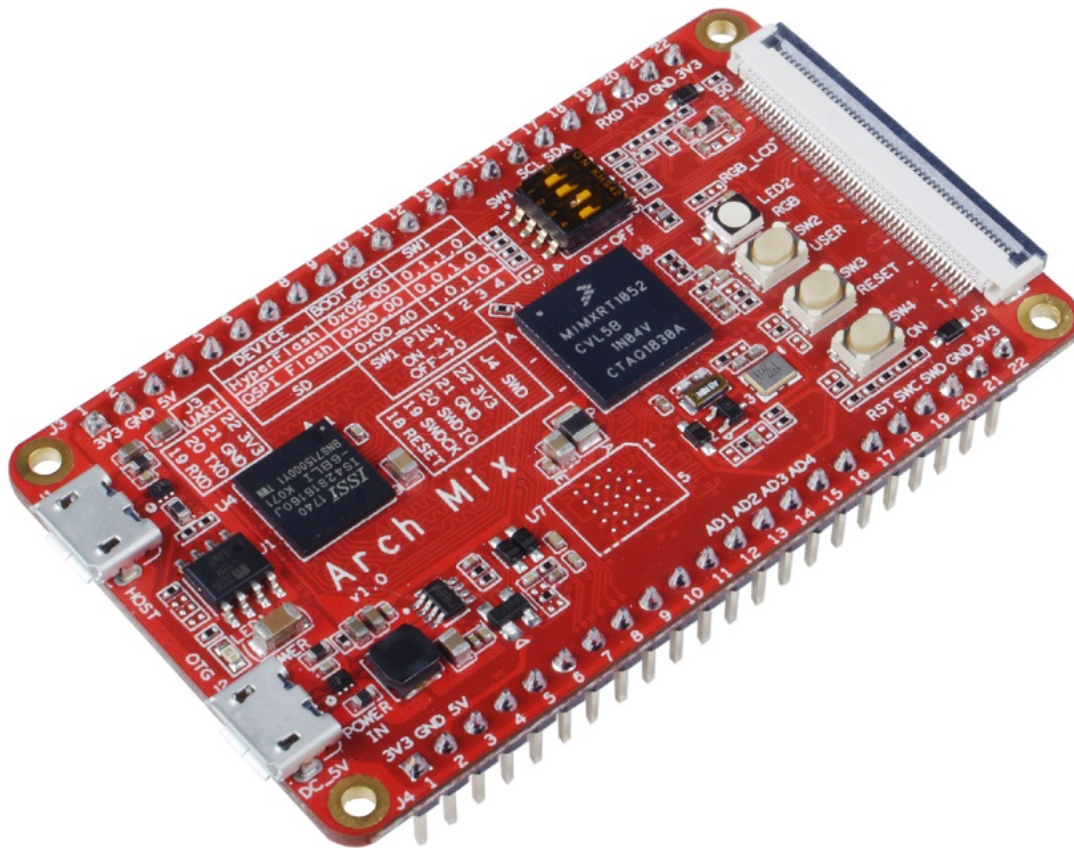


Arch Mix



Arch Mix is a thin, lightweight development board based on NXP i.MX RT1052 processor(3020 CoreMark/1284 DMIPS @ 600 MHz). This development board comes pre-installed RT-Thread real-time operating system and built-in micro-python. Which makes it suitable for industrial control, especially for scenes with large code and high real-time application requirements.

The i.MX RT1052 is a new processor family featuring NXP's advanced implementation of the Arm Cortex® -M7 core. Currently, the i.MX RT1052 is the highest performing Cortex-M7 solution delivering 3036 CoreMarks, which is 13 times better than the LPC1788 microcontroller. In addition to the high-speed performance it provides fast real-time responsiveness. The i.MX RT1050 also has rich audio and video features, including LCD display, basic 2D graphics, camera interface, SPDIF, and I2S audio interface.

The RT-Thread is an open source IoT operating system for embedded devices. The kernel has real-time multi-task scheduling, semaphore, mutex, mail box, message queue, signal etc. This is a lightweight system that loads quickly. For more detail about the RTOS, please refer to the [Github Page](#).

Get One Now 

Application Ideas

- Industrial Control
- Smart Building
- Industrial Human Machine Interfaces
- Automation & Process Control
- Robot

Feature

- ARM® Cortex®-M7 600MHz microcontroller(NXP i.MX RT1052)
- Comes with real-time operating system RT-Thread
- Build-in micro-python
- Ultra-fast system loading speed
- Rich peripheral interface: RMII, CAN, I2C, UART, CSI, I2S, ADC, SPDIF IN/OUT, SWD
- Smaller than other Demo boards of RT1052/1050: 67mm x 39mm

Specification

Parameters	Value
Processor: NXP i.MX RT1052	
Platform	ARM Cortex-M7 MPCore
Frequency	600 MHz
Boot ROM	96KB
ON-Chip RAM	512KB
Memory	
SDRAM	32MB
QSPI Flash	8MB
HyperFlash(Optional)	64MB
Connectivity	
USB 2.0 Host	x1
USB 2.0 OTG, and DC 5V Power In	x1
Boot configuration DIP switch	x1
LED	Power LED x1 User RGB LED x1
Buttons	Reset button x1, On/Off button x1, User button x1
24bit RGB LCD interface	x1
Micro SD card connector	x1
RTC 3V battery connector	x1

Parameters	Value
22Pin header	RMII, CAN, I2C, UART, CSI, I2S, ADC, SPDIF IN/OUT, SWD

Table 1.Specification

Hardware Overview

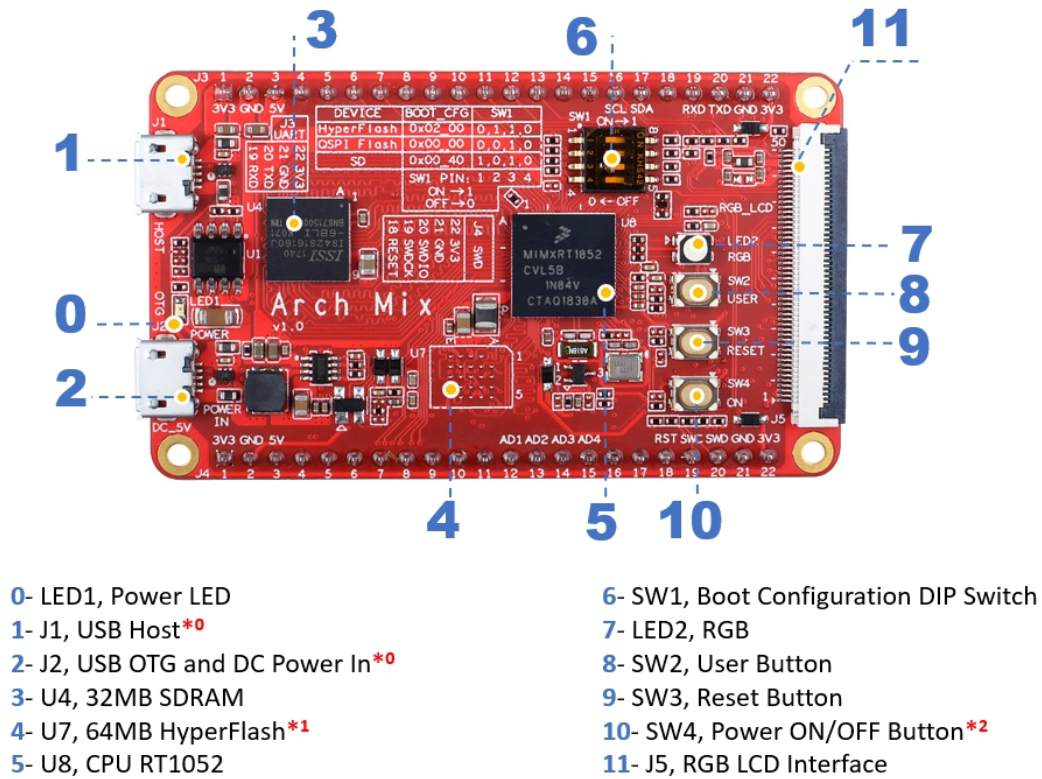


Figure 1. Front Hardware Overview

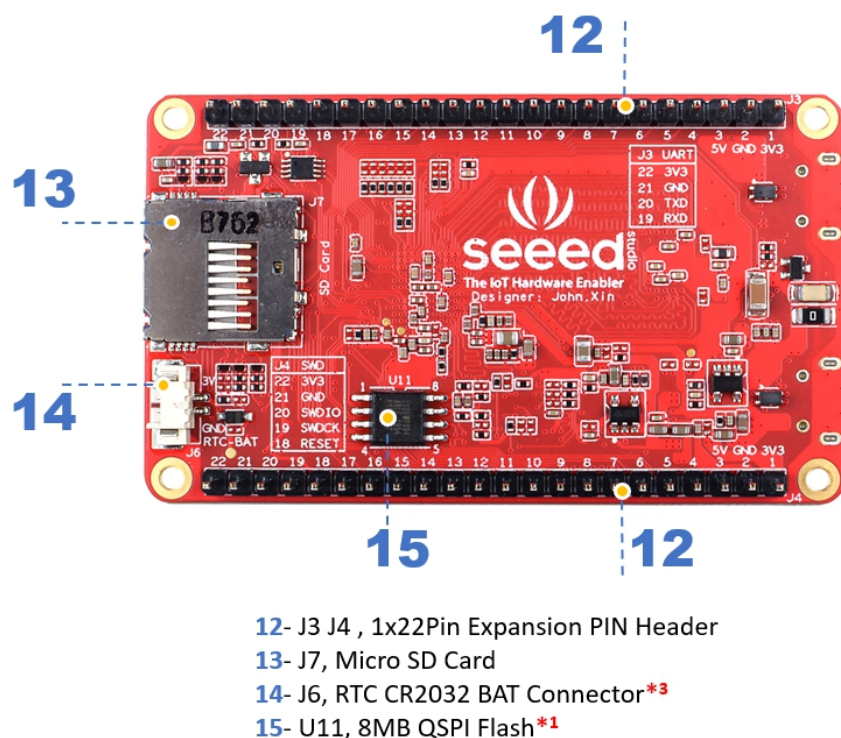


Figure 2. Back Hardware Overview

!!!Annotation *0 You need to power the Arch Mix by the USB OTG port. For the difference between the USB HOST and USB OTG, Please check [here](#).

*1 We provide two options for the flash, you can use 64M HyperFlash(U7-default DNP) or 8M QSPI Flash(U11-default selection).

*2 After the board is powered by USB OTG, you can switch the system on and off by pressing and holding(about 3~5 seconds) this button.

*3 Please note that this port is a 1.25mm CR2032 Battery port, do not plug in a Li-Po battery. If you want to use the RTC function, you can search the 'CR2032 Battery with Wire Leads' in the Amazon or other web.

Power

Please supply power through the Micro-USB **OTG** port.

!!!Danger - The input power supply voltage is 5V, can not exceed 5.5V.

- All digital and analog IO interface levels are 3.3V. Please do not input more than 3.3V, otherwise the CPU may be damaged.
- RTC's battery-powered interface(J6) can only be connected to a button battery of about 3V, and the voltage cannot exceed 3.6V.

Switch

The Arch Mix can be configured into three different boot modes: HyperFlash, QSPI Flash and SD Card. We use QSPI Flash by default, when you change the boot mode, you need to change the DIP switch to the corresponding position.

DEVICE	BOOT_CFG	SW1 four keys value
HyperFlash	0x02_00	0 , 1 , 1 , 0
QSPI Flash	0x00_00	0 , 0 , 1 , 0
SD	0x00_40	1 , 0 , 1 , 0

Table 2.BOOT Configuration

Button

There are three buttons on this board, please check the function table.

Name	Function	Detail
SW2	User Button	For user configure, for this development board No. 125 pin is SW2
SW3	RESET	System reset, when you press this button the system will restart

Name	Function	Detail
SW4	Power On/OFF	Switch the system on and off by pressing and holding(about 3~5 seconds) this button

Table 3.Button Function table

LCD Interface

As you can see, there is a 50 pin LCD Interface on this board, it Support up to 1366 x 768 WXGA resolution. In case you need a LCD screen for this board, you can use the LCD8000 serial screen. Check the links below.

[LCD from NXP](#)

[LCD from Embest](#)

Pinout

seeed The IoT Hardware Enabler		ARCH MIX	
DEFAULT_FUNCTION	GPIO	GPIO	DEFAULT_FUNCTION
DBG_SWD_3V3			DBG_UART1_3V3
GND			GND
DBG_SWD_DIO	GPIO_AD_B0_06	GPIO_AD_B0_12	DBG_UART1_TXD
DBG_SWD_CLK	GPIO_AD_B0_07	GPIO_AD_B0_13	DBG_UART1_RXD
POR_B			VDD_COIN_3V
SPDIF_IN	GPIO_AD_B1_03	GPIO_AD_B1_01	I2C1_SDA
SPDIF_OUT	GPIO_AD_B1_02	GPIO_AD_B1_00	I2C1_SCL
SAI1_TX_SYNC/AD4_X+	GPIO_AD_B1_15	GPIO_AD_B0_15	CAN2_RX
SAI1_TX_BCLK/AD3_X-	GPIO_AD_B1_14	GPIO_AD_B0_14	CAN2_TX
SAI1_TXD/AD2_Y+	GPIO_AD_B1_13	GPIO_B1_10	ENET_TX_CLK
SAI1_RXD/AD1_Y-	GPIO_AD_B1_12	GPIO_B1_09	ENET_TXEN
SAI1_RX_BCLK	GPIO_AD_B1_11	GPIO_B1_07	ENET_TXD0
SAI1_RX_SYNC	GPIO_AD_B1_10	GPIO_B1_08	ENET_TXD1
SAI1_MCLK	GPIO_AD_B1_09	GPIO_B1_04	ENET_RXD0
GPIO_AD_B1_08	GPIO_AD_B1_08	GPIO_B1_05	ENET_RXD1
GPIO_AD_B1_07	GPIO_AD_B1_07	GPIO_EMC_40	ENET_MDC
GPIO_AD_B1_06	GPIO_AD_B1_06	GPIO_EMC_41	ENET_MDIO
GPIO_AD_B1_05	GPIO_AD_B1_05	GPIO_B1_06	ENET_CR5_DV
GPIO_AD_B1_04	GPIO_AD_B1_04	GPIO_B1_11	ENET_RXER
5V_OTG			5V_OTG
GND			GND
3V3			3V3

Figure 3. Pinout, click the image to view the original file

!!!Tip Most of the pins of NXP i.MX RT1050 processor have multiplexing function, you can click the attachment below to view the detailed pin multiplexing.

[Arch Mix Pin Definition Table](#)

Blcok Diagram

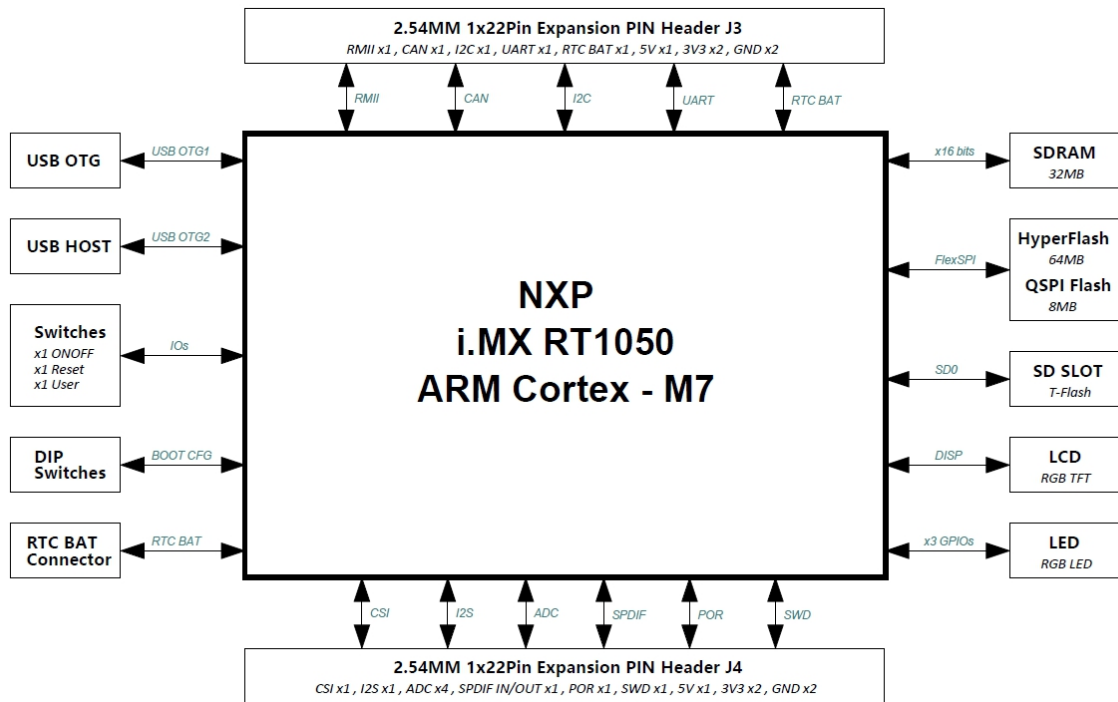
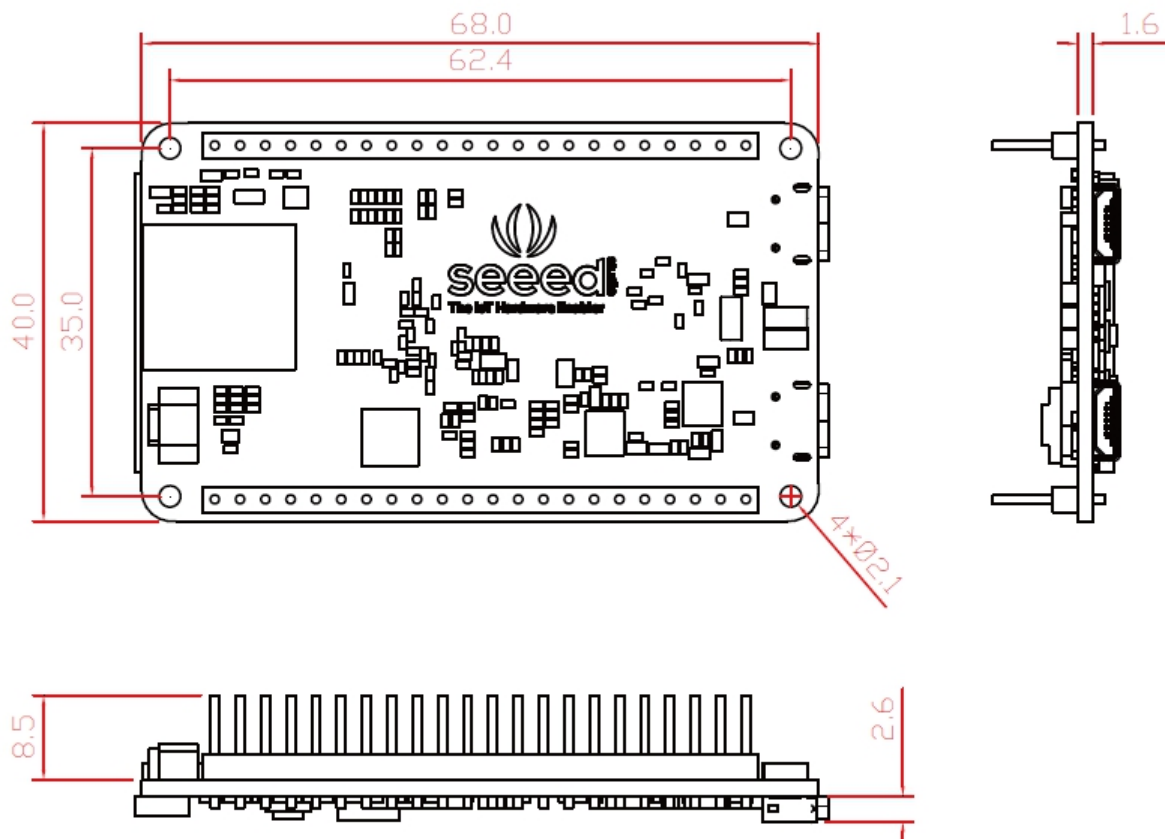


Figure 4. Arch Mix Block Diagram, [click the image to view the original file](#)

Dimension Diagram



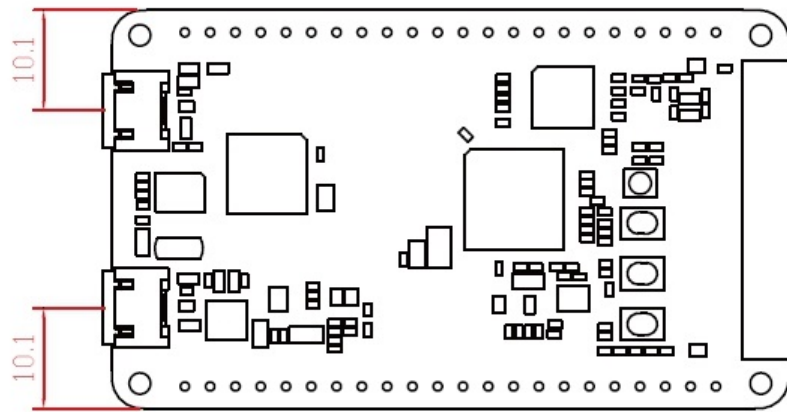


Figure 5. Board Dimension, Unit(mm)

Hardware Connection

Materials required

Arch Mix x1

USB to Serial (TTL) Module&Adapter x1

Micro USB Cable X1

Dual-female jumper x4

- **Step 1.** Connect Arch Mix and USB to Serial (TTL) Module&Adapter using the Dual-female jumper.

Module	PIN Connection			
Arch Mix	VCC	GND	TXD	RXD
USB to Serial Module	VCC	GND	RX	TX

Table 4.UART connection

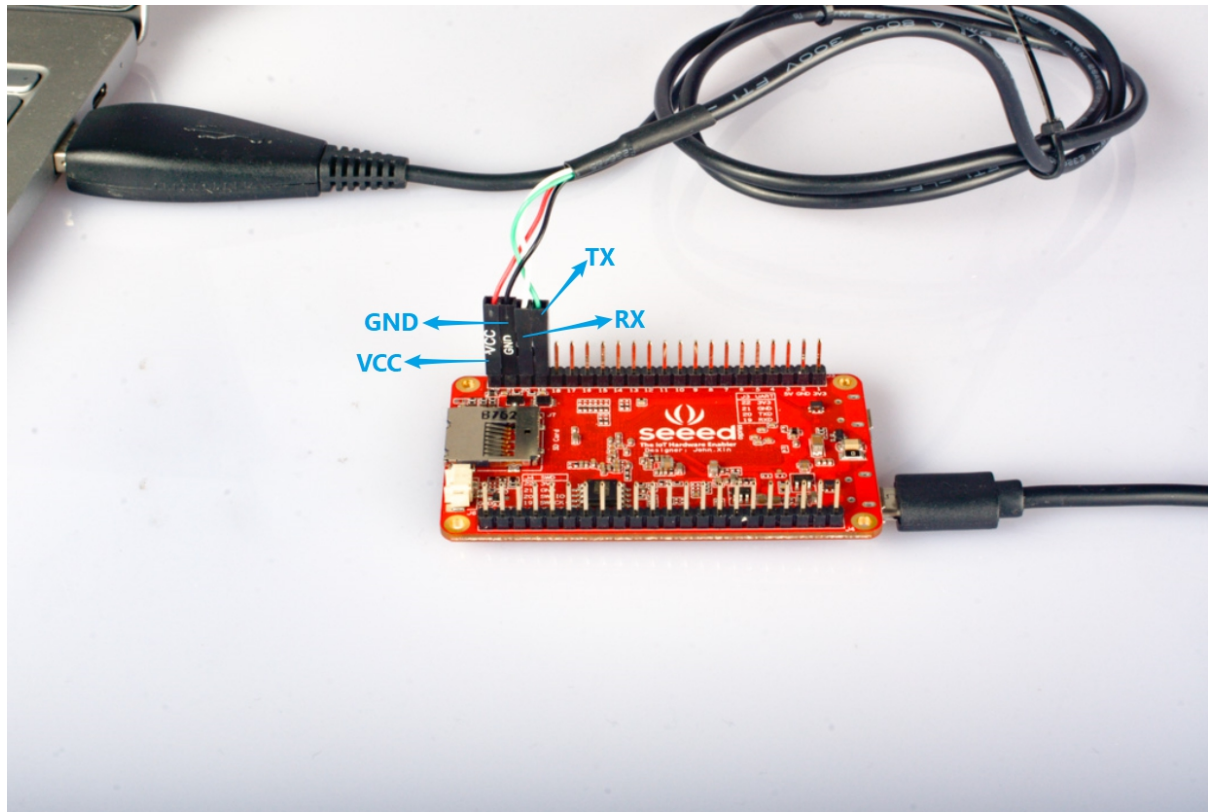


Figure 6. *UART Connection*

- **Step 2.** Plug the USB to Serial Module to your computer.
- **Step 3.** Power Arch Mix through the OTG port. The on-board Power LED will light up and RGB LED will turn green.

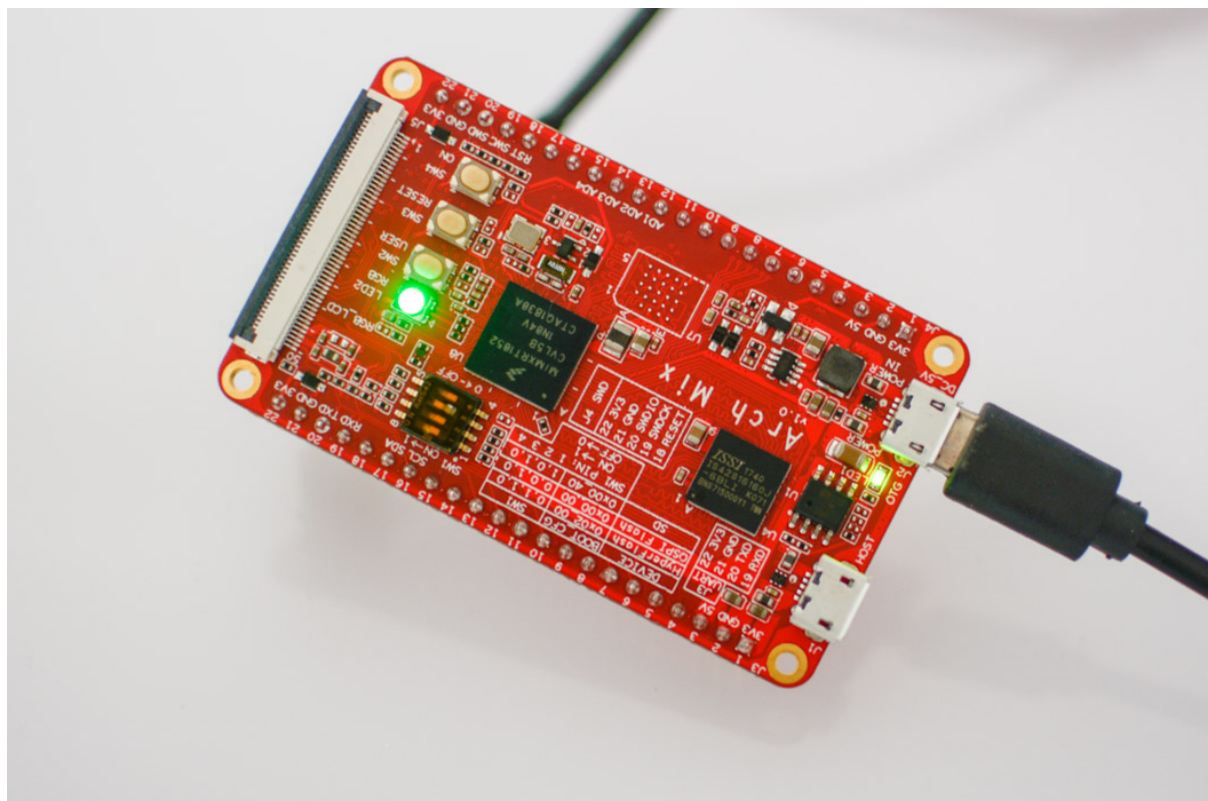


Figure 7. *Power On*

- **Step 4.** Open your **Computer Management**, find **Device Manager**. You will see The **RT-Thread Debug Bridge** and the corresponding COM port, keep in mind the COM port number. As you can see, this tutorial is COM8.

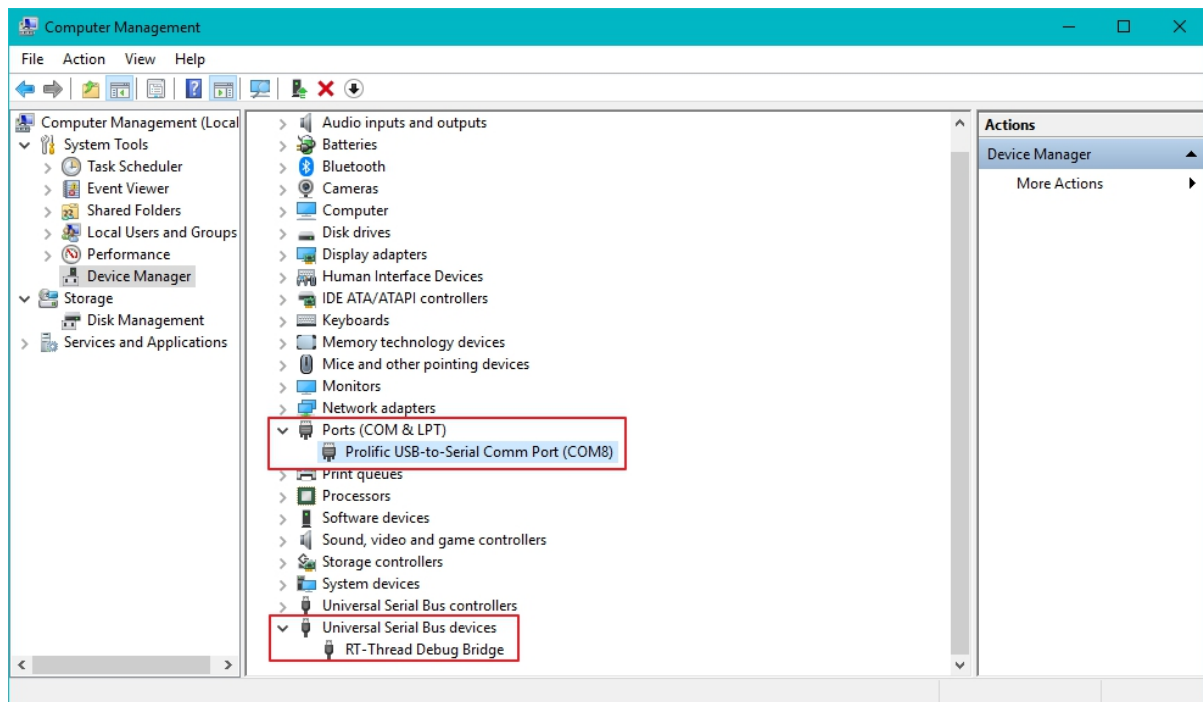


Figure 8. Check the COM port, ckick the image to view the original file

- **Step 5.** Use the serial port tool (For example: [Putty](#)) to read the serial port data. Select the corresponding port, set the baud rate to **115200**.

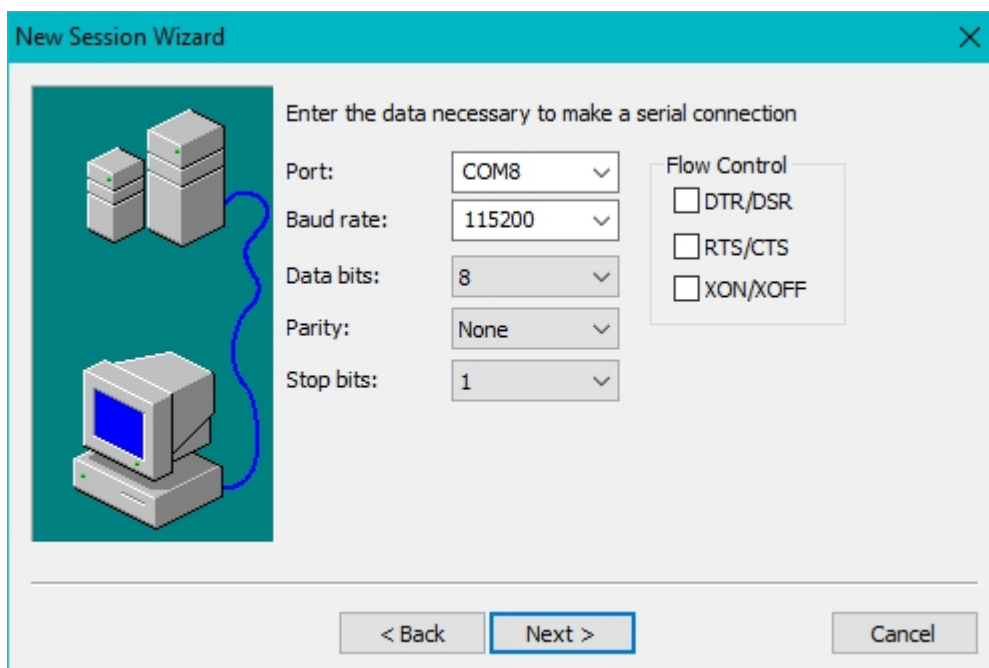


Figure 9. Configure the serial tool

- **Step 6.** Press the **Reset** button, to refresh the serial output.

RT-Thread

About RT-Thread

The RT-Thread was born in 2006, the license is similar to FreeRTOS and is released in an open source, free way. Unlike FreeRTOS and uC/OS, RT-Thread was released with a middleware platform that included components such as the network, file system, and GUI interface. After a short transition period, the Cortex M MCU was supported in 2009 and received a lot of developer recognition and support. After 2011, due to its mature and stable components, it is widely used in industrial control, electric power, new energy, high-speed rail, medical equipment, water conservancy, consumer electronics and other industries. We've made a comparison table for those three RTOS.

Item	FreeRTOS	μC/OS	RT-Thread
Kernel size	5KB ROM, 2KB RAM	6KB ROM, 1KB RAM	3KB ROM, 1KB RAM
Kernel mechanism	Mailbox ✗ Event ✓ Coroutine ✓	Mailbox ✓ Event ✓	Mailbox ✓ Event ✓ Message queue ✓
Development tools	Support a variety of mainstream tools, full toolchain	Support a variety of mainstream tools, full toolchain	Support a variety of mainstream tools, full toolchain, provide accessibility tools
Debug tools	Shell SystemView	SystemView	Shell Logging system NetUtils ADB SystemView
Testing system	Don't support	Don't support	Unit test framework Auto test system
Support chip and CPU architecture	Support ARM, MIPS, RISC-V, xtensa and other mainstream CPU architecture	Support ARM, MIPS and other mainstream CPU architecture	Support ARM, MIPS, RISC-V and other mainstream CPU architecture
File system	Support FAT	Need authorization	Provide various file systems layer. Support fatfs, littlefs, jffs2, romfs and the popular file systems.
Low power consumption	Partial support	Partial support	Support
GUI	None	μC/GUI	Provide GUI engine
Component ecology	Provide network, debugging, security related components	There are some, but need to be authorized	Provide a software package platform, currently about 100 components, covering a wide range

Item	FreeRTOS	µC/OS	RT-Thread
IoT component	TCP/UDP/AWS	Need authorization	TCP/UDP, Azure, Ayla, Aliyun, onenet, webclient, mqtt, websocket, WebNet...

Table 5. Technical and ecological comparison of three embedded operating systems

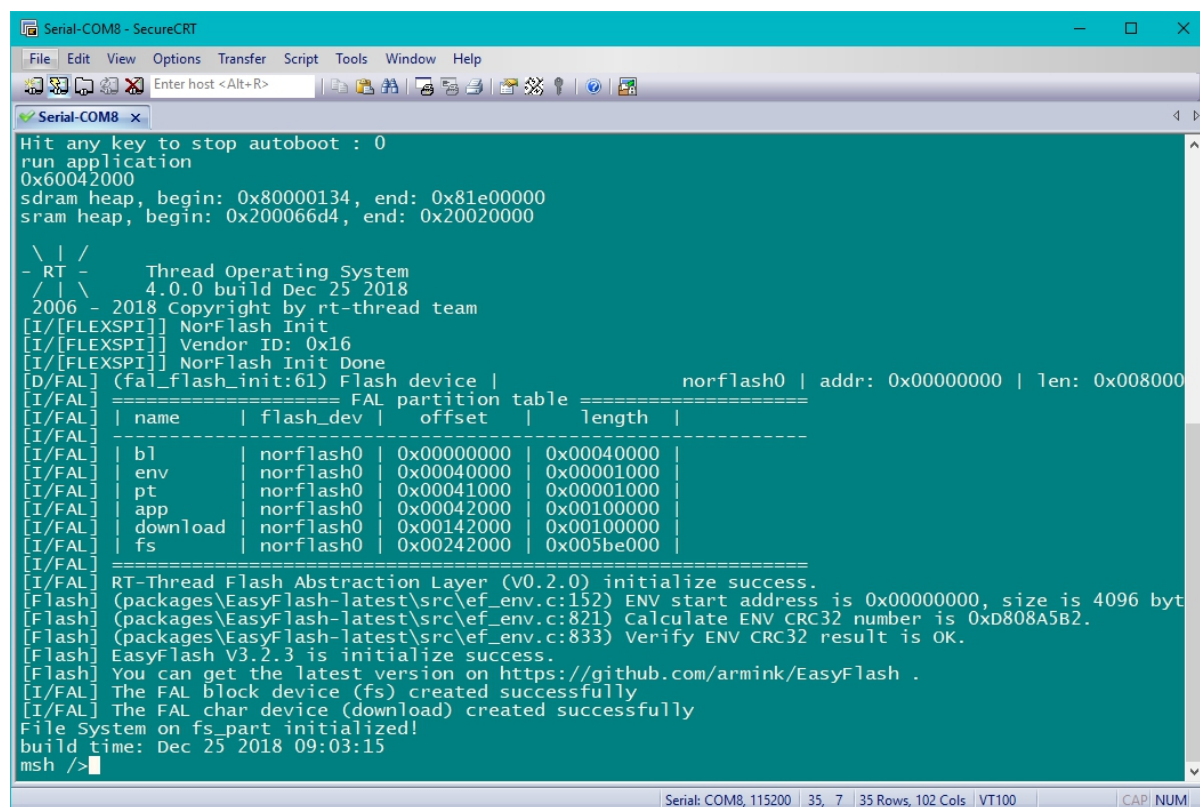
RT-Thread Resource and Manual

Thanks the RT-Thread official team, they will continue to translate relevant resources and documents, we will publish the latest English documents here.

- [RT-Thread Programming-Manual](#)

Running MicroPython

This development board comes pre-installed RT-Thread real-time operating system(RTOS) and built-in micro-python, so when you follow the steps above to connect the hardware and power on the system, you will see the system log. The RTOS is a lightweight system which loads very quickly, one or two seconds later, the system starts up and you will see the following interface:



```

Serial-COM8 - SecureCRT
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+R>
Serial-COM8 x
Hit any key to stop autoboot: 0
run application
0x60042000
sdram heap, begin: 0x80000134, end: 0x81e00000
sram heap, begin: 0x200066d4, end: 0x20020000

\ | /
- RT -      Thread Operating System
/ | \      4.0.0 build Dec 25 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: 0x008000
[I/FAL] ===== FAL partition table =====
[I/FAL] | name | flash_dev | offset | length |
[I/FAL] |-----|-----|-----|-----|
[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 | 0x005be000 |
[I/FAL] =====
[I/FAL] RT-Thread Flash Abstraction Layer (V0.2.0) initialize success.
[Flash] (packages\EasyFlash-latest\src\ef_env.c:152) ENV start address is 0x00000000, size is 4096 byt
[Flash] (packages\EasyFlash-latest\src\ef_env.c:821) Calculate ENV CRC32 number is 0xd808a5b2.
[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/FAL] The FAL block device (fs) created successfully
[I/FAL] The FAL char device (download) created successfully
File System on fs_part initialized!
build time: Dec 25 2018 09:03:15
msh />

```

Figure 10. RTOS Startup Interface

Enter **python** in the Finsh/MSH command line to enter MicroPython's interactive command line -- REPL(Read-Evaluate-Print-Loop). You can see the following interface on the terminal:

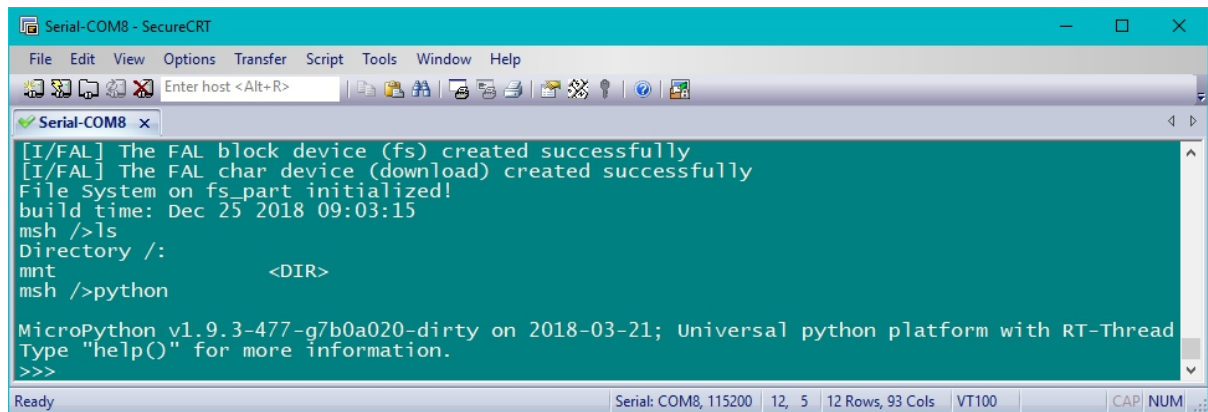


Figure 11. Enter the REPL(Read-Evaluate-Print-Loop)

You can tap ++ctrl+d++ or input `quit()` or `exit()` to exit REPL and return to RT-Thread Finsh/MSH.

Paste Mode

MicroPython is a lean and efficient implementation of the Python 3 programming language that includes a small subset of the Python standard library and is optimised to run on microcontrollers and in constrained environments.

- MicroPython has a special paste mode than the normal python interactive environment, which can be used to paste multiple lines of python code at a time.
- At the command line prompt, press ++ctrl+e++, and the prompt will appear: paste mode;
- ++ctrl+c++ to cancel, ++ctrl+d++ to finish. After pasting the code you need to run, press ++ctrl+d++ to exit the paste mode and the code you enter will be executed automatically.

```
msh />python
MicroPython v1.9.3-477-g7b0a020-dirty on 2018-03-21; Universal python platform with RT-Thread
Type "help()" for more information.
>>>
paste mode; Ctrl-C to cancel, Ctrl-D to finish
===
```

Figure 12. Paste Mode

MicroPython Demo

Flashing Light

As you can see there is a RGB LED on this board, usually this LED shows green. This demo will show you how to control the RGB LED.

!!!Note The RGB LED connected to the No. 52 pin of RT1052 chip.

- You can press ++ctrl+e++ to enter paste mode.
- Then paste the following code block into the command line.
- Press the ++ctrl+d++ to exit the paste mode and the code you enter will be executed automatically

```
import time
from machine import Pin
```

```
LED = Pin("LED1", 52), Pin.OUT_PP) #Set pin 52 to output mode
while True:
    LED.value(1)
    time.sleep_ms(500)
    LED.value(0)
    time.sleep_ms(500)
```

Now you will see the RGB LED flashing.

Button Light

Beside the RGB LED, you can find a USER button, this demo will show you how to use the USER button to control the RGB LED.

!!!Note - The RGB LED connected to the No. 52 pin of RT1052 chip.

- The button connected to the No. 152 pin of RT1052 chip.

- You can press ++ctrl+e++ to enter paste mode.
- Then paste the following code block into the command line.
- Press the ++ctrl+d++ to exit the paste mode and the code you enter will be executed automatically

```
from machine import Pin

led = Pin("LED1", 52), Pin.OUT_PP
key = Pin("KEY", 125), Pin.IN, Pin.PULL_UP) #Set pin 125 to pull-up input mode
while True:
    if key.value():
        led.value(0)
    else:
        led.value(1)
```

Now the code is running, the RGB LED will turn yellow, and when you press and hold the USER button, the RGB LED will turn green.

Firmware upgrade

This Arch Mix comes pre-installed RT-Thread real-time operating system(RTOS) and built-in micro-python. In case you need to burn the firmware or upgrade the firmware, you can refer the Guide and download the tools.

[Arch Mix Firmware Guide](#)

[Tools](#)

Resources

- [\[ZIP\] Firmware and Tools](#)
- [\[PDF\] PDF Format Wiki](#)
- [\[PDF\] i.MX RT1050 Datasheet](#)

- **[PDF]** [Dimension Diagram](#)
- **[xlsx]** [Arch Mix_v1.0_Pin Function](#)

Tech Support

Please submit any technical issue into our [forum](#) or drop mail to techsupport@seeed.cc.