AVNET®

MANUFACTURING SERVICES

MaaXBoard Nano

Yocto Lite User Manual

V1.0

Copyright Statement:

- The MaaXBoard Nano single board computer and its related intellectual property are owned by Avnet Manufacturing Services.
- Avnet Manufacturing Services has the copyright of this document and reserves all rights. Any part of the document should not be modified, distributed or duplicated in any approach and form without the written permission issued by Avnet Manufacturing Services.

Disclaimer:

- Avnet Manufacturing Services does not take warranty of any kind, either expressed or implied, as to the program source code, software and documents provided along with the products, and including, but not limited to, warranties of fitness for a particular purpose; The entire risk as to the quality or performance of the program is with the user of products.

Revision History

Rev.	Description	Author	Date
V1.0	Initial version	Sandy	20201028

Catalog

Chapter 1Introduction

1.1 Package Content

The content of software release package is subject to the actual release sources. For the file structure and instructions, refer to the following table:

Release Folder tree

```
├──01Doc
|    |    MaaXBoard-Nano-Linux-Yocto-Lite-ReleaseNote-Vxx.pdf
|    └──EN
|              MaaXBoard-Nano-Hardware_UserManual-Vxx.pdf
|              MaaXBoard-Nano-Linux-Yocto-Lite-Development_Guide-Vxx.pdf
|              MaaXBoard-Nano-Linux-Yocto-Lite-UserManual-Vxx.pdf
└──04Linux Yocto Lite
|    ├──01LinuxSourceCode
|    |         MaaXBoard-Nano-LinuxSourceCode-Yocto-Lite-Vxx.rar
|    ├──02LinuxShipmentImage
|    |         MaaXBoard-Nano-LinuxShipmentImage-Yocto-Lite-Vxx.rar
|    └──03LinuxTools
|              MaaXBoard-Nano-LinuxTools-Vxx.rar
```

01Doc	**Description**
MaaXBoard-Nano-Linux-Yocto-Lite-ReleaseNote-Vxx.pdf	Release Note
MaaXBoard-Nano-Linux-Yocto-Lite-UserManual-Vxx.pdf	User Manual
MaaXBoard-Nano-Linux-Yocto-Lite-DevelopmentGuide-Vxx.pdf	Development Guide
01LinuxSourceCode	**Description**
MaaXBoard-Nano-LinuxSourceCode-Yocto-Lite-Vxx.rar	MaaXBoard Nano Linux Yocto source code package
02LinuxShipmentImage	**Description**
lite-image-maaxboard-nano-ddr4-1g-sdcard-*.rootfs.wic	Yocto image with firmware, system image file
03LinuxTools	**Description**
xxx	Other tools

1.2 Feature List

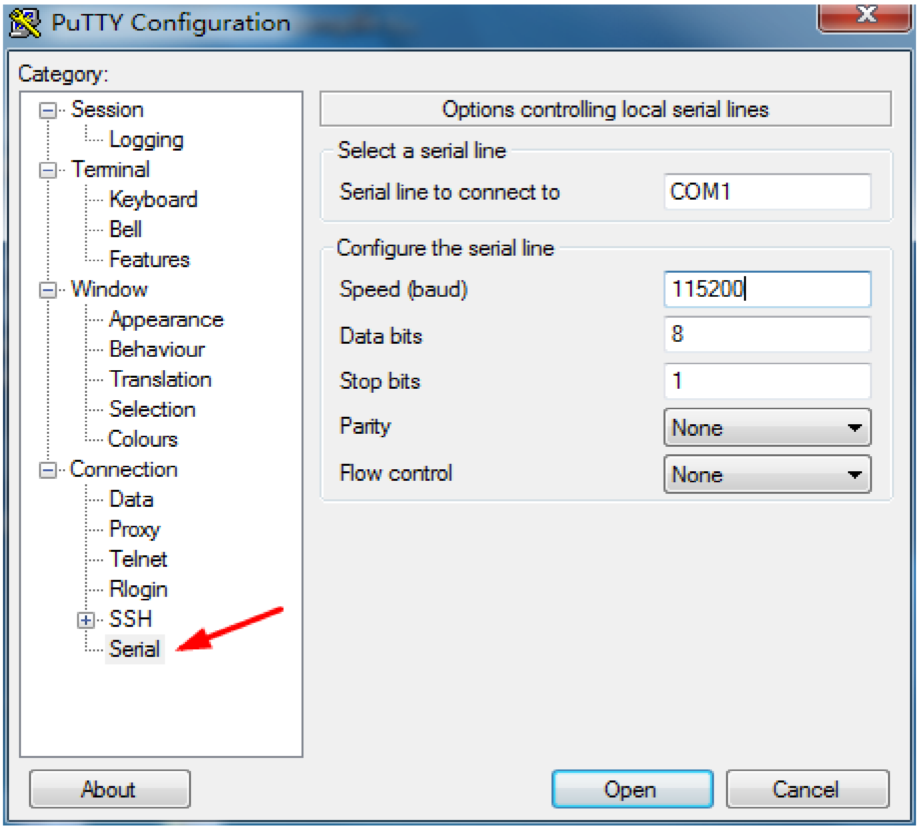
- Yocto version: zeus (3.0.4), based on NXP SDK version: imx-5.4.24-2.1.0
- U-Boot version: 2020.04
- Kernel version: 5.4.24
- Evaluation image: Yocto Lite Image
- Development based on NXP i.MX 8M Nano
- Micro SD boot
- eMMC boot
- 1 Gigabit Ethernet (RJ45)
- 4 x USB 2.0 Host
- 2 UART (TTL) include debug port
- External interfaces(I2C, UART,SPI and GPIO)
- WIFI & BLE 4.2
- MIPI-DSI display
- MIPI camera/USB Camera

Chapter 2 Quick Start

MaaXBoard Nano has burned the system image to eMMC by default, so we can boot from eMMC directly. If you need to use the latest system image, refer to Chapter 4 Burn or update the system Image. For the hardware connection and accessories details, please check the QSG.

2.1 Boot from eMMC

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



PuTTY Configuration

Options controlling local serial lines

Select a serial line

Serial line to connect to: COM1

Configure the serial line

Speed (baud): 115200

Data bits: 8

Stop bits: 1

Parity: None

Flow control: 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.
- Disconnect Boot Select Pin J3
- Powered the board with a 5V, 2A, Type-C interface power (to J10).
- When the system boot up, the serial terminal will print the following information:

```
NXP i.MX Release Distro 5.4-zeus maaxboard-nano ttymxc1
maaxboard-nano-ddr4-1g-sdcard login:
```

- Enter username as “root”, password as “avnet” to login.

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

2.2 Boot from SD Card

The boot process is **almost** the same with boot from SD card, but you need to execute the following 2 operations before power on the board:

- Insert the SD card into the card slot J10.
- Short Connect Boot Select Pin J3

2.3 Login system

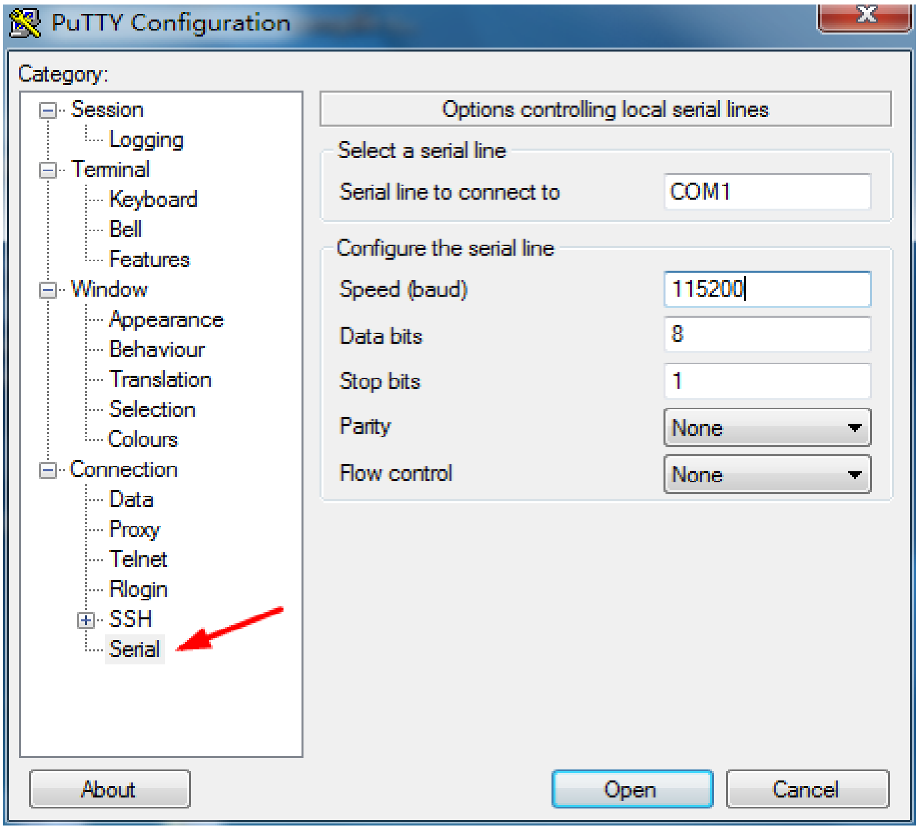
Yocto Lite system support the following login methods: login directly, login from debug serial, login from SSH.

2.3.1 Login Directly

Connect screen and keyboard to MaaXBoard Nano. When the system boot up, it will run the Weston Wayland Desktop Environment directly, click Wayland Terminal to operate the board.

2.3.2 Login from Debug Serial

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



PuTTY Configuration

Options controlling local serial lines

Select a serial line

Serial line to connect to: COM1

Configure the serial line

Speed (baud): 115200

Data bits: 8

Stop bits: 1

Parity: None

Flow control: 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”, password as “avnet” to login.

2.3.3 Login from SSH

Yocto OS install and startup SSH service automatically by default. Connect to internet, then login the system using SSH. Use scp to transfer files.

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

2.3.3.1 Preparation

In default situation, users are not allowed to login from SSH as root user, so you need to modify the configuration file or create a common user and login.

Modify methods:

Add following info to the configuration file **/etc/ssh/sshd_config**:

PermitRootLogin yes

Then, check the IP of MaaXBoard: The IP will be used in ssh login.

```
root@maaxboard-nano:~# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 06:87:b9:d0:fc:f5
          inet addr:192.168.2.108  Bcast:192.168.2.255  Mask:255.255.255.0
          inet6 addr: fe80::b2f3:a13d:a557:4c18/64 Scope:Link
          inet6 addr: fe80::487:b9ff:fed0:fcf5/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:46 errors:0 dropped:0 overruns:0 frame:0
          TX packets:54 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:5944 (5.8 KiB)  TX bytes:9217 (9.0 KiB)
```

2.3.3.2 Login Command line

In this example, the IP of MaaXBoard is 192.168.2.108, enter following command in command line window to connect: ssh root@192.168.2.108. Enter yes in the first connection, then enter password “avnet” to login, enter exit to logout.

```
embest@compiler:~$ ssh root@192.168.2.108
The authenticity of host '192.168.2.108 (192.168.2.108)' can't be established.
ECDSA key fingerprint is SHA256:fhQwJmGBD/Jmp15IfvB/gw6wZpnzDYKRt2LGQs1HEpE.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.2.108' (ECDSA) to the list of known hosts.
root@192.168.2.108's password:
Last login: Mon Oct 26 04:48:49 2020
```

```
root@maaxboard-nano:~#
```

To copy files using scp command, enter the following command and password:

Copy file from computer to MaaXBoard:

```
embest@compiler:~$ scp test.txt root@192.168.2.108:/root/1.txt
```

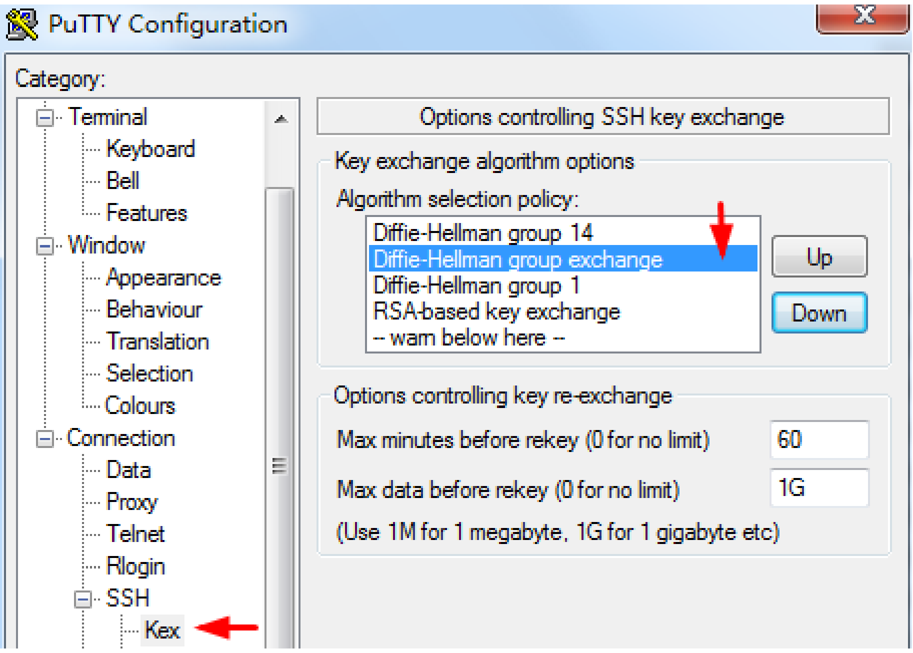
Copy file from MaaXBoard to computer:

```
embest@compiler:~$ scp root@192.168.2.108:/root/1.txt ./1.txt
```

2.3.3.3 PuTTY

PuTTY support SSH, setting method as follows:

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



Options controlling SSH key exchange

Key exchange algorithm options

Algorithm selection policy:

Diffie-Hellman group 14
Diffie-Hellman group exchange
Diffie-Hellman group 1
RSA-based key exchange
-- warn below here --

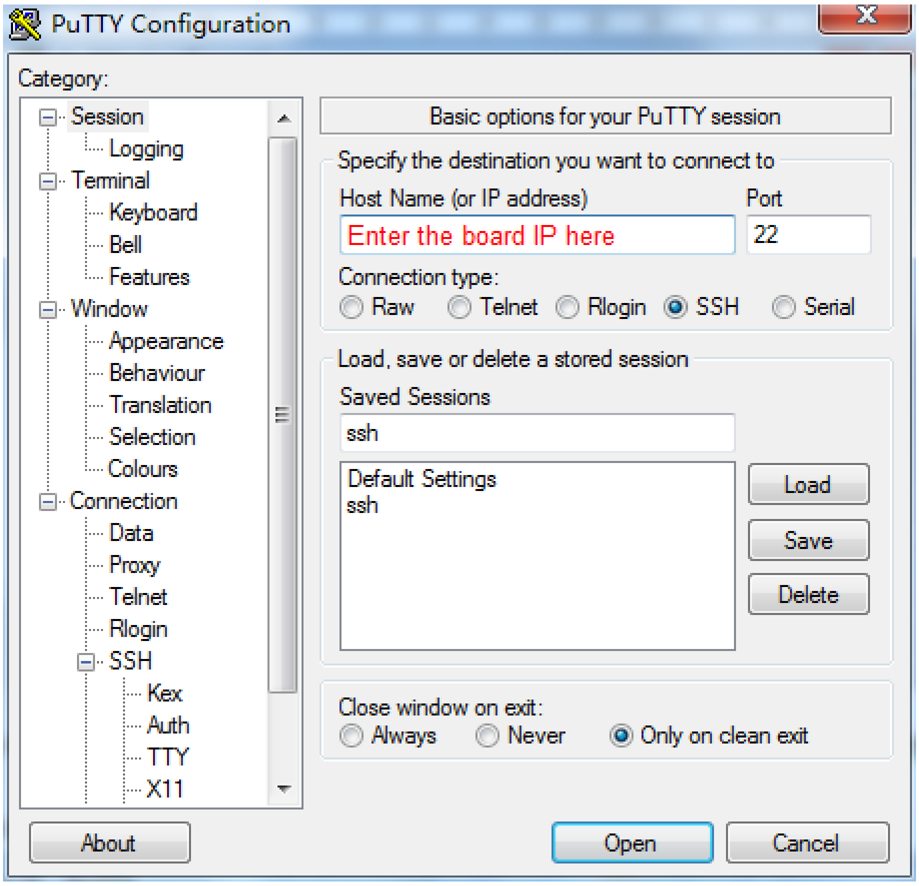
Options controlling key re-exchange

Max minutes before rekey (0 for no limit) 60

Max data before rekey (0 for no limit) 1G

(Use 1M for 1 megabyte, 1G for 1 gigabyte etc)

2. In Session, enter IP address, e.g. 192.168.2.108, port 22, and Connection type SSH, then click Open.



Specify the destination you want to connect to

Host Name (or IP address): Enter the board IP here

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

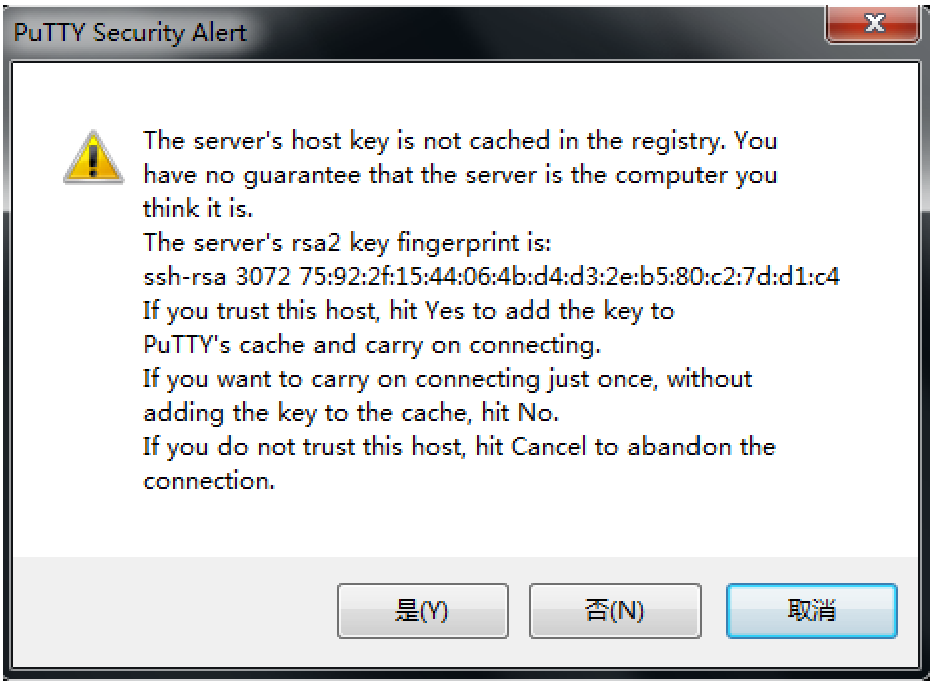
Load, save or delete a stored session

Saved Sessions: ssh

Default Settings
ssh

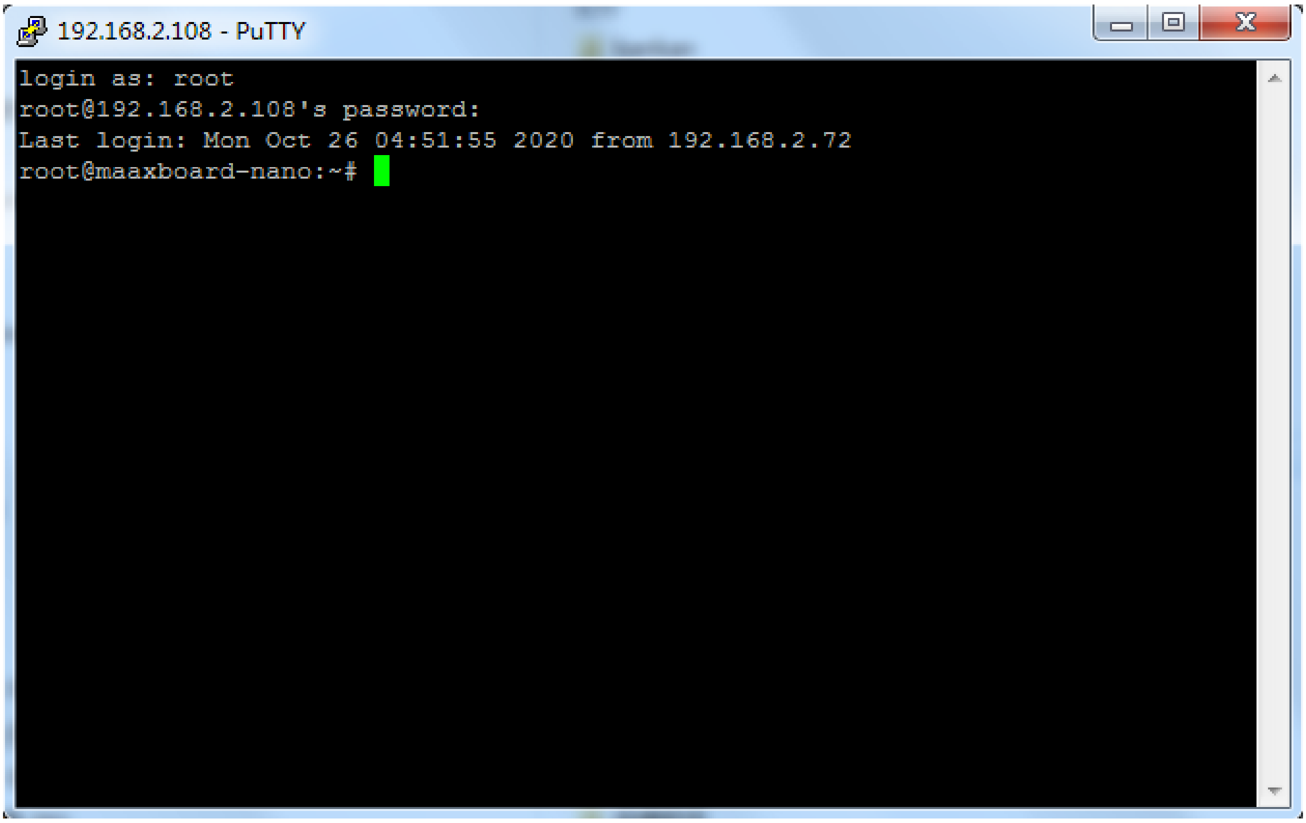
Close window on exit: ○ Always ○ Never ● Only on clean exit

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



The server's host key is not cached in the registry. You have no guarantee that the server is the computer you think it is.
The server's rsa2 key fingerprint is:
ssh-rsa 3072 75:92:2f:15:44:06:4b:d4:d3:2e:b5:80:c2:7d:d1:c4
If you trust this host, hit Yes to add the key to PuTTY's cache and carry on connecting.
If you want to carry on connecting just once, without adding the key to the cache, hit No.
If you do not trust this host, hit Cancel to abandon the connection.

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



```
login as: root
root@192.168.2.108's password:
Last login: Mon Oct 26 04:51:55 2020 from 192.168.2.72
root@maaxboard-nano:~#
```

Chapter 3Feature 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 USER LED

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

Light out LED:

```
root@maaxboard-nano:~# echo 0 | tee /sys/class/leds/usr_led/brightness
root@maaxboard-nano:~# echo 0 | tee /sys/class/leds/sys_led/brightness
```

Light up LED:

```
root@maaxboard-nano:~# echo 1 | tee /sys/class/leds/usr_led/brightness
root@maaxboard-nano:~# echo 1 | tee /sys/class/leds/sys_led/brightness
```

3.2 Button

MaaXBoard Nano support 3 button: USR_KEY and PWR.

1. Test USR_KEY button with following instructions:

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

```
root@maaxboard-nano:~# evtest
No device specified, trying to scan all of /dev/input/event*
Available devices:
/dev/input/event0:       30370000.snvs:snvs-powerkey
/dev/input/event1:       fts_ts
/dev/input/event2:       gpio_keys
Select the device event number [0-2]: 2
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 256 (BTN_0)
```

```
Properties:
Testing ... (interrupt to exit)
Event: time 1603702232.929224, type 1 (EV_KEY), code 256 (BTN_0), value 1
Event: time 1603702232.929224, -------------- SYN_REPORT ------------
Event: time 1603702233.205239, type 1 (EV_KEY), code 256 (BTN_0), value 0
Event: time 1603702233.205239, -------------- SYN_REPORT ------------
```

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@maaxboard-nano:~# 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:        bd718xx-pwrkey
Select the device event number [0-2]: 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 1596184675.913800, type 1 (EV_KEY), code 116 (KEY_POWER), value 1
Event: time 1596184675.913800, -------------- SYN_REPORT ------------
Event: time 1596184676.169726, type 1 (EV_KEY), code 116 (KEY_POWER), value 0
Event: time 1596184676.169726, -------------- SYN_REPORT ------------
```

3.3 Displayer

MaaXBoard Nano supports MIPI-DSI screen.

Users can connect the screen to the board 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 Nano file system.

Screen Type	Screen Resolution	Interface
MIPI-DSI	1280*720	J5

Display device could be chosen by modify the fdt_file 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@maaxboard-nano:~# mkdir mount
root@maaxboard-nano:~# mount /dev/mmcblk1p1 mount/
root@maaxboard-nano:~# nano mount/uEnv.txt
```

3.3.1 MIPI-DSI Screen

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

```
fdt_file=maaxboard-nano-mipi.dtb
```

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@maaxboard-nano:~# echo 7 > /sys/class/backlight/backlight/brightness
```

3.4 Touchscreen

The MIPI-DSI support touch screen. If the desktop environment run automatically, user could open the Wayland Terminal, click or drag the window, etc. You could also use evtest command to test it.

```
root@maaxboard-nano:~# evtest /dev/input/touchscreen0
Input driver version is 1.0.1
Input device ID: bus 0x18 vendor 0x0 product 0x0 version 0x0
Input device name: "fts_ts"
```

```
Supported events:
  Event type 0 (EV_SYN)
  Event type 1 (EV_KEY)
    Event code 102 (KEY_HOME)
    Event code 139 (KEY_MENU)
    Event code 158 (KEY_BACK)
    Event code 330 (BTN_TOUCH)
  Event type 3 (EV_ABS)
    Event code 47 (ABS_MT_SLOT)
      Value      0
      Min        0
      Max        9
    Event code 48 (ABS_MT_TOUCH_MAJOR)
      Value      0
      Min        0
      Max      255
    Event code 53 (ABS_MT_POSITION_X)
      Value      0
      Min        0
      Max      720
    Event code 54 (ABS_MT_POSITION_Y)
      Value      0
      Min        0
      Max     1280
    Event code 57 (ABS_MT_TRACKING_ID)
      Value      0
      Min        0
      Max    65535
    Event code 58 (ABS_MT_PRESSURE)
      Value      0
      Min        0
      Max      255
Properties:
  Property type 1 (INPUT_PROP_DIRECT)
Testing ... (interrupt to exit)
Event: time 1597392253.449259, type 3 (EV_ABS), code 57 (ABS_MT_TRACKING_ID), value 19
Event: time 1597392253.449259, type 3 (EV_ABS), code 58 (ABS_MT_PRESSURE), value 63
Event: time 1597392253.449259, type 3 (EV_ABS), code 53 (ABS_MT_POSITION_X), value 31
```

```
Event: time 1597392253.449259, type 3 (EV_ABS), code 54 (ABS_MT_POSITION_Y), value 1024
Event: time 1597392253.449259, type 1 (EV_KEY), code 330 (BTN_TOUCH), value 1
Event: time 1597392253.449259, -------------- SYN_REPORT ------------
Event: time 1597392253.515228, type 3 (EV_ABS), code 57 (ABS_MT_TRACKING_ID), value -1
Event: time 1597392253.515228, type 3 (EV_ABS), code 58 (ABS_MT_PRESSURE), value 0
Event: time 1597392253.515228, type 1 (EV_KEY), code 330 (BTN_TOUCH), value 0
Event: time 1597392253.515228, -------------- SYN_REPORT ------------
```

3.5 Audio

MaaXBoard Nano support on-board audio output interface, 4 on-board MIC, USB audio device and Bluetooth audio device.

3.5.1 Check Device ID

Before use audio device to play or record, you need to check the device id.

The on-board audio output interface, on-board MIC and USB audio device could use **arecord -l** and **aplay -l** to check the device ID.

In default, you will see the following device:

```
root@maaxboard-nano:~# arecord -l
**** List of CAPTURE Hardware Devices ****
card 1: imxaudiomicfil [imx-audio-micfil], device 0: micfil hifi snd-soc-dummy-dai-0 []
  Subdevices: 1/1
  Subdevice #0: subdevice #0
root@maaxboard-nano:~# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: cs4344audio [cs4344-audio], device 0: HiFi cs4344-hifi-0 []
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: cs4344audio [cs4344-audio], device 1: HiFi-ASRC-FE (*) []
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

3.5.1.1 On-board Audio Output Interface

The audio output interface J9 is the default audio output device of MaaXBoard Nano. Connect the audio device such as 3.5mm headset to J9 to use it. Use command **aplay -l** to check that the device id is 0, device name is cs4344audio.

3.5.1.2 On-board MIC

MaaXBoard Nano support 4 on-board MICs, which could record the audio file use 4 channels. Use command **arecord -l** to check that the device id is 1, device name is imxaudiomicfil.

3.5.1.3 USB Audio Device

MaaXBoard Nano could support USB audio device (which do not need specified driver) to play audio. You can record and play audio from USB audio device. Use command **arecord -l** and **aplay -l** to check that the device id is 2.

3.5.1.4 Bluetooth Audio

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

Pair and connect the Bluetooth device use pulseaudio, you could use command **aplay -L** to check the audio device.

```
root@maaxboard-nano:~# aplay -L
null
    Discard all samples (playback) or generate zero samples (capture)
pulse
    PulseAudio Sound Server
sysdefault:CARD=cs4344audio
    cs4344-audio,
    Default Audio Device
```

Currently we only support part of Bluetooth audio device.

3.5.2 Record Audio

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

```
root@maaxboard-nano:~# arecord -f S16_LE -r 48000 -c 1 -Dhw:1 audio.wav
```

Note: press Ctrl+C ro exit record.

In above command, **S16_LE** means audio format, -r 48000 means the sample rate of the audio file is 48KHz, -c 1 means use 1 channel to record the audio, -Dhw:1 means use audio card 1(device id of the on-board MIC) to record. Change those parameters according to your device.

Most of headphone only supports 1 record channel, so we use -c1. The on-board MIC support 4 channels, so we could use -c 4.

```
root@maaxboard-nano:~# arecord -f S16_LE -r 48000 -c4 -Dhw:1 audio.wav
```

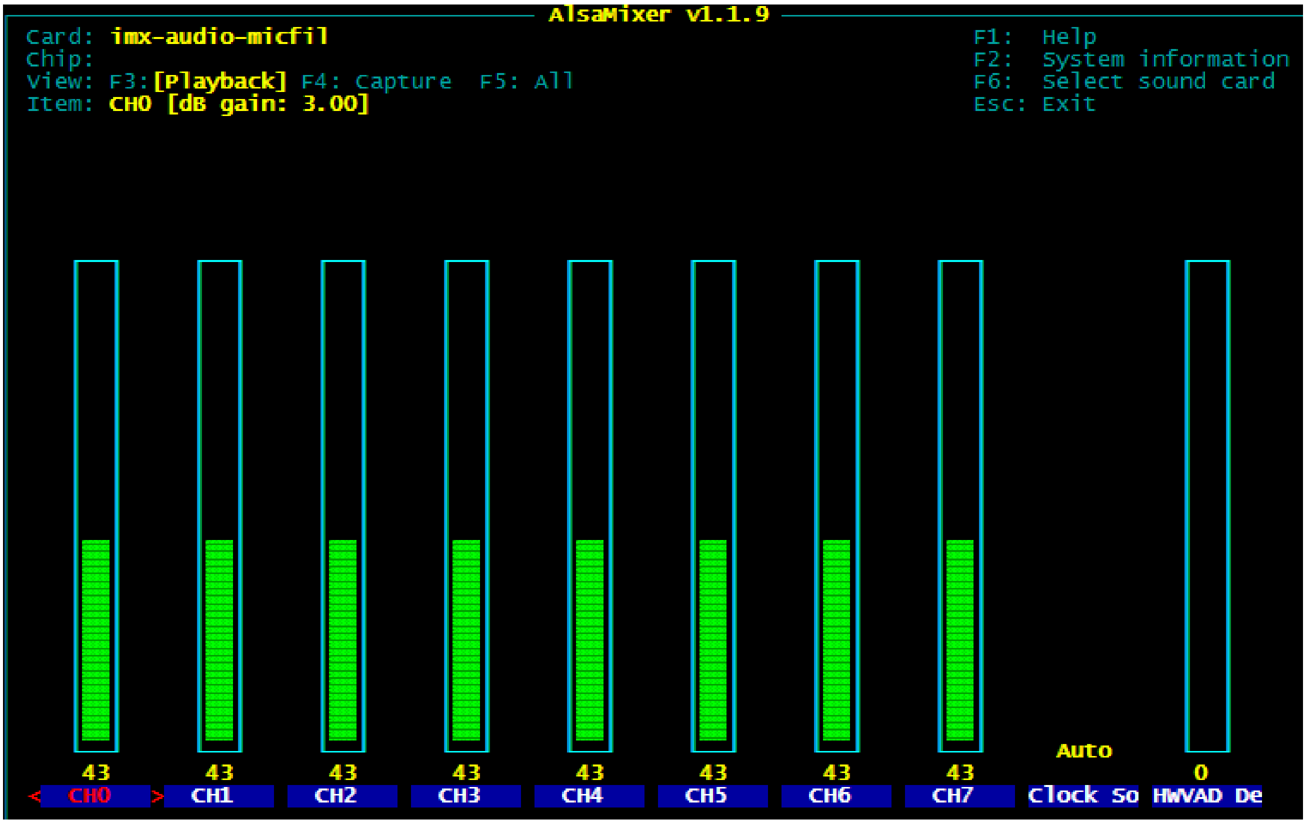
You can also record audio from the on-board MIC and play it from on-board audio output interface, use the following command: here we use -c 2 because the audio output interface only support 2 channels.

```
root@maaxboard-nano:~# arecord -f S16_LE -r 48000 -c 2 -Dhw:1 | aplay -Dhw:0
```

To adjust the volume of record audio, use the following command to open a GUI

```
root@maaxboard-nano:~# alsamixer -c1
```

Use the Up-down-Left-Right button to adjust the volume of different channel, press Esc button to exit.



```
                              AlsaMixer v1.1.9
Card: imx-audio-micfil                                  F1:  Help
Chip:                                                   F2:  System information
View: F3:[Playback] F4: Capture  F5: All                F6:  Select sound card
Item: CH0 [dB gain: 3.00]                               Esc: Exit

                                                     Auto
  43     43     43     43     43     43     43     43            0
< CH0 >  CH1    CH2    CH3    CH4    CH5    CH6    CH7  Clock So HWVAD De
```

3.5.3 Play Audio File

```
root@maaxboard-nano:~# aplay audio_sample.wav
root@maaxboard-nano:~# gst-play-1.0 audio_sample.wav
root@maaxboard-nano:~# mpg123 audio_sample.mp3
```

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

When use above command. It will play audio from the default device, on-board audio output interface.

To play the audio from specified device, use the following:

```
root@maaxboard-nano:~# aplay -Dhw:2 audio.wav
```

Here we use -Dhw:2, which means use audio card 2 to play the audio.

To play audio from Bluetooth audio device, use the following command after Bluetooth audio device is connected.

```
root@maaxboard-nano:~# aplay -D pulse test.wav
```

Note: Most of audio output device only support 2 channels.

3.6 Video

Yocto system support play video file in mp4 format, the largest support resolution is 1080p, use the following command:

```
root@maaxboard-nano:~# gst-play-1.0 embest.mp4
```

3.7 Camera

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

3.7.1 Check Device ID

```
root@maaxboard-nano:~# ls /dev/video*
/dev/video0    /dev/video1
```

In default, MIPI-CSI camera is /dev/video0, USB Camera is /dev/video1. The device ID will be used in following command.

3.7.2 Preview

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

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

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

3.7.3 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@maaxboard-nano:~# 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.

3.7.4 Record Video

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

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

3.8 Gigabit Ethernet Interface

Connect the network cable to J7, enter the following instructions to set the IP address:

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

3.8.1 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@maaxboard-nano:~# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 06:87:b9:d0:fc:f5
          inet addr:192.168.2.108  Bcast:192.168.2.255  Mask:255.255.255.0
          inet6 addr: fe80::b2f3:a13d:a557:4c18/64 Scope:Link
          inet6 addr: fe80::487:b9ff:fed0:fcf5/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:46 errors:0 dropped:0 overruns:0 frame:0
          TX packets:54 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:5944 (5.8 KiB)  TX bytes:9217 (9.0 KiB)
```

3.8.2 Set Static IP

If you need to set a static IP, execute the following 2 steps:

1. Set the static MAC for the Board: modify the ethaddr 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@maaxboard-nano:~# mkdir mount
root@maaxboard-nano:~# mount /dev/mmcblk1p1 mount/
root@maaxboard-nano:~# nano mount/uEnv.txt
```

For example: `ethaddr=cb:3a:e1:44:62:2d`

2. Set Static IP info: use **nano** command to modify **/etc/dhcpcd.conf**, add following info.

```
interface eth0
static ip_address=192.168.2.100/24
static routers=192.168.2.1
static domain_name_servers=192.168.2.252 8.8.8.8
```

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 Storage

MaaXBoard support on-board eMMC, Flash, SD Card interface, it could boot from SD Card or eMMC.

3.9.1 SD Card

The storage node for SD Card is /dev/mmcblk1.

To boot from SD Card, short connect Boot Select Pin J3 before power on the board.

3.9.2 eMMC

The size of on-board eMMC is 16GB.The storage node for eMMC is /dev/mmcblk2.

3.9.3 Flash

The size of on-board Flash is 32MB.The storage node for Flash is /dev/mtd0.

To program Flash, use the following command:

```
root@maaxboard-nano:~# mount -t jffs2 /dev/mtdblock0 /mnt
root@maaxboard-nano:~# echo "Hello, flash" > /mnt/hi.txt
root@maaxboard-nano:~# cat /mnt/hi.txt
root@maaxboard-nano:~# umount /mnt
```

To erase the Flash, use flash_erase command, for example:

```
root@maaxboard-nano:~# flash_erase /dev/mtd0 0 0
```

3.10 USB 2.0 Interface

MaaXBoard Nano support 4 USB 2.0 Host Interfaces.

3.10.1 USB Host

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

```
[   108.102562] usb 1-1.3: new high-speed USB device number 3 using ci_hdrc
[   108.154161] usb-storage 1-1.3:1.0: USB Mass Storage device detected
[   108.161226] scsi host0: usb-storage 1-1.3:1.0
[   109.184992] scsi 0:0:0:0: Direct-Access      Kingston DataTraveler 3.0        PQ: 0 ANSI: 6
[   109.196299] sd 0:0:0:0: [sda] 30218842 512-byte logical blocks: (15.5 GB/14.4 GiB)
[   109.204707] sd 0:0:0:0: [sda] Write Protect is off
[   109.210058] sd 0:0:0:0: [sda] Write cache: disabled, read cache: enabled, doesn't support DPO
or FUA
[   109.249451]  sda: sda1
[   109.256908] sd 0:0:0:0: [sda] Attached SCSI removable disk
```

Execute the following instructions on the serial terminal:

```
root@maaxboard-nano:~# ls /dev/sd*
/dev/sda   /dev/sda1
```

The storage node for U disk is /dev/sda1, uses could mount the storage device to the file system to read and write data.

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

3.11 Wi-Fi

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

3.11.1 Enable Wi-Fi

```
root@maaxboard-nano:~# ifconfig mlan0 up
```

3.11.2 Connect Wi-Fi

Execute the following instructions on the serial terminal to search Wi-Fi network:

```
root@maaxboard-nano:~# iwlist mlan0 scan | grep SSID
```

It will print the information for all available network.

If you want to search for specific Wi-Fi network, use the following:

```
root@maaxboard-nano:~# iwlist mlan0 scan essid Embest_WiFi
wlan: SCAN COMPLETED: scanned AP count=5
mlan0      Scan completed :
          Cell 01 - Address: 80:81:00:56:2F:88
                    ESSID:"Embest_WiFi"
                    Mode:Master
                    Frequency=2.437 GHz (Channel 6)
                    Quality:3/5  Signal level:-66 dBm  Noise level:-96 dBm
                    Encryption key:on
                    Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 6 Mb/s
                              9 Mb/s; 12 Mb/s; 18 Mb/s; 24 Mb/s; 36 Mb/s
                              48 Mb/s; 54 Mb/s
                    Extra:Beacon interval=222
                    IE: IEEE 802.11i/WPA2 Version 1
                        Group Cipher : TKIP
                        Pairwise Ciphers (2) : CCMP TKIP
                        Authentication Suites (1) : PSK
                    IE: WPA Version 1
                        Group Cipher : TKIP
                        Pairwise Ciphers (2) : CCMP TKIP
                        Authentication Suites (1) : PSK
                    IE: Unknown:
DD180050F2020101800003A4000027A4000042435E0062322F00
                    Extra:band=bg
```

Connect Wi-Fi network:

```
root@maaxboard-nano:~# nano /etc/wpa_supplicant.conf
```

Add following info into this file:

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

Then execute the following command:

```
root@maaxboard-nano:~# wpa_supplicant -B -i mlan0 -c /etc/wpa_supplican
Successfully initialized wpa_supplicant
```

If the connection succeeds, it will print the following info:

```
IPv6: ADDRCONF(NETDEV_CHANGE): mlan0: link becomes ready
```

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

```
root@maaxboard-nano:~# ping www.baidu.com -I mlan0
PING www.a.shifen.com (103.235.46.39) 56(84) bytes of data.
64 bytes from 103.235.46.39: icmp_seq=1 ttl=50 time=122 ms
Auto Connect Wi-Fi Network
```

3.11.3 Auto Connect Wi-Fi

To connect Wi-Fi after reboot automatically, use the following method:

Prepare config file:

```
mkdir -p /etc/wpa_supplicant
cp /etc/wpa_supplicant.conf /etc/wpa_supplicant/wpa_supplicant-mlan0.conf
```

Note: Add network info to /etc/wpa_supplicant/wpa_supplicant-mlan0.conf, refer to above chapter.

```
systemctl start wpa_supplicant@mlan0
systemctl enable wpa_supplicant@mlan0
```

3.12 Bluetooth 4.2

System will initialize the Bluetooth Module automatically; use the following instruction to connect:

3.12.1 Connect Bluetooth Device

Use **bluetoothctl** to connect Bluetooth Device:

```
root@maaxboard-nano:~# pulseaudio -D -v
root@maaxboard-nano:~# bluetoothctl
[bluetooth]# power on
[bluetooth]# pairable on
[bluetooth]# agent on
[bluetooth]# default-agent
```

Make the MaaXBoard 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
```

Exit **bluetoothctl**.

```
[Mi Sports BT Earphones Basic]# quit
```

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

3.12.2 Send Files

Run the OBEXD daemon and connect to the target Bluetooth device

```
root@maaxboard-nano:~# export $(dbus-launch)
root@maaxboard-nano:~# /usr/libexec/bluetooth/obexd -r /home/root -a -d & obexctl
[obex]# connect 94:87:E0:DF:90:2D
[94:87:E0:DF:90:2D]# send /home/root/1.txt
```

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 Nano supports 2 UART interface.

MaaXBoard Nano (CPU)	Interface Type
UART2	UART TTL (Debug Interface)
UART4	UART TTL

3.13.1 UART 4

In the Yocto system, the node for UART4 is /dev/ttymxc3. Users could also write their own applications to control the uart.

3.14 Control 40 Pin 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

GPIO number	PINMUX	Function	PIN	PIN	Function	PINMUX	GPIO number
		3.3V	1	2	5V		
145	GPIO5_IO17	SDA1	3	4	5V		
146	GPIO5_IO18	SCL1	5	6	GND		
130	GPIO5_IO2	GPIO	7	8	UART_TX	GPIO5_IO25	153
		GND	9	10	UART_RX	GPIO5_IO24	152
130	GPIO5_IO2	GPIO	11	12	GPIO/PCM_CLK	GPIO5_IO0	128
125	GPIO4_IO29	GPIO	13	14	GND		
124	GPIO4_IO28	GPIO	15	16	GPIO	GPIO5_IO28	156
		3.3V	17	18	GPIO	GPIO5_IO29	157
135	GPIO5_IO7	MOSI	19	20	GND		
136	GPIO5_IO8	MISO	21	22	GPIO	GPIO2_IO8	40
134	GPIO5_IO6	SCLK	23	24	CE0	GPIO5_IO9	137
		GND	25	26	CE1	GPIO5_IO5	133
147	GPIO5_IO19	SDA0	27	28	SCL0	GPIO5_IO18	146
15	GPIO1_IO15	GPIO/GPCLK	29	30	GND		
1	GPIO1_IO1	GPIO/GPCLK	31	32	GPIO/PWM1	GPIO5_IO4	132

131	GPIO5_IO3	GPIO/PWM2	33	34	GND		
127	GPIO4_IO31	GPIO/PCM_FS	35	36	GPIO	GPIO2_IO11	43
41	GPIO2_IO9	GPIO	37	38	GPIO/PCM_DIN	GPIO4_IO30	126
		GND	39	40	GPIO/PCM_DOUT	GPIO5_IO1	129

Here we use PIN 11 as an example:

1. In above table, find the GPIO Number of PIN11, which is 130.

2. Set the function of Pin11 to be GPIO output.

```
root@maaxboard-nano:~# echo 130 >/sys/class/gpio/export
root@maaxboard-nano:~# echo out >/sys/class/gpio/gpio130/direction
```

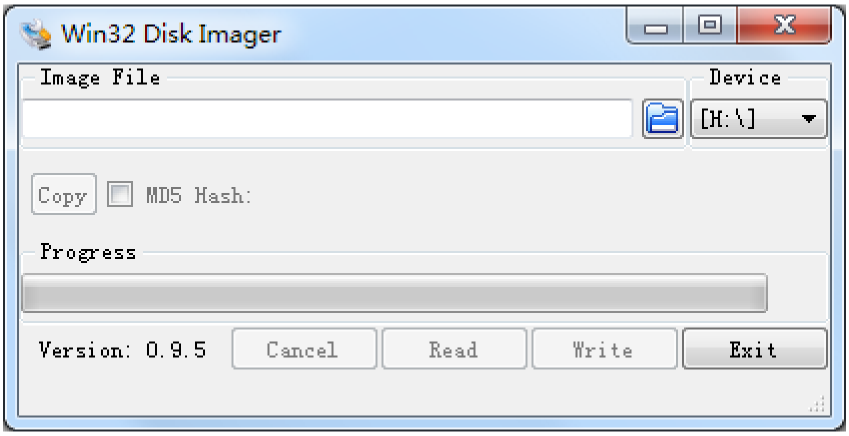
3. Set the level of Pin 11, 0 means low, 1 means high.

```
root@maaxboard-nano:~# echo 1 >/sys/class/gpio/gpio130/value
```

Chapter 4Burn or update the system Image

4.1 Burn the System Image to SD Card under Windows OS

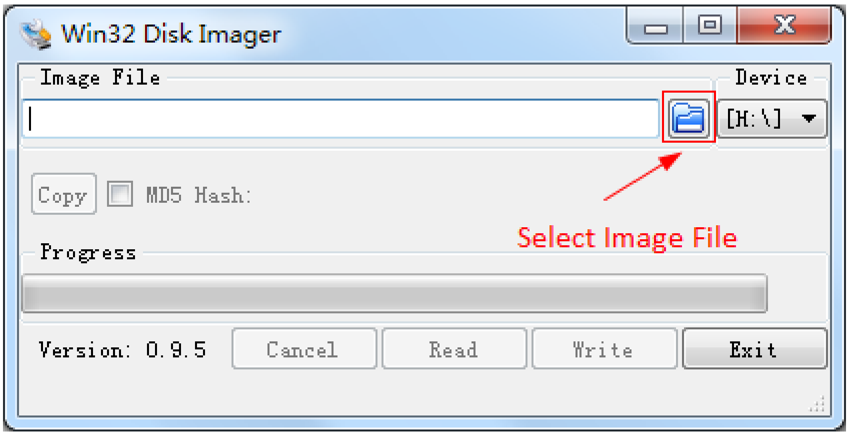
1. Firstly, you should prepare a SD card, which is no less than 8GB.
2. Then, download and install “Win32 Disk Imager” from: https://sourceforge.net/projects/win32diskimager/.



Device: [H:\]

Version: 0.9.5

3. Select the system images file:



Device: [H:\]

Version: 0.9.5

Select Image File

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

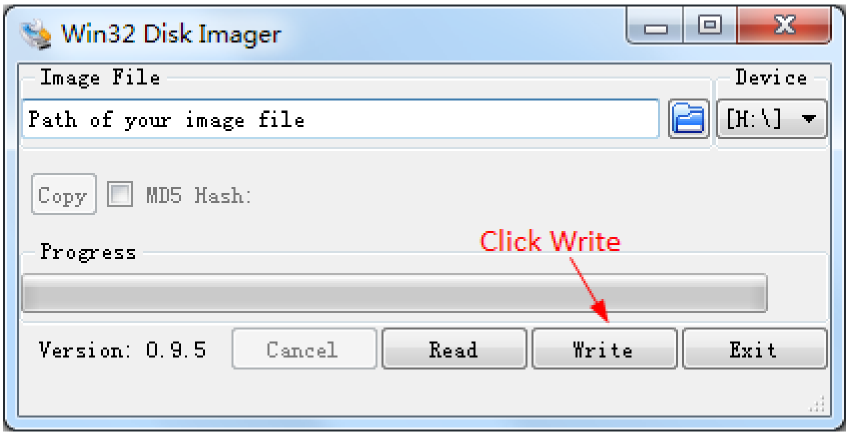


Image File: Path of your image file

Device: [H:\]

Version: 0.9.5

Click Write

4.2 Burn the System Image to SD Card under Linux OS

In Ubuntu or Debian OS, you can use bmap-tool to burn the image to SD Card. Here we use lite-image-maaxboard-nano-ddr4-1g-sdcard-20201116084413.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 automatically, umount it.

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

3. Burn the SD card with following instructions:

```
$ sudo dd if= lite-image-maaxboard-nano-ddr4-1g-sdcard*.rootfs.wic of=/dev/sdc bs=10M
conv=fsync
$ sync
```

If you get the bz2 file after compilation, use **bunzip2** command to extract.

```
$ bunzip2 lite-image-maaxboard-nano-ddr4-1g-sdcard*.rootfs.wic.bz2
```

4.3 Update System Image in eMMC

Here we use lite-image-maaxboard-nano-ddr4-1g-sdcard-20201116084413.rootfs.wic as an example: Extract the image file, copy it to a U-disk, refer to 2.2, boot the system from SD Card, and then plug the U disk to USB interface:

Execute the following instructions on the serial terminal:

```
root@maaxboard-nano:~# mount /dev/sda1 /mnt/
root@maaxboard-nano:~# dd
if=/mnt/lite-image-maaxboard-nano-ddr4-1g-sdcard-20201116084413.rootfs.wic of=/dev/mmcblk2
```

Note: Burn the EMMC takes a long time, please wait patiently.

Power off the board after the burning finished, disconnect J3 power on the board to boot from EMMC:

Chapter 5 Appendix

5.1 Hardware

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

5.2 Software

MaaXBoard Nano supports Linux Yocto system system, for the detail software introduction, please refer to related user manual.

- **Linux**
  - MaaXBoard Nano Linux Software Release Note
  - MaaXBoard Nano Linux Software User Manual
  - MaaXBoard Nano Linux Software Development Guide

5.3 Verified Device List

Unless otherwise specified, peripherals supported by this software, such as HDMI displays, Wi-Fi devices, etc., are compatible with common types of devices. However, we provide the following list of verified peripheral devices, for customers' reference.

If you meet problems in using other devices, contact the technical support.

- MIPI LCD: MaaXBoard DISPLAY_MIPI
- MIPI Camera: MaaXBoard Camera
- USB Camera: QQSJ-8810, Logitech C270
- Bluetooth Headphone: QCY-T1, EDIFIER W25BT, Mi Sports BT Earphones Basic
- Wireless Router：HUAWEI WS5200, gee HC5861, TL-WDR5620
- U Disk: SanDisk SDCZ880-128G, Kingston DataTraveler G4, WD WDBYVG0020BBK 2TB
- SD：Raspberry Pi 16G, SanDisk Ultra 16G/64G
- Power Supply: Pisen TypeC, ZMI TypeC

Chapter 6Technical Support and Warranty

6.1 Technical Support

- Avnet Manufacturing Services provides its product with one-year free technical support including:
- Providing software and hardware resources related to the embedded products of Avnet Manufacturing Services;
- Helping customers properly compile and run the source code provided by Avnet Manufacturing Services;
- Providing technical support service if the embedded hardware products do not function properly under the circumstances that customers operate according to the instructions in the documents provided by Avnet Manufacturing Services;
- Helping customers troubleshoot the products.
- The following conditions will not be covered by our technical support service. We will take appropriate measures accordingly:
  - Customers encounter issues related to software or hardware during their development process;
  - Customers encounter issues caused by any unauthorized alter to the embedded operating system;
  - Customers encounter issues related to their own applications;
  - Customers encounter issues caused by any unauthorized alter to the source code provided by Avnet Manufacturing Services.

6.2 Warranty Conditions

- 12-month free warranty on the PCB under normal conditions of use since the sales of the product;
- The following conditions are not covered by free services; Avnet Manufacturing Services will charge accordingly:
  - Customers fail to provide valid purchase vouchers or the product identification tag is damaged, unreadable, altered or inconsistent with the products;
  - Not according to the user's manual operation causes damage to the product;
  - Products are damaged in appearance or function caused by natural disasters (flood, fire, earthquake, lightning strike or typhoon) or natural aging of components or other force majeure;
  - Products are damaged in appearance or function caused by power failure, external forces,

water, animals or foreign materials;

- Products malfunction caused by disassembly or alter of components by customers or, products disassembled or repaired by persons or organizations unauthorized by Avnet Manufacturing Services, or altered in factory specifications, or configured or expanded with the components that are not provided or recognized by Avnet Manufacturing Services and the resulted damage in appearance or function;
- Product failures caused by the software or system installed by customers or inappropriate settings of software or computer viruses;
- Products purchased from unauthorized sales;
- Warranty (including verbal and written) that is not made by Avnet Manufacturing Services and not included in the scope of our warranty should be fulfilled by the party who committed. Avnet Manufacturing Services has no any responsibility.

- Within the period of warranty, the freight for sending products from customers to Avnet Manufacturing Services should be paid by customers; the freight from Avnet Manufacturing Services to customers should be paid by us. The freight in any direction occurs after warranty period should be paid by customers;

- Please contact technical support if there is any repair request.

- **Avnet Manufacturing Services will not take any responsibility on the products sent back without the permission of the company.**

Chapter 7 Contact Information

- Tel: +86-755-33190846/33190847/33190848
- E-mail:
  - Technical support: support@embest-tech.com
  - Sales contact: globalsales@embest-tech.com
- Fax: +86-755-25616057
- Website: http://www.embest-tech.com/
- Address: Tower B 4/F, Shanshui Building, Nanshan Yungu Innovation Industry Park, Liuxian Ave.No.4093,Nanshan District, Shenzhen, Guangdong, China