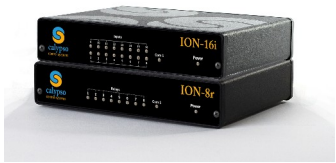
PRO I/O & ION SERIES DEVICE CONTROLLERS

USERS MANUAL



Main Office

944 O'Keefe Road
Hudson, WI 54016
(Tel) 715-381-9646
(Fax) 715-381-9647
www.calypsocontrol.com
sales@calypsocontrol.com
support@calypsocontrol.com

Manual: Pro I/O & ION Series
Rev 2.1

Table of Contents

Chapter 1: Introduction to Calypso Device Controllers

This chapter introduces Calypso device controllers to dealers and system integrators and includes product overview and technical support contact information. Detailed configuration and usage information is provided in subsequent chapters and in the attached troubleshooting guide. Note that this manual applies to all Calypso device controllers, including the Pro I/O and all ION Series controllers. Though controllers such as the Pro I/O, ION-e, ION-LT and others differ in terms of numbers of serial, IR, relay, input ports and even in size, they all share a common architecture, configuration interface and usage characteristics.

1.1 Device Controller Overview

Calypso manufactures intelligent, network-attached device controllers designed specifically to address the requirements of the professional audio-visual (Pro AV) industry. The Calypso architecture is unique to the industry in that it is designed around an embedded "event database" used to translate any incoming trigger (e.g., IR, serial, input, or Ethernet) into any sequence of outbound commands. The "personality" of the event database is defined for each project through an on-board web interface, used to assign inputs (i.e., triggers) and outputs (i.e., actions) to the database. Other basic configuration tasks such as network and COM port settings, and security settings are also addressed through the web interface.

Through the event database structure, Calypso controllers respond to any device capable of sending IR, serial or Ethernet commands, or of opening/closing relays, including nearly all 3rd party remote controls and button panels. This architecture greatly reduces the cost and complexity of control solution development in that it removes the need to embed "control logic" within custom software. The event database eliminates all "if-then-else" control logic code-writing. Once the event database is configured - a task easily accomplished by a competent AV technician with no programming skills – output actions are triggered by an appropriate incoming serial, relay, IR or network command, or by simply calling appropriate "event numbers" from software or through a standard hyperlink. No special control code-writing training is required.

In addition to functioning through its event database, Calypso controllers are capable of operating in "command line mode" where control commands are routed through the device, bypassing the event database entirely. This approach is most relevant when the controller is used with custom or 3rd party control interface software that requires direct access to control ports and/or bi-directional communications. This software interface would then be routed

through the device using the Calypso Network Access Protocol (CNAP). Thus bypassing the event database. It is important to note CNAP and the event database can both be used simultaneously. In addition, basic Calypso controller features include:

- Ethernet connectivity
- Built-in IR learner (only on controllers with IR outputs)
- Event database with 130 user-defined “triggers” and 512 user-defined output “actions”
- On-board web server with HTML user interface
- c_Link control hyperlink with 4 level 128 bit encrypted network access security
- 4-level 128 bit network encryption security
- User configurable space for custom web based control
- Internal clock for scheduled or repeated events

1.2 Control Hyperlink: c_Link

A hyperlink is a simple and common way for programs to communicate with one another. Most often found in web usage, hyperlinks have become standard across most programs and platforms, including Windows, Macintosh, Linux and many PDA variations. The Calypso architecture supports a version of hyperlink-initiated control, called c_Link, which allows standard hyperlink calls to be used to trigger control events. The only information required to use c_Link is the network address of the controller (i.e., IP address), a Calypso control header, and the number of the event within the event database that corresponds with the desired control actions.

Hyperlink control example: http://192.168.1.101/button.cgi?Event=1

The c_Link architecture not only brings enormous simplicity to control interface development, it provides the basis for integrating control functions into any hyperlink compatible platform or software like Microsoft PowerPoint or Windows desktop icons. The goal, of course, is to place simple, effective device control options into the hands of end-users in the most convenient and useful manner.

1.3 Configuration Interface

The controller configuration interface, used for system configuration and to set up and manage the event database, is compatible with any standard browser on Windows, Macintosh and Linux platforms. With standard Ethernet access, the web application is launched by simply entering the IP address of the controller into the URL address window of the browser. The default IP address on all controllers is 192.168.1.101. Once launched, the user navigates through an intuitive web interface to complete system setup, learn IR signals, and test control commands.

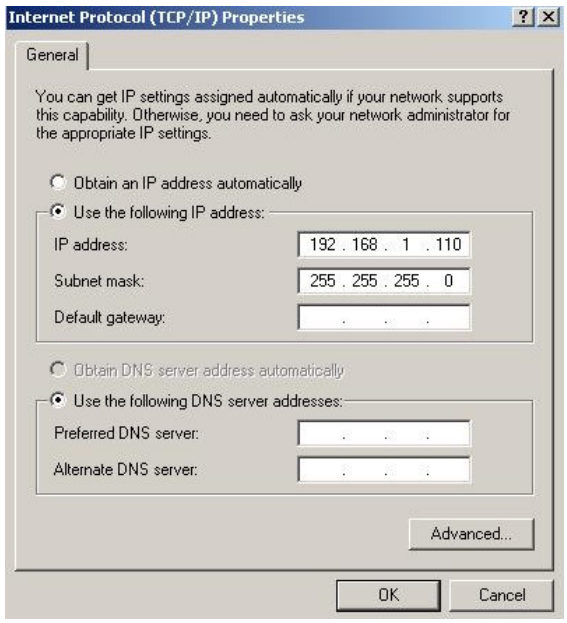
Chapter 2: Controller Setup

Calypso controllers are shipped from the factory with a default IP address of 192.168.1.101 and an empty event database. The process of configuring the controller for any specific application requires a network connection and use of the built-in browser-based application.

2.1 Connecting to the Controller for the First Time

Controller setup and configuration requires a network connection between the user's computer and the controller itself using the default network IP address on the controller of 192.168.1.101. While there are a number of ways to configure any network, the approach that minimizes potential network conflicts is to use an Ethernet "crossover" cable to connect the user's PC directly to the controller. Steps include:

1. Connect one end of an Ethernet crossover cable to the Ethernet port on the PC
2. Connect the other end of the Ethernet crossover cable to the Ethernet port on the controller
3. Change the PC's default IP address, found within the computer's TCP/IP protocol setup screen, so that it matches the default network of the controller. In other words, the default IP address of the user's computer must be equal to 192.168.1.X where X is any 3-digit number between 1 and 254, other than 101 (which is the default number used by the controller). In most cases, the screen used to set the PC's IP address looks as follows and is found within the Windows Control Panel for Network Setup Properties:



4. Launch an Internet browser and type the default controller IP address (192.168.1.101) into the browser's URL address window. This will automatically launch the controller's configuration application.
5. After controller configuration is complete, it is important to return the IP settings of the user's PC to its previous state. Make sure you record all settings prior to making any changes.
6. Resetting the IP and Passwords: if the controller needs to have the IP address or the password set back to the factory default use the following prodedure:

- Power down the controller
- Short the RX and TX of com 1
- Power up the controller
- Remove the RX and TX short before 4 seconds

The password and IP address will now be set back to factory default

2.2 Setup and Configuration

2.2.1 Logging In

The user must first log into the interface using the administration user name and default password.

Administration user name: admin
Default password: calypso

2.2.2 Interface Basics

Links to each administration page are visible at all times in the permanent, expanded menu bar that appears on the left side of the screen. Once logged in, users navigate between pages by using a cursor to select any of the available options at any time.

Menu bar page links are grouped into four topics, highlighted in blue on the menu, including page links to Main, Configuration, Command Line and Event Database. Links within each topic are orange and the "current" selected page is highlighted in red. By default, the interface launches in its "Main / Product View" mode, showing front and back images of the controller.

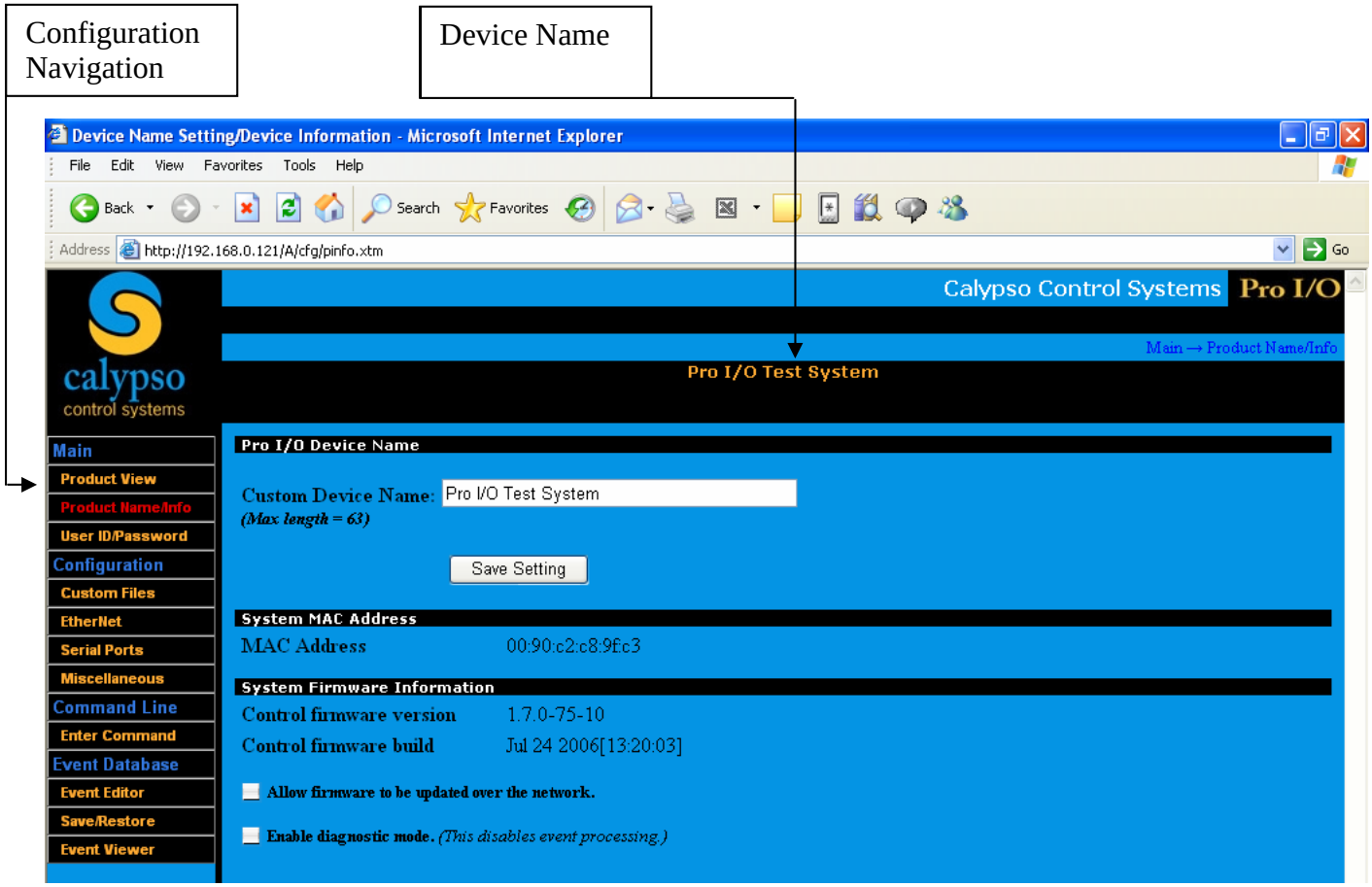


Figure 2.1 - Default Configuration Screen - Main / Product View

2.2.3 “Main” Administration Pages

Product View - see Figure 2.1

Product Name/Info - provides an opportunity to assign a logical “device name” to the controller with a 63 character maximum. This can be useful when using 3rd party network tools to manage large numbers of devices distributed across local- or wide-area networks. This screen displays the MAC address (fixed) of the controller should it be needed for network administration, as well as the unit’s current firmware version. “Allow firmware to be updated over the network” is a maintenance option that should not be selected unless the user or integrator is instructed to do so by Calypso technical support. In addition this page also allows the device to run in a diagnostics mode. Enabling diagnostics places the device in it’s diagnostic mode and shuts down the event database. Please contact the factory before running in this mode.

User ID/Password - allows an administrator to change user passwords and to select an appropriate security level.

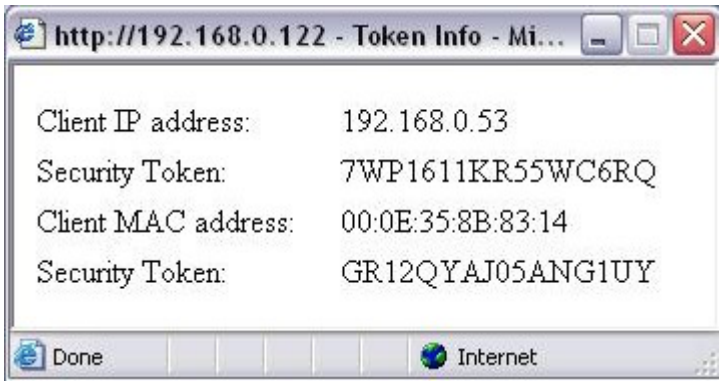
NONE: Allows any IP address to send control commands to the controller. In addition, this mode allows any and all pages to be accessed by any IP address.

ADMIN: Allows any IP address to send control commands, but users must have the “admin” password to access the administration and configuration pages.

ADMIN + USER: Same as ADMIN with the added security of requiring the “admin” or “user” password to access the controller’s internal custom web pages.

FULL: FULL is the maximum security setting for the Calypso controllers. In addition to the “ADMIN + USER” security restrictions, using this setting will block all unauthorized users from activating network-based control commands. Access to these commands is now accomplished through the use of security “tokens”, which are added to the commands in an appropriate manner. These tokens are linked to the IP address or the MAC address of the PC host that is sending the control command. Only the host with the matching IP or MAC address can initiate network-based commands using a particular security token.

Security Token Generation Process: When FULL is selected, the controllers "User ID/Password" page will display additional items, including a "Client IP" text field and a "Show Token Info" button. Clicking on this new button will generate token information for the IP entered in the text field, or if the field is left blank, for the IP of the PC making the request. (Note that if an IP address is entered, that host must be accessible online for the information to be generated.) This information will be displayed in a new window, similar to the picture shown below.



The first line represents the IP address of the PC that requested the security tokens. If the security token is to be linked to the IP address then you would use the token directly beneath the IP address line. If the security token were to be linked to the MAC address then you would use the token directly beneath the MAC address line.

How To Use A Security Token:

1. Control Hyperlinks (c_Link): This includes PowerPoint and any other hyperlink-based control requests.

PowerPoint: The format is the same as that seen in section 1.2, with the addition of a "Token" parameter as show here:

http://192.168.0.130/button.cgi?Event=20&Token=33P9RRX6U70VDK64

2. Remote Port (CNAP) requests:

Enclose the standard CNAP command inside a security wrapper: #SEC1{<token>, <command>}

For example,
#SEC1{33P9RRX6U70VDK64, #EVT20[];}
or #SEC1{33P9RRX6U70VDK64, #gpo1["close",D64];}

NOTE: SEC1 is case sensitive.

2.2.4 “Configuration” Administration Pages

Custom Files – this tab allows for loading and management of custom web pages that can be served to any web browser by the on-board controller web server. Note that these HTML and graphic files are completely separate and distinct from the web application used for controller setup and configuration. In most cases, this web interface is used to provide end-users with a custom, browser-based control interface served directly from the controller. The controller provides approximately 512k Bytes of memory for custom HTML and graphic file loading. This is covered in more depth under section 2.3.

Ethernet – allows the user to define basic network variables, including:

IP Address	Defined by network administrator
Subnet Mask	Defined by network administrator
Router IP Address	Optional; defined by network administrator if needed
Primary/Secondary NameServer IP Address	Defined by network administrator
Web Server Port	Fixed.
Remote Management Port	Port used by external software to send commands to the controller. Used for command-line (CNAP) functions only.
Remote Port Timeout	Set the duration in minutes before the controller “releases” the remote communication port after a period of inactive controller communication.

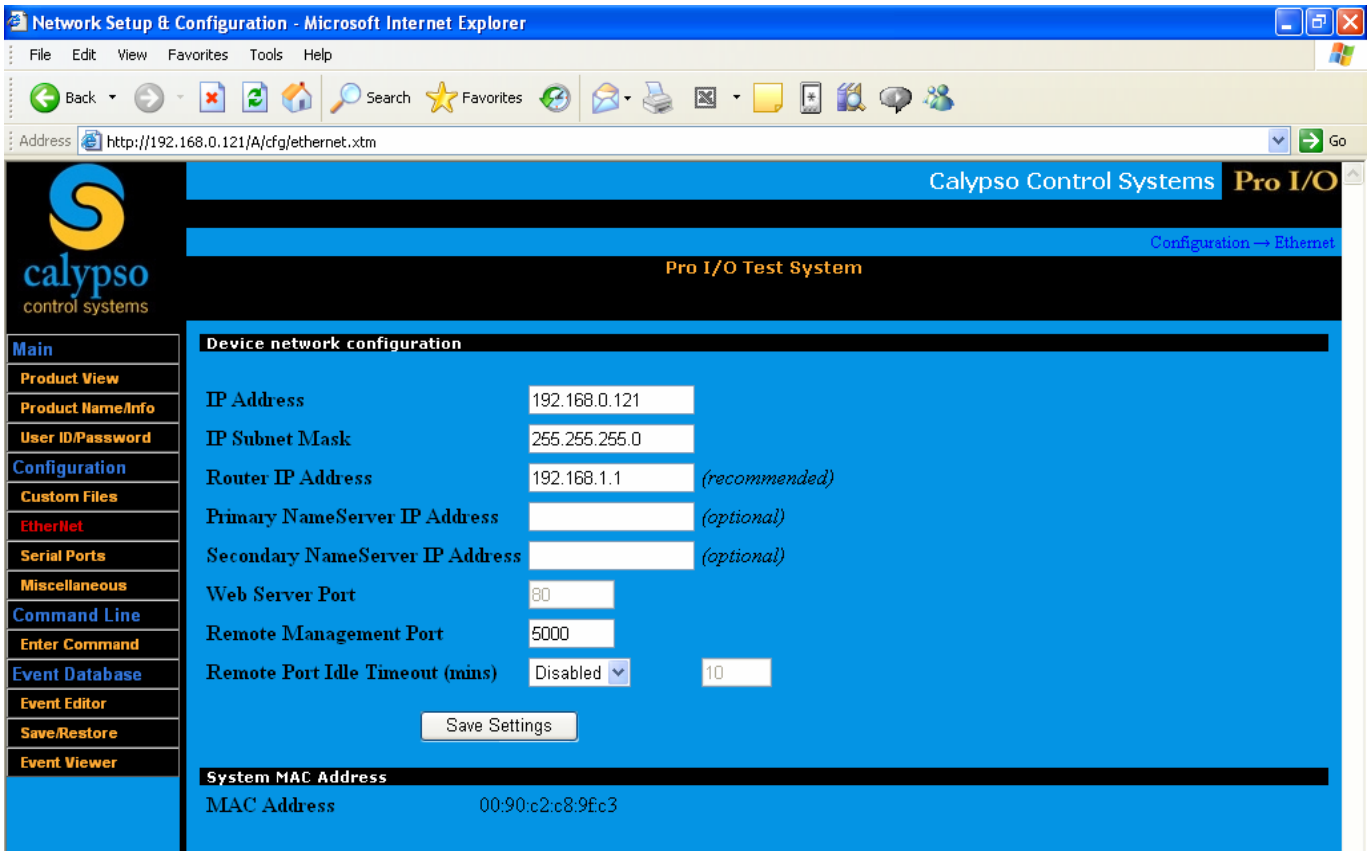


Figure 2.2 Ethernet configuration screen

Serial Port Settings - allows users to define standard COM port settings, including baud rate, stop bits and parity. Note that both RS232 and RS422 choices are available for the Pro I/O and ION-e products, whereas all other controllers are RS-232 only. Also, controllers with 2 or more serial ports offer multi-port modes such as pass-thru and replication, which can be enabled and disabled.

Miscellaneous – provides various functions, including:

- GPI de-bounce. For controllers with input functions this setting defines the amount of time, in milliseconds, for a contact switch to remain closed before it registers as a valid input trigger.

- IR Mode – Some controllers with IR functionality have IR inputs on both the front and back panels. The IR Mode pull down menu allows for selection of the appropriate input(s) routing.

- Events Per Page – the user/integrator can select how many events are displayed on the event viewer page.

- Set Time – set the day of the week and time for timer function reference. Time is entered using the 24-hour clock method.

2.2.5 "Command Line": Enter Command, including IR Learn.

This page is used for command line control and IR learning. For a complete list of available Event Database commands, see Chapter 3: Command Examples and Special Features.

- Test Commands – The command line window provides a convenient tool for assembling and testing relay, input, serial, IR, and network commands as available on each controller model. Using the pull down menu, users may select any available command, which automatically places an example into the "Send" window, with proper syntax. Variables such as COM port number, serial strings, relay port number, etc. are then modified by the user and are executed when the "Send CMD" button is pressed. Users have the option of copying and pasting appropriate commands from the "Send Command" window into the Event Database for use as either input triggers or output commands steered through appropriate serial, relay, IR, and network ports.

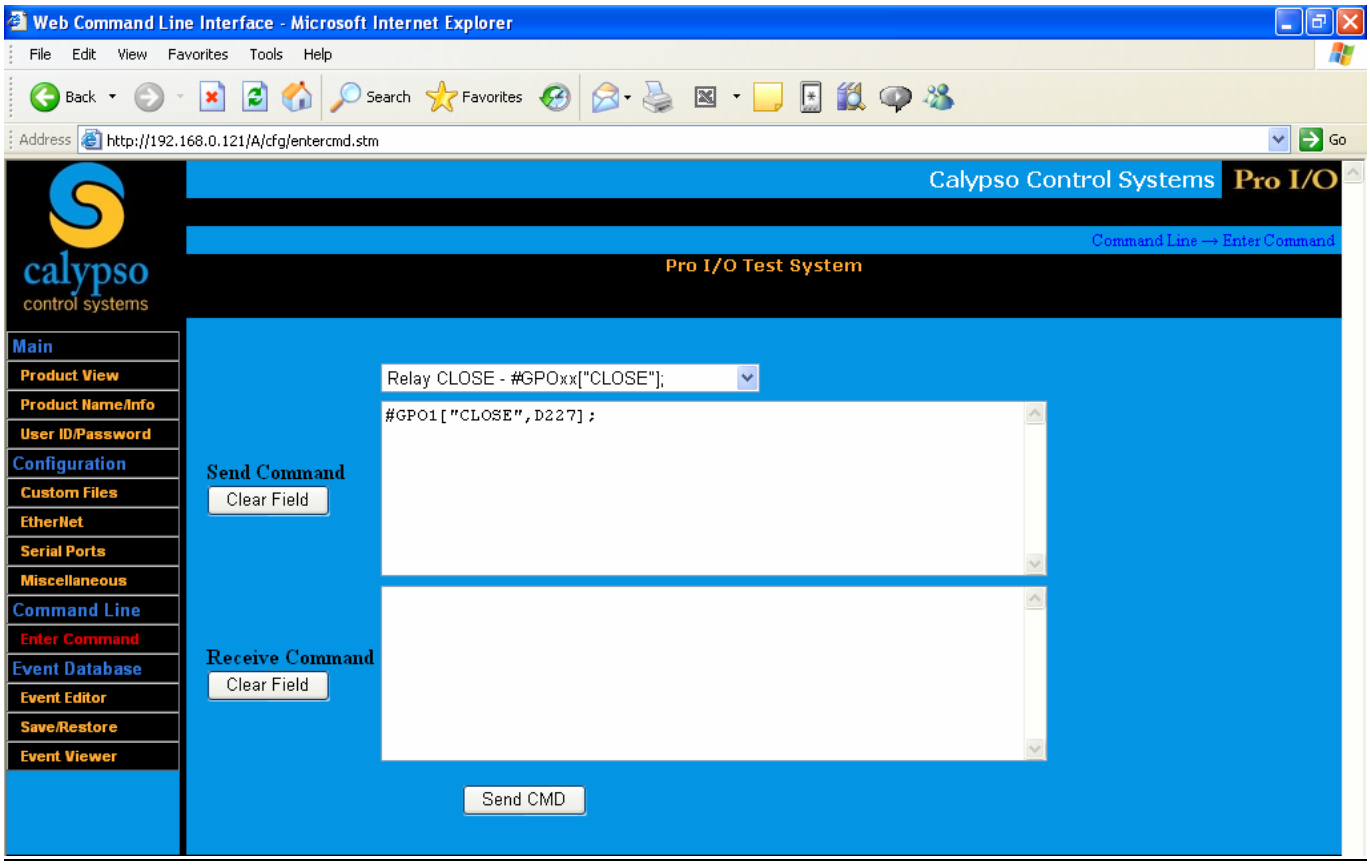


Figure 2.3 Command Line Example - close relay

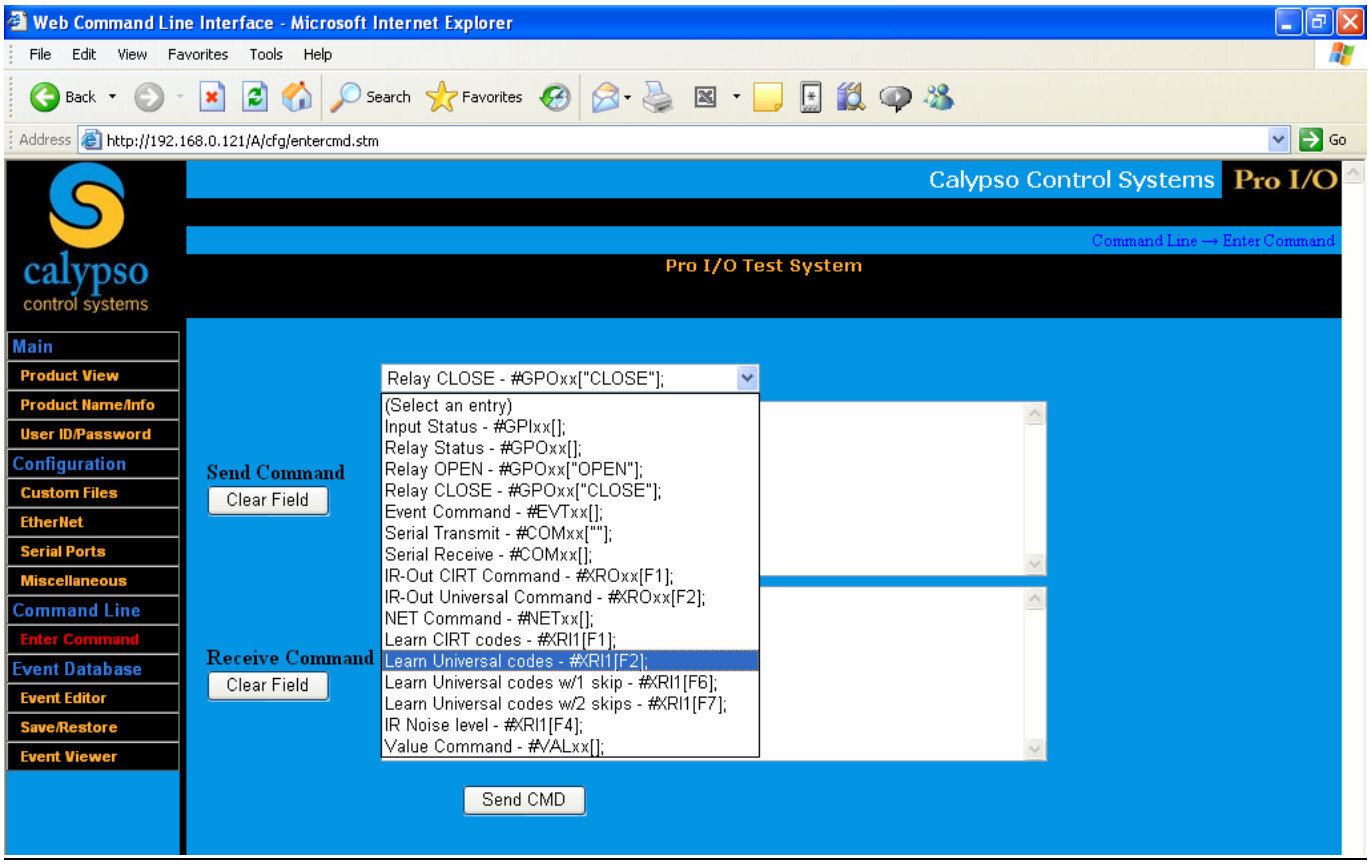


Figure 2.4 Command Line menu pull down options

IR Learning - In addition to providing a convenient testing tool, the pull-down window provides options that place the controller into "IR Learn" mode, for both CIRT[1] and Universal IR codes for controllers with IR input / output functionality.

- Select either the "Learn CIRT" or "Learn Universal" option from the pull down menu
- Press the "Send CMD" button
- Hold the external IR remote control within 6 inches of the front panel IR window and press the appropriate remote control button
- The IR string appears in the "Receive Command" window, and is available for copying and pasting to the Event Database.
- CIRT codes are encoded into a 4-digit sequence for convenience.

Note: CIRT (Calypso InfraRed Trigger) codes are restricted to Sony 12-bit sequences for use as input triggers. All other Sony IR codes are treated like standard "Universal IR" codes.

[1] Calypso IR Trigger (CIRT) is a collection of over 4000 4-char IR codes, derived from a larger set of Sony 12-bit IR codes. IR input triggers are limited to CIRT library commands. These specific codes were selected because (a) they use minimal memory and (b) are commonly found programmed into all universal remote controls, making for convenient IR learning and usage.

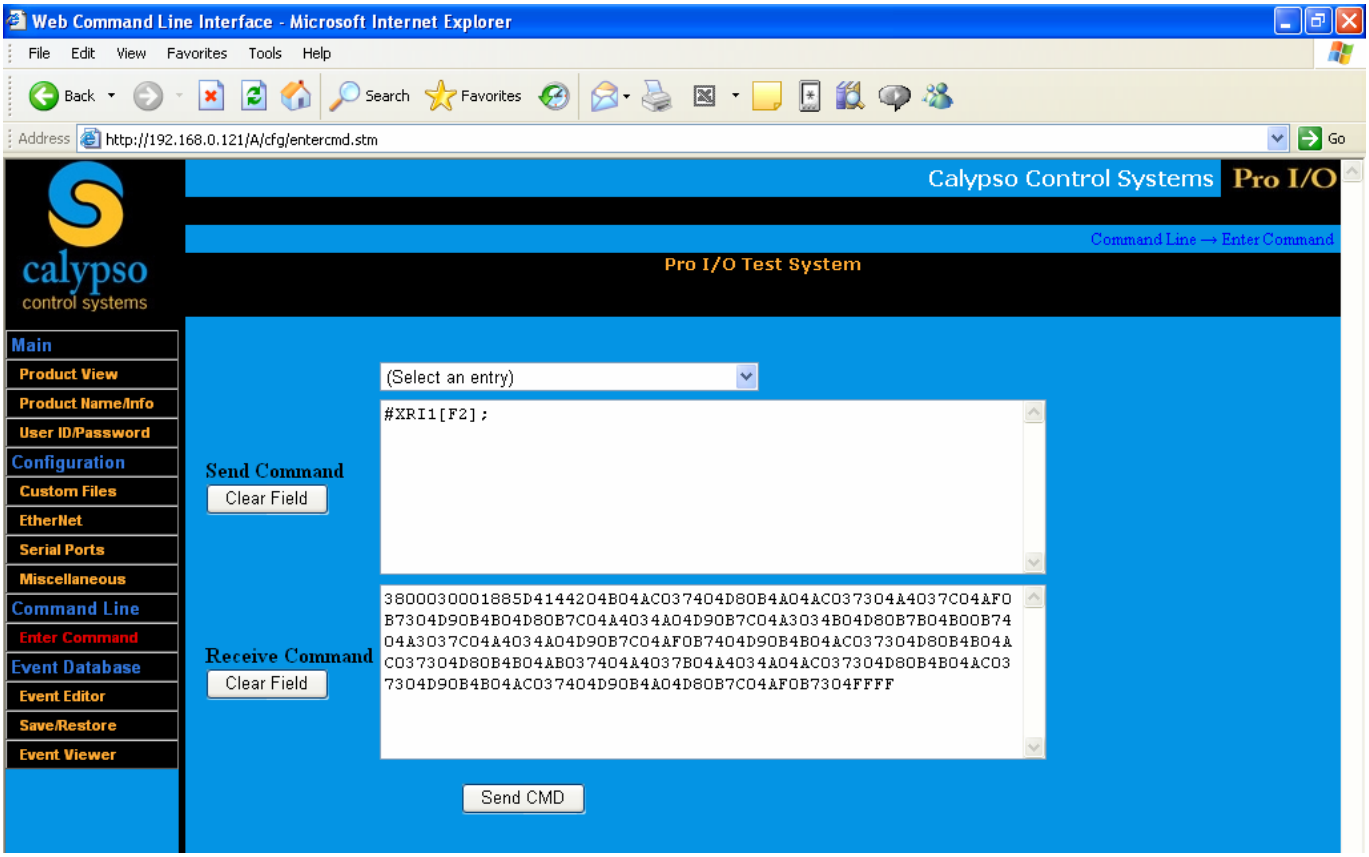


Figure 2.5 Learning Universal IR Codes

CIRT codes (Calypso InfraRed Triggers): Special usage for database event triggers

Calypso controllers with IR functionality have a unique feature used to receive and decode Sony 12-bit IR commands. This feature allows these codes to act as event triggers, making it possible to use standard off-the-shelf universal remote controls to control sophisticated “non-IR” systems. While operating in CIRT mode, if the Sony IR code bit structure is something other than 12-bit the controller will display an “invalid code” message. Once the appropriate Sony code is learned it is automatically encoded into a convenient 4-digit sequence. This new 4-digit code can now be copied and pasted into an Event and used as an IR trigger.

IR Event Trigger Inputs:

Controllers with IR functionality respond to CIRT codes from the front panel IR window and/or each of the back panel IR connectors. The Pro I/O and IONe controllers have the option of turning this function on or off from the Miscellaneous page.

Universal IR Command Header:

When an IR command is learned the controller inserts an “IR Header” into the front of the command. Each Hex byte of this header contains information pertaining to certain IR parameters and defines how the IR signal is to be transmitted. It is helpful to understand the first 6 bytes of this header when dealing with difficult IR devices.

Example IR header: 40 00 03 00 00 54

BYTE 1: Carrier frequency of the IR command (30Khz, 32khz, 36khz, 38khz, 40khz and 56khz) that is informational and cannot be changed.

BYTE 2: Idle time - a user adjustable delay time between IR bursts when the IR command is sent more than once. Allowable timing increments are 04h - 1C (must be in Hex form!), with 5ms delay for each increasing hex value. For example, a idle time value of 0A (decimal 10) would insert a delay of 50ms between the IR bursts. Default is 00h, which means that the controller will use the idle time that is part of the learned IR code.

BYTE 3: Repeat - selects how many times the IR command is to be repeated. Allowable values are from 01h to FFh (must be in Hex form!). Default is 03h and 00h is invalid.

BYTE 4: Repeat Mode - three selections: 00h = repeat the entire IR command, 01 = repeat the IR header only once, 02h = repeat the IR command with an offset. Default is 00h.

BYTE 5: Offset - selects an offset value if a repeat command needs to drop Bytes in addition to the header. Allowable values are 00h to 78h (must be in Hex form!).

BYTE 6: Data Length - displays the size of the data string in Hex form including the header.

2.2.6 “Event Database” Administration Pages

The event database acts as a type of translation table capable of converting incoming event triggers into outgoing control actions. The event database is the key enabling technology that provides Calypso controllers with a unique “personality” for any given application. Further, it forms the basis for an open architecture solution capable of integrating easily with nearly any user interface device from any manufacturer. In the course of setting up a Calypso controller, most of the configuration time will be spent working with the Event Database.

Event Editor - this is where the user will create and edit up to 130 “trigger” events (plus a special Power On Event 0) and up to 512 corresponding output “actions”. A trigger event is defined as any type of input signal the controller is capable of recognizing. Options that vary based upon each controller model include:

Event Triggers:

What	How	Restriction
CIRT code	IR window or IR wired connectors	CIRT code (12-bit Sony IR code)
Input (open/close)	General purpose input connectors	None
Serial String	Serial compare	COM1 only; max 8 character length match, ASCII or HEX
Direct IP port data string	NET compare - Ethernet connection	Max 8 character length match, ASCII or HEX
Time of Day or defined time interval	Internal clock	No seconds increments
Test value command	Compares against a value	Up to 50 unique value placeholders

Likewise, output actions are defined as any output signal the controller is capable of generating through its unique collection of output ports, and the Ethernet connection. A summary list of available event database actions includes:

Event Actions:

Command	Description
#GPO1[“close”]; #GPO1[“open”];	Close or Open relay 1.
#COM1[T1,"hello\r"];	Sends serial “hello” with a carriage return to com port 1.
#EVT1[ ];	Execute event number 1.
#NET[I192.168.0.100, P5000, "#EVT1[ ];"];	This command will send the data string “EVT1[ ];” via Ethernet to port 5000 of a device with the IP address of 192.168.0.100
#XRO1[F1, "08B9"];	IR output 1 will transmit the CIRT (Sony 12-bit) code 08B9
#XRO1[F2, "large pasted IR code"];	IR output 1 will send out a copied/pasted universal learned IR sequence
#TRG[“OFF”, 32];	The external input trigger for event 32 on the controller is disabled.
#ALM[“ON”,D227];	Turns on the internal alarm for 1 second. Only applies to the ION-LT and ION-LT2
#VAL1[12];	Save the value 12 into stored value location 1
#NOP[ ];	A valid default command that is displayed where no output action or input event trigger is defined.

For example, the user may define the “trigger” for event number 1 as a particular CIRT code and the corresponding output actions as a sequence of two serial commands followed by a network command sent to a second controller in a different location. Once defined within the event editor, the controller constantly “scans” all inputs, searching for an event that matches pre-defined “triggers”. When a matching event is recognized, the corresponding output actions are executed in rapid succession.

Note that it is often useful to include a simple description of each event within the appropriate description fields. Event descriptions are used within the “Event Viewer” to simplify the process of visually scanning through the entire list of events.

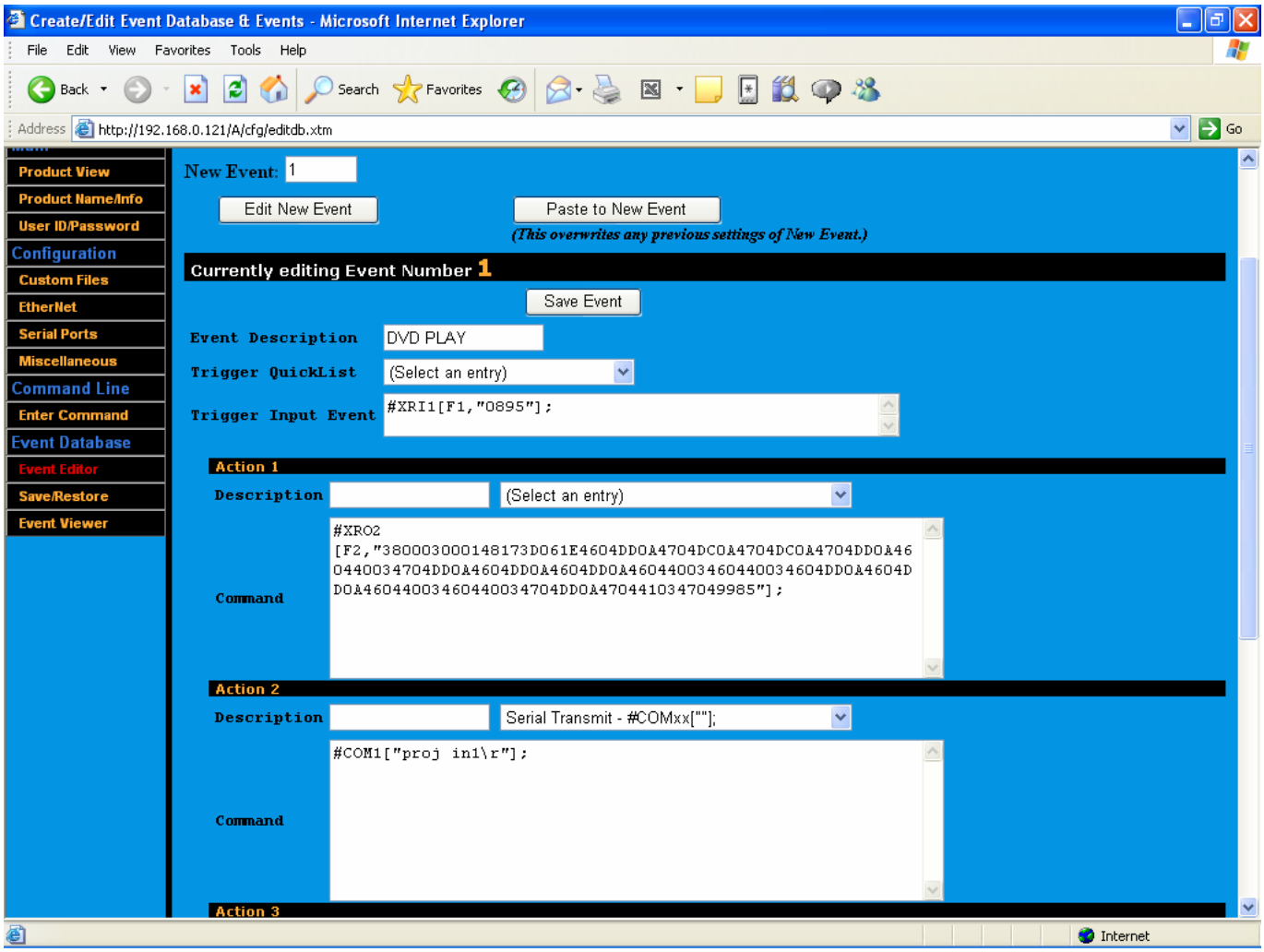


Figure 2.6 Event Editor Screen

⇨ Autostart Event. It is important to note that event number 0 (zero) is reserved for a control sequence triggered automatically upon system startup. Event number 0 otherwise functions the same as any other event as defined within the event database.

Save/Restore - The functions within this window allow users to save the database to a PC and to load an existing database from a PC to a controller. The goal is to save time otherwise spent manually entering multiple events in projects requiring the deployment of more than one controller or for projects that can take advantage of previously assembled databases.

IMPORTANT: When saved, a controller database file will be placed into the default web browser download location. From there the file can be moved and renamed if required. For example, the Pro I/O saves its database with the file name proio.edb and the ION-e saves it's file as ion_e.edb. The extension edb is the designator for "Event Database". Regardless of the number of events programmed in the event database the file size when saved should always be about 280K for any given controller. It is highly recommended that the security level be temporarily set to "None" before performing a database "Load". This is to avoid a potential for the upload to fail when security is on. Once the upload has completed, change the security level back to it previous desired setting.

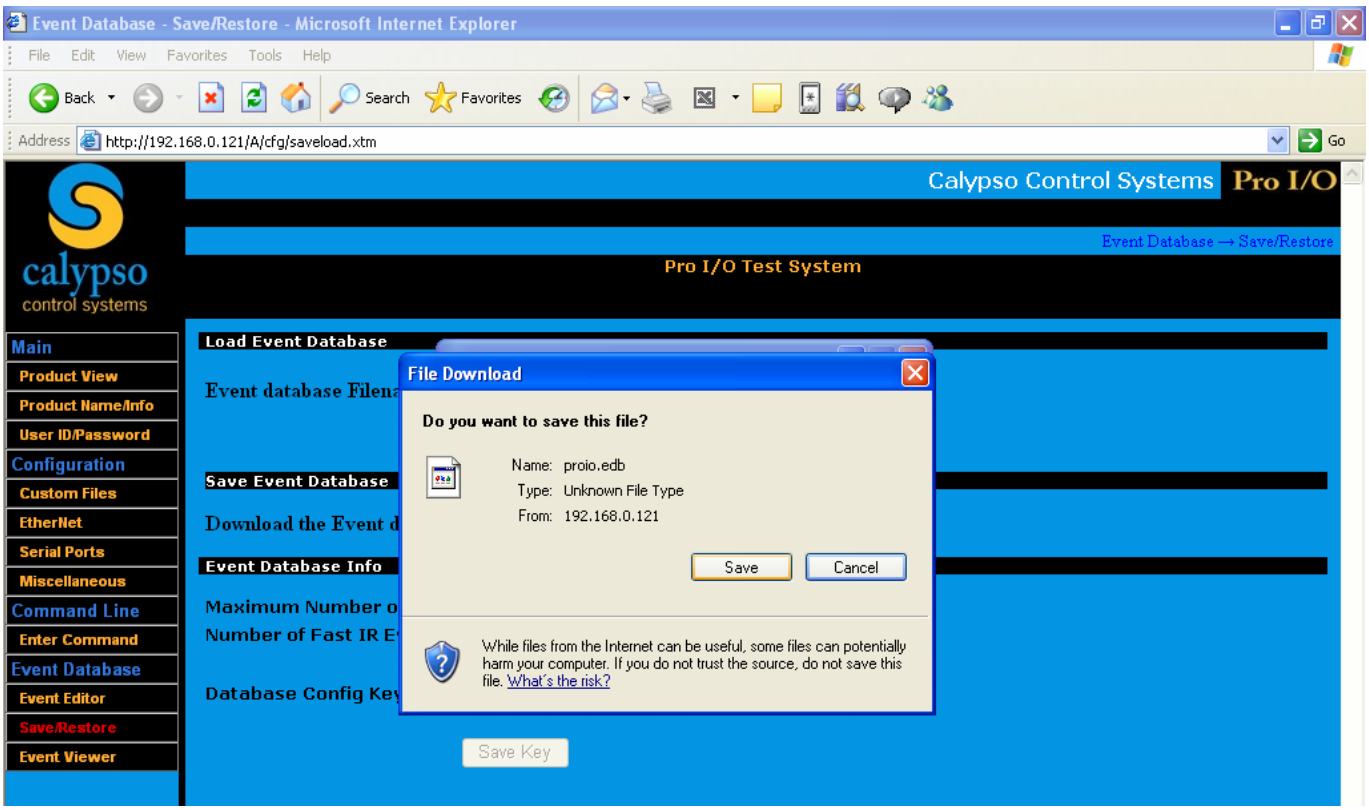


Figure 2.7 Event Database Save/Restore Screen

Event Viewer - As the size of the event database grows it becomes useful to navigate through the database in summary fashion rather than being forced to use the Event Editor to review each event. The Event Viewer provides a scrollable summary of the database along with several functions designed to speed the process of database setup.

The Event Viewer screen provides users with the ability to:

- Scroll through events, organized in rows
- View each event trigger and each of the four actions, if entered
- Hyperlink directly to the Event Editor screen for any given event
- Enable and disable events
- Test events
- Delete events, either individually or the entire database

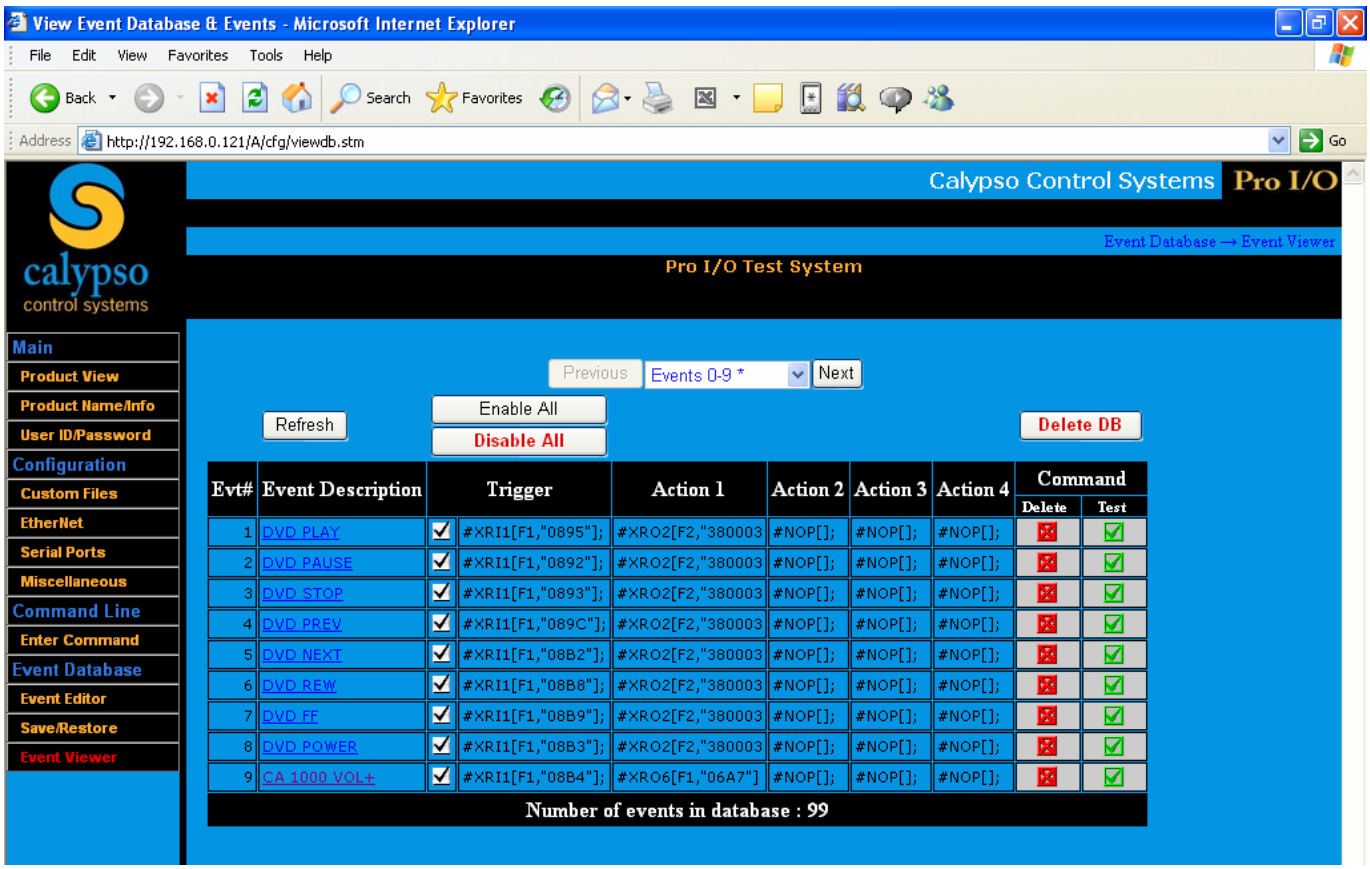
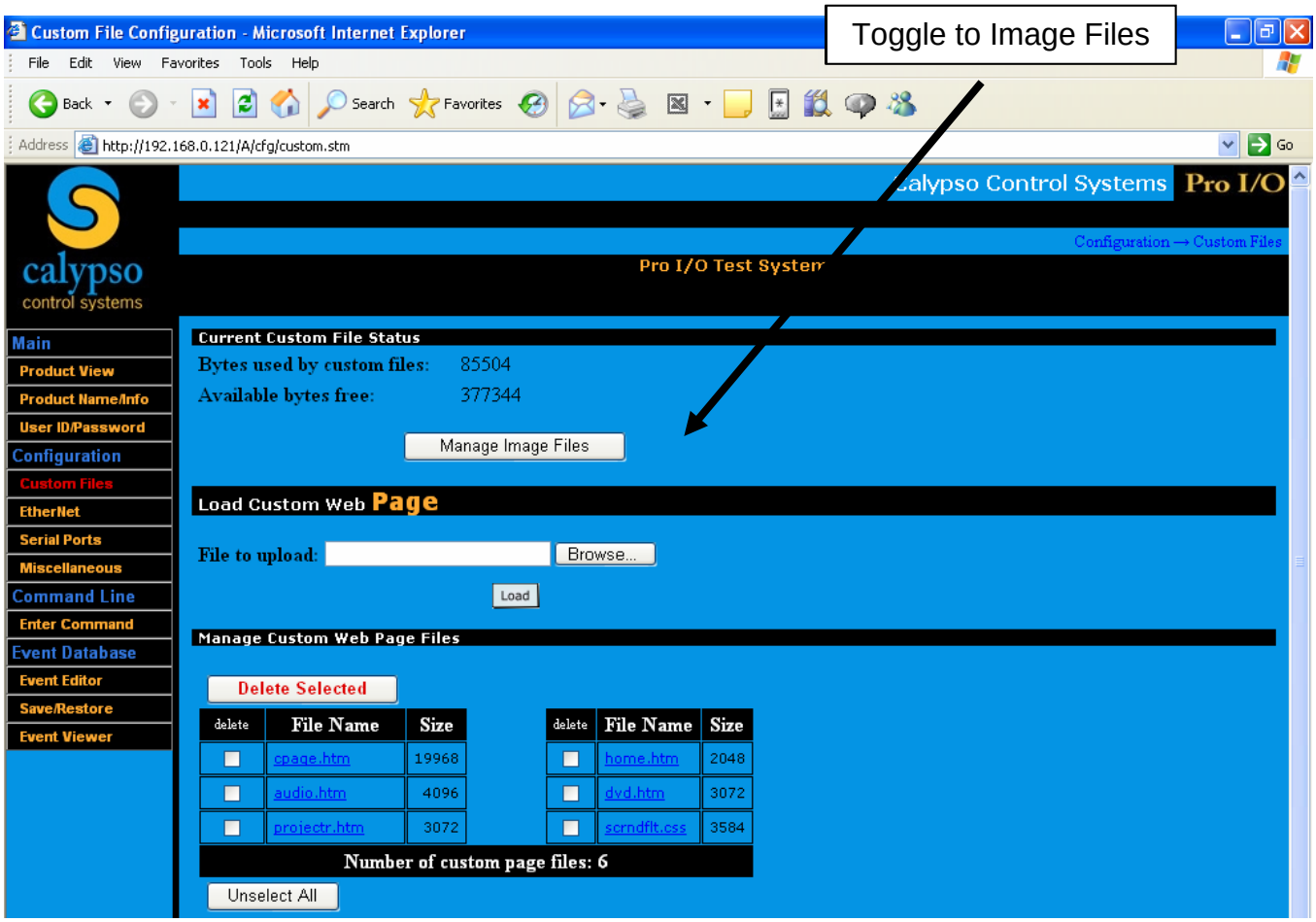


Figure 2.8 Event Database

2.3 USING THE CONTROLLER TO SERVE A CUSTOM WEB PAGE

All Calypso controllers are capable of directly serving user-created web pages to PC and PDA-based browsers. This feature is useful for creating simple, graphic control interfaces for those with password access. This approach offers simplicity and reliability while eliminating the need for a separate PC to serve as a web-server. For example, this feature allows a guest speaker to enter a room, call up the control URL on their laptop or PDA, enter a password and gain instant access to room controls, directly from their web browser.



2.3.1 Custom Files

Custom HTML pages and images are developed “off-line” and are then loaded onto the controller, limited to approximately 512KB worth of files. Control applications using as many as 25 pages have been implemented. The actual number of pages possible depends on the HTML layout and images used.

2.3.2 Custom File Loading and Deleting

The Custom File interface toggles between managing (e.g., loading, deleting) HTML page files and corresponding image files.

Note: the name of the custom page can be no longer than 8 characters, not including the suffix (e.g. “.htm”).

2.3.3 Custom Page Viewing

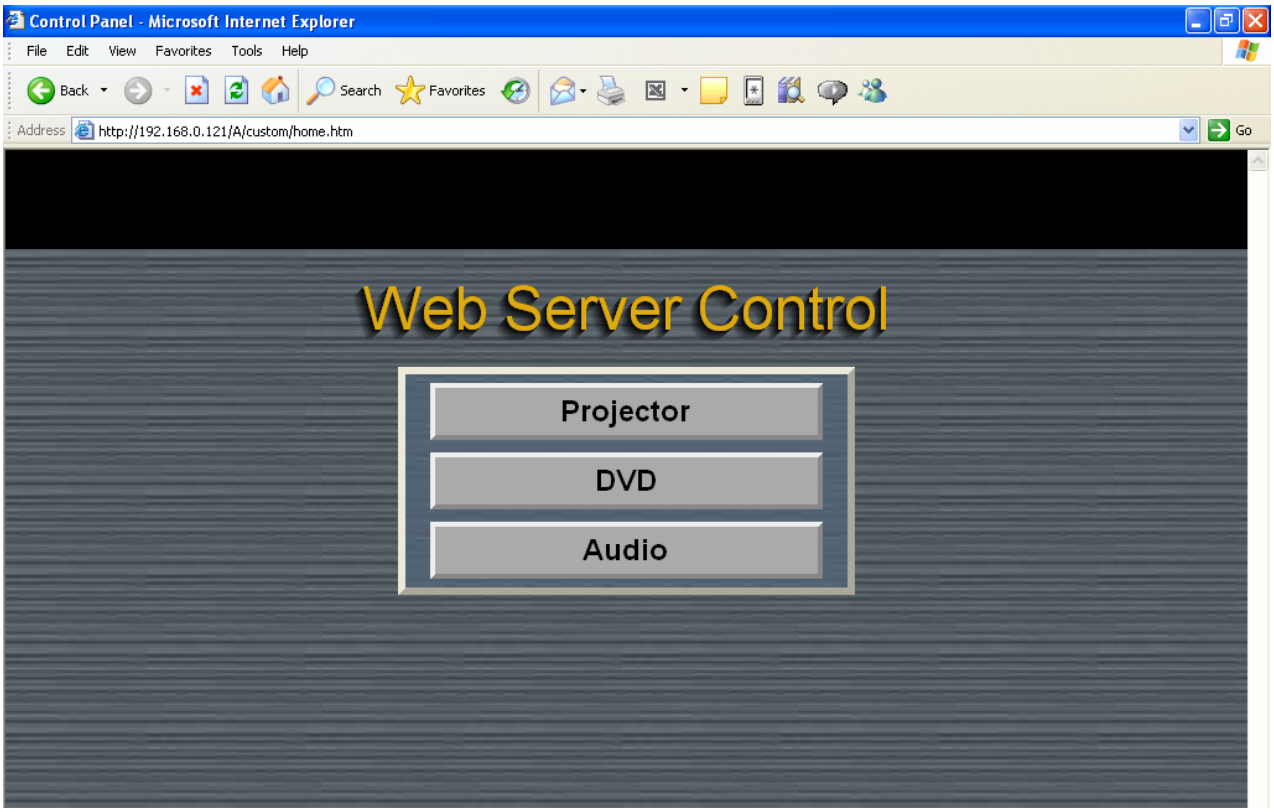
To view the default control web page stored on the controller, which gives some example methods for creating web pages for controlling devices, enter the controllers IP address followed by the extension “/A/custom/cpage.htm”

Example: http://192.168.1.101/A/custom/cpage.htm

Note that the custom page does not need to be named “cpage”, but does require the “htm” extension. If the custom page name is anything other than cpage.htm be sure to record the name of the page for later viewing through a web browser.

2.3.4 Custom Page Example

The figure below is a screenshot of the Home Page of the default web control pages available on the controller. It can be accessed at "/A/custom/home.htm" (for example "http://192.168.1.101/A/custom/home.htm"). These are actual working web pages that will trigger events associated with the buttons, if those events are defined in the database. Note that the event numbers may need to be changed to match the current database if these pages are to be used as a real control system." (there are no IP addresses encoded in the sample web pages).



Supported file and image types:

1. For custom pages, the following file types are supported:

.htm	standard HTML files
.css	external cascading style sheet files
.js	external javascript files
.jar	Java applet archive files
.swf	Flash application files

2. For custom images, the following file types are supported:

.gif	GIF image files
.jpg	JPEG image files
.png	PNG image files

Chapter 3: Command Protocols and Special Features

3.1 Event Database Command Summary - QUICK REFERENCE

Event database commands are used alternatively as “triggers”, used to launch an event and as “actions” in response to a valid trigger. The following summary provides complete details and examples for all event database triggers and actions.

Type	**Relay**
Description	Open and close relays
Command	#GPOx["y",Dz]; #GPOx["y",Dz];
Arguments	x = relay number y = OPEN or CLOSE D = optional delay duration before returning to prior state z = duration; unit = 4.4ms
Example Description	Open relay 1 and hold for .5 sec
Example	#GPO1["OPEN",D114];

Type	**Alarm**
Description	Turns the alarm on and on for the ION-LT and ION-LT2
Command	#ALM["y",Dz]; #ALM["y",Dz];
Arguments	y = BEEP or ON or OFF D = optional delay duration before returning to prior state z = duration; unit = 4.4ms
Example Description	Sound alarm for 1 second
Example	#ALM["ON",D227];

Type	**Event**
Description	Trigger event by number
Command	#EVTx[Wy];
Arguments	x = event number W = optional "wait" duration before next action y = duration in seconds
Example Description	Wait 5 seconds before executing event 15
	#EVT15[W5];

Type	**Serial Transmit**
Description	Transmit serial commands in ASCII and HEX
Command	#COMx[Ty,"insert string", Dz]
Arguments	x = com port number y = 1 for ASCII string y = 2 for HEX string D = delay z = duration; unit = 4.4ms
Example Description	Send HELLO with a carriage return through COM1 as ASCII followed by .5 sec delay #COM1[T1,"HELLO\r", D114];

Type	**IR Output**
Description	Transmit a CIRT command
Command	#XR0x[Fy,"z"];
Arguments	x = IR output port number y = 1 for CIRT y = 2 for Universal IR code z = 4-char IR code for CIRT or IR string for universal
Example Description	Transmit CIRT code out IR port 1 #XR01[F1,"08B9"];

Type	**Trigger Enable/Disable**
Description	Turn external event trigger on/off for a specific Event
Command	#TRG["x",y]
Arguments	x = ON or OFF y = event number
Example Description	Disable the trigger for event 36 #TRG["OFF",36];

Type	**Network Command**
Description	Transmit command to an IP address
Command	#NET[IP Address, Px,Ty,"command"];
Arguments	IP Address = target IP address x = network port number y = 1 for ASCII y = 2 for Hex command = actual command string
Example Description	Send a Hex string through Port 5000 to an IP address #NET[192.168.1.101,P5000,T2,"02001B2C03"];

Type	**Input Trigger**
Description	Detect change of input state
Command	#GPIx["y"];
Arguments	x = input number y = OPEN or CLOSE
Example Description	Trigger event when Input 1 is OPEN #GPI1["OPEN"];

Type	**Serial Compare Trigger**
Description	Compare incoming string to a template string
Command	#CMP1[Cx,"string"];
Arguments	x = 1 for ASCII string x = 2 for Hex string
Example Description	Trigger event when incoming COM1 string is HELLO #CMP1[C1,"HELLO"];

Type	**IR Input Trigger**
Description	Use incoming CIRT code as a trigger
Command	#XRIy["F1,"CIRT code"];
Arguments	y = IR input port number CIRT code = 4-char representation of 12-bit Sony IR command
Example Description	Trigger event when the CIRT code 08B9 is received from IR input 1 #XRI1[F1,"08B9"];

Type	**Network Compare Trigger**
Description	Compare incoming string to a template string
Command	#NCM[Cx,"string"];
Arguments	x = 1 for ASCII string x = 2 for Hex string
Example Description	Trigger event when the network string is HELLO #NCM[C1,"HELLO"];

Type	**Weekly Timer**
Description	Use the internal clock to trigger events on a weekly basis
Command	#TMR[Wx:y:z];
Arguments	x = 1 for Sunday x = 2 for Monday x = 3 for Tuesday x = 4 for Wednsday x = 5 for Thursday x = 6 for Friday x = 7 for Saturday y = hour (00 - 23) z = minutes (00 - 59)
Example Description	Trigger event every Monday at 8:00 am #TMR[W1:08:00];

Type	**Daily Timer**
Description	Use the internal clock to trigger events on a daily basis
Command	#TMR[Dx:y];
Arguments	x = hour (00 - 23) y = minutes (00 - 59)
Example Description	Trigger event daily at 8:00 am #TMR[D08:00];

Type	**Hourly Timer**
Description	Use the internal clock to trigger events on an hourly basis
Command	#TMR[Hx];
Arguments	x = minutes past the hour
Example Description	Trigger event hourly at 20 minutes past the hour #TMR[H20];

Type	**Interval Timer**
Description	Use the internal clock to trigger events every x minutes
Command	#TMR[Mx];
Arguments	x = minutes
Example Description	Trigger event every 7 minutes #TMR[M7];

3.2 CONTROL LANGUAGE SYNTAX

A control language is used to define event triggers (trigger context), event actions (action context), and commands delivered over the network or commands entered from a controllers “Enter Command” web page (command context). Not all commands can be used in every context, as noted below. Commands are not case sensitive and there are no spaces or leading zeros.

Calypso Action Control Language (CACL)	Version: **4.2** 04 Apr 2006

Formatted strings:
Special characters, which may be used in the "string" portion of COM, CMP, NET, and NCM commands below.

Command		Description
ccc..\ esc..ccc		non-hex mode formatted string (character sequence)
Escape sequences:		
\a	insert character 07 (the bell character) into the string character sequence.	
\b	insert character 08 (the backspace character) into the string character sequence.	
\f	insert character 0C (the form-feed character) into the string character sequence.	
\n	insert character 0A (the newline character) into the string character sequence.	
\r	insert character 0D (the carriage return character) into the string character sequence.	
\vn	insert a base-10 string representation of stored value n into the string character sequence; this escape sequence can result in multiple characters being inserted.	
\vxn	insert a base-16 string representation of stored value n into the string character sequence; this escape sequence results in 4 characters being inserted.	
\xnn	insert character 0xnn into the string character sequence (allows for a single hexadecimal-value character to be inserted).	
\xvn.0	insert the low-order byte of stored value n into the string character sequence.	
\xvn.1	insert the high	
\c	insert the character c into the string character sequence; e.g. "\\"; this is the result if c is not one of the characters listed above.	

GPI (inputs):

Command		Description
#GPIn**[***F*n**];**		command context only
#GPIn**[**string**];**		trigger context only
n	a decimal value between 1 and maxGPI, which targets a specific input connection.	
string	a sequence of printable characters delimited by double quotes; the sequence should be either "OPEN" or "CLOSE" (case insensitive).	
Fn	flag setting; n indicates the error return mode.	

GPO (relays):

Command		Description
#GPOn**[***F*n**];**		command context only
#GPOn**[**string, *D*n**];**		action or command context
n	a decimal value between 1 and maxGPO, which targets a specific relay connection.	
string	a sequence of printable characters delimited by double quotes; the sequence should be either "OPEN" or "CLOSE" (case insensitive).	
Dn	duration of pulse; causes the relay to temporarily change to the given state for a period of n units of time, and then change to the opposite state.	
Fn	flag setting; n indicates the error return mode.	

IR Input:

Command		Description
#XRIn[*Fn***];**		command context only
#XRIn[*F1*, string**];**		trigger context only
n	a decimal value between 1 and maxIRin, which targets a specific IR input line (most products have only a single IR input line with multiple connection points).	
string	a sequence of printable characters delimited by quotes; either single or double quotes may be used, as long as they are matched. The sequence must be a valid CIRT code (two hexadecimal pairs, case insensitive).	
Fn	flag for type; n indicates the type of IR data returned or expected: 1 = CIRT data, which is the default and is the only setting allowed for triggers 2 = Universal data (grab first or only burst) 4 = Noise data 6 = Universal data, skip first burst (grab second burst) 7 = Universal data, skip first 2 bursts (grab third burst) Only the F1 setting may be used for triggers, and may be omitted entirely since it is the default. The other settings may be used in learn command context.	

IR Outputs:

Command		Description
#XROn[*flags*, string**];**		action or command context
n	a decimal value between 1 and maxIRout, which targets a specific IR output line.	
string	a sequence of printable characters delimited by quotes; either single or double quotes may be used, as long as they are matched. The sequence may be either a valid CIRT code or a valid Universal code including the Calypso header data. (Must be pairs of hexadecimal characters, case insensitive.)	
Flags:		
Bn	break period; specifies an inter-command idle period of n units of time.	
Fn	flag for type; n indicates the type of IR data returned or expected: 1 = CIRT data, which is the default 2 = Universal data No other flag values are valid for output.	
Rn	repeat count; specifies the number of times to repeat the command (with a break between each repeat, as specified above).	

Serial Commands:

Command		Description
#COMn[*Wn***];**		command context only
#COMn[*flags*, string**];**		action or command context
n	a decimal value between 1 and maxCOM, which targets a specific serial port connection.	
string	a sequence of printable characters delimited by quotes; either single or double quotes may be used, as long as they are matched. The sequence may include special characters by using certain escape (backslash) sequences; or the entire string may be encoded as hexadecimal character pairs (ref hex flag below).	
Flags:		
Dn	delay period; specifies a delay after transmission for a period of n units of time.	
Fn	flag setting; n indicates the error return mode [IGNORED].	
M1	master mode; causes ALL serial ports to transmit the string.	
Tn	type of data string; indicates the type of data returned and/or expected: 1 = standard string data, the default 2 = hexadecimal encoded data	
Wn	wait period; specifies a wait before grabbing the receive buffer for a period of n seconds. Note that this will cause the command to return "received" data, even when also transmitting data, and therefore should NOT be used in action context.	

Serial Compare:

Command		Description
#CMPn**[***flags*, string**];**		trigger context only
n	a decimal value which targets a specific serial port connection; currently only serial port 1 may be used for this command.	
string	a sequence of printable characters delimited by quotes; either single or double quotes may be used, as long as they are matched. The sequence may include special characters by using certain escape (backslash) sequences; or the entire string may be encoded as hexadecimal character pairs (ref hex flag below).	
Flags:		
Cn	type of match string; indicates the type of data expected: 1 = standard string data, the default 2 = hexadecimal encoded data	
Ln	match length expected; a non-zero value means the number of bytes remaining in the receive buffer starting from the match position must be n; a value of zero means this test will not be performed (length doesn't matter). A non-zero value must always be greater than or equal to the length of the match string.	
Mn	match position; a non-zero value means start the match at 1-relative position n in the receive buffer; a value of zero means the match will succeed if the match string is found anywhere in the receive buffer.	

Event Commands:

Command		Description
#EVTn**[***flags***];**		action or command context
n	a decimal value between 0 and maxEvent, which targets a specific database event.	
Flags:		
Fx	flag setting; a pair of hexadecimal characters which specify a set of bit flag values, all OR-ed together: 01 = run action 1 02 = run action 2 04 = run action 3 08 = run action 4 10 = obey the enable state of the event (don't run it if it is disabled) The resulting flag value may NOT be zero. By default this command will ignore the enable state of the event. The flag value F0F is the default setting and means run all actions.	
Wn	wait period; specifies a wait of n seconds before invoking the event.	

Net Commands:

Command		Description
#NET[*flags*, *Aaddr* or *Iaddr*, string**];**		action or command context
string	a sequence of printable characters delimited by quotes; either single or double quotes may be used, as long as they are matched. The sequence may include special characters by using certain escape (backslash) sequences; or the entire string may be encoded as hexadecimal character pairs (ref hex flag below).	
Aaddr	address string (no quotes); specifies a DNS-style named address, e.g. "ion.calypsocontrol.com". One, and only one, of the Aaddr or Iaddr arguments must be given.	
Iaddr	IP address string (no quotes); specifies an IPv4-style numeric address, e.g. "192.168.1.101". One, and only one, of the Iaddr or Aaddr arguments must be given.	
Flags:		
Fn	flag setting; specifies the output mode: 1 = use TCP; the default 2 = use UDP (not yet implemented)	
Pn	port number; specifies a target port number to be used for the network communication (uses the currently configured Remote Port by default).	
Tn	type of data string; indicates the type of data expected: 1 = standard string data, the default 2 = hexadecimal encoded data	

Net Compares:

Command		Description
#NCM[*flags*, string**];**		trigger context only
string	a sequence of printable characters delimited by quotes; either single or double quotes may be used, as long as they are matched. The sequence may include special characters by using certain escape (backslash) sequences; or the entire string may be encoded as hexadecimal character pairs (ref hex flag below).	
Flags:		
Cn	type of match string; indicates the type of data expected: 1 = standard string data, the default 2 = hexadecimal encoded data	
Ln	match length expected; a non-zero value means the number of bytes remaining in the receive buffer starting from the match position must be n; a value of zero means this test will not be performed (length doesn't matter). A non-zero value must always be greater than or equal to the length of the match string.	
Mn	match position; a non-zero value means start the match at 1-relative position n in the receive buffer; a value of zero means the match will succeed if the match string is found anywhere in the receive buffer.	
Note that since the network receive buffer is filled from the same port as that on which CNAP commands are received, strings longer than 6 characters which also begin with '#' will not be seen as potential NCM strings. Instead, such strings will be treated as potential CNAP commands and processed accordingly.		

Value Commands:

Command		Description
#VALn**[***F2***];**		command context only
#VALn**[***F1, func***value];**		action or command context
n	a decimal value between 1 and maxVAL, which targets a specific stored value.	
value	a numeric value; if the value string begins with "0x" the rest of the string is interpreted as a hexadecimal number. The maximum value a stored value can hold is 0xFFFF (a 16-bit unsigned value).	
Fn	flag setting; specifies various control options: 1 = allow function to cause a rollover, e.g. 0xFFFF + 1 = 0; default is no rollover, e.g. 0xFFFF + 1 = 0xFFFF 2 = retun current value in hexadecimal format; default is to return the value in decimal format	
Functions		
+	add the value to the current stored value.	
-	subtract the value from the current stored value.	
none	if no function is specified, set the stored value to the given value.	

Test Value Commands:

Command		Description
#TSTn**[***func***value];**		trigger context only
#TSTn**[**value-value**];**		trigger context only
n	a decimal value between 1 and maxVAL, which targets a specific stored value.	
value	a numeric value; if the value string begins with "0x" the rest of the string is interpreted as a hexadecimal number. The maximum value a stored value can hold is 0xFFFF (a 16-bit unsigned value).	
Functions		
=	true if the current stored value is EQUAL to the given value.	
!	true if the current stored value is NOT EQUAL to the given value.	
<	true if the current stored value is LESS THAN the given value.	
>	true if the current stored value is GREATER THAN the given value.	
-	true if the current stored value is BETWEEN the 2 given values, inclusive.	

Timer Triggers:

Command		Description
#TMR[Wweek**];**		trigger context only
#TMR[Dday**];**		trigger context only
#TMR[Hhour**];**		trigger context only
#TMR[Mmodulus, *O*offset**];**		trigger context only
week	a numeric time value; specifies a specific time of the week in day:hour:min form, e.g. 6:13:45 means every Friday at 1:45 pm, while 0:0:0 means every Sunday at midnight.	
day	a numeric time value; specifies a specific time of the day in hour:min form, e.g. 18:30 means every day at 6:30 pm.	
hour	a numeric time value; specifies a specific time in the hour in minutes, e.g. 15 means every hour at a quarter after the hour.	
modulus	a numeric time value; specifies a cycle period in minutes, e.g. 60 means every hour, 1440 mean every day, 10 means every 10 minutes.	
offset	a numeric time value; specifies an offset in minutes from the top of the hour but within the modulus, e.g. offset = 10 with modulus = 60 means 10 minutes after every hour. By default the offset is 0.	
Note that the W, D, and H formats are easy to use shorthands for certain modulus + offset pairs. In particular, the character specifies the modulus and the value is converted into an offset in minutes. So W2:0:10 (every Tues at 10 after midnight) is the same as M10080, O2890.		

Alarm Commands (ION-LT and ION-LT2 only):

Command		Description
#ALM[];		command context only
#ALM[string, *D*n, *S*n**];**		action or command context
string	a sequence of printable characters delimited by double quotes; the sequence should be one of "ON", "BEEP" or "OFF" (case insensitive).	
Dn	duration of alarm; causes the alarm to temporarily change to the given state for a period of n units of time, and then turn OFF.	
Sn	sound mode; specifies the mode to be used: 1 = higher frequency tone, the default 2 = lower frequency tone This setting is ignored for "OFF".	

Trigger Commands:

Command		Description
#TRG[state**,** list**];**		action or command context
state	a sequence of printable characters delimited by double quotes; the sequence should be either "ON" or "OFF" (case insensitive). Must come first.	
list	a comma-separated list of event ranges; each range can be either a single event number, or a dash-separated range. For example, "3, 5, 20-25, 31, 36-42".	
Temporarily changes the "enable" state of the specified events to the given state. On reset or power-cycle, the state of the events will return to that saved in the database.		

Control Commands:

Command		Description
#CTL[string**];**		non-URL command context only
string	a sequence of printable characters delimited by quotes; either single or double quotes may be used, as long as they are matched. Case sensitive. Must be one of: "CLR" = clear the command queue (often used to terminate in infinite loop); "SOFTRESET" = cause the ZWorld code to reset itself.	
This is a special purpose command and should be used with care.		

3.3 ASCII to Hex Conversion Table

Dec	Hex	Oct	Char	Description
0	0	000		null
1	1	001		start of heading
2	2	002		start of text
3	3	003		end of text
4	4	004		end of transmission
5	5	005		enquiry
6	6	006		acknowledge
7	7	007		bell
8	8	010		backspace
9	9	011		horizontal tab
10	A	012		new line
11	B	013		vertical tab
12	C	014		new page
13	D	015		carriage return
14	E	016		shift out
15	F	017		shift in
16	10	020		data link escape
17	11	021		device control 1
18	12	022		device control 2
19	13	023		device control 3
20	14	024		device control 4
21	15	025		negative acknowledge
22	16	026		synchronous idle
23	17	027		end of trans. block
24	18	030		cancel
25	19	031		end of medium
26	1A	032		substitute
27	1B	033		escape
28	1C	034		file separator
29	1D	035		group separator
30	1E	036		record separator
31	1F	037		unit separator
32	20	040		space
33	21	041	!	
34	22	042	"	
35	23	043	#	
36	24	044	$	

Dec	Hex	Oct	Char
64	40	100	@
65	41	101	A
66	42	102	B
67	43	103	C
68	44	104	D
69	45	105	E
70	46	106	F
71	47	107	G
72	48	110	H
73	49	111	I
74	4A	112	J
75	4B	113	K
76	4C	114	L
77	4D	115	M
78	4E	116	N
79	4F	117	O
80	50	120	P
81	51	121	Q
82	52	122	R
83	53	123	S
84	54	124	T
85	55	125	U
86	56	126	V
87	57	127	W
88	58	130	X
89	59	131	Y
90	5A	132	Z
91	5B	133	[
92	5C	134	\
93	5D	135	]
94	5E	136	^
95	5F	137	_
96	60	140	`
97	61	141	a
98	62	142	b
99	63	143	c
100	64	144	d

37	25	045	%	
38	26	046	&	
39	27	047	'	
40	28	050	(	
41	29	051	)	
42	2A	052	*	
43	2B	053	+	
44	2C	054	,	
45	2D	055	-	
46	2E	056	.	
47	2F	057	/	
48	30	060	0	
49	31	061	1	
50	32	062	2	
51	33	063	3	
52	34	064	4	
53	35	065	5	
54	36	066	6	
55	37	067	7	
56	38	070	8	
57	39	071	9	
58	3A	072	:	
59	3B	073	;	
60	3C	074	<	
61	3D	075	=	
62	3E	076	>	
63	3F	077	?	

101	65	145	e
102	66	146	f
103	67	147	g
104	68	150	h
105	69	151	i
106	6A	152	j
107	6B	153	k
108	6C	154	l
109	6D	155	m
110	6E	156	n
111	6F	157	o
112	70	160	p
113	71	161	q
114	72	162	r
115	73	163	s
116	74	164	t
117	75	165	u
118	76	166	v
119	77	167	w
120	78	170	x
121	79	171	y
122	7A	172	z
123	7B	173	{
124	7C	174	|
125	7D	175	}
126	7E	176	~
127	7F	177	DEL

Chapter 4: Troubleshooting

The table given below provides corrective action for possible troubleshooting situations. In case, you require further assistance, please contact a Calypso Control Systems service representative.

PROBLEM	SOLUTION
Controller does not power up, no lights	Check the DC power connection to the controller and make sure that the external power supply is properly plugged into the wall outlet or power strip. If the blue power LED still does not illuminate, call Calypso tech support.
Unable to launch the controller web browser directly from my computer	When connecting directly to the controller the host computer's IP address must be configured to match the IP range of the controller. The controller default IP address is 192.168.1.101. After connecting to the controller the IP address can be changed. A crossover Ethernet cable is required for direct connection.
Unable to launch the controller web browser over the network	When connecting to the controller over a network you will need to use a straight through Ethernet cable. The host computer's IP address must be configured to match the range of the controller default IP address when the first connection is made. The controller default IP address is 192.168.1.101. The network administrator must allow for connection from the host computer to the controller default IP address

Password has been lost or IP address needs to be reset:	The controller will need to be reset back to its factory default. Reset back to factory default by following these steps: All controllers • Power down the controller • Short the RX and TX of com 1 • Power up the controller • Remove the RX and TX short before 4 seconds • The password and IP address will now be set back to factory default NOTE: when the controller is reset back to factory default, all personality configurations are reset. However, the event database is unaffected.
C_Link does not control a device	In order for the c_link feature to work there are 3 fundamental items that need to happen - 1) The exact Hyperlink syntax needs to be inserted on the control button. 2) There needs to be a valid event configured within the controller event database. 3) A valid Ethernet connection needs to be in place.
Cannot control a Serial device	Check that the proper Calypso protocol is adhered to. Check the com port settings of the controller, they must match the device to be controlled. Check the com port pin outs. It is very common for RS232 connections to require that pins 2 and 3 be swapped (null modem) between devices.
Cannot control an IR device	IR outputs are “steered” to the proper IR ports, check the IR port address of your command. Make sure the proper Calypso command syntax is adhered to. Check to see that the IR emitter cable “winks” when the IR command is imitated. Make sure that the IR command has been learned, copied and pasted properly.
Cannot control a Relay device	Make sure the proper Calypso command syntax is adhered to.

Input IR does not control a device	Makes sure that the IR input “event trigger” codes are Sony 12 bit codes. The controller learning feature will show the bit length of IR codes.
Input contact closure does not control a device	Make sure the proper Calypso command syntax is adhered to for input triggering. A valid event must be defined.
Controller has locked up	Power down and then power back up. Wait a few minutes for the controller to complete the power up cycle.

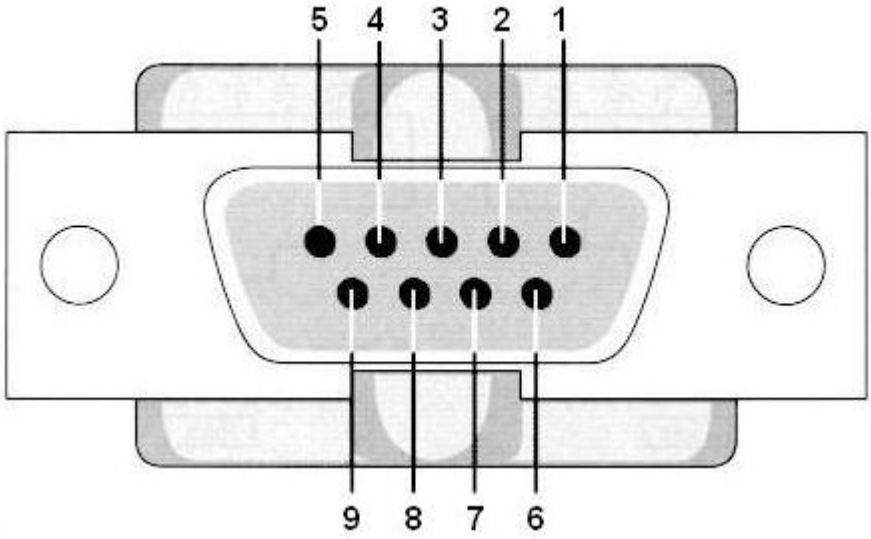
Chapter 5: Specifications

Specifications - ProIO / ION / ION_LT series

(specifications apply to all CCS Ethernet based controllers)

Front Panel	Note: Devices may have common or individual LED's
Blue LED	Power Indicator
Green LED's	Communication Activity
Yellow LED's	Input Status
Amber LED's	Output Status (Relays)
Red LED's	IR Status (Learn or Output)
Back Panel Connectors	
Power Jacks are 2.5mm	12VDC
Ethernet RJ45	10BaseT
RJ11	Inactive
1/8" Stereo Jacks IR or 3.81mm block	IR inputs (T/S common) or IR output (T/R/S common)
Serial ports - DB9F or 3.81mm block	RS-232C or RS422 (ProIO only)
3.81mm block removable (2,3,4,5 Pin)	GPI (contact closure sensing to ground inputs) GPO (relay outputs, N.O. & FORM C(ION_LTx only)) IR (ION series) Serial (ION series, RS232 only) Power out (12vdc @ 500ma max) except ION_LT series
Specifications	
Power IN	12VDC @ 1amp, 12Watts (Standard) ProIO and ION series ProIO and IONe (do not source power) ION_4S, ION_8R, ION_16I (source power 12vdc @ 500ma) 12VDC @ 500ma, 6Watts (Standard) ION_LTx series ION_LT, ION_LT2 (do not source power)
Operating Ambient Temperature	Range 5 C (41 F) to 45 C (113 F)
Operating RH Range	10 - 90% (non-condensing)
Relays	SPST 5A @ 24VDC, SPDT 5A @ 24VDC (ION_LTx only)
Dimensions	ProIO, IONe, ION_4S, ION_8R, ION_16I use 1/2 rack standard H 1.5 in (38mm) x W 7.15 in (182mm) x L 5 in (127mm) ION_LT and ION_LT2 wall/ceiling mount brackets
Material	Aluminum
Weight	ProIO and ION series, Rack Case Style 1lb 8oz ION_LTx series, Compact Case 9oz
Certifications	FCC/CE

Com Port Connector: Pro I/O and ION-e DB 9 sub miniature F type



1	RX-	(RS422)
2	RX	(RS232)
3	TX	(RS232)
4	TX+	(RS422)
5	GND	
6	RX+	(RS422)
7	NC	
8	NC	
9	TX-	(RS422)

Calypso Control Systems 3-year Product Warranty Statement

This Non-Transferable warranty is provided to original purchasing end user, herein referred as "customer", of Calypso Control Systems product line defined as: Pro I/O, ION-e, ION-LT1, ION-LT2, UT-500, ION-4s, ION-8r, ION-16i, CB-1000, CR-1200R, CA-500, CA-1000, and CA-1050, herein referred as "product".

This warranty is applicable to product sold or distributed to customer by an authorized Calypso Control Systems Dealer, OEM, Value Added Reseller or sold directly to the end user by Calypso Control Systems, LLC. This warranty becomes effective from the moment the end user completes purchase and receives product. This warranty shall remain in effect for 3 years from the moment of purchase as long as the original customer of the product continues to own and use the product. This warranty does not apply to accessories such as power supplies and cables, which carry standard 12-month manufacturer warranties.

Terms

Calypso Control Systems warrants that product shall be materially free of defects in material and workmanship under normal use and service during the warranty period. In the event that Calypso Control Systems receives notice from the customer during the warranty period that product does not conform to this warranty, Calypso Control Systems shall, at its sole option, either repair or replace the non-conforming product. The warranty on the replacement or repaired product shall continue for the duration of the original warranty. All returned product becomes the property of Calypso Control Systems.

Procedures

A product may only be returned with the prior written approval of Calypso Control Systems. Such approval shall reference a Return Material Authorization number (RMA) issued by authorized Calypso Control Systems technical support personnel. Transportation costs, if any, incurred in connection with the return of a defective item to Calypso Control Systems shall be borne by the Customer. Transportation costs incurred in connection with the re-delivery of a repaired or replaced item to the Customer shall be borne by Calypso Control Systems. However, such costs shall be borne by the Customer if Calypso Control Systems, reasonably determines that the product is not defective. If Calypso Control Systems determines, in its sole discretion, that the allegedly defective product is not covered by the terms of the warranty provided hereunder, or that a warranty claim is made after the warranty period, the cost of repair by Calypso Control Systems, including all shipping expenses, shall be reimbursed by the Customer. Calypso Control Systems shall have no liability with respect to data contained in any system returned to Calypso Control Systems.

Exclusions

The foregoing warranties and remedies are for the Customer's exclusive benefit and are non-transferable. Any and all warranties shall be void regarding System components that are damaged or rendered unserviceable by: (1) acts or omissions of non-Calypso Control Systems personnel; (2) misuse, theft, vandalism, fire, water, or other peril; (3) alterations of or additions to the System or any element thereof performed by personnel not certified by Calypso Control Systems to perform such alterations and additions or (4) the Customer's failure to operate the product in conformance with Calypso Control Systems published operating parameters, including environmental specifications.

Disclaimer of Warranty

TO THE EXTENT ALLOWED BY APPLICABLE LAW, THE LIMITED WARRANTIES REFERRED TO IN THE PARAGRAPHS ABOVE SHALL BE IN LIEU OF ALL OTHER WARRANTIES WHETHER EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE. CALYPSO CONTROL SYSTEMS, LLC SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Limitation of Liability

TO THE EXTENT ALLOWED BY APPLICABLE LAW, CALYPSO CONTROL SYSTEMS, LLC AND ITS SUPPLIERS EXCLUDE THEMSELVES FROM ANY LIABILITY FOR ANY LOST REVENUE OR PROFIT, LOSS OF BUSINESS, LOSS OF INFORMATION OR DATA, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL, OR PUNITIVE DAMAGES OF ANY KIND CAUSED OUT OF OR IN CONNECTION WITH THE SALE, INSTALLATION, MAINTENANCE, USE, PERFORMANCE, FAILURE, OR INTERRUPTION OF ITS PRODUCTS, EVEN IF CALYPSO CONTROL SYSTEMS, LLC AND ITS AUTHORIZED RESELLERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT SHALL CALYPSO CONTROL SYSTEMS, LLC OR ITS SUPPLIER'S TOTAL LIABILITY TO THE CUSTOMER, WHETHER IN CONTRACT NEGLIGENCE, STRICT LIABILITY, TORT OR OTHERWISE, EXCEED THE PRICE PAID BY THE CUSTOMER. THE FOREGOING LIMITATIONS SHALL APPLY EVEN IF ANY REMEDY PROVIDED HEREIN SHALL FAIL ITS ESSENTIAL PURPOSE. THIS LIMITATION OF LIABILITY, HOWEVER, WILL NOT APPLY TO ANY CLAIMS FOR PERSONAL INJURY.