

# REV-SA02

SMARC Evaluation Carrier Board

# User's Manual

---

1<sup>st</sup> Ed – 28 July 2016

# Content

<b>1. Getting Started</b>	<b>4</b>
1.1 Safety Precautions	4
1.2 Packing List	4
1.3 Document Amendment History	5
1.4 Manual Objectives	6
1.5 System Specifications	7
1.6 Architecture Overview—Block Diagram	9
<b>2. Hardware Configuration</b>	<b>10</b>
2.1 Product Overview	11
2.2 Jumper and Connector List	12
2.3 Setting Jumpers & Connectors	14
2.3.1 Boot Mode selector (SSET1)	14
2.3.2 RS485-2 selector (JRS485-2)	15
2.3.3 Camera input selector (JPCAM1)	15
2.3.4 LCD brightness PWM voltage select (JPWM1)	16
2.3.5 General purpose I/O connector (JGPIO1)	16
2.3.6 RTC Battery connector (JBAT1)	17
2.3.7 USB2.0 connector (JUSB5)	17
2.3.8 LVDS Interface connector (JLVDS1)	18
2.3.9 CAN Bus connector 1 (JCAN1)	19
2.3.10 CAN Bus connector 2 (JCAN2)	19
2.3.11 DMIC_WM8962 connector (JDMIC-W1)	20
2.3.12 DMIC_ALC892 connector (JDMIC-A1)	20
2.3.13 Inverter connector (JINV1)	21
2.3.14 Power connector (JPWR1)	21
2.3.15 Serial port 1 in RS-485 mode (J485-1)	22
2.3.16 Serial port 2 in RS-485 mode (J485-2)	22
2.3.17 Speaker connector (JSPR1)	23
2.3.18 Speaker connector (JSPL1)	23
2.3.19 Battery connector (JBATC1)	24
2.3.20 I2C connector (JI2C1)	24
2.3.21 Camera connector (JMIPI1)	25
<b>3. Linux User Guide</b>	<b>26</b>
3.1 Download Source code for building Ubuntu image file	27
3.2 Set up a Linux host for building U-boot & Kernel Image	27
3.3 Building up U-boot & Kernel image	27

## REV-SA02 User's Manual

3.4	Use MfgTool to flash Ubuntu into onboard eMMC .....	28
3.5	Download Android Source Code for building image file .....	31
3.6	Set up for building Android image file.....	31
3.7	Building up Android image file.....	32
3.8	Use MfgTool to flash Android into onboard eMMC.....	33

# 1. Getting Started

## 1.1 Safety Precautions

### Warning!



Always completely disconnect the power cord from your chassis whenever you work with the hardware. Do not make connections while the power is on. Sensitive electronic components can be damaged by sudden power surges. Only experienced electronics personnel should open the PC chassis.

### Caution!



Always ground yourself to remove any static charge before touching the CPU card. Modern electronic devices are very sensitive to static electric charges. As a safety precaution, use a grounding wrist strap at all times. Place all electronic components in a static-dissipative surface or static-shielded bag when they are not in the chassis.

**Always note** that improper disassembling action could cause damage to the motherboard. We suggest not removing the heatsink without correct instructions in any circumstance. If you really have to do this, please contact us for further support.

## 1.2 Packing List

Before you begin installing your single board, please make sure that the following materials have been shipped:

- 1 x REV-SA02 Micro Module
- 4 x Screw (M2.5\*4mm)



---

If any of the above items is damaged or missing, contact your retailer.

### 1.3 Document Amendment History

Revision	Date	By	Comment
1st	July 2016	Avalue	Initial Release

### 1.4 Manual Objectives

This manual describes in details Avalue Technology REV-SA02 Single Board.

We have tried to include as much information as possible but we have not duplicated information that is provided in the standard IBM Technical References, unless it proved to be necessary to aid in the understanding of this board.

We strongly recommend that you study this manual carefully before attempting to set up REV-SA02 series or change the standard configurations. Whilst all the necessary information is available in this manual we would recommend that unless you are confident, you contact your supplier for guidance.

Please be aware that it is possible to create configurations within the CMOS RAM that make booting impossible. If this should happen, clear the CMOS settings, (see the description of the Jumper Settings for details).

If you have any suggestions or find any errors regarding this manual and want to inform us of these, please contact our Customer Service department with the relevant details.

## 1.5 System Specifications

<b>System</b>	
<b>SSD</b>	SD Socket x1
<b>I/O</b>	
<b>External I/O Connector</b>	2 RCA , 90D 1 RJ45 connector for GBE (with LED) 1 Micro SD Socket , 90D 1 Mini USB , OTG 1 USB 2.0 1 USB Type A 2.0 double deck HDMI Displayport VGA 2 RS232
<b>USB</b>	USB 2.0 : 1 Double Deck USB 2.0 Type A connector 1 USB 2.0 Type A One Mini USB (OTG) 1 USB2.0, pitch 2.0mm WB One is used by Mini PCIe connector
<b>Internal I/O Connector</b>	Camera : FPC ,0.5mm,20p Speaker : speaker connector x2 D-MIC Mini PCIe connector(SATA) Mini PCIe connector(USB) SIM Card GPIO : One 10 x 2, pitch 2.00mm Header I2C : One 1 x 5, pitch 2.00mm Header USB2.0: 1 x 5, pitch 2.00mm Wafer PCIE-1 Battery connector LVDS Connect One 20 x 2, pitch 1.25mm Wafer Inverter Connector One 5p, pitch 2.00mm Wafer 2 RS485 : 4p, pitch 2.00mm Wafer 2 CAN : 4p, pitch 2.00mm Wafer WB (reserve)
<b>Switch &amp; LED</b>	RST Switch Power Switch

## REV-SA02 User's Manual

	Power LED (Green / Yellow) DIP SW
<b>Audio</b>	
<b>I2C/I2S/HDA Socket</b>	I2C/I2S/HDA Socket for different Audio codec
<b>Audio codec accessory board</b>	Line out/Line in/AMIC , DMIC in interface X86 platform => ALC892 ARM platform => WM8962
<b>Ethernet</b>	
<b>Ethernet Interface</b>	1 RJ45 connector for GBE (with LED)
<b>Mechanical &amp; Environmental</b>	
<b>Power Requirement</b>	DC +12V
<b>Power Type</b>	DC input
<b>Operating Temp.</b>	-40 ~ 85°C
<b>Storage Temp.</b>	-40 ~ 85°C
<b>Operating Humidity</b>	0% ~ 90% Relative Humidity, Non-condensing
<b>Size (L x W)</b>	170mm x 170mm (Mini ITX Form Factor) (Please consult product engineers for the production feasibility if the size is larger than 410 x 360mm or smaller than 80 x 70mm)
<b>Weight</b>	250g

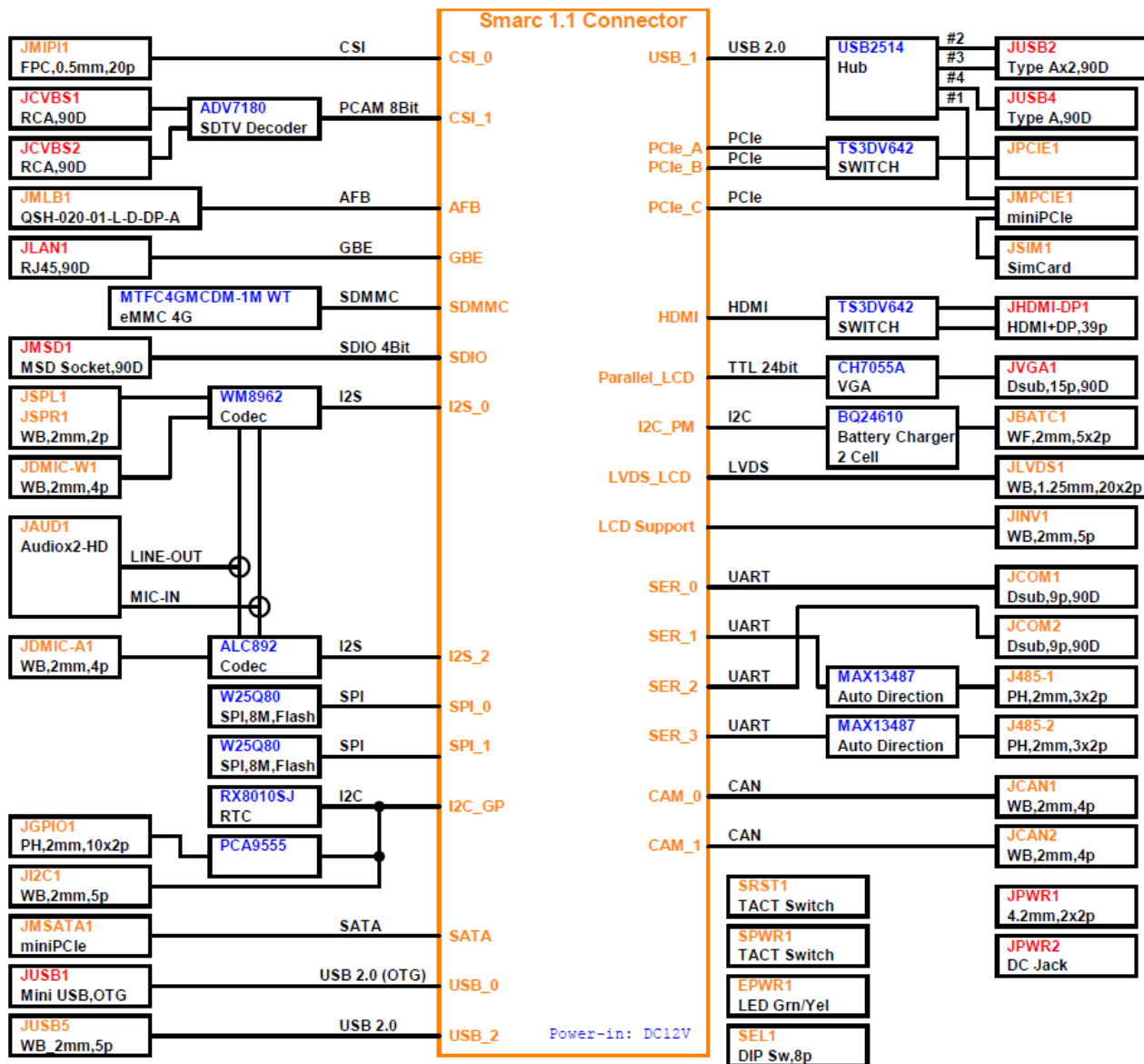


**Note:** Specifications are subject to change without notice.



## 1.6 Architecture Overview—Block Diagram

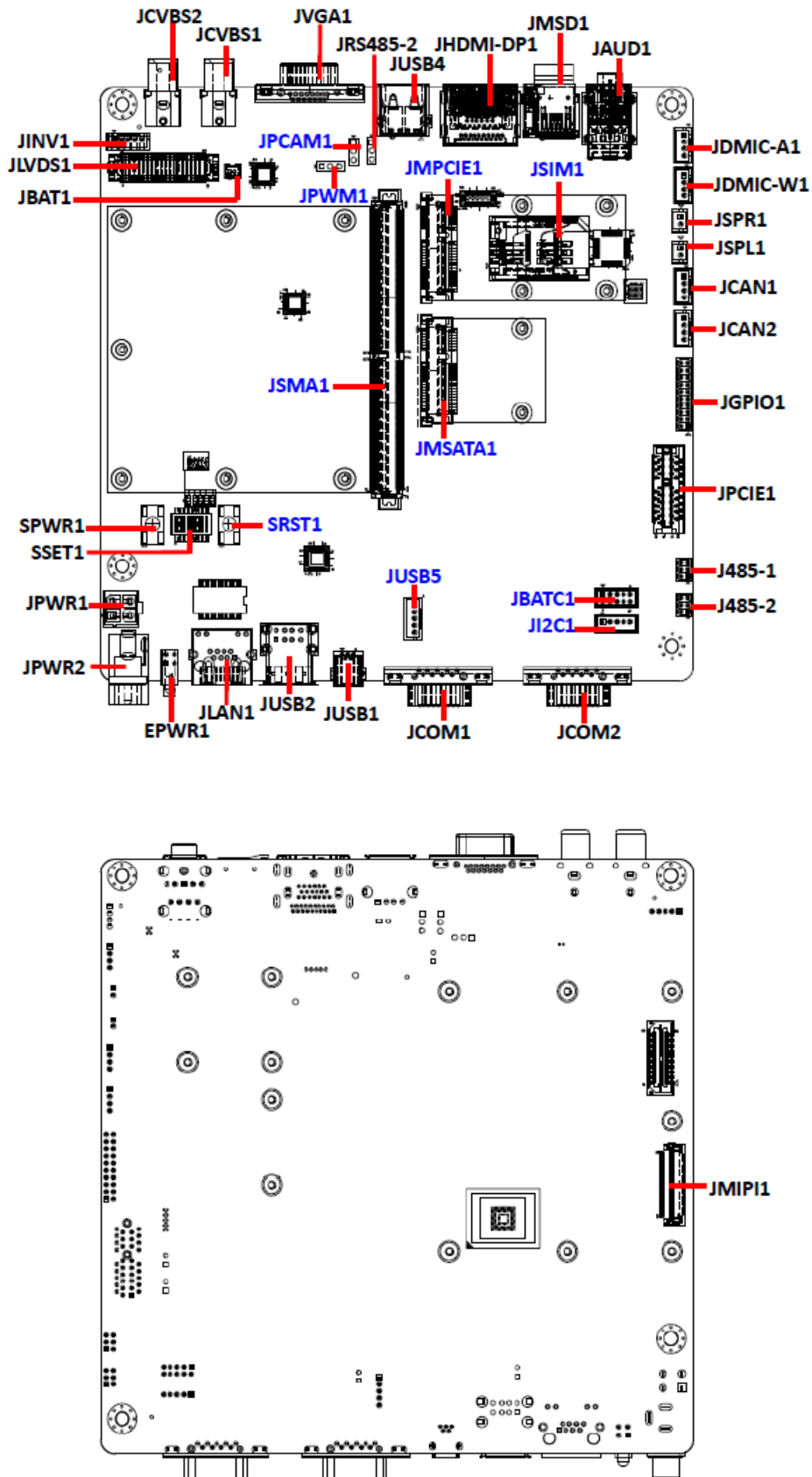
The following block diagram shows the architecture and main components of REV-SA02.



# 2. Hardware Configuration

---

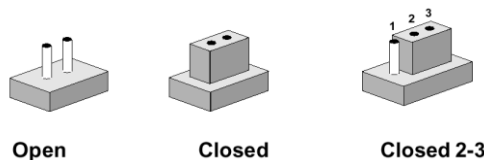
## 2.1 Product Overview



## 2.2 Jumper and Connector List

You can configure your board to match the needs of your application by setting jumpers. A jumper is the simplest kind of electric switch.

It consists of two metal pins and a small metal clip (often protected by a plastic cover) that slides over the pins to connect them. To “close” a jumper you connect the pins with the clip. To “open” a jumper you remove the clip. Sometimes a jumper will have three pins, labeled 1, 2, and 3. In this case, you would connect either two pins.



The jumper settings are schematically depicted in this manual as follows:



A pair of needle-nose pliers may be helpful when working with jumpers. Connectors on the board are linked to external devices such as hard disk drives, a keyboard, or floppy drives. In addition, the board has a number of jumpers that allow you to configure your system to suit your application.

If you have any doubts about the best hardware configuration for your application, contact your local distributor or sales representative before you make any changes.

The following tables list the function of each of the board's jumpers and connectors.

### Jumpers

Label	Function	Note
<b>SSET1</b>	Boot Mode selector	DIP Switch 8P
<b>JRS485-2</b>	RS485-2 selector	3 x 1 header, pitch 2.54mm
<b>JPCAM1</b>	Camera input selector	3 x 1 header, pitch 2.54mm
<b>JPWM1</b>	LCD brightness PWM voltage select	3 x 1 header, pitch 2.54mm

### Connectors

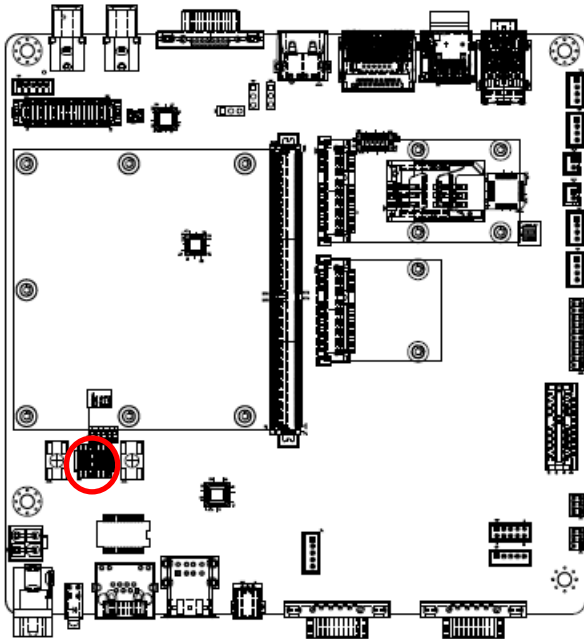
Label	Function	Note
<b>JBAT1</b>	RTC Battery connector	2 x 1 wafer, pitch 1.25mm
<b>JGPIO1</b>	General purpose I/O connector	10 x 2 header, pitch 2.00mm
<b>JCAN1</b>	CAN Bus connector 1	4 x 1 wafer, pitch 2.00mm
<b>JCAN2</b>	CAN Bus connector 2	4 x 1 wafer, pitch 2.00mm
<b>JSMA1</b>	Smart Mobility ARChitecture slot	

## REV-SA02 User's Manual

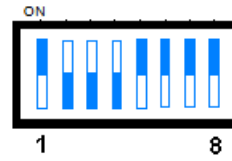
<b>JLAN1</b>	RJ-45 Ethernet connector	
<b>JVGA1</b>	VGA connector	
<b>JUSB1</b>	Mini USB connector for Boot/Debug	MINI USB-MAB_5P
<b>JUSB2</b>	2 x USB2.0 connector	
<b>JUSB4</b>	USB2.0 connector	
<b>JUSB5</b>	USB2.0 connector	5 x 1 wafer, pitch 2.00mm
<b>JHDMI-DP1</b>	HDMI/DP connector	
<b>JLVDS1</b>	LVDS Interface connector	20 x 2 wafer, pitch 1.25mm
<b>JPWR1</b>	Power connector	2 x 2 wafer, pitch 4.20mm
<b>JPWR2</b>	Power connector	
<b>JMPCIE1</b>	Mini PCIe connector	
<b>JSIM1</b>	SIM Card Slot	SDCARD_9H, Push/Push Type
<b>JDMIC-A1</b>	DMIC_ALC892 connector	4 x 1 wafer, pitch 2.00mm
<b>JDMIC-W1</b>	DMIC_WM8962 connector	4 x 1 wafer, pitch 2.00mm
<b>JSPR1</b>	Speaker connector	2 x 1 wafer, pitch 2.00mm
<b>JSPL1</b>	Speaker connector	2 x 1 wafer, pitch 2.00mm
<b>JPCIE1</b>	PCIe connector	
<b>JCOM1/2</b>	Serial Port 1/2 connector	
<b>J485-1</b>	Serial port 1 in RS-485 mode	3 x 2 header, pitch 2.00mm
<b>J485-2</b>	Serial port 2 in RS-485 mode	3 x 2 header, pitch 2.00mm
<b>EPWR1</b>	Power LED (Green / Yellow)	
<b>JBATC1</b>	Battery connector	5 x 2 wafer, pitch 2.00mm
<b>JI2C1</b>	I2C connector	5 x 1 wafer, pitch 2.00mm
<b>JINV1</b>	Inverter connector	5 x 1 wafer, pitch 2.00mm
<b>JMSD1</b>	Micro SD Socket	
<b>JAUD1</b>	Audio connector	
<b>JMSATA1</b>	SMBUS & SATA connector	
<b>JCVBS1/2</b>	2 x RCA connector	
<b>SRST1</b>	Reset SW	
<b>SPWR1</b>	Power SW	
<b>JMIPI1</b>	Camera connector	36 x 1 header, pitch 0.50mm

## 2.3 Setting Jumpers & Connectors

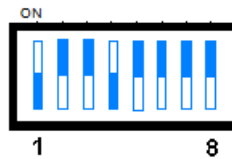
### 2.3.1 Boot Mode selector (SSET1)



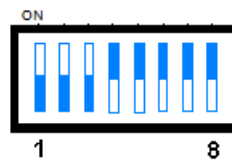
#### Booting from onboard eMMC



#### Boot from SD card



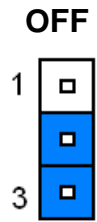
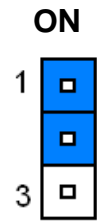
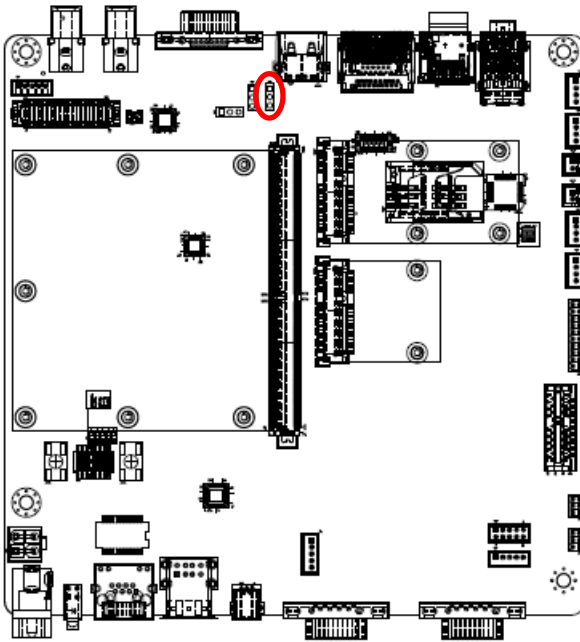
#### MFG tool mode mode(burning image file to onboard eMMC)



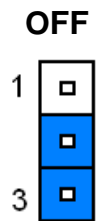
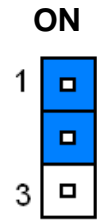
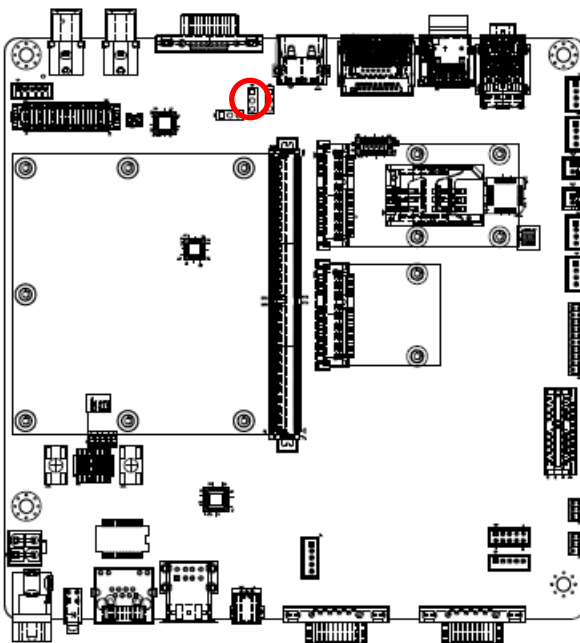
Signal	PIN	PIN	Signal
GND	1	16	BOOT_SEL0#
GND	2	15	BOOT_SEL1#
GND	3	14	BOOT_SEL2#
GND	4	13	FORCE_RECOV#
GND	5	12	SW_CODEC#
GND	6	11	SW_PCIE#
GND	7	10	SW_HDMI#
GND	8	9	SW_HDMI_PD#

PIN	ON	OFF
5	WM8962	ALC892
6	Port A	Port B
7	HDMI Port	Display Port
8	HDMI Pull down Off	HDMI Pull down On

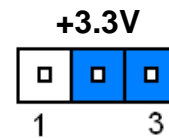
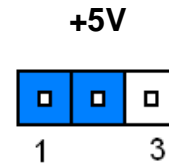
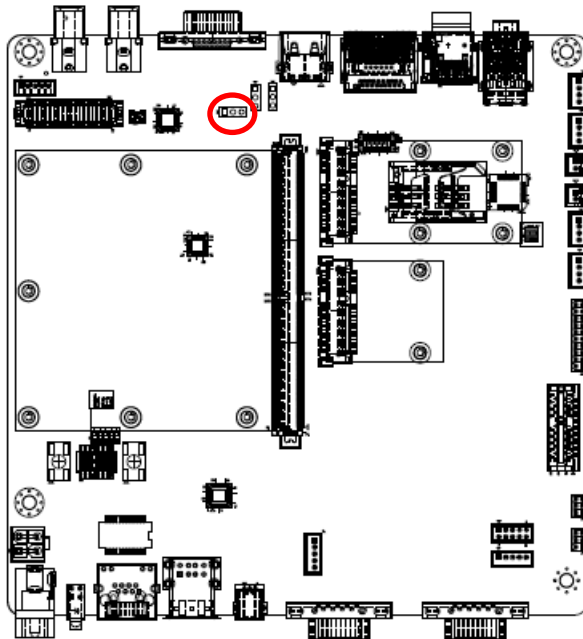
2.3.2 RS485-2 selector (JRS485-2)



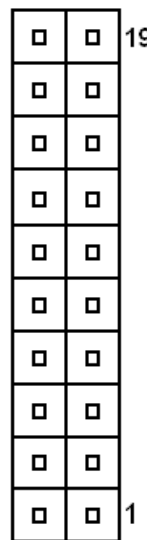
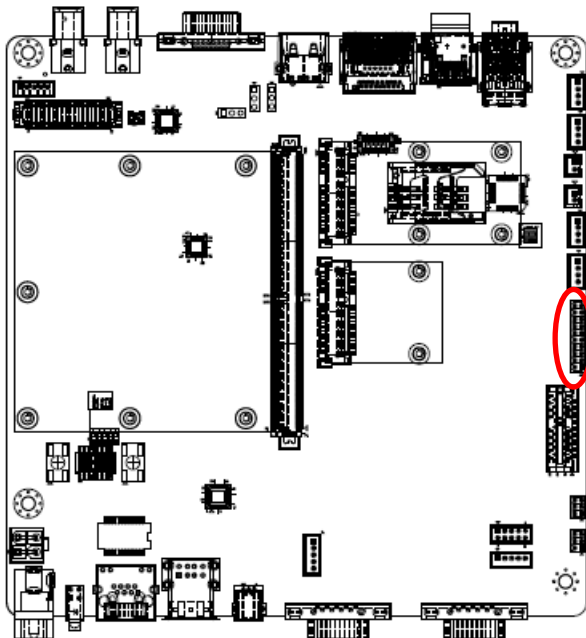
2.3.3 Camera input selector (JPCAM1)



### 2.3.4 LCD brightness PWM voltage select (JPWM1)



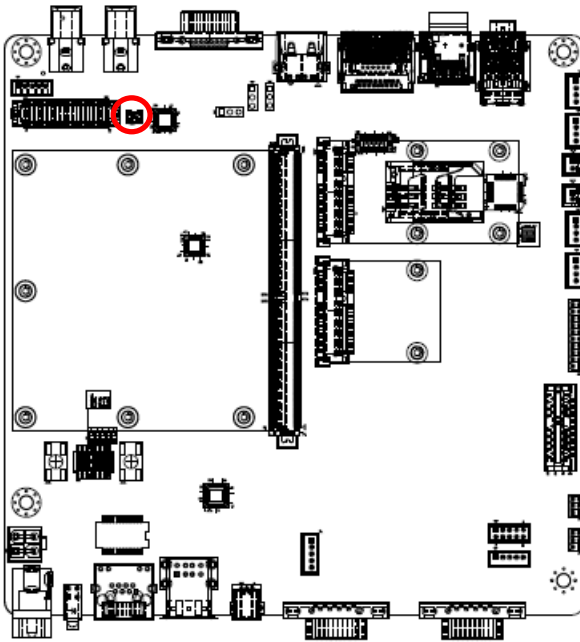
### 2.3.5 General purpose I/O connector (JGPIO1)



Signal	PIN	PIN	Signal
+5V	20	19	GND
DIO_SDA	18	17	DIO_SCL
DO7	16	15	DI7
DO6	14	13	DI6
DO5	12	11	DI5
DO4	10	9	DI4
DO3	8	7	DI3
DO2	6	5	DI2
DO1	4	3	DI1
DO0	2	1	DI0

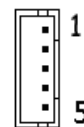
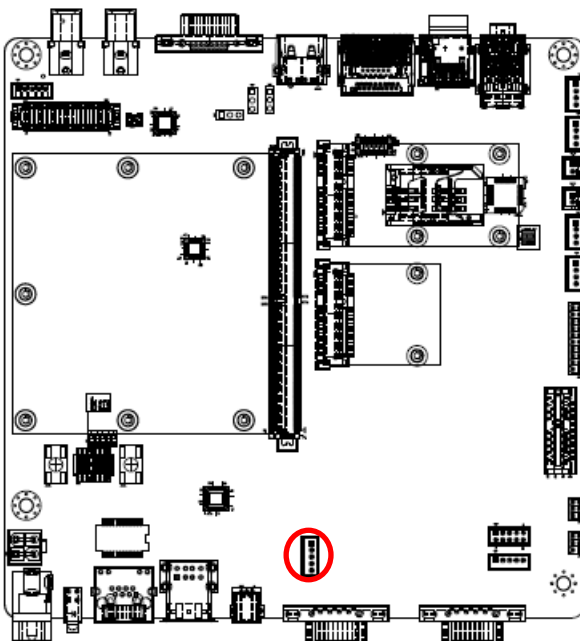


2.3.6 RTC Battery connector (JBAT1)



Signal	PIN
+V_BAT	1
GND	2

2.3.7 USB2.0 connector (JUSB5)

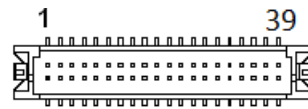
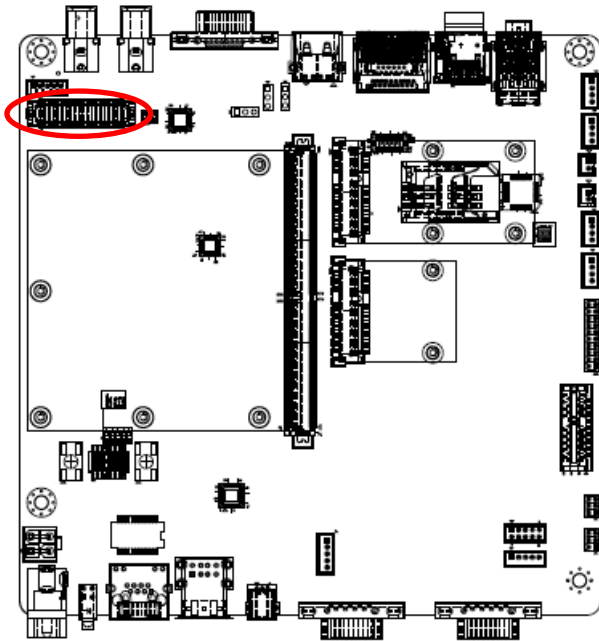


Signal	PIN
+5V	1
USB_NP5	2
USB_PP5	3
GND	4
GND	5

**Note:**

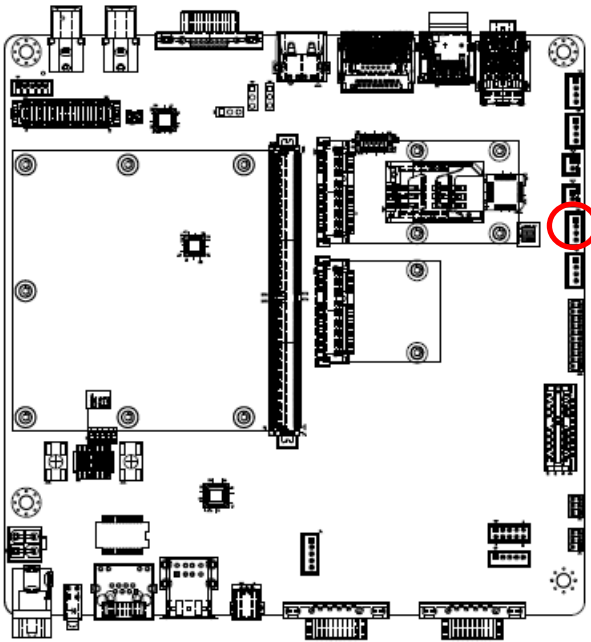
Not compatible with SMA-IMX6.

2.3.8 LVDS Interface connector (JLVDS1)



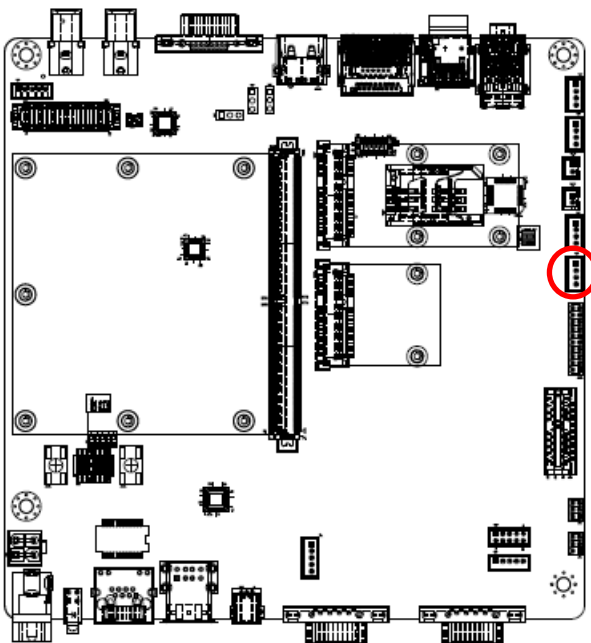
Signal	PIN	PIN	Signal
+5V	2	1	+3.3V
+5V	4	3	+3.3V
I2C_LCD_DAT	6	5	I2C_LCD_CK
GND	8	7	GND
LVDS0+	10	9	LVDS1+
LVDS0-	12	11	LVDS1-
GND	14	13	GND
LVDS2+	16	15	LVDS3+
LVDS2-	18	17	LVDS3-
GND	20	19	GND
NC	22	21	NC
NC	24	23	NC
GND	26	25	GND
NC	28	27	NC
NC	30	29	NC
GND	32	31	GND
LVDS_CK+	34	33	NC
LVDS_CK-	36	35	NC
GND	38	37	GND
+12V	40	39	+12V

2.3.9 CAN Bus connector 1 (JCAN1)



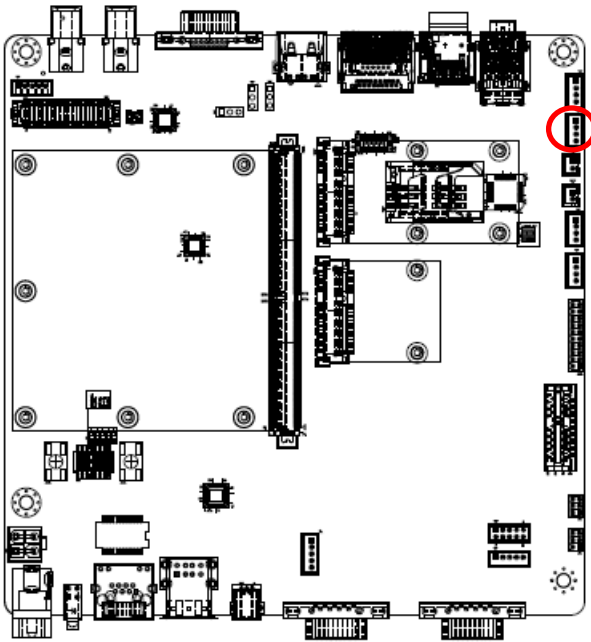
Signal	PIN
CAN0_H	1
GND	2
CAN0_L	3
GND	4

2.3.10 CAN Bus connector 2 (JCAN2)



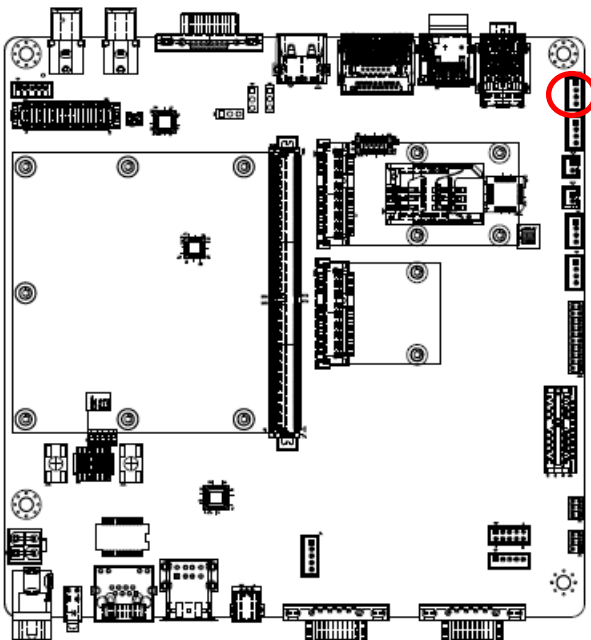
Signal	PIN
CAN1_H	1
GND	2
CAN1_L	3
GND	4

2.3.11 DMIC\_WM8962 connector (JDMIC-W1)



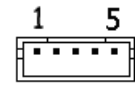
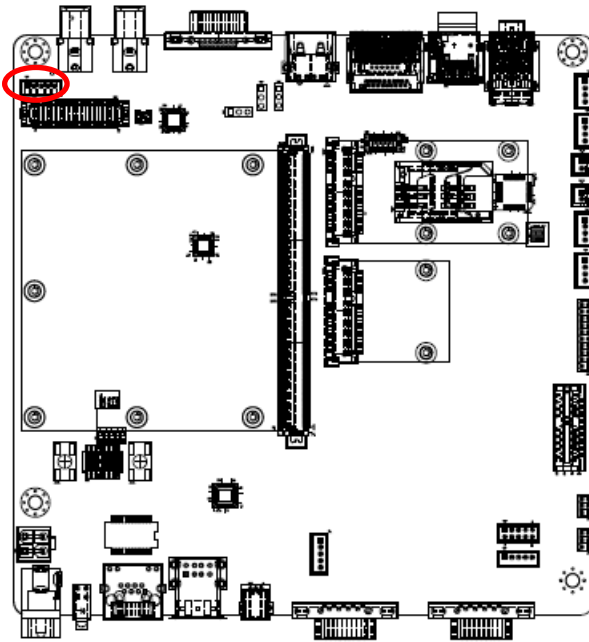
Signal	PIN
+3.3V	1
WM_DMIC_CLK_V3S	2
WM_DMIC_DAT_V3S	3
GND	4

2.3.12 DMIC\_ALC892 connector (JDMIC-A1)



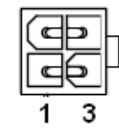
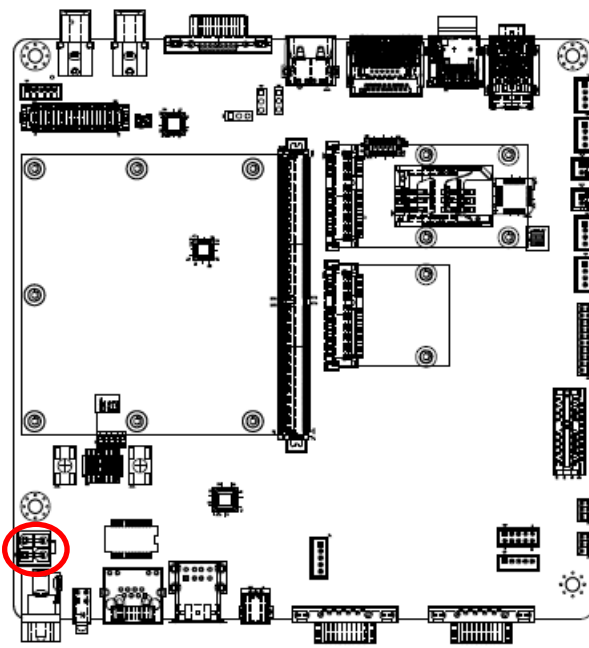
Signal	PIN
+3.3V	1
ALC_DMIC_CLK	2
ALC_DMIC_DAT	3
GND	4

2.3.13 Inverter connector (JINV1)



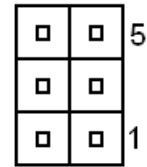
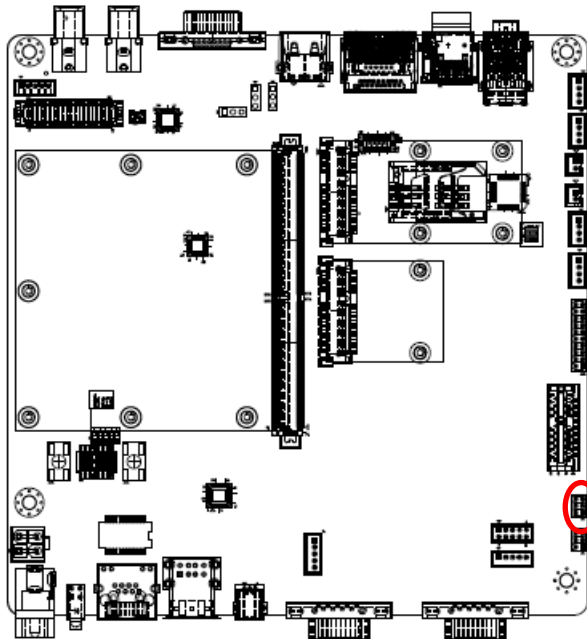
Signal	PIN
+12V	1
GND	2
BKLT_EN	3
INV_PWM	4
+5V	5

2.3.14 Power connector (JPWR1)



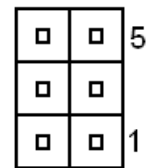
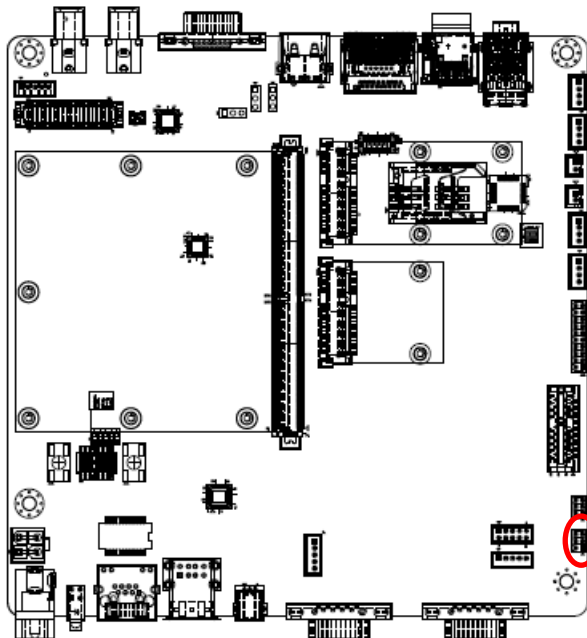
Signal	PIN	PIN	Signal
GND	2	4	+12V
GND	1	3	+12V

2.3.15 Serial port 1 in RS-485 mode (J485-1)



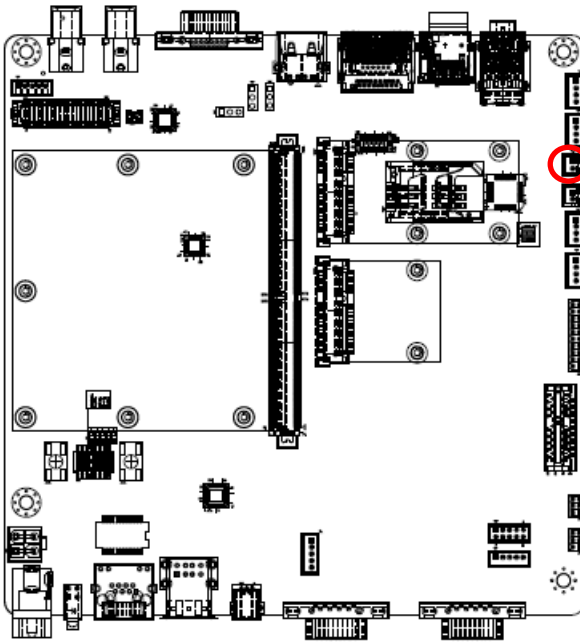
Signal	PIN	PIN	Signal
GND	6	5	+5V
NC	4	3	485TX1+
NC	2	1	485TX1-

2.3.16 Serial port 2 in RS-485 mode (J485-2)



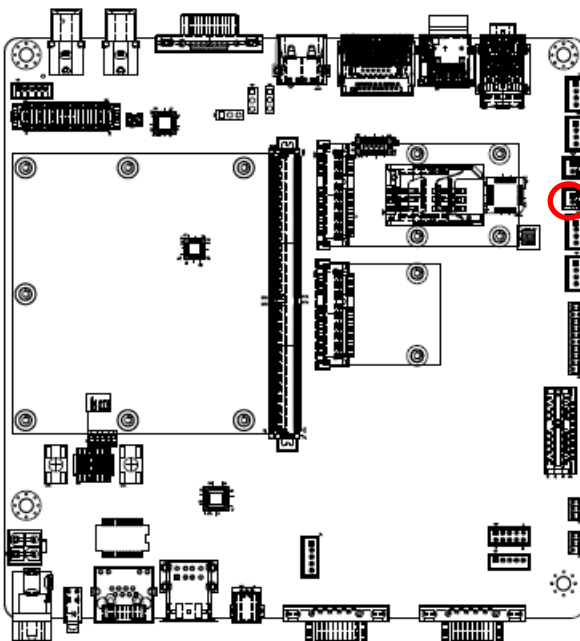
Signal	PIN	PIN	Signal
GND	6	5	+5V
NC	4	3	485TX2+
NC	2	1	485TX2-

### 2.3.17 Speaker connector (JSPR1)



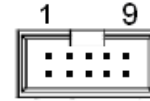
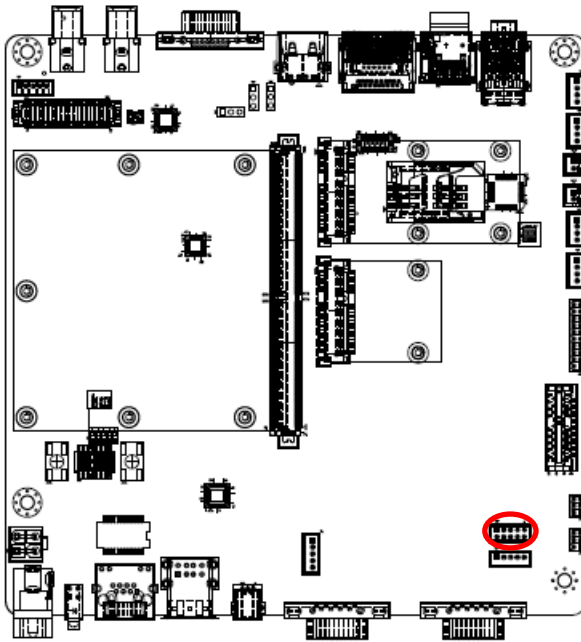
Signal	PIN
SPKR_N	1
SPKR_P	2

### 2.3.18 Speaker connector (JSPL1)



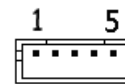
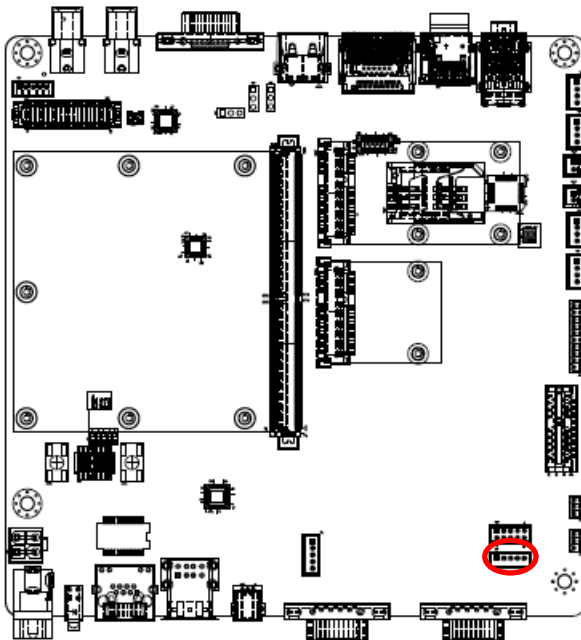
Signal	PIN
SPKL_N	1
SPKL_P	2

2.3.19 Battery connector (JBATC1)



Signal	PIN	PIN	Signal
+V_BATTERY	1	2	+V_BATTERY
+V_BATTERY	3	4	GND
GND	5	6	BAT_DATA
BATTS_R	7	8	BAT_CLK
GND	9	10	GND

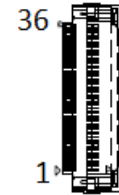
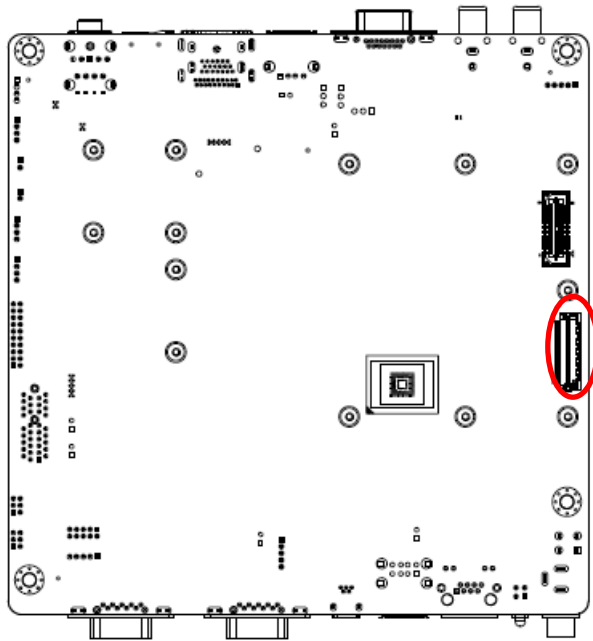
2.3.20 I2C connector (JI2C1)



Signal	PIN
+3.3C	1
I2C_GP_CK_V3S	2
I2C_GP_DAT_V3S	3
I2C_INT	4
GND	5



2.3.21 Camera connector (JMIP11)



Signal	PIN
NC	36
NC	35
GND	34
NC	33
NC	32
NC	31
NC	30
NC	29
NC	28
NC	27
NC	26
GND	25
NC	24
NC	23
NC	22
CAM_MCK	21
CSI_PWRDOWN	20

Signal	PIN
I2C_MIPI_DAT	19
I2C_MIPI_CK	18
GND	17
CSI0_CK-	16
CSI0_CK+	15
GND	14
NC	13
NC	12
CAM0_RST#	11
NC	10
NC	9
GND	8
CSI0_D1-	7
CSI0_D1+	6
GND	5
CSI0_D0-	4
CSI0_D0+	3
+3.3V	2
+3.3V	1

# 3. Linux User Guide

---

### 3.1 Download Source code for building Ubuntu image file

Please make a folder for storing the source code first then typing the command below to get started for the source code download.

```
$ Sudo apt-get install git
$ git clone guest@202.55.227.57:BSP-Source-Code/Freescale-Kernel.git -b 3.0.35-4.0.0
$ cd Freescale-Kernel/
$ git clone gitlab@aes.avalue.com.tw:REV-SA02/REV-SA02-KERNEL-PATCH.git -b
3.0.35-4.0.0
$ git am REV-SA02-KERNEL-PATCH/*.patch
```

About Password Please check with Avalue Sales or PM to get password.

### 3.2 Set up a Linux host for building U-boot & Kernel Image

We support to compile u-boot & Kernel on Ubuntu 12.04 (64bit version), other version of Ubuntu is not currently supported and may have built issues.

Install host packages needed by building code. This document assumes you are using Ubuntu. Not a requirement, but the packages may be named differently and the method of installing them may be different.

```
sudo apt-get install ia32-libs
sudo apt-get install uboot-mkimage
```

### 3.3 Building up U-boot & Kernel image

You can follow up the steps below to compile the u-boot & Kernel after downloading the source code.

1. Please move to folder "Freescale-Kernel" then start to compile both the u-boot & Kernel.

```
$ cd Freescale-Kernel/
```

```
avalue@avalue-virtual-machine:~$ ls
Freescale-Kernel
avalue@avalue-virtual-machine:~$ cd Freescale-Kernel/
avalue@avalue-virtual-machine:~/Freescale-Kernel$
```

2. Type the command to compile both u-boot & Kernel.

```
$ make rev-sa02 -j number
```

(-j **number** means multi jobs for more efficient building, you can add it according to your CPU performance of PC, e.g. mine is "-j16" as below ).

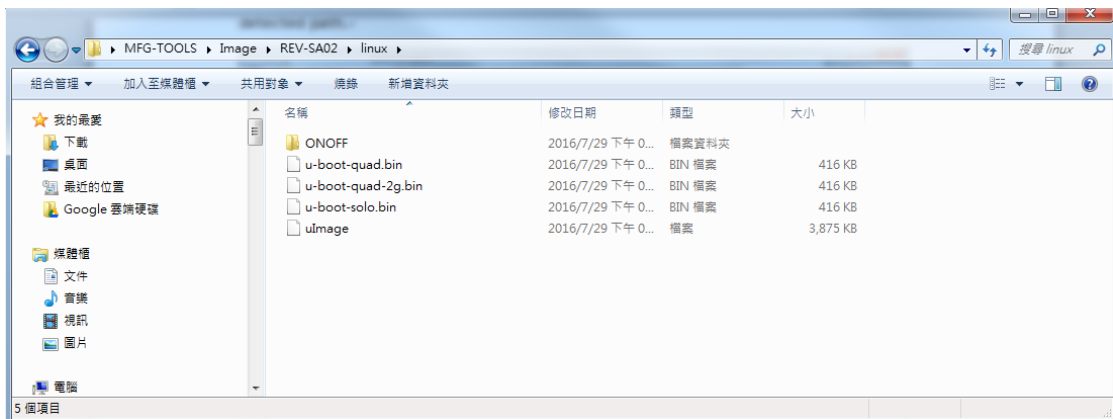
```
avalue@avalue-virtual-machine:~/Freescale-Kernel$ make rev-sa02 -j16
```

## REV-SA02 User's Manual

3. You can find the u-boot(u-boot-quad-2g.bin, u-boot-quad.bin and u-boot-solo.bin) & Kernel(ulmage) under folder "Freescale-Kernel" as below after the compiling is finish.

```
avalue@avalue-virtual-machine:~/Freescale-Kernel$ ls
fsl      Makefile      u-boot      u-boot-quad.bin  uImage
kernel  REV-SA02-KERNEL-PATCH  u-boot-quad-2g.bin  u-boot-solo.bin
```

PS: If you would like to use **Mfgtool** for flashing image file, you must put the file u-boot.bin and ulmage under "MFG-TOOLS\Image\REV-SA02\linux" for right detected path.

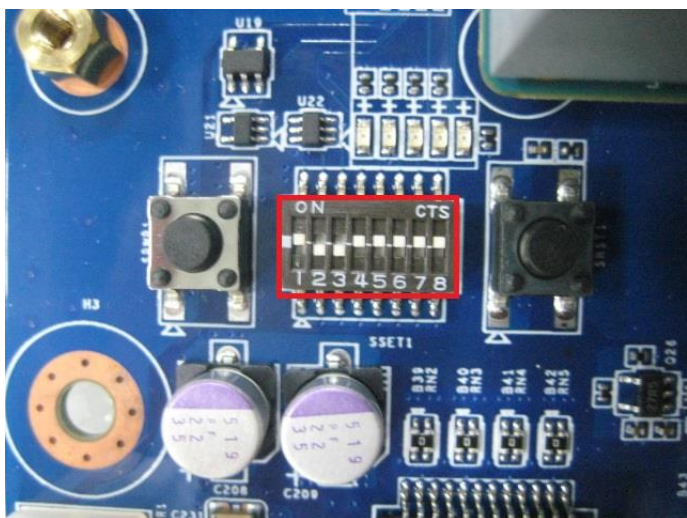


### 3.4 Use MfgTool to flash Ubuntu into onboard eMMC

Manufacturing tool, a successor of ATK, provides a series of new features to power your mass production work. The features like windows style GUI, multiple devices support, explicit status monitoring, versatile functionalities and highly flexible architecture make it a best choice to meet your critical timing, cost and customization requirements.

For using Mfgtool to flash image file into onboard eMMC, please follow up the steps below

1. Please turn on the **Pin4** of the DIP switch as below into burning mode of Mfgtool.



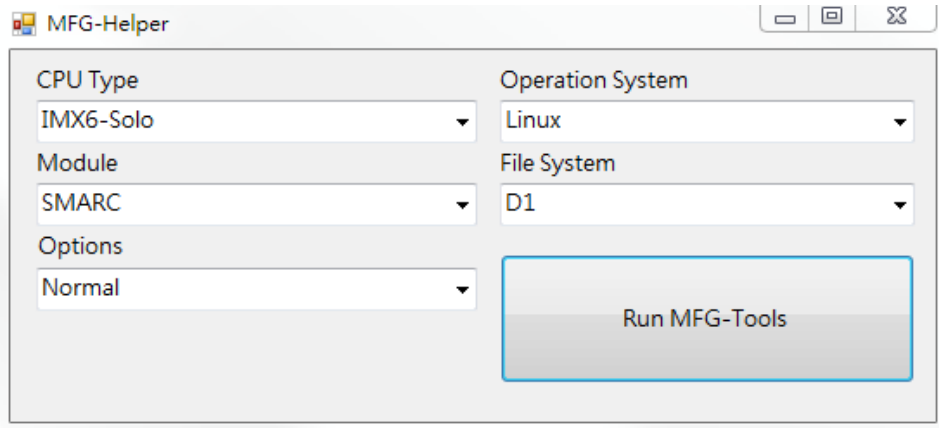
2. Power on the mainboard then plug the cable from OTG socket to PC.

3. Click the folder "~\MFG-Tools", e.g. mine is  
"E:\MFG-TOOLS\Mfg-REV-SA02"

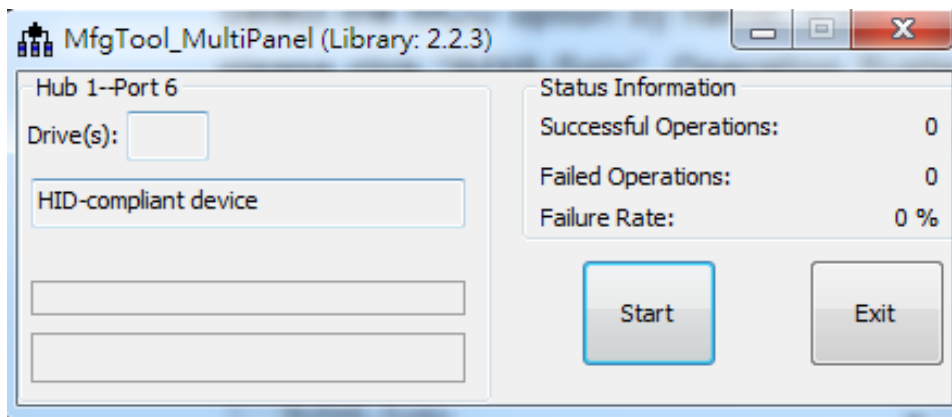
## REV-SA02 User's Manual

4. Click the “MfgLoader.exe”.

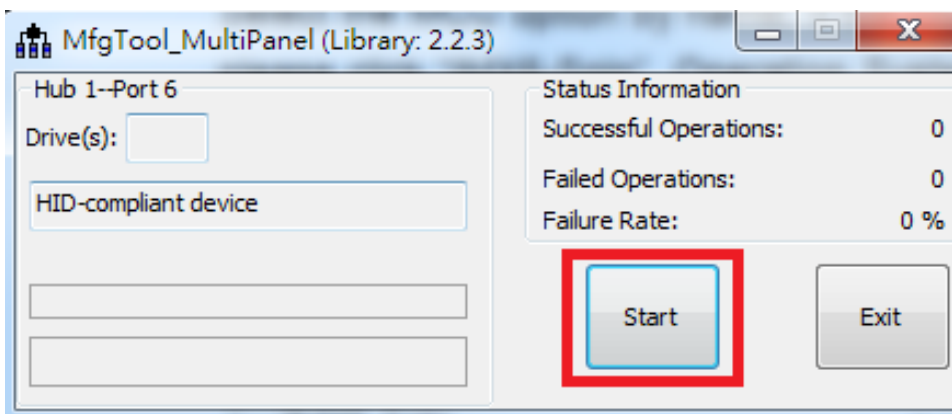
Select the MCU option by name, if the MCU of module board is “IMX6 Solo”, please click “IMX6-Solo”, Operation System select Linux for the OS of flashing, Module select “SMARC”, File System select “D1”, Options select “Normal” and the click “Run MFG-Tools”.



5. The second screen will show up after clicking “Run MFG Tool”, and please check whether it shows “HID-compliant device” as below, if not, please re-check the cable connection and DIP switch setting between mainboard and PC.

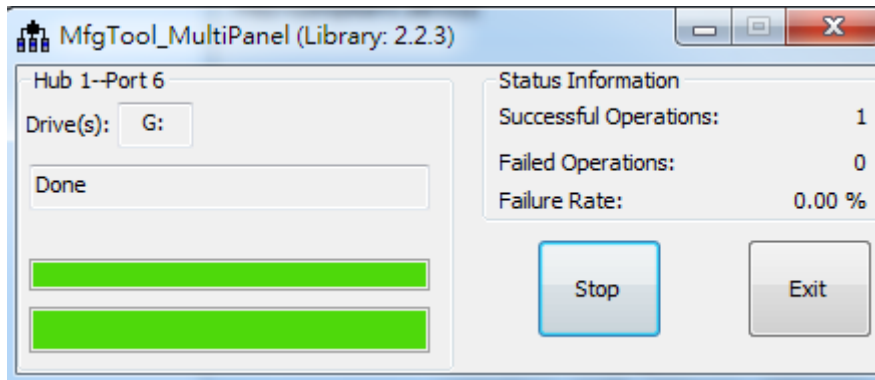


6. Click “Start” to flash image file.

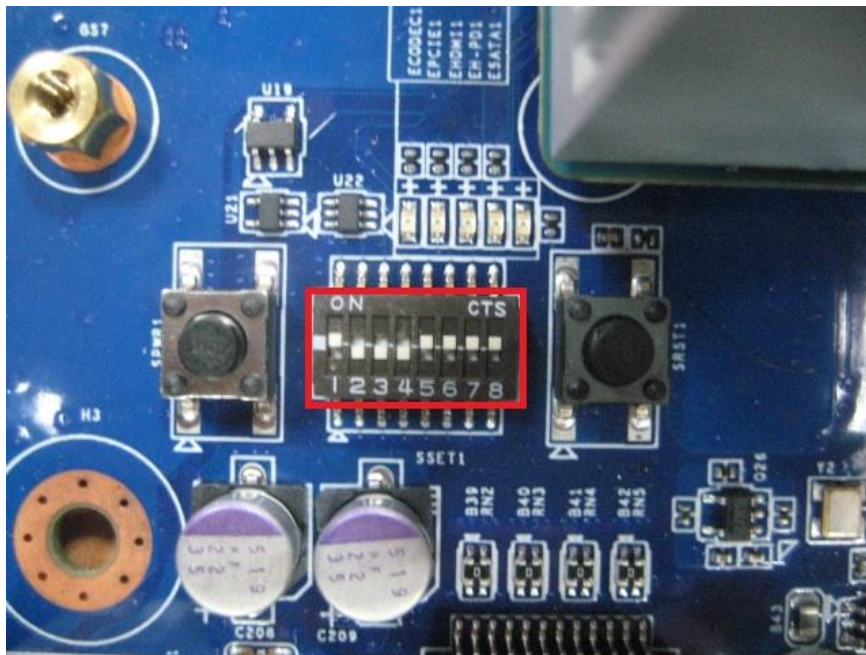


## REV-SA02 User's Manual

7. It will show “Done” after flashing is finish, then click “Stop” and “Exit” to close the screen.



8. Power off the system and turn on pin1 of DIP switch then reboot.



### 3.5 Download Android Source Code for building image file

Please make a folder for storing the source code first then typing the command below to get started for the source code download.

```
$ sudo apt-get install git
$ git clone guest@202.55.227.57:freescall/imx6/Android.git -b 4.4.2-pos
$ cd Freescale-Android/
$ git clone gitlab@aes.avalue.com.tw:REV-SA02/REV-SA02-ANDROID-PATCH.git -b
4.4.2
$ git am REV-SA02-ANDROID-PATCH/*.patch
```

About Password Please check with Avalue Sales or PM to get password.

### 3.6 Set up for building Android image file

We support to compile u-boot & Kernel on Ubuntu 12.04 (64bit version), other version of Ubuntu is not currently supported and may have built issues.

Install host packages needed by building code. This document assumes you are using Ubuntu. Not a requirement, but the packages may be named differently and the method of installing them may be different.

- 1) Please follow up the commands below to install "Oracle JDK6.0" first for building up Android image file.

```
$ sudo apt-get install python-software-properties
$ sudo add-apt-repository ppa:webupd8team/java
$ sudo apt-get update $ sudo apt-get install oracle-java6-installer
$ sudo update-alternatives --config java
```

- 2) Please follow up the commands below to install the necessary package for build image file.

```
libc6-dev libncurses5-dev x11proto-core-dev gcc g++ \ libx11-dev:i386
libreadline6-dev:i386 \ libgl1-mesa-dev g++-multilib mingw32 openjdk-6-jdk tofrodos \
python-markdown libxml2-utils xsltproc zlib1g-dev:i386 \ ia32-libs u-boot-tools minicom
lib32ncurses5-dev \ uuid-dev liblzo2-dev libz-dev\
```

### 3.7 Building up Android image file

You can follow up the steps below to compile Android image file after download the source code.

1. Please move to the folder "Freescale-Android" then start to compile image file.
2. Type the command to compile image file. \$ ./run.sh -j16 (-j number means multi jobs for more efficient building, you can add it according to your CPU performance of PC, e.g. mine is "-j16" as below ).

```

avalue@avalue-virtual-machine:~$ cd Freescale-Android/
avalue@avalue-virtual-machine:~/Freescale-Android$ ls
abi          dalvik      frameworks  ndk          run.sh
art          developers  hardware     out          sdk
bionic      development kernel_imx   packages     system
bootable    device      libcore      pdk          tools
build       docs        libnativehelper prebuilts
cts         external    Makefile     REV-SA02-ANDROID-PATCH
avalue@avalue-virtual-machine:~/Freescale-Android$ ./run.sh -j16
    
```

3. You can find the finished image file(u-boot-6dl.bin, u-boot-6q2g.bin, u-boot-6q.bin, u-boot.bin, ulmage, system.img, recover.img, boot.img) as below after compiling on the directory ~ /Freescale-Android/out/target/product/smarc.

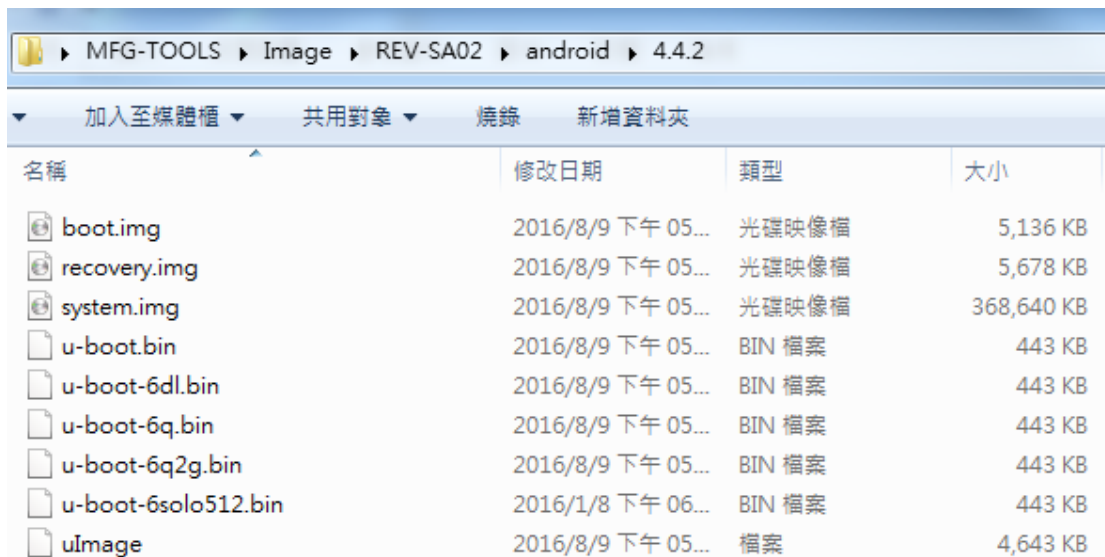
```

avalue@avalue-virtual-machine:~/Freescale-Android/out/target/product/smarc$ ll
total 392276
drwxrwxr-x 9 avalue avalue 4096 Aug 9 17:26 ./
drwxrwxr-x 3 avalue avalue 4096 Aug 9 17:18 ../
-rw-rw-r-- 1 avalue avalue 12 Aug 9 17:26 android-info.txt
-rw-rw-r-- 1 avalue avalue 5259264 Aug 9 17:18 boot.img
-rw-rw-r-- 1 avalue avalue 40703 Aug 9 17:26 clean_steps.mk
-rwxr-xr-x 1 avalue avalue 84819 Aug 9 17:18 .config*
drwxrwxr-x 4 avalue avalue 4096 Aug 9 17:19 data/
drwxrwxr-x 2 avalue avalue 4096 Aug 9 17:18 fake_packages/
-rw-rw-r-- 1 avalue avalue 57652 Aug 9 17:18 installed-files.txt
-rwxr-xr-x 1 avalue avalue 4753448 Aug 9 17:19 kernel*
drwxrwxr-x 14 avalue avalue 4096 Aug 9 17:26 obj/
-rw-rw-r-- 1 avalue avalue 457 Aug 9 17:18 previous_build_config.mk
-rw-rw-r-- 1 avalue avalue 499924 Aug 9 17:19 ramdisk.img
-rw-rw-r-- 1 avalue avalue 1054767 Aug 9 17:18 ramdisk-recovery.img
drwxrwxr-x 3 avalue avalue 4096 Aug 9 17:19 recovery/
-rw-rw-r-- 1 avalue avalue 5814272 Aug 9 17:26 recovery.img
drwxrwxr-x 9 avalue avalue 4096 Aug 9 17:18 root/
drwxrwxr-x 5 avalue avalue 4096 Aug 9 17:19 symbols/
drwxrwxr-x 13 avalue avalue 4096 Aug 9 17:19 system/
-rw-rw-r-- 1 avalue avalue 377487360 Aug 9 17:19 system.img
-rwxr-xr-x 1 avalue avalue 453232 Aug 9 17:18 u-boot-6dl.bin*
-rwxr-xr-x 1 avalue avalue 453504 Aug 9 17:19 u-boot-6q2g.bin*
-rwxr-xr-x 1 avalue avalue 453472 Aug 9 17:19 u-boot-6q.bin*
-rwxr-xr-x 1 avalue avalue 453232 Aug 9 17:19 u-boot.bin*
-rwxr-xr-x 1 avalue avalue 4753512 Aug 9 17:26 uImage*
avalue@avalue-virtual-machine:~/Freescale-Android/out/target/product/smarc$
    
```

PS: If you would like to use Mfgtool for flashing image file into mainboard, you must put all the files u-boot-6dl.bin, u-boot-6q2g.bin, u-boot-6q.bin, u-boot.bin, ulmage, system.img, recover.img, boot.img under the path "~\POS-Android-MfgTools\Image\POS\Android" on MFGTool folder for right detected.



## REV-SA02 User's Manual



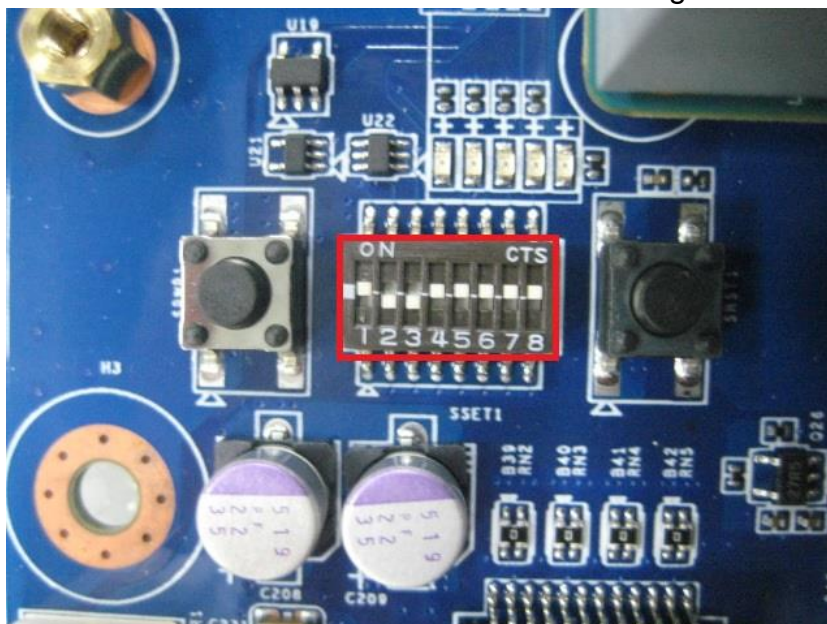
名稱	修改日期	類型	大小
boot.img	2016/8/9 下午 05...	光碟映像檔	5,136 KB
recovery.img	2016/8/9 下午 05...	光碟映像檔	5,678 KB
system.img	2016/8/9 下午 05...	光碟映像檔	368,640 KB
u-boot.bin	2016/8/9 下午 05...	BIN 檔案	443 KB
u-boot-6dl.bin	2016/8/9 下午 05...	BIN 檔案	443 KB
u-boot-6q.bin	2016/8/9 下午 05...	BIN 檔案	443 KB
u-boot-6q2g.bin	2016/8/9 下午 05...	BIN 檔案	443 KB
u-boot-6solo512.bin	2016/1/8 下午 06...	BIN 檔案	443 KB
uImage	2016/8/9 下午 05...	檔案	4,643 KB

### 3.8 Use MfgTool to flash Android into onboard eMMC

Manufacturing tool, a successor of ATK, provides a series of new features to power your mass production work. The features like windows style GUI, multiple devices support, explicit status monitoring, versatile functionalities and highly flexible architecture make it a best choice to meet your critical timing, cost and customization requirements.

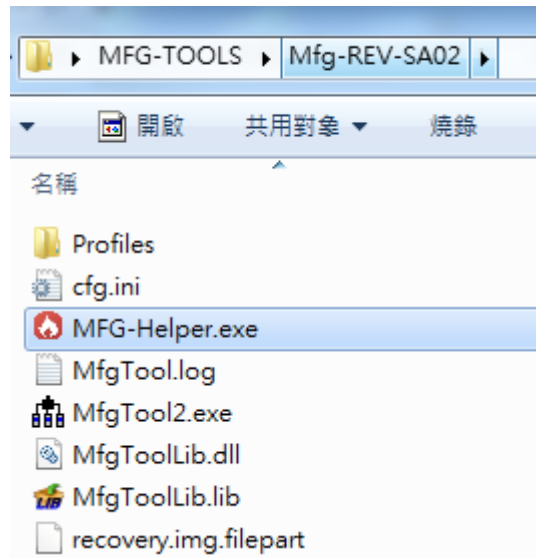
For using Mfgtool to flash image file into onboard eMMC, please follow up the steps below

1. Please turn on the **Pin4** of the DIP switch as below into burning mode of Mfgtool.

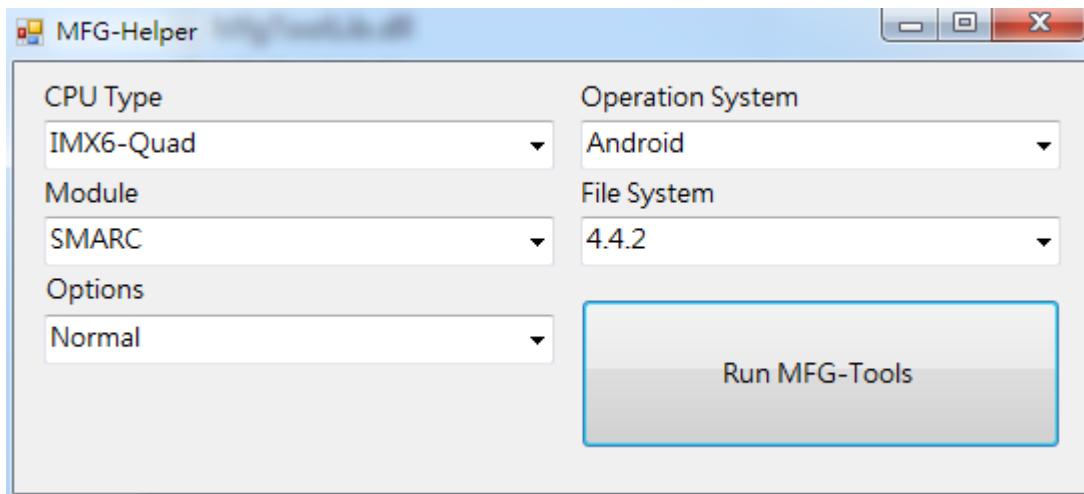


- 1) Power on the mainboard then plug the cable from OTG socket to PC.
- 2) Please click "MFG-Helper" under the path "~MFG-TOOLS\Mfg-REV-SA02"

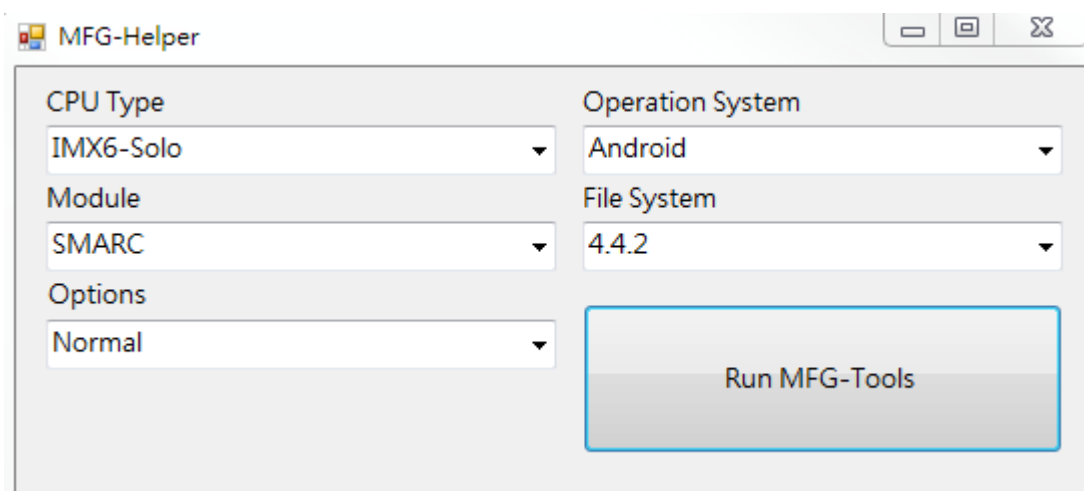
## REV-SA02 User's Manual



3) If the CPU of the mainboard is Quadcore version, please select as below, then click “Run MFG-Tools”.

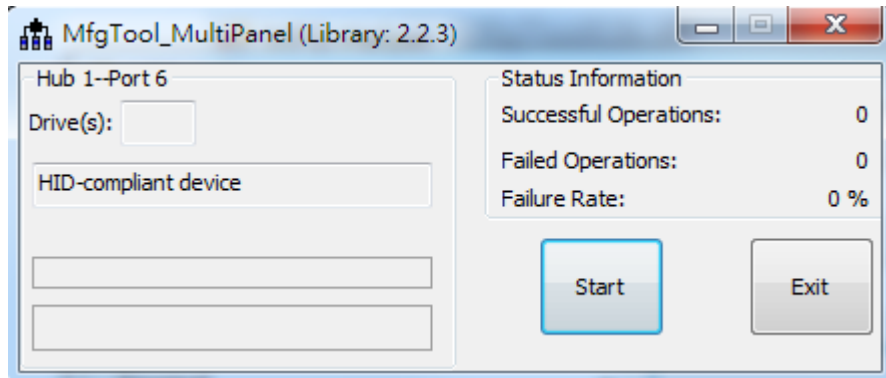


4) If the CPU of the mainboard is DualLite version, please select as below, then click

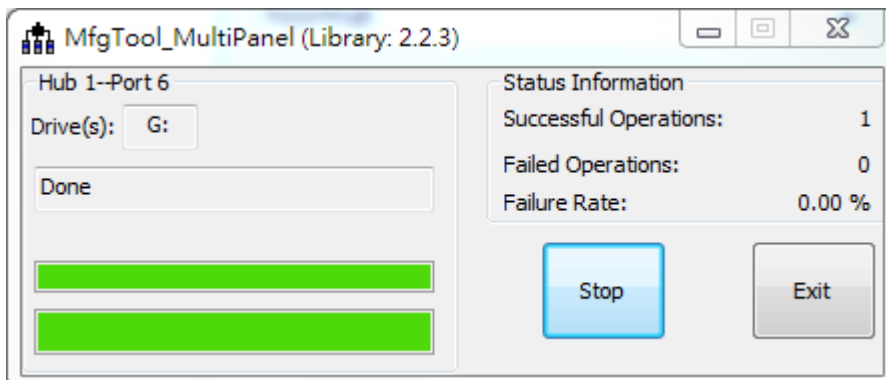


## REV-SA02 User's Manual

5) Click "Start" to flash image file.



6) It will show "Done" after flashing is finish, then click "Stop" and "Exit" to close the screen.



7) Power off the system and turn on pin1 of DIP switch then reboot.

