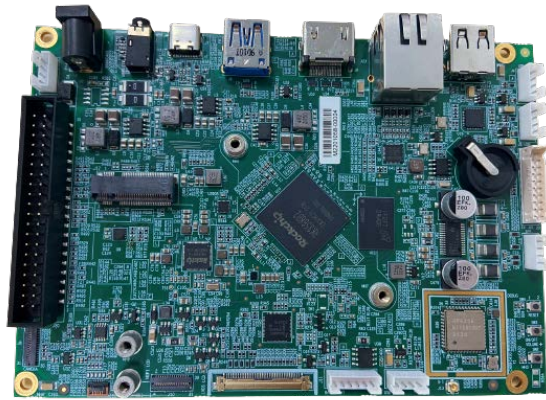


VT-SBC-3568 Single Board Computer



User Manual

Version: 1.4

© Vantron Technology, Inc. All rights reserved.

Revision History

No.	Version	Description	Date
1	V1.0	First release	Jan. 14, 2022
2	V1.1	Updated pinout description of GPIO	Jun. 16, 2022
3	V1.2	Added a chapter for Yocto project development	Mar. 3, 2023
4	V1.3	Added Android system description and updated the typesetting	Jun. 2, 2023
5	V1.4	Updated the debugging instructions on UART & GPIO in Debian system	Nov. 28, 2023

Table of Contents

Foreword	1
CHAPTER 1 INTRODUCTION	5
1.1 Product Overview	6
1.2 Terminology	6
1.3 Block Diagram	7
1.4 Specifications	8
1.5 Operating system	9
1.6 Mechanical Dimensions	9
1.7 Power Supply and Consumption	9
1.8 Environmental Specifications	9
CHAPTER 2 CONNECTOR DESCRIPTION	10
2.1 Product Layout	11
2.2 Memory and Storage	11
2.2.1 LPDDR4 RAM	11
2.2.2 eMMC Flash	11
2.2.3 EEPROM	11
2.3 Identification of Pin 1	12
2.4 Connectors and Jumpers	12
2.4.1 Power input (1)	12
2.4.2 J23 Ethernet port (2)	13
2.4.3 J13 Wi-Fi and Bluetooth (3)	13
2.4.4 J9 HDMI (4)	13
2.4.5 J11 LVDS (5)	13
2.4.6 J12 Backlight connector (6)	15
2.4.7 J10 MIPI DSI (7)	15
2.4.8 J8 MIPI CSI (8)	16
2.4.9 J22 I ² C (9)	18
2.4.10 U40 USB 2.0 Type-A (10)	18
2.4.11 J3/J4/J5 USB 2.0 (11)	19
2.4.12 J6 USB 3.0 (12)	19
2.4.13 J7 USB 2.0 Type-C (13)	19
2.4.14 J21 GPIO (14)	20
2.4.15 J20 40-pin header (15)	21
2.4.16 J17 M.2 Key B slot (16)	22
2.4.17 J16 Audio jack (17)	23
2.4.18 J15 Speaker connector (18)	24
2.4.19 Buttons (19-21)	24
2.4.20 J18 Micro SIM (22)	24
CHAPTER 3 ANDROID SYSTEM MANUAL	25
3.1 Enable Developer Options	26
3.2 ADB Setup on the Windows Host	26
3.3 App Installation via ADB Commands	28
3.4 Serial port	29
3.5 CAN	29
3.6 Connecting to Internet	30

CHAPTER 4	DEBIAN SYSTEM MANUAL	31
4.1	Interface Definition	32
4.1.1	Serial port	32
4.1.2	USB interface	33
4.1.3	GPIO	33
4.2	Ethernet	34
4.3	Wi-Fi	35
4.4	Cellular Network	37
4.5	RTC	40
4.6	Watchdog Timer	40
4.7	Firmware Upgrade in Windows Environment	41
4.8	Firmware Upgrade in Ubuntu Environment	43
CHAPTER 5	YOCTO PROJECT DEVELOPMENT	44
5.1	Build Environment	45
5.1.1	Prerequisites	45
5.1.2	Dependency Installation	45
5.1.3	SDK installation	45
5.1.4	Test and verification	46
5.1.5	Compiling “Hello World”	47
5.2	Image flashing	48
5.2.1	Prerequisites	48
5.2.2	Hardware connection	48
5.2.3	Preparation of the firmware package	49
5.2.4	Image flashing	49
CHAPTER 6	DISPOSAL AND PRODUCT WARRANTY	51
6.1	Disposal	52
6.2	Warranty	53

Foreword

Thank you for purchasing VT-SBC-3568 single board computer (“the Board” or “the Product”). This manual intends to provide guidance and assistance necessary on setting up, operating or maintaining the Product. Please read this manual and make sure you understand the functionality of the Product before putting it into use.

Intended Users

This manual is intended for:

- Embedded software developer
- Custom development software engineer
- Other technically qualified personnel

Copyright

Vantron Technology, Inc. (“Vantron”) reserves all rights of this manual, including the right to change the content, form, product features, and specifications contained herein at any time without prior notice. An up-to-date version of this manual is available at www.vantrontech.com.

The trademarks in this manual, registered or not, are properties of their respective owners. Under no circumstances shall any part of this user manual be copied, reproduced, translated, or sold. This manual is not intended to be altered or used for other purposes unless otherwise permitted in writing by Vantron. Vantron reserves the right of all publicly released copies of this manual.

Disclaimer

While all information contained herein has been carefully checked to assure its accuracy in technical details and typography, Vantron does not assume any responsibility resulting from any error or features of this manual, nor from improper uses of this manual or the software.

It is our practice to change part numbers when published ratings or features are changed, or when significant construction changes are made. However, some specifications of the Product may be changed without notice.

Technical Support and Assistance

Should you have any question about the Product that is not covered in this manual, contact your sales representative for solution. Please include the following information in your question:



- Product name and PO number;
- Complete description of the problem;
- Error message you received, if any.

Vantron Technology, Inc.

Address: 48434 Milmont Drive, Fremont, CA 94538
Tel: (650) 422-3128
Email: sales@vantrontech.com

Symbology

This manual uses the following signs to prompt users to pay special attention to relevant information.







	Caution for latent damage to system or harm to personnel
	Attention to important information or regulations

General Safety Instructions

The Product is supposed be installed by knowledgeable, skilled persons familiar with local and/or international electrical codes and regulations. For your safety and prevention of damage to the Product, please read and observe carefully the following safety instructions prior to installation and operation. Keep this manual well for future reference.

- Do not disassemble or otherwise modify the Product. Such action may cause heat generation, ignition, electronic shock, or other damages including human injury, and may void your warranty.
- Keep the Product away from heat source, such as heater, heat dissipater, or engine casing.
- Do not insert foreign materials into any opening of the Product as it may cause the Product to malfunction or burn out.
- To ensure proper functioning and prevent overheating of the Product, do not cover or block the ventilation holes of the Product.
- Follow the installation instructions with the installation tools provided or recommended.
- The use or placement of the operation tools shall comply with the code of practice of such tools to avoid short circuit of the Product.
- Cut off the power before inspection of the Product to avoid human injury or product damage.

Precautions for Power Cables and Accessories

-  Use proper power source only. Make sure the supply voltage falls within the specified range.
-  Place the cables properly at places without extrusion hazards.
-  There is a coin cell battery for powering the RTC. Therefore, please avoid short circuit of the battery during transportation or operation at high temperatures.
-  Cleaning instructions:
 - Power off before cleaning the Product
 - Do not use spray detergent
 - Clean with a damp cloth
 - Do not try to clean exposed electronic components unless with a dust collector
-  Power off and contact Vantron technical support engineer in case of the following faults:
 - The Product is damaged
 - The temperature is excessively high
 - Fault is still not solved after troubleshooting according to this manual
-  Do not use in combustible and explosive environment:
 - Keep away from combustible and explosive environment
 - Keep away from all energized circuits
 - Unauthorized removal of the enclosure from the device is not allowed
 - Do not change components unless the power cable is unplugged
 - In some cases, the device may still have residual voltage even if the power cable is unplugged. Therefore, it is a must to remove and fully discharge the device before replacement of the components.

CHAPTER 1 INTRODUCTION

1.1 Product Overview

VT-SBC-3568 single board computer is based on Rockchip RK3568 processor that integrates quad-core ARM Cortex-A55 CPU and Mali G52 GPU to provide optimized performance at lower power consumption, and offer high-quality video encoding and decoding for better display performance.

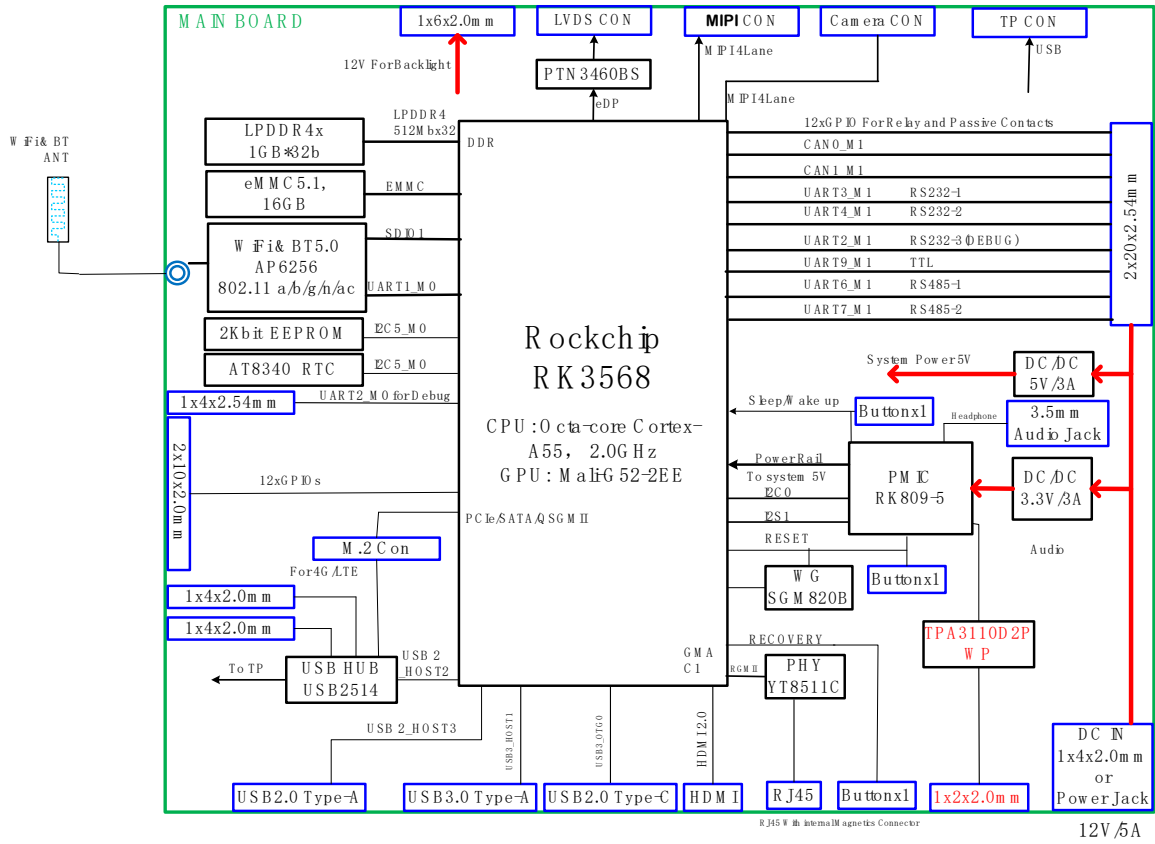
While both wired and wireless network accesses are available, user data is kept safe and secure in transmission. Meanwhile, VT-SBC-3568 provides a range of customer expansion options to meet the requirements of different application purposes, especially in industrial IoT scenarios.

Featuring high flexibility and high performance, VT-SBC-3568 could work under extreme environments at industrial-grade temperatures ranging from 0°C to +60°C, making it a reliable industrial IoT solution.

1.2 Terminology

Terminology	Description
NC	No connection
VCC	Voltage common collector
GND	Ground
/	Active low signal
+	Positive of difference signal
-	Negative of difference signal
I	Input
O	Output
I/O	Input/output
P	Power or ground
A	Analog
OD	Open drain
CMOS	3.3 V CMOS
LVC MOS	Low Voltage CMOS
LVTTL	Low Voltage TTL
3.3V	3.3 V signal level
5V	5V signal level
USB	5V tolerant signal
PCIe	PCI Express signal, not 3.3 V tolerant
MMC	Multimedia Card

1.3 Block Diagram



1.4 Specifications

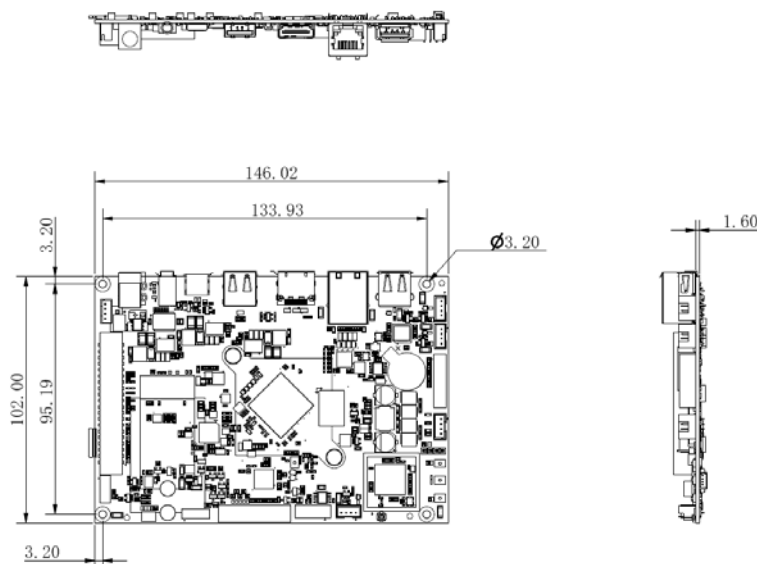
VT-SBC-3568		
System	CPU	RK3568, Quad-core ARM Cortex-A55 MPCore, up to 2.0 GHz
	GPU	ARM Mali-G52
	Memory	4GB LPDDR4 (Optional: 2GB)
	Storage	16GB eMMC V5.1, up to 128GB 2Kb EEPROM
Communication	Ethernet	1 x RJ45, 10/100/1000Mbps 100Base-T4 (surge suppression)
	Cellular	4G/5G (Optional)
	Wi-Fi & BT	802.11 a/b/g/n/ac & BT 5.0
Media	Display	Dual LVDS, Resolution up to 1920 x 1080, with backlight connector (Optional: eDP, Resolution up to 2K) 1 x MIPI DSI
	TP	Infrared touch panel (Optional)
	HDMI	1 x HDMI 2.0
	Camera	1 x 4-lane MIPI CSI
	Audio	1 x 3.5mm audio jack 1 x Speaker, up to 15W
I/Os	Serial	5 x UART (TTL) 1 x UART for debug
	USB	1 x USB 3.0 3 x USB 2.0 1 x USB 2.0 Type-C
	GPIO	24 x GPIO, up to 40 x GPIO
	SIM slot	1 x Micro SIM slot
	RTC	Supported
	WDT	Supported
Expansion	Bus	2 x CAN, without transmitter 1 x I ² C for TP 1 x USB 2.0 host for TP 1 x M.2 key B for 4G/5G module or SATA SSD
System Control	Button	1 x Reset 1 x Power button 1 x Volume button
Power	Input	12V/3A DC
Software	Operating system	Debian 10, Android 11+
Mechanical	Dimensions	146.02 x 102mm
	Cooling mode	Fanless
Environment Condition	Temperature	Operating: 0°C~+60°C (RK3568), -20°C~+70°C (RK3568J) Storage: -20°C~+70°C (RK3568), -40°C~+85°C (RK3568J)
	Humidity	RH ≤96% (Non-condensing)

1.5 Operating system

VT-SBC-3568 supports Debian 10 and Android 11+ operating systems.

1.6 Mechanical Dimensions

- 146.02mm x 102mm



1.7 Power Supply and Consumption

VT-SBC-3568 works with 12V/3A DC power supply.

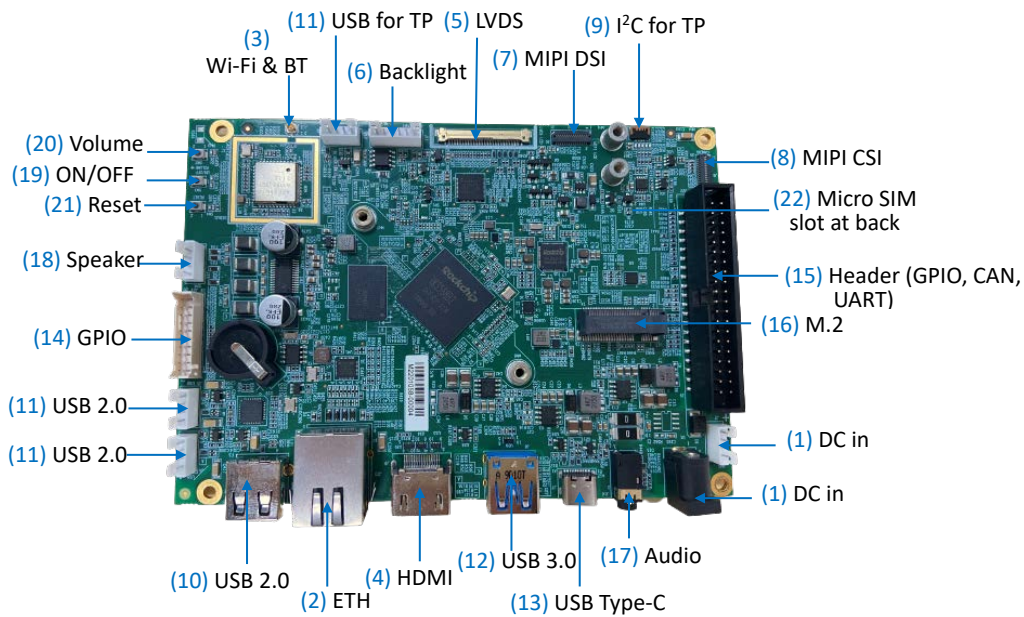
The power consumption of the Board is 30W at the maximum. It should be pointed out that power consumption is largely determined by the RAM, storage capacity, and other configurations of the board.


1.8 Environmental Specifications

VT-SBC-3568 works at a temperature ranging from 0°C to +60°C (for 3568J: -20°C to +70°C) and is designed to be stored at a temperature ranging from -20°C to +70°C (for 3568J: -40°C~+85°C) and at relative humidity of no more than RH 96% for non-condensing purpose.

CHAPTER 2 CONNECTOR DESCRIPTION

2.1 Product Layout



 The board I/Os will be described in detail in 2.4 Connectors and Jumpers following the sequencing numbers provided here.

2.2 Memory and Storage

2.2.1 LPDDR4 RAM

VT-SBC-3568 is equipped with a 4GB LPDDR4 RAM by default, and users also have the option of 2GB RAM.

2.2.2 eMMC Flash

VT-SBC-3568 provides an eMMC 5.1 flash up to 128 GB, and the default capacity is 16GB. It is used as the default boot and storage device.

2.2.3 EEPROM

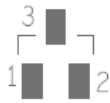
VT-SBC-3568 provides a 2Kb EEPROM to store hardware configuration information.

2.3 Identification of Pin 1

Unless otherwise stated, pin 1 of a connector is seated on a square pad that is different from the round pads used for other pins. Sometimes, pin 1 is next to a trigonal mark on the board. When there are two rows of pins on a connector, the row with pin 1 is composed of odd numbers and the other is composed of even numbers.



Usually, there will be numbers or marks next to the pins of a connector on the board to indicate the pinouts.

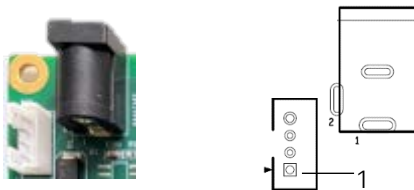


2.4 Connectors and Jumpers

This section is going to brief the connectors/jumpers on the Board with corresponsive pinout description.

2.4.1 Power input (1)

VT-SBC-3568 provides a 4-pin power connector and a power jack to supply power for the Board.



Pinout description of the 4-pin power connector:

Pin	Name	Type	Description
1	+VDC	P	DC-IN POWER +
2	+VDC	P	DC-IN POWER +
3	-VDC	P	DC-IN POWER -
4	-VDC	P	DC-IN POWER -

2.4.2 J23 Ethernet port (2)

VT-SBC-3568 offers an RJ45 Ethernet jack with two LEDs, green for activity indication and yellow for link indication. The port supports 10/100/1000 Mbps transmission rate.

2.4.3 J13 Wi-Fi and Bluetooth (3)

VT-SBC-3568 offers an AP6256 1T1R combo SiP module, combining Wi-Fi 802.11 a/b/g/n/ac and Bluetooth 5.0. There is an antenna interface next to the module for connecting the Wi-Fi & Bluetooth antenna.

2.4.4 J9 HDMI (4)

VT-SBC-3568 offers a standard HDMI Type-A interface for image output. The pinout description of the interface is in line with the pin assignment of standard HDMI Type-A interface.

2.4.5 J11 LVDS (5)

VT-SBC-3568 offers a dual LVDS interface to connect high-definition displays (resolution up to 1920 x 1080).

Specifications: 1 x 40, 0.5mm, 0.5A, 1.00mm (H), male, RA, SMT, RoHS



Pinout description:

Pin	Name	Type	Description
1	PANEL_BL_PWM	O	LCD backlight PWM control output
2	PANEL_BKLTEN	O	LCD backlight power control output
3	LVDS_B_D3+	O	LVDS B Lane3 +
4	LVDS_B_D3-	O	LVDS B Lane3 -
5	LVDS_B_CLK+	O	LVDS B CLK+
6	LVDS_B_CLK-	O	LVDS B CLK-
7	NC		
8	LVDS_B_D2+	O	LVDS B Lane2 +
9	LVDS_B_D2-	O	LVDS B Lane2 -
10	LVDS_B_D1+	O	LVDS B Lane1 +
11	LVDS_B_D1-	O	LVDS B Lane1 -

12	SEL68	O	
13	NC		
14	eDP_HPDP	I	Hot Plug Detect signal
15	LVDS_B_D0+	O	LVDS B Lane0 +
16	LVDS_B_D0-	O	LVDS B Lane0 -
17	LVDS_DDC_DATA	I/O	I2C_SDA
18	LVDS_DDC_CLK	O	I2C_SCL
19	GND	P	Ground
20	LVDS_A_D3+	O	LVDS A Lane3 +
21	LVDS_A_D3-	O	LVDS A Lane3 -
22	GND	P	Ground
23	LVDS_A_CLK+	O	LVDS A CLK +
24	LVDS_A_CLK-	O	LVDS A CLK -
25	GND	P	Ground
26	LVDS_A_D2+	O	LVDS A Lane2 +
27	LVDS_A_D2-	O	LVDS A Lane2 -
28	GND	P	Ground
29	LVDS_A_D1+	O	LVDS A Lane1 +
30	LVDS_A_D1-	O	LVDS A Lane1 -
31	GND	P	Ground
32	LVDS_A_D0+	O	LVDS A Lane0 +
33	LVDS_A_D0-	O	LVDS A Lane0 -
34	GND	P	Ground
35	NC		
36	NC		
37	LCD_VDD	P	Power supply for LCD
38	LCD_VDD	P	Power supply for LCD
39	LCD_VDD	P	Power supply for LCD
40	LCD_VDD	P	Power supply for LCD

2.4.6 J12 Backlight connector (6)

VT-SBC-3568 offers a backlight connector that is designed to connect a backlight to increase readability of the LCD in low light conditions.

Specifications: 1 x 6, 2.0mm, 2A, 6mm (H), male, RA, -25°C~85°C, THR, RoHS



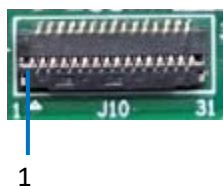
Pinout description:

Pin	Name	Type	Description
1	LCD_BLK	P	Power supply 12V for backlight
2	LCD_BLK	P	Power supply 12V for backlight
3	PANEL_BKLTEN	O	LCD backlight power control output
4	PANEL_BL_PWM	O	LCD backlight PWM control output
5	GND	P	Ground
6	GND	P	Ground

2.4.7 J10 MIPI DSI (7)

VT-SBC-3568 offers a 4-lane MIPI DSI connector for connecting displays.

Specifications: 1 x 31, 0.3mm, 1.0mm(H), RA, Bot, WDT, SMT, RoHS



Pinout description:

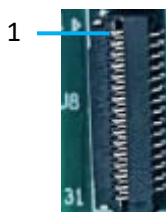
Pin	Name	Type	Description
1	LED+	P	Power supply for backlight
2	LED+	P	Power supply for backlight
3	LED+	P	Power supply for backlight
4	NC		
5	LED-	P	Feedback for current of backlight
6	LED-	P	Feedback for current of backlight
7	LED-	P	Feedback for current of backlight
8	LED-	P	Feedback for current of backlight
9	GND	P	Ground
10	GND	P	Ground

11	MIPI_D2P	O	MIPI DSI Lane2 +
12	MIPI_D2N	O	MIPI DSI Lane2 -
13	GND	P	Ground
14	MIPI_D1P	O	MIPI DSI Lane1 +
15	MIPI_D1N	O	MIPI DSI Lane1 -
16	GND	P	Ground
17	MIPI_CLK+	O	MIPI DSI CLK+
18	MIPI_CLK-	O	MIPI DSI CLK-
19	GND	P	Ground
20	MIPI_D0P	O	MIPI DSI Lane0 +
21	MIPI_D0N	O	MIPI DSI Lane0 -
22	GND	P	Ground
23	MIPI_D3P	O	MIPI DSI Lane3 +
24	MIPI_D3N	O	MIPI DSI Lane3 -
25	GND	P	Ground
26	ID	I	LCD ID
27	RST_LCD	O	LCD RESET
28	NC		
29	VCCIO_1V8	P	power supply 1.8V
30	VCC3V3_LCD	P	power supply 3.3V
31	VCC3V3_LCD	P	power supply 3.3V

2.4.8 J8 MIPI CSI (8)

VT-SBC-3568 offers a 4-lane MIPI CSI connector for connecting cameras.

Specifications: 1 x 31, 0.3mm, 1.0mm(H), RA, Bot, WDT, SMT, RoHS



Pinout description:

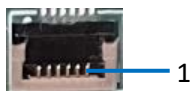
Pin	Name	Type	Description
1	GND	P	Ground
2	MIPI_CSI_D3N	A	MIPI CSI Lane3 -
3	MIPI_CSI_D3P	A	MIPI CSI Lane3 +
4	GND	P	Ground
5	MIPI_CSI_D2N	A	MIPI CSI Lane2 -
6	MIPI_CSI_D2P	A	MIPI CSI Lane2 +

7	GND	P	Ground
8	MIPI_CSI_D1N	A	MIPI CSI Lane1 -
9	MIPI_CSI_D1P	A	MIPI CSI Lane1 +
10	GND	P	Ground
11	MIPI_CSI_D0N	A	MIPI CSI Lane0 -
12	MIPI_CSI_D0P	A	MIPI CSI Lane0 +
13	GND	P	Ground
14	MIPI_CSI_CLKON	A	MIPI CSI CLK-
15	MIPI_CSI_CLKOP	A	MIPI CSI CLK+
16	GND	P	Ground
17	I2C2_SCL_M1_1V8	O	I2C_SCL
18	I2C2_SDA_M1_1V8	I/O	I2C_SDA
19	MIPI_CAM_RST	O	Camera RESET
20	MIPI_CAM_PWN	O	Camera Power Down
21	GND	P	Ground
22	MIPI_CAM_MCLK	O	Camera Main CLK
23	GND	P	Ground
24	NC		
25	VCC1V8_DVP	P	Power supply 1.8V
26	VCC1V8_DVP	P	Power supply 1.8V
27	VCC1V5_DVP	P	Power supply 1.5V
28	VCC2V8_DVP	P	Power supply 2.8V
29	VCC2V8_DVP	P	Power supply 2.8V
30	NC		
31	GND	P	Ground

2.4.9 J22 I²C (9)

VT-SBC-3568 offers an I²C interface for connecting a touch panel.

Specifications: 1 x 6, 0.5mm, 0.4A, 0.9mm(H), female, RA, WDT, SMT, RoHS



Pinout description:

Pin	Name	Type	Description
1	VCC3V3_PMU	P	Power supply 3.3V
2	GND	P	Ground
3	I2C1_SCL_TP	O	I2C_SCL for TP
4	I2C1_SDA_TP	I/O	I2C_SDA for TP
5	TP_INT	I	TP interrupt
6	TP_RST	O	TP RESET

2.4.10 U40 USB 2.0 Type-A (10)

VT-SBC-3568 is designed to connect peripherals via the USB 2.0 Type-A interface to expand the functions.

The pinout description of the USB 2.0 Type-A interface is in line with the pin assignment of standard USB 2.0 Type-A.



USB 2.0 Type-A

2.4.11 J3/J4/J5 USB 2.0 (11)

There are three USB 2.0 connectors, one could be used for connection of a touch panel, and the other two are in host mode by default (chipset USB2514).

Specifications: 1 x 4, 2.0mm, 2A, 6mm(H), male, RA, WDT, THR, RoHS



Pinout description:

Pin	Name	Type	Description
1	GND	P	Ground
2	HUB_HOST_DP	I/O	USB DATA+
3	HUB_HOST_DM	I/O	USB DATA-
4	HUB_HOST_5V	P	Power supply 5V for USB

2.4.12 J6 USB 3.0 (12)

VT-SBC-3568 offers a USB 3.0 (Type-A) interface with pinout in line with the pin assignment of standard USB 3.0 Type-A.



2.4.13 J7 USB 2.0 Type-C (13)

VT-SBC-3568 offers a USB 2.0 Type-C interface.

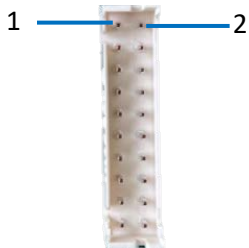
Specifications: Max. current output: 0.5A, OTG supported.



2.4.14 J21 GPIO (14)

There is a GPIO header supporting 12 GPIOs and user can customize the use of the pins based on need.

Specification: 2 x 10, 2.0 mm, 6.5 mm (H), Vert, -25~85°C, THR, RoHS



Pinout description (based on a user case involving the use of the pins for LED and buttons):

Pin	Name	Type	Description
1	LED_1	O	LED output signal
2	BUTTON_1	I	Button input data
3	LED_2	O	LED output signal
4	BUTTON_2	I	Button input data
5	LED_3	O	LED output signal
6	BUTTON_3	I	Button input data
7	LED_4	O	LED output signal
8	BUTTON_4	I	Button input data
9	LED_5	O	LED output signal
10	BUTTON_5	I	Button input data
11	LED_6	O	LED output signal
12	BUTTON_6	I	Button input data
13	VCC3V3_EXT	P	Power supply 3.3V
14	VCC3V3_EXT	P	Power supply 3.3V
15	GND	P	Ground
16	GND	P	Ground
17	VCC_1V8	P	Power supply 1.8V
18	VCC_1V8	P	Power supply 1.8V
19	GND	P	Ground
20	GND	P	Ground

2.4.15 J20 40-pin header (15)

The 40-pin header on VT-SBC-3568 supports 2 CAN buses, 6 UART ports, and 12 GPIOs (customized as alarm in and out as set out in the pinout table). The UART ports operate at TTL level, with one dedicated to debugging (UART 2), and the remaining ports (UART 3, UART 4, UART 6, UART 7, UART 9) designated for serial communication. The communication ports are mapped as ttyS3, ttyS4, ttyS6, ttyS7, ttyS9, respectively in the software system. Users also have the option to configure the aforementioned ports as GPIOs, thereby supporting up to 40 GPIOs.

Specifications: 2 x 20, 2.54mm, 8.8 mm(H), Vert, -45~105°C, SMT, RoHS



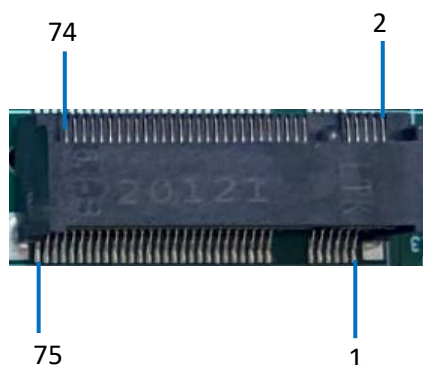
Pinout description:

Pin	Name	Type	Description
1	CAN0_RX_M1	I	CAN0 serial port data input
3	CAN0_TX_M1	O	CAN0 serial port data output
5	CAN1_RX_M1	I	CAN1 serial port data input
7	CAN1_TX_M1	O	CAN1 serial port data output
9	ALARM_OUT1	O	Alarm output
11	ALARM_OUT2	O	Alarm output
13	ALARM_OUT3	O	Alarm output
15	ALARM_OUT4	O	Alarm output
17	ALARM_IN1	I	Alarm input
19	ALARM_IN2	I	Alarm input
21	ALARM_IN3	I	Alarm input
23	ALARM_IN4	I	Alarm input
25	ALARM_IN5	I	Alarm input
27	ALARM_IN6	I	Alarm input
29	ALARM_IN7	I	Alarm input
31	EXGND	P	Ground
33	EXGND	P	Ground
35	EXGND	P	Ground

37	EXGND	P	Ground
39	EXGND	P	Ground
2	UART9_RX_M1	I	UART9 receive data
4	UART9_TX_M1	O	UART9 transmit data
6	UART2_RX_M0_DEBUG	I	UART2 receive data
8	UART2_TX_M0_DEBUG	O	UART2 transmit data
10	UART4_RX_M1	I	UART4 receive data
12	UART4_TX_M1	O	UART4 transmit data
14	UART3_RX_M1	I	UART3 receive data
16	UART3_TX_M1	O	UART3 transmit data
18	UART7_RX_M1	I	UART7 receive data
20	UART7_TX_M1	O	UART7 transmit data
22	RS485_DIR1_GPIO3_B5	I/O	GPIO3_B5 control output signal
24	UART6_RX_M1	I	UART6 receive data
26	UART6_TX_M1	O	UART6 transmit data
28	RS485_DIR2_GPIO3_B6	I/O	GPIO3_B6 control output signal
30	ALARM_IN8	I	Alarm input
32	DC_IN	P	Power input
34	DC_IN	P	Power input
36	DC_IN	P	Power input
38	DC_IN	P	Power input
40	DC_IN	P	Power input

2.4.16 J17 M.2 Key B slot (16)

VT-SBC-RK3568 offers an M.2 Key B slot, supporting PCIe to connect a 3G/4G/5G module and supporting SATA to connect an SSD.



Pinout description:

Pin	Name	Type	Description	Comment
1, 20, 22, 23, 24, 28, 29, 31, 35, 37, 38, 40, 42, 44, 46, 48, 56, 58, 59, 60, 61, 62, 63, 64, 65, 66, 68	NC	NC		
2, 4, 70, 72, 74	PCIE_3V3	P		
3, 5, 11, 27, 33, 39, 45, 51, 57, 71, 73, 76, 77	GND	P	Ground	
36	5G_USIM_PWR	P	USIM power output	Either 1.8 V or 3.0 V is supported by the module automatically.
34	5G_USIM_DATA	I/O	USIM data signal	
32	5G_USIM_CLK	I	USIM clock signal	
30	5G_USIM_RESET	I	USIM reset signal	
66	SIM_DETECT1	O	USIM Plug Detect signal	
7	USBDP_DN4	IO	USB data+	
9	USBDM_DN4	IO	USB data-	
41	PCIE_TXN_A	O	PCIE 2.0 transmit data-	
43	PCIE_TXP_A	O	PCIE 2.0 transmit data+	
47	PCIE_RXN_A	I	PCIE 2.0 receive data-	
49	PCIE_RXP_A	I	PCIE 2.0 receive data+	
53	PCIE_CLKN_A	I	PCIE 2.0 CLK-	
55	PCIE_CLKP_A	I	PCIE 2.0 CLK+	
6	5G_PWR_OFF#1_1V8	I	5G power on and off control	1.8 V power domain
8	5G_DISABLE#1_1V8	O	5G Flight mode control data	
67	WWAN_RESET#1	I	5G RESET	

2.4.17 J16 Audio jack (17)

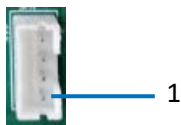
VT-SBC-RK3568 offers a 3.5mm audio jack.



2.4.18 J15 Speaker connector (18)

VT-SBC-RK3568 offers a left/right speaker connector that is designed to connect a 5W/8Ω speaker.

specifications: 1 x 4, 2.0mm, 2A, 6mm (H), male, Vert, WDT, THR, RoHS



Pinout description:

Pin	Name	Type	Description
1	OUTL+A	A	5W speaker +
2	OUTL-A	A	5W speaker -
3	OUTR-A	A	5W speaker -
4	OUTR+A	A	5W speaker +

2.4.19 Buttons (19-21)

VT-SBC-3568 offers three buttons, including a volume button, an on/off button, and a reset button.



2.4.20 J18 Micro SIM (22)

VT-SBC-3568 offers a Micro SIM card slot.

Specifications: Micro SIM, push-push, -25°C~90°C, No WP, SMT, RoHS

CHAPTER 3 ANDROID SYSTEM MANUAL

3.1 Enable Developer Options

To enable Developer Options of VT-SBC-3568, follow the steps below:

1. Connect the Board to a mouse, a keyboard, and a display for easier operations;
2. After the system has booted, swipe up to access the application drawer;
3. Click **Settings > About <device>** in sequence;
4. Scroll down to **Build number**, and click it consecutively for at least 7 times to enable **Developer options**;
5. Go back to **Settings > System > Advanced > Developer options** and toggle on **USB debugging**, then you can customize the Board settings.

 Depending on the Android version, the entry might vary slightly.

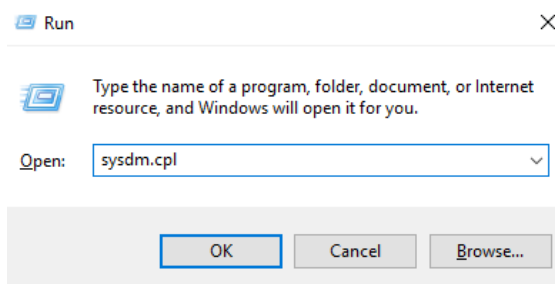
3.2 ADB Setup on the Windows Host

Android Debug Bridge (ADB) is a tool that is designed to connect your development workstation directly to your Android device for debugging, device upgrading, app installation, etc.

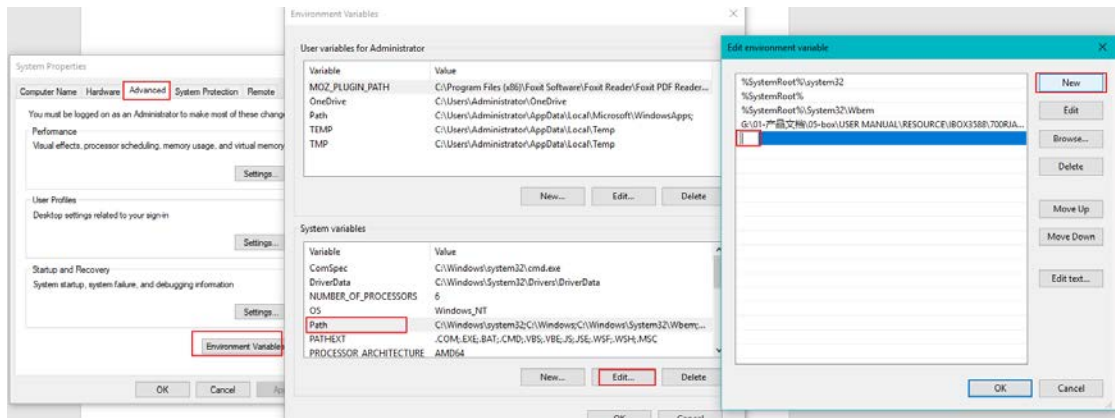
Adding the ADB executable file to the system's environment variable allows you to run the ADB tool regardless of your current working directory.

Follow the steps below to set up the ADB on the Windows host computer.

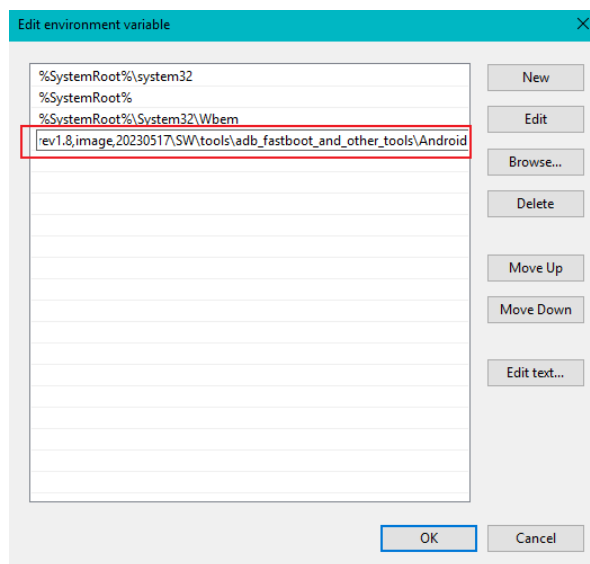
1. Unzip the software release package and navigate to the following directory: \SW\tools;
2. Extract the **adb_fastboot_and_other_tools_for_windows** zip file;
3. Navigate to the **Android** folder that contains the adb tool and copy the folder path;
4. Press "Win + R" and input `sysdm.cpl` in the dialogue box to open the settings interface;



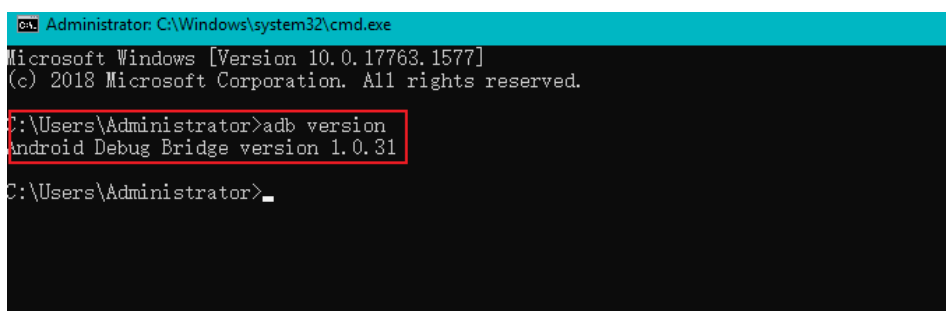
5. Click in sequence **Advanced > Environment Variables > System Variables > Path > Edit**, and click **New** in the pop-up;



6. Paste the folder path of the adb tool, and click **OK** one by one to confirm and exit the dialog;



7. Press "Win + R" and input `cmd` to open the command prompt;
8. Input `adb version` in the command prompt to check if the adb tool is installed.



3.3 App Installation via ADB Commands

In addition to the standard pre-installed Android applications, users can install their own applications on the Board. All tools to be used for the installation are available in the product release package provided by Vantron.

You can install the apps either in the Windows environment or Linux environment as long as the Android SDK Platform Tools are installed and accessible on the host computer. The commands for the installation are the same. The Windows environment is used for illustration in the following section.

1. Connect VT-SBC-3568 to the Windows host computer via a USB Type-A to Type-C cable;
2. Press “Win + R” and input `cmd` in the dialogue box to open the command prompt;
3. Input `adb devices -l` in the command prompt to check if the Board is connected to the host computer;

```
C:\Users\Administrator>adb devices -l
List of devices attached
874cc0aaede7d049    device product:occam model:Nexus_4 device:mako transport_id:1
```


4. When the device information is displayed under the command, you can copy the serial number (squared as shown above) for next-step use;
5. Input the following command line to install the app;


```
adb -s <serial number> install <APP path>
```


6. The installation will be executed after the command is input and the result of installation will be displayed below;

```
C:\Users\Administrator>adb -s 674cc0aaede7d049 install "C:\Users\Administrator\Desktop\Libraries for developers_v3.83_apkpure.com.apk"
Performing Streamed Install
Success
```

7. The newly installed app will be displayed on the App drawer in the Alphabetic order.

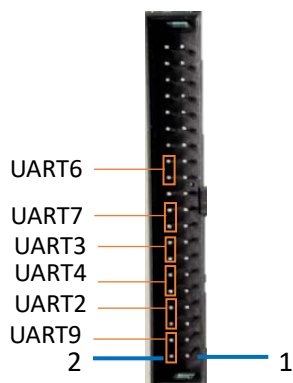
 *The screenshot is for illustration only and is not intended to represent the actual device number and other information of the device currently in your possession.*

 *In step 5 shown above, you can drag the .apk file from the local directory to the command line to replace the <App path> you typed in manually.*

 *If you failed to install the apk, try using the absolute path of the .apk file enclosed in double quotation marks.*

3.4 Serial port

VT-SBC-3568 offers 6 UART ports operating at TTL level, with one dedicated to debugging (UART 2), and the remaining ports (UART 3, UART 4, UART 6, UART 7, UART 9) designated for serial communication. The communication ports are mapped as ttyS3, ttyS4, ttyS6, ttyS7, ttyS9, respectively in the software system.



To test if the serial ports function properly, use a USB-to-TTL adapter to connect a serial port (TX-RX, RX-TX, GND-GND) to the host computer first, then use a serial communication program to check if the serial port functions properly.

1. Receive data sent to the serial port (UART 6 for instance);

```
# cat /dev/ttyS6
```

2. Send data ("TEST") from the serial port.

```
# echo TEST > /dev/ttyS6
```

3.5 CAN

VT-SBC-3568 implements two CAN interfaces (CAN0 & CAN1) with a default bitrate of 20 kHz. Refer to 2.4.15 for the pinout description.

1. Bring up the interface (CAN0 for instance) with a bitrate that the device on the CAN network is operating at (for instance, 125000):

```
# ip link set can0 up type can bitrate 125000
```

2. Receive data:

```
# candump can0
```

3. Transmit data:

```
# cansend can0 123#1DEADBEEF
```

3.6 Connecting to Internet

For Ethernet communication, VT-SBC-3568 is designed to have internet access a few seconds after you have plugged a live Ethernet cable into the RJ45 jack of the Board.

For Wi-Fi connection, navigate to **Settings > Network & internet > Wi-Fi** to enable Wi-Fi and join the target access point.


For 4G connection, insert an activated SIM card into the SIM slot of the Board. The card will be automatically registered and connected to internet.

The Board follows a networking priority of Ethernet > Wi-Fi > 4G.

CHAPTER 4 DEBIAN SYSTEM MANUAL

This chapter is mainly about the first-use debugging of interfaces and software applications, and most of the operations are done in the console of VT-SBC-3568.

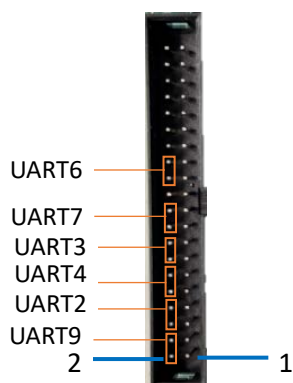
On the other hand, you can also use a USB to TTL (3.3V) serial adapter to connect UART 2 (baud rate: 150000bps) to a Linux host for the debugging purpose.

 Please execute all the commands with the root privilege, and **no** password is required.

4.1 Interface Definition

4.1.1 Serial port

VT-SBC-3568 offers 6 UART ports operating at TTL level, with one dedicated to debugging (UART 2), and the remaining ports (UART 3, UART 4, UART 6, UART 7, UART 9) designated for serial communication. The communication ports are mapped as ttyS3, ttyS4, ttyS6, ttyS7, ttyS9, respectively in the software system.



To test if the serial ports function properly, use a USB-to-TTL adapter to connect a serial port (TX-RX, RX-TX, GND-GND) to the host computer first, then use a serial communication program to check if the serial port functions properly.

3. Set the baud rate a serial port (UART 6 for instance);

```
# stty -F /dev/ttyS6 speed 115200
```

4. Receive data sent to the serial port;

```
# cat /dev/ttyS6
```

5. Send data ("TEST") from the serial port.

```
# echo TEST > /dev/ttyS6
```

4.1.2 USB interface

Insert a USB device to a USB interface of the Board, and input the following commands to check/configure the USB interface.

1. Check the information of USB interfaces:

```
# lsusb
```

2. Mount a USB device to /mnt directory:

```
# mount /dev/xxx /mnt
```

3. Check the mounted content:

```
# ls /mnt
```

4. Unmount the USB device:

```
# umount /dev/xxx
```

4.1.3 GPIO

As mentioned in 2.4.14 and 2.4.15, the I/O pins on both J21 and J20 can be configured as GPIOs based on the use case of the customer. Currently, pin 22 (GPIO3_B5) and pin 28 (GPIO3_B6) on J20 are available for use as GPIOs in Debian system, which are mapped as gpio109 and gpio110, respectively.

1. Check the status of GPIOs:

```
# cat /sys/kernel/debug/gpio
```

2. Export GPIO3_B5:

```
# echo 109 > /sys/class/gpio/export
```

3. Set the pin direction of GPIO3_B5 as output:

```
# echo out > /sys/class/gpio/gpio109/direction
```

4. Set the value of the pin to 0 or 1 (corresponding to low or high):

```
# echo 0 > /sys/class/gpio/gpio109/value [set it low], or
```

```
# echo 1 > /sys/class/gpio/gpio109/value [set it high]
```

5. Read the GPIO value.

```
# cat /sys/class/gpio/gpio109/value
```

Commands to export and set the level of GPIO3_B6 are same as those for GPIO3_B5 except that you need to replace 109 with the 110 when it comes to the value of GPIO3_B6.

4.2 Ethernet

Connect the Ethernet jack of the Board to switch/gateway with an Ethernet cable, then check the settings of the Ethernet jack.

```
# ifconfig
```

```
root@vantron:~#  
root@vantron:~# ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.9.135 netmask 255.255.255.0 broadcast 192.168.9.255  
    inet6 fe80::2532:bf6e:b268:9f21 prefixlen 64 scopeid 0x20<link>  
    ether 06:46:c6:0b:0a:3c txqueuelen 1000 (Ethernet)  
    RX packets 1951 bytes 219986 (214.8 KiB)  
    RX errors 0 dropped 23 overruns 0 frame 0  
    TX packets 167 bytes 16785 (16.3 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
    device interrupt 24  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1 (Local Loopback)  
    RX packets 68 bytes 4780 (4.6 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 68 bytes 4780 (4.6 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Judging from the above screenshot, the Ethernet jack is identified as eth0 in the system with an IP address 192.168.9.135.

Set a static IP address to the Ethernet jack and verify the IP address:

```
# ifconfig eth0 192.168.9.10  
  
# ping 192.168.9.10  
  
PING 192.168.9.10 (192.168.9.10): 56 data bytes  
  
64 bytes from 192.168.9.10: seq=0 ttl=64 time=1.296 ms  
  
64 bytes from 192.168.9.10: seq=1 ttl=64 time=1.358 ms
```

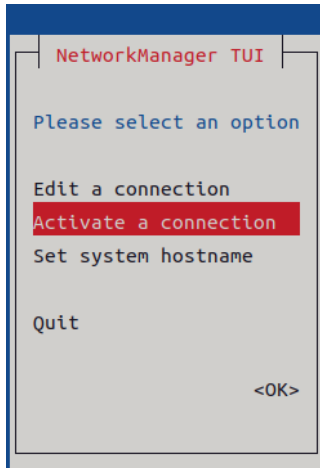
4.3 Wi-Fi

VT-SBC-3568 supports Wi-Fi and Bluetooth functions. You are recommended to use nmtui to connect the Board to an existing Wi-Fi access point.

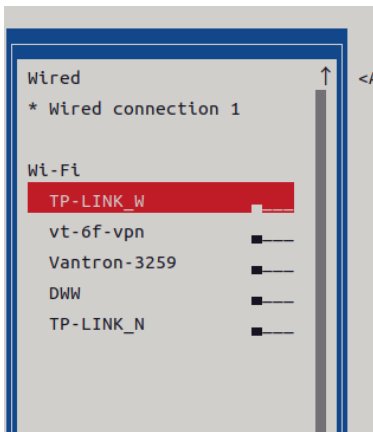
1. Open a terminal and input the following command to set up the network;

```
$ nmtui
```

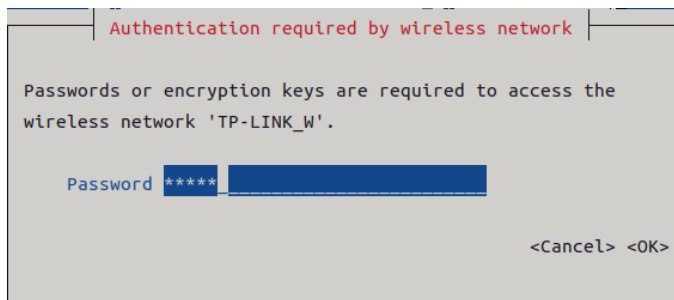
2. Use the arrows on the keyboard to navigate to **Activate a connection** and press **Enter**;



3. Select a Wi-Fi network from the list;



4. Input the password of the access point and verify;



5. You will connect to the access point after successful verification.

You can also connect a Wi-Fi network using commands.

1. Input the following command in the terminal:

```
# nmcli dev wifi connect <SSID> password <PASSWD>
```

```
root@vantron:~# nmcli dev wifi connect denghaiting password 12345678
[ 1183.548930] [dhd-wlan0] wl_run_escan : LEGACY_SCAN sync ID: 22, bssid: 0
[ 1184.391761] [dhd-wlan0] wl_iw_event : Link Down with b0:f1:ec:82:45:ae, reason=4
[ 1184.391909] [dhd-wlan0] wl_ext_iapsta_event : [S] Link down with b0:f1:ec:82:45:ae, WLC_E_LINK(16), reason 4
[ 1184.451418] [dhd-wlan0] wl_cfg80211_connect : Connecting with 38:37:8b:e9:7a:30 ssid "denghaiting", len (11), sec
=wpa2psk/aes, channel=1
[ 1184.451418] [dhd-wlan0] wl_notify_connect_status : wl_bss_connect_done succeeded with 38:37:8b:e9:7a:30
[ 1184.576719] [dhd-wlan0] wl_iw_event : Link UP with 38:37:8b:e9:7a:30
[ 1184.576867] [dhd-wlan0] wl_ext_iapsta_event : [S] Link UP with 38:37:8b:e9:7a:30
[ 1184.619664] [dhd-wlan0] wl_notify_connect_status : wl_bss_connect_done succeeded with 38:37:8b:e9:7a:30 vndr_oui:
AC-85-3D
device 'wlan0' successfully activated with '23cb37fb-8600-47d2-b30d-4ea133dfed2d'.
```

2. Use the ping command to check if the network is connected properly;

```
# ping www.baidu.com
```

```
root@vantron:~# ping www.baidu.com
PING www.a.shifen.com (39.156.66.18): 56 data bytes
64 bytes from 39.156.66.18: icmp_seq=0 ttl=51 time=81.353 ms
64 bytes from 39.156.66.18: icmp_seq=1 ttl=51 time=76.533 ms
64 bytes from 39.156.66.18: icmp_seq=2 ttl=51 time=79.002 ms
64 bytes from 39.156.66.18: icmp_seq=3 ttl=51 time=87.612 ms
^C--- www.a.shifen.com ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 76.533/81.125/87.612/4.115 ms
```

3. If the data packets are transmitted and received properly, the network is connected properly.

4.4 Cellular Network

VT-SBC-3568 implements a mini PCIe slot for connecting a 4G cellular module for wireless communication.

Before you set up the cellular network, please insert a cellular module to the mini PCIe slot, install the antennas and plug in an activated SIM card. Power up the Board and generally the cellular module will be booted.

Insert the activated SIM card and wait for the loading of the USB driver to obtain the name of the USB port:

1. Display the module information;

```
# lsusb
```

2. List the serial interface (ttyUSB) enumerate by the cellular module.

```
# ls /dev/ttyUSB*
```

▶ In the following section, a Quectel EC20 module is used to illustrate the steps for debugging the cellular module. Please refer to the manual of your own cellular module to determine which interface is used for the module debugging using AT commands and the general AT commands for that purpose.

- AT control

Use a serial port communication program (e.g., minicom) to configure the serial port **ttyUSB2** to send AT commands (only AT (ttyUSB2 in this case) and Modem (ttyUSB3 in this case) nodes of a serial port are used for sending AT commands).

1. Open ttyUSB2 via minicom;

```
# minicom -D /dev/ttyUSB2 -b 115200
```

2. In normal cases, the program will return the module information;

```
Welcome to minicom 2.7.1
OPTIONS: I18n
Compiled on Oct 15 2019, 11:16:53.
Port /dev/ttyUSB2, 09:39:53

Press CTRL-B Z for help on special keys

at
OK
ati
Quectel
EC20F
Revision: EC20CEHDLGR06A04M1G
OK
```

3. Input the following commands to check/test relevant information.

ATI -> Check product information

AT+CPIN? -> Check if a SIM card is identified with the corresponding PIN code information

AT+CSQ -> Check the signal quality

AT+CREG? -> Check the registration status of the SIM card

AT+COPS? -> Check the carrier information

- PPP dial-up

There are two ways to start PPP dial-up:

```
# pppd call quectel-ppp & // method 1
# /etc/ppp/peers/quectel-pppd.sh /dev/ttyUSB2 <APN> <username> <password>
// method 2
```

▶ APN, username and password are provided by the carrier. Please modify the arguments based on the actual information (the screenshots below are for illustration only).

```
root@vantron:~# /etc/ppp/peers/quectel-pppd.sh /dev/ttyUSB2 3gnet
quectel-pppd options in effect:
devname /dev/ttyUSB2 # (from command line)
apn 3gnet # (from command line)
user user # (default)
password passwd # (default)
root@vantron:~# pppd options in effect:
debug # (from command line)
nodetach # (from command line)
dump # (from command line)
noauth # (from command line)
user user # (from command line)
password ?????? # (from command line)
/dev/ttyUSB3 # (from command line)
115200 # (from command line)
lock # (from command line)
connect 'chat -s -v ABORT BUSY ABORT \"NO CARRIER\" ABORT \"NO DIALT
OK ATE0 OK ATI\\;+CSUB\\;+CSQ\\;+CPIN?\\;+COPS?\\;+CGREG?\\;\\&D2 OK
ONNEC
# (from command line)
disconnect chat -s -v ABORT ERROR ABORT \"NO DIALTONE\" SAY \"\\nSend
AY \"\\nGood bay\\n\" # (from command line)
nocrtscts # (from command line)
modem # (from command line)
asynmap 0 # (from /etc/ppp/options)
lcp-echo-failure 4 # (from /etc/ppp/options)
```

.....

```
rcvd [PAP AuthAck id=0x1 "" 00]
PAP authentication succeeded
sent [IPCP ConfReq id=0x1 <addr 0.0.0.0> <ms-dns1 0.0.0.0> <ms-dns2 0.0.0.0>]
rcvd [IPCP ConfReq id=0x2]
sent [IPCP ConfNak id=0x2 <addr 0.0.0.0>]
rcvd [IPCP ConfNak id=0x1 <addr 10.52.120.179> <ms-dns1 119.7.7.7> <ms-dns2 119.
6.6.6>]
sent [IPCP ConfReq id=0x2 <addr 10.52.120.179> <ms-dns1 119.7.7.7> <ms-dns2 119.
6.6.6>]
rcvd [IPCP ConfReq id=0x3]
sent [IPCP ConfAck id=0x3]
rcvd [IPCP ConfAck id=0x2 <addr 10.52.120.179> <ms-dns1 119.7.7.7> <ms-dns2 119.
6.6.6>]
could not determine remote IP address: defaulting to 10.64.64.64
not replacing default route to usb0 [192.168.43.1]
local IP address 10.52.120.179
remote IP address 10.64.64.64
primary DNS address 119.7.7.7
secondary DNS address 119.6.6.6
script /etc/ppp/ip-up started (pid 1213)
Script /etc/ppp/ip-up finished (pid 1213), status = 0x0
```

If the above steps go smoothly, users can input **ifconfig** in the terminal to check the network information, including the route (ppp0) and the IP address.

```
ppp0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500
      inet 10.52.120.179 netmask 255.255.255.255 destination 10.64.64.64
      ppp txqueuelen 3 (Point-to-Point Protocol)
      RX packets 4 bytes 52 (52.0 B)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 9 bytes 297 (297.0 B)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

- IP/DNS/Route

1. Check the DNS

```
# /etc cat resolv.conf
```

If PPPD acquires the IP and DNS, the `/etc/ppp/ip-up` script will be called to set the system DNS. The DNS will be automatically saved to `/etc/resolv.conf` after the setting.

2. Check the route


```
# /etc route -n
```

3. Test the network connection

```
# ping www.baidu.com
```

4. If the network connection fails, please try to add a default route and continue to test the network connection

```
# route add default gw <xx.xx.xx.xx>
```

 *xx.xx.xx.xx is the IP address of the route (ppp0)*

```
root@vantron:~#
root@vantron:~# route add default gw 10.52.120.179
root@vantron:~#
root@vantron:~# ping www.baidu.com
PING www.a.shifen.com (110.242.68.4): 56 data bytes
64 bytes from 110.242.68.4: icmp_seq=0 ttl=53 time=56.510 ms
64 bytes from 110.242.68.4: icmp_seq=1 ttl=53 time=43.238 ms
64 bytes from 110.242.68.4: icmp_seq=2 ttl=53 time=43.457 ms
^C--- www.a.shifen.com ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 43.238/47.735/56.510/6.206 ms
```

4.5 RTC

RTC module provides accurate time and date information to the system. Before the test, make sure a coin cell battery is installed on the Board and the voltage is kept at 3.0V.

If you are using the RTC for the first time, please calibrate the RTC time to the system time.

1. Set the system date & time;

```
# date -s "2023-02-24 14:38:10"
```

2. Synchronize the RTC time with the system time;

```
# hwclock -w
```

3. Reboot the Board;

```
# reboot
```

4. Check the RTC time information.

```
# hwclock -r
```

4.6 Watchdog Timer

The watchdog timer is disabled by default. When the watchdog timer is enabled, the system will reboot automatically within 45 seconds if the dog is not fed.

Enable the watchdog timer or feed the dog:

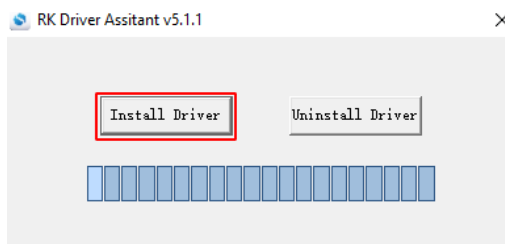
```
# echo 1 > /dev/watchdog
```

Disable the watchdog timer:

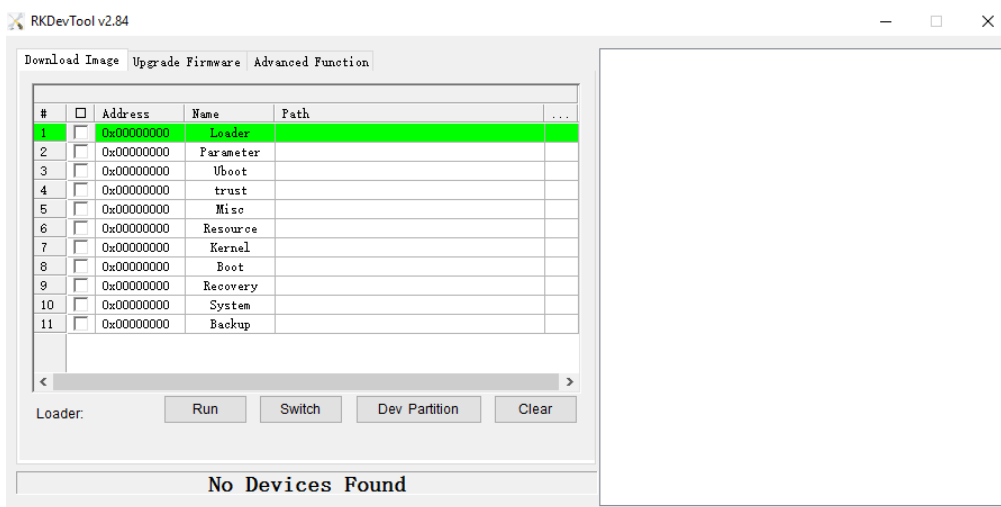
```
# echo V > /dev/watchdog
```

4.7 Firmware Upgrade in Windows Environment

1. Follow the steps in 3.2 to finish ADB setup on the host computer;
2. Unzip the release package and open the directory of the upgrade driver (\SW\downloadertools\windows\DriverAssitant_vxxx);
3. Right click the mouse and run the driver program **DriverInstall.exe** as administrator;
4. Click **Install Driver** and wait for the installation to proceed;

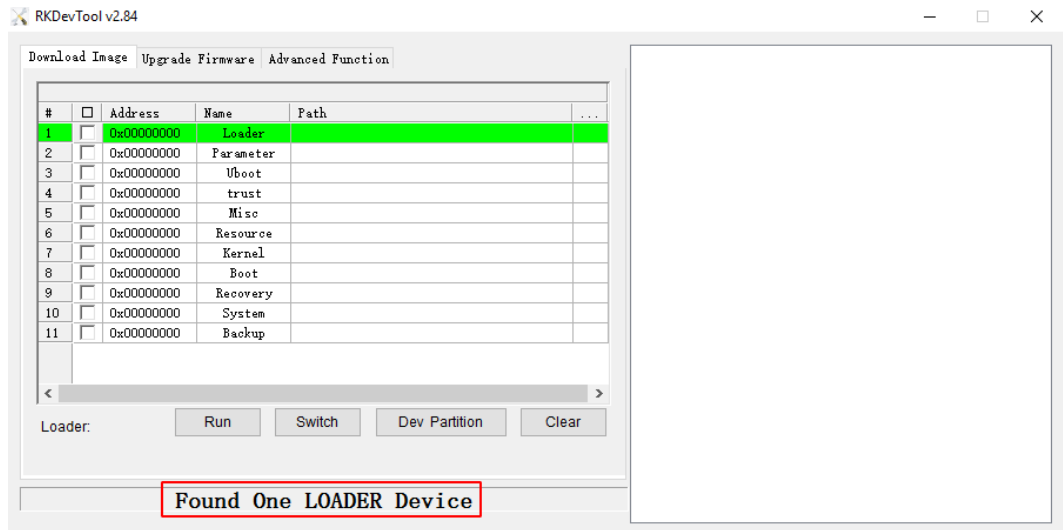


5. Open the directory of the upgrade tool (\SW\downloadertools\windows\ RKDevTool_ Release_vxxx);
6. Double click the driver for the upgrade tool **RKDevTool.exe** to open the upgrade window;

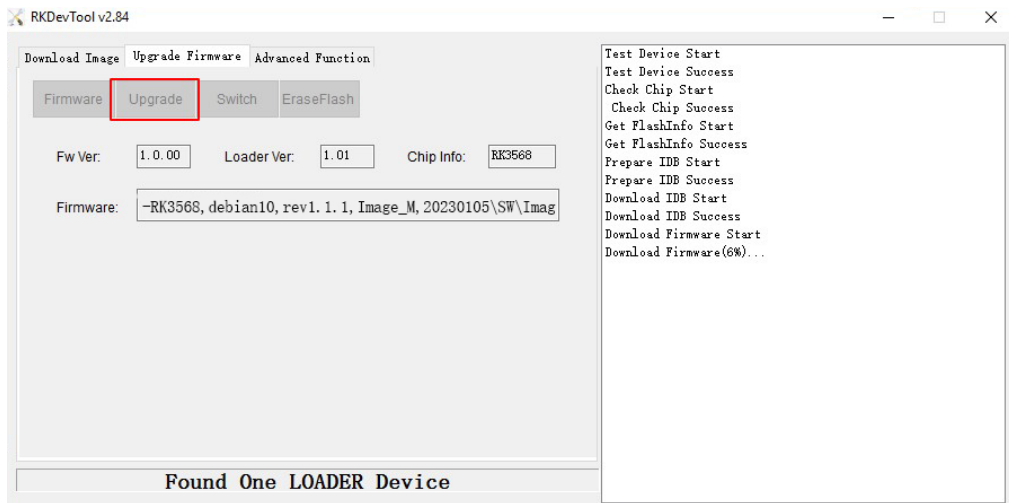


7. Connect VT-SBC-3568 to the Windows host with the USB Type-A to Type-C cable;
8. Press “Win + R” and input `cmd` in the dialog box to open the command prompt;
9. Input `adb devices` in the command prompt to check if the Board is connected to the Windows host;
10. Once the Board is identified by the Windows host, input `adb reboot loader` to reboot the device into the bootloader mode;

11. Then the upgrade window will prompt for the existence of a Loader device, indicating that the upgrade process is ready;



12. Click **Upgrade Firmware** > **Firmware** in the upgrade window;
13. Select the upgrade file (**update.img**) from the directory (\SW\Image) and click **Open**, and the firmware details will be automatically populated in the boxes;
14. Click the **Upgrade** button and the Board will start to download the image and upgrade the firmware automatically;



15. When the upgrade finishes, the Board will reboot automatically.


4.8 Firmware Upgrade in Ubuntu Environment

1. Connect VT-SBC-RK3568 to the Ubuntu host with the USB Type-A to Type-C cable;
2. Open a terminal and input `adb devices` to check if the Board is recognized by the host;
3. Run `adb shell` to open a shell on the Board;
4. Run `reboot loader` to reboot the Board to the bootloader mode;

```
root@vt-sbc-rk3568-gd32-ark:~# reboot loader
```

5. Copy the release package to the Ubuntu system or save the release package to a USB drive and mount the USB drive to the Ubuntu system;
6. Open the directory of the upgrade tool (named as “upgrade_tool” in the package) in the Ubuntu system (\SW\downloadetools\linux\Linux_Upgrade_Tool);
7. Right click the mouse in an empty space of the folder to open a terminal with the following command;

```
$ sudo ./upgrade_tool uf xxx/SW/update.img
```

 “xxx” is the name of the release package.

8. The system will start upgrading after the download finishes, and it will reboot automatically when the upgrade finishes.

CHAPTER 5 YOCTO PROJECT DEVELOPMENT

Yocto Project is an open-source collaboration project for Linux-based systems regardless of the hardware architecture. The project provides a flexible set of tools and a space where embedded developers can share technologies, software stacks, configurations, and best practices to create tailored Linux images for embedded and IoT devices.

This chapter will brief on the basics to build a Yocto project.

5.1 Build Environment

5.1.1 Prerequisites

- It is best to have a native Linux machine function as the development host. Since the Yocto project is not compatible with Windows Subsystem for Linux (WSL), you are advised not to use a build host that is running WSL. This chapter is based on Ubuntu 18.04.
- Reserve at least 50GB of free disk space for image building. 120GB is recommended.
- Required Git, tar, and Python Versions:
 - Git 1.8.3.1 or greater
 - tar 1.27 or greater
 - Python 3.4.0 or greater

Please refer to [Yocto Project Reference Manual \(System Requirements\)](#) for the supported Linux distributions and host requirements.

5.1.2 Dependency Installation

In order to complete a basic development or compilation task, you need install packages for the build host. For Ubuntu and Debian hosts, please install the following. Otherwise, please refer to [Yocto Project Reference Manual \(Required Packages for the Build Host\)](#).

```
$ sudo apt-get update
$ sudo apt-get install gawk wget git-core diffstat unzip texinfo gcc-multilib \
  build-essential chrpath socat cpio python3 python3-pip python3-pexpect \
  xz-utils debianutils iputils-ping python3-git python3-jinja2 libegl1-mesa libsdl1.2-dev \
  xterm
```

5.1.3 SDK installation

Run the installation script as follows:

```
$. /vtlinux-x86_64-toolchain-4.0.4.sh
VtLinux - (A Yocto Project Based Distro) SDK installer version 4.0.4
=====
Enter target directory for SDK (default: /opt/vtlinux): /opt/rk3568
You are about to install the SDK to "/opt/rk3568". Proceed [Y/n]? Y
```

```
[sudo] password for hey:
Extracting
  SDK.....
  .....
  .....done
Setting it up...done
SDK has been successfully set up and is ready to be used.
Each time you wish to use the SDK in a new shell session, you need to source the
environment setup script e.g.
$ ./opt/rk3568/environment-setup-armv8a-vtlinux-linux
```

5.1.4 Test and verification

Use the “source” command to update the current environment script and initialize the compilation environment:

```
$ source /opt/rk3568/environment-setup-armv8a-vtlinux-linux
```

Check the environment variables:

```
$ echo $ARCH
arm64
```

```
$ echo $CROSS_COMPILE
aarch64-vtlinux-linux-
```

Check the GCC version:

```
$ $CC --version
aarch64-vtlinux-linux-gcc (GCC) 11.3.0
Copyright (C) 2021 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

Check the SDK version:

```
$ echo $OECORE_SDK_VERSION
4.0.4
```

 If you get stuck at any of above steps, please re-install the SDK.

5.1.5 Compiling “Hello World”

```
$ cat hello.c
#include <stdio.h>
int main(void)
{
    printf("Hello World..\n");
    return 0;
}
$ CC hello.c -o hello
```

Copy the compiled program to the Board and run it:

```
$ ./hello
Hello World..
```

If the above steps go smoothly, the SDK environment is successfully built.

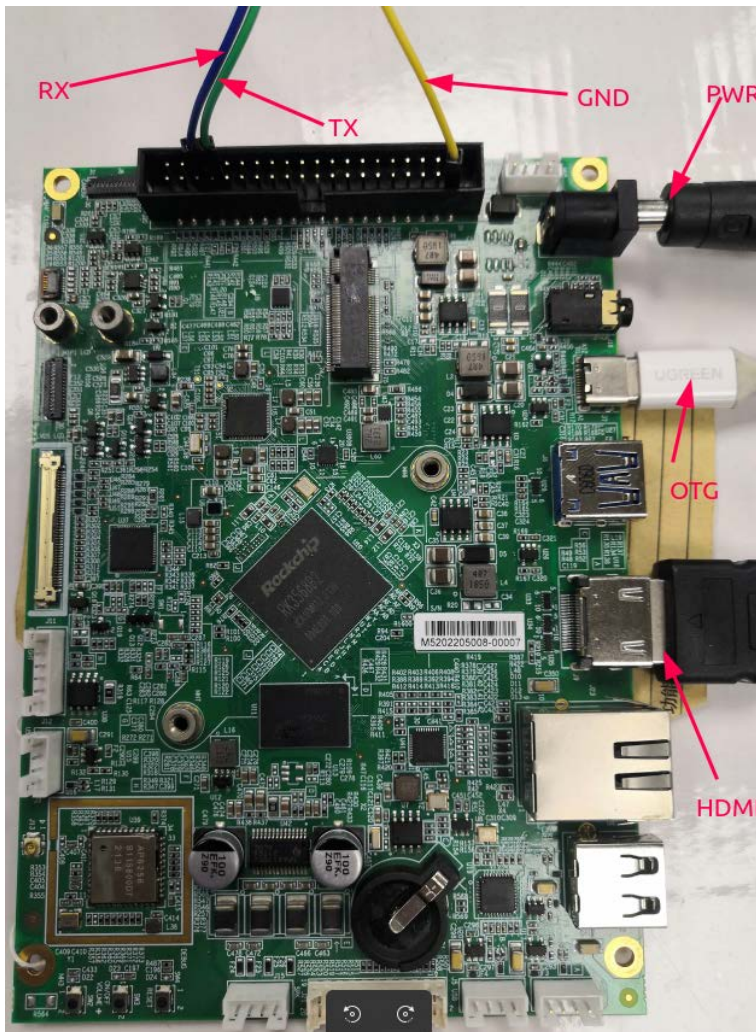
5.2 Image flashing

5.2.1 Prerequisites

- Software release package
- Device power supply: DC 12V 3A
- USB Type-C cable
- Serial port communication program, TTL serial cable
- A terminal host (Windows or Linux host with serial port communication program installed)

5.2.2 Hardware connection

Refer to the following figure to connect the Board with the host and peripheral.

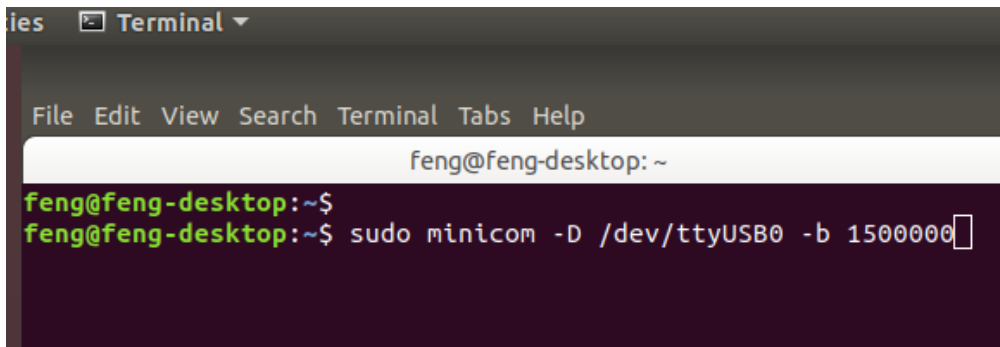


5.2.3 Preparation of the firmware package

Please unzip the software release package sent by Vantron.

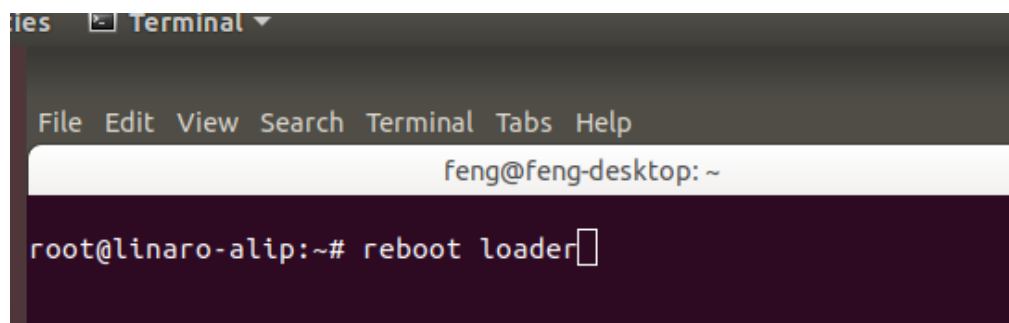
5.2.4 Image flashing

1. After connecting the debug serial port and power on the Board as indicated in 4.2.2, use the serial port communication program (e.g., minicom) to open the serial port with baud rate set to 1500000. The following example employs the Ubuntu system:



```
Terminal
File Edit View Search Terminal Tabs Help
feng@feng-desktop: ~
feng@feng-desktop:~$
feng@feng-desktop:~$ sudo minicom -D /dev/ttyUSB0 -b 1500000
```

2. Run “reboot loader” command so that the Board enters the Loader mode to prepare for the image flashing;



```
Terminal
File Edit View Search Terminal Tabs Help
feng@feng-desktop: ~
root@linaro-alip:~# reboot loader
```

3. The system will initialize for the flashing;


```
U-Boot 2017.09-gd0cde57-dirty #wangbingsen (Mar 01 2023 - 14:03:52 +0800)

Model: Rockchip RK3568 Evaluation Board
PreSerial: 2, raw, 0xfe660000
DRAM: 4 GiB
System: init
Relocation Offset: ed240000
Relocation fdt: eb9f87c0 - eb9fecd8
CR: M/C/I
Using default environment

dwmmc@fe2b0000: 1, dwmmc@fe2c0000: 2, sdhci@fe310000: 0
Bootdev(atags): mmc 0
MMC0: HS200, 200Mhz
PartType: EFI
DM: v1
boot mode: loader
Found DTB in boot part
DTB: rk-kernel.dtb
HASH(c): OK
I2c0 speed: 100000Hz
vsel-gpios- not found! Error: -2
vdd_cpu init 900000 uV
PMIC: RK8090 (on=0x40, off=0x00)
vdd_logic init 900000 uV
vdd_gpu init 900000 uV
vdd_npu init 900000 uV
io-domain: OK
INFO: ddr dmc_fsp already initialized in loader.
Could not find baseparameter partition
Model: Vantron RK3568 Board
Rockchip UB00T DRM driver version: v1.0.1
VOP have 2 active VP
vp0 have layer nr:3[1 3 5 ], primary plane: 5
vp1 have layer nr:3[0 2 4 ], primary plane: 4
vp2 have layer nr:0[], primary plane: 0
Using display timing dts
edp@fe0c0000: detailed mode clock 108000 kHz, flags[a]
  H: 1280 1372 1392 1484
  V: 1024 1044 1047 1067
bus_format: 100a
VOP update mode to: 1280x1024p68, type: eDP0 for VP1
VP1 set crtc_clock to 108000KHz
VOP VP1 enable Smart0[640x480->640x480@320x272] fmt[1] addr[0xedfe2000]
Link Training success!
final link rate = 0x0a, lane count = 0x02
hdmi@fe0a0000 disconnected
enter Rockusb!
RKUSB: LUN 0, dev 0, hwpart 0, sector 0x0, count 0x1d5a000
√
```

4. Copy the release package to the Ubuntu system or save the release package to a USB drive and mount the USB drive to the Ubuntu system;
5. Open the directory of the upgrade tool (“upgrade_tool”) in the Ubuntu system and right click the mouse in an empty space of the folder to open a terminal with the following command;

```
$ sudo upgrade_tool uf update-yocto-2023-2-10.img
```

 “update-yocto-xxx.img” is the name of the image.

6. The system will reboot after image flashing.

CHAPTER 6 DISPOSAL AND PRODUCT WARRANTY

6.1 Disposal

When the device comes to end of life, you are suggested to properly dispose of the device for the sake of the environment and safety.

Before you dispose of the device, please back up your data and erase it from the device.

It is recommended that the device is disassembled prior to disposal in conformity with local regulations. Please ensure that the abandoned batteries are disposed of according to local regulations on waste disposal. Do not throw batteries into fire or put in common waste canister as they are explosive. Products or product packages labeled with the sign of “explosive” should not be disposed of like household waste but delivered to specialized electrical & electronic waste recycling/disposal center.

Proper disposal of this sort of waste helps avoid harm and adverse effect upon surroundings and people’s health. Please contact local organizations or recycling/disposal center for more recycling/disposal methods of related products.

6.2 Warranty

Product warranty

VANTRON warrants to its CUSTOMER that the Product manufactured by VANTRON, or its subcontractors will conform strictly to the mutually agreed specifications and be free from defects in workmanship and materials (except that which is furnished by the CUSTOMER) upon shipment from VANTRON. VANTRON's obligation under this warranty is limited to replacing or repairing at its option of the Product which shall, within **24 months** after shipment, effective from invoice date, be returned to VANTRON's factory with transportation fee paid by the CUSTOMER and which shall, after examination, be disclosed to VANTRON's reasonable satisfaction to be thus defective. VANTRON shall bear the transportation fee for the shipment of the Product to the CUSTOMER.

Out-of-Warranty Repair

VANTRON will furnish the repair services for the Product which are out-of-warranty at VANTRON's then-prevailing rates for such services. At customer's request, VANTRON will provide components to the CUSTOMER for non-warranty repair. VANTRON will provide this service as long as the components are available in the market; and the CUSTOMER is requested to place a purchase order up front. Parts repaired will have an extended warranty of 3 months.

Returned Products

Any Product found to be defective and covered under warranty pursuant to Clause above, shall be returned to VANTRON only upon the CUSTOMER's receipt of and with reference to a VANTRON supplied Returned Materials Authorization (RMA) number. VANTRON shall supply a RMA, when required within three (3) working days of request by the CUSTOMER. VANTRON shall submit a new invoice to the CUSTOMER upon shipping of the returned products to the CUSTOMER. Prior to the return of any products by the CUSTOMER due to rejection or warranty defect, the CUSTOMER shall afford VANTRON the opportunity to inspect such products at the CUSTOMER's location and no Product so inspected shall be returned to VANTRON unless the cause for the rejection or defect is determined to be the responsibility of VANTRON. VANTRON shall in turn provide the CUSTOMER turnaround shipment on defective Product within **fourteen (14) working days** upon its receipt at VANTRON. If such turnaround cannot be provided by VANTRON due to causes beyond the control of VANTRON. VANTRON shall document such instances and notify the CUSTOMER immediately.