DNA
DeviceNet Controller Module

User's Manual

Software Version
Major Revision 47
Minor Revision 4

EPROM marked *DNA V4.7c*

17 August 2007

Group3 Technology Ltd., 2 Charann Place, Avondale, Auckland 1026. P.O. Box 71-111, Rosebank, Auckland 1348, New Zealand.
Phone: +64 9 828 3358 Fax: +64 9 828 3357 email: info@group3technology.com web: www.group3technology.com

Thank you for purchasing and using Group3 Control equipment. We hope you will join the growing number of people who are enthusiastic about the features Group3 Control has to offer.

Group3 has been designing and building specialized control systems for over fourteen years. We are constantly upgrading and improving our products and the supporting documentation. We welcome input from our customers, so if there are aspects of the system you particularly like, or things you would like to see implemented, improved or developed in the way of hardware, software or documentation please contact your Group3 representative, or Group3 directly with your suggestions.

The Group3 website (http://www.group3technology.com) contains some pages of information on Group3 products. This site will be regularly updated, and you may choose to check it from time to time, to learn about recent developments.

Listed on the "What's New" page are the recent developments and enhancements to the product range, grouped by the year of introduction.

The page "User Technical Notes" lists current versions of software, and also details past versions, with a brief list of the features they introduced.

Group3 Technology Ltd.
2 Charann Place, Avondale, Auckland 1026
PO Box 71-111, Rosebank, Auckland 1348, New Zealand.
Phone: +64 9 828 3358
Fax: +64 9 828 3357
Email: info@group3technology.com
Web: http://www.group3technology.com

Contents

DNA - Group3 DeviceNet Module

1 Specifications

Introduction

The DNA is a precision control and monitoring module that can be integrated into a DeviceNet system. It provides 16 bit resolution analog input and output capabilities, and isolated digital inputs and outputs in the same small unit.

When used with the SCM I/O board it has facilities to power up the Group3 corrected analog Hall probe, and use the signal from that as the prime input to a PID control algorithm embedded within its operating system.

The unit consists of a main circuit board contained entirely within a metal enclosure, linked by a short length of 40way ribbon cable to an I/O board mounted on top of the case. There are three possible I/O boards - the SCA and SCA2 which have rows of screw terminals for general I/O use, and a more specialised version - SCM - which is intended for use with the Group3 HPC analog Hall probe.

The module incorporates several features to ensure reliable operation: in addition to the checks built into the microprocessor itself there is a separate supervisor (watchdog) chip to reset the microprocessor if it detects incorrect operation as a result of an unusually powerful electrical transient. The software running within the module is of a multitasking nature, and one of the independent tasks is a dedicated self checking process that continually monitors all the other tasks and variables in the microprocessor. If anything is found to be out of normal bounds then a reset is forced.

A diagnostic port on each module allows system monitoring, configuring and debugging. The diagnostic port is invaluable at all phases of system commissioning. By using a standard terminal the engineer can access a module independently of the DeviceNet communications. A built-in menu system provides for a wide range of diagnostic, simulation, and set-up functions. A restricted access level security system prevents undesirable tampering by unauthorised personnel.

DNA Features / Specifications:

Two Analog Input channels

16 bit resolution, each with differential inputs.

Input Ranges:

bipolar	DNA-1,	-10 to +10 volt (1 lsb = 0.3mV approx.) and -100 to +100mV (1 lsb = 3 µV approx.)
unipolar	DNA-2,	0 to +10 volt (1 lsb = 0.15mV approx.) and 0 to +100mV (1 lsb = 1.5 µV approx.)

Input impedance: DNA-1, DNA-2 94kΩ

Both channels are sampled 33 times per second.

One Analog Output channel, 16 bit resolution

Output signal, current or voltage selectable:

current: all models: 4 to 20 mA (1 lsb = 0.25 µA)

voltage: DNA-1 -10 to +10 volt range (1 lsb = 0.3mV approx.)

DNA-2 0 to +10 volt range (1 lsb = 0.15mV approx.)

Output impedance: 10Ω

Eight Digital Inputs, sampled at time of message poll.

The main board senses TTL level inputs on the 40 way connector.
Inputs from sensors and switches are wired to the I/O board first, which opto-isolates the signals before passing them on to the main board.

Eight Digital Outputs, set at time of message poll.

The main board generates output signals by open drain MOSFET drivers, sinking current to digital ground. There is a 1000 ohm pull-up to +5V on these lines. The output drive signals are passed up to the I/O board where they are used to switch relays, which isolate the module from the output loads.

The I/O board is available in three forms:-

SCA Relays are SPST (form A), switching 100 volt 500mA max.
screw terminals

SCA-2 Relays are SPDT (form C), switching 110 volt, 1 amp max.
screw terminals

SCM Relays are SPST (form A), switching 100 volt 500mA max.
D connectors

Note:

The SCA-2 board requires a 5 watt power supply installed on the DNA-1 and DNA-2 main boards. Correct operation cannot be guaranteed if the SCA-2 is used with a DNA-1 or DNA-2 main board fitted with the older 3 watt supply, particularly if a hand held display terminal is to be powered from the diagnostic port.

Power supply input required to power up the isolated processing and I/O sections:

18 to 36 volt DC, or 14 to 26 volt (rms) AC,

3 Watts as simple I/O device,

5 Watts if used with Group3 corrected analog Hall probe.

Connector required - Phoenix MSTB 2.5/2-ST-5.08, or Klippon equivalent.

Unit contains an internal switch mode supply, providing isolation up to 200 volts.
Inrush current at switch-on less than 3 amps peak, half height width of 200microseconds.

No internal manual adjustments - all setup and calibration is implemented by software through the diagnostic port.

DeviceNet Interface:

Communications by standard DeviceNet protocol, as a Group2 only slave.

Pre-defined Master-Slave connection set.
All values to be output from the module can be sent in one message.
All values measured by module can be read back in one message.
Module able to handle polls at 2ms intervals at 500Kbaud.
Full implementation of Parameter Object - 88 instances, defined in appendix B.

Settings: - screwdriver adjustment of 10 position rotary switches, on end panel of unit.

Baud rate - rotary switch selection of 125K, 250K, 500K
10 posn, of which 0-2 are valid, remainder cause module status LED to flash red.

MAC ID - two rotary switches, for tens and units.
10 position, valid values 0-63, remainder cause module status LED to flash red.

Indicators: on end panel of unit.

Two bicolour LEDs showing Module Status and Network Status.
Use as specified by ODVA.

Connector:

5 pin micro DeviceNet connector, male, on end panel of unit.

All I/O lines of the module are isolated from the DeviceNet signal and power lines.

Opto-couplers on data lines.
Opto-coupled bus power sense.

Current draw from bus nominal 24V is 50mA.

PID Control

The module has an embedded PID control algorithm that can be invoked by sending the appropriate command, or by selecting it through the diagnostic port.
As a general I/O device, without PID, all inputs are general purpose inputs, and the value sent to the analog output is used directly to set the output voltage.
If the unit is being a controller, with PID invoked, then the value sent by the control computer to the analog output data location is the desired setting for the control variable - the setpoint.

Also, when used in PID controller mode, certain digital inputs can be selected to have local and immediate effects, or they can be used as general digital inputs. The main benefit of activating their special functions is to prevent a large build-up of the time integral term of the PID function if, for instance, the power supply is not switched on, or is in slew rate limited mode.

There is also a facility to send a command over DeviceNet to a module operating in PID mode, to hold the output at its current value.
It is envisaged that these PID mode options, and the facility to switch between modes, are really to be seen as aids in the development and setting up of a system, rather than features that would be needed once a machine has entered production.

Diagnostic Port

The Diagnostic Port allows configuration and local control over-ride. It provides direct access to all input and output channels, with useful system diagnostic features. System set-up parameters, calibrations and resets can all be performed through this port.
Access is from a terminal with an RS-232 serial port, via an 8 pin mini-DIN socket. A small hand held terminal with an LCD screen is available from Group3. Communication is fixed at 9600bd, 7data, even parity, 2 stop bits.

All features of the diagnostic port can also be accessed over the DeviceNet network by using the Parameter Object implementation, as described in Appendix B of this manual.

Noise Immunity

Immunity to noise and electrical transients is greatly enhanced by hardware, software and enclosure design techniques.

Several layers of isolation are employed - the DeviceNet interface is opto-isolated, the module processor and I/O sections are isolated from the local power supply by a DC to DC converter, the digital outputs are isolated by the use of relays, and the digital inputs are opto-isolated.
The power supplied to the Group3 corrected analog Hall probe is isolated by yet another DC to DC converter. The signal returned by the probe is a floating signal (with respect to its case and power supply), and is referenced to the level of analog input 0 of the controller.

Two stage transient suppression -all I/O pins on the main controller board have fast acting, voltage limiting components installed, and further suppression and isolation components are on the I/O board.
Particular attention has been paid to hardware design for minimal noise susceptibility - good grounding and decoupling layout, minimising signal lengths and areas, and inclusion of hardware watchdog circuitry.

Robust embedded software.
Self diagnosing and fault tolerant software allows graceful recovery after a disturbance caused by a transient. The unit is continually running self checks on its operating state - if anything is found to be out of allowed limits the microprocessor can recover to a known good state without operator intervention.

The case is of an all metal design, providing the best EMI shielding.

I/O Boards – SCA, SCA-2 & SCM - Features / Specifications:

These boards provide additional transient protection and/or isolation on all I/O channels.
The SCA boards provide rows of screw terminals to allow easy field wiring to each I/O channel, while the SCM uses 'D' connectors for use with pre wired harnesses.

Eight Digital Inputs

Inputs from sensors and switches are wired to the I/O board, which opto-isolates the signals before passing them on to the main board.

Opto-couplers on each channel can be selected to operate in two possible modes:

> Signal powered channels are completely independent, minimum of 5mA.
>
> Contact closure channels share a common return potential (which is isolated from the rest of the unit up to 500V).

Each digital input channel has a three position screw terminal block associated with it. Two of the terminals are for the signals, and the third is provided to terminate the shield of the signal wires - shielded twisted pair should be used for all signal wiring.

Digital input signals are isolated from the rest of the module by opto-couplers. Signal currents must flow in the correct direction to operate the opto-couplers.
The input pins are labelled (In), and (Ret).
The inputs can be configured in two different ways:-

Signal powered

> If set to signal powered (LOOP), then the input wiring must be such that the more positive wire is joined to the (In) pin, so that signal current flows into the (In) pin, and returns out of the (Ret) pin.
> In many systems this would mean that the active signal output from the device being sensed would be wired to the (In) input, and the devices signal return / signal common / signal ground would be connected to the (Ret) pin.

Contact closure

> If the inputs are set to sense a contact closure, then the SCA board generates the current required to power the opto LED. A few mA of current will flow out of the (Ret) terminal, through the external contacts to be sensed, and back in the (In) terminal.

Eight Digital Outputs

SCA Outputs are SPST (form A) normally open relay contacts.
Switching 100 volt, 500mA maximum,
Each channel has a three position screw terminal block associated with it. Two positions are for the relay contacts, while the third is provided to terminate the shield of the signal wires.

SCA-2 Outputs are SPDT (form C) relays providing both normally open and normally closed contacts.
Switching 110 volt, 1A maximum.
Each channel has a four position screw terminal block associated with it. Three positions are for the NO, COM, and NC contacts, while the fourth is provided to terminate the shield of the signal wires.

SCM Outputs are SPST (form A) normally open relay contacts.
Switching 100 volt, 500mA maximum,
the contacts are taken to the 'D' connectors on the board.

Isolation of contacts from rest of unit - 300 volts

Analog Inputs

Each of the analog channels has a four position terminal block of 3.81mm pitch.
The terminal block for analog inputs gives terminals for the non-inverting input (+), the inverting input (-), analog ground, and a termination for the shield.

Analog Output

SCA Four position terminal block, with separate terminals for the voltage output, 4-20 mA output, analog ground, and shield termination. Selection of voltage output or 4-20mA output must be done through the diagnostic port, and wiring to the appropriate terminal.

SCA-2 Three position terminal block, with terminals for the output, analog ground, and shield termination. Selection of voltage output or 4-20mA output must be done through the diagnostic port, and positioning a plug-in resistor in the appropriate labelled position.

SCM The analog output is available on the central D connector (see page 2-11)

Physical Details

Connectors

Between the boards –
40 pin dual row 0.1" spacing boxed header on main board,
40 way ribbon cable with matching socket attached to I/O board.

Diagnostic Port - 8 way miniDIN round socket.

DeviceNet power and data connector - sealed micro style, right angle male.

Module power - Phoenix MSTB 2.5/2-ST-5.08, or Klippon equivalent.

Case

Aluminium extrusion 140 x 92 x 29 mm
Main board slides into slots within the hollow section.
I/O board slides into slots extruded on the outside of the case.

Stainless steel endplates with integral DIN rail locking system.

Aluminium shield plate mounted over top of I/O board, covering components but allowing access to the I/O connectors.
Single captive knurled metal screw secures shield plate.

Mounting

Stainless Steel DIN rail clamp is integral to the metalwork of the unit.

Unscrewing one captive knurled screw, and unplugging the 40way ribbon cable from the main board allows the I/O board to be slid out from the mounting rails, with all the I/O wiring still in place.

Moving the DIN rail locking bar then allows the main unit to be lifted off the rail.

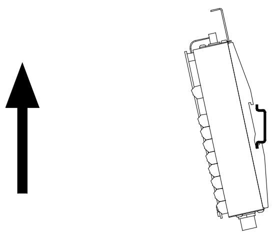
No tools are required to remove the main module from an installation, allowing quick replacement or upgrades.

Physical size

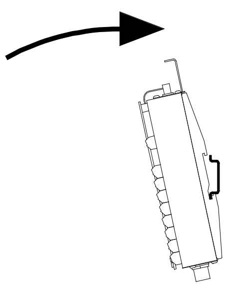
Maximum dimensions of entire unit, with DIN rail mounting
160 mm long x 92 mm wide ,
x 52 mm high off the DIN rail mounting surface.

Mounting a Module onto a DIN rail.

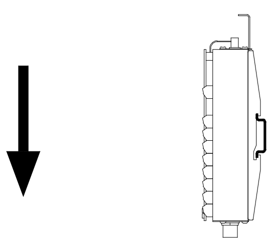
DIN rail



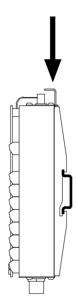
Lift the locking tab,
then lift the DNA, hooking the bottom edge
of the DIN rail with the backplate.



Rotate the DNA to a vertical position



Slide the DNA downwards so that the top edge
of the DIN rail is hooked by the backplate.



Push down the locking tab.

2 User Settings

DNA Settings

I/O channels need to be set up using the diagnostic port and a terminal.
See section 3 of this manual for a full description of the diagnostic port functions.

Analog Inputs: for each of the two input channels:

Select the desired range: bipolar versions	-10 to +10 volts (default),	or	-100 to +100 mV
unipolar versions	0 to +10 volts (default),	or	0 to +100 mV

Set the desired filter factor and filter window.

Filtering:
The user can select various time constants for a software based time averaging filter, and optionally a window within which the filtering is active.

If measuring single ended voltage inputs (signals referenced to analog ground), tie the inverting input (labelled -) to analog ground, and apply the signal to be measured to the non-inverting (+) input.

Analog output:

Select either the voltage range (default), or 4 to 20 mA range.
Note that the module can be set to deliver either voltage output or current output, but not both at the same time. See the section on I/O board settings later in this section, for hardware settings required to select the type of analog output.

Digital Inputs:

Set the polarities (default is LOW)

Low polarity means if an input opto-coupler has current flowing through it, a '1' will be read at the control computer.

Digital Outputs:

Set the polarities (default is LOW)

Low polarity means for outputs that if the control computer sends a '1' to that channel, then the output relay will turn on.

Polarities are adjusted through the diagnostic port, in the System configuration / Digital menu.

Polarities of digital channels can get quite complex and confusing, with several adjustments to the polarities available in hardware and software. Unless there are good reasons to change, leave all polarities set to the factory default of Low.

Reset behaviour:

On power up all digital outputs are cleared to OFF.
A jumper link on the board can be set to determine the behaviour of the outputs on a watchdog or self check reset. If the link is in the HOLD position the outputs will remain at their last value if a watchdog reset occurs. If the link is in the CLEAR position, the outputs will be cleared to OFF if a watchdog reset occurs.
The modules are shipped with the link in the CLEAR position. If the link needs changing see the section on "Removing the main board from a DNA module" on page 6 of section 1.
The HOLD position may be preferable if a momentary turning off of outputs could lead to other interlocks tripping out, while the CLEAR position may be the best for a safety critical situation.

Using PID

The module can be set to execute a PID algorithm, to hold a control variable steady no matter what external drifts and influences there are.
If set into PID mode, the module uses Analog input channel 0 as the feedback input, and the analog output is altered by the module itself to achieve a desired value on the feedback input. This is closed loop control - all the control computer has to do is send the module what the desired input value is - the setpoint. The module then alters the analog output so as to achieve the new desired input value, as fast as possible, with minimum overshoot. Tuning of the loop coefficients is a complicated process - we will describe a couple of approaches.
If the unit is set to operate in PID mode, then the user has the option to enable various special uses of certain inputs, to reset or halt the PID algorithm.

Selecting PID coefficients

First, a word of warning - when you select PID, you are handing control of the programming output over to the module - if you have incorrect PID coefficients then the module will faithfully use them! It is very much like a computer - if you put garbage data into it, you will get garbage out, very quickly. If your module is being used to program a multi kW power supply, then drastic things can happen.

The feedback sensor must be connected and powered up before you invoke PID - if the probe is not connected, you won't get any feedback signal, and the controller will just swing the power supply to full scale, trying desperately to get the input signal to change.
You can only tune a closed loop system if the loop is in fact closed - the system has to be live to some extent. However, when setting up a system for the first time we suggest you minimise the consequences if you enter incorrect numbers - e.g. don’t have beam running, and be ready to hit an emergency cut off at any time!.

You will need some method of displaying the response of the system over a period of a few seconds - digital scope, computer data logger, chart recorder etc,

There are whole text books on control theory and how to optimise the coefficients in a PID control system to get the fastest response with minimal errors. Such a treatise is beyond the scope of this manual, but we will attempt to describe a couple of ways of getting a system running with reasonable efficiency. Also remember that some theoretical approaches to the subject assume that you have available “ideal” components - eg a power supply with unlimited voltage and current capability - this is seldom the case!.
Systems can have quite non-linear responses to inputs - tuning optimally for small scale step changes may mean that when a large step is requested the power supply is not capable of supplying the voltage or current, and so goes into slew-rate limited mode. This can allow large error terms to build up, and significant overshoot can result.
In some cases it may be more sensible to de-tune from optimal small step response, to allow for maximum rate of change on large step inputs without pushing the power supply into current or voltage limiting. After all, the large step inputs are going to take the most time to respond to, and may be more important to tune for. Small step inputs will be actioned more quickly just because they involve smaller changes. It really depends on how the system will be used once set-up - just be aware of the compromises involved.

The Simple Method

This method will get your system up and running with the least chance of having it burst into oscillation , forcing you to hit the STOP button. It will not necessarily produce the fastest transitions, but it gives you the chance to slowly play with the system, to get a feel for it. If you want to go straight for fully optimised tuning, then see the next section.

Start with Integral term only - use the diagnostic port to set the P and D terms to zero.

The module runs the PID algorithm every 30ms. This means that if you give a command to the module to action a step change in output, the module will calculate the error (difference between the actual output and the new desired output), multiply it by the Integral term, and add it to the actual output, every 30ms.
For example, if the present, actual output is zero volts, and you give a command for a full scale change (for this module that is 10 volts) then the resulting error is 10V. This will be multiplied by the Integral term, and the result added to the output every 30ms. Of course as the actual output approaches the desired output the error gets less and less, but initially at least the error is maximum, as stated.

If you set the Integral term to 0.01, then with a 10V error, every 30ms the output will have 10V x 0.01 = 0.1V added to it. So adding steps of 0.1volt to the output means that the output will take 100 steps to reach 10 volts, and 100 steps at 30ms per step will take 3 seconds. This is highly approximate, because of the diminishing error, but at least it gives you an idea of the order of magnitude.

Another way to approach a completely new system is to decide how long you think you need to observe the response of the system - we would suggest about 10 seconds for a full scale swing - nothing ought to change so fast that you can't hit the STOP switch!.
So ten seconds amounts to approximately 333 periods of 30 ms. Taking the required swing of 10 volts at the start, and dividing it into 333 steps gives 0.03V (30mV) per step. So we want 30mV added to the output every 30ms to get a total of 10V change in ten seconds. At the start the maximum error is 10 volts, and the module will multiply this by the integral term (**I**) to get 30mV as the step increment 10V x **I** = 30mV

=> **I** = 0.003

The above two approaches give you a starting value for the Integral term, so you can watch the behaviour of the system, check your monitoring instruments etc.
You will most probably want to increase the speed of the response, so increase the Integral term - doubling the value will approximately halve the time taken to attain the final value.

As you increase the speed of the system, keep an eye on the behaviour of the power supply - it may go into voltage or current limit. If this happens the controller can not make the system respond any faster - you are at the limits of what the power components are capable of.

If you are not being limited by the other components, you can try adjusting the other PID terms - the Proportional term, and the Differential term.

A larger **D** will give a faster rise initially, but may increase overshoot and oscillation.
Bear in mind that the Differential term is a time based factor so, if you are trying to calculate it, you need to know that the controller does the PID calculations every 30ms. Time must be expressed in terms of this 30ms period.

Increasing **P** too much may lead to uncontrolled, exponentially increasing oscillations, so take care, only do small changes, - and be ready with the STOP switch!.

In general, an optimised system will have the **D** term at approximately one quarter the value of the **I** term. **D = I/4**

Tuning for Ultimate Sensitivity

This method derives the PID coefficients from empirically determined values. It does involve provoking oscillations in the system, so care must be taken to ensure there are no consequences in other parts of the machine. To use this method effectively the components in the system must be sized so that everything is operating in a linear mode - it cannot be used if the power supply is entering voltage limiting mode during step changes.

By considering the design of the control loop, determine the overall gain of the process.

Consider a 10% step increase into the power supply. If the controller output were to be disconnected from the power supply, and 10% increase independently added to the input of the power supply, by what percentage of its full range would the feedback transducer value change?

The overall process or loop gain, designated **K,** is the ratio

$$\frac{\textbf{change in transducer output} \quad \text{(as a percent of full scale)}}{\textbf{programming input step} \quad \text{(as a percent of full scale)}}$$

The **change in transducer output** is the change in the output of the feedback transducer when used in the system This is the feedback supplied to the controller module.- e.g a magnetic field sensor, producing 0-10 volts output for a 0 to 1 Tesla field.

The **programming input step** is the step input injected at the programming input. This is the output that the controller normally generates to feed into , for example, a 0 to 150 amp power supply with a 0-10volt programming input.

For the sake of this example, let us say that the power supply is a 0 to 150 amp output, controlled by a 0 to 10 volt programming input. The magnet is designed to produce a field of 1T for a current of 150A, and the field sensor produces a 0-10 volt output for a 0-1 T field.

In this case a 10% step input to the power supply will increase the current by 15 amps, increasing the field by 0.1T, so the sensor will increase its reading by 10% of its full scale.

So the overall gain is:- $\mathbf{K} = \frac{10\%}{10\%} = 1$

If the components of a system have been sized optimally then the ratio K is likely to be 1, otherwise the feedback sensor span or the supply are oversized for what they are required to do.

Use the diagnostic port to set the **I** and **D** terms to zero.
Set the **P** term to a value such that **P.K** = 2

For our example, set **P**=2

Now begins a process of trying to find the value of **P** which causes the system to break into oscillation. Take care as the value approaches the optimum, because if you exceed the optimum the oscillations will grow exponentially in amplitude.

Now alter the setpoint to say 30% of full scale, allow time for the system to settle, then add another 10% - ask for 40% of full scale. Observe the trace of the sensor feedback. It should rise, overshoot, then settle back to the desired level.
Set the setpoint back to 30%, then increase **P** by the same amount again. Ask for 40% output, and watch the systems response. The overshoot and ringing should increase.

Continue the process of increasing **P** , giving a 10% step increase in setpoint, and watching the behaviour of the system. You are trying to find the value of **P** for which a small step input causes the system to go into oscillations of constant amplitude - that is, the oscillations neither die away, as they did at the start, nor increase in amplitude.

Note down this ultimate value of **P**, designated $\mathbf{P_u}$
Also measure and note the period (in seconds) of the oscillations at this value, designate it $\mathbf{T_u}$

Text books indicate that empirically it has been found that the optimum values for the coefficients in a PID system are as follows:

$$\mathbf{P} = 0.6\ \mathbf{P_u}$$
$$\mathbf{I} = 0.5\ \mathbf{T_u}$$
$$\mathbf{D} = 0.125\ \mathbf{T_u}$$

However, these equations are not immediately applicable because, for the Group3 controller, time must be expressed in units of the 30ms recalculation rate of the PID algorithm.

The period (in seconds) of the oscillations, designated $\mathbf{T_u}$ above, must be divided by 33 to express it in units of the 30ms PID rate .

The above equations then become:

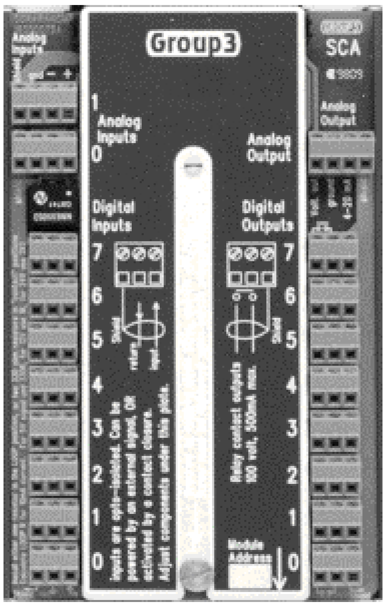
$$\mathbf{P} = 0.6\ \mathbf{P_u}$$
$$\mathbf{I} = 0.015\ \mathbf{T_u}$$
$$\mathbf{D} = 0.0038\ \mathbf{T_u}$$

These are the values to be entered into the Group3 controller.

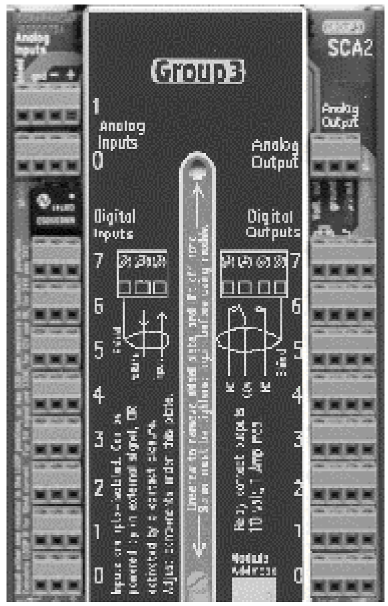
These values are supposed to produce the fastest response with minimum overshoot, and quickest settling. However, as with all generalisations, use them as a reasonable approximation, and watch closely as you first start the controller using these new coefficients.

I/O Board Settings - SCA, SCA-2, and SCM

There are three I/O boards available for the DNA module. This page outlines the main points about each type - settings are described on the following pages.



SCA



SCA-2

SCM

Common to all

Optocouplers on digital inputs

Relays on digital outputs

Transient suppression and filtering
on analog inputs and outputs

SCA

Output Relays - SPST, form A

Rows of screw terminals:-

- eight x 3-way for digital inputs
- eight x 3-way for digital outputs
- two x 4-way for analog inputs
- one x 4-way for analog output

SCA-2

Output Relays - SPDT, form C

Rows of screw terminals:-

- eight x 3-way for digital inputs
- eight x 4-way for digital outputs
- two x 4-way for analog inputs
- one x 3-way for analog output

SCM

Output Relays - SPST, form A

D9 to go to Hall Probe
 1 analog input, 1 digital input
 ±15V supply to probe

D15, to go to controlled device.
 analog output, analog input
 2 digital input, 1 digital output

D25, for other digital channels,
 5 digital input, 7 digital output

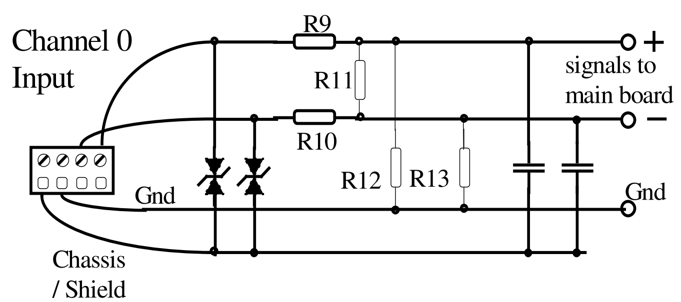
Power supply providing ±15V
 for Group3 analog Hall probe

Settings

The shield plate can be removed by unscrewing the knurled screw a turn or two - it does not need to be released completely. Gently lift the other end of the shield plate to "pop" it off the plastic stand-off, then slide the plate out from under the head of the knurled screw. This gives access to various resistors to allow for voltage division on the analog channels (default is a 1:1 - straight pass-through, no division), Resistors are also accessible to alter the nature of the digital inputs - whether the input opto-couplers are used to sense contact closures (the default), or are to be powered by the input signal itself. Details about these adjustments are given in the following pages. The adjustments and component labelling are the same for all types of I/O board. The positions of components and the final I/O connectors are different for the different I/O boards.

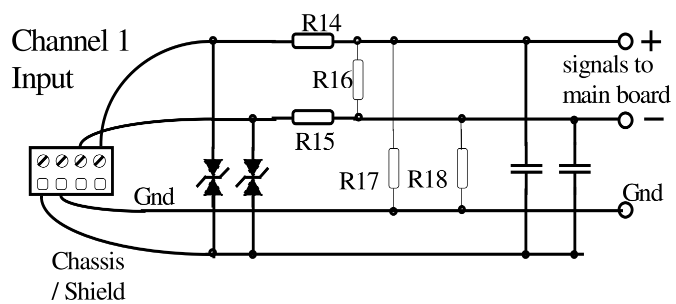
Analog Channels:

Analog Inputs 0 and 1.
The resistive dividers can be set to function in the following manner:-



Considering Channel 0:
As shipped, R9 and R10 are installed as 10Ω resistors. This feeds the signal through unmodified. If the signal needs to be attenuated, then there are two methods.

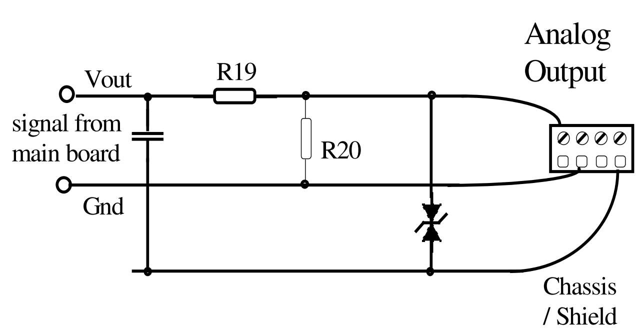
1) For sensing voltages greater than 10 volts:- use R9 and R12 as the divider on the +input, and R10 and R13 as the divider on the -input. The values should be at least a few tens of Kohm - it depends on how much load the signal can support. To preserve balance in the system, choose values R9=R10 and R12=R13, Resistors should be precision 0.1% tolerance, so as not to introduce differential errors as the absolute value of the signal varies.



2) To divide smaller voltages down, so as to use the 100mV range on the inputs, then a more accurate way is to use R9, R10, and R11 as the voltage divider resistors.

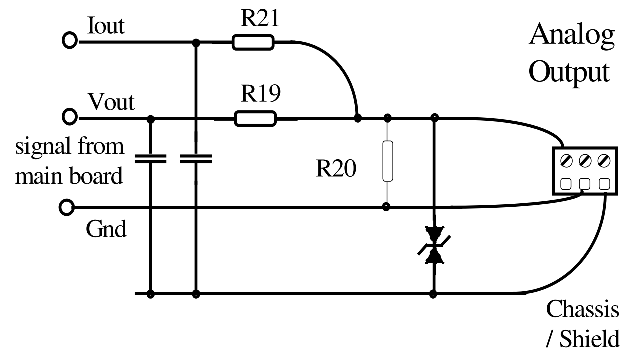
SCA Analog Output:

R19 is installed as a 10 ohm resistor, effectively passing the signal through. If the output voltage needs to be scaled down, then replace R19, and choose R20 to suit. The sum of R19 and R20 should not be less than 2kohm. The 4-20mA output has its own output terminal.



SCA-2 and SCM Analog Output:

These two I/O boards have one terminal for the analog output, and the voltage or current signals are taken to it by selecting an appropriate resistor.



Install <u>Either</u> R19 for Voltage output, <u>or</u>
R21 for 4-20 mA output
but NOT both at the same time.

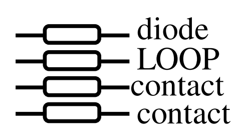
For 10 volt output install R19 as a 10 ohm resistor, effectively passing the signal straight through.
If the output voltage needs to be scaled down, then replace R19, and choose R20 to suit. The sum of R19 and R20 should not be less than 2kohm

For 4-20 mA current output, remove R19, (and R20 if fitted) and install R21 as a 10 ohm resistor.

Digital Inputs

The digital inputs are isolated from the rest of the DNA module by opto-couplers. By positioning selected resistors in the long socket strip under the shield plate, these inputs can be used in two different ways - Contact Closure, or Loop Powered. Note that a channel should either have two 330R resistors in the "contact" positions, OR one appropriate resistor in the LOOP position, never all three resistors. The diode remains in the sockets in either case.

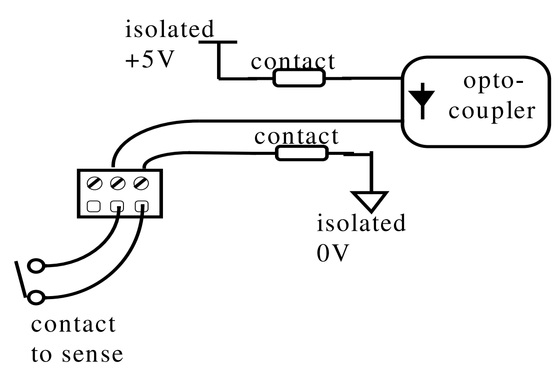
Each channel:-



1) **Contact Closure** (as shipped)

In this mode two 330 ohm resistors are required, installed in the positions labelled "contact". Closing an external contact allows a small current of 5 mA to flow through the opto-coupler, thereby triggering the input. The current is supplied by a small DC to DC converter, providing 5V DC isolated from the rest of the module.
Note that this isolated 5 volt supply is used by all the input channels that are set to sense contact closure - therefore all contacts sensed by this module must be at the same potential.

for contact closure:-

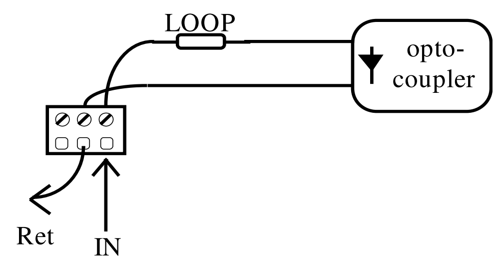


2) Loop powered.

Some or all of the input opto-couplers can be set to be turned on by a signal loop applied externally. The two 330 ohm resistors must be removed from the positions labelled "contact", and one resistor of the correct value placed in the position labelled "LOOP". The current must flow into the terminal labelled "IN", and out of the terminal labelled "ret" (for "return"). The resistor value should be selected to limit the current flow through the opto-coupler to 10 mA or so.
Suggested values are:
for a 5 volt signal, use 330R
for a 12 volt signal, use 1K ohm
for a 24 volt signal, use 2K2

Each channel set to "LOOP" mode is isolated from all other channels.

for loop powered:-



Digital Outputs

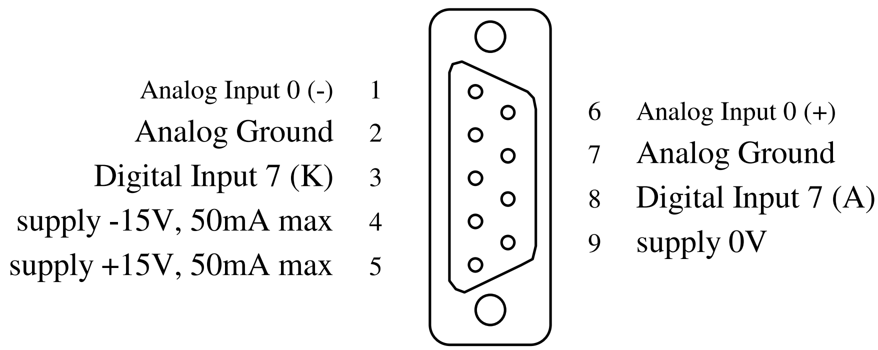
There are no settings for digital output channels.
The SCA and SCM I/O boards have small reed relays - single pole, normally open. The contacts are taken to the I/O terminals.
The SCA-2 I/O board has coil relays, single pole change over style.
The common, normally open, and normally closed contacts are taken to the I/O terminals.

SCM I/O board pin-out

I/O signals are brought into the module by way of vertical mount female D-sub connectors. Different connectors are used for each grouping of signals, in part to avoid the possibility of mis-connecting in the field.

Probe connector

D9, female, vertical mount.
Supplies isolated +/- 15 volt power to the probe
(generated by small DC to DC converter on the I/O board)
Brings back analog signal from probe, past extra transient protection, and into Analog input 0.
Returns isolated digital signal to digital input 7.
(used for “over temperature” signal from Group3 Hall probe.)



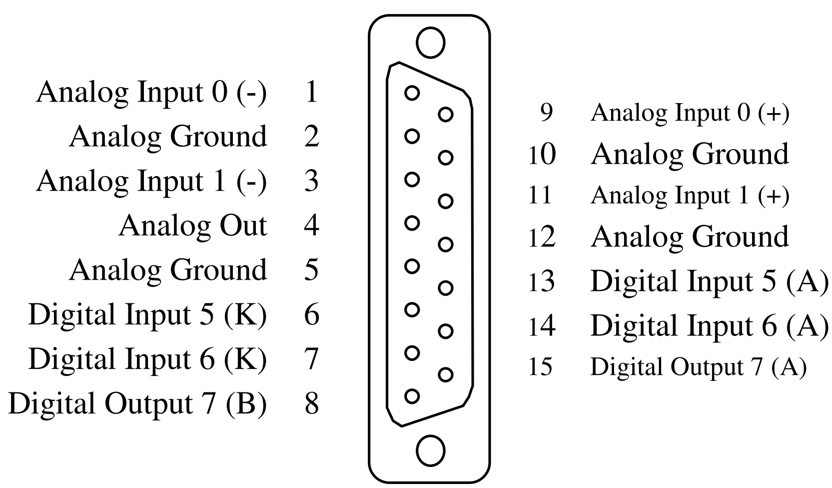
Power Supply Connector (Analog I/O connector if not using Hall probe)

D15, female, vertical mount.
Supplies output signal to the programming input of the power supply.
Brings in signal from the power supply, as voltage or current readback, past extra transient protection, and into analog input 1.

(Also has Analog input 0 available, for use as a generic I/O device,)
(only if 2 links beside the D15 connector are in place)

2 digital inputs:
- input 5, as "power is on" readback from the power supply.
- input 6, as general digital readback from supply, can be set to "hold PID", if for instance the power supply is in slew rate limited operation

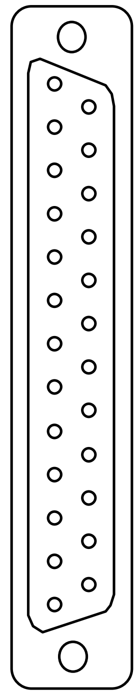
1 digital output:
- output 7 as "power ON/OFF" control to supply.



Digital I/O Channels Connector

D25, female, vertical mount.

The remaining 5 digital inputs and 7 digital outputs available on this connector.



Digital Input 4 (A)	1	14	Digital Input 4 (K)
Digital Input 3 (A)	2	15	Digital Input 3 (K)
Digital Input 2 (A)	3	16	Digital Input 2 (K)
Digital Input 1 (A)	4	17	Digital Input 1 (K)
Digital Input 0 (A)	5	18	Digital Input 0 (K)
no connection	6	19	Digital Output 6 (A)
Digital Output 6 (B)	7	20	Digital Output 5 (A)
Digital Output 5 (B)	8	21	Digital Output 4 (A)
Digital Output 4 (B)	9	22	Digital Output 3 (A)
Digital Output 3 (B)	10	23	Digital Output 2 (A)
Digital Output 2 (B)	11	24	Digital Output 1 (A)
Digital Output 1 (B)	12	25	Digital Output 0 (A)
Digital Output 0 (B)	13		

Digital Outputs are contacts of reed relays, with 100volt, 500mA maximum rating. The contacts are wired directly to the D25 connector pins - therefore the pins designated (A) and (B) of each output are exactly equivalent.

Digital Inputs are isolated from the rest of the module by opto-couplers. Signal currents must flow in the correct direction to operate the opto-couplers. The input pins are labelled (A) for anode of the opto LED, and (K) for cathode of the opto LED.
The inputs can be configured in two diferent ways:-
If set for signal powered inputs then the input wiring must be such that the more positive wire is joined to the (A) pin, so that signal current flows into the (A) pin, and returns out of the (K) pin. In many systems this would mean that the active signal output from the device being sensed would be wired to the (A) input, and the devices signal return / signal common / signal ground would be connected to the (K) pin.

If the inputs are set to sense a contact closure, then the SCM board generates the current required to power the opto LED. A few mA of current will flow out of the (K) terminal, through the external contacts to be sensed, and back in the (A) terminal.

CAUTION

Observe antistatic procedures when handling circuit boards.

The circuit boards of this module form a precision scientific instrumentation system. The circuitry is protected against the normal static discharge from a human body while it remains in the case, (or the packing it is shipped in).

Whenever circuit boards are removed from their normal mountings, or when they are unpacked from the anti-static bags that they are shipped in, the handler MUST observe industry standard antistatic handling procedures.

This means working at a bench that has a grounded conductive top surface, and using a conductive wrist strap or similar to ground the operator.
Failure to observe these precautions can cause damage, which if not immediately noticeable, can lead to unreliable operation and premature failure.

Removing the main board from a DNA module

1) Lift the locking bar to release the unit from the DIN rail.

2) unscrew the knurled nut that secures the shield plate - make sure the screw is fully retracted - the shield plate can remain in place.

3) unplug the 40 way ribbon cable connector from the end of the case. Note that the end of the ribbon cable that is attached to the I/O board is permanently attached to that board - do not try to remove it!

4) slide out the I/O board from the aluminium extrusion case - all the I/O wiring to the connectors can remain in place.

5) take the main body of the unit over to an approved anti-static workstation.

6) unscrew the four screws on the end that has the 40 way connector on it. The screws on the other end (the one with the LEDs, switch etc.) should be left securely fastened.

7) Gently lift the end plate out over the 40 way connector and push it down out of the way so that the inner circuit board can be withdrawn. There is enough flexibility in the stainless steel backplate to allow the circuit board to be withdrawn, but do not bend the backplate any further than necessary or permanent deformation will result.

There are three reasons that a user may wish to gain access to the main board:
- to change the EPROM
- to replace the fuse - use Shurter OMF63, 3 amp rating.
- to change the behaviour of digital outputs on an internally generated RESET.
There is a 3 pin header, with a mini-link jumper on it, up by the 40 way connector.
The two possible positions are labelled on the board as "Hold" and "Clear".

Re-assembly
Essentially the reverse of the above procedure, but take note of the following:-

- when sliding the main board back into the case make sure both sides go into the lowest slot possible.

- ensure the LEDs and switch stem protrude through the holes in the other end plate.
  do not force the board in - guide it gently.

- guide the end plate over the 40 way connector housing.

- when replacing the screws, do not overtighten, or the aluminium thread will strip. If this does happen, replace the screw with a longer shafted #4 self tapping screw.

3 Group3 Diagnostic Port

Introduction

The diagnostic port provides the facility for configuring and monitoring the module (Device Interface / CNA / DNA). Operating parameters and configuration setup are stored in non-volatile memory within the module. This approach eliminates the need for hardware switches and trim-pots as well as improving overall system reliability and ease of use. The diagnostic port is designed to interface to an ASCII terminal, which could be a PC running a standard terminal emulation program, such as MS-Windows 'Terminal', or a small hand-held terminal.

The features of the diagnostic port for each module can also be accessed from the main control computer, by using appropriate software. The diagnostic port functions can be accessed by either the main computer over the network, or by the diagnostic port on the module, but not by both at the same time. To switch between the two options, press **<ESC>** or **<Enter>** on the keyboard you wish to have control.

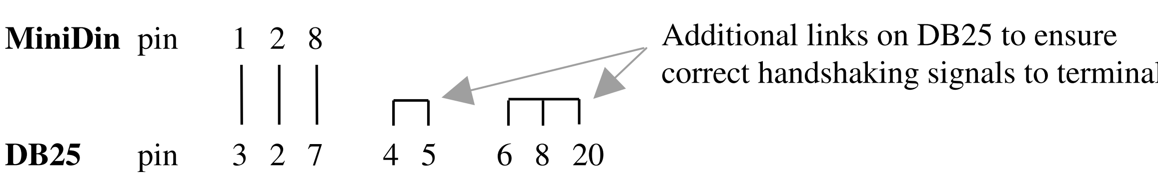
Note that the diagnostic port cable must be disconnected from the module while the system is operating in the vicinity of electrical transients and high voltage discharges.

Diagnostic Port Connector

Electrical: Standard RS232C voltage levels. No hardware handshaking.
Connector: MiniDin 8.

Pin	Use	Pin	Use
1	Transmit data (output)	6	Not Connected - factory use only
2	Receive data (input)	7	Five volts output (85 mA max.)
3	(constant high level)	8	Signal ground
4	Not Connected - factory use only	9	Chassis ground
5	Not connected		

A diagnostic port cable, length 2 meters, with a MiniDin 8 connector on one end and a DB25 connector on the other is available from Group3, part number DPC2. The wiring between the connector pins is given below:



Diagnostic Port communication parameters are fixed in software, and are not user adjustable.

Baud Rate	9600
Data Bits	7
Stop Bits	2
Parity	even
Flow control	none

Diagnostic Port Menu System

Overview - Refer to the Menu "tree" diagram - following.

The diagnostic port software has a hierarchical menu structure in the user interface. To use the diagnostic port a correctly configured terminal should be attached to the port. Pressing **Esc**, or **X** will bring up the main menu screen. Any invalid keystrokes will print **???** on the terminal and redraw the current menu. In all sub-menu levels typing **X** will back up one level and display the previous menu.

Most of the system set-up parameters are under the menu selection **S**ystem. This covers things such as communications, security, analog ranges and filter settings, etc.

To view and adjust the I/O values of a module there are a series of sub-options under **I** for I/O monitoring. A typical sequence required to perform a local control or monitoring function is as follows:

1. select the function required - simulate input, set an output, etc.
2. select the channel/channels - n, n-m, n,m,o or A for all
3. enter the values required.

These steps are described in detail in the following pages.

Channel Selection

Various diagnostic port menu selections will prompt the user to enter the channel or channels to act upon. This channel selection prompt occurs after the function to be performed has been specified. A channel selection can be made in any of the following ways:-
(n, m, o represent any valid channel numbers for that board)

n	a single channel
n-m	a range of channels
n,m,o	several individual channels
A	all channels.

Security

To restrict access to certain critical menu options and prevent undesirable tampering with critical parameters such as calibration factors, a security system consisting of three different access levels (low, medium and high) is available. The access level of the module runs at either low, medium or high and all menu options are assigned one of these levels. The access level of the module determines which menu options can be selected. When the module is turned on its access level defaults to low. The access level of the module can be raised to medium or high by entering the appropriate access code.

To select a menu item which only requires low level access the access level of the module can be set to anything. To select a menu item which requires medium level access the access level of the module must be set to medium (or high). To select a menu item which requires high level access the access level of the module must be set to high.

Security codes must be set up for medium and high level access before the security system provides this blocking. When shipped from the factory, the security system is disabled - there are no passwords assigned to the medium and high levels and the main menu will display "Access: Unrestricted".

See also the **Access Level** section later in this chapter.

Diagnostic Port Menu Map

Subscripts:

- **H** High level access required
- **M** Medium level access required
- ***** Master password required

Main Menu

- **D** Display
  - **S** Small Terminal
  - **T** Dumb Terminal
  - **P** Percentage Units
  - **V** Volts/Amps Units
  - **H** Hexadecimal Units
  - **D** Decimal Units
- **I** I/O Control
  - **[D]**$_M$ Disable - use local value for output
  - **[E]**$_M$ Enable - use network value for output
  - **[S]**$_M$ Simulated - use simulated input value
  - **[A]**$_M$ Actual - use actual input value
  - **[I]**$_M$ Input - input simulated value
  - **[O]**$_M$ Output - set value of an output
  - **[T]**$_M$ Toggles the state of last modified digital channels
  - **[C]**$_M$ Clear all simulated and disabled status
  - **[R]**$_M$ Reset PID Integral
  - **[H]**$_M$ Hold PID Integral
- **M** Monitor communications – received/sent message counts
- **S** System
  - **C** Communication
    - **M**$_H$ Select comm's Mode
    - **Z**$_H$ Zero outputs on comm's timeout
    - **T**$_H$ Select comm's Timeout value
    - **B**$_H$ Bus-off hold-off
  - **R** Range
    - **I**$_H$ Input Ranges
    - **O**$_H$ Output Range
  - **F** Analog Filter
    - **F**$_H$ Filter factor
    - **W**$_H$ Filter window
  - **P** PID settings
    - **C**$_H$ enable PID control mode
    - **N**$_H$ Invert PID
    - **R**$_H$ Reset PID using dig-in-5
    - **H**$_H$ Hold PID using dig-in-6
    - **P**$_H$ Proportional term
    - **I**$_H$ Integral term
    - **D**$_H$ Differential term
    - **U**$_H$ Upper limit on Integral
    - **L**$_H$ Lower limit on Integral
    - **W**$_H$ PD dead-zone window
  - **D** Digital channels
    - **I**$_H$ Input channel polarities
    - **O**$_H$ Output channel polarities
  - **T** Output zeroing on timeout
    - **Z**$_H$ Zero on timeout
    - **-**$_H$ No action on timeout
  - **E** EEPROM access
    - **C** Clear last reset code
  - **A** Access level
    - **H**$_H$ set High
    - **M**$_M$ set Medium
    - **L** set Low
    - **C**$_*$ Change passwords
      - **H** High level p/w
      - **M** Medium level p/w
  - **ZP**$_H$ Zero / restart processor

Main Menu

This screen shows the module software version number, the board category and serial number. These are followed by the current access level and the main menu options – which are:

D	**Display Options**
I	**Input / Output Control and Monitoring**
M	**Monitor Communications**
S	**System Configuration**

D Display Options

The display options menu is used to specify the type of terminal in use on the diagnostic port and to choose the units in which analog values are to be expressed.

Terminal Type

S	**Small Terminal**	32 characters x 16 rows
T	**Dumb Terminal**	

Units Selection

The diagnostic port can display analog values in one of several units. The selected units are used both for displaying analog values and also when the user is required to enter analog values.

P	**Percentage**	values are shown as a percentage of full scale
V	**Volts/Amps**	values are expressed as true voltages (or amps as appropriate)
H	**Hexadecimal**	values are shown as a hex number
D	**Decimal**	values are shown as a decimal number

The decimal option represents analog voltages/current as a decimal count of bit increments. The range is -32000 to 32000 for 16 bit bipolar values and 0 to 64000 for 16 bit unipolar values.

I Input / Output Control and Monitoring

Command Summary

[**D**]	Disable - output value entered by [O], not from network
[**E**]	Enable - output back under network control
[**S**]	Simulate - send simulated input value over network
[**A**]	Actual - send actual measured value over network
[**I**]	Input simulation - enter value to send over network
[**O**]	Output set - according to entered value, not from network
[**T**]	Toggle - digital outputs
[**C**]	Clear all simulated and disabled status
[**R**]	Reset PID Output and calculations
[**H**]	Hold PID Output and calculations

To aid in control system commissioning and testing, a facility for monitoring or simulating inputs and controlling outputs is available. It is sometimes useful to simulate "hardwire" an input to a known state for checking control system software operation. The outputs can also be controlled from the diagnostic port, and overwriting of outputs by data from the communication can be disabled.

Simulation of inputs and disabling of communication to outputs is accessible only in medium access level. Inadvertently leaving channels simulated or disabled can cause confusion - it appears to another operator as if the module is not working properly. To remind users that some I/O is simulated, a message reading “SIMULATED DATA” is displayed at the top of every menu screen until the channels are cleared of simulated data. Once simulation has been finished with, clear all simulations using [**C**].

Underneath the menu heading for this screen are several letters in square brackets, indicating key-press options.

[**D**] Changes the status of the chosen output(s) to **D**isabled and outputs the value entered using the [**O**] option, rather than the value from the network.

[**E**] Changes the status of the chosen output(s) to **E**nabled and outputs the value from the network, rather than the value entered using the [O] option.

[**S**] Changes the status of the chosen input(s) to **S**imulated and uses the value entered using the [**I**] option, rather than the actual measured value. [S] and [A] can be used to toggle between the actual and simulated values.

[**A**] Changes the status of the chosen input(s) to **A**ctual, and uses the actual measured value, rather than the simulated value to send over the network. [S] and [A] can be used to toggle between the actual and simulated values.

[I] Prompts the user to enter a value which is sent over the network to the control computer - automatically switching the status of the chosen input(s) to **S**imulated. The control computer receives this entered value as if it were a true reading. The entered value is held in a "simulation" register, and is kept for future use when the input is switched back to 'actual'.

[O] Prompts the user to enter a value which is then output - automatically switching the status of the chosen output(s) to **D**isabled. The channel will output the entered value regardless of what value the control computer is sending over the network.

[C] Clears all status flags - all inputs read the actual value (status: **A**ctual) and all outputs are controlled from the network (status: **E**nabled). The values entered using the [I] option remain stored, and can be used again by using the [S] option. Note that [C] only clears the Enabled/Disabled status, it does not clear the value of the output. An output may remain at the value set by the [O] option until a new packet of I/O data is sent over the network from the control computer.

The monitoring menu indicates with an **S** or **a** (digitals) (or the text **Simulate** or **Actual**) (analogs) as to whether simulated or actual values are being used as the input values to send over the network. Similarly for outputs the display reads **D** or **e** (digitals) (or the text **Disabled** or **Enabled**) (analog) to indicate whether or not an output is disabled from network control and using the locally entered value.

Digital channels

[T] Toggles the states of the last selected digital output channels **D**isabled for simulation. Channels that were OFF turn ON, or those that were ON turn OFF.

[O] Under the [O] option for digital output channels there is the facility to have the output(s) perform a **R**epeat toggle function. The selected channel(s) will repeatedly turn On then Off with a one second period. Press any key to end the repeat toggle. This is to facilitate the checking of the operation of mechanical devices like relays and solenoids, or for general tracing of wiring etc.

PID functions

These two items appear if PID control mode is enabled.

[R] Resets the PID, and sets the control output to zero. Pressing [R] again, or [C] reverts to normal PID mode.
This mimics the action of Digital-input-5 if it is enabled to function as PID reset. Dig-in-5 is simulated and set to value 1.

[H] Puts the PID in Hold mode and causes the control output to be held at its current value. Pressing [H] again, or [C] reverts to normal PID mode.
This mimics the action of Digital-input-6 if it is enabled to function as PID hold. Dig-in-6 is simulated and set to value 1.

A status line on the screen indicates the current mode: Normal / Held / Reset.
If in both reset and hold, reset mode overrides.

M Monitor Communications

Displays a running total of the number of messages received / replied-to that were addressed to this module. This can be a useful indicator that the network is running at correct speed. The counts can be reset to zero by pressing **C**.

S System Configuration

System configuration allows the selection of different communications parameters, analog input and output voltage ranges, analog input filtering, PID parameters, diagnostic port access levels and more. All system configurations can be viewed by anyone but changes can be made only with high level access.

C Communication

A summary of communication mode, communication time out period, the timeout (enabled/disabled) state of output channel zeroing and the module address and baud rate as set by the rotary switches is displayed.

M Select Communication Mode

Only one mode, DeviceNet, is available to the user.

Z Zero Outputs on Communication Timeout

Enabling this feature causes the specified output channels to be forced to zero if a communication timeout is detected. Pressing Z toggles this feature on and off. See also the **T** option under the **S**ystem Configuration menu for specifying which channels are zeroed.

T Select Communication Timeout Value

The timeout value can be entered in units of milliseconds, where the timeout period must be greater than 20 ms and less than 65535 ms (65 seconds).

B Select Bus-off Hold-off Value

This setting is useful in high noise installations to stop the DNA entering the "Bus-Off" condition until the programmed time has elapsed. Time is in units of 100ms, and the number must be between 0 and 255. If a value of 0 is entered, the module will function as per the DeviceNet specification, with the module entering "bus-off" after a time based on the expected packet rate, if communications errors still exist. A value of 1 will extend this time by 100ms; a value of 255 will extend this time by 25½ seconds.

R Input / Output Range

Each analog channel has a configurable voltage range. A summary of these is displayed along with options for changing the input and output ranges. A comprehensive lists of the available ranges can be found in sections 1 of this manual.

I Input Range

O Output Range

F Analog Input Filter

The module software features a non-linear filtering algorithm which can combine fast tracking of large, rapid input changes with heavy filtering of slowly changing signals. Through the diagnostic port the user can set the filtering time constant and the window within which the filtering is active. The action of the filter is as follows:

While the raw input signal remains within a window width of the previous filtered value, the reported value is the updated filtered value. However, if the raw input signal rapidly changes by more than a window width, the filtering is temporarily suspended, and the new reported value will be the actual instantaneous value, not a time average. In this way the reported value can be heavily filtered to eliminate small random noise, but still quickly track a large, rapid input change.

F Filter Factor

After selecting a channel the user can then select a filter time constant. Note that if the filter window width is set to 0 then no time averaged filtering will take place, no matter what value this filter constant is set to.

W Filter Window

Enter the voltage by which the input must rapidly deviate from the running average value before filtering is disabled.

A window width of 0 will effectively disable filtering, and the reported value will always reflect the instantaneous measured value.

If the non-linear window filtering is not required, but ordinary time averaged filtering is wanted, then choose a window width equal to the full-scale range of the channel.

P PID Settings

The module can be set to run in PID (closed loop) mode.

In non-PID mode, the analog output is taken from the network and reproduced on the hardware as requested.

In PID mode, the module uses analog input channel 0 as the feedback input, and the analog output is altered by the module itself to achieve a desired value on the feedback input, no matter what external drifts and influences there are. This is closed loop control. All the control computer has to do is send the module the desired input value - the setpoint. The module then alters the analog output so as to achieve the new desired input value, as fast as possible and with minimum overshoot.

PID refers to **P**roportional, **I**ntegral and **D**ifferential loop coeficients stored in the module. These can be adjusted by the user to achieve optimal control of the loop. Tuning of the loop coefficients can be a complicated process and guidance on this topic is given in section 2 of this manual.

If the module is set to operate in PID mode, then the user can optionally enable certain digital inputs for special uses - to reset or hold the PID algorithm when they are asserted.

C use PID Control mode

Pressing C will alternately enable or disable the use of PID.

N invert PID

Can be used to invert the sense of the PID algorithm. Sometimes a PID loop will require an inverting function - for instance if the feedback sensor is one where an increase in the sensed variable results in a decrease in the sensor output. The default is “disabled” - only enable with caution - wrong use will result in the system locking up at full scale output!

Pressing N will alternately enable or disable the invert function.

R Dig In 5 Resets PID

Digital input 5 can be used to control the PID algorithm. It can be used to reset the PID if, for instance, the device being controlled is not switched on. Feeding a “power on” signal into DigIn5, and enabling the reset function would prevent the build up of a large, and erroneous, integral term if the device being controlled was actually switched off.

H Dig In 6 Holds PID

Digital input 6 can be used to control the PID algorithm. It can be enabled to act as a “Hold” control - the actual analog output is held constant at its current output value while DigIn6 is switched on.

P Proportional term

Enables the user to enter the proportional term for use in the PID algorithm.

I Integral term

Enables the user to enter the integral term for use in the PID algorithm.

D Differential term

Enables the user to enter the differential term for use in the PID algorithm.

U Upper limit on Integral

Enables the user to set the upper limit beyond which the integral term will not be allowed to climb. Placing a limit on the integral value allows for quicker recovery when control is regained - the PID calculations do not have to spend a long time reducing an extremely large accumulated integral after a positive step input.

L Lower limit on Integral

Enables the user to set the lower limit below which the integral term will not be allowed to pass. Placing a limit on the integral value allows for quicker recovery when control is regained - the PID calculations do not have to spend a long time reducing an extremely large accumulated integral after a falling step has been programmed.

If the upper integral limit has the same value as the lower integral limit then both limits are disabled.

W PD dead-zone Window

This feature sets a window, about the set-point, within which the P and D terms are not used in the loop calculations. The function becomes Integral-only when the sensor input is within a window width of the set point.

This feature can be useful to smooth the output once the system has attained the setpoint.

For example, in a system where P is a large number, a single bit of flicker on the input sensor, when multiplied by the P term, could lead to many bits of flicker on the output.

If the PD dead-zone window is set to 0, this feature is disabled.

D Digital Channels

Each digital channel can be configured individually for either active high or active low logic.

I Input channels

Select the input channel(s) to be configured, then
H for active High or **L** for active Low (L is the factory default).

O Output channels

Select the output channel(s) to be configured, then
H for active High or **L** for active Low, (L is the factory default).

T Output Zeroing On Timeout

The analog output and each digital output channel can be configured individually to be forced to zero if a communication timeout is detected. See also the **Z** option under the **C**ommunication sub-menu.

Z Zero on Timeout

Select the output channel(s) to be zeroed on timeout.

- No Action on Timeout

Select the output channel(s) **not** to be zeroed on timeout.

E EEPROM Access

This screen displays a number called the last reset code. If this code is non-zero it represents a particular mode of memory corruption and indicates that the module was forced to reset by its internal self checking program. Memory corruptions are cured on reset and such resets are almost always the result of an unusually powerful electrical transient in which case this code is likely to change each time the module is reset.

If resets persist and the code changes each time then the electrical isolation and shielding of the module should be examined.

If resets persist and the code remains the same each time then consult the factory, quoting this code.

C Clear reset code in EEPROM

This option clears the value of the reset code, which is stored in EEPROM. If the user wishes to monitor this code over time it will be necessary to perform this operation each time a non-zero reset code is detected.

A Access Level

The access level feature restricts the use of certain critical menu commands to those persons with appropriate authorisation. On power up the default access level for the module is “low” but it can be set to any level through this menu. Additionally, if a user attempts an operation that requires a security level which is higher than the current access level of the module then the user is prompted for the relevant access code.

High and medium level codes are configured using the **C** sub-menu command which first prompts for the master access code to be entered. The master access code is provided in an envelope shipped with the module.

If the high and medium level access codes are set to zero then the security features are disabled and anyone has unrestricted access to all functions. This is convenient in the initial stages of setting up a system to avoid having to log in each time to make adjustments. However it is recommended that security codes be established once installation is complete.

Having security enabled (non-zero access codes) also adds to the robustness of the system in the event of electrical disturbance. If a spark disrupts the program and the internal self checking processes find the program in an area, randomly doing things that require an access code, but the security level has not been granted, then the module will reset itself. If access codes are left at zero this checking can not take place.

H set High Level

The high level access code is required to lift the module access to this level.

M set Medium Level

The medium (or high) level access code is required to lift the module access to this level.
No access code is required to reduce the module access to this level.

L set Low Level

This is the default level on power up.
No access code is required to reduce the module access to this level.

C Change Passwords

This menu option prompts for the master access code to be entered. Its sole purpose is to change the high and medium level access codes.

H High Level p/w

Set the high level password.

M Medium Level p/w

Set the medium level password.

ZP Zero / Restart Processor

This sub command of the **S**ystem menu forces the module to restart itself, without removing power.

4 Installation / Commissioning / Fault Finding

Installation

1) **Position the Modules as close to the controlled points as possible**.

The main aim should be to keep all signal wires as short as possible, to minimise the pick up of transients. This applies particularly to the analog I/O which are not isolated from the main module processor. Therefore site the controller module so that the wiring of the analog channels (inputs and output) is as short as possible. Mount the DI behind the power supplies etc, right beside the control inputs.
Digital channels on the other hand are isolated by opto-couplers and relays, and so are a less likely entry point for noise.
The stainless steel backplates of the module clips onto a 100 mm length of standard 35mm DIN rail. If the rail is steel, then generally the module case is automatically grounded by the backplate. If aluminium DIN rails are used then frequently the anodised finish will effectively insulate the backplate, and adequate grounding will not be automatic. In these cases run a short grounding lead from under one of the screws on the backplate to a nearby chassis ground point.
In almost all circumstances it is a good idea to have the case of the module grounded. However, very occasionally this has caused problems. In these rare circumstances it may be necessary to isolate the DIN rail from the plate it is mounted on.

The modules and signal wiring should really be within an enclosure of some sort - a rack enclosure or something similar. In high voltage machinery there can be very fast, massive changes in potential when an arc-over or breakdown occurs. This can easily capacitively couple through to exposed signal wiring. If the wiring cannot be enclosed, place shield plates to form a Faraday cage to screen the wiring from any such rapid capacitively coupled noise.

2) **Check the signal levels**
Check that the signal levels are within the allowable limits, as given in the specifications. If necessary use resistive dividers on the SCA or SCM I/O board to bring the signals within allowed voltage limits. It is easier to insert the appropriate resistors before wiring up all the signals.

3) **Connect up the modules to the controlled equipment**.
Make up the signal wiring using shielded cables.

Use twisted shielded wires for all I/O wiring.
Terminate the shields at the module terminals only, not at the other end of the signal wires.

Do not run signal wires alongside high current, high voltage or high frequency cables.

4) **Apply power to the module.**
The power connector is located on the end of the module. A suitable cable connector is supplied with each unit. Power is low-voltage ac or dc (see specs.) and can be reticulated to several modules in parallel without causing interaction or ground loop problems. Model PS24D15 power supply is a convenient power source for modules.

5) **Configure the module**

Each module must be set to its own unique address on the network, using the address switches on the front panel. Set the correct baud rate for the network.
Use the diagnostic port to configure the module. Go through the system configuration menu, as described in section 3 of this manual.

Check polarities
Check analog range settings, filter and filter windows
Record these setup parameters.

Return to the main menu (press 'X' a few times) and then press 'I' to go to the I/O control and monitoring menu. The I/O values should be displayed - check that they have sensible values. At this stage outputs can be set to values using the [O] option.

BUT **be careful**!! outputs will be set, supplies could be turned on, motors start etc. Always be safety conscious and check with all other personnel working on the installation before altering anything.

Commissioning & Fault Finding

Firstly a word of caution, without wanting to sound too depressing:-

On a complex hardware and software system it is extremely unlikely that everything will work 100% first time.

Be prepared for some time to be spent commissioning and debugging the entire system. There are likely to be software mistakes (such as defining an analog channel in the software package as one range when the module is set for another) and hardware mistakes (for instance, a shield wire not grounded, thereby allowing transients in to upset one of the modules.)

Here are some stages in the commissioning process, with things to try if all is not working correctly.

Power up all the modules on the network. Start the control software. Keep things simple at the start and do not turn on any high voltages on the machine just yet.

Communications

Make sure that the network is communicating correctly.
Both LEDs - network status (NS) and module status (MS) on the front of every module on the network should appear continuously lit green.

If the LEDs are not continually lit, check:

- modules all have auxilliary power on - the LEDs on the front of each module should be lit in some manner - red or green or flashing red or green.

- modules configured correctly - check baud rate and address switch settings ,

Software

Check the software configuration and setup in the control computer:

- is the main software program set to use the correct application directory?
  sometimes there are a number of directory settings that have to be altered if changing over to another application.

- do the definitions of the modules in the software setup correspond with the actual modules? - check module address switches.

Sometimes software might not get initialised properly, or gets upset if things are not all done in the right order. If all else fails, turn off the computer and start it up again.

Communications on the network must be established before going on to further tests.

If you can't get a complex network going, start with just one module powered up on the bench. This allows you to confirm that the network is being initialised properly, and that communications can be established with each module singly in turn.
Make a separate small test application (in another directory!) to allow you to check your setup and configurations.
Use this method of dividing off the problem into small manageable parts. Complex systems of hardware and software are very confusing and it is easy to spend a long time trying to track down a problem in the system as a whole.

It may take a little time to make a small test application in the software package, but it will probably save a lot of time in the long run.

Once good communications are established you can move on to checking the function and operation of the control software.

Check that changes to inputs are reflected on the screen of the control computer.

Use the diagnostic port features at the module to check what the module is actually reading. The fault may be with the sensor or wiring, rather than with the module setup or software. Always check at the module level with the diagnostic port before doing much else. Use the I - Monitor Inputs/Outputs feature.
Also use the [I] function to simulate known input values. This makes it easier to check your control software readouts are responding correctly.

For analog channels:

If the objects on the screen exhibit too much or not enough travel then check the range settings in the module, and check the scaling factors used in defining the point in the control software.

Some control programs can get temporarily overloaded if a lot of channels all change at once. Unless great care is taken in installation the 16 bit analog inputs are bound to have some noise and flutter on them. This can unnecessarily overload the system, particularly if they are measuring slow changing parameters such as temperatures or magnet currents. For these channels, and in general unless the full speed accuracy is required, use the filtering within the module , and use filtering (or a deadband system) within the control software package. These should reduce unnecessary overhead for the control software.

For digital channels:

Check the definitions of polarity for the channel, both in the module and in the control software.

Once all channels are operating correctly turn on the high voltage supplies.

This will probably unleash a new set of problems.
Because each and every accelerator is so different the operation of a control system is only specific to one machine, and the operational reliability is critically dependent on the installation practice followed.

Switching on the high voltage will most probably uncover some installation deficiencies.

Noise problems will most probably be indicated by one or more modules performing a reset when there is arcing or a discharge from the high voltage sections. The reset is evident by the LEDs on the front of the module blinking for a second or so, and possibly a message on the control screen saying "network error" or similar.

If this occurs check:

- the shields of the signal wiring

- the grounding of the supplies within the accelerator

- the grounding of the module case. Note that in general the case should be grounded, but it has been seen in some installations that this causes problems. Try disconnecting the grounding of the case.

If it is a complex network of modules try to isolate off one or two modules at a time, and run them on a network of their own for a while, to see if they are the source of the resets.

At this point it is sensible to try to create some repeatable arc over, to create known noise source. Using wire or pipe, create a spark gap bridging the extraction voltage, or the acceleration voltage. Adjust the gap and the voltage so the arcs occur every second or so. This gives some repeatable noise source from which some statistics can be built up, so that modifications to the installation can be sensibly evaluated.

Here are some further simple things to try, which have proved effective in some installations.

Run the low voltage auxiliary power lead to the module through a ferrite toroid for a few turns, just before it enters the module .

Use ferrites on signal cables

Place shielding plates over the modules and any exposed wiring to screen them from capacitively coupled transients.
The modules and signal wiring should really be within an enclosure of some sort - a rack enclosure or something similar. In high voltage machinery there can be very fast, massive changes in potential when an arc-over or breakdown occurs. This can easily capacitively couple through to exposed signal wiring. If the wiring cannot be enclosed, place shield plates to form a Faraday cage to screen the wiring from any such rapid capacitively coupled noise.

On very powerful pulsed or unusually noisy machines it may be necessary to enclose all the control electronics in a shielded screened cabinet, and use feed through capacitors to bring any outside signal wiring into the system.

5 DeviceNet Communications

The full Device Profile for this product is bound in to this manual as Appendix A.
This section will give some more information on the nature and use of the various pieces of data that are sent to and received from the module.

I/O Assembly Data Attribute Format

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	0	Analog Output / Setpoint for PID control - LSB							
	1	Analog Output / Setpoint for PID control - MSB							
Outputs sent to module	2	Digital Output 7	Digital Output 6	Digital Output 5	Digital Output 4	Digital Output 3	Digital Output 2	Digital Output 1	Digital Output 0
	3	Spare	Spare	Spare	Spare	Spare	Spare	PID Output Hold (0:not held)	PID Integ' Reset (0:not Reset)
2	0	Analog Input #0 LSB							
	1	Analog Input #0 MSB							
inputs	2	Analog Input #1 LSB							
read	3	Analog Input #1 MSB							
from module	4	Digital Input 7	Digital Input 6	Digital Input 5	Digital Input 4	Digital Input 3	Digital Input 2	Digital Input 1	Digital Input 0
	5	Spare	Spare	Board needs calibrating (1)	One or more simulations active (1)	PID upper or lower limit reached (1)	ADC chan-1 over / under-flow (1)	ADC chan-0 over / under-flow (1)	DAC over / under-flow (1)

Analog Output / Setpoint for PID control

This location in the message packet has two functions, depending on whether the module is running PID control or not. If the module is used as a simple I/O module then this location is used to send the value to generate the output voltage or current directly. If the module is set to run PID control, then this location is used to send the value of the setpoint for the control algorithm.

If you want to deal in counts, not volts, you can put the display of the Group3 diagnostic port into the appropriate mode - under the Display Options choose Decimal to display the analog quantities as decimal number counts.

Analog Output

This is a 16 bit value, with the range depending on the output type selected.

Voltage output:
- bipolar : -32,000 at negative full scale (-10V) to +32,000 at positive full scale (+10V)
- unipolar : 0 at 0V to +64,000 at full scale (+10V)

Current output: 0 at lowest current (4mA) to +64,000 at the maximum current (20mA).

Note that the numerical range is somewhat less than the maximum allowed by 16 bits - the range sent over DeviceNet to this module is 64,000 counts, not the full 65,536. This gives the module processor some "headroom" to apply calibration coefficients without causing a 16 bit overflow. The two byte number is sent least significant byte first.

Therefore, to set the analog output to the value required, send the appropriate number of counts, according to the following:

For bipolar voltage output counts sent $= \frac{\text{voltage required}}{10} \times 32{,}000$

For unipolar voltage output counts sent $= \frac{\text{voltage required}}{10} \times 64{,}000$

Normally a control package will sort out this scaling for you.
Note that negative numbers are represented as 2's complement values.

For current output counts sent $= \frac{(\text{mA required} - 4)}{16} \times 64{,}000$

Setpoint for PID control

If the module is set to operate as a PID controller then you do not send the output value directly. The value the DeviceNet application sends to the controller is the value that you want the feedback input (Analog input0) to be reading. i.e. although you are sending what appears to be an output value, the module actually uses it as a desired input value, and does everything possible (by altering its DAC output) to make its readback (analog Ch0) equal the value you sent.

What you send to the controller is the value you desire to see at input 0.

As an example, suppose you have a transducer probe attached to analog input channel 0 that returns a 0 to 10 volt signal. If you want the controlled variable to be half full scale, what you are going to send in the DeviceNet message packet is a request for 5.000V, as seen at analog input0.

Now the messaging is done in numbers of counts where 10V equates to 32000 counts, so asking for an input of 5.000V means you have to send a value of 16000, in the message location corresponding to the Analog Output / Setpoint data. (Bytes 0 and 1 of the message).

The controller then alters its actual DAC output in whatever way it can to achieve the value 5.000V at its input channel0. You have no direct control over the DAC output in PID mode.

So, to state it again, what you send in the output data / setpoint location is the the value you wish to see at the probe input channel.(analog input 0)

Digital Outputs

The eight digital outputs are controlled by one byte in the DeviceNet message sent to the module. Each bit in the byte corresponds to a channel.
The polarity of the bits is programmable on a channel by channel basis by using the Group3 diagnostic port - under the **System / Digital/Output Polarities** selection. The default setting for polarities is LOW but can be changed to achieve an inversion. If the polarity is selected as LOW, a 1 in a bit location will cause the corresponding channel's relay contact to close.

PID Control Byte

Reset
Bit 0 is used to zero the control output and clear all PID calculations.
If bit0 is set to 1 then all of the following occur:
- the controller output is set to zero.
- the integral accumulation is cleared to zero.
- all PID calculations are stopped.

These actions can also be invoked by any one of the following:-
Digital-input-5, if it is enabled to function as PID reset.
Pressing [R] on the diagnostic port I/O monitoring menu.
Simulating Digital-input-5 , and setting it to value 1.

Hold
Bit 1 is used to hold the output and PID calculations
If bit1 is set to 1 then all of the following occur:
- the controller output is held at its last value.
- the integral accumulation is halted.

These actions can also be invoked by any one of the following:-
Digital-input-6, if it is enabled to function as PID hold.
Pressing [H] on the diagnostic port I/O monitoring menu.
Simulating Digital-input-6 , and setting it to value 1.

Analog Inputs

Channel 0 and channel 1 are identical.
The values read from these locations are 16 bit numbers, within the range
bipolar units: -32,000 at negative full scale to +32,000 at positive full scale.
unipolar units: 0 at zero, +64,000 at full scale.
Full scale is either 100mV, or 10V, depending on the range selected.

Note that the numerical range is somewhat less than the maximum allowed by 16 bits - the range sent by this module over DeviceNet is 64,000 counts, not the full 65,536. This gives the module processor some "headroom" to apply calibration coefficients without causing a 16 bit overflow. The two byte numbers are sent least significant byte first.

Therefore, to read an analog input, read the number of counts from the incoming message, and scale it to the appropriate input range setting.

For ±10V input range $$\text{voltage} = \frac{\text{counts read}}{32{,}000} \times 10\ \text{V}$$

For 0 to 10V input range $$\text{voltage} = \frac{\text{counts read}}{64{,}000} \times 10\ \text{V}$$

For ±100mV input range $$\text{voltage} = \frac{\text{counts read}}{32{,}000} \times 100\ \text{mV}$$

Normally a control package will sort out this scaling for you.
Note that negative numbers are represented as 2's complement values.

If you want to deal in counts, not volts, you can put the display of the Group3 diagnostic port into the appropriate mode - under the Display Options choose Decimal to display the analog quantities as decimal number counts.

Digital Inputs

The eight digital inputs are read as one byte in the DeviceNet message sent by the module. Each bit in the byte corresponds to a channel.
The polarity of the bits is programmable on a channel by channel basis by using the Group3 diagnostic port - under the **System / Digital/Input Polarities** selection. The default setting for polarities is LOW but can be changed to achieve an inversion. If the polarity is selected as LOW, and if current is flowing in an input opto-coupler, a 1 will be read in the bit location corresponding to that channel.

Module Status Byte

This byte is returned as part of the message. Various bits within the byte are used to indicate certain conditions that a controlling application should know about.
All bits indicate their described status if set. This enables an application program to test the value of the entire byte, and proceed no further if it is zero.

DAC over or **under flow**

Bit 0 is used to indicate to the application program that a number was passed to the module which, after application of the calibration coefficients, produced a number which could not be represented with 16-bits, i.e. outside the range -32768 to +32767 for bipolar or 0 to +65535 for unipolar.
The module adjusted the resulting value to the most +ve or -ve value possible (as appropriate) and set this overflow/underflow status flag.
Note that if the module is operating in PID mode, this status flag is not necessarily indicative of a fault condition . If the required output range is within the capability of the module then momentary attempts to produce analog output values outside the boundaries described above are quite normal.

Channel 0 and Channel 1 ADC over or **under flow**

bit 1 overflow / underflow: analog input channel 0
bit 2 overflow / underflow: analog input channel 1

If one of these bits is read as a 1, it means that a number outside the range -32000 to +32000 (bipolar) or 0 to +64000 (unipolar) was generated by the analog input componentry on the module. It doesn't necessarily mean that the reading is wrong, but that it is outside the specified range of the product.

PID calculation overflow

Bit 3 indicates that in calculating the PID output a number was created that exceeded the specified limits.
If the PID integral value exceeds the upper or lower limit, as configured via the diagnostic port terminal, the integral is clamped to either the upper or lower limit (as appropriate) and bit 3 set.
If the PID overall analog output correction signal lies outside the range -32768 to +32767 (bipolar) or 0 to +65535 (unipolar) then the signal is adjusted to the appropriate limit and again status bit 3 is set.

Note that if the module is operating in PID mode, this status flag is not necessarily indicative of a fault condition . If the required output range is within the capability of the module then momentary attempts to produce analog output values outside the boundaries described above are quite normal.

Simulations active

Bit 4 indicates that one or more simulations are active.
If this bit is set it means that someone, via the diagnostic port terminal, has caused the module to simulate one or more of its input or output values. This is important information to the control application, as any simulations could mislead an application and prevent normal control.

Board needs calibrating

Bit 5 indicates that the board needs calibrating
This bit will be set if the calibration data is deliberately cleared within the module.

Appendix A

DNA - ANALOG & DIGITAL I/O
DEVICE PROFILE

24 April 2001

Table of Contents

1 Introduction

This document describes the structure and operation of the Group3 DeviceNet DNA product.

2 Object Model

The Object Model in Figure 1 represents the DeviceNet software library structure.

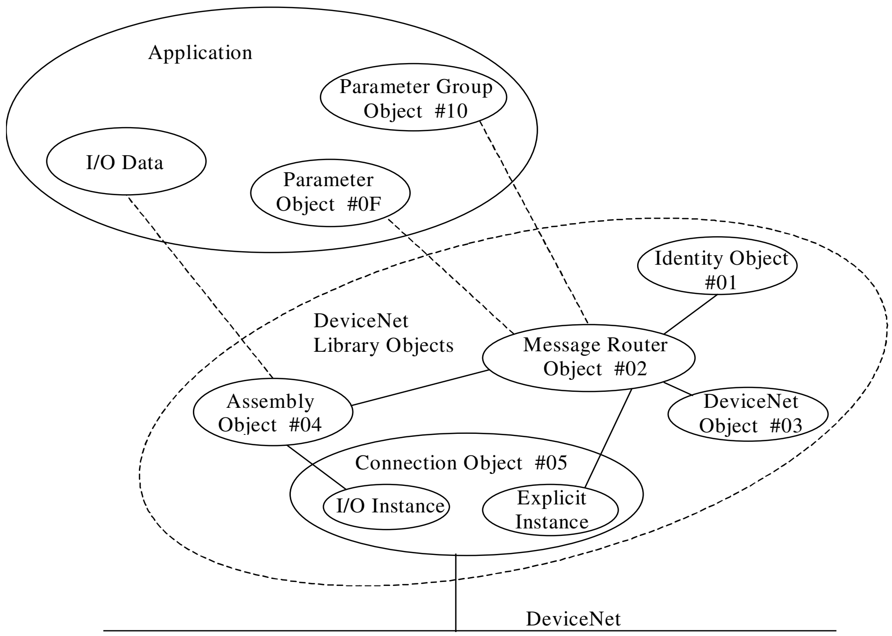


Figure 1 - DeviceNet Library Object Map

The table below indicates:

- The object classes present
- The class ID
- The number of instances present in each class

Object Class:	Class ID:	# of Instances:
Identity Object	01hex	1
Message Router Object	02hex	1
DeviceNet Object	03hex	1
Assembly Object	04hex	1
Connection Object	05hex	2
Parameter Object	0Fhex	89
Parameter Group Object	10hex	9

3 DeviceNet Communication Object Classes

The DeviceNet Communications Objects manage and provide the run-time exchange of messages. The *Services, Attributes,* and *Behaviours* associated with the Communication Objects are detailed in this chapter.

3.1 IDENTITY OBJECT

Class ID: 01hex

3.1.1 Class Attributes

Class only attributes are not supported within the Identity Object of the library.

3.1.2 Instance Attributes

The following table shows the identity object instance attributes:

Instance:	Attribute:	Access:	Name:	Data Type:	Description:
1	1	GET	Vendor ID	UINT	Identification of each vendor by number
1	2	GET	Device Type	UINT	Identification of general type of product
1	3	GET	Product Code	UINT	Vendor product identification code
1	4	GET	Revision Major Revision Minor Revision	STRUCT of: USINT USINT	Product revision code
1	5	GET	Status	WORD	Summary status of device
1	6	GET	Serial Number	UDINT	Serial number of the device
1	7	GET	Product Name	SHORT_STRING	Product Name String *(max. 32 characters)*

3.1.3 Common Services

The Identity Object provides the following Common Services:

Service Code	Provided in Implementation		Service Name	Description of Service
	Class	Instance		
0Ehex	No	Yes	Get_Attribute_Single	Returns the contents of the Specified attribute.
05hex	Yes	Yes	Reset	Invokes the Reset service for the device.

3.2 Message Router Object

Class ID: 02hex

The Message Router Object has no publicly accessible attributes and therefore supports no services.

3.3 DeviceNet Object

Class ID: 03hex

3.3.1 Class Attributes

Attribute:	Access:	Name:	Data Type:	Description:
1	GET	Revision	UINT	Revision of the DeviceNet Object upon which the implementation is based (*fixed at 0002 hex*)

3.3.2 Instance Attributes

The following table shows the identity object instance attributes:

Instance:	Attribute:	Access:	Name:	Data Type:	Description:
1	1	GET	MAC ID	USINT	DeviceNet node address *(0 – 63)*
1	2	GET	Baud Rate	USINT	DeviceNet baud rate *(0 – 2)*
1	3	GET	BOI	BOOL	Buss-off interrupt action
1	4	GET/SET	Buss-Off Counter	USINT	Number of times CAN went to the bus-off state *(0 – 255)*
1	5	GET	Allocation information: Allocation choice byte masters MAC ID	STRUCT of: BYTE USINT	*See DeviceNet specs, Vol II, section 5-5.4.2* MAC ID of Master *(0 – 63, 255)*

3.3.3 Common Services

The DeviceNet Object provides the following Common Services:

Service Code	Provided in Implementation		Service Name	Description of Service
	Class	Instance		
0Ehex	Yes	Yes	Get_Attribute_Single	Used to read a DeviceNet Object attribute value.
10hex	No	Yes	Set_Attribute_Single	Used to modify a DeviceNet Object attribute value.

3.4 ASSEMBLY OBJECT

Class ID: 04hex

3.4.1 Class Attributes

Class only attributes are not supported within the Assembly Object of the library.

3.4.2 Instances

Two instances exist within the Assembly Object, one output instance and one input instance.

Instance:		Type:	Name:
Decimal	Hex		
1	01	Output	Consumption I/O Data
2	02	Input	Production I/O Data

3.4.3 Instance Attributes

Access is restricted to the following attributes:

Instance:	Attribute:	Access:	Name:	Data Type:	Description:
1,2	3	GET/SET	Data	ARRAY	Access the data of the specified instance

3.4.4 Common Services

The Assembly Object provides the following Common Services:

Service Code	Provided in Implementation		Service Name	Description of Service
	Class	Instance		
0Ehex	No	Yes	Get_Attribute_Single	Used to read a DeviceNet Object attribute value.
10hex	No	Yes	Set_Attribute_Single	Used to modify a DeviceNet Object attribute value.

3.4.5 I/O Assembly Data Attribute Format   format shown below

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	0	Analogue Output / Setpoint for PID control - LSB							
	1	Analogue Output / Setpoint for PID control - MSB							
	2	Digital Output 7	Digital Output 6	Digital Output 5	Digital Output 4	Digital Output 3	Digital Output 2	Digital Output 1	Digital Output 0
	3	Spare	Spare	Spare	Spare	Spare	Spare	PID Output Hold (0:not held)	PID Integ' Reset (0:not Reset)
2	0	Analogue Input #0 LSB							
	1	Analogue Input #0 MSB							
	2	Analogue Input #1 LSB							
	3	Analogue Input #1 MSB							
	4	Digital Input 7	Digital Input 6	Digital Input 5	Digital Input 4	Digital Input 3	Digital Input 2	Digital Input 1	Digital Input 0
	5	Spare	Spare	Board needs calibrating (1)	One or more simulations active (1)	PID upper or lower limit reached (1)	ADC chan-1 over / under-flow (1)	ADC chan-0 over / under-flow (1)	DAC over / under-flow (1)

3.5 CONNECTION OBJECT

Class ID: 05hex

3.5.1 Class Attributes

Class only attributes are not supported within the Connection Object of the library.

3.5.2 Instances

Two instances of the connection object exist within the library, one is the explicit messaging instance and one is the assembly I/O instance.

Instance:		Type:	Name:
Decimal	Hex		
1	01	Explicit	Explicit Messaging Connection Port
2	02	I/O	Assembly I/O

3.5.3 Instance Attributes

Access is restricted to the following attributes:

Instance:	Attribute:	Access:	Name:	Data Type:	Description:
1,2	1	GET	State	USINT	State of the object
1,2	2	GET	Instance Type	USINT	Indicates either I/O or Messaging connection *(00hex = Explicit, 01hex = I/O)*
1,2	3	GET	Transport Class Trigger	BYTE	Defines the behaviour of the connection *(83hex = Explicit, 82hex = I/O)*
1,2	4	GET	Produced Connection ID	UINT	Placed in the CAN identifier field when the connection transmits
1,2	5	GET	Consumed Connection ID	UINT	CAN identifier that denotes the message to be received
1,2	6	GET	Initial Comms Characteristics	BYTE	Defines the Message Group(s) across which production and consumption occurs *(21hex = Explicit, 01hex = I/O)*
1,2	7	GET	Produced Connection Size	UINT	Maximum number of bytes transmitted across this connection
1,2	8	GET	Consumed Connection Size	UINT	Maximum number of bytes received across this connection
1,2	9	GET/SET	Expected Packet Rate	UINT	Defines the timing associated with this connection
1,2	12	GET	Watchdog Timeout Action	USINT	Defines how to handle Inactivity/Watchdog timeouts
1,2	13	GET	Produced Connection Path Length	UINT	Number of bytes in the produced connection path attribute
1,2	14	GET	Produced Connection Path	Array of USINT	Specifies the application object(s) whose data is to be produced by this connection object
1,2	15	GET	Consumed Connection Path Length	UINT	Number of bytes in the consumed connection path attribute
1,2	16	GET	Consumed Connection Path	Array of USINT	Specifies the application object(s) that are to receive the data consumed by this connection object

3.5.4 Common Services

The Connection Object provides the following Common Services:

Service Code	Provided in Implementation		Service Name	Description of Service
	Class	Instance		
0Ehex	No	Yes	Get_Attribute_Single	Used to read an attribute value.
10hex	No	Yes	Set_Attribute_Single	Used to modify an attribute value.
05hex	No	Yes	Reset	Used to reset the Inactivity/Watchdog Timer associated with a Connection Object. When a Connection in the Timed Out or Deferred Delete state receives a Reset request it also transitions back to the Established state.
09hex	Yes	Yes	Delete	Used to delete a Connection Object and to release all associated resources.

Appendix B

DNA - ANALOG & DIGITAL I/O

Parameter Object

24 January 2003

Group3 DNA - Parameter Object Instances

Instance	Name	Help Text	Settable	Data Type	Data Size	Min. Value	Max. Value	Default Value	Supplementary Enumeration Help Strings
Group3 specific information		(Parameter Group Object Instance #1)							
1	mnfc' name	manufacturer name		SHORT_STRING	22				
2	G3 h/w name	Group3 hardware name		SHORT_STRING	3				
3	G3 s/w ver	Group3 s/w version		SHORT_STRING	5				
4	G3 board type	Group3 board type		USINT	1	0	255	201	
5	G3 board cat	Group3 board category		USINT	1	1	3	1	e_category
6	(reserved)	reserved parameter		USINT	1			0	
DeviceNet specific information		(Parameter Group Object Instance #2)							
7	vendor ID	vendor ID		UINT	2	0	65535	233	
8	device type	device type		UINT	2	0	65535	0	
9	major rev no	major revision number		USINT	1	0	255	0	
10	minor rev no	minor revision number		USINT	1	0	255	0	
11	product name	product name		SHORT_STRING	11				
12	product code	product code		UINT	2	0	65535	0	
13	serial no	board serial number	Y	UINT	2	0	65535	0	
diagnostic information		(Parameter Group Object Instance #3)							
14	last reset	last reset code	Y	UINT	2	0	65535	0	
15	bad sw setting	bad MAC-ID or baud-rate switch setting		USINT	1	0	1	0	e_sw_OK
16	diag out	send a diagnostic char to DNA	Y	SINT	1	0	127	0	
17	diag in count	count of ready diagnostic char's at DNA		USINT	1	0	29	0	
18	diag in	get ready diagnostic char's from DNA		SHORT_STRING	30				
19	start mode	soft/hard startup	Y	USINT	1	0	1	0	e_start_mode
20	priv' access	high or medium level access enabled	Y	USINT	1	0	1	0	e_dis_en
21	bus-off hold-off	bus-off hold-off time (units of 100ms)	Y	USINT	1	0	0	0	

Instance	Name	Help Text	Settable	Data Type	size	Min	Max	Default	Suppl. Enum
diagnostic port configuration		(Parameter Group Object Instance #4)							
22	hi-sec p/w	high level password	Y	UINT	2	0	65535	0	
23	med-sec p/w	medium level password	Y	UINT	2	0	65535	0	
24	term units	diagnostic terminal display units	Y	USINT	1	1	4	2	e_term_units
25	term type	diagnostic terminal type	Y	USINT	1	0	1	1	e_term_type
reserved		(Parameter Group Object Instance #5)							
26	(reserved)	reserved parameter		USINT	1			0	
27	(reserved)	reserved parameter		USINT	1			0	
28	(reserved)	reserved parameter		USINT	1			0	
29	(reserved)	reserved parameter		USINT	1			0	
30	(reserved)	reserved parameter		USINT	1			0	
31	(reserved)	reserved parameter		USINT	1			0	
32	(reserved)	reserved parameter		USINT	1			0	
33	(reserved)	reserved parameter		USINT	1			0	
data simulations		(Parameter Group Object Instance #6)							
34	an-in-0 sim	analog i/p chan-0 simulation flag	Y	USINT	1	0	1	0	e_sim
35	an-in-1 sim	analog i/p chan-1 simulation flag	Y	USINT	1	0	1	0	e_sim
36	dig-in sim	digital i/p simulation mask	Y	BYTE	1	0	255	0	
37	an-out sim	analog o/p simulation flag	Y	USINT	1	0	1	0	e_sim
38	dig-out sim	digital o/p simulation mask	Y	BYTE	1	0	255	0	
39	any sims	any simulations active	Y	USINT	1	0	1	0	e_sim
data configuration		(Parameter Group Object Instance #7)							
40	an-in-0 rng	analog i/p chan-0 range selection	Y	USINT	1	1	2	1	e_range
41	an-in-1 rng	analog i/p chan-1 range selection	Y	USINT	1	1	2	1	e_range
42	an-out rng	analog o/p range selection	Y	USINT	1	1	2	1	e_range
43	an-in r1 FS	analog i/p range-1 full scale		REAL	4				
44	an-in r2 FS	analog i/p range-2 full scale		REAL	4				
45	an-out r1 FS	analog o/p range-1 full scale		REAL	4				

Instance	Name	Help Text	Settable	Data Type	size	Min	Max	Default	Suppl. Enum
46	an-out r2 FS	analog o/p range-2 full scale		REAL	4				
47	an-in r1 B/U	analog i/p range-1 Bipolar/Unipolar		USINT	1	0	1	0	e_bi_uni
48	an-in r2 B/U	analog i/p range-2 Bipolar/Unipolar		USINT	1	0	1	0	e_bi_uni
49	an-out r1 B/U	analog o/p range-1 Bipolar/Unipolar		USINT	1	0	1	0	e_bi_uni
50	an-out r2 B/U	analog o/p range-2 Bipolar/Unipolar		USINT	1	0	1	0	e_bi_uni
51	an-in-0 filt	analog i/p chan-0 filter factor	Y	USINT	1	0	5	2	
52	an-in-1 filt	analog i/p chan-1 filter factor	Y	USINT	1	0	5	2	
53	an-in-0 win	analog i/p chan-0 filter window	Y	INT	2	0	32000	4096	
54	an-in-1 win	analog i/p chan-1 filter window	Y	INT	2	0	32000	4096	
55	dig-in pol	digital i/p inversion mask	Y	BYTE	1	0	255	0	
56	dig-out pol	digital o/p inversion mask	Y	BYTE	1	0	255	0	
57	io fault action	zero outputs if no comm's	Y	USINT	1	0	1	0	e_zer_op
58	no-comm's t/o	no comm's timeout (ms)	Y	UINT	2	20	65535	1000	
59	an-out Z	analog output zero-on-timeout	Y	USINT	1	0	1	0	
60	dig-out z-mask	digital output zero-on-timeout mask	Y	USINT	1	0	255	0	
calibration data			(Parameter Group Object Instance #8)						
61	an-in-0 r1 zer	analog i/p chan-0 range-1 (cal) zero offset	Y	INT	2	-3200	3200	0	
62	an-in-0 r1 cal	analog i/p chan-0 range-1 (cal) scale factor	Y	REAL	4	0.8	1.2	1	
63	an-in-0 r2 zer	analog i/p chan-0 range-2 (cal) zero offset	Y	INT	2	-3200	3200	0	
64	an-in-0 r2 cal	analog i/p chan-0 range-2 (cal) scale factor	Y	REAL	4	0.8	1.2	1	
65	an-in-1 r1 zer	analog i/p chan-1 range-1 (cal) zero offset	Y	INT	2	-3200	3200	0	
66	an-in-1 r1 cal	analog i/p chan-1 range-1 (cal) scale factor	Y	REAL	4	0.8	1.2	1	
67	an-in-1 r2 zer	analog i/p chan-1 range-2 (cal) zero offset	Y	INT	2	-3200	3200	0	
68	an-in-1 r2 cal	analog i/p chan-1 range-2 (cal) scale factor	Y	REAL	4	0.8	1.2	1	
69	an-out r1 zer	analog o/p range-1 (cal) zero offset	Y	INT	2	-768	3200	0	
70	an-out r1 hi cal	analog o/p range-1 (cal) hi scale factor	Y	REAL	4	0.9	1.1	1	
71	an-out r1 lo cal	analog o/p range-1 (cal) lo scale factor	Y	REAL	4	0.9	1.1	1	
72	an-out r2 zer	analog o/p range-2 (cal) zero offset	Y	INT	2	-3200	3200	0	
73	an-out r2 cal	analog o/p range-2 (cal) scale factor	Y	REAL	4	0.9	1.1	1	

Instance	Name	Help Text	Settable	Data Type	size	Min	Max	Default	Suppl. Enum
PID configuration		(Parameter Group Object Instance #9)							
74	PID on/off	PID: use PID	Y	USINT	1	0	1	0	e_dis_en
75	PID inverted	PID: inverted feedback	Y	USINT	1	0	1	0	e_dis_en
76	PID reset	PID: use digital i/p bit-5 to reset PID	Y	USINT	1	0	1	0	e_dis_en
77	PID hold	PID: use digital i/p bit-6 to hold PID	Y	USINT	1	0	1	0	e_dis_en
78	(reserved)	reserved parameter		USINT	1				
79	(reserved)	reserved parameter		USINT	1				
80	(reserved)	reserved parameter		USINT	1				
81	(reserved)	reserved parameter		USINT	1				
82	PID P-term	PID: proportional term	Y	REAL	4			1	
83	PID I-term	PID: integral term	Y	REAL	4			0	
84	PID D-term	PID: differential term	Y	REAL	4			0	
85	PID I lo limit	PID: integral low limit	Y	INT UINT	2 2	-32768 0	32767 65535	-32768 0	
86	PID I hi limit	PID: integral high limit	Y	INT UINT	2 2	-32768 0	32767 65535	32767 65535	
87	PD window	PD deadband window	Y	USINT	1	0	255	0	
88	(reserved)	reserved parameter		USINT	1				
89	(reserved)	reserved parameter		USINT	1				

Supplementary Enumeration Help Strings

e_category	
1	Default Ranges 1
2	Default Ranges 2
3	Custom Ranges

e_term_type	
0	Dumb
1	Small ANSI

e_bi_uni	
0	Bipolar
1	Unipolar

e_start_mode	
0	Hard start
1	Soft start

e_sw_OK	
0	Switches OK
1	Switches Bad

e_zer_op	
0	Hold Last State
1	Turn Outputs Off

e_sim	
0	Not Simulated
1	Simulated

e_term_units	
1	Percentage
2	Volts/Amps
3	Hexadecimal
4	Decimal

e_dis_en	
0	Disabled
1	Enabled

e_range	
1	Range 1
2	Range 2

Parameter Group Object Instances

Instance	Parameter Group	number in group	Parameters in group
1	Group3 specific	6	1 to 6
2	DeviceNet	7	7 to 13
3	Diagnostic	8	14 to 21
4	Diagnostic Port	4	22 to 25
5	Reserved	0	-
6	Data Simulation	6	34 to 39
7	Data Configuration	21	40 to 60
8	Calibration Data	13	61 to 73
9	PID Configuration	16	74 to 89

Notes:

Minimum and maximum values of “REAL” numbers are specified in IEEE 754 for the basic floating point format. Requesting the minimum, maximum and default values for parameter instances of this type will mostly return 00.
The parameters which use “REAL” as a data type are:

1) calibration data , where the only numbers that should be sent to the DNA are values that have previously been uploaded from that DNA. Calibration data sent to the DNA is checked for validity before storage regardless.
The “invalid attribute value” error code is returned if they fail the validity test.
2) PID P, I and D coefficients for which any value is acceptable.

3) Analog range full scale values. These are non-settable data and so don't require minimum, maximum and default values.

Parameters with enumerated help strings: the minimum and maximum values for the parameter specify the first and last help string available.

Parameter instance 16 (send a diagnostic character to the DNA) can be read but will always return 0. It is in essence a write-only parameter.

Parameter instances 14 and 34 to 39 can be written to with any data value. The effect in all cases is to clear the status. For parameter 14 this means resetting the "last reset error" to 0. In the case of parameters 34 to 38 this means disabling the specific simulation flag or mask. In the case of parameter 39 this means disabling all simulations. Data simulations should be disabled for normal operation of the DNA. If any simulations at all are active, bit 4 of the status byte in the polled IO data will be set. Also, parameter 39 will return the value 1, and one or more of the parameters 34 to 38 will be non-zero.

The parameter data values are independent of the diagnostic terminal display units (see parameter 24).

Board Serial Number

Parameter 13 (board serial number) can be written to, but the DNA will not store this value. Instead it is compared to the DNA's serial number and if it matches, "secure access" is granted for the next parameter transmitted. This is only relevant for downloading calibration data to the DNA (parameters 61 to 73). If the transmitted serial number does not match the DNA's serial number, an error status will be returned.

Analog Channel Calibration Data

Parameters 61 to 73 are concerned with the calibration data held in the DNA. This information is specific to each DNA and should not be erased or overwritten with calibration data extracted from another DNA.
It is advisable that this information be backed up onto computer so that in the unlikely event that it is corrupted by severe electrical transients, it can be restored.

The process for getting the calibration data out of the DNA is the same as for any other parameter. The process for restoring it to a DNA is not.

To help prevent the wrong calibration data being sent to a DNA, it is necessary to transmit the correct board serial number (parameter 13) to the DNA prior to each transmission of calibration data. If the correct serial number is not first sent, attempting to set an item of calibration data will result in the return of an error status.
The DNA has very sophisticated storage and recovery mechanisms for permanently stored data such as calibration information. Additionally the hardware has been designed with electrically noisy environments in mind, and prototypes have undergone stringent testing in extremely electrically hostile environments.

Despite all of this, it is still advisable to back up the calibration data onto a computer. Should the need to download calibration data ever arise, it will be evident because bit 5 of the status byte in the polled IO data will be set. Also, every menu screen of the diagnostic terminal feature will be preceded by the message "BOARD NEEDS CALIBRATING". Once this status has been set, it will remain set even after the calibration data has been downloaded, and will require explicit clearing via the diagnostic terminal: 'R' option in the '**E**'EPROM menu off the '**S**'ystem menu.

PID Integral High and Low Limits

Parameters 85 and 86 are integral limits for the analog output when the DNA is operating in PID mode. They can be used to prevent a phenomenon known as "integrator windup". Depending on the selected output range or category of DNA, the output can be either bipolar or unipolar. A bipolar output takes values in the range -32768 to 32767 (nominally -32000 to 32000) and a unipolar output takes values in the range 0 to 65535 (nominally 0 to 64000). For this reason, the parameterised data type, minimum, maximum and default value for each of these parameters depends on the bi/uni polarity of the selected output range.

Analog Filter Factors

Parameters 51 and 52 (analog filter factors): a value of 0 causes no filtering to occur. The actual filter time constants can be obtained through the diagnostic terminal.

Bad Switch Setting

If the Baud rate switch on the DNA is positioned to other than 0, 1 or 2, the module status LED will blink red/off, the diagnostic terminal (if present) will display the message "BAD SWITCH SETTING !" and parameter 15 (bad MAC-ID or baud-rate switch setting) will return the value 1. The same fault conditions will occur if the MAC-ID switches in combination are set to a value greater than 63. In both cases, the last good value selected will be retained for use by the DNA.

Security

The DNA was designed to operate in extremely electrically noisy environments. It performs a lot of dynamic data integrity checking to detect whether memory has been corrupted by electrical transients, and initiate reset and data recovery routines. The user can gain additional security by setting non-zero high and medium level passwords within the DNA. If the DNA discovers itself operating in a restricted mode (such as using simulated data) without the appropriate security level having been attained, it will correct the situation. This particular level of protection is not available if passwords have not been set.

Diagnostic Terminal Feature

The DNA provides an RS232 terminal interface for configuring and inspecting data. Two types of terminal are supported by the DNA: dumb terminal and a small ANSI terminal. The DNA is configured for small ANSI terminal by default. This is the preferred terminal type. The screen for this option is expected to have a minimum width of 32 characters and a minimum height of 16 characters and should be able to respond to the following two ANSI escape sequences:

Esc[H home, meaning go to top left of screen
Esc[2J clear screen

The DNA parameter object supports remote access to this terminal feature via parameter instances 16, 17 and 18.

Parameter 16 (send a diagnostic character to the DNA).
Parameter 17 (get the count of ready diagnostic characters at the DNA).
parameter 18 (get ready diagnostic characters from the DNA).

The remote feature is useful if the DNA is located in an inaccessible location, or for centralised configuration of several DNAs.

Because the DNA can communicate to a terminal either through its own electrical connector, or remotely via the parameter object, a switch-over mechanism is required. This is quite straight forward. From whichever terminal is required, send two characters to the DNA: either the Esc character (1B hex) or the carriage return character (0D hex) to initiate communications. Once sent, it is not necessary to resend this sequence unless changing terminal type. By default the DNA wakes up ready to talk directly to a terminal plugged into its electrical connector. If only using this type of terminal access, no start sequence is required.

Parameter 16 should be used for sending each character to the DNA as entered from the keyboard. The application program will need to handle its own local echo of the input characters. When entering multi-digit numeric values via a remote terminal, send each digit individually as entered. The DNA will also respond to the backspace character (08hex) for editing purposes.

Parameter 17 returns the size of the buffer being held for transmission. If this returns 0, there is no need to invoke parameter 18. It is not necessary to use parameter 17 before 18, but it will save time overall. The recommended strategy is to use parameter 17 to poll the DNA to see if there is a buffer to request, and to use parameter 18 only if there are characters ready.

Parameter 18 returns a string of 30 characters. The first character in the string is a length byte, which will be in the range 0 to 29. It specifies how many of the following characters in the string are relevant. As mentioned above, if the DNA is configured for small ANSI terminal, some of these characters may form an ANSI escape sequence. Such sequences will not necessarily occur within one string as requested with parameter 18. It is necessary to treat successive strings as contiguous, and construct the escape sequences from this stream.

Diagnostic terminal characters will not be lost if the DNA isn't polled rapidly. The DNA will wait for the ready characters to be requested before generating more.

For connection to the terminal connector on the DNA, Group3 can provide a hand held terminal which is powered by the DNA. Alternatively, there are a number of third party terminal emulation programs which can be run on a portable computer and will take care of all of this.

Removed Parameters

At DNA software version 4.7, parameters 26 through 33 were made reserved and parameter group object instance 5 was effectively disabled. This group of parameters previously reported live data which is also available via the polled I/O connection and the diagnostic port I/O screen.

GROUP3 TECHNOLOGY LTD

LIMITED WARRANTY

Group3 Technology Ltd. (hereinafter called the Company) warrants instruments and other products of its manufacture to be free from defects in materials and workmanship that adversely affect the product's normal functioning under normal use and service for a period of one year from the date of shipment to the purchaser.

The obligation of this warranty shall be limited to repairing or replacing, at the discretion of the Company and without charge, any equipment which the Company agrees is defective as set out above within its warranty period. The Company will reimburse lowest freight rate two-way charges on any item returned to the Company's factory or any authorised distributor or service centre, provided that prior written authorization for such return has been given by the Company.

This warranty shall not apply to any equipment which the Company determines to have become defective owing to mishandling, improper installation, alteration, negligence, inadequate maintenance, incorrect use, exposure to environmental conditions exceeding specifications, or any other circumstance not generally acceptable for equipment of a similar type.

The Company reserves the right to make changes in design without incurring any obligation to modify previously manufactured units.

No other warranties are expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Company is not liable for consequential damages.

83000001

Group3 Technology Ltd - DISTRIBUTORS & REPRESENTATIVES

European Region

United Kingdom

Pulse Power & Measurement Ltd
65 Shrivenham Hundred Business Park
Watchfield, Swindon, Wiltshire, SN6 8TY, UK
Tel: +44 (0)1793 784389 Fax: +44 (0)1793 784391
email: sales@ppm.co.uk website: www.ppmpower.co.uk

Denmark, Sweden, Norway, Finland, Iceland, Belgium, Holland, Italy, Turkey, Russia, India

Danfysik A/S
Møllehaven 31, P.O. Box 29, DK-4040 Jyllinge, Denmark.
Tel. +45 4679 0000 Fax +45 4679 0001 Contact: Erik Steinmann
email: es@danfysik.dk website: www.danfysik.com

Germany, Poland, Czech & Slovak Republics, Ukraine

Schaefer Technologie GmbH
Mörfelder Landstrasse 33, D-63225 Langen, Germany.
Tel. +49 6103 30098-0 Fax +49 6103 30098-29 Contact: Martin Schaefer
email: info@schaefer-tec.com website: www.schaefer-tec.com

Switzerland, Austria

Schaefer-Tec AG
Badimatte 21, Postfach 431, CH-3422 Kirchberg, Switzerland
Tel. +41 34 423 70 70 Fax +41 34 423 70 75 Contact: Martin Bossard
email: ch@schaefer-tec.com website: www.schaefer-tec.com

France, Spain, Portugal

Schaefer-Techniques Sarl
1, Rue du Ruisseau Blanc, F-91620 Nozay, France
Tel. +33 1 6449 6350 Fax +33 1 6901 1205 Contact: Christophe Dubegny
email: info@schaefer-tech.com website: www.schaefer-tech.com

Italy

Schaefer Italia SRL
Via Minzoni, 57, I-45100 Rovigo, Italy
Tel. +39 0425 460 218 Fax +39 0425 462 064 Contact: Paulo Bariani
email: italia@schaefer-tec.com website: www.schaefer-tec.com

China

MT Electronic Co. Ltd.
Room 503, No.24 Building Jing Tong Yuan, Sunny Uptown International Department,
Chao Yang District, Beijing, China 100024
Tel./Fax +86 10 6570 0095, mobile: +86 130 0116 1549, Contact: Liang Qing (Rosalind)
email: lqrosalind@yahoo.com.cn website: www.mt-elec.com

India

Transact India Corporation
5/1A, Grants Building, Arthur Bunder Road, Colaba, Mumbai 400 005, India
Tel. +91 22 2285 5261, or 2283 4962 extn 22, or 2202 8735 Fax +91 22 2285 2326
email: sales@transact.co.in Contact: Arish Patel arish@transact.co.in direct dial +91 22 563 64866

Israel

Scientific Products & Technology 3000 Ltd.
P.O. Box 1425, Rosh Ha'Ayin 40850, Israel
Tel. +972 3 901 4479 Fax +972 3 901 4481 Contact: Rafael Thaler
email: info_spt@netvision.net.il website: www.spt.co.il

Japan

Hakuto Company Ltd., Scientific Equipment Department,
1 - 13, Shinjuku 1-chome, Shinjuku-ku, Tokyo 160-8910, Japan
PO Box 25 Tokyo Central 100-8691
Tel. +81 3 3225 8051 Fax +81 3 3225 9011 website: www.hakuto.co.jp
Contact: Mr Tsugio Saitoh email: saito-tsugio@hakuto.co.jp
Contact: Mr Shunsuke Takahashi email: takahashi-shunsuke@hakuto.co.jp

United States & Canada

GMW Associates - magnets, magnetic instrumentation, control systems
955 Industrial Road, San Carlos, CA 94070.
P.O. Box 2578, Redwood City, CA 94064, U.S.A.
Tel. +1 650 802 8292 Fax +1 650 802 8298 Contact: Brian Richter
email: brian@gmw.com website: www.gmw.com

VI Control Systems - LabVIEW programming, control systems
2173 Deer Trail, Los Alamos, NM 87544.
Tel. (505) 662 1461 Fax (866) 422 2931 Contact: Neal Pederson
email: np@vicontrols.com website: www.vicontrols.com

Manufacturer

Group3 Technology Ltd.,
2 Charann Place, Avondale, Auckland 1026, New Zealand.
P.O. Box 71-111, Rosebank, Auckland 1348, New Zealand.
Tel. +64 9 828 3358 Fax +64 9 828 3357
email: info@group3technology.com website: www.group3technology.com