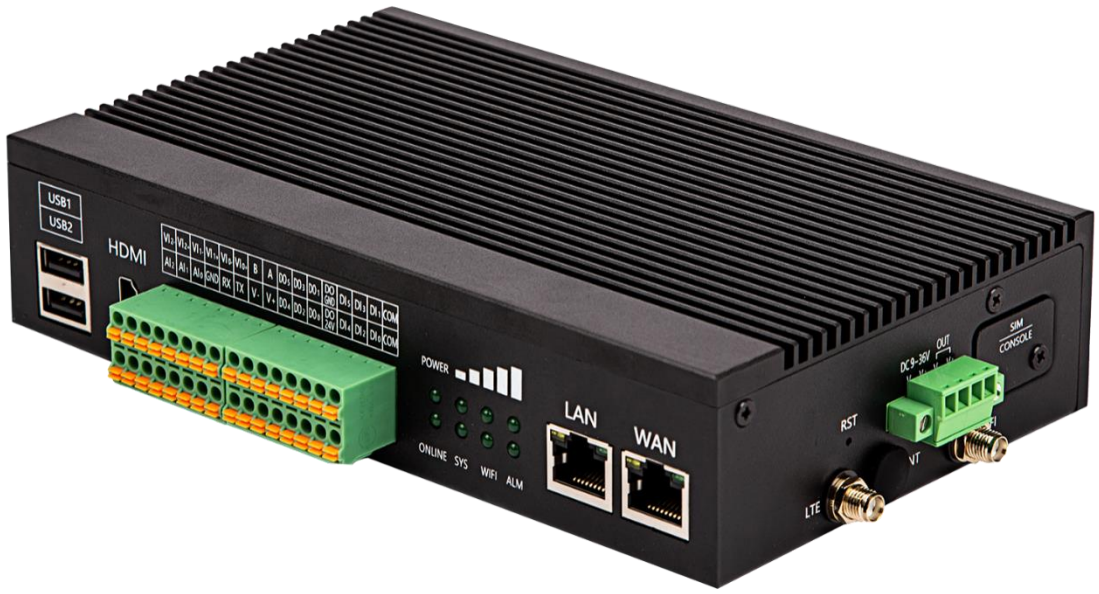
Edge Computing Gateway

EG500

Administrator Programming Guide



Contents

1. Initial Driver Installation

1.1 Install Ethernet Port Driver

The Ethernet port labeled “LAN” on EG500 was bridged from PCIe by Realtek network chipset, you may need install the Ethernet port driver to run it up.

```
$ cd ~
$ sudo apt-get update
$ sudo apt-get install raspberrypi-kernel-headers
$ git clone https://github.com/Elastel/r8168.git
$ cd r8168
$ sudo ./autorun.sh
```

1.2 Install commands

Also Elastel provide some prebuilt commands for I/O ports and others operating. They are,

Command Name	Usage description
read_di	Read the Digital input ports value
read_do	Read the Digital output ports value
write_do	Write the Digital output ports value
read_ai	Read the current type analog input value
read_vi	Read the voltage type analog input value
led_ctl	LED control
quectl-CM	The cellular module dialing program

Use below commands to install it.

$ cd ~

$ git clone https://github.com/Elastel/elastel_command.git

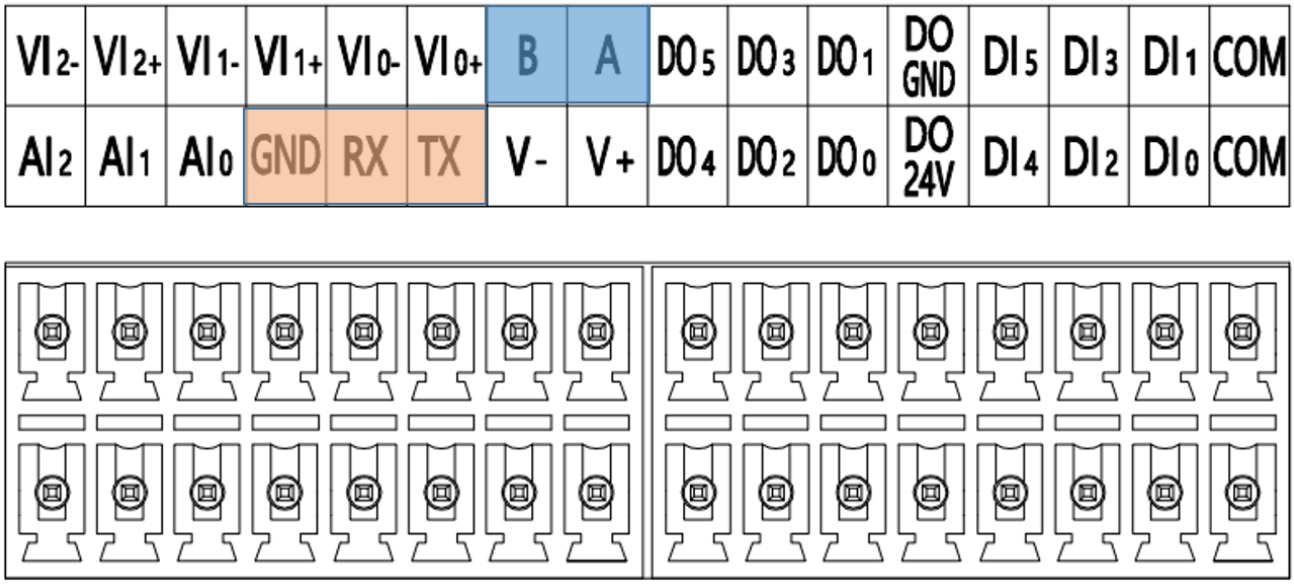
$ cd elastel_command

$ chmod +x install.sh

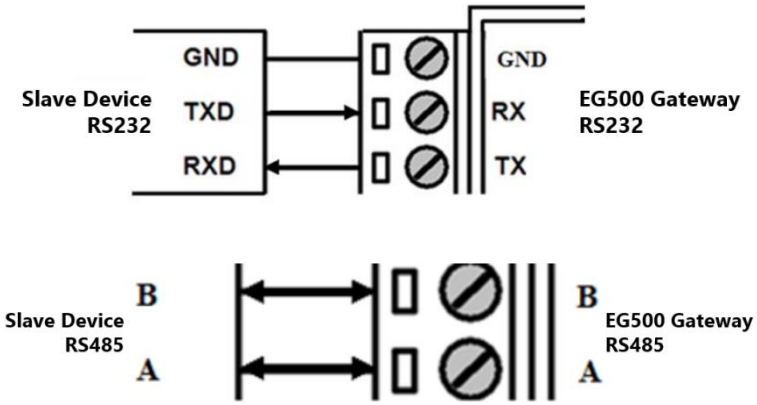
$ sudo ./install.sh

2. Serial Port (RS232 and RS485)

2.1 PIN Definition and Wiring



PIN	Description
TX	RS232 transmit line
RX	RS232 receive line
GND	Ground (Reference potential)
A	RS485 difference line high
B	RS485 difference line low



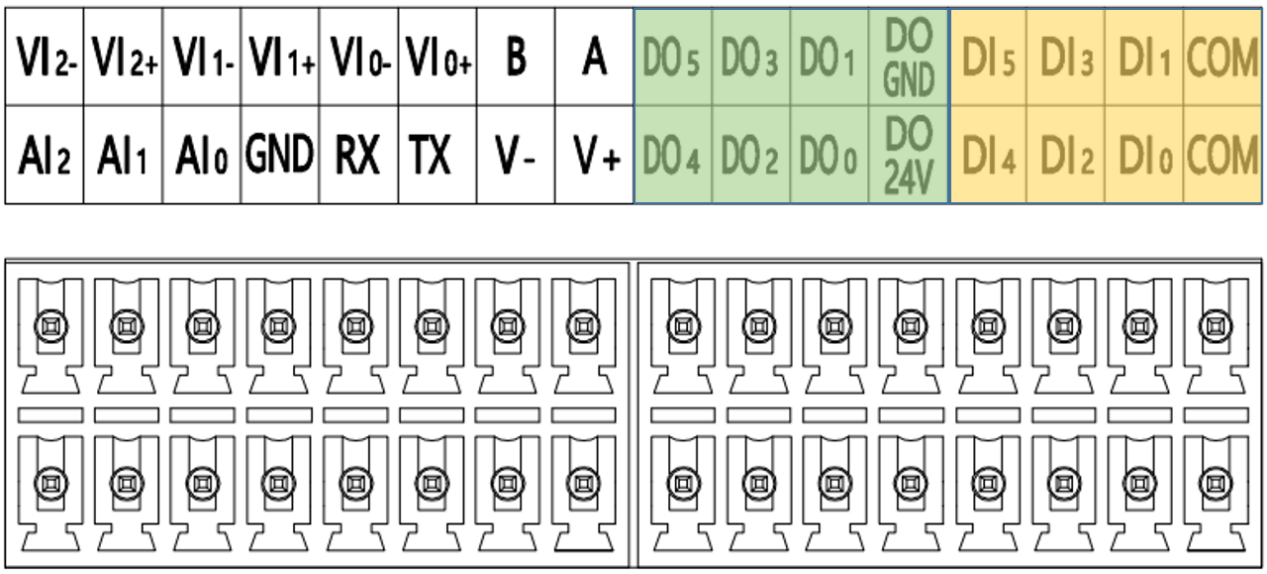
2.2 Programming

There are two individual serial ports in the system. The /dev/ttyACM1 for RS232 port, and /dev/ttyACM0 for RS485 port. Use RS232 as an example.

```
$ python
>>> import serial
>>> ser=serial.Serial('/devttyACM1',115200,timeout=1)
>>> ser.isOpen()
true
>>> ser.write('1234567890')
10
```

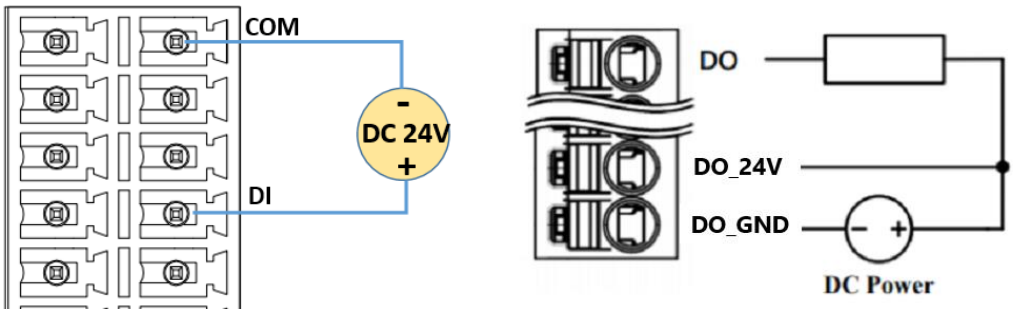
3. DI & DO

3.1 PIN Definition and Wiring



Interfaces	PIN Level of active	PIN of GPIO from BCM2711
DI0	HIGH	GPIO4
DI1	HIGH	GPIO17
DI2	HIGH	GPIO18
DI3	HIGH	GPIO27
DI4	HIGH	GPIO23
DI5	HIGH	GPIO24
COM	/	/
COM	/	/
DO0	HIGH	GPIO22
DO1	HIGH	GPIO20
DO2	HIGH	GPIO13
DO3	HIGH	GPIO12
DO4	HIGH	GPIO16
DO5	HIGH	GPIO19

DO_24V	/	/
DO_GND	/	/



3.2 Programming

Use DI0 as an example to test the DI function.

```
$ sudo -i                #enable root account privileges
$ cd /sys/class/gpio
$ echo 4 > export        #GPIO4 which is user of DO0
$ cd gpio4
$ echo in > direction
$ cat value              # View the value of DI
```

Use DO0 as an example to test the DO function.

```
$ sudo -i                #enable root account privileges
$ cd /sys/class/gpio
$ echo 22 > export       #GPIO22 which is user of DO0
$ cd gpio22
$ echo out > direction
$ echo 1 > value         # turn on the DO0, HIGH active
OR
$ echo 0 > value             # turn off the DO0
```

Another way is using the Elastel prebuilt commands read_di, read_do, write_do to control the interfaces. Examples as below:

$ sudo read_di

read_di <0|1|2|3|4|5>, eg: read_di 1

$ sudo read_di 0

DI0 = 0

$ sudo read_do

read_do <0|1|2|3|4|5>, eg: read_do 1

$ sudo read_do 0

DO0 = 1

$ sudo write_do

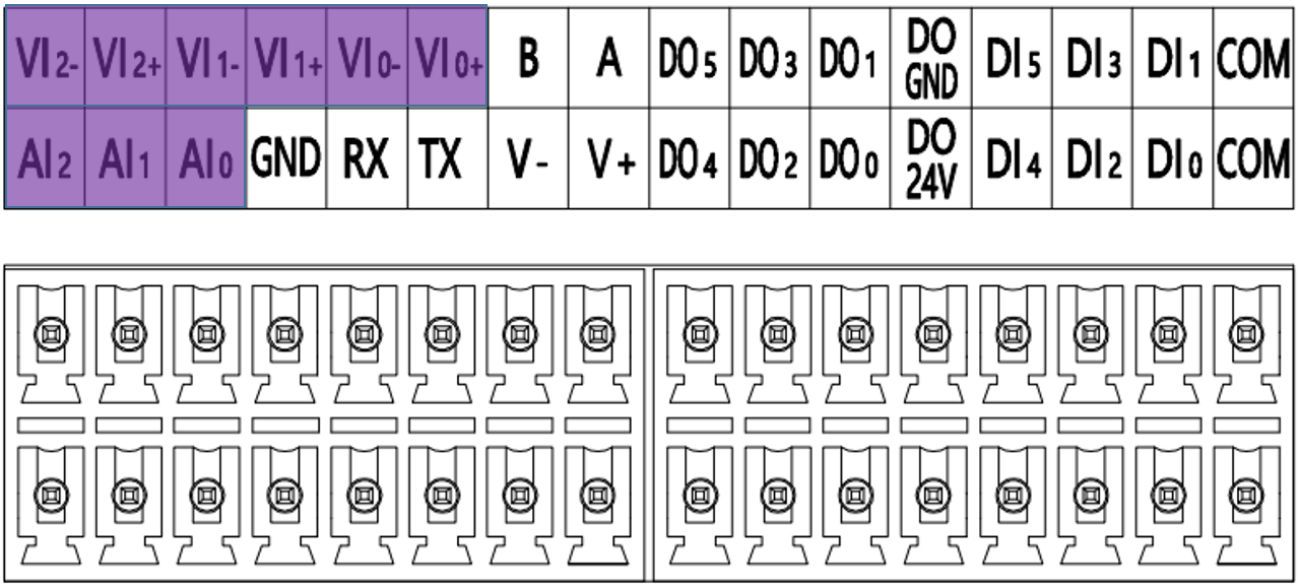
write_do <0|1|2|3|4|5> 1|0, eg: write_do 1 1

$ sudo write_do 0 0

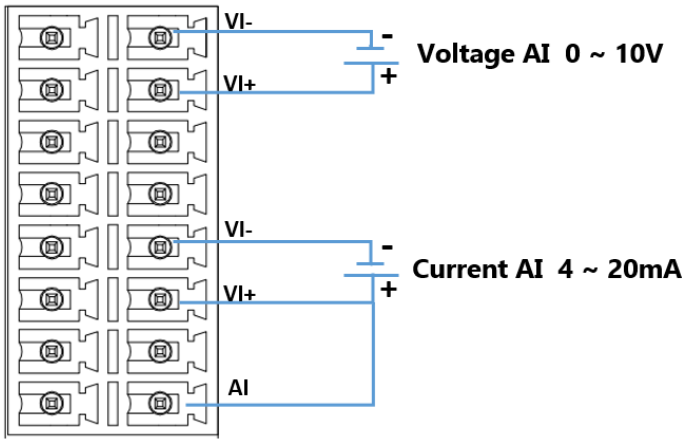
success to write DO0:0

4. VI & AI

4.1 PIN Definition and Wiring



PIN	Description
VI0+	Voltage type analog input channel 0 +
VI0-	Voltage type analog input channel 0 -
VI1+	Voltage type analog input channel 1 +
VI1-	Voltage type analog input channel 1 -
VI2+	Voltage type analog input channel 2 +
VI2-	Voltage type analog input channel 2 -
AI0	Current type analog input channel 0
AI1	Current type analog input channel 1
AI2	Current type analog input channel 2



4.2 Programming

Use the Elastel prebuilt commands read_vi and read_ai to get the values of analog input value.

Examples as below:

$ read_vi

read_vi <0|1|2>; eg: read_vi 1

$ read_vi 0

VI0 = 0.009V

$ read_ai

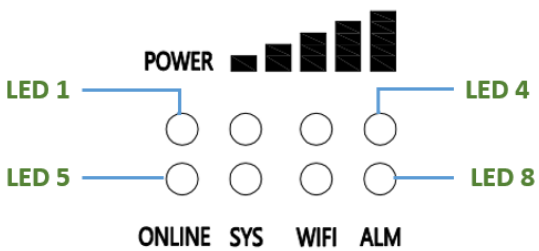
read_ai <0|1|2>; eg: read_ai 1

$ read_ai 0

AI0 = 0.000mA

5. LED

5.1 LED Definition



LED #	LED	Description
LED 1	POWER	Power indicator
LED 2~4	SIG	Cellular Signal strength indicator
LED 5	ONLINE	Network online status indicator
LED 6	SYS	System status indicator
LED 7	WiFi	WiFi status indicator
LED 8	ALM	Alarm indicator

5.2 Programming

Use the Elastel prebuilt command **led_ctl** to control the LEDs. Examples as below:

$ led_ctl

led_ctl -m < system | wifi | alarm | online | sig > -s <on | off |none | low | mid | high>

$ sudo led_ctl

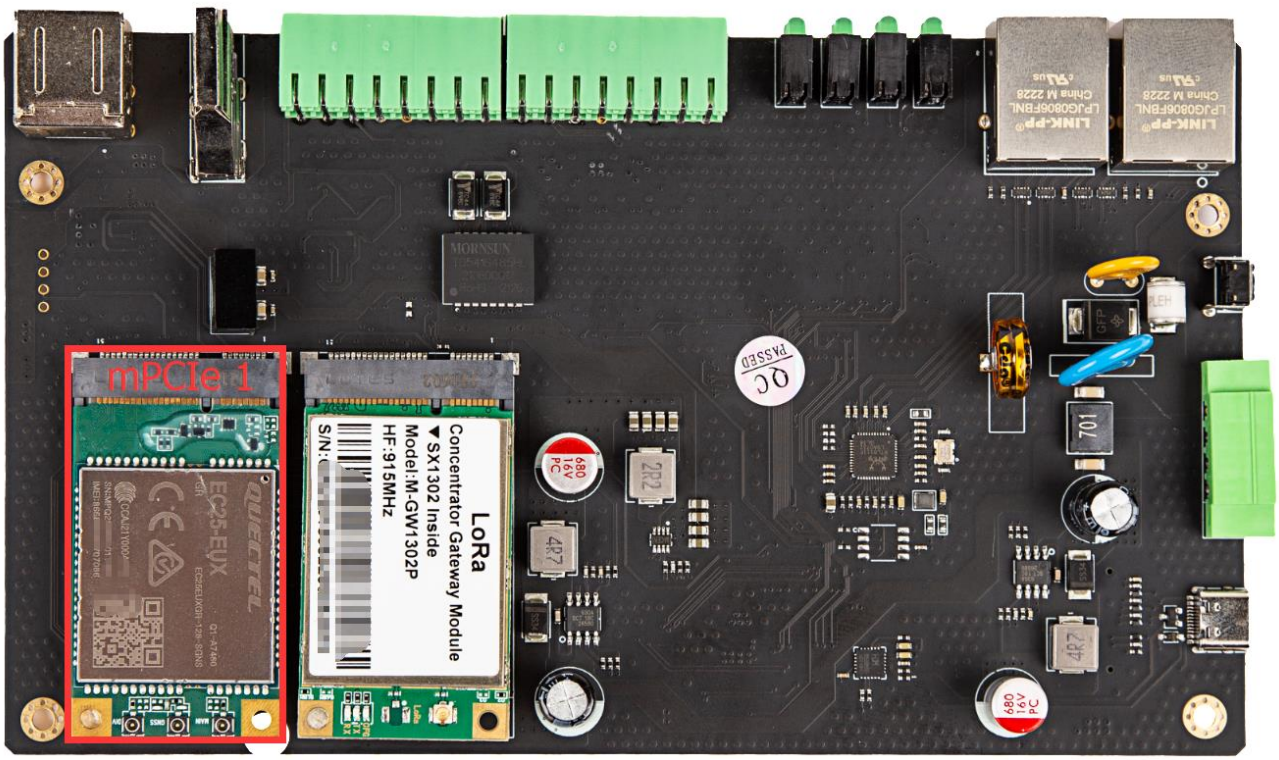
led_ctl -m < system | wifi | alarm | online | sig > -s <on | off | none | low | mid | high>

$ sudo led_ctl -m system -s on

Status	Description
on	Turn on led, does not support in SIG
off	Turn off led, does not support SIG
none	no signal, only for SIG
low	low signal, only for SIG
mid	Medium signal, only for SIG
high	high signal, only for SIG

6. Cellular Module over Mini-PCIe 1

6.1 Module Installation



Use Quectel EC25 as an example and following the steps :

1. **Insert the EC25 into Mini-PCIe socket 1 and micro SIM card in related slot, connect the LET antenna (Please check the user guide for more installation instruction).**
2. **Log in the system via console use admin/admin.**
3. **Turn on the power of Mini-PCIe socket and release the reset signal.**

```
$ sudo -i                    #enable root account privileges
$ cd /sys/class/gpio
$ echo 5 > export            #GPIO5 which is reset signal
$ cd gpio5
$ echo out > direction
$ echo 1 > value             # release the reset signal of Mini PCIe
```

Check the device:

$ lsusb

Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub

Bus 001 Device 005: ID 046d:c534 Logitech, Inc. Unifying Receiver

Bus 001 Device 004: ID 2c7c:0125 Quectel Wireless Solutions Co., Ltd. EC25 LTE modem

Bus 001 Device 003: ID 1a86:55d2 QinHeng Electronics USB Dual_Serial

Bus 001 Device 002: ID 1a86:8091 QinHeng Electronics USB2.0 HUB

Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub

$ dmesg

[3.563608] usbserial: USB Serial support registered for GSM modem (1-port)

[3.564137] option 1-1.2:1.0: GSM modem (1-port) converter detected

[3.574237] usb 1-1.2: GSM modem (1-port) converter now attached to ttyUSB0

[3.574645] option 1-1.2:1.1: GSM modem (1-port) converter detected

[3.575282] usb 1-1.2: GSM modem (1-port) converter now attached to ttyUSB1

[3.601785] qmi_wwan 1-1.2:1.4: cdc-wdm0: USB WDM device

[3.603381] qmi_wwan 1-1.2:1.4 wwan0: register 'qmi_wwan' at usb-fe9c0000.xhci-1.2, WWAN/QMI device, xx:xx:xx:xx:xx:xx

[3.603654] usbcore: registered new interface driver qmi_wwan

[3.603691] option 1-1.2:1.2: GSM modem (1-port) converter detected

[3.609664] usb 1-1.2: GSM modem (1-port) converter now attached to ttyUSB2

[3.610081] option 1-1.2:1.3: GSM modem (1-port) converter detected

[3.610498] usb 1-1.2: GSM modem (1-port) converter now attached to ttyUSB3

$ ifconfig -a

……

wwan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500

inet 169.254.58.161 netmask 255.255.0.0 broadcast 169.254.255.255

inet6 fe80::9e8a:5252:d4c8:15cb prefixlen 64 scopeid 0x20<link>

ether c6:00:46:68:12:80 txqueuelen 1000 (Ethernet)

RX packets 0 bytes 0 (0.0 B)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 133 bytes 33576 (32.7 KiB)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

6.2 Dialing Programming

Connect to the network using the Elastel prebuilt command quectel-CM

Execute $quectel-CM –h for more details.

$ sudo quectel-CM

[08-02_09:39:09:900] Quectel_QConnectManager_Linux_V1.6.0.24

[08-02_09:39:09:904] Find /sys/bus/usb/devices/1-1.2 idVendor=0x2c7c idProduct=0x125, bus=0x001, dev=0x004

[08-02_09:39:09:905] Auto find qmichannel = /dev/cdc-wdm0

[08-02_09:39:09:905] Auto find usbnet_adapter = wwan0

[08-02_09:39:09:906] netcard driver = qmi_wwan, driver version = 5.15.32-v8+

[08-02_09:39:09:906] Modem works in QMI mode

[08-02_09:39:09:939] cdc_wdm_fd = 7

[08-02_09:39:10:038] Get clientWDS = 20

[08-02_09:39:10:070] Get clientDMS = 2

[08-02_09:39:10:104] Get clientNAS = 4

[08-02_09:39:10:135] Get clientUIM = 1

[08-02_09:39:10:167] Get clientWDA = 1

[08-02_09:39:10:199] requestBaseBandVersion EC20CEFAGR06A14M4G

[08-02_09:39:10:327] requestGetSIMStatus SIMStatus: SIM_READY

[08-02_09:39:10:360] requestGetProfile[1] ///0

[08-02_09:39:10:391] requestRegistrationState2 MCC: 460, MNC: 11, PS: Attached, DataCap: LTE

[08-02_09:39:10:423] requestQueryDataCall IPv4ConnectionStatus: DISCONNECTED

[08-02_09:39:10:424] ifconfig wwan0 0.0.0.0

[08-02_09:39:10:432] ifconfig wwan0 down

[08-02_09:39:10:487] requestSetupDataCall WdsConnectionIPv4Handle: 0x87338dc0

[08-02_09:39:10:615] ifconfig wwan0 up

[08-02_09:39:10:621] No default.script found, it should be in '/usr/share/udhcpc/' or '/etc//udhcpc' depend on your udhcpc version!

[08-02_09:39:10:621] busybox udhcpc -f -n -q -t 5 -i wwan0

udhcpc: started, v1.30.1

udhcpc: sending discover

udhcpc: sending select for xx.xx.xx.xx

udhcpc: lease of xx.xx.xx.xx obtained, lease time 7200

[08-02_09:39:10:749] ip -4 address flush dev wwan0

[08-02_09:39:10:753] ip -4 address add xx.xx.xx.xx/30 dev wwan0

[08-02_09:39:10:759] ip -4 route add default via xx.xx.xx.xx dev wwan0

6.3 GPS Programming

Some cellular module like Quectel EC25 support built-in base station GPS data acquisition. To get GPS data from cellular module, install cellular module on mini PCIe 1 as above section indicated.

You may also need an external GPS antenna installed on site for better signal.

1) Send AT command to enable GPS data acquisition to /dev/ttyUSB2 block

   AT+QGPS=1 # To enable GPS acquisition.

   AT+QGPS=0 # To disable GPS acquisition.

   Example:

   echo -en "AT+QGPS=1\r\n" > /dev/ttyUSB2 | cat /dev/ttyUSB2 &

   return:

   OK

2) Obtain the GPS data from /dev/ttyUSB1

   Example:

   cat /dev/ttyUSB1

   Return:

$GPVTG,,T,,M,,N,,K,N*2C

$GPGSA,A,1,,,,,,,,,,,,,*1E

$GPGGA,,,,,,0,,,,,,,,*66

$GPRMC,,V,,,,,,,,,,N*53

3) Common useful AT commands

AT Command	Description
AT+ICCID	The ICCID (Integrated Circuit Card Identifier) number of the SIM card
AT+CIMI	Query IMSI number of SIM which is attached to ME
AT+CGSN	The International Mobile Equipment Identity (IMEI).
AT+COPS?	The current operators and their status and allows to set automatic or manual network selection.
AT+CSQ	indicates the received signal strength <rssi> and the channel bit error rate <ber>

7. WiFi

Check the device:

$ ifconfig

wlan0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500

ether e4:5f:01:8b:be:d1 txqueuelen 1000 (Ethernet)

RX packets 6760 bytes 877872 (857.2 KiB)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 376 bytes 31652 (30.9 KiB)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

7.1 STA Mode

1. Start up the wlan

   sudo ifconfig wlan0 up

2. Scan the available WiFi Hotspot

   sudo iw dev wlan0 scan |less

3. Create the password file and connecting

   # Create the config file

   sudo wpa_passphrase ESSID PWD > xxx.conf

   # Use the config file to connect

   sudo wpa_supplicant -B -iwlan0 -c ./xxx.conf

   # Check the connection status

   sudo iwconfig wlan0

7.2 AP Mode

Use the “creat_ap” tool to enable the WiFi hotspot.

1. Install the necessary dependency

   sudo apt-get install hostapd dnsmasq iptables

2. Install create_ap tool from Github

   git clone https://github.com/oblique/create_ap

   cd create_ap

   make install

3. Enable WIFI Hotspot

   (1) Create WiFi hotspot without password.

   create_ap wlan0 eth0 MyAccessPoint

   (2) Create WiFi hotspot with WPA+WPA2 authentication.

   create_ap wlan0 eth0 MyAccessPoint MyPassPhrase

   (3) Create WiFI Hotspot without internet sharing

   create_ap -n wlan0 MyAccessPoint MyPassPhrase

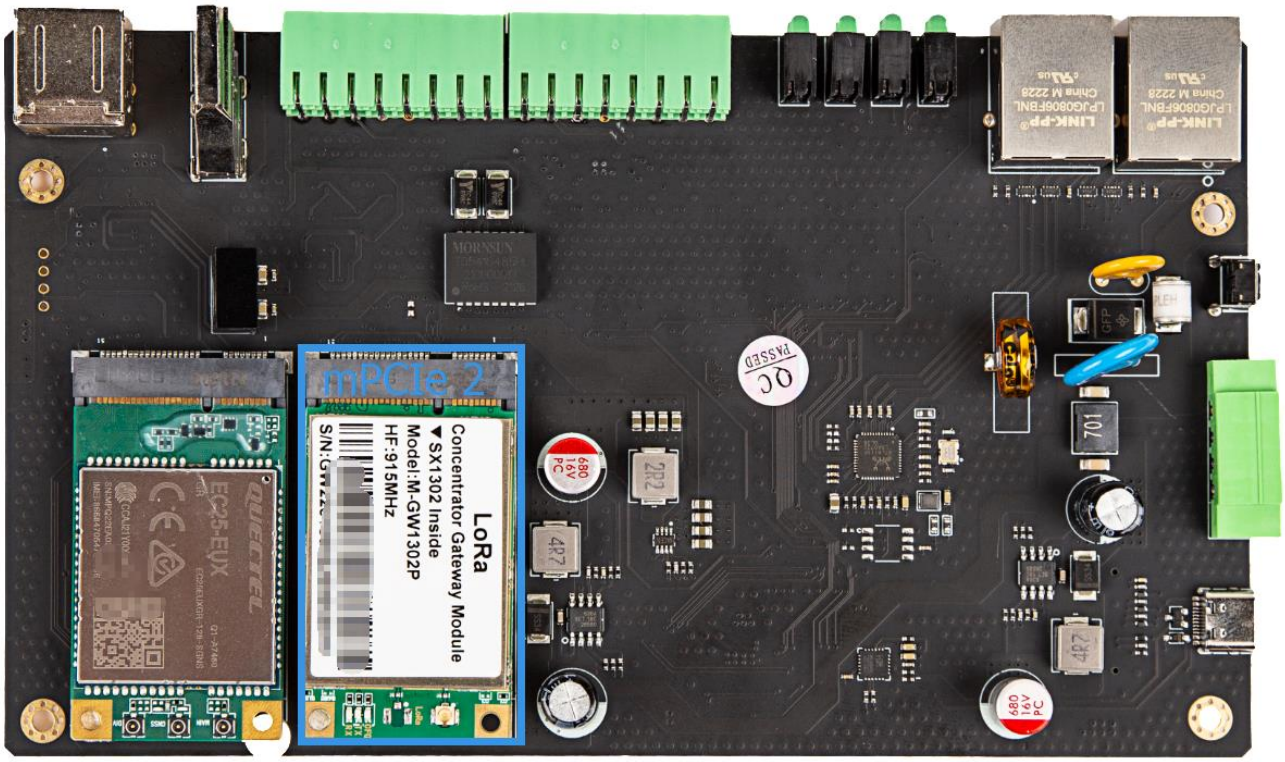
(4) Other methods please refer to https://github.com/oblique/create_ap

4. Initially start up the hotspot after system boot up

Add the hotspot enabling commands in /etc/rc.local

8. LoRaWAN Module over Mini-PCIe 2

8.1 Module Installation



EG500 reserved another mini-PCIe socket for LoRaWAN module which develop from Semtech SX1301, SX1302 solution. Users are allowed to install a LoRaWAN module into mini-PCIe socket 2 as drawing blue area add-on module position, only one M2x5 screw is needed.

1. Insert the LoRaWAN module into Mini-PCIe socket 2 and, connect the antenna for getting better signal.
2. Log in the system via console use admin/admin.

8.2 Driver installation

1. **Download the source code from Elastel GitHub**

   git clone https://github.com/Elastel/sx1302.git

2. **Installation**

   $ cd sx1302

   $ chmod +x install.sh

   $ ./install.sh

Notice:

- During the install.sh execution, SCP will be used, keyGen will be generated, and a dialog will be displayed. Press Enter
- When you run install.sh, you may be asked to enter the password. The default user name and password are admin and admin.

3. **Check the process status**

   $ systemctl status loragw.service

   ● loragw.service - lorawan gw

   Loaded: loaded (/lib/systemd/system/loragw.service; enabled; vendor preset: enabled)

   Active: active (running) since Tue 2022-08-02 12:32:18 BST; 5min ago

   Main PID: 27741 (gwstart.sh)

   Tasks: 5 (limit: 1830)

   CPU: 23.425s

   CGroup: /system.slice/loragw.service

   ├ –27741 /bin/sh /home/admin/sx1302/bin/gwstart.sh

   └─27753 ./lora_pkt_fwd

   Aug 02 12:36:51 Elastel gwstart.sh[27753]: ##### 2022-08-02 11:36:51 GMT #####

   Aug 02 12:36:51 Elastel gwstart.sh[27753]: ### [UPSTREAM] ###

   Aug 02 12:36:51 Elastel gwstart.sh[27753]: # RF packets received by concentrator: 0

   Aug 02 12:36:51 Elastel gwstart.sh[27753]: # CRC_OK: 0.00%, CRC_FAIL: 0.00%, NO_CRC:

0.00%

Aug 02 12:36:51 Elastel gwstart.sh[27753]: # RF packets forwarded: 0 (0 bytes)

Aug 02 12:36:51 Elastel gwstart.sh[27753]: # PUSH_DATA datagrams sent: 1 (122 bytes)

Aug 02 12:36:51 Elastel gwstart.sh[27753]: # PUSH_DATA acknowledged: 0.00%

Aug 02 12:36:51 Elastel gwstart.sh[27753]: ### [DOWNSTREAM] ###

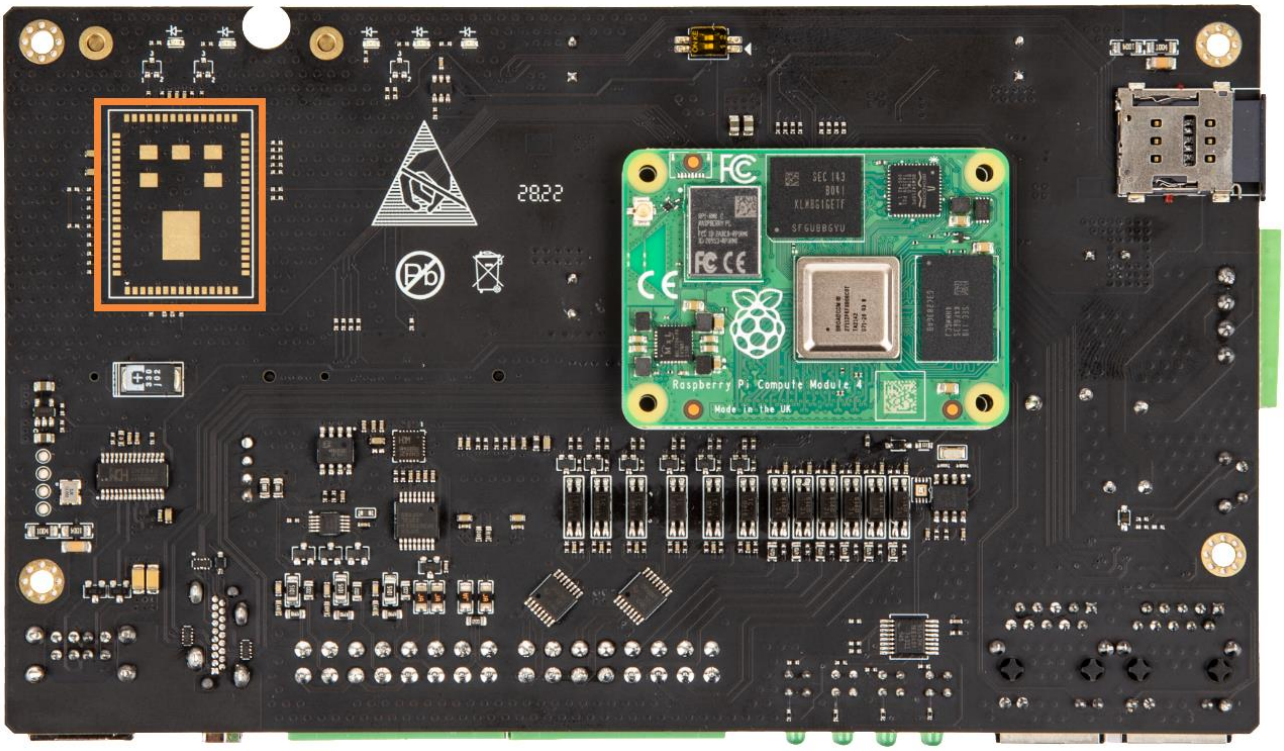
Aug 02 12:36:51 Elastel gwstart.sh[27753]: # PULL_DATA sent: 3 (100.00% acknowledged)

Aug 02 12:36:51 Elastel gwstart.sh[27753]: # PULL_RESP(onse) datagrams received: 0 (0 bytes)

Note, all the bin files and configuration files were installed in /home/admin/sx1302/bin directory, please edit the “global_conf.json” configuration file for LoRa frequency change, gateway address modification, and other settings.

9. WiFi HaLow

9.1 Module Installation



EG500 IoT gateway support 802.11ah WiFi (also named WiFi HaLow) network powered by SX-NEWAH module from SilexTechnology, the first industrial IEEE 802.11ah Wi-Fi module that operates in the Sub 1GHz band.

Visit SX-NEWAH to learn more benefit of WiFi HaLow technology in IoT.

EG500 carrier board reserved the corresponding mount points for wearing SX-NEWAH module. Check the above drawing orange area for module position. You are allowed to mount SX-NEWAH by yourself. Attention! You may need special weld tool and skill to mount SX-NEWAH on EG500 by yourself. Incorrect operation may destroy the carrier board and SX-NEWAH.

Purchase EG500 WiFi HaLow version from Elastel directly would be highly recommended.

https://www.elastel.com/product/industrial-cellular-router/eg500-wifi-halow-gateway/

9.2 Driver Installation

```
$ cd ~
$ git clone https://github.com/Elastel/halow-wifi.git
$ cd halow-wifi
$ sudo chmod +x install.sh
$ sudo ./install.sh
$ sudo reboot
```

9.3 How to Start in AP Mode

- Sample command to start in AP mode
  ① sudo insmod /lib/modules/$(uname -r)/nrc.ko fw_name=nrc7292_cspi.bin bd_name =nrc7292_bd.dat hifspeed=16000000
  ② sudo ifconfig wlan0 192.168.200.1
  ③ sudo ifconfig wlan0 up
  ④ cli_app set txpwr 17
  ⑤ cli_app set maxagg 1 8
  ⑥ cli_app set gi short
  ⑦ sudo hostapd -B ~/halow-wifi/conf/US/ap_halow_sae.conf

9.4 How to Start in STA Mode

- Sample Command to Start in STA Mode
  ① sudo insmod /lib/modules/$(uname -r)/nrc.ko fw_name=nrc7292_cspi.bin bd_name=nrc7292_bd.dat hifspeed=16000000
  ② sudo ifconfig wlan0 192.168.200.2
  ③ sudo ifconfig wlan0 up
  ④ cli_app set txpwr 17
  ⑤ cli_app set maxagg 1 8
  ⑥ cli_app set gi short

⑦ sudo wpa_supplicant -i wlan0 -D nl80211 -B -c ~/sx-newah/conf/US/sta_halow_sae.conf

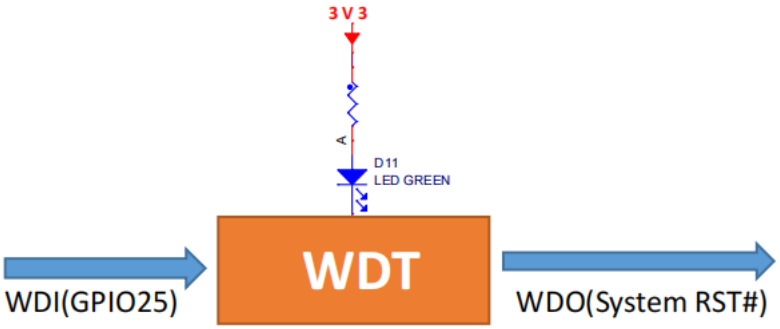
NOTICE:

When ”wpa_state=COMPLETED” is displayed after "sudo wpa_cli -i wlan0 status" is executed, AP connection is completed.

10. WDT (Watch Dog Timer)

10.1 Block Diagram

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



10.2 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. Go to 2.

11. RTC (Real Time Clock)

11.1 Block Diagram

The chip of RTC is PCF8563 from microchip. It is mounted on the system I2C bus.

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

11.2 Configuration

Adding the dtoverlay parameter to the /boot/config.txt file.

```
# Uncomment some or all of these to enable the optional hardware interfaces
dtparam=i2c_arm=on
dtoverlay=i2c-rtc,pcf8563
```

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.

12. Bluetooth

Below example describe the process of setting up a Bluetooth connectivity.

1. Execute bluetoothctl command to operate Bluetooth feature.

   $ sudo bluetoothctl

2. Enable the agent on to scan and discovery other Bluetooth devices.

   [bluetooth]# agent on

   [bluetooth]# scan on

The return result example:

[bluetooth]# scan on

Discovery started

[CHG] Controller DC:A6:32:05:7F:06 Discovering: yes

[NEW] Device 51:B8:16:6A:6F:C6 51-B8-16-6A-6F-C6

[NEW] Device 40:23:43:3F:4E:58 BRAVIA 4K UR2

The third column of the results are the MAC address of other Bluetooth devices, you will use it to pair with them. The Fourth column is the device name of other Bluetooth devices.

3. Use pair command to connect with it from one of the discovery result list.
   pair [XX:XX:XX:XX:XX:XX]

4. The fist time pair, it will connect with it immediately. If the connectivity disconnected since Bluetooth device out of range, we need use “connect” command to reconnect it.
   connect [XX:XX:XX:XX:XX:XX]

5. Use “trust” command to make the Bluetooth device be trusted after first time pair.
   trust [XX:XX:XX:XX:XX:XX]

-END-