ViewMarq™
LED MESSAGE DISPLAY

**Hardware / Software
User Manual**

MD-USER-M

⚡ ADVERTENCIA ⚡

Gracias por comprar equipo de automatización de **Automationdirect.com**®. Deseamos que su nuevo equipo de automatización opere de manera segura. Cualquier persona que instale o use este equipo debe leer esta publicación (y cualquier otra publicación pertinente) antes de instalar u operar el equipo.

Para reducir al mínimo el riesgo debido a problemas de seguridad, debe seguir todos los códigos de seguridad locales o nacionales aplicables que regulan la instalación y operación de su equipo. Estos códigos varían de área en área y usualmente cambian con el tiempo. Es su responsabilidad determinar cuales códigos deben ser seguidos y verificar que el equipo, instalación y operación estén en cumplimiento con la revisión mas reciente de estos códigos.

Como mínimo, debe seguir las secciones aplicables del Código Nacional de Incendio, Código Nacional Eléctrico, y los códigos de (NEMA) la Asociación Nacional de Fabricantes Eléctricos de USA. Puede haber oficinas de normas locales o del gobierno que pueden ayudar a determinar cuales códigos y normas son necesarios para una instalación y operación segura.

Si no se siguen todos los códigos y normas aplicables, puede resultar en daños al equipo o lesiones serias a personas. No garantizamos los productos descritos en esta publicación para ser adecuados para su aplicación en particular, ni asumimos ninguna responsabilidad por el diseño de su producto, la instalación u operación.

Nuestros productos no son tolerantes a fallas y no han sido diseñados, fabricados o intencionados para uso o reventa como equipo de control en línea en ambientes peligrosos que requieren una ejecución sin fallas, tales como operación en instalaciones nucleares, sistemas de navegación aérea, o de comunicación, control de tráfico aéreo, máquinas de soporte de vida o sistemas de armamentos en las cuales la falla del producto puede resultar directamente en muerte, heridas personales, o daños físicos o ambientales severos ("Actividades de Alto Riesgo"). **Automationdirect.com** específicamente rechaza cualquier garantía ya sea expresada o implicada para actividades de alto riesgo.

Para información adicional acerca de garantía e información de seguridad, vea la sección de Términos y Condiciones de nuestro catálogo. Si tiene alguna pregunta sobre instalación u operación de este equipo, o si necesita información adicional, por favor llámenos al número 770-844-4200 en Estados Unidos.

Esta publicación está basada en la información disponible al momento de impresión. En **Automationdirect.com** nos esforzamos constantemente para mejorar nuestros productos y servicios, así que nos reservamos el derecho de hacer cambios al producto y/o a las publicaciones en cualquier momento sin notificación y sin ninguna obligación. Esta publicación también puede discutir características que no estén disponibles en ciertas revisiones del producto.

Marcas Registradas

⚡ AVERTISSEMENT ⚡

Nous vous remercions d'avoir acheté l'équipement d'automatisation de **Automationdirect.com®**, en faisant des affaires comme, **AutomationDirect**. Nous tenons à ce que votre nouvel équipement d'automatisation fonctionne en toute sécurité. Toute personne qui installe ou utilise cet équipement doit lire la présente publication (et toutes les autres publications pertinentes) avant de l'installer ou de l'utiliser.

Afin de réduire au minimum le risque d'éventuels problèmes de sécurité, vous devez respecter tous les codes locaux et nationaux applicables régissant l'installation et le fonctionnement de votre équipement. Ces codes diffèrent d'une région à l'autre et, habituellement, évoluent au fil du temps. Il vous incombe de déterminer les codes à respecter et de vous assurer que l'équipement, l'installation et le fonctionnement sont conformes aux exigences de la version la plus récente de ces codes.

Vous devez, à tout le moins, respecter toutes les sections applicables du Code national de prévention des incendies, du Code national de l'électricité et des codes de la National Electrical Manufacturer's Association (NEMA). Des organismes de réglementation ou des services gouvernementaux locaux peuvent également vous aider à déterminer les codes ainsi que les normes à respecter pour assurer une installation et un fonctionnement sûrs.

L'omission de respecter la totalité des codes et des normes applicables peut entraîner des dommages à l'équipement ou causer de graves blessures au personnel. Nous ne garantissons pas que les produits décrits dans cette publication conviennent à votre application particulière et nous n'assumons aucune responsabilité à l'égard de la conception, de l'installation ou du fonctionnement de votre produit.

Nos produits ne sont pas insensibles aux défaillances et ne sont ni conçus ni fabriqués pour l'utilisation ou la revente en tant qu'équipement de commande en ligne dans des environnements dangereux nécessitant une sécurité absolue, par exemple, l'exploitation d'installations nucléaires, les systèmes de navigation aérienne ou de communication, le contrôle de la circulation aérienne, les équipements de survie ou les systèmes d'armes, pour lesquels la défaillance du produit peut provoquer la mort, des blessures corporelles ou de graves dommages matériels ou environnementaux («activités à risque élevé»). La société **AutomationDirect** nie toute garantie expresse ou implicite d'aptitude à l'emploi en ce qui a trait aux activités à risque élevé.

Pour des renseignements additionnels touchant la garantie et la sécurité, veuillez consulter la section Modalités et conditions de notre documentation. Si vous avez des questions au sujet de l'installation ou du fonctionnement de cet équipement, ou encore si vous avez besoin de renseignements supplémentaires, n'hésitez pas à nous téléphoner au 770-844-4200.

Cette publication s'appuie sur l'information qui était disponible au moment de l'impression. À la société **AutomationDirect**, nous nous efforçons constamment d'améliorer nos produits et services. C'est pourquoi nous nous réservons le droit d'apporter des modifications aux produits ou aux publications en tout temps, sans préavis ni quelque obligation que ce soit. La présente publication peut aussi porter sur des caractéristiques susceptibles de ne pas être offertes dans certaines versions révisées du produit.

Marques de commerce

ViewMarq™
LED MESSAGE DISPLAY

HARDWARE / SOFTWARE
USER MANUAL

▼AUTOMATIONDIRECT™.com

Please include the Manual Number and the Manual Issue, both shown below, when communicating with Technical Support regarding this publication.

TABLE OF CONTENTS

GETTING STARTED

CHAPTER

1

In This Chapter...

1

Introduction

ViewMarq is an LED message display utilizing the latest LED and communications technologies for industry. It utilizes both serial and Ethernet communications. The display can be connected to a PLC or any device capable of sending serial ASCII, Modbus RTU, or Modbus TCP communications. It can accept commands over Modbus and Modbus TCP, or serial ASCII.

Some of the features of the ViewMarq LED message displays are:

- Powered by 100 to 240 VAC or 24 VDC
- Low power consumption, high-efficiency switch mode supply
- CE, UL508, NEMA12 indoor, NEMA 4 indoor
- Multiple mounting options
- Free, easy to use configuration software
- No configuration DIP switches. Everything is set through software for ease of use.
- RS232/485 Serial ASCII and Modbus RTU and Ethernet Modbus TCP communications can be used at the same time.

Purpose of the Software

The ViewMarq Configuration Software serves four purposes:

1) Create and Preview Messages
2) Display messages on the ViewMarq
3) Configure the ViewMarq LED display
4) Create ASCII String Commands that may be used in a PLC to control the LED message display
5) Create and load a Message list to the ViewMarq LEDdisplay.

Purpose of this Manual

This user manual provides information that will help you mount, wire, configure, use, troubleshoot, and maintain your ViewMarq message display. The manual includes information that is critical to the safety of the personnel who will install and use the display, and to the machinery, processes, and equipment that interact with the display.

1

Conventions Used

When you see the exclamation point icon in the left-hand margin, the paragraph to its immediate right will be a warning. This information could prevent injury, loss of property, or even death in extreme cases. Any warning in this manual should be regarded as critical information that should be read in its entirety. The word Warning in boldface will mark the beginning of the text.

When you see the attention icon in the left-hand margin, the paragraph to its immediate right is in-tended to draw attention to important information that will affect the functionality of the device. The word Attention! in boldface will mark the beginning of the text.

*When you see the notepad icon in the left-hand margin, the paragraph to its immediate right will be a spe-cial note. Notes represent information that may make your work quicker or more efficient. The word **NOTE:** in boldface will mark the beginning of the text.*

Key Topics for Each Chapter

The beginning of each chapter will list the key topics that can be found in that chapter.

Getting Started! CHAPTER **1**

In This Chapter...

What's in the Box?

Included in the shipping carton are the following:

1) The ViewMarq LED message display unit

2) Mounting Brackets
   - Two Wall Mount Bracket assemblies with hardware (disassembled)
   - Two Chain Mount Brackets with hardware (disassembled)

3) Back Covers
   - Power Back Cover
     · 1/2" cable gland for power cable (with 10mm grommet installed)

   - Communication Back Cover
     · Two 1/2" cable gland (6mm and 10 mm grommets installed)
     · 3/4" cable gland (with 6mm split grommet installed)
     · Two cable gland plugs (1 - 6mm, 1 - 10 mm)

4) Accessories
   · 7 ft. Ethernet Patch Cable
   · Installation Guide

Getting Started

About Getting Started

If you are familiar with LED message displays and PLC's in general, then following the simple steps in this chapter may be all you require to start being productive using a ViewMarq LED display. After you have completed the steps, your ViewMarq LED display will be displaying messages you create on your PC. If you are new to the world of LED displays and PLCs, be sure to familiarize yourself with all of the chapters in this user manual to help you get started.

Supplemental Manuals and Other Help.

The ViewMarq Configuration Software, (MD-PGMSW), can be downloaded free from the AutomationDirect web site (**http://www.automationdirect.com**). This User Manual (MD-USER-M) is free as a download. The ViewMarq Configuration Software also includes searchable online help topics covering all aspects of the software and ASCII Command specifications.

Technical Support

We strive to make our manuals the best in the industry. We rely on your feedback to let us know if we are reaching our goal. If you cannot find the solution to your particular application, or, if for any reason you need technical assistance, please call us at:

<p align="center">**770–844–4200**</p>

Our technical support group will work with you to answer your questions. They are available Monday through Friday from 9:00 A.M. to 6:00 P.M. Eastern Time. We also encourage you to visit our web site where you can find technical and non-technical information about our products and our company.

<p align="center">**http://www.automationdirect.com**</p>

1

Before You Begin...

It is recommended that the following items be available to make this short step-by-step introduction to the ViewMarq LED message display go smoothly.

ViewMarq LED message display

AC Power Cord

DC Power Supply Example: PSP24-60S

or

ViewMarq User Manual and Installation Guide

ViewMarq Software installed on a PC (Windows XP SP3 minimum)

Ethernet Patch Cable (One 7 ft. cable supplied)

#1 Phillips Head Screwdriver and a Small Flat Head Screwdriver Examples: TW-SD-VPH-1 & TW-SD-VSL-1

Wire Strippers Example: DN-WS

Hookup Wire

Step 1 - Install Configuration Software

The following are the minimum system requirements for installing and running the ViewMarq Configuration Software on a PC:

- Personal computer with Windows XP (Service Pack 3), Vista, Windows 7 (32 and 64-bit) or Windows 8.
- Serial or Ethernet connection to ViewMarq display.

1) Insert the Software CD into the PC's CD-ROM drive.
   or
   Download the ViewMarq Configuration Software from the following AutomationDirect.com web site: **http://support.automationdirect.com/downloads.html** and UnZip the downloaded .zip file.

2) Double click ViewMarq Install.exe

3) The ViewMarq Configuration Software splash screen will appear after a short delay.

4) Click on the Install button and follow the instructions to install the ViewMarq Configuration Software.

1

Step 2 - Launch Configuration Software

After installing the ViewMarq Configuration Software, MD-PGMSW, from the PC's Start menu, select All Programs, AutomationDirect, ViewMarq and click the ViewMarq Configuration Software.

The ViewMarq Configuration Software will start up and display the Main Window as shown.

NOTE: The minimum screen size for the ViewMarq Configuration Software is 1024 X 768 pixels.
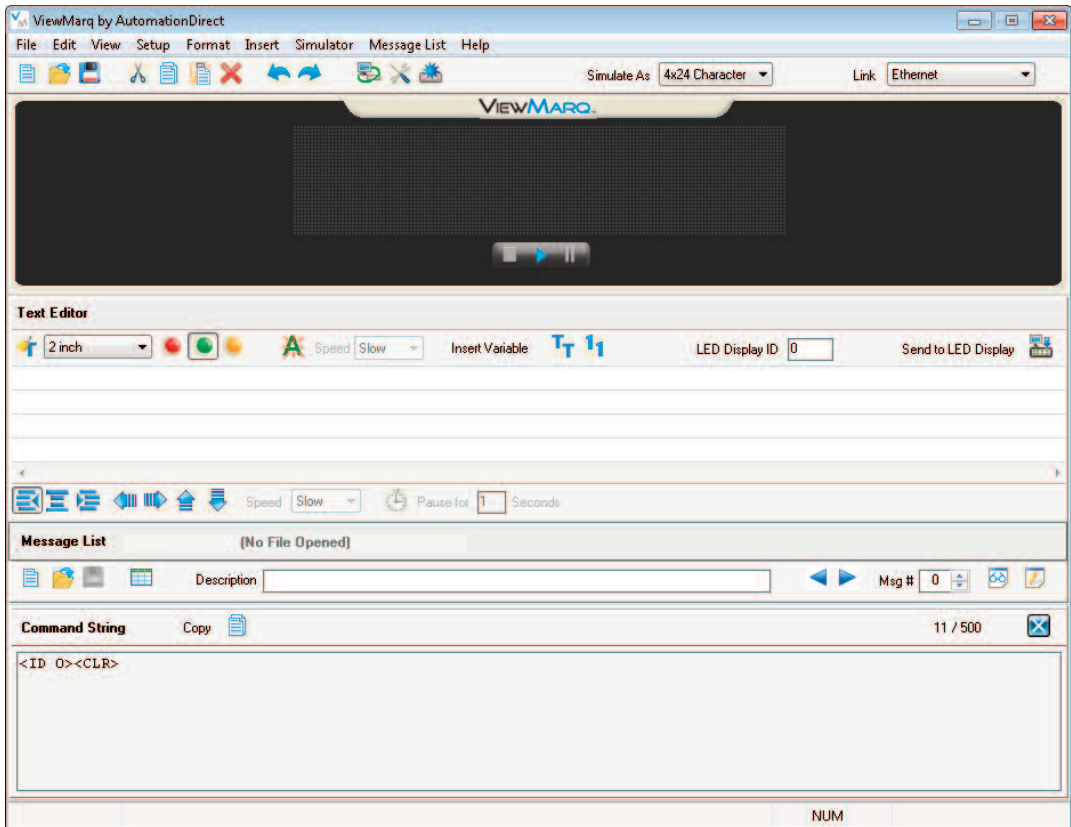
Step 3 - Connect Power Wiring

The ViewMarq LED message display can accept DC power (24 VDC) or AC power (120 or 240 VAC). There is a power terminal for each.

⚠ **Warning: Only one power source should be connected to the ViewMarq display. Observe all of the wiring precautions stated in this User Manual and the Installation Guide. For more details on wiring and installation refer to Chapter 3, Hardware Installation.**

Connect either DC or AC power per the following diagram. Be sure to connect Earth Ground to the Ground terminal regardless of the supply voltage. Wiring terminals should be torqued to 4.5 in·lbs (0.5 N·m).
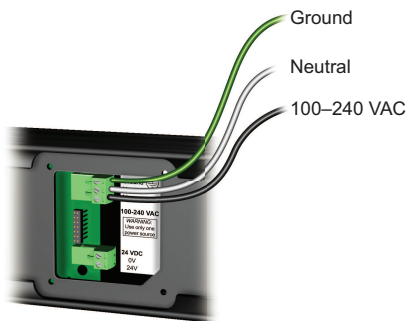
AC Power Input

DC Power Input

Model	Max Input Power
MD4-0112T	22W
MD4-0124T	38W
MD4-0212T	
MD4-0224T	74W
MD4-0412T	
MD4-0424T	123W

Model	Max DC Current
MD4-0112T	1A
MD4-0124T	1.5A
MD4-0212T	2A
MD4-0224T	3.5A
MD4-0412T	
MD4-0424T	4A

1
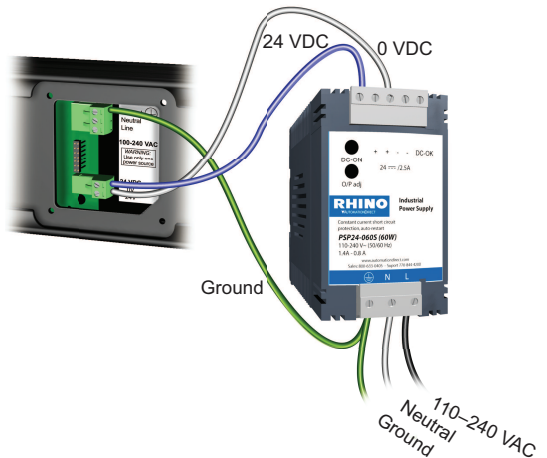
Step 4 - Connect the ViewMarq Message Display to a PC

The default settings of the ViewMarq LED display make connecting to a PC for the first time very simple. You can connect in one of two ways; Ethernet or Serial. We will cover Ethernet for this example. Refer to Chapter 4 – PC to ViewMarq Communication for other communications options.

Ethernet

The simplest way to connect the ViewMarq to a PC is through the Ethernet Port.

1) Using a straight-through or crossover Ethernet Patch cable, connect the ViewMarq to the PC's Ethernet port. The system will automatically detect the cable type.(One seven foot straight-through cable is supplied with the ViewMarq).

2) The ViewMarq LED display can also be connected to a local Ethernet network via a straight-through or crossover cable.
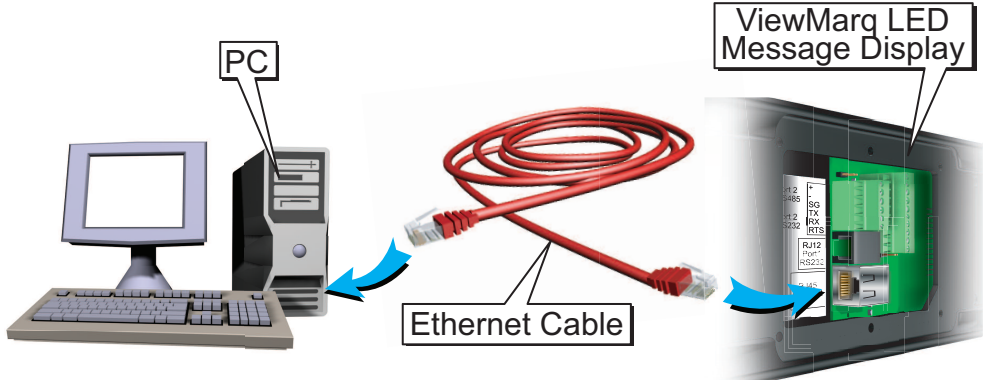
Step 5 - Apply Power to the Display

Turn on the power source and check for the Start-Up Message scrolling on the display.

Typical Start-Up Message

ViewMarq Display_438F FW AI V1.55 ** ETH 169.254.15.12** P1 AS 1 38400 8O1**P2 MB 1 38400 8O1**

** ViewMarq Display_438F - View Marq Identifier and Display Name

** FW AI V1.55 - Firmware Version

** ETH 169.254.15.12 - Ethernet IP Address

** P1 AS 1 38400 8O1 - Port 1 Settings
> AS - ASCII, MB - Modbus
> Node ID - 1 to 247
> Baudrate - 38400, 19200, 9600, 4800, 2400
> Data bits - 8 or 7
> Parity - O - Odd, E - Even, N - None
> Stop Bits - 1 or 2

**P2 MB 1 38400 8O1 - Port 2 Settings
> AS - ASCII, MB - Modbus
> Node ID - 1 to 247
> Baudrate - 38400, 19200, 9600, 4800, 2400
> Data bits - 8 or 7
> Parity - O - Odd, E - Even, N - None
> Stop Bits - 1 or 2

Refer to Chapter 5 for more information on the port settings for the display.

Step 6 - Create Link between the ViewMarq and your PC

1

Establishing Ethernet Communications with the ViewMarq Message Display

NOTE: See Chapter 4 for establishing serial communication with the ViewMarq message display.

First, select the Ethernet link:

1) From the Setup menu select **PC to LED Display Communications**.
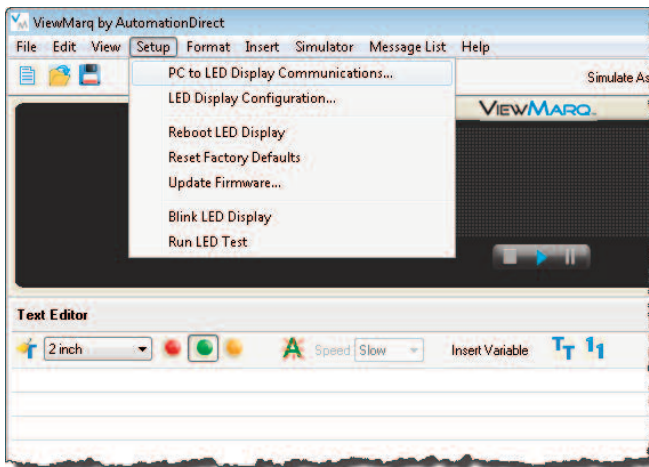
2) In the PC to LED Display Communications dialog box, select Ethernet in the Link List.
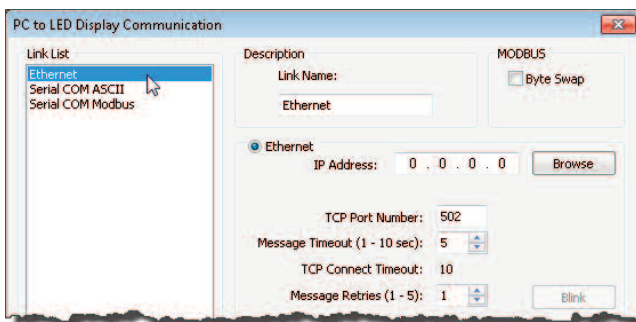
3) In the PC to LED Display Communication dialog box, select the **Browse** button to browse the network for any connected ViewMarq LED display.

4) The **Ethernet Browse** dialog box will open and the connected ViewMarq LED display will be listed as shown below.
   If the ViewMarq does not show up on the list:

          A. Check the physical connection.

          B. Check that the Ethernet port on the PC is enabled.

   If the PC has multiple Network Interface cards, disable those not in use.
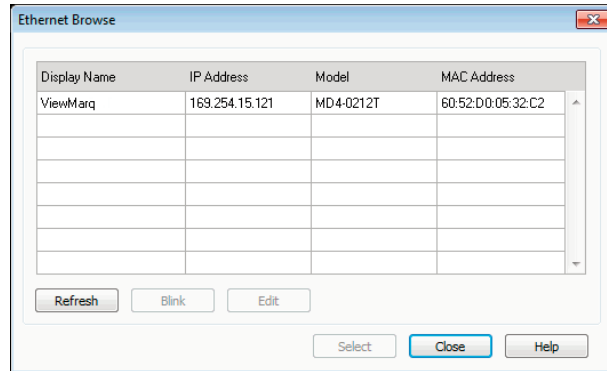
5) Select the ViewMarq LED message display in the table.

6) If the IP Address is correct on your local Subnet, click **Select** and **proceed to step 14.**

NOTE: *If the IP Address of the ViewMarq is 169.254.XXX.nnn then no DHCP server was found to issue an IP Address. Check the IP Address of the ViewMarq and set the IP Address of the ViewMarq to be on the same subnet as the PC.*

7) If the IP Address is not on your Subnet it must be changed, select it and click **Edit** to open the **Ethernet Browse Editor**.
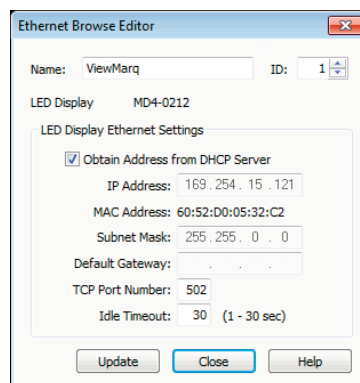
8) Deselect **Obtain IP Address from DHCP Server**.

9) Enter an IP Address, Subnet Mask and Default Gateway compatible with your PC's Ethernet settings or check with your IT department for proper settings.

10) You may also choose to change the name of the ViewMarq display and ID number at this time.

11) Select **Update**. The ViewMarq LED display will automatically reset. The new settings will scroll on the screen.

12) In the **Browse** window select the **Refresh** button.

13) When the ViewMarq is listed in the table again select it and click **Select**.
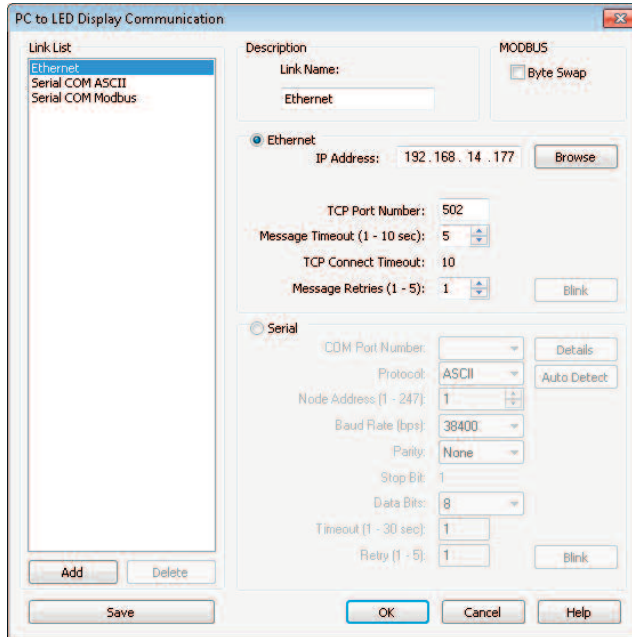
14) Select **Save** and **OK**

You have now established an Ethernet link from your PC to your ViewMarq LED display. You are ready to send messages to your ViewMarq.

For more information on connecting to the ViewMarq display and creating links, see Chapter 4 - PC to ViewMarq Communication.

Step 7 - Creating Messages

Creating a new message in the ViewMarq software is simple, just start typing.

1) To create a message, start typing in the Text Editor window.
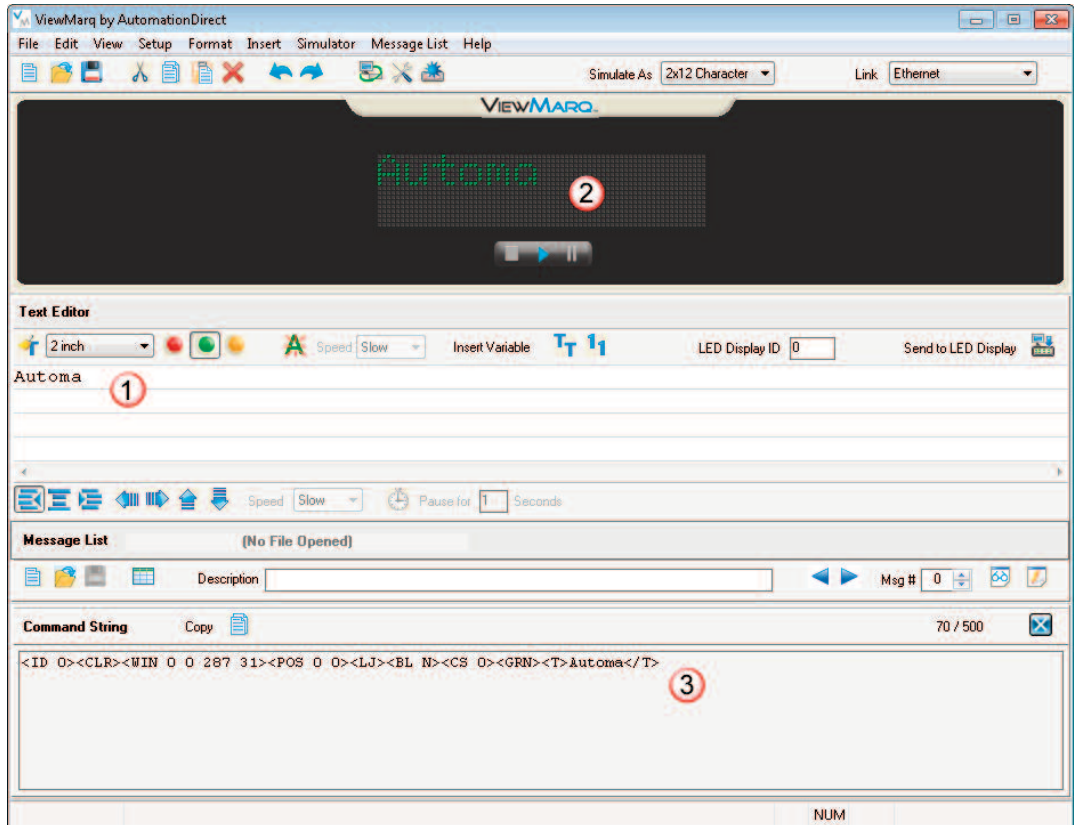
2) As text is entered, it is automatically displayed in the simulator.

3) Also, an ASCII formatted string is automatically created for use in your PLC instruction.

To get more attention with your message, you may change the Text size, Color and / or make it Blink. You can also choose to justify the text to the Center or Right or make the text Scroll. Experiment with the different effects that can be used to modify and display your text in creative ways.

For more information on Formatting Messages, see Chapter 5 - Creating and Previewing Messages.

NOTE: The <ID 1> is set by entering "1" in the LED Display ID box in the Text Editor window.
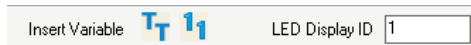
Step 8 - Simulating Messages

1

The ViewMarq Simulator shows all text colors, blinking text, scrolling messages and inserted variables just as they appear on the ViewMarq display.

> **NOTE:** Shades of color, blink rates and scroll speeds may vary depending on your PC environment.

To see what your message will look like on the ViewMarq display you can simulate it as if it were on the particular model that you have. In the example below, the model selected is a MD4-0424T, the 4-line by 24-character tri-color model.
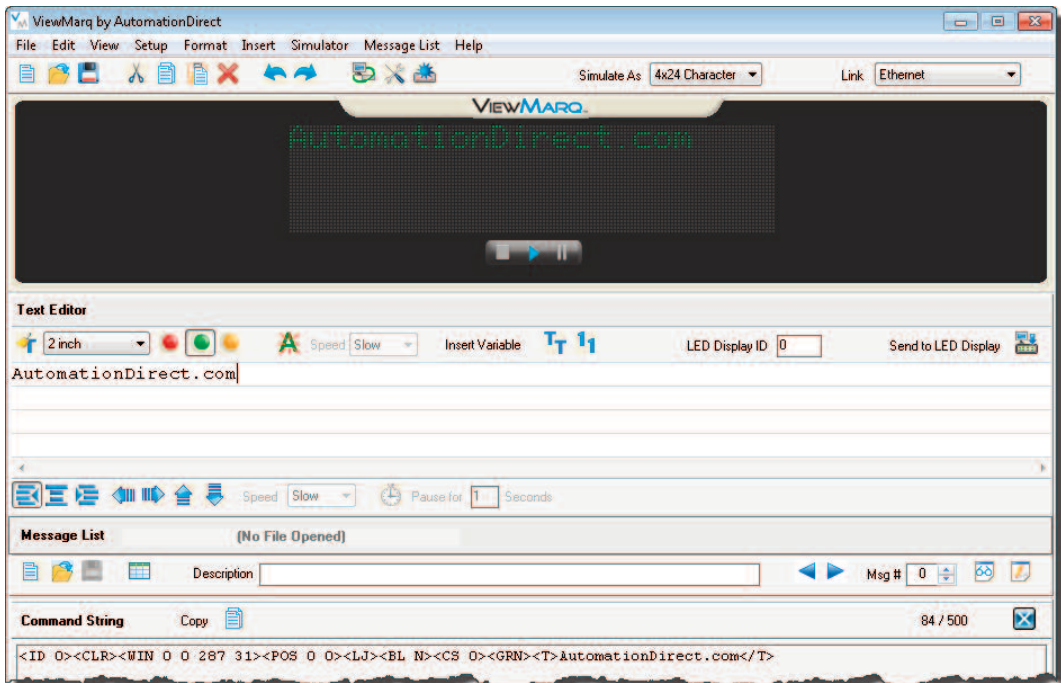
Stop / Play

By default the Simulator is in **Play** mode. This means that any motion such as scroll or blink will be seen in the Simulator as it will look on a message display. Selecting **Stop** will stop the Simulator from running until **Play** is selected. When **Play** is selected, the simulation will start over from the beginning. Even when it is stopped, the Simulator will still show text as it is typed.

Pause

When the Simulator is in **Play** mode and **Pause** is selected, the simulation will stop where it is. When **Pause** is selected again, the simulation will continue where it left off. If the message is modified while paused, the simulator will clear until un-paused.

Step 9 - Sending Messages to the ViewMarq Message Display

Now that you have:

1) Established a Link with your ViewMarq

2) Created a message

3) Previewed it on the Simulator,

it is time to display the message on the ViewMarq message display.

With your Link selected simply press the **Send to LED Display** Button below the Simulator.
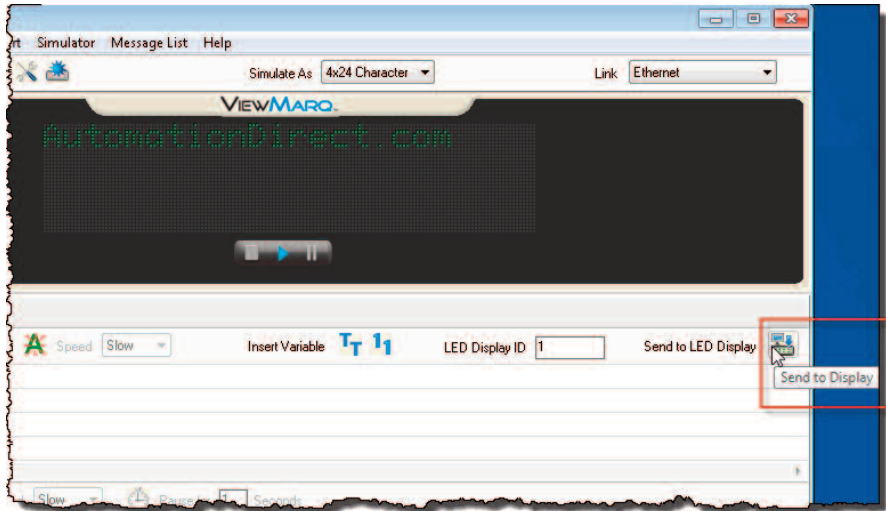
Your text should be displayed on the LED message display.

1

Step 10 - Sending Messages from your PLC to the ViewMarq

Sending messages one at a time from the PC software is not always convenient. It is much more common to use your control system to display a series of messages or to control messages/data to be displayed based on your control system conditions.

As you create a message in the ViewMarq software, the software creates an ASCII string of commands that the ViewMarq recognizes to display the message you created.

Command String	📋 Copy	86 / 500	✖

`<ID 1><CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N><CS 0><GRN><T>Hello World</T>`

CLR NUM
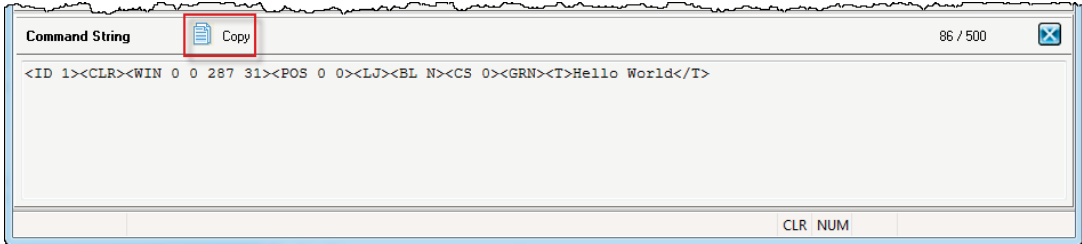
This string may be copied to the PC clipboard and pasted into your PLC's instruction or memory, and sent either serially or over Ethernet to the ViewMarq.

The ViewMarq can receive an ASCII Command String by:

Protocol	Connection
ASCII	RS-232 (Port 1), RS485 (Port 2)
Modbus RTU	RS-232 (Port 1), RS485 (Port 2)
Modbus TCP	Ethernet
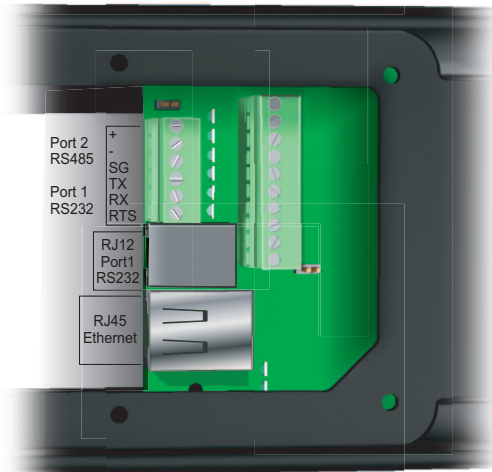
The following example shows you how to send messages to the ViewMarq from an AutomationDirect CLICK PLC using RS-232 and the ASCII protocol.

See Chapter 7 - Sending Messages from a PLC to ViewMarq for more information.

1) Using the CLICK Programming Software set Port 2 on the CLICK PLC Com Port to match the default settings for the ViewMarq LED message display as shown below:
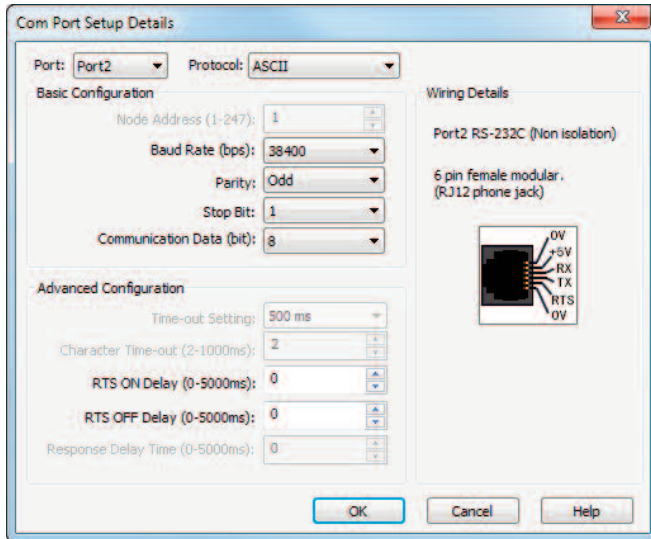
Protocol: ASCII
Baud Rate: 38400
Parity: Odd
Stop Bit: 1
Data Bits: 8

2) In the ViewMarq Software, type and configure a message. Copy the string in the Message String window by selecting Copy on the Message String Toolbar.

1
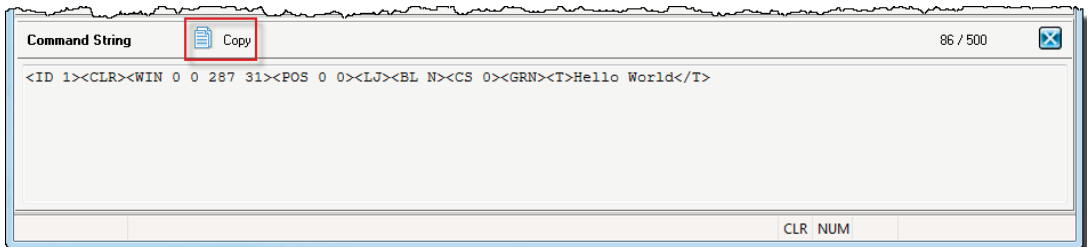
3) In the CLICK Programming software use a SEND instruction as shown below.
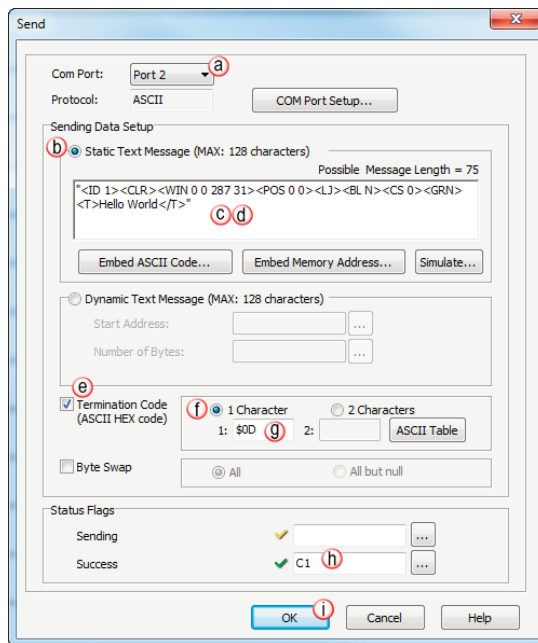
a) Select Port 2 for RS232 (or 3 for RS 485) for the Com Port.

b) Select the Static Text Message radio button

c) Click in the Text Box and press Ctrl-V on your keyboard to paste the command string into the instruction.

d) Add quotes to the beginning and end of the string.

e) Select the Termination Code checkbox

f) Select 1 Character radio button

g) Enter $0D (Dollar Zero Dee) to embed a Carriage Return at the end of the string.

h) Select an address for the success bit. For example C1.

i) Select OK

```
            C10                  Send (Port:2)              ASCII       C1
            ┤↑├                  Type                       Static       ○
                                 Byte Swap                   None    Success
                                 Terminate Code                CR
                                 Send Message
                                 "<ID 1><CLR><WIN 0 0 287 31><POS
                                       0 0><LJ><BL N><CS
                                    0><GRN><T>Hello World</T>"
```

NOTE: *To prevent the string from being sent with every scan of the PLC use an EDGE triggered (or One Shot) instruction.*

SPECIFICATIONS

CHAPTER
2

In This Chapter...

Overview

The ViewMarq LED display is a tri-color completely enclosed unit that is intended to be controlled over serial RS-232/485 or Ethernet networks by a PLC utilizing ASCII or Modbus. It is configured using the ViewMarq Software running on a PC over either RS-232/485 or Ethernet.

The ViewMarq is a fully enclosed NEMA 4/12 indoor use only unit, hose-down rated, corrosion resistant display. When the supplied wall mount brackets are used and customer supplied flexible conduit is used for all cables, the message display meets UL508 requirements.

The display may be powered by either 24 VDC or 120/240 VAC nominally.

NOTE: *The ViewMarq LED message display can accept either DC power (24 VDC) or AC power (120 or 240 VAC) but not both simultaneously.*

Specifications

ViewMarq 1-Line Message Displays Specifications		
Part Number	MD4-0112T	MD4-0124T
Description	1 Line X 12 Character	1 Line X 24 Character
Display		
• **Type**	LED matrix: Tri-color (Green, Red, Amber)	
• **LED Life**	100,000 hours	
• **LED Pitch**	0.19 in [5 mm]	
• **Center to Center Spacing of each LED**	0.27 in [7 mm]	
• **Number of Lines**	1 (Based on 5 x 7 pixel sized characters)	
• **Text Height**	1.25 in, 2 in (Depends on character set selected)	
• **Character Set**	English	
• **Approximate Viewing Distance**	100 ft	
AC Electrical		
• **Input Voltage**	100-240 VAC (+10% / -15%), 50 / 60 Hz	
• **Maximum Current**	0.5A	0.8A
• **Power Consumption (MAX)**	22W	38W
• **Maximum Inrush Current (Cold and Hot)**	25A	26A
• **Input Fuse Protection (Internal)**	Yes (not user replaceable)	
• **Output Protection for Over-current, Over Voltage and Over Temperature**	Yes – Self Resetting	
• **Insulation Resistance**	> 10 MOhms @ 500 VAC	
DC Electrical		
• **Input Voltage**	24 VDC (+10% / -15%)	
• **Maximum Current**	1A	1.5 A
• **Maximum Inrush Current (Cold and Hot)**	3A	4.5 A
• **Input Fuse Protection (Internal)**	Yes (not user replaceable)	
• **Reverse Polarity Protection**	Yes	
Dimensions / Weight		
• **Enclosure Approximate Weight**	4.9 lbs [2.2 kg]	9.0 lbs [4.1 kg]

2

ViewMarq 2-Line Message Displays Specifications		
Part Number	MD4-0212T	MD4-0224T
Description	2 Line X 12 Character	2 Line X 24 Character
Display		
• **Type**	LED matrix: Tri-color (Green, Red, Amber)	
• **LED Life**	100,000 hours	
• **LED Pitch**	0.19 in [5 mm]	
• **Center to Center Spacing of each LED**	0.27 in [7 mm]	
• **Number of Lines**	2 (Based on 5 x 7 pixel sized characters)	
• **Text Height**	1.25 in, 2 in, 4 in (Depends on character set selected)	
• **Character Set**	English	
• **Approximate Viewing Distance**	200 ft	
AC Electrical		
• **Input Voltage**	100-240 VAC (+10% / -15%), 50 / 60 Hz	
• **Maximum Current**	1A	2A
• **Power Consumption (MAX)**	38W	74W
• **Maximum Inrush Current (Cold and Hot)**	26A	24A
• **Input Fuse Protection (Internal)**	Yes (not user replaceable)	
• **Output Protection for Over-current, Over Voltage and Over Temperature**	Yes – Self Resetting	
• **Insulation Resistance**	> 10 MOhms @ 500 VAC	
DC Electrical		
• **Input Voltage**	24 VDC (+10% / -15%)	
• **Maximum Current**	2A	3.5A
• **Maximum Inrush Current (Cold and Hot)**	10.5A	
• **Input Fuse Protection (Internal)**	Yes (not user replaceable)	
• **Reverse Polarity Protection**	Yes	
Dimensions / Weight		
• **Enclosure Approximate Weight**	7.3 lbs [3.3 kg]	13.1 lbs [6.0 kg]

Specifications continued on the next page...

2

ViewMarq 4-Line Message Displays Specifications		
Part Number	MD4-0412T	MD4-0424T
Description	4 Line X 12 Character	4 Line X 24 Character
Display		
• **Type**	LED matrix: Tri-color (Green, Red, Amber)	
• **LED Life**	100,000 hours	
• **LED Pitch**	0.19 in [5 mm]	
• **Center to Center Spacing of each LED**	0.27 in [7 mm]	
• **Number of Lines**	4 (Based on 5 x 7 pixel sized characters)	
• **Text Height**	1.25 in, 2 in, 4 in, 6 in, 8 in (Depends on character set selected)	
• **Character Set**	English	
• **Approximate Viewing Distance**	400 ft	
AC Electrical		
• **Input Voltage**	100-240 VAC (+10% / -15%), 50 / 60 Hz	
• **Maximum Current**	2A	2A
• **Power Consumption (MAX)**	74W	123W
• **Maximum Inrush Current (Cold and Hot)**	24A	9.5A
• **Input Fuse Protection (Internal)**	Yes (not user replaceable)	
• **Output Protection for Overcurrent, Over Voltage and Over Temperature**	Yes – Self Resetting	
• **Insulation Resistance**	> 10 MOhms @ 500 VAC	
DC Electrical		
• **Input Voltage**	24 VDC (+10% / -15%)	
• **Maximum Current**	3.5A	4A
• **Maximum Inrush Current (Cold and Hot)**	10.5A	12A
• **Input Fuse Protection (Internal)**	Yes (not user replaceable)	
• **Reverse Polarity Protection**	Yes	
Dimensions / Weight		
• **Enclosure Approximate Weight**	12.1 lbs [5.5 kg]	22.5 lbs [10.2 kg]

Specifications continued on the next page...

Environmental Specifications

ViewMarq Message Displays Specifications	
Environmental	
• *Storage Temperature*	−30 to +85 °C (−22 to +185 °F)
• *Operating Temperature*	0 to 60 °C (32 to +140 °F)
• *Humidity*	5 – 95% non-condensing
• *Enclosure Ratings*	NEMA 12 indoor only, NEMA 4 indoor only
• *Vibration*	IEC 60068-2-6 (Test Fc)
• *Shock*	IEC 60068-2-27 (Test Ea)
• *Noise Immunity*	EN61131-2:2007
• *Agency Approvals*	UL508, NEMA 4, NEMA 12, RoHS, REACH CE (EN61131-2:2007) - Pollution Degree 2, Overvoltage Category II

Power Supply and Communication Specifications

Power Supply Removable Terminal Blocks

Power Supply Removable Terminal Blocks							
Part Number	*Terminal*	*Connector*	*Wire Size*	*Screw Torque*	*Voltage Rating*	*Current Rating*	*Temperature Rating*
MD-TERM-SET	AC Power	Removable 3-pin terminal block	12-14 AWG solid or stranded	4.5 in·lbs (0.5 N·m)	300V	15A	105 °C (221 °F)
	DC Power	Removable 2-pin terminal block					
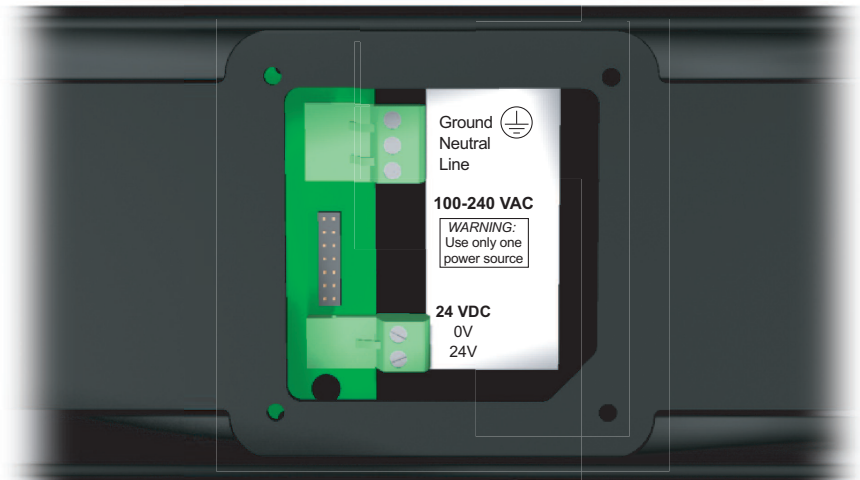
AC Power Supply Specifications

AC Power Supply General Specifications						
Part Number	MD4-0112T	MD4-0124T	MD4-0212T	MD4-0224T	MD4-0412T	MD4-0424T
Input Voltage	100-240 VAC (+10% / -15%), 50 / 60 Hz					
Maximum Current	0.5A	0.8A	1A	2A		
Power Comsumption (MAX)	22W	38W		74W		123W
Maximum Inrush Current (Cold and Hot)	25A	26A		24A		9.5A
Input Fuse Protection (Internal)	Yes (not user replaceable)					
Output Protection for Overcurrent, Over Voltage and Over Temperature	Yes – Self Resetting					
Under Input Voltage Lockout	25-60 VAC	40-65 VAC		40-70 VAC		30-70 VAC
Over Input Voltage Lockout	No	265-285 VAC				
Input Transient Protection	Varistor Plus Input Choke and Passive Filter					Active Power Factor Correction
Voltage Withstand (Dielectric)	AC Mains to SELV SELV = Secondary Extra Low Voltage (RS232, RS485, Ethernet)					3250 VDC / 2 Seconds
Insulation Resistance	> 10 MOhms @ 500 VAC					

External DC Power Supply Requirements

DC Power Supply General Specifications						
Part Number	MD4-0112T	MD4-0124T	MD4-0212T	MD4-0224T	MD4-0412T	MD4-0424T
Input Voltage	24 VDC (+10% / -15%)					
Maximum Current	1A	1.5A	2A	3.5A		4A
Maximum Inrush Current (Cold and Hot)	3A	4.5A	10.5A			12A
Reverse Polarity Protection	Yes					

2

Communication and Discrete Input Connections
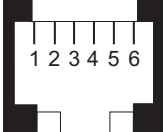
The following table describes the communication connections available in the ViewMarq display.

Communications and Discrete Input Connections				
Port	**Type**	**Connector**	**Wire Size**	**Screw Torque**
Port 1	RS-232	RJ12	n/a	
Port 1 / Port 2	RS-232 / RS-485	Removable 6-pin terminal block	14-28 AWG	1.7 in·lbs (0.2 N·m)
Port 3	Ethernet	RJ45	n/a	
Port 4	Discrete Inputs	Removable 10-pin terminal block	14-28 AWG	1.7 in·lbs (0.2 N·m)

RS-232 Port 1 RJ12 Connector

Port1 - RS-232 RJ12 Connector		
Description	**Specification**	
Designation	Port 1	
Serial Communications	Physical Layer: Non-isolated, Conforms to RS-232 Communications Interface	
Communication Port Settings	2400, 9600,19200, 38400 Data Length: 7/8 bits, Stop Bit: 1 bit, Parity: None, Even/Odd, No RTS signal	
Connector Type	Female RJ12 1 2 3 4 5 6	**Pin Number** **Signal Name** 1 Signal GND 2 Not Used 3 RXD 4 TXD 5 Not Used 6 Signal GND

Port 1 RS232 Terminal Block Connection

Port 1 - RS-232 Terminal Block	
Description	**Specification**
Designation	Port 1
Serial Communications	Physical Layer: Non-isolated, Conforms to RS-232 Communications Interface
Communication Port Settings	2400, 9600,19200, 38400 Data Length: 7/8 bits, Stop Bit: 1 bit, Parity: None, Even/Odd, RTS signal on pin 6
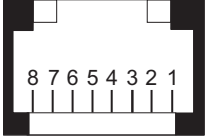
Port 2 RS-485 Terminal Block Connection

Port 2 - RS-485 Terminal Block	
Description	**Specification**
Designation	Port 2
Serial Communications	Physical Layer: Non-isolated, Conforms to RS485 Communications Interface
Communication Port Settings	9600,19200, 38400 Data Length: 7/8 bits, Stop Bit: 1 bit, Parity: None, Even/Odd
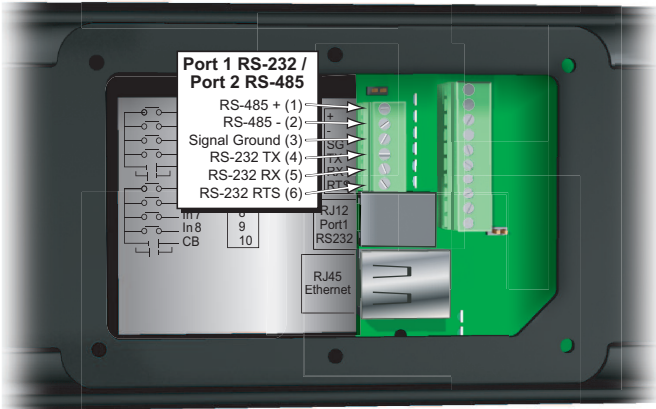
6 Position Terminal Block Pin Number	Signal Name
1	RS-485 +
2	RS-485 -
3	Signal Ground
4	RS-232 TX
5	RS-232 RX
6	RS-232 RTS

Port 3 Ethernet Connection

Port 3 - Ethernet		
Description	**Specification**	
Standard Specification	Conforms to IEEE802.3	
Communication Speed	10/100BASE-T(auto crossover)	
Cable Specification	Cat5e	
Physical Jack	RJ45, Module jack	
Connector Type	Female RJ45	Pin Number	Signal Name
		1	TD+
		2	TD-
		3	RD+
	8 7 6 5 4 3 2 1	4	No Connection
		5	No Connection
		6	RD-
		7	No Connection
		8	No Connection

Port 3 RJ45
(Ethernet)

8 7 6 5 4 3 2 1

2
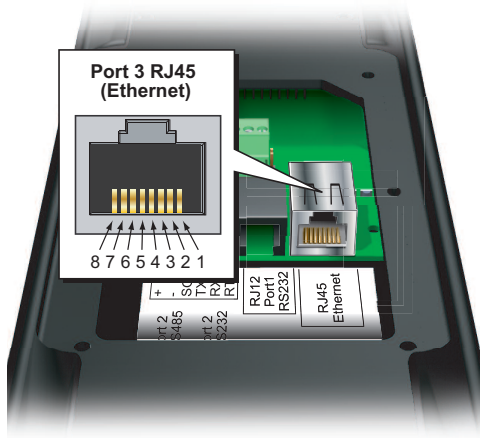
Discrete Input Specifications

Discrete Inputs	
Description	*Specification*
Inputs per Module	8 (sinking / sourcing)
Commons per Module	2 isolated (4 inputs per common)
Input Voltage Range	10-28 VDC
ON Voltage / Current Level	9 VDC / 3mA
OFF Voltage / Current Level	4 VDC / 0.5 mA
Maximum Input Current	13 mA @ 28 VDC
Input Impedance	2.2 kohm @ 24 VDC
OFF to ON Response	2 to 9ms
ON to OFF Response	2 to 9ms
Peak Voltage	30 VDC

10 Position Terminal Block Pin Number	*Signal Name*
1	Input 1
2	Input 2
3	Input 3
4	Input 4
5	Common A
6	Input 5
7	Input 6
8	Input 7
9	Input 8
10	Common B

Sourcing Input Wiring

Sinking Input Wiring
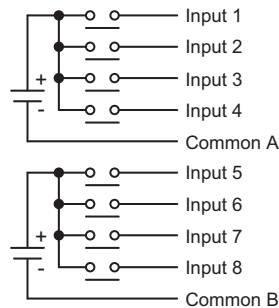
NOTE: *In order to maintain UL508 rating, discrete inputs must be powered from a Class 2 power supply.*

2

Communication LEDs

Communication LEDs			
LED	**Color**	**Silkscreen Label**	**Meaning**
Ethernet Speed		SPD	OFF = 10M Connection, ON = 100M Connection
Ethernet Link Status Condition		LNK	OFF - No link, ON = Link, Blinking = Network Activity
RS-232 Request to Send		RTS	Blinking = RS232 is attempting to establish connection
RS-232 Receive Data	Green	RX1	Blinking = Receiving Data
RS-232 Transmit Data		TX1	Blinking = Transmitting Data
RS-485 Receive Data		RX2	Blinking = Receiving Data
RS-485 Transmit Data		TX2	Blinking = Transmitting Data
Error		ERR	Communication Error

2
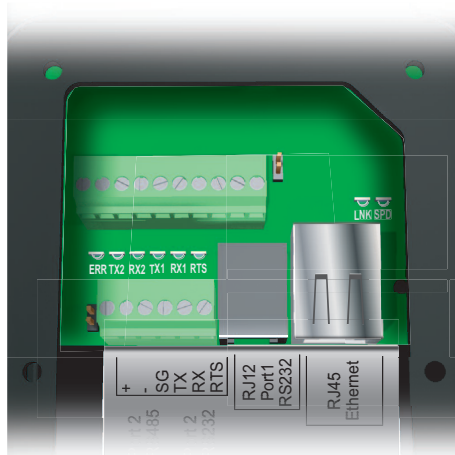
Mechanical Features

Front View

End Cap

Aluminum Extrusion

Lens

End Cap

Back View

End Cap

Aluminum Extrusion

End Cap

Power Connection Opening

Communication Connection Opening
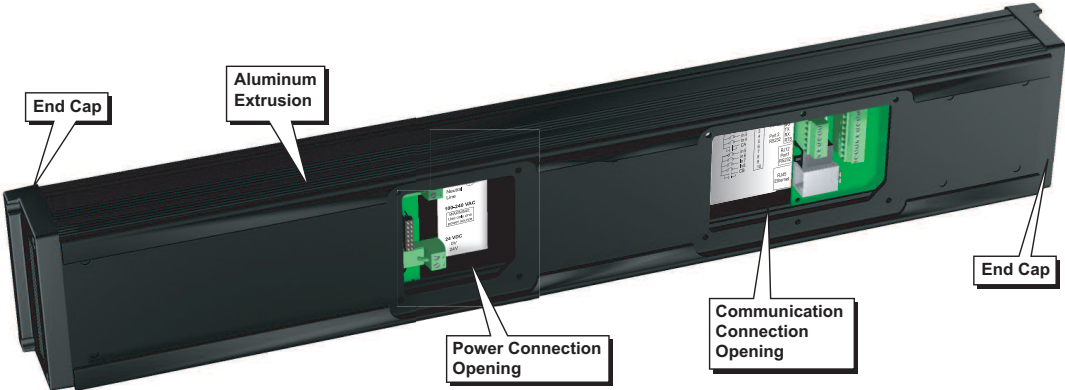
ViewMarq Housing Materials	
Aluminum Extrusion	AL6063-T5 Black Anodized
End Caps	Polycarbonate
End Cap Gaskets	Silicon
Back Covers	Polycarbonate
Back Cover Gaskets	Silicon
Lens	Polycarbonate
Lens Seal	Silicon

Dimensional Drawings

MD4-0112T and MD4-0124T

Dimensions: in [mm]

2

DIMENSION TABLE

PART NO.	DIM "A"	DIM "B" MAX
MD4-0112T	23.30 [591.8]	20.77 [527.6]
MD4-0124T	44.89 [1140.2]	42.38 [1076.5]

2.66 [67.6]

3.77 [95.8]

6.32 [160.5]

5.20 [132.1]

8X Ø0.28 [Ø7.1]

"A"

"B" MAX

2
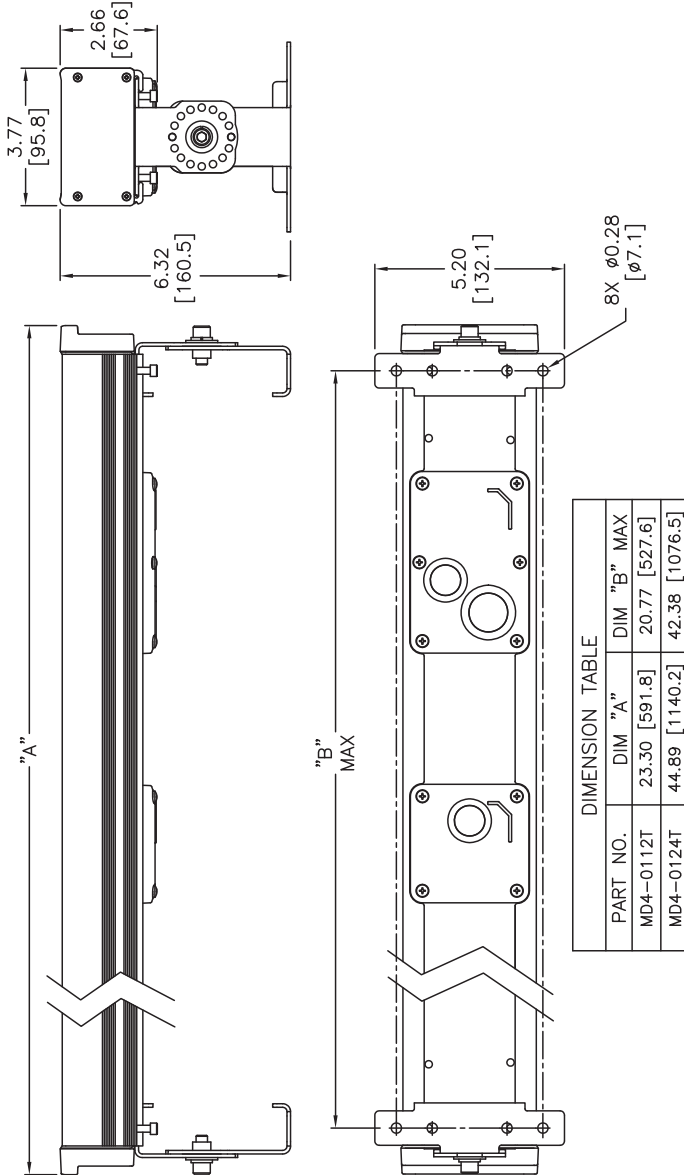
MD4-0212T and MD4-0224T

Dimensions: in [mm]
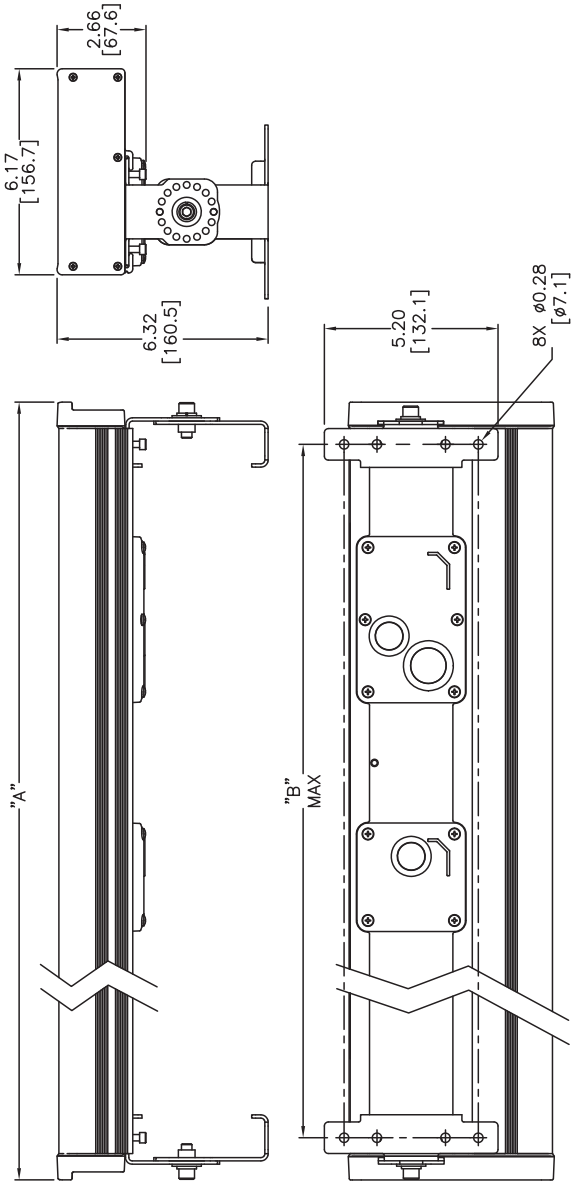
DIMENSION TABLE		
PART NO.	DIM "A"	DIM "B" MAX
MD4-0212T	23.30 [591.8]	20.77 [527.6]
MD4-0224T	44.89 [1140.2]	42.38 [1076.5]

2

MD4-0412T and MD4-0424T

Dimensions: in [mm]

DIMENSION TABLE		
PART NO.	DIM "A"	DIM "B" MAX
MD4-0412T	23.30 [591.8]	20.77 [527.6]
MD4-0424T	44.89 [1140.2]	42.38 [1076.5]

2.66 [67.6]

10.97 [278.6]

6.32 [160.5]

5.20 [132.1]

8X Ø0.28 [Ø7.1]

"A"

"B" MAX

HARDWARE INSTALLATION

In This Chapter...

Safety Guidelines

NOTE: Products with CE marks perform their required functions safely and adhere to relevant standards as specified by CE directives, provided they are used according to their intended purpose and that the instructions in this manual are followed. The protection provided by the equipment may be impaired if this equipment is used in a manner not specified in this manual. A listing of our international affiliates is available on our web site at http://www.automationdirect.com.

Warning: Providing a safe operating environment for personnel and equipment is your responsibility and should be your primary goal during system planning and installation. Automation systems can fail and may result in situations that can cause serious injury to personnel or damage to equipment. Do not rely on the automation system alone to provide a safe operating environment. You should use external electromechanical devices, such as relays or limit switches, that are independent of the PLC application to provide protection for any part of the system that may cause personal injury or damage. Every automation application is different, so there may be special requirements for your particular application. Make sure you follow all national, state, and local government requirements for the proper installation and use of your equipment.

Plan for Safety

The best way to provide a safe operating environment is to make personnel and equipment safety part of the planning process. You should examine every aspect of the system to determine which areas are critical to operator or machine safety. If you are not familiar with PLC system installation practices, or your company does not have established installation guidelines, you should obtain additional information from the following sources.

- NEMA — The National Electrical Manufacturers Association, located in Washington, D.C., publishes many different documents that discuss standards for industrial control systems. You can order these publications directly from NEMA. Some of these include:
    ICS 1, General Standards for Industrial Control and Systems
    ICS 3, Industrial Systems
    ICS 6, Enclosures for Industrial Control Systems

- NEC — The National Electrical Code provides regulations concerning the installation and use of various types of electrical equipment. Copies of the NEC Handbook can often be obtained from your local electrical equipment distributor or your local library.

- Local and State Agencies — many local governments and state governments have additional requirements above and beyond those described in the NEC Handbook. Check with your local Electrical Inspector or Fire Marshall office for information.

Introduction to ViewMarq Mechanical Design

All ViewMarq displays are similar in appearance. They differ only in size and aspect ratio. The mounting brackets are identical as are the back covers and all of the power and communication connections.

The diagram below will allow you to familiarize yourself with the two main compartments, power and communications.
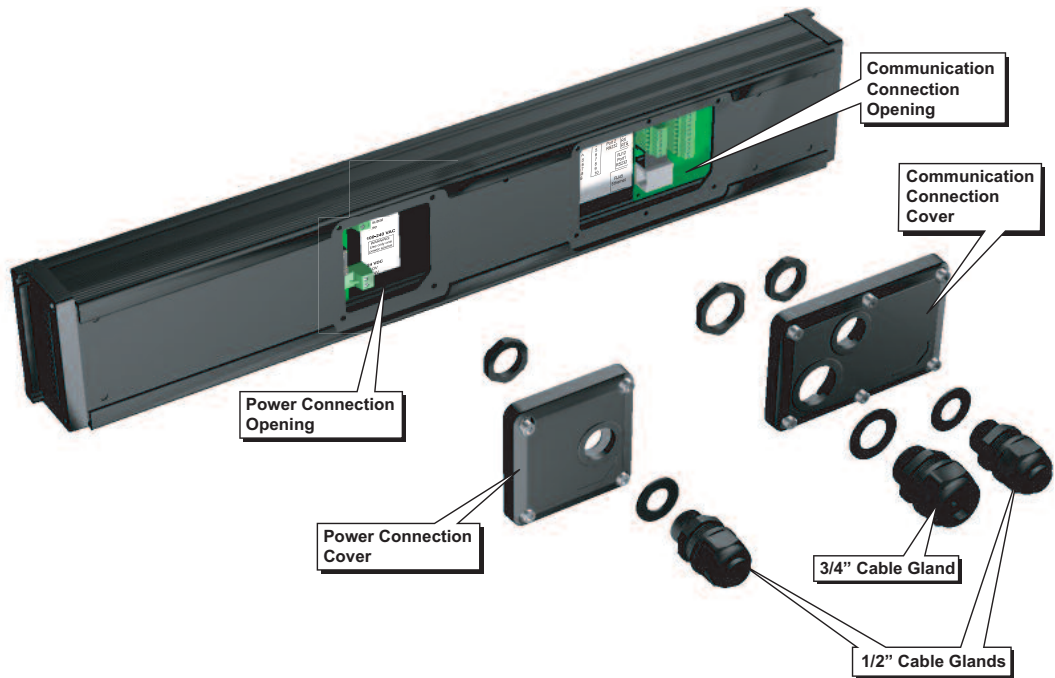
Power Connections and Specifications

Power Connections Cover

The power connections cover has a 0.83 in (21.1 mm) diameter opening for the supplied 1/2"
NPT water tight cable gland or customer supplied flexible conduit connection for the power
cable. The power connections cover must be installed for all applications. Cover screws should
be torqued to 10 in·lbs (1.1 N·m).

NOTE: *The supplied cable glands cannot be used for UL installations. Flexible Conduit to protect cables is required to maintain UL508 rating.*

Power Installation

The ViewMarq message display can be powered by 100-240 VAC or by a user supplied external
24 VDC power supply.

Warning: Do not connect both AC and DC power at the same time. Internal circuit protection will prevent damage to the message display, but this is an invalid configuration and is not UL compliant.

Power Supply Removable Terminal Blocks

There is a removable 2-pin 24 VDC and a removable 3-pin 120 VAC power connection terminal block supplied with each ViewMarq message display. Replacement terminal blocks, Part No. MD-TERM-SET, are available at *AutomationDirect.com*.

Power Supply Removable Terminal Blocks				
Part Number	*Terminal*	*Connector*	*Wire Size*	*Screw Torque*
MD-TERM-SET	AC Power	Removable 3-pin terminal block	12-14 AWG solid or stranded	4.5 in·lbs (0.5 N·m)
	DC Power	Removable 2-pin terminal block		

3
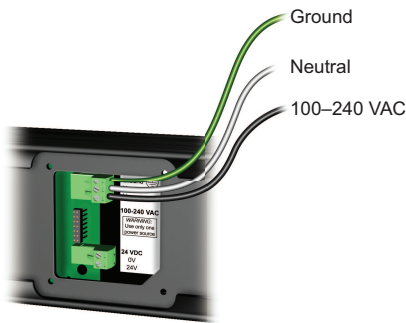
AC Power Input

Model	Max Input Power
MD4-0112T	22W
MD4-0124T	38W
MD4-0212T	
MD4-0224T	74W
MD4-0412T	
MD4-0424T	123W
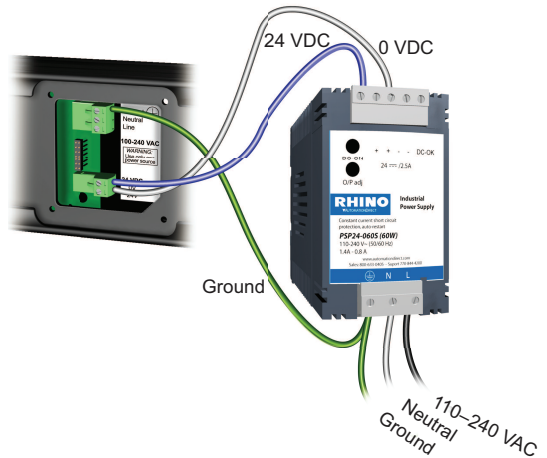
DC Power Input

Model	Max DC Current
MD4-0112T	1A
MD4-0124T	1.5 A
MD4-0212T	2A
MD4-0224T	3.5A
MD4-0412T	
MD4-0424T	4A

3

Grounding

The ground terminal on the ViewMarq must be connected to a single point ground. Use copper stranded wire to achieve low impedance.

A good common ground reference (Earth ground) is essential for proper operation of the ViewMarq. One side of all control and power circuits and the ground lead on flexible shielded cable must be properly connected to Earth ground. There are several methods of providing an adequate common ground reference, including:

a) Installing a ground rod as close to the panel as possible

b) Connection to incoming power system ground

Ambient Temperature

Evaluate any installations where the ambient temperature may approach the lower or upper limits of the specifications.

Communications Connections

Communication Connections Cover

The communication connections cover has a 0.83 in (21.1 mm) diameter and a 1.06 in (26.9 mm) diameter opening for the supplied 1/2" NPT and 3/4" NPT water tight cable glands or customer supplied flexible conduit connections for communications. The communication connections cover must be installed for all applications. Cover screws should be torqued to 10 in·lbs (1.1 N·m).

3

> **NOTE:** The supplied cable glands cannot be used for UL installations. Flexible Conduit to protect cables is required to maintain UL508 rating.

Communication Connectors
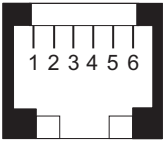
The following table describes the communication connections available in the ViewMarq display.

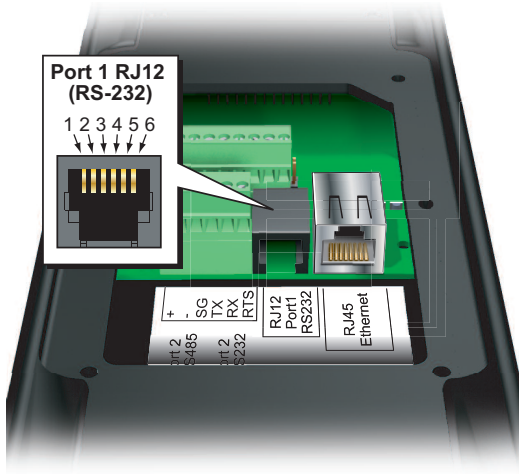| Communications and Discrete Input Connections				
Port	*Type*	*Connector*	*Wire Size*	*Screw Torque*
Port 1	RS-232	RJ12	n/a	
Port 1 / Port 2	RS-232 / RS-485	Removable 6-pin terminal block	14-28 AWG	1.7 in·lbs (0.2 N·m)
Port 3	Ethernet	RJ45	n/a	
Port 4	Discrete Inputs	Removable 10-pin terminal block	14-28 AWG	1.7 in·lbs (0.2 N·m)

RS-232 Port 1 RJ12 Connector

Port1 - RS-232 RJ12 Connector		
Female RJ12	Pin Number	Signal Name
	1	Signal GND
	2	Not Used
	3	RXD
	4	TXD
	5	Not Used
	6	Signal GND

3

Port 1 RS-232 / Port 2 RS-485 Terminal Block Connection

6 Position Terminal Block Pin Number	Signal Name
1	RS-485 +
2	RS-485 -
3	Signal Ground
4	RS-232 TX
5	RS-232 RX
6	RS-232 RTS

Port 3 Ethernet Connection

Port 3 RJ45 (Ethernet)

8 7 6 5 4 3 2 1
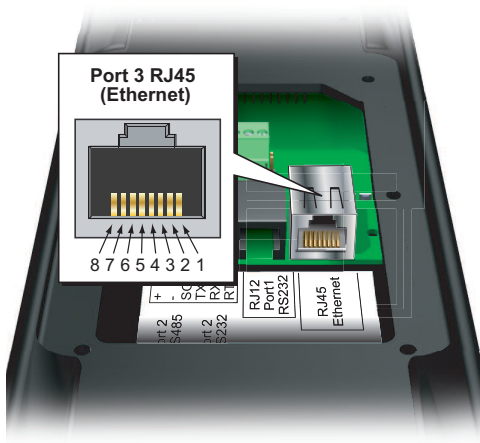
RJ45 Ethernet Connector Pin Number	Signal Name
1	TD+
2	TD-
3	RD+
4	No Connection
5	No Connection
6	RD-
7	No Connection
8	No Connection

Port 4 Discrete Inputs

Discrete Input
(1) Input 1
(2) Input 2
(3) Input 3
(4) Input 4
(5) Common A
(6) Input 5
(7) Input 6
(8) Input 7
(9) Input 8
(10) Common B

10 Position Terminal Block Pin Number	Signal Name
1	Input 1
2	Input 2
3	Input 3
4	Input 4
5	Common A
6	Input 5
7	Input 6
8	Input 7
9	Input 8
10	Common B

Sourcing Input Wiring

Input 1
Input 2
Input 3
Input 4
Common A
Input 5
Input 6
Input 7
Input 8
Common B

Sinking Input Wiring

Input 1
Input 2
Input 3
Input 4
Common A
Input 5
Input 6
Input 7
Input 8
Common B

NOTE: *In order to maintain UL508 rating, discrete inputs must be powered from a Class 2 power supply.*

3

ViewMarq Mounting

Two sets of mounting bracket assemblies are supplied with each display. There is one pair of wall mounting brackets and one pair of chain mount brackets.

These are the general guidelines for mounting the display:

- Only qualified personnel should mount the ViewMarq LED display.
- Verify correct operation of the display on a test bench before mounting. After testing, disconnect power and communications until after the display is mounted.
- The ViewMarq LED Display is rated for indoor use only and should not be mounted outdoors.
- Protect the lens from scratches while installing the display.
- Do not remove the end caps. This will invalidate the NEMA ratings and void the warranty.
- Do not drill or cut holes in any part of the display. This will invalidate the NEMA ratings and void the warranty.

Installation Notes

- In order to maintain UL508 rating, flexible conduit must be used in place of included cable glands.
- The UL508 rating does not apply for a chain mounted display.
- In UL508 installations the allowable ambient air temperature range is 0 - 60 °C (32 - 140 °F)
- In order to maintain UL508 rating, discrete inputs must be powered from a Class 2 power supply.

Use the appropriate hardware and fasteners to hang or suspend the display. All hardware, fasteners and mounting methods must be rated for a minumum of **Four** times the weight of the display.

ViewMarq LED Display Weights			
Part Numbers	**Weight**	**Weight with Wall Mount Brackets**	**Weight with Chain Mount Brackets**
MD4-0112T	4.9 lbs (2.2 kg)	6.1 lbs (2.8 kg)	5.4 lbs (2.5 kg)
MD4-0124T	9.0 lbs (4.1 kg)	10.2 lbs (4.6 kg)	9.6 lbs (4.4 kg)
MD4-0212T	7.3 lbs (3.3 kg)	8.5 lbs (3.9 kg)	7.8 lbs (3.5 kg)
MD4-0224T	13.1 lbs (6.0 kg)	14.3 lbs (6.5 kg)	14.7 lbs (6.7 kg)
MD4-0412T	12.1 lbs (5.5 kg)	9.5 lbs (4.3 kg)	9.7 lbs (4.4 kg)
MD4-0424T	22.5 lbs (10.2 kg)	18.5 lbs (8.4 kg)	18.7 lbs (8.5 kg)

Mounting Position

The ViewMarq LED display is intended to be mounted horizontally. It may be mounted in either horizontal direction. When the unit is flipped, the display text will automatically adjust to the correct orientation. To disable this feature, see Chapter 6 - Configuring the ViewMarq LED display.

Mounting Clearance

Mount the ViewMarq LED display so as not to exceed the minimum bending radius of the flexible conduit and / or cable glandss and cables. Mounting the display with the included wall mounting brackets should allow for adequate clearance. Be sure to provide ample clearance when using chain mounts.

Viewing Angle and Distances

The ViewMarq LED display should be mounted based on the expected viewing angle and viewing distance. The horizontal and vertical viewing angle of all signs is 140°.
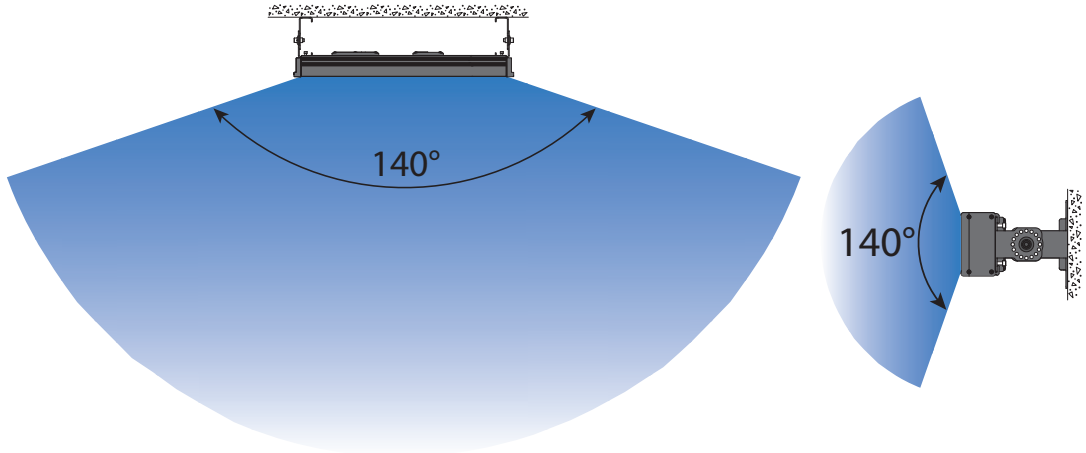
The table below shows the recommended viewing distances based on the character size that will be used.

Part Numbers	Font Size	Minimum Recommended Viewing Distance	Maximum Recommended Viewing Distance
MD4-0112T MD4-0124T MD4-0212T MD4-0224T MD4-0412T MD4-0424T	1-1/4 in	6 ft (1.8m)	60 ft (18m)
MD4-0112T MD4-0124T MD4-0212T MD4-0224T MD4-0412T MD4-0424T	2 in	10 ft (3.0m)	100 ft (31m)
MD4-0212T MD4-0224T MD4-0412T MD4-0424T	4 in	20 ft (6.1m)	200 ft (61m)
MD4-0412T MD4-0424T	6 in	20 ft (6.1m)	300 ft (91m)
MD4-0412T MD4-0424T	8 in	20 ft (6.1m)	400 ft (122m)
MD4-0412T MD4-0424T	10 in	20 ft (6.1m)	500 ft (152m)

Mounting Brackets

Wall mount and chain mount brackets are shipped with each display.
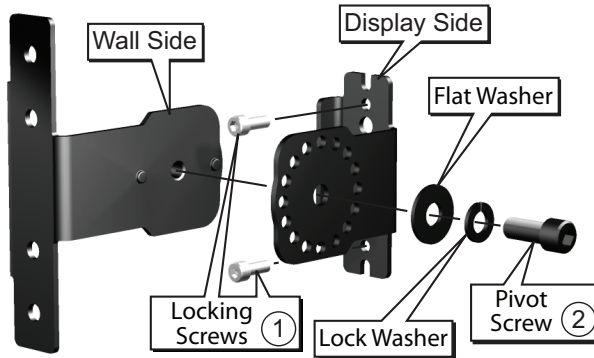
Wall Mount Bracket - 2 each
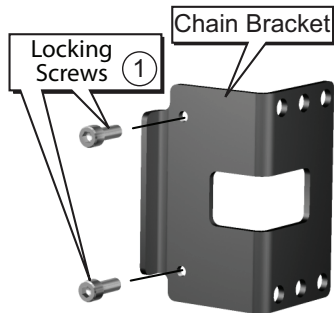
Chain Mount Bracket - 2 each

① Locking Screws - (#10-32 x 1/2" SHCS) for 4mm Hex Key.
② Locking Screws - (5/16-18 x 5/8" SHCS) for 6mm Hex Key.

Wall Mounting

This is an example of a wall mounted installation.
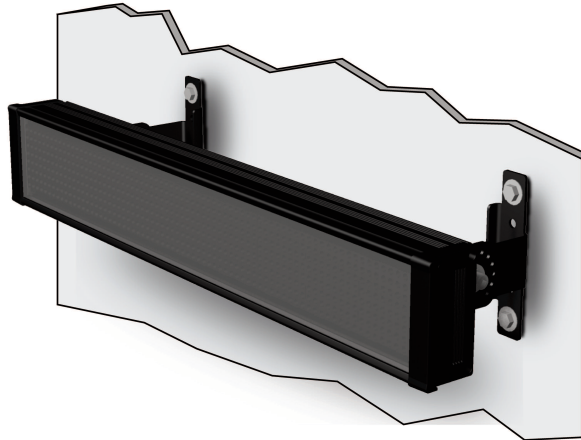
1) Assemble both adjustable wall brackets as shown above. Slide a bracket into the display mounting rail that runs the length of the back of the display. Repeat for the opposite end. Place each bracket in the desired position and finger-tighten the two locking screws in each bracket to prevent them from sliding.
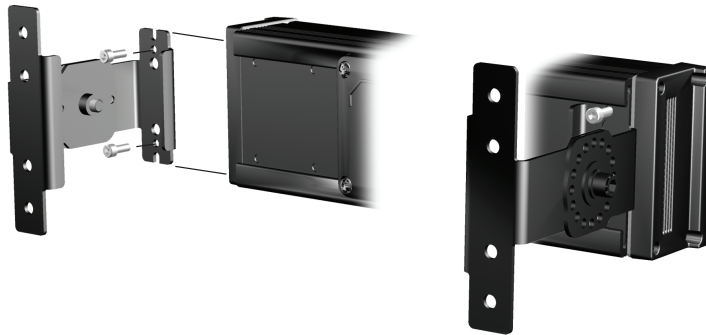
2) Hold the ViewMarq display in the desired mounting position and mark where the mounting bracket holes are positioned on the mounting surface.

3) Disassemble the mounting brackets, leaving the "display side" of the brackets in the display mounting rail.

4) Attach the "wall side" of the brackets to the mounting surface using appropriate fasteners for the type of mounting surface. Make sure the brackets are level with each other. (Wall mounting hardware is not included.)

5) Install the power and communication cabling through the appropriate cable glands and covers. Connect power and communication cables to their ports on the message display.

6) Install the covers and make sure that there is enough cable inside the housing to provide a "service loop" so as not to exceed the minimum bend radius of the cable

Service Loop
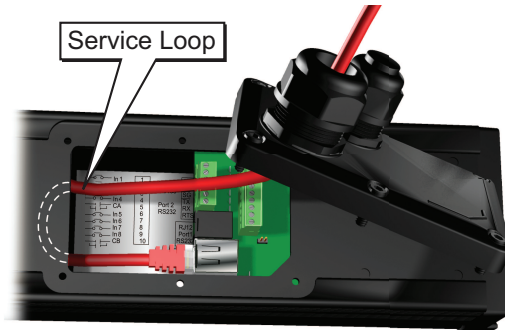
7) Install covers being careful to observe the orientation keying. Tighten the captive cover screws to 10 in·lbs (1.1 N·m). Properly tighten the cable glands to maintain the NEMA rating on the enclosure.

8) Loosen the "display side" brackets to allow movement in the display mounting rail. Align one "display side" bracket on the display with the corresponding "wall side" bracket on the wall and finger tighten the pivot screw with the flat washer and lock washer.

9) While holding the display in position slide the other "display side" bracket in the mounting rail on the display to line up with the other "wall side" bracket and insert the 6mm pivot screw with the flat washer and lock washer as shown.

10) Set the desired viewing angle and tighten the pivot screws to 6 in·lbs (0.7 N·m) of torque. Tighten the locking screws to 9 in·lbs (1.0 N·m) of torque.
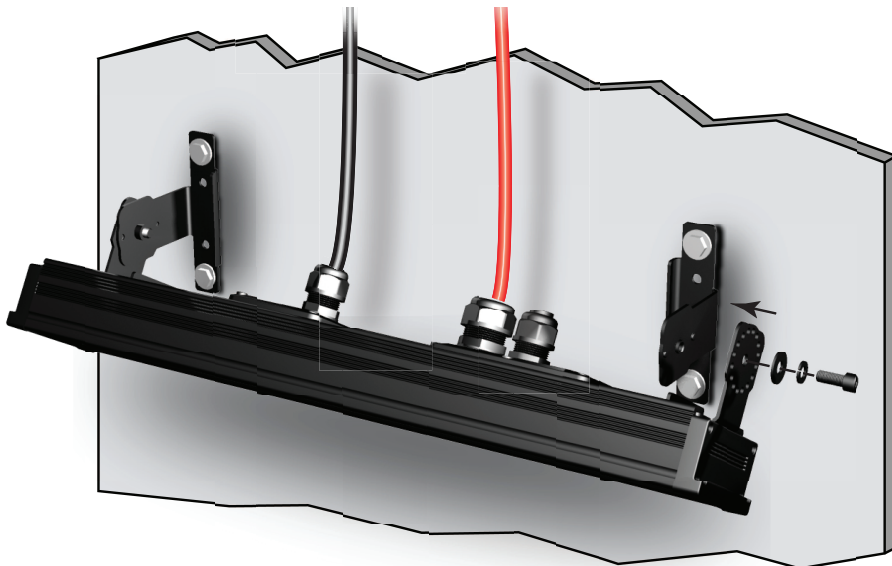
Chain Mounting

This is an example of a chain mounted installation.
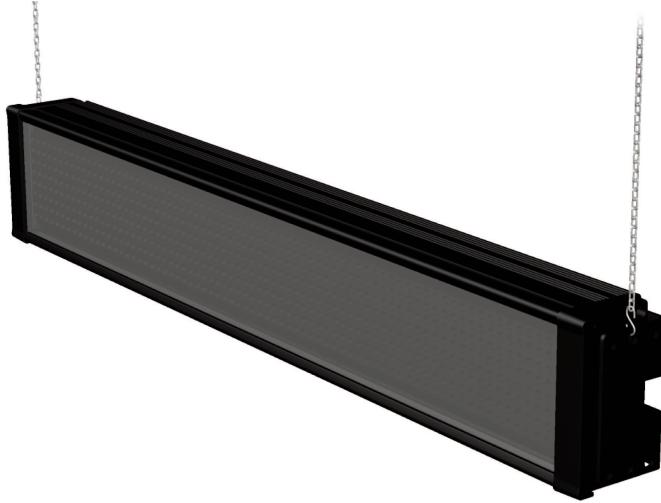
1) Slide the chain mount brackets into the display mounting rail on each end of the display. Tighten the locking screws to 9 in-lbs (1.0 N-m).

2) Attach a chain to an appropriate mounting surface or structure at the width listed in the table below. Connect the chain ends to one of the chain mounting holes on each bracket to achieve the desired viewing angle.

3) Install the appropriate cabling per steps 5 – 7 in the wall mounting procedure on the previous pages.

Continued on Next Page.

3
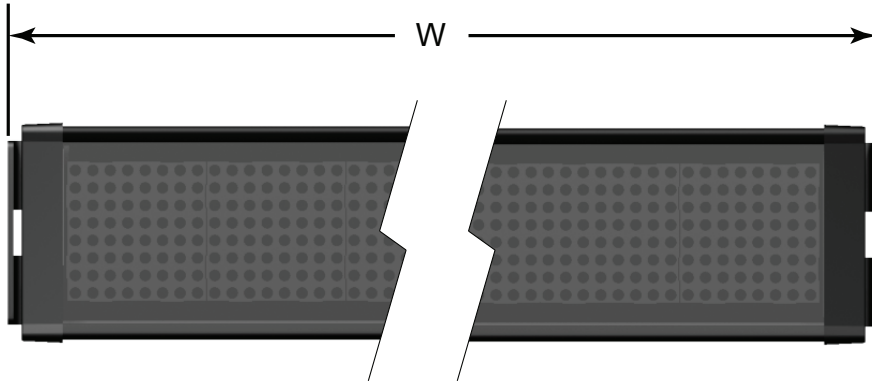
ViewMarq Chain Mount Width	
Part Number	**Overall Width with Chain Mount Brackets (W)**
MD4-0112T	23.60 in [599.4 mm]
MD4-0124T	45.21 in [1148.3 mm]
MD4-0212T	23.60 in [599.4 mm]
MD4-0224T	45.21 in [1148.3 mm]
MD4-0412T	23.60 in [599.4 mm]
MD4-0424T	45.21 in [1148.3 mm]

PC TO VIEWMARQ COMMUNICATION

CHAPTER

4

In This Chapter...

Introduction

This chapter explains how to communicate with the ViewMarq display using a Windows based personal computer.   It will cover:

- Links and how they are used with the ViewMarq
- Creating Ethernet Links
- Browsing the Ethernet network for ViewMarq displays
- Editing ViewMarq display Ethernet settings online
- Creating serial Links

The user will need to communicate with the ViewMarq for various reasons:

1) Read the existing LED display configuration

2) Configure the ViewMarq display communication ports

3) Send messages to the ViewMarq display

4) Reset the display

5) Reset factory defaults in the display

6) Update firmware in the display

7) Test the LED's

Links

The ViewMarq Configuration software utilizes Links to communicate to ViewMarq displays. A Link is a set of PC communication parameters that allows the software to communicate to a display or a set of displays when necessary without having to stay "connected" to the display, tying up a port on the PC. Links are created and configured in the PC to LED display Communication dialog and once created may be easily selected from a Link drop down list at the top right of the software window as shown.
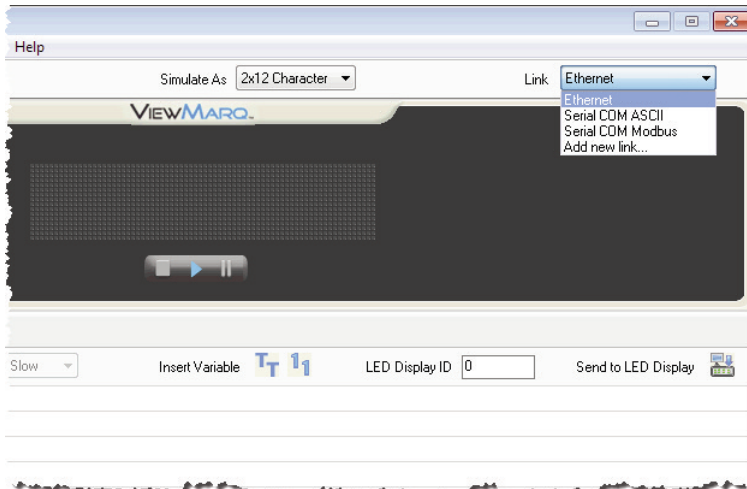
There are three types of Links:

1) Ethernet Modbus/TCP

2) Serial ASCII

3) Serial Modbus

Ethernet Modbus/TCP

Ethernet links are the easiest to setup and are Modbus/TCP protocol only. They are point to point, in other words, only one display can be addressed at one time via its IP address.

1) Using a straight-through or crossover Ethernet Patch cable, connect the ViewMarq to the PC's Ethernet port. The system will automatically detect the cable type.(One seven foot straight-through cable is supplied with the ViewMarq).

2) The ViewMarq LED display can also be connected to a local Ethernet network via a straight-through or crossover cable.
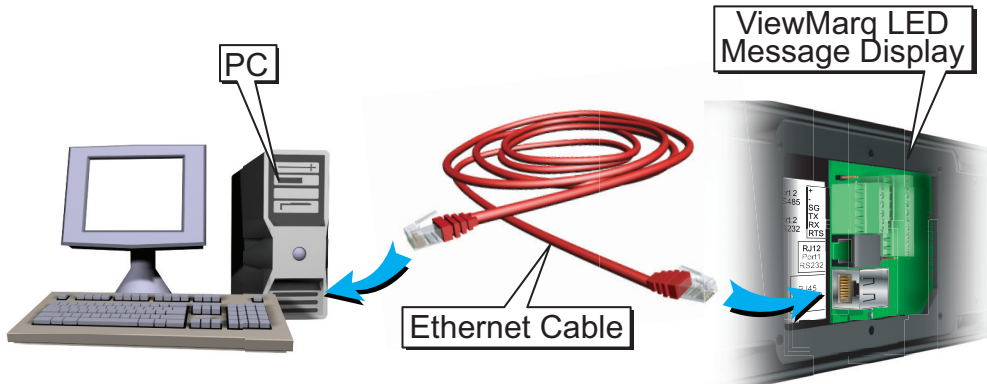
Serial ASCII

Serial ASCII Links are the most versatile. They can be set up over RS-232 or RS-485. Serial ASCII does not address a particular display by node number like Modbus or an IP address. This allows the ID in the ASCII string to be utilized to address multiple displays. See Chapter 5 for more details on LED display ID.

Serial Modbus

Serial Modbus links are point to point. Each transaction is directed to a particular panel via a Slave Node Number which is unique for each display on a network.
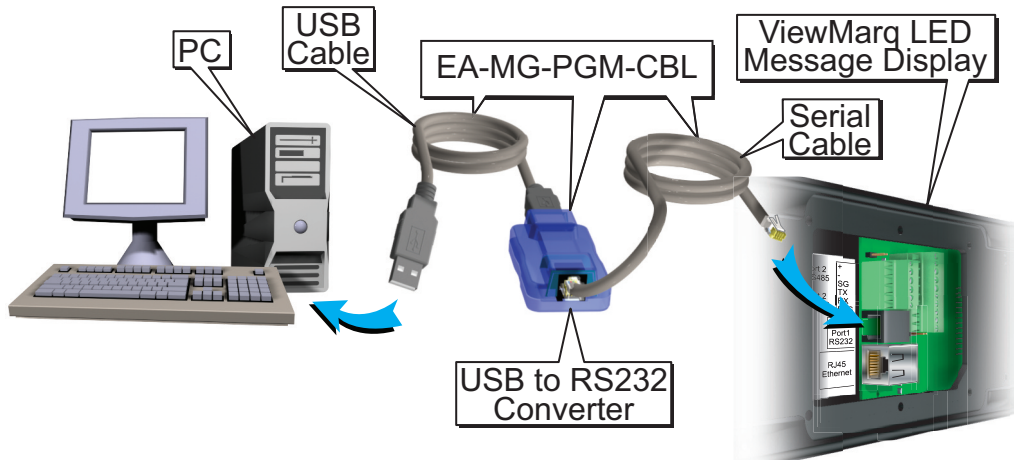
Adding a New Link

To add a new link, select PC to LED Display Communications from the Setup dropdown menu...
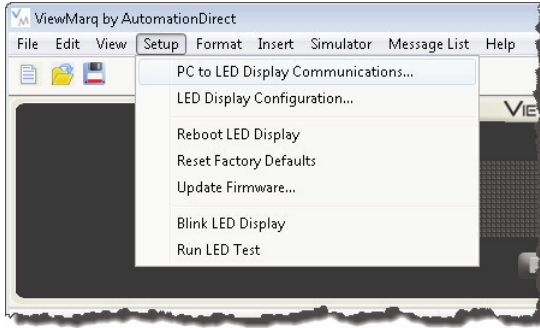
...or select the PC to LED Display Communications toolbar button.

This will open the PC to LED Display Communications dialog.
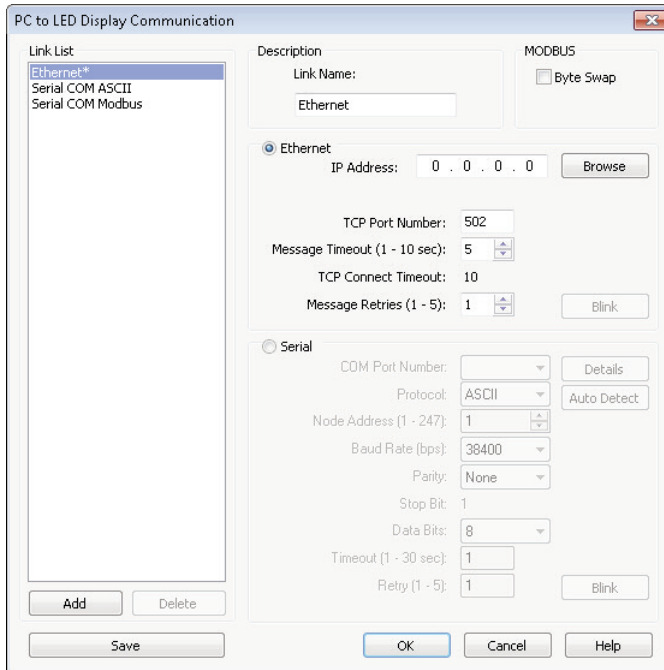
Ethernet Link

Follow these steps to add an Ethernet Link.

1) With the **PC to LED Display Communication** dialog open, click the **Add** button.
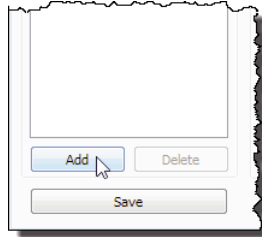
Notice that a new link named "New Link" has been added. The * indicates that the new link has not been saved yet.
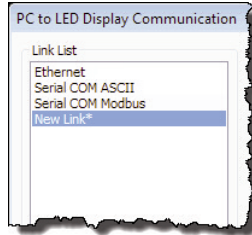
2) Select the **Ethernet** radio button.
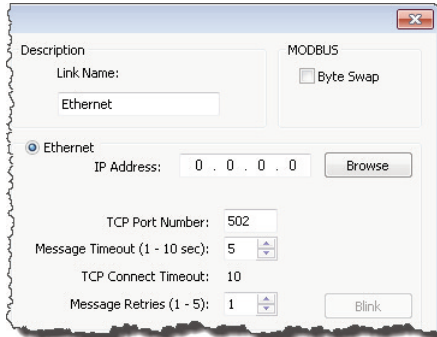
3) If you know the IP address of the display, enter it here and skip to step 15.

4) Otherwise click the **Browse** button to browse the network for any connected ViewMarq displays.

5) The Ethernet Browse dialog will open and the connected ViewMarq display will be listed as shown below.

    A. If the ViewMarq does not show up on the list:

      i. Check your physical connection.

      ii. Check that the Ethernet Port on the PC is enabled.
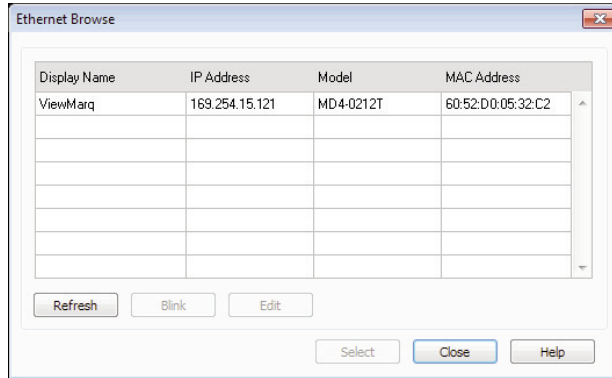
6) Select the ViewMarq LED message display in the table.

7) If the IP Address is correct on your local Subnet, click **Select** and **proceed to step 15**.

8) If the IP Address is not on your Subnet it must be changed, select it and click **Edit** to open the **Ethernet Browse Editor**.
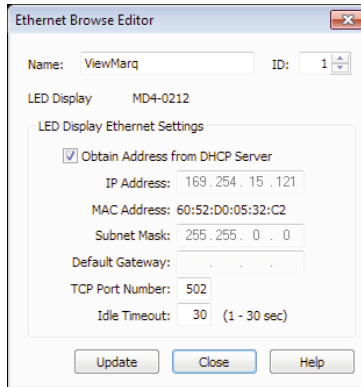
9) Deselect **Obtain IP Address from DHCP Server**

10) Enter an IP Address, Subnet Mask and Default Gateway compatible with your PC's Ethernet network settings or check with your IT department for the proper settings.

NOTE: *For more information on configuring ViewMarq Communication ports see Chapter 6 - Configuring the ViewMarq LED Display.*

NOTE: *If no DHCP server is found, a default IP Address of 169.254.XXX.nnn will be set. The notation .nnn will vary for each display on the network.*

11) You may also choose to change the name of the ViewMarq display and ID number at this time.

12) Click **Update.**

The display will automatically reset. The new settings will scroll on the display.

13) In the Browse Window click the **Refresh** button if the marquee is not listed.

14) When the ViewMarq is listed in the table again, select it and click **Select**.

15) The TCP Port Number default value is 502 which is the Ethernet port for the Modbus TCP protocol. This should not be changed unless it has been changed on the ViewMarq also.

NOTE: *To communicate with the ViewMarq message display through a router, port 502 must be open.*

---

16) The Message Timeout and Message Retry values are adjusted to make network communications more tolerant of network delays and noise. The TCP connection timeout is always twice the message timeout and not configurable.

17) With the new link highlighted in the Link List, enter a new name for the link in the Link Name box.

18) Click **Save** and **OK**

Serial Link

Follow these steps to add a Serial Link.

1) With the **PC to LED Display Communication** dialog open, click the **Add** button.
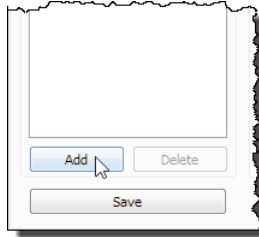
Notice that a new link named "New Link" has been added. The * indicates that the new link has not been saved yet.
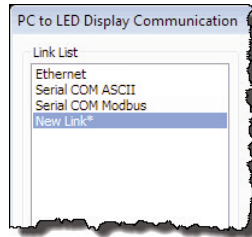
2) Select the **Serial** radio button.
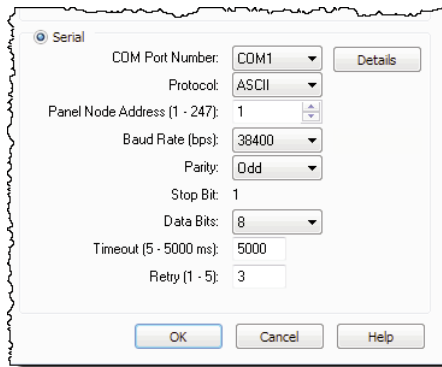
3) If you know the COM Port Number on the PC that the View Marq is connected to, then select the COM Port number dropdown. Only the available active PC COM ports will be shown in the list.

NOTE: *If using the EA-MG-PGM-CBL USB to RS-232 converter, and you aren't sure which PC COM Port is assigned, click the **Detail…** button to identify it. The screen below shows the Koyo USB-Serial Comm Port devise assigned to COM4. Select it and click **OK**.*

4

Serial Port Details

COM1   Communications Port
COM4   Koyo USB-Serial Comm Port

Select   Cancel
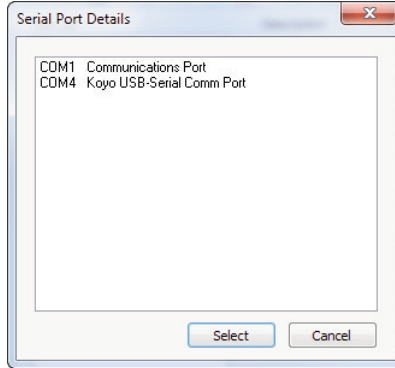
4) If you know the Serial Communication settings of the Serial port on the ViewMarq display, change Protocol, Panel Node Address (Modbus only), Baud Rate, Parity and Data Bits to match the settings of the ViewMarq.

NOTE: *For more information on configuring the ViewMarq Communication Ports see Chapter 6 – Configurating the Viewmarq LED Display*

5) Default Timeout and Retry settings are usually best depending on your connection quality. You may adjust these as necessary if errors occur.
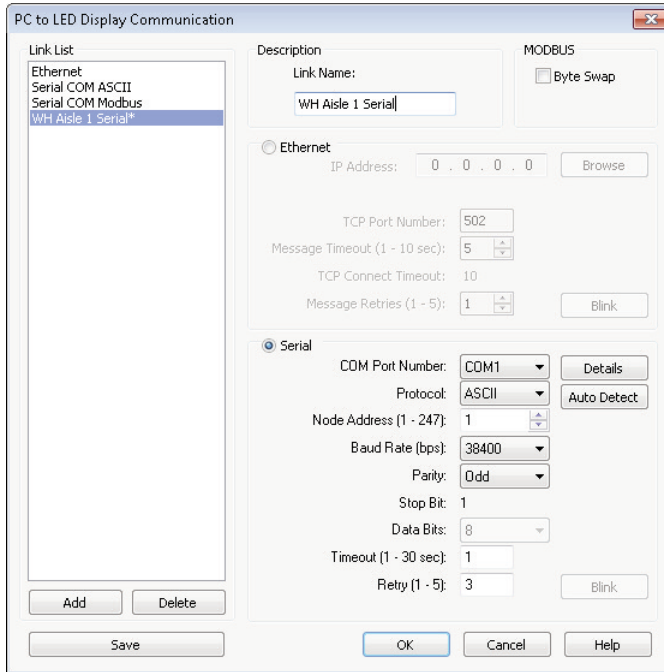
6) With the New Link highlighted in the Link List, enter a new name for the link in the Link Name box.

7) Select **Save** and **OK**.

CHAPTER 5

CREATING AND PREVIEWING MESSAGES

In This Chapter...

Creating and Previewing Messages

The Text Editor is designed to make message creation easy by allowing users to simply start typing. As a message is entered, the software converts the formatted message into an ASCII string that is displayed in the Command String window located at the bottom of the workspace. This ASCII String can then be copied and used in control devices such as a PLC that can be connected to the ViewMarq display. These devices will send the command string to the ViewMarq display. This ASCII string contains the text and format that is to be presented. This software basically reduces the need for users to learn or type the ASCII command strings needed to control messaging.

One key item you will notice when you first begin to type is that the text being entered will immediately appear in the Simulator window located directly above the Text Editor window. This WYSIWYG (an acronym for "what you see is what you get") simulator will display the text being edited precisely as it will appear on the ViewMarq display. The Simulator reduces the need to send the message to the display for testing purposes. You don't need hardware connected to see what your message will look like. You will quickly find that the Simulator greatly reduces the time needed to create and test messages.

As you enter your message, try using the format tools located on the toolbars above and below the Text Editor. Try changing the font color, blink rate, scroll settings, and position and see how it appears in the Simulator. You can then continue reading on to get a more detailed understanding of how to use the features in this chapter.

The Software Workspace
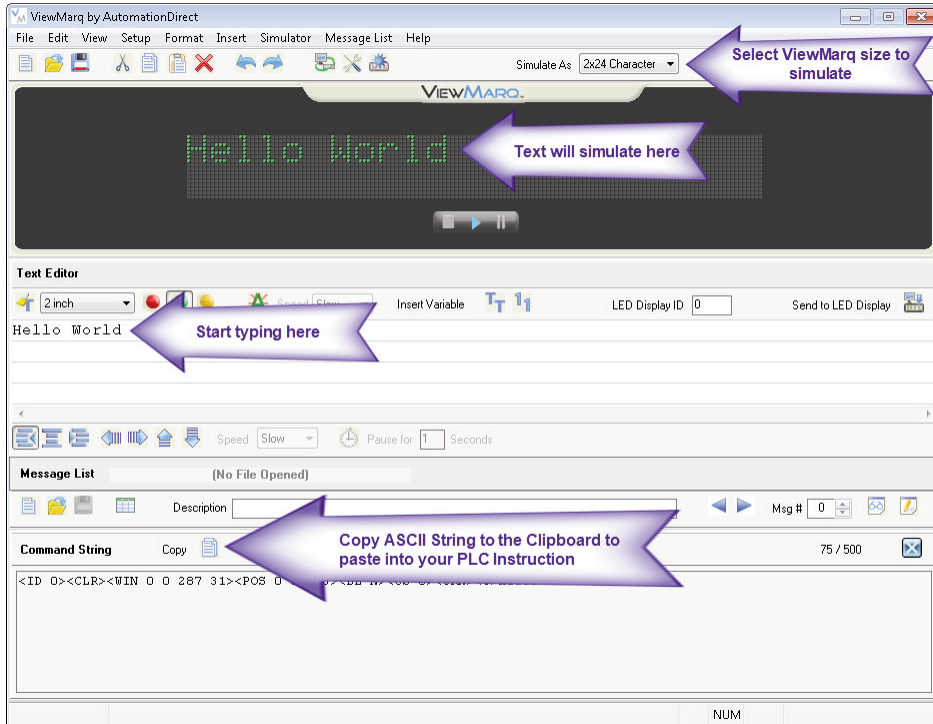
Select the ViewMarq Size

Before creating a message, you should determine which size of ViewMarq display you wish to simulate. The display size will affect your decisions on the character set and text position that you may want to use. For example, if you have a single line display then the larger size characters will not be legible.
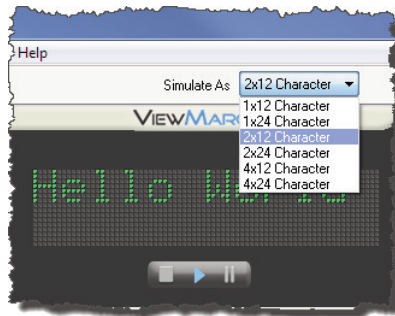
NOTE: *Selecting the Simulator size does not affect the ASCII string, only the way it will look in the Simulator.*

Saving the Message

Once the message is created and it simulates as intended, it may be saved for use or editing later. To save the message, open the File drop down menu and click Save, or click on the Save toolbar button.
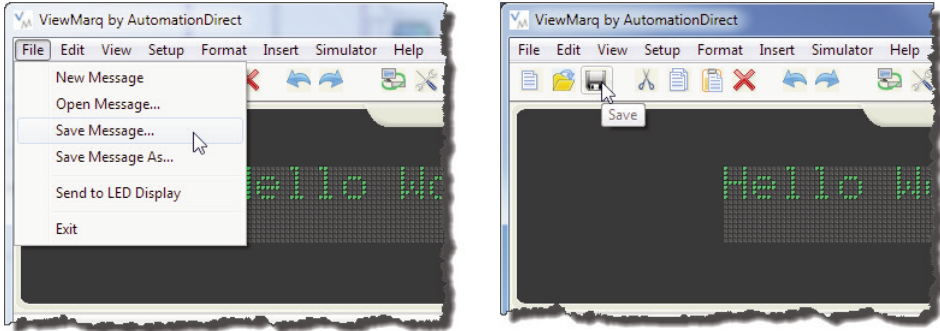
If the message has not been saved already, the Save As dialog will open.
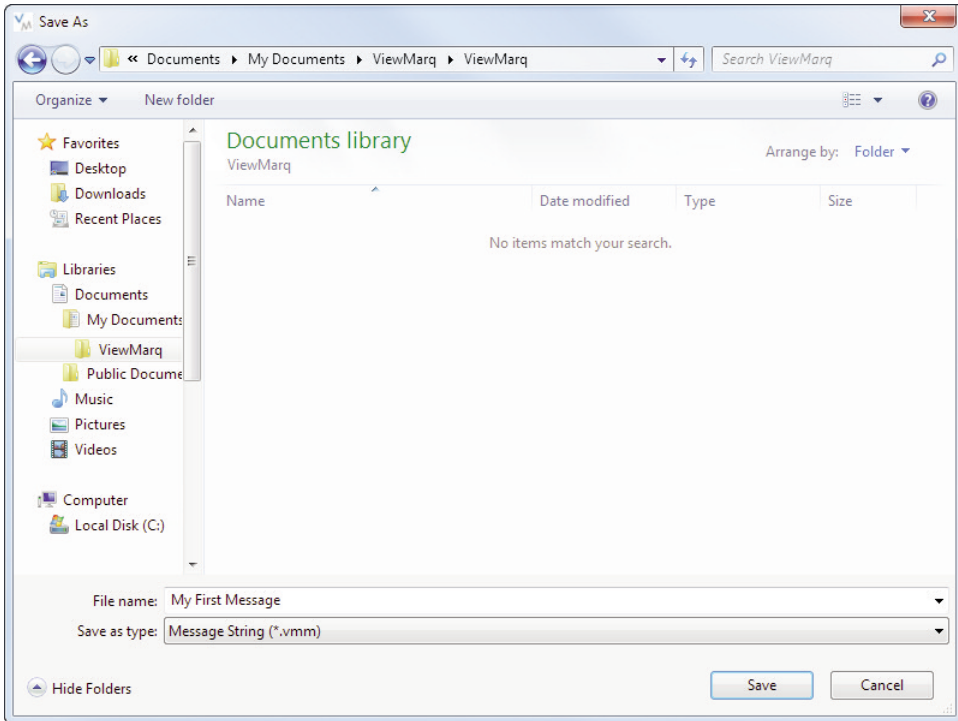
Enter a unique name for the message and click Save.

Open a Saved Message

Messages that have been previously saved may be opened, edited and saved again. To open a saved message open the File drop down menu and click Open, or click on the Open toolbar button.
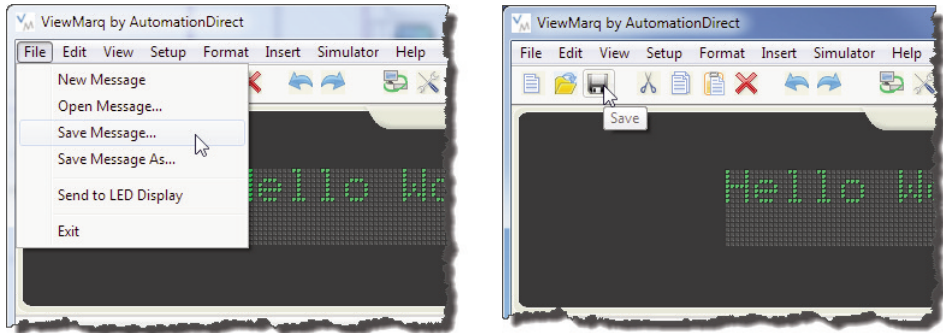
The Open dialog will open.
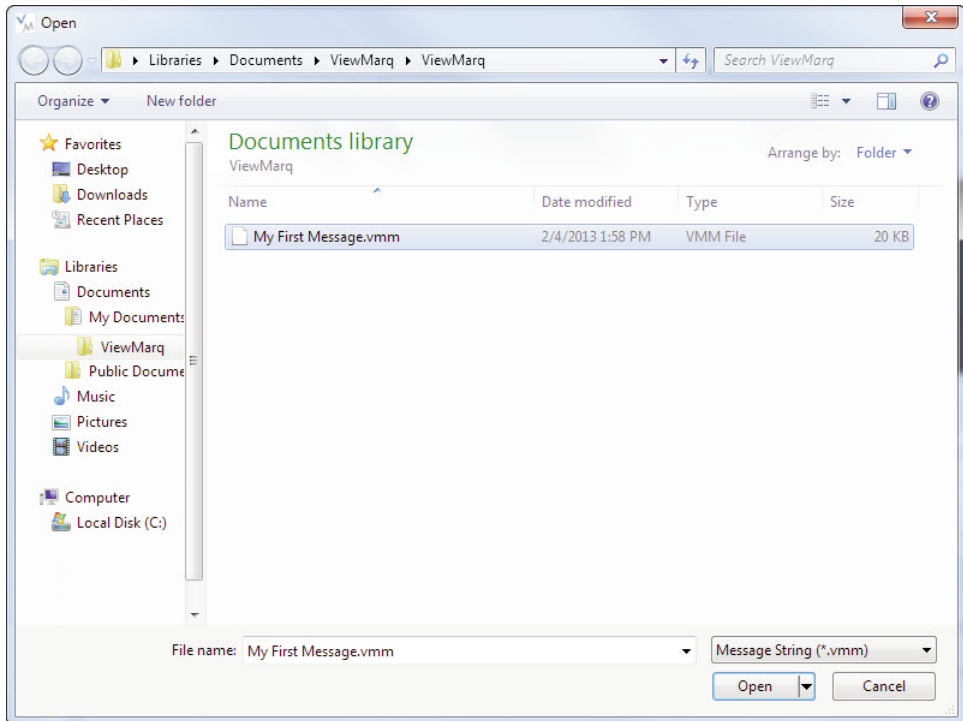
Select the desired message file and click Open. The selected message will open, ready for editing or sending to a ViewMarq display.

Text Editor Overview

In the Text Editor you simply select a line and start typing. The Simulator displays the text and helps you to understand how it will look on the ViewMarq display. There are a few basic guidelines to learn that will help you when creating a message.

Software Purpose

The primary purposes of the software are:

- Create and Preview Messages
- Display messages on the ViewMarq
- Configure the ViewMarq LED display
- Create ASCII String Commands that may be used in a PLC to control the LED message display

Supported Characters

The Text Editor supports 94 of the 95 ASCII characters: standard keyboard characters. The exception is the back apostrophe (`). It is reserved to indicate the degree symbol (°).

Message Length

A maximum of 299 text characters can be typed into each line of the Text Editor. The entire ASCII string may only be 500 charcters long including the commands and formatting characters.

The character size and the screen width of the hardware will determine the number of characters viewable. The Text Editor does not truncate the text that is not displayed on the screen. Text is entered from left to right when left justified and will continue off the screen. See the example below:

Example of a message with the 1x12 character ViewMarq display with the 2-inch character set.

Example of a message with the 1x24 character ViewMarq display with the 2-inch character set.

Message Formatting

Message formatting is accomplished on three different levels:

1) Character
2) Line
3) Message

Character Toolbar

The character toolbar contains tools used to format individual or multiple characters much like your common word processors. The character toolbar has three functions:

1. Character Set

You may choose from 11 available character sets. Individual characters can be mixed to get the desired effect as shown below. Select the character set and start typing, or highlight existing text and select the character set and the text will change.

2 inch
2 inch
2 inch Full Line
Compact
4 inch Narrow
4 inch
4 inch Full Line
6 inch Narrow
6 inch
8 inch Narrow
8 inch
8 inch Full Line
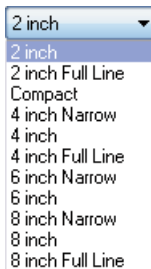
2. Character Color

Three character colors are available; Red, Green and Amber. Like the character sets, colors may be mixed to achieve a desired effect. Select the character color and start typing, or highlight existing text and select the character color and the text will change.

3. Character Blink and Blink Speed

Individual or groups of characters may also be set to blink. There are three blink rates choose from. Like the other character settings, blink speeds may set differently for individual characters to achieve a desired effect.
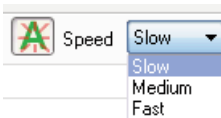
Line Toolbar

The line toolbar contains tools used to add justification or scrolling to each line.

1. Line Justification

Line justify determines which starting position on the line the text will appear, Left, Center, or Right. This setting affects the entire line. Justify defaults to the left side. If scrolling is enabled, it will override any justification setting.

2. Line Scroll

Line scroll is a line based setting. It is used to animate messages to grab the viewer's attention.
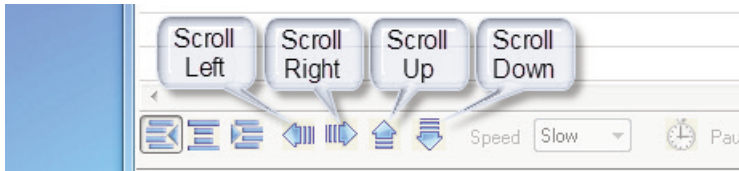
3. Line Scroll Pause

When enabled on an individual line, the line will scroll across the viewable area of the display. The ViewMarq supports scroll left, scroll right, scroll up and scroll down. Click any Line Justification button to stop the scroll and set the static position of the text.
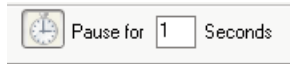
4. Scroll Speed

The scroll speed is a message based setting and determines the rate at which the lines in a message will scroll; slow, medium, or fast. The default speed is slow. The speed selection tool is disabled until the scroll has been enabled.
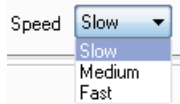
NOTE: *There can be only one scroll speed per message. All scrolling lines will scroll at the same rate no matter what direction they scroll. If the speed is changed for one line, all line scroll speeds will be changed.*

Format Options	
Option	*Effects*
Character Size	Character Based
Character Color	
Blink	
Blink Rate	
Justify Left	Line Based
Justify Center	
Justify Right	
Scroll Left	
Scroll Right	
Scroll Up	
Scroll Down	
Scroll Speed	Message Based

Scrolling Multiple Lines Together

Sometimes it is desirable to have multiple lines scroll together. If adjacent lines are selected to scroll in the same direction, they will scroll in a synchronized manner. For example, if text is entered on all four lines and Scroll Up is selected for all four lines, then all four lines will scroll up together across the entire viewable area of the screen. If line 3 is selected to be Justified Left, then the text on Lines 1 and 2 will Scroll together up the area of Lines 1 and 2 while the text on Line 4 will scroll up the area of line 4.

Other Tools

There are additional tools available in the ViewMarq software to aid in formatting and editing text. There are also some ASCII string editor tools.

New Message and Clear Text Editor

The New Message and Clear Text Editor Toolbar buttons both will clear the Text Editor and reset all formatting options back to their default settings.

New Message

Clear Text Editor

Variables

ViewMarq has 16 String and 32 Numeric Variables available that may be updated using ASCII string commands or by writing directly to slave Modbus registers. See Chapter 8 for more details on inserting and updating variables.
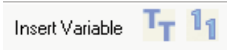
Insert Variable

NOTE: *The display format options such as color or character set for variables is set in the static text string where the <DEC> and <STR> commands are used; that is, color or character set commands do not work with, <SETS> or <SETV> commands.*

LED Display ID

LED Display ID adds the <ID#> command to the ASCII string. This identifies the target LED display.

Insert Variable T_T 1₁   LED Display ID 1

LED Display ID			
Type	**Format**	**Example**	**Description**
Broadcast (Default)	<ID 0>	<ID 0>	All ViewMarq displays on the network will process the commands in the ASCII string.
Single	<ID n>	<ID 1>	The ViewMarq display on the network with the designated ID will process the ASCII commands.
Multiple	<ID a,c,e,g>	<ID 1,3,5,7>	The ViewMarq displays on a serial RS-485 network with the designated ID's will process the ASCII commands.
Range	<ID a - g>	<ID 1-8>	The ViewMarq displays on a serial RS-485 network with ID's within the designated range will process the ASCII commands.
Combination	<ID a,c,e-h>	<ID 1,3,5-8>	Both Multiple and Range formats may be combined.

NOTE: *The factory default ID number for all ViewMarq LED displays is 1. It is recommended that multiple panels on the same network each have different ID numbers.*

Attention!: When communicating by the Modbus protocol, this number must match the Modbus Node Address of the display.

Message Simulator

One of the most helpful features in the ViewMarq software is the message Simulator. It helps give the user an accurate representation of how the message will look on the ViewMarq LED display.
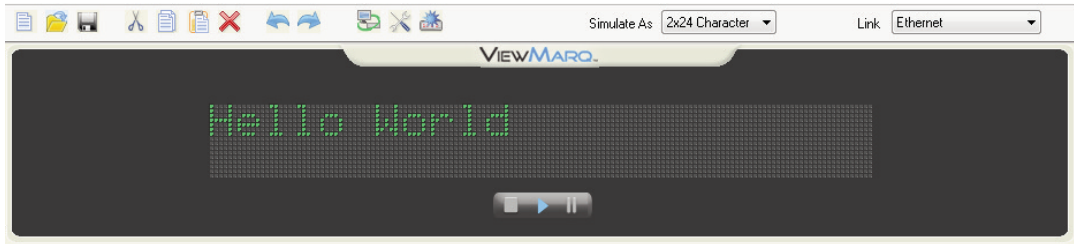
Here are some key points about the Simulator:

- By default, the Simulator is active when the software starts.
- The user should select the correct ViewMarq display hardware on which the messages will be displayed; otherwise there will not be a true representation of the message on the Simulator.
- The Simulator can be started, stopped, or paused with the control toolbar located in the Simulator window.
- The Simulator is designed to accurately represent how the message will operate on the ViewMarq hardware; however the performance of some PC's running the software may affect the Simulator speed.

1) Select the proper LED display size.
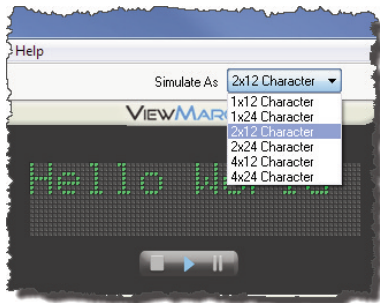
2) Start typing text.

3) If you wish to pause scrolling or blinking text, select the Pause button [image]. To continue scrolling or blinking, deselect Pause.

4) When Stop [image] is selected Play [image] must be selected to start again. Any scrolling will start from the beginning.

Command String Window

The Command String Window is where the ViewMarq software builds the ASCII command string that will be used by another device such as a PLC. This window may be closed by selecting the ⊠ icon. To reopen the window, select the **View** menu then **Command String Window**.
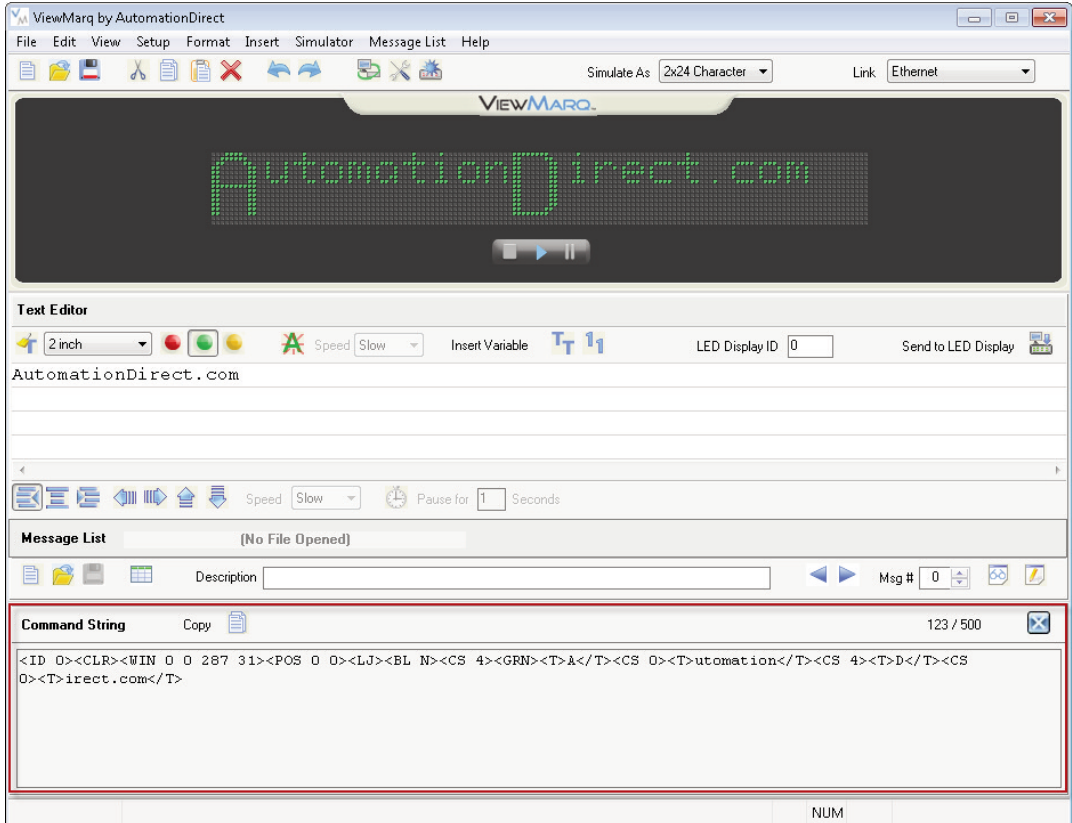
For more information on the ASCII command string syntax, see Appendix A – ViewMarq ASCII Command Specification.

There are a couple of important tools located in the Command String Window.
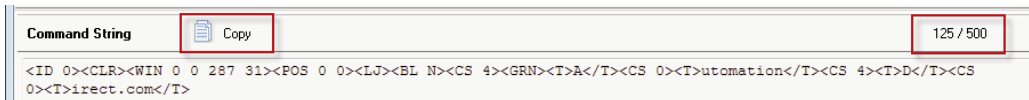
Copy

Copy puts the ASCII Command String onto the PC Clipboard so that it can be pasted into your PLC program instruction or text file for archive purposes.

Character Count

The maximum number of characters in an ASCII Command String that the ViewMarq can accept is 500. This includes the text and all formatting characters and commands. If the string becomes more than 500 characters, the Character Count will turn RED and a warning message will be displayed.

Command String   📄 Copy   501 / 500   ✖

ViewMarq for Windows

❌ The command string contains more characters than the ViewMarq will accept.
500 characters are allowed - shorten the string by 1 character(s) before sending it to the LED display.

OK
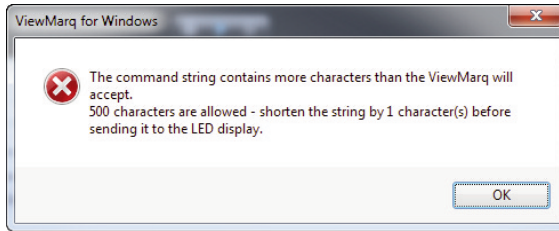
NOTE: The software will not prevent you from creating a message that is too long. It is up to the user to remove the extra characters to shorten the string to 500 characters or less.

CHAPTER 6

CONFIGURING THE VIEWMARQ LED DISPLAY

In This Chapter...

Introduction

The ViewMarq LED message display is fully configurable from the ViewMarq Software. It is not necessary to set any dip switches or jumpers on the display to configure it. Through the ViewMarq Software the following can be configured:

1) Display Name and ID

2) Global Communication Settings and Options

3) Ethernet Communications

4) RS-232 Communications (Port 1)

5) RS-485 Communications (Port 2)

The configuration settings may be saved to a file for storage and may be opened later to load to a message display.

6

NOTE: *This configures the ports on the ViewMarq display hardware as opposed to "PC to LED Communications" links which configure your PC to communicate to the Viewmarq display.*

Message Display Selection and Configure Message Display

1) Select the Link to the message display to be configured. See Chapter 4 - PC to ViewMarq Communication. The Link may be an Ethernet Modbus TCP, Serial ASCII or Serial Modbus.
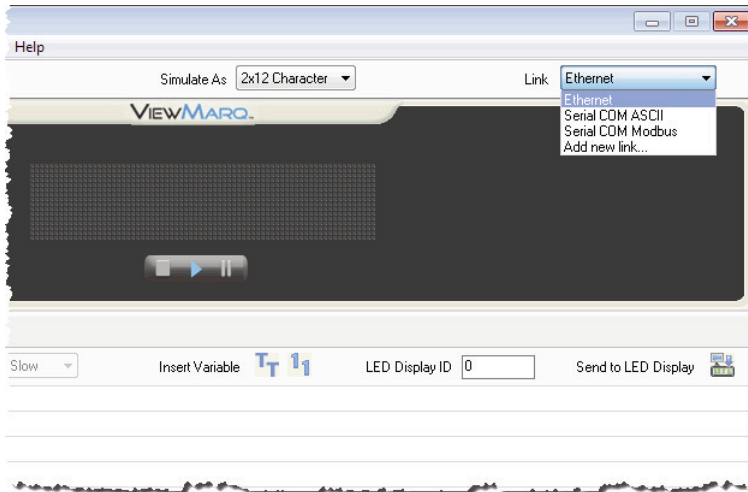
ATTENTION

Attention!: Message display configuration can only be accomplished point-to-point, PC to one ViewMarq.

2) To begin the message display configuration, select the LED Display Configuration button...

...or choose **LED Display Configuration** from the **Select** menu as shown.
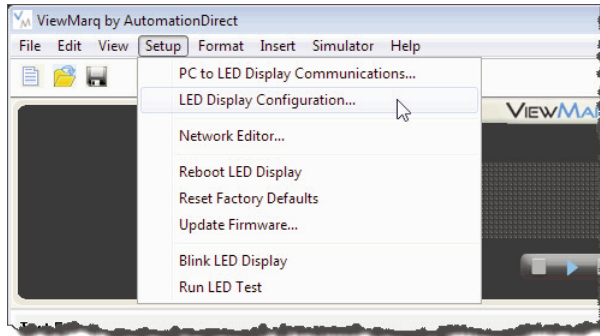
> **NOTE:** *When the LED Display Configuration Dialog opens, the software will attempt to read the settings from the display associated with the selected Link. If there is no connection, an error message will be displayed. A Configuration file may still be created. Simply select **OK** and continue.*

If the software detects a message display, it will read the configuration and display it in the LED Display Configuration dialog as shown.

> **NOTE:** *If the ViewMarq LED display panel has the Startup Message displayed, the display may go blank while the LED Display Configuration is aquiring information from the panel.*
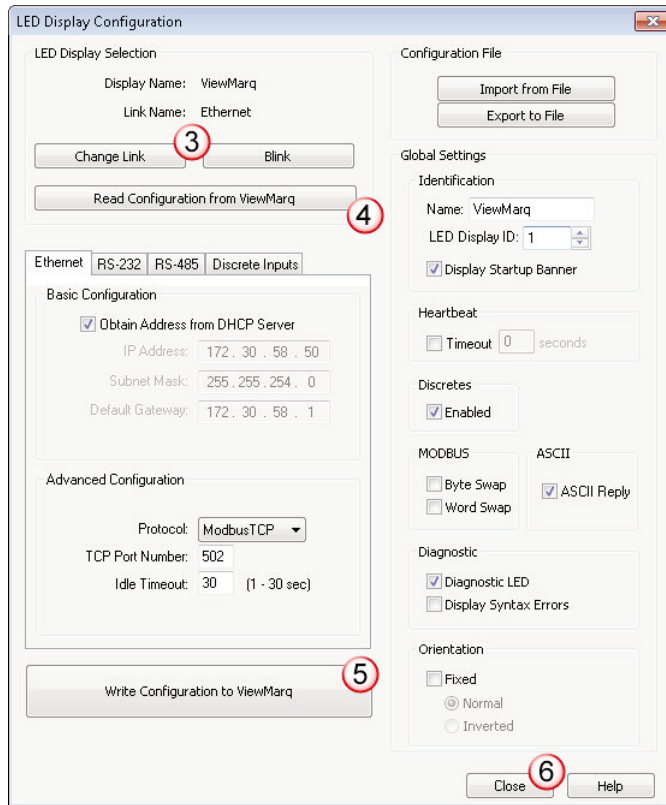
3) Select the **Blink** button to make sure you are connected to the message display. If not, click the **Change Link** button and select the proper link for the message display you are configuring.

4) Select **Read Configuration from ViewMarq** to read the current configuration from the message display.

NOTE: If the ViewMarq LED display panel has the Startup Message displayed, the display may go blank while the LED Display Configuration is aquiring information from the panel.

5) Adjust the settings as needed and click the **Write Configuration to ViewMarq** button.

6) Select **Close**.

Message Display Settings

Global Settings

Most of the settings in the configuration belong to a particular communication port; the global settings affect the entire message display. For example, if it is a communication setting such as Byte Swap, then it sets byte swap on all communication ports set to Modbus.

1) Name:

The Name cell is used for unique identification of the ViewMarq LED message display. There is a 15 character maximum for the name. This is the name that will be displayed on the startup message when the message display is powered up. It is also the name that is displayed in the Network Editor.

2) ID:

The ID is both the ASCII ID of the message display and the Modbus RTU Slave ID. The ID range is 1 – 247. The ID must be unique for Modbus communications.

3) Heatbeat:

When heart beat is enabled, the display is expecting a communication within the heart beat time (0 to 60 seconds). If this time elapses without a communication, then a Com Error message is shown on the LED display.

4) ASCII Reply:

When the ASCII Reply box is checked, the message display will send a reply message back to the originator of an ASCII message. The reply will either be OK or an error message. See Chapter 9 - Maintenance and Troubleshooting for a description of the reply messages.

NOTE: *If the message is sent to multiple ViewMarq displays (the ID in the ASCII string is 0 or contains multiple ID's), then the message display will not reply even though the ASCII Reply box is checked. This is to prevent multiple displays from talking over each other.*

5) MODBUS:

Byte Swap:

Depending on the Modbus Master configuration, the Byte Swap box may need to be checked in order to match the byte order of the master.

Word Swap:

Depending on the Modbus Master configuration, the Word Swap box may need to be checked in order to match the word order of the master.

NOTE: *If Byte Order is swapped in the LED display, the byte order must also be swapped in the Link for the PC to be able to send messages to the LED display.*

6) Diagnostic LED

The diagnostic LED is a way for a user to see if communications are successful to the panel without removing the back covers to watch the communication LED's flicker. When this box is checked, the lower left LED on the marquee display will stay lit. Each time the message display receives a message the LED will change color to indicate the communication was received. The message may not be displayed due to a Syntax Error or incorrect ID number, but the LED will still change color. It is a good practice to leave this setting on until you are sure everything is working correctly with your LED display and your messages are displaying as expected.

7) Display Syntax Errors

Display Syntax Errors enables the Viewmarq to scroll any Syntax errors that may occur. This setting is enabled by default in the display to aid in the development of your messages. See Chapter 9 - Maintenance and Troubleshooting for a complete list of the error messages.

8) Orientation

The ViewMarq LED message display has a unique feature that allows it to be mounted in either direction horizontally and the message will display right-side-up automatically. This is accomplished by an accelerometer in the CPU board. There may be times when this behavior is not desired. In a high vibration situation, the accelerometer may cause the display to swap back and forth. Or, you may wish to display a message up-side-down. If so, check this box and select an orientation, Normal or Inverted.

NOTE: The "artificial lean" seen in scrolling text is an optical illusion caused by the update rate of the display and the update rate of the human eye. Because of how the ViewMarq display is updated, the text will lean in a different direction depending on whether it is Normal or Inverted.

Communication Setting

Ethernet
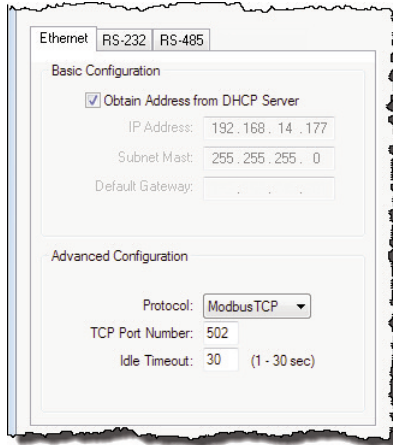
Leave the Obtain Address from DHCP Server checked if your network has a DHCP server and you want the IP Address to be set automatically. Otherwise uncheck this box and set a Static IP Address and Subnet Mask.

Attention!: DHCP is not recommended on a control network where a static IP address is essential. With this feature enabled, a different IP address may be assigned on each power up.

The Modbus TCP Protocol is the only protocol available for Ethernet.

The default port for Modbus TCP is 502. Only change this port number if you have experience with Modbus TCP communications. If the port number is changed, port 502 is still active.

Set the Idle timeout as is necessary on your network conditions depending on network traffic and delays.

RS-232 (Port 1)

1) Protocol:

The available protocols are ASCII and Modbus. This will need to match the master device connected to this port.

2) Node ID:

Node ID is set by the Global ID setting for the message display. It only applies to Modbus. Node ID can be 1 – 247.

3) Baud Rate

The Baud rate must match the master device. The available settings are 38400, 19200, 9600, 4800, 2400 bps. For longer cable runs or where there is communication noise, reducing the baud rate may prevent communication timeouts.

4) Parity

The Parity must match the master device. It can be set to Odd, Even or None. This setting is purely preference or determined by the master device limitations.

5) Stop Bit

The only setting for Stop Bit is 1.

6) Data Bits

The number of Data Bits must match the master device. Modbus only allows 8, so this setting is disabled when the Modbus protocol is selected. When ASCII is selected, the available settings are 7 or 8.

7) Response Delay

ViewMarq is a slave device. The Response delay is used to adjust the Delay time before ViewMarq responds to the other device.

8) RTS Off Delay

Set this parameter to Delay the timing to turn OFF the RTS (Request To Send) signal after message is sent. Typically the RTS Off delay is only required if a media converter is used on the port.
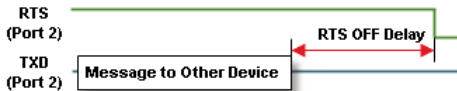
9) RTS On Delay

Set this parameter to Delay the timing to start sending the Message through Port 1 after the RTS (Request To Send) signal turns ON.

RS-485 (Port 2)
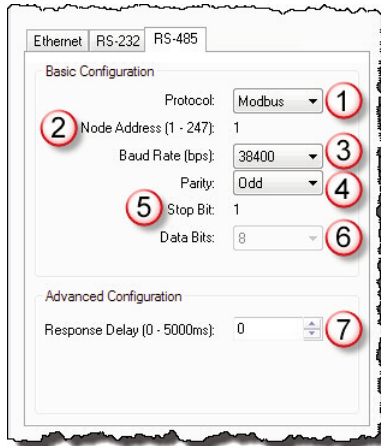
1) Protocol:

The available protocols are ASCII and Modbus. This will need to match the master device you plan to communicate to this port.

2) Node ID

Node ID is set by the Global ID setting for the marquee. Node is not active when ASCII is selected. Node ID can be 1 – 247.

3) Baud Rate

The Baud rate must match the master device. The available settings are 38400, 19200, 9600, 4800, 2400 bps. For longer runs or where there is communication noise, reducing the baud rate may prevent communication timeouts.

4) Parity

The Parity must match the master device. It can be set to Odd, Even or None. This setting is purely preference or determined by the master device limitations.

5) Stop Bit

The only setting for Stop Bit is 1.

6) Data Bits

The number of Data Bits must match the master device. Modbus only allows 8, so this setting is disabled when the Modbus protocol is selected. When ASCII is selected, the available settings are 7 or 8.

7) Response Delay

ViewMarq is a slave device. The Response delay is used to adjust the Delay time before ViewMarq responds to the other device.

Export and Import Configuration

In order to save time, configurations can be saved to your PC and retrieved for later use.
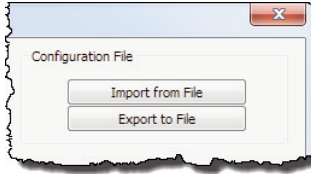
Export Configuration File

To save a configuration, select the **Export to File** button.
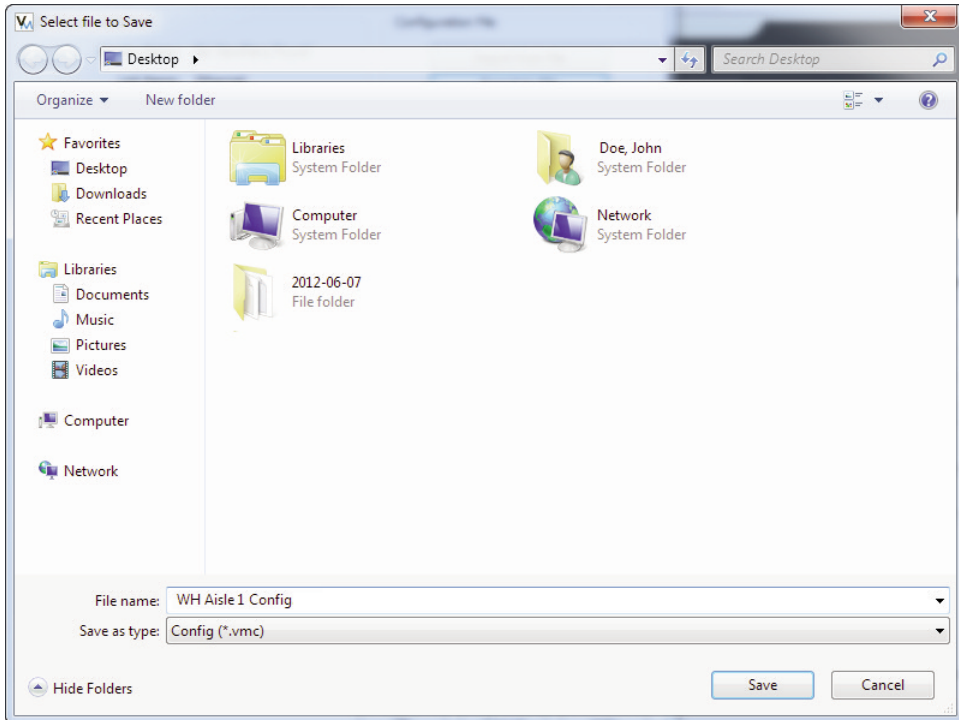
Enter a file name and select **Save**.

Import Configuration File

To use a saved configuration, select the **Import From File** button.
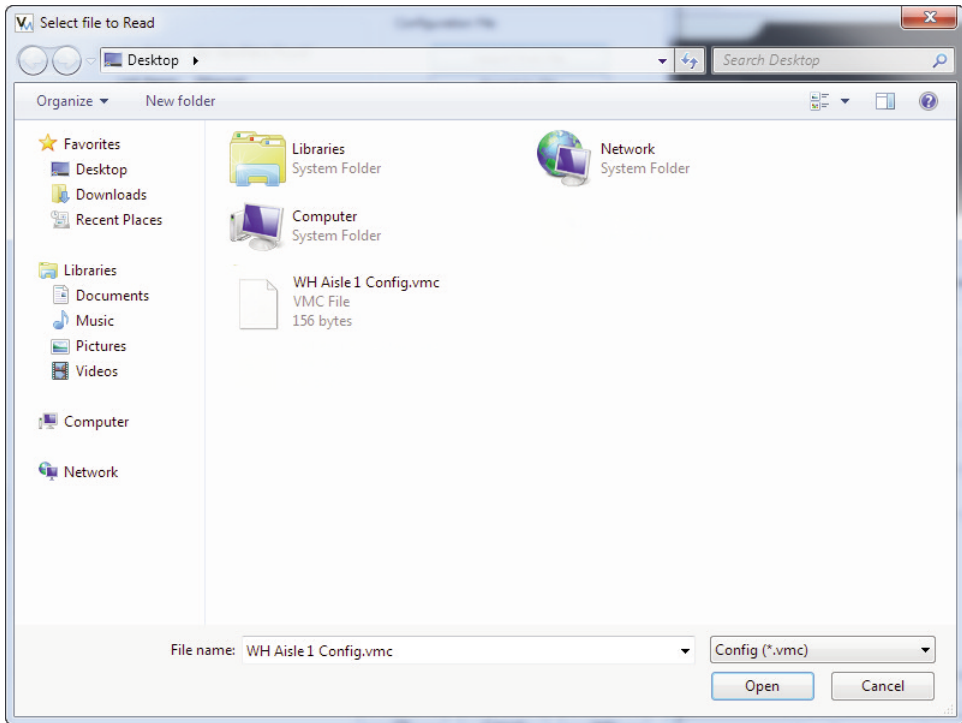
Browse to the desired configuration file location. Choose the file and select **Open**.

SENDING MESSAGES FROM A PLC TO VIEWMARQ

In This Chapter...

Introduction

Command String   📄 Copy

`<ID 0><CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N><CS 0><GRN><T>AutomationDirect.com</T>`

As you create a message in the ViewMarq software, the software creates an ASCII string of commands, called a "Command String" in the ViewMarq software. This string may be pasted into your PLC's instruction or memory, then logic in the PLC will send the message to the ViewMarq.

Click the Copy button to place the command string on to the clipboard. Then paste the string into your PLC instruction.

The ViewMarq LED message display can receive an ASCII Command String by:

Protocol	Connection
ASCII	RS-232 (Port 1), RS485 (Port 2)
Modbus RTU	RS-232 (Port 1), RS485 (Port 2)
Modbus TCP	Ethernet
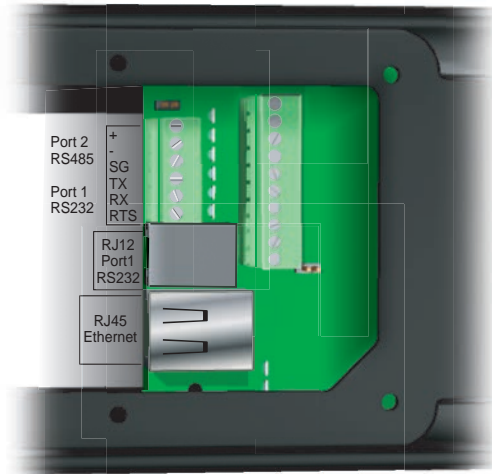
ASCII

When the ViewMarq display serial port and the PLC serial port are both set for ASCII, the PLC may send the Command String directly to the display's port. If the Command String is addressed to a single display, for example <ID 1>, and the display is configured with ASCII Reply turned on, the display will reply with an ASCII string on the same port. If the PLC can receive ASCII strings as well as send them, then you may read the ASCII Reply with the PLC. For more information about **ASCII Reply** see Chapter 6 – Configuring the ViewMarq LED Display.
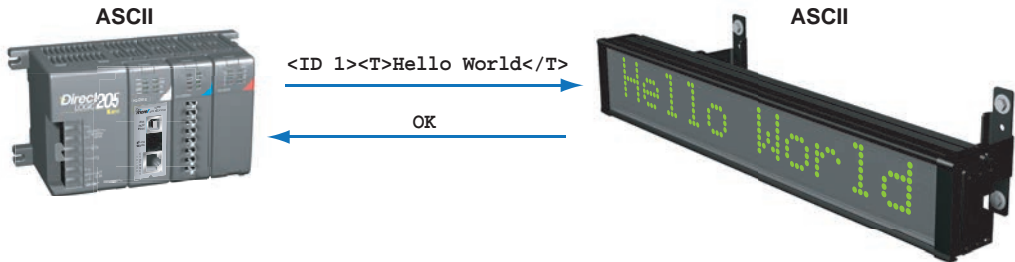
NOTE: *The carriage return termination (0x0d) is required in the Command String that is sent from the PLC.*

Modbus

When using the Modbus RTU protocol with the ViewMarq, the same ASCII Command Strings are used but they are embedded within the data portion of the Modbus message (placed into the Modbus registers). The same ASCII Reply is also embedded within the Modbus data section of a message and can be read from a separate set of Modbus registers.

Command Strings are written to the ViewMarq Command String buffer starting at Modbus Registers 411000 (up to 256 words).

ASCII Replies are read from the ViewMarq Status Buffer starting at Modbus Registers 411500 (up to 256 words).

NOTE: *The carriage return termination (0x0d) is still required in the Command String that is embedded with the Modbus message.*

ATTENTION

Attention!: Command Strings should be sent at least 100 ms apart.

Reading the ASCII Reply when using Modbus

To accurately read the ASCII reply from the ViewMarq display, follow the steps below:

1) Write the Command String to the display Modbus register block starting at 411000

2) Monitor the value in the first Modbus register 411000

3) When the value in register changes to 0 (zero), this indicates the message has been processed and the ASCII Reply buffer has been updated.

4) Read the updated ASCII Reply from the register block starting at 411500

Write: <ID 1><T>Hello World</T> to 41100

ASCII

Read: 0 from 411000

Read: OK from 411500
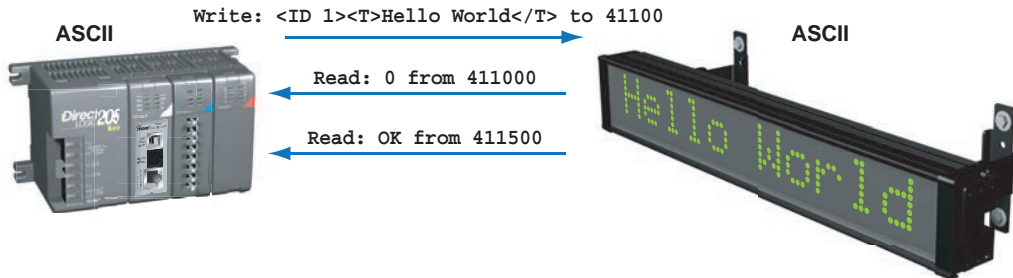
ASCII

String Length Limitations with PLC's

Depending on the PLC, instruction and protocol you may not be able to send an ASCII command string that is 500 characters long in one PLC ASCII or Modbus instruction. For example, the AutomationDirect PLC's have the following limitations.

PLC	Instruction	Protocol	Maximum Characters
CLICK	Send	ASCII	128
CLICK	Send	Modbus	246
P3000	ASCII Out	ASCII	128
P3000	MWX - String	Modbus	128
P3000	MWX - Integers	Modbus	240
Do-more	STREAMOUT	ASCII	1023
Do-more	MWX	Modbus	246
DirectLogic	PRINT	ASCII	128
DirectLogic	VPRINT	ASCII	128
DirectLogic	MWX	Modbus	250

In order to send a String greater than the limit of the PLC instruction, the string will need to be sent in multiple parts. The ViewMarq Display is looking for an <ID n> and a Termination Character $0D (Carriage Return) before it processes its buffer. Therefore a long command string may be sent like this:

PLC Instruction 1
<ID n> *Command String Part 1*

PLC Instruction 2
Command String Part 2 $0D

Sending a Command String in Multiple Parts using a CLICK PLC

CLICK ASCII Send Example

For example, the following Command String is 181 characters and is too long for a CLICK ASCII Send Instruction.

```
<ID  0><CLR>><WIN  0  0  287  31><POS  0  0><CJ><BL  N><CS  0><GRN>
<T>AutomationDirect</T><POS  0  8><CJ><RED><T>#1  in</T><POS  0
16><CJ><T>Service</T><POS  0  24><CJ><AMB><T>12  Years  in  a
row</T>
```

It needs to be sent in two parts:

Command String Part 1

```
<ID  0><CLR><WIN  0  0  287  31><POS  0  0><CJ><BL  N><CS  0><GRN>
<T>AutomationDirect</T>
```

Command String Part 2

```
<POS  0  8><CJ><RED><T>#1  in</T><POS  0  16><CJ><T>Service</T>
<POS  0  24><CJ><AMB><T>12  Years  in  a  row</T>
```

It does not matter where the Command String is broken apart because it will not be processed by the ViewMarq Display until the Termination Character at the end of the Command String is received.

<table>
<tr><td>Command String Part 1</td><td>Command String Part 2</td></tr>
</table>

Send Part 1 of the Command String to the ViewMarq without a Termination Character

```
  ☐ C1
 ─┤↑├─
```

```
Send (Port:2)                         ASCII
Type                                  Static
Byte Swap                             None
Terminate Code                        None   Success
Send Message
"<ID 0><CLR><WIN 0 0 287 31><POS
          0 0><CJ><BL N><CS
  0><GRN><T>AutomationDirect</T>"
```

```
☐ C2
 ○
```

Send Part 2 of the Command String to the ViewMarq with a Termination Character ($0d)

```
  ☐ C2
 ─┤↑├─
```

```
Send (Port:2)                         ASCII
Type                                  Static
Byte Swap                             None
Terminate Code                          CR   Success
Send Message
   "<POS 0 8><CJ><RED><T>#1
         in</T><POS 0
  16><CJ><T>Service</T><POS 0
  24><CJ><AMB><T>12 Years in a
            row</T>"
```

```
☐ C3
 ○
```

CLICK Modbus Send Example

When using Modbus, ASCII strings must be an even number of bytes in length because Modbus registers are 16 bits (2 bytes) long.

Since the Modbus Write instructions are limited depending on the PLC, longer ASCII strings must be sent by using successive Modbus Write instructions. This example uses the CLICK PLC which limits the Modbus Write to 246 characters.

For example, the ASCII string below is 274 characters long including the Termination Character ($0D).

```
<ID  0><CLR><WIN  0  0  287  31><POS  0  0><CJ><BL  N><CS
9><GRN><T>A</T>  <CS  3><AMB><T>utomation</T><CS
5><GRN><T>D</T><CS  3><AMB><T>  irect</T><POS  0  16><CJ><CS
0><GRN><T>      #1  in  Service</T><POS  0  24><RJ><AMB><T>12
</T><RED><T>Years</T><GRN><T>  in</T><AMB><T>  a
</T><RED><T>row</T>
```

To send it to the ViewMarq using a Modbus Write, the ASCII string will need to be sent in two parts. The first is sent without a Termination Character to Modbus address 411000.

Command String Part 1 - 228 Characters

```
<ID  0><CLR><WIN  0  0  287  31><POS  0  0><CJ><BL  N><CS
9><GRN><T>A</T>  <CS  3><AMB><T>utomation</T><CS
5><GRN><T>D</T><CS  3><AMB><T>irect  </T><POS  0  16><CJ><CS
0><GRN><T>      #1  in  Service</T><POS  0  24>  <RJ><AMB><T>12
</T><RED><T>Years</T>
```

The second is sent with a Termination Character ($0D) to Modbus address 411114 = [411000 + 228 characters / 2 characters per byte)].

Command String Part 2 - 44 characters + termination character ($0D) = 45 characters

```
<GRN><T>  in</T>  <AMB><T>  a  </T><RED><T>row</T>
```

Notice the Termination Character is only added to the last string. This increases the length to 45 characters. As you will see, because this length is an odd number of characters, it makes it necessary to add one to the length to keep the number even in the Modbus Write instruction. It is not shown in the string above, because it is not added by ViewMarq, but added in the PLC instruction

Dividing up the Command String

Each Command string may be broken at any location that creates an even length string. Consecutive strings must be sent to the very next Modbus register after the previous string. The last string may be even or odd because the length in the Modbus Write instruction can be rounded up. This extra character is allowed in the last command string because ViewMarq will only process the string up to the Termination Character (Carriage Return). Any characters following this will be ignored

If the strings are NOT written to the correct address, then they may:

1) Overlap causing a syntax error

2) Leave gaps between the parts of the ASCII string that contain unexpected characters or NULL's.

Unexpected characters may cause a syntax error. If a NULL is encountered by the ViewMarq, it will stop processing the string at the NULL and wait until the NULL is replaced

In the example below, the Command String has already been copied into the CLICK memory location TXT1 – TXT272 and the Termination Character ($0D) has been copied into TXT273. (See the following Section "CLICK PLC by AutomationDirect – Modbus" for details about copying the Command String into memory.)

Command String Part 1   **Command String Part 2**

Send		
Com Port:	Port2	
Protocol:	MODBUS	COM Port Setup...

Sending Data Setup

Slave ID (0-247):	2	
Modbus Function Code:	16 - Write Multiple Registers	
Addressing Type:	Modbus 984 Addressing	
Starting Slave Address:	✔ 411000	(400001 to 465535)
Starting Master Address:	✔ TXT1	...
Number of Master Addresses:	228	(2 to 246 : Even)
Word Swap:	OFF	

Status Flags

Sending	✔	...
Success	✔ C11	...
Error	✔ C12	...
Exception Response (Error Code)	✔	...

OK   Cancel   Help

Send		
Com Port:	Port2	
Protocol:	MODBUS	COM Port Setup...

Sending Data Setup

Slave ID (0-247):	2	
Modbus Function Code:	16 - Write Multiple Registers	
Addressing Type:	Modbus 984 Addressing	
Starting Slave Address:	✔ 411114	(400001 to 465535)
Starting Master Address:	✔ TXT229	...
Number of Master Addresses:	46	(2 to 246 : Even)
Word Swap:	OFF	

Status Flags

Sending	✔	...
Success	✔ C13	...
Error	✔ C14	...
Exception Response (Error Code)	✔	...

OK   Cancel   Help

Send Part 1 of the Command String to the ViewMarq Command String buffer starting at Modbus address 411000

C1

Send (Port:2)	MODBUS	
Slave ID	2	C11 Success
Modbus Function Code	16	
Slave Addr	411000	
NO. of Master Addresses	228	C12 Error
Word Swap	OFF	
Master	TXT1	

Send Part 2 of the Command String to the ViewMarq Command String buffer starting at Modbus address 411114
[411000 + (228 characters / 2 characters per byte)]

C11

Send (Port:2)	MODBUS	
Slave ID	2	C13 Success
Modbus Function Code	16	
Slave Addr	411114	
NO. of Master Addresses	46	C14 Error
Word Swap	OFF	
Master	TXT229	

Sending Strings from multiple PLC's

The ViewMarq Message LED message display is a slave device. If more than one PLC, master in this case, is sending strings to the display, the PLC logic must be written in such a way as to prevent the two master PLC's from interfering with each other.

Care must be taken so that one PLC has completed sending its command string before another PLC sends a command string. Once a complete command string (<ID n> and $0D) has been received by the LED Display, time must be given to process it. A delay between complete command strings of 100ms is required

PLC #1

<ID 1><T>Hello World</T>

ASCII

100ms delay between
Command Strings

PLC #2

<ID 1><T>AutomationDirect</T>
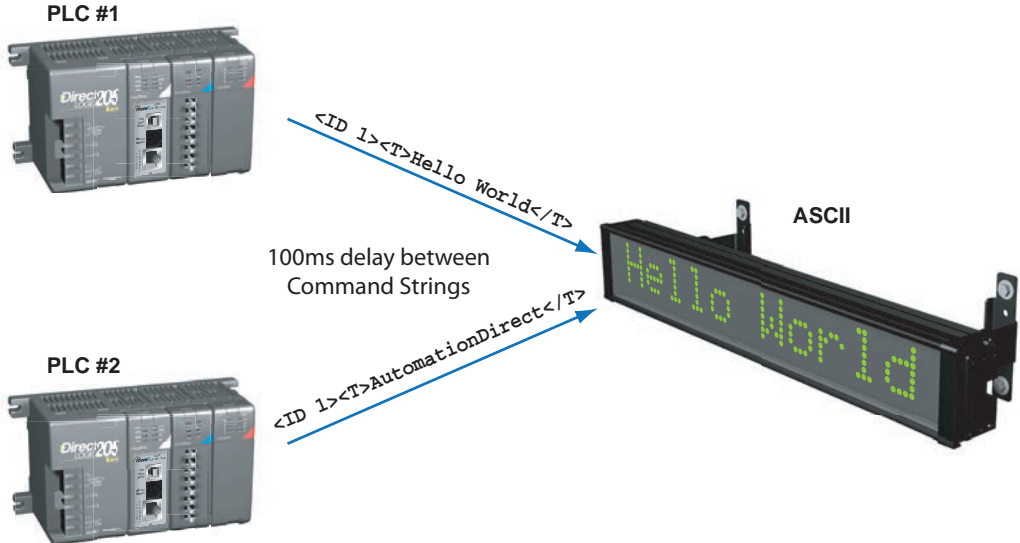
Modbus with Multiple Displays or other Slave Devices.

The ViewMarq is a Standard Modbus Slave. On a multi-drop Modbus RTU network there may be multiple Modbus slave devices including one or more ViewMarq displays. If Modbus requests are being made of alternating slave devices, the ViewMarq displays require a 55ms delay between these packets. If the ViewMarq Display is polled during this time it will not respond and a timeout error will occur in the Master device.

Whenever possible, the poll rate of the master device should be set to 55ms or longer to create this delay between packets. As an example, this can be accomplished with the AutomationDirect Productivity3000 PAC by setting the "Response / Request Delay" to 55ms or higher for the RS-485 port.

For masters devices without a port delay setting the user will need to create delays between communication instructions in their PLC code.
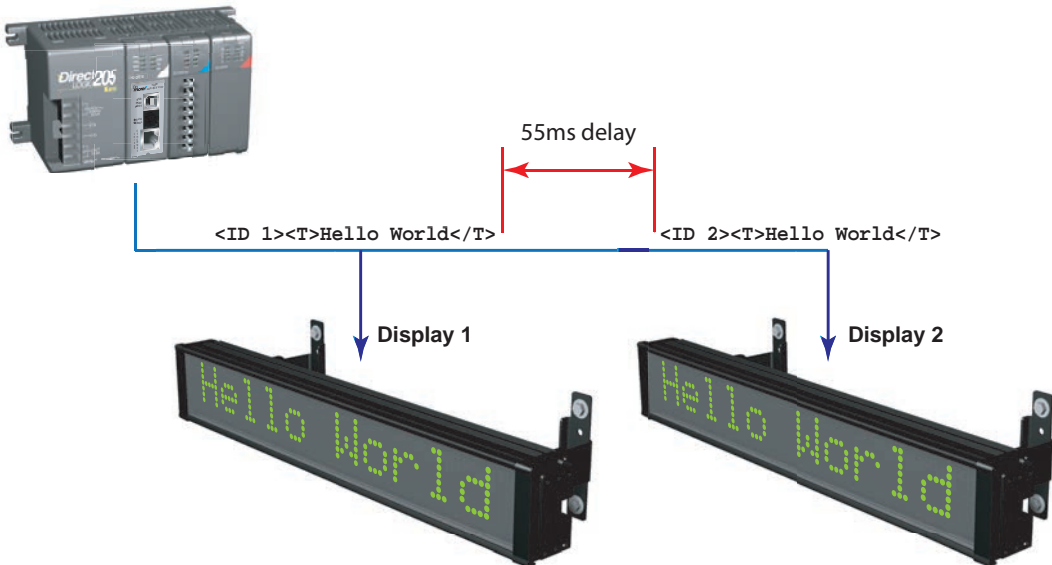
CLICK PLC by AutomationDirect

The CLICK PLC has two communication networks available, RS-232 or RS-485. Either may be used with ViewMarq.

RS-232

Connect the ViewMarq Port 1 RJ-12 to the Click PLC Port 2 RJ-12 using a D0-CBL serial cable.
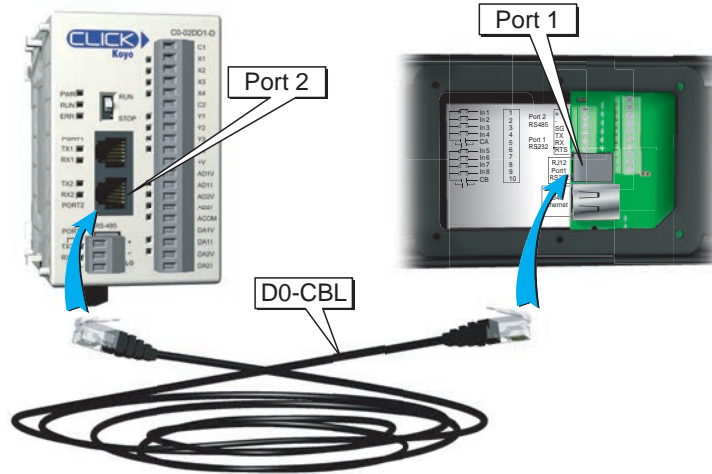
RS-485

Connect the ViewMarq Port 2 terminals to the Click PLC Port 3 terminals using a 3 conductor RS-485 Cable, AutomationDirect Part No. L19773.
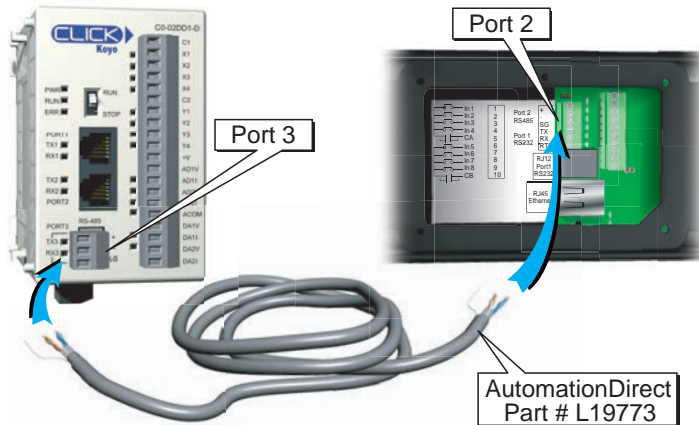
The Click PLC Send instruction may utilize ASCII or Modbus depending on how the port is configured. Both methods are shown.

ASCII

1) Using the CLICK Programming Software configure the CLICK Port 2 or Port 3 for ASCII protocol.
2) Set the port on the CLICK PLC Com Port to match the settings for the ViewMarq display.

3) In the ViewMarq Software, type and configure a message. Copy the string in the Command String window by selecting Copy on the Command String Toolbar.

Take note of <ID 0> in the string above. This identifies which ViewMarq LED display is intended to display the message on a multiple displayl network such as RS485. Refer to Other Tools, LED Display ID in Chapter 5. The default ID for all ViewMarq LED message displays is 1 and is set using the LED Display Configuration Dialog which is covered in Chapter 6.

4) In the CLICK Programming software use a SEND instruction as shown below.
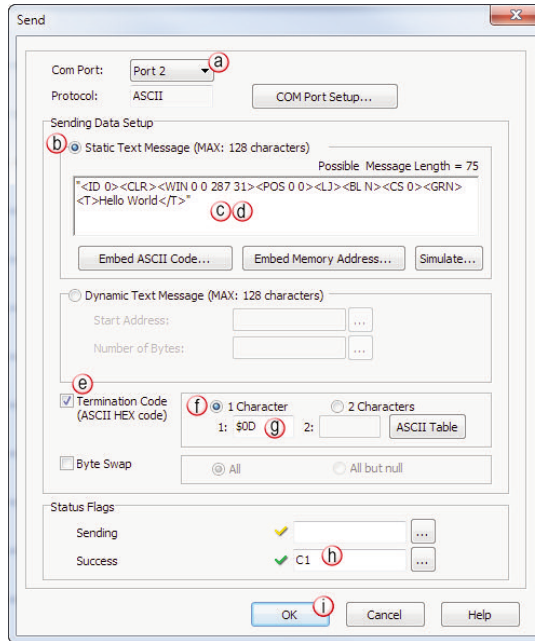
a) Select Port 2 for RS232 (or 3 for RS 485) for the Com Port.

b) Select the Static Text Message radio button

c) Click in the Text Box and press Ctrl-V on your keyboard to paste the String into the instruction.

d) Add quotes to the beginning and end of the string.

e) Select the Termination Code checkbox

f) Select 1 Character radio button

g) Enter $0D or $0A to embed a Carriage Return at the end of the string.

h) Select an address for the success bit. For example C1.

i) Select OK

Example CLICK PLC code for sending ASCII string.
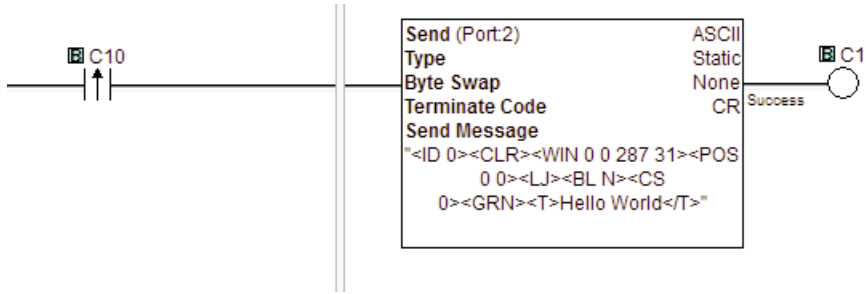
```
                          Send (Port:2)          ASCII
         C10              Type                   Static         C1
        ─┤↑├─             Byte Swap              None    ───( )───
                          Terminate Code           CR  Success
                          Send Message
                          "<ID 0><CLR><WIN 0 0 287 31><POS
                                0 0><LJ><BL N><CS
                              0><GRN><T>Hello World</T>"
```

NOTE: *To prevent the string from being sent with every scan of the PLC use an EDGE triggered (or One Shot) instruction.*

ATTENTION

Attention!: Command Strings should be sent at least 100 ms apart.

Modbus

1) Using the CLICK Programming Software set the Port on the CLICK PLC Com Port to Modbus.

2) Set the port on the CLICK PLC Com Port to match the settings for the ViewMarq display.
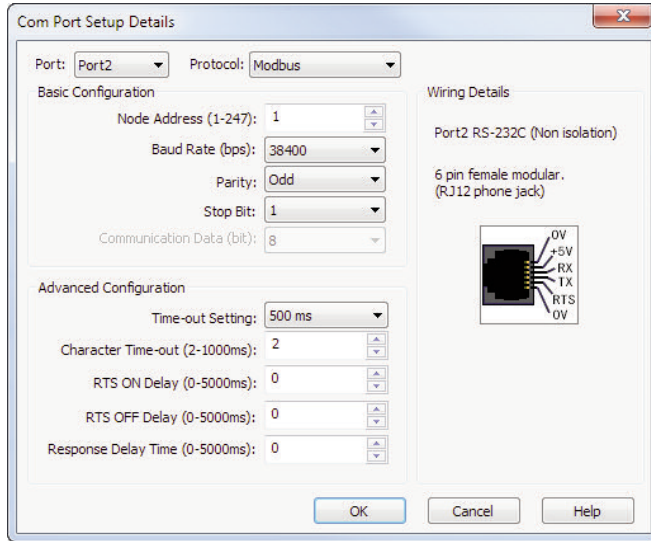
3) In the ViewMarq Software, type and configure a message. Copy the string in the Command String window by selecting Copy on the Command String Toolbar.
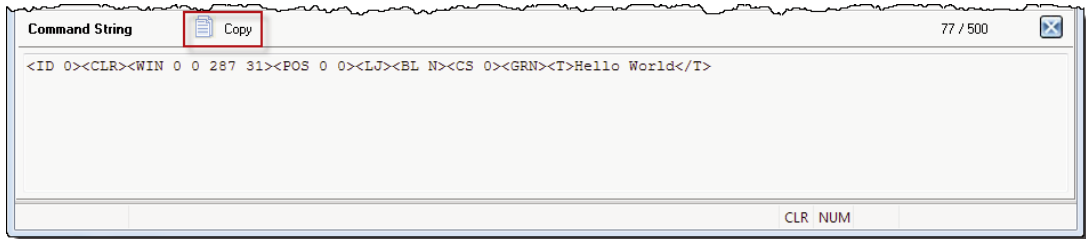
Take note of <ID 0> in the string above. This identifies which ViewMarq LED display is intended to display the message on a multiple display network such as RS485. Refer to Other Tools, LED Display ID in Chapter 5. The default ID for all ViewMarq LED message displays is 1 and is set using the LED Display Configuration dialog which is covered in Chapter 6.

4) Paste the string into a Copy command as shown.
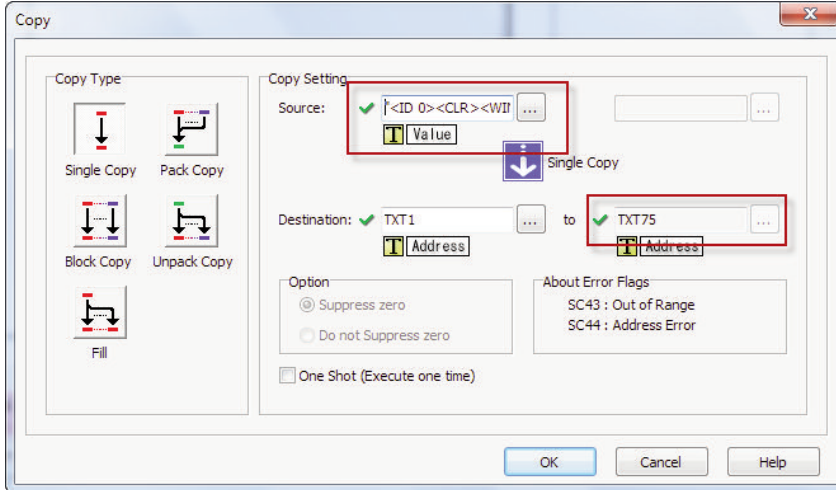
NOTE: *Quotation marks must be placed at the beginning and end of the String that was pasted into the "Source" field of this instruction. Enter in the beginning TXT address of the block where the String will reside. Note the ending address of the destination. This will be used in the next command.*

5) Add a carriage return to the end of the string using another Copy command.
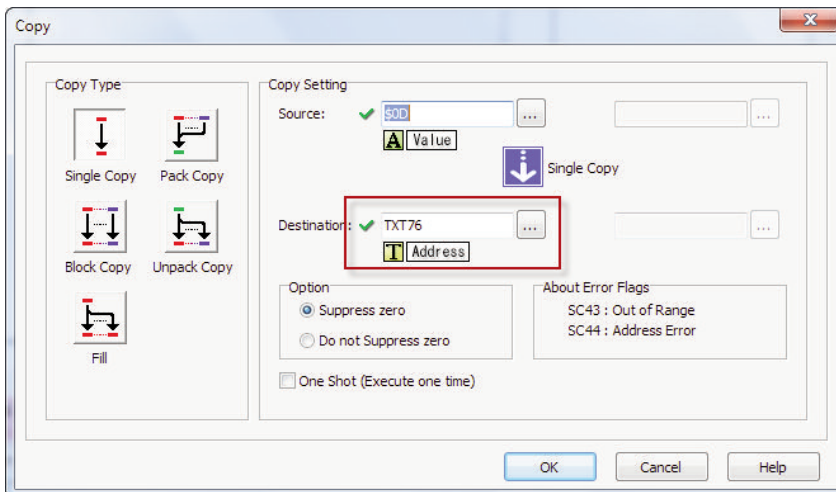
The carriage return character (entered as $0D in the Source field), should be placed into the next TXT address after the end of the block used in the previous COPY command. In this example, the end TXT address from the block used in the previous COPY command was TXT75, so TXT76 is used in this COPY command Destination address.

6) In the CLICK Programming software use a SEND instruction as shown below.
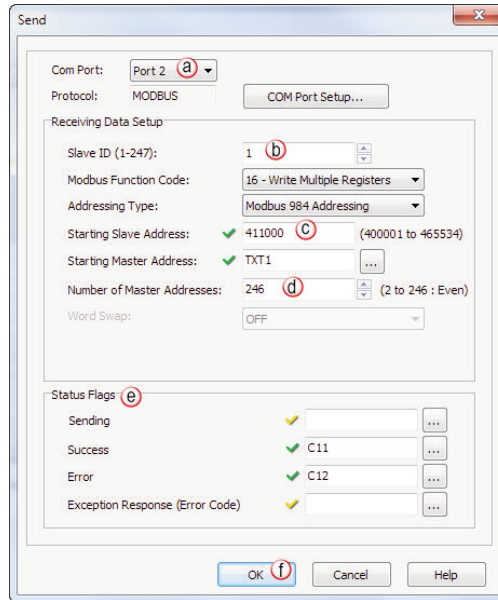
a) Select Port 2 for RS232 (or 3 for RS 485) for the Com Port.

b) Select the Slave ID (Node Address) for the connected ViewMarq.

c) The Command String is written to ViewMarq Command String Buffer starting at Modbus address 411000.

d) The number of Master Addresses may be set to the maximum possible value for the instruction.

e) Configure any Status Flags desired for the program control.

f) Select OK

NOTE: *To prevent the string from being sent with every scan of the PLC use an EDGE triggered (or One Shot) instruction.*

Example CLICK PLC code for sending ASCII string over Modbus.

Copy the ViewMarq String to to TXT memory location

Copy	Single
Src	"<ID 0><CLR><WIN 0 0 2...
Des	⊞ TXT1 ⊞ TXT75

Copy a Carraige Return to the next TXT memory location

Copy	Single
Option	Suppress
Src	Ⓐ $0D
Des	⊞ TXT76

Send the String to the ViewMarq Command String buffer
starting at Modbus address 411000

⊞ C1

Send (Port:2)	MODBUS
Slave ID	1
Modbus Function Code	16
Slave Addr	411000
NO. of Master Addresses	246
Word Swap	OFF
Master	⊞ TXT1

⊞ C11 Success

⊞ C12 Error

Optional Error Checking

You may choose to read the ViewMarq status to make sure the ASCII string was received with no errors.

1) In the Click Programming software, use a Receive instruction as shown below to read the Command Block address until it equals 0. This indicates that the ViewMarq display has finished processing the command and the status is ready to be read.
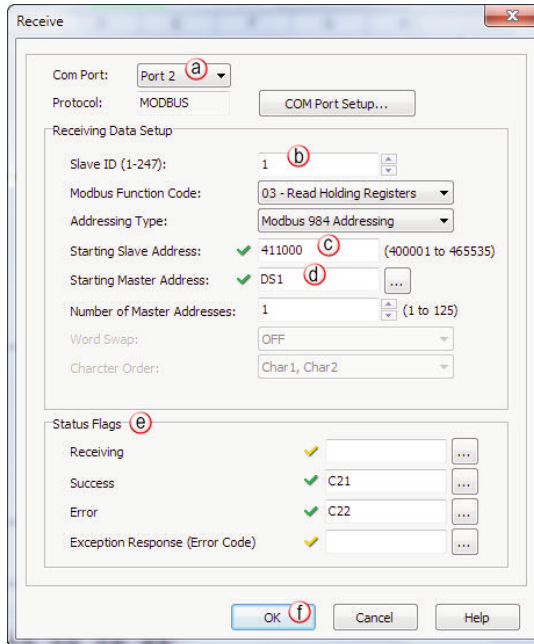
a) Select Port 2 for RS232 (or 3 for RS 485) for the Com Port that you previous set to Modbus.

b) Select the Slave ID (Node Address) for the connected ViewMarq.

c) Read the first address of the Command String Buffer, Modbus address 411000.

d) Choose a Master Address that is an unused, Integer address (such as the DS data type) that can be compared to 0. Once this register is equal to 0, the Status Block can be read.

e) Configure any Status Flags desired for program control.

f) Select OK

2) Once the Command Block is equal to 0, use a Recieve instruction to read the Status Block to verify that the Command String written was accepted by the ViewMarq display.
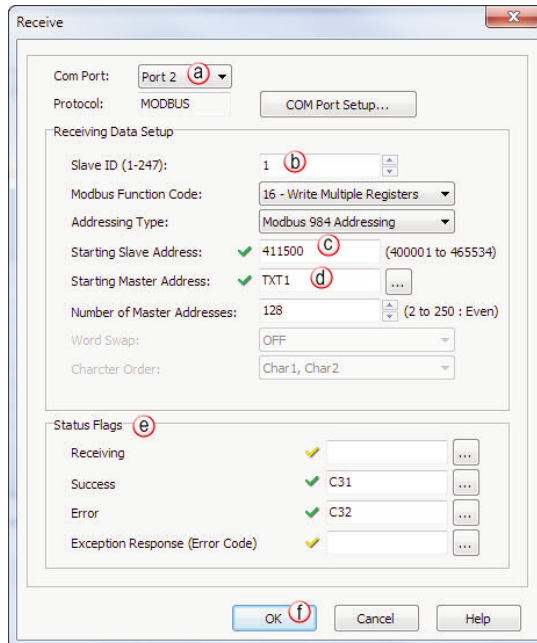
a) Select Port 2 for RS232 (or 3 for RS 485) for the Com Port that you previous set to Modbus.

b) Select the Slave ID (Node Address) for the connected ViewMarq.

c) The Status String can be read from the ViewMarq display starting at Modbus address 411500.

d) Choose an unused, available block of 128 TXT addresses.

e) Configure any Status Flags desired for program control.

f) Select OK

3) Once the string in the Status Block has been read, check the value of the string for the text "OK" using the Search instruction.
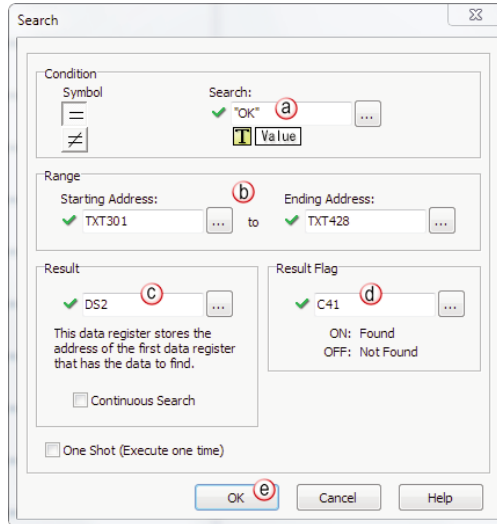
a) Enter search text "OK"

b) Enter the starting and ending addresses of the block of TXT addresses in the previous Receive instruction.

c) Choose an available Integer address for the Result.

d) Choose an available C address for the Result Flag.

e) Select OK.

Example CLICK PLC code for checking the Viewmarq Status Block.

Send the String to the ViewMarq Command String buffer
starting at Modbus address 411000

```
       C1
      ┤↑├
```

Send (Port:2)	MODBUS	C11
Slave ID	1	
Modbus Function Code	16	Success
Slave Addr	411000	
NO. of Master Addresses	246	C12
Word Swap	OFF	
Master	TXT1	Error

Read the the ViewMarq Command String buffer starting
at Modbus address 411000 until it equals 0.

```
      C11         DS1          0
      ┤ ├         ┤ ├         ┤≠├
```

Receive (Port:2)	MODBUS	C21
Slave ID	1	
Modbus Function Code	03	Success
Slave Addr	411000	
NO. of Master Addresses	1	C22
Word Swap	OFF	
Master	DS1	Error

When the Command String Buffer equals 0, read the
ViewMarq Status String buffer starting at Modbus
address 411500

```
      DS1          0
      ┤= ├
```

Receive (Port:2)	MODBUS	C31
Slave ID	1	
Function Code	03	Success
Slave	411500	
Mstr Addr	128	C32
Word Swap	OFF	
Char order	Char1 Char2	Error
Master	TXT301	

When the Status String Buffer Read is successful,
check Status String for the text string "OK"

```
      C31
      ┤ ├
```

Search		DS2
Continuous Search	OFF	Result
Search =	"OK"	
		C41
Range TXT301 TXT428		Found

Productivity3000 PAC by AutomationDirect

The Productivity3000 PAC has three communication networks available, Ethernet, RS-232 or RS-485. Either of the three may be used with ViewMarq.

Ethernet

Connect the ViewMarq RJ-45 Ethernet port to the Productivity3000 RJ-45 Ethernet port using a Cat5e Ethernet crossover cable.
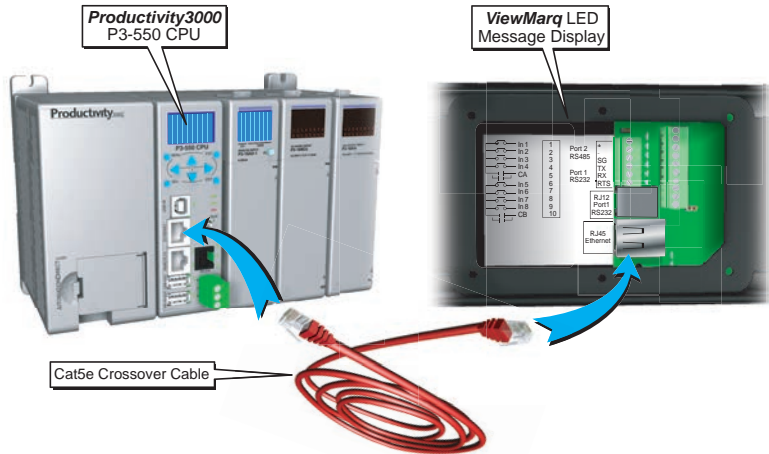
RS-232

Connect the ViewMarq RJ-12 Port 1 to the Productivity3000 RJ-12 RS-232 port using a D0-CBL serial cable.
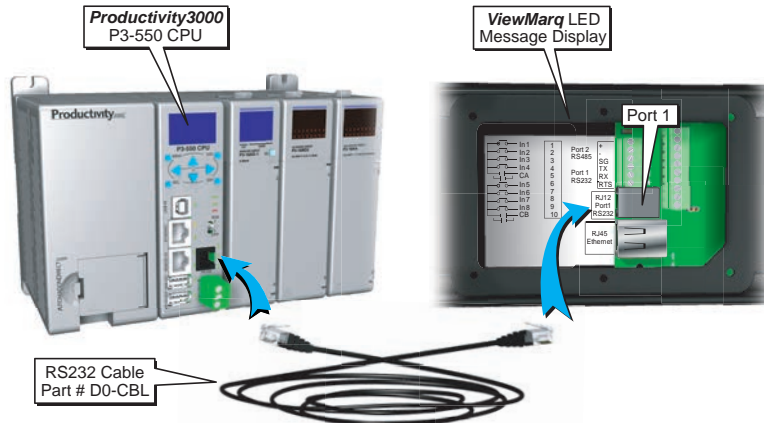
RS-485

Connect the ViewMarq Port 2 terminals to the Productivity3000 RS-485 terminals using a 3 conductor RS-485 Cable, AutomationDirect Part No. L19773.
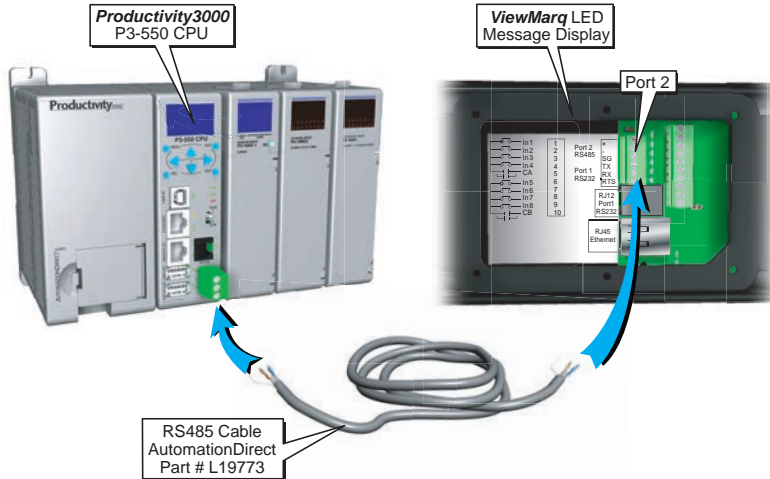
The Productivity3000 PAC may communicate with the ViewMarq LED display by ASCII, Modbus RTU or Modbus TCP.

This section discusses:

• Sending a Command String by ASCII over Serial
• Sending a Command String by Modbus over Serial
• Sending a Command String by Modbus TCP over Ethernet

Embedding the String into PAC memory

1) In the ViewMarq software, type and configure a message. Copy the string in the Command String window by selecting Copy on the Command String Toolbar.

Command String   📄 Copy   77 / 500   ☒

```
<ID 0><CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N><CS 0><GRN><T>Hello World</T>
```

CLR NUM
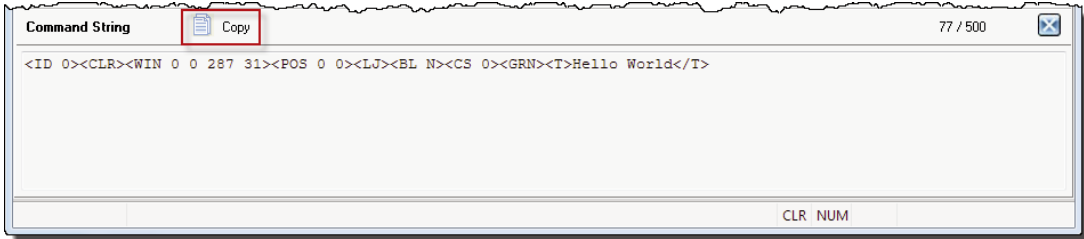
2) In the Productivity Suite Programming Software Paste (Ctrl-V) the Command String into the Copy Data (CPD) Instruction as shown.

Copy Data (CPD)

Source	Destination
"<ID 0><CLR><WIN 0 0 2...	ViewMarq_SW_String

◉ Copy Value   ○ Copy Binary

☐ Show Instruction Comment

Monitor   OK   Cancel   Help
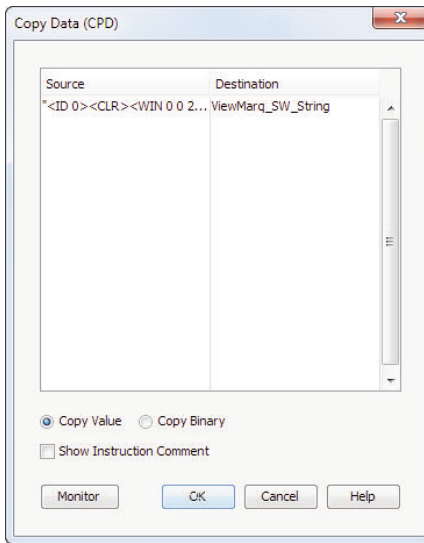
NOTE: *Quotation marks must be placed around the message that has been pasted into the CPD instruction. ViewMarq_SW_String must be a String Data type in the Productivity3000 PAC.*

Sending Strings from Productivity3000 to the ViewMarq display by ASCII over Serial

The PAC port must be configured for "ASCII / Custom Protocol" in order to allow ASCII strings to be sent.

1) Go to Setup>Hardware Configuration and double click on the P3-550 box in the Center window:

2) Click the **Serial Ports** tab.

P3-550		
	RS-232 (RJ-12)	RS-485 (TB Style)
Port Name	CPU-232	CPU-485
Port Security	Read/Write	Read/Write
Protocol	ASCII / Custom Protocol	Modbus RTU
Baud Rate	38.4K	38.4K
Node Address	1	1
Parity	Odd	Odd
Data Bits	8	8
Stop Bits	1	1
RTS Off Delay Time (0-5,000 msec)		
RTS On Delay Time (0-5,000 msec)		
Timeout between query and response (100-30,000 msec)	5 x100 msec	5 x100 msec
Modbus Character Timeout (0-10,000 msec)		
Response/Request Delay (1-5,000 msec)		
Comm Heartbeat Value (2-1,000 sec)		

Module Info   OK   Cancel   Help
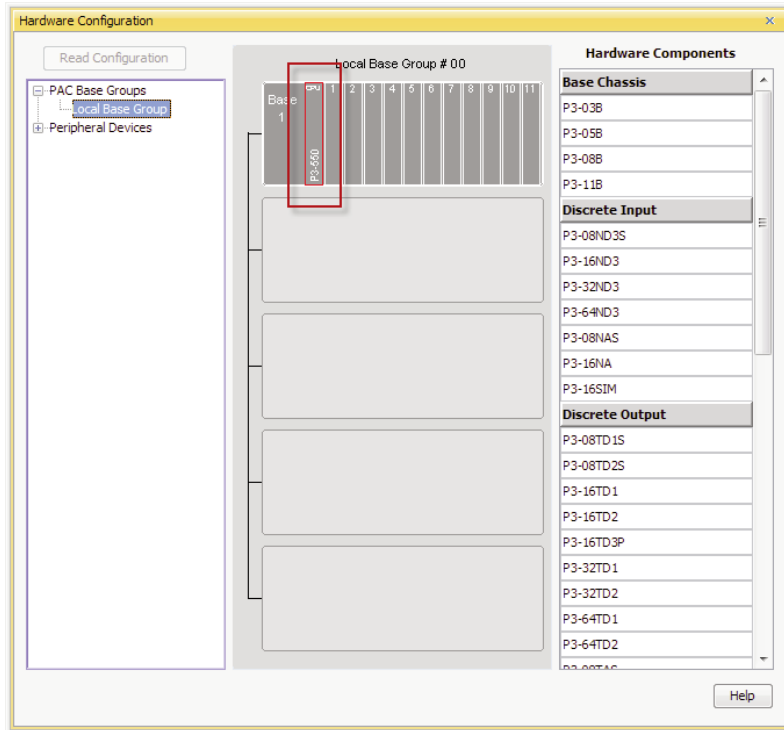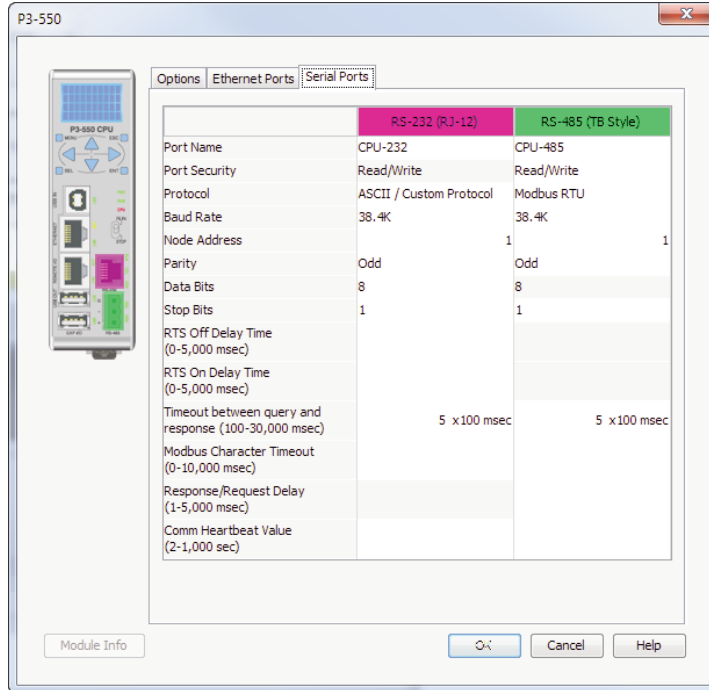
3) Match the Baud Rate, Data Bits and Stop Bits to the serial port settings of the ViewMarq display serial port.

4) Choose **ASCII / Custom Protocol** on the Protocol selection.

5) Now that the port has been configured correctly, use the AOUT (ASCII Out) instruction as shown below to choose the String tag created previously, and to send out the serial port.
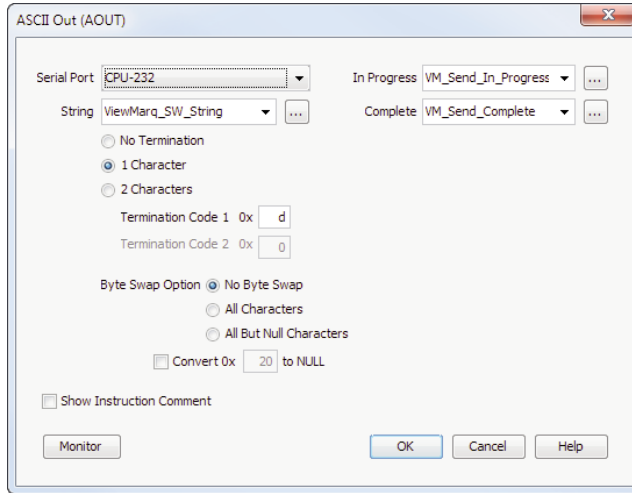
Remember to add the one character termination code for a carriage return, 0x0d.

NOTE: The AOUT instruction is Edge-triggered so the String will be sent only once when the enable leg goes from low to high.

The "ASCII Reply" option in the ViewMarq should be disabled when sending ASCII strings with the AOUT instruction in Productivity3000. See Chapter 6 - Configuring the ViewMarq LED Display for more information. If the application requires more reliable error detection and handshaking consider using Modbus communications instead.

Example P3000 code for sending an ASCII string out the serial port.
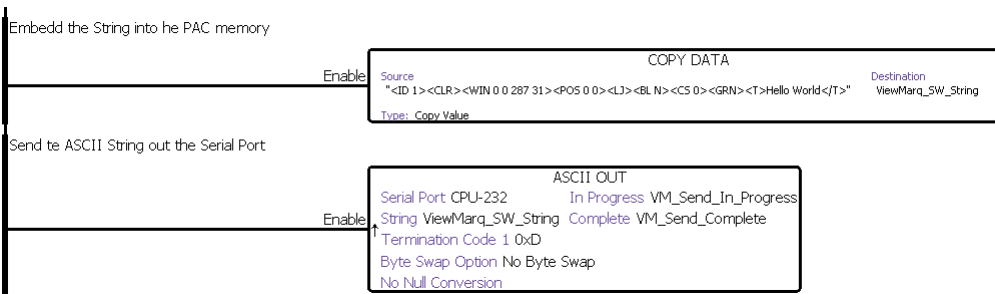
Attention!: Command Strings should be sent at least 100 ms apart.

Sending Strings from Productivity3000 to the ViewMarq display via Modbus

To send a string by Modbus, a couple of steps need to be taken.

1) String length must be calculated.

2) Termination Codes must be added to the end of the string.

Calculating String Length

Add a String Length (SLEN) Instruction to move the length of the string into a Signed Integer 32 tag to be used later.
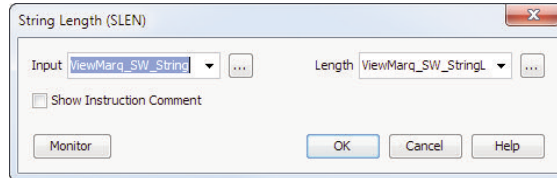
Adding Termination Codes

You will need to add a termination character ("$0d") to the end of the message string. In Productivity3000, non-printable characters cannot be directly inserted into a string tag. Here are the steps to insert the characters at the end of the string:

1) Create an Unsigned Int 8 array and use a CPD instruction copy 0x0D to this array.
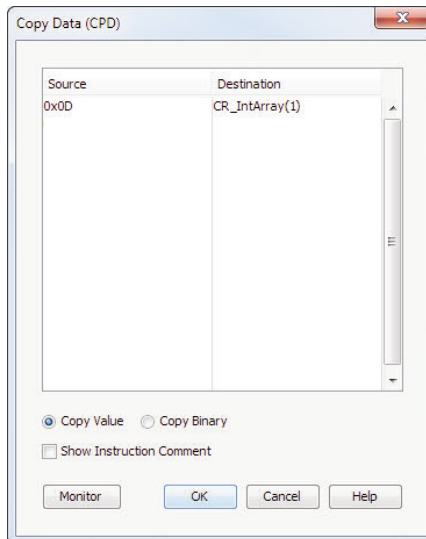
2) Use the CPC (Copy Character) instruction to create a string of 1 character to move 1 byte into.
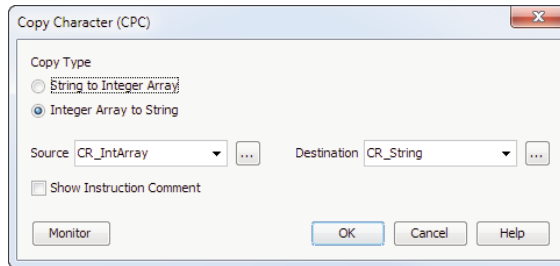
3) Combine the two strings together into a string ready to send out the port using the PKS (Pack String) instruction.
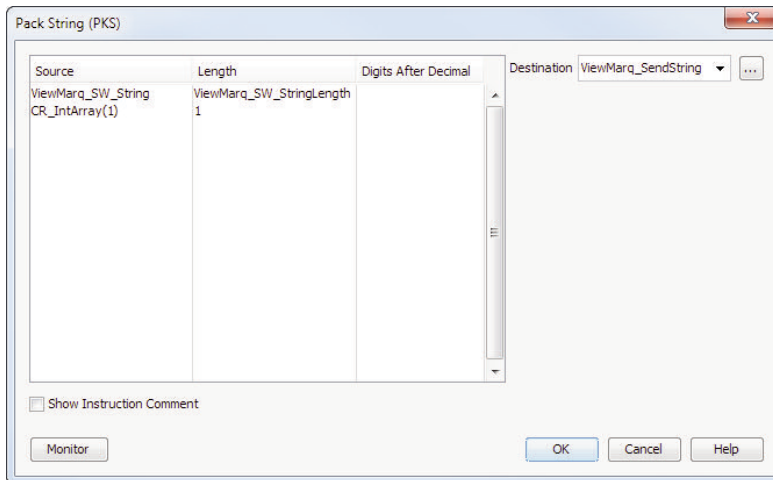
4) Finally, set the destination tag string length to 128.

Example P3000 code for sending an ASCII string out the serial port.

Embedd the String into he PAC memory

Enable
```
                                        COPY DATA
Source                                                              Destination
"<ID 1><CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N><CS 0><GRN><T>Hello World</T>"      ViewMarq_SW_String

Type: Copy Value
```

Calculate the length of the ASCII String

Enable
```
                        STRING LENGTH
Input ViewMarq_SW_String Length ViewMarq_SW_StringLength
```

Create a Carraige Return Integer

Enable
```
              COPY DATA
Source    Destination
0x0d       CR_IntArray(1)

Type: Copy Value
```

Make Carraige Return String

Enable
```
                   COPY CHARACTER
Copy Type Integer Array to String
Source CR_IntArray Destination CR_String
```

Pack ASCII String with Carraige Return String

Enable
```
                              PACK STRING
                                Digits After   Destination ViewMarq_SendString
Source              Length      Decimal Point
ViewMarq_SW_String  ViewMarq_SW_StringLength
CR_String           1
```

Modbus Serial

> **NOTE:** *See previous section for details about preparing the string to send via Modbus.*

The PAC port must be configured for "Modbus RTU" in order to allow raw ASCII strings encapsulated within a Modbus serial packet to be sent.

    1) Go to Setup>Hardware Configuration and double click on the P3-550 box in the Center window
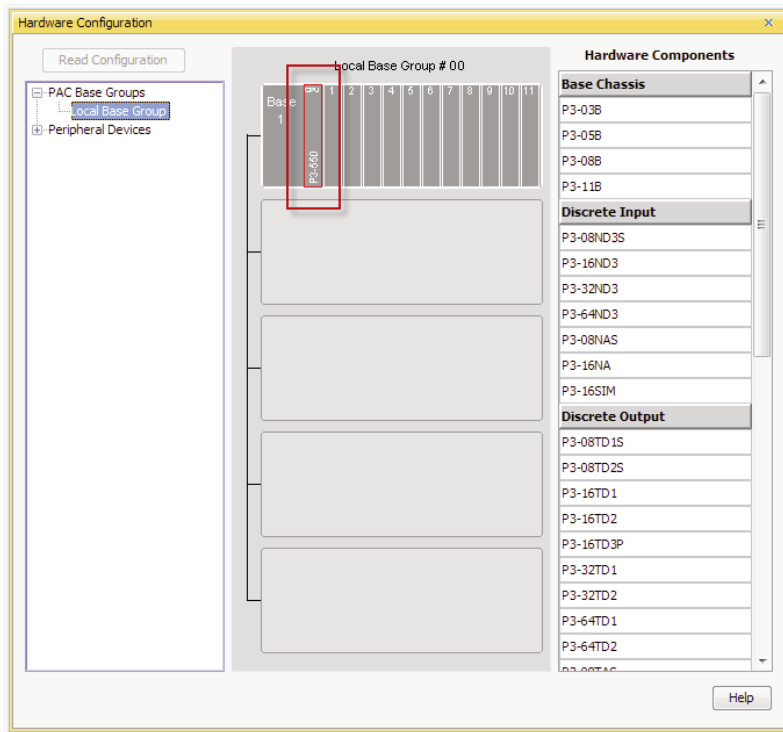
2) Click the **Serial Ports** tab.
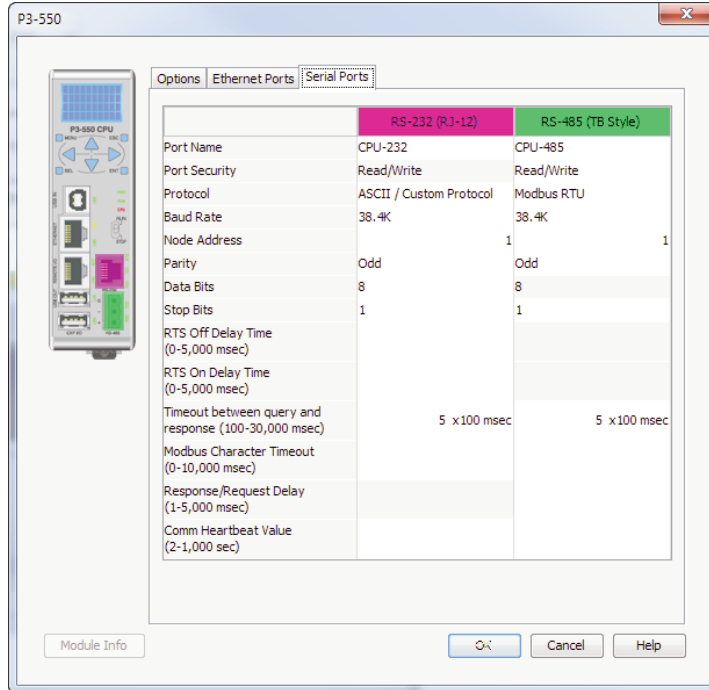
3) Match the Baud rate, Parity, Data bits and Stop bits to the serial port settings of the ViewMarq display serial port.

4) Choose **Modbus RTU** on the Protocol selection.

Now that the port has been configured correctly, use a Modbus Write (MWX) Instruction to write the data over as shown.

1) Choose the Serial Port option and select which CPU the message will be sent from.

    a) The Slave Node Number should match the node in the LED Display Configuration.

    b) Word Swap and Byte Swap should be checked assuming the selections are Off in the ViewMarq display.

    c) Slave Modbus Memory Starting Address is the location of the Command Block within the ViewMarq display (400000+11000 = 411000).

    d) The Function Code should be set to 16 – Write Multiple Registers.

NOTE: *The Productivity3000 MWX instruction uses the Modicon style addressing where the highest digit (4 in this case) is the Modbus data type indicator (Holding Registers).*

e) Choose the "String" option at the bottom and select the String tag previously created in the Pack String Instruction.

f) The Number of Characters should be set at least as high as the Character Count in the ViewMarq Software Command String Viewer.

In this example, 77 would be sufficient, but using the maximum of 128 will also work correctly. ViewMarq ignores any data after the 0D.

g) Create Tags for the "In Progress", "Complete", "Success", "Error", "Timeout" and "Exception Response String" fields to ensure that the Modbus message was configured correctly.

Optional Error Checking

After the Modbus Write is successful, the ViewMarq display will process the message. When the display has finished processing the message it will clear the Command Block (411000).

1) The next step of the logic should be to read the first element of Command Block using a Modbus Read (MRX) Instruction until it reads 0.

a) Slave Modbus Starting Address is the location of the Command Block within the ViewMarq display (400000+11000 = 411000).

b) The Function Code should be set to 3 – Read Holding Registers.

NOTE: *The Productivity3000 MRX instruction uses the Modicon style addressing where the highest digit (4 in this case) is the Modbus data type indicator (Holding Registers).*

c) Choose the "Non-array" option at the bottom and create an Unsigned Integer 16 tag to read the first register of the Command Block into.

d) Keep executing this MRX command until the first register of the Command Block reads 0.

2) After the first register of the Command Block returns a 0, the Reply Status Block should be read using another MRX instruction.

a) Slave Modbus Starting Address is the location of the Reply Status Block within the ViewMarq display. (400000+11500 = 411500).

b) Word Swap and Byte Swap should be checked assuming the selections are Off (default) in the ViewMarq display.

c) The Function Code should be set to 3 – Read Holding Registers.

NOTE: *The Productivity3000 MRX instruction uses the Modicon style addressing where the highest digit (4 in this case) is the Modbus data type indicator (Holding Registers).*

        d) Choose the "String" option at the bottom and create a String tag to read the Status info into. Choose 128 characters as the length.

3) Once the string in the Reply Status Block has been read, check the value of the string for the text "OK" using the Find instruction.
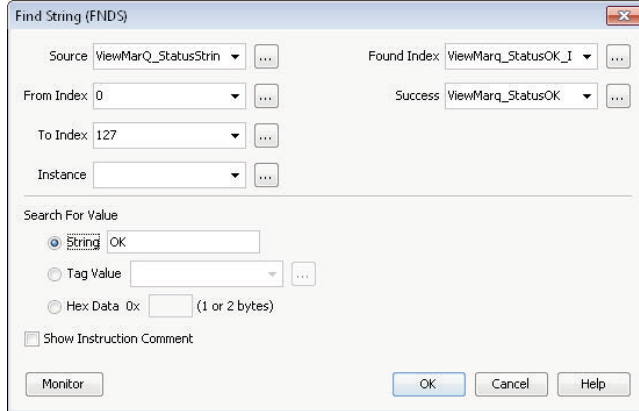
a) Select the Source String where the Status Reply was stored

b) Enter a Found Index Tag, this tag is required, but not important in this case

c) Enter the search Range from 0 to 128

d) Enter a Tag for the Success bit. This tag will be on if "OK" is found

e) Enter search text "OK"

f) Select OK

Example P3000 Code for writing an ASCII string to ViewMarq over Modbus Serial

Write Packed String to ViewMarq

```
                                                                    MODBUS WRITE
                                            Serial Port  CPU-232           In Progress  VM_Write_InProgress
                                            Slave Node Number  1           Complete  VM_Write_Complete
                                            No Automatic Polling           Success  VM_Write_Success
                                            Don't Skip Execution           Error  VM_Write_Error
                                            Word Swap                      Timeout  VM_Write_Timeout
                                            No 32 Bit Data to 16 Bit Mapping   Exception
    SendString                              Enable ↑ Slave Modbus Starting Address  11000   Response String  VM_Write_ExceptionResponse
     ┤ ├                                           Modbus Decimal Addressing
                                                   Modbus Function Code
                                                       16: Write Multiple Registers
                                            Tag Name Mapping
                                               String  String Name  ViewMarQ_SendString
                                                       Number of Chars  128
                                                       Byte Order Swapped
```

When write is complete, read the first element
of the Command Block until it is equal to 0.

```
                                                                    MODBUS READ
                                            Serial Port  CPU-232           In Progress  VM_ChkCMD_InProgress
                                            Slave Node Number  1           Complete  VM_ChkCMD_Complete
                                            No Automatic Polling           Success  VM_ChkCMD_Success
                                            Don't Skip Execution           Error  VM_ChkCMD_Error
  VM_Write_Complete  CMD_BLK_Reg1    0      Word Swap                      Timeout  VM_ChkCMD_Timeout
     ┤ ├             ┤≠├                    No 32 Bit to 16 Bit Mapping    Exception
                                            Enable ↑ Slave Modbus Starting Address  11000   Response String  VM_ChkCMD_ExResp
                                                   Modbus Decimal Addressing
                                                   Modbus Function Code
                                                       3: Read Holding Registers
                                            Tag Name Mapping
                                               Non-Array  Tag
  VM_ChkCMD_Complete     SendString                      CMD_BLK_Reg1
     ┤ ├                 ┤ ├
                                            SendString
                                            ─( RST )
                                            VM_Write_Complete
                                            ─( RST )
                                            VM_ChkCMD_Complete
                                            ─( RST )
```

When Command Block is equal to 0,
Read the Reply Status Block

```
                                                                    MODBUS READ
                                            Serial Port  CPU-232           In Progress  VM_RdStat_InProgress
                                            Slave Node Number  1           Complete  VM_RdStat_Complete
                                            No Automatic Polling           Success  VM_RdStat_Success
                                            Don't Skip Execution           Error  VM_RdStat_Error
                                            Word Swap                      Timeout  VM_RdStat_Timeout
  VM_ChkCMD_Complete  CMD_BLK_Reg1    0     No 32 Bit to 16 Bit Mapping    Exception
     ┤ ├             ┤=├                    Enable ↑ Slave Modbus Starting Address  11500   Response String  VM_RdStat_ExResp
                                                   Modbus Decimal Addressing
                                                   Modbus Function Code
                                                       3: Read Holding Registers
                                            Tag Name Mapping
                                               String  String Name  ViewMarQ_StatusString
                                                       Number of Chars  128
                                                       Byte Order Swapped
```

Turn on Flag is Status if "OK"

```
                                                                    FIND STRING
  VM_RdStat_Complete                        Source  ViewMarQ_StatusString   Found Index  ViewMarq_StatusOK_Index
     ┤ ├                                    From Index  0                   Success  ViewMarq_StatusOK
                                            Enable  To Index  127
                                            Instance  1 (default)
                                            Search For
```

Ethernet Modbus TCP

Sending messages from the Productivity3000 to the ViewMarq display via Modbus TCP is the same as sending messages over Modbus RTU with one change to the MRX and MWX instructions.

Instead of choosing the "Serial Port" option in the MWX and MRX instructions, choose the "Ethernet Port" option and enter in the IP address of the ViewMarq display. Leave the TCP Port Number as 502 and the Slave Node Number as 255.

1) With the Ethernet port properly configured in the Productivity3000 PAC, use a Modbus Write (MWX) Instruction to write the data over as shown.

a) Enter in the IP address of the ViewMarq display.

b) Leave the TCP Port Number as 502 and the Slave Node Number as 255.

c) Word Swap and Byte Swap should be checked assuming the selections are Off (default) in the ViewMarq display.

d) Slave Modbus Memory Starting Address is the location of the Command Block within the ViewMarq display (400000+11000 = 411000).

e) The Function Code should be set to 16 – Write Multiple Registers.

NOTE: *The Productivity3000 MRX instruction uses the Modicon style addressing where the highest digit (4 in this case) is the Modbus data type indicator (Holding Registers).*

f) Choose the "String" option at the bottom and select the String tag previously created in the Pack String instruction.

g) Number of Characters will be 128 to capture the entire Strength Length in this case.

NOTE: *Be sure to select the "Byte Swap" checkbox option when writing String data to the ViewMarq display from the Productivity3000.*

h) Create Tags for the "In Progress", "Complete", "Success", "Error", "Timeout" and "Exception Response String" fields to ensure that the Modbus message was configured correctly.

7

Optional Error Checking

After the Modbus Write is successful, the ViewMarq display will process the message. When the display has finished processing the Message it will clear the Command Block (411000).

1) The next step of the logic should be to read the first element of Command Block using a Modbus Read (MRX) Instruction until it reads 0.

a) Slave Modbus Starting Address is the location of the Command Block within the ViewMarq display (400000+11000 = 411000).

b) The Function Code should be set to 3 – Read Holding Registers.

NOTE: *The Productivity3000 MRX instruction uses the Modicon style addressing where the highest digit (4 in this case) is the Modbus data type indicator (Holding Registers).*

c) Choose the "Non-array" option at the bottom and create an Unsigned Integer 16 tag to read the first register of the Command Block into.

d) Keep executing this MRX command until the first register of the Command Block

reads 0.

2) After the first register of the Command Block returns a 0, the Reply Status Block should be read using another MRX instruction.

a) Slave Modbus Starting Address is the location of the Command Block within the ViewMarq display (400000+11500 = 411500).

b) Word Swap and Byte Swap should be checked assuming the selections are Off in the ViewMarq display.

c) The Function Code should be set to 3 – Read Holding Registers.

NOTE: *The Productivity3000 MRX instruction uses the Modicon style addressing where the highest digit (4 in this case) is the Modbus data type indicator (Holding Registers).*

d) Choose the "String" option at the bottom and create a String tag to read the Status info into. Choose 128 characters as the length.

Example P3000 code for writing an ASCII string to ViewMarq over Modbus Ethernet.

Write Packed String to ViewMarq

```
                                                              MODBUS WRITE
                                    Ethernet Port  CPU-ETH-Ext        In Progress  VM_Write_InProgress
                                       IP Address  10.11.0.109        Complete  VM_Write_Complete
                                    TCP Port Number  502              Success  VM_Write_Success
                                    Slave Node Number  255            Error  VM_Write_Error
                                    No Automatic Polling              Timeout  VM_Write_Timeout
                                    Don't Skip Execution              Exception
        SendString                  Word Swap                         Response String  VM_Write_ExceptionResponse
                          Enable    No 32 Bit Data to 16 Bit Mapping
         ] [                        Slave Modbus Starting Address  11000
                                       Modbus Decimal Addressing
                                       Modbus Function Code
                                          16: Write Multiple Registers
                                    Tag Name Mapping
                                       String  String Name  ViewMarQ_SendString
                                                Number of Chars  128
                                                Byte Order Swapped
```

When write is complete, read the first element
of the Command Block until it is equal to 0.

```
                                                              MODBUS READ
                                    Ethernet Port  CPU-ETH-Ext        In Progress  VM_ChkCMD_InProgress
                                       IP Address        10.11.0.109  Complete  VM_ChkCMD_Complete
                                    TCP Port Number  502              Success  VM_ChkCMD_Success
                                    Slave Node Number  255            Error  VM_ChkCMD_Error
                                    No Automatic Polling              Timeout  VM_ChkCMD_Timeout
                                    Don't Skip Execution              Exception
   VM_Write_Complete  CMD_BLK_Reg1      0    Word Swap                Response String  VM_ChkCMD_ExResp
                          Enable    No 32 Bit to 16 Bit Mapping
     ] [          ]≠[                 Slave Modbus Starting Address  11000
                                       Modbus Decimal Addressing
                                       Modbus Function Code
                                          3: Read Holding Registers
                                    Tag Name Mapping
                                       Non-Array  Tag
                                                CMD_BLK_Reg1

   VM_ChkCMD_Complete      SendString       SendString
                                               ( RST )
     ] [              ] [                  VM_Write_Complete
                                               ( RST )
                                           VM_ChkCMD_Complete
                                               ( RST )
```

When Command Block is equal to 0,
Read the Reply Status Block

```
                                                              MODBUS READ
                                    Ethernet Port  CPU-ETH-Ext        In Progress  VM_RdStat_InProgress
                                       IP Address        10.11.0.109  Complete  VM_RdStat_Complete
                                    TCP Port Number  502              Success  VM_RdStat_Success
                                    Slave Node Number  255            Error  VM_RdStat_Error
                                    No Automatic Polling              Timeout  VM_RdStat_Timeout
                                    Don't Skip Execution              Exception
  VM_ChkCMD_Complete  CMD_BLK_Reg1     0    Word Swap                 Response String  VM_RdStat_ExResp
                          Enable    No 32 Bit to 16 Bit Mapping
     ] [          ]=[                 Slave Modbus Starting Address  11500
                                       Modbus Decimal Addressing
                                       Modbus Function Code
                                          3: Read Holding Registers
                                    Tag Name Mapping
                                       String  String Name  ViewMarQ_StatusString
                                                Number of Chars  128
                                                Byte Order Swapped
```

Turn on Flag is Status if "OK"

```
                                                              FIND STRING
   VM_RdStat_Complete               Source  ViewMarQ_StatusString  Found Index  ViewMarq_StatusOK_Index
                          Enable    From Index  0                  Success  ViewMarq_StatusOK
     ] [                            To Index  127
                                    Instance  1 (default)
                                    Search For
```

7

Do-More PLC by AutomationDirect

The Do-More PLC has three communication networks available; RS-232, RS-485 and Ethernet. Any of the three may be used with the ViewMarq LED display.

RS-232

Connect the ViewMarq RJ-12 Port 1 to the Do-more H2-DM1E CPU RJ-12 port or H2-SERIO-4 RJ-12 port using a D0-CBL serial cable.
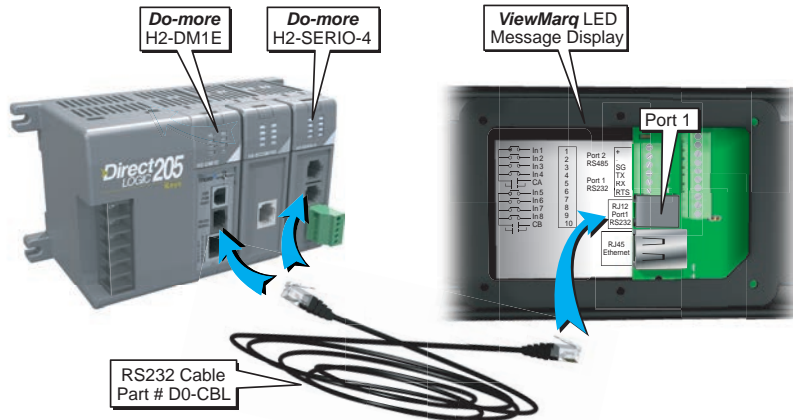
RS-485

Connect the ViewMarq Port 2 terminals to the Do-more H2-SERIO-4 RS-485 terminals using a 3 conductor RS-485 Cable, AutomationDirect Part No. L19773
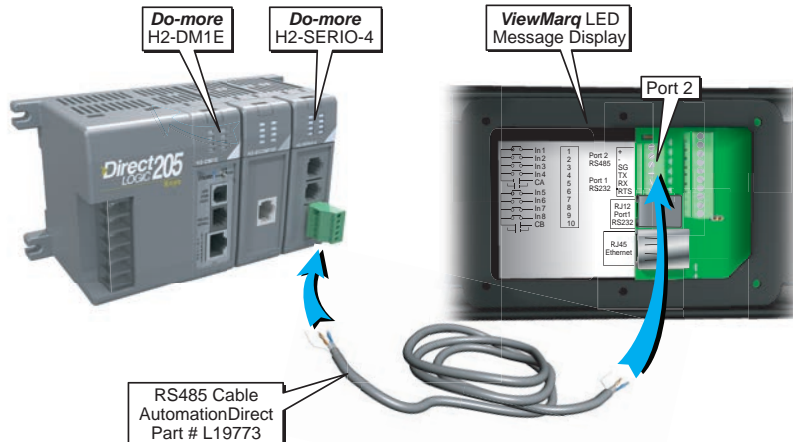
Ethernet

Connect the ViewMarq RJ-45 Ethernet port to the Do-more H2-DM1E Ethernet port using a Cat5e Ethernet cable.
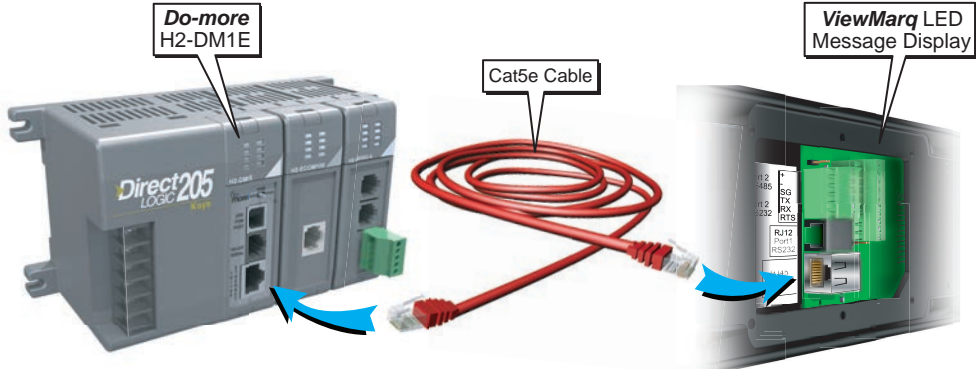
ASCII

1) Use the Do-More Programming Software to configure the PLC port for "General Purpose" in order to allow ASCII strings to be sent. Go to PLC>System Configuration and open the CPU Configuration dialog shown below and select "General Purpose".
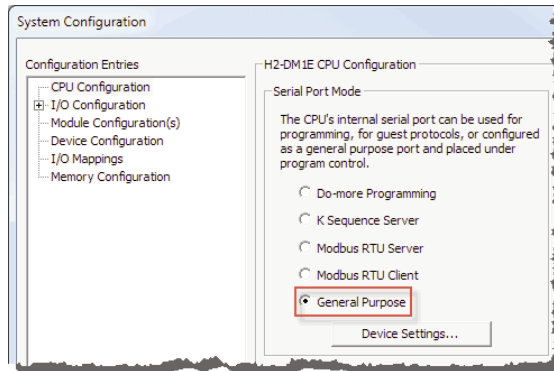
2) Click on the "Device Settings" in this window to match the Baud Rate, Data Bits, Stop Bits, and Parity to the settings of the serial port in the ViewMarq display:
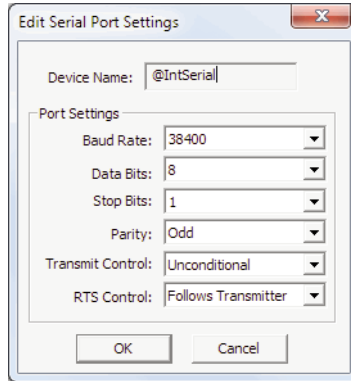
Sending messages to the ViewMarq Display with the Do-More PLC

1) In the ViewMarq software, type and configure a message. Copy the string in the Command String window by selecting Copy on the Command String Toolbar.
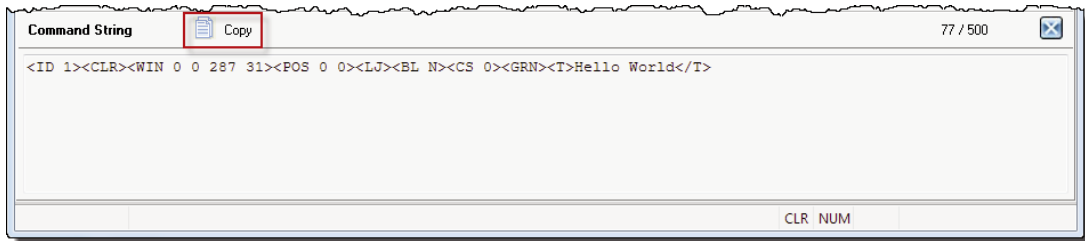
2) In the Do-More Programming Software use the STRPRINT command to embed the Command String into Do-More PLC memory.

3) Paste the Command into a STRPRINT instruction

> **Attention!: Quotation marks must be placed around the message that has been pasted into the STRPRINT instruction. You will also need to add a termination character ("$0d") to the end of the message string. Use the "SL" memory type as it allows for more characters (256).**

```
STRPRINT ──────────────── Print to String

Print to                                     SL0
No space separator
Print Script "<ID 1><CLR><WIN 0 0 287 31><POS
0 0><LJ><BL N><CS 0><AMB><T>Hello
</T><GRN><T>World</T>" "$0D"
```
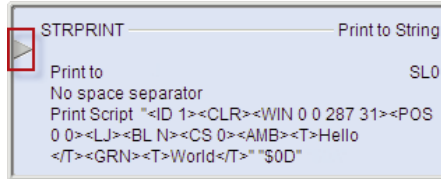
> *NOTE: The STRPRINT instruction is an "Edge Triggered" instruction (as indicated by the Gray arrow) so if any changes within the text of the instruction (such as a dynamic variable), it will need to be re-triggered. If the data is changing often, consider using a transitioning bit within the contact such as the ST5 (100ms toggle) bit.*

Sending the String to the ViewMarq

Now that the port has been configured correctly and the Command String is embedded into the memory, use the STREAMOUT instruction to choose the String to send out the serial port. Ensure that the "Device" selected is for the serial port that was configured earlier.

```
✓ ✗ ☒ ?                                           ●
STREAMOUT ──────────────── Stream Out Data to Device

Device                                   @IntSerial    ▾
┌─ Data Source ──────────────────────────────────────┐
│   ⦿ String Structure          SL0                   │
│                                                     │
│   ○ Numeric Data Block        Create Byte Buffer... │
│       Buffer Start            D0                    │
│       Number of Bytes to Output  64                 │
└─────────────────────────────────────────────────────┘
┌─ Endian Settings ──────────────────────────────────┐
│   ☐ Swap Byte     ☐ Swap Word                       │
└─────────────────────────────────────────────────────┘
☑ Flush INPUT device first
On Success:  ⦿ Set bit  ○ JMP to Stage    C10
    On Error:  ⦿ Set bit  ○ JMP to Stage    C11
```
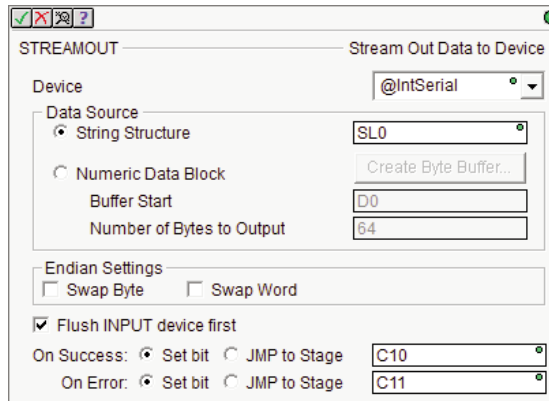
> *NOTE: The STREAMOUT instruction is an "Edge Triggered" instruction so that the enable leg logic must transition from OFF to ON for every message being sent to the display.*

Reading the ASCII Reply from the ViewMarq (Optional)

If the "ASCII Reply" option is enabled in the "LED Display Configuration" settings AND you are sending to a single ID (example <ID 1>), you may use a STREAMIN instruction to receive the reply from the display and load into a String as shown.

NOTE: *An ID of 0 causes the Viewmarq Display to not reply.*

Use the String Compare instruction to check the string for the text "OK" and set the discrete flag.

NOTE: *The STREAMIN instruction is an "Edge Triggered" instruction so that the enable leg logic must transition from OFF to ON for every message being received into the serial port.*

ATTENTION

Attention!: Command Strings should be sent at least 100 ms apart.

Example Do-more logic for writing an ASCII string to ViewMarq over ASCII.ok

Load the String into PLC Memory

```
C0
─┤ ├─                                          STRPRINT                    Print to String
                                               Print to                              SL0
                                               No space separator
                                               Print Script  "<ID 1><CLR><WIN 0 0 287 31><POS
                                               0 0><LJ><BL N><CS 0><GRN><T>Hello
                                               World</T>$0D"
```

Send the String out of the Serial Port to the to the ViewMarq

```
C1
─┤↑├─                                          STREAMOUT      Stream Out Data to Device
                                               Device                         @IntSerial
                                               String Stream                        SL0
                                               Flush INPUT device first
                                               On Success, Set bit                  C10
                                               On Error, Set bit                    C11
```

Read the ASCII Reply from the ViewMarq

```
C10
─┤↑├─                                          STREAMIN       Stream in Data from Device
                                               Device                         @IntSerial
                                               Complete when length is 128 characters OR
                                               received the delimiter character 0x0D
                                               String Structure                     SL1
                                               On Success, Set bit                  C13
                                               On Error, Set bit                    C14
```

Compare the ASCII Reply to "OK"

```
C13
─┤ ├─                                          STRCMP                      String Compare
                                               First String                         SL1
                                               Case Sensitive Yes, upper/lower case are different
                                               Set If Equal                         C15
                                               Second String or Text  "OK"
```

Modbus

Sending Strings from Do-More to the ViewMarq display via Modbus Serial

1) Use the Do-More Programming Software to configure the PLC port for Modbus RTU client in order to send Strings encapsulated within a Modbus RTU message to the ViewMarq display. Go to PLC>System Configuration and open the CPU Configuration dialog shown below and select "Modbus RTU Client".
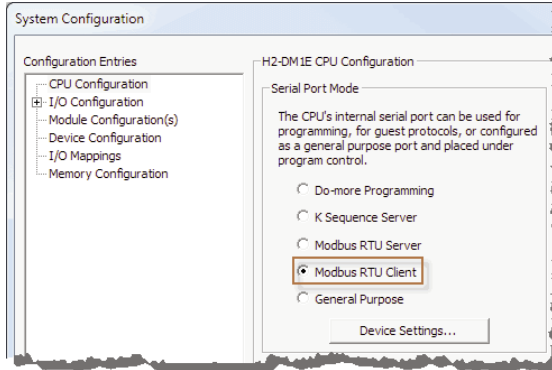
2) Click on Device Settings to configure the Baud Rate, Parity, Data bits and Stop bits to the settings that match the serial port on the ViewMarq display.
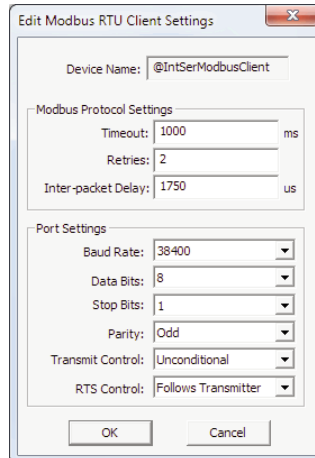
To send messages to the ViewMarq display with the Do-More PLC:

Embed the String in PLC Memory

1) In the ViewMarq software, type and configure a message. Copy the string in the Command String window by selecting Copy on the Command String Toolbar.

Command String	📄 Copy	77 / 500	☒

`<ID 0><CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N><CS 0><GRN><T>Hello World</T>`

CLR NUM

2) In the Do-More Programming Software use the STRPRINT command to embed the Command String into Do-More PLC memory.

3) Paste the Command into a STRPRINT instruction

STRPRINT — Print to String

Print to   SL0

☐ Append to String
☐ Automatically insert space after each term

Print Script

`"<ID 1><CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N><CS 0><AMB><T>Hello World</T>" "$0D"`

Attention!: Quotation marks must be placed around the message that has been pasted into the STRPRINT instruction. You will also need to add a termination character ("$0d") to the end of the message string. Use the "SL" memory type as it allows for more characters (256).

STRPRINT — Print to String

Print to   SL0
No space separator
Print Script `"<ID 1><CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N><CS 0><AMB><T>Hello </T><GRN><T>World</T>" "$0D"`

NOTE: The STRPRINT instruction is an "Edge Triggered" instruction (as indicated by the Gray arrow) so if any changes within the text of the instruction (such as a dynamic variable), it will need to be re-triggered. If the data is changing often, consider using a transitioning bit within the contact such as the ST5 (100ms toggle) bit.

Move the String to V-Memory

1) In the Do-More PLC, Strings cannot be directly accessed using Modbus commands so the data must be moved into integer memory first. To do this, use a STRGETB command.

STRGETB	Get Bytes Out of a String
Get out of String	SL0
Start at Index	0
Length in Bytes	SL0.Length
Into Starting Element	V0
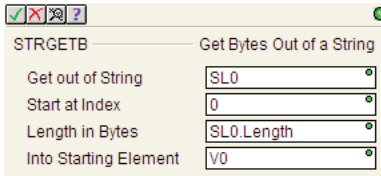
2) This instruction will move the characters from the String and put them into V0. We use the .Length member of the String in the length field so that if our String size changes, it will still move the correct amount over.

3) Next adjust the value to convert it from bytes to words since the V memory type is 16 bit and that is what is required to be sent on Modbus. So divide the byte count by 2 (2 bytes per word) and then add 1 in case the String length works out to an odd number of bytes so that 1 character of the String doesn't get chopped off.

MATH	Calculate Expression
Result	V1000
Expression	
(SL0.Length / 2) + 1	
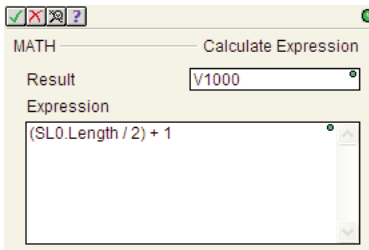
Sending the String to the ViewMarq

1) Now the data is prepared to be written over Modbus and with the calculated request size. Use a MWX instruction to write the data over.

a) The Unit ID should match the node address setting of the serial port configuration in the ViewMarq LED Display Configuration.

b) The Function Code should be set to 16 – Write Multiple Registers.

c) The Modbus Offset Address is the location of the Command Block within the ViewMarq display and is 11000.

d) Number of Modbus Registers should contain the V memory location that was the result of our calculated Modbus request size from above.

e) From Do-More Memory Address should be the resulting memory location of the STRGETB instruction that contains the String data converted to Integer.

NOTE: *The enable options for the Modbus Network Write instruction. It is recommended to use the "Once on Leading Edge" option with this instruction when writing to the ViewMarq display. Subsequent actions (explained below) should be taken before sending another message.*

Load String into PLC Memory
Convert String into Integer Datatype
Convert String Length from bytes to words
$100ms
ST5

STRPRINT   Print to String
Print to   SL0
No space separator
Print Script "<ID 1><CLR><WIN 0 0 287 31><POS
0 0><JL><BL N><CS 0><AMB><T>Hello
</T><GRN><T>World</T>" "$0D"

STRGETB   Get Bytes Out of a String
Get out of String   SL0
Start at Index   0
Length in Bytes   SL0.Length
Into Starting Element   V0

MATH   Calculate Expression
Result   V1000
Expression (SL0.Length / 2) + 1

Write the string to ViewMarq Command String Buffer
C0

MWX   Modbus Network Write
Device   @IntSerModbusClient
Unit ID   1
Function Code   16 - Write Multiple Registers
      Modbus Address 4... + offset value entered below
To Modbus Offset Address   11000
Number of Modbus Registers   V1000
From Do-more Memory Address   V0
On Success, Set bit   C3
On Error, Set bit   C4

C0
RST

7

Reading the ASCII Reply from the ViewMarq (Optional)

After the Modbus Write is successful, the ViewMarq display will process the message. When the display has finished processing the Message it will clear the Command Block.

1) The next step of the logic should be to read the first element of Command Block until it reads 0.

a) Unit ID should be the same as the previous MWX instruction.

b) Function Code should be 3 – Read Holding Registers.

c) From Modbus Offset Address is still 11000 which is the Command Block of the display.

d) Number of Modbus Registers should be 1.

e) The Do-More Memory Address should be an available, unused memory location. This location should be checked after every read until it goes to 0.

2) After the first register of the Command Block returns a 0, the Status Block should be read.

a) Unit ID should be the same as the previous MWX and MRX instructions.

b) Function Code should be 3 – Read Holding Registers.

c) From Modbus Offset Address should be 11500 which is the location of the Status Block in the ViewMarq display.

d) Number of Modbus Registers should be 64 (128 bytes) which is the size of the Status Block area.

e) To Do-More Memory Address should be the starting location of 64 consecutive, available V memory locations.

3) The final step to make the Status reply more readable is to convert the integer block location of the MRX into a String using the STRPUTB instruction.

After this instruction has been enabled, the Status Reply should be readable in text form in the String location that was entered.

3) Once the string in the Status String has been read and converted, check the value of the string for the text "OK" using the String Compare instruction.
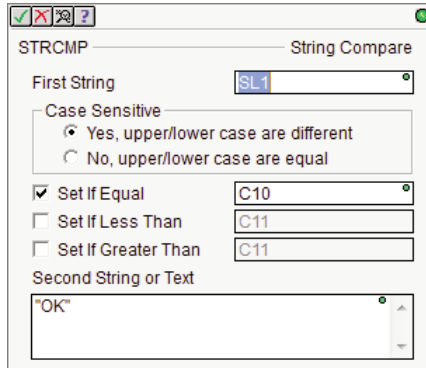
a) Select the Source String where the Status String was stored

b) Select for Case sensitive

c) Select "Set if Equal

d) Enter an available discrete tag for the flag

Example Do-more PLC code for sending an ASCII string over serial Modbus.

Write the string to ViewMarq Command String Buffer

```
C0
─┤↑├──────────────────────────────┐
                                   │  MWX                    Modbus Network Write
                                   │  Device                    @IntSerModbusClient
                                   │  Unit ID                                    1
                                   │  Function Code         16 - Write Multiple Registers
                                   │      Modbus Address 4... + offset value entered below
                                   │  To Modbus Offset Address                  11000
                                   │  Number of Modbus Registers                 V1000
                                   │  From Do-more Memory Address                   V0
                                   │  On Success, Set bit                          C3
                                   │  On Error, Set bit                            C4
                                   │
                                   │                    C0
                                   └──────────────────( RST )
```

```
                                                         WriteFinished
C3                                                          C8
─┤↑├──────────────────────────────────────────────────( SET )
```

Read the ViewMarq Command String Buffer

```
WriteFinished
C8
─┤ ├──────────────────────────────┐
                                   │  MRX                    Modbus Network Read
                                   │  Device                    @IntSerModbusClient
                                   │  Unit ID                                    1
                                   │  Function Code         3 - Read Holding Registers
                                   │      Modbus Address 4... + offset value entered below
                                   │  From Modbus Offset Address                11000
                                   │  Number of Modbus Registers                    1
                                   │  To Do-more Memory Address                  V400
                                   │  On Success, Set bit                          C6
                                   │  On Error, Set bit                            C7
                                   │
                                   │            WriteFinished
                                   │                C8
                                   └──────────────( RST )
```

Reading ViewMarq Command String successful

```
                                                      ProcessDoneCheck
C6                                                         C9
─┤ ├──────────────────────────────────────────────────( SET )
```

If ViewMarq Command String is not 0,
then read again in 1 second

```
ProcessDoneCheck
C9        V400  0
─┤ ├──────┤≠├──────────────────────────────┤E/R├  TMR                Timer

                                                   Timer Struct           T0
                                                   Preset             1.000s
```

Logic continued next page.

```
T0.Done                                                    WriteFinished
                                                               C8
 ┤ ├                                                      ( SET )

                                                         ProcessDoneCheck
                                                               C9
                                                          ( RST )
```

If the iewMarq Command String is equal to 0,
then read the ViewMarq Status Buffer

```
ProcessDoneCheck                                         ReadStatRegister
      C9          V400    0                                    C5
    ┤ ├         ┤  =  ├                                   ( OUT )

                                                         ProcessDoneCheck
                                                               C9
                                                          ( RST )
```

When the Command String Buffer equals 0,
read the ViewMarq Status String Buffer

```
ReadStatRegister
      C5        ┌─────────────────────────────────────────────────────────┐
    ┤↑├         │ MRX                              Modbus Network Read      │
                │                                                           │
                │ Device                              @IntSerModbusClient   │
                │ Unit ID                                               1   │
                │ Function Code                 3 - Read Holding Registers  │
                │          Modbus Address 4... + offset value entered below │
                │ From Modbus Offset Address                        11500   │
                │ Number of Modbus Registers                           64   │
                │ To Do-more Memory Address                          V200   │
                │  Do-more Range                             V200 - V263    │
                │ On Success, Set bit                                  C1   │
                │ On Error, Set bit                                    C2   │
                └───────────────────────────────────────────────────────────┘
```

Convert the Integer data to the String data type
to make it more readable

```
$On
ST1             ┌─────────────────────────────────────────────────────────┐
 ┤ ├            │ STRPUTB          Put Bytes Into a String                 │
                │                                                          │
                │ Put Into String                                   SL1    │
                │ Start at Index                                      0    │
                │ Length in Bytes                                   128    │
                │ From Starting Element                            V200    │
                └──────────────────────────────────────────────────────────┘
```

Check the Status String for "OK" reply

```
C1              ┌─────────────────────────────────────────────────────────┐
┤↑├             │ STRCMP                               String Compare      │
                │ First String                                     SL1     │
                │ Case Sensitive Yes, upper/lower case are different       │
                │ Set If Equal                                     C10      │
                │ Second String or Text  "OK"                              │
                └──────────────────────────────────────────────────────────┘
```

Modbus TCP

The same exact steps used above for sending Modbus Serial Strings should be used for sending Strings over Modbus TCP with only one difference in configuration of the MWX and MRX instructions.

Choose the "@IntModTCPClient" Device and enter in the IP address of the ViewMarq display. Leave the TCP Port Number as 502 and the Unit ID as 255. Everything else should be the same as the configuration for Modbus Serial.

Example Do-more PLC code for sending an ASCII string over Ethernet Modbus.

Write the string to ViewMarq Command String Buffer

	MWX	Modbus Network Write
C0	Device	@IntModTCPClient
	IP Address	192.168.1.109
	TCP Port Number	502
	Unit ID	255
	Function Code	16 - Write Multiple Registers
	Modbus Address 4...	+ offset value entered below
	To Modbus Offset Address	11000
	Number of Modbus Registers	V1000
	From Do-more Memory Address	V0
	On Success, Set bit	C3
	On Error, Set bit	C4

C0
RST

WriteFinished
C3
C8
SET

Read the ViewMarq Command String Buffer

WriteFinished
C8

	MRX	Modbus Network Read
	Device	@IntModTCPClient
	IP Address	192.168.1.109
	TCP Port Number	502
	Unit ID	255
	Function Code	3 - Read Holding Registers
	Modbus Address 4...	+ offset value entered below
	From Modbus Offset Address	11000
	Number of Modbus Registers	1
	To Do-more Memory Address	V400
	On Success, Set bit	C6
	On Error, Set bit	C7

WriteFinished
C8
RST

Reading ViewMarq Command String successful

ProcessDoneCheck
C6
C9
SET

If ViewMarq Command String is not 0
then read again in 1 second

ProcessDoneCheck
C9   V400   0
≠

TMR	Timer
E/R	
Timer Struct	T0
Preset	1.000s

Logic continued next page.

T0.Done

WriteFinished
C8
(SET)

ProcessDoneCheck
C9
(RST)

If the iewMarq Command String is equal to 0,
then read the ViewMarq Status Buffer

ProcessDoneCheck
C9   V400   0

ReadStatRegister
C5
(OUT)

ProcessDoneCheck
C9
(RST)

When the Command String Buffer equals 0,
read the ViewMarq Status String Buffer

ReadStatRegister
C5

MRX	Modbus Network Read
Device	@IntModTCPClient
IP Address	192.168.1.109
TCP Port Number	502
Unit ID	255
Function Code	3 - Read Holding Registers
	Modbus Address 4... + offset value entered below
From Modbus Offset Address	11500
Number of Modbus Registers	64
To Do-more Memory Address	V200
Do-more Range	V200 - V263
On Success, Set bit	C1
On Error, Set bit	C2

Convert the Integer data to the String data type
to make it more readable

$On
ST1

STRPUTB	Put Bytes Into a String
Put Into String	SL1
Start at Index	0
Length in Bytes	128
From Starting Element	V200

Check the Status String for "OK" reply
C1

STRCMP	String Compare
First String	SL1
Case Sensitive Yes, upper/lower case are different	
Set If Equal	C10
Second String or Text "OK"	

7

DirectLogic PLC by AutomationDirect

The Direct Logic PLC may communicate with the ViewMarq LED display by ASCII, Modbus RTU or Modbus TCP.

This section discusses:

Sending a Command String by ASCII over Serial

Sending a Command String by Modbus over Serial

Sending a Command String by Modbus TCP over Ethernet

Sending Strings from DirectLogic to the ViewMarq display by ASCII over Serial

1) There are two methods for sending ASCII strings to the ViewMarq over serial.

    a. PRINT

    b. PRINTV

2) First the PLC port must be configured for "Non-Seq(ASCII)" in order to allow ASCII strings to be sent.

    a. Go to PLC>Setup>Setup Secondary Comm Port:

    b. Set the Protocol to Non-Seq(ASCII)
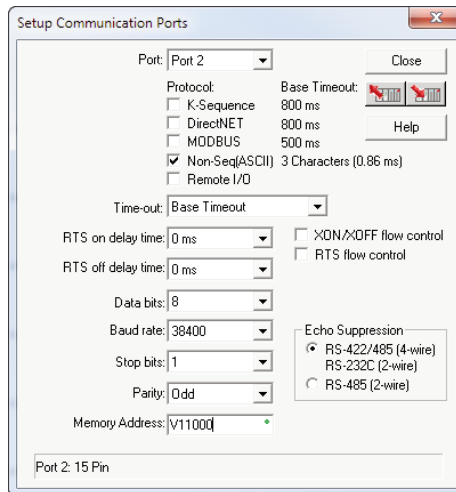
    c. Match the Data bits, Baud rate, Stop bits and Parity to the serial port settings of the ViewMarq display serial port. The Memory Address V memory location should be an available, unused block of registers

Using the DirectLogic PRINT instruction

1) In the ViewMarq software, type and configure a message. Copy the string in the Command String window by selecting Copy on the Command String Toolbar.
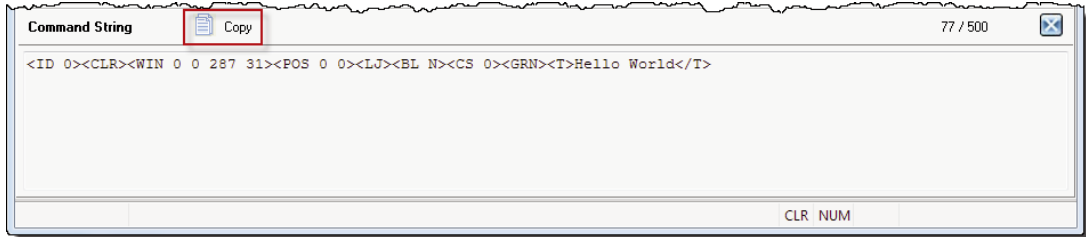
2) In the DirectSoft Programming Software paste (Ctrl-V) the ASCII String into the PRINT instruction as shown below.
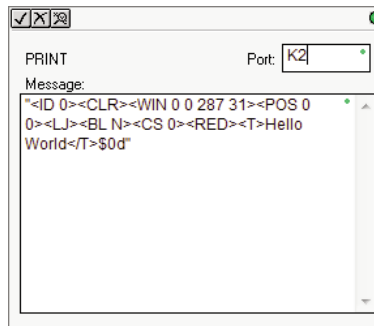
NOTE: *Quotation marks must be placed around the message that has been pasted into the PRINT instruction. You will also need to add termination character ("$0d") to the end of the message string.*

Example of DirectLogic PRINT instruction to send ASCII

NOTE: *Use a Store Positive Differential (STRPD) instruction when enabling a PRINTV instruction in order to send only 1 String of data out the port.*

The "ASCII Reply" option should be disabled when sending ASCII strings with the D2-260 or D0-06. Those PLCs only support one-way ASCII communications. See Chapter 6 - Configuring the ViewMarq LED Display for further details. If the application requires more reliable error detection and handshaking consider using Modbus communications instead

Using PRINTV instruction (D2-260 or D0-06 only)

You may also choose to use the PRINTV instruction to send the ASCII string to the ViewMarq. In order to use the PRINTV instruction, the ASCII string must first be embedded into the V-memory of the PLC.

Embedding the String into PLC memory

1) In the ViewMarq software, type and configure a message. Copy the string in the Command String window by selecting Copy on the Command String Toolbar.
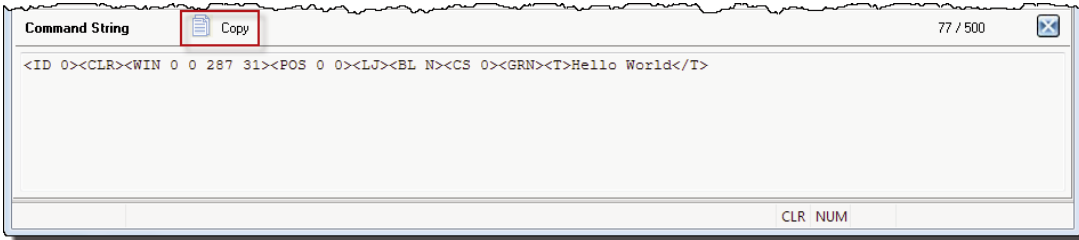
2) In the DirectSoft Programming Software Paste (Ctrl-V) the VPRINT Instruction as shown.
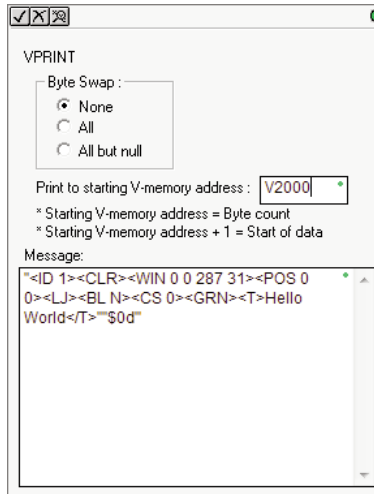
Attention!: Quotation marks must be placed around the message that has been pasted into the VPRINT instruction. You will also need to add a termination character ("$0d") to the end of the message string.

NOTE: For reference later, the V-memory location entered will be where the number of the characters entered into the Message field will be located. The actual ASCII data will start at the 2nd location of the V-memory block. For the "Print to starting V-memory address:" use any available unused block of V memory addresses.

3) Use the PRINTV instruction as shown below to choose the block of ASCII characters to send out the serial port.

a) Port Number should be set to K2 to indicate the HD-15 pin port (Port 2).

b) Start Address should be the second V-memory location of the V-memory block entered in the VPRINT instruction from earlier.

c) Number of Bytes should be the first V-memory location specified in the V-memory block entered in the VPRINT instruction earlier.

NOTE: *Use a Store Positive Differential (STRPD) instruction when enabling a PRINTV instruction in order to send only 1 String of data out the port.*

ATTENTION

Attention!: Command Strings should be sent at least 100 ms apart.

The "ASCII Reply" option should be disabled when sending ASCII strings with the D2-260 or D0-06. Those PLCs only support one-way ASCII communications. See Chapter 6 - Configuring the ViewMarq LED Display for further details. If the application requires more reliable error detection and handshaking consider using Modbus communications instead.

Sending Strings from DirectLogic to the ViewMarq display by Modbus over Serial

1) The serial port must be configured for Modbus in order to send Strings encapsulated within a Modbus RTU message to the display.

    a) Go to PLC>Setup>Setup Secondary Comm Port:
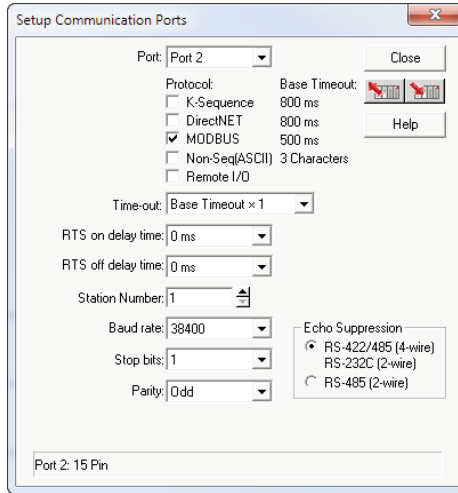
    b) Match the Data bits, Baud rate, Stop bits and Parity to the serial port settings of the ViewMarq display serial port.

Calculate the Word Count
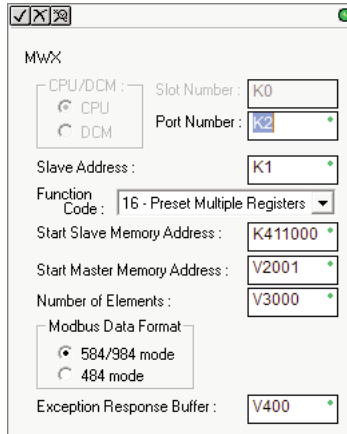
The ASCII string data was placed in V-memory as Byte Data earlier using the VPRINT instruction. In order to send it over Modbus the Word Count is needed.

1) Divide the byte count by 2 (2 bytes per word) and then add 1 in case the String length works out to an odd number of bytes so that we don't chop off 1 character of the String.
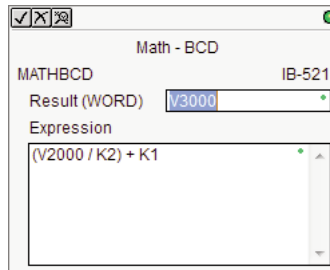
Continued on Next Page.

2) Now the data is prepared to be transported over Modbus and we have the calculated request size. Use a MWX instruction to write the data out the serial port.

a) The Port Number should be K2 to specify the secondary comm. port (Port 2) of the D2-260 or D0-06.

b) The Slave Address should match the Node Address setting of the serial port configuration in the LED Display Configuration.

c) The Function Code should be set to 16 – Preset Multiple Registers.

d) Start Slave Memory Address is the location of the Command Block within the ViewMarq display.

NOTE: *The DirectLogic MWX instruction uses the Modicon style addressing where the highest digit (4 in this case) is the Modbus data type indicator (Holding Registers).*

e) Start Master Memory Address should be the resulting memory location of the VPRINT instruction + 1 that contains the String data converted to Integer.

f) Number of Elements should contain the V memory location that was the result of our calculated Modbus request size from above.

g) Modbus Data Format should be 584/984 mode to match the example addressing shown above.

h) Exception Response Buffer should be an available, unused V-memory address. Note that this field uses 3 consecutive V-memory addresses.

Example DirectLogic PLC code for sending a string to ViewMarq over serial Modbus.

Embed String into PLC Memory
```
  _On
  SP1                                          VPRINT
───┤ ├──────────────────────────────             Byte Swap :                    None
                                                 "Print to" Address :           V2000
                                         "<ID 1><CLR><WIN 0 0 287 31><POS 0
                                         0><LJ><BL N><CS 0><AMB><T>Hello
                                         World 2<T>" "$0d" "$0a"
```

Calculate the Word Count
```
  _On
  SP1                                              Math - BCD
───┤ ├──────────────────────────────             MATHBCD                      IB-521

                                                 Result (WORD)                  V3000
                                                 Expression  (V2000 / K2) + K1
```

Write the String in PLC Memeory to the ViewMarq
```
  C0
───┤↑├──────────────────────────────    MWX
                                          CPU/DCM Slot :                        CPU
                                          Port Number :                          K2
                                          Slave Address :                        K1
                                          Function Code :  16 - Preset Multiple Registers
                                          Start Slave Memory Address :      K411000
                                          Start Master Memory Address :       V2001
                                          Number of Elements :                V3000
                                          Modbus Data type :            584/984 Mode
                                          Exception Response Buffer :          V400
```

Optional Error Checking

1) After the Modbus Write is successful, the ViewMarq display will process the message. When the display has finished processing the Message it will clear the Command Block. Use the MRX instruction to read the first element of Command Block until it reads 0.
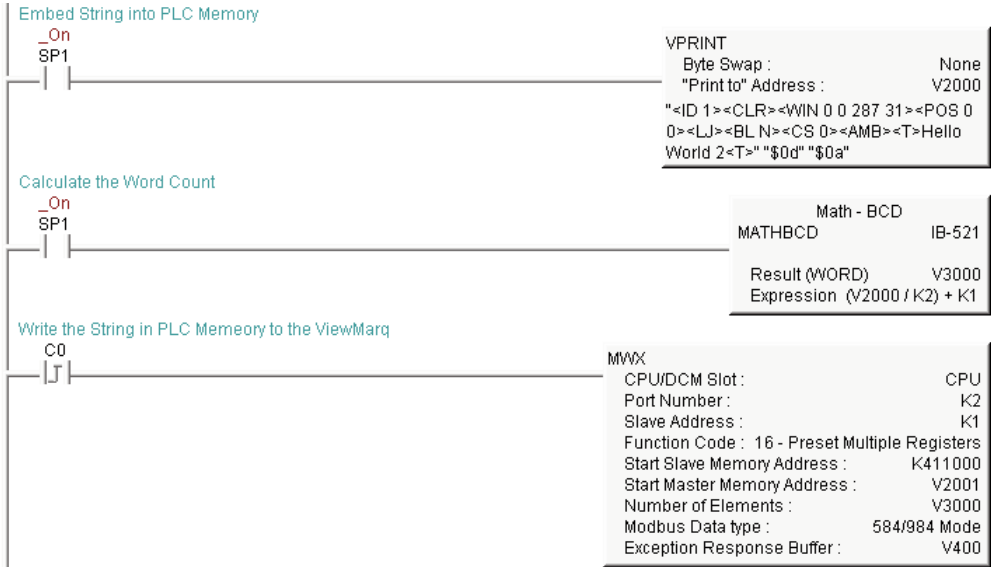

a) The Port Number should be K2 to specify the secondary comm. port (Port 2) of the D2-260 or the D0-06.

b) The Slave Address should match the Node Address setting of the serial port configuration in the LED Display Configuration.

c) The Function Code should be set to 03 – Read Holding Registers.

d) Start Slave Memory Address is the location of the Command Block within the ViewMarq display.

NOTE: *The DirectLogic MWX instruction uses the Modicon style addressing where the highest digit (4 in this case) is the Modbus data type indicator (Holding Registers).*

e) Start Master Memory Address should be an available, unused V-memory location that can be used to compare to 0. The PLC should keep reading until this V-memory location indicates a 0.

f) Number of Elements should be 1 as we are reading only the first register of the Command Block.

g) Modbus Data Format should be 584/984 mode to match the example addressing shown above.

h) Exception Response Buffer should be an available, unused V-memory address. Note that this field uses 3 consecutive V-memory addresses.

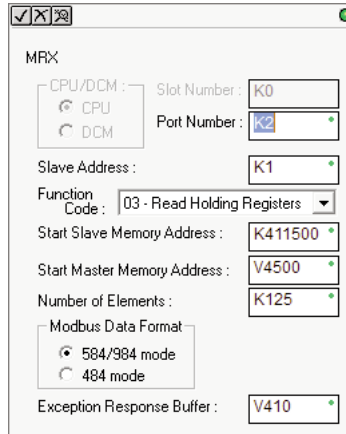2) After the first register of the Command Block returns a 0, use the MRX instruction again to read the Status Block.

a) The Port Number should be K2 to specify the secondary comm. Port of the D2-260 or the D0-06.

b) The Slave Address should match the Node Address setting of the serial port configuration in the LED Display Configuration.

c) The Function Code should be set to 03 – Read Holding Registers.

d) Start Slave Memory Address is the location of the Status Block within the ViewMarq display.

> **NOTE:** The DirectLogic MWX instruction uses the Modicon style addressing where the highest digit (4 in this case) is the Modbus data type indicator (Holding Registers).

e) Start Master Memory Address should be an available, unused block of V-memory locations.

f) Number of Elements should be 125.

g) Modbus Data Format should be 584/984 mode to match the example addressing shown above.

h) Exception Response Buffer should be an available, unused V-memory address. Note that this field uses 3 consecutive V-memory addresses.

To view the Status string in clear text, enter in the starting V-memory location of the last MRX instruction and change the view to "Text" and the size to 40 in Data View.

> **NOTE:** If the response was an error and the error text is longer than 40 characters, you will have to enter in a V-memory location further into the block and change the view for that location to text as well.

Example DirectLogic PLC code for reading the Status Block in the ViewMarq.

Read the ViewMarq Commnd Block until it is equal to 0

```
V4000      K0        SP116                              MRX
 ─|≠|──────|/|────────────────────────────────         CPU/DCM Slot :                    CPU
                                                        Port Number :                      K2
                                                        Slave Address :                    K1
                                                        Function Code :   03 - Read Holding Registers
                                                        Start Slave Memory Address :   K411000
                                                        Start Master Memory Address :     V4000
                                                        Number of Elements :               K1
                                                        Modbus Data type :          584/984 Mode
                                                        Exception Response Buffer :       V405

  C0                                                             C0
 ─| |──                                                        ─( RST )─

                                                                C1
                                                              ─( SET )─
```

When the ViewMarq Command Block is equal to 0,
read the ViewMarq Status Block

```
V4000      K0      C1      SP116                        MRX
 ─| = |────|| |────|/|──────────────────────           CPU/DCM Slot :                    CPU
                                                        Port Number :                      K2
                                                        Slave Address :                    K1
                                                        Function Code :   03 - Read Holding Registers
                                                        Start Slave Memory Address :   K411500
                                                        Start Master Memory Address :     V4500
                                                        Number of Elements :             K125
                                                        Modbus Data type :          584/984 Mode
                                                        Exception Response Buffer :       V410

                                                                C1
                                                              ─( RST )─

                                                                C2
                                                              ─( SET )─
```

Set a Flag if Status is "OK"

```
V410      K4b4f      C2                                                            C3
 ─| = |────|| |──────────────────────────────────────────────────────────────  ─( SET )─

                                                                                   C2
                                                                                 ─( RST )─
```

Sending Strings from DirectLogic to the ViewMarq display by Modbus over Ethernet

Sending messages from the PLC over Modbus TCP requires that an H2-ECOM100 module is used with the D2-260 and a H0-ECOM100 module with the D0-06.

The same steps used above for sending Modbus Serial Strings should be used for sending Strings over Modbus TCP but the Modbus instructions will be quite different.

Configuring the ECOM100 Module

Before the instructions are entered, the ECOM100 module must first be configured for a Modbus TCP message. The NetEdit tool is required for this and can be downloaded for free from *www.automationdirect.com*.

1) Start the NetEdit software after it has been downloaded and installed on the PC. It will automatically scan the network attached to the PC and bring up the devices in a window that looks like the window below.
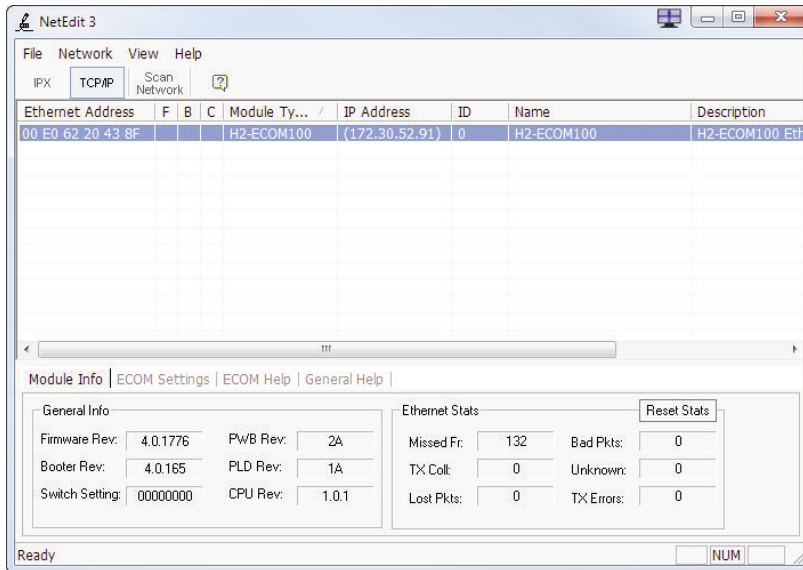
2) Click on the "ECOM Settings" tab at the bottom and then click on the "Peer to Peer Config…" button.
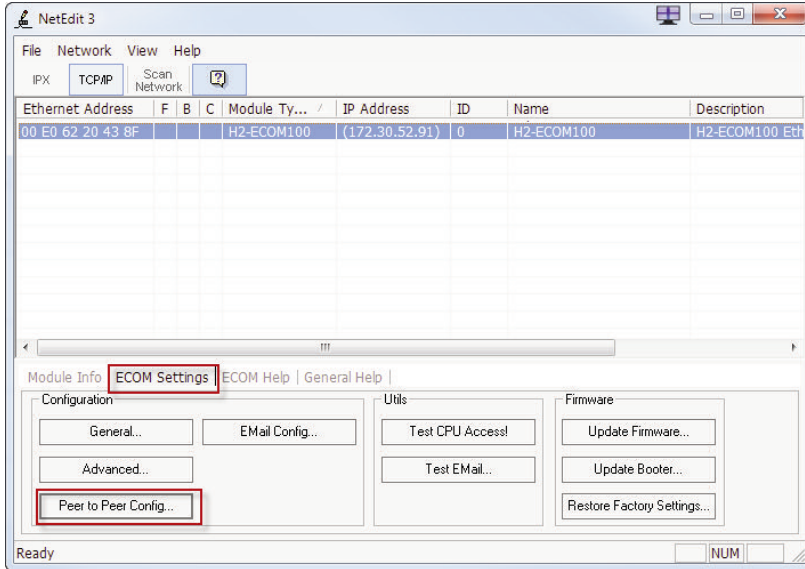
3) In the Peer to Peer Configuration window, click on the "Add" button to the right.
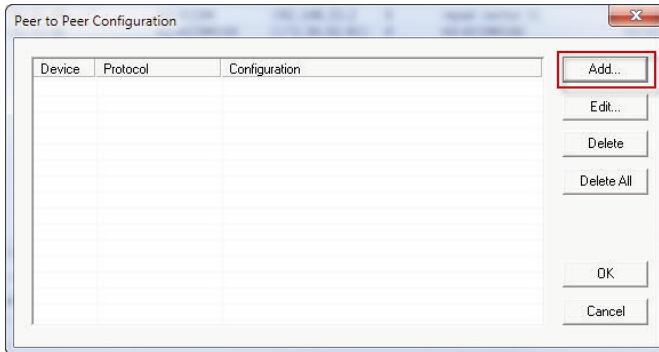
4) Fill in the Add Device Address window as shown below but using the IP address of your ViewMarq display.

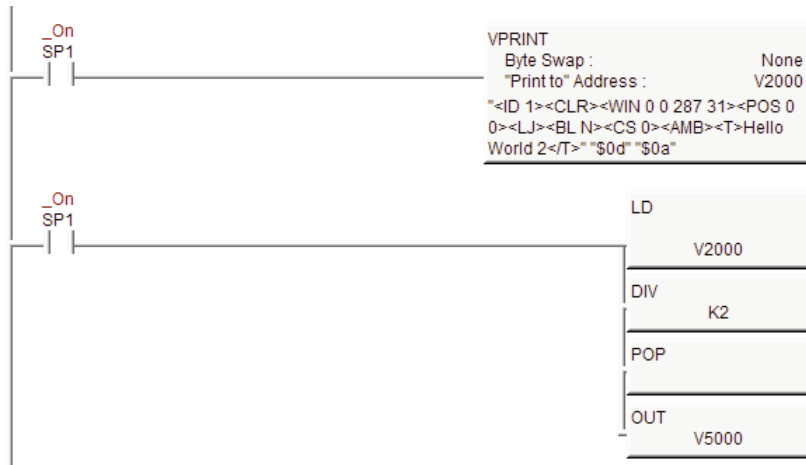**NOTE:** *The RX/WX Device Number will be needed in your Modbus TCP instruction setup.*

5) Click the Ok button to send this configuration to the ECOM100 module.
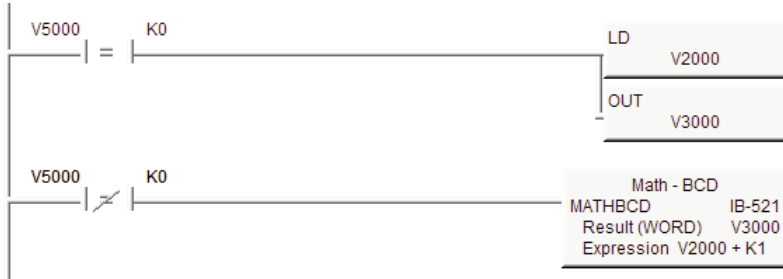
Calculating the Byte Count

To send a Modbus TCP message, you must use the RX and WX instructions. It is not just one instruction but a series of 4 instructions that work together. Here we focus primarily on how these instructions are used with the ViewMarq display. To find more details on the usage of the RX and WX instruction when doing Modbus TCP, consult the ECOM100 manual.

To send a Modbus TCP write message, we must first adjust the size we are writing. The WX instruction uses a byte count for sending data but it must be an even count.

1) The first thing to do is to divide the byte count (V2000 in this case) by 2 and check to see if there is a remainder. The DIV instruction will place the remainder (if there is one) into the second stack location. In order to retrieve that value, we must use a POP instruction and then an OUT to move the value to a V-memory location.

```
 _On
 SP1                                          VPRINT
                                               Byte Swap :                    None
──┤ ├──────────────────────────────            "Print to" Address :          V2000
                                              "<ID 1><CLR><WIN 0 0 287 31><POS 0
                                              0><LJ><BL N><CS 0><AMB><T>Hello
                                              World 2</T>" "$0d" "$0a"


 _On
 SP1                                                         LD
                                                                     V2000
──┤ ├──────────────────────────────────────────────────
                                                            DIV
                                                                     K2

                                                            POP

                                                            OUT
                                                                     V5000
```

7

2) Now that we know where the remainder is, we can do a simple comparison to see if there is a remainder or not. If there is no remainder we simply move the count to the V-memory location that will be our Modbus TCP write size. If there is a remainder, we simply add 1 to the original count to make it even and place the result in the V-memory location that will be our Modbus TCP write size.

```
V5000      K0                                    LD
 ──| = |────────────────────────────              V2000

                                                OUT
                                                  V3000

V5000      K0                                   Math - BCD
 ──| ≠ |────────────────────────              MATHBCD         IB-521
                                              Result (WORD)   V3000
                                              Expression V2000 + K1
```

7

Sending the String

1) Now that the byte count is adjusted, send the string to the ViewMarq using the WX instruction. The logic for sending the first Modbus TCP write is as follows:

```
C0                                              LD
 ──| ↑ |───────────────────────────────           K201

                                                LD
                                                  V3000

                                                LDA
                                                  O2001

                                                WX
                                                  V25367
```

a) The first LD instruction specifies the slot number that the ECOM100 is located in and the RX/WX Device number that was configured earlier in NetEdit. The upper byte specifies the Slot (2 in the example above). The lower byte specifies the RX/WX Device number (01 in the example above).

b) The second LD instruction specifies the size, in bytes, that will be sent. V3000 is the result of the calculation explained above.

c) The LDA instruction specifies where the data will be written from. In this case, V2001 is where the String data resides from the VPRINT instruction configured earlier.

d) The WX instruction specifies the location of the Command Block within the ViewMarq display.

When doing Modbus TCP, the V-memory location specified is an alias of the Modbus address and has to be converted. V-memory locations in the DirectLogic PLCs are in octal format.

To convert the Modbus address to the alias address in DirectLogic, you take the desired Modbus address in 6 digit format (example 411000), subtract 400001 and then convert the value to Octal. So 411000 – 400001 = 10999 Convert to Octal = 25367. (See http://support.automationdirect.com/technotes.html for the Modbus Conversion Utility AN-MISC-010).
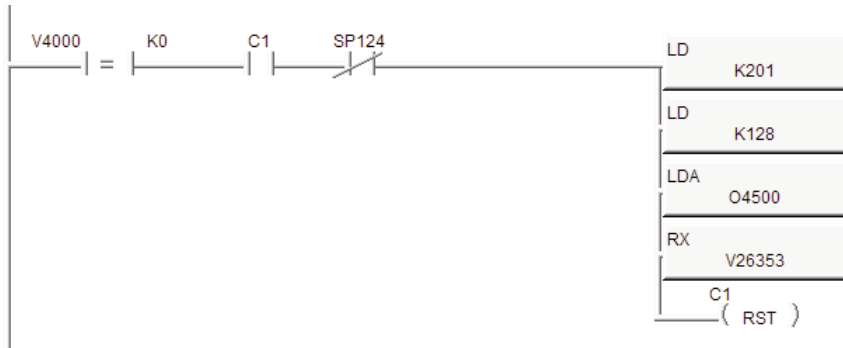
2) After the Modbus Write is successful, the ViewMarq display will process the message. When the display has finished processing the Message it will clear the Command Block. Use the RX instruction to read the first element of Command Block until it reads 0.
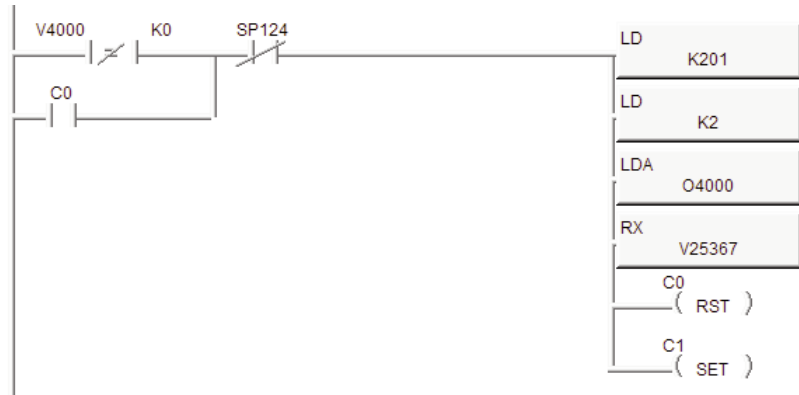
a) The first LD instruction specifies the slot number that the ECOM100 is located in and the RX/WX Device number that was configured earlier in NetEdit. The upper byte specifies the Slot (2 in the example above). The lower byte specifies the RX/WX Device number.

b) The second LD instruction specifies the size, in bytes, that will be read. K2 specifies 1 Modbus element, which is the first register of the Command Block.

c) The LDA instruction specifies where the data will be written into. In this case, we chose an available V-memory location that we can use to check for 0 data.

e) The instruction specifies the location of the Command Block within the ViewMarq display.

When doing Modbus TCP, the V-memory location specified is an alias of the Modbus address and has to be converted. V-memory locations in the DirectLogic PLCs are in octal format. To convert the Modbus address to the alias address in DirectLogic, you take the desired Modbus address in 6 digit format (example 411000), subtract 400001 and then convert the value to Octal. So 411000 – 400001 = 10999 Convert to Octal = 25367.

3) After the first register of the Command Block returns a 0, use the RX instruction again read the Status Block.

```
   V4000      K0        C1       SP124              LD
  ──| = |──┤   ├──┤ ├──┤/├─────────────────────        K201

                                                   LD
                                                       K128

                                                   LDA
                                                       O4500

                                                   RX
                                                       V26353

                                                       C1
                                                   ──( RST )
```

a) The first LD instruction specifies the slot number that the ECOM100 is located in and the RX/WX Device number that was configured earlier in NetEdit. The upper byte specifies the Slot (2 in the example above). The lower byte specifies the RX/WX Device number.

b) The second LD instruction specifies the size, in bytes, that will be read. K128 specifies 64 Modbus elements.

c) The LDA instruction specifies where the data will be written into. In this case, we chose an available V-memory block that we can place the Status data into.

d) The RX instruction specifies the location of the Status Block within the ViewMarq display.

When doing Modbus TCP, the V-memory location specified is an alias of the Modbus address and has to be converted. V-memory locations in the DirectLogic PLCs are in octal format. To convert the Modbus address to the alias address in DirectLogic, you take the desired Modbus address in 6 digit format (example 411500 which is where the Status data is located), subtract 400001 and then convert the value to Octal. So 411500 – 400001 = 11499 Convert to Octal = 26353.

PLC Cabling Chart

ViewMarq LED message displays can communicate with any master serial ASCII device, serial Modbus device or Modbus TCP controller. This chart below is a reference for communicating from AutomationDirect PLC's to the ViewMarq LED message displays.

Cable	Network	AutomationDirect Controller			ViewMarq Port
		Family	Model	Port	
Cat 5e	Ethernet	Productivity3000	P3-550	Ethernet (Modbus TCP)	RJ-45
		Do-more	H2-DM1E		
			H2-ECOM-100		
		DirectLogic	H2-ECOM-100		
D0-CBL	RS-232	CLICK	C0-00	RJ-12 Serial Port	Port 1 RJ-12
			C0-01		
			C0-02		
		Productivity3000	P3-530		
			P3-550		
		Do-more	H2-DM1		
			H2-DM1E		
		DirectLogic	DL05		
			DL06		
			DL105		
			DL205		
			H2-SERIO		
			H2-SERIO-4		
			D3-350		
			D4-450		
DO-CBL + FA-HD15	RS-232	DirectLogic	D0-DCM	D-SUB 15HD Serial Port	Port 1 RJ-12
			DL06		
			D2-250-1		
			D2-260		
D2-DSCBL-2	RS-485	DirectLogic	D0-DCM	D-SUB 15HD Serial Port	Port 1 Terminal
			DL06		
			D2-250-1		
			D2-260		
DO-CBL + FA-CABKIT	RS-232	DirectLogic	DL-405	D-SUB 15HD Serial Port	Port 1 RJ-12
Cable L19773	RS-485	CLICK	C0-02	Terminal Block Serial Port	Port 2 Terminal Block
		Productivity3000	P3-530		
			P3-550		
		Do-more	H2-SERIO		
			H2-SERIO-4		
		DirectLogic	H2-SERIO		
			H2-SERIO-4		

AutomationDirect PLC to ViewMarq Cabling

Allen Bradley MicroLogix and SLC PLCs

Depending on the PLC model*, the Allen Bradley MicroLogix and SLC series PLC's can utilize RS-232 with either ASCII or Modbus RTU to send ASCII strings to the ViewMarq LED Display.

RS-232

Connect the ViewMarq Port 1 terminal block to the Allen Bradley 8-pin Mini Din port or the 9-pin D-Sub port. The cables for this will need to be constructed by the user.


Allen-Bradley MicroLogix™ 1000/1100/1200/1400/1500 RS-232C

To AB MicroLogix RS-232 communication channel

To ViewMarq Panel Port1 RS-232

Mini Din 8-pin Male

1 = do not use
2 = Signal GND
3 = do not use
4 = RXD
5 = do not use
6 = do not use
7 = TXD
8 = do not use

0.35" DIA.   0.290"

Nonstandard keying

Wiring Diagram

GND 2 —— 3 ⏚
RXD 4 —— 4 TX
TXD 7 —— 5 RX

shield

1 = RS-485 +
2 = RS-485 -
3 = Signal Ground
4 = RS-232 TX
5 = RS-232 RX
6 = RS-232 RTS

Note: Use the above wiring diagram to make your own cable. We recommend using 22 AWG shielded cable.

Allen-Bradley SLC 5-03/04/05, ControlLogix, CompactLogix, FlexLogix, MicroLogix, DF1, RS-232C

To PLC 9-Pin Port

To ViewMarq Panel Port1 RS-232

9-pin D-sub (female)

1 = do not use
2 = RXD
3 = TXD
4 = do not use
5 = Signal GND

6 = do not use
7 = do not use
8 = do not use
9 = do not use

Wiring Diagram

RXD 2 —— 3 ⏚
TXD 3 —— 4 TX
GND 5 —— 5 RX

shield

1 = RS-485 +
2 = RS-485 -
3 = Signal Ground
4 = RS-232 TX
5 = RS-232 RX
6 = RS-232 RTS

Note: Use the above wiring diagram to make your own cable. We recommend using 22 AWG shielded cable.

*NOTE: Consult Allen Bradley user documentation to determine which models support ASCII control and / or Modbus instructions.

ASCII

In the LED Display Configuration window configure the ViewMarq RS-232 port to communicate using ASCII.
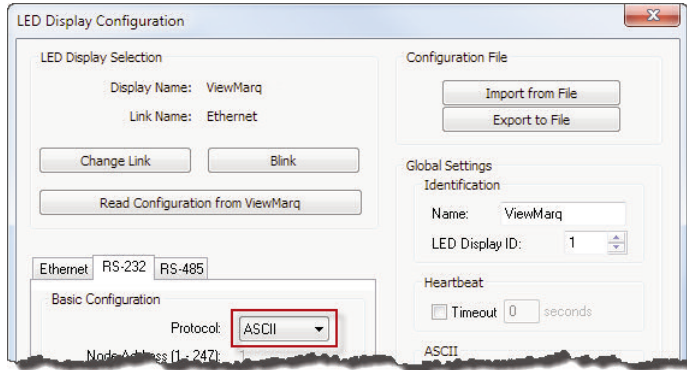
Configure the Allen Bradley PLC port to communicate ASCII with the ViewMarq LED Display.

1) In RSLogix500, open the Channel Configuration for the PLC...
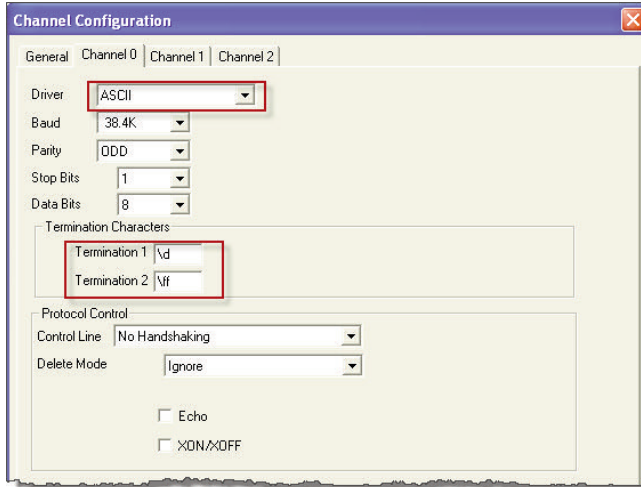
2) For the Channel being used on the PLC (Channel 0 shown above)...

    a. Set the Driver to ASCII

    b. Match the Baud Rate, Parity, Stop Bits and Data Bits to match those of Port 1 of the ViewMarq LED Display

    c. If the ASCII Reply from the ViewMarq will be read, set the Termination Characters as shown.

i. \d is a Carriage Return

ii. \ff means not used

3) On the General Tab, for the Channel configured in Step 2...

    a. Set the AWA Append Characters as shown below.

        i. \d is a Carriage Return

        ii. \ff means not used
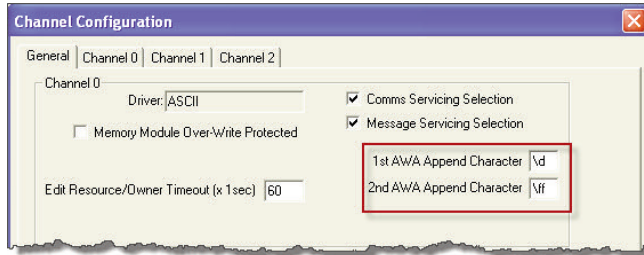
In order for the PLC to send the String it must be in the PLC Memory as a String Data Type.

1) Copy the ASCII String from the ViewMarq Software.
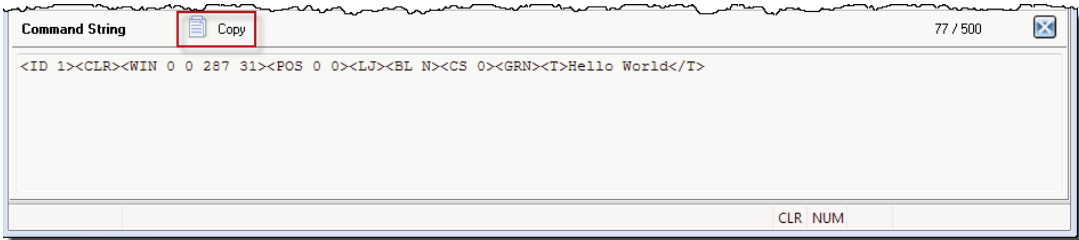
2) In RSLogix500, open the String data file you are using and paste the ASCII String into the file.
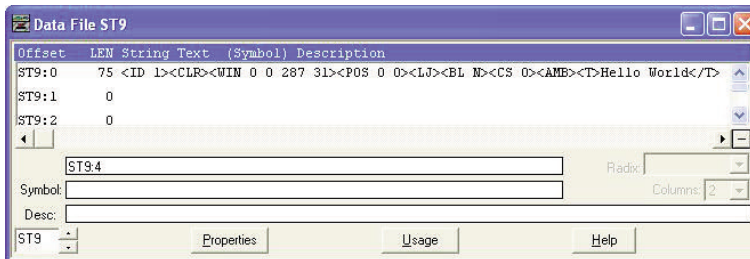
Now that the Port has been configured and the ASCII String is in PLC memory, use an AWA (ASCII Write Append) instruction to send the string out the serial port.

```
┌──── AWA ─────────────────┐
│ ASCII Write Append        │
─┤ Channel              0    │
│ Source            ST9:0    │
│ Control            R6:0    │
│ String Length       82<    │
│ Characters Sent      76    │
│ Error               00h    │
└───────────────────────────┘
```

1) Select the Channel number that was configured earlier and that is connected to the ViewMarq.

2) For the Source select the String Data Address the ASCII String was Pasted into.

3) Choose an unused Control Address.

4) Enter 82 for the String Length Maximum. For strings longer than 82 characters, see the next section, "Long ASCII Strings".

NOTE: *The AWA instruction is a "Level-enabled" instruction so be sure to put a One Shot (ONS) instruction for the enable.*
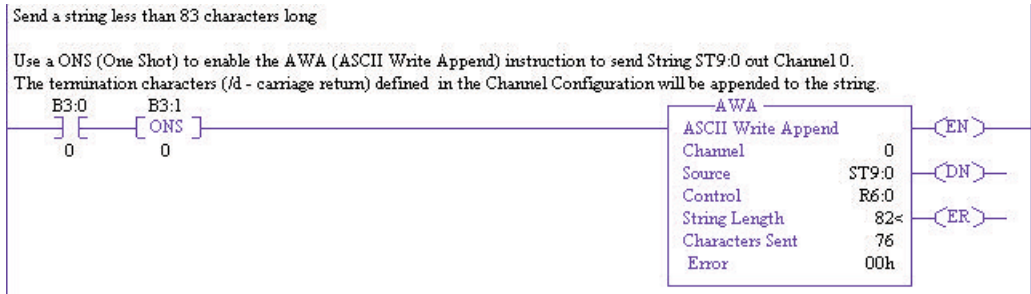
Send a string less than 83 characters long

Use a ONS (One Shot) to enable the AWA (ASCII Write Append) instruction to send String ST9:0 out Channel 0.
The termination characters (/d - carriage return) defined in the Channel Configuration will be appended to the string.

```
  B3:0      B3:1                                    ┌──── AWA ─────────────────┐
──┤ ├──────┤ONS├────────────────────────────────┤ ASCII Write Append        │──(EN)──
   0         0                                    │ Channel              0    │
                                                  │ Source            ST9:0    │──(DN)──
                                                  │ Control            R6:0    │
                                                  │ String Length       82<    │──(ER)──
                                                  │ Characters Sent      76    │
                                                  │ Error               00h    │
                                                  └───────────────────────────┘
```

Reading the ASCII Reply

If the "ASCII Reply" option is enabled in the "LED Display Configuration" settings AND you are sending to an ID other than 0, your PLC may Read the Reply Response String that is sent by the ViewMarq LED Display.

Use an ABL (ASCII Test For Line) instruction to check for the Termination characters entered in the Channel configuration and then a ARL (ASCII Read Line) instruction to read in the Status reply response String.

1) For the ABL instruction, select the Channel number that was configured earlier and that is connected to the ViewMarq.

2) Choose an unused Control Address.

3) Trigger the ABL instruction with a normally closed contact using the Control Address Enable to trigger the instruction each PLC scan.

4) For the ARL instruction, select the Channel number that was configured earlier and that is connected to the ViewMarq.

5) Choose an unused Control Address.

6) Trigger the ARL instruction with the Found (FD) bit of the ABL instruction Control Address.
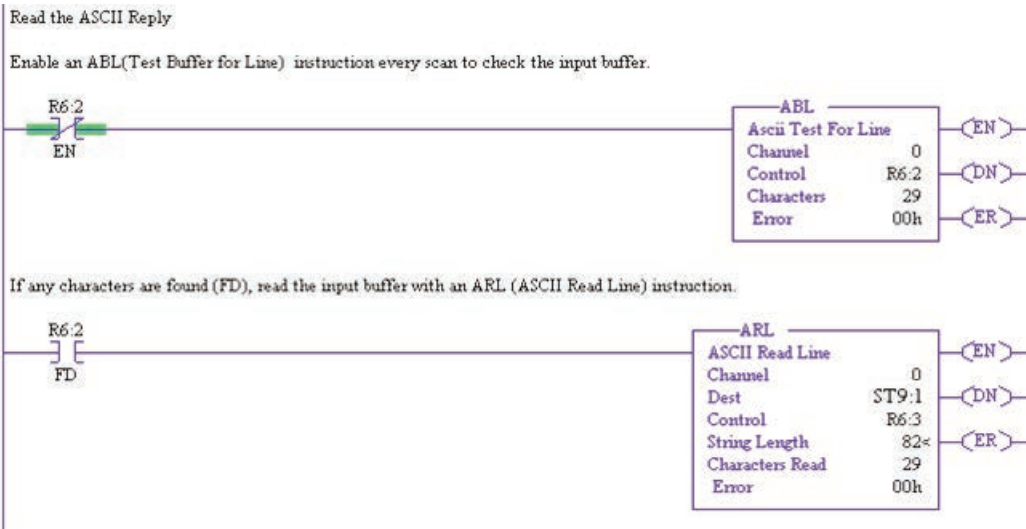
Read the ASCII Reply

Enable an ABL(Test Buffer for Line) instruction every scan to check the input buffer.

```
R6:2                                          ──ABL────────────
─┤/├──────────────────────────────────────   Ascii Test For Line      ─(EN)─
 EN                                           Channel          0
                                              Control       R6:2        ─(DN)─
                                              Characters      29
                                              Error          00h        ─(ER)─
```

If any characters are found (FD), read the input buffer with an ARL (ASCII Read Line) instruction.

```
R6:2                                          ──ARL────────────
─┤ ├──────────────────────────────────────   ASCII Read Line         ─(EN)─
 FD                                           Channel          0
                                              Dest          ST9:1       ─(DN)─
                                              Control       R6:3
                                              String Length   82<       ─(ER)─
                                              Characters Read  29
                                              Error          00h
```

If the syntax sent to the ViewMarq display was correct, you will see an "OK" in your Response String. If there is a problem, you will get a Syntax error and the position where the error occurred.
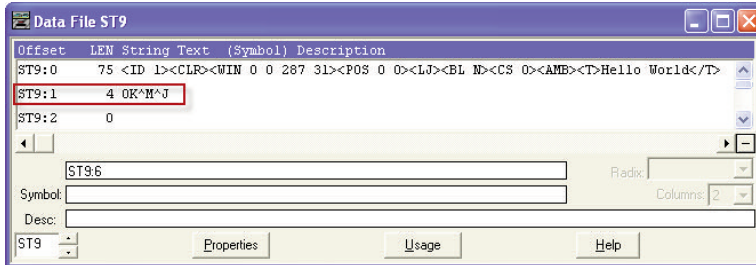
Example of a good String
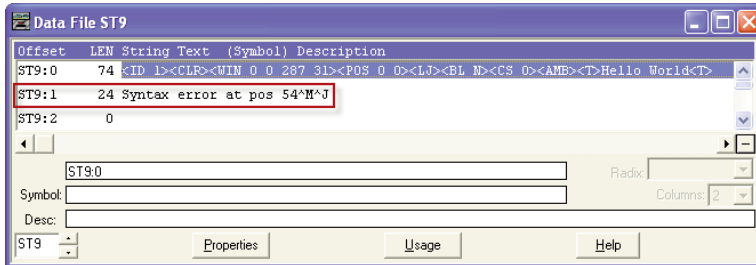
Example of a syntax error String

Sending Long ASCII Strings

If the ASCII String is longer than 82 characters, multiple String Files and multiple instructions need to be used.

> 1) Copy the ASCII String from the ViewMarq software.
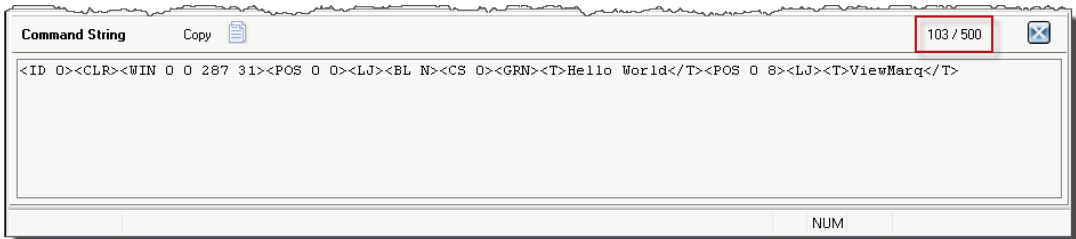
> 2) Paste up to 82 characters of the String into the first String File.
>
> 3) Paste the up to 82 of the next characters into the next available String File and so on until the entire string is contained in memory.

NOTE: *It does not matter where the string is broken apart. It will be assembled again in the ViewMarq display when a Carriage Return character is received.*
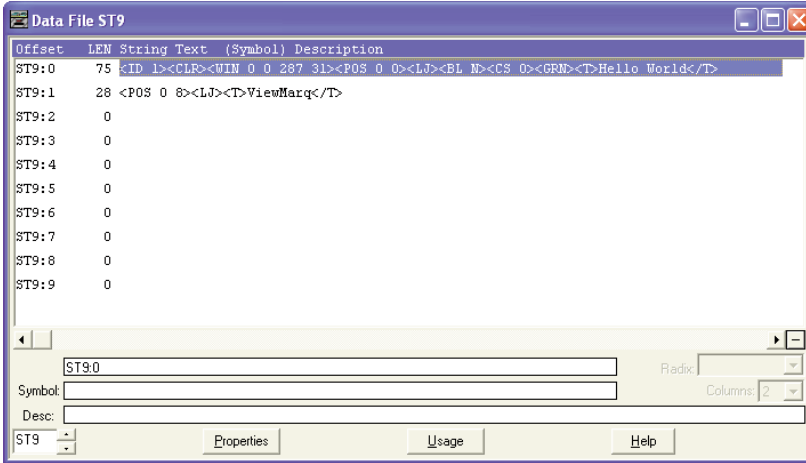
The AWA instruction will always "append" the termination character to the end of the string. To get around this, use the AWT (ASCII Write) Instruction as shown below. The AWT instruction sends the ASCII String out the Serial port **WITHOUT** the termination character.

Use as many instructions necessary to send each String File triggering the instruction with the Done (DN) bit of the previous instruction. The LAST instruction must be an AWA instruction to send the termination character.
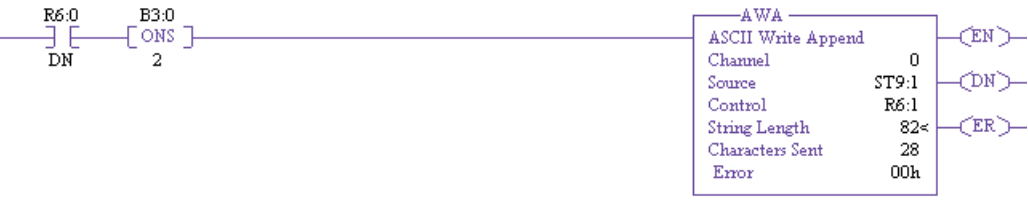
Modbus

Writing the ASCII String to the Command Block

In the LED Display Configuration window configure the ViewMarq RS-232 port.

1) Set the Protocol to Modbus.

2) Set the LED Display ID to the desired Modbus Node number.

3) Select Byte Swap because Allen Bradley Modbus is the opposite Byte order from ViewMarq Modbus.
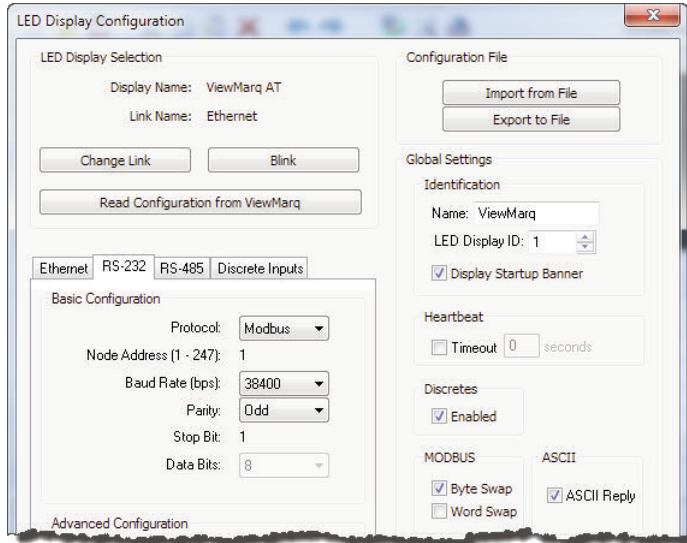
Configure the Allen Bradley PLC port to communicate Modbus with the ViewMarq LED display.

    1) In RSLogix500, open the Channel Configuration for the PLC.
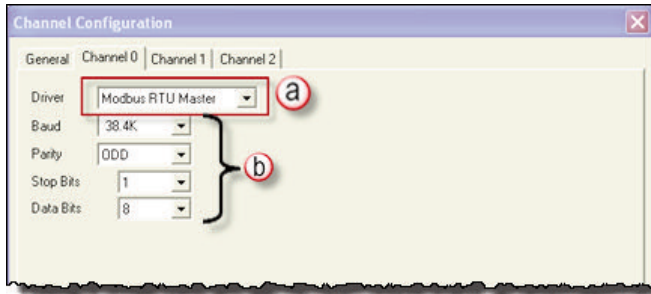
    2) For the Channel being used on the PLC (Channel 0 shown above)

        a. Set the Driver to Modbus.

        b. Match the Baud Rate, Parity, Stop Bits and Data Bits to match those of Port 1 of the ViewMarq LED display.

In order for the PLC to send the String it must be in the PLC Memory as an Integer Data Type.

    3) Copy the ASCII String from the ViewMarq Software.

| Command String | Copy 📄 | | 103 / 500 | ✕ |

`<ID 0><CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N><CS 0><GRN><T>Hello World</T><POS 0 8><LJ><T>ViewMarq</T>`

| | NUM | |
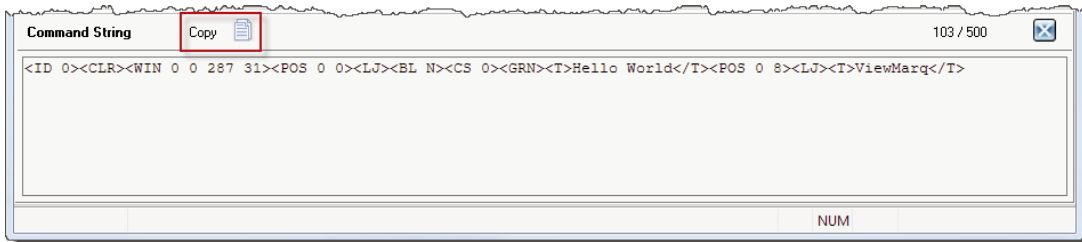
4) In RSLogix500 open the Integer data file you are using and paste the ASCII String into the file. Open the Integer data file and change the Radix to "ASCII".
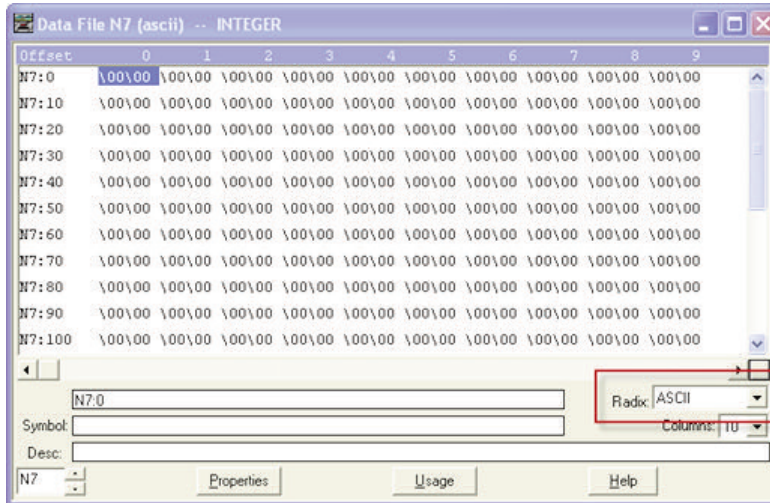
5) Double click on the first Integer file location and Paste the data in and hit Enter. It should look something like the window below.
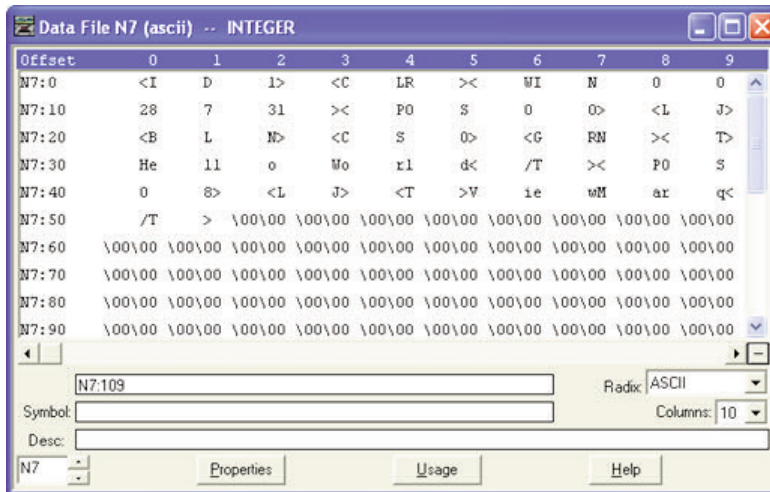
6) An additional byte (termination) must be added to the end of the data. If the String data is an odd number of bytes, the two termination characters will have to be separated between 2 different Integer addresses. Double click on the first field and add "\d" after the ">".
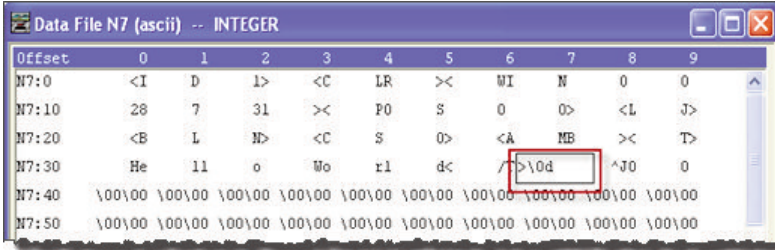
Offset	0	1	2	3	4	5	6	7	8	9
N7:0	<I	D	1>	<C	LR	><	WI	N	0	0
N7:10	28	7	31	><	P0	S	0	0>	<L	J>
N7:20	<B	L	N>	<C	S	0>	<A	MB	><	T>
N7:30	He	11	o	Wo	rl	d<	/T	>\0d	^J0	0
N7:40	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00
N7:50	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00

If the String data is an even number of bytes, enter in "\0d\00" in the next Integer address after the String data.

NOTE: Please note that when you hit enter, the display changes the \0d to ^M.

Offset	0	1	2	3	4	5	6	7	8	9
N7:0	<I	D	1>	<C	LR	><	WI	N	0	0
N7:10	28	7	31	><	P0	S	0	0>	<L	J>
N7:20	<B	L	N>	<C	S	0>	<G	RN	><	T>
N7:30	He	11	o	Wo	rl	d<	/T	><	P0	S
N7:40	0	8>	<L	J>	<T	>V	ie	wM	ar	q<
N7:50	/T	>^M	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00
N7:60	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00
N7:70	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00	\00\00
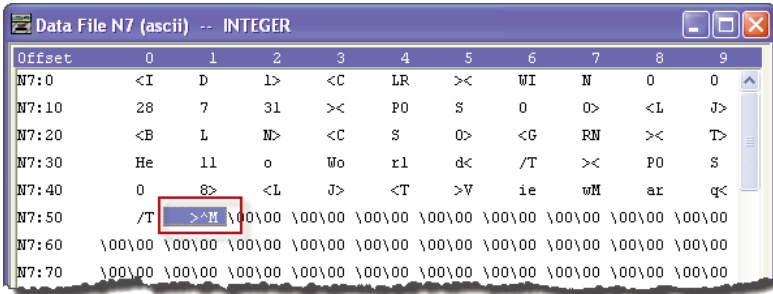
Now that the data is prepared, use a MSG (Read/Write Message) instruction to send the data out the port to the ViewMarq Display.

```
 ─── MSG ───
 Read/Write Message
 MSG File      MG10:0
        Setup Screen   <
```

1) Choose an available, unused MG address.

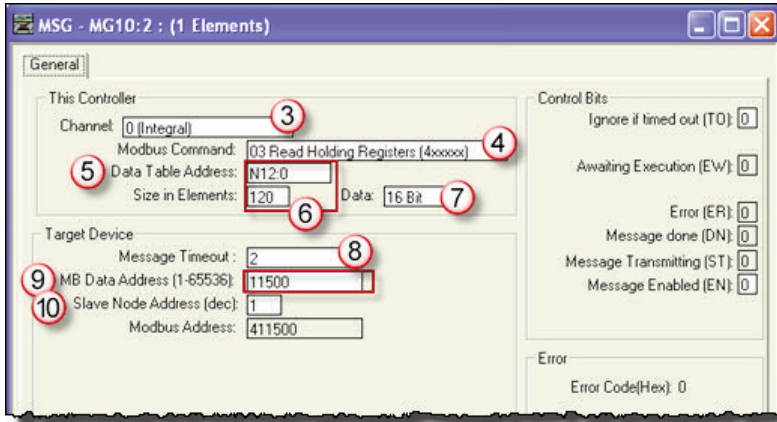2) Click on the "Setup Screen" option to configure the instruction.

3) Select the Channel number that was configured earlier and that is connected to the ViewMarq.

4) Select Modbus Command: 16 Write Multiple Registers (4xxxxx)

5) In the Data Table Address field, enter the starting address of the Integer data file that the copy and pasted data was placed into.

6) The Size in Elements field should be sufficient in size to include all of the populated Integer addresses. This can be up to 120 registers in length. . For strings longer than 240 characters, see the next section, "Long ASCII Strings".

7) Choose "16 Bit" for the Data field.

8) Leave the Message Timeout at 2. This may need to be increased if the cable quality is suspect or if the environmental conditions are less than ideal.

9) The MB Data Address should be 11000 for the Command Block. This equates to 411000 in Modicon Address style.

10) The Slave Node Address should match the LED Display ID (Node Address) configured in the LED Display for the port you are connecting to.

Write the ASCII String stored starting at N7:0 to the ViewMarq ASCII Command Block (Modbus address 411000)
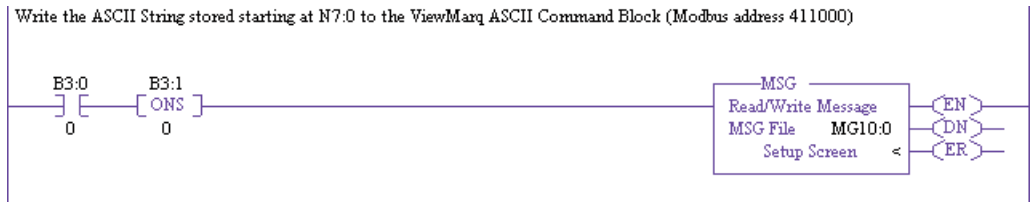
Reading the Status Reply Buffer

After the Modbus Write is successful, the ViewMarq display will process the message. When the display has finished processing the message it will clear the Command Block. The next step of the logic should be to read the 1st element of Command Block until it is 0.

Use another MSG (Read/Write Message) instruction to read the Command Block in the ViewMarq Display.

1) Choose an available, unused MG address.
2) Click on the "Setup Screen" option to configure the instruction.
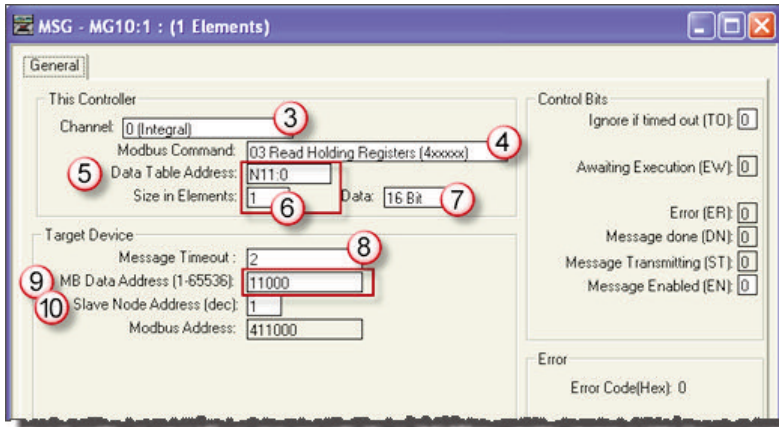
3) Select the Channel number that was configured earlier and that is connected to the ViewMarq
4) Select Modbus Command: 03 Read Multiple Registers (4xxxxx)
5) In the Data Table Address field, enter an available, unused Integer address that can be compared to 0.
6) The Size in Elements field should be 1 to read the first word of the Command Block.
7) Choose "16 Bit" for the Data field.
8) Leave the Message Timeout at 2. This may need to be increased if the cable quality is suspect or if the environmental conditions are less than ideal.
9) The MB Data Address should be 11000 for the Command Block. This equates to 411000 in Modicon Address style.
10) The Slave Node Address should match the LED Display ID (Node Address) configured in the LED Display for the port you are connecting to.
11) Keep executing the Read MSG of the Command Block until the 1st element is 0. Once the value equals 0, the Status can be read.

When the Modbus Write is complete, Read the first register of the ViewMarq ASCII Command Block and put the value in N11:0.

```
  B3:0                                                              ─MSG──────────────────
──┤ ├──────────────────────────────────────────────────────────  Read/Write Message      ─(EN)─
   1                                                               MSG File      MG10:1    ─(DN)─
                                                                       Setup Screen    <   ─(ER)─

                                                                                            B3:0
                                                                                           ─(U)─
                                                                                            1

                                                                                            B3:0
                                                                                           ─(U)─
                                                                                            0
```

Compare N11:0 to 0 (zero). If it is 0, continue with the next Read. If it is not equal to 0, then repeat the previous Read

```
  MG10:1      ─EQU──────────           B3:1                                      B3:0
──┤ ├──────   Equal                 ─[ONS]──────────────────────────────────────(L)─
   DN         Source A    N11:0         3                                          2
                            0<
              Source B        0
                            0<


  MG10:1      ─NEQ──────────           B3:1                                      B3:0
──┤ ├──────   Not Equal             ─[ONS]──────────────────────────────────────(L)─
   DN         Source A    N11:0         4                                          1
                            0<
              Source B        0
                            0<
```

Use another MSG (Read/Write Message) instruction to read the Status Reply Buffer in the ViewMarq Display.

```
        ─MSG──────────────────
      ─ Read/Write Message
        MSG File      MG10:2
            Setup Screen    <
```

1) Choose an available, unused MG address.

2) Click on the "Setup Screen" option to configure the instruction.

3) Select the Channel number that was configured earlier and that is connected to the ViewMarq

4) Select Modbus Command: 03 Read Multiple Registers (4xxxxx)

5) In the Data Table Address field, enter in the starting Integer address of an available, unused Data Block that the Status data can be read into.

6) The Size in Elements field should be 120 to read the first word of the Status Block.

7) Choose "16 Bit" for the Data field.

8) Leave the Message Timeout at 2. This may need to be increased if the cable quality is suspect or if the environmental conditions are less than ideal.

9) The MB Data Address should be 11500 for the Command Block. This equates to 411500 in Modicon Address style.

10) The Slave Node Address should match the LED Display ID (Node Address) configured in the LED Display for the port you are connecting to.

Read the ViewMarq Status Block (Modbus address 411500) and put the value in N12:0
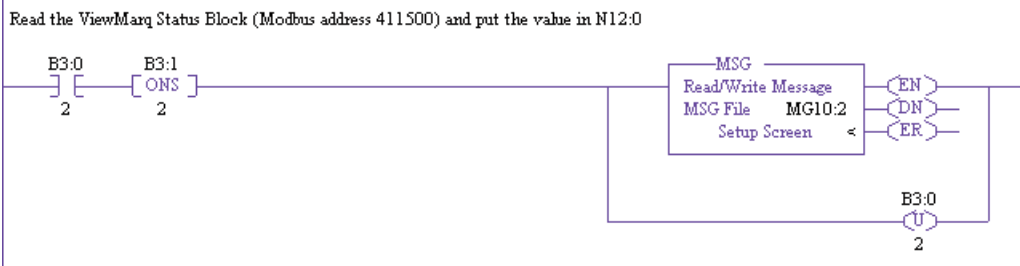
After this instruction has been enabled and is successful, the Status Reply can be read by opening the Data File where the Status data was read into and changing the Radix to ASCII.

If the syntax sent to the ViewMarq Display was correct, you will see an "OK" in your Response String. If there is a problem, you will get a Syntax error and the position where the error occurred.
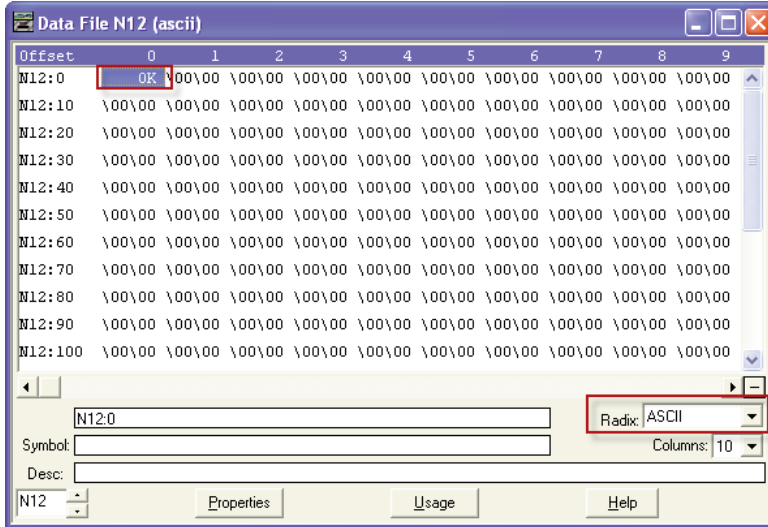
Sending Long ASCII Strings over Modbus

If the ASCII String is longer than 120 characters, multiple files and multiple instructions need to be used.

1) Copy the ASCII String from the ViewMarq Software.
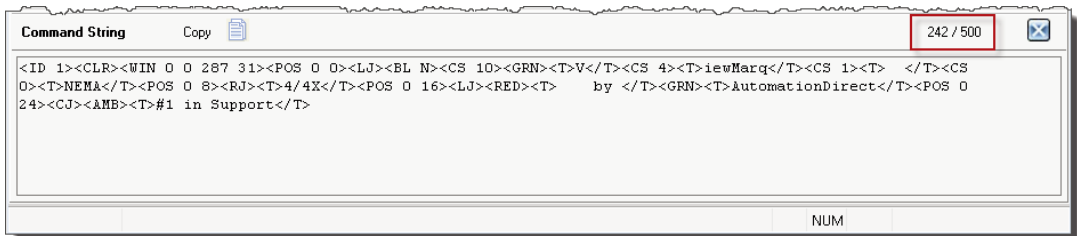
2) Double click on the first Integer file location and Paste the data in and hit Enter. It should look something like the following window.
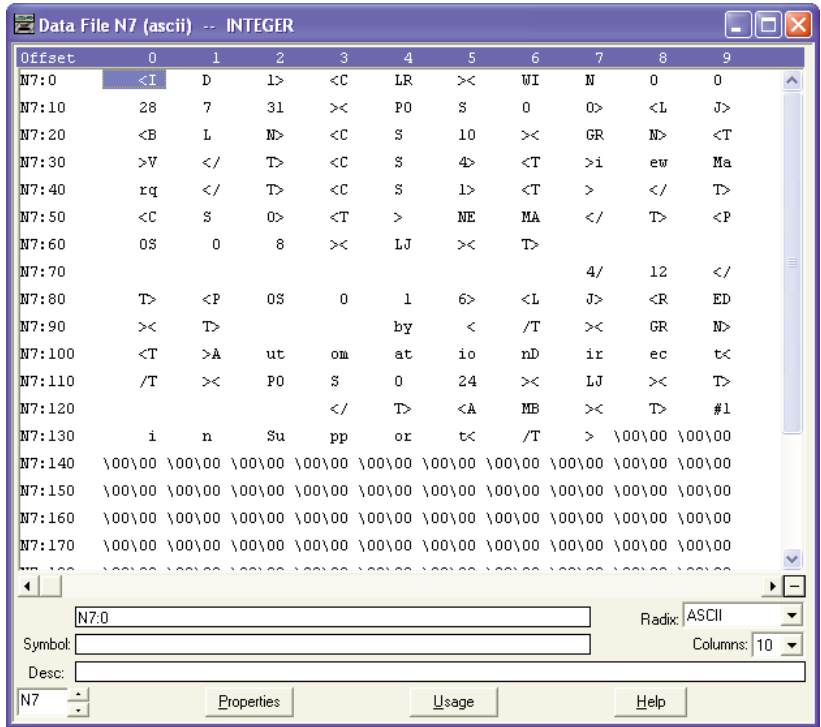
3) An additional byte (termination) must be added to the end of the data. If the String data is an odd number of bytes, the two termination characters will have to be separated between 2 different Integer addresses. Double click on the first field and add "\d\00".
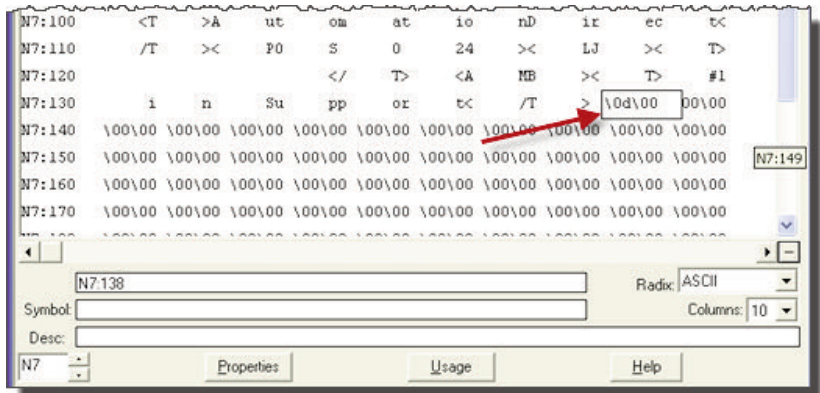
If the String data is an even number of bytes, enter in "\0d\00" in the next Integer address after the String data.

NOTE: Please note that when you hit enter, the display changes the \0d to ^M.
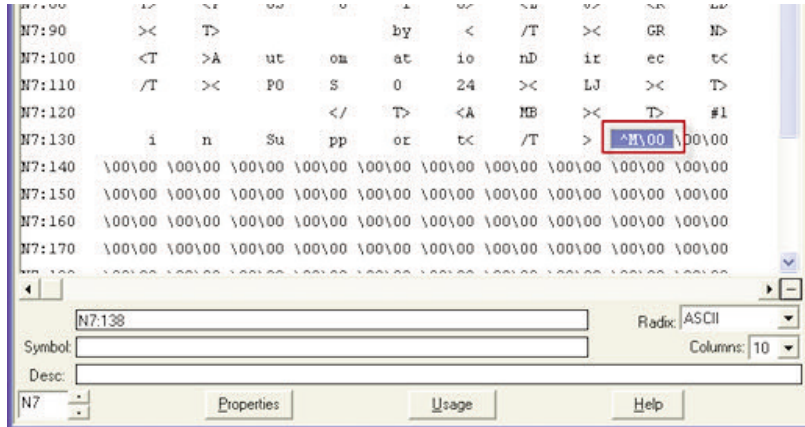
Now that the data is prepared, use multiple MSG (Read/Write Message) instructions to send the data out the port to the ViewMarq Display.
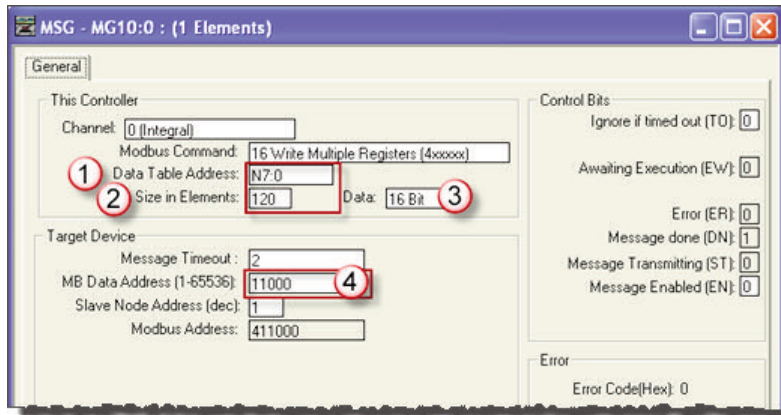
1) In the Data Table Address field, enter the starting address of the Integer data file that the copy and pasted data was placed into.

2) The Size in Elements field should be 120 registers in length.

3) Choose "16 Bit" for the Data field.

4) The MB Data Address should be 11000 for the Command Block.

Trigger the next MSG instruction with the DN (Done) bit of the previous MSG instruction. The second MSG instruction will pick up where the first instruction left off.
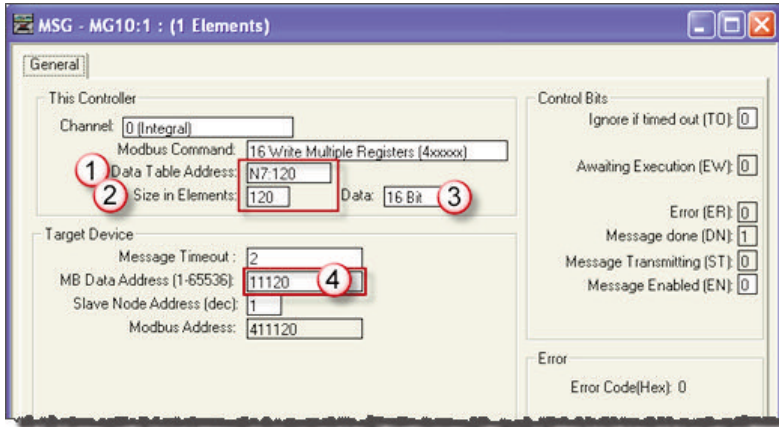
1) In the Data Table Address field, enter the next value of the Integer data file where the last instruction left off. (N7:0 + 120 = N7:120) .

2) The Size in Elements field should be 120 registers in length.

3) Choose "16 Bit" for the Data field.

4) The MB Data Address should be 11000+120 = 11120 for the Command Block.
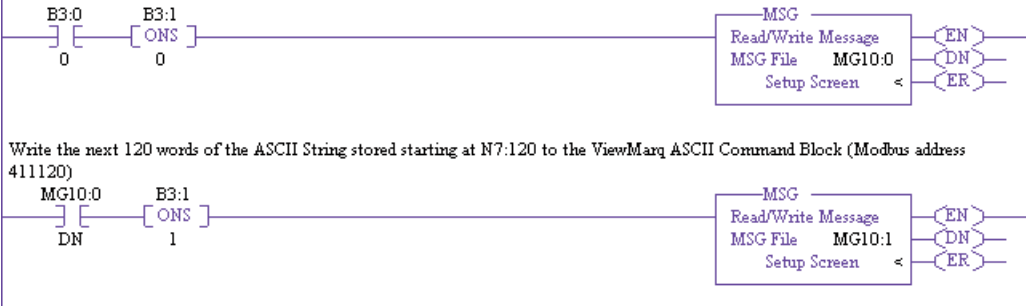
Allen Bradley CompactLogix and ControlLogix PLCs

The Allen Bradley ControlLogix and CompactLogix PLCs both utilize RS-232 and the ASCII protocol to send ASCII strings to the ViewMarq LED display

RS-232

Connect the ViewMarq Port 1 terminal block to the Allen Bradley 9-pin D-Sub port. The cable for this will need to be constructed by the user.
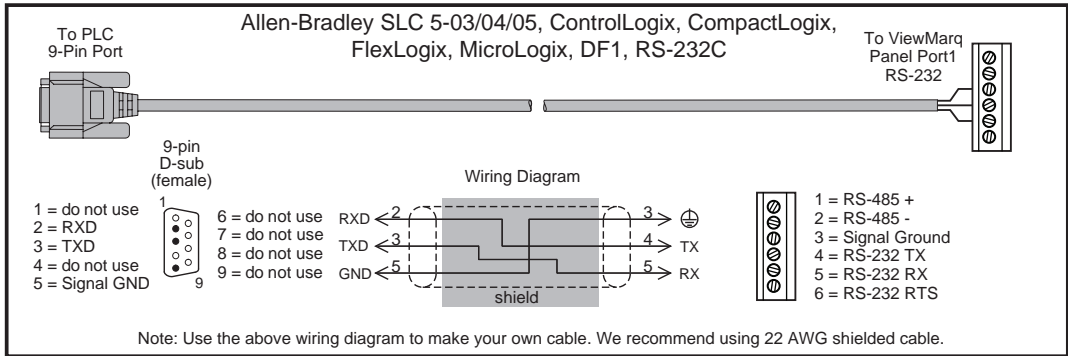
Allen-Bradley SLC 5-03/04/05, ControlLogix, CompactLogix, FlexLogix, MicroLogix, DF1, RS-232C

To PLC 9-Pin Port

9-pin D-sub (female)

Wiring Diagram

To ViewMarq Panel Port1 RS-232

1 = do not use
2 = RXD
3 = TXD
4 = do not use
5 = Signal GND

6 = do not use
7 = do not use
8 = do not use
9 = do not use

RXD 2 ⟶ 3
TXD 3 ⟶ 4 TX
GND 5 ⟶ 5 RX

shield

1 = RS-485 +
2 = RS-485 -
3 = Signal Ground
4 = RS-232 TX
5 = RS-232 RX
6 = RS-232 RTS

Note: Use the above wiring diagram to make your own cable. We recommend using 22 AWG shielded cable.

ASCII

In the LED Display Configuration window configure the ViewMarq RS-232 port to communicate using ASCII.
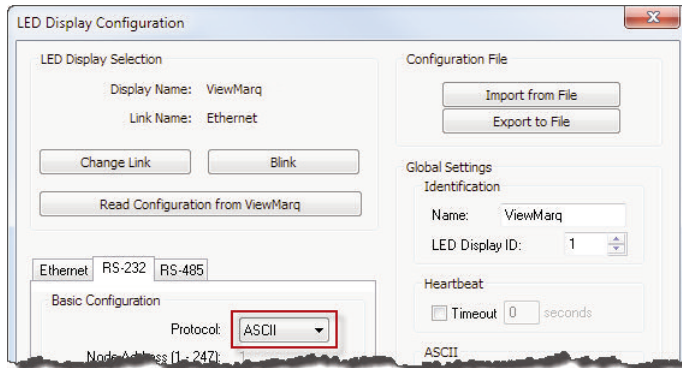
Configure the Allen Bradley PLC port to communicate ASCII with the ViewMarq LED Display.

    1) In RSLogix5000, open the Controller Properties for the PLC.
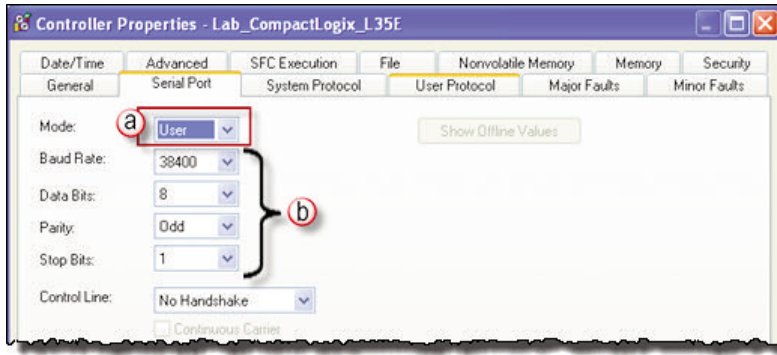
2) On the Serial Port tab

    a. Set the Mode to User.

    b. Match the Baud Rate, Parity, Stop Bits and Data Bits to match those of Port 1 of the ViewMarq LED Display.
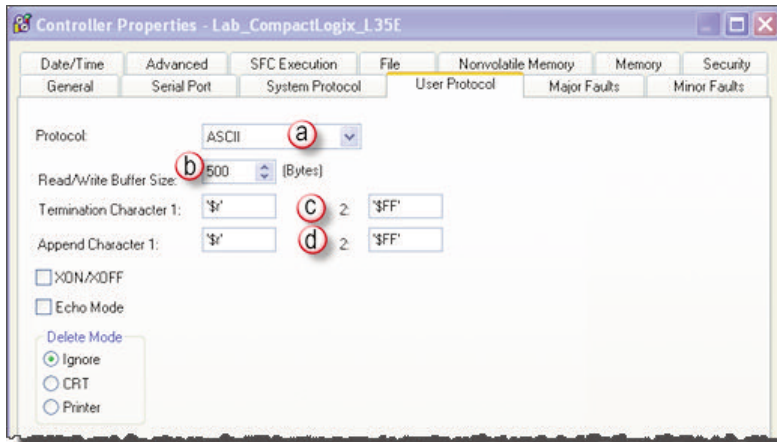
3) On the User Protocol tab

    a. Set the Protocol to ASCII.

    b. Set the Buffer size to 500 Byte. This is the maximum String the ViewMarq can accept.

      c. If the ASCII Reply from the ViewMarq will be read, set the Termination Characters as shown.

            • Termination Character 1 – '$r' is a Carriage Return

            • Termination Character 2 - '$FF' means not used

      d. Set the Append Characters as shown. These will be appended to the end of the String that is sent.

            • Append Character 1 – '$r' is a Carriage Return

            • Append Character 2 - '$FF' means not used

In order for the PLC to send the String it must be in the PLC Memory. A String data type 500 characters long is needed.

    1) In RSLogix5000 create a new String Data Type.

    2) Set Maximum Characters to 500.

Name	VM_String_Data_Type
Description:	
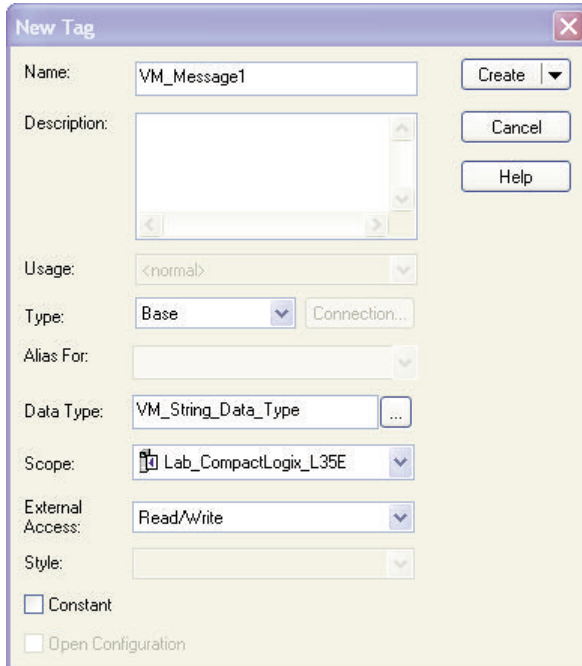
Maximum Characters: 500

Members: Data Type Size: 504 byte(s)

	Name	Data Type	Style	Description	External Access
	LEN	DINT	Decimal		Read/Write
	DATA	SINT[500]	ASCII		Read/Write

Use this new String Data Type when creating new strings to send to the ViewMarq.

7) Create a New String Tag using the Data Type just created.

8) Create a New Tag using the predefined SERIAL_PORT_CONTROL Data type.

9) Copy the ASCII String from the ViewMarq Software.

10) In RSLogix5000 open the String data file just created and paste the ASCII String into the file.

VM_Message1	...	' '	VM_String_Data_Type
+ VM_Message1.LEN		0 Decimal	DINT
+ VM_Message1.DATA		{...} ASCII	SINT[500]
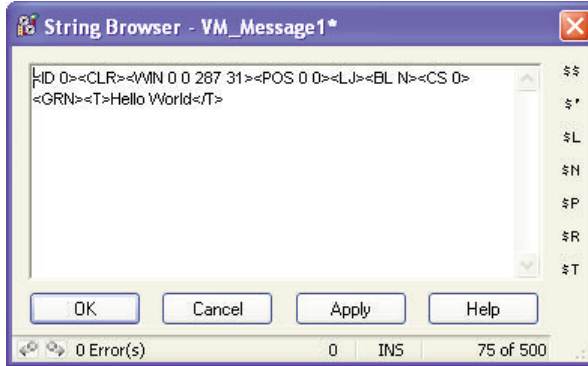
Now that the Port has been configured and the ASCII String is in the PLC Tag, use an AWA (ASCII Write Append) instruction to send the string out the serial port.
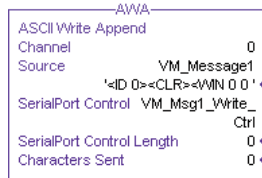
1) Select the Channel number that was configured earlier and that is connected to the ViewMarq.

2) For the Source select the String Data File the ASCII String was Pasted into.

3) For the Serial Port Control choose the SERIAL_PORT_CONTROL tag created above.
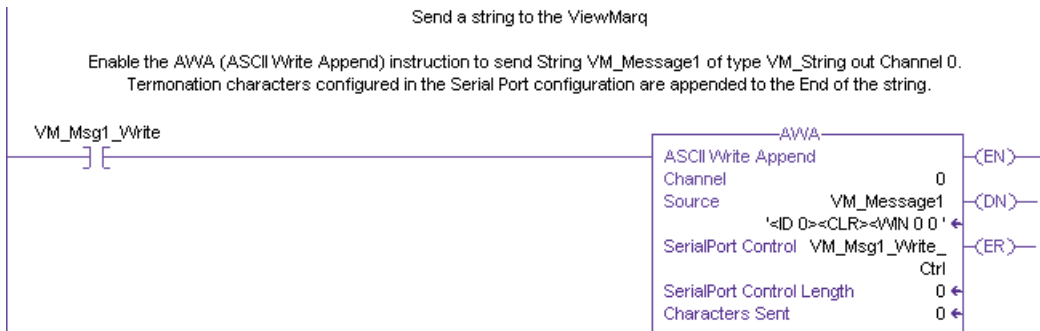
Send a string to the ViewMarq

Enable the AWA (ASCII Write Append) instruction to send String VM_Message1 of type VM_String out Channel 0.
Termonation characters configured in the Serial Port configuration are appended to the End of the string.

Reading the ASCII Reply

If the "ASCII Reply" option is enabled in the "LED Display Configuration" settings AND you are sending to an ID other than 0, your PLC may Read the Reply Response String that is sent by the ViewMarq LED Display.

1) Create a New String Tag using the new String Data Type created.
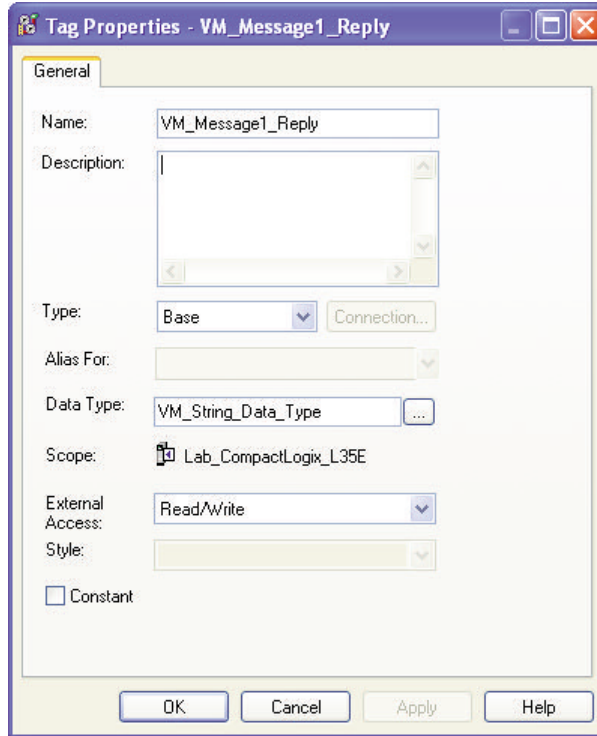
2) Create a New Tag using the predefined SERIAL_PORT_CONTROL Data type.
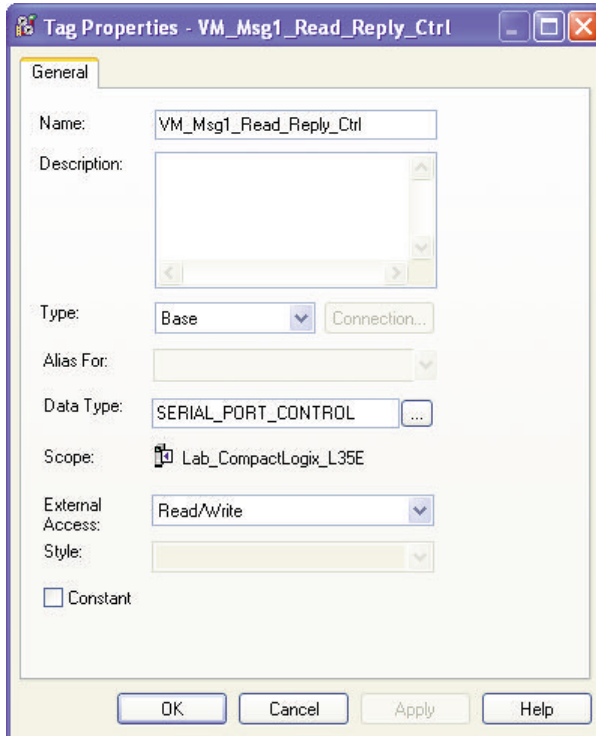
Use an ABL (ASCII Test For Line) instruction to check for the Termination characters entered in the Channel configuration and then an ARL (ASCII Read Line) instruction to read in the Status reply response String.

1) For the ABL instruction, select the Channel number that was configured earlier and that is connected to the ViewMarq.

2) For the Serial Port Control choose the SERIAL_PORT_CONTROL tag created above.

3) Trigger the ABL instruction with a normally closed contact using the Control File Enable to trigger the instruction each PLC scan.

4) For the ARL instruction, , select the Channel number that was configured earlier and that is connected to the ViewMarq.

5) For the Serial Port Control choose the SERIAL_PORT_CONTROL tag created for Read Reply above.

6) Trigger the ARL instruction with the Found (FD) bit of the ABL instruction Control File.

Read the ASCII Reply

Enable an ABL(Test Buffer for Line) instruction every scan to check the input buffer.

```
VM_Msg1_Read_C
     trl.EN                                              ┌─────ABL─────────────────────────┐
──────┤/├────────────────────────                        │ ASCII Test For Buffer Line       ├─(EN)──
                                                          │ Channel                        0 ├─(DN)──
                                                          │ SerialPort Control  VM_Msg1_Read_C├─(ER)──
                                                          │                            trl   │
                                                          │ Character Count                0 ←│
                                                          └─────────────────────────────────┘
```

If any characters are found (FD), read the input buffer VM_Msg1_Reply with an ARL (ASCII Read Line) instruction.

```
VM_Msg1_Read_C
     trl.FD                                              ┌─────ARL─────────────────────────┐
──────┤ ├────────────────────────                        │ ASCII Read Line                  ├─(EN)──
                                                          │ Channel                        0 │
                                                          │ Destination      VM_Message1_Re  ├─(DN)──
                                                          │                            ply   │
                                                          │                  '$IOK$r' ←      ├─(ER)──
                                                          │ SerialPort Control  VM_Msg1_Read_R│
                                                          │                     eply_Ctrl    │
                                                          │ SerialPort Control Length   499 ←│
                                                          │ Characters Read               4 ←│
                                                          └─────────────────────────────────┘
```

If the syntax sent to the ViewMarq Display was correct, you will see an "OK" in the String Tag . If there is a problem, you will get a Syntax error and the position where the error occurred.

VARIABLES

<div style="text-align: right">

CHAPTER

8

</div>

In This Chapter...

Variables

ViewMarq supports Dynamic messages that can contain multiple embedded data variables in each message. These variables may display string or numeric PLC register data updated in real time by the PLC, by ASCII string commands or by writing directly to Modbus addresses in the ViewMarq memory.

Variables may be used multiple times in one command yet formatted differently each time. Variables may be used in as many different command strings as needed and formatted in any way required for that string. For example, Numeric Variable #1 may be displayed as 500.0 mm in one string and or 50.00 cm just by changing the implied decimal places.

> **NOTE:** *The display format options such as color or character set for variables is set in the static text string where the <DEC> and <STR> commands are used; that is, color or character set commands do not work with, <SETS> or <SETV> commands.*

Inserting a Variable

To insert a variable into a message:

1) Click in the area of the Text Editor where you want the variable to appear.
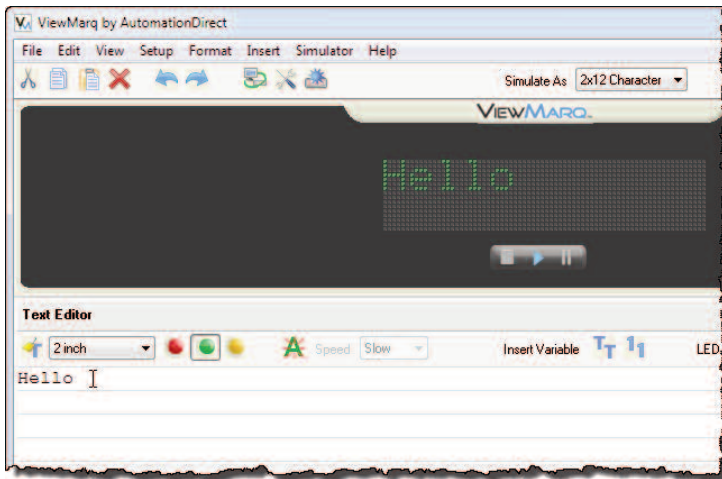
2) Select either Insert Numeric Variable or Insert String Variable from the Insert dropdown menu.
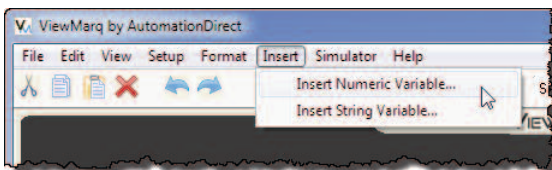
...or select the appropriate Variable toolbar button.
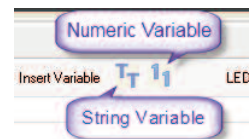
The Variable Setup Dialog will open.
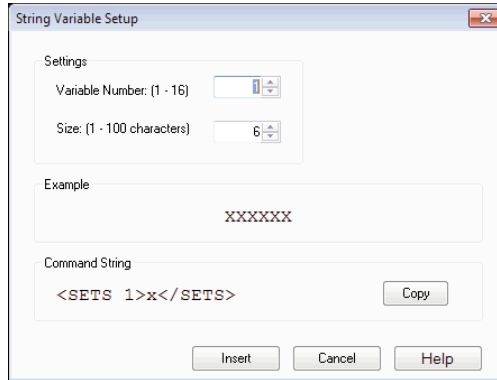
3) Select Insert to insert the variable into the Text Editor and into the Command String
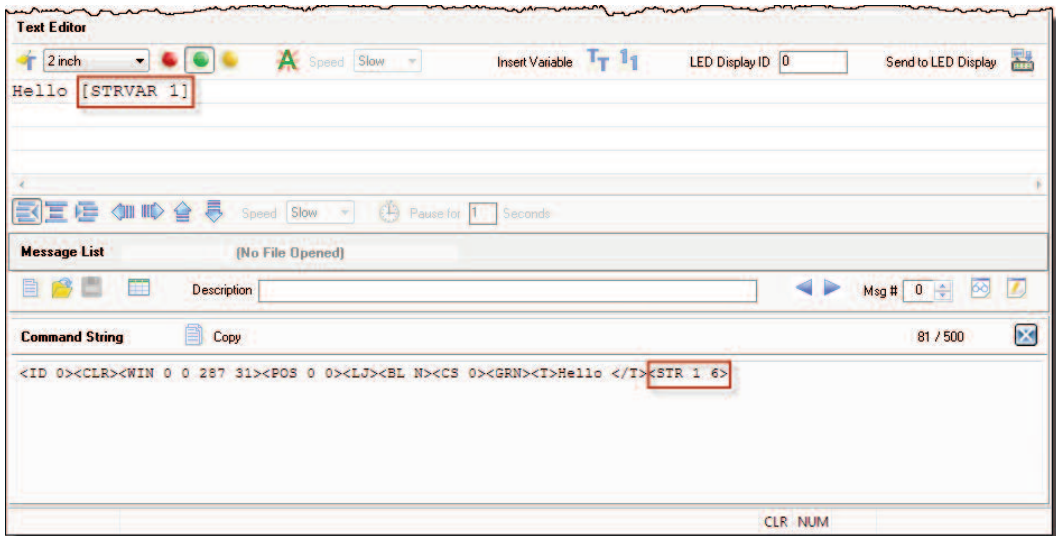
Editing a Variable

To edit a variable in a message, simply double click on the variable in the Text Editor and the setup window will open.

String Variables

String Variables are used to update ASCII text in a message without resending the entire message.

There are a few key items to remember when using String variables:

- String Variables can be inserted anywhere in the message.
- Up to 16 string variables can be used in a single message.
- 16 individual string variables are available for use
- String variables can be up to 100 characters in length.
- If a string variable value has not been set, then a number of blank spaces will be displayed equal to the number of characters in the variable in that portion of the message.
- The <SETS n>string</SETS> command string is used to set a string variable. This is covered later in this chapter.

String Variable Setup

The String Variable Setup dialog is shown below.
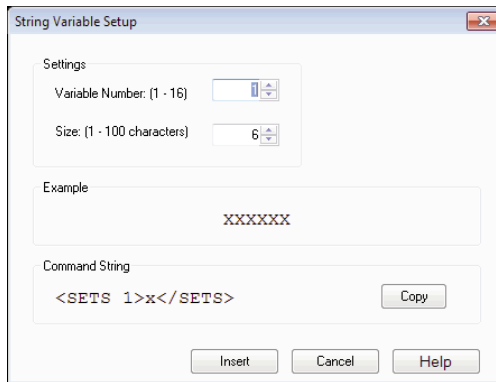
1) Select the Variable Number (1 to 16).

2) Select the Size of the string (100 characters maximum).

3) Select the **Copy** button to copy the <SETS n>string </SETS> command onto the PC clipboard. (This will be used later to set the value of the string from the PLC.)

4) Select the **Insert** button to insert the variable placeholder into your message.

The result can be seen below. The value of the string variable is not set, so there are spaces in the area that it occupies.

5) Copy and Paste the Command String into the PLC instruction.

 Attention!: Command Strings should be sent at least 100 ms apart.

When the instruction is executed, the display will look like this.

Setting String Variable Value

The value of the String Variable gets set from the PLC in one of two methods::

    1) ASCII Command String - <SETS n>string</SETS>

    2) Modbus Register Write

Using an ASCII Command String to Update the String Value

To update String Variable data using an ASCII Command string, use the Set String command <SETS n>string</SETS>

Where n is the string number (1 – 16) and string is the actual text that you wish to display.

Example 1 – Set String Value

1) From the String Variable Setup dialog copy the Set String command.
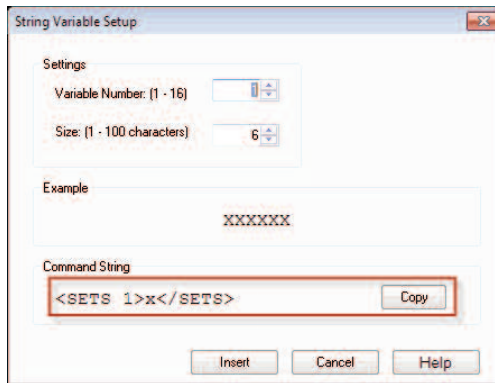
2) Paste this Command String in the PLC instruction; Add the ID command and the string = "World" to be displayed
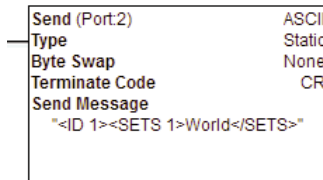
ATTENTION

Attention!: Variables should be updated no more than every 100ms.

When the above instruction is executed, the display will now look like this.

Example 2 – New Set String Value

If the instruction below with a different string="Smiley" is executed…

```
Send (Port:2)                          ASCII
Type                                   Static
Byte Swap                               None
Terminate Code                            CR
Send Message
   "<ID 1><SETS 1>Smiley</SETS>"
```

...the message will look like this.

Example 3 – String Variable used in another String Command

If another String is sent to the ViewMarq LED Display using the same String Variable, the same value will be displayed.
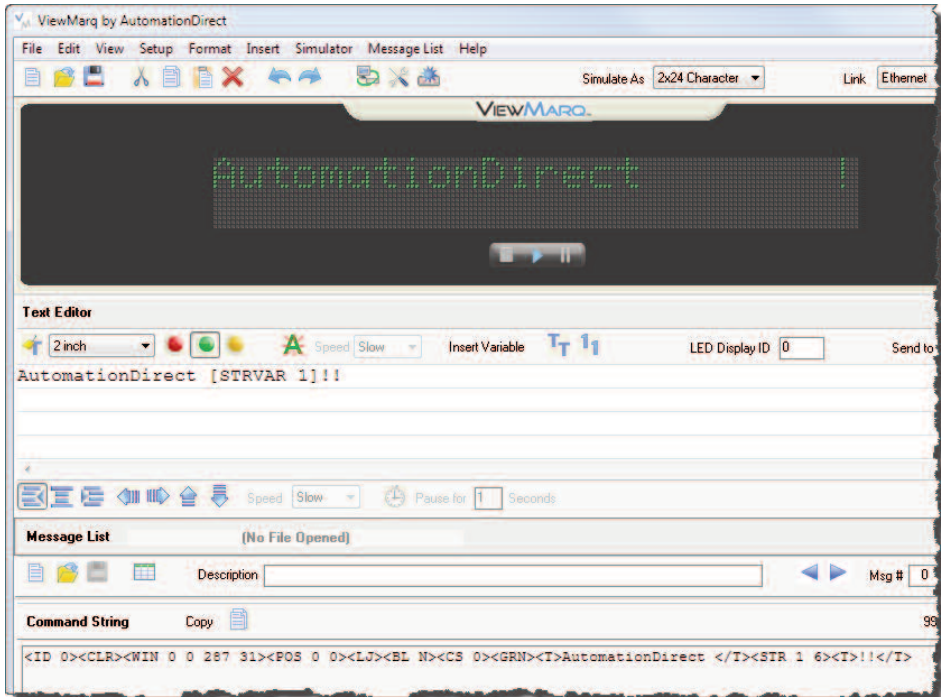
Copy and Paste the Command String into the PLC instruction.

```
Send (Port:2)                    ASCII
Type                            Static
Byte Swap                        None
Terminate Code                     CR
Send Message
"<ID 1><CLR><WIN 0 0 287 31><POS
        0 0><LJ><BL N><CS
    0><GRN><T>AutomationDirect
        </T><STR 1 6><T>!!</T>"
```

When the above instruction is executed, the display will look like this.

AutomationDirect Smiley!

Variable #1 is still equal to "Smiley" until it is set by another <SETS> command.

NOTE: *The conditions for executing the instructions above depend upon the controller used. The above instructions are the SEND instruction from the CLICK PLC and are given as examples only. Please refer to Chapter 7 for the details of sending strings from other PLC models.*

8

Using Modbus to Update the String Value

The String Value may also be set by writing directly to the associated Modbus address. See Chapter 7 - Sending Messages from your PLC to the ViewMarq for more information on writing to Modbus addresses in the ViewMarq Display.

Following is a table of the corresponding ViewMarq String Variables and Slave Modbus addresses.

ViewMarq String	Modbus Address
String Variable #1	400200
String Variable #2	400250
String Variable #3	400300
String Variable #4	400350
String Variable #5	400400
String Variable #6	400450
String Variable #7	400500
String Variable #8	400550
String Variable #9	400600
String Variable #10	400650
String Variable #11	400700
String Variable #12	400750
String Variable #13	400800
String Variable #14	400850
String Variable #15	400900
String Variable #16	400950

Numeric Variables

Numeric Variables are used to update Numeric values within a message without changing or resending the entire message.

Here are a few key items to remember when using Numeric variables:

- Numeric Variables can be inserted anywhere in the message.
- Up to 32 Numeric variables are available for use.
- Numeric variables are 32 bit, bi-polar, two compliment.
- Numeric variable range is - 2147483647 to 2147483647.
- Up to 10 Implied decimal places can be configured for each instance of a Numeric variable.
- If a Numeric variable has not been set, then zeroes or spaces will be displayed in the message.
- The <SETV n 1234> command is used to set the value of a Numeric variable.

Numeric Variable Setup

The Numeric Variable Setup dialog is shown below.
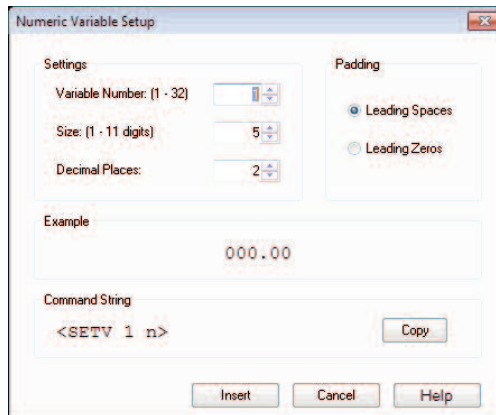
1) Select the Variable Number (1 to 32).

2) Select the Size of the Number (10 digits maximum, 11th place for sign).

3) Select the Implied Decimal place up to 10.

4) Select Leading Spaces or Leading Zeros.

5) Select the **Copy** button to copy the <SETV 1 n> command onto the PC clipboard (this will be used later to set the value of the variable from the PLC).

6) Select the **Insert** button to insert the variable placeholder into your message.

The result can be seen below. The value of the numeric variable is not set, so there are zeros with leading spaces in the area that it occupies.

7) Copy and Paste the Command String into the PLC instruction.

ATTENTION

Attention!: Command Strings should be sent at least 100 ms apart.

When the above instruction is executed, the display will look like this.

Setting Numeric Variable Value

The value of the Numeric Variable gets set from the PLC in one of two methods::

1) 1) ASCII Command String - <SETV 1 n>

2) Modbus Register Write

Using an ASCII Command String to Update the Numeric Value

To update Numeric Variable data using an ASCII Command string, use the Set String command.

<SETV 1 n>

Where 1 is the variable number (1 – 32) and n is the actual numerical data without decimals.

Example 1 – Set Numeric Value

1) From the Numeric Variable Setup dialog, copy the Set String command.
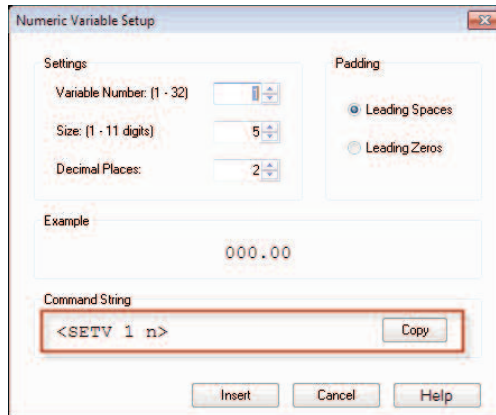
2) Paste this Command String in the PLC instruction; Add the ID command and numerical data to be displayed as shown below.
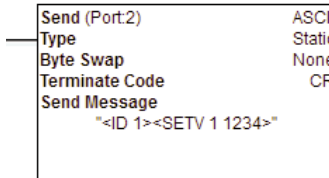
ATTENTION

Attention!: Variables should be updated no more than every 100ms.

When this instruction is executed, the display will look like this.

Example 1 – Set New Numeric Value

If the instruction below with a different n="5432" is executed…

```
Send (Port:2)              ASCII
Type                       Static
Byte Swap                  None
Terminate Code               CR
Send Message
    "<ID 1><SETV 1 5432>"
```

…the message will look like this.

Example 3 – Numerical Variable used in another String Command

If another String is sent to the ViewMarq LED Display using the same String Variable, the same value will be displayed.

Copy and Paste the Command String into the PLC instruction.

```
Send (Port:2)                    ASCII
Type                            Static
Byte Swap                         None
Terminate Code                      CR
Send Message
"<ID 1><CLR><WIN 0 0 287 31><POS
        0 0><LJ><BL N><CS
0><GRN><T>Press(x10)= </T><DEC
            1 4 1>"
```

When the above instruction is executed, the display will look like this.

Press (x10) = 543.2

Variable #1 is still equal to "5432" until it is set by another <SETV> command.

NOTE: *The conditions for executing the instructions above depend upon the controller used. The above instructions are the SEND instruction from the CLICK PLC and are given as examples only. Please refer to Chapter 7 for the details of sending strings from other PLC models.*

8

Using Modbus to Update the String Value

The Numeric Value may also be set by writing directly to the associated Modbus address. See Chapter 7 - Sending Messages from your PLC to the ViewMarq for more information on writing to Modbus addresses in the ViewMarq Display.

Below is a table of the corresponding ViewMarq Numeric Variables and Slave Modbus addresses.

ViewMarq Variable	Modbus Address	
	High Word	Low Word
Numeric Variable #1	400100	400101
Numeric Variable #2	400102	400103
Numeric Variable #3	400104	400105
Numeric Variable #4	400106	400107
Numeric Variable #5	400108	400109
Numeric Variable #6	400110	400111
Numeric Variable #7	400112	400113
Numeric Variable #8	400114	400115
Numeric Variable #9	400116	400117
Numeric Variable #10	400118	400119
Numeric Variable #11	400120	400121
Numeric Variable #12	400122	400123
Numeric Variable #13	400124	400125
Numeric Variable #14	400126	400127
Numeric Variable #15	400128	400129
Numeric Variable #16	400130	400131
Numeric Variable #17	400132	400133
Numeric Variable #18	400134	400135
Numeric Variable #19	400136	400137
Numeric Variable #20	400138	400139
Numeric Variable #21	400140	400141
Numeric Variable #22	400142	400143
Numeric Variable #23	400144	400145
Numeric Variable #24	400146	400147
Numeric Variable #25	400148	400149
Numeric Variable #26	400150	400151
Numeric Variable #27	400152	400153
Numeric Variable #28	400154	400155
Numeric Variable #29	400156	400157
Numeric Variable #30	400158	400159
Numeric Variable #31	400160	400161
Numeric Variable #32	400162	400163

Maintenance and Troubleshooting

CHAPTER
9

In This Chapter...

Introduction

Even though the ViewMarq requires virtually no maintenance, there are a few steps you can take to insure the longevity of your product and to maintain the warranty, safety and environmental ratings.

- To maintain the NEMA ratings the cable entry points in the back cover must be properly sealed. The ViewMarq is provided with cable glands to seal all of these points whether a cable is used or not. The provided covers must be properly installed as well.

- To provide for proper heat dissipation, keep the ViewMarq clear of all dust and debris.

- Regularly inspect ViewMarq mounting brackets and the mounting surfaces.

- To maintain viewing distances and clarity, clean the lens with a damp cloth as needed.

NOTE: *To maintain UL508 rating, Flexible Conduit must be used for all conductors entering the message display.*

Updating Firmware

Occasionally new firmware will be released to enhance your ViewMarq. The firmware may be downloaded from **automationdirect.com/downloads** and updated using the ViewMarq software.

1) Select the Link for the display to update.
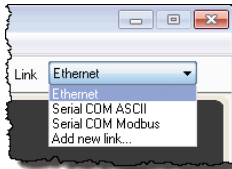
2) From the Setup dropdown menu, select Update Firmware as shown.
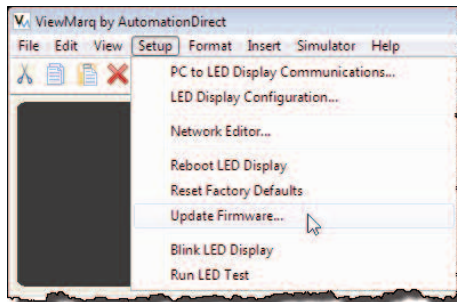
3) The Update Firmware dialog will open. The panel information will be read from the slected Link; Display Name, Firmware version and Link Name.

The Firmware is in two parts:
    A1 – ASCII character set version
    FW – Base Firmware (Operating System).
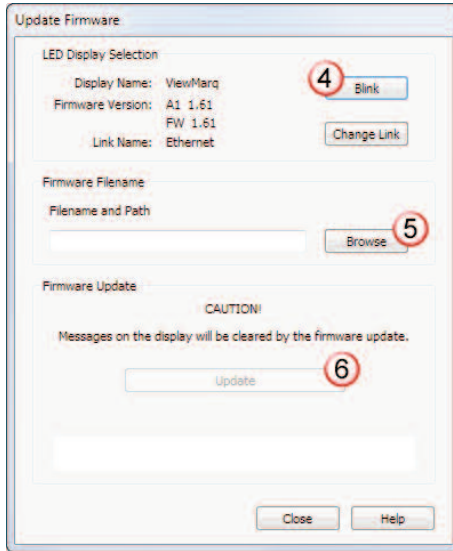
4) Select the Blink button to make sure you have the correct display.

5) Select the Browse button and navigate to the location you saved the firmware file. It should be named something like VM-A1015300.vmf .

6) Select the Update button. You will see a warning like the one shown below. Make sure all messages saved in the message list have been saved on your hard drive. They will be cleared from the panel. All communication settings will be preserved in the panel.
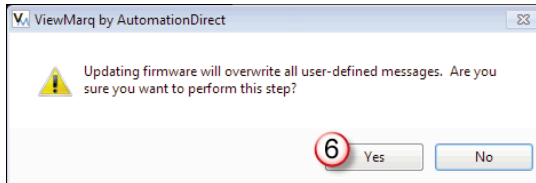
7) Click the Yes button. While upgrading, the software will display an activity bar. As long as the green shade in the activity bar is moving, the display is still upgrading. When the activity bar stops, click the Close button. The ViewMarq will display the following messages during the firmware update.

  a) FW Loading
  b) FW Updating
  c) Load Firmware
  d)FW Loading

8) When the firware update is complete, the display will reset and the default startup screen begins to scroll.

Load Firmware Message

If the software has completed the Firmware Update process, but the ViewMarq shows a Load Firmware message, repeat the firmware update process. Do not change any communication settings, simply repeat step 5.

Reboot LED Display

Sometimes you may wish to clear the display or power cycle the display but it is not within reach. To Reboot the LED Display, select the Link for the Display to reboot.
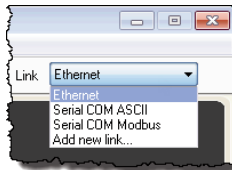
From the Setup dropdown menu, select Reboot LED Display as shown.
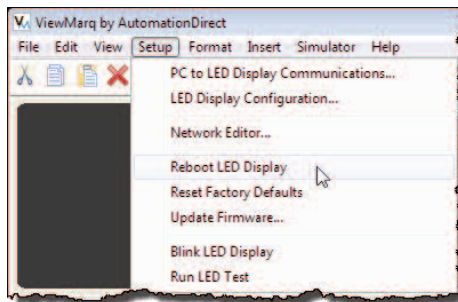
Reset Factory Defaults

For troubleshooting purposes, there may be a need to reset the display to factory defaults. To return the LED display to factory defaults, select the Link for the Display to reset.
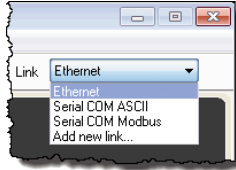
From the Setup dropdown menu, select Reset Factory Defaults as shown.
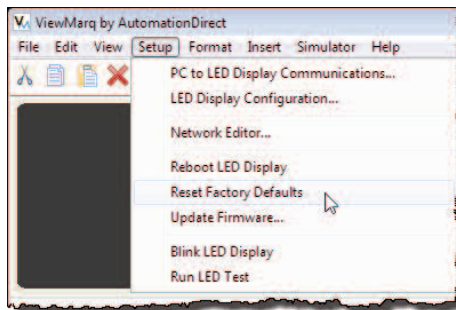
A Reset Factory Defaults action will set communication parameters to the defaults:

- The ViewMarq name will be reset
- The IP address will be set back to Obtain Address from DHCP Server
- Port 1 will be set to ASCII, node address 1, baud rate 38400bps, 8 data bits, odd parity and 1 stop bit
- Port 2 will be set to Modbus, node address 1, baud rate 38400bps, 8 data bits, odd parity and 1 stop bit

Click Yes to reset the display back to factory defaults.
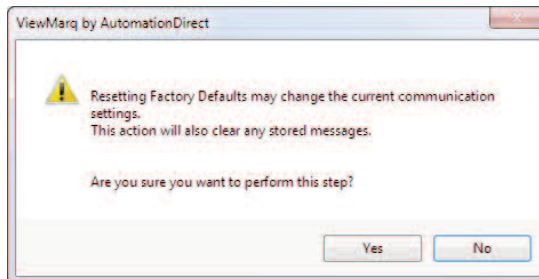
LED Test

If you suspect that an LED component has failed, you may verify the LED components with the LED test. The test will initiate a pattern of scrolling lines cycling horizontally then vertically through each of the three colors (red, amber, green). To perform an LED Test, select the Link for the display to test.
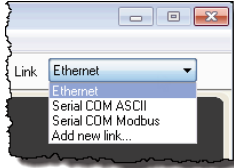
From the Setup dropdown menu, select **Run LED Test** as shown below.
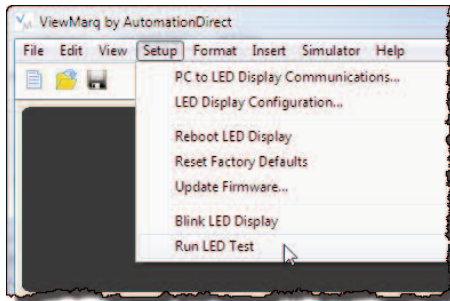
The LED Test dialog will open
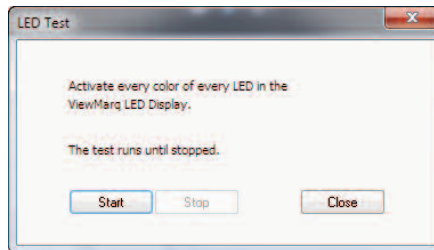
Select Start to run the LED test. Check all of the LEDs on the ViewMarq display as they change to red, amber and green to make sure all LEDs are working properly.

Select Stop or Close to stop the LED test.

NOTE: *LEDs are NOT user replaceable.*

Error Messages

There are only a few error messages issued by the ViewMarq Software. All of these messages are communication related. They typically have to do with a faulty connection, incorrect communication settings or an electrically noisy environment.

Qualified technicians that know good wiring, shielding and grounding practices should be followed when installing all communication and power conductors to ensure proper operation of the LED display. Below is each possible error message and suggested solutions.

Error	Suggested Action
Unable to connect to ViewMarq	Serial: Serial Port Unavailable Ensure that some other application does not have control of the COM port and that the specified COM port is valid *Note: USB serial converters may choose different COM ports when disconnected and re-connected*
Unable to connect to ViewMarq	Ethernet: Ethernet - TCP Connection Timeout 1) Check the Cable and Ethernet switch 2) Increase the TCP Connect Timeout time in the Link Setup 3) Check the IP Address and Port Number in the Link Setup 4) Eliminate possible causes of electrical noise 5) Try to PING the display. If PING fails ensure that the PC and the display are on compatible subnets.
Modbus Timeout Occurred	1) Check communication settings 2) Increase timeout time in the Link Setup 3) Insure a good physical connection and cable condition 4) Eliminate possible causes of electrical noise
Invalid Modbus Checksum Received from ViewMarq	1) Check communication settings 2) Insure a good physical connection and cable condition 3) Eliminate possible causes of electrical noise
Invalid Modbus Response Received	1) Check communication settings 2) Insure a good physical connection and cable condition 3) Eliminate possible causes of electrical noise

9

ASCII Command String Syntax Error Messages

The ViewMarq checks the ASCII command String for errors and reports these error in several ways.

1) If ASCII Reply is enabled in the Display, then the Display will respond on the port on which the ASCII command String was received. This may be read by PLC's that are capable of asynchronous communications.

2) The ASCII Reply text may also be read from the ASCII Response Buffer over Modbus or Modbus TCP started at register 415000. The buffer is 512 bytes long. See Chapter 7 for more information on reading the ASCII Reply over Modbus.

3) If the Syntax Error Check option is enabled in the ViewMarq Display Configuration, syntax errors will scroll across the display. See Chapter 6 for more on enabling this feature.

ASCII Command String Syntax Error Messages		
Error Message	*Description*	*Notes*
E1: Invalid SETS format	Missing string number or out of range	
E2: Invalid SETV format	Missing variable number or out of range	
E3: Invalid Pause Time	Pause time must between 0 and 1000 Seconds	
E5: Invalid SETS Value	SETS String too long, maximum is 100 characters	
E6: Display Memory Full	Use the CLR command to prevent memory from becoming full. There can be a maximum of 31 <T>, <DEC>, <STR>, <PORTPAR>, <IPADDR>, <VER>, <NAME> commands total in the displayed message.	
E7: Text Memory Full	Maximum text characters that can be displayed is 1500. Use the <CLR> command	
E8: Invalid STR/DEC Format	Syntax error in STR or DEC command	
E9: Invalid PORTPAR Format	Syntax error in PORTPAR command	
E10: Parameter Out of Range	CS parameter out of range	
E11: DO command not allowed	A DO command is calling a string with an embedded DO command	
E12: Download Error	Error Downloading strings (Future)	
E13: Invalid STR String Number	String number must be 1-16	
E14: Invalid DEC String Number	Variable number must be 1-32	
E16: Invalid DEC Size	Maximum length is 11 characters	
E17: Invalid STR Size	Maximum length is 100 characters	
E18: Invalid Message Number in DO	Invalid Message Number, Message does not exist (Future)	
E19: Invalid Message Number in DS	Invalid Message Number, Message does not exist (Future)	
E20: Status Buffer is Empty	Status buffer in empty probably because STATUS command is used before any other command	
E21: Syntax error at pos #	General Syntax error at position #	Syntax errors could be caused by sending ASCII strings too fast with no handshaking.
E22: Syntax error in DO at pos #	General Syntax error in DO at position # (Future)	
E23: Error in MSG # %d:	Error in MSG # %d - Error is displayed on ViewMarq (Future)	
E24: Invalid ID Format	ID number must be between 0 - 247	These are not displayed errors. They will be stored in the Status register (411500)
E25: Missing <ID #>	ID command contains no ID number	

Diagnostic LED

The diagnostic LED is a way for a user to see if communications are successful to the panel without removing the back covers to watch the communication LED's flicker. When this box is checked, the lower left LED on the marquee display will stay lit.

Each time the message display receives a message the LED will change color to indicate the communication was received. The message may not be displayed due to a Syntax Error or incorrect ID number, but the LED will still change color. It is a good practice to set this setting on until you are sure everything is working correctly with your LED display and your messages are displaying as expected.

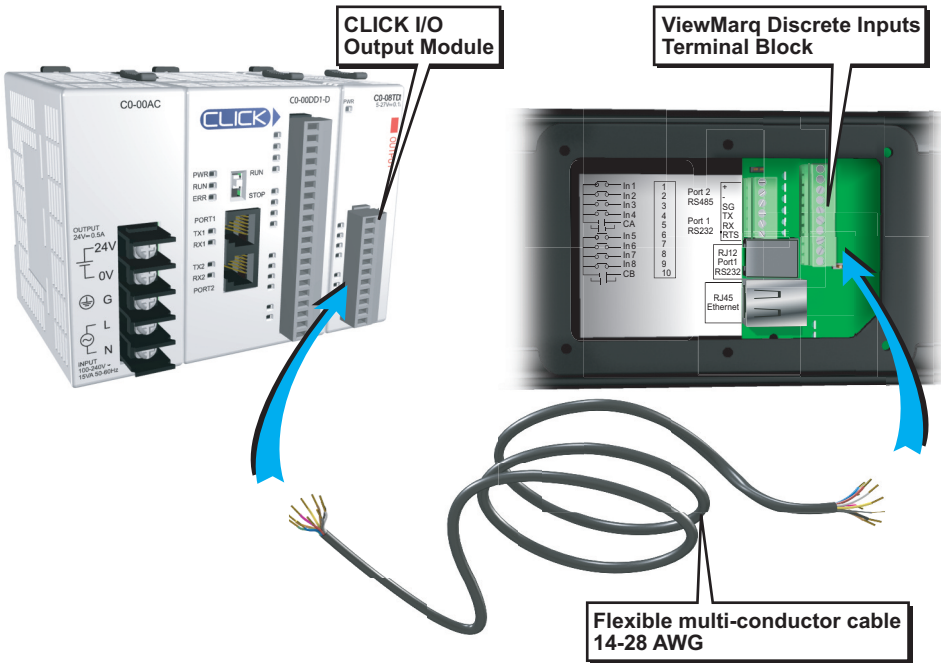# MESSAGE LIST

In This Chapter...

Introduction

Message lists are created using the ViewMarq configuration software. The Message List can be downloaded to the ViewMarq and triggered for display using ASCII commands or discrete signals wired to the ViewMarq input terminals.

Message List Features

- Up to 200 Messages can be added to a Message List and saved on the PC
- Multiple Message Lists can be saved to the PC
- A single Message List can be downloaded to the ViewMarq memory
- The Message List in the display is retained on power cycle
- Messages downloaded to the ViewMarq can be triggered by using the following:
  - <DO n> ASCII command from the PLC
  - Discrete input terminals inside the communication connection opening of the ViewMarq
- Message #0 becomes the Startup message.

CLICK I/O Output Module

ViewMarq Discrete Inputs Terminal Block

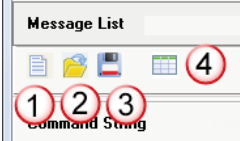**Flexible multi-conductor cable 14-28 AWG**

Main Window Message List Toolbar

The Message List control toolbar is in the Main Window of the ViewMarq Configuration Software.
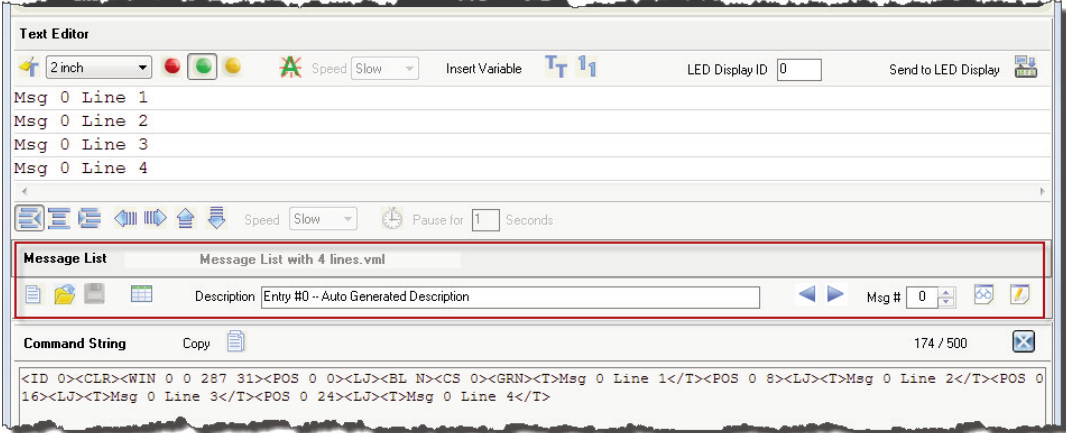
The Message List toolbar allows you to:

1) Create a new Message List
2) Open an existing Message List
3) Save a current Message List
4) Open the Message List Manager

The Message List toolbar also shows:

1) The Message List file name
2) The Description for the present message in the list
   a) The Description is auto-generated for convenience but can be edited by the user.

From the Message List toolbar you can:

1) Read the previous message in the Message List into the editor
2) Read the next message in the Message List into the editor
3) Select a particular message number location to Read or Write
4) Read the message from the selected location into the Text Editor
5) Write the present message in the Text Editor to the selected location

The Message List Management Window

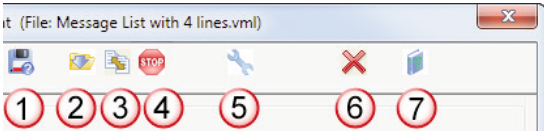The Message List Window is opened using the Open Message List Management Dialog button.
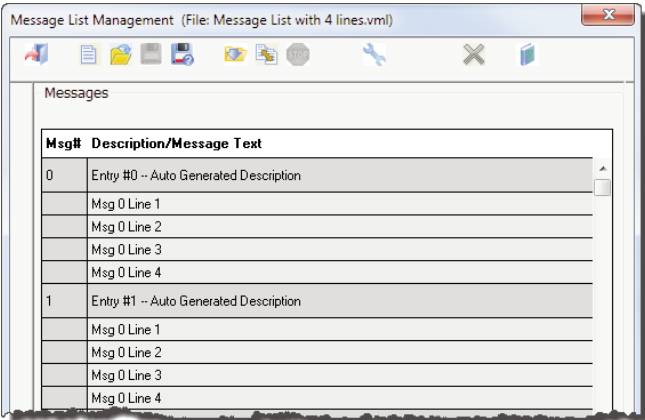
In the Message List Manager you can:

1) Close the Message List Management dialog
2) Create a new Message List
3) Open an existing Message List
4) Save the current Message List

The Message List Management toolbar also includes:

1) Save the current Message List with a different file name
2) Download the Message List to a connected ViewMarq LED display
3) Compare a Message List file to the Message List in the connected ViewMarq display
4) Stop the Message List download or compare
5) Open the Message List test dialog
6) Delete the selected Message from the list
7) Open Message List help

Adding Messages to a Message List
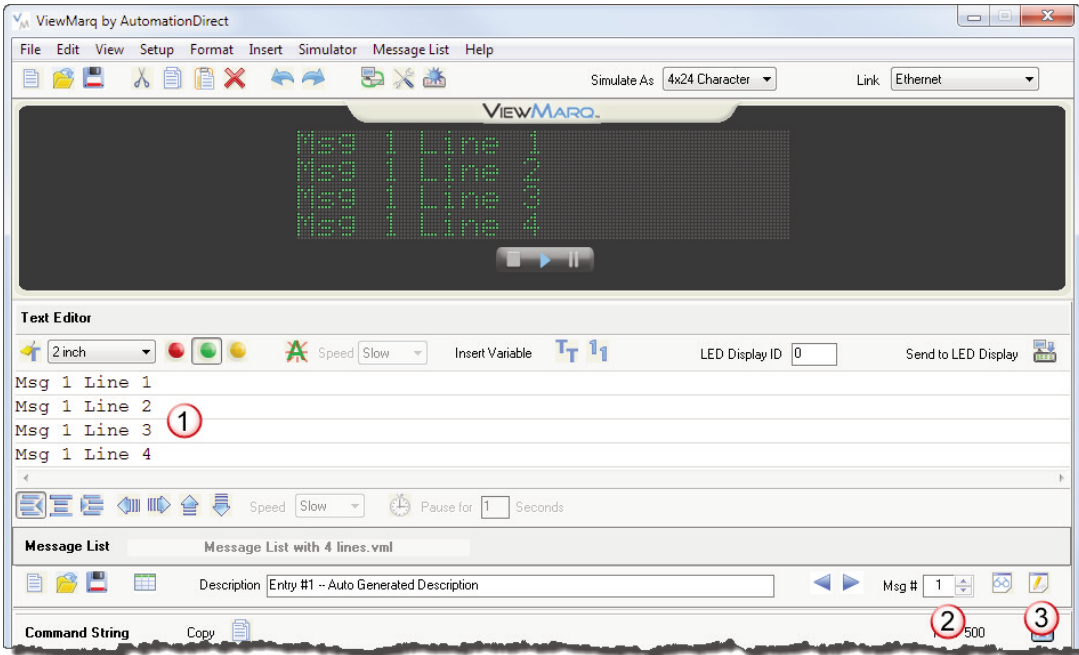
1) Create a message in the ViewMarq Text Editor.
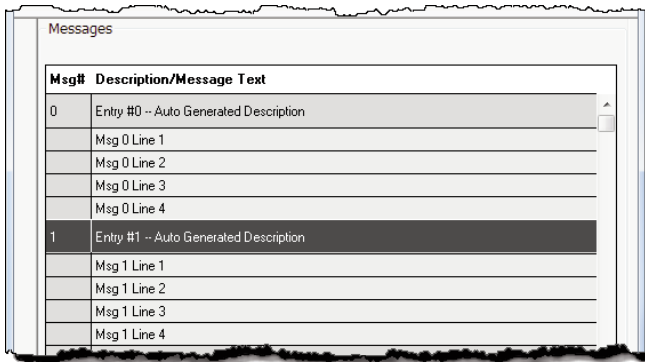2) Select a message number on the Message List Toolbar.
3) Select the Write Message toolbar button

The message in the Text Editor with the Description and Text is saved to the Message List to the position selected as shown.

Editing an existing Message List

1) Open a previously saved Message List
2) Select the Next or Previous Message buttons until the message is selected

or

3) Select the message number
4) Select the Read Message toolbar button
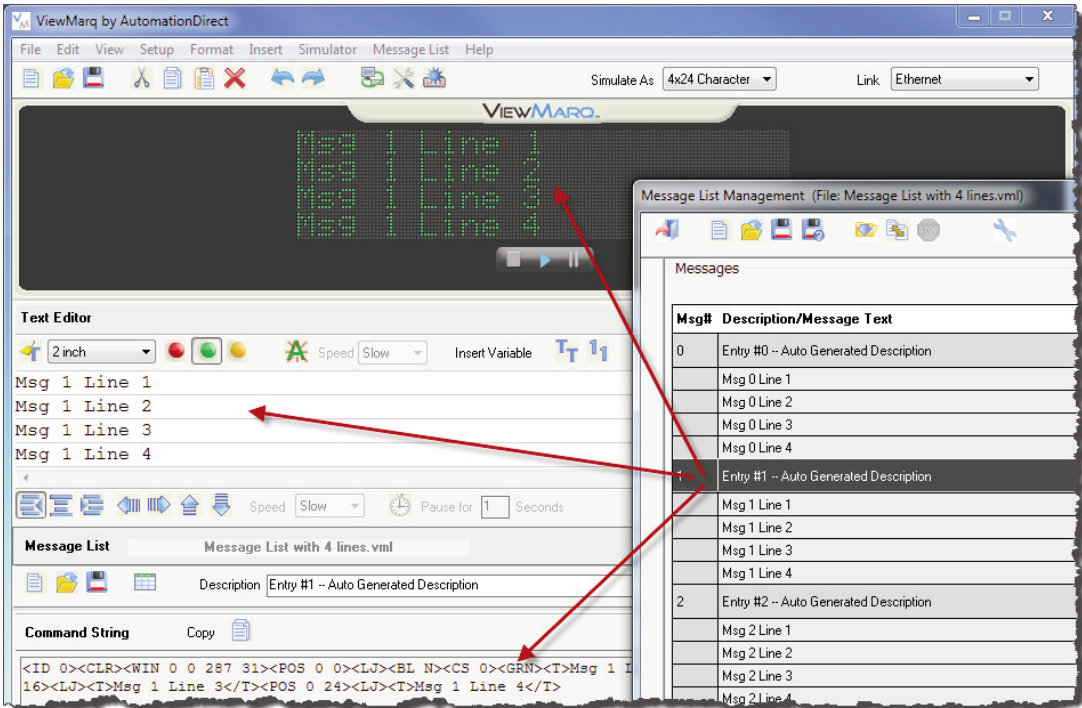
The message selected will be read from the Message List into the text editor. Make any modifications necessary.

5) Select the Write Message toolbar button

The message in the Text Editor with description and text is saved to the Message List.

Autoloading Message

Automatically load a message into the text editor by selecting it in the Message List Manager window.

Configuring the ViewMarq display for Message List

Starting with Firmware version 1.65, the ViewMarq LED display is configured to accept and display messages from a downloaded Message List. This may be accomplished using an ASCII command or the discrete inputs.
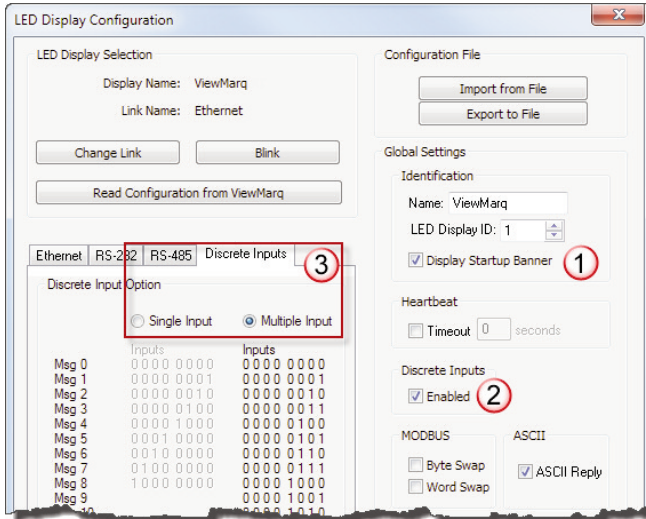
1) Display Startup Banner:

NOTE: This setting only takes affect if there is no message loaded into Message #0.

If Display Startup Banner is **selected**, the ViewMarq Name, Firmware and Communication Settings will be displayed on power up when there are no Discrete Inputs turned on.

If Display Startup is **not selected**, then the Display will be blank on power up when there are no Discrete Inputs turned on.

2) Discrete Inputs:

To use the hardwired Discrete Inputs to display messages in the Message List, the Discrete Inputs checkbox must be selected.

3) Discrete Input Option:

Single or Multiple Inputs - See **Triggering Messages** in this chapter.

Message Exists in Panel Memory	Discrete Inputs Enabled	Display Startup Banner Enabled	On Start Up ViewMarq will display
X	X	X	Triggered Message per Discrete Inputs
X	X		Triggered Message per Discrete Inputs
X		X	Startup Banner (Comm Settings)
X			Blank Display
	X	X	Startup Banner (Comm Settings)
	X		Blank Display
		X	Startup Banner (Comm Settings)
			Blank Display

At startup, the ViewMarq will be in one of three states:

1) The message triggered by Discrete Inputs, if the message exists in the panel memory and the Discrete Inputs are inabled and on

2) The Startup Banner (Communication Settings), if the the Startup Banner is enabled and discrete inputs are not enabled or the Startup Banner is enabled and the messaged triggered is blank

3) The discplay is blank and the Startup Banner is not enabled and Discrete Inputs are not enabled or the Startup Banner is not Enabled and the message triggered by the Discrete Inputs is blank

NOTE: *After any changes are made, the configuration needs to be written to the ViewMarq display in order for them to take effect.*

10

Downloading Messages to the ViewMarq Display
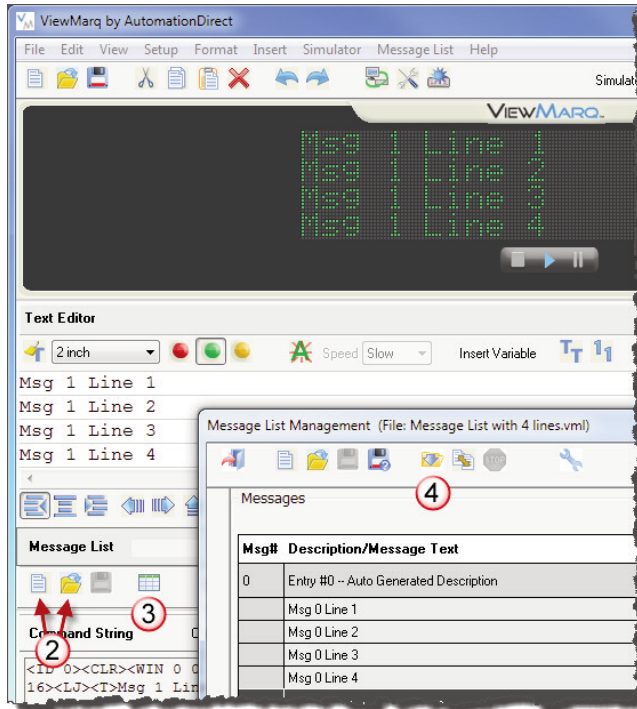
Open the Message List Management window and select the following.

1) Make sure you have a good link to a connected ViewMarq display.
2) Make a new Message List or open an existing Message List.
3) Open the Message List Management window.
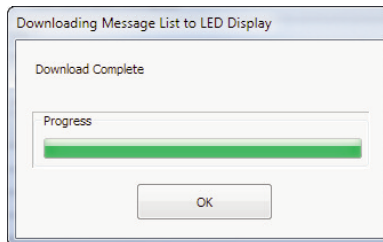4) Select the Download Message List Toolbar Button and select Yes in the Message List Download dialog box to confirm the Message List overwrite.

Triggering Messages in the Display

Messages can be triggered to display on your ViewMarq two different ways.

1) <DO n> ASCII command where "n" is the Message Number.
2) Discrete Inputs on the back of the display (if enabled in LED Display Configuration).

Using the <DO n> command

From the PLC an ASCII String can be sent to the ViewMarq to trigger a stored message. For example, if the following message is stored as Message 6 in ViewMarq Node 1:

`<CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N><CS 0><GRN><T>Hello World</T>`

The Command String from the PLC to trigger this message is:

`<ID 1><DO 6>`

Using the Discrete Inputs

There are two modes for triggering the discrete inputs depending on how the ViewMarq display is configured

1) Single Input - up to eight messages
2) Multiple Input - up to 200 messages

Using the last example of a message stored in as Message 6.

1) In Single Input Mode – Turn on input 6
2) In Multiple Input Mode – Turn on inputs 2 and 3

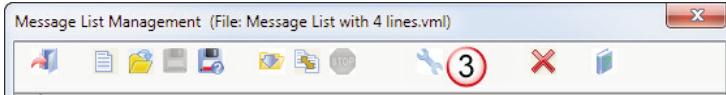The maximum rate at which messages can be triggered is 500ms per message.

NOTE: *See Chapters 2 and 3 for more information on Discrete input specifications and connections.*

Testing a Message List

The Message List can be tested in the ViewMarq Display, or on the Simulator or both at the same time.

1) Open or create a new Message List.

2) Make sure you have a good link to a ViewMarq LED display and download the Message List to be tested to the display.

3) From the Message List Management window, select the Test Message List toolbar button.
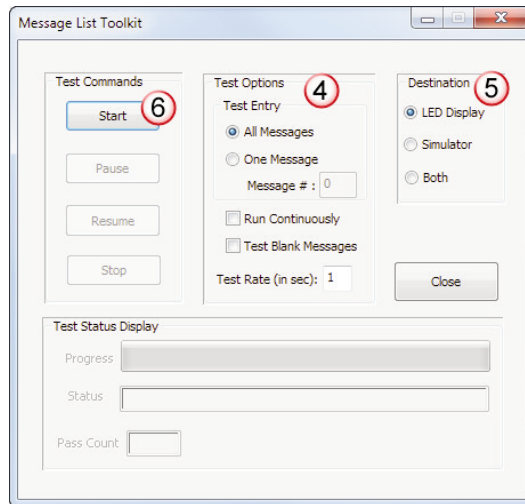
4) Select the desired Test Options.
   - a. Run Continuous - This selection will loop through the message list indefinitely. This is only available when the destination is the LED display.
   - b. Test Blank Messages - When selected the software will attempt to test all 200 messages to see whether a string is stored or not. For blank messages an E19 error will display on the ViewMarq display.

5) Select the Destination for the test.
   - a. LED Display
   - b. Simulator
   - c. Both

6) Select Start

NOTE: *The LED Display will show the messages in memory. The Simulator will show the messages in the open Message List. If these do not match, you will get unexpected results. Testing Blank Messages will cause an E19 Error to be displayed.*

Compare Message Lists

The Message List in the ViewMarq display may be compared to any list that has been saved.
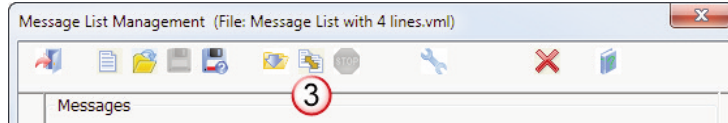
1) Open or create a new Message List.

2) Make sure you have a good link to a ViewMarq LED display.

3) From the Message List Management window, select the Compare Message List toolbar button.
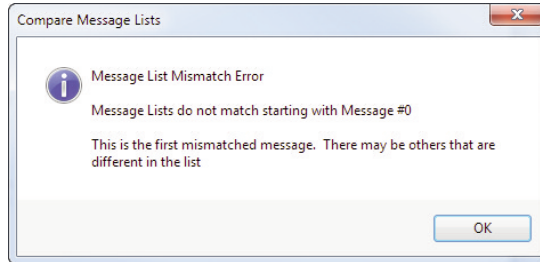
4) When the compare operation reaches a message in the display that does not match the same message in the open list, it will stop and indicate the first message that is different.

Warning: If the Message List in the Display was downloaded with Byte Swap selected on the Link, but the Link used during the compare is not Byte Swapped, the Compare will fail at Message #0.

10

Backup a Message List to a File

Any Message List in a display can be backed up as a Tab Delimited TXT file and later restored to the display.
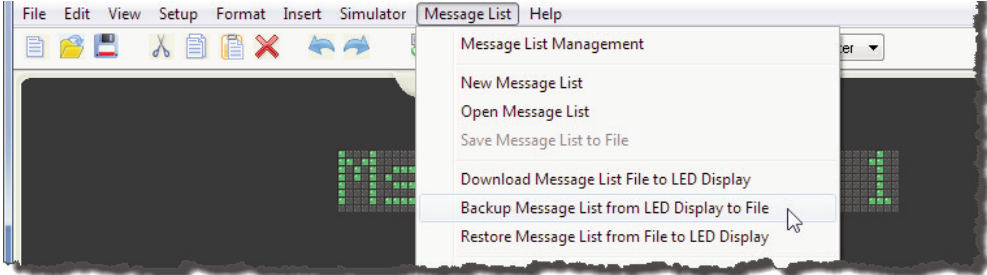
1) Make sure you have a good link to a ViewMarq LED display.

2) From the pulldown menu, select Message List, Backup Message List from LED Display to File.
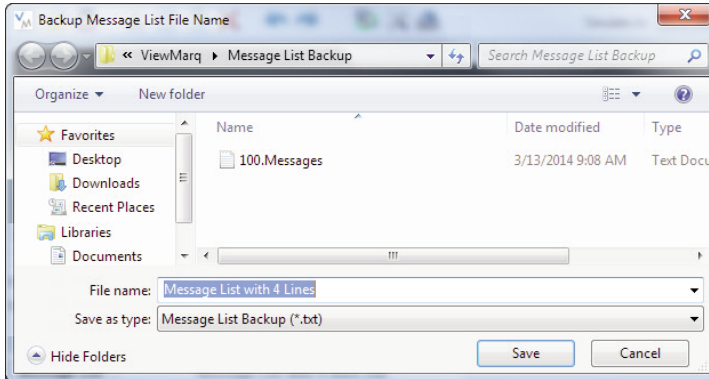
3) Enter a filename and click the Save button.

4) The Backing up Message List from LED Display progress dialog will open to show the progress of the backup. Click OK when backup is complete.

The resulting .txt file contains the:

1) Message number

2) ASCII string

NOTE: *Backup Message Lists cannot be opened in the ViewMarq Software for editing. They can only be restored directly to the display.*

Restore a Message List File to a ViewMarq Display

A Backup Message List can be restored directly to a connected LED Display.
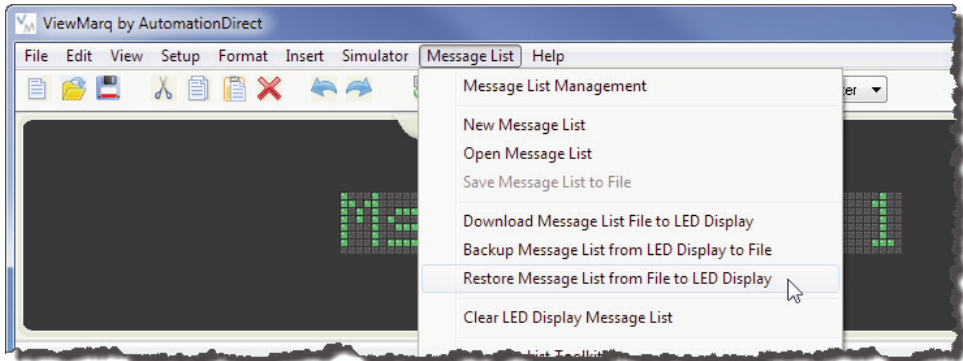
1) Make sure you have a good link to a ViewMarq LED display.

2) From the pulldown menu, select Message List, Restore Message List from File to LED Display.
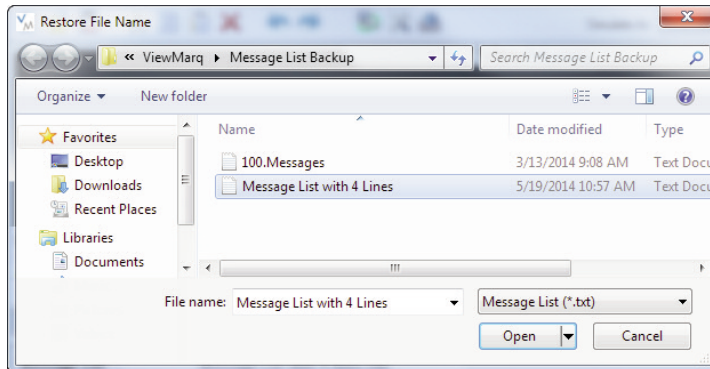
3) Select the file to restore and click the Open button.

4) The Restoring Message List to LED Display progress dialog will open to show the progress of the backup.

5) Click OK when restore is complete.

Clear Message List in the LED Display
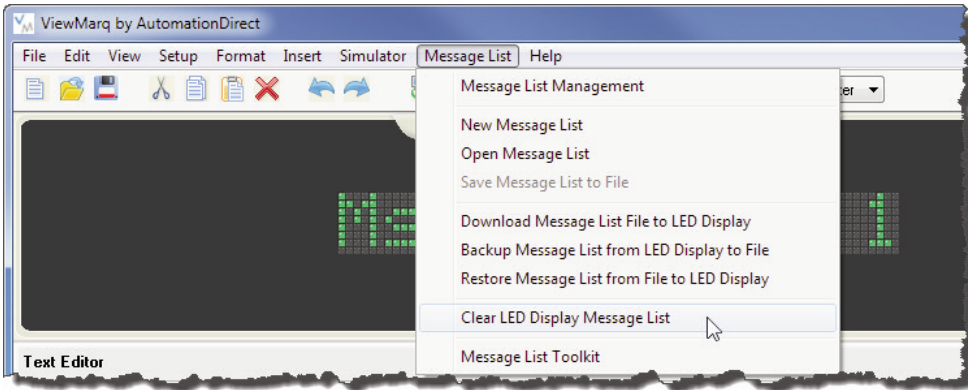
1) Make sure you have a good link to a ViewMarq LED display.

2) From the pulldown menu, select Message List, Clear LED Display Message List.

Format of Stored Messages

The above message is stored as a ViewMarq message on the PC as:

```
<ID 1><CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N>
<CS0><GRN><T>Hello World</T>
```

It is stored in the ViewMarq display as:

```
<CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N>
<CS0><GRN><T>Hello World</T>
```

The **<ID 1>** is stripped off since it is an addressing command that is no longer needed when the message is stored in a particular ViewMarq display.

VIEWMARQ ASCII COMMAND SPECIFICATIONS

APPENDIX A

In This Appendix...

A

ViewMarq Commands

ViewMarq Commands	
Format	**Description**
<ID #>	Specifies the SystemID of the displays that should process the command, always first command in a command string
<CLR>	Clear display
Text Display – Cause information to be displayed	
<T>Text</T>	Put Text on the display at the current position
<STR N L>	Put String Variable N on the display L characters wide
<DEC N L DP>	Put Numeric Variable N on the display using L digits with implied decimal place DP digits with leading spaces
<DECZ N L DP>	Put Numeric Variable N on the display using L digits with implied decimal place DP digits with leading zeros
Text Parameters – Changes the way Text is displayed	
<WIN ox oy sx sy>	Define a window position and size
<POS x y>	Set Position to X Y from Upper Left Corner of display
<OFF x y>	Set Position to X Y from Upper Left Corner of window
<CS n>	Select Character Set n (0-9)
<GRN>	Set color to Green
<RED>	Set color to Red
<AMB>	Set color to Amber
<BL x>	Blink Rate x – S=Slow, M=Medium, F=Fast, N=None
<S x>	Scroll Speed x – S=Slow, M=Medium, F=Fast
<LJ>	Text Effect - Left Justify
<CJ>	Text Effect - Center Justify
<RJ>	Text Effect - Right Justify
<SL>	Text Effect - Scroll Left
<SR>	Text Effect - Scroll Right
<SU>	Text Effect - Scroll Up
<SD>	Text Effect - Scroll Down
<SE PP n>	SE = Scroll Direction, PP = Pause Position, n = Pause time Example: <SL LJ 1> = Scroll Left and Pause Left Justified for 1 sec.
Variable Update	
<SETS n>value</SETS>	Set String Variable N to 'value'
<SETV n value>	Set Numeric Variable N to value
Advanced Commands	
<VER>	Put FirmwareVersion on display at current position
<NAME>	Put Display Name on display at current position
<IPADDR>	Put Ethernet Port IP Address on display at current position
<PORTPAR n>	Put Serial Port Paramters for Port n on display at current position
<STATUS>	Return the status of the last command executed on the current Port
<VERSION>	Return current Version String on the current Port
<MTN n>	Used to Test LEDs: n = 0) Green, 1) Red, 2) Amber 3) Pattern

ASCII String Commands

The ViewMarq LED message display utilizes ASCII Strings. The ASCII String contains text to be displayed and commands and parameters that determine how the text will be formatted such as the Character Set/Size, Color, Scrolling, Blinking, etc. The format of the text, commands and parameters are described in this section.

The ViewMarq receives these ASCII strings from an external device such as a PC or a PLC through one of the serial ports or the Ethernet port.

External Device

The ViewMarq is a slave device. It does not request data from other devices, it only receives it. So when a properly formatted string is detected on one of the communication ports and the <ID #> of the message matches the ID of the particular ViewMarq display, then the message in the ASCII String that follows will be displayed as formatted by the ASCII String.

ASCII String format

In order for the ViewMarq message display to understand the ASCII String, it must be formatted as described below.

One line format:

The one line format is the way the string is sent to the ViewMarq. The order of the formatting commands does not matter so long as they precede the text or variable (<T>, STR, DEC)commands.

<ID #><CLR><WIN OX OY EX EY><POS X Y><LJ><BL #><CS #><GRN><T>Message to Display**</T>CR**Breakdown

<ID #>	;Display ID
<CLR>	;Clear Display
<WIN OX OY EX EY>	;Display Window
<POS X Y>	;Position
<LJ>	;Effect
<BL #>	;Blink speed
<CS #>	;Character Set
<GRN>	;Color
<T>Message to Display**</T>**	;Text Block
CR	;0x0D Carriage Return

A

Defaults

Each command or parameter has an implied default. Every String must have at least the **<ID>** and text (**<T></T>**) or set variable (**<SETV>** or **<SETS>**) commands. There is no default for **<CLR>**.

Window	**<WIN 0 0 287 31>**
Position	**<POS 0 0>**
Effect	**<LJ>**
Blink	**<BL N>**
Character Set	**<CS 0>**
Color	**<GRN>**

Short Format with implied defaults:

The two command strings below will display the same message.

<ID #><CLR><WIN 0 0 287 31><POS 0 0><LJ><BL N><CS 0><GRN><T>Message to Display**</T>**

<ID #><CLR><T>Message to Display**</T>**

Basic Commands

A

Module ID: `ID`

The Module ID command is used to identify the display that the ASCII string is targeting. If the string is received by a display that has an ID different from what is in the ID command, that display will ignore the string until it receives another ID command.

Multiple displays may be specified by listing multiple ID's.

Format: **`<ID #>`** or **`<ID#>`**

Required: Yes

Usage: **`<ID #><T>`**`Message to Display`**`</T>CR`**

Single Display

Example: Send "Hello World" to Display with ID = 1

`<ID 1><CLR><T>Hello World</T>`

Multiple Displays

Example: Send "Hello World" to displays with ID = 1, 3, 8, 9, 10

NOTE: *The ID numbers can be in any order.*

`<ID 1 3 8 9 10><CLR><T>Hello World</T>`

Or send "Hello World" to a range of displays, 1 through 10

NOTE: *Ascending order only*

`<ID 1-10><CLR><T>Hello World</T>`

Broadcast to all displays

Example: Send "Hello World" to all displays on the network, use ID = 0

`<ID 0><CLR><T>Hello World</T>`

NOTE: *This is only useful using the ASCII protocol over RS-485.*

A

Clear: `CLR`

It is good practice to clear the display first to prevent overlapping text. The Clear command is used to clear an entire display.

>   Format: **`<CLR>`**

>   Required: No

>   Usage: **`<ID #><CLR>CR`**

Example 1: To clear a single display designate a single panel with the ID command

>   **`<ID 1>`**_**`<CLR>`**_

Example 2: To clear all signs, use the ID broadcast

>   **`<ID 0>`**_**`<CLR>`**_

> **NOTE:** If the CLR command is not used, the display buffer will eventually fill and no new message will be displayed. If "Display Syntax Error" is enabled a syntax error E6 will be displayed.

Text: `T`

The Text commands are used to define the text for the string. Everything between the begin text and end text commands will be displayed on the message display.

>   Format:

>   Begin Text: **`<T>`**

>   End Text: **`</T>`**

>   Required: Yes

>   Usage: **`<ID #><T>`**`Message to Display`**`</T>`**`CR`

Example 1: Display the text "Hello World" on a 1 x 12 character display.

>   **`<ID 1><CLR>`**_**`<T>`**_**`Hello World`**_**`</T>`**_

> **Attention!: `<T>` is a display command. There is a maximum of 31 display commands that can be used on the ViewMarq display at one time. More than 31 display commands will result in an error E6. Additional display commands are listed on Page A-19**

A

Window: `WIN`

The WIN command defines the display area of the text blocks that follow. The WIN command may be used to create multiple display areas on the message display.

The Window corners are designated in pixels, or LED's. In the example below, of a 2 x 12 character display...

> **OX** = 0
>
> **OY** = 0
>
> **EX** = 71
>
> **EY** = 15

Format: **`<WIN OX OY EX EY>`**

Required: No

Default: **`<WIN 0 0 X' Y'>`**

X' and Y' are the extreme End of Line and Bottom of the display respectively. When EX and EY are set to 31 and 287 respectively, the display will use the extreme far right edge and bottom respectively. In this way, a string meant to cover the entire display will work with any size message display without changing the string.

Usage: **`<ID #><WIN OX OY EX EY><POS X Y><T>Message to Display</T>CR`**

A

Single Window

Example 1: Display the text "Hello World" in a Window that covers the entire display area. Note that because the EX and EY are set to extreme values, this will work with any size message display.

> **<ID 1><CLR>_<WIN 0 0 287 31>_<POS 0 0><LJ><BL N><CS 0><GRN><T>Hello World</T>**
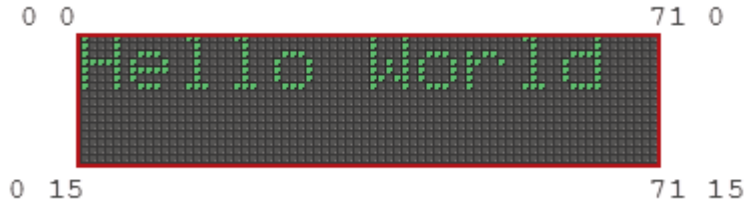
Window for each line

Example 2: Display the text "Line One" on Line 1 Scrolling Right and the text "Line Two" on Line 2 Scrolling Left. This is accomplished in the ViewMarq Software.

> **NOTE:** *A window may contain only one scrolling "Effect", i.e. Scroll Left* **<SL>***, Scroll Right* **<SR>***, Scroll Up* **<SU>***, Scroll Down* **<SD>***. To have two different effects for the two different lines, they must reside in two different Windows.*

> **<ID 1><CLR>_<WIN 0 0 287   7>_<POS 0 0><SR><S S><BL N><CS 0><GRN><T>Line One</T>_<WIN 0 8 287 15>_<POS 0 8><SL><S S><BL N><CS 0><GRN><T>Line Two</T>**

Windows side by side

Example 3: Display the text "Right" Scrolling Right in a display window that covers the left half of the display and the text "Left " Scrolling Left in a display window that covers the right half of the display.

NOTE: *A window may contain only one scrolling "Effect", i.e. Scroll Left* **<SL>**, *Scroll Right* **<SR>**, *Scroll Up* **<SU>**, *Scroll Down* **<SD>**. *To have two different effects for the two different lines, they must reside in two different Windows.*

```
<ID 1><CLR><WIN 0 0 35 31><POS 0 0><SR><S S><BL N><CS
3><GRN><T>Right</T><WIN 36 0 287 31><POS 32 0><SL><S
S><BL N><CS 3><GRN><T>Left </T>
```

A
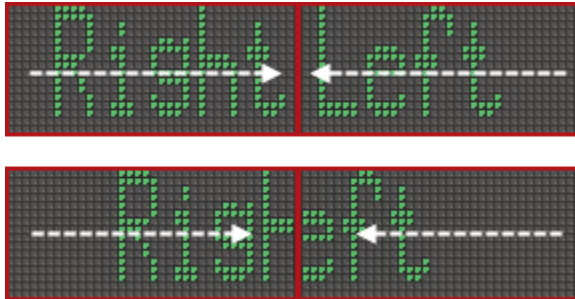
Position: `POS`

The Position command is used to assign the starting point of the Text Block on the display. The position is designated in pixels, or LED's. Below is an example of the LED positions on a 1 x 12 character display.

The default position if XY is not specified is (0 0). Notice this is the physical upper left corner of the display. Also, characters are anchored at their upper left corner.

Format: **`<POS X Y>`**

Required: No

Default: **`<POS 0 0>`**

Usage: **`<ID #><POS X Y><T>Message to Display</T>CR`**

Example 1: Display the text "Hello World" 12 pixels to the right of 0 on a 1 x 12 character display.

    `<ID 1><CLR><POS 12 0><T>Hello World</T>`

Example 2: Display the text "Hello World" 12 pixels to the right of 0 and 4 pixels down from 0 on a 2 x 12 character display.

    `<ID 1><CLR><POS 12 4><T>Hello World</T>`

Negative XY values

The X and Y values may be negative in order to position a text block off the edge of the display. This would be handy when used with scrolling text.

Example: Display the text "Hello World" 2 pixels up from 0 on a 1 x 12 character display.

    `<ID 1><CLR><POS 0 -2><T>Hello World</T>`

Multiple POS commands

Multiple POS commands within a ID block are used to make multiline messages.

Example 1: Display the two lines of text "Hello" and "World" a 2 x 12 character display.

> **<ID 1><CLR>_<POS 0 0>_<T>Hello</T>_<POS 0 8>_<T>World</T>**

Or

> **<ID 1><CLR><T>Hello</T>_<POS 0 8>_<T>World</T>**
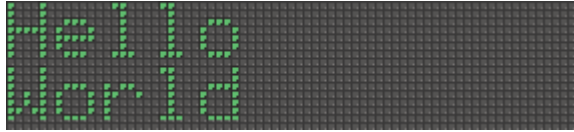
Example 2: Display the two lines of text "Hello" and "World" on a 1 x 12 character display.

> **<ID 1><CLR>_<POS 0 0>_<T>Hello</T>_<POS 36 0>_<T>World</T>**

Offset: OFF

The Offset command is very similar to the Position command. It is the relative starting point of the Text Block in a <u>Window designated by the WIN command.</u> The upper left corner of the window serves as the origin, instead of the upper left corner of the physical display. The offset is designated in pixels, or LED's. The default position if XY is not specified is (0 0).

Format: **<OFF X Y>**

Required: No

Default: **<OFF 0 0>**

Usage: **<ID #><WIN 0 0 71 31><OFF X Y><T>Message to Display</T>CR**

Example: Display the text "Hello World" on a 2 x 12 character display in window 2, 3, 71, 31 with an offset of 2, 2. The text is positioned at 4, 5 (2+2, 3+2).

> **<ID 1><CLR>_<WIN 2 3 71 31><OFF 2 2>_<T>Hello World</T>**

A

Character Set: **CS**

The Character Set command is used to select any of 11 character sets that are loaded in the display. The default character sets loaded are all English in each size as shown below.

Character Set No.	Size	Description
0	5x7	2 inch - 1 line
1	5x8	2 inch - 1 line - Full Line
2	5x5	1-1/4 inch - Compact
3	5x14	4 inch - 2 line Narrow
4	10x14	4 inch - 2 line
5	10x16	4 inch - 2 line - Full Line
6	10x21	6 inch - 3 line Narrow
7	15x21	6 inch - 3 line
8	15x28	8 inch Narrow
9	20x28	8 inch - 4 line
10	23x32	8 inch - 4 line - Full Line

Format: **<CS #>**

Required: No

Default: **<CS 0>**

Usage: **<ID #><CLR><CS #><T>Message to Display</T>CR**

The Character Set is retentive within one ID command across multiple Text Blocks until changed. In order to have a different font used within the same string, it must be specified.

Example: Display the text "Hello World" in two different fonts on a 2 line x 12 character display.

> **<ID 1><CLR>_<CS 3>_<T>Hello</T>_<CS 0>_<T> World</T>**

A

Color:

The Color command is used to select one of three colors for the text to be displayed. The default color is green.

Format: **<GRN>, <RED>, <AMB>**

Required: No

Default: **<GRN>**

Usage: **<ID #><CLR><RED><T>Message to Display</T>CR**

The Color is retentive within one ID command across multiple Text Blocks until changed. In order to have a different color used within the same string, it must be specified.

Example: Display the text "Hello World" in two colors on a 1 x 12 character display

> **<ID 1><CLR><u>*<RED>*</u><T>Hello</T><u>*<GRN>*</u><T> World</T>**

Blink: BL

The Blink command can be used to attract attention to the displayed message. It is used to cause the text to blink or not and also to select the rate at which it blinks. The default is None.

N	None
S	Blink Slow
M	Blink Medium
F	Blink Fast

Format: **<BL #>**

Required: No

Default: **<BL N>**

Usage: **<ID #><CLR><BL #><T>Message to Display</T>CR**

Like Font and Color, Blink is retentive within one ID command across multiple Text Blocks until changed. In order to have a portion of the text blinking and the following text not, the Blink must be turned off.

Example: Display the text "Hello World" with "Hello" blinking slow and "World" not blinking on a 1 x 12 character display

> **<ID 1><CLR><u>*<BL S>*</u><T>Hello</T><u>*<BL N>*</u><T> World</T>**

A

Effects

The Effect commands determine how the text will be aligned in the display and/or how it will enter the display area. The default is Left Justified <LJ>.

NOTE: *Each Window can have only one scrolling effect.*

Format:

<LJ> Left Justify
<CJ> Center Justify
<RJ> Right Justify

<SL> Scroll Left

<SL LJ #>	Scroll Left and Pause for # seconds Left Justified
<SL CJ #>	Scroll Left and Pause for # seconds Center Justified
<SL RJ #>	Scroll Left and Pause for # seconds Right Justified

<SR> Scroll Right

<SR LJ #>	Scroll Right and Pause for # seconds Left Justified
<SR CJ #>	Scroll Right and Pause for # seconds Center Justified
<SR RJ #>	Scroll Right and Pause for # seconds Right Justified

<SU> Scroll Up

<SU TOP #>	Scroll Up and Pause for # seconds Top Justified
<SU BOT #>	Scroll Up and Pause for # seconds Bottom Justified

<SD> Scroll Down

<SD TOP #>	Scroll Down and Pause for # seconds Top Justified
<SD BOT #>	Scroll Down and Pause for # seconds Bottom Justified

Required: No

Default: **<LJ>**

Usage: **<ID #><CLR><LJ><T>Message to Display</T>CR**

Example: Display the text "Hello" Center Justified on line 1 and the text "World" scrolling left on line 2 pausing center justified for 5 seconds on a 2 line x 12 character display.

```
<ID 1><CLR><WIN 0 0 287  7><POS 0 0><CJ><BL N><CS 0>
<GRN><T>Hello</T><WIN 0 8 287 31><POS 0 8><SL CJ 5>
<S S><BL N><CS 0><GRN><T>World</T>
```


Notice the two Windows to support the two different effects.

Pause

There is no specific Pause command. It is an implied command when a Justification command is added to a Scroll Command. Along with the Justification Command is a time to pause in seconds. Specifying 0 will cause the scroll to pause indefinitely.

LJ, CJ and **RJ** are used with the Scroll Left and Scroll Right Commands.

TOP and **BOT** are used with the Scroll Up and Scroll Down Commands

Format: **<SL LJ #>**

Usage: **<ID #><CLR><SU TOP 5><T>Message to Display</T>CR**

Scroll Speed: <S #>

The Scroll Speed command is used in conjunction with the Scroll commands. The default speed is Medium.

S	Slow
M	Medium (default)
F	Fast

Format: **<S #>**

Usage: **<ID #><CLR><SL><S #><T>Message to Display</T>CR**

NOTE: Only one scroll speed is effective on the **display** at one time. It is not possible to have two windows scrolling at two different speeds. The last *<S #>* command received by the display will be the speed that is used.

A

Embedded Variables

Embedded variables are used to update data on the display without having to send an entire message. Two different variables are available, the String variable and the Numeric variable.

The Variable values are volatile and are cleared when the display is powered off.

String Variables: STR

The STR command is used to display a string variable that is easily updated with the SETS command. The String variable has two parameters, String Number and Length.

There can be up to 16 strings of up to 100 characters each.

Format: **\<STR N L>**

> **N** – String Number
> **L** – Field Length in characters – Left Justified

Usage: **\<ID #>\<CLR>\<T>Message to Display\</T>\<STR N L>CR**

Set String: SETS

The SETS command is used to set the value of a String variable.

Format: **\<SETS N>string\</SETS>**

> **N** – String Number

Example 1: Display the text "Hello" followed by the value of String 1. Reserve 12 characters for String 1.

> **\<ID 1>\<CLR>\<T>Hello \</T>\<STR 1 12>**

Set String 1 to " World"

\<SETS 1>World\</SETS>

Set String 1 to " Dolly"

\<SETS 1>Dolly\</SETS>

> **NOTE:** The display format options such as color or character set for variables is set in the static text string where the \<DEC> and \<STR> commands are used; that is, color or character set commands do not work with, \<SETS> or \<SETV> commands.

Decimal Variables: `DEC` and `DECZ`

The DEC and DECZ commands are used to display a decimal variable that is easily updated with the SETV command. The Decimal variable has three parameters; Decimal Number, Decimal Length and Decimal Place.

`DEC` – Format with Leading Spaces

A 4 digit variable with 2 decimal places that is set to zero will be displayed as 0.00

`DECZ` – Format with Leading Zeros

A 4 digit variable with 2 decimal places that is set to zero will be displayed as 00.00

There can be up to 32 variables of 16 digits each.

Format: **`<DEC N L DP>`**
    `<DECZ N L DP>`

`N` – String Number (1 – 32)
`L` – Number of Digits to Display (1 – 11)
`DP` – Decimal Place (0 – 10)

Usage: **`<ID #><CLR><T>Message to Display</T><DEC N L DP>CR`**

Set String: SETV

The SETV command is used to set the value of a Decimal variable.

Format: **`<SETV N [decimals]>`**

`N` – Decimal Number

Example 1: Display the text "Temp °C: " followed by the value of Decimal 1. Reserve 4 characters for Decimal 1 with an implied decimal place of 1.

`<ID 1><CLR><T>Temp`C:</T><DEC 1 4 1>`

Set Decimal 1 "5678"

`<SETV 1 5678>`

Set Decimal 1 to " 987"

`<SETV 1 987>`

> **NOTE:** *The display format options such as color or character set for variables is set in the static text string where the <DEC> and <STR> commands are used; that is, color or character set commands do not work with, <SETS> or <SETV> commands.*

Advanced Commands

A

Maintenance: `MTN`

The Maintenance Command is used to test the LEDs on the display. The four different commands give you the ability to run four different test patterns.

Format: `<ID #><MTN N>CR`

`0)`	Display Test Green
`1)`	Display Test Red
`2)`	Display Test Amber
`3)`	Advanced Test Pattern

Turn off by using the CLR command

Advanced format: `<ID #><MTN # X Y SizeX SizeY>`

Display test in a portion of the display starting at X,Y SizeX wide and SizeY High.

Error: If an invalid option is selected the Command Status "Invalid Option must be 0-3" will be returned.

Serial Diagnostics: `DIAG`

Format: `<ID #><DIAG N>CR`

If N is 1 it will toggle the lower left LED each time a string it received as an ASCII command, via Serial or Modbus.

Display Commands

The following commands do not display text messages designed by the user on the display. They display other information on the display that may be helpful to the user.

Version Display: **VER**

Format: **<VER>**

Usage: **<ID #><CLR><VER>CR**

Display the Firmware Version String on the display

Display Name: **NAME**

Format: **<NAME>**

Usage: **<ID #><CLR><NAME>CR**

Display the Display-Name String on the display. The display name is set using the ViewMarq Software.

IP Address: **IPADDR**

Format: **<IPADDR>**

Usage: **<ID #><CLR><IPADDR>CR**

This will display the current IP Address of the display or "Acquiring IP…" if DHCP is actively attempting to obtain an IP address or " ** NO LINK ** " if no network cable is inserted.

Port Parameters: **PORTPAR**

Format: **<PORTPAR #>**

> **1** – RS232 Port
>
> **2** – RS485 Port

Display the serial port parameters for port 1 (RS232) or port 2 (RS485).

Usage: **<ID #><CLR><PORTPAR #>CR**

The displayed string will be in the form "38400 8N1" for 38.4k baud, 8 bits, no parity and 1 stop bit.

Example 1: Display the Version, Display Name, IP Address and both Serial Port's Parameters all Scrolling Slow across the display.

> **<ID #><CLR><SL><S S><VER><NAME><IPADDR><PORTPAR 1>**
> **<PORTPAR 2>**

ATTENTION

Attention!: There is a maximum of 31 display commands that can be used on the ViewMarq display at one time. More than 31 display commands will result in an error E6. <T> is also a display command.

Non Display Commands

The following commands do not cause any text to be displayed on the message display. They cause data to be transmitted out the serial port on which they were received.

Command Status **<STATUS>**

Format: **<STATUS>**

If issued via a serial port the reply will be the reply that was generated due to the previous command that was processed. This can be used at any time, but most likely to be used after a command is sent to many displays to make sure it was received correctly.

The reply on the Serial Port will be "NULL", or "NO MESSAGE" if the Command Status buffer is empty or the Command Status will be sent. ("OK" etc)

Firmware Version **<VERSION>**

Format: **<VERSION>**

This command will cause the current Version String to be sent out on the serial port if the request came in on the serial port.

MODBUS REGISTERS

APPENDIX
B

In This Appendix...

Modbus Registers

Modbus Registers			
Modbus Register	Length (Words)	Description	Read / Write
400100	2	Variable #1 High Word	Read / Write
400101		Variable #1 Low Word	Read / Write
400102	2	Variable #2 High Word	Read / Write
400103		Variable #2 Low Word	Read / Write
400104	2	Variable #3 High Word	Read / Write
400105		Variable #3 Low Word	Read / Write
		etc	
400160	2	Variable #31 High Word	Read / Write
400161		Variable #31 Low Word	Read / Write
400162	2	Variable #32 High Word	Read / Write
400163		Variable #32 Low Word	Read / Write
400200	50	String Variable #1 (100 Bytes)	Read / Write
400250	50	String Variable #2 (100 Bytes)	Read / Write
400300	50	String Variable #3(100 Bytes)	Read / Write
400350	50	String Variable #4 (100 Bytes)	Read / Write
400400	50	String Variable #5 (100 Bytes)	Read / Write
400450	50	String Variable #6 (100 Bytes)	Read / Write
400500	50	String Variable #7 (100 Bytes)	Read / Write
400550	50	String Variable #8 (100 Bytes)	Read / Write
400600	50	String Variable #9 (100 Bytes)	Read / Write
400650	50	String Variable #10 (100 Bytes)	Read / Write
400700	50	String Variable #11 (100 Bytes)	Read / Write
400750	50	String Variable #12 (100 Bytes)	Read / Write
400800	50	String Variable #13 (100 Bytes)	Read / Write
400850	50	String Variable #14 (100 Bytes)	Read / Write
400900	50	String Variable #15 (100 Bytes)	Read / Write
400950	50	String Variable #16 (100 Bytes)	Read / Write
411000	256	Command String Buffer (512 Bytes max)	Read / Write
411500	256	Status Buffer (512 Bytes max)	Read / Write

B

Modbus Registers Cont'd			
Modbus Register	**Length (Words)**	**Description**	**Read / Write**
445000	10	Display Model Number (16 char max)	Read
445010	10	ASCII Version String	Read
445020	10	Firmware Version String	Read
445050	3	Ethernet MAC Address	Read
445100	8	Customer Defined Display Name	Read
445108	1	Heartbeat Timeout (Seconds)	Read
445109	1	Option Flags (See Option Flags Table)	Read
445110	1	TCP/IP Port for Modbus (1-247)	Read
445111	1	TCP/IP Idle timeout (Seconds)	Read
445112	1	ID for ASCII Commands	Read
445200	2	IP Address	Read
445202	2	Subnet Mask	Read
445204	2	Default Gateway	Read
445206	2	DNS Server	Read
445300	1	Port 1 – Mode (RS232)	Read
445301	1	Port 1 – Modbus ID (1-247)	Read
445302	2	Port 1 – Baud Rate	Read
445304	1	Port 1 – Parity (0=N, 1=Odd, 2=Even)	Read
445305	1	Port 1 – Data Bits (7 or 8)	Read
445306	1	Port 1 – Response Delay (ms)	Read
445307	1	Port 1 – RTS On Delay (ms)	Read
445308	1	Port 1 – RTS Off Delay (ms)	Read
445400	1	Port 2 – Mode (RS485)	Read
445401	1	Port 2 – Modbus ID (1-247)	Read
445402	2	Port 2 – Baud Rate	Read
445404	1	Port 2 – Parity (0=N, 1=Odd, 2=Even)	Read
445405	1	Port 2 – Data Bits (7 or 8)	Read
445406	1	Port 2 – Response Delay (ms)	Read

Option Flags

B

Option Flags (Read Only)	
Bit of Word 445109	**Discription**
0x0001	Display Startup Banner
0x0008	Display Heartbeat Error on coms loss
0x0010	Disable ASCII Command Reply on Serial
0x0020	Enable DHCP for Ethernet Port
0x0040	Word Swap User Variables
0x0080	Byte Swap Strings
0x0100	Enable Diagnostics LED
0x0200	Orientation Fixed
0x0400	Fixed Upside Down
0x0800	Display Syntax Errors

Supported Modbus Function Codes

Supported Modbus Function Codes	
Function Code	**Discription**
03	Read Holding Registers
06	Write Single Holding Register
16	Write Multiple Holding Registers

VIEWMARQ CHARACTERS

APPENDIX
C

In This Appendix...

ViewMarq Character List

ViewMarq Characters					
Key	**Display**	**Key**	**Display**	**Key**	**Display**
!	!	A	A	a	a
"	"	B	B	b	b
#	#	C	C	c	c
$	$	D	D	d	d
%	%	E	E	e	e
&	&	F	F	f	f
'	°	G	G	g	g
(	(	H	H	h	h
)	)	I	I	i	i
*	*	J	J	j	j
+	+	K	K	k	k
,	,	L	L	l	l
-	-	M	M	m	m
.	.	N	N	n	n
/	/	O	O	o	o
0	0	P	P	p	p
1	1	Q	Q	q	q
2	2	R	R	r	r
3	3	S	S	s	s
4	4	T	T	t	t
5	5	U	U	u	u
6	6	V	V	v	v
7	7	W	W	w	w
8	8	X	X	x	x
9	9	Y	Y	y	y
:	:	Z	Z	z	z
;	;	[	[	{	{
<	<	\	\	|	|
=	=	]	]	}	}
>	>	^	^	~	~
?	?	_	_	`	.
[space]	[space]	@	@		