USER'S MANUAL

SCM187

NXP i.MX 8M Mini ARM-based SMARC Module

Hardware & BSP User's Manual

AXIOMTEK®

www.axiomtek.com

Disclaimers

This manual has been carefully checked and believed to contain accurate information. Axiomtek Co., Ltd. assumes no responsibility for any infringements of patents or any third party's rights, and any liability arising from such use.

Axiomtek does not warrant or assume any legal liability or responsibility for the accuracy, completeness or usefulness of any information in this document. Axiomtek does not make any commitment to update the information in this manual.

Axiomtek reserves the right to change or revise this document and/or product at any time without notice.

No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Axiomtek Co., Ltd.

CAUTION

If you place the wrong batteries, it might cause the danger of explosion. It is recommended by the manufacturer that you follow the manufacturer's instructions to only replace the same or equivalent type of battery, and dispose the used ones.

December 2023,

Version A1.2

Printed in Taiwan

ESD Precautions

Computer boards have integrated circuits sensitive to static electricity. To prevent chipsets from electrostatic discharge damage, please take care of the following jobs with precautions:

- Do not remove the anti-static packaging from boards or integrated circuits until you are ready to install them.
- Before holding the board or integrated circuit, touch an unpainted portion of the system unit for a few seconds. It discharges static electricity from your body.
- Wear a wrist-grounding strap, available from most electronic component stores, when handling boards and components.

Trademarks Acknowledgments

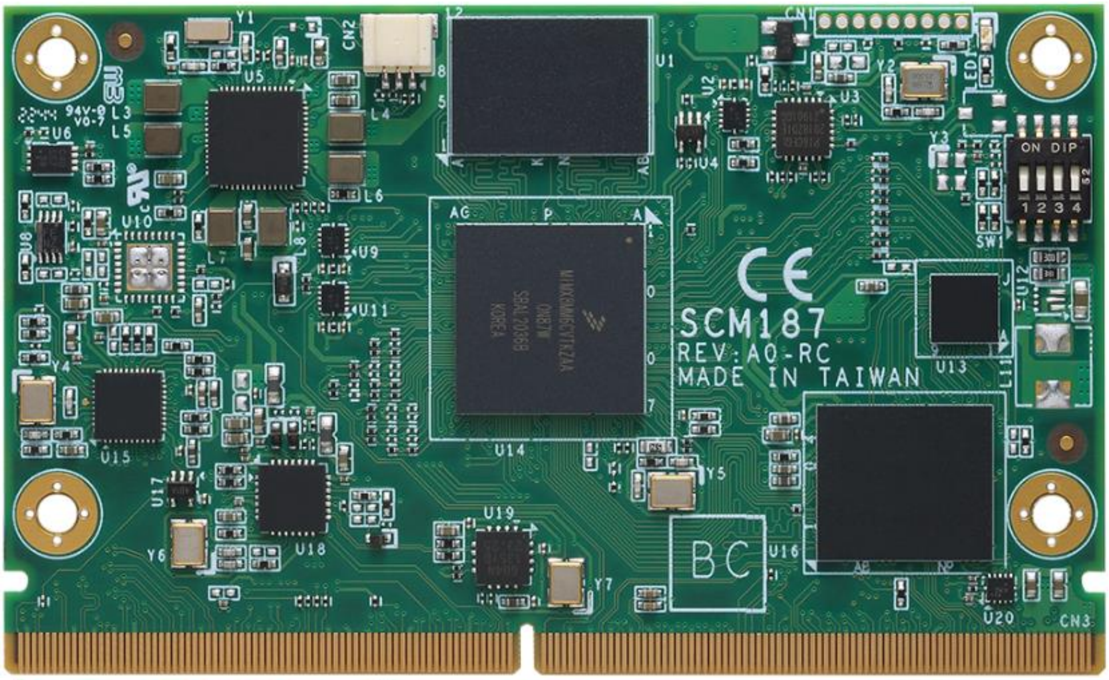
Axiomtek is a trademark of Axiomtek Co., Ltd.

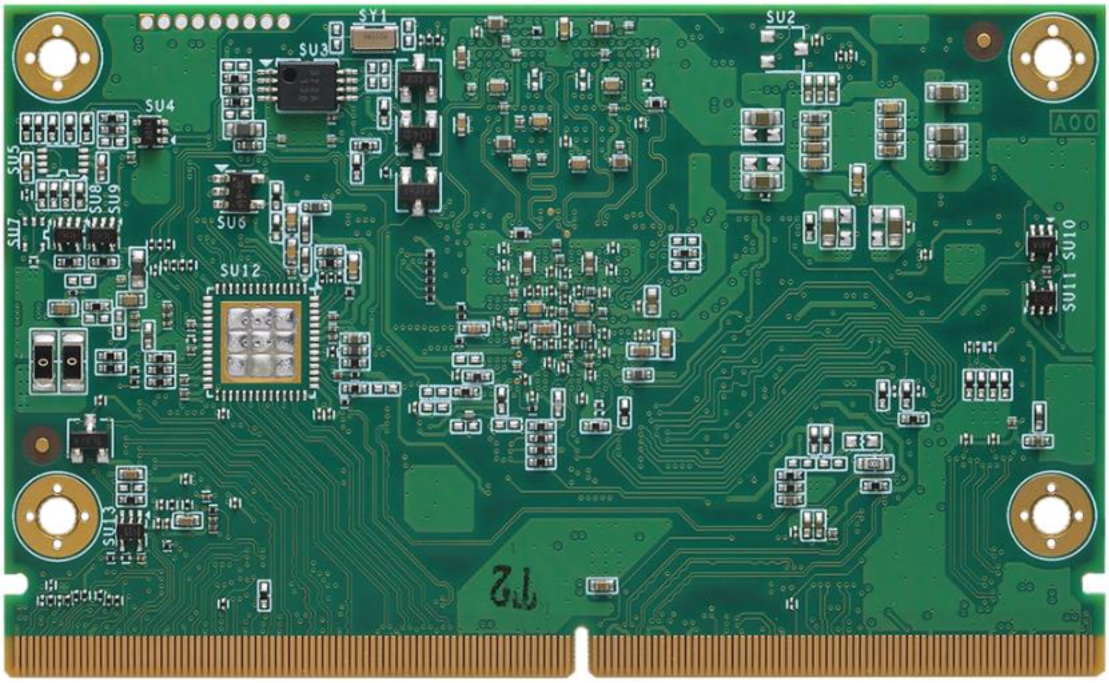
ARM is a trademark of ARM Ltd.

Other brand names and trademarks are the properties and registered brands of their respective owners.

Table of Contents

Section 1
Introduction





The SCM187 is a SMARC 2.1 module based on NXP i.MX 8M Mini SoC Quad A53 with onboard LPDDR4 RAM and eMMC. It integrates system memory, storage as embedded eMMC or SDIO interface, UART, audio, LAN, USB, dual-channel LVDS, MIPI-CSI, CANbus, GPIO, PCIe and various features. Taking the advantages of low power consumption of the ARM RISC architecture, the SCM187 is extremely suitable to be deployed in the embedded applications such as HMI, data logger, extended temperature embedded controllers, etc.

1.1 Features

- SMARC 2.1 (82 x 50 mm)
- NXP i.MX 8M Mini SoC with A53 Quad Core Processor
- Cortex-M4 for Real-time Control
- Dual CANbus
- Dual-channel 24-bit LVDS or HDMI 1.4
- 10/100/1000 Mbps Ethernet
- USB 2.0
- MIPI-CSI
- Audio
- Yocto 4.0

1.2 Specifications

- **CPU**
  SCM187 is available in three variants (two industrial and one commercial). Refer to the 1.4 Model List, the table shows the different configurations available.
  - NXP i.MX 8M Mini SoC with A53 Quad Core Processor

- **System Memory**
  SCM187 is available in three variants (two industrial and one commercial). Refer to the 1.4 Model List, the table shows the different configurations available.
  - LPDDR4 2GB / 4GB

- **Operating System**
  - Yocto 4.0, Linux Kernel 5.15.52

- **Storage**
  - SDIO3.0 interface
  - eMMC NAND flash chip on module; default is 8GB

- **Graphics**
  SCM187 is available in three variants (two industrial and one commercial). Refer to the 1.4 Model List, the table shows the different configurations available.
  - Dual-channel 24-bit LVDS or HDMI 1.4

- **Ethernet**
  - One supports 10/100/1000 Mbps data transfer rates

- **USB Interface**
  - Triple USB 2.0

- **COM**
  - Dual UART with Tx/Rx/RTS/CTS
  - Dual UART with Tx/Rx (for debug console)

- **I2C**
  - Triple I2C interface

- **QSPI**
  - Single QSPI

- **SPI**
  - Single SPI

- **CANbus**
  - Dual CANbus

- **GPIO**
  - 14 x GPIO

- **OS Flash**
  - Single OS Flash for flash image

- **Audio**
  - Single I2S interface

- **Camera Input**
  - Single MIPI-CSI 4-lane

- **Security**
  - TPM2.0 (option, none in default)

- **PCIe**
  - Single PCIe Gen2 x1

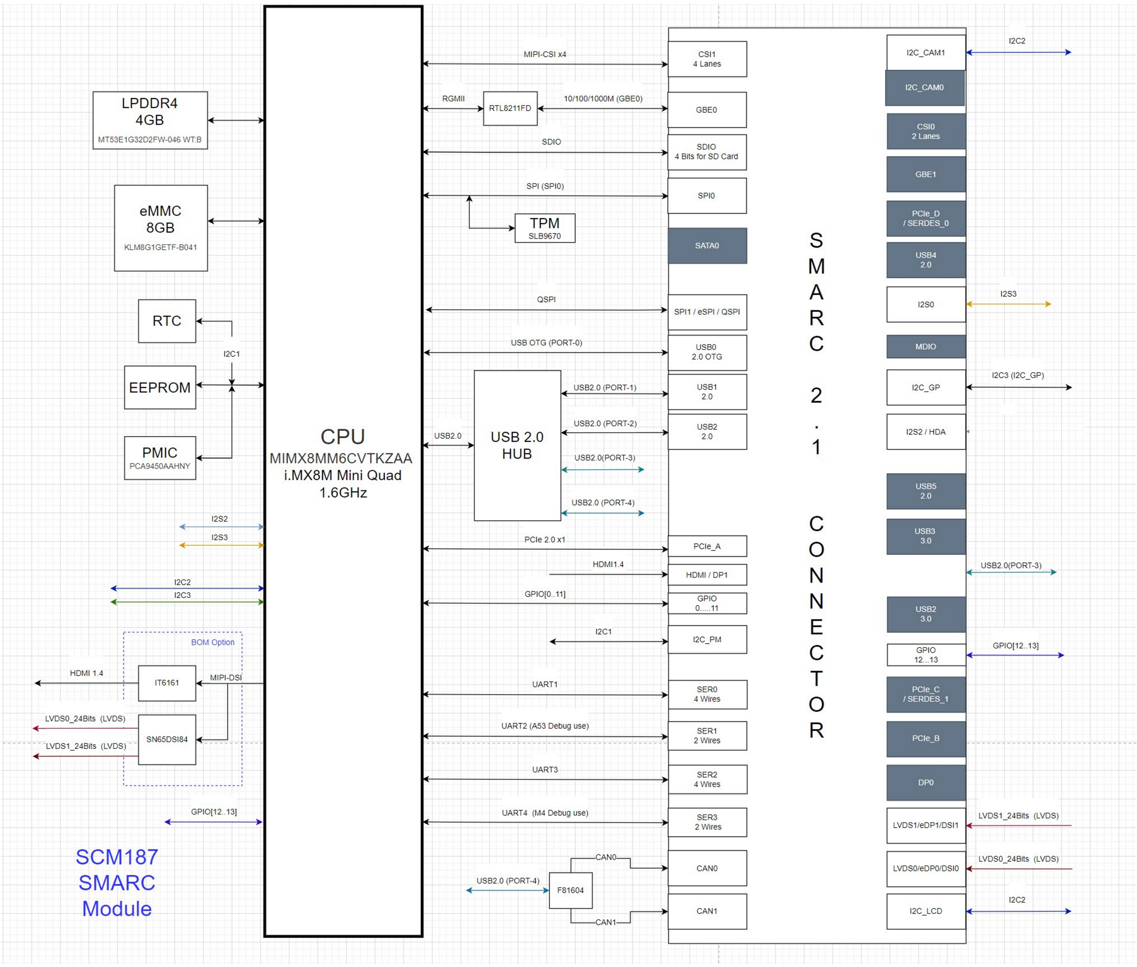
- **Power Requirement**
  - +5V ± 5%

- **Form Factor**
  - 82mm x 50mm (approx. 3.23” x 1.97”)
  - Thickness as 1.2mm ± 0.1mm
  - SMARC specifications V2.1 compliant

- **Environments**
  - Temperature (industrial variants)
    - Operating: -40°C to 85°C (-40°F to +185°F)
    - Storage: -40°C to 85°C (-40°F to +185°F)
  - Temperature (commercial variant)
    - Operating: 0°C to 70°C (-32ºF ~ 158ºF)
    - Storage: -40°C to 85°C (-40°F to +185°F)
  - Humidity:
    - Operating: 10% to 95%, non-condensing
    - Storage: 10% to 95%, non-condensing

<u>Note</u>: All specifications and images are subject to change without notice.

1.3 Block Diagram



1.4 Model List

SCM187-4R8E-IL (P/N: E38S187100)	SMARC Module, NXP i.MX 8M Mini 1.6GHz SoC Quad A53 with 4GB RAM, 8GB eMMC, industrial variant, and LVDS
SCM187-2R8E-IL (P/N: E38S187102)	SMARC Module, NXP i.MX 8M Mini 1.6GHz SoC Quad A53 with 2GB RAM, 8GB eMMC, industrial variant, and LVDS
SCM187-4R0E-CH (P/N: E38S187101)	SMARC Module, NXP i.MX 8M Mini 1.8GHz SoC Quad A53 with 4GB RAM, without eMMC, commercial variant, and HDMI

1.5 Optional Accessory List

SCB184-EMMC-AX (P/N: E39E184100)	Baseboard of Embedded SMARC Module, default boot mode at eMMC, for SMARC Module with eMMC onboard
SCB184-SD-AX (P/N: E39E184101)	Baseboard of Embedded SMARC Module, default boot mode at Micro SD card, for SMARC Module without eMMC onboard
ACC100-187-HS (P/N: E39R100103)	SCM187 Heatsink Kit (-40°C to +75°C)
ACC100-187-HSP (P/N: E39R100104)	SCM187 Heat Spreader Kit (This is used for installing a customized heatsink or a thermal chassis as a proper thermal interface for cooling solutions)

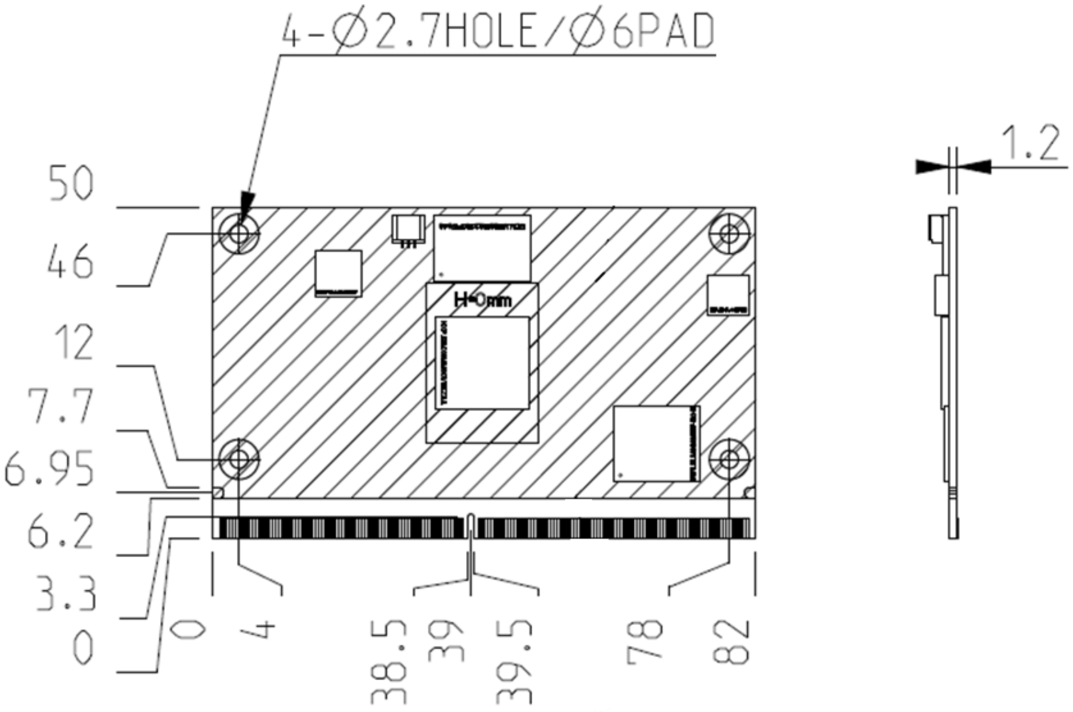
This page is intentionally left blank.

Section 2
Board and Pin Assignments

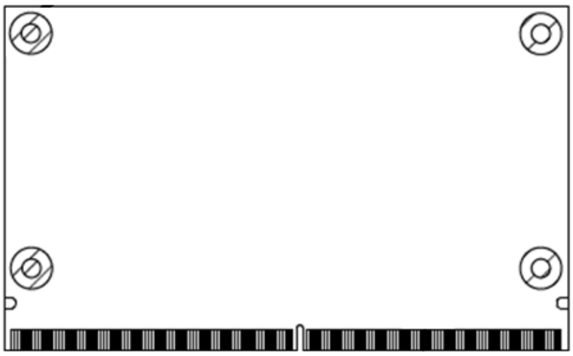
2.1 Board Dimensions and Fixing Holes

Top View

Unit: mm

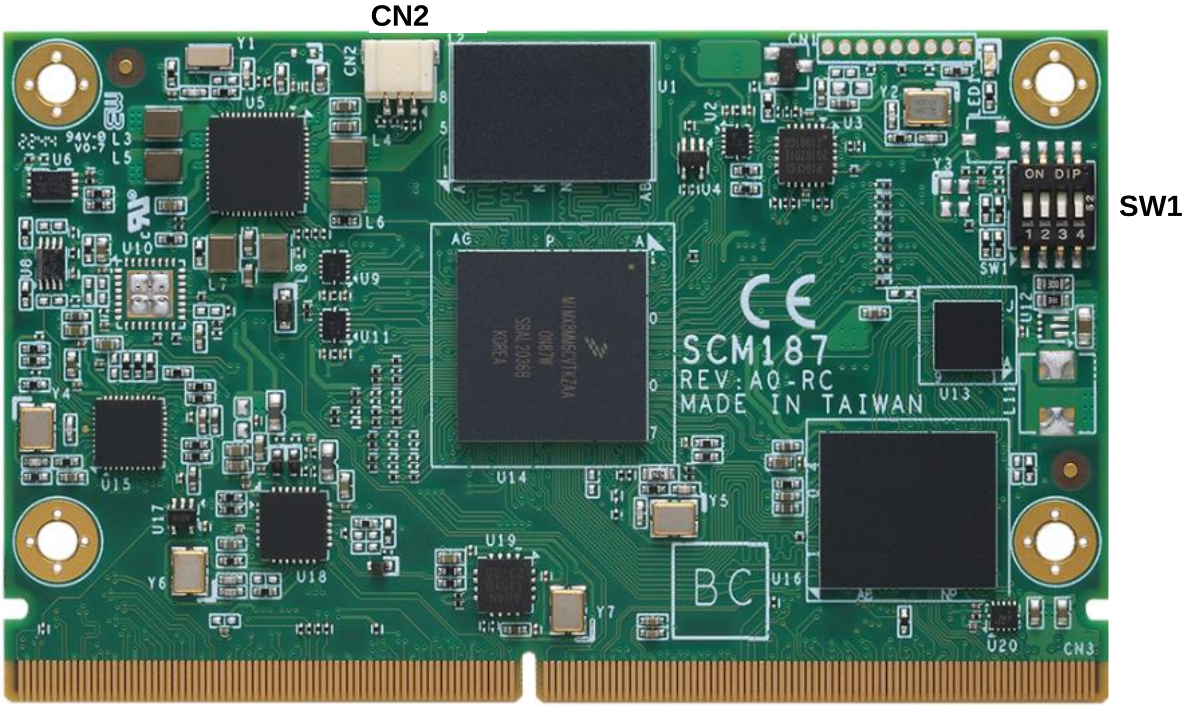


Bottom View

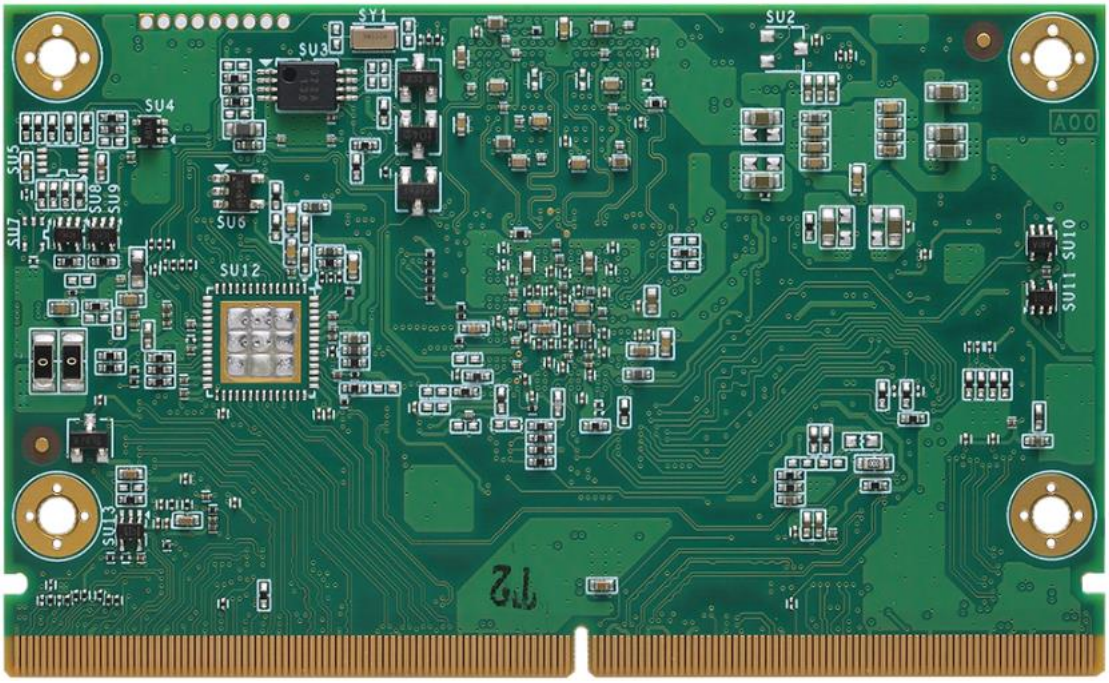


2.2 Board Layout

Top View



Bottom View



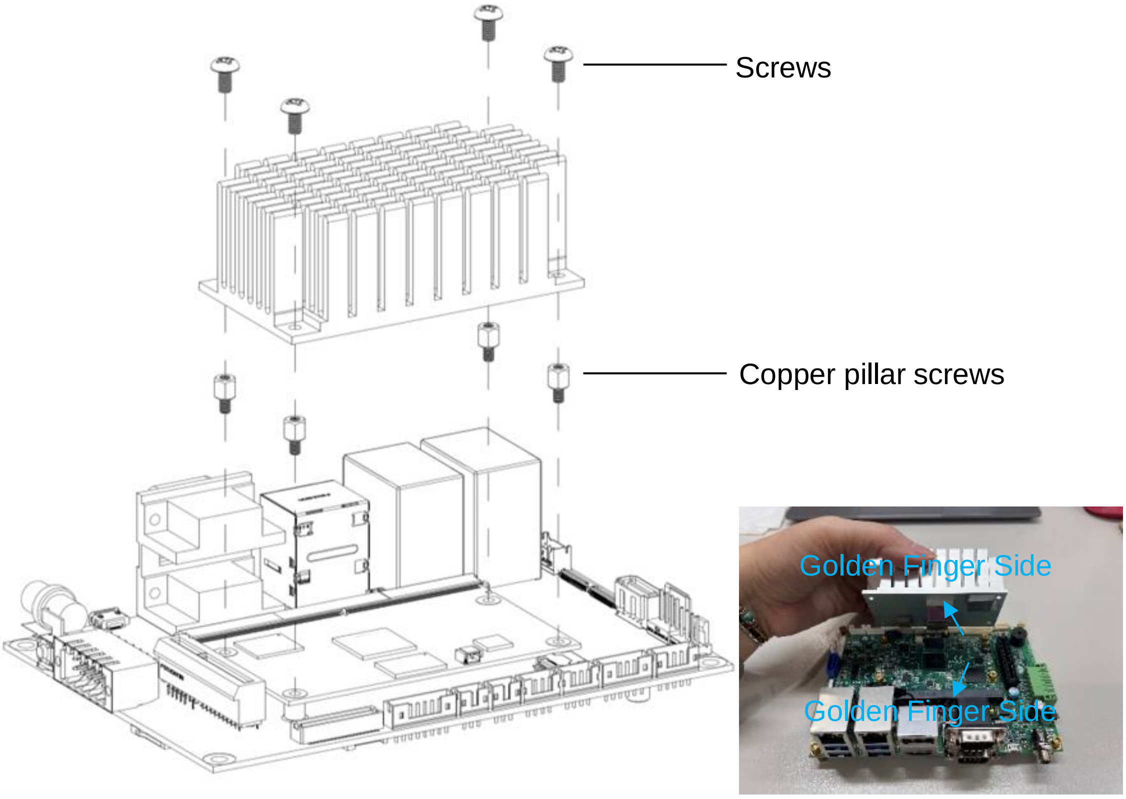
2.3 Installing Heatsink

For optimum performance SCM187 has to be cooled by a passive heatsink optionally available for ordering. For thermal dissipation, the heatsink enable the SCM187 components to dissipate heat efficiently. All heat generating components are thermally conducted to the heatsink in order to avoid hot spots. Below procedures illustrate how to install the heatsink on SCM187.

2.3.1 Heatsink

The heatsink is designed for the SCM187 module. The thermal pad on the heatsink is designed to make contact with the necessary components on the SCM187 module. **When mounting the heatsink you must make sure that the thermal pads on the heatsink make complete contact (no space between thermal pad and component) with the corresponding components on the SCM187 module. This is especially critical for SCM187 module that is with high CPU speed to ensure that the heatsink acts as a proper thermal interface for cooling solutions.**

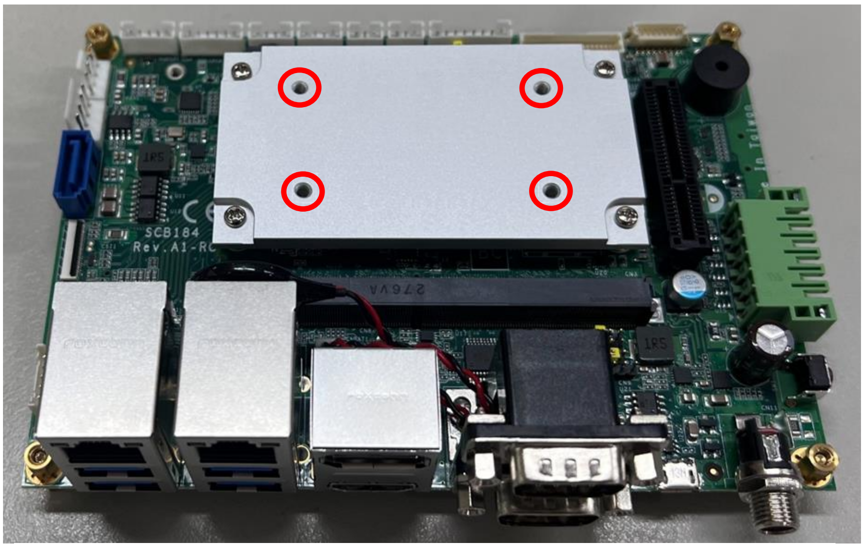
This SMARC module has four assembly holes for installing heatsink. Use the four copper pillar screws and four screws to secure the heatsink to the SCM187. Be careful not to over-tighten the screws.

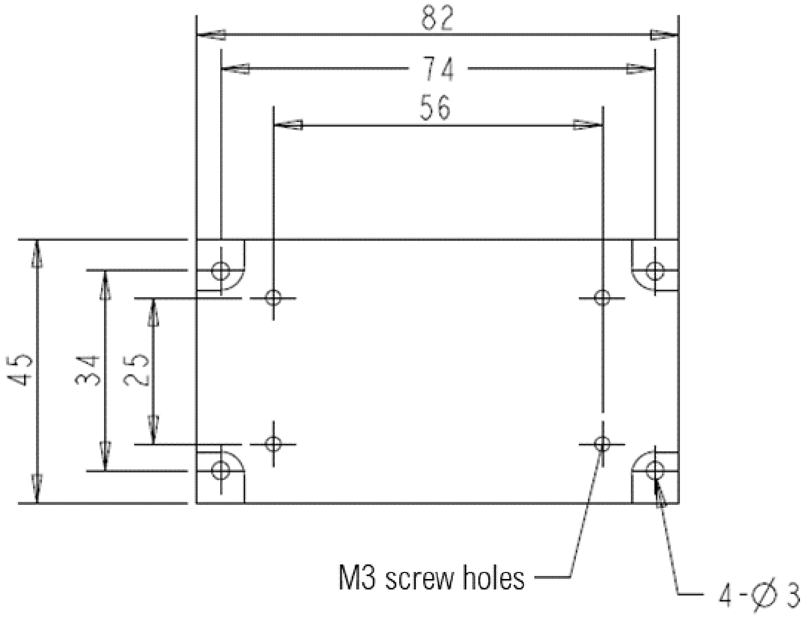


2.3.2 Heat Spreader

The heat spreader is designed for the SCM187 module. The thermal pad on the heat spreader is designed to make contact with the necessary components on the SCM187 module. When mounting the heat spreader, you must make sure that the thermal pads on the heat spreader make complete contact (no space between thermal pad and component) with the corresponding components on the SCM187 module, and then a customized heatsink or thermal chassis must be completely contacted with the heat spreader. This is especially critical for SCM187 module that is with high CPU speed to ensure that the heat spreader acts as a proper thermal interface for cooling solutions.

This SMARC module has four assembly holes for installing heat spreader plate. Use the four copper pillar screws and four screws to secure the heat spreader to the SCM187 module. The heat spreader has another four assembly holes (using M3 screws) can be used for installing a customized heatsink or a thermal chassis as a proper thermal interface for cooling solutions. Be careful not to over-tighten the screws.





2.4 Switch Settings

2.4.1 SWITCH (SW1)

SCM187 switch setting modes as followed.

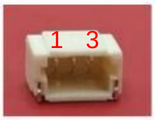
Function	Setting	Switch Setting
Boot by Carrier Board's setting (SCM187 Default)	SW1 OFF SW2 OFF SW3 OFF SW4 OFF	
OS Flash (Flash image)	SW1 ON SW2 OFF SW3 OFF SW4 ON	
Boot from eMMC	SW1 ON SW2 OFF SW3 ON SW4 OFF	
Boot from SD Card	SW1 ON SW2 ON SW3 OFF SW4 OFF	

2.5 Connector

2.5.1 Debug Header (CN2)

This board has a Debug Port (CN2) for debugging.

Pin	Signal
1	UART2_TXD (1.8V TTL Level)
2	UART2_RXD (1.8V TTL Level)
3	GND



To display the console content on the computer, please use TTL 1.8V UART to USB converter to connect between the SMARC module and the computer.

2.6 SMARC Module Top/Bottom Side Pinout Table

P-Pin	Primary (Top) Side	S-Pin	Secondary (Bottom) Side
P1	SMB_ALERT#	S1	I2C3_SCL
P2	GND	S2	I2C3_SDA
P3	CSI1_CK+	S3	GND
P4	CSI1_CK-	S4	NC
P5	NC	S5	NC
P6	NC	S6	NC
P7	CSI1_RX0+	S7	NC
P8	CSI1_RX0-	S8	NC
P9	GND	S9	NC
P10	CSI1_RX1+	S10	GND
P11	CSI1_RX1-	S11	NC
P12	GND	S12	NC
P13	CSI1_RX2+	S13	GND
P14	CSI1_RX2-	S14	NC
P15	GND	S15	NC
P16	CSI1_RX3+	S16	GND
P17	CSI1_RX3-	S17	NC
P18	GND	S18	NC
P19	GBE0_MDI3-	S19	NC
P20	GBE0_MDI3+	S20	NC
P21	GBE0_LINK100#	S21	NC
P22	GBE0_LINK1000#	S22	NC
P23	GBE0_MDI2-	S23	NC
P24	GBE0_MDI2+	S24	NC
P25	GBE0_LINK_ACT#	S25	GND
P26	GBE0_MDI1-	S26	NC
P27	GBE0_MDI1+	S27	NC
P28	NC	S28	NC
P29	GBE0_MDI0-	S29	NC
P30	GBE0_MDI0+	S30	NC
P31	NC	S31	NC
P32	GND	S32	NC
P33	SDIO_WP	S33	NC
P34	SDIO_CMD	S34	GND
P35	SDIO_CD#	S35	NC
P36	SDIO_CK	S36	NC
P37	SDIO_PWR_EN	S37	NC
P38	GND	S38	AUDIO_MCK
P39	SDIO_D0	S39	AUDIO _TXFS
P40	SDIO_D1	S40	AUDIO _TXD

P-Pin	Primary (Top) Side	S-Pin	Secondary (Bottom) Side
P41	SDIO_D2	S41	AUDIO _RXD
P42	SDIO_D3	S42	AUDIO _TXC
P43	SPI0_CS0#	S43	NC
P44	SPI0_CK	S44	NC
P45	SPI0_MISO	S45	NC
P46	SPI0_MOSI	S46	NC
P47	GND	S47	GND
P48	NC	S48	I2C_GP_CK
P49	NC	S49	I2C_GP_DAT
P50	GND	S50	NC
P51	NC	S51	NC
P52	NC	S52	NC
P53	GND	S53	NC
P54	QSPI_CS0#	S54	NC
P55	NC	S55	NC
P56	QSPI_CK	S56	QSPI_IO_2
P57	QSPI_IO_1	S57	QSPI_IO_3
P58	QSPI_IO_0	S58	NC
P59	GND	S59	NC
P60	USB2.0_0D+	S60	NC
P61	USB2.0_0D-	S61	GND
P62	USB0_EN_OC#	S62	NC
P63	USB0_VBUS_DET	S63	NC
P64	USB0_OTG_ID	S64	GND
P65	USB2.0_1D+	S65	NC
P66	USB2.0_1D-	S66	NC
P67	USB1_EN_OC#	S67	GND
P68	GND	S68	USB2.0_3D+
P69	USB2.0_2D+	S69	USB2.0_3D-
P70	USB2.0_2D-	S70	GND
P71	USB2_EN_OC#	S71	NC
P72	NC	S72	NC
P73	NC	S73	GND
P74	USB3_EN_OC#	S74	NC
P75	PCIE_A_RST#	S75	NC
P76	NC	S76	NC
P77	NC	S77	NC
P78	PCIE_A_CKREQ#	S78	NC
P79	GND	S79	NC
P80	NC	S80	GND
P81	NC	S81	NC
P82	GND	S82	NC
P83	PCIE_A_REFCK+	S83	GND
P84	PCIE_A_REFCK-	S84	NC
P85	GND	S85	NC
P86	PCIE_A_RX+	S86	GND
P87	PCIE_A_RX-	S87	NC
P88	GND	S88	NC
P89	PCIE_A_TX+	S89	GND

P-Pin	Primary (Top) Side	S-Pin	Secondary (Bottom) Side
P90	PCIE_A_TX-	S90	NC
P91	GND	S91	NC
P92	HDMI_D2+	S92	GND
P93	HDMI_D2-	S93	NC
P94	GND	S94	NC
P95	HDMI_D1+	S95	NC
P96	HDMI_D1-	S96	NC
P97	GND	S97	NC
P98	HDMI_D0+	S98	NC
P99	HDMI_D0-	S99	NC
P100	GND	S100	NC
P101	HDMI_CK+	S101	GND
P102	HDMI_CK-	S102	NC
P103	GND	S103	NC
P104	HDMI_HPD	S104	NC
P105	HDMI_DDC_CK	S105	NC
P106	HDMI_DDC_DAT	S106	NC
P107	NC	S107	NC
P108	GPIO0	S108	LVDS1_CK+
P109	GPIO1	S109	LVDS1_CK-
P110	GPIO2	S110	GND
P111	GPIO3	S111	LVDS1_0+
P112	GPIO4	S112	LVDS1_0-
P113	GPIO5	S113	NC
P114	GPIO6	S114	LVDS1_1+
P115	GPIO7	S115	LVDS1_1-
P116	GPIO8	S116	NC
P117	GPIO9	S117	LVDS1_2+
P118	GPIO10	S118	LVDS1_2-
P119	GPIO11	S119	GND
P120	GND	S120	LVDS1_3+
P121	I2C_PM_CK	S121	LVDS1_3-
P122	I2C_PM_DAT	S122	NC
P123	BOOT_SEL0#	S123	GPIO13
P124	BOOT_SEL1#	S124	GND
P125	BOOT_SEL2#	S125	LVDS0_0+
P126	Power On reset	S126	LVDS0_0-
P127	RESET_IN#	S127	LCD0_BKLT_EN
P128	POWER_BTN#	S128	LVDS0_1+
P129	SER0_TX	S129	LVDS0_1-
P130	SER0_RX	S130	GND
P131	SER0_RTS#_OUTPUT	S131	LVDS0_2+
P132	SER0_CTS#_INPUT	S132	LVDS0_2-
P133	GND	S133	LCD0_VDD_EN
P134	SER1_TX	S134	LVDS0_CK+
P135	SER1_RX	S135	LVDS0_CK-
P136	SER2_TX	S136	GND

P-Pin	Primary (Top) Side	S-Pin	Secondary (Bottom) Side
P137	SER2_RX	S137	LVDS0_3+
P138	SER2_RTS#_OUTPUT	S138	LVDS0_3-
P139	SER2_CTS#_INPUT	S139	I2C2_SCL
P140	SER3_TX	S140	I2C2_SDA
P141	SER3_RX	S141	LCD0_BKLT_PWM
P142	GND	S142	GPIO12
P143	CAN0_TX	S143	GND
P144	CAN0_RX	S144	NC
P145	CAN1_TX	S145	WATCH_DOG
P146	CAN1_RX	S146	PCIE_WAKE#
P147	+5V	S147	VDD_RTC
P148	+5V	S148	LID
P149	+5V	S149	SLEEP
P150	+5V	S150	VIN_PWR_BAD#
P151	+5V	S151	CHARGING
P152	+5V	S152	CHARGER_PRSNT
P153	+5V	S153	CARRIER_STBY
P154	+5V	S154	CARRIER_PWR_ON
P155	+5V	S155	FORCE_RECOV
P156	+5V	S156	Pull Hi 1.8V
		S157	NC
		S158	GND

This page is intentionally left blank.

Section 3
BSP User Guide

3.1 Abbreviations

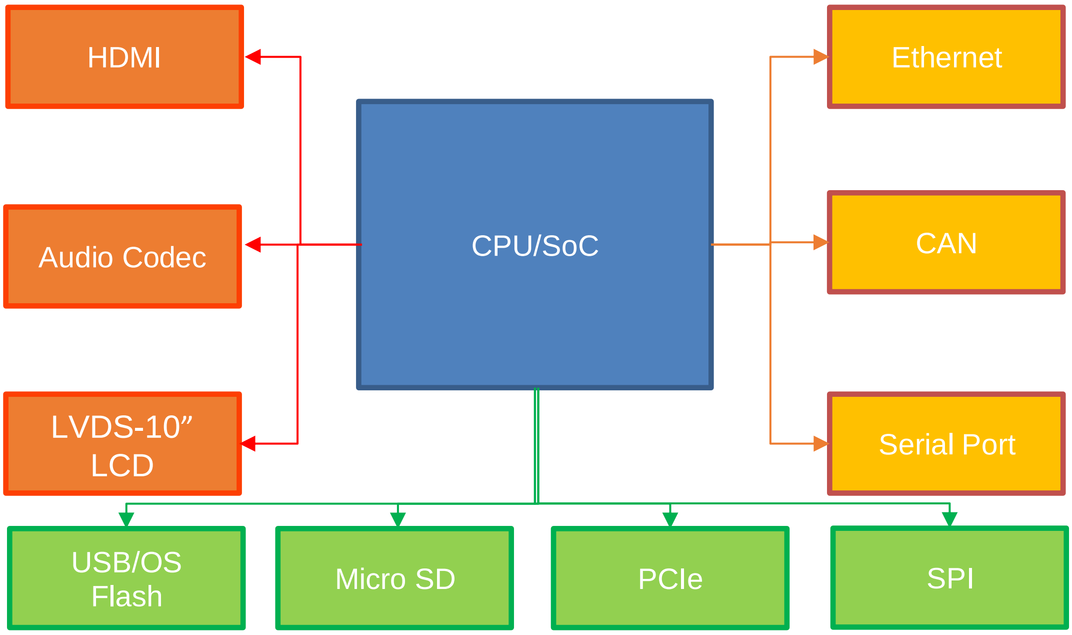
DVI	Digital Visual Interface
DP	DisplayPort
FST	Factory Test Suite
MS	Mechanical System
BSP	Board Support Package

3.2 System Introduction

The SCM187 with SCB184 is designed as an evaluation kit, the SCM187 stands for a SoM, i.e., the system-on-module by SMARC spec., and the SCB184 stands for the part of carrier. We will note the set of evaluation kit for later sections.

3.2.1 System Block Diagram

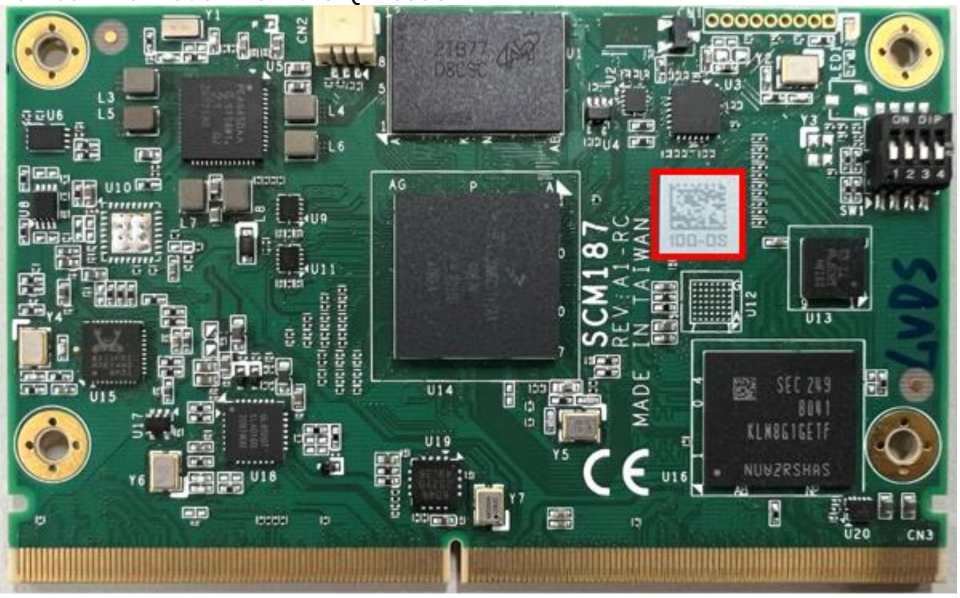
We list the functional components for the evaluation kit as following diagram.



3.2.2 System Overview

The evaluation kit is comprised by several major components, supported function is listed as below for V2.0.0:

- There are two different display interfaces for the evaluation kit, the LVDS or HDMI. You can identify the SMARC module supports LVDS or HDMI through reading the part number information from the QR code.



- Micro SD card is not only for the purpose of storage, but also can be the boot device. (SCM187 switch setting as “Boot from SD Card” mode is required.)
- Single 1 Gbit Ethernet LAN for networking.
- Dual serial ports support RS232 & RS232/485.
- Quad USB support USB 2.0 interface and one USB support USB 2.0 interface with USB connector.
- 10 external GPIO for user usage.
- An audio codec is designed for active speaker and mic in.
- CANbus support standard broadcast communication mechanism protocol.
- PCIe x4 is used for PCIe devices.
- QSPI/ECSPI are supported by external connector.

3.3 Image Programming

This section explains how to flash BSP image to eMMC/SD card.

3.3.1 Accessories Requirement

3.3.1.1 Flash in SD card

To flash the image in SD card for evaluation kit platform, following items are required:

- USB Card reader, or other similar SD card reader.
  - The Transcend is verified for the test, but other functional SD card reader would also work with the same procedure.

- Host PC (Linux/Windows)
- Image for the evaluation kit
  - The format for the evaluation kit image: <evaluation kit_BSP_image>.wic.bz2

3.3.1.2 Flash Image by Linux PC

To flash image by Linux PC, please follow below steps:

- Insert SD card using SD card reader to the PC.
- Execute below command to check which device node that SD card attached. SD card will attach at /dev/sd<x>.
  ```
  $ sudo fdisk -l
  ```
- Change path to image folder. The image will name like: <evaluation kit _image_name>.wic.bz2
- Execute below command to flash image.
  ```
  $ bzcat <evaluation kit_image_name>.wic.bz2 | sudo dd of=/dev/sd<x> bs=1M conv=fsync
  ```

Example:

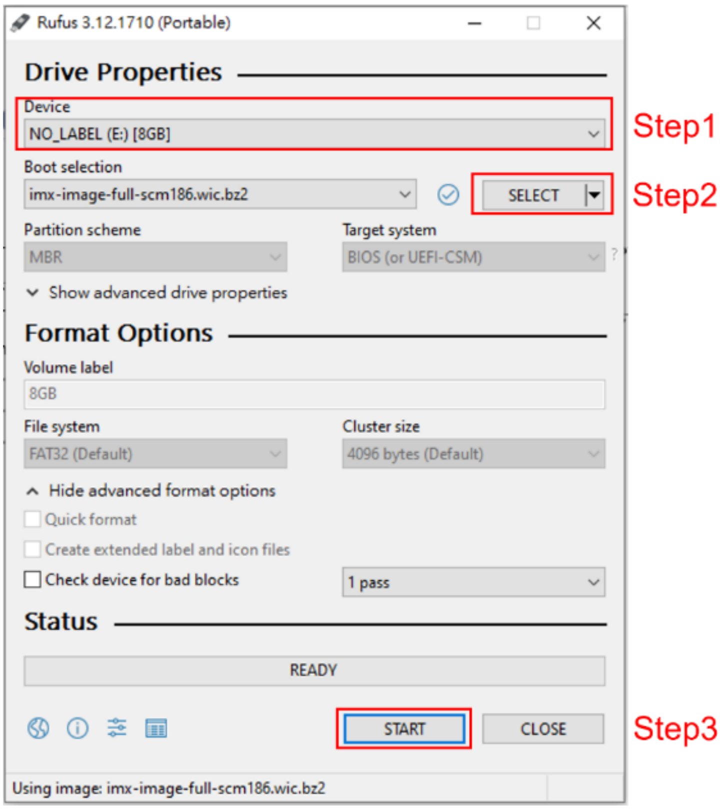
Assume SD card attached at /dev/sdf, image name is imx-image-full-scm187.wic.bz2

```
$ bzcat imx-image-full-scm187.wic.bz2 | sudo dd of=/dev/sdf bs=1M conv=fsync
0+1290634 records in
0+1290634 records out
6493345792 bytes (6.5 GB, 6.0 GiB) copied, 153.449 s, 42.3 MB/s
```

3.3.1.3 Flash Image by Windows PC

To flash image by Windows, please download the “Rufus” by following link.
https://rufus.ie/downloads/
The link for the verified Rufus v3.12 follows:
https://github.com/pbatard/rufus/releases/download/v3.12/rufus-3.12p.exe
The step to flash BSP image to SD card by Windows:

- Insert SD card using SD card reader to the PC.
- Execute flash utility, use Rufus for example
- Step1: check SD card has been recognized.
- Step2: click select button to select image: <evaluation kit_image_name>.wic.bz2
- Step3: click start button, it will format SD card then start flashing image.



Rufus 3.12.1710 (Portable)

Drive Properties

Device: NO_LABEL (E:) [8GB]

Boot selection: imx-image-full-scm186.wic.bz2

Partition scheme: MBR

Target system: BIOS (or UEFI-CSM)

Format Options

Volume label: 8GB

File system: FAT32 (Default)

Cluster size: 4096 bytes (Default)

□ Quick format

□ Create extended label and icon files

□ Check device for bad blocks: 1 pass

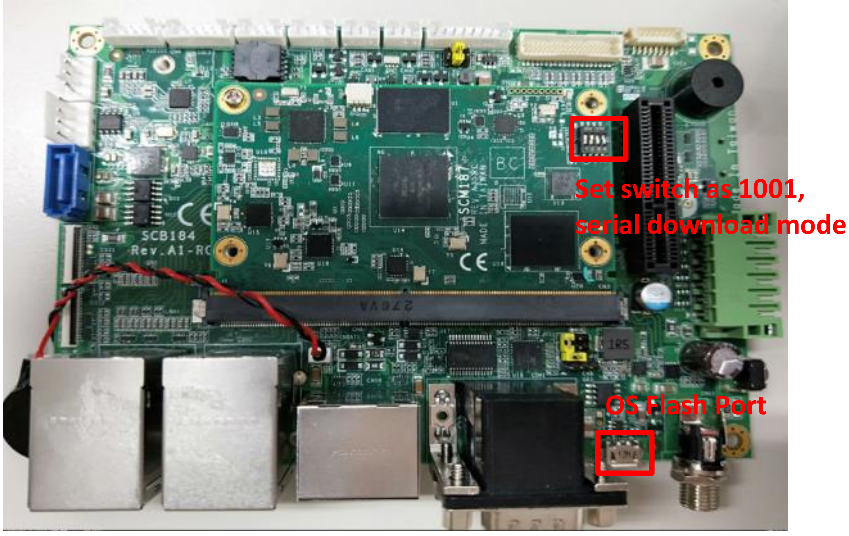
Status: READY

Using image: imx-image-full-scm186.wic.bz2

3.3.1.4 Flash in eMMC

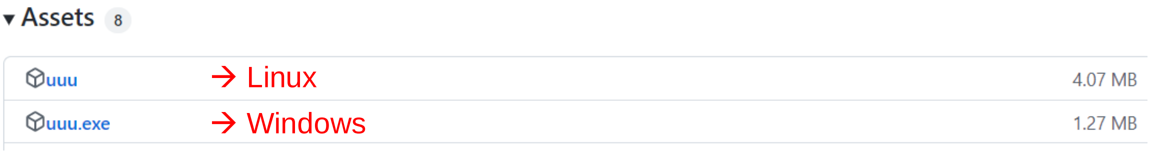
We use UUU tool to flash the image in eMMC for evaluation kit platform, following items are required:

- Micro USB Cable.
- Windows or Linux OS PC (recommend is Windows).
- UUU tool (https://github.com/NXPmicro/mfgtools/releases)
- Set the switch on SCM187 SOM to 1001 as below figure.



3.3.1.5 Flash Image by PC

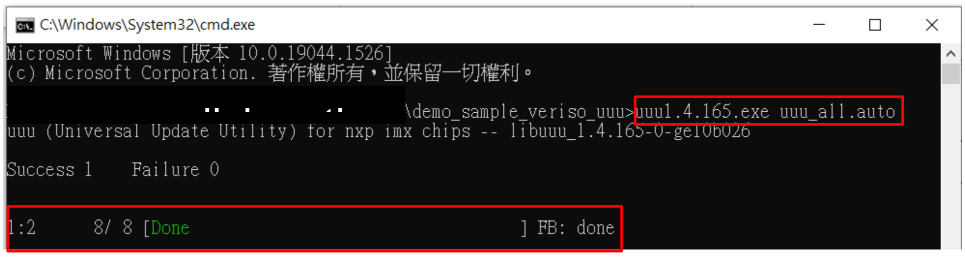
To flash image via Windows/Linux PC, please follow below steps.
The link for the verified UUU tool 1.4.165 follows:
https://github.com/NXPmicro/mfgtools/releases/tag/uuu_1.4.165



▾ Assets 8

uuu	→ Linux	4.07 MB
uuu.exe	→ Windows	1.27 MB

- Connect evaluation kit board and PC via Micro USB cable.
- Open terminal from PC and change directory to image path.
- Execute below command to flash image:
  For Linux PC: *$ sudo ./uuu uuu_all.auto*
  For Windows PC: *$ uuu.exe uuu_all.auto*
- Terminal will show Done after flash successful.



```
Microsoft Windows [版本 10.0.19044.1526]
(c) Microsoft Corporation. 著作權所有，並保留一切權利。

[redacted]\demo_sample_veriso_uuu>uuu1.4.165.exe uuu_all.auto
uuu (Universal Update Utility) for nxp imx chips -- libuuu_1.4.165-0-ge10b026

Success 1    Failure 0

1:2      8/ 8 [Done                                  ] FB: done
```

- Remove Micro USB cable and follow chapter 3.4 for boot steps.

p.s. The image flash package is maintained by Axiomtek, please contact us for more details.

3.4 Boot the Board

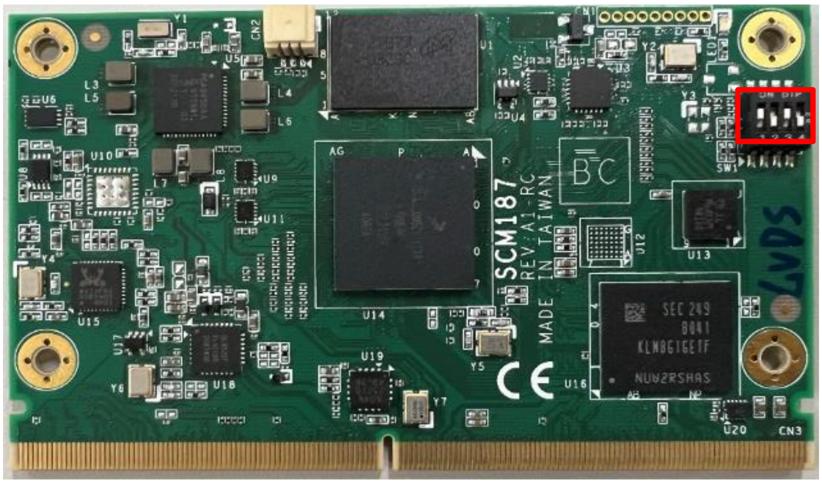
This section explains the step and notices for boot-up evaluation kit platform.

3.4.1 Preparation

Please check the following items before booting the system, some procedures may require to be operated without the case.

Boot from EMMC

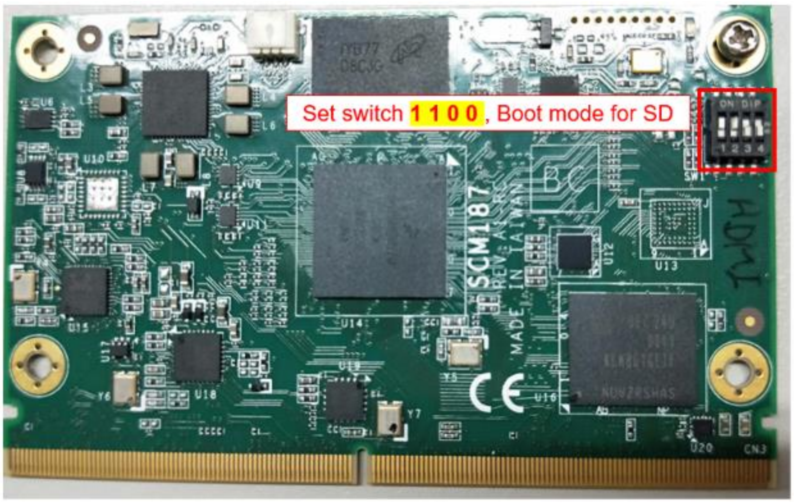
- To make sure the switch on SCM187 SoM is set 0b'1010' as below figure.



Set switch as 1010, boot by eMMC

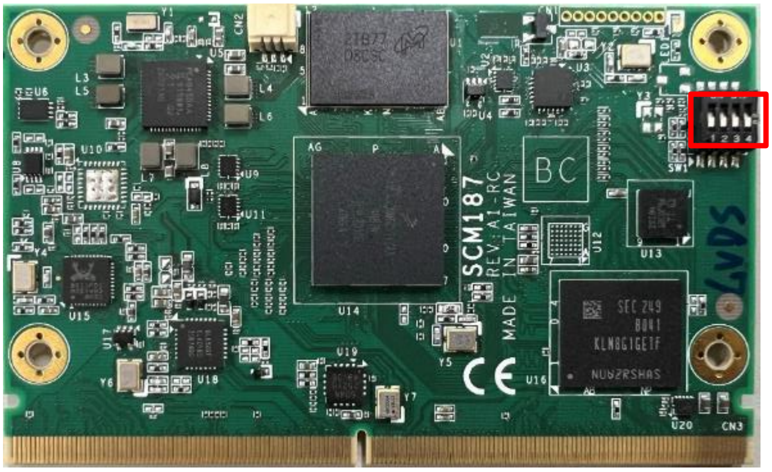
Boot from Micro SD

- To make sure the switch on SCM187 SoM is set 0b'1100' as below figure.
- Micro SD card is installed in the slot, and the Micro SD card is programmed with the procedure of CH.2.



Set switch **1 1 0 0** , Boot mode for SD

Boot From Carrier Board Setting (Default)



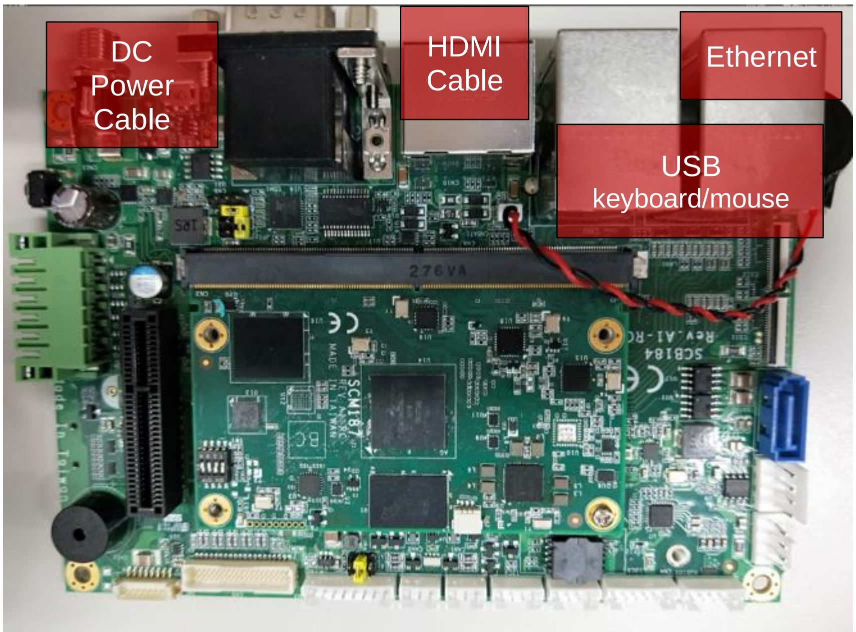
Set switch as 0000, Boot Carrier Board Setting (Default)

- For the platform boot procedure, these accessories are required.
  - HDMI cable
  - USB keyboard and mouse
  - (Optional)Console cable and USB to RS232 bridge cable
  - (Optional)Ethernet

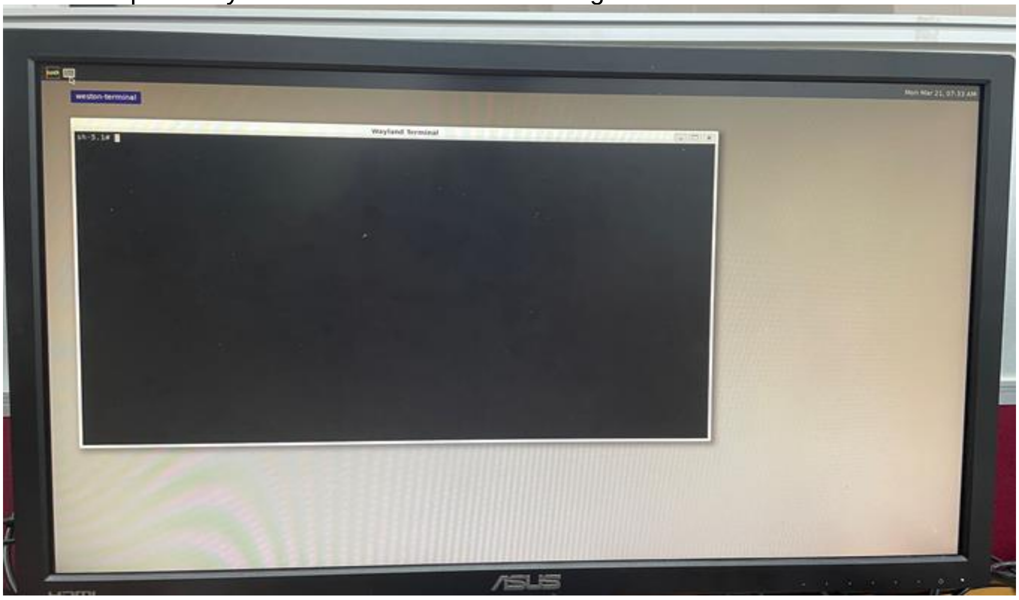
3.4.2 Boot Procedure

Please follow below steps to boot up the evaluation kit platform.

1. Make sure the boot setting for eMMC or SD, then connect the power cord to DC Input to boot the board.



2. The desktop after system booted show as below figure.



3.5 Linux Peripheral Testing

This section of the document explains about how to test the peripherals in Linux OS level for evaluation kit platform.

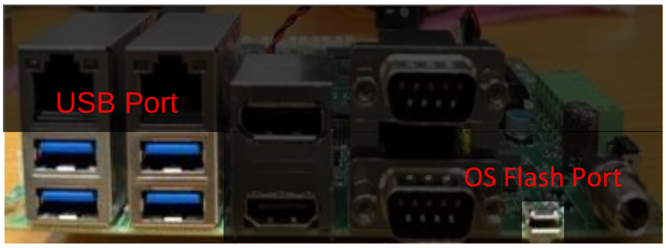
3.5.1 Test Items

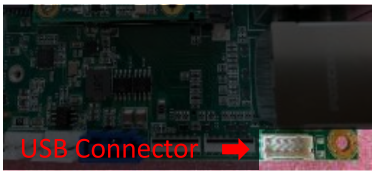
This part of the document includes all peripherals on the evaluation kit platform

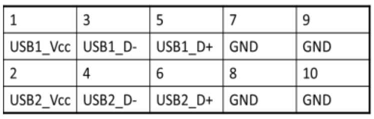
1. USB
2. SD Card
3. HDMI
4. Ethernet
5. Serial Port
6. I2C
7. GPIO
8. Audio Codec
9. CAN
10. PCIE
11. QSPI/ECSPI
12. PWM Fan

3.5.1.1 USB

Evaluation kit board support four USB 2.0 ports, one OS Flash port and single USB connector for one USB 2.0.
The OS Flash port is only used for flash image, this port is reserved.







1	3	5	7	9
USB1_Vcc	USB1_D-	USB1_D+	GND	GND
2	4	6	8	10
USB2_Vcc	USB2_D-	USB2_D+	GND	GND

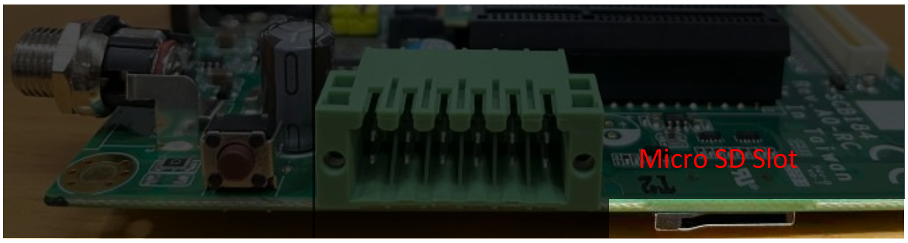
- The Standard USB will mount in below mentioned directories.
  USB - /run/media/sdb1, /run/media/sdb2 … etc.
- To view the contents, please execute below command.
  sh-5.1# ls /<mount_directory>
- To create a directory and remove a directory from the mounted partition, please execute below commands respectively.
  sh-5.1# mkdir <directory_name>
  sh-5.1# rm -rf <target_directory>
- To copy a file to the mounted partition, please execute below command.
  sh-5.1# cp <source_file> <Destination>
- To exit from the mount folder, please execute below command.
  sh-5.1# cd /home/root

If the system cannot mount the USB thumb drive automatically, we have to manually mount the USB thumb drive, please follow the next few steps to mount the USB thumb drive.

- Execute below command to check which device node that USB attached. USB will attach at /dev/sd<x>.
  sh-5.1# fdisk -l
- Create a directory as temp mount node
  sh-5.1# mkdir <temp_directory_name>
- Mount device node on the directory created last step.
  sh-5.1# mount /dev/sd<x> <temp_directory_name>
- View the contents in temp mount node, it will list same as USB device
  sh-5.1# ls <temp_directory_name>
- Unmount the temp mount node to check there are no process on device
  sh-5.1# umount /dev/sd<x>

3.5.1.2 SD Card

The evaluation kit board supports Micro SD card slot for both bootable and storage device, the default mode is for storage.



- The SD will mount in below mentioned directories.
  SD - /run/media/mmcblk1p1 … etc.
- To view the contents, please executing below command.
  sh-5.1# ls /<mount_directory>
- To create a directory and remove a directory from the mounted partition, please execute below commands respectively.
  sh-5.1# mkdir <directory_name>
  sh-5.1# rm -rf <target_directory>
- To copy a file to the mounted partition, please execute below command.
  sh-5.1# cp <source_file> <Destination>
- To exit from the mount folder, please execute below command.
  sh-5.1# cd /home/root

If the system cannot mount the SD automatically, we must manually mount the SD card, please follow the next few steps to mount the SD card.

- Execute below command to check which device node that USB attached. USB will attach at /dev/mmcblk<x>p<y>.
  sh-5.1# fdisk -l
- Create a directory as temp mount node
  sh-5.1# mkdir <temp_directory_name>
- Mount device node on the directory created last step.
  sh-5.1# mount /dev/mmcblk<x>p<y> <temp_directory_name>
- View the contents in temp mount node, it will list same as USB device
  sh-5.1# ls <temp_directory_name>
- Unmount the temp mount node to check there are no process on device
  sh-5.1# umount /dev/mmcblk<x>p<y>

.5.1.3 HDMI

The evaluation kit board provides one HDMI port. This port supports up to 1920x1080@60fps.

- To play a video file, please execute the below command. The test takes .mp4 with audio track, please refer to the log for detail.

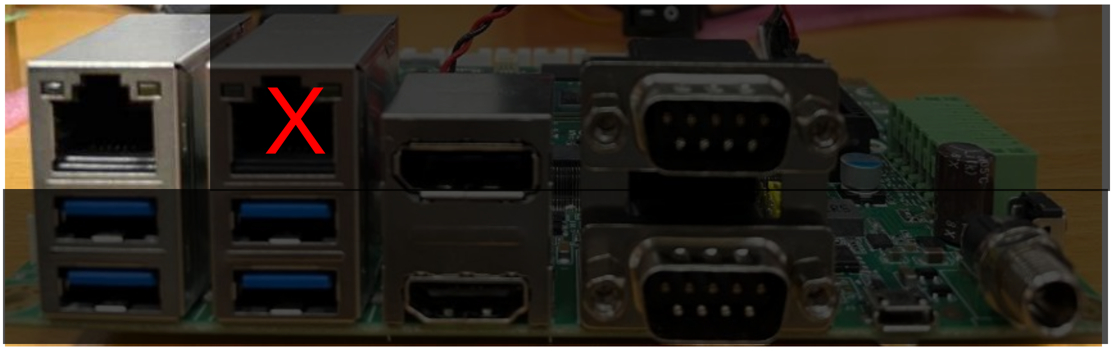
```
sh-5.1# gst-play-1.0 v1080@60.mp4
        Press 'k' to see a list of keyboard shortcuts.
        Now playing /run/media/mmcblk1p1/v1080@60.mp4
        Prerolling...
        ====== AIUR: 4.6.1 build on May 11 2021 03:19:55. ======
        Core: MPEG4PARSER_06.17.18 build on Apr 14 2021 02:39:48
        file: /usr/lib/imx-mm/parser/lib_mp4_parser_arm_elinux.so.3.2
        ------------------------
        Track 00 [video_0] Enabled
        Duration: 0:05:37.704044000
        Language: und
        Mime:
        video/x-h264, parsed=(boolean)true, alignment=(string)au, stream-
        format=(string)avc, width=(int)1920, height=(int)1080,
        framerate=(fraction)90000/1501,
        codec_data=(buffer)0164002affe1001c6764002aacd301e0089f97016a020202
        800001f48000ea60078c189c01000468eebcb0
                ------------------------
                ------------------------
                Track 01 [audio_0] Enabled
                Duration: 0:05:37.757460000
                Language: und
                Mime:
                audio/mpeg, mpegversion=(int)4, channels=(int)2, rate=(int)44100,
                     bitrate=(int)127999, stream-format=(string)raw,
                     codec_data=(buffer)1210000000000000000000000000000
                ------------------------
                ====== BEEP: 4.6.1 build on May 11 2021 03:19:55. ======
                Core: AAC decoder Wrapper build on Jul 28 2020 10:45:34
                file: /usr/lib/imx-mm/audio-codec/wrap/lib_aacd_wrap_arm_elinux.so.3
                CODEC: BLN_MAD-MMCODECS_AACD_ARM_03.09.00_ARMV8
                     build on Sep 20 2017 15:02:50.
                ====== V4L2DEC: 1.18.0 build on Jun 3 2021 08:59:40. ======
                ===!!! Current pulsesink device is alsa_output.platform-sound-
                     wm8524.stereo-fallback !!!===
                Redistribute latency...
                0:00:09.8 / 0:05:37.7
```

***Note*: The HDMI only supports video output, audio is supported by audio codec.**
***Note*: The connected SMARC module must also support the features for the connector to function. Refer to the SMARC module's datasheets for information about the supported features.**

3.5.1.4 Ethernet

Network Devices Test

Although SCB184 has dual Ethernet ports, the evaluation kit supports only one Ethernet port (LAN1), this section explains how to check MAC address and test the Ethernet.



- Connect the Ethernet cable and execute the below command to check MAC address is same with the sticker on the RJ45 connector, and then check IP address has been set by DHCP.

```
sh-5.1# ifconfig eth0
eth0: flags=******<UP,BROADCAST,RUNNING,MULTICAST,DYNAMIC>  mtu 1500
 inet ********  netmask 255.255.254.0  broadcast *********
 inet6 fe80::**************  prefixlen 64  scopeid 0x20<link>
 ether ************  txqueuelen 1000  (Ethernet)
 RX packets 45182  bytes 3442113 (3.2 MiB)
 RX errors 262  dropped 0  overruns 0  frame 262
 TX packets 8842  bytes 120569043 (114.9 MiB)
 TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

- To test the connectivity, please execute the below command.

```
sh-5.1# ping <any_ip_addr>
PING ********** (**********) 56(84) bytes of data.
64 bytes from **********: icmp_seq=1 ttl=64 time=**** ms
64 bytes from **********: icmp_seq=2 ttl=64 time=**** ms
```

Network Speed and Duplex Settings

The “ethtool” utility configures the Ethernet settings such as speed, auto-negotiation on many network devices, especially Ethernet devices. This part explains how to set the Ethernet speed to 10/100/1000 Mbps or half/full duplex in evaluation kit board.

- Execute the below command to set the desired speed/duplex in the ethernet.

```
sh-5.1# ethtool -s eth0 speed [SPEED] duplex [DUPLEX]
```

Example:

```
sh-5.1# ethtool -s eth0 speed 100 duplex half
sh-5.1# ethtool -s eth0 speed 10 duplex full
```

- Execute the below command to check the current Ethernet settings.

```
sh-5.1# ethtool eth0
```

<u>Note</u>: The Max. Speed 1000Mbps is supported if only if the evaluation kit board is connected in the 1000Mbps network. features.

3.5.1.5 Serial Port

The evaluation kit supports the serial port with different mode & baud rates. This section explains how to test the switch mode and test.

Configuration for Serial Ports
Evaluation kit supports below several serial ports, these ports provide varies protocols, please refer to following list for the supported list.

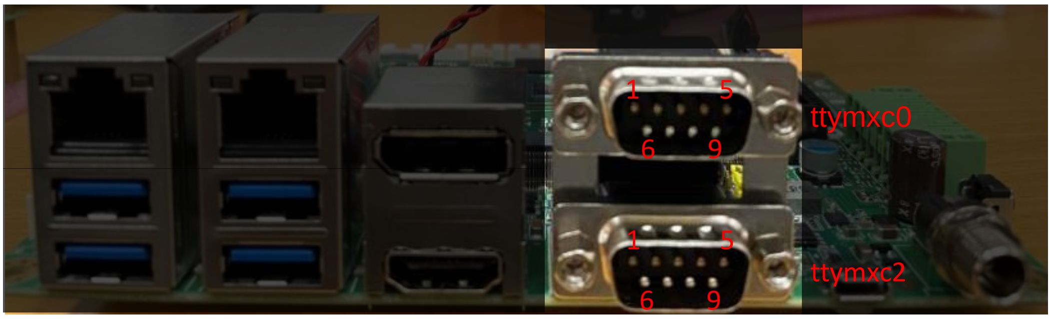
- /dev/ttymxc0
  - RS232
- /dev/ttymxc2
  - RS232
  - RS485

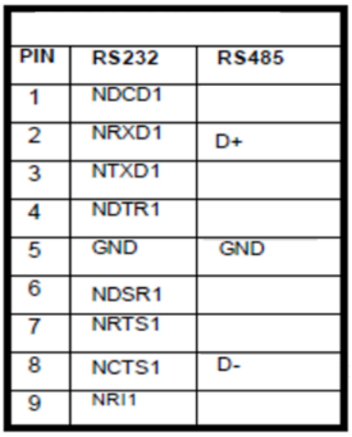
To exercise the serial port for /dev/ttymxc2, there are several modes must be configured by the switch, the following command demonstrates the test though all protocols, the default mode is RS232.

- Enable serial port switch
  ```
  sh-5.1# cd /sys/class/gpio
  sh-5.1# echo 496 > export
  sh-5.1# echo 497 > export
  sh-5.1# echo out > gpio496/direction
  sh-5.1# echo out > gpio497/direction
  ```
- Switch serial port to RS232
  ```
  sh-5.1# echo 1 > gpio496/value
  sh-5.1# echo 0 > gpio497/value
  ```
- Switch serial port to RS485
  ```
  sh-5.1# echo 0 > gpio496/value
  sh-5.1# echo 1 > gpio497/value
  ```

Serial Port Test

The below table and figure present that serial port pin define on the evaluation kit board





PIN	RS232	RS485
1	NDCD1	
2	NRXD1	D+
3	NTXD1	
4	NDTR1	
5	GND	GND
6	NDSR1	
7	NRTS1	
8	NCTS1	D-
9	NRI1	

- Switch the mode and connect the correct pin with serial port of host PC or loopback by using cable.
- To set evaluation kit serial port baud rate, please execute below command.

  ```
  sh-5.1# stty -F /dev/ttymxc<x> <Baud rate>
  ```

  Example:

  ```
  sh-5.1# stty -F /dev/ttymxc0 115200
  ```

- Open another UART console in host PC and set the serial port settings as mentioned below.

  Bits per second: 115200 bps
  Data bits: 8
  Parity: none
  Stop bits: 1
  Flow control: none

- To transmit data through the serial port, please execute the below command.

  ```
  sh-5.1# echo "uart_test_message" > /dev/ttymxc<x>
  ```

- To receive the data by serial port, please execute the below command.

  ```
  sh-5.1# cat /dev/ ttymxc<x>
  ```

- To set the Baud rate to a different value, please execute the below command.

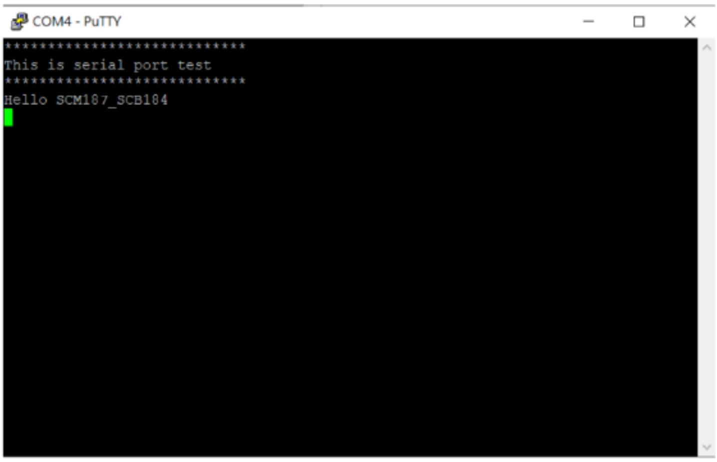
  ```
  sh-5.1# stty -F /dev/ttymxc<x><Baud rate>
  ```

  Example:

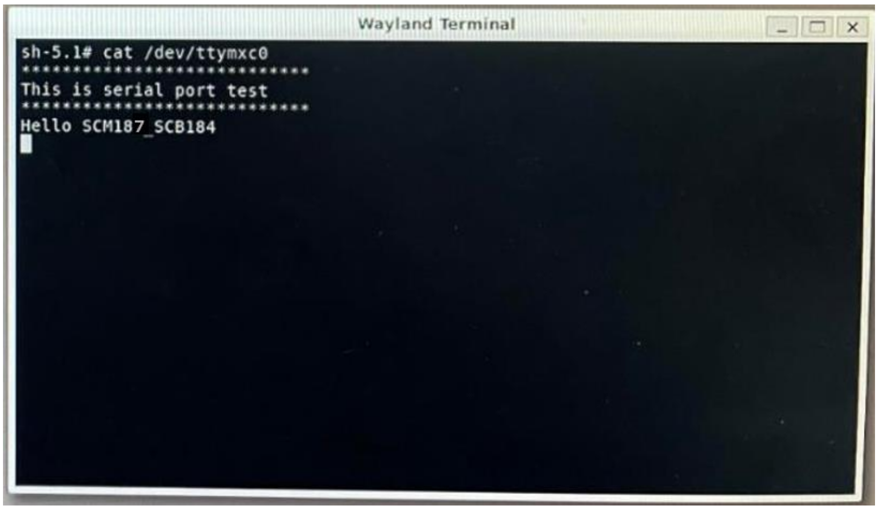
  ```
  sh-5.1# stty -F /dev/ttymxc0 115200
  ```

***Note*: The tested Baud rates are given below: 300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200**

Example:
Host PC sent message, evaluation kit received

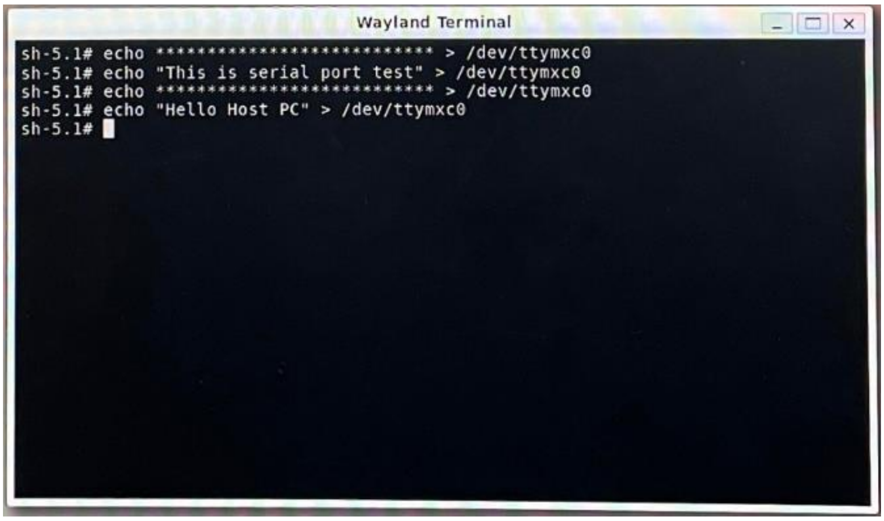


```
****************************
This is serial port test
****************************
Hello SCM187_SCB184
```

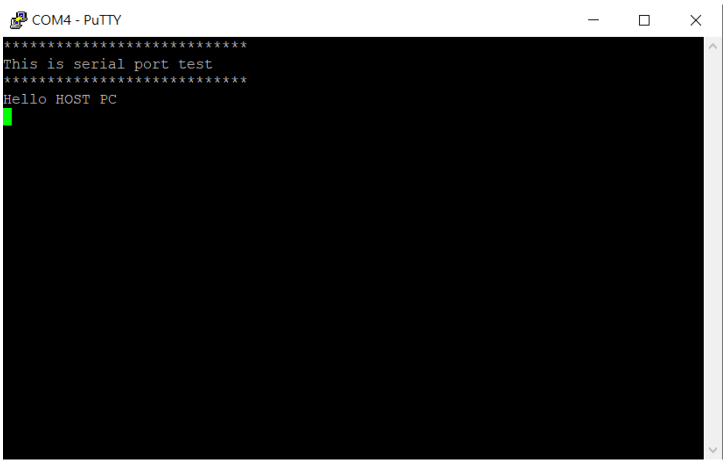


```
sh-5.1# cat /dev/ttymxc0
****************************
This is serial port test
****************************
Hello SCM187_SCB184
```

Evaluation kit sends message, Host PC received.



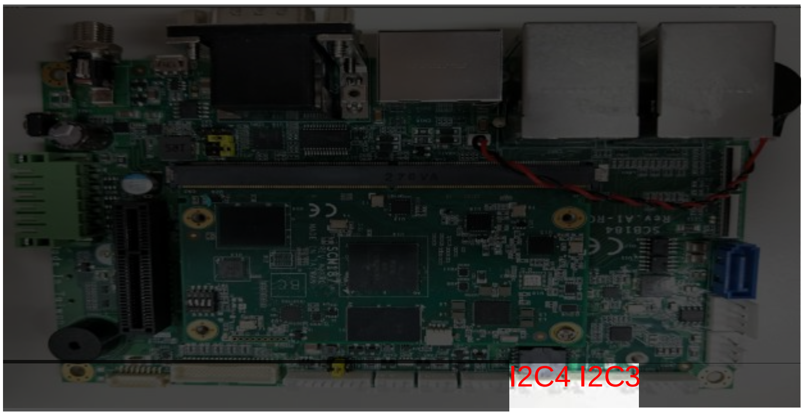
```
sh-5.1# echo **************************** > /dev/ttymxc0
sh-5.1# echo "This is serial port test" > /dev/ttymxc0
sh-5.1# echo **************************** > /dev/ttymxc0
sh-5.1# echo "Hello Host PC" > /dev/ttymxc0
sh-5.1# 
```



```
****************************
This is serial port test
****************************
Hello HOST PC
```

3.5.1.6 I2C

The evaluation kit supports 2*i2c connector for using external i2c devices. This section explains how to control external i2c devices by i2c-tool.



	4	3	2	1
I2C Connector	GND	I2C_DATA	I2C_CLK	Vcc_3.3V

SCB184 Connector	I2C Bus Control	Example
I2C3	0	i2cdetect –y 0
I2C4	2	i2cdetect –y 2

There are i2c-tools usage below:

- i2cdetect – This function can detect exist i2c bus and list all devices in selected i2c bus.

```
Usage: i2cdetect [-y] [-a] [-q|-r] I2CBUS [FIRST LAST]
       i2cdetect -F I2CBUS
       i2cdetect -l
  I2CBUS is an integer or an I2C bus name
  If provided, FIRST and LAST limit the probing range.
```

- i2cdump – This function can dump all data value in selected i2c device address.

```
Usage: i2cdump [-f] [-y] [-r first-last] [-a] I2CBUS ADDRESS [MODE [BANK
[BANKREG]]]
  I2CBUS is an integer or an I2C bus name
  ADDRESS is an integer (0x08 - 0x77, or 0x00 - 0x7f if -a is given)
  MODE is one of:
    b (byte, default)
    w (word)
    W (word on even register addresses)
    s (SMBus block)
    i (I2C block)
    c (consecutive byte)
    Append p for SMBus PEC
```

- i2cset – This function can set value to data address in i2c devices.

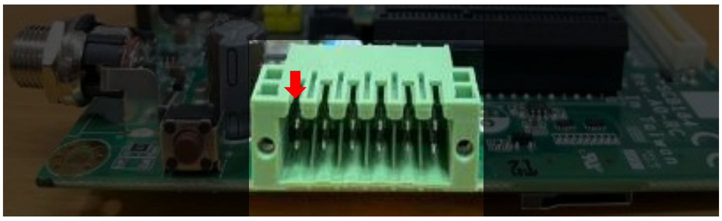
```
Usage: i2cset [-f] [-y] [-m MASK] [-r] [-a] I2CBUS CHIP-ADDRESS DATA-ADDRESS
[VALUE] ... [MODE]
  I2CBUS is an integer or an I2C bus name
  ADDRESS is an integer (0x08 - 0x77, or 0x00 - 0x7f if -a is given)
  MODE is one of:
    c (byte, no value)
    b (byte data, default)
    w (word data)
    i (I2C block data)
    s (SMBus block data)
    Append p for SMBus PEC
```

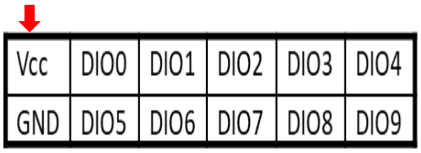
- i2cget – This function can get more specific value for each data address in i2c devices.

```
Usage: i2cget [-f] [-y] [-a] I2CBUS CHIP-ADDRESS [DATA-ADDRESS [MODE]]
  I2CBUS is an integer or an I2C bus name
  ADDRESS is an integer (0x08 - 0x77, or 0x00 - 0x7f if -a is given)
  MODE is one of:
    b (read byte data, default)
    w (read word data)
    c (write byte/read byte)
    Append p for SMBus PEC
```

3.5.1.7 GPIO

The evaluation kit supports GPIO connector for 10 external GPIO control. This section explains how to control GPIO.





Vcc	DIO0	DIO1	DIO2	DIO3	DIO4
GND	DIO5	DIO6	DIO7	DIO8	DIO9

To exercise the external GPIO, it needs to export each GPIO by their GPIO chip number, there are, the following table and command demonstrates the GPIO chip number and several steps to configure GPIO.

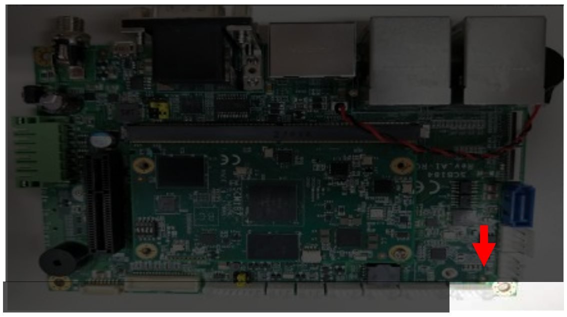
	Vcc	DIO0	DIO1	DIO2	DIO3	DIO4
GPIO number	N/A	43	33	6	35	36
	GND	DIO5	DIO6	DIO7	DIO8	DIO9
GPIO number	N/A	133	40	41	42	34

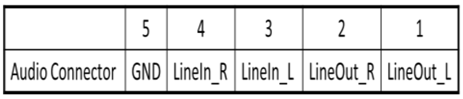
- To export GPIO, please execute the below command.
  ```
  sh-5.1# cd /sys/class/gpio
  sh-5.1# echo “GPIO number” > export
  ```
- To set GPIO direction as input, please execute the below command.
  ```
  sh-5.1# echo in > gpio“GPIO number”/direction
  ```
- To set GPIO direction as output, please execute the below command.
  ```
  sh-5.1# echo out > gpio“GPIO number”/direction
  ```
- To get GPIO status, please execute the below command.
  ```
  sh-5.1# cat gpio“GPIO number”/direction
  ```
- To get GPIO value, please execute the below command.
  ```
  sh-5.1# cat gpio“GPIO number”/value
  ```
- To set GPIO value, please execute the below command. (GPIO status out only)
  - GPIO set low
    ```
    sh-5.1# echo 0 > gpio“GPIO number”/value
    ```
  - GPIO set high
    ```
    sh-5.1# echo 1 > gpio“GPIO number”/value
    ```

p.s. The default status of all GPIO is high.

3.5.1.8 Audio Codec

The evaluation kit supports the SGTL5000 audio devices. This section explains how to test SGTL5000 audio codecs.





	5	4	3	2	1
Audio Connector	GND	LineIn_R	LineIn_L	LineOut_R	LineOut_L

Please following the pin definition to connect audio signal.

- To list the available audio interfaces, please execute the below command.

```
sh-5.1# aplay -l
```

- The registered audio interfaces will be displayed on terminal as shown below,

```
**** List of PLAYBACK Hardware Devices ****
card 0: sgtl5000audio [sgtl5000-audio], device 0: HiFi sgtl5000-0 [HiFi sgtl5000-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: sgtl5000audio [sgtl5000-audio], device 1: HiFi-ASRC-FE (*) []
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

The evaluation kit BSP supports the Gstreamer framework. This part explains how to play a multimedia file using Gstreamer framework.

- To list the available plug-ins, please execute the below command.

```
sh-5.1# gst-inspect-1.0
```

- To play a video file using gst-play utility, please execute the below command.

```
sh-5.1# gst-play-1.0 /<path to file>/<file_name>
```

- To play an audio file using a particular audio device, please execute the below command.

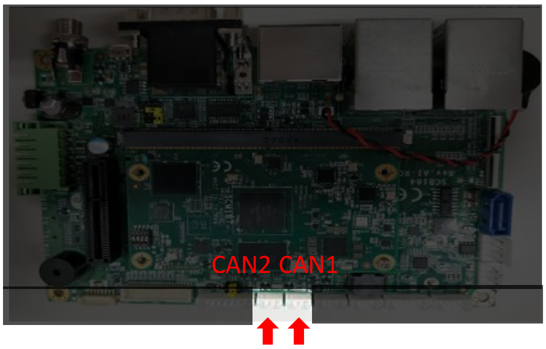
```
sh-5.1# gst-launch-1.0 filesrc location=<audio file> ! decodebin ! audioconvert ! alsasink
device=plughw:<card>,<subdevice>
```

- To record an audio file using a particular audio device, please execute the below command.

```
sh-5.1# gst-launch-1.0 -v alsasrc device= plughw:<card>,<subdevice> ! 'audio/x-raw,
rate=48000, channels=2' ! lamemp3enc ! filesink location=<file_name.mp3>
```

3.5.1.9 CAN

The evaluation kit supports the 2 CAN connectors. This section explains how to execute and test CAN modules.



	3	2	1
CAN Connector	GND	CAN_H	CAN_L

Note:

i. *can0 is mapping to CAN1, can1 is mapping to CAN2 on board.*

ii. *Executing CAN receive command in receiver side before transmitting the data.*

- Connect *the evaluation kit platform CAN port to another evaluation kit CAN port (*it may be either CAN1 or CAN2 port).
- Set the bitrate and enable the CAN device, execute the below command.

```
sh-5.1# ip link set <CAN_DEVICE> up type can bitrate <bitrate>
The bitrate supports up to 1000000.
```

- To ensure the CAN network status, please execute the below command.

```
sh-5.1# ifconfig
can0: flags=193<UP,RUNNING,NOARP>  mtu 16
unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 10  (UNSPEC)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 38
can1: flags=193<UP,RUNNING,NOARP>  mtu 16
unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 10  (UNSPEC)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
                              device interrupt 39
```

- To receive the data and display in console, execute the below command.

```
sh-5.1# candump <candev_no> &
candev_no 123 [4] AA BB CC DD
```

- To transmit the data, execute the below command.

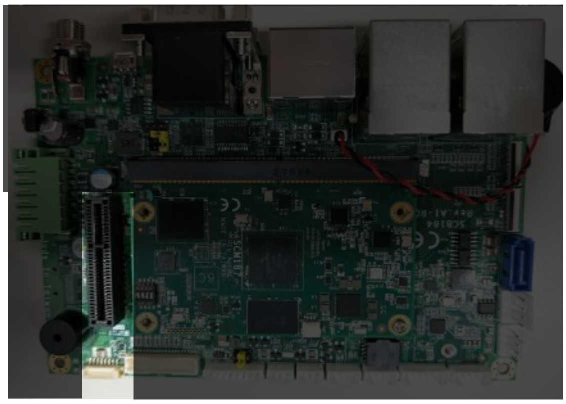
```
sh-5.1# cansend <candev_no> 123#AABBCCDD
```

- To disable the CAN modules, execute the below command.

```
Sh-5.1# ifconfig <candev_no> down
```

3.5.1.10 PCIe

The evaluation kit supports PCIe switch and End point devices. This section explains how to list the connected PCIe switches and End point devices.

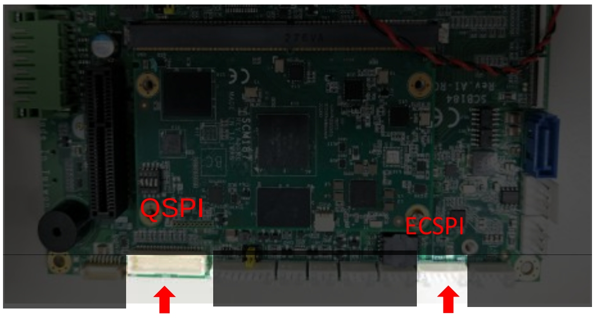


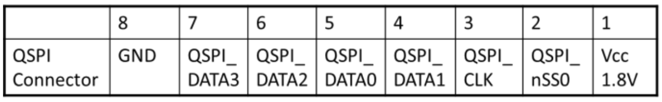
- Connect the PCIe device in evaluation kit platform before powering ON.
- To list out the PCIe device, connected with board, please execute the below command(using pcie test card to example):

```
sh-5.1# lspci
00:00.0 PCI bridge: Synopsys, Inc. DWC_usb3 / PCIe bridge (rev 01)
01:00.0 USB controller: ASMedia Technology Inc. ASM1142 USB 3.1 Host Controller
```

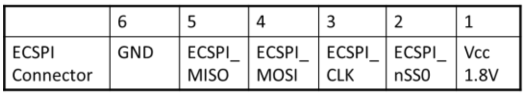
3.5.1.11 QSPI/ECSPI

The evaluation kit supports 1 QSPI and 1 ECSPI external connector. User can connect spi device and porting customize configuration.





	8	7	6	5	4	3	2	1
QSPI Connector	GND	QSPI_ DATA3	QSPI_ DATA2	QSPI_ DATA0	QSPI_ DATA1	QSPI_ CLK	QSPI_ nSS0	Vcc 1.8V



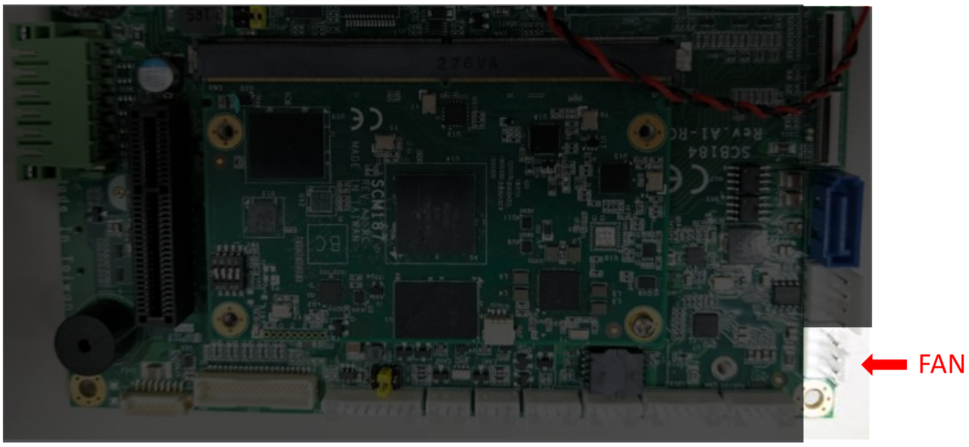
	6	5	4	3	2	1
ECSPI Connector	GND	ECSPI_ MISO	ECSPI_ MOSI	ECSPI_ CLK	ECSPI_ nSS0	Vcc 1.8V

Please follow below commands to execute QSPI/ECSPI device (using SPI NOR flash for example)

- To display the SPI NOR flash information, execute the below command.
  sh-5.1# cat /proc/mtd
- To mount the SPI NOR flash partitions, execute the below command.
  sh-5.1# flash_erase -jq /dev/mtd0 0 0
  sh-5.1# mkdir /<mount_directory>
  sh-5.1# mount -t jffs2 /dev/mtdblock0 /<mount_directory>
- To view the files and folders in mounted partitions, execute the below command.
  sh-5.1# cd /<mount_directory>
  sh-5.1# ls
- To create a directory and remove a directory in mounted partition, execute the below commands respectively.
  sh-5.1# mkdir <directory_name>
  sh-5.1# rm -rf <target_directory>
- To copy a file to the mounted partition, execute the below command.
  sh-5.1# cp <soruce_file> <Destination>
- To exit from the mount partitions and to unmount, execute the below command.
  sh-5.1# cd /root
  sh-5.1# umount <Mount Directory>

3.5.1.12 PWN FAN

The evaluation kit supports pwm fan connector using typical 4 pin fan. This section explains how to control pwm fan.



Please follow below commands to execute pwm fan.

- To change directory to pwm chip path, execute the below command.
  ```
  sh-5.1# cd /sys/class/pwm/pwmchip1
  ```
- To export and enable pwm control configuration, execute the below command.
  ```
  sh-5.1# echo 0 > export
  sh-5.1# echo 1 > pwm0/enable
  ```
- To set pwm parameters period and duty cycle, execute the below command.
  ```
  sh-5.1# echo <period> > pwm0/period
  sh-5.1# echo <duty_cycle> > pwm0/duty_cycle
  ```
- To unexport and disable pwm control, execute the below command.
  ```
  sh-5.1# echo 0 > pwm0/enable
  sh-5.1# echo 0 > unexport
  ```