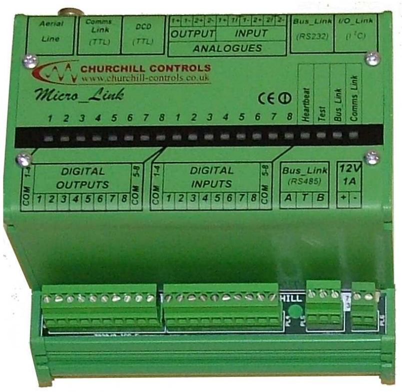
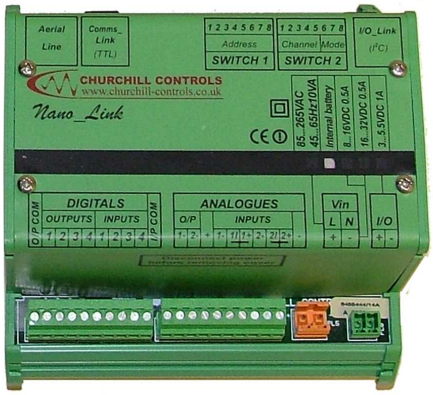
CHURCHILL CONTROLS
www.churchill-controls.co.uk

Data_Link 2000

TECHNICAL MANUAL

RADIO/LAND LINE TELEMETRY & TELECONTROL SYSTEM WITH PLC/SCADA INTERFACE





Issue	Date	Revision
1	15/09/97	
2	16/10/97	
3	27/10/97	***Micro_Link*** v1.0, ***Nano_Link*** v1.2
4	12/03/98	***Micro_Link*** v1.22, ***Nano_Link*** v1.21 I/O Expansion modules & Leased Line added
5	01/09/98	***Micro_Link*** v1.30, ***Nano_Link*** v1.30 Section 9 changed to use data routing parameters
6	28/04/99	Updated to cover ***Micro_Link*** V1.40 & ***Nano_Link*** V1.40 software features. Also ***Nano_Link*** chapter extracted to dedicated manual
7	23/10/00	Updated to cover ***Micro_Link*** V2.00, ***Nano_Link*** V2.00 and DCD V2.00. New features include exception reporting and enhanced DCD.
8	01/03/01	Updated to cover ***Micro_Link*** V2.01 and ***Nano_Link*** V2.01. New features include battery repeaters and Mitsubishi protocol.
9	22/01/02	Enhances exception reporting
10	04/09/02	Format improved. Outstation comms fail alarm time-out changed
11	01/04/03	Initiation Delay option added (requires ***Micro_Link*** V2.23, ***Nano_Link*** V2.20 and DCD V2.13 or higher).
12	13/09/04	Incorporates alternative radio bands and mapping digitals to registers on Modbus
13	19/01/05	Add need for catching diodes on digital outputs, and delete 7150-3
14	20/07/05	Add Dual-Comms functionality
15	20/07/05	Add in Expansion Comms Alarm & Watchdog
16	07/09/05	Revise obsolete part nos. Add in addressing notes for battery repeaters.
17	27/02/06	Remove Modbus ASCII, expand Allen-Bradley
18	09/11/07	Add alternative method of copying digitals to registers (See 7.5.2). V3.10 firmware and later.
19	08/02/08	Mitsubishi Protocol enhanced
20	21/07/08	Allen-Bradley DF1 Protocol enhanced (V3.18 firmware and later)
21	27/02/09	Calculating Average Flow Rate From Pulse Flowmeters enhanced (see 8.11) Generating Pulse Outputs enhanced (see 8.12)

The contents of this document must not be disclosed to any third party without the written consent of Churchill Controls Limited, nor are they to be used for any purpose other than to configure and maintain equipment supplied by Churchill Controls Limited.

Whilst every attempt has been made to ensure the accuracy of this document, Churchill Controls Limited will not be held liable for any errors or omissions.

As part of our policy of continuous improvement we would welcome any suggestions for changes to the document.

Churchill Controls Ltd.
30 Wellington Business Park
Crowthorne
Berkshire
RG45 6LS
England

Tel: 01344 750233
Fax: 01344 752304
e-mail: sales@churchill-controls.co.uk
website: www.churchill-controls.co.uk

TABLE OF CONTENTS

1. Preface

This manual is intended to give the installer, user and maintenance personnel all the information they are likely to need for implementing telemetry systems from the ***Data_Link 2000*** product range.

The ***Data_Link 2000*** product is based on ***Nano_Link*** (which is a simple, low power, self-contained unit) and ***Micro_Link*** (which is a more complex product with considerable expansion capacity and many configurable functions). This manual concentrates on ***Micro_Link***, and its relationship with ***Nano_Link***. If a system uses both ***Nano_Link*** and ***Micro_Link***, this manual should be read in conjunction with the '***Nano_Link* Technical Manual**'. If a system does not use ***Micro_Link***, then this manual is irrelevant.

The ***Data_Link 2000*** product range is subject to continuous evolution, so new features are constantly being added by a combination of software enhancements and new hardware modules. This manual describes the features of ***Micro_Link*** software version 7030 V2.01, ***Nano_Link*** software version 7020 V2.01 and DCD software version 2.00. Some of the features described may not be available, or may function slightly differently, in earlier software versions. However, it is Churchill Control's policy to ensure that wherever possible software is backward compatible. This means that wherever possible the features described will be present in all future software issues.

The software version fitted within ***Micro_Link*** by running DCD Diagnostics on a PC connected to ***Micro_Link*** and examining the Station Status by pressing the 'S' key. The software version fitted within ***Nano_Link*** can be found removing the cover and examining the label on the EPROM or via an Alpha-numeric Display module connected to the I/O_Link port. The software version of DCD Configuration can be examined by selecting 'About DCD Configuration' from the Help menu.

Every attempt has been made to lay out this manual in a logical sequence. Chapters 2 and 3 describe the features of the system, and should be read before designing a system around ***Data_Link 2000***. Chapters 0 and 4 describe ***Micro_Link*** in detail, and should be read before commencing installation and wiring. Chapter 5 explains the method of installation.

Chapter 6 described the method of configuring ***Micro_Link*** to suit each individual application. Chapter 7 describes the method of communicating with PLC's, SCADA systems etc, using the serial interface designated ***Bus_Link***.. Chapter d) describes the special features of ***Micro_Link*** that can be configured in software. Finally, chapter 9 described the configuration and diagnostic software that a user can run on a computer to assist with the setting up of a system.

Before shipping any ***Data_Link 2000*** system, Churchill Controls configure it to their best understanding of the specific application requirements, and test it as a complete system. After installation it should therefore begin operation immediately. If the configuration does not meet the requirements of the application then the user can modify the configuration after reading chapter 6. The configuration is defined and downloaded to ***Micro_Link*** using the DCD terminal, as described in chapter d).

2. Data_Link 2000 Overview

2.1 Introduction

The ***Data_Link 2000*** telemetry system offers a very flexible approach to the problem of transferring signals between industrial plant on distributed sites. By utilising radio or private wires as a communication medium it offers low installation charges and minimal ongoing running costs. Applications range from simple point to point systems to large schemes interfacing with PLC's and SCADA systems.

Data_Link 2000 configures an industrial data communications network from ***Nano_Link*** and ***Micro_Link*** units. ***Nano_Link*** has been optimised for low power consumption, and low cost applications with limited I/O. ***Micro_Link*** is a more powerful device with additional communication ports and expansion capacity.

The simplest network comprises two ***Nano_Link*** modules providing a point-to-point link transferring 4 digitals and 2 analogues in each direction.

More complex networks are built around a ***Micro_Link*** base-station and any combination of ***Micro_Link*** and ***Nano_Link*** outstations. The base-station can poll up to 250 outstations, and every outstation can be used as a repeater if required to access more distant outstations and/or repeaters. The data routing is fully configurable, with any input on the network being available at any output, and data transparently passed to/from PLC's and SCADA systems.

2.2 Nano_Link

Nano_Link is a general-purpose outstation with 4 digital inputs and 2 analogue inputs, with the option of also including 4 digital outputs and 2 analogue outputs. It is housed in a compact plastic enclosure, which can be clipped onto a standard DIN rail. It can be supplied in waterproof polycarbonate enclosures if required, or in a roadside cabinet.

The internal power supply can be either three 'D' alkaline batteries or a mains power supply with battery back-up. The module has been optimised for low power consumption, to the extent that it will operate for up to 3 years on low-cost Duracell batteries. Battery operation has the advantage of significantly lower installation and running costs.

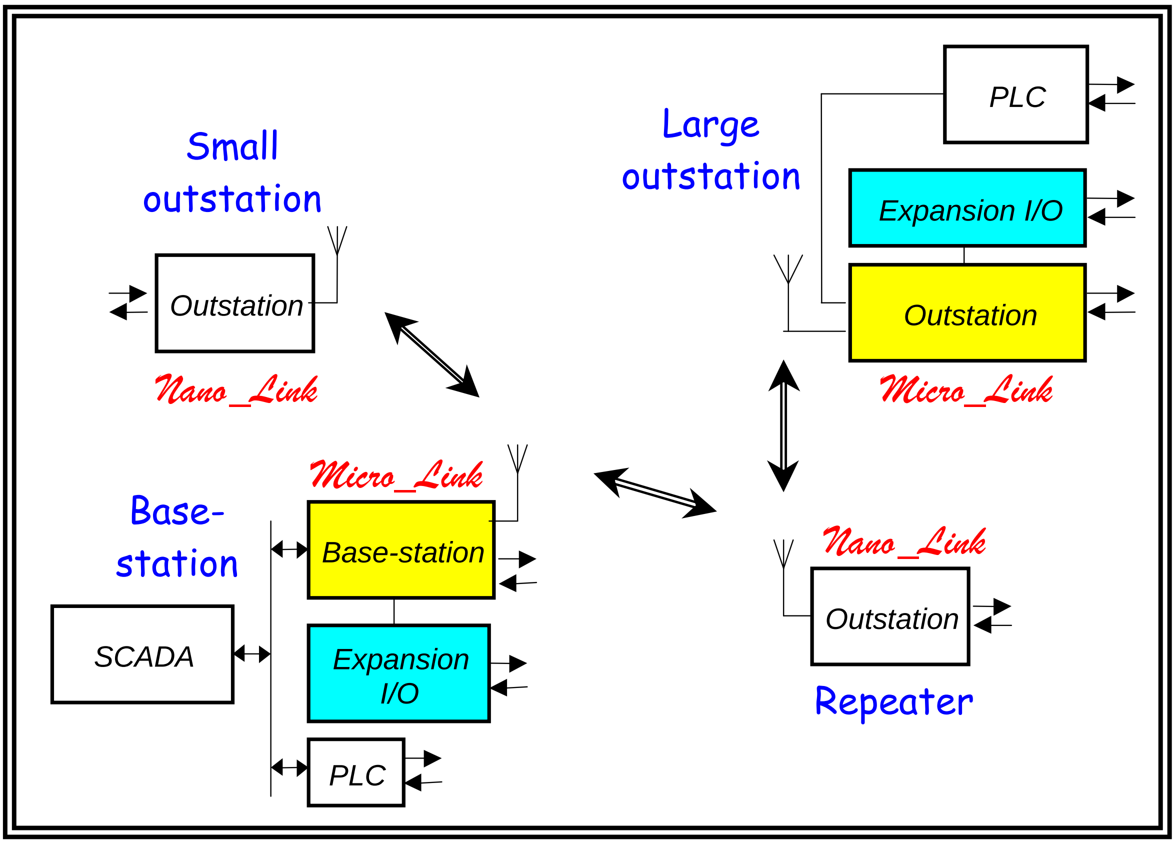
The four digital inputs can each be used to monitor alarm or status conditions, and also to count pulses from totalised flow transducers. Extra pseudo inputs can be used to monitor alarm conditions.

The two analogue inputs can be used with low-cost millivolt transducers or more conventional current loop transmitters. A pseudo analogue input monitors the battery voltage and another monitors the radio receive signal strength.

On fully equipped variants the four digital outputs are volt-free relay contacts, and the two analogue outputs source 0…20mA.

The capacity can be increased to 20 digital inputs and/or 20 digital outputs by adding optional digital expansion modules.

Nano_Link is described in detail in the '***Nano_Link*** Technical Manual'.



Although ***Nano_Link*** is intended primarily for use as an outstation or repeater, it can be configured as a base-station to provide a simple point-to-point link to another ***Nano_Link***, transferring up to 2 analogues and 20 digitals in both directions.

However, to access pseudo inputs, use repeaters, or use more than one outstation, a ***Micro_Link*** base-station should be used:

2.3 Micro_Link

Micro_Link is mechanically similar to ***Nano_Link***, but has been designed to provide expansion capacity, more processing power and more communication ports with the trade-off of higher power consumption. It has eight digital inputs, the first four of which double as pulse count inputs. Extra pseudo digital inputs can be used to monitor alarm conditions. It also has two fully isolated analogue inputs compatible with transducers that give either voltage or current outputs. Its eight digital outputs are volt-free relay contacts, and it has two 0…20mA analogue outputs. The I/O capacity can be increased virtually without limit by the addition of I/O expansion modules. A range of expansion modules is available to cater for a variety of plant I/O types.

Micro_Link's ***Bus_Link*** communication port allows it to communicate with a wide variety of PLC's and SCADA systems using proprietary serial protocols such as Modbus.

The Data_Link Configuration and Diagnostic (DCD) port can be connected to an IBM compatible PC running DCD terminal software (free-issued by Churchill Controls) for configuring the system and/or running diagnostics to analyse problems.

Micro_Link can be configured to function as a base-station or an outstation. As a base-station it can communicate with up to 250 outstations. In either configuration it can be equipped with expansion modules to increase its input/output capacity without any practical limits.

2.4 Repeaters

Every ***Micro_Link*** and ***Nano_Link*** outstation in a network is capable of repeating messages to/from more distant outstations. Up to 8 levels of repeating can be used to greatly extend radio range. The network is fully configurable from the base-station.

2.5 Hardware I/O

Data_Link 2000 can copy inputs at any point on the network to outputs at any other point, maintaining the same electrical format. It can therefore be visualised as the equivalent of a multi-core cable network. ***Micro_Link*** can also be configured to pass data to/from PLC's, SCADA systems and regional telemetry schemes via the ***Bus_Link*** port, thus obviating the need for hardware I/O.

2.6 Bus_Link

Bus_Link is Churchill Controls' terminology for a general-purpose serial interface, which can be configured to use a standard PLC bus protocol. It allows data to be passed between ***Data_Link 2000,*** intelligent instruments, PLC's, SCADA systems and third party telemetry schemes. Although the industry-standard is Modbus, other protocols such as Allen-Bradley can also be used.

2.7 DCD Terminal

The Data_Link Configuration and Diagnostic terminal comprises software free-issued by Churchill Controls for running on any IBM compatible computer as a diagnostic aid, and for configuring ***Data_Link 2000***.

In configuration mode the software allows all system, network and ***Bus_Link*** configurable parameters to be edited, saved to disc, downloaded to ***Micro_Link*** and uploaded from ***Micro_Link***.

In diagnostic mode it allows ***Micro_Link***'s inputs and outputs, and its internal database, to be examined and modified, and serial communications with ***Bus_Link*** and with other ***Micro_Link***'s and/or ***Nano_Link***'s to be eavesdropped.

2.8 Expansion Modules

The I/O capacity of ***Micro_Link*** can be increased by adding I/O expansion modules. The modules currently available are:

- **Digital Input**
  This module has 16 inputs, each designed to interface to external volt-free contacts, or to monitor control voltages in the range 0…24VDC
- **Digital Output**
  This module has 16 outputs, each being a volt-free relay contact.
- **Digital Input/Output**
  This module is a combination of the above, with 8 inputs and 8 outputs.
- **Analogue Input**
  This module has 8 inputs, all fully isolated and scaled for 0…20mA or 0…5V (user selectable). They read the inputs with an accuracy of ±0.1%.
- **Analogue Output**
  This has 4 active outputs, all fully isolated and generating a current in the range 0…20mA at ±0.1% into loads of up to 1000Ω.
- **Display Module**
  Two forms of this are available: The digital version has a 4-decade seven-segment LED display with selectable decimal point, so is capable of indicating values in the range 0.001…9999. The alphanumeric version has a 2 line x 16-character LCD display and four pushbuttons. Its main use is as a diagnostic tool.

These modules are described in more detail is Chapter 4.

Micro_Link can be fitted with up to 32 expansion modules of any mix, plus a further 32 digital display modules if required.

Nano_Link's expansion capacity is limited to one digital input module, one digital output module and/or one alphanumeric display module. For more details please refer to the '***Nano_Link*** Technical Manual'

2.9 District Monitoring

Data_Link 2000 has been specifically designed to address the need for district monitoring to detect leaks in water distribution networks. The total cost of ownership has been considered by addressing installation, operational and maintenance costs as well as the capital cost of the equipment.

District monitoring requires remote reading of totalised and/or instantaneous flow from one or two transducers, with the option of also monitoring pressure, at numerous points around the network. In urban areas the density could be up to one outstation per square kilometre, whilst in rural areas the network could spread over many tens of square kilometres. Outstation aerials must be small and protected against vandalism, since they are likely to be in public locations. In some instances they must be under manhole covers.

The ***Nano_Link*** outstation is optimised for district monitoring applications by incorporating the following features:

- *BATTERY OPERATION*
  saves the cost of connecting to mains supply. Three-year life from Duracell batteries minimises operational costs.
- *USES DE-REGULATED RADIO*
  No recurring charges.
- *WORKS WITH LOW-COST TRANSDUCERS*
  Outstations derive instantaneous flow from turbine or electromagnetic meters and pressure from strain gauge transducers. All transducers can be powered from the outstation battery.
- *BI-DIRECTIONAL COMMUNICATION*
  Allows outstations to be remotely configured, and gives more reliable data transmission.

These features are complemented by using ***Micro_Link*** as a base-station to give the following facilities:

- *BUS INTERFACE AT BASE-STATION*
  Allows access by a wide range of systems using industry-standard protocols such as Modbus.
- *FAST BI-DIRECTIONAL COMMUNICATIONS*
  Allows 100 outstations to be scanned by each base-station every 15 minutes
- *REPEATER OPTION*
- *Allows extended area coverage, with easily configurable message routing*

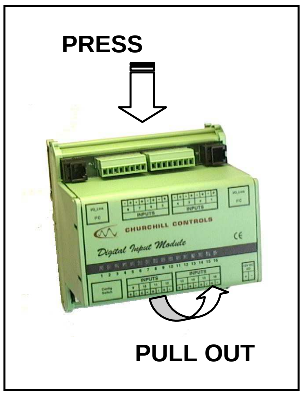
3. Common Features

3.1 Mechanical

The following features are common to both ***Micro_Link*** and ***Nano_Link***:

3.1.1 Housing

All modules are housed in plastic cases, which can be clipped onto either G or 'top hat' DIN rails. The housings measure 125mm H x 125mm W x 110mm D (when mounted on a vertical surface). All electrical connections are made through two part screw terminals along the top and bottom edges. Connections between modules and with other devices are made through FCC68 RJ11 and RJ45 jacks. No internal circuitry is accessible without removing the top cover.



The top cover is retained by internal clips, and can be removed by inserting a fingernail or screw-driver into the centre of the gap at each side of the cover and pulling outwards. Some modules also have screws.

To unclip modules from 'top hat' rails on a vertical surface, press down on the top of the module and lift it out from the bottom.

3.1.2 Polycarbonate Enclosures

A range of polycarbonate enclosures is available for applications requiring wall mounting. They range in size from 190mm x 190mm x 135mm (suitable for a single module) up to 380mm x 560mm x 135mm (for up to 4 modules). The smallest enclosure is rated IP68 (submersible for 24 hours to a depth of 2m), whilst all others are IP67 (resistance to temporary submersion to a depth of 1m). The enclosures are fitted with DIN rails to which the unit can be clipped and a cable to extend the aerial connector to a TNC aerial socket on the top of the enclosure. It is assumed that the user will fit any glands needed to bring plant I/O into the enclosure.

It should be noted that the IP rating is dependent on the lid being screwed down tightly, and all glands being correctly fitted and secured.

Larger systems can be supplied in steel enclosures if required.

3.2 Network Communications

3.2.1 Radio

Both ***Micro_Link*** and ***Nano_Link*** can be equipped with synthesised VHF or UHF transceivers which are approved to ETSI standard EN 300 220-1 and can thus be used so can be used in all the European Community member states (subject to national requirements). In the UK the de-regulated bands are designated MPT1328 (VHF) or MPT1329 (UHF) bands. Any of 32 channels can be selected to avoid conflict with other users.

The UHF MPT1329 band is the most popular, since it allows for transmit powers of up to 500mW ERP (Effective Radiated Power), but doesn't require a licence. There are 32 channels at 12.5KHz spacing, in the band 458.500MHz to 458.925MHz. However, other countries allocate a different part of the UHF spectrum for de-regulated applications, so variants of the radio can be fitted to accommodate national requirements.

The radio range depends on the aerials used and the topography of the area but will typically be up to 8Km in urban environments and up to 25Km with elevated aerials.

There are two adjustments on the radio modules, Modulation Level and Transmit Power, both of which are factory set:

The Modulation Level trimmer sets the transmitter frequency deviation, and should never be changed.

The Transmit Power trimmer sets the output power in the range 20mW to 1W, and is pre-set to 500mW. *It can only be changed by qualified personnel equipped with a calibrated UHF power level meter*. The output level can be set below 500mW to minimise power consumption and reduce potential interference with other users. It must be set below 500mW if the aerial has gain. It can only be set above 500mW if the aerial and/or aerial feeder has an overall gain of less than unity, and then only to compensate for the losses. **The ERP must never be set to more that 500mW when using MPT1329.**

Other versions can be supplied to operate within the regulated MPT1411 band of 457.5MHz to 458.5MHz and 463.0MHz to 464MHz at power levels up to 5W, or the unregulated MPT1328 band of 173.20MHz to 173.35MHz at a power level of 10mW.

3.2.2 Leased Line

Micro_Link can alternatively be equipped with modems for use on leased telephone lines supplied by national network providers such as BT and Mercury. The modems are approved for use through the European Union.

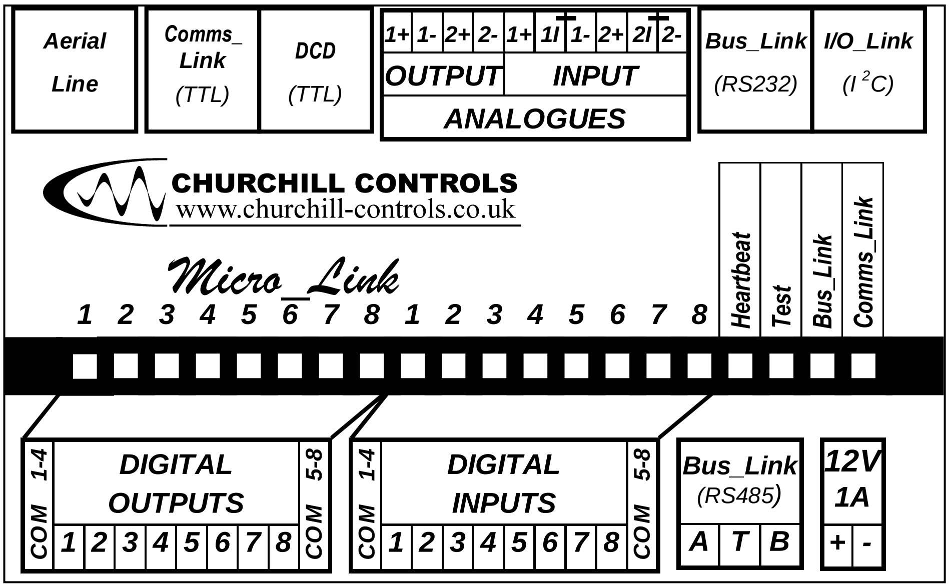
The modems are capable of operating with an end-to-end loss of up to 27dB, and provide correct impedance matching in accordance with the regulations. Normal telephone wires have a loss of about 1.5dB per Km, giving a range of typically 18Km. However, if more than one outstation is used, the line becomes mismatched, thus increasing the loss and effectively reducing the maximum line length. This can be overcome by adding line amplification and/or impedance matching pads. The network provider will normally arrange this.

3.2.3 Private Wire

Private wires are cable pairs similar to leased lines, but owned by the user. The user is thus responsible for maintenance of the cable, but does not have to pay rental charges.

Over short distance the characteristics of the cable are unimportant. However, for distances greater than around 2Km the cable should be matched to the terminating impedance of the modems, namely 600Ω. The modems are capable of operating with an end-to-end loss of up to 27dB. Normal telephone wires have a loss of about 1.5dB per Km, giving a range of typically 18Km. However, if more than one outstation is used, the line would become mismatched, due to the additional load impedance. This increases the loss and effectively reduces the maximum allowable line length. A high impedance version of the modem is available to overcome this and allow large numbers of outstations to be connected across a single pair of wires, with appropriate terminations at each end of the cable.

Micro_Link



3.3 General

Micro_Link is a general-purpose module with 8 digital inputs, 8 digital outputs, 2 analogue inputs and 2 analogue outputs. It can be fitted with an internal radio module or leased-line modem, or used with an external communications device. It also has a serial port that can be configured for either RS232 or RS485 communication with external devices such as PLC's or SCADA systems. An ***I/O_Link*** port is also included for connection to I/O expansion modules.

Micro_Link's function is determined by internal software, which is configurable by using DCD software running on a computer plugged into its DCD port. This facility can also be used for monitoring and debugging the system.

3.4 Indicators

Micro_Link has a row of 20 LED's, visible through windows in the top cover.

The first eight LED's monitor the digital outputs, and are lit when the respective output contact is closed.

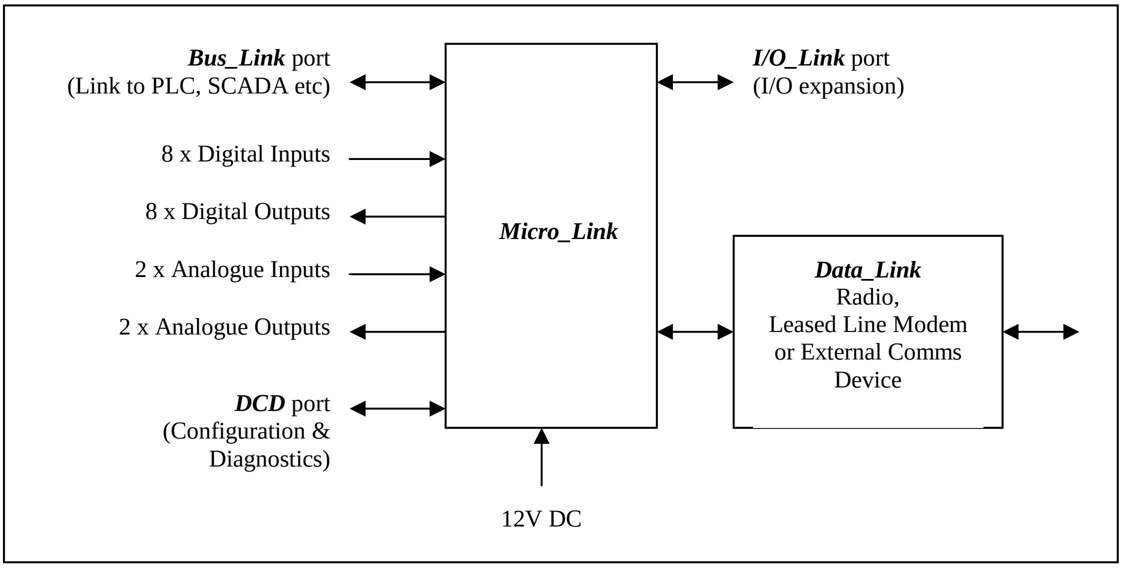
The next eight LED's monitor the digital inputs, and are lit when the respective external input contact is closed.

The Heartbeat LED flashes at a rate of 4 pulses/second to indicate that the unit is operating.

The Test LED lights whenever the radio transmitter is enabled.

The ***Bus_Link*** LED flashes green when data is being sent to the ***Bus_Link*** port and red when data is received from it.

The ***Comms_Link*** LED flashes green and red when data is being sent to the ***Data_Link 2000*** communications network, and red when data is received from it.



3.5 Power Supply

Micro_Link operates from a supply of nominally 12V DC. Current consumption is typically 20mA in normal operation, rising to 500mA when the radio transmitter is enabled. If expansion modules are fitted their power consumption will be added to these figures.

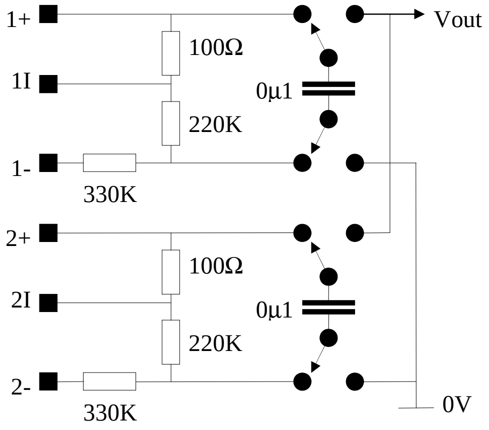
3.6 Plant I/O:

3.6.1 Analogue Inputs

Micro_Link has a total of four analogue inputs (two real and two pseudo):

3.6.1.1 Analogue Inputs 1 and 2

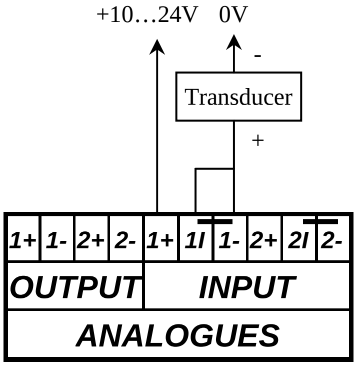
These inputs are identical and fully isolated. Their internal circuit is as follows:



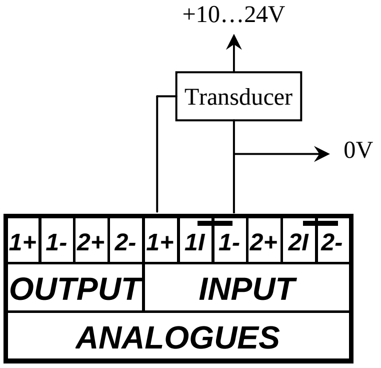
Vout is measured relative to an internal 2.00V reference, giving an input range of 0...5.000V between 1+ and 1- or 2+ and 2-. However, if an external link is fitted between 1I and 1- or 2I and 2-, the respective input sensitivity changes to 0...20mA into 100Ω (compatible with 0…10mA and 4-20mA transducers). The span and offest can be changed, as described in section 8.7.

The ADC has a resolution of 12-bits (0…4095), and an overall accuracy of ±0.1%. It is calibrated to give a reading of 4000 for a full-scale input so over-range inputs can be detected.

It should be apparent that the transducer in current loop application can be either in the negative leg, as illustrated, or in the positive leg. In all cases the power for the transducer can be either sourced externally or derived from the system 12V supply.



Current Loop Transducer



Voltage Output Transducer

3.6.1.2 Pseudo Analogue Input 1 - Supply Volts

Micro_Link measures its supply voltage, and stores it as if it was an additional analogue input, scaled so 0…20V is read as 0…16000:

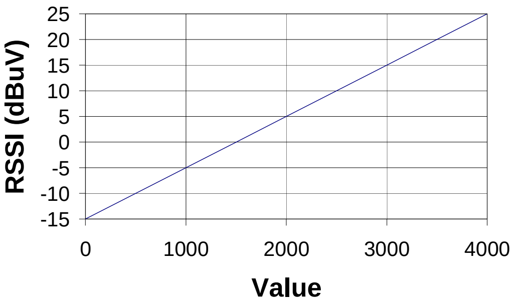
Supply Voltage = Value / 800 V

3.6.1.3 Pseudo Analogue Input 2 - Radio Received Signal Strength Indicator (RSSI))

When using radio communication it is important to know the strength of the received signal. The RSSI can be used to deduce the margin by which the path can deteriorate before communication will be lost. RSSI is measured in dBµV, according to the following formula:

RSSI = Value / 100 - 15 dBµV

This can be illustrated graphically as follows:

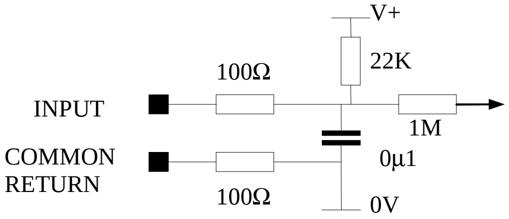


The radio receiver has a sensitivity of about -10dBμV, and a good radio path should have a margin of at least 10dBμV so the RSSI should ideally read at least 1500.

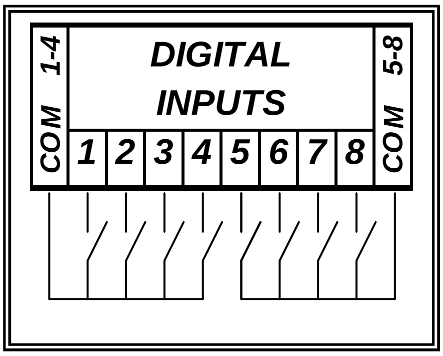
Note that the reading is the signal strength of the last message received. The value at a base-station will therefore change each time it interrogates a different outstation.

3.6.2 Digital Inputs

Micro_Link has 8 identical digital inputs, with a common return, intended for use with volt-free contacts. The input circuit is as follows:



Two Common Returns are provided, one for inputs 1-4 and the other for inputs 5-8.



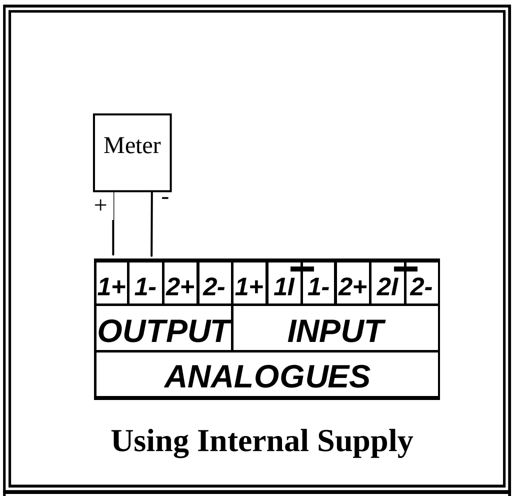
The series resistors give protection against EMC and excessive voltages on input terminals. The capacitor serves the dual purpose of providing a surge wetting current when the input contact first closes and providing switch bounce filtering.

Micro_Link records the current state of each digital input. In addition, it maintains an internal 16-bit counter for each of the first four inputs, which is incremented each time the contact closes. Each input can therefore be used to monitor a digital state or to count input pulses. The count is copied into non-volatile memory every 10 minutes, so in the event of a total power failure in the worst case only 10 minutes of counts will be lost.

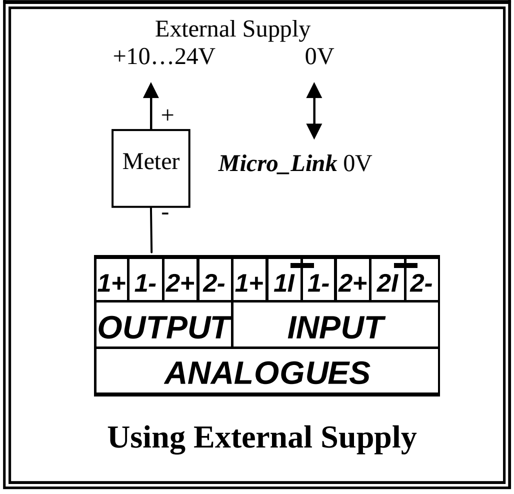
The counters can be reset to zero using a special test mode (see 8.14.3).

3.6.3 Analogue Outputs

Micro_Link has 2 analogue outputs, each sinking a current of 0…20mA (although the span and offset can be changed as described in section 8.8) through the '-' terminal to the internal 0V rail. The current can be sourced from ***Micro_Link***'s operating 12V supply (which is presented via an EMC filter on the respective '+' terminal) or from an external supply of up to 24VDC:



Using Internal Supply



Using External Supply

The outputs are calibrated to give full scale for a digital value of 4000, so will sink 20.47mA for a maximum reading of 4095, allowing over-range values to be monitored.

Note that all analogue outputs are initialised to zero on power-up. During normal operation they will copy the appropriate output value. If there is a subsequent communications failure the output value may obviously no longer be correct. The user can configure each analogue output to freeze at its last valid state in the event of a comms failure, or to go to any required value. The means of configuring this is given in section 8.4.

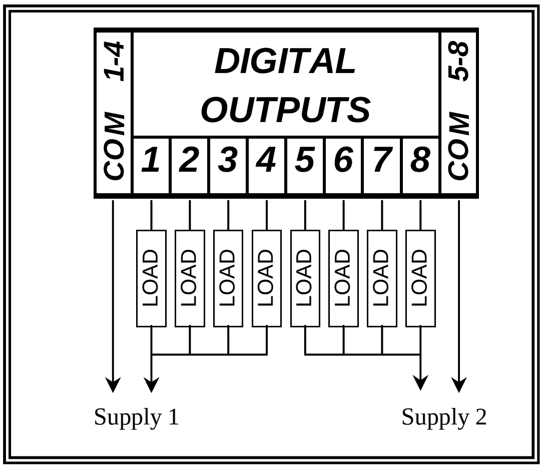
3.7 Digital Outputs

Micro_Link has 8 digital outputs, each being a volt-free relay contact rated 125VAC/1A/60VA max. or 60VDC/1A/30W max. Outputs 1…4 share a common return, as do outputs 5...8, all being fully isolated from the internal circuitry. Each relay contact also has a 120V transient suppressor in parallel.

Although the switch contacts are rated at up to 125V, it is the user's responsibility to ensure that external power sources do not compromise the operator's safety.

If the relays are used to switch inductive loads, such as interposing relays, the load must include transient suppression to prevent excessive voltages during switching. If the load is DC, this is most easily achieved by connecting a reverse-biased diode across the load. If it is AC, a bi-directional suppressor such as transorb or a voltage dependent resistor should be used

Note that all digital outputs are initialised to OFF on power-up. During normal operation they will copy the appropriate output value. If there is a subsequent communications failure the output value may obviously no longer be correct. The user can configure each digital output to freeze at its last valid state in the event of a comms failure, or to go ON or OFF. The means of configuring this is given in section 8.3.



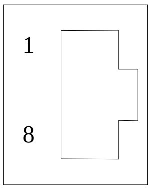
3.7.1 Bus_Link Port

The ***Bus_Link*** port is used for communication with PLC's, SCADA systems and larger telemetry schemes. It is configured through the DCD terminal, and presented in either RS232C or RS485 format. The RS232 interface is provided through an 8-pin RJ45 socket, configured as follows: Columns 3 and 4 show typical connections to a personal computer, column 5 shows the three connections needed for the RS232 connection to an AB processor whilst column 6 shows that connection to a Modicon Micro PLC requires only an 8-way RJ45 reverse cable (linking pin 1 - 8, 2 - 7, 3 - 6, etc.). Connection to any other device should be readily derived from this table.

The drawing shows the numbering convention adopted by ***Micro_Link*** for RJ45 sockets. Ready-made cable assemblies can be purchased from Distributors such as RS Components and Farnell Electronic Components. Typical stock numbers are as follows:

	RS
Flat 3m	446-664
5m reversed	405-5391
Category 5 1m	405-4590
Category 5 2m	405-4629
Category 5 5m	405-4641
Category 5 10m	405-4567

RJ45 Pin Numbering



View looking into socket

Colour code

	Category 5	**Flat cable**
1	Brown	Grey
2	White/Brown	Brown
3	Green	Yellow
4	White/Blue	Green
5	Blue	Red
6	White/Green	Black
7	Orange	Orange
8	White/Orange	Blue

Category 5 cables are recommended, since they are more robust, and can be terminated into conventional connectors at one end if required. For connection to a D-Type plug or socket an RJ45 adapter (e.g. RS 452-6736 or Farnell 225-162) can be used.

If ***Micro_Link*** has been configured via the DCD terminal to use the RS485 ***Bus_Link*** interface the I/O is directed instead to a 3-pin connector on ***Micro_Link***, with pins designated A, T and B. RS485 is a balanced two-wire bus that can be used to both transmit and receive data. The line can be multi-dropped to 32 devices, and should be terminated at both ends with 100Ω. ***Micro_Link*** has a 100Ω resistor between the A and T terminals, so linking T to B will incorporate a termination resistor. Note that RS485 is polarity-conscious, so all A terminals must be joined together, as must all B terminals.

	Bus_Link (RS232) RJ45	**PC 25-way D**	**PC 9-way D**	**Allen-Bradley 9-way D**	**Modicon PLC RJ45**
+5V					← 8
NC	1	8-19, 21-25	9		
DSR	2 →	→ 6	→ 6		→ 7
TXD	3 ←	← 2	← 3	→ 3	← 6
RXD	4 →	→ 3	→ 2	← 2	→ 5
GROUND	5 <>	<> 7	<> 5	<> 5	<> 4
RTS	6 ←	← 4	← 7		← 3
CTS	7 →	→ 5	→ 8		→ 2
Shield	8 <>	<> 1	<> 1		<> 1
DTR		← 20	← 4		
BCD					
+12V					

3.7.2 DCD Port

Micro_Link's functionality is configured by connecting a DCD terminal via an 8-way RJ45 socket:

	DCD (TTL) RJ45	**PC 25-way D**	**PC 9-way D**
+5V	1 →		
RXD	2 →	→ 3	→ 2
+12V	3,4 →		
TXD	5 ←	← 2	← 3
RTS	6 ←	← 4	← 7
DCD	7 ←		
GROUND	8 <>	<> 7	<> 5
NC		8-19, 21-25	9
DSR		→ 6	→ 6
CTS		→ 5	→ 8
Shield		<> 1	<> 1
DTR		← 20	← 4

This data is at TTL levels, so a level converter is required between ***Micro_Link*** and a PC. Churchill Controls Ltd supplies a cable assembly with an integral level converter (Part Number 7039-1) with every order that includes a ***Micro_Link***. The adapter also converts the mechanical format from RJ45 to a 9-way D socket suitable for connection to a PC.

3.7.3 Comms_Link Port

Whenever ***Micro_Link*** needs to communicate over the ***Data_Link*** communications network it first tests to see if there is an internal radio fitted. If not, it then tests the ***Comms_Link*** port to see if the EXT COMMS pin is linked to 0V. If not, it assumes it is using an internal leased line modem. The ***Comms_Link*** port thus allows external communication devices to be used in place of the internal options (provided an internal radio is not fitted). It provides the following terminations:

TXD, RXD and RTS are at TTL levels. If the external device uses RS232 signal levels, a level converter must be used, such as the Churchill Controls 7039-1.

	Comms_ Link (TTL) RJ45	**PC 25-way D**	**PC 9-way D**
+5V	1 →		
RXD	2 →	→ 3	→ 2
+12V	3,4 →		
TXD	5 ←	← 2	← 3
RTS	6 ←	← 4	← 7
EXT COMMS	7 ←		
0V	8 <>	<> 7	<> 5
NC		8-19, 21-25	9
DSR		→ 6	→ 6
CTS		→ 5	→ 8
Shield		<> 1	<> 1
DTR		← 20	← 4

3.7.4 I/O_Link Port

I/O_Link provides the capability for I/O expansion. It provides a limited amount of power to expansion modules, but the current rating of the cable only allows power to about three output modules (analogue or digital). If more expansion modules are fitted the cable should be supplemented by a power lead (see chapter 4).

3.8 Alarm Handling

Micro_Link automatically detects a number of possible fault conditions, and stores them as flags which can be sent over the communications network for monitoring remotely. All alarm flags are at a '1' in the normal state and '0' in alarm.

3.8.1 Comms Fail Alarm

3.8.1.1 Base-station

A ***Micro_Link*** base-station it will generate a comms fail alarm if it gets no response to any given data block command for a defined number of consecutive polls. The delay before an alarm is generated depends on the configuration downloaded into it (see **9.1.1.3**) and the type of outstation it is polling. If it is polling a 'normal' (mains-powered) outstation and the **Normal Outstation: Scanning Window** is set to a value greater than zero, it will make as many attempts as it can within the defined time window, allowing a 2-second extended retry for the last attempt. If all attempts fail it will raise a comms fail alarm for that outstation. It treats low power outstations in the same way, except it uses the value set in **Low Power Outstation: Scanning Window**. If it is polling a 'normal' (mains-powered) outstation and the **Normal Outstation: Scanning Window** is set to zero, the number of attempts will be the value set in **Normal Outstation: Retries**, again allowing a 2-second extended retry for the last attempt.

If all attempts fail, the base-station will raise a Comms Fail Alarm for the relevant outstation, and set the RSSI reading for that outstation to zero. It will also raise a common Comms Fail Alarm for the base-station, to indicate that one or more outstations has failed.

If the base-station is fitted with a Comms Expansion Module, see Section 8.15.3 for further details.

3.8.1.2 Outstation

A ***Micro_Link*** outstation will generate a comms alarm if it receives no communication from the base-station for a defined period. It calculates the time-out period from the time between successive scans, plus a contingency time. It cannot derive the time between scans until it has received two polls from the base-station. It will therefore indicate a Comms Fail Alarm until the second poll.

The contingency period is calculated as twice the value downloaded as the **Normal Outstation: Scanning Window**. This will normally be the default value, and will be the same as the value downloaded to the base-station.

For example, if the outstation detects that the base-station is polling it every 25 seconds, and the **Normal Outstation: Scanning Window** is set to 5 seconds, it will raise a Comms Fail Alarm 35 seconds after the last poll.

If the outstation is fitted with a Comms Expansion Module, see Section 8.15.4 for further details.

3.8.2 Complete Comms Fail Alarm

A ***Micro_Link*** base-station also generates a Complete Comms Fail Alarm to signify that it has lost communication with all outstations. This could, for example, be used to indicate a failure of the base-station aerial, or a break in a leased line.

3.8.3 Mains Fail Alarm

Micro_Link assumes it is powered from a 12V lead-acid battery-backed mains power supply. During normal operation the battery is trickle-charged to about 13.6V. If the mains power fails the system will continue working from the battery, but the supply will drop to around 12V. ***Micro_Link*** detects this and generates a Mains Fail Alarm Flag.

The system can be powered from other types of supply, in which case the Mains Fails Alarm flag can be ignored.

3.8.4 Battery Low Alarm

If a battery powers the system, the supply voltage will fall as the battery becomes discharged. If it drops below 10.5V ***Micro_Link*** raises a Battery Low Alarm flag. This indicates an impending system failure, but the time before the system fails will depend on the battery capacity and the total load being drawn.

3.8.5 Hardware Fail

When ***Micro_Link*** is first powered it tests all possible I/O expansion modules to identify the level of equipping. It then continuously scans these modules, passing data between them and the database. If any module subsequently fails to respond the system will flag a hardware failure alarm.

3.8.6 Bus_Link Comms Fail

If the ***Bus_Link*** interface is configured to operate in slave mode (i.e. responding to commands from an external master device) it cannot detect any error conditions, so this alarm is invalid.

If the ***Bus_Link*** interface is configured to operate in master mode (i.e. communicating with one or more slave devices), this alarm will be generated if a slave fails to respond to a pre-set number of consecutive commands.

As well as setting the alarm flag, ***Bus_Link*** also sets a bit corresponding to the slave address in a pseudo input data block numbered 251. This data block is not sent over the communications network. However, an internal data transfer can be used to copy the flag to any other required register to identify which slave device has failed. For example, if a PLC allocated slave address 3 fails and the ***Micro_Link*** station address is 5, the common PLC alarm flag will be set by clearing the ***Bus_Link*** comms fail bit of input data block 5. In addition, bit 3 of pseudo input data block 251 will be set.

3.8.7 Watchdog

Micro_Link also has a watchdog register that continuously toggles state, as long as the internal processor is running. This may be mapped to a digital output, to provide a hardware watchdog signal.

3.9 Micro_Link Specifications

POWER SUPPLY:	10.0…14.0V DC @ 1A max. (Excluding expansion modules)
HARDWARE I/O:	
Digitals:	
Inputs:	4 digital / count inputs compatible with external volt free contacts
	4 digital inputs compatible with external volt free contacts
Energising voltage:	5.0V
Switching current:	50mA wetting current, 250µA continuous
Maximum count rate:	50 pulses/second
Outputs:	8 volt free contacts rated
	125VAC/1A/60VA max. 60VDC/1A/30W max.
	Separate common returns for two groups of 4
Analogues:	
Inputs:	2 Analogue inputs (fully isolated).
Range:	0…20mA into 100Ω or 0…5V into 500KΩ
Accuracy:	± 0.1%
Common mode range:	± 250V
Pseudo inputs:	Battery volts (0…20V ±1%)
	RSSI (0…100%).
Outputs:	2 Analogue outputs (non-isolated).
Range:	0…20mA into 0…500Ω
Accuracy:	± 0.5%
Temperature :	For ***Micro_Link*** and all expansion modules -10 to +50 Degrees Centigrade

4. Input/Output Expansion Modules

Micro_Link can be equipped with a range of expansion modules to increase its hardware I/O capacity. Up to 32 modules can be 'daisy chained' on the ***I/O_Link*** bus. As well as transferring data between ***Micro_Link*** and the expansion modules, this bus also distributes power. It should be noted, however, that it is not capable of carrying sufficient current to power more than typically four expansion modules. To overcome this limitation, each expansion module has a 12V power connector. On large systems the power supply should be fed to every fourth output expansion module as well as to ***Micro_Link***. On small systems the 12V power connectors on expansion modules can be used to take power from the system, within limits.

Each expansion module includes an 8-way DIPswitch, poles 1…5 of which are used to set the modules address. Each module connected to ***Micro_Link*** must be given a unique address, and addresses must be contiguous, starting from 0. On power-up ***Micro_Link*** tests each module, starting from 0, to recognise the type of module fitted. This test ceases when it reaches an address from which it receives no response.

The address on each expansion module is set in binary:

Address	**Switch**				
	1	**2**	**3**	**4**	**5**
0	0	0	0	0	0
1	1	0	0	0	0
2	0	1	0	0	0
3	1	1	0	0	0
4	0	0	1	0	0
5	1	0	1	0	0
6	0	1	1	0	0
7	1	1	1	0	0
8	0	0	0	1	0
9	1	0	0	1	0
10	0	1	0	1	0
11	1	1	0	1	0
12	0	0	1	1	0
13	1	0	1	1	0
14	0	1	1	1	0
15	1	1	1	1	0
16	0	0	0	0	1
17	1	0	0	0	1
18	0	1	0	0	1
19	1	1	0	0	1
20	0	0	1	0	1
21	1	0	1	0	1
22	0	1	1	0	1
23	1	1	1	0	1
24	0	0	0	1	1
25	1	0	0	1	1
26	0	1	0	1	1
27	1	1	0	1	1
28	0	0	1	1	1
29	1	0	1	1	1
30	0	1	1	1	1
31	1	1	1	1	1

where 1 = on, 0 = off.

Micro_Link effectively appends any I/O expansion to its on-board I/O capacity of 16 digital inputs (the first 8 being pseudo), 16 digital outputs (the first 8 being pseudo), 8 analogue inputs (the first 6 being pseudo) and 8 analogue outputs (the first 6 being pseudo). Expansion modules of the same type are suffixed in the sequence of their module addresses. For example, suppose the following expansion modules are fitted:

Module Address	**Type**
0	8-channel analogue input
1	16-channel digital input
2	16-channel digital output
3	8-channel analogue input
4	4-channel analogue output
5	16-channel digital input
6	8-channel analogue input

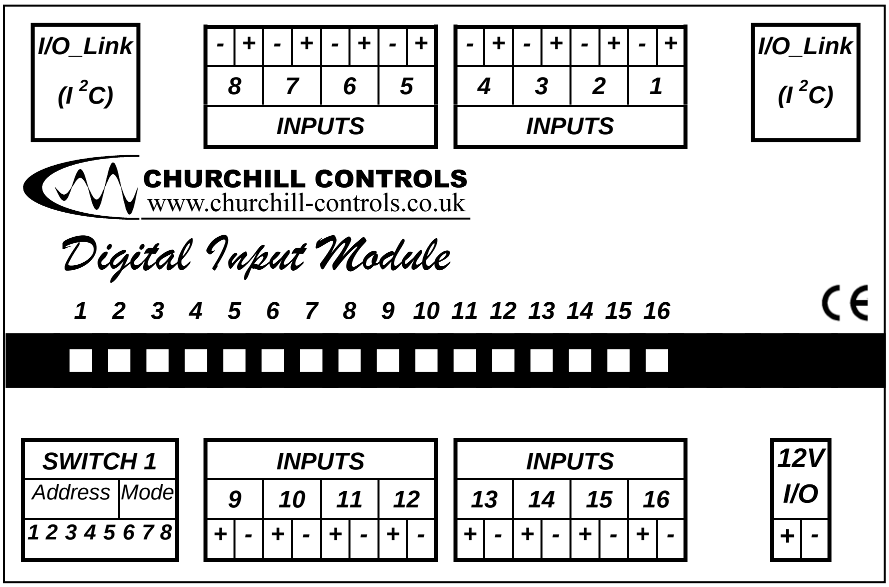
The capacity of this system (including the ***Micro_Link***) will be (8+16+16) = 40 digital inputs, (8+16) = 24 digital outputs, (2+8+8+8) = 26 analogue inputs and (2+4) = 6 analogue outputs. Digital inputs 1…8 will be on ***Micro_Link***, whilst digital inputs 9…24 will be on module address 1 and 25…40 will be on module address 5. Similarly analogue inputs 1 & 2 will be on ***Micro_Link***, whilst analogue inputs 3…10 will be on module address 0, 11…18 will be on module address 3 and 19…26 will be on module address 6.

The mix of modules, and their physical location on the bus, is irrelevant. Normal practice, however, is to locate the module set to address 0 adjacent and to the right of ***Micro_Link*** and succeeding modules in sequence from it, using short jumper links. The ***I/O_Link*** bus can be extended to a total length of 5m if required.

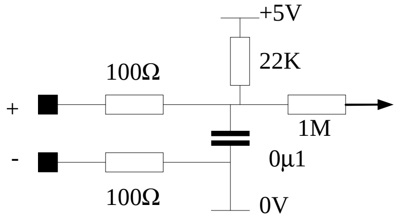
As well as passing data to/from expansion modules, the ***I/O_Link*** bus also distributes power to them. However, its current capacity is limited to 1.0A. Although digital and analogue input modules draw very little power, each digital output module can draw up to 225mA, and each analogue output module can draw up to 200mA. The ***I/O_Link*** bus can therefore only power up to 5 output modules. If a system is equipped with more than this, power distribution must be supplemented by bussing together the 12V I/O terminals on the expansion modules with the 12V terminal on ***Micro_Link***. It is recommended that every 4th output module is linked using 16/0.2mm wire.

The following expansion modules are available:

4.1 7150-1 Digital Input Module



This module has 16 inputs, each with two terminals marked '+' and '-'. The input circuit of each is as follows:



The inputs are designed for use with volt-free contacts, which are supplied a wetting current of 50mA on closing, with a normal sense current of 250µA. However, they can also be used to monitor a DC voltage of up to 24V relative to 0V.

The module usually generates a logic '0' state when the contact is open or the monitored voltage is greater than 3.5V, and a logic '1' state when the contact is closed or the voltage is less than 1.5V. However, the sense can be inverted if required by setting switch 8 of the DIPswitch ON. This can be summarised as follows:

Input State	Logic State	
	Switch 8 OFF	**Switch 8 ON**
Volt-free contact open	0	1
Volt-free contact closed	1	0
Sense voltage > 3.5V	0	1
Sense voltage < 1.5V	1	0

The module draws minimal current from the power supply.

Specifications

CURRENT CONSUMPTION:

12V:	0mA
5V:	10mA max.

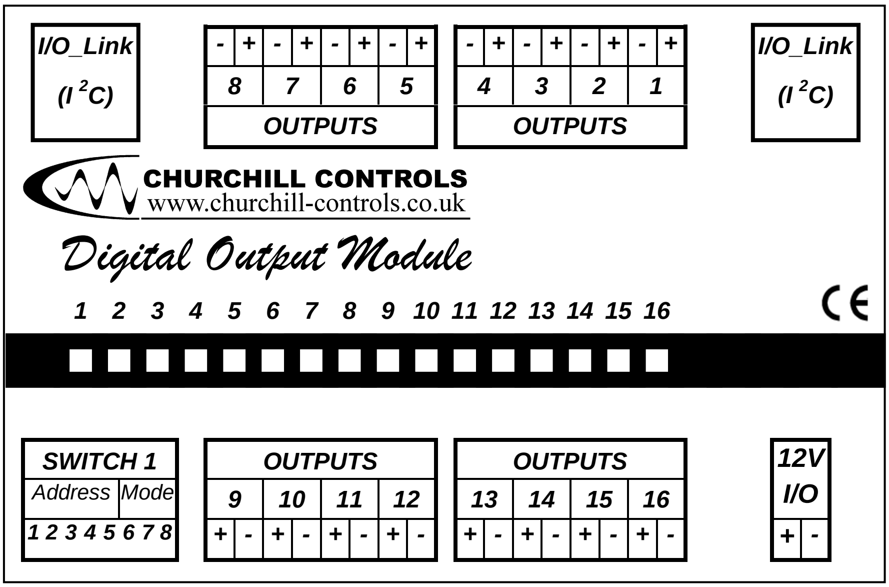
INPUT VOLTAGE:

Logic 0:	min. -24V	max. +1.5V
Logic 1:	min. +3.5V	max. +24V

ENERGISING CURRENT:

50mA wetting current, 250µA continuous

4.2 7150-2 Digital Output Module



This module has 16 fully isolated outputs. Each comprises a volt-free relay contact rated 125VAC/1A/60VA max. 60VDC/1A/30W max. A 120V transient suppressor is fitted across each contact to prevent damage and/or interference when switching inductive loads. Each contact is isolated to 500V min from all other contacts and from 0V.

Although the switch contacts are rated at up to 125V, it is the user's responsibility to ensure that external power sources do not compromise the operator's safety.

If the relays are used to switch inductive loads, such as interposing relays, the load must include transient suppression to prevent excessive voltages during switching. If the load is DC, this is most easily achieved by connecting a reverse-biased diode across the load. If it is AC, a bi-directional suppressor such as transorb or a voltage dependent resistor should be used

The module usually closes the contact for logic '1' state and opens it for logic '0'. However, the sense can be inverted if required by setting switch 8 of the DIPswitch ON. This can be summarised as follows:

Logic State	**Output State**	
	Switch 8 OFF	**Switch 8 ON**
0	OFF	ON
1	ON	OFF

The module draws a maximum of 240mA from the power supply when all outputs are on.

Specifications

CURRENT CONSUMPTION:

12V: 225mA max. (All outputs on)
5V: 10mA max**.**

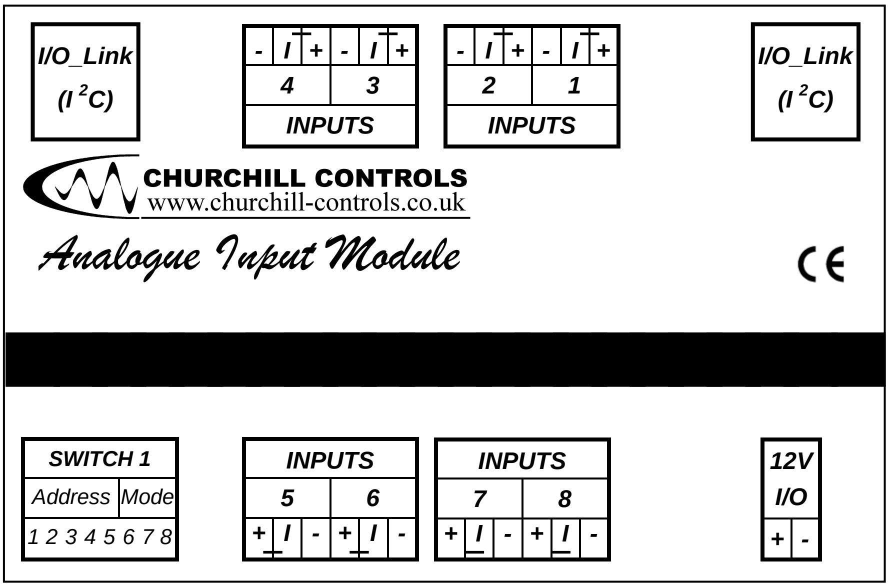
SWITCH RATING:

125VAC/1A/60VA max. 60VDC/1A/30W max.

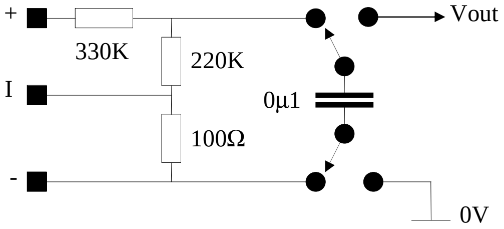
ISOLATION:

500VAC

4.3 7160 Analogue Input Module



This module has 8 inputs, each with three terminals marked '+', 'I' and '-'. The input circuit is as follows:



The input sensitivity is ±5 volts if the 'I' terminal is left open-circuit, or ±20mA if the 'I' terminal is linked to the '+' terminal. All inputs are fully isolated using the 'flying capacitor' technique, and are read into an ADC with a resolution of 12-bits plus sign. Although this gives a numeric range of -4096…+4095, it is calibrated to give a full-scale reading of ±4000. This allows over-range inputs to be indicated, so faulty transducers can be identified.

Analogue outputs on ***Micro_Link*** and on analogue output expansion modules cannot replicate negative values, so convert them to positive values. Therefore if the analogue inputs are inadvertently reversed the resulting output will still be presented correctly.

The calibration of this module can be checked by applying inputs from a voltage or current reference and using DCD Diagnostics (see 9.2) to monitor the resulting register contents.

Specifications

CURRENT CONSUMPTION:

12V:	0mA
5V:	10mA max.

INPUT SENSITIVITY:

±5.000V into 550KΩ or
±20.00mA into 100Ω
(For a digital output of 4000)

RESOLUTION:

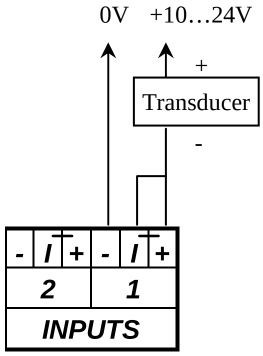
12 bits plus sign (-4096…+4095)

OVERALL ACCURACY:

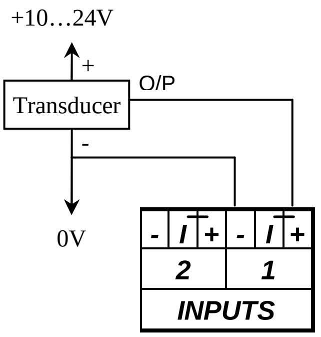
±0.1%

INPUT ISOLATION:

500VAC

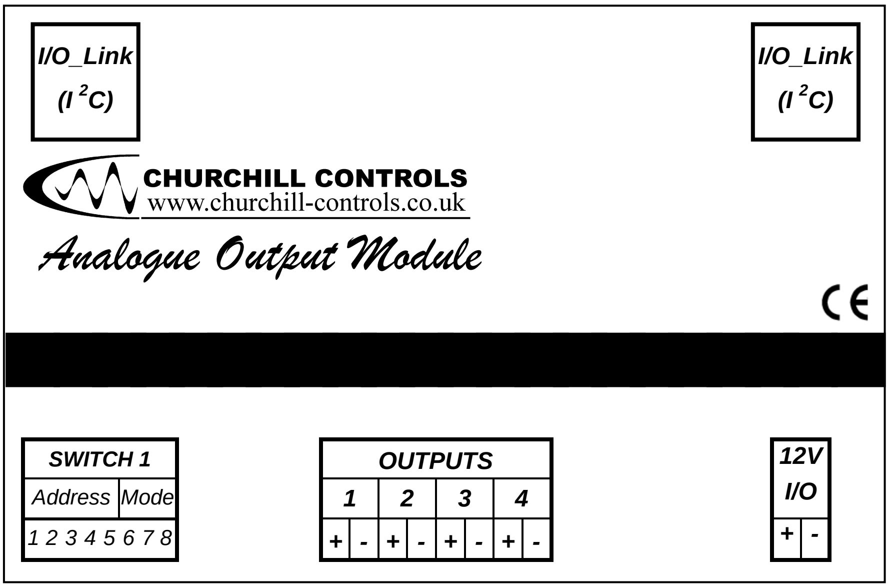


Current Loop Transducer



Voltage Output Transducer

4.4 7170 Analogue Output Module



This module has 4 outputs, each with two terminals marked '+' and '-'. Each output is fully isolated and generates a current of 0…20mA at a voltage of up to 20V (i.e. into a load of up to 1000Ω).

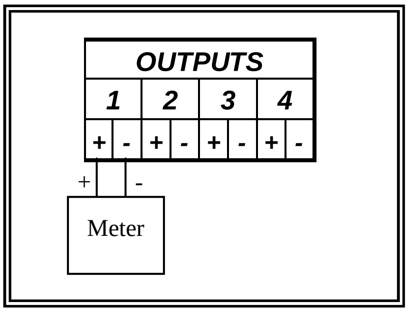
The outputs are derived from digital-to-analogue converters which have 12-bit resolution (0…4095), and are accurate to ±0.1%. They automatically invert negative values.

The outputs are calibrated to give full scale for a digital input of 4000. This allows over-range values to be indicated.

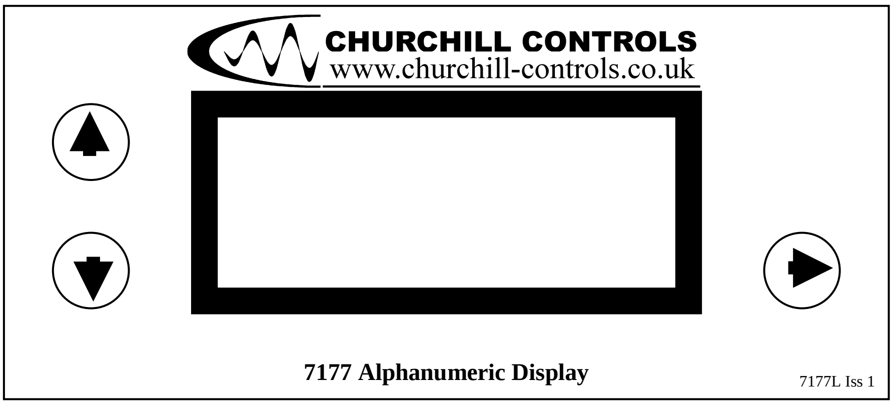
Switch 8 of the DIPswitch can be used to check the calibration of the module and any meters connected to it. If this switch is closed (on) all four outputs generate 20.000mA.

Specifications

CURRENT CONSUMPTION:
12V: 200mA max
(All outputs at full scale)
5V: 10mA max.
OUTPUT CALIBRATION:
0…20.00mA into 1KΩ max.
(For a digital input of 4000)
RESOLUTION:
12-bit (0…4095)
OVERALL ACCURACY:
±0.1%
OUTPUT ISOLATION:
500VAC



4.5 7177 Display Module



The display module is the only expansion module that is not housed in a standard DIN rail enclosure. It is instead fitted in an ABS box that can be hand-held or fitted to a control panel. The module is connected to the host system (***Micro_Link*** or ***Nano_Link***) via the ***I/O_Link*** bus, and its connecting cable can be up to 5m long, allowing the display to be remote from the host system. Up to 64 digital display modules can be daisy-chained on the bus, but only one alphanumeric display module can be used on any given host.

An 8-pole DIPswitch is fitted inside the module, along with four screw terminals, as described below.

4.5.1 7175-1 Alphanumeric Display

The main use of the alphanumeric display is as a diagnostic tool for use with ***Nano_Link***. It is therefore described in the '***Nano_Link*** Technical Manual'

4.5.2 7175-2 Digital Display

This version can only be used with ***Micro_Link***. It has four 7-segment LED displays capable of displaying values in the range 0.001…9999. The bezel on the front of the module can be extended through the panel to give a neat finished appearance. It allows analogue values and counts to be displayed directly without the need for external meters.

Each digit is 0.8" high, so visible from a long distance. The decimal point is user-configurable via the host, and the display automatically blanks leading zeros. The value displayed can be scaled into true engineering units within ***Micro_Link*** (see 8.9).

The LED digits are powered from +12V. Although this can be derived from the ***I/O_Link*** bus, the current capacity of the cable limits it to only one display module. If additional displays are used power must be fed to internal screw terminals (designated 0V and +12V) to supplement the cable.

Poles 1…6 of the DIPswitch define the module address, allowing up to 64 display modules to be accessed. The remaining poles of the switch are not used.

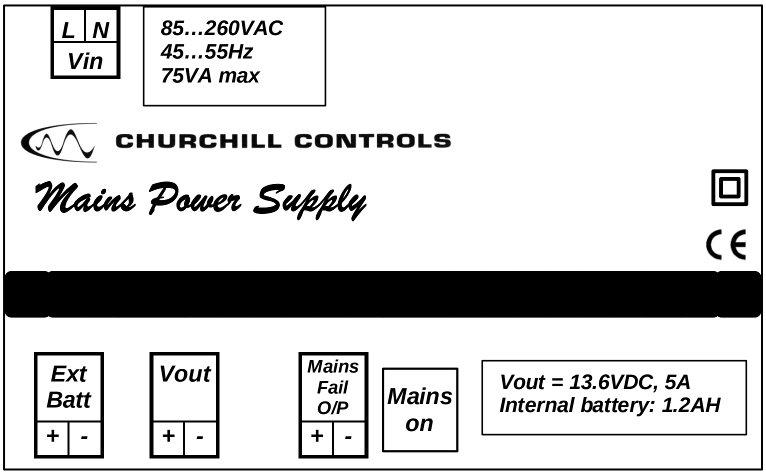
Specifications

SUPPLY:

Digital Display:
5V @ 10mA
12V @ 150mA max

Alphanumeric Display:
Awake: 5V @ 10mA
Asleep: 5V @ 20µA max

4.6 7181 Mains Power Supply



This module can operate from any supply in the range 85…260VAC at 50Hz or 60Hz. It provides a regulated output of 13.6VDC at up to 5A. This supply also trickle-charges an internal sealed lead acid battery. If the mains fails the load is supported by the battery, but the output voltage drops to around 12V.

The time for which the load can be supported by the battery obviously depends on the size of the battery and the total load current. Since ***Micro_Link*** draws typically 200mA, the internal 1.2AH battery will support it for around 6 hours. However, if expansion modules are used, or the supply is required to also power external devices such as transducers, the standby period would be reduced unless the internal battery is supplemented by an external battery. The battery size can be calculated by assuming that ***Micro_Link*** and each output expansion module draws 200mA (input expansion modules draw negligible power). The power supply can support an external battery of up to 25AH via the connector provided.

The module also provides a ‘mains fail’ output that can be fed into any digital input if required. The output is an open collector transistor, which is switched on when mains is present. The ‘+’ terminal accesses the collector via an internal 100Ω protection resistor and the ‘-’ terminal is connected to 0V via an internal 100Ω protection resistor.

All outputs are protected by self-resetting fuses that will reset if the load is disconnected for a few seconds. The mains input is protected by an internal fuse that will only blow if a serious fault develops within the module.

Specifications

MAINS INPUT SUPPLY:
85…260VAC 45…65Hz
100VA max.

OUTPUT VOLTAGE:
13.6VDC ±0.2V when mains present
10V…13VDC during power failure
(Dependent on battery charge state)

OUTPUT CURRENT:
5A max

BATTERY CAPACITY:
1.8AH internal
(may be supplemented by external battery)

5. Installation

5.1 Mechanical

Data_Link 2000 outstations and base-stations are usually supplied in either steel or polycarbonate enclosures that can be attached to a wall using conventional fixings. The enclosures provide a high degree of protection against water ingress, but care is needed to ensure this isn't compromised by the method of installation.

Most enclosures include cable glands, which are supplied with blanking plugs to maintain a seal. The sealing plugs must be removed before feeding cables through the glands, and the glands should be tightened around the cables to maintain the seal. Blanking plugs should be left in any unused glands.

5.2 Aerials

The four types of aerial commonly used on ***Data_Link 2000*** are whips, end-fed dipoles, folded dipoles, co-linear dipoles and Yagis.

Whips and end-fed dipoles are omni-directional (radiate equally in all directions), so their orientation is not important, but they must be mounted vertically. Whip aerials are usually attached directly to ***Nano_Link*** or ***Micro_Link***, or to the top of the enclosure in which they are mounted. ENF450 end-fed dipole aerials fit into the top of a 2" (OD) pole mounted externally.

CDF450 folded dipoles are useful for mounting on the side of a pole when the top is not available for an ENF450. Although they radiate in all directions, the signal is slightly stronger in the direction in which the balun is pointing (i.e. out of the page, towards the reader, in the illustration shown).

Co-linear aerials are similar to end-fed dipoles, but are longer. They are designed to radiate in a horizontal disc pattern, so energy is not wasted by being radiated upwards. They radiate equally in all directions in the horizontal plane, and produce a signal which is typically 6dB higher than a dipole. They can, however, only be used on flat terrain, since they don't radiate upwards or downwards.

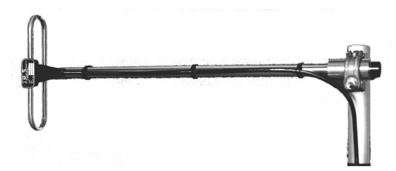
Yagis are similar to end-fed dipoles, but with reflectors which focus the signal in the direction in which they are pointing (i.e. to the left in the illustration shown). This results in signal gain in one direction, at the expense of loss in all other directions. The directivity and hence gain is related to the number of elements. A typical Yagi, the UHF8, has 8 elements and a gain of 10dB.

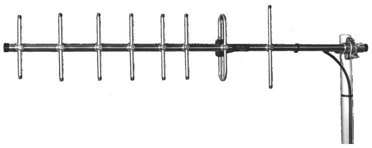
Note that approval regulations limit the maximum effective radiated power (ERP) that can be emitted from a transmitter. If a co-linear or Yagi is used the transmitter output power should be reduced to compensate for the aerial gain. The benefit of aerial gain is that it increases the sensitivity of the receiver, hence increasing the permissible path loss. Since ***Data_Link 2000*** uses two-way radio communication, there is no benefit in using aerials with diferent gains, since this only improves transmission in one direction.

All aerials should be vertically polarised (i.e. the elements should be vertical, not horizontal). Yagis and end-fed dipoles also have a defined top and bottom. They must be installed in the orientation marked by labels attached to them.

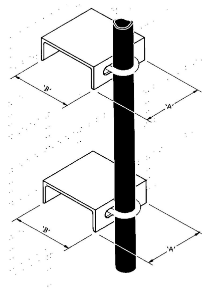
External aerial poles can be fixed to walls using either CS6 or SAB brackets. CS6 brackets space the pole 6" from the wall. SAB brackets allow the poles to be spaced further from the wall to clear soffits and gutters, as well as providing a stronger fixing capable of supporting longer poles.

CDF450 Folded Dipole

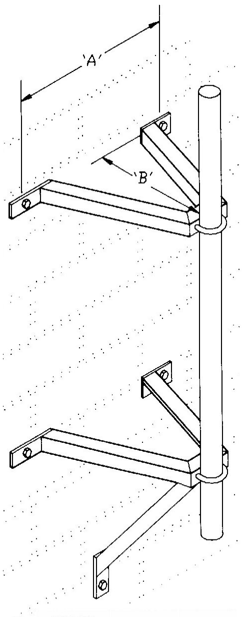




UHF8 Yagi



CS6



STOCK CODE	A	B
SAB 3/12	365mm	305mm
SAB 3/18	460mm	460mm
SAB 3/24	620mm	600mm
SAB 3/36	900mm	900mm

SAB

5.3 Surge Protection

Lightning is caused by static electricity building up in the atmosphere to the level at which flash-over occurs. Discharges between clouds are spectacular, but not a major cause of damage. Discharges to ground, however, can release vast amounts of energy and cause extensive damage. Since these discharges take the path of least resistance, they are more prevalent on elevated, moist ground, and are attracted to high points such as masts. Unfortunately, these are often the locations needed for telemetry systems, which often monitor reservoirs that are by necessity located on high ground. The aerials of radio outstations are particularly vulnerable. ***Zap Gap*** is designed to protect these systems from damage by lightning.

A direct lightning strike discharges so much energy that it will destroy anything in its path. One means of preventing this damage is to dissipate the charge before it achieves the flash-over potential. This requires a path of low electrical resistance between earth and the top of elevated constructions such as masts. Charged particles are attracted to the mast and flow safely to ground, inducing a substantial but controlled current in the conductor. Unfortunately the aerial is usually mounted at the top of the mast, so a significant current can flow through it to the radio equipment and thus to ground. ***Zap Gap*** is a surge arrestor that prevents damage by diverting the current directly to ground, and thence by-passing the radio equipment.

Surges can also be induced in long cables from both nearby lightning strikes and electrical power distribution faults. Cables at exposed sites on high ground (such as water reservoirs) is particularly susceptible. The surges will typically be a few hundred amps at a few hundred volts. Equipment can be protected against these by fitting surge arrestors.

A surge arrestor designed for use on signal cables works by diverting the transient to earth and thus limiting the maximum voltage that can be passed to the equipment. The main component used is a gas discharge tube (GDT), which is normally an open circuit, but will typically ionise at around 150V. In its ionised state it will clamp the line to around 10V for transient currents of up to 10,000A, then return to the open circuit state when the current drops below a threshold of around 50mA. Special zener diodes are used to limit the maximum voltage presented to the equipment to typically 30V ***relative to the surge arrestor's earth terminal***.

Surge arrestors are very effective in protecting equipment against induced transients, ***provided they are correctly installed***. The most important aspect is the provision of a good earth to which the transient can be diverted. Figure 1 below graphically illustrates the potential problem caused by an incorrect installation. Figure 2 illustrates the correct method of installation.

The illustrations correctly indicate that the integrity of the earth connection is irrelevant, provided there is no other path to earth. However, there will inevitably be other paths, via other equipment connected to the outstation, or ultimately by arcing. The earth connection should therefore be as low impedance as possible. Mains earth is not adequate, since it is only designed to take fault current of tens of amps.

Ideally the earth connection should be to a buried copper mat, or at least a substantial earth stake. The connecting cable should be at least 10mm^2, and should be braid or copper strip to minimise its inductance.

Surge arrestors should be mounted close to the equipment being protected, but the input and output wiring should be kept apart to prevent electromagnetic coupling.

The only other means of protecting against lightning is to dissipate the static charge in the atmosphere before it reaches a high enough level to flashover. This is achieved by fitting lightning conductors to any elevated points. Unfortunately, radio aerials form very effective lightning conductors! They are thus prone to receive power surges before lightning occurs. They should thus ideally also be protected by surge arrestors. The above comments regarding earthing are equally applicable.

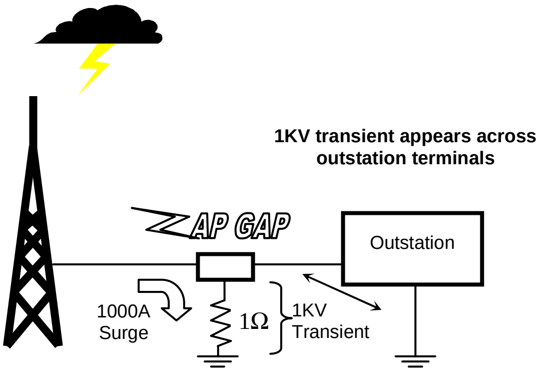


Fig 1 Poor installation

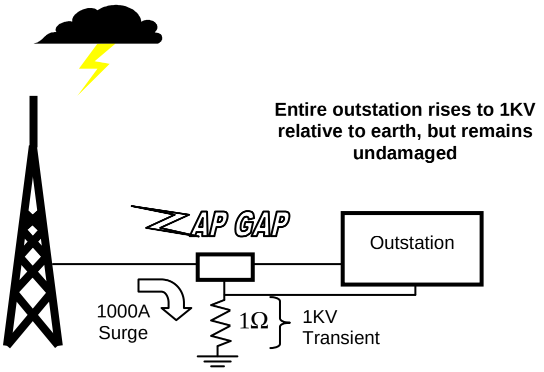

Entire outstation rises to 1KV relative to earth, but remains undamaged
ZAP GAP
Outstation
1000A Surge
1Ω
1KV Transient

Fig 2 Correct installation

6. System Configuration

6.1 Introduction

Every ***Data_Link 2000*** system comprises a base-station and one or more outstations, communicating via radio or leased lines. The base-station is normally ***Micro_Link***, although simple point-to-point systems can use a ***Nano_Link*** base-station, as described in the '***Nano_Link*** Technical Manual'. Each outstation can be either ***Nano_Link*** or ***Micro_Link***, the choice being dictated by the required functionality. The base-station and each ***Micro_Link*** outstation can be fitted with any mix of expansion modules, to a maximum of 32 modules.

The system configuration is user-programmable via the ***Data_Link Configuration and Diagnostic*** (DCD) terminal. This is a software package that runs on any IBM-compatible PC. It allows the user to create a configuration file, save it to disc, download it to ***Micro_Link,*** and to upload it from ***Micro_Link***. It is fully described in Chapter 9, but before a user can begin to configure a system, it is necessary to understand the system configuration, as described in the remainder of this chapter.

6.2 Physical

All data transfer is normally instigated by the base-station, and all data is routed through it. The location of the base-station is at the discretion of the system integrator, but it will usually be at the site at which most data is accessed. A communications path must be available between the base-station and every outstation, but some outstations can be designated as repeaters to relay data to/from more distant outstations to overcome limitations on radio range.

6.3 Database

To understand the addressing requirements of a ***Data_Link 2000*** system, a basic understanding of the database is necessary. The base-station maintains a complete database, whilst outstations and repeaters only maintain the part of the database which is relevant to them. It is divided into an input database and an output database, both of which are further divided into 251 data blocks, numbered 0…250:

INPUT DATABASE

Input data block 0
Input data block 1
…
Input data block *n*
…
Input data block 250

OUTPUT DATABASE

Output data block 0
Output data block 1
…
Output data block *n*
…
Output data block 250

Each data block comprises 32 digital states (numbered 0…31) and eight registers (numbered 0…7). Each digital state holds a value 0 or 1 (which could represent states on/off, open/closed etc). Each register holds a 16-bit value, which can represent a numeric value in the range 0…65,535 or -32,768…+32,768 (e.g. water level, pressure or totalised flow).

Generic Data Blocks:

INPUT DATA BLOCK

Digital Input 0	Input Register 0
Digital Input 1	
Digital Input 2	
Digital Input 3	
Digital Input 4	Input Register 1
Digital Input 5	
Digital Input 6	
Digital Input 7	
Digital Input 8	Input Register 2
Digital Input 9	
Digital Input 10	
Digital Input 11	
Digital Input 12	Input Register 3
Digital Input 13	
Digital Input 14	
Digital Input 15	
Digital Input 16	Input Register 4
Digital Input 17	
Digital Input 18	
Digital Input 19	
Digital Input 20	Input Register 5
Digital Input 21	
Digital Input 22	
Digital Input 23	
Digital Input 24	Input Register 6
Digital Input 25	
Digital Input 26	
Digital Input 27	
Digital Input 28	Input Register 7
Digital Input 29	
Digital Input 30	
Digital Input 31	

OUTPUT DATA BLOCK

Digital Output 0	Output Register 0
Digital Output 1	
Digital Output 2	
Digital Output 3	
Digital Output 4	Output Register 1
Digital Output 5	
Digital Output 6	
Digital Output 7	
Digital Output 8	Output Register 2
Digital Output 9	
Digital Output 10	
Digital Output 11	
Digital Output 12	Output Register 3
Digital Output 13	
Digital Output 14	
Digital Output 15	
Digital Output 16	Output Register 4
Digital Output 17	
Digital Output 18	
Digital Output 19	
Digital Output 20	Output Register 5
Digital Output 21	
Digital Output 22	
Digital Output 23	
Digital Output 24	Output Register 6
Digital Output 25	
Digital Output 26	
Digital Output 27	
Digital Output 28	Output Register 7
Digital Output 29	
Digital Output 30	
Digital Output 31	

Every station on a ***Data_Link 2000*** network is allocated at least one input data block and one output block. Each station must be given a unique station address, and this defines the **Root Data Block** used by the station. Station address 0 is the base-station and all other addresses are outstations or repeaters.

A ***Nano_Link*** outstation only occupies the root data block, even if it is equipped with its full compliment of expansion modules. The function of the digitals and registers used by it are as follows:

Nano_Link Root Data Block Usage

	INPUT DATA BLOCK			**OUTPUT DATA BLOCK**	
	Digital	**Register**		**Digital**	**Register**
0	Comms Fail alarm	Totalised count for digital input 1	0	-	-
1	Battery Low alarm			-	
2	Spare (read as 1)			-	
3	Spare (read as 1)			-	
4	Spare (read as 1)	Totalised count for digital input 2	1	-	-
5	Mains Fail			-	
6	Spare (read as 1)			-	
7	Spare (read as 1)			-	
8	Digital Input 1	Totalised count for digital input 3	2	Digital Output 1[1]	-
9	Digital Input 2			Digital Output 2[1]	
10	Digital Input 3			Digital Output 3[1]	
11	Digital Input 4			Digital Output 4[1]	
12	Comms Fail alarm	Totalised count for digital input 4	3	-	-
13	Battery Low alarm			-	
14	Mains Fail			-	
15	Spare (read as 1)			-	
16	Digital Input 9[2]	Battery Volts	4	Digital Output 9[3]	-
17	Digital Input 10[2]			Digital Output 10[3]	
18	Digital Input 11[2]			Digital Output 11[3]	
19	Digital Input 12[2]			Digital Output 12[3]	
20	Digital Input 13[2]	Radio Receiver Signal Strength (RSSI)	5	Digital Output 13[3]	-
21	Digital Input 14[2]			Digital Output 14[3]	
22	Digital Input 15[2]			Digital Output 15[3]	
23	Digital Input 16[2]			Digital Output 16[3]	
24	Digital Input 17[2]	Analogue Input 1	6	Digital Output 17[3]	Analogue Output 1[1]
25	Digital Input 18[2]			Digital Output 18[3]	
26	Digital Input 19[2]			Digital Output 19[3]	
27	Digital Input 20[2]			Digital Output 20[3]	
28	Digital Input 21[2]	Analogue Input 2	7	Digital Output 21[3]	Analogue Output 2[1]
29	Digital Input 22[2]			Digital Output 22[3]	
30	Digital Input 23[2]			Digital Output 23[3]	
31	Digital Input 24[2]			Digital Output 24[3]	

All alarm bits are ‘1’ in the normal state and ‘0’ when in alarm

NOTES

[1] Analogue and digital outputs are not available on units designated ‘Input Only’

[2] Only available if a digital input expansion module is fitted, otherwise read ‘0’

3 Only available if a digital output expansion module is fitted, otherwise ignored

A ***Micro_Link*** outstation or base-station equipped with up to one digital input expansion module and/or one digital output expansion module also occupies only the root data block:

Micro_Link Root Data Block Usage

	INPUT DATA BLOCK			OUTPUT DATA BLOCK	
	Digital	**Register**		**Digital**	**Register**
0	Comms Fail alarm	Totalised count for digital input 1	0	-	Calculated flow rate digital input 1
1	Battery Low alarm			-	
2	Hardware Fail alarm			-	
3	***Bus_Link*** fail alarm			-	
4	Complete Comms Fail	Totalised count for digital input 2	1	-	Calculated flow rate digital input 2
5	Mains Fail			-	
6	Expansion Comms Alarm / Watchdog			-	
7	Dual Comms Alarm			-	
8	Digital Input 1	Totalised count for digital input 3	2	Digital Output 1	Calculated flow rate digital input 3
9	Digital Input 2			Digital Output 2	
10	Digital Input 3			Digital Output 3	
11	Digital Input 4			Digital Output 4	
12	Digital Input 5	Totalised count for digital input 4	3	Digital Output 5	Calculated flow rate digital input 4
13	Digital Input 6			Digital Output 6	
14	Digital Input 7			Digital Output 7	
15	Digital Input 8			Digital Output 8	
16	Digital Input 9[1]	Battery Volts	4	Digital Output 9[2]	-
17	Digital Input 10[1]			Digital Output 10[2]	
18	Digital Input 11[1]			Digital Output 11[2]	
19	Digital Input 12[1]			Digital Output 12[2]	
20	Digital Input 13[1]	Radio Receiver Signal Strength (RSSI)	5	Digital Output 13[2]	-
21	Digital Input 14[1]			Digital Output 14[2]	
22	Digital Input 15[1]			Digital Output 15[2]	
23	Digital Input 16[1]			Digital Output 16[2]	
24	Digital Input 17[1]	Analogue Input 1	6	Digital Output 17[2]	Analogue Output 1
25	Digital Input 18[1]			Digital Output 18[2]	
26	Digital Input 19[1]			Digital Output 19[2]	
27	Digital Input 20[1]			Digital Output 20[2]	
28	Digital Input 21[1]	Analogue Input 2	7	Digital Output 21[2]	Analogue Output 2
29	Digital Input 22[1]			Digital Output 22[2]	
30	Digital Input 23[1]			Digital Output 23[2]	
31	Digital Input 24[1]			Digital Output 24[2]	

All alarm bits are ‘1’ in the normal state and ‘0’ when in alarm

NOTES

[1] Only available if a digital input expansion module is fitted, otherwise read ‘0’

[2] Only available if a digital output expansion module is fitted, otherwise ignored

If more than one digital input expansion module is fitted it will ‘spill over’ into the next input data block, as will any analogue input expansion modules. Similarly, if more than one digital output expansion module is fitted it will ‘spill over’ into the next output data block, as will any analogue output expansion modules. A ***Micro_Link*** base-station or outstation could therefore occupy up to 33 consecutive data blocks in the worst case (i.e. if it is fitted with a full compliment of 32 8-channel analogue input modules).

Note that, if the *Micro_Link* senses that it is fited with a Comms Expansion Module, the Input Data Block Digital 6 is automatically assigned for ‘Expansion Comms Alarm’ and the Watchdog Alarm is not available.

If a Comms Expansion Module is not fitted, the Expansion Comms Alarm is not relevant, and the input is assigned to the Watchdog.

6.4 I/O Expansion

Micro_Link can be fitted with up to 32 expansion modules (as described in Chapter 4) by 'daisy-chaining' them using the ***I/O_Link*** expansion bus. Each module must be set to a unique address in the range 0…31, using the DIPswitches on each module. The addresses must be contiguous, starting from 0. ***Micro_Link*** automatically detects the expansion modules when power is first applied, then sorts them by type, and concatenates them to its built-in I/O, allocating them I/O addresses as appropriate. If the I/O exceeds the capacity of the Root data block, it will 'spill over' into the next data block(s).

The physical location and sequence of each module along the ***I/O_Link*** expansion bus is unimportant.

The following example shows how ***Micro_Link*** will allocate I/O when expansion modules are fitted to a base-station (i.e. root data block = 0):

Module Type	**Address**	**I/O address allocated**	**I/O register**
16-channel Digital Input	0	Digital input 9…24	Input Data Block 0, Digital 16…31
8-channel Analogue Input	1	Analogue input 3…10	Input Data Block 1, Register 0…7
8-channel Analogue Input	2	Analogue input 11…18	Input Data Block 2, Register 0…7
16-channel Digital Output	3	Digital output 9…24	Output Data Block 0, Digital 16…31
4-channel Analogue Output	4	Analogue output 3…6	Output Data Block 1, Register 0…3
4-channel Analogue Output	5	Analogue output 7…10	Output Data Block 1, Register 4…7
Digital Display Module	6	Analogue output 11	Output Data Block 2, Register 0

In this example the modules have been allocated logical addresses, starting with digital inputs and progressing through analogue inputs, digital outputs and analogue outputs. ***Micro_Link*** automatically appends onto the end of its own I/O contiguous addresses for I/O on expansion modules, in ascending order of the addresses set on their DIPswitches.

The physical location of I/O modules on the expansion bus is not important, neither is the sequence of I/O modules. For example, suppose this system is expanded at a later date by adding expansion modules. To overcome the need for re-routing cabling to expansion modules, the modules can be appended in any order:

Module Type	**Address**	**I/O address allocated**	**I/O register**
16-channel Digital Input	0	Digital input 9…24	Input Data Block 0, Digital 16…31
8-channel Analogue Input	1	Analogue input 3…10	Input Data Block 1, Register 0…7
8-channel Analogue Input	2	Analogue input 11…18	Input Data Block 2, Register 0…7
16-channel Digital Output	3	Digital output 9…24	Output Data Block 0, Digital 16…31
4-channel Analogue Output	4	Analogue output 3…6	Output Data Block 1, Register 0…3
4-channel Analogue Output	5	Analogue output 7…10	Output Data Block 1, Register 4…7
Digital Display Module	6	Analogue output 11	Output Data Block 2, Register 0
16-channel Digital Input	7	Digital input 25…40	Input Data Block 1, Digital 0…15
4-channel Analogue Output	8	Analogue output 12…15	Output Data Block 2, Register 1…4
8-channel Analogue Input	9	Analogue input 19…26	Input Data Block 3, Register 0…7

Micro_Link automatically appends the new digital inputs to the end of the previous ones, even though the module addresses are not contiguous. The new analogue outputs have been appended after the last analogue output register used, which happened to have been allocated to a digital display module.

6.5 Station Addresses

Every ***Data_Link 2000*** system comprises one base-station and one or more outstations. The base-station is by default set to station address 0. Each outstation has to be set to a unique station address in the range 1…250, but they don't need to be contiguous, or to start from 1.

As stated above, the station address defines the root data block. A ***Micro_Link*** station with expansion modules may occupy more than one data block, as illustrated in the previous example. To ensure that there is no conflict between data blocks used by outstations and those used by base-station expansion modules, it is advisable to set the first outstation to address 10, to allow for any future expansion of the base-station. Similarly, if there is any possibility of future expansion of ***Micro_Link*** outstations, they should not be given consecutive addresses. A good plan is to uses addresses 10, 20, 30… allowing up to 24 large outstations to be used. However, since ***Nano_Link*** outstations cannot be expanded, they can be given consecutive addresses.

6.6 Data Routing

Once it has identified the internal and expansion I/O with which it is equipped, a ***Micro_Link*** station can pass data to/from its own I/O using the addresses it has allocated for them. However, the base-station also need to know what data to pass to/from outstations. Until this is configured it does not know how many outstations exist, what their addresses are, or how big they are.

6.6.1 Basic system

Suppose three stations are used on a system, and set to addresses 0, 10 and 20. Station address 0 is by definition the base-station, and maintains a full database. The others are outstations. All stations automatically map their hardware I/O to their Root Data Blocks:

Base-station (Address 0)

	Input Block	Output Block
Hardware Inputs ⇒	0	
Hardware Outputs ⇐		0
	1	
		1
	…	
		…
	10	
		10
	11	
		11
	…	
		…
	20	
		20
	21	
		21
	…	
		…
	250	
		250

Outstation Address 10

Input Block	Output Block	
10		⇐ Hardware Inputs
	10	⇒ Hardware Outputs

Outstation Address 20

Input Block	Output Block	
20		⇐ Hardware Inputs
	20	⇒ Hardware Outputs

It should be apparent that at this stage no meaningful data is being sent around the network, since hardware inputs are not mapped to hardware outputs. The base-station doesn't know how many outstations are installed, so it doesn't communicate with any.

This is solved by configuring the base-station to carry out internal data transfers:

6.6.2 Internal Data Transfers

Internal data transfers can be easily configured in a ***Micro_Link*** base-station by using the DCD Configuration terminal, as described in Section 9.1.2. They allow the user to copy digitals and registers from any given location to any other location(s). Obviously care must be taken in configuring internal data transfers to ensure that no digital or register is written data from more than one source, or unpredictable results may occur.

For example, the following entries could be made in the data routing table of the base-station:

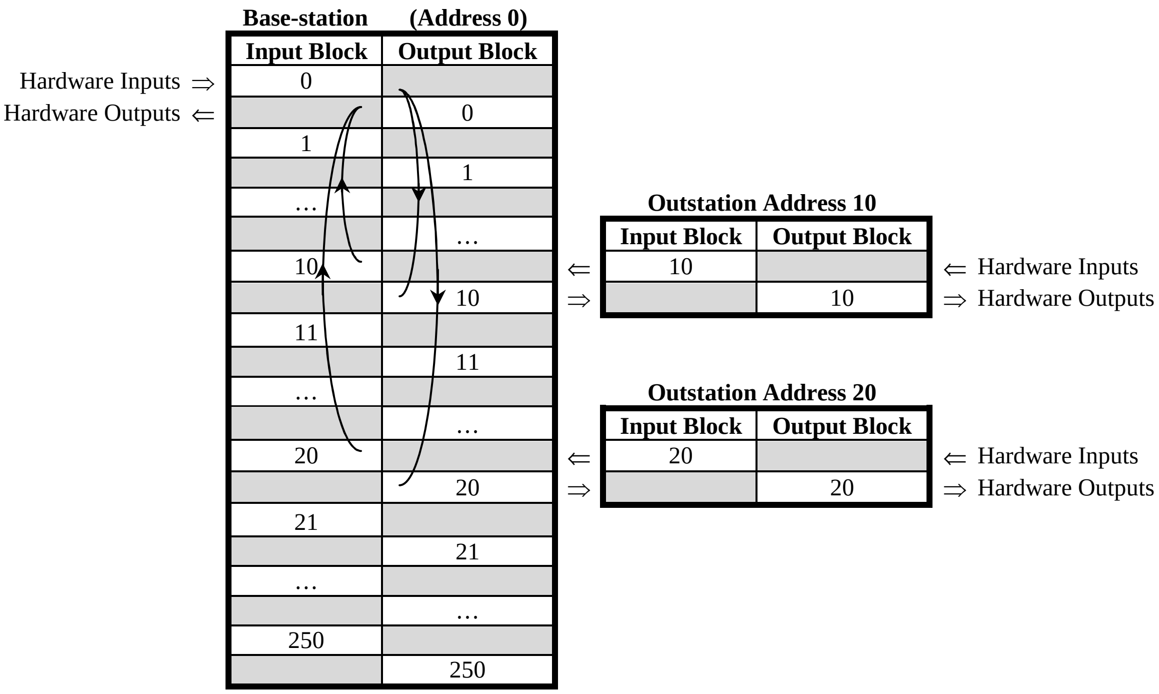
Outstation Address 10 Analogue Input 1 - 2	→	Local Analogue Output 1 - 2
Outstation Address 20 Analogue Input 1 - 2	→	Local Analogue Output 3 - 4
Outstation Address 10 Comms Fail Alarm	→	Local Digital Output 1
Outstation Address 20 Comms Fail Alarm	→	Local Digital Output 2
Outstation Address 10 Battery Low Alarm	→	Local Digital Output 3
Outstation Address 20 Battery Low Alarm	→	Local Digital Output 4

When the base-station recognises an entry in the data routing table which refers to a data block which is not allocated to its internal I/O or to expansion I/O, it assumes it must get the data from an outstation. It thus adds the outstation to its poll sequence.

If the following lines were added:

Local Digital Input 1 - 4 → Outstation Address 10 Digital Output 1 - 4
Local Digital Input 5 - 8 → Outstation Address 20 Digital Output 1 - 4

Then the base-station will also send a data block to each outstation. This can be visualised as follows:



The poll sequence for the above system would be:

Request Input Data Block 10
Send Output Data Block 10
Request Input Data Block 20
Send Output Data Block 20

6.6.3 Bus_Link Data Transfers

Data can also be passed to and from the database using serial communications through the ***Bus_Link*** port. This is explained in detail in Chapter 7. If an external device accesses any register which is not allocated to local I/O, ***Micro_Link*** will assume it is to be passed to/from an outstation, and incorporate that outstation in its poll sequence.

6.7 Polling

The smallest system configuration comprises two ***Nano_Link***'s configured as a point-to-point link. One is defined as a base-station (station address 0) and the other as an outstation. This configuration is described in the '***Nano_Link*** Technical Manual'.

All other systems are built around a ***Micro_Link*** base-station, which is configured via the DCD Terminal.

A ***Micro_Link*** base-station knows which data blocks are being used because they are either being accessed by internal data transfers stored in the DCD Configuration file, or they are being accessed via ***Bus_Link.***

After a power interruption (and after downloading a configuration), a ***Micro_Link*** base-station first scans the configuration table to read any exception data included within it (see 8.1). It then polls all outstations, writing data to them and reading data from them, so that all I/O is initialised without undue delay. In the process of doing this, it sends to each outstation any relevant exception configuration data. The responses from the outstations identify their type (e.g. battery-powered ***Nano_Link***).

When it has finished this initial scan, the base-station starts polling the outstations at the rate defined in the global configuration downloaded into it. Two poll rates are overlaid:

All outstations except battery-powered ***Nano_Link***'s are scanned at the rate defined in DCD Configuration by **Normal Outstation Scanning Window** (see 9.1.1.3). The base-station will attempt to write and/or read data to/from an outstation at the beginning of a window. It will then wait until the end of the window before communicating with the next outstation. When it has scanned all outstations it will repeat the cycle. The overall scan time is thus (*n * Normal Outstation Scanning Window*), where *n* is the number of data blocks used. Note that if the **Normal Outstation Scanning Window** is set to 0 the base-station will scan outstations continuously. This scan will, however, be interrupted by Low Power outstation scanning:

Battery-powered ***Nano_Link*** outstations are scanned at the rate defined by **Low Power Outstation Scanning Window**. The base-station will attempt to write and/or read data to/from an outstation at the beginning of a window. It will then wait until the end of the window before communicating with the next outstation. When it has scanned all outstations it will wait until the **Low Power Outstation Scan Rate** has lapsed before repeating the cycle. The overall scan rate is thus fixed regardless of the number of outstations, provided that **Low Power Outstation Scan Rate** is greater than (*n * Low Power Outstation Scanning Window*). If the scan rate is set to a time less than (*n * Low Power Outstation Scanning Window*) the base-station will automatically reduce the scanning window to suit.

If none of the outstations are configured as a battery-powered ***Nano_Link*** then the Low Power Scanning parameters are irrelevant. Similarly, if all of the outstations are configured as a battery-powered ***Nano_Link*** then the Normal Outstation Scanning parameters are irrelevant.

If exception reporting is configured (see 8.1) the base-station checks its inputs for exceptions during the pauses between outstation polls. If any configured input change occurs at the base-station it will immediately send an exception report to the relevant outstation. Similarly, each outstation monitors its inputs for exception changes and, if detected, sends an exception report to the base-station. The recipient of an exception report will send back an acknowledgement. If an acknowledgement is not received an outstation will repeatedly send the exception report at random time intervals (in the range 2…10 seconds). This ensures success even if multiple outstations initiate exception reports simultaneously.

Exception reporting can only be used with battery-powered ***Nano_Link*** outstations and/or with outstations that are not scanned continuously (i.e. **Normal Outstation Scanning Window** cannot be 0).

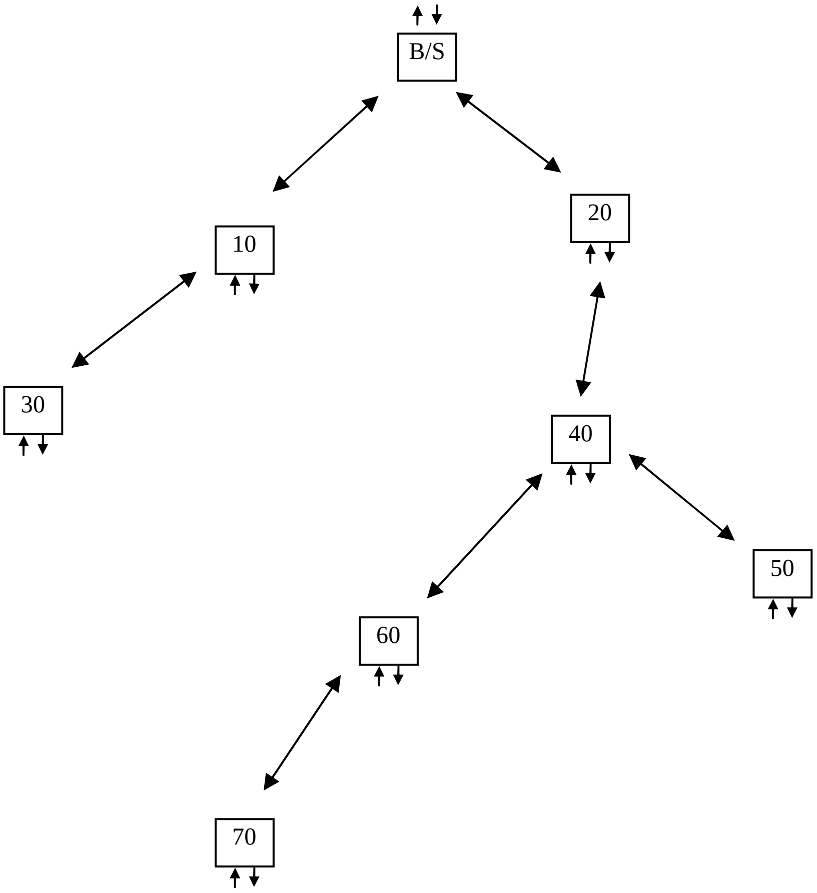
6.8 Network Routing

A ***Data_Link 2000*** system could range from a simple small point-to-point link up to a large polling system with multiple outstations, repeaters and interfaces to PLC's, SCADA systems and/or regional telemetry schemes. The same hardware and system configuration addresses all needs.

A fundamental consideration of any system using de-regulated radio is the limitation on radio transmit power, which is an inherent requirement to restrict interference with other users. MPT1329 is the UK standard for telemetry applications, which allocates 32 channels in the 458.5MHz band, with the transmit power limited to 500mW. (Other countries may use a different band, but most still allow up to 32 channels within the band). With modern technology the corresponding receivers are sensitive enough to achieve a range of up to 25Km (15 miles) with good elevated aerials and no obstructions. The range will be reduced if smaller aerials are used near ground level, if there are obstacles such as large buildings, or if there are geographical features such as hills. In a typical urban environment with a good base-station aerial and outstations in roadside cabinets the range could typically be reduced to 5Km (3 miles). Communication with outstations below ground, within manholes, is possible, but the range is reduced to typically 500m. Extending the radio range beyond this requires the use of repeaters.

Note : When using battery-powered repeaters, the address of the repeater **must** be lower than that of the destination outstation.
e.g. Basestation = Address 0, Repeater = Address 10, Outstation = Address 20.

A hypothetical system might be as follows:



Every outstation inherently also functions as a repeater. The illustration shows 4 levels of repeaters to reach outstation 70. In fact, up to 8 levels of repeaters can be used to reach distant outstations.

The DCD Configuration software only expects the user to identify the last repeater used to access each outstation. If no entry is made for a specific outstation the base-station assumes it has direct communications. If the last repeater is configured to be accessed via another repeater, the software will automatically derive the chain of repeaters needed. For example, for the above system the user may first identify that to access outstation 40 the last repeater is 20. The routing table for this network will thus be:

Address	**Route**
40	20

If another entry is made to access outstation 60 via 40, the table will become:

Address	**Route**
40	20
60	40 20

The final table for the above system will be:

Data Block	**Last Repeater**
30	10
40	20
50	40 20
60	40 20
70	60 40 20

If an outstation occupies more than one data block, an entry is required for each data block used. For example, if outstation 40 includes expansion modules that occupy two additional data blocks, then the network routing table will need to be modified thus:

Data Block	**Last Repeater**
30	10
40	20
41	20
42	20
50	40 20
60	40 20
70	60 40 20

7. Bus_Link

One of the key features of ***Micro_Link*** is its ability to communicate through a serial data interface with external devices, using one of a variety of standard protocols. Churchill Controls proprietary name for this interface is ***Bus_Link***.

Industrial automation has developed around Programmable Logic Controllers (PLC's) which are available from a wide range of manufacturers. Originally these were stand-alone devices, but the need soon arose for passing data between them. Supervisory, Control and Data Acquisition (SCADA) systems were also developed to provide a man-machine interface with features such as alarm annunciation, data archiving and closed loop control.

Unfortunately, each PLC manufacturer has evolved his own communications protocol. Some have a hierarchy of protocols optimised for different applications, some of which use multiple master protocols operating on high-speed LAN networks. However, all have low-speed variants operating in master-slave mode at either RS232 or RS485 signal levels.

Because of the plethora of protocols, all general-purpose devices that need to communicate with PLC's (such as SCADA systems) have to be designed with a variety of software drivers, which can be selected to match the required standard. ***Bus_Link*** has been designed in the same way.

Bus_Link can be configured to use either a standard RS232 interface or a multi-drop RS485 interface. It can be configured to emulate either a slave device or a master device, and to communicate using a variety of different protocols. All configuration is carried out using the DCD Terminal, which is described in Chapter 7.

All PLC protocols are based around registers. In all master-slave protocols the communication is instigated by the master device, which can copy data between its own registers and registers in the slave device(s). Some registers contain 16-bit values, while others represent the state of individual digitals. ***Micro_Link*** maintains a database that closely resembles that of any PLC.

7.1.1 Absolute and Relative Addresses within Data Blocks

Any given register or digital can be identified by an absolute address or a relative address from a reference Data Block. For example, each of the following refers to the same register:

Input Register 172

Input Block 0, Register 172

Outstation Address 21, Battery Volts (21 x 8 + 4)

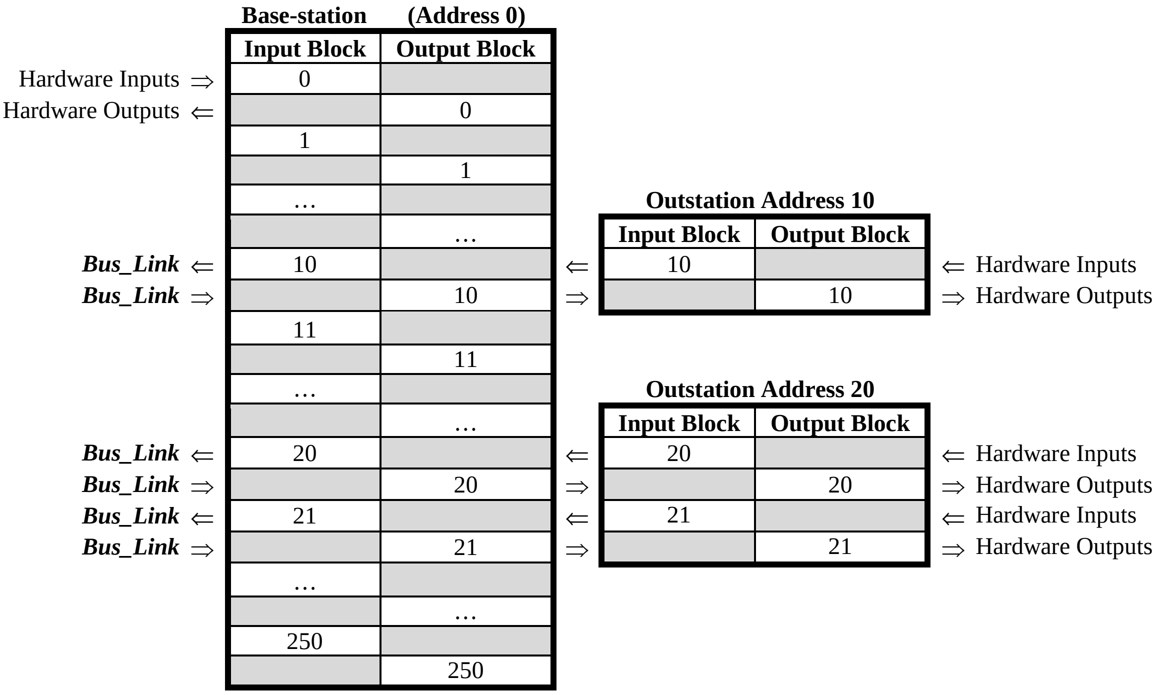
Outstation Address 20, Analogue Input 6 (20 x 8 + 12)

Absolute addressing (format (a)) is the format used by PLC's and SCADA systems communicating via serial protocols such as Modbus. All the other formats identify the register relative to a given start point. Relative addressing is convenient for configuring ***Data_Link 2000***, since the Root Data Block can be used to identify the relevant outstation address, with the register number identifying the relevant digital or register within the outstation.

Note that the ***Data_Link 2000*** numbering convention always starts from 0. Some PLC protocols (e.g. Modbus) start counting from 1, so the register in the above example would be regarded by some Modbus systems as register 173.

7.2 Configuring Bus_Link

Bus_Link could typically be used in a ***Micro_Link*** base-station to allow data to be passed to/from a SCADA system. Since most SCADA systems emulate a PLC master, ***Bus_Link*** would be configured in slave mode:



Base-station (Address 0)

	Input Block	Output Block	
Hardware Inputs ⇒	0		
Hardware Outputs ⇐		0	
	1		
		1	
	…		
		…	
Bus_Link ⇐	10		⇐
Bus_Link ⇒		10	⇒
	11		
		11	
	…		
		…	
Bus_Link ⇐	20		⇐
Bus_Link ⇒		20	⇒
Bus_Link ⇐	21		⇐
Bus_Link ⇒		21	⇒
	…		
		…	
	250		
		250	

Outstation Address 10

Input Block	Output Block	
10		⇐ Hardware Inputs
	10	⇒ Hardware Outputs

Outstation Address 20

Input Block	Output Block	
20		⇐ Hardware Inputs
	20	⇒ Hardware Outputs
21		⇐ Hardware Inputs
	21	⇒ Hardware Outputs

Since the SCADA system is a PLC master it will be configured to identify which data it requires to access, using absolute addressing. For example, to read outstation 20 battery volts the SCADA system should read Output Register 164 at the base-station (20 x 8 + 4).

Any register or digital can be copied to any destination(s) via ***Bus_Link*** and/or internal transfers. However, any given output register or digital must only be fed data from one source.

Bus_Link is also available on ***Micro_Link*** outstations. Its most likely application here is to exchange data with PLC's or intelligent instruments. Since these usually emulate PLC slave protocols the outstation usually needs to be configured as a master. The user must decide which registers and digitals to use in the outstation. Small amounts of data can use spare capacity in the outstation's Root Data Block or expansion data blocks. Larger amounts of data can be directed to/from any other data block (except those used by other outstations!). It is most logical to append ***Bus_Link*** data to the end of the hardware I/O, although though should be given to the possibility of expanding the hardware I/O at some time in the future:

	Base-station	(Address 0)				
	Input Block	**Output Block**				
Hardware Inputs ⇒	0					
Hardware Outputs ⇐		0				
	1					
		1				
	…			**Outstation Address 10**		
		…		**Input Block**	**Output Block**	
Bus_Link ⇐	10		⇐	10		⇐ Hardware Inputs
Bus_Link ⇒		10	⇒		10	⇒ Hardware Outputs
	11					
		11				
	…			**Outstation Address 20**		
		…		**Input Block**	**Output Block**	
Bus_Link ⇐	20		⇐	20		⇐ Hardware Inputs
Bus_Link ⇒		20	⇒		20	⇒ Hardware Outputs
Bus_Link ⇐	21		⇐	21		⇐ Hardware Inputs
Bus_Link ⇒		21	⇒		21	⇒ Hardware Outputs
Bus_Link ⇐	22		⇐	22		⇐ ***Bus_Link***
Bus_Link ⇒		22	⇒		22	⇒ ***Bus_Link***
	…					
		…				
	250					
		250				

In the above example data to/from ***Bus_Link*** on outstation 20 has arbitrarily been routed to/from data block 22. The base-station treats the data exactly as it would data from expansion modules at the outstation. By this means, data is transferred seamlessly between hardware I/O and PLC's, SCADA systems and third party equipment

The ***Bus_Link*** interface at the outstation has to write data to input registers and read from output registers. If the outstation is configured to emulate a slave device the external master system need to follow this convention. Some PLC protocols don't support commands to comply with this requirement. Therefore ***Micro_Link*** translates any command received by an outstation in ***Bus_Link*** slave mode as accessing the correct register type, regardless of the command detail.

7.3 Modbus Protocol

The Modbus protocol was originally defined by AEG Modicon (now part of Schneider Electric Telemecanique) for communicating with their PLC's. However, it has now become the de facto standard and has been adopted by many other PLC manufacturers. In Modicon terminology digital outputs are Coils, digital inputs are Status's, output registers are Holding Registers and input registers are Input Registers. To further distinguish one from another, they are given unique addresses:

00001…08000	Coils
10001…18000	Status's
30001…32000	Input Registers
40001…42000	Holding Registers

However, Modbus protocol defines within each command the type of data required, so the x000 prefix is not required. Furthermore although the database registers start counting from 1, Modbus protocol starts from 0.

Bus_Link interface makes allowance for the x000, prefix, so for example the first status input is designated 1, not 10001. However, some PLC implementations of Modbus also allow for the 0/1 discrepancy, so they may refer to the first digital as digital 0 rather than digital 1. Since there is no universal conformity the user has may have to adjust all addresses by 1 to allow for the translation within the PLC's Modbus driver.

7.3.1 Modbus Master Mode

If ***Bus_Link*** is configured to operate in Modbus master mode, it will send the following commands to a slave device:

Bus_Link **action**	**Function Code**	**Modbus meaning**
Read slave digital outputs	01	Read Coil Status
Read slave digital inputs	02	Read Input Status
Read slave output registers	03	Read Holding Registers
Read slave input registers	04	Read Input Registers
Write to slave digital inputs	15	Force Multiple Coils
Write to slave output registers	16	Pre-set Multiple Registers

For example, the following line within the data routing table of a base-station will copy the first two analogues from outstation 10 to holding registers 10…11 (i.e. registers 40010…40011) of a PLC set to address 0, using Function Code 16:

Outstation 10 Analogue Input 1-2 -> Bus_Link Add 0 *Analogue o/p register 10-11

This line will copy digital inputs 20…27 from the PLC to the first 8 digital outputs at outstation 20, using Function Code 02:

Bus_Link Add 0 *Digital i/p register 20-27 -> Outstation 20 Digital Output 1-8

7.3.2 Modbus Slave Mode

If ***Bus_Link*** is configured to operate in Modbus slave mode, it will respond to the following commands from a master device:

Function Code	**Modbus meaning**	***Bus_Link*** **action**
01	Read Coil Status	Read digital outputs
02	Read Input Status	Read digital inputs
03	Read Holding Registers	Read output registers
04	Read Input Registers	Read input registers
05	Force Single Coil	Write to digital output
06	Pre-set Single Register	Write to output register
15	Force Multiple Coils	Write to digital outputs
16	Pre-set Multiple Registers	Write to output registers

As described in 7.2 above, an outstation configured as a ***Bus_Link*** slave will interpret any command that is reading any digital/register other than those used by hardware I/O as addressing an output digital/register. Similarly any command writing to a digital/register will address the relevant input.

7.4 Allen Bradley DF1 Protocol

Micro_Link supports serial communication using DF1 protocol on its ***Bus_Link*** port. This allows it to interface to Allen-Bradley (Rockwell Automation) PLC's and similar instrumentation or SCADA systems that support the DF1 instruction set. The ***Micro_Link*** DF1 protocol driver uses full duplex communications, no handshaking, and BCC error correction.

In DF1 PLC's the database is constructed from a number of files, with each file containing a number of elements. Each element can contain one or more sub-elements, which are each 16-bit registers. Most PLC's limit the number of flies, elements and sub-elements to a maximum of 254, thus allowing 8-bit addressing of each. However, the address 255 signifies that the PLC uses extended addressing, so is followed by a 16-bit value, thus greatly extending the address range.

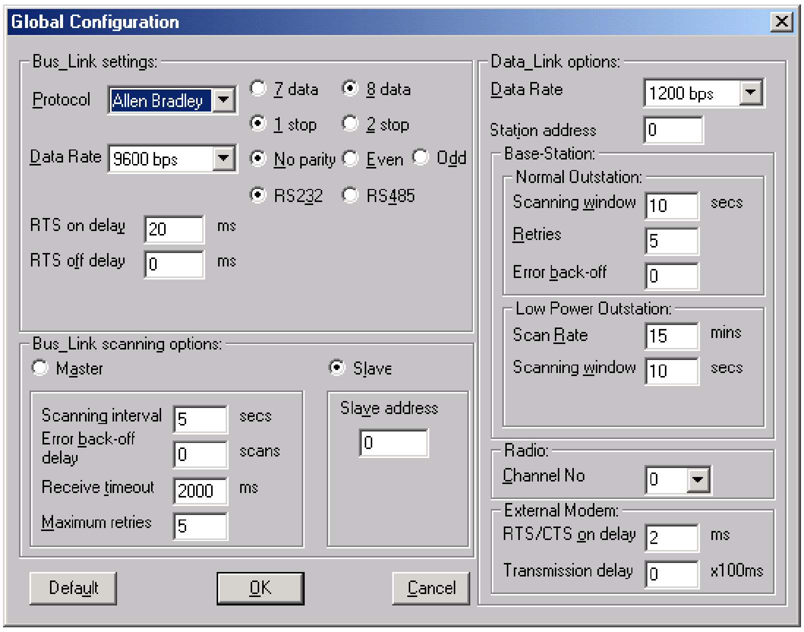
The function of each register is defined by the user, so there is no fundamental distinction between digital data, integer values and counts. However, when a register is accessed via the DF1 protocol the register type is included in the command. The PLC may reject a DF1 command if the register type defined within it differs from that defined by the user.

Micro_Link supports only Protected Typed Read/Write commands with three address fields, and doesn't support extended addressing. These commands have six supplemental bytes, namely Function Code, File Type, Byte Size, File Number, Element Number and Sub-element Number. Since ***Micro_Link*** uses 16-bit addressing it treats each internal address as two 8-bit values, where the most significant bits represent the file number and the least significant bits represent the element number. The sub-element is always 0.

When passing digital data to/from DF1 registers, if the number of bits is less than 16 they are located in the least significant bits with the remainder being 0. It is not possible to concatenate bits from non-sequential ***Micro_Link*** digital registers into a given DF1 register.

Micro_Link can be configured in either slave (server) or master (client) mode. All recent Allen Bradley PLC's can also be configured as either client or server, but it is usually easier to configure ***Micro_Link*** as a master (client) and the PLC as a server.

Being a full-duplex protocol, DF1 makes provision for commands to be transmitted faster that the receiver can respond, so each message is numbered, and the originator has to check the message number of each response to identify the command to which it relates. If the receiver misreads a command it can send an ENQ response to ask the transmitter to re-send it. However, ***Micro_Link*** only operates in a half-duplex mode, so it expects a response to each command before it sends the next. Timing is therefore critical for successful communications. The ***Bus_Link*** RTS delays should be left at their default value, and the Receive time-out should be set to 2 seconds:



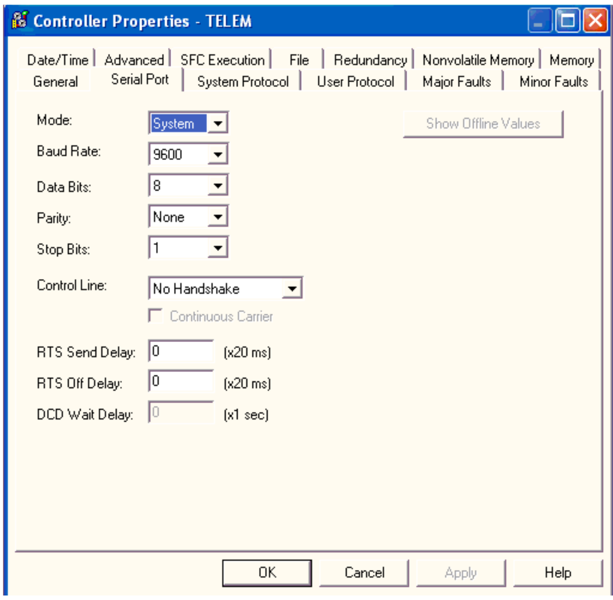
Global Configuration

Bus_Link settings:
Protocol: Allen Bradley
○ 7 data ● 8 data
● 1 stop ○ 2 stop
Data Rate: 9600 bps
● No parity ○ Even ○ Odd
● RS232 ○ RS485
RTS on delay: 20 ms
RTS off delay: 0 ms

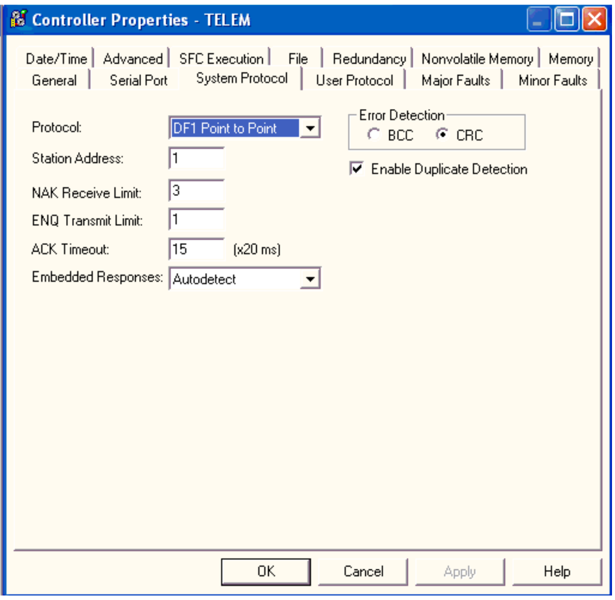
Bus_Link scanning options:
○ Master ● Slave
Scanning interval: 5 secs
Error back-off delay: 0 scans
Receive timeout: 2000 ms
Maximum retries: 5
Slave address: 0

Data_Link options:
Data Rate: 1200 bps
Station address: 0
Base-Station:
Normal Outstation:
Scanning window: 10 secs
Retries: 5
Error back-off: 0
Low Power Outstation:
Scan Rate: 15 mins
Scanning window: 10 secs
Radio:
Channel No: 0
External Modem:
RTS/CTS on delay: 2 ms
Transmission delay: 0 x100ms

The DF1 RTS delays should be set to 0, with no handshaking and the protocol should be set to Point to Point. ***Micro_Link*** 's with firmware earlier than V3.18 use BCC error detection, but V3.18 and later use more secure CRC checking. The ACK timeout should be set to 300ms. The following screen shots are typical if using ControlLogix configuration program:


Controller Properties - TELEM
Date/Time | Advanced | SFC Execution | File | Redundancy | Nonvolatile Memory | Memory
General | Serial Port | System Protocol | User Protocol | Major Faults | Minor Faults
Show Offline Values
OK
Cancel
Apply
Help

Mode: System
Baud Rate: 9600
Data Bits: 8
Parity: None
Stop Bits: 1
Control Line: No Handshake
□ Continuous Carrier
RTS Send Delay: 0 (x20 ms)
RTS Off Delay: 0 (x20 ms)
DCD Wait Delay: 0 (x1 sec)


Controller Properties - TELEM
Date/Time | Advanced | SFC Execution | File | Redundancy | Nonvolatile Memory | Memory
General | Serial Port | System Protocol | User Protocol | Major Faults | Minor Faults
OK
Cancel
Apply
Help

Protocol: DF1 Point to Point
Station Address: 1
NAK Receive Limit: 3
ENQ Transmit Limit: 1
ACK Timeout: 15 (x20 ms)
Embedded Responses: Autodetect
Error Detection ○ BCC ● CRC
☑ Enable Duplicate Detection

7.4.1 DF1 Master Mode

If ***Bus_Link*** is configured to operate in Allen Bradley master mode, it will send the following commands to a slave device:

***Bus_Link* action**	**DF1 command**
Read slave data	Protected typed logical read with three address fields
Write slave data	Protected typed logical write with three address fields to an integer register

To specify these in the ***Micro_Link*** mapping table the user must concatenate the File Number and Element Number into a single two-byte number, where the most significant byte is the File Number and the least significant byte is the element number. For example:

***Bus_Link* action**	**Address Calculation**	**Address (Hex)**	**Address (Decimal)**
Read/Write DF1 File No 31, Element 0	31 * 256 + 0	0x1F00	7936
Read/Write DF1 File No 20, Element 123	20 * 256 + 123	0x147B	5243

The sub-element is always 0.

If the transfer is passing analogue data to/from ***Bus_Link*** the file type will be set to 0x89 (integer), so the data will be transferred from/to an N register. If it is passing digital data the file type will be 0x85 (bit) so the transfer will be from/to a B register. If the user needs the data in a different type of register he must configure the PLC to use a dummy register and transfer the data internally within the PLC to/from this register.

For example, to copy the least significant bits of register B20:123 (File No 20, Element 123) of a PLC set to slave address 0 to the first two digital outputs at outstation 20 the following line would be needed:

```
Bus_Link Add 1 *Digital o/p reg 5243-5244      ->      Outstation 20 Digital Output 1-2
```

This is a little deceptive in that it suggests the source data is from two registers. However, since only 2 bits are required it will copy only bits 0 and 1 from PLC register 5243 (File No 20, Element 123), and all the remaining bits will be ignored. If more than 16 bits were specified then ***Bus_Link*** will read successive elements and concatenate the bits from them to the relevant outstation digital outputs.

To copy data the other way (i.e. to copy the first two digital inputs at outstation 20 to the least significant bits of register B20:123) the following line would be needed:

```
Outstation 20 Digital Input 1-2      ->      Bus_Link Add 1 *Digital o/p register 5243-5244
```

In this case since only 2 bits are specified, the outstation inputs will be copied to bits 0 and 1 of PLC register 5243 (File No 20, Element 123) and all remaining bits of this register will be set to 0. PLC register 5244 (File No 20, Element 124) will be unchanged. However, if the number of bits specified was greater than 16 then ***Bus_Link*** will write to successive elements.

To copy registers N31:0 and N31:1 (File No 31, Element 0…1) to analogue outputs 1 & 2 at outstation 10 the following line would be needed in a base-station:

Bus_Link Add 0 *Analogue i/p register 7936 -> Outstation 10 Analogue Output 1-2

Since DF1 protocol does not distinguish between inputs and outputs. *Analogue i/p register and *Analogue o/p register have the same meaning, as do * Digital i/p register and * Digital o/p register

7.4.2 DF1 Slave Mode

If ***Bus_Link*** is configured to operate in DF1 slave mode, it will respond to the following commands from a master device:

DF1 command	Command	Function	***Bus_Link* action**
Protected typed logical read with three address fields	0x0F	0xA2	Read input or output digitals or registers
Protected typed logical write with three address fields	0x0F	0xAA	Write to input or output digitals or registers

These commands have six supplemental bytes, namely Function Code, File Type, Byte Size, File Number, Element Number and Sub-element Number.

Bus_Link uses the File Type to define the type of data contained, file type 0x85 being bit (digital) data and file type 0x89 be integer (analogue) data. Byte Size will always be a multiple of 2 and defines the number of bytes being read/written. File Number is used to define the block number and the Element to define the address within the block. The sub-element is not used. For example, a base-station could receive the following commands:

Command	Function	File Type	File Number	Element	No of bytes	***Bus_Link* action**
0x0F	0xA2	0x85	10	0	2	Read Outstation 10 digital inputs. Note that bits 0…7 are the alarm flags, and bits 8…15 are the digital outputs (see 6.3 above)
0x0F	0xA2	0x85	10	16	4	Read Outstation 10, first two digital input expansion modules.
0x0F	0xAA	0x85	10	8	2	Write to Outstation 10 digital outputs. Bits 0…7 will be copied to the digital outputs (see 6.3 above). Bits 8…15 will be copied to the first 8 outputs of the first digital expansion module (if fitted).
0x0F	0xAA	0x85	10	16	4	Write to Outstation 10, first two digital output expansion modules.
0x0F	0xA2	0x89	20	6	4	Read Outstation 20 analogue inputs 1 & 2
0x0F	0xAA	0x89	20	6	4	Write to Outstation 20 analogue outputs 1 & 2

As described in 7.2 above, an outstation configured as a ***Bus_Link*** slave will interpret any command that is reading any digital/register other than those used by hardware I/O as addressing an output digital/register. Similarly any command writing to a digital/register will address the relevant input.

7.5 Mitsubishi Protocol

Mitsubishi PLC's can be configured to either dedicated protocol mode or no protocol mode, and offer a wide selection of data rates and character formats. No protocol mode requires the user to define the protocol in ladder logic, which is an onerous task. ***Bus_Link*** therefore supports the dedicated protocol. There are two slightly different variants, designated FX and non-FX, depending on the type of PLC. When configured as a master device ***Micro_Link*** accommodates this by sending a test message to the PLC at power-up to identify the type. When configured as a slave device ***Micro_Link*** emulates FX mode.

The default mode of FX PLC's is no protocol, so the protocol must be configured within it. This requires a line within the PLC programme to write the value H6881 (or decimal 26753) to register D8120 if using RS232 communications, or H6081 (decimal 24705) if using RS485. This configures the serial port to dedicated protocol at 9600 baud, using 8 bits/character, no parity and 1 stop bit. The ***Bus_Link*** port on ***Micro_Link*** must be configured to match this, via the Global parameters of DCD Configuration.

Note that some Mitsubishi serial interface modules do not support dedicated protocol.

By default, any Mitsubishi PLC is allocated as address 0 on its serial port. This can be changed on FX PLC's by writing a different value to register D8121, but this would only be necessary if connecting more than one PLC to the ***Bus_Link*** port.

The PLC configuration defined also sets the port to ignore control lines. However, non-FX PLC's use flow control when configured for RS232. If using RS232 the following interconnecting cable connections are recommended:

Function	PLC (9-pin D plug)	*Bus_Link* port	Category 5 cable colour
TXD	3	3	Green
RXD	2	4	White/Blue
0V	5	5	Blue
RTS	4	Link together	
CTS	6		
DSR	7	Link together	
DTR	8		

Mitsubishi PLC's support numerous types of registers, including:

Bit Registers (holding binary data):

Range from	To	Address	Function
X0000	X07FF	Hex/Octal*	Digital Input
Y0000	Y07FF	Hex/Octal*	Digital Output
M0000	M2047	Decimal	Internal Relay
M9000	M9255	Decimal	Special Relay
L0000	L2047	Decimal	Latch Relay
S0000	S2047	Decimal	Step Relay
B0000	B03FF	Hexadecimal	Link Relay

* FX PLC's use octal addressing, all others use hexadecimal

Word Registers (holding 16-bit values):

Range from	To	Address	Function
D0000	D1023	Decimal	Data Register
W0000	W03FF	Hexadecimal	Link Register
R0000	R8191	Hexadecimal	File Register

7.5.1 Mitsubishi Master Mode

The type of register accessed via ***Bus_Link*** is determined by the definition of input/output and the address range used:

a) Digital Inputs

0…2047	reads registers X0000…X07FF(hex)
10000…19999	reads registers M0000…M9999
20000…29999	reads registers L0000…L9999
30000…39999	reads registers S0000…S9999
40000…42047	reads registers Y0000…Y07FF(hex)
50000…51023	reads registers B0000…B03FF(hex)

For example, to copy the state of register M0001 to the first digital output on a ***Micro_Link*** the following line would be included in the data routing table:

Bus_Link Add 0 *Digital i/p register 10001 -> Local Digital Output 1

b) Digital Outputs

0…2047	writes to registers Y0000…Y07FF(hex)
10000…19999	writes to registers M0000…M9999
20000…29999	writes to registers L0000…L9999
30000…39999	writes to registers S0000…S9999
40000…42047	writes to registers X0000…X07FF(hex)
50000…51023	writes to registers B0000… B03FF(hex)

For example, to copy the state of the first digital input on a ***Micro_Link*** to register L0003 the following line would be included in the data routing table:

Local Digital Input 1 -> Bus_Link Add 0 *Digital o/p register 20003

c) Register Inputs

0…9999	reads registers D0000…D9999
10000…19999	reads registers W0000…W9999
20000…29999	reads registers R0000… R270F (hex)

For example, to copy the state of register W0001 to the first analogue output on a ***Micro_Link*** the following line would be included in the data routing table:

Bus_Link Add 0 *Register Input 10001 -> Local Analogue Output 1

d) Register Outputs

0…9999	writes to registers D0000…D9999
10000…19999	writes to registers W0000…W9999
20000…29999	writes to registers R0000… R270F(hex)

For example, to copy the state of the first analogue input on a ***Micro_Link*** base-station to register R0003 the following line would be included in the data routing table:

Local Analogue Input 1 -> Bus_Link Add 0 *Digital o/p register 20003

7.5.2 Mitsubishi Slave Mode

In slave mode ***Bus_Link*** emulates a non-FX PLC supporting only bit register read/write in the range 0…8000 and word register read/write in the range 0…2000.

As with all ***Bus_Link*** slave implementations the user must calculate the required register within the ***Micro_Link*** database. For example, to read the first digital input at outstation address 10 the required register is (32 * 10 + 8) = 328, so the host system would have to read for example M0328 or X0148 (328 decimal = 148 hex). Similarly to write to digital output 4 at outstation address 20 the required register is (32 * 20 + 11) = 651, so the host would have to write to M0651 or Y028B (651 decimal = 028B hex).

As described in 7.2 above, an outstation configured as a ***Bus_Link*** slave will interpret any command that is reading any digital/register other than those used by hardware I/O as addressing an output digital/register. Similarly any command writing to a digital/register will address the relevant input.

7.6 Special Features of Bus_Link Configuration

7.6.1 Using specific I/O registers

It should be apparent from the above that if ***Bus_Link*** is configured as a slave device the host must be configured to read the appropriate address for the particular parameter required. For example, to read the value of analogue input 1 at outstation address 10 it should read register 87.

Some host systems, however, may have their mapping fixed in software and, for example expect to read analogue inputs as consecutive registers starting at address 1200. In this case, a ***Micro_Link*** base-station would normally recognise that ***Bus_Link*** is accessing data blocks starting from 150 (i.e. 1200 / 8) and try polling outstations starting from 150. If any of these existed they would return a standard data block containing 4 totalised counts followed by battery volts, RSSI, then the two analogue inputs. The data may not therefore be in the locations the host device expects.

To overcome this, a ***Micro_Link*** base-station recognises registers 10,000…11,999 as equivalent to 0…1,999, but will not attempt to poll outstations to read or write data. For example, if the following line is included in the data routing table:

Bus_Link Add 10 Analogue i/p 1 - 2 -> Local *Register Input 11200 - 11201

The host system can read the value of outstation 10 analogue inputs 1…2 from registers 1200…1201 without causing ***Micro_Link*** to try to poll outstation 15.

Obviously, if this feature is used the user must ensure that no outstations are allocated addresses within the range used by the host.

7.6.2 Copying Digital States to Registers

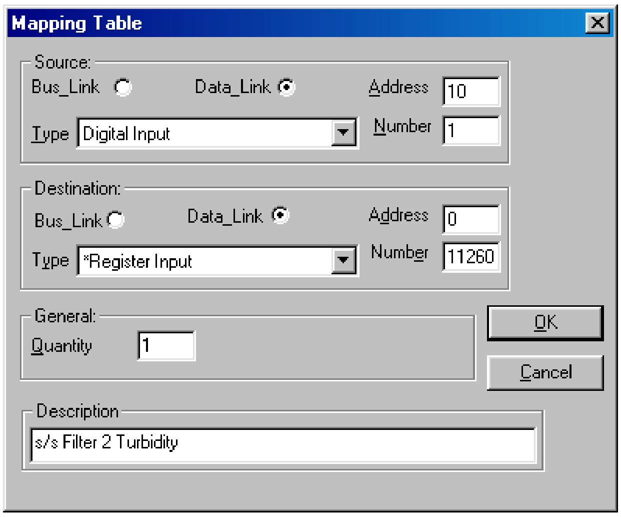
Situations sometimes arise whereby digitals need to be mapped into analogue registers. The most common application of this is where ***Bus_Link*** is used to pass data to/from a PLC or SCADA system that only processes 16-bit registers. These systems define digitals as particular bits within a register.

Micro_Link incorporates two methods for achieving this. The first, which was introduced in V2.25 (released in 2003) allowed one digital at a time to be transferred into a register, and used the Quantity field within the Mapping Table to identify the bit within the register. This has the disadvantage that the Mapping Table summary is not logical, and multiple entires are needed to transfer more than one digital. The second method, introduced in V3.10, is more logical and allows multiple digitals to be transferred in each line. It also allows data to be passed the other way (i.e. bits within a register to be transferred to digital states).

The two methods are described in detail:

7.6.2.1 Copying Digital States to Registers - Method 1:

If a digital is copied to a register in the range 10,000…11,999, ***Micro_Link*** treats it as a special case, and maps the digital into the bit position defined by Quantity. This is best illustrated by two examples:

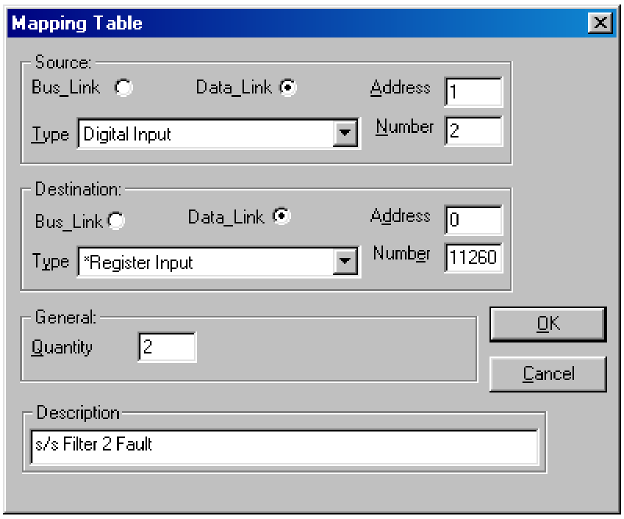


Source:
Bus_Link ○ Data_Link ● Address 10
Type Digital Input Number 1

Destination:
Bus_Link ○ Data_Link ● Address 0
Type *Register Input Number 11260

General:
Quantity 1

Description
s/s Filter 2 Turbidity



Source:
Bus_Link ○ Data_Link ● Address 1
Type Digital Input Number 2

Destination:
Bus_Link ○ Data_Link ● Address 0
Type *Register Input Number 11260

General:
Quantity 2

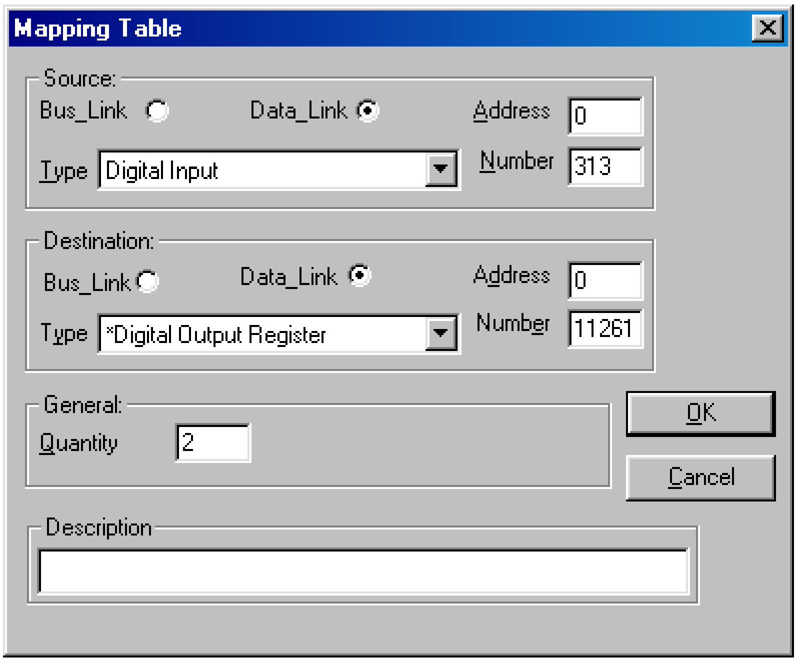
Description
s/s Filter 2 Fault

In these example digital input 1 at outstation address 10 will be copied into the first bit of register 1120 (i.e. bit 0) and digital input 1 at outstation address 10 will be copied into the second bit of register 1120 (i.e. bit 1).

The summary shown in the data routing table is a little confusing, but can be interpreted:

Outstation Address 10 Digital Input 1	->	Local *Register Input 11260
Outstation Address 10 Digital Input 2	->	Local *Register Input 11260 - 11261

If the user wishes to access alarm flags (such as Comms Fail) there is a further complication. DCD config software knows that an alarm flag is only a single bit, so will not allow a Quantity of greater than 1. Furthermore, DCD will attempt to translate any absolute addressing within This can be overcome by resorting to absolute addressing. Since the Comms Fail alarm is the first digital bit in the relevant data block, and there are 32 digitals per data block, the Comms Fail alarm for, say, outstation 10 is absolute address (32 * 10) + 0 = 320. Therefore the following entry:



Mapping Table

Source:
Bus_Link ○ Data_Link ● Address 0
Type Digital Input Number 313

Destination:
Bus_Link ○ Data_Link ● Address 0
Type *Digital Output Register Number 11261

General:
Quantity 2

Description

will cause outstation 10 Comms Fail alarm to be copied to the second bit of register 1261. When this is entered, however, DCD will translate it into the equivalent of a digital input:

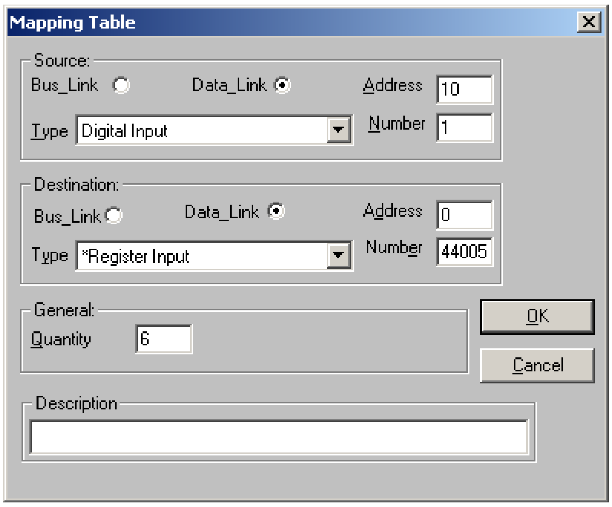
Local Digital Input 313 - 314 → Local *Register Input 11261 - 11262

7.6.2.2 Copying Digital States to Registers - Method 2:

This is the preferred method, but can only be used in V3.10 or later software. It requires the user to calculate a new value for the register as (x + 1000) * 20. Thus, for example, register 1200 will become register 44000 and register 1201 will become register 44020. The user must then add to this new value the required bit number, so bit 5 in register 1200 becomes register 44005 and bit 12 in register 1201 becomes register 44032.

Once the required register has been calculated any transfer can be achieved between digital and analogue registers.

For example, if the following entries are included in a base-station data routing table:

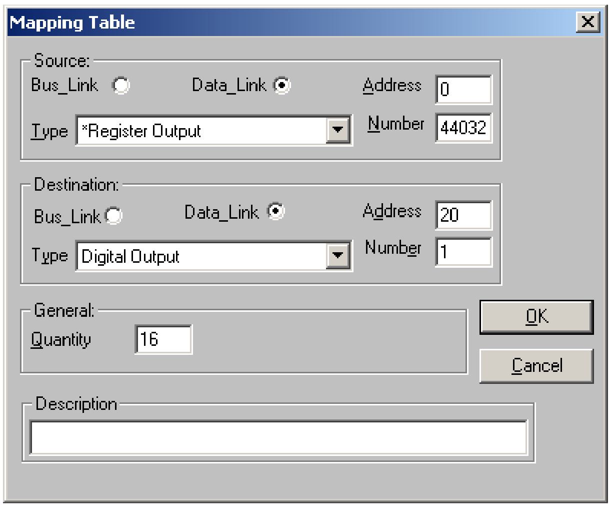

Mapping Table

Source:
Bus_Link ○ Data_Link ● Address 10
Type Digital Input Number 1

Destination:
Bus_Link ○ Data_Link ● Address 0
Type *Register Input Number 44005

General:
Quantity 6

Description


Mapping Table

Source:
Bus_Link ○ Data_Link ● Address 0
Type *Register Output Number 44032

Destination:
Bus_Link ○ Data_Link ● Address 20
Type Digital Output Number 1

General:
Quantity 16

Description

The mapping table summary for these two lines will show:

Outstation 10 Digital Input 1 - 6	→	Local *Reg Input 44005 - 44010
Local *Reg Output 44032 – 44047	→	Outstation 20 Digital Input 1 - 16

To decipher these lines, divide the register number by 20 and subtract 1000 to derive the register address, and derive the number of the first bit as the remainder after dividing by 20. The first line will therefore copy the first six digital inputs from outstation 10 to bits 5…10 of Input Register 1200. The second line will copy bits 12…15 of register 1201 to digital outputs 1…4 at outstation 20 and bits 0…11 of register 1202 to digital outputs 5…16 at outstation 20.

8. Special Features of Micro_Link

Many special features can be activated in ***Micro_Link*** by configuring it appropriately. Configuration changes are carried out using the DCD Terminal, as described in chapter 9.1.

8.1 Exception Reporting

8.1.1 General

Exception Reporting permits critical changes to be reported without waiting for the relevant outstation to be polled. It functions in both directions, so the base-station can interrupt its normal poll sequence to send a critical change to an outstation. Conversely, an outstation can send an unsolicited message to the base-station without needing to wait until it is next polled. Typical applications are:

- A battery-powered outstation may be monitoring parameters that only change occasionally. Polling it frequently would waste battery life unnecessarily.
- Digital inputs at an outstation may be used to monitor alarm states. These need to be reported as rapidly as possible.
- Analogue values from an outstation may be used within control loops, where instability or excess inertia may result if changes are not reported quickly.
- One or more digital inputs at the base-station may be used to operate controls at an outstation. It may be important that the response time is minimised.

Exception Reporting can be applied to the internal I/O of ***Micro_Link*** and ***Nano_Link***, expansion modules connected to ***Micro_Link***, and to external I/O accessed via the ***Bus_Link*** serial port of a ***Micro_Link***. However, it cannot be applied to digital expansion modules connected to a ***Nano_Link*** outstation, or to a ***Nano_Link*** configured as a base-station.

Exception reporting from a base-station to an outstation can only be used with mains powered outstations, since battery-powered outstations will put themselves to sleep between normal polls, so will not be able to receive exception reports. Exception reporting from an outstation to a base-station can only be used if the system is configured to not scan continuously, since there must be silent periods during which exception reports can be sent.

The effect of Exception Reporting on the poll sequence is described in Section 6.7.

8.1.2 Configuring Exception Reporting

8.1.2.1 Digital exceptions from the base-station to an outstation

To ensure that digital changes at the base-station are reported to an outstation without delay, the base-station configuration table needs an entry such as the following

Exception Digital → Outstation 10 Digital Output 5 - 8

This will ensure that any changes at the base-station which affect digital outputs 5, 6, 7 or 8 at outstation address 10 will be sent immediately. For example, the following line could also be present in the data routing table:

Local Digital Input 9 - 12 → Outstation 10 Digital Output 5 - 8

A change on base-station digital inputs 9, 10, 11 or 12 will be sent immediately to the outstation. The input data could also be derived via the ***Bus_Link*** port at the base-station.

As stated above, exception reporting from the base-station to an outstation can only be used with mains-powered outstations.

8.1.2.2 Digital exceptions from an outstation to the base-station

To ensure that digital changes at an outstation are reported to the base-station without delay, the base-station configuration table needs an entry such as the following:

Outstation 10 Digital Input 1 - 4 → Local Digital Output 1 - 4

Exception Digital → Outstation 10 Digital Input 1 - 3

This will ensure that any changes on digital inputs 1, 2 or 3 at outstation address 10 will be reported by exception. Digital changes on other inputs at that outstation will not cause exception reports to be generated. In the example shown, changes on digital input 4 at the outstation will only be copied to the relevant output at the base-station next time the outstation is polled. However, if the outstation sends an exception report, the state of all inputs will be sent, so input 4 will be updated at the same time.

8.1.2.3 Analogue exceptions from the base-station to an outstation

To closely track an analogue value it is may be necessary to update it when a significant change occurs. Since analogue inputs are calibrated to give a numeric value in the range 0…4000, a change of, say 2% would represent a difference of 80. To report such a change at the base-station to an outstation without undue delay the following entry could be made:

Exception Analogue = 80 → Outstation 10 Analogue Output 1 - 2

This will cause the base-station to send an exception report to the outstation if the required analogue out value changes by more than 2% from the last value sent to it.

If the data for this outstation was derived from analogue inputs within the base-station the following entry may be included:

Local Analogue Input 2 - 3 → Outstation 10 Analogue Output 1 - 2

Since a ***Micro_Link*** base-station scans its analogue inputs every 5 seconds, there could be a delay of this amount before the change is sent to the outstation.

However, if the data was derived from a device connected to the ***Bus_Link*** port of the base-station, any significant changes will be sent immediately they are passed to the base-station.

As stated above, exception reporting from the base-station to an outstation can only be used with mains-powered outstations.

8.1.2.4 Analogue exceptions from an outstation to the base-station

To closely track an analogue value at the outstation a similar methodology is applied. For example, the following lines could be included in the base-station configuration:

Exception Analogue = 80 → Outstation 10 Analogue Input 1

Exception Analogue = 200 → Outstation 10 Analogue Input 2

This will ensure that a change of more than 2% on analogue input 1 or 5% on analogue input 2 at outstation address 10 will be reported to the base-station by exception. Obviously the lower the value set the more closely the base-station output will track it, but the more often the outstation will send exception reports.

Note, however, that analogue inputs are only sampled at defined times. A ***Micro_Link*** outstation scans all analogue inputs every 5 seconds. ***Nano_Link*** outstations normally only scan analogue inputs when interrogated by the base-station.

However, if analogue exceptions are configured into a ***Nano_Link*** outstation it will by default scan analogue inputs every 10 seconds. This can be over-ridden by adding the line

Exception Scan Rate (x10secs) = 12 → Outstation 10 Analogue Input 1

Only one such line is necessary for any given ***Nano_Link*** outstation, and it will apply to all its analogue inputs. The above line will cause analogue inputs to be sampled every 2 minutes, independent of scanning by the base-station. The Analogue Scan Rate must be in the range 1…255 or 10 seconds to 42.5 minutes.

The outstation's 10VDC power output can be used to power the transducers if required. If the outstation is battery-powered it will switch on the 10V supply before taking a reading, and wait for the transducers to stabilise. As explained in the ***Nano_Link*** Technical Manual, the warm-up time of analogue transducers can be set to either 250ms, 1 second or 30 seconds by changing a DIPswitch on the outstation. 250ms is generally long enough for pressure transducers, but standard ultra-sonic transducers may need 30 seconds before they give a valid output.

The current drawn from the battery rises from a quiescent level of around 200µA to about 3 x the transducer current, so could be 60mA for the time the transducer is on. It also rises to around 1A for about 0.5 seconds when it transmits a message to the base-station. To maximise battery life when reading analogues at an outstation therefore adopt the following:

- Configure exception reporting
- Increase the background poll rate to, say, 12 hours
- Configure the analogue scan rate to the longest time consistent with the required response time
- Set the analogue exception trip level to the highest acceptable value
- Do not use standard ultra-sonic transducers (use either ultra-sonic transducers designed for battery operation, or another technology)

Note that the above description applies equally to registers that may be read into a ***Micro_Link*** outstation via ***Bus_Link***. These could originate in a PLC, and may be up to 16 bits.

8.1.2.5 Digital exceptions from powered transducers at a Nano_Link outstation

Some digital transducers require power before they will give a valid reading. An example is the Stormlog level sensor, which requires a 5V supply to energise a photo-electric cell. A battery-powered ***Nano_Link*** outstation can be configured to operate with these using a combination of the above techniques.

Nano_Link has 4 digital inputs (1…4), plus two pseudo inputs (5 & 6) (Comms Fail and Battery Low). Digital inputs 7 and 8 copy digital inputs 1 and 2, but are only updated when the outstation has raised 10V. If exception reporting is configured for these inputs, the outstation interprets the requirement as exception reporting on powered digital transducer connected to inputs 1 and 2 respectively. For example, the following entry could be made in the base-station configuration table:

Outstation 10 Digital Input 1 → Local Digital Output 3

Exception Digital → Outstation 10 Digital Input 7

This will configure the outstation to ensure that digital input 1 is only read when the 10VDC supply is present. By default the outstation will sample the input every 10 seconds, but this can be over-ridden by setting the analogue scan rate:

Exception Scan Rate (x10secs) = 30 → Outstation 10 Analogue Input 1

This example will ensure that every 5 minutes all transducers configured for exception reporting will be powered, then read after allowing time for them to 'warm-up'. The Analogue Scan Rate must be in the range 1…255 or 10 seconds to 42.5 minutes. The warm-up time allowed will be either 250ms, 1 second or 30 seconds, depending on the DIPswitch settings at the outstation.

Note that the power provided for the transducers is 10VDC. If they operate from 5VDC an external voltage regulator will be needed.

8.1.2.6 Count exceptions from an outstation to the base-station

It may only be necessary to send totalised counts from an outstation when a certain number of increments have occurred:

Exception Count = 20 → Outstation 10 Count 1

This will cause the outstation to transmit an exception message when the total count on digital input 1 reaches 20 more than the last value reported to the base-station. The value of Count Exception must be in the range 1…255.

8.2 Initiation Delays

8.2.1 General

Some applications require that short duration digital input changes at outstations should be ignored. A typical example is monitoring of sewage vacuum chambers, where a valve may open for up to 5 minutes in normal operation, but an alarm must be raised if it remains open longer than this.

The facility of setting initiation delays allows digital changes that last less than a defined period to be ignored. It is of particular relevance when used in conjunction with exception reporting, since it prevents unnecessary transmissions from a battery-powered outstation that would otherwise waste battery power. It also overcomes the need for external processing to distinguish between alarm conditions and normal operation.

Initiation delays are implemented via a ***Micro_Link*** base-station by adding lines in the data routing table such as:

Initiation Delay (x2 secs) = 5	→	Outstation 10 Digital Input 1
Initiation Delay (x2 secs) = 150	→	Outstation 10 Digital Input 2-3

The effect of this is that a change of state on digital input 1 at outstation 10 will be ignored unless is persists for 10 seconds. Note this applies to both contact closure and contact opening. Similarly, changes of state on digital inputs 2 or 3 must persist for 5 minutes (300 seconds) before they are recognised.

8.3 Defining Digital Output States After Communications Failure

By default, all digital outputs freeze at their last valid state in the event of a communications failure. Setting the *Value on Comms Fail* via DCD Configuration can change this.

A base-station generates a comms fail alarm if a given outstation fails to respond after a defined number of re-tries (see 9.1.1.3).

An outstation generates a comms fail alarm if it is not interrogated when it expects to be. It calculates when it expects to be interrogated by measuring the time between successive identical commands it receives from the base-station. It adds to this time a contingency of twice the Scan Window value downloaded to it from DCD Configuration (see 9.1.1.3).

Example 1:

Suppose the following lines are included in the data routing table of a base-station that is communicating with an outstation set to address 10:

Value on Comms Fail = 0	→	Outstation Address 10 Digital Input 1-2
Value on Comms Fail = 1	→	Outstation Address 10 Digital Input 3-4

The base-station will interpret digital inputs 1 & 2 at the outstation as ‘0’ in the event of a comms failure, and inputs 3 & 4 as ‘1’. All other inputs will remain in their last valid state.

Example 2:

Suppose the following lines are included in the data routing table of an outstation:

Value on Comms Fail = 0	→	Local Digital Output 1-2
Value on Comms Fail = 1	→	Local Digital Output 3-4

Suppose furthermore that the base-station Normal Scanning Window (defined in the Global settings) is set to 5 seconds.

The digital outputs will normally copy the state written to them by the base-station. Each time the base-station writes to the outstation, the outstation will preset an internal comms fail timer with the time interval since the last write command from the base-station plus 10 seconds (i.e. 2 x the Scanning Window). If this timer times out before the base-station next writes to the outstation, then the outstation will flag a comms fail alarm. This will cause it to force digital outputs 1 & 2 to ‘0’ and digital outputs 3 & 4 to ‘1’. All other outputs will remain in their last valid state.

8.4 Defining Analogue Output States After Communications Failure

By default, all register outputs freeze at their last valid state in the event of a comms failure. Setting the *Value on Comms Fail* via DCD Configuration can change this.

A base-station generates a comms fail alarm is a given outstation fails to respond after a defined number of re-tries (see 9.1.1.3).

An outstation generates a comms fail alarm if it is not interrogated when it expects to be. It calculates when it expects to be interrogated by measuring the time between successive identical commands it receives from the base-station. It adds to this time a contingency of twice the Scan Window value downloaded to it from DCD Configuration (see 9.1.1.3).

Example 1:

Suppose the following lines are included in the data routing table of a base-station that is communicating with an outstation set to address 10:

Value on Comms Fail = 0	→	Outstation Address 10 Analogue Input 1-2
Value on Comms Fail = 2000	→	Outstation Address 10 Analogue Input 3-4
Value on Comms Fail = 4000	→	Outstation Address 10 Analogue Input 5

The base-station will interpret analogue inputs 1 & 2 at the outstation as dropping to zero in the event of a comms failure, inputs 3 & 4 as 50% of full scale and analogue 5 as full scale. All other inputs will remain in their last valid state.

Example 2:

Suppose the following lines are included in the data routing table of an outstation:

Value on Comms Fail = 0	→	Local Analogue Output 1-2
Value on Comms Fail = 2000	→	Local Analogue Output 3-4
Value on Comms Fail = 4000	→	Local Analogue Output 5

Suppose furthermore that the base-station Normal Scanning Window (defined in the Global settings) is set to 15 seconds.

The analogue outputs will normally copy the state written to them by the base-station. Each time the base-station writes to the outstation, the outstation will preset an internal comms fail timer with the time interval since the last write command from the base-station plus 30 seconds (i.e. 2 x the Scanning Window). If this timer times out before the base-station next writes to the outstation, then the outstation will flag a comms fail alarm. This will cause it to force analogue outputs 1 & 2 to zero, analogue outputs 3 & 4 to 50% of full scale and analogue output 5 as full scale. All other outputs will remain in their last valid state.

8.5 Inverting Digital Inputs

By default all digital inputs register a value of ‘1’ when the corresponding input is a closed contact (or a voltage less than 2V), and a logic ‘0’ when the corresponding input is an open circuit (or a voltage greater than 3V). However, this can be changed:

Example:

Suppose the following line is included in the data routing table:

```
Invert digital                              →      Local Digital Input 1-3
```

The first 3 digital inputs will be inverted.

8.6 Inverting Digital Outputs

By default all digital outputs are contacts which close for a logic ‘1’ state and open for logic ‘0’. However, this can be changed:

Example:

Suppose the following line is included in the data routing table:

```
Invert digital                              →      Local Digital Input 1-3
```

The first 3 digital inputs will be inverted.

However, this can be changed:

Example:

Suppose the following lines are included in the data routing table of a base-station:

```
Outstation 10 Digital Input 1               →      Local Digital Output 1
Outstation 10 Digital Input 1               →      Local Digital Output 2
Invert digital                              →      Local Digital Output 2
```

The first two lines copy the first digital input from the outstation to the first and second outputs at the base-station. The third line inverts the second output, thus creating a changeover contact mimicking the outstation input.

8.7 Scaling Analogue Inputs

All analogue inputs give a register span of 0…4000 for their calibrated input range. For example, an input calibrated 0…20mA will give a register value of 0 for 0.00mA, 2000 for 10.00mA and 4000 for 20.00mA. Note that an input of 4…20mA will give a span of 800…4000, which will be presented as 4…20mA at the remote end. However, applying the following formula can change this:

$$\frac{Multiplier}{Divisor} * (Value - Offset)$$

Where *Value* is the raw input reading. The calculation will only be performed if *Multiplier* is defined. The default values for *Divisor* and *Offset* are 1 and 0 respectively. Negative results are converted to zero.

Example:

Suppose a ***Micro_Link*** has analogue inputs calibrated 0…20mA, but is required to give a register range of 0…4000 for inputs over the range 4…20mA (i.e. 800…4000) on input 1 and 0…10mA on input 2. The following lines should be included in the base-station data routing table:

```
Multiplier = 4000                           →      Local Analogue Input 1
Divisor = 3200                              →      Local Analogue Input 1
Offset = 800                                →      Local Analogue Input 1
Multiplier = 2                              →      Local Analogue Input 2
```

Note that the default values of *Divisor* = 1 and *Offset* = 0 will be applied for analogue input 2.

8.8 Scaling Analogue Outputs

All analogue outputs are calibrated for a register span of 0…4000. For example, an output calibrated 0…20mA will give an output of 0.00mA for a register value of 0, 10.00mA for 2000 and 20.00mA for 4000. Note that a span of 800…4000 (as produced by an unscaled input at the remote end) will give an output range of 4…20mA. However, applying the following formula can change this:

$$\frac{Multiplier}{Divisor} * Value + Offset$$

Where *Value* is the raw input reading. The calculation will only be performed if *Multiplier* is defined. The default values for *Divisor* and *Offset* are 1 and 0 respectively.

Example:

Suppose a ***Micro_Link*** has analogue outputs calibrated 0…20mA, but output 1 is required to give a span of 4…20mA from a register range of 0…2000 (e.g. from a remote 0…10mA input), and output 2 is required to give a span of 0…10mA for a span of 0…4000. The following lines should be included in the base-station data routing table:

Multiplier = 3200	→	Local Analogue Output 1
Divisor = 2000	→	Local Analogue Output 1
Offset = 800	→	Local Analogue Output 1
Multiplier = 2000	→	Local Analogue Output 2
Divisor = 4000	→	Local Analogue Output 2

Note that the default value of *Offset* = 0 will be applied for analogue input 2.

8.9 Creating BCD Outputs from Analogue Inputs

Analogue outputs and totalised counts are sometimes required to be presented in real units on digital displays, which typically give 3, 4 or 5 decimal digits. For example, a reservoir depth may be measured by a transducer that gives a 4…20mA output, but need to be displayed as a digital value in the range 0…25.00 metres.

Micro_Link can be configured to present registers in Binary Coded Decimal (BCD) by copying the relevant register output to the first required digital output, and setting the block size to define the number of digitals. Note that each digit of a BCD output requires 4 bits. The register value is processed using the following formula before copying it to the digital outputs:

$$\frac{Multiplier}{Divisor} * (Value - Offset)$$

Example:

Suppose a 4…20mA transducer on analogue input 1 at outstation 10 is required to give a 4-decade BCD output, scaled 0…2500, from digital outputs 16…31 at the base-station. The following lines are required in the base-station's data routing table:

Outstation Address 10 Analogue Input 1	→	Local Digital Output 16 - 31
Multiplier = 2500	→	Outstation Address 10 Analogue Input 1
Divisor = 3200	→	Outstation Address 10 Analogue Input 1
Offset = 800	→	Outstation Address 10 Analogue Input 1

The block size should obviously be large enough to accommodate the largest possible output value, so will normally be a multiple of 4. If it is set to less than 5, ***Micro_Link*** interprets the command as a pulse rate output (see 0)

8.10 Driving Digital Display Modules

Micro_Link can output analogue and count values on the 7175-2 Digital Display Modules (see 4.5.2) as an alternative to current and digital outputs.

8.11 Calculating Average Flow Rate From Pulse Flowmeters

Many flowmeters produce only a pulse output, which may comprise a switch contact that briefly closed for, say, every 10 litres of water that passes through it. Turbine flowmeters are a good example, where the speed of rotation of the turbine is directly proportional to the rate of flow. An advantage of this type is that they don't require a power supply.

The output of these flowmeters can be monitored via an outstation using the digital/count inputs. The outstation could be a battery-powered ***Nano_Link*** or a permanently-powered ***Nano_Link*** or ***Micro_Link***. If required the corresponding base-station can be configured to generate an analogue output proportional to the pulse rate, using the formula:

$$Output = \frac{New\_Count - Last\_Count}{Scale\_Factor} * (4000 - Offset) + Offset$$

Where *New_Count* is the latest count read from the outstation and *Last_Count* is the reading taken some time earlier. *Scale_Factor* is the number of pulses counted between successive readings required to give full-scale output (i.e. 4000).

To achieve a reasonable resolution the time between successive updates should be long enough to accumulate a significant number of pulses, but it needs to be short enough to track varying flow rates relatively accurately. A reasonable compromise is to update the reading every 15 minutes, and use a flowmeter that gives more than 100 pulses in 15 minutes at the highest flow rate.

For example, if the flowmeter gives one pulse for every 10L, and the analogue output is required to be 20mA for a flow of 10000L/hour, then if updating every 15 minutes the *Scale_Factor* will be 250. If the output is to read 4mA when the flow rate drops to zero the value for *Offset* will be 800.

If using battery-powered outstations the time between successive updates will be the Low Power Scan Rate, which would typically be set to 15 minutes to conserve battery life.

If using permanently powered outstations the time between successive updates is defined by including a Pulse Average Time (mins) in the data routing table. **Note:** Requires ***Micro_Link*** firmware V3.21 or later and DCD Configuration software V2.19 or later.

If this feature is configured the count output at the base-station will only update each time the pulse rate is re-calculated, regardless how many times the outstation is polled between updates. Therefore if the count is also replicated on a digital output (see 8.12) the output pulse stream will only be triggered each time the pulse rate is re-calculated. The calculated flow rate is stored in the Result 1…4 of the outstation, and can be copied to any required analogue output.

If the data is not updated exactly when expected (e.g. if re-tries are needed to retrieve the data, or there is an intermittent comms fail) the base-station will automatically adjust the calculation based on the actual time between updates.

Example 1:

Suppose a ***Micro_Link*** base-station interrogates a battery-powered ***Nano_Link*** outstation (set to address 10) every 15 minutes, and the following lines are included in the base-station data routing table:

Scale Factor = 250	→	Outstation Address 10 Count 1
Scale Factor = 50	→	Outstation Address 10 Count 2
Offset = 800	→	Outstation Address 10 Count 1 - 2
Outstation Address 10 Result 1 - 2	→	Local Analogue Output 1 - 2
Value on Comms Fail = 4000	→	Outstation Address 10 Result 1

Analogue output 1 at the base-station will give a reading of 4…20mA for a pulse rate of 0…1000 pulses/hour on the first digital input at the outstation. Analogue output 2 will give a reading of 4…20mA for a pulse rate of 0…200 pulses/hour on the second digital input at the outstation (but note the resolution will be only 1 part in 50, or 2%). The calculated flow rate for count 1 will be forced to full scale in the event of a communications failure, while the flow rate for count 2 will freeze at the last valid value.

Example 2:

Suppose a ***Micro_Link*** base-station interrogates a permanently-powered outstation (set to address 20). The poll rate will depend on the Normal Scanning Window and the number of outstations being polled, but would usually be less than 5 minutes. The following lines could be included in the base-station data routing table:

Scale Factor = 250	→	Outstation Address 20 Count 1
Scale Factor = 50	→	Outstation Address 20 Count 2
Offset = 800	→	Outstation Address 20 Count 1 - 2
Pulse Average Time (mins) = 15	→	Outstation Address 20 Count 1 - 2
Outstation Address 20 Result 1 - 2	→	Local Analogue Output 1 - 2
Value on Comms Fail = 4000	→	Outstation Address 20 Result 1

It should be apparent that this is similar to the example above, but has one extra line. The calculation will be updated when more than 15 minutes has passed since the outstation was last updated. If the system is relatively large the poll rate could be, for example, 4 minutes. In the worst case 19 minutes could lapse between successive updates, but the calculated values will be corrected for this.

Note that only one Pulse Average Time can be set for any given outstation, regardless how many pulse counts are being used. The count output at the base-station will only be updated each time the calculated pulse rate is updated, but any counts that are not configured to give pulse rate outputs will be updated each time the outstation is polled.

8.12 Generating Pulse Outputs

The first four digital inputs at any station are accumulated into counts 1…4. These can be replicated as pulses at any other location by copying them to one of the first 16 digital outputs (i.e. the outputs on ***Micro_Link*** and/or the first 8 outputs on the first Digital Output Expansion module). ***Micro_Link*** maintains a record of the total number of pulses it has generated on each pulse output. Pulses are produced at a rate of 5 pps such that the total number of output pulses matches the total number of input pulses to the counter. It should be noted, however, that the output pulses occur in bursts after each scan, so the pulse rate may not exactly mimic the timing of the input pulses.

Micro_Link stores the total in non-volatile memory every 10 minutes and restores the total from this memory after a reset.

When a system is first installed, or if a module is subsequently changed, there is obviously a possibility that the remote input count total and the current running total of local output pulses could be significantly different. This could result in a large number of pulses being generated to bring the systems into sync. To prevent this, the output count total is forced to the input count total whenever the configuration is uploaded or downloaded, thus clearing pulse outputs.

Example:

Suppose a ***Micro_Link*** base-station is scanning two Nano_Link outstations (set to addresses 10 & 20), and is required to copy pulses fed to the first two digital inputs at each outstation onto its first 4 digital outputs. The following lines should be included in the base-station data routing table:

Outstation Address 10 Count 1 - 2	→	Local Digital Output 1 - 2
Outstation Address 20 Count 1 - 2	→	Local Digital Output 3 - 4

Since each outstation only accumulates counts for its first 4 digital inputs, the maximum block size is 4.

8.13 Bus_Link Monitoring and Control of Slow Scan Mode

The default Scanning Interval and Scanning Window are downloaded into a ***Micro_Link*** base-station by the DCD terminal. However, there may be times when the SCADA system needs to change the Scanning Interval. For example, under flood conditions there may be a need to increase the scan rate temporarily. To achieve this, the Scanning Interval and Scanning Window registers are mapped into the database, so the SCADA system can over-write them.

If the Scanning Interval is set to the same period at which the database is being read by a SCADA, there is a risk of data duplication or data loss:

Suppose the clock in a ***Micro_Link*** base-station is running slightly slower than the SCADA system clock. At some point in time the clocks will get into synchronism with each other. Prior to that time, the base-station will interrogate an outstation just before the SCADA system reads the data, so return to the SCADA system data which is has just undated. After that time the base-station will interrogate an outstation just ***after*** the SCADA system reads the data, so will return the data to it that it read on the previous outstation interrogation. It will thus return the same data to the SCADA system on two consecutive scans.

Conversely, if the ***Micro_Link*** clock is running slightly faster than the SCADA system clock, one reading received from the outstation will not be returned to the SCADA system.

This eventuality can be overcome by synchronising the Scanning Window to the SCADA system. This also provides an alternative method for the SCADA system to control the Scanning Window.

The relevant registers are as follows:

:Input Register Address	Output Register Address	Function
3001	3001	Scanning Interval
3002	3002	Scanning Window
3003		Time to next scan start
3004		Time to next window start
	3010	Reset scanning interval/window
3100		Scanning auto-sync

Reading registers 3001 or 3002 will return the current scanning interval and window length. Writing to 3001 and 3002 will change them.

Reading registers 3003 or 3004 will return the time to elapse before the start of the next scan cycle or window respectively (in seconds).

Writing to register 3010 will reset the scanning interval and window length to the values downloaded by DCD Terminal.

Periodic reads of register 3100 will set the scanning interval to be equal to the time between reads, provided they occur at regular intervals. If ***Micro_Link*** senses three consecutive reads of this register that are equally spaced in time it will modify the Scanning Interval to match and synchronise subsequent polls to them. The value returned will be the scanning interval.

Note that if the Scanning Window is decreased, any battery powered outstations will be asleep when the base-station next interrogates them, so a Comms Fail alarm will be generated on the first scan after the change is made. However, synchronisation will be restored after the expiry of the previous scanning window, and the Comms fail alarm will clear.

8.14 Test Modes

Configuring ***Micro_Link*** to certain station addresses sets special test modes (station addresses are set using the DCD configuration terminal described in section 9.1:

8.14.1 Address 250 - Test Transmitter

The radio transmitter (or the modem transmitter) is set permanently on, allowing the output level and modulation amplitude to be adjusted. The transmit frequency will be as set by DCD. In this mode the Test LED flashes in inverted sync with the Heartbeat LED.

8.14.2 Address 249 - Test Receiver/Transmitter

The function is this mode is set by the state of the 8 digital inputs on ***Micro_Link*** (NB Inputs are regarded as ‘1’ when a switch is closed, ‘0’ when it is open):

Inputs 1, 2, 3, 4 & 5 determine the radio channel frequency, in accordance with the following table.

Input 54321	**Channel Selected**	**Frequency (MHz)**
00000	0	458.5000
00001	1	458.5125
00010	2	458.5250
00011	3	458.5375
00100	4	458.5500
00101	5	458.5625
00110	6	458.5750
00111	7	458.5875
01000	8	458.6000
01001	9	458.6125
01010	10	458.6250
01011	11	458.6375
01100	12	458.6500
01101	13	458.6625
01110	14	458.6750
01111	15	458.6875
10000	16	458.7000
10001	17	458.7125
10010	18	458.7250
10011	19	458.7375
10100	20	458.7500
10101	21	458.7625
10110	22	458.7750
10111	23	458.7875
11000	24	458.8000
11001	25	458.8125
11010	26	458.8500
11011	27	458.8625
11100	28	458.8750
11101	29	458.8875
11110	30	458.9125
11111	31	458.9250

Note that this table lists the channels used in the UK, which start at 458.5000MHz. Other countries may allocate a different band, in which case an appropriate label will be fitted to ***Micro_Link***, and to the EPROM within it, identifying the start frequency.

Inputs 6, 7 & 8 define the test mode:

000 Test the transmitter, sending an unmodulated carrier

100 Test the transmitter, sending a constant mark state.

010 Test the transmitter, sending a constant space state..

110 Test the transmitter, sending alternate marks and spaces.

In each of the above modes the TEST LED flashes in opposite sync with the HEARTBEAT LED to indicate that the transmitter is active.

111 Test the radio/modem receiver. The receive signal strength (RSSI) is copied to analogue output 1 (if using a modem it will be 0mA if no carrier is present, or 20mA if a carrier is detected). The TEST LED flashes in sync with the HEARTBEAT LED if no carrier is detected, and in opposite sync if a valid carrier is present.

Note that of all inputs are off (or disconnected) ***Micro_Link*** transmits an unmodulated carrier on channel 0.

8.14.3 Address 248 – Calibrate Analogues / Test hardware outputs / Reset Counters

Setting this address will reset all pulse counters, then carry out an action defined by the state of the 8 digital inputs on ***Micro_Link*** (NB Inputs are regarded as ‘1’ when a switch is closed, ‘0’ when it is open). Note that calibration of analogues requires an accurate current source and an accurate ammeter, and must follow the procedure listed below. Calibration of RSSI is not possible without the use of a radio test set, so is normally only carried out in the factory.

12345678

10000001 Record analogue input reading for 20.00mA.

01000001 Record analogue input reading for 2.000mA, calculate correction factors and save in NOVRAM

00100001 Record value needed to give 20.00mA on analogue output 1

00010001 Record value needed to give 2.000mA on analogue output 1, calculate correction factors and save in NOVRAM

00001001 Record value needed to give 20.00mA on analogue output 2

00000101 Record value needed to give 2.000mA on analogue output 2, calculate correction factors and save in NOVRAM

10000011 Record RSSI reading for a signal level of +25dBµV

01000011 Record RSSI reading for a signal level of -15dBµV, calculate correction factors and save in NOVRAM

…THEN…

xxxxxx00 Sequence all digital outputs from all off to all on, and all analogue outputs in 20% increments

xxxxxx10 Sequence all digital outputs from all off to all on, and analogue output 2 in 20% increments. Copy corrected RSSI to analogue output 1. Flash TEST LED in opposite sync to HEARTBEAT if radio is receiving a valid carrier.

xxxxxx01 Copy digital inputs to digital outputs and uncorrected analogue inputs to analogue outputs. Flash TEST LED in opposite sync to HEARTBEAT if analogue or RSSI maximum value has been recorded.

xxxxxx11 Copy digital inputs to digital outputs and uncorrected analogue input 2 to analogue output 2. Copy uncorrected RSSI level to analogue output 1. Flash TEST LED in opposite sync to HEARTBEAT if analogue or RSSI maximum value has been recorded.

To calibrate analogues, the digital inputs should be set to 00000001, then changed to the required state, and returned to 00000001 before progressing to the next state.

To calibrate analogue inputs, apply a current of 20.00mA to both inputs in series, then select state 10000001. The TEST LED should flash in opposite sync to the HEARTBEAT to indicate that the reading has been recorded. Next, change the input current to 2.000mA, and then select state 01000001. The TEST LED should revert to flashing in sync with the HEARTBEAT to indicate that the calibration has been completed.

To calibrate analogue output 1, apply a current of 20mA to both inputs in series, and monitor the current on analogue output 1. Adjust the input current until the output reads 20.00mA, then select state 00100001. The TEST LED should flash in opposite sync to the HEARTBEAT to indicate that the reading has been recorded. Next, adjust the input current until the output reads 2.000mA, then select state 00010001. The TEST LED should revert to flashing in sync with the HEARTBEAT to indicate that the calibration has been completed.

To calibrate analogue output 2, repeat the above procedure, using test states 00001001 and 00000101.

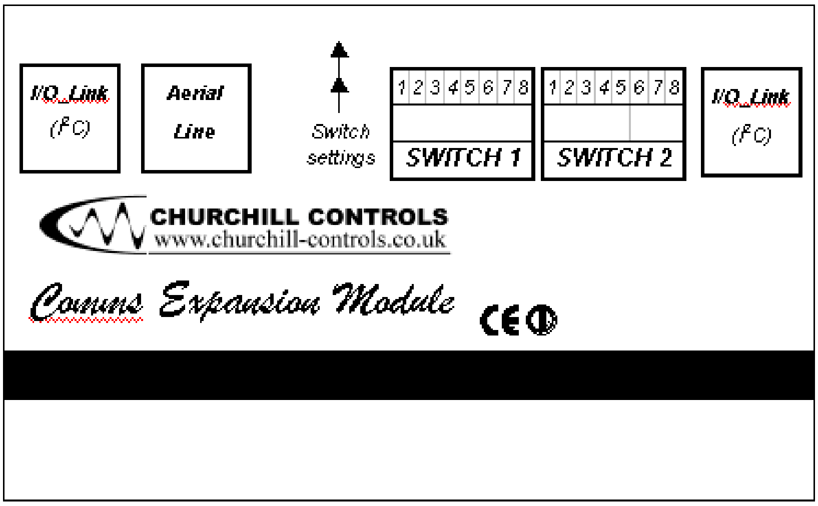
8.15 Dual Communications

Micro_Links of Software Version 2.30 or later can support dual communications.

This allows a ***Data_Link*** network to be set up working over two separate communications paths. Each path may be radio, leased line or private wire as previously described in Section 3.2.

This facility effectively allows the user to set up a dual-redundant comms system, which sends and receives data on both paths, allowing one path to fail and the data still get through.

In order to use this facility, the ***Micro_Link*** must be fitted with a Comms Expansion Module 7124, fitted with either a Radio or a Leased-Line / Private Wire Modem as required..



Switches S1 and S2 are used to set all configurable parameters on the Comms Expansion Module.

8.15.1 Switch S1 - Test

Switch S1 has no bearing in normal operation. However, if the Comms Expansion module is fitted with a radio, the requirement may arise – especially on factory test – to test the radio in isolation. There are therefore various test modes which can be set up as shown below. These switch settings must be avoided in normal operation.

Switch S1 **12345678**	**Decimal Value**	**Function**
01111111	254	Radio receiver test **(see Nano_Link manual: 9.3.2)**.
10111111	253	Radio transmitter test **(see Nano_Link manual: 9.3.3)**
00111111	252	Hardware I/O test **(see Nano_Link manual: 9.3.4)**

8.15.2 Switch S2 - Radio Versions – Set Channel

If the Comms. Expansion Module is equipped with an internal de-regulated radio, these switches set the operating frequency. The transmission baud rate is fixed at 1200 baud.

Note that the channels are at 12.5KHz spacing, but omit frequencies of 458.8250MHz, 458.8375MHz and 458.9000MHz, in accordance with MPT1329.

If the Comms Expansion Module is fitted with a Leased-Line modem, these switches are not used.

The switch settings are shown overleaf :

Channel Selected	**Switch 12345**	**Frequency (MHz)**	**Actual Channel No**
0	00000	458.5000	0
1	10000	458.5125	1
2	01000	458.5250	2
3	11000	458.5375	3
4	00100	458.5500	4
5	10100	458.5625	5
6	01100	458.5750	6
7	11100	458.5875	7
8	00010	458.6000	8
9	10010	458.6125	9
10	01010	458.6250	10
11	11010	458.6375	11
12	00110	458.6500	12
13	10110	458.6625	13
14	01110	458.6750	14
15	11110	458.6875	15
16	00001	458.7000	16
17	10001	458.7125	17
18	01001	458.7250	18
19	11001	458.7375	19
20	00101	458.7500	20
21	10101	458.7625	21
22	01101	458.7750	22
23	11101	458.7875	23
24	00011	458.8000	24
25	10011	458.8125	25
26	01011	458.8500	28
27	11011	458.8625	29
28	00111	458.8750	30
29	10111	458.8875	31
30	01111	458.9125	33
31	11111	458.9250	34

8.15.3 Use with a Basestation

A *Micro_Link* Basestation fitted with a Comms. Expansion Module will poll Outstations for data in the normal way – except that it will send the poll messages on both Communications paths (i.e. its using internal comms. Device and using the comms expansion module). It will then ‘listen’ for the outstation to reply on either of the two paths, and will accept and act on data from either.

If it only receives a reply on one paths it will raise a ‘Dual Comms Fail’ alarm. If the path that has failed is the comms expansion module, it will also raise an ‘Expansion Comms Fail’ Alarm.

If no reply is received on either path it will raise the normal ‘Comms Fail’ alarm.

These may be mapped to digital outputs or read via Bus_Link in the normal way.

8.15.4 Use with an Outstation

A *Micro_Link* Outstation fitted with a Comms. Expansion Module listens for poll requests from the outstations on both paths. It will accept and act on data from either, and send its reply on both paths.

8.15.5 Use with Repeaters

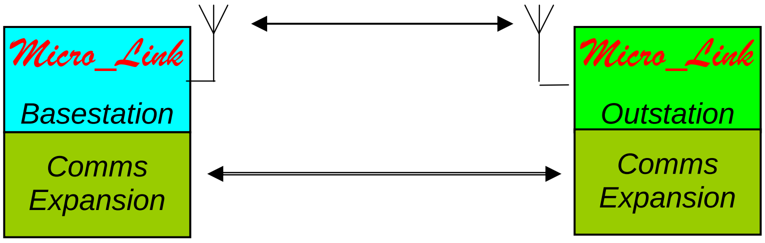
The Dual Comms facility can also be used with repeaters – or as a repeater. In this way, similar to above, a Dual Comms repeater will listen on both paths, and repeat onwards on both paths any message it receives. This could also be used as a ‘Comms Converter’ converting radios message to leased-line, or vice versa.

8.15.6 Sample Configurations

A number of sample Dual Comms Configurations are shown below. These are by no means comprehensive, but are intended to illustrate some of the applications for Dual Comms.

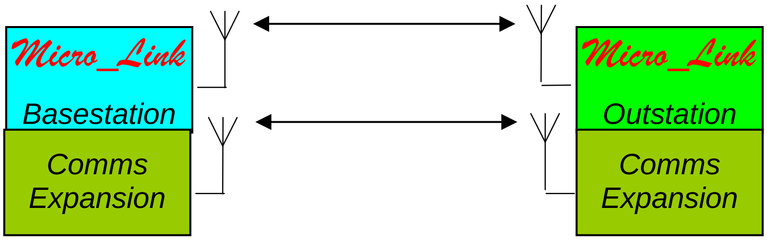
8.15.6.1 Point to Point – Radio & Leased Line

This shows a Dual Comms system as a simple point-to-point link with internal radios and leased line expansion.



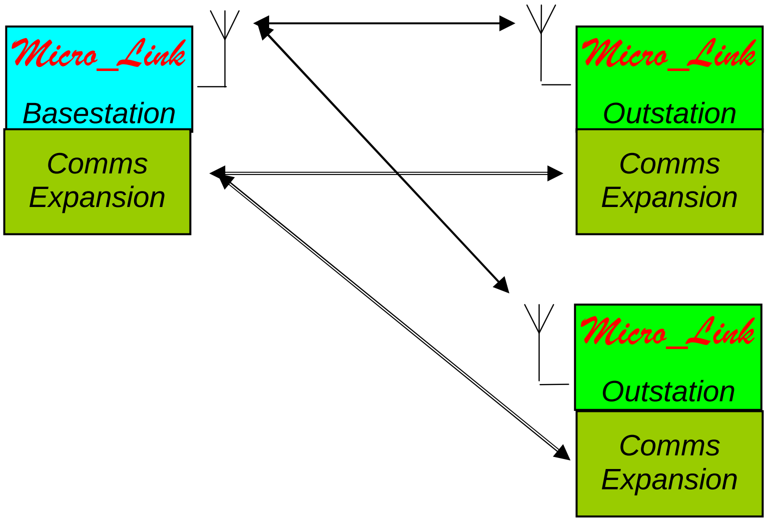
8.15.6.2 Point to Point – Radio & Radio

This shows a Dual Comms system as a simple point-to-point link with internal radios and radio expansion. This could be used if there were suspected problems with interference on one radio channel. Setting the expansion module to a different radio channel would allow a dual frequency system to be set up.



8.15.6.3 Multiple Dual Comms Outstations

This shows a Point-to-Many system using Dual Comms.



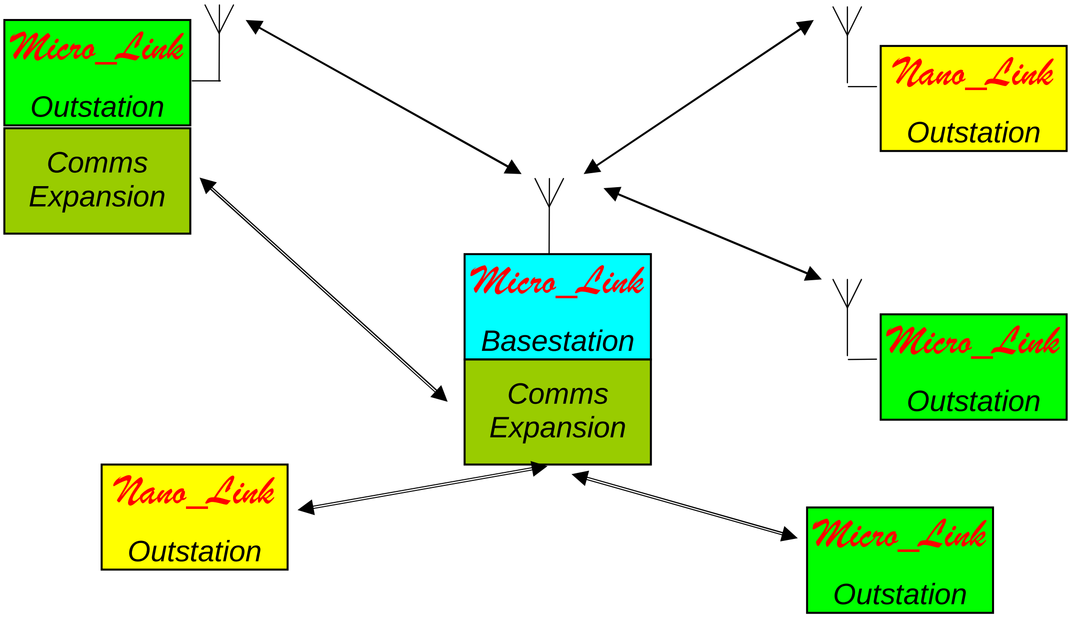
8.15.6.4 Single and Dual Comms Outstations

This shows a system using a Dual Comms Basestation, and a variety of outstation types – leased line *Micro_Links*, radio *Micro_Links*, Dual Comms *Micro_Links* and *Nano_Links*.

The Basestation will poll round the outstations – issuing poll commands on each comms. path.

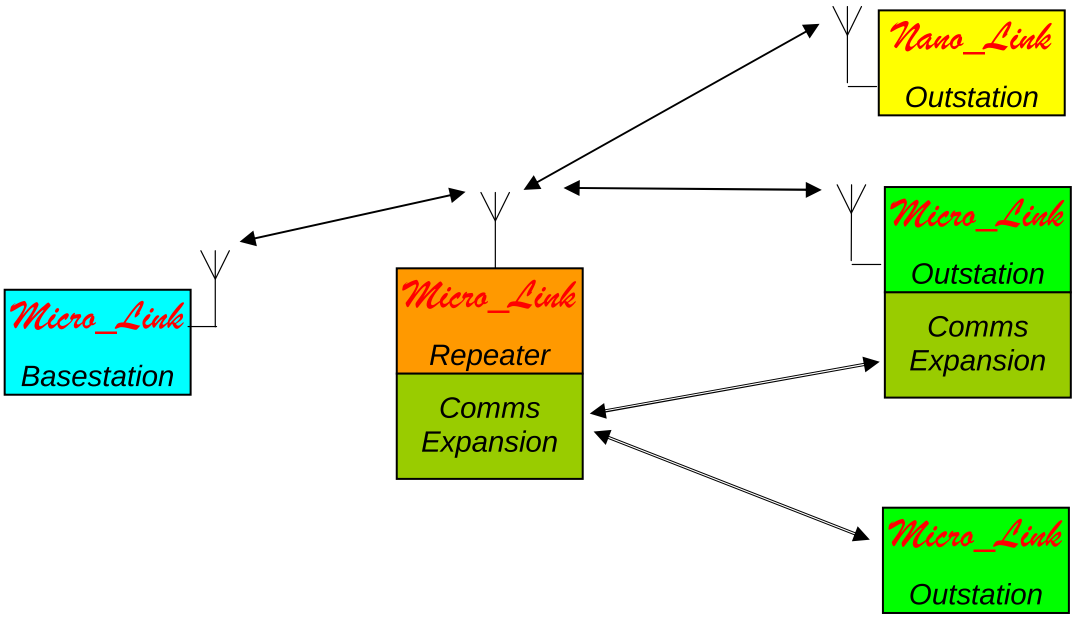
The outstations will then reply on whichever comms. paths they are fitted with.

This could be used to augment a radio system with leased-line outstations in areas of poor radio coverage.



8.15.6.5 Dual Comms Repeater

This shows a system where a Dual Comms repeater is used to repeat signals from a radio basestation over both radio and leased line. Again, this could be used to allow a radio basestation to poll outstations in areas of little or no radio coverage, by converting the path to leased line at the repeater. This need not however affect radio outstations already on the system.



9. DCD Terminal

DCD Terminal is used to configure and diagnose problems on ***Micro_Link***. It comprises two software packages running on a PC, DCD Configuration and DCD Diagnostics. It is supplied on the Distribution CD included with all *Micro_Links*, and is compatible with Windows 95 and all subsequent versions. To install the software navigate to the DCD folder on the disc, run the Setup.exe file and follow the instructions.

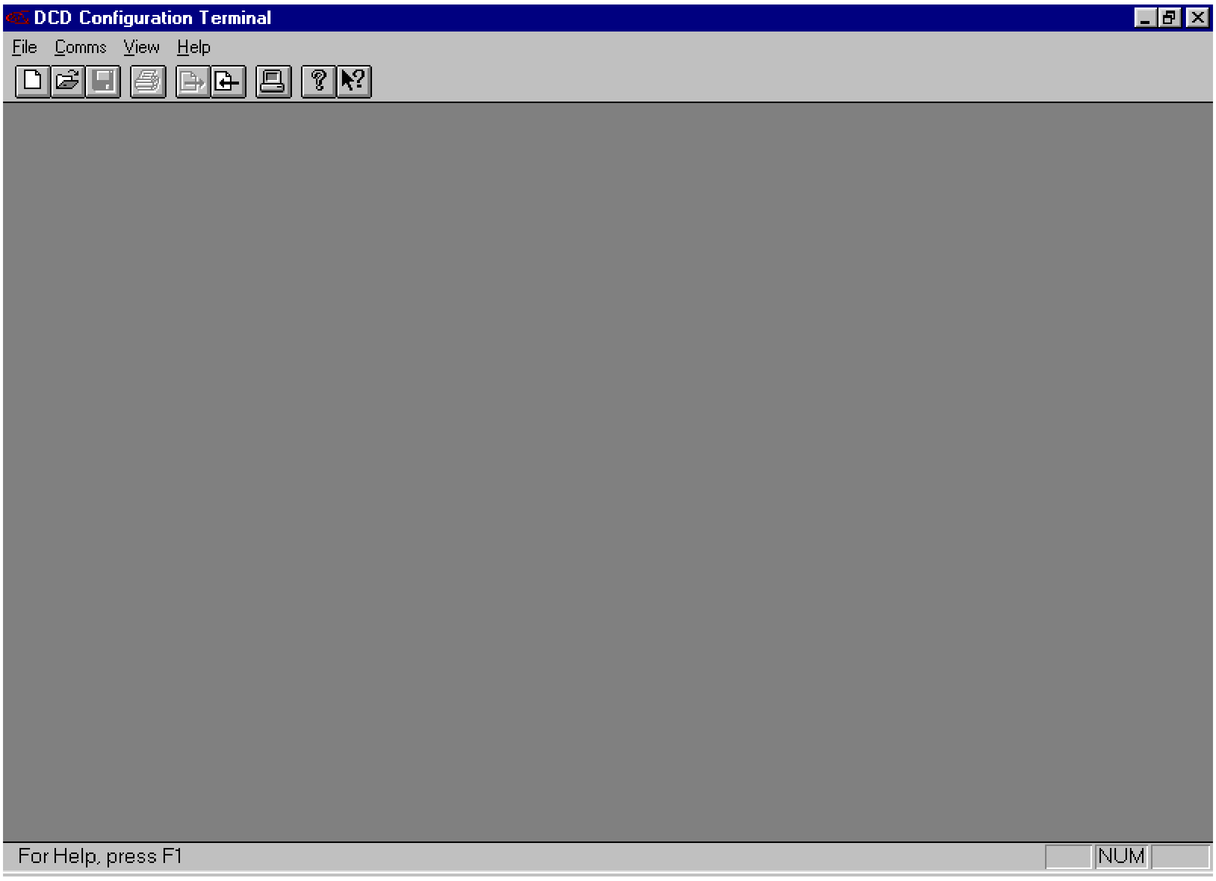
The installation process will create a programme group containing an icon named DCD Configuration, and one named Readme. The latter contains hints and details of any changes that may have been made after the documentation was printed.

Connect a spare COM port from the PC to the DCD port on ***Micro_Link*** using a Churchill Controls Ltd adapter type 7039/2 (which converts from RJ45 to a 9-pin D socket and also provides level conversion from TTL to RS232) and a parallel RJ45 cable. Note that all applications assume that COM1 is used. If this port is not available, any other serial port can be used, but DCD Configuration and DCD Diagnostics will need to be configured accordingly. The means of doing this should be self-explanatory, but assistance is available through the HELP menu.

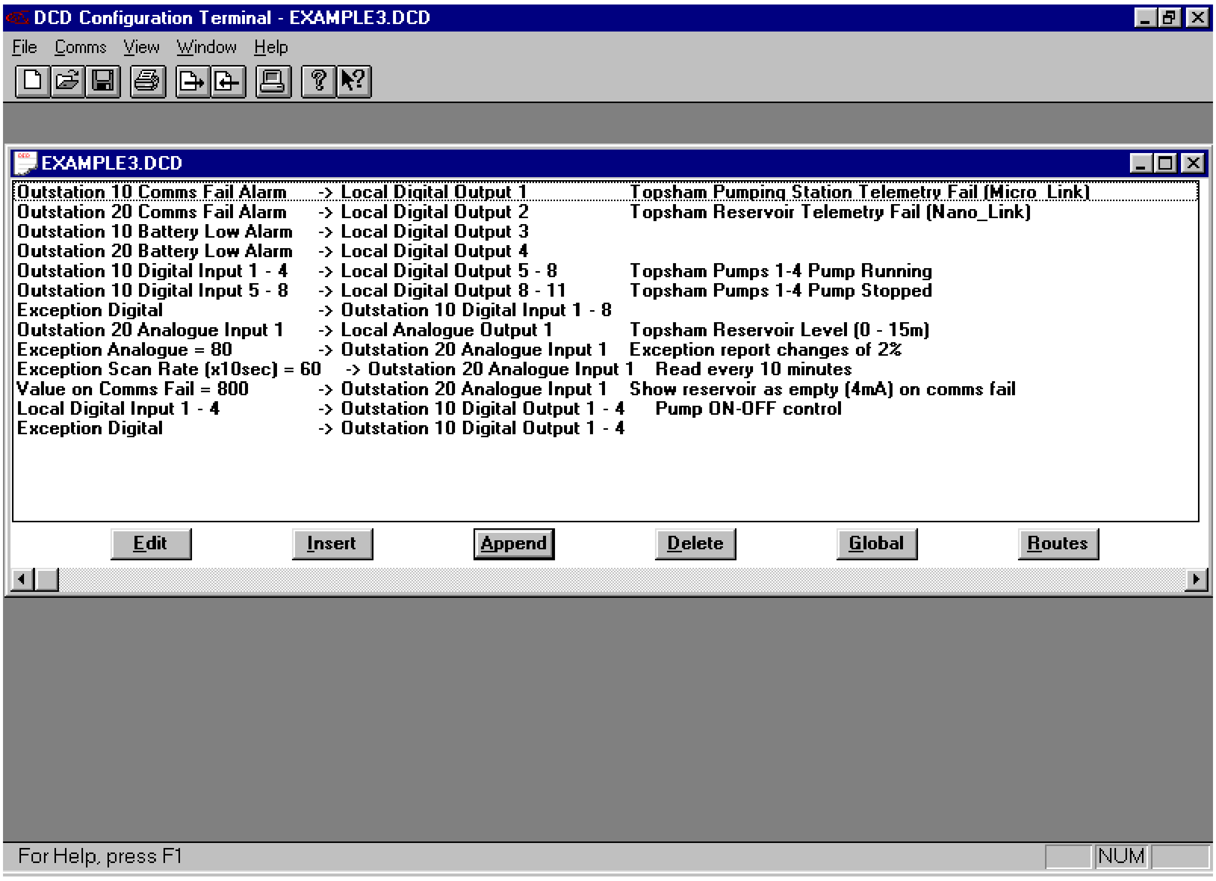
Note that a DCD Configuration can be run without connection to ***Micro_Link***, allowing configurations to be created and saved to disc before going to site.

9.1 DCD Configuration

Double-click on the DCD icon and the following window should open:



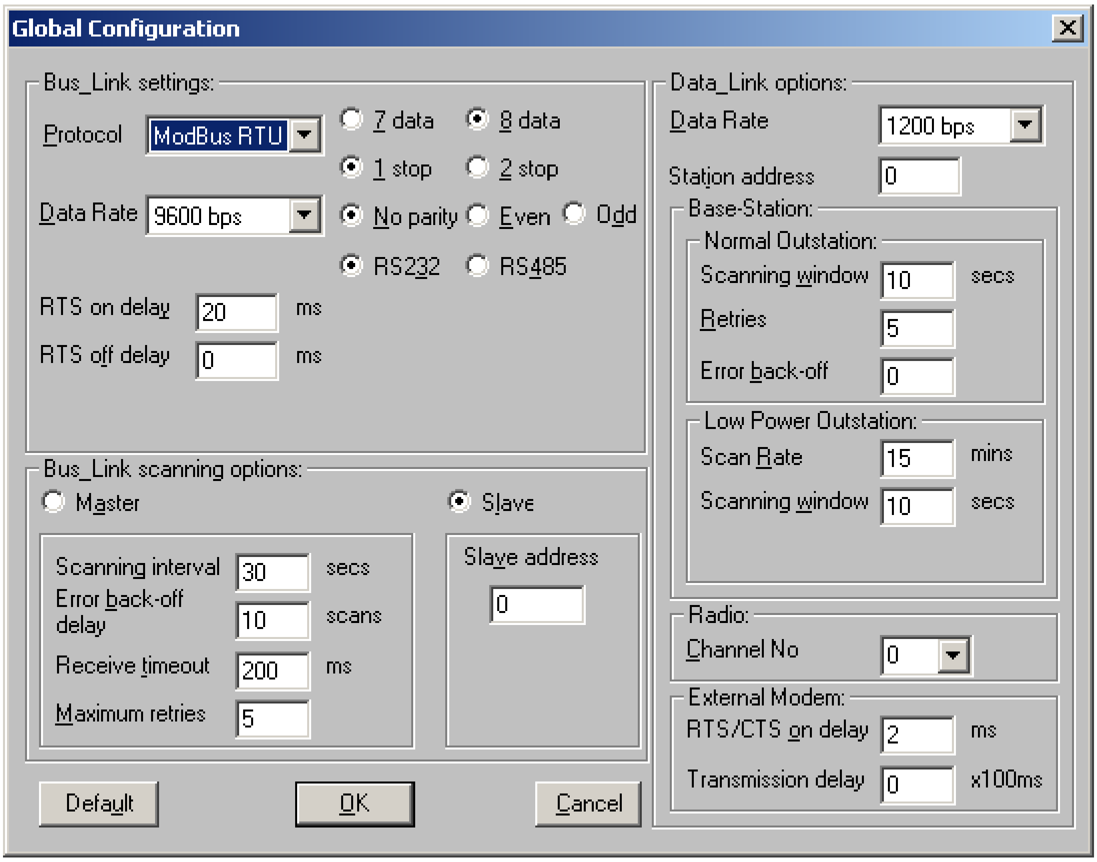
Select File/Open, and then select file Example3 from the installation disc. The screen will now show:



```
Outstation 10 Comms Fail Alarm     -> Local Digital Output 1         Topsham Pumping Station Telemetry Fail (Micro_Link)
Outstation 20 Comms Fail Alarm     -> Local Digital Output 2         Topsham Reservoir Telemetry Fail (Nano_Link)
Outstation 10 Battery Low Alarm    -> Local Digital Output 3
Outstation 20 Battery Low Alarm    -> Local Digital Output 4
Outstation 10 Digital Input 1 - 4  -> Local Digital Output 5 - 8     Topsham Pumps 1-4 Pump Running
Outstation 10 Digital Input 5 - 8  -> Local Digital Output 8 - 11    Topsham Pumps 1-4 Pump Stopped
Exception Digital                  -> Outstation 10 Digital Input 1 - 8
Outstation 20 Analogue Input 1     -> Local Analogue Output 1        Topsham Reservoir Level (0 - 15m)
Exception Analogue = 80            -> Outstation 20 Analogue Input 1   Exception report changes of 2%
Exception Scan Rate (x10sec) = 60    -> Outstation 20 Analogue Input 1   Read every 10 minutes
Value on Comms Fail = 800          -> Outstation 20 Analogue Input 1   Show reservoir as empty (4mA) on comms fail
Local Digital Input 1 - 4          -> Outstation 10 Digital Output 1 - 4     Pump ON-OFF control
Exception Digital                  -> Outstation 10 Digital Output 1 - 4
```

The opening window shows the Data Routing Table. However, before describing this, the Global Data must be defined. Click on the Global button:

9.1.1 Global Configuration



Global Configuration

Bus_Link settings:

Protocol: ModBus RTU

○ 7 data ● 8 data

● 1 stop ○ 2 stop

Data Rate: 9600 bps

● No parity ○ Even ○ Odd

● RS232 ○ RS485

RTS on delay: 20 ms

RTS off delay: 0 ms

Bus_Link scanning options:

○ Master ● Slave

Scanning interval: 30 secs

Error back-off delay: 10 scans

Receive timeout: 200 ms

Maximum retries: 5

Slave address: 0

Data_Link options:

Data Rate: 1200 bps

Station address: 0

Base-Station:

Normal Outstation:

Scanning window: 10 secs

Retries: 5

Error back-off: 0

Low Power Outstation:

Scan Rate: 15 mins

Scanning window: 10 secs

Radio:

Channel No: 0

External Modem:

RTS/CTS on delay: 2 ms

Transmission delay: 0 x100ms

This window is divided into three parts:

9.1.1.1 Bus_Link Settings

This part is only relevant if the *Bus_Link* interface is used:

Protocol must obviously match that of the device(s) to which the ***Bus_Link*** port is connected, as must the serial data parameters.

RS232/RS485 radio buttons must be selected to match the interface level (see section 3.7.1).

RTS on delay defines the time the DSR control line will be raised before sending a message. It allows for slow transceivers such as leased line modems. If set to 0 DSR is kept permanently active

RTS off delay sets the time DSR remains active after the end of the message.

NB. Word length and baud rate must also be configured to be compatible with the PLC with which the unit is communicating.

9.1.1.2 Bus_Link Options

This part is also only relevant if the *Bus_Link* interface is used:

Master/Slave radio buttons define the fundamental operation of the ***Bus_Link*** interface. In slave mode ***Bus_Link*** responds to commands from an external master, provided the address matches that in the Slave Address dialogue box.

Scanning interval sets the rate at which ***Bus_Link*** scans through the data routing table. If it is set to 0 the scan is continuous. Otherwise a timer is started at the beginning of each scan. At the end of each scan ***Bus_Link*** waits until the timer has lapsed before starting the next scan.

Receive time-out defines the maximum time ***Bus_Link*** will wait for a response to a command it has issued. If no reply has been received within this time it will log the fact and continue. It should typically be set to 200ms.

Maximum retries defines how many consecutive unsuccessful attempts are allowed for a given command before an error is flagged.

Error back-off delay defines how many scans of a faulty slave device will be skipped before repeating the retry sequence. If more than one slave is being polled, this prevents communication with the remaining slaves from being unnecessarily delayed. If only one slave is being polled, set this to 0.

Slave address only applies if the Slave radio button has been selected. It defines the address to which ***Bus_Link*** will respond to an external master.

9.1.1.3 Data_Link Options

This area configures the ***Data_Link 2000*** communications network:

Data Rate sets the data transmission rate, which is normally 1200 baud. If using leased line modems or external comms devices, set the baud rate at which data is passed over the network. All stations on a network must be set to the same speed.

Station address determines the root data block allocated to hardware I/O. If the I/O capacity exceeds that of a data block it will 'spill over' into the next higher station address as well. Address 0 has the specific meaning of defining ***Micro_Link*** as a base-station.

The parameters in the **Base-station** box apply only if the Station Address is set to 0, with the exception of **Normal Outstation: Scanning Window**, which also has relevance when configuring an outstation:

Normal Outstation: Scanning Window:
If configuring a base-station, this defines the frequency at which it will interrogate outstations other than battery-powered ***Nano_Link***. If it is set to 0 polling will be continuous. See section 6.7 for an explanation of the relevance of this.

If configuring an outstation the comms fail alarm will be raise 2 * (Normal Outstation: Scanning Window) seconds after the outstation expects to be polled. It therefore determines the window allowed for retries before raising an alarm. The value should typically be set to match the Normal Outstation: Scanning Window configured for the base-station, but it can be adjusted if required.

Normal Outstation: Retries defines the number of attempts the base-station will make to establish communications with each outstation. The last retry will be made using an extended lead-in (i.e. switching on the transmitter for 2 seconds before sending the command) in case it is a ***Nano_Link*** outstation which has reverted to sniff mode. If all attempts are unsuccessful the base-station will flag an outstation communications alarm and wait for the expiry of the Scanning Window before proceeding to the next outstation.

Normal Outstation: Error back-off defines the number of scans that will be skipped following a communications failure if more than one outstation is being polled. This prevents communication with the remaining outstations from being unnecessarily delayed.

Low Power Outstation: Scan Rate defines the overall scan rate for battery-powered ***Nano_Link*** outstations. The base-station passes this value to the outstations, so they can put themselves to sleep until the next poll is due.

Low Power Outstation: Scanning Window defines the time for which the base-station will try to communicate with any outstation that fails to respond. At the end of this period is will raise a communications fail alarm for the outstation and proceed to the next outstation. The value must be greater than 8 seconds and less than (*Scan Rate / n*), where *n* is the number of battery-powered outstations on the network. If it is set too long the base-station will automatically adjust the window size at the end of the first scan.

Low Power Outstation: Normal attempts defines the number of attempts the base-station will make at communicating with an outstation that fails to respond before using an extended lead-in (i.e. switching on the transmitter for 2 seconds before sending the command) in case the outstation has reverted to sniff mode. After each normal attempt the base-station waits for about 2 seconds before retrying, so *Normal attempts* should be less than (*Scanning Window / 2*) to ensure there is time for at least one extended retry.

Radio Channel No is only applicable if ***Micro_Link*** is fitted with a de-regulated radio, and defines the frequency at which the radio will operate. When using a standard MPT1329 UHF radio the operating frequency is as defined in the following table:

Channel Selected	Frequency (MHz)	Actual Channel No
0	458.5000	0
1	458.5125	1
2	458.5250	2
3	458.5375	3
4	458.5500	4
5	458.5625	5
6	458.5750	6
7	458.5875	7
8	458.6000	8
9	458.6125	9
10	458.6250	10
11	458.6375	11
12	458.6500	12
13	458.6625	13
14	458.6750	14
15	458.6875	15
16	458.7000	16
17	458.7125	17
18	458.7250	18
19	458.7375	19
20	458.7500	20
21	458.7625	21
22	458.7750	22
23	458.7875	23
24	458.8000	24
25	458.8125	25
26	458.8500	28
27	458.8625	29
28	458.8750	30
29	458.8875	31
30	458.9125	33
31	458.9250	34

Note that the channels are at 12.5KHz spacing, but omit frequencies of 458.8250MHz, 458.8375MHz and 458.9000MHz, in accordance with MPT1329.

Note that this table lists the channels used in the UK, which start at 458.5000MHz. Other countries may allocate a different band, in which case an appropriate label will be fitted to ***Micro_Link***, and to the EPROM within it, identifying the start frequency.

Any of these channels can be used, but all stations on a given network must be set to the same channel.

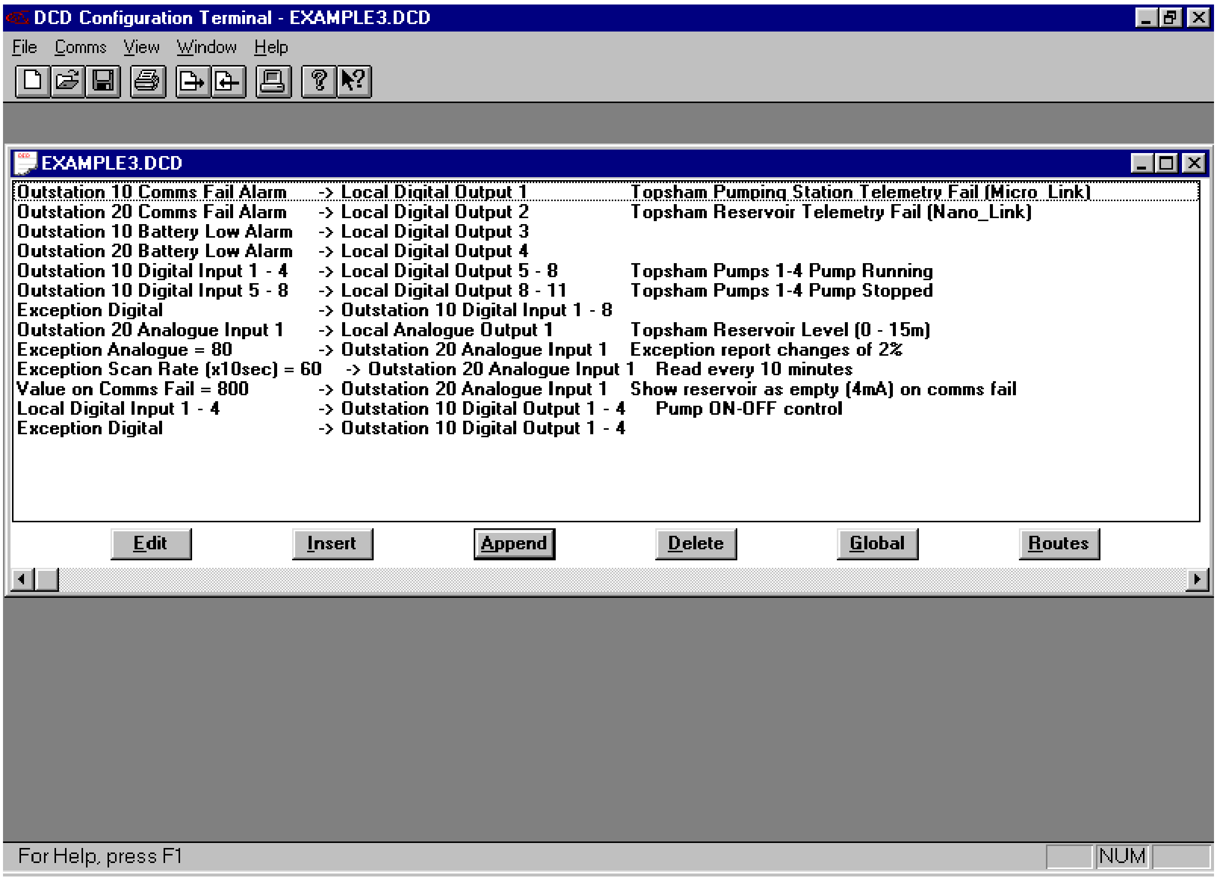
RTS on delay is only relevant if ***Micro_Link*** does not use an internal radio, and defines the time delay between raising RTS to enable the transmitter and starting to send data (see also Normal Attempts). The minimum acceptable value is 20ms, which is appropriate for most leased line modems. Any value less than this will be interpreted as 20ms.

RTS off delay is only relevant if ***Micro_Link*** does not use an internal radio, and defines the time delay between sending the last character of a message and dropping RTS, to allow for slow comms devices. It will normally be set to zero.

Press **OK** to return to the Data Routing Table:

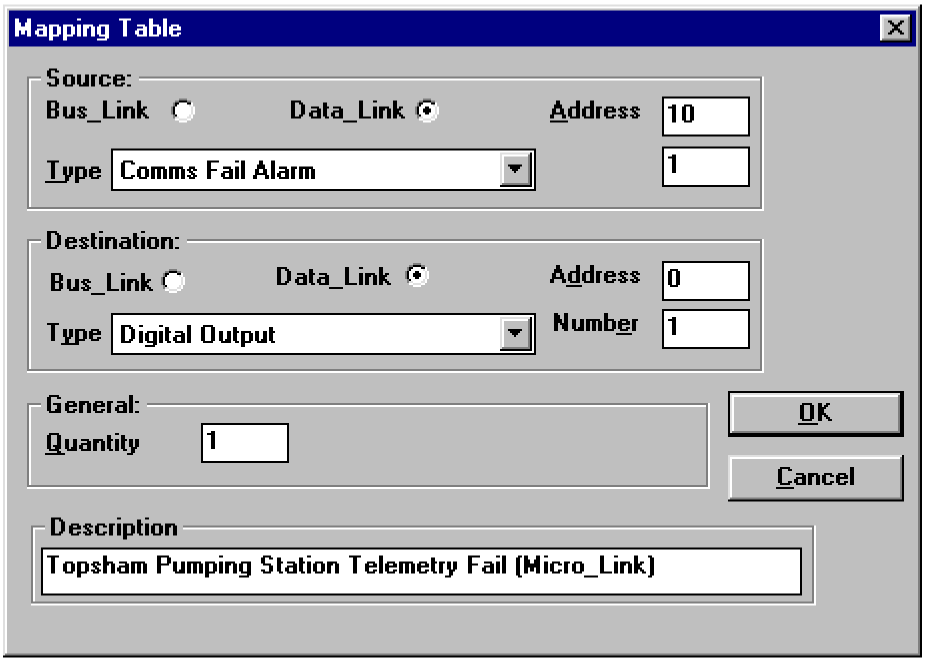
9.1.2 Data Routing Table

The file Example3 has the following Data Routing Table:


DCD Configuration Terminal - EXAMPLE3.DCD
File Comms View Window Help
EXAMPLE3.DCD

```
Outstation 10 Comms Fail Alarm     -> Local Digital Output 1          Topsham Pumping Station Telemetry Fail (Micro_Link)
Outstation 20 Comms Fail Alarm     -> Local Digital Output 2          Topsham Reservoir Telemetry Fail (Nano_Link)
Outstation 10 Battery Low Alarm    -> Local Digital Output 3
Outstation 20 Battery Low Alarm    -> Local Digital Output 4
Outstation 10 Digital Input 1 - 4  -> Local Digital Output 5 - 8      Topsham Pumps 1-4 Pump Running
Outstation 10 Digital Input 5 - 8  -> Local Digital Output 8 - 11     Topsham Pumps 1-4 Pump Stopped
Exception Digital                  -> Outstation 10 Digital Input 1 - 8
Outstation 20 Analogue Input 1     -> Local Analogue Output 1         Topsham Reservoir Level (0 - 15m)
Exception Analogue = 80            -> Outstation 20 Analogue Input 1  Exception report changes of 2%
Exception Scan Rate (x10sec) = 60    -> Outstation 20 Analogue Input 1  Read every 10 minutes
Value on Comms Fail = 800          -> Outstation 20 Analogue Input 1  Show reservoir as empty (4mA) on comms fail
Local Digital Input 1 - 4          -> Outstation 10 Digital Output 1 - 4   Pump ON-OFF control
Exception Digital                  -> Outstation 10 Digital Output 1 - 4
```

Select the first line of the table and click on Edit (or double-click). The following dialogue box will open:



Source:
Bus_Link ○ Data_Link ● Address 10
Type Comms Fail Alarm 1

Destination:
Bus_Link ○ Data_Link ● Address 0
Type Digital Output Number 1

General:
Quantity 1

Description
Topsham Pumping Station Telemetry Fail (Micro_Link)

If the Source is selected as ***Data_Link***, then the Address identifies the address of the outstation from which the data originates (Address 0 = base-station). If Address matches the Station Address set in the Global dialogue box then it will be interpreted as Local.

If the Source is selected as ***Bus_Link*** then it is assumed that Master mode has been selected from the global ***Bus_Link*** scanning options. Address then defines the address of the PLC or other device from which ***Micro_Link*** is to request the data.

The Source Type allows the user to select from a drop-down list. The items at the beginning of the list are prefixed with '*', and define absolute I/O addresses (e.g. *Digital Input Register), as required by devices accessed over the ***Bus_Link*** port. All other items either map into the I/O of ***Micro_Link*** and ***Nano_Link***, or are configuration parameters.

If the Source Type is multiple I/O the adjacent box is labelled **Number** and allows the user to define the first point (e.g. Digital Input 1 is the first digital input on ***Micro_Link*** or ***Nano_Link***).

If the Source Type is a configuration parameter that requires a value, the adjacent box is labelled **Value**.

If the Source Type is a single I/O point (e.g. Comms Fail Alarm), or a configuration parameter that does not require a value (e.g. Invert Digital) the label on the adjacent box is blank, and any value entered in it is ignored.

The Destination box functions are identical to the Source box.

The **Quantity** allows successive points to be configured. For example, if the Source is Digital Input Number 1 the Destination is Digital Output Number 10 and the Quantity is 10, then Digital Inputs 1…10 will be copied to Digital Outputs 10…19.

The **Description** box allows free-format text to be added to each line entry. This text is included in any printed copies, and saved to disc as part of the configuration file. However, it is not downloaded to ***Micro_Link***, so cannot be restored if the configuration is subsequently uploaded from ***Micro_Link***.

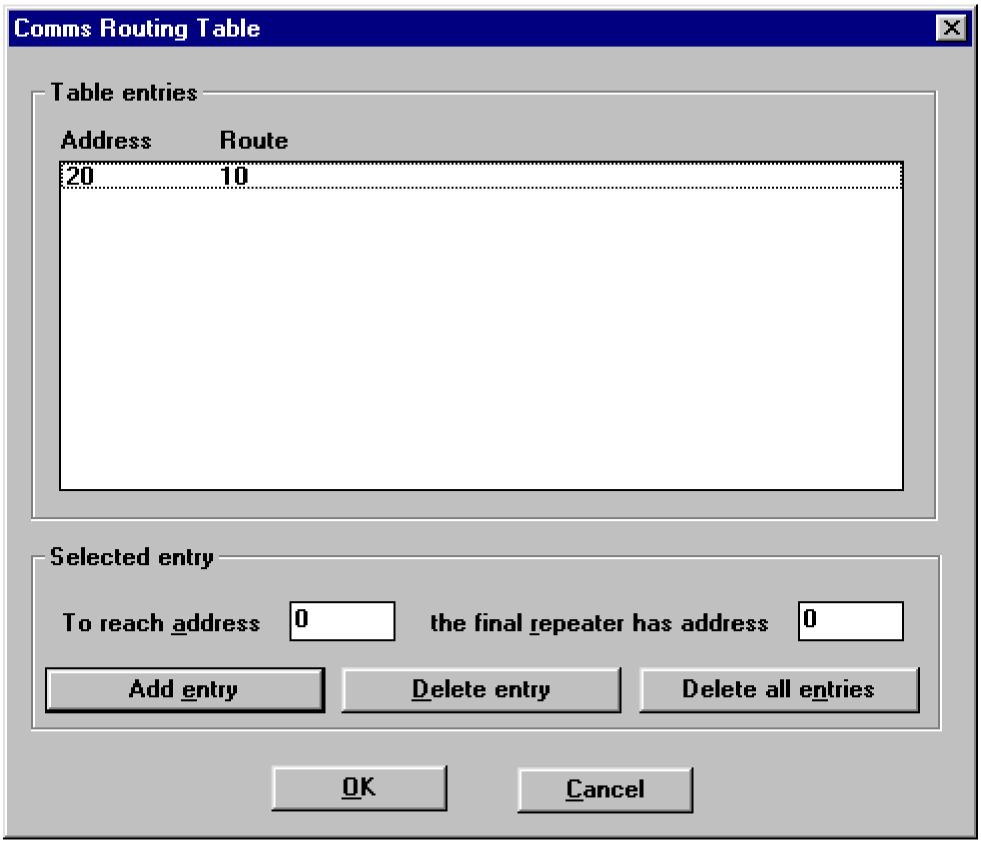
Hardware I/O numbering starts from the ***Micro_Link*** module, and continues contiguously onto expansion modules. For example, analogue outputs 1 and 2 are on ***Micro_Link***, with outputs 3, 4, 5 & 6 on the first analogue output expansion module, 7, 8, 9 & 10 on the next, an so on.

Mapping data between the ***Micro_Link*** database and ***Bus_Link*** is only needed if ***Bus_Link*** is configured in master mode within the Global dialogue box. If it is configured as a slave then the host device will be programmed to define the required registers within ***Micro_Link***. If it is configured as a master, the user must define both the slave address of the device to which data is being transferred (in the Address field) and the address of the register within the slave (in the Number field). Absolute addresses should be used to access registers via ***Bus_Link*** (using one of the data Types with a * prefix).

The system configuration defined in this example should be obvious. Note that the table has been constructed so that destinations (i.e. the right hand column) are in logical sequence. This is preferred, since it is immediately apparent if an attempt is made to write data from different sources to the same destination.

Now click on the Routes button to edit the Network Routing Table:

9.1.3 Network Routing Table



Address	Route
20	10

To reach address: 0

the final repeater has address: 0

This shows the communications route to reach each outstation. The example shows that outstation 10 can be reached directly by the base-station (by default, because there is no entry for it). Outstation 20 is reached by using outstation 10 as a repeater. Try entering To reach address **30** the final repeater is **20**, and note that the route to 30 is specified as ’20 10’.

9.1.4 Saving Configuration Files to Disc

The configuration file can be saved to disc at any time by selecting File/Save or File/Save As. It is good practice to use meaningful file names that relate to the relevant station.

9.1.5 Downloading and Uploading Station Configurations

Before attempting to communicate with ***Bus_Link***, check that the serial port configuration is correct by selecting Comms/Setup.... Check that the correct PC serial port is selected, and it is configured for 9600 bps, no parity, 1 stop bit. Any changes that are made will be automatically saved to disc for future use.

To download a new or modified configuration to ***Micro_Link***, select Comms/Download Configuration (or use the shortcut key combination of ctrl+D or click on the 🗎→ button). While downloading, ***Micro_Link*** ceases its normal operation, stores the new configuration in non-volatile memory then restarts using the new configuration.

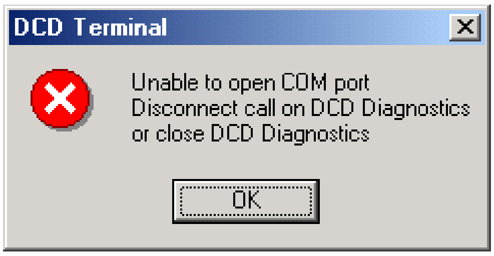
To upload the configuration from Bus_Link, select Comms/Upload configuration (or press ctrl+U or click the 🗎← button). A new window will be opened with a default name of ‘DCD1’, containing the configuration currently running on ***Micro_Link***. This can be edited and/or saved to disc if required.

9.2 DCD Diagnostics

9.2.1 General

DCD Diagnostics incorporates a terminal emulator which must be invoked to run diagnostics, by clicking on the terminal button (below Help on the menu). This should open a new window titled **DCD Diagnostics** followed by the comm port number.

DCD Diagnostics operates independently from ***DCD Configuration***, with the proviso that due to limitations inherent within the terminal emulator, If an attempt is made to upload or download a configuration via ***DCD Configuration*** while ***DCD Diagnostics*** is running a dialog box like this will open.



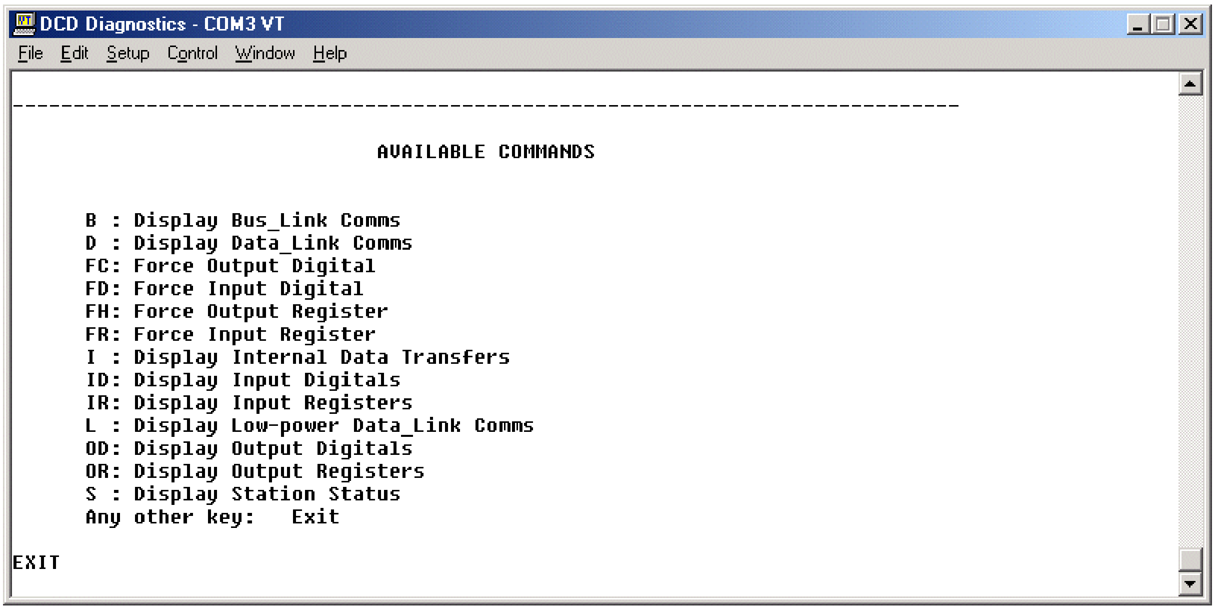
Unable to open COM port
Disconnect call on DCD Diagnostics
or close DCD Diagnostics

DCD Diagnostics distinguishes diagnostic commands from configuration commands entered via ***DCD Configuration*** by the fact that they are all one or two characters followed by enter (↵), and all keys are pressed within two seconds of each other. If these criteria are not met ***DCD Diagnostics*** will display TIME-OUT.

Displays which are constantly updating (e.g. 'S - Display station status') can be terminated by pressing enter. This also applies to commands that take a long time to display all their data (e.g. 'OD - Display output digitals').

9.2.2 Listing available commands

Enter **'?'** followed by **enter** (↵) for a list of available commands:


DCD Diagnostics - COM3 VT
File Edit Setup Control Window Help

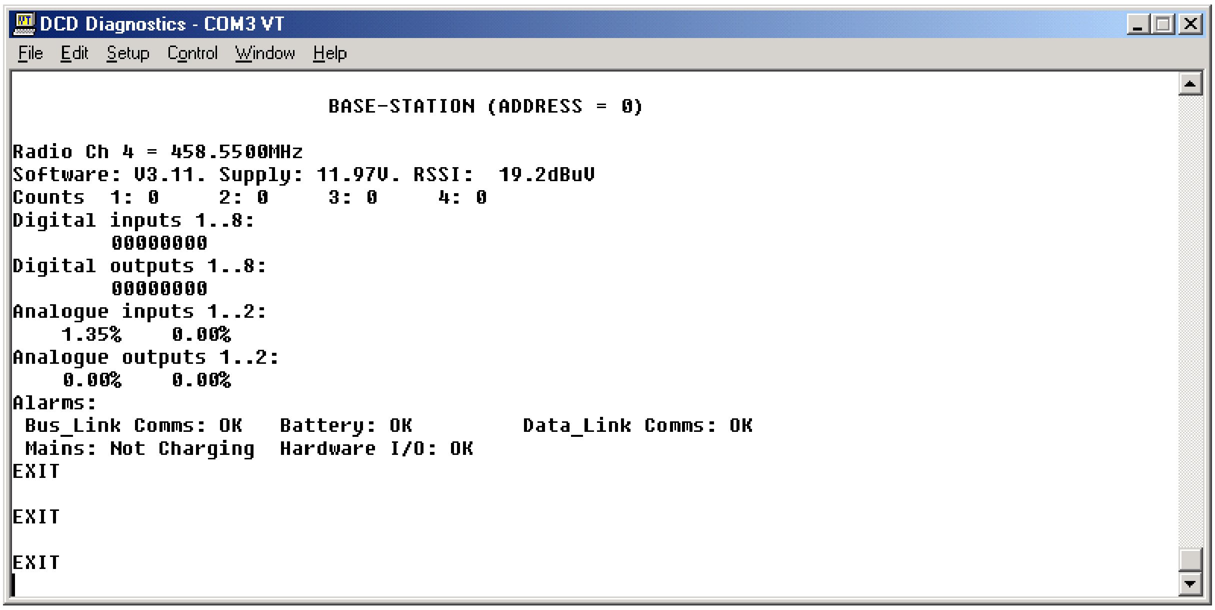
```
-------------------------------------------------------------------------------

                              AVAILABLE COMMANDS


      B : Display Bus_Link Comms
      D : Display Data_Link Comms
      FC: Force Output Digital
      FD: Force Input Digital
      FH: Force Output Register
      FR: Force Input Register
      I : Display Internal Data Transfers
      ID: Display Input Digitals
      IR: Display Input Registers
      L : Display Low-power Data_Link Comms
      OD: Display Output Digitals
      OR: Display Output Registers
      S : Display Station Status
      Any other key:   Exit

EXIT
```

Note that all commands must be terminated by the enter (↵) key, and that any one or two key command other than those listed exits the current mode. The screen will display the text EXIT as shown above.

Data that has scrolled off the top or bottom of the screen can be viewed by dragging the scroll bar.

The following sections describe some of the command functions, approximately in order of relevance:

9.2.3 S - Display Station Status

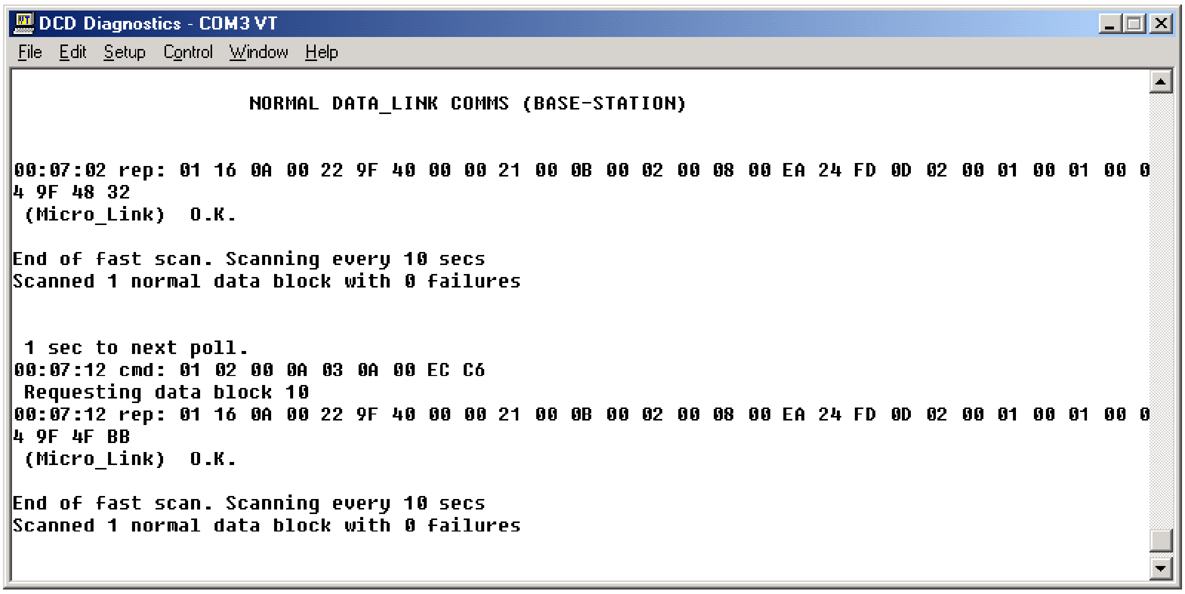


DCD Diagnostics - COM3 VT
File Edit Setup Control Window Help

```
                          BASE-STATION (ADDRESS = 0)

Radio Ch 4 = 458.5500MHz
Software: V3.11. Supply: 11.97V. RSSI:  19.2dBuV
Counts   1: 0     2: 0     3: 0     4: 0
Digital inputs 1..8:
        00000000
Digital outputs 1..8:
        00000000
Analogue inputs 1..2:
    1.35%    0.00%
Analogue outputs 1..2:
    0.00%    0.00%
Alarms:
 Bus_Link Comms: OK   Battery: OK         Data_Link Comms: OK
 Mains: Not Charging  Hardware I/O: OK
EXIT

EXIT

EXIT
```

This reads and displays the station address and comms configuration followed by the states of all plant I/O. The I/O capacity displayed includes any expansion modules that ***Micro_Link*** has detected in addition to its own I/O.

9.2.4 D – Display Normal Data_Link comms



DCD Diagnostics - COM3 VT
File Edit Setup Control Window Help

```
                    NORMAL DATA_LINK COMMS (BASE-STATION)


00:07:02 rep: 01 16 0A 00 22 9F 40 00 00 21 00 0B 00 02 00 08 00 EA 24 FD 0D 02 00 01 00 01 00 0
4 9F 48 32
 (Micro_Link)  O.K.

End of fast scan. Scanning every 10 secs
Scanned 1 normal data block with 0 failures


 1 sec to next poll.
00:07:12 cmd: 01 02 00 0A 03 0A 00 EC C6
 Requesting data block 10
00:07:12 rep: 01 16 0A 00 22 9F 40 00 00 21 00 0B 00 02 00 08 00 EA 24 FD 0D 02 00 01 00 01 00 0
4 9F 4F BB
 (Micro_Link)  O.K.

End of fast scan. Scanning every 10 secs
Scanned 1 normal data block with 0 failures
```

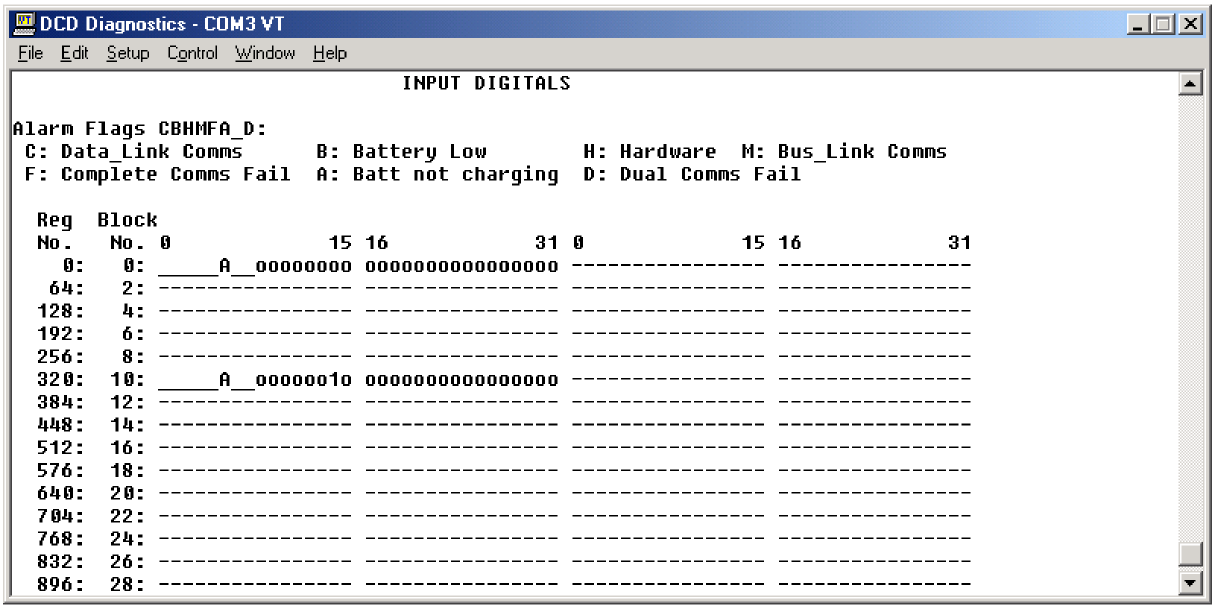
This display shows the communication between the base-station and the outstation(s). It normally shows all communications. However, if it is run on a base-station that is communicating with a combination of continuously powered outstations and battery-powered outstations, then it will only show communications with the continuously powered outstations.

Note that each line is prefixed by a time stamp. The absolute time is not relevant, but the time stamp can be useful for indicating the relative time between commands.

9.2.5 L – Display Low-power Data_Link comms

This is similar to the 'D' command, and will show the same display, unless it is run on a base-station that is communicating with both normal and battery-powered outstations. It will then only display the communications with the battery-powered outstations.

9.2.6 ID - Display Input Digitals



```
                                INPUT DIGITALS

Alarm Flags CBHMFA_D:
 C: Data_Link Comms      B: Battery Low        H: Hardware  M: Bus_Link Comms
 F: Complete Comms Fail  A: Batt not charging  D: Dual Comms Fail

  Reg  Block
  No.   No. 0             15 16             31 0             15 16             31
    0:   0: _____A__oooooooo oooooooooooooooo ---------------- ----------------
   64:   2: ---------------- ---------------- ---------------- ----------------
  128:   4: ---------------- ---------------- ---------------- ----------------
  192:   6: ---------------- ---------------- ---------------- ----------------
  256:   8: ---------------- ---------------- ---------------- ----------------
  320:  10: _____A__oooooo1o oooooooooooooooo ---------------- ----------------
  384:  12: ---------------- ---------------- ---------------- ----------------
  448:  14: ---------------- ---------------- ---------------- ----------------
  512:  16: ---------------- ---------------- ---------------- ----------------
  576:  18: ---------------- ---------------- ---------------- ----------------
  640:  20: ---------------- ---------------- ---------------- ----------------
  704:  22: ---------------- ---------------- ---------------- ----------------
  768:  24: ---------------- ---------------- ---------------- ----------------
  832:  26: ---------------- ---------------- ---------------- ----------------
  896:  28: ---------------- ---------------- ---------------- ----------------
```

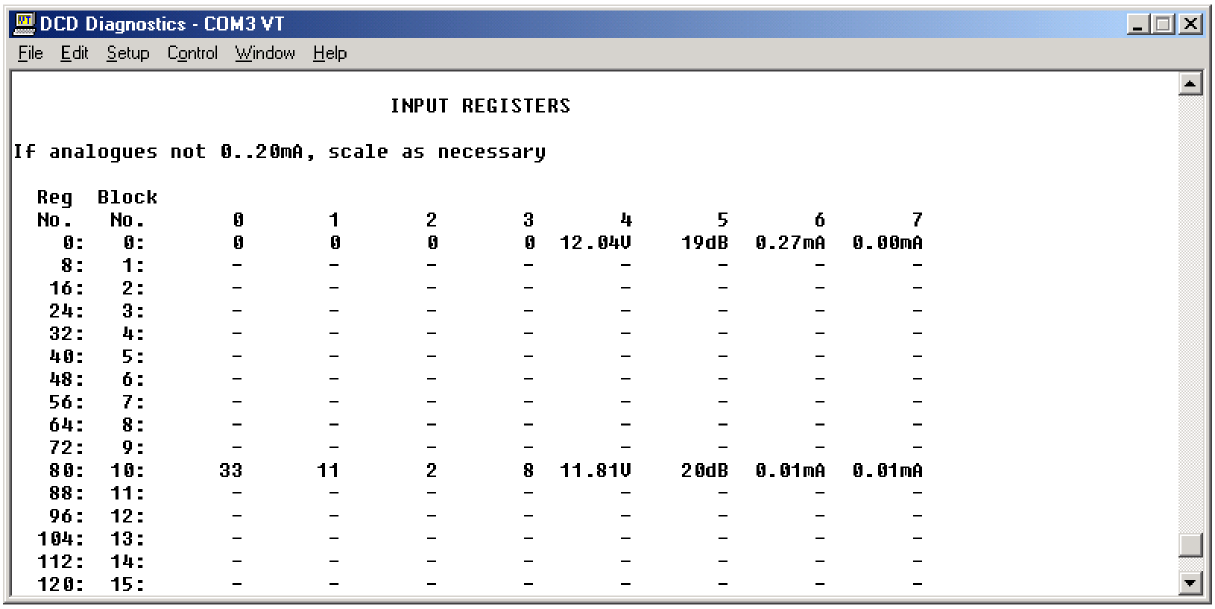
This displays the current state of all digital input data blocks. The display will scroll until either all 250 blocks have been shown or until the user exits (by pressing any key followed by enter).

Note that the base-station only displays data from the data blocks that are actually used. Furthermore, it learns which are root data blocks and displays the alarm flags contained within them as illustrated above. Each data block has 32 digitals, so the root data blocks can also incorporate one 16-channel digital input expansion module. If more are fitted the inputs will 'spill over' into the next data block.

Input data blocks that are not used by the local I/O at the base-station copy the respective inputs from an outstation. In the above illustration the base-station is obviously interrogating an outstation at address 10, so the first 8 inputs in this block are the alarm flags from that outstation, followed by its digital inputs.

Digital output data blocks can be similarly displayed using the OD command.

9.2.7 IR - Display Input Registers


DCD Diagnostics - COM3 VT
File Edit Setup Control Window Help

```
                                  INPUT REGISTERS

If analogues not 0..20mA, scale as necessary

  Reg   Block
  No.    No.       0       1       2       3       4       5       6       7
    0:    0:       0       0       0       0  12.04V    19dB  0.27mA  0.00mA
    8:    1:       -       -       -       -       -       -       -       -
   16:    2:       -       -       -       -       -       -       -       -
   24:    3:       -       -       -       -       -       -       -       -
   32:    4:       -       -       -       -       -       -       -       -
   40:    5:       -       -       -       -       -       -       -       -
   48:    6:       -       -       -       -       -       -       -       -
   56:    7:       -       -       -       -       -       -       -       -
   64:    8:       -       -       -       -       -       -       -       -
   72:    9:       -       -       -       -       -       -       -       -
   80:   10:      33      11       2       8  11.81V    20dB  0.01mA  0.01mA
   88:   11:       -       -       -       -       -       -       -       -
   96:   12:       -       -       -       -       -       -       -       -
  104:   13:       -       -       -       -       -       -       -       -
  112:   14:       -       -       -       -       -       -       -       -
  120:   15:       -       -       -       -       -       -       -       -
```

The software learns which are root data blocks and displays the data in the appropriate units. Thus in the above illustration it is readily apparent that it is a base-station interrogating an outstation set to address 10. The base-station power sully is 12.04V and the outstation is 11.81V. The signal strength of the last command received by the outstation was +20dBuV and the last response received by the base-station was +19dBuV. If either station was equipped with analogue input expansion modules the value of each analogue input would be displayed in mA, as are the two within each root blocks.

If any register inputs are read via ***Bus_Link***, they will be displayed as 16-bit values in the range 0…65535.

9.2.8 P - Display PLC Communications

The format of this display will depend on the protocol used on the ***Bus_Link*** port for communications with a PLC, SCADA system or some other device. Each line will begin with cmd: or rep: to define if the following data is a command or a reply to a command. If ***Bus_Link*** is configured as a slave device commands will originate from the host that is connected to it, whereas a master ***Bus_Link*** port will originate commands.

As well as showing the encoded message, the display also shows a plain English interpretation.

9.2.9 FR - Force Register

This command and the associated FD, FH and FC commands allow the user to force values into the database. The user is prompted for a block number, a register number within the block and a new value to enter. These commands can typically be used to force digital and analogue outputs to defined states to check the integrity of installation wiring and instruments. Note that the outputs will revert to their correct state when the system next updates. If these commands are used it is preferable to disconnect the aerial or leased line to prevent true data over-writing the test values. Alternatively the configuration can be temporarily modified (e.g. by changing to a different radio channel) for the duration of the tests.

INDEX