Instruction Manual

Serial Comms Interface

Description	Item Number	Description	Item Number
nXDS6i	A73501983	nXR30i 100-127/200-240 V 1ph 50/60 Hz NW25 Inlet	A90301983
nXDS10i	A73601983	nXR40i 100-127/200-240 V 1ph 50/60 Hz NW25 Inlet	A90401983
nXDS15i	A73701983	nXR60i 100-127/200-240 V 1ph 50/60 Hz NW40 Inlet	A90502983
nXDS20i	A73801983	nXR90i 100-127/200-240 V 1ph 50/60 Hz NW40 Inlet	A90602983
nXDS6iC	A73502983	nXR120i 100-127/200-240 V 1ph 50/60 Hz NW40 Inlet	A90702983
nXDS10iC	A73602983		
nXDS15iC	A73702983		
nXDS20iC	A73802983		
nXDS6iR	A73503983		
nXDS10iR	A73603983		
nXDS15iR	A73703983		
nXDS20iR	A73803983		

Original Instructions

EDWARDS

Copyright notice

Trademark credits

Edwards and the Edwards logo are trademarks of Edwards Limited, Innovation Drive, Burgess Hill, West Sussex RH15 9TW.

Disclaimer

The content of this manual may change from time to time without notice. We accept no liability for any errors that may appear in this manual nor do we make any expressed or implied warranties regarding the content. As far as practical we have ensured that the products have been designed and constructed to be safe and without risks when properly installed and used in accordance with their operating instructions.

We accept no liability for loss of profit, loss of market or any other indirect or consequential loss whatsoever.

Product warranty and limit of liability are dealt with in our standard terms and conditions of sale or negotiated contract under which this document is supplied.

You must use this product as described in this manual. Read the manual before you install, operate, or maintain the product.

Contents

Section Page

2021-AVG-GCN-0034

Contents

Illustrations

Figure — Page

Tables

Table — Page

Associated publications

Publication title	Publication number
Vacuum Pump and Vacuum System Safety	P40040100
nXDS Scroll Pump Instruction Manual	A73501880
nXRi Instruction Manual	A90301880

1 Introduction

1.1 Scope of this manual

This manual provides operational instructions for the Edwards serial communication protocol. Read this manual before you attempt to operate your product using serial communication protocol.

For safety and operating information, please refer to the relevant product Instruction manual.

1.2 Description

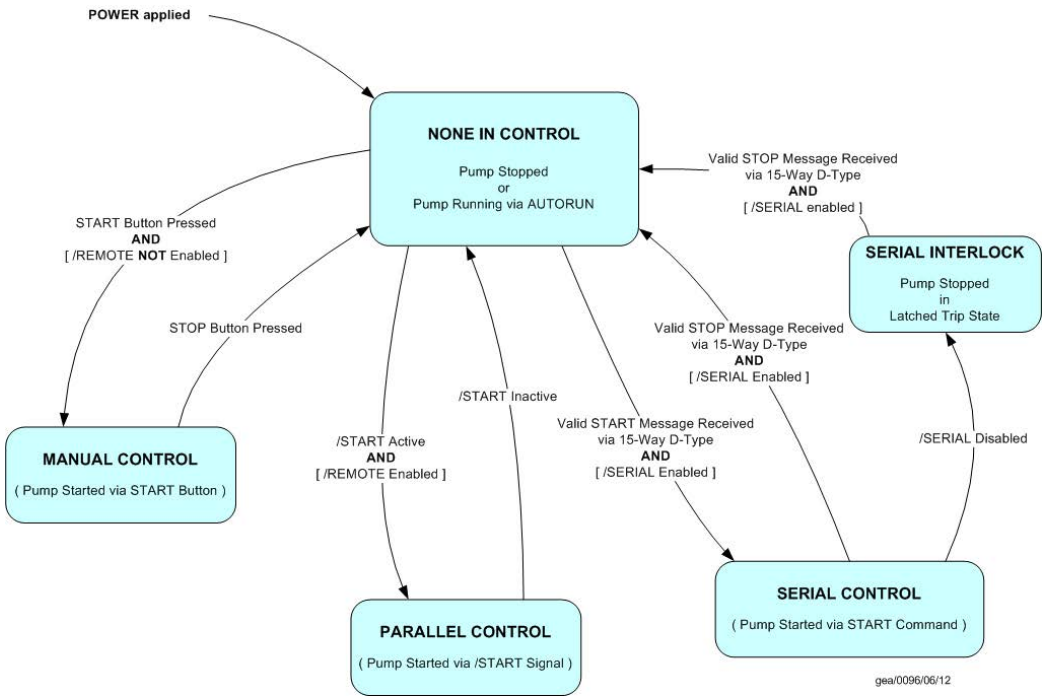
The pump can be operated in four control modes:

- None in Control Mode - Inactive control mode
- Manual Control Mode - Active control mode
- Parallel Control Mode - Active control mode
- Serial Control Mode (including Serial Interlock) - Active control mode

The control mode is determined by the way the pump is started. All the control modes, and transitions between them, are defined in Figure 1.

Note: *The pump can only be stopped by the mode in which it was started.*

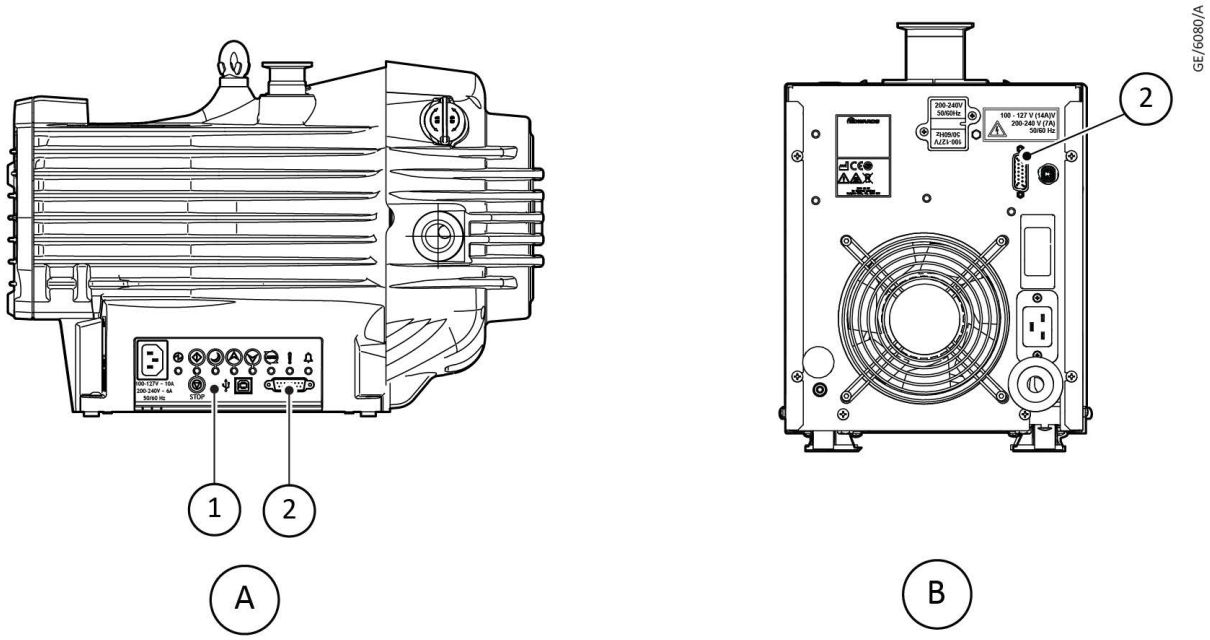
Figure 1 - Start / Stop Control Diagram



This instruction manual details the connection and operation of the pump. The parallel, manual and none in control modes are detailed within the main product instruction manual.

The pump can be connected directly to the RS485 or RS232 serial input on your control equipment or a PC, using a suitable connector mating half (not supplied). Full serial control is realised by using the following two signal lines: serial enable and RS232 / RS485 control inputs.

Figure 2 - Logic interface locations



A. nXDS scroll pump
B. nXRi roots pump

1. User interface panel
2. 15-way logic interface connection

2 Technical data

2.1 Logic interface

The connector has a male 15-way D-type logic interface connector located on the user interface panel (Figure 2, Item 2). The logic interface connector can be plugged directly into your control equipment or a PC using a suitable connector mating half (not supplied). Refer to Table 1 for the logic interface pins for the electrical connections and Table 2 for the interface technical data.

Table 1 - Logic interface connection pins

Pin Number	Signal	Use
1	Analogue Speed Enable-control Input	Connect to Pin 2 (0 V) to enable analogue speed control via Pin 9.
2	0 V Control Reference	0 V reference for ALL control and status signals listed within this table.
3	START / STOP - Control Input	Connect to Pin 2 (0 V) to START the pump system.
4	STANDBY - Control Input / RS232 Rx / RS485 A-	Connect to Pin 2 (0 V) to enable STANDBY speed when the SERIAL ENABLE control input is inactive.
5	Serial Enable - Control Input	Connect to Pin 2 (0 V) to enable serial communications.
6	RS232 / RS485 - Control Input	Default configuration is RS232 with Pin 6 unconnected. Connect to Pin 2 (0 V) to enable RS485 serial communications.
7	FAIL - Status Output / RS232 Tx / RS485 B+	Logic HIGH when a fail / fault condition exists and the SERIAL ENABLE control input is inactive.
8	0 V Control Reference	0 V reference for ALL control and status signals listed within this table.
9	Analogue Speed - Control Input	0-10 V Analogue Input: 0 V = 0% Speed; +10 V = 100% Speed
10	Chassis / Screen	Screen
11	+10 V Analogue Reference - Control Output	+10 V analogue voltage reference output: 5 mA; uni-polar output, diode protected.
12	Chassis / Screen	Screen
13	Not Connected	Unused control pin.
14	REMOTE - Control Input	Connect to Pin 2 (0 V) to enable remote operation via Parallel control mode.
15	NORMAL - Status output	Logic LOW when the pump rotational speed is at normal speed or above.

Table 2 - Logic interface technical data

Logic interface description	
Connector*	15-way D-type (male)
Start, serial enable and remote enable:	
Enable control voltage: low (closed)	0 to 0.8 V d.c. (I_{OUT} = 0.55 mA nominal)
Disable control voltage: high (open)	4 to 26.4 V d.c. (Internal pull up to 6.4 V nominal)
Standby control input:	
Enable control voltage: low (closed)	0 to 0.8 V d.c. (I_{OUT} = 0.3 mA nominal)
Disable control voltage: high (open)	4 to 26.4 V d.c. (Internal pull up to 3.2 V nominal)
Analogue and RS485 enable control inputs:	
Enable control voltage: low (closed)	0 to 0.8 V d.c. (I_{OUT} = 0.55 mA nominal)
Disable control voltage: high (open)	4 to 52.8 V d.c. (Internal pull up to 6.4 V nominal)
Analogue speed input	0 to 10 V d.c. directly proportional to the motor speed e.g. 0 V = 0 Hz, 10 V = 30 Hz
Speed set accuracy	± 5% full scale
NORMAL status output:	
Type	Open collector transistor plus pull up resistor.
< Normal speed (default 80%)	OFF (4.7 k pull up + diode to 12 V d.c.)
≥ Normal speed	ON (< 0.8 V d.c. sinking 10 mA)
Maximum current rating	10 mA
Maximum voltage rating	28.8 V d.c.
FAIL status output:	
Type	Open collector transistor plus pull up resistor.
Fail	OFF (4.7 k pull up + diode to 12 V d.c.)
OK	ON (< 0.8 V d.c. sinking 10 mA)
Maximum current rating	10 mA
Maximum voltage rating	28.8 V d.c.
Analogue 10 V reference	+ 10 V d.c. analogue voltage reference Unipolar output with diode protection
Voltage accuracy	± 2%
Output current	≤ 5 mA for specified accuracy

* *Mating half of connector not supplied*

3 Connection for serial control and monitoring

The serial interface allows you to control the pump and to interrogate its operational status using a number of serial commands. There is also a multi-drop mode that allows you to connect more than one pump to a single serial port on your control system.

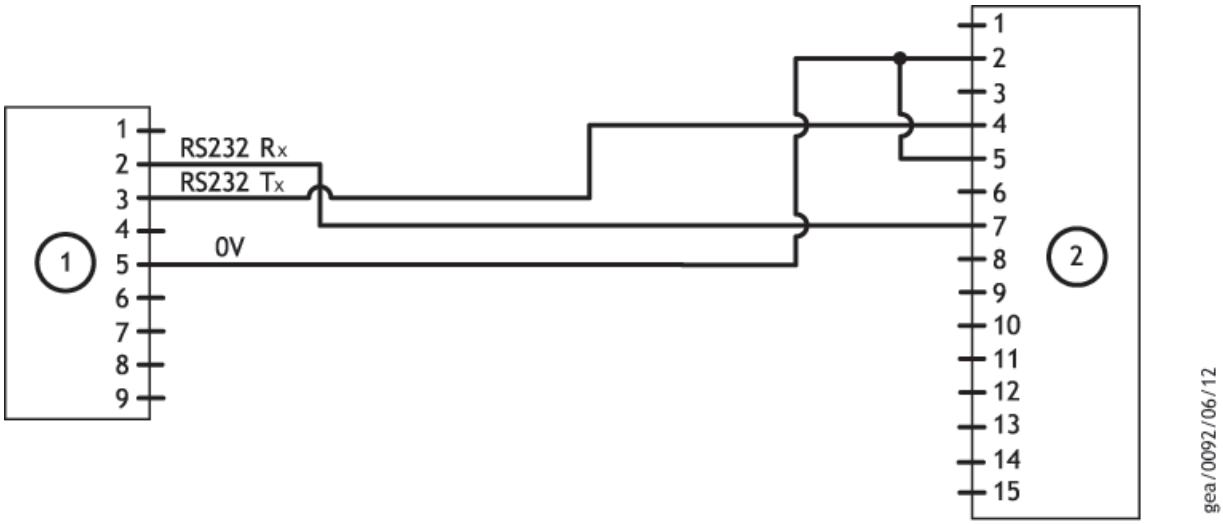
3.1 Serial connection

The pump can connect directly to the RS485 or RS232 serial input on your control equipment or a PC as shown in Figure 3 and 4. In this configuration the PC is the serial link master and the pump is the slave. The RS232 serial link is capable of operating reliably at distances up to 6m. The RS485 serial link is recommended to maintain reliable serial communications at distances greater than 6m. Alternatively an interface circuit, external to the pump, may be required to communicate using the RS232 serial link over longer distances.

The software in the pump is capable of operating with several pumps connected to a single serial link master. This is referred to as multi-drop mode (refer to Section 3.7). The RS485 option is recommended for multi-drop mode.

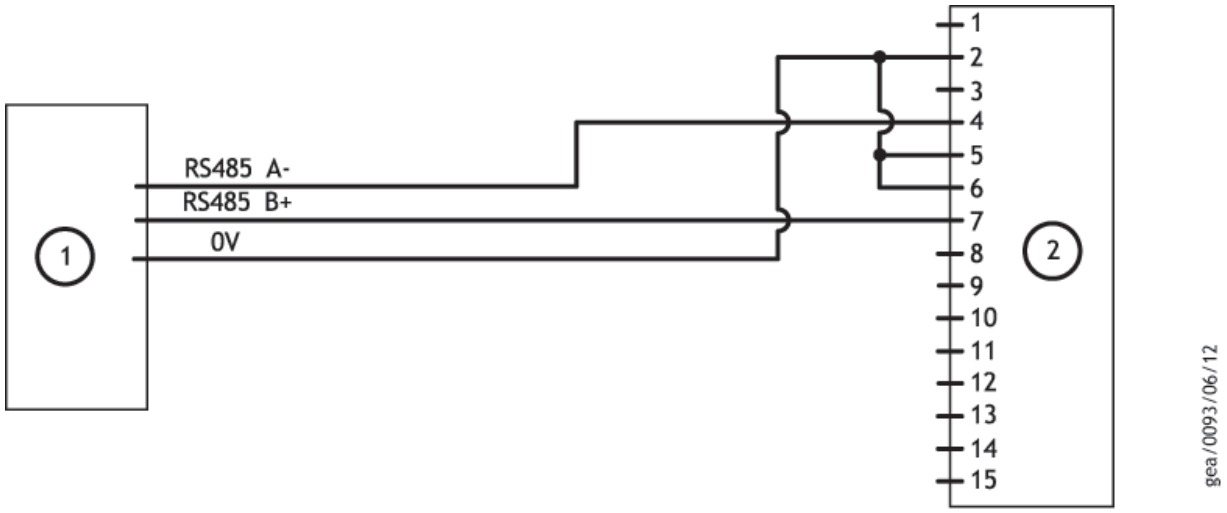
To enable the RS485 option, link the RS485 input signal to the 0 V Control Reference (pin 6 to pin 2) of your logic interface mating half.

Figure 3 - Logic interface connections - RS232 serial control



1. RS232 interface on control equipment or PC
2. Pump logic interface

Figure 4 - Logic interface connections - RS485 serial control



1. RS485 interface on control equipment or PC
2. Pump logic interface

3.2 Serial enable

To send a serial message you must first activate serial enable. This is achieved by linking the serial enable input signal (pin 5) to pin 2 of your logic interface mating half. We recommend that you incorporate this link into your serial communications cable so that the serial enable is only activated when the serial cable is connected. When you subsequently remove the cable, serial enable will become inactive.

Serial enable acts as an interlock for start commands sent over the serial interface. If the pump is running in serial control mode (having been sent a serial start command) and the serial enable subsequently becomes inactive, the pump will trigger a fail condition and will decelerate to rest. To clear this fail condition, you must re-activate the serial enable and send a serial stop command.

3.3 Serial protocol

The serial interface link is set to 9600 Baud, 8 bits, 1 stop, no parity with no handshaking. The commands are made up from printable ASCII characters. The maximum message size you can send is 80 characters, including start and end characters.

All alphabetical characters must be sent in upper case format. Response may contain lower case characters.

Every complete command message you send will receive a response - either a status code or a data return. The pump can only deal with one message at a time. It will only accept a new message once the response to the previous message has been returned.

If the pump receives characters that are not framed inside start and stop characters, it will ignore them. Messages with the stop character missing will be discarded with no response when a new start character is received. If the pump receives an unrecognisable message between the start and stop characters, it will return an appropriate error message.

3.4 Message structure

The message structure and command set are the same for RS485 and RS232 options. To communicate a message to the pump you must send the characters in a specific order. If the message does not conform to the correct structure it will be ignored and no reply will be sent.

There are two basic types of message sent to the pump:

- A command sending information to the pump; this is prefixed with a '!' character
- A query requesting information from the pump; this is prefixed with a '?' character

Data is stored and accessed via two memory types within the Pump:

- Non-volatile memory - this provides access to persistent data which is restored after power-cycling; the prefix 'S' indicates persistent data.
- Volatile memory - this provides access to non-persistent data which is NOT restored after power-cycling; the prefixes 'C' and 'V' indicate non-persistent data.

The correct structure to use is as follows:

- a valid start character, either a '!' character for a store operation or a '?' character for a query operation, followed by
- a command, which will be an upper case alphabetical character, followed by
- an object number, comprising three decimal digits, followed by
- for some commands only, a data field, comprising a sequence of characters separated from the object number by a space, followed by
- a terminating carriage return, as the stop character

An extended message protocol is used in multi-drop mode, refer to Section 3.7.

3.5 Command set

Table 3 shows a summary of the full set of commands available for controlling and monitoring the pump.

Table 4 shows the abbreviations that are used to define commands in the following sections and Table 5 shows the error codes that might be returned.

Table 3 - Summary of the commands that can be sent to the pump

Object name	Command	Parameter range	Factory setting	Data type	Units	Comments
Identification	?S0	-	-	-	-	Acts as wildcard object number to identify instrument. Reply identical to ?S801
Node	?S800 !S800	0..98	0	Decimal	Slave Address	Multi-drop address (RS485) 0 = disable multi-drop mode;
Pump type	?S801	[1..8]	-	String	ASCII Characters	Pump type (e.g. nXDS)
		[1..11]	-	String	ASCII Characters	Motor-Control Software Version number (Dxxxxxxxx Y) (where Dxxxxxxxx is the drawing number and Y is the revision)
		1..255	-	String	Hz	Design frequency (= Nominal mechanical frequency of the pump)
Pump control	!C802	0	-	Decimal	-	Stop the pump
		1	-	Decimal		Start the pump
	?V802	0..255	-	Decimal	Hz	Reported motor frequency
		64-bits encoded as four 16-bit words: <0000..FFFF>; <0000..FFFF>; <0000..FFFF>; <0000..FFFF>	-	Hexadeci-mal	-	System status: System status register 1; System status register 2; Warning register and Fault register*
Speed control	!C803	0	-	Decimal	-	Use full speed
		1	-	Decimal		Use standby speed
Normal speed threshold	?S804 !S804	50..100	80	Decimal	%	Normal speed status output trigger level: Percentage (%) of selected speed
Standby speed setting	?S805 !S805	66..100	70	Decimal	%	Standby speed of pump: Percentage (%) of full mechanical speed (see 801) !S stores standby speed to non-volatile memory (use !C805 for real-time speed control, faster execution and extended non-volatile memory life)
	!C805					!C retains value in volatile memory only
Auto-run	?S806 !S806	0 1	0	Decimal	-	Run the Pump-System from power-on: Enable = 1 Disable = 0

Table 3 - Summary of the commands that can be sent to the pump (continued)

Object name	Command	Parameter range	Factory setting	Data type	Units	Comments
Temperature readings	?V808	0..150	-	Decimal	°C	Measured pump temperature†
		0..150	-	Decimal	°C	Measured pump-controller temperature†
Link parameter readings	?V809	0..5000	-	Decimal	0.1V	Measured link voltage
		±0..300	-	Decimal	0.1A	Measured motor current
		±0..15000	-	Decimal	0.1W	Measured motor power
Run hours	?V810	0..99999 ~11 years	-	Decimal	Hours	Total run hours - time pump has run
Pump cycles	?V811	0..99999	-	Decimal	Cycles	Total number of start/stop cycles
Drive run time	?V813	0..99999 ~11 years	-	Decimal	Hours	Total run hours - time pump-controller has run
		0..99999 ~11 years	-	Decimal	Hours	Hours until recommended controller replacement
Time run since last tip seal service	?V814	0..99999 ~11 years	-	Decimal	Hours	Number of pump running hours since last tip seal service
Run time to tip seal service indicator		0..99999 ~11 years	-	Decimal	Hours	Number of pump running hours left until tip seal service due. Decreases until due at zero
Service indicator reset	!C814	1	-	Decimal	-	Command to reset service indicators, 'time since' to zero and 'time to' to service interval
Time run since last bearing service	?V815	0..99999 ~11 years	-	Decimal	Hours	Number of pump running hours since last bearing service
Run time to bearing service indicator		0..99999 ~11 years	-	Decimal	Hours	Number of pump running hours left until bearing service due. Decreases until due at zero
Bearing service indicator reset	!C815	1	-	Decimal	-	Command to reset service indicators, 'time since' to zero and 'time to' to service interval
Fault history 1	?V816	0..99999 64-bits encoded as four 16-bit words: <0000..FFFF>; <0000..FFFF>; <0000..FFFF>; <0000..FFFF>	-	Decimal Hexidecimal	Hours -	Fault history at last trip: Pump-controller powered time (hours) system status register 01; system status register 02; warning register 01 and fault register 01*

Table 3 - Summary of the commands that can be sent to the pump (continued)

Object name	Command	Parameter range	Factory setting	Data type	Units	Comments
Fault history 2	?V817	0..99999 64-bits encoded as four 16-bit words: <0000..FFFF>; <0000..FFFF>; <0000..FFFF>; <0000..FFFF>	-	Decimal Hexidecimal	Hours -	Fault history at 2nd last trip: Pump-controller powered time (hours) system status register 01; system status register 02; warning register 01 and fault register 01*
Fault history 3	?V818	0..99999 64-bits encoded as four 16-bit words: <0000..FFFF>; <0000..FFFF>; <0000..FFFF>; <0000..FFFF>	-	Decimal Hexidecimal	Hours -	Fault history at 3rd last trip: Pump-controller powered time (hours) system status register 01; system status register 02; warning register 01 and fault register 01*
Fault history 4	?V819	0..99999 64-bits encoded as four 16-bit words: <0000..FFFF>; <0000..FFFF>; <0000..FFFF>; <0000..FFFF>	-	Decimal Hexidecimal	Hours -	Fault history at 4th last trip: Pump-controller powered time (hours) system status register 01; system status register 02; warning register 01 and fault register 01*
Customer interface software version	?S820	[1..11]	-	String	ASCII Characters	Customer interface software version number (Dxxxxxxxx Y) (where Dxxxxxxxx is the drawing number and Y is the revision)
Factory settings	!C821	1	-	Decimal	-	Reset all configuration options and parameters to factory settings
Motor-control boot-loader version	?S822	[1..11]	-	String	ASCII Characters	Motor-control boot-loader version number (Dxxxxxxxx Y) (where Dxxxxxxxx is the drawing number and Y is the revision)
Customer interface boot-loader version	?S823	1..11]	-	String	ASCII Characters	Customer interface boot-loader version number (Dxxxxxxxx Y) (where Dxxxxxxxx is the drawing number and Y is the revision)

Table 3 - Summary of the commands that can be sent to the pump (continued)

Object name	Command	Parameter range	Factory setting	Data type	Units	Comments
Service setting	!S825 ?S825	0..3	0	Decimal	-	0: Service indication on Service LED 1: Service indication on Service LED and FAIL line 2: No service indication on Service LED or FAIL line 3: Service indication on FAIL line
Service status	?V826	One 16-bit word: <0000..FFFF>	-	Hexidecimal	-	Service status word - Contact Edwards for more information‡
Service	?S835	[1..30]	-	String	ASCII Characters	Serial numbers: pump; drive-module and power/control PCA (fixed at manufacture, 9 characters each)
		[1..36]	-	String	ASCII Characters	Pump type and build

* *See Section 3.9.1 for status word decoding*
† *If either returned value is '-200', then this means that this temperature is not utilised within the product*
‡ *See Section 3.9.2 for service word decoding*

Table 4 - Command abbreviations

Abbreviation	Meaning
cr	carriage return character
chars	characters
d	decimal ASCII character ***Note:*** *Fields showing multiple d characters are to indicate typical length. All data fields have a maximum of 5 decimal characters (prefixed by a minus number for negative numbers).*
h	hexadecimal ASCII character
r	Returned error code - refer to Table 5
sp	space character
string	may have several ASCII characters
X	Multi-drop decimal ASCII character ***Note:*** *Fields showing multiple X characters are to indicate maximum length and not fixed length.*

Table 5 - Error codes

Returned error code	Meaning
0	No error
1	Invalid command for object ID
2	Invalid Query/Command
3	Missing parameter
4	Parameter out of range

Table 5 - Error codes

Returned error code	Meaning
5	Invalid command in current state - e.g. serial command to start/stop when in parallel control mode

3.6 Operating the pump

3.6.1 Start the pump

To start the pump, send the following command over the serial communications link:

Command	*!*	*C*	*8*	*0*	*2*	*sp*	*1*	*cr*

The reply you receive will be in the following format:

Reply	*	*C*	*8*	*0*	*2*	*sp*	*r*	*cr*

The pump will then accelerate up to the target speed and the green run LED will flash whilst it is doing so. When the pump reaches its target speed, the green run LED will remain illuminated.

3.6.2 Standby speed

To run the pump at standby speed, send the following command over the serial communications link:

Command	*!*	*C*	*8*	*0*	*3*	*sp*	*1*	*cr*

The reply you receive will be as follows:

Reply	*	*C*	*8*	*0*	*3*	*sp*	*r*	*cr*

If the pump is currently below standby speed then it will accelerate until it reaches standby speed. If it is running faster than standby speed, it will decelerate until standby speed is reached.

To return the pump to full speed, send the following command:

Command	*!*	*C*	*8*	*0*	*3*	*sp*	*0*	*cr*

The reply you receive will be as follows:

Reply	*	*C*	*8*	*0*	*3*	*sp*	*r*	*cr*

3.6.3 Stop the pump

To stop the pump, send the following command over the serial communications link:

Command	*!*	*C*	*8*	*0*	*2*	*sp*	*0*	*cr*

The reply you receive will be in the following format:

Reply	*	*C*	*8*	*0*	*2*	*sp*	*r*	*cr*

On successful receipt of the stop command, the pump will decelerate to rest.

3.7 Multi-drop operation

Using multi-drop mode, a single computer system can communicate with more than one pump. Each pump must be assigned its own individual address, or node, before it can be fitted into a multi-drop system. The command to assign the multi-drop address is sent in standard message format and is detailed in Section 3.7.1 below.

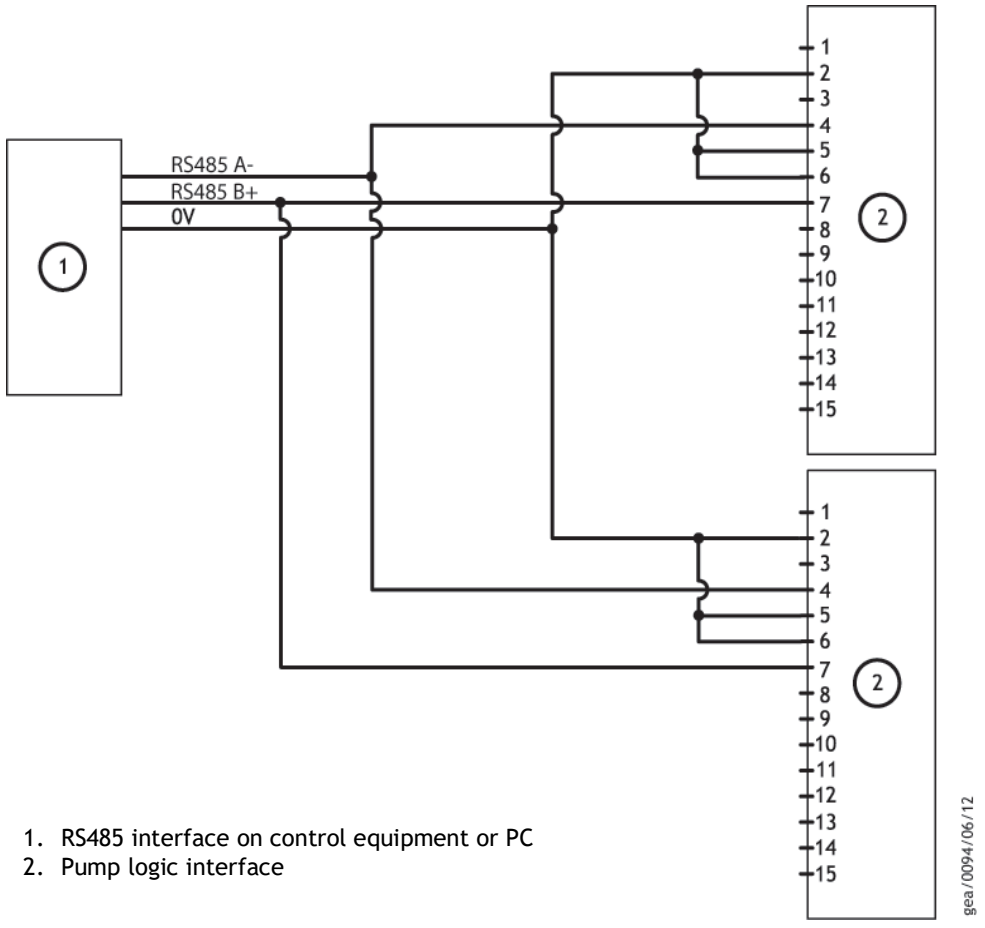
The message protocol in multi-drop mode is marginally different to that described for serial messages in single pump systems (Section 3.4). The main differences in multi-drop message protocol are detailed below:

- All multi-drop commands, queries or replies have the start character #.
- All multi-drop commands, queries and replies include a header, which contains the address of the node that the message is to, followed by the address of the node that the message is from.
- There is a delimiter character ':' (colon) which separates the two multi-drop addresses in the header.
- The remainder of the message (command, query or reply) follows the same protocol as already described for single pump systems.
- The wild card address 99 means 'any' node.

After a pump has been assigned a multi-drop address, it will ignore any messages in the format for single pumps. An individual pump will remain silent and ignore all command messages unless the multi-drop address matches its own address.

Figure 5 shows a schematic diagram of an example multi-drop connection system, which can be expanded to accommodate multiple pumps.

Figure 5 - RS485 multi-drop connections



1. RS485 interface on control equipment or PC
2. Pump logic interface

3.7.1 Assigning a multi-drop address

When you receive your pump it will have multi-drop mode disabled by default. Each individual pump must be programmed with its own multi-drop address, via a point-to-point connection, before introduction into a multi-drop network.

Send the following command to assign a multi-drop address (where the 'd' characters represent the address):

Command	!	S	8	0	0	sp	d	d	cr

Note: The address can be any decimal number from 1 to 98. The address number 0 is used to disable multi-drop mode. The address number 99 is reserved as a wild card and is used in the query set up detailed later.

The reply you receive will be as follows:

Reply	*	S	8	0	0	sp	r	cr

The multi-drop address is stored within the pump.

You can also send a query to the pump to find out whether it already has a multi-drop address. Send the following command:

Reply	?	S	8	0	0	cr

If you receive the reply shown below, your pump has multi-drop mode disabled:

Reply	=	S	8	0	0	sp	0	cr

If your pump already has a multi-drop address you will receive no reply and you must then communicate with your pump in multi-drop message protocol.

Use the following query (using wild card address 99 which means ‘any’ node) to find out the multi-drop address of the pump:

Command	#	9	9	:	9	9	?	S	8	0	0	cr

The reply you receive will be as follows, where dd denotes the multi-drop address of the pump:

Reply	#	9	9	:	9	9	=	S	8	0	0	sp	d	d	cr

You can disable multi-drop mode by assigning the pump an address 0. To do this, send the following command (where dd denotes the multi-drop address of the pump and xx denotes the address of the node that is sending the command):

Command	#	d	d	:	x	x	!	S	8	0	0	sp	0	cr

The reply you receive will be as follows:

Reply	#	x	x	:	d	d	*	S	8	0	0	sp	0	cr

Once multi-drop mode is disabled, the pump will no longer respond to multi-drop commands.

3.8 Mixed parallel and serial operation

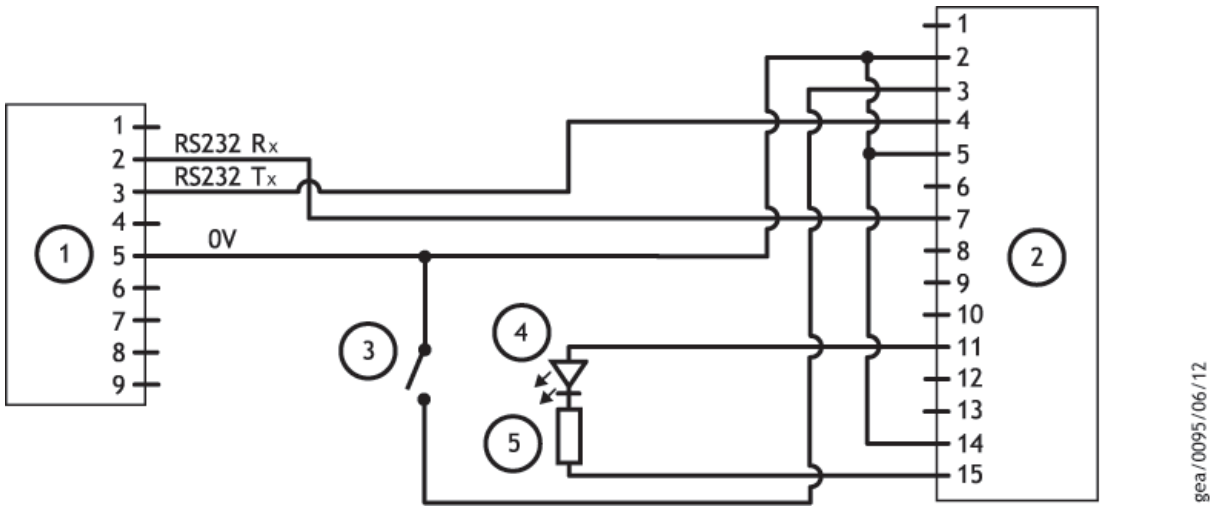
You can control the pump using the parallel interface control inputs and at the same time monitor various pump parameters using the serial interface. Alternatively you can control the pump using commands sent over the serial interface and at the same time monitor the normal signal using the parallel interface. Figure 6 shows a schematic diagram of an example mixed operation system that would allow you to do this. Many of the individual functions available in either parallel or serial operations are also available in mixed parallel and serial operation; but note that whilst serial enable is active, the parallel standby and fail signals are not available.

The following functions can also be used in conjunction with mixed parallel and serial operation:

- Multi-drop operation; described in Section 3.7.
- Analogue speed control; described in the relevant pump instruction manual.

For more information on the parallel control and monitoring, please refer to the relevant pump instruction manual.

Figure 6 - Logic interface connection - mixed parallel and serial operation



1. RS232 interface on control equipment or PC
2. Pump logic interface
3. Start / Stop switch
4. Optional LED indicator - normal speed
5. Current limit resistor for LED

You cannot control the pump using both the parallel and serial interfaces simultaneously. For example, if you start the pump by sending a start command over the serial interface, you cannot then stop the pump by using the parallel interface; you must stop the pump by sending a stop command over the serial interface. Similarly, if you start the pump by using the start / stop switch on the parallel interface, you cannot then stop the pump by using the serial interface; you must stop the pump by using the start / stop switch on the parallel interface.

3.9 Decoding status words

3.9.1 Decoding system status word

If you are using the serial communications link you will be able to access further information that may be useful for fault finding. When you send a query to monitor measured motor speed, the pump also returns a system status word.

The send command is as follows:

Command	*?*	*V*	*8*	*0*	*2*	*cr*

You will receive the following reply, where the first returned number refers to motor rotational speed in revolutions per second (Hz):

Reply	=	*V*	*8*	*0*	*2*	*sp*	*d*	*d*	*d*	;	*h*	*h*	*h*	*h*	;	*h*	*h*	*h*	*h*	;	*h*	*h*	*h*	*h*	;	*h*	*h*	*h*	*h*	*cr*

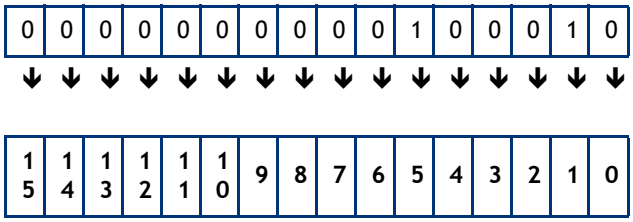
The system status word returned is made up of 4 separate status words, each made up of 4 hexadecimal digits and are separated by a semi-colon ';'. The first status word is 'System status register 1', then 'System status register 2', then 'Warning register' and the final status word is 'Fault register'. To decode each individual status word, you must convert each hexadecimal digit into a 4-digit binary number. (Table 6 is provided as an aid.) Follow the example below:

2	2	8	3
↓	↓	↓	↓
0 0 1 0	0 0 1 0	1 0 0 0	0 0 1 1

Table 6 - Hexadecimal conversion table

Hexadecimal	Binary	Decimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

Each binary digit (bit) represents a flag that is either active (state 1) or not active (state 0). To help decode each of the system status words, each bit is numbered (starting with 0 for the least significant to 15 for the most significant), as shown below:



The following 4 tables each contain a list of the 16 status flags that will be used for decoding the status or fault finding for the pump. Table 7 contains the 16 flags used to decode the 'System status register 1', Table 8 contains the 16 flags used to decode the 'System status register 2', Table 9 contains the 16 flags used to decode the 'Warning register' and Table 10 contains the 16 flags used to decode the 'Fault register'.

Table 7 - System status register 1 flags

Bit	Status Flag	Active Flat Meas
0 (lsb)	Deceleration	Stop command received and pump-controller is in the deceleration/ ramp down process
1	Acceleration/running	Accelerating or running
2	Standby speed	Standby active
3	Normal speed	Above normal speed
4	Above ramp speed	Operating above the ramp speed threshold
5	Above overload speed	Operating above the overload speed threshold
6	Control mode	Bits 6, 7 and 13 indicate which control mode the pump-controller is operating in (bit 13; bit 7; bit 6). 000=none; 001=serial; 010=parallel; 011=manual 100...111=reserved
7		
8	Reserved	-
9	Reserved	-
10	Serial enable	Serial enable active
11	Reserved	-
12	Reserved	-
13	Control mode	Used in conjunction with bits 6 and 7 above
14	Reserved	-
15 (msb)	Reserved	-

System status regis

Table 8 - System status register 2 flags

Bit	Status Flag	Active Flat Meas
0 (lsb)	Upper power regulator active	Power limit is active - i.e. pump operating on power limit
1	Lower power regulator active	Acceleration is limited to manage link voltage
2	Upper voltage regulator active	Deceleration is limited to manage link voltage
3	Reserved	-
4	Service due	Service is due - See hours counters to identify what needs replacing or use command ?V826
5	Reserved	-
6	Warning	Warning condition - See 'Warning register' for detail

Table 8 - System status register 2 flags (continued)

Bit	Status Flag	Active Flat Meas
7	Alarm	Fault condition - See 'Fault register' for detail
8	Reserved	-
9	Reserved	-
10	Reserved	-
11	Reserved	-
12	Reserved	-
13	Reserved	-
14	Reserved	-
15 (msb)	Reserved	-

Table 9 - Warning register 2 flags

Bit	Status Flag	Active Flat Meas
0 (lsb)	Reserved	-
1	Low pump-controller temperature	Pump-controller temperature is below the minimum measurable value
2	Reserved	-
3	Reserved	-
4	Reserved	-
5	Reserved	-
6	Pump-controller temperature regulator active	Output current is being restricted due to high pump-controller temperature
7	Reserved	-
8	Reserved	-
9	Reserved	-
10	High pump-controller temperature	Pump-controller temperature is above the maximum measurable value
11	Reserved	-
12	Reserved	-
13	Reserved	-
14	Reserved	-
15 (msb)	Self test warning	Non-critical problem with EEPROM or other internal function

Table 10 - Fault register flags

Bit	Status Flag	Active Flat Meas
0 (lsb)	Reserved	-
1	Over voltage trip	Fault due to excessive link voltage
2	Over current trip	Fault due to excessive motor current
3	Over temperature trip	Fault due to excessive pump-controller temperature
4	Under temperature trip	Pump-controller temperature sensor failure
5	Power stage fault	Power stage failure
6	Reserved	-
7	Reserved	-
8	H/W fault latch set	Hardware fault latch active, see bits 0-7 for detail
9	EEPROM fault	Fault due to a critical EEPROM problem (e.g. Parameter upload incomplete)
10	Reserved	-
11	No parameter set	Parameter set upload required
12	Self test fault	Self test fault (e.g. Invalid software code)
13	Serial control mode interlock	Fault because the serial enable input went inactive whilst operating with a serial start command
14	Overload time out	Fault because the output frequency fell below the threshold for more than the allowable time (with an active start command)
15 (msb)	Acceleration time out	Fault because the output frequency did not reach the threshold in the allowable time (following a start command)

3.9.2 Decoding service status word

The service status may be accessed directly via the serial link. This method of accessing service status will give the most complete picture of current and future service requirements and will allow preventative maintenance activities to be scheduled.

A summary of the current pending service status is provided in response to the service status command:

Command	*?*	*V*	*8*	*2*	*6*	*cr*

You will receive the following reply:

Reply	=	*V*	*8*	*2*	*6*	*sp*	*h*	*h*	*h*	*h*	*cr*

The service status word is made up of 4 hexadecimal digits. To decode this word, you must convert each hexadecimal digit into a 4-digit binary number as described in Section 3.9.1.

Each binary digit (bit) represents a flag that is either active (state 1) or not active (state 0). To help decode the service status word, each bit is numbered (starting with 0 for the least significant to 15 for the most significant) as shown in Section 3.9.1. The meaning of each bit in the service status word is given in Table 11.

Table 11 - Service flags

Bit number	Status flag	Active flag means
0	Tip seal service due	Set when hours until tip seal service due = 0
1	Bearing service due	Set when hours until bearing service due = 0
2	Reserved	-
3	Controller service due	Set when hours until controller service due = 0
4	Reserved	-
5	Reserved	-
6	Reserved	-
7	Service due	Service is due. Specific operation required should be determined by checking the bits above
8 - 15	Reserved	-