

## Table of contents

Table of contents .....	1
Function Description .....	2
Platform.....	2
Hardware Platform .....	2
Software Platform .....	2
Burning Step .....	3
Step1 .....	3
Step2 .....	4
step3.....	4
Code Structure .....	5
System initialization .....	5
RF configuration .....	5
ESB packet setting .....	6
Result analysis .....	6

## Function Description

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

This document needs to be used with the generic fsk TX mode demo (gen\_fsk\_tx demo). The function of the generic fsk TX mode demo is to send data to receiver which in the generic fsk RX mode.

## Platform

if you want to send data with the generic fsk TX mode, you need to configure the generic fsk TX 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
- generic fsk TX software (GEN\_FSK\_TX.bin)

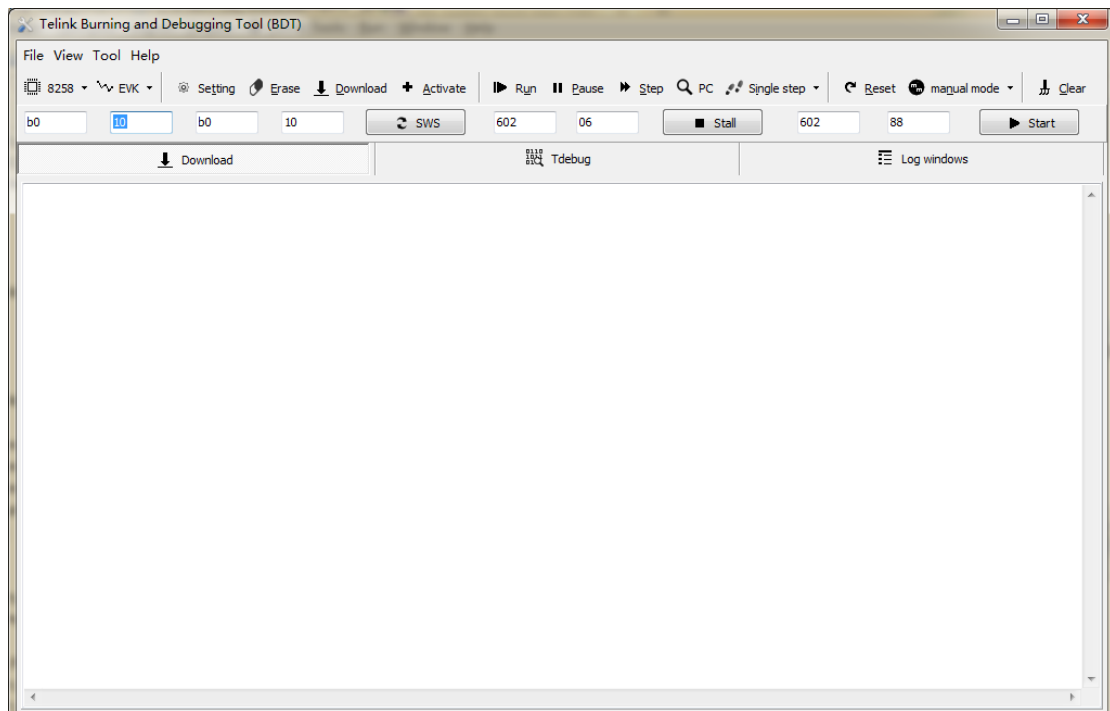


Figure 3 Telink Burning and Debugging Tool

## 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.

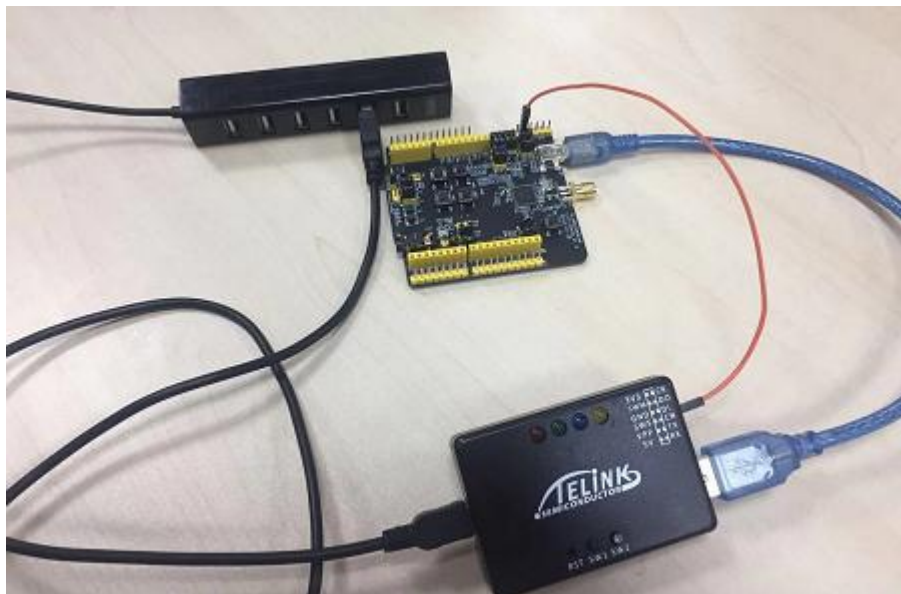


Figure 4 Connect way

## 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.

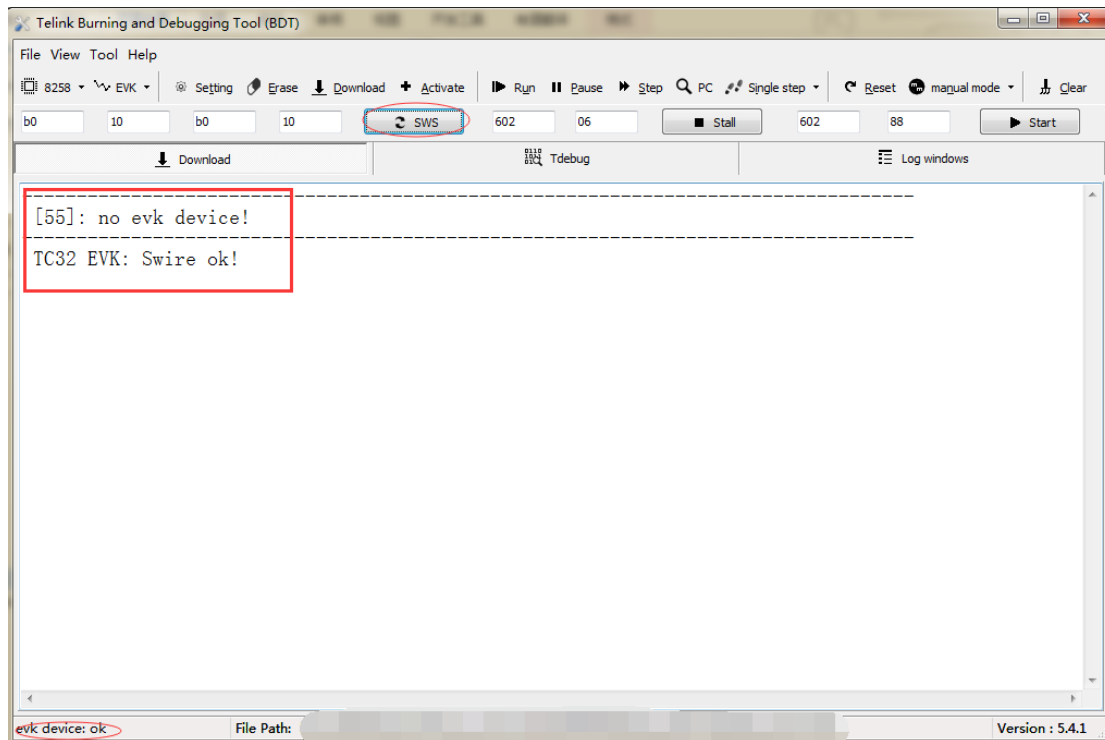


Figure 5 BDT information

## 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!

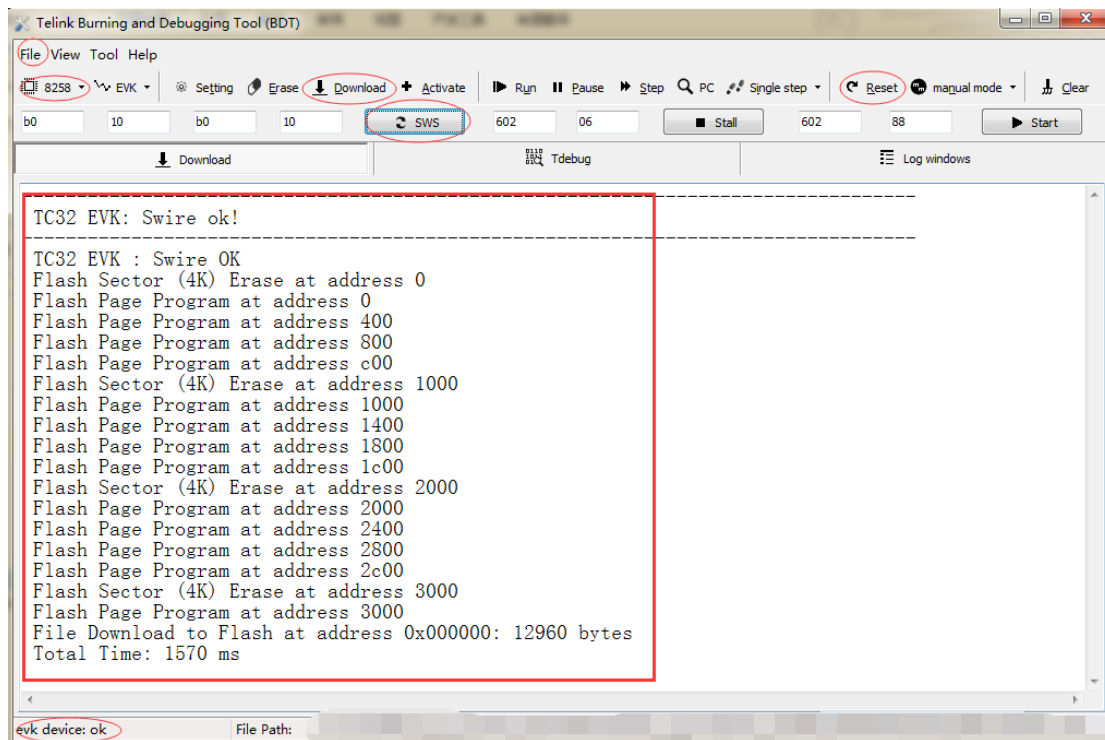


Figure 6 BDT information

## Code Structure

### System initialization

Include cup initialization and clock initialization.

```
cpu_wakeup_init();
clock_init(SYS_CLK_24M_Crystal);
```

### RF configuration

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

```
//generic FSK Link Layer configuration
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, sizeof(tx_payload));
gen_fsk_radio_power_set(GEN_FSK_RADIO_POWER_0DBM);
gen_fsk_channel_set(7); //set rf freq as 2403.5MHz
gen_fsk_radio_state_set(GEN_FSK_STATE_TX); //set transceiver to basic TX state
WaitUs(130); //wait for tx settle
```

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

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

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.

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

## Generic fsk TX mode packet setting

```
gen_fsk_tx_start(tx_buffer); //start the Radio transmission
```

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

The screenshot shows the Telink Burning and Debugging Tool (BDT) interface. The top menu bar includes File, View, Tool, and Help. Below the menu is a toolbar with various icons for settings, erasing, downloading, activating, running, pausing, stepping, and searching. The main window is divided into two panes. The left pane displays a list of variables with their names, addresses, lengths, and values. The right pane shows a memory dump with addresses and hex data, along with a total time measurement.

Variable Name	Addr	Len	Value
gbl_24Mrc_ana_33	42c00	1	00000080
tx_payload	42c04	8	...
irq_stk	42c10	384	...
tx_buffer	42d90	64	...
tx_done_flag	42dd0	1	00000001
gen_fsk_current_channel	42dd2	2	00000007
gen_fsk_current_state	42dd4	1	00000000
pm_tick32kGet_cb	42dd8	4	00000000
RF_TrxState	42ddc	1	00000000
uart_TxIndex	42ddd	1	00000000
tick_32k_calib	42de0	2	00000000
pmParam	42de4	3	00000000
tick_32k_cur	42de8	4	00000000
tick_cur	42dec	4	00000000
g_RFRxPingpongEn	42df0	1	00000000
g_RFMode	42df2	2	00000200

Memory Dump:

```

042c04: 01 02 03 04 05 06 07 08
Total time: 20 ms

042d90: 08 00 00 00 0d 02 03 04 05 06 07 08 00 00 00 00
042da0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
042db0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
042dc0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Total Time: 21 ms
  
```

Annotations in the image:

- A red box highlights the `tx_payload` variable in the left pane.
- A red box highlights the `tx_buffer` variable in the left pane.
- A red box highlights the memory address `042c04` and its data in the dump.
- A red box highlights the memory address `042d90` and its data in the dump.
- A red arrow points from the `tx_payload` variable to the `042c04` memory location.
- A red arrow points from the `tx_buffer` variable to the `042d90` memory location.
- The text `tx_payload` is written in red next to the `042c04` memory location.
- The text `TX data` is written in red next to the `042d90` memory location.
- The text `tx_buffer` is written in red next to the `042d90` memory location.

At the bottom of the window, the status bar shows:

- evk device: ok
- File Path: 3359\_2\_4g\_proprietary\_sdk\GEN\_FSK\_TX\GEN\_FSK\_TX.bin
- Version : 5.4.1

From Figure 7, you can know what data you have sent, and what the RF channel you have set is.