (LPT)	98
M-Bus	100
Math	102
Mobile Display Digital Interface (MDDI)	104
MDIO	106
MHL-CBUS	108
MII/RMII	110
Microwire	113
MIPI DSI	115
MIPI RFFE	117
MIPI SPMI	118
MMC	120
ModBus	123
NAND Flash	125

NEC IR -----	133
PECI-----	135
PMBus -----	137
ProfiBus -----	139
PS/2-----	141
PWM -----	143
QI -----	147
RC-5-----	149
RC-6-----	152
RGB Interface -----	154
S/PDIF -----	156
SDIO -----	159
Serial Flash -----	162
Serial IRQ-----	168
Serial General Purpose Input Output (SGPIO) -----	172
Smart Card (ISO7816) -----	174
System Management Bus (SMBus) -----	176
Serial Microprocessor Interface (SMI) -----	179
Serial Peripheral Interface (SPI) -----	181
Serial Peripheral Interface NAND (SPI NAND) -----	187
SSI -----	189
ST7669-----	193
Serial VID Interface 2.0 (SVI2)-----	195
Serial VID (SVID) (Upon Request)-----	197
Serial Wire Debug (SWD) -----	199
SWP -----	203

Universal Asynchronous Receiver/Transmitter (UART)-----	205
UNI/O-----	208
USB1.1 -----	210
USB PD 2.0-----	212
Wiegand -----	214
<b>Chapter 2    Bus Trigger-----</b>	<b>216</b>
Bus Trigger -----	217
CAN Trigger -----	218
I <sup>2</sup> C Trigger-----	221
I <sup>2</sup> S Trigger -----	230
SPI Trigger -----	235
SVID Trigger (Upon Request) -----	238
UART Trigger-----	242
Clause Trigger-----	244
eSPI Trigger -----	248
LIN Trigger-----	251
LPC Trigger -----	253
MIPI SPMI Trigger -----	256
NAND Flash Trigger-----	258
SD/eMMC Trigger -----	265
Serial Flash Trigger -----	268
SMBus/PMBus Trigger-----	271
SVI2 Trigger -----	273
USB1.1 Trigger -----	276

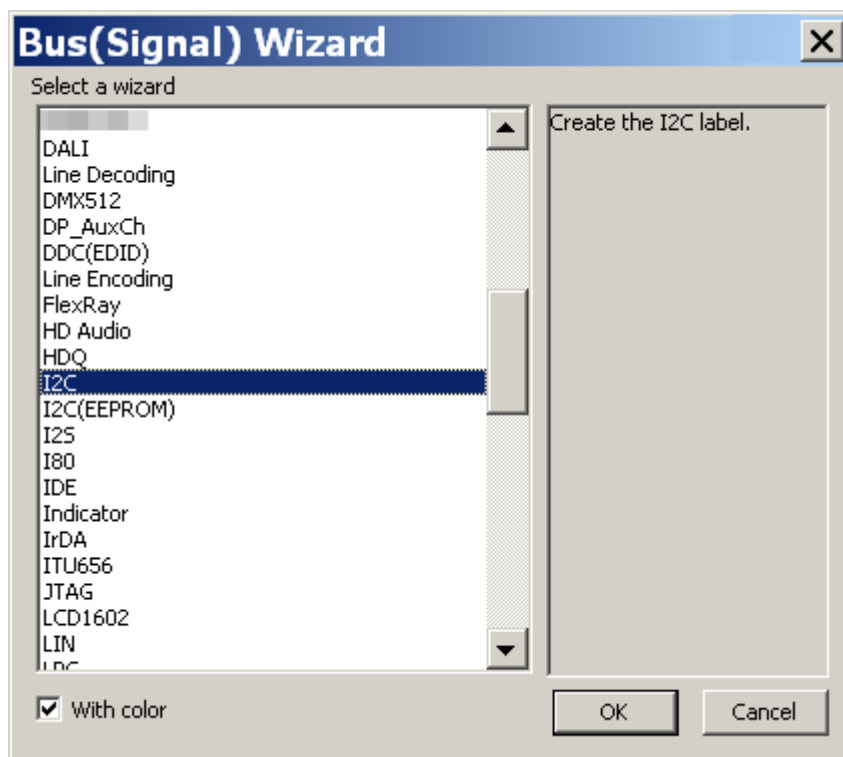
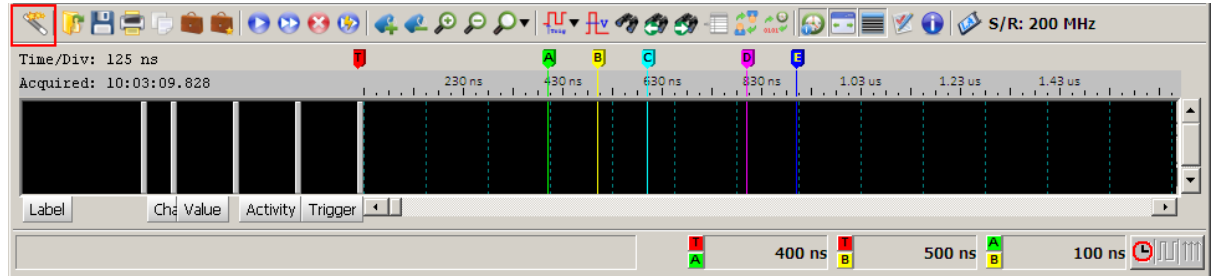


# **Chapter 1   Bus Decode**

## Add a Bus Decode:

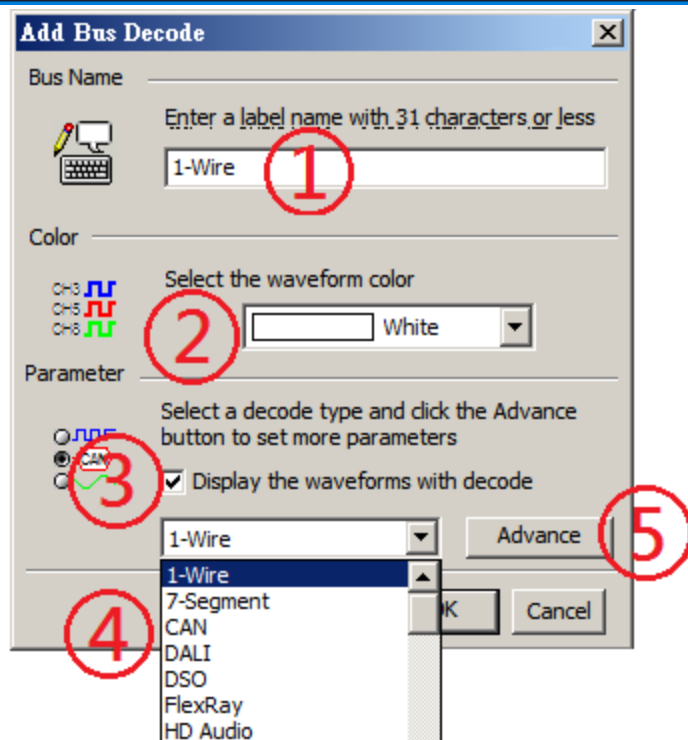
### Method 1:

Click the wizard in the menu, and select the bus decode.

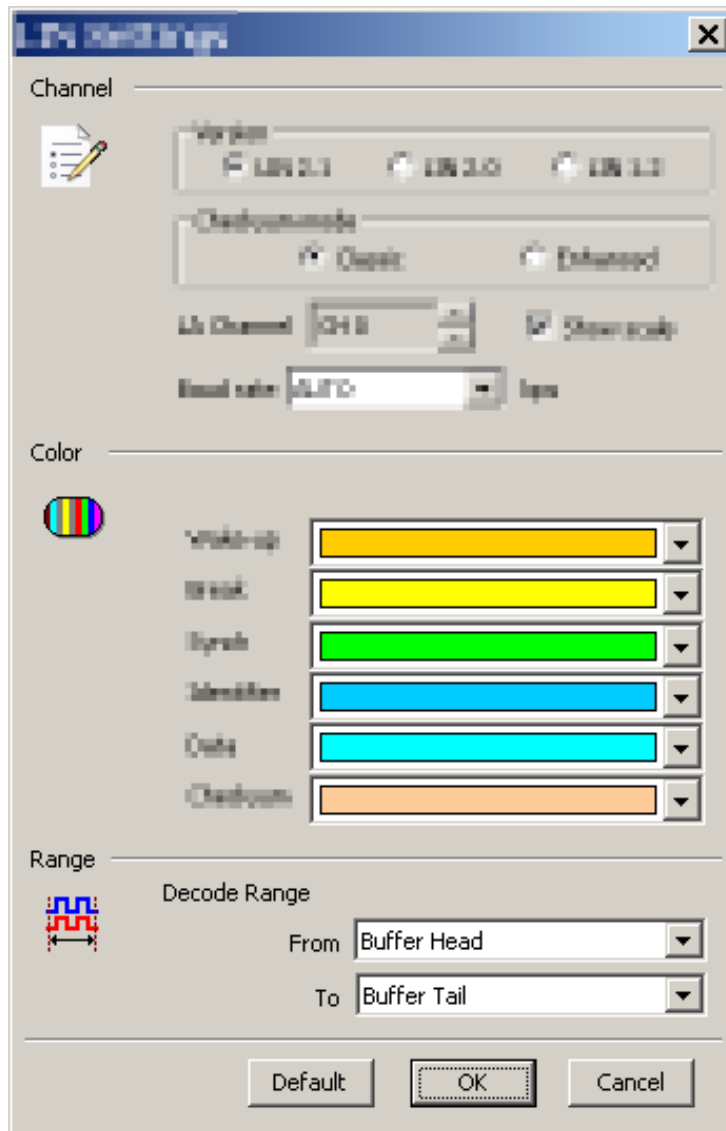


### Method 2:

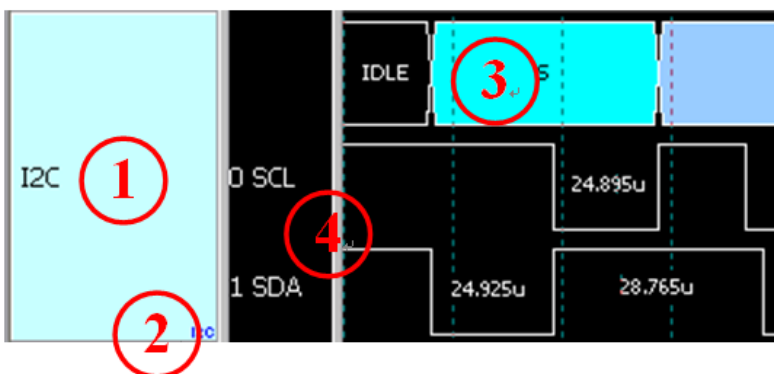
Click Add Bus Decode in the Label menu or right-click the label field to show the dialog box.



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



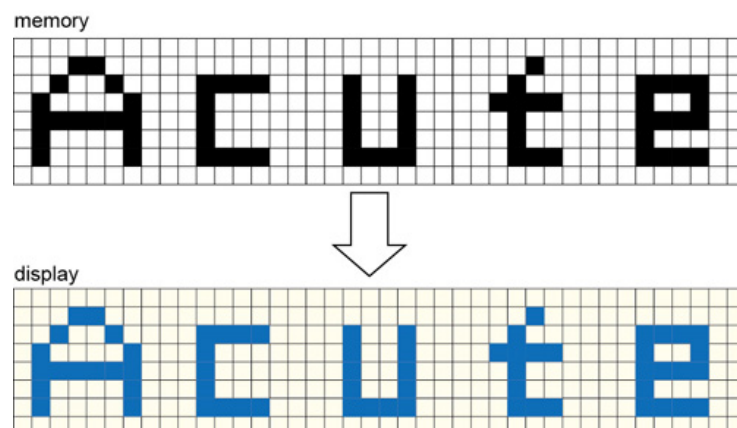
1. **Bus name**
2. **Decode type**
3. **Result**
4. **Channel name & signal**

### **Specially Bus Decode:**

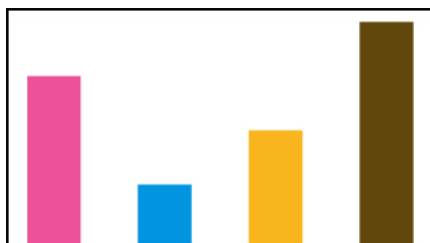
The bus decodes display the data in text format, but some decodes are able to show the original form for the data such as voice (I2S decode...), image, analog waveform (ADC decode) etc.

**In the future, we will extend the original form display to more applications like:**

### **LCD/CMOS image sensor bus decodes:**



### **Statistic or Bar chart:**



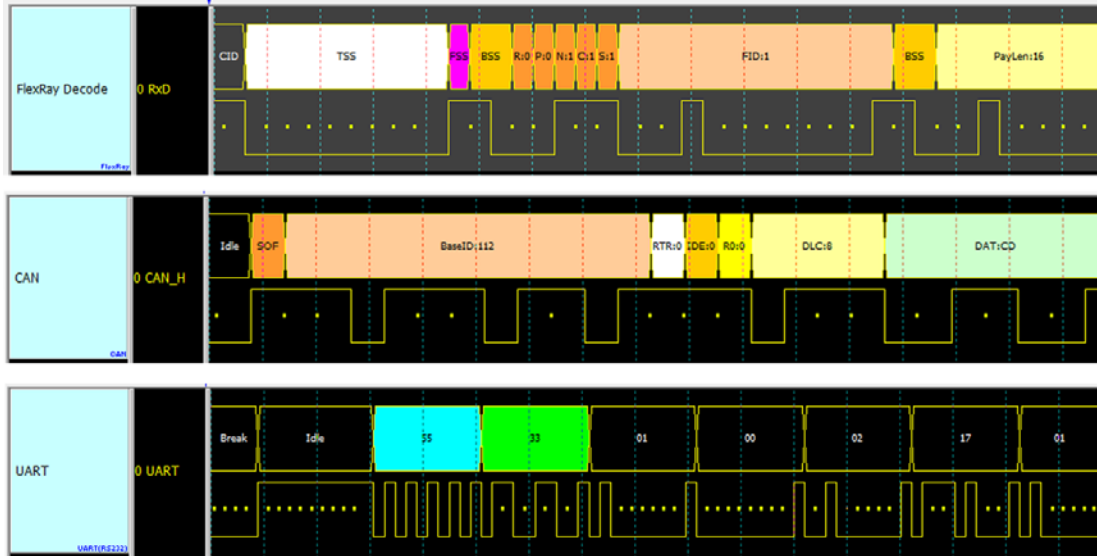
### **RPM:**



**The following original data form display for bus decodes are already available in the past.**

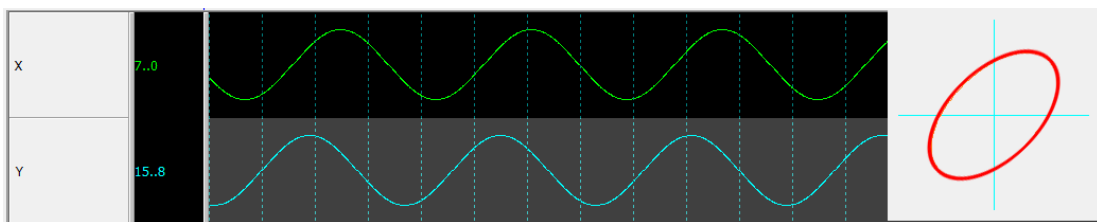
## UART/CAN/FlexRay..bus decodes (released in 2009/9, LA Viewer Ver. 2.0):

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



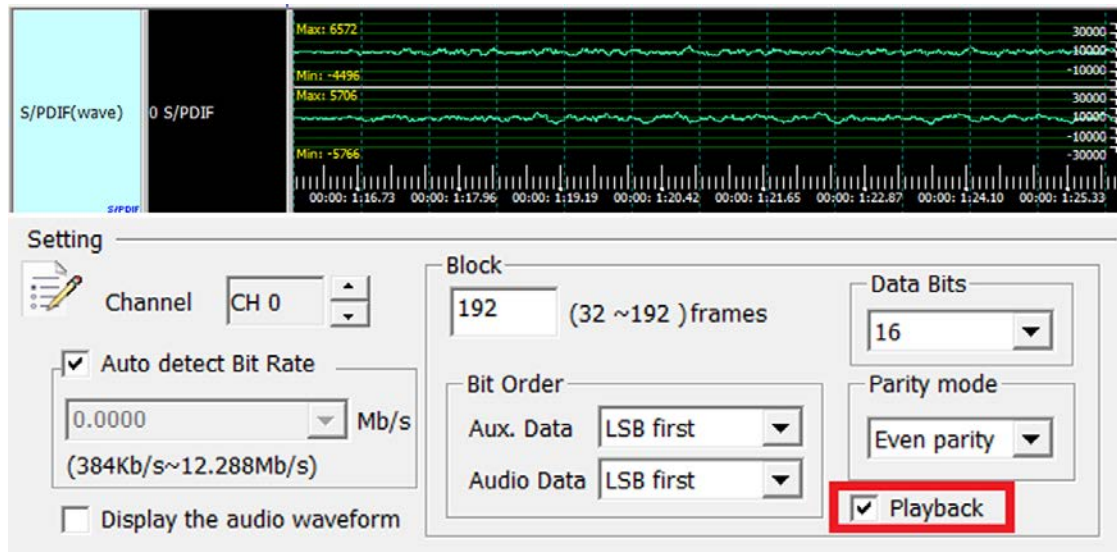
## Lissajous analysis (released in 2009/09, LA Viewer Ver.2.0):

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



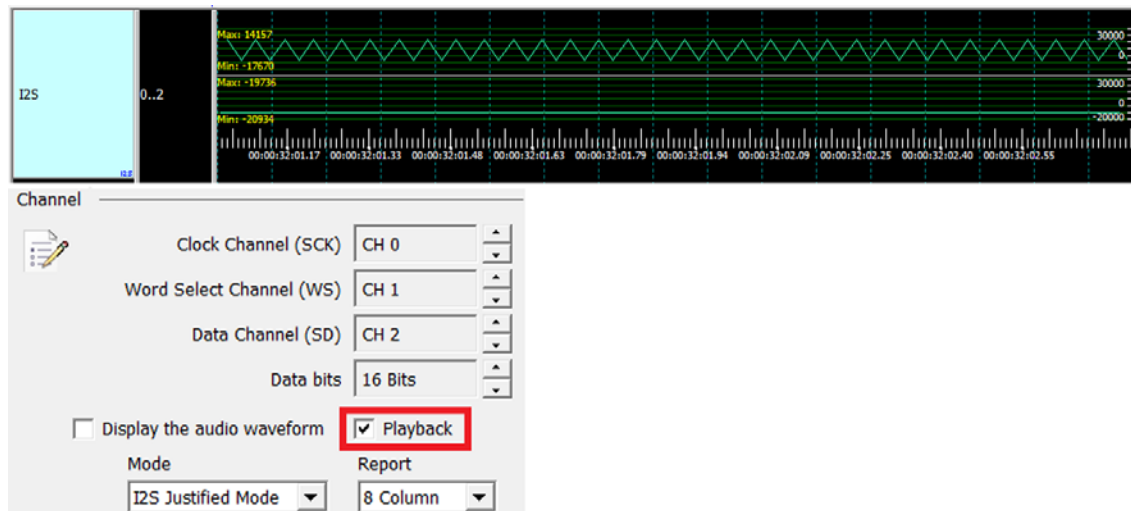
## S/PDIF analysis (2010/11, LA Viewer Ver. 2.5):

Display the data in sound waveform ◦



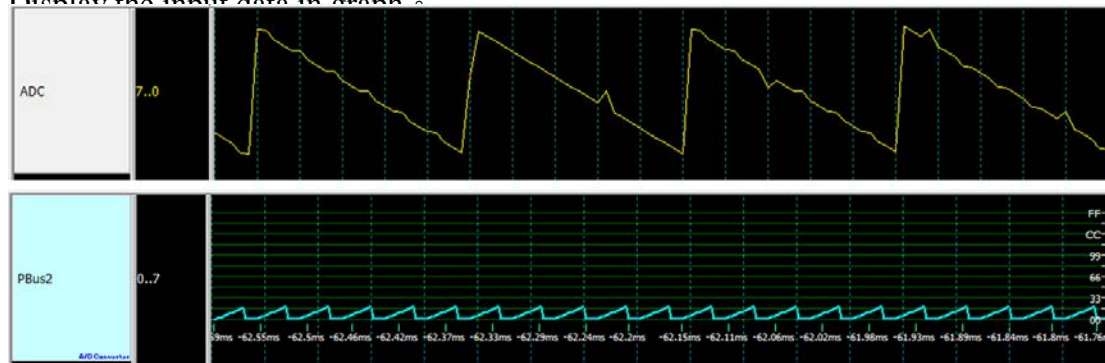
## I2S analysis (2011/09, Ver. 2.6.3): 分析 (2011/09, LA Viewer Ver 2.6.3)

Display the data in sound waveform ◦



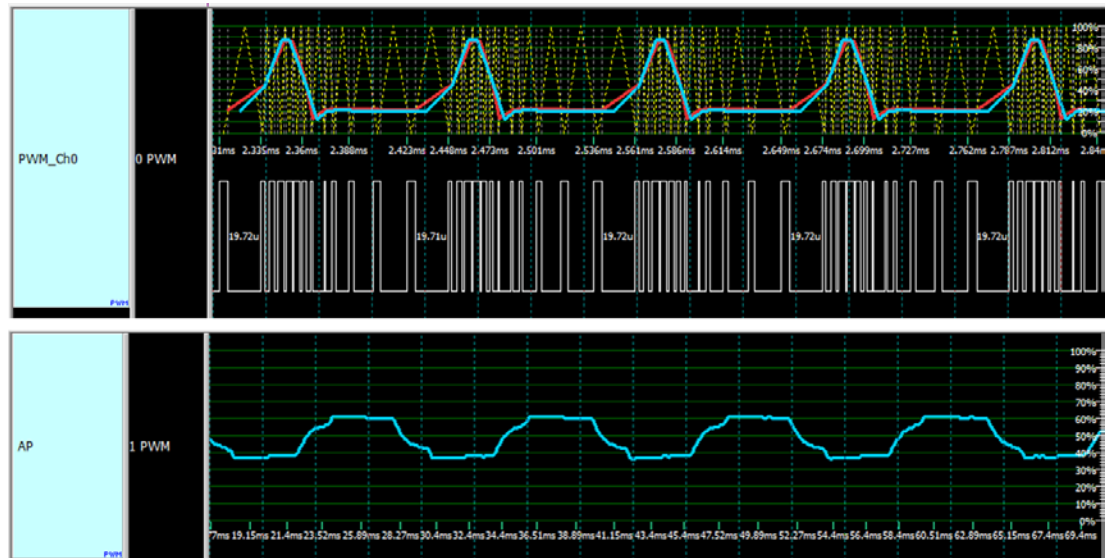
### ADC bus decode (2012/08, LA Viewer Ver. 2.7.3):

Display the input data in graph ◦



### PWM analysis (2012/08, LA Viewer Ver. 2.7.3):

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

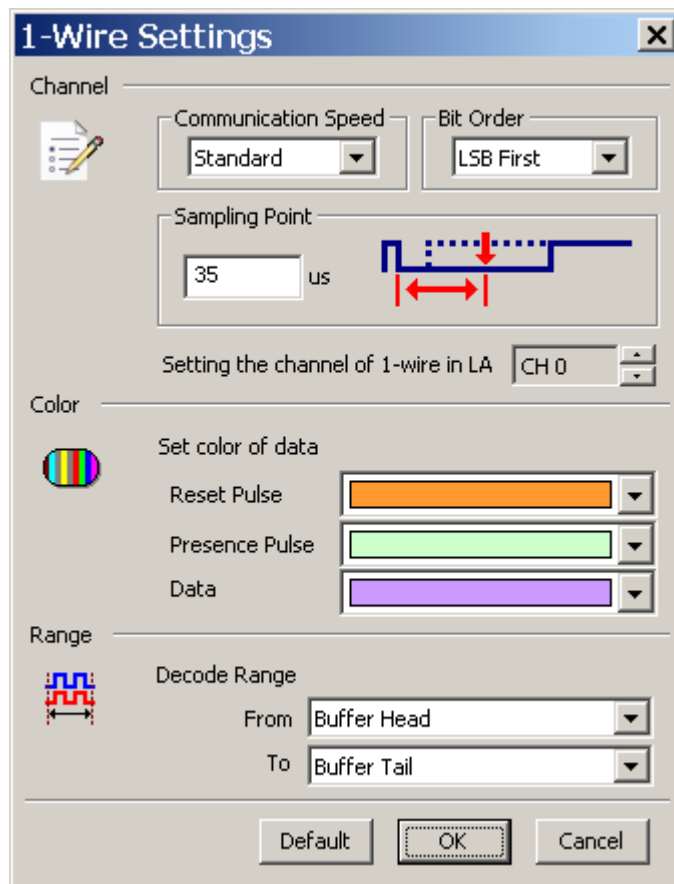




## 1-Wire

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.

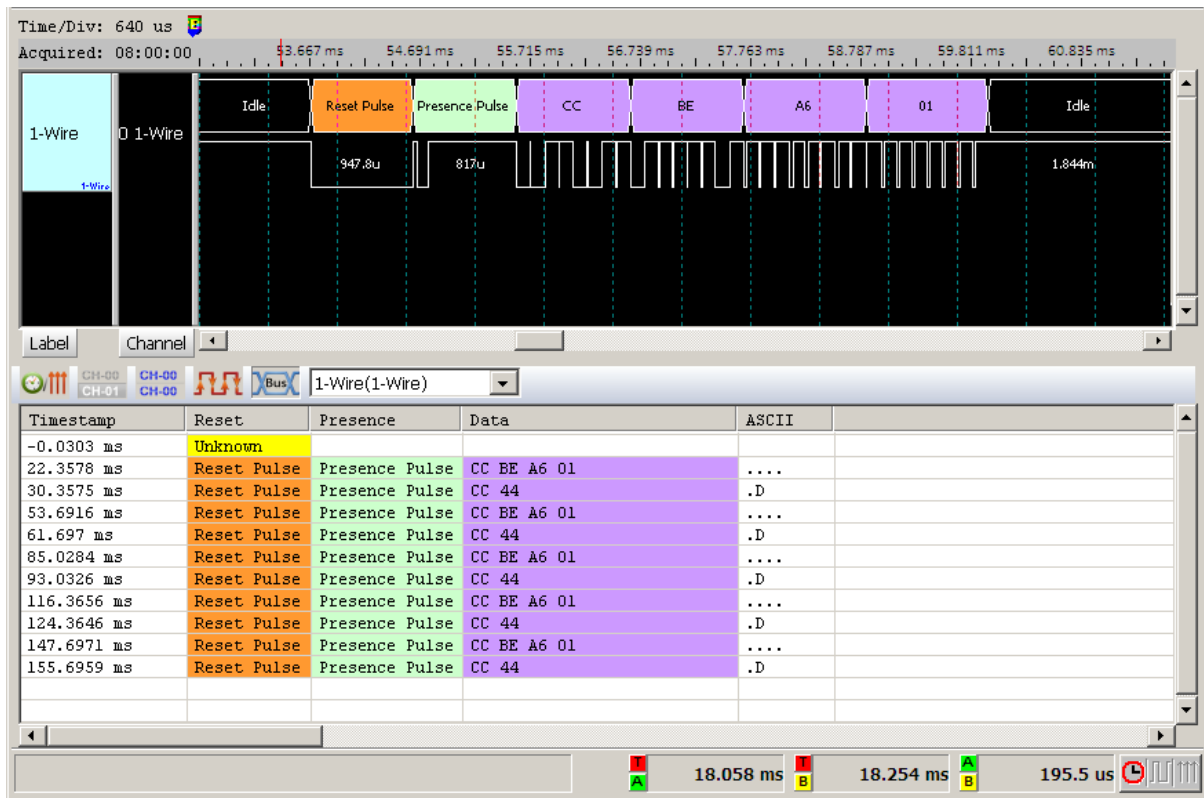
**Bit Order:** LSB first or MSB first.

**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

**3-Wire (HOLTEK) Settings**

**Parameters**

Channel

CS CH 0

WR CH 1

☒ RD CH 3

DATA CH 2

**Color**

OPERATION

ADDRESS

COMMAND

DATA

START

**Application**

☒ LED Driver IC

☐ LCD Driver IC

☐ EEPROM

HT1620x

HT93LC46

x8

**Range**

Decode Range

From Buffer Head

To Buffer Tail

**Latch**

Chip Select Edge

☐ Active High ☒ Active Low

Data Edge

☒ Rising ☐ Falling

Default OK Cancel

**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.

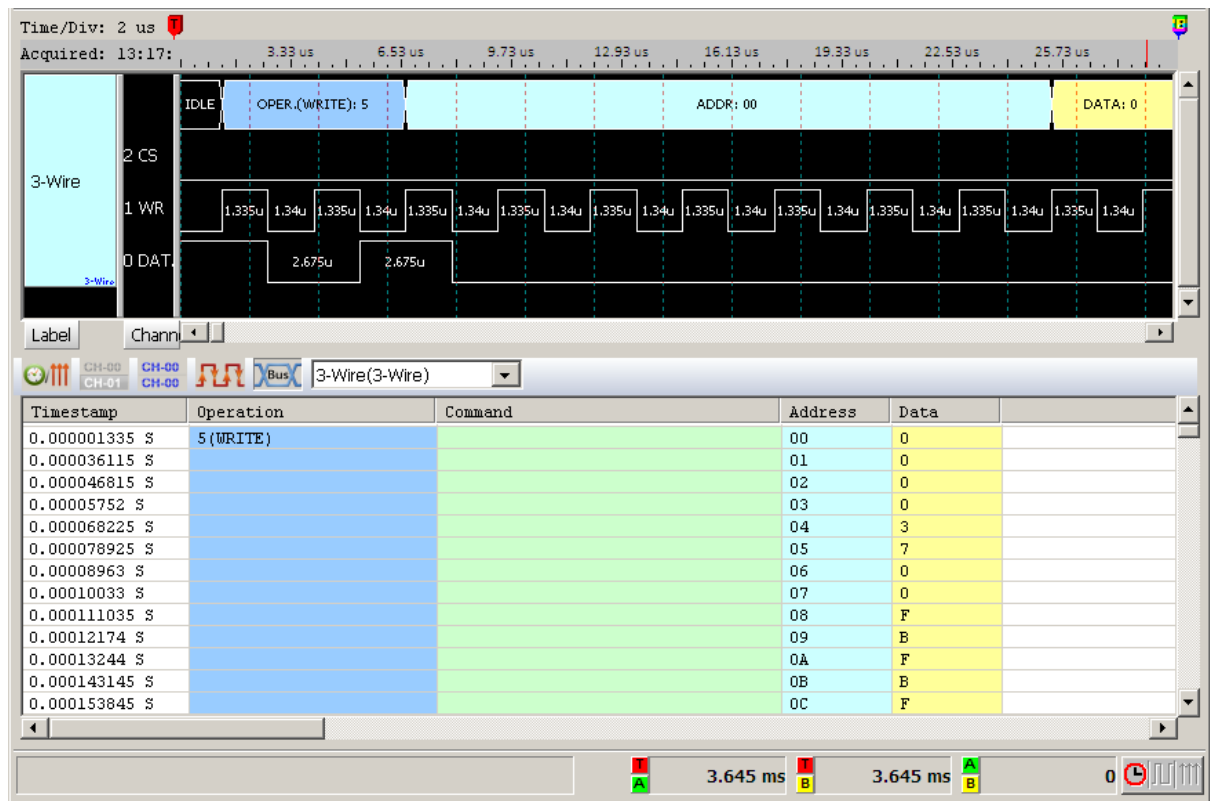
**Active High:** Select Active High.

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

**Rising:** Select Rising Data Edge.

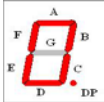
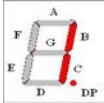
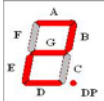
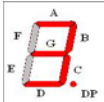
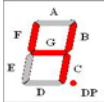
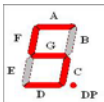
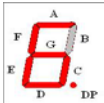
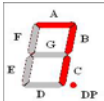
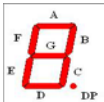
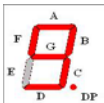
**Falling:** Select Falling Data Edge.

## Result

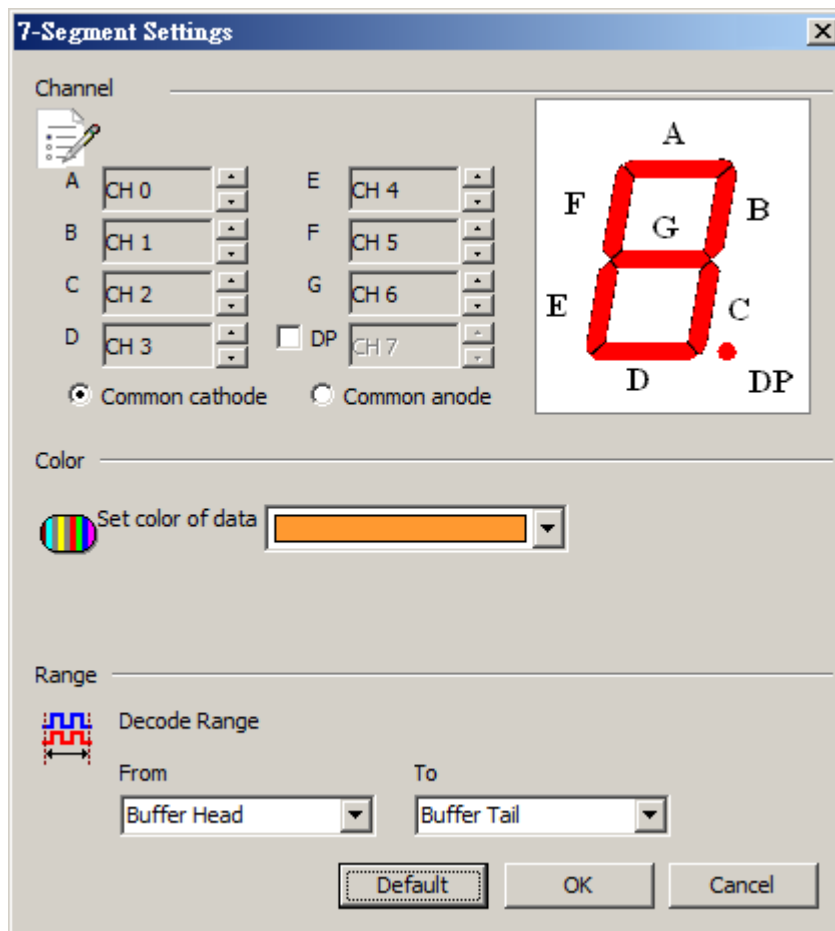


## 7-Segment

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



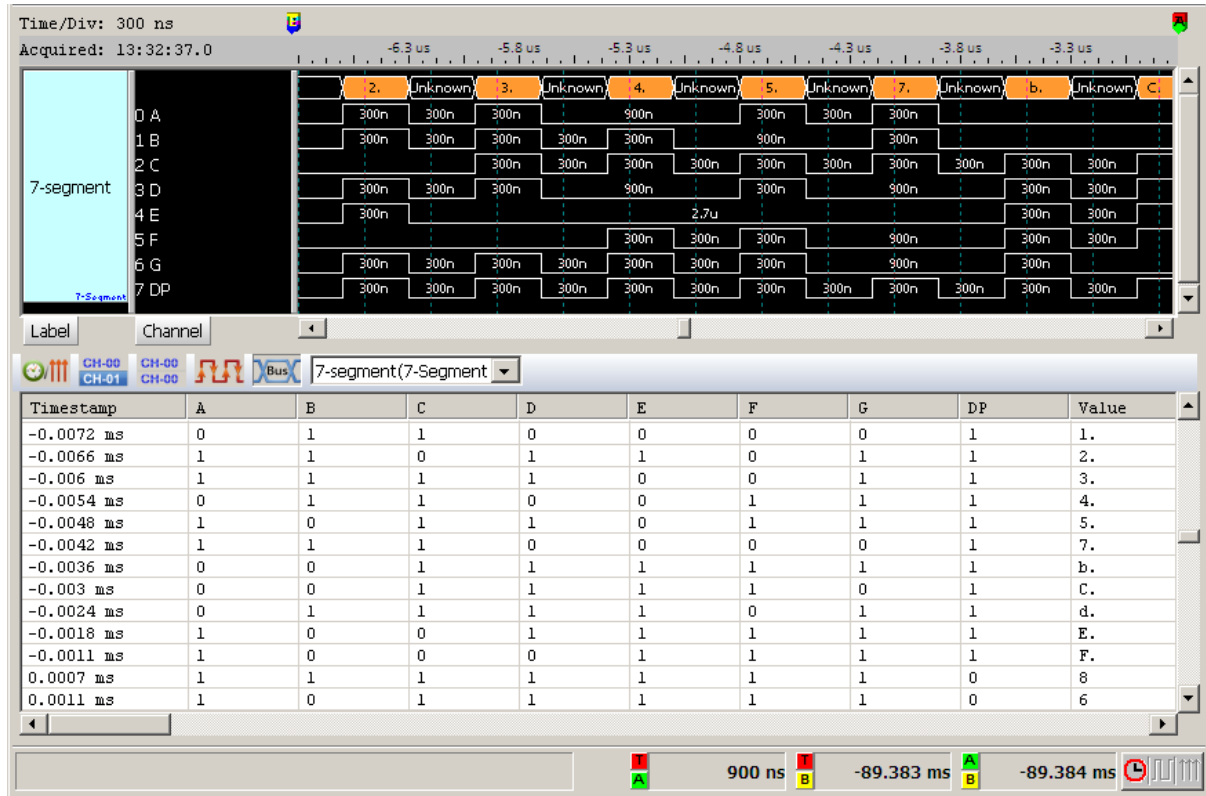
**Channel:** Show the selected channel (CH 0).

**DP:** to analysis decimal point.

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

## Result

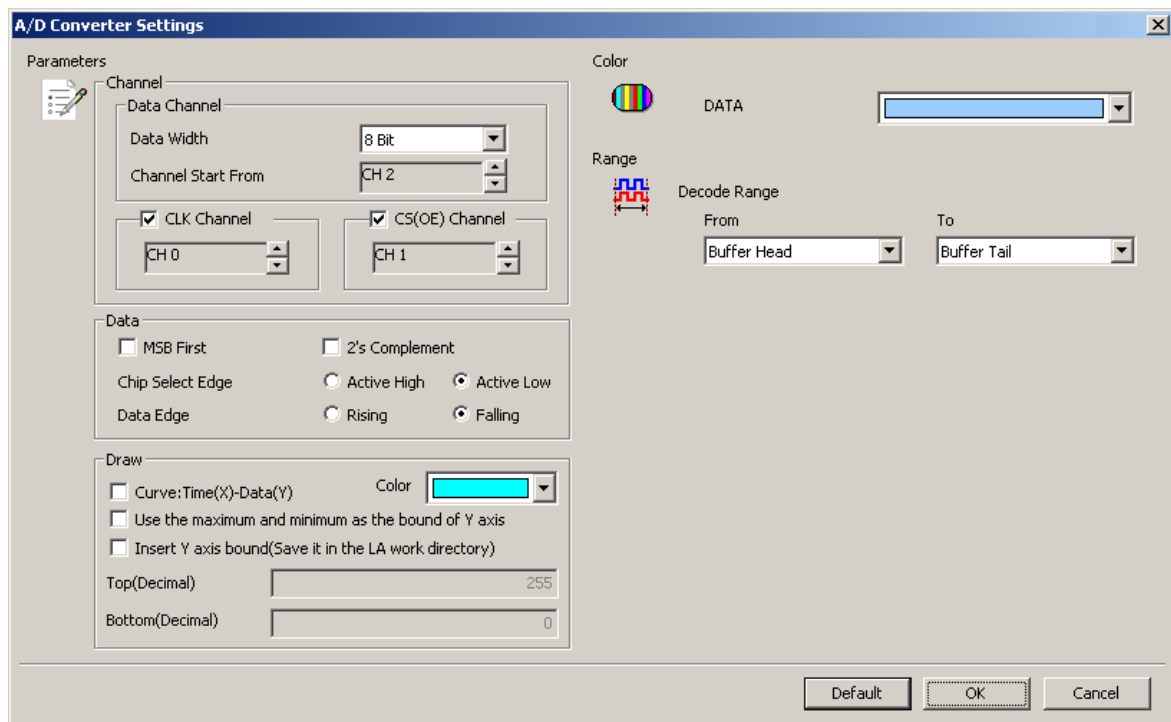
Click **OK** to run the 7 Segment decode and see the result on the Waveform Window below.



## 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.

**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.

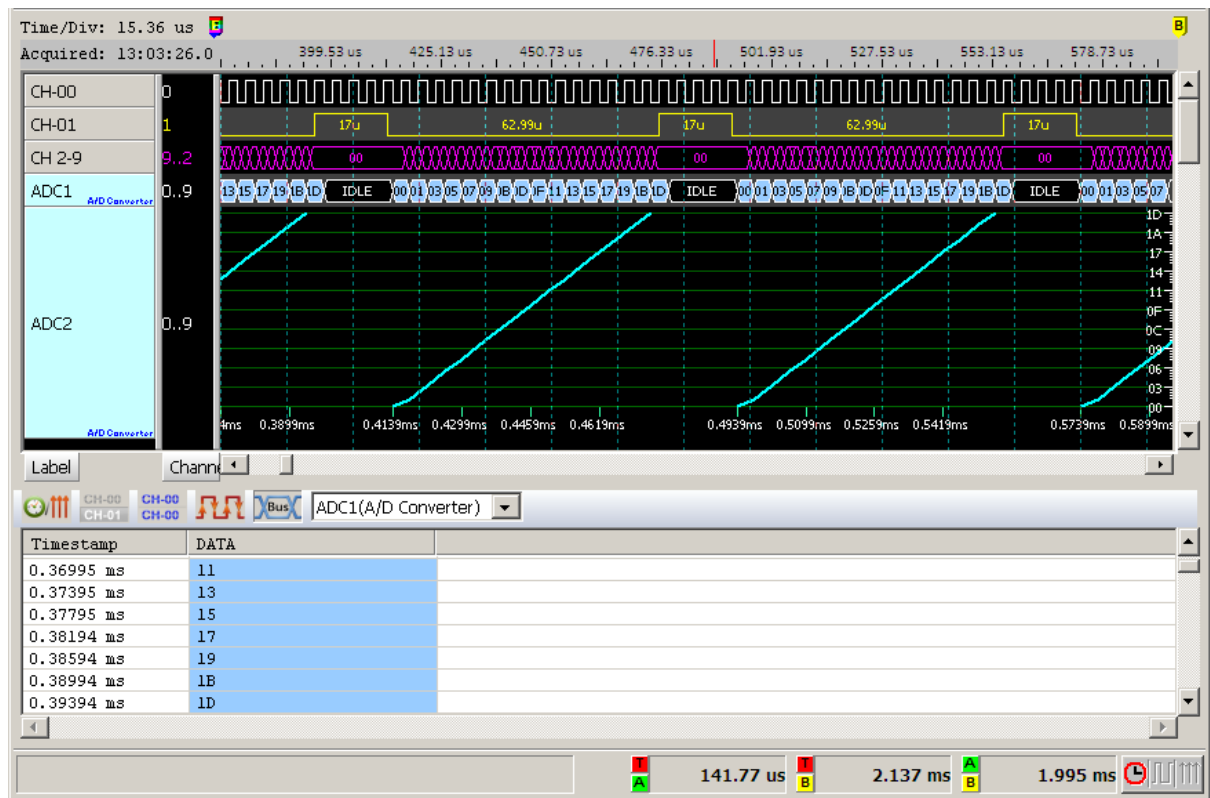
**Insert Y axis bound:** Set the maximum and minimum bounds of Y axis.

**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.

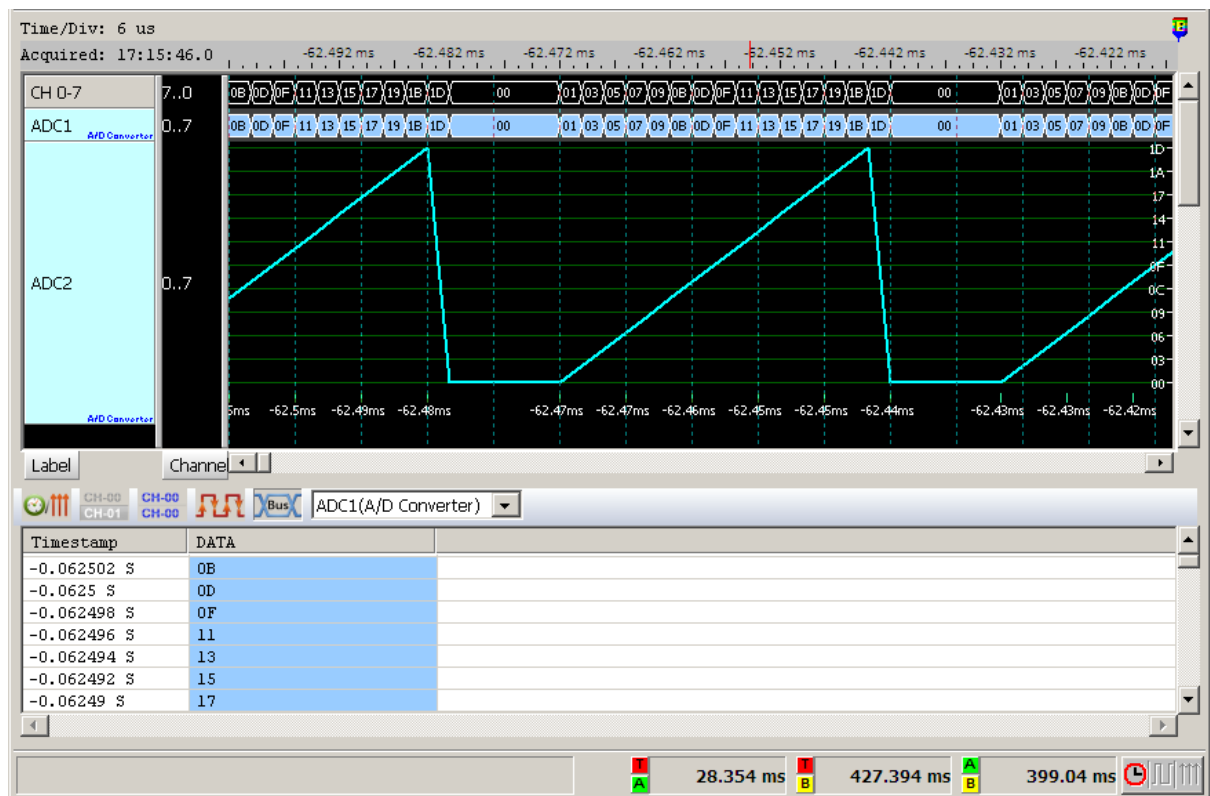
## **Result**

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

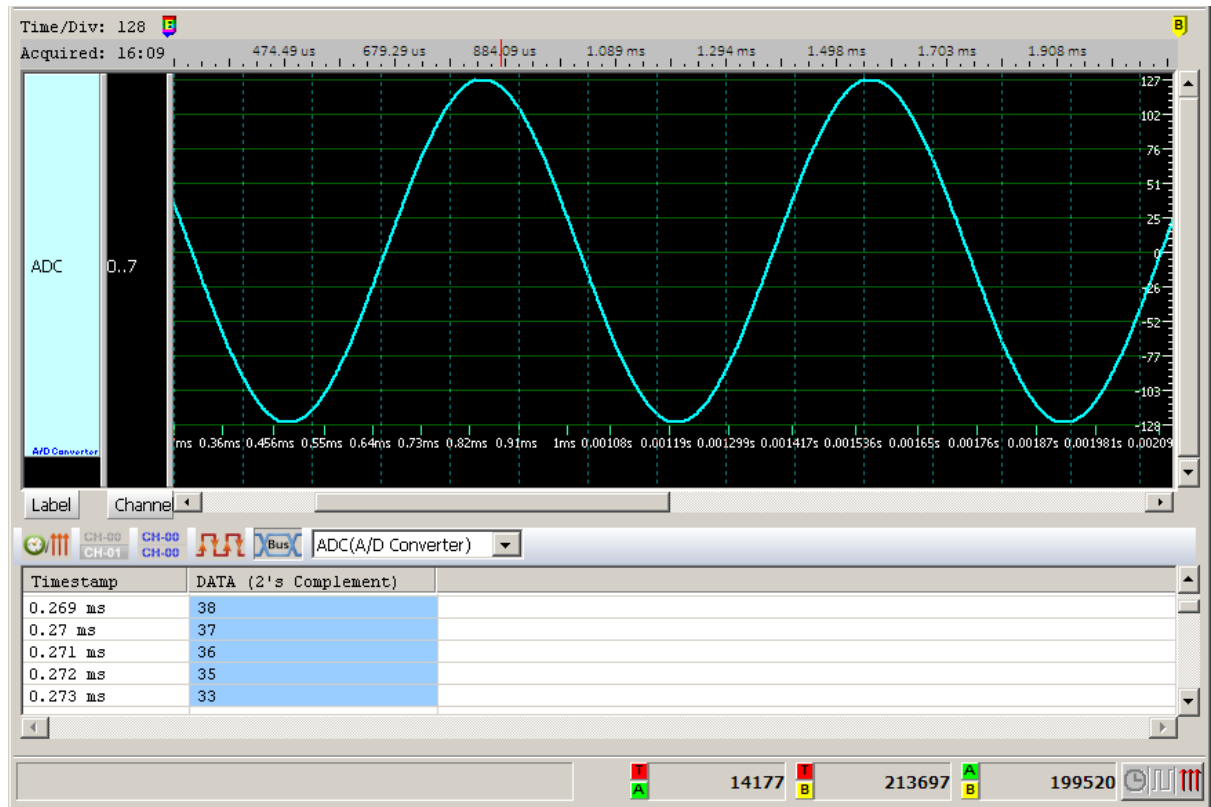
Select 8-bit data, CLK/CS channels:



Select 8-bit data:



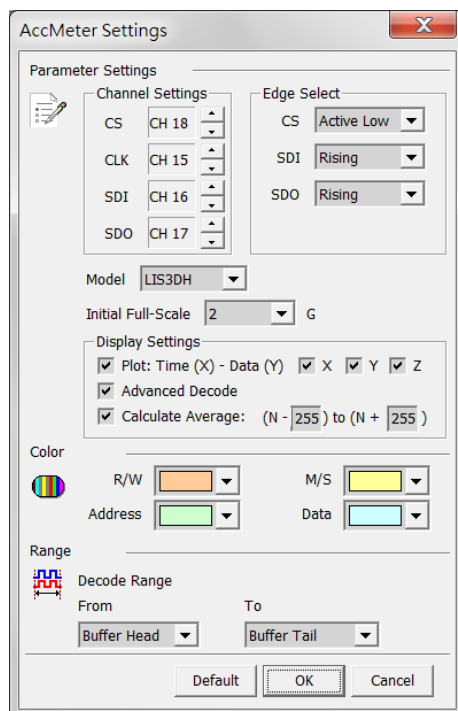
Select 8-bit data, 2's complement



## Accelerometer

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

### Settings



**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.

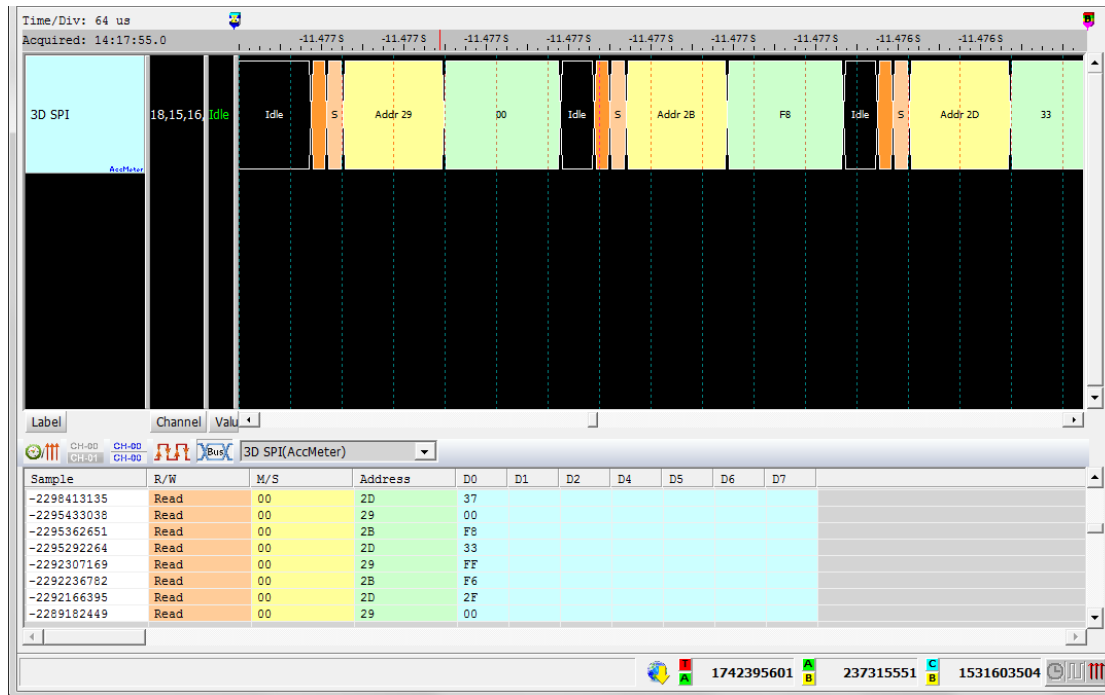
**Initial Full-Scale:** The default Full-Scale setting.

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

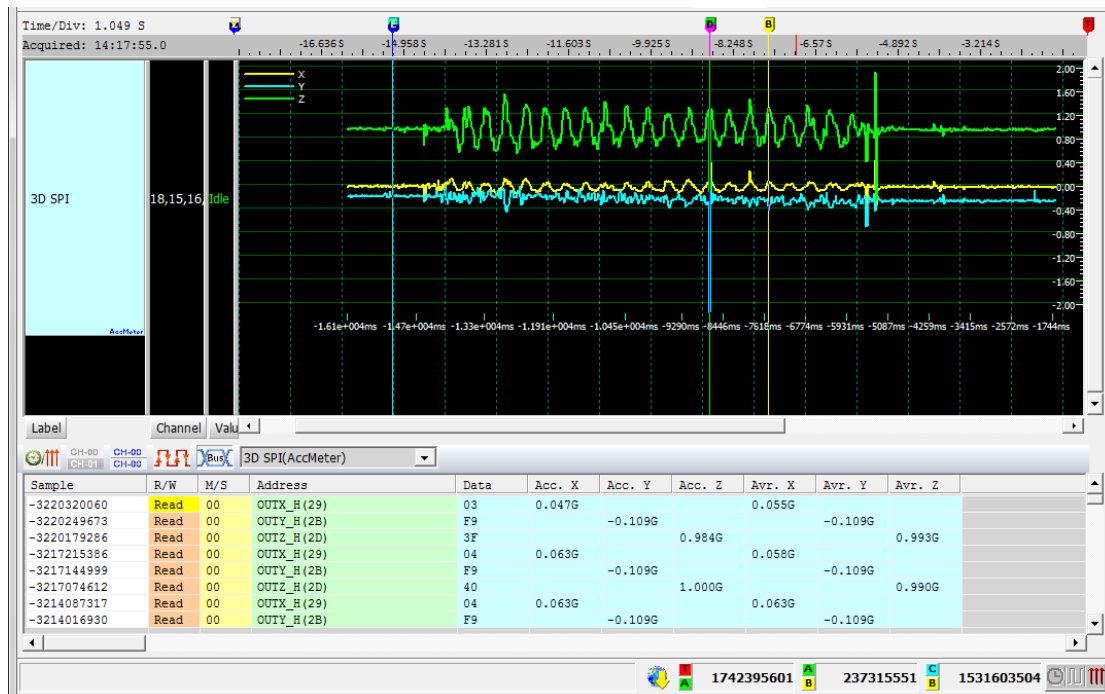
**Advanced Decode:** Enable/Disable the address, value convert function.

## Result

### Standard decoder result



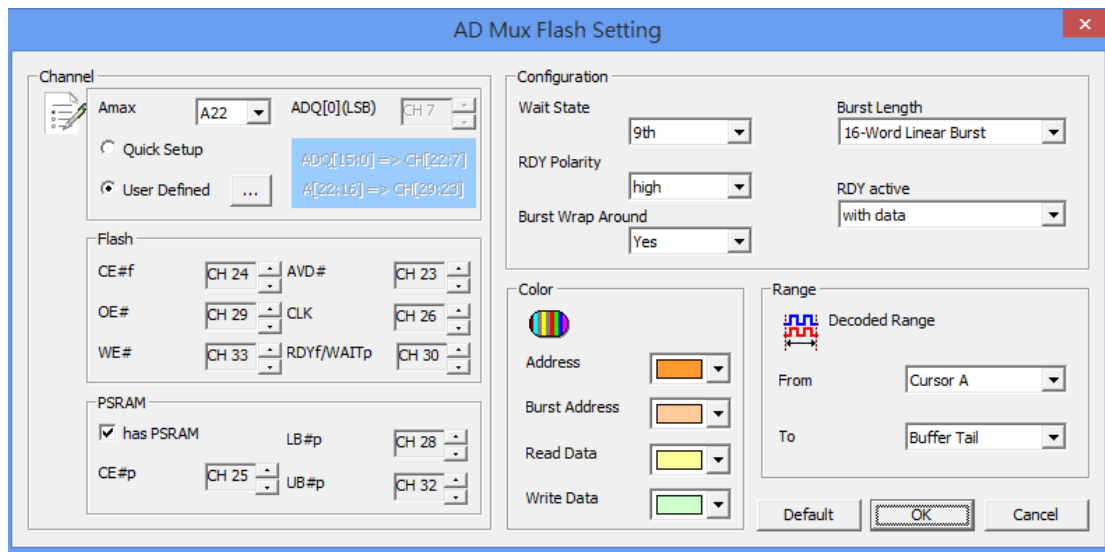
### Advanced decode result + Time-Value curve display



## AD-Mux Flash

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

### Settings



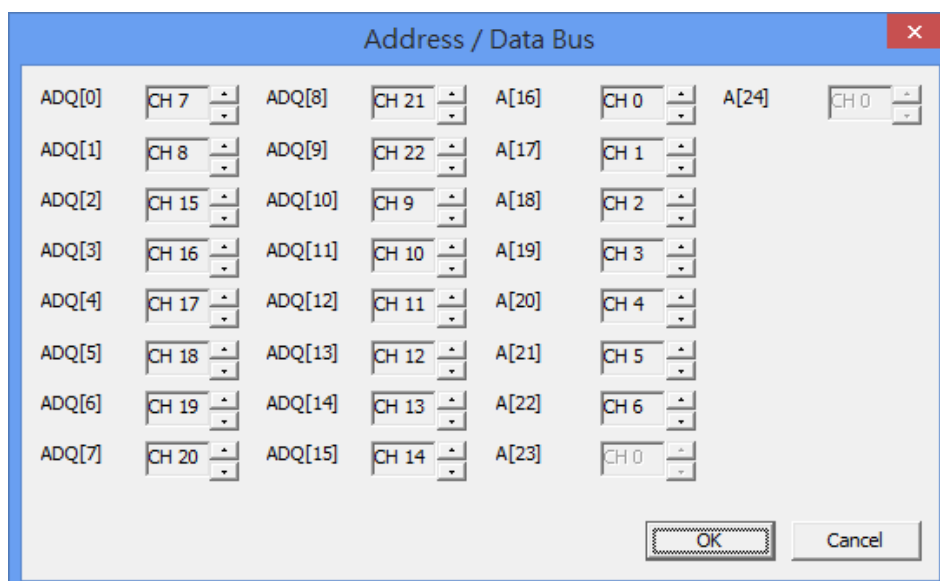
The **AD Mux Flash Setting** dialog box is divided into several sections:

- Channel:** Includes a dropdown for **Amax** (set to A22) and a dropdown for **ADQ[0] (LSB)** (set to CH 7). It has radio buttons for **Quick Setup** and **User Defined**. Below these are two lines of mapping: **ADQ[15:0] => CH[22:7]** and **A[22:16] => CH[29:23]**.
- Flash:** Contains settings for **CE#f** (CH 24), **AVD#** (CH 23), **OE#** (CH 29), **CLK** (CH 26), **WE#** (CH 33), and **RDYf/WAITp** (CH 30).
- PSRAM:** Includes a checkbox for **has PSRAM** (checked), **LB#p** (CH 28), **CE#p** (CH 25), and **UB#p** (CH 32).
- Configuration:** Includes **Wait State** (9th), **Burst Length** (16-Word Linear Burst), **RDY Polarity** (high), **Burst Wrap Around** (Yes), and **RDY active** (with data).
- Color:** Includes color selection for **Address** (orange), **Burst Address** (orange), **Read Data** (yellow), and **Write Data** (green).
- Range:** Includes a **Decoded Range** section with **From** (Cursor A) and **To** (Buffer Tail) dropdowns.

Buttons at the bottom include **Default**, **OK**, and **Cancel**.

**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:



The **Address / Data Bus** dialog box displays a grid of channel assignments for address and data lines:

Line	Channel	Line	Channel	Line	Channel
ADQ[0]	CH 7	ADQ[8]	CH 21	A[16]	CH 0
ADQ[1]	CH 8	ADQ[9]	CH 22	A[17]	CH 1
ADQ[2]	CH 15	ADQ[10]	CH 9	A[18]	CH 2
ADQ[3]	CH 16	ADQ[11]	CH 10	A[19]	CH 3
ADQ[4]	CH 17	ADQ[12]	CH 11	A[20]	CH 4
ADQ[5]	CH 18	ADQ[13]	CH 12	A[21]	CH 5
ADQ[6]	CH 19	ADQ[14]	CH 13	A[22]	CH 6
ADQ[7]	CH 20	ADQ[15]	CH 14	A[23]	CH 0

Buttons at the bottom include **OK** and **Cancel**.

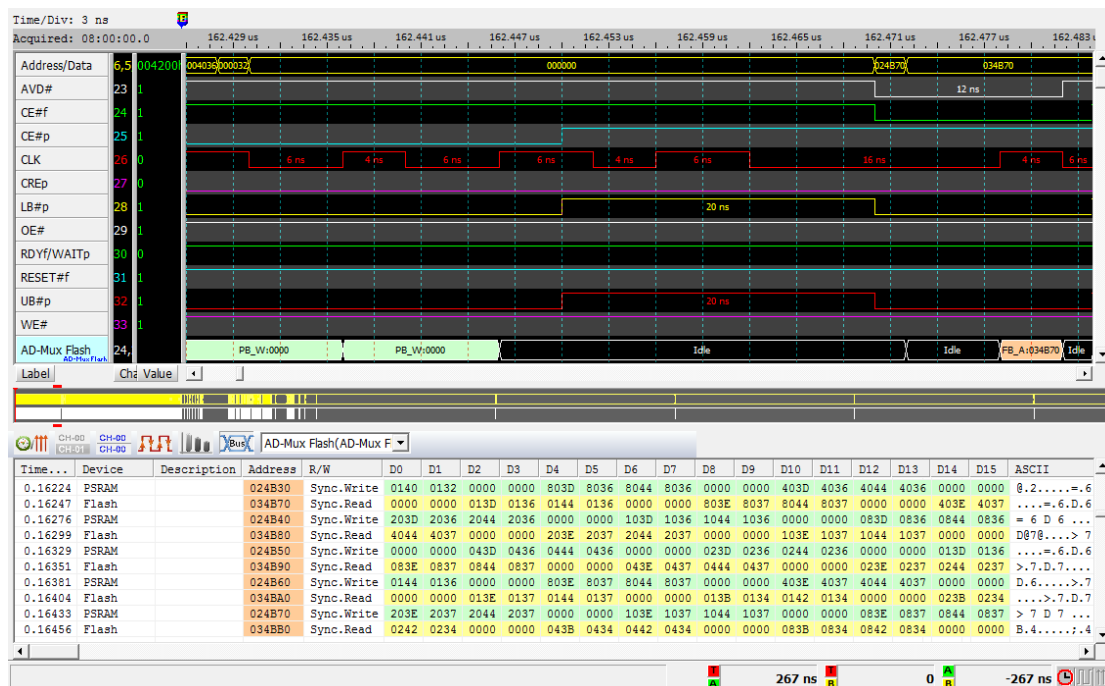
**Flash:** Control pins of flash.

**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

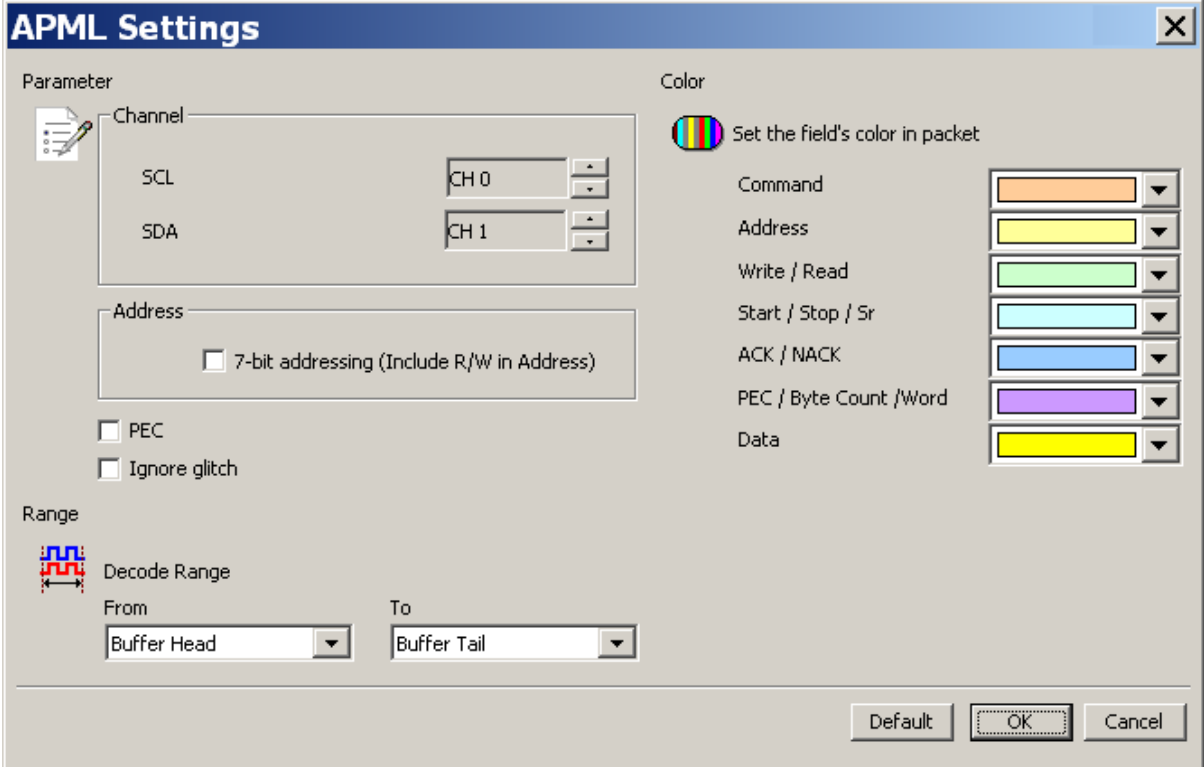
Click OK to run the AD-Mux Flash Decode and see result on the Waveform Windows below.



## Advanced Platform Management Link (APML)

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

### Settings



The APML Settings dialog box is divided into two main sections: Parameter and Color.

**Parameter Section:**

- Channel:** A list box containing 'SCL' and 'SDA'. To the right of each is a dropdown menu showing 'CH 0' for SCL and 'CH 1' for SDA.
- Address:** A checkbox labeled '7-bit addressing (Include R/W in Address)'.
- PEC:** A checkbox labeled 'PEC'.
- Ignore glitch:** A checkbox labeled 'Ignore glitch'.
- Range:** A section with a 'Decode Range' icon (a red and blue waveform) and two dropdown menus labeled 'From' and 'To', both set to 'Buffer Head' and 'Buffer Tail' respectively.

**Color Section:**

- Color:** A color selection icon (a circle with vertical bars of different colors) and the text 'Set the field's color in packet'.
- Command:** A dropdown menu with an orange color swatch.
- Address:** A dropdown menu with a yellow color swatch.
- Write / Read:** A dropdown menu with a light green color swatch.
- Start / Stop / Sr:** A dropdown menu with a light blue color swatch.
- ACK / NACK:** A dropdown menu with a medium blue color swatch.
- PEC / Byte Count / Word:** A dropdown menu with a purple color swatch.
- Data:** A dropdown menu with a yellow color swatch.

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

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

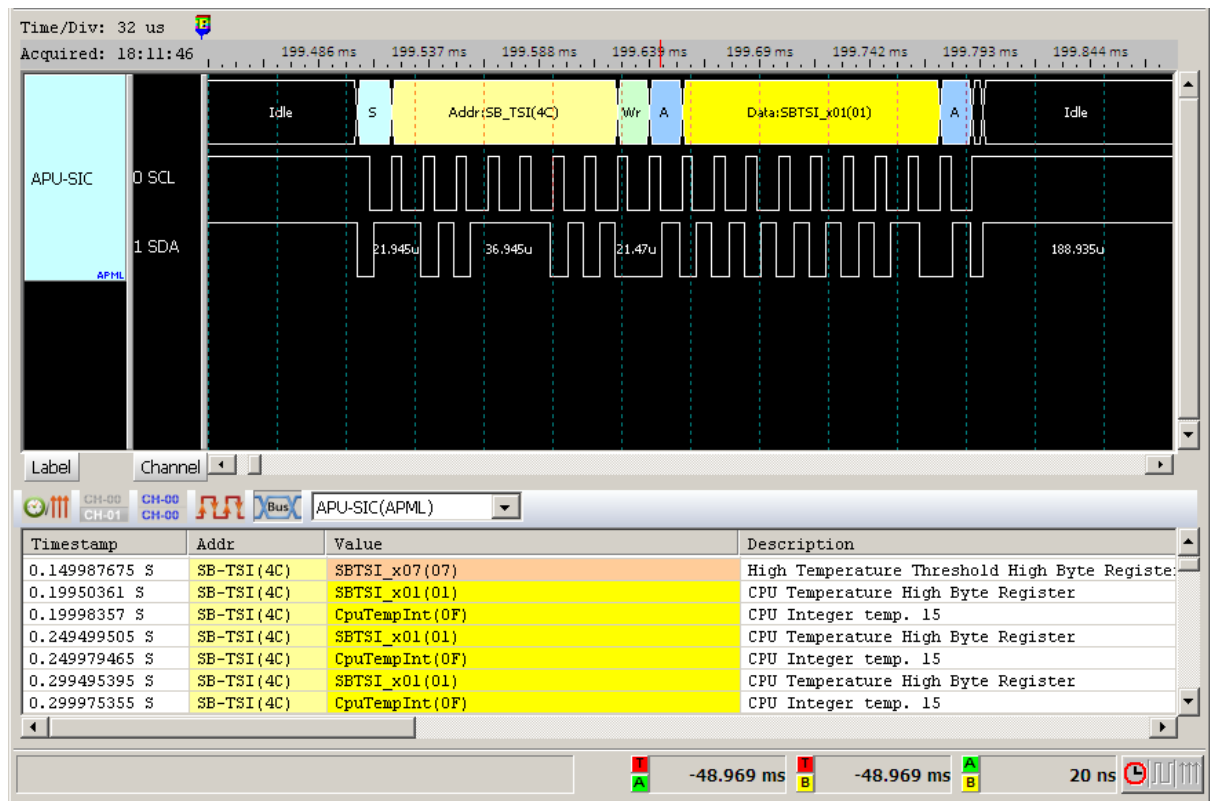
**7-bit addressing (Include R/W in Address):** Show 8-bit addressing (include 7-bit addressing and 1-bit R/W).

**PEC:** Packet Error Check.

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



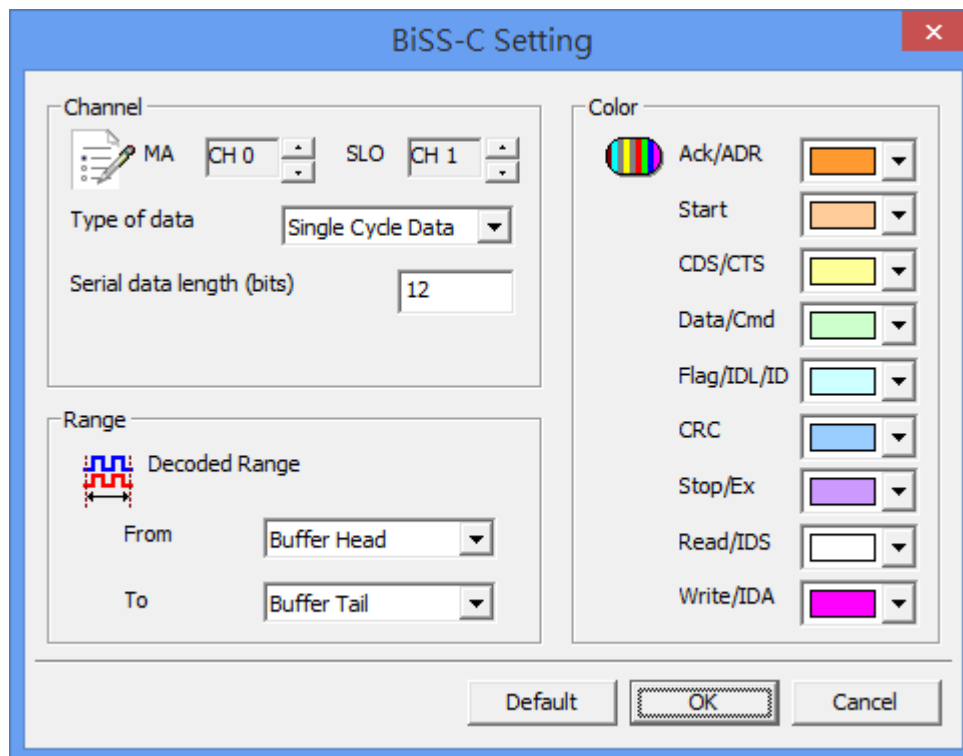
## Result



## 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



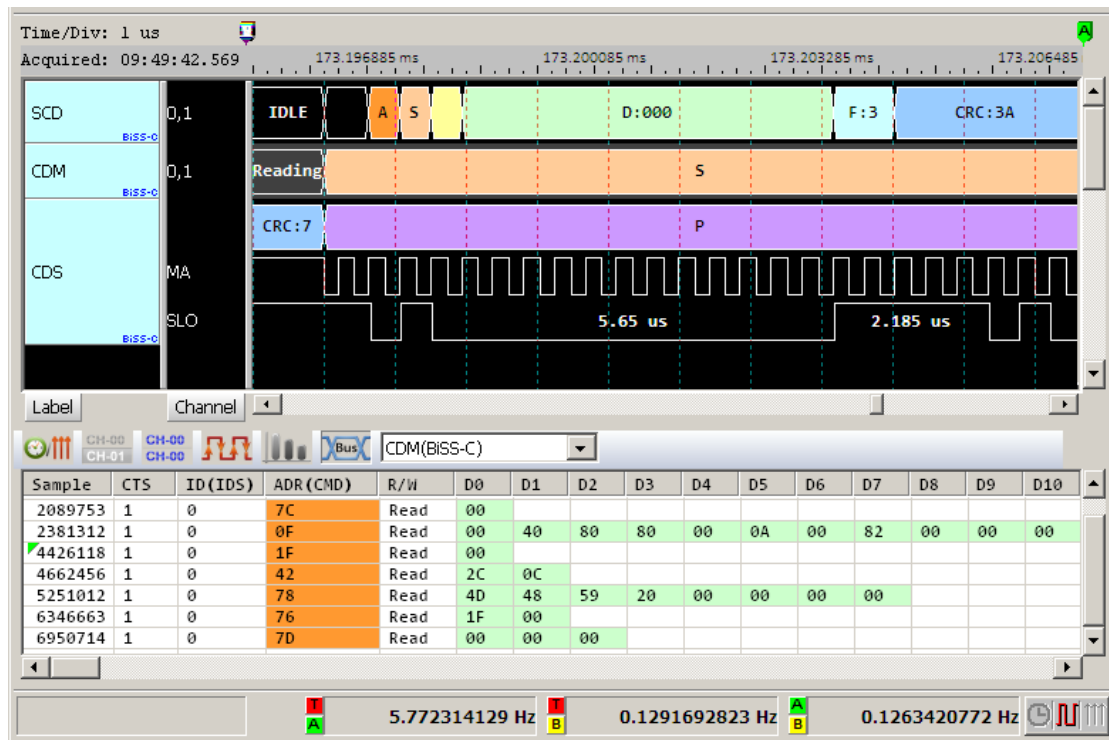
**MA/SLO:** Setting the channel of MA and SLO.

**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

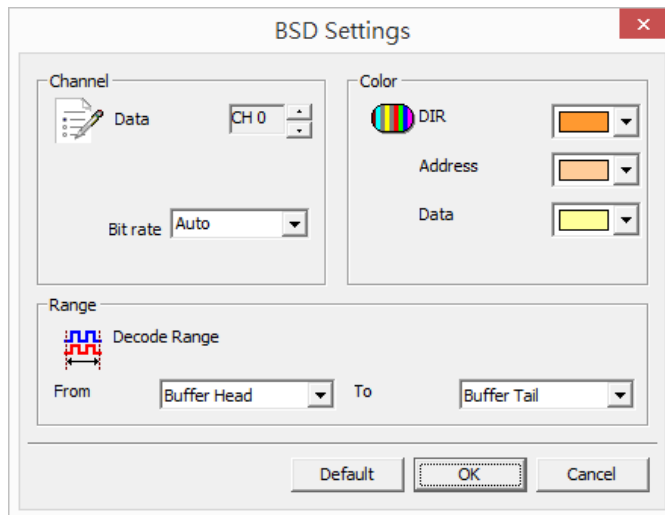
Click OK to run the BiSS-C Decode and see result on the Waveform Windows below.



## BSD

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

### Settings

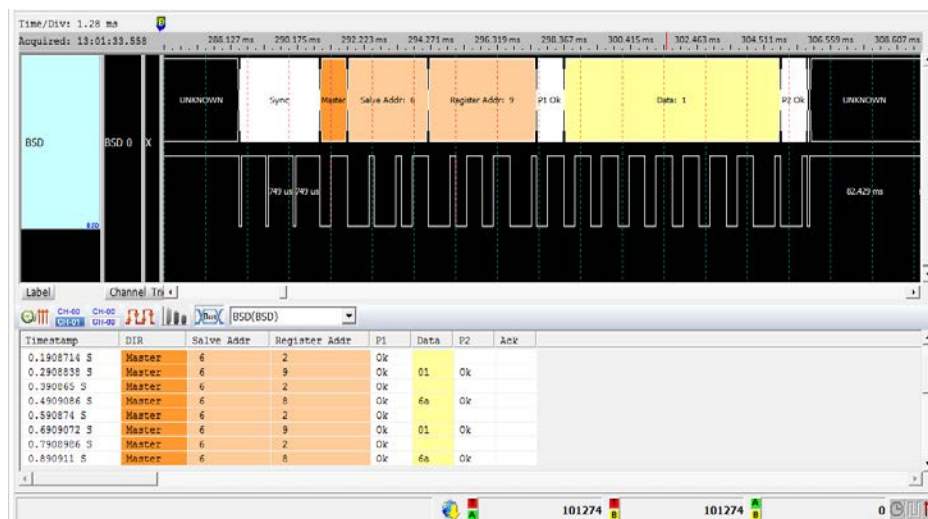


**Data:** The BSD data.

**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.

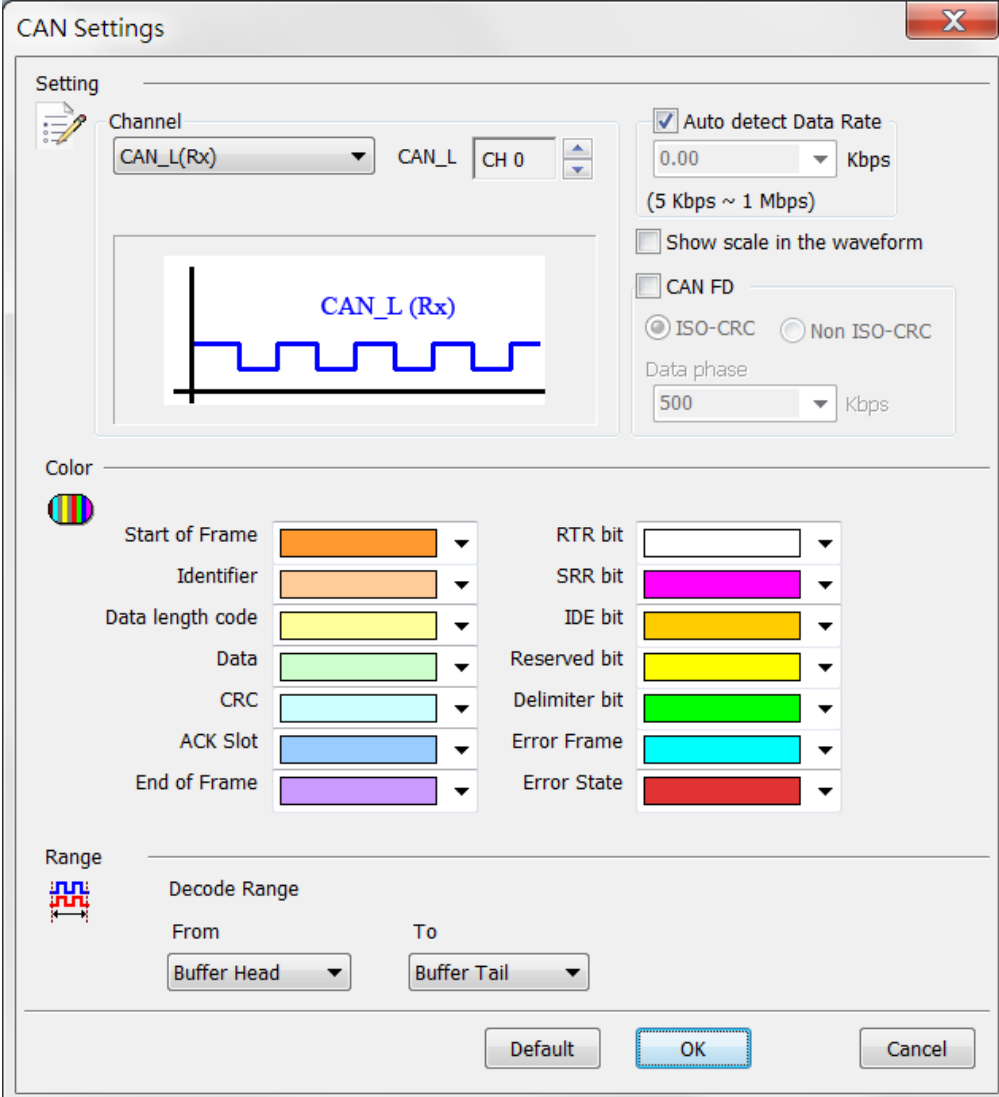


## CAN 2.0B/ CAN FD

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



The CAN Settings dialog box is divided into several sections:

- Setting**: Includes a Channel dropdown menu set to "CAN\_L(Rx)", a CAN\_L dropdown set to "CH 0", and a checkbox for "Auto detect Data Rate" which is checked. Below this is a speed selector set to "0.00 Kbps" with a range of "(5 Kbps ~ 1 Mbps)". There is also a checkbox for "Show scale in the waveform" which is unchecked.
- Color**: A section for configuring the colors of various CAN bus signals. It includes a color palette icon and two columns of color selection buttons:
  - Start of Frame (Orange)
  - Identifier (Light Orange)
  - Data length code (Yellow)
  - Data (Light Green)
  - CRC (Light Blue)
  - ACK Slot (Blue)
  - End of Frame (Purple)
  - RTR bit (White)
  - SRR bit (Magenta)
  - IDE bit (Yellow)
  - Reserved bit (Yellow)
  - Delimiter bit (Green)
  - Error Frame (Cyan)
  - Error State (Red)
- Range**: A section for defining the decode range. It includes a "Decode Range" label, a "From" dropdown set to "Buffer Head", and a "To" dropdown set to "Buffer Tail".

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

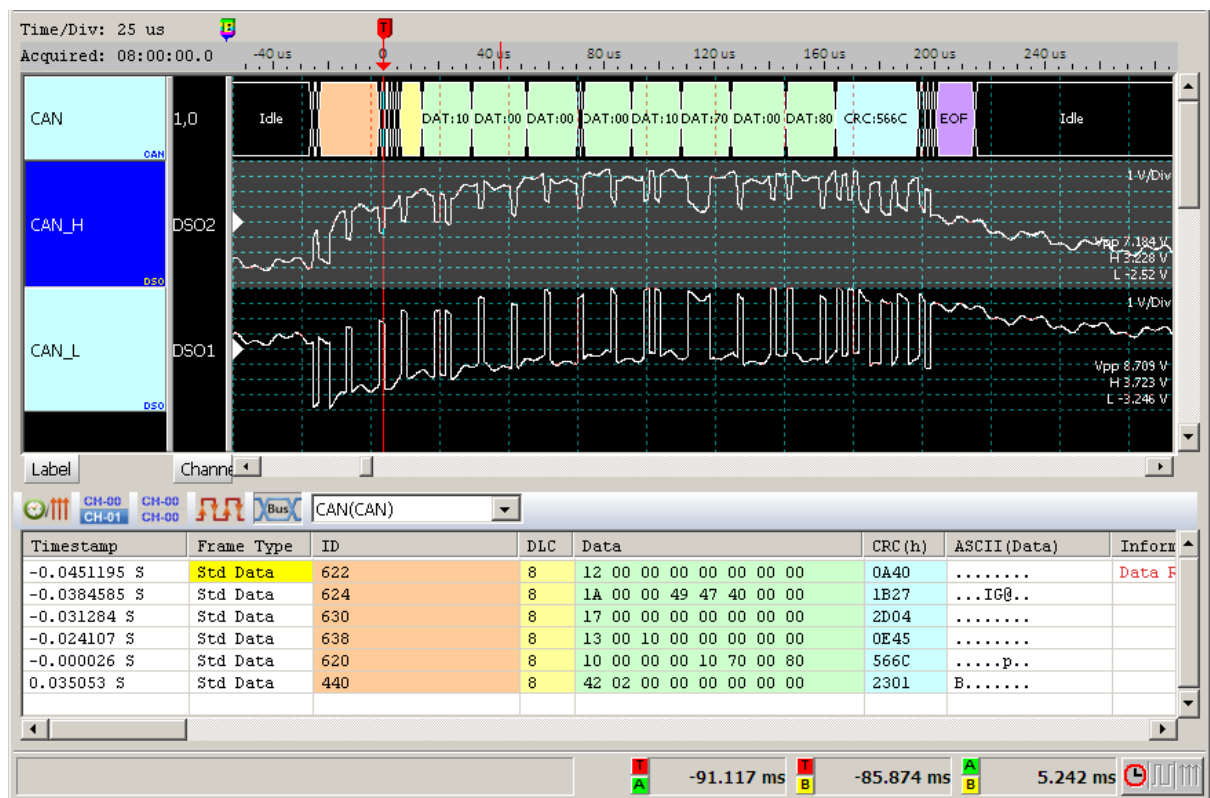
**Channel:** The differential data from the DSO channel (CAN\_H or CAN\_L) is shown by default.

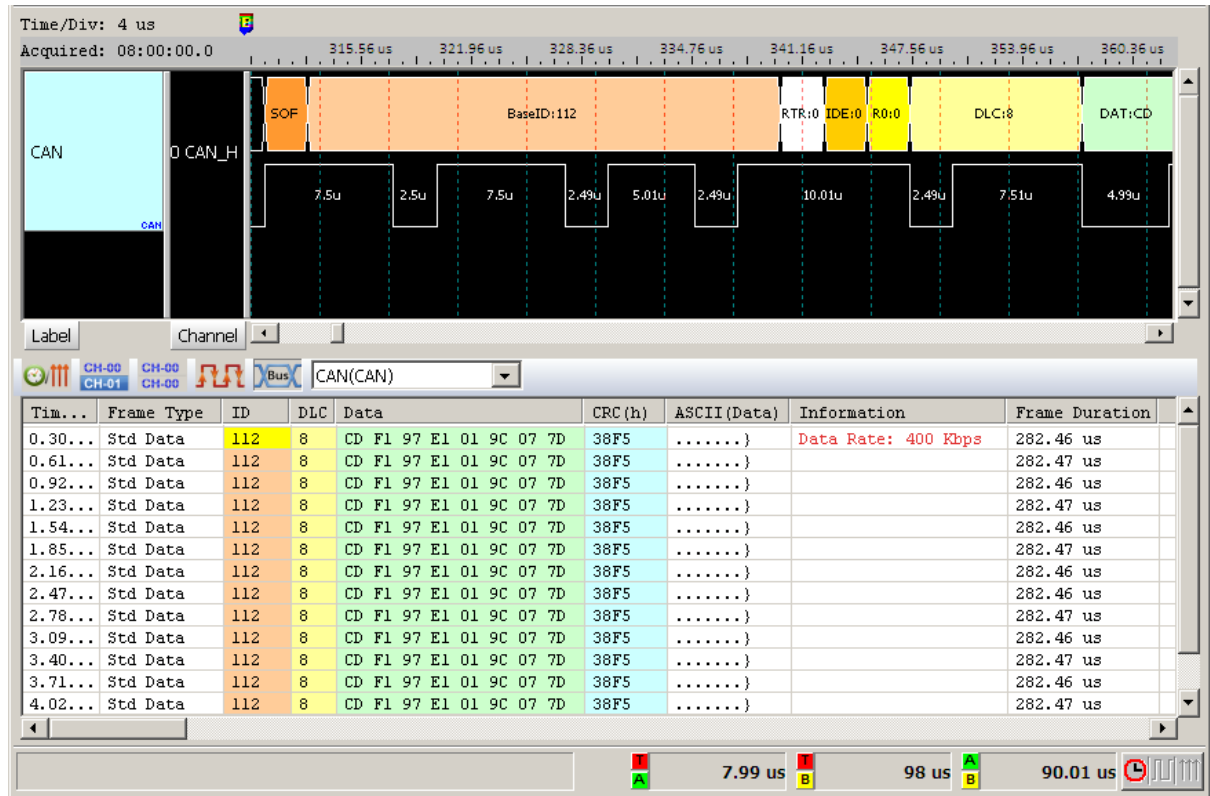
**Auto detect Data Rate:** Check this option for auto-detecting the CAN bit rate by the LA Viewer; this option will be disabled when enabling CAN FD decode, the maximum input range of the Data rate is from 5Kbps-1Mbps.

**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.

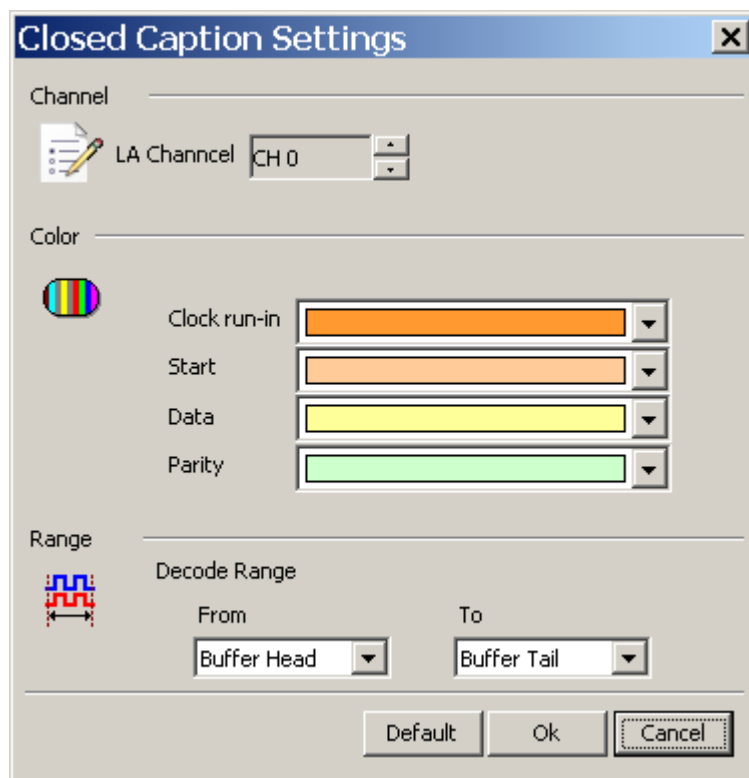




## Closed Caption

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

### Settings

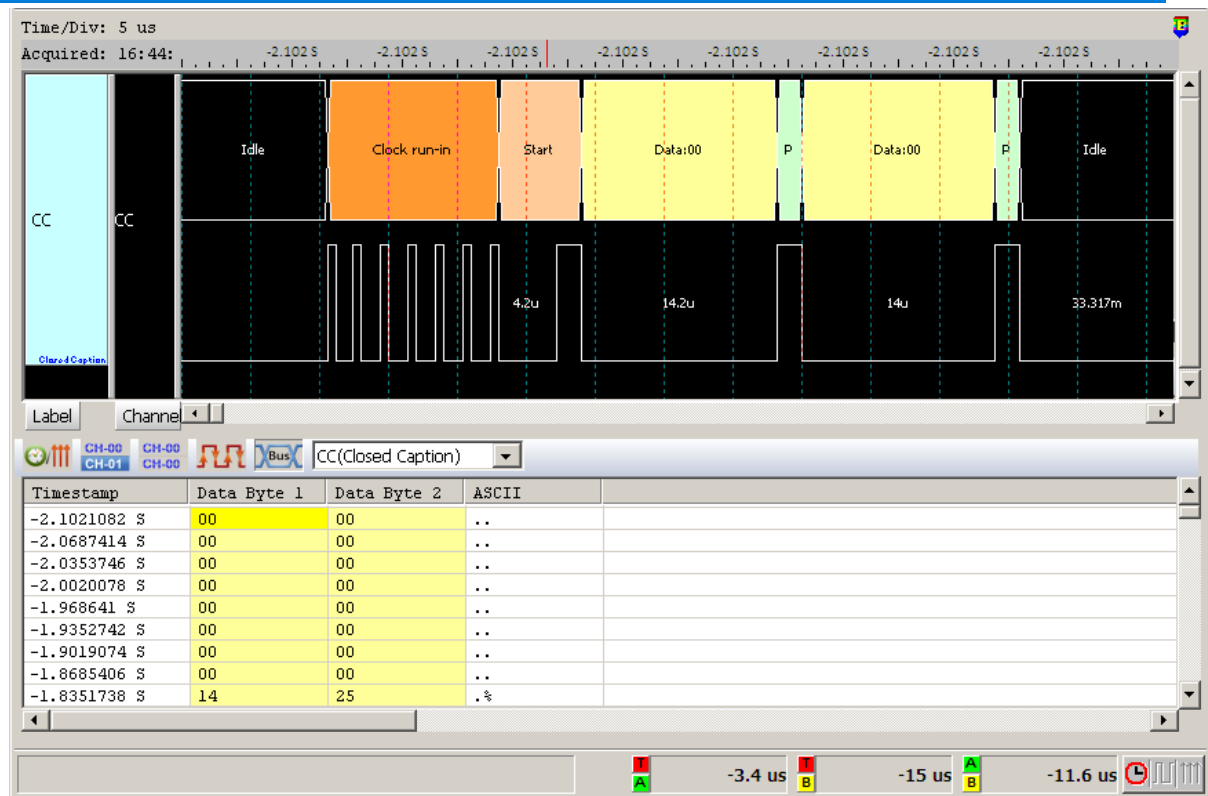


**LA Channel:** Show the selected channel (CH0).

### Result

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

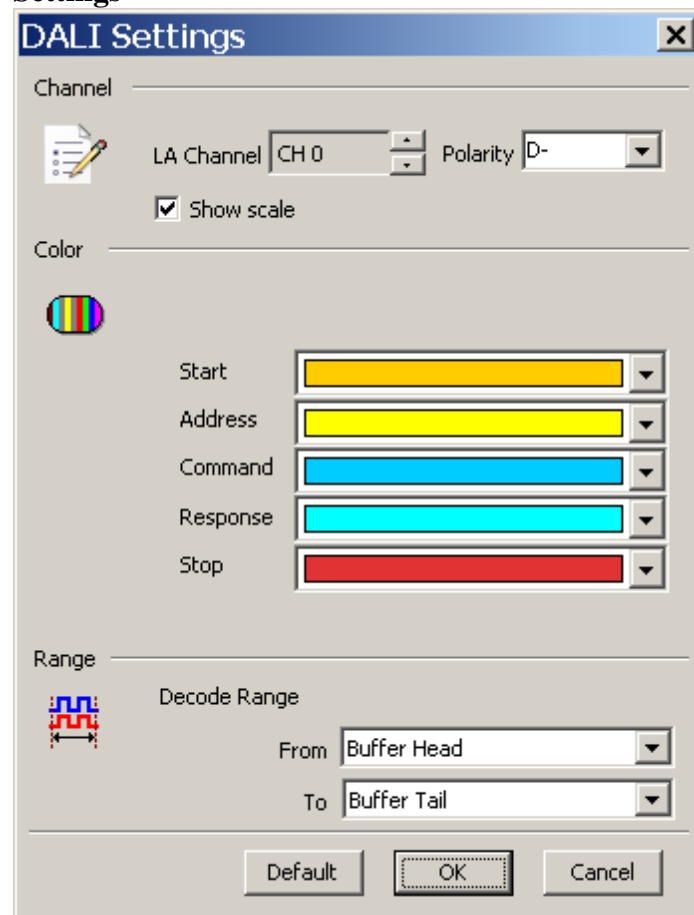




## DALI

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



The image shows a 'DALI Settings' dialog box with the following sections and controls:

- Channel**: A section with a notepad icon, an 'LA Channel' dropdown menu set to 'CH 0', and a 'Polarity' dropdown menu set to 'D-'. Below these is a checked checkbox labeled 'Show scale'.
- Color**: A section with a color wheel icon and five color-coded dropdown menus: 'Start' (yellow), 'Address' (yellow), 'Command' (blue), 'Response' (cyan), and 'Stop' (red).
- Range**: A section with a waveform icon and a 'Decode Range' section containing 'From' and 'To' dropdown menus, both set to 'Buffer Head' and 'Buffer Tail' respectively.
- Buttons**: At the bottom are three buttons: 'Default', 'OK', and 'Cancel'.

**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+.

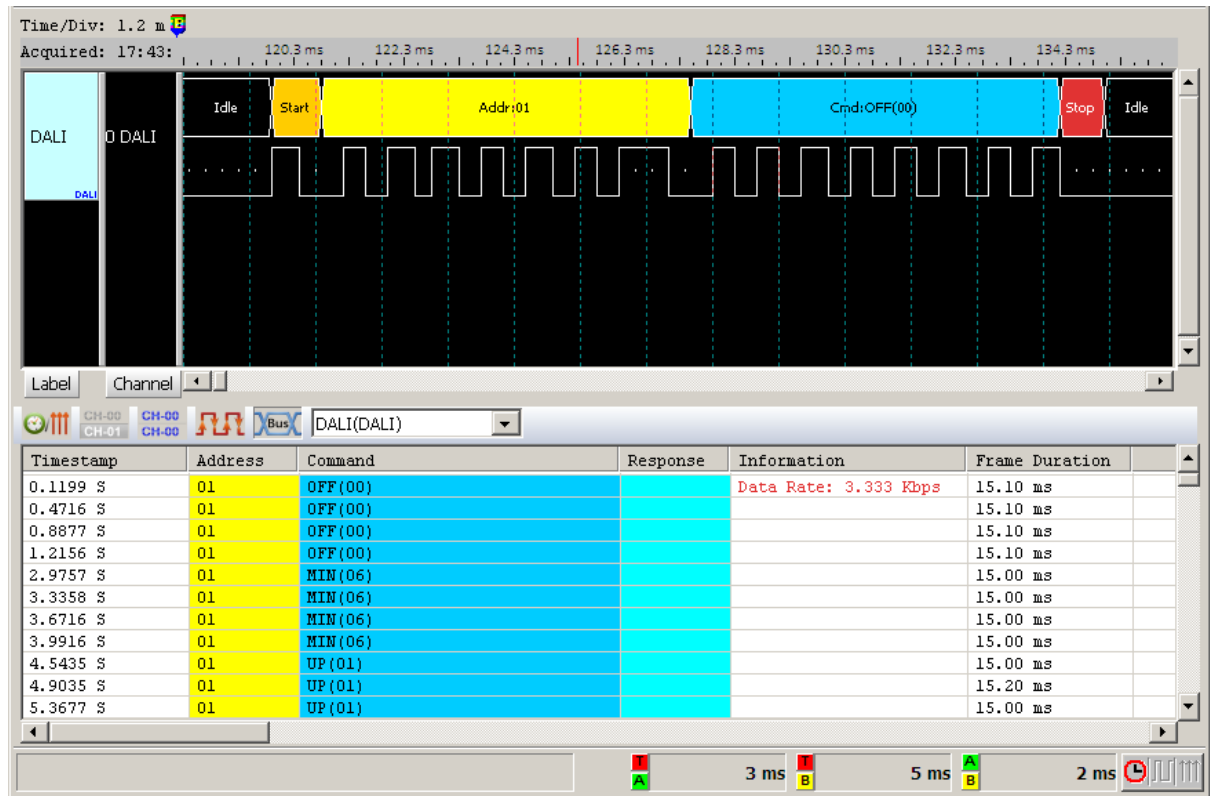
**Auto:** Automatically detect the polarity of the access end signal.

**Show scale:** Show Scale on the waveform.

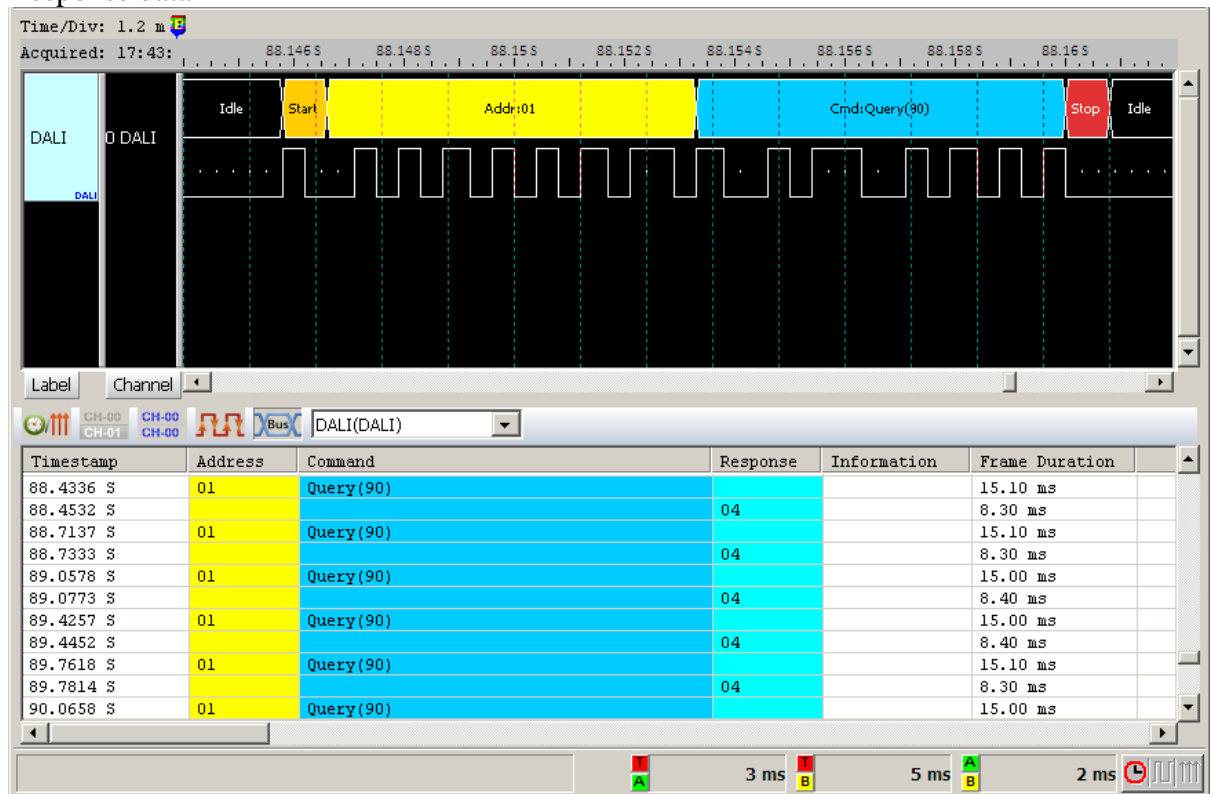
## Result

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

Send data



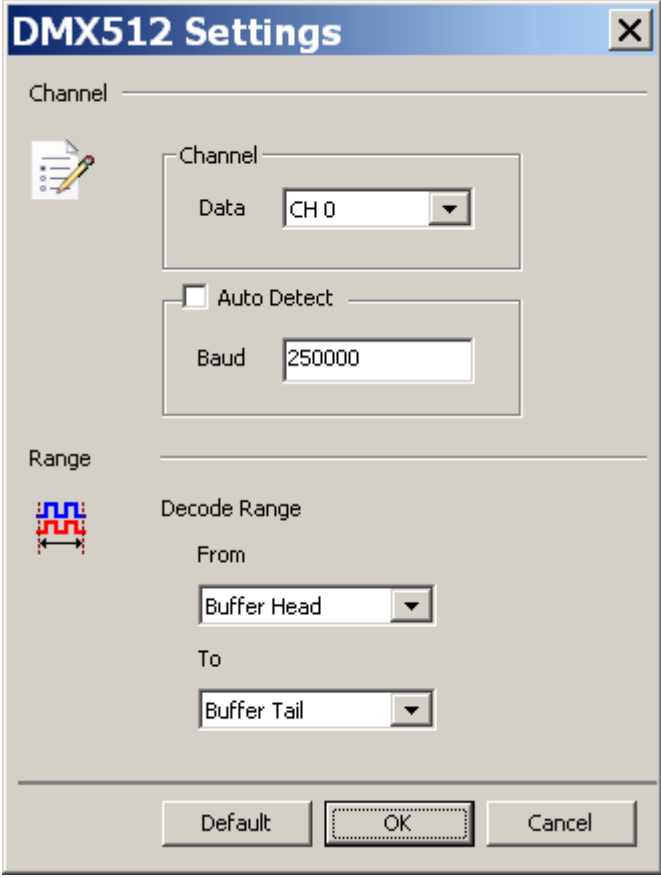
Response data



## DMX512

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

### Settings



The image shows a Windows-style dialog box titled "DMX512 Settings". It has a close button (X) in the top right corner. The dialog is divided into two main sections: "Channel" and "Range".

**Channel Section:**

- There is a "Channel" label followed by a text box containing "CH 0" and a dropdown arrow.
- Below this is an "Auto Detect" checkbox, which is currently unchecked.
- Below the checkbox is a "Baud" label followed by a text box containing "250000".

**Range Section:**

- There is a "Range" label followed by a small icon of a waveform with a double-headed arrow.
- Below this is a "Decode Range" section with two labels: "From" and "To".
- The "From" label is followed by a text box containing "Buffer Head" and a dropdown arrow.
- The "To" label is followed by a text box containing "Buffer Tail" and a dropdown arrow.

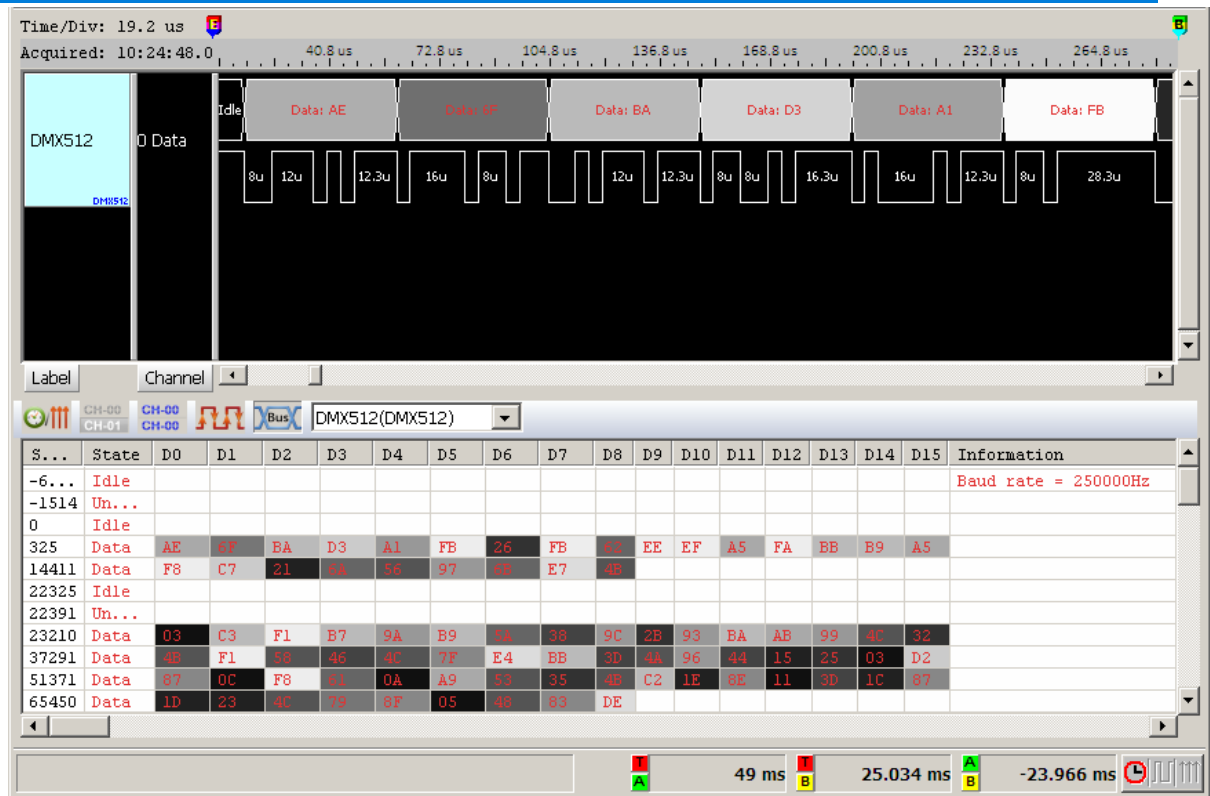
At the bottom of the dialog, there are three buttons: "Default", "OK", and "Cancel". The "OK" button is highlighted with a dashed border.

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

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

### Result

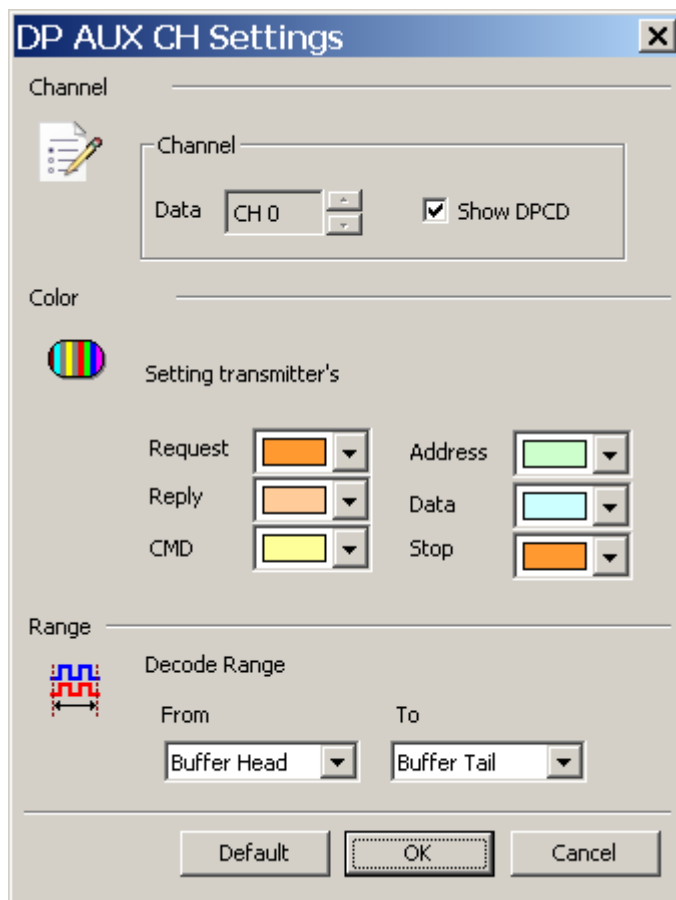
Use grayscale to display the decode results.



## 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.

### Settings



The image shows a Windows-style dialog box titled "DP AUX CH Settings". It is divided into three main sections: Channel, Color, and Range.

- Channel Section:** Contains a "Channel" label, a "Data" dropdown menu set to "CH 0", and a checked checkbox labeled "Show DPCD".
- Color Section:** Features a color calibration icon and the text "Setting transmitter's". It includes six color selection boxes arranged in two columns: Request (orange), Reply (light orange), CMD (yellow) on the left; and Address (light green), Data (light blue), Stop (orange) on the right.
- Range Section:** Includes a waveform icon and the text "Decode Range". It has "From" and "To" labels with dropdown menus set to "Buffer Head" and "Buffer Tail" respectively.

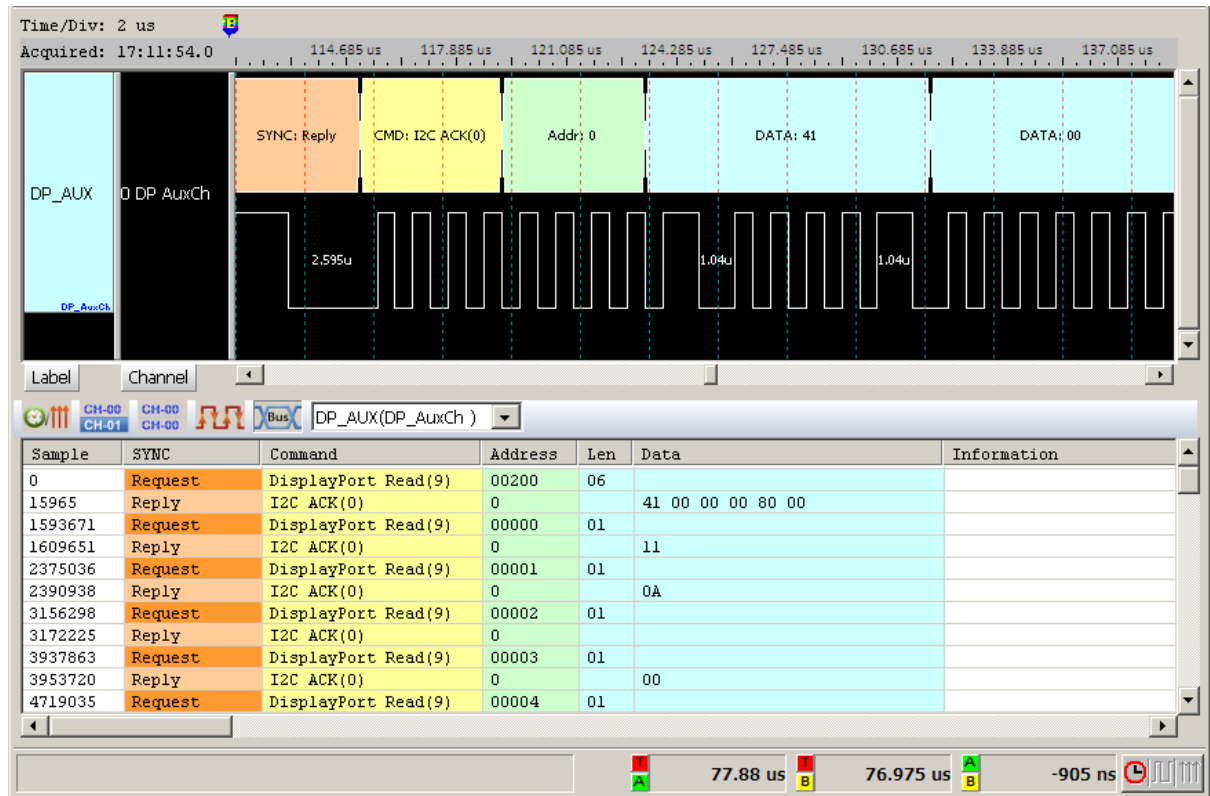
At the bottom of the dialog are three buttons: "Default", "OK", and "Cancel".

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

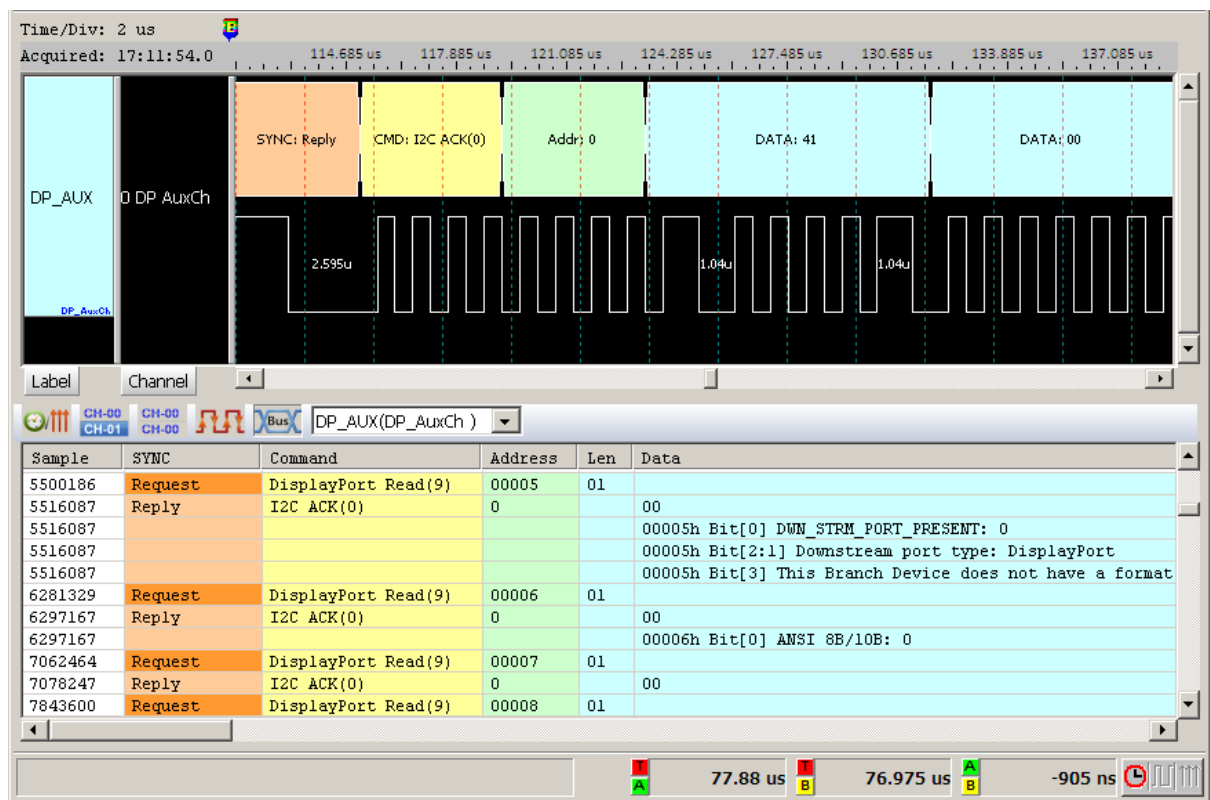
**Show DPCD:** Show the Display Port Configuration data.

## Result

Without the DPCD information



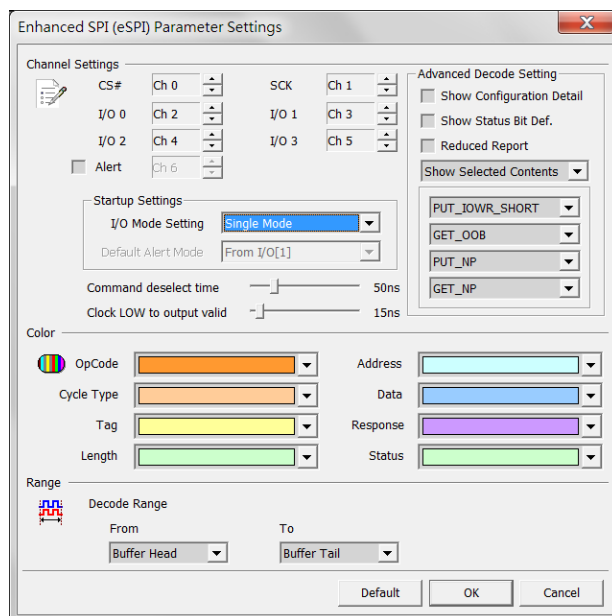
Show the DPCD information



## 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



### Channel:

- CS#: Chip Select (Active Low)
- SCK: Clock
- I/O0 – I/O3: Data input / output
- Alert: Alert 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.

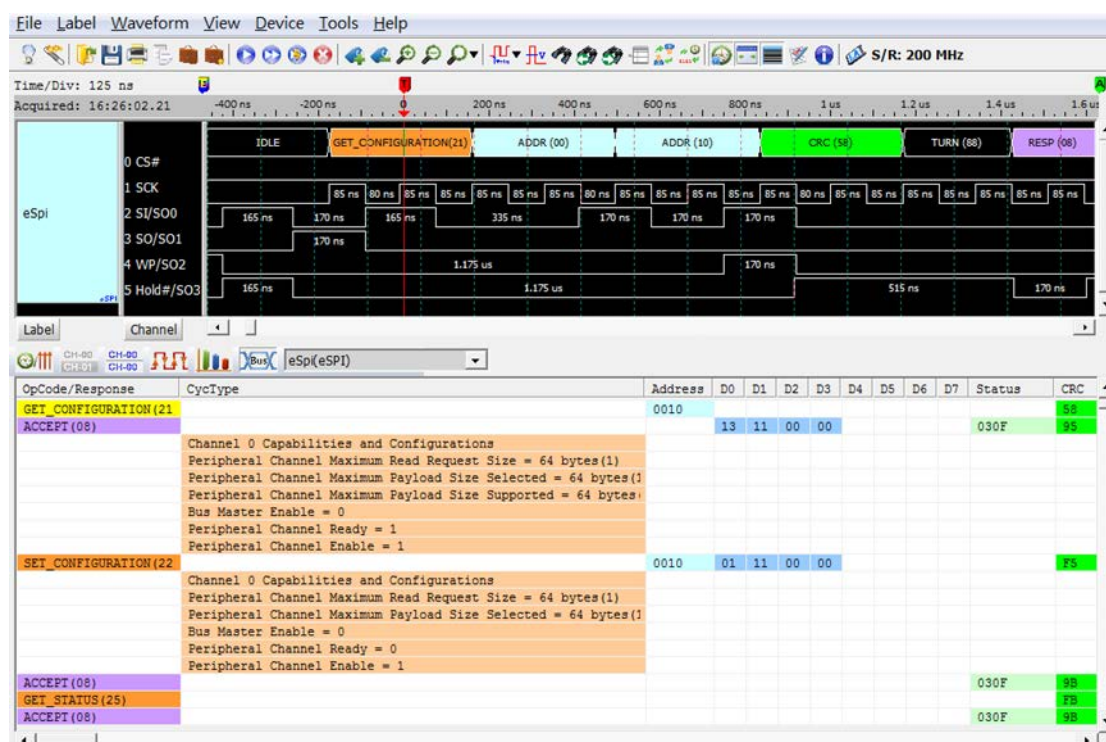
**Reduced Report:** Reducing the report is easy to check the Command Flow.

**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.

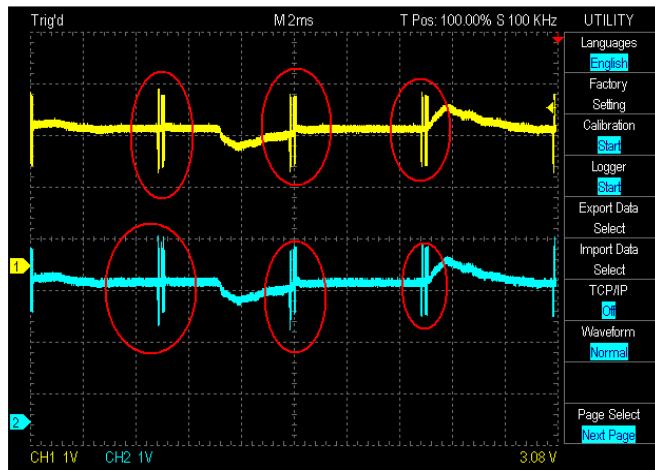


## FlexRay

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

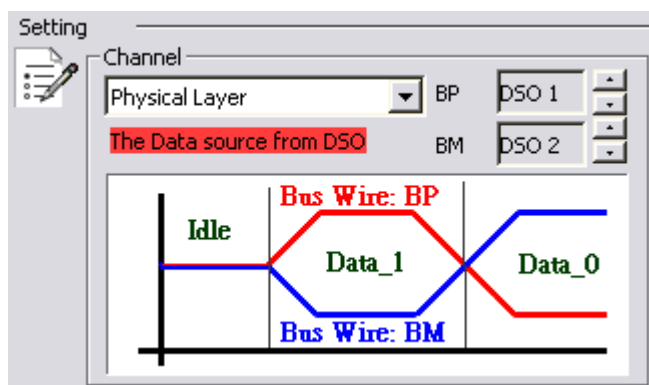
### FlexRay signal

**Physical Layer:** We use the DSO to measure the differential signal at the FlexRay physical layer and get the blue-color BP signal and the yellow-color BM signal as the picture below.



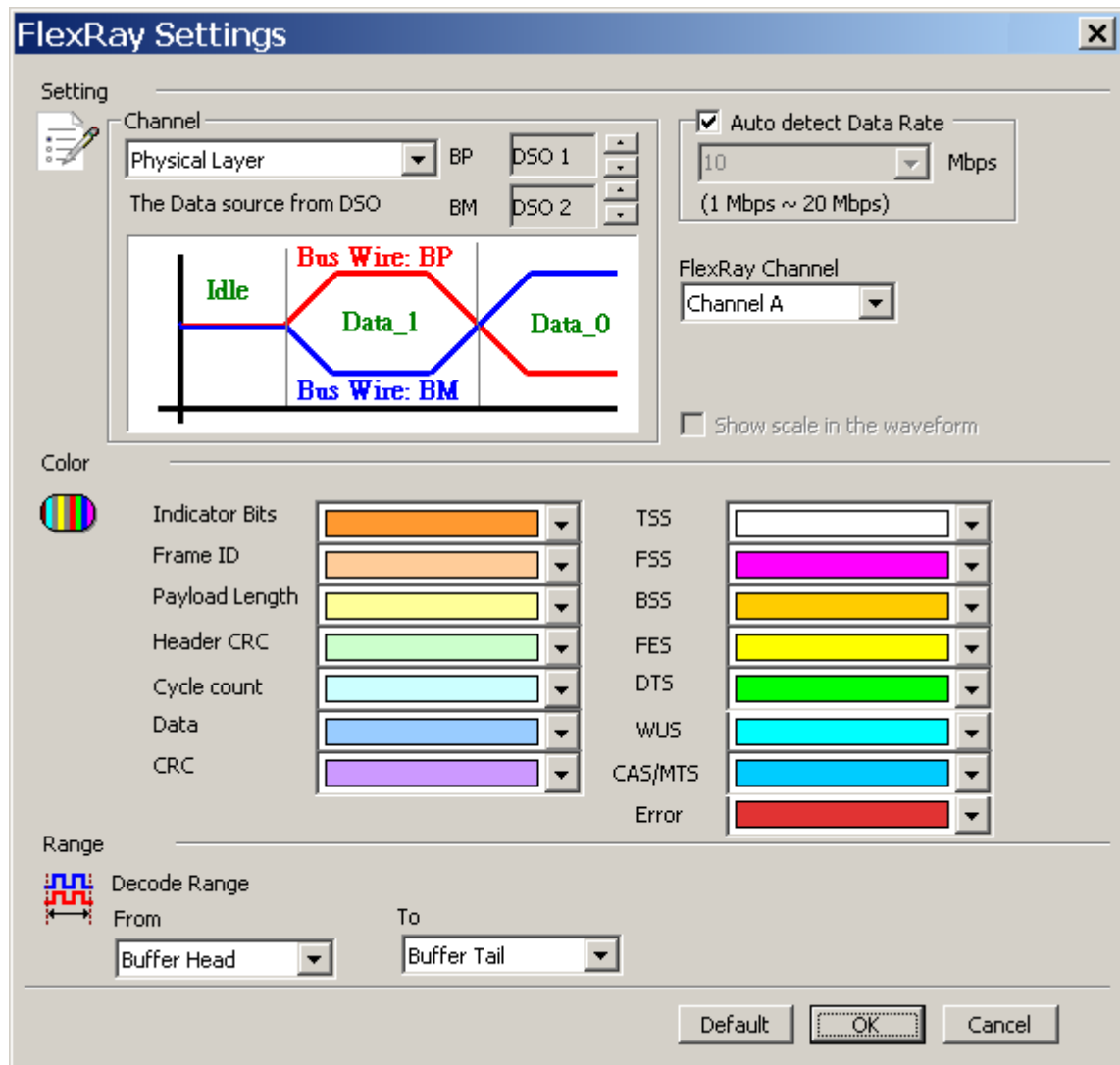
Then, we subtract BP and BM values and get the red-color signal used by the logic analyzer to decode the FlexRay bus as the picture above.

In order to trigger from the logic analyzer, you need to connect the logic analyzer to either the BP or BM pin in the setting dialog on the right.



**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.

**Physical Layer:** The signals (BP, BM) are from the DSODSO channels can be Ch1 - Ch6.

**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

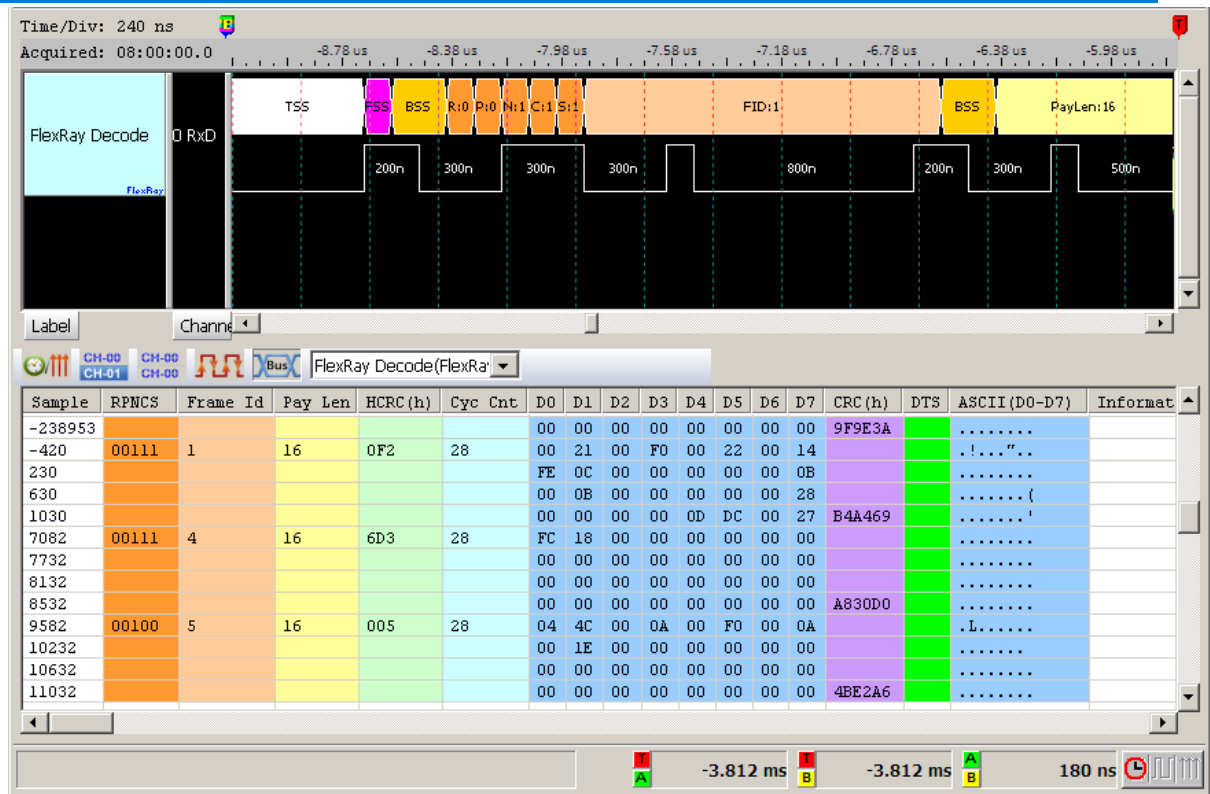
**Abbreviations are:**

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

Channel: CH 0 I/O 0: CH 3 Direction: ☒ SDI ☐ SDO

Color

Color: Stream Data Preamble Length Stream ID Sample Response (SDI) Valid Reserved UnSol Response Command (SDO) Reserved NID Payload CAd Verb ID

Range

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

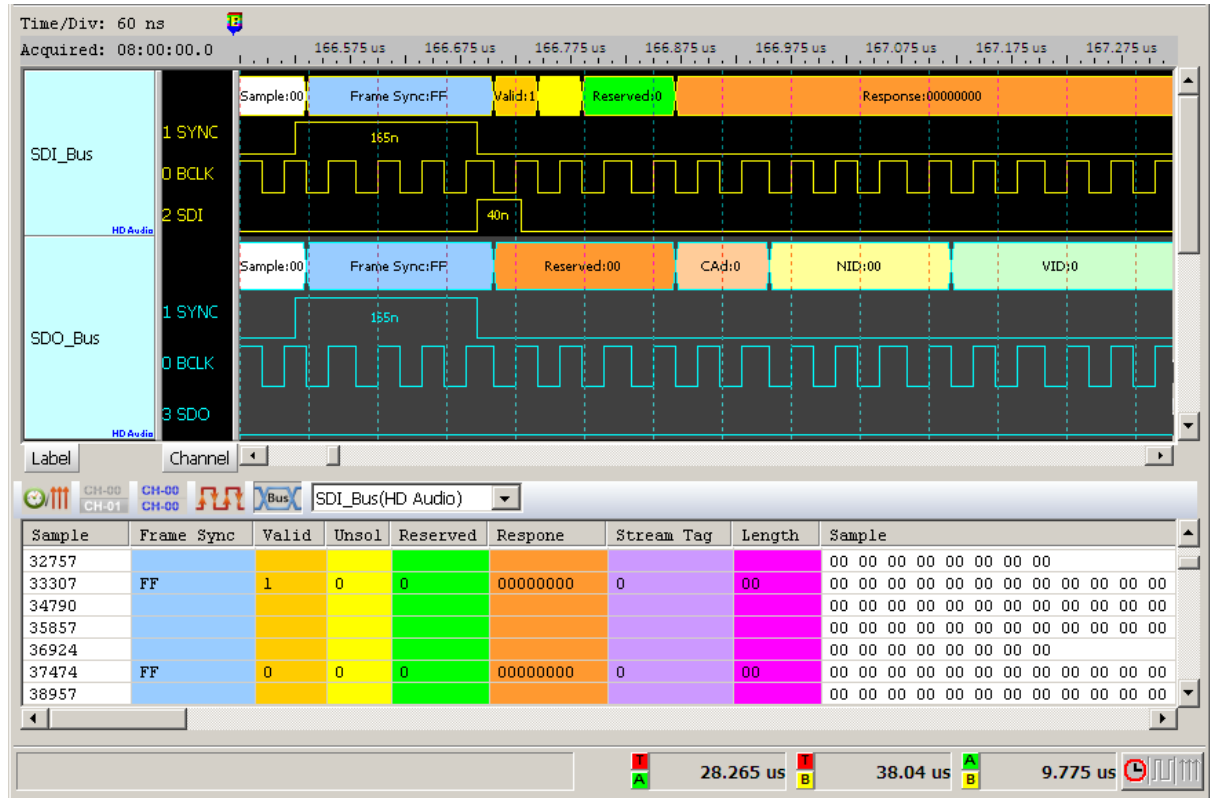
Default OK Cancel

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

**Direction:** Show the data with SDI or SDO decoding.

## Result

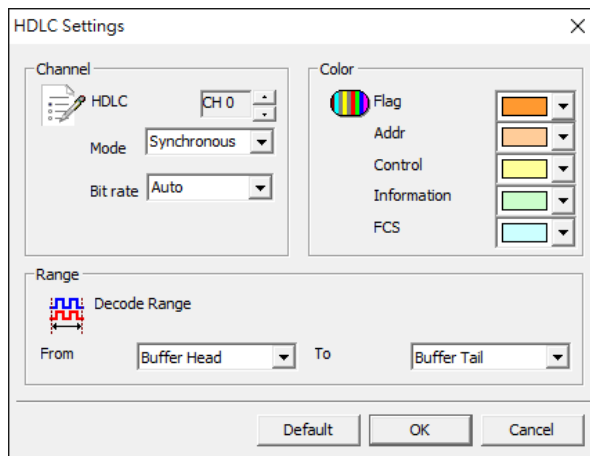
Click **OK** to run the HD Audio decode and see the result on the Waveform Window below.



## HDLC

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

### Settings

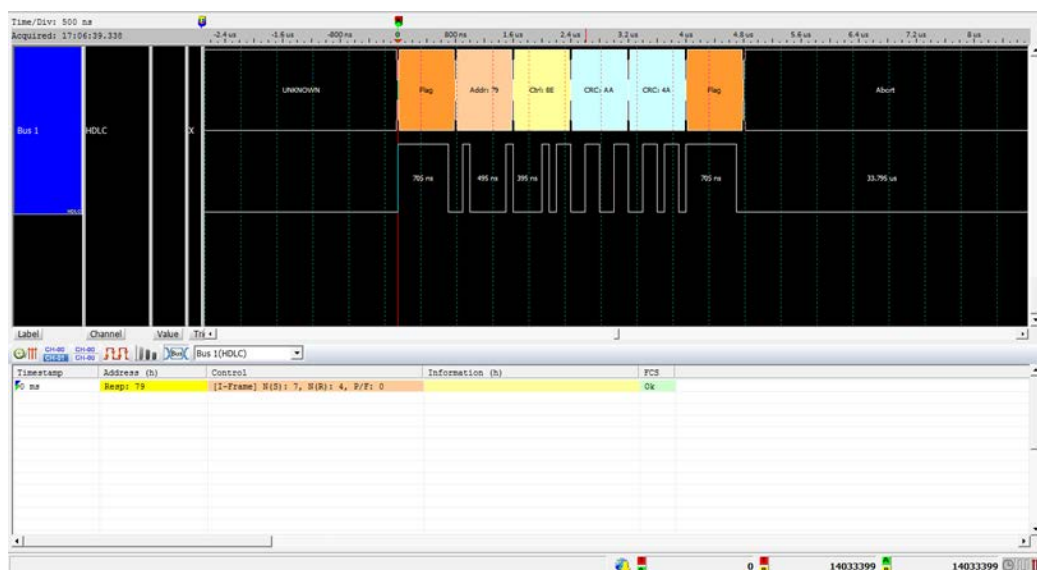


**HDLC:** Set the channel of the signal.

**Mode:** Synchronous or Asynchronous mode.

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

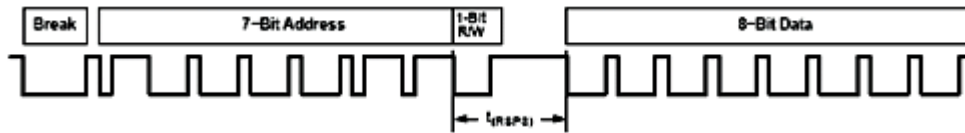
### Result



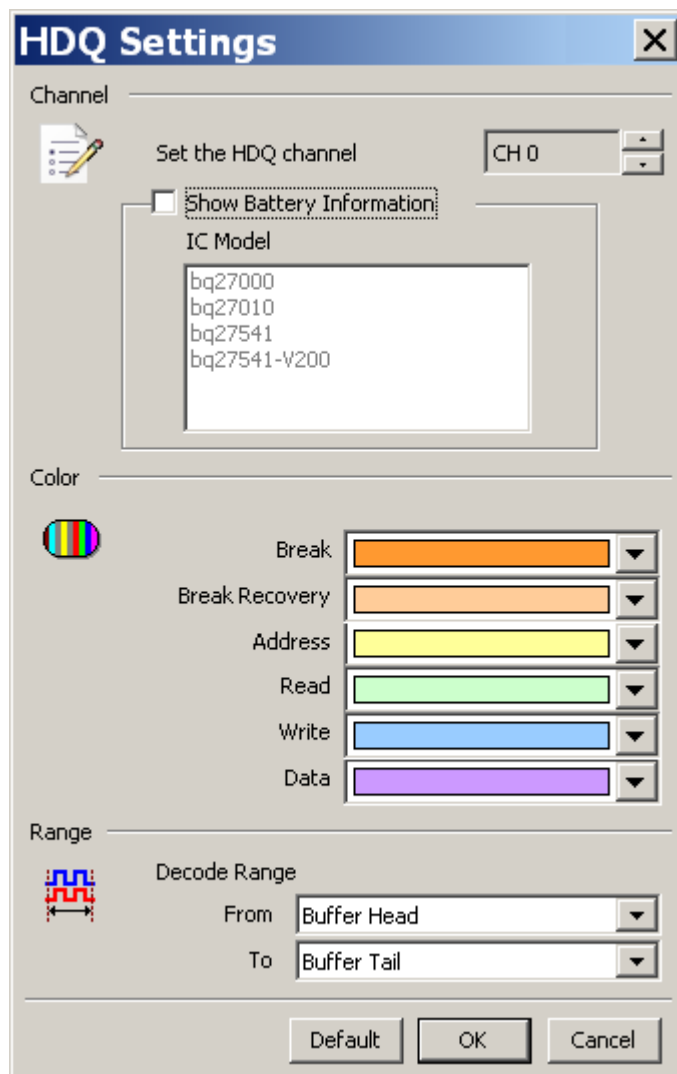


## HDQ

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



## Settings

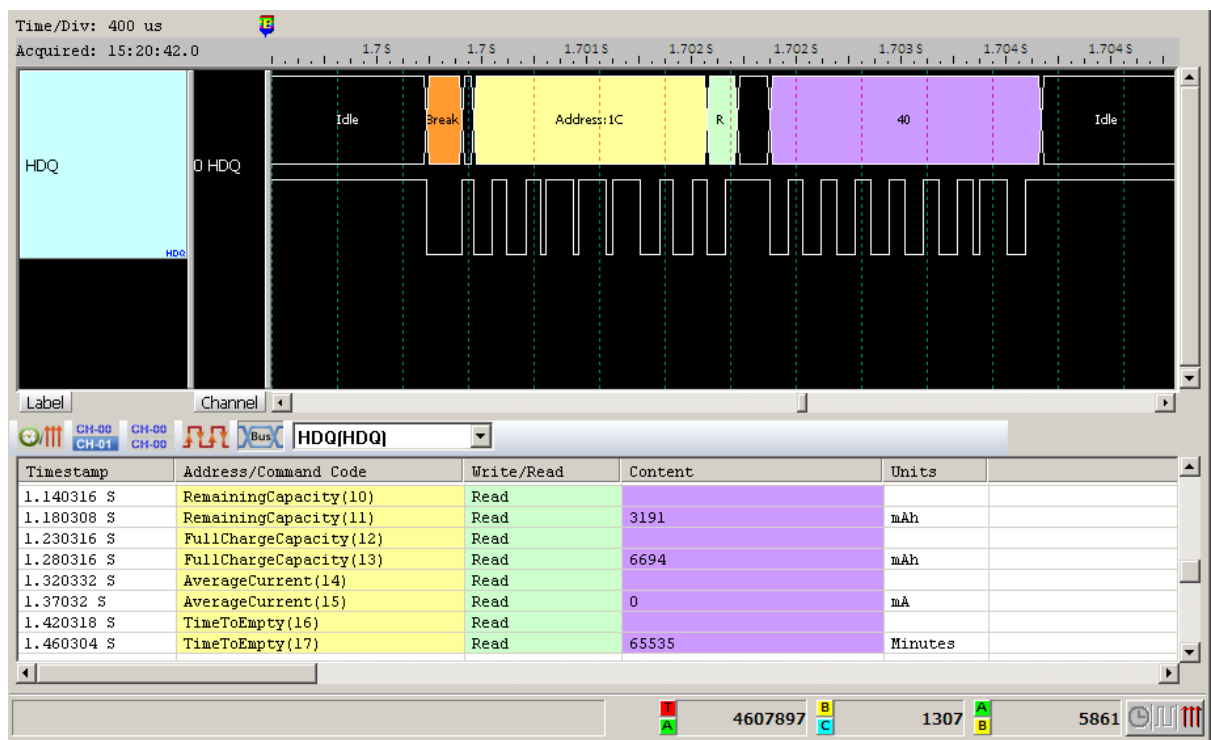
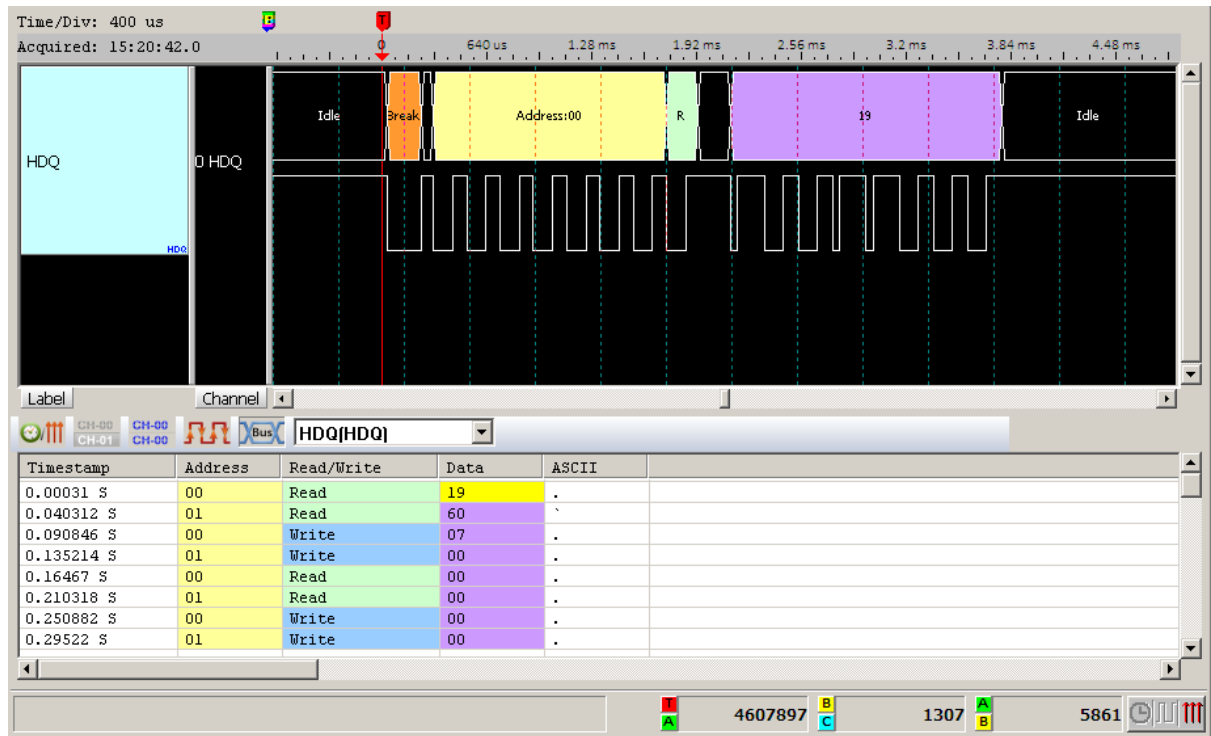


**Channel:** Set the HDQ channel: Show the selected channel (CH 0).

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

## Result

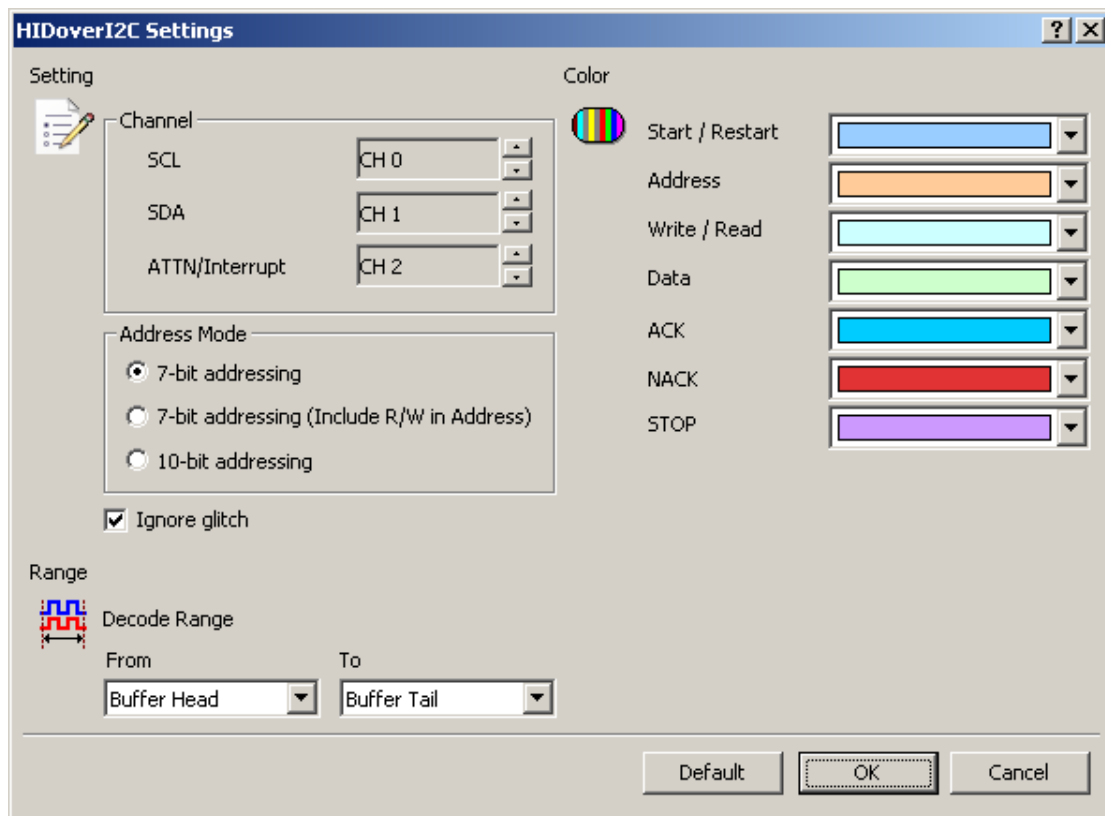
Click **OK** to run the HDQ decode and see the result on the Waveform Window below.



## HID Over I<sup>2</sup>C

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).

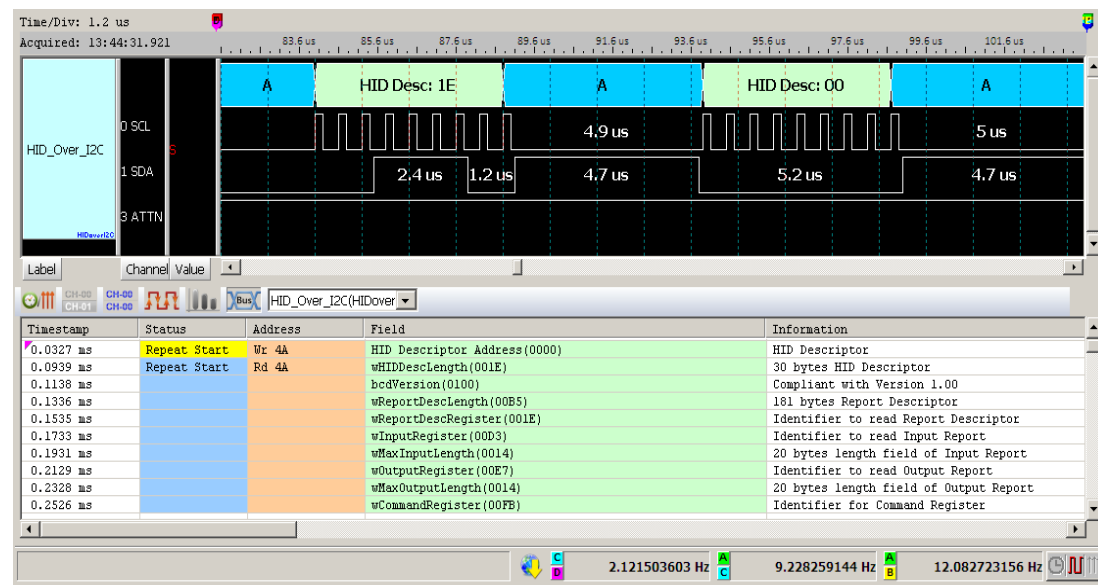
**7-bit addressing:** Show 7-bit addressing.

**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.

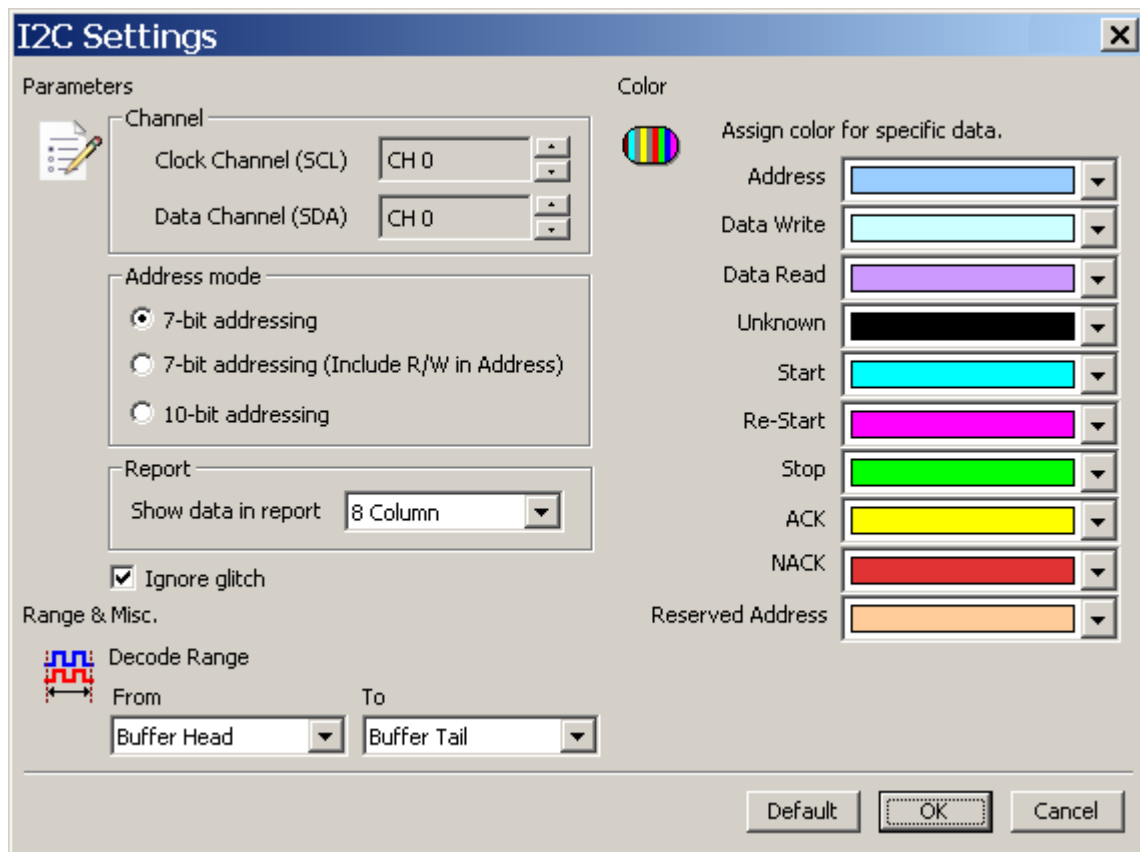
## Result



## I<sup>2</sup>C

The Inter-Integrated Circuit (I<sup>2</sup>C) bus has two data bits: Serial Data (SDA) and Serial Clock (SCL).

### Settings



The I2C Settings dialog box is divided into two main sections: Parameters and Color. The Parameters section includes fields for Clock Channel (SCL) and Data Channel (SDA), both set to CH 0. It also has radio buttons for Address mode: 7-bit addressing (selected), 7-bit addressing (Include R/W in Address), and 10-bit addressing. A Report section shows 'Show data in report' set to 8 Column. There is a checkbox for 'Ignore glitch' which is checked. The Range & Misc. section includes a 'Decode Range' section with 'From' and 'To' dropdowns, both set to 'Buffer Head' and 'Buffer Tail' respectively. The Color section, titled 'Assign color for specific data.', contains a color palette icon and a list of data types with corresponding color dropdowns: Address (light blue), Data Write (light cyan), Data Read (light purple), Unknown (black), Start (cyan), Re-Start (magenta), Stop (green), ACK (yellow), NACK (red), and Reserved Address (orange). At the bottom are buttons for Default, OK, and Cancel.

**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.

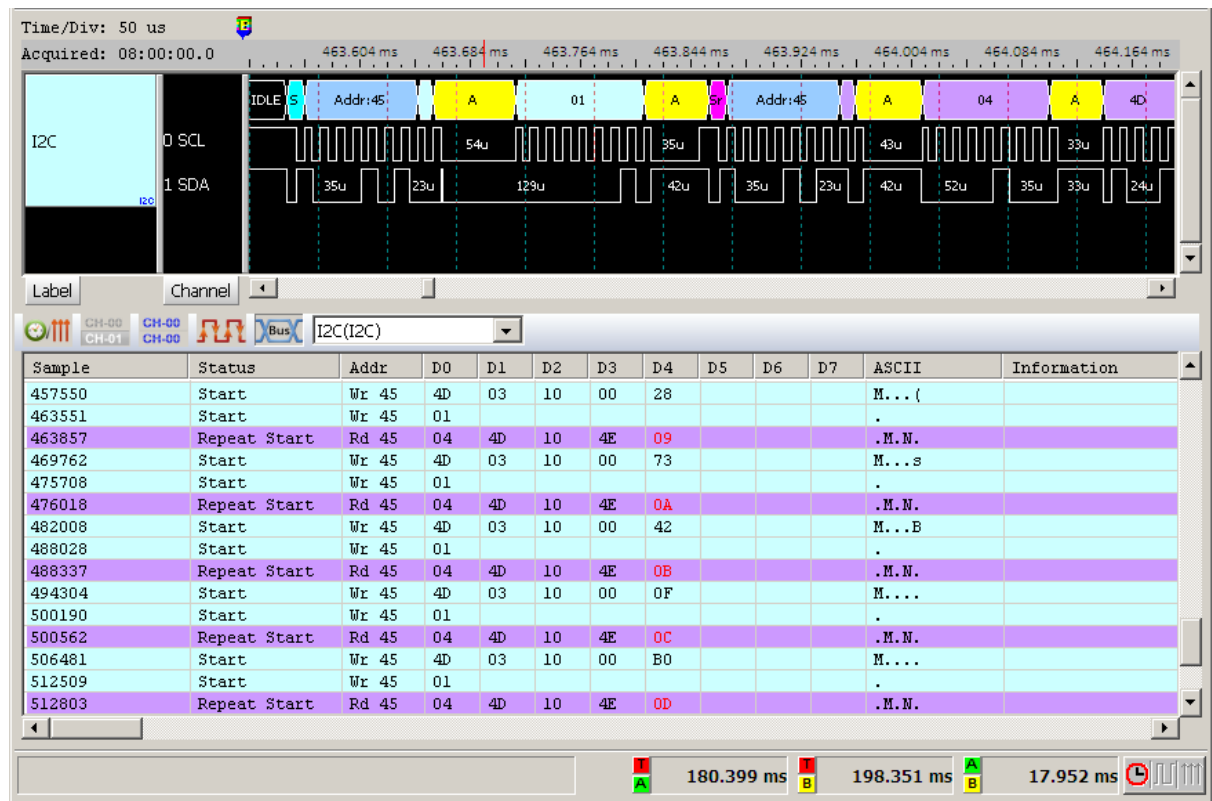
**Report:** Show either 8 or 16-columns data in the report window.

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

## Result

Click **OK** to run the I<sup>2</sup>C decode and see the result on the Waveform Window below.

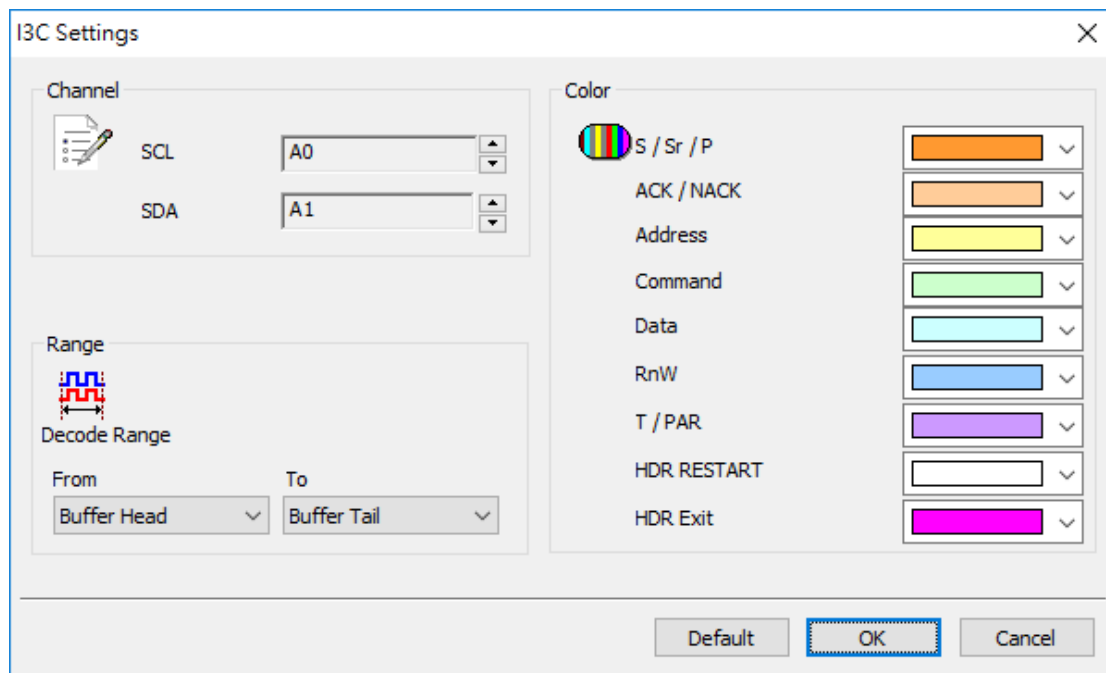
It shows frequency in the Information field.



## I3C

I<sup>3</sup>C 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

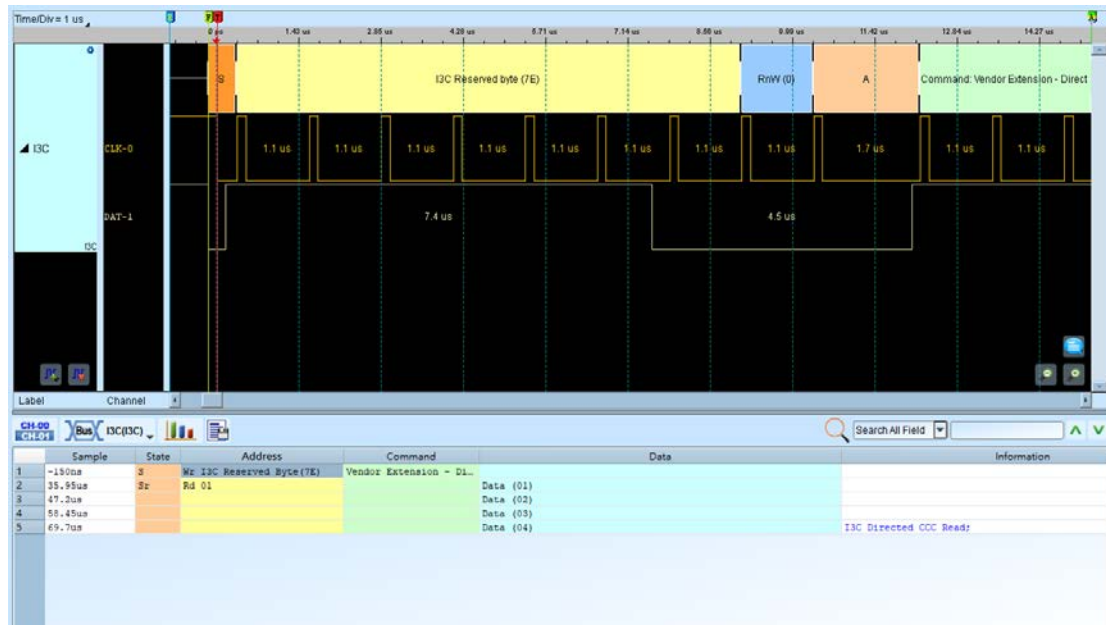


**Channel:** Display the channels (CH0 and CH1).

## Result

Click **OK** to run the I<sup>2</sup>C decode and see the result on the Waveform Window below.

It shows frequency in the Information field.







## I<sup>2</sup>C 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).

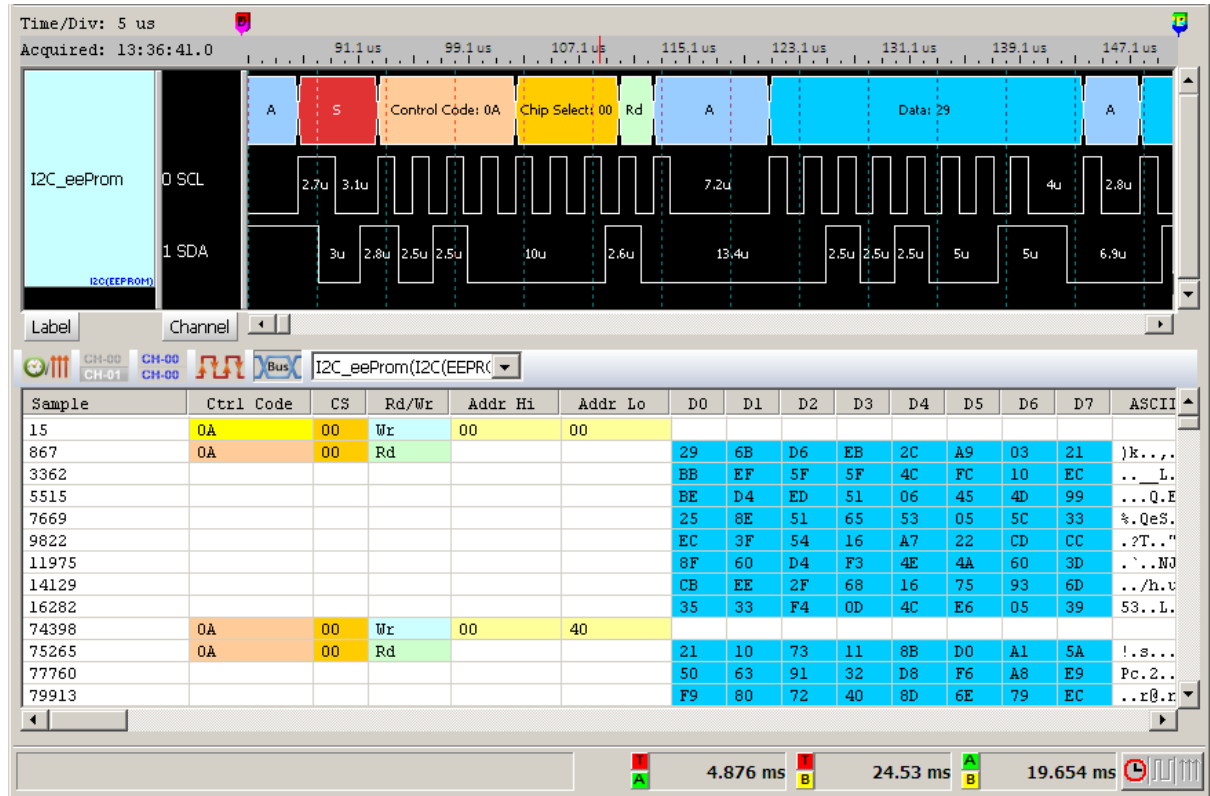
**24LCS61/24LCS62:** 24LCS61/24LCS62 EEPROM protocol.

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

## Result

Click OK to run the I<sup>2</sup>C EEPROM decode and see the result on the Waveform Window below.

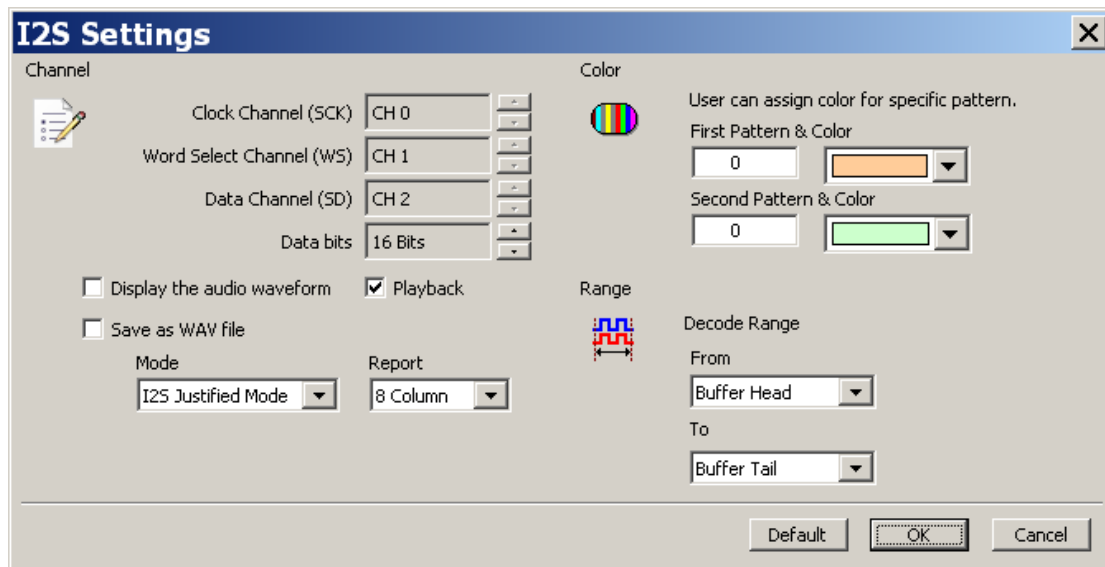
Window below.



## I<sup>2</sup>S

The Inter-Integrated Circuit Sound (I<sup>2</sup>S) 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..

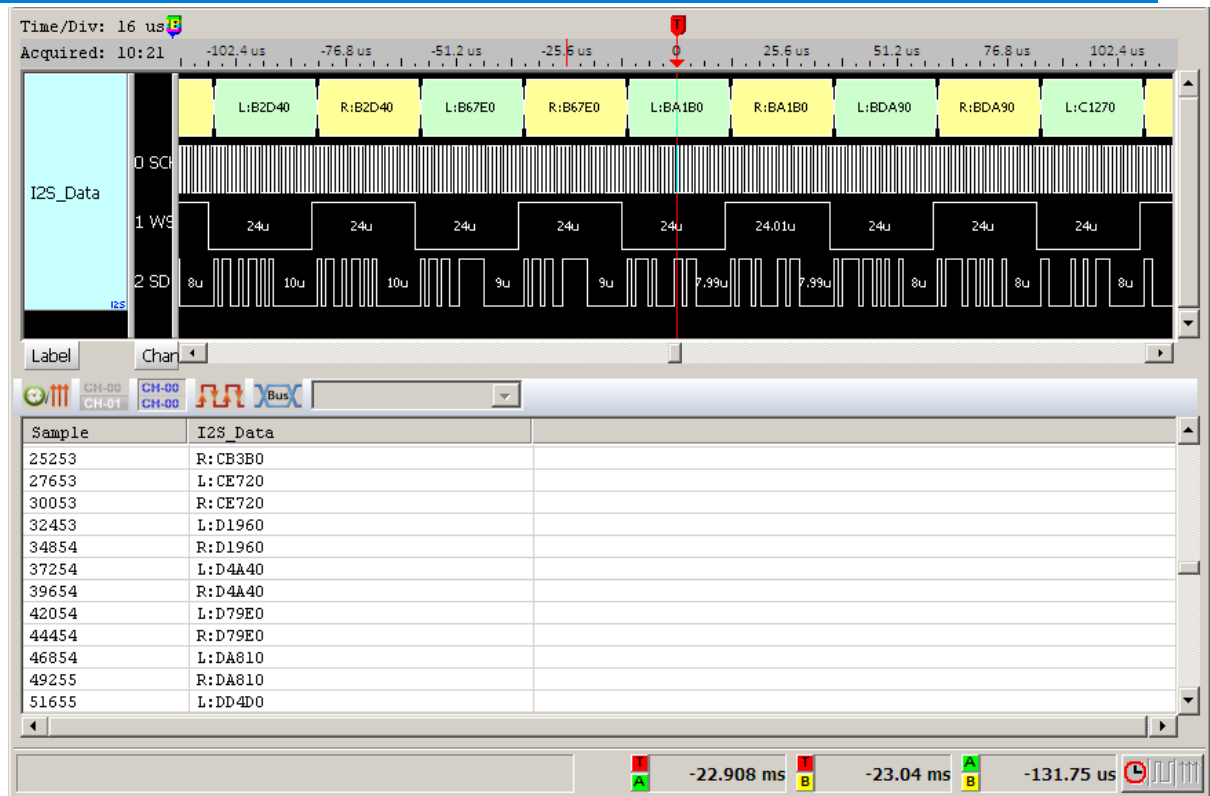
**Save as WAV file:** Click to save the data to a audio file(.wav) in the work directory.

**Mode:** I<sup>2</sup>S Justified/MSB Justified/LSB Justified/PCM/TDM.

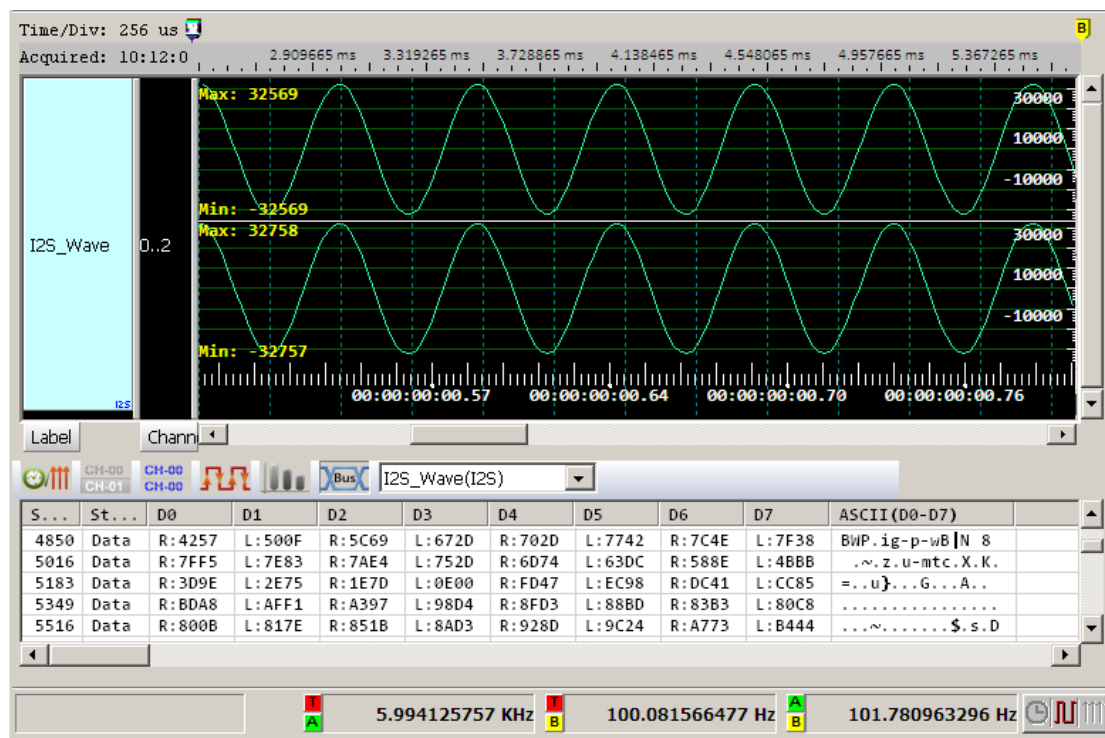
**Report:** Select the column number.

### Result

Click **OK** to run I<sup>2</sup>S decode and see the result on the Waveform Window below.



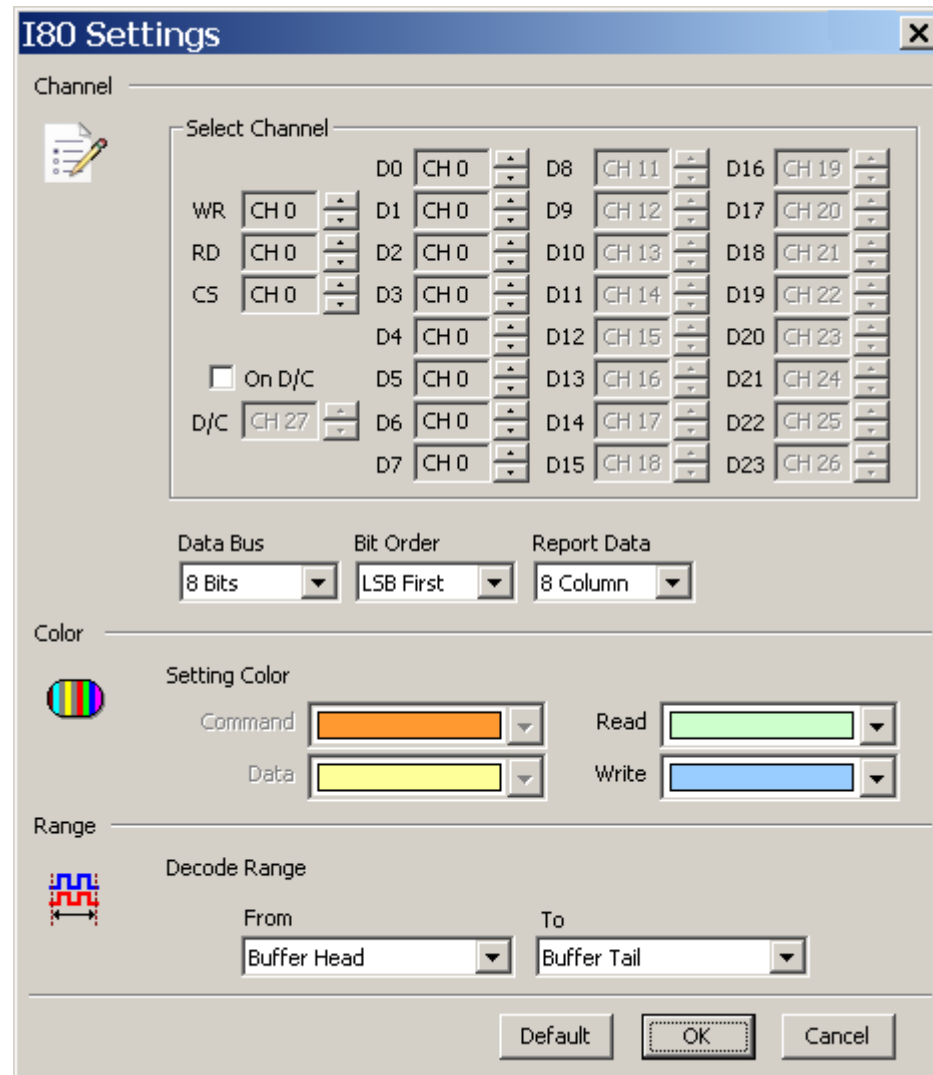
Display the audio waveform



## I80

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

### Settings



The I80 Settings dialog box is divided into several sections:

- Channel:** Contains a 'Select Channel' section with a grid of 24 channels (D0-D23) and control lines (WR, RD, CS, D/C). Each channel has a dropdown menu set to 'CH 0'. The D/C line is set to 'CH 27'. There is an 'On D/C' checkbox which is currently unchecked.
- Data Bus:** A dropdown menu set to '8 Bits'.
- Bit Order:** A dropdown menu set to 'LSB First'.
- Report Data:** A dropdown menu set to '8 Column'.
- Color:** Contains a 'Setting Color' section with color pickers for 'Command' (orange), 'Data' (yellow), 'Read' (light green), and 'Write' (light blue).
- Range:** Contains a 'Decode Range' section with 'From' and 'To' dropdown menus. 'From' is set to 'Buffer Head' and 'To' is set to 'Buffer Tail'.

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

**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).

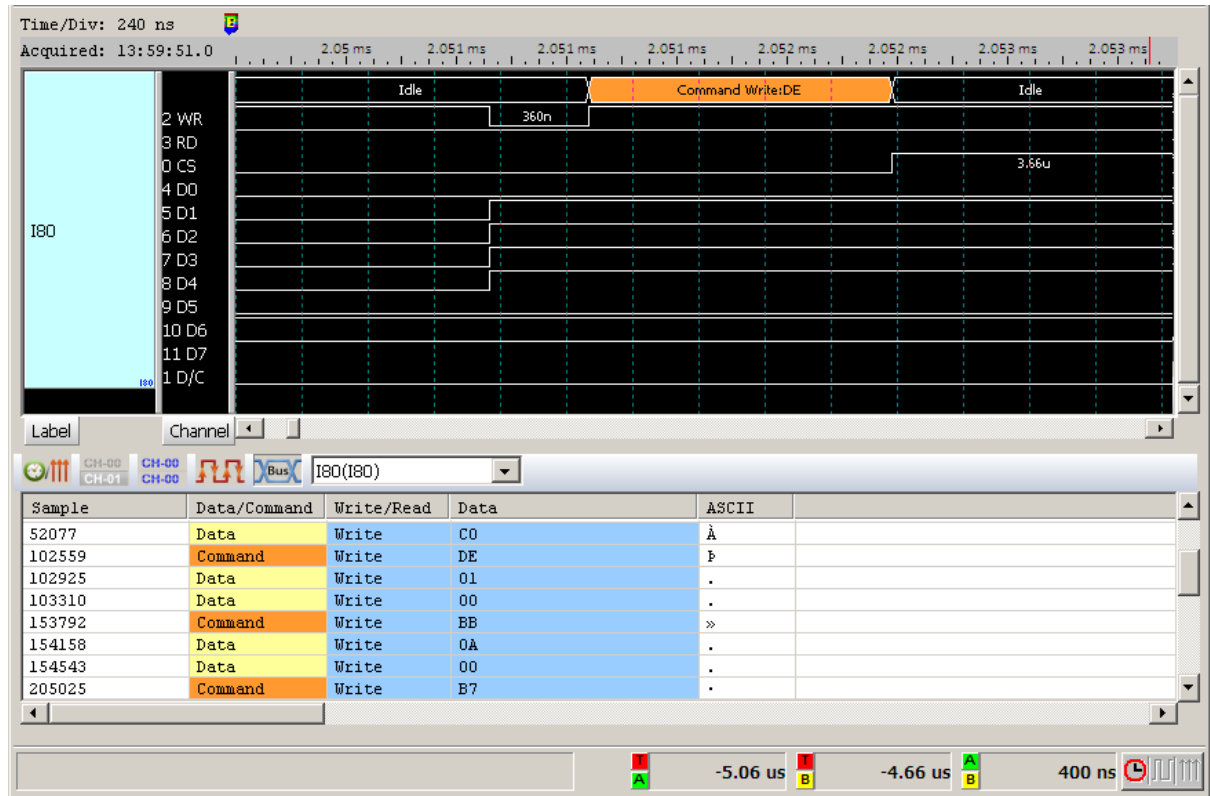
**Data Bus:** Select 4 Bit, 8 Bit, 12 Bit, 16 Bit, 20 Bit, or 24 Bit.

**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

**IDE Settings**

Channel

General Register Data Bus

DIOR-:HDMARDY-:HSTROBE CH 19 PDIAG-:CBLID- CH 26

DIOW-:STOP CH 18 DASP- CH 31

DMARQ CH 17 RESET- CH 0

IORDY-:DDMARDY-:DSTROBE CH 20 CSEL CH 21

DMACK- CH 22 IOCS16 CH 24

INTRQ CH 23

Color and Setting

Transferring Mode Register Color Analysis Report

Transferring Mode	Max Transferring Rate	Standard
<input type="checkbox"/> DMA Single word, Mode 2	8.33MByte/sec	ATA
<input type="checkbox"/> DMA Multiple word, Mod...	4.17MByte/sec	ATA
<input type="checkbox"/> DMA Multiple word, Mod...	13.3MByte/sec	ATA-2
<input type="checkbox"/> DMA Multiple word, Mod...	16.7MByte/sec	ATA-3
<input type="checkbox"/> ULTRA DMA Mode 0	16.6MByte/sec	ATA-4
<input type="checkbox"/> ULTRA DMA Mode 1	25MByte/sec	ATA-4
<input type="checkbox"/> ULTRA DMA Mode 2	33MByte/sec	ATA-4
<input type="checkbox"/> ULTRA DMA Mode 3	44MByte/sec	ATA-5
<input type="checkbox"/> ULTRA DMA Mode 4	66MByte/sec	ATA-5
<input type="checkbox"/> ULTRA DMA Mode 5	100MByte/sec	ATA-6
<input checked="" type="checkbox"/> ULTRA DMA Mode 6	133MByte/sec	ATA-7

Range

Decode Range

From To

Buffer Head Buffer Tail

Default OK Cancel

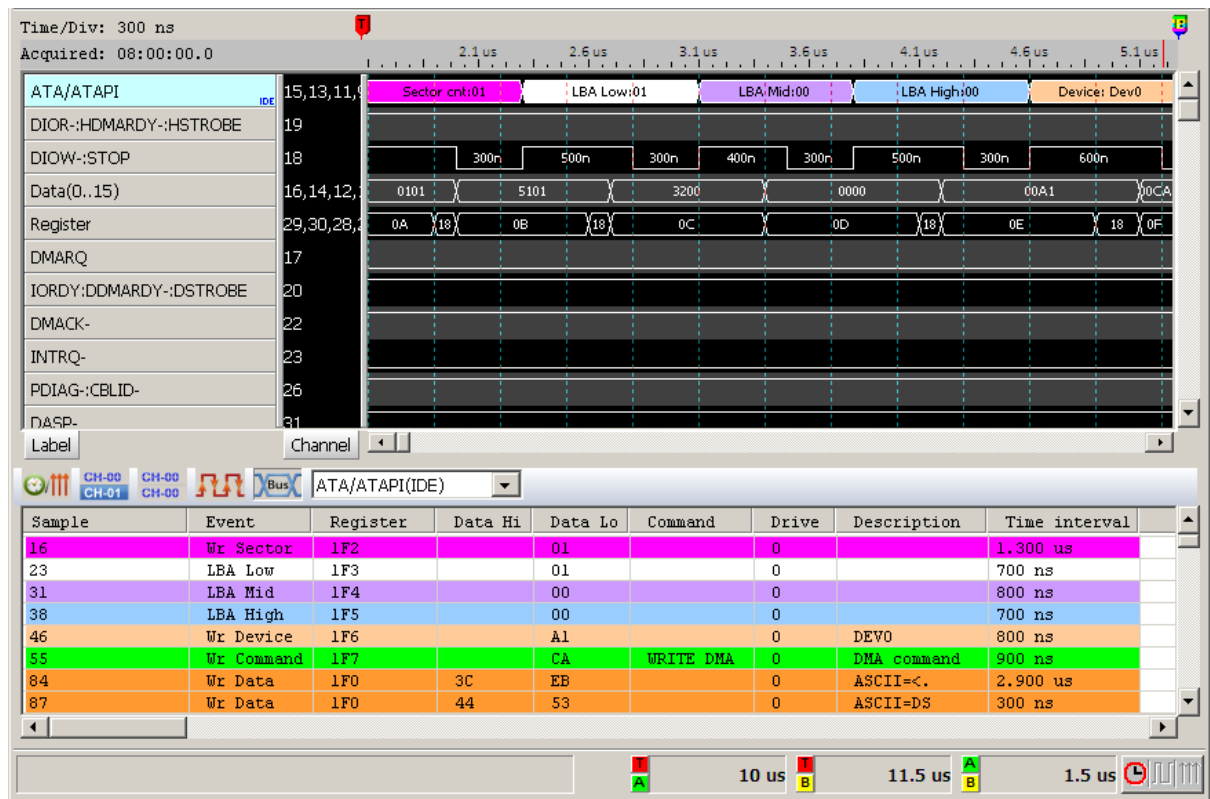
**Channel:** Set channel number for General, Register, and Data Bus.

**Transferring Mode:** Select the target system.

**Analysis Report:** Filter the data in the Report Window.

## Result

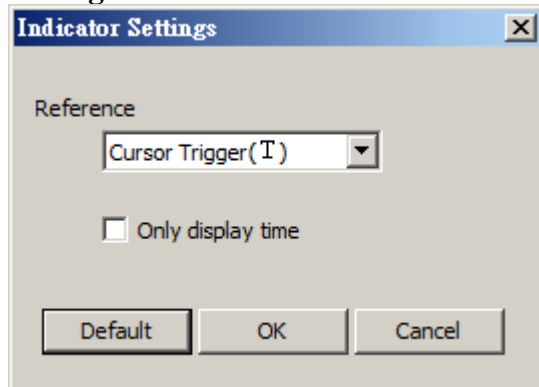
Click **OK** to run the IDE decode and see the result on the Waveform Window below.



## Indicator

Indicator is to display with the time or clock samples on the Waveform window to be helpful to read the waveform.

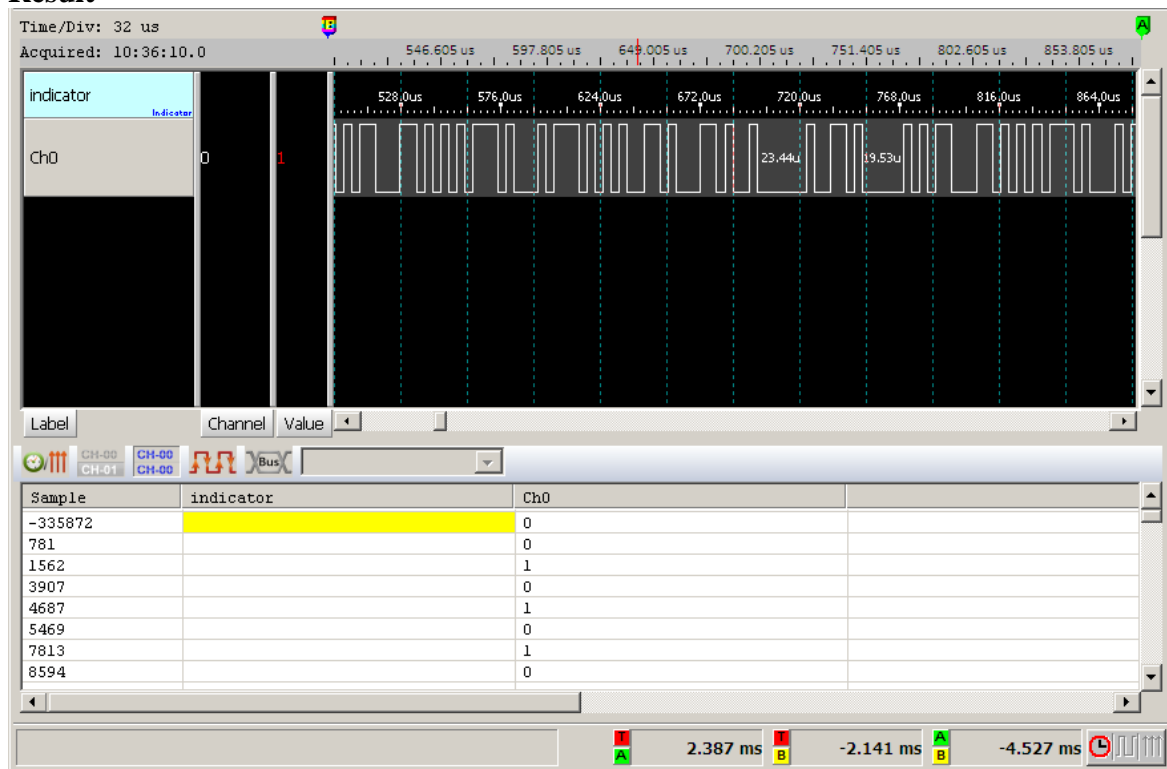
### Settings



**Reference:** Set the reference position (Trigger cursor), time value on the right (left) is positive (negative). Default reference position is the trigger cursor and also shows time or sample rate.

**Only display time:** to show time only.

### Result

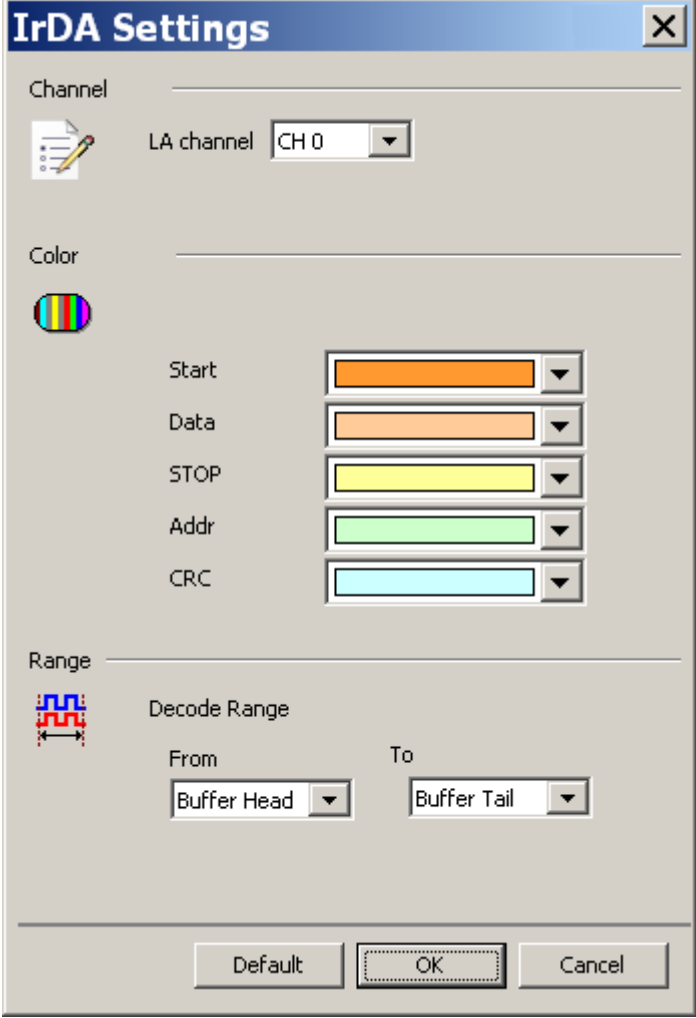


Add two Indicators for two channels with two different cursors as the references to display the result on the Waveform Window below.

## 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

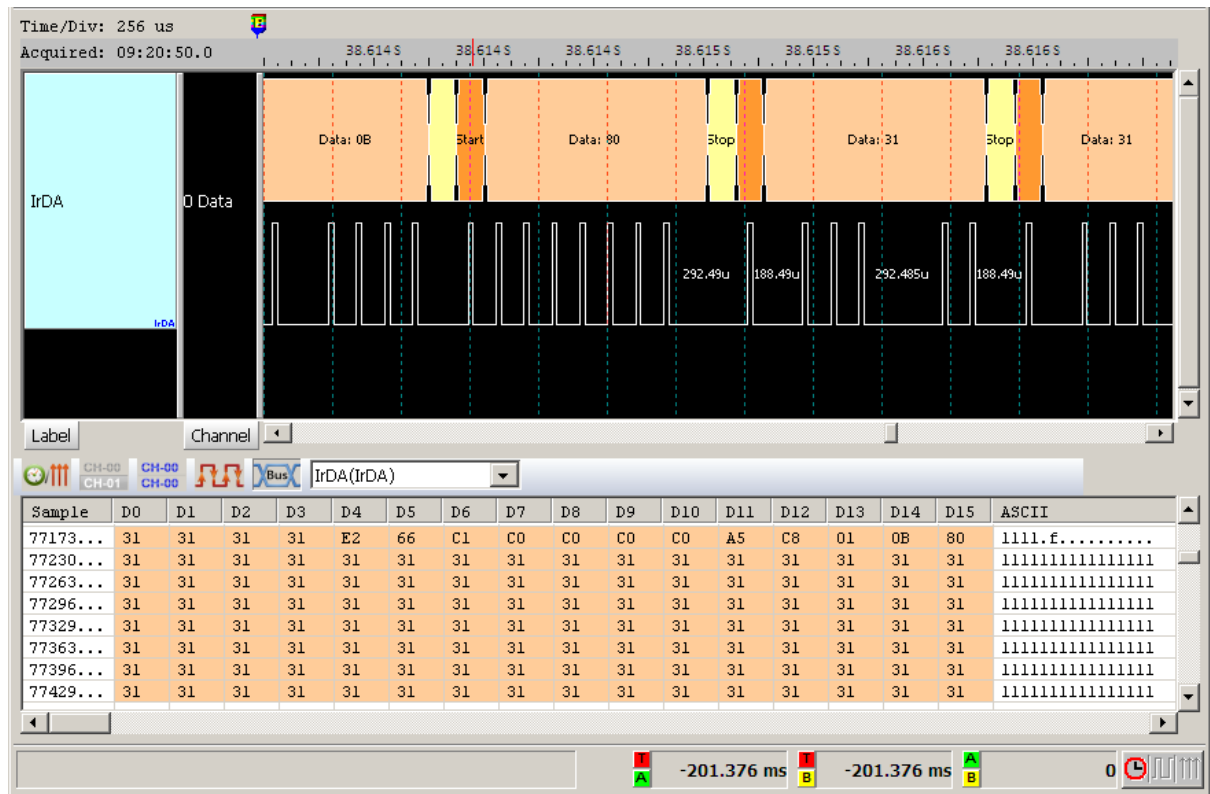


The **IrDA Settings** dialog box is used to configure IrDA parameters. It features three main sections: Channel, Color, and Range. The Channel section includes a list icon and a dropdown for 'LA channel' set to 'CH 0'. The Color section includes a color bar icon and five color selection boxes for 'Start' (orange), 'Data' (light orange), 'STOP' (yellow), 'Addr' (light green), and 'CRC' (light blue). The Range section includes a waveform icon and a 'Decode Range' section with 'From' and 'To' dropdowns, both set to 'Buffer Head' and 'Buffer Tail' respectively. At the bottom are 'Default', 'OK', and 'Cancel' buttons.

Section	Parameter	Value
Channel	LA channel	CH 0
	Color	
Color	Start	Orange
	Data	Light Orange
	STOP	Yellow
	Addr	Light Green
	CRC	Light Blue
Range	Decode Range	
	From / To	Buffer Head / Buffer Tail

**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

Clk	CH 0	Data 5	CH 6
Data 0	CH 1	Data 6	CH 7
Data 1	CH 2	Data 7	CH 8
Data 2	CH 3	Data 8	CH 9
Data 3	CH 4	Data 9	CH 10
Data 4	CH 5		

Data Bits: 8 Bits

**Color**

Setting transmitter's color

SAV: [Color Bar] CR: [Color Bar]

EAV: [Color Bar] CB: [Color Bar]

Blanking: [Color Bar] Y: [Color Bar]

**Range**

Decode Range

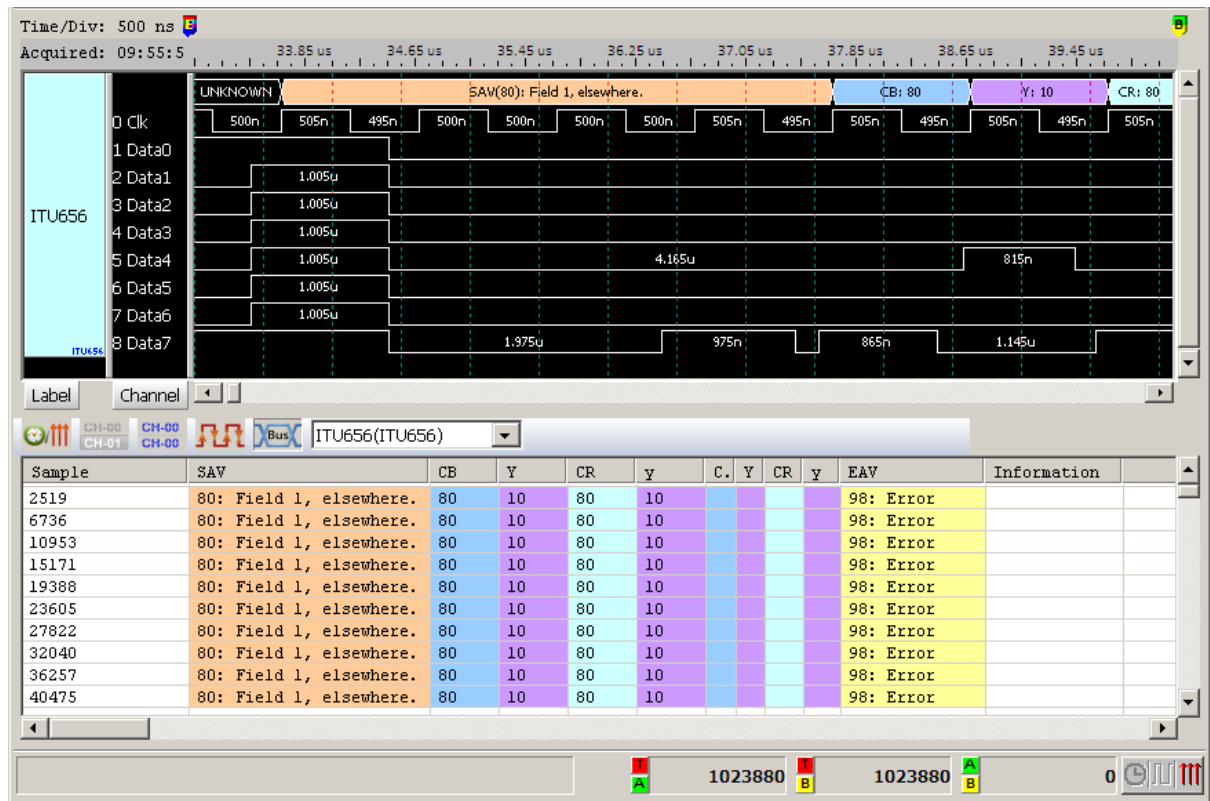
From: Buffer Head To: Buffer Tail

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.

### Settings

**JTAG Settings**

Setting

Channel | Setting | Report

Test Clock (TCK) CH 0

Test Mode Select (TMS) CH 1

Test Data Input (TDI) CH 2

Test Data Output (TDO) CH 3

Test Reset (TREST) ☒ CH 4

Color

TEST\_LOGIC\_RESET

RUN\_TEST\_IDLE

SELECT-IR

SELECT-DR

CAPTURE-IR

CAPTURE-DR

SHIFT-IR

SHIFT-DR

EXIT1-IR

EXIT1-DR

PAUSE-IR

PAUSE-DR

EXIT2-IR

EXIT2-DR

UPDATE-IR

UPDATE-DR

Range

Decode Range

From Buffer Head To Buffer Tail

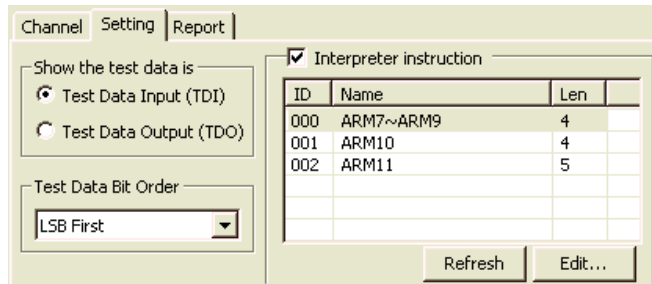
Default Ok Cancel

**Settings:** Includes Channel, Setting, and Report.

**Channel:** Set the channel number.

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

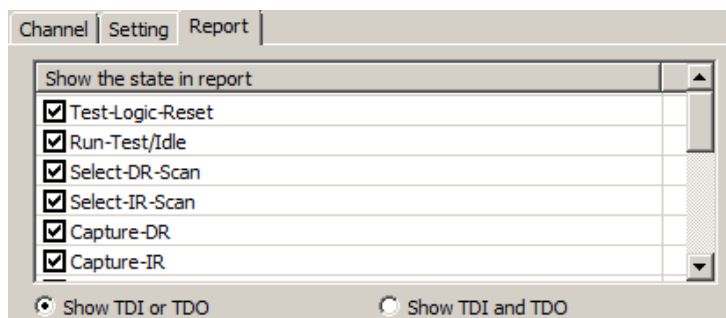
**Test Data Bit Order:** LSB First or MSB First.



**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.

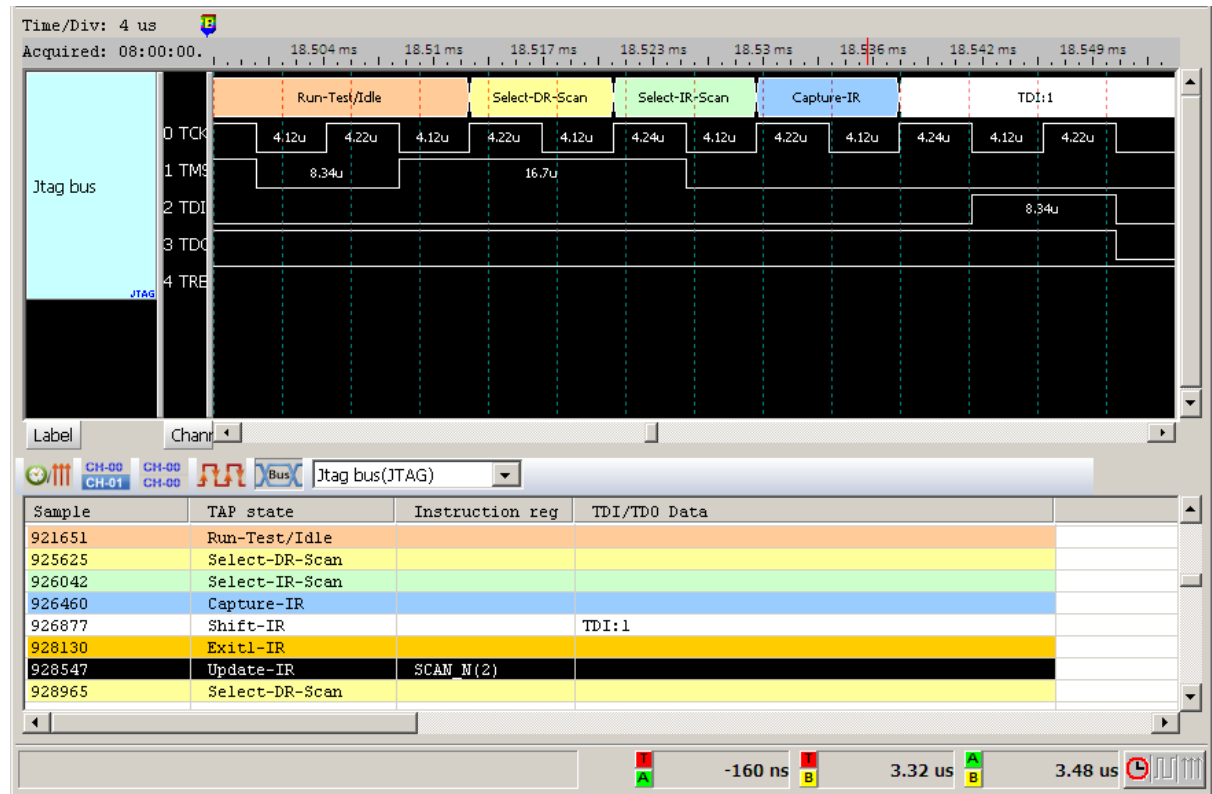
**Report:** You can filter the data you want to see on the Report Window.



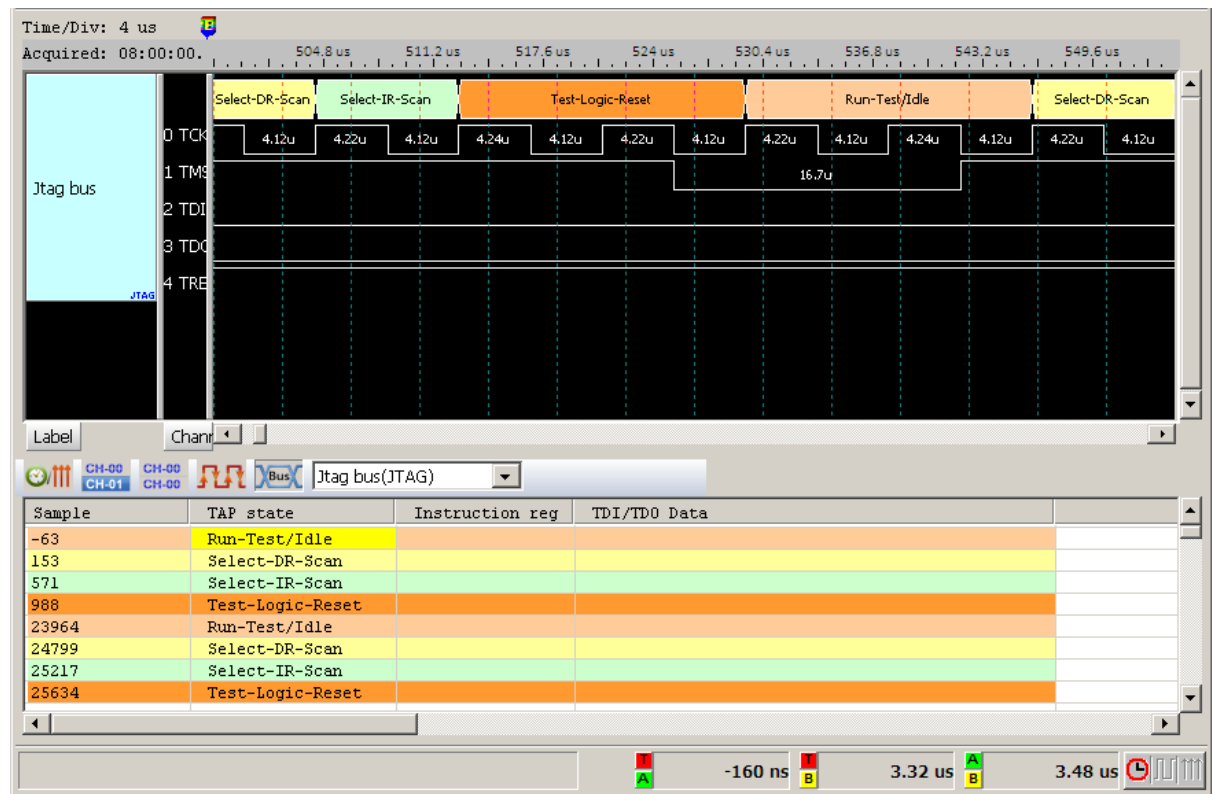
**Show TDI or/and TDO**

## Result

The Altera EPM3256AT144 Programming Schematic JTAG decode on the Waveform Window



## The ARM7 Read IDCODE JTAG decode on the Waveform Window



### 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

**LCD1602 Settings**

**Channel**

Channel

RS	CH 0	DB7	CH 3	DB3	CH 7
RW	CH 1	DB6	CH 4	DB2	CH 8
E	CH 2	DB5	CH 5	DB1	CH 9
		DB4	CH 6	DB0	CH 10

Data Mode

☒ 8 Lines ☐ 4 Lines ☒ To merge the same command

**Color**

Setting Commands Color

SCREEN CLEAR	[Yellow]	CGRAM AD SET	[Red]
CURSOR RETURN	[Cyan]	DDRAM AD SET	[Orange]
INPUT SET	[Purple]	FUNCTION SET	[Light Blue]
DISPLAY SWITCH	[Magenta]	DATA WRITE	[Blue]
SHIFT	[Yellow]	DATA READ	[Orange]
BUSY/AD READ CT	[Light Green]		

**Range**

Decode Range

From: Buffer Head To: Buffer Tail

Default OK Cancel

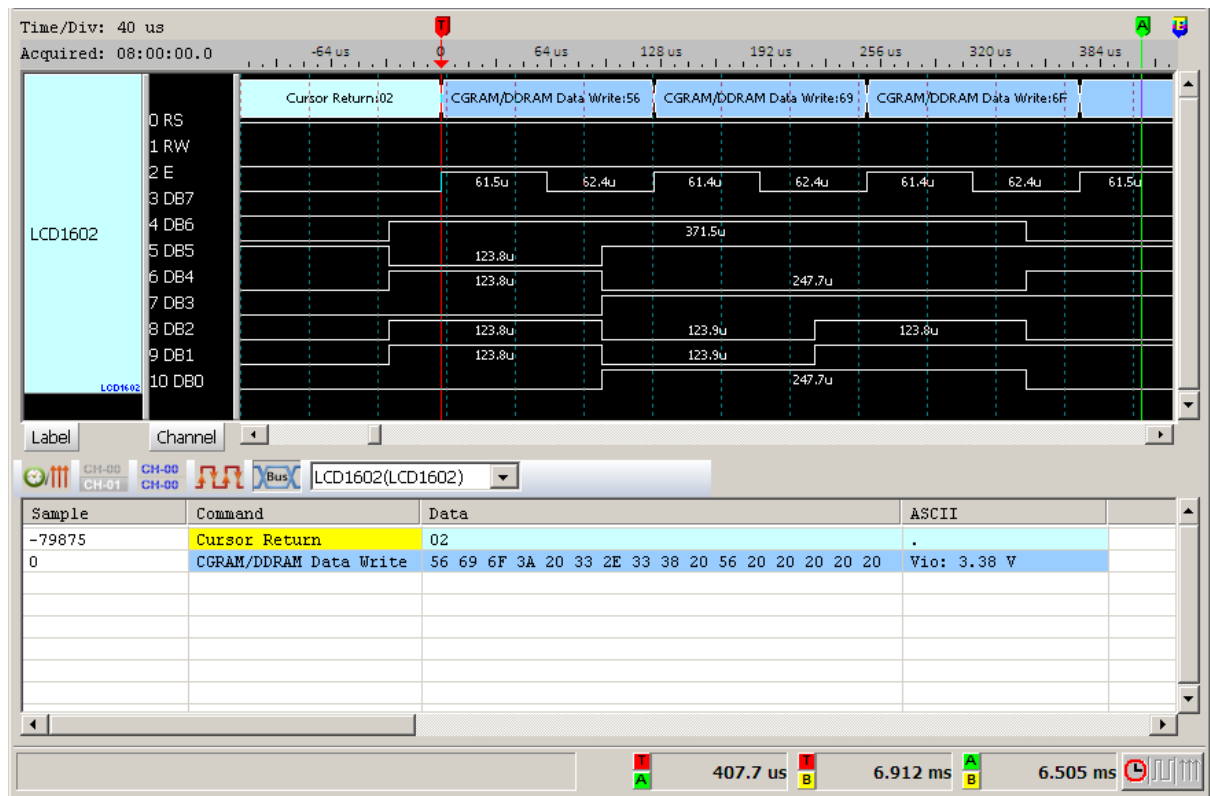
**Channel:** Show the selected channels (RS: CH0, RW: CH1,..., DB0: CH10).

**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.



## LIN

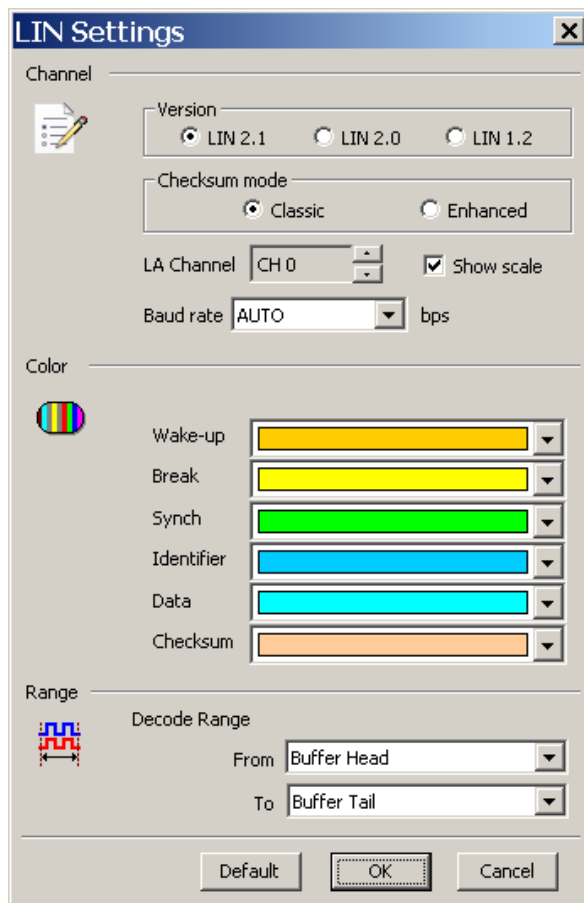
The Local Interconnect Network (LIN) bus (version 2.1) has two message types:

Header and Response.

- (1) Header contains three data frames: Synchronization Break (Break), Synchronization Field (Sync) and Identifier Field (Identifier).
- (2) Response contains two data frames: Data Field (Data) and Checksum Field (Checksum). Checksum contains data and identifier (Enhanced mode), but version 1.3 or below (Classic mode) contains data only.

LIN bus has two states - Sleep mode and Active mode. While data is on the bus, all LIN nodes are in active state; but after a specified timeout, the nodes enter Sleep mode and will be released back to active state by a WAKEUP frame.

### Settings





**Version:** Select LIN version.

**Checksum Mode:** Select Classic: data only or Enhanced: data and identifier.

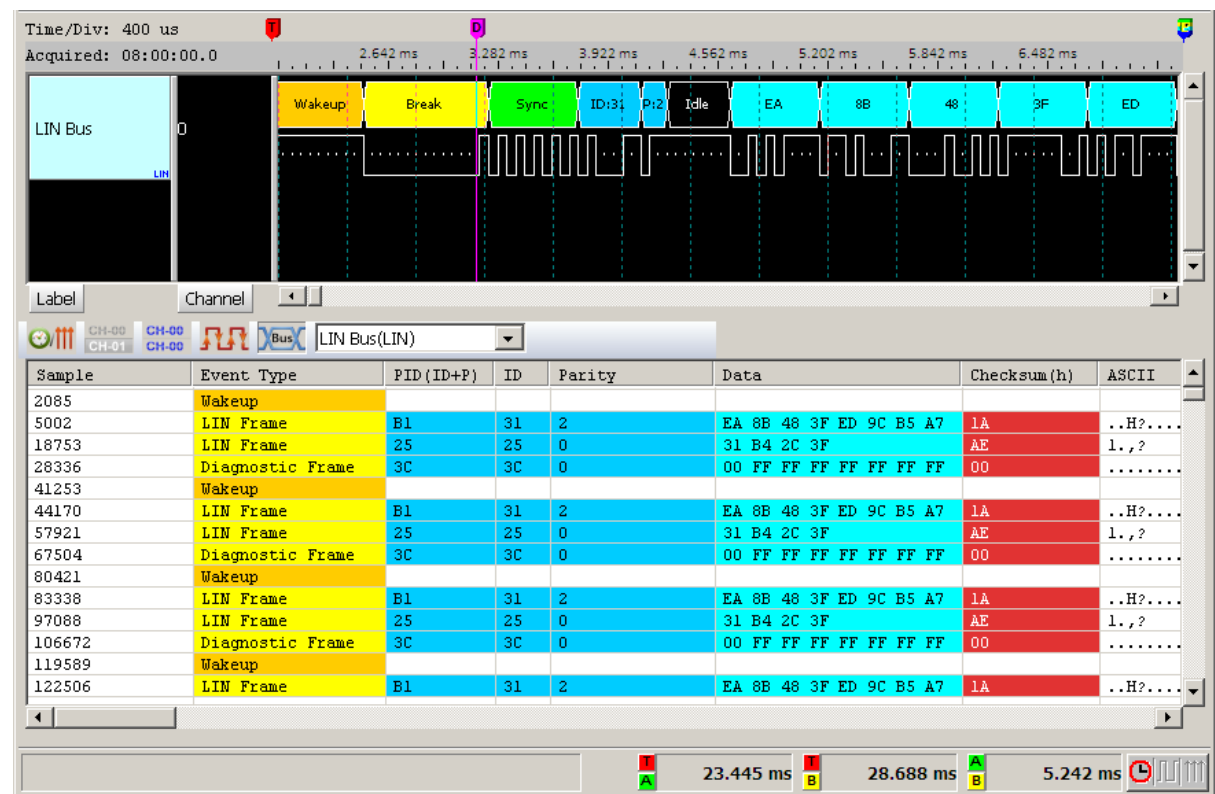
**LA Channel:** Show the selected channel (CH0 for LIN).

**Show Scale:** Show scale on the waveform.

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

## Result

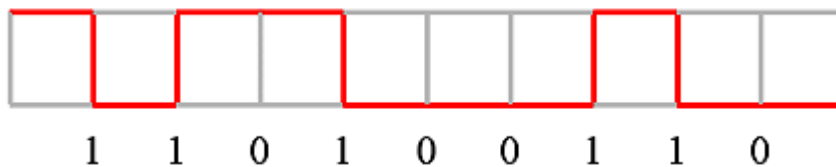
Click **OK** to run the Lin decode and see the result on the Waveform Window below.



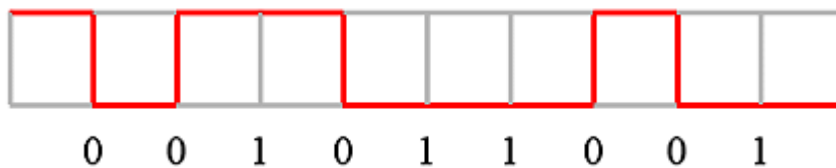
## Line Decoding

**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.

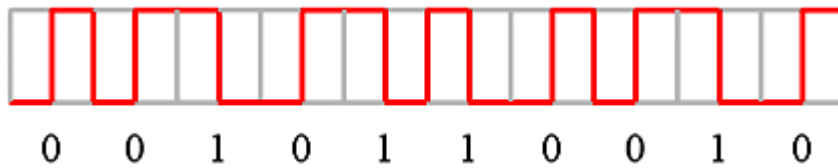


**NRZI (Transition occurs for a zero):** A 0 is represented by a transition of the physical level, a 1 has no transition.

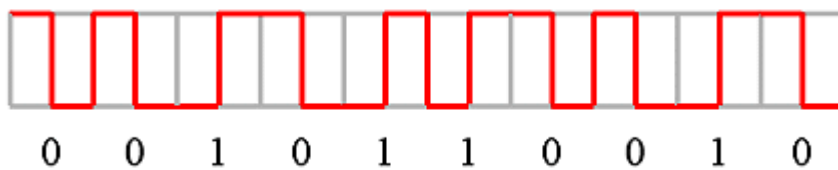


**Manchester:** In telecommunication, Manchester code is a line code in which the encoding of each data bit has at least one transition and occupies the same time. It is, therefore, self-clocking, which means that a clock signal can be recovered from the encoded data. There are three modes:

**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.

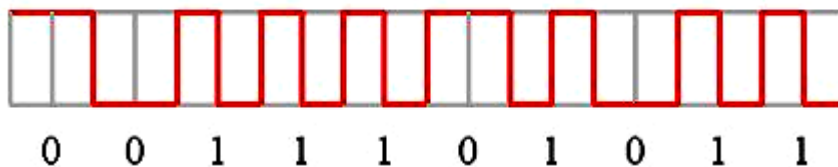


**Differential Manchester:** A 1 bit is indicated by making the first half of the signal equal to the last half of the previous bit's signal i.e. no transition at the start of the bit-time. A '0' bit is indicated by making the first half of the signal opposite to the last half of the previous bit's signal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



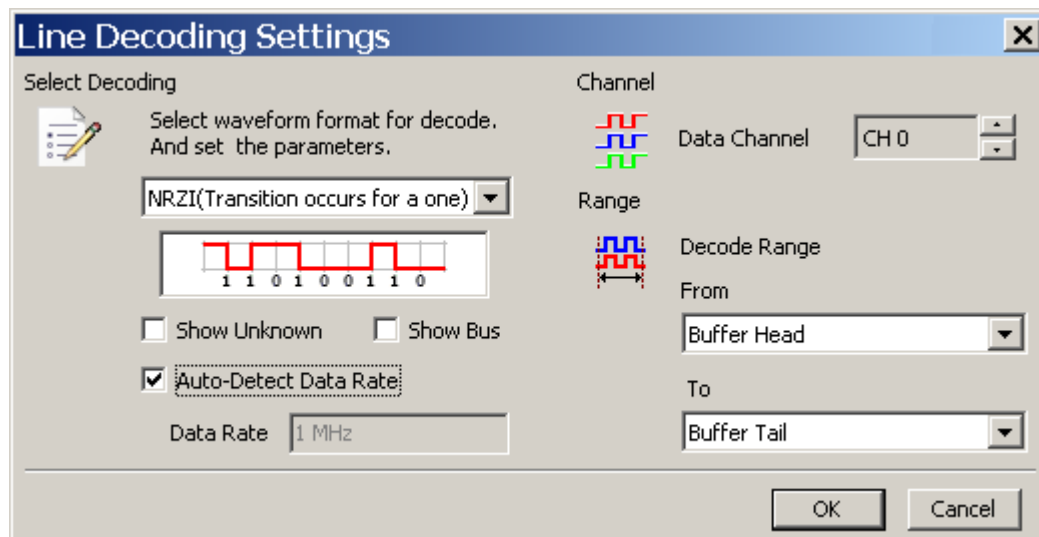
**Modified Miller:** The Modified Miller (M 2) demodulator facilitates demodulation of M 2 modulation data to NRZ-L (non-return-to-zero-level) data, composed of a data separation circuit for producing synchronizing clock pulses from the M 2 modulation data which is reproduced by a data recording device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

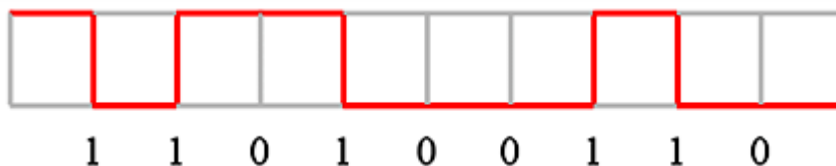
**Channel:** Show the selected channel (CH 0).

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

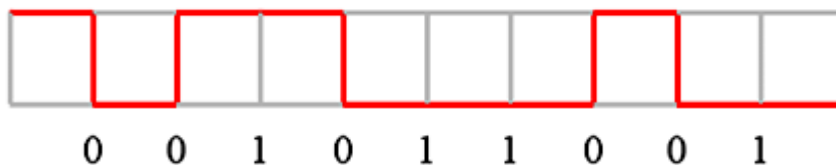
## Line Encoding

**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.

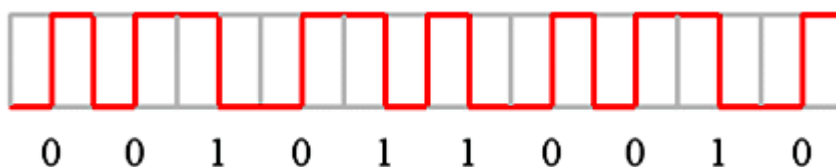


**NRZI (Transition occurs for a zero):** A 0 is represented by a transition of the physical level, a 1 has no transition.

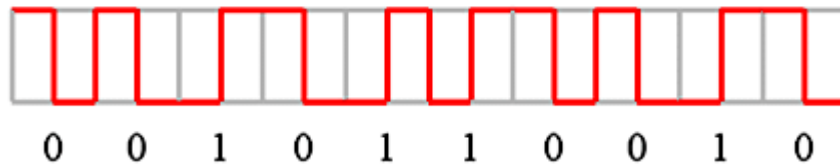


**Manchester:** In telecommunication, Manchester code is a line code in which the encoding of each data bit has at least one transition and occupies the same time. It is, therefore, self-clocking, which means that a clock signal can be recovered from the encoded data. There are three modes:

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



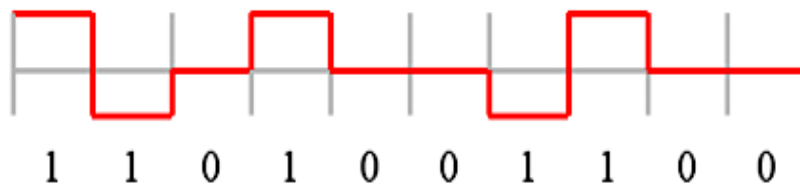
**Manchester (IEEE802.3):** A 1 is expressed by a low-to-high transition, a 0 by high-to-low transition.



**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:

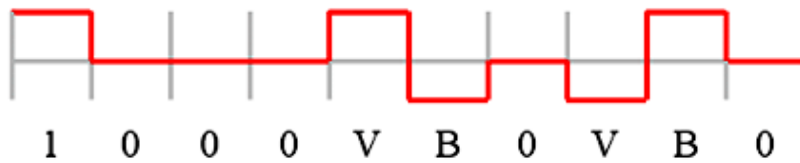
**AMI (Standard):** AMI (Alternate Mark Inversion) is a synchronous clock encoding technique that uses bipolar pulses to represent logical 1 value. It is therefore a three level system. A logical 0s is represented by no symbol, and a logical 1 is represented by alternating-polarity pulses.



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

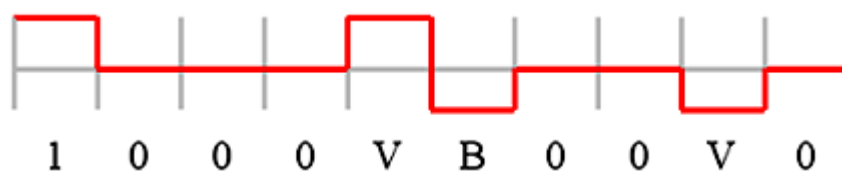
If 1 is +, 00000000 is represented to 000+-0-+

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



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

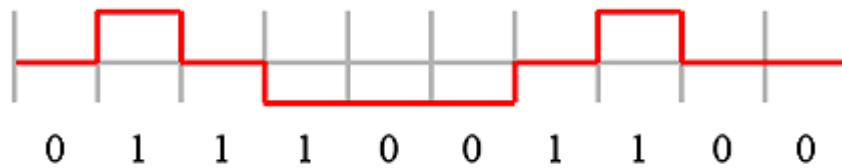
The HDB3 code is a bipolar signaling technique (i.e. relies on the transmission of both positive and negative pulses). It is based on Alternate Mark Inversion (AMI), but extends this by inserting violation codes whenever there is a run of 4 or more 0's. This and similar (more complex) codes have replaced 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.



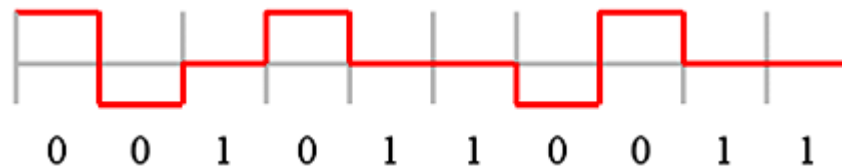
**MLT-3: Multilevel Transmission 3:** A 0 means no transition happens, a 1 is



represented by a transition (0, +, 0, -).



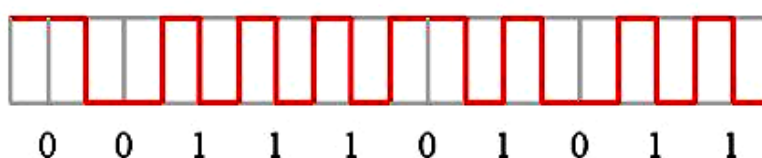
**Pseudoternary:** A 1 is always zero, a 0 is represented by a transition (+, -).



**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

**Line Encoding Settings**

Select Encoding: NRZI (Transition occurs for a one) ▼

☒ Auto-Detect Data Rate

Data Rate: 1 MHz

Channel: Data Channel: CH 0

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

OK Cancel

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

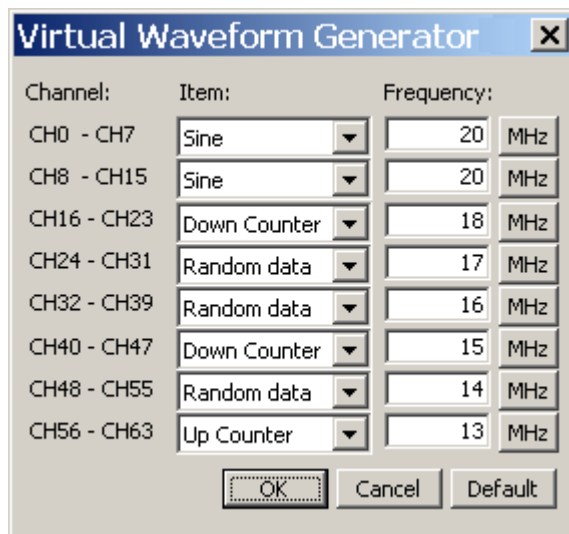
**Auto-Detect Data Rate:** Enter the Data Rate manually if the Auto-Detect Date Rate is not selected.

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

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

## Lissajous

**Add a Lissajous decode:** The Virtual Waveform Generator, only available when no instrument is connected to your PC, can generate sine waves to form a Lissajous graph. Click Virtual Waveform Generator from the Device menu or click Virtual Waveform Generator button on the Toolbar to show the dialog box below.



Set parameters like Item (Sine, ..., Up Counter) and Frequency (20MHz, ..., 13MHz).

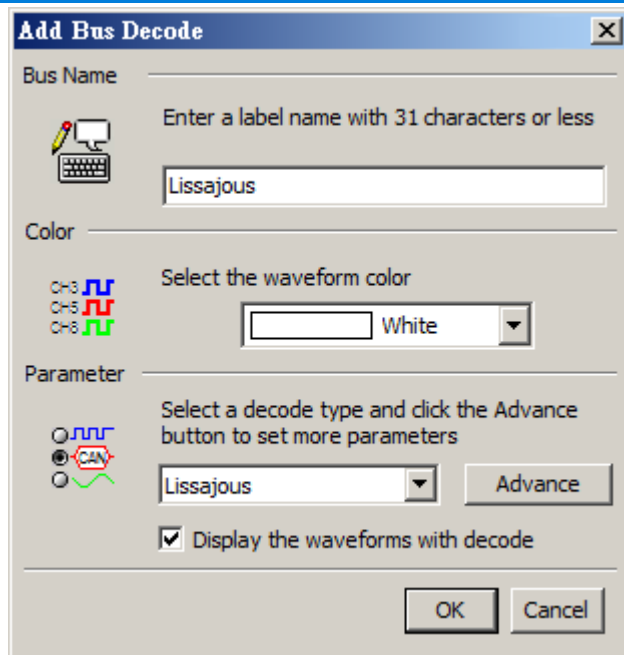
**Note:** CH0 and CH8 are the least significant bits (LSB), CH7 and CH15 are the most significant bits (MSB).

Combine the eight labels (CH0-CH7/CH8-CH17) to form a Sine wave bus and enter a new name (X/Y), then double click on the label name (X/Y) to show the Label

Settings dialog box below.

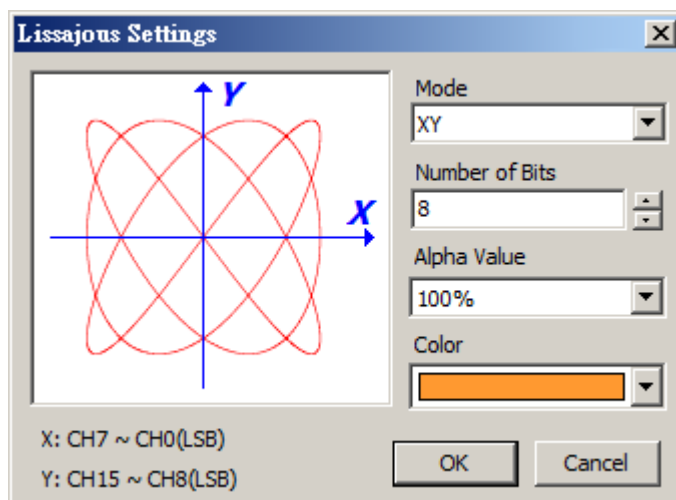
The screenshot shows the NI-MAX software interface. On the left, there is a table with two columns: 'Label' and 'Value'. The 'Label' column has two entries: 'X' and 'Y'. The 'Value' column has two entries: '7.0' and '15.8'. Below the table, there is a 'Channel' column with two entries: 'X' and 'Y'. To the right of the table, there is a large graph area displaying two waveforms. The top waveform is green and labeled 'X', and the bottom waveform is blue and labeled 'Y'. Both waveforms are sinusoidal and oscillate between approximately -1 and 1. The graph area has a black background with a grid of dashed lines. At the bottom of the interface, there is a status bar with several indicators: a red 'F' icon, a green 'A' icon, a yellow 'B' icon, and a green 'A' icon. There are also numerical values '0' and '0' and a small icon of a clock and a square.

Acute Technology Inc.  
Copyright ©2018



Select Lissajous in Parameter and click Advance to show the dialog box below.

## Settings



**Mode:** Set the mode for the axis (XY or IQ).

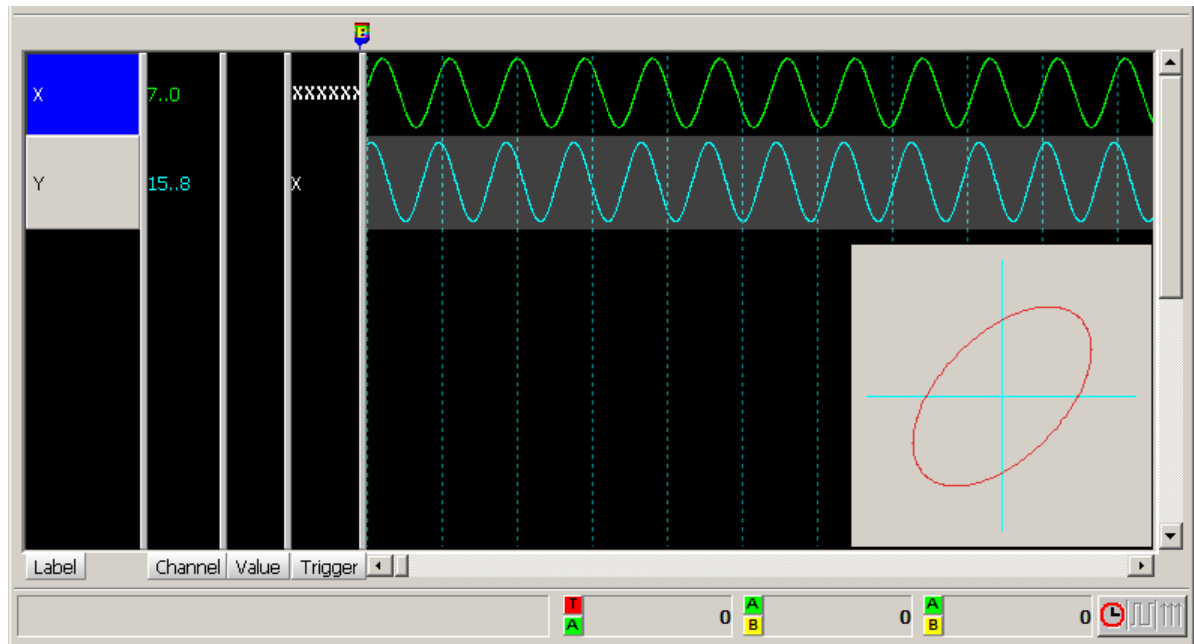
**Number of Bits:** Set the number of data bits (8).

**Alpha Value:** The higher, the less transparent for the Lissajous graph on the Waveform Window.

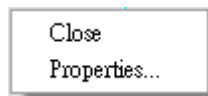
**Color:** Set the color for the Lissajous graph.

## Result

Click **OK** to see the Lissajous graph on the lower-right corner of the Waveform Window.



Right-click the Lissajous graph to show the dialog box below.



**Close:** Close the Lissajous graph.

**Properties:** Return to the Lissajous Settings.

## Low Pin Count (LPC)

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

### Settings

**LPC Settings**

Channel

LFRAME# CH 1 LAD[2] CH 4 LCLK CH 0  
 LAD[0] CH 2 LAD[3] CH 5 Data Edge Falling  
 LAD[1] CH 3

Show the field in report

☒ START  
☒ CYCLETYP+DIR  
☒ SIZE  
☒ TAR

Color

START ADDR  
 CYCTYPE+DIR DATA  
 CHANNEL SYNC  
 TAR IDSEL  
 SIZE/MSIZE STOP

Range

Decode Range  
 From To  
 Buffer Head Buffer Tail

Default OK Cancel

**Channel:** Show the selected channels.

**LFRAME#:** Frame indicator.

**LAD[0-3]:** Data bits.

**LCLK:** Clock.

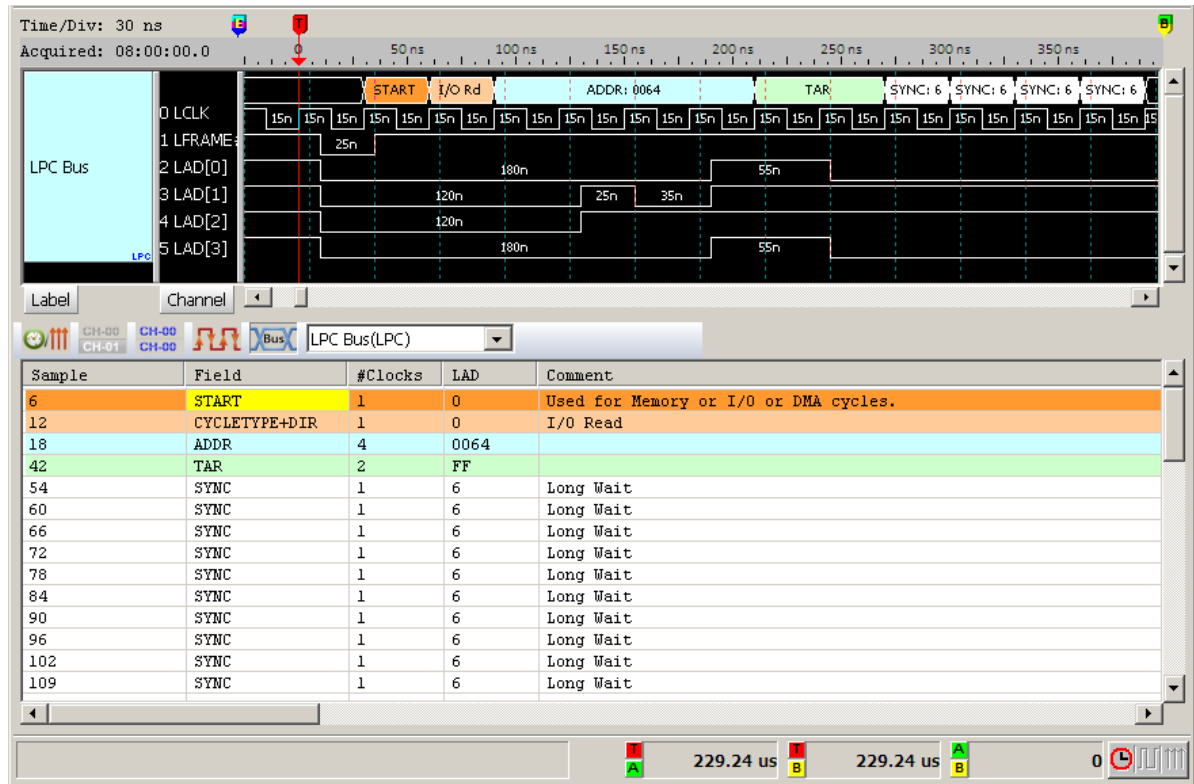
**Filter the data in the report window:** You can filter the data in the Report Window.



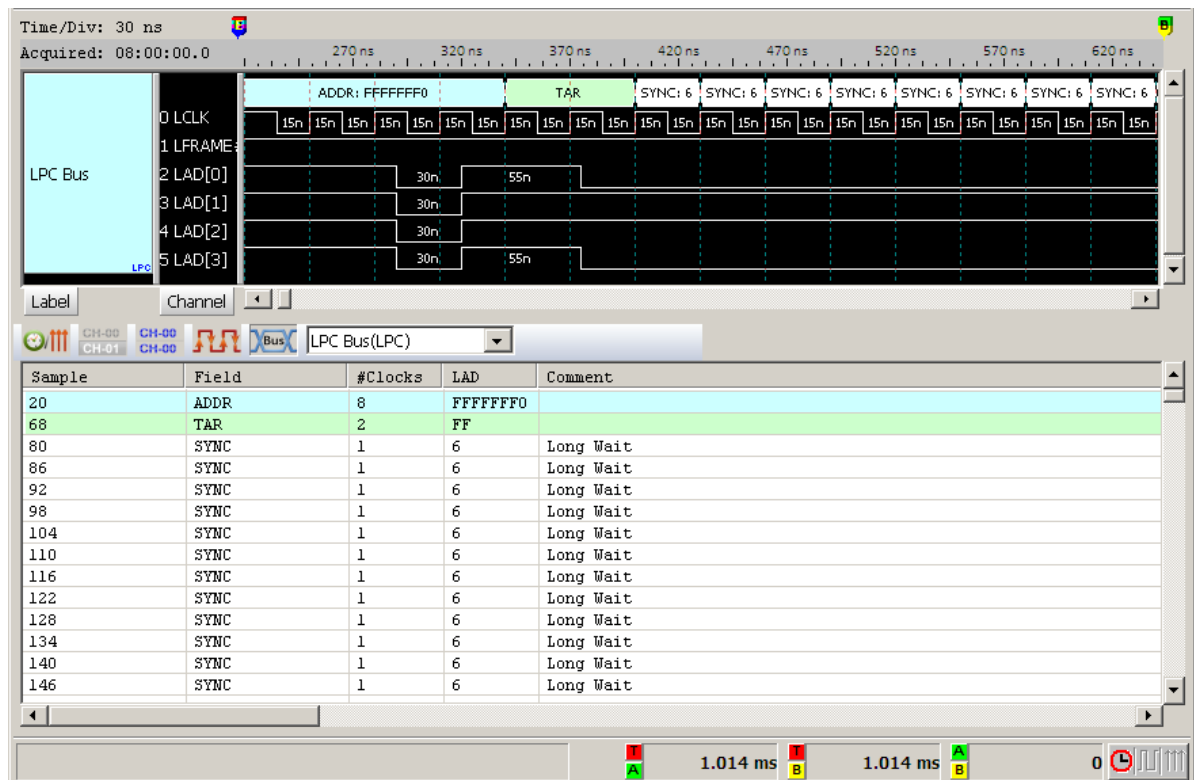
## Result

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

I/O Read Cycle.



Memory Read Cycle

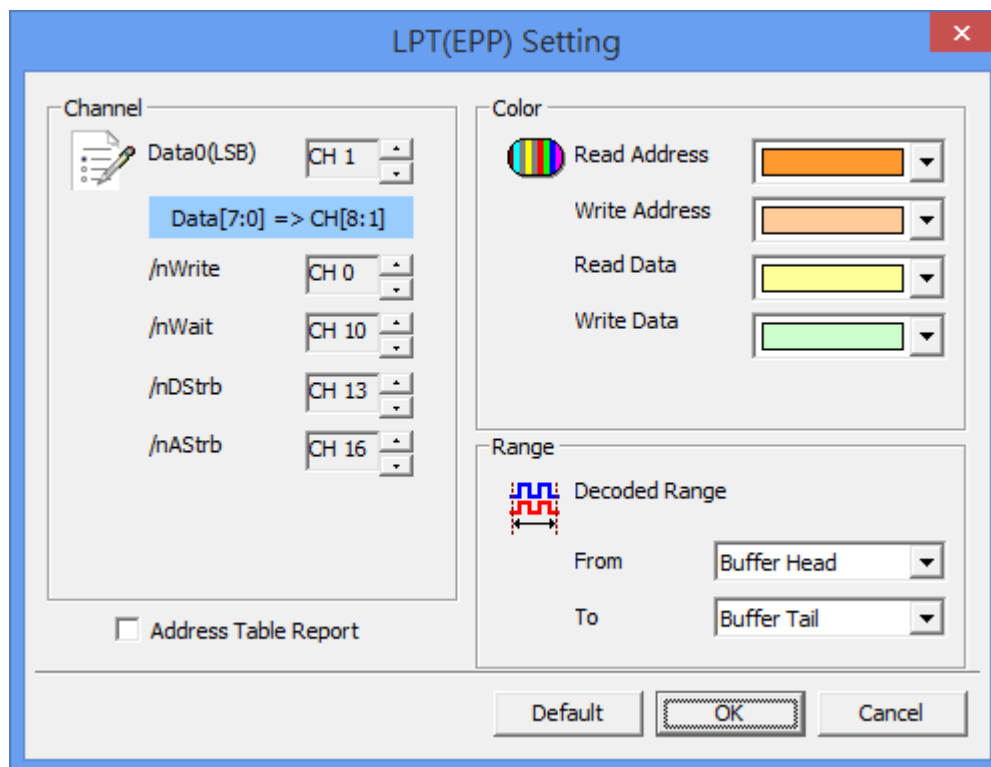


## Line Printer Terminal Port (LPT)

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.

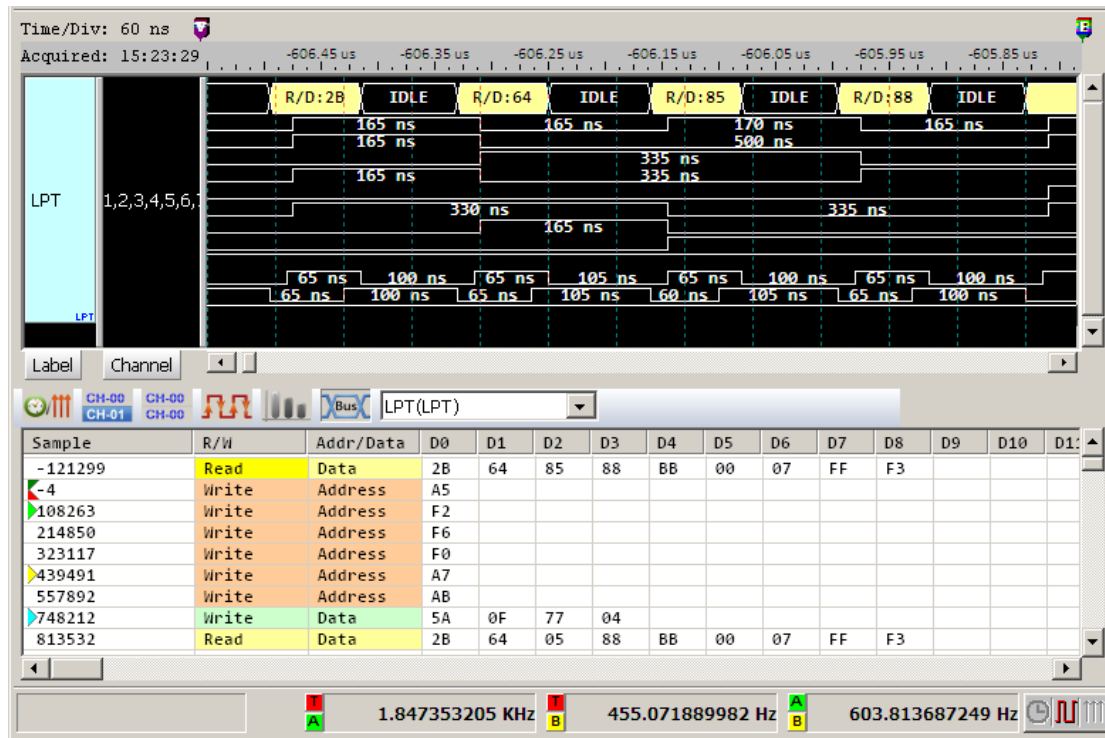
**/nAStrb:** Indicates the address cycle.

**/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:  ☒ Auto Detect

Master:  Baud Rate:

Polarity:  Parity:

☐ Slave:  ☐ MSB first

Polarity:  ☐ Adv. report

**Color**

User can assign color for specific pattern.

Start / Stop:  CI Field:

L Field:  Data:

C Field:  Check Sum:

A Field:

**Range**

From:  To:

Default OK Cancel

**Channel:** Set the channel of the signal and Polarity.

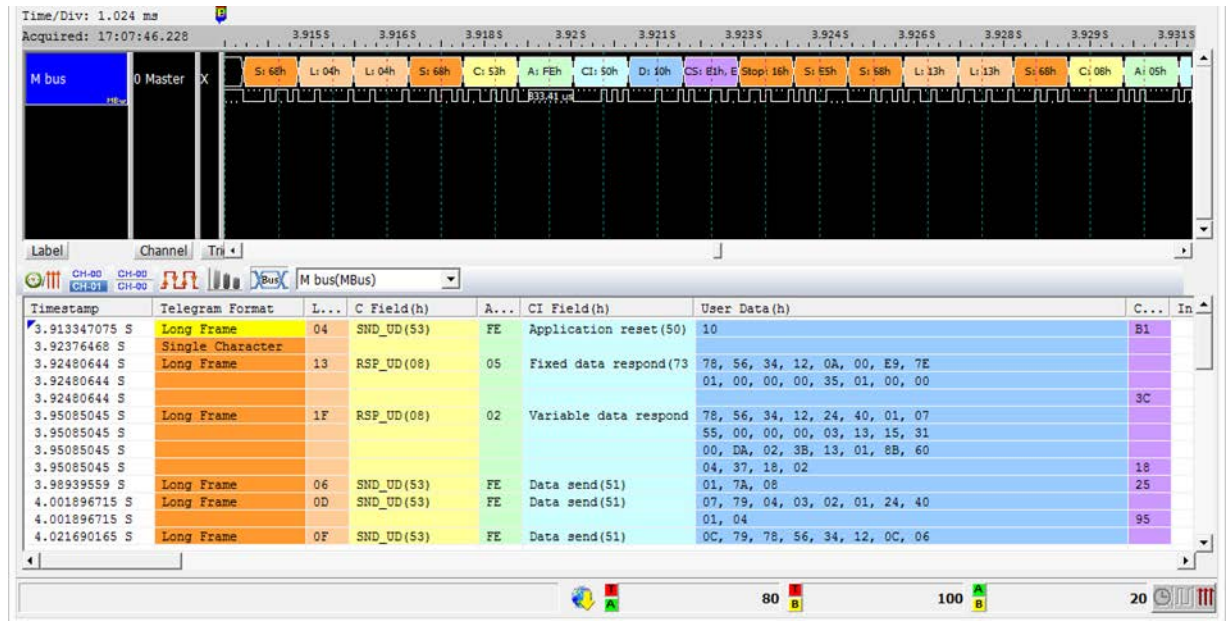
**Baud rate:** Set the specific data rate or auto detection.

**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

The image displays three sequential screenshots of the 'Math Settings' dialog box, illustrating the configuration options available in different tabs.

**Top Screenshot: Channel Settings**

- Channel Settings** (selected), Waveform Settings, Case Settings
- Setting** icon
- Operand 1:** B
- +** button
- =** button
- Operand 2:** B
- Math List:**
  - A + B
  - A XOR B
  - (A + B) + (A XOR B)
- Add List Item to Operand** button
- Delete List Item** button
- OK** button
- Cancel** button

**Middle Screenshot: Waveform Settings**

- Channel Settings, **Waveform Settings** (selected), Case Settings
- Math List:**
  - A + B
  - A XOR B
  - (A + B) + (A XOR B)
- Color** icon
- Range** icon
- Color** dropdown menu
- Calculational Range**
  - From:** Buffer Head
  - To:** Buffer Tail
- OK** button
- Cancel** button

**Bottom Screenshot: Case Settings**

- Channel Settings, Waveform Settings, **Case Settings** (selected)
- Math List:**
  - A + B
  - A XOR B
  - (A + B) + (A XOR B)
- Case Color** icon
- Calculational Result**
  - =** button
  - 00h
- Color** dropdown menu
- OK** button
- Cancel** button

**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.

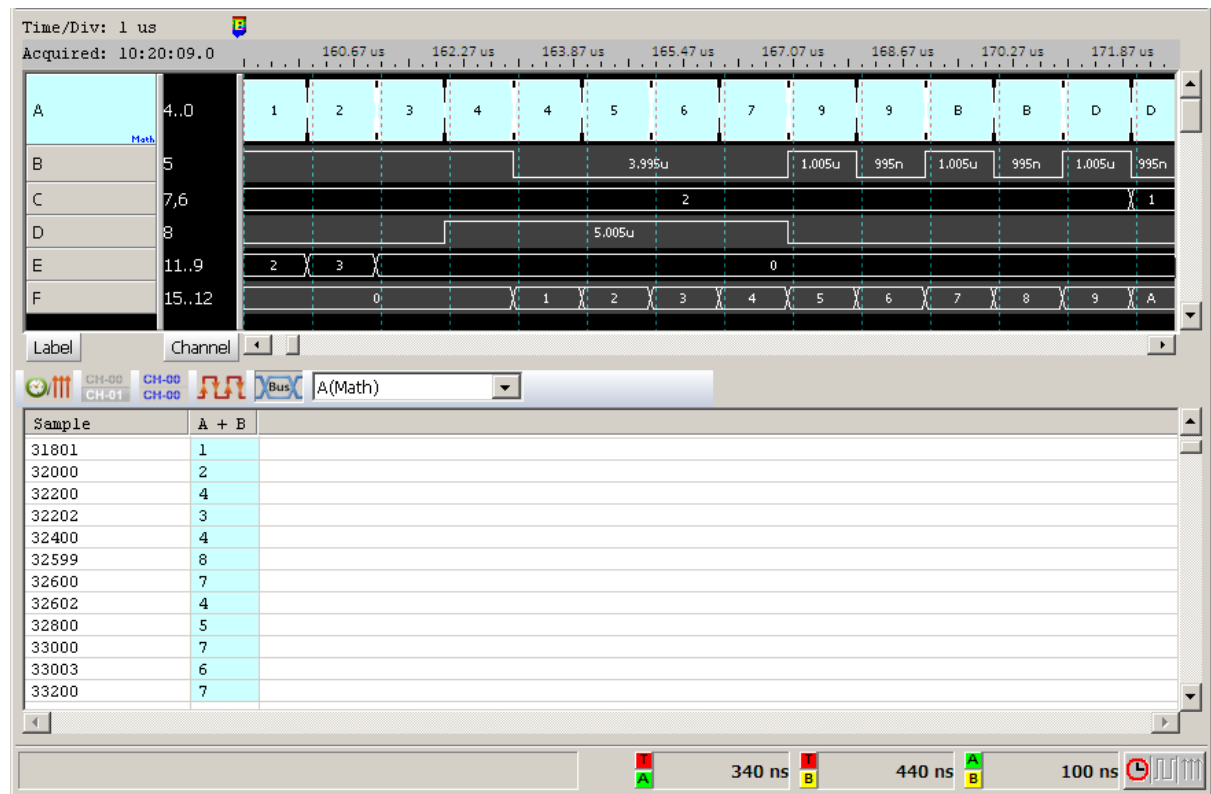
**Delete List Item:** Delete operation type from the list.

**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

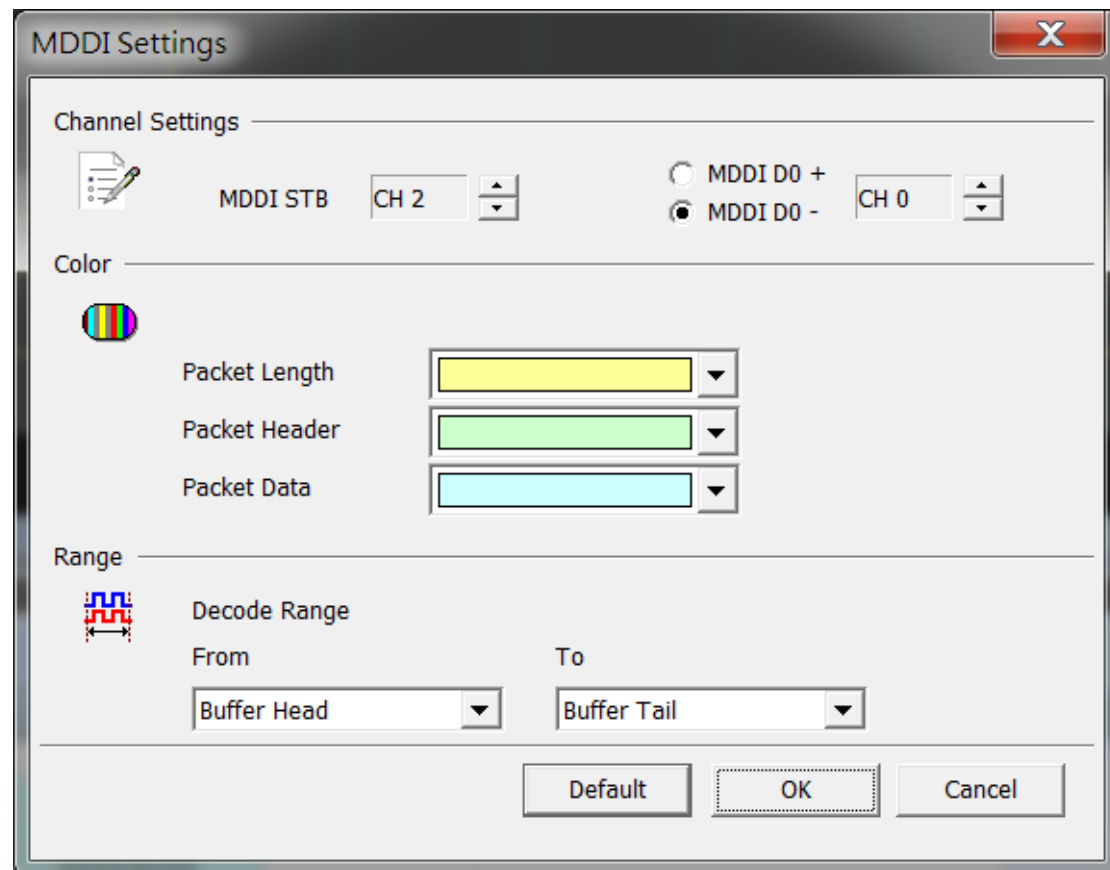


**Note:** When the Math settings are finished, the settings will always be saved as an independent text file named as AqMath.txt, different from the waveform file, at work directory unless other name assigned. If you need the specific Math settings, please save it, so you can reload the settings next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

## 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

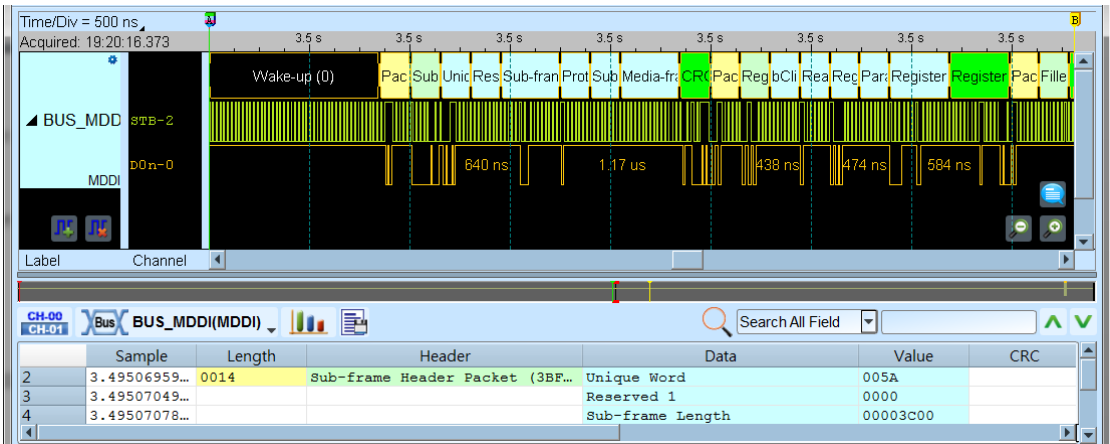
MDDI D0+/-: MDDI Data 0 +/-

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



from Data 0+ or Data 0-.

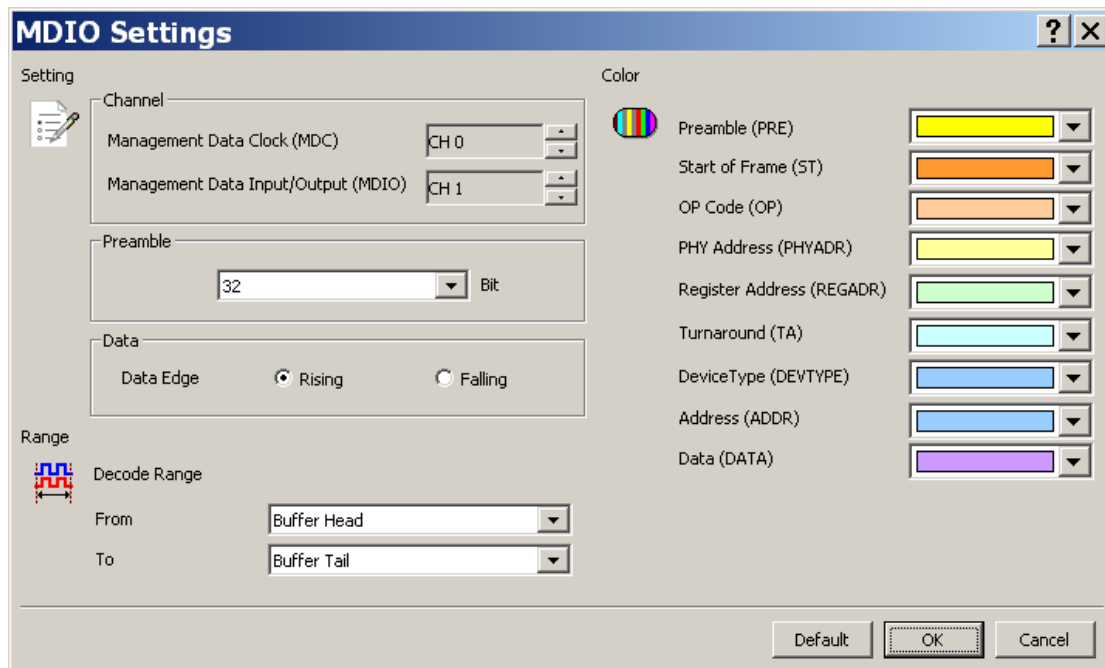
2. Result



## 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



The MDIO Settings dialog box is divided into several sections. On the left, under 'Setting', there is a 'Channel' section with dropdowns for 'Management Data Clock (MDC)' (set to CH 0) and 'Management Data Input/Output (MDIO)' (set to CH 1). Below this is a 'Preamble' section with a dropdown set to '32' and a 'Bit' label. Further down is a 'Data' section with 'Data Edge' options: 'Rising' (selected) and 'Falling'. At the bottom left is a 'Range' section with a 'Decode Range' icon and dropdowns for 'From' (set to 'Buffer Head') and 'To' (set to 'Buffer Tail'). On the right, under 'Color', there is a list of fields with corresponding color selection buttons: 'Preamble (PRE)' (yellow), 'Start of Frame (ST)' (orange), 'OP Code (OP)' (light orange), 'PHY Address (PHYADR)' (yellow), 'Register Address (REGADR)' (light green), 'Turnaround (TA)' (light blue), 'DeviceType (DEVTYPE)' (blue), 'Address (ADDR)' (blue), and 'Data (DATA)' (purple). At the bottom right are 'Default', 'OK', and 'Cancel' buttons.

**MDC:** Clock.

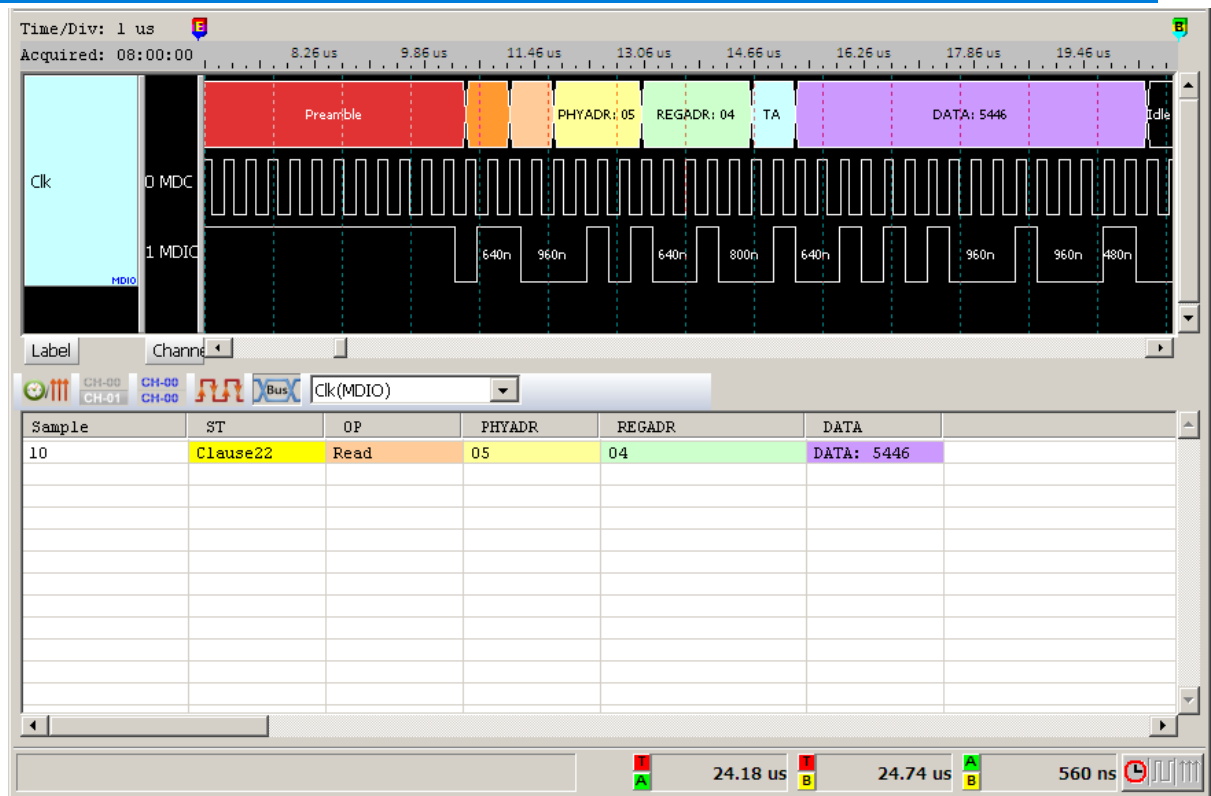
**MDIO:** Data Input / Output.

**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

**CBUS Settings**

**Channel**

Channel: CH 0

**Range**

Decode Range

From: Buffer Head

To: Buffer Tail

**Color**

Setting transmitter's

SYNC: [Orange]

HEADER: [Light Orange]

cPacket: [Yellow]

dPacket: [Light Green]

CMD / DATA: [Light Blue]

PARITY: [Blue]

ACK: [Purple]

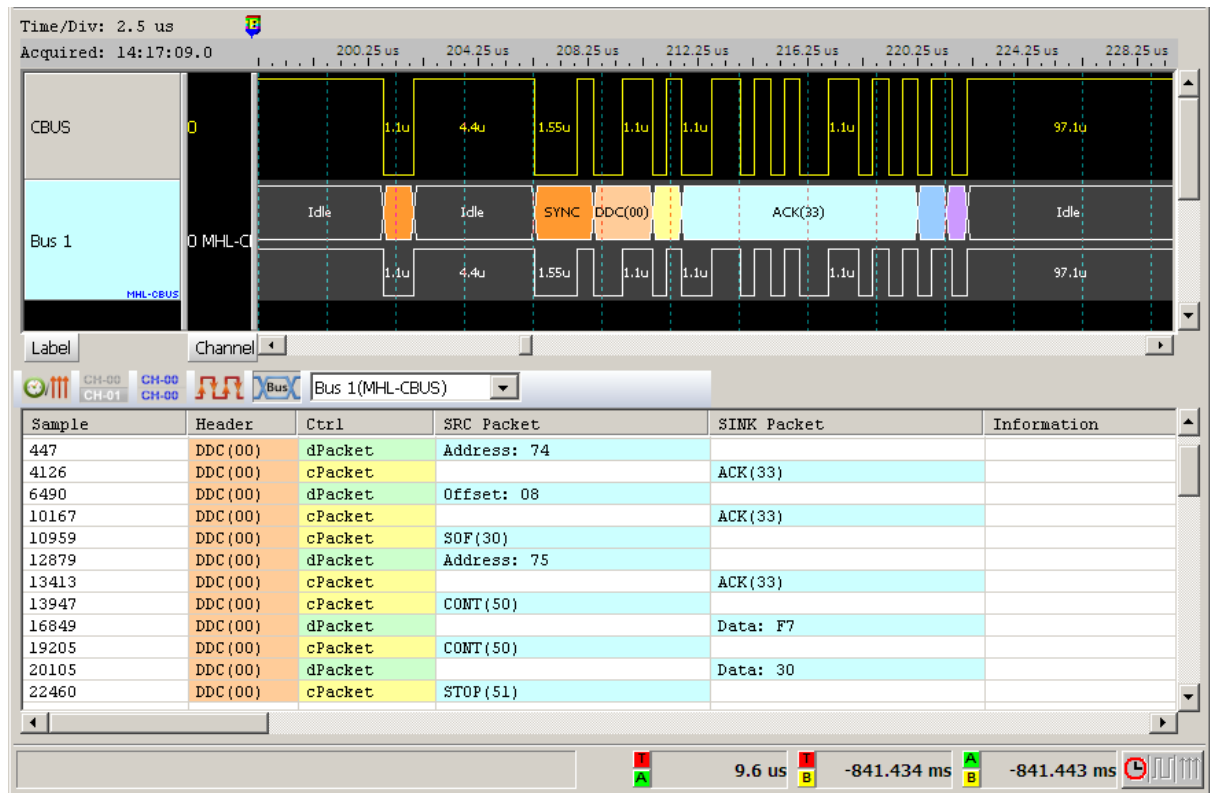
Arbitration: [Orange]

Default OK Cancel

**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

**MII / RMII Settings**

**Setting**

Bus Decode  
☒ MII ☐ RMII ☐ Only CLK and DATA pins used

Mode  
☒ Transmit(Tx)  
☐ Receive(Rx)  
☐ Duplex(Tx+Rx)

Channel

Transmit(Tx)		Receive(Rx)	
TX_CLK	CH 0	RX_CLK	CH 0
TX_D0	CH 1	RX_D0	CH 1
TX_D1	CH 2	RX_D1	CH 2
TX_D2	CH 3	RX_D2	CH 3
TX_D3	CH 4	RX_D3	CH 4
TX_EN	CH 5	RX_DV	CH 5
TX_ER	CH 6	RX_ER	CH 6
TX_COL	CH 7		

**Data**  
 Data Edge  
☒ Rising ☐ Falling

**Report**  
 Data columns  
☒ 8 Columns ☐ 16 Columns

**Color**

Data: [Yellow]  
 Error: [Red]  
 Collision: [Green]  
 Idle: [Black]  
 Unknown: [Blue]  
 Preamble / SFD: [Cyan]

**Range**

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

Default OK 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

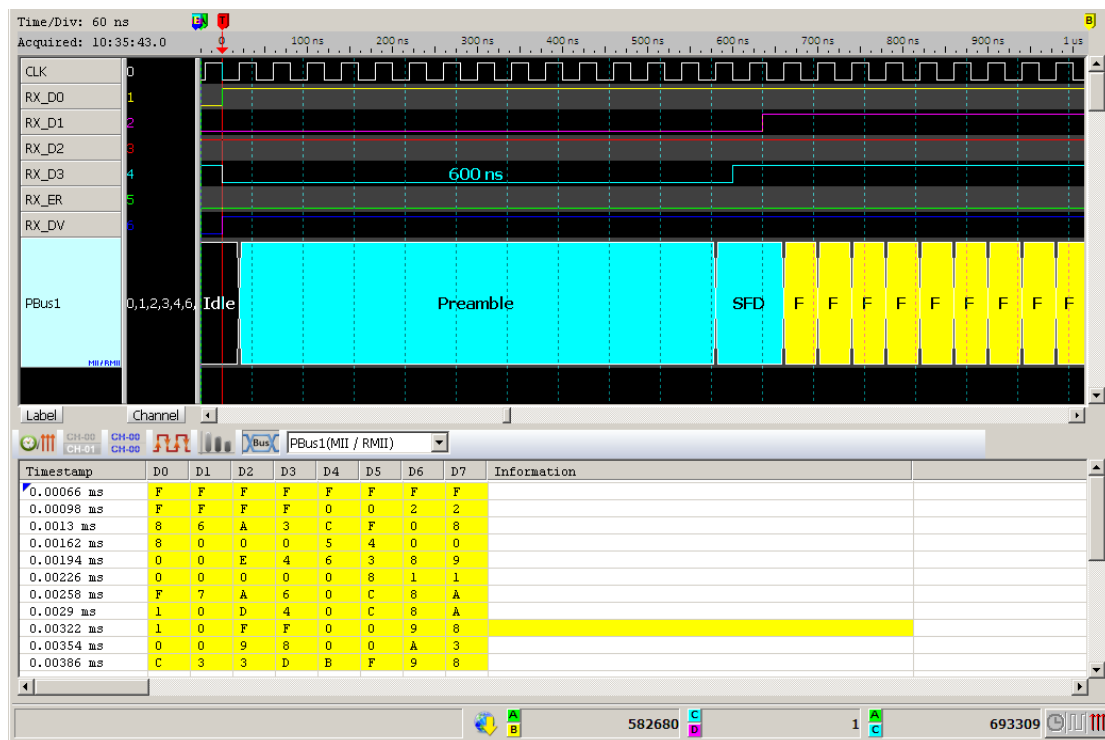
**8 columns:** show 8 columns data field in the report window

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

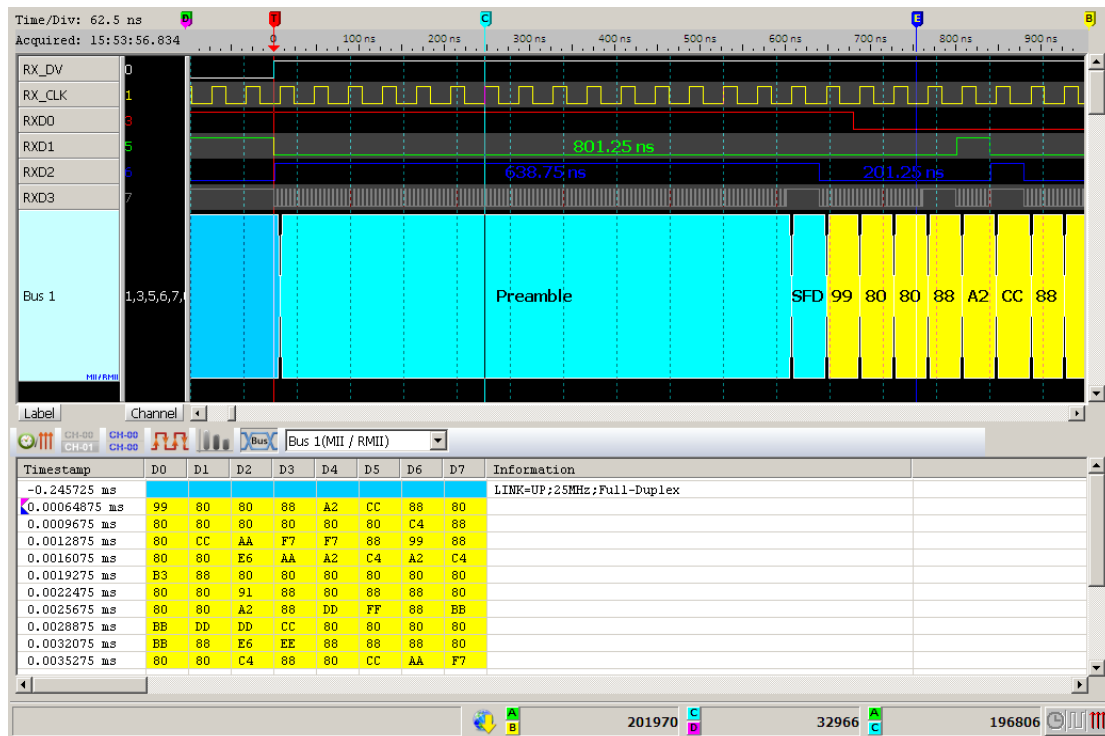
## **Result**

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

## MII



## RGMII

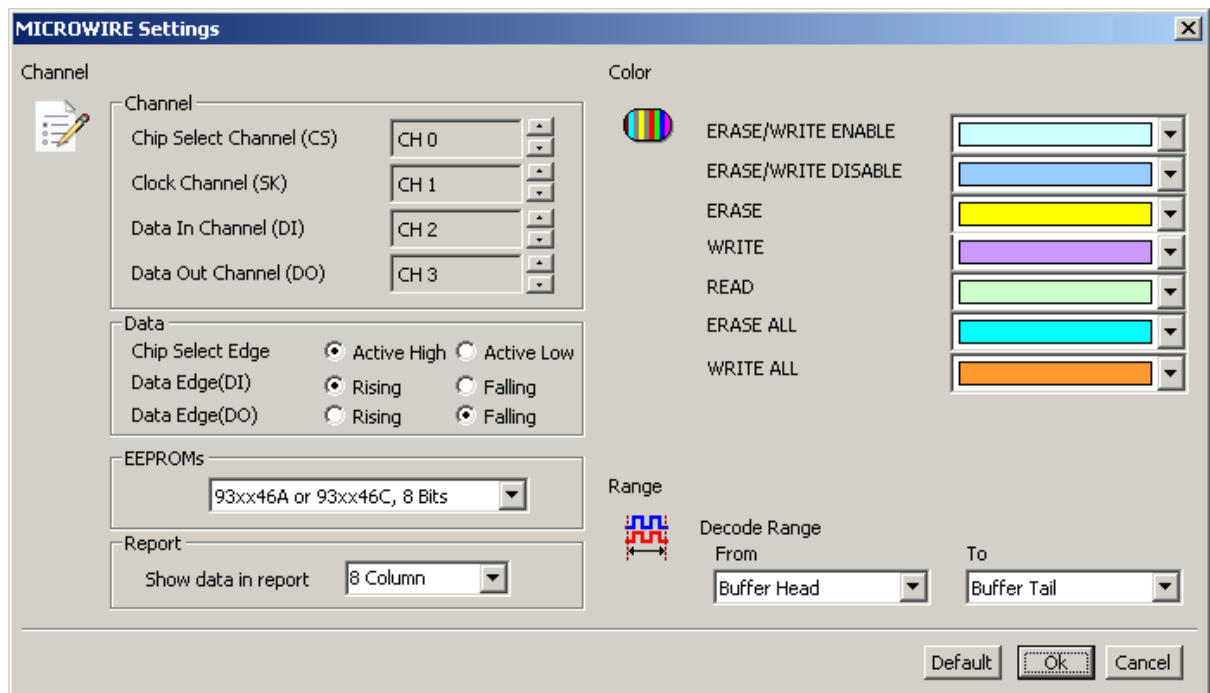




## Microwire

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

### Settings



The MICROWIRE Settings dialog box is divided into several sections:

- Channel:** Contains four dropdown menus for Channel Select Channel (CS), Clock Channel (SK), Data In Channel (DI), and Data Out Channel (DO). The selected values are CH 0, CH 1, CH 2, and CH 3 respectively.
- Data:** Contains three sets of radio buttons for edge settings:
  - Chip Select Edge: ☒ Active High, ☐ Active Low
  - Data Edge(DI): ☒ Rising, ☐ Falling
  - Data Edge(DO): ☐ Rising, ☒ Falling
- EEPROMs:** A dropdown menu showing "93xx46A or 93xx46C, 8 Bits".
- Report:** A dropdown menu showing "8 Column".
- Color:** A list of operations with corresponding color swatches:
  - ERASE/WRITE ENABLE: Light blue
  - ERASE/WRITE DISABLE: Blue
  - ERASE: Yellow
  - WRITE: Purple
  - READ: Green
  - ERASE ALL: Cyan
  - WRITE ALL: Orange
- Range:** A section with a "Decode Range" label and two dropdown menus:
  - From: Buffer Head
  - To: Buffer Tail

At the bottom right are buttons for "Default", "Ok", and "Cancel".

**Channel:** Show the selected channels.

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

**Data Edge:** Rising or Falling.

**EEPROMs:** Select EEPROMs.

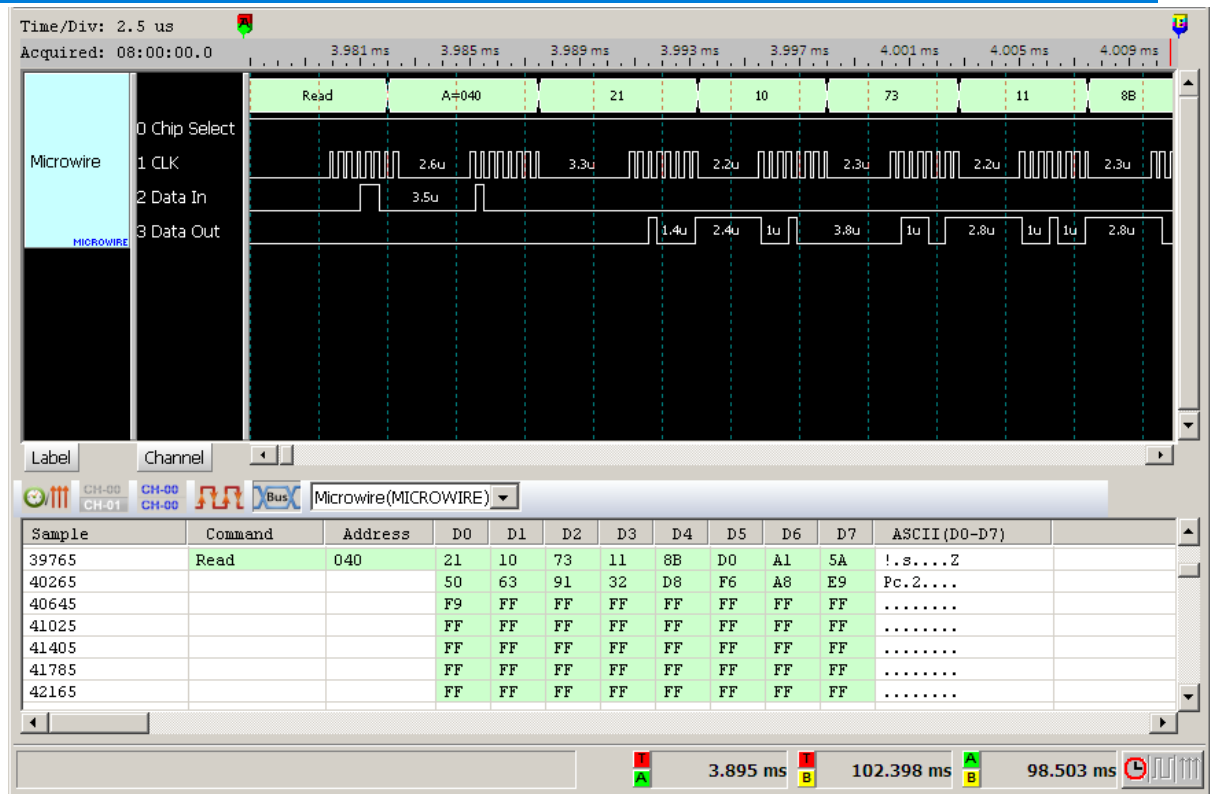
**Report:** Show data in report.

### Result

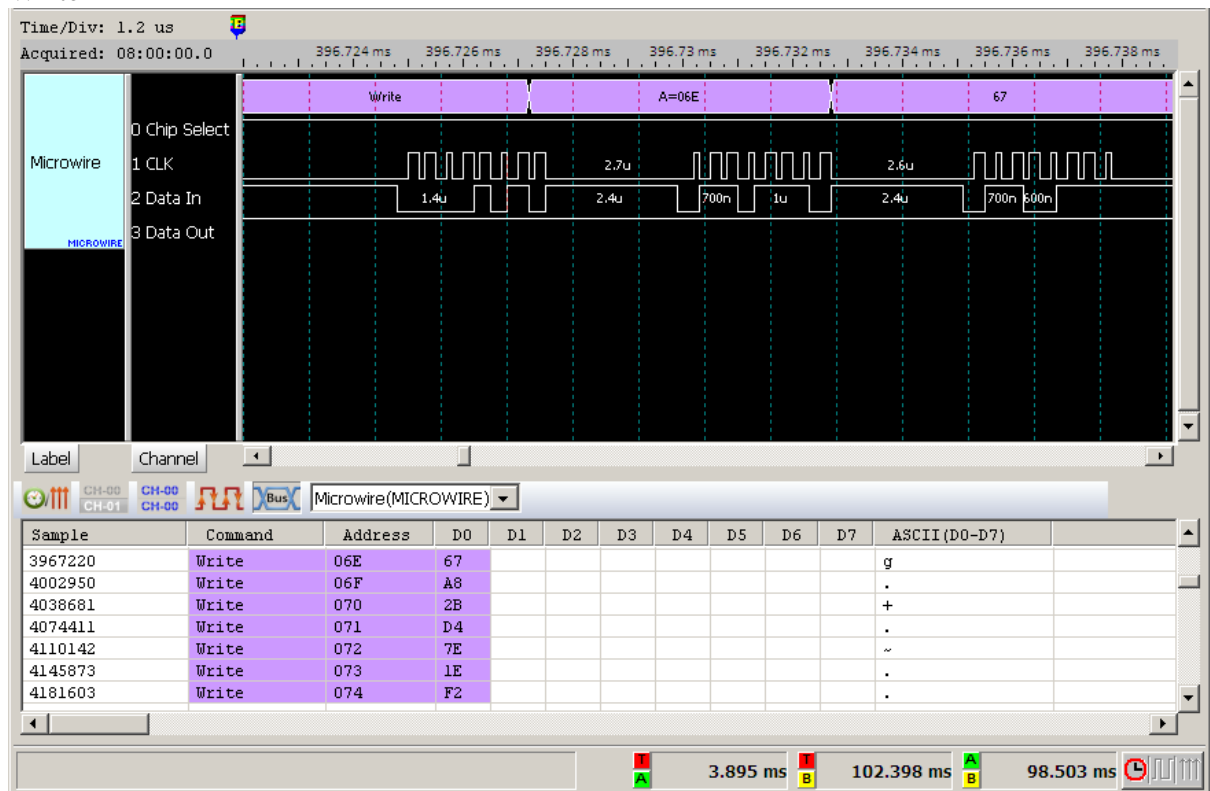
Click **OK** to run the Microwire decode and see the result on the Waveform Window

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.

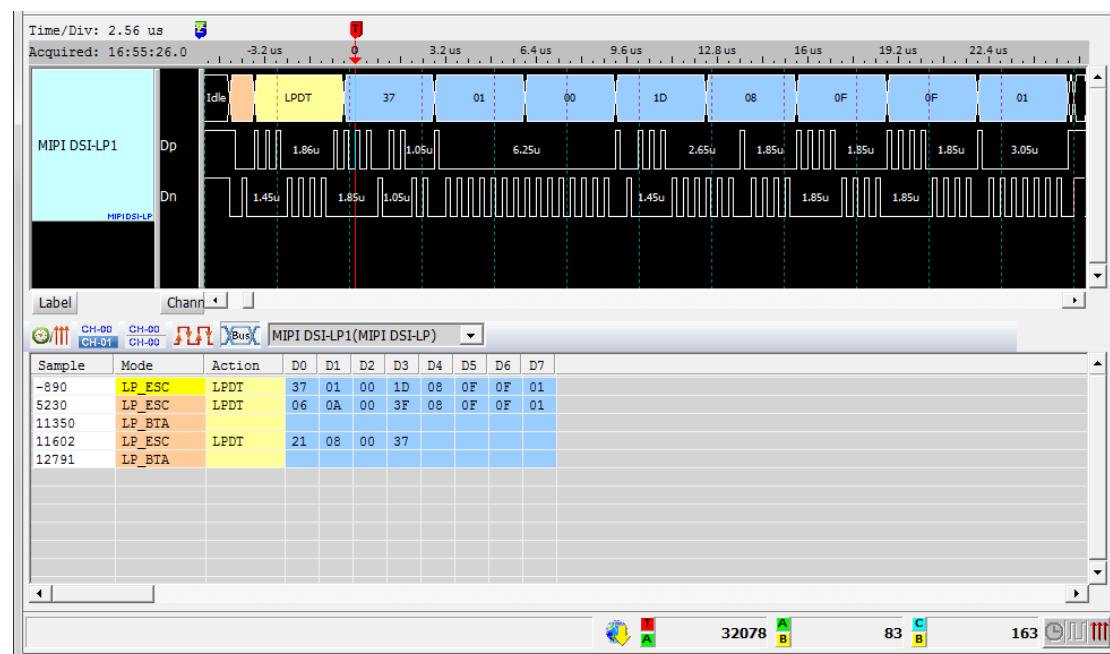
**Show DCS Command:** Enable DCS Command decode and display.

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

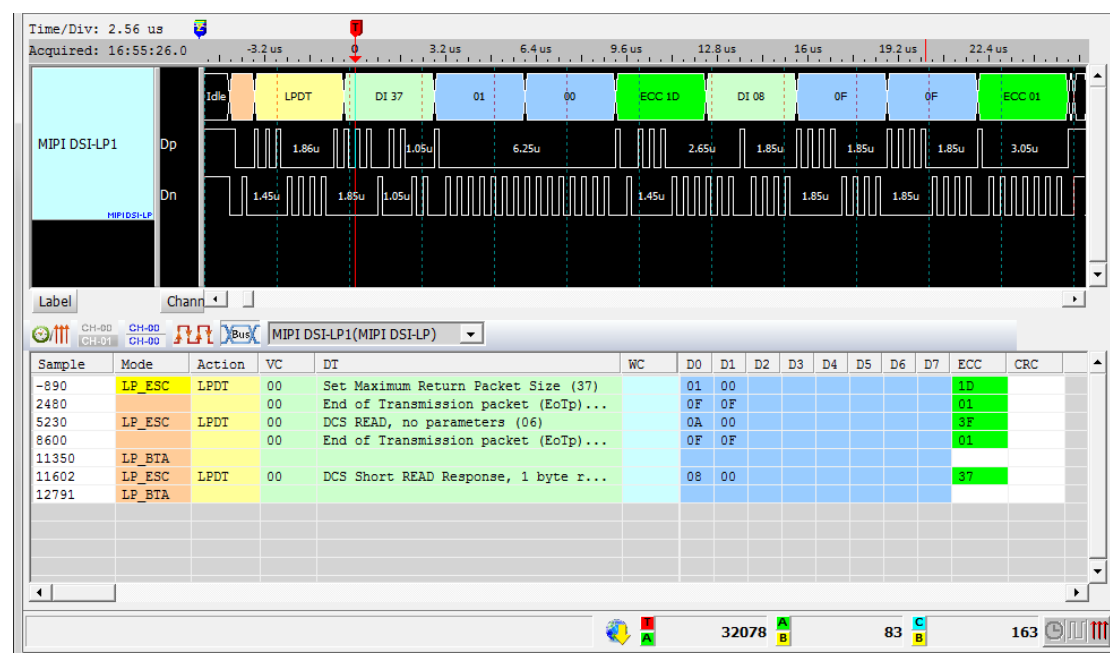
**Initial Bus Direction:** Select the Initial direction of the bus transmission.

## Result

Advanced Decode Disabled:



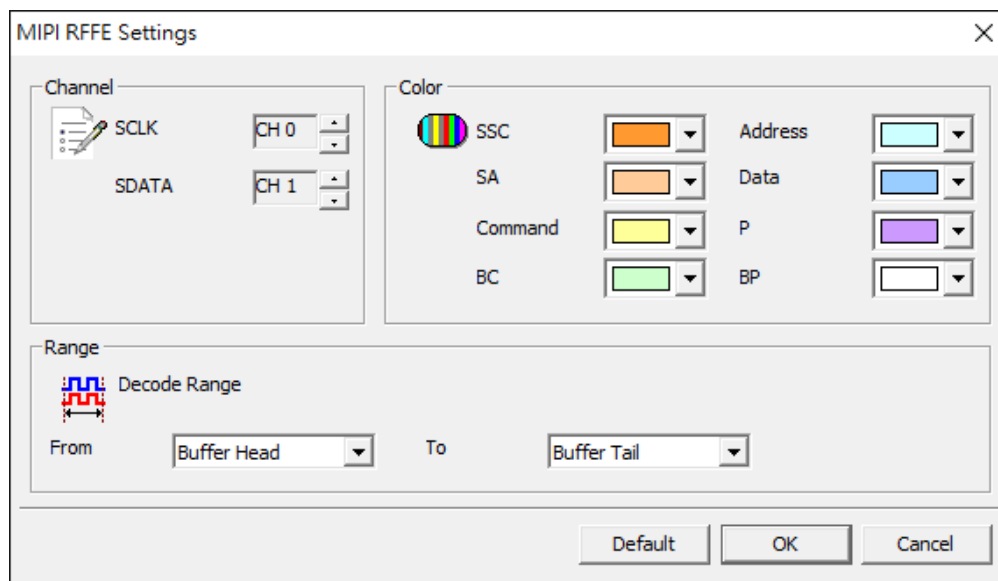
Advanced Decode Enabled:



## 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

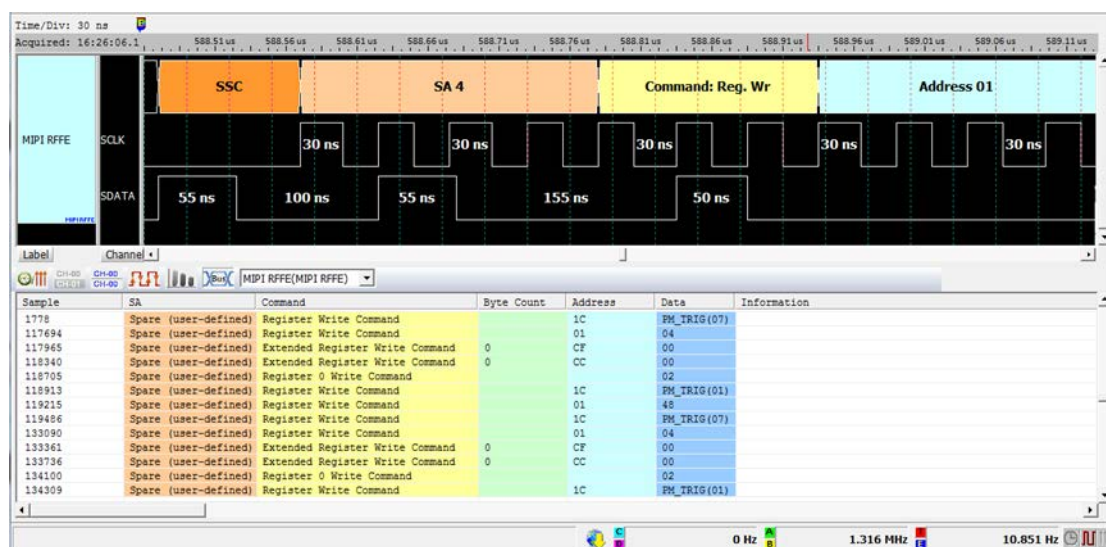


The MIPI RFFE Settings dialog box contains the following sections:

- Channel:** SCLK (CH 0) and SDATA (CH 1) dropdown menus.
- Color:** Color selection for SSC (orange), SA (orange), Command (yellow), BC (green), Address (cyan), Data (blue), P (purple), and BP (white).
- Range:** Decode Range section with a waveform icon and a range selector from "Buffer Head" to "Buffer Tail".
- Buttons:** Default, OK, and Cancel buttons at the bottom right.

**Channel:** Set the channels of SCLK and SDATA.

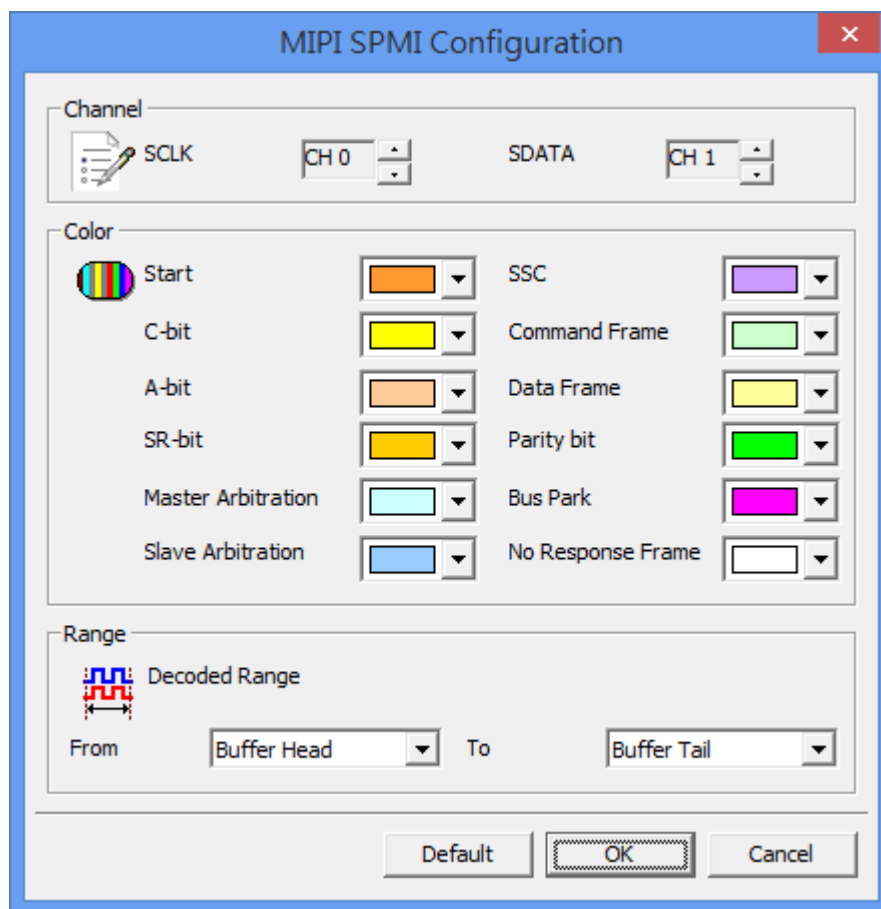
### Result



## 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).

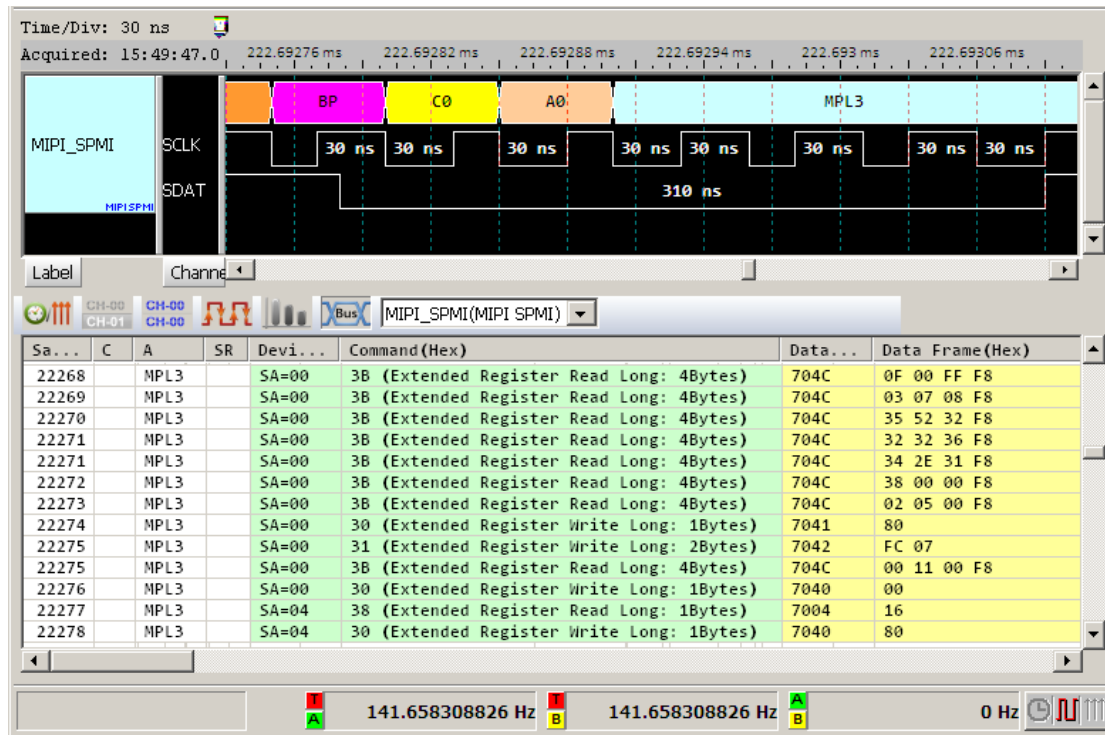
### Settings



**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 5.0 is a flash memory card standard.

### Settings

**Channel:** Show the selected channels.

**Command only:** Analyze the command.

**Data only:** Analyze the data

**Command + Data:** Analyze Command and Data in the report window.

**Ref. DAT0:** To help analyze the R1/R1b of the response.

**Adv. Report:** To analyze the command argument.

**Don't care clock:** To decode only depend on the CMD channel without the CLK channel.

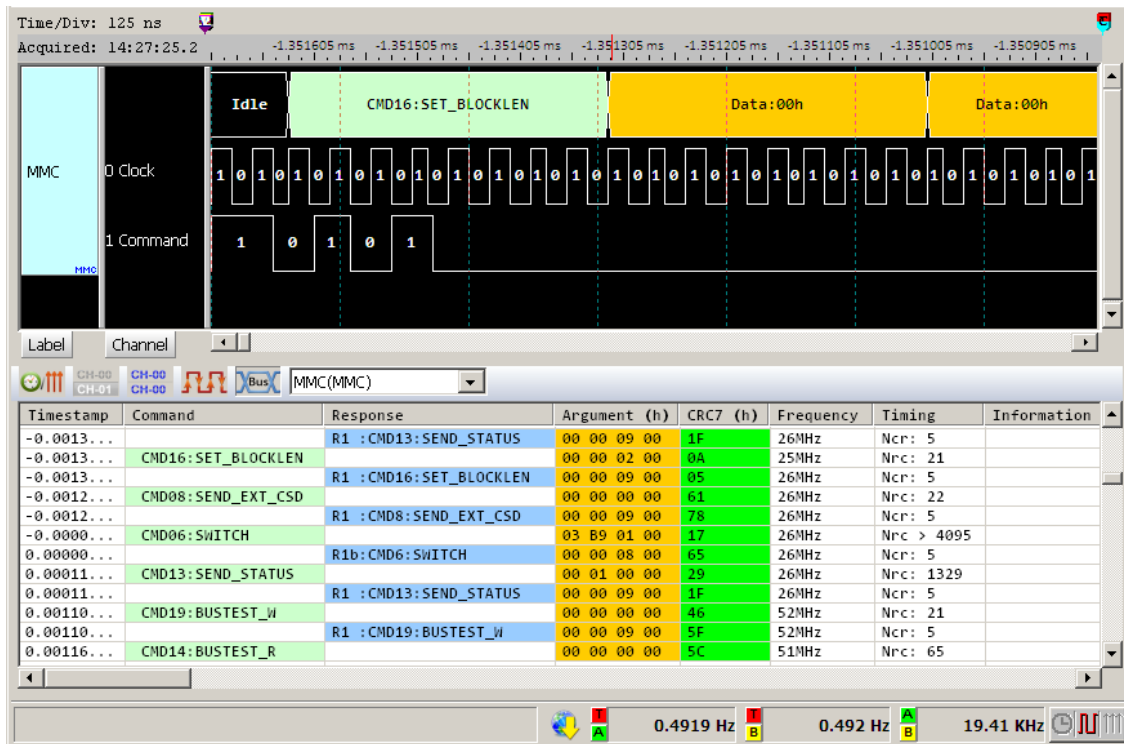
**Data:** 1/4/8 bits or DDR mode, check “DDR mode” and ” “Non-interleaved” to analyze data without interleaved. Check “Data Strobe” to analyze data with the DQS channel.



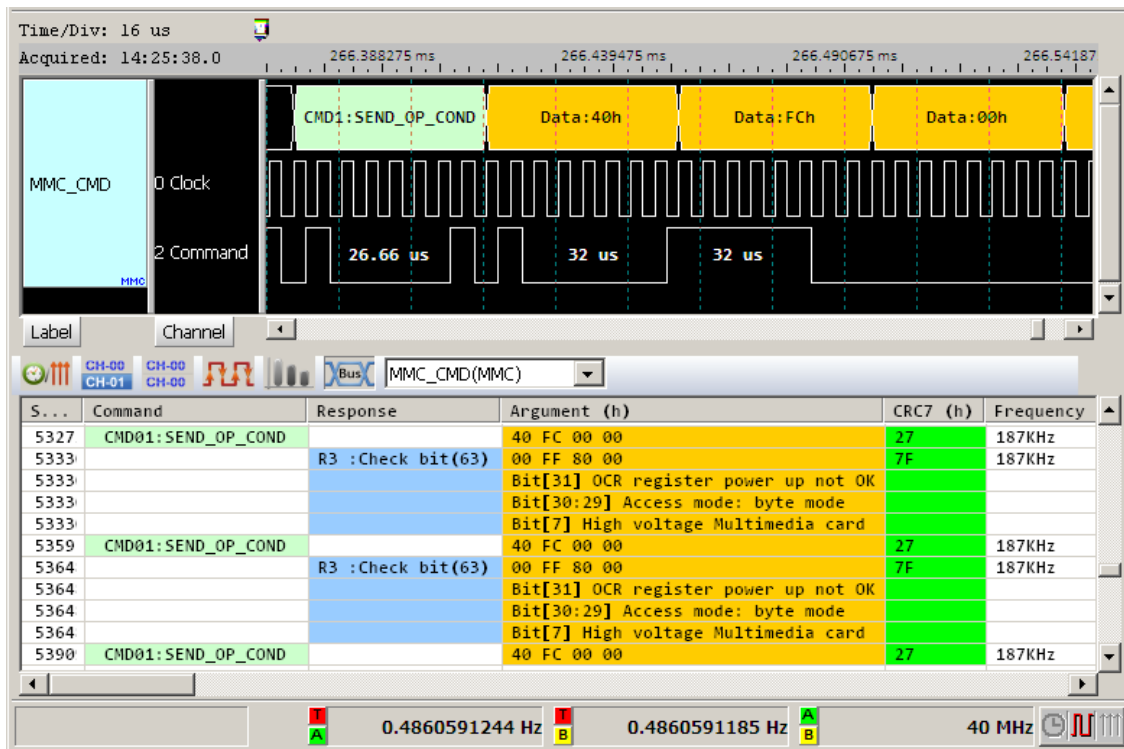
**Data length:** Set the number of data bits.

## Result

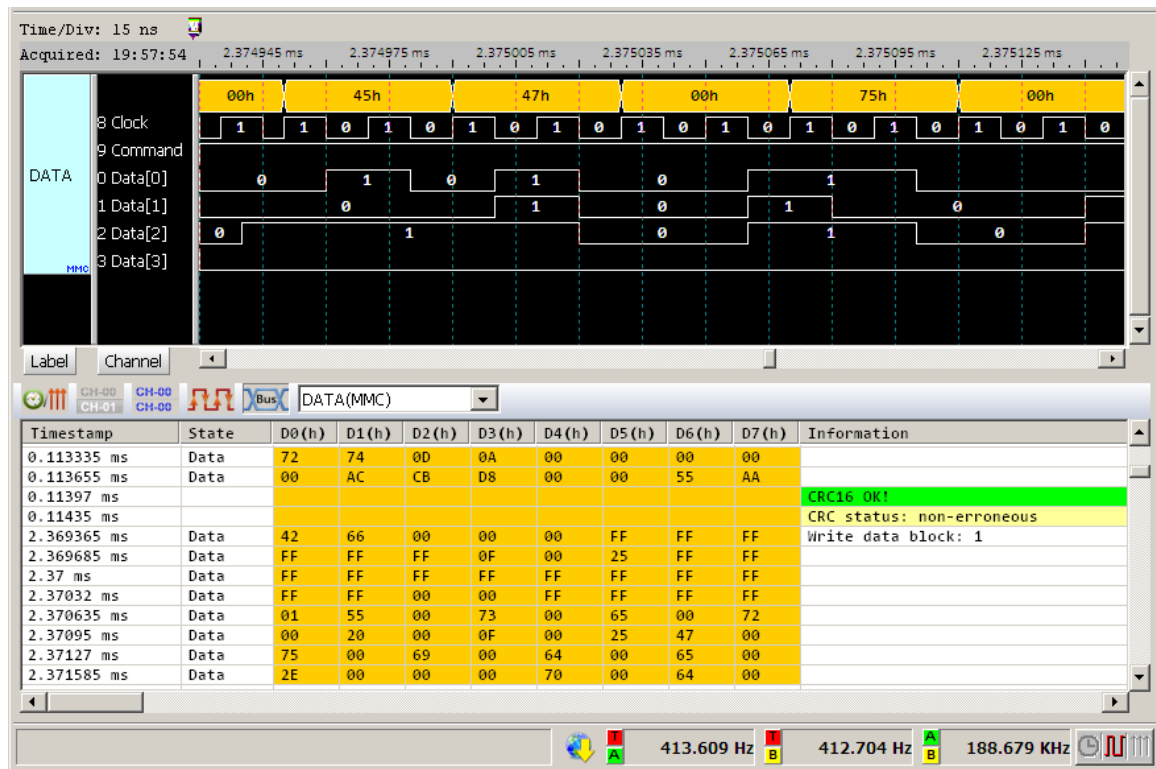
Command:



Adv. Report:



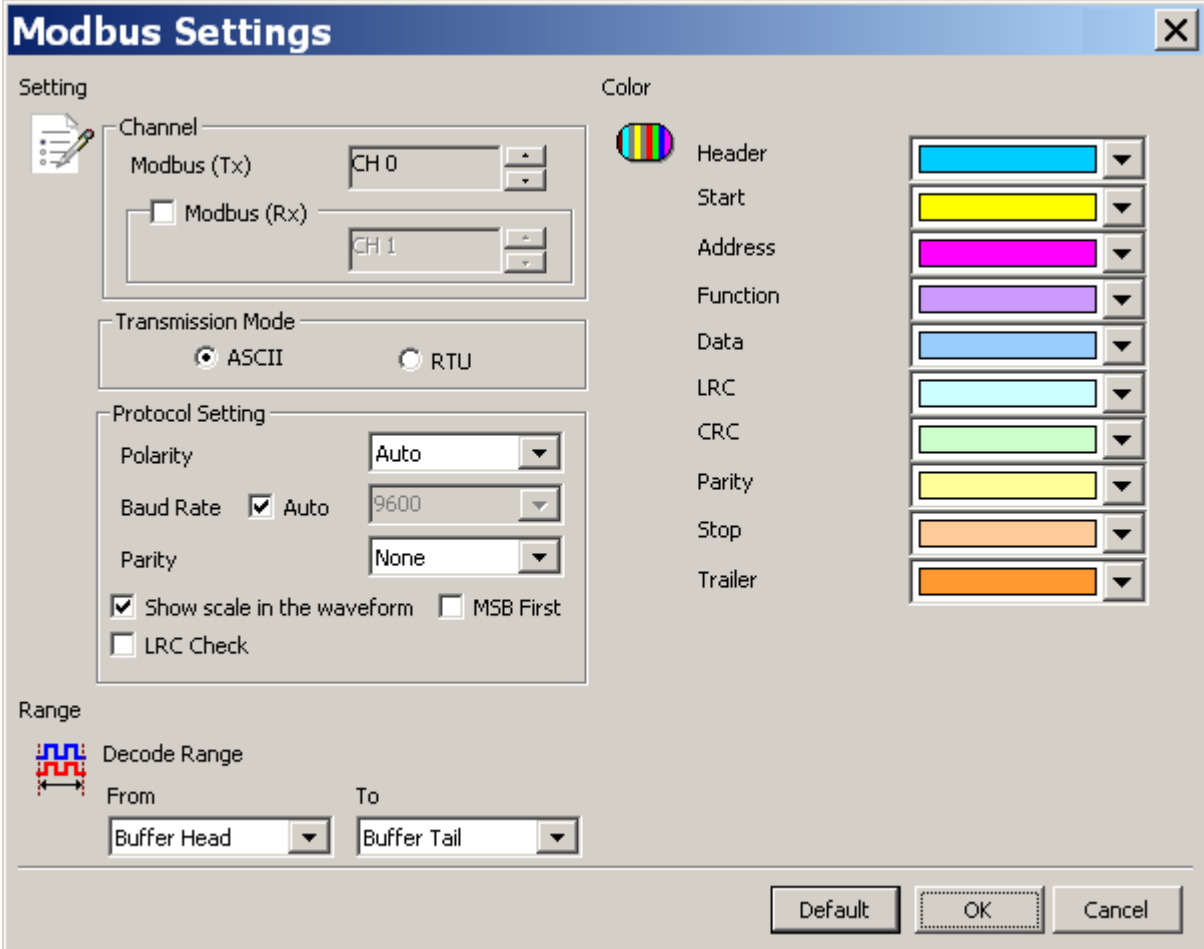
Data:



## 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



The **Modbus Settings** dialog box is used to configure the Modbus communication parameters. It includes sections for Setting, Color, Range, and buttons for Default, OK, and Cancel.

**Setting**

- Channel:**
  - Modbus (Tx): CH 0
  - Modbus (Rx): CH 1
- Transmission Mode:**
  - ☒ ASCII
  - ☐ RTU
- Protocol Setting:**
  - Polarity: Auto
  - Baud Rate: ☒ Auto 9600
  - Parity: None
  - ☒ Show scale in the waveform
  - ☐ MSB First
  - ☐ LRC Check

**Color**

- Header: [Blue]
- Start: [Yellow]
- Address: [Magenta]
- Function: [Purple]
- Data: [Light Blue]
- LRC: [Cyan]
- CRC: [Light Green]
- Parity: [Yellow]
- Stop: [Orange]
- Trailer: [Dark Orange]

**Range**

- Decode Range:**
  - From: Buffer Head
  - To: Buffer Tail

Buttons: Default, OK, Cancel

**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.

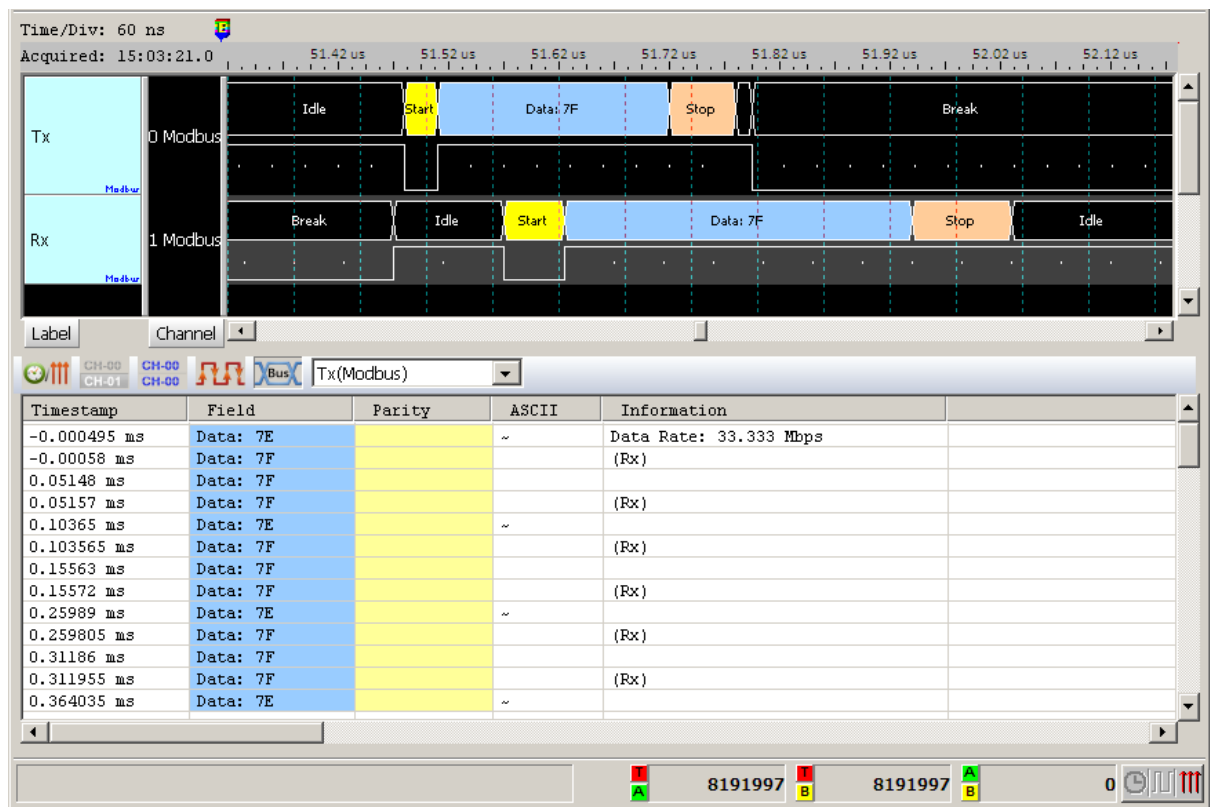
**Baud Rate:** Data rate (bits per second), and the range is 110 ~ 2M (bps).

**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:

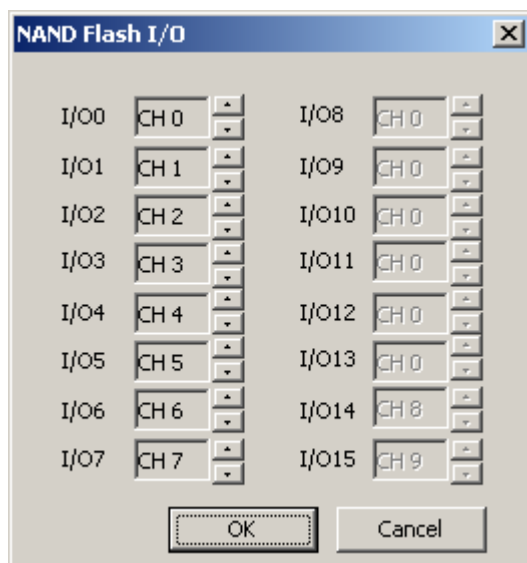
Async	Ssync	Description
I/Ox	DQx	NAND Flash data channels
CLE	CLE	Command Latch Enable channel
ALE	ALE	Address Latch Enable channel

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.

**tREA / tDQSQ:** Set the delay time to access the NAND data under SDR / DDR **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.

**Reduced Report:** Only show NAND Flash command in the Report window when check it.

**Show the DDR Data Output/Input Timing:** Only show timing information under DDR mode.

**Reduced Command in the waveform window:** Only show NAND Command value in the waveform window.

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

Description of file name as following:

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

Ex:NF\_DI\_Row017821h\_Col0000h\_CE1\_1.bin

NF\_DO\_Row017821h\_Col0000h\_CE1\_2.bin

NF\_DO\_Row\_Col\_CE1\_3.bin

Compare the content of file with the one of report.

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

```

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

```

## Device information

**Vendors:** Select the NAND Flash Vendor. Please refer to the

following details when select the **Custom** item.

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

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

```

Manufacturer=Samsung
PartNo=K9XXXXXXXXX
#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 is composed of several parts, it's divided with comma.
	1. Complete command name.



	2. Abbreviation of command.
	Name of first busy time check. Put a space and add a comma if unused.
	Value of first busy time check. Its unit is micro seconds. Put a space and add a comma if unused.
	Name of second busy time check. Put a space and add a comma if unused.
	Value of second busy time check. Its unit is micro mseconds. Put a space and add a comma if unused.
	First flag. It's acceptable command during busy.
	Second flag. It can be inserted by some command or not.
	Third flag. It can insert into some multi plane command or not.
	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

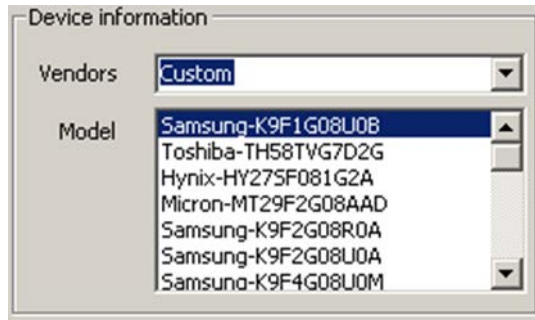
Read Status / Two-Plane Page Program : complete command.

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.

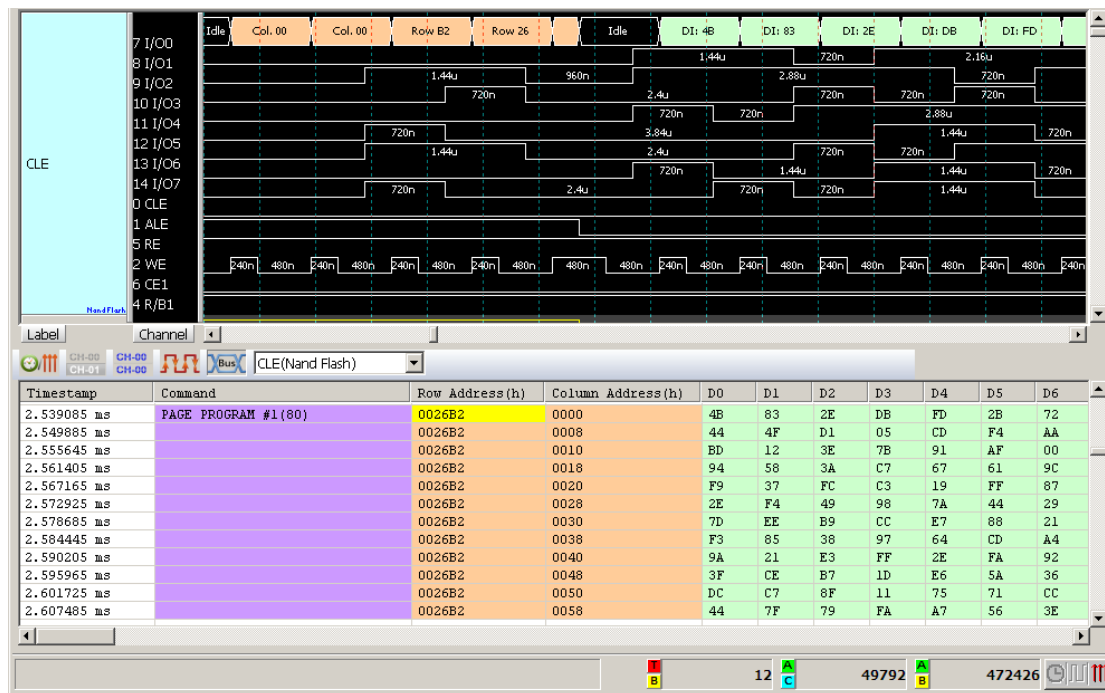
3 Flags: 1<sup>st</sup> flag of "Read Status" is Y means it's acceptable command

During busy; 2<sup>nd</sup> flag of "Two-Plane Page Program" and 3<sup>rd</sup> flag of "Read Status" means any command between 11h and 81h is prohibited except "Read Status (70h)".

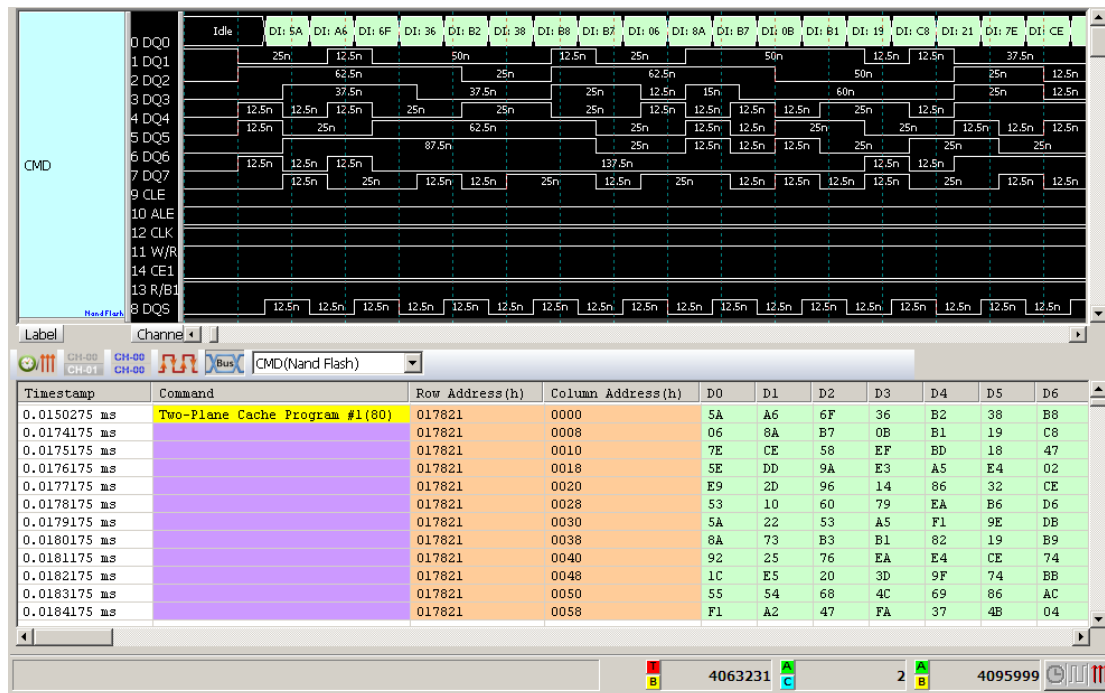


## Result

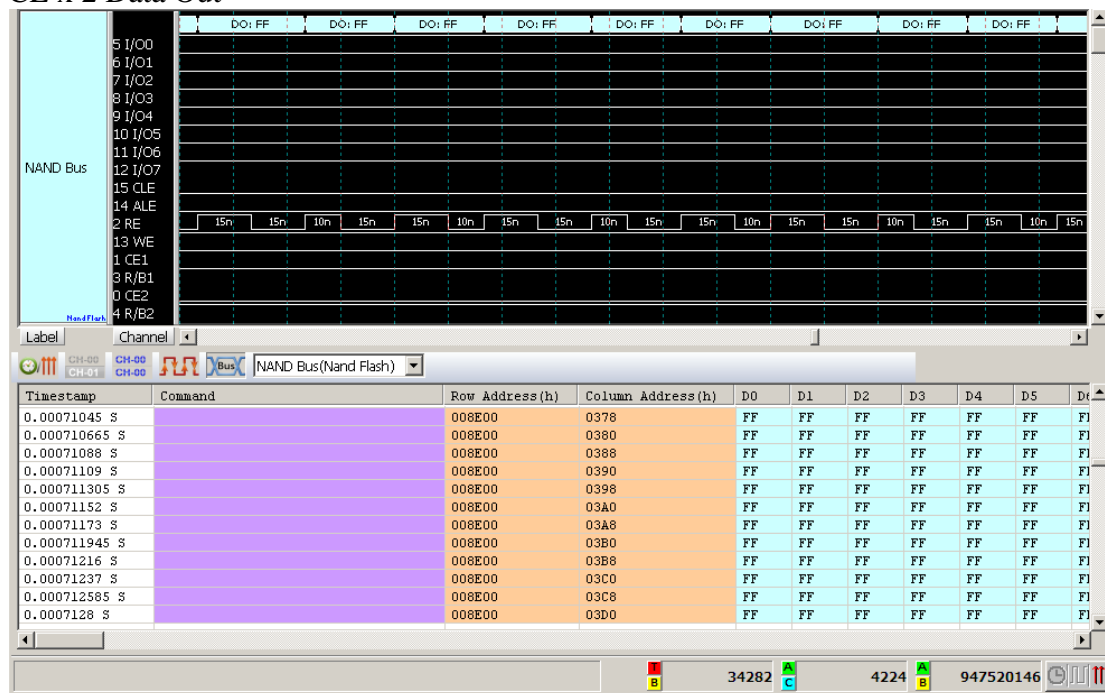
### SDR Data In



### DDR Data Out



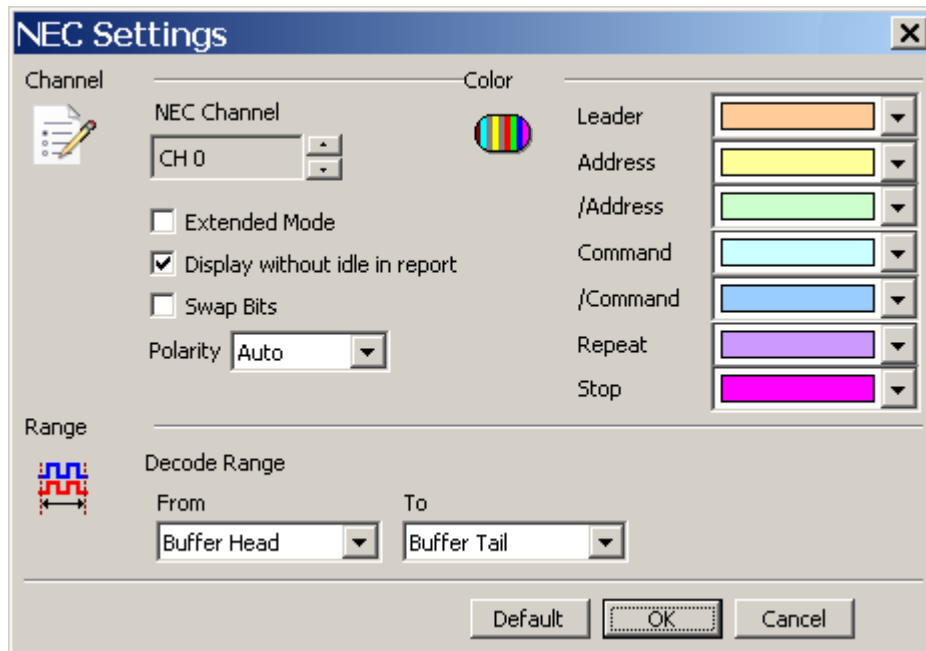
## CE x 2 Data Out



## 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.

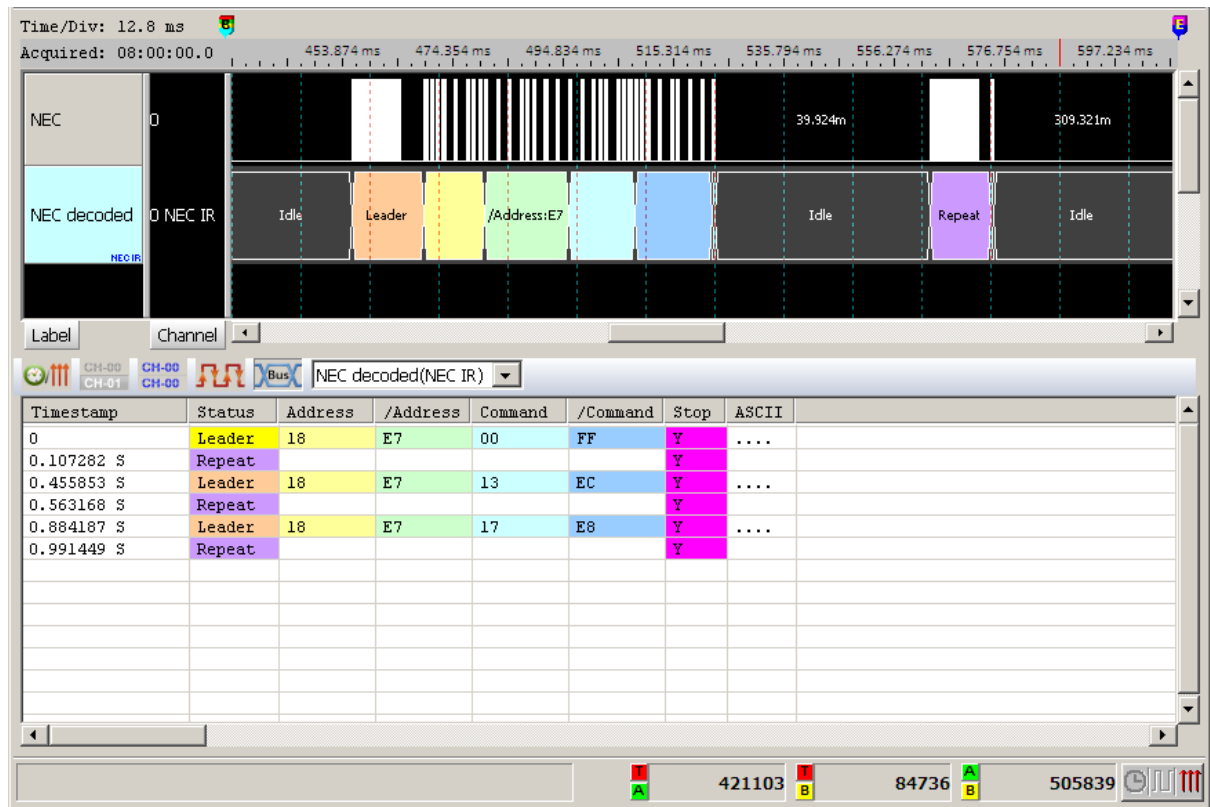
**Auto:** Shows High or Low when auto detection Idle.

**Idle high:** Idle condition shows High.

**Idle low:** Idle condition shows Low.

## Result

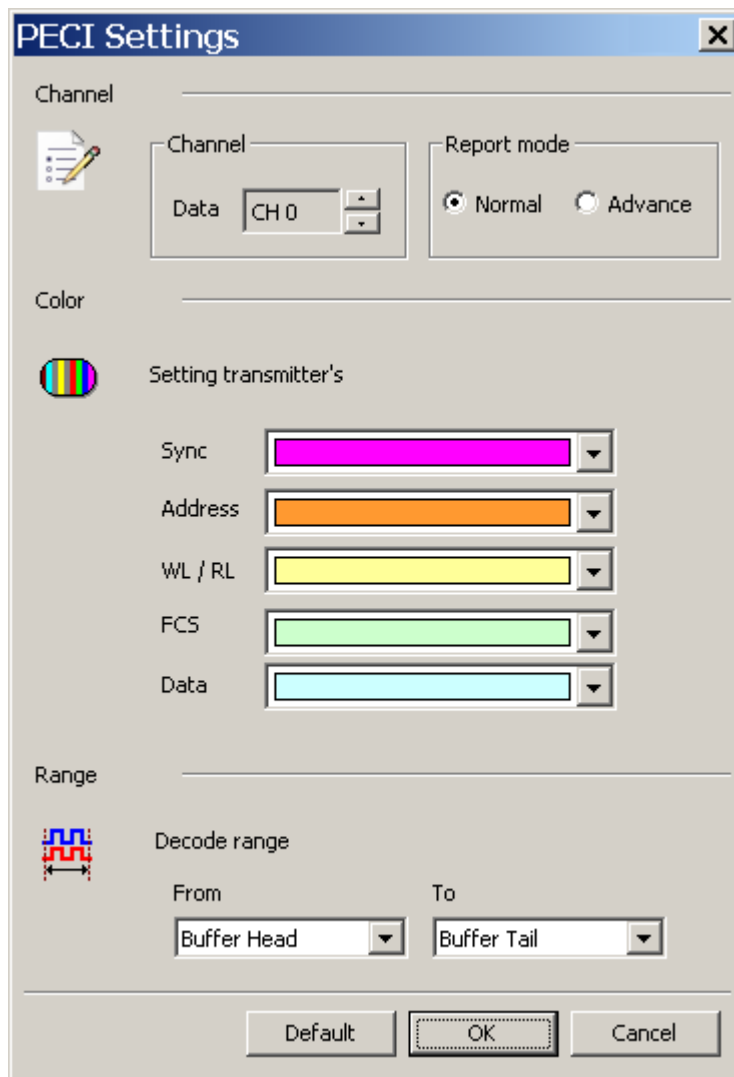
Click **OK** to run the NEC IR decode and see the result on the Waveform Window below.



## PECI

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

### Settings



The image shows a 'PECI Settings' dialog box with a title bar and a close button. It is divided into three main sections: Channel, Color, and Range. The Channel section has a 'Channel' label, a 'Data' dropdown menu set to 'CH 0', and a 'Report mode' section with 'Normal' (selected) and 'Advance' radio buttons. The Color section has a 'Color' label, a 'Setting transmitter's' icon, and five color selection boxes for 'Sync' (magenta), 'Address' (orange), 'WL / RL' (yellow), 'FCS' (light green), and 'Data' (cyan). The Range section has a 'Range' label, a 'Decode range' icon, and two dropdown menus for 'From' (set to 'Buffer Head') and 'To' (set to 'Buffer Tail'). At the bottom are 'Default', 'OK', and 'Cancel' buttons.

**Channel**

Channel

Data CH 0

Report mode

☒ Normal ☐ Advance

**Color**

Setting transmitter's

Sync

Address

WL / RL

FCS

Data

**Range**

Decode range

From To

Buffer Head Buffer Tail

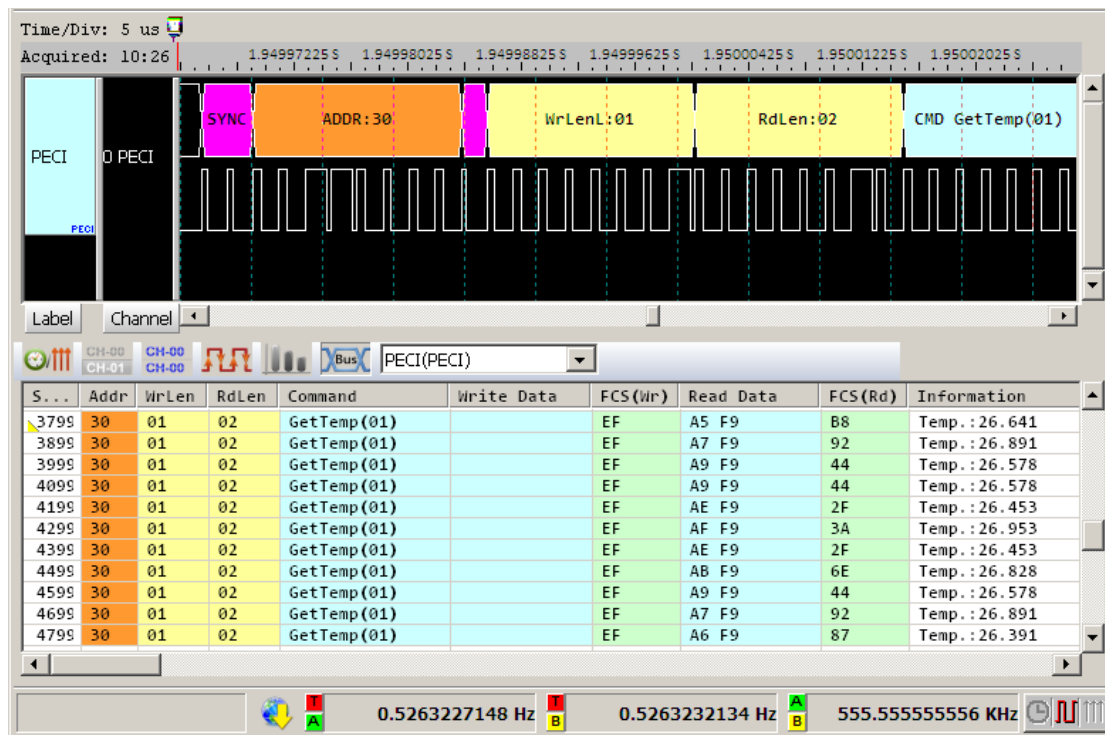
Default OK Cancel

**Channel:** Show the selected channels.

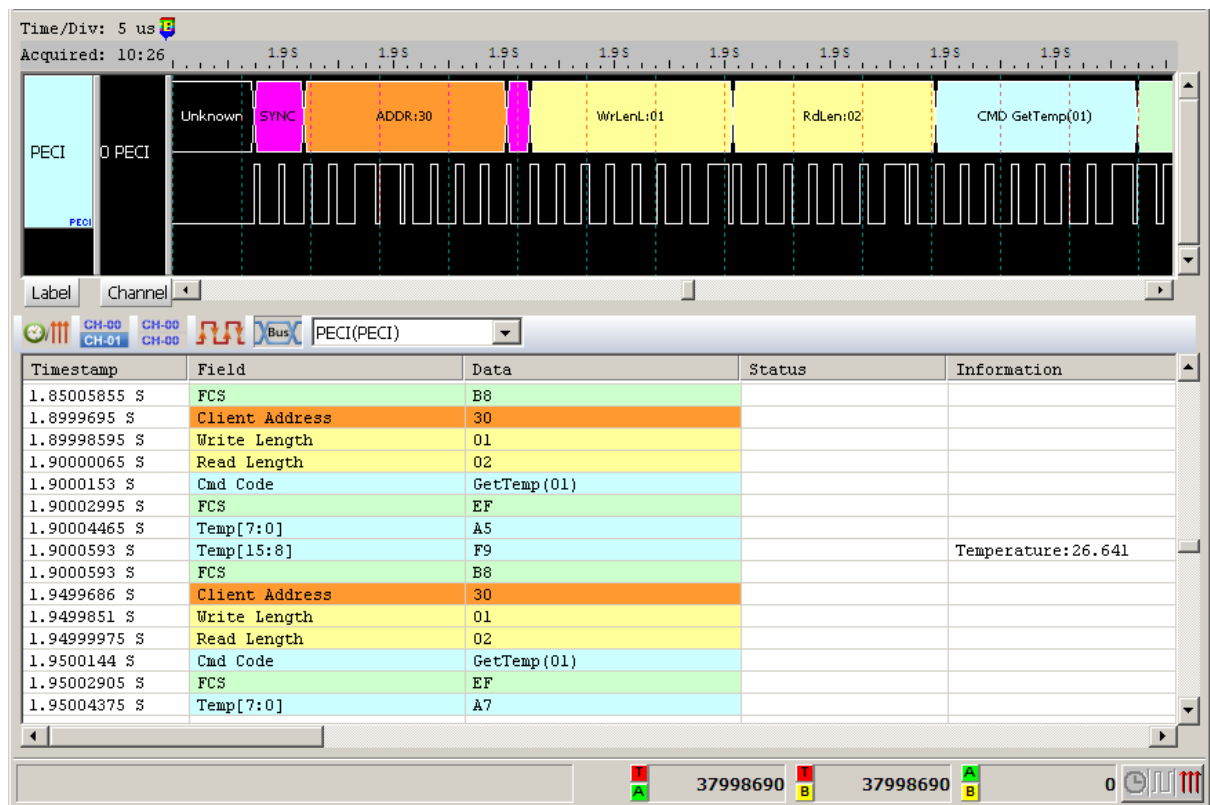
**Report mode:** Normal or Advance

## Result

### Normal mode



### Advance mode

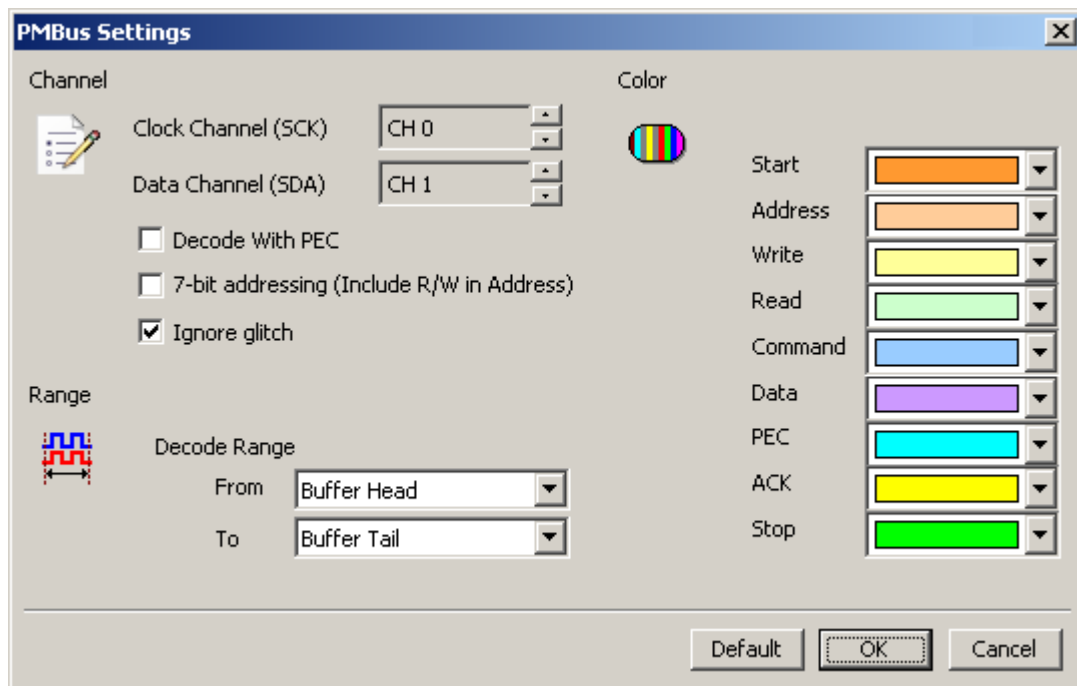




## PMBus

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.

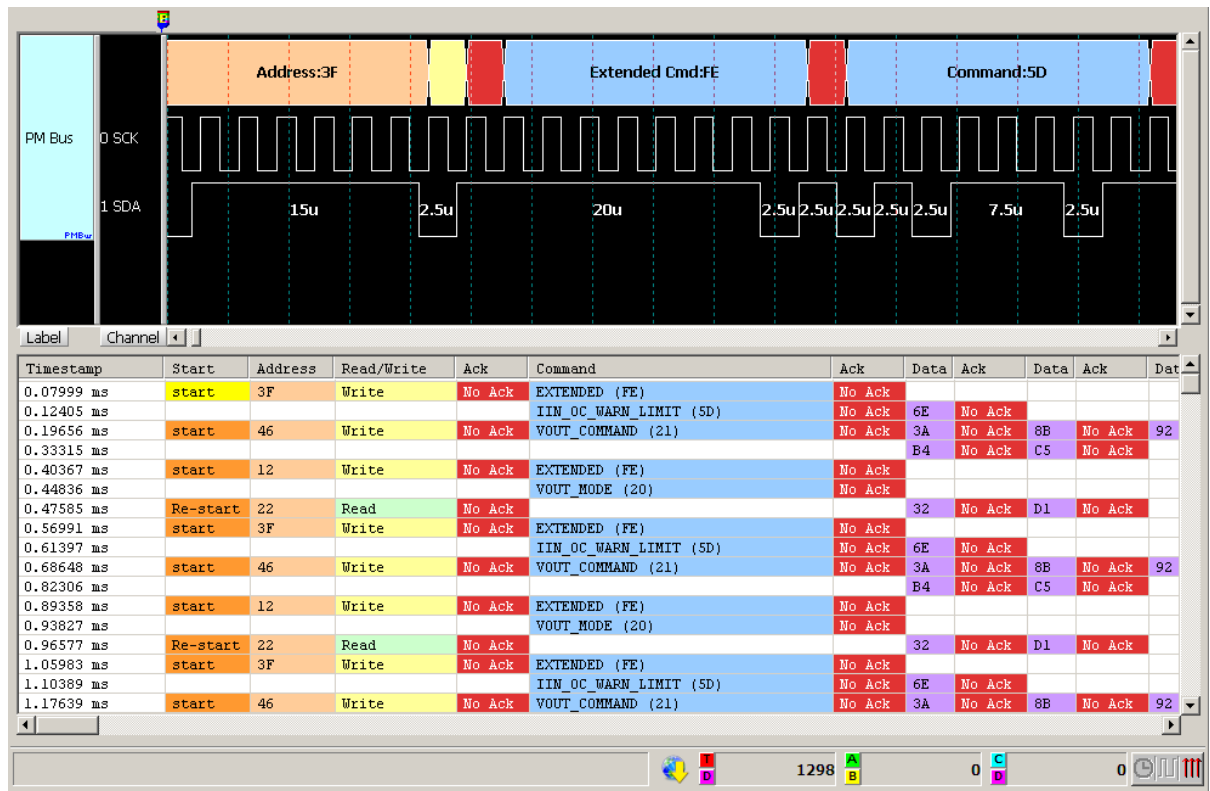
**Decode With PEC:** Group command protocol with PEC.

**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

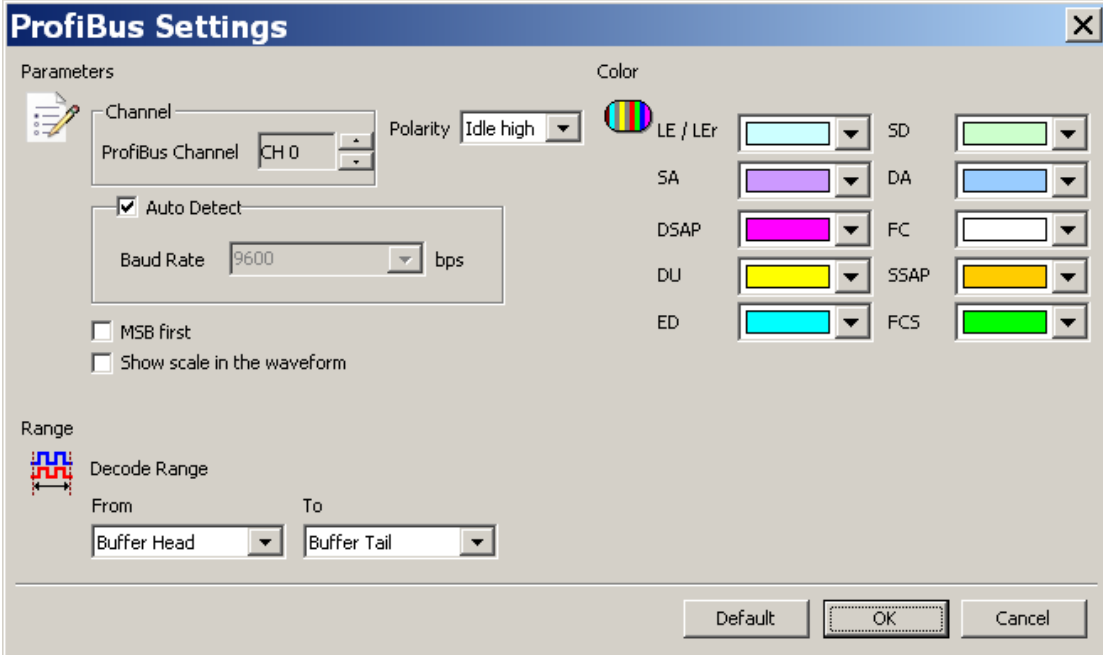
Click **OK** to run the PMBus decode and see the result on the Waveform Window below.



## ProfiBus

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

### Settings



The image shows a 'ProfiBus Settings' dialog box with the following sections and controls:

- Parameters:**
  - Channel:** A dropdown menu showing 'CH 0'.
  - Polarity:** A dropdown menu showing 'Idle high'.
  - Auto Detect:** A checked checkbox.
  - Baud Rate:** A dropdown menu showing '9600' with 'bps' as the unit.
  - MSB first:** An unchecked checkbox.
  - Show scale in the waveform:** An unchecked checkbox.
- Color:** A section with a color wheel icon and a table of color-coded fields:

Field	Color
LE / LEr	Cyan
SA	Purple
DSAP	Magenta
DU	Yellow
ED	Light Blue
SD	Light Green
DA	Blue
FC	White
SSAP	Orange
FCS	Green
- Range:**
  - Decode Range:** A section with a waveform icon.
  - From:** A dropdown menu showing 'Buffer Head'.
  - To:** A dropdown menu showing 'Buffer Tail'.

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

**Channel:** Set the ProfiBus Channel

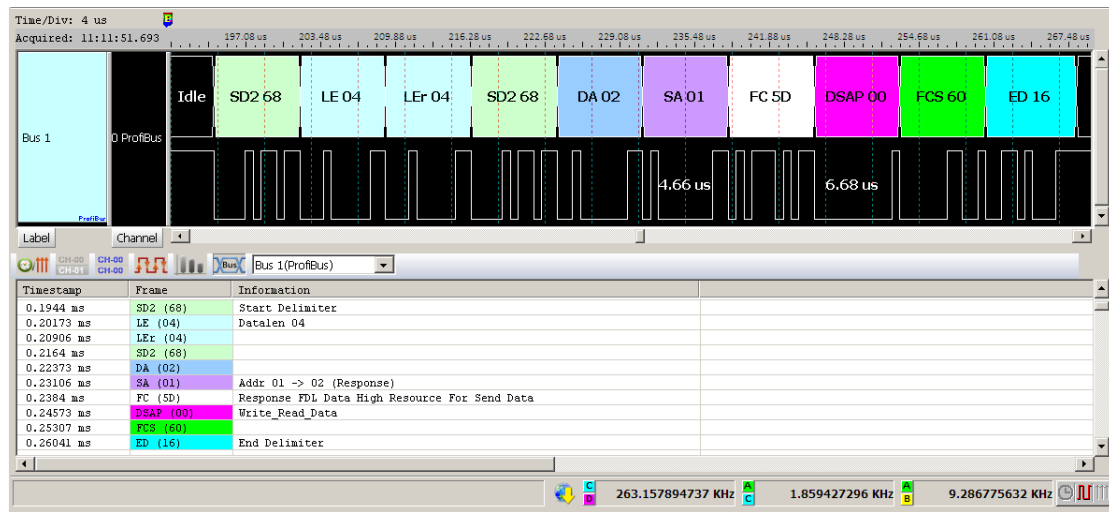
**Polarity:** Set the polarity Idle high / Idle low

**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

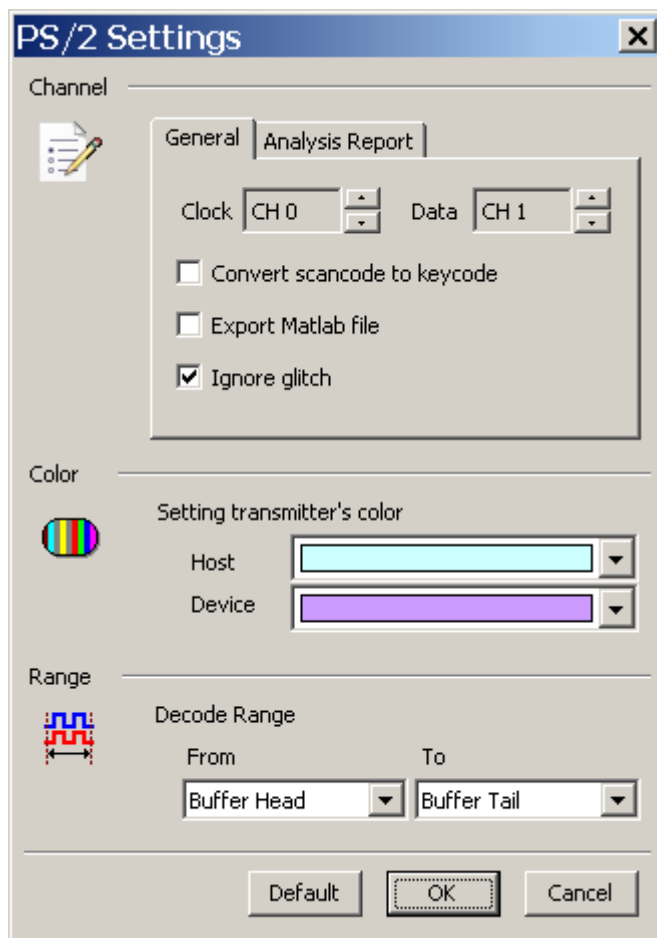
## Result



## 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 ... ]

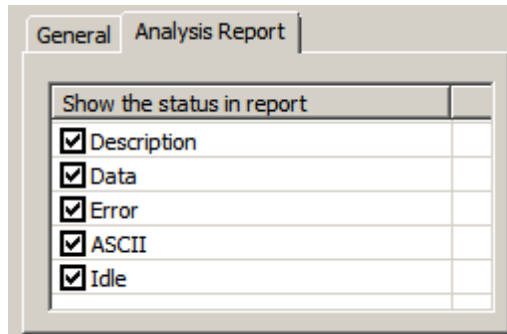
☐ DH = Device to Host, HD = Host to Device

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

The file (PS2\_Matlab.m) will be saved at work directory

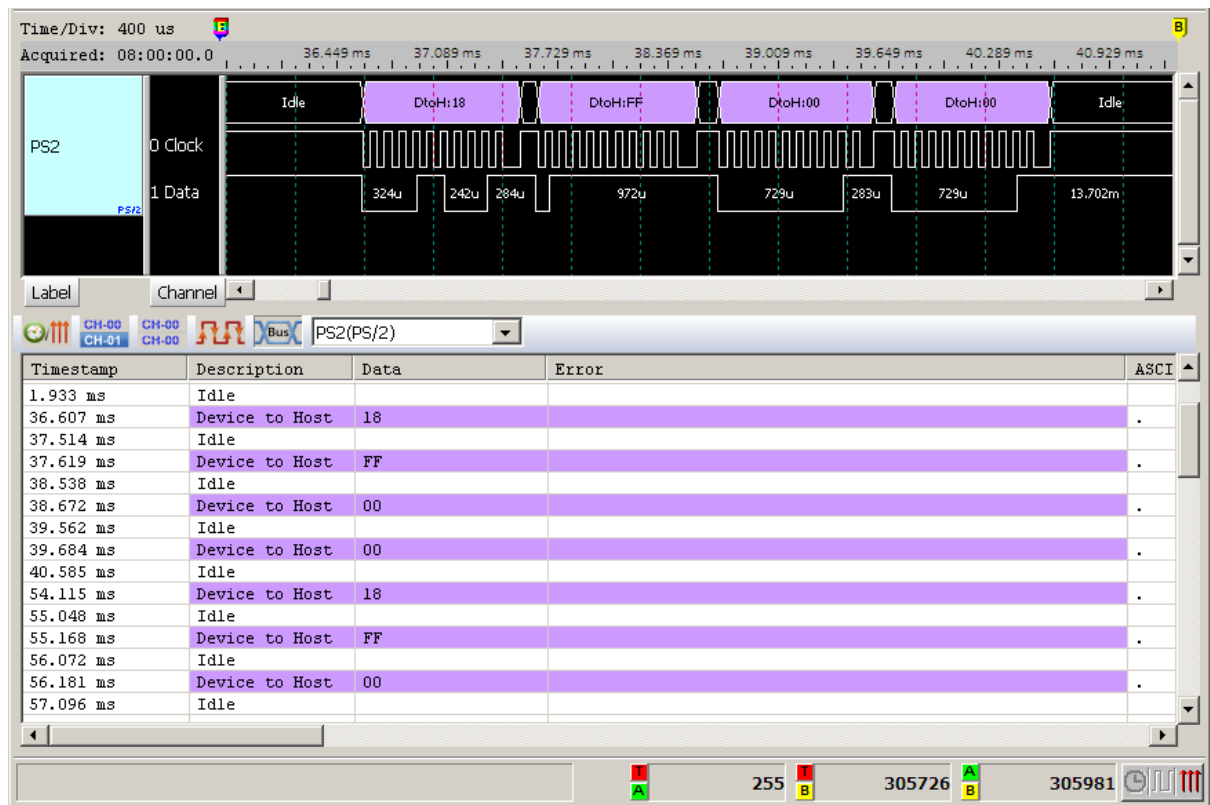
**Ignore glitch:** Ignore the glitch when the slow transitions.

**Analysis Report:** Show the selected status in report.



## Result

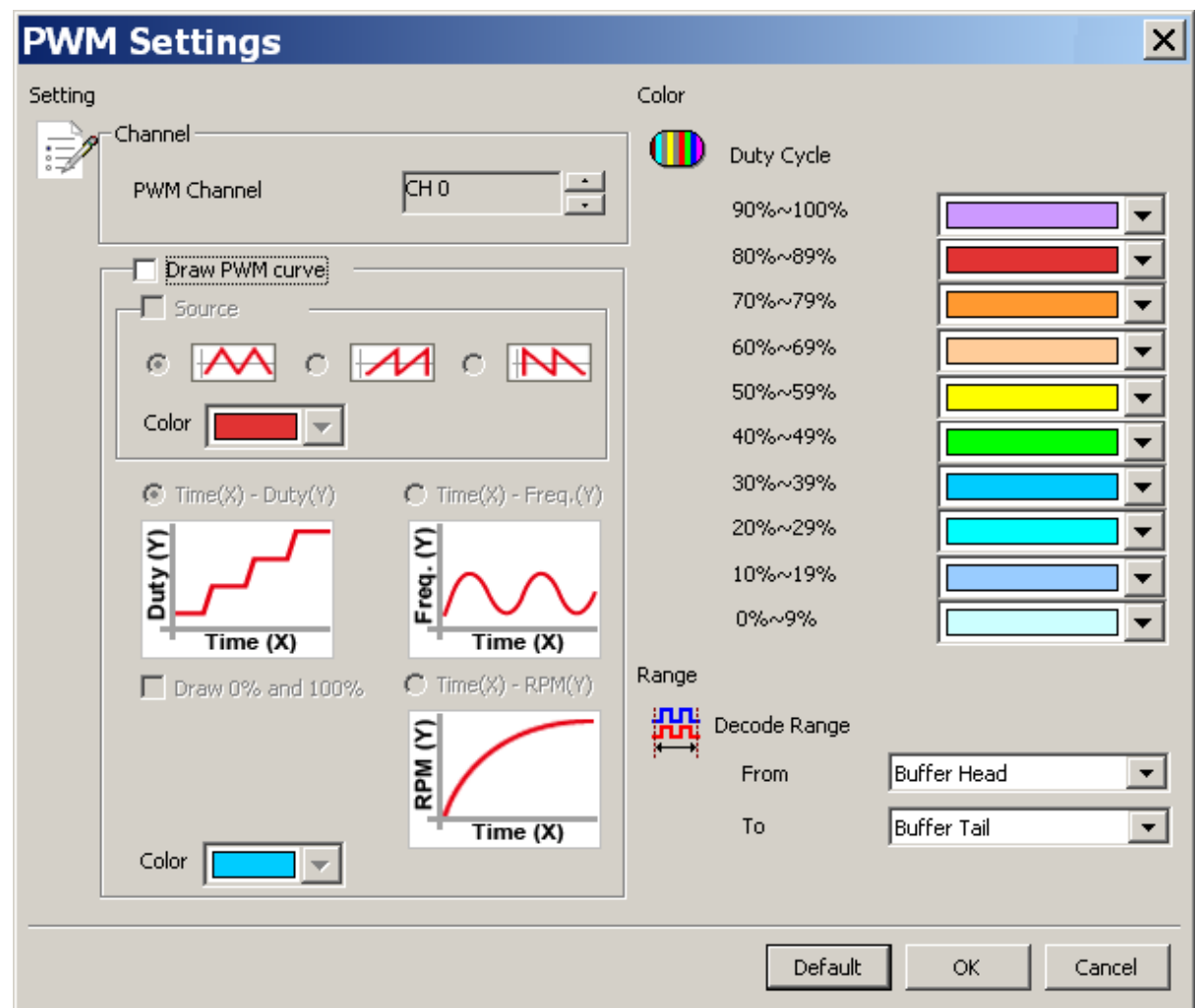
Click **OK** to run the PS/2 decode and see the result on the Waveform Window below.



## 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)

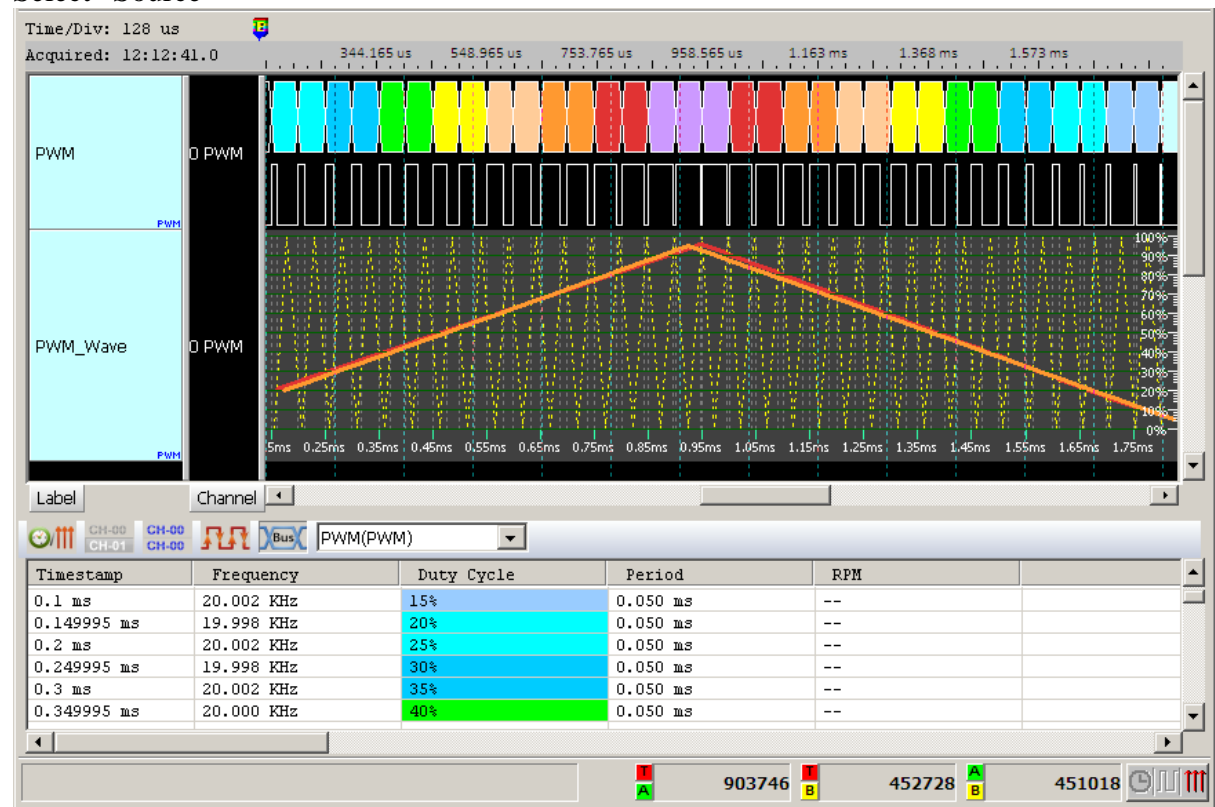
**Draw 0% and 100%:** When select the Time(X)-Duty(Y) drawing and check Draw 0% and 100%, the program will draw this duty curve of 0% or 100%; it will draw this duty curve of 0% or 100% when uncheck Draw 0% and 100%.

**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

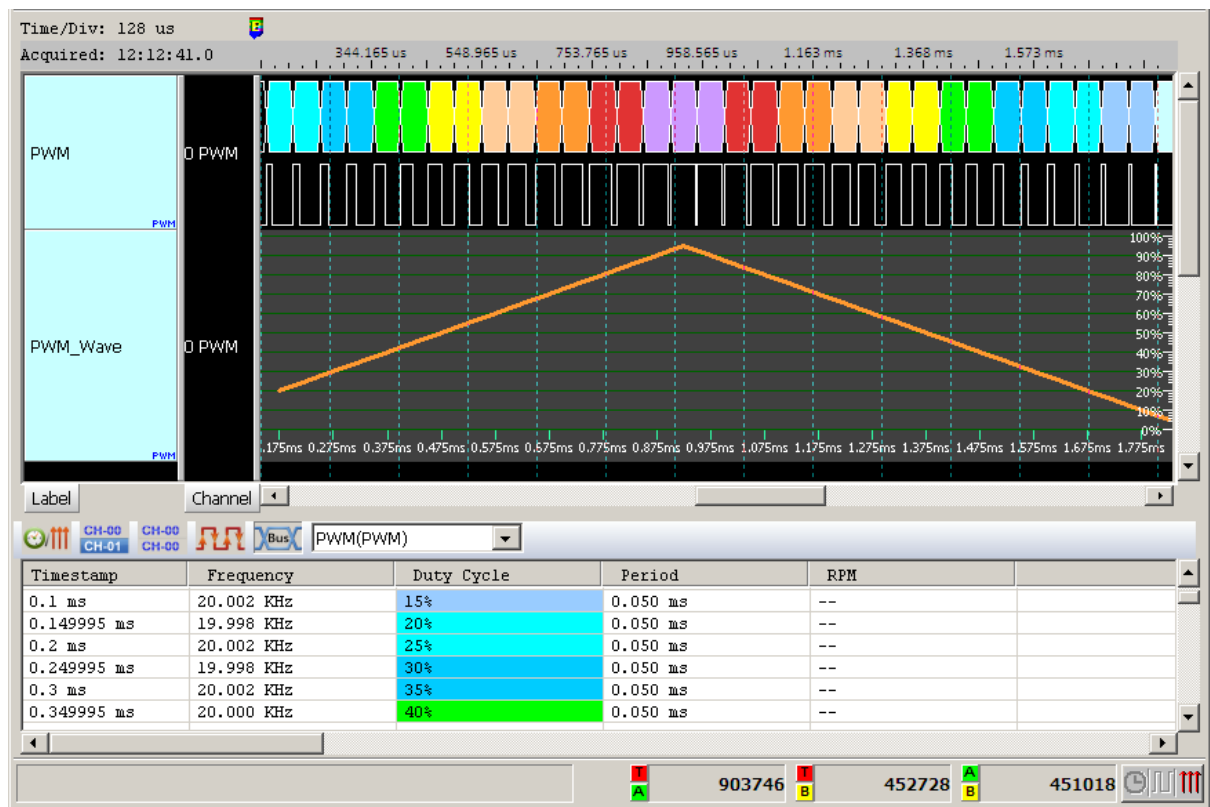
Click **OK** to run the PWM decode and see the result on the Waveform Window below.

Select “Source”

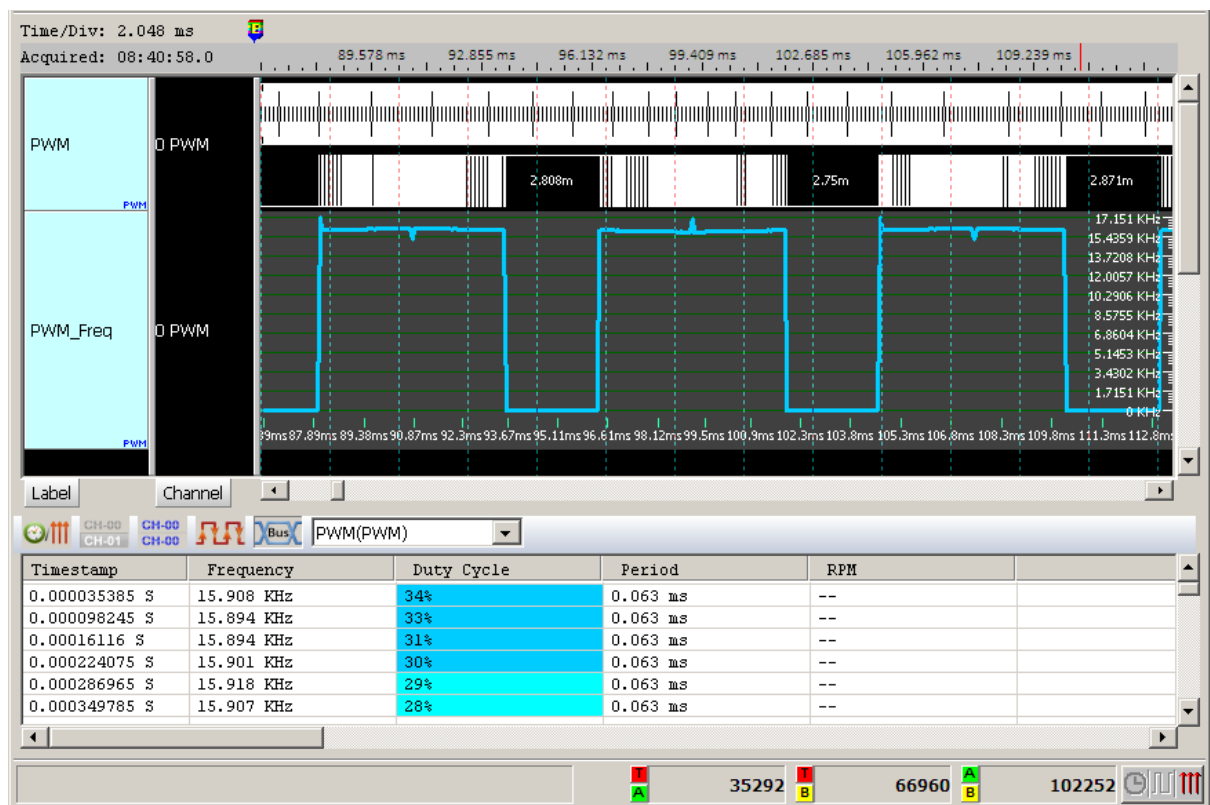




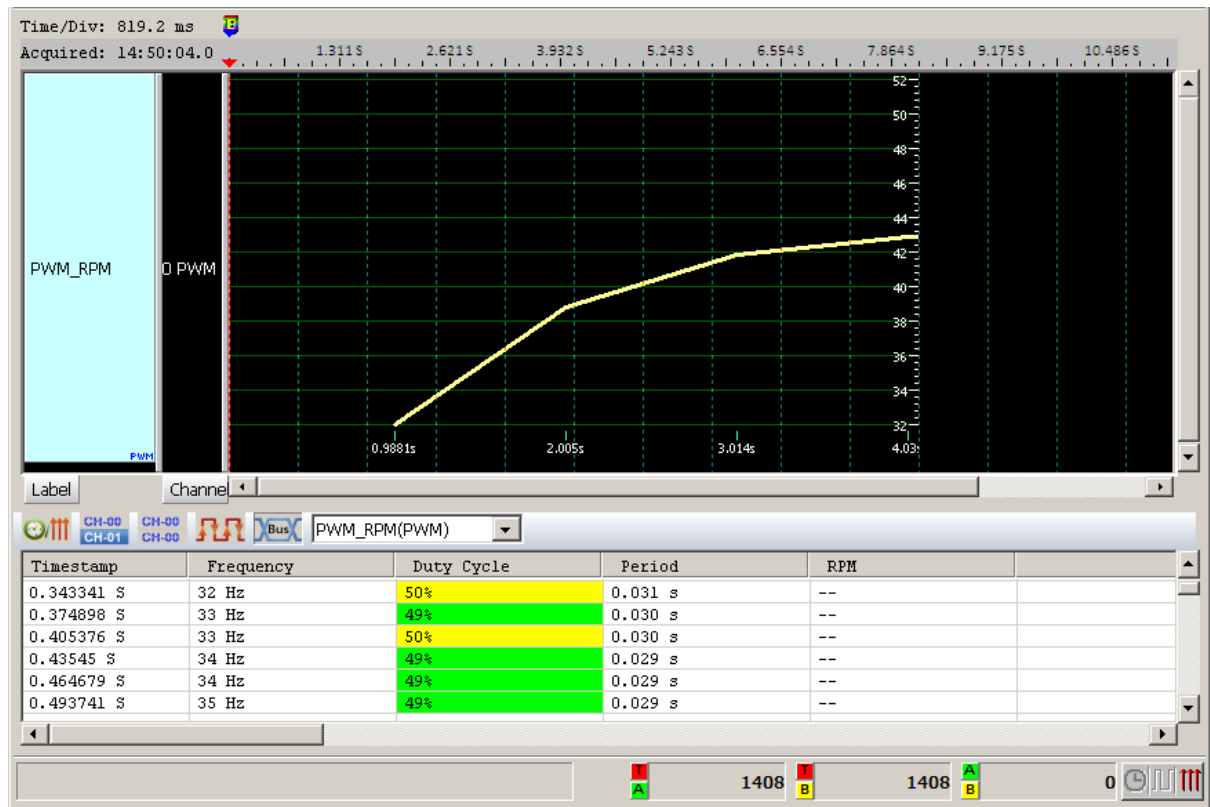
## Select Time(X)-Duty(Y)



## Select Time(X)-Freq.(Y)



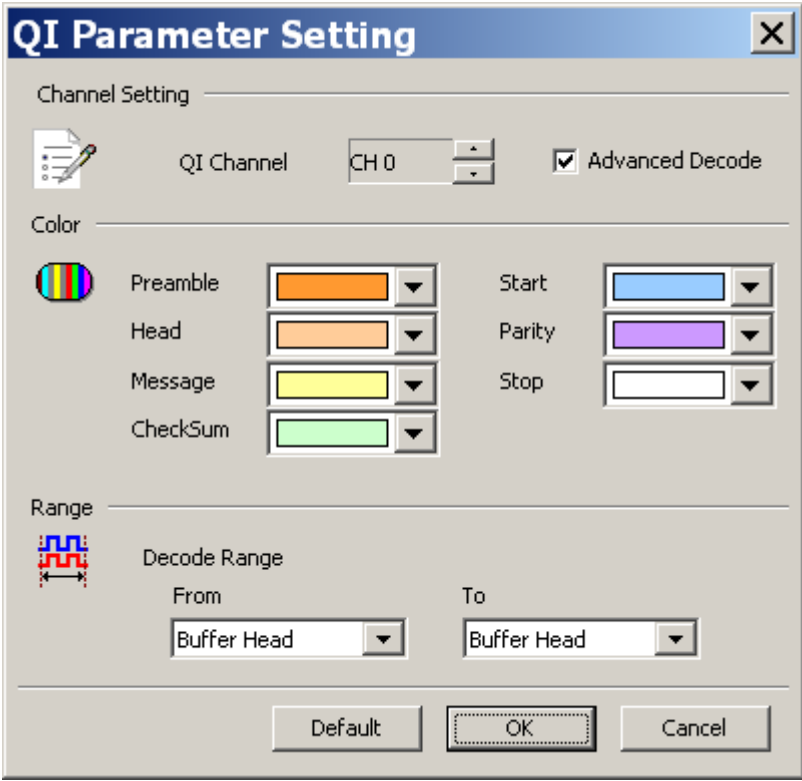
## Select Time(X)-RPM(Y)



## 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

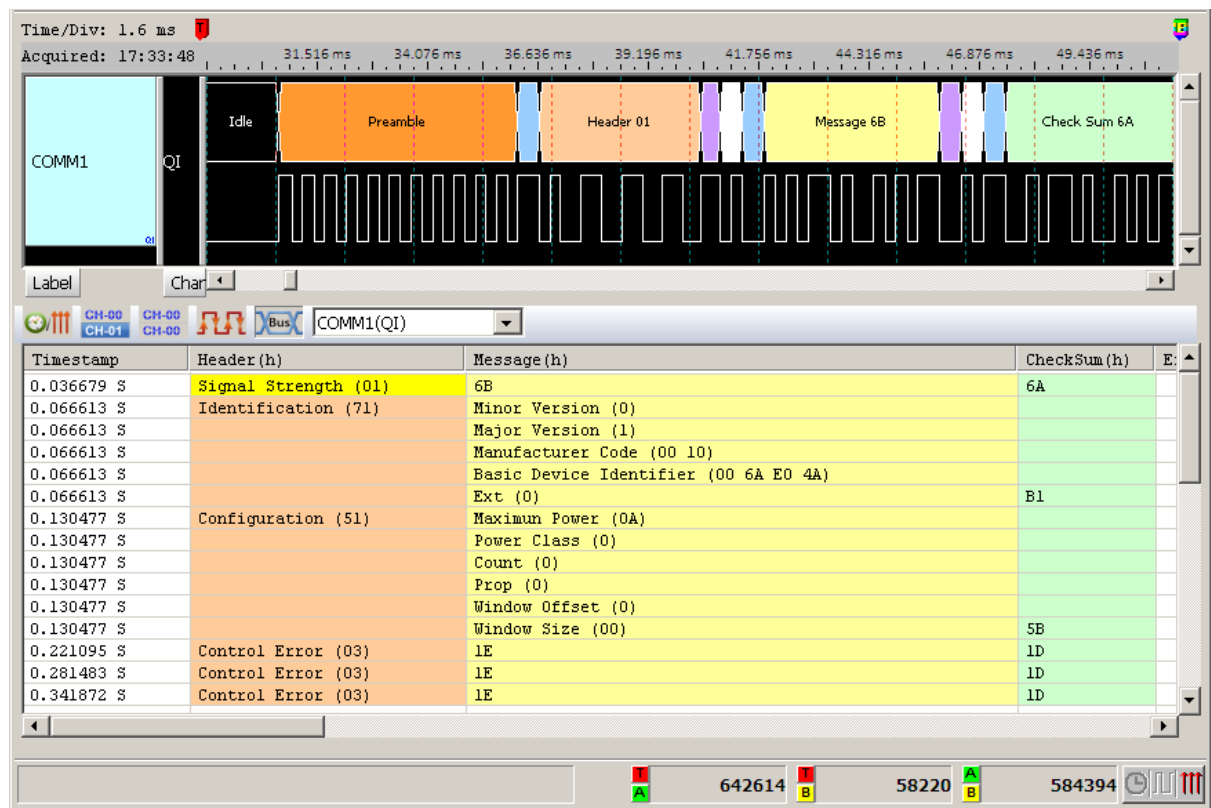


The image shows a 'QI Parameter Setting' dialog box with a blue title bar and a close button. It is divided into three main sections: 'Channel Setting', 'Color', and 'Range'.  
 - **Channel Setting:** Includes a 'QI Channel' dropdown menu set to 'CH 0' and a checked 'Advanced Decode' checkbox.  
 - **Color:** Features a color selection icon and two columns of color pickers. The left column is for 'Preamble' (orange), 'Head' (light orange), 'Message' (yellow), and 'CheckSum' (green). The right column is for 'Start' (blue), 'Parity' (purple), and 'Stop' (white).  
 - **Range:** Includes a range selection icon and a 'Decode Range' section with 'From' and 'To' dropdown menus, both set to 'Buffer Head'.  
 At the bottom are three buttons: 'Default', 'OK', and 'Cancel'.

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

**Advance Decode:** show detail message decode

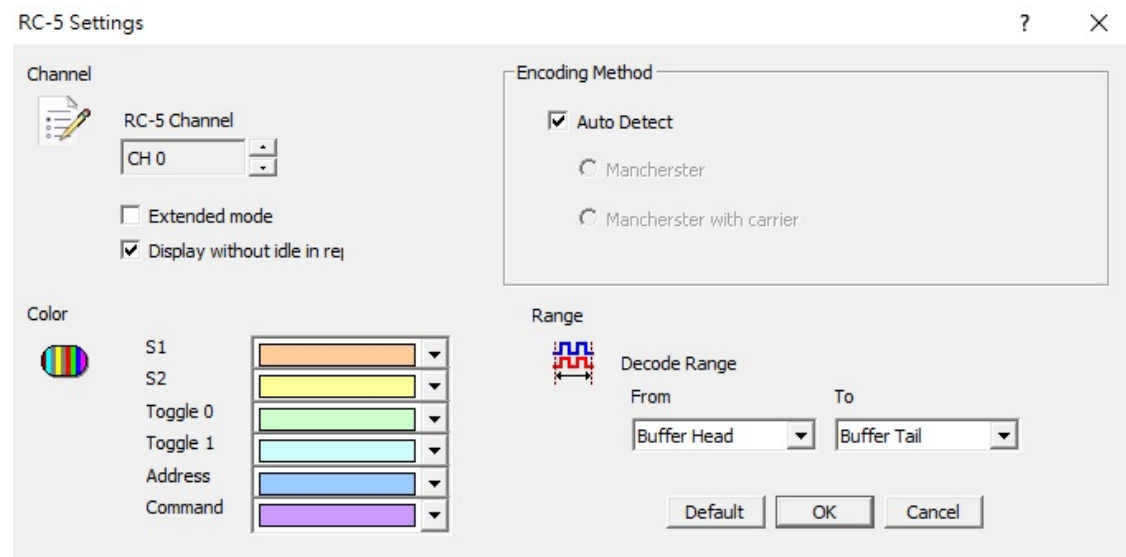
## Result



## 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.

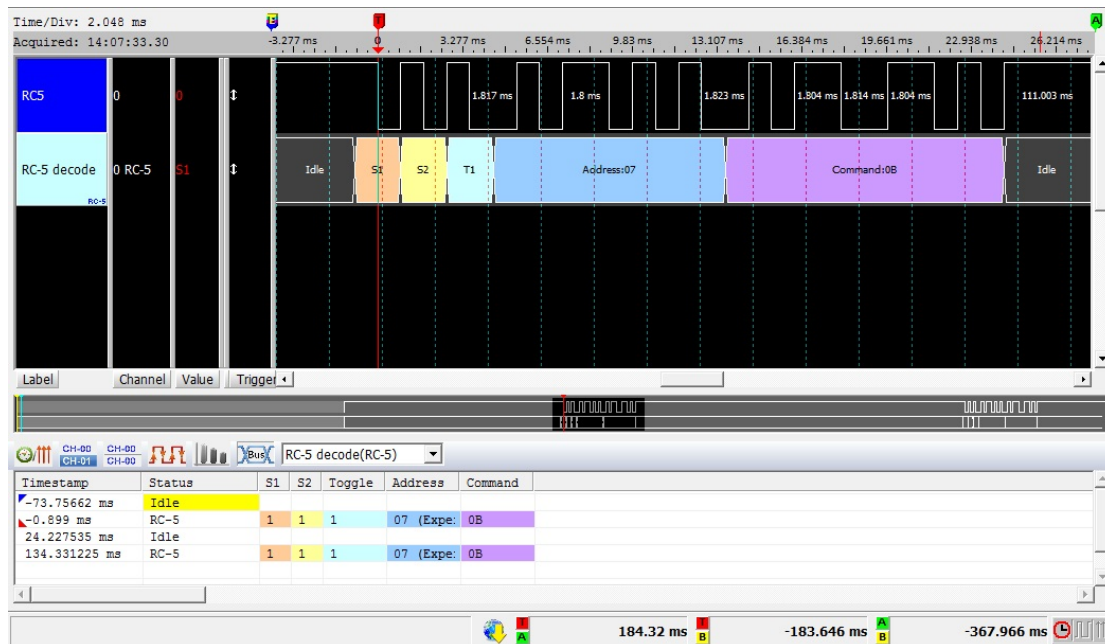
**Address:** To represent different device addresses.

**Command:** To represent the different button commands.

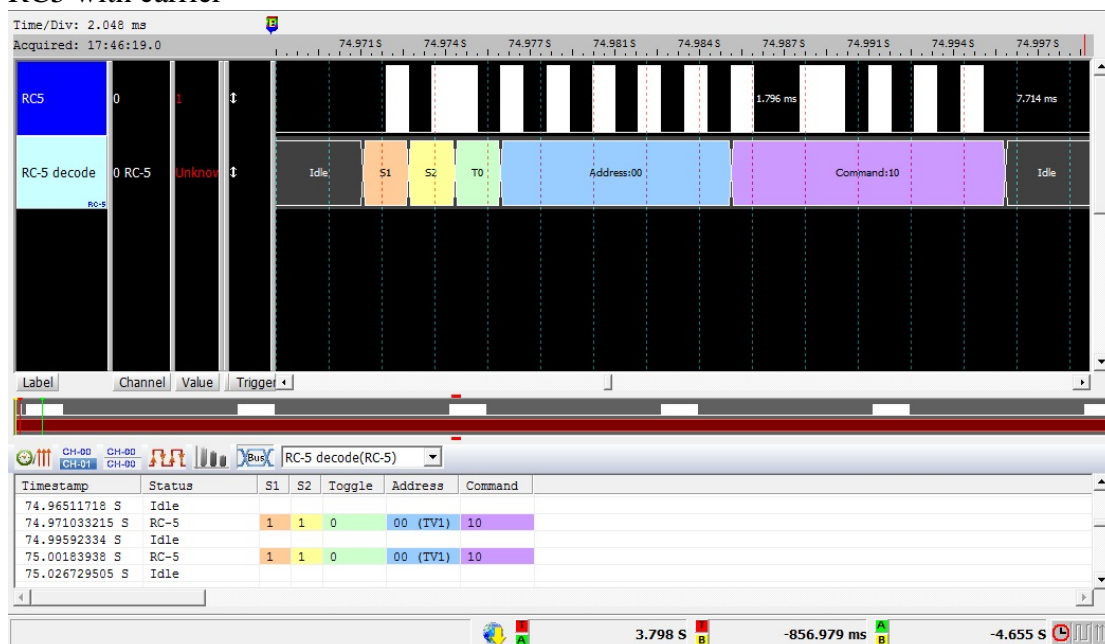
## Result

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

### RC5 without carrier



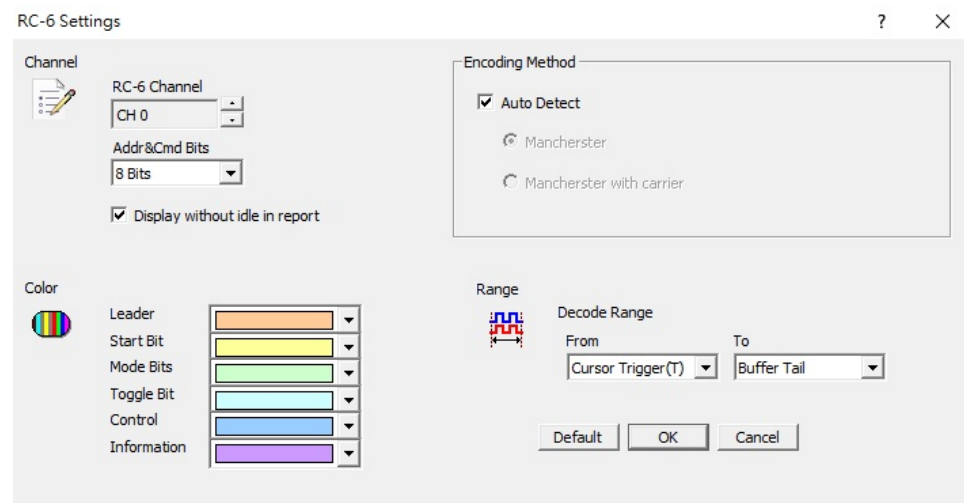
### RC5 with carrier



## 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.

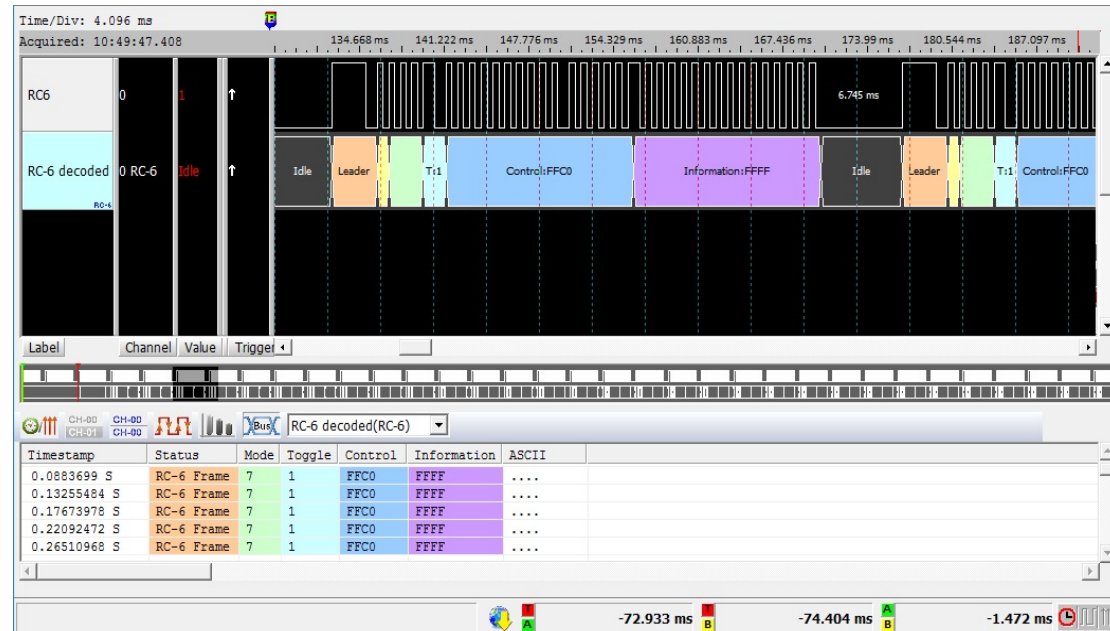
**Encoding Method:** Auto detect mode, Manchester mode, Manchester with carrier mode.



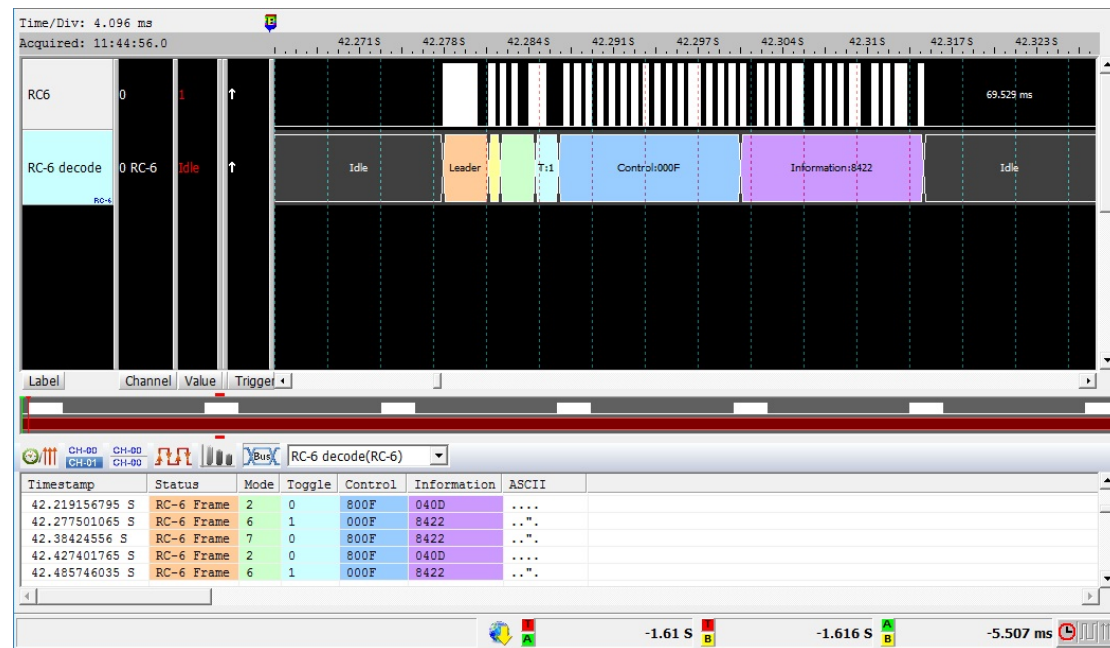
## Result

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	CH 17	R0	CH 14	G0	CH 2	B0	CH 16
DE	CH 17	R1	CH 5	G1	CH 17	B1	CH 17
Hsync	CH 12	R2	CH 3	G2	CH 9	B2	CH 17
VSYNC	CH 1	R3	CH 4	G3	CH 10	B3	CH 0
		R4	CH 6	G4	CH 13	B4	CH 7
		R5	CH 17	G5	CH 15	B5	CH 8
		R6	CH 0	G6	CH 0	B6	CH 0
		R7	CH 0	G7	CH 0	B7	CH 0

**Format**

User defined ☐ Save as JPG file

A (Alpha) R (Red) G (Green) B (Blue) L (Luminance)

0 bits 6 bits 6 bits 6 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.

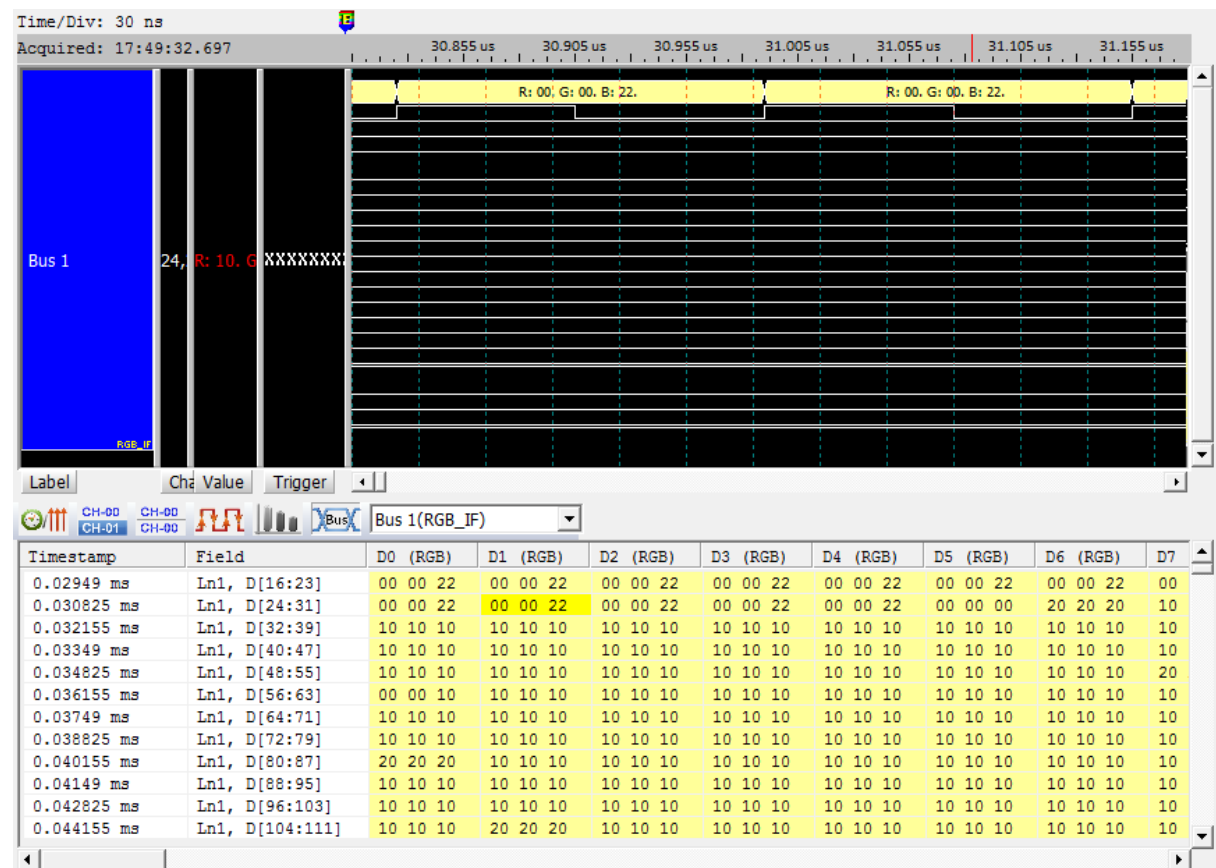
**Vsync:** The Vertical synchronization pin.

**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.

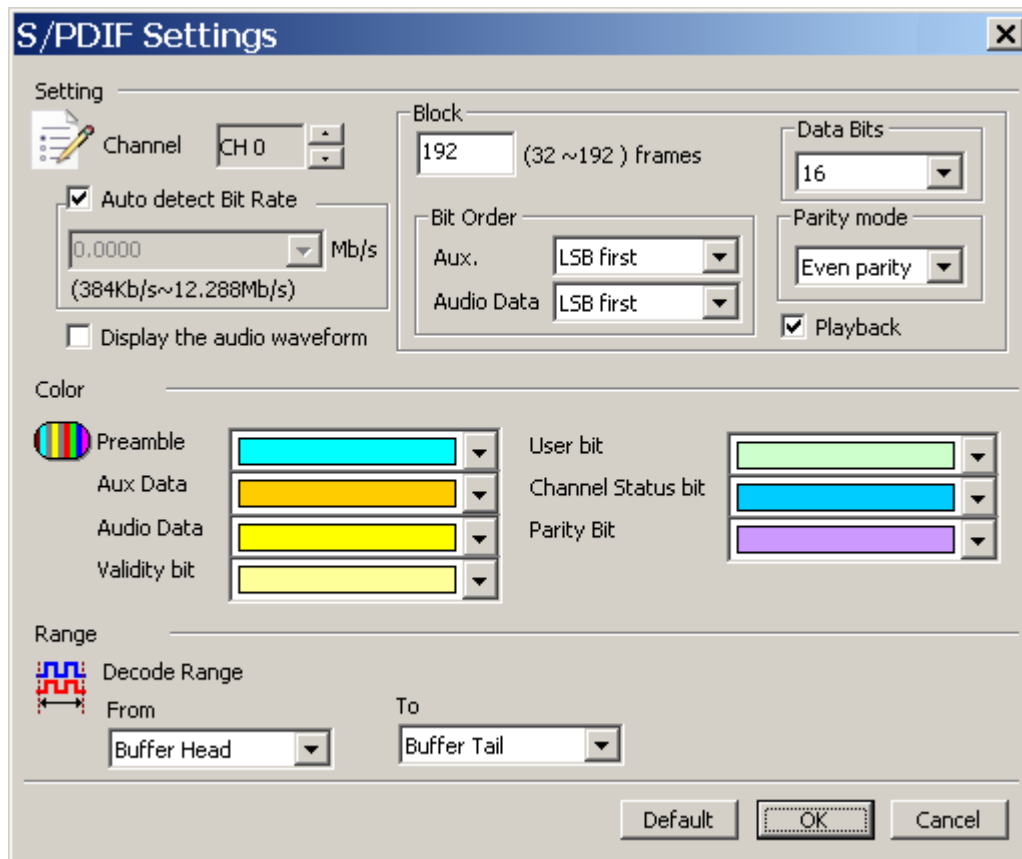
**U (User) bit:** User log information.

**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.

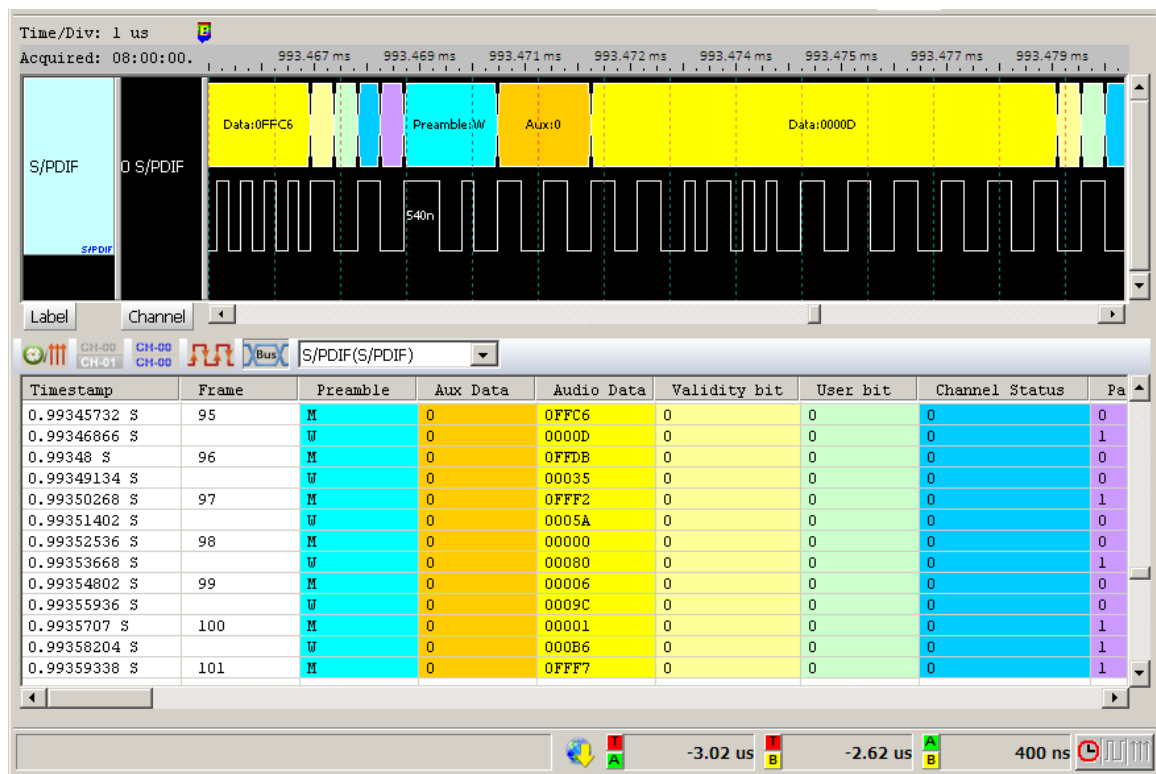
**Parity mode:** The default is even parity.

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

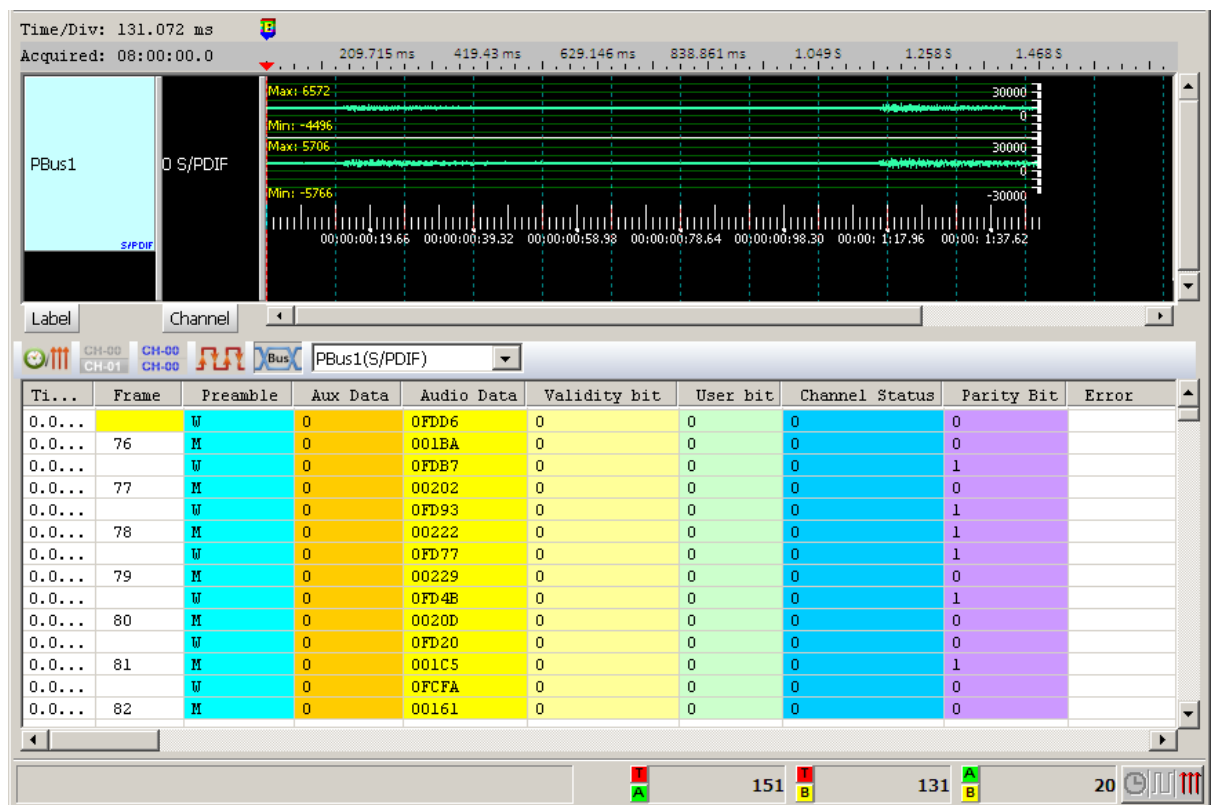
## Result

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

below.



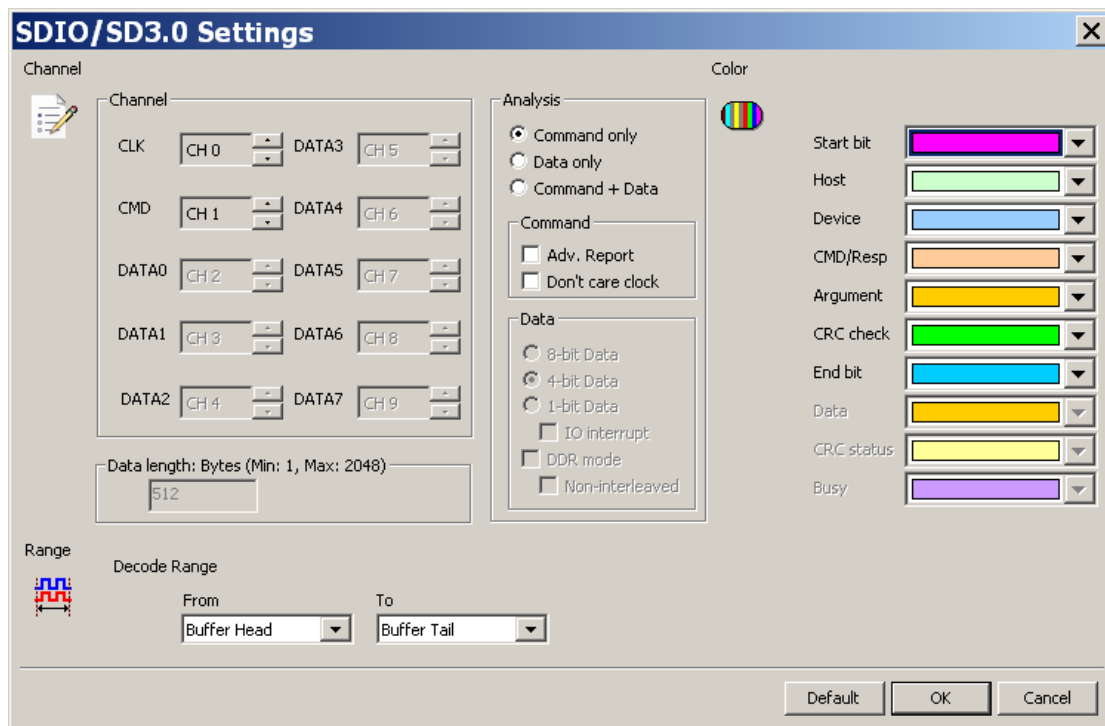
Show wave:



## 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



The dialog box is titled "SDIO/SD3.0 Settings". It is divided into several sections:

- Channel:** Contains a grid of channel selection buttons. The first column has CLK (CH 0), CMD (CH 1), DATA0 (CH 2), DATA1 (CH 3), and DATA2 (CH 4). The second column has DATA3 (CH 5), DATA4 (CH 6), DATA5 (CH 7), DATA6 (CH 8), and DATA7 (CH 9). Below this grid is a "Data length: Bytes (Min: 1, Max: 2048)" field with the value "512" entered.
- Analysis:** Contains radio buttons for "Command only", "Data only", and "Command + Data". Below these are checkboxes for "Adv. Report" and "Don't care clock". Further down are radio buttons for "8-bit Data", "4-bit Data", and "1-bit Data", followed by checkboxes for "IO interrupt", "DDR mode", and "Non-interleaved".
- Color:** A vertical list of color-coded fields: Start bit (magenta), Host (light green), Device (light blue), CMD/Resp (orange), Argument (yellow), CRC check (bright green), End bit (cyan), Data (gold), CRC status (pale yellow), and Busy (purple).
- Range:** Contains a "Decode Range" section with "From" and "To" dropdown menus. The "From" menu is set to "Buffer Head" and the "To" menu is set to "Buffer Tail".

At the bottom right are buttons for "Default", "OK", and "Cancel".

**Channel:** Show the selected channels.

**Command only:** Analyze the command.

**Data only:** Analyze the data.

**Command + Data:** Analyze Command and Data in the report window.

**Adv. Report:** Analyze the command argument

**Don't care clock:** decode only depend on the CMD channel.

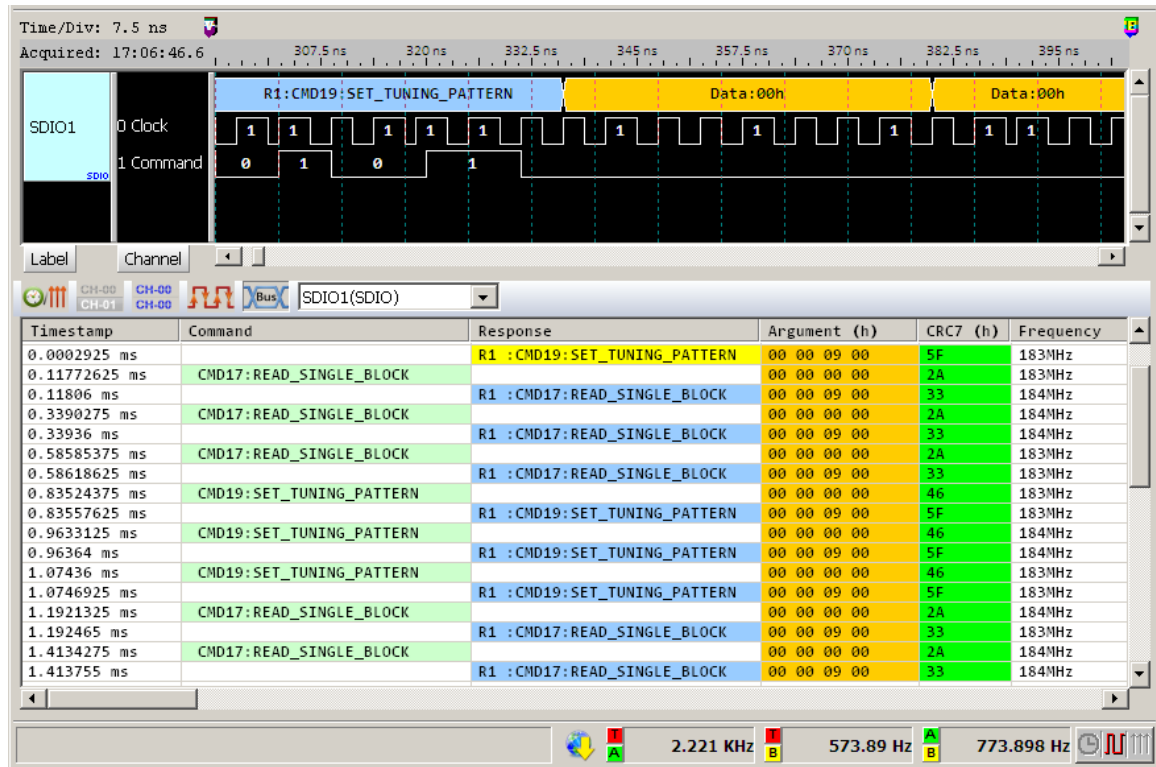
**Data:** 1/4/8 bits or DDR mode, check "IO interrupt" when SDIO 1 bit mode and support the IO interrupt decode via DATA1 channel, check "DDR mode" and "Non-interleaved" to analyze data without interleaved.

**Data length:** Set the length of data.

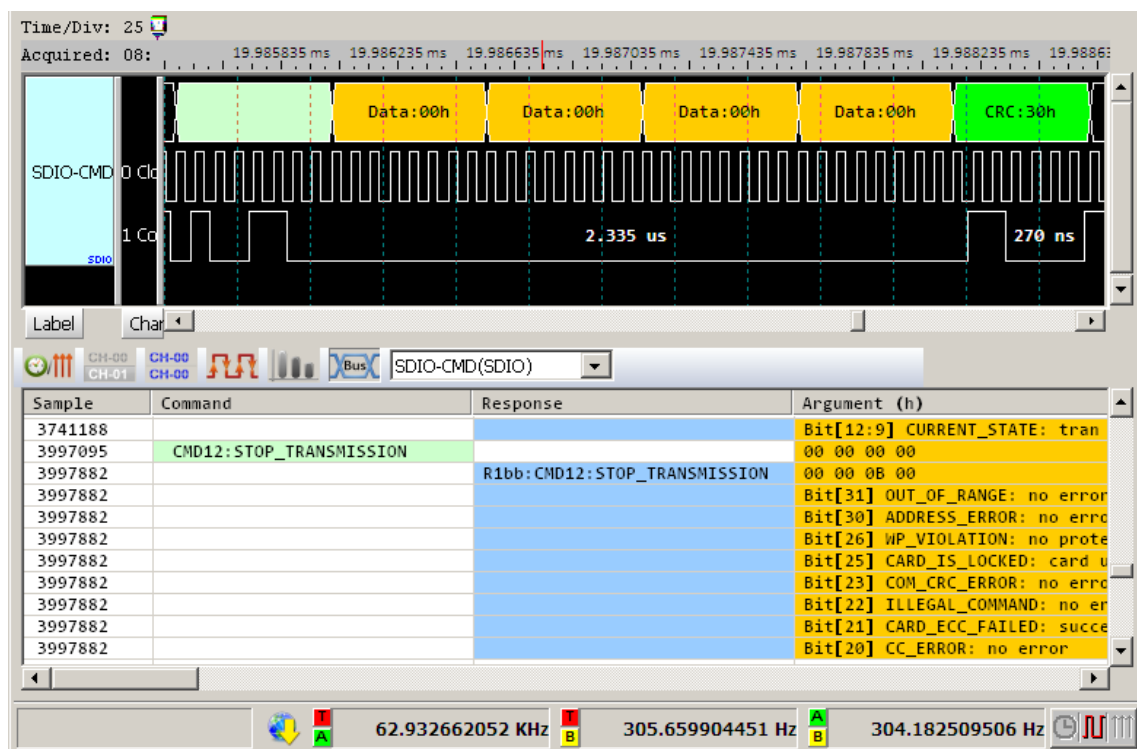
## Result

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

## CMD mode

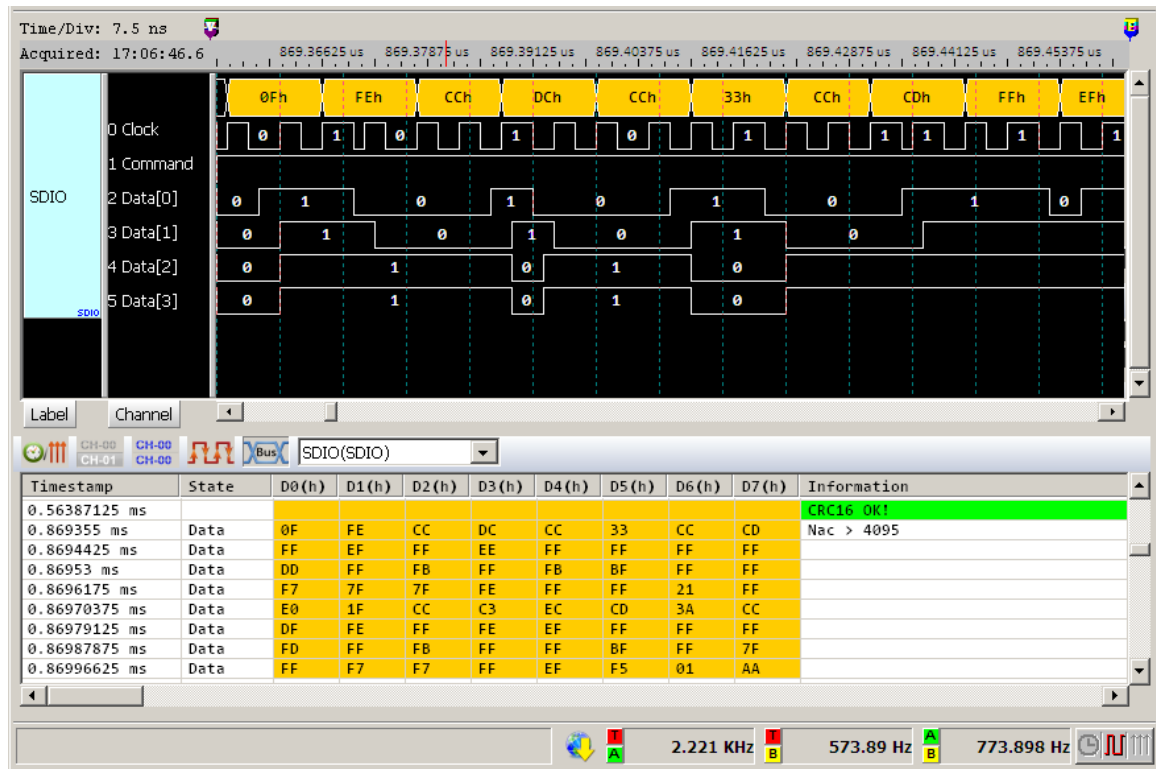


## Adv. Report





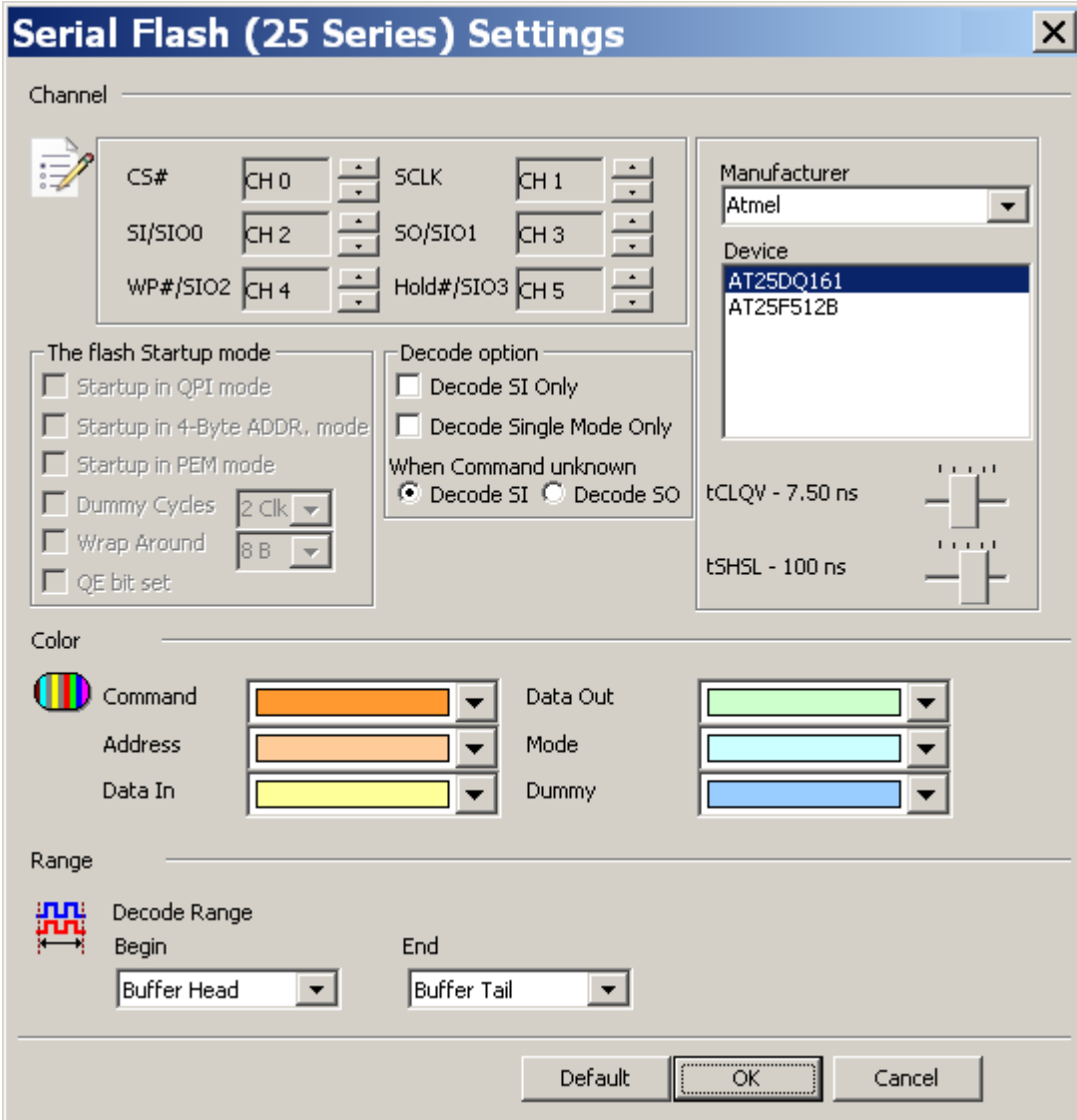
## Data mode



## 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



**Serial Flash (25 Series) Settings**

Channel

CS# CH 0 SCLK CH 1  
SI/SIO0 CH 2 SO/SIO1 CH 3  
WP#/SIO2 CH 4 Hold#/SIO3 CH 5

The flash Startup mode  
☐ Startup in QPI mode  
☐ Startup in 4-Byte ADDR. mode  
☐ Startup in PEM mode  
☐ Dummy Cycles 2 Clk  
☐ Wrap Around 8 B  
☐ QE bit set

Decode option  
☐ Decode SI Only  
☐ Decode Single Mode Only  
 When Command unknown  
☒ Decode SI ☐ Decode SO

Manufacturer  
Atmel

Device  
AT25DQ161  
AT25F512B

tCLQV - 7.50 ns  
tSHSL - 100 ns

Color

Command Address Data In Data Out Mode Dummy

Range

Decode Range  
 Begin End  
 Buffer Head Buffer Tail

Default OK Cancel

**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.

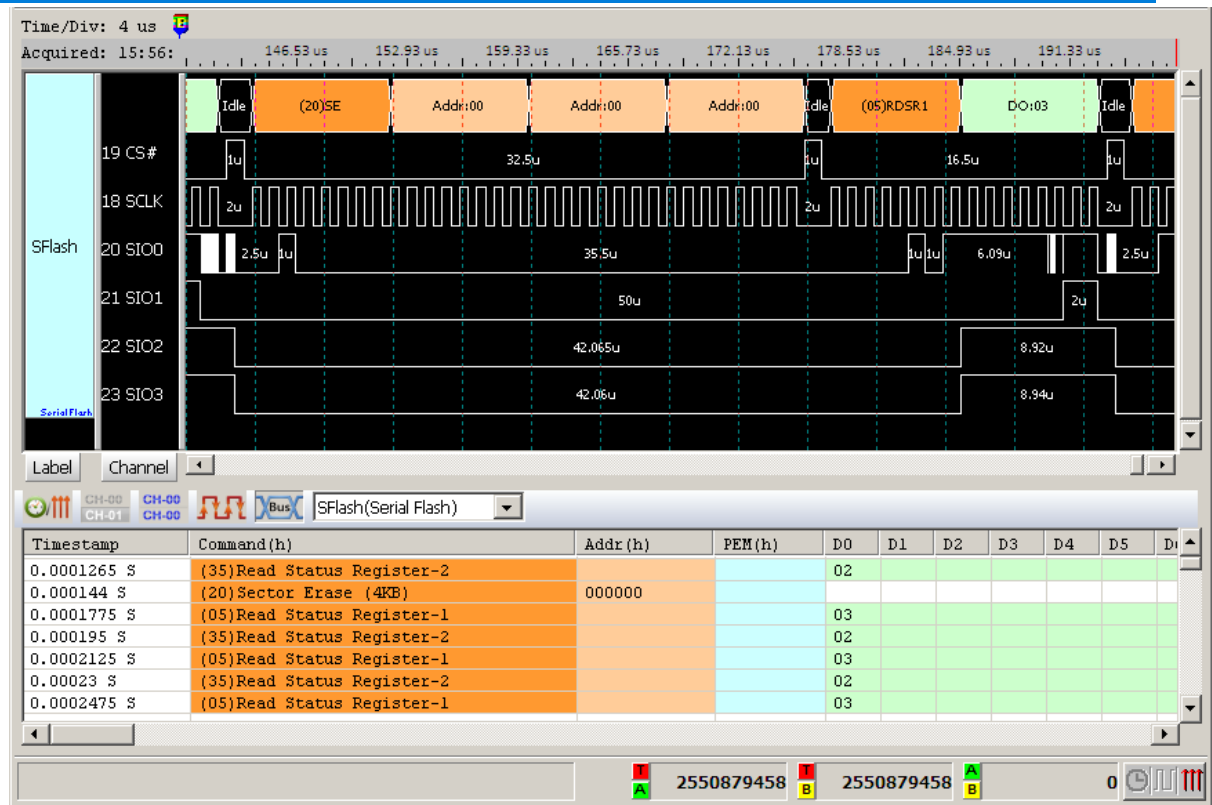
**Decode Single Mode Only:** Single mode, 4-wire → CS, Clock, SI, SO.

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

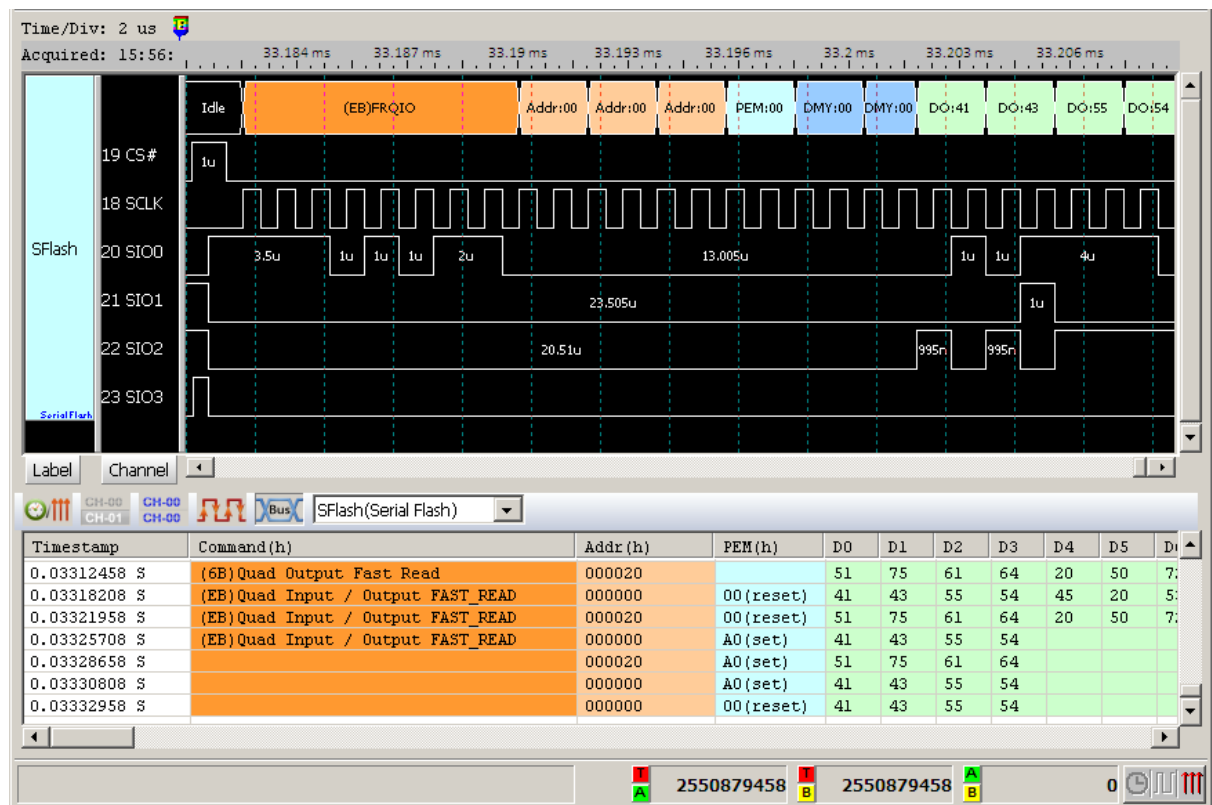
## **Result**

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

Serial Flash decode within SPI mode.



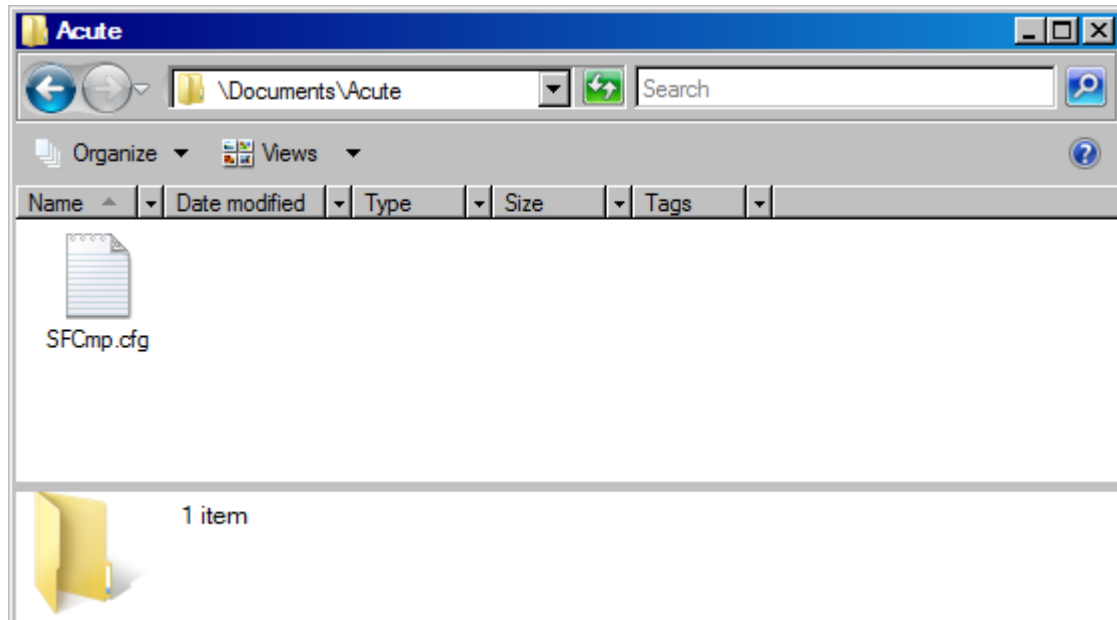
Serial Flash decode within QPI mode.



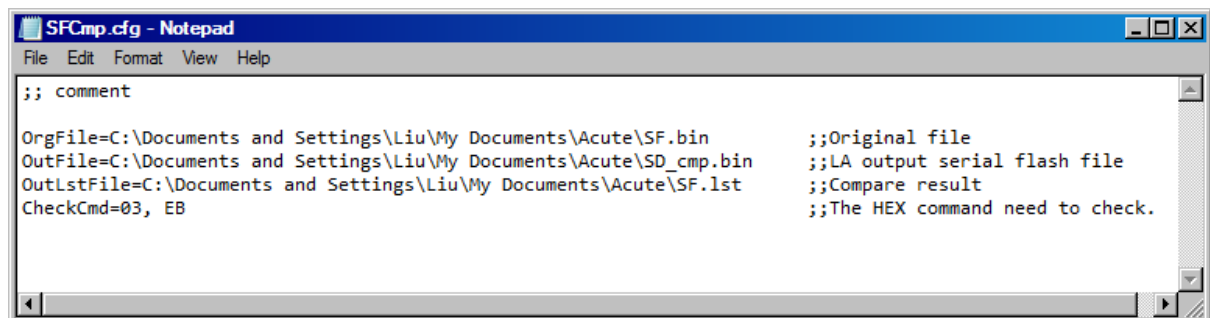
**Serial Flash data Comparison :** Compare the Serial Flash data by the waveform

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:



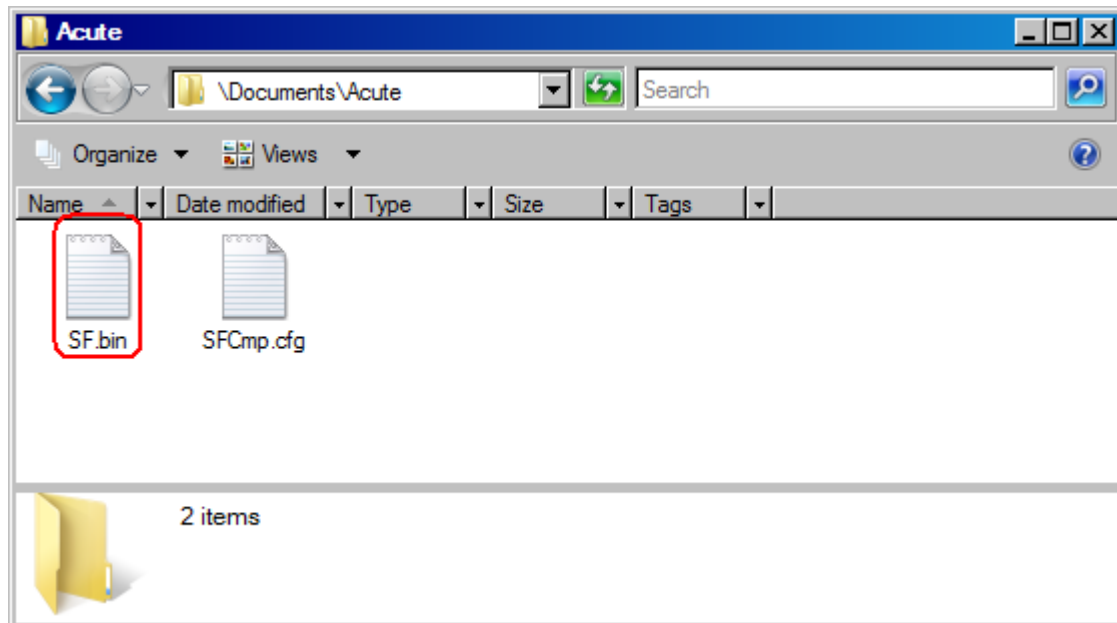
**OrgFile=File\_Path:** Key in the file path of the original Serial Flash data file (.bin).

**OutFile=File\_Path:** Key in the file path of the Serial Flash output file.

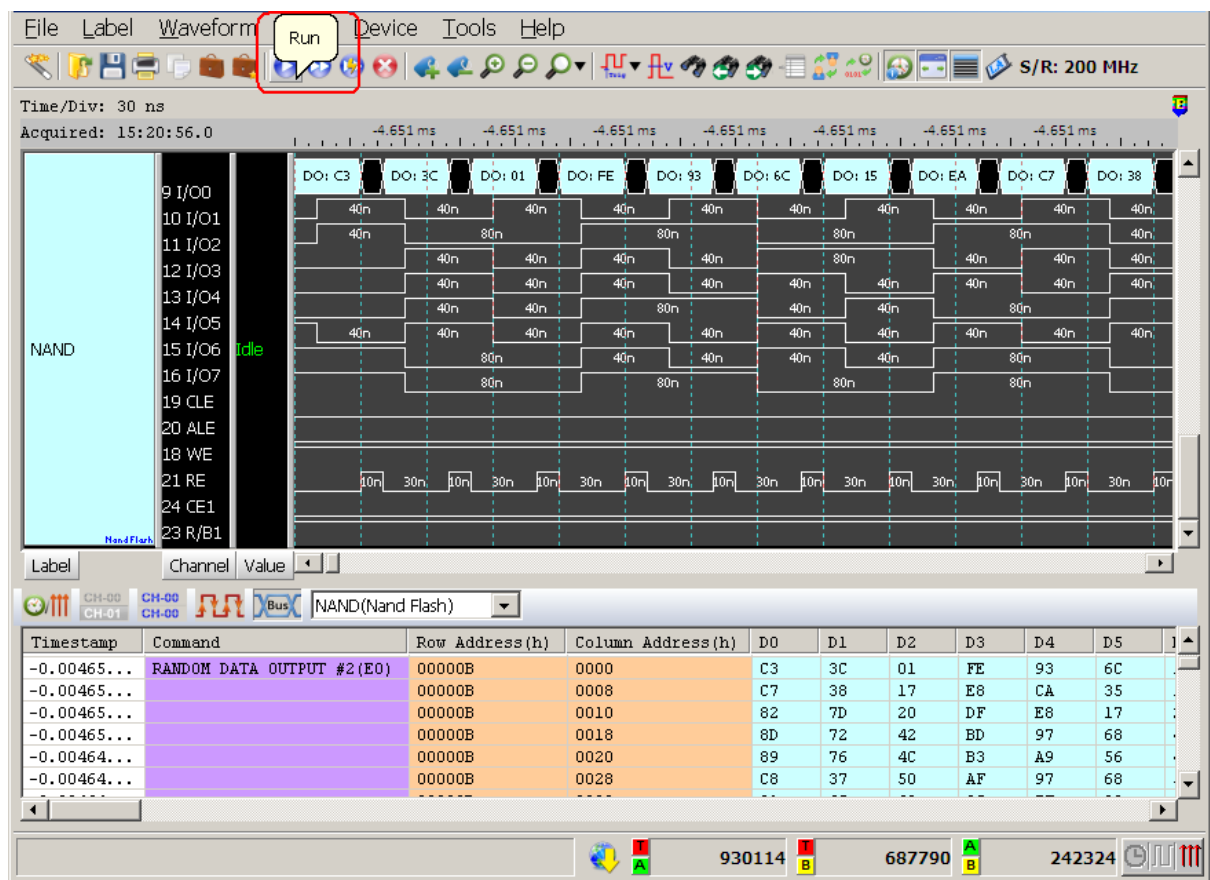
**OutLstFile=File\_Path:** Key in the file path of the comparison result. The file name will has extension “.lst”.

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

Save the OrgFile to the OrgFile file path.



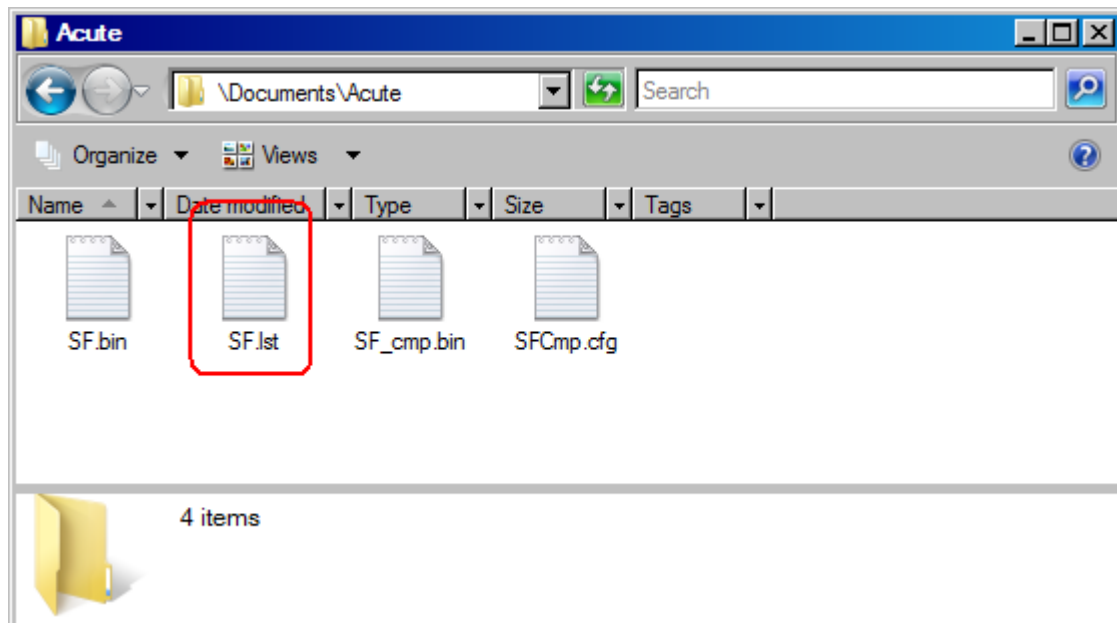
Run the Serial Flash Bus Decode to capture the Serial Flash signal.



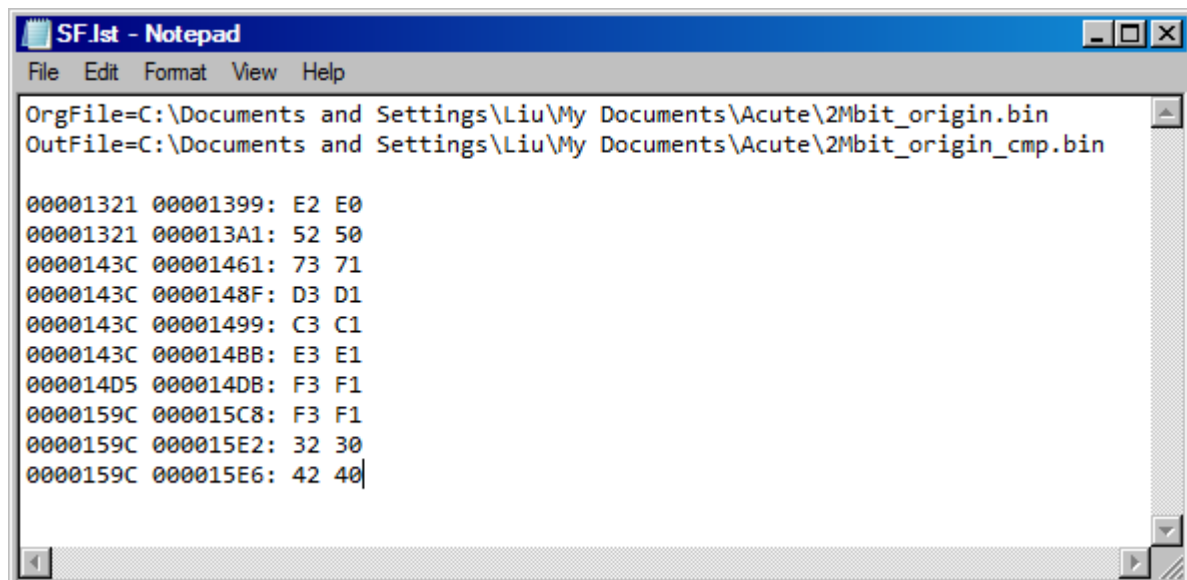
If the OutFile does not exist, it will copy the OrgFile to the OutFile and write the data

to it according to the CheckCmd.

Compare result:



The OutLstFile:

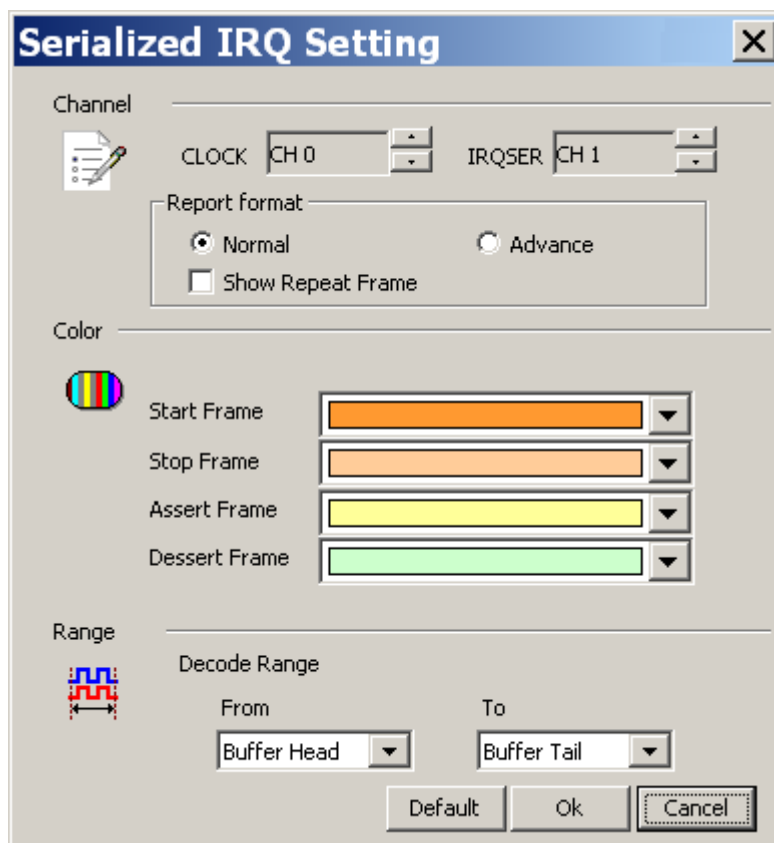


The first column is the compared address from OrgFile, the second column is the different address from OutFile.

## 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

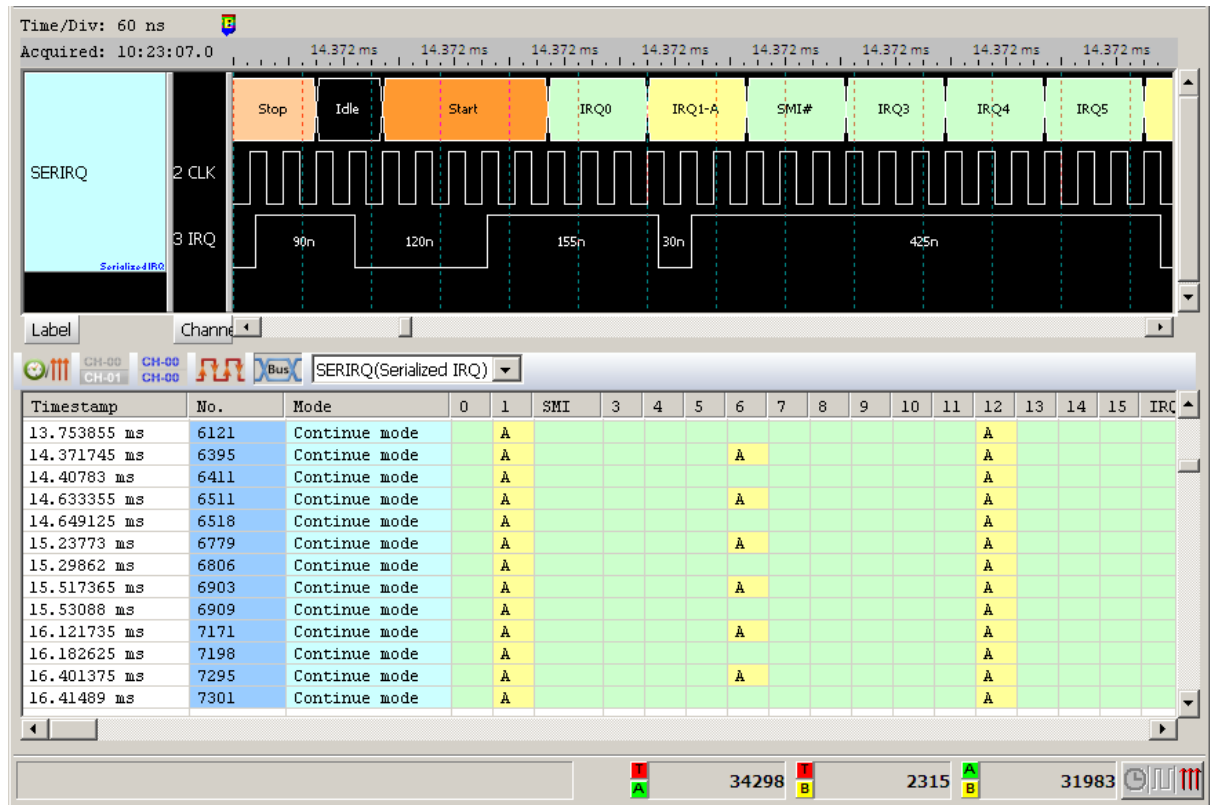
Clock	No.	Mode	0	1	SMI	3	4	5	6	7	8	9	10	11	12	13	14	15	IRQ
10457887	23209	Continue mode		A											A				
10458338	23210	Continue mode		A											A				
10458789	23211	Continue mode		A											A				
10459240	23212	Continue mode		A					A						A				
10459690	23213	Continue mode		A					A						A				
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:

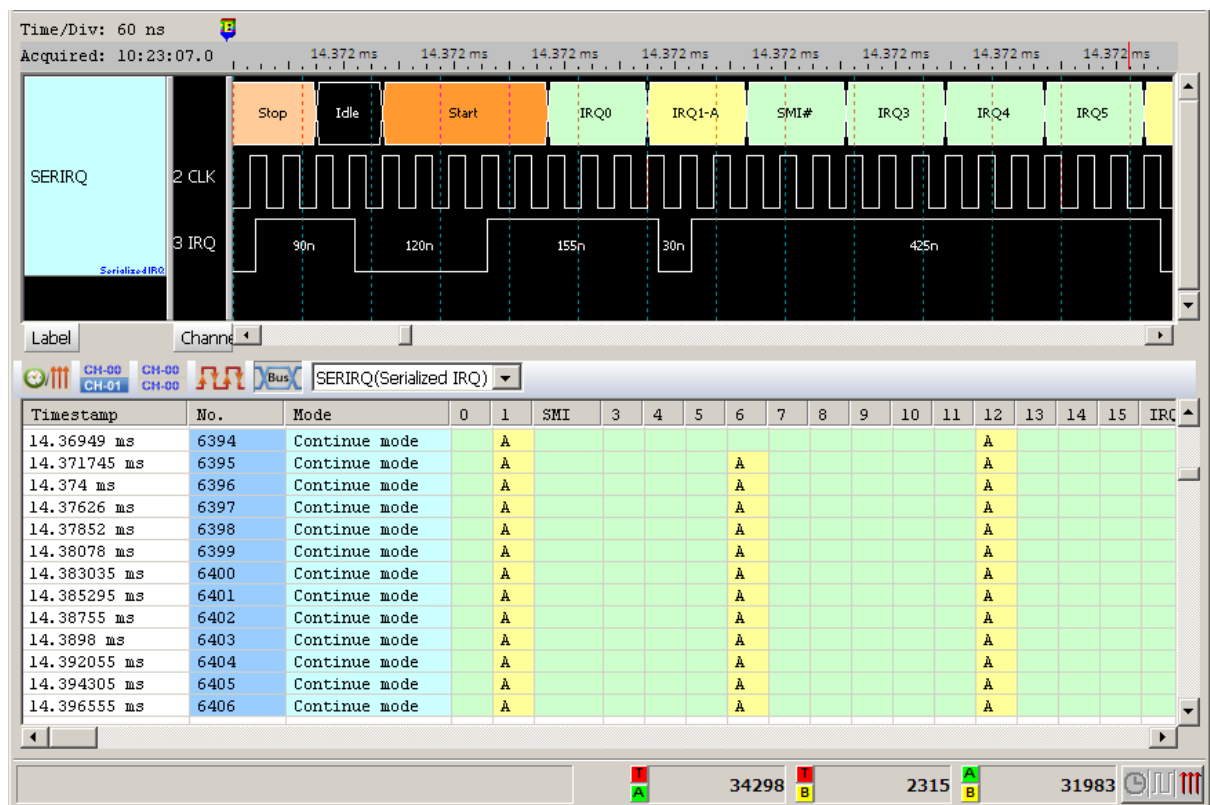
Clock	IRQ/Data Frame	Signal Sampled	# of clocks past Start
-9470	1	IRQ0	2
-9452	2	IRQ1	5
-9434	3	SMI#	8
-9416	4	IRQ3	11
-9398	5	IRQ4	14
-9380	6	IRQ5	17
-9362	7	IRQ6	20
-9344	8	IRQ7	23
-9326	9	IRQ8	26
-9308	10	IRQ9	29
-9290	11	IRQ10	32
-9272	12	IRQ11	35
-9254	13	IRQ12	38
-9236	14	IRQ13	41
-9217	15	IRQ14	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

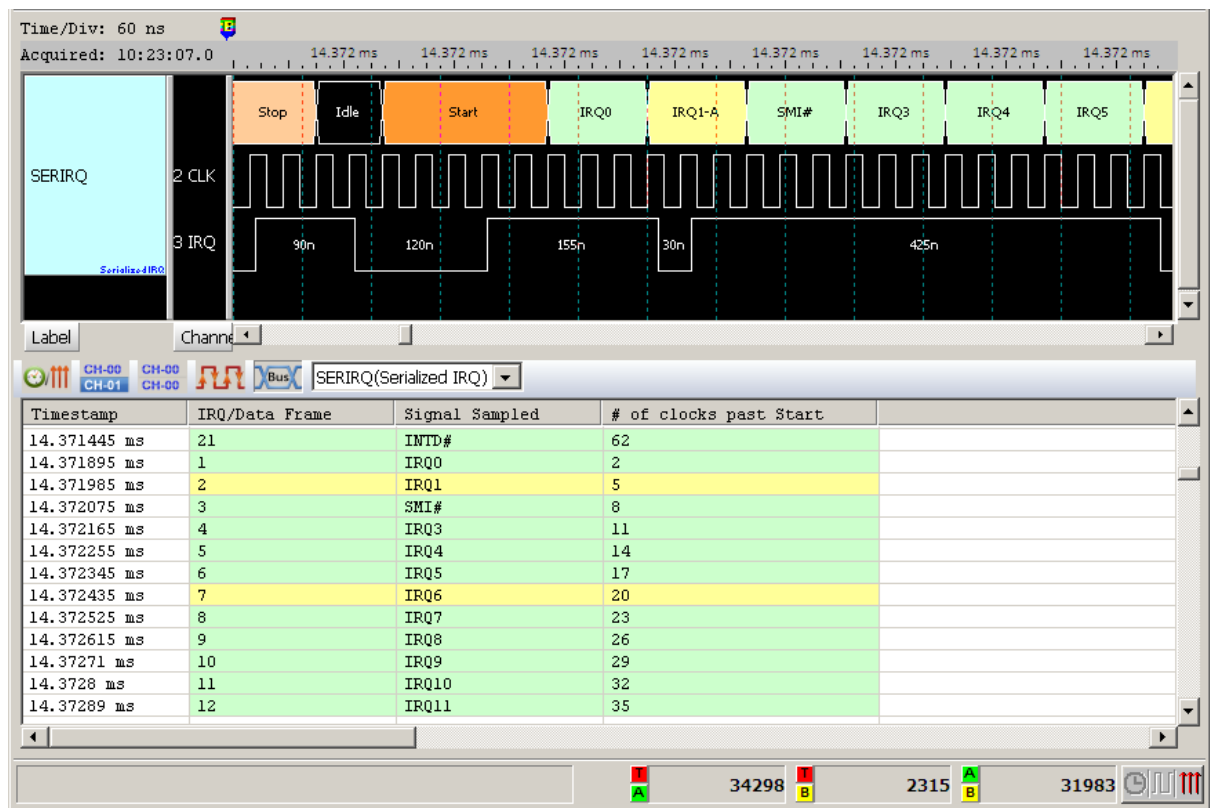
Normal mode



Normal mode(Show repeat frame)



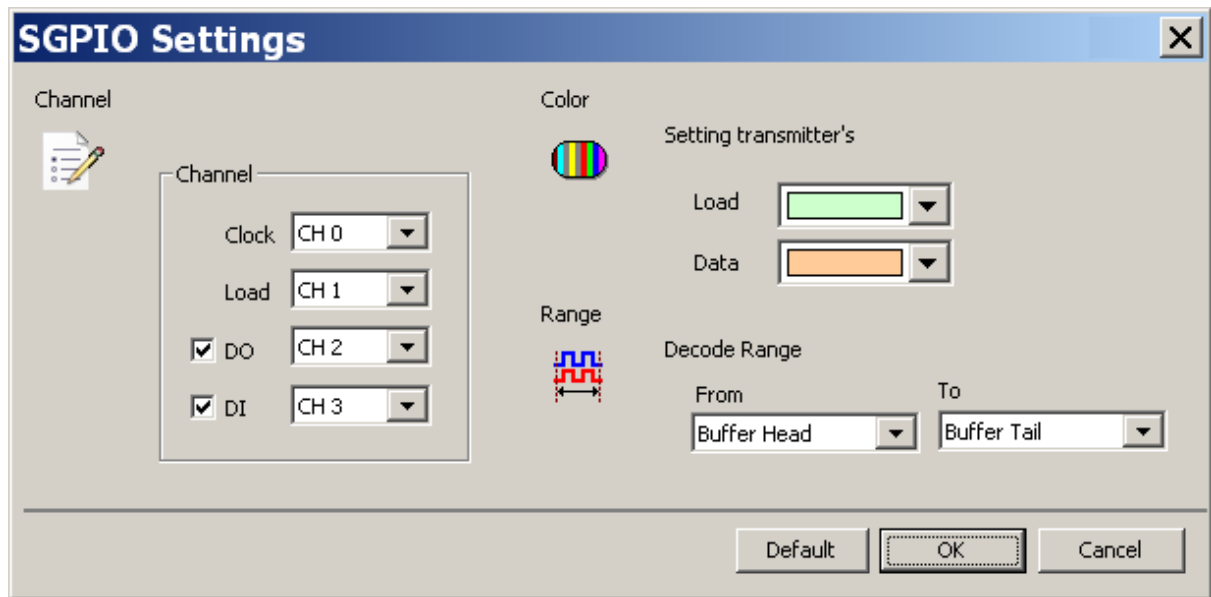
## Advance mode



## 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.

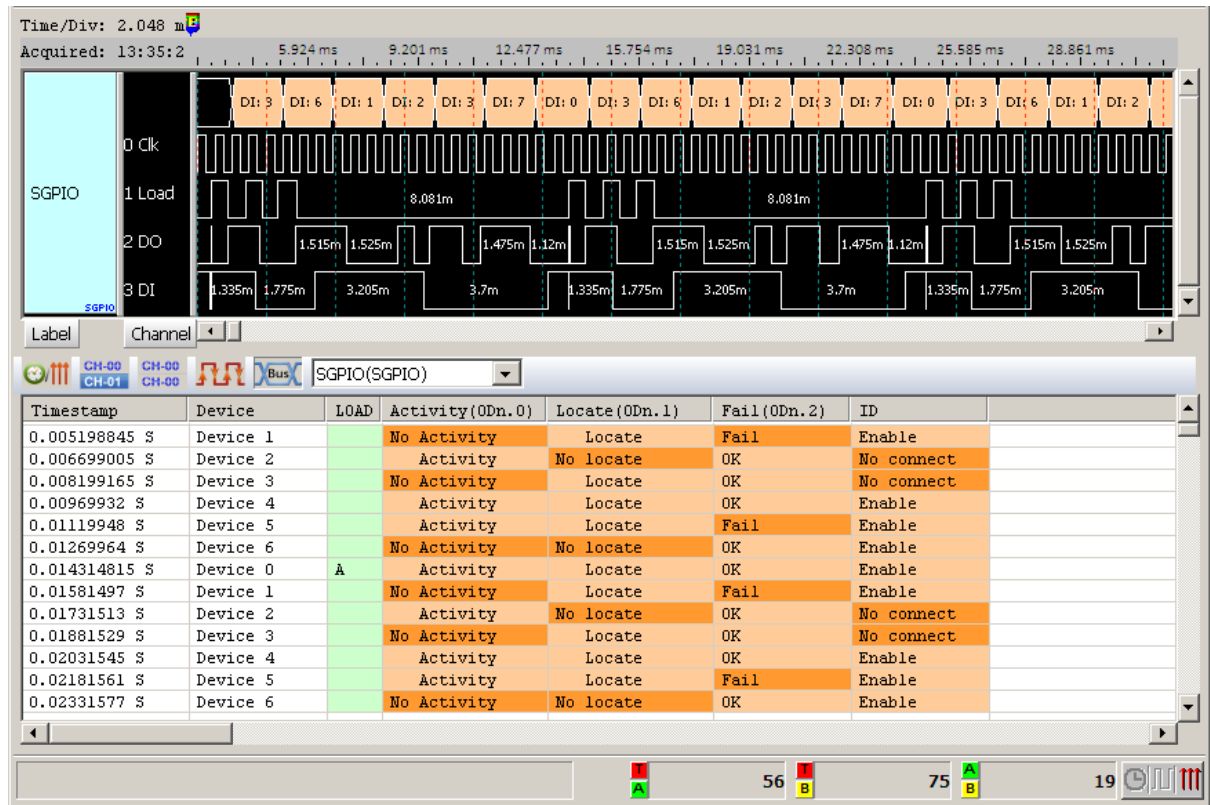
### Settings



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

### Result:

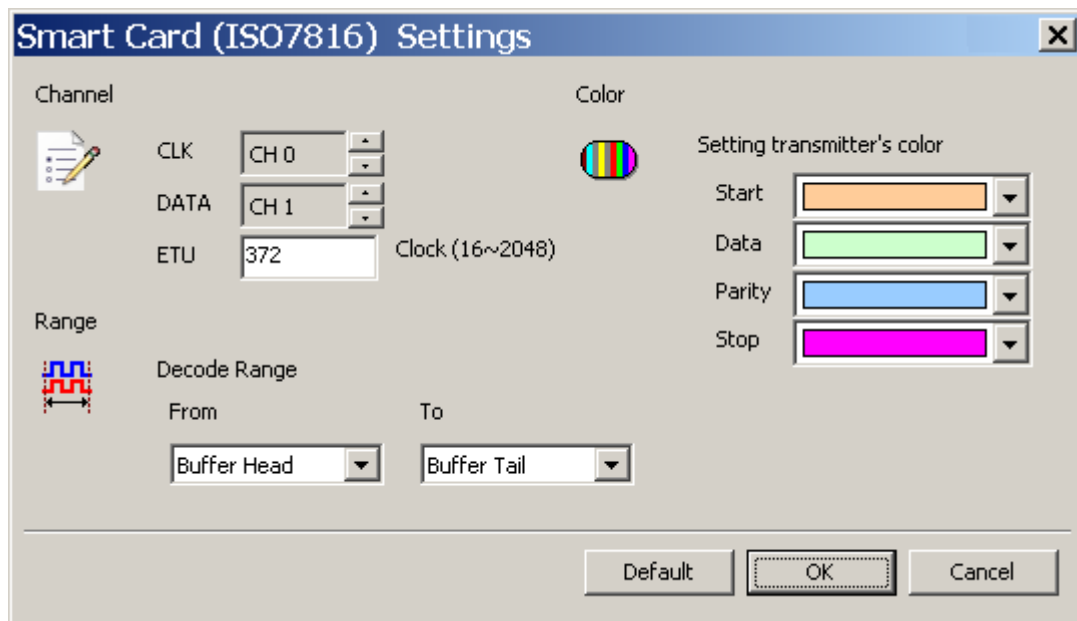
Click OK to run the Smart Card decode and see the result on the Waveform Window below.



## Smart Card (ISO7816)

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

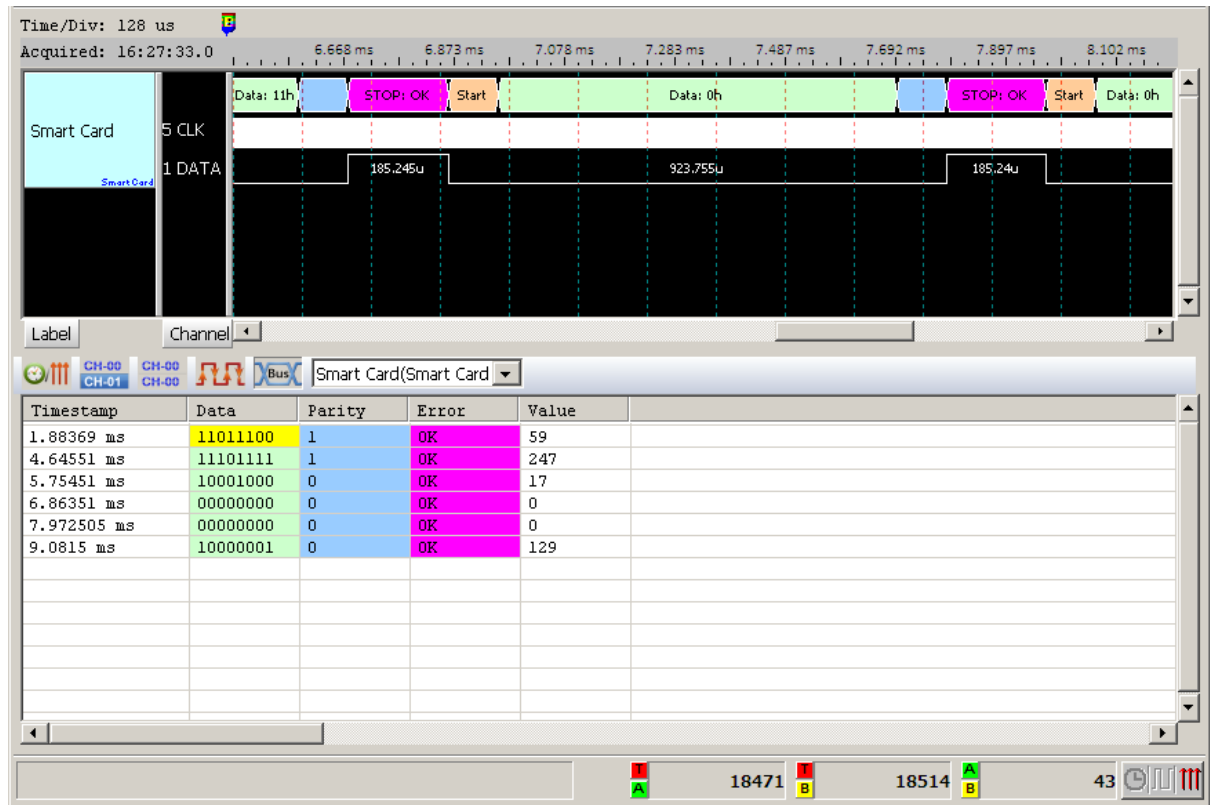
### Settings



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

### Result

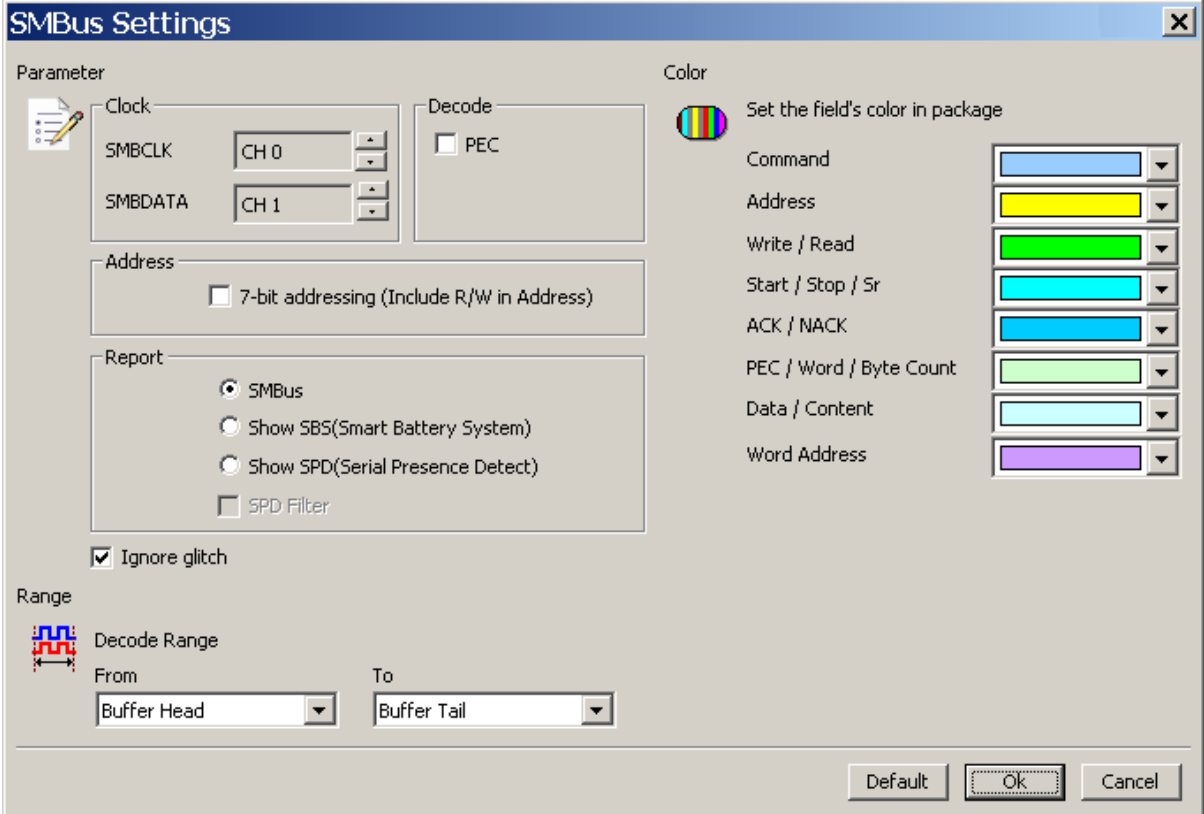
Click **OK** to run the Smart Card decode and see the result on the Waveform Window below.



## System Management Bus (SMBus)

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

### Settings



The image shows the 'SMBus Settings' dialog box. It is divided into several sections:

- Parameter:**
  - Clock:** SMBCLK is set to CH 0, and SMBDATA is set to CH 1.
  - Decode:** There is a checkbox for 'PEC' which is currently unchecked.
  - Address:** A checkbox for '7-bit addressing (Include R/W in Address)' is unchecked.
  - Report:**
    - ☒ SMBus
    - ☐ Show SBS(Smart Battery System)
    - ☐ Show SPD(Serial Presence Detect)
    - ☐ SPD Filter
  - ☒ Ignore glitch
- Range:**
  - Decode Range:**
    - From: Buffer Head
    - To: Buffer Tail
- Color:**
  - Set the field's color in package
  - Command: Blue
  - Address: Yellow
  - Write / Read: Green
  - Start / Stop / Sr: Cyan
  - ACK / NACK: Light Blue
  - PEC / Word / Byte Count: Light Green
  - Data / Content: Very Light Blue
  - Word Address: Purple

At the bottom right, there are buttons for 'Default', 'Ok', and 'Cancel'.

**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.

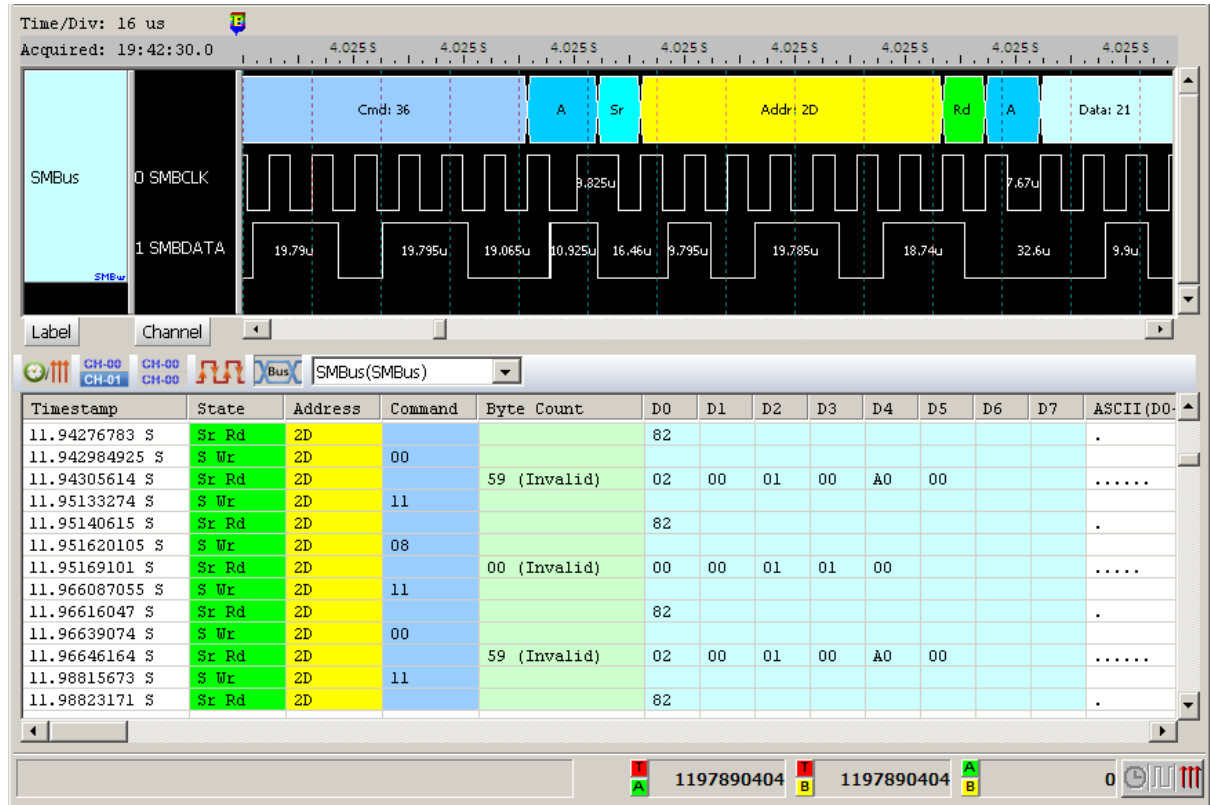
**Ignore glitch:** Ignore the glitch when the slow transitions.



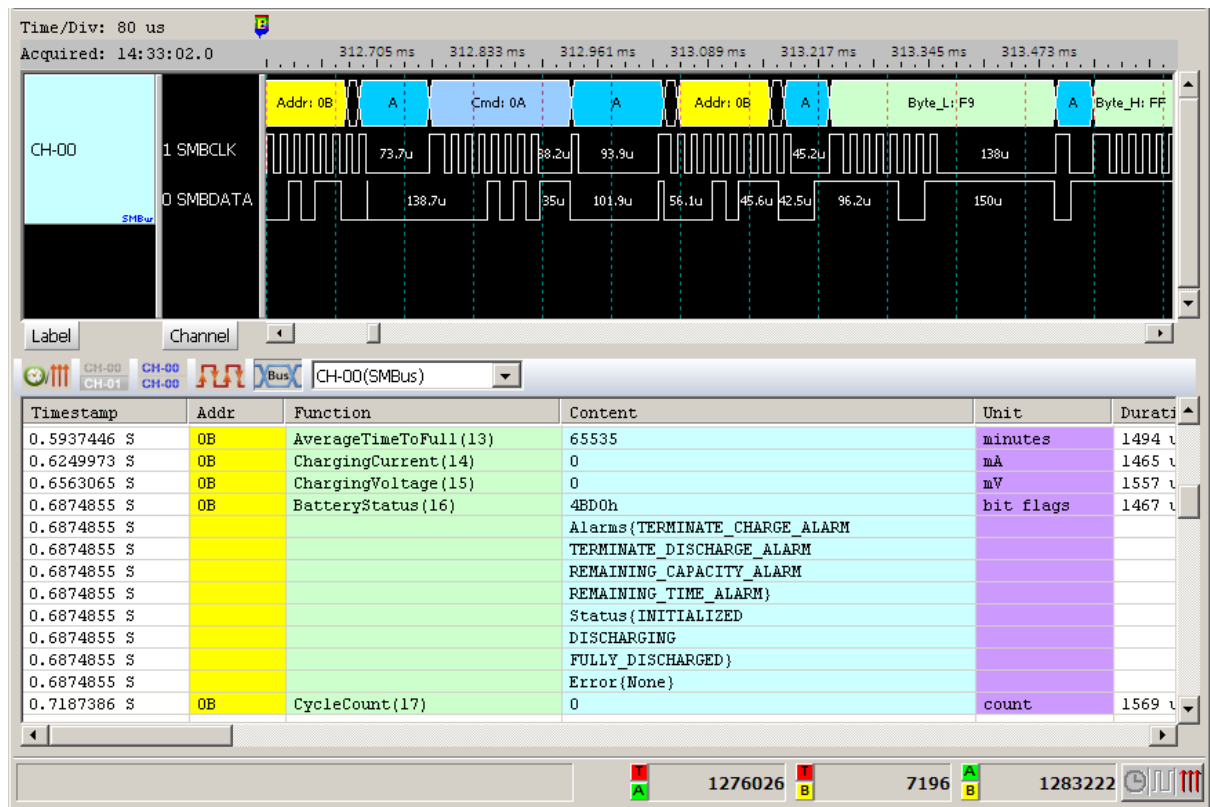
## Result

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

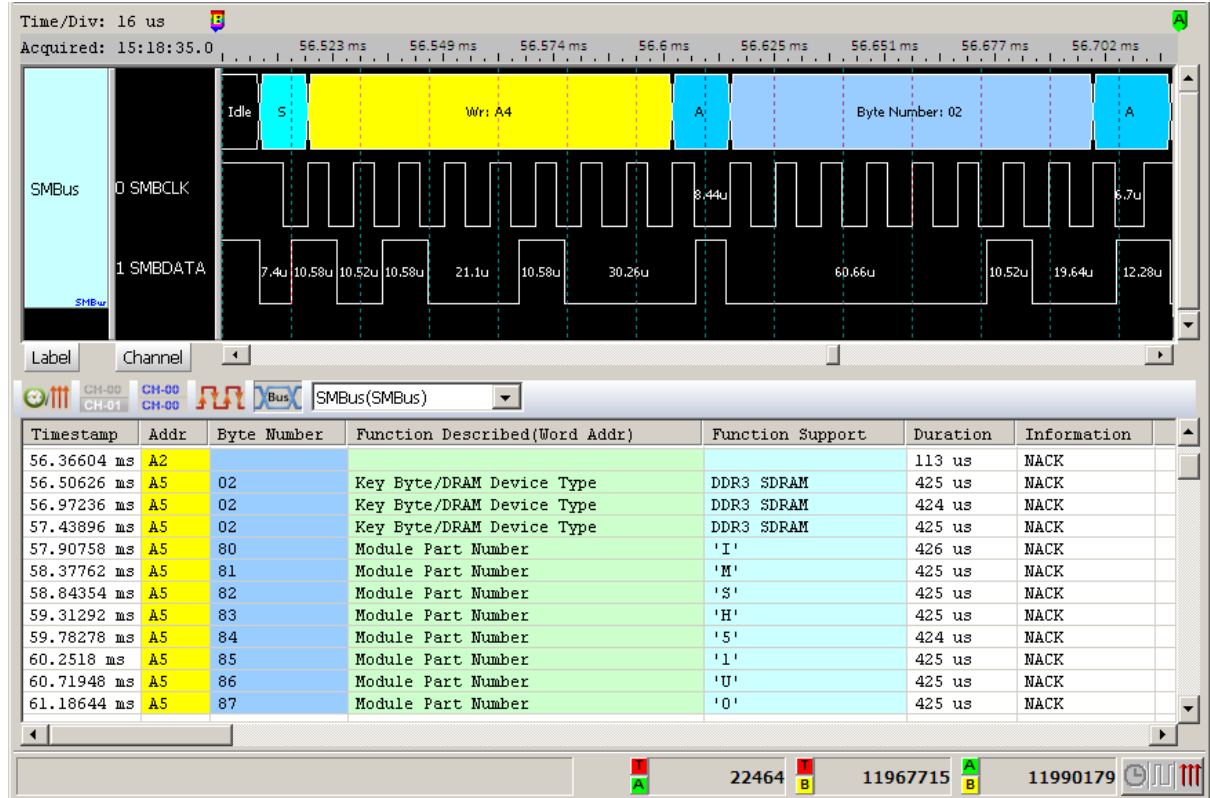
## SMBus



## Show SBS (Smart Battery System)



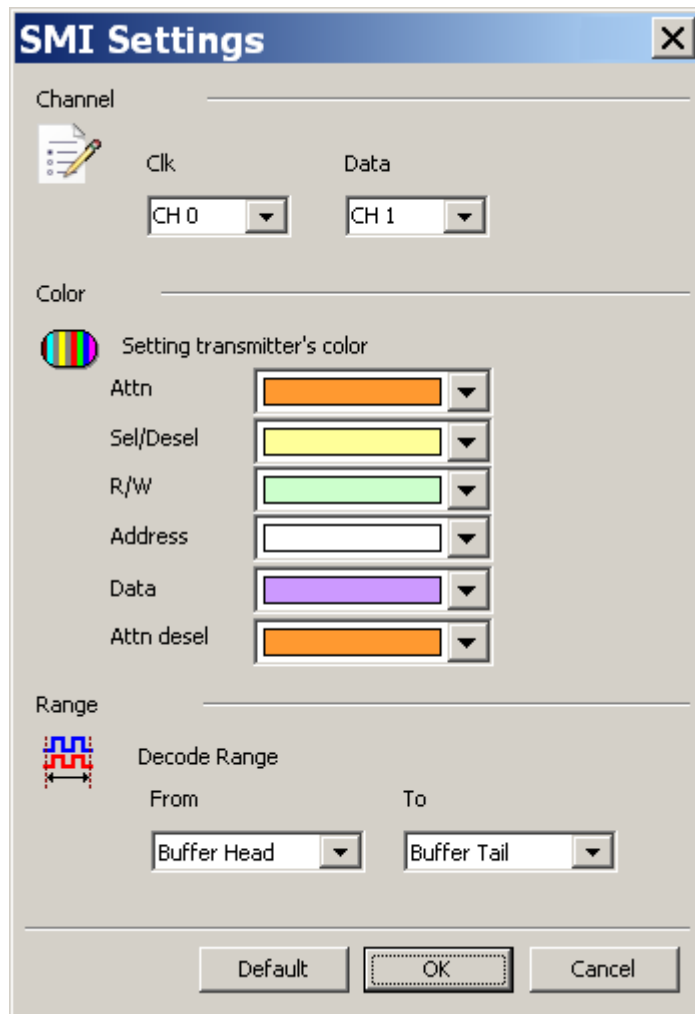
## Show SPD (Serial Presence Detect)



## Serial Microprocessor Interface (SMI)

The SMI is a two-wire bus.

### Settings



The SMI Settings dialog box is divided into three main sections: Channel, Color, and Range. The Channel section has a 'Channel' label and a list icon, followed by 'Clk' and 'Data' dropdown menus set to 'CH 0' and 'CH 1' respectively. The Color section has a 'Color' label and a color wheel icon, followed by the text 'Setting transmitter's color' and six color-coded dropdown menus for 'Attn' (orange), 'Sel/Desel' (yellow), 'R/W' (green), 'Address' (white), 'Data' (purple), and 'Attn desel' (orange). The Range section has a 'Range' label and a waveform icon, followed by the text 'Decode Range' and 'From' and 'To' dropdown menus set to 'Buffer Head' and 'Buffer Tail' respectively. At the bottom are 'Default', 'OK', and 'Cancel' buttons.

**SMI Settings**

Channel

Clk: CH 0      Data: CH 1

Color

Setting transmitter's color

Attn: [Orange]      Sel/Desel: [Yellow]      R/W: [Green]      Address: [White]      Data: [Purple]      Attn desel: [Orange]

Range

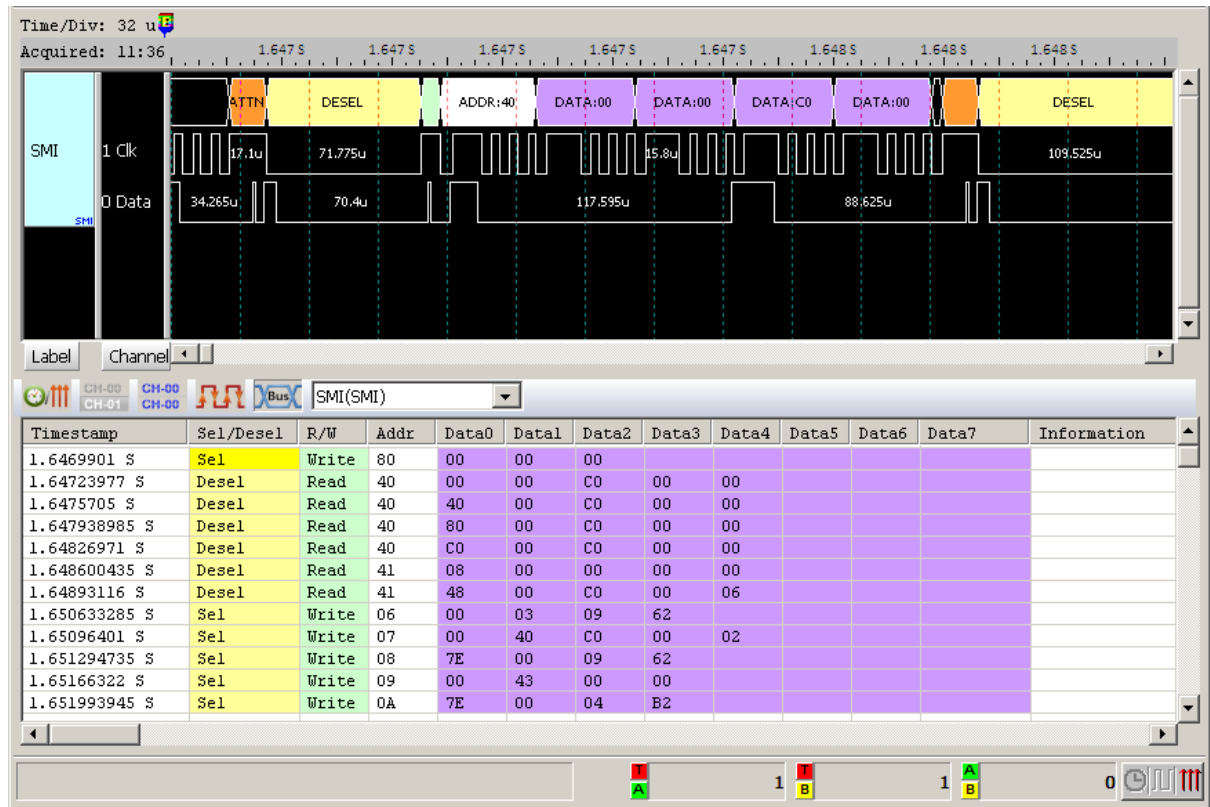
Decode Range

From: Buffer Head      To: Buffer Tail

Default      OK      Cancel

**Channel:** Show the selected channels (Clk and Data)

## Result

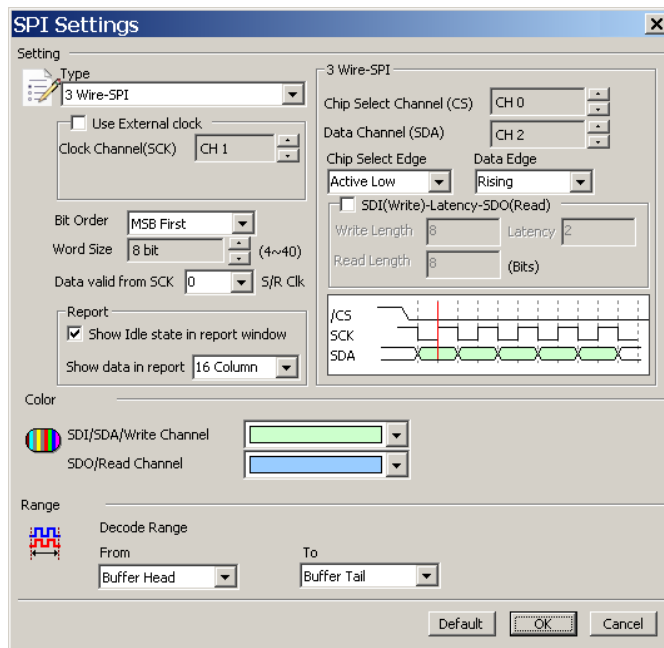


## Serial Peripheral Interface (SPI)

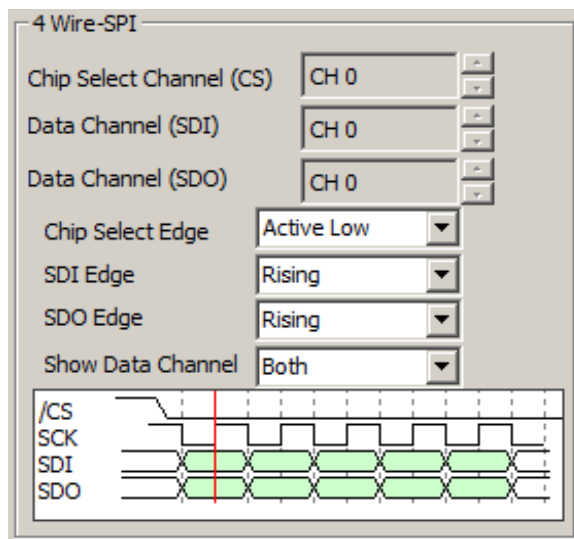
The SPI, is one kind of 4-wires synchronous serial data link. The SPI bus can be 4 wires, 3 wires, or 2 wires.

### Settings

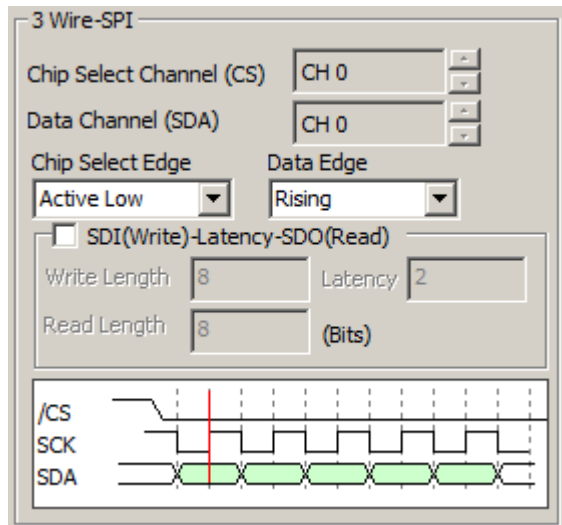
**Type:** The default is 3 Wire-SPI.



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

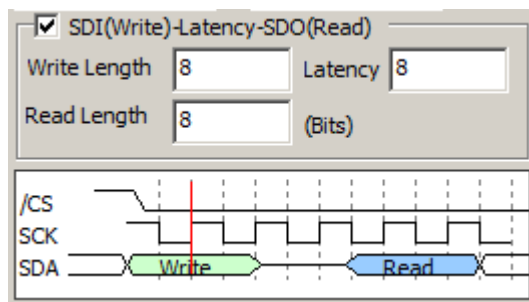


### 3 Wire-SPI dialog box → CS, SCK, SDA

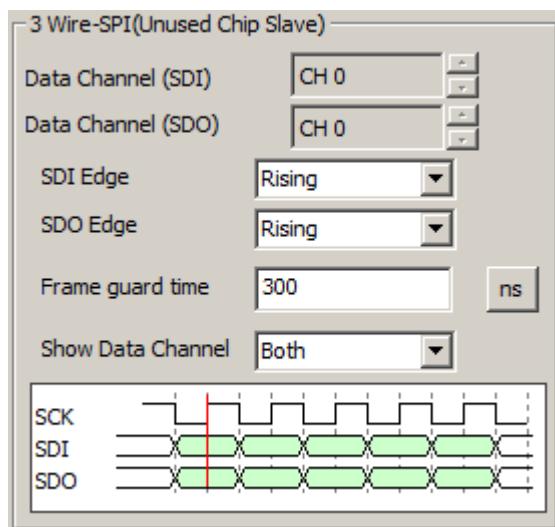


Click the SDI(Write)-Latency-SDO(Read) to show the dialog below.

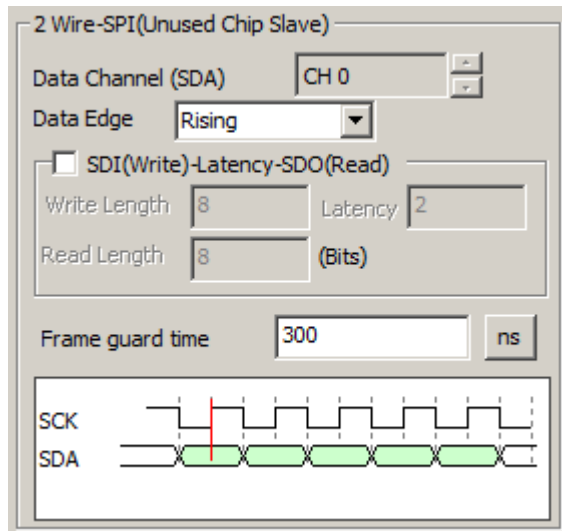
The maximum length is 65535 (Bits)



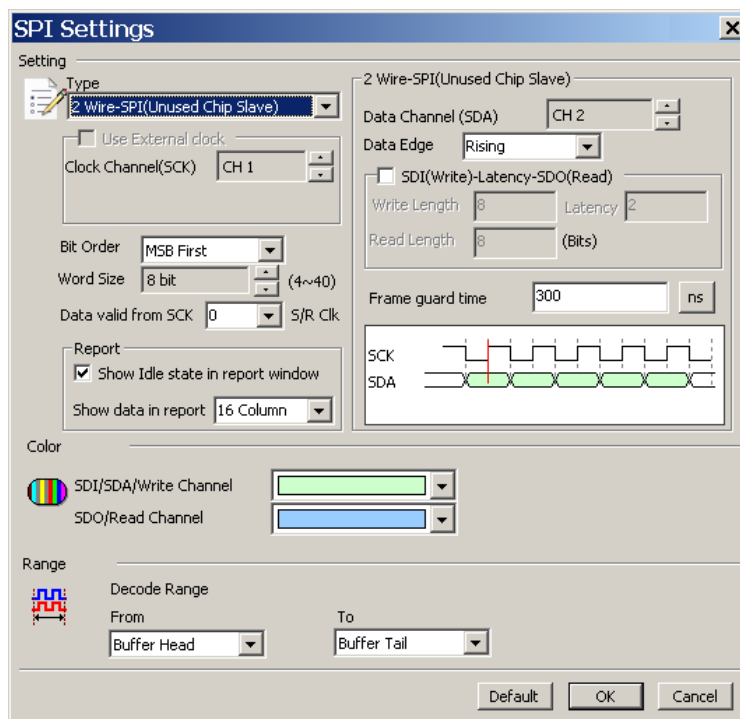
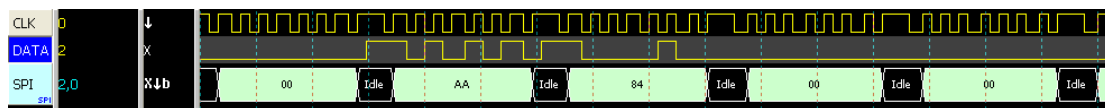
### 3 Wire-SPI (Unused Chip Slave) dialog box → SCK, SDI, SDO



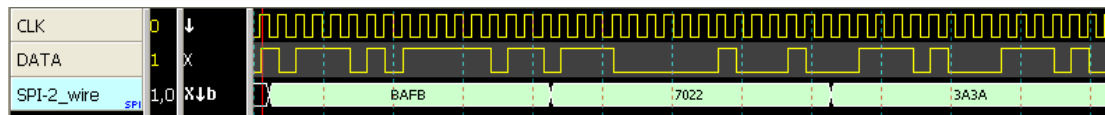
## 2 Wire-SPI (Unused Chip Slave) dialog box → SCK, SDA



If the chip slave is not used and the interval between frames is not 0. You can set the interval as 6 $\mu$ s, any data bit higher than 6 $\mu$ s 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.



**SPI Settings**

Setting

Type: 2 Wire-SPI(Unused Chip Slave)

☐ Use External clock

Clock Channel(SCK): CH 1

Bit Order: MSB First

Word Size: 8 bit (4~40)

Data valid from SCK: 0 S/R Clk

Report

☒ Show Idle state in report window

Show data in report: 8 Column

2 Wire-SPI(Unused Chip Slave)

Data Channel (SDA): CH 2

Data Edge: Rising

☐ SDI(Write)-Latency-SDO(Read)

Write Length: 8 Latency: 2

Read Length: 8 (Bits)

Frame guard time: 300 ns

SCK: [Timing Diagram]

SDA: [Timing Diagram]

Color

SDI/SDA/Write Channel: [Color Selection]

SDO/Read Channel: [Color Selection]

Range

Decode Range

From: Buffer Head To: Buffer Tail

Default OK Cancel

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

☒ SDI(Write)-Latency-SDO(Read)

Write Length: 8 Latency: 2

Read Length: 8 (Bits)

Frame guard time: 0 ns

SCK: [Timing Diagram]

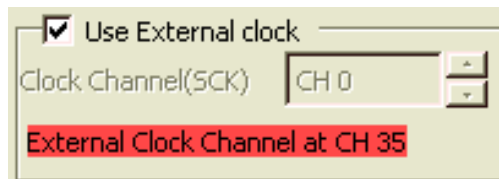
SDA: [Timing Diagram]

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



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

**Use External clock:** Connect to the last channel of your instrument.



**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



## 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

**SPI NAND Settings**

**Parameter Settings**

CS# Ch 0 SCK Ch 1 Winbond  
SI/SO0 Ch 2 SO/SO1 Ch 3 W25N01GV  
WP/SO2 Ch 4 HOLD#/SO3 Ch 5

Start up reading mode Continuous Read

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

**Color**

OpCode Dummy  
Address Data In  
Data Out

**Range**

Decode Range  
From To  
Buffer Head Buffer Tail

Default OK Cancel

**CS#:** Chip select

**SCLK:** Clock

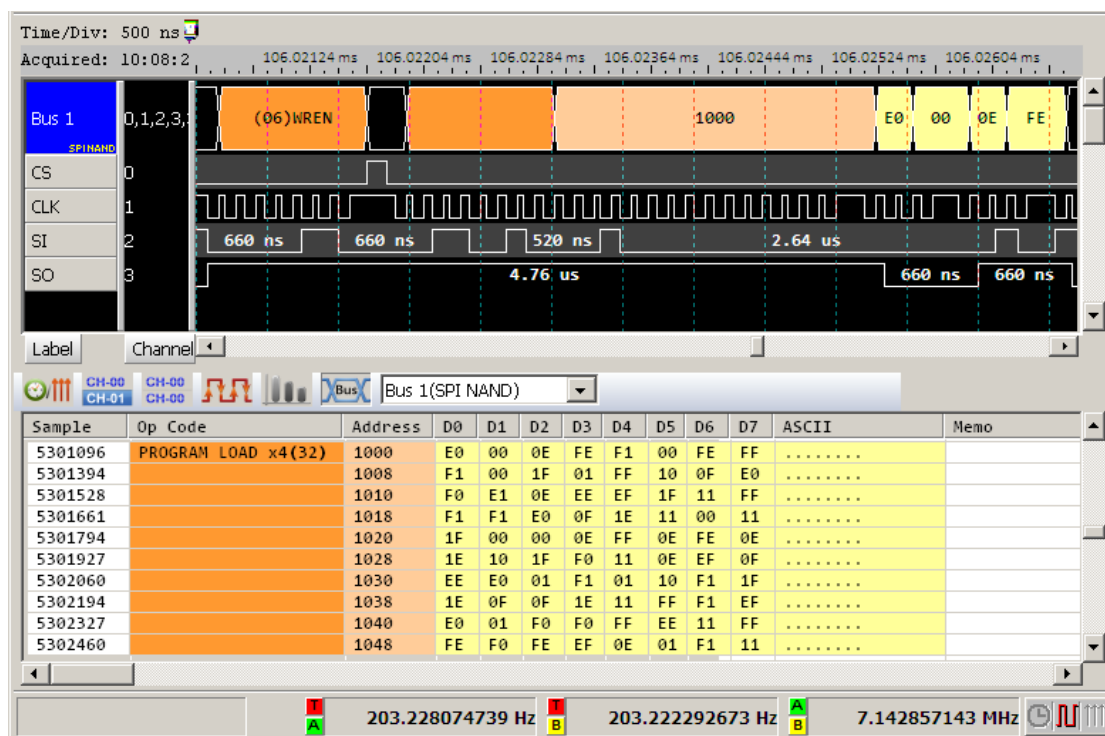
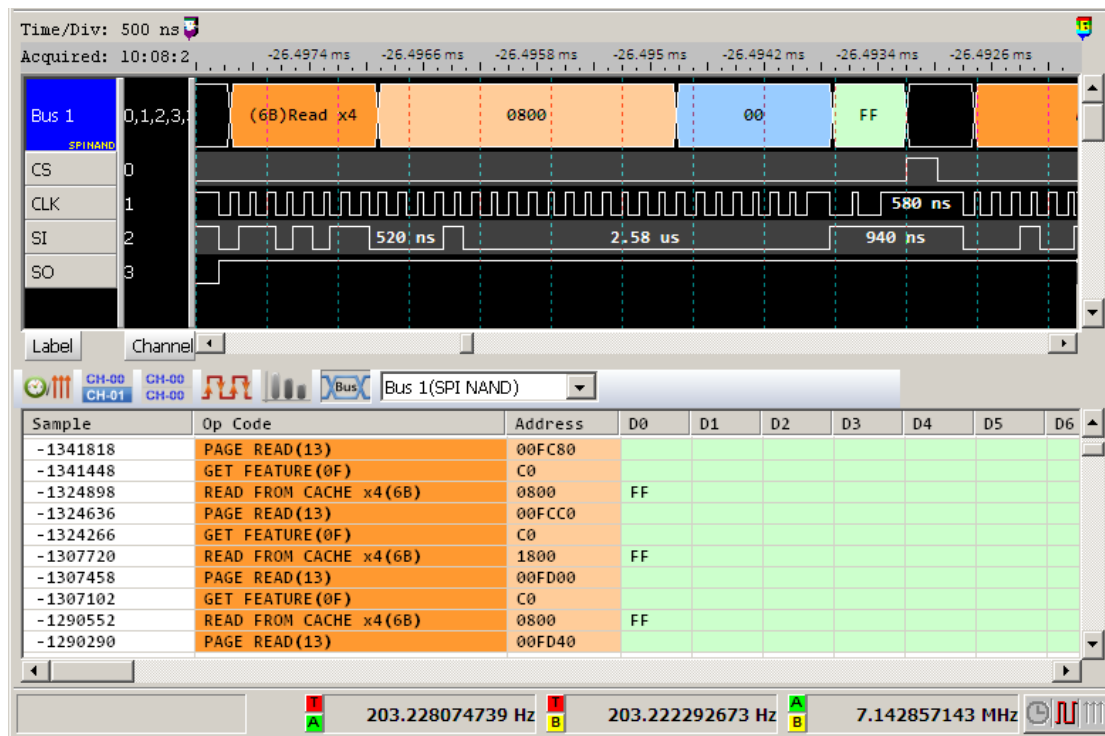
**SIO0 – SIO3:** Data

**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.

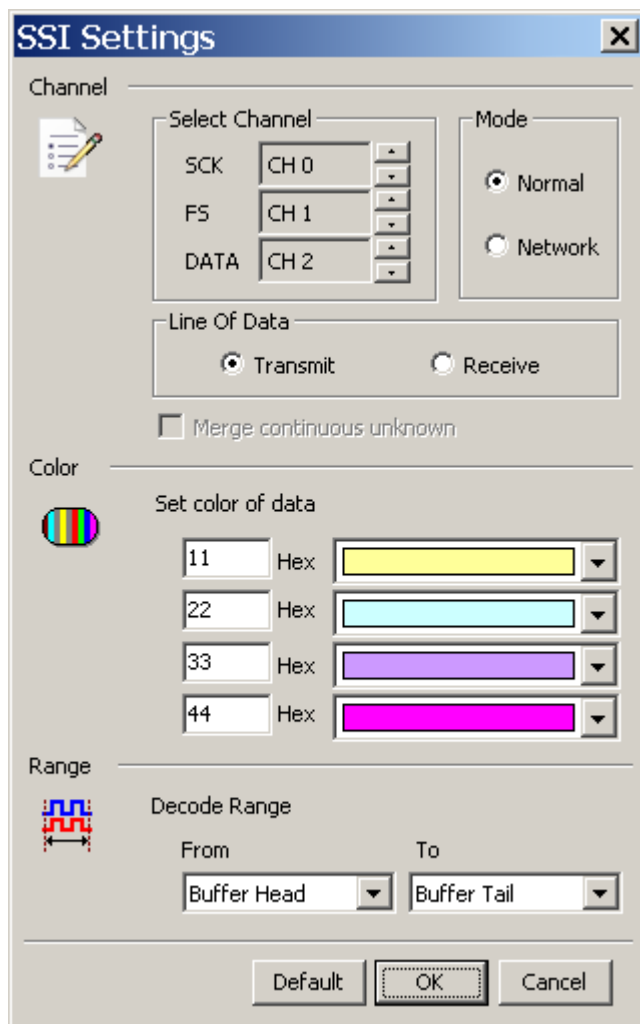
## Result



## 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.

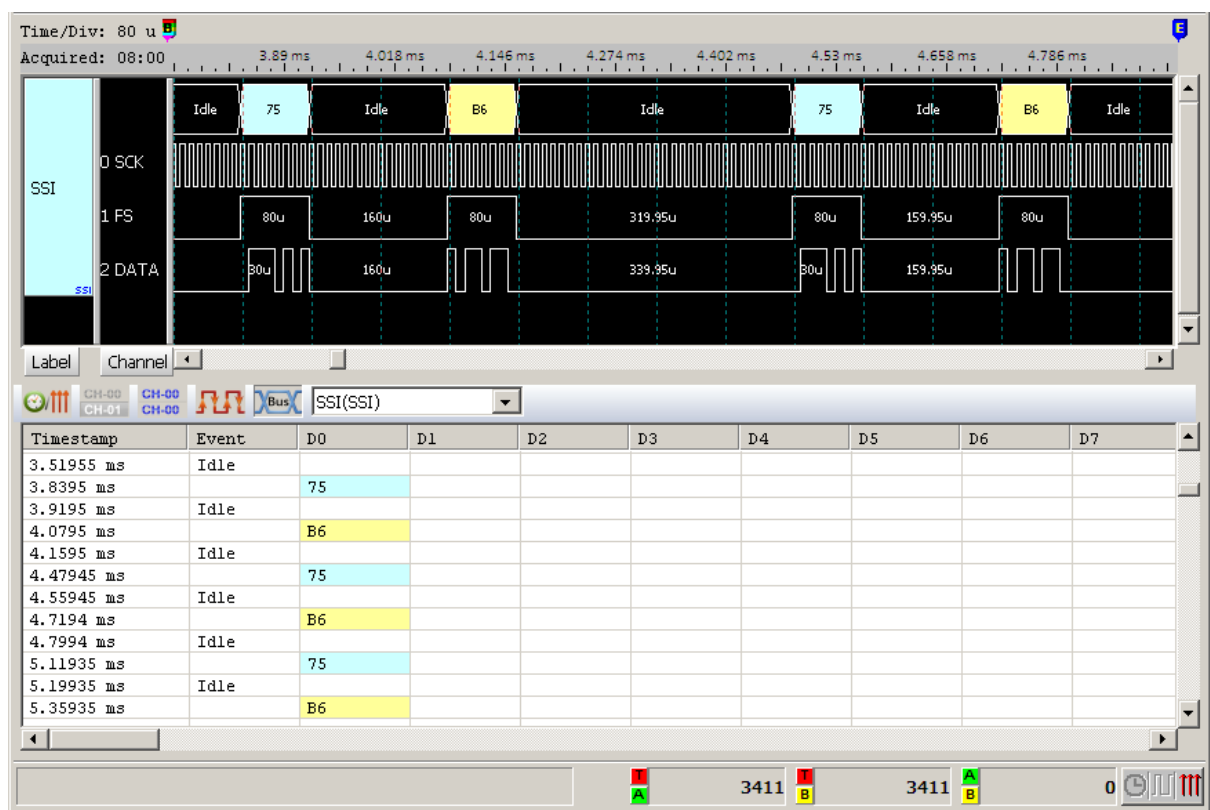
**Line Of Data:** Transmit or Receive.

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

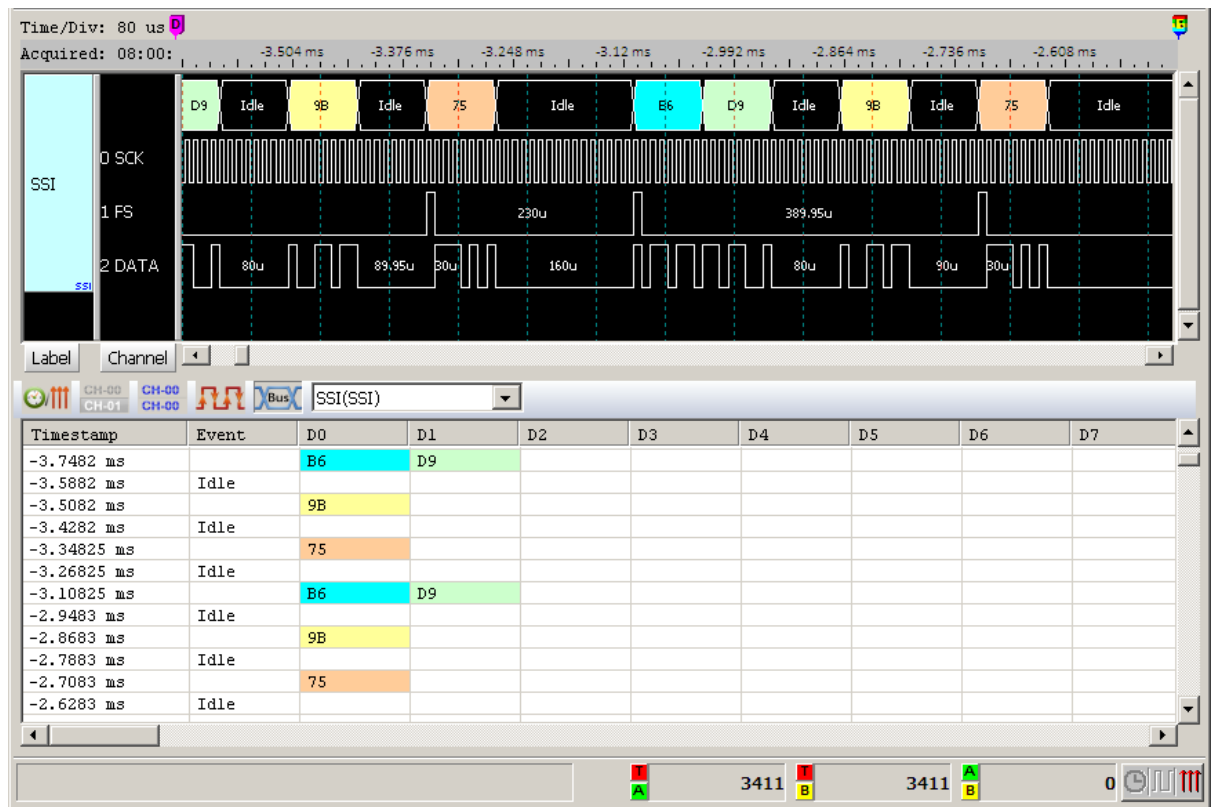
## Result

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

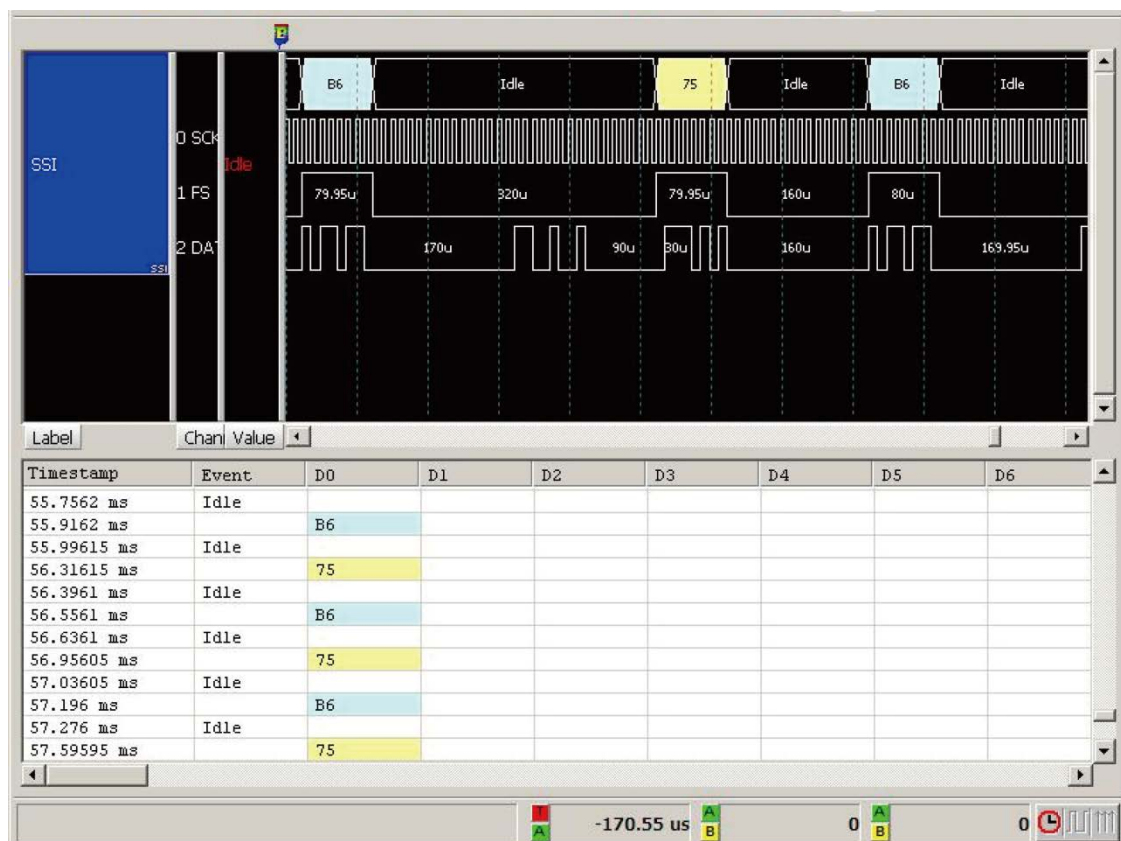
Normal Transmit



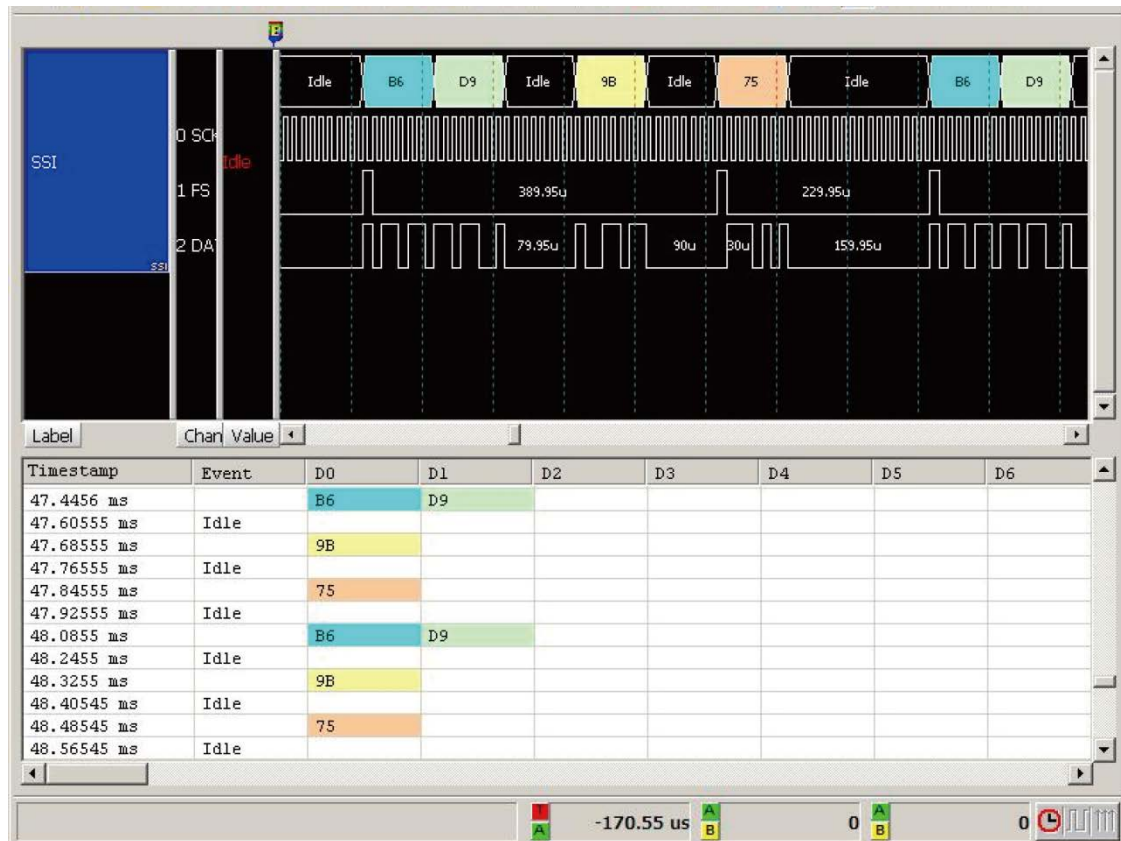
## Normal Receive



## Network Transmit



## Network Receive





## ST7669

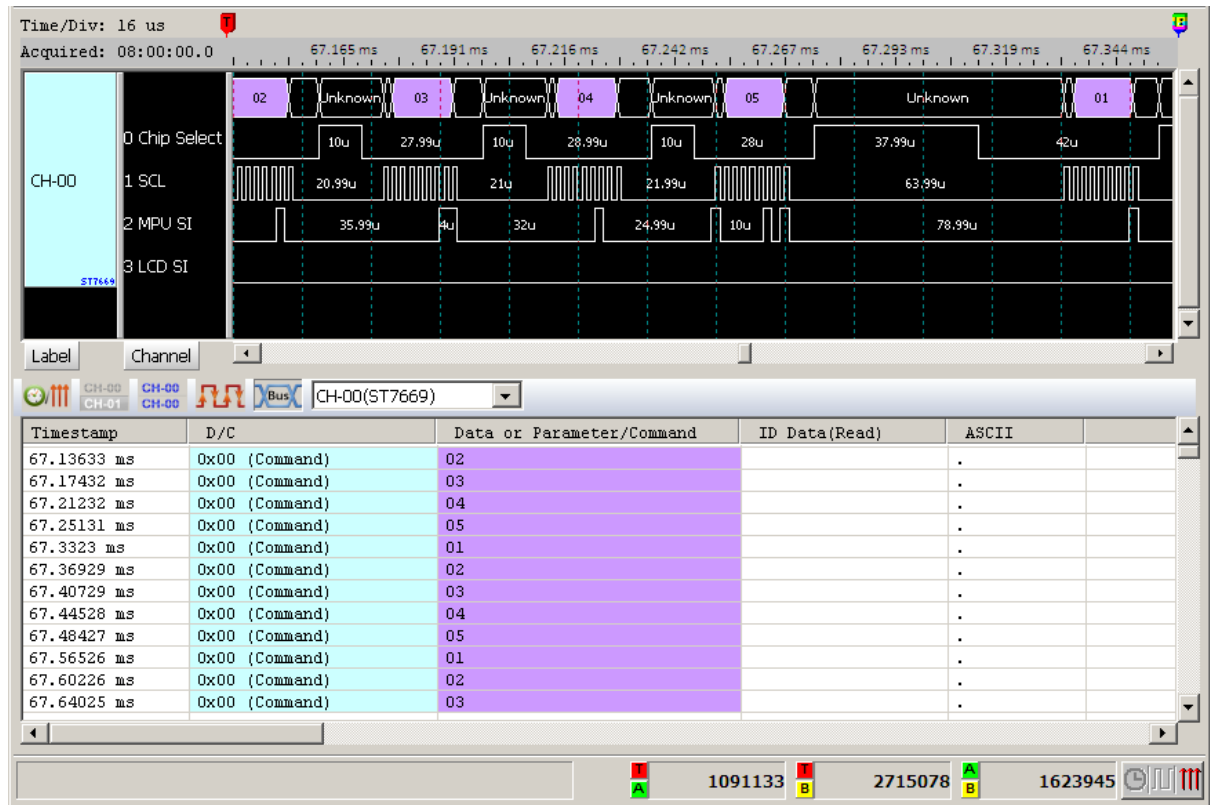
ST7669 was developed by Sitronix to interface with the LCD module.

### Settings

**Channel:** Show the selected channels.

### Result

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

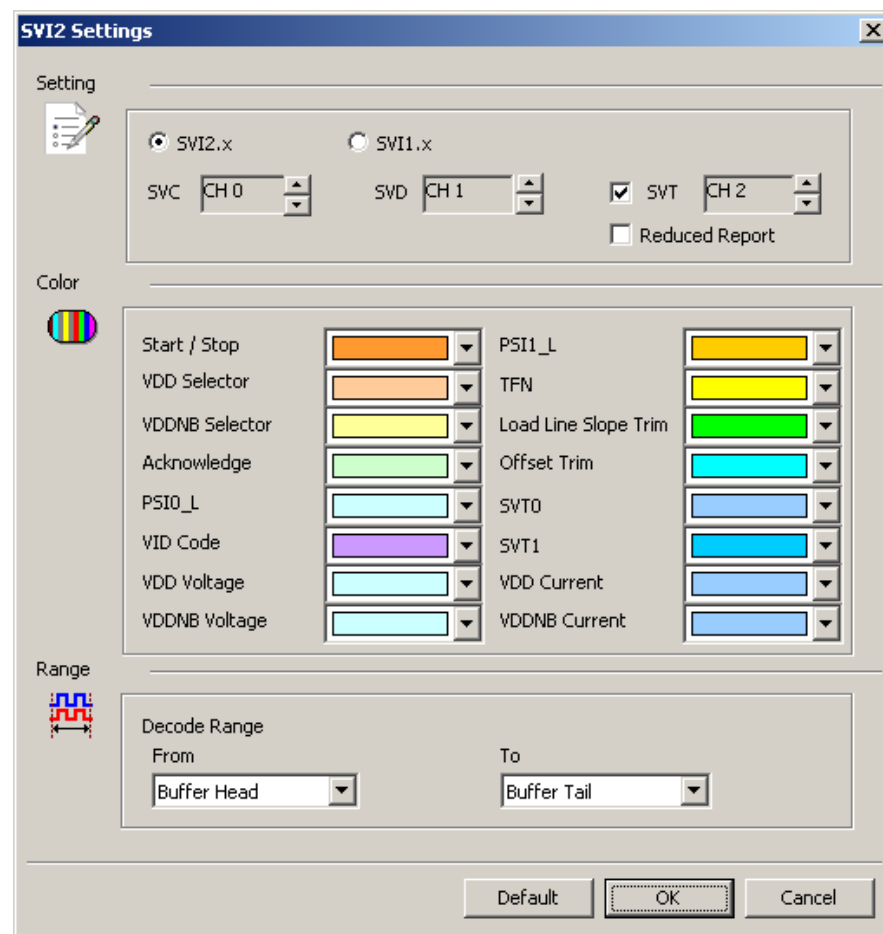


## Serial VID Interface 2.0 (SVI2)

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



The **SVI2 Settings** dialog box is used to configure the SVI2 protocol settings. It contains three main sections: **Setting**, **Color**, and **Range**.

- Setting:**
  - Protocol selection: ☒ SVI2.x, ☐ SVI1.x
  - SVC: CH 0 (dropdown)
  - SVD: CH 1 (dropdown)
  - SVT: ☒ CH 2 (dropdown)
  - ☐ Reduced Report
- Color:**
  - Start / Stop: Orange
  - VDD Selector: Orange
  - VDDNB Selector: Yellow
  - Acknowledge: Light Green
  - PSIO\_L: Light Blue
  - VID Code: Purple
  - VDD Voltage: Light Blue
  - VDDNB Voltage: Light Blue
  - PSI1\_L: Yellow
  - TFN: Yellow
  - Load Line Slope Trim: Green
  - Offset Trim: Cyan
  - SVT0: Blue
  - SVT1: Blue
  - VDD Current: Blue
  - VDDNB Current: Blue
- Range:**
  - Decode Range:
    - From: Buffer Head (dropdown)
    - To: Buffer Tail (dropdown)

Buttons at the bottom: Default, OK, Cancel.

**SVC:** SVI2 clock.

**SVD:** SVI2 data.

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

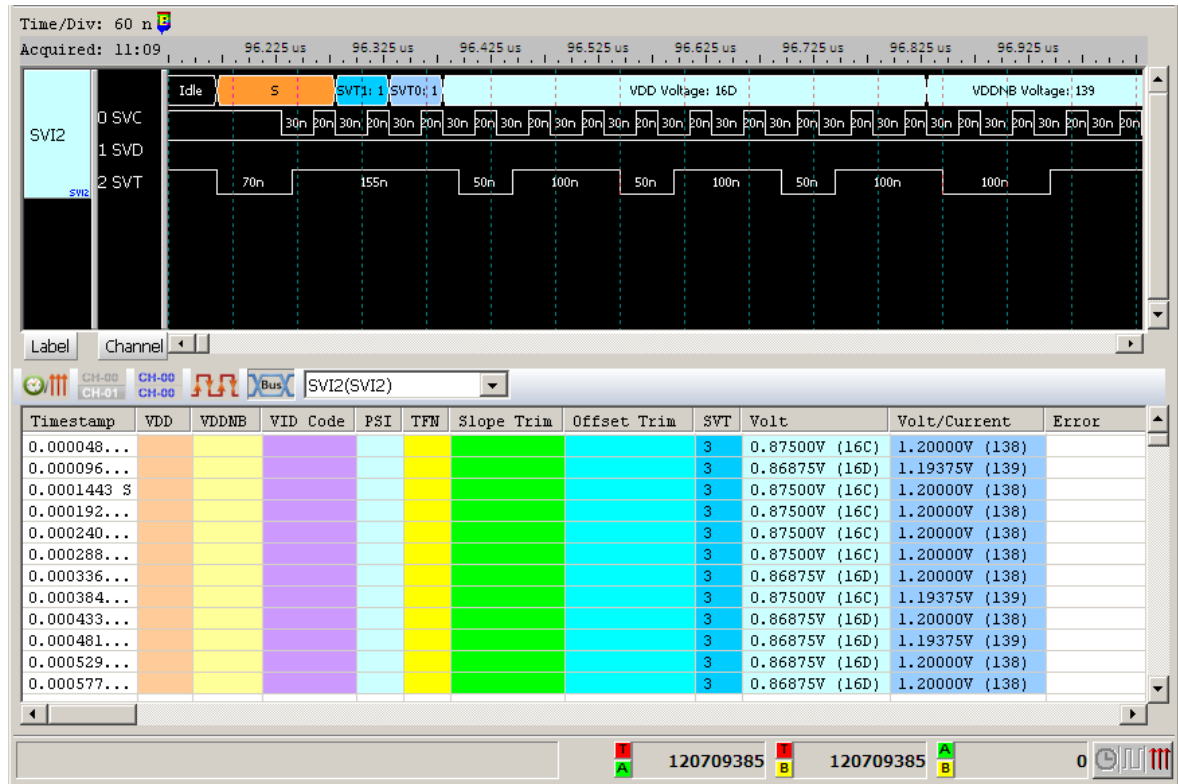
**SVI2.x / SVI1.x:** Select SVI2 / SVI protocol to decode.

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

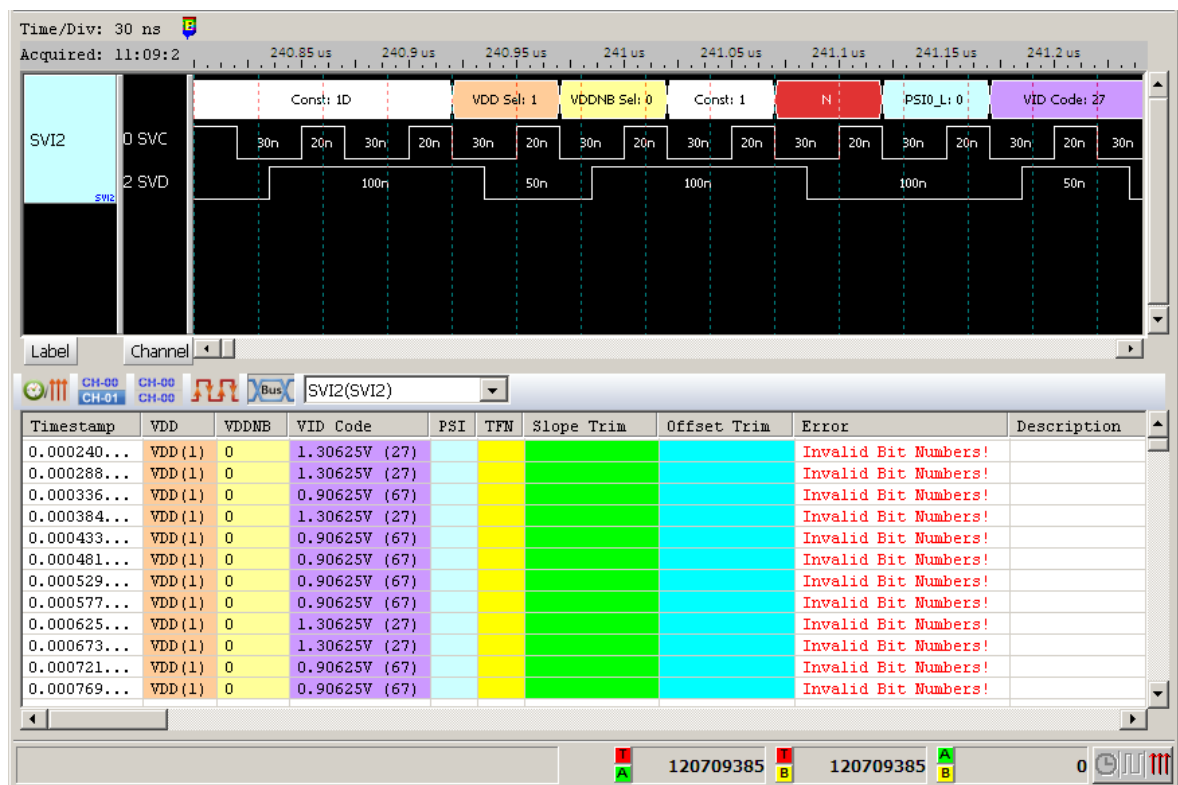
## Result

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

## Show SVT



## Without SVT



## Serial VID (SVID) (Upon Request)

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:

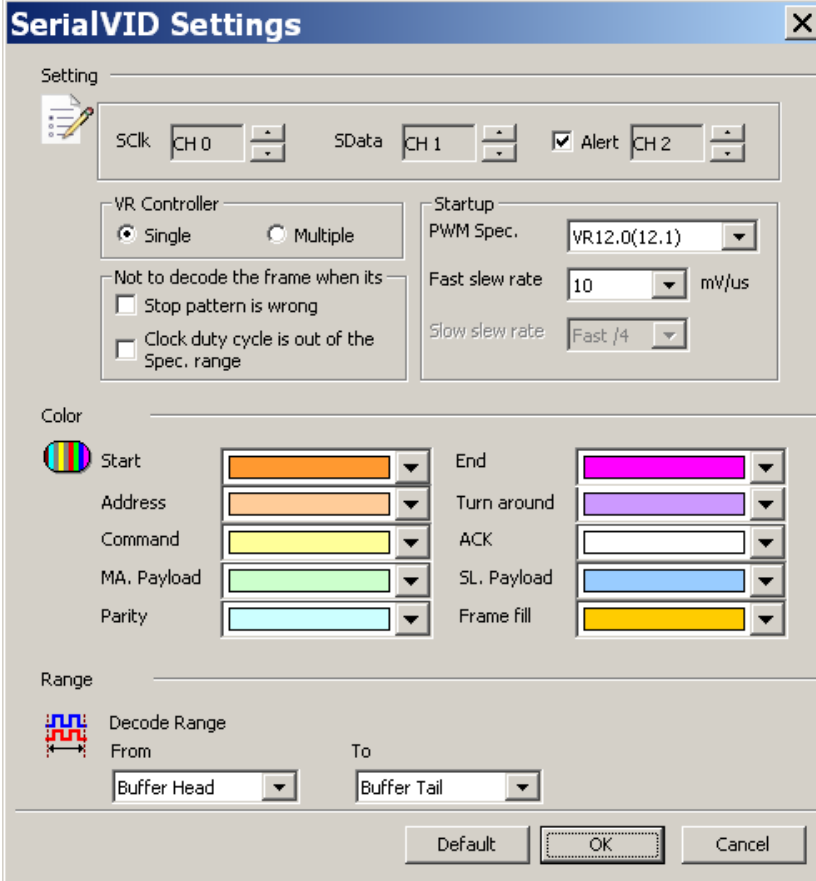
IMVP7/VR12, VR12.1, VR12.5, VR12.6

IMVP8/VR13

IMVP9/VR14

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

### Settings



The SerialVID Settings dialog box is used to configure the SVID protocol. It includes sections for Setting, Color, and Range.

**Setting**

- SCLK: CH 0
- SData: CH 1
- Alert: ☒ CH 2
- VR Controller: ☒ Single, ☐ Multiple
- Startup PWM Spec.: VR12.0(12.1)
- Fast slew rate: 10 mV/us
- Slow slew rate: Fast / 4
- Not to decode the frame when its:
  - ☐ Stop pattern is wrong
  - ☐ Clock duty cycle is out of the Spec. range

**Color**

Start	End
Address	Turn around
Command	ACK
MA, Payload	SL, Payload
Parity	Frame fill

**Range**

Decode Range

From: Buffer Head To: Buffer Tail

Buttons: Default, OK, Cancel

**SCLK:** SVID clock.

**SData:** SVID data.

**Alert:** Alert, optional reminder.

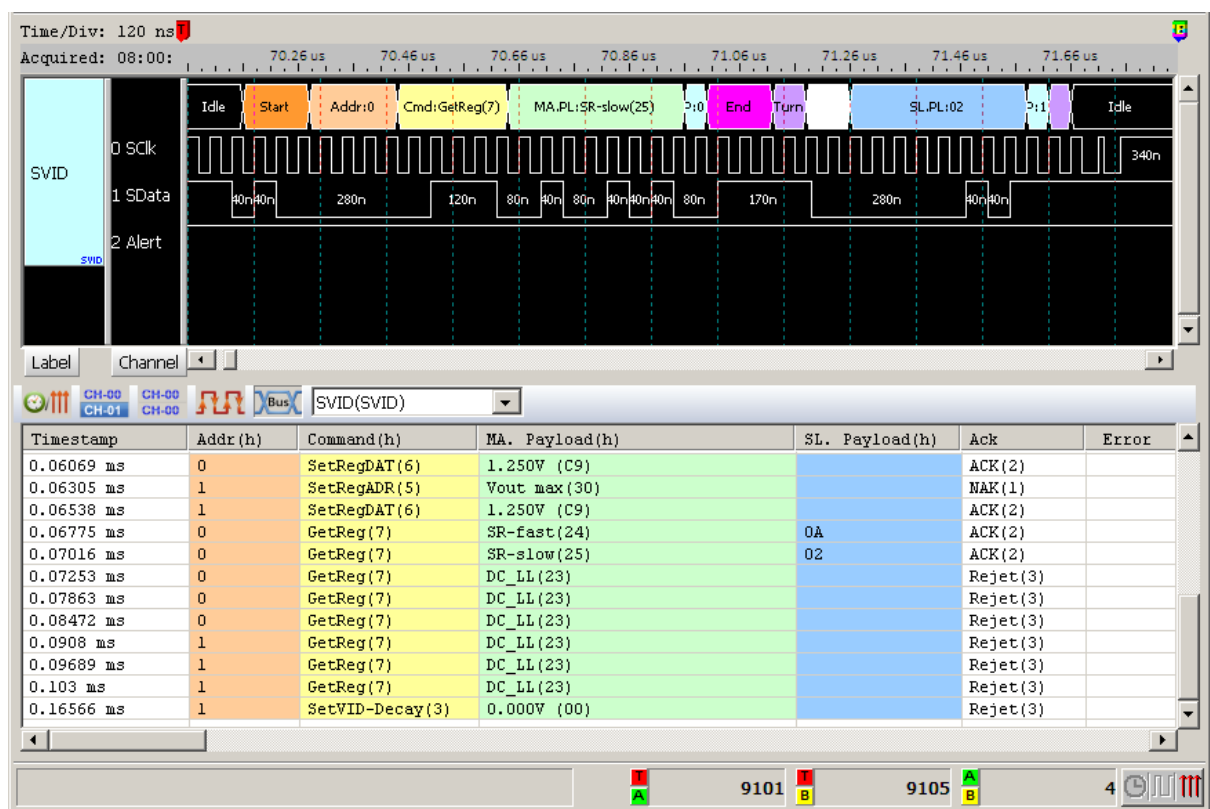
**VR Controller:** Set single or multiple controllers in current VR. Set different address.

**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

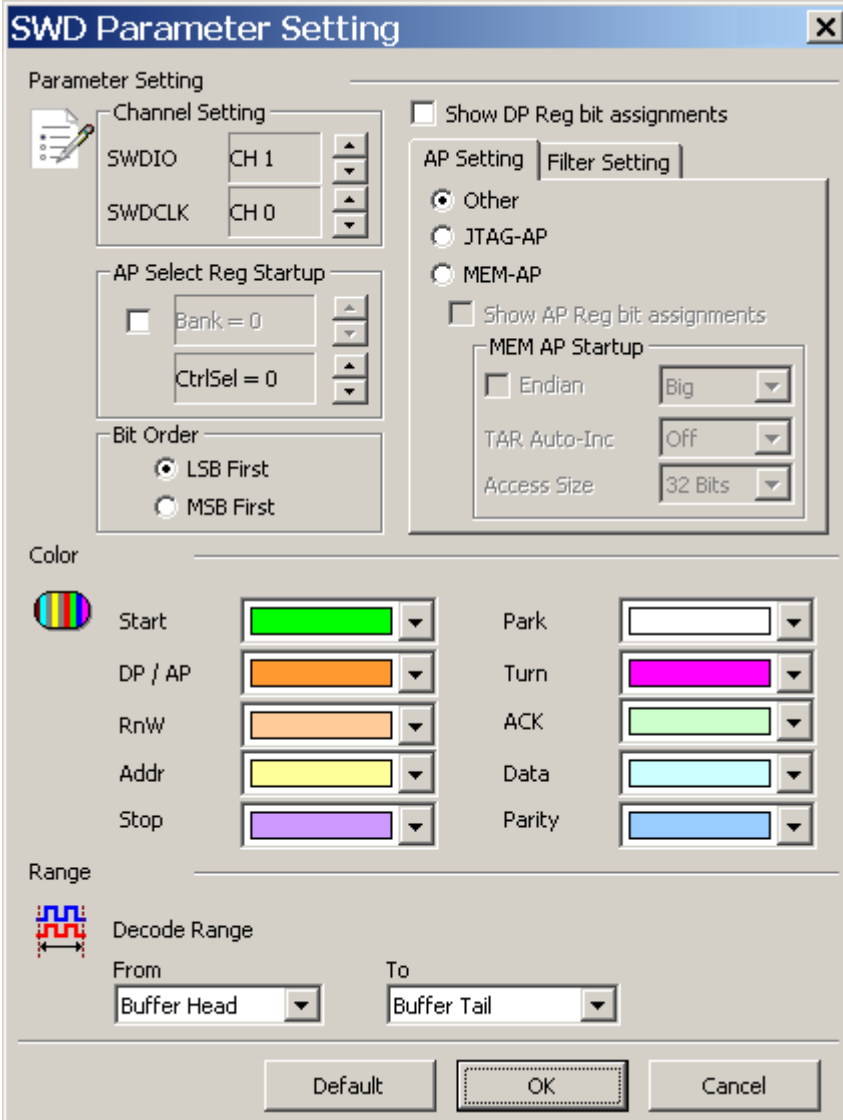
below.



## 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



The image shows the 'SWD Parameter Setting' dialog box. It has a title bar with a close button. The main area is divided into several sections:

- Parameter Setting**: Contains a 'Channel Setting' section with 'SWDIO' set to 'CH 1' and 'SWDCLK' set to 'CH 0'. Below this is 'AP Select Reg Startup' with 'Bank = 0' and 'CtrlSel = 0'. At the bottom is 'Bit Order' with 'LSB First' selected.
- AP Setting**: Contains a 'Show DP Reg bit assignments' checkbox (unchecked). Below it are radio buttons for 'Other' (selected), 'JTAG-AP', and 'MEM-AP'. There is a 'Show AP Reg bit assignments' checkbox (unchecked).
- Filter Setting**: Contains a 'MEM AP Startup' section with 'Endian' set to 'Big', 'TAR Auto-Inc' set to 'Off', and 'Access Size' set to '32 Bits'.
- Color**: A section with a color wheel icon and two columns of color selection boxes. The first column includes 'Start' (green), 'DP / AP' (orange), 'RnW' (light orange), 'Addr' (yellow), and 'Stop' (purple). The second column includes 'Park' (white), 'Turn' (magenta), 'ACK' (light green), 'Data' (cyan), and 'Parity' (blue).
- Range**: Contains a 'Decode Range' section with 'From' set to 'Buffer Head' and 'To' set to 'Buffer Tail'.

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

**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.



MEM-AP		Select	RnW	Address (h)	ACK	Data
<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.

MEM-AP					
<input checked="" type="checkbox"/> Show AP Reg bit assignments					
MEM AP Startup					
<input checked="" type="checkbox"/> Endian Big					
TAR Auto-Inc Single					
Access Size 16 Bits					
AP	Read	DRW Register (C)	OK	00 00 00 0D	TAR Address = E000EFF0
				Big-Endian	
				000D Access to E000EFF0	
				0000 Access to E000EFF2	
AP	Read	DRW Register (C)	OK	00 00 00 E0	TAR Address = E000EFF2
				Big-Endian	
				00E0 Access to E000EFF2	
				0000 Access to E000EFF4	

**Filter Setting:** Filter the unwanted Registers.

AP Setting

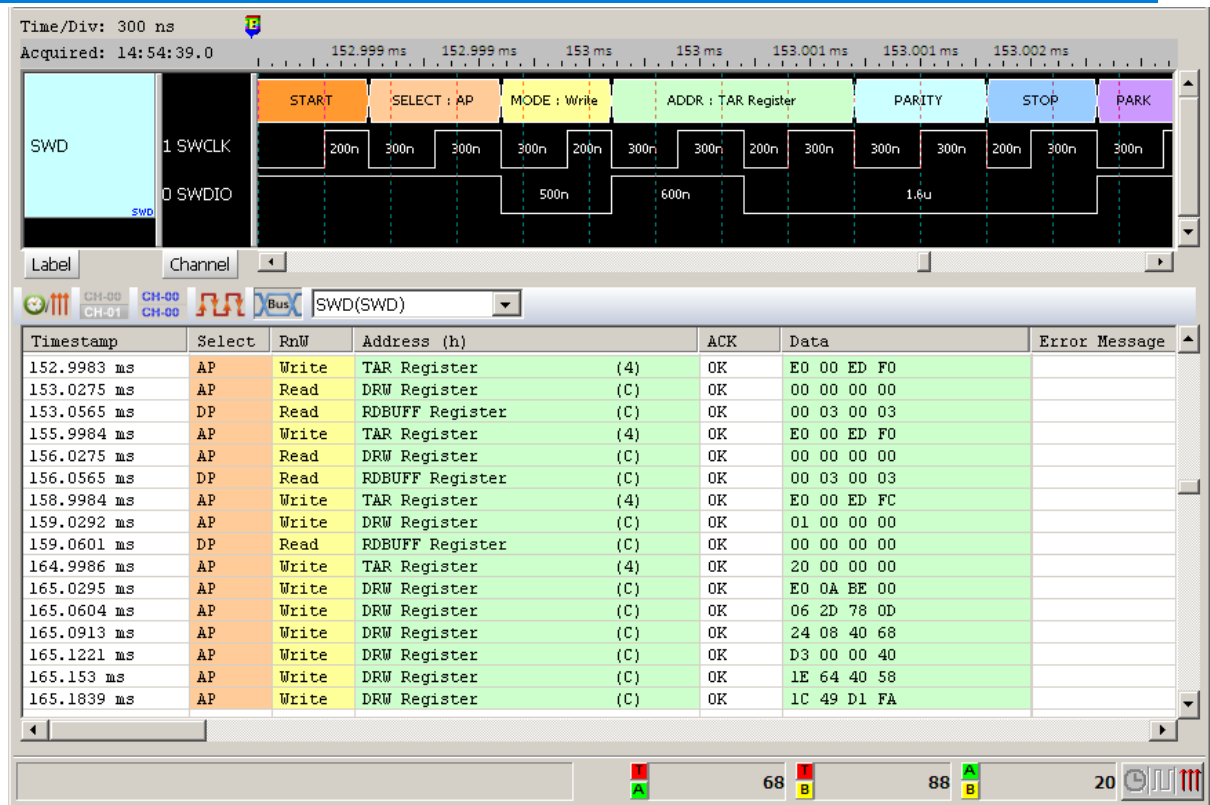
Filter Setting

Register Display List

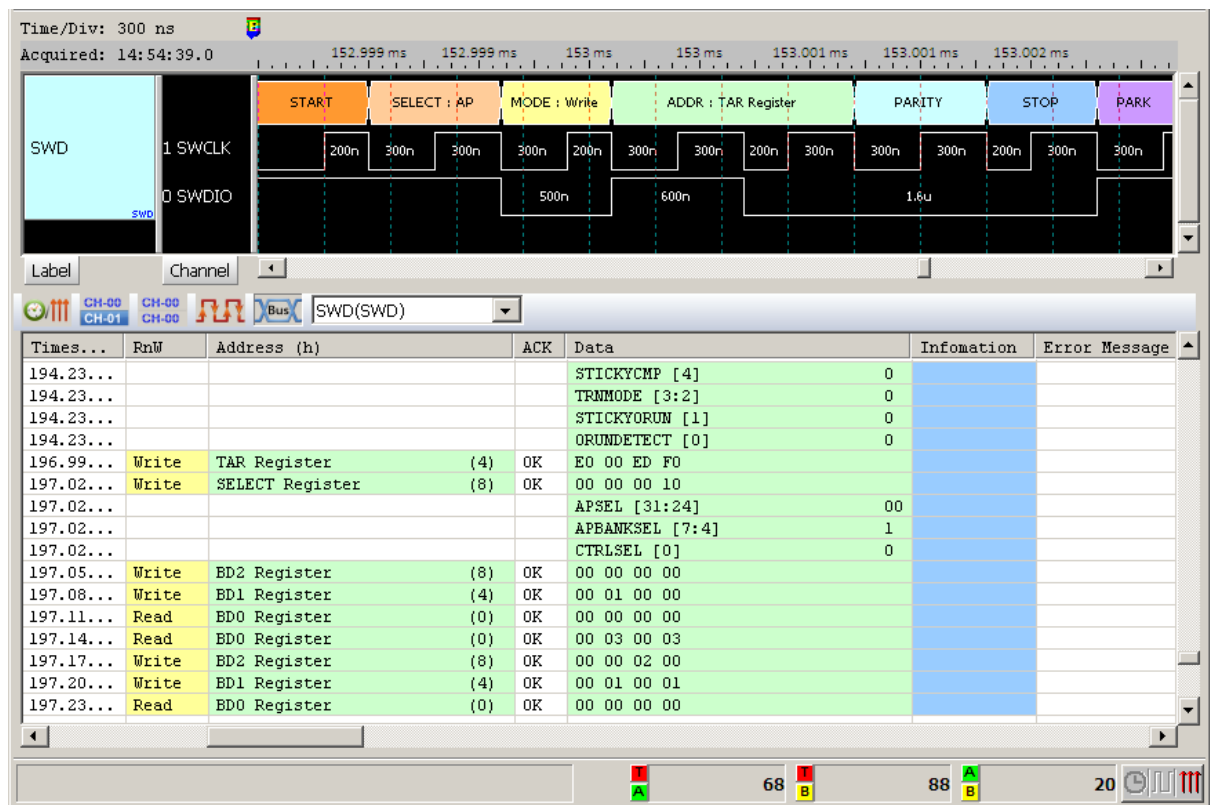
- ☒ DP - ABORT Register
- ☒ DP - IDCODE Register
- ☒ DP - CTRL/STAT Register
- ☒ DP - WCR Register
- ☒ DP - SELECT Register
- ☒ DP - RESEND Register
- ☒ DP - ROUTESEL Register
- ☒ DP - RDRI IFF Register

## Result

Display the decoded waveform without the MEM-AP bit assignments.



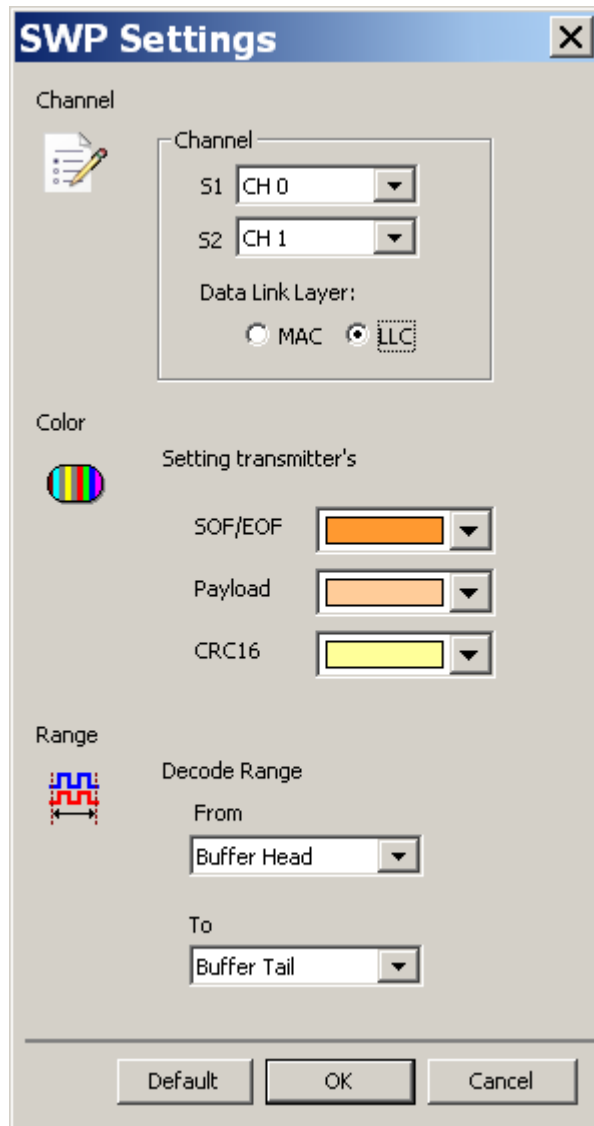
Display the decoded waveform with the MEM-AP bit assignments.



## SWP

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

### Settings

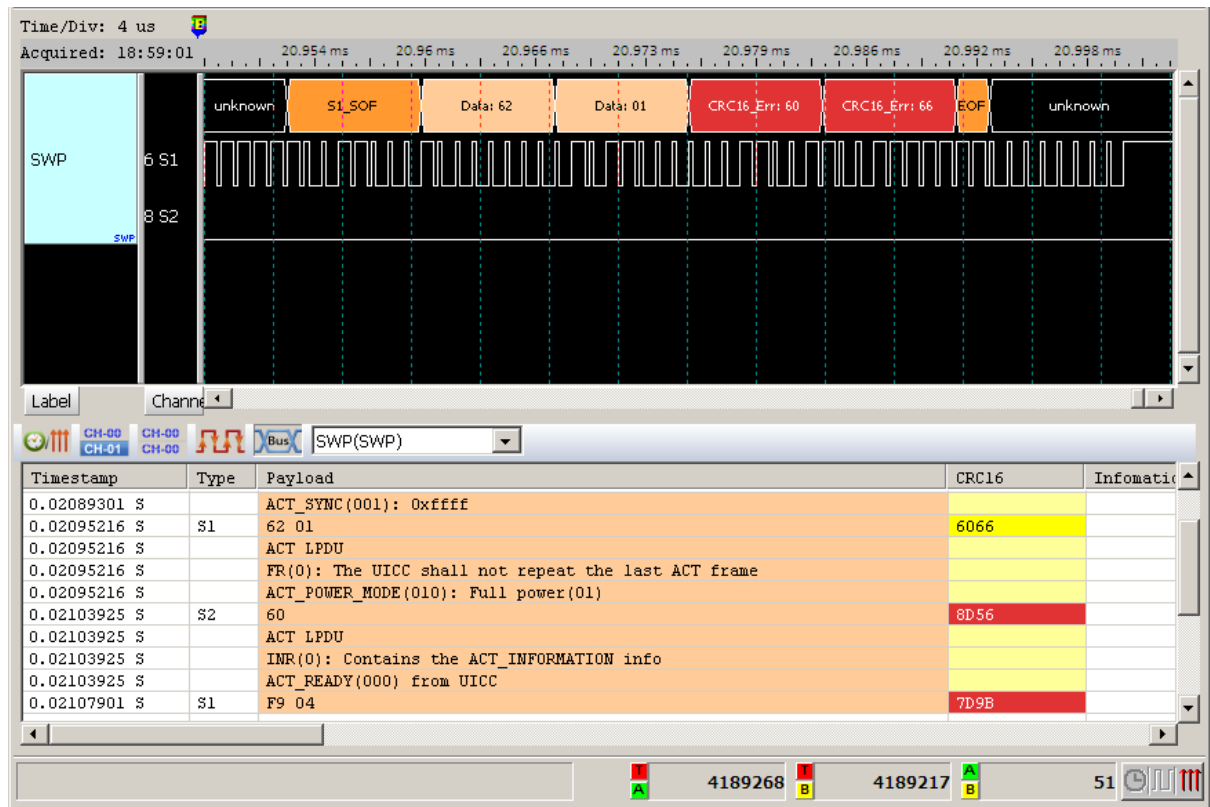


**S1:** I/O data.

**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.

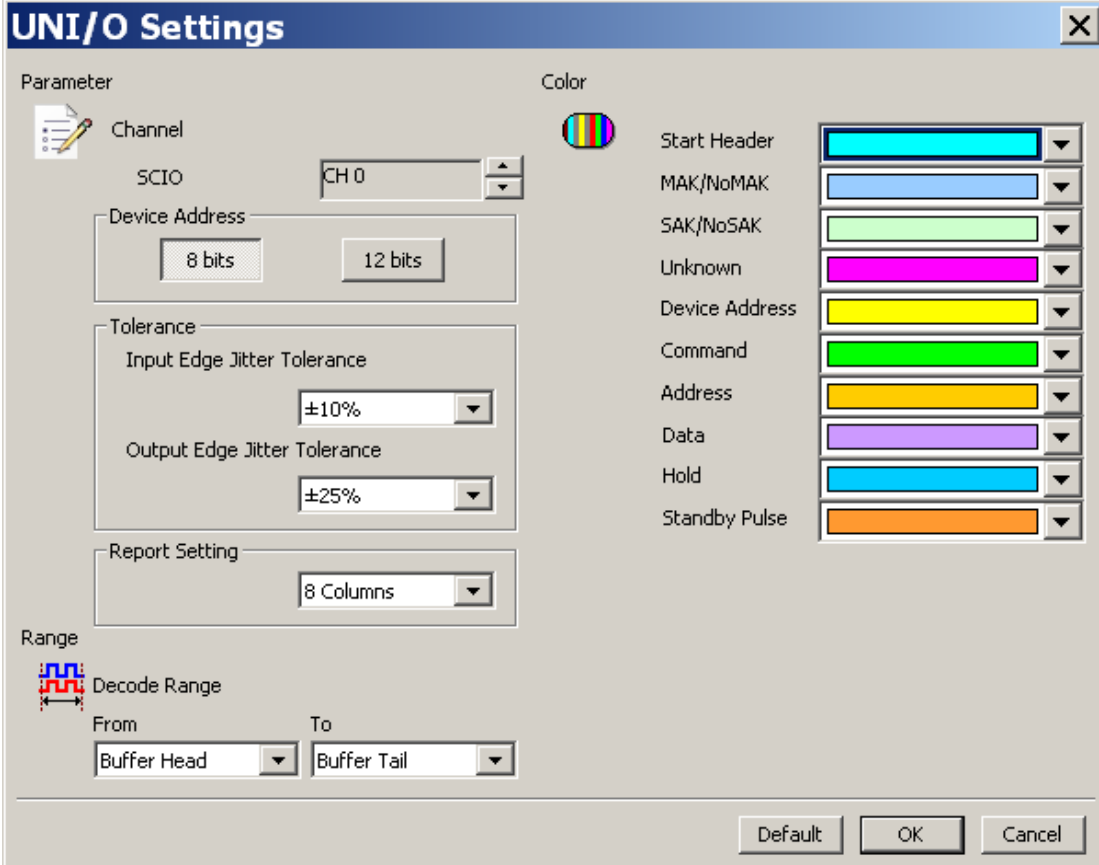
**Line Wrap Data:** Use 1-2 value (hex) as header of data.



## UNI/O

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

### Settings



The image shows the 'UNI/O Settings' dialog box. It has a title bar with a close button. The dialog is divided into several sections:

- Parameter:**
  - Channel:** A dropdown menu showing 'SCIO' and 'CH 0'.
  - Device Address:** Two buttons labeled '8 bits' and '12 bits'.
  - Tolerance:**
    - Input Edge Jitter Tolerance:** A dropdown menu showing '±10%'.
    - Output Edge Jitter Tolerance:** A dropdown menu showing '±25%'.
  - Report Setting:** A dropdown menu showing '8 Columns'.
- Color:** A color selection icon (a circle with vertical bars of different colors).
- Color List:** A list of color-coded fields with dropdown menus:
  - Start Header
  - MAK/NoMAK
  - SAK/NoSAK
  - Unknown
  - Device Address
  - Command
  - Address
  - Data
  - Hold
  - Standby Pulse
- Range:**
  - Decode Range:** A section with 'From' and 'To' dropdown menus. 'From' is set to 'Buffer Head' and 'To' is set to 'Buffer Tail'.

At the bottom right, there are three buttons: 'Default', 'OK', and 'Cancel'.

**Channel:** Show the selected channels (SCIO CH0).

**Device Address:** Set data bits for the device address.

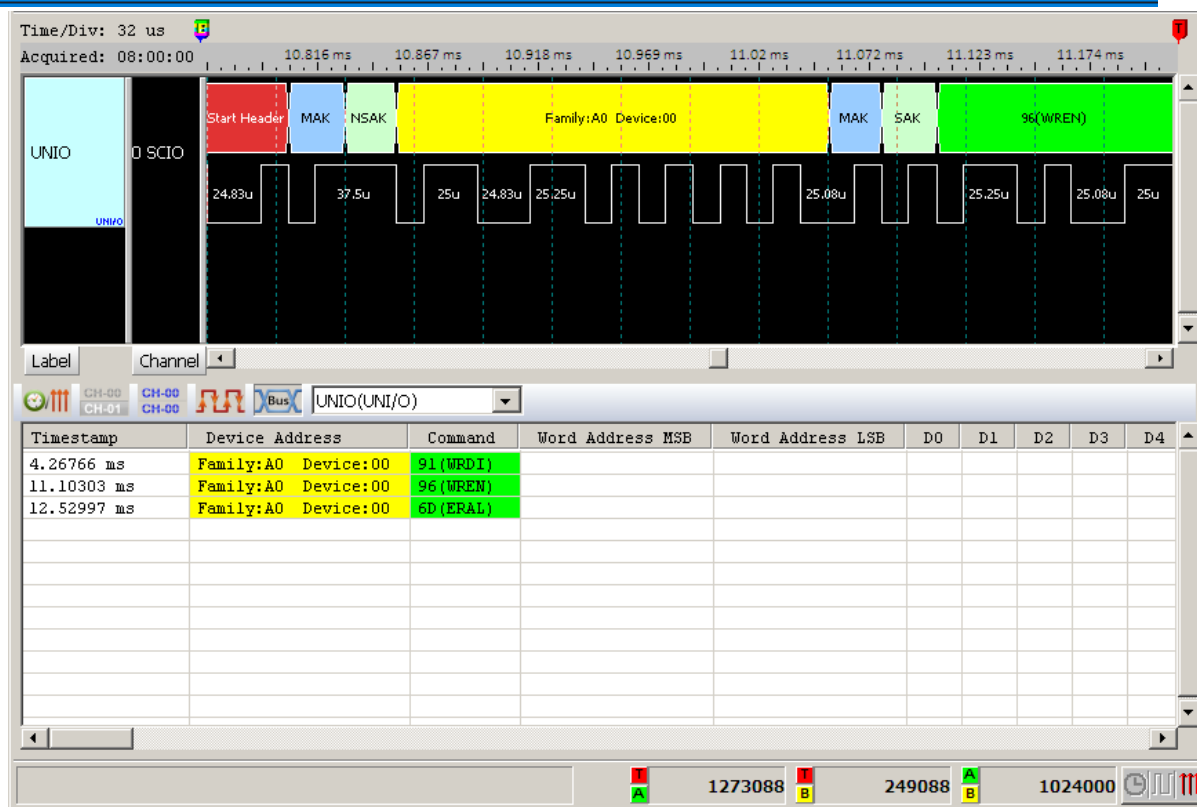
**Tolerance:** Set the Input / Output Edge Jitter Tolerance, ±10% / ±25% default.

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

### Result

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

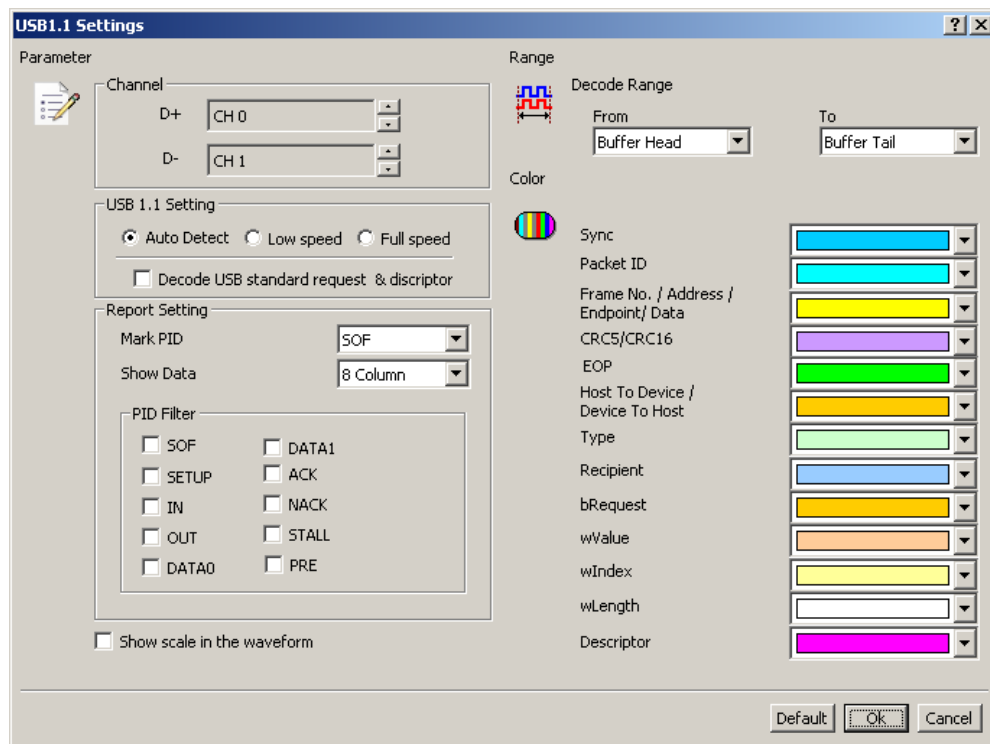




## USB1.1

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

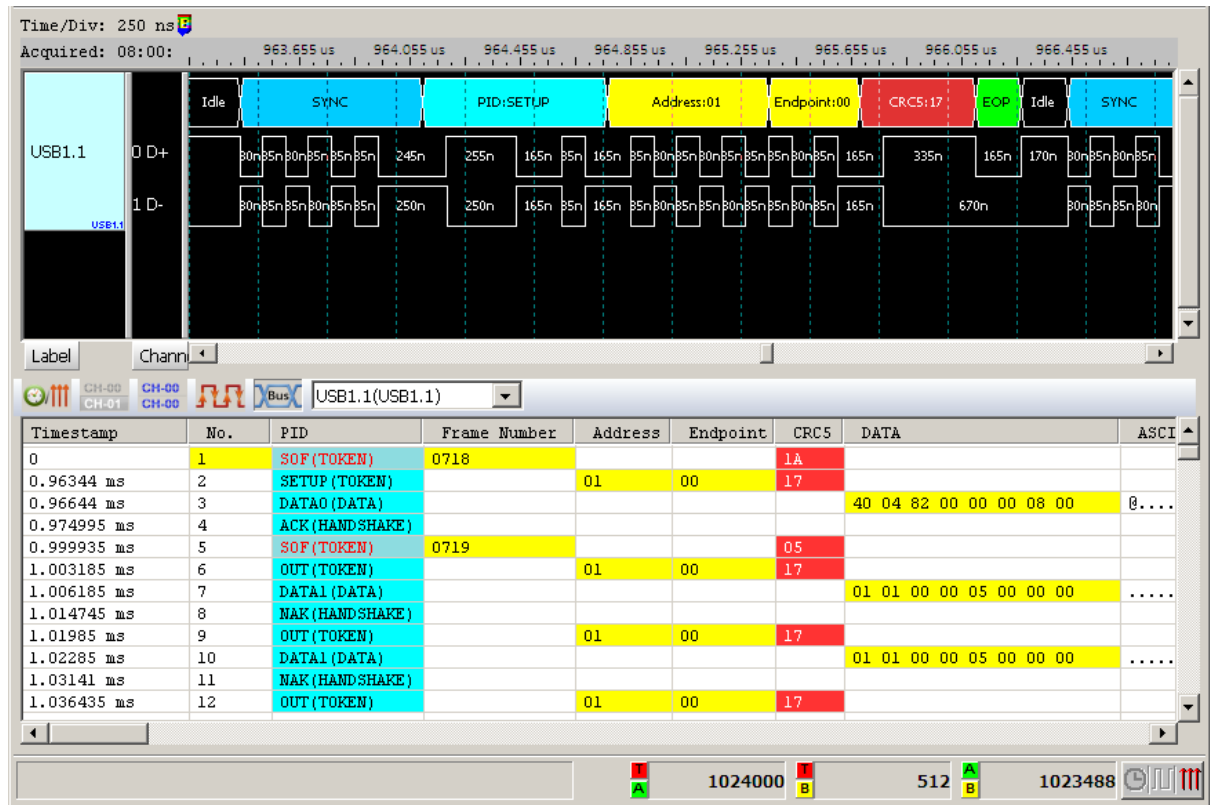


**Channel:** Show the selected channels (D+ CH0 and D- CH1).

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

### Result

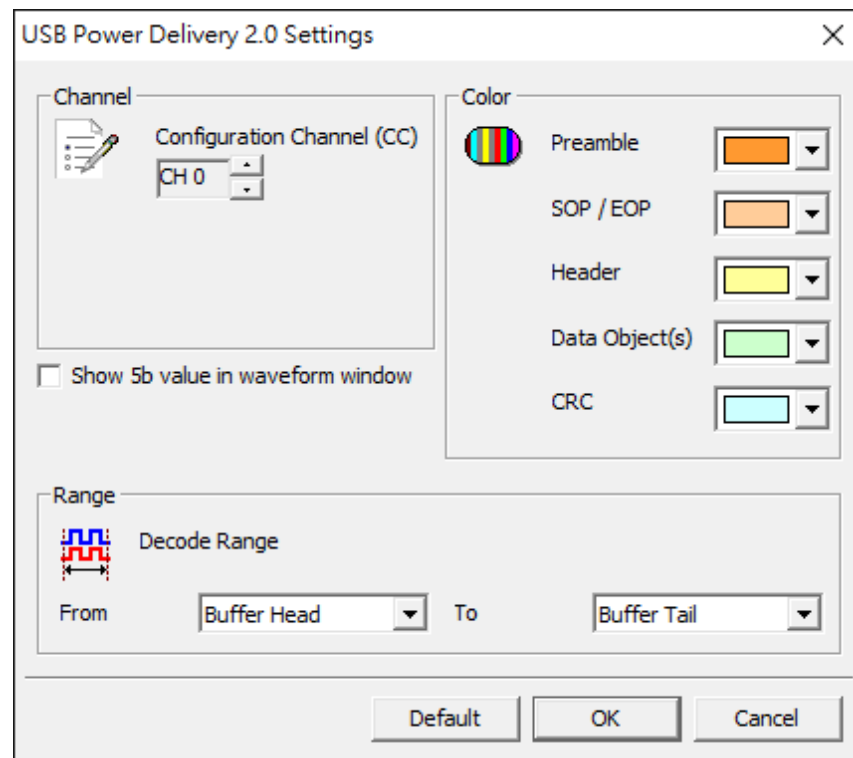
Click **OK** to run the USB1.1 decode and see the result on the Waveform Window.



## USB PD 2.0

USB PD (Power Delivery) 2.0 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 100W, and the users can charge the device by supporting USB Type-C connector.

### Settings

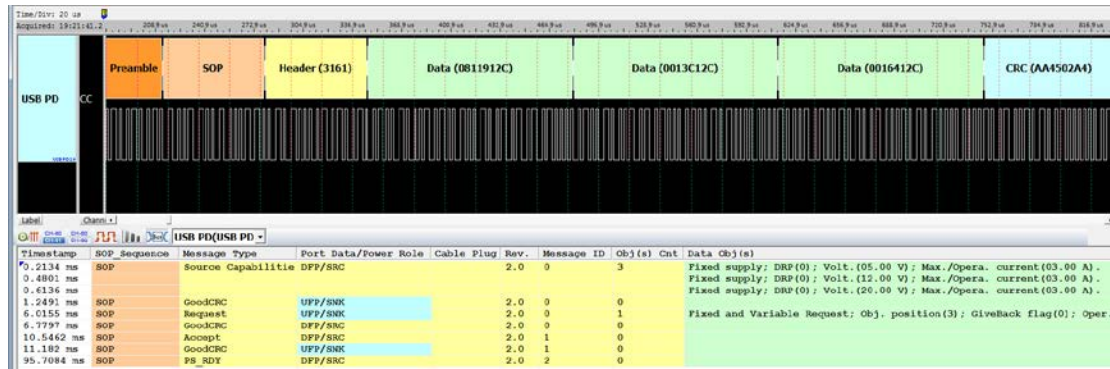


**Channel:** Set Configuration Channel (CC).

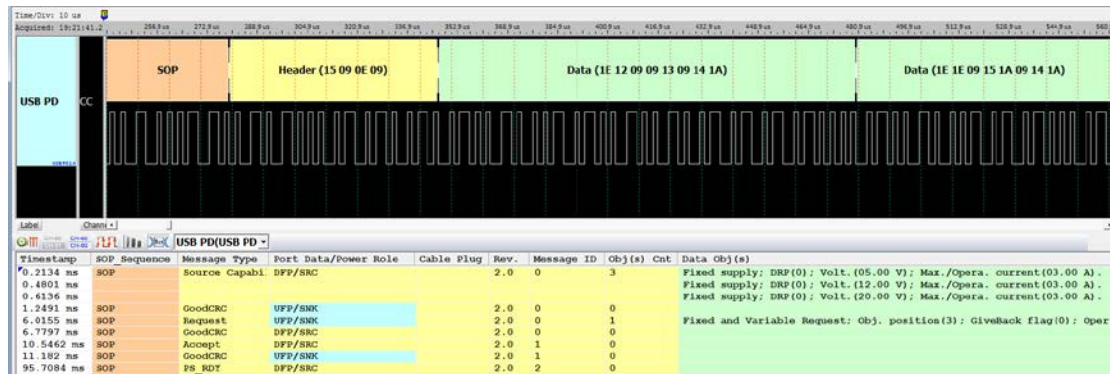
**Show 5b value in waveform window:** Show 4b or 5b value in the waveform.

## Result

Show 4b value.



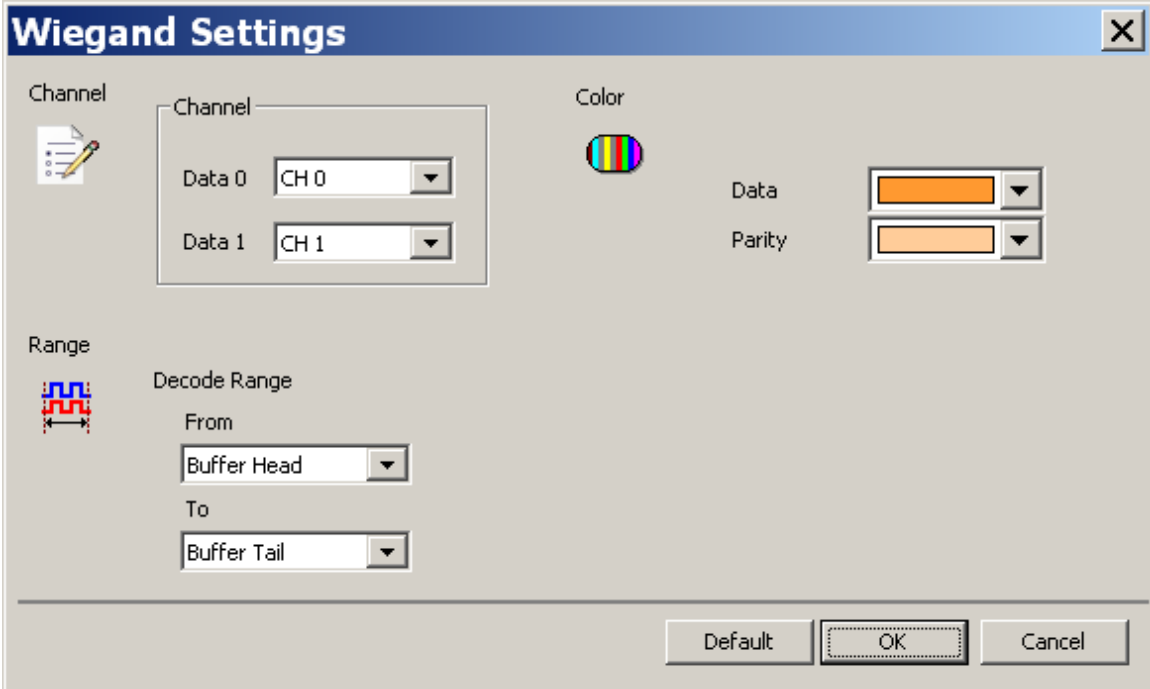
Show 5b value.



## Wiegand

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

### Settings

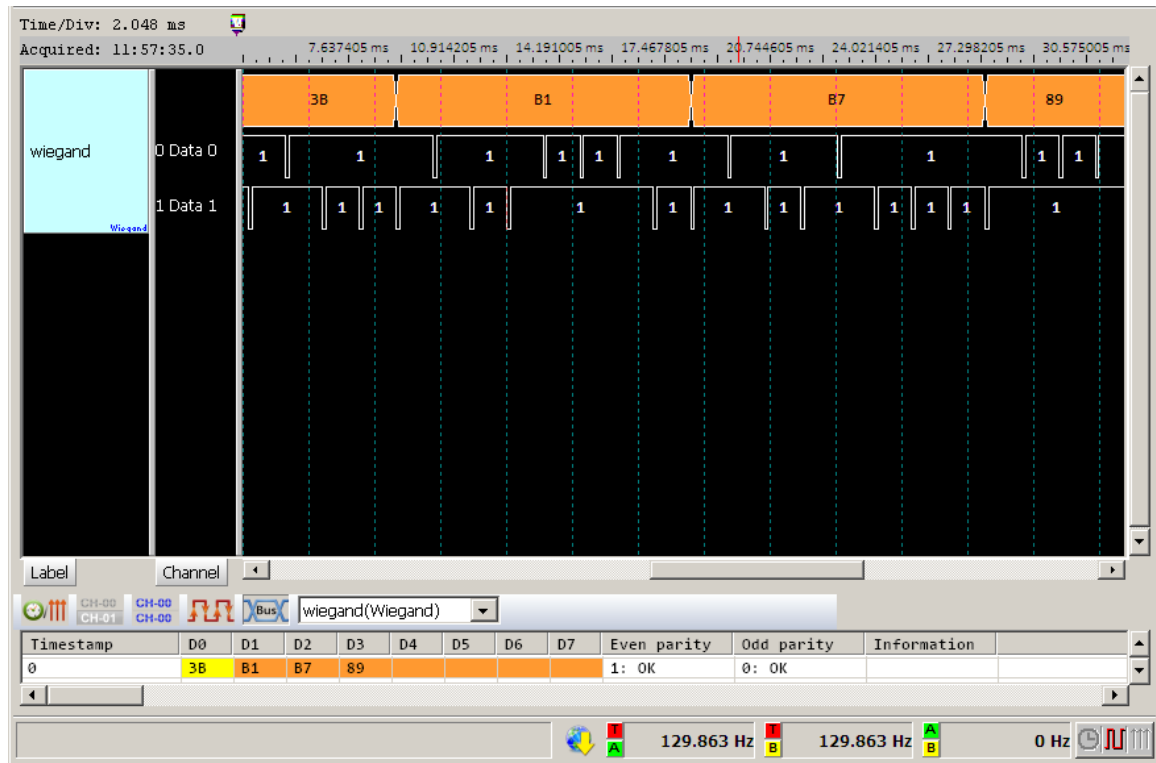


The **Wiegand Settings** dialog box is used to configure the Wiegand interface. It features a blue title bar with a close button. The main area is divided into several sections:   
 - **Channel**: Includes a notepad icon and a group box labeled 'Channel' containing 'Data 0' (set to 'CH 0') and 'Data 1' (set to 'CH 1') dropdown menus.   
 - **Color**: Includes a color wheel icon and two dropdown menus for 'Data' (orange) and 'Parity' (light orange).   
 - **Range**: Includes a waveform icon and a 'Decode Range' section with 'From' (set to 'Buffer Head') and 'To' (set to 'Buffer Tail') dropdown menus.   
 - **Buttons**: At the bottom right are 'Default', 'OK', and 'Cancel' buttons.

**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 ULA-4G36 AL 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	⊙	⊙		

## CAN Trigger

The settings dialog box below.

Mode	Min. S/R	Max. S/R	Available ch.	Min. Mem.	Max. Mem.
<b>CAN Trigger</b>	1Hz	200MHz	Adjustable	256	Adjustable
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Auto
<b>CAN Trigger-36</b>	1Hz	200MHz	Adjustable	256	2M
CAN Trigger-18	1Hz	200MHz	Adjustable	256	4M
CAN Trigger-12	1Hz	200MHz	Adjustable	256	6M
CAN Trigger-9	1Hz	200MHz	Adjustable	256	8M
CAN Trigger-6	1Hz	200MHz	Adjustable	256	12M
CAN Trigger-4	1Hz	200MHz	Adjustable	256	18M
CAN Trigger-2	1Hz	200MHz	Adjustable	256	36M
CAN Trigger-1	1Hz	200MHz	Adjustable	256	72M
I2C Trigger	1Hz	200MHz	Adjustable	256	Adjustable
I2S Trigger	1Hz	200MHz	Adjustable	256	Adjustable
SPI Trigger(800M)-9	800MHz	800MHz	9	256	8M
SPI Trigger	1Hz	200MHz	Adjustable	256	Adjustable

### Trigger Settings

Select CAN Trigger-36 where "-36" means 36 channels and the sample rate is always 10 times of data rate, then click OK. Click the Trigger Settings button (  ) on Toolbar (or Device menu) to show the CAN Trigger Settings dialog box below.

**CAN Trigger Settings**

Data Rate: 400K Channel: 0

Trigger On: Start of Frame

☒ CAN\_L ☐ CAN\_H

☐ 11 Bits ID ☐ 29 Bits ID 0h

DATA Length: 1 DATA Compare: =

DATA1: xxh DATA2: xxh  
DATA3: xxh DATA4: xxh  
DATA5: xxh DATA6: xxh  
DATA7: xxh DATA8: xxh

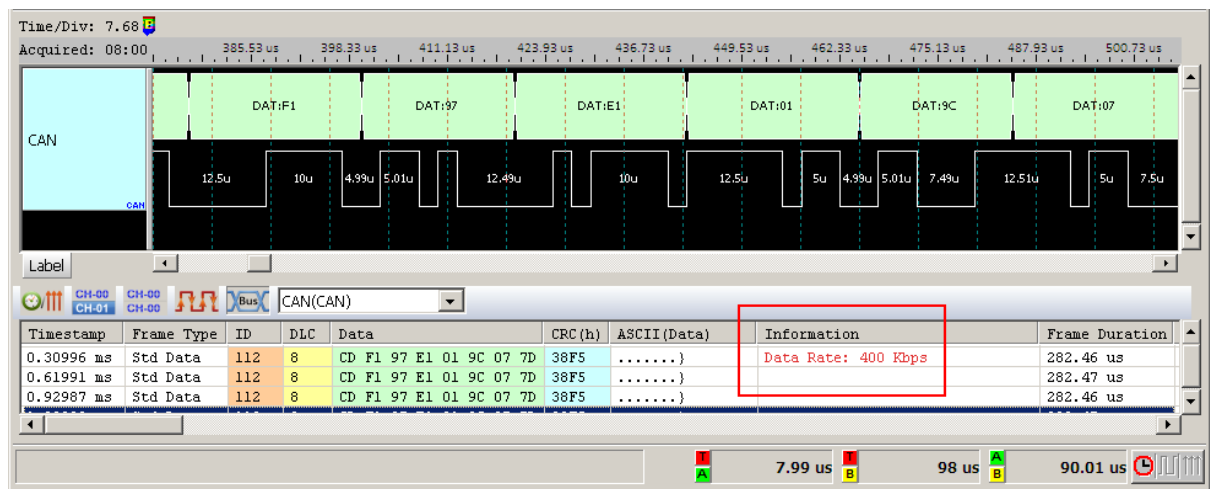
Trigger on start of frame.

☐ Pre-Trigger Pass Count: 0

Load Save Clear All OK Cancel

**Data Rate:** Select a default data rate or run the CAN decode to measure the actual

data rate to enter (highly recommended). When the selected data rate is about 5% deviation from the actual data rate will lead to CAN trigger failure.



**Channel:** Select channels.

**Trigger On:** Select the data field to trigger on:

Start of Frame

ID Match Data Frame

Remote Frame

Error Frame

Overload Frame

Stuffing Error

CRC Error

Data Value

Missing ACK

End of Frame

ID Match & Data Value

**CAN\_H/CAN\_L:** Select CAN\_H or CAN\_L as the trigger channel.

**11 Bits ID/29 Bits ID:** Data bits of the Identification field.

**DATA Length:** The number of transmitted data, by Byte unit.

**DATA Compare:** To compare the data.

**DATA1 ~ DATA8:** Data like binary codes (e.g. 01000001b), decimal codes (e.g. 65), hexadecimal codes (e.g. 41h). If trigger on Date Value and need to pass some data, please enter XX. For example: To trigger the data 38h in the third byte of data frame, enter:

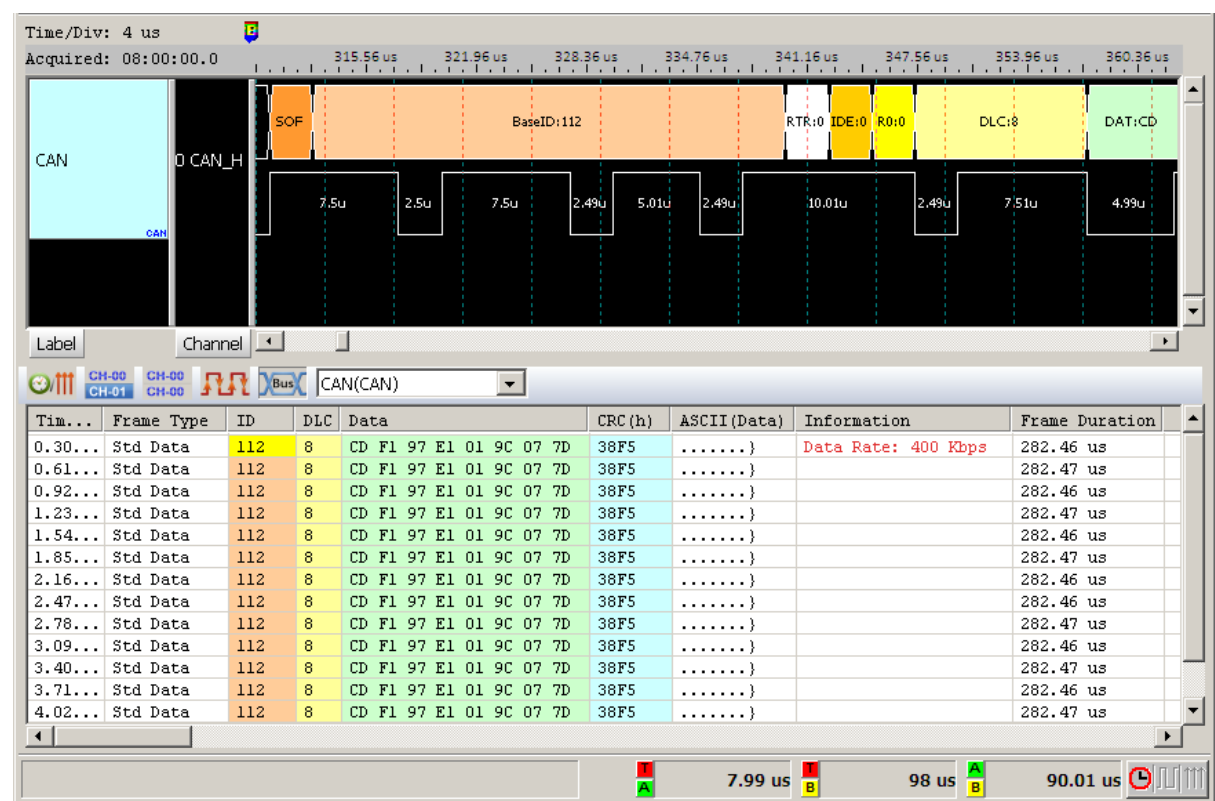
DATA Length = 3

DATA1 = XX

DATA2 = XX

DATA3 = 38h

## Result




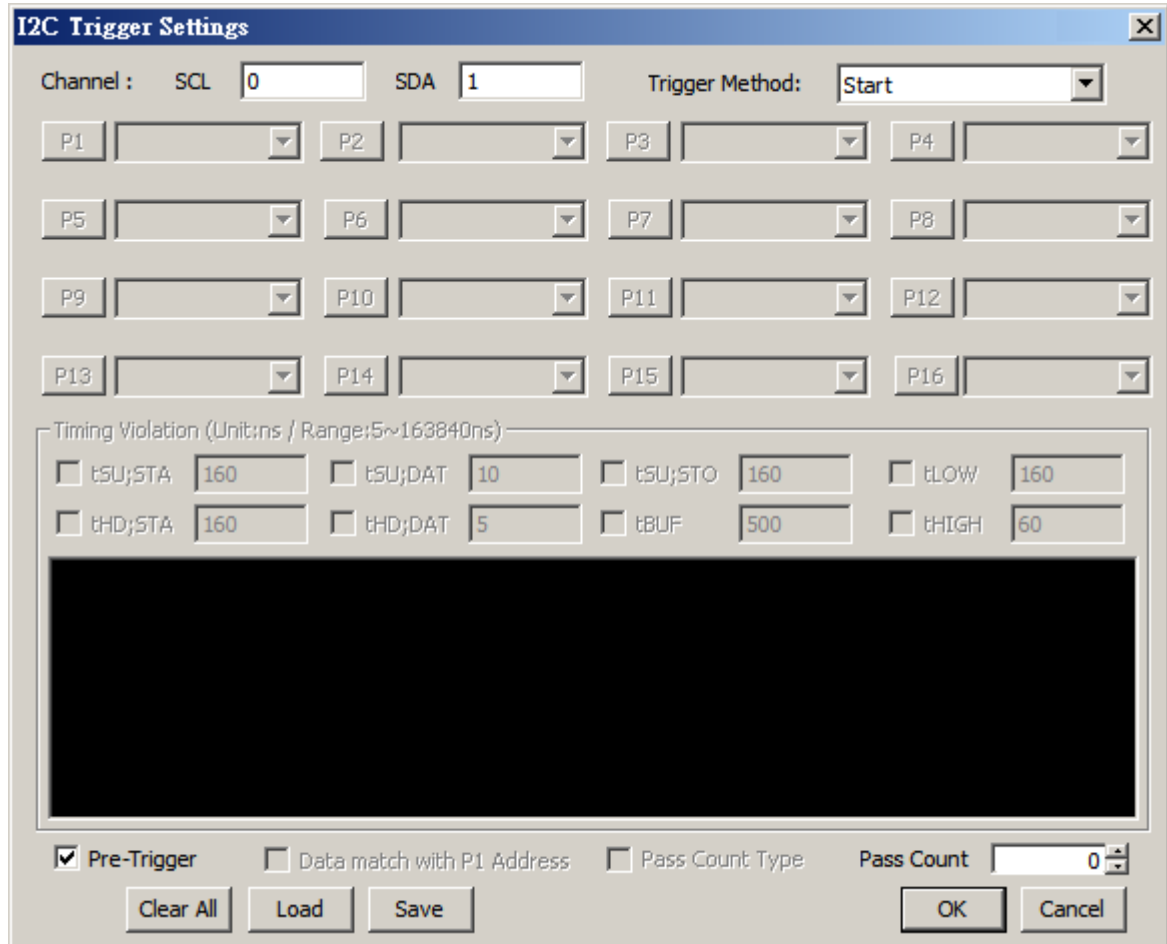
## I<sup>2</sup>C Trigger

The settings dialog box below.

Mode	Min. S/R	Max. S/R	Available ch.	Min. Mem.	Max. Mem.
800M	800MHz	800MHz	9	256	8M
400M	400MHz	400MHz	18	256	4M
200M	1Hz	200MHz	Adjustable	256	Adjustable
UART Trigger	Baud Rate x 16	Baud Rate x 16	Adjustable	256	Adjustable
CAN Trigger	Data Rate x 10	Data Rate x 10	Adjustable	256	Adjustable
I <sup>2</sup> C Trigger	1Hz	200MHz	Adjustable	256	Adjustable
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Auto
I <sup>2</sup> C Trigger-36	1Hz	200MHz	Adjustable	256	2M
I <sup>2</sup> C Trigger-18	1Hz	200MHz	Adjustable	256	4M
I <sup>2</sup> C Trigger-12	1Hz	200MHz	Adjustable	256	6M
I <sup>2</sup> C Trigger-9	1Hz	200MHz	Adjustable	256	8M
I <sup>2</sup> C Trigger-6	1Hz	200MHz	Adjustable	256	12M
I <sup>2</sup> C Trigger-4	1Hz	200MHz	Adjustable	256	18M
I <sup>2</sup> S Trinner	1Hz	200MHz	Adjustable	256	Adjustable

### Trigger Settings

Select I<sup>2</sup>C Trigger-36 where "-36" means 36 channels and click OK. Then click the Trigger Settings button (  ) on Toolbar (or Device menu) to show the I<sup>2</sup>C Trigger Settings dialog box below.



The dialog box is titled "I2C Trigger Settings". It contains the following elements:

- Channel:** SCL (0) and SDA (1) input fields.
- Trigger Method:** A dropdown menu currently set to "Start".
- P1 through P16:** A grid of 16 dropdown menus, each with a label (P1 to P16) and a small downward arrow.
- Timing Violation (Unit:ns / Range:5~163840ns):** A section containing several checkboxes and input fields:
  - ☐ tSU;STA: 160
  - ☐ tSU;DAT: 10
  - ☐ tSU;STO: 160
  - ☐ tLOW: 160
  - ☐ tHD;STA: 160
  - ☐ tHD;DAT: 5
  - ☐ tBUF: 500
  - ☐ tHIGH: 60
- Pre-Trigger:** A checked checkbox.
- Data match with P1 Address:** An unchecked checkbox.
- Pass Count Type:** An unchecked checkbox.
- Pass Count:** A numeric input field set to 0.
- Buttons:** "Clear All", "Load", "Save", "OK", and "Cancel".

**Channel:** Select two channels for the serial clock (SCK) and the serial data (SDA).

**Trigger Method:** Offer nine models for user to choose.

Start

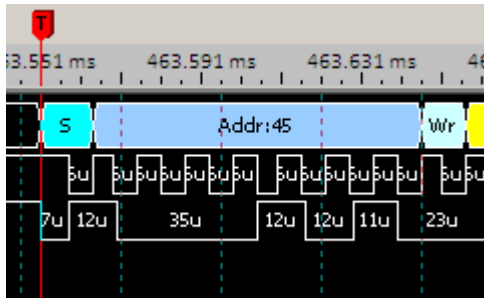
Re-Start

Start or Re-Start

Stop

Missing Ack (Not Acknowledge (NACK))

For the first five ways of Trigger Method, Cursor T will stay at the beginning of the condition as below if the trigger succeeds.



**Match Sequentially:** Similar to the Sequence Trigger, the I<sup>2</sup>C Trigger has up to 16 levels of sequential conditions below.

**I2C Trigger Settings**

Channel: SCL 0 SDA 1 Trigger Method: Match Sequentially

P1 Next P2 ThenIf P3 Next P4 Next

P5 Next P6 ThenIf P7 ThenIf P8 ThenIf

P9 ThenIf P10 Next P11 Next P12 Then Trigger

P13 Then Trigger P14 Then Trigger P15 Then Trigger P16 Then Trigger

Timing Violation (Units:ns / Range:5~163840ns)

☐ tSU;STA 160 ☐ tSU;DAT 10 ☐ tSU;STO 160 ☐ tLOW 160

☐ tHD;STA 160 ☐ tHD;DAT 5 ☐ tBUF 500 ☐ tHIGH 60

P1 P2 P3 P4 P5 P6 P7 P8 P9 P10 P11 P12 →

D D D D D D D D D D D D

= = = = = = = = = = = =

xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh

☒ Pre-Trigger ☐ Data match with P1 Address ☐ Pass Count Type Pass Count 0

Clear All Load Save OK Cancel

Click any condition (P1 ~ P16) and the I<sup>2</sup>C Value Setting dialog box below will show.

**Address:**

**I2C Value Setting**

☒ Address (7-bit addressing)

☐ Write Only

☐ Read Only

☒ Read or Write

☐ Include R/W in Address

7	6	5	4	3	2	1	R/W
0	1	0	0	0	1	1	X

= 46h

☐ Check Acknowledge

☒ ACK

☐ NACK

Data/Address:

☒ = ☐ > ☐ >=

☐ != ☐ < ☐ <=

OK Cancel

Write Only, Read Only or Read or Write. When Address is checked, it is the address will be analyzed. Address starts from bit 1 without including R/W in address.

**I2C Value Setting**

☒ Address (7-bit addressing)

☐ Write Only

☐ Read Only

☒ Read or Write

☒ Include R/W in Address

7	6	5	4	3	2	1	R/W
0	0	1	0	0	0	1	X

= 23h

☐ Check Acknowledge

☒ ACK

☐ NACK

Data/Address:

☒ = ☐ > ☐ >=

☐ != ☐ < ☐ <=

OK Cancel

Address starts from bit R/W including R/W in address.

**I2C Value Setting**

☐ Address (7-bit addressing)

☐ Write Only

☐ Read Only

☒ Read or Write

☐ Include R/W in Address

7	6	5	4	3	2	1	R/W
X	X	X	X	X	X	X	X

= 00h

☒ Check Acknowledge

☒ ACK

☐ NACK

Data/Address:

☒ = ☐ > ☐ >=

☐ != ☐ < ☐ <=

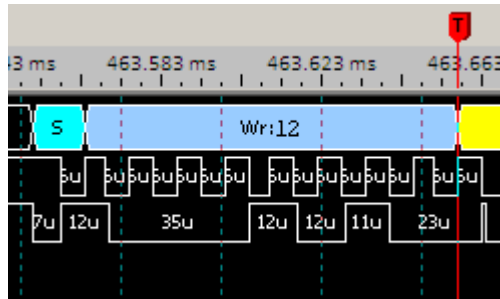
OK Cancel

If Address is not checked, the data (Data 12h) will be analyzed.



**Check Acknowledge:** ACK (Acknowledge), NACK (Not Acknowledge). If the Check Acknowledge is not checked, it can be either ACK or NACK.

**Data/Address:**



Data or Address can be in binary or hexadecimal code like 00010010b or 12h. Also, you can input X (Don't care) as the value; if the data is 10h, 20h, 30h, you can set X0h or 00XX0000b.

Cursor T will stay at the end of the condition as below if the trigger succeeds.

**All Match:** All input conditions must be satisfied to trigger.

**I2C Trigger Settings**

Channel: SCL 0 SDA 1 Trigger Method: All Match

P1 Used P2 Used P3 Unused P4 Unused  
P5 Unused P6 Unused P7 Unused P8 Unused  
P9 Unused P10 Unused P11 Unused P12 Unused  
P13 Unused P14 Unused P15 Unused P16 Unused

Timing Violation (Unit:ns / Range:5~163840ns)

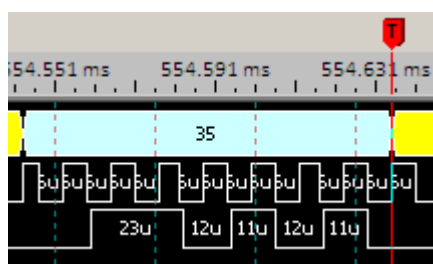
☐ tSU;STA 160 ☐ tSU;DAT 10 ☐ tSU;STO 160 ☐ tLOW 160  
☐ tHD;STA 160 ☐ tHD;DAT 5 ☐ tBUF 500 ☐ tHIGH 60

**P1 P2**  
D D  
= =  
xxh xxh

☒ Pre-Trigger ☐ Data match with P1 Address ☐ Pass Count Type Pass Count 0

Clear All Load Save OK Cancel

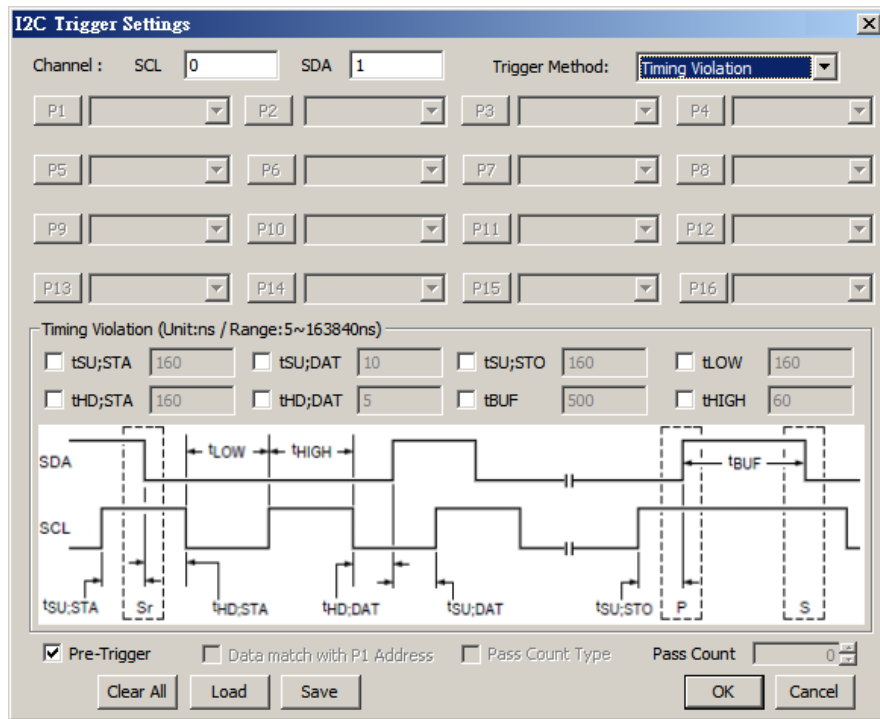
If  $P1 > 30h$  and  $P2 < 40$ , Cursor T will stay at the end of the condition as below if the trigger succeeds.



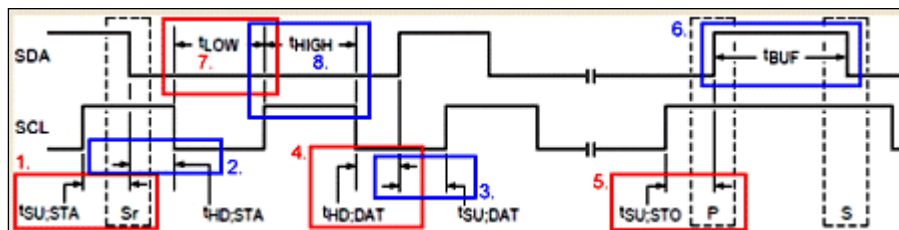
**Any Match:** Any of input conditions is satisfied to trigger.

**Timing Violation:** There are eight timing values provided as the trigger condition. If the timing of the signal is less than the input value, then the instrument will trigger.

For this function, the 200MHz timing analysis is strongly recommended for better resolution.



About each:



**Red 1: tSU;STA:** The setup time of Re-Start, also the time when the clock (SCL) is from high to low.

**Blue 2: tHD;STA:** The hold time of Re-Start, also the time when the clock (SCL) is from high to low.

**Blue 3: tSU;DAT:** The setup time of Data, also the time when the clock (SCL) is from rising to falling.

**Red 4: tHD;DAT:** The hold time of Data, also the time when the clock (SCL) is from falling to rising.

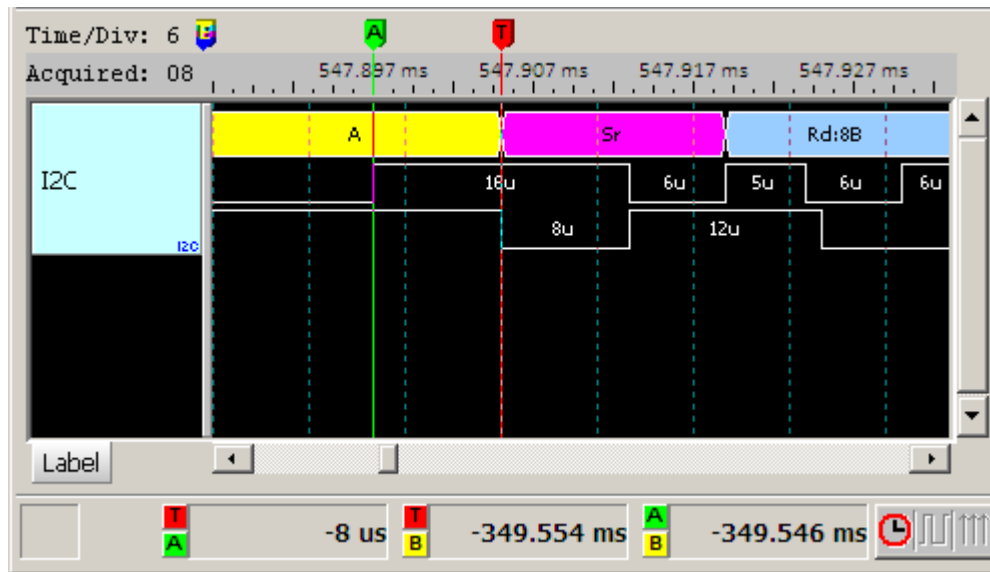
**Red 5: tSU;STO:** The setup time of Stop, also the time when the clock (SCL) is from high to low.

**Blue 6: tBUF:** Bus Free Time, that is between Start and Stop.

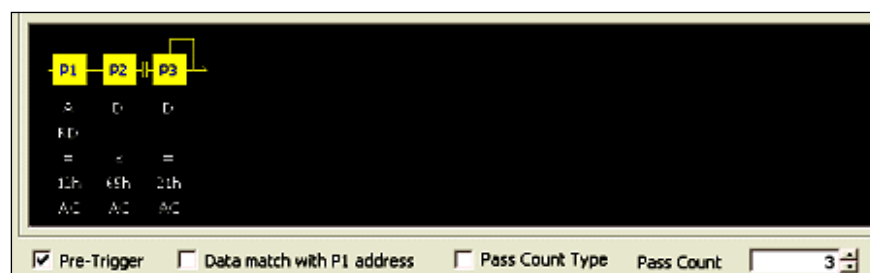
**Red 7: tLOW:** The time the clock (SCL) is low.

**Blue 8: tHIGH:** The time the clock (SCL) is high.

The following figure is an example to trigger the setup time (8005ns) of Re-Start (tSU; STA), where the time between cursor T and cursor A is 8us (8000ns < 8005ns).



**Pass Count:** Pass Count will pass N times that triggers occurred in the non-sequential trigger condition. In the following example, the trigger will pass three triggers in P3 (non-sequential trigger condition) and trigger at the fourth time.



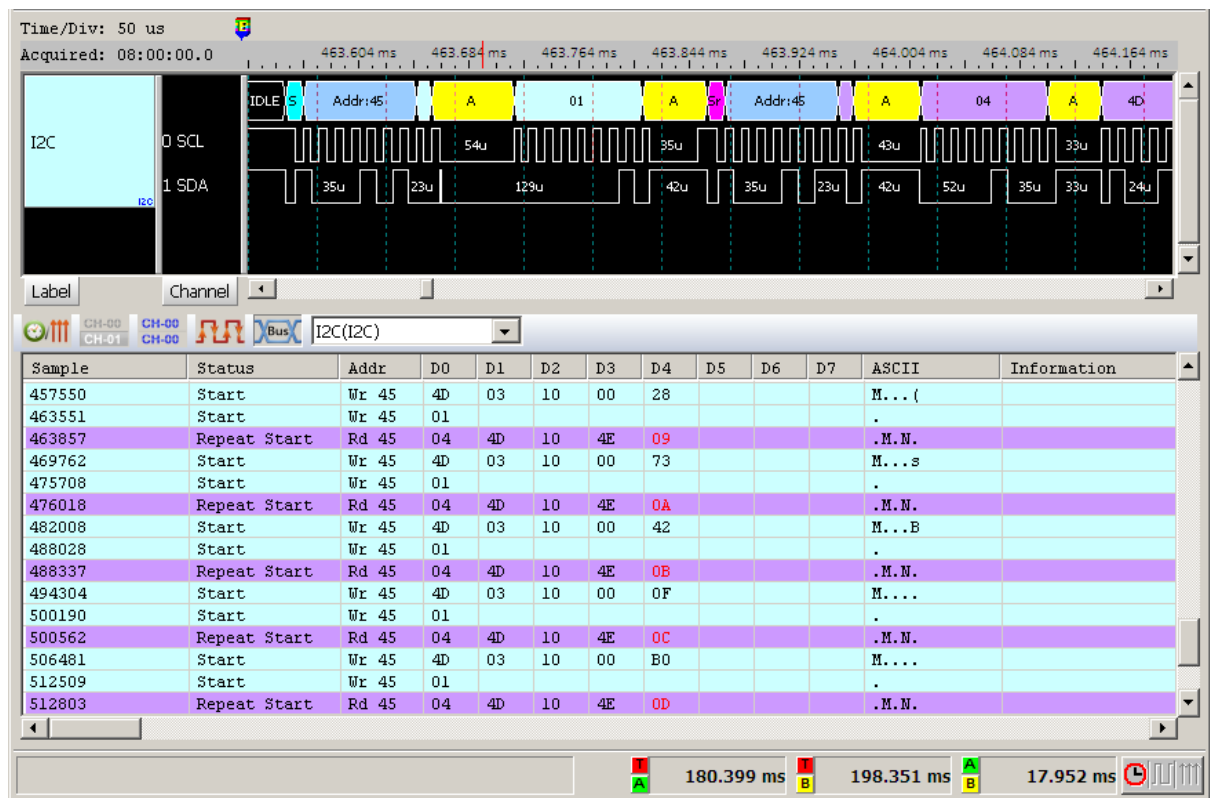
**Pass Count Type:** If Pass Count Type is checked, then the loop will start run from the first trigger condition (P1).



**Data match with P1 address:** P1 must be set as address and followed by data; this function is only available in Match Sequentially. And the first level (P1) parameter must be set the address (not data), the function will be open.

For more information of I<sup>2</sup>C trigger, please refer to I<sup>2</sup>C Application Note.

## Result




## I<sup>2</sup>S Trigger

The settings dialog box below.

Mode	Min. S/R	Max. S/R	Available ch.	Min. Mem.	Max. Mem.
400M	400MHz	400MHz	18	256	4M
200M	1Hz	200MHz	Adjustable	256	Adjustable
UART Trigger	Baud Rate x 16	Baud Rate x 16	Adjustable	256	Adjustable
CAN Trigger	Data Rate x 10	Data Rate x 10	Adjustable	256	Adjustable
I2C Trigger	1Hz	200MHz	Adjustable	256	Adjustable
<b>I2S Trigger</b>	<b>1Hz</b>	<b>200MHz</b>	<b>Adjustable</b>	<b>256</b>	<b>Adjustable</b>
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Auto
I2S Trigger-36	1Hz	200MHz	Adjustable	256	2M
I2S Trigger-18	1Hz	200MHz	Adjustable	256	4M
I2S Trigger-12	1Hz	200MHz	Adjustable	256	6M
I2S Trigger-9	1Hz	200MHz	Adjustable	256	8M
I2S Trigger-6	1Hz	200MHz	Adjustable	256	12M
I2S Trigger-4	1Hz	200MHz	Adjustable	256	18M
SPT Trimmer(800M)-9	800MHz	800MHz	9	256	8M

### Trigger Setting

Select I<sup>2</sup>S Trigger-36 where "-36" means 36 channels and click OK. Then click the Trigger Settings button (  ) on Toolbar (or Device menu) to show the I<sup>2</sup>S Trigger Settings dialog box below.

**I2S Trigger Settings**

Channel: SCK  WS  SD

Data Bits:  Bits Full Scale Voltage:  mV

Method:

Channel: ☒ Both ☐ Left ☐ Right

Pattern Unit: ☒ Value ☐ Voltage ☐ dB

Pattern A:  Pattern B:

☒ = ☐ != ☐ < ☐ > ☐ In Range ☐ Out Range

Duration(# of frames):

Timing Violation:

Default Settings: ☒ Master ☐ Slave ☐ Custom

SCK Period Min.:  ns SCK Period Max.:  ns

SCK High Duty Min.:  ns SCK Low Duty Min.:  ns

Setup Time:  ns Hold Time:  ns

☒ Pre-Trigger Pass Count:

**Channel:** There are three channels: Serial clock (SCK), word select (WS) and serial data (SDA).

**Data Bits:** 1-32 bits, normally it is 8, 12, 16, 24 or 32.

**Channel:** Both, Left or Right.

**Pattern Unit:** Value, Voltage or dB as the trigger condition. If it is Value, only hexadecimal (xxxh) or decimal numbers are allowed.

Trigger method: = (is equal), != (isn't equal), < (less than), > (more than), In Range (between Pattern A and B), or Out Range (out of Pattern A and B)

Duration (# of frames): Continuous frames and trigger when the condition is matched.

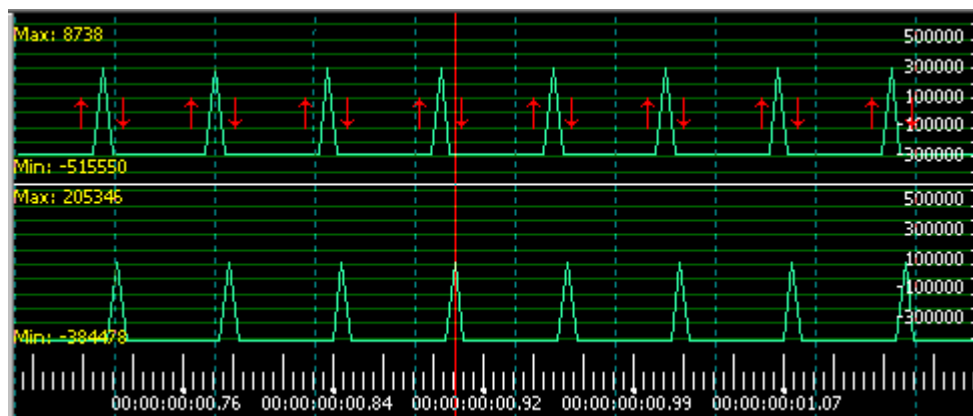
**Method:**

**Data Match:** Trigger when the conditions matched.

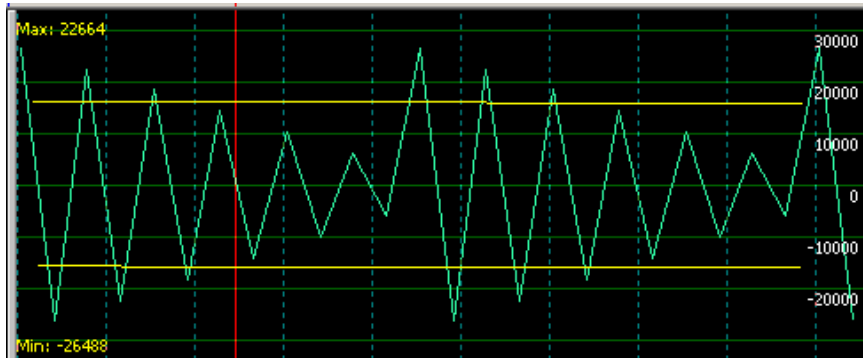
**Rising Edged:** Trigger when it is a rising edge between two patterns.

**Falling Edged:** 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:** There are two default (Master or Slave) and one custom types of timing violation triggers as below.

**Master:** select the default Master values as the trigger condition.

**Slave:** select the default Slave values as the trigger condition.

**Custom:** set trigger parameters manually.

**SCK Period Min.:** Trigger when the value is less than the minimum SCK period.

**SCK Period Max.:** Trigger when the value is more than the maximum SCK period.

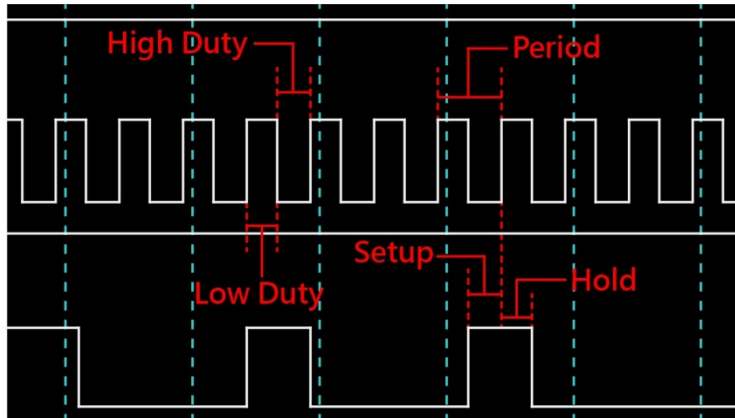
**SCK High Duty Min.:** Trigger when the value is less than the minimum SCK High Duty.

**SCK Low Duty Min.:** Trigger when the value is less than the minimum SCK Low Duty.

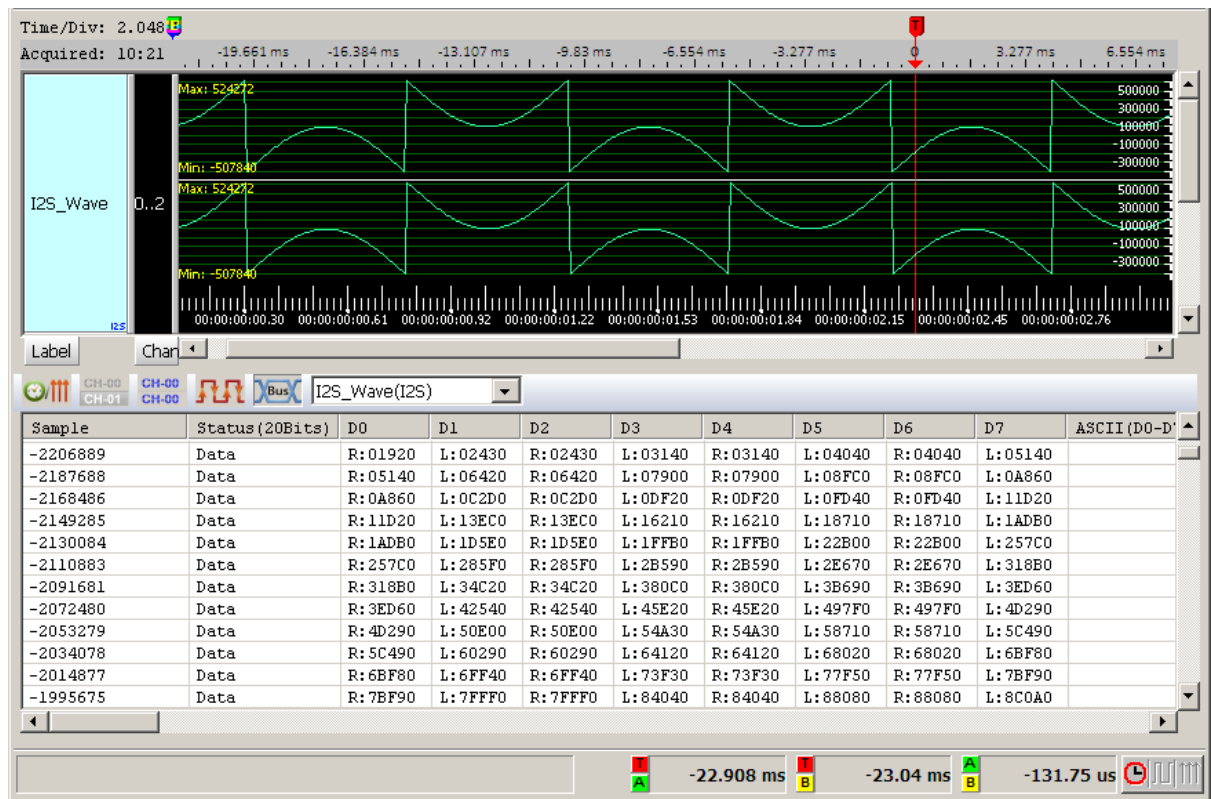
**Setup Time:** Trigger when the value is less than the Setup time.

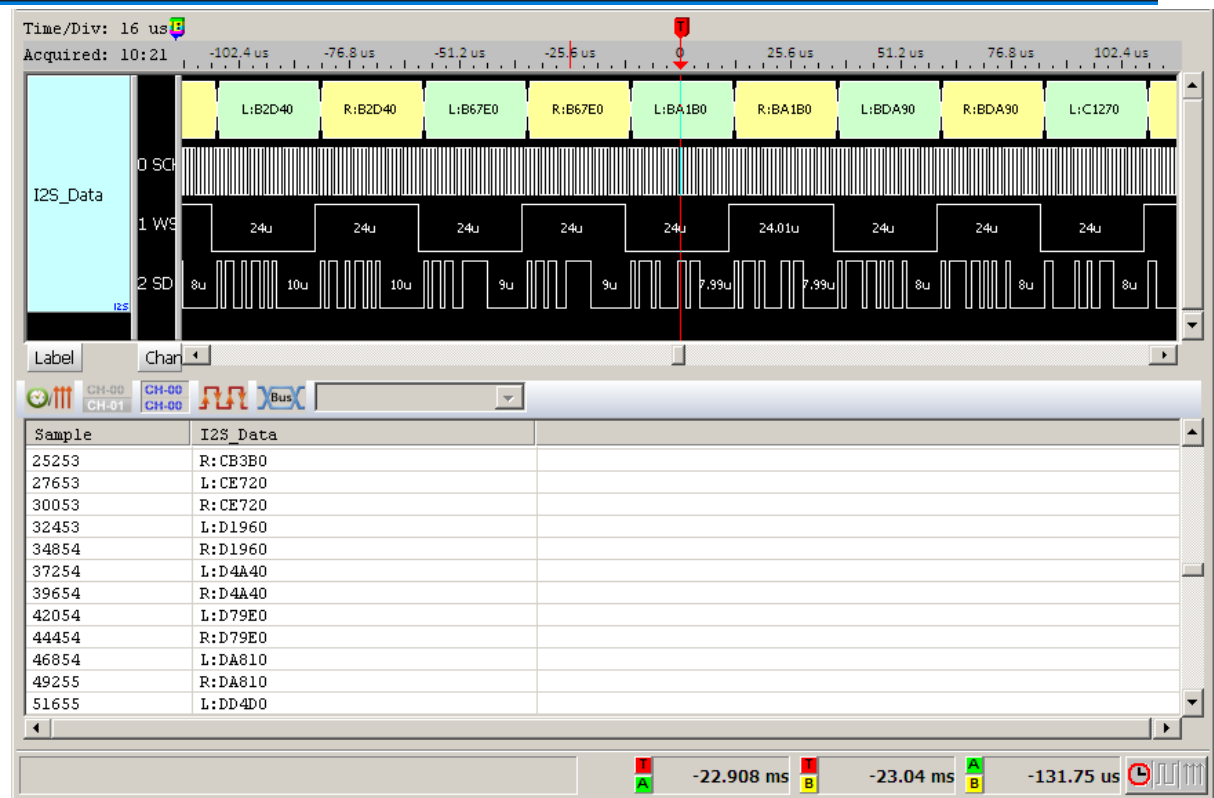
**Hold Time:** Trigger when the value is less than the Hold time.





## Result






## SPI Trigger

The settings dialog box below.

**If need higher Sample rate for SPI, choose SPI-800M which is 800MHz Sample Rate with 9 channels.** In SPI-800M, you can only use the default parameter settings.

Mode	Min. S/R	Max. S/R	Available ch.	Min. Mem.	Max. Mem.
+ UART Trigger	Baud Rate x 16	Baud Rate x 16	Adjustable	256	Adjustable
+ CAN Trigger	Data Rate x 10	Data Rate x 10	Adjustable	256	Adjustable
+ I2C Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ I2S Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ SPI Trigger(800M)-9	800MHz	800MHz	9	256	8M
- <b>SPI Trigger</b>	<b>1Hz</b>	<b>200MHz</b>	<b>Adjustable</b>	<b>256</b>	<b>Adjustable</b>
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Auto
SPI Trigger-36	1Hz	200MHz	Adjustable	256	2M
SPI Trigger-18	1Hz	200MHz	Adjustable	256	4M
SPI Trigger-12	1Hz	200MHz	Adjustable	256	6M
SPI Trigger-9	1Hz	200MHz	Adjustable	256	8M
SPI Trigger-6	1Hz	200MHz	Adjustable	256	12M
SPI Trigger-4	1Hz	200MHz	Adjustable	256	18M
+ SWID Trinner	1Hz	200MHz	Adjustable	256	Adjustable

## Trigger Setting

Select SPI Trigger-36 where "-36" means 36 channels and click OK. Then click the Trigger Settings button (  ) on Toolbar (or Device menu) to show the SPI Trigger Settings dialog box below.

**SPI Trigger Settings**

<b>Channel</b> CS: 0 SCK: 1 SDA: 2	<b>Chip Select</b> <input checked="" type="radio"/> Active Low <input type="radio"/> Active High <b>Clock Latch Data</b> <input checked="" type="radio"/> Rising Edge <input type="radio"/> Falling Edge	<b>Trigger Value</b> <input type="checkbox"/> IF <input type="checkbox"/> Then <input type="checkbox"/> After CS <input type="checkbox"/> After CS <input type="checkbox"/> OR IF <input type="checkbox"/> Then <input type="checkbox"/> After CS <input type="checkbox"/> After CS <input type="checkbox"/> OR IF <input type="checkbox"/> Then <input type="checkbox"/> After CS <input type="checkbox"/> After CS <input type="checkbox"/> OR IF <input type="checkbox"/> Then <input type="checkbox"/> After CS <input type="checkbox"/> After CS
<b>Word Size Setup</b> Word Size: 8 <input checked="" type="radio"/> MSB <input type="radio"/> LSB		
<input checked="" type="checkbox"/> Pre-Trigger   Pass Count: 0		
<div style="background-color: black; height: 100px; width: 100%;"></div>		
<div> <span>Check</span> <span>Load</span> <span>Save</span> <span>OK</span> <span>Cancel</span> </div>		

**Channel:** Select three channels for Chip Select (CH 0), serial clock (CH 1), and serial data (CH 2).

**Chip Select:** Select Active Low or Active High to enable Chip Select.

**Clock Latch Data:** Select Rising Edge or Falling Edge to latch data.

**Word Size Setup:** Select the number of data bits (4~24).

**MSB/LSB:** Select MSB first (MSB → LSB) or LSB first (LSB → MSB) to transmit the data.

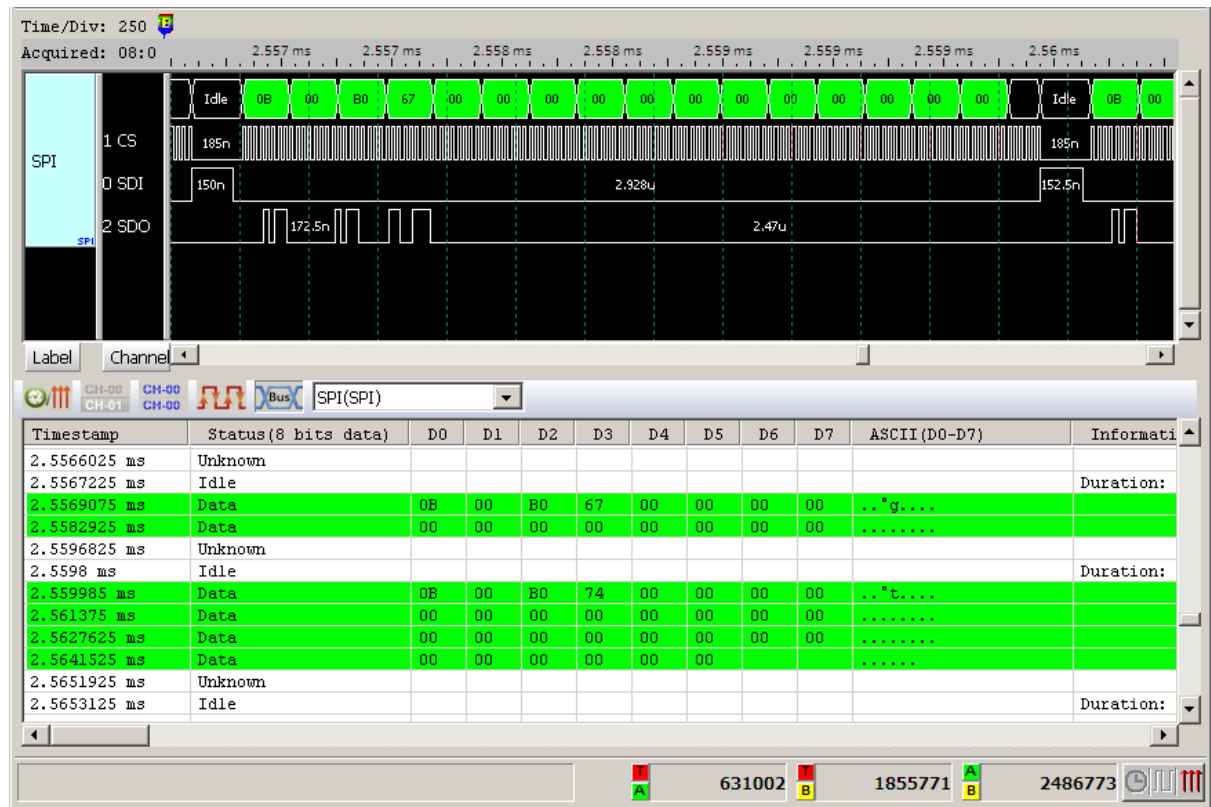
**Trigger Value:** The SPI Trigger has four IF/Then triggers, but ONLY one will trigger when its condition is met first. The condition can be in characters, a string, decimal numbers or hexadecimal codes with no more than 16 characters and: Characters and the string MUST be within single or double quotation marks, e.g. 'A' or "Acute"; A string can be entered with characters, a string and hexadecimal code with a blank space between each two different inputs, 'A'\_"cute" or 'A'\_63h\_'u'\_'t'\_65h e.g.

**Pass Count:** Pass Count will pass N times that triggers occurred. If the Pass Count is 5 as the dialog box below, all triggers in all IF/Then conditions are counted and the instrument will trigger at the 6th trigger.

Channel	CS	SCK	SDA
1	0	8clk	DDh
2	1	8clk	98h

**Check Trigger:** Check the validity of trigger conditions.

## Result




## SVID Trigger (Upon Request)

The settings dialog box below.

Mode	Min. S/R	Max. S/R	Available ch.	Min. Mem.	Max. Mem.
+ CAN Trigger	Data Rate x 10	Data Rate x 10	Adjustable	256	Adjustable
+ I2C Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ I2S Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ SPI Trigger(800M)-9	800MHz	800MHz	9	256	8M
+ SPI Trigger	1Hz	200MHz	Adjustable	256	Adjustable
- SVID Trigger	1Hz	200MHz	Adjustable	256	Adjustable
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Auto
SVID Trigger-36	1Hz	200MHz	Adjustable	256	2M
SVID Trigger-18	1Hz	200MHz	Adjustable	256	2M
SVID Trigger-12	1Hz	200MHz	Adjustable	256	2M
SVID Trigger-9	1Hz	200MHz	Adjustable	256	2M
SVID Trigger-6	1Hz	200MHz	Adjustable	256	2M
SVID Trigger-4	1Hz	200MHz	Adjustable	256	2M

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

### Trigger Settings

Select SVID Trigger-36 where "-36" means 36 channels and click OK. Then click the Trigger Settings button  on Toolbar (or Device menu) to show the SVID Trigger Settings dialog box below.

**SVID Trigger Settings**

Channel: SCLK  SDATA  ☐ ALERT  Trigger Method:

	Address	Command	Master Payload	When Alert	ACK	Slave Payload	Trigger
P1	<input type="text" value="XXh Don't care"/>	<input type="text" value="XXh Don't care"/>	<input type="text" value="XXh"/>	<input type="text" value="X Don't care"/>	<input type="text" value="X Don't care"/>	<input type="text" value="XXh"/>	<input type="text" value="Then Trigger"/>
P2	<input type="text" value="XXh Don't care"/>	<input type="text" value="XXh Don't care"/>	<input type="text" value="XXh"/>	<input type="text" value="X Don't care"/>	<input type="text" value="X Don't care"/>	<input type="text" value="XXh"/>	<input type="text" value="Then Trigger"/>
P3	<input type="text" value="XXh Don't care"/>	<input type="text" value="XXh Don't care"/>	<input type="text" value="XXh"/>	<input type="text" value="X Don't care"/>	<input type="text" value="X Don't care"/>	<input type="text" value="XXh"/>	<input type="text" value="Then Trigger"/>
P4	<input type="text" value="XXh Don't care"/>	<input type="text" value="XXh Don't care"/>	<input type="text" value="XXh"/>	<input type="text" value="X Don't care"/>	<input type="text" value="X Don't care"/>	<input type="text" value="XXh"/>	<input type="text" value="Then Trigger"/>

☒ Pre-Trigger Pass Count

**Channel:** Set the 2 channels (SCLK and SDATA) to form the trigger signal, ALERT

is option.

### Trigger Mode:

Start: Trigger when the SVID packet is valid.

Frame Data: Trigger according to the frame data.

Parity Error: Trigger when the parity error happens.

### Frame Data Settings:

	Address	Command	Master Payload	When Alert	ACK	Slave Payload	Trigger
P1	xxh Don't care	=	xxh Don't care	xxh	X Don't care	X Don't care	xxh
P2	xxh Don't care	=	xxh Don't care	xxh	X Don't care	X Don't care	xxh
P3	xxh Don't care	=	xxh Don't care	xxh	X Don't care	X Don't care	xxh
P4	xxh Don't care	=	xxh Don't care	xxh	X Don't care	X Don't care	xxh

☒ Pre-Trigger    Pass Count: 0    [Load] [Save] [Clear All] [OK] [Cancel]

You can set 4 (most) frame data packets as trigger conditions. Each column in those frame data can be set as any value or Don't care (xxh) and set trigger condition equal to the command or not.

The Payload column is 1 byte only, can be character (like 'A'), decimal number (like 65) or hexa-decimal number (like 41h), but any input value over 1 byte will be filtered.

The ALTER channel can be '0', '1' or don't care.

Trigger column includes:

**Then Trigger:** If conditions matched, trigger right away.

**Then If:** If conditions matched, wait for the next matched frame data.

**Next:** If conditions matched, wait for any next frame data.

**Pass Count:** The Pass Count function in the SVID Trigger is very unique. If you set the frame data conditions like P1 and P2 in the SVID Trigger Settings table below, the

pass count will only count 1 time when both P1 and P2 are matched. It will trigger after 2 matches (P1 and P2).

**SVID Trigger Settings**

Channel: SCLK 0 SDATA 1 ☐ ALERT 2 Trigger Method: Frame Data

	Address	Command	Master Payload	When Alert	ACK	Slave Payload	Trigger
P1	XXh Don't care	= XXh Don't care	XXh	X Don't care	X Don't care	XXh	Next
P2	XXh Don't care	= XXh Don't care	XXh	X Don't care	X Don't care	XXh	Then Trigger
P3	XXh Don't care	= XXh Don't care	XXh	X Don't care	X Don't care	XXh	Then Trigger
P4	XXh Don't care	= XXh Don't care	XXh	X Don't care	X Don't care	XXh	Then Trigger

☒ Pre-Trigger Pass Count 0

Load Save Clear All OK Cancel

If you set the frame data conditions like P1, P2 and P3 in the SVID Trigger Settings table below, the pass count will only count 1 time when both P1 and P2 are matched first and then P3 is matched. It will trigger after 2 matches (P3).

**SVID Trigger Settings**

Channel: SCLK 0 SDATA 1 ☐ ALERT 2 Trigger Method: Frame Data

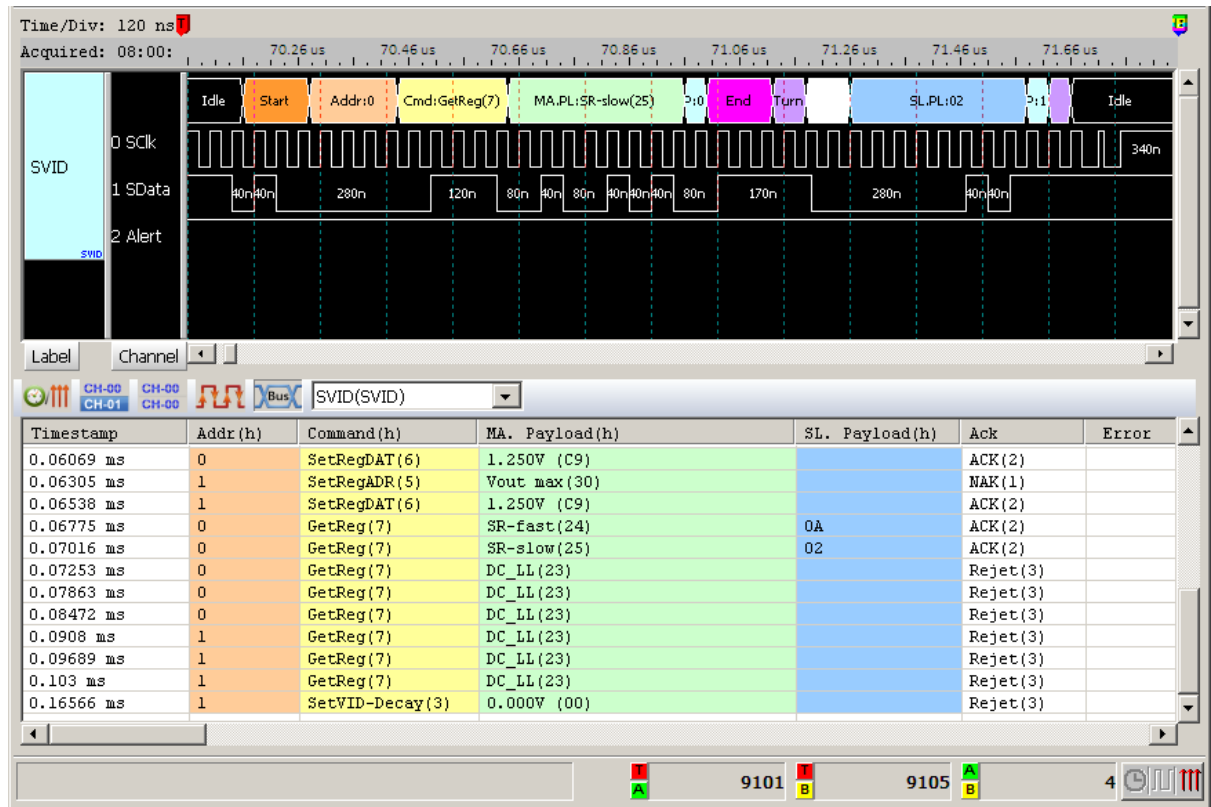
	Address	Command	Master Payload	When Alert	ACK	Slave Payload	Trigger
P1	XXh Don't care	= XXh Don't care	XXh	X Don't care	X Don't care	XXh	Next
P2	XXh Don't care	= XXh Don't care	XXh	X Don't care	X Don't care	XXh	Then If
P3	XXh Don't care	= XXh Don't care	XXh	X Don't care	X Don't care	XXh	Then Trigger
P4	XXh Don't care	= XXh Don't care	XXh	X Don't care	X Don't care	XXh	Then Trigger

☒ Pre-Trigger Pass Count 0

Load Save Clear All OK Cancel



## Result




## UART Trigger

The settings dialog box below.

Mode	Min. S/R	Max. S/R	Available ch.	Min. Mem.	Max
800M	800MHz	800MHz	9	256	8M
400M	400MHz	400MHz	18	256	4M
+ 200M	1Hz	200MHz	Adjustable	256	Adj
- UART Trigger	1Hz	200MHz	Adjustable	256	Adj
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Aut
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Aut
UART Trigger-36	1Hz	200MHz	Adjustable	256	2M
UART Trigger-18	1Hz	200MHz	Adjustable	256	4M
UART Trigger-12	1Hz	200MHz	Adjustable	256	6M
UART Trigger-9	1Hz	200MHz	Adjustable	256	8M
UART Trigger-6	1Hz	200MHz	Adjustable	256	12M
UART Trigger-4	1Hz	200MHz	Adjustable	256	18M
UART Trigger-2	1Hz	200MHz	Adjustable	256	36M
UART Trigger-1	1Hz	200MHz	Adjustable	256	72M
+ CAN Trigger	1Hz	200MHz	Adjustable	256	Adj

### Trigger Settings

Select UART Trigger-36 where "-36" means 36 channels and click OK. Then click the Trigger Settings button(  ) on Toolbar (or Device menu) to show the UART Trigger Settings dialog box below.

UART Trigger Settings

Trigger Channel: 0

Baud Rate: 9600

Data Bits: 8

Parity: NONE

Stop Bits: 1

☐ IF
 ☐ OR IF
 ☐ OR IF
 ☐ OR IF

☐ Pre-Trigger

Polarity

☒ Idle High
 

Start Bit D0.....D7 Parity Stop Bit

☐ Idle Low
 

Start Bit D0.....D7 Parity Stop Bit

☐ Then
 ☐ Then
 ☐ Then
 ☐ Then

Pass Count 0

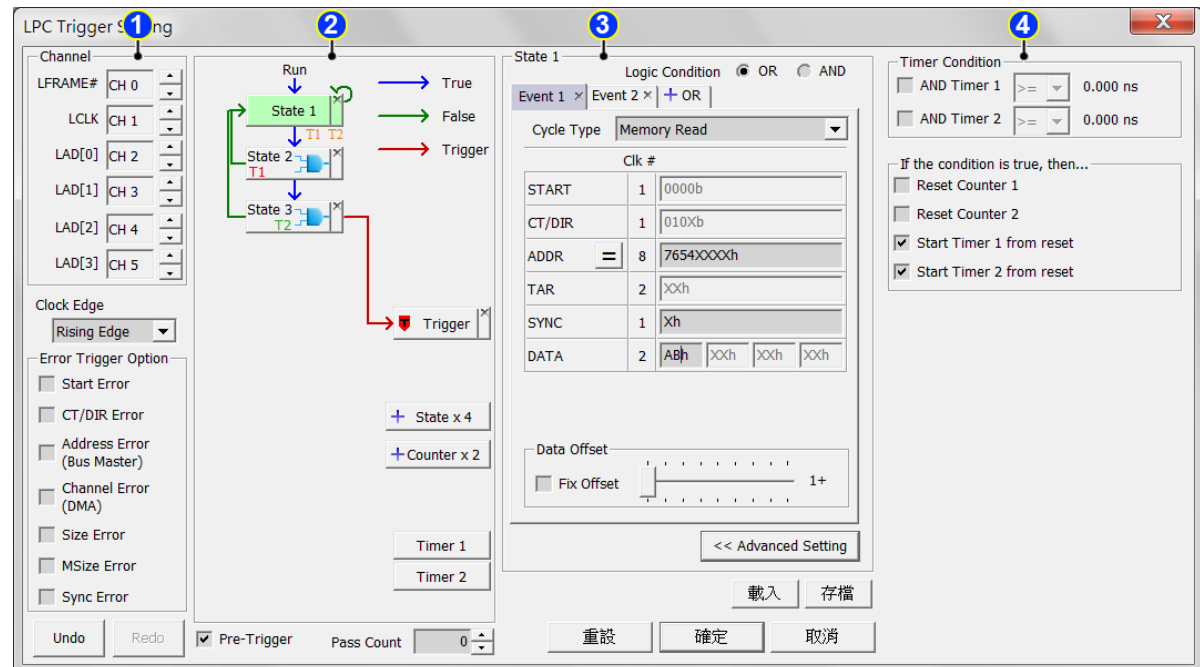
Check Load Save

OK Cancel



## Clause Trigger

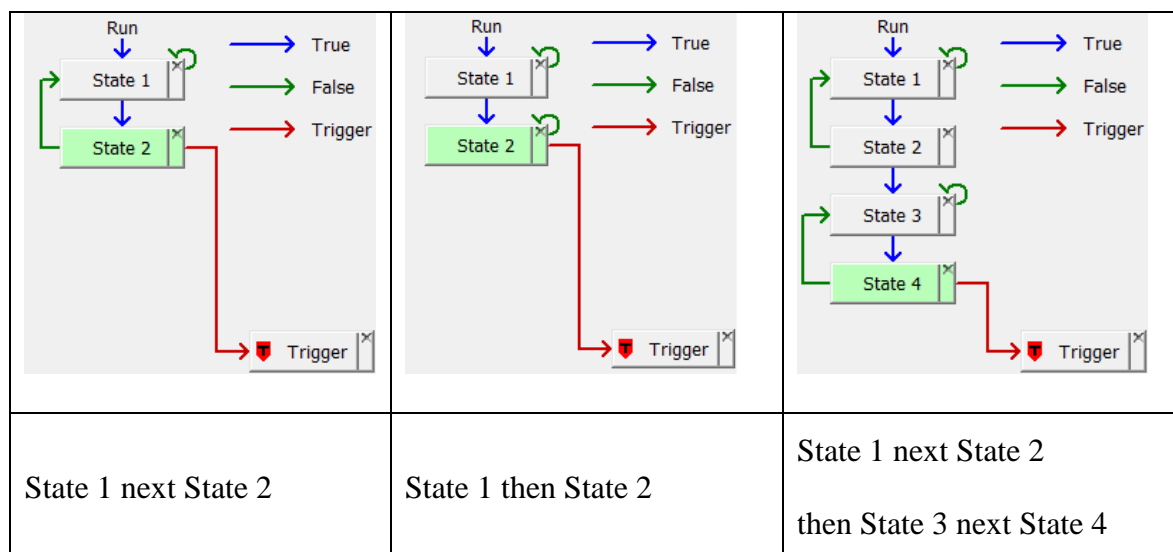
The Clause trigger settings dialog box is as below.

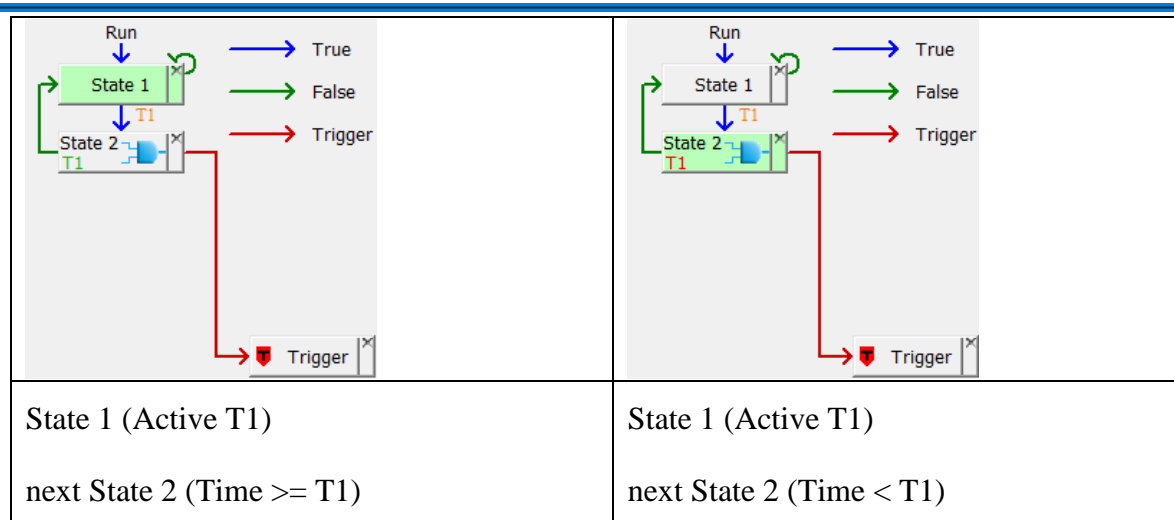


### 1. Channel and bus trigger settings:

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

### 2. Flow chart





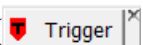
A state button represents a trigger state. The true state condition will only lead the branch to the next state, on the other hand, the false state condition could lead the branch to any state as below:


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

State 1

Logic Condition ☒ OR ☐ AND

Event 1 × Event 2 × + OR

Cycle Type Memory Read

	Clk #	
START	1	0000b
CT/DIR	1	010Xb
ADDR	8	7654XXXXh
TAR	2	XXh
SYNC	1	Xh
DATA	2	ABh XXh XXh XXh

Data Offset

☐ Fix Offset

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).
4. The bus trigger settings.

The relations between events and the trigger condition:

	AND	OR
Event settings		
Trigger condition		

#### 4. Timer and Counter settings

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

**Timer Condition**

☒ AND Timer 1 < 5.000 ns

☒ AND Timer 2 >= 5.000 ns

**If the condition is true, then...**

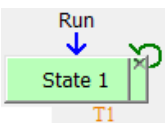
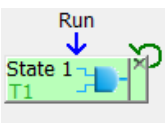
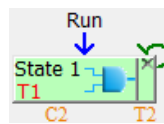
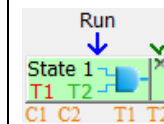
☒ Reset Counter 1

☒ Reset Counter 2

☒ Start Timer 1 from reset

☒ Start Timer 2 from reset

The following table describes the state button icons:

				
Conditions	State 1	State 1 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

## eSPI Trigger

Select eSPI Trigger in Hardware Settings.

Mode	Min. S/R	Max. S/R	Available ch.	Min. Mem.	Max. Mem.
+ 200M	1Hz	200MHz	Adjustable	256	Adjustable
+ CAN Trigger	1Hz	200MHz	Adjustable	256	Adjustable
- eSPI Trigger	1Hz	200MHz	Adjustable	256	Adjustable
Transitional Storage(400M)-16	400MHz	400MHz	Fixed	Auto	Auto
eSPI Trigger(400M)-18	400MHz	400MHz	18	256	4M
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Auto
eSPI Trigger-36	1Hz	200MHz	Adjustable	256	2M
eSPI Trigger-18	1Hz	200MHz	Adjustable	256	4M
eSPI Trigger-12	1Hz	200MHz	Adjustable	256	6M
eSPI Trigger-9	1Hz	200MHz	Adjustable	256	8M
eSPI Trigger-6	1Hz	200MHz	Adjustable	256	12M
+ I2C Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ I2S Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ LIN Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ LPC Trigger	1Hz	200MHz	Adjustable	256	Adjustable

## Trigger Settings

Select “eSPI” in Trigger Settings.

Enhanced SPI (eSPI) Trigger Settings

Channel

CS# CH 0

SCLK CH 1

I/O 0 CH 2

I/O 1 CH 3

I/O 2 CH 4

I/O 3 CH 5

Alert CH 6

Start Up Settings

Single Mode

Alert From I/O[1]

CRC Check Enable

Trigger On

☐ Format Error

☐ OPCode Error

☐ Response Error

☐ Status Error

☐ CRC Error

tCLQV 15ns

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Run

State 1

True

False

Trigger

Timer 1

Timer 2

+ State x 7

+ Counter x 2

State 1

Logic Condition ☒ OR ☐ AND

Event 1 +OR

Command Any Command (00h)

Response Any Response (00h)

Data Dir ☒ Command ☐ Response

Data Offset ☒ AnyOfs 0+

Command	Response
OPCode	Response
xxh	xxh

Advanced Setting >>

Undo Redo

☒ Pre-Trigger

Pass Count 0

Load Save Default OK Cancel



**Channel:** Please refer to the Bus Decode of eSPI.

**Start Up Settings:** There are three modes including single, dual and quad mode. Set the current mode to be any of them and the setting would be automatically switched by monitoring the bus in hardware. Then choose the Alert signal and the CRC Check.

**Trigger On:** Select any error of format, OP Code, Response, Status and CRC to trigger.

**tCLQV:** The delay time of I/O and Clock in Response. The incorrect setting may cause that Response cannot be triggered.

### Trigger State Settings:

State 1

Logic Condition ☒ OR ☐ AND

Event 1 +OR |

Command GET\_CONFIGURATION (21h)

Response ACCEPT (x4 Data) (08h)

Data Dir ☐ Command ☒ Response

Data Offset ☒ AnyOfs 0+

Command	Response
Address	Data (D0+)
XXh	13h
Address	Data (D1+)
XXh	11h
CRC	Data (D2+)
XXh	XXh
	Status (D4+)
	XXh
	Status
	XXh
	Status
	XXh

Advanced Setting >>

eSPI Trigger offers many kinds of Command and Response. You may set “Any Command” to capture the signal, and choose the proper setting of trigger by the decode of eSPI.

Timestamp	OpCode/Response	CycType	Tag	LEN	Address	D0	D1	D2	D3	D4	D5	D6	D7	ASCII	Status	CRC	Memo
0.00000245 S	GET_CONFIGURATION(21)				0010											58	
0.00000086 S	ACCEPI(08)					13	11	00	00					....	030F	95	
0.000003 S	SET_CONFIGURATION(22)				0010	01	11	00	00					....		F5	
0.000005935 S	ACCEPI(08)															030F	9B
0.000008455 S	GET_STATUS(25)															FB	
0.000009365 S	ACCEPI(08)															030F	9B
0.001601195 S	GET_CONFIGURATION(21)				0010												58
0.001602795 S	ACCEPI(08)					13	11	00	00					....	030F	95	
0.001606635 S	SET_CONFIGURATION(22)				0010	01	11	00	00					....		F5	
0.001609575 S	ACCEPI(08)															030F	9B

eSPI decode result

**Data Dir:** Trigger the data in the Command or Response.

**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.

## LIN Trigger

Select LIN Trigger Settings.

Mode	Min. S/R	Max. S/R	Available ch.	Min. Mem.	Max
+ LPC Trigger	1Hz	200MHz	Adjustable	256	Adj
+ NAND Flash Trigger	1Hz	200MHz	Adjustable	256	Adj
+ SWI2 Trigger	1Hz	200MHz	Adjustable	256	Adj
+ SMBus/PMBus Trigger	1Hz	200MHz	Adjustable	256	Adj
+ USB 1.1 Trigger	1Hz	200MHz	Adjustable	256	Adj
- LIN Trigger	1Hz	200MHz	Adjustable	256	Adj
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Aut
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Aut
LIN Trigger-36	1Hz	200MHz	Adjustable	256	2M
LIN Trigger-18	1Hz	200MHz	Adjustable	256	4M
LIN Trigger-12	1Hz	200MHz	Adjustable	256	6M
LIN Trigger-9	1Hz	200MHz	Adjustable	256	8M
LIN Trigger-6	1Hz	200MHz	Adjustable	256	12M
LIN Trigger-4	1Hz	200MHz	Adjustable	256	18M
+ External Clock	1Hz	200MHz	Adjustable	256	Adj

## Trigger Settings

The LIN trigger settings dialog box is as below.

**LIN Trigger Settings** [?] [X]

Channel: LIN CH 0

Baud rate: 9600

Trigger frame:

- ☐ Break
- ☐ Sync
- ☐ Data
- ☐ End
- ☐ Wake up

Error Detect:

- ☐ Sync Error
- ☐ Parity Error
- ☐ Stop Error
- ☐ Checksum Error

Classic mode

Run State 1

Logic Condition: ☒ OR ☐ AND

Event 1: +OR

ID: xxh Parity: xh

D0: xxh D4: xxh

D1: xxh D5: xxh

D2: xxh D6: xxh

D3: xxh D7: xxh

Advanced Setting >>

Undo Redo ☒ Pre-Trigger Pass Count: 0 Load Save Default OK Cancel

**Channel:** Select channels.

**Baud rate:** Select baud rate.

**Trigger frame:** Provide Break / Sync / Data / End / Wake up triggers.

**Error Detect:** Provide Sync / Parity / Stop / Checksum error triggers, checksum error detect include “Classic” and “Enhanced” mode.

**Redo / Undo:** Click these buttons to redo/restore previous setting.

**Trigger settings:**

ID	<input type="text" value="xxh"/>	Parity	<input type="text" value="xh"/>
D0	<input type="text" value="xxh"/>	D4	<input type="text" value="xxh"/>
D1	<input type="text" value="xxh"/>	D5	<input type="text" value="xxh"/>
D2	<input type="text" value="xxh"/>	D6	<input type="text" value="xxh"/>
D3	<input type="text" value="xxh"/>	D7	<input type="text" value="xxh"/>

Provide the ID / Parity / Data triggers °

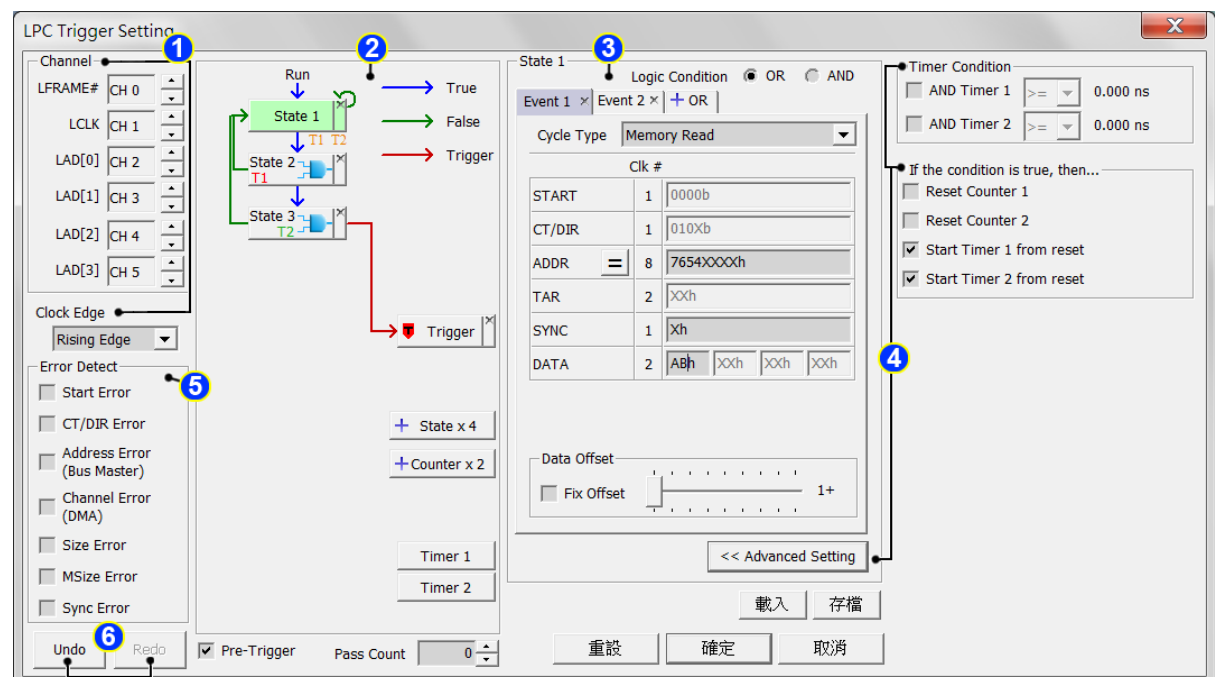
## LPC Trigger

The LPC trigger settings dialog box is as below.

Mode	Min. S/R	Max. S/R	Available...	Min. M...	Max. Mem.
SPI Trigger(800M)-9	800MHz	800MHz	9	256	8M
+ SPI Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ SVID Trigger	1Hz	200MHz	Adjustable	256	Adjustable
Serial Flash Trigger(800M)-9	800MHz	800MHz	9	256	8M
+ SD/eMMC Trigger	1Hz	200MHz	Adjustable	256	Adjustable
- LPC Trigger	1Hz	200MHz	Adjustable	256	Adjustable
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Auto
LPC Trigger-36	1Hz	200MHz	Adjustable	256	2M
LPC Trigger-18	1Hz	200MHz	Adjustable	256	4M
LPC Trigger-12	1Hz	200MHz	Adjustable	256	6M
LPC Trigger-9	1Hz	200MHz	Adjustable	256	8M
LPC Trigger-6	1Hz	200MHz	Adjustable	256	12M
+ NAND Flash Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ SVID Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ USB 1.1 Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ External Clock	1Hz	200MHz	Adjustable	256	Adjustable

## Trigger Settings

Select LPC Trigger Settings.



### 1. Channel

Select channels

Clock Edge  
Rising Edge ▼

: Latch data when Clock Rising or Falling Edge

## 2. Clause trigger settings

Please refer to the Clause Trigger

## 3. Trigger settings

Cycle Type I/O Read ▼

	Clk #	
START	1	0000b
CT/DIR	1	000Xb
ADDR	4	0068h
TAR	2	XXh
SYNC	1	Xh
DATA	2	00h XXh XXh XXh

Data Offset  
☐ Fix Offset  1+

Cycle Type I/O Read ▼

- Start of Frame
- I/O Read
- I/O Write
- Memory Read
- Memory Write
- DMA Read
- DMA Write
- Firmware Memory Read
- Firmware Memory Write
- Bus Master 0 - I/O Read
- Bus Master 0 - I/O Write
- Bus Master 0 - Memory Read
- Bus Master 0 - Memory Write
- Bus Master 1 - I/O Read
- Bus Master 1 - I/O Write
- Bus Master 1 - Memory Read
- Bus Master 1 - Memory Write

The LPC trigger supports many cycle types. Please select “Start of Frame” and capture, then choose the cycle type you need.

Sample	Field	#Clocks	LAD	Comment
7002909	START	1	0	Used for Memory or I/O or DMA cycles.
7002917	CYCLETYPEDIR	1	0	I/O Read
7002925	ADDR	4	0068	
7002959	TAR	2	FF	
7002975	SYNC	1	6	Long Wait
7002984	SYNC	1	6	Long Wait
7002992	SYNC	1	6	Long Wait
7003000	SYNC	1	0	Ready
7003009	DATA	2	00	
7003025	TAR	2	FF	
7082283	START	1	0	Used for Memory or I/O or DMA cycles.
7082291	CYCLETYPEDIR	1	0	I/O Read
7082299	ADDR	4	0064	

LPC decoder result

Other settings :

Switch ☐ ☐ ☐ ☐ by click ☐ buttonmn.

The input ‘X’ means don’t care.



: Data offset setting

4. Timer and Counter

Please refer to the Clause Trigger

5. Trigger when an error is detected based on the Intel® Low Pin Count

Interface Specification.

6. Redo/Undo

## MIPI SPMI Trigger

Select eSPI Trigger in Hardware Settings.

Mode	Min. S/R	Max. S/R	Available ch.	Min. Mem.	Max. ^
+ LIN Trigger	1Hz	200MHz	Adjustable	256	Adju
+ LPC Trigger	1Hz	200MHz	Adjustable	256	Adju
- MIPI SPMI Trigger	1Hz	200MHz	Adjustable	256	Adju
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Auto
MIPI SPMI Trigger-36	1Hz	200MHz	Adjustable	256	2M
MIPI SPMI Trigger-18	1Hz	200MHz	Adjustable	256	4M
MIPI SPMI Trigger-12	1Hz	200MHz	Adjustable	256	6M
MIPI SPMI Trigger-9	1Hz	200MHz	Adjustable	256	8M
MIPI SPMI Trigger-6	1Hz	200MHz	Adjustable	256	12M
MIPI SPMI Trigger-4	1Hz	200MHz	Adjustable	256	18M
+ NAND Flash Trigger	1Hz	200MHz	Adjustable	256	Adju
+ SD/eMMC Trigger	1Hz	200MHz	Adjustable	256	Adju
Serial Flash Trigger(800M)-9	800MHz	800MHz	9	256	8M
+ SMBus/PMBus Trigger	1Hz	200MHz	Adjustable	256	Adju

## Trigger Settings

Select “MIPI SPMI” in Trigger Settings.

MIPI SPMI Trigger Settings

Channel

SPMI CLK CH 0

SPMI DAT CH 1

☐ AUX CH 2

Trigger On

☐ SSC

☐ Bus Timeout

☐ No Response Frame

Error Check

☐ CMD Frame Format

☐ CMD Parity

☐ Data Address Parity

☐ Data Frame Parity

☐ Bus Park / Bus Handover

Bus Timeout

96 us

Run

State 1

True

False

Trigger

+ State x 7

+ Counter x 2

Timer 1

Timer 2

State 1

Logic Condition ☒ OR ☐ AND

Event 1 +OR

Device Address Xh

Command XXh

Data Address XXXXh

Data

XXh XXh XXh XXh

XXh XXh XXh XXh

XXh XXh XXh XXh

XXh XXh XXh XXh

AUX Xh

Advanced Setting >>

Undo Redo ☒ Pre-Trigger

Pass Count 0

Load Save Default OK Cancel



**Trigger On:** Select Sequence Start Condition (SSC), Bus Timeout, or No Response Frame to trigger.

**Error Check:** Check the error of Frame Format, Parity, and Bus Park / Bus Handover.

**Bus Timeout:** Set the period of time to wait for SSC.

**State Settings:**

Device Address	5h		
Command	38h		
Data Address	XXXXh		
Data			
A3h	XXh	XXh	XXh
XXh	XXh	XXh	XXh
XXh	XXh	XXh	XXh
XXh	XXh	XXh	XXh
AUX			
Xh			

MIPI SPMI Trigger offer Address, Command and Data parameter to trigger. You may capture the data with don't care (X). Trigger the specific address, command or data by the decode of MIPI SPMI.

Timestamp	A	SR	Device Address (Hex)	Command (Hex)	Data Address - High (Hex)	Data Address - Low (Hex)	Data Frame
-0.838595 ms	MPL3	SA=00	38	(Extended Register Read Long: 1Bytes)	5C	46	00
-0.82933 ms	MPL3	SA=00	30	(Extended Register Write Long: 1Bytes)	5C	46	80
-0.02058 ms	MPL3	SA=05	38	(Extended Register Read Long: 1Bytes)	1D	40	00
-0.011155 ms	MPL3	SA=05	30	(Extended Register Write Long: 1Bytes)	1D	40	00
-0.00293 ms	MPL3	SA=05	38	(Extended Register Read Long: 1Bytes)	1D	41	A3
0.006135 ms	MPL3	SA=05	30	(Extended Register Write Long: 1Bytes)	1D	41	D2
3.5403 ms	MPL3	SA=00	38	(Extended Register Read Long: 1Bytes)	56	46	00
3.549725 ms	MPL3	SA=00	30	(Extended Register Write Long: 1Bytes)	56	46	80
4.058525 ms	MPL3	SA=00	38	(Extended Register Read Long: 1Bytes)	57	46	00
4.06811 ms	MPL3	SA=00	30	(Extended Register Write Long: 1Bytes)	57	46	80
4.343265 ms	MPL3	SA=00	38	(Extended Register Read Long: 1Bytes)	5B	46	00
4.352585 ms	MPL3	SA=00	30	(Extended Register Write Long: 1Bytes)	5B	46	80
4.618265 ms	MPL3	SA=00	38	(Extended Register Read Long: 1Bytes)	5C	46	00

MIPI SPMI decode result

## NAND Flash Trigger

The settings dialog box below.

Mode	Min. S/R	Max. S/R	Available...	Min. M...	Max. Mem.
+ CAN Trigger	Data Rate...	Data Rate...	Adjustable	256	Adjustable
+ I2C Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ I2S Trigger	1Hz	200MHz	Adjustable	256	Adjustable
[-] SPI Trigger(800M)-9	800MHz	800MHz	9	256	8M
+ SPI Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ SVID Trigger	1Hz	200MHz	Adjustable	256	Adjustable
[-] Serial Flash Trigger(800M)-9	800MHz	800MHz	9	256	8M
+ SD/eMMC Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ LPC Trigger	1Hz	200MHz	Adjustable	256	Adjustable
- NAND Flash Trigger	1Hz	200MHz	Adjustable	256	Adjustable
[-] NAND Flash Trigger(400M)-18	400MHz	400MHz	18	256	4M
[-] Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto
[-] NAND Flash Trigger-36	1Hz	200MHz	Adjustable	256	2M
[-] NAND Flash Trigger-18	1Hz	200MHz	Adjustable	256	4M
+ SVI2 Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ USB 1.1 Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ External Clock	1Hz	200MHz	Adjustable	256	Adjustable

## Trigger Settings

Select NAND Flash trigger settings dialog box is as below.

**NAND Flash Trigger Settings**

Channel: ☒ x 8 ☐ x 16

☒ I/O Quick Setup ☐ I/O User Defined

I/O (LSB): CH 0

I/O [7:0]

CLE: CH 8

ALE: CH 9

RE: CH 10

WE: CH 11

CE: CH 12

R/B: CH 13

☐ DQS

The Flash Startup mode: ☐ Toggle / ONFI DDR Mode

tREA >= 5.0ns

tDQSQ >= 5.0ns

☒ Cmds. accepted during busy

☐ Busy time check

☒ Pre-Trigger

Pass Count: 0

Load Save Default OK Cancel

State 1

Logic Condition: ☒ OR ☐ AND

Event 1: +OR

Command: xxh

Address: 3-Byte Row Address 4-Byte Row Address

Row: xxxxxxh

Column / Feature: ☒ xxxxxh

Data: Data Offset

xxh xxh

xxh xxh

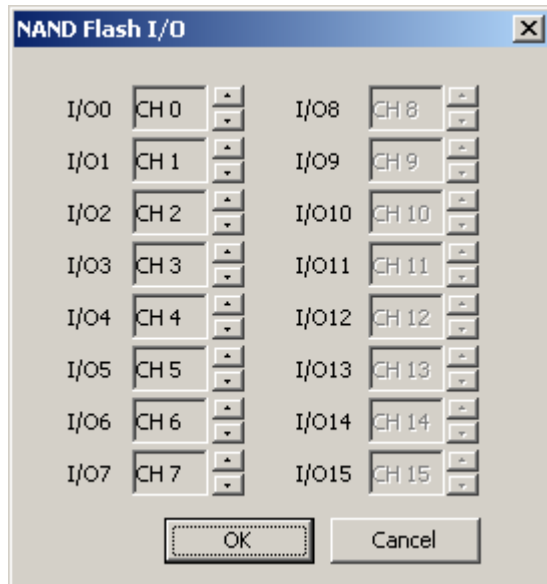
Advanced Setting >>

## Channel:

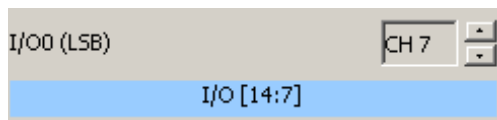
Select x8 or x16 to trigger 8 / 16 NAND Flash. Just set the I/O0 (LSB) channel

when check the I/O Quick Setup; pressing the button

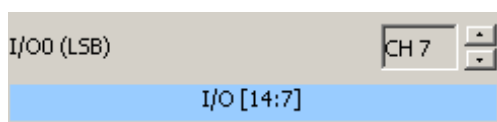
when check the I/O User Defined will show the dialog below:



## I/O Quick Setup:



Set LSB = CH 0, MSB = CH 7



Set LSB = CH 7, MSB = CH 14

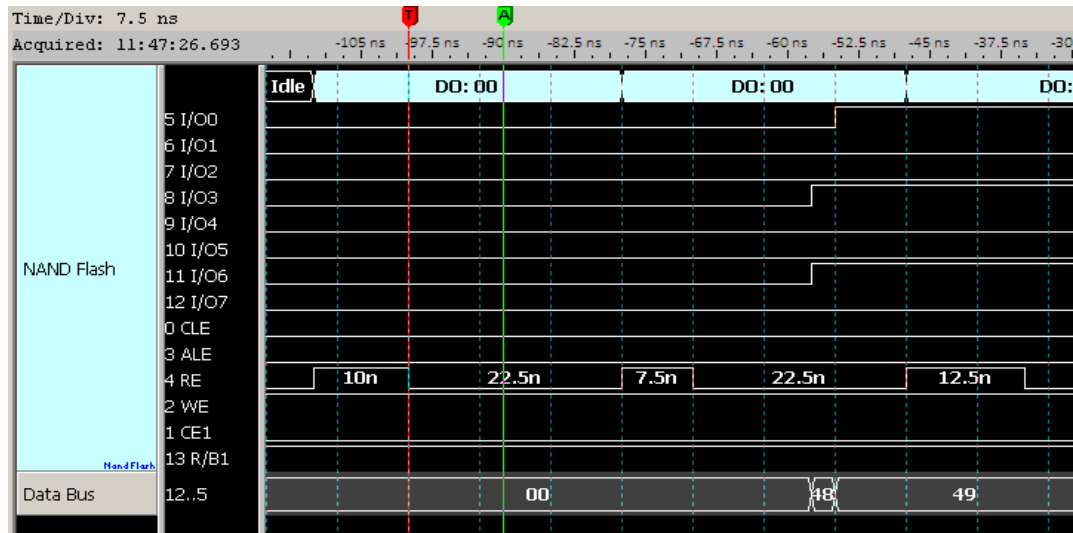
**DQS:** Check DQS pin to select the DDR (Double Data Rate) mode; SDR (Single Data Rate) mode default.

**The Flash Startup mode:** Check DQS pin and Toggle / ONFI DDR Mode when trigger NAND Command/Address/Data under NAND DDR mode.

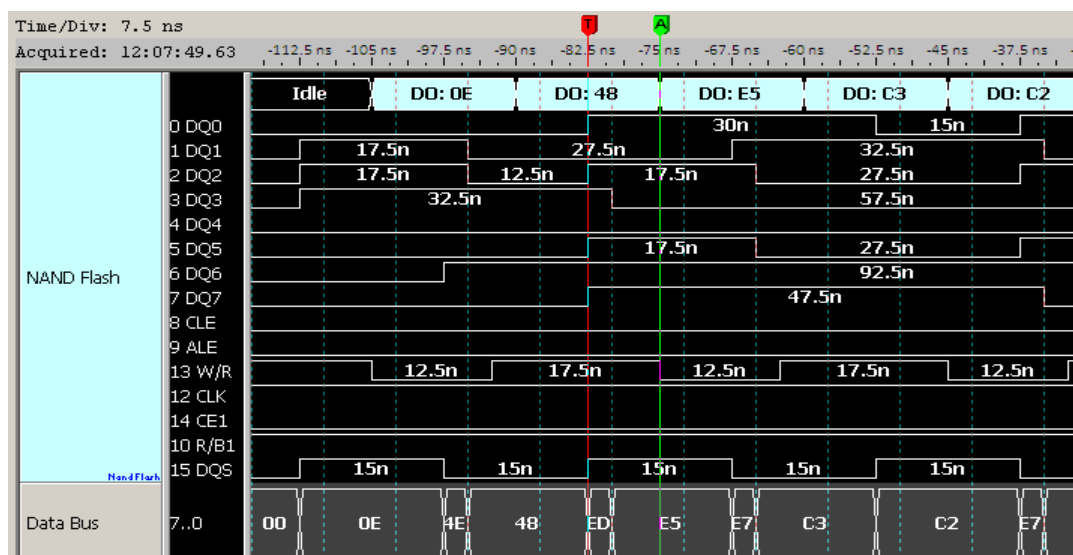
## tREA / tDQSQ:



NAND will access data from the edge delay a period. This period call tREA under SDR mode and tDQSQ under DDR mode. The unit is 5 ns under 200 MHz sampling rate and 2.5 ns under 400MHz sampling rate of LA. The period between cursor T and cursor A is tREA.



The period between cursor T and cursor A is tDQSQ.



**Commands accepted during busy / Busy time check:**

☒ Cmds. accepted during busy ...

☐ Busy time check ...

The Command accepted during busy checked default, pressing  will show the

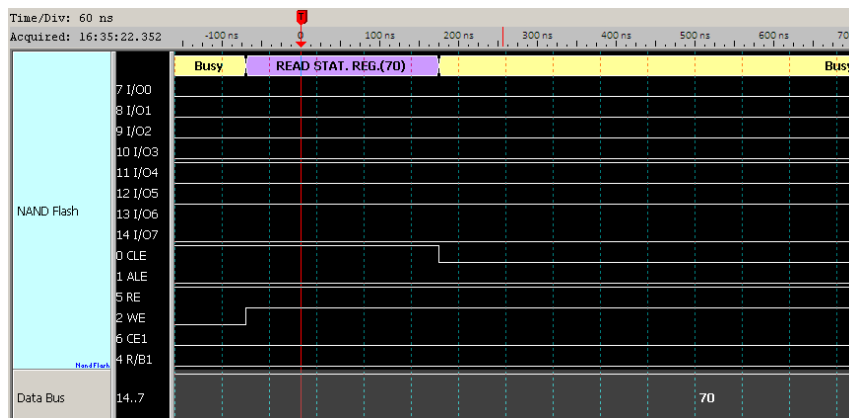
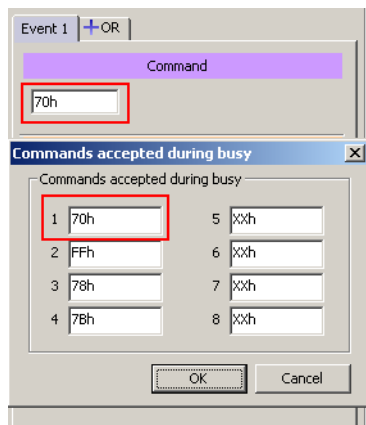
dialog below:




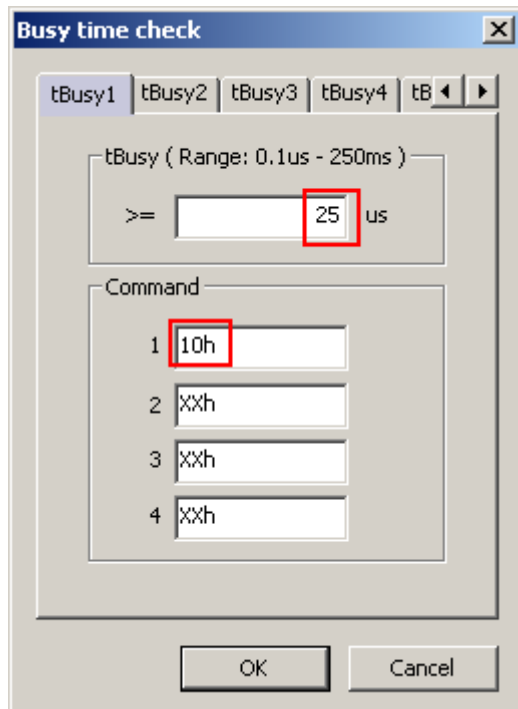
It means NAND commands still can be triggered while checking the Busy time check.

NAND command (70h/FFh/78h/7Bh) filled default.

### Trigger Command 70h during busy time

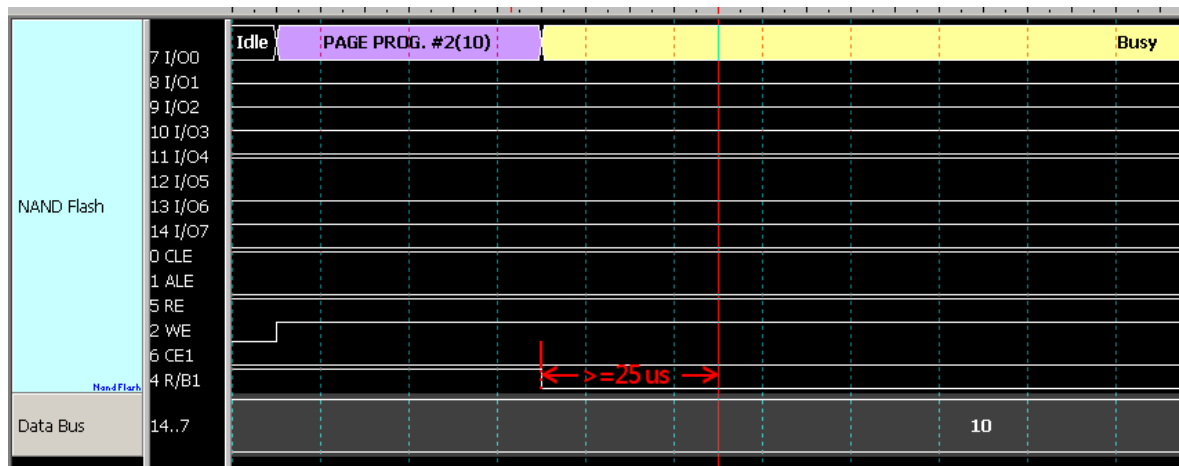


Check the Busy time check function and pressing  will show the dialog below:



The Busy time check function provide 6 NAND Flash busy time to check and every busy time can be used by 4 commands. It will trigger when the busy time is more than or equal to the tBusy users filled.

**Trigger the busy time  $\geq 25$  us after NAND command 10h**



**State / Event: Trigger NAND Command / Address / Data**

PAGE PROGRAM #1(80)	0026B1	0000	7B	9D	ED	8A	C3	E7	00	30
	0026B1	0008	26	A0	71	CD	BC	57	EA	25
	0026B1	0010	61	66	31	77	58	AC	39	56
	0026B1	0018	07	BE	9B	63	74	36	C5	B8
	0026B1	0020	4D	C5	68	F0	3B	84	58	14

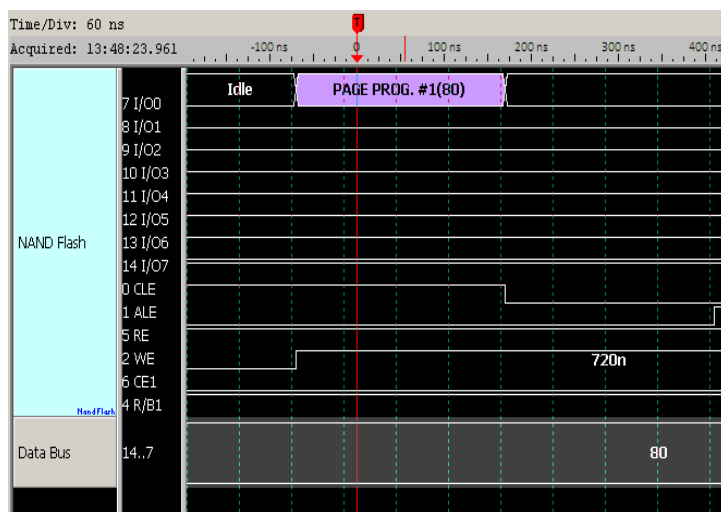
**Trigger Command 80h**

Event 1 **+OR**

**Command**  
80h

**Address**  
☒ 3-Byte Row Address ☐ 4-Byte Row Address  
 Row: xxxxxxh  
 Column / Feature: ☒ xxxxh

**Data**  
 Data Offset: [Slider]  
 [xxh] [xxh]  
 [xxh] [xxh]



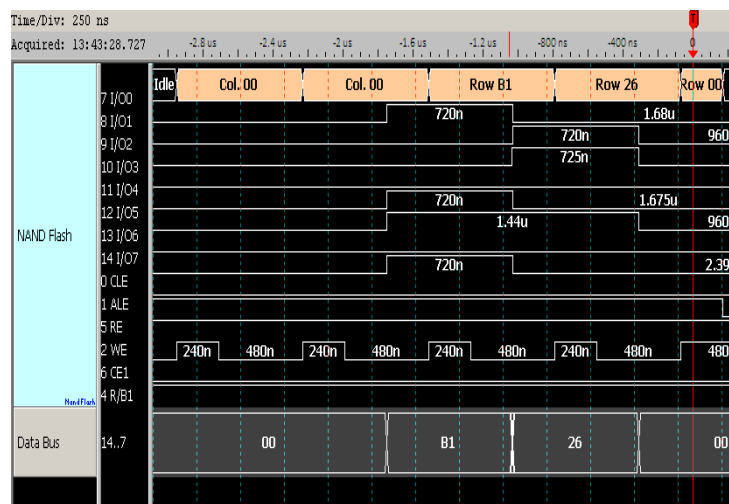
**Trigger Row Address: 0026B1h, Column Address: 0000h**

Event 1 **+OR**

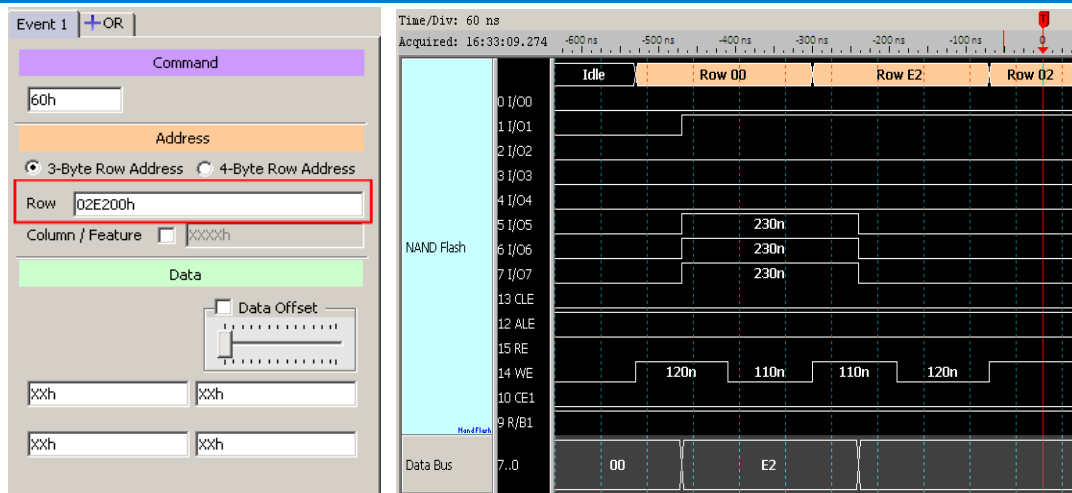
**Command**  
80h

**Address**  
☒ 3-Byte Row Address ☐ 4-Byte Row Address  
 Row: 0026B1h  
 Column / Feature: ☒ 0000h

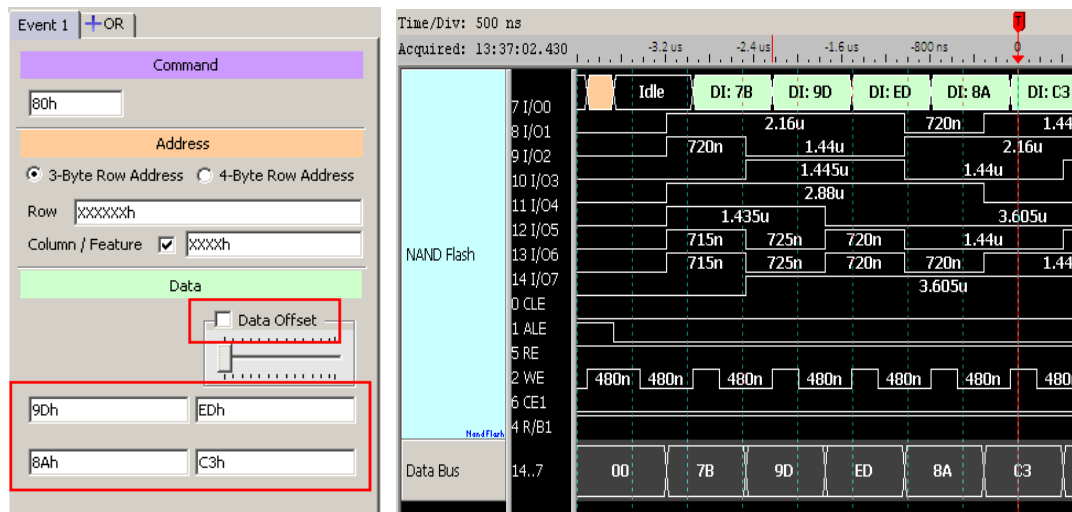
**Data**  
 Data Offset: [Slider]  
 [xxh] [xxh]  
 [xxh] [xxh]



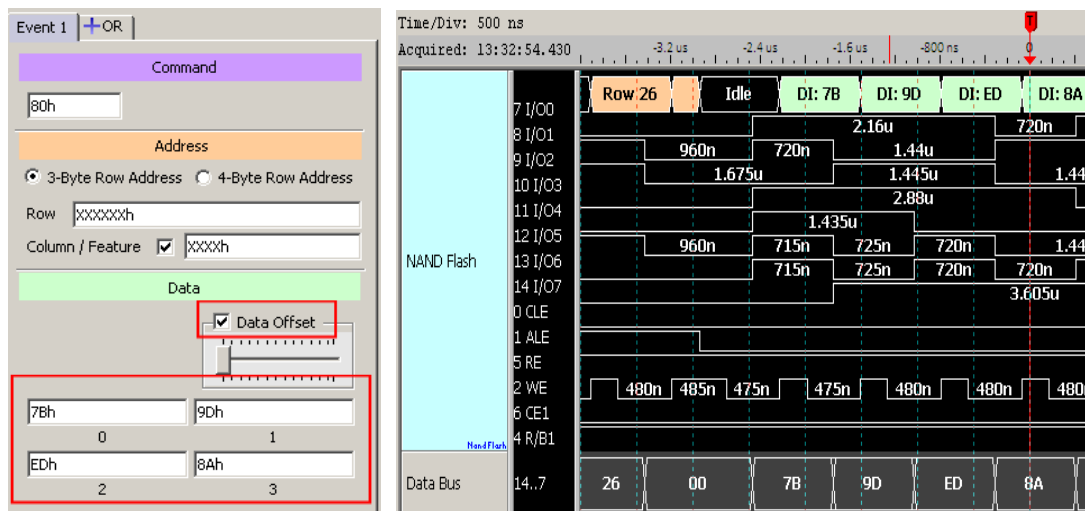
**Trigger Row Address: 02E200h, no Column Address**



## Trigger the NAND Data



## Trigger the NAND Data (fixed offset)





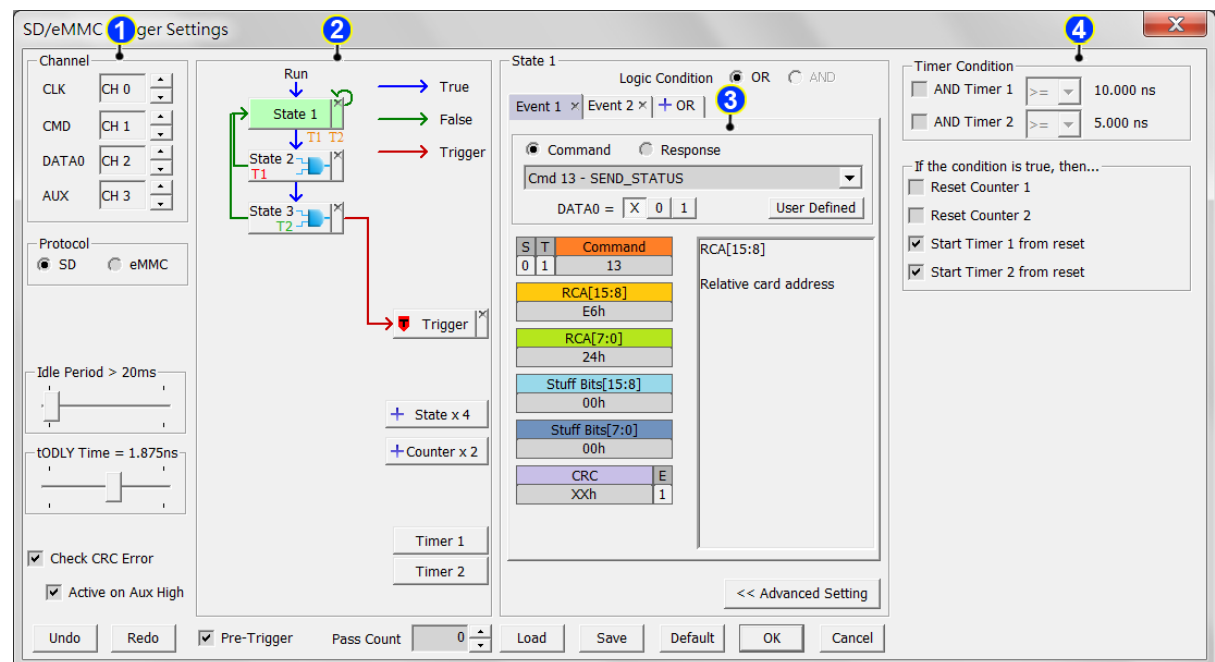
## SD/eMMC Trigger

The settings dialog box below.

Mode	Min. S/R	Max. S/R	Available...	Min. M...	Max. Mem.
+ I2S Trigger	1Hz	200MHz	Adjustable	256	Adjustable
[-] SPI Trigger(800M)-9	800MHz	800MHz	9	256	8M
+ SPI Trigger	1Hz	200MHz	Adjustable	256	Adjustable
+ SVID Trigger	1Hz	200MHz	Adjustable	256	Adjustable
[-] Serial Flash Trigger(800M)-9	800MHz	800MHz	9	256	8M
- SD/eMMC Trigger	1Hz	200MHz	Adjustable	256	Adjustable
[-] SD/eMMC Trigger(1.6G)-4	1.6GHz	1.6GHz	4	256	16M
[-] SD/eMMC Trigger(800M)-9	800MHz	800MHz	9	256	8M
[-] SD/eMMC Trigger(400M)-18	400MHz	400MHz	18	256	4M
[-] SD/eMMC Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto
[-] SD/eMMC Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Auto
[-] SD/eMMC Trigger-36	1Hz	200MHz	Adjustable	256	2M
[-] SD/eMMC Trigger-18	1Hz	200MHz	Adjustable	256	4M
[-] SD/eMMC Trigger-12	1Hz	200MHz	Adjustable	256	6M
[-] SD/eMMC Trigger-9	1Hz	200MHz	Adjustable	256	8M
[-] SD/eMMC Trigger-6	1Hz	200MHz	Adjustable	256	12M
[-] SD/eMMC Trigger-4	1Hz	200MHz	Adjustable	256	18M
+ LPC Trigger	1Hz	200MHz	Adjustable	256	Adjustable

## Trigger Settings

Select SD/eMMC Trigger Settings dialog box below.



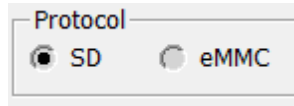
### 1. Channel

CLK: Clock signal.

CMD: Command signal.

Data0: eMMC only and judge response is R1 or R1b.

AUX: auxiliary CRC check.



Protocol  
☒ SD    ☐ eMMC

select SD or eMMC trigger.

Idle Period: If clock keep idle and over the idle period value, then reset the state.

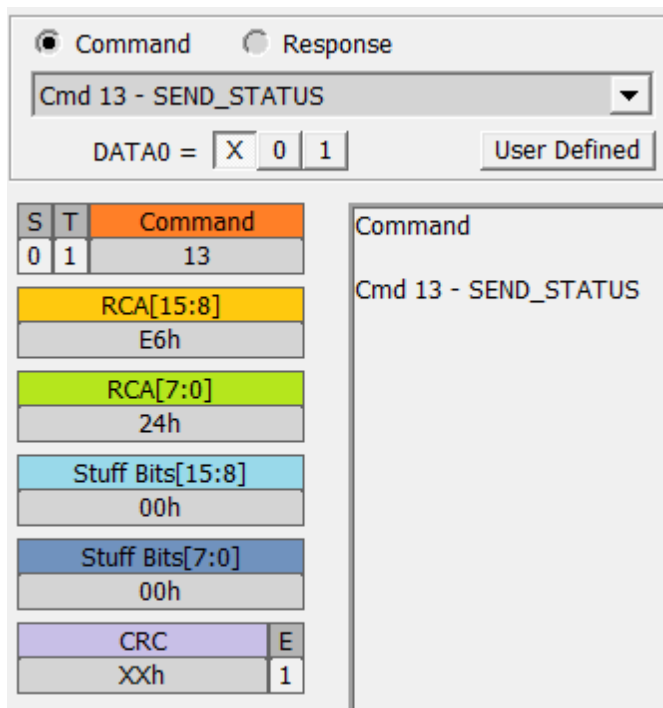
tODLY (Output delay) Time: When response, the delay time after clock edge is the valid command data.

Check CRC Error: Trigger the CRC error, if “Active on AUX High” is checked, the CRC check only when the AUX channel keeps high.

## 2. Clause trigger settings

Please reference Clause Trigger chapter

## 3. Trigger settings



☒ Command    ☐ Response

Cmd 13 - SEND\_STATUS

DATA0 = ☒ X    ☐ 0    ☐ 1    User Defined

S	T	Command
0	1	13
		RCA[15:8]
		E6h
		RCA[7:0]
		24h
		Stuff Bits[15:8]
		00h
		Stuff Bits[7:0]
		00h
		CRC
		XXh
		E
		1

Command  
Cmd 13 - SEND\_STATUS

Timestamp	Command	Response	Data	CRC7	Information
0.011239375 ms	CMD18:READ_MULTIPLE_BLOCK		0042 59C0h	6Ah	
0.011560625 ms		R1 :CMD18:READ_MULTIPLE_BLOCK	0000 0900h	69h	
0.413851875 ms	CMD12:STOP_TRANSMISSION		0000 0000h	30h	
0.414173125 ms		R1b:CMD12:STOP_TRANSMISSION	0000 0B00h	3Fh	
0.976969375 ms	CMD13:SEND_STATUS		E624 0000h	38h	
0.977285 ms		R1 :CMD13:SEND_STATUS	0000 0900h	1Fh	
0.98829625 ms	CMD18:READ_MULTIPLE_BLOCK		0042 5CC0h	4Dh	
0.9886175 ms		R1 :CMD18:READ_MULTIPLE_BLOCK	0000 0900h	69h	
1.330894375 ms	CMD12:STOP_TRANSMISSION		0000 0000h	30h	
1.331215625 ms		R1b:CMD12:STOP_TRANSMISSION	0000 0B00h	3Fh	
2.150086875 ms	CMD13:SEND_STATUS		E624 0000h	38h	
2.1504025 ms		R1 :CMD13:SEND_STATUS	0000 0900h	1Fh	
2.161419375 ms	CMD18:READ_MULTIPLE_BLOCK		0043 4000h	0Ah	
2.161740625 ms		R1 :CMD18:READ_MULTIPLE_BLOCK	0000 0900h	69h	

SD/eMMC decoder

Parameter :

DAT0 = ☐ 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.

add the command reserved:

The 'User Defined Settings' dialog box contains a table with two columns: 'Command' and 'Description'. The table has 14 rows. The first two rows are pre-filled: 'Cmd 1' with 'Test1' and 'Cmd 5' with 'Test2'. The remaining 12 rows are empty and each contains the text '(Double Click...)' in the 'Command' column. At the bottom right of the dialog are 'OK' and 'Cancel' buttons.

The 'Command Description' dialog box features a dropdown menu with 'Cmd 5' selected and a text input field containing 'Test2'. Below these are 'OK' and 'Cancel' buttons.

#### 4. Timer and Counter

Please refer to the Clause Trigger

## Serial Flash Trigger

The settings dialog box below.

Mode	Min. S/R	Max. S/R	Available...	Min. M...	Max. Mem.
800M	800MHz	800MHz	9	256	8M
400M	400MHz	400MHz	18	256	4M
200M	1Hz	200MHz	Adjustable	256	Adjustable
UART Trigger	Baud Rat...	Baud Rat...	Adjustable	256	Adjustable
CAN Trigger	Data Rate...	Data Rate...	Adjustable	256	Adjustable
I2C Trigger	1Hz	200MHz	Adjustable	256	Adjustable
I2S Trigger	1Hz	200MHz	Adjustable	256	Adjustable
SPI Trigger(800M)-9	800MHz	800MHz	9	256	8M
SPI Trigger	1Hz	200MHz	Adjustable	256	Adjustable
SVID Trigger	1Hz	200MHz	Adjustable	256	Adjustable
Serial Flash Trigger(800M)-9	800MHz	800MHz	9	256	8M
SD/eMMC Trigger	1Hz	200MHz	Adjustable	256	Adjustable
LPC Trigger	1Hz	200MHz	Adjustable	256	Adjustable
NAND Flash Trigger	1Hz	200MHz	Adjustable	256	Adjustable
SVI2 Trigger	1Hz	200MHz	Adjustable	256	Adjustable
USB 1.1 Trigger	1Hz	200MHz	Adjustable	256	Adjustable
External Clock	1Hz	200MHz	Adjustable	256	Adjustable

## Trigger Settings

Select Serial Flash Trigger Settings dialog box below.

**Serial Flash / SPI-NAND Flash Trigger Settings**

**Channel** (1)

CS# CH 0  
SCLK CH 1  
SI/SIO0 CH 2  
SO/SIO1 CH 3  
WP#/SIO2 CH 4  
Hold#/SIO3 CH 5

**CS# Glitch Trigger** (4)

None  
Width < 1.25 ns

**tSHSL - 5 ns** (5)

**tCLQV - 8.75 ns**

**Run** (2)

State 1  
State 2  
State 3

Logic Condition (3)

Event 1 + OR

Single Mode  
8 Cycles  
Command = 0Bh AND XXh

Address  
16b 24b 32b = XX0000h

Quad Mode  
6 Cycles  
Dummy cycles  
2 Dummy cycle

Data  
Quad Mode  
Data Offset 0 Byte(s)  
4 I/O Data  
In Out  
XXh 01h 02h 03h  
0 1 2 3

Advanced Setting >>

Undo Redo (6)

Pre-Trigger

Pass Count 0

Load Save Default OK Cancel

### 1. Channel:

Select channels

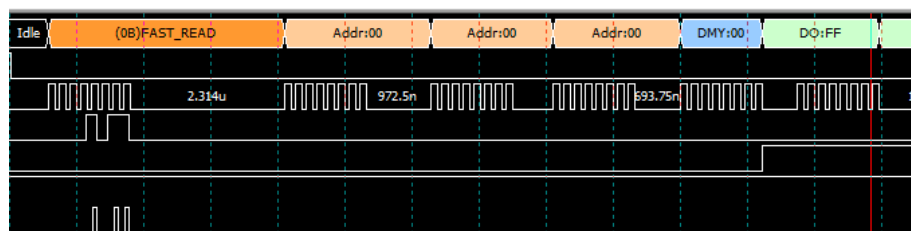
### 2. Clause trigger settings

Please reference Clause Trigger chapter

### 3. Trigger settings

The setting result will show in the trigger flow chart.

-	S	D	Q	Command
Single Mode 8 Cycles				
= 0Bh AND 0Bh				
-	S	D	Q	Address
16b 24b 32b = 000000h				
Single Mode 24 Cycles				
Dummy cycles 8 Dummy cycle				
-	S	D	Q	Data
Single Mode 1 I/O Data				
<input checked="" type="checkbox"/> Data Offset 0 Byte(s)				
In	Out	= FFh XXh XXh XXh		
0 1 2 3				



Select the mode of command:

- S D Q	- S D Q	- S D Q	- S D Q
	Single Mode 8 Cycles	Dual Mode 4 Cycles	Quad Mode 2 Cycles
Don't Care	Single Mode	Dual Mode	Quad Mode



Specified values in command, address and data fields, or input 'X' for don't care.

Check ☒ Data Offset to trigger on data at specified position.

EX: if Data Offset = 0, Data = FFh XXh XXh XXh. It will trigger on Data0= FFh and Data1, 2, 3=any data.

Dummy cycles is needed when data field is specified, it could be 0 to 8 clocks wide

depends on the waveform.

Click  button to switch NOT trigger , it can have 2 input values when NOT trigger was selected.

#### 4. CS Glitch Trigger

Trigger when CS glitch (High/Low/Either) pulse. The glitch pulse setting can be from 0.625ns to 80ns.

#### 5. tSHSL and tCLQV.

#### 6. Redo/Undo



**Protocols Select:** Select Protocols, it provides SMBus / PMBus trigger function.

**Triggers:** Provide the Repeat Start / Stop / ACK / NACK and Check PEC triggers.

**Redo / Undo:** Click these buttons to redo/restore previous setting.

**Trigger settings:**

Fields	
Address	<div>Write <input type="button" value="v"/> X <input type="button" value="v"/></div> <div>0Bh</div>
Command	<div>ACK <input type="button" value="v"/></div> <div>01h</div>
Data	
<div> <div>ACK <input type="button" value="v"/></div> <div>ACK <input type="button" value="v"/></div> <div>NACK <input type="button" value="v"/></div> <div>X <input type="button" value="v"/></div> </div> <div> <div>2Ch</div> <div>01h</div> <div>8Eh</div> <div>XXh</div> </div>	

Provide the Wr/Rd Address / Command / Data and Acknowledge triggers.



## SVI2 Trigger

Select SVI2 Trigger Settings.

Mode	Min. S/R	Max. S/R	Available...	Min. M...	Max. Mem.	
+ SPI Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
+ SVID Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
+ Serial Flash Trigger(800M)-9	800MHz	800MHz	9	256	8M	
+ SD/eMMC Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
+ LPC Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
+ NAND Flash Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
- SVI2 Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto	
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Auto	
SVI2 Trigger-36	1Hz	200MHz	Adjustable	256	2M	
SVI2 Trigger-18	1Hz	200MHz	Adjustable	256	4M	
SVI2 Trigger-12	1Hz	200MHz	Adjustable	256	6M	
SVI2 Trigger-9	1Hz	200MHz	Adjustable	256	8M	
SVI2 Trigger-6	1Hz	200MHz	Adjustable	256	12M	
SVI2 Trigger-4	1Hz	200MHz	Adjustable	256	18M	
+ USB 1.1 Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
+ External Clock	1Hz	200MHz	Adjustable	256	Adjustable	

## Trigger Settings

The SVI2 trigger settings dialog box is as below.

The SVI2 Trigger Settings dialog box is shown with the following configuration:

- Channel:** SVC (CH 0), SVD (CH 1), SVT (CH 2), AUX (CH 3).
- Error Detect:** SVD Packet Error, SVT Packet Error.
- State 1 Logic:** Run (blue arrow) to State 1, True (blue arrow) to State 1, False (green arrow) to State 1, Trigger (red arrow) to State 1.
- Event 1 Logic:** OR logic condition.
- SVD Packet:** VDD (xh), VDDNB (xh), PS10\_L (xh), PS11\_L (xh), TFN (xh), Load Line Slope Trim (xh), Offset Trim (xh).
- SVT Packet:** SVT1 (xh), SVT0 (xh), VDD Voltage (xxxh), VDDNB Voltage (xxxh).
- AUX:** (xh).
- Buttons:** Undo, Redo, Pre-Trigger (checked), Pass Count (0), Load, Save, Default, OK, Cancel.

**Channel:** Select channels, SVT checked default.

**Error Detect:** Detect SVD / SVT packet error. It will check the size of packet.

**Redo/Undo:** Click these buttons to redo/restore previous setting.

**Trigger settings:** Trigger SVD / SVT packet.

### SVD Packet:

SVD Packet configuration window showing the following settings:

- VDD: 0h Not VDD
- VDDNB: 0h Not VDDNB
- PSIO\_L: 1h
- PSI1\_L: 1h
- TFN: 1h
- Load Line Slope Trim: 3h Initial LL Slope
- Offset Trim: 2h Use Initial Offset
- SVID: 0.60000V (98h)

Timestamp	VDD	VDDNB	SVID Code	PSI	TFN	Slope Trim	Offset Trim	SVT	Volt	Volt/Current
-0.00144 ms	0	0	0.60000V (98)	3	1	Initial LL Slope(3)	Use Initial Offset(2)			
0.016945 ms	VDD(1)	0	1.30000V (28)	3	0	Initial LL Slope(3)	Use Initial Offset(2)			

### SVT Packet:

SVT Packet configuration window showing the following settings:

- SVT1: 1h
- SVT0: 1h
- VDD Voltage: 1.33750V (122h)
- VDDNB Voltage: 1.15625V (13Fh)

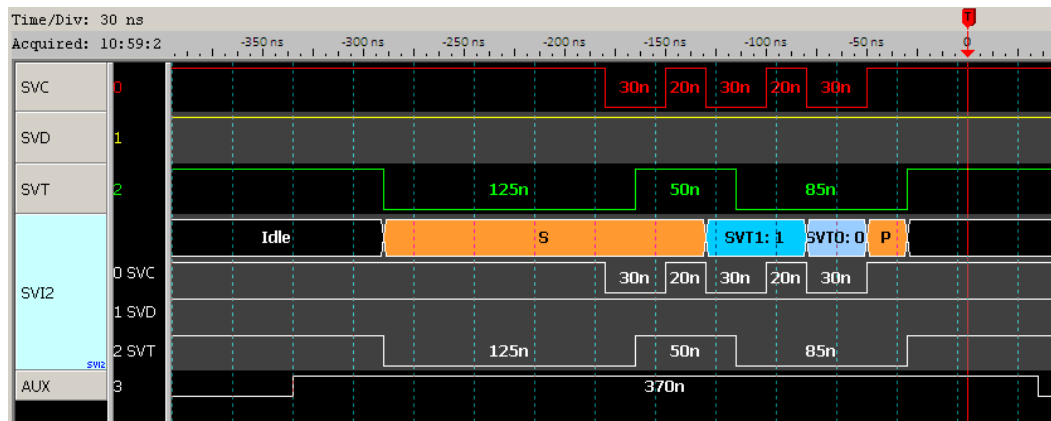
Timestamp	VDD	VDDNB	SVID Code	PSI	TFN	Slope Trim	Offset Trim	SVT	Volt	Volt/Current
-0.001175 ms								3	1.33750V (122)	1.15625V (13F)
0.018485 ms								3	1.16250V (13E)	1.15000V (140)
0.038145 ms								3	0.97500V (15C)	1.15000V (140)

### VOTF Complete + AUX High

SVT Packet  
SVT1  
= 1h  
SVT0  
= 0h  
VDD Voltage  
= xxxh  
VDDNB Voltage  
= xxxh

AUX  
1h

Timestamp	Offset Trim	SVT	Volt	Volt/Current	Error	Description
-0.014045 ms		3	0.97500V (15C)	1.15000V (140)		Voltage Only
-0.00029 ms		2				VOTF Complete
0.00561 ms		3	0.75625V (17F)	1.15000V (140)		Voltage Only



**Other settings:** Switch ☐ ☒ by clicking ☐ button when trigger VDD,

VDDNB...Offset Trim in SVD packet.

Switch ☐ ☒ ☐ ☐ by clicking ☐ button when trigger SVID in SVD packet.

Switch ☐ ☒ by clicking ☐ button when trigger SVT1, SVT0 in SVT packet.

Switch ☐ ☒ ☐ ☐ by clicking ☐ button when trigger VDD Voltage /

VDDNB Voltage in SVT packet.

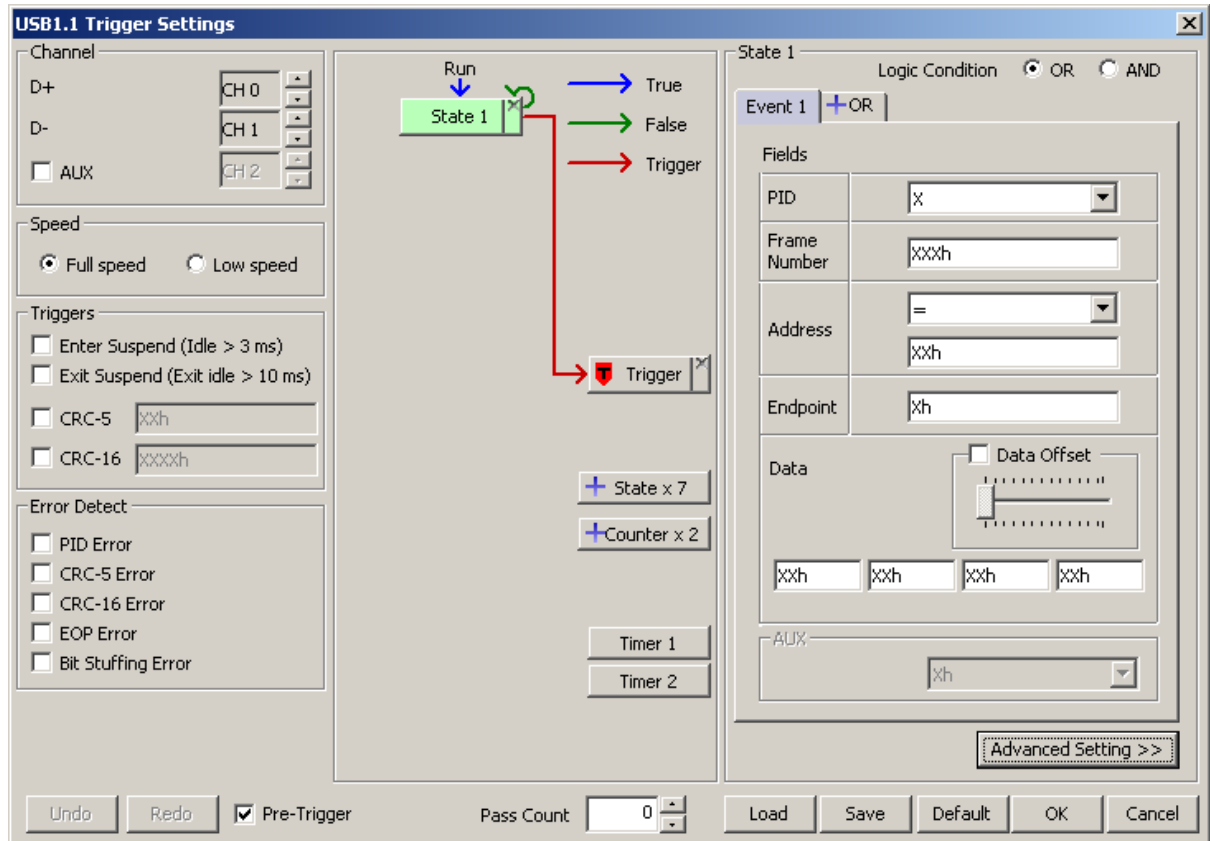
## USB1.1 Trigger

Select USB1.1 Trigger Settings.

Mode	Min. S/R	Max. S/R	Available...	Min. M...	Max. Mem.	
+  SPI Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
+  SVID Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
+  Serial Flash Trigger(800M)-9	800MHz	800MHz	9	256	8M	
+  SD/eMMC Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
+  LPC Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
+  NAND Flash Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
+  SVID Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
-  USB 1.1 Trigger	1Hz	200MHz	Adjustable	256	Adjustable	
Transitional Storage-32	200MHz	200MHz	Fixed	Auto	Auto	
Transitional Storage-8	200MHz	200MHz	Fixed	Auto	Auto	
<b>USB 1.1 Trigger-36</b>	1Hz	200MHz	Adjustable	256	2M	
USB 1.1 Trigger-18	1Hz	200MHz	Adjustable	256	4M	
USB 1.1 Trigger-12	1Hz	200MHz	Adjustable	256	6M	
USB 1.1 Trigger-9	1Hz	200MHz	Adjustable	256	8M	
USB 1.1 Trigger-6	1Hz	200MHz	Adjustable	256	12M	
USB11 Trigger-4	1Hz	200MHz	Adjustable	256	18M	
+  External Clock	1Hz	200MHz	Adjustable	256	Adjustable	

## Trigger Settings

The USB1.1 trigger settings dialog box is as below.



**Channel:** Select channels. AUX unchecked default.

**Speed:** Speed mode. Select Full / Low speed.

**Triggers:** Trigger Enter Suspend / Exit Suspend / CRC-5 / CRC-16.

**Error Detect:** Trigger PID Error / CRC-5 Error / CRC-16 Error / EOP Error / Bit Stuffing Error.

**Redo/Undo:** Click these buttons to redo/restore previous setting.

**Trigger settings:** Trigger PID:SETUP; Address = 01h; Endpoint = 0h

Fields

PID: SETUP

Frame Number: XXXh

Address: =

Endpoint: 01h

Data: Data Offset

XXh XXh XXh XXh

Timestamp	No.	PID	Frame Number	Address	Endpoint	CRC5	DATA	ASCII	CRC16	Packet Duration
-0.00591 ms	357	SOF (TOKEN)	0288			01				3 us(33 Bits)
-0.002245 ms	358	SETUP (TOKEN)		01	00	17				3 us(33 Bits)
0.00142 ms	359	DATA0 (DATA)					C0 0C 84 00 00 00 01 00	.....	060E	8 us(96 Bits)
0.010425 ms	360	ACK (HANDSHAKE)								1 us(17 Bits)
1.009085 ms	361	SOF (TOKEN)	0289			1E				3 us(33 Bits)
1.012755 ms	362	OUT (TOKEN)		01	02	03				3 us(33 Bits)
1.01642 ms	363	DATA0 (DATA)					5A 0F 66 01	Z.f.	EC06	6 us(64 Bits)
1.022755 ms	364	ACK (HANDSHAKE)								1 us(17 Bits)

Trigger PID:DATA0; Data: 5Ah, 0Fh, 66h, 01h

Fields

PID: DATA0

Frame Number: XXXh

Address: =

Endpoint: Xh

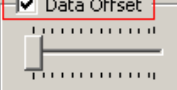
Data: Data Offset

5Ah 0Fh 66h 01h

Timestamp	No.	PID	Frame Number	Address	Endpoint	CRC5	DATA	ASCII	CRC16	Packet Duration
-45.379 ms	1	SOF (TOKEN)	0289			1E				3 us(33 Bits)
-45.375335 ms	2	OUT (TOKEN)		01	02	03				3 us(33 Bits)
-45.371665 ms	3	DATA0 (DATA)					5A 0F 66 01	Z.f.	EC06	5 us(64 Bits)
-45.365335 ms	4	ACK (HANDSHAKE)								2 us(17 Bits)
-45.362915 ms	5	SOF (TOKEN)	0288			01				3 us(33 Bits)
-45.35925 ms	6	SETUP (TOKEN)		01	00	17				3 us(33 Bits)
-45.355585 ms	7	DATA0 (DATA)					C0 0C 84 00 00 00 01 00	.....	060E	8 us(96 Bits)
-45.346585 ms	8	ACK (HANDSHAKE)								2 us(17 Bits)

Trigger PID:DATA0; 4 Byte Data(Fixed offset): C0h, 0Ch, 84h, 00h

Fields

PID	DATA0
Frame Number	xxxh
Address	= xxh
Endpoint	xh
Data	<input checked="" type="checkbox"/> Data Offset 
	<div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid red; padding: 2px;">C0h</div> <div style="border: 1px solid red; padding: 2px;">0Ch</div> <div style="border: 1px solid red; padding: 2px;">84h</div> <div style="border: 1px solid red; padding: 2px;">00h</div> </div> <div style="display: flex; justify-content: space-around; margin-top: 2px;"> <span>0</span> <span>1</span> <span>2</span> <span>3</span> </div>

Timestamp	No.	PID	Frame Number	Address	Endpoint	CRC5	DATA	ASCII	CRC16	Packet Duration
2.03475 ms	369	SOF(TOKEN)	0289			1F				3 us(33 Bits)
2.038415 ms	370	OUT(TOKEN)		01	02	03				3 us(33 Bits)
2.042085 ms	371	DATA0(DATA)					5A 0F 66 01	Z.f.	EC06	5 us(64 Bits)
2.04842 ms	372	ACK(HANDSHAKE)								1 us(17 Bits)
2.050835 ms	373	SOF(TOKEN)	0288			01				3 us(33 Bits)
2.054505 ms	374	SETUP(TOKEN)		01	00	17				3 us(33 Bits)
2.05817 ms	375	DATA0(DATA)					C0 0C 84 00 00 00 01 00	.....	060E	8 us(96 Bits)
2.06717 ms	376	ACK(HANDSHAKE)								1 us(17 Bits)

**Other settings:** Select =, <, >, <=, >=, InRange, Not InRange in the Address field.