

Industrial Raspberry Pi IoT Gateway

EG410

User Manual



Copyright

Copyright © XIAMEN ELASTEL TECHNOLOGIES CO., LTD. All rights reserved.

Trademark

ELASTEL logo is a registered trademark of Xiamen Elastel Technologies Co., Ltd. All other trademarks belong to their respective vendors or manufactures.

Disclaimer

Product specifications and information in this document are subject to change without any notice, and ELASTEL reserves the right to improve and change this user guide at any time. Users should take full responsibility for their application of products, and Xiamen Elastel Technologies Co., Ltd. disclaims all warranties and liability for the accurateness, completeness of the information published.

Revision History

Revision	Date	Changes
0.1	2022-03-10	Created
1.0	2023-08-02	Optimized

Contents

1. Product Introduction	4
1.1 Overview	4
1.2 Features	4
1.3 Specifications	4
1.4 Hardware Interfaces and Dimension	6
2. Installation	7
2.1 Mounting	7
2.2 Connectors and Interfaces	7
2.2.1 Power supply	7
2.2.2 Serial Port (RS232 and RS485)	8
2.2.3 DI & DO	8
2.3.4 HDMI	9
2.3.5 Ethernet	9
2.3.6 USB HOST	9
2.3.7 Console (USB typeC)	9
2.3.8 LED	9
2.2.9 SMA Connector	10
2.3.10 NANO SIM card slot	10
2.3.11 Mini-PCIe	10
2.2.12 M.2 Connector	11
3. Drivers and Programming Interfaces	11
3.1 LED	11
3.2 Serial Port (RS232 and RS485)	12
3.3 Cellular over Mini-PCIe (Optional)	12
3.4 WDT	16
3.4.1 Block Diagram of WDT	16
3.4.2 How it works	16
3.5 RTC	17
3.5.1 Block Diagram of RTC	17
3.5.2 How it works	17
4. Electrical specifications	17
4.1 Power consumption	17
Technical Support	18

1. Product Introduction

1.1 Overview

The Elastel EG410 is an industrial IoT Edge Gateway designed for IoT applications. Powered by Raspberry Pi Compute Module 4 (CM4) with our rugged, industrial-grade carrier board, and whole Aluminum fanless cooling technology, it's the Raspberry Pi IoT solution for you apply in Industrial Gateway, Automation Controller, Data Acquisition and Edge Computing...

1.2 Features

- Powered by Raspberry Pi industrial compute module 4 (CM4), up to 8GB RAM and 32GB eMMC
- Built-in mini-PCIe sockets for 4G LTE/3G/2G/NB-IoT cellular module, Lora or Zigbee
- 1x Gigabit Ethernet, built-in dual-band 2.4G/5G WiFi
- Isolated DI/DO/RS232/RS485 interfaces
- Embedded watchdog for work stability
- Aluminium Fanless enclosure for rugged structural and wide operating temperature -25~70°C
- Fully compatible with Raspbian, OpenWRT, Ubuntu OS, etc.

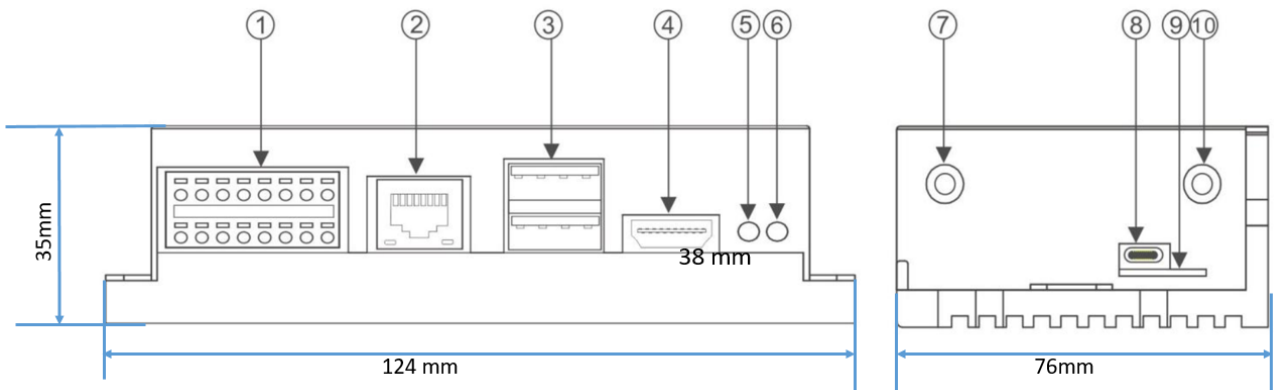
1.3 Specifications

Hardware platform	
CPU	Broadcom BCM2711, Quad-core A72 (ARM v8)@ 1.5 GHz
Memory	2GB (2GB/4GB/8GB optional) LPDDR4
FLASH	8GB (16GB/32GB optional) eMMC
Network & Interfaces	
Ethernet	1x Gigabit Ethernet, (1-WAN+1-LAN or 2-LAN Configurable)
Cellular	4G LTE, 3G, 2G, NB-IoT, CAT-M1 Via mPCIe socket
Wi-Fi	Dual Band 2.4GHz & 5GHz + Bluetooth 5.0 from CM4
LoRaWAN	Supported (Optional, mPCIe slot for LoRaWAN module)
GPS	Cellular Module built-in supported (Optional)
SIM	1.8 V/3 V; drawer-type Nano card holder × 1
	15KV ESD Protection

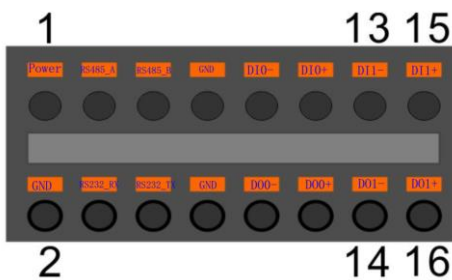
Antenna	4G: inside SMA x 1、WiFi/Buletooth: inside SMA x 1, LoRa/WiFi Halow: inside SMAx1
Industrial Serial Port	RS-232 x 1, RS-485 x 1;
	RS-232 signal: TXD, RXD, GND; RS-485 signal: A, B, GND; ESD protection: 15KV
I/O	2-channel digital input DI (0..24VDC, Configurable Status/Count mode)
	2-channel digital/pulse output DO (0..60V, Max. power 5fficiency: 500 mA)
USB	USB 2.0 x 2 for peripherals,
	USB-C x 1 for Console
HDMI	HDMI 2.0 x 1, (Up to 4kp60 video & audio output)
Additional drive	1 x M.2 socket with 2280 NVME SSD support
Power	
Power supply	Wide Range Voltages 9~36V DC, recommended 24V/1.5A
Power Terminal	Unpluggable industrial terminal connection
Power consumption	Less than 10 W (system)
Mechanical features	
Protection	IP30
Housing	Aluminum Heatsink
Cooling	Fanless cooling
Dimensions (cm)	124mm x 76mm x 35mm
Installation	DIN-rail, wall mounting
Weight	480g
Ambient temperature and humidity	
Storage temperature	-30 ~ 75°C
Ambient humidity	5 ~ 95% (non-condensing)
Operating temperature	-25 ~ 70°C (industrial grade)
EMC index	
Static electricity	level 3
Radiated electric field	level 3
Surge	level 3
Others	
Watchdog	Standalone Hardware Watchdog
Warranty	Standard 12 Months
Software Features	
OS	Raspbian / Optimized OpenWRT
Configuration	WebUI, Local CLI, Remote configuration
Upgrade operate	WebUI, Local USB upgrade and remote upgrade (optional)
Application features	Data Acquisition, Modbus RTU, VPN, Router, Firewall, Media Player... Python, Node-RED, Docker..., Support Secondary Development with SDK

Network Features	WWAN and WAN Failover, Load Balance, PPP, PPPoE, SNMP v1/v2c/v3, TCP, UDP, DHCP, RIPv1/v2, OSPF, BGP, DNS, DDNS, Modbus RTU/TCP, Siemens S7, OPC UA, HTTP, MQTT, ARP, QoS, SNTP, Telnet, SSH
------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

1.4 Hardware Interfaces and Dimension



①	Multi-Func phoenix connector	⑥	LED1
②	RJ45 Ethernet	⑦	SMA antenna 1 for 3G/4G LTE
③	USB 2.0 x 2	⑧	Console(USB type C)
④	HDMI	⑨	SIM card slot
⑤	LED2	⑩	SMA antenna 2 for Wi-Fi

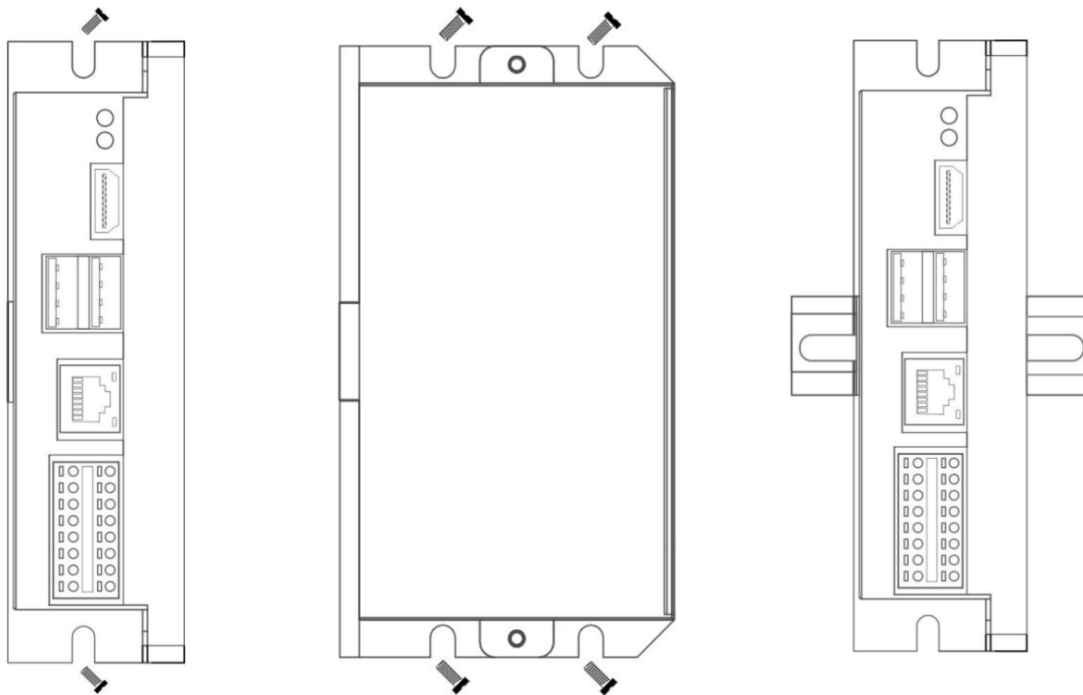


Func name	PIN #	PIN#	Func name
POWER	1	2	GND
RS485_A	3	4	RS232_RX
RS485_B	5	6	RS232_TX
RS485_GND	7	8	RS232_GND
DI0-	9	10	DO0_0
DI0+	11	12	DO0_1
DI1-	13	14	DO1_0
DI1+	15	16	DO1_1

2. Installation

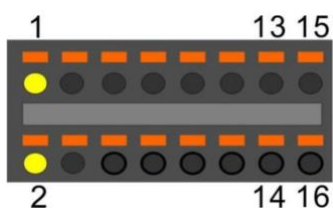
2.1 Mounting

The EG410 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

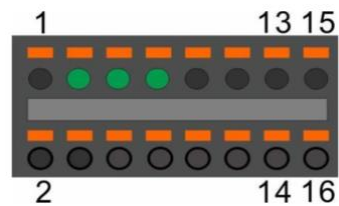
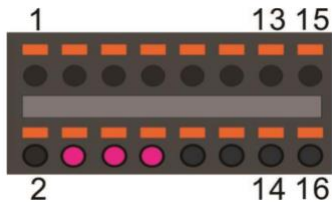
2.2.1 Power supply



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

Note, 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)



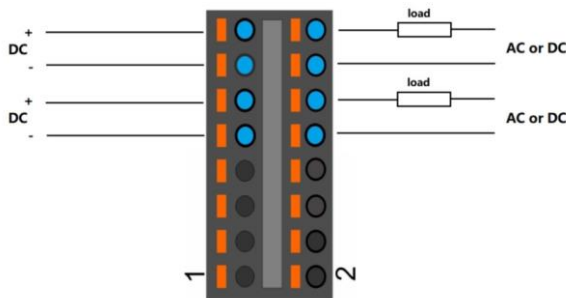
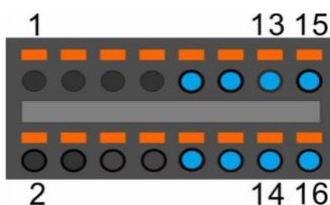
Pin#	Signal	Description
4	RS232_RX	RS232 receive line
6	RS232_TX	RS232 transmit line
8	GND	Ground (Reference potential)

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
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:

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

2.3.4 HDMI

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

2.3.5 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.

2.3.6 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.

2.3.7 Console (USB typeC)

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 “admin” and password is “admin”.

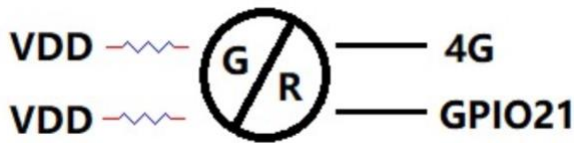
2.3.8 LED

EG410 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.



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-PCIe socket. The ANT1 is default used for Mini-PCIe socket 4G LTE cellular module, and ANT2 is for Internal WI-FI signal from CM4 module. Which you can find the info from printed lable on panel.

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

2.3.10 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.

2.3.11 Mini-PCIe

EG410 cellular network is implemented with a mini-PCIe cellular module. To setup EG410 for cellular functionality, install a cellular module into mini-PCIe socket add-on module position, only one M2x5 screw is needed.

You can also include cellular network function when place an order from Elastel, Elastel logistic team will preinstall that from factory.

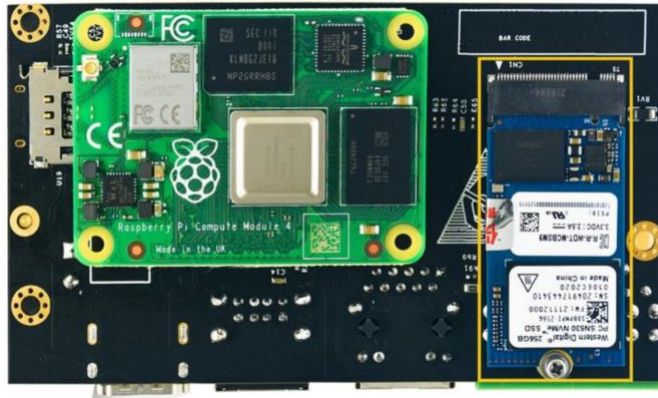
Elastel supplies EG410 with following cellular module options as default,

- 4G/LTE CAT4 cellular module, Quectel EC25-E (for EMEA/APAC bands)
- 4G/LTE CAT4 cellular module, Quectel EC25-AF (for NA bands)
- 4G/LTE CAT4 cellular module, Quectel EC25-AU (for ANZ/LATAM bands)

Other cellular modules are supported as well, please contact with Elastel representative if any specific requirements.

The mini-PCIe for 4G cellular modules driver have been prebuilt in OS, no need further configuration but just plug-and-play.

2.2.12 M.2 Connector



EG410 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 a 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/ ttyACM1` as RS232 port and `/dev/ ttyACM0` as RS485 port. Use RS232 as an example.

```
$ python
>>> import serial
>>> ser=serial.Serial('/dev/ttyACM1',115200,timeout=1)
>>> ser.isOpen()
true
>>> ser.isOpen()
>>> ser.write('1234567890')
10
```

3.3 Cellular over Mini-PCle (Optional)

Use Quectel EC25 as an example and following 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 PCIe
AND
$ cd gpio5
$ echo out > direction
$ echo 1 > value # release the reset signal of Mini PCIe
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
NOTE: 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/ttyACM0 'USB Dual_Serial'
--- 2: /dev/ttyACM1 'USB Dual_Serial'
--- 3: /dev/ttyAMA0 'ttyAMA0'
--- 4: /dev/ttyUSB0 'Android'
--- 5: /dev/ttyUSB1 'Android'
--- 6: /dev/ttyUSB2 'Android'
--- 7: /dev/ttyUSB3 'Android'
--- Enter port index or full name:
$ miniterm /dev/ttyUSB5 115200
```

Some useful AT command :

```
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

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
```

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
```

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).

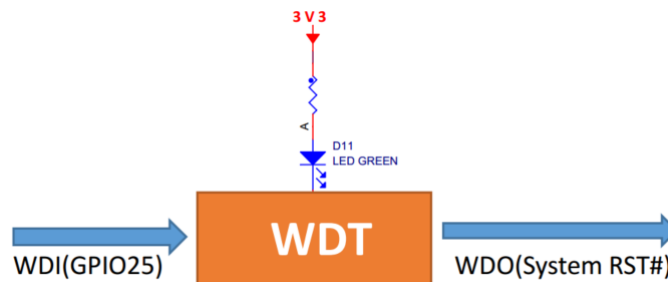
AT+COPS? The current operators and their status and allows to set automatic or manual network selection.

AT+CSQ indicates the received signal strength <rss> and the channel bit error rate <ber>

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.

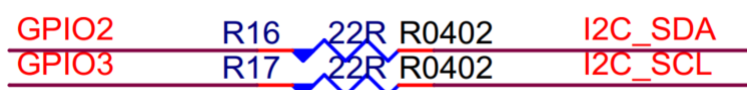
3.4.2 How it works

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 Block Diagram of RTC

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 How it works

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.

4. Electrical specifications

4.1 Power consumption

The power consumption of the EG410 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 EG410:

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 &

Technical Support

Send Email to Elastel Support center (support@elastel.com) for firmware upgrade, product documents, FAQ, technical support and more.

-END-