5.0   IF-11 GPIB / RS232 / LAN CONTROL MODULE

Contents

Cautions:

1) Cytec allows the user complete control over the switches and there is nothing built into the firmware that prevents you from connecting things that shouldn't be connected. You should always become familiar with the switch and test your code without any live signals connected! **Systems that switch high voltage, high current, power supplies or high frequency signals are especially dangerous since connecting the wrong things can result in damage to the switch, damage to other pieces of expensive equipment, or personal injury to you or others!**

2) Microwave relay systems present a particular problem since they will often allow you to turn on more than one channel on a specific multi-port relay. In normal use, this would create a problem since the relay would no longer maintain characteristic impedance, and the user would most likely notice. But when testing the system or running code without signals connected this will not be obvious and no warning will be issued. **Many microwave relays will overheat and be damaged if multiple ports on a single relay are turned on at once.** Under certain circumstances this can happen as quickly as 60 seconds. Please gain an understanding of your system before simply turning switch points on and off!

Introduction

CYTEC's IF-11 RS232/LAN Control Module, **Drwg**. **#11-14-50**, is designed to control single chassis mainframes. Three forms of control are available on the module: IEEE488 (GPIB), RS232 and Ethernet LAN as well as an optional manual control. All four interfaces may be active and used simultaneously.

Interface options: GPIB, RS232 and LAN are standard but the Manual Control must be specified when purchasing the system. On some systems where panel space is limited, only two of the three interface connectors may be included.

Non-volatile RAM
The control module contains non-volatile memory allowing the system to be configured for various matrix addressing combinations. Interface controls for Baud Rate, Answerback, Echo, Verbose, TCP/IP, Netmask and Port numbers are also saved in this memory allowing these functions to be set-up once and then be restored every time the power is turned on.

Memory Sanitation Procedure
This control module includes a RCM6700 core module which contains volatile and non-volatile RAM. No user information other than switch configurations and interface settings are stored. If the system is ever removed from service or needs to be sanitized for disposal all Volatile and Non-Volatile RAM can be erased using one of the following methods. **Refer to Drwg # 11-14-10**.

1) Easiest with least damage. Remove cover. Locate IF-11, install Jumper JP1 across the pins. Cycle the power. All memory will be erased and the system will be set to factory default values. Remove Jumper JP1 and offset. System can be reset to user values when needed.

2) Permanent. Remove cove. Locate IF-11. Remove the RMC6700 core board by disconnecting the LAN cable and prying the module off the IF-11 board. Destroy the RMC6700 board. Unit is non functional until IF-11 has been replaced.

Basic Command Syntax

All commands are composed of alpha characters followed by a series of integer values separated by a white space character. The alpha characters are not case sensitive and will accept whitespace or no whitespace between the alpha characters and the first integer value.

There are two types of commands:

1) Set-up and Parameter commands which are used to establish control options and configuration set-ups. Typically, the set-up and parameter commands are only used when you first get the system and then the switch control commands are used to operate the switch. Some examples include P commands for system parameters, snet commands for LAN set-up and E command for turning RS232 Echo on and off. The integer value 73 is often used as an access control value on set-up commands to prevent inadvertent or unintentional commands from being processed.

   Examples:

   P6 0 73   Turn off hardware handshaking for RS232 control.
   Snettcpport 0 8088   set-up port number for TCP/IP socket 0.

2) Switch Control commands cause actions to occur at the connected switching system. These are the commands that are used to control what gets connected to what through the switch. These commands include L for latch, C for clear, etc..

   Examples:

   L0 2 17   Latch Matrix #0, Module #2, Switch #17
   C 0   Clear all connections in Matrix # 0.

Numeric entries must always be separated from each other by spaces!
Some Cytec programming examples may refer to Mod #, Rly # (Relay #).
The terms Switch (Sw) and Relay (Rly) mean the same thing.

Multiple commands may be sent as a single string but each command must be separated by a semi colon. Up to 50 characters are allowed on one line so the number of commands will be dependent on the characters required. Only the first 15 characters sent will actually show up on the front panel LCD display (if that option has been included). Cytec does not encourage using this programming technique as it makes status feedback almost unusable but we do recognize that there are specific instances where this can greatly speed up operation of the system.

Commands are covered in detail throughout the rest of the Control sections 5.1 through 5.8.

5.1   RS232 INTERFACE

Signal Connections
The control module is pre-configured at the factory to operate as Data Communications Equipment (DCE) per the EIA RS232D Standard. In this configuration, the module transmits on the RxD Pin and receives on the TxD Pin. RTS is required to be high for the control module to transmit and CTS is output high by the control module to indicate a ready for data state and low when busy. Jumpers on the module allow for reversing the functions of these pins (TxD/RxD, RTS/CTS) and operating as Data Terminal Equipment (DTE) although there is virtually no practical reason to do this anymore. The RS232 rear panel connector is a D9P (male) and can be run directly from a D9 computer com port with a straight through (one to one) D9S to D9S cable. A null modem cable will not work with the factory default settings! Adaptors are available at any computer store to convert from D25 to D9. Do not use any adaptor that also acts as a null modem converter. If you are building your own cables, consult CYTEC Corp., for D25 to D9 pin out conversion.

D9P (male) PIN OUTS

Pin	Signal	Function
1	DCD	Control Module provides +V.
2	RxD	Data out of Control Module.
3	TxD	Data in to Control Module.
4	DTR	*Data Terminal Ready (not used)*
5	Common	Signal Ground.
6	DSR	Control Module provides +V.
7	RTS	Control Module requires + V to transmit.
8	CTS	Control Module provides +V when ready
9	RI	*Ring Indicator (not used)*

COM Port Default Values

As delivered from Cytec, the system uses the default Windows Com Port settings shown below:

Bits per second (Baud)	**9600**
Data bits	**8**
Parity	**None**
Stop bits	**1**
Flow control	**Hardware**

Only Baud rate (Bits per second) and Flow Control (Hardware handshaking) can be modified. See **Section 5.1.1** for command syntax.

5.1.1 RS232 INTERFACE I/O CONFIGURATION

I/O Configuration
Module #11-14-50

Default values:

Baud Rate	9600
Hardware Handshaking	On
Answerback	On
Echo	Off
Verbose	Off

The RS232 interface can be accessed using any standard terminal emulation program such as Hyperterm (Windows). Connect a one to one D9 female to D9 female cable between the computer and the Cytec switch.

The first thing you should do is turn on Echo. This will enable you to see what you are typing. Make sure you turn echo back off when you are done with the terminal session. Echo being left on will normally interfere with programs written specifically to control the switch.

Echo
Echos the characters back to your screen while you type them so you can see what you type.

Command:	"E 0 73"	Turns Echo Off
	"E 1 73"	Turns Echo On

Answerback
Answerback allows the Control Module to return information to the com port. Answerback should almost always be left on. If Answerback is enabled, the Answerback byte **must** be read back by the requesting device. Failure to do so could have unpredictable results.

Command:	"A 0 73"	Turns Answerback Off
	"A 1 73"	Turns Answerback On

Verbose
Verbose causes the system to return more specific information when you request status or read answerback characters. It is sometimes helpful when troubleshooting but it slows the interface down a lot. While there may occasionally be a good reason to turn on Verbose during a Hyperterm session, it is almost never used in a programmatic interface. All of the same information could be generated in code based on the non-verbose responses without slowing down the RS232 interface.

Command:	"V 0 73"	Turns Verbose Off
	"V 1 73"	Turns Verbose On

Baud Rate

Baud rate is set at the factory at 9600 Baud. Change is under software control and the control module must be connected to a serial interface to effect the change.

Baud	Baud# n
2400	4
4800	5
9600	6
19200	7
38400	8
57600	9
115200	10
230400	11 (untested, you should consider LAN)
460800	12 (untested, you should consider LAN)

Command: "P19 n 73"
                     "P19 7 73"   sets baud rate to 19200.

If the Baud rate is inadvertently set to an unknown rate, the default value may be restored. See the section on Setting Defaults for procedure.

Obviously as soon as you reset the Cytec baud rate you will no longer be able to communicate with the switch until you reset the baud rate on your computer or communication device.

CTS/RTS Handshake

The Clear to Send (CTS) and Request To Send (RTS) hardware handshaking functions may be modified by the 'P6' command.

Command: "P6 handshake 73"

handshake = 0   Handshaking off
handshake = 1   Handshaking on (default)

Example
            "P6 0 73"   Turn handshaking off.

5.2   IEEE488 INTERFACE

Signal Connections
Per IEEE Std. 488.1-1987 specification.

Device Subsets
The following list summarizes the device subsets implemented by the control module:

SH1	Source Handshake: Complete capability.
AH1	Acceptor Handshake: Complete capability.
T6	Talk Functions: Talk capability, Serial Poll, no talk only mode, untalk if MLA.
TE0	No Talker extended address capability.
L4	Listen Functions: Listen capability, no listen only mode, unlisten if MTA.
LE0	No Listener extended address capability.
SR1	Service Request: complete capability.
RL1	Remote Local: complete capability.
PP0	Parallel Poll: no capability.
DC1	Device Clear: complete capability.
DT0	Device Trigger: no capability.
C0	Controller: no capability.

SRQ Response
The service request function (SRQ) may be used to signal when the control module has completed an operation. The Program command (P) allows for enabling or disabling the SRQ response. When enabled the SRQ line will be set true when the control module has completed the last command received. The line will be set regardless of completion status.

P 1 0 73   SRQ off, no SRQ function. Default.
P 1 1 73   SRQ active and set true upon completion of each command.

Completion status may be read by a serial poll (SPE, MTA) or by enabling the talk address alone (MTA). After reading the status the SRQ line will be sent false.

Serial Poll, Talk Mode

Upon being addressed to talk the control module will return a single byte with EOI true indicating the switch-point and command completion status of the last matrix operation. If the previous command was for multi-character output (status, interrogate, revision ...) all data must be read before reading the command completion character. If the SRQ line was asserted it will be set passive false.

Char	Hex	Completion Status
'0'	30	Successful Operation, switch open.
'1'	31	Successful Operation, switch closed.
'2'	32	Unknown Command, the first character of the command string was unrecognizable.
'3'	33	Incorrect entries, the number or type of entries was incorrect.
'4'	34	Entries out of limits, a switch-point was requested that was outside the limits of the specified matrix.
'5'	35	Invalid access code, the code number 73 was not included or incorrect in a command requiring the access code or in the wrong entry position.
		As above with SRQ true.
'p'	70	Success, Switch Open.
'q'	71	Success, Switch Closed.
'r'	72	Unknown Command.
's'	73	Incorrect entries.
't'	74	Limits Error.
'u'	75	Access Error.

Remote/Local Operation
The interface responds to the Go To Local (GTL) and Local Lockout (LLO) commands by enabling or disabling front panel controls. Response to LLO, GTL, and REN signals may be disabled by the Program command 'P 4 0 73' and enabled by 'P 4 1 73'. Enabling or disabling is stored in non-volatile memory and will not be affected by turning power off then on. Factory default is disabled.

Device Clear, Selected Device Clear
The interface will respond to both device clear (DCL) and selected device clear (SDC) by clearing any pending operations, aborting pending output and optionally clearing the control module.

The matrix clear feature may be disabled by the Program command 'P 3 0 73' and enabled by the command 'P 3 1 73'. Enabling or disabling is stored in non-volatile memory and will not be affected by turning power off then on.

5.2.1 IEEE488 INTERFACE TALK/LISTEN ADDRESS
Talk/Listen Address
Set for P14 Command. **See Section 5.6.1 Additional Matrix Programming/Setup, P Command, P14**.

5.2.2 IEEE488.2 SPECIFIC MATRIX COMMANDS

These commands are ignored by the RS232 interface.

*IDN? - Revision Number   (Same as Cytec "N" - Revision Command)

     Syntax: "*IDN?"

     The 'IDN?' command will cause the matrix to return its current revision number followed by an end of line.

Send:	"*idn?"	Request revision number.
Receive:	"Cytec VDX/32x32 11-01-13 1.0" eol	Text string indicating revision.

*RST - Reset (same as C - Clear command)

     The '*RST' command will clear (open all switches) in the matrix.

Send:	"*rst"	Reset.
Receive:	"0" eol	Returns '0'.

5.3   LAN INTERFACE

The Cytec LAN interface is a 10BaseT / 100BaseT auto-ranging Ethernet interface with an RJ45 connector. A Category 5e or better cable is recommended.

Default settings:

IP Address: 10.0.0.144
Netmask: 255.0.0.0
Gateway   0.0.0.0
Telnet Port: 23
TCP/IP Port 0:   8080
TCP/IP Port 1:   8081
Telnet Lockout: 0 (disabled) Telnet Sessions allowed by default.
Telnet Echo: 0 (disabled) Telnet Sessions do not echo.
TCP Idle   60

The Default LAN settings may be modified by any control port (RS232 or LAN). The most common methods are:

Telnet to default telnet port 23. Any Telnet program will work. Depending on the program you may have to turn on Echo to see the commands that you type. Some Telnet programs automatically echo even with ours turned off. If you turn on Echo, be sure to turn it back off when you are done. Telnet Echo is different from RS232 Echo.

Telnet via Raw TCP/IP to ports 8080 or 8081. This works the same as above except it will probably never automatically Echo.

Use Hyperterm, puTTy, Tera Term or any other terminal emulation program to communicate to the RS232 port from a com port on your computer. See section 5.1 for RS232 details.

Use Hyperterm, puTTy, Tera Term or any other terminal emulation program to Telnet to port 23 or speak Raw TCP/IP to ports 8080 or 8081.

TCP/IP commands

D   Display current settings

Typing the letter "d" and hitting enter will return a list of current settings:

A1, E1, V0   % Answerback = ON, Echo = ON, Verbose = OFF%
Baudnumber = 6, RS Handshaking = 1
IP Address = 10.0.0.26
Netmask = 255.255.255.0
Gateway = 0.0.0.0
Port0 = 8080, Port1 = 8081
TCP idle = 600
Telnetlock = 0, Telnet Echo = 0
Battery Ram = 0, Default List = 0

IFCONFIG   Set IP Address and Netmask

ifconfig aaa.aaa.aaa.aaa nnn.nnn.nnn.nnn

a = ip address in dotted decimal format
n = subnet mask in dotted decimal format

Example: ifconfig 10.0.0.100 255.0.0.0

Typing ifconfig and hitting the enter key will return the current settings.

Since you may be connected via Telnet to do this, **the IP address will not actually change until you reboot the Cytec switch**. This helps prevent anyone from mistakenly setting the IP to an unknown address by accident. It is a good idea to double check the settings with the D command before you reboot.

SNET TCP PORT   Set Port number for TCP/IP sockets

SNET TCP PORT n m where = equals one of two sockets and m is the port number

Example:

        snet tcp port 0 8088   socket 0 is port #8088
        snet tcp port 1 8089   socket 1 is port #8089

Port numbers must be between 1024 and 65535.

The Telnet port (23) may also be available. See TELNETLOCK command

TELNET LOCK   Disable Telnet socket.

TELNETLOCK n   where n = 0,1

Example:

        telnet lock 0   Connections to Port 23 are allowed
        telnet lock 1   Connections to Port 23 are not allowed

TELNET ECHO   Echo characters sent to Telnet socket.

TELNET ECHO n   where n = 0,1

Example:

        telnet echo 0   Character are not echoed back to telnet application
        telnet lock 1   Character are echoed back to telnet application
                        (May cause double characters).

GATEWAY

Hosts   xxx.xxx.xxx.xxx   // to set gateway

Hosts   //display gateway
Gateway = 0.0.0.0

SOCKET LIFE

The LAN Interface uses a program called TCP Tick to tell whether or not the network connection is intact between the computer that established the socket and the LAN interface. The socket will remain until the requesting IP address terminates the socket or there is a break in the network connection. Sockets will automatically terminate after a pre-determined period of inactivity. That period is set by the SNET TCP Idle commands. Each open socket needs activity to stay alive.

SNET TCP Idle n   (n=seconds)   (1 to 3600 sec)
Default = 60 sec

SNET TCP Idle   (display)
TCP Idle = 60

SNET TCP Idle 0   Socket never dies until the computer that established the socket kills it.

Caution!
Setting the TCP Idle to 0 will force the socket to stay alive until the program that established the socket kills it. **WARNING:** This can lead to issues if there is a network disconnect or the computer that established the socket locks up. If the computer that establishes the socket cannot kill the socket, no one will be able to connect to the switch until the Cytec unit is rebooted.

TCPAnswerback - Answerback
    Syntax: TCPANSWERBACK n   n = 0, 1 or 2

    Answerback will enable or disable the transmission of a single character followed by an end of line upon the completion of all commands. The Answerback character will be a 1 or 0 depending on what command is sent. It is used to verify that the command was accepted and can verify completion of relay control commands. See Section

    Eg.   "TCPANSWERBACK 0 "   Turn answerback off.
          "TCPANSWERBACK 1 "   Turn answerback on
          "TCPANSWERBACK 2 "   Turn answerback plus terminator on

    Note: TCPANSWERBACK 2:
    This setting appends a set of square brackets to the answerback byte.
    Eg.   Send: "L0 0"   Latch Module 0 Switch 0.
               Receive: "1[]"   End of line follows the terminator

5.4   **COMMAND FORMAT/COMPLETION**

COMMAND FORMAT

All commands consist of at least one ASCII character indicating the command followed by optional values. After the command string is sent, an End of Line Character must be sent to affect the command.

If values are included with the command, the first value does not need to be separated from the command; all subsequent values MUST be separated by spaces or commas, eg. L1 2.

Multiple commands may also be sent on one line. Commands must be separated by a semi-colon character. Command line length is limited to 19 characters.

> Examples: "L2 7;C"   Connects Input 2 to Output 7 then clear
> "U4 7;L 1 2"   Unlatch Mod 4, Sw 7 then Latch Mod 1, Sw 2

COMMAND COMPLETION

A code representing the last requested switch point status (open or closed) and command completion will be stored by the matrix.

If the LAN or RS232 answerback function is enabled, a single character followed by end of line will be sent upon completion of all commands. Answerback may also include a termination character. With IEEE488 interfaces enabling the talk address (MTA) will cause the single code byte to be transmitted along with the END line true. This code is also available by a serial poll. DIO1 will indicate status from the last operation affecting switch-point status and DIO2 through DIO4 will be the command completion code.

Note: Command Completion is NOT updated until the matrix finishes the requested operation.

Command Completion Codes

Char	Hex	Completion Status
'0'	30	Successful operation, switch open
'1'	31	Successful operation, switch closed
'2'	32	Unknown command, the first character of the command string was unrecognizable
'3'	33	Incorrect entries, the number or type of entries was incorrect
'4'	34	Entries out of limits, a switch point was requested that was outside the limits of the specified matrix
'5'	35	Invalid access code, the code number 73 was not included or entered incorrectly
'6'	36	Setup error. Incorrect setup parameter

5.4.1 End of Line Character (EOL)

A received end of line character will cause the control module to execute the ASCII command string. The end of line character may be sent as a carriage return (CR) or New Line / Line Feed (NL/LF) for RS232 interfaces and a New Line / Line Feed (NL/LF) for IEEE488 interfaces or LAN interfaces. The IEEE488 also allows for the END control line being true with the last data character to initiate the command.

Valid end of Lines:

CR	RS232 only
LF or NL	LAN, RS232 or IEEE488
CR and END	IEEE488
LF/NL and END	IEEE488
DAB and END	IEEE488

Note that the terms New Line and Line Feed are often used to mean the same thing.
Both are expressed as \n in most programming languages and are shown on the ASCII table as "LF".
LF = Line Feed / New Line represented as \n, on ASCII table it is Decimal 10, or Hex A (0xA).
CR = Carriage Return represented as \r, on ASCII table it is = Decimal 13 or Hex D (0xD).

When any data is returned from the switch, the data will also be followed by an End Of Line character (EOL).

RS232:

1. The end of line will be a carriage return with Echo disabled or a carriage return and line feed if Echo is enabled.
2. If Answerback is enabled, the answerback character and end of line will follow the status output and then the EOL.
3. If Answerback with terminator is enabled, the answerback and terminator character will follow the status or interrogate output and then the EOL.

LAN:

1. The end of line character will be a new line (/n).
2. If Answerback is enabled, the answerback character and end of line will follow the status output and then the EOL.
3. If Answerback with terminator is enabled, the answerback and terminator character will follow the status or interrogate output and then the EOL.

IEEE488:

1. The end of line character will be a line feed.
2. A DCL or SDC command will abort output.

Notes - All Interfaces: Upon requesting status output characters MUST be received by the requesting device. Failure to do this will prevent further use of the matrix.

Access Code

Some commands require an access code number to be included with the command. This code prevents inadvertent operation of system modifying commands. The access code is 73.

5.5   **SWITCH COMMANDS**

General Notes

For LAN and RS232, after sending any command the Cytec control will return an integer Answerback character if Answerback is ON. Answerback is turned on by default and is Cytec's preferred operation since it allows you to verify commands are accepted before continuing.

If the command was a switch operation command such as Latch (L) or Unlatch (U), the character will be a meaningful status response where 1 = switch latched and 0 = switch unlatched. This may be used to verify that the command was received correctly.

Any other commands sent will also generate an answerback character which may be either a 1 or 0 and either character will indicate the command was received but the value is meaningless so either is acceptable.

Answerback may be turned off when using LAN or RS232 although it is not recommended.
Answerback can also include a termination character for the LAN or RS232 interface.

Error Characters

If a command is sent incorrectly, an error character will be generated and added to the answerback character. Since the answerback character may be a 1 or 0, there may be two values for error characters as described below.

Answerback returned:

Dec	Hex	
0	30	Latch completed without errors.
1	31	Unlatch completed without errors.
2 or 3	32 or 33	Unknown command, first character unrecognizable.
4 or 5	34 or 35	Incorrect entries, number or type of entries incorrect.
6 or 7	36 or 37	Entries out of limits, switch point out of usable range.
8 or 9	38 or 39	Invalid access code, number 73 not included when required.

COMMAND FORMAT

All commands consist of at least one ASCII character indicating the command followed by optional values. The optional values are normally the Module# and Switch# address integers or a set-up command parameter. After the command string is sent, a Carriage Return (CR) or Line Feed (LF) character must be sent to affect the command.

If values are included with the command, the first value does not need to be separated from the command; all subsequent integer values MUST be separated by spaces or commas. While the system will accept a space character between the command and the first value, it is unnecessary and simply adds characters.
The command character is not case sensitive and may be sent as UPPER or lower case. Our examples always use upper case for clarity since "l" and "1" look entirely too similar in most fonts.

Examples:

L1 2	Latch Module #1, Switch #2 (Turn it on)
U3 14	Unlatch Module #3, Switch #14 (Turn it off)
L14 12	Latch Module #14, Switch #12 (Turn it on)
C	Clear command needs no values. (Turn all relays off)

Also acceptable:

L 1 2	Space between L and 1. Latch Module #1, Switch #2 (Turn it on)
u3,14	Comma between 3 and 14. Unlatch Module #3, Switch #14 (Turn it off)
L 14,12	Space between command, comma separator.
c	Lowercase works fine. Clear command needs no values.

Not acceptable:

L12	No separators. Will accept but completely misinterpret command or give error.
U,3 14	Will give unknown command error.
L14, 12	Too many separators between values. Results unpredictable.

Multiple commands may also be sent on one line. Commands must be separated by a semi-colon character. Command line length is limited to 19 characters.

Examples:		
	"L2 7;L3 5"	Latch Module 2, Switch 7 and Module 3, Switch 5.
	"U3 5;L0 2;L1 3"	Unlatch Mod 3, Sw 5, Latch Mod 0 Sw 2 and Mod 1, Sw 3.

5.5.1   DELAYS TO PREVENT ERRORS

It is important to recognize that with modern computers and control interfaces, it is possible to stream commands to the switch matrix faster than the relays can physically operate. Many electro-mechanical relays may take between 2 to 20 ms to close or open. This can result in unpredictable results if certain operations are streamed together without considering this delay.

A good example of this type of problem occurs if a Latch command is sent and is immediately followed by a status request. Many of Cytec's products actually base status on current flow through the relay drives so it is possible to send a command and request status before the relay has physically operated, resulting in incorrect status feedback.

Typically, a 5 to 20 ms delay between commands requiring feedback can ensure that this is never an issue.

5.5.2   ANSWERBACK CHARACTER

Both the LAN and RS232 controls allow an Answerback Character to be returned whenever a command is sent. Answerback is turned on by default so this character must be read back before any other commands can be issued or there is a good chance that the control will hang. The answerback character is simply a "1" or "0" followed by end of line.

Answerback may be turned off using the "A0 73" command. See sections 5.11 (RS232) and 5.3 (LAN) for complete command syntax.

What does the Answerback Character Mean?

The answerback character simply confirms that the command was accepted and completed. It is useful for verifying that the command you sent was processed.

For most commands, the character being a "1" or "0" does not matter. Either character simply means that the command was processed and the value can be thought of as a "Don't care" Boolean.

For the L (Latch) and U (Unlatch) commands, the value tells you the status of the switch point.
So when you send an L command, the answerback character should always be a "1".
When you send a U command, the answerback should always be a "0".

5.5.3 SWITCH CONTROL COMMANDS

L,U,X - Latch, Unlatch, Multiplex Commands
> Syntax: Cmd Switch
>   Cmd Module, Switch
> *The specified switchpoint is operated on.*
>
> (Cmd = 'L', 'U' or 'X')
> **L = Latch = Turn switch ON**   Closes the specified point, all others unaffected.
> **U = Unlatch = Turn switch OFF**   Opens the specified point, all others are unaffected.
> **X = Multiplex= Clear + Latch**   Opens all points, then Latches the specified point.
>
> Eg.   "U2 3"   Module 2, Switch 3 is opened. (OFF)
>   "L0 1"   Module 0, Switch 1 is closed. (ON)
>   "L13 7"   Module 13, Switch 7 is closed. (ON)
>   "X3 0"   Clear all switch points (turn them all OFF) then Latch Module 3, Switch 0.

Note that the X command cannot be used if you need more than one switch on at a time. There is not a X command that only works on a single module.

If a single integer value is sent, the control module assumes it is a switch value and defaults to the last module value sent.

> Eg.   "L2 3"   Module 2, Switch 3 is closed (ON). Then,
>   "L4"   Assumes Module 2. Module 2, Switch 4 is closed (ON)

Note: For systems with less than or equal to 32 relay drives possible you may simply address them as 0 to 31 without using the Module #. These systems will accept either syntax.

> Eg.   "L3 5"   Module 3, Switch 5 is closed (ON).
>   "L29"   Module 3, Switch 5 is closed (ON). Same as above. Do the math.

Some Cytec programming examples may refer to Mod #, Rly # (Relay #).
The terms Switch (Sw) and Relay (Rly) mean the same thing.

For Unidirectional matrix switches, specifically DX, DXM, VDX, VDM and TX, the Module # may be thought of as Input #, and the Switch or Relay # may be thought of as the Output #. See section 5.6.

C - Clear Command
> Syntax: C
> *All points in the chassis are opened.*
>
> Eg.   "C"   All switches in the chassis are opened.

There is not a Clear Module # function. Clear simply turns off every switch in the chassis.
If you need a Clear module function it can be written in code fairly easily.
For IEEE488.2, The C command is the same as the *RST (reset) function.

5.5.4 STATUS AND INTEROGATE COMMANDS

The Status and Interrogate commands return information to the user so they can tell what state each switch point is in before proceeding. They can be used to simply check the switch configuration, to verify connections, or to prevent unwanted connections.

The information returned by these commands can be different depending on what type of system you have. Please find the Status or Interrogate section for your specific system before writing code that is dependent on what is returned.

S - Status Command

Syntax: S

                 S Module, Switch

Status may be requested of a single switch point or for the entire chassis. After receipt of the Status command the Matrix will return a character or string of characters representing the status, open or closed, of a switch point or switch points. A one, '1', signifies a closed switch point (ON) and a zero, '0', an open switch point (OFF).

In the case of a single switch point a single character is returned followed by an end of line.

For multiple switch points, a stream of 1's and 0's will be returned. How they are returned and what they mean is dependent on the type of system.

Status return is broken down into four different groups:

Section 5.5.4.1   **Systems with 8 to 32 relays drives possible.**
Section 5.5.4.2   **Systems with 64 to 512 relay drives possible.**
Section 5.5.4.3   **RJV systems which are unique.**
Section 5.5.4.4   **Large Matrix systems with over 512 switch points or solid state matrix systems.**

Please see the sections on the following pages for the information for your particular system

5.5.4.1   Status Returned Values up to 32 drives

For systems with 8 to 32 relay drives (any model designated as xxx/16 or xxx/32)
These Models Include:
>   **CXAR/16 and /32**
>   **CXM/16 and /32**
>   **HXV/16 and HXV/32**
>   **RS/8 or /16**
>   **GX/4 or /8 or /16 or /32**
>   **FO/16 or /32**

Model #'s are shown on the rear panel label. If the model # is simply a # such as "6754" you should see the addendum for the system or contact Cytec with questions if you do not know what it is returning.

The S command sent by itself will return a continuous string of 16 or 32 1's and 0's followed by the answerback character (if on) and the end of line character. The string is ordered from lowest value switch to highest value switch.

Eg. 1   CXAR/32 chassis
Send: "S"
Receive: "00010001000000000001100000000010" eol   32 switch points + Answerback + EOL

This is interpreted as Switches 3, 7, 20, 21 and 30 are ON, the rest are OFF.
In Module#, Switch # syntax this is:
>   Mod 0, Switches 3 and 7 are ON.
>   Mod 1, no switches are ON.
>   Mod 2, Switches 3 and 4 are ON.
>   Mod 3, Switch 7 is ON.
>   All other switches are OFF.

If the Answerback function is on, the last 1 or 0 before the EOL will be the Answerback character and the value is a "don't care". If Answerback + termination character is ON, there will also be the termination character [].

5.5.4.2   Status Returned Values 64 to 512 drives

For systems with 64 to 512 relay drives
(any model designated as xxx/64, xxx/128, xxx/256, xxx/512)
These Models Include:

> **LXA or LXB (all)**
> **CXAR/64, /128 or /256**
> **CXB systems (all)**
> **CXG, CXF, CXS systems (all)**
> **CXM/64, /128 or /256**
> **HXV/96 or /128, or /256**
> **RSM (all)**
> **RJM (all)**
> **JX (all)**
> **VX (all)**
> **PX (all)**
> **GX/64 or /128**
> **FO/64 or /128**

Model #'s are shown on the rear panel label. If the model # is simply a # such as "6754" you should see the addendum for the system or contact Cytec with questions if you do not know what it is returning.

The S command sent by itself will return a row / column pattern of 1's and 0's that mimic the LED display on the front panel (if equipt), where the columns are the Module # and the Rows are the Switch #.

Eg. 2   A 16 Module, 8 Switch Matrix (your configuration may be different):
          Send: "S"   Status of chassis
          Receive: "0001000100000000" eol   Switch 0, Module 3 and 7 closed
                      "0000000000000000" eol   Switch 1, none closed
                      "1111111111111111" eol   Switch 2, all closed
                      "1000000000000001" eol   Switch 3, Module 0 and 15 closed
                      "1010101010101010" eol   Switch 4, odd Modules closed
                      "0101010101010101" eol   Switch 5, even Modules closed
                      "0110000000000000" eol   Switch 6, Module 1 and 2 closed
                      "0000000000000110" eol   Switch 7, Module 13 and 14 closed
                       "0"eol   Answerback character

If the Answerback function is on, the last 1 or 0 before the EOL will be the Answerback character and the value is a "don't care". If Answerback + termination character is ON, there will also be the termination character [].

5.5.4.3 Status Returned Values RJV Systems

RJV systems are unique and the returned status is specific to those systems.
Skip this section if you do not have an RJV/48 or RJV/144 system.

S - Status Command
Syntax: S
          S Module, Switch
Status may be requested of a single switch point or for the entire chassis. After receipt of the Status command the Matrix will return a character or string of characters representing the status, open or closed, of a switch point or switch points. A one, '1', signifies a closed switch point and a zero, '0', an open switch point.

In the case of a single switch point a single character is returned followed by an end of line.
For multiple switch points a line of ones and zeros will be returned for each switch module in the system. For the RJV/48 there will be four rows of 24 1's or 0's. For the RJV/144 there will be 12 rows of 24 1's or 0's. Each line returned will be followed by an end of line (EOL). For LAN and RS232 interfaces there may also be an Answerback character before the EOL if Answerback is enabled.
At the end of all output rows an end of line will be sent for both *interfaces.* See the command completion and answerback sections for details.

Eg. 1
          Send: "S 3,4"   Test Module 3, Switch 4
          Receive: "0" eol   Switchpoint open, end of line.

Eg. 2   An RJV/48 with four 6x4 Modules:

Send: "S"   Status of chassis
Receive:
"000100010000000000000000" eol   Module 0, Switchs 3 and 7 closed
"000000000000000000000000" eol   Module 1, none closed
"100000010000000100000001" eol   Module 2, Switchs0, 7, 15 and 23 closed.
"000000100000000000100000" eol   Module 3, Switchs 6 and 18 closed.

An RJV/144 will return 12 rows with 24 1's or 0's in each row.

RJV Module Status table:
Use this table to relate switch #'s to Input and Output Ports.

RJV/12x1-X-XX (any 12x1 module)
These modules only use the first 12 characters of the returned string of 24 1's or 0's.
The second set of 12 characters are "don't cares" and may be returned as 1's or 0's.
They should be ignored.

Ports 0 to 11   NA
001000000000   000000000000

Indicates port 2 is on.

RJV/6x4-X-XX (all 6x4 modules)
These modules use all 24 characters and each common port has six associated 1's or 0's:
0 to 23 = Switch #'s

Port Labels	0	1	2	3	4	5
C0	0	4	8	12	16	20
C1	1	5	9	13	17	21
C2	2	6	10	14	18	22
C3	3	7	11	15	19	23

So the 24 characters line up as:
C0/0 C1/0 C2/0 C3/0 C0/1 C1/1 C2/1 C3/1 C0/2 C1/2 C2/2 C3/2 C0/3 C1/3 C2/3 C3/3 etc...

RJV/6x2 Switch Modules

These modules only use every other 2 characters in the returned string of 24 1's or 0's.

0 to 23 = Switch #'s

Port Labels	0	1	2	3	4	5
C0	0	4	8	12	16	20
C1	1	5	9	13	17	21
NA ignore!	2	6	10	14	18	22
NA ignore!	3	7	11	15	19	23

So the 24 characters line up as:

C0/0 C1/0 NA NA C0/1 C1/1 NA NA C0/2 C1/2 NA NA C0/3 C1/3 NA NA C0/4 C1/4 etc...

NA digits are floating and may return as 1's or 0's but must be ignored!

RJV/4(2x1)-X Switch Modules

These modules only use the first 4 characters of the returned string of 24 1's or 0's.

 2x1 # 0 to 3   NA

   0000   00000000000000000000   (0 = Normally Closed Position, 1 = Normally Open)

RJV/4(2x1)-X-Latching Switch Modules

These modules only use the first 8 characters of the returned string of 24 1's or 0's.

      2x1 #'s

C0/0 C1/1 C2/2 C3/3 C0/4 C1/5 C2/6 C3/7   NA

 0   0   0   0   0   0   0   0   0000000000000000

5.5.4.4   Status Returned Values Large Systems

**For systems with more than relay drives, large matrices, or solid state switches
These Models Include:**

> **VDX/16x16 or /32x32 (all)**
> **VDM/32x32 (all)**
> **DX/64x64 or /256x256 (all)**
> **DXM/64x64 or /128x128**
> **FX (all)**
> **TX (all)**

These systems do not make use of the S command but instead use the I command to return information.

Some of these systems would return up to 65,536 characters if they accepted the S command which becomes very difficult to deal with. Instead, they use the I command which only returns the address of switch points which are on. This limits the amount of returned data to a maximum of 256 addresses which is much easier and quicker to deal with.

These systems will interpret any "S" command sent as an "I" command and respond accordingly.

See the Interrogate Command Section for details.

Any other modules not listed please see addendum or call factory.

5.5.5 I – Interrogate Command

Syntax: I
The Interrogate function will return a list of all closed (ON) switch points. Each switch point will be followed by an "end of line" (EOL). The switch point is listed as the Module# and then Switch#. For matrix applications such as a 16x16 this often translates into "Input # then Output #" or "Output # then Input #". Since many systems are bi-directional Input vs Output may be dependent on how you are using it. For uni-directional systems, such as VDM, or DXM, the input vs output relationship will be carved in stone and you should be familiar with it.

Send: <I>   Request interrogation.
Receive: <Module#><comma><space><Switch#><EOL>

Eg.   Send   "I"
             Receive   "0, 0" eol   Module 0, Switch 0 Closed. End of line.
                           "1, 6" eol   Module 1, Switch 6 Closed. End of line.
                           "3, 2" eol   Module 3, Switch 2 Closed. End of line.
                           "0" eol   Answerback character (if enabled). End of line.

For system such as a DX/256x256 the "I" command may return up to 256 addresses. Be sure your buffer size can handle the amount of returned data.

For Unidirectional matrix switches, specifically DX, DXM, VDX, VDM and TX, the Module # may be thought of as Input #, and the Switch or Relay # may be thought of as the Output #. See section 5.6.

.

5.5.6 OTHER COMMANDS

F - Front Panel

Syntax: F n 73   n = 0 or 1

Front panel lock-out will be initiated by the receipt of a 0 character and enabled by the receipt of a 1 character followed by the access code. The access code prevents inadvertent lock-out from occurring. Lock-out will prevent any operation of the system from the front panel until it is terminated from the remote (F 1) or power is turned off then on. Preset to panel enabled at power on.

Eg.   "F 0,73"   Lock-out local operation.
      "F 1 73"   Enable local operation.

P - Program

Syntax: P n1,n2,73

The program command allows the operator to setup matrix dependent variables. These include matrix switch configuration and certain interface functions. Use of the P commands is complicated and varies greatly between systems. Your system should have been provided with the correct P command set-up.

If you need to change the matrix configuration, number of allowed modules, or other obscure set-up configurations on your system we recommend you contact Cytec and we can walk you through the P commands needed for your specific system. Please provide the serial # of your system when you contact us.

N - Revision Number   (Same as IEEE488 *IDN?)

Syntax: N

The 'N' command will cause the matrix to return its current revision number followed by an integer identifier, followed by an end of line.

Eg. Send: "N"   Request revision Number
      Receive: "Cytec 11-14-10-1, 1.23 0" eol   Text string indicating revision.

Where: "Cytec" = manufacturer.
       "11-14-10-1" = control module board number.
       "1.23" = Firmware Revision # (example).
       "0" = Integer identifier.

Note: When requesting the Revision number, all characters must be received before the system can be resumed.

NOTE - The text string received from the 'N' Command will vary depending on the type of system.

Integer identifier

The N command now includes a single byte which can be used as an identifier for Cytec systems. The identifier is a single byte integer so it may 0 to 255. WE do not assign this and it has no meaningful relationship to any product. It is simply a number which may be assigned to a chassis so that the end user can acknowledge that a specific Cytec chassis is communicating. It is up to the customer to assign the number and keep track of it. It allows them to poll multiple chassis and know that the one they are talking to is, for example, the JX/256 that they assigned the identifier "13" to.

Command to enter or change the number:

P90 n 73   where n is the number from 0 to 255

5.6   INPUT / OUTPUT vs MODULE / SWITCH NOMENCLATURE

Most of the switching systems sold by Cytec are completely bi-directional and can be used in a variety of ways by the customer so it is impossible for us to use the terms Input and Output, even though it is what probably makes the most sense to the end user when connecting signals to the switch.

We label and control the switches using Module# and Switch# to avoid this confusion since for most systems either can be considered an Input or an Output.

However some Cytec systems are uni-directional and do have assigned Inputs and Outputs:

Uni-directional Systems with defined Input and Output ports:

DX and DXM high speed digital switch matrix systems
VDX and VDM analog or digital switch matrix systems
TX analog systems
FX fiber optic systems

For all of these systems:

Module # = Input
Switch # = Output

Note that these systems will all allow multiple Outputs to be connected to a single Input, but will never allow multiple Inputs to be connected to one Output. There are internal controls to prevent this and attempting to do it will simply disconnect a previously set path.

5.7   MATRIX COMMAND SUMMARY

COMMAND   FUNCTION

L sw   Latch switch point.
L mod, sw

U sw   Unlatch switch point.
U mod, sw

X sw   Multiplex switch point.
X mod, sw

C   Clear entire system.

S   Return status.
S mod, sw

I   Interrogate Closed Points.

F 0/1 73   Disable/Enable Front Panel.

P parameter value 73   Program parameter.

N   Revision Number

RS232 Specific Commands

R baud, RTS/CTS 73   Baud Rate, RTS/CTS operation.

A 0/1 73   Disable/Enable Answerback.

E 0/1 73   Disable/Enable Echo.

V 0/1 73   Disable/Enable Verbose.

TCP/IP Specific Commands

TCPANSWERBACK 0/1/2   Disable/Enable Answerback/Termination

IFCONFIG aaa.aaa.aaa.aaa nnn.nnn.nnn.nnn   a = ip address in dotted decimal format
                                            n = subnet mask in dotted decimal format

SNET TCP PORT n m   Where n = equals one of two sockets and m is the port number

5.8   IF-9 / IF-11 (RS232/LAN/GPIB) DEFAULT CONFIGURATION SETTINGS

Should the system's non-volatile memory fail, default setup parameters will be placed in memory on power up. The unit may then be properly set-up by P commands.

Forcing Defaults

Default values may be forced by setting the Jumper on the control module and toggling power. See section 5.0, Memory Sanitation Procedure.

Default Values

P	0	1	One Matrix
P	7	0	NV RAM disabled
P	8	0	Load List #0 if NV RAM disabled
P	10	X	X = 4 Logical Modules for RJV/48 or 12 for RJV/144
P	20	12	12 Logical Switches
P	24	0	0 to 31 addressing
P	19	6	Baud rate 9600
P	90	0	ID# = 0
E	0		Echo off
V	0		Verbose off
A	1		Answerback on
IP			10.0.0.144
Netmask			255.0.0.0
Gateway			0.0.0.0
Port 0			8080
Port 1			8081
Socket Timeout			60 seconds
GPIB Address			7

5.9　　　　　LCD DISPLAY/KEYPAD MANUAL CONTROL OPTION

The Keypad/Display option **(Drwg.#6-057)** allows manual control of the matrix from the front panel. Keypad operation is always enabled at power on but may be disabled by the remote command, 'F'.

Display
The display contains two lines with sixteen characters per line. The top line displays matrix commands and numeric entry. The bottom line displays the status of the entry or operation. The display will also show the last command entered from the remote computer interface when the front panel is enabled.

Keypad
The keypad consists of ten numeric keys, four function keys, a space key and an enter key.

Key	Function
0-9	Numeric entries.
space	Delimits between numeric entries.
L	Latch operation.
U	Unlatch operation.
X	Multiplex operation.
C	Clear operation.
ENTR	Execute displayed operation.

Operation
A matrix command key, **L**, **U**, **X** or **C**, MUST be pressed before numeric entry keys. Pressing any key except a matrix command key causes the message **Enter Cmd First** to be displayed. After pressing a matrix command key the command and a cursor are displayed. The switch point to be operated on may now be entered with the numeric and space keys. The entry format is the same as described in the MATRIX OPERATION section and described briefly by the following table:

Command Key	Display Line 1	Line 2
L	Lat _	Enter Point
U	Unl _	Enter Point
X	Mux _	Enter Point
C	Clr _	Enter Matrix

The numeric keypad now allows selection of the Module and Relay (Input and Output) to be operated on. Each entry may be multiple digits and a space must be pressed between selections.

Key	Line 1	Line 2
L	Lat _	Enter Point
1	Lat 1_	
space	Lat 1 _	
2	Lat 1 2_	
3	Lat 1 23_	
Enter Key	Lat 1 23_	1

Number of Entries	Leftmost Entry	2nd Entry
1	Relay	-
2	Module	Relay

The **ENTR** key may now be pressed to execute the displayed operation. If the displayed entry is incorrect or the operation is not desired, pressing any matrix command key will clear the display and restart the entry.

Status Display

After the **ENTR** key is pressed, the displayed operation is attempted to be executed by the control module. If the execution is successful, a **Point Closed** or **Point Open** message will be displayed on line 2. If the operation cannot be executed, an error message will be displayed.

Line 2 Message	Status
Ready	Displayed after power on.
Enter Point	The ENTR key has not been pressed, command and selection mode.
Point Closed	The selected point was closed.
Point Open	The selected point was opened.
Points Open	All points opened, Clear operation.
***Err: limits	The selected point is outside the programmed size of the matrix.
***Err: entry	An incorrect entry was selected.

Front Panel Disable

The 'F' command allows enabling or disabling front panel operation. If the front panel is disabled, no operation can be performed from the keypad.

Remote Command	Line 1	Line 2
F 0 73	Panel	Disabled
F 1 73	Panel	Enabled

Contrast and LED Backlight Adjustment

Controls are provided to adjust the LCD contrast and LED backlight level. These controls should only need adjustment in extremely bright or dim environments or for acute viewing angles. Both LCD and LED circuits have temperature sensing elements that will automatically adjust the output level for changes in the ambient temperature.

APPENDIX B -QBASIC RS232 EXAMPLE PROGRAM

```
'Cytec Matrix Test Program for IBM-PC Compatibles using QBASIC and COM1 port.
'    16x16  (Change parameters X and Y for other configurations.)
'
'    Matrix should be set up for:
'         E 0 73        Echo off
'         V 0 73         Verbose off
'         A 1 73         Answerback on
'    Set-up may be done by any Terminal emulation program or
'    by adding these lines to the program, be sure to clear
'    input buffer after set-up.
'The Answerback character is read after every write operation and can be
'    tested for successful operation if desired.
'
CLS
OPEN "COM1:9600" FOR RANDOM AS #1
COM(1) ON
PRINT "Setting up and Clearing Matrix"
PRINT #1, "E0 73;V0 73;A1 73;C"
'Some delay gather Answers from the previous command
INPUT "Press ENTER to continue", a$
WHILE NOT EOF(1)
   INPUT #1, a$
WEND
'-----------------------------------------------------------
PRINT ""
PRINT "Cycling through 256 channels using Latch/Unlatch Commands"
FOR Y% = 0 TO 15   'Loop For Selecting Input
   FOR X% = 0 TO 15    'Loop For Selecting Output
        'Write Instruction To Matrix (Command,Input,Output)
      PRINT #1, "L" + STR$(Y%) + STR$(X%): INPUT #1, a$
        'Check Status (S module, relay)
      PRINT #1, "S" + STR$(Y%) + STR$(X%): INPUT #1, a$
      INPUT #1, Stat$
      IF Stat$ = "0" THEN PRINT "Error latching"; Y%; X%
      PRINT #1, "U" + STR$(Y%) + STR$(X%): INPUT #1, a$
      PRINT #1, "S" + STR$(Y%) + STR$(X%): INPUT #1, a$
      INPUT #1, Stat$
      IF Stat$ = "1" THEN PRINT "Error unlatching"; Y%; X%
    NEXT X%
NEXT Y%
'-----------------------------------------------------------
CLOSE #1
END
```

APPENDIX C - TURBO C/IEEE488 TEST PROGRAM

```c
/* CYTEC Matrix / National Instruments IEEE488 Test Program */
/*   This program was compiled under TURBO C ver. 2.0 and linked */
/*   with National Instruments file: TCIBS.OBJ.  Though not       */
/*   tested with other compilers it has been written following    */
/*   ANSI C rules and should be compatible with other compilers.*/
/*   National Instruments configuration program IBCONF was used  */
/*   to assign dev7 listen and talk addresses of the matrix (7).   */
#include <stdio.h>
#include "decl.h"
static int brd0, matrix;   /* device ID variables */
static char cmd_line[10];
/* ------------- initialize I/O ---------------------------- */
int init_GPIB()
{
        if ((brd0 = ibfind("gpib0")) < 0) /* 'Find' devices        */
                return(1);                        /* that were setup by    */
        ibsic(brd0);                    /* IBCONF program         */
        if ((matrix = ibfind("dev7")) < 0)     /* dev7 setup for Matrix */
                return(2);
        matrix_clear();
        return(0);
}
/* ------------- clear matrix ----------------------------- */
int matrix_clear()
{
ibwrt(matrix,"C",1);          /* "C", alternate methods of  */
/*   ibcmd(brd0,"\x14",1);          DCL, clearing the matrix   */
/*   ibclr(matrix);          */
}
/* ------------- switchpoint operation --------------------- */
int switchpoint_operation(cmd, inp, outp)
int cmd, inp, outp;
{
        int  sp_status;
        sprintf(cmd_line,"%c%d %d", cmd, inp, outp); /* build   */
        ibwrt(matrix,cmd_line,strlen(cmd_line));   /* and send  */
        ibrsp(matrix,&sp_status);
        return(sp_status & 0x3f);     /* ret status less SRQ bit  */
}
/* ------------- switchpoint status ----------------------- */
int switchpoint_status(inp, outp)
int inp, outp;
{
lkjhljhsdfkljdfgsdfgsg    sprintf(cmd_line,"S%d %d", inp, outp);   /* Request status */
        ibwrt(matrix,cmd_line,strlen(cmd_line));
        ibrd(matrix,cmd_line,10);                /* and read        */
        return(cmd_line[0] & 0x3f);
}
/* ********************************************************** */
#define N_INPUTS 16
#define N_OUTPUTS 8
#define RCV_BUFFER_SIZE N_INPUTS * N_OUTPUTS + N_INPUTS + 1
```

```c
int main()
{
        int i,inp,outp;
        char rcv_buffer[RCV_BUFFER_SIZE];
        printf("Cytec Matrix Test Program.\n\n");
        /* initialize and trap errors */
        if (init_GPIB()) {
                printf("Initialize error.\n");
                return(1);
        }
        /* Latch & Unlatch Switches */
        for (inp=0; inp<N_INPUTS; inp++)
                for (outp=0; outp<N_OUTPUTS; outp++) {
                        if (switchpoint_operation('L',inp,outp) != '1')
                                printf("Error: point %d %d not closed.\n",
                                                inp,outp);
                        if (switchpoint_operation('U',inp,outp) != '0')
                                printf("Error: point %d %d not open.\n",
                                                inp, outp);
    }
        /* Latch 'Random' switches */
        for (outp=0; outp<N_OUTPUTS; outp++) {
                switchpoint_operation('L',outp,outp);
                /* direct status read */
                if (switchpoint_status(outp,outp) != '1')
                        printf("Error: point %d %d not closed.\n",outp,outp);
        }
        /* request status of entire matrix */
        ibwrt(matrix,"S",1);
        ibrd(matrix,rcv_buffer,RCV_BUFFER_SIZE);

        /* display status */
        printf("Bytes read = %d.\n",ibcnt);
        for (i=0; i<ibcnt; i++) {
                if (rcv_buffer[i] == ';')
                        printf("\n");
                else
                        printf("%c",rcv_buffer[i]);
        }
        ibloc(matrix);    /* Leave local controls enabled when done */
}
```

```
/* Cytec Matrix Test Program for LAN              */
/* This program uses Microsoft's WS2_32 Library   */
/* and winsock2.h. These are available in the     */
/* Microsoft SDKs and can be downloaded from       */
/* Microsoft's Developer Network                   */
/* https://msdn.microsoft.com/en-us/default.aspx */

#include <stdio.h>
#include <winsock2.h>
#include <stdlib.h>   /* for exit() */

int init_LAN(int sock);
int point_ops(int sock,int cmd, int mod, int rly);
int matrix_clear(int sock);
void DieWithError(char *errorMessage);

#define MAX_MOD      4
#define MAX_RLY 12

int main(int argc, char *argv[])
{
        int sock;
        char *servIP = "10.0.0.144";  /*Default IP Address*?
        struct sockaddr_in servAddr; /* IP address */
        unsigned short servPort = 8080;  /* Port */
        int mod, rly,status;

        if (argc == 3)
        {
                servIP = argv[1];
                servPort = atoi(argv[2]);
        }


        WSADATA wsaData;      /* Structure for WinSock setup communication */
        WSAStartup(0x202, &wsaData);  /* Load Winsock 2.2 DLL */

        /* Create a reliable, stream socket using TCP */
        if ((sock = socket(PF_INET, SOCK_STREAM, IPPROTO_TCP))<0)
                DieWithError("socket() failed");

        /* Construct the server address structure */
        memset(&servAddr, 0, sizeof(servAddr));     /* Zero out structure */
        servAddr.sin_family = AF_INET;        /* Internet address family */
        servAddr.sin_addr.s_addr = inet_addr(servIP);   /* Server IP address */
        servAddr.sin_port  = htons(servPort); /* Server port */


/* Establish the connection to the server */
        if (connect(sock, (struct sockaddr *) &servAddr, sizeof(servAddr))<0)
                        DieWithError("connect() failed");


/* Initialize Device using init_LAN Function */
```

```c
        init_LAN(sock);

        /* Send Clear Command to Device with matrix_clear Function*/
        if ((status = matrix_clear(sock)) != 48)
                printf("Error clearing device/n");

        /* Simple looping through switchpoints */
        for (mod=0; mod<MAX_MOD; mod++)    {
                for (rly=0; rly<MAX_RLY; rly++)  {
                        if (((status = point_ops(sock,'L',mod,rly))) !=49)
                                printf("Error point %d %d %d not closed\n",mod,rly);
                        else
                                printf("Latched point %d %d\n",mod, rly);

                        if (((status = point_ops(sock,'U',mod,rly))) !=48)
                                printf("Error point %d %d %d not open\n",mod,rly);
                        else
                                printf("Unlatched point %d %d\n",mod, rly);
                }
          }
        }
        closesocket(sock);
        WSACleanup();  /* Cleanup Winsock */
        return 0;
}
/*--------------Initialize-------------------------*/

int init_LAN(int sock)
{
        char rcvString[40];    /* Buffer for device response */
        int rcvStringLen;      /* Length of device response */
        void DieWithError(char *errorMessage);
        /* Initialize Device */
        if ((send(sock, "E0 73;V0 73;TCPANSWERBACK 1\n", 28, 0)) != 28)
        {
                DieWithError("send() failed");
        }
        Sleep(1000);       /* Wait for Response from Device */
        if ((rcvStringLen = recv(sock, rcvString, 9, 0)) < 9)
        {
                DieWithError("recv() failed or connection closed prematurely");
        }
        rcvString[rcvStringLen] = '\0';
        return 0;
}
```

```c
/*----------------Clear Matrix---------------------*/
int matrix_clear(int sock)
{
        char rcvString[8];    /* Buffer for device response */
        int rcvStringLen;     /* Length of device response */

        if ((send(sock,"C\n",2,0)) != 2)
                DieWithError("send() failed");

        Sleep(200);   /* Wait for Response

        /* Receive Response from Device */
        if ((rcvStringLen = recv(sock, rcvString, 10, 0)) <= 0)
                DieWithError("recv() failed or connection closed prematurely");
        rcvString[rcvStringLen] = '\0';
        int status = rcvString[0] & 0x3f;
        return status;
}
/*-----------Switchpoint Operation------------------ */

int point_ops(int sock,int cmd, int mod, int rly)
{
        char cmd_str[40];          /* Formatted command string */
        char rcvString[8];    /* Buffer for device response */
        int rcvStringLen;     /* Length of device response */

        /* Format String */
        sprintf(cmd_str,"%c%d %d\n",cmd,mod,rly);

        /* Send Command to Device */
        if ((send(sock,cmd_str,strlen(cmd_str),0)) != strlen(cmd_str))
                        DieWithError("send() failed");

        Sleep(200);   /* Wait for Response

        /* Receive Response from Device */
        if ((rcvStringLen = recv(sock, rcvString, 10, 0)) <= 0)
                DieWithError("recv() failed or connection closed prematurely");
        rcvString[rcvStringLen] = '\0';
        int status = rcvString[0] & 0x3f;
        return status;
}

/*------------Error Handling Function--------------------*/

 void DieWithError(char *errorMessage)
{
        fprintf(stderr,"%s: %d\n", errorMessage, WSAGetLastError());
        getchar();
        exit(1);
}
```

APPENDIX F – Programming Examples

Cytec can provide programming examples for almost any Language. With the numerous programming languages available for the LAN interface, RS232 interface and GPIB interface it is increasingly difficult to provide them in a text file.

Please go to the Cytec support page at

http://cytec-ate.com/support.htm

or e-mail us at:

sales@cytec-ate.com

for complete programming examples in the language needed.

For tech support:

Call 1-800-346-3117
Or 585-381-4740