

Raspberry PI CM4 Based Edge computer



Revision History

Revision	Date	Changes
1.0	01-05-2021	Created



Contents

1. Introduction.....	4
1.1 Features.....	4
1.2 Interfaces.....	5
1.3 Block Diagram.....	7
2. Installation.....	8
2.1 Mounting.....	8
2.2 Connectors and Interfaces.....	9
2.2.1 Power supply.....	9
2.2.2 Serial Port (RS232 and RS485).....	9
2.2.3 DI&DO.....	10
2.2.4 HDMI.....	11
2.2.5 Ethernet.....	11
2.2.6 USB HOST.....	12
2.2.7 Console(USB typeC).....	12
2.2.8 LED.....	12
2.2.9 SMA Connector.....	
2.2.10 NANO SIM card slot.....	14
2.2.11 Mini-PCIe.....	15
2.2.12 M.2.....	16
3. Drivers and Programming Interfaces.....	17
3.1 LED.....	17
3.2 Serial Port (RS232 and RS485).....	18
3.3 Cellular over Mini-PCIe.....	18
3.4 WDT.....	21
3.4.1 Block Diagram of WDT.....	21
3.4.2 How it works.....	21
3.5 RTC.....	21
3.5.1.....	21
3.5.2.....	22
3.10 UPS for safe shut down.....	22
4. Electrical specifications.....	24
4.1 Power consumption.....	24
4.2 UPS.....	24
5. Mechanical Drawings.....	24



1. Introduction

EdgeBox-RPI4 is a rugged fanless Edge Computing Controller with Raspberry Pi Computer Module 4(CM4) for harsh industry environment. It can be used to connect the field networks with cloud or IoT applications. It is designed from the ground up to meet the challenges of rugged applications at competitive prices, ideal for small business or small order with scale multi-level demands.

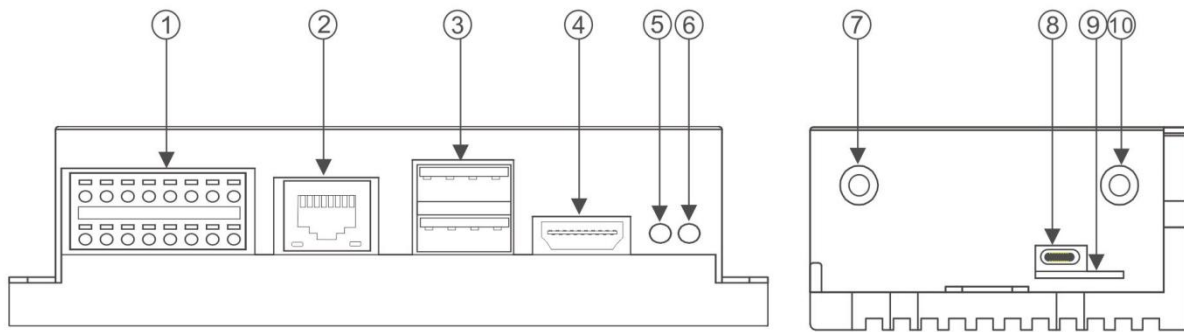
1.1 Features

- State-of-the-art Aluminium chassis for Harsh environment
- Integrated passive heat sink
- Built-in mini PCIe socket for RF module, such as 4G, WI-FI, Lora or Zigbee
- SMA antenna holes x2
- Built in UPS with supercap for safe shutdown
- Encryption chip ATECC608A
- Hardware Watchdog
- RTC with Super Capacitor
- Isolated DI&DO terminal
- 35mm DIN Rail support
- Wide power supply from 9 to 36V DC

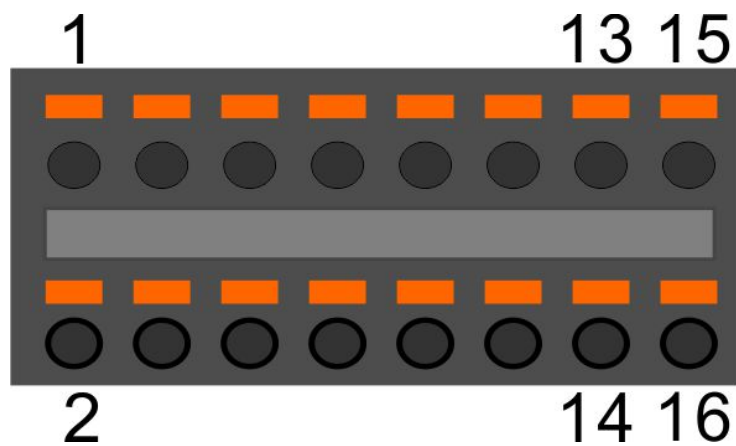
These features make the EdgeBox-RPI4 designed for easy setup and quick deployment for typical industrial applications, such as status monitoring, facility management, digital signage and remote control of public utilities. Furthermore, it is a user-friendly gateway solution with 4 cores ARM Cortex A72 and most industry protocols can save on total deployment costs including electrical power cabling cost and help reduce the product's deployment time. Its ultra-lightweight and compact design is the answer for applications in space-constricting environments ensures it can operate reliably in a variety of extreme environments including in-vehicle applications.



1.2 Interfaces



① Multi-Func phoenix connector



Note	Func name	PIN #	PIN#	Func name	Note
	POWER	1	2	GND	
	RS485_A	3	4	RS232_RX	
	RS485_B	5	6	RS232_TX	
	RS485_GND	7	8	RS232_GND	
	DI0-	9	10	DO0_0	
	DI0+	11	12	DO0_1	
	DI1-	13	14	DO1_0	
	DI1+	15	16	DO1_1	

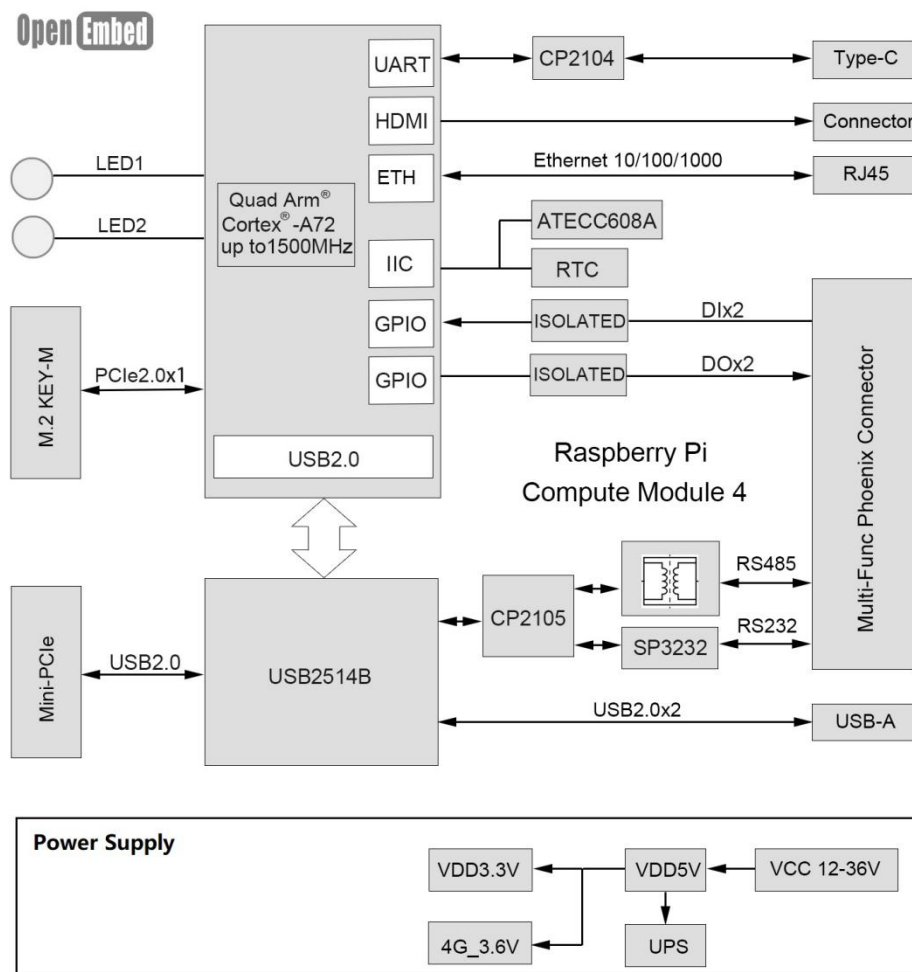
NOTE: 24awg to 16awg cable are suggested

- ② Ethernet connector
- ③ USB 2.0 x 2
- ④ HDMI
- ⑤ LED2
- ⑥ LED1
- ⑦ SMA antenna 1
- ⑧ Console(USB type C)
- ⑨ SIM card slot
- ⑩ SMA antenna 2



1.3 Block Diagram

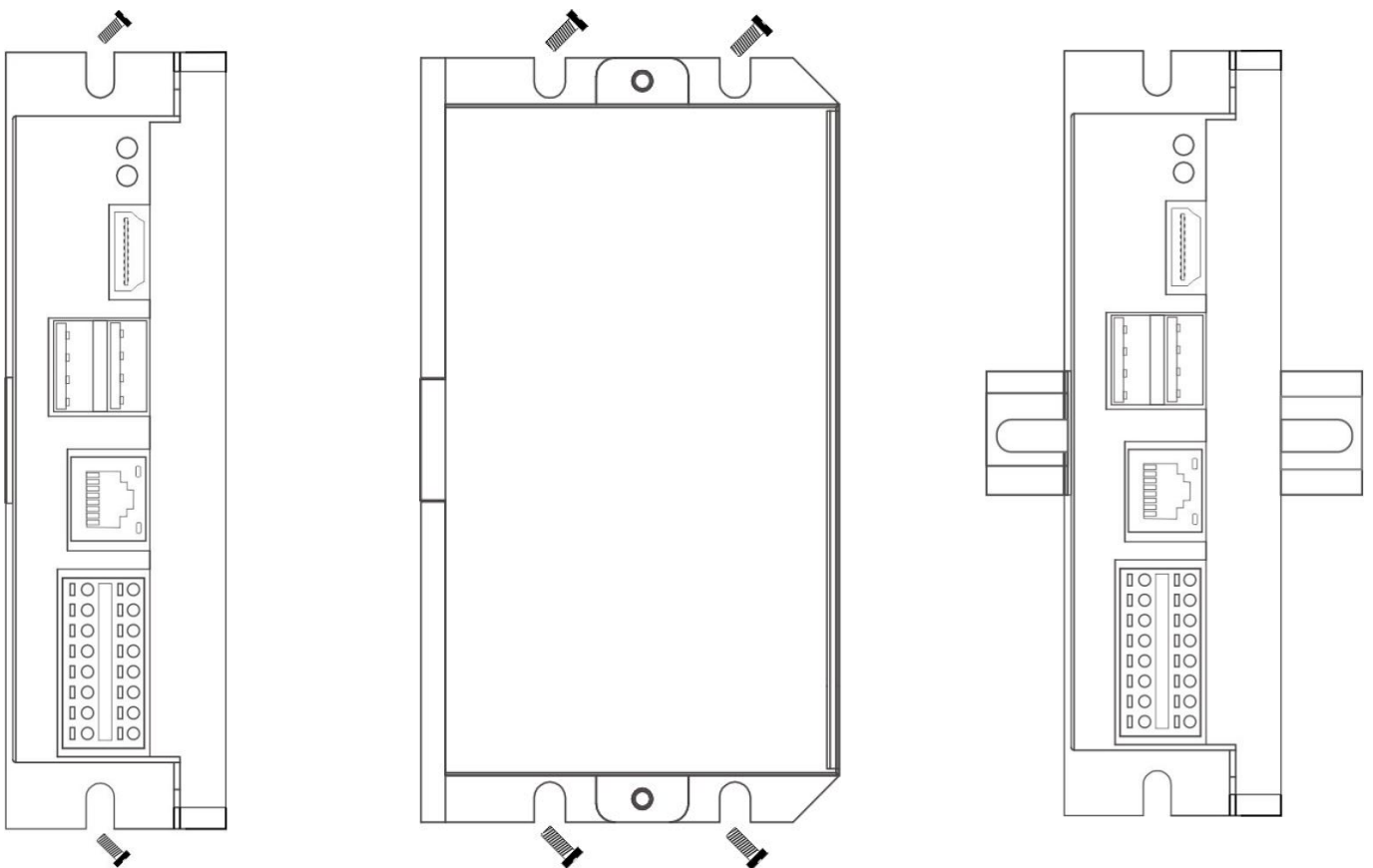
The processing core of the EdgeBox-RPI4 is a Raspberry CM4 board. A OpenEmbed specific base board implements the specific features. Refer to next figure for the block diagram.



2. Installation

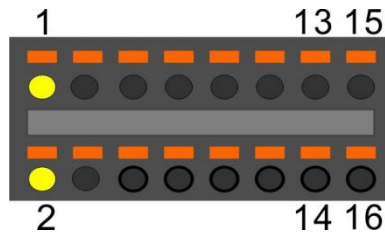
2.1 Mounting

The EdgeBox-RPI4 is intended for two wall mounts, as well one with 35mm DIN-rail . Refer to next figure for the recommended mounting orientation.



2.2 Connectors and Interfaces

2.2.1 Power supply

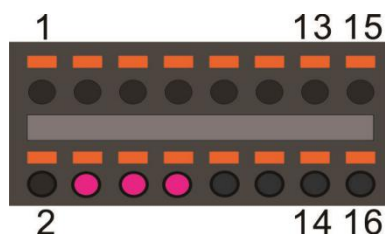


Pin#	Signal	Description
1	POWER_IN	DC 9-36V
2	GND	Ground (Reference potential)



The PE signal is optional. If there is no EMI present ,the PE connection can left open.

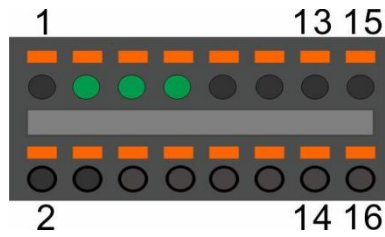
2.2.2 Serial Port (RS232 and RS485)



Pin#	Signal	Description
4	RS232_RX	RS232 receive line
6	RS232_TX	RS232 transmit line
8	GND	Ground (Reference potential)



EdgeBox-RPI4 User Manual

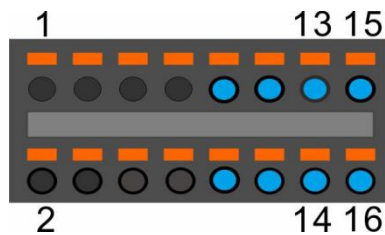


Pin#	Signal	Description
3	RS485_A	RS485 difference line high
5	RS485_B	RS485 difference line low
7	RS485_GND	RS485 Ground (isolated from GND)

The RS485_GND signal is isolated with “GND” signal. If a shielded twisted pair wire is used ,the RS485_GND is connected to the shield.

NOTE:The 120 Ohm termination resistor for RS485 has been installed inside.

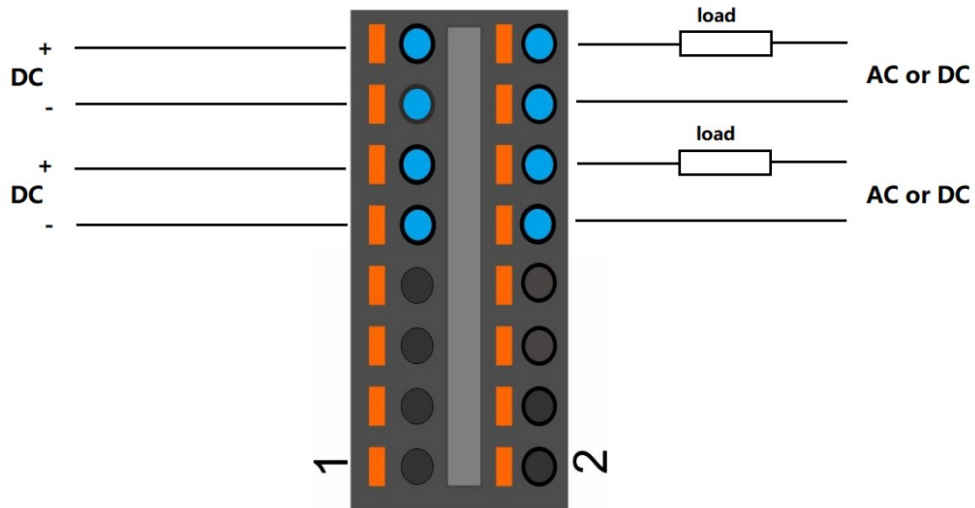
2.2.3 DI&DO



Pin#	Signal of terminal	PIN Level of active	PIN of GPIO from BCM2711	NOTE
09	DI0-	HIGH	GPIO17	
11	DI0+			
13	DI1-	HIGH	GPIO27	
15	DI1+			
10	DO0_0	HIGH	GPIO23	
12	DO0_1			
14	DO1_0	HIGH	GPIO24	
16	DO1_1			

NOTE:





NOTE:

1. DC voltage for input is 24V(+/- 10%).
2. DC voltage for output should be under 60V ,the current capacity is 500ma.
3. Channel 0 and channel 1 of input are isolated to each other
4. Channel 0 and channel 1 of output are isolated to each other

2.2.4 HDMI

Directly connected to the Raspberry PI CM4 board with TVS array.

2.2.5 Ethernet

Ethernet interface is same as Raspberry PI CM4,10/100/1000-BaseT supported, available through the shielded modular jack. Twisted pair cable or shielded twisted pair cable can be used to connect to this port.

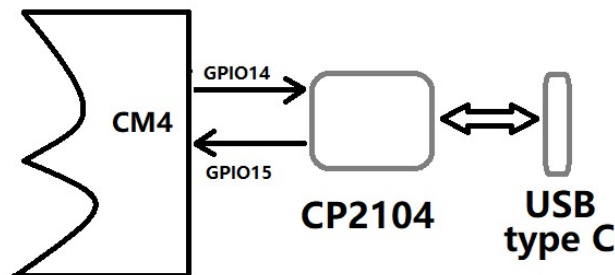


2.2.6 USB HOST

There are two USB interfaces at the connector panel. The two ports share the same electronic fuse.

NOTE: Max current for both ports is limited to 1000ma.

2.2.7 Console(USB typeC)



The design of console used a USB-UART converter, most OS of the computer have the driver, if not, the link below may be useful:

<https://www.silabs.com/products/interface/usb-bridges/classic-usb-bridges/device.cp2104>

This port is used as a Linux console default. You can log into the OS use the settings of 115200,8n1 (Bits: 8, Parity: None, Stop Bits: 1, Flow Control: None). A terminal program such as putty is needed, too.

The default user name is pi and password is raspberry.

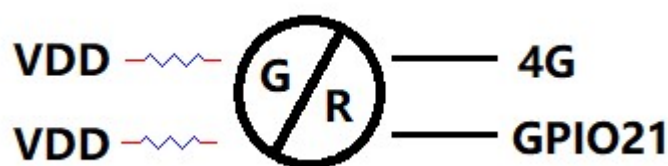
2.2.8 LED

EdgeBox-RPI4 use two green/red dual colour LED as outside indicators.

LED1: green as power indicator and red as eMMC active.



LED2: green as 4G indicator and red as user programmable led connected to GPIO21, low active, programmable.

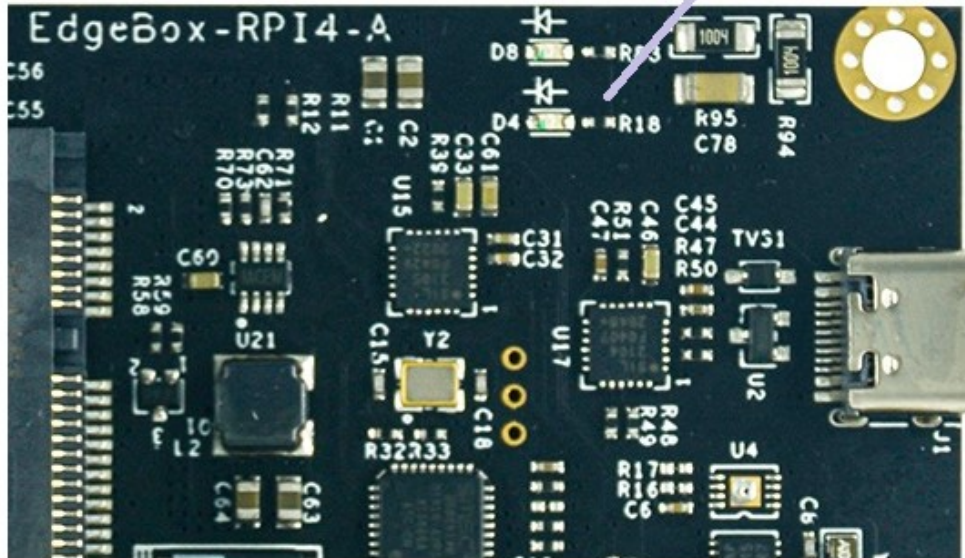


EdgeBox-RPI4 User Manual

EdgeBox-RPI4 also use two green colour LED for debug.

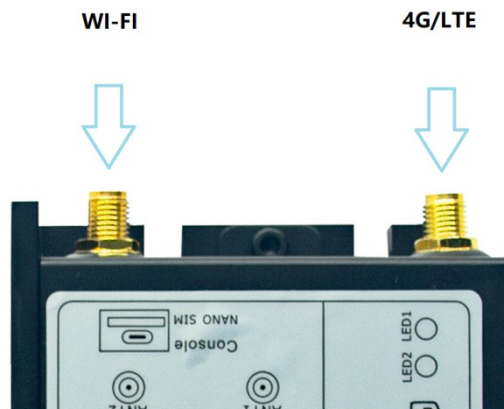
WDT LED

NVME SSD LED



2.2.9 SMA Connector

There are two SMA Connector holes for antennas. The antenna types are very depend on what modules fitted into the Mini-PCle socket. The ANT1 is default used for Mini-PCle socket and ANT2 is for Internal WI-FI signal from CM4 module.

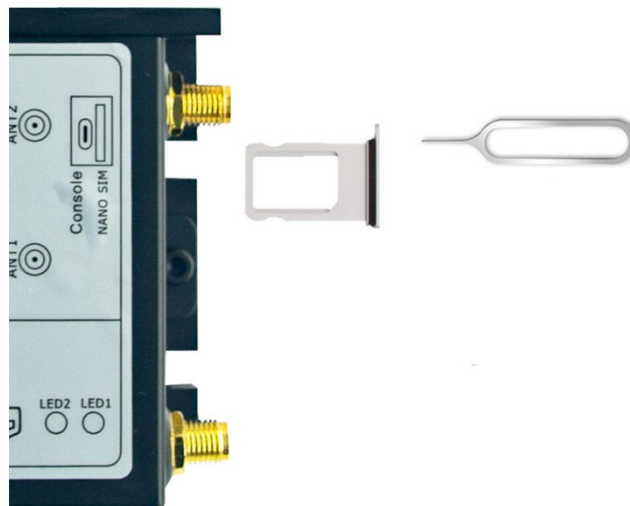


NOTES:

1. The functions of the antennas are not fixed,maybe adjusted to cover other usage.

2.2.10 NANO SIM card slot

The sim card is only needed in cellular(4G,LTE or others based on cellular technology) mode.

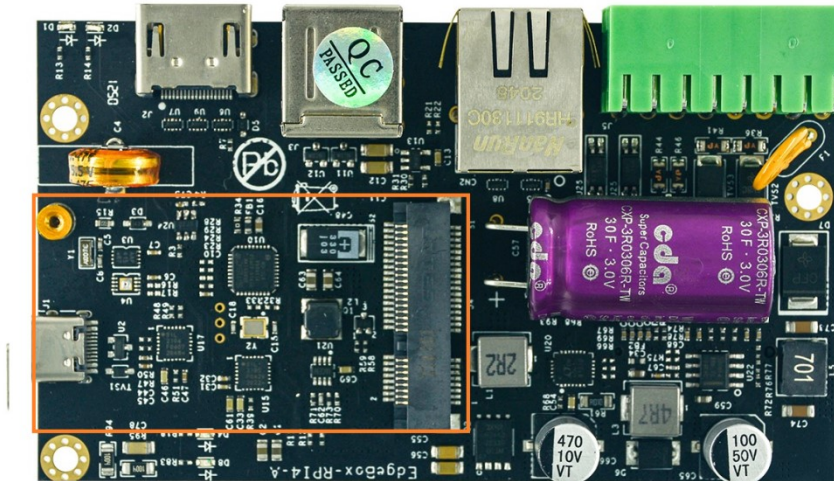


NOTES:

- 1.Only NANO Sim card is accepted,pay attention to the card size.
- 2.The NANO sim card is inserted with chip side top.

2.2.11 Mini-PCle

The orange area is the rough Mini-PCle add-on card position,only one m2x5 screw is needed.



The table below show all the signals. Full size Mini-PCle card are supported.

Signal	PIN#	PIN#	Signal
	1	2	4G_PWR
	3	4	GND
	5	6	USIM_PWR
	7	8	USIM_PWR
GND	9	10	USIM_DATA
	11	12	USIM_CLK
	13	14	USIM_RESET#
GND	15	16	

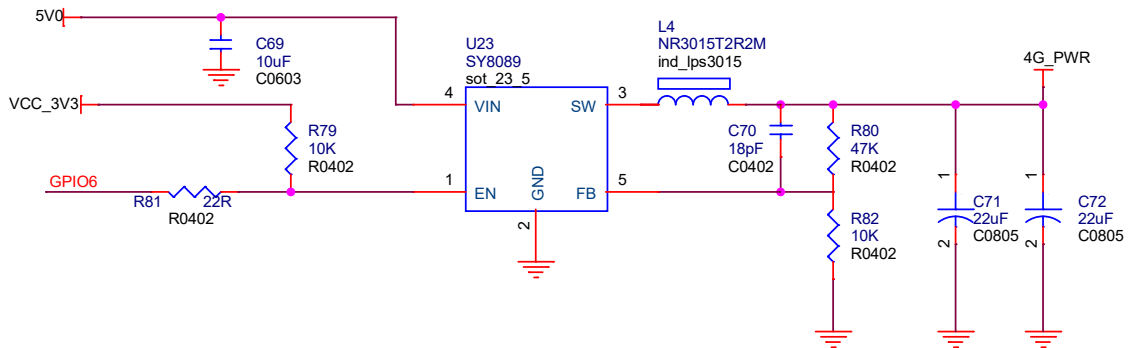


EdgeBox-RPI4 User Manual

	17	18	GND
	19	20	
GND	21	22	PERST#
	23	24	4G_PWR
	25	26	GND
GND	27	28	
GND	29	30	UART_PCIE_TX
	31	32	UART_PCIE_RX
	33	34	GND
GND	35	36	USB_DM
GND	37	38	USB_DP
4G_PWR	39	40	GND
4G_PWR	41	42	4G_LED
GND	43	44	USIM_DET
SPI1_SCK	45	46	
SPI1_MISO	47	48	
SPI1_MOSI	49	50	GND
SPI1_SS	51	52	4G_PWR

NOTE 1: All blank signals are NC(no connect).

NOTE 2: 4G_PWR is the individual power supply for Mini-PCle card . It can be shut down or turn on by the GPIO6 of CM4,the control signal is high active.

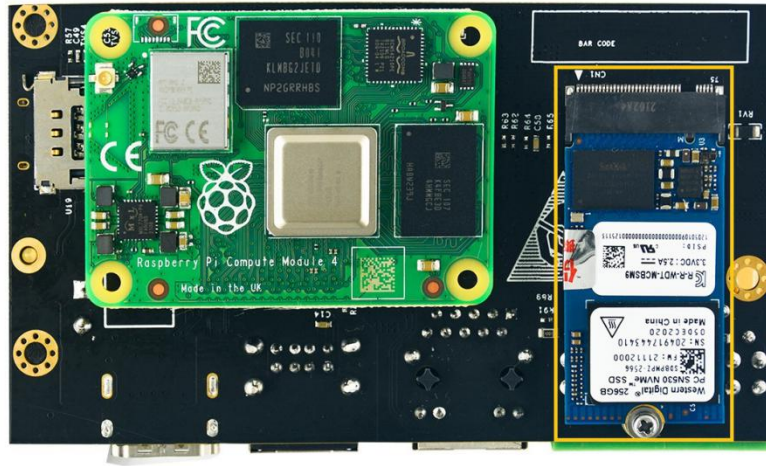


NOTE 3: 4G_LED signal is connected to LED2 internally, refer to section of 2.2.8.

NOTE 4: SPI1 signals are used only for LoraWAN card, such as SX1301,SX1302 from the third company.

2.2.12 M.2

EdgeBox-RPI4 equipped a M.2 socket of M KEY type .ONLY 2242 size NVME SSD card is support,NOT msata.



3. Drivers and Programming Interfaces

3.1 LED

There is a LED used as user indicator, refer to 2.2.8 .

Use LED2 as an example to test the function.

```
$ sudo -i    #enable root account privileges
$ cd /sys/class/gpio
$ echo 21 > export    #GPIO21 which is user LED of LED2
$ cd gpio21
$ echo out > direction
```



```
$ echo 0 > value # turn on the user LED, LOW active
OR
$ echo 1 > value # turn off the user LED
```

3.2 Serial Port (RS232 and RS485)

There are two individual serial ports in the system. The /dev/ttyUSB1 as RS232 port and /dev/ttyUSB0 as RS485 port. Use RS232 as a example.

```
$ python
>>> import serial
>>> ser=serial.Serial('/dev/ttyUSB1',115200,timeout=1)
>>> ser.isOpen()
true
>>> ser.isOpen()
>>> ser.write('1234567890')
10
```

3.3 Cellular over Mini-PCle

Use Quectel EC20 as a example and follow the steps :

1. **Insert the EC20 into Mini-PCle socket and micro sim card in related slot, connect the antenna.**
2. **Log in the system via console use pi/raspberry.**
3. **Turn on the power of Mini-PCle socket and release the reset signal.**

```
$ sudo -i #enable root account privileges
$ cd /sys/class/gpio
$ echo 6 > export #GPIO6 which is POW_ON signal
$ echo 5 > export #GPIO5 which is reset signal

$ cd gpio6
$ echo out > direction
$ echo 1 > value # turn on the power of Mini PCle
AND
$ cd gpio5
$ echo out > direction
$ echo 1 > value # release the reset signal of Mini PCle
```

NOTE: Then the LED of 4G is start to flash.

4. **Check the device:**

```
$ lsusb
$ Bus 001 Device 005: ID 2c7c:0125 Quectel Wireless Solutions Co., Ltd. EC25 LTE modem
.....
$ dmesg
$
```



.....

```
[ 185.421911] usb 1-1.3: new high-speed USB device number 5 using dwc_otg
[ 185.561937] usb 1-1.3: New USB device found, idVendor=2c7c, idProduct=0125, bcdDevice= 3.18
[ 185.561953] usb 1-1.3: New USB device strings: Mfr=1, Product=2, SerialNumber=0
[ 185.561963] usb 1-1.3: Product: Android
[ 185.561972] usb 1-1.3: Manufacturer: Android
[ 185.651402] usbcore: registered new interface driver cdc_wdm
[ 185.665545] usbcore: registered new interface driver option
[ 185.665593] usbserial: USB Serial support registered for GSM modem (1-port)
[ 185.665973] option 1-1.3:1.0: GSM modem (1-port) converter detected
[ 185.666283] usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB2
[ 185.666499] option 1-1.3:1.1: GSM modem (1-port) converter detected
[ 185.666701] usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB3
[ 185.666880] option 1-1.3:1.2: GSM modem (1-port) converter detected
[ 185.667048] usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB4
[ 185.667220] option 1-1.3:1.3: GSM modem (1-port) converter detected
[ 185.667384] usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB5
[ 185.667810] qmi_wwan 1-1.3:1.4: cdc-wdm0: USB WDM device
[ 185.669160] qmi_wwan 1-1.3:1.4 wwan0: register 'qmi_wwan' at usb-3f980000.usb-1.3, WWAN/QMI
device,xx:xx:xx:xx:xx:xx
```

xx:xx:xx:xx:xx:xx is the MAC address.

\$ ifconfig -a

.....

```
wwan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 169.254.69.13  netmask 255.255.0.0  broadcast 169.254.255.255
    inet6 fe80::8bc:5a1a:204a:1a4b  prefixlen 64  scopeid 0x20<link>
    ether 0a:e6:41:60:cf:42  txqueuelen 1000  (Ethernet)
    RX packets 0    bytes 0 (0.0 B)
    RX errors 0    dropped 0  overruns 0  frame 0
    TX packets 165  bytes 11660 (11.3 KiB)
    TX errors 0    dropped 0 overruns 0  carrier 0  collisions 0
```

5. How to use AT command

\$ miniterm --- Available ports:

```
--- 1: /dev/ttyAMA0      'ttyAMA0'
--- 2: /dev/ttyUSB0      'CP2105 Dual USB to UART Bridge Controller'
--- 3: /dev/ttyUSB1      'CP2105 Dual USB to UART Bridge Controller'
--- 4: /dev/ttyUSB2      'Android'
--- 5: /dev/ttyUSB3      'Android'
--- 6: /dev/ttyUSB4      'Android'
--- 7: /dev/ttyUSB5      'Android'
```

--- Enter port index or full name:

\$ miniterm /dev/ttyUSB5 115200

Some useful AT command :



EdgeBox-RPI4 User Manual

- AT //should return OK
- AT+QINISTAT //return the initialization status of (U)SIM card, the response should be 7
- AT+QCCID //returns the ICCID (Integrated Circuit Card Identifier) number of the (U)SIM card

6. How to dial

```
$su root
$ cd /usr/app/linux-ppp-scripts
$ ./quectel-pppd.sh
```

Then the 4G led is flashing.

If success, the return like this:

```
pi@raspberrypi:~$ ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether e4:5f:01:1f:e1:46 txqueuelen 1000 (Ethernet)
    RX packets 32437 bytes 4605705 (4.3 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 3177 bytes 370881 (362.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 1000 (Local Loopback)
    RX packets 16 bytes 1328 (1.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 16 bytes 1328 (1.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ppp0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500
    inet 10.162.91.105 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 4 bytes 58 (58.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wwan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 169.254.165.219 netmask 255.255.0.0 broadcast 169.254.255.255
    inet6 fe80::d837:131c:a6c7:9399 prefixlen 64 scopeid 0x20<link>
    ether b6:a0:18:51:e3:ef txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 75 bytes 9395 (9.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

7. Add the router path

```
$ route add default gw 10.64.64.64 or your gateway XX.XX.XX.XX
```

Then have a test

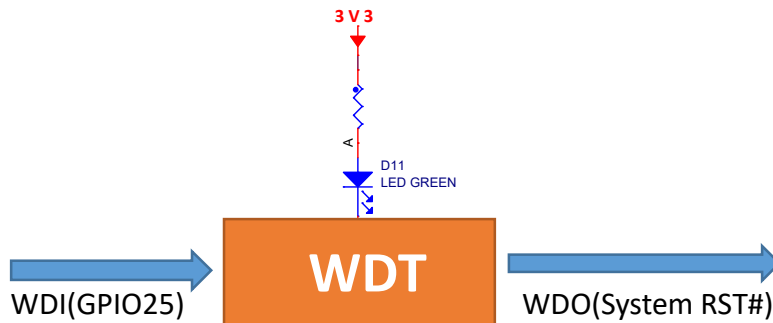
```
$ ping google.com
```



3.4 WDT

3.4.1 Block Diagram of WDT

The WDT module have three terminals, input ,output and LED indicator.



Note: The LED is optional and not available in earlier hardware version.

3.4.2 How it works

1. System POWER ON.
2. Delay 200ms.
3. Send WDO a negative pulse with 200ms low level to reset the system.
4. Pull up WDO.
5. Delay 120 seconds while the indicator flashing(typical 1hz).
6. Turn off the indicator.
7. Wait for 8 pulses at WDI to active WDT module and light the LED.
8. Get Into WDT-FEED mode , at least one pulse should be feed into WDI in at least every 2 seconds, if not,the WDT module should output a negative pulse to reset the system.
9. Goto 2.

3.5 RTC

3.5.1

The chip of RTC is MCP79410 from microchip.It is mounted on the system I2C bus.



The OS itself has the driver inside, only we need are some configurations.

3.5.2

Open `/etc/rc.local` AND add 2 lines:

```
echo "mcp7941x 0x6f" > /sys/class/i2c-adapter/i2c-1/new_device  
hwclock -s
```

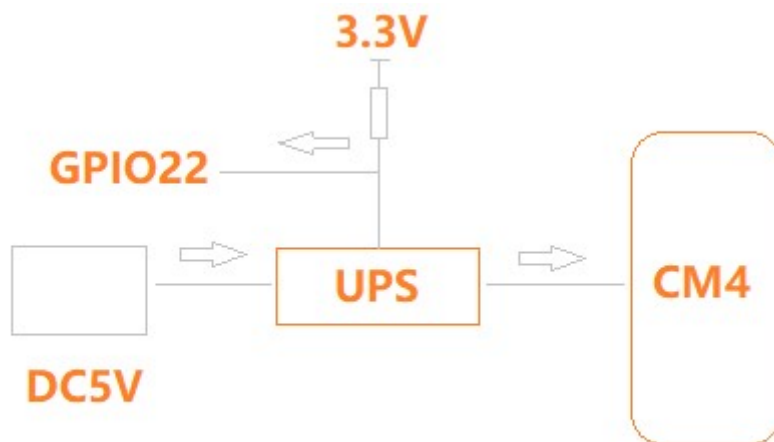
Then reset the system and the RTC is working.

Note:

1. make sure the i2c-1 driver point is open, and the point is closed default.
2. the estimated backup time of the RTC is 15 days.

3.10 UPS for safe shut down

The UPS module diagram is listed below.



The UPS module is inserted between the DC5V and CM4, a GPIO is used to alarm CPU when the 5V power supply is down. Then the CPU should do something urgent in a script before energy exhaustion of super capacitor and run a “\$shutdown”

Another way to use this function is Initiate a shutdown when GPIO pin changes. The given GPIO pin is configured as an input key that generates KEY_POWER events. This event is handled by systemd-logind by initiating ashutdown. Systemd versions older than 225 need an udev rule enable listening to the input



device:

Use /boot/overlays/README as reference, then modify /boot/config.txt.

dtoverlay=gpio-shutdown, gpio_pin=GPIO22,active_low=1

NOTE:The alarm signal is active LOW.



4. Electrical specifications

4.1 Power consumption

The power consumption of the EdgeBox-RPI4 strongly depends on the application, the mode of operation and the peripheral devices connected. The given values have to be seen as approximate values. The following table shows power consumption parameters of the EdgeBox-RPI4:

Note: On condition of power supply 24V, no add-on card in sockets and no USB devices.

Mode of operation	Current(ma)	Power	Remark
Idle	81		
Stress test	172		stress -c 4 -t 10m -v &

4.2 UPS

The backup time of UPS module is very depend on the system load of the system. Some typical conditions are listed below. The test module of CM4 is 4GB LPDDR4,32GB eMMC with Wi-Fi module.

Mode of operation	Time(second)	Remark
Idle	55	
Full load of CPU	18	stress -c 4 -t 10m -v &

5. Mechanical Drawings

TBD

