MaaXBoard Mini Linux Yocto User Manual V2.0

Regulatory Compliance:

- MaaXBoard Mini single board computer has passed the CE, FCC & SRRC certification.

Revision History

Rev.	Description	Author	Date
V1.0	Initial version	Sandy	20200903
V1.1	Updated to Yocto 3.0	Nick	20210322
V2.0	Updated Yocto to Langdale(4.1), BSP_VERSION to lf-6.1.1-1.0.0, Converts the file format to markdown	Lily	20230714

Catalog

Chapter 1 Introduction

1.1 Target Board: MaaXBoard Mini

The MaaXBoard Mini is a development board developed by Avnet, Which is a low-cost, NXP i.MX 8M Mini processor-based, single-board computer ideal for embedded computing and smart edge IoT applications. The i.MX 8M Mini family of application processors are based on the Arm® Cortex®-A53 and Cortex-M4 cores, which provide industry-leading audio, voice and video processing for applications that scale from consumer home audio to industrial building automation and embedded computers. The MaaXBoard Mini is production ready, FCC, CE and RoHS certified. It is available in quantities from one to thousands.

The MaaXBoard Mini contains everything necessary to support and create Linux, Android or other OS-based systems. The platform offers several onboard peripherals including 2GB of DDR4 memory, a gigabit Ethernet port, quad USB 2.0 host ports, MIPI-DSI, MIPI-CSI, WiFi, Bluetooth low energy and a MicroSD card slot. A Raspberry Pi HAT compatible expansion connector also provides interfaces for UART, SPI, I2C and GPIO. These combined capabilities make it an ideal platform for investigating AI, IoT, Industrial and multimedia applications.

1.2 Introduction

This document provides a guide to prepare MaaXBoard Mini to boot up with the Verified Linux Package and introduces how to use the functions of MaaXBoard Mini.

1.3 Feature List

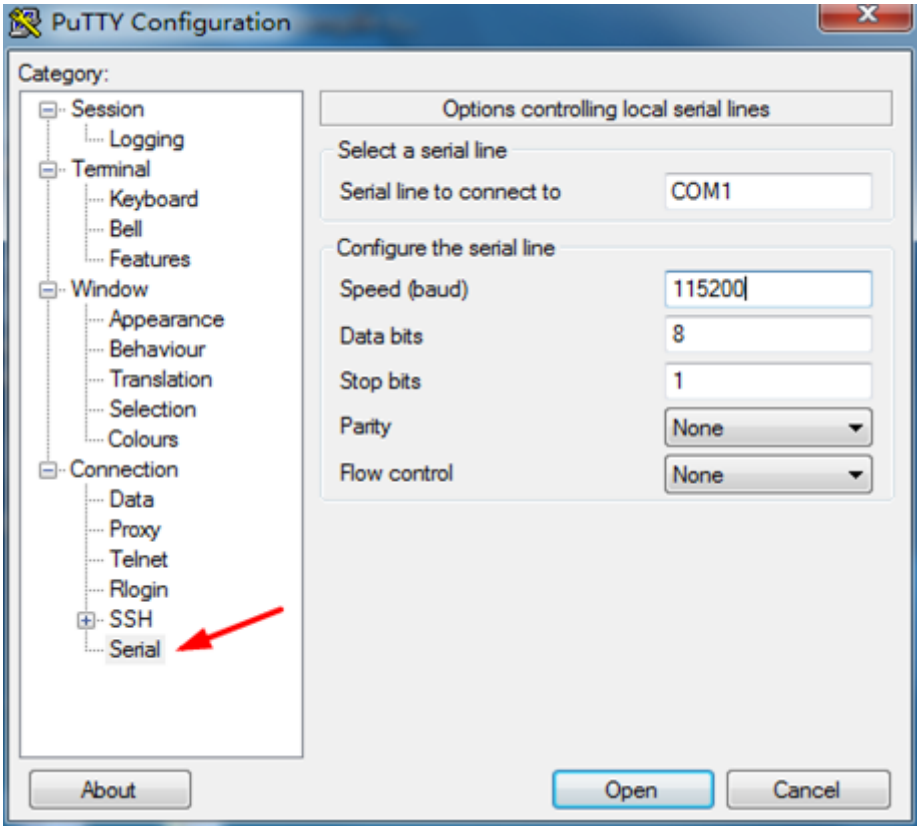
- U-Boot version: 2022.04
- Kernel version: 6.1.1
- Evaluation image: Yocto Langdale(4.1)
- Development based on NXP i.MX 8M Mini
- Micro SD boot / eMMC boot
- Device-tree Overlay support
- Desktop (Weston 11.0)
- Support QT 6.4
- 1 Gigabit Ethernet (RJ45)
- 4 USB 2.0 can work in host & device mode
- 2 UART (TTL) include debug port
- External interfaces(I2C, UART,SPI ,PWM and GPIO)
- WIFI & BLE 4.1
- MIPI-DSI display
- MIPI-CSI Camera/USB Camera

Chapter 2 Quick Start

The default version of MaaXBoard Mini supports SD Card. Avnet also provides eMMC version for users to customize. To program the image into SD Card or eMMC, refer to Chapter 4 Program or update the system Image. For the hardware connection and accessories details, please check the QSG.

2.1 Boot from SDCard

- Install the Serial Communication software (e.g. PuTTY), select the corresponding port number, baud rate as 115200, data bits as 8, stop bits as 1, parity as none.



- Connect the debug interface to PC with USB to TTL converter. Pin 6, 8 and 10 of J1 to the GND, RXD and TXD pin of the USB to TTL converter.
- Insert the SD card (with pre-programmed image) into the card slot J9.
- Powered the board with a 5V, 2A, Type-C interface power (to J11).
- When the system boot up, the serial terminal will print the following information:

```
NXP i.MX Release Distro 6.1-langdale maaxboardmini ttymxc0

maaxboardmini login:
```

- Enter username as "**root**" to login.

```
maaxboardmini login: root
root@maaxboardmini:~#
```

- Users could also connect keyboard and mouse to MaaXBoard Mini to login Yocto system.

2.2 Boot from eMMC

If you are using the eMMC version, the boot process is the same with boot from SD card, just ignore insert SD Card step.

2.3 Login system

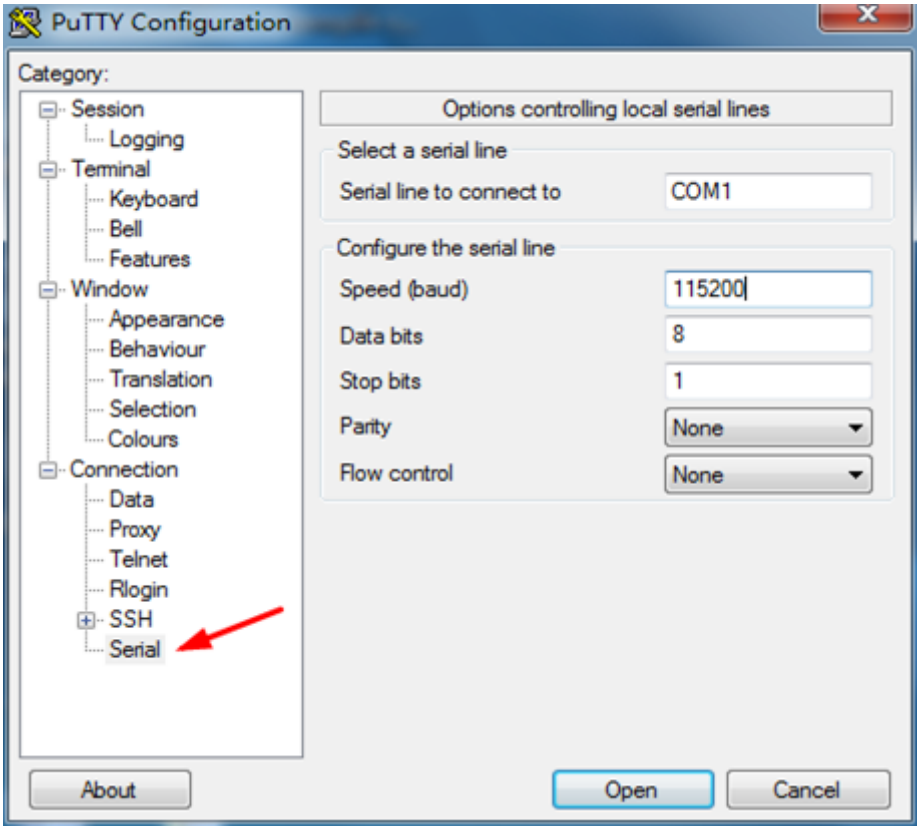
Yocto Lite system support the following login methods: login directly, login from debug serial, login from SSH.

2.3.1 Login Directly

Connect screen and keyboard to MaaXBoard Mini, username as “root” to login Yocto system.

2.3.2 Login from Debug Serial

- Install the Serial Communication software (e.g. PUTTY), select the corresponding port number, baud rate as 115200, data bits as 8, stop bits as 1, parity as none.



- Connect the debug interface to PC with USB to TTL converter. Pin 6, 8 and 10 of J1 to the GND, RXD and TXD pin of the USB to TTL converter.
- Enter username as “root” to login.

2.3.3 Login from SSH

MaaXBoard Mini Yocto OS install and startup SSH service automatically by default. Connect to internet, then login the system using SSH.

Linux system support ssh in default, in windows OS, you can install ssh by yourself, or use other software which support ssh, such as PuTTY, WinSCP, etc.

2.3.3.1 Preparation

Check the IP of MaaXBoard Mini: The IP will be used in ssh login.

```
root@maaxboardmini:~# ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.2.246  netmask 255.255.255.0  broadcast 192.168.2.255
        inet6 fe80::805f:bbff:fe1c:e1f3  prefixlen 64  scopeid 0x20<link>
        ether 82:5f:bb:1c:e1:f3  txqueuelen 1000  (Ethernet)
        RX packets 83  bytes 6521 (6.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 47  bytes 6002 (5.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

2.3.3.2 Login Command line

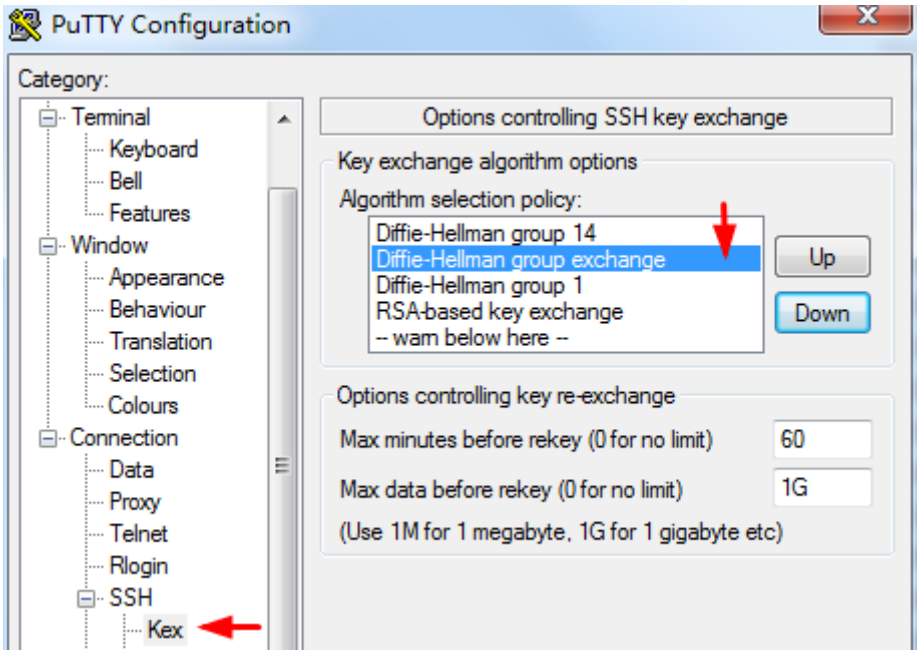
In this example, the IP of MaaXBoard Mini is 192.168.2.246, enter following command in command line window to connect: **ssh root@192.168.2.246**. Enter **yes** in the first connection, then will login to MaaXBoard Mini, enter **exit** to logout.

```
$ ssh root@192.168.2.246
PS C:\Users\username\Desktop> ssh root@192.168.2.246
The authenticity of host '192.168.2.98 (192.168.2.246)' can't be established.
ED25519 key fingerprint is SHA256:NZtcHgAxE+XRBskQqrkXw3yQyenrs9hCbG3F2IrsOBk.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.2.246' (ED25519) to the list of known hosts.
Last login: Tue Jul 25 02:57:28 2023 from 192.168.2.225
root@maaxboardmini:~#
```

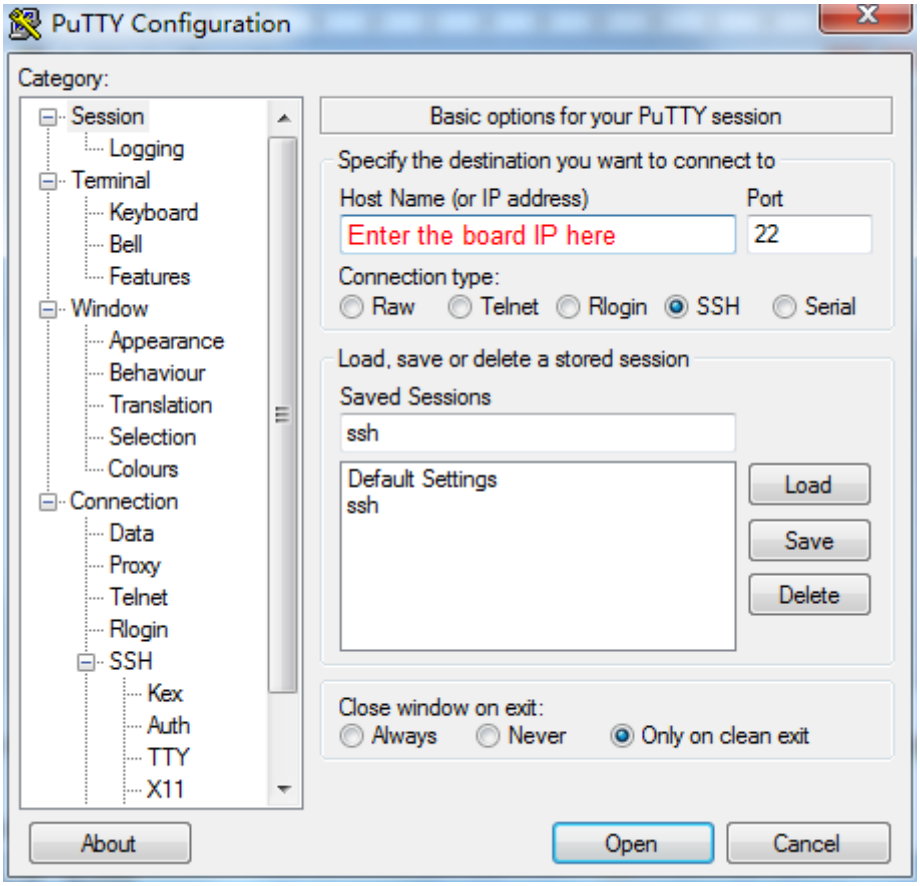
2.3.3.3 PuTTY

PuTTY support SSH, setting method as follows:

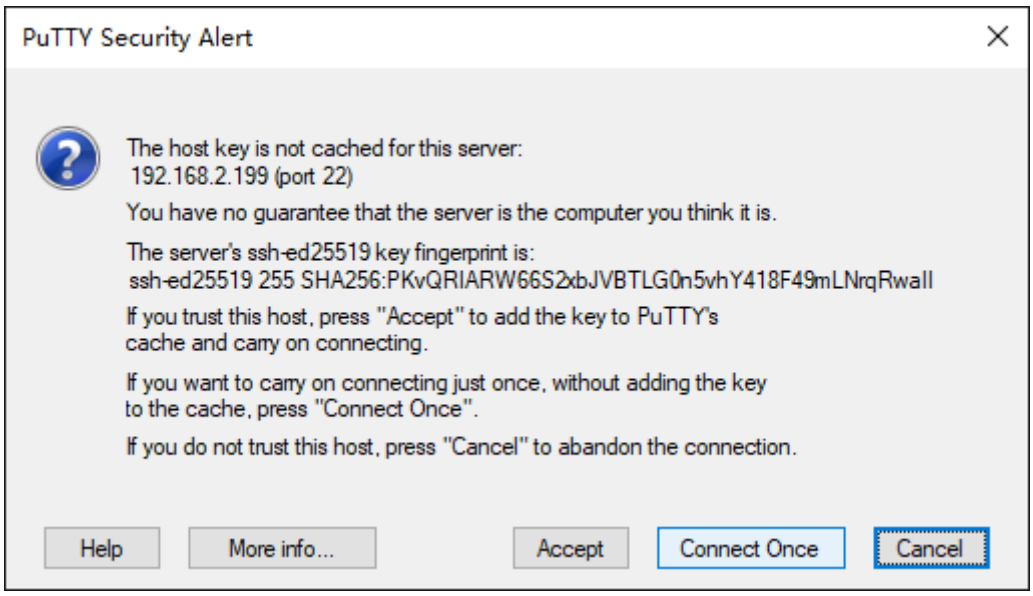
1. Run PUTTY, in Connection->SSH->Kex, change the **sequence** of algorithm.



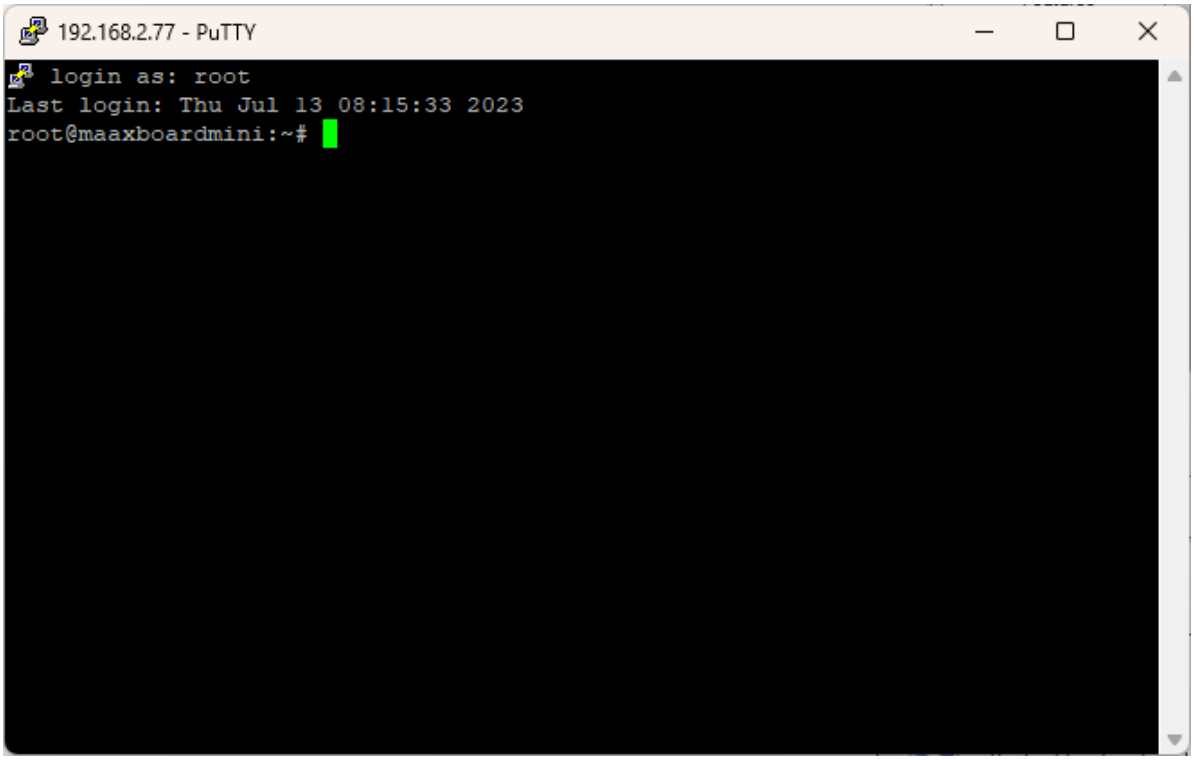
2. In Session, enter IP address, e.g. 192.168.2.246, port 22, and Connection type SSH, then click Open.



3. In the first connection, click Accept in the popout window.



4. Enter username as “root” to login, enter **exit** to logout.



Chapter 3 Feature Configuration & Introduction

First of all, please refer to the previous chapter and boot up the system. Then configure or use the functions according to the following guidance.

3.1 Settings in uEnv.txt

User could configure some environment variables in uEnv.txt, which can be loaded in the U-boot stage. The uEnv.txt file has a very simple file format. The format is a single *property=value* statement on each line, where value is either an integer or a string. Comments may be added, or existing configuration values may be commented out and disabled, by starting a line with the # character.

The device-tree overlay function is supported from this version and the device-tree overlay file (*.dtbo) is placed in the overlay/ directory in the FAT partition of the eMMC. To load the device-tree overlay file (*.dtbo), you need to set "***fdt_file***" and "**dtoverlay_**" prefix variable in uEnv.txt. Also you could add other configurations defined in U-boot to the uEnv.txt file.

The specific description is as follows:

Environment Variable	Value if Set (other invalid)	dtbo to be Loaded in U-boot
dtoverlay_camera	ov5640	camera-ov5640.dtbo
dtoverlay_display	mipi	display-mipi.dtbo
dtoverlay_usb0	device	usb0-device.dtbo
dtoverlay_gpio	'1' or 'yes'	ext-gpio.dtbo
dtoverlay_i2c	'2', '3', '2 3'	ext-i2c{2,3}.dtbo
dtoverlay_pwm	'1', '2', '3','1 2 3'	ext-pwm{1,2,3}.dtbo
dtoverlay_spi	'1'	ext-spi1.dtbo
dtoverlay_uart	'2'	ext-uart2.dtbo
dtoverlay_wm8960	'1' or 'yes'	ext-wm8960.dtbo
dtoverlay_extra	Other dtbo files to be loading, such as xxx.dtbo	
fdt_file	Board base dtb file, should be maaxboard-mini.dtb	
console	Some u-boot environment variables	

Note: ***fdt_file*** must be set to a device tree binary blob, which is the basis for applying dtbo file. ***fdt_file*** should be set, other configurations are optional.

Here is the default setting in uEnv.txt:

```
# Refer to readme.txt for more information on setting up U-Boot Env

# Device tree base file
fdt_file=maaxboard-mini.dtb

# Camera can support ov5640
#dtoverlay_camera=ov5640

# Display can support mipi only
#dtoverlay_display=mipi

# Enable USB0 work as device mode, default as host
#dtoverlay_usb0=device

# Set 40-pin extended GPIO pin default work as GPIO
dtoverlay_gpio=yes

# Enable i2c2/i2c3 on 40-pin extended GPIO pin
#dtoverlay_i2c=2 3

# Enable pwm1/pwm2/pwm3 on 40-pin extended GPIO pin
#dtoverlay_pwm=1 2 3

# Enable spi1 on 40-pin extended GPIO pin
```

```
#dtoverlay_spi=1

# Enable uart2 on 40-pin extended GPIO pin
#dtoverlay_uart=2

# Enable wm8960 on 40-pin extended GPIO pin
#dtoverlay_wm8960=yes

# Extra other device tree overlay
#dtoverlay_extra=1.dtbo 2.dtbo 3.dtbo

# U-boot bootargs for console
console=ttymxc0,115200 console=tty1
```

Modify uEnv.txt methods:

Mount the first partition of SD card or eMMC to the system, Then use **nano** or **vi** command to modify the uEnv.txt. After the modification, execute **sync** and **reboot** command to make it effect.

```
root@maaxboardmini:~# mkdir mount
root@maaxboardmini:~# mount /dev/mmcblk0p1 mount/
root@maaxboardmini:~# vi mount/uEnv.txt
```

We can edit the uEnv.txt as needed and save it.

```
# Refer to readme.txt for more information on setting up U-Boot Env

# Device tree base file
fdt_file=maaxboard-mini.dtb

# Camera can support ov5640
#dtoverlay_camera=ov5640

# Display can support mipi only
#dtoverlay_display=mipi

# Enable USB0 work as device mode, default as host
#dtoverlay_usb0=device

# Set 40-pin extended GPIO pin default work as GPIO
dtoverlay_gpio=yes

# Enable i2c2/i2c3 on 40-pin extended GPIO pin
#dtoverlay_i2c=2 3

# Enable pwm1/pwm2/pwm3 on 40-pin extended GPIO pin
#dtoverlay_pwm=1 2 3

# Enable spi1 on 40-pin extended GPIO pin
#dtoverlay_spi=1

# Enable uart2 on 40-pin extended GPIO pin
#dtoverlay_uart=2

# Enable wm8960 on 40-pin extended GPIO pin
#dtoverlay_wm8960=yes
```

```
# Extra other device tree overlay
#dtoverlay_extra=1.dtbo 2.dtbo 3.dtbo

# U-boot bootargs for console
console=ttymxc0,115200 console=tty1
```

After the modification, execute **sync** and **reboot** command to make it effect.

```
root@maaxboardmini:~# sync
root@maaxboardmini:~# reboot
```

3.2 USER LED

User can control the 2 single color LED indicators, LED0 and LED1 (corresponding to usr_led and sys_led) on MaaXBoard Mini Board. Execute the following instructions in serial terminal to control them.

Turn off the LEDs:

```
root@maaxboardmini:~# echo 0 > /sys/class/leds/usr_led/brightness
root@maaxboardmini:~# echo 0 > /sys/class/leds/sys_led/brightness
```

Turn on the LEDs:

```
root@maaxboardmini:~# echo 1 > /sys/class/leds/usr_led/brightness
root@maaxboardmini:~# echo 1 > /sys/class/leds/sys_led/brightness
```

3.3 Button Switches

There are three push-button switches on MaaXBoard Mini: BACK, HOME and PWR.

1. Test BACK and HOME button with following instructions:

Enter **evtest** command, then choose the event id for **gpio_keys**

```
root@maaxboardmini:~# evtest
No device specified, trying to scan all of /dev/input/event*
Available devices:
/dev/input/event0:      30370000.snvs:snvs-powerkey
/dev/input/event1:      gpio_keys
Select the device event number [0-1]: 1
Input driver version is 1.0.1
Input device ID: bus 0x19 vendor 0x1 product 0x1 version 0x100
Input device name: "gpio_keys"
Supported events:
  Event type 0 (EV_SYN)
  Event type 1 (EV_KEY)
    Event code 158 (KEY_BACK)
    Event code 172 (KEY_HOMEPAGE)
Properties:
Testing ... (interrupt to exit)
Event: time 1689060552.393152, type 1 (EV_KEY), code 172 (KEY_HOMEPAGE), value 1
Event: time 1689060552.393152, -------------- SYN_REPORT ------------
Event: time 1689060552.870408, type 1 (EV_KEY), code 172 (KEY_HOMEPAGE), value 0
```

```
Event: time 1689060552.870408, -------------- SYN_REPORT ------------
Event: time 1689060554.691100, type 1 (EV_KEY), code 158 (KEY_BACK), value 1
Event: time 1689060554.691100, -------------- SYN_REPORT ------------
Event: time 1689060554.926876, type 1 (EV_KEY), code 158 (KEY_BACK), value 0
Event: time 1689060554.926876, -------------- SYN_REPORT ------------
```

Use "**Ctrl+C**" to exit this test.

2. Press PWR button for 8s, system will enter suspend mode, press PWR again for 1s, the system will reboot.

Users could also test short press PWR button using **evtest** command:

```
root@maaxboardmini:~# evtest
No device specified, trying to scan all of /dev/input/event*
Available devices:
/dev/input/event0:      30370000.snvs:snvs-powerkey
/dev/input/event1:      gpio_keys
Select the device event number [0-1]: 0
Input driver version is 1.0.1
Input device ID: bus 0x19 vendor 0x0 product 0x0 version 0x0
Input device name: "30370000.snvs:snvs-powerkey"
Supported events:
  Event type 0 (EV_SYN)
  Event type 1 (EV_KEY)
    Event code 116 (KEY_POWER)
Properties:
Testing ... (interrupt to exit)
Event: time 1689060611.953774, type 1 (EV_KEY), code 116 (KEY_POWER), value 1
Event: time 1689060611.953774, -------------- SYN_REPORT ------------
Event: time 1689060612.145766, type 1 (EV_KEY), code 116 (KEY_POWER), value 0
Event: time 1689060612.145766, -------------- SYN_REPORT ------------
```

Use "**Ctrl+C**" to exit this test.

3.4 Display Output

MaaXBoard Mini supports MIPI-DSI screen display.

Users can connect the screen to MaaXBoard Mini before boot up the system according to the following table. When the system boot up, the screen will print the related startup message and login UI. Users can connect keyboard to login the MaaXBoard Mini file system.

Screen Type	Screen Resolution	Interface
MIPI-DSI	1280*720	J7

Display device could be chosen by modify the dtoverlay_display value in uEnv.txt.

Modify methods:

Mount the first partition of SD card or eMMC to the system, then use **nano** or **vi** command to modify the uEnv.txt. After the modification, execute **sync** and **reboot** command to make it effect.

```
root@maaxboardmini:~# mkdir mount
root@maaxboardmini:~# mount /dev/mmcblk0p1 mount/
root@maaxboardmini:~# vi mount/uEnv.txt
```

3.4.1 MIPI-DSI Screen

Choose MIPI-DSI screen, the dtoverlay_display value should be:

```
dtoverlay_display=mipi
```

MIPI-DSI supports backlight brightness adjustment. The backlight brightness has a range from 0 to 9, in which 9 means highest brightness, 0 means lowest.

Execute the following instructions on the serial terminal to implement the backlight test:

```
root@maaxboardmini:~# echo 5 > /sys/class/backlight/pwm-backlight/brightness
root@maaxboardmini:~# echo 0 > /sys/class/backlight/pwm-backlight/brightness
root@maaxboardmini:~# echo 9 > /sys/class/backlight/pwm-backlight/brightness
```

3.5 Touchscreen

The MIPI-DSI screen support touch screen. Use **evtest** command to test it.

```
root@maaxboardmini:~# evtest
No device specified, trying to scan all of /dev/input/event*
Available devices:
/dev/input/event0:      30370000.snvs:snvs-powerkey
/dev/input/event1:      gpio_keys
/dev/input/event2:      Goodix Capacitive TouchScreen
Select the device event number [0-2]: 2
Input driver version is 1.0.1
Input device ID: bus 0x18 vendor 0x416 product 0x38f version 0x1060
Input device name: "Goodix Capacitive TouchScreen"
Supported events:
  Event type 0 (EV_SYN)
  Event type 1 (EV_KEY)
    Event code 59 (KEY_F1)
    Event code 60 (KEY_F2)
    Event code 61 (KEY_F3)
    Event code 62 (KEY_F4)
    Event code 63 (KEY_F5)
    Event code 64 (KEY_F6)
    Event code 125 (KEY_LEFTMETA)
    Event code 330 (BTN_TOUCH)
  Event type 3 (EV_ABS)
    Event code 0 (ABS_X)
      Value      0
      Min        0
      Max      719
    Event code 1 (ABS_Y)
      Value      0
      Min        0
      Max     1279
    Event code 47 (ABS_MT_SLOT)
      Value      0
```

```
      Min        0
      Max        0
    Event code 48 (ABS_MT_TOUCH_MAJOR)
      Value      0
      Min        0
      Max      255
    Event code 50 (ABS_MT_WIDTH_MAJOR)
      Value      0
      Min        0
      Max      255
    Event code 53 (ABS_MT_POSITION_X)
      Value      0
      Min        0
      Max      719
    Event code 54 (ABS_MT_POSITION_Y)
      Value      0
      Min        0
      Max     1279
    Event code 57 (ABS_MT_TRACKING_ID)
      Value      0
      Min        0
      Max    65535
Properties:
  Property type 1 (INPUT_PROP_DIRECT)
Testing ... (interrupt to exit)
Event: time 1689060996.501590, type 3 (EV_ABS), code 57 (ABS_MT_TRACKING_ID),
value 0
Event: time 1689060996.501590, type 3 (EV_ABS), code 53 (ABS_MT_POSITION_X),
value 291
Event: time 1689060996.501590, type 3 (EV_ABS), code 54 (ABS_MT_POSITION_Y),
value 567
Event: time 1689060996.501590, type 3 (EV_ABS), code 48 (ABS_MT_TOUCH_MAJOR),
value 33
Event: time 1689060996.501590, type 3 (EV_ABS), code 50 (ABS_MT_WIDTH_MAJOR),
value 33
Event: time 1689060996.501590, type 1 (EV_KEY), code 330 (BTN_TOUCH), value 1
Event: time 1689060996.501590, type 3 (EV_ABS), code 0 (ABS_X), value 291
Event: time 1689060996.501590, type 3 (EV_ABS), code 1 (ABS_Y), value 567
Event: time 1689060996.501590, -------------- SYN_REPORT ------------
Event: time 1689060996.600863, type 3 (EV_ABS), code 53 (ABS_MT_POSITION_X),
value 294
Event: time 1689060996.600863, type 3 (EV_ABS), code 54 (ABS_MT_POSITION_Y),
value 569
Event: time 1689060996.600863, type 3 (EV_ABS), code 0 (ABS_X), value 294
Event: time 1689060996.600863, type 3 (EV_ABS), code 1 (ABS_Y), value 569
Event: time 1689060996.600863, -------------- SYN_REPORT ------------
```

Use "**Ctrl+C**" to exit this test.

3.6 Audio

MaaXBoard Mini supports USB audio device and Bluetooth audio device.

3.6.1 Check Audio Device IDs

Before playing or recording an audio interface, you should check the device ID.

Use the **aplay -l** and **arecord -l** commands to list the audio playback- and record- device IDs.

3.6.1.1 USB Audio Device

MaaXBoard Mini supports USB audio device (which do not need specified driver) to play audio. You can record and play audio from USB audio device. Connect a USB audio device to the USB port on MaaXBoard Mini, Use command **arecord -l** and **aplay -l** to check that the device id.

```
root@maaxboardmini:~# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: Seri [Plantronics Blackwire 3225 Seri], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

3.6.1.3 Bluetooth Audio

Yocto system also supports to play audio files via the Bluetooth audio device such as Bluetooth headset. For detail, refer to Bluetooth part.

3.6.2 Record Audio

Use the following command to record audio to file audio.wav:

```
root@maaxboardmini:~# arecord -c 2 -f S16_LE -r 48000 audio_sample.wav -D hw:0,0
```

Note: press **Ctrl+C** ro exit record.

In the above command:

S16_LE = audio format

-r 48000 = sample rate of the audio file (48KHz),

-c 2 = 2 channel audio recording,

-Dhw:0 = use audio card 0 to record (device id of the codec-connected MIC),

Use command **arecord -l** and **aplay -l** to check the device ID.

Change those parameters according to your device.

3.6.3 Play Audio File

MaaXBoard Mini can play audio files with the connected USB or Bluetooth device using the following commands:

```
root@maaxboardmini:~# pulseaudio -D -v
root@maaxboardmini:~# aplay audio_sample.wav
root@maaxboardmini:~# gst-play-1.0 audio_sample.wav
root@maaxboardmini:~# mpg123 audio_sample1.mp3
```

The **aplay** command supports audio file in wav format, **gst-play-1.0** command supports wav, mp3 and aac format, while the **mpg123** command supports mp3 format.

3.7 Video

Yocto system supports to play video file in mp4 format, the largest support resolution is 4K, Use **gplay-1.0** or **gst-launch-1.0** command to play video files.

Connect MaaXBoard Mini to the MIPI-DSI screen display, take video.mp4 file as an example, select one of the following four commands and enter it in the serial terminal.

```
root@maaxboardmini:~# gplay-1.0 video.mp4
root@maaxboardmini:~# gplay-1.0 --video-sink=waylandsink video.mp4
root@maaxboardmini:~# gst-launch-1.0 playbin uri=file:///home/root/video.mp4
root@maaxboardmini:~# gst-launch-1.0 filesrc location=video.mp4 typefind=true !
video/quicktime ! ax ! queue max-size-time=0 ! vpudec ! autovideosink
```

3.8 Camera

MaaXBoard Mini supports USB Camera and MIPI-CSI Camera. This part will introduce how to preview, photograph and record video under Command line.

To use the MIPI-CSI camera, the ***dtoverlay_camera*** value should be set:

```
dtoverlay_camera=ov5640
```

Note: To show the camera preview on the desktop, it is recommended that the ***dtoverlay_display*** option should be set in **uEnv.txt**.

3.8.1 Check Device ID

Connect the MIPI-CSIcamera and USB camera, use following command to check the device IDs

```
root@maaxboardmini:~# ls /dev/video*
/dev/video0  /dev/video1  /dev/video2  /dev/video3  /dev/video4
```

In default, MIPI-CSI camera is /dev/video0, USB Camera is /dev/video3. The device ID will be used in following command.

3.8.2 Preview

Use the following instruction to open Camera and preview the video on the screen.

```
root@maaxboardmini:~# gst-launch-1.0 v4l2src device=/dev/video0 ! autovideosink
root@maaxboardmini:~# gst-launch-1.0 v4l2src device=/dev/video3 ! autovideosink
```

Note: Press **Ctrl+C** to exit, change /dev/video0 to your device ID.

3.8.3 Take Photo

Use the following instruction to take a photo and saved to specific location.

```
gst-launch-1.0 v4l2src device=[video] num-buffers=1 ! jpegenc ! filesink
location=[filename]
```

In above command, replace [video] to the camera device ID, filename] to the path and name of saved file. For example:

```
root@maaxboardmini:~# gst-launch-1.0 v4l2src device=/dev/video0 num-buffers=1 !
jpegenc ! filesink location=sample.jpg
```

Copy the photo to other device, such as computer to display it.

3.8.4 Record Video

Use the following instruction to record a video in mp4 format and saved to specific location.

```
root@maaxboardmini:~# gst-launch-1.0 -e v4l2src device=/dev/video0 num-
buffers=100 ! video/x-raw,format=YUY2,framerate=30/1, width=640, height=480 !
videoconvert ! x264enc ! video/x-h264, profile=baseline ! mp4mux ! filesink
location=output.mp4
```

In above command, modify the camera device ID, the width and height of the video, the path and name of saved file, etc. The video file can be copy to other device, such as computer to display, or use gst-play-1.0 to display it on the screen directly.

```
root@maaxboardmini:~# gst-play-1.0 output.mp4
```

3.9 Gigabit Ethernet Interface

Connect the network cable to J8, enter the following instructions to set the IP address:

The below IP address are example, replace it with your real network environment

3.9.1 Network Test

After connecting the network cable, it will automatically obtain the IP by default. You can use the ifconfig command to view the IP information and use the following command to perform the network test:

```
root@maaxboardmini:~# ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.2.230  netmask 255.255.255.0  broadcast 192.168.2.255
        inet6 fe80::b876:34ff:fe59:e77b  prefixlen 64  scopeid 0x20<link>
        ether ba:76:34:59:e7:7b  txqueuelen 1000  (Ethernet)
        RX packets 1159  bytes 103974 (101.5 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 88  bytes 7836 (7.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@maaxboardmini:~# ping www.avnet.com
PING www.avnet.com (184.50.161.105) 56(84) bytes of data.
64 bytes from a184-50-161-105.deploy.static.akamaitechnologies.com
(184.50.161.105): icmp_seq=1 ttl=50 time=315 ms
64 bytes from a184-50-161-105.deploy.static.akamaitechnologies.com
(184.50.161.105): icmp_seq=2 ttl=50 time=352 ms
64 bytes from a184-50-161-105.deploy.static.akamaitechnologies.com
(184.50.161.105): icmp_seq=3 ttl=50 time=255 ms
```

Note: Press **Ctrl+C** to exit

3.9.2 Set Static IP

If you need to set a static IP, execute the following commands:

```
root@maaxboardmini:~# vi /etc/systemd/network/01-eth0.network
[Match]
Name=eth0

[Network]
Address=192.168.2.77/24
Gateway=192.168.2.1
DNS=8.8.8.8
DNS=114.114.114.114

root@maaxboardmini:~# systemctl restart systemd-networkd
```

In above command, replace the IP address, router, DNS with your real network environment. Execute **sync** after the modification, then **reboot** the system to make it effect.

3.9.3 Set Dynamic IP

```
root@maaxboardmini:~# vi /etc/systemd/network/01-eth0.network
[Match]
Name=eth0

[Network]
DHCP=yes
root@maaxboardmini:~# systemctl restart systemd-networkd
```

Execute **sync** after the modification, then **reboot** the system to make it effect.

3.10 USB 2.0 Interface

MaaXBoard Mini supports 4 USB Interfaces, the lower one in J2 is USB0, the upper one in J2 is USB3, the lower one in J4 is USB2, the upper one in J4 is USB1. All of the 4 USB 2.0 interfaces support USB HOST function, only USB0 supports USB Device function.

3.10.1 USB Host

Insert a U-disk to USB interface, serial terminal will display the disk information:

```
[ 1661.672508] usb 2-1.1: USB disconnect, device number 4
[ 1667.119898] usb 2-1.3: new high-speed USB device number 5 using ci_hdrc
[ 1667.239149] usb-storage 2-1.3:1.0: USB Mass Storage device detected
[ 1667.245977] scsi host0: usb-storage 2-1.3:1.0
[ 1668.269441] scsi 0:0:0:0: Direct-Access     Generic  MassStorageClass 1536 PQ:
0 ANSI: 6
[ 1668.444300] sd 0:0:0:0: [sda] 15273984 512-byte logical blocks: (7.82 GB/7.28
GiB)
[ 1668.454183] sd 0:0:0:0: [sda] Write Protect is off
[ 1668.460592] sd 0:0:0:0: [sda] Write cache: disabled, read cache: enabled,
doesn't support DPO or                                  FUA
[ 1668.477413]  sda: sda1 sda2
[ 1668.481002] sd 0:0:0:0: [sda] Attached SCSI removable disk
```

Execute the following instructions on the serial terminal:

```
root@maaxboardmini:~# ls /dev/sd*
/dev/sda  /dev/sda1  /dev/sda2
```

The storage node for U disk is /dev/sda, users could mount the storage device to the file system to read and write data.

MaaXBoard Mini also supports other USB device such as key board, mouse, Camera, etc.

3.10.2 USB Device

USB0 supports USB Device function could be used to program the system image or used as USB Serial adapter.

3.10.2.1 Programming Mode

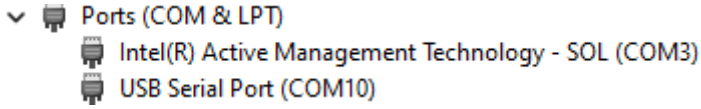
Connect USB0 and PC before power on the board. The system will not boot normally, it will enter programming mode. Then users could program the system image to the development board using uuu tools. For the detail information, refer to 4.2 Program Images using *uuu*.

3.10.2.2 USB Serial Adapter

To use USB0 as USB slave device: serial adapter, users should enable the following command in uEnv.txt and reboot the system.

```
dtoverlay_usb0=device
```

Connect USB0 to PC after the system start up, open the device manager, and check if the following device is recognized:



Please follow the steps listed below to finish USB Device test：

1. Install the gadget driver for the serial device

```
root@maaxboardmini:~# modprobe g_serial
[   25.609133] g_serial gadget.0: Gadget Serial v2.4
[   25.613916] g_serial gadget.0: g_serial ready
```

2. Set the serial port device parameters and communicate with the PC

```
root@maaxboardmini:~# busybox microcom -s 115200 /dev/ttyGS0
abcdfghjjj
```

   Note: Press **Ctrl+X** to exit the communication

3. Remove the module

```
root@maaxboardmini:~# rmmod g_serial
```

3.11 Wi-Fi

The on-board Wi-Fi module support 2.4G/5G network and hotspot.

3.11.1 Enable Wi-Fi

User can create wpa_supplicant.conf in /run/media/boot-mmcblk0p1/ to load the Wi-Fi driver and firmware (take "TEST23"&"12345678" as an example), Run the following commands to start Wi-Fi:

```
root@maaxboardmini:~# vi /run/media/boot-mmcblk0p1/wpa_supplicant.conf

ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=0
update_config=1
network={
scan_ssid=1
ssid="TEST23"
psk="12345678"
}

root@maaxboardmini:~# systemctl enable wpa-conf.service wpa-conf.timer
root@maaxboardmini:~# reboot
```

After system reboot, User can check the mlan0 and test Wi-Fi network with ping command:

```
root@maaxboardmini:~# iwconfig wlan0
wlan0     IEEE 802.11  ESSID:"TEST23"
          Mode:Managed  Frequency:5.745 GHz  Access Point: 80:8F:1D:8A:F6:D2
          Bit Rate=24 Mb/s   Tx-Power=31 dBm
          Retry short limit:7   RTS thr:off   Fragment thr:off
          Encryption key:off
          Power Management:on
          Link Quality=52/70  Signal level=-58 dBm
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:0  Invalid misc:0   Missed beacon:0
```

```
root@maaxboardmini:~# ifconfig wlan0
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.2.240  netmask 255.255.255.0  broadcast 192.168.2.255
        inet6 fe80::8291:33ff:fe4a:1049  prefixlen 64  scopeid 0x20<link>
        ether 80:91:33:4a:10:49  txqueuelen 1000  (Ethernet)
        RX packets 1602  bytes 293802 (286.9 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 158  bytes 15391 (15.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
root@maaxboardmini:~# ping www.avnet.com -I wlan0
PING www.avnet.com (184.50.161.105) 56(84) bytes of data.
64 bytes from a184-50-161-105.deploy.static.akamaitechnologies.com
(184.50.161.105): icmp_seq=1 ttl=50 time=315 ms
64 bytes from a184-50-161-105.deploy.static.akamaitechnologies.com
(184.50.161.105): icmp_seq=2 ttl=50 time=352 ms
64 bytes from a184-50-161-105.deploy.static.akamaitechnologies.com
(184.50.161.105): icmp_seq=3 ttl=50 time=255 ms
```

Use "**Ctrl+C**" to exit this test.

3.11.2 Connect Wi-Fi Manually

Execute the following instructions on the serial terminal to search Wi-Fi network, It will print the information for all available network:

```
root@maaxboardmini:/etc# ifconfig wlan0 up
root@maaxboardmini:~# iwlist wlan0 scan | grep ESSID
                    ESSID:"TEST23"
                    ESSID:"AAAA_2.4G" [2]
                    ESSID:"Development " [3]
                    ESSID:"XXXX-5G" [4]
```

Configure SSID and SSID_PASSWD with the following command: (take "TEST23" as an example)

```
root@maaxboardmini:~# wpa_passphrase "TEST23" "12345678" >>
/etc/wpa_supplicant.conf
```

Or edit ***/etc/wpa_supplicant.conf*** directly and append the following parameters:

```
root@maaxboardmini:~# vi /etc/wpa_supplicant.conf
```

Add following info into this file:

```
network={
    ssid="TEST23"
    psk="12345678"
}
```

Then execute the following command:

```
root@maaxboardmini:~# wpa_supplicant -B -i wlan0 -c /etc/wpa_supplicant.conf
Successfully initialized wpa_supplicant
rfkill: Cannot open RFKILL control device
rfkill: Cannot get wiphy information
```

Run the following command to check wireless connection, it will print the following info once connected:

```
root@maaxboardmini:~# iwconfig wlan0 | grep ESSID
wlan0   IEEE 802.11 ESSID:"TEST23"
```

Run the command to get the IP address:

```
root@maaxboardmini:~# udhcpc -i wlan0 -n -R
root@maaxboardmini:~# ifconfig wlan0
```

Test Wi-Fi network with **ping** command:

```
root@maaxboardmini:~# ping www.root@maaxboardmini:~# ping www.avnet.com -I wlan0
PING www.avnet.com (184.50.161.105) 56(84) bytes of data.
64 bytes from a184-50-161-105.deploy.static.akamaitechnologies.com
(184.50.161.105): icmp_seq=1 ttl=50 time=315 ms
64 bytes from a184-50-161-105.deploy.static.akamaitechnologies.com
(184.50.161.105): icmp_seq=2 ttl=50 time=352 ms
64 bytes from a184-50-161-105.deploy.static.akamaitechnologies.com
(184.50.161.105): icmp_seq=3 ttl=50 time=255 ms
```

Use "**Ctrl+C**" to exit this test.

3.11.3 Wi-Fi Hotspot

Use the following steps to configure and start the 2.4 GHz/5 GHz Access Point from the wireless module.

Make sure the Wi-Fi is disconnected:

```
root@maaxboardmini:~# killall wpa_supplicant
root@maaxboardmini:~# killall hostapd
```

Then use the following steps to set up Wi-Fi hotspot.

Edit the configuration file for hostapd:

```
root@maaxboardmini:~# vi /etc/hostapd.conf
```

Parameter values in the configuration file:

```
interface=wlan0
# specify the band: hw_mode=g (2.4 GHz) and hw_mode=a (5 GHz)
hw_mode=g
channel=1
country_code=US
ssid=MY_HOSTAP
ieee80211n=1
```

Note: If you want to configure the 5 GHz Access Point, change hw_mode=a and default is 2.4 GHz AP.

If you want to configure WPA2 for the AP using open source supplicant, need to add the following additional lines:

```
wpa=2
wpa_key_mgmt=WPA-PSK
rsn_pairwise=CCMP
wpa_passphrase=123456789
```

Note: You can modify your **ssid** and **wpa_passphrase** in hostapd.conf file.

Create the configuration file for udhcp server:

```
root@maaxboardmini:~# vi /etc/udhcpd.conf
```

Add the following content to **udhcpd.conf** file:

```
interface wlan0
start 192.168.6.10
end 192.168.6.100
opt router 192.168.6.1
opt dns 114.114.114.114 8.8.8.8
```

Note: The IP address 192.168.6.x can be modified at will but it must be consistent with its related IP.

Command to start the 5 GHz Access Point and start udhcp server to assign the IP address:

```
root@maaxboardmini:~# ifconfig wlan0 192.168.6.1 netmask 255.255.255.0 up
root@maaxboardmini:~# udhcpd /etc/udhcpd.conf
root@maaxboardmini:~# hostapd -B /etc/hostapd.conf
```

At this time, you can use other devices to scan the access point **"MY_HOSTAP"**, and enter the password **"123456789"** to connect. After obtaining the IP address, the device will display a status of "Connected, no Internet".

If the Ethernet interface is connected to the Internet, you can use the following commands to add packet forwarding rules so that the devices connected to the hotspot can access the Internet.

```
root@maaxboardmini:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@maaxboardmini:~# iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
root@maaxboardmini:~# iptables -A FORWARD -i eth0 -o wlan0 -m state --state
RELATED,ESTABLISHED -j ACCEPT
root@maaxboardmini:~# iptables -A FORWARD -i wlan0 -o eth0 -j ACCEPT
```

3.12 Bluetooth 4.1

3.12.1 Initialize the Bluetooth Module

Execute the following instructions on the serial terminal:

```
root@maaxboardmini:~# hciattach /dev/ttymxc3 bcm43xx 3000000
bcm43xx_init
Set Controller UART speed to 3000000 bit/s
Flash firmware /etc/firmware/BCM4345C0.1MW.hcd
Set Controller UART speed to 3000000 bit/s
Setting TTY to N_HCI line discipline
Device setup complete
root@maaxboardmini:~#  hciconfig hci0 up
```

3.12.2 Connect Bluetooth Device

Use **bluetoothctl** to connect Bluetooth Device:

```
root@maaxboardmini:~# bluetoothctl
[bluetooth]# power on
[bluetooth]# pairable on
[bluetooth]# agent on
[bluetooth]# default-agent
```

Make the MaaXBoard Mini discoverable by other Bluetooth device:

```
[bluetooth]# discoverable on
```

Enable and Disable Scan:

```
[bluetooth]# scan on
[bluetooth]# scan off
```

Pair and connect the device:

```
[bluetooth]# pair E8:EC:A3:21:57:6C
[bluetooth]# trust E8:EC:A3:21:57:6C
[bluetooth]# connect E8:EC:A3:21:57:6C
[device name]# disconnect E8:EC:A3:21:57:6C
```

Exit **bluetoothctl**.

```
[bluetooth]# exit
```

In above instructions, **E8:EC:A3:21:57:6C** is the address of the Bluetooth device, change it according to your device.

3.12.3 Configure A2DP sink or source

When connected to a remote Bluetooth device that supports the A2DP sink feature, MaaXBoard Mini can be configured as an A2DP Source. This type of bluetooth devices are usually bluetooth headsets, bluetooth speakers, etc.

Set the baud rate to 3M bps,

```
root@maaxboardmini:~# hciattach /dev/ttymxc3 bcm43xx 3000000
bcm43xx_init
Set Controller UART speed to 3000000 bit/s
Flash firmware /etc/firmware/BCM4345C0.1MW.hcd
Set Controller UART speed to 3000000 bit/s
Setting TTY to N_HCI line discipline
Device setup complete
root@maaxboardmini:~# hciconfig hci0 up
```

```
root@maaxboardmini:~# killall hciattach
[ 132.691727] Bluetooth: hci0: sending frame failed (-49)
```

```
root@maaxboardmini:~# hciconfig
hci0:   Type: Primary  Bus: UART
        BD Address: 70:66:55:8A:AE:B2  ACL MTU: 1016:5  SCO MTU: 60:12
        DOWN
        RX bytes:735 acl:0 sco:0 events:43 errors:0
        TX bytes:470 acl:0 sco:0 commands:43 errors:0
```

Run the following command to verify the Audio Sink Profile capability of the connected Bluetooth device:

```
root@maaxboardmini:~# pulseaudio -D -v
root@maaxboardmini:~# bluetoothctl info
```

Command output example showing the feature:

```
UUID: Audio Sink                (0000110b-0000-1000-8000-00805f9b34fb)
        UUID: A/V Remote Control Target (0000110c-0000-1000-8000-00805f9b34fb)
        UUID: A/V Remote Control        (0000110e-0000-1000-8000-00805f9b34fb)
        UUID: PnP Information           (00001200-0000-1000-8000-00805f9b34fb)
        UUID: Generic Access Profile    (00001800-0000-1000-8000-00805f9b34fb)
        UUID: Generic Attribute Profile (00001801-0000-1000-8000-00805f9b34fb)
        Modalias: usb:v1D6Bp0246d0525
```

Connect Bluetooth device:

```
root@maaxboardmini:~# bluetoothctl
[bluetooth]# power on
[bluetooth]# pairable on
[bluetooth]# default-agent
[bluetooth]# scan on
[NEW] Device 8C:53:C3:21:8A:EE device-name
[bluetooth]# pair 8C:53:C3:21:8A:EE
Attempting to pair with 8C:53:C3:21:8A:EE
```

```
[CHG] Device 8C:53:C3:21:8A:EE Connected: yes
[CHG] Device 8C:53:C3:21:8A:EE UUIDs: 0000110b-0000-1000-8000-00805f9b34fb
[CHG] Device 8C:53:C3:21:8A:EE UUIDs: 0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device 8C:53:C3:21:8A:EE UUIDs: 0000110e-0000-1000-8000-00805f9b34fb
[CHG] Device 8C:53:C3:21:8A:EE UUIDs: 00001200-0000-1000-8000-00805f9b34fb
[CHG] Device 8C:53:C3:21:8A:EE UUIDs: 00001800-0000-1000-8000-00805f9b34fb
[CHG] Device 8C:53:C3:21:8A:EE UUIDs: 00001801-0000-1000-8000-00805f9b34fb
[CHG] Device 8C:53:C3:21:8A:EE ServicesResolved: yes
[CHG] Device 8C:53:C3:21:8A:EE Paired: yes
Pairing successful
[bluetooth]# scan off
[bluetooth]# connect 8C:53:C3:21:8A:EE
Attempting to connect to 8C:53:C3:21:8A:EE
[CHG] Device 8C:53:C3:21:8A:EE Connected: yes
Connection successful
[device-name]# exit
root@maaxboardmini:~#
```

Use the following command to view the current available sound cards and the default sound card:

```
root@maaxboardmini:~# pacmd list-sinks | egrep '(index|name):'
  * index: 1
        name: <bluez_sink.8C_53_C3_21_8A_EE.a2dp_sink>
root@maaxboardmini:~# pactl get-default-sink
bluez_sink.8C_53_C3_21_8A_EE.a2dp_sink
```

If the Bluetooth device isn't the default sound card, set it as follow:

```
root@maaxboardmini:~# pactl set-default-sink 1
```

Play the audio file using pulseaudio play utility, the music will be played from the connected Bluetooth device.:

```
root@maaxboardmini:~# gst-play-1.0 audio_sample.wav
root@maaxboardmini:~# mpg123 audio_sample1.mp3
root@maaxboardmini:~# paplay audio_sample.wav
```

Use the following command to adjust the Bluetooth speaker playback volume, where < volume_level> Set to 0 for silent, 0x10000 for normal volume.

```
root@maaxboardmini:~# pacmd set-sink-volume <sink_index> <volume_level>
root@maaxboardmini:~# pacmd set-sink-volume 1 0
root@maaxboardmini:~# pacmd set-sink-volume 1 0x10000
root@maaxboardmini:~# pacmd set-sink-volume 1 0x9000
root@maaxboardmini:~# pacmd set-sink-volume 1 0x15000
```

3.12.4 Send Files

Run the OBEXD daemon and connect to the target Bluetooth device

```
root@maaxboardmini:~# echo hello > 1.txt
root@maaxboardmini:~# export $(dbus-launch)
root@maaxboardmini:~# /usr/libexec/bluetooth/obexd -r /home/root -a -d & obexctl
[1] 626
```

```
[NEW] Client /org/bluez/obex
[obex]# connect 94:87:E0:DF:90:2D
Attempting to connect to 94:87:E0:DF:90:2D
[NEW] Session /org/bluez/obex/client/session0 [default]
[NEW] ObjectPush /org/bluez/obex/client/session0
Connection successful
[94:87:E0:DF:90:2D]# send /home/root/1.txt
Attempting to send /home/root/1.txt to /org/bluez/obex/client/session0
[NEW] Transfer /org/bluez/obex/client/session0/transfer0
Transfer /org/bluez/obex/client/session0/transfer0
        Status: queued
        Name: 1.txt
        Size: 6
        Filename: /home/root/1.txt
        Session: /org/bluez/obex/client/session0
[CHG] Transfer /org/bluez/obex/client/session0/transfer0 Status: complete
[DEL] Transfer /org/bluez/obex/client/session0/transfer0
[94:87:E0:DF:90:2D]# exit
```

In above instructions, 94:87:E0:DF:90:2D is the address of target device, change it according to your device.

3.13 UART

MaaXBoard Mini supports 2 UART interfaces.

MaaXBoard Mini (CPU)	Interface Type
UART1	UART TTL (Debug Interface)
UART2	UART TTL

3.13.1 UART2

In the Yocto system, the node for UART2 is /dev/ttymxc1. Users could also write their own applications to control the uart.

3.14 Pi HAT 40 Pin Expansion Interface

This chapter will provide the Control methods of 40 Pin interface, include GPIO, I2C and SPI.

3.14.1 GPIO

System use /sys/class/gpio to control the GPIO pin, refer to the following table:

Table: GPIO corresponding relation table

PINMUX	Function	PIN	PIN	Function	PINMUX
	3.3V	1	2	5V	
GPIO5_IO17	SDA2	3	4	5V	
GPIO5_IO16	SCL2	5	6	GND	
GPIO1_IO14	PWM3	7	8	UART1_TX	GPIO5_IO23
	GND	9	10	UART1_RX	GPIO5_IO22
GPIO3_IO17	GPIO	11	12	SAI3_TXC	GPIO5_IO0
GPIO3_IO4	GPIO	13	14	GND	
GPIO3_IO18	GPIO	15	16	UART2_RX	GPIO5_IO24
	3.3V	17	18	UART2_TX	GPIO5_IO25
GPIO5_IO7	SPI1_MOSI	19	20	GND	
GPIO5_IO8	SPI1_MISO	21	22	GPIO	GPIO5_IO28
GPIO5_IO6	SPI1_SCLK	23	24	SPI1_SS	GPIO5_IO9
	GND	25	26	PWM1	GPIO1_IO1
GPIO5_IO19	SDA3	27	28	SCL3	GPIO5_IO18
GPIO1_IO15	GPIO	29	30	GND	
GPIO4_IO27	GPIO	31	32	GPIO	GPIO5_IO5
GPIO5_IO4	PWM2	33	34	GND	
GPIO4_IO31	SAI3_TXFS	35	36	GPIO	GPIO3_IO16
GPIO1_IO10	GPIO	37	38	SAI3_RXD	GPIO4_IO30
	GND	39	40	SAI3_TXD	GPIO5_IO1

pinum= $(group-1) * $groupin + $pin + $pinbase（pinbase=0, groupin=32）

Here we take PIN35(GPIO4_IO31) as an example:

1. In above table, the GPIO Number of connector PIN35 is calculated to be 127.

GPIO4_IO31 means group=4, pin=31 for calculation of: (4-1)x32 + 31 + 0 = 127

2. Set the function of Pin35 to be GPIO output.

```
root@maaxboardmini:~# echo 127 >/sys/class/gpio/export
root@maaxboardmini:~# echo out >/sys/class/gpio/gpio127/direction
```

3. Set the level of Pin 35, 0 means low, 1 means high.

```
root@maaxboardmini:~# echo 1 >/sys/class/gpio/gpio127/value
```

3.14.2 SPI

To use the SPI of the 40-pins interface, enable the following options in uEnv.txt :

Add **dtoverlay_spi=5** to **uEnv.txt**, then execute **sync** and **reboot** command to make it effect.

Connect SPI_MOSI(#19) and SPI_MISO(#21), then execute ***spidev_test***, the result:

```
root@maaxboardmini:~# spidev_test -D /dev/spidev0.0 -v
spi mode: 0x4
bits per word: 8
max speed: 500000 Hz (500 kHz)
TX | FF FF FF FF FF FF 40 00 00 00 00 95 FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF F0 0D  |......@.......................|
RX | FF FF FF FF FF FF 40 00 00 00 00 95 FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF F0 0D  |......@.......................|
```

Disconnect SPI_MOSI(#19) and SPI_MISO(#21), then execute ***spidev_test***, the result:

```
root@maaxboardmini:~# spidev_test -D /dev/spidev0.0 -v
spi mode: 0x4
bits per word: 8
max speed: 500000 Hz (500 kHz)
TX | FF FF FF FF FF FF 40 00 00 00 00 95 FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF F0 0D  |......@.......................|
RX | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00  |..............................|
```

3.14.3 IIC

To use the IIC of the 40-pins interface, enable the following options in uEnv.txt :

Add **dtoverlay_i2c=2 3** to uEnv.txt, then execute **sync** and **reboot** command to make it effect.

Connect SDA2 (#3), SCL2 (#5), VCC and GND to a IIC device, then execute ***i2cdetect***, MaaXBoard Mini can detect the IIC device:

```
root@maaxboardmini:~# i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                         -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: 40 -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

Connect SDA3 (#27), SCL3 (#28), VCC and GND to a IIC device, then execute ***i2cdetect***, MaaXBoard Mini can detect the IIC device:

```
root@maaxboardmini:~# i2cdetect -y 2
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                         -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: 40 -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

3.14.4 PWM

To use the PWM of the 40-pins interface, enable the following options in uEnv.txt :

Add **dtoverlay_pwm=1** to uEnv.txt, then execute **sync** and **reboot** commands to make it effect.

Configure PWM1:

```
root@maaxboardmini:~# echo 0 > /sys/class/pwm/pwmchip0/export
root@maaxboardmini:~# echo 1 > /sys/class/pwm/pwmchip0/pwm0/enable
root@maaxboardmini:~# echo 1000000 > /sys/class/pwm/pwmchip0/pwm0/period
```

Connect PWM1(PIN26) to a test LED

Execute the following commands, set different duty_cycle values, the brightness of the LED will change according to the duty_cycle value

```
root@maaxboardmini:~# echo 1000000 > /sys/class/pwm/pwmchip0/pwm0/period
root@maaxboardmini:~# echo 100000 > /sys/class/pwm/pwmchip0/pwm0/duty_cycle
root@maaxboardmini:~# echo 10000 > /sys/class/pwm/pwmchip0/pwm0/duty_cycle
root@maaxboardmini:~# echo 0 > /sys/class/pwm/pwmchip0/pwm0/duty_cycle
```

3.15 Procedure to Increase eMMC Partition Size

Since the image is smaller than the storage device, it is not properly expanded when first flashed.

To avoid running out of eMMC or SD card space when installing the software package, perform the following steps to expand the rootfs partition.

- View the Partition Size using following command:

```
root@maaxboardmini:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       3.3G  2.1G  1.1G  66% /
devtmpfs        669M  4.0K  669M   1% /dev
tmpfs           990M     0  990M   0% /dev/shm
tmpfs           396M  8.9M  388M   3% /run
tmpfs           4.0M     0  4.0M   0% /sys/fs/cgroup
tmpfs           990M  4.0K  990M   1% /tmp
tmpfs           990M  168K  990M   1% /var/volatile
/dev/mmcblk0p1   84M   32M   52M  38% /run/media/boot-mmcblk0p1
tmpfs           198M  4.0K  198M   1% /run/user/0
```

- Expand the partition using following command:

```
root@maaxboardmini:~# expand_rootfs
[16888.526289] EXT4-fs (mmcblk0p2): resizing filesystem from 905938 to 3803136
blocks
[16888.635026] EXT4-fs (mmcblk0p2): resized filesystem to 3803136
Expand rootfs size successfully, it will be enlarged upon the next reboot.
```

- View again, the partition size has been expanded:

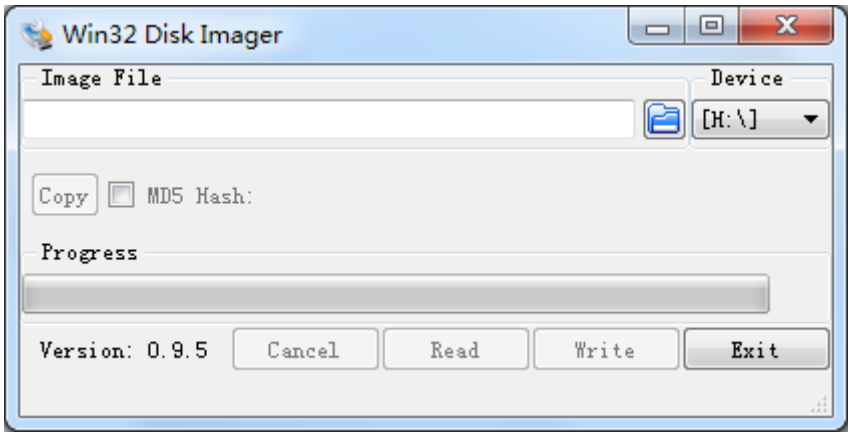
```
root@maaxboardmini:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        14G  2.1G   12G  16% /
devtmpfs        669M  4.0K  669M   1% /dev
tmpfs           990M     0  990M   0% /dev/shm
tmpfs           396M  8.9M  388M   3% /run
tmpfs           4.0M     0  4.0M   0% /sys/fs/cgroup
tmpfs           990M  4.0K  990M   1% /tmp
tmpfs           990M  168K  990M   1% /var/volatile
/dev/mmcblk0p1   84M   32M   52M  38% /run/media/boot-mmcblk0p1
tmpfs           198M  4.0K  198M   1% /run/user/0
```

Chapter 4 Program or update the system Images

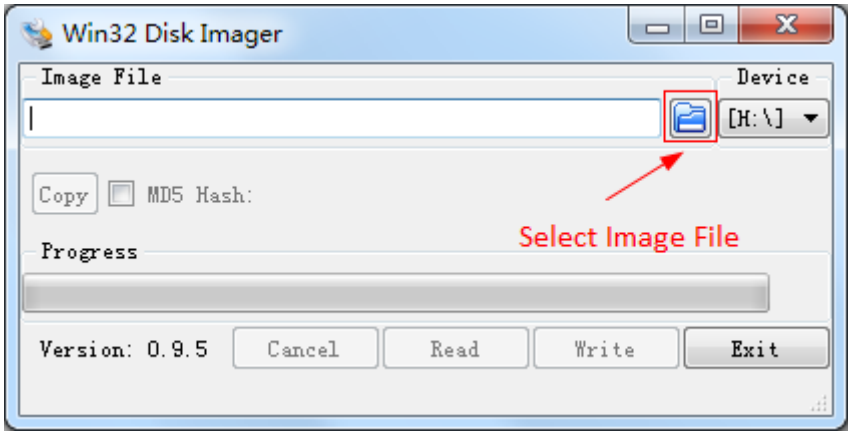
4.1 Program the system Image into SD Card directly

4.1.1 Program the System Image into SD Card under Windows OS

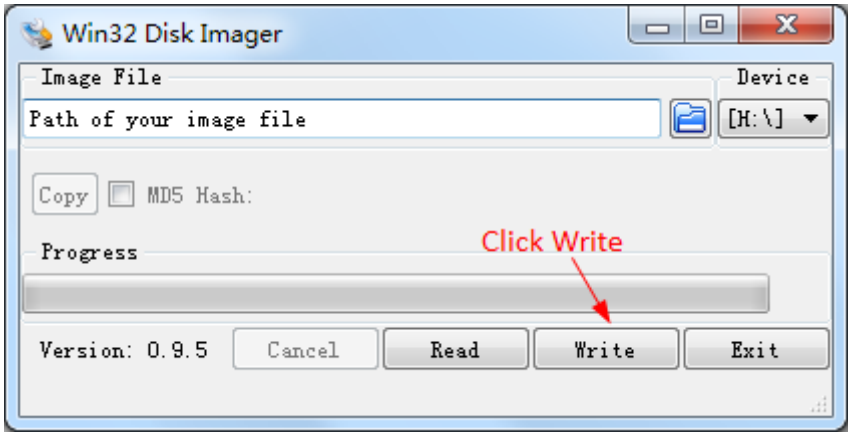
1. Firstly, you should prepare a SD card, which is no less than 8GB of capacity .
2. Then, download and install “Win32 Disk Imager” from: https://sourceforge.net/projects/win32diskimager/.



3. Select the system images file:



4. Click “Write” button to program the images:



4.1.2 Program the System Image into SD Card under Linux OS

In Ubuntu or Debian OS, you can use bmap-tool to program the image to SD Card. Here we use avnet-image-full-maaxboard-mini-20230711025928.rootfs.wic as an example:

1. Enter the following instructions in command line to check the SD Card ID, in this example is: sdc

```
$ ls /dev/sd*
/dev/sda /dev/sda2 /dev/sdb /dev/sdb2 /dev/sdc /dev/sdc2
/dev/sda1 /dev/sda5 /dev/sdb1 /dev/sdb5 /dev/sdc1
```

2. If SD Card is mounted, umount it.

```
$ sudo umount /dev/sdc1
$ sudo umount /dev/sdc2
```

3. Program the SD card with following instructions:

```
$ sudo dd if=avnet-image-full-maaxboard-mini-20230711025928.rootfs.wic
of=/dev/sdc bs=10M conv=fsync
$ sync
```

4.2 Program Images using *uuu*

USB0 (The lower one in USB interface HUB1) of MaaXBoard Mini supports programming mode. Connect USB0 and PC before power on the board. The system will enter programming mode. Then users could program the system image to the development board using uuu tool.

4.2.1 Preparation

4.2.1.1 Running Environment:

- Programming Tool: Universal Update Utility (Short as UUU), Download Link: https://github.com/NXPmicro/mfgtools/releases
- System Environment: Win10 64 bit OS or Ubuntu 64 bit OS, 16.14 LTS or later LTS version

4.2.1.2 Images:

Put the following image files into the download directory

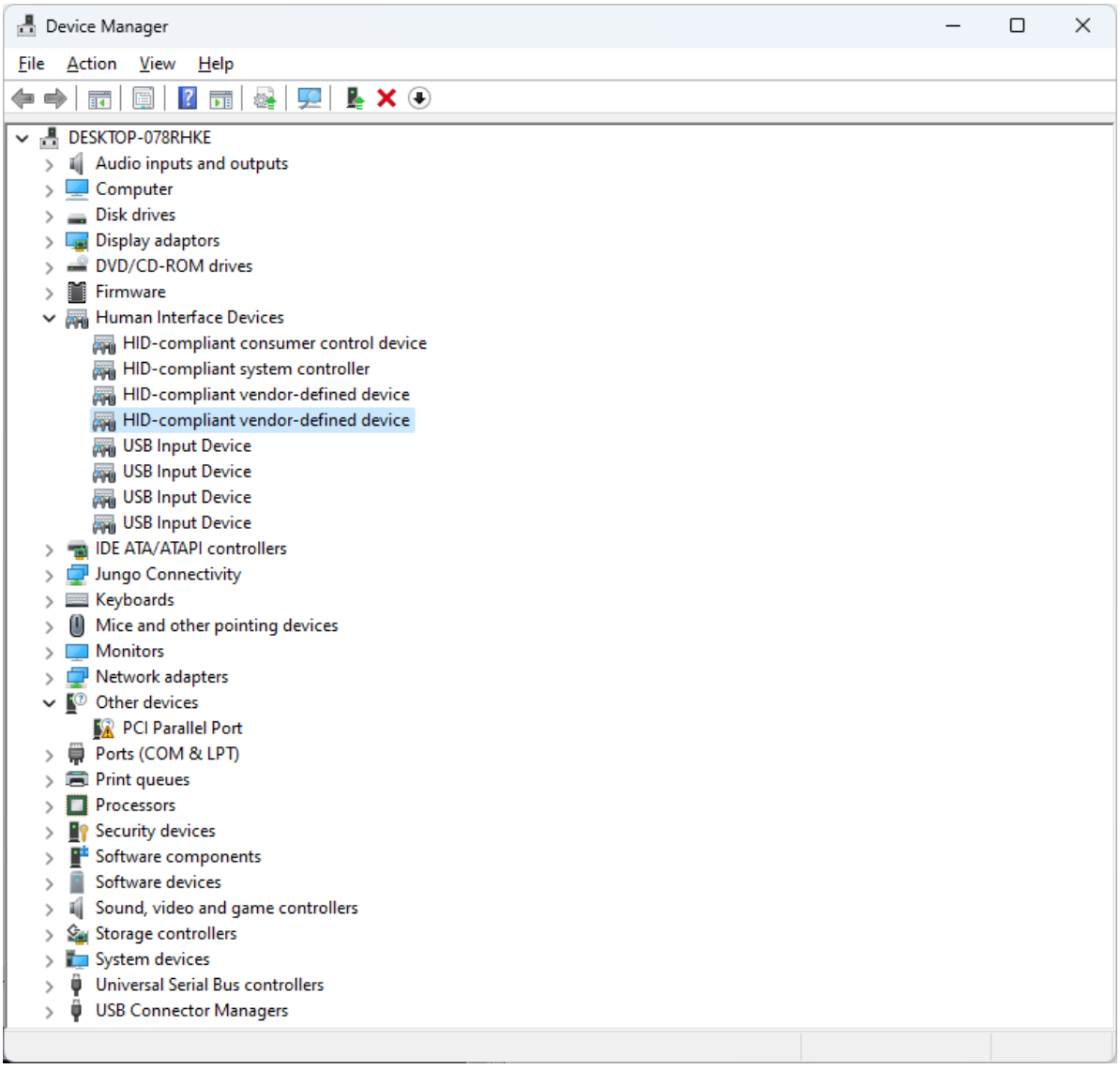
- MaaXBoard Mini Linux system image file, e.g: **avnet-image-full-maaxboard-mini-20230711025928.rootfs.wic**
- U-boot image file: **u-boot-imx8m-mini.imx** , usually the same version as the one contained in the system Image

4.2.2 Program the Image under Windows OS

4.2.2.1 Driver for Windows OS

In Windows OS, connect USB0 to PC, then power on the board. Check the device manager to see if need to install driver for the device:

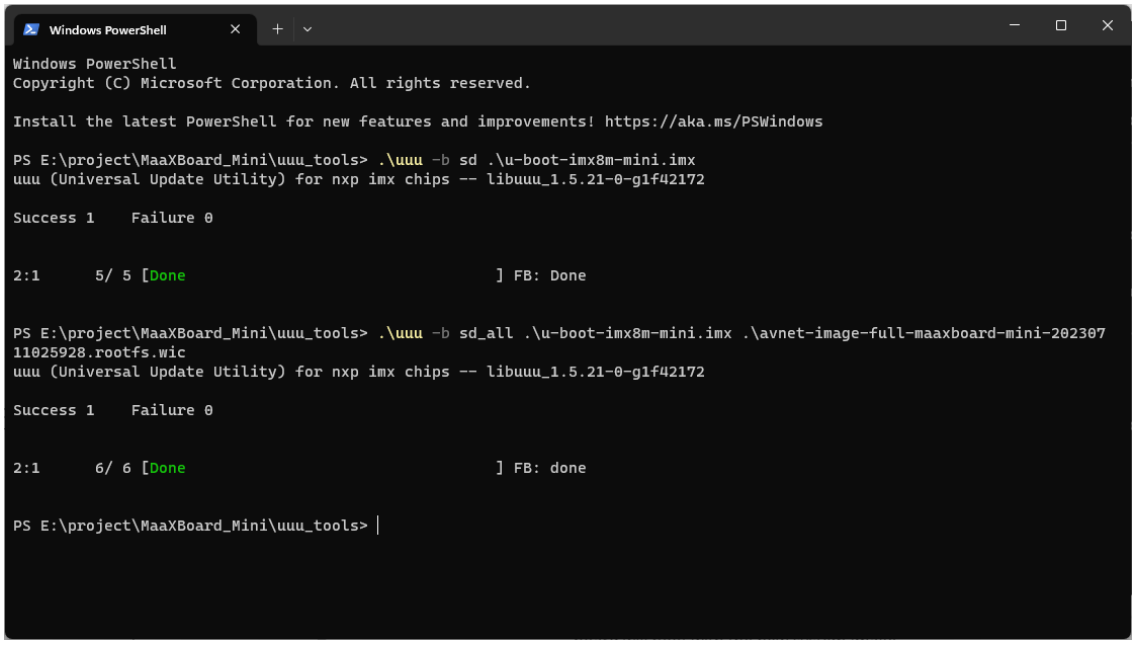
If the driver is installed normally, it will show:



4.2.2.2 Program the Image into SD Card under Windows OS

1. Connect USB0 (the lower one in USB 2.0 interface J2) and PC using USB type A cable.
2. Powered the board with a 5V, 2A, Type-C interface power (to J11).
3. Insert the SD card into the card slot J9.
4. Execute the following command to program the uboot image or the system image into the SD card.

```
.\uuu -b sd u-boot-imx8m-mini.imx
.\uuu -b sd_all u-boot-imx8m-mini.imx avnet-image-full-maaxboard-mini-20230711025928.rootfs.wic
```

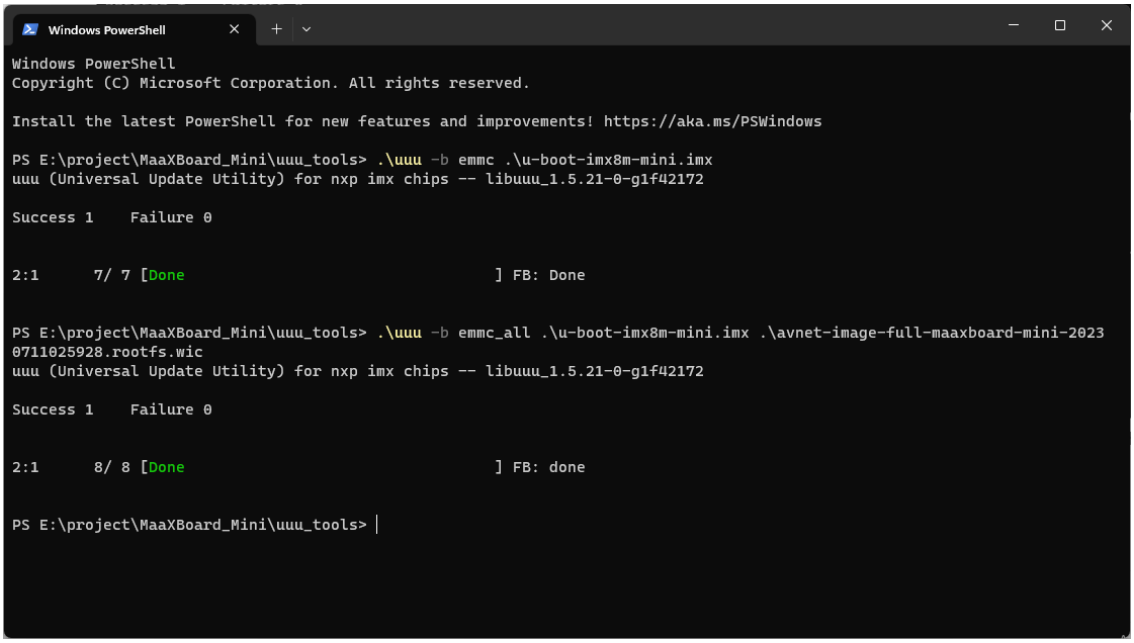


5. When programming finished, power down MaaXBoard Mini, disconnect the USB connection between USB0 and PC, power on the board again, then the board will boot with the new system image.

4.2.2.3 Program the Image into eMMC under Windows OS

1. Connect USB0 (the lower one in USB 2.0 interface J2) and PC using USB type A cable.
2. Powered the board with a 5V, 2A, Type-C interface power (to J11).
3. Execute the following commands to program the uboot image or the system image into the eMMC.

```
.\uuu -b emmc u-boot-imx8m-mini.imx
.\uuu -b emmc_all u-boot-imx8m-mini.imx avnet-image-full-maaxboard-mini-20230711025928.rootfs.wic
```



4. When programming finished, power down MaaXBoard Mini, disconnect the USB connection between USB0 and PC, power on the board again, the board will boot with the new system image.

4.2.3 Program the Image under Ubuntu OS

4.2.3.1 Program the Image into SD Card under Ubuntu OS

1. Connect USB0 (the lower one in USB 2.0 interface J2) and PC using USB type A cable.
2. Powered the board with a 5V, 2A, Type-C interface power (to J11).
3. Insert the SD card into the card slot J9.
4. Execute the following command to program the uboot image or the system image into the SD card.

```
uuu -b sd u-boot-imx8m-mini.imx
uuu -b sd_all u-boot-imx8m-mini.imx avnet-image-full-maaxboard-mini-
20230711025928.rootfs.wic
```

5. When programming finished, power down MaaXBoard Mini, disconnect the USB connection between USB0 and PC, power on the board again, then the board will boot with the new system image.

4.2.3.2 Program the Image into eMMC under Ubuntu OS

1. Connect USB0 (the lower one in USB 2.0 interface J2) and PC using USB type A cable.
2. Powered the board with a 5V, 2A, Type-C interface power (to J11).
3. Execute the following command to program the uboot image or the system image into the eMMC.

```
uuu -b emmc u-boot-imx8m-mini.imx
uuu -b emmc_all u-boot-imx8m-mini.imx avnet-image-full-maaxboard-mini-
20230711025928.rootfs.wic
```

4. When programming finished, power down MaaXBoard Mini, disconnect the USB connection between USB0 and PC, power on the board again, then the board will boot with the new system image.

Chapter 5 Appendix

5.1 Hardware Documents

For the detail hardware introduction, please refer to ***MaaXBoard Mini Hardware user manual.***

5.2 Software Documents

MaaXBoard Mini supports Yocto Linux system, for additional information, please refer to the following documents:

- ***MaaXBoard Mini Linux Yocto UserManual***

> - Describes how to boot MaaXBoard Mini and aspects of the BSP functionality (This document)

- ***MaaXBoard Mini Linux Yocto Development Guide***

> - Detailed guidance on how to rebuild the Linux system image

5.4 Contact Information

- Website: https://www.avnet.com/wps/portal/us/products/avnet-boards/avnet-board-families/maaxboard/maaxboard-mini