Quartz

Routing Switcher
Remote Control Protocol
type 1

Quartz Electronics Ltd
59 Suttons Business Park
Reading
RG6 1AZ
ENGLAND

Telephone +44 (0) 118 935 0200
Facsimile +44 (0) 118 935 0202
Web Page http://www.quartzuk.com
Support +44 (0) 118 935 0203
E-mail support@quartzuk.com
Emergency Technical Support +44 (0) 771 410 2629

Contents

Command 4.1 and 4.4, and Responses 5.2, 5.3, and 5.4 are the minimum set that needs to be implemented for router control..

1. Introduction

This document specifies a protocol suitable for simple control of a Quartz routing system by a computer or third party system. The protocol can use any of the physical interfaces available on the system, either a RS232/422 interface or Ethernet.

As the protocol is ASCII text based, remote changes can be made using a terminal or a computer running terminal emulation software. For most applications only the .S command will be required to set crosspoints, with the .I or .L command to interrogate crosspoints. The .#01 command may be used at system start up to check the RS232/422 link.

2. Serial (RS232 / 422) Physical Interface

Most Quartz products have a built in serial port that is link selectable between RS422 and RS232 mode. For older routing products see the section at the end of this document.

2.1 Quartz D9 Serial Pin-Out

The full router pin-out is shown below

RS232	
Pin	Signal
1	CHASSIS
2	RTS
3	RXD
4	N/C
5	N/C
6	GND
7	TXD
8	CTS
9	N/C

RS422	
Pin	Signal
1	CHASSIS
2	TX-
3	RX +
4	RX GND
5	N/C
6	TX GND
7	TX +
8	RX-
9	N/C

For information, the following pin-out is normally used on personal computers, but check your own computer's documentation.

D9 PLUG	
Pin	Signal
1	DCD
2	RXD
3	TXD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	RI

D25 PLUG	
Pin	Signal
2	TXD
3	RXD
4	RTS
5	CTS
6	DSR
7	GND
8	DCD
20	DTR
22	RI

2.2 PC-to-Quartz RS232 Interface Cable

The cable between a PC with a D9 connector and the router:

PC	D9 Skt	Cable	D9 Plug	Router
D9 Plug	2 (RXD)	----------	7 (TXD)	D9 Skt
	3 (TXD)	----------	3 (RXD)	
	5 (0V)	----------	6 (0V)	
	5 (GND)			

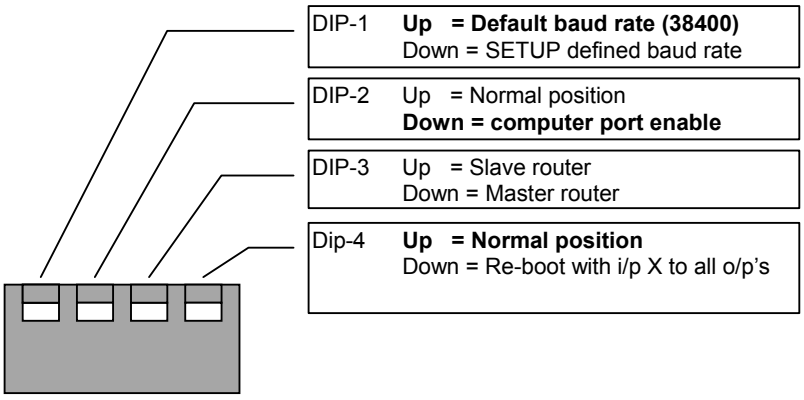
The cable between a PC with a D25 connector and the router:

PC	D25 Skt	Cable	D9 Plug	Router
D25 Plug	3 (RXD)	----------	7 (TXD)	D9 Skt
	2 (TXD)	----------	3 (RXD)	
	7 (0V)	----------	6 (0V)	
	5 (GND)			

2.3 Embedded Control System (FU-0003) Interface

2.3.1. Router DIP Switch Settings

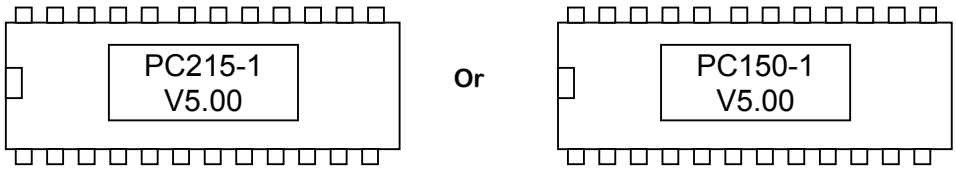
DIP switch 2 (DIP-2) must be down to enable the computer port. The port will then only respond to messages of the correct format as detailed in this protocol document.



The normal setting for a master router is Up,Down,Down,Up and for a slave router Up,Down,Up,Up.

2.3.2. Protocols

The embedded router/panel firmware EPROM has one remote control protocol built in to it. The Quartz standard ASCII protocol is indicated by a -1 at the end of the EPROM number i.e. PC215-1.



In later systems the embedded router/panel firmware FLASH device may have one or more remote control protocol built in to it. These protocols can be selected via the WinSetup configuration editor.

The port operates at 38400, 8 data bits, no parity, and 1 stop bit. Other baud rates and parity options can be selected from WinSetup.

3. Ethernet Physical Interface

More recent Quartz products such as the SC-1000, Q256, Xenon and Topaz are equipped with an Ethernet port. These support 10base2 Ethernet using CAT5 cable. The network should ideally be closed (or at least isolated with a suitable switch) for use with devices on the Quartz routing system.

Both the physical interfaces (Serial and Ethernet) use the same protocol command and reply structure, which is detailed below. The following few paragraphs just describe the Ethernet configuration and connection needed in order to communicate with these devices.

Each device on the network must be assigned a unique IP address. Changing and interrogating the IP addresses are usually sent via serial port commands as documented in 4.14. If two controllers are installed in the controller or routing frame two stream socket connections will be required to the main and backup controllers, as either of the two might be in control of the routing system at any point in time.

Normally Quartz controllers will allow a TCP/IP connection via the telnet port number (23).

On the SC-1000 additional TCP/IP ports can be added to the configuration file currently selected by the SC-1000. This is configured from WinSetup under the “System”->”SC-1000 Configuration”->”Options” dialog. A TCP/IP port should be added using the Quartz protocol and at this point the user can specify a port number for the conversation (please note this port number should be greater than 1024). Please make sure the server check box is also ticked.

Quartz controllers use standard TCP/IP stream sockets (sometimes referred to as Berkeley sockets) to communicate with other network devices needing to control the routing system. Stream sockets are connection oriented, and so a connection must be opened and maintained for the duration of communications. Stream sockets are supported for many different host environments and operating systems.

In order to start communication with a controller the computer or other host device must originate the communications. In other words the router controller will be the server and the computer will be the client.

The client should :

1. Create a stream socket.
2. Connect the socket to the IP address of the desired main/reserve controller on the port specified in the configuration. (Note If there are redundant controllers, each controller will have its own unique IP address and need a separate stream socket. However, only the active controller will be able to accept a connection).
3. Once the connection has been successful, commands can then be sent to the routing controller and the controller will reply according to the command sent as detailed below in section 4.
4. Once the connection is finished with, it should be closed as normal.
5. The routing controller will terminate all connections on a download of a new configuration so the clients program should cope with losing and re-establishing this connection to the controller.

The SC-1000 controller allows the user to see a list of the currently active TCP/IP connections via both the configuration port and the LCD window on the front of the unit.

4. Commands

All characters used should be in upper case. Values in { } brackets are variable fields. Values in [ ] brackets are optional variable fields. Values in () brackets are non-printable characters i.e carriage return (cr). No space or tab characters are used. All crosspoint numbering starts from one (1) and not zero (0). The (cr) character is ASCII carriage return (code 0D hex). A reply is only generated after a (cr).

Some examples are given of 'C' code suitable to output messages. These are only intended to clarify the command structure. They assume a separate 'C' routine RS232_printf which is similar to the standard printf but with is output directed to the RS232 serial port of the host computer.

4.1 Set Xpt Message

This allows a route to be made (crosspoint set) and uses the command format

.S{level}{dest},{srce}(cr)

The legal levels are V,A,B,C,D,E,F,G in 8 level systems and V,A,B,C,D,E,F,G,H,I,J,K,L,M,N,O in 16 level systems.

TYPICAL MESSAGE	Q1600 LEVEL	TYPICAL USE
.SV1,2(cr)	level 1	video
.SA2,4(cr)	level 2	audio 1
.SB4,1(cr)	level 3	audio 2
.SC31,12(cr)	level 4	control/time code

The router will reply with a .U{level}{dest},{srce}(cr) message. The error message .E(cr) is generated if the command is not recognised.

SEND	REPLY	
.SV001,002(cr)	.UV001,002(cr)	
.SC9,3(cr)	.UC009,003(cr)	
.SZ1,1(cr)	.E(cr)	/* unknown level */
.MA1,1(cr)	.E(cr)	/* unknown command */

Multiple levels can be set by including more level identifiers in the message

.SVA1,2(cr)	level 1,2
.SAC1,2(cr)	level 2,4
.SCV1,2(cr)	level 4,1
.SVABC1,2(cr)	level 1,2,3,4

The router will reply with an update message (see section 5.4)

.SBA1,2(cr)	.UAB001,002(cr)

Example 'C' code to set a crosspoint on levels 1, 2, & 3.

```
level = "VAB";                  /* levels to be controlled */
dest = 5;                       /* matrix output */
srce = 2;                       /* matrix input */
RS232_printf(".S%s%d,%d\r", level, dest, srce);
```

4.1.1 Multiple Set Xpt Message

This command allows a block of up to 16 routes to be set from one command providing that the command does not exceed 256 bytes in length. The command allows the level to be optionally specified for any of the routes, if the level is not specified for the first destination then it assumes all levels for that destination will be routed to the source given, if a level is not specified for a subsequent destination then it assumes the last level defined for a destination. The format of the command is given below :

.M{destination},{source},{destination},{source} …. (cr)

where {destination} = [level$_1$]{dest$_1$}[+{destination}][-{dest$_3$}[+{destination}]]
where {source} = {srce$_1$}[+{source}][-{srce$_3$}[+{source}]]

The ',' character is used to delimit a destination or a source. The '+' and '-' character can be used to specify a number of parameters of the same type (destinations/ sources). The '+' character indicates that another parameter of the same type will be added. The '-' character indicates that the next parameter of the same type defines the end of a range.

The simplest form of the command is given below.

.M[level$_1$]{dest$_1$},{srce$_1$},[level$_2$]{dest$_2$},{srce$_2$},…..[level$_n$]{dest$_n$},{srce$_n$}(cr)

The command

.M[level$_1$]{dest$_1$}+[level$_2$]{dest$_2$}+[level$_3$]{dest$_3$},{srce$_1$}(cr)

would set all three destinations dest$_{1,2,3}$ to the specified source.

The command

.M[level$_1$]{dest$_1$}-{dest$_3$},{srce$_1$}(cr)

would set all logical destinations in the range dest$_1$ to dest$_3$ to the specified source on level$_1$. Note no optional level is permitted on the end of a destination range.

The command

.MVA001-005,010-014(cr)

would set destination 1 to source 10, destination 2 to source 11, destination 3 to source 12, destination 4 to source 13 and destination 5 to source 14 on control levels 1 and 2.

The router replies to this command with either an acknowledge or error message. The acknowledge command will indicate purely that the command has been received and is being processed. The routing system will then process the route takes within the command and issue update messages for each valid route request. Please note that this command was added on version V1.17 (15/11/01) of the RCP-T01 protocol.

4.2 System destination lock

This allows the specified destination to be locked within the system, It will inhibit the destination being changed on any level from any standard control panel in the system without turning this lock off. This command combines the functionality of locking, unlocking and interrogating the system destination locks.

To lock a destination use the command

.BL{dest}(cr)	Lock a destination

To unlock a destination use the command

.BU{dest}(cr)	Unlock a destination

To interrogate the systems destination lock status use the command

.BI{dest}(cr)	Interrogate destination lock status

The router replies to all of the above commands with

.BA{dest},{lock status}(cr)	Destinations lock status

where the {lock status} field is in ASCII and consists of up to three characters. The values of which should be interpreted as follows:

"0"	destination unlocked
"1" … "254"	protected destination lock, locked by panel at Q-link address 'n'-1
"255"	unprotected destination lock

4.3 Fire System Salvo

This allows the specified salvo to be 'fired' within the system.

To 'fire' a system salvo use the command

.F{salvo}(cr)	Fire a Salvo

where the {salvo} field is in ASCII and consists of up to three numeric characters, the value of which should be between 1 and 32.

The router replies to the command with

.A (cr)

4.4 Interrogate route

This allows a single destination to be interrogated to find the current source routed to it

.I{level}{dest}(cr)

The router replies with

.A{level}{dest},{srce}(cr)

Please note that destinations that have been routed via tielines modify the returned source value to indicate the control level the source has been routed from. If systems ignore the level information then only the lower 12 bits of the source value should be used. For more information please refer to section 4.1.

Example ‘C’ code to interrogate a crosspoint on levels 1, 2, & 3.

```
level = “V”;                    /* levels to be controlled */
dest = 5;                       /* matrix output */
RS232_printf(".l%s%d\r", level, dest);
```

4.5 Connect routes

This command is reserved for later use but will have the following format.

This allows up to 4 crosspoints to be set in a single command line.

.C{level}{dest},{srce} {level}{dest},{srce} .. (cr)

The lev,dest,srce part of the message can be repeated up to 4 times with a space between each group.

4.6 List routes

This command allows a block of up to 8 routes to be interrogated. The command has 2 formats. In both cases the command specifies a level and start destination. The router replies with a list of up to 8 routes. Less than 8 routes are returned if the maximum destination is exceeded, or the search conditions are not met.

Format (1) : List from the specified destination showing current sources.

.L{level}{dest},-(cr)

The router replies with

.A{level}{dest},{srce}{level}{dest},{srce} ... {level}{dest},{srce}(cr)

Example ‘C’ code to list routes (format 1).

```
level = “V”;                    /* levels to be controlled */
dest = 5;                       /* matrix output */
RS232_printf(".L%s%d,-\r", level, dest);
```

Format (2) : List from the specified destination only those destinations using the specified source.

.L{level}{dest},{srce}(cr)

The router replies with

.A{level}{dest},{srce}{level}{dest},{srce} ... {level}{dest},{srce}(cr)

Example ‘C’ code to list routes (format 2).

```
level = “V”;                    /* levels to be controlled */
dest = 5;                       /* matrix output */
srce = 2;                       /* matrix input */
RS232_printf(".L%s%d,%d\r", level, dest, srce);
```

4.7 Read name table

This allows the specified destination/source or level mnemonic to be read back from the system.

.RD{dest}(cr) Read a destination mnemonic

where the {dest} field is a ASCII representation of a number between 1 and the maximum destination.

.RS{source}(cr) Read a source mnemonic

where the {source} field is a ASCII representation of a number between 1 and the maximum source.

.RL{level}(cr) Read a level mnemonic

where the {level} field is one of the following ASCII characters: (V,A,B,C,D,E,F,G).

The router replies to the above commands with one of the following responses dependant on the firmware, a comma delimiter should be checked for to determine which response has been sent.

.RA[D/S/L]{mnemonic string}(cr)
.RA[D/S/L]{dest/source/level},{mnemonic string}(cr)

where the {mnemonic string} field is in ASCII and consists of eight characters. The string is delimited by whitespace.

The following commands allow the specified 10 character destination/source or level mnemonic to be read back from the system. These mnemonics appear on the LCD button range of panels and are displayed in a two rows of five character orientation.

.RE{dest}(cr) Read a destination mnemonic

where the {dest} field is a ASCII representation of a number between 1 and the maximum destination.

.RT{source}(cr) Read a source mnemonic

where the {source} field is a ASCII representation of a number between 1 and the maximum source.

.RM{level}(cr) Read a level mnemonic

where the {level} field is one of the following ASCII characters: (V,A,B,C,D,E,F,G).

The router replies to the above commands with one of the following responses dependant on the firmware, a comma delimiter should be checked for to determine which response has been sent.

.RA[E/T/M]{mnemonic string}(cr)
.RA[E/T/M]{dest/source/level},{mnemonic string}(cr)

where the {mnemonic string} field is in ASCII and consists of ten characters. The string is delimited by whitespace.

4.8 Write name table

The following commands allow the specified 8 character destination/source or level mnemonic to be changed in the system.

.WD{dest},{mnemonic string}(cr) Write a destination mnemonic

.WS{source},{mnemonic string}(cr) Write a source mnemonic

.WL{level},{mnemonic string}(cr) Write a level mnemonic

where the {dest}/{source} field is a ASCII representation of a number between 1 and the maximum destination/source.

where the {level} field is one of the following ASCII characters: (V,A,B,C,D,E,F,G).

where the {mnemonic string} field is in ASCII and consists of eight characters. The string is delimited by whitespace.

The router replies to the above commands with an update to the router name in the form shown below and then an acknowledge.

The router replies to the above commands with

.RA[D/S/L]{dest/source/level},{mnemonic string}(cr)

followed by an acknowledge response.

The following commands allow the specified 10 character destination/source or level mnemonic to be changed in the system. These mnemonics appear on the LCD button range of panels and are displayed in a two rows of five character orientation.

.WE{dest},{mnemonic string}(cr) Write a destination mnemonic

.WT{source},{mnemonic string}(cr) Write a source mnemonic

.WM{level},{mnemonic string}(cr) Write a level mnemonic

where the {dest}/{source} field is a ASCII representation of a number between 1 and the maximum destination/source.

where the {level} field is one of the following ASCII characters: (V,A,B,C,D,E,F,G).

where the {mnemonic string} field is in ASCII and consists of ten characters. The string is delimited by whitespace.

The router replies to the above commands with

.RA[E/T/M]{dest/source/level},{mnemonic string}(cr)

followed by an acknowledge response.

4.9 Read configuration (Embedded control system only)

This allows the routers internal configuration EPROM to be read back eight bytes at a time. The legal address range is 0 to 3FFFh.

.?{addr}(cr)

The router replies with

.A{addr},{byte 1},{byte 2}, ... ,{byte 8}(cr)

4.10 Write configuration (Embedded control system only)

This allows the routers internal configuration EPROM to be re-written. For this option to work the router must be fitted with EEPROM or NVRAM in place of the standard EPROM. The legal address range is 0 to FFFFh in normal mode and 0 to 1EFFFh in extended addressing mode. Between one and eight bytes can be written on each command.

.!{addr},{byte 1},{byte 2}, ... ,{byte 8}(cr)

The router replies with

.A{addr},{byte 1},{byte 2}, ... ,{byte 8}(cr)

The addr parameter is 4 bytes long in normal mode and five bytes long in extended addressing mode.

4.11 Queue Command (SC-1000 controller only)

This command allows the user to dynamically create, destroy, modify and fire multiple salvos during runtime.

It should be noted that whilst a salvo buffer is selected, all set xpt messages received will go into that buffer and not made until that salvo is fired. Any set xpt commands that are directed to a salvo will be acknowledged with the standard acknowledge response (.A(cr)) and not an update take message, as the update take message will be generated when the salvo is fired. Once the salvo has been deselected, then takes will be made instantly and acknowledged normally. Salvo numbers are counted from 1, with 0 having a special function, which changes depending on the command sent.

Creating and Selecting the Salvos

.QC{n}(cr) - Changes salvo to salvo n, creating the buffer it does not exist. If n is zero, then the currently selected salvo will be deselected. If n is not specified, the router will return the currently selected salvo

If the command succeeds, the router replies with:

.QC{n}(cr)

Emptying the a Salvo

.QR{n}(cr) - Resets the content of salvo n. If n is zero, then all (if any) salvos are emptied. If n is not specified, the currently selected salvo will be emptied (if one is selected).

If the salvo selected exists, or a reset all is sent, the router will respond with:

.A(cr)

Fire a Salvo

QS{n}(cr) - Fires salvo n. If n is not specified, then the currently selected salvo (if one is selected) will be fired.

If the salvo exists, the router will respond with:

.A(cr)

Destroy a Salvo

.QD{n}(cr) - Destroys salvo n. If n is zero, then all (if any) salvos are destroyed.

If n is not specified, the currently selected salvo will be destroyed (if one is selected). Note, that this automatically deselects the current salvo

If the salvo selected exists, or a destroy all is sent, the router will respond with:

.A(cr)

List the Number of Items in a Salvo

.QL{n}(cr) - Lists the number of items in salvo n, if n is zero, then all (if any) salvos are listed. If n is not specified, the number of items in the current salvo will be listed (if one is selected).

If the salvo selected exists, or a destroy all is sent, the router will respond with:

.QL{n},{m}(cr) - Where n is the current salvo and m is the number of items in it. This will be repeated for each salvo that exists, if list all is selected.

Example

Action	Command	Response
Create salvo 1	.QC1(cr)	.QC1(cr)
Send take to salvo 1	.SV1,1(cr)	.A(cr)
Send take to salvo 1	.SV3,2(cr)	.A(cr)
Send current salvo	.QS(cr)	.A(cr)
		.UV1,1(cr)
		.UV3,2(cr)
Empty current salvo	.QR(cr)	.A(cr)
Deselect current salvo, return to normal take mode	.QC0(cr)	.QC0(cr)

4.12 General Engineering commands

All the values in brackets {xx} are two digit hexadecimal numbers.

.#00(cr)	Enquire the devices Q-Link Protocol version and address. The router replies with (.)(A){ver},{xx}(cr). The xx parameter is the hexadecimal Q-link address of the device and the ver parameter is the Q-link version number. 106 is equivalent to V1.06.
.#01(cr)	Test if router is connected. The router replies with (.)(A)(cr).
.#02(cr)	Put router in extended addressing mode. The router replies with (.)(A)(cr).
.#12,{xx}(cr)	Reset router. This can be used after re-configuration. The xx parameter is optional and causes all routes to be set to source xx.
.#23(cr)	Halt vertical interval switching. This allows complex salvos to be set up to occur in one vertical interval
.#34(cr)	Restart vertical interval switching. As above.
.#40,{ddd}(cr)	On-line update to source number ddd. Master replies with an Ack.
.#41,{ddd}(cr)	On-line update to destination number ddd. Master replies with an Ack.
.#42,{ddd}(cr)	On-line update to source mnemonic number ddd. Master replies with an Ack.
.#43,{ddd}(cr)	On-line update to destination mnemonic number ddd. Master replies with an Ack.
.#44,{xx}(cr)	Force device xx offline. Master replies with an Ack.
.#45,{xx}(cr)	Return Q-link status of device xx. .Axx,01 = on-line .Axx,02 = off-line .Axx,03 = this unit
.#46,{xx}(cr)	Get general error number for address
.#47,{xx},{nn}(cr)	Get specific error number occurrences for device
.#48,{xx}(cr)	Clears error count for device.
.#49,{xx}(cr)	Get general report number for address (can then get details)
.#50,{xx}(cr)	Get type and version numbers of Q-link device from master – uses stored information retrieved from polling (or slave, but only own data).
.#51,{xx}(cr)	Force master to enquire type and version numbers of Q-link device (uses fetched data).
.#56(cr)	Reserved (halt panel updates).
.#67(cr)	Reserved (restart panel updates).
.#78(cr)	Reserved (enable extended protocol).

.#80,75(cr)	Jump to boot loader.
.#81(cr)	Boot loader command, DO NOT USE .

4.13 Video Status Display

These commands only work when the status display is operating in stand alone mode.

.VP-(cr)	Display page decrement.
.VP+(cr)	Display page increment.
.VP{xx}(cr)	Display page xx (01, 02, etc)

4.14 Engineering commands

All the values in brackets {xx} are two digit hexadecimal numbers.

.&LOCALTCPIP,192.0.2.100(cr)	Set TCP/IP address of the controller you are connected to.
.&REMOTETCPIP,192.0.2.101(cr)	Set TCP/IP address of the other controller in a dual redundant configuration.
.&TCPGATE,192.0.2.52(cr)	Sets TCP/IP address of the network gateway .
.&TCPMASK,192.0.2.52(cr)	Sets TCP/IP address mask.

5. Responses

5.1 Acknowledge

Acknowledge is used as a no error response to some messages

.A(cr)

5.2 Errors

If an error is detected in any sequence prior to a (cr), the matrix replies with

.E(cr)

5.3 Power up

At power up or reset the matrix outputs a (cr).P(cr) to inform a remote computer that the matrix is now on-line.

5.4 Update

If any routes are changed in the matrix, either by remote panels or the remote control protocol, the matrix replies with an update message for each route changed. For a normal take, where the destination and sources are routed on the same control level updates are of the form:

.U{levels}{dest},{srce}(cr)

The response message always replies with levels in the following order : V,A,B,C,D,E,F,G…

5.4.1 Tieline Update

Since a tie-line take has a different level for the destination as for the source then the level information is included as;

.U{dest level}{dest},{ {srce level} ORed with {srce} }

In this circumstance there will be only one level defined for the source and destination. The source value will contain both the source number and the level from which it has been routed to.

When the source number is represented in hexadecimal, the lower 12 bits of the source value define the source number (this gives the source number a range of 1-4095). The next 5 bits define the level value (between 1 and 31) of the tieline source.

For example the take request :

.SVA1,2(cr)

Could result in the following response :

.UV001,8194(cr)

This response means that destination 1 on control level 1 ('V') is routed via a tieline to source 2 on control level 2. As 8194 is 2002 in hexadecimal lower 12 bits is 002 or source 2, upper 5 bits is 2 or control level 2.

Below is a table that précis the responses from the commonly used routing commands

Command	Command Data	Normal Response/s	Note
Set Xpt	.S{level}{dest},{srce}(cr)	No explicit response	.U response conditional on route being made
Multiple Set Xpt	.M{destination},{source}, {destination},{source}…. (cr)	.A	.U response/s conditional on route being made
Interrogate Destination status	.I{level}{dest}(cr)	.A{level}{dest},{srce}(cr)	
List Routes	.L{level}{dest},-(cr)	.A{level}{dest},{srce}{level}{dest}, {srce} ... {level}{dest},{srce}(cr)	
Lock Destination	.BL{dest}(cr)	No explicit response	.BA response conditional on change being made
Unlock Destination	.BU{dest}(cr)	No explicit response	.BA response conditional on change being made
Interrogate Destination Lock	.BI{dest}(cr)	.BA{dest},{lock status}(cr)	
Fire Salvo	.F{salvo}(cr)	.A (cr)	.U response/s conditional on route being made

Note: Solicited and unsolicited destination route and lock update responses (.U and .BA) will only be sent by the router if the control system has made changes.

All commands might return a error response if the syntax is incorrect.

6. Changes by panels

If a panel changes a xpt, then the computer port outputs an update message as if the computer port had changed the xpt. If more than one level was used during the set xpt, then more messages will be output.

Panel	Reply
L=1, D=2, S=3	.UV002,003(cr)
L=1,2 D=20, S=12	.UV020,012(cr)
	.UA020,012(cr)
L=1,2,4 D=1, S=9	.UV001,009(cr)
	.UA001,009(cr)
	.UC001,009(cr)

Note that this message can be issued at any time, even just after the receipt of an external .SV message. Therefore the controlling computer software design must take this into account.

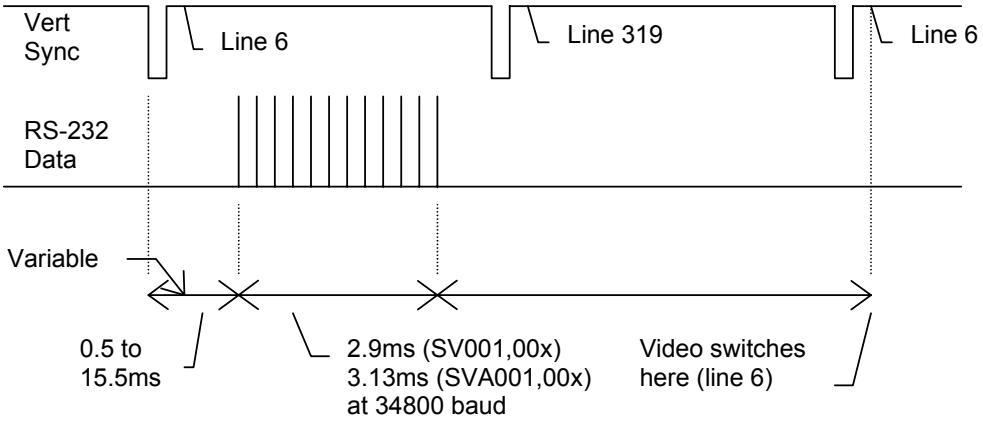
7. Examples

Note that (cr) = ASCII carriage return (code 0D hex).

From Computer	From Router	Description
	(cr).P(cr)	power on or reset
.SV003,001(cr)	.UV003,001(cr)	set dest-3 to srce-1 on level-1
.SV3,1(cr)	.UV003,001(cr)	set dest-3 to srce-1 on level-1
.SVAB02,001(cr)	.UVAB002,001(cr)	set dest-2 to srce-1 on levels-1,2,3
.SZ1,3(cr)	.E(cr)	level Z not allowed
.SB45,12	.E(cr)	dest to large
.IV1(cr)	.AV001,001(cr)	get status of level-1 dest-1
.IA3(cr)	.AA003,002(cr)	get status of level-2, dest-3
.IVABC9(cr)	.E(cr)	only one level can be interrogated
.LV5,-(cr)	.AV005,002V006,009V007,010V008,031V009,001V010,001V011,017V012,003(cr)	get status of level-1 dest-5 to dest-12
.LB1,1(cr)	.AB001,001B007,001B(cr)	only dest-1 and dest-7 using srce-1
	.UV007,003(cr)	system change by a control panel or another computer interface
.#01(cr)	.A(cr)	check to see if router connected
.?0C40(cr)	.A0C40,02,34,7F,2D,20,20,20,0D(cr)	read of system setup data
.!328E,01,02,56,7F(cr)	.A328E,01,02,56,7F (cr)	write of system setup data

8. Timing (Embedded Control Systems)

With an embedded control system (V5 firmware) operating at 38400 baud the system can cope with an overall crosspoint set rate of one change every 16ms when the serial link is directly into the master. For links into slave devices the timing and crosspoint set rate are dependent on the number of Q-Link devices.



The delay between the vertical sync and the RS232 message can be adjusted from 0.5ms to 12ms and this has no effect on the video switch point. Delays between 12ms and 18ms cause the video to start switching in this frame OR the next frame some of the time. A delay beyond 19ms causes the video always to switch in the next frame.

Transmit times (calculated and tested)

Baud Rate	Serial Format	1 byte time	12 byte time
38400	1 start, 7 data, 1 stop	0.234ms	2.80ms
38400	1 start, 8 data, 1 stop	0.260ms	3.12ms
9600	1 start, 7 data, 1 stop	0.938ms	11.26ms
9600	1 start, 8 data, 1 stop	1.041ms	12.49ms

The current message structure packs the destination and source numbers out to 3 digits (001, 002). The router will allow numbers without leading zeros (1,2).

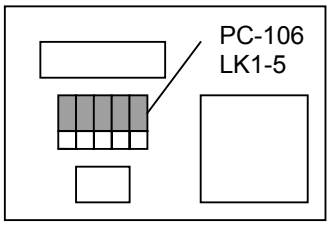
Leading zeros can be removed which will vary the message length from 7 bytes (.SVA1,2cr) to 10 bytes (.SVA32,32cr). At 38400, 7 data this gives a message time of between 1.872ms and 2.340ms. This will also slightly reduce the routers internal processing time.

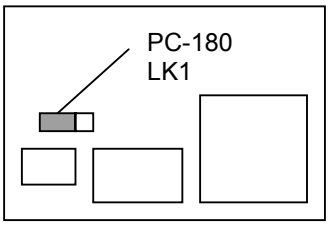
On an eight level system of 128x128, the maximum message length will be 18 bytes e.g. .SVABCDEFG123,123cr.

9. Older Systems

Older products that do NOT use the FU-0003 processor must have a Computer Interface fitted before the RS232/422 port can be used. As this is an option it must be ordered separately. RS232 or RS422 are link selectable on the CI-0001. There are two versions of this

PCB Number	Link(s)	RS-232	RS-422	Supplied
PC106	LK1-LK5	Away from U3	Toward U3	Up to Dec 1996
PC180	SELECT	232 position	422 position	Dec 1996 onwards





The older products operate at 9600, 8 data bits, no parity, and 1 stop bit. Other baud rates and parity options can be factory selected.