Title	**General Remote Control Communication Protocol** **Part I - General Commands**
Document No	SW-P-08
Issue No	32
Date	24 March 2014
Authors	**Dennis Yeomans** **Malcolm Butler** **Brian Pearce**
Approved	

Contents

Change Record

Issue	Date	Change By	Change Details
1	20/12/89		First issue for System II editors
2	22/05/90		Correct Command Bytes (Pg 7) and No. of Pages.
3	03/06/90		Major Update to incorporate commands with codes above 50.
4	12/06/90		Alteration to structure of commands 73,75,83 and associated changes concerning message lengths.
5	02/01/91		Restructuring of Section 2.2.2 and addition of Appendix 1.
6	28/08/91		Remove incorrect text from Section 3.2.3 and reformatted in Gothic type.
7	28/01/91		Amend errors in 3.2.4 and 4.2.1 in accordance with SW-DCR-016 dated 22/01/92.
8	28/08/92		Commands 45 and 46 added at Sections 4.1.7, 4.2.5 and 8.2 due to added Editor Database Version numbers to System 3 (see SW-DCR-041). Table of Contents and main document reformatted.
9	17/10/95		SW-DCR-0153. Format changed to Amipro. Edits and additions were made to the following sections: 3.1.3, 3.1.7, 3.1.8, 3.1.9, 3.1.10, 3.1.11, 3.2.8, 3.2.9, 3.2.10, 3.3.1, 3.3.3, 3.3.4, 3.3.5, 3.3.6, 3.3.7, 3.3.8.
10	17/06/97		Added additional connect on go salvo commands allowing groups of salvos to be configured.
11	12/11/97		Added command 124 and 125.
12	08/07/98		Document translated to Microsoft Word and divided into two sections: General Remote Part I (SW-P-08) and Part II (SW-P-74) and re-structured. Added new commands for Freeway (8,9,25,26,27,28). Extended protocol commands added (129,130,131,132,138,139,140,141,142,143,153,154,155, 156, 248,250,253)
13	08/09/98		Existing extended commands (listed above) modified for separate matrix and level bytes. New extended commands added with separate matrix and level bytes: 147,148,149,151,228,229,230,231,232,233,234,235,236.
14 Draft	02/10/98		Added timed stamped crosspoint, error and status logging. These commands are (47,48,49,50)
14 Draft V2.0	07/10/98		This added an additional message to 3.2.23 Logging String Messages and removed the polling enable flag bit from 3.1.24 Status Configuration and Enable Flags Request Message.
14 Draft V3.0	08/10/98		Reformatted some of the paragraphs - use soft returns instead of tabbing onto the next line. Added new message 252. Modified response 253 to use a 2 byte connect index.

Issue	Date	Change By	Change Details
15	20/11/98		First release of status error messages, Command headers changed (76-79) from draft. Added database write protect checks Command 47 & 48.Reformatted some of the paragraphs - use soft returns instead of tabbing onto the next line. Added new message 252. Modified response 253 to use a 2 byte connect index.
16	06/01/99		Modified error reporting table Ref. 3.2.23. Logging Strings Messages
17161 4 Draft V3.0	27/01/9906/ 01/9908/10/ 98		Modified error reporting table Ref. 3.2.23. Logging Strings Messages. Modified messages 1023 to 1032 - structure only. Modified messages 1033 to 1036 - Now sub module specific. Added messages 1043 to 1050 - sub module errors. Modified error reporting table Ref. 3.2.23. Logging Strings Messages Reformatted some of the paragraphs - use soft returns instead of tabbing onto the next line. Added new message 252. Modified response 253 to use a 2 byte connect index. Added messages 3272 to 3274 - sub module status. Added messages 3275 and 3276 - timecode status. Added message 4 - Panel TAKE message.
18	24/05/99		For Command Byte 78, added error messages 1051 and 1052 and status message 32777 for the internal Cues sub modules. Modified message number 0 and 1.
19	27/10/99		Re-added Logging strings message 1007, 1008, 1011 and 1012, which somehow got removed.
20	18/01/2001		Commands 29 and 30 added. Document formatting tidied up.
21	19/03/2001		Maintenance command (command 07) added, Diagnostic Data Request command (command 41) and Diagnostic Data Response command extended. 12 character names added to commands 30, 100,101,102,103,106,107,228,229,230,231,234 and 235
22	30/01/2003		New commands added: Single Source Association Name Request (command 115), All Source Association Names Request (command 114), Source Association Name Response (command 116), Names updated command (30) updated to allow for the addition of source association names. References to AXIS/SIRIUS routers added to section 9. Also Command table in section 9 updated/corrected.
23	10/12/2003		Company Logo modified, copyright and confidentiality statement modified. References to Nebula Controller added to section 9.
24	23/11/2004	AS	Cmd 41 and 43 not implemented in Aurora. Removed from command table
25	13/04/2006	DY	Group Salvo commands (120,121,122,123,124,125) added to Nebula Controller, Application examples command table updated.

Issue	Date	Change By	Change Details
26	29/03/2007	DY	New commands added to allow control of Aurora Soft Key assignments from remote ports: Command enable/disable set/read (Command 0) Soft Key Assignment Tally request (command 87) Soft Key Assignment Tally response (command 88) Soft Key Assignment Set Request (command 89) Soft Key Assignment Set Response (command 90) Commands 08/09 now implemented in Aurora, application examples command table updated. New command added (command 117) to allow source/destination names in the system database of the controller to be remotely changed.
27	12/01/2010	BP	Controller implementation table changed to add implemented commands for the Nucleus (2450) controller. Added Pre-Processing I/O commands supported by the Nucleus controller. Added commands 81 to 86
28	15/03/2011	NH	Updated Application Example tables to include commands supported by the Centra Controllers
29	16/11/2012	DY	Added implementation note to section 2.2 (Transmission protocol) regarding speed of receiver device sending ACK/NAK responses back to sender device. Updated Application Example tables and controller types.
30	08/05/2013	BP	Minor typo fixed (Section 3 description of x MOD y). Added section on using the protocol over IP. Dual controller status response message : Fixed document error which indicated a source and destination number was returned with this message. Added bit 1 to byte 1 to indicate Active / Idle status.
31	02/10/2013	BP	Dual controller status response message (09). Added response 2 to Byte 2 indicating the idle controller status is unknown.
32	24/03/2013	DY	Added 32 character name support (Source names, Source and Destination association names). Added note to indicate which controller types have support for 16/32 character names Fixed errors on extended Destination association name and extended UMD label commands where a level was included when it shouldn't be. Amended text on Source Association Name command to remove reference to level as no level in this command.

1. INTRODUCTION

This General Remote Control protocol document has been divided into two parts. The first part (this document) contains all the commands commonly used by external controllers. The second part, SW-P-74, contains commands for specific applications or, are only used within internal Pro-Bel/Snell applications. A list of all commands is in Appendix II.

This protocol has been developed to provide a common and robust method of interfacing to Pro-Bel/Snell control systems in a variety of standard and custom applications and is the standard method of interfacing to a controller from a remote device.

The philosophy of the protocol is that the Pro-Bel/Snell controller performs as a controlled device with respect to an external controlling computer. In this configuration it receives commands to perform actions and to return status. The only action initiated across the link by the controller is the spontaneous issuing of status messages as appropriate.

The protocol consists of three levels, these being the PHYSICAL layer, the DATA LINK layer and the MESSAGE or APPLICATION layer. The lower two layers provide a point-to-point communication link for reliable transfer of the command messages. It is conceivable that these layers might at a later date be enhanced or changed to provide network capability.

There is no blocking or segmentation layer and consequently the messages are structured so that they can be segmented by the application layer if necessary.

All implementation of the protocol will consist of the PHYSICAL and LINK layer components, however, not all applications will require the implementation of all command messages as the functions of many of these are application specific. Consequently, this document has been structured so that the command messages are grouped by normal application although this does not preclude alternative command sets.

On occasion it is necessary to provide more than one communication link between a Pro-Bel/Snell control system and an external computer system. This may be to provide redundancy or, more commonly to cope with large quantities of logging data that is generated by some systems. When two such lines are provided they are designated as `Control line' and `Logging line' interfaces. Both lines may be used for control purposes with responses returned on the line through which the control or data request was made. The implementation of Control and Logging lines is application specific and the use of such commands on systems which do not have this facility will result in the responses being returned on the line through which the request was made.

Pro-Bel SYSTEM2 controllers were the first controllers to utilise this protocol and were the vehicle for its initial development. The implementation provides support for the PC editors and crosspoint control.

The greater flexibility and capability of Pro-Bel/Snell SYSTEM3/AURORA controllers and editors produced a requirement for an expansion of the protocol to cover many more standard and custom situations. The command set is therefore more comprehensive than the SYSTEM2 implementation.

2. PHYSICAL AND DATA LINK LAYERS

2.1. The Physical Layer

2.1.1. Electrical

The Physical layer may use RS232 or RS422 as the electrical standard. Data rates are typically 9.6k baud on an RS232 link and 38.4k baud on the RS422 link. The selection may be application specific.

2.1.2. Data

The Data is transmitted Asynchronously on the Physical medium normally as 8 bit data bytes, no parity framed by one start bit and one stop bit to make a 10 bit data frame. Parity may be used if the Pro-Bel/Snell controller supports it.

For this application, the interface between the Physical layer and the Data-link layer may be regarded as input and output queues which buffer the incoming and outgoing data on a byte-by-byte basis.

2.2. The Data Link Layer

The Data Link Layer is a proprietary, character orientated framing protocol allowing for transparent transmission of passed messages. The packaging structure is based on the recommendations in BS4505 except that a linear checksum is employed as opposed to a cyclic redundancy check.

The interface between the Message and the Data Link Layer is on a single message in, a single message out basis, with messages queued if necessary at a higher level.

2.2.1. Message Structure

The data from the Message layer is packaged into the following structure.

SOM	DATA	BTC	CHK	EOM

Where :

SOM	=	Start of message (DLE, STX) 2 bytes
DATA	=	Message data from the application layer
BTC	=	Byte count for the data section
CHK	=	Checksum (8 bit, 2's compliment of DATA & BTC) 1byte
EOM	=	End of message (DLE, ETX) 2 bytes
DLE	=	10 HEX,
STX	=	02 HEX,
ETX	=	03 HEX

On output the data is packaged in the following order.

The checksum is calculated on the DATA and byte count (BTC), then the byte count and checksum are added to the end of the message.

The data, byte count and checksum are searched for occurrence of the DLE character. When a DLE is found it is replaced with DLE DLE to prevent the data, byte count or checksum being interpreted as a command.

The Start of Message and End of Message are added.

On input the data is unpackaged in the following order.

The Start Of Message and End Of Message are removed.

All occurrences of DLE DLE are replaced with a single DLE.

The checksum is checked.

The byte count is checked

The maximum size of the DATA field (before DLE padding) that is guaranteed to work with all systems is 128 bytes, although custom applications may use up to 255 bytes.

2.2.2. Transmission Protocol

Following transmission of a message, the remote device must wait for an ACK response, indicating message received correctly.

If a NAK response is received, it indicates an error with the message packaging (incorrect format, byte count, checksum etc.). The remote device must implement its own error handling procedure in this circumstance (retry or abort), although it is recommended that a `5 times retry' procedure is adopted which complements the procedure performed by the Pro-Bel/Snell system described below.

If there is a data reply for the remote device, it will follow an ACK response.

The remote device should receive the data reply and check the validity of the received message. If the received message packaging is correct, the remote device should return an ACK.

If the Pro-Bel/Snell controller receives a NAK, or does not receive a response within 1 second, it will re-send the data reply to the remote device. The re-send operation will be repeated a maximum of 5 times in the absence of ACK responses.

Embedded ACK/NAKs within transmitted messages are permitted in the protocol. The necessary controls are implemented to ensure no confusion with embedded DLEs.

NOTE:
Whilst a notional 1 second timeout is set for the low level ACK/NAK responses, the receiving device needs to respond with its ACK/NAK as quick as possible otherwise delay and loss of responses in the sending device may occur especially, if a large amount of status data has changed (e.g. lots of crosspoints changing or the controller is just booting up and broadcasting the initial crosspoint tally status). The receiving device ideally needs to be sending the ACK/NAK within 10ms of the sending of the end of message (ETX).

An example follows:

Remote Device		Pro-Bel Controller
Request Matrix data		Wait for a message
	Data request message	
.	----------→	.
.		.
.		Check message
Wait for ACK		packaging. Send
.		ACK/NAK.
.	ACK/NAK reply	.
.	←----------	.
.	--	.
(IF ACK)		(IF valid) Send Matrix
Wait for message data		data.
.	Matrix data	.
.	←----------	.
.	--	Wait 1 sec for a
Check message		response.
packaging.		.
Send ACK/NAK		.
.		.
.	ACK/NAK reply	.
(IF NOT valid)	----------→	(IF NAK
Wait for re-transmission		or no response)
		Re-send data
.	Re-send Matrix data	.
.	←----------	.
.		.
.		.

Where - ACK = Acknowledge message (DLE, ACK) 2 bytes.
NAK = No acknowledge message (DLE, NAK) 2 bytes
ACK = 06 HEX
NAK = 15 HEX

3. MESSAGE LAYER - GENERAL COMMANDS

The Message Layer provides the Command assembler and parser. The general format of each message is a COMMAND byte followed by the Message content.

This section describes commands that are considered to be `general' in nature and not application specific.

NOTE:

As a shorthand, the following operators have been used where:-

x DIV y= the integer (z) which is produced when `x' is divided by `y' and the digits after the decimal point are discarded.

x MOD y = the value which is left when `y' is multiplied by z [see above] and subtracted from `x'.

INT x = the integer which is produced when the digits of `x' after the decimal points are discarded.

All command and byte numbers are decimal unless otherwise specified.

3.1. General Commands Received

3.1.1. Command Enable/Disable Set/Read Message

This message is issued by a remote device to enable/disable a command from being sent on the port or to read the status of which commands are enabled/disabled. The controller will respond with a COMMAND ENABLE/DISABLE SET/READ RESPONSE (command byte 0).

Command byte: 00

Message		
Byte	**Field Format**	**Notes**
Byte 1	Command No	request a checksum
Byte 2	Status	1 = Enable 0 = Disable

Byte 2 is only sent for a 'set' command.

For a set command the 'Command 0' (zero) is not allowed.

For a read command, if the command no is 0 (zero) the status of all commands is sent in the response otherwise just the status of the specified command.

3.1.2. Crosspoint Interrogate Message

This message is a request for Tally information by matrix no., level and destination, issued by the remote device. The controller will respond to this message with a CROSSPOINT TALLY message (normal or extended) (Command Bytes 03 or 131).

Command Byte: 01

Message		
Byte	**Field Format**	**Notes**
Byte 1		Matrix/Level number
	Bits[4-7] Bits[0-3]	Matrix Number Level Number
Byte 2	Multiplier	this field allows sources and dests of up to 1023 to be used and provides source status info from a TDM or HD Digital Video router
	Bit[7] Bits[4-6] Bit[3] Bits[0-2]	0 Dest number DIV 128 TDM and Digital Video source "bad" status (0 = good source) Source number DIV 128
Byte 3	Dest number	Destination number MOD 128

Note:

When all destination bits in byte 2 are set to 1, the subsequent 7-bit byte refers to a monitoring row, hence practically, this means Dests 0 - 895 are main router dests & dests 896 to 1023 are output monitor rows (HD series routers only).

3.1.3. Crosspoint Connect Message

This message is issued by the remote device in order to set crosspoints. The controller will respond with a CROSSPOINT CONNECTED message (Command Byte 04).

Command Byte: 02

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Multiplier	Multiplier as defined in 3.1.2.
Byte 3	Dest number	Destination number MOD 128
Byte 4	Src number	Source number MOD 128

3.1.4. Maintenance Message

This message is issued by the remote device and allows various maintenance functions to be performed on the controller, i.e. hard reset, soft reset, clear protects, configure installed modules, database transfer. The number of bytes following the command byte is dependent on the operation being performed.

Command Byte: 07

Message		
Byte	**Field Format**	**Notes**
Byte 1	Maintenance Function	Sections 3.1.4.1 to 3.1.4.5 describe the functions and following message data.

3.1.4.1. Hard Reset (Byte 1 = 00)

The Hard reset command will force the controller to completely reset, as though power had just been applied. This would normally be achieved by forcing the watchdog timer to time out thus initiating a hardware reset.

There are no further message bytes.

3.1.4.2. Soft Reset (Byte 1 = 01)

The soft reset command will force the controller to do a software reset, e.g. re-initialising after a database download or a main loop restart. This is similar to the hard reset command but may not re-initialise all the hardware.

There are no further message bytes.

3.1.4.3. Clear Protects (Byte 1 = 02)

This command will clear all crosspoint protects on the controller no matter who has set the protect. This command acts as the 'MASTER' protect override.

Two additional bytes of data follow:

Message		
Byte	**Field Format**	**Notes**
Byte 2	Matrix no	Matrix No (0-19) or FFh = Clear all matrices
Byte 3	Level no	Level No (0-15) or FFh = Clear all levels

If the matrix number is set to 0FFH and the Level number is not set to 0FFH then that level on all matrices will have their protects cleared.

If the level number is set to 0FFH and the matrix number is not set to 0FFH then all levels on that matrix will have their protects cleared.

If both the matrix number and the level number are set to 0FFH then all levels on all matrices will have their protects cleared.

3.1.4.4. Configure Installed Modules (Byte 1 = 03)

This command will force the controller to read all the installed modules and re-configure itself accordingly. An auto soft reset will be performed as a result.

Two additional bytes of data follow:

Message		
Byte	**Field Format**	**Notes**
Byte 2	Matrix no	Matrix No (0-19) or FFh = configure all matrices
Byte 3	Level no	Level No (0-15) or FFh = configure all levels

If the matrix number is set to 0FFH and the Level number is not set to 0FFH then that level on all matrices will have their installed modules read and be re-configured

If the level number is set to 0FFH and the matrix number is not set to 0FFH then all levels on that matrix will have their installed modules read and be re-configured.

If both the matrix number and the level number are set to 0FFH then all levels on all matrices will have their installed modules read and be re-configured.

The current installed modules state is obtained using the Diagnostic request message (command byte 41). See SW-P-08 Part 1 Issue 12 for details.

3.1.4.5. Database Transfer (Byte 01 = 04)

This command is used on dual processor controllers and will force the database to be transferred from the ACTIVE controller to the IDLE controller.

There are no further message bytes.

3.1.5. Dual Controller Status Request Message

This message is issued by the remote device to the active controller to ascertain the status of both controllers in a dual system. The controller will respond with a DUAL CONTROLLER STATUS RESPONSE message (Command Byte 09).

Command Byte: 08

Message		
Byte	**Field Format**	**Notes**
Byte 1		No Message Bytes

3.1.6. Protect Interrogate Message

This message is issued by the remote device to get the current protect status of a destination. The controller will respond with a PROTECT TALLY message (Command Byte 11).

Command Byte: 10

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Multiplier Bit[7] Bits[4-6] Bit[3] Bits[0-2]	 0 Dest number DIV 128 0 Device number DIV 128 (0- 1023 Devices)
Byte 3	Dest number	Destination number MOD 128

3.1.7. Protect Connect Message

This is issued by the remote device to protect a destination. The controller will respond with a PROTECT CONNECTED message (Command Byte 13).

Command Byte: 12

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Multiplier	Multiplier as defined in 3.1.6
Byte 3	Dest number	Destination number MOD 128
Byte 4	Device number	Device number MOD 128

3.1.8. Protect Dis-Connect Message

This message is issued by the remote device to remove protection from a destination. The controller will issue a PROTECT DIS-CONNECTED message (Command Byte 15) in response.

Command Byte: 14

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Multiplier	Multiplier as defined in 3.1.6
Byte 3	Dest number	Destination number MOD 128
Byte 4	Device number	Device number MOD 128

3.1.9. Protect Device Name Request Message

This message is issued by the remote device or controller when the device name protecting a particular destination is required. When issued by a controller an OEM device name is being requested, when issued by the OEM remote device a Pro-Bel panel name is being requested. The remote device or controller will respond with a PROTECT DEVICE NAME RESPONSE message (Command Byte 18).

Command Byte: 17

Message		
Byte	**Field Format**	**Notes**
Byte 1	Multiplier Bit[7] Bits[3-6] Bits[0-2]	 0 Not used Device number DIV 128 (0-1023 – Total 1024)
Byte 2	Device number	Device number MOD 128

3.1.10. Protect Tally Dump Request Message

This message allows all the Protect information to be requested from the Pro-Bel Controller when a Remote device initialises. The controller will respond with a PROTECT TALLY DUMP message (Command Byte 20).

Command Byte: 19

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 3	Dest number	Destination number DIV 256
Byte 4	Dest number	Device number MOD 256

3.1.11. Crosspoint Tally Dump Request Message

This message is issued by the remote device to request a tally table dump for a given matrix/level combination. The controller will respond with CROSSPOINT TALLY DUMP (byte & word) messages (Command Bytes 22 & 23 dependent on matrix size).

Command Byte: 21

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2

3.1.12. Router I/O Parameters Interrogate Message

This message is issued by a remote device to request router source and destination parameter data from XD and Freeway routers. The controller will respond with a ROUTER I/O PARAMETERS TALLY message (Command Byte 26).

Command Byte: 25

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Type Select	Router type
	XD Routers	
	0 1 2	XD Analogue Video Source XD Digital Video Source XD AES Sources
	Freeway Routers	
	0 1 2	Audio Source Parameter Audio Destination Parameter Source Trigger Metod Parameter
Byte 3	Dest/Src Mult	Destination/Source Multiplier
Byte 4	Start number	Start Destination or Source number MOD 128
Byte 5	N	N = number of Dest or Sources (1-64) Maximum 64

3.1.13. Router I/O Parameters Connect Message

This message is issued by a remote device to set router source and destination parameter data on XD and Freeway routers. The controller will respond with a ROUTER I/O PARAMETERS CONNECTED message (Command Byte 28).

Command Byte: 27

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Type Select	Router type
Byte 3	Dst/Src Mult	Dest/Source Multiplier as defined in 3.1.12
Byte 4	Dst/Src num	Dest or Source MOD 128
Byte 5	Dst/Src Param	Destination or Source Parameters
	XD Routers : AV Reference	
	0 1	Vref 1 (PAL) Vref 2 (NTSC)
	XD Routers : DV Standard	
	0 1 2 3 4 5 6 7	360 Mbits (PAL) 270 Mbits (PAL) 177 Mbits (PAL) 143 Mbits (NTSC) 360 Mbits (NTSC) 270 Mbits (NTSC) 177 Mbits (PAL) 143 Mbits (NTSC)
	XD Routers : AES Parameter	
	Bits[3-7] Bit[2] Bit[1] Bit[0]	0 0 = forced to valid, 1 = transparent validity 0 = original CS 1 = new CS 0 = reframed 1 = synchronous
	Freeway Routers : Source Audio Parameter	
	0 1 2 3	Normal Stereo Source Left channel to both Right channel to both Left and Right channels swapped
	Freeway Routers Destination Audio Parameter	
	0 3 5	Normal Stereo Dest Left and Right channels swapped Mono or Sum left/Right channels
	Freeway Routers : Source Trigger Method	
	0 1 2	625 PAL Ref on Vref 1 525 NTSC Ref on Vref 2 TTL level pulse on Vref 1

3.1.14. Master Protect Connect Message

This message is issued by the remote device to protect a destination. The controller will claim any existing protection applied by any panel in the same way that a master panel can. The controller will respond with a PROTECT CONNECTED message (Command byte 13).

Command Byte: 29

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest mult	Destination number multiplier DIV 256
Byte 4	Dest number	Destination number MOD 256
Byte 5	Device mult	Device number multiplier DIV 256
Byte 6	Device number	Device number MOD 256

3.1.15. Diagnostic Request Message

This message is issued by the remote device to request diagnostic reports from the controller. The type of data required is defined by the message data byte. The controller will respond with a DIAGNOSTIC message (Command Byte 43).

Command Byte: 41

Message		
Byte	**Field Format**	**Notes**
Byte 1	Diagnostic 01 11 12 21 31 32 33	Type of diagnostic required Primary panel diagnostics (connectivity) Primary matrix diagnostics (connectivity) Secondary matrix status (card fault report for a given port) Installed Modules status (Modules installed in router) Software Versions Switch Settings EPROM Checksums
Byte 2	Port number	Port number for Secondary Matrix Request, otherwise ignored (set to FFh)

3.1.16. Soft Key Assignment Tally Request Message

This message is issued by a remote device to request the soft key assignments for a particular set of soft keys (suite). The controller will respond with one or more SOFT KEY ASSIGNMENT TALLY RESPONSE messages (command Byte 88).

Command byte: 87

Message		
Byte	**Field Format**	**Notes**
Byte 1	Suite No	(0 – 19)
Byte 2	Soft Key Index	Soft Key Index (0 – 23 or 255, 255 = request all indices)

3.1.17. Soft Key Assignment Set Request Message

This message is issued by a remote device to set a type/value to a matrix of a soft key index. The controller will respond with a SOFT KEY ASSIGNMENT SET RESPONSE (command byte 90) for any assignments that have changed.

Command byte: 89

Message		
Byte	**Field Format**	**Notes**
Byte 1	Suite No	(0 – 19)
Byte 2	Soft Key Index	Soft Key Index (0 – 23 or 255, 255 = request all indices)
Byte 3	Matrices 16-23	Matrices 16-23 with soft key assignments bit map where MSBit = matrix 23, LSBit = matrix 16. Bit = 1: Soft Key value assigned (type and value follows) Bit = 0: Soft Key value not assigned
Byte 4	Matrices 8-15	Matrices 8-15 with soft key assignments bit map as byte 3 except: MsBit = matrix 15, LSBit = matrix 8.
Byte 5	Matrices 0-7	Matrices 0-7 with soft key assignments bit map as byte 3 except: MsBit = matrix 7, LSBit = matrix 0.
Byte 6		1st matrix with bit set Soft Key type (0 = source association, 1 = destination association)
Byte 7		1st matrix with bit set Soft Key type value DIV 256
Byte 8		1st matrix with bit set Soft Key type value MOD 256

Then for the first matrix with a bit set in the above bit map (starting with the lowest matrix number):

Bytes 6,7 and 8 are repeated for as many matrices with bits set in the matrix bit map specified in bytes 3, 4 and 5.

A soft key can be unassigned by setting all the bits in the matrix bit map to zeros. No other bytes will follow the bit map in this case.

A value of 65535 (0xFFFF) for the soft key type value can be used to clear the soft key assignment for the selected matrix.

3.1.18. All Source Names Request Message

This message is issued by the remote device to request the names for all the sources on a given matrix and level. The controller will respond with one or more SOURCE NAME RESPONSE messages (Command Byte 106).

Command Byte: 100

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Name Length 00 01 02 03 04	Length of Names Required 4 char names 8 char names 12 char names 16 char names 32 char names

3.1.19. Single Source Name Request Message

This message is issued by the remote device to request the name for a single source. The controller will respond with a single SOURCE NAME RESPONSE message (Command Byte 106).

Command Byte: 101

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Name Length	Length of Names Required as defined in 3.1.18
Byte 3	Src multiplier	Source number DIV 256
Byte 4	Src number	Source number MOD 256

3.1.20. All Destinations Association Names Request Message

This message is issued by the remote device to request the names for all the destination associations for a given matrix. The controller will respond with one or more DESTINATION ASSOCIATION NAMES RESPONSE messages (Command Byte 107).

Command Byte: 102

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number Bits[4-7] Bits[0-3]	 Matrix number Not used
Byte 2	Length	Length of Names Required as defined in 3.1.18

3.1.21. Single Destination Association Names Request Message

This message is issued by the remote device to request the name for a single destination association for a given matrix. The controller will respond with one DESTINATION ASSOCIATION NAMES RESPONSE message (Command Byte 107).

Command Byte: 103

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	Matrix number as defined in 3.1.20
Byte 2	Name Length	Length of Names Required as defined in 3.1.18
Byte 3	Dest Ass mult	Destination Association number multiplier DIV 256
Byte 4	Dest Ass num	Destination Association number MOD 256

3.1.22. All UMD Labels Request Message

This message is issued by the remote device to request for all the alternative UMD labels on a given matrix. The controller will respond with one or more UMD LABELS RESPONSE messages (Command Byte 108).

Command Byte: 104

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	Matrix number as defined in 3.1.20
Byte 2	Type of labels 00 01 02 03	Type of Labels Required 4 char labels 8 char labels 12 char labels 16 char labels

3.1.23. Single UMD Label Request Message

This message is issued by the remote device to request a single alternative UMD labels on a given matrix. The controller will respond with a single UMD LABEL RESPONSE message (Command Byte 108).

Command Byte: 105

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	Matrix number as defined in 3.1.20
Byte 2	Label type	Type of Labels required as defined in 3.1.22
Byte 3	Src multiplier	Source number multiplier DIV 256
Byte 4	Src number	Source number MOD 256

3.1.24. All Source Association Names Request Message

This message is issued by the remote device to request the names for all the source associations for a given matrix. The controller will respond with one or more SOURCE ASSOCIATION NAMES RESPONSE messages (Command Byte 116).

Command Byte: 114

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	Matrix number as defined in 3.1.20
Byte 2	Name Length	Length of Names required as defined in 3.1.18

3.1.25. Single Source Association Name Request Message

This message is issued by the remote device to request the name for a single source association for a given matrix. The controller will respond with one SOURCE ASSOCIATION NAMES RESPONSE message (Command Byte 116).

Command Byte: 115

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	Matrix number as defined in 3.1.20
Byte 2	Name Length	Length of Names Required as defined in 3.1.18
Byte 3	Src Ass mult	Source Association number multiplier DIV 256
Byte 4	Src Ass num	Source Association number MOD 256

3.1.26. Update Name Request Message

This message is issued by a remote device to remotely update a name in the system database of the controller. There is no response to this command.

NOTE:
Using this command on an Aurora or similar controller will result in a database mismatch when the system configuration editor is next connected online. This will mean that before edits are done using the system editor the database from the controller will need to be uploaded first otherwise any remote name changes will be lost.

Command Byte: 117

Message		
Byte	**Field Format**	**Notes**
Byte 1	Name Type 00 01 02 03	 Source Name Source Association Name Destination Association Name UMD Label
Byte 2	Name Length 00 01 02 03	Length of Names Required 4 Character 8 Character 12 Character 16 Character
Byte 3	Matrix number	Matrix Number (0 – 19)
Byte 4	Level number	Level number, only applicable to Source Name type (0 -15)
Byte 5	1st Name mult	First name number multiplier DIV 256
Byte 6	1st Name num	First name number MOD 256
Byte 7	N	N = Number of names to follow ...
Bytes 8 to n	Name Chars	See notes below

In this message maximum number of names in Byte 7 is 32 for 4-char names, 16 for 8-char names, 10 for 12-char names and 8 for 16-char names

- The significance of the rest of the message is determined by Byte 2
  - If byte 2 = 0
    - Bytes 8-11 First 4 char name
    - Bytes 12-15 Second 4 char name
    - Bytes 16-19 Third 4 char name
    - Bytes 20-23 Fourth 4 char name
    - Etc.
  - If byte 2 = 1
    - Bytes 8-15 First 8 char name
    - Bytes 16-31 Second 8 char name
    - Bytes 32-39 Third 8 char name
    - Bytes 40-47 Fourth 8 char name
    - Etc.

- If byte 2 = 2
  - Bytes 8-19 First 12 char name
  - Bytes 20-31 Second 12 char name
  - Bytes 32-43 Third 12 char name
  - Bytes 44-55 Fourth 12 char name
  - Etc
- If byte 2 = 3
  - Bytes 8-31 First 16 char name
  - Bytes 32-47 Second 16 char name
  - Bytes 48-63 Third 16 char name
  - Bytes 64-79 Fourth 16 char name
  - Etc.

3.1.27. Crosspoint Tie-Line Connect Message

This message is issued by the remote device to set a multi-stage route. The remote device provides source association and destination association numbers along with a bit map of enabled levels. There is currently no response to this command.

Command Byte: 111

Message		
Byte	**Field Format**	**Notes**
Byte 1	Dest Matrix	Destination Matrix Number (0-19)
Byte 2	Dest Ass Mult	Destination Association Number DIV 256
Byte 3	Dest Ass num	Destination Association Number MOD 256
Byte 4	Dest Levels 8 - 15	Destination Levels 8-15 Bitmap where: MSBit = Level 15, LSBit = Level 8 bit = 1 Routing is enabled to this level of the dest assoc. bit = 0 Routing is disabled to this level of the dest assoc.
Byte 5	Dest Levels 0 - 7	Dest Levels 0-7 Bit Map as per byte 4 except: MSBit = Level 7, LSBit = Level 0.
Byte 6	Src Matrix	Source Matrix Number (0-19)
Byte 7	N	Source Association Number DIV 256
Byte 8	Name Chars	Source Association Number MOD 256
Byte 9	Src Levels 8 - 15	Source Levels 8-15 Bit Map where: MSBit = Level 15, LSBit = Level 8 bit = 1 Routing is enabled to this level of the source assoc. bit = 0 Routing is disabled to this level of the source assoc
Byte 10	Src Levels 0 - 7	Source Levels 0-7 Bit Map as per Byte 9 except: MSBit = Level 7, LSBit = Level 0

3.1.28. Crosspoint Tie-Line Interrogate Message

This message is issued by the remote device to request tallies for a given matrix and destination association number. The controller responds with a CROSSPOINT TIE LINE TALLY message (Command Byte 113)

Command Byte: 112

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	Destination Matrix Number (0-19)
Byte 2	Dest Ass mult	Destination Association Number multiplier DIV 256
Byte 3	Dest Ass num	Destination Association Number MOD 256

3.1.29. Crosspoint Connect On Go Group Salvo Message

This message is issued by the remote device to set up salvo switches. Routing information is held in the receiving device until activated by CROSSPOINT GO GROUP SALVO command (Command Byte 121). The controller will respond with a CROSSPOINT CONNECT ON GO GROUP SALVO ACKNOWLEDGE message (Command Byte 122) to indicate that the routing information has been stored successfully.

N.B. The group salvo commands are only implemented on the XD and ECLIPSE router ranges.

Command Byte: 120

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Multiplier	Multiplier as defined in 3.1.2
Byte 3	Dest number	Destination number MOD 128
Byte 4	Src number	Source number MOD 128
Byte 5	Salvo number Bit[7] Bits[0-6]	Holds the Salvo group number to configure 0 Salvo number 0-127 (Destination and source will always overwrite previous data).

3.1.30. Crosspoint Go Group Salvo Message

This message is issued by the remote device to triggers the receiving device to set all routes in / clear the previously received CROSSPOINT CONNECT ON GO GROUP SALVO messages. A CROSSPOINT GO DONE GROUP SALVO ACKNOWLEDGE message (Command Byte 123) will be issued to indicate that the command has been executed.

Note : No individual CONNECTED messages (Command Byte 04) are issued. It is the responsibility of the controlling / listening devices to use the CROSSPOINT CONNECT ON GO GROUP SALVO ACKNOWLEDGE (Command Byte 122) and CROSSPOINT GO DONE GROUP SALVO (Command Byte 123) to keep their tally Information up to date.

N.B. The group salvo commands are only implemented on the XD and ECLIPSE router ranges.

Command Byte: 121

Message		
Byte	**Field Format**	**Notes**
Byte 1	00 01	Set previously received messages Clear previously received messages
Byte 2	Salvo number	Salvo group number as defined in 3.1.29.

3.1.31. Crosspoint Salvo Group Interrogate Message

This message is issued by the remote device to request the status of the GROUP SALVOS. The controller will respond with a CROSSPOINT GROUP SALVO TALLY message (Command Byte 125).

N.B. The group salvo commands are only implemented on the XD and ECLIPSE router ranges.

Command Byte: 124

Message		
Byte	**Field Format**	**Notes**
Byte 1	Salvo number	Salvo group number as defined in 3.1.29
Byte 2	Connect index	this specifies the index into the SALVO GROUP specified in byte1. This command is called recursively from connect index 0, until no crosspoint data in the specified group is left.

3.1.32. Status Configuration and Enable Flags Request Message

This message is issued by the remote device to enable and set the mode of various error and status monitoring messages. On initialisation of the controller these messages on all remote ports are disabled (except with remote control ports 1 & 2 on a System 3 if enabled from the NT/OS2 editors logging).

Note time stamp logging and error status reporting is only available with the 6001/6006 control platform.

Command Byte: 76

Message		
Byte	**Field Format**	**Notes**
Byte 1	En and Mode	Enable and mode flags. Message structure. Defined in 3.2.28
	Bit[7]	Not used
	Bit[6]	Set to enable spontaneous response(except NT/OS2 Editors Logging) System will always respond to a polled request
	Bit[5]	Not used
	Bit[4]	Set to enable error logging. As defined in 3.2.28
	Bit[3]	Set to enable status logging. As defined in 3.2.28
	Bit[2]	Set to enable Date & Time Crosspoint connected messages. This will always default to a spontaneous response and overrides bit 1 as defined in 3.2.28
	Bit[1]	Set to enable logging in format for the NT/OS2 editors required. As defined in 3.2.16 log message.
	Bit[0]	Set to request current status of enable and mode flags. **NB** If this bit is set all other bits are ignored and response defined in 3.2.27 is issued.
Byte 2		Not Used

3.1.33. Error and Status Request Message

This message is issued to get current error and status. This will be a snap shot of the errors and status at the time of receipt of this message. A response will only occur if either one of the status or error enable logging bits are set.

Command Byte: 79 Request error and status data.

Message		
Byte	**Field Format**	**Notes**
Byte 1		Not used

3.1.34. Pre-Processing Config Request Message

This message triggers the receiving device to respond with Pre-Processing config response message (Command byte 82).

Command Byte: 81

Message: None

3.1.35. Pre-Processing Interrogate Message

This message requests Pre-Processing tally information. The device responds with the Pre-Processing tally message (Command byte 85).

Command Byte: 83

Message		
Byte	**Field Format**	**Notes**
Byte 1	I/P Card mult	Input card DIV 128
Byte 2	I/P Card num	Input card MOD 128.
Byte 3	PPIO number	Pre-Processing I/O No (0 to No of PPIO per card – 1)

3.1.36. Pre-Processing Connect Message

This message sets / clears Pre-Processing routing. The device will make the route and respond with a Pre-Processing connected message (Command byte 86).

Command Byte: 84

Message		
Byte	**Field Format**	**Notes**
Byte 1	I/P Card mult	Input card DIV 128
Byte 2	I/P Card num	Input card MOD 128.
Byte 3	PPIO number	Pre-Processing I/O No (0 to No of PPIO per card – 1)
Byte 4	Input to PPIO	Input to PPIO output (127 = clear, 126 = no change)
Byte 5	PPIO to Input	PPIO to Input (127 = clear, 126 = no change)

3.2. General Commands Transmitted

3.2.1. Command Enable/Disable Set/Read Response Message

This message is issued by the controller in response to a Command Enable/Disable Set/Read message (command byte 0).

Command byte: 00

Message		
Byte	**Field Format**	**Notes**
Byte 1	Cmd number	Command Number (0-255)
	If not command No 0 (zero) or a response to a set request	
Byte 2	Enab/Disable	1 = Enable 0 = Disable.
	If a response to a read request of command 0 (zero)	
Byte 2	Bitmap	Bitmap status for all commands Commands 0 - 7 (bit1 = command enabled)
Byte 3		Commands 8 – 15
Byte 4		Etc.

NOTES FOR COMMAND ENABLE/DISABLE Message:

- This is implemented in the 2637 software not in the main Aurora controller software.
- All commands except command 90 are enabled by default.

3.2.2. Crosspoint Tally Message

This message returns router tally information in response to a CROSSPOINT INTERROGATE message (Command Byte 01).

Command Byte: 03

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Multiplier	Multiplier as defined in 3.1.2
Byte 3	Dest number	Destination number MOD 128
Byte 4	Src number	Source number MOD 128

3.2.3. Crosspoint Connected Message

This message is issued spontaneously by the controller on all ports after it has confirmation that a route has been made through the router. The message is effectively broadcast.

Command Byte: 04

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Multiplier	Multiplier as defined in 3.1.2
Byte 3	Dest number	Destination number MOD 128
Byte 4	Src number	Source number MOD 128

3.2.4. Dual Controller Status Response Message

This message is issued by the controller on power-up and in response to a DUAL CONTROLLER STATUS REQUEST (Command byte 08).

Command Byte: 09

Message		
Byte	**Field Format**	**Notes**
Byte 1	Bit[0]	Active Card Status 0 = MASTER is active 1 = SLAVE is active
	Bit[1]	Active status 0 = Inactive, 1 = Active
Byte 2	Idle Card 0 1 2	Idle Card Status Idle controller is OK Idle controller is missing/faulty Idle controller unknown

3.2.5. Protect Tally Message

This message is issued by a controller in response to a PROTECT INTERROGATE message (Command Byte 10). It returns the current protect status of a destination.

Command Byte: 11

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Protect 0 1 2 3	Protect details Not Protected Pro-Bel Protected Pro-Bel override Protected *(Cannot be altered remotely)* OEM Protected
Byte 3	Multiplier	Multiplier as defined in 3.1.6
Byte 4	Dest number	Destination number MOD 128
Byte 5	Device number	Device number MOD 128

3.2.6. Protect Connected Message

This message is issued by the controller in when the protect data is altered and also if the data was unsuccessfully altered as a result of a PROTECT CONNECT message (Command Bytes 12). This message is broadcast on all ports.

Command Byte: 13

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Protect	Protect details as defined in 3.2.5
Byte 3	Multiplier	Multiplier as defined in 3.1.6
Byte 4	Dest number	Destination number MOD 128
Byte 5	Device number	Device number MOD 128

Following the issue of a protect connected message SYSTEM3 will then issue a PROTECT DEVICE NAME REQUEST message (Command Byte 17). The remote device should then respond with a PROTECT DEVICE NAME RESPONSE message (Command Byte 18).

3.2.7. Protect Dis-Connected Message

This command is issued by the controller when the protect data is altered i.e. a destination has been unprotected and also if the data was unsuccessfully altered as a result of a PROTECT DIS-CONNECT message (Command Bytes 14). This message is broadcast on all ports.

Command Byte: 15

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Protect	Protect details as defined in 3.2.5
Byte 3	Multiplier	Multiplier as defined in 3.1.6
Byte 4	Dest number	Destination number MOD 128
Byte 5	Device number	Device number MOD 128

3.2.8. Protect Device Name Response Message

This message is issued by a controller or remote device in response to a PROTECT DEVICE NAME REQUEST message (Command Byte 17), returning the device name protecting a particular destination. When received by the controller the name is assumed to be an OEM device and when received by the remote device the name is assumed to be a panel name.

Command Byte: 18

Message		
Byte	**Field Format**	**Notes**
Byte 1	Multiplier	Multiplier as defined in 3.1.6
Byte 2	Device numr	Device number MOD 128
Bytes 3 - 10	Name	Eight character ASCII device name

3.2.9. Protect Tally Dump Message

This message is issued by the Pro-Bel Controller in response to a PROTECT DUMP REQUEST (Command Byte 19). It returns all the Protect Information.

Command Byte: 20

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Num Protect	Number of Protect Tallies (Maximum 64)
Byte 3	1st Dest mult	First Destination Number DIV 256
Byte 4	1st Dest num	First Destination Number MOD 256
Byte 5 & 6	Dev and data Bit[15] Bits[12-14] 0 1 2 3 Bits[10-11] Bits[0-9]	First Device Number & Protect Data Not used Protect Data = Not Protected = Pro-Bel Protected = Pro-Bel override Protected (cannot be altered remotely) = OEM Protected Not used Device number
Byte 7 & 8		Second Device Number & Protect Data (As bytes 5 & 6 above)
Etc ...		

The message length is limited to 133 bytes, therefore a message that exceeds this will require more than one message.

3.2.10. Crosspoint Tally Dump (Byte) Message

This message is issued by a controller in response to a CROSSPOINT TALLY DUMP REQUEST (Command Byte 21). It provides tally table data for a given matrix/level combination. This message assumes a maximum destination/source number of 191.

Command Byte: 22

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Tallies	Number of tallies returned
Byte 3	1st Dest num	First destination number
Byte 4	1st Src num	First source number
Byte 5	2nd Src num	Second source number

3.2.11. Crosspoint Tally Dump (Word) Message

This message is issued by a controller in response to a CROSSPOINT TALLY DUMP REQUEST (Command Byte 21). It provides tally table data for a given matrix/level combination. This message assumes a maximum destination/source number of 65535.

Command Byte: 23

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Tallies	Number of tallies returned (Max 64)
Byte 3	1st Dest mult	First destination number multiplier DIV 256
Byte 4	1st Dest num	First destination number MOD 256
Byte 5	1st Src mult	First source number multiplier DIV 256
Byte 6	1st Src num	First source number MOD 256
Byte 7	2nd Src mult	Second source number DIV 256
Byte 8	2nd Src num	Second source number MOD 256
		Etc

The message length is limited in total to 133 bytes, thus, tables that would exceed this length would cause more than one message to be issued with the appropriate destination value in the bytes 3 and 4 of the message.

3.2.12. Router I/O Parameters Tally Message

This message is issued by a controller in response to a ROUTER I/O PARAMETER INTERROGATE message (Command Byte 25).

Command Byte: 26

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Type	Type Select as defined in 3.1.12
Byte 3	Dest/Src mult	Dest/Source Multiplier as defined in 3.1.12
Byte 4	Start Dest/Src	Start Destination or Source MOD 128
Byte 5	N	N = number of Destinations or Sources (max 64)
Byte 6	1st Dest/Src	First Dest/Source's Parameter as defined in byte 5 in 3.1.13
Byte 7	2nd Src mult	Second Dest/Source's Parameter
		Etc ...

3.2.13. Router I/O Parameter Connected Message

This message is issued spontaneously by the controller on all ports after it has confirmation that a router I/O parameter has been changed, though generally in response to a ROUTER I/O PARAMETER CONNECT message (Command Byte 27).

Command Byte: 28

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Type	Type Select as defined in 3.1.12
Byte 3	Dest/Src mult	Dest/Source Multiplier as defined in 3.1.12
Byte 4	Dest/Src num	Destination or Source MOD 128
Byte 5	Parameter	Dest/Source's Parameter as defined in byte 5 in 3.1.13

3.2.14. Names Updated Message

This is issued by the controller in order to inform remote devices that one or more names have been updated on a range of matrices. Multiple matrices may be specified in a single message by setting the appropriate bits in the matrix bitmap. The message is sent 5 seconds after the last name update was detected on any matrix. This prevents unnecessary multiple message being sent. Following a database initialisation (after a system reboot or a new database download), this message will be sent out with all configured matrices set in the matrix bitmap.

Command Byte: 30

Message		
Byte	**Field Format**	**Notes**
Bytes 1 - 4	Matrix Bitmap	32 bit matrix bitmap, high byte first (byte 4, lsb = Matrix 1)
Bytes 5 - 8	Name Ident Bits[13-31] Bit[12] Bit[11] Bit[10] Bit[9] Bit[8] Bit[7] Bit[6] Bit[5] Bit[4] Bit[3] Bit[2] Bit[1] Bit[0]	Name Identifier Flags (32 bit bitmap), high byte first. Reserved 12 character source association names 8 character source association names 4 character source association names 12 character destination names 12 character source names 16 character UMD labels 12 character UMD labels 8 character UMD labels 4 character UMD labels 8 character destination names 4 character destination names 8 character source names 4 character source names

3.2.15. Diagnostic Response Message

This message is issued by the controller in response to a DIAGNOSTIC REQUEST message (Command Byte 41).

Command Byte: 43

Message		
Byte	**Field Format**	**Notes**
Byte 1	Type of diag 01 11 12 21 31 32 33	Type of diagnostic message Primary panel diagnostics (Connectivity) Primary matrix diagnostics (Connectivity) Secondary matrix status (Card fault report for a given port) Installed Modules (Modules installed in router) Software Versions Switch Settings EPROM Checksums
Byte 2	Port number	Port number for secondary Matrix Request,otherwise ignored (set to OFFhex)
Byte 3	Onwards ...	See Sections 3.2.15.1 to 3.2.15.7

3.2.15.1.Primary Panel Diagnostic Response

Message		
Byte	**Field Format**	**Notes**
Bytes 3&4		Number of Multi-drop ports (m) OR number of point-to-point panels (n)
Byte 5	Onwards ...	Bit patterns defining connectivity of each panel. 2m bytes for multi-drop systems, n/8 bytes for point-to-point systems. Bit pattern is in Port/panel number order. The maximum number for m is 112 and for n is 512.

3.2.15.2.Primary Matrix Diagnostic Response

Message		
Byte	**Field Format**	**Notes**
Bytes 3&4		Number of Matrix ports (m)
Byte 5 to 6 + INT(m/8)	Onwards ...	Bit pattern defining connectivity of each matrix port in port number order.

3.2.15.3.Secondary Matrix Diagnostic

The secondary matrix diagnostic command will pass on to the remote device the diagnostic messages received from the router controllers as defined in the General Switcher Protocol (SW-P-02). This will enable a remote device to report the fault or process the information further if required.

Message		
Byte	**Field Format**	**Notes**
Bytes 3	router type	STATUS RESPONSE command byte (SW-P-02, Command Bytes 08, 09, 10, 16, 17).
Byte 4	Onwards ...	Message data as defined by SW-P-02 command bytes 08, 09, 10, 16 and 17..

The message lengths is limited to 128 bytes, thus, diagnostic messages that would exceed this length will be passed as more than one message by the Application layer.

3.2.15.4.Installed Modules Diagnostic

The installed module diagnostic command will pass on to the remote device the installed module messages received from the router controllers as defined in the General Switcher Protocol (SW-P-02). This will enable the remote device to ascertain which crosspoints are available. Currently only HD, XD, ECLIPSE and Freeway routers support this command.

Message		
Byte	**Field Format**	**Notes**
Bytes 3		INSTALLED MODULES RESPONSE command byte. (SW-P-02, command bytes 32, 34 and 42..
Byte 4	Onwards ...	Message data as defined by SW-P-02 command bytes 32, 34 and 42.

3.2.15.5.Software Versions

This message returns the software version strings of all the different software elements in the system. The number of software version strings will vary from controller to controller. Each version string is a null terminated free format ASCII string.

The following bytes follow byte 2.

Message		
Byte	**Field Format**	**Notes**
Bytes 3		Number of bytes in Software version strings
Byte 4		Number of Software version strings
Byte 5	Onwards ...	Software version strings, each separated by a null (00H) character

The Freeway implementation returns three software version strings where the first software version string is for the BOOT program, the second software version string is for the MAIN program and the third version string is for the OEM protocol option.

Freeway will return version strings in the following format:

‘BOOT Program Vxx.yybnn’
‘MAIN Program Vxx.yybnn’
‘OEM Protocol - none’ or ‘OEM Protocol - oem name Vxx.yybnn’

3.2.15.6.Switch Settings

This message returns the state of the various switches/jumpers in the controller that can be read and used by the software. The allocation of the switch functions and switch numbers will depend on the controller. A switch is represented by a bit in a byte with the value indicating its absolute logic state.

This is a variable length message depending on the number of switches.

The following bytes follow byte 2:

Message		
Byte	**Field Format**	**Notes**
Bytes 3	No of switches	No. of switches (If zero, no bytes follow)
Byte 4	Switch setting	Byte (No. of switches-1) DIV 8 - Switch settings

In Freeway there are 8 switch settings available as defined in original functional specification (SW-SF-180) with LSB = SW2_1 and MSB=SW2_8.

3.2.15.7.EPROM Checksums

This message returns 16 bit checksums for each EPROM or EPROM section of the controller.

This is a variable length message depending on the number of checksums returned.

The following bytes follow byte 2:

Message		
Byte	**Field Format**	**Notes**
Bytes 3	No of c/s	No. of Checksums
Byte 4-5	C/S 1	Checksum 1
Byte 6-7	C/S 2	Checksum 2
		Etc

In Freeway there are 4 checksums returned where:

Checksum 1 = Boot Program Checksum
Checksum 2 = Main Program Checksum
Checksum 3 = OEM Protocol Checksum
Checksum 4 = Fixed Database Checksum

For checksums 1, 2 and 3 the first byte if the checksum is an 8 bit sum of the bytes and the second byte of the checksum is the 2's compliment of the 8 bit sum (first byte) while for checksum 4 the value is a 16 bit sum of all the bytes in the fixed database.

3.2.16. Log Message

This message is issued spontaneously by the controller to provide the ASCII log of crosspoint transactions. The logging function is switched on/off by control of a data element with the system database (i.e. controlled through the editor system).

Command Byte: 44

Message		
Byte	**Field Format**	**Notes**
Byte 1	Onwards ...	Log String

The maximum log string may be 128 characters.

The log string is free format, thus may vary from system to system.

3.2.17. Soft Key Assignment Tally Response Message

This message is issued by a controller in response to a SOFT KEY ASSIGNMENT TALLY REQUEST message, (command byte 87). The length of the message part is limited to 147 bytes, thus soft key tables that would exceed this length would cause more than one message to be issued with the appropriate soft key index in the first soft key index field of the message.

Command byte: 88

Message		
Byte	**Field Format**	**Notes**
Byte 1	Suite number	Suite number (0-19)
Byte 2	Num Soft keys	Number of Soft Key indices to follow (1 – 24)
Byte 3	1st Soft Key	First Soft Key Index No (0 – 23)

The next part of the message represents the soft key assignments and these are variable in size depending on what has been assigned to the soft leys.

For each soft key index there is a 24 bit bitmap indicating which matrices have something assigned to the soft key. Then for each matrix that indicates something is assigned there is a 3 byte block indicating the soft key type and value for that matrix.

So for the first Soft key index of the message:

Message		
Byte	**Field Format**	**Notes**
Bytes 4	No of c/s	Matrices 16-23 with soft key assignments bit map where: MSBit = matrix 23, LSBit = matrix 16. Bit = 1: Soft Key value assigned (type and value follows) Bit = 0: Soft Key value not assigned
Byte 5	C/S 1	Matrices 8-15 with soft key assignments bit map as byte 1 except: MsBit = matrix 15, LSBit = matrix 8.
Byte 6-7	C/S 2	Matrices 0-7 with soft key assignments bit map as byte 1 except: MsBit = matrix 7, LSBit = matrix 0

Then if any of the bits in the bitmap are set then:

Message		
Byte	**Field Format**	**Notes**
Byte 7	Soft key type	1st matrix with bit set Soft Key Type (0 = source association, 1 = destination association)
Byte 8	Softkey val	1st matrix with bit set Soft key type value DIV 256 of Soft key index.
Byte 9	Softkey val	1st matrix with bit set Soft key type value MOD 256 of Soft key index

Bytes 7 – 9 are repeated for as many bits that are set in the matrix bitmap.

Bytes 4 upwards are then repeated for as many soft key indices that are in the message (byte 2).

3.2.18. Soft Key Assignment Set Response Message

This message is issued spontaneously by a controller on all remote ports after a soft key assignment has been changed. The message is effectively broadcast.

Command byte: 90

Message		
Byte	**Field Format**	**Notes**
Byte 1	Suite number	Suite number (0-19)
Byte 2	Src Chan num	Source Channel No (0 – 23)
Byte 3	Matrix num	Matrix Number (0 – 19)
Byte 4	Soft Key type	Soft Key type (0 = source association, 1 = destination association)
Byte 5	Soft Key mult	Soft Key type value multiplier DIV 256
Byte 6	Soft Key val	Soft Key type value MOD 256

A value of 65535 (0xFFFF) for a soft key value indicates nothing is assigned to the soft key for that matrix.

3.2.19. Source Names Response Message

This command is issued by the controller in response to ALL SOURCE NAMES REQUEST and SINGLE SOURCE NAME REQUEST messages (Command Bytes 100 and 101). The message length is limited to 134 bytes (except 32 character names where it's 230 bytes), thus name tables that would exceed this length would cause more than one message to be issued with the appropriate source number value in the third field of the message.

Command Byte: 106

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Names length	Length of Source Names Returned as defined in 3.1.18
Byte 3	1st Src mult	First Source number multiplier DIV 256
Byte 4	1st Src num	First Source number MOD 256
Byte 5	Num of names	Number of Source Names to follow
If Byte 2 = 00 (4-Char names)		
Bytes 6-9	1st Name	First 4-char Source name
Bytes 10-13	2nd Name	Second 4-char Source name
Bytes 14-17	3rd Name	Third 4-char Source name
Etc ...		
If Byte 2 = 01 (8-Char names)		
Bytes 6-13	1st Name	First 8-char Source name
Bytes 14-21	2nd Name	Second 8-char Source name
Bytes 22-30	3rd Name	Third 8-char Source name
Etc ...		
If Byte 2 = 02 (12-Char names)		
Bytes 6-17	1st Name	First 12-char Source name
Bytes 18-29	2nd Name	Second 12-char Source name
Etc ...		
If Byte 2 = 03 (16 char names)		
Bytes 6 – 21	1st Name	First 16-char Source name
Bytes 22-37	2nd Name	Second 16-char Source name
Etc …		
If Byte 2 = 04 (32 char names)		
Bytes 6 - 37	1st Name	First 32-char Source name
Bytes 38 - 69	2nd Name	Second 32-char Source name
Etc …		

Note: Byte 5 will always be set to 01 in response to command 101. Also in this message, a maximum of 32 4-char names, 16 8-char names, 10 12-char names, 8 16-char names or 7 32 char names.

3.2.20. Destination Association Names Response Message

This command is issued by the controller in response to an ALL DESTINATION ASSOCIATION NAMES REQUEST or SINGLE DESTINATION ASSOCIATION NAME REQUEST messages (Command Bytes 102 and 103). The message length restrictions in 3.2.19 apply.

Command Byte: 107

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2 (level always 0)
Byte 2	Names length	Length of Dest Ass Names Returned as defined in 3.1.18
Byte 3	1st Dest mult	First Dest Association number multiplier DIV 256
Byte 4	1st Dest num	First Dest Association number MOD 256
Byte 5	Num of names	Number of Dest Association Names to follow
If Byte 2 = 00 (4-Char names)		
Bytes 6-9	1st Name	First 4-char Destination Association name
Bytes 10-13	2nd Name	Second 4-char Destination Association name
Bytes 14-17	3rd Name	Third 4-char Destination Association name
Etc ...		
If Byte 2 = 01 (8-Char names)		
Bytes 6-13	1st Name	First 8-char Destination Association name
Bytes 14-21	2nd Name	Second 8-char Destination Association name
Bytes 22-30	3rd Name	Third 8-char Destination Association name
Etc ...		
If Byte 2 = 02 (12-Char names)		
Bytes 6-17	1st Name	First 12-char Destination Association name
Bytes 18-29	2nd Name	Second 12-char Destination Association name
Etc ...		
If Byte 2 = 03 (16-Char names)		
Bytes 6-21	1st Name	First 16-char Destination Association name
Bytes 22-37	2nd Name	Second 16-char Destination Association name
Etc ...		
If Byte 2 = 04 (32-Char names)		
Bytes 6-37	1st Name	First 32-char Destination Association name
Bytes 38-69	2nd Name	Second 32-char Destination Association name
Etc ...		

Note: Byte 5 will always be set to 01 in response to command 103. Also in this message has a maximum of 32 4-char names, 16 8-char names, 10 12-char names, 8 16-char names or 7 32 char names.

3.2.21. UMD Labels Response Message

This command is issued by the controller in response to an ALL UMD LABELS REQUEST or SINGLE UMD LABEL REQUEST message (Command bytes 104 and 105). The message length restrictions in 3.2.19 apply.

Command Byte: 108

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix	Matrix number as defined in 3.1.20
Byte 2	Type of label	Type of Labels Returned as defined in 3.1.22
Byte 3	1st Src mult	First Source number multiplier DIV 256
Byte 4	1st Src num	First Source number MOD 256
Byte 5	Num of labels	Number of UMD labels to follow
If Byte 2 = 00 (4-Char names)		
Bytes 6-9	1st Label	First alternative 4-char Label
Bytes 10-13	2nd Label	Second alternative 4-char Label
Etc ...		
If Byte 2 = 01 (8-Char names)		
Bytes 6-13	1st Label	First alternative 8-char Label
Bytes 14-21	2nd Label	Second alternative 8-char Label
Etc ...		
If Byte 2 = 02 (12-Char names)		
Bytes 6-17	1st Label	First alternative 12-char Label
Bytes 18-29	2nd Label	Second 12- alternative 12-char Label
Etc ...		
If Byte 2 = 03 (16-Char names)		
Bytes 6-21	1st Label	First alternative 16-char Label
Bytes 22-37	2nd Label	Second alternative 16-char Label
Etc ...		

Note: Byte 5 will always be set 01 in response to command 105. Also this message will have a maximum of 32 4-char labels, 16 8-char labels or 10 for 12-char/16-char labels.

3.2.22. Source Association Names Response Message

This command is issued by the controller in response to an ALL SOURCE ASSOCIATION NAMES REQUEST or a SINGLE SOURCE ASSOCIATION NAME REQUEST message (Command Bytes 114 and 115). The message length restrictions in 3.2.19 apply.

Command Byte: 116

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix	Matrix number.
Byte 2	Names length	Length of Src Ass Names Returned as defined in 3.1.18
Byte 3	1st Src mult	First Source Association number multiplier DIV 256
Byte 4	1st Src num	First Source Association number MOD 256
Byte 5	Num of names	Number of Source Association Names to follow
If Byte 2 = 00 (4-Char names)		
Bytes 6-9	1st Name	First 4-char Source Association name
Bytes 10-13	2nd Name	Second 4-char Source Association name
Bytes 14-17	3rd Name	Third 4-char Source Association name
Etc ...		
If Byte 2 = 01 (8-Char names)		
Bytes 6-9	1st Name	First 8-char Source Association name
Bytes 10-13	2nd Name	Second 8-char Source Association name
Bytes 14-17	3rd Name	Third 8-char Source Association name
Etc ...		
If Byte 2 = 02 (12-Char names)		
Bytes 6-17	1st Name	First 12-char Source Association name
Bytes 18-29	2nd Name	Second 12-char Source Association name
Etc ...		
If Byte 2 = 03 (16-Char names)		
Bytes 6-21	1st Name	First 16-char Source Association name
Bytes 22-37	2nd Name	Second 16-char Source Association name
Etc ...		
If Byte 2 = 04 (32-Char names)		
Bytes 6-37	1st Name	First 32-char Source Association name
Bytes 38-69	2nd Name	Second 32-char Source Association name
Etc ...		

Note: Byte 5 will always be set to 01 in response to command 115. Also in this message, max of 32 for 4 char names, 16 for 8 char names, 10 for 12 char names, 8 for 16 char names and 7 for 32 char names).

3.2.23. Crosspoint Tie Line Tally Message

This message is issued by the controller returning association based tally information in response to a CROSSPOINT TIE LINE INTERROGATE message (Command Byte 112). The command returns a tally for every level where a source is connected to the destination association.

Command Byte: 113

Message		
Byte	**Field Format**	**Notes**
Byte 1	Dest Matrix	Destination Matrix Number (0-19)
Byte 2	Dest Ass mult	Destination Association Number DIV 256
Byte 3	Dest Ass num	Destination Association Number MOD 256
Byte 4	Num Srcs	Number of Sources returned
Byte 5	Src 0 Matrix	Source 0 Matrix
Byte 6	Src 0 Lev	Source 0 Level
Byte 7	Src 0 mult	Source 0 DIV 256
Byte 8	Src 0 num	Source 0 MOD 256
Byte (n*4)+5	Src n Matrix	Source n Matrix
Byte (n*4)+6	Src n Lev	Source n Level
Byte (n*4)+7	Src n mult	Source n DIV 256
Byte (n*4)+8	Src n num	Source n MOD 256

where n = 0 to the number of destinations in the destination association.

Comments: The Tie Line Tally command contains a Source Matrix, Source Level and Source Number for every Destination in the specified Destination Association.

3.2.24. Crosspoint Connect On Go Group Salvo Acknowledge Message

This message is issued by the controller in response to a CROSSPOINT CONNECT ON GO GROUP SALVO message (Command Byte 120).

N.B. The group salvo commands are only implemented on the XD and ECLIPSE router ranges.

Command Byte: 122

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Multiplier	Multiplier as defined in 3.1.2 (Except bad source bit always 0).
Byte 3	Dest num	Destination number MOD 128
Byte 4	Srcs num	Source number MOD 128
Byte 5	Salvo num	Salvo group number as defined in 3.1.29

3.2.25. Crosspoint Go Done Group Salvo Acknowledge Message

This message is issued by the controller in response to a CROSSPOINT GO GROUP SALVO message (Command Byte 121).

N.B. The group salvo commands are only implemented on the XD and ECLIPSE router ranges.

Command Byte: 123

Message		
Byte	**Field Format**	**Notes**
Byte 1	Crosspoint stat 00 01 02	Crosspoint status = Crosspoints set. = Stored crosspoints cleared. = No crosspoints to set / clear.
Byte 2	Salvo num	Salvo group number as defined in 3.1.29

3.2.26. Crosspoint Group Salvo Tally Message

This message is issued by the controller in response to a CROSSPOINT GROUP SALVO INTERROGATE message (Command Byte 124). It returns the SALVO data for the specified SALVO GROUP. The data returned is for a particular index, the validity flag indicates whether any more data is present.

N.B. The group salvo commands are only implemented on the XD and ECLIPSE router ranges.

Command Byte: 125

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix/Level	Matrix/Level number as defined in 3.1.2
Byte 2	Multiplier	Multiplier as defined in 3.1.2 (Except bad source bit always 0).
Byte 3	Dest num	Destination number MOD 128
Byte 4	Srcs num	Source number MOD 128
Byte 5	Salvo group	Salvo group number as defined in 3.1.29
Byte 6	Connect index	Connect index as defined in 3.1.31
Byte 7	Validity 00 01 02	Validity flag Valid connect index returned, more data available Valid connect index returned, last in queue Invalid connect (no data in SALVO

3.2.27. Status Configuration and Enable flags tally Message

This message is issued by the controller in response to a STATUS CONFIGURATION AND ENABLE FLAGS REQUEST MESSAGE (Command BYTE 76) to indicate the state the mode of various error and status monitoring messages. On reset of the controller these messages on all remote ports are disabled (except with remote control ports 1 & 2 on a System 3 if enabled for NT/OS2 editors logging) and this message is broadcast.

Command Byte: 77

Message		
Byte	**Field Format**	**Notes**
Byte 1	Status Flags	Status of Enable and mode flags. Message structure Defined in 3.2.28**.**
	Bit[7]	Not used
	Bit[6]	Set if spontaneous response enabled. (except NT/OS2 Editors Logging).
	Bit[5]	Not used
	Bit[4]	Set if Error logging enabled. As defined in 3.2.28
	Bit[3]	Set if Status logging enabled. As defined in 3.2.28
	Bit[2]	Set if logging of Date & Time Crosspoint connected messages enabled. As defined in 3.2.28
	Bit[1]	Bit Set if logging in format of NT/OS2 editors is enabled. As defined in 3.2.16 log message.
	Bit[0]	Not used
Byte 2	Mode / Enable	Not used.

3.2.28. Logging Strings Messages

This message is issued by the controller according to the flags in response to a STATUS CONFIGURATION AND ENABLE FLAGS MESSAGE (Command BYTE 76) . This message is a maximum of 128 bytes, the error or status string is limited to 125 bytes.

The command field is designed to be used by an external controller as a error level trigger, the level of the trigger should be the determined by the external control system. The table below list the error messages.

When a fault occurs a status command is issued indicating a fault, this will only be re-issued if the information is polled. However when the fault is cleared a message is sent to indicated this. The maximum number of each error type stored within the controller is up to 49. If the maximum number of error occurs, once one of error is cleared it is replaced by any outstanding error. System 3 can store 49 separate errors for each logical matrix (i.e. up to 20).

Command Byte: 78

Message		
Byte	**Field Format**	**Notes**
Byte 1-2	Cmd word 0 – 1000 1001 – 32767 32788 - 65535	Command word reserved for timed stamped crosspoint logging messages. reserved for error messages. reserved for status messages.
Byte 3-n		ASCII Text message

COMMAND	**Approximate String ‘ ‘ format and explanation**
0	‘dd-mm-yyyy<TAB>HH:MM:SS:FF<TAB>Matrix name…..<TAB>LevelNAM<TAB>DestName<TAB>SrceName<CR><LF>’ This message indicates the Date and Time when a crosspoint is set (Connected Message), this message is always spontaneous if enabled. If the destination is not in any destination association, then the destination number prefixed with ‘*’ is substituted for DestName.
1	‘dd-mm-yyyy<TAB>HH:MM:SS:FF<TAB> Matrix name…..<TAB>LevelNAM<TAB>DestName<TAB>SrceName<TAB>Salvo nn<CR><LF>’ This message indicates the Date and Time when a crosspoint salvo acknowledge is received (Connected Message), this message is always spontaneous if enabled. If the destination is not in any destination association, then the destination number prefixed with ‘*’ is substituted for DestName.
2	‘dd-mm-yyyy<TAB>HH:MM:SS:FF<TAB>Salvo NN go received<CR><LF>’ This message is sent when a connect on go acknowledge has been received and indicates which salvo group.
3	‘dd-mm-yyyy<TAB>HH:MM:SS:FF<TAB>Salvo NN cleared<CR><LF>’ This command is issued from the controller if a salvo has been cleared.
4	‘dd-mm-yyyy<TAB>HH:MM:SS:FF<TAB>PanelNam<TAB>Matrix name.....<TAB>DestIndx<CR><LF> This message is sent when a panel performs a TAKE function. NOTE the panel must have “Log Takes To The PC” turned on in the database for that panel.
	Note – Commands in bold indicate a fault, followed by the next command indicating a fault removed.

COMMAND	Approximate String ‘ ‘ format and explanation
1005	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…..<TAB>LevelNAM <TAB>Port nn<TAB>Fault <CR><LF> This message is sent when communication is lost to a physical router.
1006	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…..<TAB>LevelNAM <TAB>Port nn<TAB>Fixed <CR><LF> This message is sent when communication is Restored to a physical router.
1007	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fault<CR><LF>’ This is sent when a general card fault is detected within a router connected to physical port nn, where id shows the fault type (different routers display different fault types – please see the technical manuals for each router type). Note fault types ‘P’, ’F’, ‘I’, ‘O’, ‘C’ have dedicated messages listed below.
1008	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fixed<CR><LF>’ This is sent when a general card fault previously detected is fixed within a router connected to physical port nn, where id shows the fault type (different routers display different fault types – please see the technical manuals for each router type). Note fault types ‘P’, ’F’, ‘I’, ‘O’, ‘C’ have dedicated messages listed below.
1009	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…..<TAB>LevelNAM <TAB>Port nn<TAB> Frame Port nn<TAB>Fault <CR><LF>’ This message is sent when communication has been lost between an expansion interface and a section of its physical router. i.e. Interface has two 256*128routers to make a physical 256*256 and communication is lost to a 256*128 section.
1010	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…..<TAB>LevelNAM <TAB>Port nn<TAB> Frame Port nn<TAB>Fixed <CR><LF>’ This message is sent when communication is restored between an expansion interface and a section of its physical router. i.e. Interface has two 256*128routers to make a physical 256*256 and communication is restored to a 256*128 section
1011	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fault<CR><LF>’ This message is sent when a general card fault is detected within a physical router connected via an expansion. i.e. The interface has two 256*128 routers to make a physical 256*256, where id shows the fault type (different routers display different fault types – please see the technical manuals for each router type). Note fault types ‘P’, ’F’, ‘I’, ‘O’, ‘C’ have dedicated messages listed below.
1012	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fixed<CR><LF>’ This message is sent when a general card fault previously detected is fixed within in a physical router connected via an expansion. i.e. The interface has two 256*128 routers to make a physical 256*256, where id shows the fault type (different routers display different fault types – please see the technical manuals for each router type). Note fault types ‘P’, ’F’, ‘I’, ‘O’, ‘C’ have dedicated messages listed below.
1013	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fault<CR><LF>’ This is sent when a PSU fault is detected within a router connected to physical port nn, where id = ‘P’, P=PSU failure.
1014	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fixed<CR><LF>’ This is sent when a PSU fault previously detected is fixed within a router connected to physical port nn, where id = ‘P’, P=PSU failure

COMMAND	Approximate String ‘ ‘ format and explanation
1015	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fault<CR><LF>’ This is sent when a Fan fault is detected within a router connected to physical port nn, where id = ‘F’, F= fan failure.
1016	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fixed<CR><LF>’ This is sent when a Fan fault previously detected is fixed within a router connected to physical port nn, where id = ‘F’, F= fan failure.
1017	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fault<CR><LF>’ This is sent when a input card fault is detected within a router connected to physical port nn, where id = ‘I’, I= input card.
1018	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fixed<CR><LF>’ This is sent when a input card fault previously detected is fixed within a router connected to physical port nn, where id = ‘I’, I= input card.
1019	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fault<CR><LF>’ This is sent when a output card fault is detected within a router connected to physical port nn, where id = ‘O’, O= output card.
1020	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fixed<CR><LF>’ This is sent when a output card fault previously detected is fixed within a router connected to physical port nn, where id = ‘O’, O= output card.
1021	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fault<CR><LF>’ This is sent when a card fault is detected within a router connected to physical port nn, where id = ‘C’, C= crosspoint card
1022	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB>Card nn<TAB>id<TAB>Fixed<CR><LF>’ This is sent when a card fault previously detected has been fixed connected to physical port nn, where id = ‘C’, C= crosspoint card.
1023	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fault<CR><LF>’ This message is sent when a PSU fault is detected in a physical router connected via an expansion. i.e. The interface has two 256*128 routers to make a physical 256*256. Where id = ‘P’, P=PSU failure.
1024	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fixed<CR><LF>’ This message is sent when a PSU fault previously detected has been fixed in a physical router connected via an expansion. i.e. The interface has two 256*128 routers to make a physical 256*256. Where id = ‘P’, P=PSU failure.
1025	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fault<CR><LF>’ This message is sent when a Fan fault is detected in a physical router connected via an expansion. i.e.

COMMAND	**Approximate String ' ' format and explanation**
	The interface has two 256*128 routers to make a physical 256*256. Where id = 'F', F=fan failure.
1026	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fixed<CR><LF>' This message is sent when a Fan fault previously detected has been fixed in a physical router connected via an expansion. i.e. The interface has two 256*128 routers to make a physical 256*256. Where id = 'F', F=fan failure..
1027	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fault<CR><LF>' This message is sent when a input card fault is detected in a physical router connected via an expansion. i.e. The interface has two 256*128 routers to make a physical 256*256. Where id = 'I', I= input card.
1028	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fixed<CR><LF>' This message is sent when a input card fault previously detected has been fixed in a physical router connected via an expansion. i.e. The interface has two 256*128 routers to make a physical 256*256. Where id = 'I', I= input card.
1029	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fault<CR><LF>' This message is sent when a output card fault is detected in a physical router connected via an expansion. i.e. The interface has two 256*128 routers to make a physical 256*256. Where id = 'O', O= input card.
1030	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fixed<CR><LF>' This message is sent when a output card fault previously detected has been fixed in a physical router connected via an expansion. i.e. The interface has two 256*128 routers to make a physical 256*256. Where id = 'O', O= input card.
1031	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fault<CR><LF>' This message is sent when a card fault is detected in a physical router connected via an expansion. i.e. The interface has two 256*128 routers to make a physical 256*256. Where id = 'C', C= crosspoint card.
1032	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Matrix name…<TAB>LevelNAM<TAB>Port nn<TAB> Frame Port nn<TAB>Card nn<TAB> id<TAB>Fixed<CR><LF>' This message is sent when a card fault has been fixed in a physical router connected via an expansion. i.e. The interface has two 256*128 routers to make a physical 256*256. Where id = 'C', C= crosspoint card.
1033	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Display Sub System Fault<CR><LF>' This message is sent when the 6001 controller detects that the internal Display Sub System has failed to initialise or has a fault.
1034	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB> Display Sub System Fixed<CR><LF>' This message is sent when the 6001 controller fixes a previously detected fault on the internal Display Sub System.
1035	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Internal Cue<TAB>Sub Module No.<TAB>XY<TAB>Fault<TAB>2638 switch:<TAB>BBBBBBBB<CR><LF>' This message is sent when the 6001 controller detects a fault on one of its internal cue boards, where XY is the two digit sub module number and BBBBBBBB is either 'LOCAL ' if the sub module is on the local 2634 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards).

COMMAND	Approximate String ‘ ‘ format and explanation
1036	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Internal Cue<TAB>Sub Module No.<TAB>XY<TAB>Fixed<TAB>2638 switch:<TAB>BBBBBBBB<CR><LF>’ This message is sent when the 6001 controller fixes a previously detected fault on one of its internal cue boards, where XY is the two digit sub module number and BBBBBBBB is either ‘LOCAL ’ if the sub module is on the local 2634 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards).
1037	‘dd-mm-yyyy<TAB>HH:MM:SS:FF<TAB>Panel<TAB>Panel Name<TAB>Port nn<TAB>Position nn<TAB>Missing<CR><LF>’ This message indicates that a panel has been defined in the database and placed on a multi-drop chain, but the System cannot detect it.
1038	‘dd-mm-yyyy<TAB>HH:MM:SS:FF<TAB>Panel<TAB>Panel Name<TAB>Port nn<TAB>Position nn<TAB>Found<CR><LF>’ This message indicates that a panel that was missing has now been found on a multi-drop chain.
1039	‘dd-mm-yyyy<TAB>HH:MM:SS:FF<TAB>Device <TAB>Unknown <TAB>Port nn<TAB>Position nn<TAB>Found <CR><LF>’ This message is sent when an unknown device is detected on a panel chain, i.e. not entered within the database.
1040	‘dd-mm-yyyy<TAB>HH:MM:SS:FF<TAB>Device<TAB>Unknown <TAB>Port nn<TAB>Position nn<TAB>Gone <CR><LF>’ This message is sent when the unknown device is no longer detected i.e. entered within the database.
1041	‘dd-mm-yyyy<TAB>HH:MM:SS:FF<TAB>UMD<TAB>UMD Name<TAB>Port nn<TAB>Position nn<TAB>Missing<CR><LF>’ This message indicates that a UMD has been defined in the database and placed on a multi-drop chain, but the System cannot detect it.
1042	‘dd-mm-yyyy<TAB>HH:MM:SS:FF<TAB>UMD<TAB>UMD Name<TAB>Port nn<TAB>Position nn<TAB>Found<CR><LF>’ This message indicates that a UMD that was missing has now been found on a multi-drop chain.
1043	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Device <TAB>Sub Module No.<TAB>XY<TAB>Fault<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>’ This message is sent when the 6001 controller detects a fault on one of its internal Device boards, where XY is the two digit sub module number, BBBBBBBB is either ‘LOCAL ’ if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards), and Z is a 3 or 4 indicating that the sub module sits on a 2633 or 2634 card.
1044	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Device <TAB>Sub Module No.<TAB>XY<TAB>Fixed<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>’ This message is sent when the 6001 controller fixes a previously detected fault on one of its internal Device boards, where XY is the two digit sub module number, BBBBBBBB is either ‘LOCAL ’ if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards), and Z is a 3 or 4 indicating that the sub module sits on a 2633 or 2634 card.
1045	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Remote Control<TAB>Sub Module No.<TAB>XY<TAB>Fault<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>’ This message is sent when the 6001 controller detects a fault on one of its internal Remote Control boards, where XY is the two digit sub module number, BBBBBBBB is either ‘LOCAL ’ if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards), and Z is a 3 or 4

COMMAND	**Approximate String ' ' format and explanation**
	indicating that the sub module sits on a 2633 or 2634 card.
1046	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Remote Control<TAB>Sub Module No.<TAB>XY<TAB>Fixed<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>' This message is sent when the 6001 controller fixes a previously detected fault on one of its internal Remote Control boards, where XY is the two digit sub module number, BBBBBBBB is either 'LOCAL ' if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards), and Z is a 3 or 4 indicating that the sub module sits on a 2633 or 2634 card.
1047	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Matrix <TAB>Sub Module No.<TAB>XY<TAB>Fault<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>' This message is sent when the 6001 controller detects a fault on one of its internal Matrix boards, where XY is the two digit sub module number, BBBBBBBB is either 'LOCAL ' if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards), and Z is a 3 or 4 indicating that the sub module sits on a 2633 or 2634 card.
1048	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Matrix <TAB>Sub Module No.<TAB>XY<TAB>Fixed<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>' This message is sent when the 6001 controller fixes a previously detected fault on one of its internal Matrix boards, where XY is the two digit sub module number, BBBBBBBB is either 'LOCAL ' if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards), and Z is a 3 or 4 indicating that the sub module sits on a 2633 or 2634 card.
1049	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB>External Timecode<TAB>Fault <CR><LF>' This message is sent when the 6001 controller detects that external timecode is no longer present. NOTE - External timecode must have been previously present for this error to occur, otherwise a status message is generated (see message 32775 and 32776).
1050	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB>External Timecode<TAB>Fixed <CR><LF>' This message is sent when the 6001 controller detects that external timecode is back again. NOTE - External timecode must have been previously present then lost to cause an error, otherwise a status message is generated (see message 32775 and 32776).
1051	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Internal Cues <TAB>Sub Module No.<TAB>XY<TAB>Fault<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>' This message is sent when the 6001 controller detects a fault on one of its internal Cues boards, where XY is the two digit sub module number, BBBBBBBB is either 'LOCAL ' if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards), and Z is a 3 or 4 indicating that the sub module sits on a 2633 or 2634 card.
1052	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Internal Cues <TAB>Sub Module No.<TAB>XY<TAB>Fixed<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>' This message is sent when the 6001 controller fixes a previously detected fault on one of its internal Cues boards, where XY is the two digit sub module number, BBBBBBBB is either 'LOCAL ' if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards), and Z is a 3 or 4 indicating that the sub module sits on a 2633 or 2634 card.
32768	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB>System A Active<CR><LF>' Indicate state of control frame A (if active) remote port currently connect too.
32769	'dd-mm-yy<TAB>HH:MM:SS:FF<TAB>System A Idle<CR><LF>'

COMMAND	Approximate String ‘ ‘ format and explanation
	Indicate state of control frame A (if idle) remote port currently connect too.
32770	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>System B Active<CR><LF>’ Indicate state of remote control frame B (if active) remote port currently connect too.
32771	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>System B Idle<CR><LF>’ Indicate state of remote control frame B (if idle) remote port currently connect too.
32772	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Device <TAB>Sub Module No.<TAB>XY<TAB>O.K.<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>’ This status message is part of the response to a Error and Status Request Message (message 79). The 6001 controller detects and generates a status message for all Device sub modules that are present and O.K., where XY is the two digit sub module number, Z is a 3 or 4 indicating that the sub module sits on a 2633 or 2634 card and BBBBBBBB is either ‘LOCAL ’ if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards).
32773	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Remote Control<TAB>Sub Module No.<TAB>XY<TAB>O.K.<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>’ This status message is part of the response to a Error and Status Request Message (message 79). The 6001 controller detects and generates a status message for all Remote Control sub modules that are present and O.K., where XY is the two digit sub module number, Z is a 3 or 4 indicating that the sub module sits on a 2633 or 2634 card and BBBBBBBB is either ‘LOCAL ’ if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards).
32774	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Matrix <TAB>Sub Module No.<TAB>XY<TAB>O.K.<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>’ This status message is part of the response to a Error and Status Request Message (message 79). The 6001 controller detects and generates a status message for all Matrix sub modules that are present and O.K., where XY is the two digit sub module number, Z is a 3 or 4 indicating that the sub module sits on a 2633 or 2634 card and BBBBBBBB is either ‘LOCAL ’ if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards).
32775	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>External Timecode<TAB>Absent <CR><LF>’ This status message is part of the response to a Error and Status Request Message (message 79). The 6001 controller sends this message if it detects that external timecode is not present.
32776	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>External Timecode<TAB>Present<CR><LF>’ This status message is part of the response to a Error and Status Request Message (message 79). The 6001 controller sends this message if it detects that external timecode is present.
32777	‘dd-mm-yy<TAB>HH:MM:SS:FF<TAB>Internal Cues <TAB>Sub Module No.<TAB>XY<TAB>O.K.<TAB>263Z switch:<TAB>BBBBBBBB<CR><LF>’ This status message is part of the response to a Error and Status Request Message (message 79). The 6001 controller detects and generates a status message for all internal Cues sub modules that are present and O.K., where XY is the two digit sub module number, Z is a 3 or 4 indicating that the sub module sits on a 2633 or 2634 card and BBBBBBBB is either ‘LOCAL ’ if the sub module is on the local 2633/4 (i.e. within the same 3U frame as the 2633 control card) or the binary switch setting on the expansion 2634 (i.e. within a 3U expansion frame containing only 2634 cards).

3.2.29. Pre-Processing Config Response Message

This message is issued by a Nucleus controller in response to the Pre-Processing config request message (Command byte 81). This message indicates the controllers support for Pre-Processing input control. Pre-Processing allows inputs on an input card to be routed out on Pre-Processing outputs and Pre-Processing inputs to be routed in place of inputs on the card.

Command Byte: 82

Message		
Byte	**Field Format**	**Notes**
Byte 1	Num of PPIO	Number of Pre-Processing I/O per input card
Byte 2	Num of subs	Number of substitutable inputs per input card
Byte 3	Num I/O Cards	Number of input cards multiplier DIV 128
Byte 4	Num I/O Cards	Number of input cards MOD 128

3.2.30. Pre-Processing Tally Message

This message returns Pre-Processing tally information in response to a Pre-Processing interrogate message (83).

Command Byte: 85

Message		
Byte	**Field Format**	**Notes**
Byte 1	I/O Card mult	Input card DIV 128
Byte 2	I/O Card num	Input card MOD 128
Byte 3	PPIO num	Pre-Processing I/O No (0 to No of PPIO per card – 1)
Byte 4	Input to PPIO	Input to PPIO output (127 = none)
Byte 5	PPIO to Input	PPIO to Input (127 = none)

3.2.31. Pre-Processing Connected Message

This message is issued by a device after it has made / cleared a Pre-Processing route usually in response to a Pre-Processing connect message (84). It is issued on ALL ports of the interface device.

Command Byte: 86

Message		
Byte	**Field Format**	**Notes**
Byte 1	I/O Card mult	Input card DIV 128
Byte 2	I/O Card num	Input card MOD 128
Byte 3	PPIO num	Pre-Processing I/O No (0 to No of PPIO per card – 1)
Byte 4	Input to PPIO	Input to PPIO output (127 = none)
Byte 5	PPIO to Input	PPIO to Input (127 = none)

3.3. MESSAGE LAYER – EXTENDED GENERAL COMMANDS

Due to the increase in size of routers needed to be controlled by remote devices (greater than 1023 sources and/or destinations and greater than 16 matrices and/or levels) the extended command set has been added.

In general if the remote device uses an extended command to get status from the controller then the controller will use the appropriate extended command for the response otherwise the controller will determine whether to use the normal or extended commands by looking at the source and destination content of the message and if either the source or destination is greater than 1023 then the extended command will be used otherwise the normal command is used. In addition, the extended command will be used if the matrix or level number is over 15.

The following sections define the extended command set.

3.4. Extended General Commands Received

3.4.1. Extended Crosspoint Interrogate Message

This message is a request for Tally information by matrix no., level and destination, issued by the remote device. The controller will respond to this message with an EXTENDED CROSSPOINT TALLY message (Command Byte 131).

Command Byte: 129

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest multiplier	Destination number DIV 256
Byte 4	Dest number	Destination number MOD 256

3.4.2. Extended Crosspoint Connect Message

This message is issued by the remote device in order to set crosspoints. The controller will respond with an EXTENDED CONNECTED message (Command Byte 132).

Command Byte: 130

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest multiplier	Destination number multipler DIV 256
Byte 4	Dest number	Destination number MOD 256
Byte 5	Source mult	Source number multiplier DIV 256
Byte 6	Source num	Source number MOD 256

3.4.3. Extended Protect Interrogate Message

This message is issued by the remote device to get the current protect status of a destination. The controller will respond with an EXTENDED PROTECT TALLY message (Command Byte 139).

Command Byte: 138

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest multiplier	Destination number DIV 256
Byte 4	Dest number	Destination number MOD 256

3.4.4. Extended Protect Connect Message

This is issued by the remote device to protect a destination. The controller will respond with a an EXTENDED PROTECT CONNECTED message (Command Byte 141).

Command Byte: 140

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest multiplier	Destination number multipler DIV 256
Byte 4	Dest number	Destination number MOD 256
Byte 5	Device mult	Device number multiplier DIV 256
Byte 6	Device num	Device number MOD 256

3.4.5. Extended Protect Dis-Connect Message

This message is issued by the remote device to remove protection from a destination. The controller will issue an EXTENDED PROTECT DIS-CONNECTED message (Command Byte 143) in response.

Command Byte: 142

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest multiplier	Destination number multipler DIV 256
Byte 4	Dest number	Destination number MOD 256
Byte 5	Device mult	Device number multiplier DIV 256
Byte 6	Device num	Device number MOD 256

3.4.6. Extended Protect Tally Dump Request Message

This message allows all the Protect information to be requested from the Pro-Bel Controller when a Remote device initialises. The controller will respond with an EXTENDED PROTECT TALLY DUMP message (Command Byte 148).

Command Byte: 147

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest multiplier	Destination number DIV 256
Byte 4	Dest number	Destination number MOD 256

3.4.7. Extended Crosspoint Tally Dump Request Message

This message is issued by the remote device to request a tally table dump for a given matrix/level combination. The controller will respond with an EXTENDED CROSSPOINT TALLY DUMP (word) message (Command Byte 151).

Command Byte: 149

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	

3.4.8. Extended Router I/O Parameters Interrogate Message

This message is issued by a remote device to request router source and destination parameter data from XD and Freeway routers. The controller will respond with an EXTENDED ROUTER I/O PARAMETERS TALLY message (Command Byte 154).

Command Byte: 153

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest multiplier	Type Select as defined in 3.1.12
Byte 4		Start Dest or Source Number DIV 256
Byte 5		Start Dest or Source MOD 256
Byte 6	Dest number	No. of Dests or Sources (1 - 64) (Maximum of 64)

3.4.9. Extended Router I/O Parameters Connect Message

This message is issued by a remote device to set router source and destination parameter data on XD and Freeway routers. The controller will respond with an EXTENDED ROUTER I/O PARAMETERS CONNECTED message (Command Byte 156).

Command Byte: 155

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Type Select	Type Select as defined in 3.1.12
Byte 4	Dest/Src mult	Dest or Source number multiplier DIV 256
Byte 5	Dest/Src num	Dest or Source number MOD 256
Byte 6	Dest number	Dest/Source Parameter as defined in 3.1.13

3.4.10. Extended All Source Names Request Message

This message is issued by the remote device to request the names for all the sources on a given matrix and level. The controller will respond with one or more EXTENDED SOURCE NAME RESPONSE messages (Command Byte 234).

Command Byte: 228

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Len of names 00 01 02 03 04	Length of Names Required 4-char names 8-char names 12-char names 16-char names 32-char names

3.4.11. Extended Single Source Name Request Message

This message is issued by the remote device to request the name for a single source. The controller will respond with a single EXTENDED SOURCE NAME RESPONSE message (Command Byte 234).

Command Byte: 229

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Len of names	Length of Names Required as defined in 3.4.10
Byte 4	Src mult	Source number multiplier DIV 256
Byte 5	Src number	Source number MOD 256

3.4.12. Extended All Destinations Association Names Request Message

This message is issued by the remote device to request the names for all the destination associations for a given matrix. The controller will respond with one or more EXTENDED DESTINATION ASSOCIATION NAMES RESPONSE messages (Command Byte 235).

Command Byte: 230

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Len of names	Length of Names Required as defined in 3.4.10

3.4.13. Extended Single Destination Association Names Request Message

This message is issued by the remote device to request the name for a single destination association for a given matrix. The controller will respond with one EXTENDED DESTINATION ASSOCIATION NAMES RESPONSE message (Command Byte 235).

Command Byte: 231

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Len of names	Length of Names Required as defined in 3.4.10
Byte 3	Dest Ass mult	Destination Association number DIV 256
Byte 4	Dest Ass num	Destination Association number MOD 256

3.4.14. Extended All UMD Labels Request Message

This message is issued by the remote device to request for all the alternative UMD labels on a given matrix. The controller will respond with one or more EXTENDED UMD LABELS RESPONSE messages (Command Byte 236).

Command Byte: 232

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Type of Labels 00 01 02 03	Type of Labels required 4-char labels 8-char labels 12-char labels 16-char labels

3.4.15. Extended Single UMD Label Request Message

This message is issued by the remote device to request a single alternative UMD labels on a given matrix. The controller will respond with a single EXTENDED UMD LABEL RESPONSE message (Command Byte 236).

Command Byte: 233

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Type of Labels	Type of Labels required as defined in 3.4.15
Byte 3	Source mult	Source Number multiplier DIV 256
Byte 4	Source num	Source Number MOD 256

3.4.16. Extended Crosspoint Connect On Go Group Salvo Message

This message is issued by the remote device to set up salvo switches. Routing information is held in the receiving device until activated by CROSSPOINT GO GROUP SALVO command (Command Byte 121). The controller will respond with an EXTENDED CROSSPOINT CONNECT ON GO GROUP SALVO ACKNOWLEDGE message (Command Byte 250) to indicate that the routing information has been stored successfully.

N.B. The group salvo commands are only implemented on the XD and ECLIPSE router ranges.

Command Byte: 248

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest mult	Destination number multiplier DIV 256
Byte 4	Dest num	Destination number MOD 256
Byte 5	Src mult	Source number DIV 256
Byte 6	Src num	Source number MOD 256
Byte 7	Salvo num Bit[7] Bits[0-6]	Holds the Salvo group number to configure 0 Salvo number 0-127. Destination and source will always overwrite previous data.

3.4.17. Extended Crosspoint Salvo Group Interrogate Message

This message is issued by the remote device to request the status of the GROUP SALVOS. The controller will respond with an EXTENDED CROSSPOINT GROUP SALVO TALLY message (Command Byte 253).

N.B. The group salvo commands are only implemented on the XD and ECLIPSE router ranges.

Command Byte: 252

Message		
Byte	**Field Format**	**Notes**
Byte 1	Salvo Group	SALVO GROUP number as defined in 3.1.29
Byte 2	Index mult	Connect index DIV 256
Byte 3	Index num	Connect index MOD 256

The connect index specifies the index into the SALVO GROUP specified in byte1. This command is called recursively from connect index 0, until no crosspoint data in the specified group is left.

3.5. Extended General Commands Transmitted

3.5.1. Extended Crosspoint Tally Message

This message returns router tally information in response to an EXTENDED CROSSPOINT INTERROGATE message (Command Byte 129).

Command Byte: 131

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest mult	Destination number multiplier DIV 256
Byte 4	Dest num	Destination number MOD 256
Byte 5	Src mult	Source number DIV 256
Byte 6	Src num	Source number MOD 256
Byte 7	Status	(For future use)

3.5.2. Extended Crosspoint Connected Message

This message is issued spontaneously by the controller on all ports after it has confirmation that a route has been made through the router. The message is effectively broadcast.

Command Byte: 132

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest mult	Destination number multiplier DIV 256
Byte 4	Dest num	Destination number MOD 256
Byte 5	Src mult	Source number DIV 256
Byte 6	Src num	Source number MOD 256
Byte 7	Status	(For future use)

3.5.3. Extended Protect Tally Message

This message is issued by a controller in response to an EXTENDED PROTECT INTERROGATE message (Command Byte 138). It returns the current protect status of a destination.

Command Byte: 139

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte3	Protect	Protect details as defined in 3.2.5
Byte 4	Dest mult	Destination number multiplier DIV 256
Byte 5	Dest num	Destination number MOD 256
Byte 6	Device mult	Device number DIV 256
Byte 7	Device num	Device number MOD 256

3.5.4. Extended Protect Connected Message

This message is issued by the controller in when the protect data is altered and also if the data was unsuccessfully altered as a result of an EXTENDED PROTECT CONNECT message (Command Bytes 140). This message is broadcast on all ports.

Command Byte: 141

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte3	Protect	Protect details as defined in 3.2.5
Byte 4	Dest mult	Destination number multiplier DIV 256
Byte 5	Dest num	Destination number MOD 256
Byte 6	Device mult	Device number DIV 256
Byte 7	Device num	Device number MOD 256

Following the issue of an extended protect connected message SYSTEM3 will then issue a PROTECT DEVICE NAME REQUEST message (Command Byte 17). The remote device should then respond with a PROTECT DEVICE NAME RESPONSE message (Command Byte 18).

3.5.5. Extended Protect Dis-Connected Message

This command is issued by the controller when the protect data is altered i.e. a destination has been unprotected and also if the data was unsuccessfully altered as a result of an EXTENDED PROTECT DIS-CONNECT message (Command Bytes 142). This message is broadcast on all ports.

Command Byte: 143

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte3	Protect	Protect details as defined in 3.2.5
Byte 4	Dest mult	Destination number multiplier DIV 256
Byte 5	Dest num	Destination number MOD 256
Byte 6	Device mult	Device number DIV 256
Byte 7	Device num	Device number MOD 256

3.5.6. Extended Protect Tally Dump Message

This message is issued by the Pro-Bel Controller in response to an EXTENDED PROTECT DUMP REQUEST (Command Byte 147). It returns all the Protect Information.

Command Byte: 148

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte3	Num of Protect	Number of Protect Tallies (Maximum 64)
Byte 4	1st Dest mult	First Destination number multiplier DIV 256
Byte 5	1st Dest num	First Destination number MOD 256
Byte 6&7	Device/protect Bit[15] Bits[12-14] 0 1 2 3 Bits[10-11] Bits[0-9]	First Device Number & Protect Data Not used Protected data Indicates Not Protected Indicates Pro-Bel Protected Pro-Bel override Protected *(Cannot be altered remotely)* Indicates OEM Protected Not used Device number
Byte 8&9	Device/protect	Second Device Number & Protect Data (As bytes 6 & 7 above)
Etc ...		

The message length is limited to 133 bytes; therefore a message that exceeds this will require more than one message.

3.5.7. Extended Crosspoint Tally Dump (Word) Message

This message is issued by a controller in response to an EXTENDED CROSSPOINT TALLY DUMP REQUEST (Command Byte 149). It provides tally table data for a given matrix/level combination. This message assumes a maximum destination/source number of 65535.

Command Byte: 151

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte3	Num of Tallies	Number of tallies returned (Maximum 64)
Byte 4	1st Dest mult	First Destination number multiplier DIV 256
Byte 5	1st Dest num	First Destination number MOD 256
Byte 6	1st Src mult	First Source number multiplier DIV 256
Byte 7	1st Src num	First Source number MOD 256
Byte 8	2nd Src mult	Second source number DIV 256
Byte 9	2nd Src num	Second source number MOD 256
Etc ...		

The message length is limited in total to 133 bytes, thus, tables that would exceed this length would cause more than one message to be issued with the appropriate destination value in the bytes 4 and 5 of the message.

3.5.8. Extended Router I/O Parameters Tally Message

This message is issued by a controller in response to an EXTENDED ROUTER I/O PARAMETER INTERROGATE message (Command Byte 153).

Command Byte: 154

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte3	Type Selected	Type Select as defined in 3.1.12
Byte 4	Start Dest/Src	Start Dest or Source multiplier DIV 256
Byte 5	Start Dest/Src	Start Destination or Source MOD 256
Byte 6	Num Dest/Src	No. of Destinations or Sources (max 64)
Byte 7	1st Dest/Src	First Dest/Source's Parameter as defined in byte 5 in 3.1.13
Byte 8	2nd Dest/Src	Second Dest/Source's Parameter
Etc ...		

3.5.9. Extended Router I/O Parameter Connected Message

This message is issued spontaneously by the controller on all ports after it has confirmation that a router I/O parameter has been changed, though generally in response to an EXTENDED ROUTER I/O PARAMETER CONNECT message (Command Byte 155).

Command Byte: 156

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte3	Type Selected	Type Select as defined in 3.1.12
Byte 4	Dest/Src mult	Dest or Source multiplier DIV 256
Byte 5	Dest/Src num	Destination or Source MOD 256
Byte 6	Parameter	Dest or Sources Parameter as defined in byte 5 of 3.1.13

3.5.10. Extended Source Names Response Message

This command is issued by the controller in response to EXTENDED ALL SOURCE NAMES REQUEST and EXTENDED SINGLE SOURCE NAME REQUEST messages (Command Bytes 228 and 229). The message length is limited to 135 bytes (except for 32 character names where it's 231), thus name tables that would exceed this length would cause more than one message to be issued with the appropriate source number value in the fourth field of the message.

Command Byte: 234

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Length of name	Length of Source Names Returned 3.4.10
Byte 4	1st Src mult	First Source number multiplier DIV 256
Byte 5	1st Src num	First Source number MOD 256
Byte 6	Num of names	Number of Source Names to follow
If Byte 3 = 00 (4 char names)		
Byte 7-10	1st Name	1st 4-char Source name
Byte 11-14	2nd Name	2nd 4-char Source name
Byte 15-18	3rd Name	3rd 4-char Source name
Etc ...		
If Byte 3 = 01 (8 char names)		
Byte 7-14	1st Name	1st 8-char Source name
Byte 15-22	2nd Name	2nd 8-char Source name
Etc ...		
If Byte 3 = 02 (12 char names)		
Byte 7-18	1st Name	1st 12-char Source name
Byte 19-30	2nd Name	2nd 12-char Source name
Etc ...		
If Byte 2 = 03 (16 char names)		
Bytes 6 – 21	1st Name	First 16-char Source name
Bytes 22-37	2nd Name	Second 16-char Source name
Etc …		
If Byte 2 = 04 (32 char names)		
Bytes 6 - 37	1st Name	First 32-char Source name
Bytes 38 - 69	2nd Name	Second 32-char Source name
Etc …		

Note: Byte 6 will always be set to 01 in response to command 229. Also, in this message, maximum of 32 for 4 char names, 16 for 8 char names, 10 for 12 char names, 8 for 16 char names and 7 for 32 char names.

3.5.11. Extended Destination Association Names Response Message

This command is issued by the controller in response to an EXTENDED ALL DESTINATION ASSOCIATION NAMES REQUEST or EXTENDED SINGLE DESTINATION ASSOCIATION NAME REQUEST messages (Command Bytes 230 and 231). The message length restrictions in 3.5.10 apply (except for being 1 byte less).

Command Byte: 235

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Length of name	Length of Destination Association Names Returned 3.4.10
Byte 3	1st Dest Assoc mult	First Destination Association number multiplier DIV 256
Byte 4	1st Dest Assoc num	First Destination Association number MOD 256
Byte 5	Num of names	Number of Destination Association Names to follow.
If Byte 3 = 00 (4 char names)		
Byte 6-9	1st Name	1st 4-char Destination Association name
Byte 10-13	2nd Name	2nd 4-char Destination Association name
Byte 14-17	3rd Name	3rd 4-char Destination Association name
Etc ...		
If Byte 3 = 01 (8 char names)		
Byte 6-13	1st Name	1st 8-char Destination Association name
Byte 14-21	2nd Name	2nd 8-char Destination Association name
Etc ...		
If Byte 3 = 02 (12 char names)		
Byte 6-17	1st Name	1st 12-char Destination Association name
Byte 18-29	2nd Name	2nd 12-char Destination Association name
Etc ...		
If Byte 2 = 03 (16 char names)		
Bytes 6 – 21	1st Name	First 16-char Destination Association name
Bytes 22-37	2nd Name	Second 16-char Destination Association name
Etc …		
If Byte 2 = 04 (32 char names)		
Bytes 6 - 37	1st Name	First 32-char Destination Association name
Bytes 38 - 69	2nd Name	Second 32-char Destination Association name
Etc …		

Note: Byte 6 will always be set to 01 in response to command 231. Also, in this message, maximum of 32 for 4 char names, 16 for 8 char names, 10 for 12 char names, 8 for 16 char names and 7 for 32 char names.

3.5.12. Extended UMD Labels Response Message

This command is issued by the controller in response to an EXTENDED ALL UMD LABELS REQUEST or EXTENDED SINGLE UMD LABEL REQUEST message (Command bytes 232 and 233). The message length restrictions in 3.5.10 apply (except for being 1 byte less).

Command Byte: 236

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Length of label	Length of Label Returned 3.4.14
Byte 3	1st Src mult	First Source number multiplier DIV 256
Byte 4	1st Src num	First Source number MOD 256
Byte 5	Num of names	Number of Source labels to follow.
If Byte 3 = 00 (4 char label)		
Byte 6-9	1st Label	1st 4-char Source Label
Byte 10-13	2nd Label	2nd 4-char Source Label
Etc ...		
If Byte 3 = 01 (8 char label)		
Byte 6-13	1st Label	1st 8-char Source Label
Byte 14-21	2nd Label	2nd 8-char Source Label
Etc ...		
If Byte 3 = 02 (12 char label)		
Byte 6-17	1st Label	1st 12-char Source Label
Byte 18-29	2nd Label	2nd 12-char Source Label
Etc ...		
If Byte 3 = 03 (16 char label)		
Byte 6-20	1st Label	1st 16-char Source Label
Byte 21-36	2nd Label	2nd 16-char Source Label
Etc ...		

Note: Byte 6 will always be set 01 in response to command 233. Also, in this message, maximum of 32 for 4 char labels, 16 for 8 char labels, 10 for 12-char labels and 8 for 16-char labels.

3.5.13. Extended Crosspoint Connect On Go Group Salvo Acknowledge Message

This message is issued by the controller in response to an EXTENDED CROSSPOINT CONNECT ON GO GROUP SALVO message (Command Byte 248).

N.B. The group salvo commands are only implemented on the XD and ECLIPSE router ranges.

Command Byte: 250

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest mult	Destination number multiplier DIV 256
Byte 4	Dest num	Destination number MOD 256
Byte 5	Src mult	Source number DIV 256
Byte 6	Src num	Source number MOD 256
Byte 7	Salvo Group	Salvo group number as defined in 3.1.29

3.5.14. Extended Crosspoint Group Salvo Tally Message

This message is issued by the controller in response to a CROSSPOINT GROUP SALVO INTERROGATE message (Command Byte 124). It returns the SALVO data for the specified SALVO GROUP. The data returned is for a particular index, the validity flag indicates whether any more data is present.

N.B. The group salvo commands are only implemented on the XD and ECLIPSE router ranges.

Command Byte: 253

Message		
Byte	**Field Format**	**Notes**
Byte 1	Matrix number	
Byte 2	Level number	
Byte 3	Dest mult	Destination number multiplier DIV 256
Byte 4	Dest num	Destination number MOD 256
Byte 5	Src mult	Source number DIV 256
Byte 6	Src num	Source number MOD 256
Byte 7	Salvo Group	Salvo group number as defined in 3.1.29
Byte 8	Connect index	Connect index DIV 256 as defined in 3.4.17
Byte 9	Connect index	Connect index MOD 256 as defined in 3.4.17
Byte 10	Validity flag 00 01 02	Validity flag Valid connect index returned, more data available Valid connect index returned, last in queue Invalid connect (no data in SALVO

4. MESSAGE LAYER – PROTOCOL COMMANDS

These commands are considered to be specific to the operation of protocol at the message level. As such, they provide a degree of high-level flow control, fault responses for invalid message formats and a method of determining the level of implementation of the protocol in any given application.

4.1. Protocol Commands Received

4.1.1. Implementation Request

This message is sent by the remote device in order to determine the message set that has been implemented in an application. The controller responds with an IMPLEMENTATION STATUS message (Command Byte 98).

Command Byte: 97

Message: There are no message bytes.

4.1.2. Invalid Message

This message is sent by either the controller or the remote device when a message is detected that matches the protocol structure but has an invalid format.

Any device that issues command 99, the invalid message command, MUST also accept the message as a valid input message, otherwise two such devices will enter an endless loop. In other words, issuing an invalid message command in response to an invalid message command is NOT permissible.

Command Byte: 99

Message		
Byte	**Field Format**	**Notes**
Byte 1		Command code of invalid message

4.2. Protocol Commands Transmitted

4.2.1. Busy Status

This BUSY status is sent immediately that the controller is unable to accept another command for the specified matrix due to operations on that matrix (such as auto-fade etc..). It is used for flow control. The NOT BUSY status is sent when the controller is able to accept another command for the specified matrix.

Command Byte: 86

Message		
Byte	**Field Format**	**Notes**
Byte 1		System number
Byte 2-3	Matrix	Matrix numbers (16 bit - bit map)
Byte 4	Status 01 02	Status number Busy Not busy

4.2.2. Implementation Status

This message is issued in response to the IMPLEMENTATION REQUEST message (Command Byte 97) and returns a list of all commands that are implemented by the application.

Command Byte: 98

Message		
Byte	**Field Format**	**Notes**
Byte 1	Num Outputs	Number of output commands (e.g. 'm')
Byte 2	Num inputs	Number of input commands (e.g. 'n')
Byte 3	1st output cmd	First output command number
Byte 4	2nd output cmd	Second output command number
Etc...		
Byte m+3	Last output	Last output command number
Byte m+4	1st input cmd	First input command number
Byte m+n+3	Last input cmd	Last input command number

5. MESSAGE LAYER - EDITOR SPECIFIC COMMANDS

These commands are considered to be specific to the operation of the SYSTEM2, SYSTEM3 and Freeway editors and the associated database structure.

Details of these are found in Part II of the General Remote Control Communication Protocol (SW-P-74).

6. MESSAGE LAYER - PRESENTATION MIXER SPECIFIC COMMANDS

These commands are considered to be specific to the operation of Presentation/Master Control mixers. Details of these can be found in Part II of the General Remote Control Communication Protocol (SW-P-74).

7. TDM 4 WIRE ROUTER

These commands are used to control the TDM 4 Wire Router. Details of these commands can be found in Part II of the General Remote Control Communication Protocol (SW-P-74).

8. APPLICATION EXAMPLES

This section details the command sets used within the four main Pro-Bel system controllers, namely SYSTEM2, SYSTEM3, AURORA, FREEWAY/AXIS/SIRIUS/NEBULA and NUCLEUS.

The table below shows which commands that are defined in this document are implemented on each of the above mentioned controllers (a similar table exits in SW-P-74 (Part II) for the commands defined in that document):

KEY:	**SYSTEM:**
01	SYSTEM2
02	SYSTEM3 (8400 VARIANTS)
03	SYSTEM3 (6000 VARIANTS)
04	FREEWAY/AXIS/HALO/SIRIUS/NEBULA
05	AURORA/SYSTEM3 (6006/6001 VARIANTS)
06	NUCLEUS (2450)
07	CENTRA/NUCLEUS2 CONTROLLERS (2330/246x)

INPUT COMMANDS									
COMMAND			**SYSTEM**						
NO	**NAME**	**REF**	**01**	**02**	**03**	**04**	**05**	**06**	**07**
00	COMMAND ENABLE/DISABLE SET/READ	3.1.1					XX		XX
01	CROSSPOINT INTERROGATE	3.1.2	XX	XX	XX	XX	XX	XX	XX
02	CROSSPOINT CONNECT	3.1.3	XX	XX	XX	XX	XX	XX	XX
07	MAINTENANCE	3.1.4				XX			
08	DUAL CONTROLLER STATUS REQUEST	3.1.5				XX	XX		
10	PROTECT INTERROGATE	3.1.6		XX	XX	XX	XX	XX	XX
12	PROTECT CONNECT	3.1.7		XX	XX	XX	XX	XX	XX
14	PROTECT DIS-CONNECT	3.1.8		XX	XX	XX	XX	XX	XX
17	PROTECT DEVICE NAME REQUEST	3.1.9		XX	XX	XX	XX		XX
19	PROTECT TALLY DUMP REQUEST	3.1.10		XX	XX	XX	XX	XX	XX
21	CROSSPOINT TALLY DUMP REQUEST	3.1.11	XX	XX	XX	XX	XX	XX	XX
25	ROUTER I/O PARAMETERS INTERROGATE	3.1.12				XX		XX	XX
27	ROUTER I/O PARAMETERS CONNECT	3.1.13				XX		XX	XX

OUTPUT COMMANDS									
COMMAND			**SYSTEM**						
NO	**NAME**	**REF**	**01**	**02**	**03**	**04**	**05**	**06**	**07**
00	COMMAND ENABLE/DISABLE SET/READ RESPONSE	3.2.1					XX		XX
03	CROSSPOINT TALLY	3.2.2	XX	XX	XX	XX	XX	XX	XX
04	CROSSPOINT CONNECTED	3.2.3	XX	XX	XX	XX	XX	XX	XX
09	DUAL CONTROLLER STATUS RESPONSE	3.2.4				XX	XX		
11	PROTECT TALLY	3.2.5		XX	XX	XX	XX	XX	XX
13	PROTECT CONNECTED	3.2.6		XX	XX	XX	XX	XX	XX
15	PROTECT DIS-CONNECTED	3.2.7		XX	XX	XX	XX	XX	XX
18	PROTECT DEVICE NAME RESPONSE	3.2.8		XX	XX	XX	XX		XX
20	PROTECT TALLY DUMP RESPONSE	3.2.9		XX	XX	XX	XX	XX	XX
22	CROSSPOINT TALLY DUMP (byte) RESPONSE	3.2.10	XX	XX	XX	XX	XX	XX	XX
23	CROSSPOINT TALLY DUMP (word) RESPONSE	3.2.11	XX	XX	XX		XX	XX	XX
26	ROUTER I/O PARAMETERS TALLY	3.2.12				XX		XX	XX
28	ROUTER I/O PARAMETERS CONNECTED	3.2.13				XX		XX	XX

INPUT COMMANDS									
COMMAND			SYSTEM						
NO	NAME	REF	01	02	03	04	05	06	07
29	MASTER PROTECT CONNECT	3.1.14					XX	XX	XX
41	DIAGNOSTIC REQUEST	3.1.15				XX			
76	STATUS CONFIGURATION AND ENABLE FLAGS REQUEST MESSAGE	3.1.32					XX		
79	ERROR AND STATUS REQUEST MESSAGE	3.1.33					XX		
81	PRE-PROCESSING CONFIG REQUEST	3.1.34						XX	
83	PRE-PROCESSING INTERROGATE	3.1.35						XX	
84	PRE-PROCESSING CONNECT	3.1.36						XX	
87	SOFT KEY ASSIGNMENT TALLY REQUEST	3.1.16					XX		
89	SOFT KEY ASSIGNMENT SET REQUEST	3.1.17					XX		
100	ALL SOURCE NAMES REQUEST	3.1.18		XX	XX	XX	XX	XX	XX
101	SINGLE SOURCE NAME REQUEST	3.1.19		XX	XX	XX	XX	XX	XX
102	ALL DESTINATION ASSOCIATION NAMES REQUEST	3.1.20		XX	XX	XX	XX	XX	XX
103	SINGLE DESTINATION ASSOCIATION NAMES REQUEST	3.1.21		XX	XX	XX	XX	XX	XX
104	ALL UMD LABELS REQUEST	3.1.22		XX	XX		XX		
105	SINGLE UMD LABELS REQUEST	3.1.23		XX	XX		XX		
111	CROSSPOINT TIE-LINE CONNECT	3.1.27		XX	XX	XX	XX	XX	
112	CROSSPOINT TIE-LINE INTERROGATE	3.1.28		XX	XX	XX	XX	XX	
114	ALL SOURCE ASSOCIATION NAMES REQUEST	3.1.24					XX	XX	XX
115	SINGLE SOURCE ASSOCIATION NAME REQUEST	3.1.25					XX	XX	XX

OUTPUT COMMANDS									
COMMAND			SYSTEM						
NO	NAME	REF	01	02	03	04	05	06	07
30	NAMES UPDATED	3.2.14					XX		
43	DIAGNOSTIC RESPONSE	3.2.15				XX			
44	LOG MESSAGE	3.2.16		XX	XX		XX		
77	STATUS CONFIGURATION AND ENABLE FLAGS TALLY MESSAGE	3.2.27					XX		
78	LOGGING STRING MESSAGES	3.2.28					XX		
82	PRE-PROCESSING CONFIG RESPONSE	3.2.29						XX	
85	PRE-PROCESSING TALLY	3.2.30						XX	
86	PRE-PROCESSING CONNECTED	3.2.31						XX	
88	SOFT KEY ASSIGNMENT TALLY RESPONSE	3.2.17					XX		
90	SOFT KEY ASSIGNMENT SET RESPONSE	3.2.18					XX		
106	SOURCE NAME RESPONSE	3.2.19		XX	XX	XX	XX	XX	XX
106	SOURCE NAME RESPONSE	3.2.19		XX	XX	XX	XX	XX	XX
107	DESTINATION ASSOCIATION NAME RESPONSE	3.2.20		XX	XX	XX	XX	XX	XX
107	DESTINATION ASSOCIATION NAME RESPONSE	3.2.20		XX	XX	XX	XX	XX	XX
108	UMD LABEL RESPONSE	3.2.21		XX	XX		XX		
108	UMD LABEL RESPONSE	3.2.21		XX	XX		XX		
113	CROSSPOINT TIE-LINE TALLY	3.2.23		XX	XX	XX	XX	XX	
116	SOURCE ASSOCIATION NAMES RESPONSE	3.2.22					XX	XX	XX
116	SOURCE ASSOCIATION NAMES RESPONSE	3.2.22					XX	XX	XX

INPUT COMMANDS									
COMMAND			SYSTEM						
NO	NAME	REF	01	02	03	04	05	06	07
117	UPDATE NAME REQUEST	3.1.26					XX		XX
120	CROSSPOINT CONNECT ON GO GROUP SALVO	3.1.29			XX	XX	XX		
121	CROSSPOINT GO GROUP SALVO	3.1.30			XX	XX	XX		
124	CROSSPOINT GROUP SALVO INTERROGATE	3.1.31			XX	XX	XX		
129	EXTENDED CROSSPOINT INTERROGATE	3.4.1					XX	XX	XX
130	EXTENDED CROSSPOINT CONNECT	3.4.2					XX	XX	XX
138	EXTENDED PROTECT INTERROGATE	3.4.3					XX	XX	XX
140	EXTENDED PROTECT CONNECT	3.4.4					XX	XX	XX
142	EXTENDED PROTECT DIS-CONNECT	3.4.5					XX	XX	XX
147	EXTENDED PROTECT TALLY DUMP REQUEST MESSAGE	3.4.6					XX	XX	XX
149	EXTENDED CROSSPOINT TALLY DUMP REQUEST MESSAGE	3.4.7					XX	XX	XX
153	EXTENDED ROUTER I/O PARAMETERS INTERROGATE	3.4.8					XX		XX
155	EXTENDED ROUTER I/O PARAMETERS CONNECT	3.4.9					XX		XX
228	EXTENDED ALL SOURCE NAMES REQUEST MESSAGE	3.4.10					XX	XX	XX
229	EXTENDED SINGLE SOURCE NAME REQUEST MESSAGE	3.4.11					XX	XX	XX
230	EXTENDED ALL DESTINATIONS ASSOCIATION NAMES REQUEST MESSAGE	3.4.12					XX	XX	XX
231	EXTENDED SINGLE DESTINATION ASSOCIATION NAMES REQUEST MESSAGE	3.4.13					XX	XX	XX

OUTPUT COMMANDS									
COMMAND			SYSTEM						
NO	NAME	REF	01	02	03	04	05	06	07
122	CROSSPOINT CONNECT ON GO GROUP SALVO ACKNOWLEDGE	3.2.24			XX	XX	XX		
123	CROSSPOINT GO GROUP SALVO ACKNOWLEDGE	3.2.25			XX	XX	XX		
125	CROSSPOINT GROUP SALVO TALLY	3.2.26			XX	XX	XX		
131	EXTENDED CROSSPOINT TALLY	3.5.1					XX	XX	XX
132	EXTENDED CROSSPOINT CONNECTED	3.5.2					XX	XX	XX
139	EXTENDED PROTECT TALLY	3.5.3					XX	XX	XX
141	EXTENDED PROTECT CONNECTED	3.5.4					XX	XX	XX
143	EXTENDED PROTECT DIS-CONNECTED	3.5.5					XX	XX	XX
148	EXTENDED PROTECT TALLY DUMP MESSAGE	3.5.6					XX	XX	XX
151	EXTENDED CROSSPOINT TALLY DUMP (WORD) MESSAGE	3.5.7					XX	XX	XX
154	EXTENDED ROUTER I/O PARAMETERS TALLY	3.5.8					XX		XX
156	EXTENDED ROUTER I/O PARAMETERS CONNECTED	3.5.9					XX		XX
234	EXTENDED SOURCE NAMES RESPONSE MESSAGE	3.5.10					XX	XX	XX
234	EXTENDED SOURCE NAMES RESPONSE MESSAGE	3.5.10					XX	XX	XX
235	EXTENDED DESTINATION ASSOCIATION NAMES RESPONSE MESSAGE	3.5.11					XX	XX	XX
235	EXTENDED DESTINATION ASSOCIATION NAMES RESPONSE MESSAGE	3.5.11					XX	XX	XX

INPUT COMMANDS									
COMMAND			SYSTEM						
NO	NAME	REF	01	02	03	04	05	06	07
232	EXTENDED ALL UMD LABELS REQUEST MESSAGE	3.4.14					XX		
233	EXTENDED SINGLE UMD LABEL REQUEST MESSAGE	3.4.15					XX		
248	EXTENDED CROSSPOINT CONNECT ON GO GROUP SALVO	3.4.16					XX		
252	EXTENDED CROSSPOINT SALVO GROUP INTERROGATE MESSAGE	3.4.17					XX		
97	IMPLEMENTATION REQUEST	4.1.1				XX			
99	INVALID MESSAGE	4.1.2				XX			

OUTPUT COMMANDS									
COMMAND			SYSTEM						
NO	NAME	REF	01	02	03	04	05	06	07
236	EXTENDED UMD LABELS RESPONSE MESSAGE	3.5.12					XX		
236	EXTENDED UMD LABELS RESPONSE MESSAGE	3.5.12					XX		
250	EXTENDED CROSSPOINT CONNECT ON GO GROUP SALVO ACKNOWLEDGE	3.5.13					XX		
253	EXTENDED CROSSPOINT GROUP SALVO TALLY	3.5.14					XX		
98	IMPLEMENTATION STATUS	4.2.2				XX			
99	INVALID MESSAGE	4.1.2				XX			XX

NOTES:

I. For SYSTEM2 type controllers the matrix no. is always 01 and the level no. starts from 01.

II. For SYSTEM3 and AURORA type controllers the matrix no. and level no. start from 00.

III. For controller type 4, the matrix no. is always 00 and the level no. starts from 00.

IV. On controller type 4, only 2442/3 Nebula controllers fitted with V1.05 or above firmware and 2444/5 Nebula controllers fitted with V3.03 or above firmware support the Group Salvo commands 120, 121, 122, 123, 124 and 125.

V. On controller type 5, only Aurora controllers fitted with V1.10 or above firmware support the Soft Key Assignment command, 87, 88, 89 and 90.

VI. On controller type 5, only Aurora controllers fitted with V1.10 or above firmware support the Dual Controller Status request/response commands, 08 and 09.

VII. On controller type 5, only Aurora controllers fitted with V1.11 or above firmware support the Update Name request command, 117.

VIII. For commands 100, 101, 102, 103, 114, 115, 228, 229, 230, 231, 16 and 32 character names are only supported on controller type 7. (V3.15.6 or above).

9. APPENDIX 1 - GENERAL REMOTE CONTROL COMMUNICATION PROTOCOL EXAMPLES

There follows some examples of dialogue between a remote device and the Pro-Bel/Snell controller. The Pro-Bel/Snell controller is a slave (controlled) device. All communications are initiated by the remote device, except where the Pro-Bel/Snell controller spontaneously broadcasts a Connected message.

Notes relevant to all examples:

- All messages are shown in the HEX code.
- The ACK reply from the Pro-Bel/Snell controller indicates the message packaging is correct (i.e.correct format, byte count, checksum, etc). The ACK does not infer that the message content is valid.
- The Acknowledgement of message receipt by the remote device should be returned within one second. A message will be re-issued by the Pro-Bel/Snell controller a maximum of 5 times if no Acknowledge is received.
- Router sources and destinations are always indexed from 0 in the control procedure.
- All subsequent references to a `controller' refer to the Pro-Bel/Snell controller.

9.1. REQUEST PROTOCOL IMPLEMENTATION

Remote device to controller:

STX	COM ID	MESSAGE	BYTE COUNT	CHECKSUM	ETX
10 02	61		01	9E	10 03

Acknowledge from controller:

ACK
10 06

Requested data from controller:

STX	COM ID	MESSAGE	BYTE COUNT	CHECKSUM	ETX
10 02	62	08 09 03 04 16 20 22 53 62 63 01 02 15 1F 20 21 43 4A 61	14	9C	10 03

Acknowledge of message receipt by the remote device:

ACK
10 06

Notes

There is no message field for a protocol implementation request.

The protocol commands implemented, and thus the implementation reply may vary for different router applications.

9.2. INTERROGATE COMMAND

Interrogate router connection at destination 1.

Remote device to controller:

STX	COM ID	MESSAGE	BYTE COUNT	CHECKSUM	ETX
10 02	01	11 00 01	04	E9	10 03

Acknowledge from controller:

ACK
10 06

Requested data from controller - Tally reply

STX	COM ID	MESSAGE	BYTE COUNT	CHECKSUM	ETX
10 02	03	11 01 01 19	05	CC	10 03

Acknowledge of message receipt by the remote device:

ACK
10 06

Notes:

1 The first message byte indicates matrix 1, level 1.

2 The second byte bits 4, 5 & 6 are the destination index number divided by 128. All other bits are set 0. The third byte gives the remainder value. The example specifies destination index 01, which is the second router destination.

3 The Tally reply indicates source 153 routed at destination 1, level 1.

9.3. CONNECT COMMAND

This example is a Connect message to connect source 0 to destination 2 on level 1.

Remote device to controller:

STX	COM ID	MESSAGE	BYTE COUNT	CHECKSUM	ETX
10 02	02	11 00 02 00	05	E6	10 03

Acknowledge from controller:

ACK
10 06

Router connect delay … then:

Connected message from controller:

STX	COM ID	MESSAGE	BYTE COUNT	CHECKSUM	ETX
10 02	04	11 00 02 00	05	E4	10 03

Acknowledge of message receipt by the remote device:

ACK
10 06

10. APPENDIX II - UNIVERSAL PROTOCOL COMMAND SET

(See separate sheet SW-P-08C.DOC)

11. APPENDIX III – SW-P-08 OVER IP

This section details the implementation of SW-P-08 over IP on the 2330 and 2463 controllers.

11.1. System Overview

A TCP/IP server running on the controller will provide control and monitoring of a router over Ethernet. The controller is a 2330 or 2463.

Once a client has established a TCP connection to the server, the data sent is the same as would be send on a the serial link. All messages supported by the controller via serial are supported over IP. All parts of the serial protocol are required including the acknowledges.

11.2. Restrictions and constraints

The following restrictions and constraints apply:

- No guarantees can be made regarding timing of messages sent across a network. However, a dedicated network should provide a satisfactory degree of real-time control of routers.
- Up to 16 TCP clients can connect to the server at any one time.

11.3. System Operation

TCP/IP Server

The server will listen for connections from clients over a network, and will allow a limited number of clients to connect to the server at any one time.

Once a client connects to the server, the client can send raw SW-P-08 messages to the server. No packaging of the messages is required. Any messages sent from the router will in turn be sent to the client. A client is free to close the connection at any time, but would typically close the connection after receiving an expected response from the router.

The mechanism described implies that responsibility for opening and closing connections to the server lies with the client. This provides flexibility by allowing clients to manage connections in accordance with the requirements of the system.

For example, a small system consisting of a small number of clients and servers could operate using persistent TCP/IP connections, where the overhead of maintaining those connections is minimal. Alternatively, a large system consisting of a large number of clients and servers could operate using transient TCP/IP connections, where the overhead of maintaining persistent connections may be too great.

11.4. Configuration (2330 and 2463)

On these controllers the TCP/IP port used, is configured through Workbench in the “port configuration screen” of the “router device”.

11.5. Supporting dual controllers over IP.

In order to control a router with dual controllers over IP there are a number of considerations that need to be taken into account which are not required when controlling dual controllers serially

11.5.1. Overview

Using the protocol serially there is physically only one serial port which is changed over by the hardware to the active controller, but when used over IP the active and idle controller have different IP addresses, so they will need to configure two addresses and ensure they send crosspoint commands to the active controller. In order to support a dual IP connection, the SW-P-08 protocol supports messages to tell the controlling device which one is active. The controlling device then interrogates and controls the active controller. In more detail:

On all controllers (2330 and 2463) the messages "Dual controller status request" and "Dual controller status response" (08 and 09) need to be used. Bit 1 of byte 1 of the response is then used to indicate whether the controller is active or idle.

Within the protocol commands, DUAL CONTROLLER STATUS REQUEST (Command byte 08) is sent by the client and DUAL CONTROLLER STATUS RESPONSE (Command byte 09) sent by the controller. The response message contains a bit to say whether the controller is active or idle and is sent as a reply to the request command and also automatically when the status changes. The client software would need to open a connection to both controllers and determine which is active according to this message. If a changeover occurred it should automatically get this message on one of the connections indicating it has gone active (it is unlikely to get a message on the other link saying it has gone idle, since this usually means it has reset).

11.5.2. An example of dual control

During startup

1. Open connections 1 and 2.
2. Read the active status of each successfully opened connection (send a dual controller status request message (08)).
3. Determine which controller is active according to the response and whether the TCP connection opened successfully.
4. Send interrogate commands (01) to the active controller for all required crosspoints to get the current state of the router.
5. Continue to poll the idle controller with the appropriate active / idle status request message (to keep the connections open).
6. Continue to attempt to connect on any TCP connections that failed to open.

During switchover

1. This could be triggered by the loss of connection on the active controller and / or the slave controller reporting it is active.
2. If the previously idle controller is now active, send interrogate commands (01) to get the crosspoint status for all the destinations.

3. Continue to attempt to connect or send active / idle status requests to the now idle controller.

There are a number of ways of handling the dual connection depending on how the interface is to be used and the capabilities of the controlling device. The above is only an example.

11.5.3. Notes on example

1. Rather than polling for active / idle status the controlling device can rely on being told a controller has become active. This would be indicated by an unsolicited “dual controller status response (09)” being received on the TCP connection.

2. Although interrogates need to be send to the active controller only, a simple approach for crosspoint setting could be simply to send the “connect (03)” command to both controllers, only the active controller will be able to act on it, setting the crosspoint and replying with a connected.

3. A “connected (04)” message will automatically be received on the TCP connection to the active controller whenever there is a crosspoint change.