Portwell, Inc.
No. 242, Bo-Ai Street, Shu-Lin Dist., New Taipei City 238, Taiwan Tel:+886-77318888 Fax:+886-2-77319888 http://www.portwell.com.tw e-mail:info@portwell.com.tw

PJAI-1100 JP6

Linux H/W Access Guide

Revision 1.0.0

Aug. 5, 2025

Portwell Inc.

*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

No. 242, Bo-Ai Street, Shu-Lin Dist., New Taipei City 238, Taiwan Tel:+886-77318888 Fax:+886-2-77319888 http://www.portwell.com.tw e-mail:info@portwell.com.tw

Table of Contents

Revision History **3**
A. Introduction **4**
B. Hardware I/O **4**
C1. COM Port 4
C2. USB 6
C3. NVMe 7
C4. RTC 7
C5. Digital Input & Digital Output 8
C6. Ethernet 9
C7. M.2 WIFI Module 12
C8. M.2 Cellular Module 15
C9. CAN Bus 16
C10. Console Port 17

*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

No. 242, Bo-Ai Street, Shu-Lin Dist., New Taipei City 238, Taiwan Tel:+886-77318888 Fax:+886-2-77319888 http://www.portwell.com.tw e-mail:info@portwell.com.tw

Revision History

Rev.	Date	Author	Draft/Changes
V1.0.0	2025/08/05	Brian Wu	● First revision for following Portwell systems: ■ PJAI-1100-ON4G ■ PJAI-1100-ON8G ■ PJAI-1100(F)-OX8G ■ PJAI-1100(F)-OX16G

*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

A. Introduction

This document describes how to access the hardware I/O on PJAI-1100 in Linux OS.

B. Hardware I/O

This chapter list all supported hardware I/O and show how to access or test it in the Linux OS.

C1. COM Port

The table below lists the device mapping for each COM port.

System Label	Linux Device Name
COM1 (DB9 Male)	/dev/ttyCOM1
COM2 (DB9 Male)	/dev/ttyCOM2

Both COM1 and COM2 ports are versatile communication interfaces that can be configured to operate in one of three modes: RS232, RS422, or RS485. These modes represent different standards for serial communication, each with its own characteristics and use cases. By default, both COM1 and COM2 are set to RS232 mode.

To adjust the operating mode of COM1 and COM2, you can use the Linux **sysfs** interface. **Sysfs** is a virtual file system in Linux that provides a convenient way to interact with kernel objects and their attributes. The commands below illustrate how to change the mode of the COM ports through **sysfs**, allowing for flexible configuration to meet the specific requirements of your application.

```
# Set both COM1 and COM2 to RS232 mode
$ echo 4 | sudo tee /sys/devices/platform/com0_mode/mode
$ echo 4 | sudo tee /sys/devices/platform/com1_mode/mode

# Set both COM1 and COM2 to RS422 mode
$ echo 7 | sudo tee /sys/devices/platform/com0_mode/mode
$ echo 7 | sudo tee /sys/devices/platform/com1_mode/mode

# Set both COM1 and COM2 to RS485 mode
$ echo 5 | sudo tee /sys/devices/platform/com0_mode/mode
$ echo 5 | sudo tee /sys/devices/platform/com1_mode/mode
```


*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

The commands below illustrate how to verify the function via **linux-serial-test** tool.

```
# Prepare linux-serial-test tool
$ git clone --depth=1 https://github.com/cbrake/linux-serial-test
$ cd linux-serial-test
$ gcc linux-serial-test.c -o linux-serial-test

# Run TX/RX loopback test on COM1
$ sudo ./linux-serial-test -s -e -p /dev/ttyCOM1 -b 115200

# Run TX/RX loopback test on COM2
$ sudo ./linux-serial-test -s -e -p /dev/ttyCOM2 -b 115200
```


*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

C2. USB

There are three USB Type-A 3.2 ports on PJAI-1100.

You can plug some USB devices such as USB keyboard, USB mouse or USB stick, and then observed the kernel messages to check the functionality. The commands below show an example to monitor the kernel messages.

```
$ journalctl -k -f
# Connect a USB 3.0 stick on J11 Type-A port
...
... test-desktop kernel: usb 2-1.3: new SuperSpeed Gen 1 USB device
number 4 using tegra-xusb
... test-desktop kernel: usb 2-1.3: New USB device found, idVendor=0781,
idProduct=5597, bcdDevice= 1.00
... test-desktop kernel: usb 2-1.3: New USB device strings: Mfr=1,
Product=2, SerialNumber=3
... test-desktop kernel: usb 2-1.3: Product:  SanDisk 3.2Gen1
... test-desktop kernel: usb 2-1.3: Manufacturer:  USB
... test-desktop kernel: usb 2-1.3: SerialNumber:
040120d105e140df4eb09da7ff5f923b44a89e3c4f32c7c3a1f91e691380be8b96b
d000000000000000000029f4637a00913718975581071fae4964
... test-desktop kernel: usb-storage 2-1.3:1.0: USB Mass Storage device
detected
... test-desktop kernel: scsi host0: usb-storage 2-1.3:1.0
...
```


*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

C3. NVMe

The table below lists the device mapping.

Hardware	Linux Device Name
M.2 NVMe	/dev/nvme0n1

The NVMe drive serves as the primary storage medium for the Nvidia Linux system. It is specifically utilized to store critical components such as the Linux kernel image, the Linux kernel device tree, and the root file systems. In addition to these fundamental elements, the NVMe drive will typically be your main storage for all applications and user data.

C4. RTC

The table below lists the device mapping.

Hardware	Linux Device Name
Battery Backup RTC (PSEQ_RTC)	/dev/rtc0
Secondary RTC (Orin_RTC)	/dev/rtc1

The device **/dev/rtc0** is primary system RTC because it has battery backup. It means the RTC will continue to count even if unplug the power cable. The device **/dev/rtc1** has no battery backup.

You can use the "hwclock" and "date" commands to verify its function.
The commands below show some examples to use the "hwclock" and "date" commands.

```
# Set RTC and read back to verify it

# Set system to 2024/06/07 11:30:15
# Format for date command: MMDDhhmmYYYY.ss
$ sudo date 060711302024.15
# Set /dev/rtc0 from system time
$ sudo hwclock –w --rtc /dev/rtc0
# Read current date time from /dev/rtc0
$ sudo hwclock -r --rtc /dev/rtc0
2024-06-07 11:30:16.547162+00:00
```


*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

C5. Digital Input & Digital Output

The digital input and digital output (DIDO) on this system are both **DRY** contacts. This means that these contacts do not provide their own power and therefore require an external power source to operate. Specifically, a **DRY** contact is an electrical contact that does not supply voltage or current by itself; it merely acts as a switch that opens or closes a circuit.

The DIDO driver follows Linux kernel's **gpiolib** framework. The table below lists the device mapping.

PIN#	PIN Name	Direction	Linux 'gpiochip' Name	Linux 'gpiochip' Line#
10	DO_0	OUT	xr_gpio0	0
8	DO_1	OUT	xr_gpio0	1
6	DO_2	OUT	xr_gpio0	2
4	DO_3	OUT	xr_gpio0	3
9	DI_0	IN	xr_gpio0	4
7	DI_1	IN	xr_gpio0	5
5	DI_2	IN	xr_gpio0	6
3	DI_3	IN	xr_gpio0	7

The commands below show some examples to use the tools in **gpiod** to access the DIDO.

```
# Install gpiod
$ sudo apt install gpiod
# Use 'gpiodetect' to find out the gpiochip number for xr_gpio0
$ sudo gpiodetect
gpiochip0 [tegra234-gpio] (164 lines)
gpiochip1 [tegra234-gpio-aon] (32 lines)
gpiochip2 [xr_gpio0] (10 lines)
# Set DO_1 to HIGH
$ sudo gpioset gpiochip2 1=1
# Get DI_3 status
$ sudo gpioget 7

# If you are interesting in writing your own tool, you can reference:
# https://github.com/brgl/libgpiod/tree/v1.6.x
```


*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

C6. Ethernet

The table below lists the device mapping for each ethernet port.

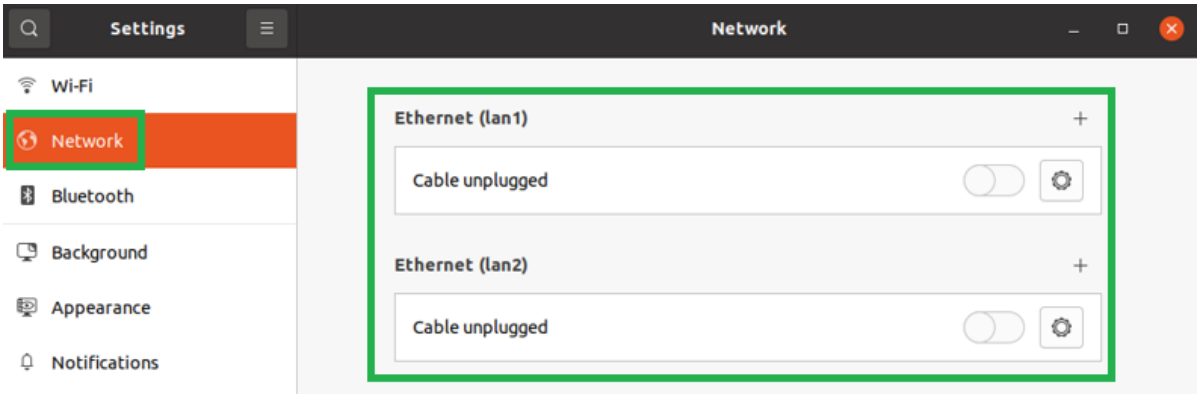
System Label	Linux Device Name
LAN1	elan1
LAN2	elan2

These network devices are default managed by Ubuntu's **NetworkManager** service. When you connect an RJ45 LAN cable, the **NetworkManager** automatically attempts to obtain an IP address via DHCP (Dynamic Host Configuration Protocol). The **NetworkManager** service is responsible for handling network connections, including both wired and wireless interfaces, and it ensures that the device is configured correctly for network access.

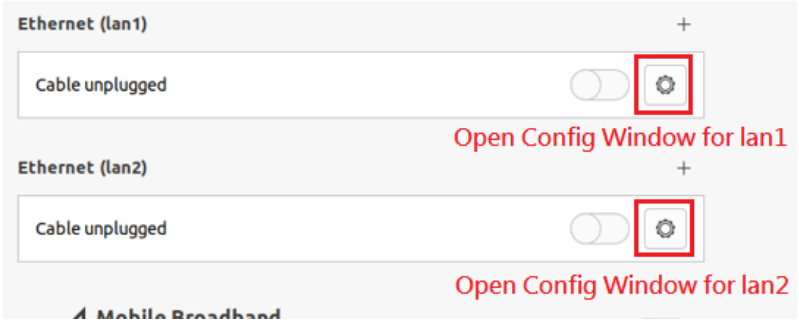
To configure it, you can choose to use GUI interface or command line tool "nmcli".

[Configure from GUI]

Open **Settings -> Network**, you will see two ethernet devices on the window.



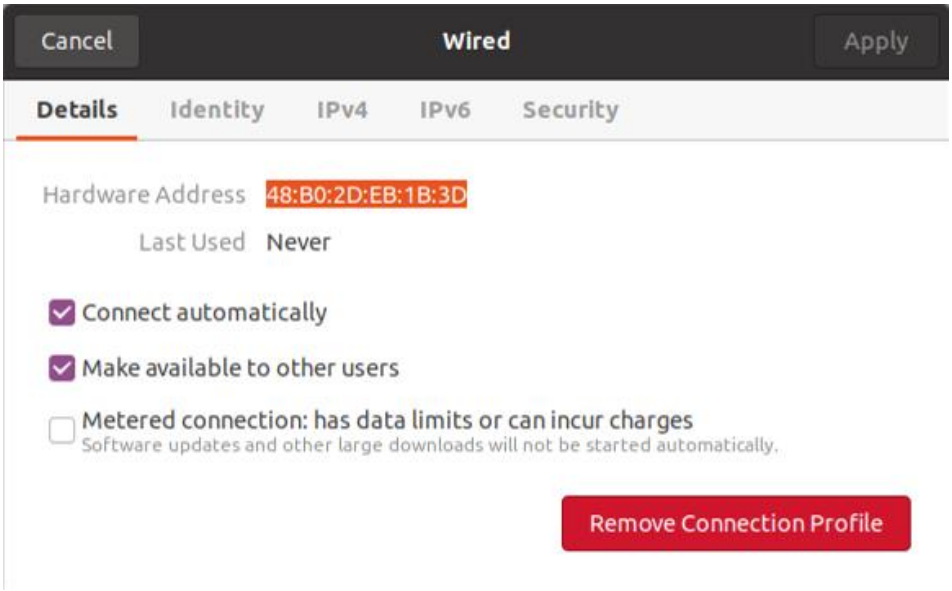
You can click the button on the right of ethernet device to open the configuration window.



*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

Below picture shows the configuration windows for 'lan1'. Now you can configure it based on your requirements.



[Configure from Command Line Tool 'nmcli']

"nmcli" is a command-line interface for **NetworkManager**, used for managing network connections. It provides a convenient way to configure and manage network settings on Ubuntu. Below are some common "nmcli" commands and their usage.

Some basic usages:

```
# View all network devices and their status
$ nmcli device status

# View all connections
$ nmcli connection show

# Enable or disable a network device
$ sudo nmcli device connect <device_name>
$ sudo nmcli device disconnect <device_name>
```


*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

No. 242, Bo-Ai Street, Shu-Lin Dist., New Taipei City 238, Taiwan Tel:+886-77318888 Fax:+886-2-77319888 http://www.portwell.com.tw e-mail:info@portwell.com.tw

Managing connections:

```
# Create a new connection with DHCPv4 enabled
$ sudo nmcli con add type ethernet ifname <interface_name> \
con-name <connection_name> ipv4.method auto

# Create a new connection with static IPv4 address
$ sudo nmcli connection add type ethernet ifname <interface_name> \
con-name <connection_name> ip4 <ip_address>/<prefix_length> \
gw4 <gateway>
# Example
$ sudo nmcli connection add type ethernet ifname elan1 \
con-name "StaticLan1" \
ip4 192.168.2.111/24 gw4 192.168.2.1

# Bring up a connection
# (Each ethernet device can have only one active connection at a time)
$ sudo nmcli connection up <connection_name>

# Bring down a connection
$ sudo nmcli connection down <connection_name>
```


*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

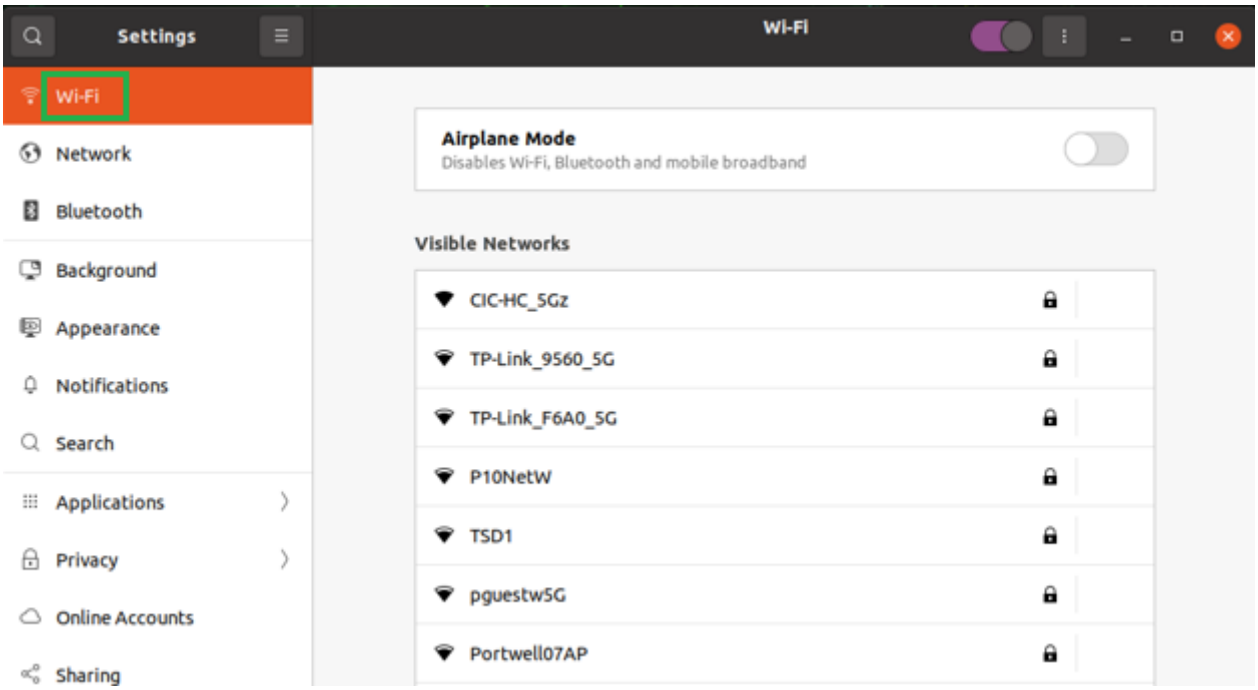
C7. M.2 WIFI Module

This system support **AzureWave AW-CB375NF** M.2 WIFI module. It is default managed by Ubuntu's **NetworkManager** service.

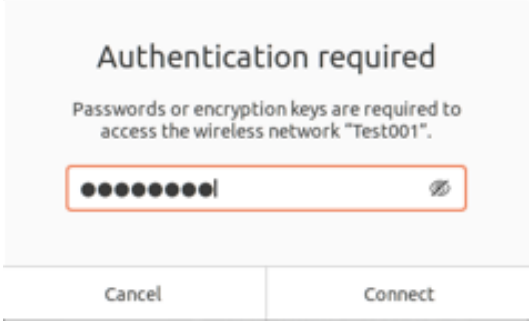
To configure it, you can choose to use GUI interface or command line tool "nmcli".

[Configure from GUI]

Open **Settings -> Wi-Fi**, you can see all available WIFI SSID on the window.



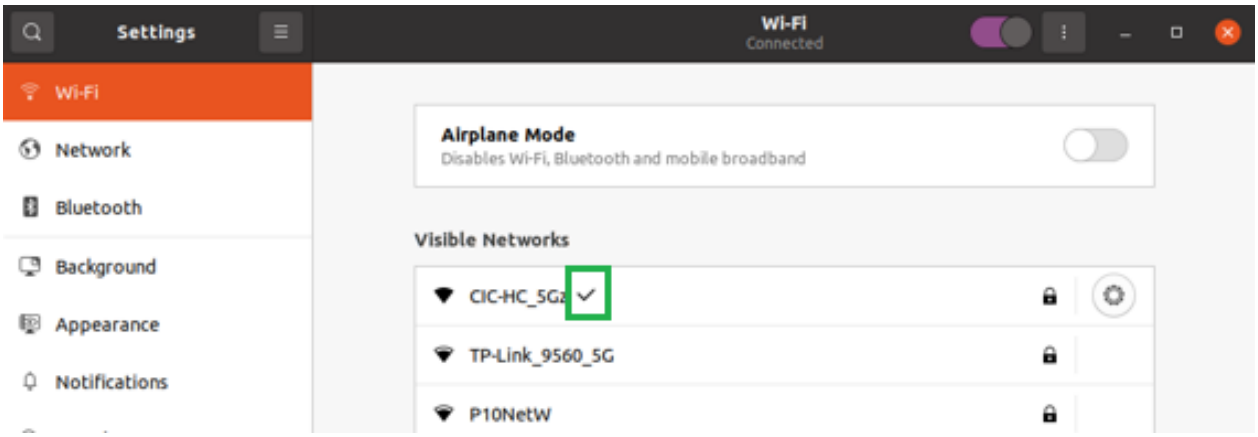
You can choose and click the SSID. It will try to connect to the SSID. You will see following window if it required authentication. Please input the password for the SSID.



*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

If the connection is successful, you will see a checkmark symbol on the SSID you selected, as shown in the picture below.



You can also check following symbol on the right of top panel.

Now you can use ping command to verify network connection.

```
# Check network
$ ping 8.8.8.8
```


*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

[Configure from Command Line Tool 'nmcli']

Below are some example usages.

```
# Scan for available Wi-Fi networks
$ nmcli device wifi list
IN-USE  BSSID             SSID           ...     SECURITY
        C6:EE:92:09:FF:74  Test001               WPA2
        74:DA:88:1C:95:5E  TP-Link_9560_5G       WPA2
...

# Connect to a WIFI network
$ sudo nmcli device wifi connect <SSID> password <password>

# Create a new Wi-Fi connection
$ sudo nmcli connection add type wifi ifname wlan0 \
con-name <connection_name> ssid <SSID>
$ sudo nmcli connection modify <connection_name> \
wifi-sec.key-mgmt wpa-psk
$ sudo nmcli connection modify <connection_name> \
wifi-sec.psk <password>

# Bring up a connection
$ sudo nmcli connection up <connection_name>

# Bring down a connection
$ sudo nmcli connection down <connection_name>

# Example (SSID: Test001, Password: 12345678)
$ sudo nmcli connection add type wifi ifname wlan0 \
con-name "MyWIFI" ssid "Test001"
$ sudo nmcli connection modify "MyWIFI" wifi-sec.key-mgmt wpa-psk
$ sudo nmcli connection modify "MyWIFI" wifi-sec.psk "12345678"
$ sudo nmcli connection up "MyWIFI"
```


*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

C8. M.2 Cellular Module

This system supports **Quectel RM520N** M.2 5G module.

[Quectel QM520N M.2 5G Module]

The commands below show how to use **Quectel QConnectManager** tool to connect the internet.

```
# The ModemManager does not support this 5G module
# Stop and disable ModemManager
$ sudo systemctl stop ModemManager
$ sudo systemctl disable ModemManager

# Get network connection via Quectel QConnectManager
$ sudo quectel-CM -p <pin_code> &
...
[06-07_11:56:36:318] requestGetSIMStatus SIMStatus: SIM_PIN
...
[06-07_11:56:56:414] requestRegistrationState2 MCC: 466, MNC: 92, PS:
Attached, DataCap: 5G_NSA
[06-07_11:56:56:478] ifconfig wwan0 up
[06-07_11:56:56:493] busybox udhcpc -f -n -q -t 5 -i wwan0
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 25.29.177.5
udhcpc: lease of 25.29.177.5 obtained, lease time 7200

# Setup the DNS
$ echo "nameserver 8.8.8.8" | sudo tee /etc/resolv.conf
# Check network
$ ping -I wwan0 8.8.8.8
```


*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

C9. CAN Bus

The following table lists the device mapping for each CAN bus port:

Hardware Connector Name	Linux Device Name
CAN	can0

The commands below show how to configurate CAN bus.

```
# Set link down so that we could configure it
$ sudo ip link set can0 down
# Set bit rate to 125kbps
$ sudo link set can0 up type can bitrate 125000 fd off

# For CAN FD, you can use following one:
# $ sudo ip link set can0 type can bitrate 500000 dbitrate 2000000 fd
on

# Set link up to enable can0
$ sudo ip link set can0 up

# Check current setting
$ sudo ip -details link show can0
```


*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

C10. Console Port

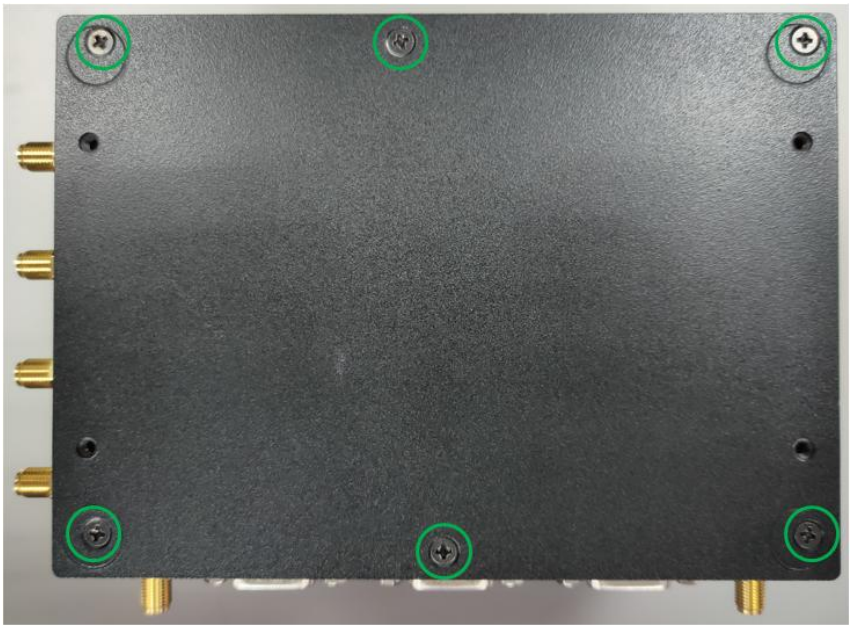
The system provides a RS-232 debug console port for receiving debug messages from the bootloader firmware and Linux kernel.

[Port Location]

The debug console shares the same physical connector as COM2 (DB9 interface).

[Configuration Steps]

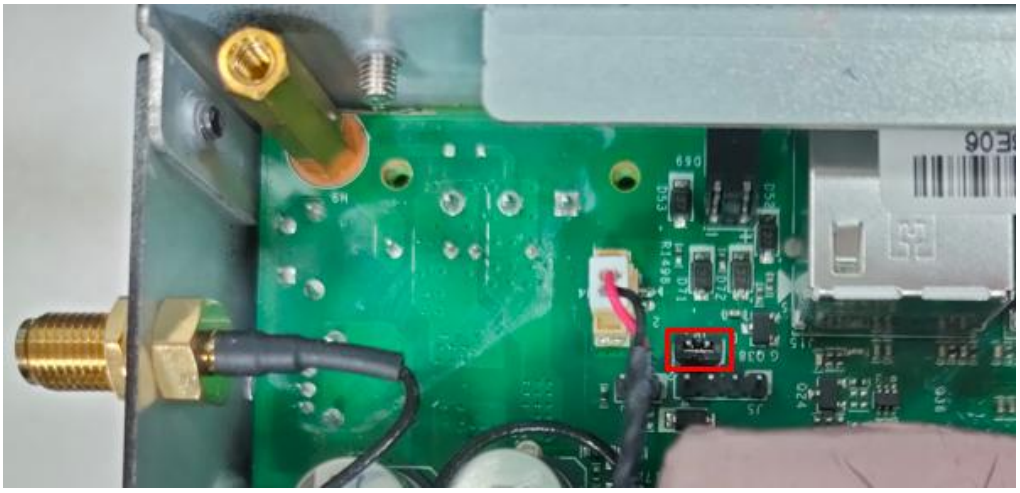
Remove the **back cover**.



*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

Locate jumper JP3 and remove its shunt to switch COM2 to console mode.



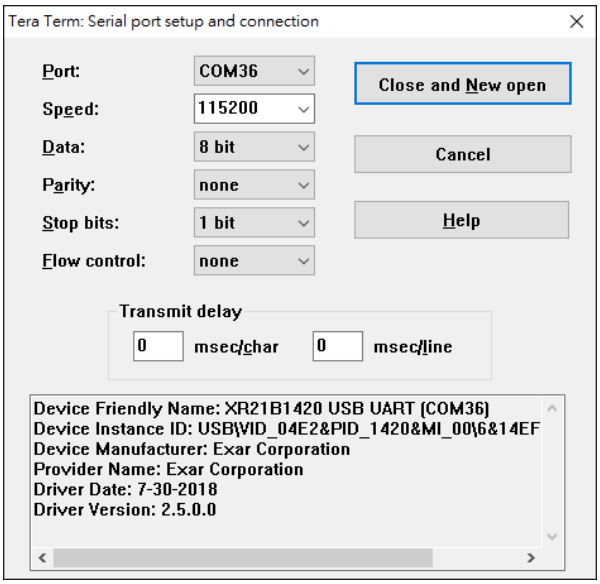
[Connection Options]

Use a standard DB9 serial cable to connect the PJAI-1100 to a host PC. If your PC lacks a DB9 port, use a USB-to-serial adapter.

[Software Setup]

On the host PC, open a terminal program (e.g., **PuTTY** or **Tera Term**) and configure the terminal with the correct COM port (e.g., 115200 baud, 8N1, no flow control).

Example **Tera Term** settings:



*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.

[Serial Console Parameters]

Baud rate: 115200

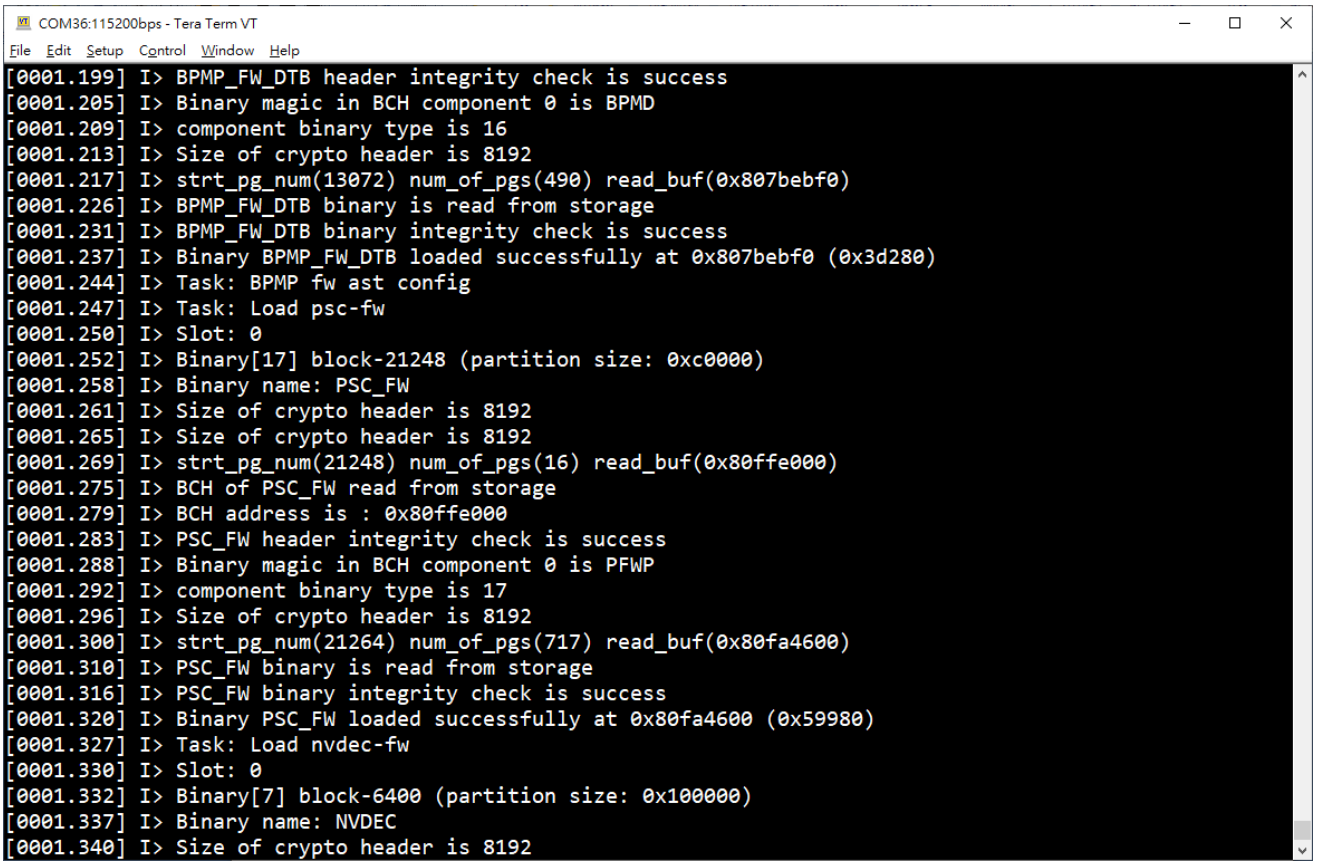
Data Bits: 8 bit

Parity Bit: None

Stop Bits: 1 bit

Flow Control: None

The picture below shows part of the debug messages of bootloader.



*Third-party brands and names are the property of their respective owners.

Copyright@ 2025 Portwell,Inc. All rights reserved
All information contained in this document is proprietary and confidential. Unauthorized use, duplication, modification or disclosure of information to a third-party by any means without prior consent of Portwell,Inc. is prohibited.