

Logix5000 Controllers ASCII Strings



Programming Manual

Catalog Numbers 1756-L1, 1756-L55, 1756-L61, 1756-L62, 1756-L63, 1769-L31, 1769-L32C, 1769-L32E, 1769-L35CR, 1769-L35E, 1789-L60, 1794-L34, PowerFlex 700S/SE

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at <http://literature.rockwellautomation.com>) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

WARNING 	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION 	Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence
SHOCK HAZARD 	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.
BURN HAZARD 	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.

Allen-Bradley, Rockwell Automation, and TechConnect are trademarks of Rockwell Automation, Inc.

Trademarks not belonging to Rockwell Automation are property of their respective companies.

Preface	Purpose of this Manual	5
	How to Use this Manual	5
	Chapter 1	
Communicating with an ASCII Device	Introduction	7
	Connect the ASCII Device	8
	Configure the Serial Port	9
	Configure the User Protocol	10
	Create String Data Types	11
	Read Characters from the Device	12
	Send Characters to the Device.	15
	Enter ASCII Characters	17
	Chapter 2	
Processing ASCII Characters	Introduction	19
	Extract a Part of a Bar Code	19
	Look Up a Bar Code	20
	Create the PRODUCT_INFO Data Type	21
	Search for the Characters	21
	Identify the Lane Number.	22
	Reject Bad Characters	22
	Enter the Product IDs and Lane Numbers	23
	Check the Bar Code Characters.	23
	Convert a Value	24
	Decode an ASCII Message	25
	Build a String	26
	ASCII Character Codes	29

Purpose of this Manual

This manual shows how to manipulate ASCII strings in Logix5000 controllers. This manual is one of a set of related manuals that show common procedures for programming and operating Logix5000 controllers. For a complete list of common procedures manuals, see the Logix 5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.

The term Logix5000 controller refers to any controller that is based on the Logix5000 operating system, such as:

- CompactLogix controllers
- ControlLogix controllers
- DriveLogix controllers
- FlexLogix controllers
- SoftLogix5800 controllers

How to Use this Manual

Some text is formatted differently from the rest of the text.

Text that is	Identifies	For example	Means
<i>italic</i>	the actual name of an item that you see on your screen or in an example	Right-click <i>User-Defined</i> ...	Right-click the item that is named User-Defined.
<i>courier</i>	information that you must supply based on your application (a variable)	Right-click <i>name_of_program</i> ...	You must identify the specific program in your application. Typically, it is a name or variable that you have defined.
enclosed in brackets	a keyboard key	Press [Enter].	Press the Enter key.

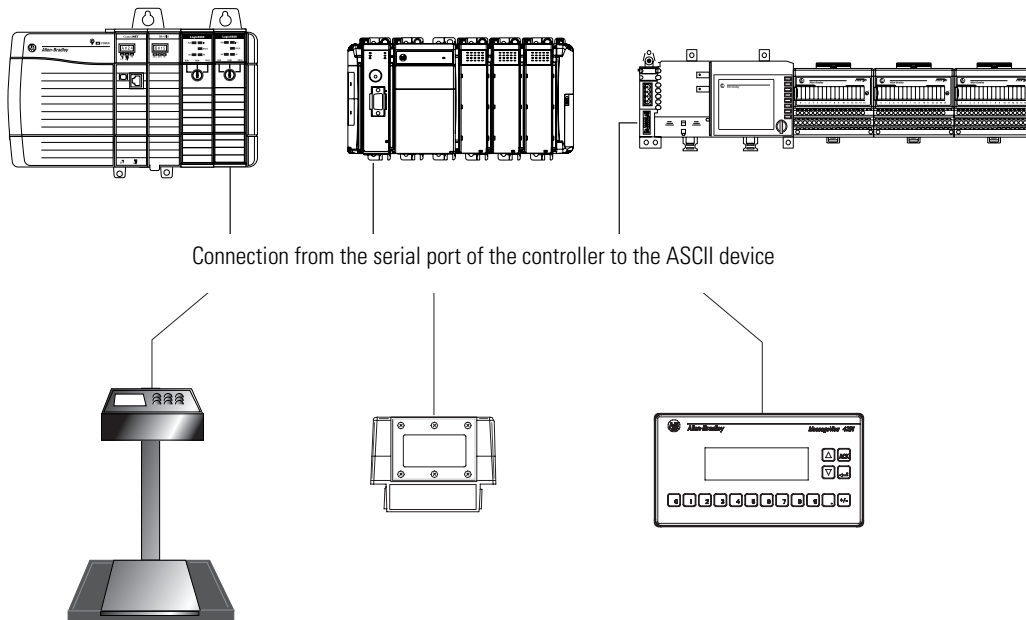
Notes:

Communicating with an ASCII Device

Introduction

You can exchange ASCII data with a device through the serial port of the controller. For example, you can use the serial port to:

- read ASCII characters from a weigh scale module or bar code reader.
- send and receive messages from an ASCII triggered device, such as a MessageView terminal.



42237

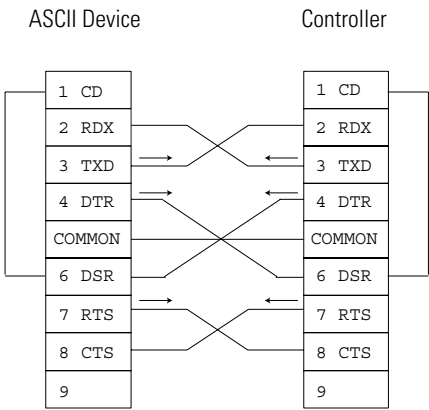
In addition to the controller serial port, firmware revision 3.1 and greater of the 1756-EWEB EtherNet/IP Web Server module supports a socket interface that lets Logix5000 controllers exchange ASCII data using TCP or UDP socket services. See the EtherNet/IP Web Server User Manual, publication ENET-UM0527, revision C or later.

Connect the ASCII Device

- 1. On the serial port of the ASCII device, determine which pins send signals and which pins receive signals.
- 2. Connect sending pins to corresponding receiving pins and attach jumpers.

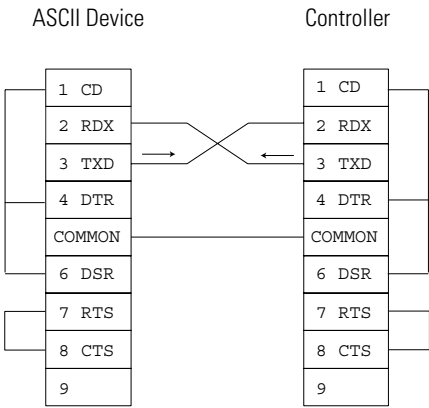
If the communications Then wire the connectors

Handshake



42231

Do not handshake

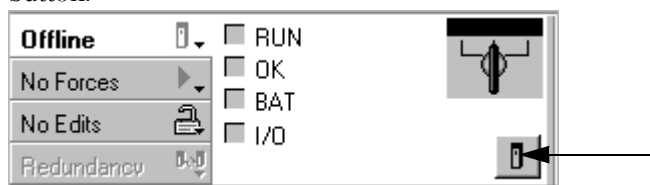


42232

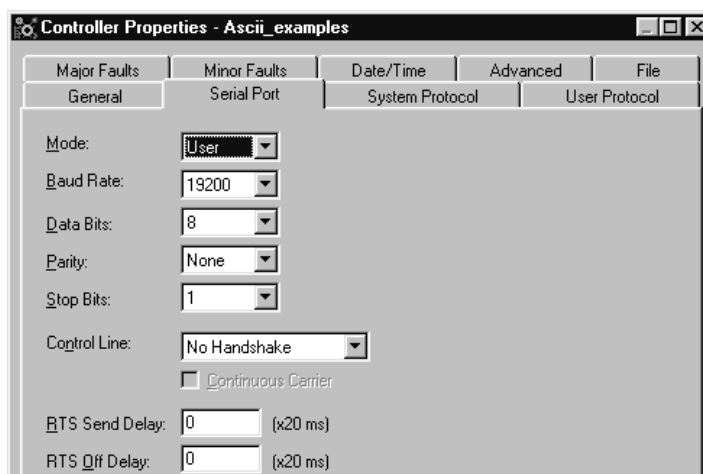
- 3. Attach the cable shield to both connectors.
- 4. Connect the cable to the controller and the ASCII device.

Configure the Serial Port

1. On the Online toolbar in the controller project, click the controller button.



2. Select the Serial Port tab.
3. Select User mode and enter the configuration settings for the serial port.



- Select the baud rate, data bits, parity, and stop bits.
- Select the Control Line option:

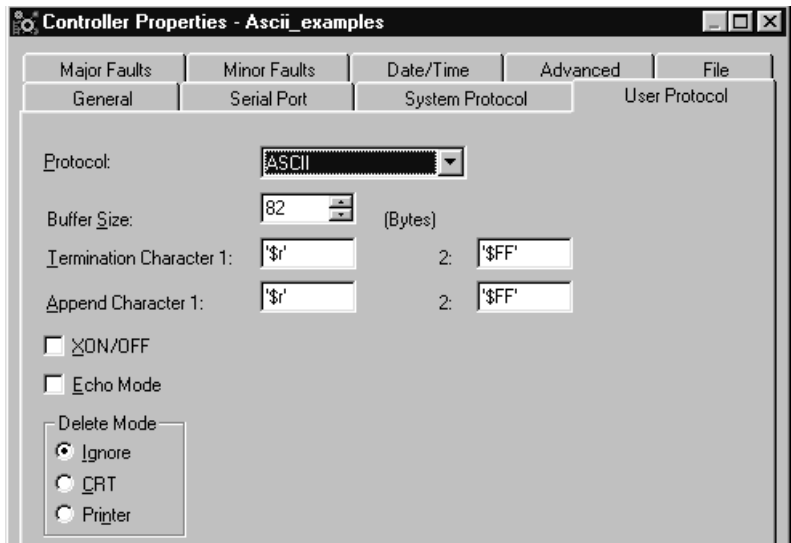
If	And	And this is the	Select	Then
You are not using a modem	→		No Handshaking	
You are using a modem	Both modems in a point-to-point link are full-duplex	→	Full Duplex	
	Master modem is full-duplex while slave modem is half-duplex	master controller. slave controller	Full Duplex Half Duplex	Check the Continuous Carrier check box.
	All modems in the system are half-duplex	→	Half Duplex	Clear the Continuous Carrier check box (default).

- For RTS Send Delay, enter the delay (in 20 ms units) between the time the RTS signal turns on (high) and the time that data is sent. For example, a value of 4 produces an 80 ms delay.
- For RTS Off Delay, enter the delay (in 20 ms units) between the time the last character is sent and the time that the RTS signal turns off (low).

4. Click Apply.

Configure the User Protocol

1. Select the User Protocol tab.



42252

- Enter a buffer size that is greater than or equal to the greatest number of characters in a transmission. (Twice the number of characters is a good guideline.)
- For ABL or ARL instructions, enter termination characters to mark the end of the data. For ASCII codes, see the back cover of this manual.

If the device sends	Then	Notes
One termination character	· In the Termination Character 1 text box, type the hexadecimal ASCII code for the first character. · In the Termination Character 2 text box, type \$FF.	For printable characters, such as 1 or A, type the character.
Two termination characters	In the Termination Character 1 and 2 text boxes, type the hexadecimal ASCII code for each character.	

- For AWA instruction, enter append characters. For ASCII codes, see the back cover of this manual.

To append	Then	Notes
One character	· In the Append Character 1 text box, type the hexadecimal ASCII code for the first character. · In the Append Character 2 text box, type \$FF.	For printable characters, such as 1 or A, type the character.
Two characters	In the Append Character 1 and 2 text boxes, type the hexadecimal ASCII code for each character.	

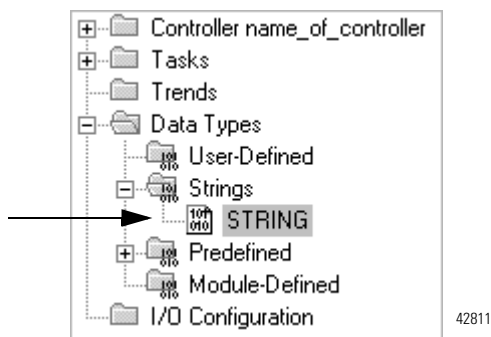
- If the ASCII device is configured for XON/XOFF flow control, select the XON/XOFF check box.
- If the ASCII device is a CRT or is pre-configured for half duplex transmission, select the Echo Mode check box.
- Select the Delete Mode:

If the ASCII device is	Select	Notes
CRT	CRT	· The DEL character (\$7F) and the character that precedes the DEL character are not sent to the destination. · If echo mode is selected and an ASCII instruction reads the DEL character, the echo returns three characters: BACKSPACE SPACE BACKSPACE (\$08 \$20 \$08).
Printer	Printer	· The DEL character (\$7F) and the character that precedes the DEL character are not sent to the destination. · If echo mode is selected and an ASCII instruction reads the DEL character, the echo returns two characters: / (\$2F) followed by the character that was deleted.
None of the above	Ignore	The DEL character (\$7F) is treated as any other character.

2. Click OK.

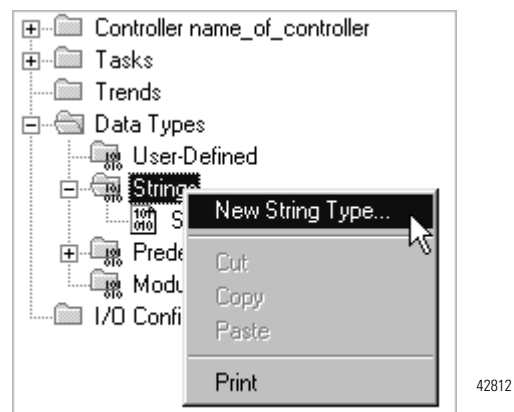
Create String Data Types

Store ASCII characters in tags that use a string data type.



You can use the default STRING data type. It stores up to 82 characters.

or

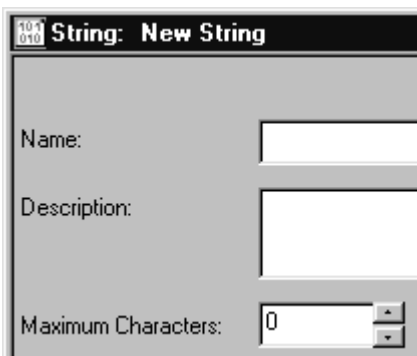


You can create a new string data type to store the number of characters that you define.

IMPORTANT

Use caution when you create a new string data type. If you later decide to change the size of the string data type, you may lose data in any tags that currently use that data type.

If you	Then
Make a string data type smaller	· The data is truncated. · The LEN is unchanged.
Make a string data type larger	The data and LEN is reset to zero.



42233

1. In the controller organizer, right-click Strings and choose New String Type...
2. Type a name for the data type.
3. Type the maximum number characters that this string data type will store.
4. Click OK.

Read Characters from the Device

As a general rule, before you read the buffer use an ACB or ABL instruction to verify that the buffer contains the required characters.

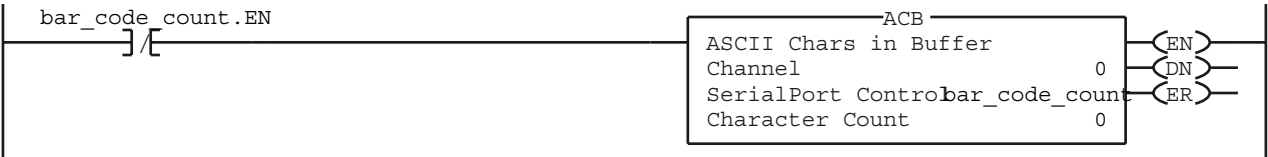
- An ARD or ARL instruction continues to read the buffer until the instruction reads the required characters.
- While an ARD or ARL instruction is reading the buffer, no other ASCII Serial Port instructions, except the ACL, can execute.
- Verifying that the buffer contains the required characters prevents the ARD or ARL from holding up the execution of other ASCII Serial Port instructions while the input device sends its data.

For additional information on ASCII Serial Port instructions, see Logix5000 Controllers General Instruction Set Reference Manual, publication 1756-RM003.

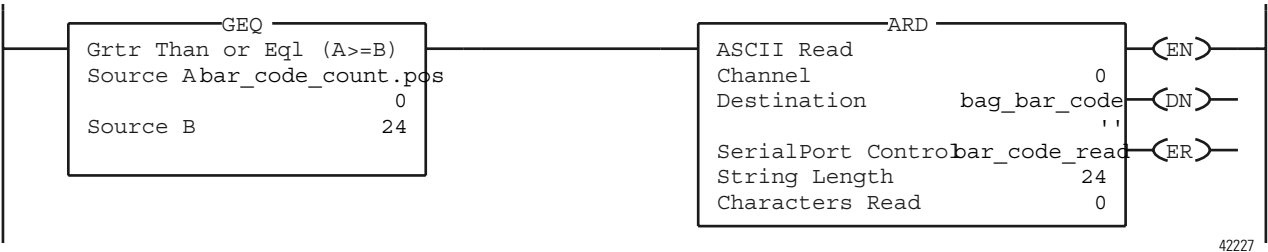
For example, the device sends a fixed number of characters, such as a bar code reader:

EXAMPLE

A bar code reader sends bar codes to the serial port (channel 0) of the controller. Each bar code contains 24 characters. To determine when the controller receives a bar code, the ACB instruction continuously counts the characters in the buffer.



When the buffer contains at least 24 characters, the controller has received a bar code. The ARD instruction moves the bar code to the `bag_bar_code` tag.



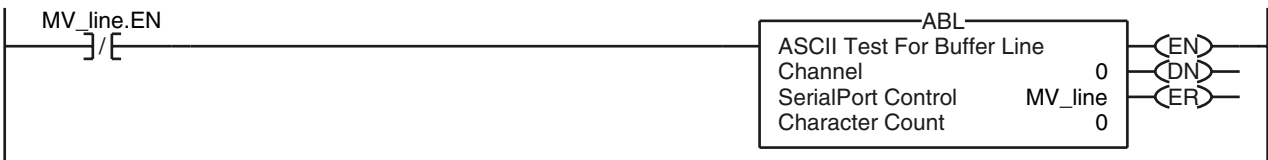
42227

For example, the device sends a variable number of characters, such as a message or display terminal.

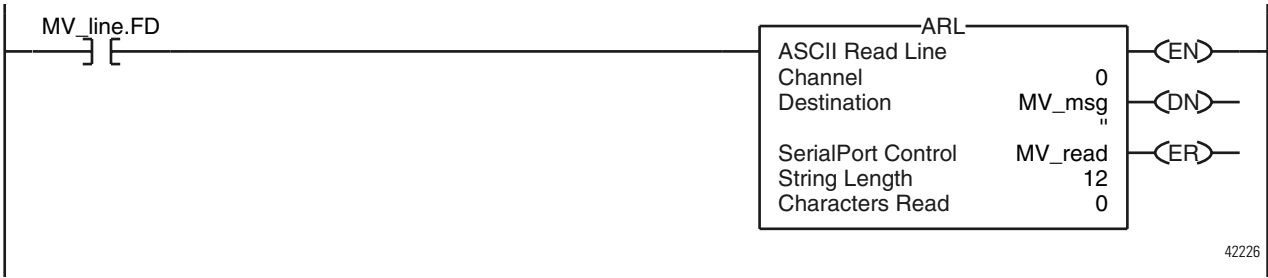
EXAMPLE

Continuously test the buffer for a message.

- Since each message ends in a carriage return (\$0D), the carriage return is configured as the termination character in the Controller Properties dialog box, User Protocol tab.
- When the ABL finds a carriage return, it sets the FD bit.



When the ABL instruction finds the carriage return (`MV_line.FD` is set), the controller removes the characters from the buffer, up to and including the carriage return, and places them in the `MV_msg` tag.



Send Characters to the Device

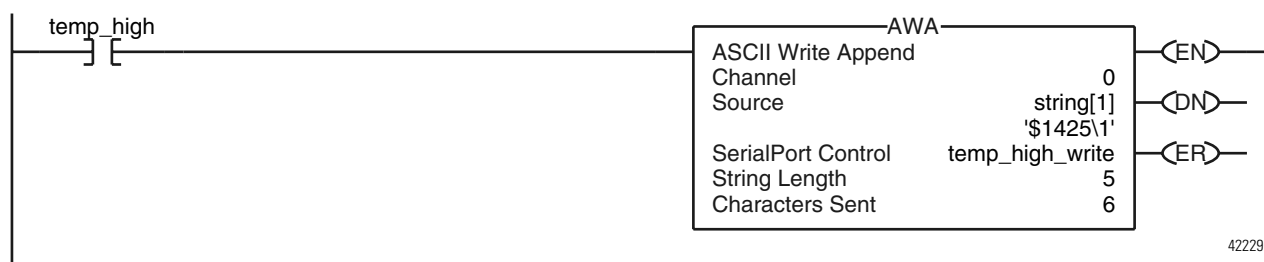
When you send characters to the device, you need to determine whether you will always send the same number of characters each time and whether you want to append terminations characters to the data.

For example, you always send the same number of characters and want to automatically append one or two characters to the end of the data.

EXAMPLE

When the temperature exceeds the high limit (temp_high is on), the AWA instruction sends five characters from the string[1] tag to a MessageView terminal.

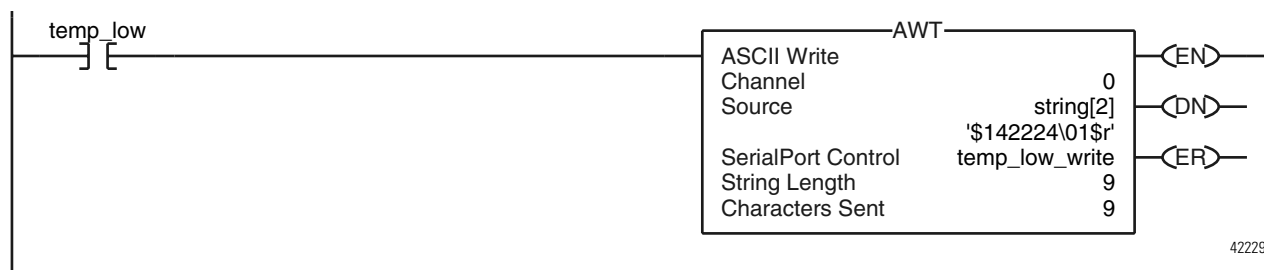
- The \$14 counts as one character. It is the hex code for the Ctrl-T character.
- The instruction also sends (appends) the characters defined in the user protocol. In this example, the AWA instruction sends a carriage return (\$0D), which marks the end of the message.



And then to always send the same number of characters:

EXAMPLE

When the temperature reaches the low limit (temp_low is on), the AWT instruction sends nine characters from the string[2] tag to a MessageView terminal. (The \$14 counts as one character. It is the hex code for the Ctrl-T character.)

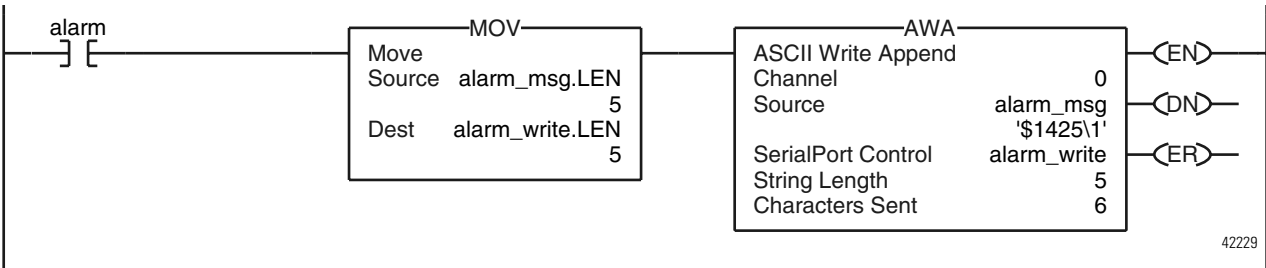


For example, you send a different number of characters each time and want to automatically append one or two characters to the end of the data:

EXAMPLE

When alarm is on, the AWA instruction sends the characters in alarm_msg and appends a termination character.

- Because the number of characters in alarm_msg varies, the rung first moves the length of alarm_msg (alarm_msg.LEN) to the length of the AWA instruction (alarm_write.LEN).
- In alarm_msg, the \$14 counts as one character. It is the hex code for the Ctrl-T character.

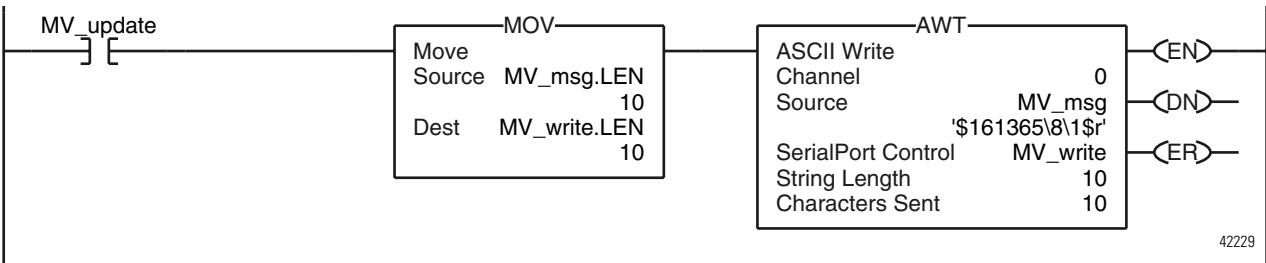


And then to send a different number of characters each time:

EXAMPLE

When MV_update is on, the AWT instruction sends the characters in MV_msg.

- Because the number of characters in MV_msg varies, the rung first moves the length of MV_msg (MV_msg.LEN) to the length of the AWT instruction (MV_write.LEN).
- In MV_msg, the \$16 counts as one character. It is the hex code for the Ctrl-V character.

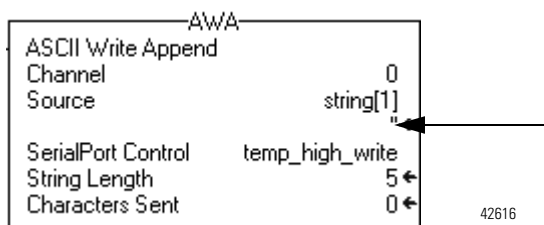


Enter ASCII Characters

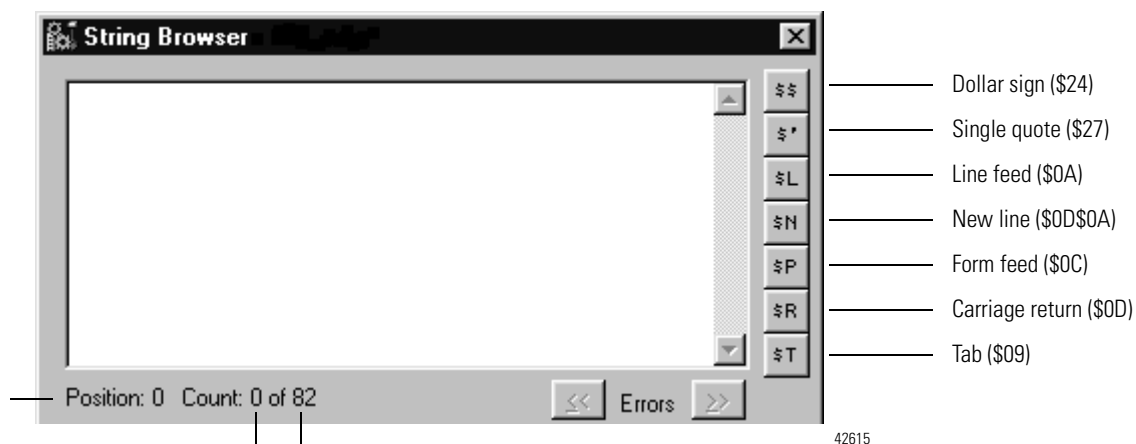
IMPORTANT

This String Browser window shows the characters up to the value of the LEN member of the string tag. The string tag may contain additional data, which the String Browser window does not show.

1. Double-click the value area of the Source.



A text entry box appears:



The number of characters that you see in the window. This is the same as the LEN member of the string tag.

The maximum number of characters that the string tag can hold.

2. Enter the characters for the string.
3. Click OK.

Notes:

Processing ASCII Characters

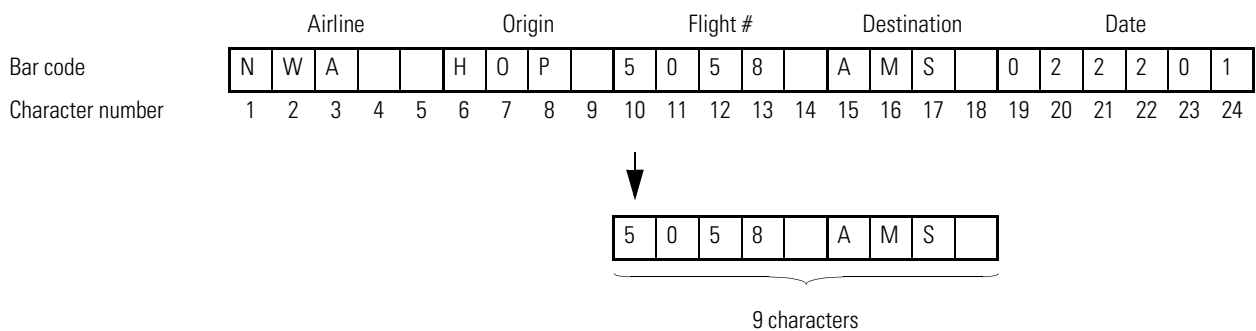
Introduction

You can process ASCII characters to:

- interpret a bar code and take action based on the bar code.
- use a weight from a weigh scale when the weight is sent as ASCII characters.
- decode a message from an ASCII triggered device, such as an operator terminal.
- build a string for an ASCII triggered device using variables from your application.

Extract a Part of a Bar Code

For example, a bar code may contain information about a bag on a conveyor at an airport. To check the flight number and destination of the bag, you extract characters 10 - 18.



EXAMPLE

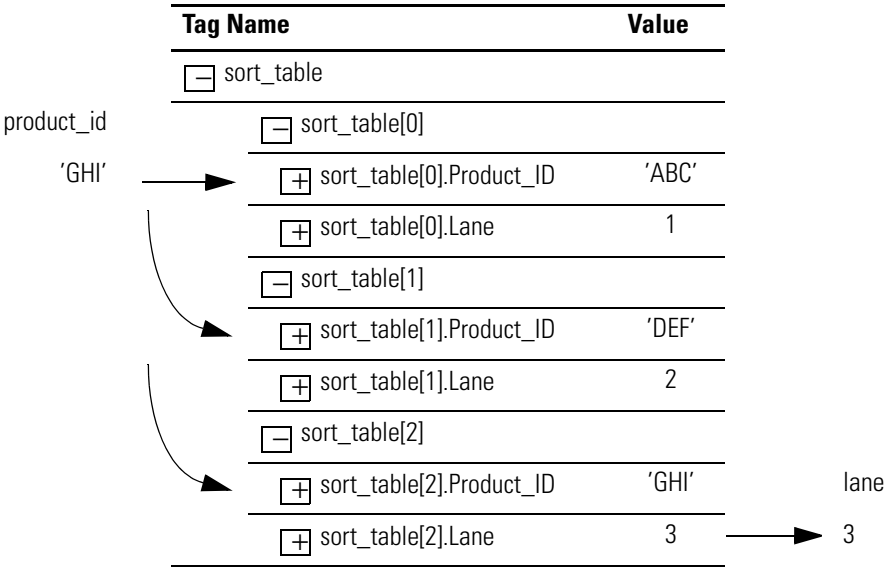
In the baggage handling conveyor of an airport, each bag gets a bar code. Characters 10 - 18 of the bar code are the flight number and destination airport of the bag. After the bar code is read (bag_read.EM is on) the MID instruction copies the flight number and destination airport to the bag_flt_and_dest tag.



42808

Look Up a Bar Code

For example, in a sorting operation, an array of a user-defined data type creates a table that shows the lane number for each type of product. To determine which lane to route a product, the controller searches the table for the product ID (characters of the bar code that identify the product).

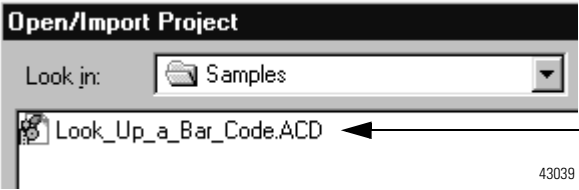


To look up a bar code:

- Create the PRODUCT_INFO Data Type.
- Search for the Characters.
- Identify the Lane Number.
- Reject Bad Characters.
- Enter the Product IDs and Lane Numbers.

TIP

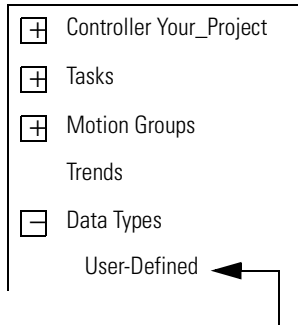
To copy the above components from a sample project, open the ...\\RSLogix 5000\\Projects\\Samples folder.



Open this project.

Create the PRODUCT_INFO Data Type

To create a new data type:

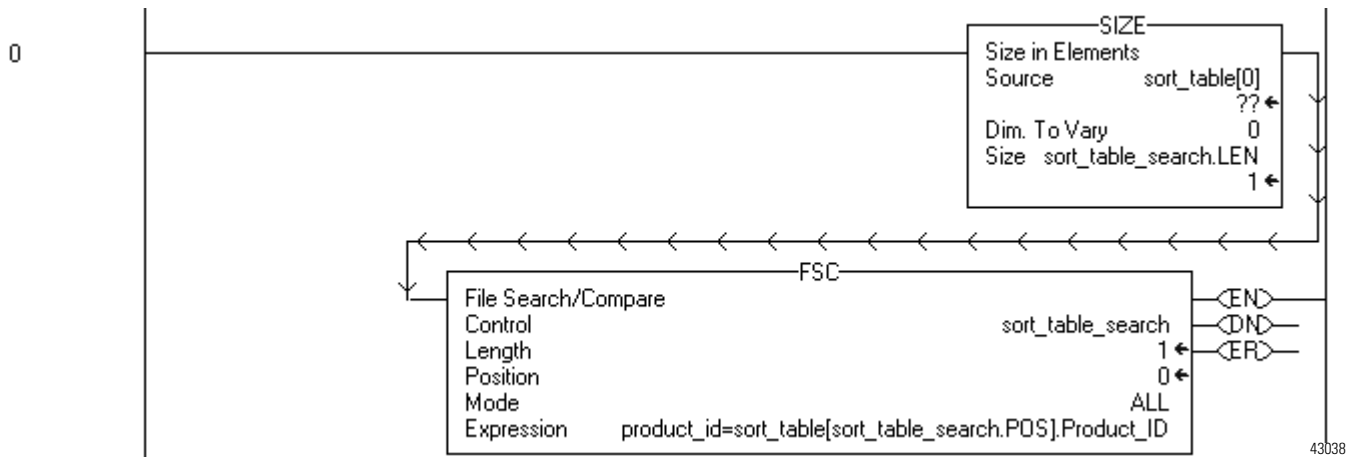


Right-click and choose *New Data Type*.

Create this user-defined data type.

Data Type: PRODUCT_INFO				
Name	PRODUCT_INFO			
	Description	Identifies the destination for an item based on an ASCII string of characters that identify the item		
Members				
	Name	Data Type	Style	Description
	Product_ID	STRING		ASCII characters that identify the item
	Lane	DINT	Decimal	Destination for the item, based on its ID

Search for the Characters



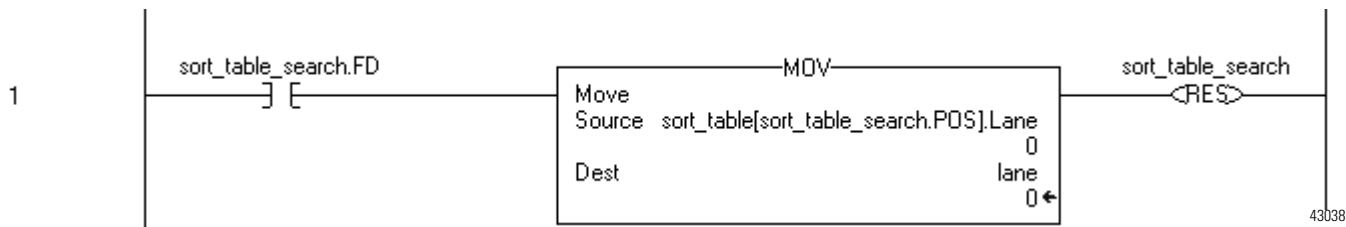
The SIZE instruction:

- counts the number of elements in the sort_table array (type PRODUCT_INFO). This array contains the product ID for each item and the corresponding lane number for the item.
- counts the number of elements in Dimension 0 of the array. In this case, that is the only dimension.
- sets the Length of the subsequent FSC instruction equal to the size of the sort_table array.

The FSC instruction searches each Product_ID member in the sort_table array until the instruction finds a match to the product_id tag.

- The sort_table_search tag controls the FSC instruction.
- Although the previous instruction sets the Length of this instruction, you enter an initial value to verify the project.
- The product_id tag contains the bar code characters that you want to find.

Identify the Lane Number



When the FSC instruction finds the product ID within the sort_table array, the instruction sets the FD bit. The POS member indicates the element number within the sort_table array of the match. The corresponding LANE member indicates the lane number of the match.

Based on the POS value, the MOV instruction moves the corresponding lane number into the lane tag. The controller uses the value of this tag to route the item.

After the MOV instruction sets the value of the lane tag, the RES instruction resets the FSC instruction so it can search for the next product ID.

Reject Bad Characters



If the FSC instruction does not find the product ID within the sort_table array, the instruction sets the DN bit. The MOV instruction moves 999 into the lane tag to notify the controller to reject or reroute the item.

After the MOV instruction sets the value of the lane tag, the RES instruction resets the FSC instruction so it can search for the next product ID.

Enter the Product IDs and Lane Numbers

In the `sort_table` array, enter the ASCII characters to identify each item and the corresponding lane number for the item.

Tag Name	Value
 <code>sort_table</code>	{...}
 <code>sort_table[0]</code>	{...}
 <code>sort_table[0].Product_ID</code>	ASCII characters that identify the first item
 <code>sort_table[0].Lane</code>	Lane number for the item
 <code>sort_table[1]</code>	{...}
 <code>sort_table[1].Product_ID</code>	ASCII characters that identify the next item
 <code>sort_table[1].Lane</code>	Lane number for the item

Check the Bar Code Characters

Use a compare instruction (`EQU`, `GEQ`, `GRT`, `LEQ`, `LES`, `NEQ`) to check for specific characters.

- The hexadecimal values of the characters determine if one string is less than or greater than another string.
- When the two strings are sorted, as in a telephone directory, the order of the strings determines which one is greater.

ASCII Characters	Hex Codes
1ab	\$31\$61\$62
1b	\$31\$62
A	\$41
AB	\$41\$42
B	\$42
a	\$61
ab	\$61\$62

l
e
s
s
e
r

↑

g
r
e
a
t
e
r

↓

— AB < B

— a > B

Use one of these compare instruction:

To see if the string is:	Enter this instruction:
Equal to specific characters	EQU
Not equal to specific characters	NEQ
Greater than specific characters	GRT
Equal to or greater than specific characters	GEQ
Less than specific characters	LES
Equal to or less than specific characters	LEQ

For example:

EXAMPLE

When bag_flt_and_dest is equal to gate[1], xfer[1] turns on. This routes the bag to the required gate.



42808

Convert a Value

You can convert the ASCII representation of a value to an DINT or REAL value that you can use in your application.

- The STOD and STOR instructions skip any initial control or non-numeric characters (except the minus sign in front of a number).
- If the string contains multiple groups of numbers that are separated by delimiters (e.g., /), the STOD and STOR instructions convert only the first group of numbers.

For example, to convert ASCII characters to a floating-point value:

EXAMPLE

After reading the weight from the scale (weight_read.EM is on), the STOR instruction converts the numeric characters in weight_ascii to a REAL value and stores the result in weight.



42810

For example, to convert ASCII characters to an integer value:

EXAMPLE

When MV_read.EM is on, the STOD instruction converts the first set of numeric characters in MV_msg to an integer value. The instruction skips the initial control character (\$06) and stops at the delimiter (\).



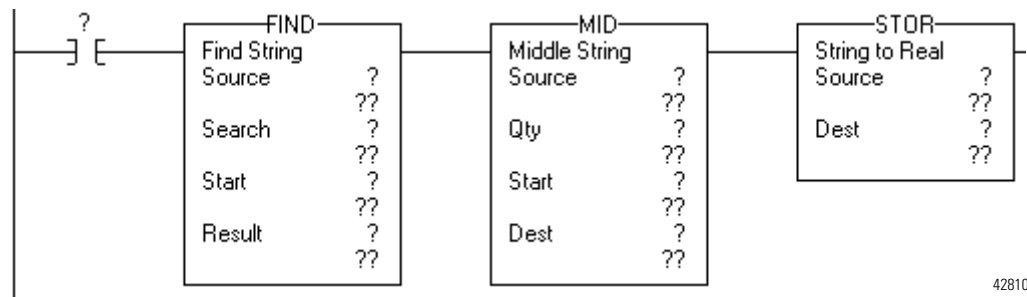
42620

Decode an ASCII Message

You can extract and convert a value from an ASCII message that contains multiple values. For example, a message may look like this:

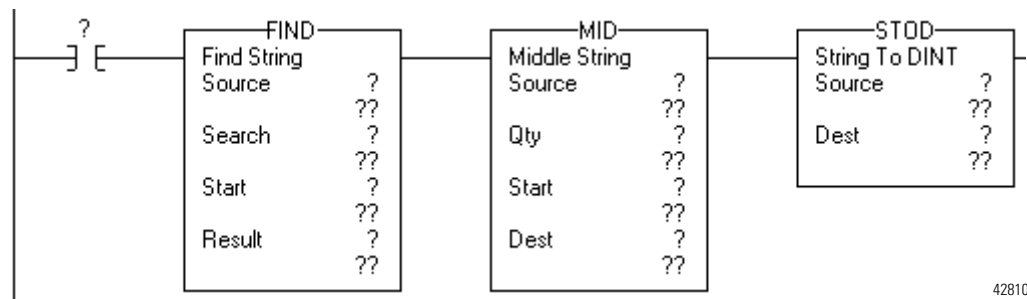
First value	Second value	Third value	
[Ctrl-F]	message #	\ F-key	\ F-key action [CR]
Control character	Delimiter	Delimiter	Termination character

Rung A: Find and Convert a Floating-Point Value



42810

Rung B: Find and Convert an Integer Value



42810

The FIND instruction locates characters within a string.

- The Source contains the string tag to search.
- The Result contains the location where the FIND instruction locates the search value you specify.

The MID instruction identifies a group of characters within a string and places them in their own string tag.

- The source is the same string tag as for the FIND instruction.
- The quantity values tells the MID instruction how many characters to pull from the source.
- The start value is the same as the Result value from the FIND instruction. This tells the MID instruction where to start pulling characters from the Source.
- The Destination contains the characters you located.

Build a String

This example builds a string that contains two variables. For example, an operator terminal may require a string that looks like this:

[Ctrl-F]	message #	\	address	[CR]
Control character		Delimiter		Termination character

- For more variables, use additional INSERT or CONCAT instructions.
- If you need to send a floating-point value, use a RTOS instruction in place of the DTOS instruction.
- The final string does not include the termination character. When you send the string, use an AWA instruction to automatically append the termination character.

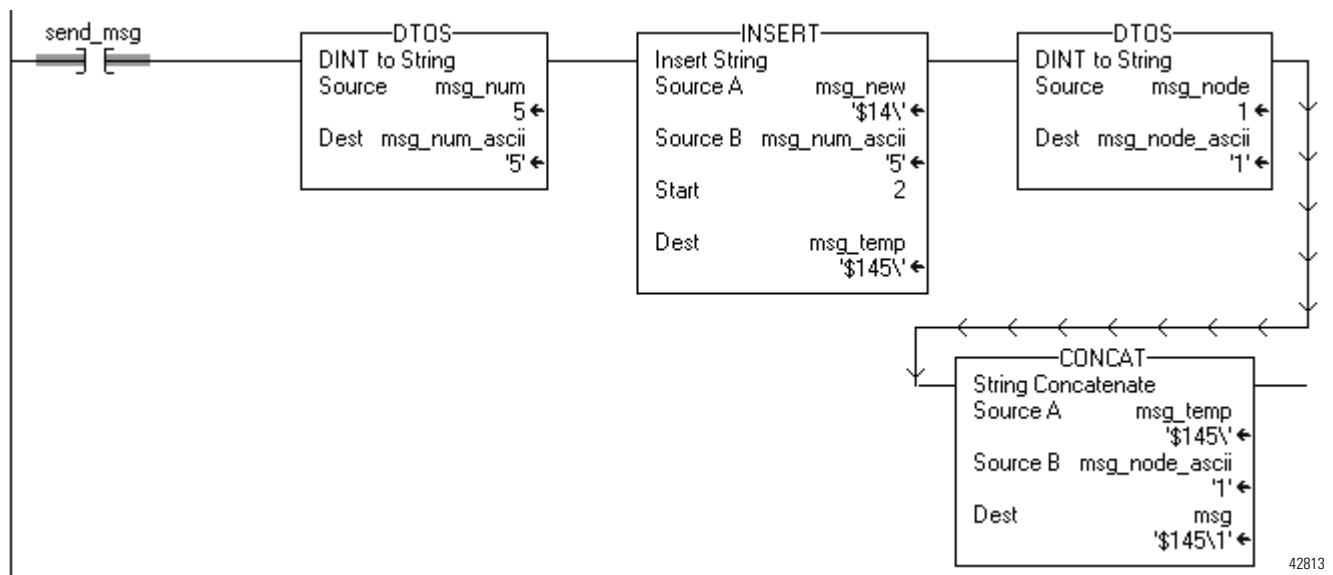
EXAMPLE

To trigger a message in a MessageView terminal, the controller sends the terminal a message in this format: [Ctrl-T] message # \ address [CR]

When send_msg is on, the rung does this:

- The first DTOS instruction converts the message number to ASCII characters.
- The INSERT instruction inserts the message number (in ASCII) after the control character [Ctrl-T]. (The hex code for Ctrl-T is \$14.)
- The second DTOS instruction converts the node number of the terminal to ASCII characters.
- The CONCAT instruction puts the node number (in ASCII) after the backslash [\] and stores the final string in msg.

To send the message, an AWA instruction sends the msg tag and appends the carriage return [CR].



42813

Notes:

ASCII Character Codes

Character	Dec	Hex
[ctrl-@] NUL	0	\$00
[ctrl-A] SOH	1	\$01
[ctrl-B] STX	2	\$02
[ctrl-C] ETX	3	\$03
[ctrl-D] EOT	4	\$04
[ctrl-E] ENQ	5	\$05
[ctrl-F] ACK	6	\$06
[ctrl-G] BEL	7	\$07
[ctrl-H] BS	8	\$08
[ctrl-I] HT	9	\$09
[ctrl-J] LF	10	\$I (\$0A)
[ctrl-K] VT	11	\$0B
[ctrl-L] FF	12	\$0C
[ctrl-M] CR	13	\$r (\$0D)
[ctrl-N] SO	14	\$0E
[ctrl-O] SI	15	\$0F
[ctrl-P] DLE	16	\$10
[ctrl-Q] DC1	17	\$11
[ctrl-R] DC2	18	\$12
[ctrl-S] DC3	19	\$13
[ctrl-T] DC4	20	\$14
[ctrl-U] NAK	21	\$15
[ctrl-V] SYN	22	\$16
[ctrl-W] ETB	23	\$17
[ctrl-X] CAN	24	\$18
[ctrl-Y] EM	25	\$19
[ctrl-Z] SUB	26	\$1A
ctrl-[ESC	27	\$1B
[ctrl-\] FS	28	\$1C
ctrl-] GS	29	\$1D
[ctrl-^] RS	30	\$1E
[ctrl-_] US	31	\$1F

Character	Dec	Hex
SPACE	32	\$20
!	33	\$21
“	34	\$22
#	35	\$23
\$	36	\$24
%	37	\$25
&	38	\$26
‘	39	\$27
(	40	\$28
)	41	\$29
*	42	\$2A
+	43	\$2B
,	44	\$2C
-	45	\$2D
.	46	\$2E
/	47	\$2F
0	48	\$30
1	49	\$31
2	50	\$32
3	51	\$33
4	52	\$34
5	53	\$35
6	54	\$36
7	55	\$37
8	56	\$38
9	57	\$39
:	58	\$3A
;	59	\$3B
<	60	\$3C
=	61	\$3D
>	62	\$3E
?	63	\$3F

Character	Dec	Hex
@	64	\$40
A	65	\$41
B	66	\$42
C	67	\$43
D	68	\$44
E	69	\$45
F	70	\$46
G	71	\$47
H	72	\$48
I	73	\$49
J	74	\$4A
K	75	\$4B
L	76	\$4C
M	77	\$4D
N	78	\$4E
O	79	\$4F
P	80	\$50
Q	81	\$51
R	82	\$52
S	83	\$53
T	84	\$54
U	85	\$55
V	86	\$56
W	87	\$57
X	88	\$58
Y	89	\$59
Z	90	\$5A
[	91	\$5B
\	92	\$5C
]	93	\$5D
^	94	\$5E
_	95	\$5F

Character	Dec	Hex
‘	96	\$60
a	97	\$61
b	98	\$62
c	99	\$63
d	100	\$64
e	101	\$65
f	102	\$66
g	103	\$67
h	104	\$68
i	105	\$69
j	106	\$6A
k	107	\$6B
l	108	\$6C
m	109	\$6D
n	110	\$6E
o	111	\$6F
p	112	\$70
q	113	\$71
r	114	\$72
s	115	\$73
t	116	\$74
u	117	\$75
v	118	\$76
w	119	\$77
x	120	\$78
y	121	\$79
z	122	\$7A
{	123	\$7B
	124	\$7C
}	125	\$7D
~	126	\$7E
DEL	127	\$7F

Rockwell Automation Support

Rockwell Automation provides technical information on the Web to assist you in using its products. At <http://support.rockwellautomation.com>, you can find technical manuals, a knowledge base of FAQs, technical and application notes, sample code and links to software service packs, and a MySupport feature that you can customize to make the best use of these tools.

For an additional level of technical phone support for installation, configuration, and troubleshooting, we offer TechConnect support programs. For more information, contact your local distributor or Rockwell Automation representative, or visit <http://support.rockwellautomation.com>.

Installation Assistance

If you experience a problem within the first 24 hours of installation, please review the information that's contained in this manual. You can also contact a special Customer Support number for initial help in getting your product up and running.

United States	1.440.646.3434 Monday – Friday, 8am – 5pm EST
Outside United States	Please contact your local Rockwell Automation representative for any technical support issues.

New Product Satisfaction Return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

United States	Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor in order to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for the return procedure.

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444

Europe/Middle East/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640

Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication 1756-PM013B-EN-P - July 2008

Supersedes Publication 1756-PM013A-EN-P - July 2007

Copyright © 2008 Rockwell Automation, Inc. All rights reserved. Printed in the U.S.A.