



**Snell
Advanced
Media**

Title	Router Control Protocols <i>SW-P-02 General Switcher Protocol – main commands</i> <i>SW-P-08 General Remote Protocol – main commands</i>
Document No	SW-P-88
Issue No	4b
Date	5 July 2016
Approved	_____

Copyright and Confidentiality

All rights reserved. All intellectual property rights including copyright subsisting in this publication are owned and/or controlled by Snell Advanced Media Limited. This publication and any information it contains is intended solely for use by the purchasers of Snell Advanced Media Limited equipment to which this publication pertains. It is to be treated as confidential and shall not be used for any other purpose. The publication shall not be copied or disclosed to third parties in whole or in part by any means without the prior written consent of Snell Advanced Media Limited.

Whilst every precaution has been taken in the preparation of this publication, Snell Advanced Media Limited accepts no responsibility for the accuracy of the information contained herein. Neither is any liability assumed for damages resulting from the information contained herein. The contents of this publication may change regularly to meet Snell Advanced Media Limited's business needs and may not maintain backwards compatibility with previous releases of the document. To this end Snell Advanced Media Limited reserves the right to change or supersede the content hereof at any time without prior written notice or consultation.

CONTENTS

1. INTRODUCTION – SW-P-02 GENERAL SWITCHER COMMUNICATION PROTOCOL	6
2. ELECTRICAL	6
3. PROTOCOL	6
3.1. Message Format	6
3.2. Message Commands	7
3.2.1. INTERROGATE Message	7
3.2.2. CONNECT Message.....	7
3.2.3. TALLY Message.....	7
3.2.4. CONNECTED Message.....	8
3.2.5. CONNECT ON Go Message	8
3.2.6. GO Message.....	8
3.2.7. CONNECT ON GO ACKNOWLEDGE Message	9
3.2.8. GO DONE ACKNOWLEDGE Message.....	9
3.3. Notes on Implementation	9
3.3.1. Philosophy.....	9
3.3.2. Further Criteria	10
4. SW-P-02 over IP	11
4.1. System Overview.....	11
4.2. Restrictions and constraints	11
4.3. System Operation.....	12
4.4. Configuration	13
5. STANDARD' PRO-BEL/SNELL CONTROLLER IMPLEMENTATIONS	14
6. INTRODUCTION – SW-P-08 GENERAL REMOTE CONTROL PROTOCOL	16
7. PHYSICAL AND DATA LINK LAYERS	17
7.1. The Physical Layer	17
7.1.1. Electrical.....	17
7.1.2. Data.....	17
7.2. The Data Link Layer	17
7.2.1. Message Structure	17
7.2.2. Transmission Protocol	18
8. MESSAGE LAYER – GENERAL COMMANDS	20
8.1. General Commands Received	20
8.1.1. Crosspoint Interrogate Message	20
8.1.2. Crosspoint Connect Message.....	21
8.1.3. Crosspoint Tally Dump Request Message.....	21
8.1.4. Crosspoint Connect On Go Group Salvo Message	22
8.1.5. Crosspoint Go Group Salvo Message	22
8.1.6. Crosspoint Salvo Group Interrogate Message	23
8.2. General Commands Transmitted	24
8.2.1. CROSSPOINT TALLY Message.....	24
8.2.2. CROSSPOINT CONNECTED Message.....	24
8.2.3. CROSSPOINT TALLY DUMP (BYTE) Message	25
8.2.4. CROSSPOINT TALLY DUMP (WORD) Message	25
8.2.5. CROSSPOINT CONNECT ON GO GROUP SALVO ACKNOWLEDGE Message.....	26
8.2.6. CROSSPOINT GO DONE GROUP SALVO ACKNOWLEDGE Message.....	26
8.2.7. CROSSPOINT GROUP SALVO TALLY Message	27
9. APPLICATION EXAMPLES	28
10. APPENDIX 1 – SW-P-08 GENERAL REMOTE CONTROL COMMUNICATION PROTOCOL EXAMPLES	29
10.1. Request Protocol Implementation	30
10.2. Interrogate Command.....	31
10.3. Connect Command.....	32
11. APPENDIX 2 – SW-P-08 over IP	32
11.1. System Overview.....	32
11.2. Restrictions and constraints	32
11.3. System Operation	33
11.4. Configuration (2330 and 2463)	33
11.5. Supporting dual controllers over IP.....	33
11.5.1. Overview.....	33
11.5.2. An example of dual control	34
11.5.3. Notes on example	34

Change Record

Issue	Date	Change Details
1	20/03/01	First Issue
2	21/05/04	Updated SW-P-08 connect example
3	15/03/2011	Updated layout, and added Centra controller into the matrix
4	16/11/2012	Updated Application example tables for both General Switcher and General Remote protocols. Added references to Nucleus2 controller for both protocols. Added SW-P-02overIP section. Added implementation note to General Remote Protocol Transmission protocol section regarding speed of receiver device sending ACK/NAK responses back to sender device.
4a	03/10/2013	Not official release – AS update to add SW-P-08 over IP info
4b	05/07/2016	Company name and logo change (AS)

1. INTRODUCTION – SW-P-02 GENERAL SWITCHER COMMUNICATION PROTOCOL

This protocol provides a robust and asynchronous method of controlling routers and is the preferred method of communication. New message commands are added as required.

2. ELECTRICAL

Normal electrical parameters are:

RS485/RS422, 8 Bit DATA, 1 STOP Bit, EVEN Parity, 38.4K Baud,

But the protocol will operate with any common RS485/RS422 and RS232 data parameters.

3. PROTOCOL

3.1. Message Format

All messages conform to the following format:

SOM	COMMAND	Message	Checksum
-----	---------	---------	----------

Where:

SOM	-	Start of message = FF hex
COMMAND	-	Command byte (this also defined message length) All command bytes listed are in decimal
Message	-	May be zero or more bytes
Checksum	-	7 bit, 2's complement sum of command and message field (m.s.b. = 0)

3.2. Message Commands

All command numbers referred to in this document are in decimal.

The following commands are defined:

3.2.1. INTERROGATE Message

Request for tally information. A device responds to this message with a TALLY message 03.2.3

Command Byte: 01

Message		
Byte	Field Format	Notes
Byte 1	Multiplier Bit[7] bit [4-6] bit [3] bits[0-2]	0 Dest number DIV 128 Either (1 = bad source) or (1 = crosspoint updated disabled.) or not used. Source number DIV 128 (Not applicable to this command) (always 0 for this command).
Byte 2	Dest number	Destination number MOD 128

Note: When all bits of Byte 1 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 monitor rows.

3.2.2. CONNECT Message

This message requests a route to be made through the router. A device will make the route and respond with a CONNECTED message 03.2.4 which also performs an effective acknowledgement.

Command Byte: 02

Message		
Byte	Field Format	Notes
Byte 1	Multiplier	Multiplier as defined in 3.2.1
Byte 2	Dest number	Destination number MOD 128
Byte 3	Src number	Source number MOD 128

Note: A single CONNECT command will suffice to set two crosspoints in the TDM 4-Wire Router 2510 local controller. The 2510 local controller will confirm this by sending two CONNECTED messages.

3.2.3. TALLY Message

This message returns tally information in response to an interrogating device. It also performs an effective acknowledgement to the INTERROGATE 3.2.1 message.

Command Byte: 03

Message		
Byte	Field Format	Notes
Byte 1	Multiplier	Multiplier as defined in 3.2.1
Byte 2	Dest number	Destination number MOD 128
Byte 3	Src number	Source number MOD 128

Note: Source number 1023 is reserved to indicate destination out of range.

3.2.4. CONNECTED Message

This message is issued by a device after it has made a router through a router, usually in response to a CONNECT 3.2.2 request. It is issued on ALL ports of the interface device.

Command Byte: 04

Message		
Byte	Field Format	Notes
Byte 1	Multiplier	Multiplier as defined in 3.2.1
Byte 2	Dest number	Destination number MOD 128
Byte 3	Src number	Source number MOD 128

3.2.5. CONNECT ON Go Message

Used in Salvo switching situations. Routing information is held in the receiving device until activated by GO command 00. A device will respond with a CONNECT ON GO ACKNOWLEDGE message 03.2.7 to indicate that the routing information has been stored successfully.

Command Byte: 05

Message		
Byte	Field Format	Notes
Byte 1	Multiplier	Multiplier as defined in 3.2.1
Byte 2	Dest number	Destination number MOD 128
Byte 3	Src number	Source number MOD 128

3.2.6. GO Message

Triggers the receiving device to set all routes in/clear the previously received CONNECT ON GO 3.2.5 messages. A GO DONE acknowledge 3.2.8 message, will be issued to indicate that the command has been executed.

Note: No individual CONNECTED messages 3.2.4 are issued. It is the responsibility of the controlling/listening devices to poll for the information, i.e. INTERROGATE commands 3.2.1

Command Byte: 06

Message		
Byte	Field Format	Notes
Byte 1	00	Set previously received messages
	01	Clear previously received messages

3.2.7. CONNECT ON GO ACKNOWLEDGE Message

This message is issued by a device in response to a CONNECT ON GO request 3.2.5.

Command Byte: 12

Message		
Byte	Field Format	Notes
Byte 1	Multiplier	Multiplier as defined in 3.2.1 (Except bad source bit always 0)
Byte 2	Dest number	Destination number MOD 128
Byte 3	Src number	Source number MOD 128

3.2.8. GO DONE ACKNOWLEDGE Message

This message is issued by a device in response to a GO message 0. If crosspoints are set then the message is issued on all ports of the receiving device.

Command Byte: 13

Message		
Byte	Field Format	Notes
Byte 1	00	Crosspoints set
	01	Stored crosspoints cleared
	02	No crosspoints to set/clear

3.3. Notes on Implementation

3.3.1. Philosophy

The philosophy of this message transfer protocol is that a controller assumes that a command has been successful until proved otherwise through, for example, a timeout.

In Pro-Bel/Snell controllers, a switcher is continually INTERROGATED as a background function to ensure that tally information is up-to-date. In Systems 1 and 2 when a CONNECT message is issued, the control system assumes that the connection will be successful and updates its internal tables accordingly. After a CONNECT message is issued, TALLY messages are ignored for a preset time interval of approx. 100mS. This ensures that any outstanding TALLY messages in the controlled device does not overwrite the new CONNECT information in the controller tally table. In System 3 the internal tally table is not updated until the connected message is received.

The background INTERROGATE function performed by a controller also indicates data link failure by timeouts on the TALLY message in the following manner:

When an INTERROGATE message is issued, a timeout counter is initialised. If a TALLY message is not received before the counter reaches zero, then a data link failure is adjudged to have occurred.

Further Note, therefore, that the timeout period must be less than the INTERROGATE interval and that this implies a fundamental lower limit to the INTERROGATE poll time of

INTERROGATE send time

- peripheral service time
- TALLY + response time
- TALLY service

Theoretically this could be as fast as

$$1.25 + 1 + 1.25 + \frac{1}{2} = 4\text{mS.}$$

Practically this should be no greater than an average of 1 INTERROGATE every 20mS.

3.3.2. Further Criteria

Further criteria to be met by any implementation of this protocol are:

- i) A system must successfully service messages being received simultaneously on all ports.
- ii) A system must successfully service up to 20 consecutive CONNECT messages on any port, at any one time.
- iii) A system must successfully service an average of 1 CONNECT message every 10mS.

4. SW-P-02 OVER IP

This section details the implementation of SW-P-02 over IP using on the Nucleus (2450) controller, a Sirius / Halo fitted with fitted with a BECK SC12 / SC13 chip or any router controlled using a Babel Fish.

4.1. System Overview

A TCP/IP server and UDP server running on the controller will provide control and monitoring of a router over Ethernet. The controller is a Nucleus (2450), a SC12 / SC13 hosted in Sirius and Halo routers or a Babel Fish.

For routers that do not have an embedded SC12 / SC13, the Babel Fish can be used. The Babel Fish also hosts an SC12, and provides two RS232/422 serial ports and an Ethernet port. The Babel Fish can be mounted alongside a router, and interfaces to it using an RS232/422 serial connection. The Babel Fish can be connected to an Ethernet network via its Ethernet port.

The server will use the General Switcher Communication Protocol to control and monitor a router on behalf of remote clients.

4.2. 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 8 TCP clients can connect to the server at any one time.
- Up to a maximum of 20 CONNECT/EXTENDED CONNECT (commands 2 and 66) messages will be serviced on each client at any one time. For all other messages, only one message will be serviced on each client at any one time. In order for the 20 connect commands to be sent to the router without waiting for a response between messages, the connect commands must be packaged into a single TCP/IP message. The client can send a message with more than 20 connects, however they will be passed on to the router in blocks of no more than 20.
- It is possible to package multiple messages other than connect commands, however they will be sent individually to the controller. It is possible for the network layer to combine TCP/IP messages, therefore a client may receive a TCP/IP response with more than one SW-P-02 message in it, even if the original commands were sent in individual packets.
- Internally, the SC12 will implement a timeout of 50ms for response messages from the controller. It is recommended that clients implement a configurable timeout of 100ms or greater to cater for various network scenarios. 50ms is the default timeout used by the controller and should be sufficient when servicing individual messages from a single TCP/IP client. If multiple connect commands are packaged into a single message this timeout needs to be increased to 200ms on the controller and should be 250ms on the client. Similarly if more than one client is being serviced by the controller the timeout also needs to be extended at the client end, since the controller may not be able to send the command to the router immediately.
- The UDP server will broadcast on up to 15 multicast addresses, allowing for up to 15 categories of messages. Two categories are implemented initially. These are connected commands sent when there is a crosspoint change and status response 6 commands sent when there is a change in the status 6 data.

4.3. 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-02 messages to the server. No packaging of the messages is required. The server will forward messages to the router. 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. This mechanism provides simple router control and monitoring.

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.

UDP Multicast Server

There may be a requirement to notify clients of events as soon as they occur. This requirement can be satisfied through the use of UDP multicasting. Clients will be required to join a UDP multicast group. A second server on the SC12 would provide this functionality.

The UDP server will broadcast CONNECTED/EXTENDED CONNECTED messages (commands 4 and 68) to all clients in the multicast group.

The SC12 will continually poll the power status of the Babel Fish or router, and report power failure. This is further described in the following section.

Secondary Status Reporting

PSU Failure Detection.

The SC12 built into Halo/Sirius routers has the ability to detect the failure of the two PSUs within its local frame. To be able to report this to connected clients SW-P-02 – Status Response 6 (Command 17 – Router type 1) has been amended to include PSU status reporting. This means that the SC12 will respond to SW-P-02 Status Requests (command 07) with the appropriate SW-P-02 Status Response 5 commands. See SW-P-02 Issue 25 or above for command details.

The normal status response commands returned by the router will be converted to the SW-P-02 Status Response 5 commands.

Two new SW-P-02 commands, Status Request 2 (command 18) and Status Response 6 (command 19) have been added. These commands can be used to determine the general state of a router or the Babel Fish, but the existing commands can be used to determine the state of the hardware in more detail. Status Response 6 will be broadcast through the UDP multicast server.

PSU failures of slave router frames cannot be reported through this mechanism.

If the Babel fish is used, the PSU failure status reported for type 1 routers is for the PSUs powering the Babel Fish.

PSU and fan failures are currently not reported for a Nucleus controller.

4.4. Configuration

Editing the CHIP.INI file located on the SC12 or in the case of the Nucleus (2450) controller the Config.ini file on the compact flash card, can change a number of configuration parameters. If an entry is not present in the file the default is used. The following define the INI settings and their defaults.

All the configuration strings are located in the section "[02overIP]".

The CHIP.INI file is edited using the SC12 configuration program (NA-00678). The Config.ini file on the compact flash using a text editor.

INI_ENABLE=true (Nucleus only)

This enables 02 over IP control. The default is true.

TCP_IP_PORT=2000

This is the port used for the TCP/IP server connection. The default is 2000.

UDP_CONNECT_PORT=2001

This is the port used for UDP broadcast message for connecteds. The default is 2001.

UDP_STAT6_PORT=2002

This is the port used for UDP broadcast messages for status 6 changes. The default is 2002.

CONNECT_MC_ADDR=224.1.1.1

Multi-cast address used for connecteds. The default is "224.1.1.1".

STAT6_MC_ADDR=224.1.1.2

Multi-cast address used for status 6 changes. The default is "224.1.1.2".

TCP_IP_LINK_TIME=60000 / 60

Time (in milli-seconds on a SC12 / SC13, seconds on the 2450) that the link remains open after the last message is received. A value of 0 causes the link to remain open indefinitely. The default is 60000 / 60 (1 minute).

COMPORT=COM1 or COM2 (SC12 / SC13 only)

This specifies the com port used for the connection to the router. The default used depends on whether the SC12 is located in a Babel Fish or directly in the router. If the SC12 is in a Babel Fish COM1 is the default, COM2 can be used by adding "COMPORT=COM2" to the INI file. If the SC12 is embedded in the router COM2 is the default since this is the port wired to the router control port. This would not normally be overridden, but it is possible by added "COMPORT=COM1" to the INI file.

BAUDRATE=38400 (SC12 / SC13 only)

This is the baud rate used on the serial link to the router. The default is 38400.

POLLTIME=1000

This is the frequency the SC12 polls the router for status, it gives the time in milli-seconds between each poll. The default is every 1000 mill-seconds. Increasing this value reduces the load on the SC12 and the router by polling it less frequently, it however increases the time before an error is reported. It may be necessary to do this on a heavily loaded SC12 or router. Reducing this value reduces the time before an error is reported but increases the

load on the SC12 and the router. The value should not be set lower than the TIMEOUT value.

TIMEOUT=50

This is the timeout in milli-seconds for a response from the router. The default is 50 milli-seconds. This should be sufficient for most routers when single commands are sent in a TCP/IP packet. If multiple connect messages are sent in a single TCP/IP packet the timeout should be increased to 200 milli-seconds. Also if the router is heavily loaded this value may need to be increased. The number of TCP/IP connections to the SC12 should not effect the timeout, since this is the timeout on the serial port to the router. The clients connecting to the SC12 may need to adjust their timeout for the response on the network due to messages on multiple TCP/IP connections being serialised to the router.

PSUMASK=FF

This is used to mask local PSU faults. The default is FF (allow all faults). The Babel fish can report two PSU faults corresponding to bits 0 and 1. Setting bit 0 or 1 to 0 will mask the corresponding PSU fault. This can be used when the Babel fish only has one PSU to prevent a fault being shown in the status response 5 and 6 messages. When in a router up to four PSU faults can be reported, using bits 0 to 3. If the SC12 is fitted to a router that does not have four PSU's, it would be desirable to mask faults on those PSU's that are not fitted by setting the corresponding bits to 0. Bits 3 to 7 are not used in a Babel fish and bits 5 to 7 are not used in a router.

FANMASK=FF

This is used to mask local Fan faults in the same way as the PSUMASK masked PSU faults. The default is FF (allow all faults). There are no local fans in a Babel fish, consequently this is not used. In a router up to five local fan faults can be reported using bits 0 to 4. Setting the appropriate bits to 0 in this mask will mask those corresponding fan faults.

RETURN_STATUS_5=YES

This determines whether status request 1 messages return the status response 5 message, or whatever response is returned from the router. The default is YES. Setting this to NO means that status response 1 to 4 are returned depending on what router the SC12 is connected to (See SW-P-02 for which routers return which responses). This is intended for use with existing equipment that does not support the status response 5 message.

USE_IPADDR_SWITCH=0 (SC12 / SC13 only)

This is in the IP section of the chip.ini file

This determines whether the last part of the IP address should be configured using the hex switch(es) on the back of a Sirius router. There are two hex switches on a Sirius Gold router, which allows the last byte to be selected in the range 00 to FF. Other Sirius routers only have a single hex switch, so only the last 4 bits of the last byte can be selected with this switch. The top 4 bits are taken from the configured IP address. Note – This function is only used if DHCP is not enabled.

A value of 1 enables this function.

5. STANDARD' PRO-BEL/SNELL CONTROLLER IMPLEMENTATIONS

The following table defines which command sets have been implemented on the various 'Standard' Pro-Bel/Snell Controller configurations.

COMMAND DESCRIPTION	NO	REF			STANDARD CONTROLLER (See Key for Details)
---------------------	----	-----	--	--	---

			01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23
INTERROGATE	01	3.2.1	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX
CONNECT	02	3.2.2	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX
TALLY	03	3.2.3	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX
CONNECTED	04	3.2.4	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX	XX
CONNECT ON GO	05	3.2.5								XX							XX								
GO	06	0								XX							XX								
CONNECT ON GO ACKNOWLEDGE	12	3.2.7								XX							XX								
GO ACKNOWLEDGE	13	3.2.8								XX							XX								

XX = Command Supported

Key: STANDARD CONTROLLERS						
	NO	CONTROLLER	NO	CONTROLLER	NO	CONTROLLER
	01	SYSTEM1 (7900)	02	SYSTEM2 (2411 & 8070)	03	SYSTEM3 (8180/8300/8400)
	04	PRODIGY (2640)	05	TS16 (16x4 2317)	06	2513/4/5 MASTER CARD
	07	2412 HD LOCAL CONTROLLER	08	2413/4 HD LOCAL CONTROLLER	09	2553/8 SERIAL INTERFACE
	10	5007/5107 TDM CONTROLLER	11	2510 4 WIRE ROUTER CONTROLLER	12	TM16/24/32 (2659)
	13	2136 SERIAL CONTROLLER	14	6170 PROTOCOL TRANSLATOR	15	XD LOCAL CONTROLLER (2419)
	16	FREEWAY/AXIS/HALO/SIRIUS/NEBULA (2440/2441/2442/2443/2444/2445)	17	SYSTEM3 (6000)	18	SYSTEM3 (6001)
	19	ECLIPSE (2400)	20	ECLIPSE (2404)	21	AURORA CONTROLLER (6006)
	22	Nucleus (2450)	23	Centra/Nucleus2 Controllers (2330/246x)		

6. INTRODUCTION – SW-P-08 GENERAL REMOTE CONTROL PROTOCOL

This protocol document contains all the general router switch commands commonly used by external controllers.

This protocol has been developed to provide a common and robust method of interfacing to Pro-Bel 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 and 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.

7. PHYSICAL AND DATA LINK LAYERS

7.1. The Physical Layer

7.1.1. 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.

7.1.2. 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 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.

7.2. 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.

7.2.1. 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.

7.2.2. 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.

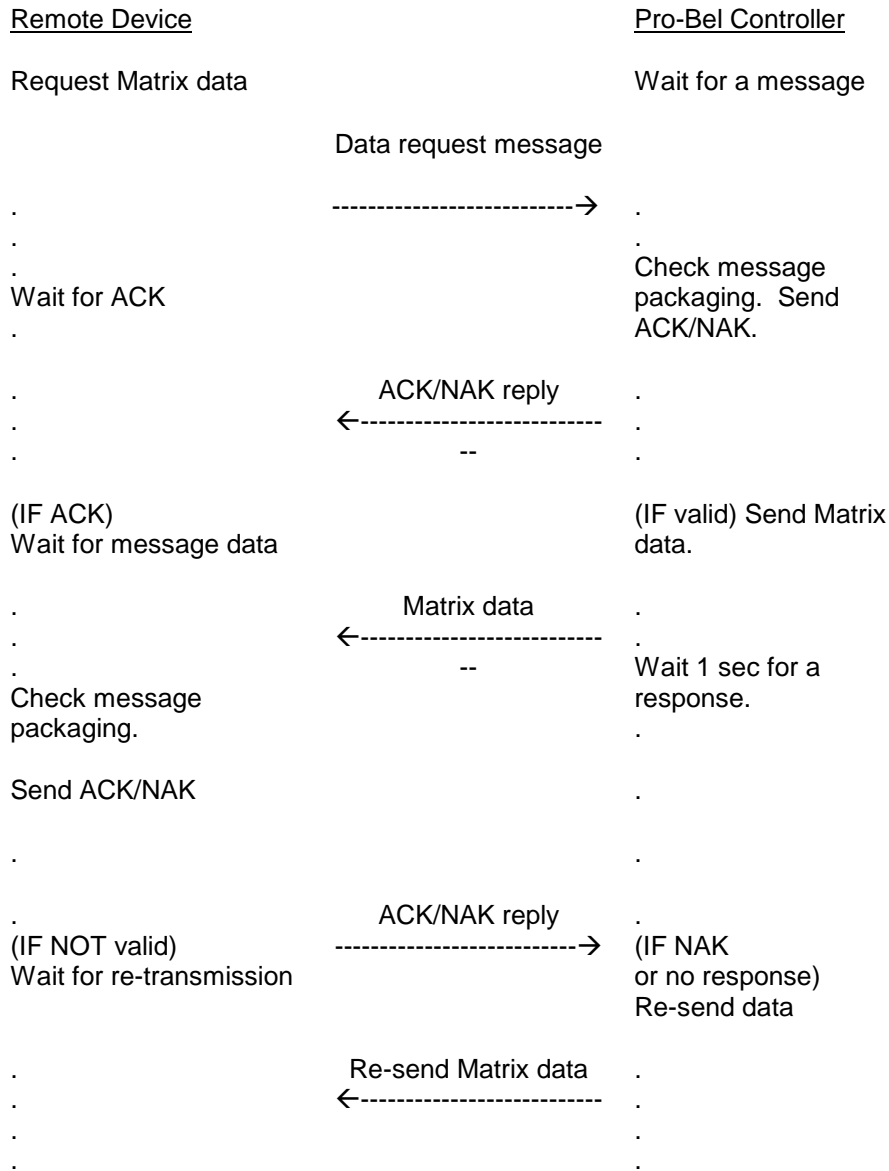
If the Pro-Bel controller receives a NAK, or does not receive a response within 1 second, it will re-send the data reply to the remote device. The re-send operation will be repeated a maximum of 5 times in the absence of ACK responses.

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:



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

8. 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:-

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

$x \text{ MOD } y =$ the value (less than unity) which is left when 'y' is multiplied by z [see above] and subtracted from 'x'.

$\text{INT } x =$ 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.

8.1. General Commands Received

8.1.1. 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 Bits[4-7] Bits[0-3]	Matrix/Level number Matrix number Level number
Byte 2	Multiplier Bit[7] Bits[4-6] Bit[3] Bits[0-2]	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 0 Dest number DIV 128 TDM and Digital Video source "bad" status (0 = good src) Source number DIV 128
Byte 3		Destination number MOD 128

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

8.1.2. 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 8.1.1
Byte 2	multiplier	Multiplier as defined in 8.1.1.
Byte 3	Destination	Destination number MOD 128.
Byte 4	Source	Source number MOD 128

8.1.3. 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 8.1.1

8.1.4. 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 8.1.1
Byte 2	multiplier	Multiplier as defined in 8.1.1.
Byte 3	Destination	Destination number MOD 128.
Byte 4	Source	Source number MOD 128
Byte 5	Salvo number	Holds the Salvo group number to configure (Destination and source will always overwrite previous data).
	Bit[7] Bits[0-6]	bit 7 0 Salvo number 0-127

8.1.5. 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 8.1.4.

8.1.6. 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 8.1.4.
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..

8.2. General Commands Transmitted

8.2.1. 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 8.1.1.
Byte 2	Multiplier	Multiplier as defined in 8.1.1.
Byte 3	Destination	Destination number MOD 128
Byte 4	Source	Source number MOD 128

8.2.2. 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 8.1.1.
Byte 2	Multiplier	Multiplier as defined in 8.1.1.
Byte 3	Destination	Destination number MOD 128
Byte 4	Source	Source number MOD 128

8.2.3. 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 8.1.1.
Byte 2	Num of Tallies	Number of tallies returned.
Byte 3	1 st Destination	First destination number
Byte 4	1 st Source	First source number
Byte 5	2 nd Source	Second source number

8.2.4. 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 8.1.1.
Byte 2	Num of Tallies	Number of tallies returned (Maximum 64)
Byte 3	1 st Destination	First destination number DIV 256
Byte 4	1 st Destination	First destination number MOD 256
Byte 5	1 st Source	First source number DIV 256
Byte 6	1 st Source	First source number MOD 256
Byte 7	2 nd Source	Second source number DIV 256
Byte 8	2 nd Source	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.

8.2.5. 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 8.1.1.
Byte 2	Multiplier	Multiplier as defined in 8.1.1. (Except bad source bit always 0).
Byte 3	Destination	Destination number MOD 128
Byte 4	Source	Source number MOD 128
Byte 5	Salvo group	Salvo group number as defined in 8.1.4

8.2.6. 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	00	Crosspoints set
	01	Stored crosspoints cleared
	02	No crosspoints to set / clear
Byte 2	Salvo group	Salvo group number as defined in 8.1.4

8.2.7. 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 8.1.1.
Byte 2	Multiplier	Multiplier as defined in 8.1.1. (Except bad source bit always 0).
Byte 3	Destination	Destination number MOD 128
Byte 4	Source	Source number MOD 128
Byte 5	Salvo group	Salvo group number as defined in 8.1.4
Byte 6	Connect index	Connect index as defined in 8.1.6
Byte 7	Validity flag	00 Valid connect index returned, more data available 01 Valid connect index returned, last in queue 02 Invalid connect (no data in SALVO)

9. APPLICATION EXAMPLES

This section details the command sets used within the four main Pro-Bel system controllers, namely SYSTEM2, SYSTEM3, AURORA and FREEWAY for basic router commands.

INPUT COMMANDS										OUTPUT COMMANDS									
COMMAND			SYSTEM							COMMAND			SYSTEM						
NO	NAME	REF	01	02	03	04	05	06	07	NO	NAME	REF	01	02	03	04	05	06	07
01	CROSSPOINT INTERROGATE	8.1.1	XX	XX	XX	XX	XX	XX	XX	03	CROSSPOINT TALLY	8.2.1	XX	XX	XX	XX	XX	XX	XX
02	CROSSPOINT CONNECT	8.1.2	XX	XX	XX	XX	XX	XX	XX	04	CROSSPOINT CONNECTED	8.2.2	XX	XX	XX	XX	XX	XX	XX
21	CROSSPOINT TALLY DUMP REQUEST	8.1.3	XX	XX	XX	XX	XX	XX	XX	22	CROSSPOINT TALLY DUMP (byte) RESPONSE	8.2.3	XX	XX	XX	XX	XX	XX	XX
										23	CROSSPOINT TALLY DUMP (word) RESPONSE	8.2.4	XX	XX	XX	XX	XX	XX	XX
120	CROSSPOINT CONNECT ON GO GROUP SALVO	8.1.4			XX	XX	XX			122	CROSSPOINT CONNECT ON GO GROUP SALVO ACKNOWLEDGE	8.2.5			XX	XX	XX		
121	CROSSPOINT GO GROUP SALVO	8.1.5			XX	XX	XX			123	CROSSPOINT GO GROUP SALVO ACKNOWLEDGE	8.2.6			XX	XX	XX		
124	CROSSPOINT GROUP SALVO INTERROGATE	8.1.6			XX	XX	XX			125	CROSSPOINT GROUP SALVO TALLY	8.2.7			XX	XX	XX		

KEY:	SYSTEM:
01	SYSTEM2
02	SYSTEM3 (8400 VARINATS)
03	SYSTEM3 (6000 VARIANTS)
04	FREEWAY/AXIS/HALO/SIRIUS/NEBULA
05	AURORA/SYSTEM3 (6006/6001 VARIANTS)
06	Nucleus Controller (2450)
07	Centra/Nucleus2 Controllers (2330/246x)

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.

10. APPENDIX 1 – SW-P-08 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 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.

10.1. 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.

10.2. 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.

10.3. Connect Command

This example is a Connect message to connect Matrix 2, Level 2, Destination 3 to Source 1 (Comments are using from base 1).

(**note** all protocol data is based from zero)

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

11. APPENDIX 2 – 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 sent 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.