

ArchMix_ Firmware Instructions

To operate MicroPython on i.MX RT1052 Development Board, the firmware of MicroPython is needed. The whole is integrated and the firmware is divided into two parts:

1. bootloader
2. firmware

bootloader is used for subsequent upgrades for the firmware. Firmware is needed to operate and run MicroPython firmware. The OTA process works the same. To start with, saving the firmware that needs to be updated locally; then restart; only when restarting will the bootloader update/expand firmware to the flash, thus overwriting the firmware.

Materials content:

The software materials include the following contents:

- ConfigTools: that is MicroPython, persimmon configuration toolkit, Net Framework 4.5 needs to be installed, which is provided in the package.
- Firmware: one firmware – Firmware
- bootloader: Bootloader
- rdb-xxx: rdb host computer
- RTT_Win7_driver: Driver package required to be installed under win7 system
- NetFramework: Net Framework 4.5 driver package

Hardware environment setup

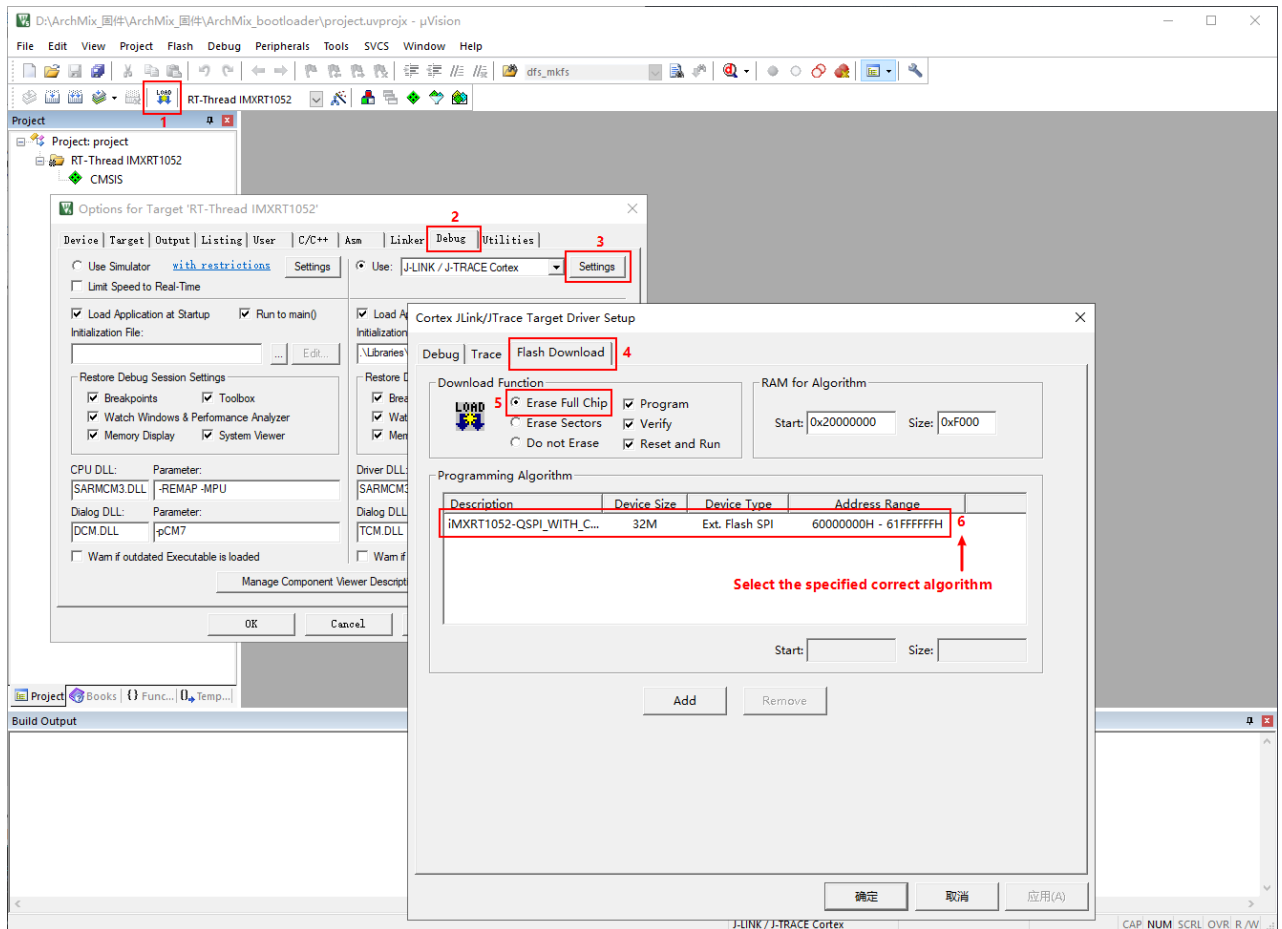
Connect the explorer to your PC using USB and connect to the emulator (by default, Jlink download). Use the USB Serial port tool to connect the GND line to the PC.

Once the Hardware environment is setup, we may begin programming the bootloader and firmware.

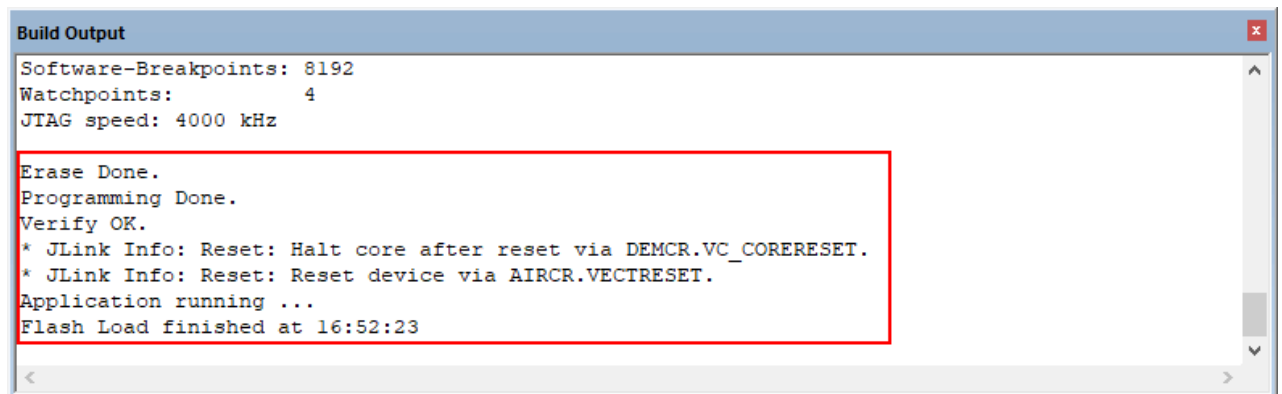
Programming bootloader

Please use the bootloader project, open it with Keil MDK, and then click the programming button (compiling is not before programming), the bootloader firmware will then be programmed to the developer flash.

1. Select the correct programming algorithm: Click `Menu Bar -> Flash->Configure Flash Tools...`; then in the pop-up box, click `Debug->Setting`; in the next pop-up box, select `Erase full chip` in the `Flash Download` column and select the specified algorithm file.



2. Start programming the project, Click **Menu Bar->Flash->Download** .
3. Verify that the project has been written in the **Build output** column.



After programming bootloader, you can open the USB serial port through putty or other serial assistant software.(Set **Connection Type** to **Serial** , set **speed** to **115200**), the serial port outputs the following information:

```

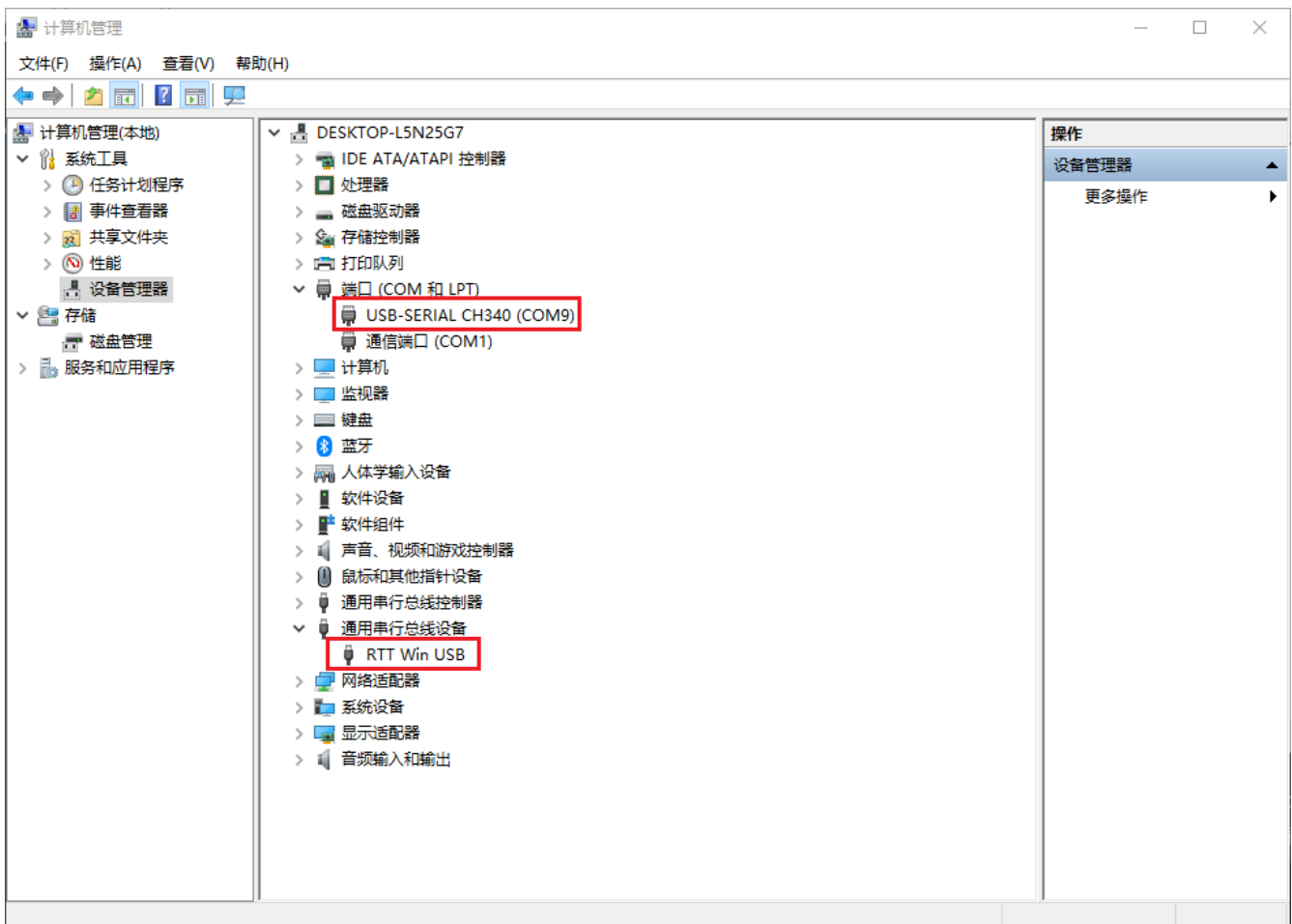
\ \ /
- RT - Thread Operating System
/ \ /
4.0.0 build Dec 21 2018
2006 - 2018 Copyright by rt-thread team
[I/[FLEXSPI]] NorFlash Init
[I/[FLEXSPI]] Vendor ID: 0x16
[I/[FLEXSPI]] NorFlash Init Done
[D/FAL] (fal_flash_init:61) Flash device | norflash0 | addr: 0x00000000 | len: 0x00800000 | blk_size: 0x00001000 | initialized finish.
[I/FAL]
----- FAL partition table -----
[I/FAL] | name | flash_dev | offset | length |
-----
[I/FAL] | b1 | norflash0 | 0x00000000 | 0x00040000 |
[I/FAL] | env | norflash0 | 0x00040000 | 0x00001000 |
[I/FAL] | pt | norflash0 | 0x00041000 | 0x00001000 |
[I/FAL] | app | norflash0 | 0x00042000 | 0x00100000 |
[I/FAL] | download | norflash0 | 0x00142000 | 0x00100000 |
[I/FAL] | fs | norflash0 | 0x00242000 | 0x00000000 |
-----
[I/FAL] RT-Thread Flash Abstraction Layer (v0.3.0) initialize success.
[I/OTA] RT-Thread OTA package(v0.2.2) initialize success.
[Flash] (packages\EasyFlash-latest\src\ef_env.c:152) ENV start address is 0x00000000, size is 262144 bytes.
[Flash] (packages\EasyFlash-latest\src\ef_env.c:821) Calculate ENV CRC32 number is 0x1DC81CF6.
[Flash] (packages\EasyFlash-latest\src\ef_env.c:833) Verify ENV CRC32 result is OK.
[Flash] EasyFlash V3.2.3 is initialize success.
[Flash] you can get the latest version on https://github.com/armink/EasyFlash .
Hit any key to stop autoboot : 0
run application
0x60042000
sdram heap, begin: 0x80000000, end: 0x81e00000
sram heap, begin: 0x200057Fc, end: 0x20020000

```

Programming firmware

When the bootloader has been burned in the development board, you can upgrade the firmware in the same manner as the OTA upgrade firmware, as follows:

To ensure that the hardware is connected to PC, open the device Manager, check if there is a corresponding serial port and `RTT Win USB`. `RTT Win USB` is drive-free in the Windows 10, but should install the specified driver package provided by RT-Thread in the Windows 7.

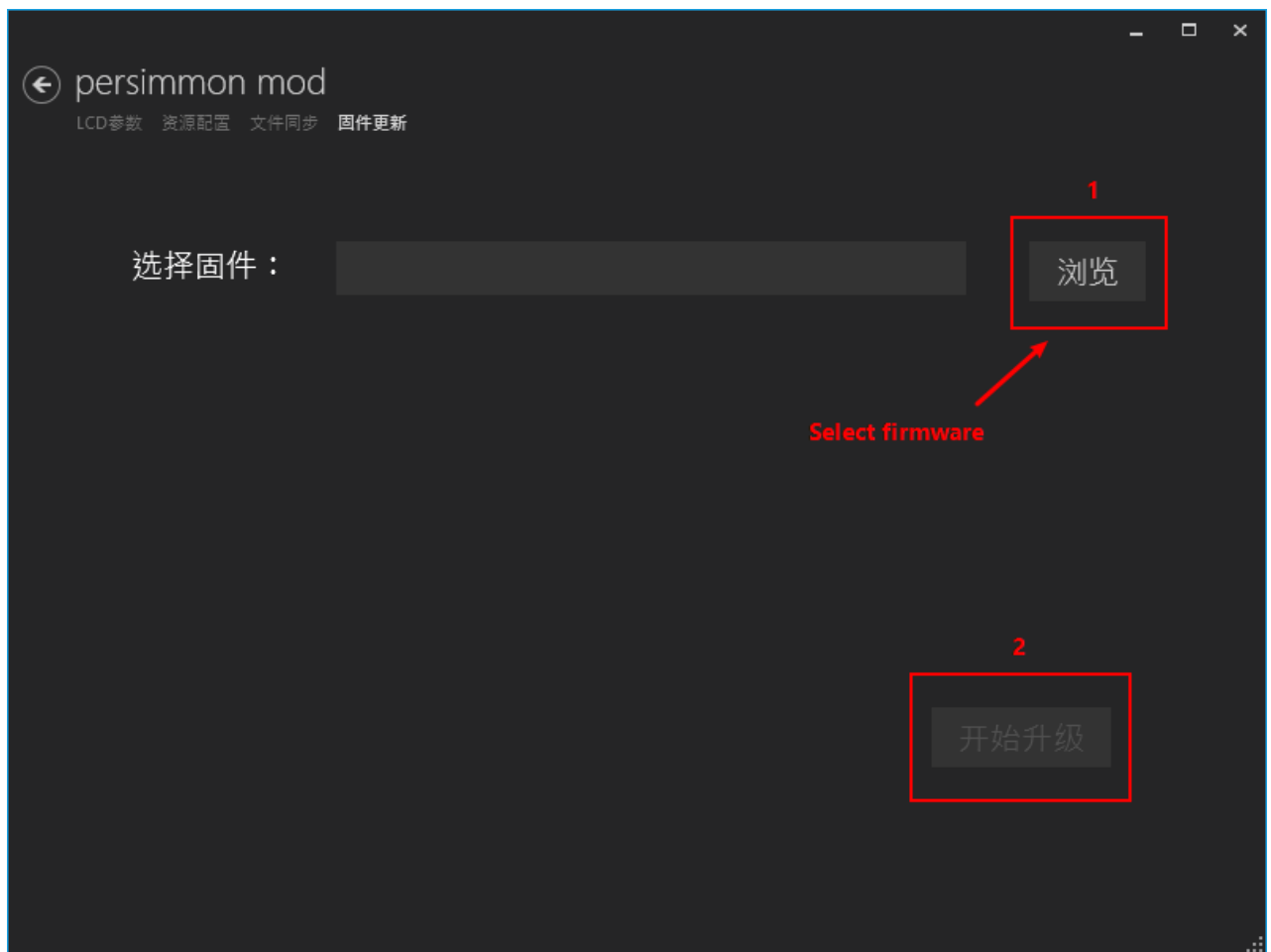


1. Restart the development board, in the serial port software will show the bootloader startup information, and information of countdown , before the countdown count jump to 0, enter any key in the serial port, bootloader will stop waiting for the command.

```
\ | /
- RT -   Thread Operating System
        4.0.0 build dec 21 2018
2006 - 2018 copyright by rt-thread team
[I/FLEXSPI] NorFlash Init
[I/FLEXSPI] Vendor ID: 0x16
[I/FLEXSPI] NorFlash Init Done
[D/FAL] (fal_flash_init:61) Flash device | norflash0 | addr: 0x00000000 | len: 0x00800000 | blk_size: 0x00001000 | initialized finish.
[I/FAL] ----- FAL partition table -----
[I/FAL] | name | Flash_dev | offset | length |
-----|-----|-----|-----|-----|
[I/FAL] | bl | norflash0 | 0x00000000 | 0x00040000 |
[I/FAL] | env | norflash0 | 0x00040000 | 0x00001000 |
[I/FAL] | pt | norflash0 | 0x00041000 | 0x00001000 |
[I/FAL] | app | norflash0 | 0x00042000 | 0x00100000 |
[I/FAL] | download | norflash0 | 0x00142000 | 0x00100000 |
[I/FAL] | fs | norflash0 | 0x00242000 | 0x00000000 |
-----|-----|-----|-----|-----|
[I/FAL] RT-Thread Flash Abstraction Layer (V0.3.0) initialize success.
[I/OTA] RT-Thread OTA package(V0.2.2) initialize success.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:152) ENV start address is 0x00000000, size is 262144 bytes.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:821) Calculate ENV CRC32 number is 0x1DC81CF6.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:833) Verify ENV CRC32 result is OK.
[Flash] EasyFlash V3.2.3 is initialize success.
[Flash] You can get the latest version on https://github.com/armink/EasyFlash .
Hit any key to stop autoboot: 1
run bootloader
msh >
msh >
```

At this point, enter any key

- 2. Open the ConfigTools, then in 固件更新, select the firmware you want to upgrade, click 开始升级, which will compile the firmware to the development board. Bootloader can be updated after automatic reboot.



- 3. The upgrade process can be seen on the serial port software as well as the progress of the firmware update.

```

\ / \
- RT - Thread Operating System
      4.0.0 build dec 21 2018
2006 - 2018 copyright by rt-thread team
[I/[FLEXSPI]] NorFlash Init
[I/[FLEXSPI]] Vendor ID: 0x16
[I/[FLEXSPI]] NorFlash Init Done
[D/FAL] (fal_flash_init:61) Flash device | norflash0 | addr: 0x00000000 | len: 0x00800000 | blk_size: 0x00001000 | initialized finish.
[I/FAL] ----- FAL partition table -----
[I/FAL] | name | flash_dev | offset | length |
-----|-----|-----|-----|-----|
[I/FAL] | b1 | norflash0 | 0x00000000 | 0x00040000 |
[I/FAL] | env | norflash0 | 0x00040000 | 0x00001000 |
[I/FAL] | pt | norflash0 | 0x00041000 | 0x00001000 |
[I/FAL] | app | norflash0 | 0x00042000 | 0x00100000 |
[I/FAL] | download | norflash0 | 0x00142000 | 0x00100000 |
[I/FAL] | fs | norflash0 | 0x00242000 | 0x00000000 |
-----|-----|-----|-----|
[I/FAL] RT-Thread Flash Abstraction Layer (V0.3.0) initialize success.
[I/OTA] RT-Thread OTA package(V0.2.2) initialize success.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:152) ENV start address is 0x00000000, size is 262144 bytes.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:821) Calculate ENV CRC32 number is 0x1DC81CF6.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:833) Verify ENV CRC32 result is OK.
[Flash] EasyFlash V3.2.3 is initialize success.
[Flash] You can get the latest version on https://github.com/armink/EasyFlash .
Hit any key to stop autoboot : 1
run bootloader
msh >
msh >Erase Down part
Start Update :280384 ← Start update
100%
Write Done
[I/OTA] Verify 'download' partition(fw ver: 1.0.0, timestamp: 1544780235) success.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:821) Calculate ENV CRC32 number is 0x04D32DB7.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:774) Erased ENV OK.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:788) Saved ENV OK.
sdram heap, begin: 0x80000000, end: 0x81e00000
sram heap, begin: 0x20006e8c, end: 0x20020000

```

```

\ / \
- RT - Thread Operating System
      4.0.0 build Dec 21 2018
2006 - 2018 Copyright by rt-thread team
[I/[FLEXSPI]] NorFlash Init
[I/[FLEXSPI]] Vendor ID: 0x16
[I/[FLEXSPI]] NorFlash Init Done
[D/FAL] (fal_flash_init:61) Flash device | norflash0 | addr: 0x00000000 | len: 0x00800000 | blk_size: 0x00001000 | initialized finish.
[I/FAL] ----- FAL partition table -----
[I/FAL] | name | flash_dev | offset | length |
-----|-----|-----|-----|
[I/FAL] | b1 | norflash0 | 0x00000000 | 0x00040000 |
[I/FAL] | env | norflash0 | 0x00040000 | 0x00001000 |
[I/FAL] | pt | norflash0 | 0x00041000 | 0x00001000 |
[I/FAL] | app | norflash0 | 0x00042000 | 0x00100000 |
[I/FAL] | download | norflash0 | 0x00142000 | 0x00100000 |
[I/FAL] | fs | norflash0 | 0x00242000 | 0x00000000 |
-----|-----|-----|-----|
[I/FAL] RT-Thread Flash Abstraction Layer (V0.3.0) initialize success.
[I/OTA] RT-Thread OTA package(V0.2.2) initialize success.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:152) ENV start address is 0x00000000, size is 262144 bytes.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:821) Calculate ENV CRC32 number is 0x04D32DB7.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:833) Verify ENV CRC32 result is OK.
[Flash] EasyFlash V3.2.3 is initialize success.
[Flash] You can get the latest version on https://github.com/armink/EasyFlash .
[I/OTA] Verify 'download' partition(fw ver: 1.0.0, timestamp: 1544780235) success.
[I/OTA] OTA firmware(app) upgrade(1.0.0->1.0.0) startup.
[I/OTA] The partition 'app' is erasing.
[I/OTA] The partition 'app' erase success.
[I/OTA] OTA write: [=====] 100%
[Flash] (packages\EasyFlash-latest/src/ef_env.c:821) Calculate ENV CRC32 number is 0x1DC81CF6.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:774) Erased ENV OK.
[Flash] (packages\EasyFlash-latest/src/ef_env.c:788) Saved ENV OK.
Hit any key to stop autoboot : 0
run application
0x60042000
sdram heap, begin: 0x80000134, end: 0x81e00000
sram heap, begin: 0x200066d4, end: 0x20020000

```

Start erasing the original APP before updating. After the update restarts can complete the burn write.

After programming firmware, RT-Thread 's logo will appear on the screen and you can start using MicroPythonon.