

NutsBoard

Pistachio SBC

Quick Started Guide

Rev 1.2 20180305

Copyright

NutsBoard Org., All rights reserved.

NutsBoard Org. reserves the right to make improvements in the products described in this manual at any time without notice. No part of this manual may be reproduced, copied, translated or transmitted in any form or by any means without prior written permission of NutsBoard Org..

Trademark

The NutsBoard logo is a registered trademark of NutsBoard Org..

All other trademarks or registered marks in this manual belong to their respective manufacturers.

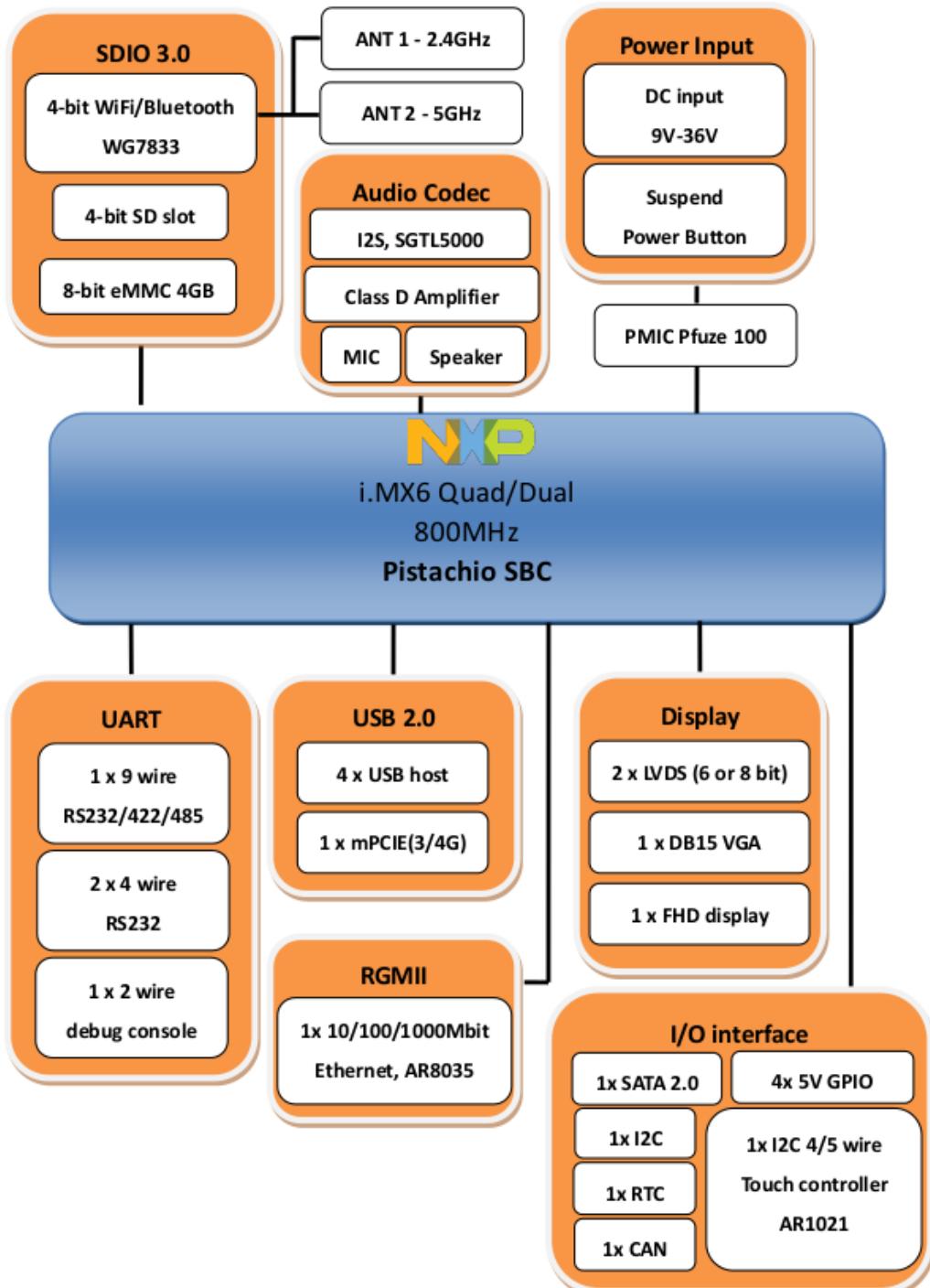
Revision History

Revision	Date	Author	Description
V1.0	08/15/2017	Wig	First Release
V1.1	09/12/2017	Wig	Add made instruction of audio device
V1.2	03/05/2018	Wig	Corrected the RS-422 table

Contents

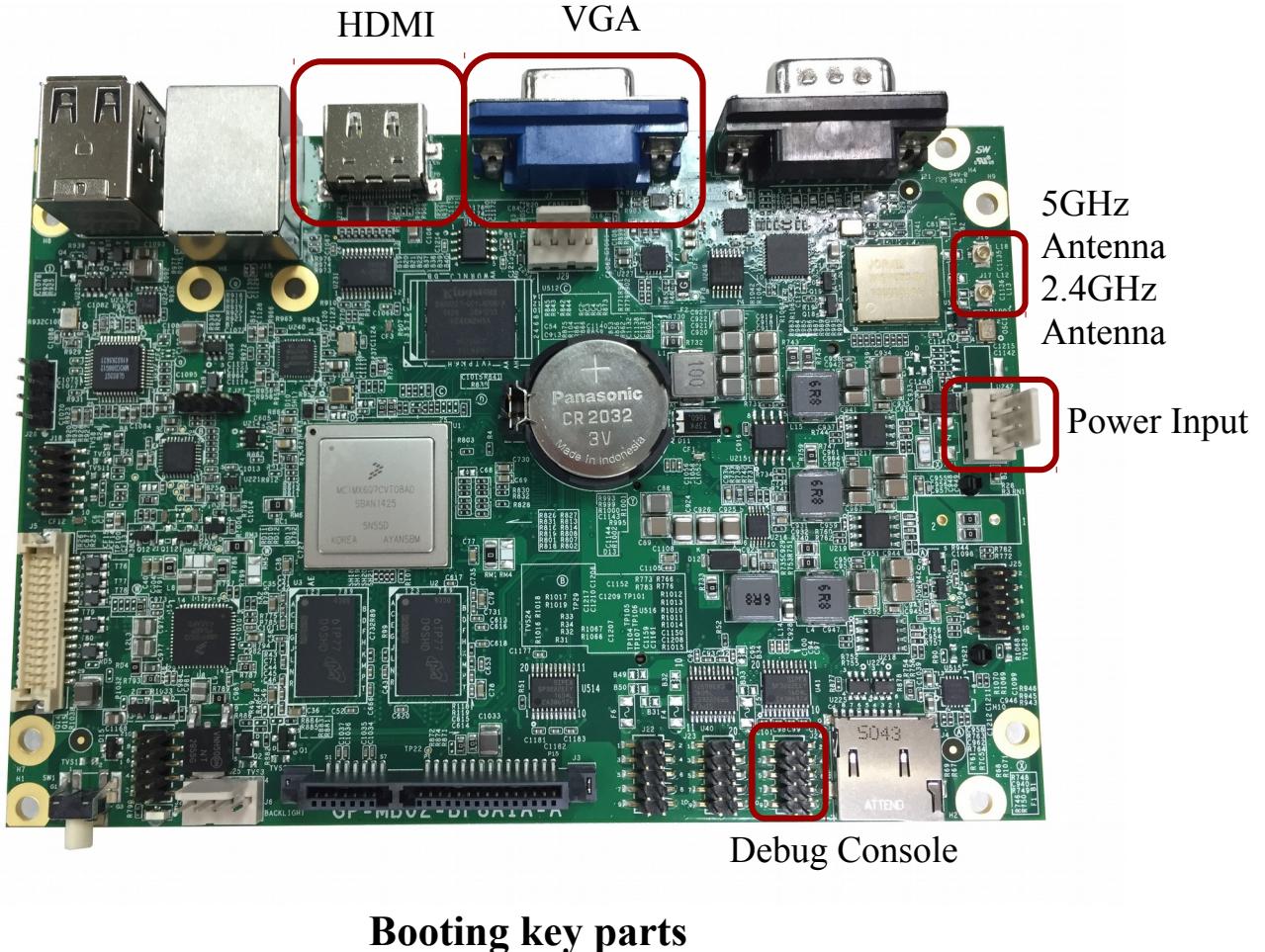
Copyright.....	2
Trademark.....	2
Revision History.....	2
1. Pistachio System Overview.....	1
2. Basic Structure.....	2
2-1. How to Boot up the Pistachio into NutsBoard customize OS?.....	2
3. Download and Flash an Operating System Demo Image.....	3
3-1. Supporting list.....	3
3-2. How to Download.....	3
3-3. How to Flash – Windows User.....	3
3-4. How to Flash – Linux User.....	5
4. Booting Steps.....	6
4-1. Power input.....	6
4-2. Debug console (optional).....	7
4-3. Wireless antenna connector (optional).....	11
4-4. Display.....	12
4-5. Change the boot mode.....	13
5. Advance Structure.....	14
5-1. Power Button.....	15
5-2. LVDS connector.....	15
5-3. Audio connector.....	15
5-4. USB connector.....	16
5-5. Ethernet connector.....	17
5-6. UART1 DB9 connector.....	17
5-7. WiFi/Bluetooth Functions.....	18
5-8. I/O Expander Pin Header.....	18
5-9. uSD Card Slot.....	18
5-10. UART2 and UART3 Pin Header.....	18
5-11. SATA Connector.....	18
5-12. Backlight Connector.....	19
5-13. I2C Touch Connector.....	19
5-14. RTC Slot.....	19
5-15. FlexCAN Connector.....	19
5-16. mPCIE Connector.....	19

1. Pistachio System Overview



2. Basic Structure

2-1. How to Boot up the Pistachio into NutsBoard customize OS?



3. Download and Flash an Operating System Demo Image

3-1. Supporting list

we have seven OS options to best fit your needs soon. **Red color** means not release yet.

Operating System	Revision	Applications
Android	7.1.1	HMI
BuildRoot	201611	IoT
Debian Stretch Lite	9.1	IoT
Debian Stretch Desktop	9.1	HMI
Yocto Morty QT4 Desktop	2.3	HMI
Ubuntu Core	16	IoT
Ubuntu Desktop	18.04	HMI

3-2. How to Download

Pistachio has 4GB of high-reliability MLC eMMC storage onboard for holding the core operating system and a limited amount of user and program data.

Because of this, it's a good idea to know how much storage software will take before flashing and installing. Where needed, additional high-speed storage can be added through Pistachio's SDIO bus of uSD slot.

Download link: http://nutsboard.org/pistachio_download

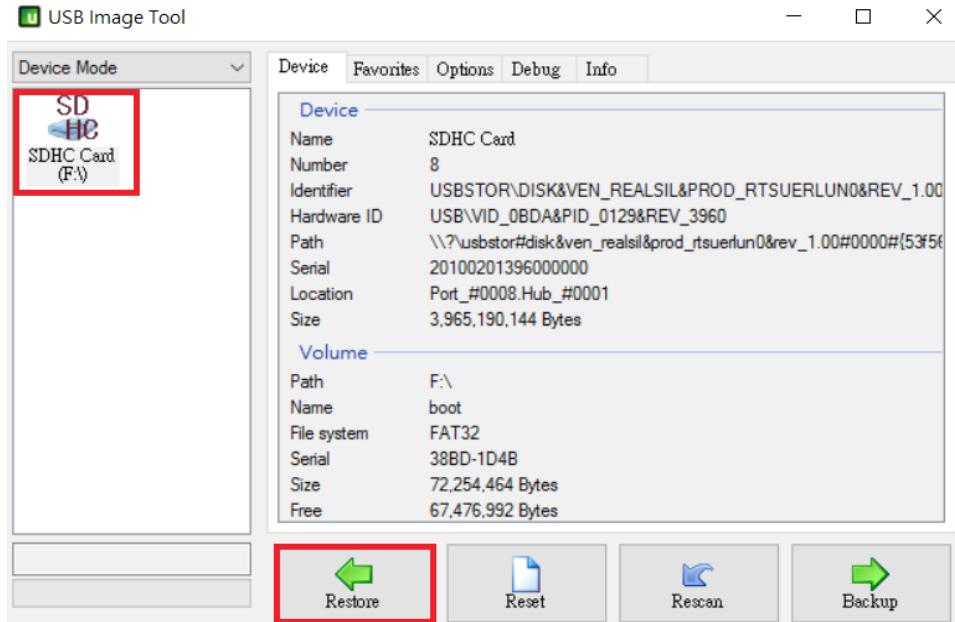
If you need eMMC boot image, please choose "installer image".
If you need uSD boot image, please choose "runtime image".

3-3. How to Flash – Windows User

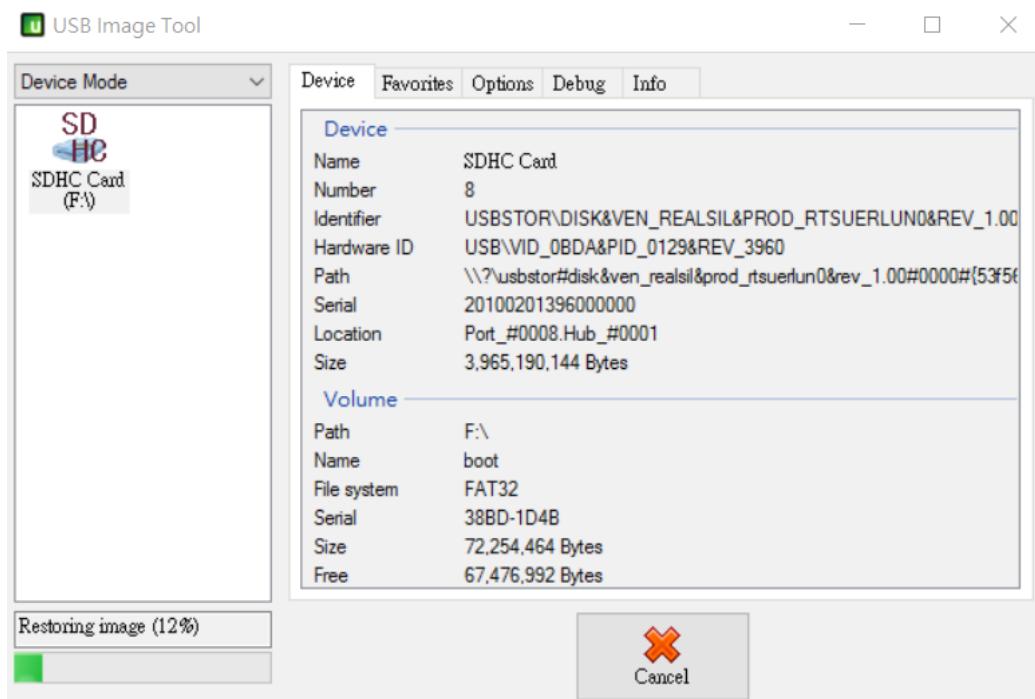
Head over to the web flasher software "usbit.zip" file at https://mega.nz/#!NsUkha4Q!NiyvA-bSPDTfGFNg8IlhSAbv7t9lGexmQIFDV3qt_b8.

opening the application, you should see your sdcard is mounted on left side.
Then press the "Restore" button to write a sdcard image.

Pistachio Quick Started Guide v1.2 - March 05 2018



Once pressed the “Open” button, the sdcard writing should start. You’ll see a progress bar on the left bottom side. Once it’s at 100% you’re ready to flashed:



3-4. How to Flash – Linux User

1. Prepare a uSD card and inserting the host PC.
2. After download the expected image, issue the command to flashing uSDcard
- “`dd if=*.img of=/dev/sdx bs=1M;sync;`”.

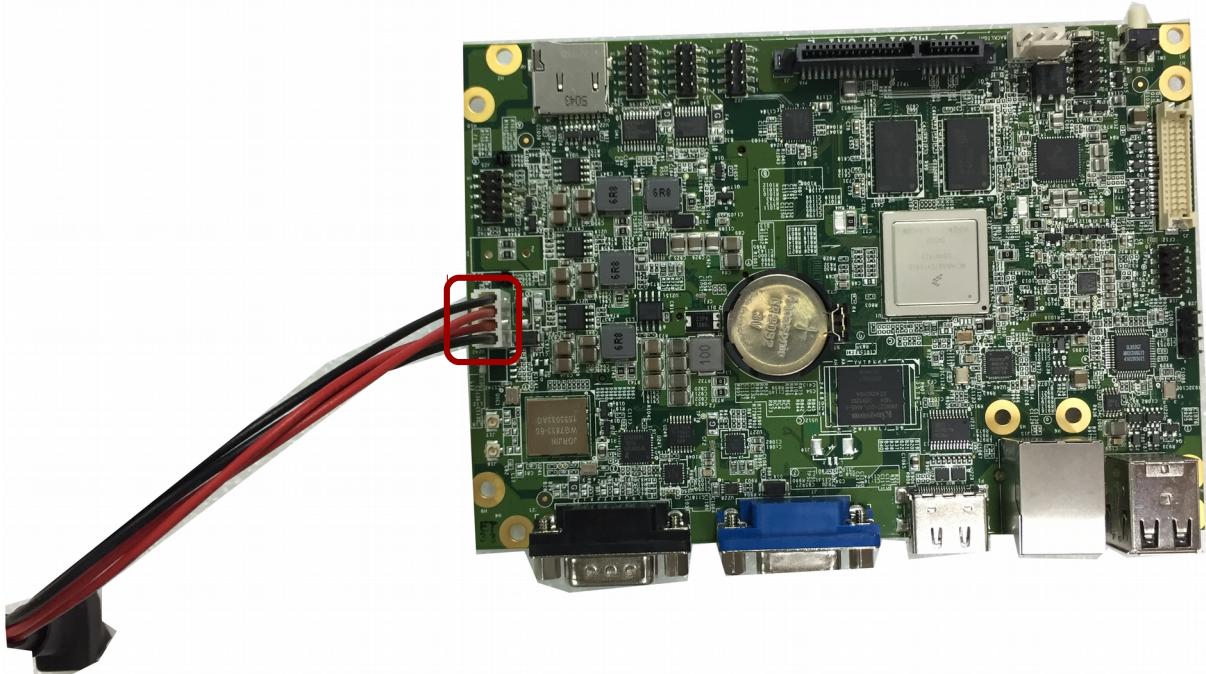
`*.img` means your expected download image.

`/dev/sdx` means the device node of sdcard what your mounted. (ex: `/dev/sdb`)

4. Booting Steps

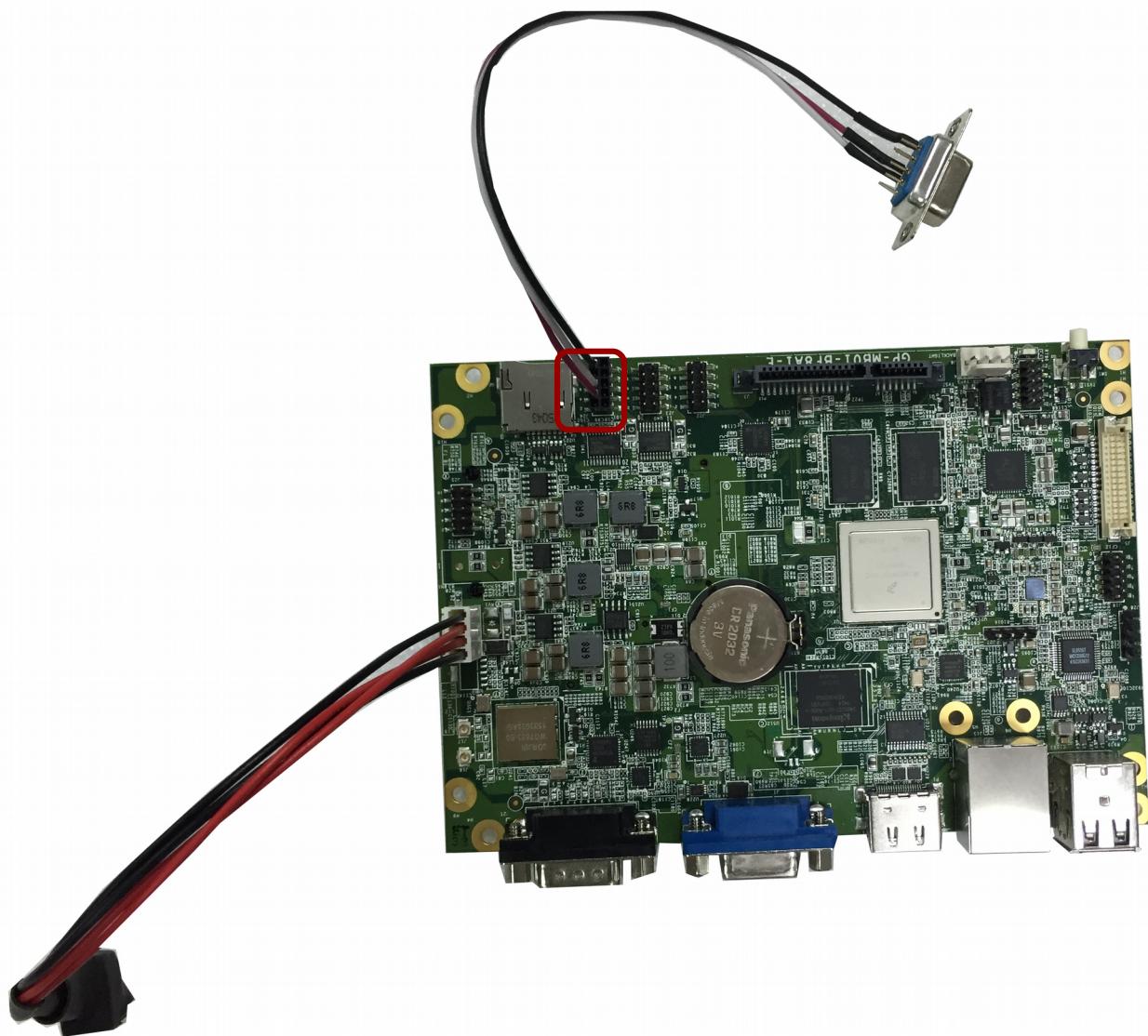
4-1. Power input

The connector of power input is as below red border, you can insert our DC-Jack power cable, because it's DC-Jack connector, so please prepare a power adapter to give it DC power, the support range is 9-36 volts, we recommends you can use 12 volts with 2A above power adapter, it's enough!



4-2. Debug console (optional)

The debug console pin headers as below red border, you can insert our customize debug console cable into 10-pin pin headers, and prepare a USB to serial cable to connect this one to your host PC, open a terminal software and setting baud rate 115200, then you can starting develop your own applications, if no any output messages for console, please reverse the connector of debug console cable and insert again.



Debug setting on windows OS

- Open the TeraTrem and set Baud rate to 115200
- Setup → Serial Port → Baud Rate “115200”
- See the results as below picture.

```
Starting logging: OK
Initializing random number generator... [ 10.916867] random: uninitialized urandom read (512 bytes read, 109 bits of entropy available)
done.
Starting network: OK
[ 11.368401] random: ssh-keygen: uninitialized urandom read (32 bytes read, 3 bits of entropy available)
Starting sshd: [ 11.489799] random: sshd: uninitialized urandom read (32 bytes read, 116 bits of entropy available)
OK

Welcome to Buildroot
buildroot login: [ 12.628765] random: python: uninitialized urandom read (0 bytes read, 126 bits of entropy available)
[ 12.947066] random: nonblocking pool is initialized
* Running on http://0.0.0.0:8080/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger pin code: 196-455-687
```

Debug setting on Linux OS

- Install minicom first

```
chris@ubuntu:~$ sudo apt install minicom
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libappindicator1 libindicator7 libpango1.0-0 libpangox-1.0-0
```

- Setting the minicom configuration

```
chris@ubuntu:~$ ls /dev/ttyUSB0
/dev/ttyUSB0
chris@ubuntu:~$ sudo minicom -s
```

- Doing minicom configuration:
 - press “A”,Modify “ttyUSBx” <your device node of serial port>
 - press “F”,change “N”

- press “Enter”

```
+----[configuration]----+
| Filenames and paths
| File transfer protocols
| Serial port setup
| Modem and dialing
| Screen and keyboard
| Save setup as dfl
| Save setup as..
| Exit
| Exit from Minicom
+-----+
```

```
A - Serial Device      : /dev/ttyUSB0
B - Lockfile Location  : /var/lock
C - Callin Program    :
D - Callout Program   :
E - Bps/Par/Bits       : 115200 8N1
F - Hardware Flow Control : No
G - Software Flow Control : No

Change which setting? ■
```

```
+----[configuration]----+
| Filenames and paths
| File transfer protocols
| Serial port setup
| Modem and dialing
| Screen and keyboard
| Save setup as dfl
| Save setup as..
| Exit
| Exit from Minicom
+-----+
```

```
Welcome to minicom 2.7

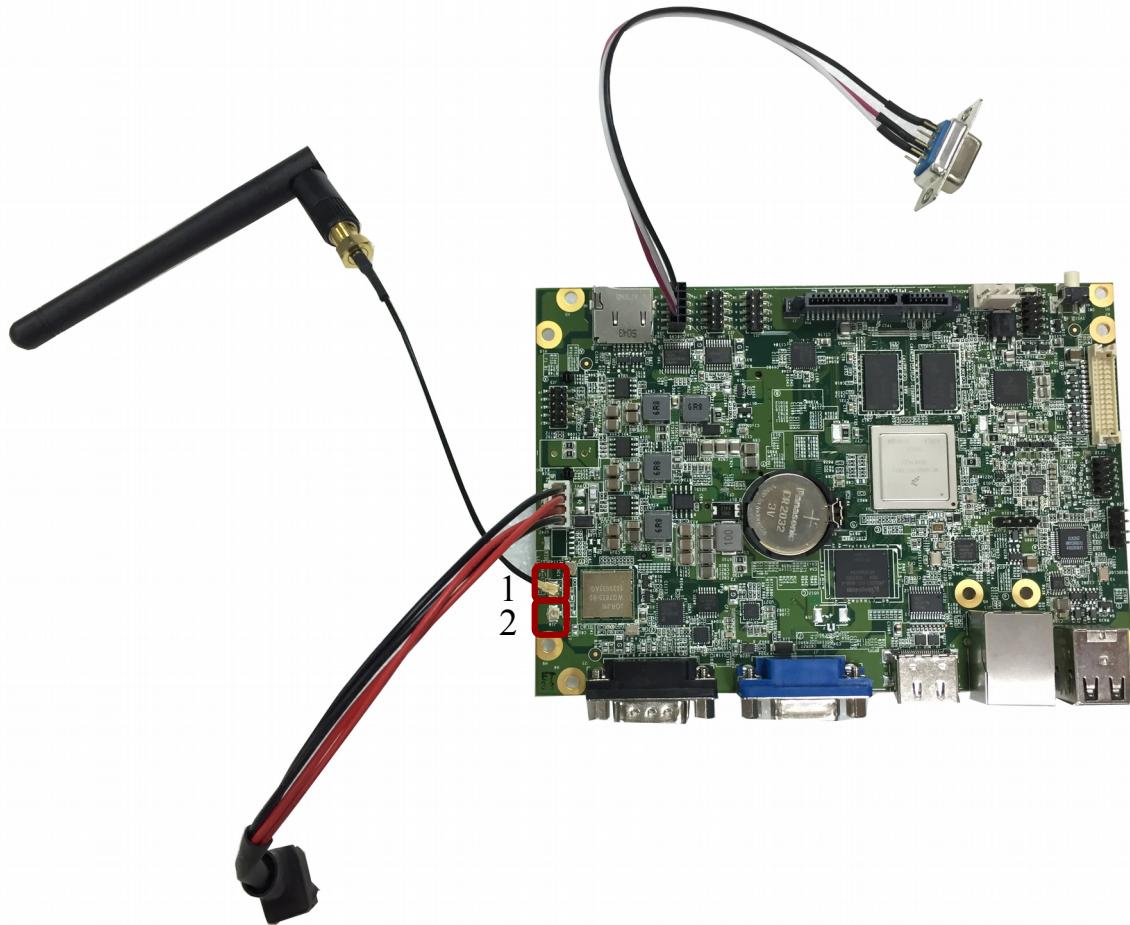
OPTIONS: I18n
Compiled on Feb  7 2016, 13:37:27.
Port /dev/ttyUSB0, 14:48:38

Press CTRL-A Z for help on special keys

Welcome to Buildroot
buildroot login: /root/qc-web
```

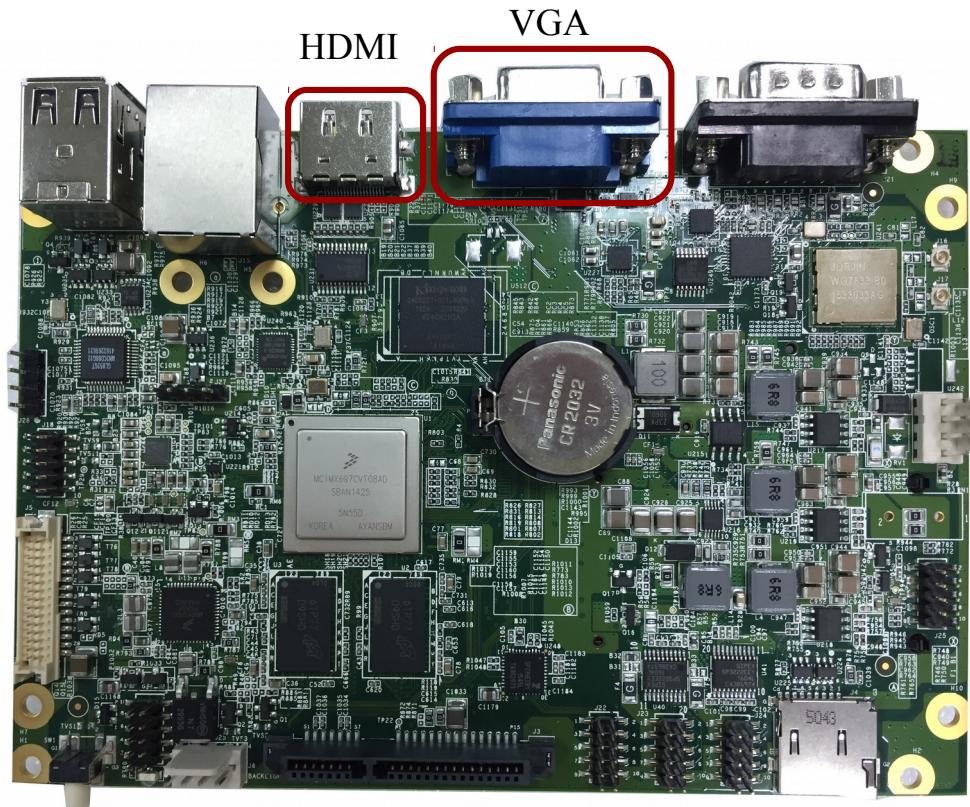
4-3. Wireless antenna connector (optional)

The Wireless connectors has separated 2 band as red border of below pictures, one is 2.4GHz band as “1” label of below picture, another one is 5GHz band as “2” label of below picture, please prepare a antenna to connected the connector as below if you have to use WiFi/BT functions.



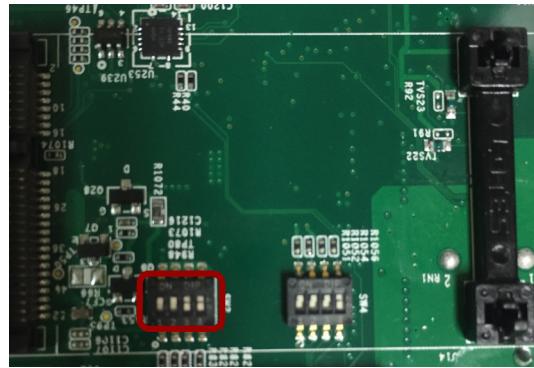
4-4. Display

It's alternative solutions by your choice, we support FHD MI and VGA port, you can prepare a FHD MI or VGA monitor with a cable and making a connection, of course you can pulg 2 type monitors at the same time, it will showing the same content for different display.

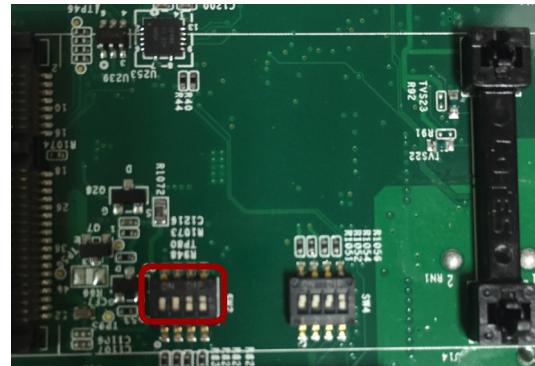


4-5. Change the boot mode

Pistachio supporting two boot modes: uSD boot and eMMC boot, default setting is use eMMC boot, the DIP switch setting is as below picture (another side of Pistachio) - ON ON OFF OFF ($\uparrow\uparrow\downarrow\downarrow$), and about uSD boot mode, the DIP switch setting is as below picture (another side of Pistachio) -OFF OFF ON ON ($\downarrow\downarrow\uparrow\uparrow$).

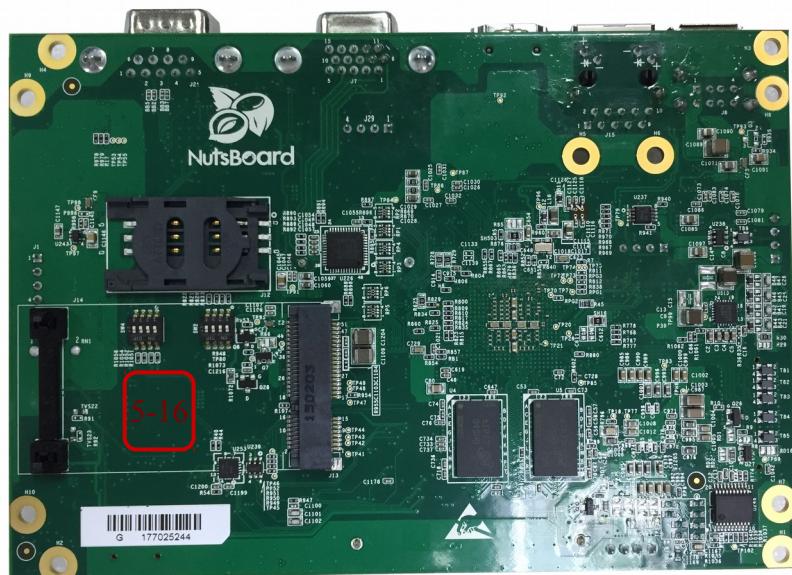
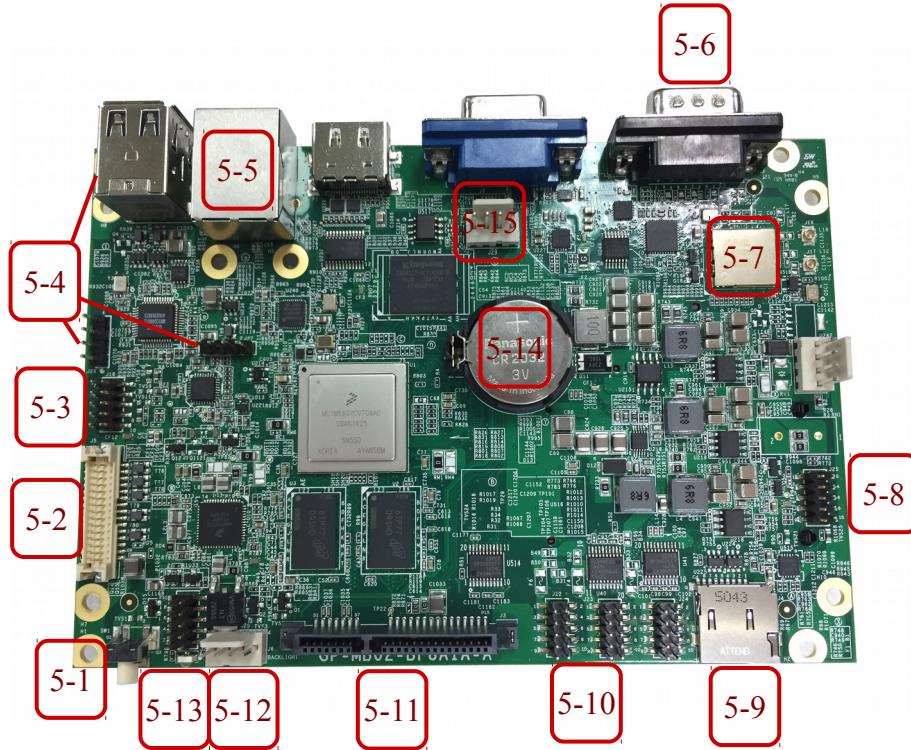


eMMC boot mode



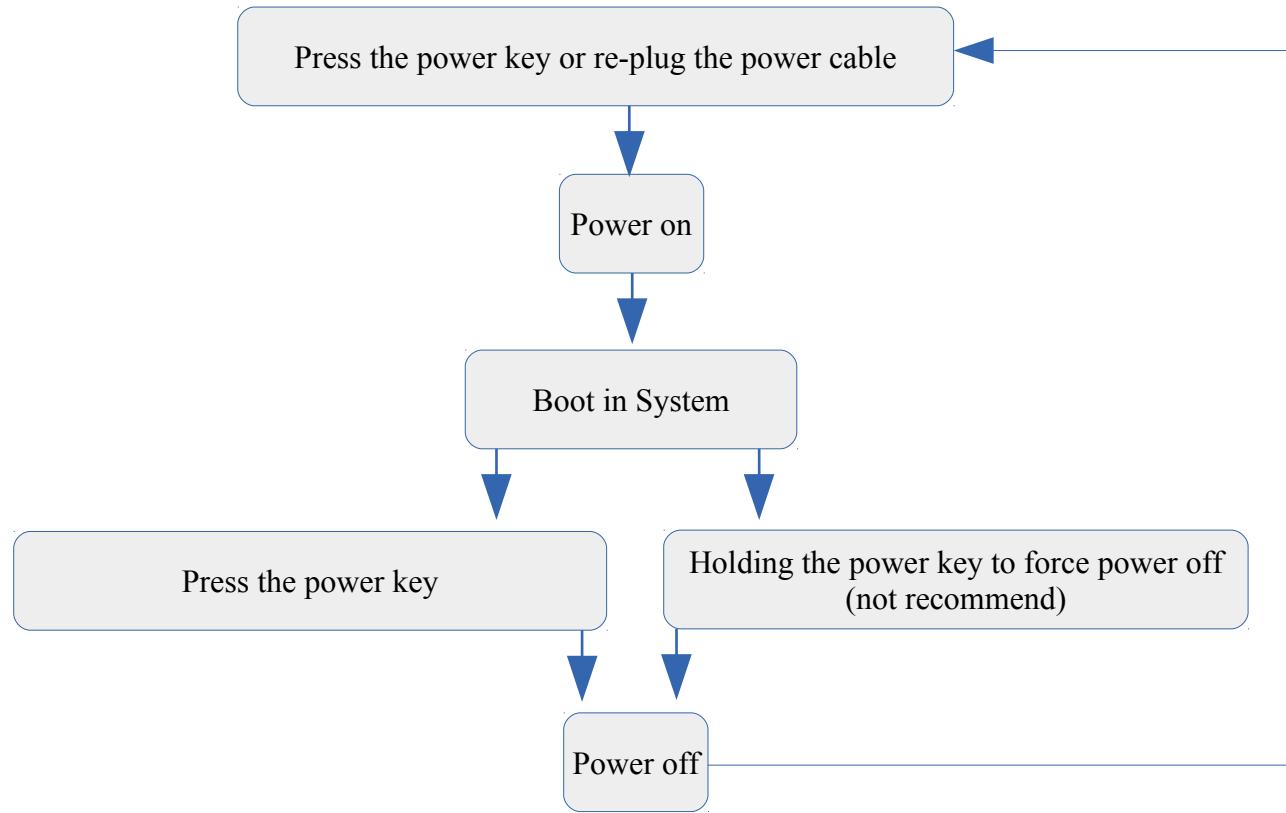
uSD boot mode

5. Advance Structure



5-1. Power Button

It's a smart power button, the control flow as following:



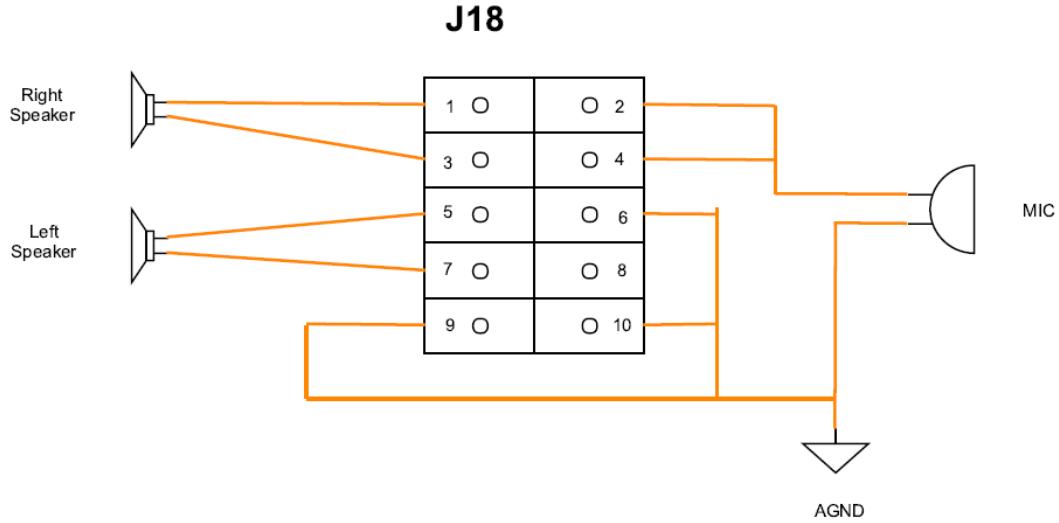
5-2. LVDS connector

Here's two LVDS channel, you can choose different panel using one or two single channel or dual channel, our limitation of bit depth is 8-bit, the pin definition as NutsBoard Pistachio Hardware Manual, please refer it.

5-3. Audio connector

We offer a connector including one Microphone, one speaker with detection pin, you can do your own audio cable with devices, we also provide a good audio quality with amplifier, the pin definition as following table, please refer it, we also provide a way to making your own audio device.

Pin #	Net Name	Description	Pin #	Net Name	Description
1	SPKOUTRP	Speaker right +	2	M_CBIAS	M_C_IN
3	SPKOUTRN	Speaker right -	4	M_CIN	M_C_IN
5	SPKOUTLP	Speaker left +	6	M_C_DET	M_C_GND
7	SPKOUTLN	Speaker left -	8	HEAD_DET	NC for Speaker apply (3W Speaker AMP Enable)
9	AGND		10	AGND	M_C_GND



NOTE: Please don't connect your cable to audio connector wrong side, it could be cause audio codec broken.

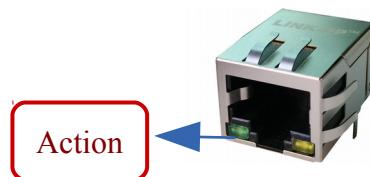
5-4. USB connector

We not only offer two standard USB Type-B connectors but also have two 4-pins pin header connectors for your USB touch applications.

NOTE: Please don't connect your cable to 4-pin connector wrong side, it could be cause USB device broken.

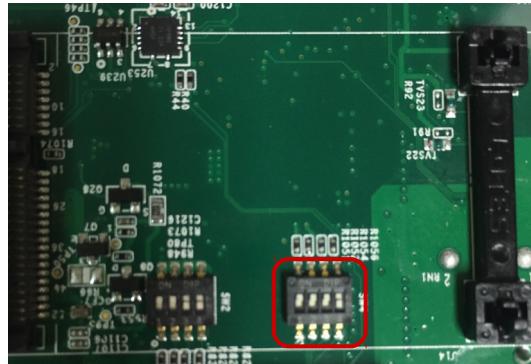
5-5. Ethernet connector

It's RGMII based PHY, so you can choose the speed what you want from 10Mb to 1000Mb (default is 1000Mb), the LED state definition of PHY connector as following:



5-6. UART1 DB9 connector

It support full function (2-wire, 4-wire and 9-wire) for RS232 mode, and also supporting RS422 and RS485 mode, you can tune the DIP switch SW4 on the back side to change mode as following:



Mode	SW3 State
RS232	On, Off, Off, Off ($\uparrow \downarrow \downarrow \downarrow$)
RS422	On, On, On, Off ($\uparrow \uparrow \uparrow \downarrow$)
RS485	Off, On, Off, Off ($\downarrow \uparrow \downarrow \downarrow$)

For example: As above picture, it's RS422 mode ($\uparrow \uparrow \uparrow \downarrow$).

NOTE: RS485 mode need modify Linux Kernel in software, please refer Linux Kernel release note if you need.

5-7. WiFi/Bluetooth Functions

WiFi is support IEEE 802.11 a/b/g/n mode for 2.4GHz/5GHz band, Bluetooth is support classic mode and BLE mode, our software also support hotspot feature.

5-8. I/O Expander Pin Header

This connector is very special including four 5 Volt GPIO pin and 1 I2C bus, you can do your cable or small PCB to connect this expander for your applications, detail definition as NutsBoard Pistachio Hardware Manual, please refer it.

5-9. uSD Card Slot

It's very important part no matter boot software from SD card or install / upgrade software to eMMC storage, the maximum size is recommends 64GB.

5-10. UART2 and UART3 Pin Header

You can expand your UART ports using these two ports for different applications, but UART 2 and UART3 only support RS232 2-wire and 4-wire mode.

5-11. SATA Connector

It's standard SATA port, you can do your own cable or buy standard cable to connect it to a hard disk (HDD) to increase your storage size such as NAS or server applicaitons.

5-12. Backlight Connector

It's PWM controller with 12 volt, detail pin definition as NutsBoard Pistachio Hardware Manual, please refer it.

5-13. I2C Touch Connector

We also support CTP, 4 Wire and 5 Wire RTP I2C touch support expect USB touch as above, detail pin definition as NutsBoard Pistachio Hardware Manual, please refer it.

5-14. RTC Slot

NOTE: It's a very important section!

Default we have no coin battery with shipping, it's due to Pistachio RTC current consumption is too high, so if your application is just power off several hours a day, you're suitable this RTC function.

But if long battery life is your absolute priority application, then using the Pistachio for a RTC is not going to be your best approach (current consumption from could be 250uA after power off), external RTC's will provide better battery performance using section 5-8 I2C bus.

5-15. FlexCAN Connector

It's standard CAN bus, you can do your own cable for your applications, detail definition as NutsBoard Pistachio Hardware Manual, please refer it.

5-16. mPCIE Connector

It's not standard mPCIE connector, it only support 3G/4G communications mPCIE based module via USB bus, detail definition as NutsBoard Pistachio Hardware Manual, please refer it.

NOTE: Every 3G/4G module is different pin definition, we recommend you compare the current module and Pistachio mPCIE connector pin definition is match or not to avoid the module is not working.