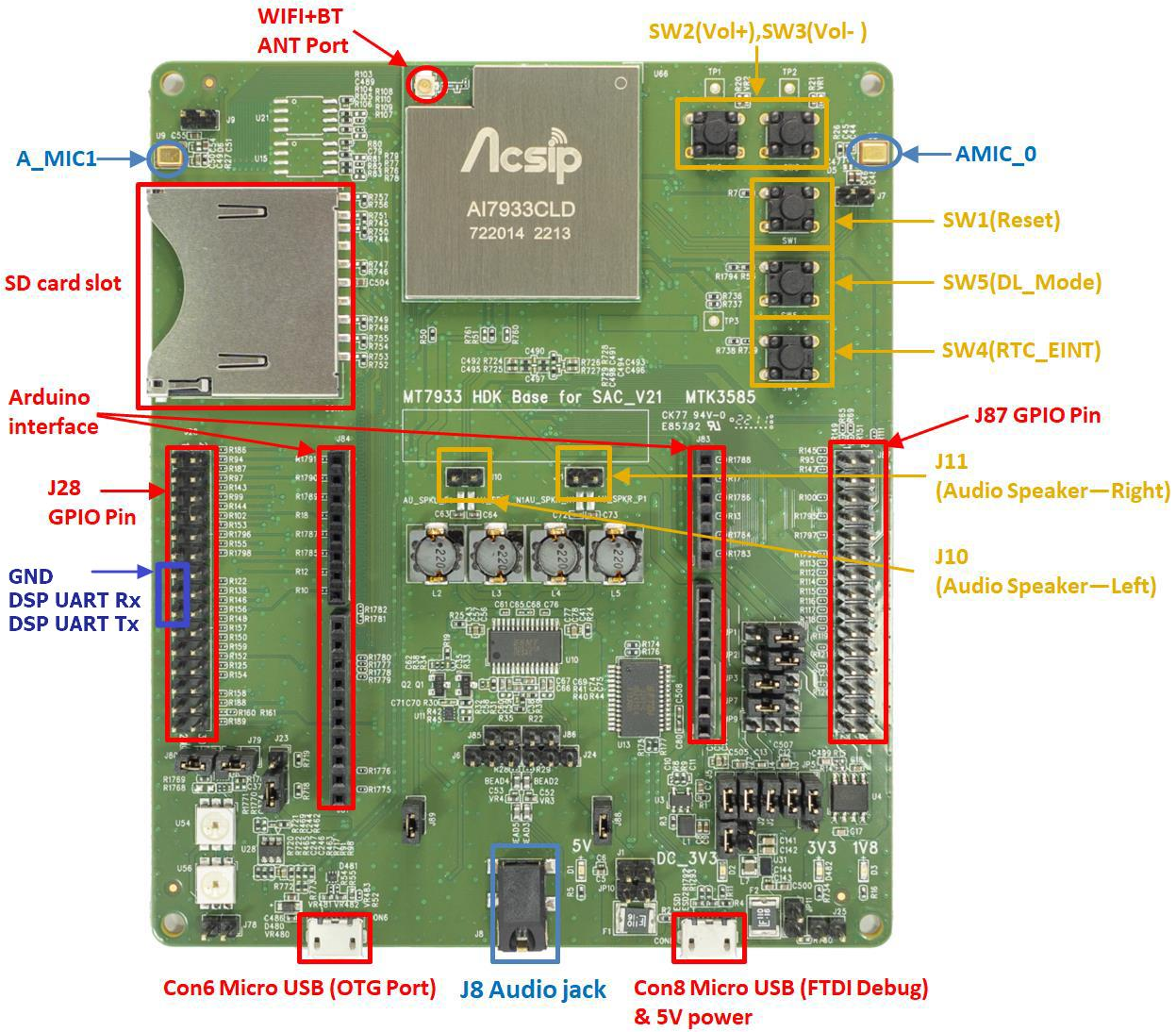
# MT7931/33 Software User's Guide

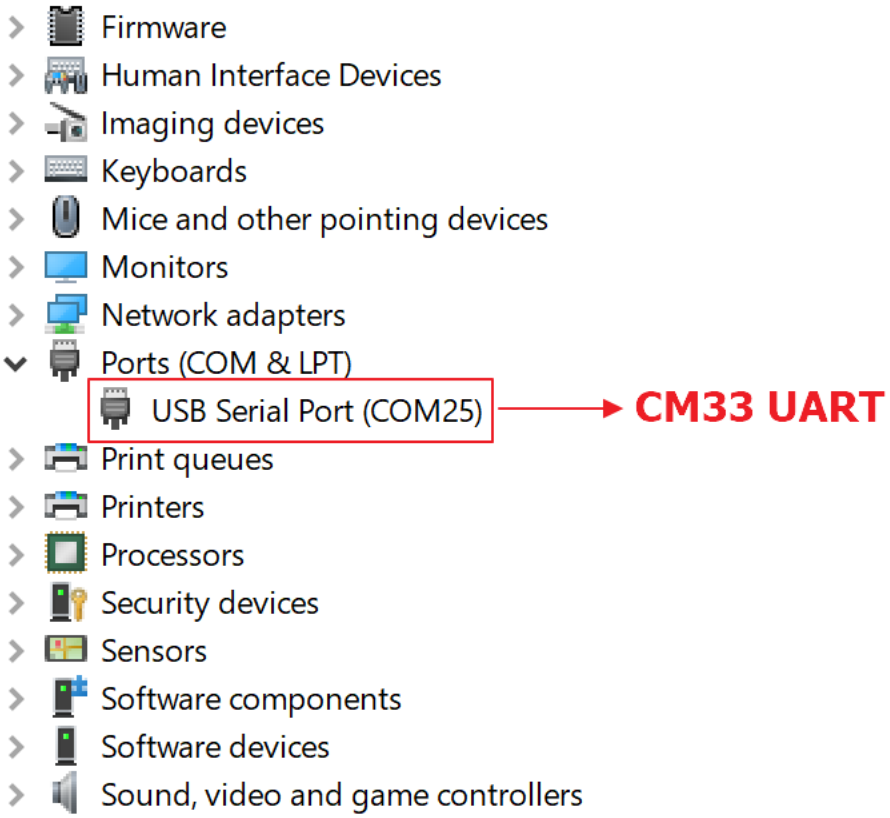

**AcSiP Technology Corp.**

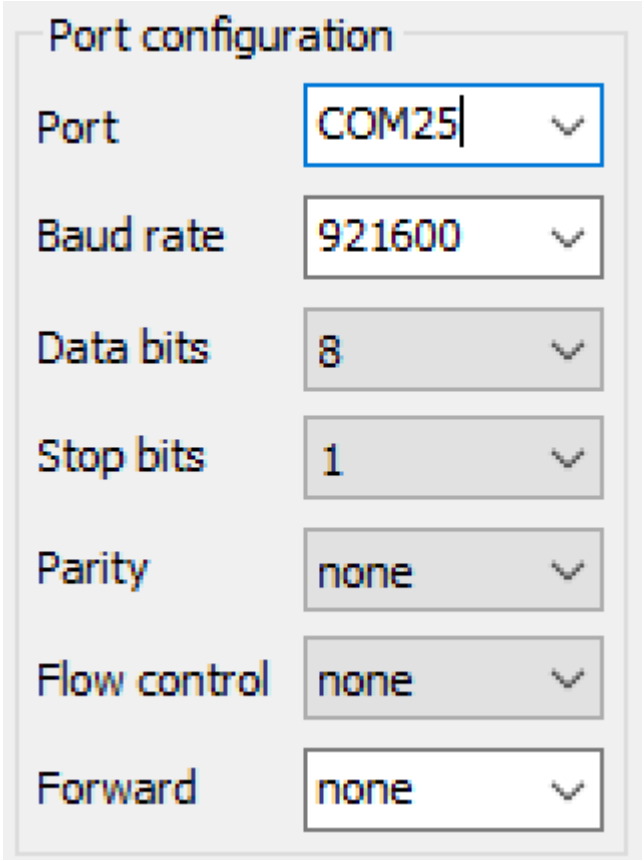
www.acsip.com.tw

# MT7931/33 HDK Configure



# Device Com Ports Configure



# SDK Build

## 1. SDK build environment :

Refer to sdk_root/doc/MT793X/03_Platform_System/
MT793X IoT SDK for Build Environment  Guide.pdf

## 2. SDK build environment virtual env installation :

Refer to sdk_root/doc/MT793X/03_Platform_System/
MT793X IoT SDK for Build Environment Virtual_Env Installation.pdf

## 3. Example project build command :

- Build qfn_sdk_demo
  ./build.sh mt7933_hdk qfn_sdk_demo -o=IMGTOOL_ENV=~/venv/imgtool/bin/activate

- Build bga_sdk_demo
  ./build.sh mt7933_hdk bga_sdk_demo -o=IMGTOOL_ENV=~/venv/imgtool/bin/activate

- Build bootloader
  ./build.sh mt7933_hdk bootloader -o=IMGTOOL_ENV=~/venv/imgtool/bin/activate

# SDK Build

- Project scatter

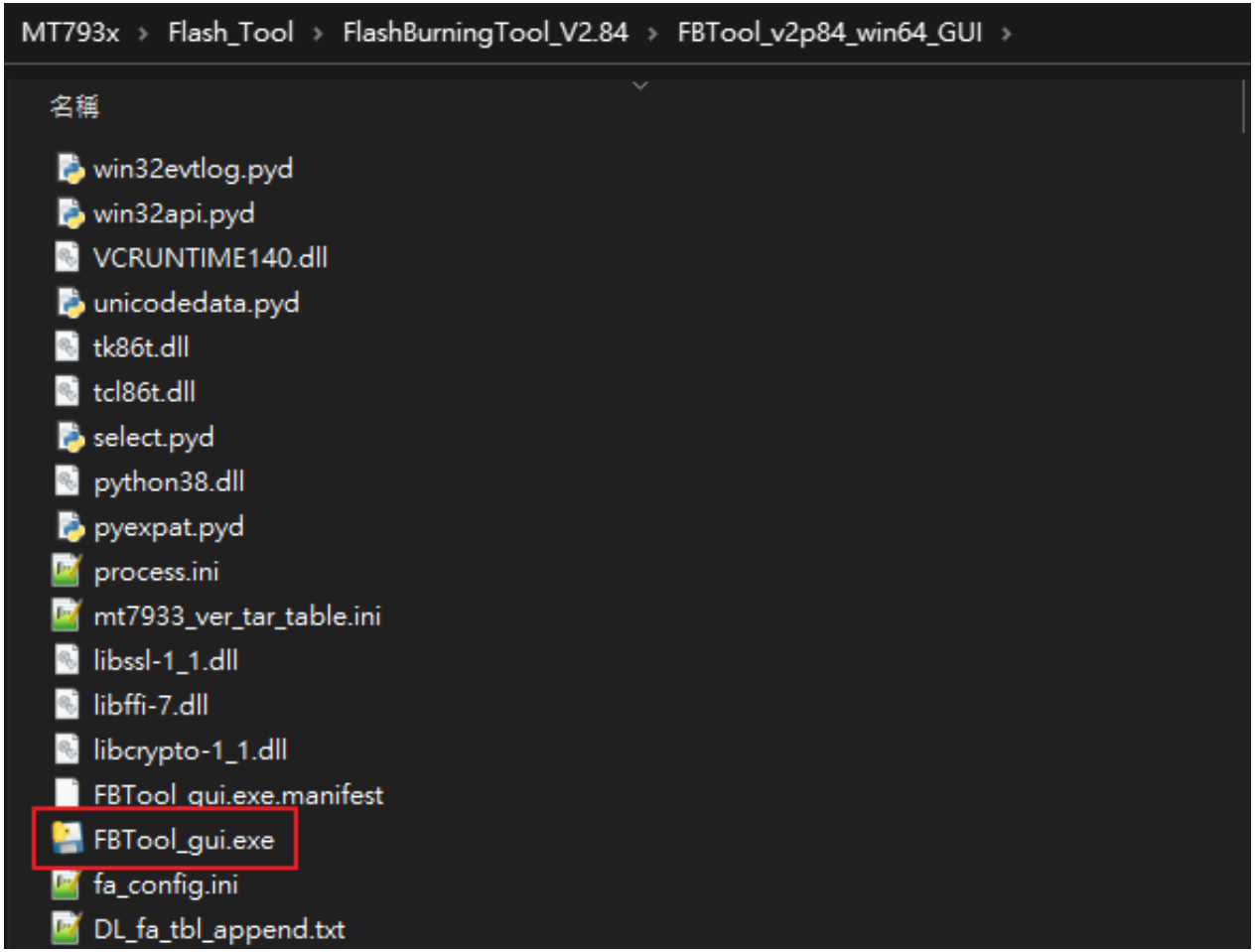
```
[ROM_BL]
enable=n
start_addr=0x00000000
partition_size=0x00010000
file_name=mt7931an_bootloader-xip.sgn
readback=n
[ROM_RTOS]
enable=y
start_addr=0x00044080
partition_size=0x002fff80
file_name=mt7933cv_xip_bga_al.bin
readback=y
[ROM_DSP]
enable=n
start_addr=0x00c41000
partition_size=0x0027f000
file_name=hifi4dsp_load.bin
readback=n
[ROM_BT]
enable=n
start_addr=0x00ec0000
partition_size=0x00091000
file_name=BT_RAM_CODE_MT7933_1_1_hdr.bin
readback=n
[ROM_WIFI_EXT]
enable=n
start_addr=0x00f60000
partition_size=0x0009f000
file_name=WIFI_RAM_CODE_MT7933_ALL.bin
readback=n
[ROM_BUFFER_BIN]
enable=n
start_addr=0x00fff000
partition_size=0x00001000
file_name=MT7933_BGA_TDD_EEPROM.bin
readback=n
```

Bootloader project

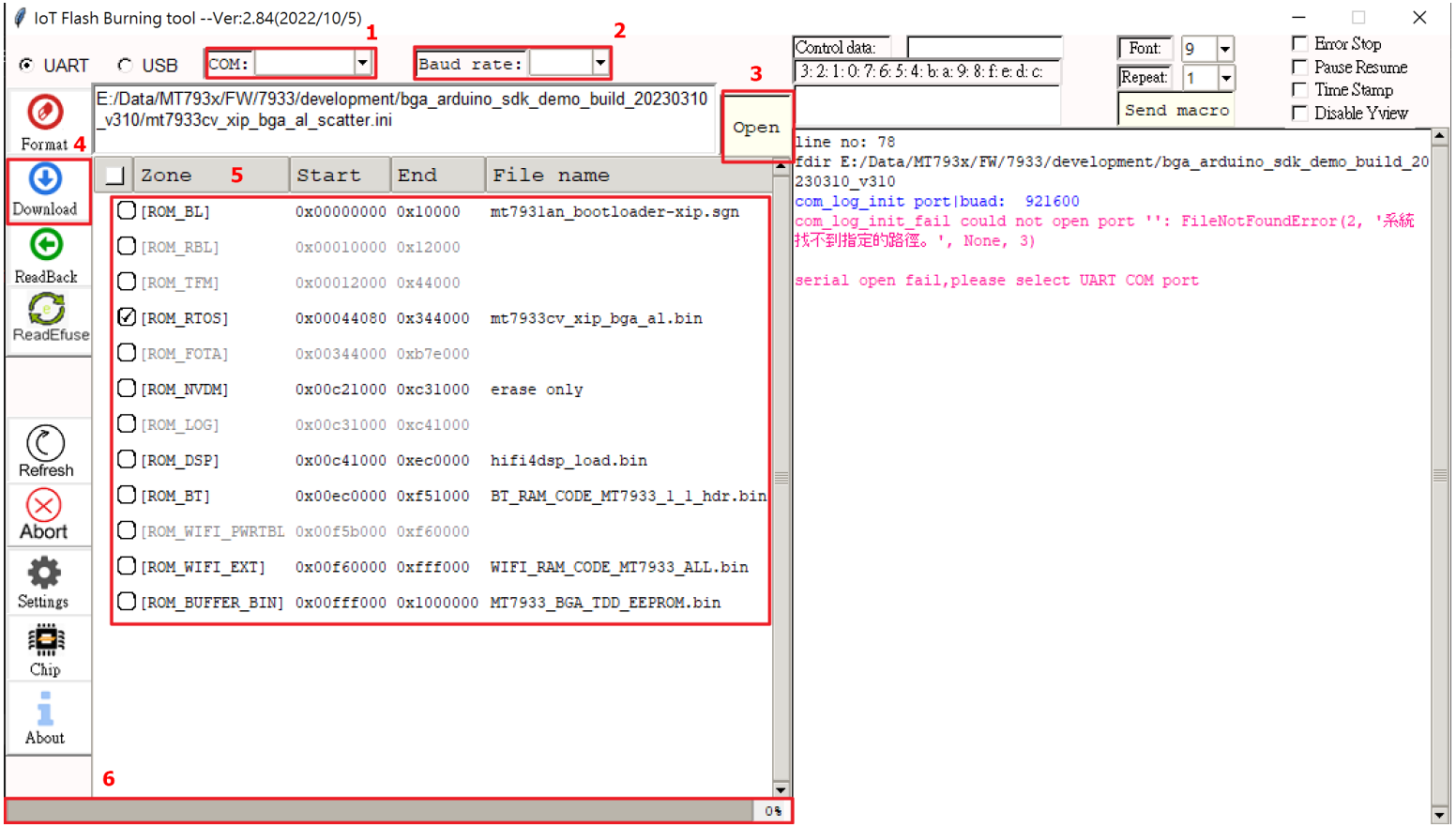
RTOS demo project
Prebuild Binary

# Flash Burning Tool

1. Open the “FBTool_gui.exe” flash burning tool (as below Figure red block) from sdk_root/tools/FlashBurningTool_V2.84/FBTool_v2p84_win64_GUI/



# Flash Burning Tool




Section 1 : Choose CM33 com port
Section 2 : Choose Baud rate 921600
Section 3 : Open scatter file
Section 4 : Start burning botton

Section 5 : Choose which file you want to burn
Section 6 : Process of burning image

# Flash Burning Tool

Step1. Select CM33 UART com port(Section 1) and Baud rate 921600(Section2).

Step2. Select the scatter file which you want to burn(Section 3).

Step3. Select zones(Section5), multiple choices of zones; only valid for Format, Download, ReadBack commands.

Step4. Click “Download” botton(Section4), keep press “SW5” and press “SW1”, waiting process bar(Section6) show yellow color(as below figure), it means handshaking successful, and release “SW5”.


Step5. Waiting for process bar(Section6) on green color keep going(as below figure).


Step6. Waiting for process bar(Section6) on green be finished to 100%(as below figure).

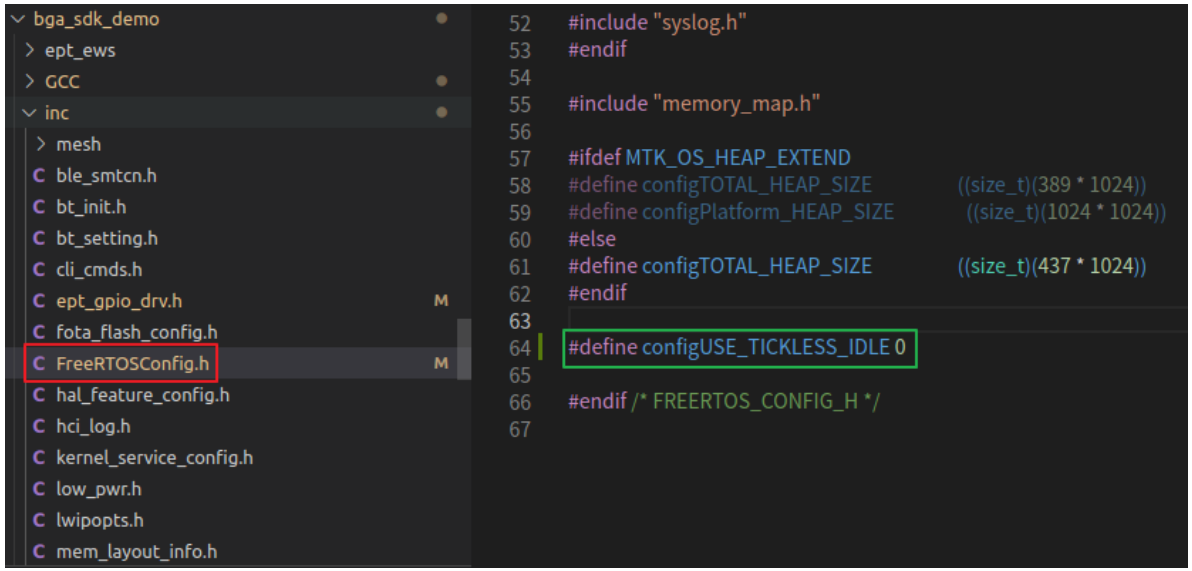

Step7. Do not power off the platform if you want to run the command again. Repeat steps 3~6.

Note : To see more information, please refer to sdk_root/tools/FlashBurningTool_V2.84/doc/ MT793X IoT SDK for Flash Burning Tool v2.84.pdf.

# Sleep Mode Switch

If you encounter the problem of uart missing characters or uart return garbled text, please follows below instruction.(Take project bga_sdk_demo for example)

- disable project bga_sdk_demo low power mode
  Add #define configUSE_TICKLESS_IDLE 0 in header file sdk_root/project/mt7933_hdk/apps/bga_sdk_demo/inc/FreeRTOSConfig.h (as below figure red and green block) and rebuild the project.



# CLI Command

Help : '?' to list commands

```
wr      - write reg
wifi    - wifi commands
iwpriv - WiFi iw command
iperf   - iperf
bt      - BT commands
en      - enter test mode
reboot - reboot
ver     - f/w ver
log     - log control
config - user config read/write/reset/show
ble     - bluetooth ble related cmd
mesh    - bluetooth mesh related cmd
picus   - bt picus command
iwpriv - WiFi iw command
wifi    - WiFi Init CLI
ping    - ping <addr> <count> <pkt_len>
iperf   - iperf
```

# CLI Command

Help : '??' to list all commands (recursive)

```
$ ??
??

mem - show memory type of <addr>

s - search <addr> <len> <pat>

d - dump memory <addr> <len>

f - fill memory

rr - read reg

wr - write reg

wifi - wifi commands

wifi on - Wifi init

wifi off - Wifi deinit

wifi info - Wifi info

wifi set_dbg - set init dbg level

wifi get_dbg - get init dbg level

wifi check_lock - check semaphore status

wifi config - wifi config

wifi config set - wifi config set

wifi config set opmode - STA/AP

wifi config set ssid - SSID

wifi config set bssid - BSSID

wifi config set sec - Security

wifi config set psk - wpa psk key
```

# CLI Command

‘en’ to Test mode
‘back’ to Normal mode

```
back    - back to normal mode
rr      - read addr
wr      - write addr
os      - os info
reboot  - reboot
ble     - bluetooth ble related cmd
picus   - bt picus command
iwpriv  - WiFi iw command
wifi    - WiFi Init CLI
lp      - sleep manager cli
lp_dvt  - Low Power DVT
iperf   - iperf
ip      - ip config
stat    - show statistics
wifitest - Wifi Test Tool
wpa_cli  - wpa_cli for wpa_supp
```

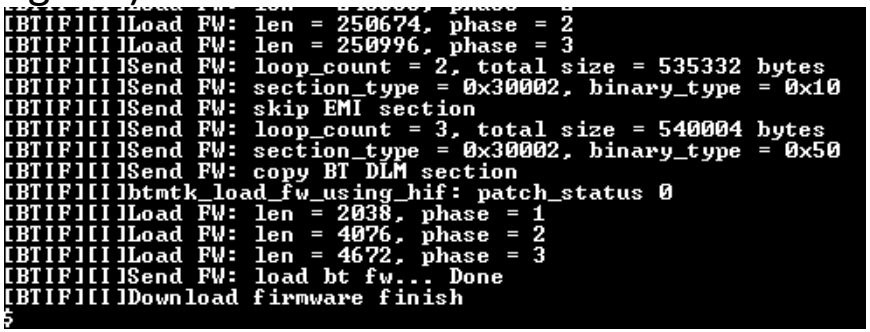
# WiFi/BT Command Reference

## 1. WiFi command reference :

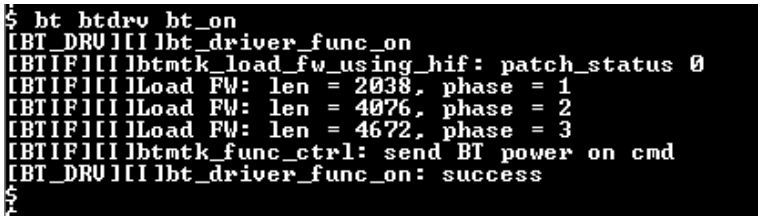
Refer to sdk_root/doc/MT793X/09_Tool/MT793X IoT SDK for Wi-Fi Test Tool.pdf

## 2. BT command reference :

- Init operation : bt btdrv dlfw (Need check “Download firmware finish”, as below figure.)




- Init operation : bt btdrv bt_on (Need check “bt_driver_func_on: success”, as below figure.)




Refer to sdk_root/doc/MT793X/09_Tool/MT793X IoT SDK for Boots User Manual.pdf

# WiFi/BT Command Reference

Note : Only MT7933 HDK supported dual mode, MT7931 HDK supported BLE only mode.

Note : SDK scatter file section [ROM_BT] file_name configuration(as below figure red block)

BT_RAM_CODE_MT7933_1_1_hdr.bin : Dual mode
BT_RAM_CODE_MT7933_2_1_hdr.bin : Ble only mode

```
[ROM_BT]
enable=y
start_addr=0x00ec0000
partition_size=0x00091000
file_name=BT_RAM_CODE_MT7933_1_1_hdr.bin
readback=y
```

# LEDs Sample CLI

Project : bga_sdk_demo

```
$ led
led
incomplete command, more options:
init   - init led spi
deinit - deinit led spi
set    - set LEDx R G B DIM
get    - get led info
on     - led on
off    - led off
```

led init : Init the LED spi

led deinit : Deinit the LED spi

led set : Set LEDs configurations

LEDx R(0~255) G(0~255) B(0~255) DIM(0~31)

ex : led set 1 255 0 0 31

led set 2 0 255 0 31

led get : Get LEDs configurations

led on : Turn on LEDs

led off : Turn off LEDs

# Barcode Scanner Sample CLI

Project : bga_sdk_demo
Scanner model : MARSON Tec. MT84G 2D Mini Scan Engine

```
$ barcode
barcode
incomplete command, more options:
init   - init barcode scanner interface
deinit - deinit barcode scanner interface
scan   - trigger scanner to scan 0:serial trigger mode, 1:hardware trigger
hid    - hid interface
uart   - uart interface
```

barcode init : Init the barcode scanner interface(HID & UART)
barcode deinit : Deinit the barcode scanner interface
barcode scan : 0 = Enable the serial trigger to scan,
1 = Hardware pin trigger

# Barcode Scanner Sample CLI

```
$ barcode hid
barcode hid
incomplete command, more options:
get     - get result
```

barcode hid get : Get HID interface decode result

```
$ barcode uart
barcode uart
incomplete command, more options:
w       - write scanner register
get     - get decode result
```

barcode uart w : Write scanner register (0xAABBBBCC)

AA = Data Length, BBBB = Register Address, CC = Data

barcode uart get : Get UART interface decode result

Note : You can find more information about MT84G in here.