QConnect

Developer Guide

Document #: 1180-3007, Revision: C

SENSITIVITY LEVEL **YELLOW**

Contents

Revision History

DATE	REV	ECO#	REASON
9/25/2018	A	02499	Initial release
11/3/2021	B	03587	Multiple updates to reflect standard model's features
6/5/2023	C	03950	Addition of: • 1180-510x specification • RS-232 Configuration • Cellular Interface Content ○ APN file usage • Harness Pin Numbers for Linux GPIOs • BlueZ Bluetooth Stack usage • GPIO memory register configuration • Linux Service Descriptions Improved readability

Introduction

The QConnect Development Kit is a next-generation heavy equipment telematics IoT solution, enabling the development of applications for operation, worldwide. This kit provides a complete development environment for vehicle-to-cloud applications and services.

Designed to demonstrate advanced functionality, the QConnect terminal provided with this Development Kit provides maximum flexibility to facilitate easy IoT deployment and reduction in overall risk.

The Development Kit features the QConnect terminal incorporating a global LTE Cat-1 modem module. A dedicated processor provides execution of user applications with 128MB of storage for applications/data and 128MB of DDR2 system memory. QConnect also offers a rich set of embedded system peripherals including GNSS, dual-band WLAN, accelerometer, real-time clock, ADCs and a variety of general purpose I/O.

User application code runs directly on the processor, leveraging Quake's Linux application development environment. A Software Development Kit (SDK) specific to QConnect provides the necessary packages to access hardware peripherals and system resources. Application code built with the SDK is loaded into the processor through Ethernet or RS232C interface on the development board eliminating the need for external proprietary JTAG cables.

Note: The QConnect terminal provided with this development kit contains optional components that may not be available for specific configurations. Contact Quake Global for more information.

Block Diagram and Features

This section summarizes the features of QConnect, followed by functional descriptions.

Features

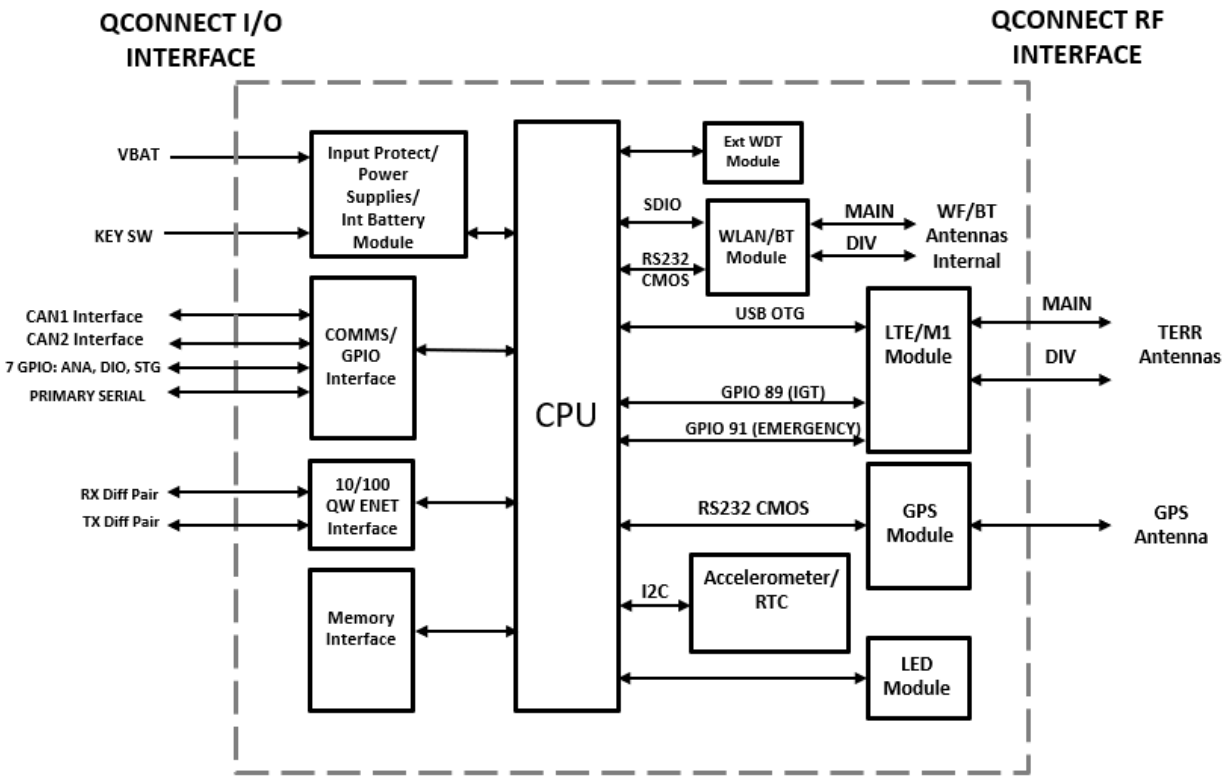
The following functions are supported.

Functions	Base 1180-5100	Fully Loaded 1180-5102
528MHz Processor	X	X
Linux Operating System	X	X
128 MByte DRAM	X	X
128 MByte NAND Flash	X	X
32 MByte NOR Flash	X	X
Hardware Watchdog	X	X
Cellular	X	X
CAN	X	X
Accelerometer 3-Axis	X	X
GNSS	X	X
Analog Input*	X	X
Digital IO*	X	X
WLAN / BLE		X
100Base-TX Ethernet		X
Internal Battery		X

*See TDS for specific details regarding GPIO's

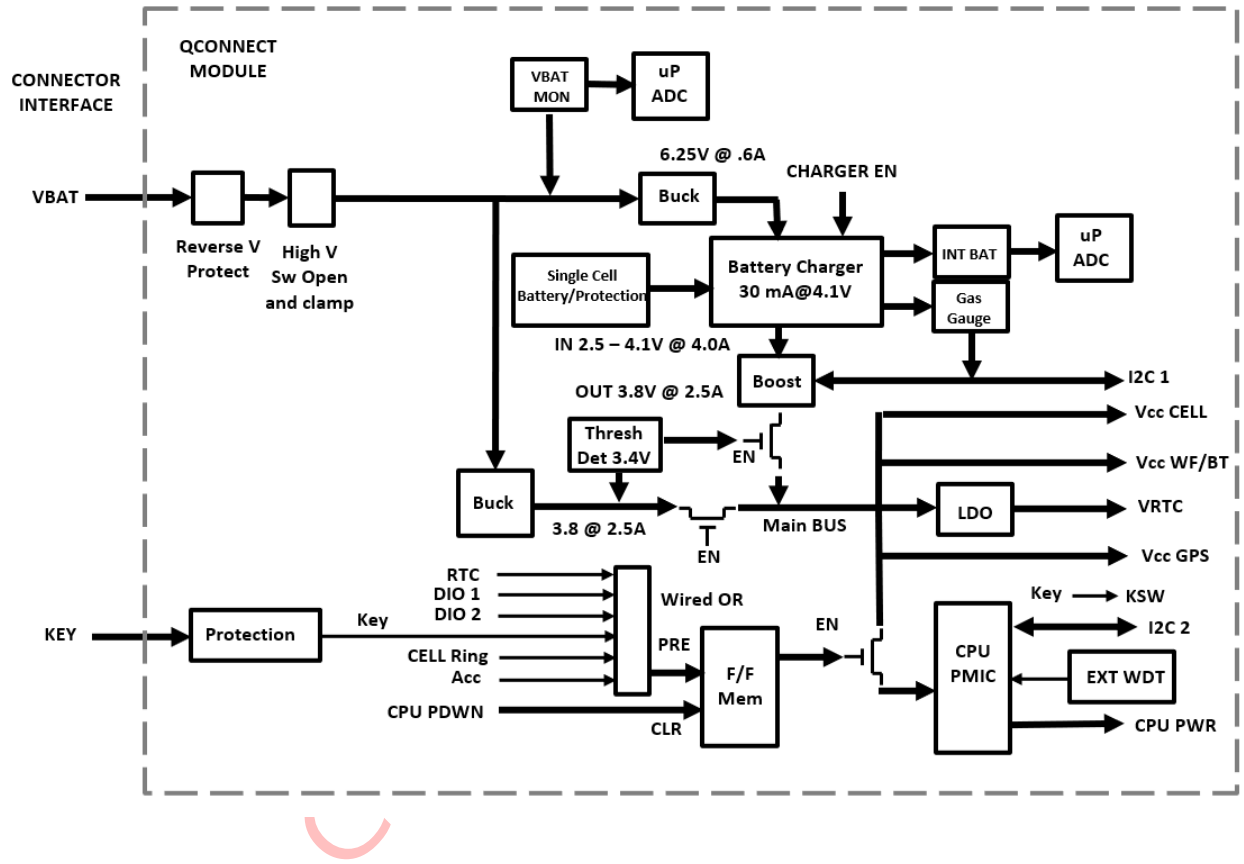
Block Diagram

This section describes the block diagram of the QConnect terminal.



Power Management

This section contains all the circuitry to power the unit and manage current consumption. The main supply steps down the VBAT input to a fixed voltage for the unit to safely run on. It has the internal and external battery (VBAT) control. Power is transferred from the external battery to the internal battery by the power rail switch automatically.



VBAT Signal

VBAT is the main input supply to the device and should be connected to 12/24V machine power. Generally, VBAT shall be supplied by the machine regardless of key position. This input provides load dump and transient protection with a series pass, reflective FET. The reverse voltage protection is provided by a high voltage series diode. VBAT input may be monitored with ADC VBAT.

Key switch

The key switch is designed to be active high when the machine is operating. This signal is protected from transients by diodes and can be used to wake the terminal from sleep. The unit will not start when the battery is first connected. Only when Key goes high unit will turn on.

Internal Battery

A 150mAhr internal battery is supplied and available on Fully Loaded. This internal battery is intended to provide limited, intermittent operation in the absence of VBAT input.

Internal Battery Charger

An internal battery charger is provided to charge the internal battery on Fully Loaded. It is designed to charge automatically, without CPU intervention, thus, whenever VBAT and Key input is present and valid, the battery will be charged until full if the Key switch is on. For QConnect, this duration is approximately 5 hours and automatically shuts off at full charge. The maximum charge rate is 32mA.

The CPU may choose to turn the charger ON or OFF while the CPU is running, however, the state of the charger enable pin (sysfs number: 30) is not persistent and when the CPU is 'OFF' the charger will revert to automatic charge mode.

The internal battery charger is not programmable.

Gas Gauge Battery Monitor

This device measures the real-time energy (W-Sec) stored in the battery as it charges. This total stored value is used to tell how much of this stored charge is left as the battery discharges and the energy monitored. A charge cycle form discharged to full charge will need to be run to get an accurate count on the stored energy.

The Battery Monitor IC with Coulomb Counter is an I2C device on CPU pins CSI_MCLK (F5) and CSI_PIXCLK (E5). This bus is I2C1.

Also, this device contains a temperature sensor. This sensor can be read over the I2C bus to estimate battery temperature. Battery temperature is read at 16-bit register addresses 10(LSB) and 11(MBS).

Internal Battery Protection

QConnect provides an internal device to protect the battery against low voltage. It will disconnect the battery below 2.5V. The current is also limited to 4A.

Power Down

Power down is supported in hardware by removal of VBAT and draining of the internal battery if installed. However, the power down of the terminal under CPU control is supported by the port assertion of CPU_PDWN. For CPU_PDWN to shut down the system the following signals must be low:

Signal	Description	Notes
ACC_WKUP	Accelerometer	Must be Low (Read Accelerometer to Clear)
CELL_WKUP	Cell Module	Must be Low
KEYSWITCH (KSW)	Key Switch	Must be Low

It is important to note that the Power Down condition places the CPU and memory in the off state but leaves the RTC, accelerometer and cellular modules on to act as wake-up sources. If these sources are to be used for wake up, the device should be set to low-power mode with the proper interrupt configuration set prior to powering down.

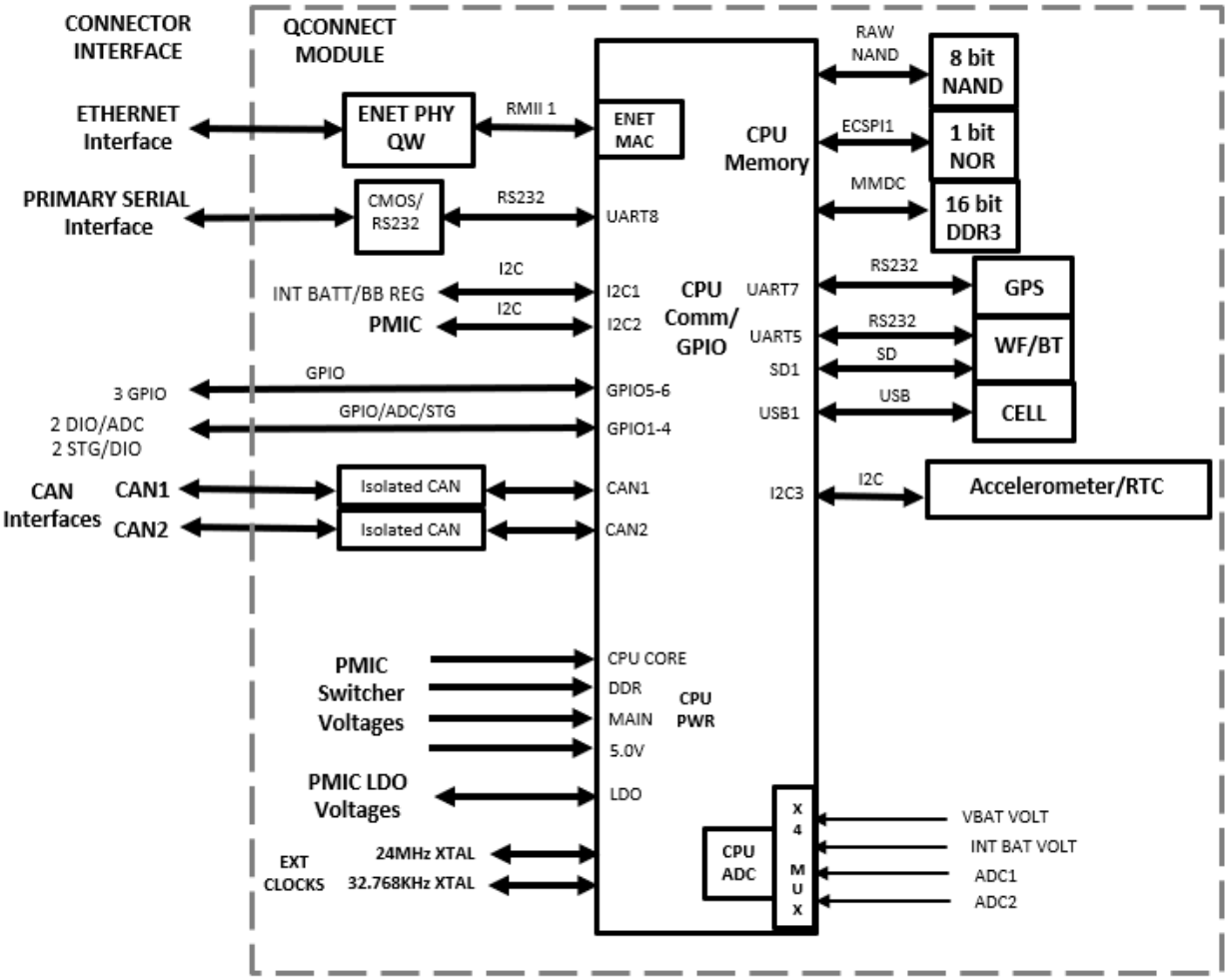
Wake on Event

Several sources may be used to wake the terminal from Power Down as illustrated in the table below. It is important to note that user-supplied Software will be required to poll the different wake sources to accurately identify the wakeup source.

Source	Notes
Cell SMS	Low-to-high transition. Non-maskable wake-up
RTC	Low-to-high transition. Non-maskable wake-up
ACCELEROMETER	Low-to-high transition. Non-maskable wake-up
DIO 1 or DIO 2	Low-to-high transition. Non-maskable wake-up
Key Switch	Low-to-high transition. Non-maskable wake-up

Microprocessor

This section provides all communication ports to the user and to internal modules. The external ports include a serial RS232 port, 10/100 Ethernet port and 2 CAN buses. All GPIO and ADC functions are controlled in this section. The DDR3, NAND, and NOR flash memories also reside in this section.



NAND Flash Memory

QConnect utilizes a 1Gbit (128MB) RAW ECC-corrected SLC NAND.

NOR Memory

QConnect utilizes a 32MB NOR-flash memory for boot and kernel storage.

DDR3 Memory

QConnect utilizes 1Gbit (128MB) DDR3 for main memory.

CPU performance

The CPU within QConnect is designed to address high performance computing requirements and use the temperature sensor to guard the processor against overheating.

CPU temperature is available via:

```
# /sys/class/thermal/thermal_zone0/temp
; Divide by 1000
```

Ethernet

QConnect provides a single 10/100 Ethernet Port over the RMII1 interface to external PHY. This interface is protected against ESD transients. The Ethernet Port is only available on Fully Loaded.

Ethernet Bring-up Sequence

The Ethernet Port may be brought up using the following command sequence, note that this sequence first brings the interface down to ensure an orderly power-up:

```
# echo 65 > /sys/class/gpio/export
# echo out > /sys/class/gpio/gpio65/direction
# echo 1 > /sys/class/gpio/gpio65/value
# ifup eth0
```

Ethernet Bring-down Sequence

Complete power-down of the ethernet interface may be accomplished with:

```
# ifdown eth0
# echo 0 > /sys/class/gpio/gpio65/value
# echo 65 > /sys/class/gpio/unexport
```

CAN Interfaces

QConnect provides two CAN bus interfaces and is capable of supporting data rates up to 500kbps.

CAN Access

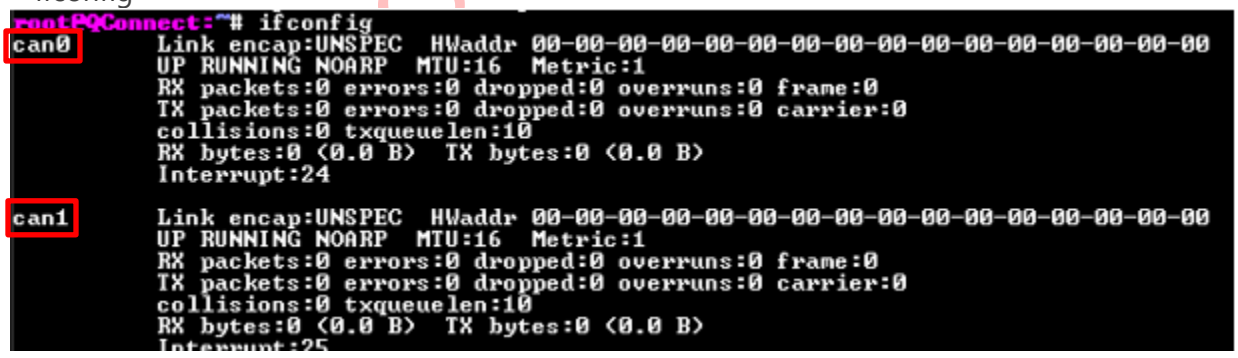
Ensure the Linux GPIOs corresponding to CAN are configured properly. This can be done with the following commands:

```
; [x] is 66(CAN0) or 67(CAN1)
# echo [x] > /sys/class/gpio/export
# echo out > /sys/class/gpio/gpio[x]/direction
# echo 0 > /sys/class/gpio/gpio[x]/value
```

If the CAN interfaces are down, they can be enabled with the following commands:

```
; [y] is 0 or 1
; [rate] is 125000, 250000, or 500000
# ip link set can[y] down
# ip link set can[y] type can loopback on
# ip link set can[y] type can bitrate <rate>
# ip link set can[y] up
```

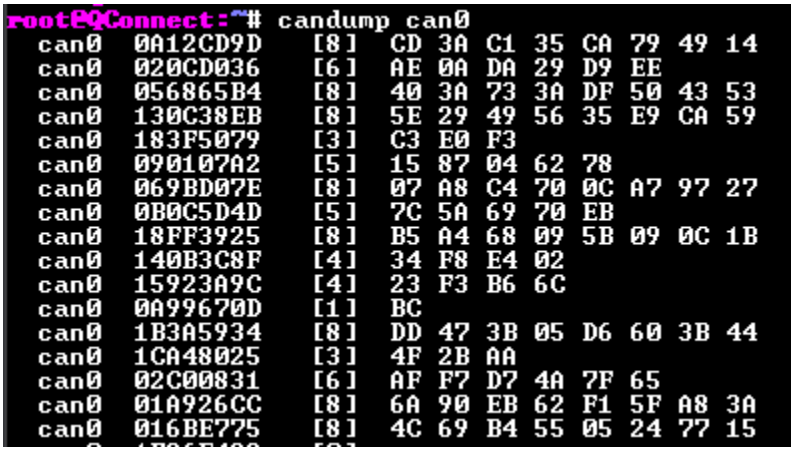
CAN interfaces can be checked with the following command:
ifconfig



It is possible to read CAN data with the following command after the specified CAN interface is brought up with the previous commands.

```
# candump can[y]
; For additional information on the command, you may use the -h flag
# candump -h
```

Example usage:



To simulate CAN data on the QConnect, you may use the following command:

```
; Where x is 0 or 1
# cangen can[x]
;For additional information on the command, you may use the -h flag
# cangen -h
```

Note: this command does not populate the physical CAN line(s) with CAN data. Therefore, if you were to send a CAN message with the above command and attempted to read it with another device, the other device would not read any signals from the pins associated with the CAN lines.

RS-232

PLEASE BE CAREFUL, IT IS POSSIBLE TO LOSE ACCESS TO THE UNIT IF YOU CHOOSE TO REPURPOSE THE SERIAL PORT, PLEASE PREPARE A BACKDOOR OR COUNTERMEASURE TO RETURN TO PORT 7 TO REGAIN SERIAL ACCESS TO THE UNIT.

WE STRONGLY RECOMMEND USING A **FULLY LOADED QCONNECT (1180-5002/1180-5102)** FOR SERIAL PORT CONFIGURATIONS RELATED TO SWITCHING THE ACTIVE PORT.

To repurpose the serial port; default is port 7:

```
; Where [x] can be set to 1, 7
# fw_setenv ttyactive [x]
# reboot
```

Example usage:

```
root@QConnect:~# fw_setenv tty_active 7
root@QConnect:~# reboot
```

To set the output of uboot for port 7 to silent:
Note: Port 1, ttymxc1 is set to silent by default and cannot be changed.

```
;Enable silent uboot output
# fw_setenv silent 1
# reboot

;Disable silent uboot output
#fw_setenv silent
# reboot
```

QConnect provides an RS-232 interface that is capable of supporting data rates up to 115200.

To configure the baud rate of RS-232 port, use the following commands:

```
; Where [y] can be set to 115200, 57600, 38400, 19200, 14400, 9600, 4800
# fw_setenv baudrate [y]
# reboot
```

Example usage:

```
root@QConnect:~# fw_setenv baudrate 9600
root@QConnect:~# reboot
```

To check the current baud rate that is set, use the following command:

```
# fw_printenv | grep baudrate
```

Example usage:

```
root@QConnect:~# fw_printenv | grep baudrate
baudrate=115200
```

Warning: Do not set Baud Rate to a speed over 115200. Serial Communication may become lost and unrecoverable if access to ethernet is not available.

Additionally, within the Developer Environment Virtual Machine, there is a file by the name of "SERIAL.h" which contains APIs that allow for additional configuration of the UART ports. The AUXPORT being the UART port related to the RS-232.

Notable APIs from the SERIAL.h file:

```
extern s16 SERIAL_openPort(UARTPORTS portNumber, u32 baudRate);
```

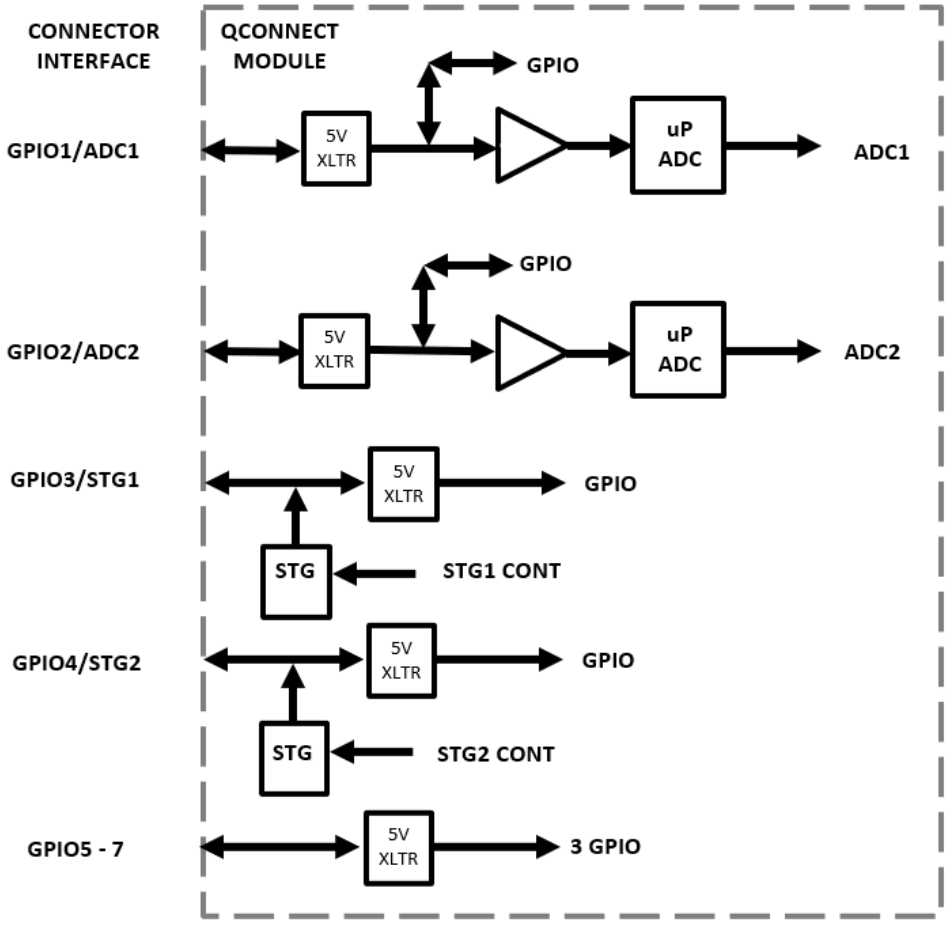
```
extern s16 SERIAL_closePort(UARTPORTS portNumber);
```

For additional information, please check the following file in the OVA:

/qconnect_sw/svc_sdk/source/inc/drivers/SERIAL.h

DIO/ADC/STG

QConnect has 7 GPIO ports that are available to the user. DIO are configured in hardware for 3.3V or 5V. GPIO 1 and 2 can be used as ADC. GPIO 3 and 4 can also be configured as high voltage and current short to ground (STG) inputs or outputs.



DIO/ADC

All GPIO ports are protected for ESD and +/- 32 volts.

DIOs are configured as CMOS 5 volt I/O.

- The GPIO drive level is very low, ≤ 30 µA for full CMOS level.

Input/Output Configuration

The GPIO memory registers are accessible via the following i2cset commands:

```
; To set all GPIOs as an output
# i2cset -y 1 0x20 7 0xFF

; To set all GPIOs as an input
# i2cset -y 1 0x20 7 0x00
```

To set specific GPIOs as both inputs and outputs, you must determine the HEX value using the table below to set the appropriate DIOs to the correct configuration.

Direction	Empty	In/Out	In/Out	In/Out	In/Out	In/Out	In/Out	In/Out
Bit Number	7	6	5	4	3	2	1	0
DIO Number	Empty	DIO7	DIO6	DIO5	DIO4	DIO3	DIO2	DIO1

Example usage:

```
; Sets DIO2 and DIO4 to outputs, all other DIOs to inputs
# i2cset -y 1 0x20 7 0x0A
```

Where **0x0A** in Binary is:
0 0 0 0 1 0 1 0

Where:

0 is **Input**

1 is **Output**

GPIO Direction	Input	Input	Input	Input	**Output**	Input	**Output**	Input
Bit Value	**0**	**0**	**0**	**0**	**1**	**0**	**1**	**0**
Bit Number	7	6	5	4	3	2	1	0
DIO Number	Empty	DIO7	DIO6	DIO5	DIO4	DIO3	DIO2	DIO1

After the memory registers are properly configured, the Linux GPIOs associated with the DIOS must be configured properly to match the memory register(s).

GPIOs are accessible via `/sys/class/gpio/`:

```
; [x] is Linux GPIO number. See Linux GPIO Number
# echo [x] > /sys/class/gpio/export
```

```
; [direction] is "in" or "out"
; [x] is Linux GPIO number
# echo [direction] > /sys/class/gpio/gpio[x]/direction

; Write value (for cases when direction is "out")
; [x] is Linux GPIO number
; [y] is 0 (Low) or 1 (High).
# echo [y] > /sys/class/gpio/gpio[x]/value

; Read value
;[x] is Linux GPIO number
# cat /sys/class/gpio/gpio[x]/value
```

ADC Access

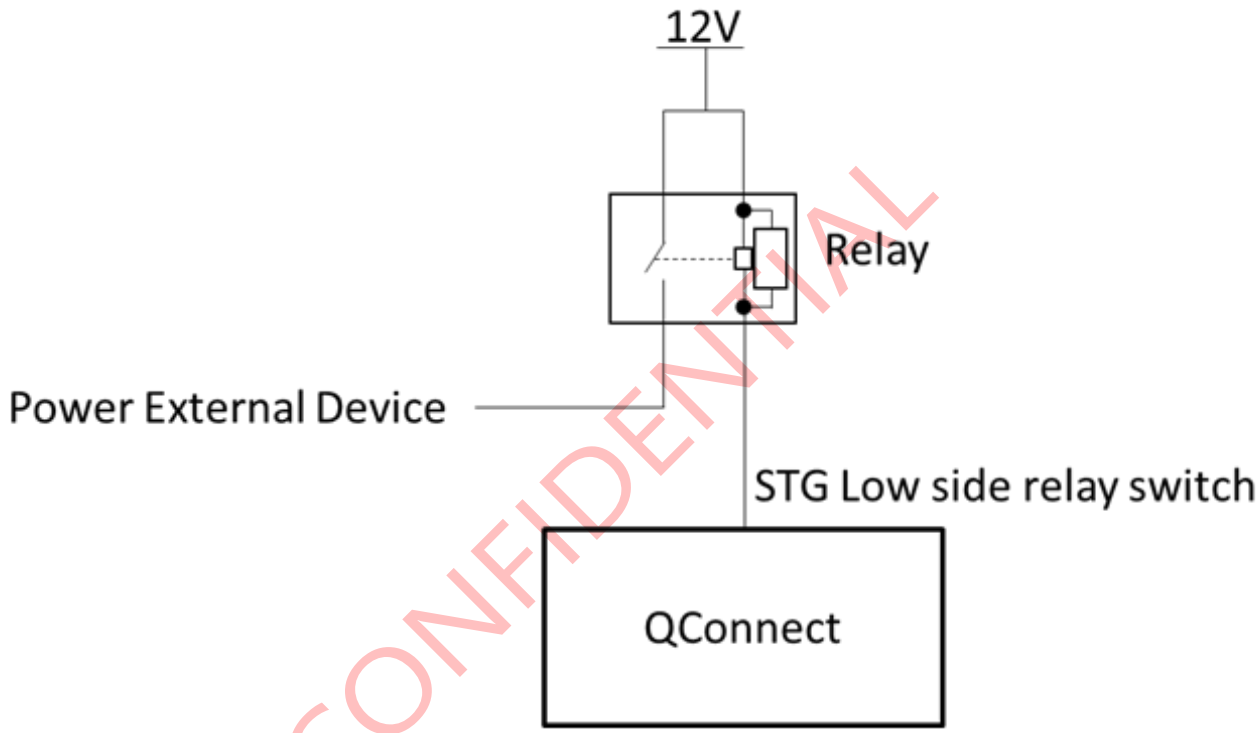
ADC values are accessible as follows:

```
; [x] is Linux GPIO number. See Linux GPIO Number
# cat /sys/bus/iio/devices/iio:device0/in_voltage[x]_raw

From the above raw value, you can caluculate the input value with the following formula;
ADC1/2 = (in_voltageX_raw*3.3)/4096*1.1
Internal Battery = (in_voltageX_raw*3.3)/4096*1.5
VBAT= A*in_voltageX_raw^5 - B*in_voltageX_raw^4 + C*in_voltageX_raw^3 - D
*in_voltageX_raw^2 + E*in_voltageX_raw
#define A (0.126359)
#define B (1.1752)
#define C (4.06304)
#define D (6.42402)
#define E (15.5354)
```

Short-To-Ground

QConnect contains 2 ports that are designed specifically to switch loads such as relay coils. GPIO 3 and 4 provide a Short-To-Ground (STG) functionality to facilitate these applications. These ports are protected against high current transients and reverse voltage application. External over-current protection is required for applications that may expect a direct connection to VBAT or other high-voltages when the switch is activated.

Example)

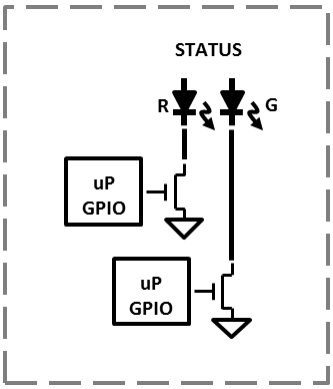


Linux GPIO Number

Harness Pin	Linux GPIO Number	Function	Notes
3	64	DIO1	
3	8	ANA1	
2	25	DIO2	
2	9	ANA2	
7	110	DIO3	
7	69	STG	
6	28	DIO4	
6	70	STG4	
4	84	DIO5	
21	78	DIO6	
18	77	DIO7	
8	112	KSW	
N/A	5	Internal Battery	Only valid on Fully Loaded
N/A	3	Vehicle Battery Voltage	

LED Section

This section provides visual feedback on the unit status. It has a red and green LED. For a third color, both red and green are turned on to for an orange appearing color.



Setting the LEDs is controlled by GPIOs, see DIO/ADC/STG for more information on setting GPIO values. LED color is controlled as follows:

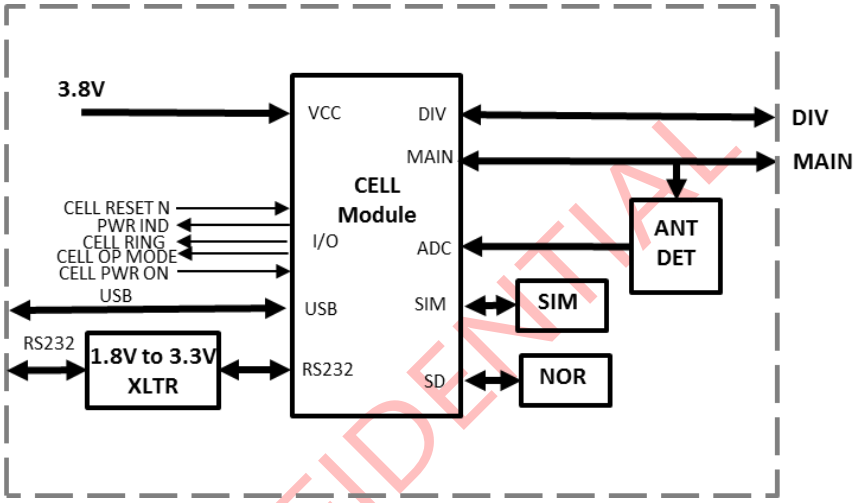
Color	GPIO85 (value)	GPIO86 (value)
Green	1	0
Red	0	1
Orange	1	1

Cellular Interface

This section provides connections to the global cellular network. The main network is LTE CAT 1. The fallback networks are 3G and 2G GPRS. There are both main and diverse receive channels available.

The application can set up messages at set intervals. Network availability and the SIM card will determine which mode the module uses.

The unit can stream video through the Ethernet port over LTE CAT 1 networks.



The cellular module is pre-certified with FCC. Quake provides a document with authorized antennas to be used with QConnect. Refer to Quake document 1180-3005, YELLOW, GUIDELINE, RECOMMENDED ANTENNAS, QCONNECT.

Cell Number

To request the CNUM or Phone Number of a QConnect use the following command:

```
# echo -e 'at+cnum\r' | microcom -t 2000 /dev/ttyACM2
```

Example usage:

```
root@QConnect:~# echo -e 'at+cnum\r' | microcom -t 2000 /dev/ttyACM2
At+cnum

+CNUM: "","+423650890856",145

OK
```

IMEI Number

To request the IMEI of a QConnect use the following command:

```
# echo -e 'at+cgsn\r' | microcom -t 2000 /dev/ttyACM2
```

Example usage:

```
root@QConnect:~# echo -e 'at+cgsn\r' | microcom -t 2000 /dev/ttyACM2
at+cgsn

358244083278525

OK
```

CCID

To request the CCID of a QConnect use the following command:

```
# echo -e 'at+ccid\r' | microcom -t 2000 /dev/ttyACM2
```

Example usage:

```
root@QConnect:~# echo -e 'at+ccid\r' | microcom -t 2000 /dev/ttyACM2
at+ccid

+CCID: 89423100180049585 77

OK
```

SIM Readiness

To request whether the cell module recognizes whether it has a SIM card, use the following command:

```
# echo -e 'at+cpin?\r' | microcom -t 2000 /dev/ttyACM2
```

Example usage:

```
root@QConnect:~# echo -e 'at+cpin?\r' | microcom -t 2000 /dev/ttyACM2
at+cpin?

+CPIN: READY

OK
```

The cell interface can be brought up using the following sequence, please note that this sequence brings the interface down initially to ensure an orderly power-up.

Cellular Bring-up Sequence

```
# echo 89 > /sys/class/gpio/export
# echo out > /sys/class/gpio/gpio89/direction
# echo 1 > /sys/class/gpio/gpio89/value
```

Then, on the serial port, you will see the following log and under /dev, you will find ttyACM ports*;

```
cdc_acm 2-1:1.0: ttyACM0: USB ACM device
cdc_acm 2-1:1.2: ttyACM1: USB ACM device
cdc_acm 2-1:1.4: ttyACM2: USB ACM device
cdc_acm 2-1:1.6: ttyACM3: USB ACM device
cdc_acm 2-1:1.8: ttyACM4: USB ACM device
```

Port (ttyACM)	Used by
0	PPPD
1	qsvc_terr (Quake Service)
2	Available for customer use
3	Available for customer use (PLS-63 only)
Other	Not supported

* ttyACM0 port is used by PPPD. ttyACM1 port is used by qsvc_terr. ACM2 and ACM3 (PLS-63 only) ports are available for the customer application. Other ports are not supported.

```
; To access AT command mode,
; use the following command where [x] is 0 to 3
# microcom /dev/ttyACM[x]
```

Example

AT+CGDCONT	Define PDP Context
AT+CSQ	Signal quality
AT+CREG	Network Registration Status
AT+CEREG	Network Registration Status (LTE)
AT+COPS	Operator Selection
AT+CCID	USIM Card Identification Number

Ctrl+x to exit from the AT command port

Cellular Bring-down Sequence

To bring the interface completely down, use the following sequence;

```
; Power off the cell module
# echo -e 'at^smso\r' | microcom -t 2000 /dev/ttyACM2
```

Then, on the serial port, you will see the following log and under /dev, you will NOT find ttyACM ports;

```
cdc_acm 1-1:1.4: acm_ctrl_irq - usb_submit_urb failed: -19
cdc_acm 1-1:1.4: failed to set dtr/rts
```

```
; to bring down completely
# echo 0 > /sys/class/gpio/gpio89/value
# echo 89 > /sys/class/gpio/unexport
```

Cellular time

To get the network time automatically updated, the followings shall be configured:

AT+CTZU=1	This will Enable automatic time zone update to RTC via NITZ
AT^SIND=nitz,2	Gets UTC when Cell is registered, then sets the RTC by network time
AT+CCLK?	Queries the network time from RTC

Establishing a Cell Connection

Only one of the two approaches is needed to establish a Cell Connection. If one approach is used, then the other can be disregarded. However, both approaches are listed for developer preference.

PPPD Daemon

```
;To bring up a PPP connection, use the following command:
        # /etc/start_pppd.sh
```

```
;To check the logs of the daemon to troubleshoot or verify connection
        # tail -F /var/log/pppd/current
```

```
;To stop the pppd daemon
        # sv stop pppd
```

```
;To check the status of the ppp daemon
        # sv status pppd
```

;To restart the ppp daemon
sv restart pppd

;To start the pppd daemon, if stopped with the sv command above
sv start pppd

Linux Utility

Alternatively, you can use the following Linux command to attempt a cell connection. This approach has less logging but is native to Linux.

;To attempt to create a ppp connection
pon

;To terminate the attempt to create a ppp connection
poff -a

;To verify if an attempt to create a ppp connection is ongoing
ps | grep ppp

Example usage:

;To view the logs associated with this cell connection approach
tail -F /var/log/messages | grep ppp

Example usage:

Verify the Cell Connection

;To verify your cell connection, ensure ppp0 is listed as an interface
ifconfig

```
root@QConnect:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 2A:68:B5:A0:99:06
          inet addr:192.168.1.133  Bcast:0.0.0.0  Mask:255.255.255.0
          inet6 addr: fe80::2868:b5ff:fea0:9906/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:336 errors:0 dropped:0 overruns:0 frame:0
          TX packets:318 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:34570 (33.7 KiB)  TX bytes:49242 (48.0 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:4862 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4862 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:290077 (283.2 KiB)  TX bytes:290077 (283.2 KiB)

ppp0      Link encap:Point-to-Point Protocol
          inet addr:10.239.209.99  P-t-P:10.64.64.64  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:265 errors:0 dropped:0 overruns:0 frame:0
          TX packets:320 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:22683 (22.1 KiB)  TX bytes:25704 (25.1 KiB)
```

Cellular Troubleshooting

;If a ppp connection cannot be established check that the APN is set correctly
echo -e 'at+cgdcont?\r' | microcom -t 2000 /dev/ttyACM2

;Otherwise, check if the registration status is valid
echo -e 'at+creg?\r' | microcom -t 2000 /dev/ttyACM2

;To soft reset the cell module
echo -e 'at+cfun=1,1\r' | microcom -t 2000 /dev/ttyACM2

;To enable additional error codes and messages
echo -e 'at+cmee=2\r' | microcom -t 2000 /dev/ttyACM2

Antennas

FAKRA antenna inputs are on the front housing are for use by external antennas. The required FAKRA mate is a Female key code D with color Claret Violet.

Female	Male	Keying Code	Color	RAL	Single way application in		Multi way application in USCAR	TE PN typical 1-7-1
					Europe	USCAR		
		D	Claret Violet	4004	GSM	GSM	Not defined	n-3

Antenna detection is available for special external antennas on the main channel through AT^SRADC. These antennas must provide a 10K resistor to ground to enable this function.

```
(Example)
; Write <1: Open ADC channel> <0: Single measurement mode>
AT^SRADC=1,0

; Return <1: Number of measured samples> <546: Measurement value>
^SRADC: 1,546

; Measurement value thresholds
```

<365	Shorted
365 - 771	Connected
>772	Open

Note: The above numbers are based on actual testing of the modems with the antenna using a 10K resistor that had 1% tolerance. Please keep in mind your thresholds may be a little different based on the resistor tolerances in your antenna. If you are using the same tolerance resistor on your antenna (1%) as Quake, the tolerances above will be fine.

SIM card

QConnect provides a standard push-push SIM holder for 4FF nano SIMs. These SIMs must be factory installed to ensure IP67 ingress protection is maintained. Chip SIMs are also available, contact Quake for more information.

Sending an SMS from the QConnect

Enter AT Command mode:

microcom /dev/ttyACM2

Set to text mode:
at+cmgf=1

Send SMS
at+cmgs="001<CNUM of QConnect 1>"
OR
at+cmgs="+<CNUM of QConnect 1>"

Write message
type message body

Send the message
ctrl + z

Exit AT Command mode:
ctrl + x

Changing the APN

The APN of the QConnect can be changed by:

1. Deregistering from the network with the following AT command:
   - AT+COPS=2
2. The use of the following AT command, change bolded portion to the APN name of your sim card: (e.g. "soracom.io")
   - AT+CGDCONT=1, "IP", "**APN**"

   Example usage:
   - AT+CGDCONT=1, "IP", "soracom.io"
3. Restarting the cell module:
   - AT+CFUN=1,1

To ensure the APN is recognized by the cell module, use the following command:
echo -e 'at+cgdcont?\r' | microcom -t 2000 /dev/ttyACM2

Changing the APN (via /etc/chatscripts/apn file)

Note: as of OSv2.2.2 the apn file is deactivated by default, we encourage you to change the APN using the first method above, however, to accommodate older apn change implementations, the following instructions demonstrate how to reenable the apn file for use.

1. Navigate to the /etc/chatscripts directory:

   # cd /etc/chatscripts

2. Redirect the softlink of the quake-gemalto file to point to the manual version.

   # ln -sfn quake-gemalto-manual quake-gemalto

   # ls -al

   Before:

   ```
   quake-gemalto -> quake-gemalto-auto
   ```

   After:

   ```
   quake-gemalto -> quake-gemalto-manual
   ```

3. At this point, the apn file should be active, now deregister from the network:

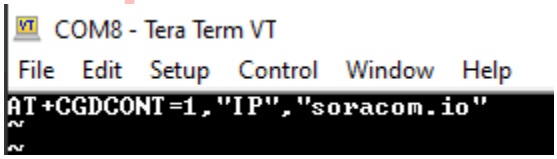
   From AT command mode:
   AT+COPS=2

   OR

   From command line:
   echo -e 'at+cops=2\r' | microcom -t 2000 /dev/ttyACM2

4. Edit the apn file and change the section in red to the apn of your choice:

   # vi /etc/chatscripts/apn

   - AT+CGDCONT=1,"IP","<apn_name>"

5. Reset the cell module:

   From AT command mode:
   AT+CFUN=1,1

   OR

   From command line:
   echo -e 'at+cfun=1,1\r' | microcom -t 2000 /dev/ttyACM2

Communication between Customer Application and Cell module

After power-up or restart ensure that the cell module is in ready state before trying to send any AT command or data.

Application is coupled with the cell module via a receive and a transmit line. Since both lines are driven by independent devices collisions may (and will) happen. For example, if the application issues an AT command and the cell module starts sending a URC*. This will probably cause the application to misinterpret the URC being part of the AT command's response.

To avoid this conflict the following measures must be taken:

- If an AT command is finished (with "OK" or "ERROR") the application always wait at least 100 ms before sending the next one. The pause between two AT commands gives the cell module the opportunity to the transmission of pending URCs and get necessary service.

  Normally cell module responds in 10 seconds.
  However, if the command is to deal with network, i.e. registration, attach or scan, it's recommended to give a longer timeout, e.g.: 60 seconds.
  It is also recommended the application check the response is OK or expected value.
  If the response is not what you expected, you can retry to send command.

- The application shall communicate with the cell module using activated echo (ATE1), i.e. the cell module echoes characters received from the application. Hence, when the application receives the echo of the first character "A" of the AT command just sent by itself it has control both over the receive and the transmit paths.

*Unsolicited Result Code: a report message issued by the cell module without being requested by the application, i.e. a URC is issued automatically when a certain event occurs.

Wake from Sleep

The unit can be put in low power, wake on SMS from the application. Instructions on how to set the QConnect to wake on SMS:

Set the needed AT^SCFG configurations for the SMS wakeup (PLS-63)

echo -e 'at^scfg="MEopMode/ExpectDTR","current"\r' | microcom -t 2000 /dev/ttyACM2
echo -e 'at^scfg="MEopMode/ExpectDTR","powerup"\r' | microcom -t 2000 /dev/ttyACM2
echo -e 'at^scfg="URC/Ringline","asc0"\r' | microcom -t 2000 /dev/ttyACM2

Set to text mode:

echo -e 'at+cmgf=1\r' | microcom -t 2000 /dev/ttyACM2

Forward SMS to TE

echo -e 'at+cnmi=2,1\r' | microcom -t 2000 /dev/ttyACM2

Use character set UCS2

echo -e 'at+cscs="UCS2"\r' | microcom -t 2000 /dev/ttyACM2

Set to low power mode

echo -e 'at^spow=2,1000,3\r' | microcom -t 2000 /dev/ttyACM2

Physically pull out keyswitch pin

Put QConnect to sleep

poweroff

Send a text message to the QConnect

QConnect should power on from SMS

Automatic Shutdown

The module will automatically shut down if the temperature exceeds the limits of the extended temperature range of the operating range.

The automatic shutdown is equivalent to the power-down initiated with an AT command, i.e. LTE module logs off from the network and the software enters a secure state avoiding loss of data. Delay to shutdown from over temperature detection is 5 seconds.

The Internal Temperature of the LTE module can be monitored using the following AT Command.

AT^SCTM read command returns:
• The URC presentation mode.
• Information about the current temperature range of the LTE device.
• The board temperature (in degree Celsius) if parameter <tempCtrl>=1.

Syntax

Test Command
AT^SCTM=?
Response(s)
^SCTM:(list of supported <UrcMode>s)[, (list of supported <tempCtrl>s)]
OK

Read Command
AT^SCTM?
Response(s)
^SCTM: <UrcMode>, <UrcCause>[, <temp>]
OK
ERROR
+CME ERROR: <err>

Warning Messages

The LTE module provides several Unsolicited Result Codes (URCs). These URCs are enabled as follows:

- AT^SCTM=1 Presentation of URCs always enabled
- AT^SCTM=0 (Default) Presentation of URCs enabled ONLY during first 2 minutes of operation.

Sending temperature alert (2min after the cell module start-up, otherwise only if URC presentation enabled)
^SCTM_B: 1 Board close to overtemperature limit.
^SCTM_B: -1 Board close to under temperature limit.
^SCTM_B: 0 Board back to non-critical temperature range.
Automatic shutdown (URC appears no matter whether or not presentation was enabled)
^SCTM_B: 2 Alert: Board equal or beyond overtemperature limit. Cell module switches off.
^SCTM_B: -2 Alert: Board equal or below under temperature limit. Cell module switches off.

Table 1 LTE Module Thermal Related AT Commands

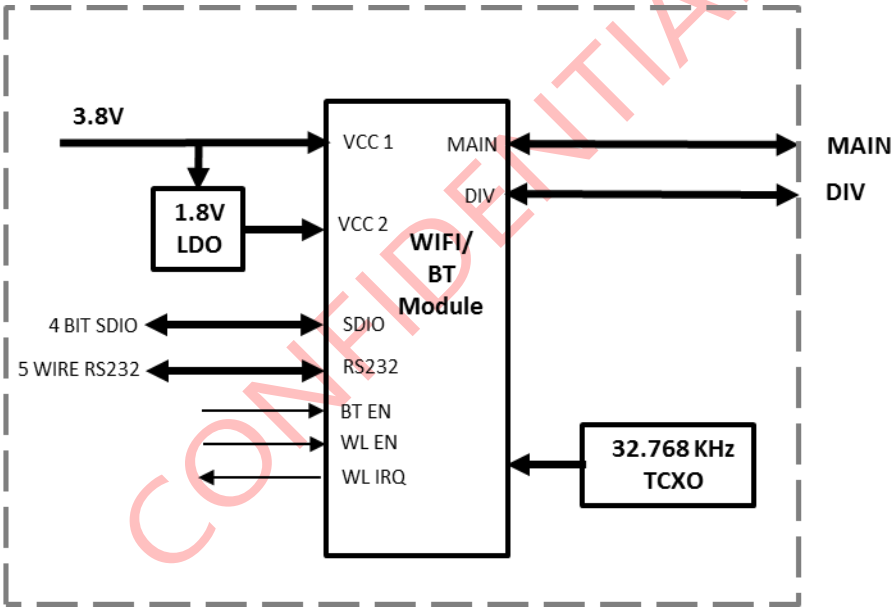
WLAN/Bluetooth Interface

QConnect is equipped with a WLAN/BLE module and is pre-certified with FCC and utilizes internal pre-approved antennas.

This module provides connectivity to WLAN and Bluetooth networks. The WLAN has full MIMO operation available and follows the 802.11a, b, g and n standards. Both 2.4 and 5G bands can be accessed. The 5G band utilizes switched diversity technology.

The BT shares the antenna with the main WLAN antenna port in the 2.4G band. This can operate concurrently with Cell. The device stack supports BLE 4.2 or higher.

The maximum supported data rate is 54Mbps and supports WLAN Direct, client mode and other, see the technical data sheet for more information.



WLAN Bring-up/down Sequence

To bring the WLAN Interface up or down, use the following command sequence.

Access Point

1. Hostapd and Udhcpd services must be removed from blacklist
   ; /usr/qk/blacklist
2. WPA service must be blacklisted

```
root@QConnect:~# cat /usr/qk/blacklist
wpa
qsvc_analog
qsvc_can
qsvc_gnss
qsvc_gpio
qsvc_ibat
qsvc_imu
qsvc_led
qsvc_terr
```

3. Configure hostapd at:/etc/network/hostadp.conf

```
##### IEEE 802.11 related configuration #######################################

# SSID to be used in IEEE 802.11 management frames
ssid=QConnect
# Alternative formats for configuring SSID
# (double quoted string, hexdump, printf-escaped string)
#ssid2="test"
#ssid2=74657374
#ssid2=P"hello\nthere"

# WPA pre-shared keys for WPA-PSK. This can be either entered as a 256-bit
# secret in hex format (64 hex digits), wpa_psk, or as an ASCII passphrase
# (8..63 characters) that will be converted to PSK. This conversion uses SSID
# so the PSK changes when ASCII passphrase is used and the SSID is changed.
# wpa_psk (dot11RSNAConfigPSKValue)
# wpa_passphrase (dot11RSNAConfigPSKPassPhrase)
#wpa_psk=0123456789abcdef0123456789abcdef0123456789abcdef0123456789abcdef
wpa_passphrase=password
```

   ; Configure SSID and password or allow for use of wpa
4. Configure udhcpd at: /etc/network/udhcp.conf
   ; Configure the number of addresses to use

Station Mode

1. Hostapd and Udhcpd services must be blacklisted

```
root@QConnect:~# cat /usr/qk/blacklist
hostapd
udhcpd
qsvc_analog
qsvc_can
qsvc_gnss
qsvc_gpio
qsvc_ibat
qsvc_imu
qsvc_led
qsvc_terr
```

2. WPA service must be removed from blacklist
3. Configure wpa at: /etc/network/wpa_supplicant.conf

```
ctrl_interface=/var/run/wpa_supplicant
##ap_scan=1

update_config=1
country=US

network={
ssid="QConnectNet"
proto=WPA2
key_mgmt=WPA-PSK
scan_ssid=1
psk="password"
#wep_key1=abcdeabcde
; Configure ssid, protocol, key, password
```

Changing MAC address

```
# fw_setsenv serial flag 1
# fw_setenv wlanaddr <new MAC address>
; Check mac address
# cat /etc/network/wlan_mac
or
# ifconfig
```

Bluetooth Bring-up/down Sequence

The Bluetooth Interface may be brought up or down using the following sequence.

```
"Bluetoothd" and "uim" services must be removed from blacklist
    ; /usr/qk/blacklist
```

Bluetooth Stacks and related Linux Services

The QConnect contains two bluestacks, the runit services related to each stack cannot run concurrently. Therefore, one set of Bluetooth service(s) should be run at a time.

Bluetopia

```
; the legacy Bluetooth stack
; Related service: "bluetopia"
; It is also possible to add or remove the above services from
the blacklist to disable/enable alternatively.
```

BlueZ

```
; the new Bluetooth stack encouraged by Quake Global
; Related services: "bluetoothd" , "uim"
; It is also possible to add or remove the above services from
the blacklist to disable/enable alternatively.
```

To access the BlueZ stack utilities:

```
; Access the bluetooth menu
# /opt/bin/bluetoothctl
; List the options
# help
```

Example Usage:

```
root@QConnect:/opt/bin# ./bluetoothctl
[NEW] Controller C8:DF:84:4A:DF:66 BlueZ 5.48 [default]
Agent registered
[bluetooth]# help
Menu main:
Available commands:

advertise                                Advertise Options Submenu
scan                                     Scan Options Submenu
gatt                                     Generic Attribute Submenu
list                                     List available controllers
show [ctrl]                              Controller information
select <ctrl>                            Select default controller
devices                                  List available devices
paired-devices                           List paired devices
system-alias <name>                      Set controller alias
reset-alias                              Reset controller alias
power <on/off>                           Set controller power
pairable <on/off>                        Set controller pairable mode
discoverable <on/off>                    Set controller discoverable mode
agent <on/off/capability>                Enable/disable agent with given capability
default-agent                            Set agent as the default one
advertise <on/off/type>                  Enable/disable advertising with given type
set-alias <alias>                        Set device alias
scan <on/off>                            Scan for devices
info [dev]                               Device information
pair [dev]                               Pair with device
trust [dev]                              Trust device
untrust [dev]                            Untrust device
block [dev]                              Block device
unblock [dev]                            Unblock device
remove <dev>                             Remove device
connect <dev>                            Connect device
disconnect [dev]                         Disconnect device
menu <name>                              Select submenu
version                                  Display version
quit                                     Quit program
exit                                     Quit program
help                                     Display help about this program
[bluetooth]#
```

```
; To scan for BLE devices
# /opt/bin/hcitool lescan
```

Example usage:

```
root@QConnect:~# /opt/bin/hcitool lescan
LE Scan ...
5F:F8:33:BE:C3:2C (unknown)
29:40:5B:EE:EB:0E (unknown)
20:A9:C1:BB:BD:3E (unknown)
6F:EC:94:54:B1:77 (unknown)
63:9F:1C:E2:86:8B (unknown)
72:A4:39:8A:D9:43 (unknown)
15:13:41:F1:FC:33 (unknown)
14:82:8C:99:A5:EF (unknown)
```

Known Limitations

Wi-Fi

Multiple SSIDs are not supported.

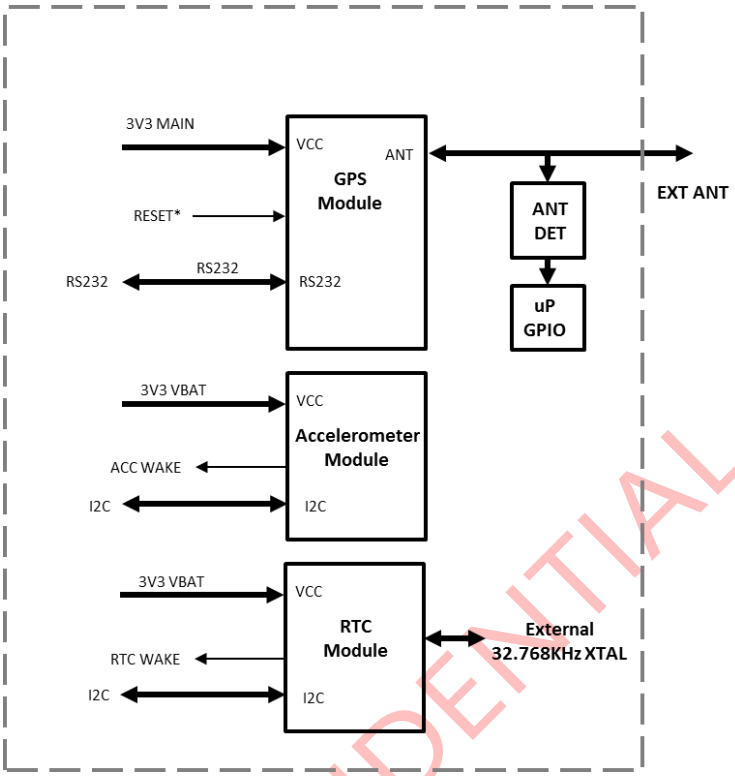
Bluetooth

After the update of a QConnect OS, kernel errors appear in the QConnect terminal which indicates an error related to the Bluetooth module. These errors should only appear on the following boot up from the update of the QConnect OS. This limitation is known to appear on Base model units (1180-5000/1180-5100) but can appear on Fully-Loaded units (1180-5002/1180-5102) if both the Bluetopia and BlueZ services are running concurrently.

Changing MAC address

```
# fw_setenv bleaddr <new BLE addr>
# fw_setenv serial flag 1
; Check mac address
# cat /etc/network/bleaddr
Or
# /opt/bin/SS1Tool cmd 0x04 0x00009
```

GPS/Accelerometer & RTC



Accelerometer

The accelerometer is available on I2C port 3. This component remains powered when VBAT is present and may be used to wake the device from power down.

Accelerometer Access

The accelerometer on 1180-5002 is accessible via the following sysfs:

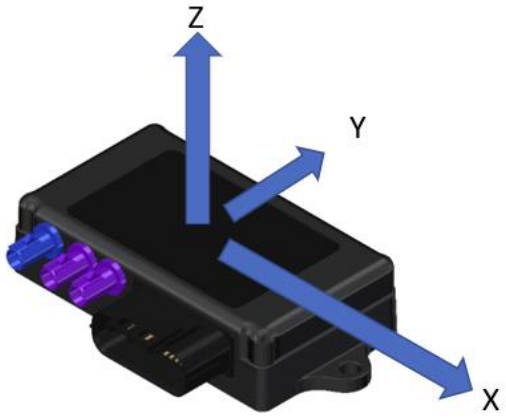
```
; [x] is 1 (Accel)
# cd /sys/class/input/input1/accel

; [y] is 0 (disable) or 1 (enable)
# echo [y] > enable

; [rate] is polling rate to be set
# echo [rate] > polling_rate

; [freq] is the sensors sampling frequency in Hz (Write only)
# echo [freq] > sampling_freq
; Available sampling frequencies of the device (Read only)
```

```
# cat sampling_freq_avail

; [scale] is the dynamic range of the data. (Write only)
; The value is set in "g" for Accel and "dps" for Gyro.
# echo [scale] > scale
; available scale settings of the device
# cat scale_avail

; [z] is 1 (Accel) or 2 (Gyro)
# cat /dev/input/event[z]
```

The accelerometer on the following models: 1180-5000, 1180-5100, 1180-5102 is accessible via the following sysfs:

```
# cd /sys/class/input/input1/device

; [x] is 0 (disable) or 1 (enable)
# echo [x] > enable_device

; [rate] is polling rate in ms
# echo [rate] > pollrate_ms

; [range] is the dynamic range of the data in "g".
# echo [range] > range

# cat /dev/input/event1
```

Directions of accelerations is as follows;



Tilt Access

Tilt functionality is provided by the driver as follows:

- Sysfs functions: enable
- Read wakeup values from /dev/input/event4

Real Time Clock

The real-time clock (RTC) is present on I2C port 3 and is powered when valid VBAT or internal battery input is supplied and can be used to wake the device from power down.

The RTC is accessible via:

```
/dev/rtc0
```

and accessed from userspace via `/sys/class/rtc/rtc0/`

```
; Reset RTC wakealarm
# echo 0 > /sys/class/rtc/rtc0/wakealarm

; Get RTC time since epoch
# cat /sys/class/rtc/rtc0/since_epoch

; Set RTC wakealarm. [x] is in second.
# echo [x] > /sys/class/rtc/rtc0/wakealarm

; Sync RTC clock to system time
# hwclock -w
```

GPS Module

The GPS Module provides concurrent, multi-constellation operation. Communication with the GPS module is performed over UART6 via the following:

/dev/ttymxc6

After bootup the gpsd service directs the data to: **/dev/gps0**

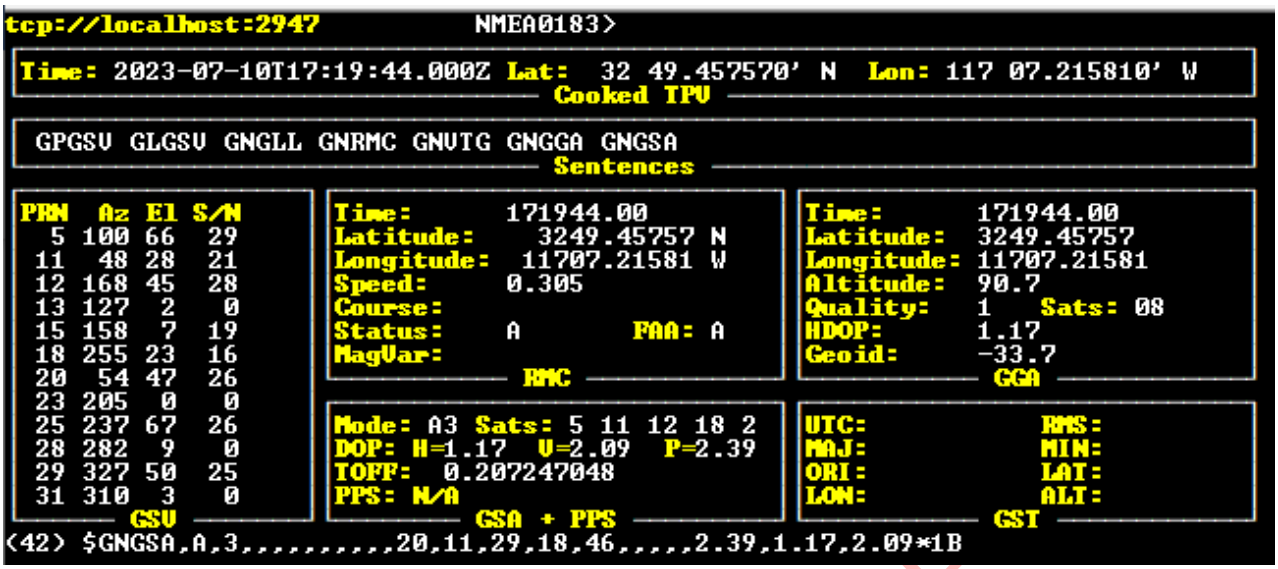
To view raw GPS data, use the following command:

```
# cat /dev/gps0
```

```
root@QConnect:~# cat /dev/gps0
$GNGGA,232715.00,3249.45778,N,11707.21328,W,2,09,0.98,91.3,M,-33.7,M,,0000*45
$GNGSA,A,3,10,18,27,23,24,08,32,46,48,,,,2.15,0.98,1.92*1C
$GNGSA,A,3,,,,,,,,,,,,,2.15,0.98,1.92*11
$GPGSV,3,1,12,08,19,316,31,10,61,316,42,15,10,043,24,16,08,249,18*77
$GPGSV,3,2,12,18,50,097,40,23,53,030,37,24,20,081,36,27,47,290,38*73
$GPGSV,3,3,12,29,00,158,09,32,44,199,39,46,50,201,38,48,51,194,38*7E
$GLGSV,1,1,00*65
$GNGLL,3249.45778,N,11707.21328,W,232715.00,A,D*6F
$GNRMC,232716.00,A,3249.45780,N,11707.21325,W,0.016,,190623,,,D*77
$GNVTG,,T,,M,0.016,N,0.030,K,D*3C
$GNGGA,232716.00,3249.45780,N,11707.21325,W,2,09,0.98,91.3,M,-33.7,M,,0000*4C
$GNGSA,A,3,10,18,27,23,24,08,32,46,48,,,,2.15,0.98,1.92*1C
$GNGSA,A,3,,,,,,,,,,,,,2.15,0.98,1.92*11
$GPGSV,3,1,12,08,19,316,30,10,61,316,38,15,10,043,25,16,08,249,*73
$GPGSV,3,2,12,18,50,097,37,23,53,030,36,24,20,081,33,27,47,290,34*7B
$GPGSV,3,3,12,29,00,158,12,32,44,199,38,46,50,201,35,48,51,194,36*76
$GLGSV,1,1,00*65
$GNGLL,3249.45780,N,11707.21325,W,232716.00,A,D*66
$GNRMC,232717.00,A,3249.45774,N,11707.21317,W,0.002,,190623,,,D*79
$GNVTG,,T,,M,0.002,N,0.004,K,D*3E
$GNGGA,232717.00,3249.45774,N,11707.21317,W,2,09,0.98,92.3,M,-33.7,M,,0000*44
$GNGSA,A,3,10,18,27,23,24,08,32,46,48,,,,2.15,0.98,1.92*1C
$GNGSA,A,3,,,,,,,,,,,,,2.15,0.98,1.92*11
$GPGSV,3,1,12,08,19,316,32,10,61,316,42,15,10,043,22,16,08,249,*7B
$GPGSV,3,2,12,18,50,097,41,23,53,030,39,24,20,081,38,27,47,290,38*72
$GPGSV,3,3,12,29,00,158,11,32,44,199,42,46,50,201,39,48,51,194,39*7B
$GLGSV,1,1,00*65
```

To view the GPS data in a human readable format, use the GPS utility with the following command:

```
# gpsmon
```



Antenna

A FAKRA antenna input is provided on the front housing for use by an external antenna. The required FAKRA mate is a Female key code C with color Blue.

Female	Male	Keying Code	Color	RAL	Single way application in		Multi way application in USCAR	TE PN typical 1-7-1
					Europe	USCAR		
		C	Signal Blue	5005	GPS telematics or navigation	GPS	GPS / GSM	n-3

The external GPS antenna is provided with 3.3V for use with an external active antenna. This interface is protected for ESD and transients.

External antenna detection is provided for detection of OPEN and SHORTED conditions via the serial interface.

Linux GPIO Number	Function	Notes
79	GPS_ANT_SHORT_N	When 1, GNSS antenna is shorted
80	GPS_ANT_OPEN	When 1, GNSS antenna is OPEN or proper antenna is not attached

Notable Linux Services

Dropbear

Dropbear is a software package written by Matt Johnston that provides a Secure Shell-compatible server and client. It is designed as a replacement for standard OpenSSH for environments with low memory and processor resources, such as embedded systems.

Kplex

kplex is a multitransport software data multiplexer which runs on GNU/Linux, Mac OS X, FreeBSD, NetBSD and OpenWRT. It works with data which conform to the NMEA-0183 standard but has been written without reference to any proprietary documentation not available in the public domain. Kplex multiplexes data inputs from sources such as serial lines, pseudo terminals and network interfaces hrefand send to any (reasonable) number of outputs. kplex can perform filtering of inputs (so you only get the data you want, or don't get the data you don't want from a given source) and outputs (so you only send what you want where you want) and can perform fine-grained failover so that for any given type of data, you specify a priority order of the source you would like to take it from.

Monit

Monit is a small Open Source utility for managing and monitoring Unix systems. Monit conducts automatic maintenance and repair and can execute meaningful causal actions in error situations.

GPSD

gpsd is a service daemon that monitors one or more GPSes or AIS receivers attached to a host computer through serial or USB ports, making all data on the location/course/velocity of the sensors available to be queried on TCP port 2947 of the host computer. With gpsd, multiple location-aware client applications can share access to supported sensors without contention or loss of data. Also, gpsd responds to queries with a format that is substantially easier to parse than the NMEA 0183 emitted by most GPS.

Chrony

Chrony is a versatile implementation of the Network Time Protocol (NTP). It can synchronize the system clock with NTP servers, reference clocks (e.g. GPS receiver), and manual input using wristwatch and keyboard. It can also operate as an NTPv4 (RFC 5905) server and peer to provide a time service to other computers in the network.

Mosquitto

Mosquitto is a broker for the MQTT protocol version 5.0/3.1.1/3.1.

Blacklist

; To stop all quake services and add them to the blacklist
/etc/service_control.sh quake stop

; To check the contents of the blacklist
cat /usr/qk/blacklist

; To edit the blacklist
vi /usr/qk/blacklist

; Enter write mode
i

; Write to the file

; Exit write mode
<esc key>

; Save and escape
wq

Profile Configuration

For customization of the Linux OS is possible to create a file called “profile” in the root directory that will allow for customization of Linux commands.

; Create the profile file
vi .profile
i
; Edit the file to look similar to the following

```
#!/bin/sh

alias cuqb='cat /usr/qk/blacklist'
alias mdt2='microcom /dev/ttyACM2'
```

; Create alias’ that meet your preferences
; When done editing
<esc key>
:wq

```
root@QConnect:~# chmod +x .profile
```

chmod +x .profile
; Log out of the QConnect to apply the changes
<ctrl+d>
; Log back into the QConnect
; Execute your alias commands

Example usage:

```
root@QConnect:~# cuqb
wpa
lighttpd
hostapd
udhcpd
bluetopia
qsvc_analog
qsvc_can
qsvc_gnss
qsvc_gpio
qsvc_ibat
qsvc_imu
qsvc_led
qsvc_terr
```

Directory /etc/startup

WARNING: please ensure that any executable or application is functioning properly and has a reliable method of termination before placing it into the /etc/startup directory. If a corrupt script or application is placed in the /etc/startup folder, it is possible to lock yourself out of the unit due to the files in this directory being automatically deployed before login on each boot up of the QConnect.

The folder /etc/startup is a folder in the QConnect that runs executable files on the startup of the QConnect. Scripts or customer applications can be placed here. However, due to the nature of when this folder is read, these executables will be run before Quake and/or Linux services are running. Therefore, adjustments to your executables must accommodate for the absence of both Linux and Quake services.

Example: Linux service, kplex, will not redirect GNSS data from ttymxc6 to gps0.

Usage

Logs

Logs will be saved in the below log file. Each file contains up to 2MB log.

```
/var/log/messages
; You may find each Quake service's log in its own directory.

/var/log/qsvc_terr/current
/var/log/qsvc_analog/current
/var/log/qsvc_gnss/current
.
.
.
```

Note: these logs will be cleared after reboot.

Mosquitto Broker

Mosquitto broker is configured to local interfaces only as default. If you want to access broker via ethernet or cell, the mosquitto configuration would need to be modified.
To change mosquitto configurations edit the /etc/mosquitto.conf file.

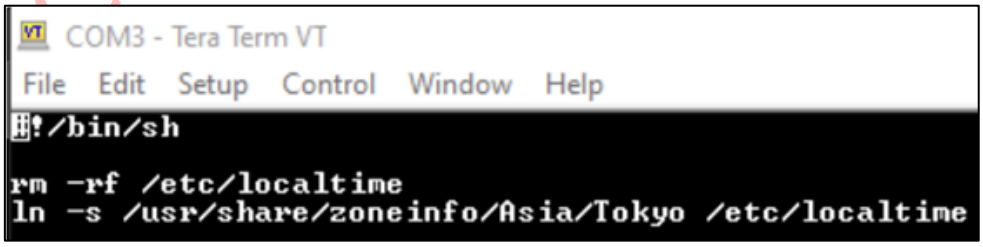
```
# ================================================================
# Default listener
# ================================================================

# IP address/hostname to bind the default listener to. If not
# given, the default listener will not be bound to a specific
# address and so will be accessible to all network interfaces.
# bind_address ip-address/host name
bind_address 127.0.0.1

# Port to use for the default listener.
#port 1883

# Bind the listener to a specific interface. This is similar to
# bind_address above but is useful when an interface has multiple addresses or
# the address may change. It is valid to use this with the bind_address option,
# but take care that the interface you are binding to contains the address you
# are binding to, otherwise you will not be able to connect.
# Example: bind_interface eth0
bind_interface lo
```

Add a "#" symbol in the front of "bind_address 127.0.0.1" and "bind_interface lo" to enable mosquitto connections from any interface.

Changing the time zone

There is a script in the /etc/startup folder called set_localtime.sh in the QConnect.
Edit this file and change the path to the time zone desired.

```
; Edit the file
# cd /etc/startup
# ls
# vi set_localtime.sh
```




```
; Change the path in bold after /zoneinfo/
; (Example) ln -s /usr/share/zoneinfo/Asia/Tokyo /etc/localtime

; Save and quit with:
(esc key)
:wq

; reboot
```

```
# reboot

; Check the current date
# date
```

Note: You may need to run the command date a few times before it shows the correct time. This is because the QConnect may need some time to get a proper fix before adjusting to the correct time.

```
; Check the current time zone:
# ls -l /etc/localtime
; (Example return) /usr/share/zoneinfo/America/Los_Angeles

; Check the time zone options
# cd /usr/share/zoneinfo
# ls
```


```
; Change directories to the country/continent desired
# cd Asia  ; Example
# ls
```

Static/Dynamic IP Address

/etc/network/interfaces file is a way to configure network interfaces. One of the common usages is to configure Static IP address to Ethernet or WLAN.

Example: Static IP address to Ethernet on QConnect

1. Edit /etc/network/interfaces (red)
    a. IP address: Configure the static IP address in the Class C range. (192.168.1.0 192.168.255.255)
    b. Subnet mask: Configure the Subnet Mask in address Class C. (255.255.255.0)

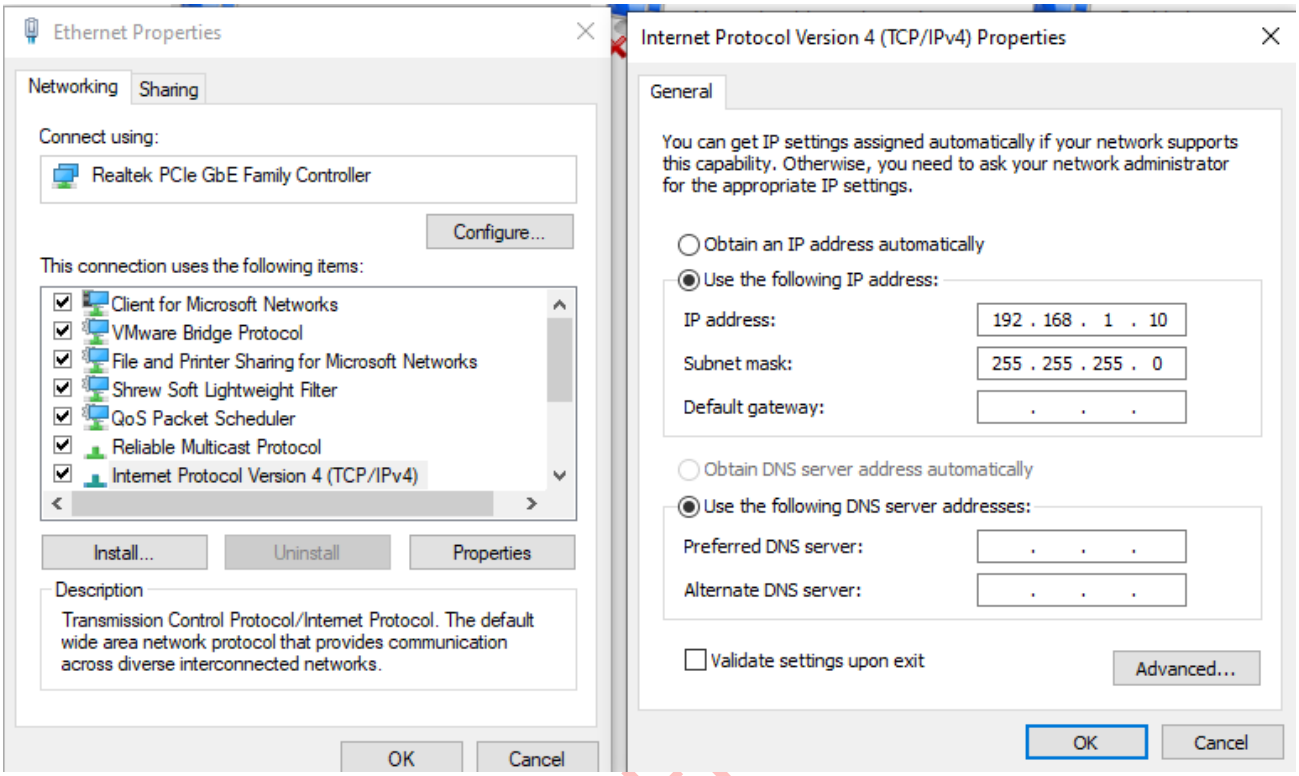
```
root@QConnect:~# cat /etc/network/interfaces
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.1.133
netmask 255.255.255.0
```

2. Turn on Ethernet (If it's already on, turn off first.)
3. Type `ifconfig` to see eth0 is configured as defined

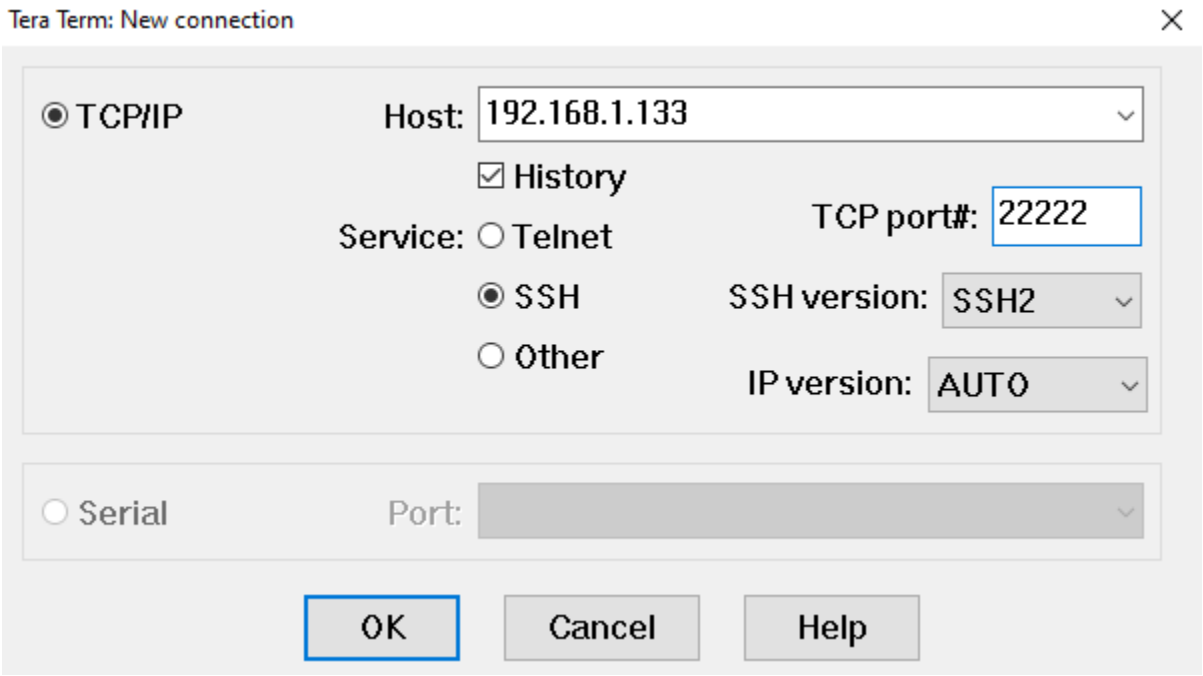
```
root@QConnect:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 2A:4D:36:26:B0:2B
          inet addr:192.168.1.133  Bcast:0.0.0.0  Mask:255.255.255.0
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:20 errors:0 dropped:0 overruns:0 frame:0
          TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2225 (2.1 KiB)  TX bytes:858 (858.0 B)
```

Connect from PC with Tera Term via Ethernet

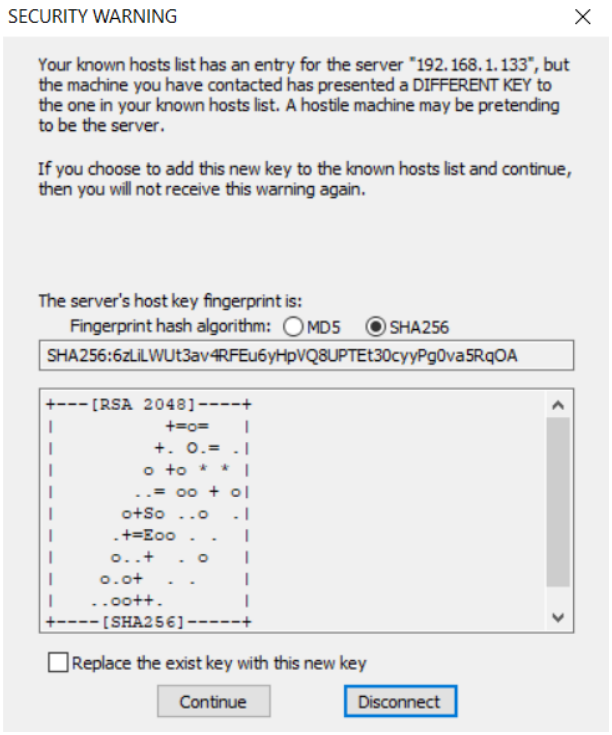
1. On Windows PC side, open Control Panel > Network and Internet > Network Connection and right click on the ethernet you will use.
2. Click " Internet Protocol Version 4(TCP/IPv4) IPv4)" and select "Use the following IP address " and "Use the following DNS server address ".
3. Set the below listed address;
    - IP address: Configure the static IP address with the same first three octet as the IP address in /etc/network/interfaces.
    - Subnet mask: Configure the Subnet Mask in address Class C. (255.255.255.0)



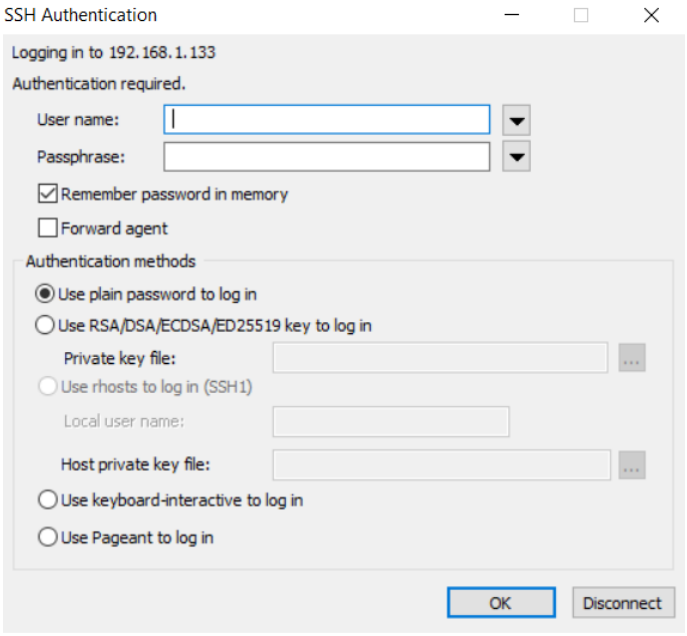
4. Open Tera Term and set Host address the IP address of QConnect and the TCP port# to **22222** then click "**OK**". Or Tera term > File > New Connection will lead you the below dialog box.



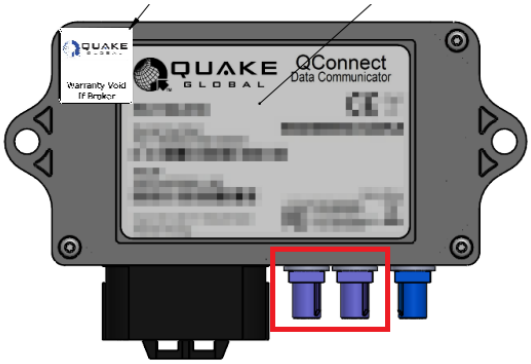
5. You may see the below security warning message. Please click "**continue**". \



6. In the SSH Authentication dialog box, please input username and passphrase as usual. (Linux login)



Troubleshooting

Trouble	Solution
No cell connection	• Check APN is correct Check APN in the APN file: `cat /etc/chatscripts/apn` See Changing the APN to edit. Check APN used by the cell module: `AT+CGDCONT?` Response: +CGDCONT: <cid>, <PDP_type>, <APN>, <PDP_address>, <data_compression>, <header_compression> Ex) `AT+CGDCONT?` `+CGDCONT: 1,"IP","soracom.io","",0,0` `OK` If incorrect, use the command; `AT+CGDCONT=1,"IPV4V6","<your APN>",0,0` • Check that cell antennas (Bordeaux) are connected.  • Check the registration status: `AT+CREG,AT+CEREG` Response: +CREG/CEREG: <URC Mode>, <status> Status: 0 Not registered, ME is not currently searching an operator to register to. 1 Registered, home network. 2 Not registered, but ME is currently trying to attach or searching an operator to register to. 3 Registration denied. 4 Unknown, e.g. out of LTE coverage 5 Registered, roaming. Ex)

	AT+CREG? +CREG: 0,0 OK AT+CEREG? +CEREG: 0,4 • Check the signal strength: AT+CSQ Response: CSQ: <RSSI>,< RXQUAL> Ex) AT+CSQ +CSQ: 99,99 OK If it's 99 or 0, the signal strength may not be good enough.
Cannot talk to the cell module	• Ports are locked When the ports are being used, ex) unit successfully establishes a PPP connection, it creates a "LCK..ttyACM0" file in /var/lock. This disables the user to send AT commands via microcom. In case no one use the port and you are still unable to send AT command, delete "LCK..ttyACM0" file. #microcom /dev/ttyACM0 microcom: can't create '/var/lock/LCK..ttyACM0': File exists #cd /var/lock #ls LCK..ttyACM0 dbus messages subsys chrony fw_printenv.lock sshd #rm LCK..ttyACM0 #microcom /dev/ttyACM0 • Reset ttyACM port In case the port become unavailable or unresponsive, you can force the cell module reset with the following steps; ;Enable access to the force cellular reset gpio #echo 91 > /sys/class/gpio/export #echo out > /sys/class/gpio/gpio91/direction

	`` ;Turn on the cellular reset gpio. Ports will be deleted #echo 1 > /sys/class/gpio/gpio91/value ;Turn off the force cellular reset gpio #echo 0 > /sys/class/gpio/gpio91/value ;Turn off the cellular access gpio #echo 0 > /sys/class/gpio/gpio89/value ;Turn on the cellular access gpio #echo 1 > /sys/class/gpio/gpio89/value ```
No PPP connection	• Check the cell connection • Check PPP logs: `tail -F /var/log/pppd/current` If PPPD is not running, run the following script; `/etc/start_pppd.sh`

Confidentiality Obligations

This document contains sensitive information and is classified "**CONFIDENTIAL**".

Its distribution is subject to the recipient's signature of a Non-Disclosure Agreement (NDA).

At all times you should comply with the following security rules (Refer to the NDA for detailed obligations):

This document may not be altered in any way that removes or obscures the Confidentiality notices.
Keep this document locked away. Additional copies can be provided on a "need to know basis", please contact your QUAKE account manager.

Please read carefully:
Information in this document is provided solely in connection with Quake Global, Inc. ("QUAKE") products. QUAKE™ reserves the right to make changes, corrections, modifications or improvements to this document and the products and services described herein at any time, without notice.

All QUAKE products are sold pursuant to QUAKE's terms and conditions of sale. Purchasers are solely responsible for the choice, selection, and use of QUAKE products and services described herein, and QUAKE assumes no liability whatsoever relating to the choice, selection or use of QUAKE products and services described herein. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third-party products or services, it shall not be deemed a license grant by QUAKE for the use of such third-party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third-party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN QUAKE'S TERMS AND CONDITIONS OF SALE, QUAKE DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF QUAKE PRODUCTS INCLUDING, WITHOUT LIMITATION, IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZED QUAKE REPRESENTATIVE, QUAKE PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIRCRAFT, SPACE, LIFE-SAVING OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE.

Resale of QUAKE products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by QUAKE for the QUAKE product or service described herein and shall not create or extend in any manner whatsoever, any liability to QUAKE.

QUAKE™ and the QUAKE logo are registered trademarks of Quake Global. Information in this document supersedes and replaces all information previously supplied.

www.quakeglobal.com