



CONTROL TECHNOLOGY CORPORATION

---

WebMON 2.0 User's Guide

# WebMON 2.0 User's Guide

*Blank*



**WARNING:** Use of CTC Controllers and software is to be done only by experienced and qualified personnel who are responsible for the application and use of control equipment like the CTC controllers. These individuals must satisfy themselves that all necessary steps have been taken to assure that each application and use meets all performance and safety requirements, including any applicable laws, regulations, codes and/or standards. The information in this document is given as a general guide and all examples are for illustrative purposes only and are not intended for use in the actual application of CTC product. CTC products are not designed, sold, or marketed for use in any particular application or installation; this responsibility resides solely with the user. CTC does not assume any responsibility or liability, intellectual or otherwise for the use of CTC products.

The information in this document is subject to change without notice. The software described in this document is provided under license agreement and may be used and copied only in accordance with the terms of the license agreement. The information, drawings, and illustrations contained herein are the property of Control Technology Corporation. No part of this manual may be reproduced or distributed by any means, electronic or mechanical, for any purpose other than the purchaser's personal use, without the express written consent of Control Technology Corporation.

The information in this document is current as of the following Hardware and Firmware revision levels. Some features may not be supported in earlier revisions. See [www.ctc-control.com](http://www.ctc-control.com) for the availability of firmware updates or contact CTC Technical Support.

<b>Model Number</b>	<b>Hardware Revision</b>	<b>Firmware Revision</b>
5200	All Revisions	> 5.01

## TABLE OF CONTENTS

Introduction.....	7
Remote Administration Features .....	7
Requirements .....	9
Java Run-Time .....	9
Obtaining Run-Time .....	9
Run-Time Installation .....	10
Invoking WebMON .....	17
HTTP Web Interface.....	17
Setup .....	21
Ethernet Tab.....	24
Current Ethernet Settings .....	24
SNTP Time Server Settings .....	27
POP3 Mail Inbox Server Settings.....	28
Serial Tab.....	31
COMM 1/2 Configuration .....	31
Summary Tab.....	33
General Controller Information .....	33
Firmware Revisions .....	33
Modules and Expansion Units .....	34
Thermocouple Tables.....	34
Disks Tab .....	36
Un-Mounting and Formatting Drives .....	37
Mounting New Drives.....	38
Example – Changing the root drive .....	39
Example – Adding a drive to root.....	42
Run Programs Tab .....	46
Quickstep Programs (.dso).....	46
Scripts (.ini).....	47
User Programs (.sr).....	47
RTC Tab.....	49
Time Synchronization to PC.....	49
Email Notification Tab .....	50
Tree View, Local/Controller.....	50
Creating/Editing New Email Template.....	51
Deleting Email Template .....	53
Authentication Tab.....	53
Changing Password.....	54
Group Name.....	54
Security Tab .....	56
System Tab.....	58
Memory Utilization (Heap).....	58
Restoring Factory Defaults .....	58

## WebMON 2.0 User's Guide

Resetting Controller (Hard Reset) .....	59
Threads.....	59
Monitor – Data Table.....	61
Monitor - Resources.....	65
Resource Selection.....	66
Configurations.....	67
I/O .....	70
Flags .....	75
Registers.....	76
Symbols.....	79
Resource Presentation/Editing .....	80
Delete/Cut/Insert .....	80
Printing.....	81
Resource Monitoring .....	81
Monitor - Tasks.....	83
Task Window .....	84
Command Result Window .....	86
Source Window.....	87
Resource Watch Window .....	89
Breakpoints Table .....	91
Menu .....	93
File Transfer (FTP) .....	95
Receive File .....	97
Send File .....	98
Update Flash .....	98
Discovery (Network) .....	101
Tunneling .....	102

*Blank*

## Introduction



This document introduces the remote administrative and diagnostic capabilities of WebMON. WebMON combines numerous, powerful features, which may be used to configure, monitor, and provide advanced diagnostics for Control Technology Controllers (presently only the 5200, and future controllers). Existing as a java applet, WebMON may be served up from the controller's internal web server on any PC running Internet Explorer 6.0 and above and Sun's JRE V1.4\_03 or greater. A general overview of this functionality is provided within this manual.

### **Remote Administration Features**

WebMON provides a graphical interface to the script commands detailed in the "*Script Language Guide, 951-520003*". Most capabilities detailed within this manual are available from within scripts and also executable from within User Programs. The WebMON graphical user interface provides a simplified approach to controller administration and diagnostics. Some of the available features are:

- **Controller Setup**
  - Network - IP, SMTP mail client, and SNTP time client configuration.
  - Serial Port – baud rate, parity, data/stop bits, protocol, address...
  - Firmware Revisions
  - Module and Expansion Configuration
  - Thermocouple table information
  - Program Execution/Selection
  - Email Creation and Storage
  - IP Security
  - Password
  - Group Designation
  - Flash/RAM Disk mounting/un-mounting
  - Thread Utilization and Memory Allocation
  - Real Time Clock Initialization
- **Resource Monitor**
  - Register/IO Real Time Monitoring
  - Simultaneous Monitoring of multiple Controllers

- Register Naming and Symbols Supported
- Test Configurations may be saved locally or to controller for later recall
- Hard Copy printout of Monitor information
- Copy/Paste cells to Excel or other Window's environments
- Simplified I/O selection
- Read/Write data
- Decimal and Hexadecimal data representation
- ***Data Table Monitor***
  - Real-time data presentation
  - Page-able view windows of rows/columns
  - Read/Write Cells
  - Copy/Paste cells to Excel or other Window's environments
  - Decimal, ASCII, and Hexadecimal data representation
- ***FTP Communications***
  - Command line FTP available for file transfer
  - Send/Receive Files
  - Re-Flash controller firmware
  - Remotely View/Create/Manage mounted drives
- ***Task Monitor/Debugger***
  - Monitor/View Task hierarchy
  - Control Task Execution
  - Source Code debugger/Monitor
  - Resource View/Monitor Window
  - Breakpoints with Step and Register value qualifiers
  - Single step, continue execution
  - Debug multiple tasks at the same time.
  - Debug multiple controllers at the same time.
- ***Network Discovery***
  - Tree view of all online controllers via broadcasts
  - Menu selectable functions for all above features, on any online controller, from single WebMON instance.
  - Network Tunnel mode allow Internet access. Controller provides broadcasting service and forwards online information via UDP.



## Requirements



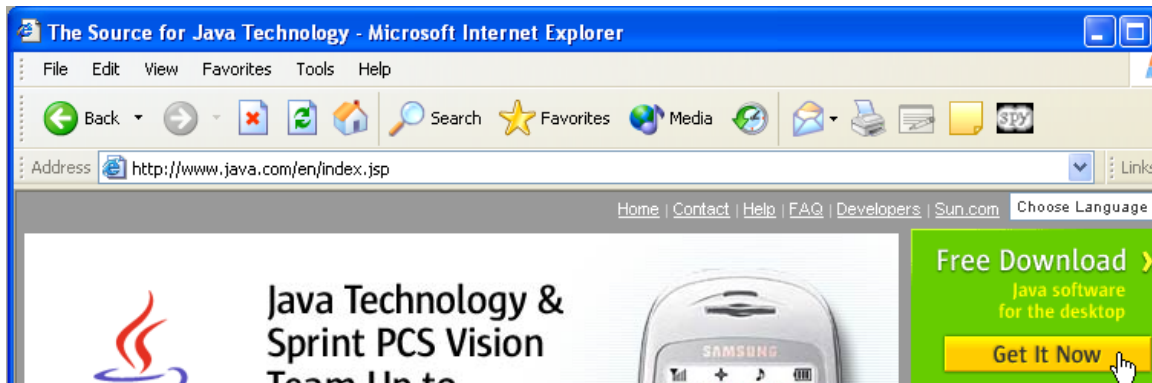
WebMON executes in a web browser. It has been designed for use with Microsoft's Internet Explorer 6.0 or above. Use in other Web Browsers is not supported. Proper execution requires some initial configuration of the host computer. The first, being the installation of Internet Explorer 6.0, available by using "Windows Updates". This may be found as a menu selection within the Internet Explorer Tools main menu. The second requirement is the installation of Sun's V1.04.03 Java Run-time, or greater. A complete installation package may be downloaded from Control Technologies Web site or directly from Sun ([www.java.com](http://www.java.com)).

### Java Run-Time

The following is detailed information for the retrieval and installation of the Sun JRE, Java Run-Time Environment. The installation shown is only valid for the revision current at the time this manual was written, 1.4.2\_03, and is subject to change.

### Obtaining Run-Time

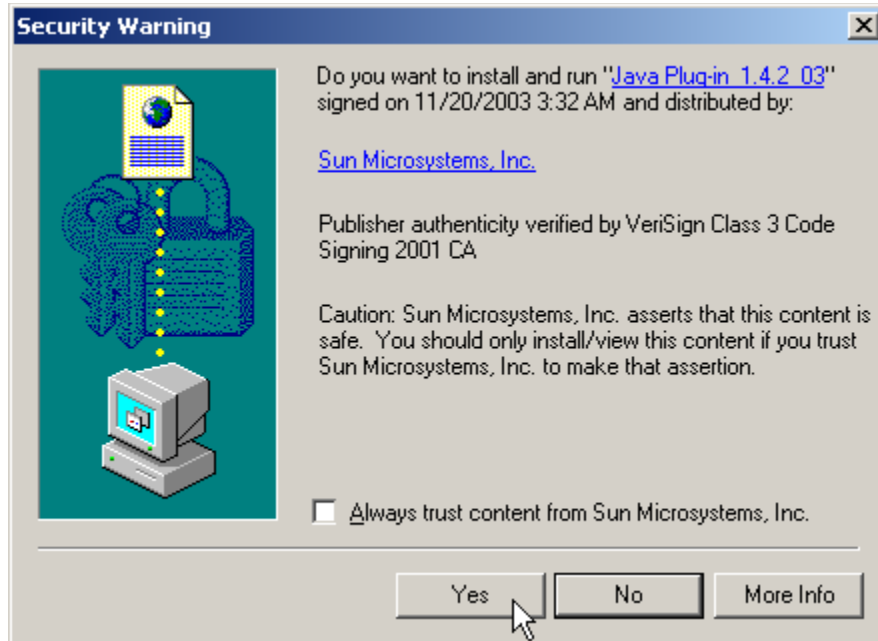
The run-time can be downloaded from Sun's Java site, [www.java.com](http://www.java.com) using Internet access. A screen similar to below will appear (subject to change) with a button available for downloading the Java Run-Time:



As shown above select the “Get It Now” button and then proceed to the Run-Time Installation section.

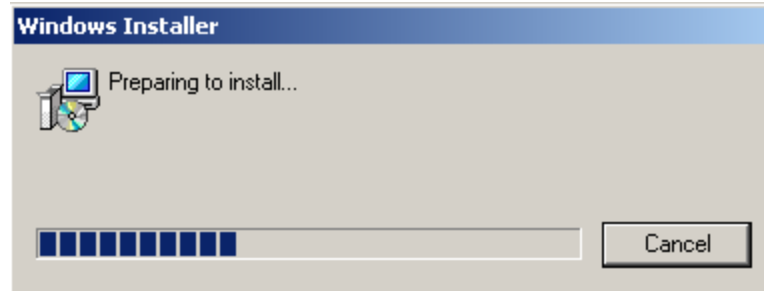
### Run-Time Installation

Upon selecting “Get It Now”, in the previous section, a Security Warning dialog will appear requesting permission to install software on your computer:

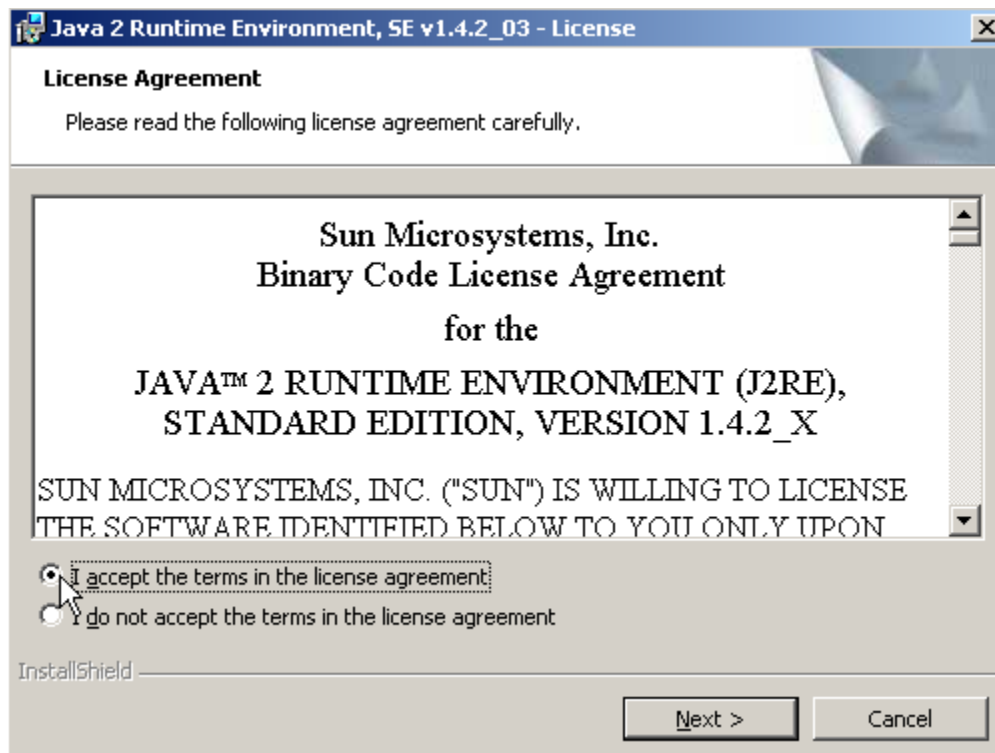


Click “Yes” and the following dialogs will appear, an initial loading screen for Java, then a Window’s installer screen and finally the License agreement:

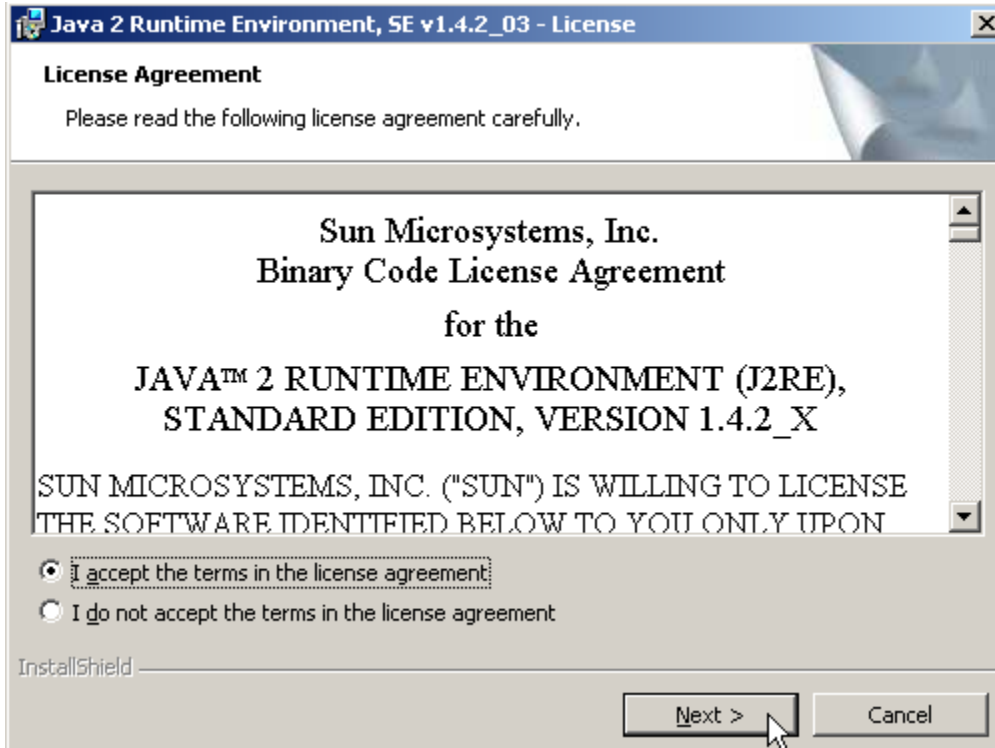




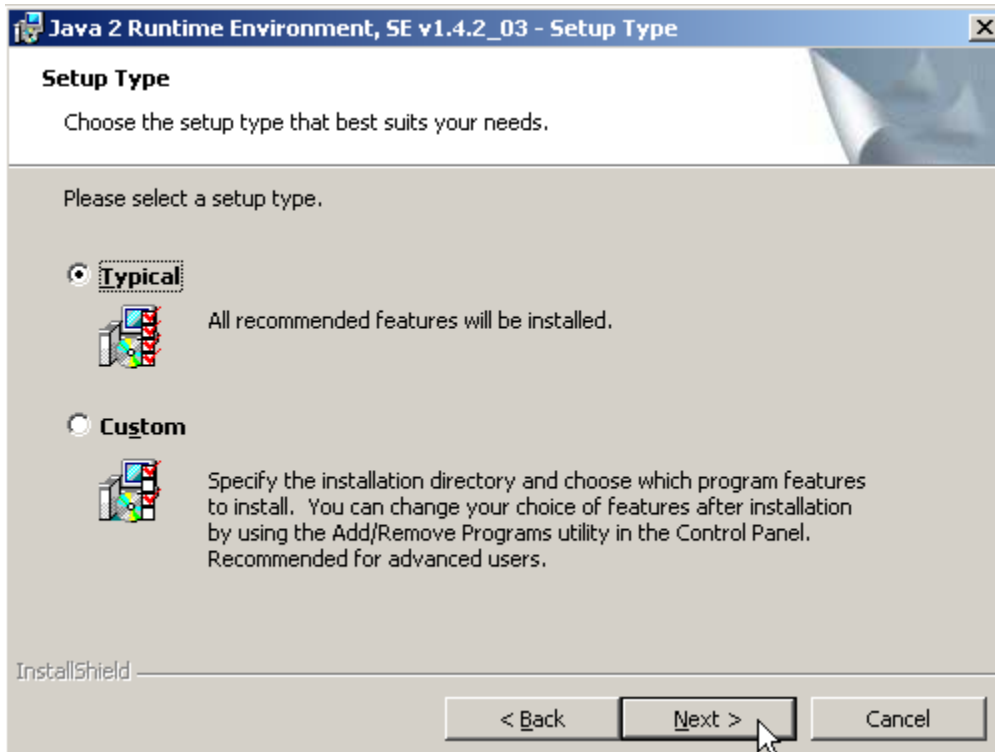
Select the "I accept the terms in the license agreement":



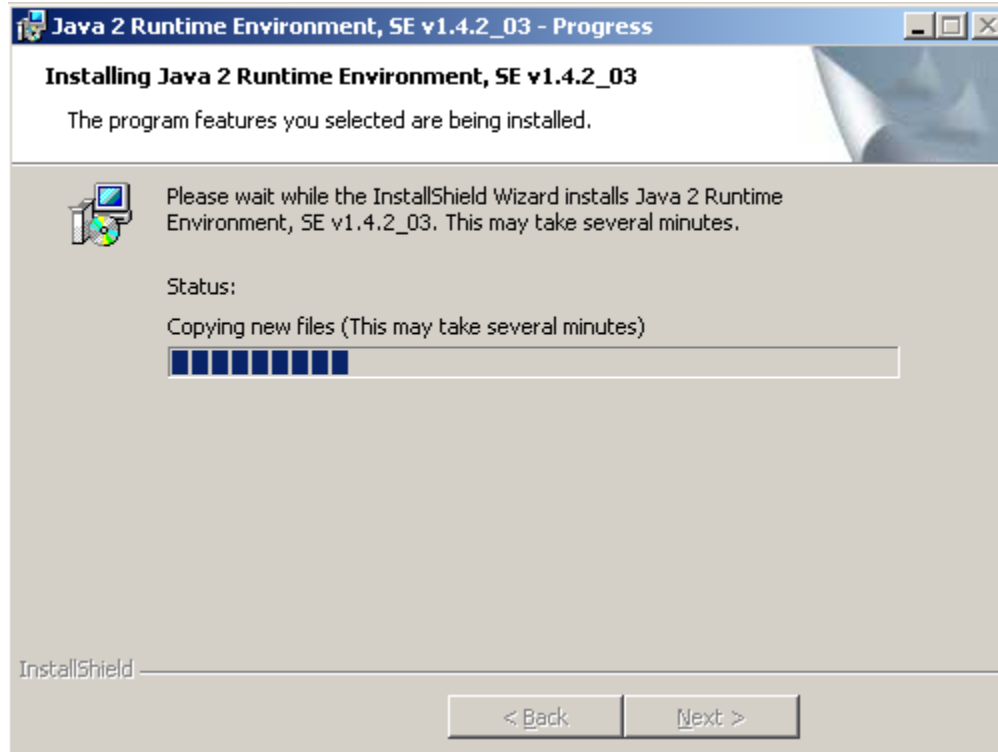
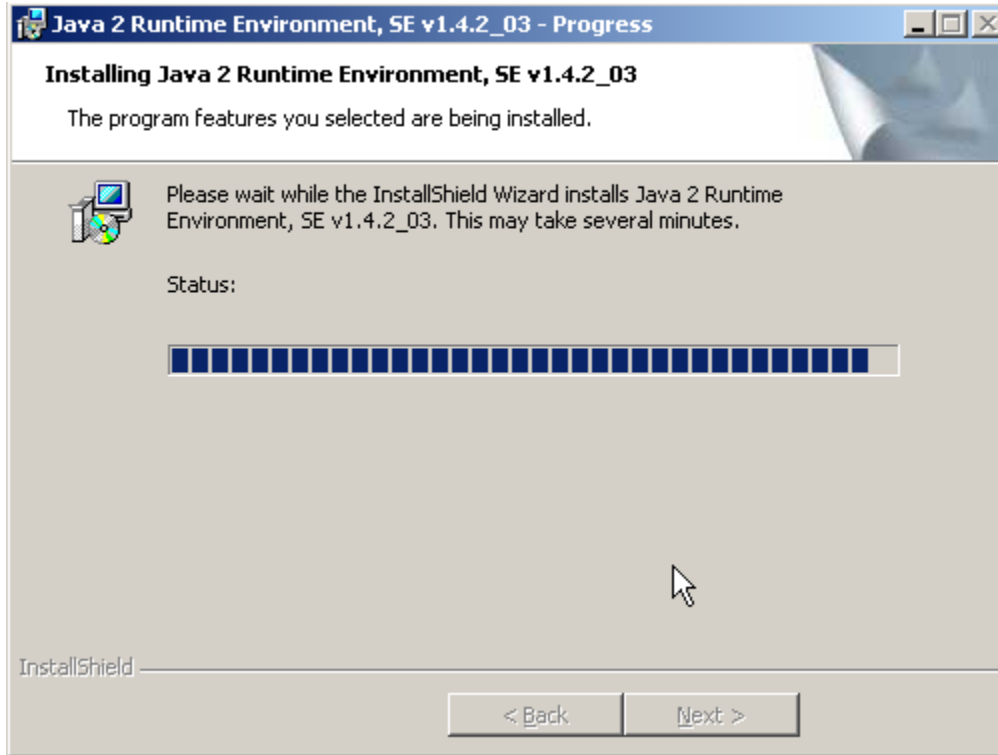
Select the "Next" button:

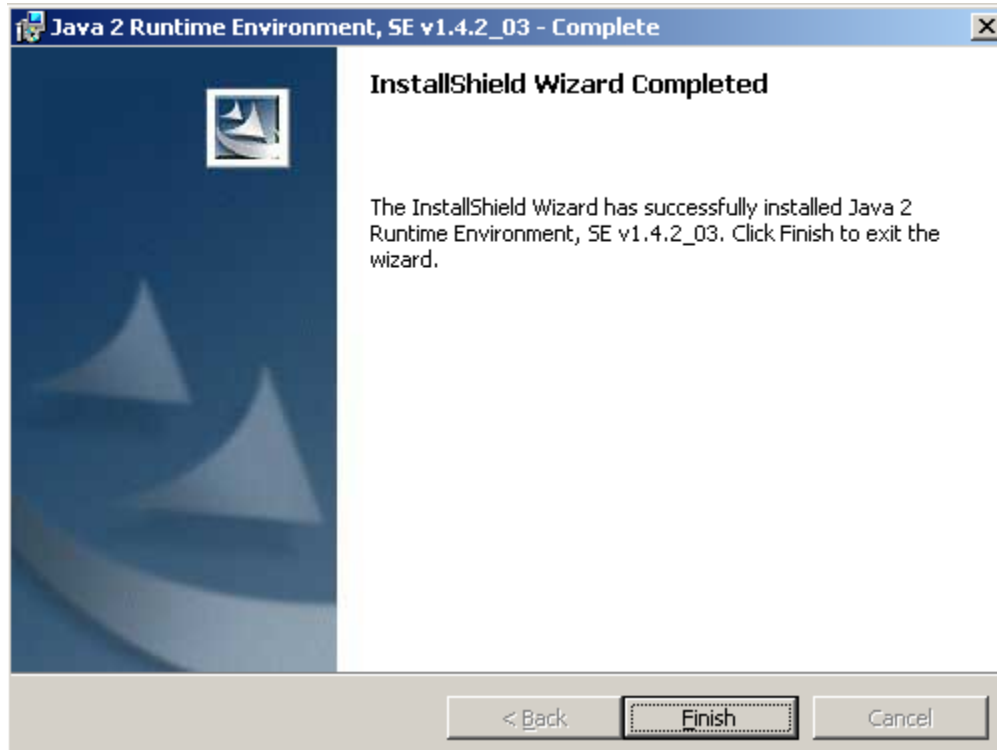
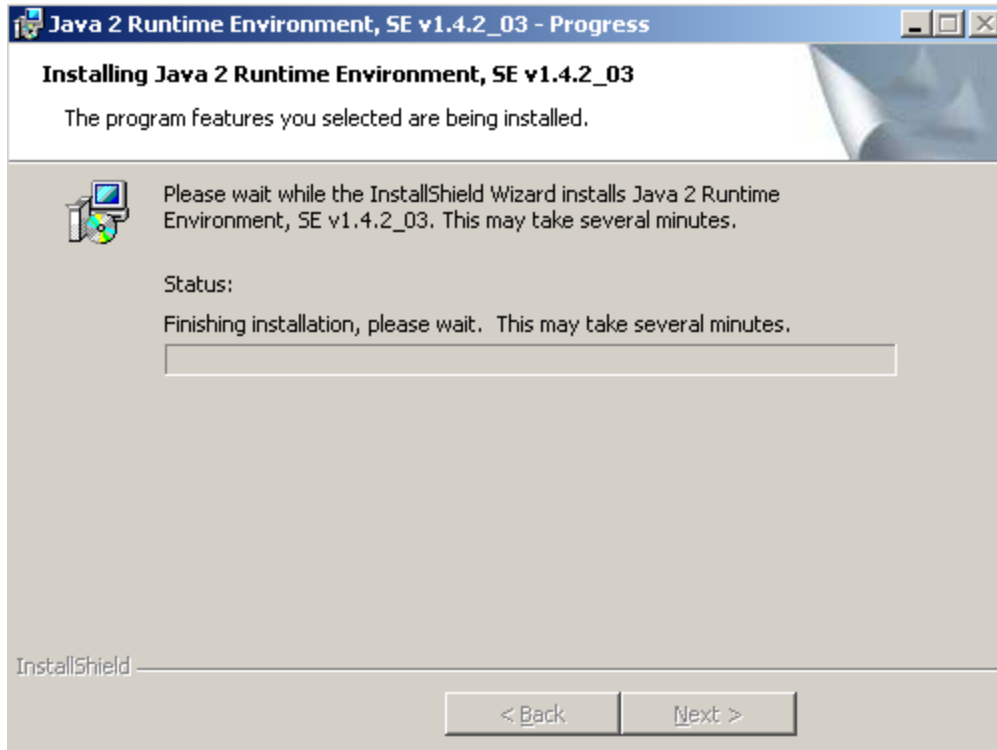


The "Typical" setup type is used, click "Next":

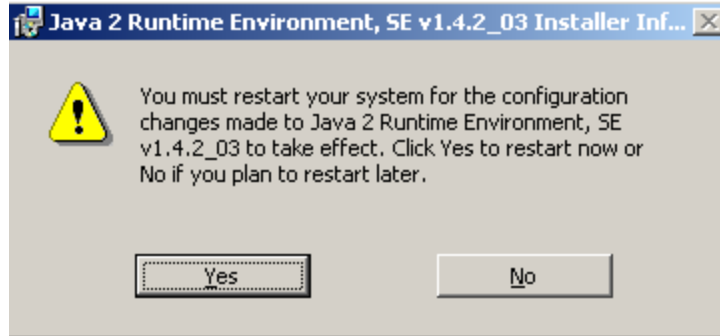


The JRE will now begin to download and install from the Internet:





Once complete the above screen will appear, click the “Finish” button and then restart your computer as directed below:



*Blank*



**CHAPTER**  
**3**

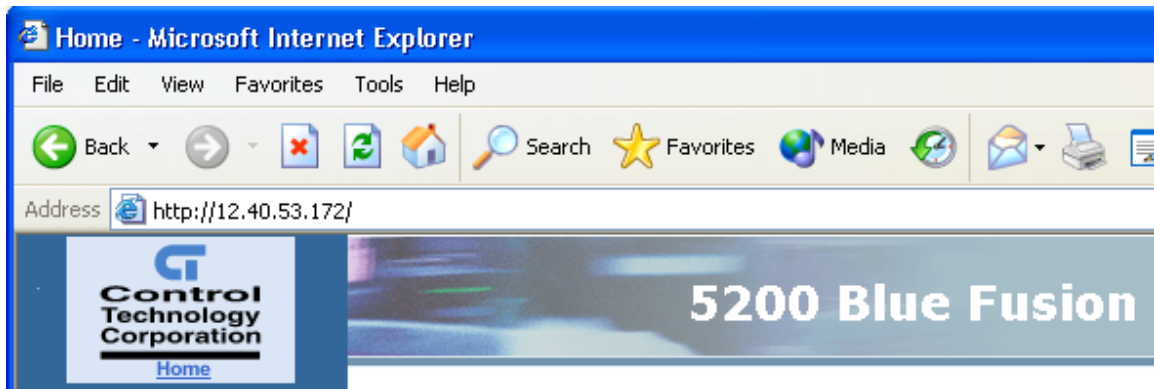
## Invoking WebMON



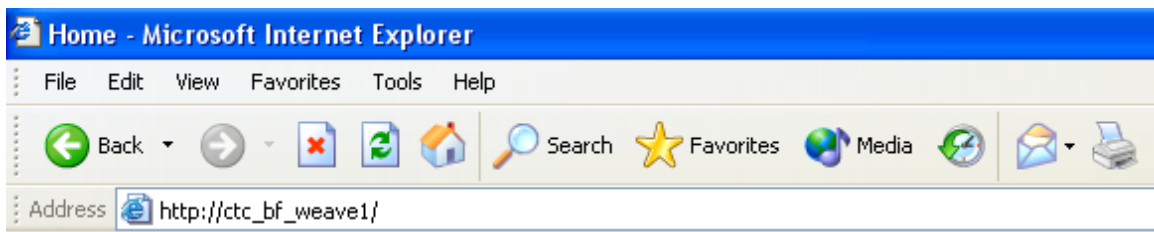
WebMON may be invoked either as a Web Interface, served up via the internal web server, or executed directly from your computer. Both methods require the use of a web browser, Microsoft Internet Explorer V6.0 or greater.

### **HTTP Web Interface**

The Web interface is invoked by typing the controller's DNS registered name, or its IP address, into a web browser. If the IP address of the controller was 12.40.53.172 then the following would cause it to be invoked from within Internet Explorer:



If using the DNS registered name, in the below example, CTC\_BF\_Weave1:



Once loaded a main browser window will appear, followed by a security alert. The security alert requests that you authorize WebMON to run since it is an unsigned applet.

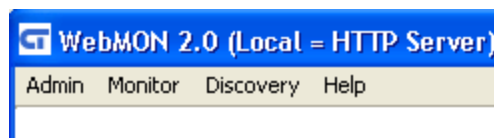
The security alert appears as:



If you fail to authorize WebMON to execute it will appear but nothing will function properly. Authorization of WebMON simply allows it to access local resources, like TCP connections on different IP addresses than the web server, local disk storage, etc.

Once authorized the full WebMON screen will appear, with full functionality available via its Main Menu selections. In addition a separate, background browser window will appear which will contain links to the Control Technology corporate web site, referencing the Controllers' documentation, specification, etc. This background browser should never be closed otherwise WebMON will exit. It is the container for the WebMON applet.

The following menu is available once WebMON is invoked:

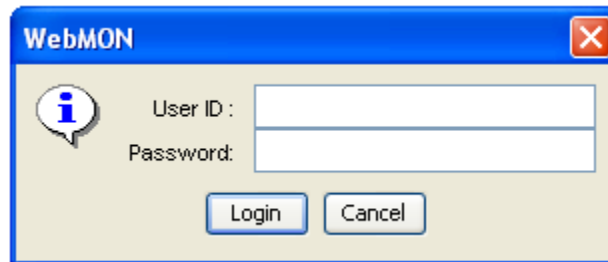


**Admin** - Administrative Functions such as configuration of a controller, file transfers, and quitting WebMON. The proper way to shutdown WebMON is via this menu selection.

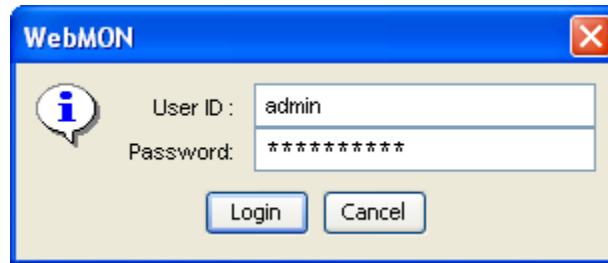
**Monitor** – Monitor controller data within data tables and resources such as registers, I/O points, etc.

**Discovery** – Broadcast a request that all controllers on the network identify themselves to WebMON. A discovery tree of all responding controllers will be viewable along with the ability to select any controllers and request Admin and/or Monitor capabilities via a popup menu.

Note that whenever a menu item is selected, from the main menu, a User Login dialog will appear. You must enter the proper User ID and Password for the controller you are accessing (default, admin/BlueFusion). Each additional time you attempt to access that same controller WebMON will remember the last used User ID and Password and fill them in for you as a default. Unless changed, you may select “Login” and use the default.



Once the User ID and Password have been entered for the first time for a controller a default will appear and the Enter key may be selected or the ‘Login’ button clicked, to use the default:



When using the Discovery mode, Chapter 9, the User ID and Password is maintained for each controller accessed, therefore if the User ID and/or Password are different for different controllers, it will still work properly.

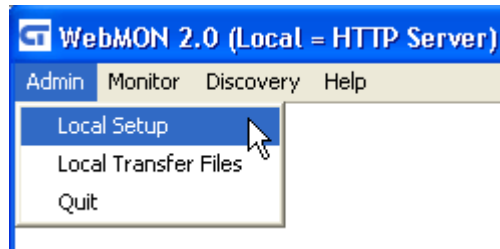
In addition, whenever a menu item is selected, it results in a connection to the controller. To conserve resources the controller will timeout on inactive connections after 30 minutes. For example if you were to access ‘Setup’, detailed in the next section, and then didn’t perform any updates for 30 minutes, you would get an error box informing you that the connection was lost the next time you tried to perform an update. Each time an update is made, the 30-minute timeout is reset.

*Blank*

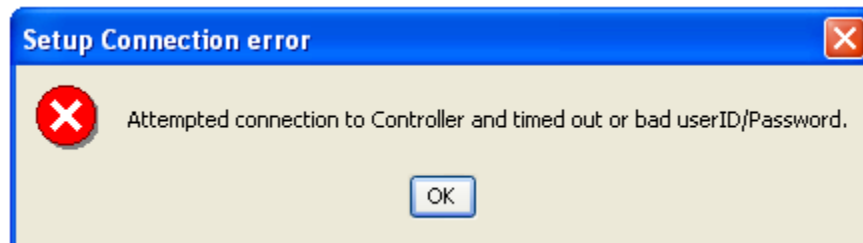
## Setup



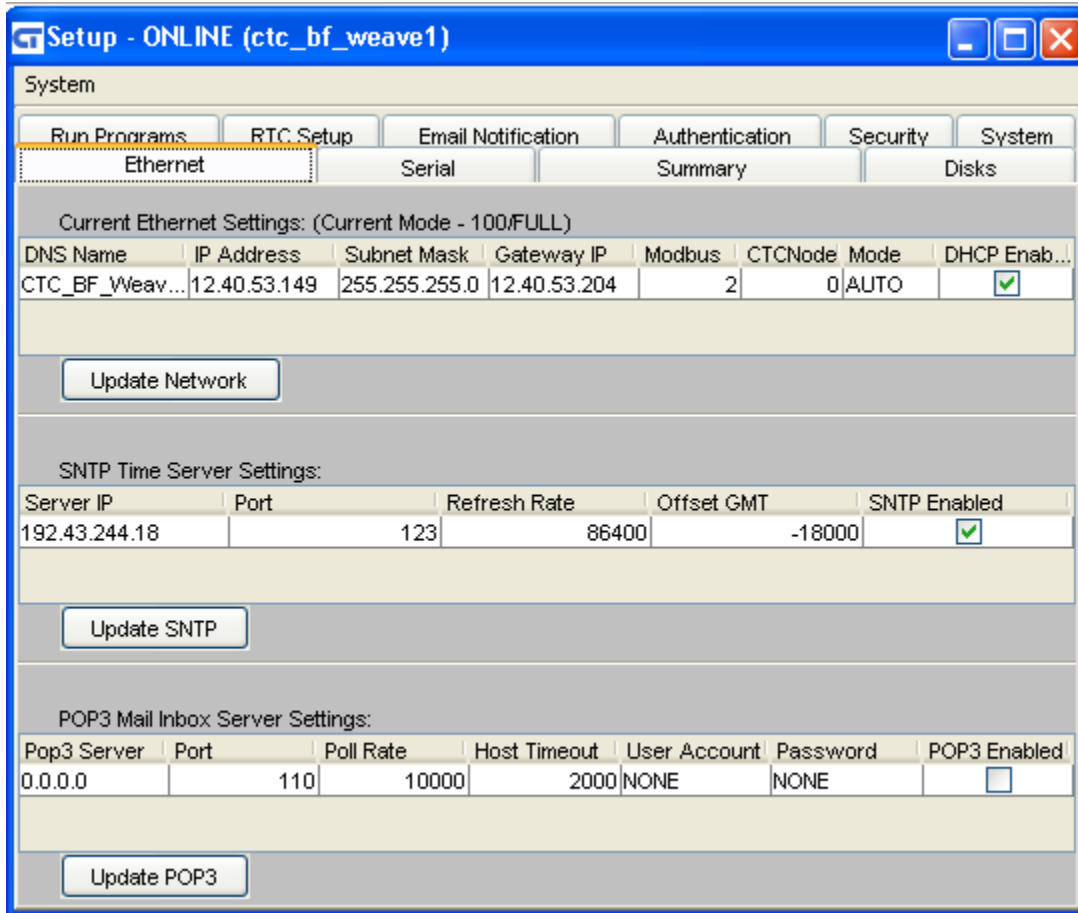
WebMON “Setup” is used to configure a controller’s parameters. It may be invoked from either the main menu, local (web server invoked from), or via the Discovery window, detailed in Chapter 9.



Once invoked, you will be prompted for a User ID and password (as detailed in the previous Chapter). A connection will then be attempted to the controller using TCP port 23 (telnet client) for all communications. The web interface, port 80, is only used to serve up the WebMON program. Should a problem occur, a communications error message similar to below will appear and the window will not open:



Once communications is established a form will appear consisting of a series of tabs.



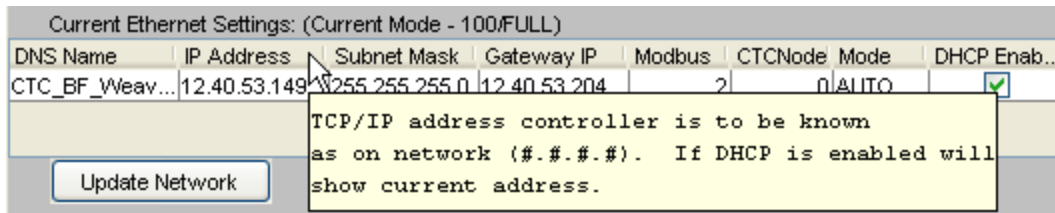
Each tab represents a sequence of parameters and status information within the controller. Available tabs are detailed within this section, in summary those available are:

- Ethernet
- Serial
- Summary
- Disks
- Run Programs
- RTC
- Email Notification
- Authentication
- System

Resident on each tab are data entry fields. These fields may be accessed by simply double-clicking the mouse on the area of interest and then typing in the required information. In some cases a check box may be provided to enable/disable the option, or a pull down box available for selection. In all cases, no changes will occur, in the controller, until the relevant 'Update' or 'Save' button is selected. Should there be any

## WebMON 2.0 User's Guide

questions with regards to the function of some parameter simply hover the mouse over the name of a table column and a tool tip will appear:



## Ethernet Tab

The Ethernet tab is used to set various network parameters. This consists of general network IP information, SNTP Time server interface and POP3 email.

### Current Ethernet Settings

The Current Ethernet Settings consists of a number of data entry fields, each with their own special functionality:

Current Ethernet Settings: (Current Mode - 100/FULL)							
DNS Name	IP Address	Subnet Mask	Gateway IP	Modbus	CTCNode	Mode	DHCP Enab...
CTC_BF_Weav...	12.40.53.149	255.255.255.0	12.40.53.204	2	0	AUTO	<input checked="" type="checkbox"/>

Update Network

- DNS Name
- IP Address
- Subnet Mask
- Gateway IP
- Modbus
- CTC Node
- Mode
- DHCP Enabled

#### ***DHCP Enabled (check box to enable)***

The controller is capable of retrieving its IP information automatically (IP Address, Subnet Mask, and Gateway IP), from a DHCP server, RFC 2131. The Dynamic Host Configuration Protocol (DHCP) is a communication protocol that lets network administrators automate assigning of IP addresses within a network.

All devices (computers, controllers, etc.), which reside on a TCP/IP network, must have an IP address assigned. Without DHCP, the IP address must be entered manually at each device. If devices move to another location in another part of the network, a new IP address must be entered. DHCP allows a network administrator to supervise and distribute IP addresses from a central point and automatically assigns a new IP address when a computer is plugged into a different location on the network. DHCP also provides other services beyond that of just an IP address. It provides Domain Name Service (DNS) server addresses, gateway information, Simple Network Time Protocol (SNTP, section X.X) servers, etc., thus allowing for fully automatic configuration of the controller IP parameters.

DHCP uses the concept of a "lease" or amount of time that a given IP address will be valid for a computer. The lease time can vary depending upon how long a user is




likely to require the network connection at a particular location. DHCP also supports static addresses for devices that need a permanent IP address.

Checking the check box on the Setup Screen enables DHCP. At power up, the controller will request to use whatever IP address is currently set (except 0.0.0.0 which enables DHCP), the DHCP server will either allow it or supply a new IP address. This final address will temporarily be written to the 20048 register block of the controller, but not permanently, and will appear in the "IP Address" data entry field. Once complete with all changes, simply press the "Update Network" button to notify the controller of changes. Values are immediately read back from the controller allowing for visual confirmation.

### ***DNS Name***

When the controller communicates with a DHCP server it also requires a unique system name that is typically used for DNS resolution (assuming the server is using dynamic DNS). Presently this name is derived from the controller's serial number, placing "CTC\_BF\_", before the number. For example if the serial number was 100-52801 then the DNS name entry for the controller would become CTC\_BF\_10052801. User settable names are also possible by simply double-clicking the data entry field and entering a unique name. Up to 20 characters are allowed in the Controllers DNS Name. When the "Update Network" button is selected the controller will immediately notify the DHCP server of a name change, if DHCP is enabled. If dynamic DNS is enabled, on your host, the name change will become available immediately on your network.

 Many software packages, and other devices with CTC communications drivers, do not have the capability to identify controllers by name, only by IP Address. Depending on how your network is configured, DHCP may change the IP address of the controller without warning, causing devices and software to lose connection or connect to the wrong controller. In this case, it is better to manually assign a static IP address to the controller. The network administrator should be contacted prior to assigning any IP address, to avoid conflicts.

### ***IP Address***

If you are not using DHCP to automatically obtain your IP Address information then the TCP/IP IP address is configured statically. It must be entered using a 'dot' notation as follows:

Example IP Address 168.254.132.34 (example)

The actual values to use will depend on the network that the controller is connected to. Contact your IT department to determine acceptable addresses for your network.

### ***Subnet Mask***

If you are not using DHCP to automatically obtain your IP Address Information then the TCP/IP subnet mask address is configured statically. It must be entered using a 'dot' notation as follows:

Example Subnet Mask: 255.255.255.0 (typical)

The actual values to use will depend on the network that the controller is connected to. Contact your IT department to determine acceptable addresses for your network.

### ***Gateway IP***

If you are not using DHCP to automatically obtain your IP Address then the TCP/IP Gateway address is configured statically. It must be entered using a 'dot' notation as follows:

Example Gateway 168.254.132.88 (example)

The actual values to use will depend on the network that the controller is connected to. Contact your IT department to determine acceptable addresses for your network. A value of 0.0.0.0 will disable the use of a gateway. A Gateway is the address to which requests will be forwarded if they are outside the range of you IP domain, as tested against the assigned subnet mask. Typically a gateway is used to forward requests to another network and/or the internet.

### ***Modbus***

The Modbus address is used to set the address which will be used by the Modbus/TCP communications protocol. It is typically referred to as the Device ID. It may be set from 1 to 255.

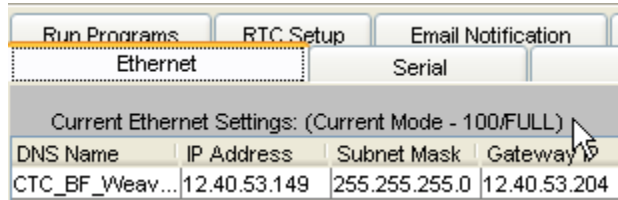
### ***CTC Node***

The CTC Node number is used by the CTNet protocol. This is a lightweight non-routable Ethernet protocol used by legacy CTC controllers. It is recommended that UDP be used, instead, whenever possible, since it is routable. Setting this node number to 0 disables its use in the controller. Be careful setting this node number since no two controllers can have the same address. Valid numbers are from 1 to 32767. Some very old CTC controllers only communicate on nodes 1 to 254.

### ***Mode***

Mode is used to set the Ethernet connection method, speed and duplex, and typically is not used. By default it is set to Auto. Auto means, auto-negotiate, or let the controller and external router/switch negotiate connection speed and duplex. The fastest possible will generally be negotiated, 100 Megabits/Full Duplex. Sometimes, where old wiring may exist or noisy environments, it is best to reduce the speed of the Ethernet interface. Also if Ethernet speed is not important, the slower speed will reduce the load on the controller and generally allow increased performance by other aspects of the controller during peak Ethernet traffic.

A pull-down box is provided to override the default. Available are 100 full/half duplex, 10 full/half duplex, and auto. Note that the current negotiated speed is shown in the text area above the data entry fields. Below shows the current speed is negotiated to 100 full duplex:



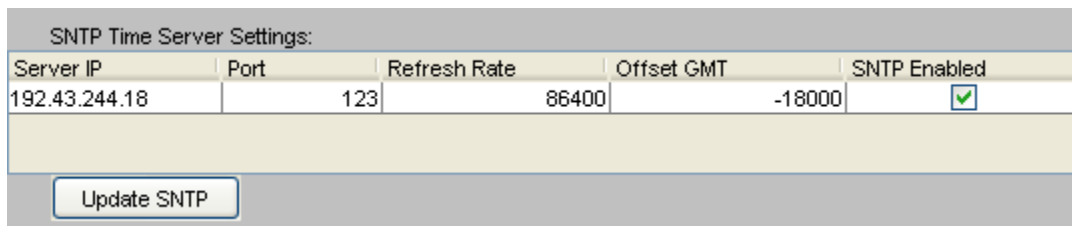
### SNTP Time Server Settings

The controller supports the Simple Network Time Protocol (SNTP) as a client connecting to a server. This protocol provides a means to synchronize a computer system clock to that of the world clock, via the internet. Government agencies provide this service for computers to query the current atomic clock time and adjust their clocks appropriately. For more detailed information reference [www.time.gov](http://www.time.gov) and [www.boulder.nist.gov/timefreq/service/its.htm](http://www.boulder.nist.gov/timefreq/service/its.htm).

The time returned is based on Coordinated Universal Time (UTC), which is Greenwich Mean Time (GMT). As such, there is no adjustment for daylight savings time or time zones, that must be done locally. To avoid daylight savings time problems it is recommended that you base the controller time on GMT (default) but provisions have been provided to automatically set the clock based on the time zone you are in, using an offset from GMT. Refer to the RTC Tab for further details.

Use of SNTP is not a requirement but typically real time clocks can be expected to drift up to 30 seconds per week. The controller may drift up to 12 seconds per week, depending on the tolerance of crystals, components, etc. Synchronization allows its real time clock to be automatically set with regards to date, year, day of week, and time.

The SNTP Time Server Settings consists of a number of data entry fields, each with their own special functionality:



- Server IP
- Port
- Refresh Rate
- Offset GMT
- SNTP Enabled

By default the controller will use the IP address of 192.43.244.18, port 123. Updates will be performed once/day and the clock is set to GMT.

**Server IP**

The “Server IP” address designates the host which will provide the time service for the controller. By default the address is 192.43.244.18. Data is entered using the “dot” notation. Entering an IP address of 0.0.0.0 will disable SNTP requests.

**Port**

The “Port” is the TCP/IP port that the Time Server will be listening on for time requests. Typically this is port 123, and is the factory default.

**Refresh Rate**

The “Refresh Rate” is the number of seconds before the next synchronization request with the SNTP server. For example 3600 would be an hour, 86400 would be 24 hours (default). When a change in time is made to this value it typically takes about 1 minute before the new value will take effect. Power cycling of the controller is not required.

**Offset GMT**

“Offset GMT” contains the number of seconds to add or subtract from GMT once the time is received from the server. The default is 0, which means to set the clock to GMT. -18000 (-5 hours) would be the value used for Eastern Standard Time during daylight savings time, -14400 (4 hours) when not. Note that the value is both positive and negative.

**SNTP Enabled**

If the check box is checked SNTP requests will be enabled and done in the background based upon the above parameters. When deselected the IP address will be forced to 0.0.0.0. If the time service is not being used it is best to ensure this box is not checked, thereby conserving CPU resources.

**POP3 Mail Inbox Server Settings**

POP3 is a fairly simple and standardize protocol allowing client devices to access email accounts resident on a POP3 Server. For proper operation controllers should be assigned their own email account. You may not share an email account with a controller since each controller will read and delete each email, as it is read and processed.

The POP3 Email Server configuration consists of a number of data entry fields, each with their own special functionality:

POP3 Mail Inbox Server Settings:						
Pop3 Server	Port	Poll Rate	Host Timeout	User Account	Password	POP3 Enabled
12.40.53.10	110	5000	2000	Test5200Email	vetx123	<input checked="" type="checkbox"/>

- POP3 Server
- Port
- Poll Rate
- Host Timeout
- User Name
- Password
- POP3 Enabled

### ***POP3 Server***

The “POP3 Server” IP address designates the host which will provide the POP3 mailbox account for the controller. This must be the servers IP address, entered in “dot” notation.

### ***Port***

The “Port” is the TCP/IP port that the POP3 Server will be listening on for mail requests. Typically this is port 110, and is the factory default.

### ***Poll Rate***

The “Poll Rate” is the time, in milliseconds, that the controller will wait until it checks for available email, within its mailbox. All available email will be read and deleted as processed, in a sequential order. After processing this time delay will occur until the next processing sequence. 10000 milliseconds (10 seconds is the default interval).

### ***Host Timeout***

The “Host Timeout” is the time, in milliseconds, that the controller will try to contact its POP3 server and wait for responses for mail requests. It is considered the error timeout. After this period of time the controller will stop trying to contact the server and wait the next poll rate interval before trying again. The default timeout period is 2000 milliseconds (2 seconds).

### ***User Name***

The “User Name” is the name needed to log into the mailbox. This is typically the mailbox name but could be set different by the POP3 server. It is limited to 30 characters.

### ***Password***

The “Password” is the password required, along with the “User Name” to log into the mailbox being supplied by the POP3 server. It is limited to 30 characters.

### ***POP3 Enabled***

## WebMON 2.0 User's Guide

A check box is available to enable the POP3 functionality, when checked POP3 is active. Once all changes have been made to the above parameters select the “Update POP3” button to make the changes current in the controller.

## Serial Tab

The “Serial” tab allows immediate configuration of the local COMM1 and COMM2 serial ports, within the controller. All changes take effect immediately and are placed in permanent storage, thereby surviving power cycling. Once parameters are updated and immediate read is done of all parameters, providing visual verification of your changes.

## COMM 1/2 Configuration

The COMM configuration provides a table of two rows, one for each serial port. It consists of a number of data entry fields, each with their own special functionality:

Run Programs	RTC Setup	Email Notification	Authentication	Security	Threads	
Ethernet	Serial		Summary	Disks		
Serial Port Settings:						
COMM	Baud Rate	Data Bits	Parity	Stop Bits	Protocol	Address
1	19200	8	None	1	CTC Binary	4
2	19200	8	None	1	CTC Binary	8

- COMM
- Baud Rate
- Data Bits
- Parity
- Stop Bits
- Protocol
- Address

### COMM

This is not an editable field. It is used to reference either COMM1 (row 1) or COMM2 (row 2).

### Baud Rate

A pull down list box is available to select the desired baud rate. Baud rates from 1200 to 115,600 are available. Note that using baud rates above 19,200 can cause system degradation, dependent upon protocol and data flow.

### Data Bits

A pull down list box is available to select either “7” or “8” data bits.

### Parity

A pull down list box is available to select “None”, “Odd”, or “Even” parity.

### Stop Bits

A pull down list box is available to select either “1” or “2” stop bits.

### Protocol

## WebMON 2.0 User's Guide

A pull down list box is available to select the individual protocols to be active on each port. Details of each is provided in the 5200 Communications Guide (950-520000). Available selections are:

- CTC Binary (Default, compatible with CTCMON and ctccom32.dll)
- Modbus Master RTU – controller polls the device.
- Modbus Master ASCII – controller polls the device.
- Modbus Slave RTU – controller polled by external device
- Modbus Slave ASCII – controller polled by external device

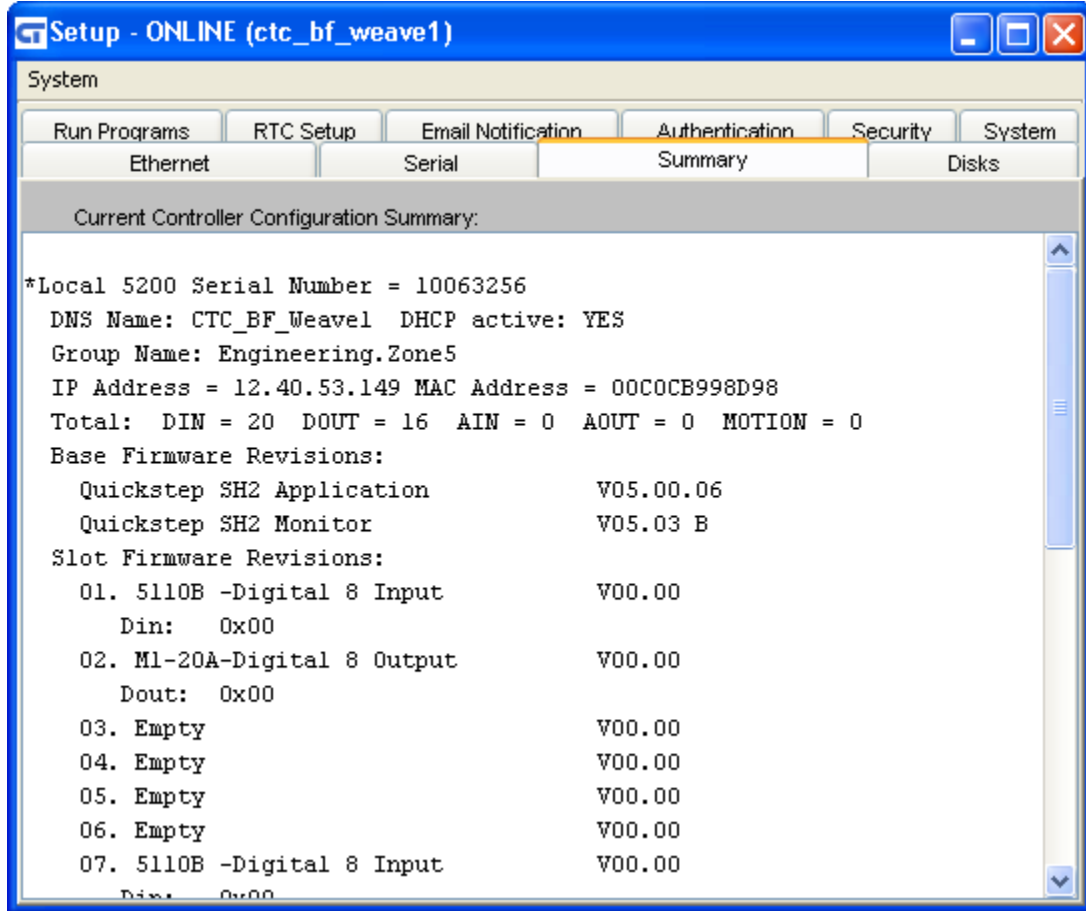
### *Address*

This is the address to be use when Modbus protocols are selected. When in Master mode only a single device may be polled. To poll multiple devices the Address register must be changed by the Quickstep program, dynamically. An address from 1 to 255 is valid.



## Summary Tab

The “Summary” tab provides an overview of the controller firmware revisions, network settings, available modules and expansion units. It is the same command executed when using Telnet, “get versions”.



## General Controller Information

General controller information consists of a network summary, including active IP address, subnet mask, gateway address, DNS Name, Group Name, MAC Address and DHCP state. Additionally serial number information and total I/O count are available. Below is an example this portion of the “Summary” information:

```
*Local 5200 Serial Number = 10063256
DNS Name: CTC_BF_Weave1 DHCP active: YES
Group Name: Engineering.Zone5
IP Address = 12.40.53.149 MAC Address = 00C0CB998D98
Total: DIN = 20 DOUT = 16 AIN = 0 AOUT = 0 MOTION = 0
```

## Firmware Revisions

There are two sections of firmware revisions, those that pertain to the main CPU (Base) and that of each individual module. The main CPU consists of a boot monitor and the main Application Firmware:

```
Base Firmware Revisions:
Quickstep SH2 Application      V05.00.05
Quickstep SH2 Monitor         V05.02 B
```

## Modules and Expansion Units

Each module plugged into the controller is listed individually within the “Summary” listing, its slot position, type, and individual firmware revision. Below is an example:

```
Slot Firmware Revisions:
01. 5120C-Digital 8 Output      V00.00
    Dout: 0x00
02. 5120C-Digital 8 Output      V00.00
    Dout: 0x00
03. Empty                       V00.00
04. 5120C-Digital 8 Output      V00.00
    Dout: 0x00
05. 5120C-Digital 8 Output      V00.00
    Dout: 0x00
06. Empty                       V00.00
```

Each expansion unit is also referenced, currently up to 6 modules/unit:

```
06. Empty                       V00.00
07. No Expansion Connected      V00.00
08. No Expansion Connected      V00.00
09. No Expansion Connected      V00.00
10. No Expansion Connected      V00.00
11. No Expansion Connected      V00.00
12. No Expansion Connected      V00.00
```

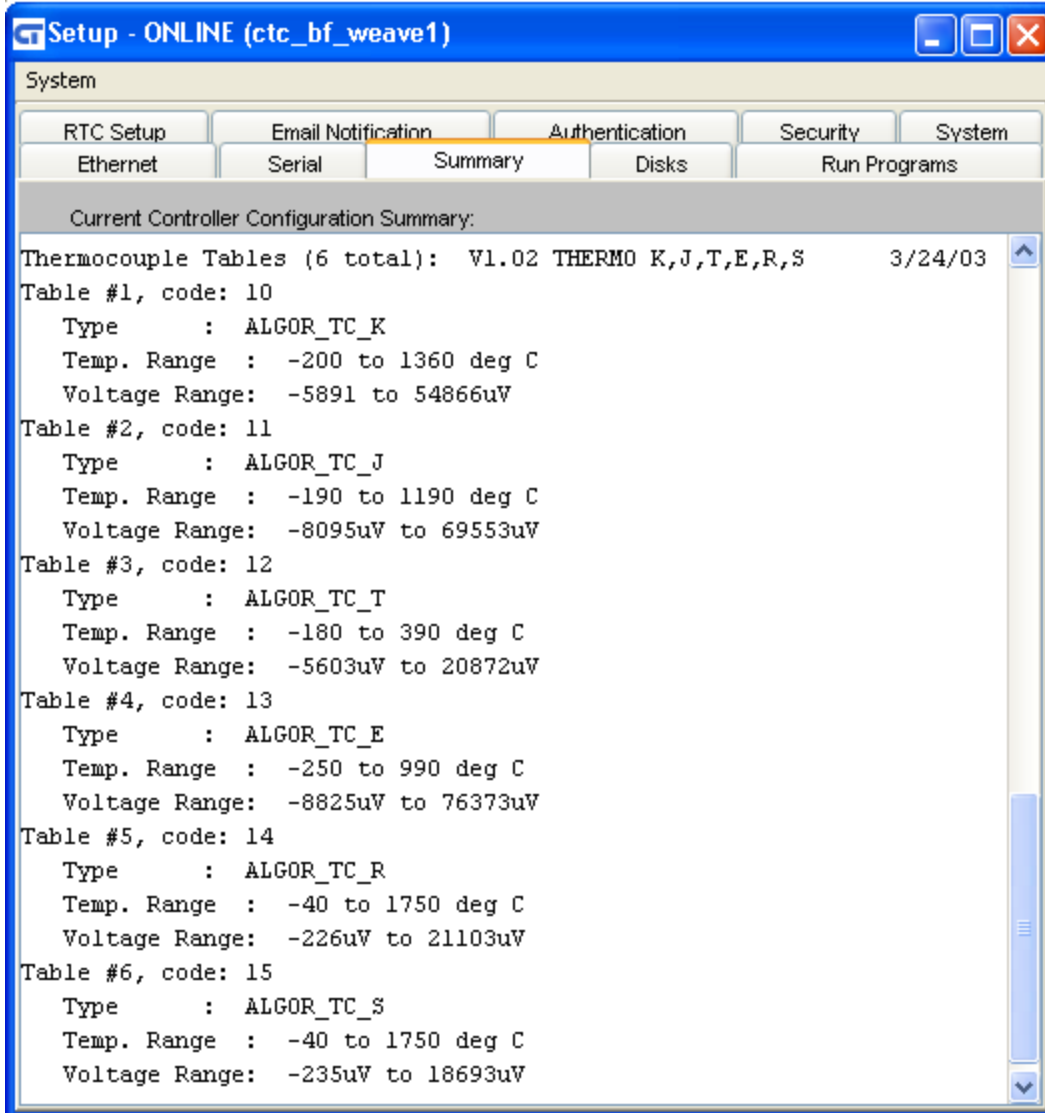
## Thermocouple Tables

If thermocouples are being used a table must be loaded called Thermocouples.tbl. The available algorithms are shown in the “Summary” listing. If none are available:

```
23. No Expansion Connected      V00.00
24. No Expansion Connected      V00.00
```

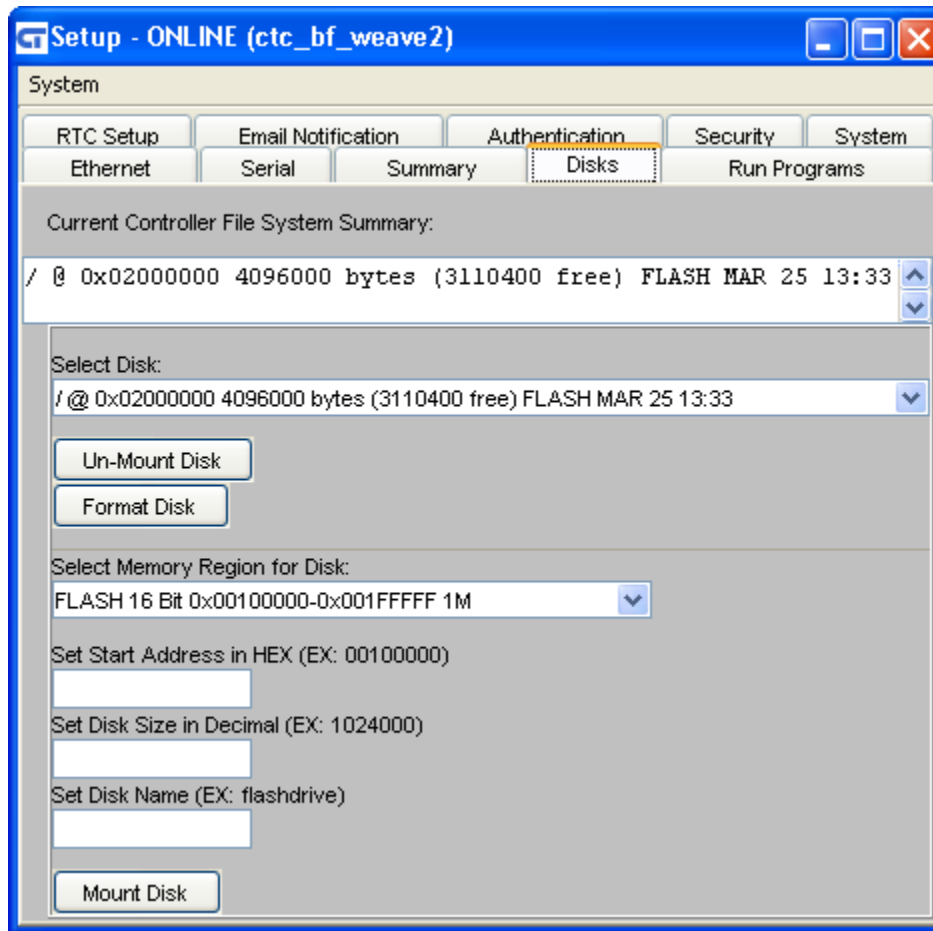
No Thermocouple.tbl file found.

If Thermocouples are available then those loaded will be displayed viewable along with the proper code to use for their selection, working temperature and voltage range. Below shows tables available for 6 tables, handling K, J, T, E, R, and S thermocouples:



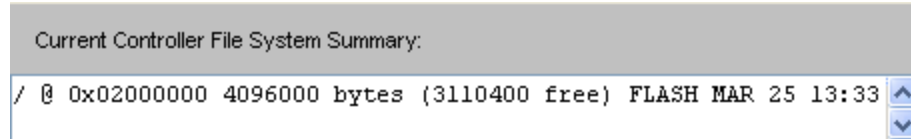
## Disks Tab

Controllers allow for numerous memory configurations. Each allows for unused space to be used as a disk drive. Both FLASH and Non-Volatile RAM are available, as well as Volatile SDRAM. Typically FLASH is used for information that is primarily written infrequently, Non-Volatile RAM when numerous changes are needed and must survive power cycling. SDRAM can be used for storage that changes frequently but does not need to survive power cycling. Note that FLASH is the slowest of all three types and is preferably used offline for write operations.



Disk drives may be created from the “Disks” tab. Mount refers to defining a drive, its size, memory start location, and type. A mounted (available) drive will automatically be mounted at power-up and multiple drives are supported. Each drive is assigned a name, up to 20 characters, with its name referenced from the root drive. The root drive’s name is always ‘/’. Thus a drive mounted with the name of ‘mydrive’ would be referenced as ‘/mydrive’.

All currently mounted drives appear in the Active Drive list window:



Each listed drive contains specific information about the mounted drives. Using the above example the first column starts by listing the name of the drive, this one is '/', it starts at memory location 0x02000000 Hex in the controller. It has an unformatted capacity of 4,096,000 bytes but only 3,110,400 bytes left for storage. The difference in sizes is the overhead of formatting the disk sectors and any files that are currently on the drive. The type of memory being using is FLASH, either RAM or FLASH will be appear, and it was created on March 25<sup>th</sup> at 13:33.

The following sections will discuss how to create (mount), remove (un-mount), format and analyze disks using the available memory in the controller.

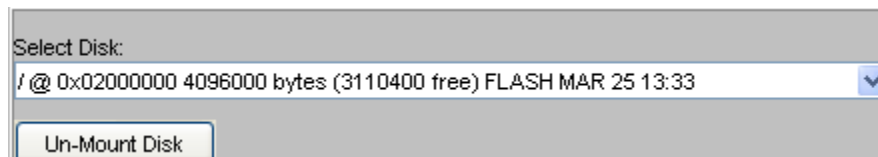
### Un-Mounting and Formatting Drives

Mounted drives may be removed (un-mounted) or formatted. Formatting will destroy all the contents of the drive, completely erasing it. To format a drive, simply select the desired drive from the pull down combo box and select the "Format Disk" button with your mouse:

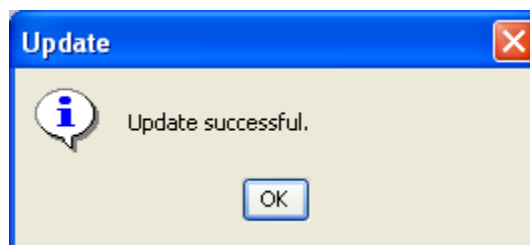


Note the Format operation will take considerable time when using FLASH memory, up to 20 seconds or more for large sizes. RAM typically formats in a few seconds. Formatting is done automatically on a newly created drive.

To un-mount an existing drive you may simply select the mounted drive from the pull down list:



Once selected, click the "Un-Mount Disk" button. Verification will appear and the Mounted Disk window will be updated, less the un-mounted drive.

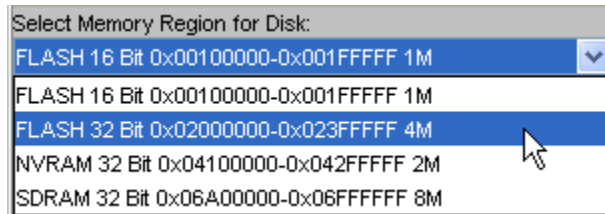


Un-mounted drives may later be remounted, with data preserved, as long as they are mounted using the exact same size, start address and name. The root drive may also be un-mounted and a new one created. To do so make sure it is the last drive un-mounted

otherwise references to the other drives will have to be created again using “Mount”, (no data loss will occur).

### Mounting New Drives

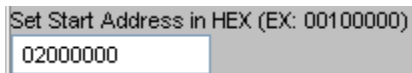
To mount a new drive the memory area desired must first be selected from the pull down memory map:



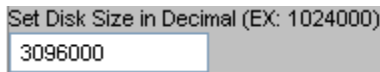
It is up to the user to track what memory is being used from the available memory purchased. **Note:** Different memory expansion options are offered for CTC controllers. The memory is installed during the controller assembly process and is not field upgradeable.

The memory maps shown above are the base regions available during the initial software release, future WebMON releases may track actual memory mapping. For the moment this is a limitation of WebMON, not the controller. Once a region is selected WebMON will only let you create a disk within that address space, entering an illegal value is not allowed.

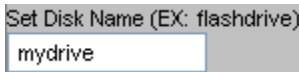
Once selected the start address for the disk is entered in HEX, 02000000 in the example below (addresses will be rounded to the next 32K block):



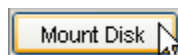
Followed by the size of the desired disk drive, in bytes (Example – 3096000):



The name to use for the drive to mount is also entered, in this case “mydrive”:

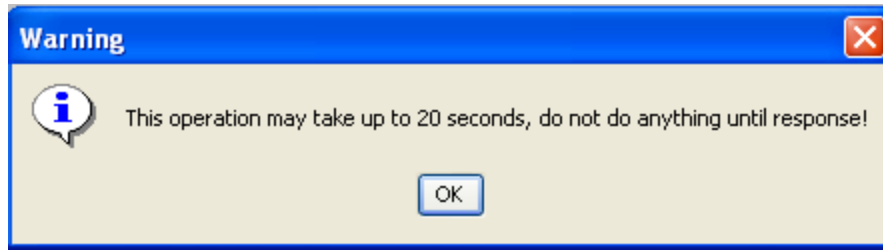


To create the drive, select the “Mount Drive” button.

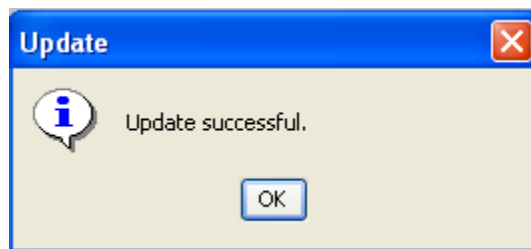


The drive will automatically be formatted, if required. If data is valid and the same name is used as a previous mount, the drive will not be formatted, and data preserved.

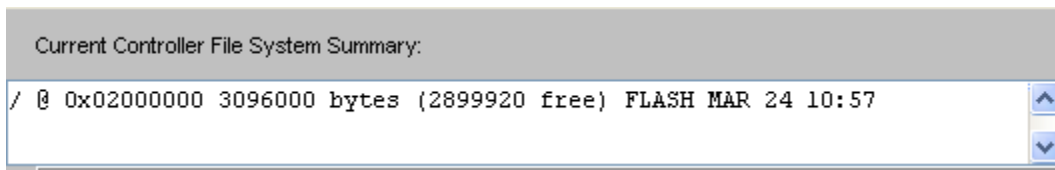
Depending on how large a disk you are creating, especially if flash, a long delay can be expected and no response from WebMON functions will be available. A warning box will appear:



Upon completion the following notice will appear, if successful:

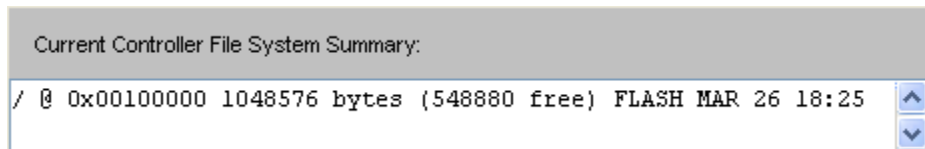


In addition the File Summary will show the drive, free space, type, and date/time created:



### Example – Changing the root drive

The manufacturing defaults, for a new controller, creates a small 1 Megabyte disk drive, for general purpose use. This is large enough for most purposes, leaving about 550K bytes of user space after WebMON is installed. Typically the following Disk Tab screen would appear on a new controller:

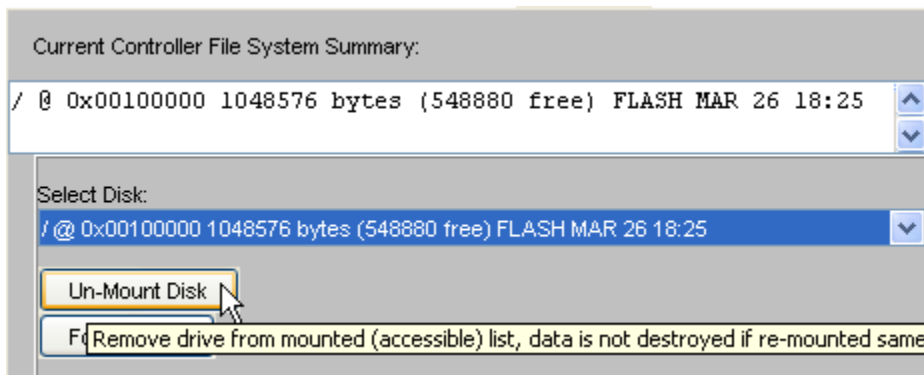


Since the root drive is where the `_system` subdirectory resides, it may be desirable to have a larger disk capacity. Assuming the option has been purchased, the root drive can be changed to boot from that drive. A larger root drive is useful for larger Web sites, more Quickstep system storage capacity, etc. In this example the controller was ordered with the additional 4 Megabyte Flash option and the controller will be updated to make that

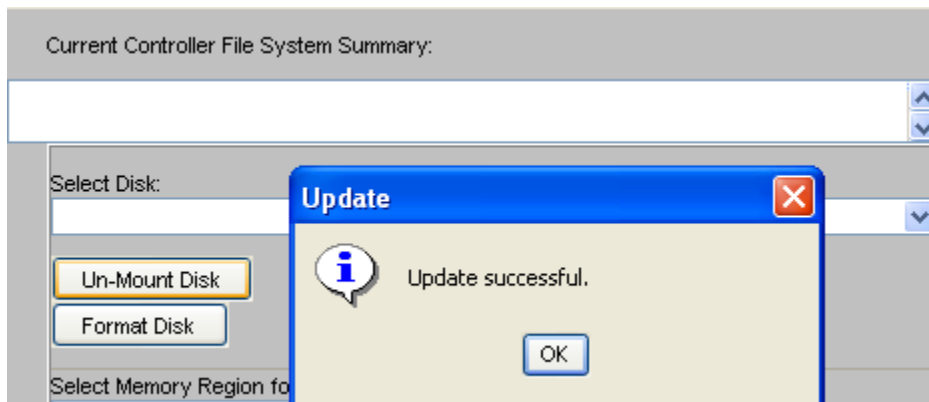
the root drive. In the next section we will show how to re-use the default 1M space, shipped with every controller, as an additional drive.

**Step 1** –Either download the WebMON environment from the CTC corporate web site ([www.ctc-control.com](http://www.ctc-control.com)) or ftp to the controller and upload all the files in the /\_system/Web and /\_system/Web/jar directories. In addition, upload any important files that you have put on the controller since they will no longer be available after the next step. Failing to do so will result in complete data loss since you are about to remove your main disk. **No tasks may be accessing the drive during disk mount/un-mount operations.**

**Step 2** – Using WebMON select the root drive using the pull down combo box and click un-mount:



After selection an acknowledgement dialog will appear, note that the File System Summary area is now empty:

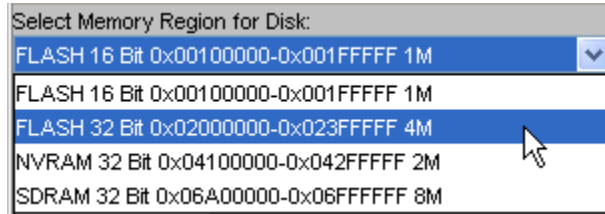


At this point no drive is mounted, if power were cycled, the controller would, by default, restore the previous factory default mounting.

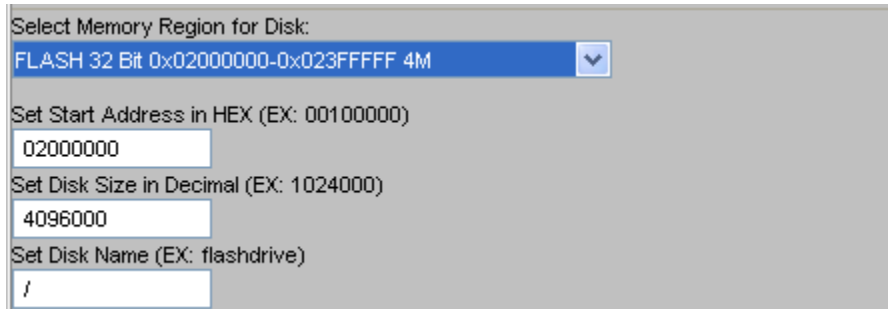
**Step 3** – Next a new drive must be created, and mounted as root, '/'. To Mount a the new drive the memory area desired must first be selected from the pull down memory map, in



this instance, a 4M Flash upgrade was purchased and is known to be available starting at 0x02000000, thus that area is selected:



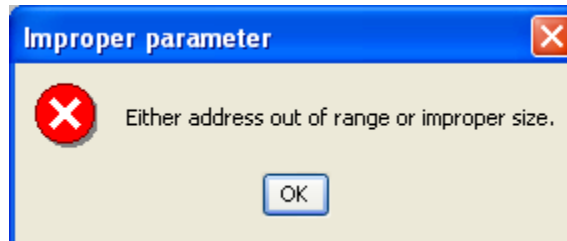
Once selected the start address for the disk is entered in HEX, 02000000, the desired disk size 4096000, in bytes, and the disk name to use. The name is '/' in this example since we are replacing the root drive. Your entries should now look as below:



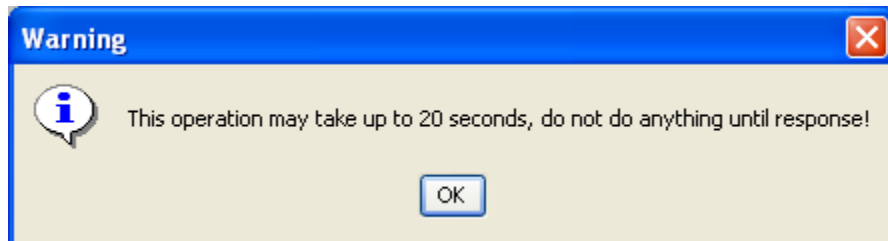
**Step 4** – Mount the new drive by selecting the 'Mount' Button:



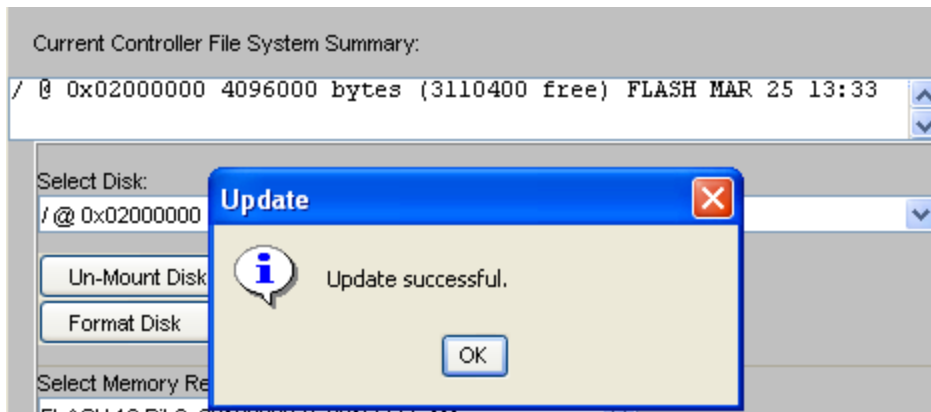
If you happened to enter a bad Start Address, such as 12000000, the following would appear when you tried to mount the drive:



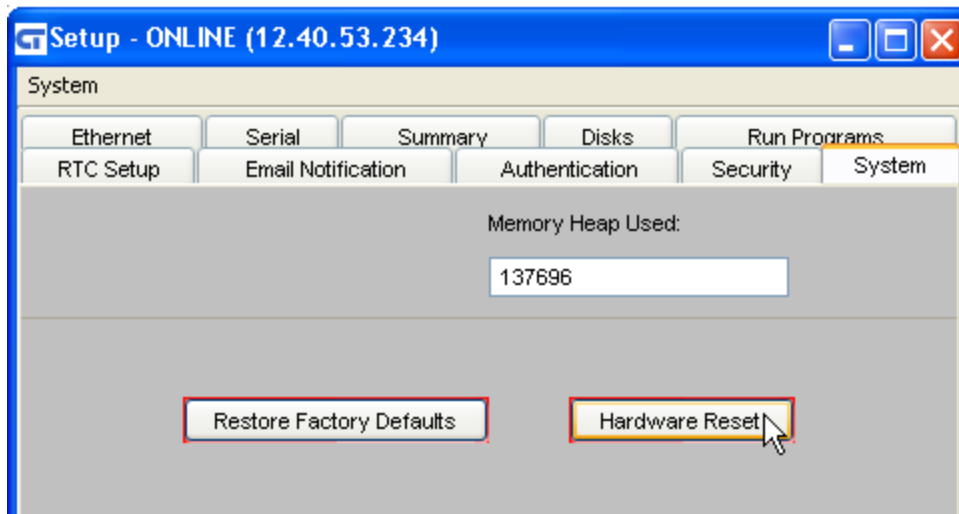
Simply correct the address and try again. You will be warned that creating a new drive can be time consuming:



After clicking OK the dialog box will close leaving a tan rectangle and WebMON will not be responsive for the time period required to create the drive. WebMON is waiting for either a controller acknowledgement or a timeout:



Note that in the background the new drive is automatically listed in the File System Summary window. A new drive has now been created and mounted in place of the old root drive. **Prior to use you must first reboot the controller.** This can be done by cycling power or using the System Tab->Hardware Reset button. The connection to the controller will be lost during this time period. It is not necessary to exit WebMON, simply select a menu item once the controller is back online and it will re-connect.

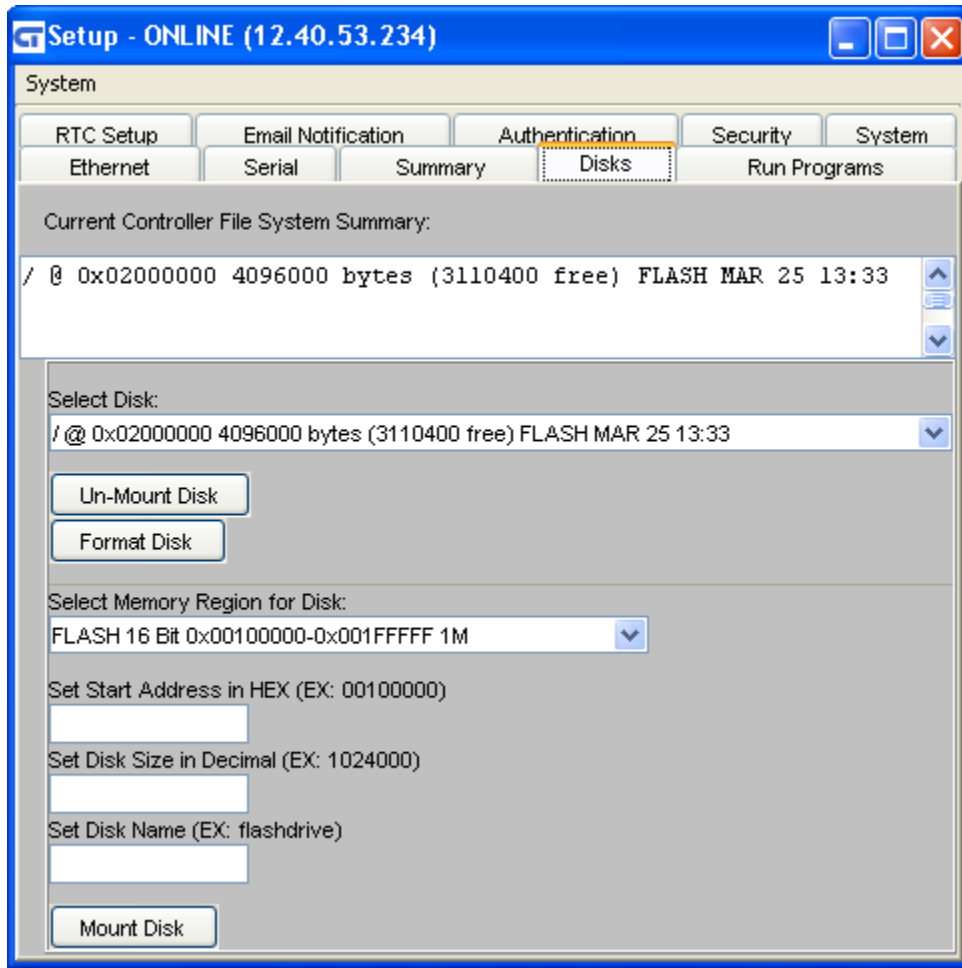


**Step 5** – Using FTP, restore any files, which existed on the drive. It may be necessary to re-create some subdirectories. For example `/_system/Web/jar` will need to be created as well as any directories and files required by your application.

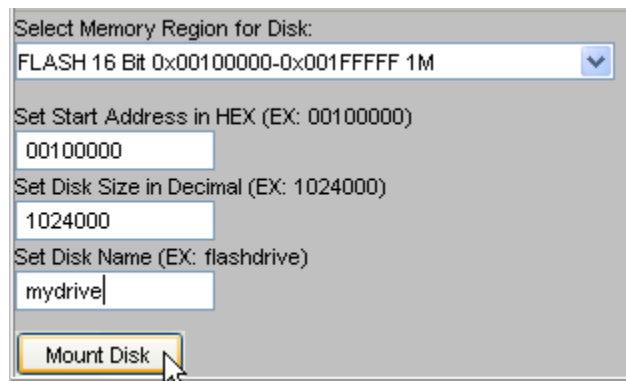
### Example – Adding a drive to root

In the previous example the root drive was replaced by a larger FLASH memory option. In this example we will create a new drive using the original flash disk, thus having two drives available, root, and our new one, which will be called 'mydrive'. Up to 5 drives may be created using different regions of memory.

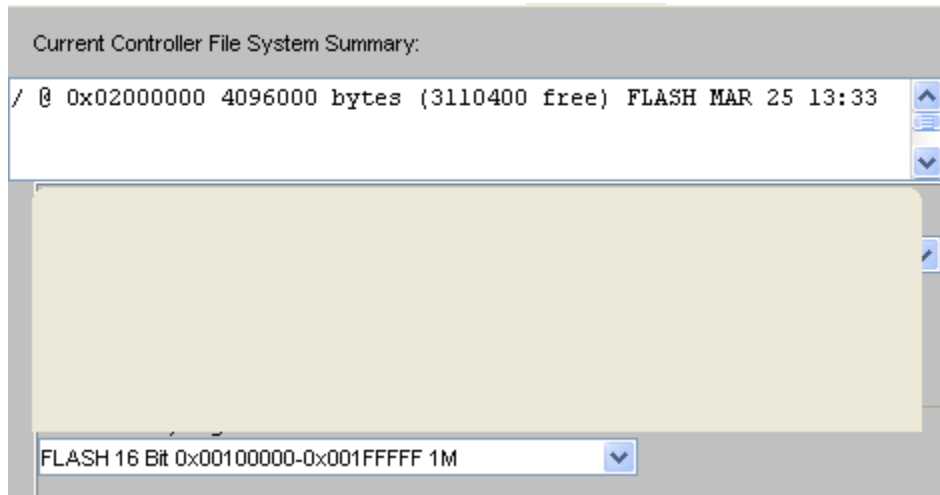
After completing the previous example, Disk Tab will appear as below:



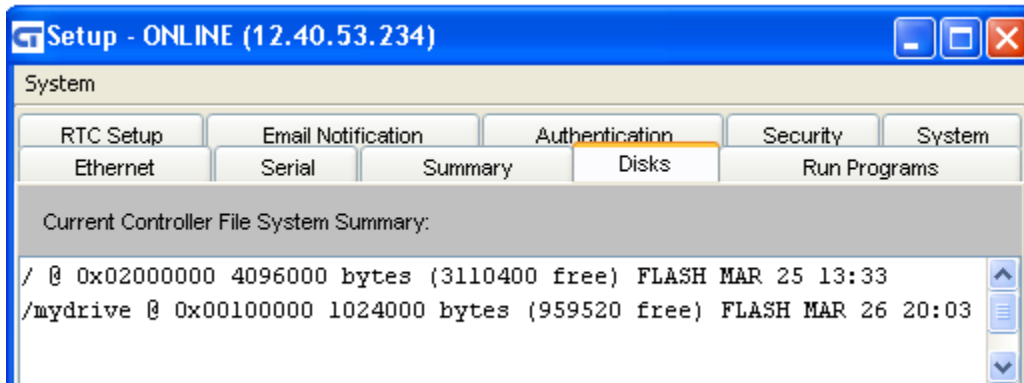
Since the original factory configuration used the 16 bit flash block from 0x00100000-0x001FFFFFF, for the root drive, that area is now free for general use. Referencing the previous example we will fill out the new Mount information for a drive starting at 00100000 Hex, 1024000 bytes in size and having a name of 'mydrive'. Select the 'Mount Disk' button once your screen appears as below:



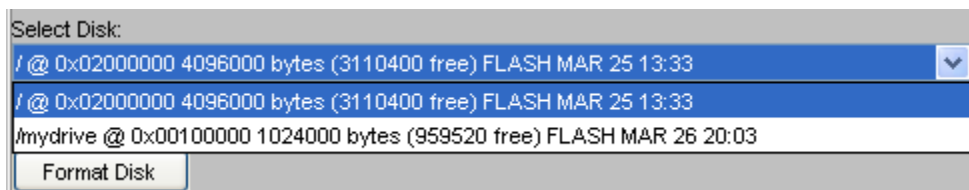
The warning dialog will once again appear, warning that there may be up to 20 seconds delay prior to completion. This occurs when it is necessary to re-format the newly mounted drive. Note that a tan colored area may appear during formatting, where the warning dialog once appeared, this is normal:



Upon completion, the acknowledgement dialog box will appear saying the update is successful, click OK. If you reference the File System Summary window you will now notice that your new drive appears and is available for access:



Also note that the drive is available within the un-mount combo box:



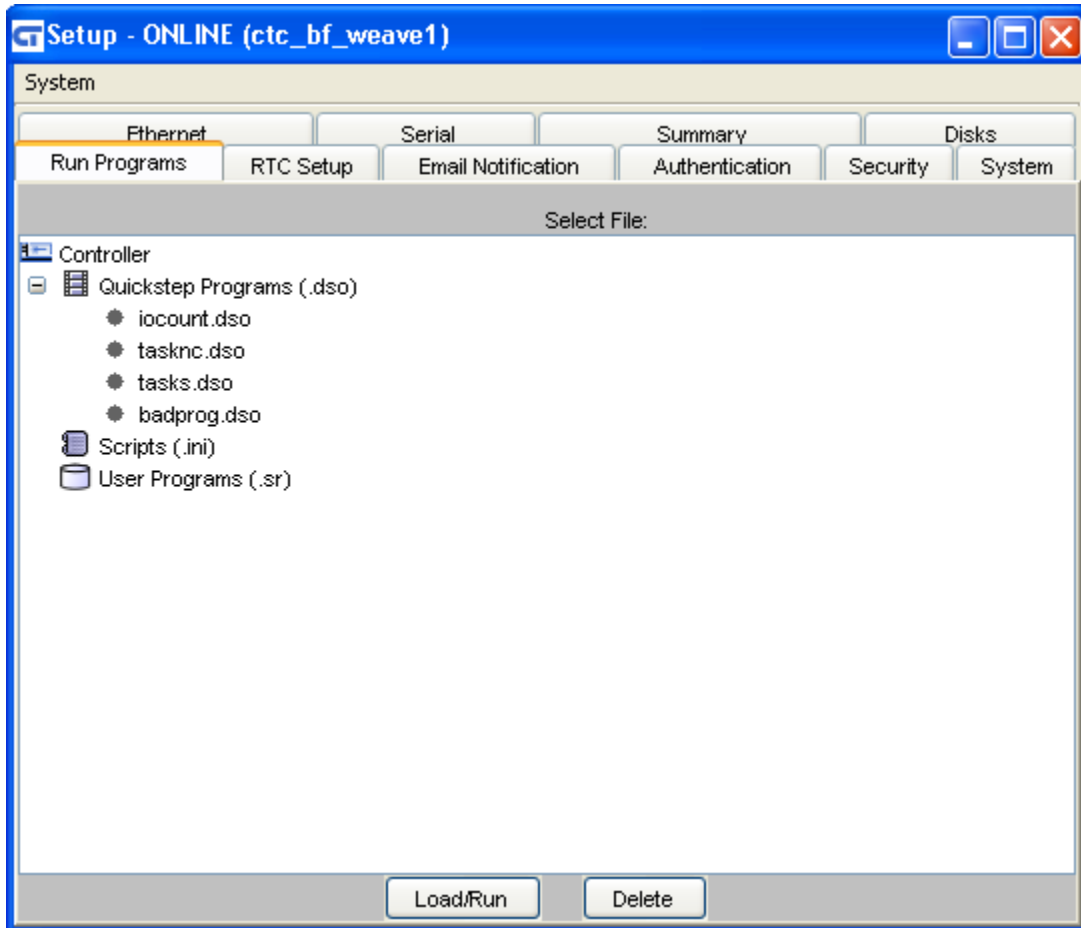
## WebMON 2.0 User's Guide

Opening an ftp session and doing a 'dir', directory, will also show how the drive appears to a program. It simply looks like a sub-directory off of root:

```
/>dir  
drw-rw-rw-  1 owner group 000256 MAR 25 13:33 _system  
drw-rw-rw-  1 owner group 959760 MAR 26 20:03 mydrive
```

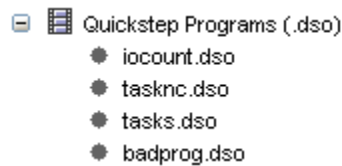
## Run Programs Tab

The “Run Programs” tab allows programs, which are capable of being executed, to be loaded and run immediately or removed from a mounted disk. Programs must reside in certain predefined directories, as detailed below. Each file that exists will appear within a tree structure:

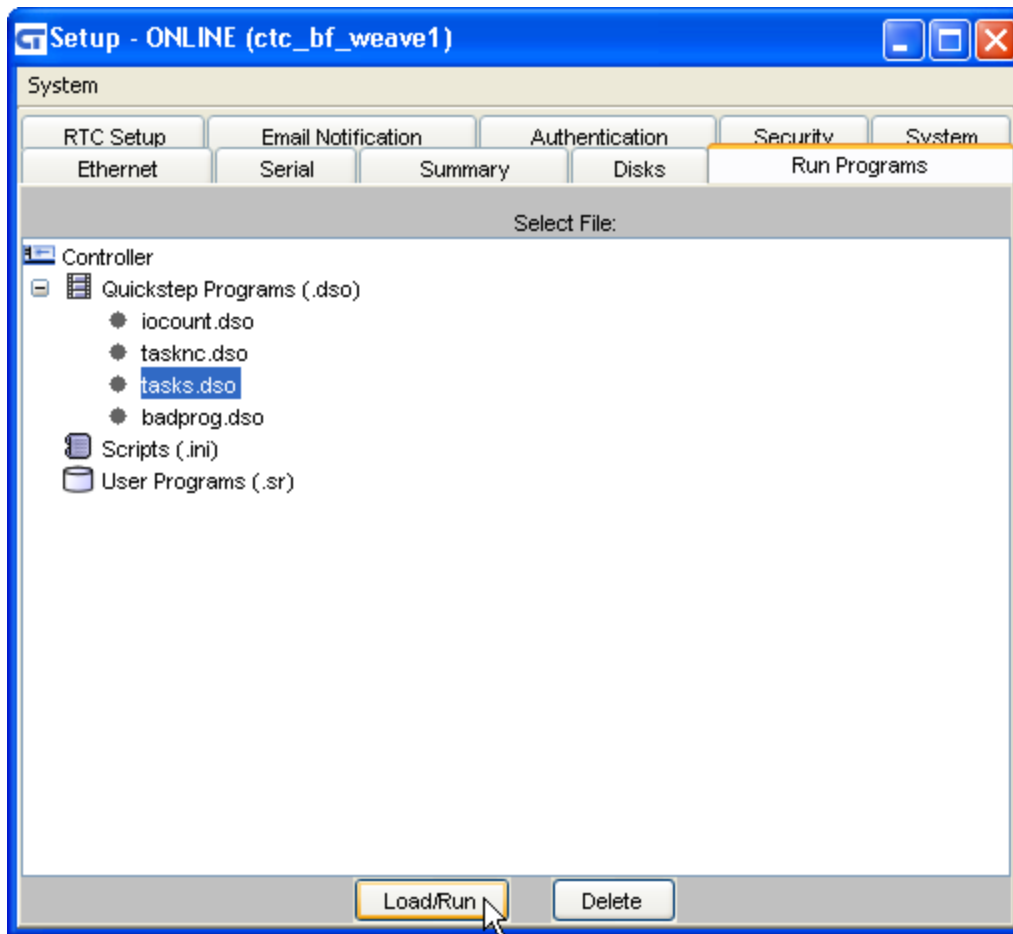


### Quickstep Programs (.dso)

Quickstep programs must be stored within the `/_system/Programs` subdirectory. These programs are the compiled output generated by the Quickstep Development Program (.dso file extension). Those available within the sub-directory will be listed as a node in the tree structure:



To execute a program simply select it (highlight), then select the “Load/Run” button. The screen below shows the selection of the Quickstep program file, tasks.dso for execution:



To permanently delete the file select the “Delete” button.

Note that only one Quickstep Program can be run at a time. Selecting a new one will terminate and unload the existing program

### Scripts (.ini)

Script programs must be stored within the `/_system/Scripts` subdirectory, on the main disk drive. These programs are the text files which contain commands as detailed within the “*Script Language Guide*” (950-520002). The file extension for these files is “.ini”. Each available script file will be listed as a node in the tree structure. To run the script simply select it (highlight), then select the “Load/Run” button. To permanently delete the file select the “Delete” button.

### User Programs (.sr)

User programs must be stored within the `/_system/Programs` subdirectory, on the main disk drive. These programs are the compiled output of “C” user programs, the Motorola S record format text files as defined within the “*C User Programmers Guide*” (950-

## WebMON 2.0 User's Guide

520004). The file extension for these files is “.sr”. Each available program file will be listed as a node in the tree structure. To load a User Program for execution simply select it (highlight), then select the “Load/Run” button. To permanently delete the file select the “Delete” button. Note that only one User Program can be run at a time. Selecting a new one will terminate and unload the existing program.

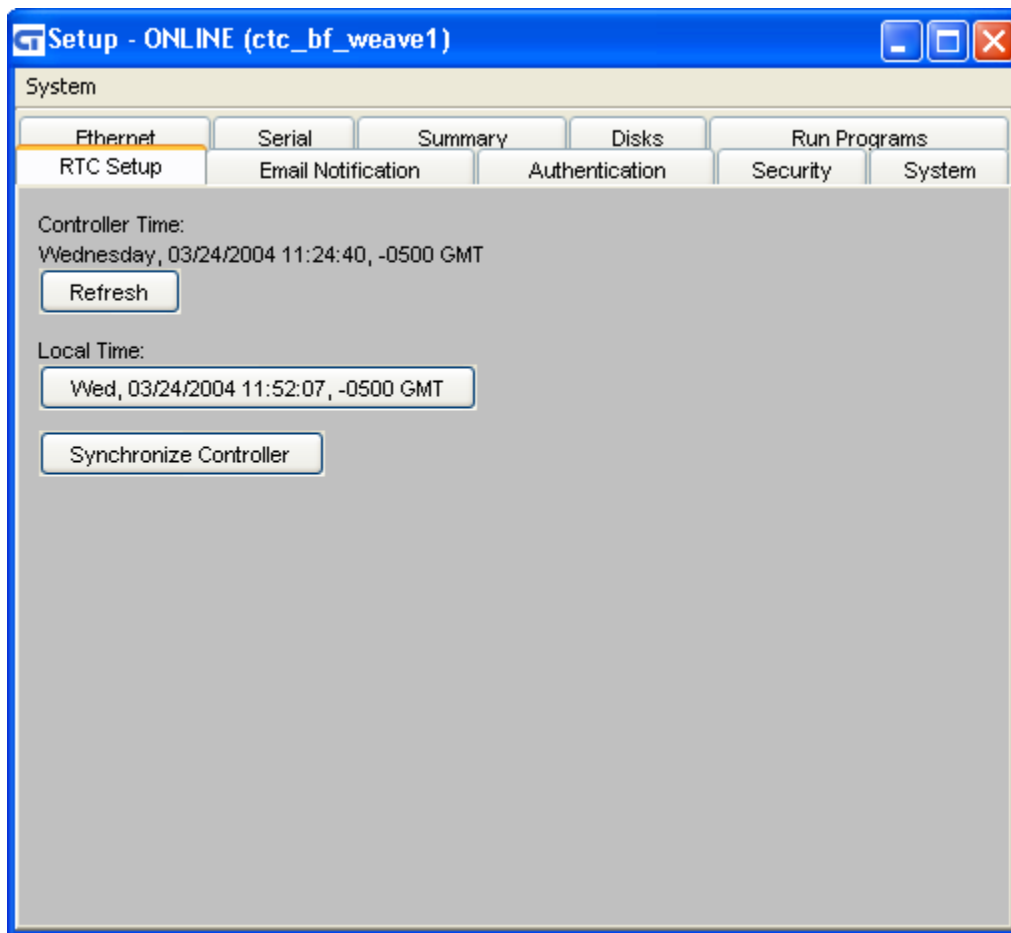


## RTC Tab

The “RTC” tab allows the user to synchronize the controller’s internal clock with the clock of the computer that is running WebMON. This includes the Greenwich Mean Time offset (time zone). Typically SNTP would be used but in some situations a time server may not be available and direct synchronization is required.

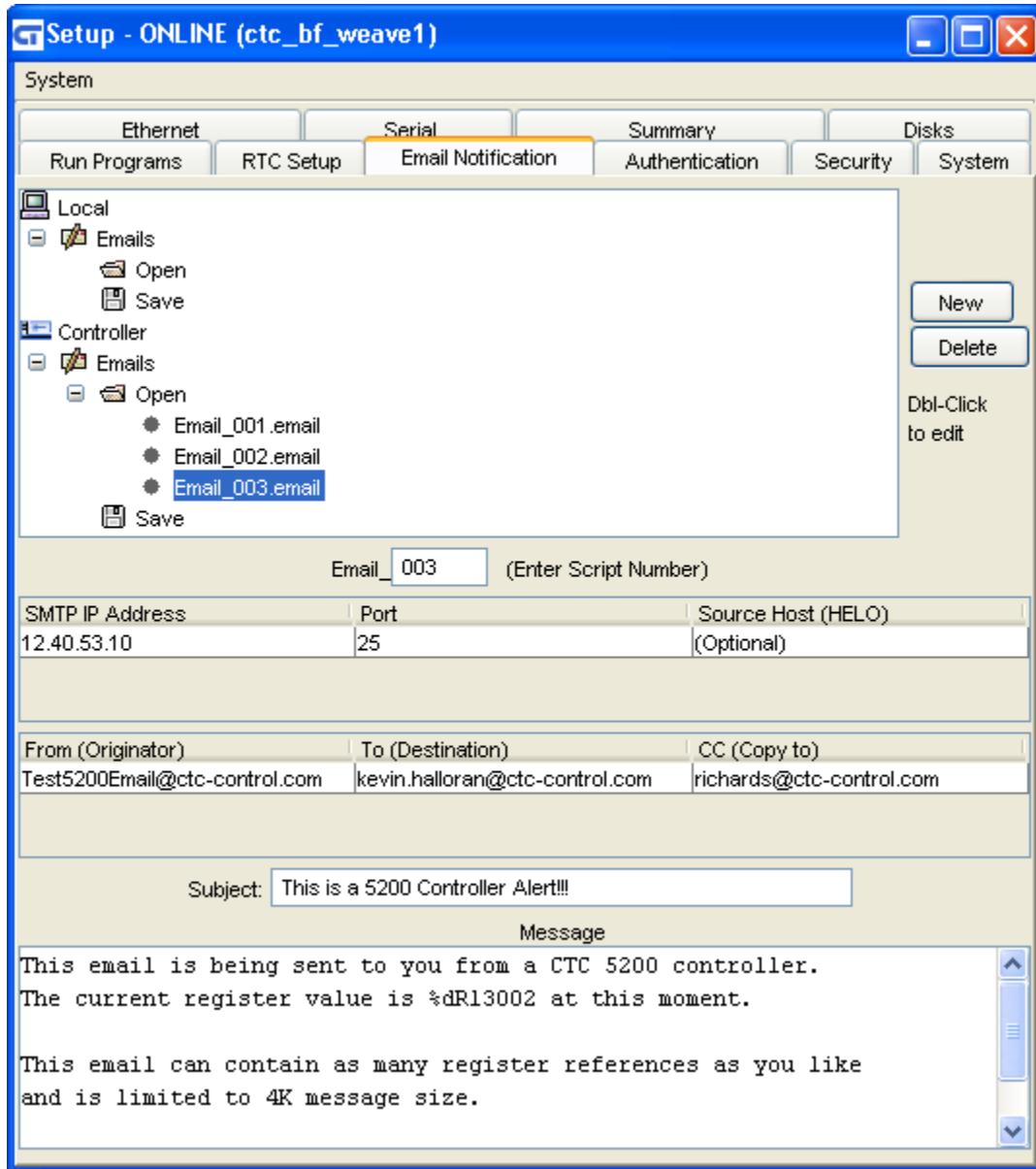
## Time Synchronization to PC

Time synchronization is a simple procedure. The current time of the controller is shown as the top line. To update that time simply select the “Refresh” button. The local time is derived from the computer running WebMON and is constantly updated on the display. To perform synchronization select the “Synchronize Controller” button below the “Local Time” display line, bottom most button.



## Email Notification Tab

The controller is capable of sending formatted email under the control of a Quickstep program or remote communications access a data register. Writing the desired email number to register 12317 accomplishes this. The email number references the sequence number in the file name Email\_###.email, where ### is 001 to 999. Each of these files reside within the /\_system/Emails subdirectory. The "Email Notification" tab can be used to automatically create, edit, and delete these files.

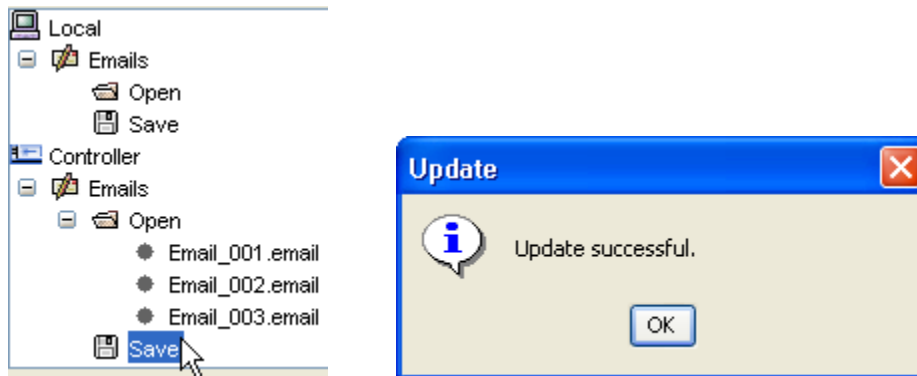


## Tree View, Local/Controller

At the top of the Email Notification tab is a tree list. This list is used to access formatted email files either locally or stored on a controller disk. Local->Email references the local disk drive of the computer running WebMON. Selecting Local->Email->Open will cause a dialog box to open and the selection of any email file for editing purposes.

Selecting Local->Email->Save will cause a dialog box to open and an email that is within the form at the bottom, to be saved to the computer's hard disk.

Files that exist within the controller's disk may be individually viewed and selected from the Controller->Email->Open tree node. Each file represents an individual node. To save a file that is created using the email template (form below the tree view), simply double click the Controller->Email->Save node. The file will be saved and named using the Script Number defined within the email template, Email###.email.



### Creating/Editing New Email Template

To create a new email, simply select the “New” button to the right of the Email Notification Tree view. This will cause all existing information to be removed from the template form and defaults to be entered. Alternatively an existing email could be loaded and modified as desired, then saved.

A number of data entry fields are available to define the email to be sent by the controller. The top most field, immediately below the tree view, allows the entry of a numeric from 1 to 999. This will become the file sequence number used within the email file name, Email###. Leading 0's will automatically be provided.

Email\_  (Enter Script Number)

The next set of data entry fields is a table whose row defines the SMTP server that is to be used for sending email. Each email may use the same and/or different SMTP servers. Make sure you are authorized for using the server and you are not attempting to relay. Relaying is restricted and occurs when you try to copy an email to someone that is not authorized, outside your domain. For example if the domain was ctc-control.com, you would not be able to send a copy of the email to hotmail.com, using POP3. Mail Servers can be configured to allow for exceptions, if desired. A typical way around this would be to use a distribution list within your mail server, that in turn sends outside the domain.

Available data entry parameters for the first table are:

SMTP IP Address	Port	Source Host (HELO)
12.40.53.10	25	(Optional)

**SMTP Server**

This is the server IP address of the server handling your email account. It is typically within the same domain as your 'From:' email address. The "dot" notation format is used.

**Port**

The standard SMTP port used is 25; it may be changed here if desired. This is the port the SMTP server will be listening on for connection requests.

**HELO**

This is an optional field which can be used to report your domain within the email. It is required by some hosts. For example the domain of [www.ctc-control.com](http://www.ctc-control.com) would be ctc-control.com.

The second table is used to define who the email is from (FROM:), who it is to be sent to (TO:), and who it is to be copied to (COPY:). Only one address is supported per entry. If larger distributions are required it is suggested that a Distribution List be created on the Email server.

The required format, of each email address, is [person@domain.com](mailto:person@domain.com). Enter each as needed. Note the COPY field is optional:

From (Originator)	To (Destination)	CC (Copy to)
Test5200Email@ctc-control.com	kevin.halloran@ctc-control.com	richards@ctc-control.com

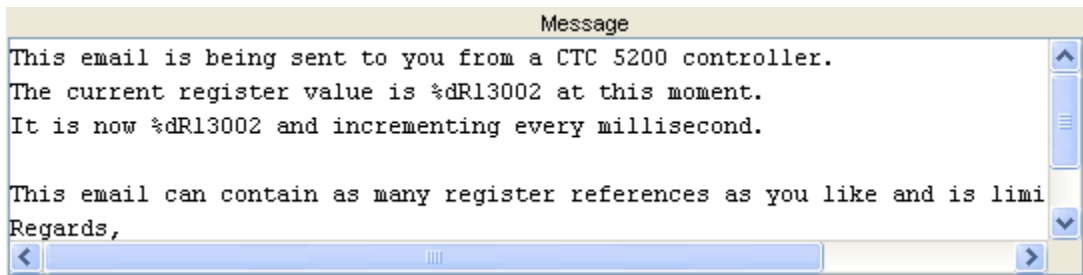
**Subject**

The Subject line will appear as the summary in an email message. Enter any desired text:

Subject:

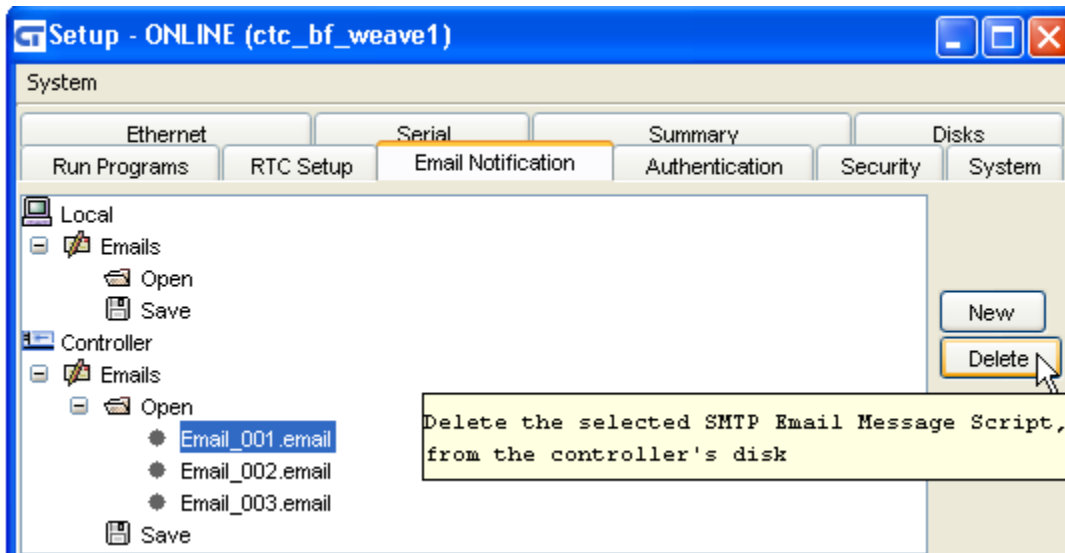
**Message**

The Message area can contain up to 4K bytes of data. Messages may be any mix of normal text characters and references to Controller registers. Registers are references using "C" style printf directives. For example, to reference the 13002 register and have its contents placed in a message a %dR13002 would be used, optionally %05dR13002 would force at least 5 characters wide with leading 0's as a filler. In printf notation %d is decimal, %x is lower case hex, and %X is upper case HEX. These are the only acceptable printf syntaxes currently supported in email messages. Below shows an example of a message which would include the current value of the 13002 register, when sent:



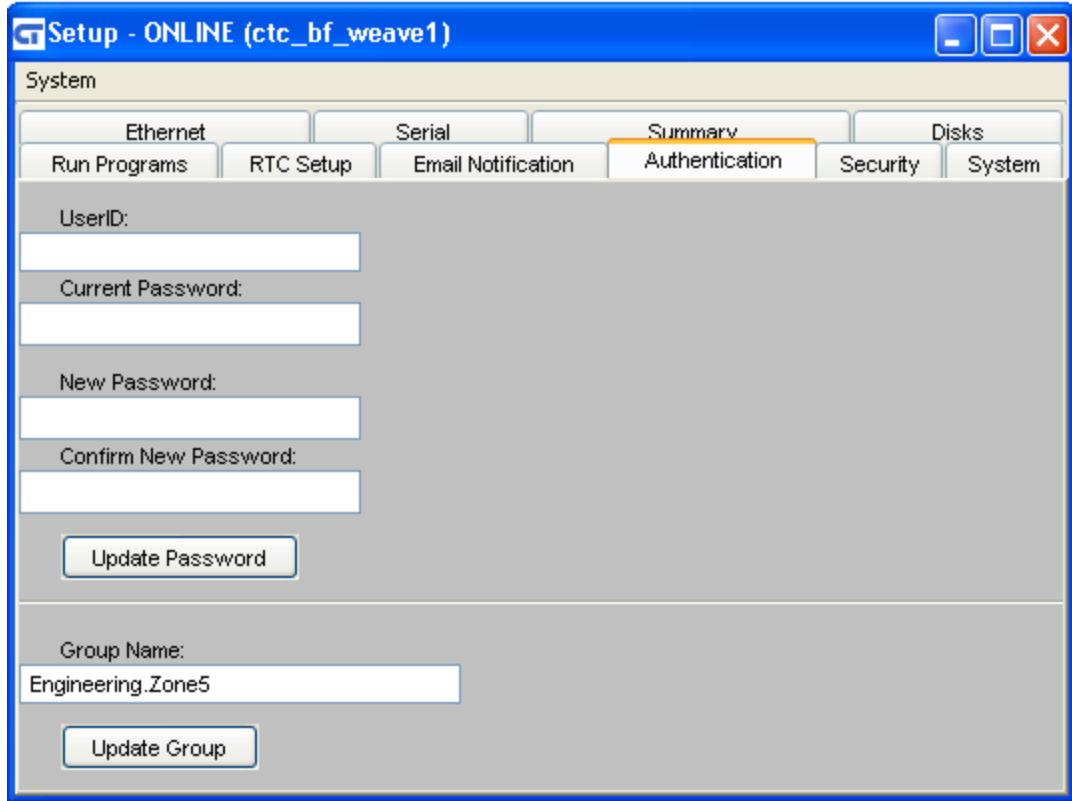
## Deleting Email Template

Deleting an email is only supported from a controller disk. To delete a file use the Controller->Email->Open tree view to list the available files. Highlight the one desired and select the “Delete” button. The file will be deleted and the tree updated.



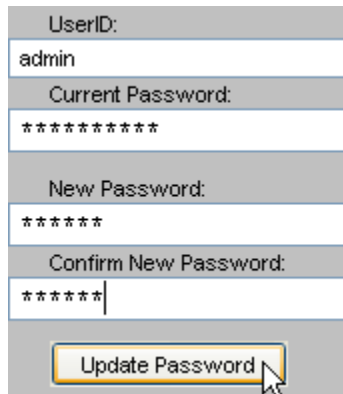
## Authentication Tab

The “Authentication” tab is a multi-function tab. It is used to change the existing password, assign Controller groups, restore factory defaults and hard reset the controller.



### Changing Password

The first section on the tab allows the user to change the current password. To do this first enter the User ID, currently “admin” in all controllers. Next, the current password is entered, followed by entering the new password twice. The second entry of the new password is simply for verification purposes (“Confirm New Password”).



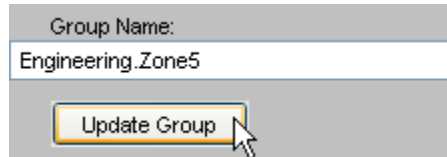
Select the “Update Password” button to immediately make the changes active.

### Group Name

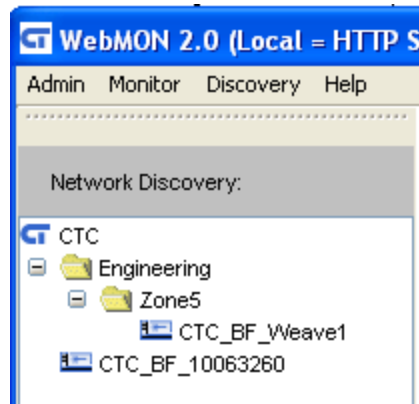
The Group Name may be optionally assigned to a controller. It is used to create a tree structure of controllers as they appear on a network, during Network Discovery. Refer to Chapter 9 for a discussion of Network Discovery. In short, controllers may be grouped up to 10 levels deep in a hierarchy within the configuration of a system. Periodically

## WebMON 2.0 User's Guide

WebMON can send a broadcast to automatically locate controllers on the network. Instead of just listing them all in one potentially confusing list, controllers can be assigned to folders, much like the directory structure of your hard disk. In the example below we assign the controller to the group Engineering.zone5, each period separates the group name. If it does not exist it will automatically be created.



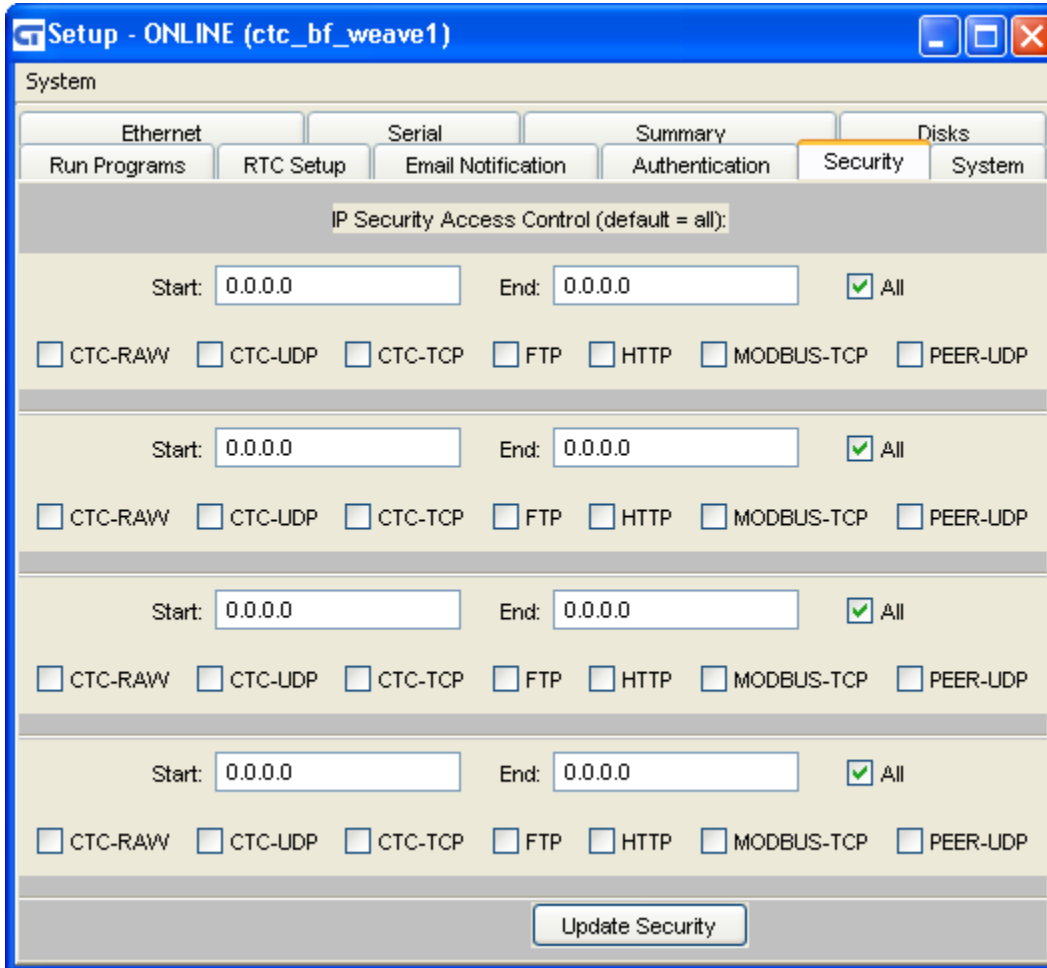
As discussed in Chapter 9, this is how it would appear when viewed from a tree structure:



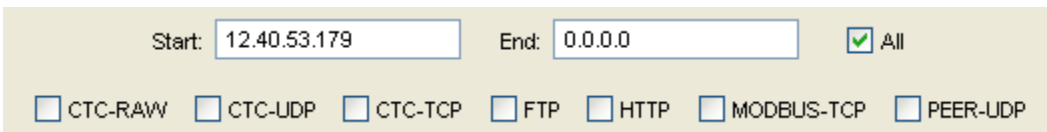
Note that the controller CTC\_BF\_10063260 does not belong to a group. Any number of controllers can be in a single group, with the default being none or the root directory when viewed as a tree.

## Security Tab

The “Security” tab is used to restrict access to the controller to only certain IP Addresses and/or ranges. Up to 4 sets of IP data may be entered each limiting which communication protocols will be allowed. By default all protocols are enabled for all IP Address:



To restrict to a single address enter only that address and leave the ‘End’ either blank or with 0.0.0.0. Below is an example that only allows IP Address 12.40.53.179 to access the controller for all protocols. No others would be allowed access:





## WebMON 2.0 User's Guide

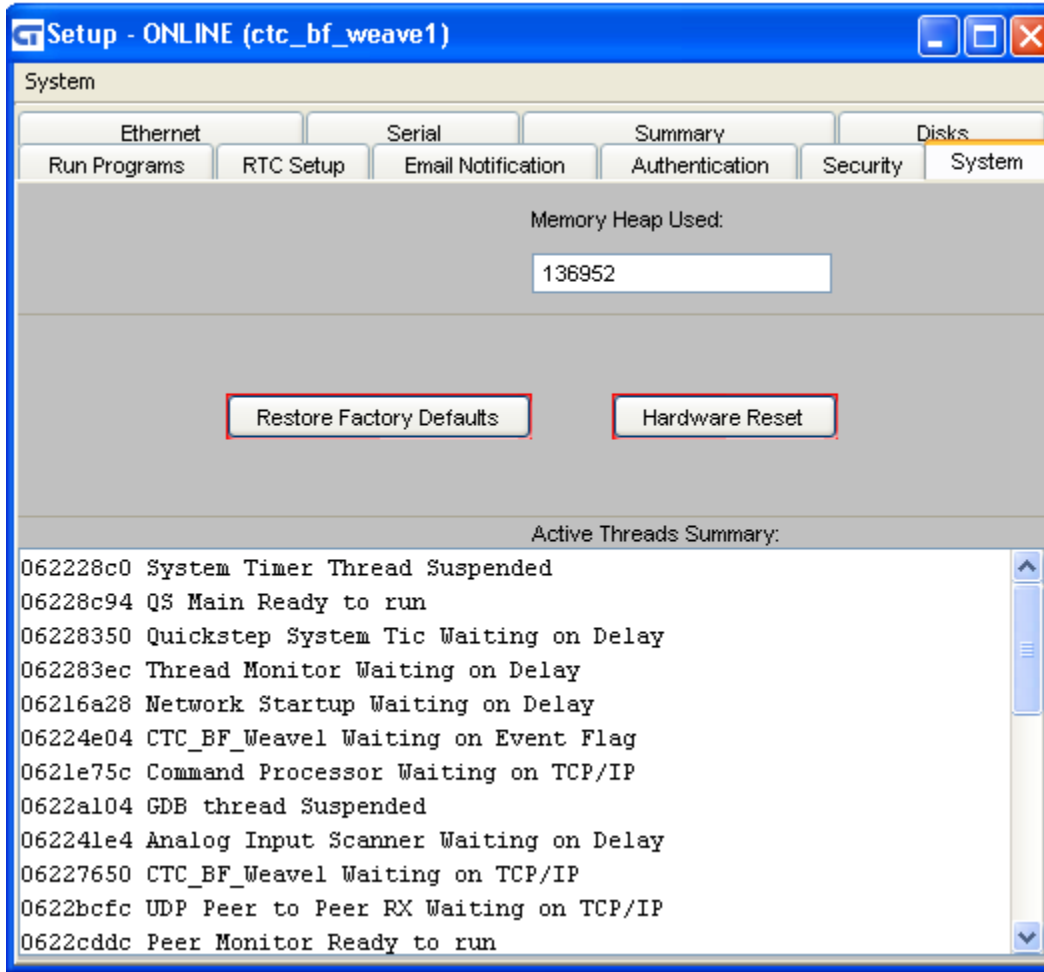
To allow 12.40.53.179 to access all protocols, but have all others restricted to non-administrative protocols:

Start: 12.40.53.179	End: 0.0.0.0	<input checked="" type="checkbox"/> All				
<input type="checkbox"/> CTC-RAW	<input type="checkbox"/> CTC-UDP	<input type="checkbox"/> CTC-TCP	<input type="checkbox"/> FTP	<input type="checkbox"/> HTTP	<input type="checkbox"/> MODBUS-TCP	<input type="checkbox"/> PEER-UDP
<hr/>						
Start: 12.40.53.1	End: 12.40.53.255	<input type="checkbox"/> All				
<input checked="" type="checkbox"/> CTC-RAW	<input checked="" type="checkbox"/> CTC-UDP	<input checked="" type="checkbox"/> CTC-TCP	<input type="checkbox"/> FTP	<input type="checkbox"/> HTTP	<input checked="" type="checkbox"/> MODBUS-TCP	<input checked="" type="checkbox"/> PEER-UDP

Note that the enable all IP address must be entered first in the list otherwise the second rule listed would have taken precedence.

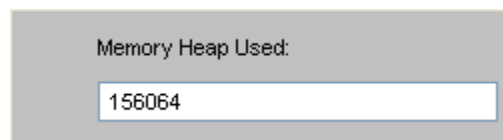
## System Tab

The “System” tab is used to display the running state of each Thread on the controller along with memory usage. Utility buttons are provided to also restore factory defaults and provide a hard system reset.



## Memory Utilization (Heap)

Memory Utilization determines the amount of memory that has been dynamically allocated by the controller since power up. Each task, TCP socket, communications protocol, etc., consumes memory as it is used. Up to 500K is available.



## Restoring Factory Defaults

Should problems develop and/or bad parameters ever be set sometimes it is useful to restore the system to factory defaults prior to making any new changes. Selecting this

button will restore all factory defaults and delete any 5100.ini file which may be loaded to override parameters. Reference the “Script Command Guide” for details of the 5200.ini file.

Restore Factory Defaults

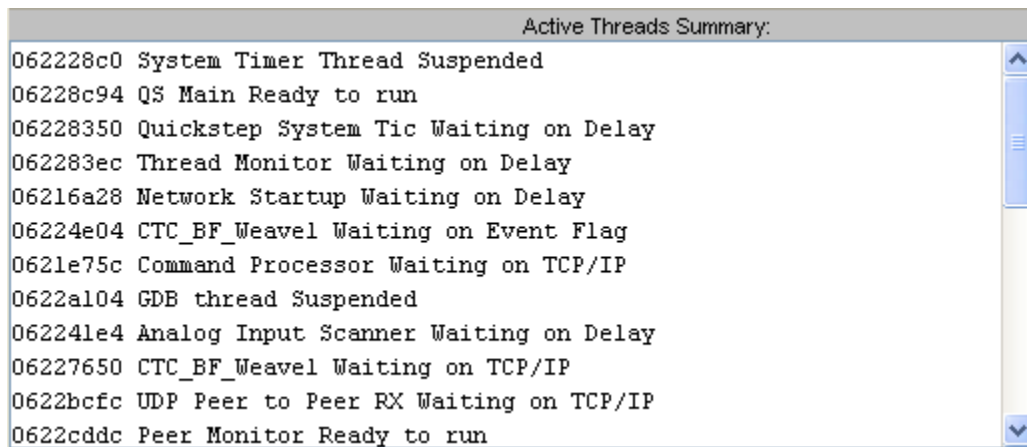
### Resetting Controller (Hard Reset)

Although not normally required, you may totally reset your controller by triggering its watchdog timer. This will create a hardware reset, typical of that occurring at power-up. To perform a hardware reset, select the “Hardware Reset” button:

Hardware Reset

### Threads

A Thread is a separate software process running in its own environment, parallel to other processes. Each thread has its own stack but shares memory space. Think of it as multiple tasks all being allowed to share the processor, running when needed. Quickstep runs as a single thread, within which it runs Quickstep program tasks.



```
Active Threads Summary:
062228c0 System Timer Thread Suspended
06228c94 QS Main Ready to run
06228350 Quickstep System Tic Waiting on Delay
062283ec Thread Monitor Waiting on Delay
06216a28 Network Startup Waiting on Delay
06224e04 CTC_BF_Weavel Waiting on Event Flag
0621e75c Command Processor Waiting on TCP/IP
0622a104 GDB thread Suspended
062241e4 Analog Input Scanner Waiting on Delay
06227650 CTC_BF_Weavel Waiting on TCP/IP
0622bcfc UDP Peer to Peer RX Waiting on TCP/IP
0622cddc Peer Monitor Ready to run
```

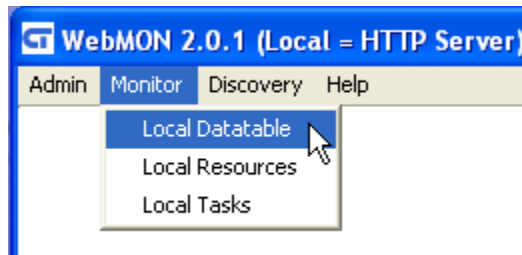
*Blank*

## Monitor – Data Table

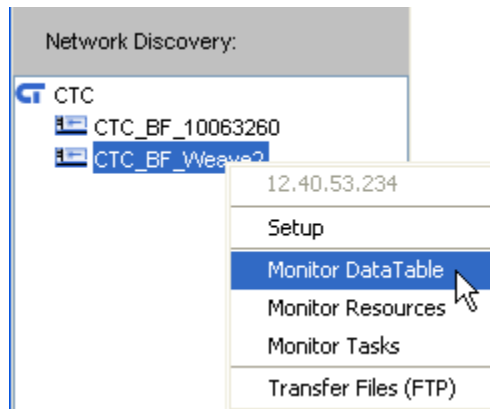


Each controller may have a single data table available for Quickstep access. This table may be a maximum of 256 columns wide and 32767 rows deep. The total of row times columns may not to exceed 32767. WebMON will allow you to view a sliding window into the data table and dynamically update the view in real time.

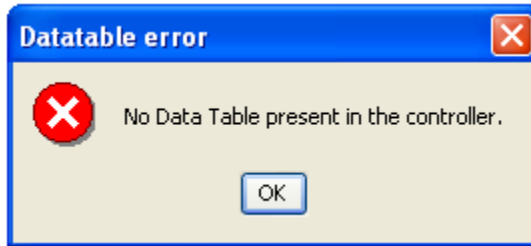
The data table monitor may be invoked from either the main menu or a Network Discovery node menu (refer to Chapter 9). When using the main menu select “Local Datable”:



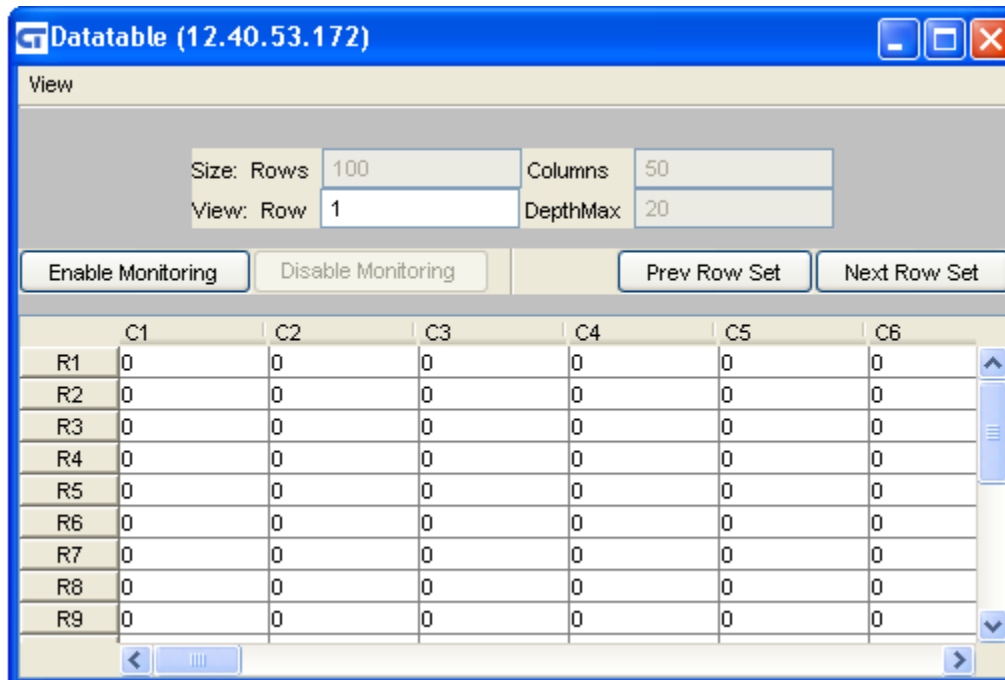
Or if Network Discovery, Monitor DataTable:



Upon establishing a connection with the controller WebMON will determine the size of the data table available. If none is present a dialog will be displayed and the data table will not load:



If the data table exists, something similar to below will appear. In this example there are 100 rows and 50 columns:



The first set of information is the Size of the entire data table within the controller:



The next set of data fields allows adjustment of the view window (View):

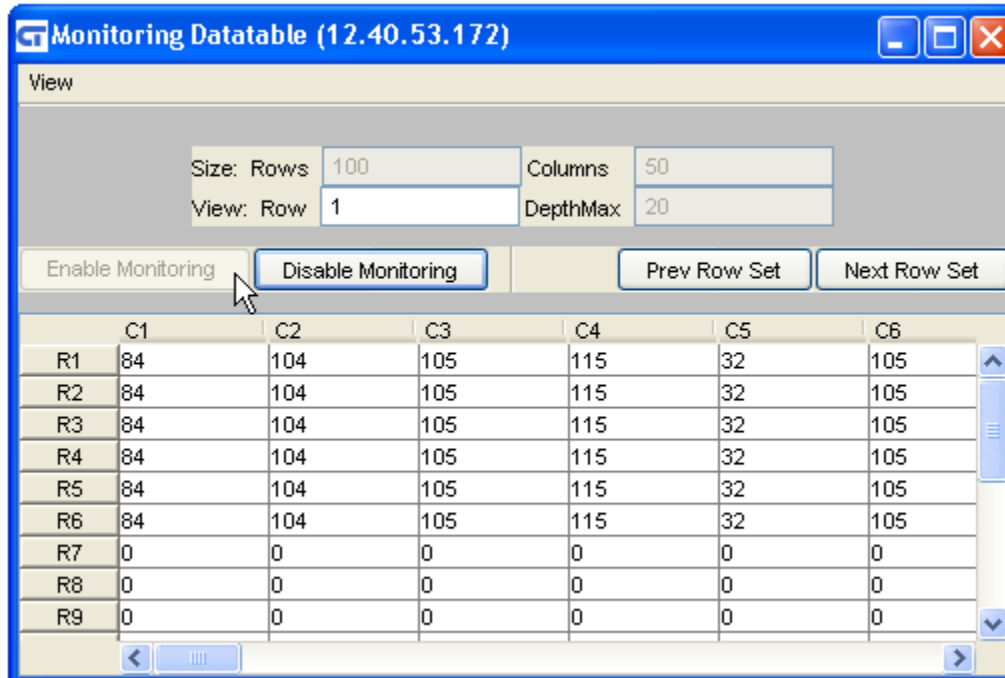


A default view size will be presented and may be modified as desired. Note that the larger the view window the more overhead involved in communication with the controller and transferring cell data. It is best to limit the view size. The view window

may be paged up and down through the data table by using the “Prev Row Set” and “Next Row Set” buttons:



Once the view window is adjusted as desired you may enable real time updates from the controller. Selecting the “Enable Monitor” button does this. When active monitoring this button will dim and the “Disable Monitor” button will be available.



The data will appear in the data grid window. Each row and column is labeled and may be scrolled left/right for viewing additional columns. To modify a value simply double-click a cell and enter the new value. You may hit the Enter key to accept the new value or select a different cell. The value is written immediately to the controller. Below shows changing the data from a 105 to a 106:

First double click the cell:

R1	84	104	105	115	32	105
----	----	-----	-----	-----	----	-----

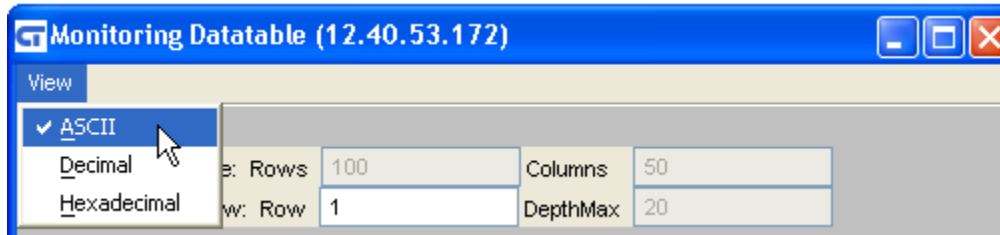
Next backspace and enter the new data:

R1	84	104	106	115	32	105
----	----	-----	-----	-----	----	-----

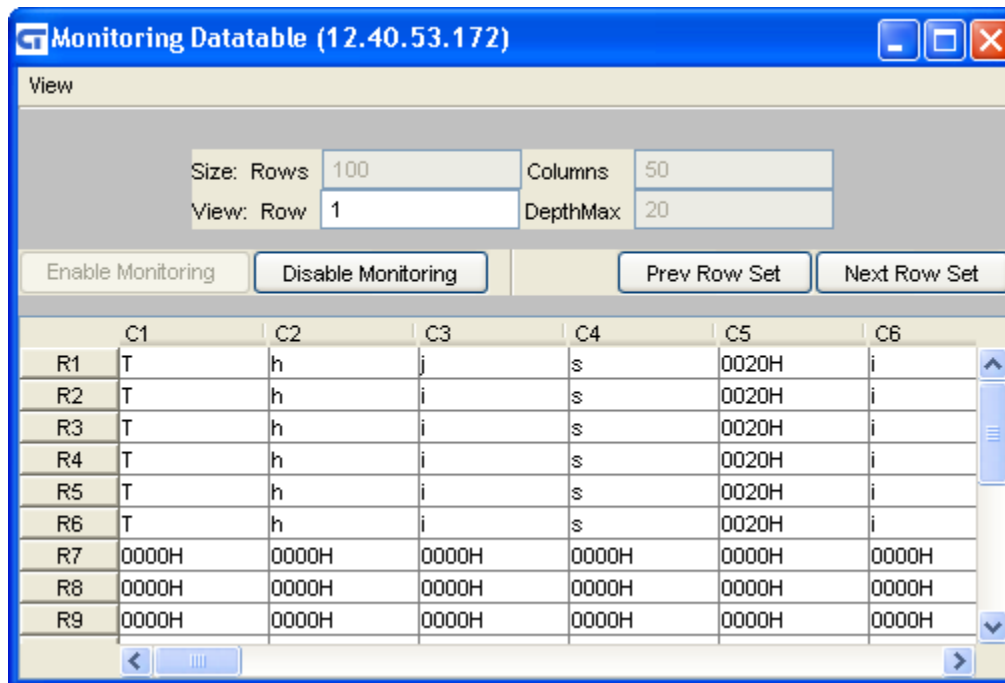
Now hit the Enter Key or click another cell, data is written to the controller when you leave the cell:

R1	84	104	106	115	32	105
----	----	-----	-----	-----	----	-----

In the above example we displayed the data as decimal values. In some cases it may be desirable to reference the data as ASCII or even Hexadecimal (base 16). This is done by selecting the 'View' memory item and then selecting the desired view type. The next time the cell is updated with information it will be changed to that requested:



Notice that the data in the cells is now viewable as ASCII text:



Note the 0020H and 0000H. Data that is not a viewable ASCII character will have its Hex representation shown.

To disable monitoring, simply click the 'Disable Monitoring' button. To exit click the red 'X' at the top right of the window.



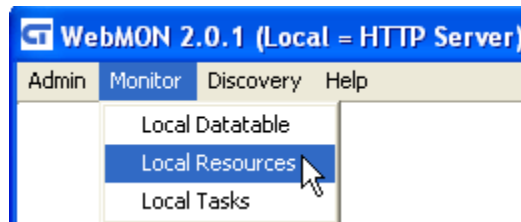
**CHAPTER**  
**6**

## Monitor - Resources

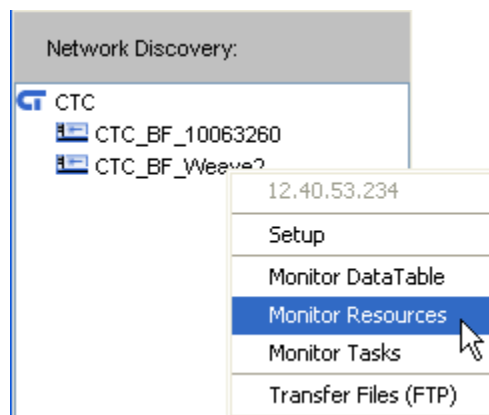


WebMON provides the ability to monitor, in real-time, resources within the controller. These resources may be any combination of data registers and/or input/outputs, both digital and analog. Formatted screens are available to lead the user through resource selection as well as presentation of data as it changes dynamically. Symbols may be loaded from Quickstep programs to automatically initialize the Monitor resources to be viewed. In addition numerous monitor sessions may be established with either the same or differing controllers, along with the ability to save configurations to a controller and/or local computer, for later recall.

The Resources Monitor may be invoked from either the main menu or a Network Discovery node menu (refer to Chapter 9). When using the main menu select “Local Resources”:

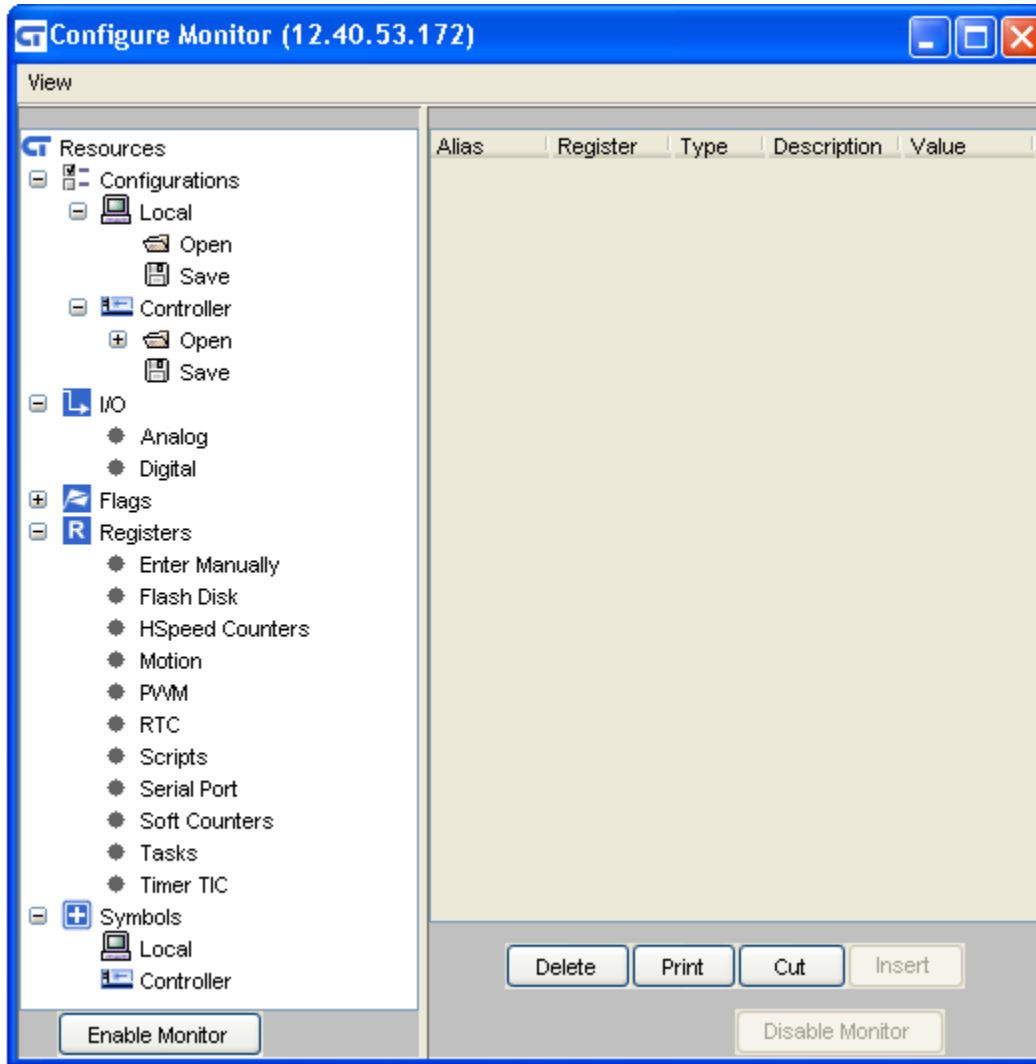


Or if Network Discovery, Monitor Resources:



## Resource Selection

The Monitor screen consists of two halves when first invoked. The left half is a tree list of available selections with the right half being those that have been selected for viewing:



When first invoked the view window on the right will be blank and must be filled in either with existing information (Configurations), or new information. To do this, add information using the tree selections available on the left view window. For example, if adding "Timer TIC", double click it and the following is added to the right view window:

Resources	Alias	Register	Type	Descrip...	Value
Configurations		13002	TIC	Timer Tic	?

Note the column headers:

Alias	Register	Type	Description	Value
-------	----------	------	-------------	-------

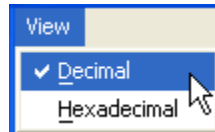
**Alias** – Alias is used to display a symbol reference to a particular resource. This can be from your program (loaded via a symbol file) and/or manually entered by double-clicking on the cell.

**Register** – The register number being monitored.

**Type** – Type of resource being monitor. For Example Soft Counter, digital input, etc.

**Description** – Useful information about the resource being monitored.

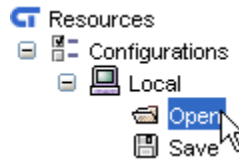
**Value** – The current value if being monitor (updated about every 2 seconds). The '?' designates an undefined resource or one that has never been enabled for data collection. If monitoring has been disabled the last valid value is displayed. The 'Value' field may be displayed in decimal or hex. Each of which is selectable via the Monitor menu:



An additional option allows for the sorting of data. Simply click the column header you wish to sort by and each click will change the sort from ascending/descending to descending/ascending order.

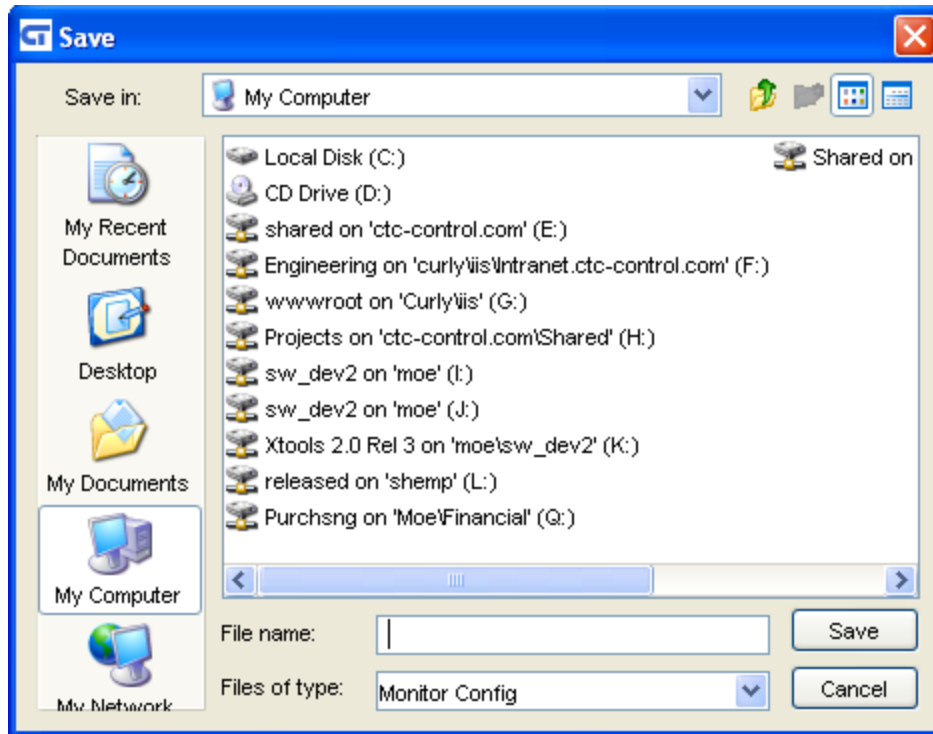
## Configurations

Resources that are selected for viewing, and placed on the right hand side, may be saved either locally or to the controller, for later recall. Each configuration saved will retain the resources selected, any alias's created (resource naming), and displayed resource values at the time it was saved. Selection is made by double-clicking one of the following tree links:



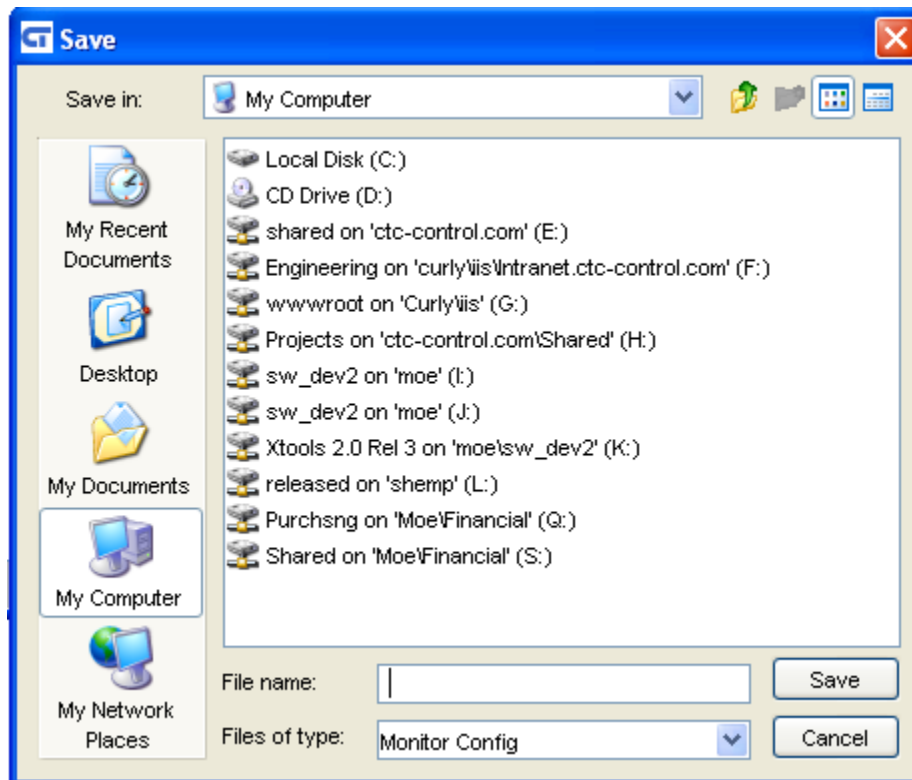
### *Local->Open*

Local Configurations references those resources that are on your computer. Selection of this tree item will cause a dialog box to appear requesting the entry of any local configurations, files ending with .cfg:



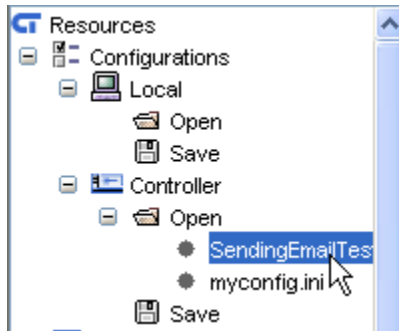
**Local->Save**

Selection of this tree item will cause a dialog box to appear requesting where to store the current resource configuration, on your local computer, and what .cfg file name to use. This file may be reloaded using the Local->Open tree selection.



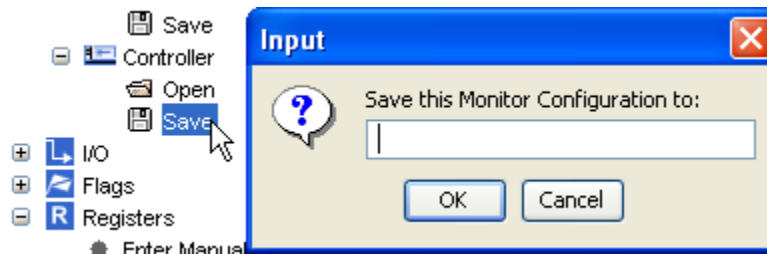
***Controller->Open***

Selection of this tree item will cause a list of available .cfg, Monitor configuration files, to appear within the tree view. These are configurations you have created previously and saved. Open allows you to load them back into the Monitor form in the same state they were when saved, including any collected data. Double-clicking one of these entries will cause that configuration file to be loaded into the right half of the Resource Monitor View Window:



***Controller->Save***

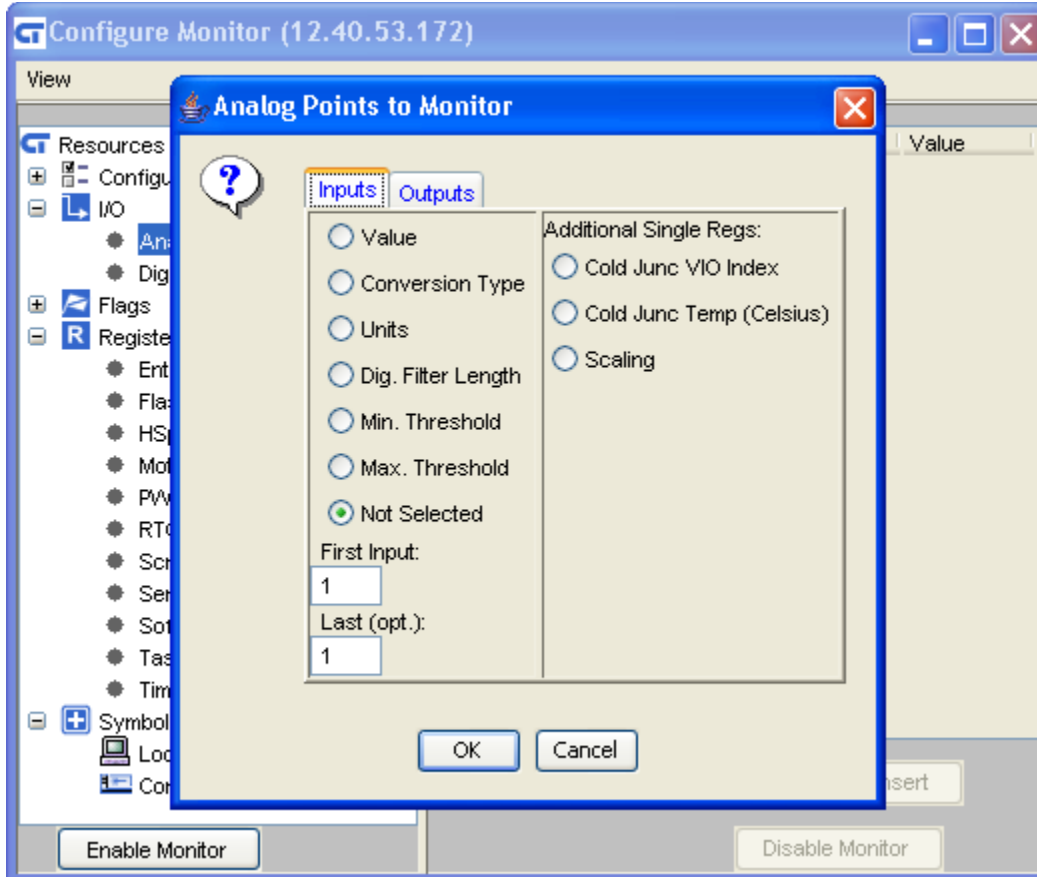
Selection of this tree item will cause a dialog to appear, requesting the name to use when saving the current Resource Monitor information to the controller. Once entered, a new file will appear on the controller in the /\_system/Diags sub-directory. It will also appear in the Controller->Open list. Double click Controller->Save and you will be prompted for the name to use when saving the configuration:



## I/O

The I/O tree selection has two sub-directories, Analog and Digital. Double-clicking either node will cause available selections in that category to appear, as a dialog box.

### Analog Input



Available selections are:

**Value** – The actual adjusted analog input value which would appear in registers 8501 to 8756.

**Conversion Type** – This selection is used to monitor the type of thermocouple conversion algorithm to run on the raw Analog Input Value, prior to presenting the adjusted value in the 8501 series registers. This returns references registers 9001 to 9256. It is read/write and defined as below:

Value	Description
<b>0</b>	<b>No Conversion</b>
<b>1</b>	Cold Junction Temperature Algorithm
<b>10</b>	K-Type Thermocouple Linearization Algorithm
<b>11</b>	J-Type Thermocouple Linearization Algorithm
<b>12</b>	T-Type Thermocouple Linearization Algorithm
<b>13</b>	E-Type Thermocouple Linearization Algorithm
<b>14</b>	R-Type Thermocouple Linearization Algorithm
<b>15</b>	S-Type Thermocouple Linearization Algorithm



**Units** – Modifies the ‘Value’ read to be based on the Units defined in the below table (R/W):

Value	Description
<b>0</b>	<b>+/- 10,000 (All modules, Also See Register 13010)</b>
<b>1</b>	+/- 10,000,000 (All modules, Also See Register 13010)
<b>2</b>	+/- 20,000 (alternate setting for +/- 20 mV, Modules G & Q)
<b>3</b>	+/- 100,000 (alternate setting for +/- 100 mV, Modules J & K)
<b>4</b>	4,000 – 20,000 (alternate setting for 4-20 mA, Modules H & R)
<b>11</b>	Celsius (for CJC and Thermocouple Conversions Only)
<b>12</b>	Fahrenheit (for CJC and Thermocouple Conversions Only)
<b>13</b>	Kelvin (for CJC and Thermocouple Conversions Only)

**Digital Filter Length** – Analog Input Digital Running Filter Length (Default = 1): R/W, range 1 to 255. References registers 18501 to 18756.

**Min. Threshold** - Analog Input Minimum Threshold Set Point: R/W. References registers 19001 to 19256.

**Max. Threshold** – Analog Input Maximum Threshold Set Point: R/W. References registers 19501 to 19756.

**Not Selected** – This is the default selection, meaning none of the above registers are selected for monitoring.

**First Input** – This is the first Analog of the above type to display for monitoring. The first available is 1, the last 256.

**Last (Opt.)** – Optional Input. This is the last Analog of the above type to display for monitoring. If this selection is set then it is assumed to be a series of incrementing Analogs, beginning with the “First Input” and ending with that designated by this value.

Additional Single Regs:

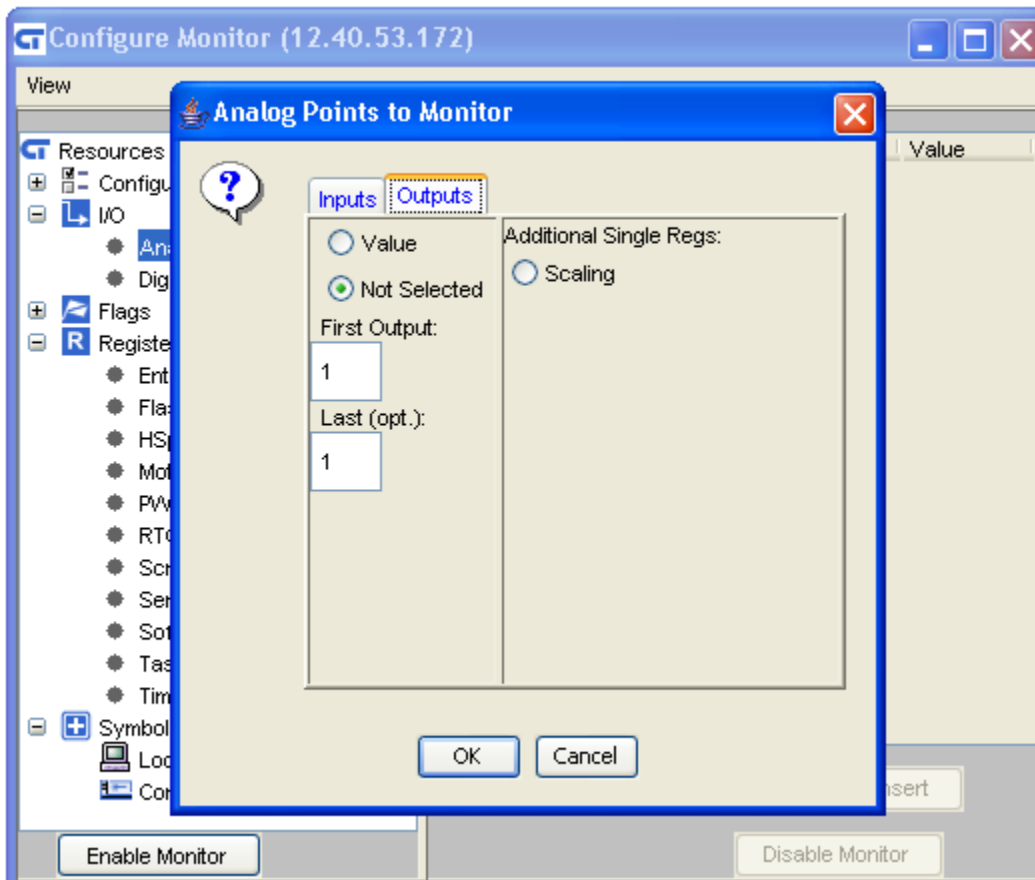


**Cold Junc VIO Index** – Register 9996, Analog Input Cold Junction Virtual IO index: R/W, 0 = local (default), references 21500 block.

**Cold Junc Temp (Celsius)** – Register 9997, Analog Input Cold Junction Temperature in Celsius: R/W. Default = 250 (i.e. 25.0°C)

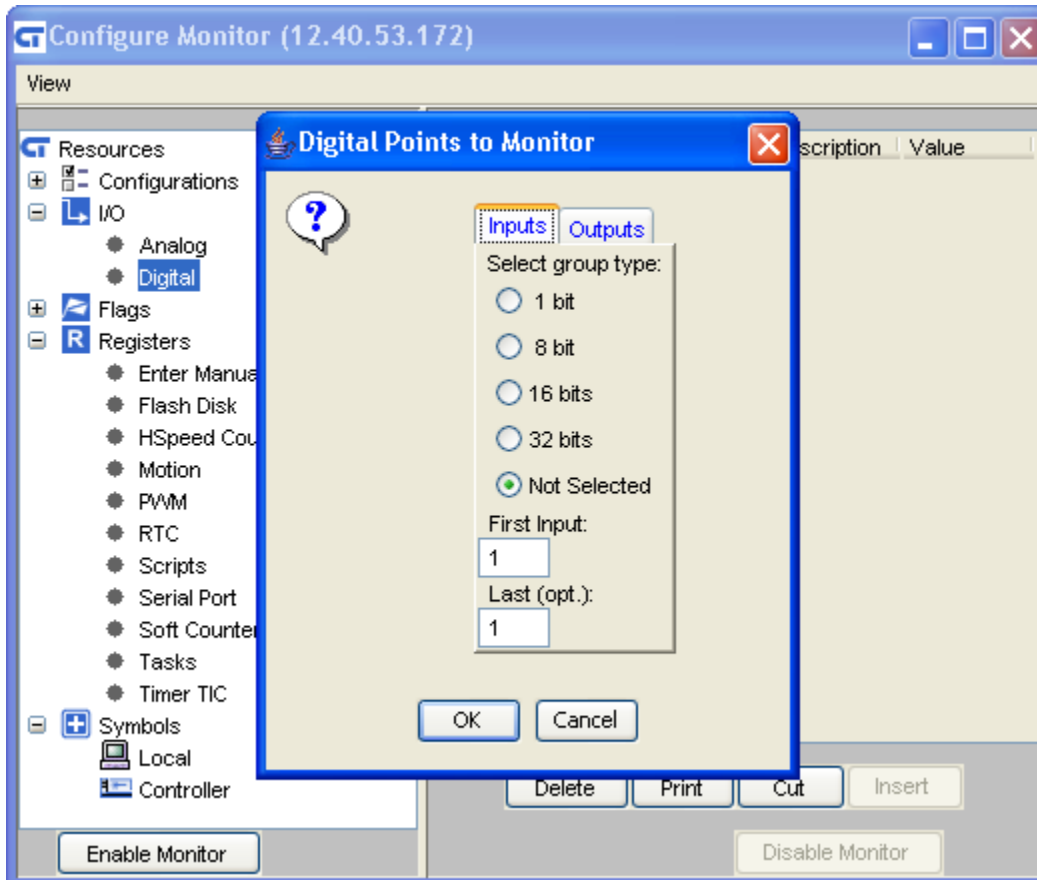
**Scaling** - Analog Input Scaling: R/W; 0 = uVolts (i.e. +/-10000000), default is 1 = mVolts (i.e. +/-10000).

**Analog Output**





**Digital Input**



2001-3024	Alternate Access to Inputs 1-1024: R only, 0 = open, 1 = closed.
11001-11032	Access Inputs as a 32-bit Number: R only
11101-11164	Access Inputs as a 16-bit Number: R only
11201-11328	Access inputs as a 8-bit Number: R only

Available selections are:

**1 Bit** – References monitoring a single digital input per view/update. Value will be 1 for on or 0 for off. Access to Inputs 1-1024: R only, 0 = open, 1 = closed. Registers 2001 – 3024.

**8 Bit** – References monitoring 8 digital inputs as one view/update. Value will be 0x00 to 0xff hex. Access inputs as a 8-bit Number: R only. Registers 11201-11328.

**16 Bits** – References monitoring of 16 digital inputs as one view/update. Value will be 0x0000 to 0xffff hex. Access inputs as a 16-bit Number: R only. Registers 11101-11164.

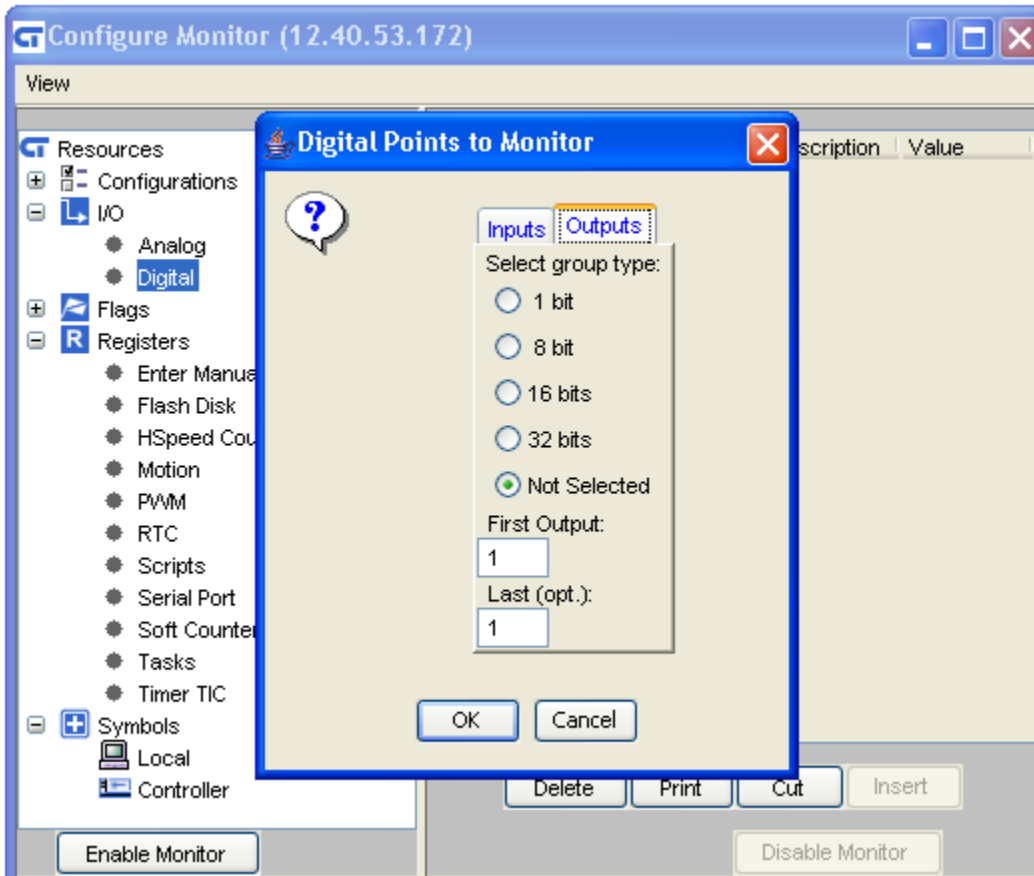
**32 Bits** – References monitoring of 32 digital inputs as one view/update. Value will be 0x00000000 to 0xffffffff hex. Access inputs as a 32-bit Number: R only. Registers 11001-11032.

**Not Selected** – This is the default selection, meaning none of the above registers are selected for monitoring.

**First Input** – This is the first Digital of the above type to display for monitoring. The first available is 1, the last 1024.

**Last (Opt.)** – Optional Input. This is the last Digital of the above type to display for monitoring. If this selection is set then it is assumed to be a series of incrementing Digital Inputs, beginning with the “First Input” and ending with that designated by this value. When grouped as bits greater than 1 the value will be round up to the next larger group.

**Digital Output**



1001-1999	Alternate Access to Outputs 1-999: R/W, 0 = off, 1 = on.
10001-10032	Access Outputs as a 32-bit Number: R/W
10101-10164	Access Outputs as a 16-bit Number: R/W
10201-10328	Access Outputs as a 8-bit Number: R/W

Available selections are:

**1 Bit** – References monitoring a single digital output per view/update. Value will be 1 for on or 0 for off. Access to Outputs 1-999: R/W, 0 = off, 1 = on. Registers 1001 – 1999.

**8 Bit** – References monitoring 8 digital inputs as one view/update. Value will be 0x00 to 0xff hex. Access inputs as a 8-bit Number: R only. Registers 10201-10328.

**16 Bits** – References monitoring of 16 digital outputs as one view/update. Value will be 0x0000 to 0xffff hex. Access outputs as a 16-bit Number: R/W. Registers 10101-10164.

**32 Bits** – References monitoring of 32 digital outputs as one view/update. Value will be 0x00000000 to 0xffffffff hex. Access outputs as a 32-bit Number: R/W. Registers 10001-10032.

**Not Selected** – This is the default selection, meaning none of the above registers are selected for monitoring.

**First Output** – This is the first Digital of the above type to display for monitoring. The first available is 1, the last 999.

**Last (Opt.)** – Optional Input. This is the last Digital of the above type to display for monitoring. If this selection is set then it is assumed to be a series of incrementing Digital Outputs, beginning with the “First Output” and ending with that designated by this value. When grouped as bits greater than 1 the value will be round up to the next larger group.

### Flags

Flags are those bits, used within Quickstep programs as status bits between Quickstep tasks. They may be monitored individually or as a group.

13005	Flag Access as 32-bit Number: R/W, Alternate access to Flags 1-32.
13021-13024	Flag Group Registers
13201-13232	Flag Bit Access as 1-bit Number: R/W, 0 = off, 1 = on.

### Group

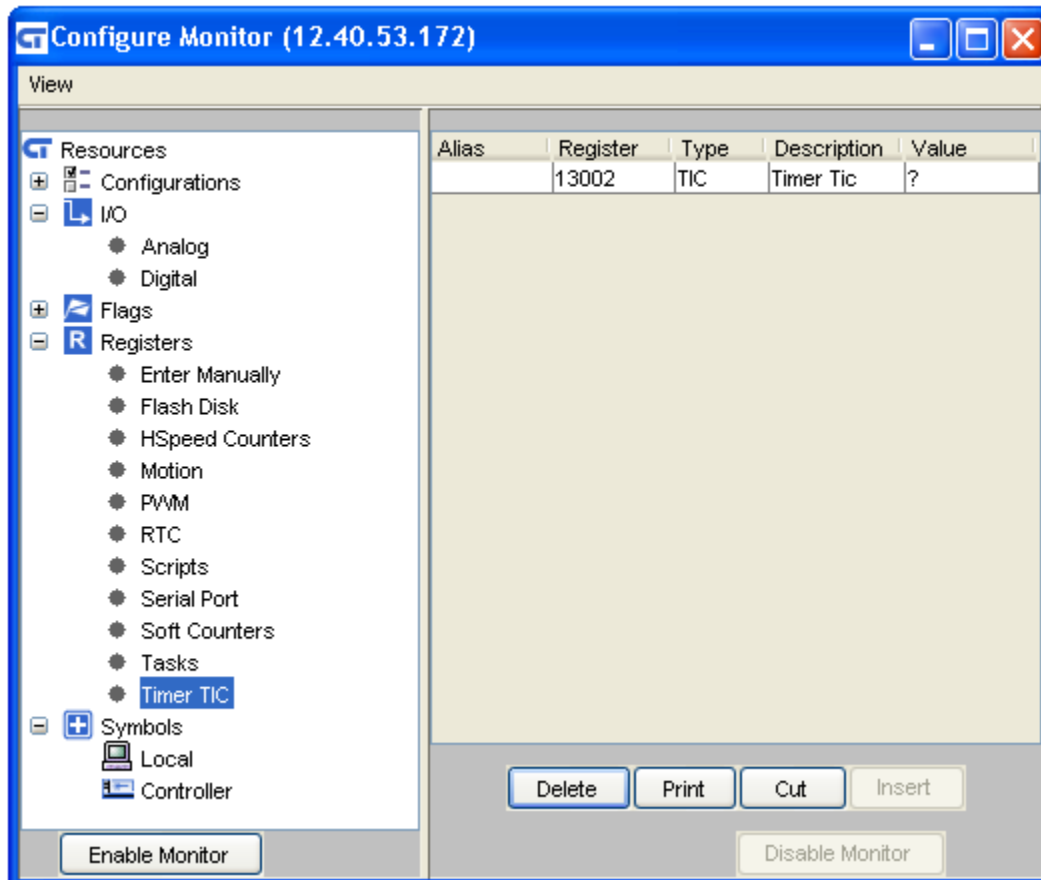


## Individual



## Registers

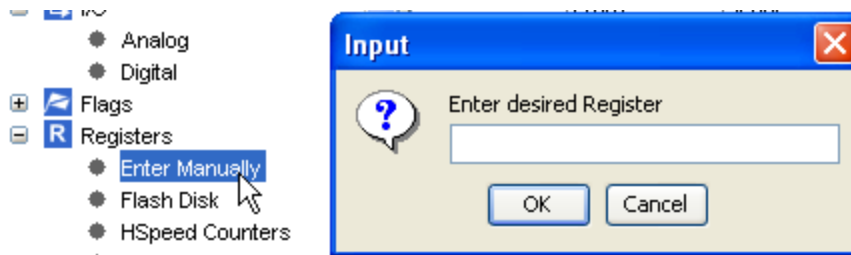
Registers are used in CTC controllers for both user assigned storage locations and for CTC dedicated functions and storage. They are defined within the "5200 Quick Reference Register Guide" (951-52006-001). This tree selection is available to make it easier to select the most commonly used registers, by their group function versus individual register number. Alternatively manual entry, by register number, is also supported. For example double-clicking the 'Timer Tic' group will add it to the Resource Viewing Window, on the right side:



The '?' mark means either the resource point is undefined or no data has yet been collected (never enabled for monitoring).

### Enter Manually

Should the desired register group not be available for automatic selection, one may be manually entered:



### Flash

This register allows the viewing of the remaining space on the root drive:

12315	Flash Disk Space Register: (R), contains the approximate free space available on the flash disk.
-------	--

### High Speed Counters

This register group consists those that pertain to the High Speed Counters:

5001	5102 Internal High Speed Counter 1 (for Dual counters, only Counter for Quadrature mode) : R/W
5002	5102 Internal High Speed Counter 2 (for Dual counters, not used in Quadrature mode) : R/W
5011	5102 Internal High Speed Counter 1 Speed : R
5100	5102 Internal High Speed Counter Mode, 0 = Quadrature, 1 = Dual Counters : R/W
5101	5102 Internal High Speed Counter Frequency Period : R/W
5102	5102 Internal High Speed Counter Frequency Value : R

### Motion

This register group consists of the main register referencing motion:

### PWM

PWM Register group references those registers which may be used to control and/or view pulse width modulation:

5901, 5905	Pulse Output Count: R/W, Number of pulses to send out of outputs 1 and 2 respectively. Storing 65535 sends pulses continuously. Any other number of pulses stored here will count down as they are output. Note: Set this register last as it will initiate the pulse upon a non-zero value.
5902, 5906	Pulse Time Output: R only, Current time output has been outputting pulses in ms. For output 1 and 2 respectively.
5903, 5907	Pulse ON Time: R/W, PWM pulse "on time" in mS. 1 mS minimum. Set this register prior to the Pulse Output Count register.
5904, 5908	Pulse Period: R/W, PWM pulse period or interval in mS. 2 ms minimum. Set this register prior to the Pulse Output Count register.

## RTC

The RTC group of registers are those that reference the Real Time Clock parameters:

13013	RTC Lock Register: R/W, 0 = locked from write operations, 1 = unlocked, OK to write
13014	Seconds: R/W
13015	Minutes: R/W
13016	Hours: R/W, 24 hour clock
13017	Day of Month: R/W
13018	Month of Year: R/W, 1-12
13019	Year: R/W, Two fields
13020	Day of Week: R/W, 1-7, where Monday = 1

## Scripts

Script Registers are those that are used to invoke scripts and display results. Both general script and email registers are available for general viewing:

12311	Script Register: (R/W), writing a numeric to this register will cause the corresponding script to be executed, for example: write a 4 and Script004.ini is executed.
12312	Script Result Register: (R), 1 = successfully executed, else error code (TBD).
12317	Email Register (R/W)
12318	Email Result Register (R)

## Serial Port

This register group references those that pertain to serial ports. Only the first 10 characters of the receive buffer are used.

12000	Select Controller Communications Port: W access, 1 = COM1, 2 = COM2, 3 - 7 = TCP raw virtual socket connections (see 22XX0 register descriptions).
12000	Message Transmission Status for Controllers: R access, 0 = not busy, 1 = busy.
12001	Transmit Message from Data Table: W only, Store row number to transmit.
12001-12009	Controller Receive Buffer Access, R only, 1 character per location. First 10 location shown for reference.

## Soft Counters

This register group references those that pertain to the soft counters:

5821-5828	Digital Input Software Counters: R/W
-----------	--------------------------------------

## Tasks

This register group references those that pertain to tasks:

13011	Task Priority: R/W, Specifies Super Task serviced on a priority basis. 1000 = Currently executing task
-------	--

13012	Current Task Number: R only
13032	Task Register Fault Status: R only
13033	Task Register Fault Step: R only
13034	Task Register Fault Task: R only
13035	Task Register Fault Data: R only
13036	Performance Adjustment: R/W
13037	Network Performance Adjustment: R/W
13101-13164	Delay Timer Registers
13038	Fault Step Register – (R/W) Step to branch to when fault occurs. Write a 0 to disable.

### Timer Tic

This is a single register, millisecond timer tic. Very useful for timing events.

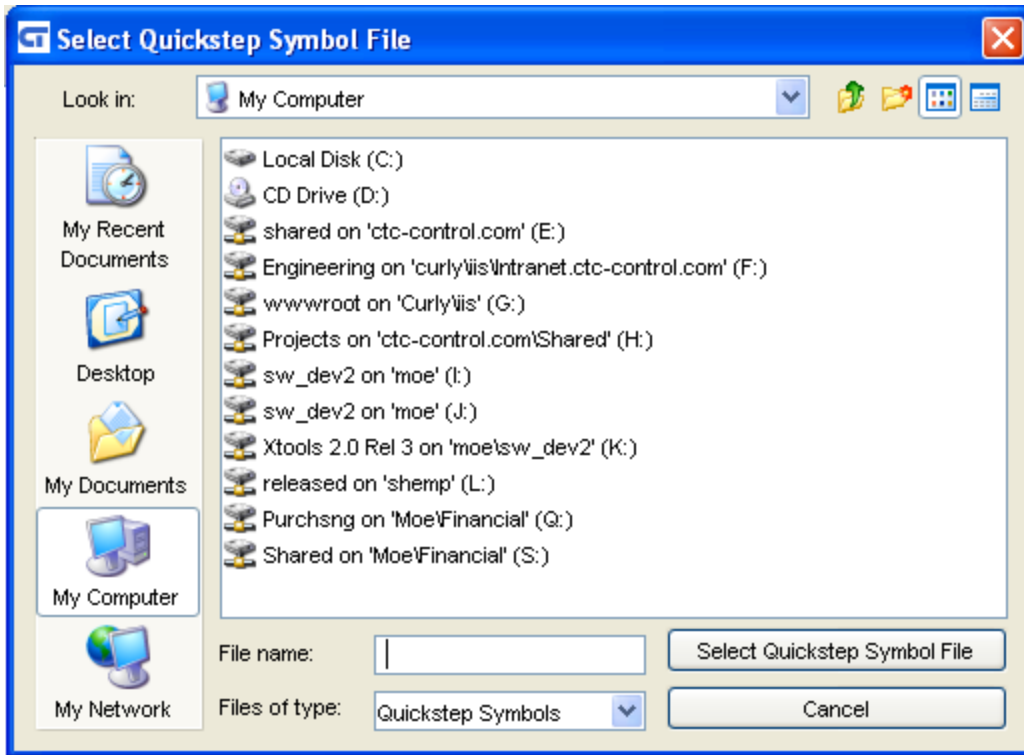
13002	Continuous millisecond counter: R/W, Increments every 1 millisecond. Range is -2,147,483,648 to +2,147,483,647.
-------	---

### Symbols

Quickstep programs generate a complete symbol table of all referenced resources. This table can be loaded into the Resource Monitor and allow for automated setup. These files have a .sym file extension. Symbol files may be stored on the controller or on your local computer.

#### *Local*

Selection of this tree node will cause a dialog box to open, requesting you to navigate to the .sym file desired:



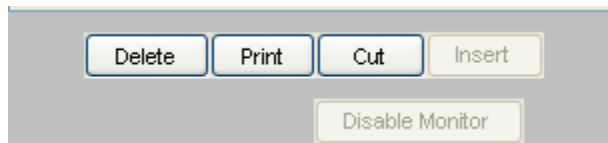
Once the file is found, and selected, its symbols will be loaded into the Resource Monitor, appended to any that already exist.

**Controller**

NOT YET IMPLEMENTED

**Resource Presentation/Editing**

The Resource Viewing Window, on the right side, contains a number of presentation/editing options to aid in register monitor preparation. The buttons at the bottom of the table are:



**Delete/Cut/Insert**

**Delete** – The ‘Delete’ button is used to remove resource entries that are not desired. Resources may be removed individually or as a group. To select more than one entry you may either drag your mouse over the rows of interest, highlighting each, or use the Control/Shift mouse options of Windows. Holding the Control key while clicking rows highlights only those rows. Holding the Shift key highlights all from the first to the current selection. Clicking the ‘Delete’ button will remove the entries.



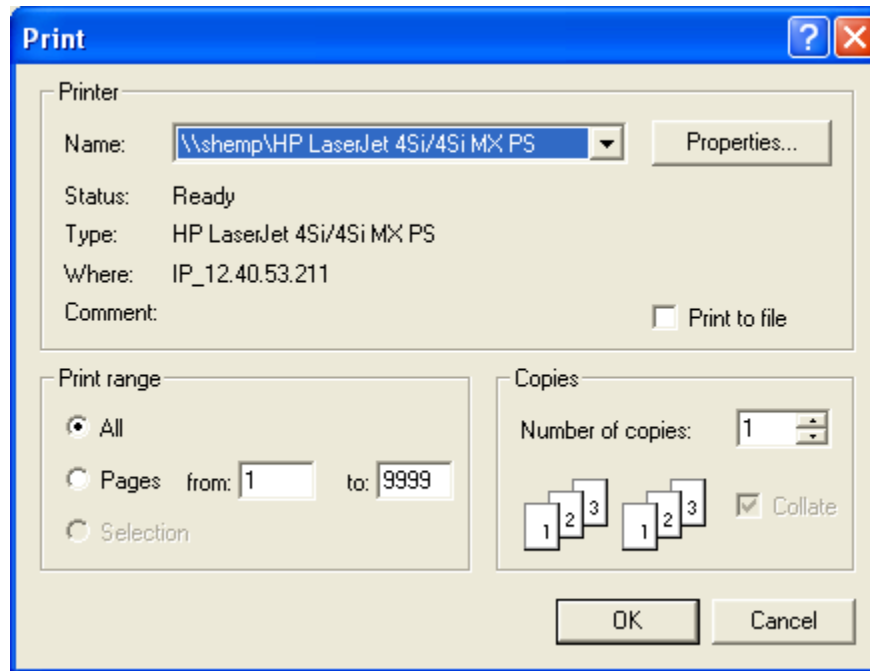
**Cut** – The ‘Cut’ button is used move a block of highlighted resources to another location in the table or to temporarily remove. There is only one cut/paste buffer.

**Insert** – The ‘Insert’ button used to insert any cut buffer at the point above the last selected row. This button will only be enabled when the cut buffer contains resources for pasting.

**Note:** When Deleting or Cutting Resource items do not sort them prior to the operation.

## Printing

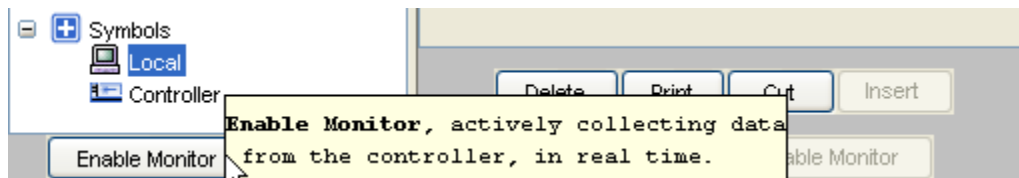
The ‘Print’ button can be used to print all the rows and columns of the Resource View Window to a local printer. A dialog box will appear requesting printer selection:



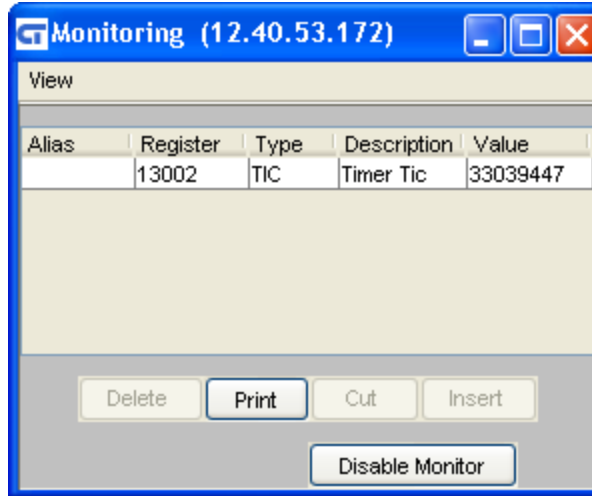
The output will be in a formatted manner, consisting of all symbolic information and any currently collected data:

## Resource Monitoring

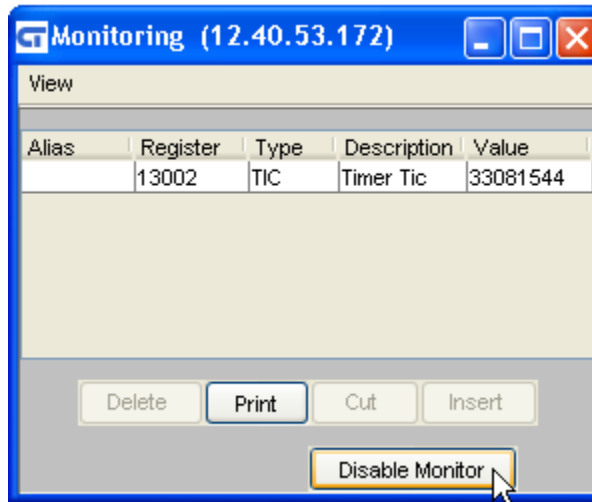
Once Resources are selected, and appear in the right hand Resource View Window, they may be monitored. Numerous monitor screens may be open at the same time, to the same or differing controllers. To enable monitoring select the “Enable Monitor” button:



Upon doing so the tree view will disappear and entries in the right hand Resource View Window will begin updating data values.



To stop monitoring and return the tree view, simply select the “Disable Monitor” button. You may enable/disable monitoring as often as you like.



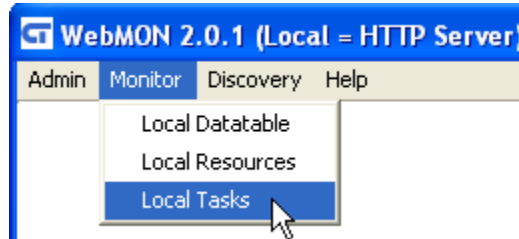
**CHAPTER**  
**7**

## Monitor - Tasks

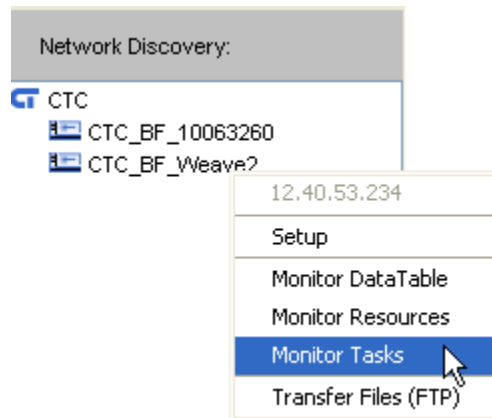


Monitoring Tasks allows the user to view the current Quickstep state of the controller and conduct diagnostic debugging where desired. You may simply view the tasks as they execute, reasons for faults, or load your actual Quickstep program for use as a source level debugger. Task execution may be controlled via breakpoints and/or single stepping, along with complete symbolic resource value viewing by highlighting the resource within its program. Resource Watch Lists are also supported to view the current contents of more than one resource at once.

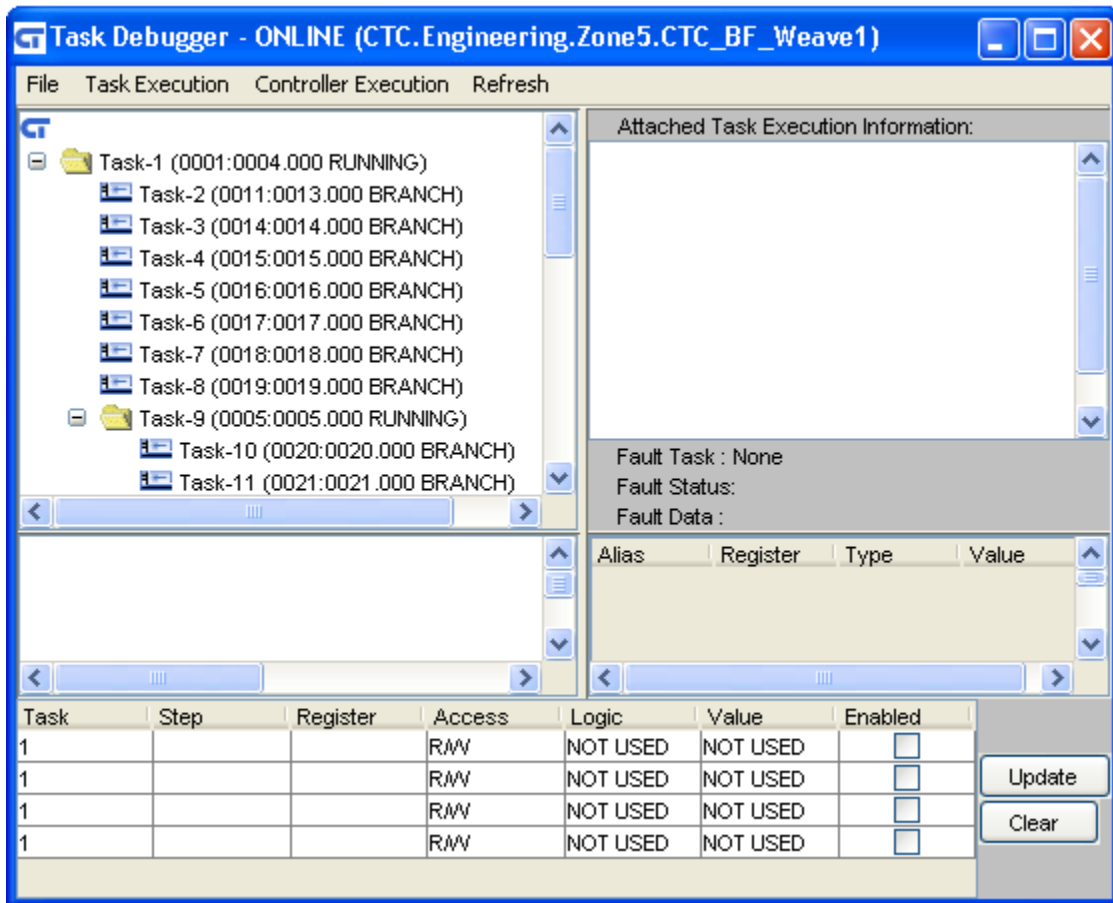
The Task Monitor may be invoked from either the main menu or a Network Discovery node menu (refer to Chapter 9). When using the main menu select “Local Tasks”:



Or if Network Discovery, Monitor Tasks:



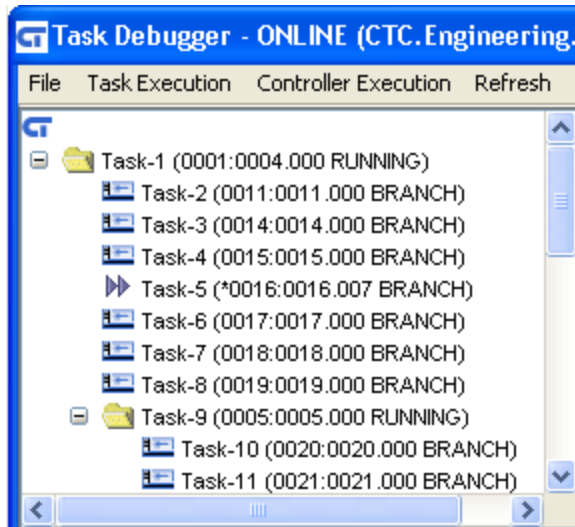
When first invoked a screen similar to the following will appear:



The top left window is known as the “Task Window”, it contains a tree structure of the currently executing tasks. The top right window is the “Command Result Window”. The window under the “Task Window” is the “Source Window”. To the right of that is the “Resource Watch Window”. Across the very bottom is the “Breakpoints Table”.

### **Task Window**

The “Task Window” displays the current state of all tasks executing when a snapshot is taken, refer to the example below:



Each task that is a parent appears as a folder, for example Task-1 and Task-9 are parents. Task-2 to Task-9 are children of Task-1 since in this example Task-1 executed a Quickstep DO statement creating the child tasks. Task-9 is a child of Task-1 but in itself is also a parent since it too executed a DO statement, causing Task-10 and Task-11 to begin. Others also are children of Task-9 but are out of view. To view those and any others simply use the scroll bars to move through the task tree.

In the view, Task-5 has a different icon than the others. When a snapshot is taken, about every 10 seconds (when not selected, detailed later). The two triangles lying on their side and pointing to the right is the icon for the currently executing step. In other words that step that had control of the controller at the moment in time the snapshot was taken.

### Task State

Each task also contains detailed information about its execution state, contained within parentheses. In the case of Task-5, the currently executing task:

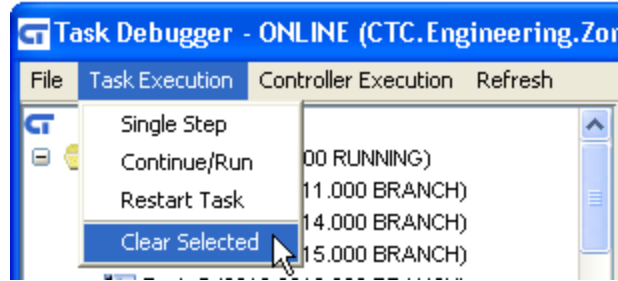
▶▶ Task-5 (\*0016:0016.007 BRANCH)

(Start Step:Current Step.Current Line State)

Therefore in the above example Task-5 began life by executing step 16. At the time of the snapshot it was executing step 16, line 7 within the step and that operation was a Branch operation. If we were to analyze the source code this task consists of a single step that loops on step 16, continuously looping.

### Task Selection

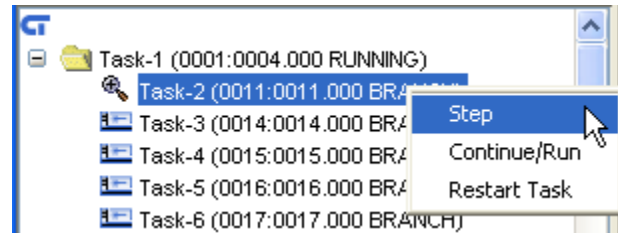
To select a task as an actively selected task, simply click it with the mouse, causing it to be highlighted and a small magnify glass to appear as its icon. Actively selected tasks are those who source code steps will be monitored in the Source Window. If none is selected the Source Window will randomly track the step that was executing when a snapshot is taken. To clear a selected task, simply select the Menu, “Task Execution->Clear Selected”.



**Command Result Window**

The “Command Result Window” echoes results directly from the controller, asynchronously. This window runs on a separate background connection to the controller and will display execution events as they occur. Typically this would be the result of single stepping, setting, triggering, and clearing breakpoints. Since it runs in the background, events can happen at anytime.

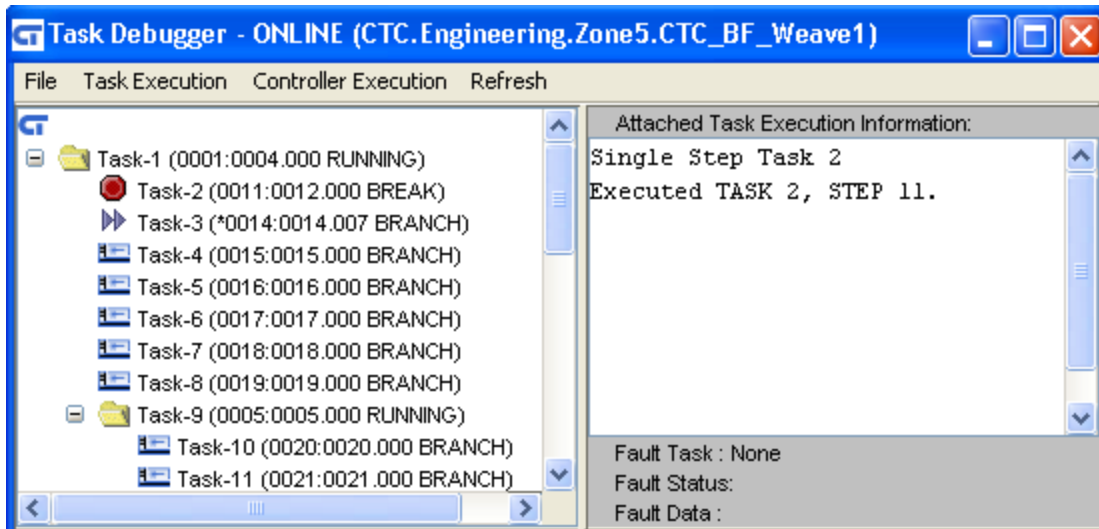
Events may be caused by commands effecting Task Execution. A popup menu will appear allowing a task’s execution to be modified by first selecting the desired task by clicking the left button of the mouse, after initial selection you may then right click the mouse anywhere within the Task Window, and the popup will appear:



Task Execution commands consist of:

**Step**

Stepping causes a task to execute one step and then stop, it will remain stopped until you continue single stepping, select Continue/Run or restart all tasks. When single stepping a selected task a red stop sign will appear as the icon, with a status of BREAK to designate the task is no longer running. Below Task-2 is currently stopped. The Command Results Window shows two lines. The first says that a Single Step Command was executed on Task 2, the second, an acknowledgement that Step 11 has been executed.



### Continue/Run

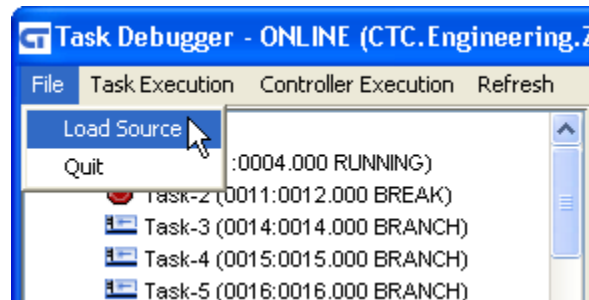
Continue/Run allows you to have a task that is either single stepping or reached a breakpoint to continue running. Execution will begin on the next step and continue executing until either single step is selected again or a breakpoint is satisfied.

### Restart Task

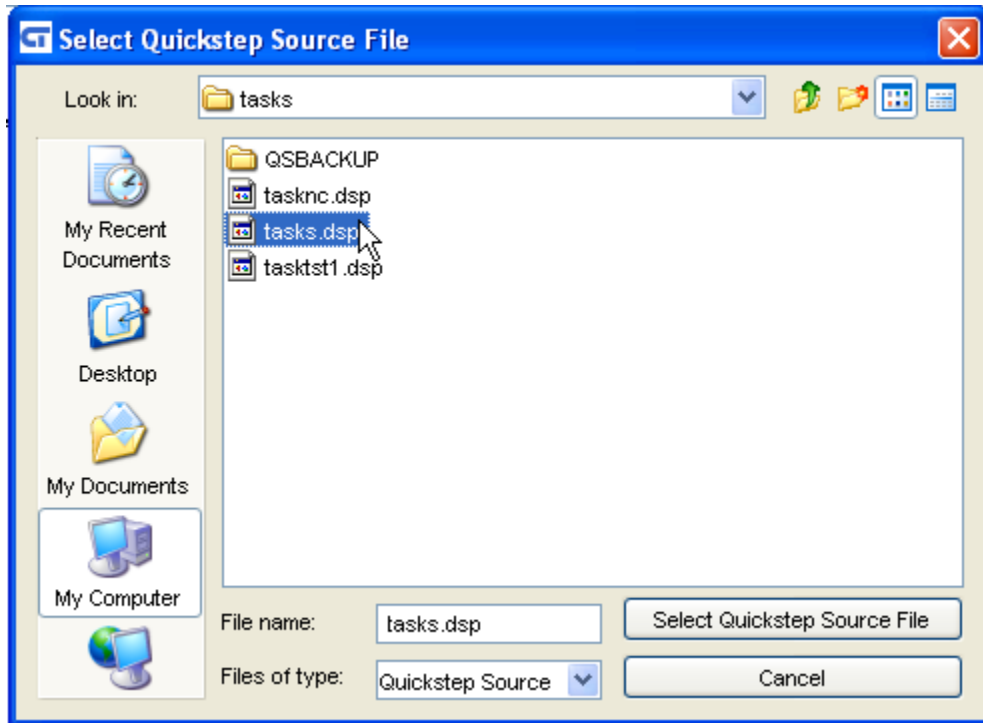
Restart Task may be used to restart a non-parent task at the step it was first created on. In the example of Task-2 this would be Step 11. Its very first execution step when referenced by the DO statement in Task-1.

## Source Window

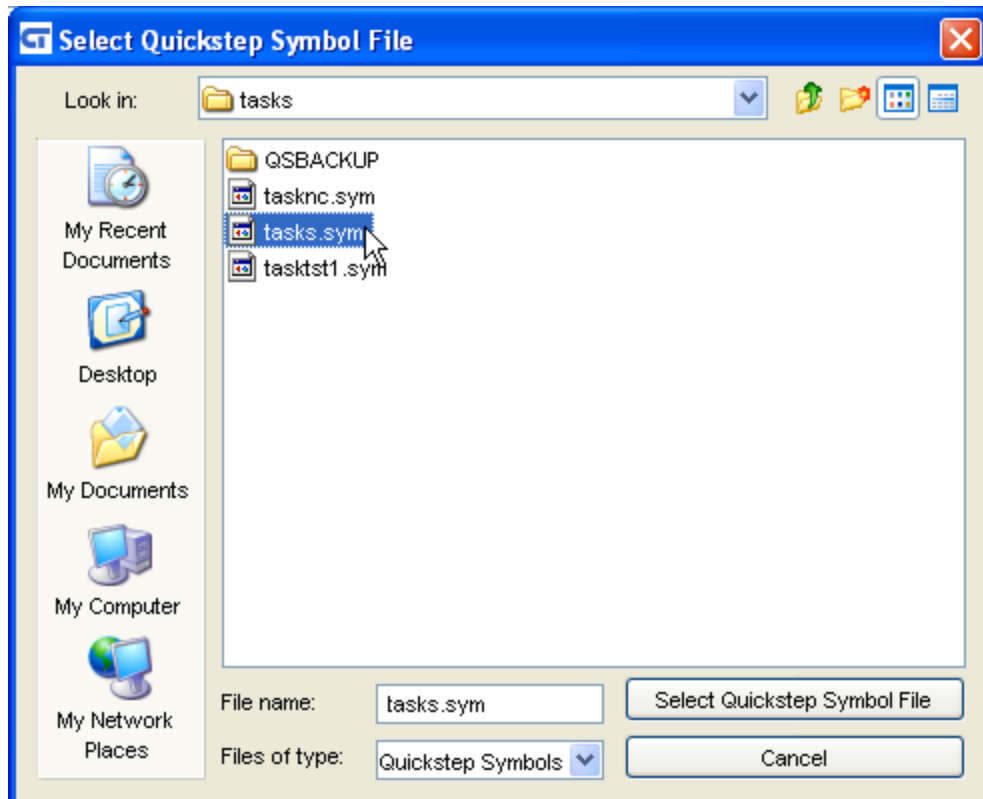
Actual Quickstep application source code may be loaded into the Source Window to view steps as they actually execute. Each executed step will be brought into view and have its step name/number highlighted in Cyan. The loaded Quickstep program should match that executing in the controller if proper matching of task execution to source code is to occur. Loading the wrong program will not cause a problem with controller execution but will show inconsistent monitoring of step execution. To load a program, select "Load Source" from the "File" menu:



A file dialog box will appear in which to navigate to the proper .dsp, Quickstep file.

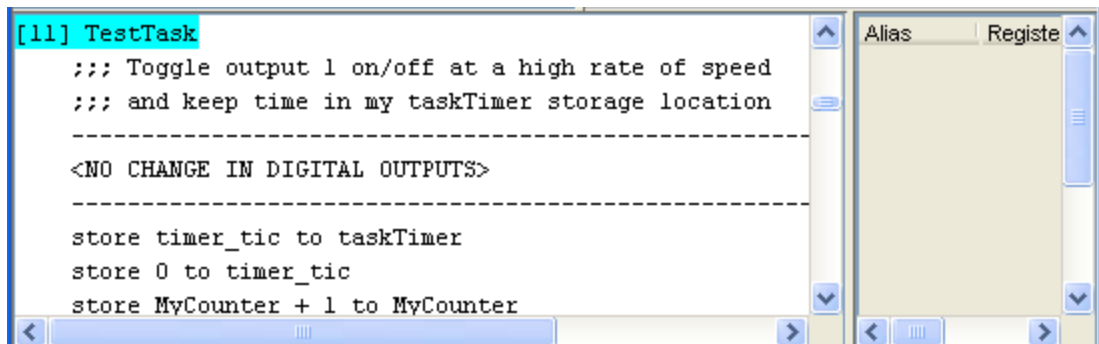


Immediately following the .dsp source file selection you will be prompted for the program symbol file as well, .sym:



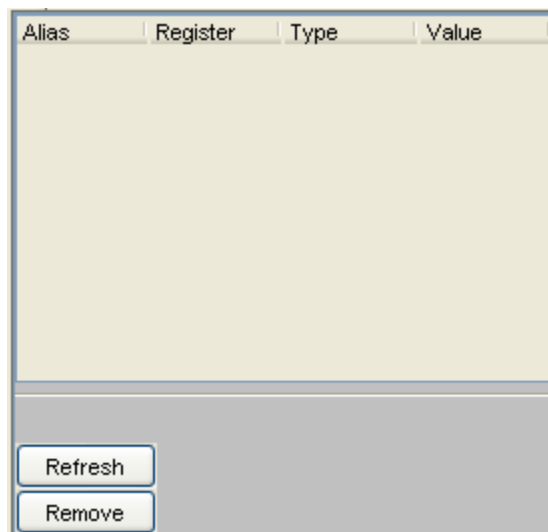


Upon selection the source will appear and as snapshots of task execution occur the executing source step will be highlighted. In the example below it is showing "TestTask" as the active step:



### Resource Watch Window

The Resource Watch Window is a window that allows you to list a number of selected variables (resources) for monitoring as you debug your program. It is located to the right of the Source Window and consists of four columns and two buttons:



Each column is identical to that found in the Resource Monitor, Alias, Register, Type, and Value. As previously defined:

**Alias** – Alias is used to display a symbol reference to a particular resource. In this case it is that symbol which was highlighted and selected for monitoring, within your Quickstep program.

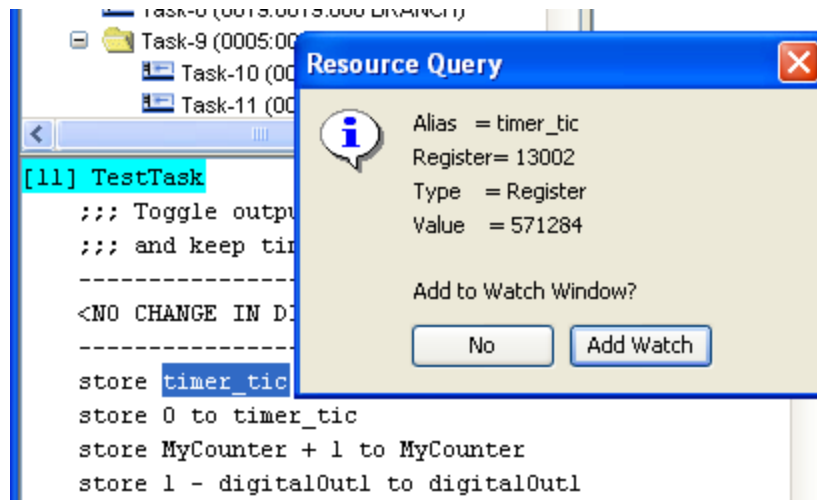
**Register** – The register number being monitored.

**Type** – Type of resource being monitor. For Example Soft Counter, digital input, etc.

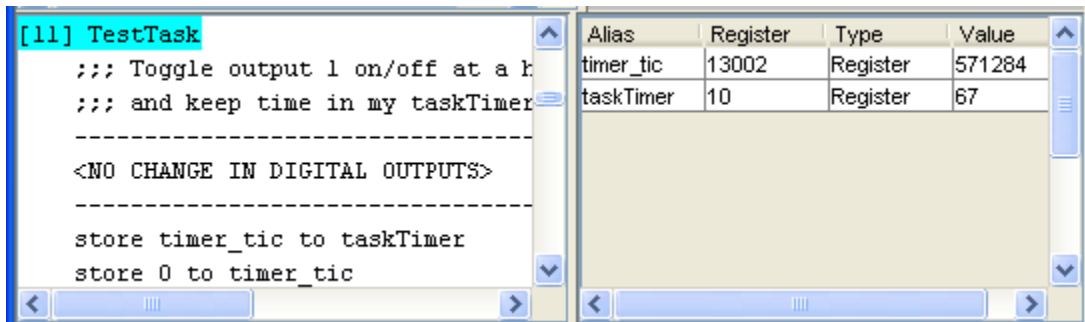
**Value** – The register value when initially added to the Watch Window or when last refreshed, via the "Refresh" button.

In order to add a variable to the watch window simply highlight that variable within the text of the loaded source code and click the right mouse. The example below is showing

the selection of the variable “timer\_tic”. Once selected you can either just view the current value, and/or select the “Add Watch” button to add it to the Resource Watch Window list.

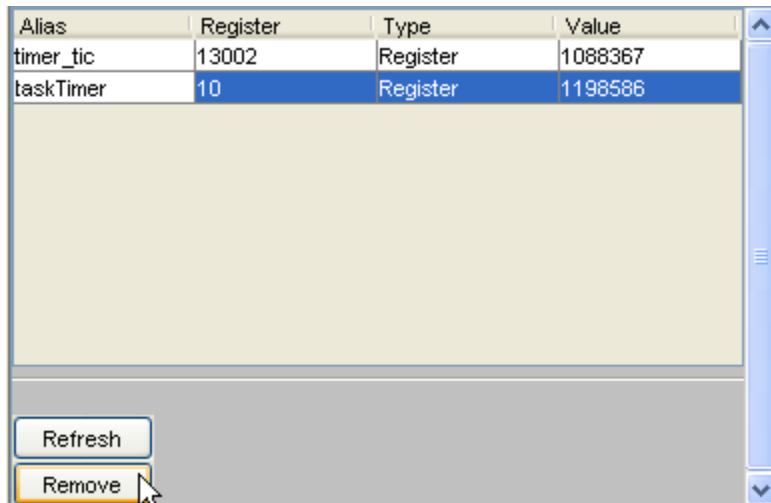


When added to the Watch Window:



Note that in the above example, both “timer\_tic” and “taskTimer” have been added from within the “TestTask” step.

To remove an item being watched, simply highlight it and select the “Remove” Button:



The “Refresh” button is use to update all of the registers listed in the Resource Watch Window. Unlike the Monitor Windows, they do not update on their own. This is useful to conserve controller resources while debugging. If you desire a constant background update you may open background Monitor Windows.

### Breakpoints Table

The Breakpoint Table allows users to set up to four breakpoints within a program. These breakpoints may be in the same or differing tasks, be based upon a particular step's execution, and/or register values. The basic layout is as follows:

Task	Step	Register	Access	Logic	Value	Enabled
2		13002	Read	>=	20	<input checked="" type="checkbox"/>
1			R/W	NOT USED	NOT USED	<input type="checkbox"/>
1			R/W	NOT USED	NOT USED	<input type="checkbox"/>
1			R/W	NOT USED	NOT USED	<input type="checkbox"/>

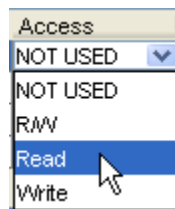
The table consists of seven columns, each functional as follows:

**Task** - This is the Task that the breakpoint is being set for. All breakpoints are task dependent and will only occur when the events desired actually occur for that task.

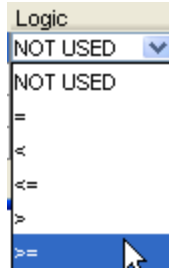
**Step** – This field is blank if breakpointing on a particular step is disabled, or set to the particular step number or symbol desired. For example in the previous section step #11 has a name of “TestTask”. You may either enter 11 or TestTask to reference this step. If entered then any time the listed Task executes this step a breakpoint will occur and the task will stop execution, causing the red stop sign icon to appear and a status of BREAK to be shown. Note that you may highlight the step name in the Source Window and use CNTL C to copy from the window and CNTL V to paste the symbol to the Breakpoints Table.

**Register** – This field is blank if no register breakpoint qualifiers are enabled otherwise set to the desired register whose contents you want to monitor during all modifications. You may enter the symbol name or the register number. Note that the “Access”, “Logic”, and “Value” column may be used to refine the rules of access, causing a breakpoint. By default any read/write of the defined register will cause a break to occur. You may refine the parameters as detailed within those sections.

**Access** – “Access” defines the type of access that will cause the register break. It may be the default Read/Write, Read, or Write only. A pull down box is provided:



**Logic** – “Logic” defines that when a particular register access occurs how to handle the register value at that moment. You may further refine the register breakpoint by setting a range of the value, which will cause the break. A pull down menu is provided:



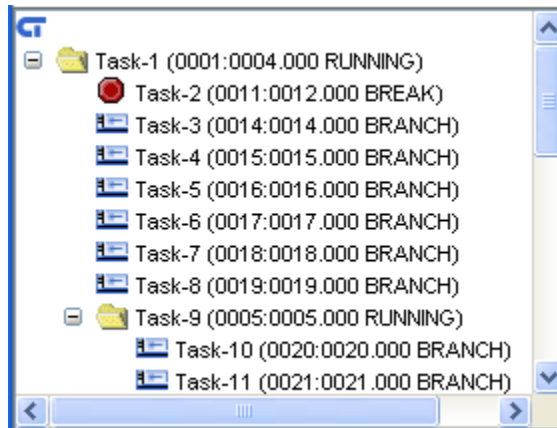
**Value** – “Value” defines a range, if desired, with which a register breakpoint should occur. It is used in conjunction with “Access” and “Logic”.

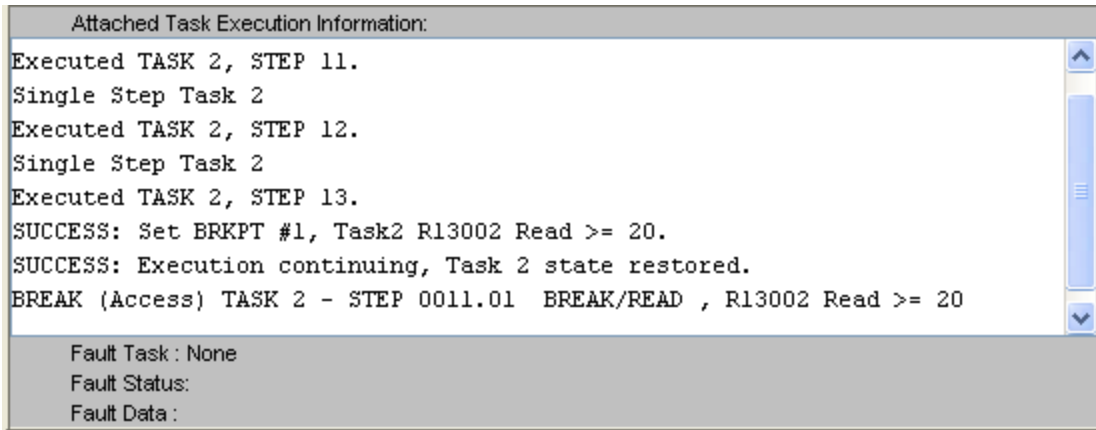
**Breakpoint Example:**

Task	Step	Register	Access	Logic	Value	Enabled
2		13002	Read	>=	20	<input checked="" type="checkbox"/>
1			R/W	NOT USED	NOT USED	<input type="checkbox"/>
1			R/W	NOT USED	NOT USED	<input type="checkbox"/>
1			R/W	NOT USED	NOT USED	<input type="checkbox"/>

The above example will cause a register break only. The Step portion of this breakpoint is disabled since it is blank. If Task 2 reads register 13002 and the “Value” at that time is greater than or equal to (>=) 20, a break will occur and the task will stop executing.

Once the breakpoint is set in the table you must enable it by selecting the checkbox and then pressing the “Update” button. It is then written to the controller and an acknowledgement will be sent by the controller, appearing in the Command Result Window. Once a breakpoint occurs the red icon will appear and a notification of the break condition is shown in the Command Result Window:





The status line of “SUCCESS: Set BRKPT #1, Task2...” shows the breakpoint being set and accepted by the controller. Once the breakpoint status is satisfied the red task icon appears on Task-2 and the “BREAK (Access) Task 2 – STEP 0011.01...” status is shown in the Command Result Window. The 0011.01 means that the break occurred on step 11, instruction line 1 of the source code.

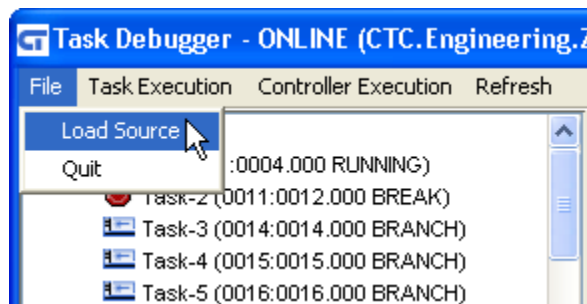
## Menu

A Menu is available at the top of the Task Monitor. It provides a number of functions, some of which have already been discussed and may be redundant. The main menu headers are:



**File** – This menu selection allows the loading of source code and exiting the task monitor.

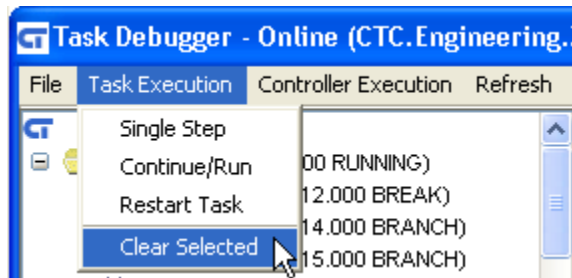
- Load Source – Load the desired Quickstep .dsp file into the Source Window.
- Quit – Exit and close the Task Monitor.



**Task Execution** – This menu provides selections that are task specific. Most of the selections are also available by first selecting a Task, left clicking the mouse, then right clicking for a popup menu. All of the selections have been previously discussed in “Command Results Window” section.

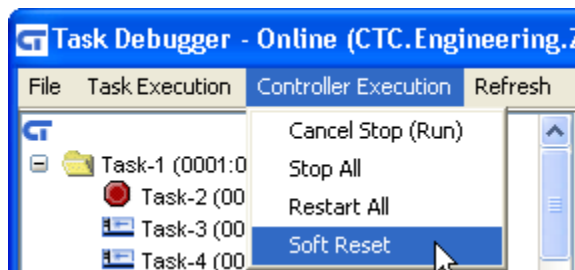
- Single Step – Single step a task, one step at a time.

- Continue/Run – Let a single stepped task, or one that has stopped due to a breakpoint, continue execution at full speed.
- Restart Task – Restart a task at the very first step that it executed when it came into being. For Task #1 this would be the very first step. For others it would be based on the parent DO statement.
- Clear Selected – When the mouse is clicked on a running task it becomes the active task, causing the source code window to track that task's execution. Selecting this menu item makes it as though no task is active and the source code window will track that of the snapshot, randomly moving to that active at that moment.

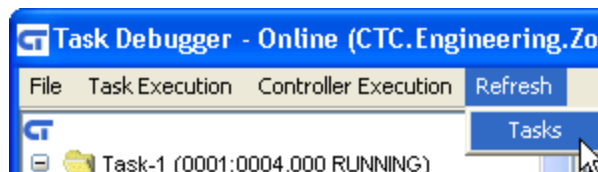


**Controller Execution** – This menu provides selections that control all tasks.

- Cancel Stop (Run) – Cancel a “Stop All” and begin running all stopped tasks. Also required after a “Restart All” to run all tasks.
- Stop All – Stop all tasks. They may be continued using the “Cancel Stop” selection.
- Restart All – Restart all tasks from the beginning, do not reset the controller.
- Soft Reset – Reset the controller by doing a soft reset of all hardware and restarting all tasks from the beginning, including invoking any \_startup.ini file.



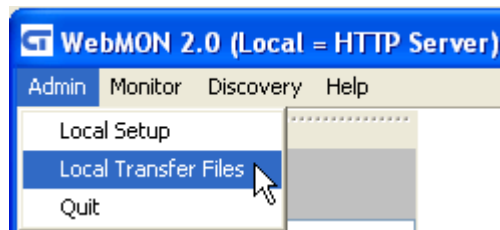
**Refresh** – This menu option reveals a single selection, Tasks. Selection of this menu option will cause the Task Window to immediately update the current task state.



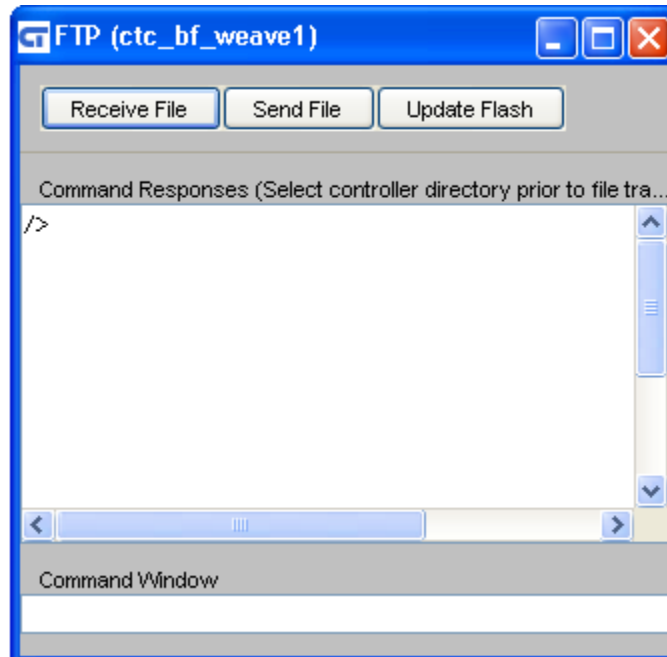
## File Transfer (FTP)



FTP, File Transfer Protocol, is the main file transfer protocol supported by the controller. To ease access a simple command line screen is available from both the main menu's and Discovery popup:



Once invoked the following screen will appear:



The bottom most window consists of a single command line. The large window is the result window and displays all information returned by the controller. Supported commands are as follows:

**cd-** Change to a different directory.

**delete**- Delete a file.

**dir** - Displays a directory consisting of the filename, date of creation and size.

**get** - Copies a file from the controller to a local directory on your computer.  
Example: get filename.ext

**mkdir** - Create a new directory inside the current directory.

**send** - Copies a file from a local directory on your computer to the controller.  
Example: “send filename.ext” or “send filename.ext newname.ext”

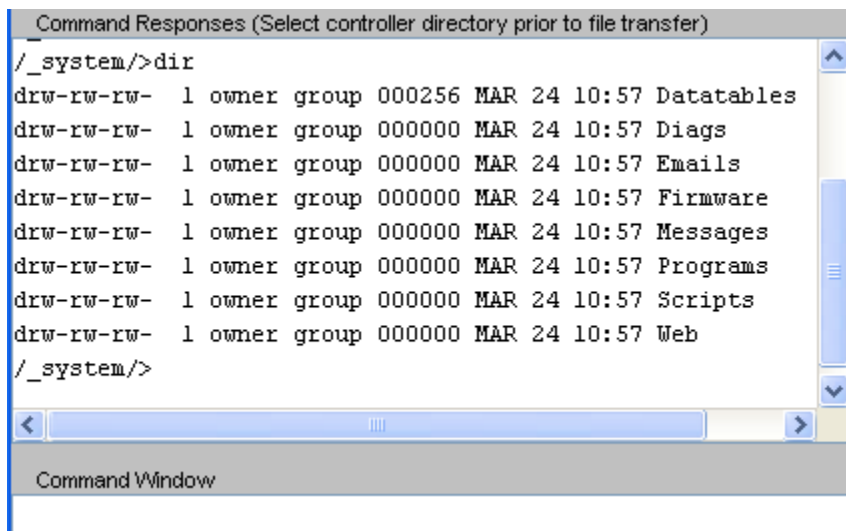
**pwd** - Displays the current directory path.

**quit** - Quit the FTP session and close the window.

**rmdir** - Remove directory.

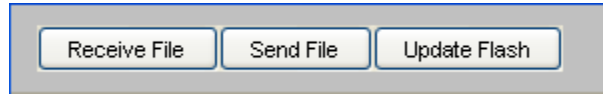
**Root Drive System Sub-Directories:**

- *Datatables* – All Quickstep loadable data table images are required to be here.
- *Diags* – All Resource Monitor configuration files reside here (.cfg).
- *Emails* – All Email SMTP files would reside here (.email).
- *Firmware* – All modules, monitor, and communications firmware that is to be loadable using the “update” command must be located.
- *Messages* – The “message.ini” file, for formatted output messages must be stored here (refer to *5200 Script Language Guide* for details of its use).
- *Programs* – All Quickstep programs (.dso) should be placed here to be run from scripts or telnet commands using the “run program qsprog.dso” command.
- *Scripts* – All system scripts, .ini files, except messages, should be placed in this sub-directory.
- *Web* – Root directory used by the resident web server. Index.html resides here.



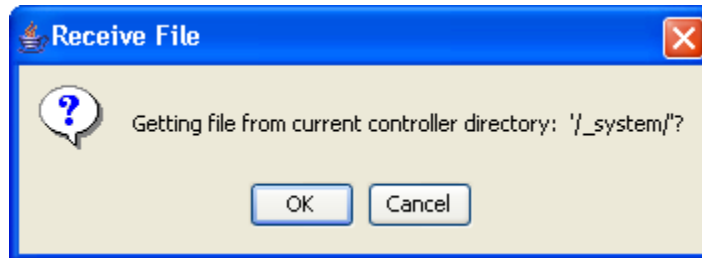


Three buttons are also available across the top of the screen, 'Receive File', 'Send File' and 'Update Flash'.

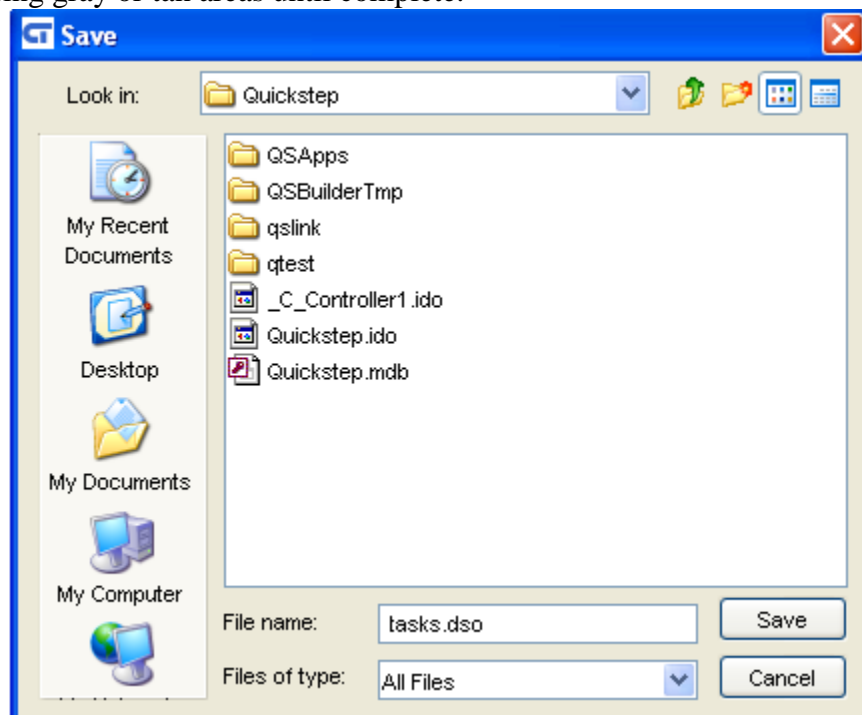


## Receive File

Files may be automatically loaded from the controller to the local computer by simply opening a file dialog box. Prior to receiving the file you must first set the current directory where the file resides, on the controller, using the command line window. Next click the "Receive File" button and the following dialog will appear verifying that the proper directory is set:

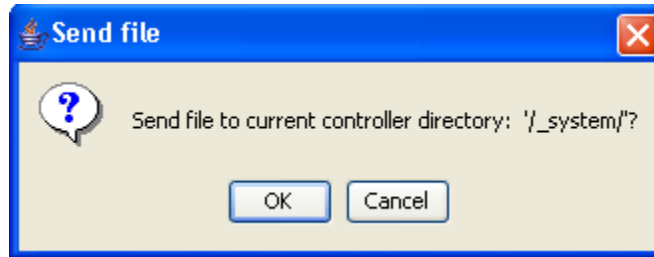


With the proper directory to retrieve the file from verified, select 'OK' and a dialog box of appears allowing you to set where on your computer to place the file being loaded from the controller. Note that the filename you enter must match that on the controller. Press, Save to begin the transfer. Refresh of your screen will stop on large files; this is normal, causing gray or tan areas until complete.

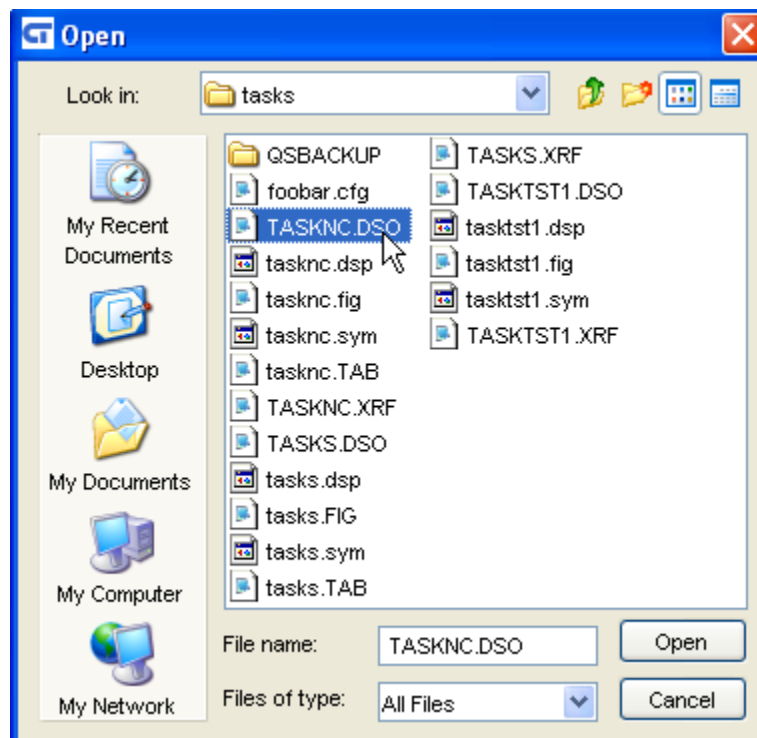


## Send File

Files may be automatically transferred from the local computer to the controller using the “Send File” button. Prior to sending the file you must first set the desired directory, on the controller, where you wish the file to be transferred to. Upon selecting ‘Send’ you will be asked to verify the location, on the controller, where the file is to be stored.



Next click the “OK” button and the following dialog will appear:



Double-click the desired file (or select and click the Open button), once navigated to, and the file will be immediately transferred.

## Update Flash

The “Update Flash” button can be used to expose a dialog box from which you can select a firmware update to be sent to the controller. Any file selected will be transferred to the

root directory of the controller, causing a firmware update. Typically this will cause communications loss with the controller if it is the main controller firmware.

All 5200 Firmware is field upgradeable via an Ethernet FTP connection. This typically consists of transferring either a binary (.bin) or S Record (.sr and .sr1) file to the controller. There are numerous different files available, each to update differing functionality and modules. Filenames used are created as follows and are case sensitive:

*[base name] V [revision, #####].[file type extension]*

Where ##### is the revision level of the firmware.

**BF5200V#####.sr1** – Main Quickstep operating environment firmware, typically 1.9 meg S record file which can not be stored directly to the flash disk due to its large size. Place only in flash disk root directory for immediate update.

**5130V#####.bin** – Analog Input Module firmware, about 3K binary file.

**5131V#####.bin** – Analog Input Module firmware, about 3K binary file.

**5132V#####.bin** – Analog Output Module firmware, about 3K binary file.

**5140V#####.sr** – Servo Module firmware, about 350K S Record file.

**5150V#####.sr** – Stepper Module firmware, about 350K S Record file.

**mon5200V#####.sr** – Main board CPU boot monitor file, rarely updated, about 100K S Record file.

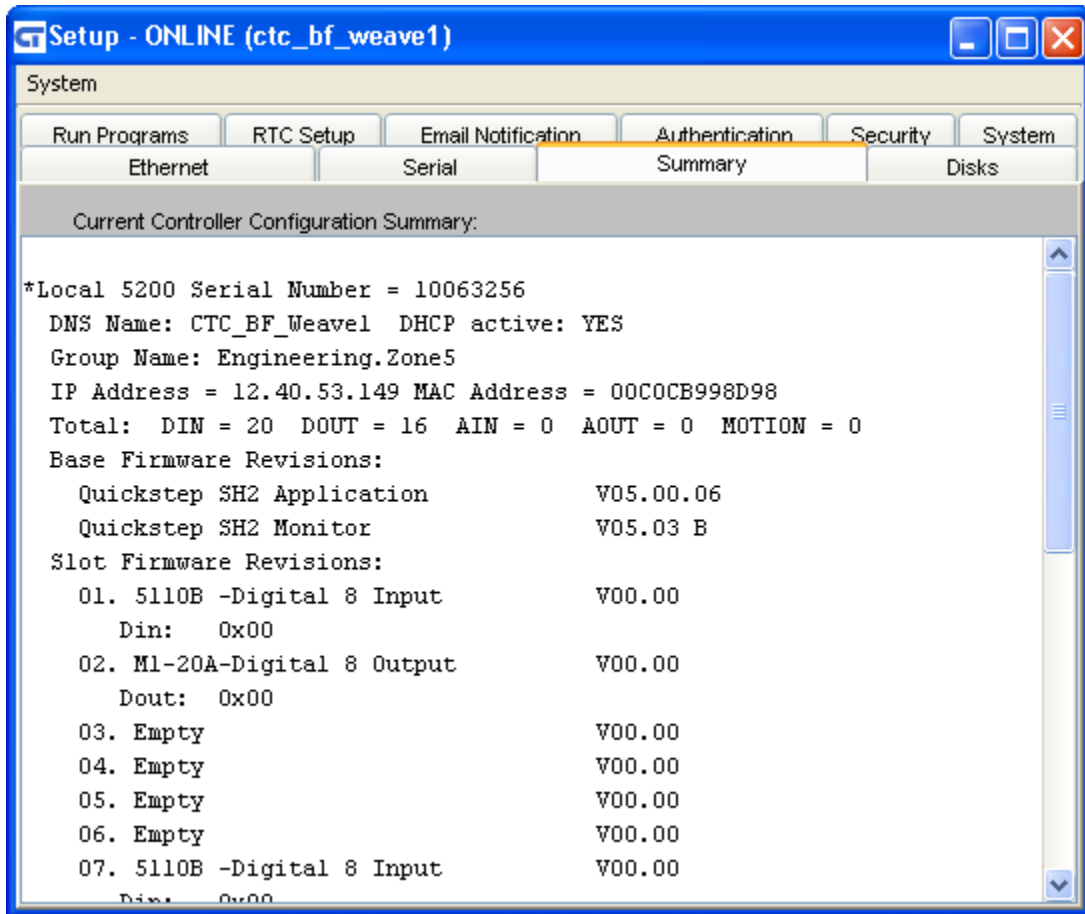
File transfers to the root directory of the 5200 file system are temporarily stored in dynamic memory and then immediately begin re-flashing the appropriate module. Small firmware files can be stored on the flash disk in the /\_system/Firmware sub-directory to allow updating of modules via the telnet “update” script command, at any time. Quickstep programs are automatically stopped during re-flash operations and once all firmware is updated a “reset” command must be initiated to re-activate VBIAS and restart the application program.

Note: Upon initiating a firmware update the 5200 FAULT LED will provide the following visual feedback:

1. Flash once then turn off during the erase cycle.
2. Rapidly flash at different rates as blocks are programmed.
3. Flash 3 times at completion to signify the flash has been programmed correctly.

Once firmware has been updated and the unit reset, it is best to observe the current revision levels of firmware to confirm that the modules updated are operating properly.

This can be done by viewing the Summary tab within the Administrative Setup screen (Admin->Setup).



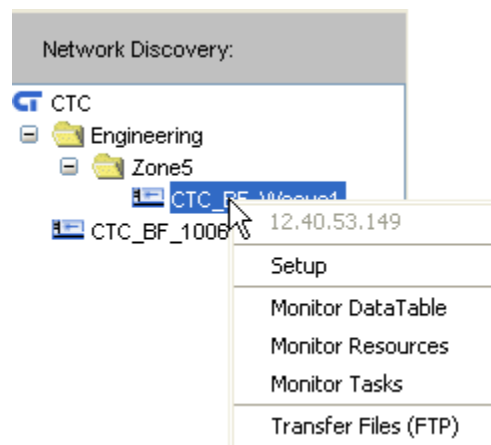
**CHAPTER**  
**9**

## Discovery (Network)



When Discovery->Network is selected from the main menu it will cause a tree view, to appear, consisting of any controllers residing on the network (5200 series). WebMON then begins broadcasting (255.255.255.255) queries to all controllers. Those responding will be listed in a tree structure and may be viewed based upon their Group Name hierarchy. The example to the left shows two controllers, one in a group and the other, part of the root directory. The controller in the group, contains folders, would be read Engineering.zone5.CTC\_BF\_Weave1, where the group name is Engineering.zone5 and the domain (DNS) name of the controller is CTC\_BF\_Weave1. The standalone controller is CTC\_BF\_10063260.

Clicking the mouse on any controller node will cause a popup menu to appear. This menu lists all of the options, which are on the main menu Admin/Monitor main menus.



The difference is the main menu is for “Local” access to the controller from which the WebMON applet was served while the tree view selection is for both the local web server controller and remote controllers found via broadcasts. Note the controller’s IP address is grayed out on the top of the menu.

To temporarily remove the Discovery window from view simply click the “Hide Discovery” button. Discovery requests are sent to controllers, about every 10-15 seconds, refreshing the tree accordingly. It may be re-selected for viewing at any time from the main menu.

### ***Tunneling***

Discovery only works when the controllers and WebMON are on the same local network since broadcasts are not routable across differing networks. The “Enable Tunneling” button allows WebMON to contact the controller, from which it was served, and request it to do the broadcasts for it. This allows WebMON to reside anywhere on the Internet and still be able to fully access any controller. Note that if any firewall exists between the controller and WebMON, ports 80 and 23 TCP connections, inbound to the controller, must be open. The controller being used for Tunneling, will relay broadcast online information, not direct resource-monitoring information. The Tunnel is used to retrieve the controller’s IP address information. If that IP address is not valid (10 or 192, local only.) then the controller cannot be contacted from the external network. The IP address must be routable.