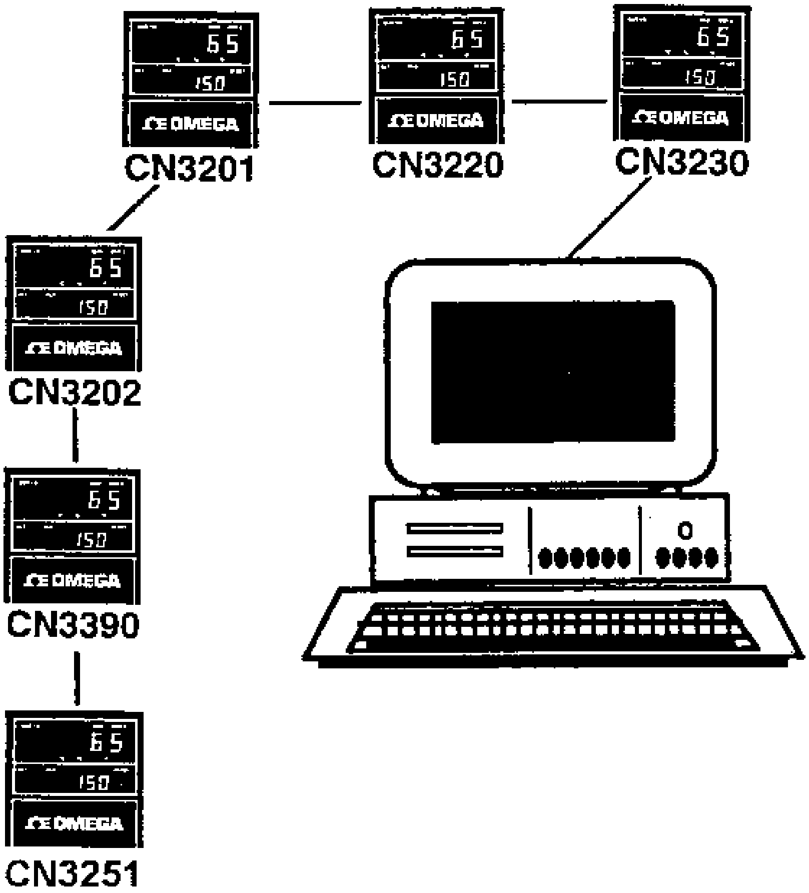
Digital Communications for CN3201, CN3202, CN3220, CN3230, CN3390, CN3251



OMEGA®
An OMEGA Technologies Company

Operator's Manual

THE TEMPERATURE HANDBOOK
THE PRESSURE STRAIN AND FORCE HANDBOOK
OMEGA COMPLETE VOLUME 28 HANDBOOK & ENCYCLOPEDIA SET
OMEGA
DATA ACQUISITION SYSTEMS
THE FLOW AND LEVEL HANDBOOK
THE GREEN BOOK
OMEGA

Table of Contents

List of Figures

1

Introduction

Understanding the 4 Digital Communications Operating Modes

Modes for Connection to Terminal or Printer

CN 3200-SOFT Standard Software

Understanding the 4 Digital Communications Operating Modes

This supplement contains the basic programming and operating instructions for the Digital Communications option for the OMEGA CN3200 Series controller models CN3201, CN3202, CN3220, CN3230, CN3251 and the CN3390. The Digital Communications Interface has four operating modes:

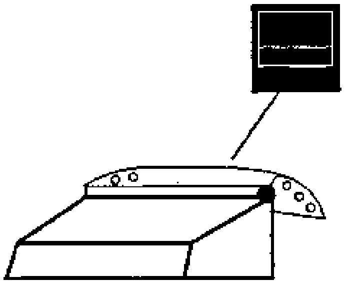
Automatic Data Logging Mode
Terminal Interface Mode
ASCII Line Mode
Computer Interface Mode

Modes for Connection to Terminal or Printer

Two of these modes allow you to communicate with one controller, and may be utilized without any further programming or software interface:

Automatic Data Logging Mode gives the controller the ability to periodically transmit controller/process status information to a computer, printer or recorder. This gives you a continuous record of the controller status (information same as shown on PAGE 0: DISPLAY PAGE). The commands of the Terminal Mode (description follows) can also be accessed while in the Automatic Data Logging Mode.

Figure 1.1
Automatic Data Logging to Printer



■ Understanding the 4 Digital Communications Operating Modes (continued)

Terminal Interface Mode allows you to remotely access, monitor and program one controller using a terminal or computer. There are 7 basic commands in this mode.

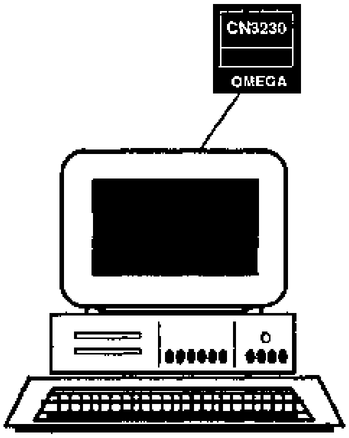


Figure 1.2
Terminal Interface Mode

Two Modes for Specialized Application Programs

In applications requiring more sophisticated digital communication with one or more controllers, up to as many as 255 controllers, there are two methods of communication:

ASCII Line Mode
Computer Interface Mode (CPIF)

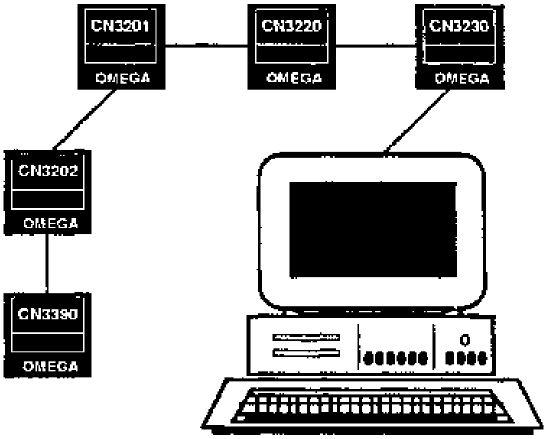


Figure 1.3
Multidrop Controller Network

■ Understanding the 4 Digital Communications Operating Modes (continued)

If a customized application program is called for, these Digital Communications modes will allow you to write your own programs to generate command messages and interpret response messages.

ASCII Line Mode is a format of commands and responses which OMEGA controllers understand. In this manual, format is described so that you can generate a program on your computer to communicate with one or several OMEGA controllers. In this mode, command and response messages consist of ASCII characters which represent hexidecimal (HEX) numbers.

The Line Mode uses check sums for error checking. This method provides reliability for correct data transmission. This mode is simpler to implement than the Computer Interface Mode (described below). The ASCII Line Mode is recommended for developing customized applications. Because ASCII characters are used instead of binary numbers, the ASCII Line Mode is much simpler to implement and requires no other protocol-type documentation.

Computer Interface Mode (CPIF), uses a byte oriented protocol to coordinate communication between the computer and multiple controllers in a multidrop network configuration. The CPIF mode corresponds with Allen-Bradley Data Highway protocol and, for customized programs, requires that you are familiar with this protocol for implementation. This mode of communication is faster and has better error recovery provisions than ASCII mode, but is more difficult to implement.

■ CN 3200-SOFT Standard Software

If a prepackaged software package is preferred, OMEGA offers CN 3200-SOFT remote operator interface software. CN 3200-SOFT operates on an IBM-PC or compatible computer and communicates with the controllers via a serial interface port.

CN 3200-SOFT may be used to perform the following functions:

1. Change all Set-Up Parameters.
2. Remotely Operate Ramp/Soak Programs.
3. SAVE and LOAD files containing Ramp/Soak profiles and controller configuration.
4. Generate Graphic Representations of the Process on the computer screen.
5. Datalog parameters and send them to a file.

More detailed information on CN 3200-SOFT is available from OMEGA.

2

Wiring and Hardware Set-Up

■ Jumper Positions

To use the Digital Communication interface option on your controller you must make appropriate wiring connections. In some cases, internal jumpers in the controller hardware must be positioned for the communication interface you are using (RS232, RS422 or RS485). If internal jumper positioning is necessary, you will find it easier to position the jumpers before wiring the controller.

Jumper locations and positions are given for models CN3201/CN3202, CN3220 and CN3230 in figures 2.1 through 2.3. Note that in all cases, the internal jumpers are positioned for RS232 when shipped from the factory. If you are using RS232, then it is not necessary to reposition the jumpers.

Figures 2.4 through 2.7 give controller wiring terminal connections for connecting the controller and computer.

This document may be used in conjunction with the OMEGA User's Manual that came with your controller, which contains more explicit and detailed installation and wiring information for the digital communications interface. This information will be found on the following pages in the controller User's Manual:

Product	Manual	Pages
CN3201	CN3201 User's Manual	55
CN3202	CN3202 User's Manual	55
CN3220	CN3220 User's Manual	41
CN3230	CN3230 User's Manual	55
CN3390	CN3390 User's Manual	60
CN3251	CN3251 User's Manual	9-1

Figure 2.1
Models CN3201 and
CN3202 Jumper Locations

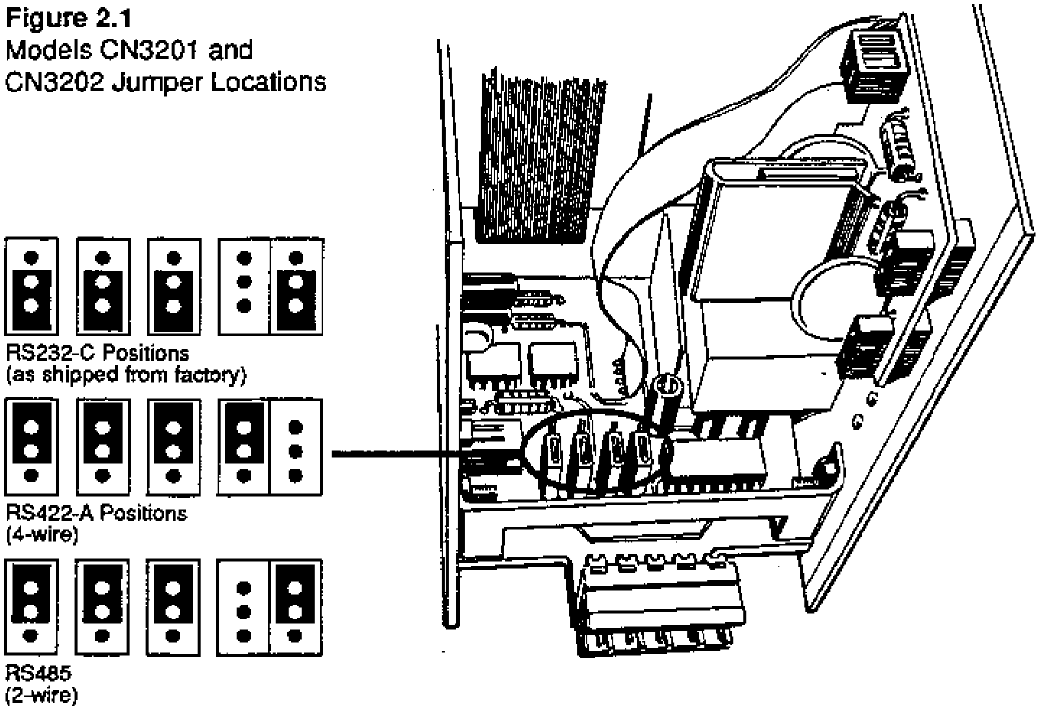
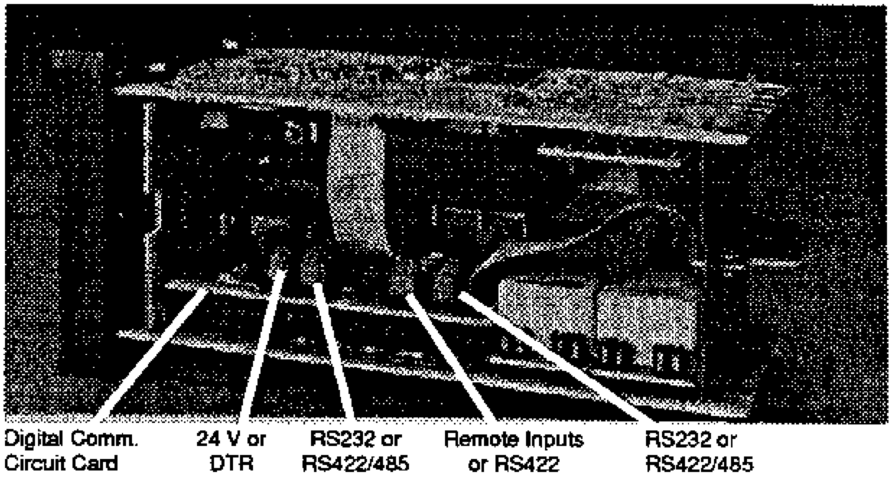


Figure 2.2
Model CN3220
Jumper Locations
and Positons



Jumper Positions

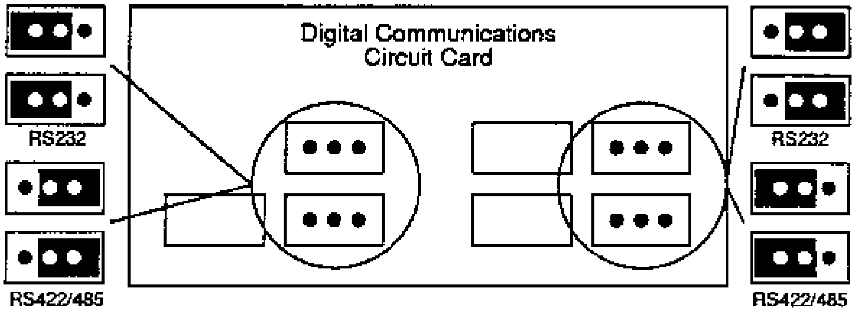
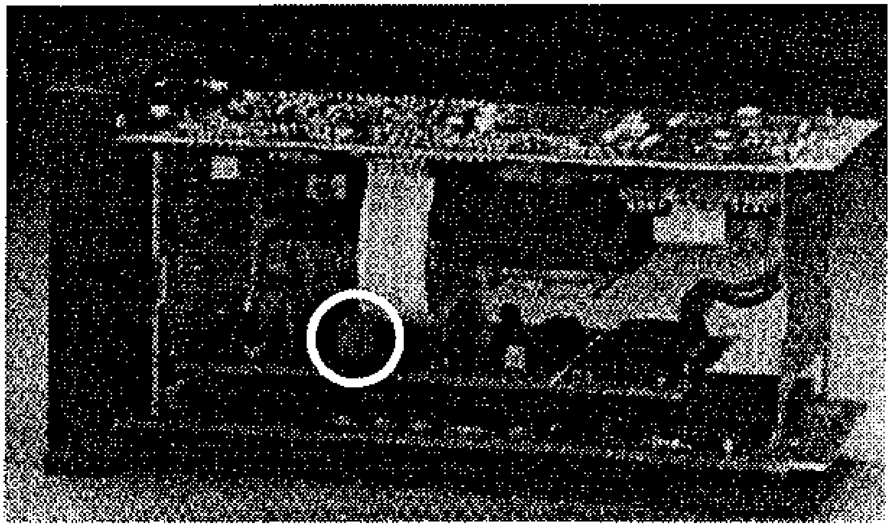
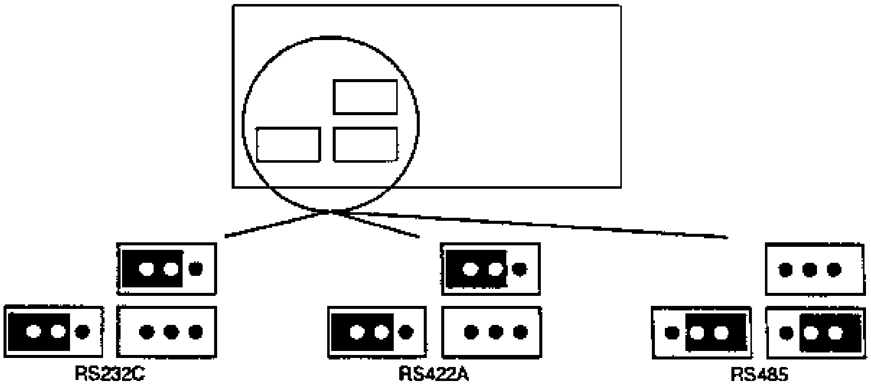


Figure 2.3
Model CN3230
Jumper Location
and Position



Jumper Positions



Computer to Controller Wiring

OMEGA controllers have three serial communications standards to choose from: RS-232C, RS-422A and RS-485. RS-232C is the simplest to use since it is limited to a single drop configuration, i.e. one controller and one computer.

RS-422A and RS-485 are implemented when multiple controllers are connected to one computer. RS-422A is a full-duplex protocol (the controller can transmit and receive simultaneously). This speeds up the communication, but RS-422A requires 5 conductors (2 transmit, 2 receive and 1 ground). RS-485 only needs three conductors, but uses a half-duplex protocol (the controller can only transmit or receive at one time) which can slow down communication.

Note: When specifying wire for a RS422 or RS485 network, use low capacitance wire specifically designed for use with RS422 or RS485, such as Belden 8164.

The following table gives the controller terminal numbers for connecting the computer to the controller(s). For example, if using RS-232C and a Model CN3230 controller, terminal #42 is XMT (Transmit), #43 is RCV (Receive) and #44 is SG (Signal Ground).

Figure 2.4
Controller Terminal Designations

OMEGA Controller	RS-232C			RS-422A					RS-485		
Model	XMIT	RCV	SG	XMT+	XMT-	RCV+	RCV-	G	RT+	RT-	G
CN3201/CN3202	3	1	5	3	4	1	2	5	3	4	5
CN3220*	9	10	L	9	10	7	8	L	9	10	L
CN3230	42	43	44	39	40	37	38	41	39	40	41
CN3390	2	3	7	23	24	12	13	7	23	24	7

* Terminals located on digital interface connector.

RS-232C

The following table shows the RS-232C controller to computer connections.

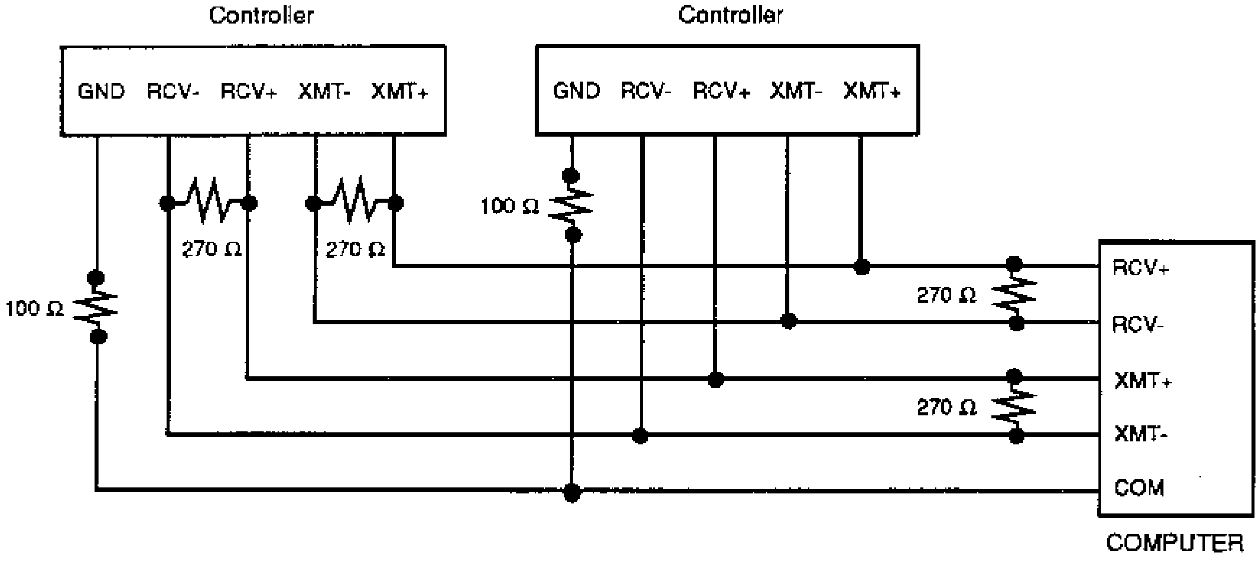
Figure 2.5
Computer to Controller Wiring for RS-232C

Computer		OMEGA Controller (models CN3201, CN3202, CN3220, CN3230, CN3390)
RCV	←	XMT (Transmit)
XMT	→	RCV (Receive)
DTR	→	DSR (3390 only)
SG	→	SIG GROUND

■ RS-422A

For 4-wire RS-422A connections, follow Figure 2.6. Note that the 100 ohm resistors are not needed if the controller has isolated digital communications (available on all OMEGA CN3390 and some model CN3220 controllers).

Figure 2.6
Computer to Controller Wiring for RS-422A

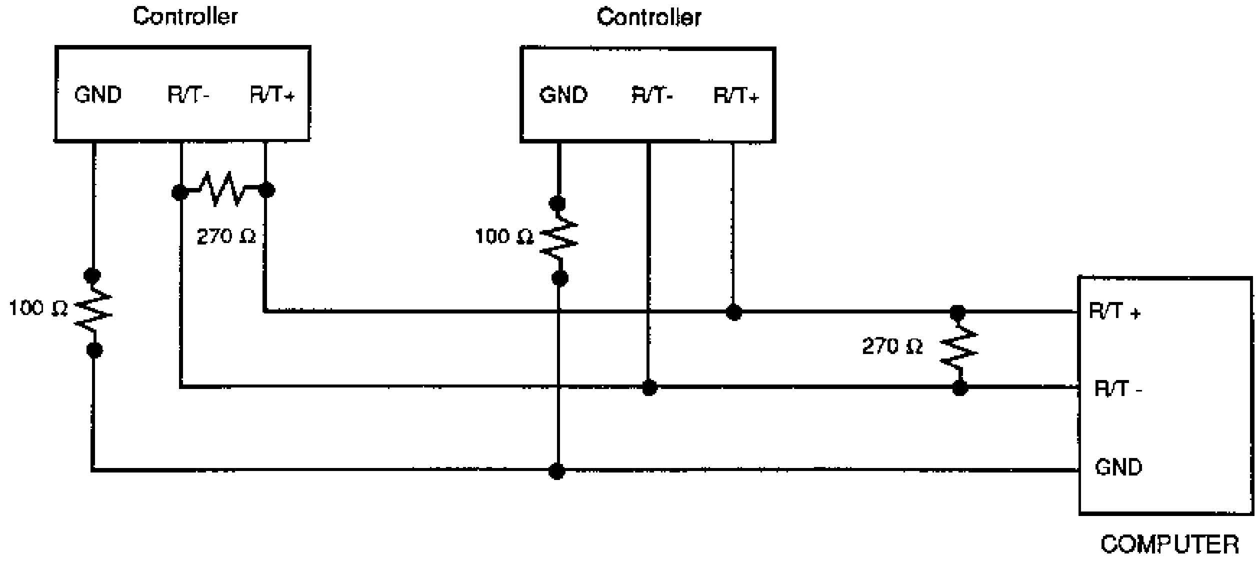


NOTE : 270 Ω resistors recommended across receive line and transmit line on computer and last controller.

■ RS-485

For 2-wire RS-485 connections, follow Figure 2.7. Note that the 100 ohm resistors are not needed if the controller has isolated digital communications (available on all CN3390 and on some CN3220 controllers).

Figure 2.7
Computer to Controller Wiring for RS-485



NOTE : 270 Ω resistors recommended across receive line oncomputer and last controller.

Controller Menu Settings

The following MENUs contain the settings for using the Digital Communications option. The Digital Communications PAGE, where the MENU settings are found, is shown below for each controller model.

Controller Model	Digital Communications PAGE
CN3201/CN3202	3
CN3220	9
CN3230	21
CN3390	15

Menu 1 selects the communications mode (or Protocol).

Digital Communications PAGE

Alpha	MENU	Selection	Available Setting	Factory Setting	Sec.
MODE	1	Mode Selection	Off Terminal Auto Log Comp Interface ASCII Line	Off	D

Menus 2-5 are active only when the Auto Log mode is selected in Menu 1. Menus 3 and 4 select which Menus on PAGE 0 will be logged. Menu 5 refers to the character length of each Menu description.

Alpha	MENU	Selection	Available Setting	Factory Setting	Sec.
LOG INT	2	Automatic Data Log Interval	1 to 9999	1 min.	
FROM M#	3	First PAGE/MENU # to Display	1 to 40	1	
To M#	4	Last PAGE 0/MENU # to Display	1 to 40	40	
CHRLNGTH	5*	Character Length	Under 17 Over 16	Over 16	

Menus 6 and7 are active when the Terminal Mode is selected in Menu 1. The Home and Clear Screen characters refer to the dumb terminal settings.

Alpha	MENU	Selection	Available Setting	Factory Setting	Sec.
HOMECHAR	5 (6 on CN3390)	Home Character	0 to 255	30	
CLR CHAR	6 (7 on CN3390)	Clear Screen Character	0 to 255	26	

* Model 3390 only.

Controller Menu Settings

(continued)

Menus 7, 8 and 9 must be selected anytime Digital Communications is being used.

Com Type	7 (8 on CN3390)	Communication Type	RS232C RS422A RS485	RS232C
BaudRate	8 (9 on CN3390)	Baud Rate	300 600 1200 2400 4800 9600 19.2K	19.2K
Parity	9 (10 on CN3390)	Parity	None Odd Even	None

Menu 10 must be set any time changes are made on the Digital Communications PAGE. The changes are not implemented until Menu 10 is set to reconfigure.

Reconfig	10 (11 on CN3390)	Reconfigure Serial Port	Ready ▲ Reconfigure	Ready ▲

Menu 11 must be set if the Computer Interface or ASCII Line Mode are selected in Menu 1.

Address	11 (12 on CN3390)	Multidrop Address	0-255	1

3

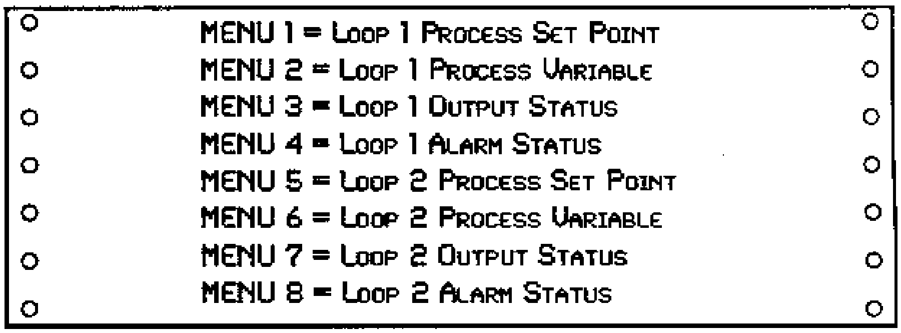
Automatic Data Logging Mode

Controller Set-Up

Automatic Data Logging Operation

The Automatic Data Logging option is designed to provide a record or printout of selected MENU variables. The data logging function works with a simple ASCII printer or terminal. The terminal or printer must have an RS232, RS422 or RS485 serial communication port. A sample automatic data logging printout from an OMEGA CN3220 controller is shown below.

Figure 3.1
Automatic Data Logging
Sample Printout



```
MENU 1 = LOOP 1 PROCESS SET POINT
MENU 2 = LOOP 1 PROCESS VARIABLE
MENU 3 = LOOP 1 OUTPUT STATUS
MENU 4 = LOOP 1 ALARM STATUS
MENU 5 = LOOP 2 PROCESS SET POINT
MENU 6 = LOOP 2 PROCESS VARIABLE
MENU 7 = LOOP 2 OUTPUT STATUS
MENU 8 = LOOP 2 ALARM STATUS
```

This print out was generated using an RS232 input printer. The MENU variables were selected from PAGE 0, with MENU 1 as the first MENU to log, and MENU 5 as the end MENU to log.

(continue on next page)

■ Controller Set-Up

The Automatic Data Logging function is enabled and defined on the Digital Communications PAGE of each controller. These programming PAGEs for the CN3201, CN3202, CN3220, CN3230 and CN3390 are presented on the following pages of this manual.

To enable the Automatic Data Logging mode, you must:

1. Select the Automatic Data Logging operation mode by selecting the value "2" or "Log Int" at MENU 1.
2. Select the time interval at MENU 2.
3. Choose the first MENU number to display at MENU 3, and
4. Choose the last MENU number to display at MENU 4.
5. Confirm that MENUs 7, 8 and 9 are set correctly for your communications interface.
6. Reconfigure set-up by entering a "1" at MENU 10.

The "first and last MENU number to display" is selected from PAGE 0 (Display PAGE) of the controller that is communicating with the terminal.

☛ **Be sure to enter a "1" at MENU 10 after making selections at MENUs 1-4 to reconfigure the set-up.**

■ Automatic Data Logging Operation

The logging interval will automatically begin 1 minute after power-up of the controller. No operation interface is required for automatic data logging.

4

Terminal Interface Mode

Terminal Interface Controller Set-Up

Terminal Interface Commands

■ Terminal Interface Controller Set-Up

The Terminal Interface Mode allows the controller to be operated from a terminal and is very simple to enable. This mode only allows a single controller to be connected to the terminal or computer. The terminal must be an ASCII terminal with an RS232 or RS422/485 communications port. Go to the appropriate Digital Communications PAGE for your controller and make the selections that define your communications interface:

CN3201 and CN3202 – PAGE 3
CN3220 – PAGE 8
CN3230 – PAGE 21

The Digital Communications PAGE/MENU table is presented on page 11-12 of this manual.

To set-up the Terminal Interface Mode:

1. Select "1" or "TRNL" = Terminal Interface at MENU 1.
2. If your terminal has a Home Character, set in decimal at MENU 5.
3. If your terminal has a Clear Screen Character, set in decimal at MENU 6.

   ☛ <u>Note:</u> MENUs 5 and 6 are used with the DISPLAY command when updating data. If these characters are not used, the DISPLAY information will scroll.
4. Select the Interface Type (RS232, RS422, RS485) at MENU 7 (or Menu 8 for CN3390). The internal jumpers in the controller must also be set for the correct interface type (see pages 5-7 for jumper positions).
5. Select the Baud Rate at MENU 8 (or Menu 9 for CN3390). This setting must agree with Baud Rate setting of the terminal/computer.
6. Enter the Parity Setting at MENU 9 (or Menu 10 for CN3390).
7. ☛ **Be sure to enter a "1" at MENU 10 (or Menu 11 for CN3390) after making all selections to reconfigure set-up.**

Terminal Interface Commands

The digital communications terminal interface consists of 7 basic operational commands and a HELP command:

```
* ACCESS
* DISPLAY
* CHANGE
* PAGE
* SHOW
* LOCK
* UNLOCK
```

After you have powered up the controller, made the digital communications MENU selections on the Digital Communications PAGE and turned on the terminal, you should get a prompt (*) on the terminal screen. If you do not, check your terminal/computer manual to make sure that the baud rate agrees with the baud rate you entered on the Digital Communications PAGE, and check the wiring between the controller and terminal.

In the operational command prompts that you see on the screen, there are 3 abbreviations used:

```
<Enter> = Enter or Carriage Return
<▼> = Cursor Down or Line Feed
<▲> = Cursor Up or Space Bar
```

Note 1

A comma is inserted in many of the commands as a delimiter. A space is interchangeable with the comma and may also be used as a delimiter (Example: DISPLAY,5,7 or DISPLAY 5 7).

Note 2

All commands should be entered in all capital letters (upper case). For example, enter the command as "HELP", not "help".

HELP or H

This command gives you a listing of all of the digital communications interface commands. Type "H" or "HELP" at the prompt:

```
*H ⏎
C or CHANGE PAGE#, MENU#
P or PAGE Page#
S or SHOW Start page#, End page#
D or DISPLAY Start menu#, End menu#
A or ACCESS Security code
L or LOCK
U or UNLOCK
```

ACCESS or A

This command allows you to adjust the security level access of your terminal. Adjusting the security level via the terminal does not affect the security level of the front panel controller pushbuttons. If a sufficient level of access is not entered, you may be able to view only a MENU setting and not change it.

To adjust the access level, simply enter the ACCESS command followed by the security code for the level of access you want:

```
*A, 458 ⏎
```

The security codes are found in the User Manual that came with your controller.

■ Terminal Interface Commands (continued)

DISPLAY or D

The DISPLAY command is designed to display and continuously update a range of MENU numbers from PAGE 0 - the Display PAGE. You select and enter the beginning and ending MENU numbers to be displayed. In the example below, the beginning MENU number is 5 and the ending MENU number is 7:

```
*D, 5, 7 ↵
DISPLAY PAGE
MENU # 05 PROGRAM # . . . . . . . . . . . 1
MENU # 06 INTERVAL # . . . . . . . . . . . 7
MENU # 07 TIME LEFT IN INTERVAL . . 0.21
```

The controller will continue to scan and update these MENU numbers until you terminate the command by pressing ^X (control X).

The range of available PAGE 0: DISPLAY MENUs is shown in the table below:

Controller	CN3201	CN3202	CN3220	CN3230	CN3390
MENU#	1-9	1-10	1-19	1-11	1-40

CHANGE or C

The CHANGE command allows you to enter new MENU values on each page via the terminal (assuming that you have the proper Security Level accessed). To change a PAGE/MENU value:

```
*C, 1, 1 ↵
PAGE # 01, MENU # 01
MENU CONTENTS . . . . . . . . . . . 100
ENTER NEW VALUE (XXXX) 95
MENU CONTENTS . . . . . . . . . . . 95
ENTER NEW VALUE (XXXX) ↵
```

Pressing <Enter> will end the CHANGE command, whereas <▲> (cursor up) or <SPACE> allows you to go to the previous MENU number and <▼> (cursor down) will increment to the next MENU number. This is helpful if you are changing several MENU settings on the same PAGE.

PAGE or P

The PAGE command simply gives you an on-screen listing of the PAGEs (from the controller's PAGE/MENU programming set-up) and the title of the PAGE. The PAGE command serves as a reference tool, allowing you to review the PAGE titles to locate a specific selection or adjustment. The PAGE command does not allow you to change or modify any MENU setting on the PAGE.

After selecting a PAGE, you can go to the next PAGE number using the <Line Feed> or ▼, and to the previous PAGE number with a <SPACE> command.

Following is an example of the PAGE command. PAGE 0 was selected, then the ▼ key used to go the next PAGE number.

```
*P, 0 ↵
PAGE # = 00 DISPLAY PAGE ▼
PAGE # = 01 RAMP/SOAK PROGRAM 1.1 ▼
PAGE # = 02 RAMP/SOAK PROGRAM 2.1 ▼
PAGE # = 03 RAMP/SOAK PROGRAM 3.1 ▼
PAGE # = 04 RAMP/SOAK PROGRAM 4.1
PAGE # = 05 RAMP/SOAK PROGRAM 1.2
PAGE # = 06 RAMP/SOAK PROGRAM 2.2
PAGE # = 07 RAMP/SOAK PROGRAM 3.2
PAGE # = 08 RAMP/SOAK PROGRAM 4.2
PAGE # = 09 RAMP/SOAK PROGRAM 1.3
PAGE # = 10 RAMP/SOAK PROGRAM 2.3
PAGE # = 11 RAMP/SOAK PROGRAM 3.3
PAGE # = 12 RAMP/SOAK PROGRAM 4.3
PAGE # = 13 RAMP/SOAK PROGRAM 1.4
PAGE # = 14 RAMP/SOAK PROGRAM 2.4
PAGE # = 15 RAMP/SOAK PROGRAM 3.4
PAGE # = 16 RAMP/SOAK PROGRAM 4.4
PAGE # = 17 NESTED PROGRAM LOOPS
PAGE # = 18 GENERAL OPERATION
PAGE # = 19 CHANNEL #1 CONTROL AND ALARM
PAGE # = 20 CHANNEL #2 CONTROL AND ALARM
PAGE # = 21 DIGITAL COMMUNICATIONS
PAGE # = 22 CALIBRATION ▲
PAGE # = 21 DIGITAL COMMUNICATIONS ▲
PAGE # = 20 CHANNEL #2 CONTROL AND ALARM ↵
```

Following is a table of the available PAGE numbers for each controller model:

Model	CN3201	CN3202	CN3220	CN3230	CN3390
PAGEs	0-4	0-4	0-12	0-22	0-27

SHOW or S

The SHOW command will allow you to display the complete MENU contents of a PAGE or range of PAGEs, by simply specifying the beginning PAGE and the ending PAGE (using the same PAGE number as the beginning and end will show that one PAGE only).

```
*S, 18, 18 ↵
PAGE # 18 GENERAL OPERATION PAGE
MENU # 01 CONTENTS. . . . . . . .736
MENU # 02 CONTENTS. . . . . . . .1
MENU # 04 CONTENTS. . . . . . . .2
MENU # 05 CONTENTS. . . . . . . .2
MENU # 06 CONTENTS. . . . . . . .3
MENU # 07 CONTENTS. . . . . . . .1
MENU # 08 CONTENTS. . . . . . . .29.92
MENU # 09 CONTENTS. . . . . . . .6
MENU # 10 CONTENTS. . . . . . . .1
MENU # 11 CONTENTS. . . . . . . .7
```

At any time during the execution of the SHOW command, you may give the following commands:

```
^W (Control W) = Pause
^X (Control X) = Abort
```

Striking any key after pause will reinstate the SHOW command.

LOCK or L

The LOCK command allows you to lock out the controller's front panel pushbuttons, thus disabling any control adjustments from the controller. Simply type the command "LOCK" at the prompt.

```
*LOCK or ↵
*L ↵
```

The front panel pushbuttons are enabled at power-up, therefore, if adjustments are to be disabled, the LOCK command must be reentered at each power-up.

UNLOCK or U

The UNLOCK command simply unlocks the front panel pushbuttons allowing adjustments to be made from the controller front panel.

```
*UNLOCK ↵
*U ↵
```

5

ASCII Line Mode

■ Description

If you choose to write your own software program to communicate with one or more controllers, the ASCII Line Mode should be used.

The ASCII Line Mode uses command/response message pairs to allow a computer to access and remotely operate the controller. The protocol consists of command strings of ASCII characters representing hexadecimal numbers. This is the preferred mode for most customized program development applications.

These command/response messages allow the computer to be used to:

1. Set-up all controller PAGE/MENU settings.
2. Check Alarm and Event status.
3. Lock the controller front panel pushbuttons
4. Remotely operate Ramp/Soak START, RESET, HOLD
5. Check status and limits of the controller.

In the ASCII line mode configuration, the computer sends command messages to the controller and receives response messages back from the controller. Thus, the controller will only respond when the computer sends it a command. This makes it possible to assign controller "addresses" and to connect multiple controllers to a single computer.

Each time a command is transmitted, a controller should respond to it. No response will be returned if there is a problem in the serial communication link, if no controller was at that address, or if the controller and computer are not configured correctly (baud rate, etc.). If no controller responds, then the computer program needs to time-out and recover.

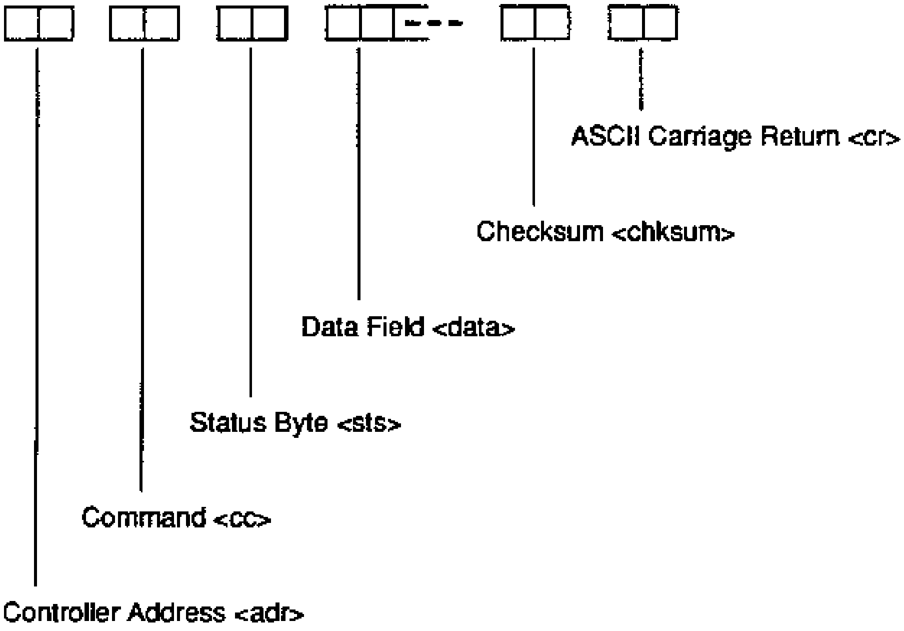
Controller Set-Up

To configure the controller(s) for ASCII Line Mode communication, you must make several selections on the Digital Communications PAGE of your controller(s). These PAGEs are shown on pages 10-11 of this manual.

1. Select "4" or LinE = ASCII Line Mode at MENU 1.
2. Select the appropriate interface, baud rate and parity at MENUs 7-9.
3. Enter the address selected for the controller at MENU 11.
4. **Be sure to enter the value "1" at MENU 10 when all selections have been made to reconfigure the controller set-up.**

Command/Response Message Format

The actual message that is sent is a string of ASCII characters representing hexadecimal numbers (0-9 and A-F) ending with ↵ . All other characters are ignored. For each byte of data in the command or response, two ASCII characters representing the hexadecimal value of the data byte are sent.



The following pages present a definition of each field in the message. In all example strings, spaces are added between bytes to make them easier to read. Spaces are not transmitted.

Command/Response Message Format

(continued)

Controller Address <adr>

Each controller is assigned a different numerical address (1-254) at MENU 11. In the command/response message, the <adr> field is a two hex digit (01-FE) equivalent to the numerical address. For example, if the controller is set up to be address 47, then that controller will only respond to commands with an address field of 2F (the HEX equivalent of 47).

Commands <cc>

The <cc> field is the command which determines the function requested by the computer to the controller. These computer commands <cc> and the controller response codes <rc> returned in the <cc> field of the response message are summarized in the following table:

Command	<cc> HEX	Data String	<rc> HEX
	From computer		From controller
Read Menu Values	01		41
Write Menu Values	08		48
Pushbutton Commands:			
Reset Program	05	0000 01 00	45
Start/Continue Program	05	0000 02 00	45
Hold Program	05	0000 04 00	45
Lock Front Panel	05	0000 08 00	45
Unlock Front Panel	05	0000 00 08	45
Alarm Acknowledge (CN3390)	05	0000 10 00	45
Change Access Security Code	09		49
Manual Output Adjust	0A		4A
Resume Auto Output	0B		4B
Return Alarm Status	0C		4C
Return Max. Viewable and Adjust. MENU # on PAGE	0D		4D
Return MENU Adjust Limits	0E		4E
Return Model Number	0F		4F
Return Event Status	13		53

Command/Response Message Format

(continued)

Status Byte <sts>

The status byte field <sts> indicates the "status" of the command message from the computer to the controller—i.e. it was either successful (no error encountered) or unsuccessful (error encountered).

In the command message from the computer, the <sts> field is always 00. The <sts> field in the response message from the controller will be 00, if no error was encountered. If an error occurred, one of the following errors will appear in the <sts> field of the response message:

Status Byte Value	Error Description
00	No error
01	Security level too low
02	Value out of range
03	Controller front panel in use
04	Invalid bit mask
05	Invalid command
06	Command string too short
07	Invalid page number
08	Invalid menu number
09	Invalid output number
0A	Manual output adjust disabled
0B	Ramp/Soak disabled

Data Field <data>

The format of the data field varies for each command and response, which is shown in the detailed Command/Response Message Examples that follow starting on page 27.

All data fields are in binary format, and the legal values and ordering are specified with each command description. Numbers of 32 bit length are transmitted least significant byte (lsb) first and most significant byte (msb) last.

Error Checking <Checksum>

The checksum is an error checking method which allows the controller or computer receiving information to check that the data sent was not garbled during transmission. Each command or response line is terminated with a checksum. This checksum is the 2's complement of the sum of the data bytes. The checksum can be calculated by adding together all of the message data bytes, then negating the sum, and finally converting the byte to two ASCII characters representing the hex value of the checksum. These two characters are transmitted followed by a CR to terminate the line.

Checksum Calculation Example

-Convert from ASCII character pairs to hex bytes.
-Add bytes together using HEX math, limiting sum to 1 byte. This is done by retaining the two least significant digits.
-Negate the sum to obtain the checksum. This is done by subtracting the 1 byte sum from 100 HEX (base 16).If the controller receives a command containing a checksum error, it will not execute the command and will return a response message containing the address, the command byte with the most significant bit set indicating a checksum error, a status byte of 0 and a checksum. The command can then be sent again with the proper checksum. The command byte in the response message will be in the range of C1 to D3 hex (i.e. if the command byte in the response message is between C1 and D3, a checksum error has occurred in the command message).

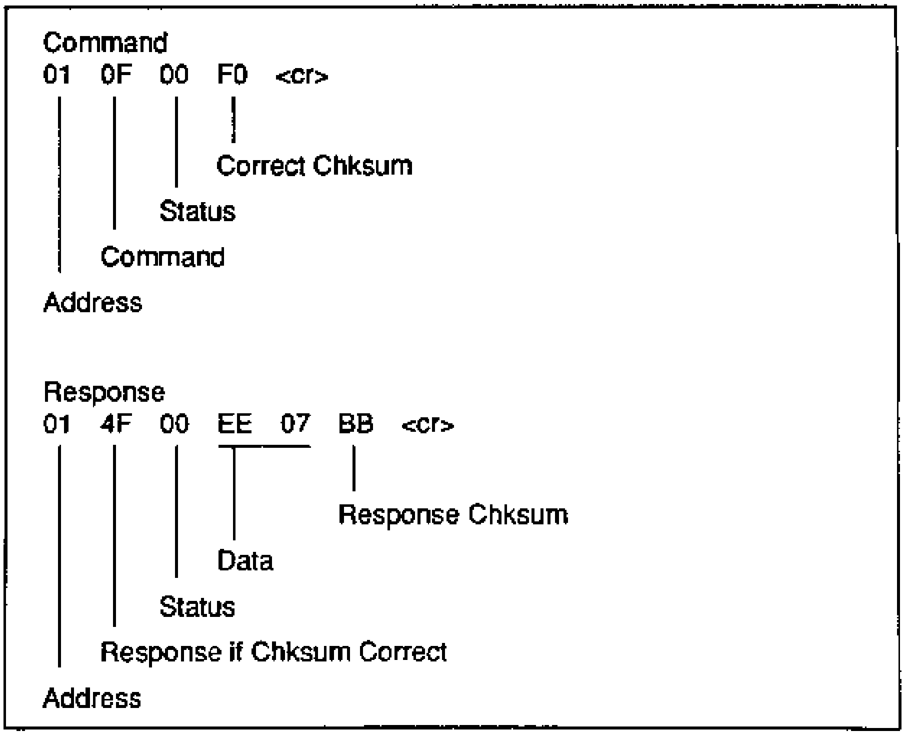
Example
Command String 01 08 00 14 01 E8 03 F7 <CR>

-Add bytes together	01+08+00+14+01+E8+03 = 109
-Limit to 1 byte	109 becomes 09
-Negate the sum	100 - 09 = F7

-Checksum is F7

The following example shows an actual command/response message pair, correctly transmitted and incorrectly transmitted.

Correct Transmission


Command
01 0F 00 F0 <cr>

Response
01 4F 00 EE 07 BB <cr>

Error Checking (continued)

Incorrect Transmission

```
Command
01  0F  00  EE  <cr>
|   |   |   |
|   |   |   Incorrect Chksum
|   |   Status
|   Command
Address

Response
01  CF  00  20  <cr>
|   |   |   |
|   |   |   Response Cheksum
|   |   Status
|   Response if Chksum Incorrect
Address
```

In the command string, the checksum was found by adding in hexadecimal 01 + 0F + 00 to get 10. The 2's complement of this is F0 (100 - 10 = F0).

This checksum is then inserted at the end of the message. To check the checksum in the received message, all of the digit pairs are added (01 + 4F + 00 + EE +07 + BB = 0). If the least significant byte of the sum is not 0, then a checksum error has occurred.

Carriage Return <cr>

Each command and response is terminated with the carriage return character (HEX 0D). All other non-numeric characters are ignored.

Example Command/Response Messages

This section defines all of the commands in the ASCII Line Mode. The response is included for each command, along with an example to demonstrate its use. The numbers and letters in the string are actual ASCII characters. These example messages contain spaces separating the numbers, which is for clarity only. The spaces are not included in the actual transmitted messages.

Read MENU Values

This command returns multiple menu values starting at the menu and page number specified. It returns 4 bytes (8 hex digits) of data for each MENU defined for the response menu value. If more MENUs are requested than exist on the given page, or are allowed to be accessed on that page, only those menus available will be returned. If more MENUs are requested than can be returned in a maximum length response, only those that will fit in the response will be returned.

command: <adr> 41 <sts> <menu value> <decpl> <units> chksum> <cr>

Variables: menu value: -32768 to 32767 (decimal) / 0000 to FFFF (4digits in HEX, lsb first)

response: <adr> 41 <sts> <menu value> <decpl> <units> <chksum> <cr>
◄------ repeated nn/2 times ------►

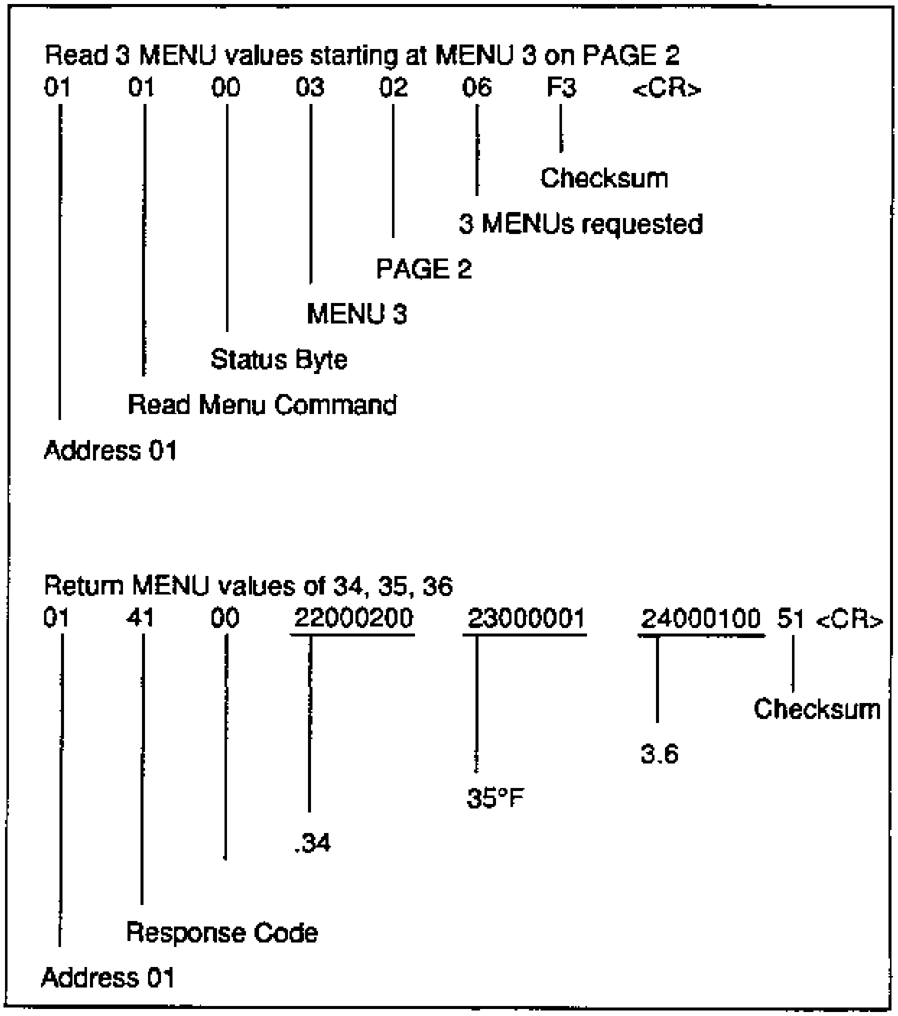
Variables: menu value: -32768 to 32767 (decimal) / 0000 to FFFF (4 digits in HEX, lsb first)
decpl: 00-03 (number of decimal places after converting to decimal)
units: 00-03 (deg. F, deg. C, %)

When a menu value is returned in a response, it consists of 4 bytes, the first two containing a 16 bit numerical value for menu, and the third and fourth containing a decimal place code and a units code. This format is summarized in the diagram below.

menu value			
data (lsb): 0-FF	data (msb): 0-FF	dec. places: 0-3	units: 0-3
		0 = no units 1 = deg. F 2 = deg. C 3 = %	

Read MENU Values
(continued)

Example:



Read 3 MENU values starting at MENU 3 on PAGE 2

```
01   01   00   03   02   06   F3   <CR>
```

Return MENU values of 34, 35, 36

```
01   41   00   22000200   23000001   24000100 51 <CR>
```

Write MENU Values

This command writes multiple values into the MENUs and PAGEs specified. Each MENU value is two bytes (4 HEX digits) in length with the least significant byte first.

When the Write MENU command is received, the controller will check to verify that the value to be placed into each MENU is within the limits allowed for that MENU. If any of the new values are out of limits, then an error code (02) is returned in the status byte and no values will be written to any MENUs. If no error occurs, all of the new values will be written to the MENUs.

The new MENU values contain no decimal places and are assumed to be adjusted for the number of decimal places returned by the Read MENU command. For example, if a MENU has 1 decimal place, and a value of 2.4 is to be written to that MENU, then 24 (18 HEX) should be entered with the Write MENU command.

Write MENU Values (continued)

command: <adr> 08 <sts> <menu#> <page#> <menu value> <chksum> <cr>
◄------ repeated ------►

Variables: page#: 0 to 255 (decimal) / 00 to FF (hex)
menu#: 0 to 255 (decimal) / 00 to FF (hex)
menu value: -32768 to 32767 (decimal) / 0000 to FFFF (4 digits in HEX, lsb first)

response: <addr> 48 <sts> <chksum> <cr>

Example:

```
Write 53, 55 & 1050 to MENUs starting at MENU 3 on PAGE 2
10    08    00    03    02    35 00    37 00    1A 04 59  <cr>
|     |     |     |     |       |        |        |    |
|     |     |     |     |       |        |        |    Chksum
|     |     |     |     |       |        |        1050
|     |     |     |     |       |        55
|     |     |     |     |       53
|     |     |     |     Page 2
|     |     |     MENU 3
|     |     Status Byte
|     Write Menu Command
Address


Response, no error
10    48    00    A8    <cr
|     |     |     |
|     |     |     Chksum
|     |     Status Byte
|     Response Code
Address
```

Reset Program

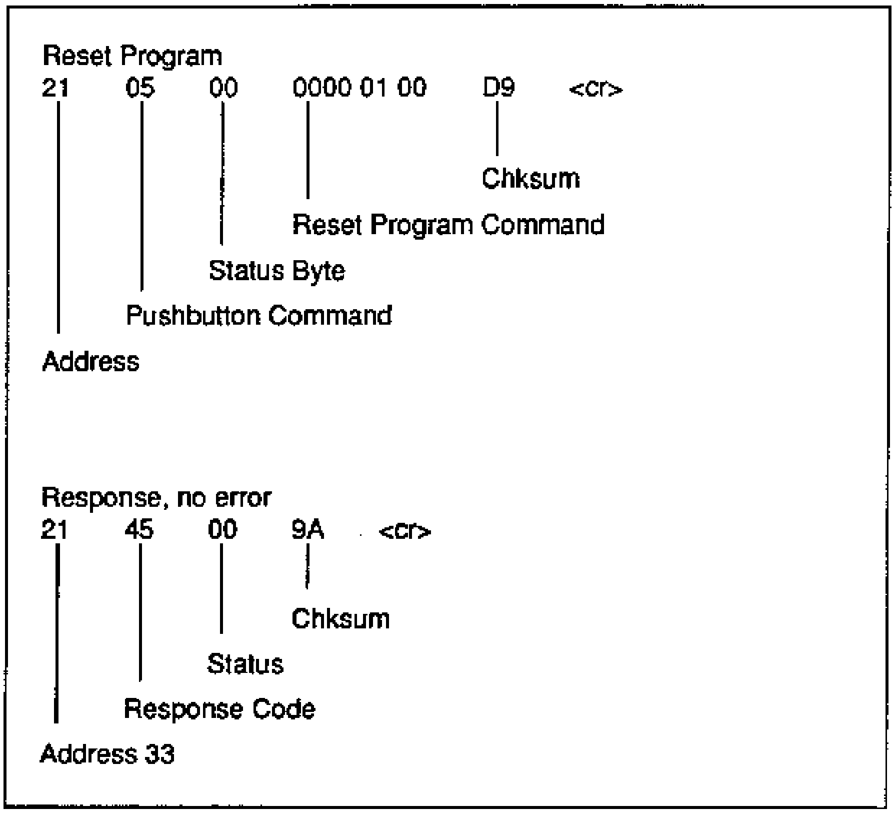
This command resets the program currently executing in the same way the front panel RESET pushbutton does. In the controller models 2001 and 2002 this command automatically returns the unit to standby.

command: <adr> 05 <sts> 0000 0100 <chksum> <cr>

response: <adr> 45 <sts> <chksum> <cr>

Reset Program
(continued)

Example:



Reset Program

21 05 00 0000 01 00 D9 <cr>

Chksum

Reset Program Command

Status Byte

Pushbutton Command

Address

Response, no error

21 45 00 9A <cr>

Chksum

Status

Response Code

Address 33

Start/Continue Program

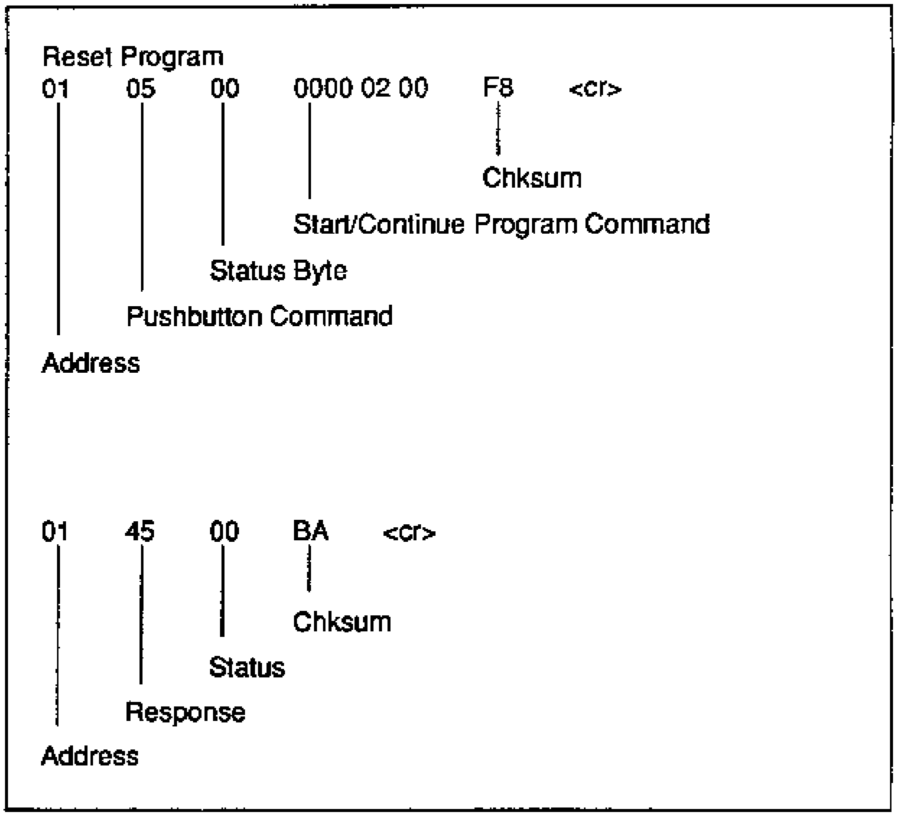
This command starts or resumes program execution in the same way as the controller's front panel pushbutton.

command: <adr> 05 <sts> 0000 0200 <chksum> <cr>

response: <adr> 45 <sts> <chksum> <cr>

Start/Continue Program
(continued)

Example:



Reset Program

01 05 00 0000 02 00 F8 <cr>

Chksum

Start/Continue Program Command

Status Byte

Pushbutton Command

Address

01 45 00 BA <cr>

Chksum

Status

Response

Address

Hold Program

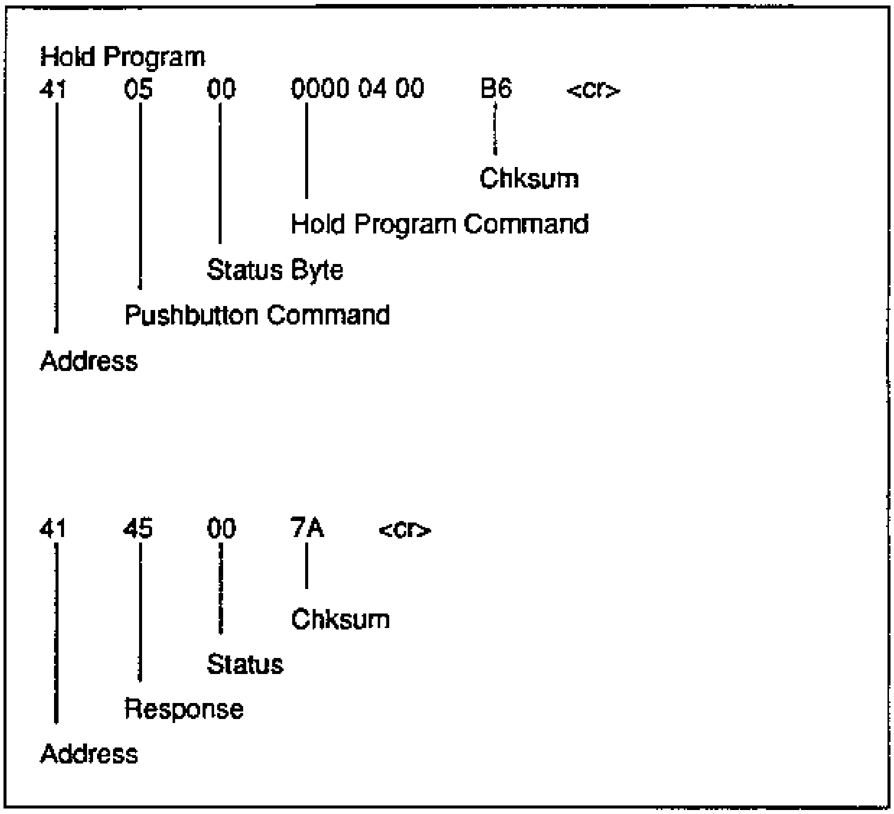
This command holds the program currently running in the same way as the controller's front panel pushbutton.

command: <adr> 05 <sts> 0000 0400 <chksum> <cr>

response: <adr> 45 <sts> <chksum> <cr>

Hold Program
(continued)

Example:



Lock/Unlock Front Panel

These commands will disable or re-enable the pushbuttons on the front panel of the controller.

LOCK command: <adr> 05 <sts> 0000 0800 <chksum> <cr>
UNLOCK command: <adr> 05 <sts> 0000 0008 <chksum> <cr>

response: <adr> 45 <sts> <chksum> <cr>

Lock/Unlock Front Panel
(continued)

Example:

```
Lock front panel, address 01
01      05      00      0000 08 00      F2      <cr>
|       |       |       |               |
|       |       |       |               Chksum
|       |       |       Lock Command
|       |       Status Byte
|       Pushbutton Command
Address


01      45      00      BA      <cr>
|       |       |       |
|       |       |       Chksum
|       |       Status
|       Response
Address
```

Example:

```
Unlock front panel, address 03
03      05      00      0000 00 08      F0      <cr>
|       |       |       |               |
|       |       |       |               Chksum
|       |       |       Unlock Command
|       |       Status Byte
|       Pushbutton Command
Address


03      45      00      B8      <cr>
|       |       |       |
|       |       |       Chksum
|       |       Status
|       Response
Address
```

Change Access Security Code

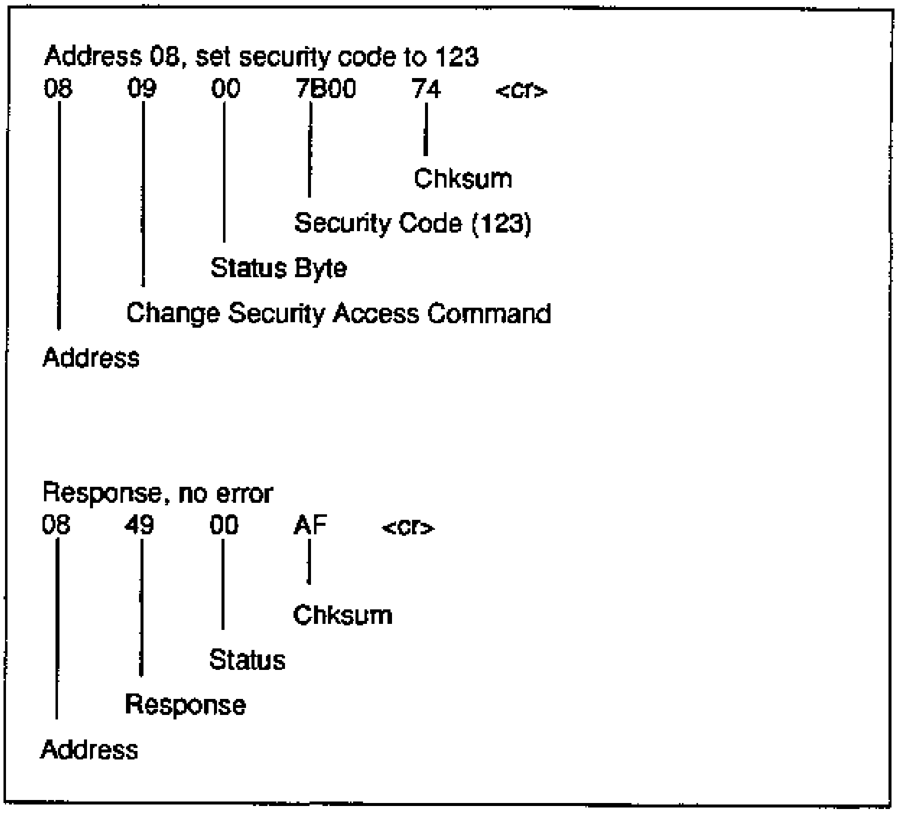
This command allows the computer to have a different security access level to view and change menus, but leaves the controller front panel at the security level determined by the security menu value.

command: <adr> 09 <sts> <security code> <chksum> <cr>

Variables: access code: -32768 to 32767 (decimal) / 0000 to FFFF (4 digits in HEX, lsb first)

response: <adr> 49 <sts> <chksum> <cr>

Example:



Address 08, set security code to 123

```
08     09     00     7B00     74     <cr>
```

Response, no error

```
08     49     00     AF     <cr>
```

Manual Output Adjust

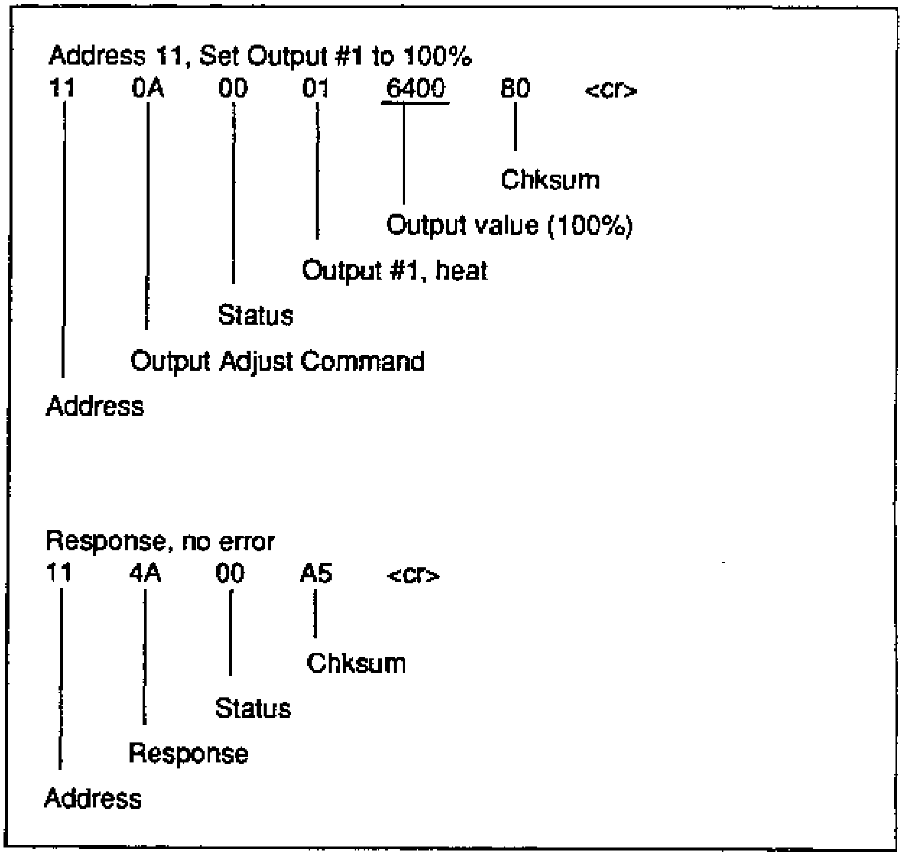
This command allows manual adjustment of a controller output. After this command is received, the output remains under manual control until the resume auto output command is received.

command: <adr> 0A <sts> <oo#> <output value> <chksum> <cr>

Variables: oo#: 01 = output #1 (heat), 02 = output #2 (cool)
output val: -32768 to 32767 (decimal) / 0000 to FFFF (4 digits in HEX, lsb first)

response: <adr> 4A <sts> <chksum> <cr>

Example:



Address 11, Set Output #1 to 100%

```
11   0A   00   01   6400   80   <cr>
```

Chksum

Output value (100%)

Output #1, heat

Status

Output Adjust Command

Address

Response, no error

```
11   4A   00   A5   <cr>
```

Chksum

Status

Response

Address

Resume Auto Output

This command causes automatic output control to resume for the specified output.

command: <adr> 0B <sts> <oo#> <chksum> <cr>

Variables: oo#: 01 to FF (see "Manual Output Adjust" above)

response: <adr> 4B <sts> <chksum> <cr>

Resume Auto Output
(continued)

Example:

```
Resume auto control on Output #1, Address 32
32   0B   00   01   C2   <cr>
|    |    |    |    |
|    |    |    |    Checksum
|    |    |    Output #01, heat
|    |    Status
|    Command, resume auto control
Address 32


Response, no error
32   4B   00   83   <cr>
|    |    |    |
|    |    |    Chksum
|    |    Status
|    Response
Address
```

Return Alarm Status

This command returns the status of all alarms. The response includes the number of alarms followed by one byte for each alarm, which is 0 if the alarm is off or 1 if the alarm is on.

command: <adr> 0C <sts> <chksum> <cr>

```
response: <adr> 4C <sts> <nn> <st> ... <chksum> <cr>
                             <-nn bytes->
```

Variables: nn: number of alarms
st: alarm status; 00-off, 01-on

Return Alarm Status
(continued)

Example:

```
Send Alarm Status of Address 27
27    0C    00    CD    <cr>
|     |     |     |
|     |     |     Chksum
|     |     Status
|     Return Alarm Status Command
Address


Response, Alarm 1 - off, Alarm 2 - on
27    4C    00    02    00    01    8A    <cr>
|     |     |     |     |     |     |
|     |     |     |     |     |     Chksum
|     |     |     |     |     Alarm #2 is on
|     |     |     |     Alarm #1 is off
|     |     |     2 Alarms total in controller
|     |     Status
|     Response
Address
```

Return Maximum Viewable and Adjustable MENU Numbers on a PAGE

This command returns the maximum viewable and adjustable MENU numbers for a given page. The viewable and adjustable MENU numbers depends on the current security access level that has been set.

command: <adr> 0D <sts> <page> <chksum> <cr>

Variables: page: page number to return limits for 0-255 (decimal) / 00-FF (HEX)

response: <adr> OD <sts> <page> <chksum> <cr>

Variables: max view: maximum viewable menu number 0-255 (decimal) / 00-FF (HEX)
max adj: maximum adjustable menu number 0-255 (decimal) / 00-FF (HEX)

Return Maximum Viewable and Adjustable MENU Numbers on a PAGE
(continued)

Example:

```
Get PAGE 1 view and adjust limits, Address 04
04    4D    00    01    EE    <cr>
|     |     |     |     |
|     |     |     |     Chksum
|     |     |     Page 01
|     |     Status
|     Return Max Menus Command
Address 04


Response, Max. view - 19, Max. adjust - 15
04    4D    00    13    0F    8D    <cr>
|     |     |     |     |     |
|     |     |     |     |     Chksum
|     |     |     |     15 Adjustable MENUs
|     |     |     19 Viewable MENUs
|     |     Status
|     Response
Address 04
```

Return MENU Adjust Limits

This command returns the adjust limits for a requested PAGE and MENU. Two 16 bit values are returned for the upper limit and the lower limit.

command: <adr> 0E 00 <menu> <page> <chksum> <cr>

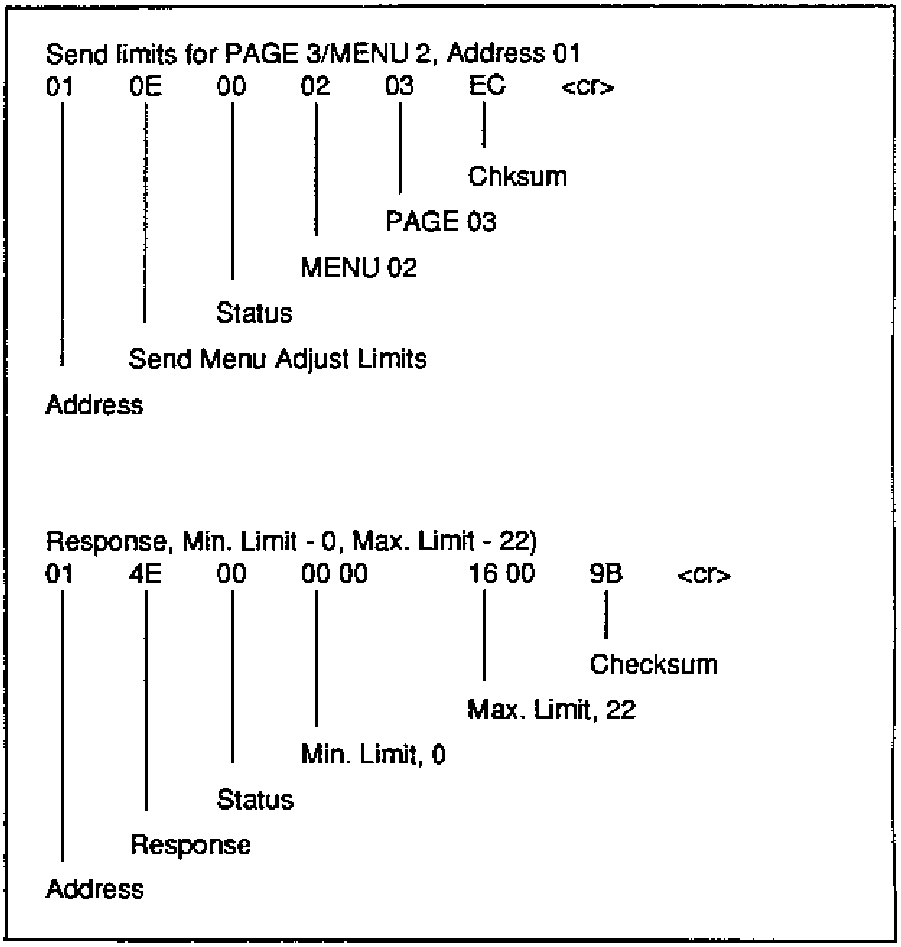
Variables: menu: 0 to 255 (decimal) / 00 - FF (HEX)
page: 0 to 255 (decimal) / 00 - FF (HEX)

response: <adr> 4E <sts> <min limit> <max limit> <chksum> <cr>

Variables: min limit: -32768 to 32767 (decimal) / 0000 to FFFF (4 digits HEX, lsb first)
max limit: -32768 to 32767 (decimal) / 0000 to FFFF (4 digits HEX, lsb first)

Return MENU Adjust Limits (continued)

Example:



Send limits for PAGE 3/MENU 2, Address 01

```
01    0E    00    02    03    EC    <cr>
```

Chksum

PAGE 03

MENU 02

Status

Send Menu Adjust Limits

Address

Response, Min. Limit - 0, Max. Limit - 22)

```
01    4E    00    00 00       16 00    9B    <cr>
```

Checksum

Max. Limit, 22

Min. Limit, 0

Status

Response

Address

Return Model Number

This command returns the controller model number of the controller at this address. The model number is a 16 bit integer.

command: <adr> 0F <sts> <chksum> <cr>

response: <adr> 4F <sts> <nnnn> <chksum> <cr>

Variables:	nnnn: model number -32768 to 32767 (decimal)
Variables:	0000 to FFFF (4 digits HEX, lsb first)

Return Model Number
(continued)

Example:

```
Send back controller model number at address 09
09    0F    00    E8    <cr>
|     |     |     |
|     |     |     Chksum
|     |     Status
|     Return Model Number Command
Address


Response, Model number = CN2030
09    4F    00    EE 07    B3    <cr>
|     |     |     |        |
|     |     |     |        Chksum
|     |     |     Model CN2030
|     |     Status
|     Response
Address
```

Return Event Status

This command returns the status of all events. The response includes the number of events followed by one byte for each event—0 if the event is off or 1 if the event is on.

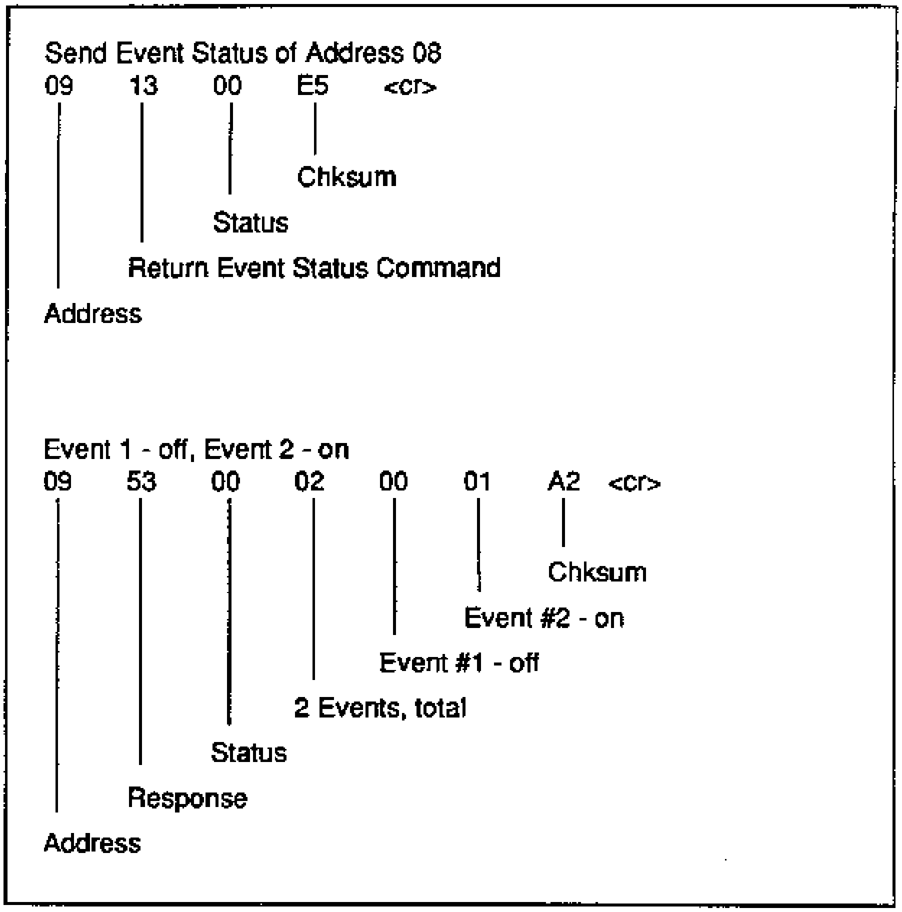
command: <adr> 13 <sts> <chksum> <cr>

response: <adr> 53 <sts> <nn> <st> ... <st> <chksum> <cr>
<-n bytes->

Variables: nn: numbers of events
st: event status; 00-off, 01-on

Return Event Status
(continued)

Example:



Send Event Status of Address 08

09 13 00 E5 <cr>

Event 1 - off, Event 2 - on

09 53 00 02 00 01 A2 <cr>

■ Using BASIC Subroutines

BASIC programs may be used to generate command messages and interpret response strings for ASCII Line Mode set-up. See example BASIC programs in Appendix 2, page 61.

6

Computer Interface Mode

■ Controller Set-Up

The Computer Interface Mode (CPIF), uses a byte oriented protocol to coordinate communication between the computer and multiple controllers in a multidrop network configuration. The CPIF mode corresponds with Allen-Bradley DATA HIGHWAY protocol. The CPIF mode of communication is faster and has better error recovery provisions than the ASCII mode, but is more difficult to implement. It is also essential that you have access to additional Allen-Bradley documentation (not available from OMEGA).

To configure the controller(s) for Computer Interface (CPIF) communication, you must make several selections on the Digital Communications PAGE of your controller(s). These PAGEs are shown on page 11 of this manual.

1. Select “3 = CPIF” at MENU 1.
2. Select the appropriate interface, baud rate and parity at MENUs 7-9.
3. Enter the address of the controller at MENU 11.
4. **Be sure to enter the value “1” at MENU 10 when all selections have been made to reconfigure the controller set-up.**

■ Protocol Definition

The communications protocol used to control the information transfer in the computer interface mode is a subset of the ANSI X3.28 standard. Sub-category D1 is used for the message transfer procedure. There is no establishment and termination control procedure. This corresponds to the full-duplex protocol used by Allen-Bradley for their Data Highway communications.

Protocol Control Characters

This protocol definition uses several control characters to indicate certain conditions. They are summarized below.

Control Character	Function
STX - start of text	Signals the beginning of a sequence of data bytes.
ETX - end of text	Signals the end of a sequence of data bytes.
BCC - block check character	Character added immediately following the ETX to facilitate error detections. See "Checksum Calculation" example, p. 25.
ENQ - enquiry	Requests a response from a secondary station after time-out.
ACK - acknowledgment	Acknowledges the successful reception of a message.
DLE - data link escape	Precedes control codes to delimit them from data.

All data is transmitted in binary form and therefore requires a way of distinguishing control characters. All control characters are preceded by a DLE. A DLE data byte (10 hex) is also preceded by DLE to identify it as data (this "extra" DLE is not included in the checksum calculation, as shown in the READ MENU example on page 27).

The block check character is a one byte checksum that is appended to each command and response message. It is calculated by adding up each byte starting with the destination address and ending with the last byte ahead of the DLE ETX. The extra DLE in a DLE DLE sequence will not be added to the checksum. After transmitting the message and the DLE STX, the checksum is limited to 1 byte and negated (2's complement). This byte is then transmitted, completing the transmit message.

Normal Protocol Operation

The computer controls the protocol state of the network. All command/response transactions are initiated by the control computer. The controller being addressed is designated as the secondary station.

The following example demonstrates the normal operation of this protocol. The host refers to the host computer and "sec" refers to the controllers which are secondary stations on the network. The station that is transmitting along with the data being transmitted is shown in the example.

Station	Transmitted data	Comments
host	<DLE> <STX>	start of transmission
host	<dst> <command data>	destination address, followed by the remainder of the command data
host	<DLE> <ETX> <BCC>	end of transmission
sec	<DLE> <ACK>	acknowledgement
sec	<DLE> <STX>	start of transmission
sec	<dst> <response data>	destination address, followed by the remainder of the response data
sec	<DLE> <ETX> <BCC>	end of transmission
host	<DLE> <ACK>	acknowledgement
sec	<DLE> <ETX> <BCC>	end of transmission
host	<DLE> <ACK>	acknowledgement

Normal Protocol Operation (continued)

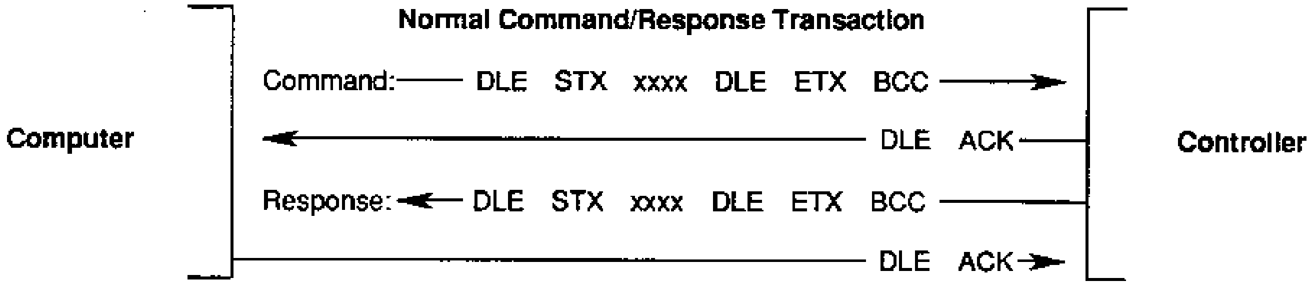
Each controller on the network has to listen for a <DLE> <STX> <dst> sequence (where <dst> is the assigned address to the controller) to determine if the message is addressed to the controller. If the address received in the message and the assigned address of the controller do not agree, then all data up until the next <DLE> <STX> can be ignored.

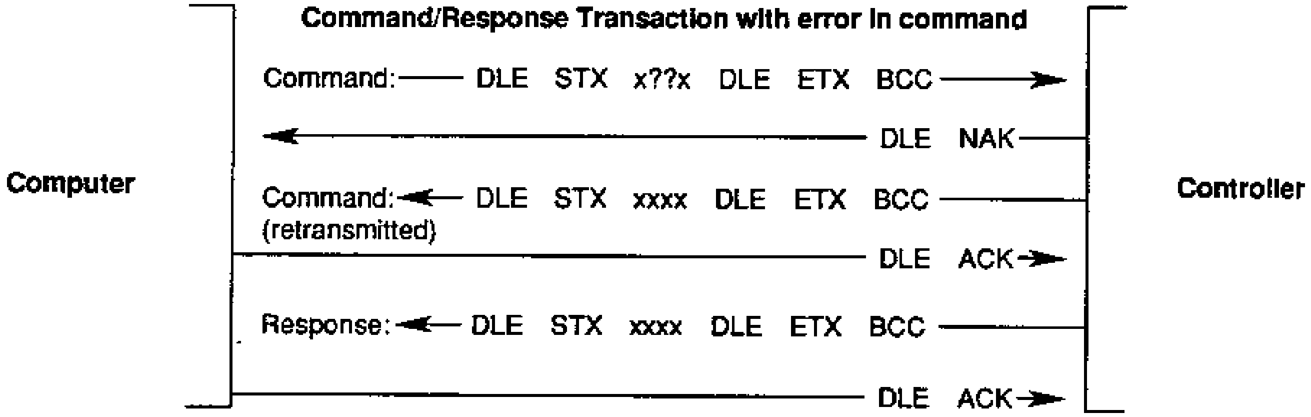
Half-Duplex and Full-Duplex Differences

Half-duplex operation will not allow simultaneous transmission of messages in both directions. Consequently, the host computer and the controller need to know when it is time to either transmit or receive. This is accomplished by maintaining the current state of the protocol interaction.

Full-duplex operation makes it possible to transmit in both directions simultaneously. Therefore, it is possible to send a command while receiving a response from a previous command. It is possible to overlap commands and responses only when multiple commands are being sent to the same controller. Communication with a different network address can only take place after all outstanding responses have been returned from the controller with the current address.

Protocol Sequence Diagrams


Normal Command/Response Transaction
Computer
Controller
Command: DLE STX xxxx DLE ETX BCC
DLE ACK
Response: DLE STX xxxx DLE ETX BCC
DLE ACK


Command/Response Transaction with error in command
Computer
Controller
Command: DLE STX x??x DLE ETX BCC
DLE NAK
Command: DLE STX xxxx DLE ETX BCC
(retransmitted)
DLE ACK
Response: DLE STX xxxx DLE ETX BCC
DLE ACK

■ Command/Response Message Format

The commands and responses in the computer interface mode have a specific format as described below. These command and response messages are enclosed by the DLE STX and DLE ETX BCC control characters.

Message Format

<dst> <src> <cc> <sts> <tns> <data>

dst: One byte destination address.

src: One byte source address.

cc: Command code (one byte), see page 23.

sts: Status byte (this will always be 00 in the command; in the response, this byte will be 00 if there no error is encountered; an error code will be returned in this byte otherwise). See page 24 for error codes.

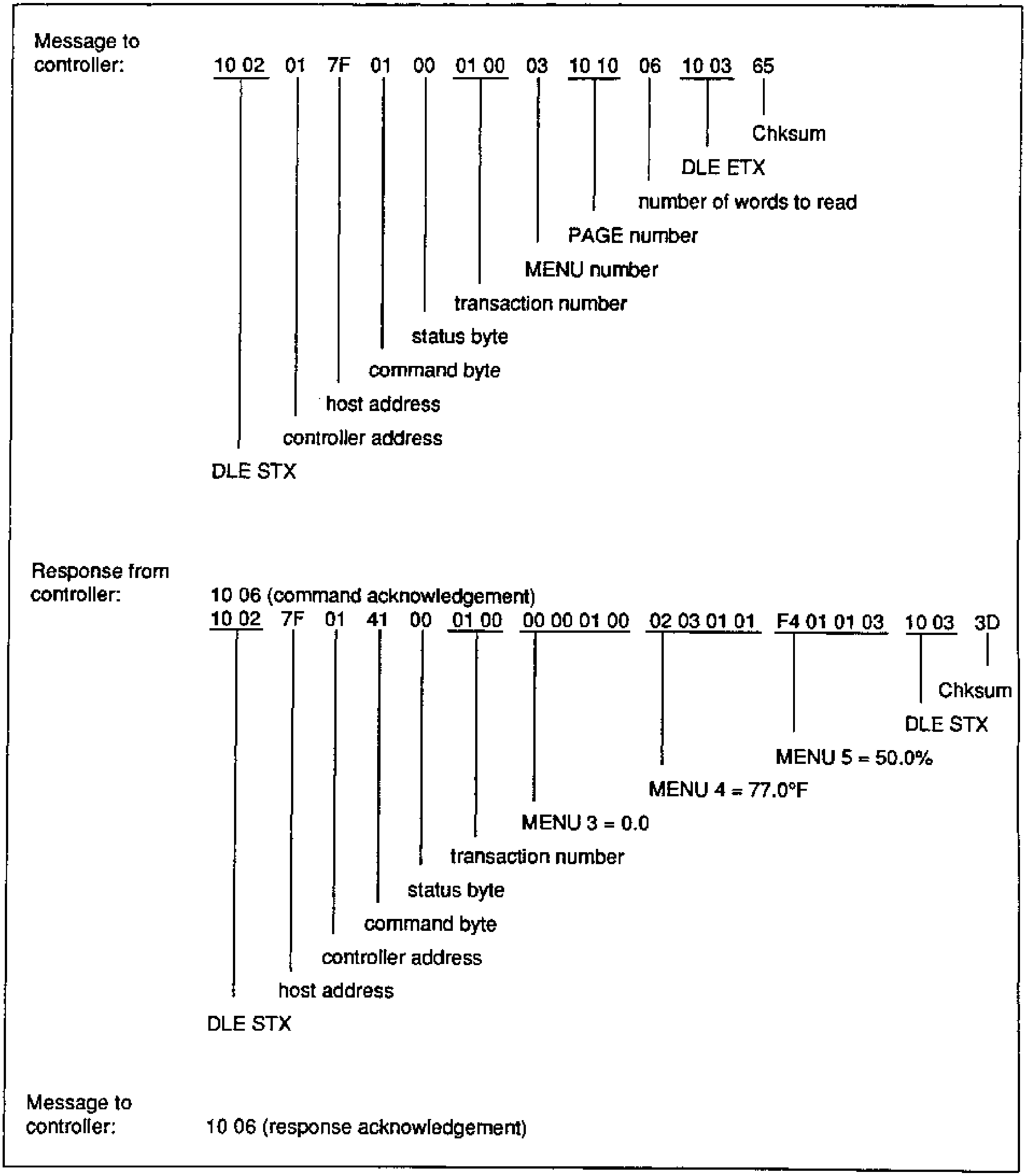
tns: 16 bit transaction code (the same for command/response pair).

data: Variable length data field that is defined for each command or response. Maximum length is 244 bytes. See page 24 for data field formats.

Note: The destination address, command code, status byte and data fields are identical to those in the ASCII Line Mode except that all numbers are transmitted as 8-bit binary values instead of converting to ASCII characters representing hexadecimal numbers.

Complete Example of Read MENU Command

The following description shows the actual bytes (in hex) received and transmitted by the controller to execute the Read Menu command. This command will read 3 MENUs starting at MENU 3 on PAGE 16.



Message to controller: 10 02 01 7F 01 00 01 00 03 10 10 06 10 03 65

- 10 02 — DLE STX
- 01 — controller address
- 7F — host address
- 01 — command byte
- 00 — status byte
- 01 00 — transaction number
- 03 — MENU number
- 10 10 — PAGE number
- 06 — number of words to read
- 10 03 — DLE ETX
- 65 — Chksum

Response from controller: 10 06 (command acknowledgement)

10 02 7F 01 41 00 01 00 00 00 01 00 02 03 01 01 F4 01 01 03 10 03 3D

- 10 02 — DLE STX
- 7F — host address
- 01 — controller address
- 41 — command byte
- 00 — status byte
- 01 00 — transaction number
- 00 00 01 00 — MENU 3 = 0.0
- 02 03 01 01 — MENU 4 = 77.0°F
- F4 01 01 03 — MENU 5 = 50.0%
- 10 03 — DLE STX
- 3D — Chksum

Message to controller: 10 06 (response acknowledgement)

Error Recovery

A primary purpose for protocol utilization is to insure error free transfer of data. The block check character (BCC) provides a moderate check for errors within the message itself. The BCC is computed by taking the 2's complement of the 8-bit sum of all data bytes between the DLE STX and the DLE ETX. When a message is received and the computed checksum does not agree with the received checksum, negative acknowledgement can be returned to cause the message to be re-transmitted.

Protocol errors and time-outs are handled by a recovery sequence that places all nodes of the network in a known state. For a complete explanation of error recovery sequences, refer to Allen-Bradley Data Highway Protocol.

Appendix 1

ASCII/HEX Chart

ASCII	Decimal	HEX	Binary
NUL	0	00	00000000
SOH	1	01	00000001
STX	2	02	00000010
ETX	3	03	00000011
EOT	4	04	00000100
ENQ	5	05	00000101
ACK	6	06	00000110
BEL	7	07	00000111
BS	8	08	00001000
HT	9	09	00001001
LF	10	0A	00001010
VT	11	0B	00001011
FF	12	0C	00001100
CR	13	0D	00001101
SO	14	0E	00001110
SI	15	0F	00001111
DLE	16	10	00010000
DC1	17	11	00010001
DC2	18	12	00010010
DC3	19	13	00010011
DC4	20	14	00010100
NAK	21	15	00010101
SYN	22	16	00010110
ETB	23	17	00010111
CAN	24	18	00011000
EM	25	19	00011001
SUB	26	1A	00011010
ESC	27	1B	00011011
FS	28	1C	00011100
GS	29	1D	00011101
RS	30	1E	00011110
US	31	1F	00011111
SP	32	20	00100000
!	33	21	00100001
"	34	22	00100010
#	35	23	00100011
$	36	24	00100100
%	37	25	00100101
&	38	26	00100110
'	39	27	00100111
(	40	28	00101000
)	41	29	00101001
*	42	2A	00101010
+	43	2B	00101011
,	44	2C	00101100
-	45	2D	00101101
.	46	2E	00101110
/	47	2F	00101111

ASCII	Decimal	HEX	Binary
0	48	30	00110000
1	49	31	00110001
2	50	32	00110010
3	51	33	00110011
4	52	34	00110100
5	53	35	00110101
6	54	36	00110110
7	55	37	00110111
8	56	38	00111000
9	57	39	00111001
:	58	3A	00111010
;	59	3B	00111011
<	60	3C	00111100
=	61	3D	00111101
>	62	3E	00111110
?	63	3F	00111111
@	64	40	01000000
A	65	41	01000001
B	66	42	01000010
C	67	43	01000011
D	68	44	01000100
E	69	45	01000101
F	70	46	01000110
G	71	47	01000111
H	72	48	01001000
I	73	49	01001001
J	74	4A	01001010
K	75	4B	01001011
L	76	4C	01001100
M	77	4D	01001101
N	78	4E	01001110
O	79	4F	01001111
P	80	50	01010000
Q	81	51	01010001
R	82	52	01010010
S	83	53	01010011
T	84	54	01010100
U	85	55	01010101
V	86	56	01010110
W	87	57	01010111
X	88	58	01011000
Y	89	59	01011001
Z	90	5A	01011010
[	91	5B	01011011
\	92	5C	01011100
]	93	5D	01011101
^	94	5E	01011110
_	95	5F	01011111

ASCII	Decimal	HEX	Binary
`	96	60	01100000
a	97	61	01100001
b	98	62	01100010
c	99	63	01100011
d	100	64	01100100
e	101	65	01100101
f	102	66	01100110
g	103	67	01100111
h	104	68	01101000
i	105	69	01101001
j	106	6A	01101010
k	107	6B	01101011
l	108	6C	01101100
m	109	6D	01101101
n	110	6E	01101110
o	111	6F	01101111
p	112	70	01110100
q	113	71	01110101
r	114	72	01110010
s	115	73	01110011
t	116	74	01110100
u	117	75	01110101
v	118	76	01110110
w	119	77	01110111
x	120	78	01111000
y	121	79	01111001
z	122	7A	01111010
{	123	7B	01111011
|	124	7C	01111100
}	125	7D	01111101
~	126	7E	01111110
DEL	127	7F	01111111
	128	80	10000000
	129	81	10000001
	130	82	10000010
	131	83	10000011
	132	84	10000100
	133	85	10000101
	134	86	10000110
	135	87	10000111
	136	88	10001000
	137	89	10001001
	138	8A	10001010
	139	8B	10001011
	140	8C	10001100
	141	8D	10001101
	142	8E	10001110
	143	8F	10001111

ASCII	Decimal	HEX	Binary
	144	90	10010000
	145	91	10010001
	146	92	10010010
	147	93	10010011
	148	94	10010100
	149	95	10010101
	150	96	10010110
	151	97	10010111
	152	98	10011000
	153	99	10011001
	154	9A	10011010
	155	9B	10011011
	156	9C	10011100
	157	9D	10011101
	158	9E	10011110
	159	9F	10011111
	160	A0	10100000
	161	A1	10100001
	162	A2	10100010
	163	A3	10100011
	164	A4	10100100
	165	A5	10100101
	166	A6	10100110
	167	A7	10100111
	168	A8	10101000
	169	A9	10101001
	170	AA	10101010
	171	AB	10101011
	172	AC	10101100
	173	AD	10101101
	174	AE	10101110
	175	AF	10101111
	176	B0	10110000
	177	B1	10110001
	178	B2	10110010
	179	B3	10110011
	180	B4	10110100
	181	B5	10110101
	182	B6	10110110
	183	B7	10110111
	184	B8	10111000
	185	B9	10111001
	186	BA	10111010
	187	BB	10111011
	188	BC	10111100
	189	BD	10111101
	190	BE	10111110
	191	BF	10111111
	192	C0	11000000
	193	C1	11000001
	194	C2	11000010
	195	C3	11000011

ASCII	Decimal	HEX	Binary
	196	C4	11000100
	197	C5	11000101
	198	C6	11000110
	199	C7	11000111
	200	C8	11001000
	201	C9	11001001
	202	CA	11001010
	203	CB	11001011
	204	CC	11001100
	205	CD	11001101
	206	CE	11001110
	207	CF	11001111
	208	D0	11010000
	209	D1	11010001
	210	D2	11010010
	211	D3	11010011
	212	D4	11010100
	213	D5	11010101
	214	D6	11010110
	215	D7	11010111
	216	D8	11011000
	217	D9	11011001
	218	DA	11011010
	219	DB	11011011
	220	DC	11011100
	221	DD	11011101
	222	DE	11011110
	223	DF	11011111
	224	E0	11100000
	225	E1	11100001
	226	E2	11100010
	227	E3	11100010
	228	E4	11100100
	229	E5	11100101
	230	E6	11100110
	231	E7	11100111
	232	E8	11101000
	233	E9	11101001
	234	EA	11101010
	235	EB	11101011
	236	EC	11101100
	237	ED	11101101
	238	EE	11101111
	239	EF	11110000
	240	F0	11110001
	241	F1	11110010
	242	F2	11110011
	243	F3	11110100
	244	F4	11110101
	245	F5	11110110
	246	F6	11110110
	247	F7	11110111

ASCII	Decimal	HEX	Binary
	248	F8	11111000
	249	F9	11111001
	250	FA	11111010
	251	FB	11111011
	252	FC	11111100
	253	FD	11111101
	254	FE	11111110
	255	FF	11111111

Appendix 2

Basic Programs

```
10 '**********************************************************
20 '*
30 '*   ******    EXAMPLE2.BAS    ******
40 '*
50 '* This program demonstrates how to utilize the
60 '* BASIC subroutines included here to communicate
70 '* with the  Omega controller.  For more details
80 '* on the available subroutines and the variables
90 '* used in each routine, look at the summary begin-
100 '* ning at line 40000 and the description preceding
110 '* each individual routine.
120 '*
130 '* In order to communicate with this program, it
140 '* will be necessary to set the controller up as
150 '* follows: First the communication mode or
160 '* function should be set to LinE (menu 1 on the
170 '* communication page).  This selects the ASCII line
180 '* mode.  Next move to menu 8 and select a baud
190 '* rate of 9600.  Now the address of the controller
200 '* can be set in menu 11.  Even if you don't change
210 '* it, you'll need to check it in order to be able
220 '* to tell the computer what address to talk to.
230 '* Finally, move back to menu 10 and select 1 to
240 '* reconfigure the communications with the new
250 '* settings.
260 '*
270 '* This program is simply an example of how to use
280 '* subroutines in your customized application.  The
290 '* subroutines start at line 40000.  To include them
300 '* in another program, copy lines 40000 through 54070
310 '* of the file into the new program file.  Then
320 '* use GOSUB to call the appropriate function.
330 '* The initialize communications (line 41000) and
340 '* send command (line 42000) routines are written to
350 '* utilize the COM1 serial port.  If some other
360 '* communication device is to be used, the initial-
370 '* ization and message input and output lines can be
380 '* easily modified.  The transmitted message is
390 '* simply an ASCII string.
400 '*
410 '************************************************************
420 '
1000 GOSUB 41000   'COMM INITIALIZATION ROUTINE
1010 '
1020 INPUT "CONTROLLER ADDRESS:",UNIT.NUM%
1030 '
1040 M.VALUE=736         'SET SECURITY ACCESS CODE
1050 GOSUB 48000
1120 '
1130 ' Read menu values
1140 INPUT "PAGE,MENU:",PG.NUM%,MENU.NUM%
1150 IF (PG.NUM%=0)AND(MENU.NUM%=0) THEN 1190
1160 GOSUB 43000
1170 PRINT "VALUE=";M.VALUE;M.UNITS$
1180 GOTO 1140
1185 '
1190 ' Write menu values
1200 INPUT "PAGE,MENU:",PG.NUM%,MENU.NUM%
1210 IF(PG.NUM%=0)AND(MENU.NUM%=0) THEN 1270
1220 INPUT "VALUE ", M.VALUE
```

```
1230 GOSUB 46000
1240 GOTO 1200
1250 '
1260 ' Ramp-soak program commands
1270 INPUT "START PROGRAM", M.VALUE
1280 GOSUB 50000
1290 INPUT "HOLD PROGRAM", M.VALUE
1300 GOSUB 51000
1310 INPUT "RESET PROGRAM", M.VALUE
1320 GOSUB 49000
1330 '
1340 ' 2030 specific routines
1350 INPUT "ENTER PROGRAM NUMBER:", PRG.NUM%
1360 GOSUB 52000 'Set active program on 2030
1370 GOSUB 53000 'Read channel 1 process variable on 2030
1380 PRINT "CHANNEL 1 PV = "; CH1PV;" "; M.UNITS$
1390 GOSUB 54000 'Read channel 2 process variable on 2030
1400 PRINT "CHANNEL 2 PV = "; CH2PV;" "; M.UNITS$
1410 END
40000 '
40010 '
40020 '
40030 '******************************************************
40040 '*
40050 '* The following subroutines communicate with the
40060 '* process controllers to allow remote operation and monitoring.
40070 '* The following is a list of these functions
40080 '* along with their line number.  The parameters used by
40090 '* each routine are described in the comments preceding it.
40095 '* The last three routines are specific to the 2030 controller.
40100 '*
40110 '* 41000 - Initialize communications
40120 '* 42000 - Send command string and get response
40130 '* 43000 - Read menu value
40140 '* 46000 - Write menu value
40150 '* 48000 - Change security access level
40160 '* 49000 - Reset Ramp/Soak program
40170 '* 50000 - Start Ramp/Soak program
40180 '* 51000 - Hold Ramp/Soak program
40190 '* 52000 - Set current program number
40192 '* 53000 - Read Channel 1 process variable
40193 '* 54000 - Read Channel 2 process variable
40200 '*
40210 '******************************************************
40220 '
40230 '
40240 '
40800 '******************************************************
40810 '*
40820 '* THIS ROUTINE WILL INITIALIZE THE COMMUNICATIONS
40830 '* PORT THAT WILL BE USED TO COMMUNICATE WITH THE
40840 '* CONTROLLER.  THIS VERSION USES THE COM1 SERIAL
40845 '* PORT.
40850 '*
40860 '******************************************************
40870 '
41000 OPEN "COM1:9600,N,8,1,CS,DS" AS 1
41010 RETURN
41020 '
41030 '
```

```
41040 '
41800 '*********************************************************
41810 '* SEND COMMAND AND INPUT RESPONSE ROUTINE
41820 '*
41830 '* THIS ROUTINE WILL USE THE COMMAND IN CMD.BUF$ TO
41840 '* COMPUTE A CHECKSUM AND WILL THEN TRANSMIT THE STRING
41850 '* THE RESPONSE STRING WILL BE RETURNED IN RSP.BUF$
41860 '* THE CHECKSUM WILL BE REMOVED BEFORE RETURNING.
41870 '*
41880 '* This routine uses the following variables
41890 '*      CMD.BUF$ - command string
41900 '*      RSP.BUF$ - response string
41910 '*      BYTE.COUNT%
41920 '*      CHECKSUM%
41930 '*      T1%
41940 '*      T2%
41950 '*
41960 '*********************************************************
41970 '
41980 ' Compute the transmit message checksum
41990 '
42000 CHECKSUM% = 0
42010 FOR BYTE.COUNT% = 1 TO LEN(CMD.BUF$)/2
42020 T1% = ASC(MID$(CMD.BUF$,(BYTE.COUNT%*2)-1,1))
42030 T2% = ASC(MID$(CMD.BUF$,(BYTE.COUNT%*2)))
42040 IF T1%>64 THEN T1% = T1%+9
42050 IF T2%>64 THEN T2% = T2%+9
42060 CHECKSUM%=CHECKSUM%+((T1% AND 15)*16)+(T2% AND 15)
42070 NEXT BYTE.COUNT%
42080 CHECKSUM% = -CHECKSUM% AND 255
42090 IF CHECKSUM%>15 THEN CMD.BUF$=CMD.BUF$+HEX$(CHECKSUM%) ELSE 42110
42100 GOTO 42120
42110 CMD.BUF$ = CMD.BUF$+"0"+HEX$(CHECKSUM%)
42120 '
42130 ' This section performsthe actual communication with the controller.
42140 ' This version uses the serial communication port.
42150 '
42160 PRINT #1, CMD.BUF$
42170 ' If using RS485 half-duplex communication, it may be necessary
42180 ' to add an extra input at this point to receive the echo of the
42190 ' command.
42200 'INPUT #1, ECHO$
42210 INPUT #1, RSP.BUF$
42220 '
42230 ' Now remove the response checksum
42240 '
42250 IF LEN(RSP.BUF$)>2 THEN RSP.BUF$=LEFT$(RSP.BUF$,LEN(RSP.BUF$)-2)
42260 RETURN
42270 '
42280 '
42290 '
42700 '*********************************************************
42710 '* READ MENU VALUE
42720 '*
42730 '* THIS ROUTINE WILL READ THE VALUE OF A MENU VARIABLE
42740 '* IN THE CONTROLLER AND RETURN IT IN NUMERICAL FORMAT
42750 '* IN THE VARIABLE M.VALUE.  THE PAGE NUMBER WILL BE
42760 '* IN PG.NUM% AND THE MENU NUMBER WILL BE IN MENU.NUM%
42770 '* THE CONTROLLER'S UNIT ADDRESS WILL BE IN UNIT.NUM%
42780 '*
```

```
42790 '* This routine uses the following variables
42800 '*      UNIT.NUM% - address of controller
42810 '*      UNIT.STR$
42820 '*      XSTR$
42830 '*      XVAL%
42840 '*      MENU.NUM% - menu number to read value from
42850 '*      MENU.STR$
42860 '*      PG.NUM% - page number to read value from
42870 '*      PG.STR$
42880 '*      CMD.BUF$ - command string
42890 '*      RSP.BUF$ - response string
42900 '*      M.VALUE - response value
42910 '*      T1%, T2%, T3%, T4%
42920 '*      M.UNITS$ - units string
42930 '*
42940 '*************************************************************
43000 XVAL% = UNIT.NUM% : GOSUB 45000 : UNIT.STR$ = XSTR$
43010 XVAL% = MENU.NUM% : GOSUB 45000 : MENU.STR$ = XSTR$
43020 XVAL% = PG.NUM% : GOSUB 45000 : PG.STR$=XSTR$
43030 CMD.BUF$=UNIT.STR$+"0100"+MENU.STR$+PG.STR$+"02"
43040 GOSUB 42000    'SEND THE COMMAND
43050 IF LEN(RSP.BUF$) < 14 THEN GOSUB 47000 : RETURN
43060 '
43070 T1% = ASC(MID$(RSP.BUF$,9))
43080 T2% = ASC(MID$(RSP.BUF$,10))
43090 GOSUB 44000
43100 T4% = T3%
43110 T1% = ASC(MID$(RSP.BUF$,7))
43120 T2% = ASC(MID$(RSP.BUF$,8))
43130 GOSUB 44000
43140 IF T4% < 128 THEN T4% = T4%*256+T3% ELSE T4%=(T4%-255)*256+(T3%-256)
43150 T1% = ASC(MID$(RSP.BUF$,11))
43160 T2% = ASC(MID$(RSP.BUF$,12))
43170 GOSUB 44000
43180 M.VALUE = T4%/(10!^T3%)
43190 T1% = ASC(MID$(RSP.BUF$,13))
43200 T2% = ASC(MID$(RSP.BUF$,14))
43210 GOSUB 44000
43220 IF T3%=0 THEN 43260
43230 IF T3%=1 THEN 43270
43240 IF T3%=2 THEN 43280
43250 IF T3%=3 THEN 43290
43260 M.UNITS$="": GOTO 43300
43270 M.UNITS$="½F": GOTO 43300
43280 M.UNITS$="½C": GOTO 43300
43290 M.UNITS$= "%"
43300 RETURN
43310 '
43320 '
43330 ' This routine will convert two ASCII characters T1% and T2%
43340 ' that represent a byte value to an integer T3%.
43350 '
44000 IF T1% > 64 THEN T1% = T1%+9
44010 IF T2% > 64 THEN T2% = T2%+9
44020 T3% = ((T1% AND 15)*16)+(T2% AND 15)
44030 RETURN
44040 '
44050 '
44060 ' This routine will convert an 8 bit integer value into
44070 ' ASCII hex representation.
```

```
44080 '
45000 XVAL% = XVAL% AND 255
45010 IF XVAL% > 15 THEN XSTR$=HEX$(XVAL%) ELSE XSTR$="0"+HEX$(XVAL%)
45020 RETURN
45030 '
45040 '
45050 '
45800 '*******************************************************
45810 '* WRITE MENU VALUE
45820 '*
45830 '* THIS ROUTINE WILL WRITE A SPECIFIED VALUE INTO A GIVEN
45840 '* PAGE AND MENU LOCATION IN THE CONTROLLER.  THE VALUE
45850 '* WILL BE IN THE VARIABLE M.VALUE.  THE PAGE NUMBER WILL
45860 '* BE IN PG.NUM% AND THE MENU NUMBER WILL BE IN MENU.NUM%
45870 '* THE CONTROLLER'S UNIT ADDRESS WILL BE IN UNIT.NUM%
45880 '*
45890 '*******************************************************
45900 '
46000 XVAL% = UNIT.NUM% : GOSUB 45000 : UNIT.STR$ = XSTR$
46010 XVAL% = MENU.NUM% : GOSUB 45000 : MENU.STR$ = XSTR$
46020 XVAL% = PG.NUM% : GOSUB 45000 : PG.STR$=XSTR$
46030 CMD.BUF$ = UNIT.STR$+"0100"+MENU.STR$+PG.STR$+"02"
46040 GOSUB 42000   'SEND READ COMMAND
46050 IF LEN(RSP.BUF$) < 14 THEN M.VAL$ = "": RETURN
46060 T1% = ASC(MID$(RSP.BUF$,11))
46070 T2% = ASC(MID$(RSP.BUF$,12))
46080 GOSUB 44000
46090 M.VALUE = M.VALUE* (10!^T3%)
46100 IF M.VALUE > 65535! THEN PRINT "VALUE OUT OF RANGE": RETURN
46110 IF M.VALUE < -65535! THEN PRINT "VALUE OUT OF RANGE": RETURN
46120 T1% = INT(M.VALUE/256!)
46140 IF CINT(M.VALUE)=0 THEN XVAL%=0 : T1% = 0 : GOTO 46160
46150 XVAL% = M.VALUE-(T1%*256!)
46160 GOSUB 45000: VAL.STR$ = XSTR$
46180 XVAL% = T1%: GOSUB 45000
46190 CMD.BUF$ = UNIT.STR$+"0800"+MENU.STR$+PG.STR$+VAL.STR$+XSTR$
46200 GOSUB 42000  'SEND WRITE COMMAND
46210 '
46220 ' This routine will check for an error in the response message
46230 '
47000 T1% = ASC(MID$(RSP.BUF$,5))
47010 T2% = ASC(MID$(RSP.BUF$,6))
47020 GOSUB 44000
47030 IF T3% = 0 THEN RETURN
47040 IF T3%=1 THEN PRINT "SECURITY LEVEL TOO LOW" : RETURN
47050 IF T3%=2 THEN PRINT "VALUE OUT OF RANGE" : RETURN
47060 IF T3%=3 THEN PRINT "CONTROLLER FRONT PANEL IN USE" : RETURN
47070 IF T3%=4 THEN PRINT "INVALID BIT MASK" : RETURN
47080 IF T3%=5 THEN PRINT "INVALID COMMAND" : RETURN
47090 IF T3%=6 THEN PRINT "COMMAND TOO SHORT" : RETURN
47100 IF T3%=7 THEN PRINT "INVALID PAGE NUMBER" : RETURN
47110 IF T3%=8 THEN PRINT "INVALID MENU NUMBER" : RETURN
47120 IF T3%=9 THEN PRINT "INVALID OUTPUT NUMBER" : RETURN
47130 IF T3%=10 THEN PRINT "MANUAL OUTPUT ADJUST DISABLED" : RETURN
47140 IF T3%=11 THEN PRINT "RAMP/SOAK DISABLED" : RETURN
47150 RETURN
47160 '
47170 '
47180 '
47800 '*******************************************************
```

```
47810 '* CHANGE SECURITY ACCESS LEVEL
47830 '* THIS ROUTINE WILL ENTER A NEW SECURITY LEVEL INTO THE
47840 '* CONTROLLER.  THE NEW ACCESS NUMBER WILL BE IN THE
47850 '* VARIABLE M.VALUE.  THE CONTROLLER'S UNIT ADDRESS WILL
47860 '* BE IN UNIT.NUM%
47870 '*
47880 '******************************************************
47890 '
48000 XVAL% = UNIT.NUM% : GOSUB 45000 : UNIT.STR$ = XSTR$
48010 IF M.VALUE > 65535! THEN PRINT "VALUE OUT OF RANGE" : RETURN
48020 T1% = FIX(M.VALUE/256)
48030 XVAL% = M.VALUE-(T1%*256!) : GOSUB 45000 : VAL.STR$ = XSTR$
48040 XVAL% = T1% : GOSUB 45000
48050 CMD.BUF$ = UNIT.STR$+"0900"+VAL.STR$+XSTR$
48060 GOSUB 42000   'SEND ACCESS COMMAND
48070 RETURN
48080 '
48090 '
48100 '
48800 '******************************************************
48810 '* RESET RAMP/SOAK PROGRAM
48820 '*
48830 '* THIS ROUTINE WILL RESET THE CURRENT RAMP/SOAK PROGRAM
48840 '* ITS FUNCTION IS THE SAME AS THE FRONT PANEL KEY.
48850 '* THE CONTROLLER'S UNIT ADDRESS WILL BE IN UNIT.NUM%
48860 '*
48870 '******************************************************
48880 '
49000 XVAL%=UNIT.NUM%:GOSUB 45000:UNIT.STR$=XSTR$
49010 IF M.VALUE>65525! THEN PRINT "VALUE OUT OF RANGE": RETURN
49020 CMD.BUF$=UNIT.STR$+"050000000100"
49030 GOSUB 42000   'SEND RESET PROGRAM COMMAND
49040 RETURN
49050 '
49060 '
49070 '
49800 '******************************************************
49810 '* START RAMP/SOAK PROGRAM
49820 '*
49830 '* THIS ROUTINE WILL START THE CURRENT RAMP/SOAK PROGRAM
49840 '* ITS FUNCTION IS THE SAME AS THE FRONT PANEL KEY.
49850 '* THE CONTROLLER'S UNIT ADDRESS WILL BE IN UNIT.NUM%
49860 '*
49870 '******************************************************
49880 '
50000 XVAL% = UNIT.NUM% : GOSUB 45000 : UNIT.STR$ = XSTR$
50010 IF M.VALUE > 65535! THEN PRINT "VALUE OUT OF RANGE" : RETURN
50020 CMD.BUF$ = UNIT.STR$ + "050000000200"
50030 GOSUB 42000    'SEND START PROGRAM COMMAND
50040 RETURN
50050 '
50060 '
50070 '
50800 '******************************************************
50810 '* HOLD RAMP/SOAK PROGRAM
50820 '*
50830 '* THIS ROUTINE WILL PUT THE CURRENT RAMP/SOAK PROGRAM
50840 '* INTO THE HOLD MODE.
50850 '* ITS FUNCTION IS THE SAME AS THE FRONT PANEL KEY.
50860 '* THE CONTROLLER'S UNIT ADDRESS WILL BE IN UNIT.NUM%
```

```
50870 '*
50880 '******************************************************
50890 '
51000 XVAL% = UNIT.NUM% : GOSUB 45000 : UNIT.STR$ = XSTR$
51010 IF M.VALUE > 65535! THEN PRINT "VALUE OUT OF RANGE": RETURN
51020 CMD.BUF$ = UNIT.STR$ + "050000000400"
51030 GOSUB 42000    'SEND HOLD PROGRAM COMMAND
51040 RETURN
51050 '
51060 '
51070 '
51800 '******************************************************
51810 '* SET CURRENT PROGRAM NUMBER (model 2030 only)
51820 '*
51830 '* THIS ROUTINE WILL SET THE CURRENT PROGRAM NUMBER.
51840 '* THIS IS PAGE 18 MENU 2 IN THE 2030.  VALID ENTRIES
51850 '* ARE 0 TO 16 AND WILL BE IN PRG.NUM%.  THE CONTROLLER'S
51860 '* UNIT ADDRESS WILL BE IN UNIT.NUM%.
51870 '*
51880 '******************************************************
51890 '
52000 XVAL% = UNIT.NUM% : GOSUB 45000 : UNIT.STR$ = XSTR$
52010 IF (PRG.NUM%<0)OR(PRG.NUM%>16) THEN PRINT "VALUE OUT OF RANGE" : RETURN
52020 XVAL% = PRG.NUM% : GOSUB 45000
52030 CMD.BUF$ = UNIT.STR$+"08000212"+XSTR$+"00"
52040 GOSUB 42000    'SEND SET CURRENT PROGRAM COMMAND
52050 RETURN
52060 '
52070 '
52080 '
52800 '******************************************************
52810 '* READ PROCESS VARIABLE FOR CHANNEL 1 (model 2030 only)
52820 '*
52830 '* This routine will read the process variable for
52840 '* channel 1.  This is done by reading page 0, menu 1
52850 '* (the setpoint) and adding it to page 0, menu 3 (the
52860 '* deviation).  UNIT.NUM% will determine the controller
52870 '* address and the result will be returned in CH1PV.
52880 '*
52890 '******************************************************
52900 '
53000 MENU.NUM% = 3
53010 PG.NUM% = 0
53020 GOSUB 43000
53030 DEV = M.VALUE
53040 MENU.NUM% = 1
53050 GOSUB 43000
53060 CH1PV = DEV + M.VALUE
53070 RETURN
53080 '
53090 '
53100 '
53800 '******************************************************
53810 '* READ PROCESS VARIABLE FOR CHANNEL 2 (model 2030 only)
53820 '*
53830 '* This routine will read the process variable for
53840 '* channel 2.  This is done by reading page 0, menu 2
53850 '* (the setpoint) and adding it to page 0, menu 4 (the
53860 '* deviation).  UNIT.NUM% will determine the controller
53870 '* address and the result will be returned in CH2PV.
```

```
53880 '*
53890 '*****************************************************
53900 '
54000 MENU.NUM% = 4
54010 PG.NUM% = 0
54020 GOSUB 43000
54030 DEV = M.VALUE
54040 MENU.NUM% = 2
54050 GOSUB 43000
54060 CH2PV = DEV + M.VALUE
54070 RETURN
```

Appendix 3

Digital Communications PAGE/MENU Table

PAGE 3: CN3201 and CN3202 Controllers
PAGE 9: CN3220 Controller
PAGE 21: CN3230 Controller
PAGE 15: CN3390 Controller

DIGITAL COMMUNICATIONS / AUTOMATIC DATA LOGGING

CUE	MENU	SELECTION	AVAILABLE SETTINGS	FACTORY SETTING	SECURITY
Mode	1	Operation Mode	0 = Disabled 1 = Terminal Interface 2 = Automatic Data Logging 3 = Computer Interface 4 = Line Mode	0 = Disabled	D
Log Int	2	Automatic Logging Interval	1 to 9999 minutes	1 minute	
From M#	3	First MENU # to Display (from PAGE 0)	1 to 9 (CN3201) 1 to 10 (CN3202) 1 to 19 (CN3220) 1 to 11 (CN3230)	1	
To M#	4	Last MENU # to Display (from PAGE 0)	1 to 9 (CN3201) 1 to 10 (CN3202) 1 to 19 (CN3220) 1 to 11 (CN3230)	9 10 19 11	
Chr Lngth	5*	Character Length	Under 17, Over 16	Over 16	
HomeChar	5 (6 on CN3390)	Home Character Code	0 to 255	30	
Clr Char	6 (7 on CN3390)	Clear Screen Character Code	0 to 255	26	
Com Type	7 (8 on CN3390)	Interface Type	0 = RS232C 1 = RS422A 2 = RS485	RS232C	
BaudRate	8 (9 on CN3390	Baud Rate Select	0 = 300 1 = 600 2 = 1200 3 = 2400 4 = 4800 5 = 9600 6 = 19200 7 = 38,400 (CN3220, CN3230 only)	19.2K	
Parity	9 (10 on CN3390)	Parity Select	0 = No Parity 1 = Odd Parity 2 = Even Parity	0 = No Parity	
Reconfig	10* * (11 on CN3390)	Reconfigure Serial Port	Ready ▲ Reconfigure	Ready ▲	
Address	11 (12 on CN3390	Multidrop Controller Address	0 to 255	1	

* Model CN3390 only.

* * Note that any time a value is changed on the Digital Communications PAGE, a "1" must be entered at MENU 10 to reconfigure the set-up. After the operator exits the menu, the controller will respond by resetting the "1" back to "0", indicating that reconfiguration is complete.

MADE IN USA

WARRANTY

OMEGA warrants this unit to be free of defects in materials and workmanship and to give satisfactory service for a period of **13 months** from date of purchase. OMEGA Warranty adds an additional one (1) month grace period to the normal **one (1) year product warranty** to cover handling and shipping time. This ensures that our customers receive maximum coverage on each product. If the unit should malfunction, it must be returned to the factory for evaluation. Our Customer Service Department will issue an Authorized Return (AR) number immediately upon phone or written request. Upon examination by OMEGA, if the unit is found to be defective it will be repaired or replaced at no charge. However, this WARRANTY is VOID if the unit shows evidence of having been tampered with or shows evidence of being damaged as a result of excessive corrosion; or current, heat, moisture or vibration; improper specification; misapplication; misuse or other operating conditions outside of OMEGA's control. Components which wear or which are damaged by misuse are not warranted. These include contact points, fuses, and triacs.

We are glad to offer suggestions on the use of our various products. Nevertheless, OMEGA only warrants that the parts manufactured by it will be as specified and free of defects.

OMEGA MAKES NO OTHER WARRANTIES OR REPRESENTATIONS OF ANY KIND WHATSOEVER, EXPRESSED OR IMPLIED, EXCEPT THAT OF TITLE AND ALL IMPLIED WARRANTIES INCLUDING ANY WARRANTY OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE HEREBY DISCLAIMED.

LIMITATION OF LIABILITY: The remedies of buyer set forth herein are exclusive and the total liability of OMEGA with respect to this order, whether based on contract, warranty, negligence, indemnification, strict liability or otherwise, shall not exceed the purchase price of the component upon which liability is based. In no event shall OMEGA be liable for consequential, incidental or special damages.

Every precaution for accuracy has been taken in the preparation of this manual; however, OMEGA ENGINEERING, INC. neither assumes responsibility for any omissions or errors that may appear nor assumes liability for any damages that result from the use of the products in accordance with the information contained in the manual.

RETURN REQUESTS / INQUIRIES

Direct all warranty and repair requests/inquiries to the OMEGA ENGINEERING Customer Service Department. Call toll free in the USA and Canada: 1-800-622-2378, FAX: 203-359-7811; International: 203-359-1660, FAX: 203-359-7807.

BEFORE RETURNING ANY PRODUCT(S) TO OMEGA, YOU MUST OBTAIN AN AUTHORIZED RETURN (AR) NUMBER FROM OUR CUSTOMER SERVICE DEPARTMENT (IN ORDER TO AVOID PROCESSING DELAYS). The assigned AR number should then be marked on the outside of the return package and on any correspondence.

FOR **WARRANTY** RETURNS, please have the following information available BEFORE contacting OMEGA:

1. P.O. number under which the product was PURCHASED,
2. Model and serial number of the product under warranty, and
3. Repair instructions and/or specific problems you are having with the product.

FOR **NON-WARRANTY** REPAIRS OR **CALIBRATION**, consult OMEGA for current repair/calibration charges. Have the following information available BEFORE contacting OMEGA:

1. Your P.O. number to cover the COST of the of the repair/calibration,
2. Model and serial number of product, and
3. Repair instructions and/or specific problems you are having with the product.

OMEGA's policy is to make running changes, not model changes, whenever an improvement is possible. That way our customers get the latest in technology and engineering.

OMEGA is a registered trademark of OMEGA ENGINEERING, INC.

OMEGA... Your Source for Process Measurement and Control

TEMPERATURE

- ☑ Thermocouple, RTD & Thermistor Probes, Connectors, Panels & Assemblies
- ☑ Wire: Thermocouple, RTD & Thermistor
- ☑ Calibrators & Ice Point References
- ☑ Recorders, Controllers & Process Monitors
- ☑ Infrared Pyrometers

PRESSURE/STRAIN FORCE

- ☑ Transducers & Strain Gauges
- ☑ Load Cells & Pressure Gauges
- ☑ Displacement Transducers
- ☑ Instrumentation & Accessories

FLOW/LEVEL

- ☑ Rotameters, Gas Mass Flowmeters & Flow Computers
- ☑ Air Velocity Indicators
- ☑ Turbine/Paddlewheel Systems
- ☑ Totalizers & Batch Controllers

pH/CONDUCTIVITY

- ☑ pH Electrodes, Testers & Accessories
- ☑ Benchtop/Laboratory Meters
- ☑ Controllers, Calibrators, Simulators & Pumps
- ☑ Industrial pH & Conductivity Equipment

DATA ACQUISITION

- ☑ Data Acquisition and Engineering Software
- ☑ Communications-Based Acquisition Systems
- ☑ Plug-in Cards for Apple, IBM & Compatibles
- ☑ Datalogging Systems
- ☑ Recorders, Printers & Plotters

HEATERS

- ☑ Heating Cable
- ☑ Cartridge & Strip Heaters
- ☑ Immersion & Band Heaters
- ☑ Flexible Heaters
- ☑ Laboratory Heaters

ENVIRONMENTAL MONITORING AND CONTROL

- ☑ Metering & Control Instrumentation
- ☑ Refractometers
- ☑ Pumps & Tubing
- ☑ Air, Soil & Water Monitors
- ☑ Industrial Water & Wastewater Treatment
- ☑ pH, Conductivity & Dissolved Oxygen Instruments

Addendum Including Models CN3240, CN3251 and CN3101

This addendum includes:

- Security (Access) settings from the computer to the controller.
- Additional CN controllers not included in the manual
- Sample ASCII Line Mode Strings for all the CN units

Security (Access) setting from the computer to the controller:

In ASCII Line Mode and Computer Interface Mode, there are two security settings. This allows the system engineer to have separate security from the controller front panel and the computer. One is set in the page menu structure of the unit and is changed via a write command (04). This security setting only allows access to the settings when accessed at the front of the controller. It does not allow changes of settings from the computer.

In order to write commands via ASCII Line Mode or Computer Interface Mode the program must first send the ACCESS command (09). The 09 command uses the same security levels as the controller i.e. 123, 458, 736. After the ACCESS Command is sent the program can write to page/menus that can be accessed by the security level.

Additional Controllers not included in **the manual:**

In addition to the models in the manual, the following units also support the ASCII Line Mode and the CPIF (Computer Interface Mode) Protocols:

CN3251
CN3240
CN3101

These units do not support the Automatic Data Logging Mode and the Terminal Interface Mode. There are only three set-up digital communications parameters on these CN Controllers.

Description	**Settings**
Mode Selection	Off CPIF (computer Interface Mode) for use with CNSoft ASCII Line Mode
Baud Rate	1200, 2400, 4800, 9600, 19.2K
Address	1 to 255

The serial string is 8 data bits, no parity, one stop bit.

In order to program in ASCII Line Mode and CPIF numeric values have been assigned to the Pages, Menus, and the Values: The tables at the end of this Addendum detail the computer values and the controller displays.

Sample ASCII Line Mode Strings for all the CN units

The following is a sample set up for the controller and computer with sample strings for each of the CN controllers with ASCII Line Mode.

Computer and controller set-up

Connect your computer and controller using RS232, RS422 or RS485. Any communications software can work. In Windows 3.1 you can use Terminal, in Windows 95 you can use Hyper-Terminal or you can use Procomm or similar communications software. Set up your software and Controller at the same baud rate i.e. 19.2K. The ASCII Line Mode and Computer Interface Mode commands are sent with 8 bits, no parity. Set the controller to 'Line' not 'CPIF', and set up for address 1. Each command sent to the controllers will initiate a response from the controller. Below are given the commands and the appropriate response you should receive back.

You can manually type each command and hit 'ENTER'. The communications software should be set up to do a 'CR' AND 'LF' that way it will scroll down the screen and you will see the response from the controller. If you set up the software for Half-Duplex, you will also see what you are typing.

CN3201 ASCII LINE MODE COMMAND

Change access code	Response
010900E00214	014900B6

Write Security Code 736 to P1 M20	Response
0108001401E00200	014800B7

Write Set Point -100ø to P1 M1	Response
0108000101640091	014800B7

Read Set Point from P0 M1	Response
010100010002FB	0141006400000159

You should be able to see the Set Point on the Controller.

CN3202 ASCII LINE MODE COMMAND

Change access code	Response
010900E00214	014900B6

Write Security Code 736 to P1 M22	Response
0108001601E002FE	014800B7

Write Set Point -100¯ to P1 M1	Response
0108000101640091	014800B7

Read Set Point from P0 M1	Response
010100010002FB	0141006400000159

You should be able to see the Set Point on the Controller.

Sample ASCII Line Mode Strings for all the CN units *(continued)*

CN3221/CN3222 ASCII LINE MODE COMMAND

Change access code	Response
010900E00214	014900B6

Write Security Code 736 to P1 M1	Response
0108000101E00213	014800B7

Write Set Point -100ø to P2 M1	Response
0108000102640090	014800B7

Read Set Point from P0 M1	Response
010100010002FB	0141006400000159

You should be able to see the Set Point on the Controller.

CN3230 ASCII LINE MODE COMMAND

Change access code	Response
010900E00214	014900B6

Write Security Code 736 to P18 M1	Response
0108000112E00202	014800B7

Write Set Point -100ø to P19 M1	Response
010800011364007F	014800B7

Read Set Point from P0 M1	Response
010100010002FB	0141006400000159

You should be able to see the Set Point on the Controller.

CN3240 ASCII LINE MODE COMMAND

Change access code	Response
010900E00214	014900B6

Write Security Code 736 to P1 M1	Response
0108000101E00213	014800B7

Write Set Point -100ø to P1 M2	Response
0108000201640090	014800B7

Read Set Point from P0 M3	Response
010100030002F9	0141006400000159

You should be able to see the Set Point on the Controller.

Sample ASCII Line Mode Strings for all the CN units *(continued)*

CN3251 ASCII LINE MODE COMMAND

Change access code	Response
010900E00214	014900B6

Write Security Code 736 to P1 M1	Response
0108000101E00213	014800B7

Write Set Point -100ø to P1 M2	Response
0108000201640090	014800B7

Read Set Point from P0 M2	Response
010100020002FA	0141006400000159

You should be able to see the Set Point on the Controller.

CN3101 ASCII LINE MODE COMMAND

Change access code	Response
010900E00214	014900B6

Write Security Code 736 to P1 M1	Response
0108000101E00213	014800B7

Write Set Point -100ø to P1 M2	Response
0108000201640090	014800B7

Read Set Point from P0 M2	Response
010100020002FA	0141006400000159

You should be able to see the Set Point on the Controller.

CN3390 ASCII LINE MODE COMMAND

Change access code	Response
010900E00214	014900B6

Write Security Code 736 to P11 M1	Response
010800010BE00209	014800B7

Write Set Point -100ø to P1 M1	Response
0108000101640091	014800B7

Read Set Point from P0 M1	Response
010100010002FB	0141006400000159

You should be able to see the Set Point on the Controller.

CN3251 Page/Menu Tables

Display Page (page 0)

The Display Page is for status only. None of the settings can be changed.

Page #	MENU #	MENU	Description	Program Value #s*	Displayed Values			Security
0	1	Proc	Process Variable		Sensor Span			A
0	2	A SP	Active Setpoint		Sensor Span			
0	3	Out1	Output #1 Command		0.0 to 100.0%			
0	4	Out2	Output #2 Command		0.0 to 100.0%			
0	5	rSP	Remote Setpoint Input		Sensor Span			
0	6	rS	Ramp/Soak Status	0 =	OFF	=	Program not running	
				1 =	run	=	Program running	
				2 =	Hold	=	Program in hold	
				3 =	StbY	=	Program in standby	
				4 =	GS	=	Guaranteed soak	
0	7	int	Ramp/Soak Interval Number		0 - 16			
0	8	LEFt	Ramp/Soak Time Left in Interval		0.0 to 999.9 hr/min/sec			
0	9	LooP	Ramp/Soak Loops Remaining		0 - 9999			
0	10	Alr	Alarm Output Status	0 =	NonE	=	No alarms	
				1 =	A3	=	Alarm Output #3	
				2 =	A4	=	Alarm Output #4	
				3 =	A43	=	Alarm Outputs #4 and #3	
				4 =	A5	=	Alarm Output #5	
				5 =	A53	=	Alarm Outputs #5 and #3	
				6 =	A54	=	Alarm Outputs #5 and #4	
				7 =	A543	=	Alarms 5, 4 and 3	
0	11	Ent	Event Output Status	0 =	NonE	=	All off	
				1 =	E3	=	Event Output #3	
				2 =	E4	=	Event Output #4	
				3 =	E43	=	Event Outputs #4 and #3	
				4 =	E5	=	Event Output #5	
				5 =	E53	=	Event Outputs #5 and #3	
				6 =	E54	=	Event Outputs #5 and #4	
				7 =	E543	=	Events 5, 4 and 3	

Control Page (page 1)

Page #	MENU #	MENU	Description	Program Value #s*	Display Values	Factory Settings	Security
1	1	LocH	Security Lock		0 to 9999	458	A
1	2	SP	Setpoint		Instrument sensor span	Span Low	B
1	3	AuSP	Auxiliary SP		Instrument Sensor Span	Span Low	
1	4	tunE	Self Tune	0 =	OFF = Self tuning disabled	OFF	C
				1 =	PrUP = Powerup tuning		
				2 =	bEGn = Begin tuning		
1	5	Pb1	Proportional Band 1		0°F to sensor range	25°F	
1	6	Ar1	Automatic Reset 1		0.00 to 99.99 repeats/minute	0.10	
1	7	rAt1	Rate 1		0 to 500 seconds	10	
1	8	db1	Dead Band 1		1 to 100°F 0.01 to 6.25% span for analog inputs	5°F	
1	9	Pb2	Proportional Band 2		0°F to sensor range	25°F	

* If the Displayed Value is a number, the program value and display are the same.

CN3251 Page/Menu Tables *(cont.)*

Control Page (page 1) cont.

Page #	MENU #	MENU	Description	Program Value #s*	Display Values	Factory Settings	Security
1	10	Ar2	Automatic Reset 2		0.00 to 99.99 repeats/minute	0.10	D
1	11	rAt2	Rate 2		0 to 500 seconds	10	
1	12	db2	Dead Band 2		1 to 100°F 0.01 to 6.25% span for analog inputs	5°F	
1	13	OFSt	Manual Reset		-99.9 to 99.9	0.0	
1	14	FL	Fuzzy Logic	0 = 1 =	OFF = Disabled On = Enabled	On	
1	15	OrnG	Open Sensor Output Command		For Heat/Cool Control, adjustable: -100.0 to 100.0%: -100.0 to -0.1 for cooling 0.1 to 100.0 for heating	0.0%	
1	16	LooP	Control Loop Protection		OFF, 0.1 to 999.9 minutes	OFF	
1	17	Auto	Auto/Manual Disintegration Timer		0 to 100 seconds	10	
1	18	rrAt	Ramp Rate	0 =	OFF, 1-9999 degrees/hour	OFF	
1	19	Cont	Controller Type	0 = 1 = 1 =	HEAt = Reverse Acting Output Controller CooL = Direct Acting Single Output Controller HtCl = Heat/Cool Controller	HEAt	
1	20	CooL	Cooling Medium	0 = 1 = 2 = 3 =	Pid2 = Uses PID2 settings for cooling Air = Air Cooling Oil = Oil Cooling H2O = Water Cooling	Pid2	
1	21	rSP	Remote Setpoint Enable	0 = 1 =	OFF On	OFF	
1	22	Enti	Event/Digital Function	0 = 1 = 2 = 3 = 4 = 5 = 6 = 7 =	nonE = Disabled Pid2 = PID2 enable AuSP = Auxiliary SP enable rSP = Remote SP enable Outd = Output disable rS = Ramp/Soak Auto = Auto/Manual Air = Alarm Reset	nonE	
1	23	Au	Auxiliary Pushbutton Function	0 = 1 = 2 = 3 = 4 = 5 =	nonE = Disabled Pid2 = PID2 enable AuSP = Auxiliary SP enable rSP = Remote SP enable Outd = Output disable Auto = Auto/Manual	nonE	
1	24	Aout	Analog Output Assignment	0 = 1 = 2 = 3 = 4 =	nonE = Disabled Proc = Process Variable ASP = Active Setpoint Out1 = Control Output ! Out2 = Control Output 2	ASP	
1	25	rSEn	Ramp/Soak	0 = 1 =	OFF On	OFF	
1	26	CodE	User Selected Security Code		0-122 = Level A 123-457 = Level B 458-735 = Level C 736-999 = Level D	0	D

* If the Displayed Value is a number, the program value and display are the same.

CN3251 Page/Menu Tables *(cont.)*

Ramp/Soak Page (page 2)

Page #	MENU #	MENU	Description	Program Value #s*	Display Values	Factory Settings	Security
2	1	unit	Time Units	0 = 1 = 2 =	SEc = seconds (1 to 9999) Min = minutes (0.1 to 999.9) hr = hours (0.01 to 99.99)	SEc	C
2	2	StbY	Standby Setpoint		Instrument Sensor Span	Span Low	
2	3	int1	Interval 1 Time		see Time Units Menu (above)	0	
2	4	SP1	Setpoint 1 • Intervals 2-15 • Time and Setpoint		nstrument Sensor Span	Span Low	
2	33	in16	Interval 16 Time		see Time Units Menu (above)	0	
2	34	SP16	Setpoint 16		Instrument Sensor Span	Span Low	
2	35	Cont	Continuous Program	0 = 1 =	OFF On	OFF	
2	36	Fro	Loop from the end of interval		1 to 16	1	
2	37	to	To the beginning of interval		1 to 16	1	
2	38	no	Number of times		0 to 9999	0	
2	39	SbEt	Standby Events	0 = 1 = 2 = 3 = 4 = 5 = 6 = 7 =	OFF = All off E3 = Event Output 3 On E4 = Event Output 4 On E43 = Event Outputs 4 & 3 On E5 = Event Output 5 On E53 = Event Outputs 5 & 3 On E54 = Event Outputs 5 & 4 On E543 = Event Outputs 5, 4, 3 On	OFF	
2	40	i1E	Interval 1 Events	0 = 1 = 2 = 3 = 4 = 5 = 6 = 7 =	OFF = All off E3 = Event Output 3 On E4 = Event Output 4 On E43 = Event Outputs 4 & 3 On E5 = Event Output 5 On E53 = Event Outputs 5 & 3 On E54 = Event Outputs 5 & 4 On E543 = Event Outputs 5, 4, 3 On	OFF	
2	55	i16E	Interval 16 Events		same as Page 2 menu 40	OFF	
2	56	GSdb	Guaranteed Soak differential	0 =	OFF, 1°F to sensor range	0°F	

* If the Displayed Value is a number, the program value and display are the same.

CN3251 Page/Menu Tables *(cont.)*

Input Page (page 3)

Page #	MENU #	MENU	Description	Program Value #s*	Display Values	Factory Settings	Security
3	1	SEnS	Sensor Type	 0 = 1 = 2 = 3 = 4 = 5 = 6 = 7 = 8 = 9 = 10 = 11 =	Sensor Type selected here must agree with dip switch settings. J = J Thermocouple K = K Thermocouple T = T Thermocouple E = E Thermocouple R = R Thermocouple S = S Thermocouple B = B Thermocouple RTD = 100Ω Pt RTD (a = .00385) 4-20 = 4 to 20mA 0-5 = 0 to 5 Vdc 1-5 = 1 to 5 Vdc RTDT = 100Ω Pt RTD (0.1° resolution)	J	C
3	2	unit	Display Units	0 = 1 = 2 =	none = no units °F = Degrees Fahrenheit °C = Degrees Celsius	°F	
3	3	CoFF	Display/ Cal. Offset		-100°F to 100°F	0	
3	4	SPLL	Setpoint Low Limit	instrument Sensor Span		Span Low	
3	5	SPUL	Setpoint Upper Limit	Instrument Sensor Span		Span High	
3	6	CALS	Sensor Calibration	0 = 1 = 2 =	InLo InHi donE	InLo	D
3	7	CALr	Remote Setopint Calibration	0 = 1 = 3 =	InLo InHi donE	InLo	
3	8	AoO	Analog Output Zero Calibration		O to 4095		
3	9	AoS	Analog Output Span Calibration		0 to 4095		
3	10	rECc	Factory Calibration Recovery	0 = 1 = 2 =	rdY = Ready --- = Wait donE = Finished	CAUTION: Calibration has to be performed with a sensor simulator at the controller.	
3	11	FiLt	Digital Filter		0 to 60 seconds		
3	12	hPrc	High (max.) Process Input		Instrument Sensor Span		
3	13	LPrc	Low (min.) Process Input		Instrument Sensor Span		
3	14	hiA	High (max.) Ambient Temp.		Instrument Sensor Span		
3	15	LoA	Low (min.) Ambient Temp.		Instrument Sensor Span		

* If the Displayed Value is a number, the program value and display are the same.

CN3251 Page/Menu Tables *(cont.)*

■ *Custom Scaling Page (page 4)*

Page #	MENU #	MENU	Description	Program Value #s* and Display Values	Factory Settings	Security
4	1	dP	Analog Sensor Input Decimal Pts.	0 = none 1 = 123.4 2 = 12.34 3 = 1.234	1	C
4	2	AinL	Analog Sensor Input Low	-500 to 5000	0.0	
4	3	AinH	Analog Sensor Input High	-500 to 5000	100.0	
4	4	AotL	Analog Output Low	-500 to 5000	Span Low	
4	5	AotH	Analog Output High	-500 to 5000	Span High	
4	6	rSPL	Remote SP Input Low	-500 to 5000	Span Low	
4	7	rSPH	Remote SP Input High	-500 to 5000	Span High	

■ *Custom Scaling Page (page 5)*

Page #	MENU #	MENU	Description	Program Value #s* and Display Values	Factory Settings	Security
5	1	CYc1	Output #1 Cycle Time	0.0 to 60.0 seconds	1.0*	C
5	2	OL1	Output #1 Limit	0.0 to 100.0%	100.0%	
5	3	HoFF	Heat Offset	0°F to PB1 setting	0	

* For 2104-A (voltage or current output) cycle time must be set to 0.0.

■ *Output Page #2 (page 6)*

Page #	MENU #	MENU	Description	Program Value #s* and Display Values	Factory Settings	Security
6	1	CYc2	Output #2	0.0 to 60.0 seconds 0.0 = Voltage/Current algorithm	1.0	C
6	2	OL2	Output #2 Limit	0.0 to 100.0%	100.0%	
6	3	CoFF	Cool Offset	0°F to PB1 setting	0	

■ *Output #3 Page (page 7)*

Page #	MENU #	MENU	Description	Program Value #s*	Display Values	Factory Settings	Security
7	1	tYP3	Output #3 Type	0 = 1 = 2 =	OFF = Disabled ALr = Alarm Output Ent = Event Output (Setup Event Output parameters on Ramp/Soak Page)	OFF	C

* If the Displayed Value is a number, the program value and display are the same.

CN3251 Page/Menu Tables *(cont.)*

Output #3 Page (page 7) cont.

Page #	MENU #	MENU	Description	Program Value #s*	Display Values	Factory Settings	Security
7	2	Al r3	Alarm #3 Type	0 =	nonE = Disabled (off)	nonE	C
				1 =	Hi = High Alarm		
				2 =	Lo = Low Alarm		
				3 =	HiLo = High-Low Alarm		
				4 =	PdE = Plus Deviation Alarm		
				5 =	-dE = Minus Deviation Alarm		
				6 =	dE = Plus/Minus Deviation Alarm		
				7 =	LooP = Control Loop Protection Alarm		
7	3	rLY3	Alarm #3 Relay Action	0 =	ndE = Normally de-energized non-latching	ndE	
				1 =	nE = Normally energized non-latching		
				2 =	ndEL = Normally de-energized latching		
				3 =	nEL = Normally energized latching		
7	4	AoL3	Alarm #3 Low Setpoint		Instrument Sensor Span	Span Low	
7	5	Ahi3	Alarm #3 High Setpoint		Instrument Sensor Span	Span High	
7	6	db3	Output #3 Dead Band (Alarm Hysteresis)		0 to 100°F	1°F	
7	7	inh3	Alarm #3 Inhibit	0 =	OFF	OFF	
				1 =	On		

Output #4 Page (page 8)

Page #	MENU #	MENU	Description	Program Value #s*	Display Values	Factory Settings	Security
8	1	tYP4	Output #4 Type	0 =	OFF = Disabled	OFF	C
				1 =	Al r = Alarm Output		
				2 =	Ent = Event Output (Setup Event Output parameters on Ramp/Soak Page)		
2	Al r4	Alarm #4 Type		0 =	nonE = Disabled (off)	nonE	
				1 =	Hi = High Alarm		
				2 =	Lo = Low Alarm		
				3 =	HiLo = High-Low Alarm		
				4 =	PdE = Plus Deviation Alarm		
				5 =	-dE = Minus Deviation Alarm		
				6 =	dE = Plus/Minus Deviation Alarm		
				7 =	LooP = Control Loop Protection Alarm		
8	3	rLY4	Alarm #4 Relay Action	0 =	ndE = Normally de-energized non-latching	ndE	
				1 =	nE = Normally energized non-latching		
				2 =	ndEL = Normally de-energized latching		
				3 =	nEL = Normally energized latching		
8	4	AoL4	Alarm #4 Low Setpoint		Instrument Sensor Span	Span Low	
8	5	Ahi4	Alarm #4 High Setpoint		Instrument Sensor Span	Span High	
8	6	db4	Output #4 Dead Band (Alarm Hysteresis)		0 to 100°F	1°F	
8	7	inh4	Alarm #4 Inhibit	0 =	OFF	OFF	
				1	=On		

* If the Displayed Value is a number, the program value and display are the same.

CN3251 Page/Menu Tables *(cont.)*

■ *Output #5 Page (page 9)*

Page #	MENU #	MENU	Description	Program Value #s*	Display Values	Factory Settings	Security
9	1	t4P5	Output #5 Type	0 =	OFF = Disabled	OFF	C
				1 =	Al r = Alarm Output		
				2 =	Ent = Event Output (Setup Event Output parameters on Ramp/Soak Page)		
9	2	Al r5	Alarm #5 Type	0 =	nonE = Disabled (off)	nonE	
				1 =	Hi = High Alarm		
				2 =	Lo = Low Alarm		
				3 =	HiLo = High-Low Alarm		
				4 =	PdE = Plus Deviation Alarm		
				5 =	-dE = Minus Deviation Alarm		
				6 =	dE = Plus/Minus Deviation Alarm		
				7 =	LooP = Control Loop Protection Alarm		
9	3	rL45	Alarm #5 Relay Action	0 =	ndE = Normally de-energized non-latching	ndE	
				1 =	nE = Normally energized non-latching		
				2 =	ndEL = Normally de-energized latching		
				3 =	nEL = Normally energized latching		
9	4	Aol 5	Alarm #5 Low Setpoint		Instrument Sensor Span	Span Low	
9	5	Ahi5	Alarm #5 High Setpoint		Instrument Sensor Span	Span High	
9	6	db5	Output #5 Dead Band (Alarm Hysteresis)		0 to 100°F1°F 0.00 to 6.25% for analog input		
9	7	inh5	Alarm #5 Inhibit	0 =	OFF	OFF	
				1 =	On		

■ *Digital Communications Page 10 :* di9 PA9E

Page #	MENU #	MENU	Description	Program Value #s*	Display Values	Factory Settings	Security
10	1	di9t	Mode Selection	0 =	OFF = Disabled	CPiF	C
				1 =	CPiF = Computer Interface		
				2 =	LinE = ASCII Line		
10	2	bAud	Baud Rate	0 =	1200	19.2K	
				1 =	2400		
				2 =	4800		
				3 =	9600		
				4 =	19.2K		
10	3	Addr	Address		1 to 255	1	

* If the Displayed Value is a number, the program value and display are the same.

CN3240 Page/Menu Tables

Display Page (page 0)

The Display Page is for status only. None of the settings can be changed.

Page #	MENU #	Alpha Cue	Program Value #s*	Display Setting	Security
0	1	PU 1		Loop 1 Process Value	A
0	2	PU 2		Loop 2 Process Value	
0	3	SP 1		Loop 1 Setpoint	
0	4	dSP1		Loop 1 Deviation from setpoint	
0	5	Out1		Loop 1 Output % ON	
0	6	ALr1		Alarm 1 Status	
				0 = Off	
				1 = On	
0	7	SP 2		Loop 2 Setpoint	
0	8	dSP2		Loop 2 Deviation from setpoint	
0	9	Out2		Loop 2 Output % ON	
0	10	ALr2		Alarm 2 Status	
				0 = Off	
				1 = On	
0	11	Out3		Output 3 Status % ON	
0	12	ALr3		Alarm 3 Status	
				0 = Off	
				1 = On	
0	13	t5tA		Timer Status	
			0 =	diSA = Disabled	
			1 =	COFF = Control Off	
			2 =	rPA = Ramp to Run Setpoint	
			3 =	rPI = Ramp to Idle Setpoint	
			4 =	SoA = Soak at Run Setpoint	
			5 =	SoI = Soak at Idle Setpoint	
0	14	Actt		Time Remaining in Active Timer	
0	15	CJCt		Cold Junction Terminal Temp.	

All parameters on PAGE 0 are real time values and not adjustable.

General Control Operations: gnrL (page 1)

Page #	MENU #	Alpha Cue	Selection	Program Value #s*	Display Setting	Factory Setting	Security
Security Code							
1	1	LocK	Security Code/Lock	0 to 999	None		A
Idle/Run Setpoints							
1	2	SP 1	Loop 1 Run Setpoint		Instrument Sensor Range	75°F	B
1	3	SP 2	Loop 2 Run Setpoint		Instrument Sensor Range	75°F	
1	4	ISP1	Loop 1 Idle Setpoint		Instrument Sensor Range	75°F	
1	5	ISP2	Loop 2 Idle Setpoint		Instrument Sensor Range	75°F	
1	6	ALt1	Alarm 1 Type	0 =	Hi = High	High	C
				1 -	Lo = Low		
				2 -	PdE = + Deviation		
				3 -	-dE = - Deviation		
				4 =	dE = +/- Deviation		
1	7	rLy1	Alarm 1 Relay Action	0 =	ndE = Normally de-energized, non-latching	NDE	
				1 =	nE = Normally energized, non-latching		
				2 =	ndEL = Normally de-energized, latching		
				3 =	nEL = Normally energized, latching		

* If the Displayed Value is a number, the program value and display are the same.

CN3240 Page/Menu Tables *(cont.)*

General Control Operations: gnrL (page 1) cont.

Page #	MENU #	Alpha Cue	Selection	Program Value #s*	Display Setting	Factory Setting	Security
1	8	ASP1	Alarm 1 Setpoint or Deviation Setpoint		Loop 1 Sensor Range	Sensor Span	
1	9	inh1	Alarm 1 Inhibit at Power-Up	0 = 1 =	oFF = Off on = On	Off	
1	10	Alt2	Alarm 2 Type	0 = 1 = 2 = 3 = 4 =	Hi = High Lo = Low Pde = + Deviation -dE = - Deviation dE = +/- Deviation	High	
1	11	rLY2	Alarm 2 Relay Action	0 = 1 = 2 = 3 =	ndE = Normally de-energized, non-latching nE = Normally energized, non-latching ndEL = Normally de-energized, latching nEL = Normally energized, latching	NDE	
1	12	ASP2	Alarm 2 Setpoint or Deviation		Loop 2 Sensor Range	Sensor Span	
1	13	inh2	Alarm 2 Inhibit at Power Up		oFF = Off on = On	Off	
1	14	tFun	Timer / External Input Functions		0 = Disabled 1 = On delay timer 2 = Off delay timer 3 = Change setpoint on remote contact input 4 = Change setpoint on momentary switch or START/STOP pushbutton 5 = Ramp/Soak on remote contact input 6 = Ramp/Soak on momentary switch or START/STOP pushbutton	0	
1	15	drtr	Delay or Ramp to Run Timer		0.00 to 50.00 (hours.minutes) 0.0 to 300.0 (hours) 0 to 3000 (hours)	P1 M21 = 0 P1 M21 = 1 P1 M21 = 2	C
1	16	Srtr	Soak at Run Timer		0.00 to 99.59 (hours.minutes) 0.0 to 999.9 (hours) 0 to 9999 (hours)	P1 M21 = 0 P1 M21 = 1 P1 M21 = 2	
1	17	rItr	Ramp to Idle Timer		0.00 to 50.00 (hours.minutes) 0.0 to 300.0 (hours) 0 to 3000 (hours)	P1 M21 = 0 P1 M21 = 1 P1 M21 = 2	
1	18	SItr	Soak at Idle SP Timer		0.00 to 99.59 (hours.minutes) 0.0 to 999.9 (hours) 0 to 9999 (hours)	P1 M21 = 0 P1 M21 = 1 P1 M21 = 2	
1	19	gSdb	Guaranteed Soak Differential		0.00 to 99.99% of sensor span 0.00		
1	20	LdSP	Lower Display Selection	0 = 1 = 2 = 3 =	L2PV = Loop 2 Process Val L1SP = Loop 1 Setpoint L2SP = Loop 2 Setpoint L2AL = Loop 2 Alarm Setpoint	Loop 2 Process Val. L2PV	
1	21	tUnt	Timer Unit Selection		0 = Hours, Minutes 1 = 000.0 Hours 2 = 0000 Hours		

* If the Displayed Value is a number, the program value and display are the same.

CN3240 Page/Menu Tables *(cont.)*

Loop #1 (page 2)

Page #	MENU #	Alpha Cue	Selection	Program Value #s*	Display Setting	Factory Setting	Security
2	1	Func	Loop 1 Function/ Control Type	0 = 1 = 2 = 3 =	dISA = Disabled OnOf = On/Off Control Pid = PID Control AlAr = Alarm	PID	C
2	2	oFSt	Manual Reset (Offset)		-99.9 to 99.9%	0.0	
2	3	Pb	Proportional Band		0.1 to 999.9%	5.0	
2	4	Ar	Automatic Reset		0.00 to 99.99 repeats/min.	0.10	
2	5	rAtE	Rate		0 to 500 seconds	0	
2	6	outL	Output Limit		0 to 100%	100	
2	7	cYcL	Cycle Time		0.1 to 60.0 seconds	10.0	
2	8	dr dr	Control Action		d = direct (cooling) r = reverse (heating)	r = reverse (heating)	
2	9	db	Deadband		1 to 99°F	5°F	
2	10	SPUL	Setpoint Upper Limit		Sensor Span	Span Max Limits, P1/M2, 4	D
2	11	SPLL	Setpoint Lower Limit		Sensor Span	Span Min Limits, P1/M2, 4	
2	12	orco	Sensor Out-of-Range	0 = 1 = 2 = 3 =	dHL = Disabled High / Low dHi = Disabled High Enabled Low dLo = Disabled Low Enabled High EnHL = Enabled High / Low	EnHL = Enabled High/Low	
2	13	oAPL	Control Output for Out-of-Range		0 to 100%	0	
2	14	COFF	Calibration Offset		Sensor Span	0	
2	15	tunE	Self-Tuning	0 = 1 = 2 = 3 =	OFF = Manual (none) Std = Standard SLo = Slow down fast responding process FASt = Speed up slow responding process		

Auxiliary Alarm & Output #3 : OutP (page 4)

Page #	MENU #	Alpha Cue	Selection	Program Value #s*	Display Setting	Factory Setting	Security
3	1	ALSP	Alarm 3 Setpoint		Instrument Sensor Span	Sensor Span	C
3	2	ALSc	Alarm 3 Sensor Input	0 = 1 = 2 =	SnS1 = Loop 1 Sensor SnS2 = Loop 2 Sensor Sn12 = Both Sensors	Sn12= Both	
3	3	ALtP	Alarm 3 Type	0 = 1 =	Hi = High Alarm Lo = Low Alarm	Hi = High	
3	4	inhb	Alarm 3 Power-Up Inhibit	0 = 1 =	Off On	Off	
Output #3 Function							
3	5	Ot3F	Output #3 Function		0 = Disabled 1 = Auxiliary Alarm 3 only 2 = Common Alarm for Alarm 1 & Auxiliary Alarm 3 3 = Common Alarm for Alarm 2 & Auxiliary Alarm 3 4 = Alarm 1 (only) 5 = Alarm 2 (only) 6 = Common Alarm for Alarms 1 & 2 7 = Loop 2 On/Off Control 8 = Loop 2 PID Control 9 = Process Enable 10 = Common Event for Alarms 1 and 2	0 = Disabled	
3	6	Ct3r	Output 3 Relay	0 = 1 = 2 = 3 =	ndE = Normally De-energized (NDE), non-latching nE = Normally Energized (NE), non-latching ndEL = NDE Latching nEL = NE Latching	ndE	

* If the Displayed Value is a number, the program value and display are the same.

CN3240 Page/Menu Tables *(cont.)*

Sensor Setup and Calibration : CAL (page 5)

Page #	MENU #	Alpha Cue	Selection	Program Value #s*	Display Setting	Factory Setting	Security
5	1	Unit	Sensor Units	0 = 1 =	°F °C	°F °C	C
5	2	SEn1	Input 1 Sensor Type	0 = 1 = 2 =	J TC = Type J TC H TC = Type K TC 385 = .00385 RTD	J T/C	
5	3	SEn2	Input 2 Sensor Type	0 = 1 = 2 =	J TC = Type J TC H TC = Type K TC 385 = .00385 RTD	J T/C	
5	4	rECc	Factory Recovery		0 = Off 1 = Recover calibration	0	D
5	5	Cnct	Cold Junction temp. at Cal.		0.0 to 150.0 °F	75.0	
5	6	CJCc	CJC Calibration Command	0 = 1 =	rdy = Ready donE = finished	rdy	
5	7	CALt	Sensor Cal. Type	0 = 1 =	SiLt = Simulator CuSr = Copper wire, millivolt source	SiLt	
5	8	CAL1	Input 1 Sensor Cal. Command	0 = 1 = 2 =	inLo = Ready for range minimum inHi = Ready for range maximum donE = finished	inLo*	
5	9	CAL2	Input 2 Sensor Cal. Command	0 = 1 = 2 =	inLo = Ready for range minimum inHi = Ready for range maximum donE = finished	inLo	

*CAUTION: Calibration has to be performed with a sensor simulator at the controller.

Digital Communictions (page 6)

Page #	MENU #	Alpha Cue	Selection	Program Value #s*	Display Setting	Factory Setting	Security
6	1	digt	Mode Selection	0 = 1 = 2 =	oFF = Disabled (none) CPiF = Computer Interface LinE = ASCII Line Mode	oFF	C
6	2	bAud	Baud Rate	0 = 1 = 2 = 3 = 4 =	1200 2400 4800 9600 19.2 (K)	19.2 (K)	
6	3	Addr	Address		0 to 255	1	

Manual Calibration (page 7)

Page #	MENU #	Alpha Cue	Selection	Display Setting	Security
7	1	CIJ0	Loop 1, J T/C Zero	See Sensor Input - Manual Calibration Instructions pp. 43 & 44.	D
7	2	CIJ5	Loop 1, J T/C Span		
7	3	C1K0	Loop 1, K T/C Zero	0 to FFFF	
7	4	C1K5	Loop 1, K T/C Span		
7	5	C1r0	Loop 1, RTD Zero		
7	6	C1r5	Loop 1, RTD Span		
7	7	C1c0	Loop 1, CJC Zero		
7	8	C1c5	Loop 1, CJC Span		
7	9	C2J0	Loop 2, J T/C Zero		
7	10	C2J5	Loop 2, J T/C Span		
7	11	C2K0	Loop 2, K T/C Zero		
7	12	C2K5	Loop 2, K T/C Span		
7	13	C2r0	Loop 2, RTD Zero		
7	14	C2r5	Loop 2, RTD Span		
7	15	C2c0	Loop 2, CJC Zero		
7	16	C2c5	Loop 2, CJC Span		

* If the Displayed Value is a number, the program value and display are the same.

CN3101 Page/Menu Tables

Display Page (page 0)

The Display Page is for status only. None of the settings can be changed.

Page #	MENU #	MENU	Description	Program Value #s*	Displays	Security
0	1	Proc	Process Variable		Sensor Span	A
0	2	L SP	Limit Setpoint		Sensor Span	
0	2	Lout	Limit Output		0.0 TO 100.0%	
0	2	toSP	Time Over Setpoint		0 to 999.9 Min	
0	2	PEAH	Peak Temperature		Instrument Sensor Span	
0	2	ALr	Alarm Output Status	0 = 1 = 2 = 3 =	nonE = No alarms AL1 = Alarm #1 AL2 = Alarm #2 AL12 = Alarm #1 and #2	

Setup Page (page 1)

Page #	MENU #	MENU	Description	Program Value #s*	Display Settings	Factory Setting	Security
1	1	LocH	Security Lock		0 to 9999	458	A
1	2	L SP	Limit Setpoint		Instrument Sensor Span	Span High	C
1	3	db	Limit Dead Band		0 to 100	1°F	
1	4	Enti	Event Input Function	0 = 1 =	nonE = Disabled ArSt = Alarm Reset	Ar-St	D
1	5	AOut	Analog Output Enable	0 = 1 =	nonE = Disabled Proc = Process Variable	nonE	
1	6	Cont	Controller Type	0 = 1 =	Hi Lo	Hi	
1	7	CodE	User Security Code		0 to 999 0-122 = Security level A 123-457 = Security level B 458-735 = Security level C 736-999 = Security level D	0	
1	8	ALo	Ambient Temp Low		-3 to 153	55	
1	9	AHi	Ambient Temp High		-3 to 153	85	
1	10	LdSP	Lower Display Enable	0 = 1 =	ON = Enabled OFF = Disabled	On	

Input Page (page 2)

Page #	MENU #	MENU	Description	Program Value #s*	Display Settings	Factory Setting	Security
2	1	SEnS	Sensor Type	 0 = 1 = 2 = 3 = 4 = 5 = 6 = 7 = 8 = 9 = 10 =	Sensor Type selected here must agree with dip switch settings. J = J Thermocouple K = K Thermocouple T = T Thermocouple E = E Thermocouple R = R Thermocouple S = S Thermocouple B = B Thermocouple rtd= 100Ω Pt RTD (α = .00385) 4-20 =4 to 20mA 0-5= 0 to 5 Vdc 1-5 = 1 to 5 Vdc	J	C

* If the Displayed Value is a number, the program value and display are the same.

CN3101 Page/Menu Tables

Input Page (page 2) cont.

Page #	MENU #	MENU	Description	Program Value #s*	Display Settings	Factory Setting	Security
2	1	unit	Display Units	0 = 1 = 2 =	nonE = none (analog inputs) °F = Degrees Fahrenheit °C = Degrees Celsius	°F	
2	2	CoFF	Calibration Offset		0 to ±100°F (±6.25% of span for analog inputs)	0	
2	3	SPLL	Setpoint Low Limit		Instrument Sensor Span	Span Low	
2	4	SPUL	Setpoint Upper Limit		Instrument Sensor Span	Span High	
2	5	CALS*	Sensor Calibration	0 = 1 = 2 =	inLo = Input low inHi = Input high donE = Calibration finished	inLo	D
2	6	Ao 0	Analog Output Zero Calibration		0 to 4095	81	
2	7	Ao S	Analog Output Span		0 to 4095	3902	
2	8	rECc	Factory Calibration Recovery	0 = 1 = 2 =	rdY = Ready ---- = Wait donE = Finished	rdY	

* CAUTION: Calibration as to be performed with a sensor simulator at the controller.

Custom Scaling Page (page 3)

Page #	MENU #	MENU	Description	Program Value Display	Factory	Security Setting
3	1	dP	Analog Input Decimal Pts.	0 = none 1 = 123.4 2 = 12.34 3 = 1.234	1	C
3	2	AinL	Analog Process Input Low	-500 to 5000	0.0	
3	3	AinH	Analog Process Input High	-500 to 5000	100.0	
3	4	AotL	Analog Process Output Low	Instrument Sensor Span	Span Low	
3	5	AotH	Analog Process Output High	Instrument Sensor Span	Span High	

Alarm #1 Page (page 4)

Page #	MENU #	MENU	Description	Program Value #s*	Display Settings	Factory Setting	Security
4	1	En1	Alarm 1 Enable	0 = 1 =	OFF = Disabled On = Enabled	OFF	C
4	2	tYP1	Alarm 1 Type	0 = 1 = 2 = 3 = 4 = 5 = 6 =	nonE = Disabled (off) Hi = High Alarm Lo = Low Alarm HiLo = High-Low Alarm PdE = Plus Deviation Alarm -dE = Minus Deviation Alm dE = Plus/Minus Dev Alm	NONE	
4	3	rlY1	Alarm 1 Relay	0 = 1 = 2 = 3 =	ndE = normally de-energized non-latching nE = normally energized non-latching ndEL = normally de-energized latching nEL = normally energized latching	ndE	

* If the Displayed Value is a number, the program value and display are the same.

CN3101 Page/Menu Tables

Alarm #1 Page (page 4) cont.

Page #	MENU #	MENU	Description	Program Value #s*	Display Settings	Factory Setting	Security
4	4	ALo1	Alarm 1 Low Setpoint		Instrument Sensor Span	Span Low	C
4	5	AHi1	Alarm 1 High Setpoint		Instrument Sensor Span	Span High	
4	6	db1	Output 1 Dead Band (Alarm Hysteresis)		0 to 100°F (.00 to 6.25% of span for analog inputs)	1°F	
4	7	inh1	Alarm 1 Inhibit	0 = 1 =	OFF On	OFF	

Alarm #2 Page (page 5) cont.

Page #	MENU #	MENU	Description	Program Value #s*	Display Settings	Factory Setting	Security
5	1	En2	Alarm 2 Enable	0 = 1 =	OFF = Disabled ON = Enabled	OFF	C
5	2	tYP2	Alarm 2 Type	0 = 1 = 2 = 3 = 4 = 5 = 6 =	nonE = Disabled (off) Hi = High Alm Lo = Low Alm HiLo = High-Low Alm PdE = Plus Deviation Alm -dE = Minus Deviation Alm dE = Plus/Minus Dev Alm	nonE	
5	3	rLY2	Alarm 2 Relay	0 = 1 = 2 = 3 =	ndE = normally de-energized non-latching nE = normally energized non-latching ndEL = normally de-energized latching nEL = normally energized latching	ndE	
5	4	ALo2	Alarm 2 Low Setpoint		Instrument Sensor Span	Span Low	
5	5	AHi2	2 High Setpoint		Instrument Sensor Span	Span High	
5	6	db2	Output 2 Dead Band (Alarm Hysteresis)		0 to 100°F (.00 to 6.25% of span for analog inputs)	1°F	
5	7	inh2	Alarm 2 Inhibit	0 = 1 =	OFF On	OFF	

Digital Communications Page: diGPAGE (page 6)

Page #	MENU #	MENU	Description	Program Settings	Available Settings	Factory Settings	Security
6	1	diGt	Mode Selection	0 = 1 = 2 =	OFF = Disabled CPiF = Computer Interface LinE = ASCII Line	CP.F	C
6	2	bAud	Baud Rate	0 = 1 = 2 = 3 = 4=	1200 2400 4800 9600 19.2K	19.2K	
6	3	Addr	Address		1 to 255	1	

* If the Displayed Value is a number, the program value and display are the same.