DATRAN Hardware & Telemetry Training Course

Course Notes & Reference Material

1.0   CONTENTS

2.0   QTECH TRAINING OBJECTIVES

2.1 Learning Objectives

Upon successful completion of this training, the trainee;

❑   Will have knowledge on all aspects of DATRAN RTUs Radios, and RTU Peripheral Hardware.   The trainee will be able to identify which components a system requires and have appreciation of the aspects required to design a complete RTU SCADA system.

❑   Will display an understanding of RTU communications, be able to identify the communication devices and have knowledge of connection standards and communication protocols required.

❑   Be able to diagnose simple problems and change settings in DATRAN RTUs using the diagnostics software available.

❑   Be able to implement RTU Data Logging and make calculations using I/O counts and Sampling Rates.

❑   The user will be able to identify which Peripheral Interface components a system requires and how they communicate with RTU.

❑   Be able to perform RTU configuration and have knowledge of RTU maintenance and upgrades in typical field situations.

❑   Will demonstrate an understanding of the overall system, such that fault detection and diagnostics is possible and show an understanding of where to look for errors.

❑   Will be able to describe how when and where the DATRAN SCADA system can be utilised as a design solution and portray the basic system requirements given a basic scenario.

❑   Have an understanding of what the DLP is, how it is written and what it is capable of. Expert users with familiarity of programming logic should be able to write basic DLPs.

3.0   OVERVIEW

The following areas will be covered to achieve the learning objectives:

DATRAN RTU Hardware
> The QTech DATRAN II eXcel and the QRTU are outlined. Their basic functionality and features are covered, along with Expansion and Communication Modules.

Communications Hardware
> All forms of QTech communication hardware are covered including Radio, Cellular & Dialup Modems, Ethernet and Direct Links. Features, functionality, protocols and cabling standards are covered for each type of device.

Interfaces to Peripheral Hardware
> The following interfaces are covered: HMIs, PLCs and the Keri Proximity Card System. Their features, functionality and communication with DATRAN are outlined.

Data Logging
> The 2 forms of DATRAN data logging are covered: RTU Data Logging and Enhanced Data Logging (EDL). Each one is explained in detail, plus RTU hardware/firmware prerequisites and memory usage are also covered.

RTU Maintenance
> The following basic RTU configuration processes are covered: Status LEDs, setting addresses, loading firmware, batteries, adjusting radio modem outputs, setting peripheral interface & comms options.

Q90
> Q90 configuration software functionality is also summarized.

Radio Networks
> Types of repeaters are primarily covered in this part of the manual, along with simplex & duplex frequency radio networks.

System Design
> Various types of RTU based SCADA systems are covered, along with advantages & disadvantages. Base Stations, Point to Point, Point to Multipoint and Master/Slave designs are investigated.

DLP – RTU Control Programs
> An introduction to DLP is covered, together with an understanding of how to design and write DLP. Example programs will be written using best-practise.

4.0   DATRAN II eXcel RTU

4.1 Specifications

This RTU is a compact, self contained DIN rail mount RTU. It is typically installed in the field, and communicates with the DATRAN VI Base Station via Radio, Dialup Modem, Cellular Modem or Direct RS485 Link. The RTU accepts Analog or Digital Inputs in the field, and forwards it back to a DATRAN VI Base Station. It can operate in polled or change of state (COS) communication modes. It can also time-stamp and log data, for retrieval at a later time. Control of field devices from the Base Station or from a local control program running in the RTU is possible using the RTUs Digital Outputs.
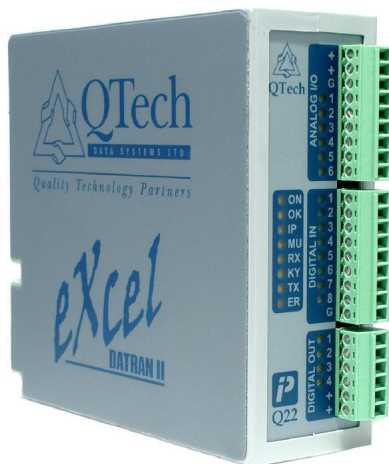
Main Features:

- 16 Bit CPU (Q03)

- 8 Digital Outputs

- 8 Digital Inputs (Opto Isolated, & perform pulse counting)

- 6 Analog Inputs (2 of which can be changed to Outputs)

- Internal 300 or 1200bps Modem option for Analog Radios on older models. Now being phased out.

- RS232 Port for Data Radio or PLC Comms etc

- IPB, Intelligent Peripheral Bus for I/O Expansion Modules (RS485 Port)

- 64KB Memory for Data Logging

- Can Run a DLP (Down Loaded Program) control program

- Rechargeable Battery Backup for Data & DLP retention

- Status LEDs (inc Radio Comms Status)

- Fully Configurable using Q90 Software

A hardware variant of the RTU is available with two RS232 ports instead of the 300/1200 modem. We call this a "Rev.D" eXcel and this is now standard.

5.0   DATRAN QRTU

5.1 Specifications

The QRTU is a EuroCard style RTU that is designed to slot into a Motherboard to connect to I/O, power and communication devices.

There are two motherboard options to best suit the requirements of the system.

The QRTU is the predecessor of QTech's new DATRAN II eXcel, but is still offered as a current product, as so many clients still use QRTUs in the field.

This RTU was originally designed with an 8 bit Q01 CPU Board (top board in diagram), which could not perform functions such as Enhanced Data Logging. The Q01 CPU could not retain a DLP if powered down, unless the DLP was burned into an EEPROM on the board.

QRTUs manufactured now come with Q03 CPU boards, and these have all the features of a DATRAN II eXcel RTU. It should be noted that the Q03 for an eXcel or QRTU is slightly different firmware though, due to the extra I/O of the QRTU. They are not interchangeable.
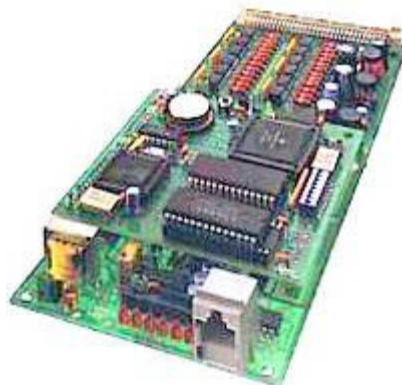
Main Features of QRTU:

- 16 Bit CPU (Q03)
- 16 Digital Outputs
- 24 Digital Inputs (Opto Isolated, & perform pulse counting)
- 4 Analog Inputs
- Internal 300 or 1200bps Modem option for Analog Radios
- RS232 Port for Data Radio, PLC Comms etc
- RS485 Port for I/O Expansion Modules
- 64KB Memory for Data Logging
- Can Run a DLP (Down Loaded Program) control program
- Status LEDs (inc Radio Comms Status)

- Early generation CPU boards are easily upgraded to Q03
- Fully Configurable using Q90 Software

Note: If upgrading a QRTU from a Q01 to a Q03, any QRTU I/O expansion modules also need a firmware change due to the additional functionality.

5.2 QRTU Accessories

As shown below there are 2 variations of Motherboard: 3 Position & Single Position. Both contain connections for power, radio audio, and RS485 terminations. Both motherboards also contain a low voltage drop-out protection circuit.
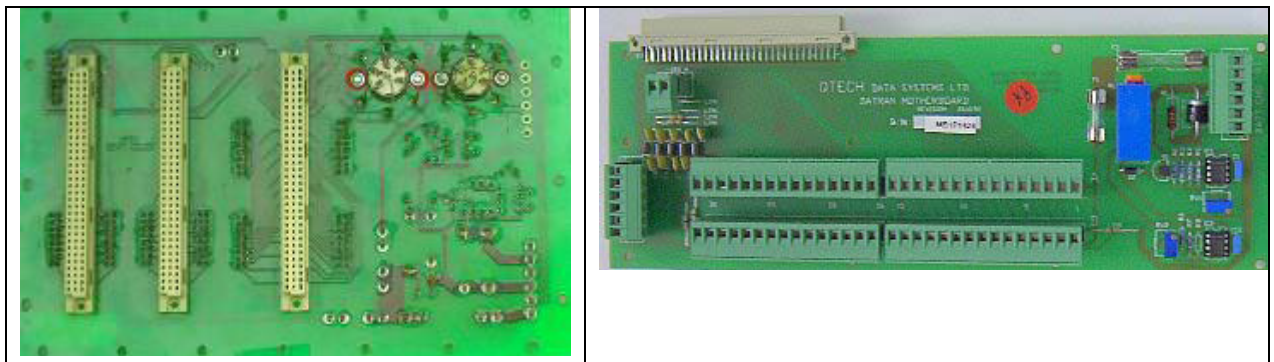
3 Position Motherboard   Single Position Motherboard

5.2.1   *Three Position Motherboard*

This is designed to fit a QRTU rack and provides slots for two expansion modules. This board contains no I/O terminals. I/O Termination Modules must be used with this configuration for ease of wiring.
The QRTU must also be installed in the socket in the centre of the motherboard, nearest the radio sockets.

5.2.2   *Single Position Motherboard*

Fits a single QRTU Board only, and contains I/O termination (24xDI, 16xDO 4xAI).

5.2.3   *Analog/Digital Expansion Boards*

Fit into the free slots in a Three Position Motherboard alongside the QRTU. Additional termination modules are required for connecting the extra I/O.

Analog Boards have either 8xAI, 8xAO or 4xAI & 4xAO. The Digital Boards are available in either has 24xDI or 24xDO

5.2.4 Comms Module Expansion Board

Fits into the 3 Position Motherboard alongside the QRTU. The Comms Module allows a Master QRTU to communicate with Slave QRTUs or Slave DATRAN II eXcel RTUs via a Radio. The later hardware revision of the Q37 Comms Module and its functionality will be discussed later.
The Comms module must always be installed in the socket next to the QRTU on the motherboard.
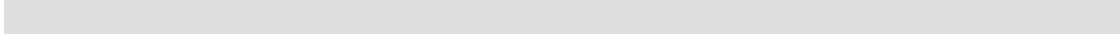
6.0   I/O EXPANSION MODULES

6.1 Q23 DATRAN Expansion Module

The Q23 I/O Expansion Module shown below is used in conjunction with the Q22 DATRAN II eXcel RTU or the QRTU (Q03). It is connected to the RS485 port, and provides the following I/O listed below.

Alternatively the Q23 can communicate directly with a Base Station running DATRAN 6.60 or later, via RS485

Features:

- 8 x Digital Inputs

   These are high speed pulse counters with firmware >IN284.

- 8 x Digital Outputs

- 6 x Analog Inputs, two of which can be turned into outputs if needed.

- Provides up to 255 I/O (total)

- Status LEDs

- Can be used in a 'Point to Point' SCADA system, discussed later

Multiple expansion modules can be used.

Expansion modules are added to the RTU via Q90.

Note: Different types of Expansion Modules can be mixed together when connected in 'daisy chain' style to the RS485 IPB (Intelligent Peripheral Bus)

ports of the DATRAN eXcel RTU. The only limitation of numbers of modules is the total I/O count (255)

6.2 Q26 DATRAN Expansion Module

The Q26 I/O Expansion Module is the same form factor as the Q23 shown in the previous section. It also can be used in conjunction with the Q22 DATRAN II eXcel RTU or QRTU (Q03). It is connected to the IPB RS485 port, and provides the following I/O listed below. Alternatively the Q26 can communicate directly with a Base Station running DATRAN 6.60 or later, via RS-485

Q26 Features:

- 16 x Digital Inputs

  Inputs 1-8 are high speed pulse counters with firmware >IN284.

- 8 x Digital Outputs

- No Analog In

- Provides up to 255 I/O (total)

- Status LEDs

Multiple expansion modules can be used.

Expansion modules are added to the RTU via Q90.

Note: Different types of Expansion Modules can be mixed together when connected in 'daisy chain' style to the RS485 IPB (Intelligent Peripheral Bus) ports of the DATRAN eXcel RTU. The only limitation of numbers of modules is the total I/O count (255)

7.0   COMMUNICATIONS HARDWARE

7.1 Overview

The most common form of communication between the DATRAN Base Station and RTUs in the field is Radio. Cellular options are increasing in popularity due to cost reductions for mobile data plans from Telcos. QTech has recently introduced a fast permanently connected Cellular IP option for CDMA. Dialup PSTN Modem is an alternative for communicating with RTUs, however, a landline must exist to the remote site, and this is not very common.

(The QComs Protocol is the standard for RTU communications across all media)

In a situation where a RTU is within close proximity to the Base Station, RS232 or RS485 connection options are available via cable. RS485 has a good immunity to EMI noise, and can run a maximum distance of 1.2km. Long cable runs should be fitted with isolation devices to prevent induced noise and EMI issues.

RS232 and RS485 comms are also available via digital radio connections.

Ethernet (TCP/IP) is also available using a Q53 Ethernet Interface.

The following sections will cover Communications Hardware; features & functionality.

7.2 Tait TM8110 Telemetry Radio

The picture above shows the most common brand of Radio used in DATRAN SCADA Systems. It can be used in two modes:

The first requires a Q82 Interface board mounted inside the case in order to be compatible with the DATRAN II eXcel RTU or QRTU's FSK port, to communicate with legacy systems at either 1200 or 300 baud.

The second utilizes an inbuilt Tait 2400 Baud FFSK modem to communicate with the DATRAN Base Station. See section 7.4.

Notes:

The Q82 Interface board has been designed to utilise the internal options connector in the TM81xx radio to provide direct connection & signal conditioning for connection to a DATRAN RTU with internal modem.

TM8110 Features:

- 10 Channels (TM 8115 model has up to 100 Channels)
- Voice Option (Not generally used)
- Adjustable 5-25W power output (excellent for long range Comms)
- Robust & Reliable
- Replaces olderT500/ TM2000 series radios (many still in use)
- NZ made and extensive local support

7.3 Q80 DATRAN Telemetry Radio

The radio shown below is a cost effective low power solution for RTU sites. It is ideal for when radio licensing requires a low transmit power.

Features:

- Internal Maxon Radio Module
- 0.5-5W Transmission Power
- Low cost alternative to Tait Radios
- Low Power Consumption (ideal for solar powered RTU sites)
- 16 Channels
- Directly connects to FSK modem of RTU

- DIN Rail Mounted & Matches Look of DATRAN II eXcel

7.4 Tait TM8105 Data Radios

These models are specifically configured as data radios and do not have the voice option as there is no microphone. They connect to the DATRAN RTU via RS232 instead of the FSK port. The integrated radio-modem converts a serial data stream from the RTU to FFSK for transmission at 2400 Baud.

To utilize the 2400 Baud option for communications to the base station, a similarly configured radio must be installed at the base station to provide a 2400 Baud comms channel. Because the DATRAN FSK modem is not required on RTUs using this option, it is possible to use an eXcel RTU Rev.D instead, which allows a cost saving.

Features:

- Provides a higher baud rate option for systems where radio network loading is a concern, or where large volumes of data are being transferred (e.g. RTU data logging)
- Ideal for Point-to-Point SCADA Systems (discussed in a later section)
- TM8105 can be optionally fitted with Q82 for use with RTU FSK Interface

7.5 Spread Spectrum Digital Radios

These digital radios are free-to-air frequency hopping spread spectrum (FHSS) data radios. They are designed to provide a transparent radio extension to an existing RS232 or RS485 connection to allow remote devices to appear as though they are connected locally to the RTU.

They can be configured in minutes to provide low-cost serial cable replacement between all types of electronic devices. Previously available was a device called the XTend. Due to RSM license condition changes in August 2010, these are no longer allowed for use in NZ.

The QTech QDR1 digital radio is its replacement and fully NZ compliant with the new NZ requirements.
Transmitting only 0.1watt with a yagi aerial gives a sensible range of 5 or 6km.

The primary application for the QRD1 digital radio is to provide a wireless radio extension for an RS485 bus, allowing an RTU to have expansion I/O modules like a Q23, Q26 or Keri Q50 installed several kilometres from the master RTU, with no need to dig trenches or lay cable.

As they can also be used for RS232 this also makes them suitable for base communications applications where the distance between the RTU and the base station is not excessive.

Digital radios are cheaper than conventional analogue radios, the aerial are physically smaller and they have a very low power consumption.

As the DATRAN expansion modules are also cheaper than a fully featured RTU, using expansion modules at a remote site leads to an overall system cost reduction.

7.6   Radio Antennas

| Yagi UHF or VHF Antenna (4 Element) | Whip Aerial |
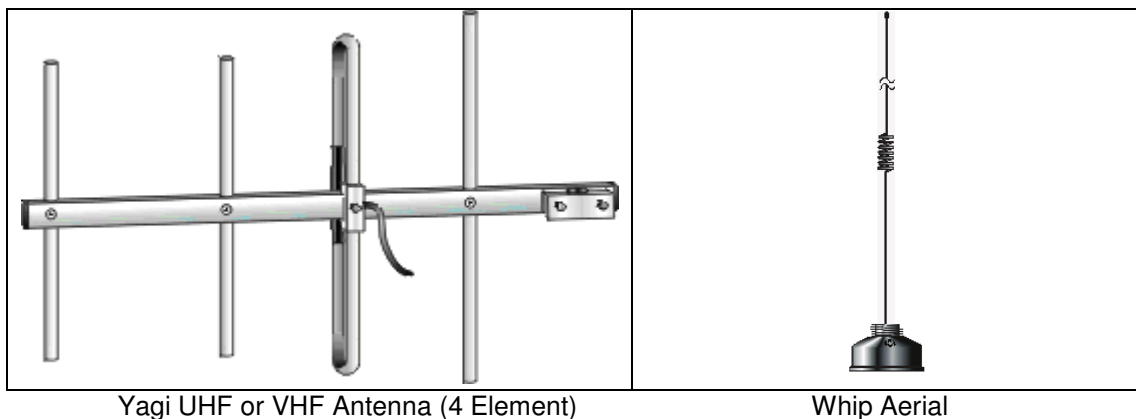
The two diagrams above show the two most popular choices of antenna for Base Stations and RTUs. Yagis are directional and provide dB gain, the higher the number of elements, the tighter the focus of the aerial, and the higher gain. Whips are omni-directional and provide little or no gain.

Typically a Whip would be used when only needing to communicate over short distances. Since this aerial is omnidirectional it is also ideal for situations where it will communicate with multiple RTU sites spread in all directions.

7.7 Q37 Comms Module

The DATRAN Comms Module is used to provide a master slave topology. This allows multiple RTU sites to pass I/O between them without the need for them all to communicate directly to the base station.
The master RTU is the only one which communicates to the base station.
This will be shown later in an example of system design and is an ideal solution where only one RTU has a comms path to the radio repeater.

Note the Q37 is now only sold "while stocks last" as it has been superseded with the Store-and-Forward functionality available in the Q03 firmware.

Features:

- RJ10 connector - RS485 to RTU IPB
- Internal 300 or 1200bps modem
- Configured using HyperTerminal on RS232 (Slave Polling Rate & Baud Rate)
- Tx & Mute Status LEDs

7.8 Q70 Base Station Modem

The Q70 is primarily used at the base station to interface to the base station radios that communicate to the RTUs.
It has been designed to reliably transmit and receive QComs data over a fixed line or radio channel using FSK. It can be set to operate at either 300 or 1200 Baud via internal DIP switches.
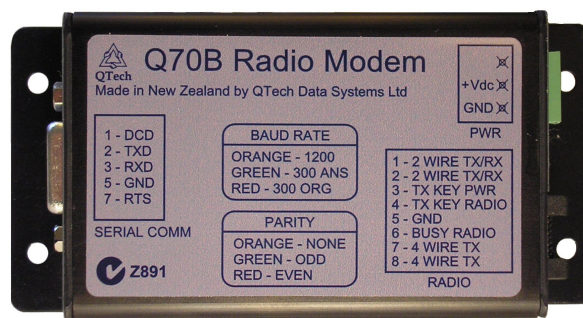
7.9 Q71 External 1200bps FSK Modem

The Q71 is a 1200 bps modem which is external to the RTU.
It provides the necessary DATRAN interface for conventional 1200bps radio systems.
DATRAN II eXcel RTUs can now be standardised on the two serial port "Rev D" type. This helps prevent having to change (or convert) the RTU hardware when high speed comms are used in the future.

More and more DATRAN users have adopted high speed data radios. This uses one RS232 port, so the previous Rev C RTU can not connect to a PLC/HMI too.

The Q71 can be connected to either RS232 port on the Rev D eXcel.

You must know which radio type the Q71 will interface to before ordering. There are different cables for the Tait TM8000 range (high density D type) or the 6 pin circular DIN for DATRAN Q80 or similar.

The Q71 can also be used in Master Slave/store and forward applications to provide an analogue link between slave sites.

7.10 Wavecom GSM Cellular Modem

The cellular modem shown below connects directly to a DATRAN II eXcel RTU or QRTU via RS232. It can be used to communicate with the Base Station via a dial up connection on Vodafone's cellular network (using QComs Protocol)

Alternatively it can send SMS Alarms directly to an operator's mobile phone. This can occur from either Base Station, or optionally from a Stand-Alone SMS RTU.

The modem can also receive SMS 'Acknowledge Alarm' replies from operators when used with an SMS RTU. These must follow a specific DATRAN format.

7.11 Intermax Quad band 3G Cellular Modem

The Intermax is a powerful quad band cellular modem.
The Intermax supports Telecom, Vodafone and 2Degress networks and has a SIM card slot.
It is used for high speed cellular IP comms, to send SMS's from an RTU or to send SMS messages form a base station. Can connect via serial RS232 or USB.
Refer to the specific application note for further details.

7.12 iPort CDMA Cellular Modem

This modem only supports the CDMA network. It was discontinued by the manufacturer in 2010. However, there are many of these installed, and probably will be until the CDMA network is closed down.

The cellular modem shown below connects directly to a DATRAN II eXcel RTU or QRTU via RS232. It can be used to communicate with the Base Station via Telecom's cellular network and can operate in either a conventional dial-up type arrangement, or can be used to connect as a dedicated Cellular IP connection allowing the base station to communicate with the RTU via the internet. Both methods use the QComs Protocol.

Alternatively it can send Alarm SMS messages directly to an operator's mobile phone. This can occur from either Base Station, or optionally from a Stand-Alone SMS RTU.

The modem can also receive SMS 'Acknowledge Alarm' replies from operators when used with an SMS RTU. These must follow a specific DATRAN format however.

Cellular Modem Notes:

1: Many people refer to these devices as 'SMS Modems'. This is only correct terminology if the modem is being used on a Base Station or RTU to send SMS Alarms (Pre-Paid plans can be used in this case). Otherwise the modem should always be referred to as a Cellular Modem, as it requires a data plan from a local Telco to operate.

2: A Base Station does not have to have a Cellular Modem to communicate with a cellular RTU site. It can have a normal PSTN dialup modem (for conventional Dial-up) or it can communicate via the internet (for a Cellular IP site). Cellular IP needs IT support to facilitate please remember to engage the IT manager in the discussions.

7.13 Q53 Ethernet Interface

This product is designed to allow DATRAN RTUs to communicate with the base station via an Ethernet connection. This connection maybe direct on a LAN or via an Ethernet port on a Wireless LAN Access point. This has even been used to achieve connection via satellite.
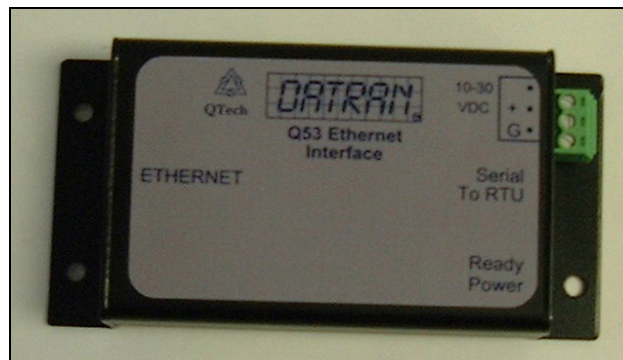
Prerequisites for Operation:

- DATRAN QComs Ver 6.50.8 or later
- An RTU with firmware IN235 or later
- Q90 Version 3.6 or later, as required by firmware version

NOTE: The device may be connected via the RS232 port on a DATRAN II eXcel RTU or the RS232 port on a Rev.D Q03 QRTU.

8.0   INTERFACES

8.1 Overview

There are various Interface devices available for RTUs. The Interfaces covered in this section will be the Q50 Keri Module (Building Security and Gate Access Control), Programmable Logic Controllers (PLCs), and Human Machine Interfaces (HMIs).

8.2 Q50 Proximity Card System (Keri Module)Overview

Shown below are the 3 main hardware items that make up the Q50 Proximity Card System. The card and the tag emit a RFID that is picked up by the Reader Module. This unique ID is then logged by the RTU (which connects to the Keri Reader via the IPB (RS485)). Access rights can be either granted or denied by the RTU, as it contains an operator list with security groupings. These lists and groups can be changed at the DATRAN Base Station and communicated to the RTU at any time.

The Q50 controls a series of relays, and locks/unlocks the door to a remote site for anyone who has the correct Card/Tag.
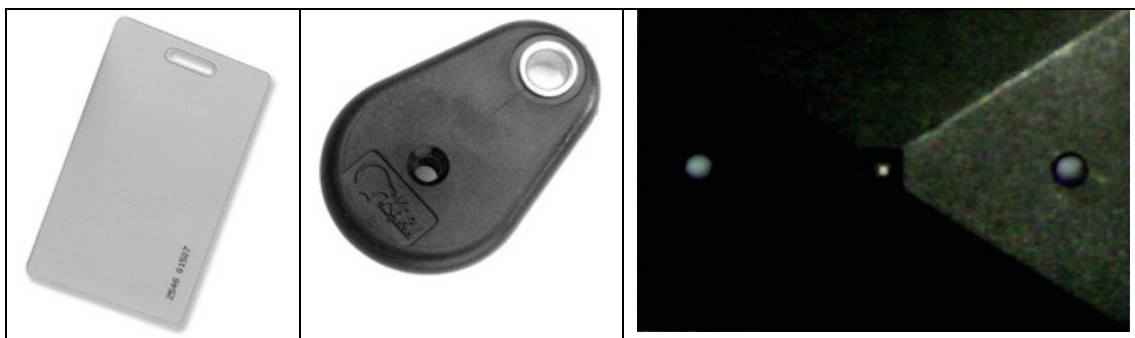
Keri Key Card   Keri Key Tag   Keri Reader Module

Important Points to Note:

The site RTU must have a Q03 processor to interface to a Keri module

Communications to a DATRAN Base Station are required to load the operator list into the Q50's memory. To do this, the DATRAN Base Station must have the 'RTU Console' service

A Max. of 255 operators can be configured,

10 operators can be logged in at one time (OSH Version)

The operator list is sent from Base to RTU using 'Routed Datagrams'

At the Base Stn, you must define the type and address for the datagram destination.

At the Base Stn, addressing is the Q50's address on the RTU's IPB bus, the type is always '80'.

In DATRAN Diagnostic Console, tick the 'QTech Comms' item to show the Routed Datagram progress.

Additional Diagnostics are available under 'RTU Console' in DATRAN

8.3 PLC Interfaces

There are 4 main Interface Drivers available to DATRAN RTUs. Depending on which brand of PLC is required to be interfaced, one of the interface drivers below will be required:

8.3.1 Allen Bradley DF1

This interface deals with the construction and processing of DF1 packets to communicate with Allen Bradley PLCs in half duplex mode.

8.3.2 MODBUS

This interface deals with the construction and processing of Modbus packets. This is the most common form of communication with generic PLC brands. At the time of writing, the RTU can only be set as a Modbus 'Slave', but development is currently underway to allow the RTU to be either a Modbus Slave OR Master. Points to note are listed below:

- RTU as a Modbus Slave cannot initiate any unsolicited communications.

- Max 256x DI & 256x DO

- Multiples of 16 Digitals Read at a time

- Max 100x AI and 100x AO

Note: The Modbus & DF1 interface options are set in the RTU using Q90 Diagnostic Software. There are licensing requirements associated with these options. An activation code in entered with Q90.

8.4 HMI Interfaces

A Magelis (Human Machine Interface) HMI is shown in the picture below.
The HMI allows a user to visualise process parameters, I/O status and historical trends without needing DATRAN VI software on a Base Station or using a laptop.

This is the most common HMI interfaced to DATRAN RTUs, and communicates via Modbus over RS232. They are available in a range of size and available colour options.

Extra accessories for the Magelis HMI:

- Connector Cable
- 256MB RAM (Optional Module)
- Logging Trending Event & Alarm Application

The Magelis software application must be programmed & configured for the requirements of each RTU site.

Note: The Magelis Unit is a Hardware HMI interface. There are other Software HMI interfaces such as iFix, InTouch and Citect. These are covered in the DATRAN Software Training Manual.

9.0   Q90 Configuration & Diagnostic Software

9.1 Q90 – Introduction

Q90 software is a powerful tool that is used to both configure and diagnose RTUs.

The Q90 software is run typically on a laptop that is connected to the RTU via the RS232 Diagnostics Port. RTU Status indicators can be uploaded, and RTU settings downloaded using this link.

Overview of Q90 Features:

- Synchronize Real Time Clock with Laptop/PC
- Set Alarm level for DC voltage alarm
- Reset RTU & records reset stats
- FSK Radio Modem Settings can be configured
- RS232 can be set. E.g. Baud Rate, Dialup/Cellular Modem, PLC Interface Protocol…
- Intelligent Peripheral Bus (IPB) addressing can be set for expansion modules
- Change Of State (COS) can be set for Inputs
- Down Loaded Program (DLP) can be downloaded or stopped.
- Data Logging can be set, along with Analog Deadbands and Digital Masking
- Analog inputs (5&6) can be changed to outputs, and also set to Voltage or Current
- Error LED can be configured for various error messages
- Remote dial in to RTU
- Viewing of communications activity

The most appropriate training on Q90 is hands on.

The following section will introduce the main functions before practical tasks.

9.2 Q90 – Module Status Screen

Contains a summary of the current status of the unit
Allows the user to synchronize the Real time clock with the PC
Allows the user to set the alarm level for the "low battery" alarm
Displays stats on the running totals of resets that have occurred

9.3 Q90 – Communications Screen

I/O Board FSK Modem - All the settings for radio comms to the base station. Settings must be the same as are configured in the 'Channel' node at the base station. Also contains extra settings for extra RS-232 port on a 'Rev D' eXcel.

RS232 Channel - contains settings for the external RS232 port on the RTU for talking to external devices such as a data radios, cellular modems, PLCs or the TCP communications module. The dialog that is displayed when the configure button is pressed changes dynamically depending on the device set in the 'Device Type' box.

COS Configuration - settings for how the RTU deals with sending COS messages to the base station. COS can be turned off or on an individual point basis. If a DLP is used, DLP may override these settings.
Also covers the number of times the RTU will retry if the base station does not acknowledge the COS message before it goes into "Comms Fail".

Intelligent Peripheral Bus (IPB) - Settings for the RTU expansion modules. Also provides diagnostics for looking at the comms on the IPB. Provides a summary I/O map of all the available I/O configured on the RTU. It also allows configuration of "master notionals" which must be used to view or data log notional I/O.

Radio Config – Contains settings to optimize radio interface settings such as leader tones. This is an important setting to help reduce waster air time on busy radio networks.

9.4 Q90 – TCP Configuration
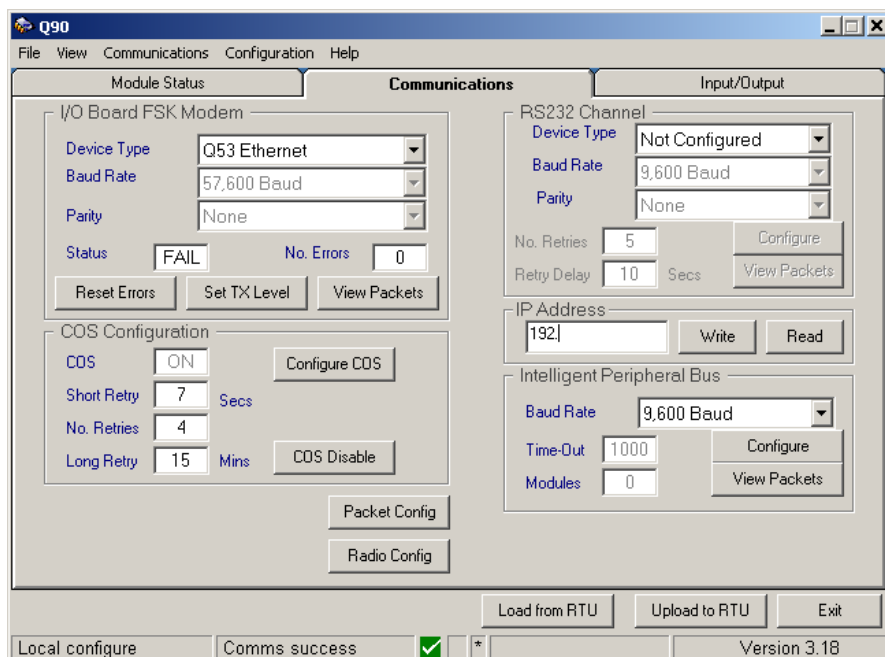
If the RTU has the Q53 Ethernet interface, the IP address is set here.

9.5 Q90 – The I/O Screen
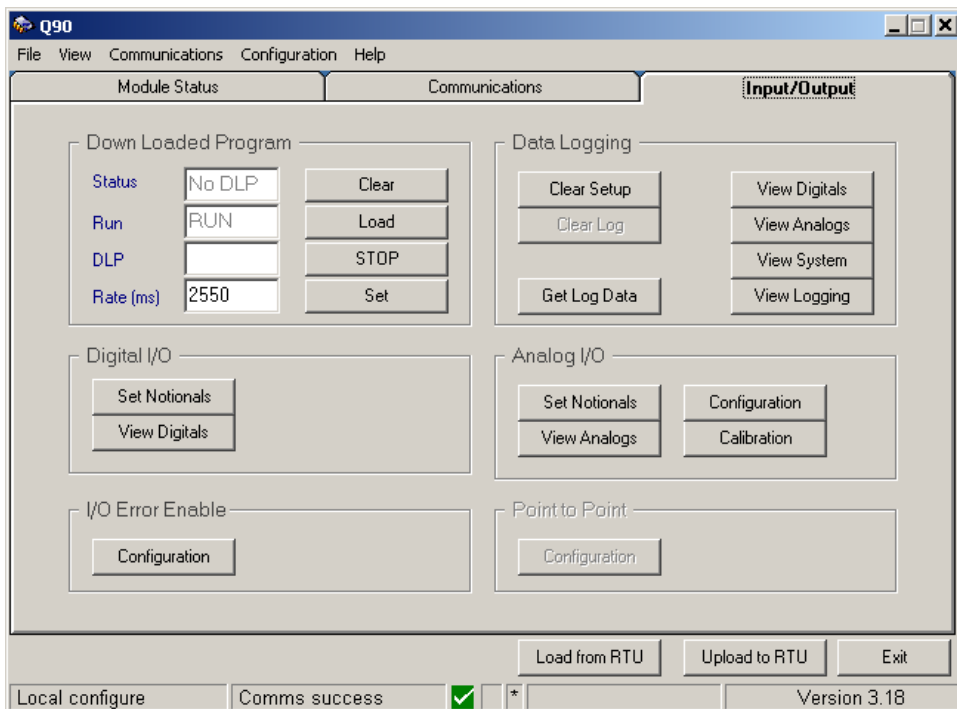
Downloaded Program (DLP)
Displays the current status of any DLP loaded into the RTU. From here you can see if any program is resident, if it is running, as well as being able to upload new DLP's.
You can also change how frequently the DLP executes. Care must be taken as settings which are too low will cause problems.

Data Logging
This refers only to EDL and allows configuration of data logging and downloading of logged data. This option is only shown if the EDL option is available. This is not used for "RTU data logging" which is configured at the base station.

Digital I/O
Displays the current Digital points configured in the RTU. The I/O is divided up into Real, Telemetry and Notional Inputs and outputs.
Telemetry I/O are the Real I/O as they are transmitted to or from the Base Station, after (potentially) being modified by the DLP.
Notional I/O are usually used for passing variables such as set-points or derived values such as pulse accumulations between the RTU and the base station. Notional I/O do not (usually) exist as real-world signals.

Analog I/O
Displays the analogue I/O in the same manner as the Digital I/O. The other options available are to configure the Analogs for current or voltage input, single or differential.
Allows channels 5 and 6 to be set as outputs if required, but remember that a jumper setting is also required. There is also a calibration section for

calibrating the analog inputs. To do this you'll need to be able to accurately apply inputs to the RTU and follow the Q90 instructions.

I/O Error Enable
Allows configuration of the Error LED on the front panel of the eXcel RTU. There are several types of error the eXcel is capable of alerting the user to by flashing the error LED, a given number of times. Problems such as short circuits across the terminals of Digital Outputs, Analog inputs being out of range, Comms Fails and Battery Failure are able to be displayed.

9.6 Q90 – Versions and Compatibility

- Current Version   3.28

- Product Compatibility   Q03
                           Q01/Q02   Hyper terminal only

10.0 RTU CONFIGURATION & MAINTENANCE

10.1 Status LEDs

The status LEDs on the front panel of the RTU give important feedback and are important for helping diagnose field issues.

The functions are defined below:

ON   This green status LED is on whenever power is applied to the Q22 Module and the module is not in low battery detect shut down mode. When the correct supply voltage is applied to the Q22 Module and this LED is not ON it could indicate that the internal fuse has blown.

OK   This green status LED flashes at a slow regular interval to indicate that the Q03 – 16 Bit Processor's central processing unit is operating correctly. This LED will flash 6 times rapidly whenever a system reset occurs. System resets occur at system power up and whenever the reset button is pressed or the Q03 – 16 Bit Processor's central processing unit's watch dog timer times out.

IP   This yellow LED flashes on for approximately 0.5 seconds every time this module transmits a packet on the RS485 Intelligent Peripheral bus.

MU   This yellow LED turns on whenever the mute signal is turned on in the FSK modem. That is, the radio channel is busy. The Q22 Module will not transmit data when the MU (mute) LED is on.

Rx   This yellow LED turns on whenever a valid data carrier is detected from the modem.

KY   This yellow LED turns on whenever the RTU is attempting to turn on the transmitter in the attached radio.

TX   This yellow LED turns on whenever the modem is generating transmit carrier and data to the radio.

ER   This red LED is used to flash coded error messages. These lights will be extinguished when no error condition exists. Whenever an error condition occurs these lights will flash ON and OFF. The number of times the light flashes ON is the error function code. Each error code is separated by a 2 second OFF period thus allowing for multiple error codes to be flashed. This functionality can be enabled/disabled in Q90.

The following number of flashes of the ER (error) LED indicate:

1 Flash	Main battery failure
2 Flashes	Clock battery low
3 Flashes	Main Base communications has failed.
4 Flashes	I.P.B. communications failure
5 Flashes	One or more real digital outputs has failed.
6 Flashes	One or more Analog input is out of range
7 Flashes	The DLP is not loaded or corrupted.

More detailed configuration is available in Q90 about which Digital Outputs or Analog Inputs should be included for error 5 and 6.

10.2 Batteries

The rechargeable battery on the Q03 processor board of the RTU should be replaced every 3-4 years. If this is not done the end user risks losing logged data & the DLP if there is a power failure to site.

Q01 and Q02 processors will begin to randomly reset themselves if their batteries are failing. Logged data and DLP will be lost every time there is a power failure as this is not backed up by battery.   Batteries are non-rechargeable, and should be replaced every 2-3 years.

Q01, Q02 and Q03 processors will all lose date and time if their onboard batteries go flat.

10.3 Setting the RTU Address

There is a block of 8 dip switches on the processor board of each RTU. These are numbered 1 to 8, with 1 being the least significant bit. The setting of the switches (On=1, Off=0) directly represents the RTUs address in binary.   The address range is 1-254 in decimal. (Address 255 is reserved for causing a factory default reset)

E.g. The following setting is RTU address 15

Switch	1	2	3	4	5	6	7	8
Position	1	1	1	1	0	0	0	0

10.4 Adjusting the RTUs internal Radio Modem

The RTUs Modem attenuation on transmit should be set to:

-3dbm (or 0.55v into 600ohms)

10.5 Changing the RTUs Baud Rate

The FSK Modem chip must be physically replaced to change the RTU from 300 bps to 1200bps. It is the only chip in a socket and is in the top-left quadrant in the Q22 motherboard picture below. When the chip is replaced, jumpers J23, J25 and J26 will need changing (See the Q22 user's manual). The Tx Audio level should be checked and adjusted as necessary using the adjustment labelled 'Tx Data Level'
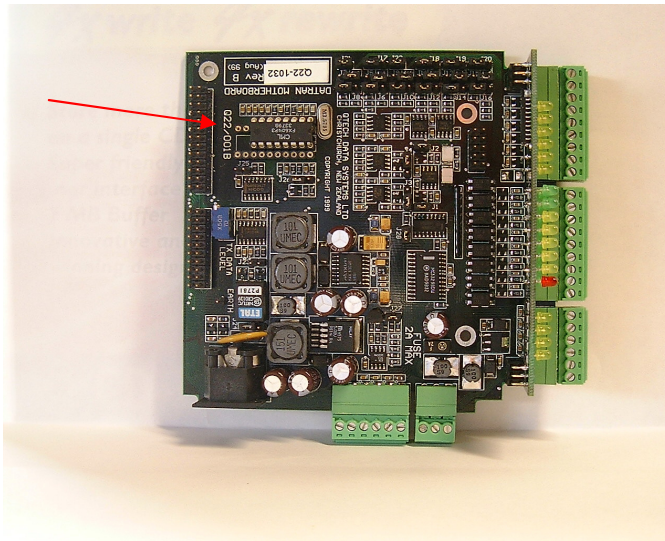
10.6 Setting the Analog Inputs

In the previous picture, a row of jumpers can clearly be seen along the top edge of the board. The position of these must be changed in order to set the type of Analog input. (See RTU Owners manual)

The Analog input channels can be individually configured for 4 different voltage and current input ranges. These are:

0 to 5 volts

0 to 10 volts

0 to 20mA

4 to 20mA

Q90 software must also be used to program the RTU for the correct input, once the input jumpers are set.

10.7 Setting the Analog Outputs

The Q22 Module has 2 Analog output channels, and they are associated with Analog terminals 5 & 6. The 2 Analog output channels have an associated yellow channel status LED to show the state of the associated Analog output channel.

Note: Analog channels 5 & 6 can be configured individually as either input or output channels via jumpers & Q90.

Each Analog output yellow channels status LED shows the following channel status:

OFF   Indicates that the Analog output is at 0V/0mA dependent upon the output mode configured.

ON   Indicates that the Analog output is within the specified voltage or current range dependent upon the output mode configured.

FLASHING   Indicates that the Analog output is at full scale output of 5V/10V/20mA dependent upon the output mode configured.

The 2 Analog output channels can each be individually configured as single ended voltage or current outputs.

The Analog output channels can be individually configured (via jumpers & Q90) for 4 different voltage and current output ranges. These are:

0 to 5 volts

0 to 10 volts

0 to 20mA

4 to 20mA.

Note: When the analog output channels are configured as voltage output channels the output impedance is very low. Output currents in excess of 100mA can damage the Analog output channel. Care must be taken not to short the outputs!

10.8 Upgrading Q03 Firmware – The EP1

Maintaining the firmware of the RTU should be considered a maintenance task.

There are many reasons to encourage this upgrade, including:

Releasing extra features

Resolving known issues and bugs

Making field diagnostics easier, you will have a known starting point

Simplified spares holding etc

There are two ways of achieving the upgrade:

1 Return the Q03 or the RTU to QTech

2 Use the EP1 programming dongle and complete it yourself.

This prevents the time and cost of returning the Q03 to QTech, dramatically improving your ability to upgrade sites.

The EP1 programming dongle is used to upgrade Revision B,C,D and E Q03 processors for both QRTU and eXcel RTUs.
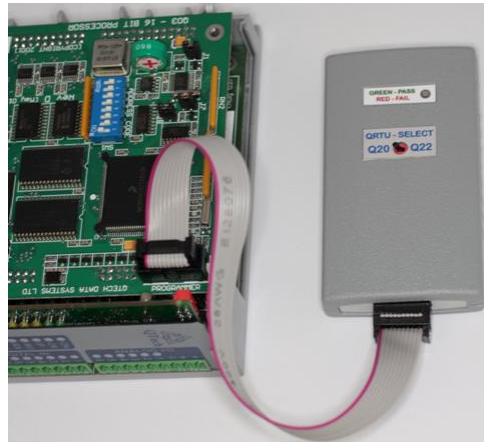
The EP1 comprises the dongle itself and a simple user interface.

The dongle has a single toggle switch to select which type of RTU you are programming. You need to know this as the firmware is subtly different between the QRTU and eXcel RTU due to the quantity of native I/O.

The user interface can be used to check how many firmware licenses remain in the EP1, the version of firmware and to load more licenses. When you purchase additional license entitlements, these are emailed to you and cost significantly less than if QTech were to perform the upgrade.

The firmware for the QRTU and eXcel loaded in to the EP1 can be different.

11.0 DATA LOGGING

11.1 Overview

There are two types of Data Logging: RTU Data Logging and Enhanced Data Logging. DATRAN II eXcel RTUs & QRTUs (Q03 variety) come standard with RTU Data Logging, which allows for time-stamping of data and storing for later transmission to the base station to be logged into the DATRAN Database.

Enhanced Data Logging (EDL) is an added feature that customers can purchase. It allows data logging configuration and retrieval of logs to be carried out with Q90. The data is exported to a CSV file format. This is primarily designed to be used in conjunction with stand-alone RTU's that do not communicate with a base station.

Both types of Data Logging cannot be used at the same time.

11.2 RTU Data Logging

RTU Data Logging is a powerful feature of DATRAN VI allowing RTUs to log data which is periodically retrieved by a Base Station over the existing communication channels.

The time stamped logged data is captured in the DATRAN SCADA database and then be graphed or reported on the Base Station with DATRAN Historical Trending and DATRAN Reporting etc.

Depending on what firmware the Q03 processor has, the RTU will have the following memory:

- 32Kb (Firmware < V5.03)
- 64Kb (Firmware > V5.04)

RTU Data Logging is achieved through configuration at the DATRAN VI base station being transmitted to the RTU.

Both Analog and Digital Data can be logged. The Data Logging can be configured to be:

- Interval (e.g. every 10 minutes)
- Digital State Change (e.g. off-on)
- Analog changes by a dead band (e.g. 5% variation).

With the release of v5.17 firmware, the user now has the ability to control the behaviour of the RTU Data logging dynamically from within the DLP. This allows the DLP to be written in such a way as to change the logging interval of a given point, log a sample of a specific point at any time, and even to halt or resume Data Logging on the entire RTU at any time.

DATRAN VI Service

The DATRAN VI Service is the "Historical Data Interface".
This service facilitates the configuration of the RTU and the retrieval and integration of logged data from DATRAN RTU into DATRAN VI.

RTU Hardware

RTU data logging can be implemented with both DATRAN II eXcel and QRTU hardware using a Q03 16 bit processor.

RTU Firmware

A Firmware upgrade is required for older RTUs to provide Data Logging capabilities. The minimum firmware version is v5.06 although there are serious improvements after v5.17.

11.2.1 Digital Input Logging

Only 1 bit is required to log 1 digital point, however the minimum log requirement is 1 byte (so the user might as well log all 8 Digital Inputs on a DATRAN II eXcel RTU!) These can be logged at regular intervals, or when there is a change of state.

11.2.2 Analog Input Logging

Each Analog is logged as 2 bytes, (16 bits total). These can be logged at regular intervals, or when there is a change of state. A deadband value is required for this, so that when the Analog value goes outside the deadband, it is logged. The deadband can be set in the RTU using Q90 software.

11.2.3 Date & Time Stamping

The Date & Time stamps on any interval logged I/O point is 4 bytes (each consisting of 8 bits).

11.2.4 Calculations

To work out how much time it will take to fill up the memory in a 64K RTU, divide 65535 bytes by the total number of Analog + Digital bytes plus their respective time stamp bytes (Note: only 1 time stamp is required per log period)

E.g. Logging one Analog Input, once a day will create a log of 7 bytes.

65535 bytes / 7 bytes = 9362 Days (until RAM is full)

Note: If Analog Inputs are being logged when there is a change of state, care should be taken with the Deadband to prevent RTU RAM filling too quickly.

11.3 Enhanced Data Logging

This Data Logging option is primarily for stand-alone RTUs, where a laptop is used to download the data periodically. The data is exported to a CSV file using Q90 software on the laptop.

Depending on what firmware the Q03 processor has, the RTU will have the following memory:

- 32Kb   (Firmware < V5.03)
- 64Kb   (Firmware > V5.04)

The same data size and calculations outlined for RTU data logging also apply to EDL.

12.0 RADIO NETWORKS

12.1 Overview

The most common way RTUs communicate in the field is Radio. This is generally due to geographical location, and low running costs (free in some cases!). The UHF Radios rely on line-of-site, and in some cases this is not practical, so a Repeater must be employed. The Repeater is usually on a hill (diagram below), so that it has direct line of sight to both radios that want to communicate. When the repeater receives a transmission from one Radio, it immediately re-broadcasts the signal to the second Radio.
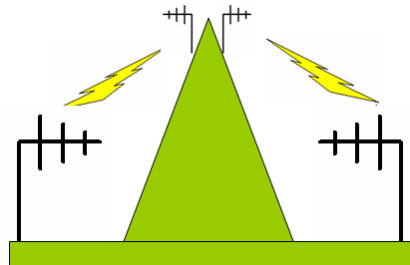
The following sections cover the 3 most common types of repeater: Back-to-Back, Standard Repeater Unit, and the Repeater with Duplexer.

12.2 Back-to-Back Repeater

This is the cheapest solution, and simply involves 2 radios and 2 antennas. The Tx & Rx lines are cross wired between each Radio, and each radio is set to a different frequency (E.g. Radio 1 receives freq A & Radio 2 receives freq B)
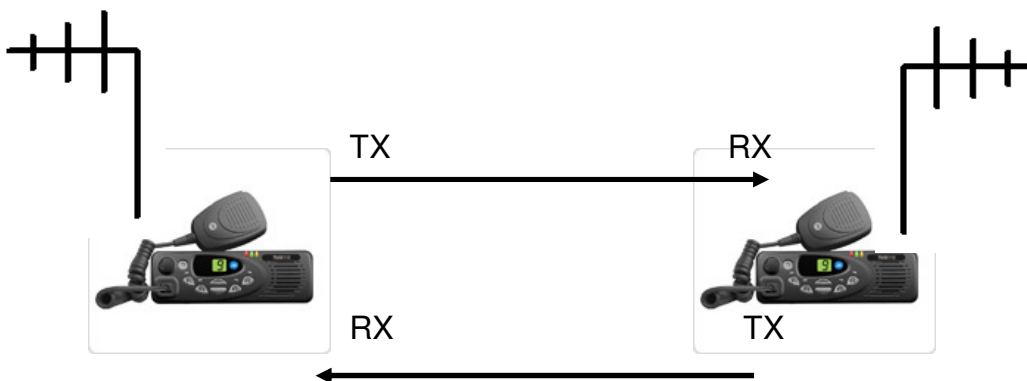
This repeater solution is ideal for small low-medium power SCADA systems with not many sites or much radio 'traffic'. The duty cycle is typically lower.

12.3 Standard Repeater

This Repeater is specifically designed unit that effectively will do the job of 2 radios (back to back), but more efficiently with a high duty cycle and less signal loss. This is ideal for medium-high power systems with many RTU sites.

A Tait TB7100 Repeater is shown below.

Note: 2 Antennas are still required, as the unit receives Freq A and transmits Freq. B

12.4 Repeater with Duplexer

This repeater option is the most expensive, as RF duplexer is fitted. The duplexer allows the repeater to both receive and transmit using the same aerial. This option however reduces the cost of installation and maintenance of a second antenna. This might be an important consideration if the antenna is large & expensive. This is ideal for medium to high power systems. The duty cycle is dependant on the quality of repeater used. Typically the TB7100 above would be used with a Duplexer.
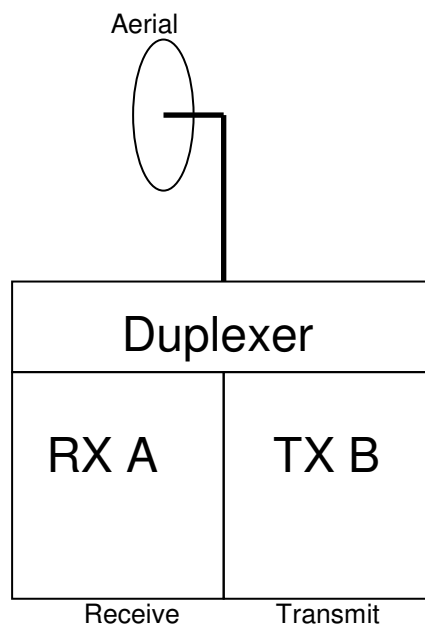
In the diagram above, the Receiver is tuned to frequency A, and the Transmitter is tuned to frequency B. They both utilise the same aerial via the Duplexer.

13.0 System Design

13.1 Overview

The following sections will give examples of how the DATRAN RTUs and communications hardware fit together. Various DATRAN SCADA System designs will be covered, and this will include the Base Station PC (which has been referred to but not covered in detail in this manual).

13.2 Q23 Point to Point System

The Q23 Point-to-Point system has direct comms from one RTU Module to the other via Data Radios. The Q23 modules require special firmware which is provided at QTech during manufacture. There is no DLP facility.

When an input occurs at one point it is mirrored as an output at the other point.
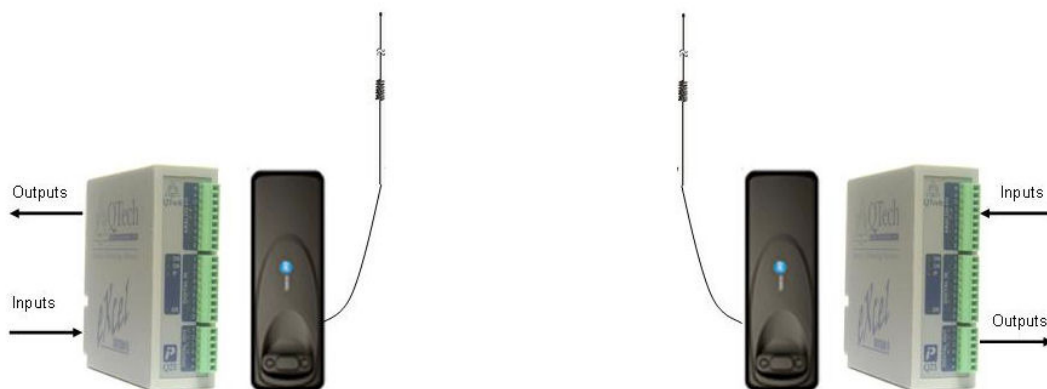
Q23 - Point to Point System

Up to 16 Point-to-Point systems can exist on the same radio channel but all must have separate radios.

Comms is either via RS485 over cable between the two Q23 Modules, or via data radios connected to the RTU on RS232.

Any transparent RS232 or RS485 data radio, such as a QTech QDR1 can be used.

13.3 DATRAN II eXcel Point to Multi-Point System

This system is covered briefly in the training for you awareness. There are a small quantity of these systems installed. It has been largely superseded by the Store-and-Forward functionality available to all Q03s

The Q22 Point-to-MultiPoint system has direct comms from one RTU 'Master' Module to the RTU 'Slaves' via analog Radios. A Master RTU can potentially have up to 8 Slaves (limited by 255 Max I/O), but this would make the radio channel very busy!

The Q22 Master requires special firmware (and special Q90), but the Slave RTUs are unmodified (just normal Q22s). As far as the Slaves are concerned they are simply communicating with a DATRAN Base Station – they do not know the difference.

The Master RTU holds an I/O mapping table that lists which inputs from the Slaves are mapped to its own or other Slave outputs

E.g. When an input occurs at a Slave RTU it is mirrored as an output at the Master RTU.

A Base Station Controlled Master-Slave system utilising a 'Q37 Comms Module' is described in the next section. This has more control & data logging capability that a Point-Multipoint system.

13.4 DATRAN II eXcel Store & Forward - Master Slave System

This is a new technique introduced in 2010 with Q03 firmware v5.22.
This essentially obsoletes the Q37 Comms module based systems.
It removes the legacy constraints of the Q37 system.
This feature is fully documented in a separate application note, however it is crucial to understand the overview scenarios of this feature.

Master Slave Forwarding Mode

This option would generally be used where forwarding is being used to achieve a communications path to a slave that would otherwise not be possible, and the Master DLP does not need to interact with the slave site(s).
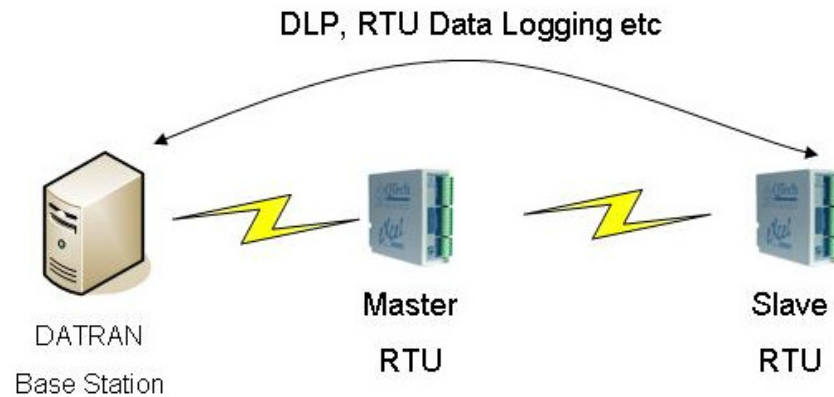
DLP, RTU Data Logging etc

Slave RTU can run a DLP	✓
Base Station can transmit DLP to Slave RTU	✓
Slave RTU can RTU Data Log	✓
Master RTU controls Slave RTU I/O	✗
Base Station controls Slave RTU I/O	✓

Master Slave Polling Mode

This example is essentially a "Q37 comms module replacement" system.
It is generally for use where the DLP in the Master RTU needs to be able to control outputs at the slave site.
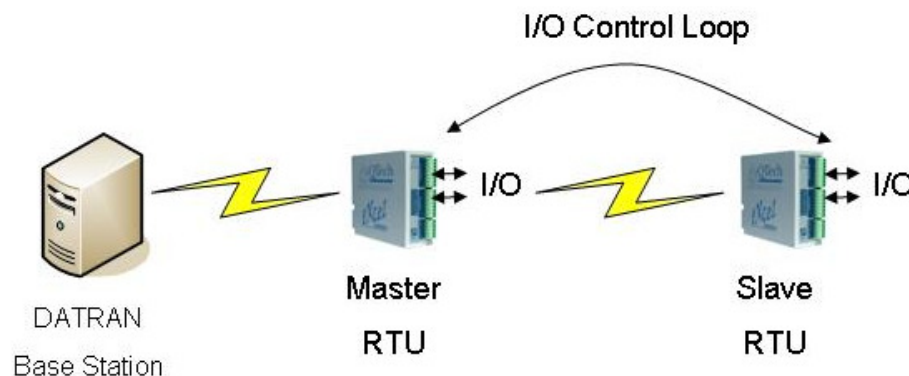
I/O Control Loop

Slave RTU can run a DLP	✓
Base Station can transmit DLP to Slave RTU	✗
Slave RTU can RTU Data Log	✗
Master RTU controls Slave RTU I/O	✓
Base Station controls Slave RTU I/O	✗

Master Slave Polling And Forwarding Mode

This configuration would be suitable for sites where the DLP of the Master RTU needs I/O information from the slaves to be able to do its job correctly, but does not need to control any outputs at the slave sites. Additionally, this configuration allows the base station to carry out data logging functions and transmit DLPs to the slaves.

This is an incredibly powerful set of features that allows interaction between RTU sites. There are many variations to the scenario outlined below, but this one has been chosen to explain the concept.
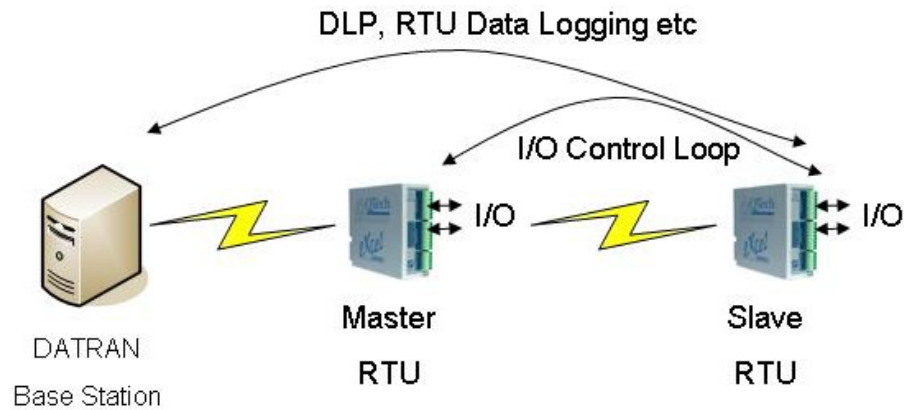
Slave RTU can run a DLP	✔
Base Station can transmit DLP to slave RTU	✔
Slave RTU can RTU Data Log	✔
Master RTU controls Slave RTU I/O	✔
Base Station controls Slave RTU I/O	✔

Note that while the table above shows that both the base and the master RTU can control the I/O at the slave, they are mutually exclusive options.

13.5 Q37 Master Slave System

As mentioned in the previous section, this technique is to be considered obsolete and not to be used for new installations.
However, there are many Q37 based systems in the field and this section covers these systems briefly so that you can support them.

In this case below, the Master RTU uses a separate radio to communicate with the Base Station. This type of system is generally used for two reasons.
- If the Base Station (or repeater) does not have clear line-of-sight to any of the slaves, the Q37 acts to extend the coverage of the base station,
- If I/O needs to be passed between the Master and the Slaves for functional reasons (e.g. a Reservoir and a Pump Station), then it is desirable to be able to do so without relying on the correct function of a base station or repeater. This makes the system more robust as the control of the reservoir will be maintained, even if the base station goes off line.
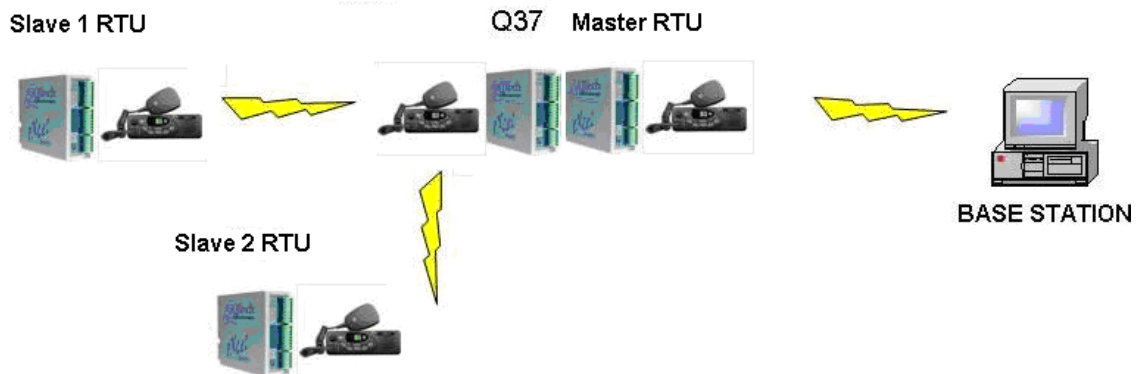
The Master RTU requires no special firmware, but requires configuration via Q90 to add the slave sites to its I/O own table.

The Q37 Comms module is required by the Master RTU to drive the second radio to communicate with the Slaves. The Comms module holds the polling intervals for the slaves and requires HyperTerminal as its configuration tool.

The Slave RTUs in the field are just standard Q22 RTUs, and they 'see' the Q37 at the Master RTU as if it were a Base Station. The I/O from the slaves appears at the base station as a subset of the I/O of the master site; The Base Station does not 'know' that the master site has slave RTUs.

Note: Both Master & Slave RTUs can have Q23 / Q26 Expansion Modules.
Note: Each RTU can have multiple Slaves.
Note: The Save RTUs can have a Q37 and their own slave sites.
Disadvantage: You can not RTU data log at the slave sites
Disadvantage: You can not send a DLP through the master to the slave RTU
Disadvantage: Commissioning these sites is difficult to achieve remotely

13.6 Radio Network Simplex

The system shown in the diagram below depicts all RTU sites receiving and transmitting on the same frequency. This is ideal for 'free-to-air' frequencies, but can have arbitration issues if 2 sites are trying to contact the Base Station at the same time, and they do not have line-of-sight to each other.

TX Freq A   RX Freq A

RX Freq A   TX Freq A   RTU 1

TX Freq A   RX Freq A   RTU 2

RX Freq A   TX Freq A
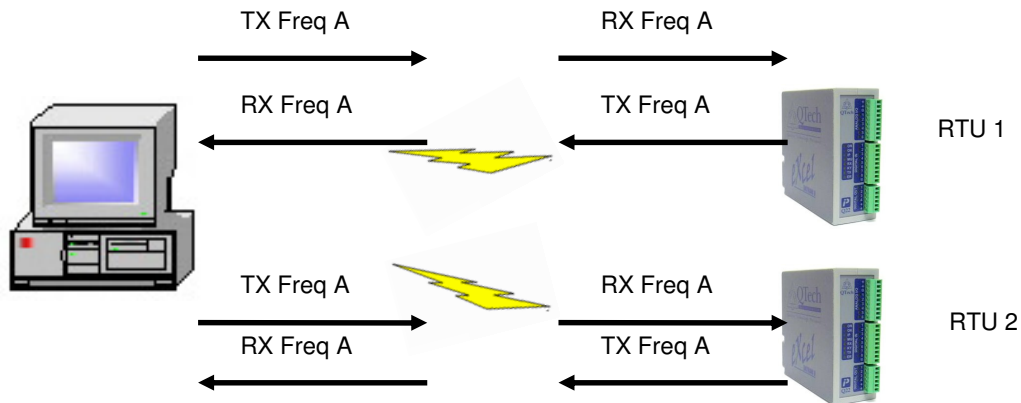
In cases where there are large numbers of sites, (i.e. a lot more radio traffic, a dual frequency system may be better.

13.7 Radio Network Duplex

The dual frequency system shown in the diagram below is typically used in situations where a repeater is necessary. Using a duplex channel without a repeater will result in even more problems than a simplex channel because it will guarantee that all sites will not be able to 'hear' each other, even if they have perfect line-of-site.

This dual frequency system, with repeater, is ideal when there are a large number of RTU sites, with a lot of radio traffic. There are less arbitration issues as a result.

TX Freq A   RX Freq A

RX Freq B   TX Freq B   RTU 1

TX Freq A   RX Freq A   RTU 2

RX Freq B   TX Freq B
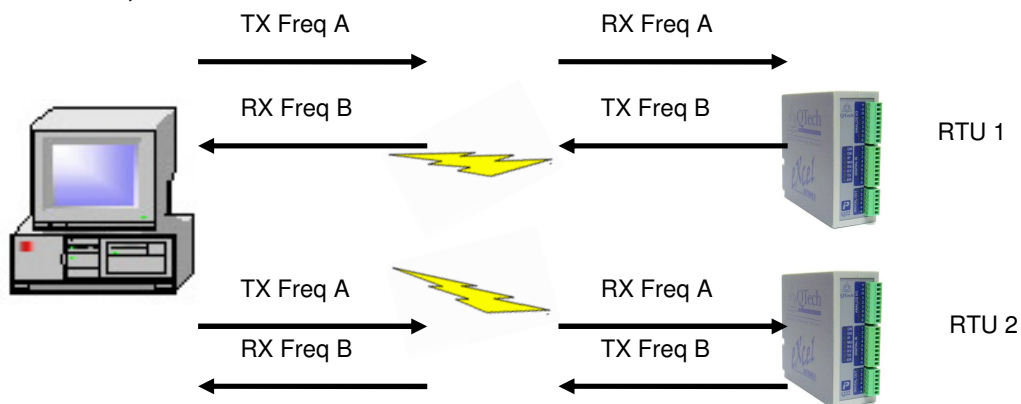
13.8 A "Typical" DATRAN SCADA System

In reality there is no 'normal' DATRAN system, but depicted below in the diagram is a common SCADA setup using Radios and/or Cellular/Dialup Modems.
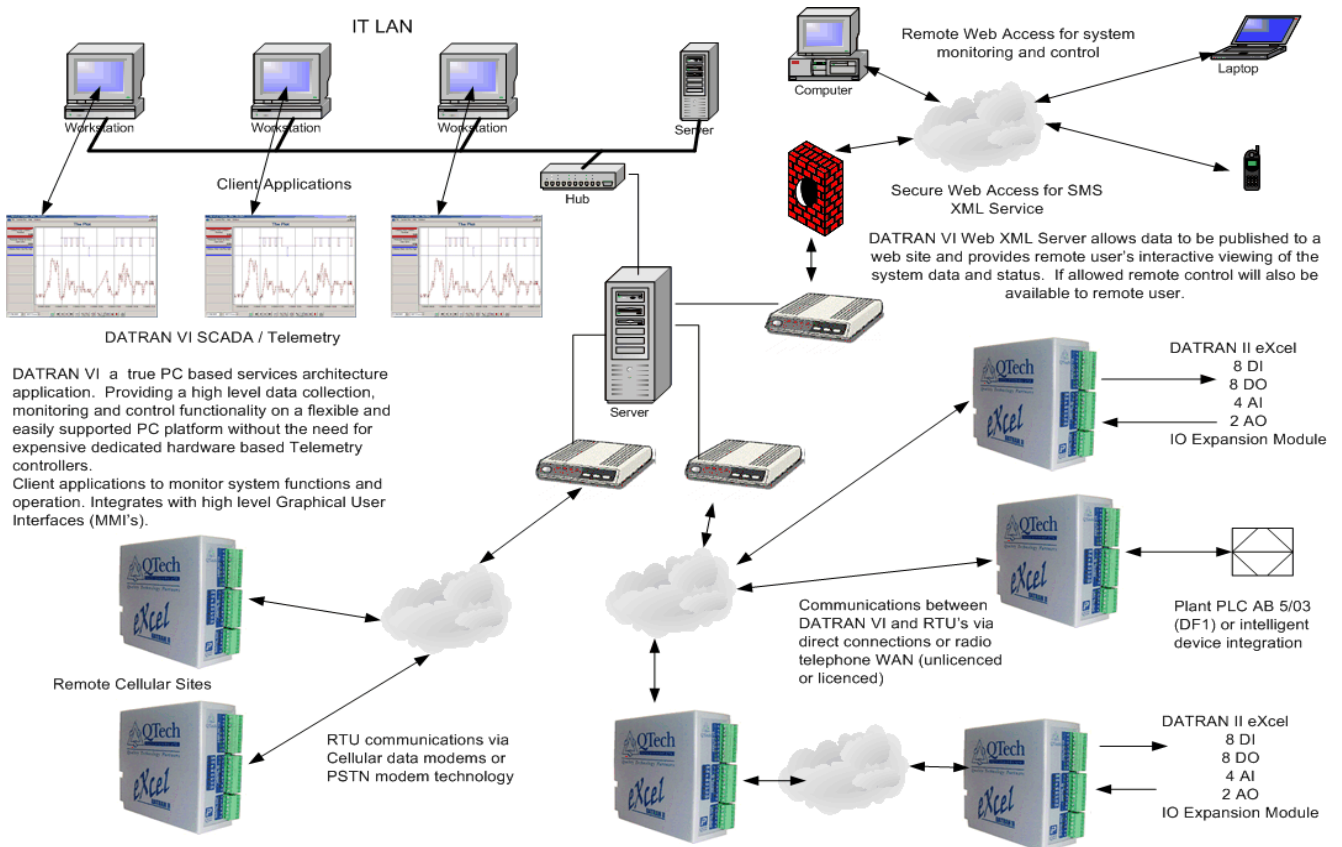
The Base Station / DATRAN Server is on a LAN, and multiple users (each with their own security level) can access Trending, Alarms and GUI screens.

The option for users/operators to have remote access via the Internet is available, with a firewall. Network security is of the utmost importance

Also featured in this setup is the option for SMS Alarms to an operator's cell phone. These Alarms can be acknowledged by the operators reply to a TXT/SMS message.

13.9 A Brief Base Station Overview

The Base Station can be built from a PC that is has the following minimum specs: 3GHz CPU, 1GB RAM (2GB+ if using a large database or Vista/Windows 7), 120GB HDD & Windows XP / Server 2003/2008 / Vista/ Windows 7.

The preferred and recommended machine is however a "server class" machine which is designed to provide high reliability and operate on a 24/7 basis.

The PC is loaded with DATRAN VI SCADA Software, including DATRAN Services as well as Applications such as: DATRANVue, DATRAN Reporting, DATRAN Supervisor etc.

Typically most DATRAN SCADA Systems only have one Base Station, and this gathers data and controls all remote RTU sites.
It is however possible for a DATRAN SCADA System to have two Base Stations. An 'Eavesdropping' Base Station can monitor Radio traffic and be used for monitoring RTU sites, but it has no ability to control.

A DATRAN SCADA system can have two fully operational Base Stations, but all the RTUs must be loaded with special firmware for the system to function without data duplication and control issues.

DATRAN Base Stations can also be configured to share data between them via DATRAN Replication.

For more information on DATRAN VI SCADA Software & Base Station configuration, please refer to the DATRAN Software Training Manual.

14.0 The DLP-Control Program

14.1 Overview

The DLP (Down Loaded Program) is a control program that can be loaded into an RTU to allow the RTU to carry out simple manipulation of the I/O and allows the RTU to be configured so it is capable of automatic control of equipment at a station.
Some of the tasks that the DLP can be used for are:

- Counting pulses on Digital Inputs.
- Delaying, or 'de-bouncing' digital inputs before they are sent to the base station.
- Generating alarms to be sent to the base station when analog values move past a certain pre-defined value.
- Turning digital outputs on and off in response to changes on the RTU inputs.

DLPs are written and compiled using a software package called DLP-IDE. This is a fully integrated DLP development environment that allows the user to code and compile DLPs. It also allows the user to upload DLPs into an RTU or even to extract and decompile a DLP from an RTU.

Once written, the DLP is compiled to produce a binary instruction block, which is either sent to an RTU from the base station or loaded directly into the RTU via Q90 or DLP-IDE.

All source code files have the extension .ASM.
All compiled files have the extension .DLP.

14.2 The Logic Accumulator

The key to understanding the way that DLPs work is the "Logic Accumulator". This is a system register that stores the true/false result of the previous logic operation. The state of the Logic Accumulator almost always affects the behaviour of each command in the DLP.

For Example, consider this small section from a DLP:

```
begin
    tce      rdin1
    setdig   rdout1
begin
    tce      rdin2
    resdig   rdout1
```

This DLP does two things, each with their own begin statement, signalling the start of a new "logic rung". This begin statement resets the state of the logic accumulator to "True".

The TCE statement stands for "test if contact is energised", in other words, is the nominated input turned on? The accumulator has to be true for the result TCE to also be true. So more specifically, TCE means "Is the accumulator true AND is the nominated input turned on?". The result of this check is then written to the logic accumulator.
The setdig and resdig commands can be expressed in English as "If the logic accumulator is true, set (or reset) the nominated digital output".

Using what we have learned above, we can break this DLP down as follows:

Begin:		Set the logic accumulator to TRUE.
TCE	RDIN1	Test If logic accum is true AND Digital input 1 is on. Write the result of the test into the logic accumulator.
Setdig	rdout1	If the logic accum is true, turn on digital output 1, else do nothing.
Begin:		Set the logic accumulator to TRUE.
TCE	RDIN2	Test If logic accum is true AND Digital input 2 is on. Write the result of the test into the logic accumulator.
resdig	rdout1	If the logic accum is true, turn off digital output 1, else do nothing.

In terms of functionality, this will turn on Digital Output 1 when Digital Input 1 is on, and will turn Digital Output 1 off when Digital Input 2 is on.

If both inputs are on at the same time, then the output will be turned off as the last section of the DLP will always take precedence when there is contention over a specific output.

14.3 I/O Types

When an RTU has no DLP, the I/O that is wired to the terminals of the RTU is directly connected to the I/O that appears at the base station. When a DLP is introduced to a system, it is possible to manipulate the I/O the base is sent such as is required when a digital input is de-bounced, for example.

The DLP not only allows digital inputs to be delayed before being sent, but it also allows for the I/O to be completely re-mapped if desired. Because of this, it has been necessary to create three different I/O types to differentiate between the I/O that is wired to the RTU, the I/O that is being sent to the base station and I/O that may be abstract values such as pulse counts or setpoints, which actually have no physical counterpart.

14.3.1   Real I/O

Real I/O represents the physical I/O as wired to the terminals on the RTU. In the DLP these are represented by names such as RDIN1 (real digital input 1) or RAIN1 (Real analog input 1). When there is no DLP installed in the RTU, these values are directly copied across to the base station and appear in the _Comms_Object_ in the base station as RDI and RAI values.

The same applies to RDO and RAO values for outputs.

See the Software Training Manual for a more detailed description of the base station _Comms_Object_.

14.3.2   Telemetry I/O

Telemetry I/O represents the I/O being passed between the RTU and the base station that would normally correspond to the real I/O when no DLP is present in the RTU. When a DLP is installed it is possible to intercept these values with the DLP and modify or completely change them. In the DLP these are represented by names such as TDIN1 (telemetry digital input 1) or TAIN1 (telemetry analog input 1). It must

be noted that these values are transmitted between the base station and the RTU and still appear in the _Comms_Object_ in the base station as RDI and RAI values.

The same applies to RDO and RAO values for outputs, which become TDOUT and RAOUT values in the DLP.
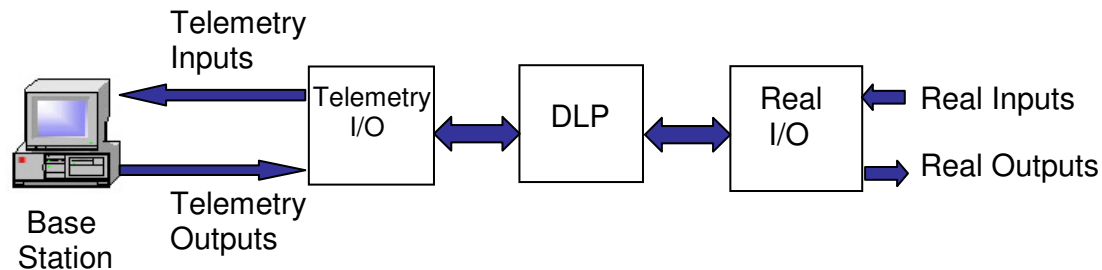
It is considered good coding practice to ensure that the value that a Telemetry I/O point represents is as close to its corresponding REAL value as possible. For example, if you have a door switch on digital input 2 that you are delaying for 30 seconds to use it as a security alarm at the base station, try to use TDIN2 as the delayed signal that gets sent back to the base station, not some other TDI register.

14.3.3 Notional I/O

Notional I/O is a set of I/O registers that are used for passing abstract values between the base station and the DLP. These should be used for values that do not have any real world I/O directly associated with them, for example:
- Alarm signals that are generated by comparing a RAI with a setpoint
- Setpoints that are used by the DLP for generating alarms
- Flow Pulse Counts
- Flow pulse or alarm reset commands
- Lockouts, Calls or other control signals that are being used to modify decisions the DLP is making.

14.3.4 Spare Registers

The DLP also has a set of spare Digital (SPD1) and spare Analog (SPA2) registers available for use as internal working registers for the DLP. These registers are not transmitted to the base and cannot even be viewed using Q90.

14.4 DLP Sections – PROGINIT, PROGSTART and PROGEND.

Any DLP is made up of two sections. The first is the Initialization (or INIT) section, which is only ever executed once at start-up. Start-up can be when the power is turned on, when the reset button is pressed, or when a new DLP is loaded into memory.

The INIT section of the DLP is started with the PROGINIT command, and it ends where the PROGSTART command is issued. The INIT section of the DLP is used for defining sensible names for all I/O registers, defining sensible initial values for each I/O point, and for setting up the COSMASK for the RTU.

```
001 ;********************************
002 ;*   Program Sections Example
003 ;*   (c) QTECH DATASYSTEMS 2009
004 ;********************************
005
006 proginit
007
008     telinp        ; Break the connection between the real inputs and the ones transmitted to base
009     telout        ; Break the connection between the real outputs and the ones transmitted from base
010
011     equ   rdin1    rdi_Pump1_Fault
012
013     initdig rdi_Pump1_Fault    false
014
015     ;
016     ;   All your other initilisation commands would be in this section between proginit and progstart
017     ;
018
019 progstart
020
021     ;
022     ;   The main body of your DLP would be in this section between progstart and progend.
023     ;
024
025 progend
026
027
```

In the screenshot above, the structure of a typical DLP is outlined. The EQU statement instructs the compiler that from that point forward, the programmer can use "rdi_Pump1_Fault" in place of "RDIN1". Doing so makes ASM files much easier to read.

> It is recommended good practice to equate every piece of I/O that the RTU has, even if it is not being used in the DLP.
> This provides a level of documentation for the system so that it can be referred to in the future.

14.5 Online Help and Code Examples

The complete instruction set and definition is contained in the online help file of DLP-IDE. Most DLP commands are an acronym of the function to perform. TCE means "Test if contact is energized", While TCD stands for "Test if contact is de-energized". You'll notice as you begin to write your own programs that analog comparison commands are different depending on whether you are comparing one register with another, or if you are comparing a register with a constant value. GTS stands for "Greater than Setpoint" while GTR stands for "Greater than Register". Each will throw an error at compile time if they are used with the incorrect argument types.

The next few pages show some examples of some commonly required functionality for inclusion in your DLPs.

14.5.1 _Digital Input Debouncing_

```
001 ;***********************************************************************
002 ;*  Digital Input Debounce Example                                     *
003 ;*                                                                     *
004 ;*  Written By  :  Daniel Wilkes (QTech Datasystems)                   *
005 ;*  Written On  :  14 August 2009                                      *
006 ;*  Purpose     :  To demonstrate the use of a DLP timer to debounce  a digital *
007 ;*                 input before forwarding it back to the Base Station.    *
008 ;*                 This DLP debounces both positive and negative edges on the  *
009 ;*                 High Float Input, but only a positive edge on the low float. *
010 ;*                 In practice, it is usually only necessary to debounce one   *
011 ;*                 edge.                                                 *
012 ;***********************************************************************
013
014
015 proginit
016
017     telinp                      ; These two commands instruct the RTU to sever the link between
018     telout                      ; the REAL and TELEMETRY I/O
019
020     equ     RDIN1   rdi_High_Level_Float    ; The high level float is wired into RDIN1
021     equ     TDIN1   tdi_High_Level_Float    ; The debounced input will be sent to the base on TDIN1
022
023     equ     RDIN2   rdi_Low_Level_Float
024     equ     tdin2   tdi_Low_Level_Float
025
026 progstart
027
028 ;   Debounce High Level Float
029
030 begin
031     tce     rdi_High_Level_Float     ; If the High Level Float is closed (energised)
032     tmr     1,150                    ; Start timer 1 for a duration of 150 x 100msec (15 seconds)
033     tce     timerd1                  ; If the timer has completed (stands for "Timer Done")
034     setdig  tdi_High_Level_Float     ; Set the telemetry back to the base station
035
036 begin
037     tcd     rdi_High_Level_Float     ; If the High Level Float is open (de-energised)
038     tmr     2,150                    ; Start timer 2 for a duration of 150 x 100msec (15 seconds)
039     tce     timerd2                  ; If the timer has completed
040     resdig  tdi_High_Level_Float     ; Reset the telemetry back to the base station
041
042 ;   Debounce Low Level Float
043
044 begin
045     tce     rdi_Low_Level_Float      ; If the Low Level Float is closed (energised)
046     tmr     3,150                    ; Start timer 3 for a duration of 150 x 100msec (15 seconds)
047     tce     timerd3                  ; and If the timer has completed (stands for "Timer Done")
048     ec      tdi_Low_Level_Float      ; set the telemetry back to base as true, else set to false.
049
050 progend
051
```

14.5.2 High Speed Pulse Counting

This code will only work on RTUs using a Q03 processor.

```
001 ;**********************************************************************
002 ;*   Pulse Analog Pulse Counting Example                              *
003 ;*                                                                    *
004 ;*   Written By  :  Daniel Wilkes (QTech Datasystems                  *
005 ;*   Written On  :  21 July 2008                                      *
006 ;*   Purpose     :  To demonstrate pulse counting using the RTU PANL  *
007 ;*                  Registers.                                        *
008 ;*                  This DLP will accumulate pulses on RDIN8          *
009 ;*                  and display the result on TAIN1                   *
010 ;*                  To reset the pulse count, toggle RDIN1            *
011 ;*                                                                    *
012 ;**********************************************************************
013
014
015 proginit
016
017     telinp
018     telout
019
020     pdiq8      :=    True              ; This tells the RTU to start pulse conting on RDIN8
021
022 progstart
023
024 begin
025     cpanl      panl8     spa1  1       ; Read from the PANL register and copy it to a spare analog for further use.
026                                        ; This automatically causes the PANL register to reset to 0
027     addr       spa1      tain1         ; Add the value from the PANL register to our running total
028     cpanl      anlacc1   tain1 1       ; and store the new total
029
030
031 begin
032     tce        rdin1
033     xor                                ; If RDIN1 has toggled
034     tce        spd1
035     xorend
036     jumpf      no-reset
037     initanl    tain1     0             ; Then reset our flow total
038 no-reset
039
040 begin
041     tce        rdin1
042     ec         spd1                    ; And remember what the value of RDIN 1 is for the next time we check it.
043 progend
```

14.5.3 Alarms derived from comparing analog values with setpoints

```
001 ;*******************************************************************
002 ;*   Alarms from Analog Inputs                                     *
003 ;*                                                                 *
004 ;*   Written By  :  Daniel Wilkes (QTech Datasystems)              *
005 ;*   Written On  :  14 August 2009                                 *
006 ;*   Purpose     :  To demonstrate the use of a DLP to generate alarms from *
007 ;*                  comparing the value from an analog input and a setpoint. *
008 ;*                  Two examples are shown, the first comparing to a static  *
009 ;*                  Value, the second comparing to a NAOUT being transmitted by *
010 ;*                  base station                                    *
011 ;*                                                                 *
012 ;*******************************************************************
013
014 proginit
015
016     equ     RAIN1     rai_Water_Pressure      ; The pressure transducer is wired into Real Analog In 1
017                                               ; The transducer is 4-20 mA = 0-10 Bar
018     equ     NAOUT1    nao_Pressure_High_SP    ; The setpoint from the base station for the high level alarm
019     equ     NDIN1     ndi_Pressure_High_Alm   ; The High Pressure Alarm will be transmitted back to the base on NDIN1
020     equ     NDIN2     ndi_Pressure_Low_Alm    ; The Low Pressure Alarm will be transmitted back to the base on NDIN2
021     equ     spd1      spd_Pressure_Low_Alm    ; A spare register to remember the state of a value the next iteration
022                                               ; of the DLP
023
024     initanl nao_Pressure_High_SP     45874    ; Initilise the setpoint for high pressure to 7 Bar
025
026 progstart
027
028 ;   High Level Alarm
029 begin
030     gtr     rai_Water_Pressure  nao_Pressure_High_SP  ; If the pressure is above the high setpoint
031     tmr     1,50                                      ; Start timer 1 for a duration of 50 x 100msec (5 seconds)
032     tce     timerd1                                   ; If the timer has completed (stands for "Timer Done")
033     tcd     ndi_Pressure_High_Alm                     ; and the alarm is not set yet
034     setdig  ndi_Pressure_High_Alm                     ; then set the alarm back to the base station
035     sendcos                                           ; and force the RTU to do a change-of-state
036
037 begin
038     ltr     rai_Water_Pressure  nao_Pressure_High_SP  ; If the pressure is below the high setpoint
039     tmr     2,50                                      ; Start timer 2 for a duration of 50 x 100msec (5 seconds)
040     tce     timerd2                                   ; If the timer has completed (stands for "Timer Done")
041     tce     ndi_Pressure_High_Alm                     ; and the alarm is currently set
042     resdig  ndi_Pressure_High_Alm                     ; then reset the alarm
043     sendcos                                           ; and force the RTU to do a change-of-state
044
045 ;   Low Level Alarm
046 begin
047     lts     rai_Water_Pressure    13107               ; If the pressure is below 2 bar (2/10 * 65535 = 13107)
048     tmr     3,100                                     ; Start timer 3 for a duration of 100 x 100msec (10 seconds)
049     tce     timerd3                                   ; If the timer has completed (stands for "Timer Done")
050     ec      ndi_Pressure_Low_Alm                      ; set the telemetry back to base as true, else set to false.
051 begin
052     tce     ndi_Pressure_Low_Alm
053     xor                                               ; If these two values are different from each other
054     tce     spd_Pressure_Low_Alm
055     xorend
056     sendcos                                           ; Force the RTU to do a COS message
057 begin
058     tce     ndi_Pressure_Low_Alm                      ; Make these two values the same so that next time around the
059     ec      spd_Pressure_Low_Alm                      ; COS will only happen if the Alarm changes state.
060
061 progend
```