

® CN76000

® Protocol for RS-485 ® Communications Option (-485)

COMMAND = 0100
COMMAND = 0102
COMMAND = 0104
COMMAND = 0105
COMMAND = 0124
COMMAND = 0116
COMMAND = 0117
COMMAND = 0110
COMMAND = 0111
COMMAND = 011A
COMMAND = 011B
COMMAND = 0121

DATA =
1st CHAR
2nd CHAR

3rd CHAR
4th CHAR
5th CHAR
6th CHAR

COMMAND = 0107
COMMAND = 0108
COMMAND = 0112
COMMAND = 0113
COMMAND = 010A
COMMAND = 010B
COMMAND = 0114
COMMAND = 0115
COMMAND = 010C
COMMAND = 010D
COMMAND = 010E
COMMAND = 010F
COMMAND = 0125
COMMAND = 0118
COMMAND = 0129
COMMAND = 0128
COMMAND = 011E
COMMAND = 012A
COMMAND = 0126
COMMAND = 0127

READ "SP1" VALUE
READ "SP2" VALUE
READ "ALLO" VALUE
READ "ALH1" VALUE
READ "INPC" INPUT CORRECTION VALUE
READ "SCAL" VALUE
READ "SCAH" VALUE
READ "SPL" VALUE
READ "SPR" VALUE
READ "PEA" PEAK VALUE
READ "VAL" VALLEY VALUE
READ "CFSP" COMM FAULT SV VALUE

SIGN 0 = POSITIVE.
SIGN 0 = POSITIVE.

(NOT 1) = NEGATIVE
(NOT 1) = NEGATIVE

MSD VALUE
VALUE
VALUE
LSD VALUE

BOTH 1st & 2nd CHARS
MUST BE "0" FOR SIGN
TO BE POSITIVE.

READ "SP1d" VALUE
READ "PUL1" VALUE
READ "S1OL" VALUE
READ "S1OH" VALUE
READ "SP2d" VALUE
READ "PUL2" VALUE
READ "S2OL" VALUE
READ "S2OH" VALUE
READ "Pb1" VALUE
READ "Pb2" VALUE
READ "RES" RESET VALUE
READ "RTE" RATE VALUE
READ "ARTE" VALUE
READ "IDPT" INPUT FAULT TIMER VALUE
READ "SENO" VALUE
READ "LPBT" VALUE
READ SP1'S MANUAL VALUE
READ SP2'S MANUAL VALUE
READ "1rt" RAMP TIME
READ "1st" SOAK TIME



Servicing USA and Canada: Call OMEGA Toll Free

USA

One Omega Drive, Box 4047
Stamford, CT 06907-0047
Telephone: (203) 359-1660
FAX: (203) 359-7700

Canada

976 Bergar
Laval (Quebec) H7L 5A1
Telephone: (514) 856-6928
FAX: (514) 856-6886

Sales Service: 1-800-826-6342 / 1-800-TC-OMEGASM
Customer Service: 1-800-622-2378 / 1-800-622-BESTSM
Engineering Service: 1-800-872-9436 / 1-800-USA-WHENSM
TELEX: 996404 EASYLINK: 62968934 CABLE OMEGA

Servicing Europe: United Kingdom Sales and Distribution Center

25 Swannington Road, Broughton Astley, Leicestershire
LE9 6TU, England
Telephone: 44 (0455) 285520 FAX: 44 (0455) 283912

The OMEGA Complete Measurement and Control Handbooks & Encyclopedias

- ✓ Temperature
- ✓ Pressure, Strain & Force
- ✓ Flow and Level
- ✓ pH and Conductivity
- ✓ Data Acquisition Systems
- ✓ Electric Heaters
- ✓ Environmental Monitoring and Control



Call for Your FREE Handbook Request Form Today: (203) 359-RUSH

**CN76000 PROTOCOL FOR RS-485 COMMUNICATIONS OPTION (-485)
TABLE OF CONTENTS**

SECTION 1 - COMMAND INTERACTION	1
Command and Response Example Notations	2
Field Position and Description of Data Frame.....	3
General Notes	5
Error Messages Reported	5
SECTION 2 - READ AND WRITE COMMANDS	6
Read and Write Command Summary	6
Read Command Description	8
Write Command Description	14
SECTION 3 - OPTION 4SP 4 STAGE SETPOINT	16
Option 4SP Read and Write Command Summary	16
Option 4SP Read Command Description	17
Option 4SP Write Command Description	20
SECTION 4 - SAMPLE PROGRAMS.....	21
Program using the OMEGA A1000 RS-232/RS-485 Converter.....	21
Program to Communicate with an OMEGA CN76000 Series Controller	22

SECTION 1 - COMMAND INTRODUCTION

The following 7 items define the conventions used in this Protocol manual

1. In the following examples, we use the terms **Host** and **Instrument**. **Host** is used to describe the computer operating as the originator of communications. **Instrument** is used to describe the process control(s) using this protocol.
2. All data is sent and received in the ASCII HEX character format using 10 bits:
 - 1 start bit
 - 8 data bits, no parity
 - 1 stop bit.
3. The **L** (ASCII 4C hex) is used as a filter character in the examples. The filter character prevents false addressing of other manufacturer's device on the same system.
Address 00h should not be used. This is for Factory Service use only.
4. The checksum generated by the **Host** is obtained by adding all ASCII characters in hex, excluding the <stx>, filter character, and the <etx>. Only the lowest 8 bits are used. The checksum is inserted into the message between the data field and the <etx>.

EXAMPLE: Host checksum calculation

		32h	0100h	26h		
<stx>	L	<addr>	<— data —>	<cksm>	<etx>	
02	4C	33	32 30 31 30 30	37	39	03
		<— cksm —>				
		33+32+30+31+30+30				
		= 126h (use only the low byte)				

-
5. The checksum generated by the **Instrument** is obtained by adding all ASCII characters in hex, excluding the <stx> and the <ack>. Only the lowest 8 bits are used. The checksum is inserted between the data field and the <ack>. Note that the filter character is included in the checksum calculation by **Instrument**.

EXAMPLE: Instrument checksum calculation

		32h	010015h	D8h		
<stx>	L	<addr>	<— data —>	<cksm>	<ack>	
02	4C	33	32 30 31 30 30 31 35	44	38	06
		<— cksm —>				
		4C+33+32+30+31+30+30+31+35				
		= 1D8h (use only the low byte)				

-
6. Checksums are generated for all **Host** commands
 7. Checksums are generated for all normal **Instrument** responses. A checksum is not sent when errors are reported to the **Host**.

COMMAND AND RESPONSE EXAMPLE NOTATIONS:

1. **[nu]** : "not used" at present time, (for future expansion). Undefined data, data returned should not be used.
2. **<data>** : These are ASCII characters shown in Hex notation. The upper case alpha character set is used for values A through F. (ASCII 46h = F, ASCII 33h = 3, etc.) Lower case is not used.
3.

ASCII control character	ASCII in hex	Description
<stx>	02	Start of Text
<etx>	03	End of Text
<ack>	06	Acknowledgment
L	4C	Filter character
4. Data can be in either Hex, BCD (Binary coded decimal), or Binary, expressed in ASCII format.

ASCII (in hex)	Hex (decimal)	BCD	Binary
30	0 h	0 d	0 0 0 0 b
31	1 h	1 d	0 0 0 1 b
32	2 h	2 d	0 0 1 0 b
33	3 h	3 d	0 0 1 1 b
34	4 h	4 d	0 1 0 0 b
35	5 h	5 d	0 1 0 1 b
36	6 h	6 d	0 1 1 0 b
37	7 h	7 d	0 1 1 1 b
38	8 h	8 d	1 0 0 0 b
39	9 h	9 d	1 0 0 1 b
41	A h		1 0 1 0 b
42	B h		1 0 1 1 b
43	C h		1 1 0 0 b
44	D h		1 1 0 1 b
45	E h		1 1 1 0 b
46	F h		1 1 1 1 b

All numbers shown in examples are in hex, unless otherwise noted. Hex = 2Fh
BCD (decimal) = 10d, Binary = 1101b. The h, d, b are symbols used in examples to indicate which number base is being used.

FIELD POSITION AND DESCRIPTION OF DATA FRAME

EXAMPLE: Typical READ command sequence.

Host wants to READ SP1 value from **Instrument** at address 32

Host command =

<stx>	L	<addr> 32h	<- data -> 0100h command	<cksm> 26h	<etx>
-------	---	---------------	--------------------------------	---------------	-------

Host command transmitted in ASCII hex =

02	4C	33 32	30 31 30 30	32 36	03
----	----	-------	-------------	-------	----

Instrument at address 32 replies with SP1 value of -15.

Instrument reply =

<stx>	L	<addr> 32h	<- data -> 01h sign 0015h value	<cksm> D8h	<ack>
-------	---	---------------	---------------------------------------------	---------------	-------

Instrument reply transmitted in ASCII hex =

02	4C	33 32	30 31	30 30 31 35	44 38	06
----	----	-------	-------	-------------	-------	----

1. **Host** starts command with start of text <stx>, and filter character.
2. The address of the **Instrument** desired is next.
3. The Read data field is next. It contains the basic command for the **Instrument**. The Read data field is four characters long.
4. The **Host** now adds the calculated checksum to the command frame. Finally the end of text <etx> is appended.
5. The **Instrument** responds when the <etx> has been received. If the data frame is correct, the **Instrument** will perform the received command. If the data field contains a bad command or bad check-sum, an error message will be sent back to the **Host**. Details on errors will be explained later.
6. The **Instrument** starts its response with the start of text <stx> character, followed by the filter character.
7. The **Instrument's** address is next.
8. The Read data field is next. It contains the response to the command received. This field may be up to 10 characters long.
9. The **Instrument** now adds the calculated checksum to the response frame. Finally the acknowledge <ack> is appended.
10. The Command and Response interchange is now complete.

FIELD POSITION AND DESCRIPTION OF DATA FRAME (CONT'D)

EXAMPLE: Typical WRITE command sequence

Host wants to WRITE -15 to SP1 on **Instrument** at address 32

Host command =

<stx>	L	<addr>	<data>		<cksm>	<etx>
		32h	0200h	0015h FFh	79h	
			command	value sign		

Host command transmitted in ASCII hex =

02 4C 33 32 30 32 30 30 30 30 31 35 46 46 37 39 03

Instrument at address 32 replies that SP1 has been accepted

Instrument reply =

<stx>	L	<addr>	<data>	<cksm>	<ack>
		32h	00h	11h	

Instrument reply transmitted in ASCII hex =

02 4C 33 32 30 30 31 31 06

1. **Host** starts command with start of text <stx> and filter character.
2. The address of the **Instrument** desired is next.
3. The Write data field is next. It contains the basic command for the **Instrument** along with any data that may be required. The Write data field can be up to 10 characters long. Write commands are used to modify the **Instrument's** parameters.
4. The **Host** now adds the calculated checksum to the command frame. Finally the end of text <etx> is appended.
5. The **Instrument** responds when the <etx> has been received. If the Data frame is correct, the **Instrument** will perform the received command. If the data frame contains a bad command or a bad checksum, an error message will be sent back to the **Host**. An error will also occur if the value is not within the **Instrument's** range. Details on errors will be explained later.
6. The **Instrument** starts its response with the start of text <stx> character, followed by the filter character.
7. The **Instrument's** address is next.
8. The Data field is next. It contains the response to the command received.
9. The **Instrument** now adds the calculated checksum to the response frame. Finally the acknowledge <ack> is appended.
10. Command and Response interchange is now complete.

GENERAL NOTES:

- Some commands have restricted access, such that they may not be entered if the menu item is not currently available. An example would be an attempt to read the P.I.D. values while the menu is set for the SELF TUNE mode of operation. An error will be reported to the **Host** if such a command is given, or if the command cannot be carried out by the **Instrument**.
- Read commands will not change the **Instrument** display.
- Some Write commands will cause the Instrument display to show the menu item to be entered, as though the "ENTER" key had been pressed.

ERROR MESSAGES REPORTED

EXAMPLE: Checksum error detected by Instrument at address 32

Instrument responds to **Host's** last command with Error message

<stx>	L	<addr>	N	<error code>	<ack>
		32h	N	02h	
02	4C	33 32	4E	30 32	06

(Note the absence of Checksum.)

N = ASCII character (4Eh) used to note Error Present

Table 1 - Table of Error Messages possible from Instrument

<error code>	Description
00	not used
01	Undefined command. Command not within acceptable range.
02	Check-sum error on received data from Host .
03	Command not performed by Instrument (option not enabled, restricted read/write menu, check message.)
04	Illegal ASCII characters received in command. Instrument accepts only ASCII characters 0 through 9, A through F, and a through f in the data field.
05	Data field error. Not enough, too many, or improper positioning of characters in data field.
06	Undefined command. Command not within acceptable range.
07	not used
08, 09	Hardware fault. Return to Factory for service
10	Undefined command. Command not within acceptable range.

SECTION 2 - READ AND WRITE COMMANDS

READ AND WRITE COMMAND SUMMARY

READ COMMANDS

00 PROCESS VARIABLE VALUE W/STATUS
 05 FULL STATUS

01 00 SP1 VALUE
 01 06 CY1 VALUE & S1Ot
 01 07 SP1d VALUE
 01 08 PUL1 VALUE
 03 13 S1St dir/rE
 01 12 S1OL VALUE
 01 13 S1OH VALUE
 03 14 S1LP 0 on/OoFF

03 35 SP1o Outb/OutA

03 36 S2t Abs/dE
 01 02 SP2 VALUE
 01 09 CY2 VALUE & S2Ot
 01 0A SP2d VALUE
 01 0B PUL2 VALUE
 03 15 S2St dir/rE
 01 14 S2OL VALUE
 01 15 S2OH VALUE
 03 16 S2LP 0 on/OoFF

03 39 tune SELF/Pid/SLO/nor/FAST

03 12 Strt YES/no
 03 38 LErn Cont/End
 03 2D dFAC VALUE
 01 0C Pb1 VALUE
 01 0D Pb2 VALUE
 03 2C rES RESET AUTO/OFS (MANUAL) MODE
 01 0E rES RESET VALUE

01 0F rtE RATE VALUE
 03 2E Pid2 On/OFF
 03 2F ArUP On/OFF
 01 25 Arte VALUE

01 04 ALLo VALUE
 01 05 ALHi VALUE
 03 37 AL OFF/Lo/Hi/HiLo
 03 17 ALt Abs/dE
 03 1B ALrE OnOF/Hold
 03 1C ALPi On/OFF
 03 3A ALiH On/OFF
 03 18 ALSt OPEn/CLOS
 03 19 ALLP 0 on/OoFF
 03 22 ALbr On/OFF

WRITE COMMANDS

02 00 SP1 VALUE XX
 02 06 CY1 VALUE XX

02 02 SP2 VALUE XX
 02 07 CY2 VALUE XX

04 03 tune MODE = SELF
 04 04 tune MODE = FULL Pid

02 08 Pb1 VALUE
 02 09 Pb2 VALUE

02 0A RESET VALUE AUTO MODE
 02 0B RESET VALUE OFFSET MODE
 02 0C RATE VALUE

02 04 ALLo VALUE XX
 02 05 ALHi VALUE XX

04 02 ALARM ACK

00 = Positive Value
 FF = Negative Value

00 = Positive Value
 FF = Negative Value

00 = Positive Value
 FF = Negative Value

READ AND WRITE COMMAND SUMMARY (CONT'D)

READ COMMANDS

03 34 SECr 1-4 SECURITY
 03 23 Inp INPUT TYPE
 03 25 OSUP On/OFF (FOR CURR & VOLT)
 03 10 Unit F/C (FOR T/C & RTD)
 03 26 Unit F/C/none
 03 24 dPt DECIMAL POINT POSITION
 01 18 InPt INPUT FAULT TIMER VALUE
 01 29 SEnC VALUE
 03 33 FiLt VALUE
 01 24 InPC INPUT CORRECTION VALUE
 01 28 LPbr VALUE
 01 16 SCAL VALUE
 01 17 SCAH VALUE
 01 10 SPL VALUE
 01 11 SPH VALUE
 03 28 Auto On/OFF (OFF = MANUAL MODE)
 01 1E AUTO/MAN SP1'S MANUAL VALUE
 01 2A AUTO/MAN SP2'S MANUAL VALUE
 01 1A PEA PEAK VALUE
 01 1B VAL VALLEY VALUE
 01 1D SP1 & SP2 PERCENT OUTPUT VALUE
 03 27 PctO On/OFF PERCENT OUTPUT INDICATION
 03 30 Prog On/OFF
 03 31 StAt On/OFF
 01 26 1rt RAMP TIME
 01 27 1St SOAK TIME
 03 32 PEnd OoFF/Hold
 01 21 CFSP COMM FAULT SV
 03 2A LorE COMM rE/LOC
 03 2B nAt NO ACTIVITY TIMER VALUE
 03 29 CFLt COMM FAULT MODE 2/1

WRITE COMMANDS

 04 05 Auto = On
 04 06 Auto = OFF
 02 0F SP1'S MANUAL VALUE
 02 10 SP2'S MANUAL VALUE
 04 07 PEAK RESET
 04 08 VALLEY RESET

 04 0B PctO = On
 04 0C PctO = OFF

 02 0E CFSP COMM FAULT SV XX
 04 00 LorE = rE (REMOTE)
 04 01 LorE = LOC (LOCAL)

00 = Positive Value

FF = Negative Value

04 0D RESET "ENTER PRESSED AT CN76000"
 FLAG TO "0"

READ COMMAND DESCRIPTION

Conventions Used

All commands appear in boldface type. Received data appears in plain type.

COMMAND = 00

READ "PV" PROCESS VARIABLE VALUE W/STATUS

RECEIVED DATA =

1st CHAR (BINARY)

AUTO ON/OFF 1=ON 0=OFF, MANUAL	COMM REM/LOC 1=REMOTE 0=LOCAL	ENTER KEY PRESSED? 1=YES 0=NO	ERROR PRESENT? 1=YES 0=NO, OK
-----------------------------------------	----------------------------------------	----------------------------------------	----------------------------------------

2nd CHAR (BINARY)

ALARM RELAY 1=ENERGIZED 0=DEENERGIZED	[NU] not used	SV TYPE CFSV/LOCAL 1=CFSV 0=LOCAL	[NU] not used
---------------------------------------------	------------------	--------------------------------------------	------------------

3rd CHAR (BINARY)

[NU] not used	[NU] not used	[NU] not used	[NU] not used
------------------	------------------	------------------	------------------

4th CHAR (BINARY)

[NU] not used	[NU] not used	NO ACTIVITY TIMER nAt 1=TIMEOUT 0=NORMAL, OK	PV SIGN NEG/POS 1=NEGATIVE 0=POSITIVE
------------------	------------------	-------------------------------------------------------	------------------------------------------------

5th CHAR MOST SIGNIFICANT DIGIT PROCESS VARIABLE VALUE
6th CHAR PROCESS VARIABLE VALUE
7th CHAR PROCESS VARIABLE VALUE
8th CHAR LEAST SIGNIFICANT DIGIT PROCESS VARIABLE VALUE

NOTE: If ERROR PRESENT BIT = "1", issue a FULL STATUS READ COMMAND "05" to determine the specific error.

READ COMMAND DESCRIPTION (CONT'D)

COMMAND = 05

READ FULL STATUS

RECEIVED DATA =

1st CHAR (BINARY)

*1=FAIL TEST	[NU] not used	*1=CHECK CAL	*1=OFL OVERFLOW
-----------------	------------------	-----------------	--------------------

2nd CHAR (BINARY)

*1=UFL UNDERFLOW	*1=BAD INPUT	*1=OPEN INPUT	*1=AREA
---------------------	-----------------	------------------	---------

3rd CHAR (BINARY)

4th CHAR (BINARY)

[NU] not used	[NU] not used	[NU] not used	[NU] not used
------------------	------------------	------------------	------------------

5th CHAR (BINARY)

[NU] not used	[NU] not used	1=IN PRIMARY OR SECOND- ARY MENU ITEMS	1=IN SECURE MENU ITEM
------------------	------------------	-------------------------------------------------	--------------------------

6th CHAR (BINARY)

[NU] not used	OUTA 1=ENERGIZED 0=DEENERGIZED	OUTB 1=ENERGIZED 0=DEENERGIZED	ALARM RELAY 1=ENERGIZED 0=DEENERGIZED
------------------	--------------------------------------	--------------------------------------	---------------------------------------------

7th CHAR (BINARY)

*1=CHECK CALIBRATION	*1=LOOP BREAK	*1=SENSOR RATE OF CHANGE	[NU] not used
-------------------------	------------------	-----------------------------	------------------

8th CHAR (BINARY)

9th CHAR (BINARY)

10th CHAR (BINARY)

[NU] not used	[NU] not used	[NU] not used	[NU] not used
------------------	------------------	------------------	------------------

NOTE: Bits marked with an "*" are error bits. Any one of these will set the "ERROR PRESENT" BIT in the "PV" PROCESS VARIABLE'S STATUS (COMMAND 00).

READ COMMAND DESCRIPTION (CONT'D)

COMMAND = 0100	READ "SP1 " VALUE
COMMAND = 0102	READ "SP2 " VALUE
COMMAND = 0104	READ "ALLo" VALUE
COMMAND = 0105	READ "ALHi" VALUE
COMMAND = 0124	READ "InPC" INPUT CORRECTION VALUE
COMMAND = 0116	READ "SCAL" VALUE
COMMAND = 0117	READ "SCAH" VALUE
COMMAND = 0110	READ "SPL " VALUE
COMMAND = 0111	READ "SPH " VALUE
COMMAND = 011A	READ "PEA " PEAK VALUE
COMMAND = 011B	READ "VAL " VALLEY VALUE
COMMAND = 0121	READ "CFSP" COMM FAULT SV VALUE

DATA =		
1st CHAR	SIGN 0 = POSITIVE,	(NOT 0) = NEGATIVE
2nd CHAR	SIGN 0 = POSITIVE,	(NOT 0) = NEGATIVE
3rd CHAR	MSD VALUE	BOTH 1st & 2nd CHARS
4th CHAR	VALUE	MUST BE "0" FOR SIGN
5th CHAR	VALUE	TO BE POSITIVE.
6th CHAR	LSD VALUE	

COMMAND = 0107	READ "SP1d" VALUE
COMMAND = 0108	READ "PUL1" VALUE
COMMAND = 0112	READ "S1OL" VALUE
COMMAND = 0113	READ "S1OH" VALUE
COMMAND = 010A	READ "SP2d" VALUE
COMMAND = 010B	READ "PUL2" VALUE
COMMAND = 0114	READ "S2OL" VALUE
COMMAND = 0115	READ "S2OH" VALUE
COMMAND = 010C	READ "Pb1 " VALUE
COMMAND = 010D	READ "Pb2 " VALUE
COMMAND = 010E	READ "rES " RESET VALUE
COMMAND = 010F	READ "rtE " RATE VALUE
COMMAND = 0125	READ "ArTE" VALUE
COMMAND = 0118	READ "InPt" INPUT FAULT TIMER VALUE
COMMAND = 0129	READ "SEnC" VALUE
COMMAND = 0128	READ "LPbr" VALUE
COMMAND = 011E	READ SP1'S MANUAL VALUE
COMMAND = 012A	READ SP2'S MANUAL VALUE
COMMAND = 0126	READ "1rt " RAMP TIME
COMMAND = 0127	READ "1st " SOAK TIME

DATA =	
1st CHAR	[NU] not used
2nd CHAR	[NU] not used
3rd CHAR	MSD VALUE
4th CHAR	VALUE
5th CHAR	VALUE
6th CHAR	LSD VALUE

COMMAND = 032D	READ "dFAC" VALUE
COMMAND = 0333	READ "Filt" VALUE
COMMAND = 032B	READ "nAt " NO ACTIVITY TIMER VALUE

DATA =	
1st CHAR	MSD VALUE
2nd CHAR	LSD VALUE

READ COMMAND DESCRIPTION (CONT'D)

DATA =		NOT 0	0	
COMMAND = 0313	READ "S1St"	dir	rE	
COMMAND = 0314	READ "S1LP"	0 on	OoFF	
COMMAND = 0335	READ "SP1o"	Outb	OutA	
COMMAND = 0336	READ "S2t "	AbS	dE	
COMMAND = 0315	READ "S2St"	dir	rE	
COMMAND = 0316	READ "S2LP"	0 on	OoFF	
COMMAND = 0312	READ "Strt"	YES	no	
COMMAND = 0338	READ "LErn"	Cont	End	
COMMAND = 032C	READ "rES "	AUTO	OFS	(OFS = OFFSET MODE)
COMMAND = 032E	READ "Pid2"	On	OFF	
COMMAND = 032F	READ "ArUP"	On	OFF	
COMMAND = 0317	READ "ALt "	AbS	dE	
COMMAND = 031B	READ "ALrE"	OnOF	Hold	
COMMAND = 031C	READ "ALPi"	On	OFF	
COMMAND = 033A	READ "ALiH"	On	OFF	
COMMAND = 0318	READ "ALSt"	OPEn	CLOS	
COMMAND = 0319	READ "ALLP"	0 on	OoFF	
COMMAND = 0322	READ "ALbr"	On	OFF	
COMMAND = 0325	READ "OSUP"	On	OFF	(For Curr or Volt input)
COMMAND = 0310	READ "Unit"	F	C	
COMMAND = 0328	READ "Auto"	On	OFF	(OFF = MANUAL MODE)
COMMAND = 0327	READ "PctO"	On	OFF	
COMMAND = 0330	READ "Prog"	On	OFF	
COMMAND = 0331	READ "StAt"	On	OFF	
COMMAND = 0332	READ "PEnd"	OoFF	Hold	
COMMAND = 032A	READ "LorE"	rE	LOC	
COMMAND = 0329	READ "CFLt"	2	1	COMM FAULT MODE
DATA =		NOT 0	0	
1st CHAR		NOT 0	0	BOTH 1st & 2nd
2nd CHAR		NOT 0	0	CHARS MUST BE "0"

COMMAND = 0106 READ "CY1 " VALUE & "S1Ot" OUTPUT TYPE
 COMMAND = 0109 READ "CY2 " VALUE & "S2Ot" OUTPUT TYPE

DATA =
 1st CHAR & 2nd CHAR

10	= "OnOF"	ON-OFF OUTPUT
08	= "PUL "	PULSE OUTPUT
04	= "Volt"	VOLTAGE OUTPUT
02	= "Curr"	CURRENT OUTPUT
01	= "Ft "	FAST TIME PROPORTIONING OUTPUT
00	= "CY "	TIME PROPORTIONING OUTPUT

IF OUTPUT = "CY ", THEN 3rd & 4th CHARS = CYCLE RATE VALUE

3rd CHAR	MSD VALUE OF CY#
4th CHAR	LSD VALUE OF CY#
5th CHAR	[NU] not used
6th CHAR	[NU] not used

READ COMMAND DESCRIPTION (CONT'D)

COMMAND = 011D READ SP1 or SP2 PERCENT OUTPUT VALUE

DATA =

1st CHAR	0 = SP1's PCT OUT,	NOT 1 = SP2's PCT OUT
2nd CHAR	0 = SP1's PCT OUT,	NOT 1 = SP2's PCT OUT
3rd CHAR	[NU] not used	BOTH 1st & 2nd CHARS
4th CHAR	MSD VALUE	MUST BE "0" TO INDICATE
5th CHAR	VALUE	SP1's PCT OUT
6th CHAR	LSD VALUE	

NOTES:

- The value in 1st, & 2nd CHAR is used to indicate that SP2's PERCENT OUTPUT VALUE is being read.
- Decimal point is implied between 5th and 6th characters (e.g., 495 = 49.5%)
- SP2's PERCENT OUTPUT will only be available if configured as a dual type instrument.
- SP1 & SP2 PERCENT OUTPUTS are alternated at a 1 second rate.

COMMAND = 0323 READ "InP" INPUT TYPE

DATA =

1st CHAR	[NU] not used	
2nd CHAR	1 = J-IC	9 = b-
	2 = CA	A = C-
	3 = E-	B = P392 RTD
	4 = t-	C = n120 RTD
	5 = L-	D = P385 RTD
	6 = n-	E = Curr
	7 = r-13	F = Volt
	8 = S-10	

COMMAND = 0324 READ "dPt" DECIMAL POINTS

DATA =

1st CHAR	[NU] not used	
2nd CHAR	0 = NONE	2 = 0.00
	1 = 0.0	3 = 0.000

COMMAND = 0326 READ "Unit" FOR CN76000 WITH CURRENT OR VOLTAGE

DATA =

1st CHAR	[NU] not used	See Command 0310
2nd CHAR	0 = NONE	for reading T/C and RTD
	1 = F	
	2 = C	

COMMAND = 0334 READ "SECr" SECURITY MODE

DATA =

1st CHAR	[NU] not used	
2nd CHAR	0 = 1 AS VIEWED IN SECr	
	1 = 2 AS VIEWED IN SECr	
	2 = 3 AS VIEWED IN SECr	
	3 = 4 AS VIEWED IN SECr	

READ COMMAND DESCRIPTION (CONT'D)

COMMAND = 0337 READ "AL" ALARM MODE

DATA =
 1st CHAR 0 = OFF
 1 = Lo
 2 = Hi
 3 = HiLo
 2nd CHAR [NU] not used

COMMAND = 0339 READ "tune" TUNE MODE

DATA =
 1st CHAR 0 = SELF
 1 = Pid (FULL)
 2 = SLO
 3 = nor
 4 = FAST
 2nd CHAR [NU] not used

WRITE COMMAND DESCRIPTION

Conventions Used

All commands appear in boldface type. Received data appears in plain type.

COMMAND = 0200	[DATA]	CHANGE "SP1 " VALUE	
COMMAND = 0202	[DATA]	CHANGE "SP2 " VALUE	
COMMAND = 0204	[DATA]	CHANGE "ALLO" VALUE	
COMMAND = 0205	[DATA]	CHANGE "ALHi" VALUE	
COMMAND = 020E	[DATA]	CHANGE "CFSP" VALUE	(COMMUNICATION FAIL SET POINT)

DATA =

1st CHAR	WRITE COMMAND
2nd CHAR	WRITE COMMAND
3rd CHAR	WRITE COMMAND
4th CHAR	WRITE COMMAND

DATA =

5th CHAR	MSD VALUE
6th CHAR	VALUE
7th CHAR	VALUE
8th CHAR	LSD VALUE
9th CHAR	SIGN 0 = POSITIVE, NOT 0 = NEGATIVE
10th CHAR	SIGN 0 = POSITIVE, NOT 0 = NEGATIVE

BOTH 9th & 10th CHARS MUST BE "0"
FOR SIGN TO BE ACCEPTED AS POSITIVE.

COMMAND = 0206	[DATA]	CHANGE "CY1" CYCLE RATE VALUE FOR SP1
COMMAND = 0207	[DATA]	CHANGE "CY2" CYCLE RATE VALUE FOR SP2

DATA =

1st CHAR	WRITE COMMAND
2nd CHAR	WRITE COMMAND
3rd CHAR	WRITE COMMAND
4th CHAR	WRITE COMMAND

DATA =

5th CHAR	[NU] not used, set to 0
6th CHAR	[NU] not used, set to 0
7th CHAR	MSD VALUE
8th CHAR	LSD VALUE
9th CHAR	[NU] not used, set to 0
10th CHAR	[NU] not used, set to 0

NOTES:

- The CY CYCLE RATE VALUE must be an even number, between 2 and 80, i.e., 02, 04, 10, etc. Odd numbers or out of range numbers will not be accepted.
- An error response will be generated if CY CHANGE is attempted and the output type S1Ot is not set to CY.

WRITE COMMAND DESCRIPTION (CONT'D)

COMMAND = 0208 [DATA] CHANGE "Pb1 " VALUE
 COMMAND = 0209 [DATA] CHANGE "Pb2 " VALUE
 COMMAND = 020A [DATA] CHANGE "rES " RESET VALUE & SET IN AUTOMATIC RESET MODE
 COMMAND = 020B [DATA] CHANGE "OFS " OFFSET VALUE & SET IN OFFSET MODE
 COMMAND = 020C [DATA] CHANGE "rtE " RATE VALUE
 COMMAND = 020F [DATA] CHANGE SP1's AUTO/MAN "MANUAL" VALUE
 COMMAND = 0210 [DATA] CHANGE SP2's AUTO/MAN "MANUAL" VALUE

DATA =
 1st CHAR WRITE COMMAND
 2nd CHAR WRITE COMMAND
 3rd CHAR WRITE COMMAND
 4th CHAR WRITE COMMAND

DATA =
 5th CHAR MSD VALUE
 6th CHAR VALUE
 7th CHAR VALUE
 8th CHAR LSD VALUE

 9th CHAR [NU] not used, set to 0
 10th CHAR [NU] not used, set to 0

COMMAND = 0403 CHANGE "tunE" MODE = SELF
 COMMAND = 0404 CHANGE "tunE" MODE = Pid (FULL)
 COMMAND = 0402 CHANGE ALARM ACK
 COMMAND = 0405 CHANGE "Auto" = On
 COMMAND = 0406 CHANGE "Auto" = OFF (MANUAL)
 COMMAND = 0407 CHANGE PEAK RESET
 COMMAND = 0408 CHANGE VALLEY RESET
 COMMAND = 040B CHANGE "PctO" = On
 COMMAND = 040C CHANGE "PctO" = OFF
 COMMAND = 0400 CHANGE "LorE" = rE (REMOTE)
 COMMAND = 0401 CHANGE "LorE" = LOC (LOCAL)
 COMMAND = 040D CHANGE RESET "ENTER PRESSED" FLAG TO "0"

DATA =
 1st CHAR WRITE COMMAND
 2nd CHAR WRITE COMMAND
 3rd CHAR WRITE COMMAND
 4th CHAR WRITE COMMAND

NOTE: Data is not required for these commands. They are specific as to their function.

SECTION 3 - OPTION 4SP 4 STAGE SETPOINT

OPTION 4SP READ COMMAND DESCRIPTION

COMMAND = 00

READ "PV" PROCESS VARIABLE VALUE W/STATUS

DATA = 1st CHAR (BINARY)

AUTO ON/OFF 1=ON 0=OFF, MANUAL	COMM REM/LOC 1=REMOTE 0=LOCAL	ENTER@ CN76000 MADE 1=ENTER 0=NONE	ERROR PRESENT 1=ERROR 0=NONE, OK
-----------------------------------------	----------------------------------------	---------------------------------------------	-------------------------------------------

2nd CHAR (BINARY)

ALARM RELAY 1=ENERGIZED 0=DEENERGIZED	NU not used	SV TYPE CFSV/LOCAL 1=CFSV 0=LOCAL	NU not used
---------------------------------------------	----------------	--------------------------------------------	----------------

3rd CHAR (BINARY) Option 4SP 4 Stage Set Point Active Set Point

NU not used	NU not used	0	0	= 1SP1
		0	1	= 2SP1
		1	0	= 3SP1
		1	1	= 4SP1

4th CHAR (BINARY)

NU not used	NU not used	NO ACTIVITY TIMER nAt 1=TIMEOUT 0=NORMAL, OK	PV SIGN NEG/POS 1=NEGATIVE 0=POSITIVE
----------------	----------------	-------------------------------------------------------	------------------------------------------------

5th CHAR MSD PROCESS VARIABLE VALUE
6th CHAR PROCESS VARIABLE VALUE
7th CHAR PROCESS VARIABLE VALUE
8th CHAR LSD PROCESS VARIABLE VALUE

NOTE: If ERROR PRESENT BIT = "1", issuing a FULL STATUS READ COMMAND "05" can be used to determine the specific error.

OPTION 4SP READ AND WRITE COMMAND SUMMARY

READ COMMANDS

01 00 1SP1 VALUE
01 01 2SP1 VALUE
01 2D 3SP1 VALUE
01 2E 4SP1 VALUE

03 39 1tun SELF/Pid/SLO/nor/FAST
03 3B 2tun SELF/Pid/SLO/nor/FAST
03 3C 3tun SELF/Pid/SLO/nor/FAST
03 3D 4tun SELF/Pid/SLO/nor/FAST

03 12 Strt. YES/no 1SP1
03 3E Strt YES/no 2SP1
03 3F Strt YES/no 3SP1
03 40 Strt YES/no 4SP1

03 38 LErn Cont/End 1SP1
03 41 LErn Cont/End 2SP1
03 42 LErn Cont/End 3SP1
03 43 LErn Cont/End 4SP1

03 2D dFAC VALUE 1SP1
03 44 dFAC VALUE 2SP1
03 45 dFAC VALUE 3SP1
03 46 dFAC VALUE 4SP1

01 0C 1Pb1 VALUE
01 2F 2Pb1 VALUE
01 30 3Pb1 VALUE
01 31 4Pb1 VALUE

03 2C 1rES RESET AUTO/OFS MODE
03 47 2rES RESET AUTO/OFS MODE
03 48 3rES RESET AUTO/OFS MODE
03 49 4rES RESET AUTO/OFS MODE

01 0E 1rES RESET VALUE
01 32 2rES RESET VALUE
01 33 3rES RESET VALUE
01 34 4rES RESET VALUE

01 0F 1rtE RATE VALUE
01 35 2rtE RATE VALUE
01 36 3rtE RATE VALUE
01 37 4rtE RATE VALUE

03 4A SP 1/2/3/4 #SP1

03 4B SPSA rE/Int

WRITE COMMANDS

00 = Positive Value

FF= Negative Value

02 00 1SP1 VALUE XX
02 01 2SP1 VALUE XX
02 11 3SP1 VALUE XX
02 12 4SP1 VALUE XX

04 03 1tun MODE = SELF
04 0E 2tun MODE = SELF
04 0F 3tun MODE = SELF
04 10 4tun MODE = SELF

04 04 1tun MODE = FULL Pid
04 11 2tun MODE = FULL Pid
04 12 3tun MODE = FULL Pid
04 13 4tun MODE = FULL Pid

02 08 1Pb1 VALUE
02 13 2Pb1 VALUE
02 14 3Pb1 VALUE
02 15 4Pb1 VALUE

02 0A 1rES VALUE AUTO MODE
02 16 2rES VALUE AUTO MODE
02 17 3rES VALUE AUTO MODE
02 18 4rES VALUE AUTO MODE

02 0B 1OFS VALUE OFFSET MODE
02 19 2OFS VALUE OFFSET MODE
02 1A 3OFS VALUE OFFSET MODE
02 1B 4OFS VALUE OFFSET MODE

02 0C 1rtE RATE VALUE
02 1C 2rtE RATE VALUE
02 1D 3rtE RATE VALUE
02 1E 4rtE RATE VALUE

04 14 SP = 1SP1
04 15 SP = 2SP1
04 16 SP = 3SP1
04 17 SP = 4SP1

OPTION 4SP READ COMMAND DESCRIPTION

Conventions Used

All commands appear in boldface type. Received data appears in plain type.

COMMAND = 0100	READ "1SP1" VALUE
COMMAND = 0101	READ "2SP1" VALUE
COMMAND = 012D	READ "3SP1" VALUE
COMMAND = 012E	READ "4SP1" VALUE
DATA = 1st CHAR	SIGN 0 = POSITIVE, (NOT 0) = NEGATIVE
2nd CHAR	SIGN 0 = POSITIVE, (NOT 0) = NEGATIVE
3rd CHAR	MSD VALUE BOTH 1st & 2nd CHARS
4th CHAR	VALUE MUST BE "0" FOR SIGN
5th CHAR	VALUE TO BE POSITIVE.
6th CHAR	LSD VALUE
COMMAND = 010C	READ "1Pb1" VALUE under 1SP1
COMMAND = 012F	READ "2Pb1" VALUE under 2SP1
COMMAND = 0130	READ "3Pb1" VALUE under 3SP1
COMMAND = 0131	READ "4Pb1" VALUE under 4SP1
COMMAND = 010E	READ "1rES" RESET VALUE under 1SP1
COMMAND = 0132	READ "2rES" RESET VALUE under 2SP1
COMMAND = 0133	READ "3rES" RESET VALUE under 3SP1
COMMAND = 0134	READ "4rES" RESET VALUE under 4SP1
COMMAND = 010F	READ "1rtE" RATE VALUE under 1SP1
COMMAND = 0135	READ "2rtE" RATE VALUE under 2SP1
COMMAND = 0136	READ "3rtE" RATE VALUE under 3SP1
COMMAND = 0137	READ "4rtE" RATE VALUE under 4SP1
DATA = 1st CHAR	NU not used
2nd CHAR	NU not used
3rd CHAR	MSD VALUE
4th CHAR	VALUE
5th CHAR	VALUE
6th CHAR	LSD VALUE
COMMAND = 032D	READ "dFAC" VALUE under 1SP1
COMMAND = 0344	READ "dFAC" VALUE under 2SP1
COMMAND = 0345	READ "dFAC" VALUE under 3SP1
COMMAND = 0346	READ "dFAC" VALUE under 4SP1
DATA = 1st CHAR	MSD VALUE
2nd CHAR	LSD VALUE
COMMAND = 0339	READ "1tun" TUNE MODE under 1SP1
COMMAND = 033B	READ "2tun" TUNE MODE under 2SP1
COMMAND = 033C	READ "3tun" TUNE MODE under 3SP1
COMMAND = 033D	READ "4tun" TUNE MODE under 4SP1
DATA = 1st CHAR	0 = SELF
	1 = Pid (FULL)
	2 = SLO
	3 = nor
	4 = FAST
2nd CHAR	NU not used

OPTION 4SP READ COMMAND DESCRIPTION (CONT'D)

		NOT 0	0	
COMMAND = 0312	READ "Strt"	YES	no	under 1SP1
COMMAND = 033E	READ "Strt"	YES	no	under 2SP1
COMMAND = 033F	READ "Strt"	YES	no	under 3SP1
COMMAND = 0340	READ "Strt"	YES	no	under 4SP1
COMMAND = 0338	READ "LErn"	Cont	End	under 1SP1
COMMAND = 0341	READ "LErn"	Cont	End	under 2SP1
COMMAND = 0342	READ "LErn"	Cont	End	under 3SP1
COMMAND = 0343	READ "LErn"	Cont	End	under 4SP1
COMMAND = 032C	READ "1rES"	AUTO	OFS	under 1SP1
COMMAND = 0347	READ "2rES"	AUTO	OFS	under 2SP1
COMMAND = 0348	READ "3rES"	AUTO	OFS	under 3SP1
COMMAND = 0349	READ "4rES"	AUTO	OFS	under 4SP1
OFS=OFFSET MODE				
COMMAND = 034B	READ "SPSA"	rE	Int	Option 485 forces 4SP SPSA=Int Menu is not affected
DATA = 1st CHAR		NOT 1	0	BOTH 1st & 2nd
2nd CHAR		NOT 1	0	CHARS MUST BE "0"

COMMAND = 034A READ "SP " CURRENT SET POINT SELECTION

DATA = 1st CHAR NU not used
2nd CHAR 0 = 1SP1 IS CURRENTLY SELECTED
1 = 2SP1 IS CURRENTLY SELECTED
2 = 3SP1 IS CURRENTLY SELECTED
3 = 4SP1 IS CURRENTLY SELECTED

OPTION 4SP WRITE COMMAND DESCRIPTION

Conventions Used

All commands appear in boldface type. Received data appears in plain type.

COMMAND = 0200 [DATA] CHANGE "1SP1" VALUE
 COMMAND = 0201 [DATA] CHANGE "2SP1" VALUE
 COMMAND = 0211 [DATA] CHANGE "3SP1" VALUE
 COMMAND = 0212 [DATA] CHANGE "4SP1" VALUE

1st CHAR WRITE COMMAND
 2nd CHAR WRITE COMMAND
 3rd CHAR WRITE COMMAND
 4th CHAR WRITE COMMAND

DATA = 5th CHAR MSD VALUE
 6th CHAR VALUE
 7th CHAR VALUE
 8th CHAR LSD VALUE
 9th CHAR SIGN 0 = POSITIVE, NOT 1 = NEGATIVE
 10th CHAR SIGN 0 = POSITIVE, NOT 1 = NEGATIVE

BOTH 9th & 10th CHARS MUST BE "0"
 FOR SIGN TO BE ACCEPTED AS POSITIVE.

COMMAND = 0208 [DATA] CHANGE "1Pb1" VALUE under 1SP1
 COMMAND = 0213 [DATA] CHANGE "2Pb1" VALUE under 2SP1
 COMMAND = 0214 [DATA] CHANGE "3Pb1" VALUE under 3SP1
 COMMAND = 0215 [DATA] CHANGE "4Pb1" VALUE under 4SP1

RESET VALUE & SET IN AUTO MODE
 COMMAND = 020A [DATA] CHANGE "1rES" RESET VALUE IN AUTO under 1SP1
 COMMAND = 0216 [DATA] CHANGE "2rES" RESET VALUE IN AUTO under 2SP1
 COMMAND = 0217 [DATA] CHANGE "3rES" RESET VALUE IN AUTO under 3SP1
 COMMAND = 0218 [DATA] CHANGE "4rES" RESET VALUE IN AUTO under 4SP1

OFFSET VALUE & SET IN OFFSET MODE
 COMMAND = 020B [DATA] CHANGE "1OFS" OFFSET VALUE IN OFS under 1SP1
 COMMAND = 0219 [DATA] CHANGE "2OFS" OFFSET VALUE IN OFS under 2SP1
 COMMAND = 021A [DATA] CHANGE "3OFS" OFFSET VALUE IN OFS under 3SP1
 COMMAND = 021B [DATA] CHANGE "4OFS" OFFSET VALUE IN OFS under 4SP1

COMMAND = 020C [DATA] CHANGE "1rtE" RATE VALUE under 1SP1
 COMMAND = 021C [DATA] CHANGE "2rtE" RATE VALUE under 2SP1
 COMMAND = 021D [DATA] CHANGE "3rtE" RATE VALUE under 3SP1
 COMMAND = 021E [DATA] CHANGE "4rtE" RATE VALUE under 4SP1

1st CHAR WRITE COMMAND
 2nd CHAR WRITE COMMAND
 3rd CHAR WRITE COMMAND
 4th CHAR WRITE COMMAND

DATA = 5th CHAR MSD VALUE
 6th CHAR VALUE
 7th CHAR VALUE
 8th CHAR LSD VALUE

9th CHAR NU not used, set to 0
 10th CHAR NU not used, set to 0

OPTION 4SP WRITE COMMAND DESCRIPTION

COMMAND = 0403	CHANGE "1tun" MODE = SELF under 1SP1
COMMAND = 040E	CHANGE "2tun" MODE = SELF under 2SP1
COMMAND = 040F	CHANGE "3tun" MODE = SELF under 3SP1
COMMAND = 0410	CHANGE "4tun" MODE = SELF under 4SP1
COMMAND = 0404	CHANGE "1tun" MODE = Pid (FULL) under 1SP1
COMMAND = 0411	CHANGE "2tun" MODE = Pid (FULL) under 2SP1
COMMAND = 0412	CHANGE "3tun" MODE = Pid (FULL) under 3SP1
COMMAND = 0413	CHANGE "4tun" MODE = Pid (FULL) under 4SP1
COMMAND = 0414	CHANGE "SP " CURRENT SET POINT = 1SP1
COMMAND = 0415	CHANGE "SP " CURRENT SET POINT = 2SP1
COMMAND = 0416	CHANGE "SP " CURRENT SET POINT = 3SP1
COMMAND = 0417	CHANGE "SP " CURRENT SET POINT = 4SP1

1st CHAR	WRITE COMMAND
2nd CHAR	WRITE COMMAND
3rd CHAR	WRITE COMMAND
4th CHAR	WRITE COMMAND

NOTE: Data is not required for these commands. They are specific as to their function.

SECTION 4 - SAMPLE PROGRAMS

SAMPLE PROGRAM #1

PROGRAM USING THE OMEGA A1000 RS-232/RS-485 CONVERTER

```
10 CLS:KEY OFF                                'CLEAR SCREEN AND SOFT KEYS
30 KEY 1,""
40 LOCATE 25,30
50 PRINT "<<STRIKE F1 KEY TO EXIT>>"          'EXIT INSTRUCTIONS
60 LOCATE 1,1,1                               'HOME CURSOR
70 ON KEY (1) GOSUB 240
80 KEY (1) ON
90 OPEN "com1:9600,n,8,1,rs,cs,cd,ds" AS #1    "OPEN COM PORT
100 INPUT " ADDRESS OF THE CN76000:";AD$
110 INPUT " ENTER COMMAND AND DATA WITH NO SPACES";CMD$
120 TEX$ = AD$ +CMD$                          'COMBINE ADDRESS AND COMMAND STRING
130 GOSUB 270
140 TEX$ = CHR$(2) + "L" + TEX$ + CS$+ CHR$(3) + CHR$(0):RES$="" :CAR$=""
150 PRINT #1, TEX$
160 REM GET RESPONSE FROM CN76000
170 CAR$ = INPUT$(1,#1)
180 RES$ = RES$ + CAR$
190 IF CAR$ <> CHR$(6) THEN GOTO 170
200 REM PRINT RESPONSE FROM CN76000
210 PRINT:PRINT " ADDRESS "; AD$
220 PRINT " THE RESPONSE IS ";RES$:PRINT:PRINT
230 GOSUB 100
240 CLOSE #1
250 END
260 REM CALCULATE CHECKSUM VALUE
270 LN = LEN(TEX$) :S=0
280 FOR I = 1 TO LN : B = (ASC(MID$(TEX$,I,1))):S =S+B:NEXT I
290 CS$ = RIGHT$(HEX$(S) ,2)
300 RETURN
```

SAMPLE PROGRAM #2

PROGRAM TO COMMUNICATE WITH AN OMEGA CN76000 SERIES CONTROLLER

```
10 REM OMEGACOM.BAS
20 REM Program to communicate with an OMEGA CN76000 Series Control
30 Written in BASIC for IBM PC or compatible computer
40 Assume that control is set up for address 32, at 9600 baud

100 CLS
110 OPEN "com1: 9600,N,8,1,cs" FOR RANDOM AS #1: REM open port 1, 9600 baud
120 RTS% = INP(&H3FC): REM ) Set up variables
130 RTSR% = RTS% AND &HFD: REM ) for comm direction
140 RTST% = RTS% OR &H2: REM ) control
150 STX$ = CHR$(2): REM start transmission character
160 ETX$ = CHR$(3): REM end transmission character
170 ACK$ = CHR$(6): REM acknowledge character
180 NUL$ = CHR$(0): REM null character

200 TXCMD$ = STX$ + "L32" + "00" + "C5" + ETX$ + NUL$: REM L32 = control
address
210 REM
220 REM
230 PRINT TXCMD$: SPC(10);
240 GOSUB 1000: REM transmit / receive subroutine
250 PRINT RXDATA$
260 KY$ = INKEY$
270 CLOSE #1
280 END

1000 TMO% = 0: REM Set up time out counter
1010 RXDATA$ = "": REM Clear receive buffer

2000 OUT &H3FC, RTST%: REM set RTS line
2010 PRINT #1, TXCMD$: REM Send data packet. Don't forget semicolon!
2020 IF (INP(&H3FD) AND &H60) <> &H60 THEN GOTO 2020: REM wait until xmit
buffer
2030 REM
2040 OUT &H3FC, RTSR%: REM reset RTS line
2050 IF LOC(1) = 0 THEN GOTO 3050: REM if receive buffer empty skip read
2060 RX$ = INPUT$(1, #1): REM fetch character
2070 RXDATA$ = RXDATA$ + RX$: REM Add to the end
2080 IF INSTR(1, RX$, ACK$) > 0 THEN GOTO 3080: REM End of RX packet?
2090 TMO% = TMO% + 1: REM time out counter
2100 IF TMO% < 2500 THEN GOTO 3010
2110 RXDATA$ = "Time Out"
2120 RETURN
```



WARRANTY/DISCLAIMER

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

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.

OMEGA's policy is to make running changes, not model changes, whenever an improvement is possible. This affords our customers 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 Gages
- ☑ 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