Manuals+ — User Manuals Simplified.

RAK7391 Modular IoT Platform For Multi Radio User Guide

Contents

RAK7391 Modular IoT Platform For Multi-Radio

Specifications

- Power Input: 10-28 VDC
- Supported PoE: IEEE 802.3 at/bt active PoE
- Ethernet Ports: 1 Gb Ethernet Port (eth0) and 2.5 Gb Ethernet port (eth1)

Product Usage Instructions

Powering on the Device

The RAK7391 can be powered via 10-28 VDC using either the Barrel Jack or Phoenix Connector. Do not power both simultaneously.

Powering via PoE

To use PoE, install a specialized PoE HAT module and connect it to the 1 GB Ethernet port.

Flashing the OS

Use RAKPiOS as the recommended OS for the RAK7391. Refer to RAKPiOS documentation for flashing and booting instructions.

Network Connectivity

Manage connectivity with the NetworkManager tool suite. Use NetworkManager to configure Ethernet, Wi-Fi, and Cellular interfaces. Ensure proper permissions for the NetworkManager configuration file.

Setting Up Ethernet

Connect to the Ethernet ports (eth0 or eth1). DHCP should work without additional configuration. To set a static IP, create a connection file in the specified folder with the required details.

FAQ

- **Can I use a general Raspberry Pi 40-pin PoE HAT with RAK7391?**
  - No, the PoE board for RAK7391 is specialized and cannot be substituted with a general Raspberry Pi PoE HAT.
- **How do I change the IP address to a static one for an Ethernet interface?**
  - To assign a static IP address, create a new connection file in the designated folder or use the nmcli command-line tool with the appropriate commands.

Quick Start Guide

The **RAK7391** can be powered via 10-28 VDC. There is one Barrel Jack and one Phoenix Connector, so the user has two options. Make sure you power either using one or the other not both at the same time.
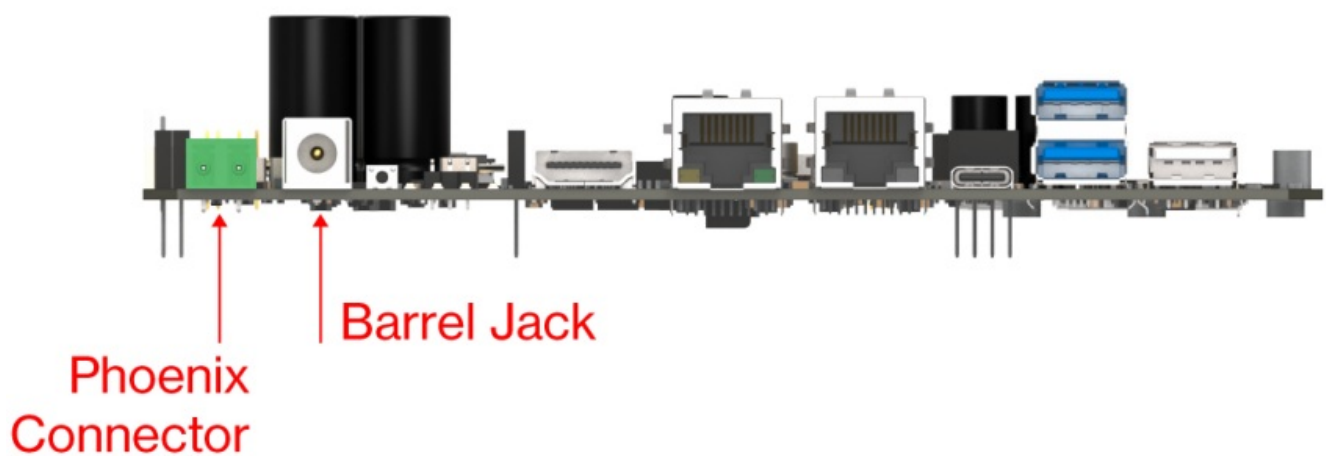

Figure 1: DC power connectors

The device also supports IEEE 802.3 at/bt active PoE in combination with a specialized PoE HAT module. Mount the PoE Hat module then you can power the board through the 1 GB Ethernet port.

NOTE

The PoE board is specialized. A general Raspberry Pi 40-pin PoE HAT cannot be used
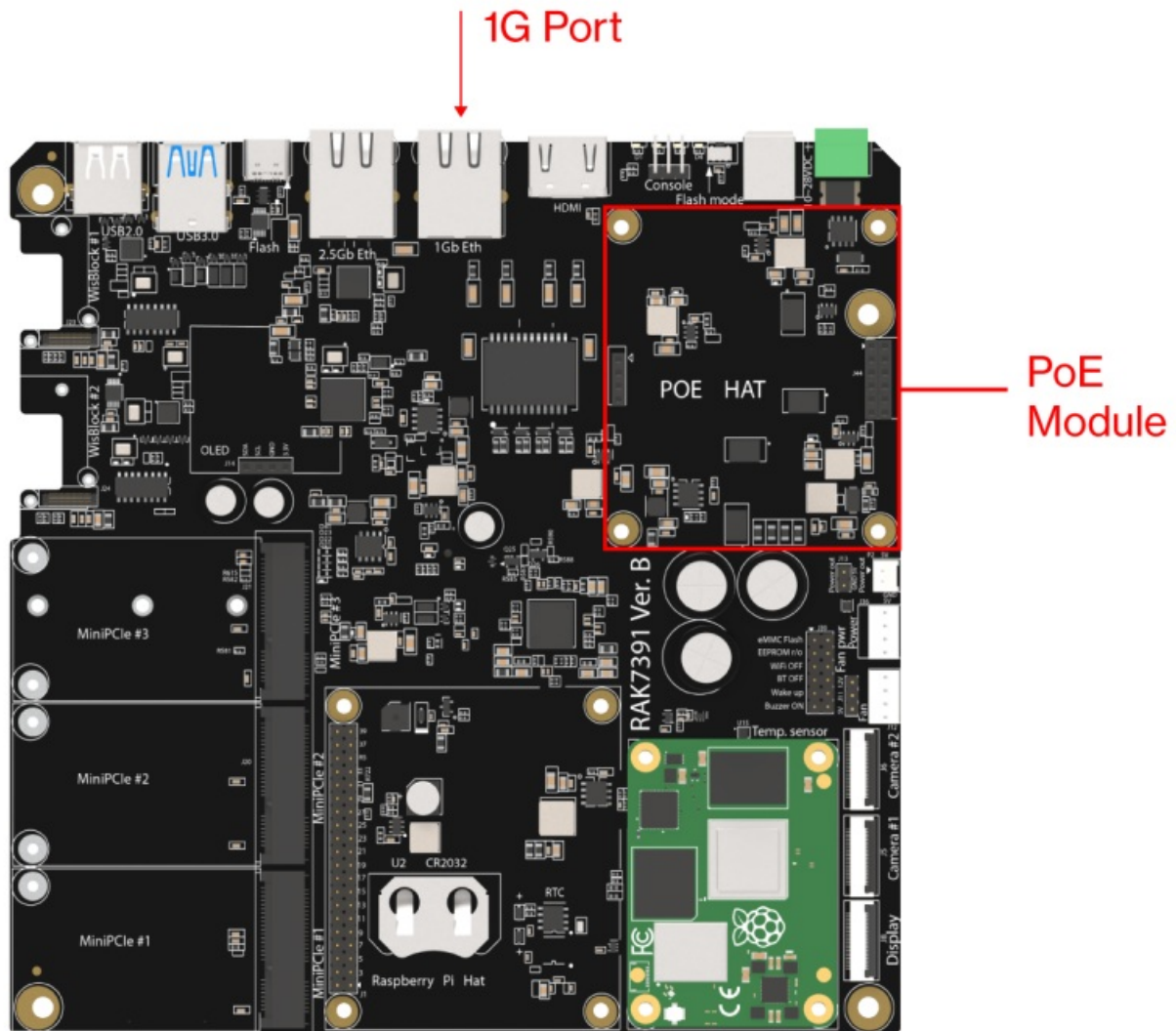
Figure 2: PoE HAT socket

Figure 3: PoE provisioning via the 1-GB port (PoE HAT installed)

Flashing the OS

RAK7391 recommended OS is RAKPiOS. Refer to the RAKPiOS documentation for instructions on flashing and booting the OS.

Network Connectivity

For the RAK7391, connectivity management is done via the **NetworkManager** tool suite. NetworkManager allows for flexibility when configuring the Ethernet, Wi-Fi, and Cellular interfaces. It is used to create, display, edit, delete, activate, and deactivate network connections via configuration files. One can also use the command-line tool **nmcli** for controlling NetworkManager.

Ethernet

There are two Ethernet ports on the gateway: one 1 Gb Ethernet Port (eth0) and one 2.5 Gb Ethernet port (eth1) . Once you connect to one of them, the Ethernet DHCP should work without requiring any specific configuration for Network Manager (you can grab the assigned address either via your router interface or from the OLED screen if you have it installed).
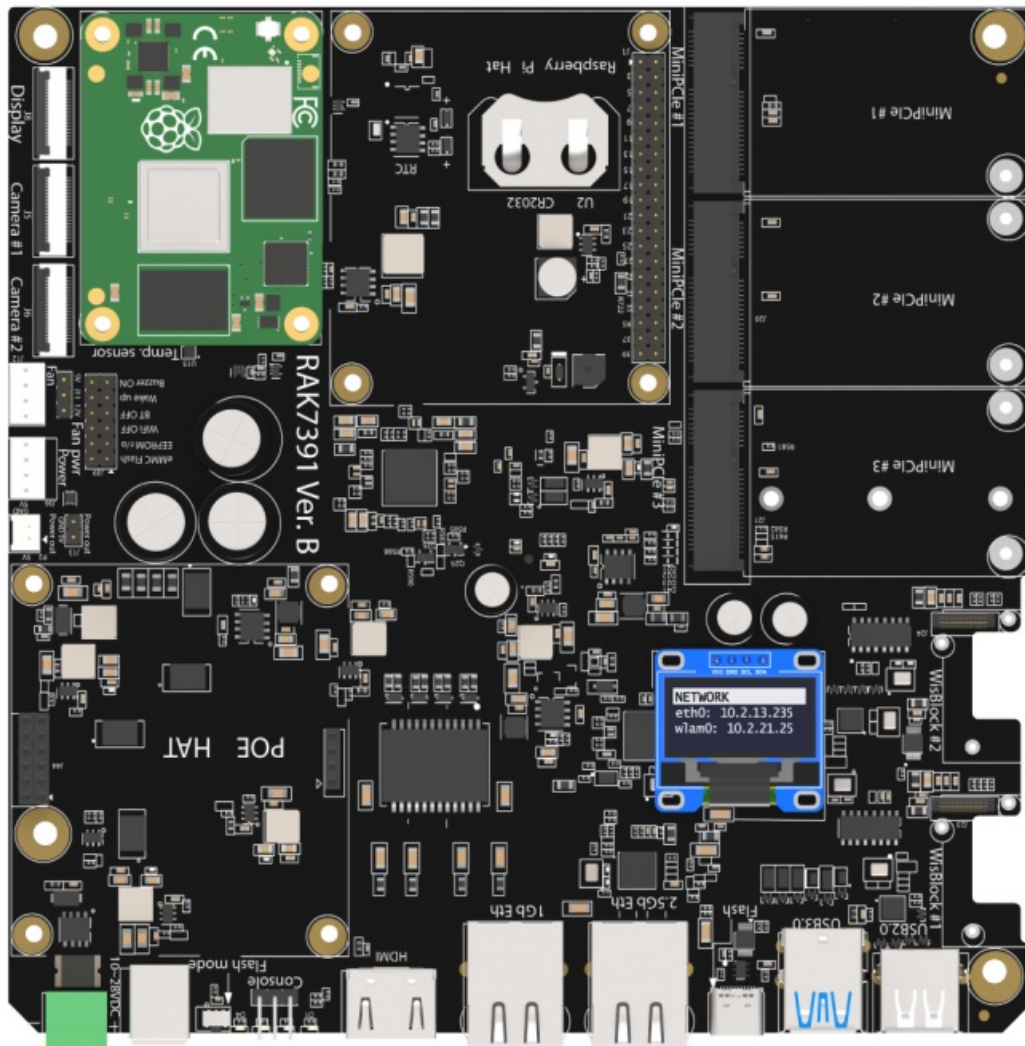
Figure 4: IP address listed on the OLED screen

If you want to change the IP to a static one for one of the Ethernet interfaces for example eth0 you need to create a new connection file named ethernet-eth0.nmconnection in the folder /etc/NetworkManager/system-connections/

The file should have the following contents:

```sh
[connection]
id=ethernet-eth0
type=ethernet
interface-name=eth0

[ipv4]
address1=192.168.1.111/24,192.168.1.1
dns=8.8.8.8
method=manual

[ipv6]
addr-gen-mode=stable-privacy
method=auto
```

The important bits to take note of here are:

- interface-name=eth0 – indicates which network interface to use, (eth0) in this example.
- address1=192.168.1.111/24,192.168.1.1 – shows which the assigned static IP address assigned to the interface is and which gateway is to be used (192.168.1.111) and the network gateway IP address 192.168.1.1 .
- dns=8.8.8.8 – indicates which Domain Name Server (DNS) is to be used, Google's in this example.

If one wants to do the above static IP setting, but using the command-line tool nmcli, execute the following command:

```sh
sudo nmcli con add type ethernet autoconnect yes ifname eth0 ipv4.addr 192.168.1.111/24 ipv4.gate
```

WiFi

WiFi 5

If the CM4 you have decided to use has built-in Wi-Fi, speeds up to Wi-Fi 5, as it supports IEEE 802.11 b/g/n/ac. Connecting is done in a similar way to how a manual Ethernet connection is set up. For example, if you want to connect to a hotspot named RAK , you should add a new connection file named RAK.nmconnection in the folder /etc/NetworkManager/system-connections/ .

The file should have the following contents:

```sh
[connection]
id=RAK
type=wifi
interface-name=wlan0

[wifi]
ssid=RAK

[wifi-security]
auth-alg=open
key-mgmt=wpa-psk
psk=rak20140629

[ipv4]
method=auto
route-metric=100

[ipv6]
addr-gen-mode=stable-privacy
method=auto
```

The configuration needs to be reloaded for the new settings to take effect via the following series of commands:

```sh
sudo chown -R root:root /etc/NetworkManager/system-connections/RAK.nmconnection
sudo chmod -R 600 /etc/NetworkManager/system-connections/RAK.nmconnection
sudo nmcli con reload
sudo nmcli con up RAK
```

The equivalent of the above with the command-line tool nmcli would be:

```sh
sudo nmcli device wifi connect RAK if name wlan0 password rak20220922
```

WiFi 6

The RAK7391 also supports Wi-Fi 6 modules through the mPCIe interface. The OS image comes with preinstalled firmware for several of Intel's Wi-Fi 6 card models (AX200, AX201 & AX210) including the default drivers. For example, one can use the **WiFi6E Intel AX210** module (tested already) as a Plug & Play device just by connecting it to the MiniPCie #3 on the RAK7391. The wireless interface name should be wlan1 in the RAKPiOS. To connect to a Wi-Fi 6 hotspot, the same procedure is used as for Wi-Fi 5, however, the only difference is that the interface name should be changed from wlan0 to wlan1 .

Cellular

RAK7391supports Cellular connectivity when used together with **RAK8213** , which is a multi-band LTE Cat M1/Cat Nb1/EGPRS module based on the Quectel BG96 with a standard PCI Express® form factor (mPCIe).

Hardware Setup

You can insert the RAK8213 into the mPCIe #2 or mPCIe #3 on the RAK7391 board, you also need a SIM card that supports at least one of the following: LTE Cat M1/Cat Nb1/EGPRS. Insert the SIM card into the SIM card slot SIM #2 or SIM #3 , both of which are located on the underside of the board.
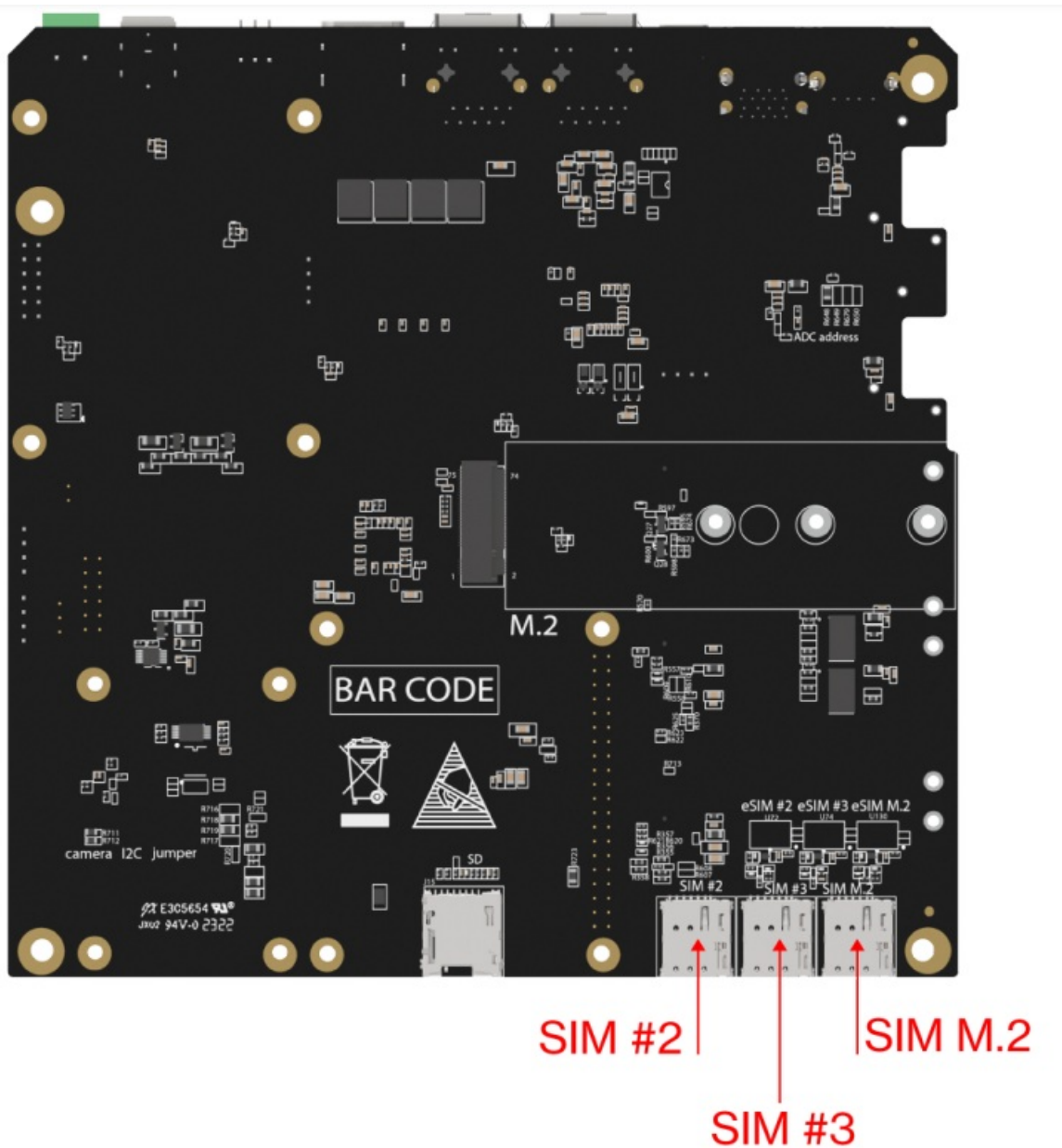
Figure 5: SIM slots

Cellular Configuration

Before you create a 4G connection file, you need to do some configuration to register the SIM card to the network provider. This requires connecting to the BG96 via the serial port and sending over a few AT commands

For this example, you are going to configure the BG96 with the serial communication tool minicom .

1. To start, execute the following command:

```
sudo minicom -D /dev/ttyUSB3
```

2. Make sure to turn the local Echo so you can input the command with the Ctrl+A, followed by pressing Z to get to the Command summary.
3. Press E to turn the local echo on/off.
   The following code block is a summary of the commands you need to execute in the same order. For detailed

descriptions of these commands, refer to this **AT command manual** from Quectel

```sh
at+cfun=0    //Minimum functionality
at+qcfg="nwscanmode",0,1 //set scanmode automatic
at+qcfg="nwscanseq",010203,1  // set searching sequence:GSM -> LTE CAT M1 -> LTE CAT NB1
at+qcfg="iotopmode",1,1
at+qcfg="servicedomain",1,1
at+qcfg="roamservice",255,1
at+qcfg="band",f,400A0E189F,A0E189F,1 //search all band
at+cpsms=0
at+cfun=1  //Full functionality
at+cops=?  //Lists the available network providers.
at+cops=1,2,"46000",0  //select and register the network providers
at+cops?  //Queries the connected web server information.
```

Connection File

Create a cellular connection file named gsm. nmconnection in the folder /etc/NetworkManager/systemconnections/

- The file should have the following contents:

```sh
[connection]
id=gsm
uuid=55e9c95e-5947-4dad-9a92-9d409449ba4d
type=gsm
interface-name=cdc-wdm0
permissions=

[gsm]
apn=iot.1nce.net

[ipv4]
dns-search=
method=auto

[ipv6]
addr-gen-mode=stable-privacy
dns-search=
method=auto

[proxy]
```
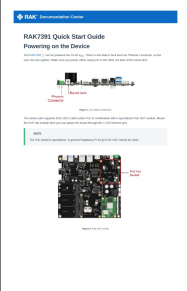
The important bits to take note of here are apn=cmnet which indicates the APN of your sim card network provider, then reboot your board and the wwan0 will get an IP address and it is also ready for the internet connection.

> **📝 NOTE**
>
> - `interface-name=cdc-wdm0` - this indicates the interface used.
> - `uuid=55e9c95e-5947-4dad-9a92-9d409449ba4d` - the UUID, should be provided with the SIM card.
> - `apn=iot.1nce.net` - every provider has a specific one, you need to obtain it via their online portal, for example, 1nce ⬀ is used, which has the APN `iot.1nce.net` .

After creating the file reboot/power cycle the gateway for the changes to take effect. This will result in the wwan0 interface getting an IP address (if the connection has been established successfully), which you can check on the OLED display.

Documents / Resources

RAK RAK7391 Modular IoT Platform For Multi Radio [pdf] User Guide
RAK7391 Modular IoT Platform For Multi Radio, RAK7391, Modular IoT Platform For Multi Radio, Platform For Multi Radio, For Multi Radio, Multi Radio, Radio

References

- 🅣 **Leading IoT Connectivity & Software Provider | 1NCE**
- 🅡 **RAKwireless Documentation Center**
- 🅡 **RAK8213 WisLink Cellular mPCIe | RAKwireless Documentation Center**
- 🅐 **nmcli(1) — Arch manual pages**
- Ⓜ **NetworkManager**
- 🅡 **RAK7391 Compute Module 4 (CM4) Carrier Board**
- 🌐 **Getting Started With Minicom - Getting Started With Minicom**
- **User Manual**