



## How to use the decoding function of SDS2000 series oscilloscopes?

### The decoding function:

Recently, Siglent launched the SDS2000 series Super phosphor oscilloscopes. With the innovative SPO (Super Phosphor Oscilloscope) technology, the SDS2000 is featured with several advanced characteristics that simple digital oscilloscopes do not have, including high capture rate, deep memory, digital trigger system and gray scale display. In addition, the SDS2000 oscilloscopes also include a variety of serial data protocol triggering & decoding features (IIC/SPI/UART/RS232/CAN/LIN).

### IIC Bus Protocol

IIC (also known as I2C or Inter Integrated-Circuit) , mainly consist of the bidirectional serial clock line SCL and bidirectional serial data line SDA. Signals transferred through IIC includes start signal, address signal, data signal, and end signal.

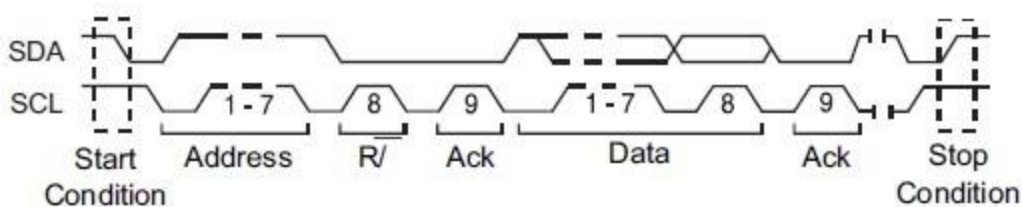


Figure 1

After connecting the oscilloscope to SDK and SCK, you will need to specify the input signal voltage level. First, set the trigger type to I2C, under which the signal could trigger on a start / stop condition, restart, no response, EEPROM data, or trigger

on a read/write frame with specified device address and data value. Then press the DECODE button to enter the decoding menu, and open the decoding list to clearly display decoded data in a list.

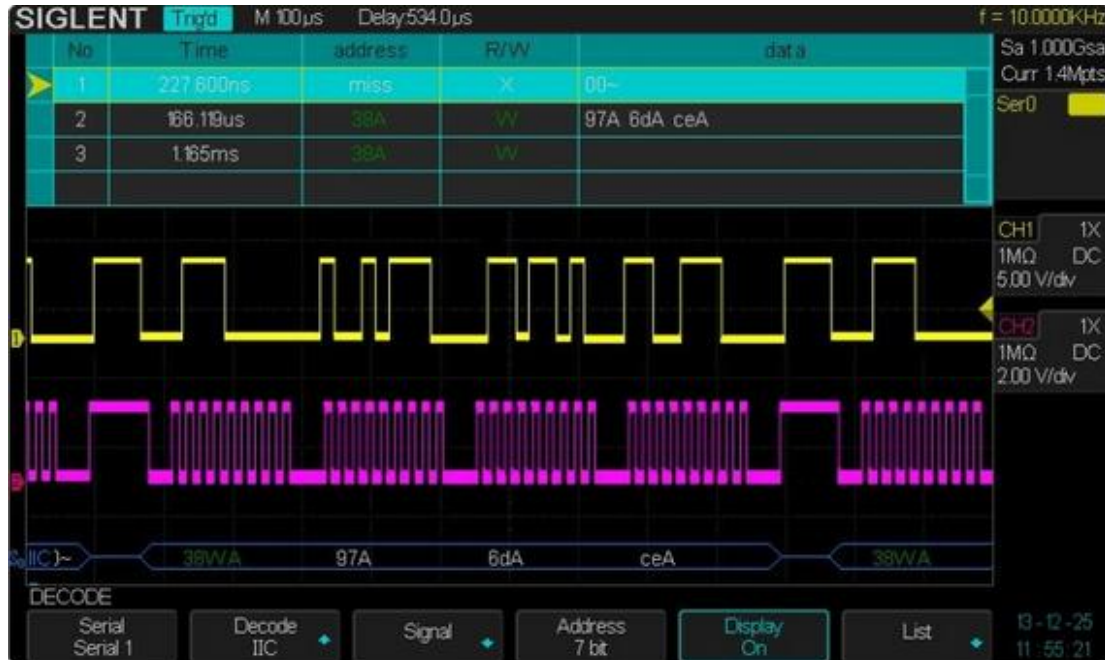


Figure 2

In hexadecimal decoded data:

The address value displays on start of the frame in dark green;

The read/write value displays closely after the address value in “W” / “R”;

“A” indicates response; “~ A” indicates no response.

Data value display in white.

If the space of the frame is insufficient, data accessed cannot display but is indicated by “...”.

### SPI Bus Protocol

SPI (Serial Peripheral Interface) is a communication bus featured with point to point, full-duplex, 4-bus synchronous serial. Signals including CS, SCK, MOSI and MISO are all available to be transferred.

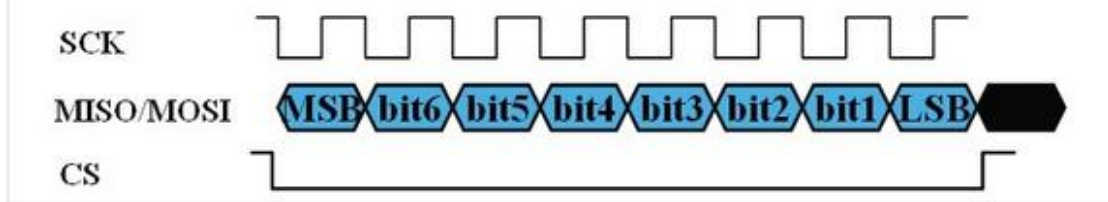


Figure 3

When using a dual-channel oscilloscope in SPI type, waveforms can be triggered only by TRIG TIME OUT. First, connect the oscilloscope to the SCK signal and select MISO, and then specify the input signal voltage level. After selecting the SPI setting, a signal at the start of the frame can now be used to trigger the oscilloscope, and the serial data string can be specified from 4~96 bits.



Figure 4

In hexadecimal decoded data:

Data that can range from 4~96 bits display is in the frame in white.

MISO indicates bus that is inputted from the main machine and outputted from slave machine.

MOSI indicates bus that is inputted from the slave machine and output from the main machine.

In decoding list:

Time (timestamp) indicates the horizontal displacement that the current data

relative to the trigger position.

MISO indicates bus that input from main machine and output from slave machine.

MOSI indicates bus that input from slave machine and output from main machine.

**UART/RS232 Bus Protocol**

UART (Universal Asynchronous Receiver / Transmitter) is an Universal Asynchronous Receiver Transmitter, working at the data link layer. As a low-speed communication protocol, UART is widely used in the communication field and other occasions.

As a kind of UART, Asynchronous serial communication protocol translates data characters one by one.

Connect a Rx (receive) or Tx (transmit) signal to the oscilloscope, and then specify the input signal voltage level. After setting the oscilloscope to RS232, you can select to trigger on a start signal, an end signal, error signal or any data, you will then need to set the Baud rate, Data length, Parity check and Stop it.

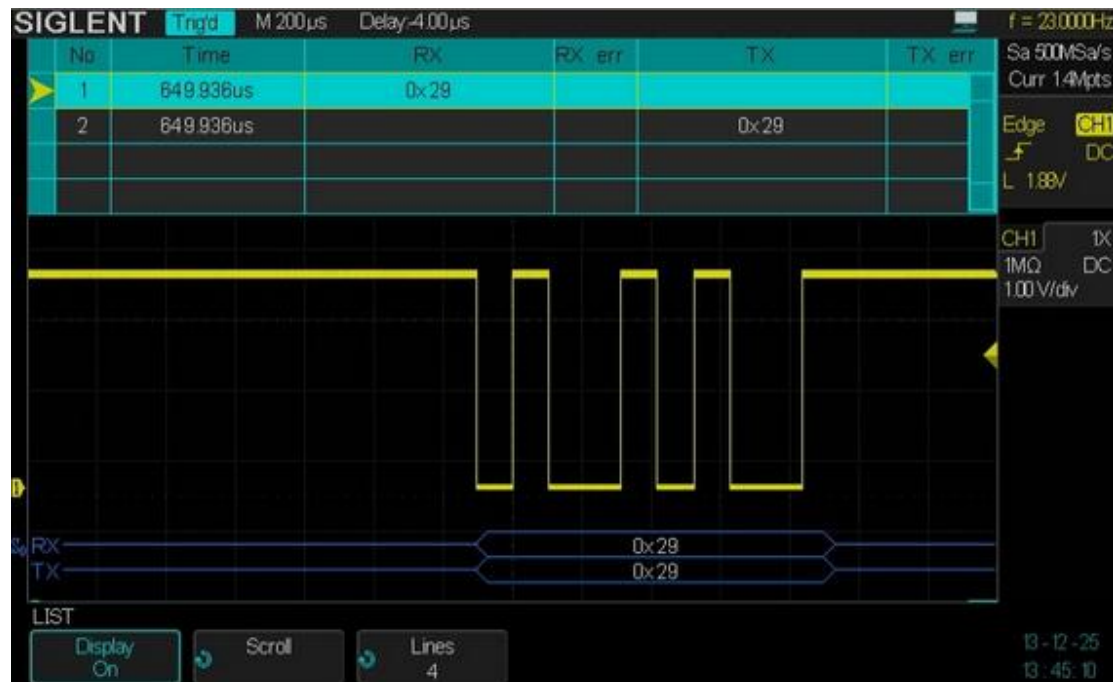


Figure 5

In hexadecimal decoded data:

Data whose length I variable from 4 to 96 bits display in the frame in white.

RX indicates receiving data;

TX indicates transmitting data.

In the decoding list:

Time (timestamp) indicates the horizontal displacement of the current data relative to the trigger position.

RX indicates receiving data while TX indicates transmitting data.

RX err indicates a parity error or unknown error when receiving data.

TX err indicates a parity error or unknown error when transmitting data.

### CAN Bus Protocol

CAN (Controller Area Network), is a controller area network widely used in the automotive industry, mainly for various processes (equipment) monitoring and control. The message transfer by CAN consist of four different kinds of frame: data frame, remote frame, error frame, overload frame. Wherein the data frames consists of seven different potential fields: frame start, arbitration field, control field, data field, CRC field, response field and frame end.

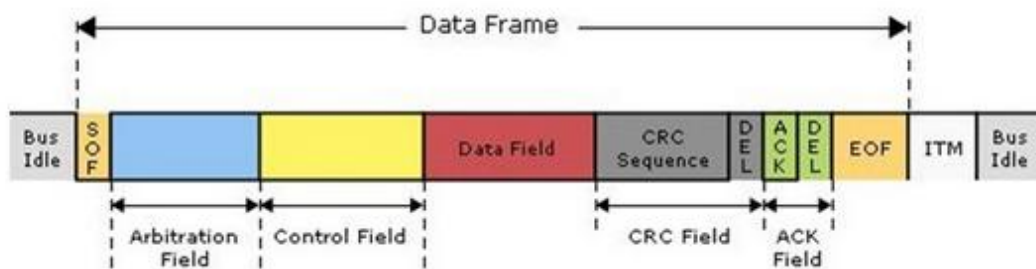


Figure 6

Connect a CAN-H or CAN-L signal to the oscilloscope, and then specify the input signal voltage level. After setting the oscilloscope to CAN, you could select to trigger on start condition, remote frame ID, data frame ID and “data frame ID + arbitrary data”, then set a desired baud rate.

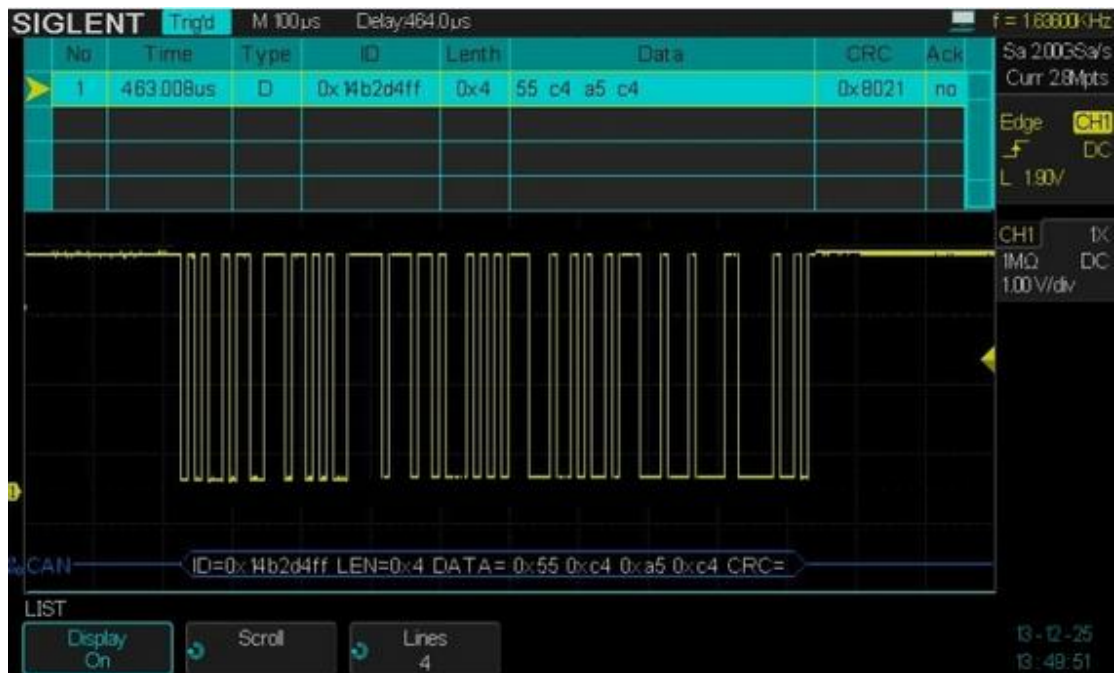


Figure 7

In hexadecimal decoded data: ID, LEN, DATA, CRC all display in the frame in white.

In the decoding list:

Time (timestamp) indicates the displacement of the current data relative to the trigger position.

The frame type consists of data frame (indicated by “D”) and remote frame (indicated by “R”). Frame ID will automatically detect 11 or 29 bits. Length indicates data length. Data indicates data byte. CRC indicates Cyclic Redundancy Check. Ack indicates acknowledgement bit.

### LIN Bus Protocol

LIN (Local Interconnect Network) is a series communication bus in the local Internet, which is featured with low-cost, low speed, mainly for the low-end Distributed applications in automobiles. LIN is the lower level network of CAN.

Connect a LIN signal to the oscilloscope, and then specify the input signal voltage level. After setting the oscilloscope to LIN, you could select to trigger on start signal, frame ID, frame ID + data or frame error, then set a desired baud rate.



Figure 8

In hexadecimal decoded data: ID, LEN, DATA, CHK display in the frame in white.

In the decoding list:

Time (timestamp) indicates the horizontal displacement of the current data relative to the trigger position.

ID indicates frame ID.

Data length indicates data length.

ID Check indicates ID parity.

Data indicates data bytes.

Checksum indicates data checksums.