# Table of contents

# Function Description

This document is used to tell users about the use of generic fsk RX mode and there is a simple example in this document. Through this document, user can learn how to receive data in generic fsk RX mode.

This document needs to be used with the generic fsk RX mode demo (gen_fsk_rx demo). The function of the generic fsk RX mode demo is to receive data from transmitter which in the generic fsk TX mode.

# Platform

if you want to receive data with the generic fsk RX mode, you need to configure the generic fsk RX mode environment as follows。

## Hardware Platform

- Telink TLSR8258 EVK(C1T139A30_V1.2)
- Telink Burning EVK(V1.0.0.0)

Note:　You need an external antenna to work normally.



Figure 1 Telink burning EVK



Figure 2 Telink TLSR8258 EVK

## Software Platform

- Telink Burning and Debugging Tool
- The generic fsk RX mode software (GEN_FSK_RX.bin)
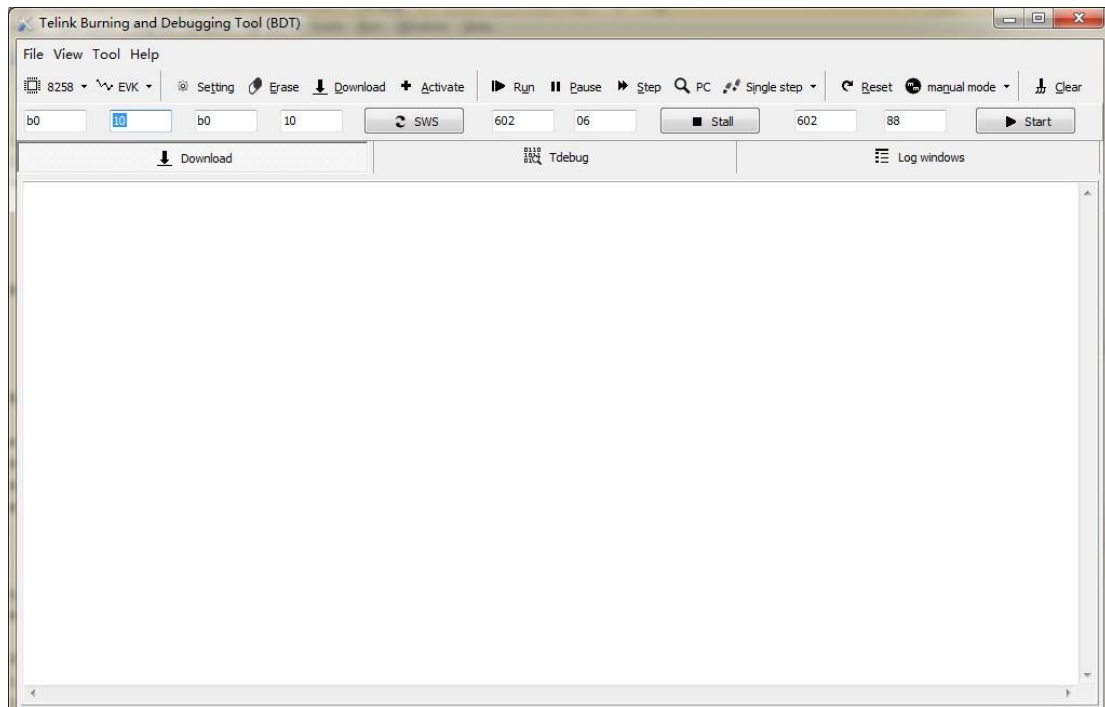- Telink Packet Sniffer (tl_uart.exe)
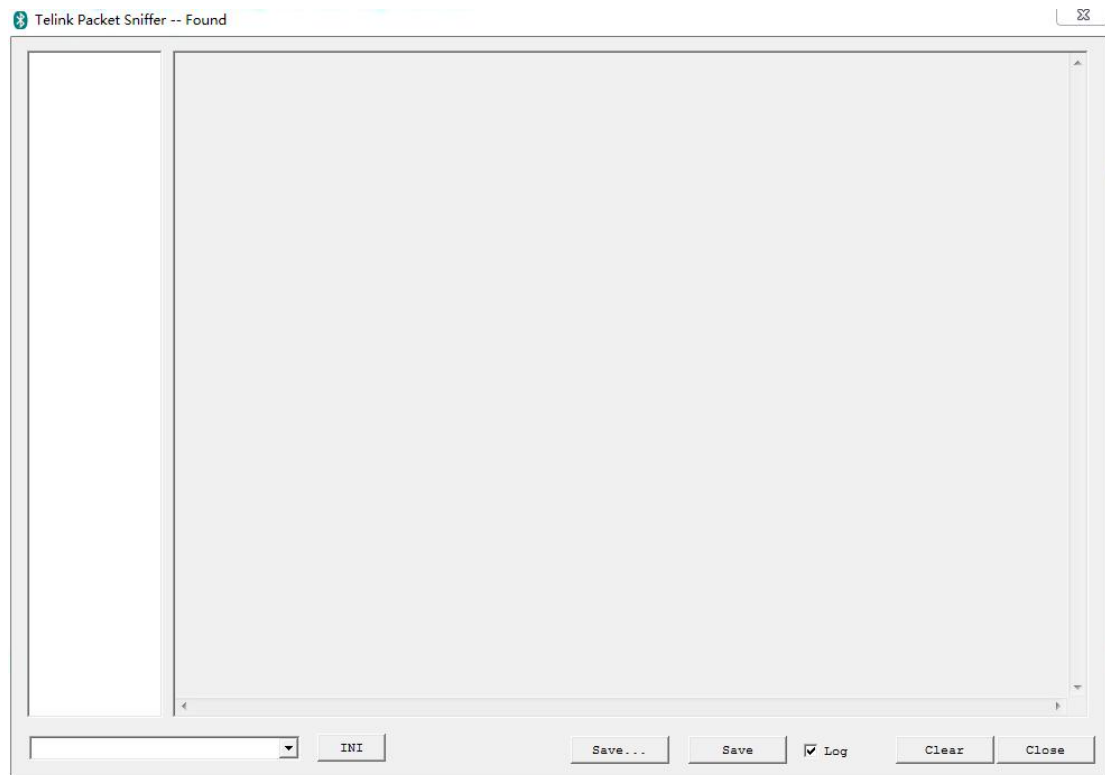
Figure 3 Telink Burning and Debugging Tool


Figure 4 Telink Packet Sniffer

# Burning Step

## Step1

Connect Telink Burning EVK and Telink TLSR8258 EVK with an usb cable, then connect with PC. Note：you also need connect Telink Burning EVK SWM pin and Telink TLSR8258 EVK SWS pin with a wire.
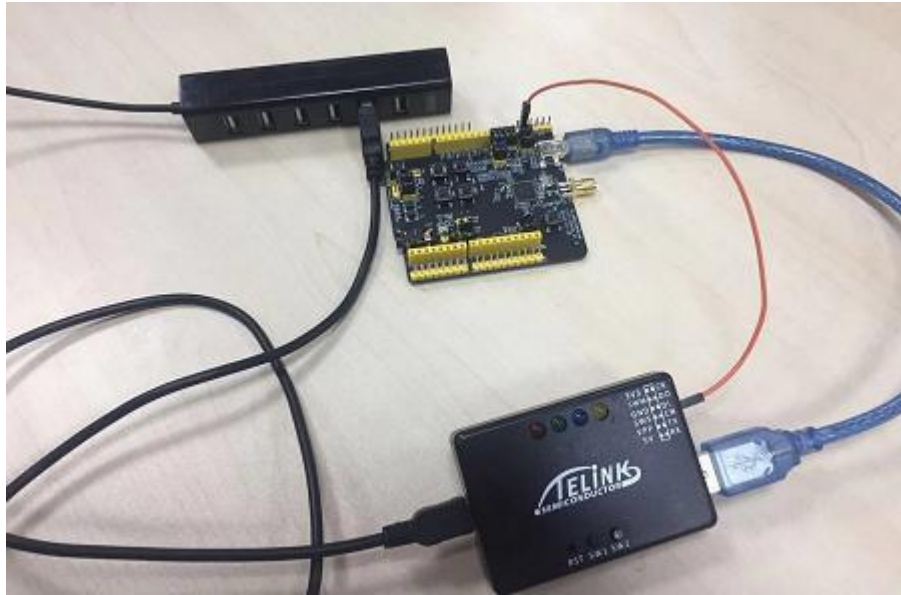


Figure 4 Connect way

## Step2

Open Telink Burning and Debugging Tool (BDT) and select 8258 chip, then click on SWS. If BDT shows ''no evk device!'', you should check whether the hardware connection is correct. If BDT shows "Swire ok!" that means hardware is ok.
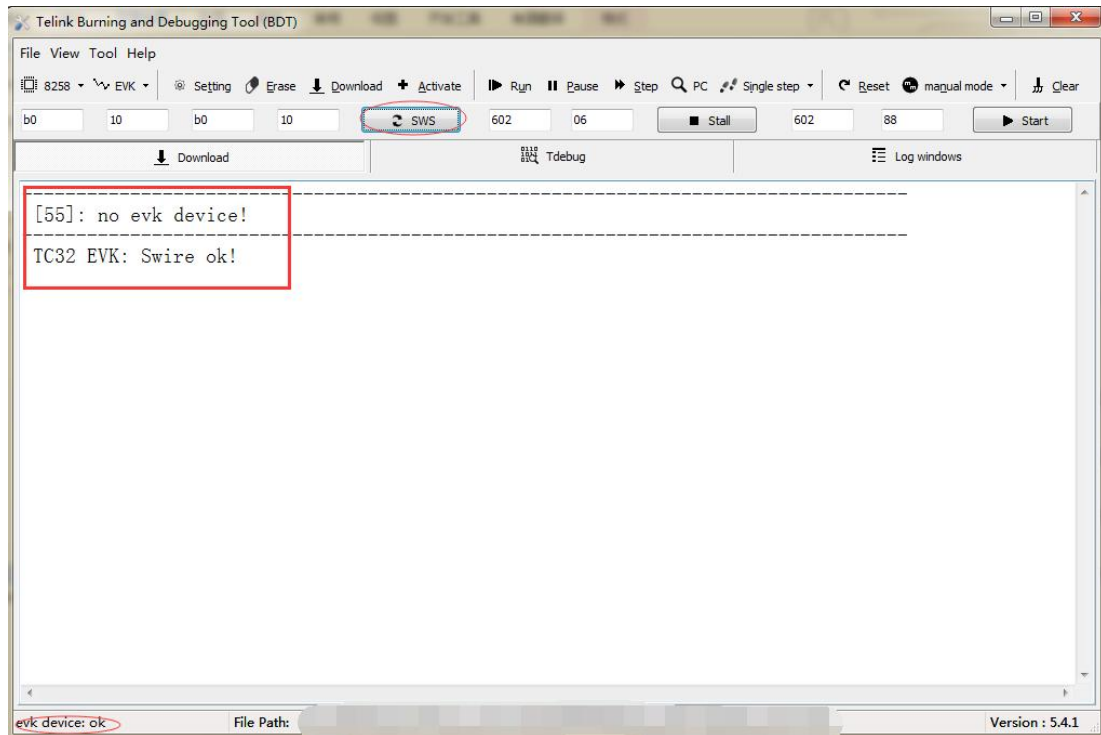
Figure 5 BDT information

## step3

Burning the program to the 8258 TLSR8258 EVK. You should click on "file" button to select your target bin file, then click on "Download" button, it will show download information correctly. Whereas, if the BDT shows "Swire err!" check your hardware or click on "Active" button, it may solves your problem!
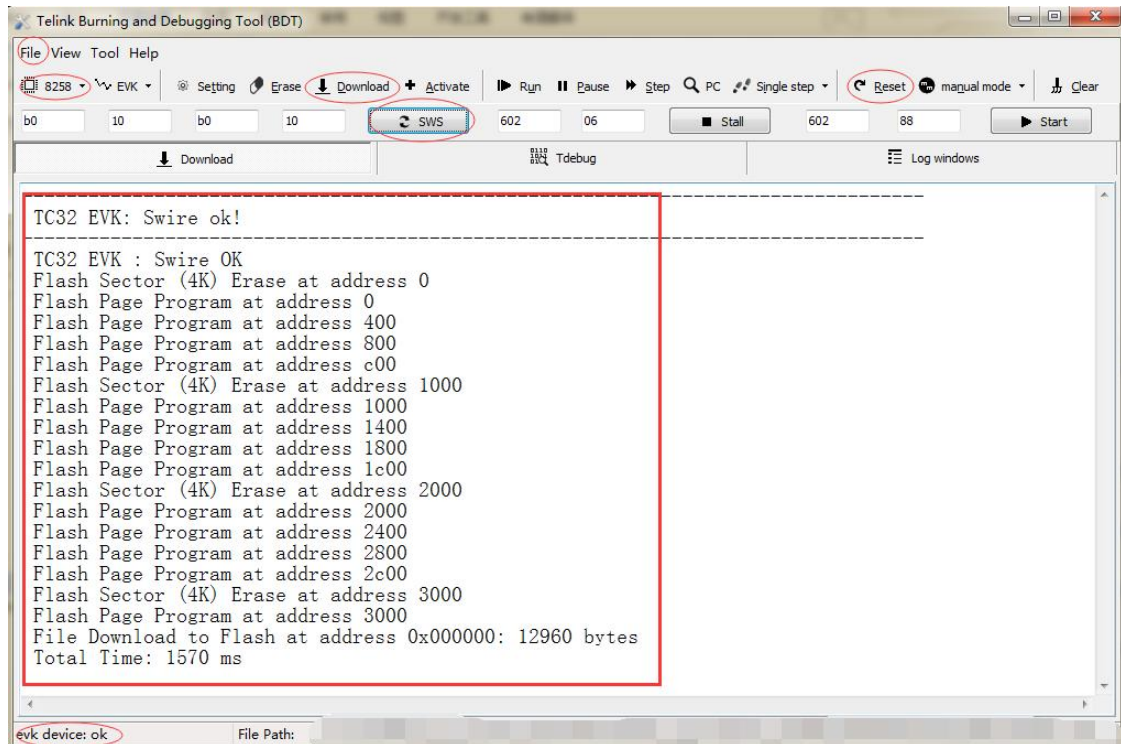
Figure 6 BDT information

## Code Structure

### System initialization

Include cpu initialization ,read calibration value,usb_log initialization and clock initialization.

```
cpu_wakeup_init();

user_read_flash_value_calib();

usb_loginit(); //config the USB Log function

clock_init(SYS_CLK_24M_Crystal);
```

### RF configuration

You need to set the basic parameter of RF in generic fsk RX mode before you send the data.

```
//generic FSK Link Layer configuratioin
gen_fsk_datarate_set(GEN_FSK_DATARATE_2MBPS); //Note that this API must be invoked first before all other APIs
gen_fsk_preamble_len_set(4);
gen_fsk_sync_word_len_set(SYNC_WORD_LEN_4BYTE);
gen_fsk_sync_word_set(GEN_FSK_PIPE0, sync_word); //set pipe0's sync word
gen_fsk_pipe_open(GEN_FSK_PIPE0); //enable pipe0's reception
gen_fsk_tx_pipe_set(GEN_FSK_PIPE0); //set pipe0 as the TX pipe
gen_fsk_packet_format_set(GEN_FSK_PACKET_FORMAT_FIXED_PAYLOAD, 8);
gen_fsk_radio_power_set(GEN_FSK_RADIO_POWER_0DBM);
gen_fsk_rx_buffer_set(rx_buf + rx_ptr*RX_BUF_LEN, RX_BUF_LEN);
gen_fsk_channel_set(7); //set rf freq as 2403.5MHz
gen_fsk_radio_state_set(GEN_FSK_STATE_RX); //set transceiver to basic RX state
gen_fsk_rx_settle_set(89);
WaitUs(90); //wait for rx settle
```

API *gen_fsk_datarate_set* is used for setting RF data rate, we provide 4 bitrate options, 2Mbps, 1Mbps, 500kbps, and 250kbpps.

API *gen_fsk_preamble_len_set* is used for setting preamble length, the length of preamble is

set to range from 1 to 16 bytes.

API *gen_fsk_sync_word_len_set* is used for setting the length of synchronization word, you

can set it at the range from 3 to 5 bytes.

API *gen_fsk_sync_word_set* is used for setting address and pipe.

API *gen_fsk_packet_format_set* is used for setting the packet format of on-air data.

API *gen_fsk_channel_set* is used for setting RF channel, notice that the unit of channel is 500 kHz.

API *gen_fsk_radio_state_*set is used for setting the generic fsk mode, you can choose generic fsk TX mode or generic fsk RX mode.

Other APIs are used to set up the RF basic configuration.

Note: This version of 8258 only has pipe0.

### Generic fsk RX mode data processing

Once the RF initialization is complete, you can start receiving data with API *gen_fsk_rx_payload_get when it's sent. There will be an interruption* when data is received, FLD_IRQ_ZB_RT_EN and FLD_RF_IRQ_RX will be set.

```
rx_payload = gen_fsk_rx_payload_get(rx_packet, &rx_payload_len);
```

## Result analysis

If the data is received successfully, LED D2 will start blinking per 500ms.

To get data information when system running, you can debug Telink Packet Sniffer.
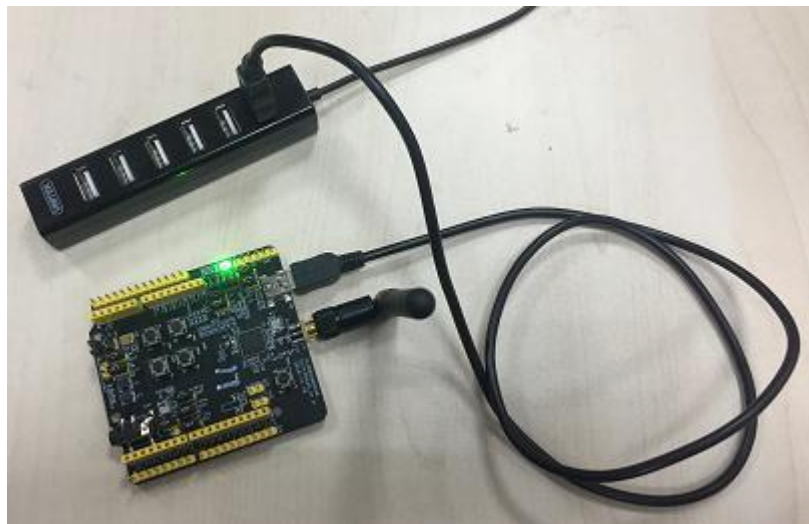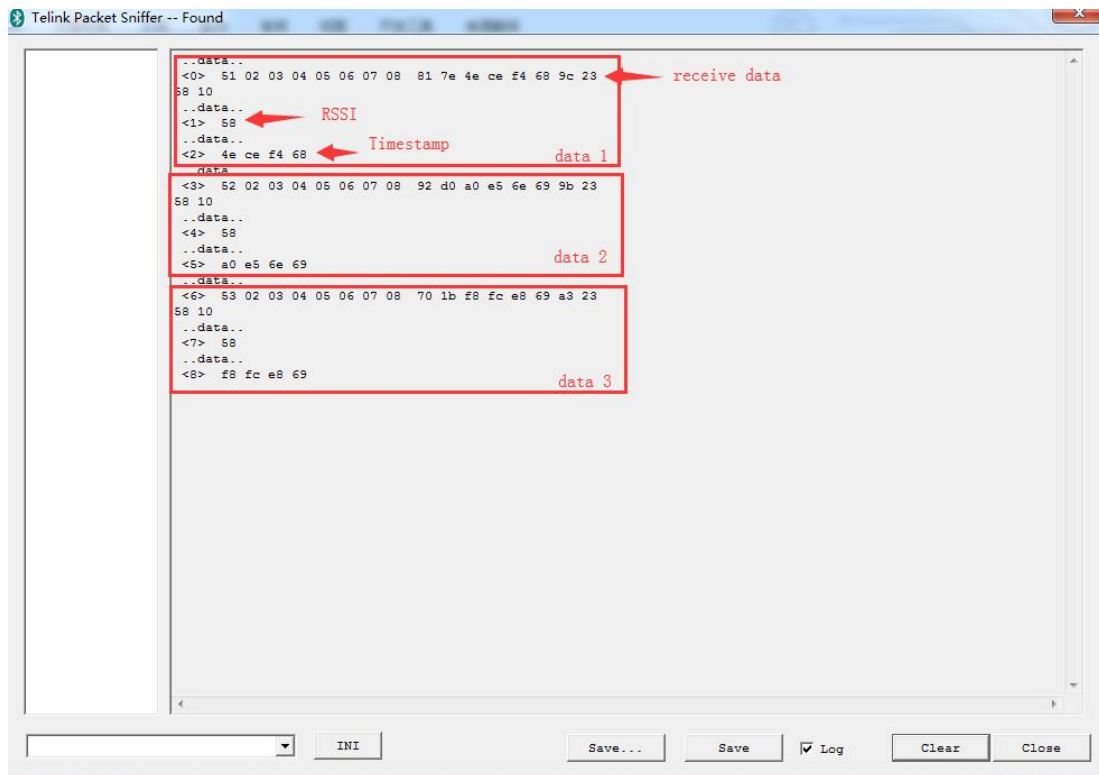


Figure 7 Hardware connection

Figure 8 Telink Packet Sniffer information

From Figure 8, you can know what data you have received.

RSSI: Receiving data power.

Actual Power (dB) = RSSI value – 110

Timestamp: The time interval between two packets.

The time interval (us) = (Timestamp – Last Timestamp)/16