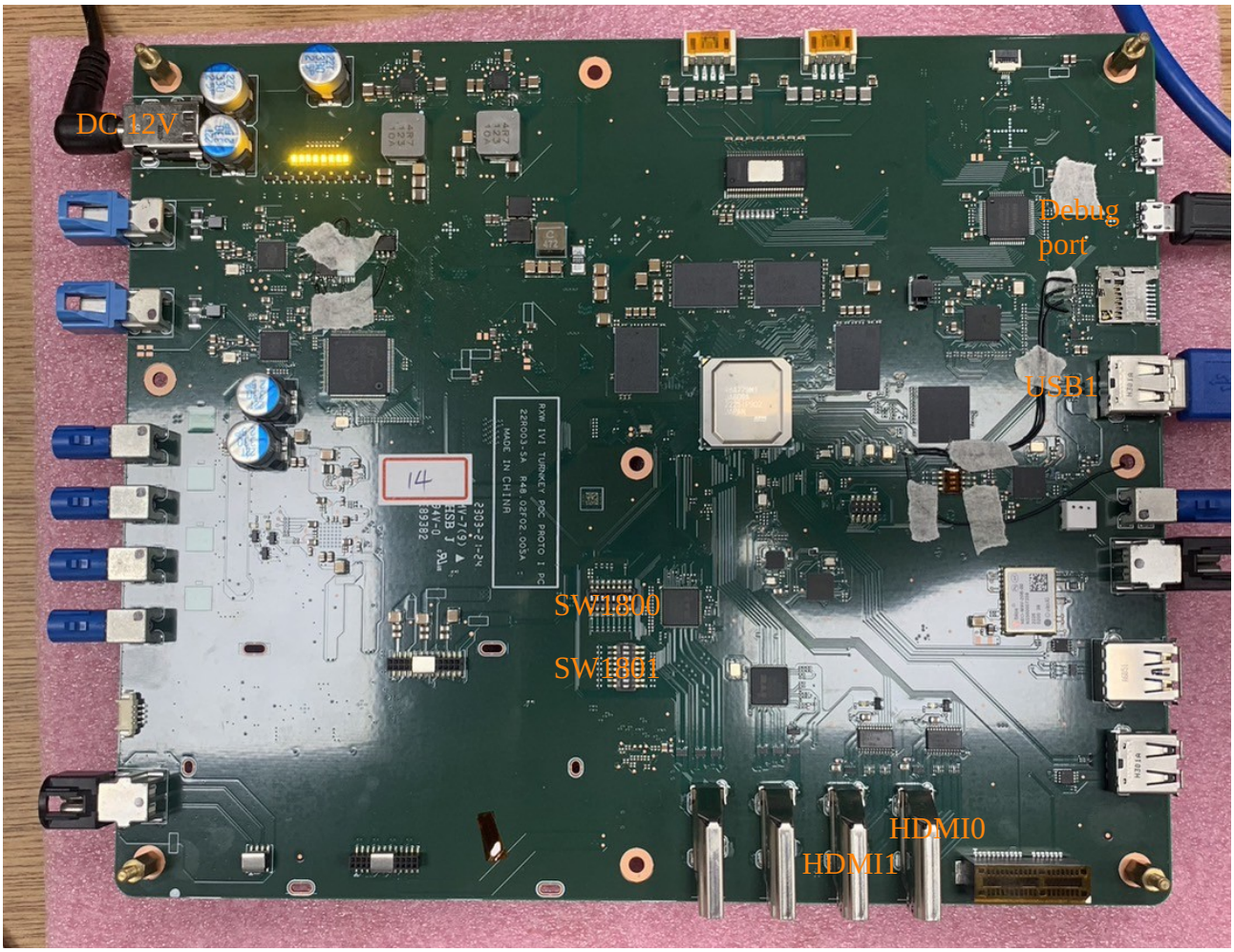
# RTX-IVI

## Start-Up Guide for Android

v0.0.2

## PCBA port:




## USB cables:

# 1. Buiding Android10

## 1.1. Tools & Dependency packages

Prerequisite packages for building the Android Filesystem (Note: This is with reference to Ubuntu 18.04 64-bit). Ubuntu 64-bit is required for the cross-compilation of Android Filesystem.

**Setup build environment according to Google Android setup guide:**
https://source.android.com/source/initializing.html#setting-up-a-linux-build-environment

## 1.2. Download source

Download source form https://github.com/RetronixTechInc/rcar-bsp.git
command :
$ git clone https://github.com/RetronixTechInc/rcar-bsp.git -b IVI_android10

## 1.3. Building Android, IPL, U-Boot, and Kernel sources

$ cd rcar-bsp  => Entry download fold.
$ ./build.sh => download source code / patch / building
That will create **IVI-16G-Android10** fold if the build is successful.

# 2.Write IPL

## 2.1 Enter Download Mode

Dip switch configuration for download mode

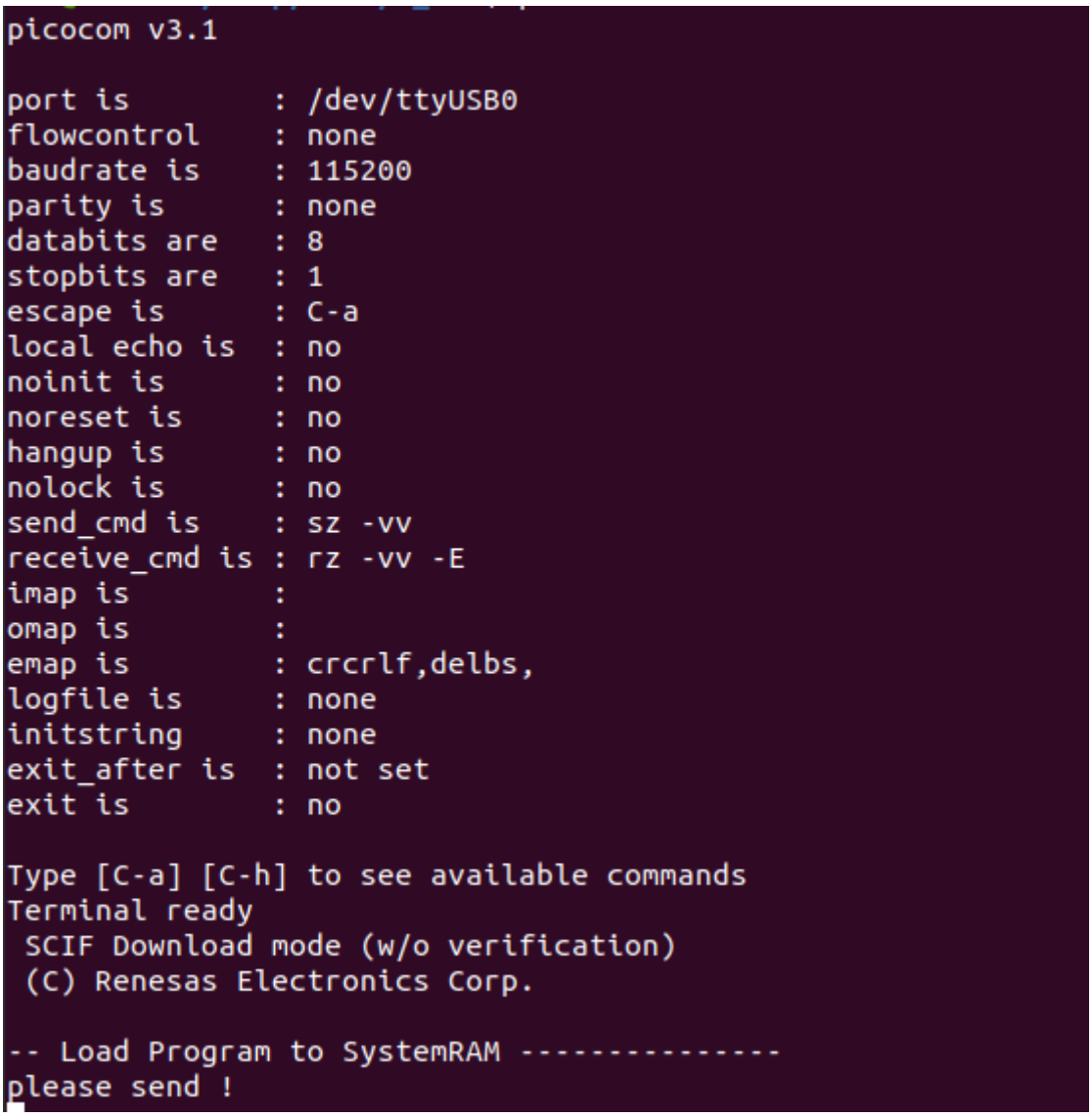
| SW Number | Pin1 | Pin2 | Pin3 | Pin4 | Pin5 | Pin6 | Pin7 | Pin8 |
|---|---|---|---|---|---|---|---|---|
| 1800 | ON | ON | ON | OFF | OFF | OFF | OFF | ON |
| 1801 | ON | ON | ON | ON | OFF | ON | ON | ON |

## 2.2 Connect USB cable to Host

micro usb cable connected debug port and PC.
Type A cable connected USB1 and PC.

## 2.3 Open Terminal (like picocom)

Execute picocom ap.
$ picocom b 115200 /dev/ttyUSB0
Plug in DC 12V then you can see Download mode message.




**2.3 Start write IPL**

Entry **IVI-16G-Android10** fold.

$ cd IVI-16G-Android10

**$ sudo ./write.sh**

Wait about 3 minutes. You can see message like bellow If finished.

**Load FlashWriter ./AArch64_Gen3_H3_M3_Scif_MiniMon_V5.13.mot.....**
**sudo dd if=./AArch64_Gen3_H3_M3_Scif_MiniMon_V5.13.mot**
**of=/dev/ttyUSB0**
**[sudo] password for tom:**
**1207+1 records in**
**….**
**6513+1 records out**
**3334786 bytes (3.3 MB, 3.2 MiB) copied, 36.0727 s, 92.4 kB/s**
**write finished!!!**

# 3.Write Android image

## 3.1 Enter Boot Mode

Dip switch configuration for boot mode

| SW Number | Pin1 | Pin2 | Pin3 | Pin4 | Pin5 | Pin6 | Pin7 | Pin8 |
|---|---|---|---|---|---|---|---|---|
| 1800 | ON | ON | ON | ON | OFF | ON | ON | ON |
| 1801 | ON | ON | ON | ON | OFF | ON | ON | ON |

## 3.2 Connect USB cable to Host

micro usb cable connected debug port and PC.
Type A cable connected USB1 and PC.

## 3.3 Open Terminal (like picocom)

Execute picocom ap.
$ picocom b 115200 /dev/ttyUSB0
Replug in DC 12V. Press enter key when you can see “Hit any key to stop autoboot” to interrupt autoboot.

### 3.4 Execute below commands on target board

**# Erase bootloader in eMMC (See “3.3. Boot sequence of IPL”)**
=> mmc dev 1 1
=> mw.b 4f000000 0 200000
=> mmc write 4f000000 0 1000
=> mmc dev 1 2
=> mw.b 4f000000 0 200000
=> mmc write 4f000000 0 1000
=> reset
**# Please interrupt autoboot**
**# Set environment values on U-boot**
=> env default -a
=> setenv ethaddr <board MAC addr>
=> editenv serialno
**Set board serial number to serialno: 0000XXXX (where XXXX = board number like 0585)**
=> editenv bootargs
**Edit bootargs: video=XXXX-X:d init_time=xxxxxxxxxx**
**“video” variable needs to set parameter related to display configuration. The default setting is below.**
**Salvator case:**
**video=LVDS-1:d video=VGA-1:d**
**“init_time” variable needs to set UNIX time.**
**You can get it by executing “date +%s” command on host PC.**
**The board don't have any RTC.**
**If time and date is not accurate, a few issues will be happened.**
=> saveenv
=> reset
Interrupt autoboot
=> fastboot

### 3.5 Execute below commands on host PC

$ sudo ./fastboot oem for mat
$ sudo ./fastboot reboot bootloader
$ sudo ./fastboot.sh --noresetenv

### 3.6 Start up to Android if flash images is finished.

Connect HDMI0 or HDMI1 to Monitor.

# Appendix 1:

**The reference document is RENESAS_RCH3M3M3N_Android_10_ReleaseNote_2020_09E.pdf**