2000-06

Allen-Bradley Data Highway PLC Interface

Printed in the United States of America
Part Number 96760-006

XYCOM
750 Nortl
Saline, M
(313) 429

XYCOM REVISION RECORD

Revision	*Description*	*Date*
A	Appendix Released	6/90

Address comments concerning this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number *96760-006A*

A 1 INTRODUCTION

This appendix contains information that is specific to the 2000-06 Allen-Bradley Data] PLC interface The protocol used by the firmware to communicate with the Data Hi a full-duplex, asynchronous, point-to-point protocol that closely conforms to the AN standard

The specific commands that allow communications are discussed in Section A 5 Th

GET
GETDIAG
GETIO
PUT
PUTIO

A 2 CABLING THE WORKSTATION TO THE PLC

The PLC and/or network connects to the 9-pin primary serial port of the 2000 Workst a cable This is not a standard cable The connectors must be fabricated Figure A next page illustrates the connections that must be made between the primary port and

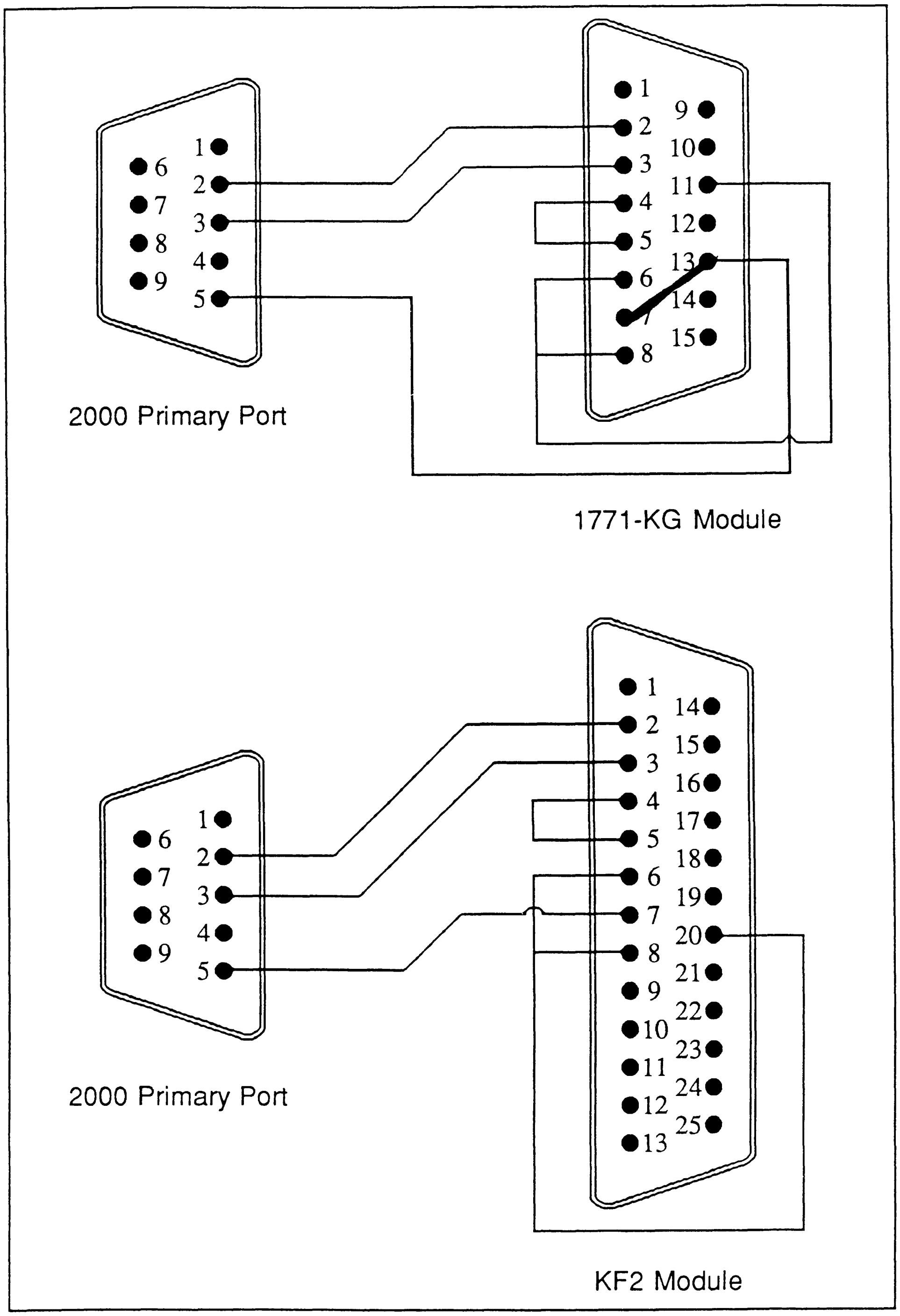


Figure A-1 Connections

The 2000-06 can be attached to one PLC or to a Data Highway Network The possibili
shown below

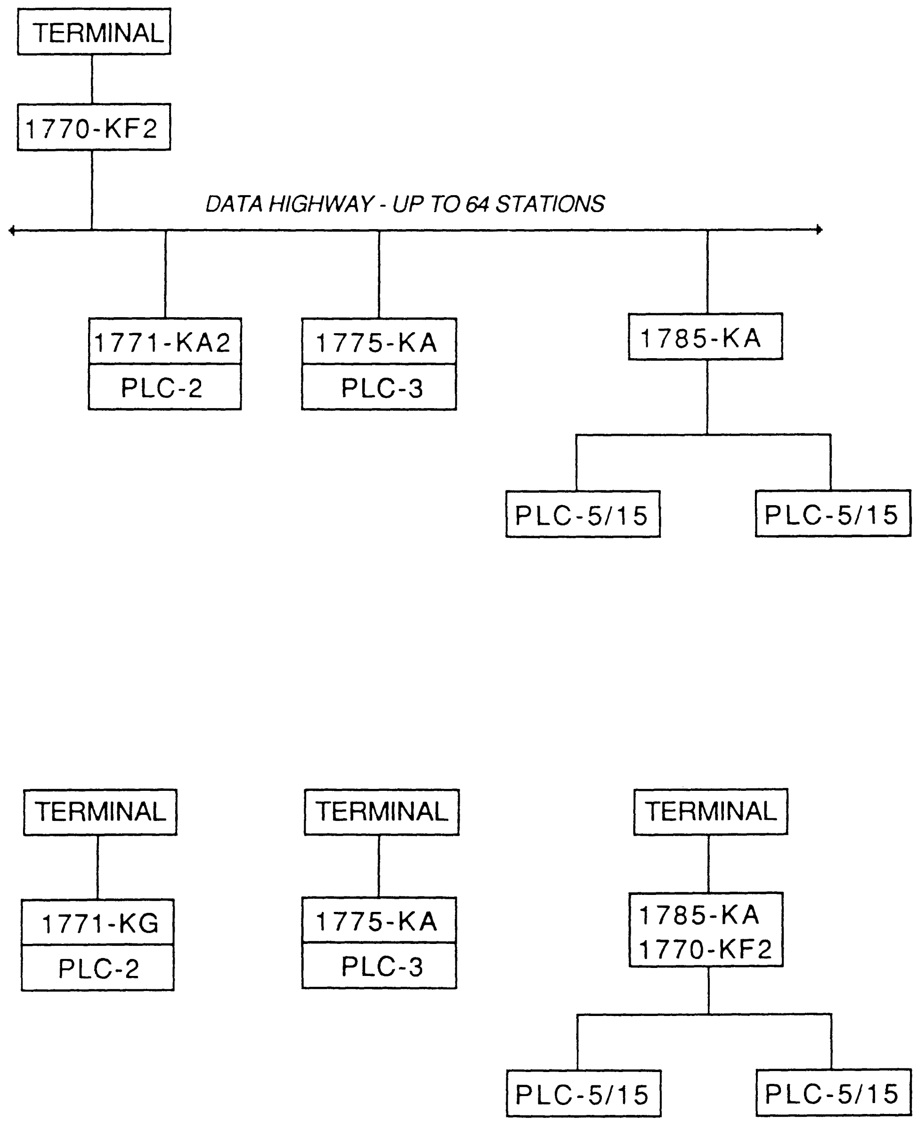


Figure A-2 Configurations

A 3 CONFIGURATION MENU

The configuration menus are called up from the Main Menu The Data Highw
Configuration Menu looks like the following

```
-- Data Highway Port Configuration Menu--

6 Baud - 1=300 2=600 3=1200 4=2400 5=4800 6=9600
1=Even Parity ------ 0=No Parity
1=Background Mode -- 0=Foreground Mode
ACK Timeout Value (1 - 6 seconds)
Response Timeout Value (1 -30 seconds)
Station Number (NNN octal)
1=Enable Multiplex Handshaking -- 0=Disable Multiplex Handshaking


Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW
Use values 0 through 9
"C" for next configuration menu, <ESC> or <ENTER> to quit
```

Figure A-3 Configuration Menu

Baud The baud rate of the channel should be set to match that of the PLC

Parity This option will set the parity used in communication on the Data Highway none or even The type selected should match the other communication device(s)

Mode This sets the Mode that the OIL commands will operate in with the Data Hig Foreground Mode the Data Highway OIL interface command executes and retuı finished

In the Background Mode, the Data Highway OIL command initiates execution and giv back to the OIL program When the OIL command has completed, the "Busy Bi Communication Status Register is cleared, allowing return processing on that comn long as the Busy Bit is set, other OIL commands (other than the Data Highway commands) can execute If the program is in background mode and a second Data interface OIL command is issued before the Busy Bit is cleared, an illegal situation o bit #5 in the Communication Status Register is set

> **NOTE**
> In Background Mode, registers used in the Data Highway command will be changed when the Busy Bit is cleared

ACK Timeout Value This option determines how long the terminal will wait for Data to acknowledge a request command The range is 1 - 6 seconds

Response Timeout Value This option determines how long the terminal will wait foı data before it signals a timeout The range is 1 - 30 seconds

Station Number This number is used to identify the 2000 as a Station on the Data network This option will be a three digit octal address in the range of 000 to 377 the PLC manual for recommended addresses

> **NOTE**
> A different address should be used for each device on the Data Highway

Multiplex Handshaking. This option is used to determine whether multiplex hands enabled Multiplex handshaking is used when a hardware multiplexing box is between the primary serial port of the 2000 and the PLC The multiplex box allow 2000 units to connect to a single PLC Data transmissions from the 2000 to the controlled by the RTS and CTS lines connected between the 2000 and the multiplex 2000 firmware automatically controls the RTS/CTS lines The Quartech multiple been thoroughly tested for use with the 2000-06

A 4 COMMUNICATION STATUS REGISTER

The Communication Status Register contains the status about the message that transferred The Most Significant Byte is the status returned in the message communication module The Least Significant Byte contains information abou transfer status on the workstation end

If the bit noted is set, the condition indicated is true

Status Register

MSB - message status from Interface Module

LSB - Bit 0 = Transfer Error
Bit 1 = Receive Error
Bit 2 = Timeout
Bit 3 = Parity Error
Bit 4 = Busy Bit
Bit 5 = Cannot Execute Now
Bits 6 and 7 = **NOT USED**

A 5 COMMANDS

This section describes the commands that are specific to the 2000-06 Allen-Bradle Highway Interface

G.5 1 GET DATA FROM DATA TABLE (GET)

Syntax

GET dest,addr,lngth,dreg,sreg

where

dest is a 3 digit (octal) station address of a remote PLC

addr is an octal address in a target station from which data is to be read

lngth is the number of words (16-bit values) to read, starting at addr

dreg is the first destination register number to store data

sreg is the Communication Status Register

The command GET gets 16-bit data from a specified area of the data table on the PLC to Section A 4 for the Communication Status Register information

Example

GET 110,300,10,#30,#20

This command will read 10 words of data starting at address 300 (octal) of th device The data returned will be stuffed into registers #30 through #39 Regi: will contain the communication status

A 5 2 **GET DIAGNOSTIC STATUS (GETDIAG)**

Syntax

GETDIAG dest,dreg,sreg

where

dest is a 3 digit (octal) station address of a remote PLC

dreg is the first destination register number to store data

sreg is the Communication status Register

The command GETDIAG reads 10 bytes of diagnostic status from a remote PLC c Station Interface Module Refer to Section A 4 for the Communication Status information

Example

GETDIAG 110,#30,#20

This command will read 10 bytes of status into consecutive 16-bit registers #3(#39 Register #20 will contain the communication status

A 5 3 GET A BIT OF DATA FROM DATA TABLE (GETIO)

Syntax

GETIO dest,addr,bit,dreg,sreg

where

dest is a 3 digit (octal) station address of a remote PLC

addr is an octal address in a target station from which data is to be read

bit is the bit number desired in the word being read (values 0 - 7)

dreg is the first destination register number to store data

sreg is the Communication Status Register

This command reads a bit of data from the device data table Refer to Section A 4 Communication Status Register information

Example

GETIO 110,47,4,#30,#20

This command will read bit 4 at byte address 47 (octal) of the target device Oı register #30 will contain either a TRUE or FALSE value corresponding to the Register #20 will contain the communication status

A 5 4 WRITE DATA TO A DATA TABLE (PUT)

Syntax

PUT dest,addr,lngth,creg,sreg

where

dest is a 3 digit (octal) station address of a remote PLC

addr is an octal address in a target station from which data is to be read

lngth is the number of words (16-bit values) to read, starting at addr

creg is the first source register number to send data from

sreg is the Communication Status Register

The command PUT writes 16-bit data to the data table on the PLC Refer to Section the Communication Status Register information

Example

GET 110,330,110,#30,#20

This command will read 110 words of data to address 300 (octal) of the target The data sent will be copied from registers #30 through #139 Register #20 will the communication status

A 5 5 WRITE A BIT OF DATA TO AN ADDRESS (PUTIO)

Syntax

PUTIO dest,addr,bit,creg,sreg

where

dest is a 3 digit (octal) station address of a remote PLC

addr is an octal address in a target station from which data is to be read

bit is the bit number desired in the word being read (values 0 - 7)

creg is the first source register number to send data from

sreg is the Communication Status Register

This command writes a bit of data to a byte address in the data table Refer to Section A 4 for the Communication Status Register information

Example

PUTIO 110,204,7,#30,#20

This command will read bit 7 at byte address 204 (octal) of the target device The value of register #30 will determine whether a TRUE or FALSE value will be written Register #20 will contain the communication status

A

2000-07

Modicon MODBUS PLC Interface

Printed in the United States of America
Part Number 96760-007

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORI

Revision	Description	Date
A	Appendix Released	6/90

Copyright Information

Address comments concerning this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number: 96760-007A

A 1 INTRODUCTION

This appendix contains information that is specific to the 2000-07 Modicon MODBUS PLC network interface

The specific commands that allow communications are listed and discussed in Section A 5 They are

GETIR
GETIS
GETOR
GETOS
PUTRM
PUTRS

A 2 CABLING THE WORKSTATION TO THE PLC

The PLC and/or network connects to the 9-pin primary serial port of the 2000 Workstation via a cable This is not a standard cable The connectors must be fabricated Figures A-1 and A-2 illustrate the connections that must be made between the primary port and the PLC

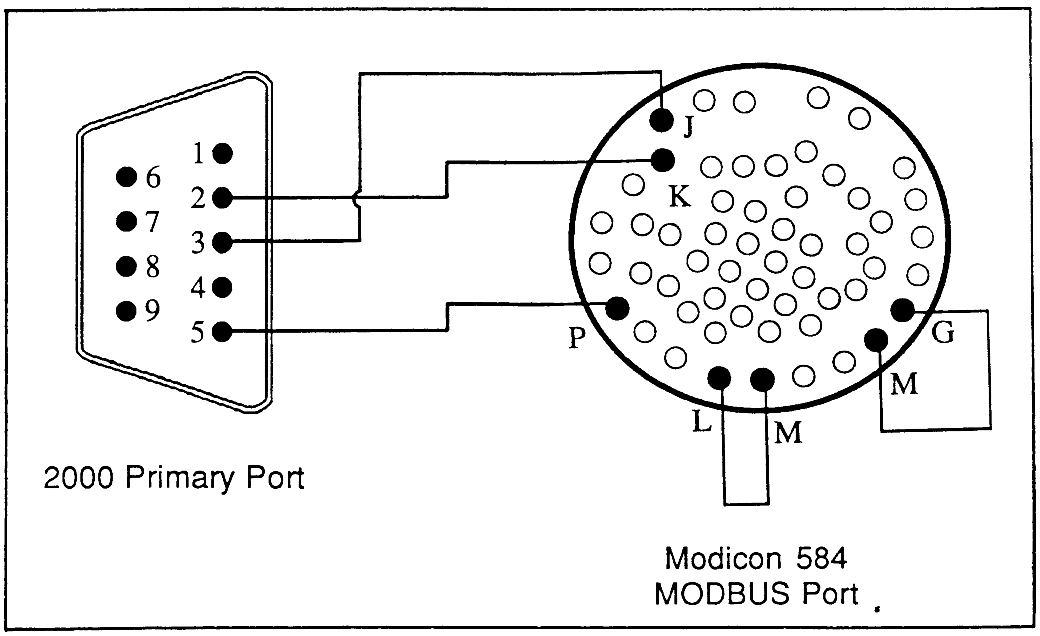


Figure A-1 Connection to Modicon 584

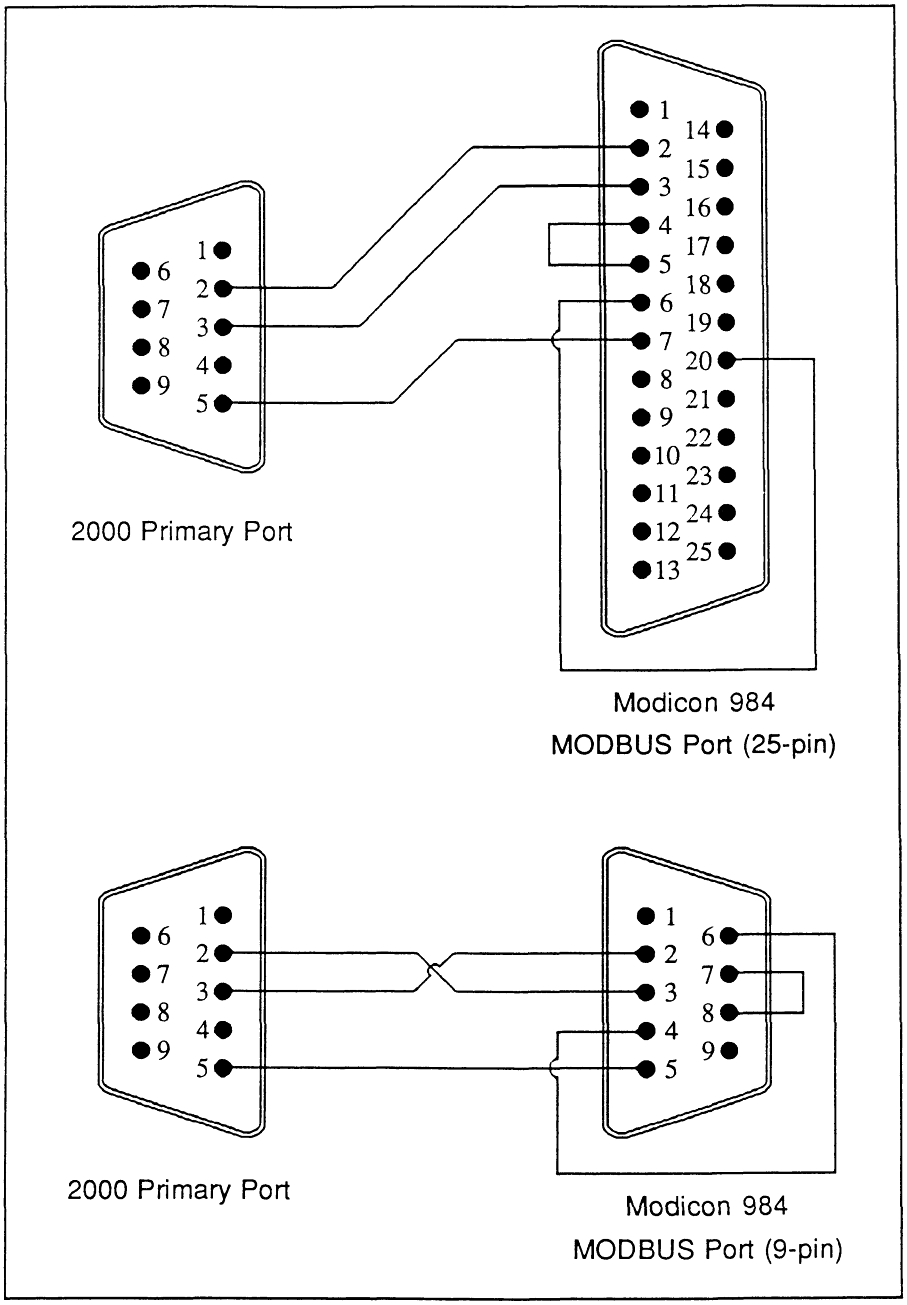


Figure A-2 Connection to Modicon 984

A 3 CONFIGURATION MENU

The configuration menus are called up from the Main Menu The MODBUS Port Configuration Menu looks like the following

```
                -- MODBUS Port Configuration Menu--

6 Baud - 1=300 2=600 3=1200 4=2400 5=4800 6=9600 7=19 2
0 Parity - 0=even 1=odd
0 1=Parity Enabled             0=Disabled
0 1=RTU                        0=ASCII
0 Enable Multiplex Handshaking -- 0=Disable Multiplex Handshaking


Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW>
Use values 0 through 9
"C" for next configuration menu, <ESC> or <ENTER> to quit
```

Figure A-3 Configuration Menu

Baud The baud rate of the channel should be set to match that of the PLC

Parity There are two parity options The first will set the parity used in communication on the MODBUS network to either odd or even The type selected should match the other communication device(s) The second will enable or disable the parity

RTU/ASCII This option selects the way that the workstation will communicate with the MODBUS, either RTU or ASCII

Multiplex Handshaking. This option is used to determine whether multiplex handshaking is enabled Multiplex handshaking is used when a hardware multiplexing box is connected between the primary serial port of the 2000 and the PLC The multiplex box allows multiple 2000 units to connect to a single PLC Data transmissions from the 2000 to the PLC are controlled by the RTS and CTS lines connected between the 2000 and the multiplex box The 2000 firmware automatically controls the RTS/CTS lines The Quartech multiplex box has been thoroughly tested for use with the 2000-07

A 4 COMMUNICATION STATUS REGISTER

The Communication Status Register contains the status about the message that was just transferred The Most Significant Byte is the status returned in the message from the communication module The Least Significant Byte contains information about message transfer status on the terminal end

If the bit noted is set, the condition indicated is true

Status Register

MSB - message status from Interface Module

LSB - Bits 0 - 2 = MODBUS exception codes*
Bit 3 = Transmit Error
Bit 4 = Receive Error
Bit 5 = Parity Error
Bit 6 = Timeout Error
Bit 7 = Not Used

* Refer to the MODBUS user manual for an explanation of the exception codes

A 5 COMMANDS

This section describes the commands that are specific to the 2000-07 Allen-Bradley Data Highway Interface

A 5 1 GET INPUT REGISTER (GETIR) (From MODBUS Slave)

Syntax

GETIR addr,streg,nreg,dreg,sreg

where

addr is an address in the MODBUS slave from which data is to be read (1 to 247)

streg is the starting input register to read It is not necessary to specify input registers as 1xxx since the command assumes the "1" prefix

nreg is the number of registers to read

dreg is the first destination register number to store data

sreg is the Communication Status Register

The command GETIR allows OIL to read the contents of input registers in the PLC This command is analogous to the MODBUS function 04, Read Input Register Refer to Section A 4 for the Communication Status Register information

Example

GETIR 1,9,1,#100,#11

This command will read input register 3009 from slave PLC number 1 The data returned will be stuffed into register #100 Register #11 will contain the communication status

A 5 2 **GET INPUT STATUS (GETIS)** (From MODBUS Slave)

Syntax

GETIS addr,spoint,npoint,dreg,sreg

where

addr is an address in the MODBUS slave from which data is to be read (1 to 247)

spoint is the starting input point to read It is not necessary to specify input registers as 3xxx since the command assumes the "3" prefix

npoint is the number of points to read

dreg is the first destination register number to store data

sreg is the Communication Status Register

The command GETIS allows OIL to read the status of input coils in the PLC The data returned is placed into consecutive data registers low byte first, so that a single byte or an odd byte appears in the low end of that register This command is analogous to the MODBUS function 02, Read Input Status Refer to Section A 4 for the Communication Status Register information

Example

```
TR 2,#40
GETIS #40,197,22,#25,#12
```

This command will read input coils 1197 to 1218 from slave PLC number 2 The data returned will be stuffed into registers #25 and #26 Register #12 will contain the communication status

A 5 3 **GET OUTPUT REGISTER (GETOR)** (From MODBUS Slave)

Syntax

GETOR addr,streg,nreg,dreg,sreg

where

addr is an address in the MODBUS slave from which data is to be read (1 to 247)

streg is the starting output register to be read It is not necessary to specify output registers as 4xxx since the command assumes the "4" prefix

nreg is the number of registers to read

dreg is the first destination register number to store data

sreg is the Communication Status Register

The command GETIR allows OIL to read the contents of output registers in the PLC This command is analogous to the MODBUS function 03, Read Output Register Refer to Section A 4 for the Communication Status Register information

Example

GETOR 2,108,3,#100,#11

This command will read output or holding registers 4108 to 4110 from slave PLC number 2 The data returned will be stuffed into registers #100 through #102 Register 11 will contain the communication status

A 5 4 **GET OUTPUT STATUS (GETOS)** (From MODBUS Slave)

Syntax

GETOS addr,spoint,npoint,dreg,sreg

where

addr is an address in the MODBUS slave from which data is to be read (1 to 247)

spoint is the starting input point to read It is not necessary to specify input registers as 0xxx since the command assumes the "0" prefix

npoint is the number of points to read

dreg is the first destination register number to store data

sreg is the Communication Status Register

The command GETOS allows OIL to read the status of output coils in the PLC The data returned is placed into consecutive data registers low byte first, so that a single byte or an odd byte appears in the low end of that register This command is analogous to the MODBUS function 01, Read Output Status Refer to Section A 4 for the Communication Status Register information

Example

GETOS 2,20,37,#25,#12

This command will read output coils 0020 to 0056 from slave PLC number 2 The data returned will be stuffed into registers #25 through #27 Bit 0 of register #25 contains the status of coil 20, while bit 15 of the register contains the status of coil 35 Bit 0 of register #26 contains the status of coil 36, while bit 15 of the register contains the status of coil 51 Bit 0 of register #27 contains the status of coil 52, while bit 4 of the register contains the status of coil 56 Bits 5-15 of register #27 are 0 Register #12 will contain the communication status

A 5 5 MODIFY MULTIPLE REGISTERS (PUTRM) (To MODBUS Slave)

Syntax

PUTRM addr,streg,nreg,dreg,sreg

where

addr is an address in the MODBUS slave from which data is to be read (1 to 247)

streg is the starting output register to be modified It is not necessary to specify output registers as 4xxx since the command assumes the "4" prefix

nreg is the number of registers to modify

dreg is the first destination register number to store data

sreg is the Communication Status Register

The command PUTRM allows OIL to modify the contents of a series of output or holding registers in the PLC This command is analogous to the MODBUS function 16, Preset Multiple Registers Refer to Section A 4 for the Communication Status Register information

Example

```
TR 100,#20
TR 101,#21
TR 102,#22
PUTRM 2,136,3,#20,#12
```

This command will modify the contents of holding registers 4136, 4137, and 4138 in slave PLC number 2 to 100, 101, and 102, respectively Register #12 will contain the communication status

A 5 6 **MODIFY A SINGLE REGISTER (PUTRS)** (To MODBUS Slave)

Syntax

PUTRS addr,reg,data,sreg

where

addr is an address in the MODBUS slave from which data is to be read (1 to 247)

reg is the output register to be modified It is not necessary to specify the output register as 4xxx since the command assumes the "4" prefix

data is the value to place into the register

sreg is the Communication Status Register

The command PUTRS allows OIL to modify the contents of a single output or holding register in the PLC This command is analogous to the MODBUS function 06, Preset Single Register Refer to Section A 4 for the Communication Status Register information

Example

PUTRS 2,136,926,#12

This command will modify the contents of holding register 4136 in slave PLC number 2 to 926 Register #12 will contain the communication status

2000-08

Texas Instruments Series 500/505 PLC Interface

Printed in the United States of America
Part Number 96760-008

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORD

Revision	*Description*	*Date*
A	Appendix Released	1/91

Address comments concerning this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number ***96760-008A***

A 1 INTRODUCTION

This appendix contains the information specific to the 2000-08 Texas Instruments Series 500/505 PLC interface The protocol used by the module to communicate with the Texas Instruments PLC is the Non-Intelligent Data Link Protocol using Series 500/505 Communication Task Codes

The specific commands that allow communications are described in Section A 6 and listed below for reference

CTOF	Characters to floating point
FPD	Floating point display
FTOI	Floating point to integer
GET	Get PLC data
GETIO	Get PLC I/O point
GETMIO	Get multiple PLC data
GETR	Get PLC register block
GETS	Get PLC S memory data
ITOF	Integer to floating point
PUT	Put PLC data
PUTIO	Put PLC I/O point
PUTR	Put PLC register block
PUTS	Put PLC S memory data

A 2 CABLING THE TERMINAL TO THE PLC

The PLC and/or network connects to the 9-pin primary serial port of the 2000 Workstation via a non-standard cable The connectors to this cable must be fabricated If using RS-422, you must first set the jumpers to RS-422 by positioning jumpers J1 to J8 to A Next, make the connections as shown in Figure A-1 below

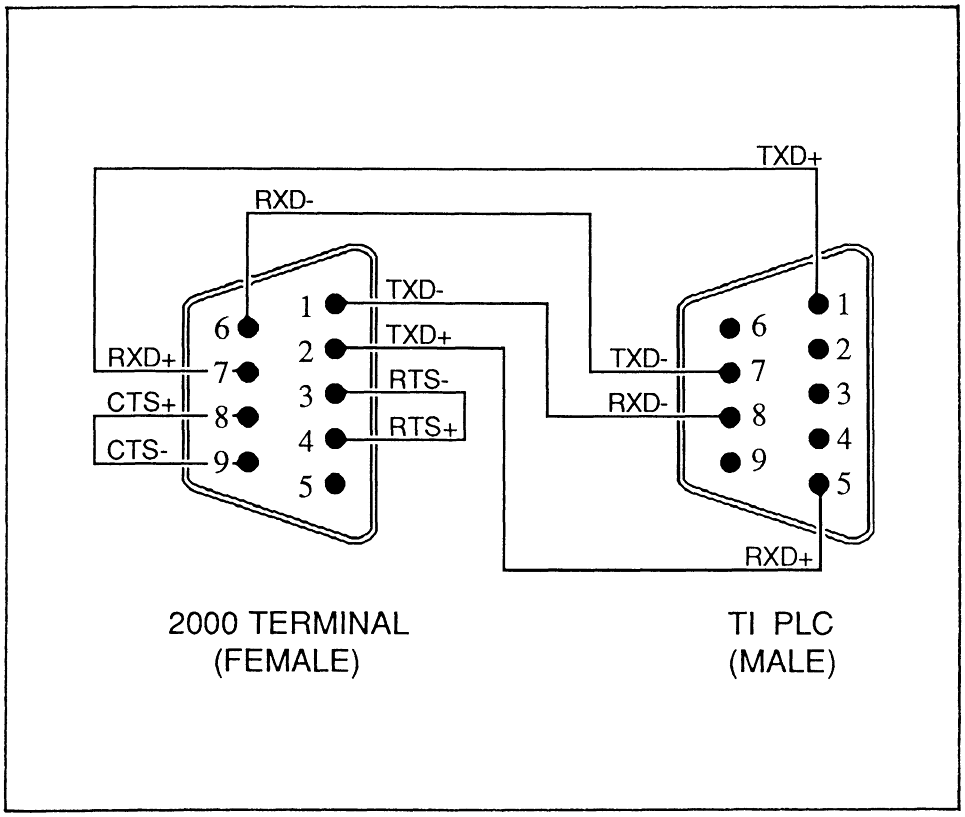


Figure A-1 Connecting to a 9-pin TI PLC via RS-422

If using RS-232C, set jumpers J1 through J8 to B, and make the connections for the appı PLC (9-pin or 25-pin) as shown in Figures A-2 and A-3 below

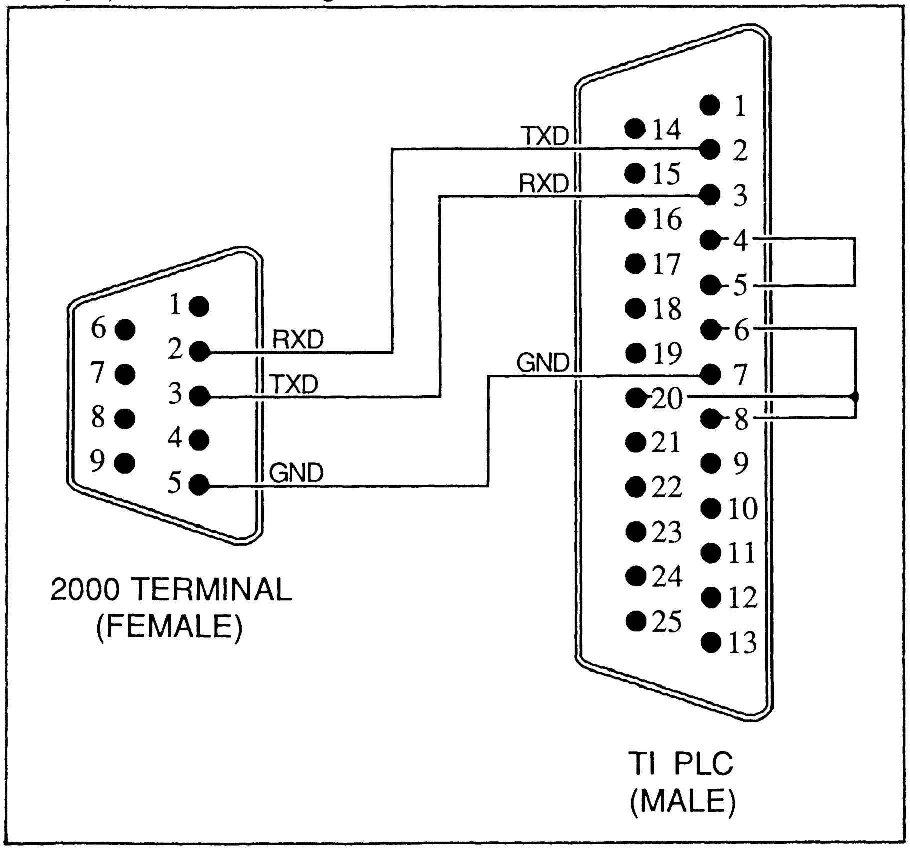


Figure A-2 Connecting to a 25-pin TI PLC via RS-232C

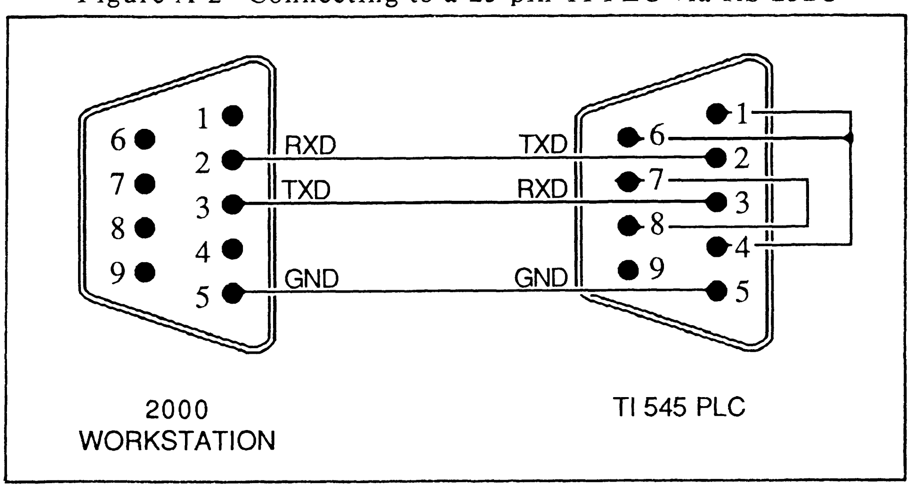


Figure A-3 Connecting to a 9-pin TI PLC via RS-232C

A 3 CONFIGURATION MENU

The configuration menus are called up from the Main Menu (discussed in Chapter 3) The Texas Instruments PLC Port Menu is shown if Figure A-4 below

```
-- Texas Instruments PLC Port (P) Configuration Menu --

Baud - 1=300 2=600 3=1200 4=2400 5=4800 6=9600 7=19 2K
1 = Parity Enabled                   0 = Parity Disabled
1 = Odd Parity                       0 = Even Parity
1 = 8 Data Bits                      0 = 7 Data Bits
1 = 2 Stop Bits                      0 = 1 Stop Bit
1 = Enable Multiplex Handshaking     0 = Disable Multiplex Handshaking

Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW>
Use values 0 through 9
"C" for next configuration menu, <RET> or <ENTER> to quit
```

Figure A-4 Configuration Menu

NOTE

Certain combinations of parity, baud rate, and stop bits are illegal If selected, they will be changed to the legal parameters shown below

Illegal parameters	**Legal parameters**
parity/8 data bits/2 stop bits	same, but with 1 stop bit
no parity/7 data bits/1 stop bit	same, but with 2 stop bits

Baud The baud rate of the channel should be set to match that of the PLC

Parity There are two parity options The first sets the parity used in communication with the PLC to either odd or even The type selected should match the other communication device(s) The second enables or disables the parity

Data Bits This sets the data bits used in communications with the PLC to either seven or eight The number selected should match the other communication device(s)

Stop Bits This sets the stop bits used in communications with the PLC to either one or two The number selected should match the other communication device(s)

Multiplex Handshaking This option enables or disables multiplex handshaking Multiplex handshaking is used when a hardware multiplexing box is connected between the primary serial port of the 2000 and the PLC The multiplex box allows multiple 2000 units to connect to a single PLC Data transmissions from the 2000 to the PLC are controlled by the RTS and CTS lines connected between the 2000 and the multiplex box The 2000 firmware automatically controls the RTS/CTS lines when this option is enabled

> **NOTE**
> If using multiplex handshaking, the CTS and RTS signals must be connected as specified by your particular hardware

A 4 DIAGNOSTIC MENU

The Diagnostics Menu is accessed from the Main Menu (see Chapter 3) The Diagnostics Menu specific to the 2000-08 is shown in Figure A-5 below

```
-- Diagnostics --

1) RAM
2) ROM Checksum
3) RS-232 Serial Loopback
4) RS-485/Multidrop Serial Loopback
5) Printer Port Test
6) Battery Test

<ESC> or <ENTER> to quit
```

Figure A-5 Diagnostics Menu

See Chapter 3 for an explanation of each of the menu choices

A 5 COMMUNICATION STATUS REGISTER

The communication status register contains status about the message that was just transferred The upper byte contains the error status returned When an error is indicated, the lower byte will contain a value indicating the specific error type A zero value in the entire register indicates a successful transfer

If the bit noted in the upper byte is set, the condition indicated by the lower byte value is true

Status Register

Bit 15 = Timeout
Bit 14 = Receive Error
Bit 13 = Transfer Error *
Bit 12 = Task Error **

* The Transfer Error Code values are

01 = character parity framing or overrun error
02 = non-HEX/ASCII character received
03 = invalid character count field
04 = character count error
05 = checksum error

** The Task Error Code values are

02 = address out of range
03 = requested data not found
04 = illegal task code request
05 = request exceeds program memory size
07 = fatal error detected
08 = keylock protect error
09 = incorrect amount of data sent with request
0A = illegal request in current PLC mode
0D = odd number of ASCII characters received
0E = illegal write to non-volatile program memory
0F = data not inserted
10 = invalid data sent with command
11 = invalid operation in current PLC mode

A 6 COMMANDS

This section describes the commands specific to the 2000-08 Texas Instruments Series 500/505 interface

A 6 1 CHARACTERS TO FLOATING POINT (CTOF)

Syntax

CTOF d,r

where

d is the first of a string of 11 registers

r is the first of a pair of registers

This command converts a string of ASCII characters (found in a sequence of registers) to floating point values

The parameter d is the first in a series of registers whose ASCII contents will be converted to IEEE single precision floating point number This string is up to 11 characters (or registers) long, with one ASCII character per register Unless the full 10 registers are used, the string is terminated by a NULL character

No more than 5 digits to the left of the decimal point and no more than 4 digits to the right of the decimal point are accepted Any non-numeric characters in the string cause an invalid parameter error and stop execution

The result of this command is an IEEE single precision floating point number that is returned in the register pair defined by the second parameter This returned value follows the same conventions as the other floating point numbers in this document

Example

CTOF #50,#21

This command converts a string of characters starting in register #50 (and possibly proceeding through register #61) into a floating point value in registers #21 and #22

A 6 2 FLOATING POINT DISPLAY (FPD)

Syntax

FPD r,j,m,f

where

r   is the first of a pair of registers

j   is justification, 0 = right, 1 = left

m   is the total display width (maximum=10, minimum=2)

f   is the fractional field width with a maximum of 4

This command displays an IEEE single precision number (found in two consecutive OIL registers) on the screen

The value to be displayed is the first parameter in the register pair This parameter is a standard OIL register, but cannot be the last available register, e g , #500 out of 500

This command supports both left and right justification If right justification is selected, enough space is inserted between characters so that the total number of characters output is exactly equal to the next parameter, m If left justification is selected, the function outputs no extra characters

The total field width (m) is the maximum number of characters that will be output by this command This parameter has a maximum value of ten and minimum of two This means that values greater than 99999 9999 cannot be displayed and that at least one digit left of the decimal point and the decimal point itself are always displayed If the function detects an output overflow (i e significant digits are lost left of the decimal point), an additional character, ">", is appended to the output string If the value output is negative, a negative sign, "-", precedes the left-most digit and is counted as part of the field width m

The fractional field width (f) is the number of characters to the right of the decimal point This number must be at least two less than the total display field width, and must also be less than or equal to four

Example

FPD #21,1,7,1

This command displays the floating point value in registers #21 and #22 as a left justified 7-character string

A 6 3 FLOATING POINT TO INTEGER (FTOI)

Syntax

FTOI r,i,f,s

where

- r is the first of a pair of registers
- i is the register to receive the 16-bit integer
- f is the register to receive the fractional portion
- s is the status flag register

The FTOI command converts an IEEE single precision floating point number into an integer format that is more easily manipulated by OIL programs This command requires the IEEE single precision number to be found in two consecutive OIL registers The value to be converted is represented by the first parameter (r) This parameter is a standard OIL register, except that it cannot be the last available register, e g #500 of 500

The value returned is a 16-bit integer This integer represents the least significant 16-bits of the absolute value of the integer portion of the IEEE single precision number The sign of the original number is indicated in the status flag word

The fractional portion is expressed as an integer which represents the number of 10000ths in the original floating point value For example, if an integer value of 1250 is returned, the fractional value would be 1250/10000 or 1/8 (125)

The status flag register has the following values

Bit 0
0 = the original number was positive
1 = the original number was negative

Bit 1
0 = the integer value returned is correct
1 = an overflow occurred, indicating the floating point value was greater than 65535

Example

FTOI #20,#30,#40,#41

This example converts the IEEE single precision floating point number in registers 20 and 21 to an integer, which is stored in register 30 Any fractional value of the integer is stored in register 40, and the status flag is stored in register 41

A 6 4 **GET PLC DATA (GET)**

Syntax

GET device,num,dreg,sreg

where

device is the device type
- 0 - input memory (WX)
- 1 - output memory (WY)
- 2 - variable memory (V)
- 3 - constant memory (K)
- 4 - drum preset counter (DPC)
- 5 - drum timer current count (DTC)
- 6 - drum timer base (DTB)
- 7 - timer/counter preset count (TCP)
- 8 - timer/counter count count (TCC)

num is the control device number or address

dreg is the destination register number to store data

sreg is the communication status register

The command GET reads a single 16-bit word, register, or control device parameter from a PLC and places the result in a specified register Refer to Section A 5 for the communication status register information

All addresses start at 1, and increment by 1 for each additional element The only unusual address form is for drum preset counters (DPCs), which require both a drum number and step values This is gained by multiplying the drum number by 256 and adding the result to the drum step For example, for drum number 1, step 2 would be calculated as (1 x 256) + 2 = 258

Example

GET 2,100,#24,#23

This command reads variable (V) memory location 100 The data returned is stored in register 24 Register 23 contains the communication status

A 6 5 **GET PLC I/O POINT (GETIO)**

Syntax

GETIO type,addr,dreg,sreg

where

type is the point type
0 - discrete IR input (X)
1 - discrete IR output (Y)
2 - discrete control point (C)

addr is the discrete address

dreg is the destination register number to store data

sreg is the communication status register

The command GET reads a discrete I/O point A "0" is returned if it is off, while a "1" is returned if on Refer to Section A 5 for the communication status register information

Example

GETIO 2,16,#24,#23

This command reads control point 16 The value returned is stored in register 24 Register 23 contains the communication status

A 6 6 GET PLC MULTIPLE I/O POINTS (GETMIO)

Syntax.

GETMIO type,spoint,npoints,dreg,sreg

where

type	is the point type 0 - Discrete IR Input (X) 1 - Discrete IR Output (Y) 2 - Input memory (WX) 3 - Output memory (WY)
spoint	is the starting control device number or address
npoints	is the number of points or words to return
dreg	is the starting destination terminal data register
sreg	is the communication status register

Discrete IR Input (X) / IR Output (Y)

The GETMIO command reads each I/O point in the specified range and returns a 0 if the I/O point is off or a 1 if the point is on The individual data I/O bits are combined into a register starting with the specified terminal data register in the GETMIO command The data I/O bit of each discrete point is placed into a register starting at the register's least significant bit (bit 0) Unfilled bits of the register are left at 0 The starting I/O point's on/off status is returned in bit 0 of the destination terminal data register The next point's on/off status is returned in bit 1, etc If more than 16 data I/O points are requested, the first 16 points are placed in the starting destination terminal data register The next n points are returned in the starting terminal data register plus n The discrete I/O point's status is returned starting in the specified destination terminal data register

Input Memory (WX) / Output Memory (WY)

The GETMIO command reads I/O memory in the specified range and returns each I/O memory register in a terminal data register The starting I/O memory register's data is returned in the first specified destination terminal data register The next I/O memory point's data is returned in the starting terminal data register plus n The maximum input and output amounts are limited only by the number of registers available in the 2000 Workstation

Example

GETMIO 2,1,10,#100,#20

This command reads input memory (WX) 1 to 10 The data returned is stored in registers #100 to #109

A 6 7 GET PLC REGISTER BLOCK (GETR)

Syntax

GETR addr,num,dreg,sreg

where

addr	is the beginning variable number or address
num	is the number of variables to read (maximum = 128)
dreg	is the destination register number to store data
sreg	is the communication status register

The command GET reads a group of 16-bit variables from a PLC and places the result in a specified register A maximum of 128 registers may be transferred in a single command Refer to Section A 5 for the communication status register information

Example

GETR 100,6,#24,#23

This command reads PLC variable (V) memory locations 100 through 106 The data returned is stored in registers 24 through 29 Register 23 contains the communication status

A 6 8 GET PLC S MEMORY DATA (GETS)

Syntax.

GETS d,l,n,dreg,sreg

Where

d	is the S memory type 0 - analog alarm data 1 - loop data
l	is the analog alarm or loop number

> **NOTE**
> A special case exists where the analog alarm or loop number (l) is defined as zero in order to search for any unacknowledged alarms If l is set to zero, n can either be zero or an analog alarm or loop number at which to start searching

n	is the analog alarm or loop data point type number (see Tables A-1 and A-2 on the following pages for definitions)
dreg	is the destination terminal data register
sreg	is the communication status register

The GETS command reads a 16 bit integer or 32 bit IEEE real number from the PLC and places it in the specified terminal data register

Loop numbers range from 0 to 64 while analog alarm numbers range from 0 to 128

Real numbers are returned in the register number specified and in the register number specified plus 1

Example

GETS 1,1,1,#100,#20

This command reads the loop gain for loop 1 The data is then stored in register 100 The communication status is stored in register 20

Table A-1 GETS - Analog Alarm Type Variables

Number		Meaning
0		Start search for unacknowledged alarms at first data point in analog alarm number
1	AHA	Analog Alarm High Alarm Limit
2	ALA	Analog Alarm Low Alarm Limit
3	APV	Analog Alarm Process Variable
4	APVH	Analog Alarm Process Variable High Limit
5	APVL	Analog Alarm Process Variable Low Limit
6	AODA	Analog Alarm Orange Deviation Alarm Limit
7	AYDA	Analog Alarm Yellow Deviation Alarm limit
8	ATS	Analog Alarm Sample Rate
9	ASP	Analog Alarm Setpoint
10	AVF	Analog Alarm V Flags (integer)
11	ACF	Analog Alarm C-Flags (32 bit integer)
12	AERR	Analog Alarm Error
13	AHHA	Analog Alarm High-High limit
14	ALLA	Analog Alarm Low-Low Limit
15	ARCA	Analog Alarm Rate-of-Change Alarm Limit
16	ASPH	Analog Alarm Setpoint High Limit
17	ASPL	Analog Alarm Setpoint Low Limit
18	AADB	Analog Alarm Deadband
19	ACFH	Analog Alarm C-Flags Most Significant (16 bit integer)
20	ACFL	Analog Alarm C-Flags Least Significant (16 bit integer)
21	AACK	Analog Alarm Alarm/Alarm acknowledge (integer)

Table A-2 GETS - Loop Data Point Variables

Number		Meaning
0		Start search for unacknowledged alarm at first data point in loop
1	LKC	Loop Gain
2	LTI	Loop Reset
3	LTD	Loop Rate (minutes)
4	LHA	Loop High Alarm Limit
5	LLA	Loop Low Alarm Limit
6	LPV	Loop Process variable
7	LPVH	Loop Process Variable High Limit
8	LPVL	Loop Process Variable Low Limit
9	LODA	Loop Orange Deviation Alarm Limit
10	LYDA	Loop Yellow Deviation Alarm Limit
11	LTS	Loop Sample Rate (in seconds)
12	LSP	Loop Setpoint
13	LMN	Loop Output
14	LVF	Loop V-Flags (integer)
15	LCF	Loop C-Flags (32 bit integer)
16	LRSF	Loop RAMP/SOAK Flags (integer)
17	LERR	Loop Error
18	LMX	Loop Bias
19	LHHA	Loop High High Alarm Limit
20	LLLA	Loop Low Low Alarm Limit
21	LRCA	Loop Rate-Of-Change Alarm Limit
22	LSPH	Loop Setpoint High Limit
23	LSPL	Loop Setpoint Low Limit
24	LADB	Loop Alarm Deadband
25	LPET	Loop Peak Elapsed Time Value
26	LRSN	Loop RAMP/SOAK Step number
27	LCFH	Loop C-Flags Most Significant (16 bit integer)
28	LCFL	Loop C-Flags Least Significant (16 bit integer)
29	LKD	Loop Derivative Gain-Limiting Coefficient
30	LACK	Loop Alarm/Alarm Acknowledge Flags (integer)

A 6 9 INTEGER TO FLOATING POINT (ITOF)

Syntax

ITOF i,f,r

where

i is the integer portion of the number to be converted

f is the fractional portion

r is the first of a pair of registers

The ITOF command converts an integer to an IEEE floating point value The integer portion is passed to a single OIL register The number cannot be greater than 65535 because the integer conversion is limited to a 16-bit value

The fractional portion of this number is expressed as an integer representing the number of 10000ths For example, the fractional value 1/4 is represented by setting the f register to 2500, to represent 2500/10000 Any value greater than 9999 is considered to be an error

The IEEE single precision floating point number that results from this conversion is placed in the register pair represented by r

Example

ITOF #31,#32,#21

This command converts the integer in register #31 and the offset fraction in register #32 into a floating point value The value is stored in registers #21 and #22

A 6 6 **PUT PLC DATA (PUT)**

Syntax

PUT device,num,srcreg,sreg

where

device — is the device type
- 0 - input memory (WX)
- 1 - not allowed in PUT
- 2 - variable memory (V)
- 3 - constant memory (K)
- 4 - drum preset counter (DPC)
- 5 - drum timer current count (DTC)
- 6 - drum timer base (DTB)
- 7 - timer/counter preset count (TCP)
- 8 - timer/counter count count (TCC)

num — is the control device number or address

srcreg — is the source register number

sreg — is the communication status register

The command PUT writes a single 16-bit terminal register to a specific PLC memory location, register, or controller device parameter Refer to Section A 5 for the communication status register information

All addresses start at 1, and increment by 1 for each additional element The only unusual address form is for drum preset counters (DPCs), which require both a drum number and step values This is gained by multiplying the drum number by 256 and adding the result to the drum step For example, for drum number 1, step 2 would be calculated as (1 x 256) + 2 = 258

Example

PUT 2,100,#24,#23

This command writes the contents of register 24 out to PLC variable (V) memory location 100 Register 23 contains the communication status

A 6 7 PUT PLC I/O POINT (PUTIO)

Syntax

PUTIO type,addr,srcreg,sreg

where

type	is the point type 0 - not allowed in PUTIO 1 - not allowed in PUTIO 2 - discrete control point (C)
addr	is the discrete address
srcreg	is the source register number (must contain a 0 or a 1)
sreg	is the communication status register

The command PUT sets a discrete control point to either a "0" or "1" Refer to Section A 5 for the communication status register information

Example

PUTIO 2,12,#22,#23

This command writes the contents of OIL register 22 (must be 1 or 0), to control point 12 Register 23 contains the communication status

A 6 8 PUT PLC REGISTER BLOCK (PUTR)

Syntax

PUTR addr,num,srcreg,sreg

where

addr	is the beginning variable number or address
num	is the number of variables to write
srcreg	is the source register number
sreg	is the communication status register

The command PUTR writes a group of 16-bit variables from the OIL register(s) to a block of PLC variable (V) memory A maximum of 128 registers can be transferred in a single command Refer to Section A 5 for the communication status register information

Example

PUTR 100,6,#24,#23

This command writes the data from OIL registers 24 through 29 to PLC variable (V) memory locations 100 through 105 Register 23 contains the communication status

A 69 PUT PLC S MEMORY DATA (PUTS)

Syntax.

PUTS d,l,n,srcreg,sreg

Where

d	is the S memory type 0 - analog alarm data 1 - loop data
l	is the analog alarm or loop number
n	is the analog alarm or loop data number (see Tables A-3 and A-4 on the following pages)
srcreg	is the destination terminal data register
sreg	is the communications status register

The PUTS command writes a 16 bit integer or 32 bit IEEE real number from the PLC and places it in the specified terminal data register

Loop numbers range from 0 to 64 while analog alarm numbers range from 0 to 128

> **NOTE**
> If a real number is written it is written from the register number specified and from the register number specified plus 1

Example

PUTS 1,1,1,#100,#20

This command writes the loop gain for loop 1 and stores the data in register 100 The communication status is stored in register 20

Table A-3 PUTS - Loop Data Variables

Number		Meaning
1	LKC	Loop Gain
2	LTI	Loop Reset
3	LTD	Loop Rate (minutes)
4	LHA	Loop High Alarm Limit
5	LLA	Loop Low Alarm Limit
6	LPV	Loop Process variable
7	LPVH	Loop Process Variable High Limit
8	LPVL	Loop Process Variable Low Limit
9	LODA	Loop Orange Deviation Alarm Limit
10	LYDA	Loop Yellow Deviation Alarm Limit
11	LTS	Loop Sample Rate (in seconds)
12	LSP	Loop Setpoint
13	LMN	Loop Output
14	LVF	Loop V-Flags (integer)
15	LCF	Loop C-Flags (32 bit integer)
16	LRSF	Loop RAMP/SOAK Flags (integer)
17	LERR	Loop Error
18	LMX	Loop Bias
19	LHHA	Loop High High Alarm Limit
20	LLLA	Loop Low Low Alarm Limit
21	LRCA	Loop Rate-Of-Change Alarm Limit
22	LSPH	Loop Setpoint High Limit
23	LSPL	Loop Setpoint Low Limit
24	LADB	Loop Alarm Deadband
25	LPET	Loop Peak Elapsed Time Value
26	LRSN	Loop RAMP/SOAK Step number
27	LCFH	Loop C-Flags Most Significant (16 bit integer)
28	LCFL	Loop C-Flags Least Significant (16 bit integer)
29	LKD	Loop Derivative Gain-Limiting Coefficient
30	LACK	Loop Alarm/Alarm Acknowledge Flags (integer)

Table A-4 PUTS - Analog Alarm Variables

Number		Meaning
1	AHA	Analog Alarm High Alarm Limit
2	ALA	Analog Alarm Low Alarm Limit
3	APV	Analog Alarm Process Variable
4	APVH	Analog Alarm Process Variable High Limit
5	APVL	Analog Alarm Process Variable Low Limit
6	AODA	Analog Alarm Orange Deviation Alarm Limit
7	AYDA	Analog Alarm Yellow Deviation Alarm limit
8	ATS	Analog Alarm Sample Rate
9	ASP	Analog Alarm Setpoint
10	AVF	Analog Alarm V Flags (16 bit integer)
11	ACF	Analog Alarm C-Flags (32 bit integer)
12	AERR	Analog Alarm Error
13	AHHA	Analog Alarm High-High limit
14	ALLA	Analog Alarm Low-Low Limit
15	ARCA	Analog Alarm Rate-of-Change Alarm Limit
16	ASPH	Analog Alarm Setpoint High Limit
17	ASPL	Analog Alarm Setpoint Low Limit
18	AADB	Analog Alarm Deadband
19	ACFH	Analog Alarm C-Flags Most Significant (16 bit integer)
20	ACFL	Analog Alarm C-Flags Least Significant (16 bit integer)
21	AACK	Analog Alarm Alarm/Alarm Acknowledge (16 bit integer)

2000-10

Square D SY/MAX PLC Interface

Printed in the United States of America
Part Number 96760-010

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORD

Revision	*Description*	*Date*
A	Appendix Released	11/90

Address comments concerning
this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number: ***96760-010A***

A 1 INTRODUCTION

This chapter contains the information specific to the 2000-10 Square-D SY/MAX Network interface The protocol that the module uses to communicate with the Square-D SY/MAX Network is a full-duplex asynchronous point-to-point protocol that closely conforms to the ANSI X3 28 standard

The specific commands that allow communications are described Section A 5 and listed below for reference

GET
GETN
GETNN
PUT
PUTN
PUTNN

> **NOTE**
> Before operating this direct connect, configure the primary port for RS-422 under Configuration in the Main Menu

A 2 CABLING THE TERMINAL TO THE PLC

The PLC and/or network connects to the 9-pin primary serial port of the 2000 Workstation via a non-standard cable The connectors on this cable must be fabricated Figure A-1 illustrates the connections that must be made between the primary serial port and either the PLC NIM module or the comm port of the processor

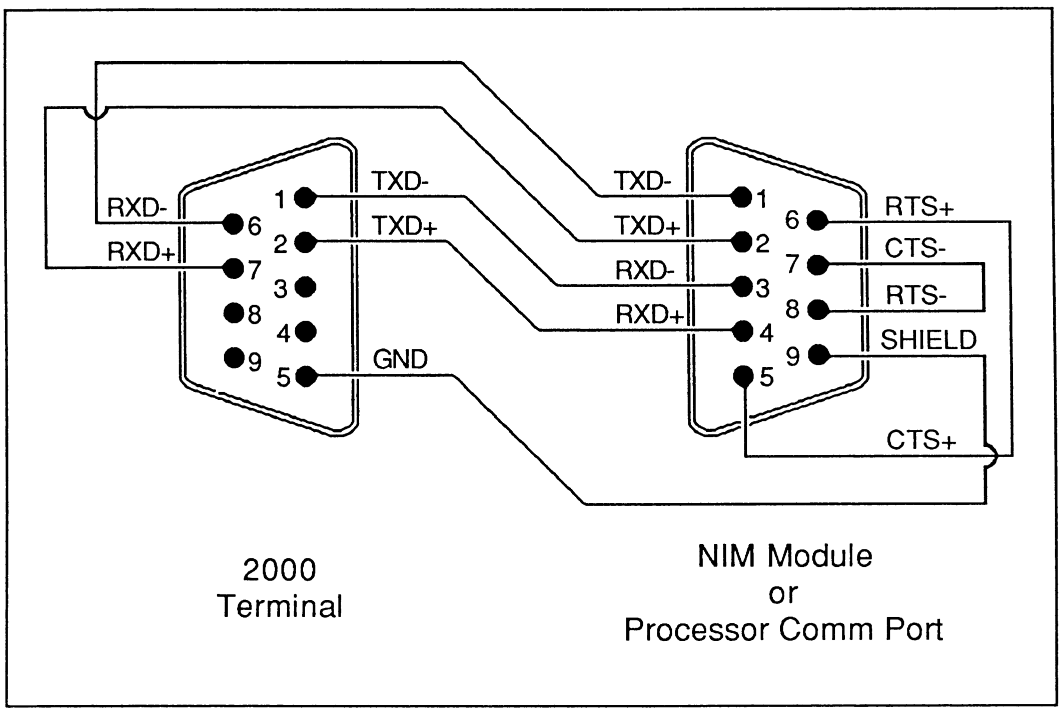


Figure A-1 Connection to the NIM or Processor

A 3 CONFIGURATION MENU

The configuration menus are called up from the Main Menu (discussed in Chapter 3) The SY/MAX Menu is shown below

```
              -- Square D Port Configuration Menu (P) --

6    Baud - 1=300 2=600 3=1200 4=2400 5=4800 6=9600
001  Network Interface Module # (000-199)
01   Response Timeout Value (1-16 seconds)
0    1=Enable Multiplex Handshaking 0=Disable Multiplex Handshaking


Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW>
Use values 0 through 9
"C" for next configuration menu, <RET> or <ENTER> to quit
```

Figure A-2 Configuration Menu

Baud The baud rate of the channel should be set to match that of the PLC

Network Interface Module # If the terminal is set up in a network configuration, one or more NIMs are used The NIM number setting in the Square-D configuration **MUST** be identical to the network device number The network device number is determined by two thumbwheel switches on the NIM COM port The first number (**X**___) is the number of the COM port being accessed (0 or 1) The second and third (_**XX**) are the thumbwheel settings (0 to 9 each)

Response Timeout Value The timeout configuration option determines how long the terminal will wait for a response before it signals a timeout The value entered is in seconds, with a range of 1 to A

Multiplex Handshaking This option enables or disables multiplex handshaking Multiplex handshaking is used when a hardware multiplexing box is connected between the primary serial port of the 2000 and the PLC The multiplex box allows multiple 2000 units to connect to a single PLC Data transmissions from the 2000 to the PLC are controlled by the RTS and CTS lines connected between the 2000 and the multiplex box The 2000 firmware automatically controls the RTS/CTS lines The Quartech multiplex box has been thoroughly tested for use with the 2000-10

A 4 COMMUNICATION STATUS REGISTER

The communication status register in the OIL command contains the status of the message previously transferred The most significant byte (MSB) is the status returned in the message from the communication module The least significant byte (LSB) contains internal XYCOM information about the transfer status

If a bit is set in the status register, the condition corresponding to that bit described below is true

Status Register

MSB Message status from the SY/MAX module

Bit 15	No Errors
Bit 14	Responding processor in RUN mode
Bit 13	Destination Network Module not found
Bit 12	See PUT command below

Bits 8-11 are different for GET and PUT commands

<u>GET command</u>

Bit 8	Error
Bits 8,10	Illegal Operation - an attempt was made to read from a device with too many registers
Bits 8,10,11	Receiver Overflow - Too many registers were sent or requested

<u>PUT command</u>

Bit 8	Error
Bits 8,9	Illegal Address - An attempt was made to write to an address that does not exist
Bits 8,10	Illegal Operation - An attempt was made to write to a device with no registers
Bits 8,11	Write to a Read Only Register
Bits 8,12	Receiver Overflow - Too many registers were sent to a remote processor

LSB Status from XYCOM Terminal

Bit 0	Transmit Error
Bit 1	Receive Error
Bit 2	Timeout
Bit 3	Parity Error
Bits 4-7	Not Used

A 5 COMMANDS

This section describes the OIL commands specific to the 2000-10 Square D interface

A 5 1 GET REGISTER FROM DESTINATION (GET)

Syntax.

    GET addr,lngth,dreg,sreg

where

addr	is an address in SY/MAX from which data is to be read
lngth	is the number of registers (16-bit values) to read, starting at addr
dreg	is the starting register number to store received data
sreg	is the communication status register

The command GET gets 16-bit data when connected directly to a PLC Refer to Section A 4 for communication register error bit responses

Example.

    GET 300,10,#30,#20

This command reads 10 registers starting at register 300 of the target device The data returned is stored in registers 30 through 39 Register 20 contains the communication status

A 52 GET A REGISTER FROM DESTINATION NETWORK DEVICE (GETN)

Syntax.

GETN dest,addr,lngth,dreg,sreg

where

dest	is the address of the network device
addr	is the decimal address in the target device from which data is to be read
lngth	is the number of registers (16 bit values) to read, starting at addr
dreg	is the starting register number to store received data into
sreg	is the communication status register

GETN reads data from a PLC via a Network Interface Module (NIM) Refer to Section A 4 for communication status register error bit responses

Example.

GETN 15,20,10,#30,#20

This command reads 10 registers from the destination SY/MAX device at network address 15 The data is read starting with register 20 of the target device The data returned is stored in registers 30 through 39 Register 20 contains the communication status

A 5 3 GET A REGISTER USING NET TO NET (GETNN)

Syntax.

GETNN dest,netn,addr,lngth,dreg,sreg

where

dest	is the address of the network device from which to GET data
netn	is the NET-to-NET routing address
addr	is the decimal address in the target device from which data is to be read
lngth	is the number of registers (16-bit values) to read, starting at addr
dreg	is the starting register number to store data into
sreg	is the communication status register

This command reads data from a PLC located on another network Refer to Section A 4 for communication status register error bit responses

Example.

GETNN 15,10,20,100,#30,#20

This command reads 100 registers from the destination SY/MAX device following NET-to-NET address 10 at destination address 15 The data is read starting with register 20 of the target device The data returned is stored in registers 30 through 129 Register 20 contains the communication status

A 5 4 **WRITE DATA TO A REGISTER (PUT)**

Syntax.

PUT addr,lngth,creg,sreg

where

addr	is a register address in SY/MAX into which data is to be written
lngth	is the number of registers (16-bit values) to write, starting at addr
creg	is the starting register number to send data from
sreg	is the communication status register

The PUT command writes 16-bit data to a register in the destination device when connected directly to the PLC Refer to Section A 4 for communication status register error bit responses

Example.

PUT 300,10,#30,#20

This command writes 10 registers starting at register 300 of the target device The data sent is from registers 30 through 39 Register 20 contains the communication status

A 5 5 WRITE A REGISTER TO THE DESTINATION DEVICE (PUTN)

Syntax.

PUTN dest,addr,lngth,creg,sreg

where

dest	is the address of the network device to PUT data into
addr	is the decimal address in the target device into which data is to be written
lngth	is the number of registers (16 bit values) to write, starting at addr
creg	is the starting register number to send data from
sreg	is the communication status register

PUTN writes data to a PLC via the NIM Refer to Section A 4 for communication status register error bit responses

Example.

PUTN 15,20,10,#30,#20

This command writes to 10 registers in the destination SY/MAX device starting at register 20 The target device is at network address 15 The data is written from registers 30 through 39 Register 20 contains the communication status

A 5 6 **WRITE A REGISTER USING NET TO NET (PUTNN)**

<u>Syntax.</u>

PUTNN dest,netn,addr,lngth,creg,sreg

where

dest	is the address of the network device in which to PUT data
netn	is the NET-to-NET routing address
addr	is the decimal address in the target device into which data is to be written
lngth	is the number of registers (16 bit values) to written, starting at addr
dreg	is the starting register number to send data from
sreg	is the communication status register

This command writes data to a PLC on another network Refer to Section A 4 for communication status register error bit responses

<u>Example.</u>

PUTNN 15,30,20,100,#30,#20

This command writes 100 registers in the destination SY/MAX device starting at register 20 The target device is along the NET-to-NET address of 30, at destination address 15 The data is written starting from our registers 30 through 129 Register 20 contains the communication status

2000-11

Westinghouse Numa-Logic PLC Interface

Printed in the United States of America
Part Number 96760-011

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORD

Revision	Description	Date
A	Appendix Released	7/90

Copyright Information

Address comments concerning this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number 96760-011A

A 1 INTRODUCTION

This chapter contains all of the information that is specific to the 2000-11 Westinghouse Numa-Logic PLC interface The protocol to be used by the module to communicate with the Westinghouse PLC is specific to the Numa-Logic PLC Refer to the Westinghouse documentation for details

The specific commands that allow communications are listed in Section A 5 They are

GET
PUT
PUTIO

A 2 CABLING THE WORKSTATION TO THE PLC

The PLC and/or network connects to the 9-pin primary serial port of the 2000 Workstation via a cable This is not a standard cable The connectors must be fabricated Figure A-1 on the next page illustrates the connections that must be made between the primary port and the PLC

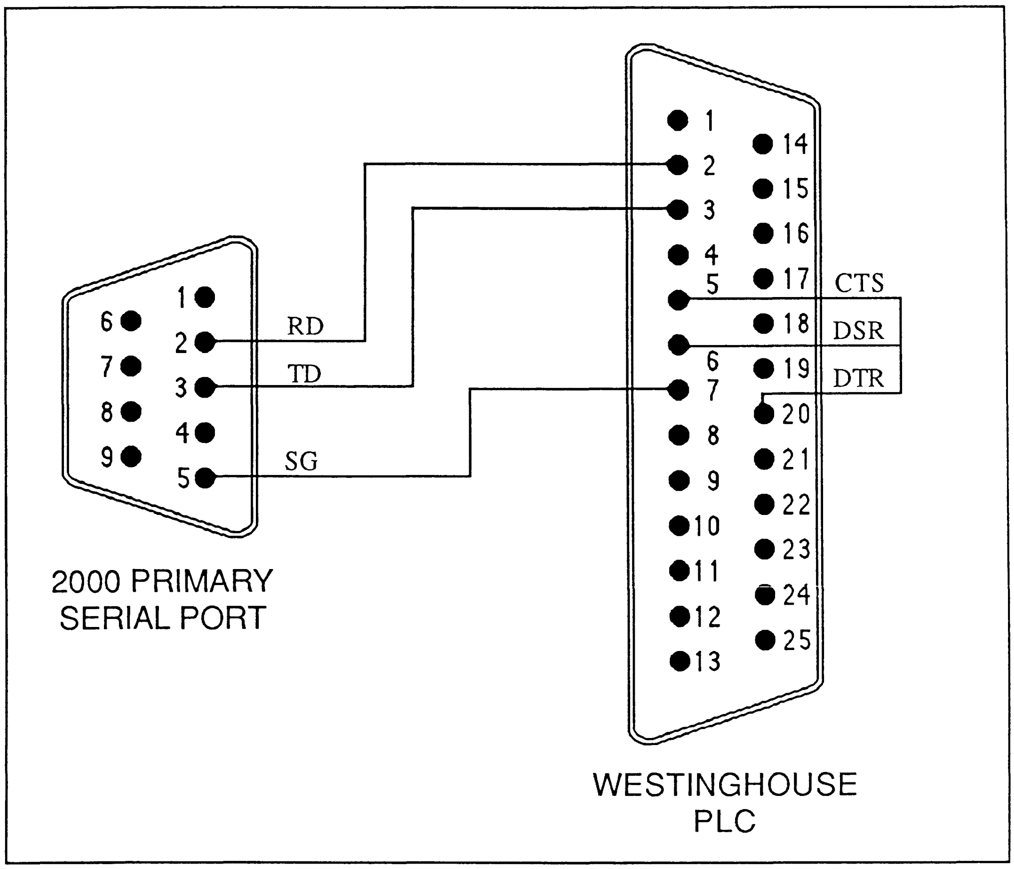


Figure A-1 Connections

A 3 CONFIGURATION MENU

The configuration menus are called up from the Main Menu (discussed in Chapter 3) The Westinghouse Menu looks like the following

```
            -- Westinghouse Port Configuration Menu (P) --

 6 Baud  -  1=300  2=600  3=1200  4=2400  5=4800  6=9600  7=19 2
 1 Parity - 0 = even   1 = odd
 1 1 = Parity Enabled              0 = Disabled
01 Communications Timeout Value (1-10 seconds)
 0 1 = Enable Multiplex Handshaking  0 = Disable Multiplex Handshaking



 Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW>
 Use values 0 through 9
 "C" for next configuration menu, <ESC> or <ENTER> to quit
```

Figure A-2 Configuration Menu

Baud The baud rate of the channel should be set to match that of the PLC

Parity This option will set the parity used in communication to the Westinghouse PLC to either none or even The type selected should match the other communication device(s)

Parity Enabled/Disabled This option will enable or disable the type of parity selected above

Communications Timeout Value The timeout configuration option determines how long the terminal will wait for a response before it signals a timeout The value entered will be in seconds, with a range of 1 to 10

Enable/Disable Multiplex Handshaking This option is used to determine whether multiplex handshaking is enabled Multiplex handshaking is used when a hardware multiplexing box is connected between the primary serial port of the 2000 and the PLC The multiplex box allows multiple 2000 units to connect to a single PLC Data transmissions from the 2000 to the PLC are controlled by the RTS and CTS lines connected between the 2000 and the multiplex box The 2000 firmware automatically controls the RTS/CTS lines The Quartech multiplex box has been thoroughly tested for use with the 2000-11

A 4 COMMUNICATION STATUS REGISTER

The Communication Status Register contains the status about the message that was just transferred The Most Significant Byte (MSB) is the status returned from the Westinghouse PLC The Least Significant Byte (LSB) contains information about the message transfer status A zero value in the entire register indicates a successful transfer

If the bit noted is set, the condition indicated is true

Status Register

MSB Bits 8-11 = **NOT USED**
Bits 12-15 = see below

BIT 15	14	13	12	
0	0	0	1	= Attempt to Write Ladder While Running
0	0	1	0	= Invalid Command Op Code
0	0	1	1	= Checksum Error
0	1	0	0	= Command Overrun
0	1	0	1	= Command Aborted
0	1	1	0	= UART Overrun
0	1	1	1	= Invalid Address
1	0	0	0	= UART Framing Error
1	0	0	1	= UART Parity Error

LSB Bit 0 = Transmit Error
Bit 1 = Receive Error
Bit 2 = Timeout
Bit 3 = Parity Error
Bit 4 = Synchronization Error
Bits 5-7 = **NOT USED**

A 5 COMMANDS

This section describes the commands that are specific to the 2000-11 Westinghouse Numa-Logic interface

A 5 1 GET PLC DATA (GET)

Syntax

GET addr,lngth,dreg,sreg

where

addr is the address in a target station to read data from

lngth is the number of words (16-bit values) to read, starting at addr

dreg is the destination register number to store data

sreg is the Communication Status Register

The command GET reads a 16-bit word(s) from a data table in the PLC and places the result in a specified register Refer to Section A 4 for the Communication Status Register information

Example

GET &1200,10,#30,#20

This command will read 10 words of data starting at address 1200 (hex) of the target device The data returned will be stuffed into registers #30 through #39 Register #20 will contain the Communication Status

A 5 2 **PUT PLC DATA (PUT)**

Syntax

PUT addr,lngth,dreg,sreg

where

addr is the address in a target station to write data into

lngth is the number of words (16-bit values) to write, starting at addr

dreg is the starting register number to send data from

sreg is the Communication Status Register

The command PUT writes a 16-bit word(s) from a specified register to a data table in the PLC Refer to Section A 4 for the Communication Status Register information

Example

PUT 330,110,#30,#20

This command will write 110 words to address 330 of the target device The data sent will be copied from registers #30 through #139 Register #20 will contain the Communication Status

A 5 5 WRITE TO A BIT IN THE TARGET PLC (PUTIO)

Syntax

PUTIO addr,bit,state,sreg

where

addr is the address in a target station to write data into

bit is the specific bit within addr to modify

state is the bit value to write
0 - RESET
1 - SET

sreg is the Communication Status Register

The command PUTIO sets a specified bit in the target device to the value in an indicated register The value of the register must be either 0 or 1 Refer to Section A 4 for the Communication Status Register information

Example

PUTIO 102,10,#30,#20

This command will SET or RESET bit 10 at address 102 of the target device according to the value contained in register #30 (must be 1 or 0) Register #20 will contain the Communication Status

2000-12

General Electric CCM Port PLC Interface

Printed in the United States of America
Part Number 96760-012

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORD

Revision	*Description*	*Date*
A	Appendix Released	7/90

Address comments concerning this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number *96760-012A*

A 1 INTRODUCTION

This appendix contains information specific to the 2000-12 General Electric Series One, Five, and Six PLC interfaces The protocol to be used by the module to communicate with the GE PLC is a full-duplex, asynchronous, point-to-point protocol that closely conforms to ANSI X3 28

The specific commands that allow communications are listed in Section A 5 They are

GETR
GETIS
GETOS
PUTR
PUTOS

The 2000-12 will communicate directly with Series One, Five, and Six PLCs The communication standard of the Series One is RS-232 The Series Five and Six PLCs can use either the RS-232 or the RS-422 communication standard All connections are made from the 9-pin primary serial port of the 2000 Workstation

For RS-422 communications with the Series Five or Six PLC, jumper the primary port for RS-485

A 2 CABLING THE TERMINAL TO THE PLC

The PLC connects to the 9-pin primary serial port of the 2000 Workstation via a cable This is not a standard cable The connectors must be fabricated Figures A-1 through A-9 illustrate the connections that must be made between the primary port and the PLC

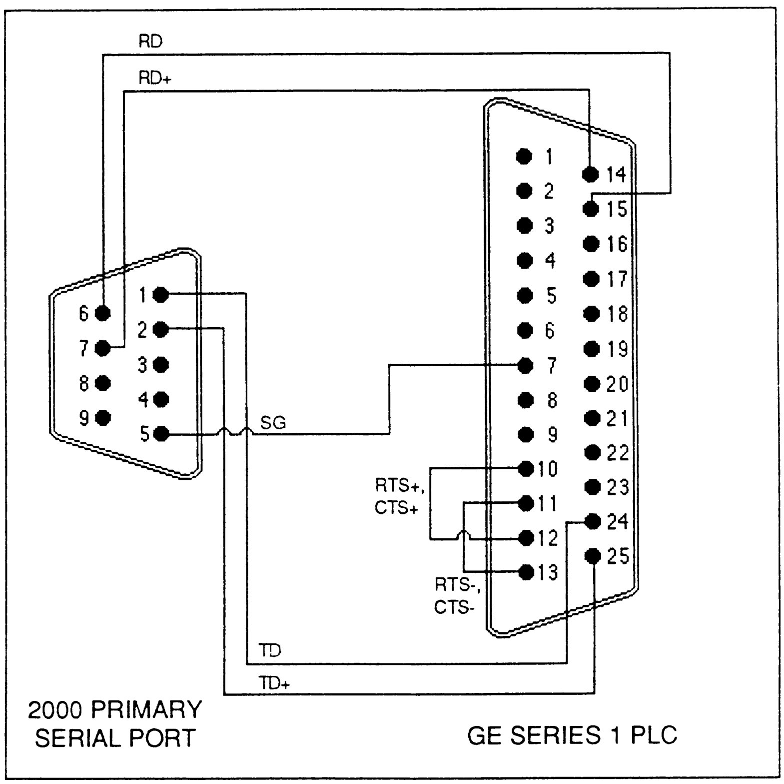


Figure A-1 Connection to GE Series One PLC

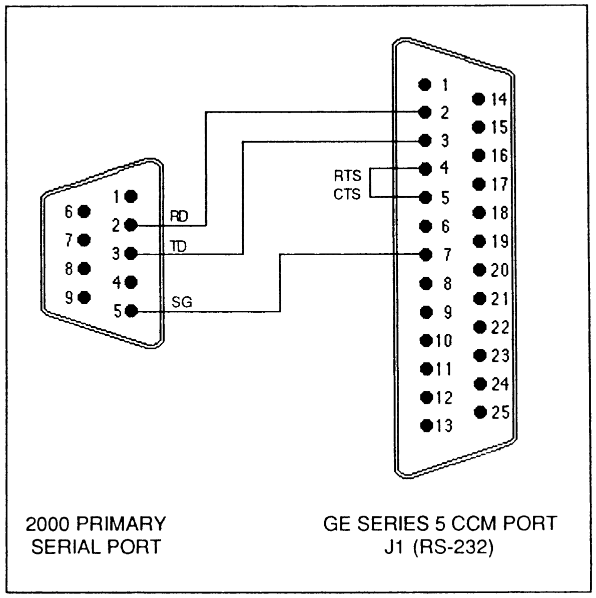


Figure A-2 Connection to GE Series Five CCM Port J1, RS-232

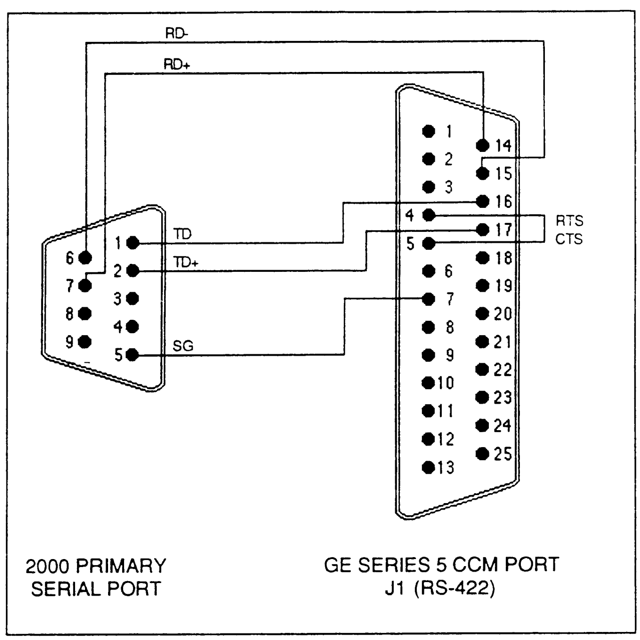


Figure A-3 Connection to GE Series Five CCM Port J1, RS-422

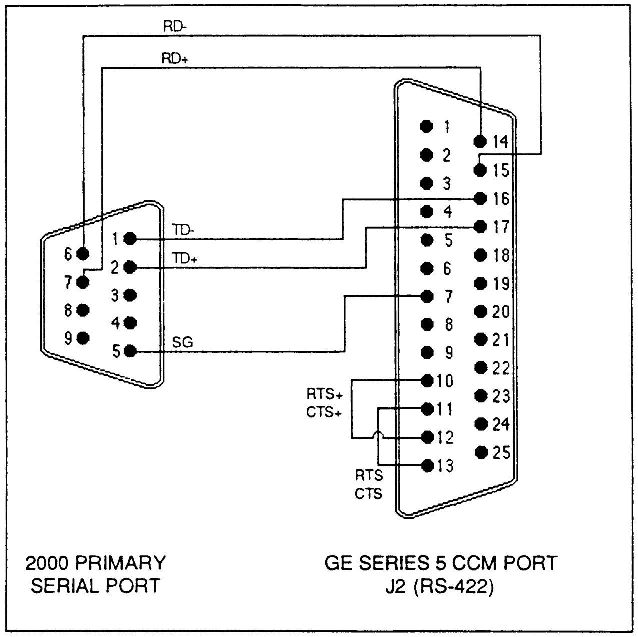


Figure A-4 Connection to GE Series Five CCM Port
J2, RS-422

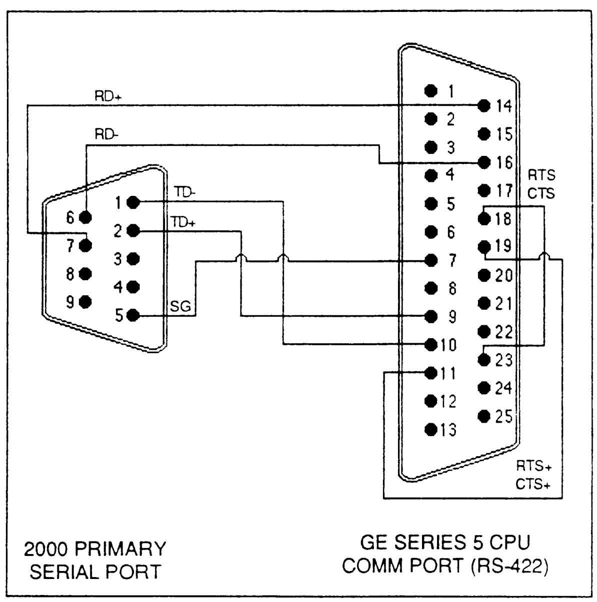


Figure A-5 Connection to GE Series Five CPU Comm Port
RS-422

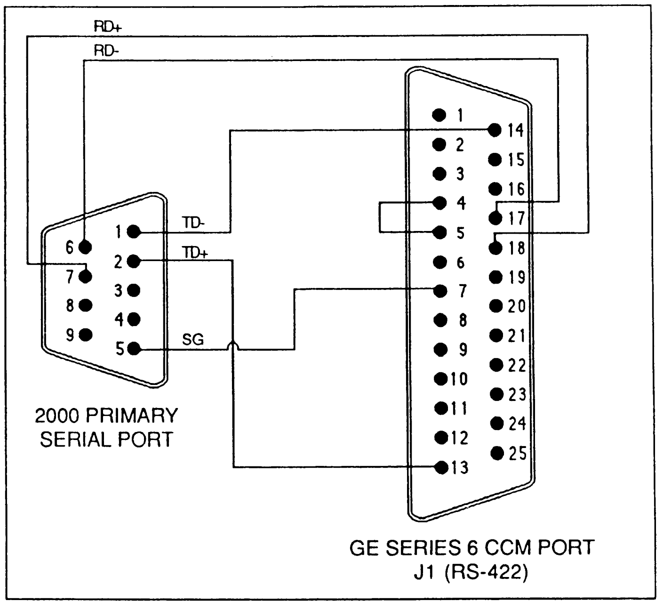


Figure A-6 Connection to GE Series Six CCM Port
J1, RS-422

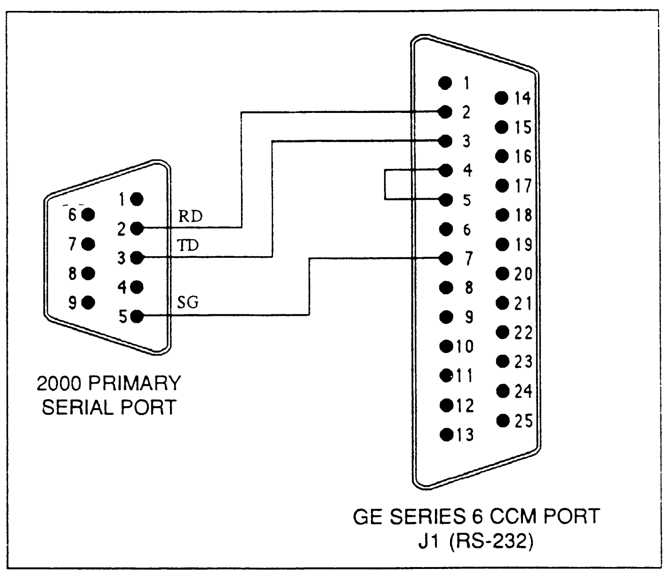


Figure A-7 Connection to GE Series Six CCM Port
J1, RS-232

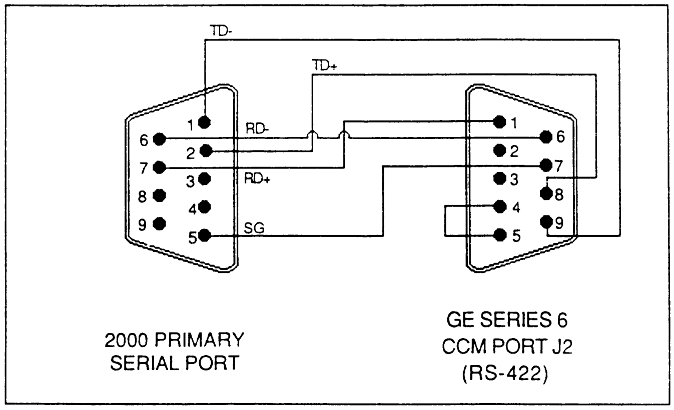


Figure A-8 Connection to GE Series Six CCM Port J2, RS-422

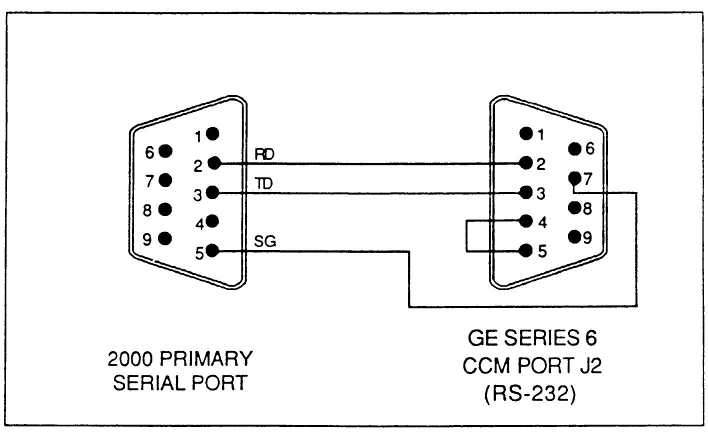


Figure A-9 Connection to GE Series Six CCM Port J2, RS-232

A 3 CONFIGURATION MENU

The configuration menus are called up from the Main Menu (discussed in Chapter 3) The General Electric Menu looks like the following

```
-- GE CCM Configuration Menu --

6  Baud - 1=300 2=600 3=1200 4=2400 5=4800 6=9600 7=19 2K
0  Parity  0=No Parity 1=Odd Parity
50 Source ID# (1-90)
1  Timeout Value (1-20 seconds)
0  1 = Enable Multiplex Handshaking  0 = Disable Multiplex Handshaking


Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW>
Use values 0 through 9
"C" for next configuration menu, <ESC> or <ENTER> to quit
```

Figure A-10 Configuration Menu

Baud The baud rate of the channel should be set to match that of the PLC

Parity This option will set the parity used in communication on the General Electric PLC to either none or odd The type selected should match the other communication device(s)

Source ID# If the terminal is set up in a network configuration, one or more communication module/PLCs will be used The Source ID number setting in the General Electric Configuration menu **MUST** be identical to the ID number of the GE PLC CPU to which the terminal is directly connected

Refer to the General Electric user reference material for information on obtaining the CPU ID number

Timeout Value The timeout configuration option determines how long the terminal will wait for a response before it signals a timeout The value entered will be in seconds, with a range of 1 to 20

Enable/Disable Multiplex Handshaking This option us used to determine whether multiplex handshaking is enabled Multiplex handshaking is used when a hardware multiplexing box is connected between the primary serial port of the 2000 and the PLC The multiplex box allows multiple 2000 units to connect to a single PLC Data transmissions from the 2000 to the PLC are controlled by the RTS and CTS lines connected between the 2000 and the multiplex box The 2000 firmware automatically controls the RTS/CTS lines The Quartech multiplex box has been thoroughly tested for use with the 2000-12

A 4 COMMUNICATION STATUS REGISTER

The Communication Status Register contains the status about the message that was just transferred The Most Significant Byte (MSB) is not used The Least Significant Byte (LSB) contains information about the message transfer status A zero value in the entire register indicates a successful transfer

If the bit noted is set, the condition indicated is true

Status Register

MSB **NOT USED**

LSB Message Transfer Status

Bit 0 = Transmit Error
Bit 1 = Receive Error
Bit 2 = Timeout
Bit 3 = Parity Error
Bit 4 = Enquiry Error
Bits 5-7 = **NOT USED**

A 5 COMMANDS

This section describes the commands that are specific to the 2000-12 General Electric Series One, Five, and Six PLC interfaces

A 5 1 GET REGISTER (GETR)

Syntax

GETR dest,reg,lngth,dreg,sreg

where

dest is the CPU ID number of destination PLC

reg is the register in the target PLC from which the data is to be read

lngth is the number of words to read, starting at addr

dreg is the destination register number to store data

sreg is the Communication Status Register

The command GETR reads data from the Register Table in the PLC Refer to Section A 4 for the Communication Status Register information

Example

GETR 10,20,15,#30,#20

This command will read 15 registers starting at register #20 of the target device with ID of 10 The data returned will be stuffed into registers #30 through #39 Register #20 will contain the Communication Status

A 5 2 **GET INPUT STATUS (GETIS)**

Syntax

GETIS dest,addr,lngth,dreg,sreg

where

dest is the CPU ID number of destination PLC

addr is the starting address of Input Status to read

lngth is the number of words to read, starting at addr

dreg is the destination register number to store data

sreg is the Communication Status Register

The command GETIS reads data from the Input Table in the PLC Refer to Section A 4 for the Communication Status Register information

Example

GETIS 12,40,5,#30,#20

This command will read 5 words of Input Status starting at address 40 of the target device with ID of 12 The data returned will be stuffed into registers #30 through #34 Register #20 will contain the Communication Status

A 5 3 GET OUTPUT STATUS (GETOS)

Syntax

GETOS dest,addr,lngth,dreg,sreg

where

dest is the CPU ID number of destination PLC

addr is the starting address of Output Status to read

lngth is the number of words to read, starting at addr

dreg is the destination register number to store data

sreg is the Communication Status Register

The command GETOS reads data from the Output Table in the PLC Refer to Section A 4 for the Communication Status Register information

Example

GETOS 12,40,5,#30,#20

This command will read 5 words of Output Status starting at address 40 of the target device with ID of 12 The data returned will be stuffed into registers #30 through #34 Register #20 will contain the Communication Status

18 6 1 **PUT REGISTER (PUTR)**

Syntax

PUTR dest,addr,lngth,srcreg,sreg

where

dest is the CPU ID number of destination PLC

addr is the register in the target PLC from which the data is to be written

lngth is the number of words to write, starting at addr

srcreg is the source register number to send data from

sreg is the Communication Status Register

The command PUTR sends data to the Register Table in the PLC Refer to Section A 4 for the Communication Status Register information

Example

PUTR 15,33,13,#30,#20

This command will write 13 words of data starting at register #33 of the PLC with ID of 15 The data sent is contained in registers #30 through #42 of the workstation Register #20 will contain the Communication Status

A 5 3 **PUT OUTPUT STATUS (PUTOS)**

<u>Syntax</u>

PUTOS dest,addr,lngth,srcreg,sreg

where

dest is the CPU ID number of destination PLC

addr is the starting address of Output Status to write

lngth is the number of words to write, starting at addr

srcreg is the source register number to write data from

sreg is the Communication Status Register

The command PUTOS writes one or more words of output to the Output Table in the PLC Refer to Section A 4 for the Communication Status Register information

<u>Example</u>

PUTOS 12,40,1,#30,#20

This command will write 1 word of Output Status starting at address 40 of the PLC with ID of 12 The data sent is contained in register #30 of the workstation Register #20 will contain the Communication Status

2000-13

Siemens PLC Interface

Printed in the United States of America
Part Number 96760-013

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORD

Revision	***Description***	***Date***
A	Appendix Released	3/91

Address comments concerning this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number ***96760-013A***

A 1 INTRODUCTION

This chapter contains information specific to the 2000-13 Siemens S5-115U PLC interface

The specific commands that allow communications are described in Section A 5 and listed below for reference

GET	Get PLC data from the PLC
GETDB	Get data block from the PLC
GETTC	Get timer/counter data from the PLC
PUTDB	Write data block to PLC
COMDB	Compress data block

> **NOTE**
> Before operating this direct connect, configure the primary port for RS-422 by positioning jumpers J13 to J20 to A

A 2 CABLING THE TERMINAL TO THE PLC

The 2000 connects to the Siemens PLC via a 20 mA current loop adapter available from Xycom This loop adapter connects to the 9-pin primary serial port of the 2000 Workstation, and to a 15-pin, user-fabricated cable The other end of the user fabricated cable then connects to the Siemens PLC as illustrated in Figure A-1 below The pinouts for the connections are shown in Figure A-2 on the following page

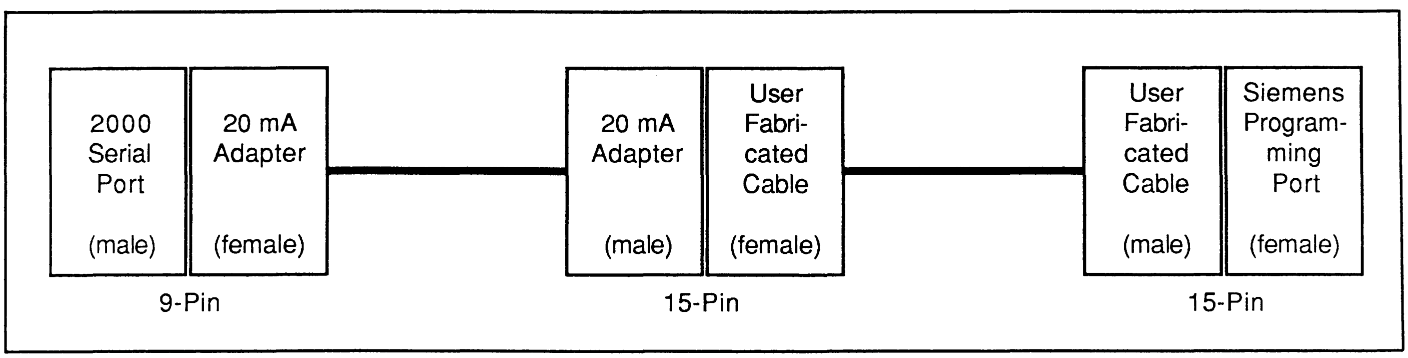


Figure A-1 20 mA Cable Connections

The connectors for this cable must be fabricated Figure A-2 below illustrates the connections that must be made between the 20 mA adapter cable and the PLC

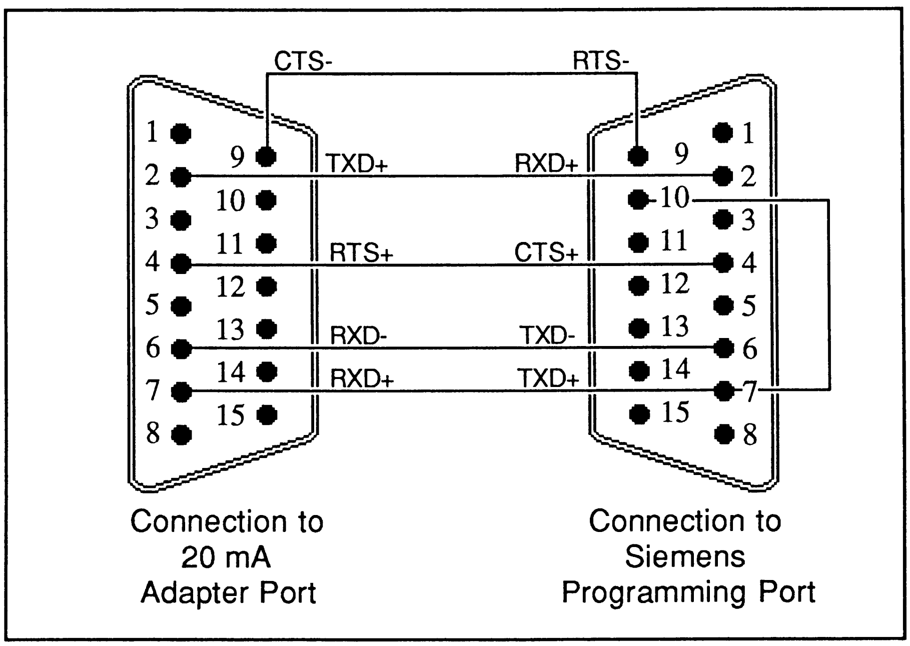


Figure A-2 Connection to the Siemens S5-115U PLC

A 3 CONFIGURATION MENU

The Configuration Menu is accessed from the 2000 Main Menu (described in Chapter 3) The Siemens Menu is shown in Figure A-3 below

```
                    -- Siemens Port Configuration Menu (P) --

0  1=Enable Multiplex Handshaking  -- 0=Disable Multiplex Handshaking





Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW>
Use values 0 through 9
"C" for next configuration menu, <RET> or <ENTER> to quit
```

Figure A-3 Primary Configuration Menu

Multiplex Handshaking This option enables or disables multiplex handshaking Multiplex handshaking is used when a hardware multiplexing box is connected between the primary serial port of the 2000 and the PLC The multiplex box allows multiple 2000 units to connect to a single PLC Data transmissions from the 2000 to the PLC are controlled by the RTS and CTS lines connected between the 2000 and the multiplex box The 2000 firmware automatically controls the RTS/CTS lines when this option is enabled

> **NOTE**
> If using multiplex handshaking, the CTS and RTS signals must be connected as specified by your particular hardware

The rest of the configuration is fixed and cannot be altered The values are as follows

Baud	=	9600
Parity	=	Even
Data Bits	=	8
Stop Bits	=	1

A 4 DIAGNOSTICS MENU

The Diagnostics Menu is accessed from the Main Menu (see Chapter 3) and lists the diagnostic tests that can be performed on the 2000 The Diagnostics Menu specific to the 2000-13 is shown in Figure A-4 below

```
-- Diagnostics --

1)  Complete Test
2)  Continuous Test
3)  RAM
4)  ROM Checksum
5)  EPROM Test
6)  Real Time Clock Test
7)  RS-232 Serial Loopback
8)  RS-485/Multidrop Serial Loopback
9)  Printer Port Test
A)  Parallel Input Test
B)  Matrix Keyboard Loopback Test
C)  Beeper Test
D)  Battery Test
E)  Dipswitch Test
F)  Character Attributes
G)  CRT Crosshatch Pattern
H)  CRT Brightness Pattern
I)  Touch Screen Test

<ESC> or <ENTER> to quit
```

Figure A-4 Diagnostics Menu

> **NOTE**
> Option I only appears if the optional touch screen is installed on your unit

See Chapter 3 for explanations of each of the menu choices

A 5 COMMUNICATION STATUS REGISTER

The Communication Status Register in the OIL command contains the status of the message previously transferred If the bit number in the specified OIL register is set, the condition corresponding to that value (as described below) is true

Status Register

0	No errors
1	Unable to establish communications
2	Timeout
3	Parity error
4	Response error
5	Data block does not exist
6	Write error

A 6 COMMANDS

This section describes the commands that are specific to the 2000-13 Siemens PLC interface

The following lists the address types within the Siemens PLC that can be read by the 2000

Table A-1 Siemens Address Types

TYPE	RANGE(b)	DESCRIPTION
Ib x	0 to 127	input image (PII)
Ob x	0 to 127	output image (PIQ)
Fb x	0 to 255	flags (F)
	0 to 127	timers
	0 to 127	counters
	2 to 255	data blocks

where
- b is the byte number in that address type (valid range given in column 2)
- x is the bit number within the specified byte

A 6 1 **GET PLC DATA (GET)**

Syntax.

GET type,addr,lngth,dreg,sreg

where

type —— is the number identifying what type of memory to read
0 = input image (PII)
1 = output image (PIQ)
2 = flags (F)

addr —— is the address of the block to read (see example below)

lngth —— is the number of bytes to read

dreg —— is the initial destination register for value(s) Each byte read from the PLC is stored in a separate OIL register

sreg —— is the communication status register

The command GET reads data from the PLC and stores it in specified OIL registers The GET command reads only byte values from the PLC The various OIL bit commands will have to be used to check a specific bit

Example.

GET 1,0,128,#100,#229

This command reads 128 output image values beginning at byte 0 The data read is placed into registers 100 through 227 Register 229 contains the communication status

A 6 2 GET DATA BLOCK (GETDB)

Syntax.

GETDB block,maxlngth,dreg,lreg,sreg

where

block —— is the number of the data block to read (2 to 255)

maxlngth–is the maximum number of words that will be stored into the OIL registers

dreg —— is the initial destination register for value(s) Each byte read from the PLC is stored in a separate OIL register

lreg —— is the length register containing the actual size of the data block

sreg —— is the communication status register

The command GETDB reads a data block from the PLC and stores it beginning with a specified OIL register The length of the data block refers only to the length of the data area and does not include the header The header is not stored

Example.

GETDB 2,1,#15,#16,#17

This command reads the first word from data block 2 The data read is placed into register 15 Register 16 stores the block length read Register 17 contains the communication status

A 63 GET TIMER/COUNTER DATA (GETTC)

Syntax.

GETTC type,addr,lngth,TCsreg,TCvreg,sreg

where

type —— is the data read
0 = timer
1 = counter

addr —— is the initial timer/counter to read

lngth —— is the number of timers/counters to read

TCsreg — is the initial destination for the timer/counter status data

TCvreg — is the initial destination for the timer/counter value data

sreg —— is the communication status register

TCsreg and TCvreg store two sections of the timer/counter data word Bits 15 - 10 of TCsreg contain the timer/counter status data Bits 9 - 0 of TCvreg contain the current timer/counter value

The command GETTC reads either the timer or counter status, and the value data from the PLC and stores them in the specified OIL registers

Example.

GETTC 0,0,1,#20,#30,#40

This command reads the data from timer 0 The timer status is placed into register 20, while register 30 holds the timer value Register 40 contains the communication status

A 6 4 PUT DATA BLOCK (PUTDB)

Syntax.

PUTDB block,lngth,srcreg,sreg

where

block —— is the number of the data block to write (2 to 255)

lngth —— is the maximum number of words that will be stored in the PLC

srcreg —— is the initial register to copy that data for the data block

sreg —— is the communication status register

> **NOTE**
> Only data blocks can be written to the PLC All other PLC memory areas are read-only

The command PUTDB reads data from the specified OIL register(s) to create a data block to write to the PLC The data block refers only to the length of the data area and does not include the header A header is automatically built when the block is created

Example.

PUTDB 2,1,#13,#14

This command reads one word from OIL register 13 and creates data block 2 Register 14 contains the communication status

> **NOTE**
> After repeated data block writes, the PLC runs out of memory and has to compact the data block area If a write fails because the PLC is out of memory, the driver commands the PLC to compact its data block memory area

A 6 5 COMPRESS DATA BLOCK (COMDB)

Syntax.

COMDB sreg

where
sreg is the communication status register

The command COMDB compresses the data block memory area in the PLC

NOTE
The data block compaction process may take up to 30 seconds, during which time the PLC may not continue running its program For example, under the following conditions, compaction occurs every 12 writes and takes seven seconds using an S2 115U CPU, running no PLC program, and using a 255 word long written data block

NOTE
This command should be issued when the PUTDB command returns with a write error

Example.

COMDB #13

This example compresses the data block area Register 13 contains the communication status

2000-14

Allen-Bradley T3 Port Interface

Printed in the United States of America
Part Number 96760-014

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORD

Revision	*Description*	*Date*
A	Appendix Released	5/91

Copyright Information

Address comments concerning
this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number ***96760-014 A***

A 1 INTRODUCTION

The protocol used by the module to communicate with the Allen-Bradley PLC is a full-duplex, asynchronous, point-to-point protocol that closely conforms to ANSI X3 28

The specific commands that allow communications are discussed in detail in Section A 5, with examples These commands are

GET
GETDIAG
GETIO
PUT
PUTIO

A 2 CABLING THE TERMINAL TO THE PLC

The primary port is attached to the PLC using a non-standard cable configured for RS-422 connection The connectors must be fabricated Figure A-1 below shows the connections between the primary port and the PLC

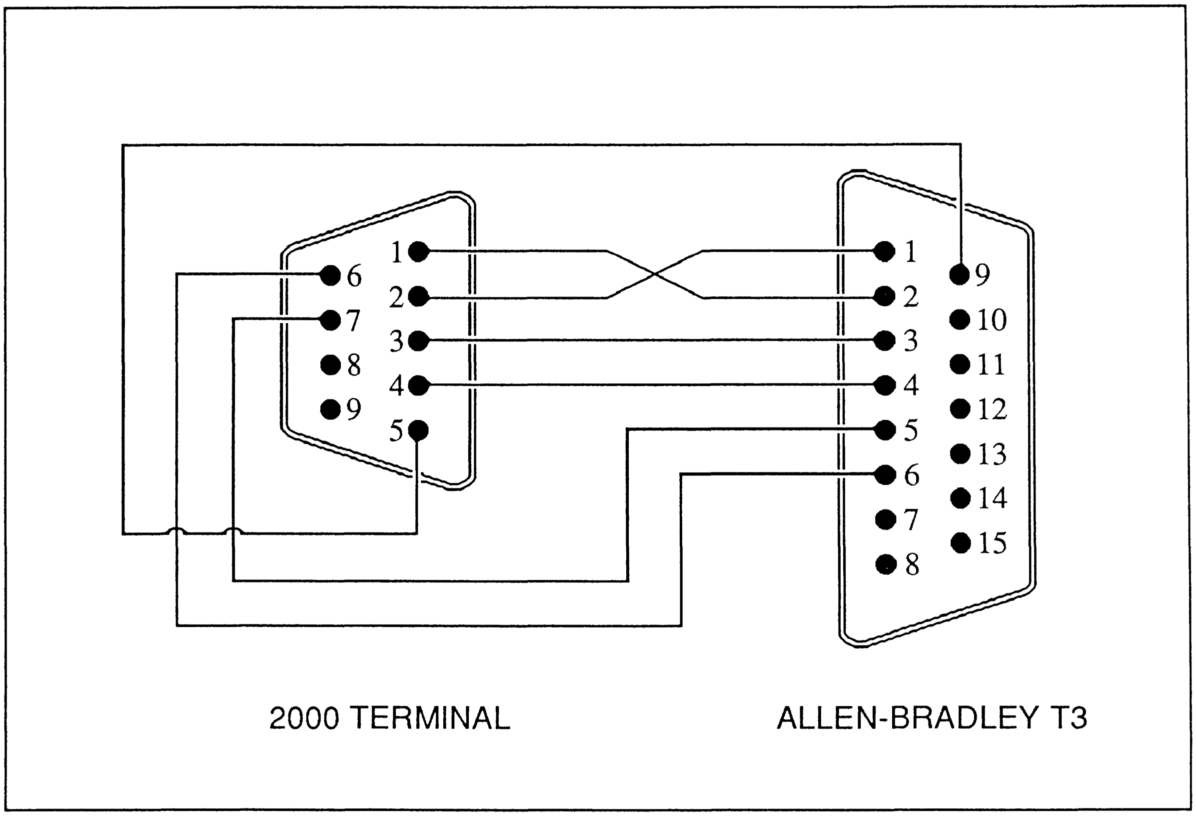


Figure A-1 Connections

NOTE
Before connecting the units, configure the 2000 primary port to RS-422 by positioning jumpers J13-J20 to A

A 3 DIAGNOSTIC MENU

The serial ports on the terminal can be tested by selecting item #4 (Serial Loop Back) from the Diagnostic Menu This test requires certain signals to be "looped back" to the terminal for signal verification The recommended means for looping signals is to construct a loop-back connector using a D-type connector of the same size as the port The figure below illustrates the configuration of the test plugs

```
                    -- Diagnostics --

1)      Complete Test
2)      Continuous Test
3)      RAM
4)      ROM Checksum
5)      EPROM Test
6)      Real Time Clock Test
7)      RS-232 Serial Loopback
8)      RS-485/Multidrop Serial Loopback
9)      Printer Port Test
A)      Parallel Input Test
B)      Matrix Keyboard Loopback Test
C)      Beeper Test
D)      Battery
E)      Dipswitch Test
F)      Character Attributes
G)      CRT Crosshatch Pattern
H)      CRT Brightness Pattern
I)      Touch Screen

<ESC> or <ENTER> to quit
```

Figure A-2 Diagnostic Menu

NOTE
Option I only appears if the optional touch screen is installed on your unit

See Chapter 3 for explanations of each of the menu choices

A 4 COMMUNICATION STATUS REGISTER

The Communication Status Register in the OIL command contains the status of the message previously transferred The data value returned in the message from the Communication Module will denote a positive decimal digit If a value is returned in the Status Register then the condition corresponding to that value described below is true

Status Register

0	No errors
1	Response error
2	Timeout
3	Parity error
4	Address outside PLC data table

A 5 COMMANDS

This section describes the commands that are specific to the 2000-14 Allen-Bradley T3 Programming Port interface

A 5 1 GET BLOCK (GET)

Syntax.

GET addr,lngth,dreg,sreg

where

addr is the Octal Address of the block to read

lngth is the Length of the read (>= 1)

dreg is the Initial Destination Register for value(s)

sreg is the Communication Status Register

The command GET will read a block of one or more words from the data table in an Allen-Bradley PLC and store it in a specified OIL register Refer to Section A 4 for the Communication Status Register information

Example.

GET 101,3,#21,#20

This command will read the block of information in data table locations 101 through 103 of the PLC The data read will be stuffed into registers 21 through 23 Register 20 will contain the Communication Status

A 5 2 GET BIT (GETIO)

Syntax.

GETIO addr,lbit,dreg,sreg

where

addr is the Octal Address of the block to read

lbit is the bit number (0-15)

dreg is the Initial Destination Register for value(s)

sreg is the Communication Status Register

The command GETIO will read a specified bit in a word from the input table in an Allen-Bradley PLC Refer to Section A 4 for the Communication Status Register information

Example.

GETIO 1,2,#21,#20

This command will read bit 2 of the word at address 1 of the PLC The data read will be stuffed into register 21 Register 20 will contain the Communication Status

A 5 3 GET DIAGNOSTIC STATUS (GETDIAG)

Syntax.

GETDIAG dreg,sreg

where

dreg is the Initial Destination Register for value

sreg is the Communication Status Register

The command GETDIAG will read the diagnostic status (size of the data table) from the Allen-Bradley PLC and return this to a specified register Refer to Section A 4 for the Communication Status Register information

Example.

GETDIAG #21,#20

This command will return the diagnostic status to register 21 Register 20 will contain the Communication Status

A 5 4 PUT BLOCK (PUT)

Syntax.

PUT addr,lngth,dreg,sreg

where

addr is the Octal Address of the block to write

lngth is the Length of the write (>= 1)

dreg is the Initial Source Register for value(s)

sreg is the Communication Status Register

The command PUT will write a block of one or more words from specified OIL registers to a location in the data table in an Allen-Bradley PLC Refer to Section A 4 for the Communication Status Register information

Example.

PUT 101,3,#21,#20

This command will write the block of information contained in OIL register 21 through 23 into data table locations 101 through 103 of the PLC Register 20 will contain the Communication Status

A 5 5 **PUT BIT (PUTIO)**

Syntax.

PUTIO addr,lbit,dreg,sreg

where

addr is the Octal Address of the block to write

lbit is the bit number (0-15)

dreg is the Initial Source Register for value(s)

sreg is the Communication Status Register

The command PUTIO will write read a specified bit value (0/1) from an OIL register to a word in the data table in an Allen-Bradley PLC Refer to Section A 4 for the Communication Status Register information

Example.

PUTIO 1,2,#21,#20

This command will write the logical value (0/1) in register 21 to bit 2 of the word at address 1 of the target device Register 20 will contain the Communication Status

2000-16

Texas Instruments Series 405/435 PLC Interface

Printed in the United States of America
Part Number 96760-016

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORD

Revision	Description	Date
A	Appendix Released	3/91

Address comments concerning this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number: 96760-016A

A 1 INTRODUCTION

This chapter contains information specific to the Texas Instruments Series 405/435 PLC interface The K-sequence protocol is used to communicate with the Texas Instruments Series 405/435 PLC

The specific commands that allow communications are described in Section A 6, and listed below for quick reference

GET	Get data from PLC
PUT	Write data to PLC

A 2 CABLING THE TERMINAL TO THE PLC

The PLC and/or PLC network connects to the 9-pin primary serial port of the 2000 Workstation via a non-standard cable The connectors for this cable must be fabricated Figure A-1 below illustrates the connections that must be made between the primary serial port and the PLC

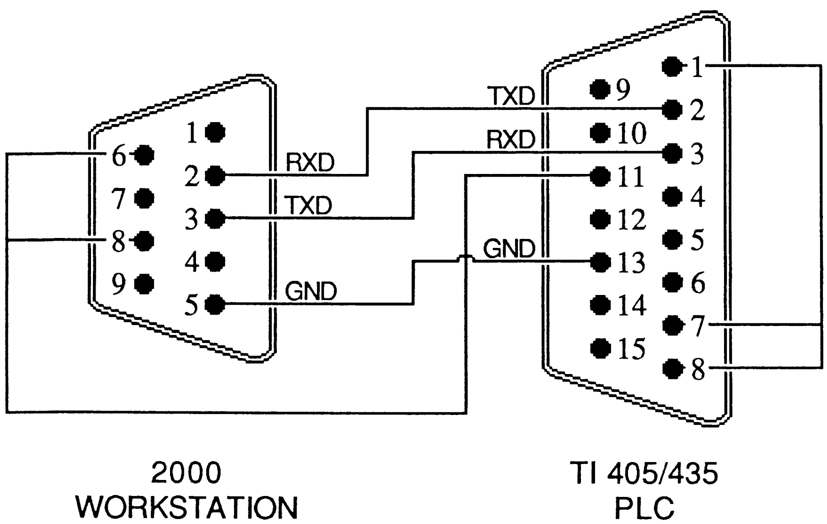


Figure A-1 Connection to the TI-405/435 PLC

A 3 CONFIGURATION MENU

The Configuration Menu is accessed from the 2000 Main Menu (described in Chapter 3) The Configuration Menu is shown in Figure A-2 below The only user-configurable item is multiplex handshaking, which is described below

```
                 -- TI 405/435 Configuration Menu (P) --

0  1=Enable Multiplex Handshaking  --  0=Disable Multiplex Handshaking




Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW>
Use values 0 through 9
"C" for next configuration menu, <RET> or <ENTER> to quit
```

Figure A-2 Configuration Menu

Multiplex Handshaking This option enables or disables multiplex handshaking Multiplex handshaking is used when a hardware multiplexing box is connected between the primary serial port of the 2000 and the PLC The multiplex box allows multiple 2000 units to connect to a single PLC Data transmissions from the 2000 to the PLC are controlled by the RTS and CTS lines connected between the 2000 and the multiplex box The 2000 firmware automatically controls the RTS/CTS lines when this option is enabled

> **NOTE**
> If using multiplex handshaking, the CTS and RTS signals must be connected as specified by your particular hardware

The rest of the configuration is fixed and cannot be altered These values are as follows

Baud	=	9600
Stop Bits	=	1
Data Bits	=	8
Start Bits	=	1
Parity	=	Odd

A 4 DIAGNOSTICS MENU

The Diagnostics Menu is accessed from the Main Menu (see Chapter 3) and lists the diagnostic tests that can be performed on the 2000 The Diagnostics Menu specific to the 2000-16 is shown in Figure A-3 below

```
-- Diagnostics --

1) Complete Test
2) Continuous Test
3) RAM
4) ROM Checksum
5) EPROM Test
6) Real Time Clock Test
7) RS-232 Serial Loopback
8) RS-485/Multidrop Serial Loopback
9) Printer Port Test
A) Parallel Input Test
B) Matrix Keyboard Loopback Test
C) Beeper Test
D) Battery Test
E) Dipswitch Test
F) Character Attributes
G) CRT Crosshatch Pattern
H) CRT Brightness Pattern
I) Touch Screen Test

<ESC> or <ENTER> to quit
```

Figure A-3 Diagnostics Menu

NOTE

Option I) only appears if the optional touch screen is installed

See Chapter 3 for explanations of each of the menu choices

A 5 COMMUNICATION STATUS REGISTER

The communication status register contains status information about the message that was just transferred The most significant byte (MSB) is the status returned from the Texas Instruments PLC The least significant byte (LSB) contains information about the message transfer status

The status register is shown in Table A-1 below If the bit noted is set, the condition indicated is true

Table A-1 Communication Status Register

STATUS REGISTER		
MSB	PLC Response	
	C0H	Successful GET for memtypes 0-12
	C6H	Successful PUT for memtypes 0-12
	D0H	Successful GET for memtype 13
	D1H	Successful PUT for memtype 13
	FFH	Unsuccessful transfer
LSB	Message Transfer Status	
	Bit 0	Transmit error
	Bit 1	Receive error
	Bit 2	Timeout
	Bit 3	Parity error
	Bit 4	Overrun error
	Bit 5	Framing error
	Bit 6	Enquiry error
	Bit 7	**NOT USED**

A 6 COMMANDS

This section describes the commands that are specific to the 2000-16 Texas Instruments Series 405/435 PLC interface

Both the GET and PUT commands use a variable called memtype This refers to the memory type within the TI-405/435 PLC that the Xycom terminal can access The types, symbols, lengths, and descriptions are given in the table below Refer to the TI-405/435 technical manual for exact memory mapping details

Table A-2 Memory Types

TYPE	SYMBOL	LENGTH	DESCRIPTION
0	T	128	Timer accumulators
1	CT	128	Counter accumulators
2	U	3228	User registers
3	GX	32	Remote I/O memory
4	X	20	Input memory
5	Y	20	Output memory
6	C	30	Control relay
7	S	24	Stage memory
8	TI	8	Timer relay
9	CTI	8	Counter relay
10	SP1	6	Special relays 1
11	SP2	11	Special relays 2
12	V	17056	Anywhere in V memory
13	SPAD		Anywhere in scratch pad memory

A 6 1 GET DATA (GET)

Syntax

GET memtype,addr,lngth,dreg,sreg

where

memtype	is the number identifying type of memory to read (see Table A-2 on the previous page)
addr	is the address of the block to read
lngth	is the number of words to read, starting at addr
dreg	is the initial destination register number to store data
sreg	is the communication status register

The command GET reads 16-bit word(s) from a specified memory location in the TI-405/435 PLC and places the result in a specified OIL register Refer to Section A 5 for communication status register information

Example

GET 2,10,20,#50,#70

This command reads 20 words of data starting at address 10 of the "User Register" memory area of the TI-405/435 PLC The data returned is placed into registers 50 through 69 Register 70 contains the communication status

A 6 2 **PUT DATA (PUT)**

Syntax

PUT memtype,addr,lngth,srcreg,sreg

where

memtype	is the number identifying type of memory to write (see Table A-2 on page A-6)
addr	is the address of the block to read
lngth	is the number of words to read, starting at addr
srcreg	is the initial source register number to copy
sreg	is the communication status register

The command PUT writes 16-bit word(s) from a specified OIL register to a memory location in the TI-405/435 Refer to Section A 5 for communication status register information

Example

PUT 2,10,20,#70,#90

This command writes the data contained in OIL registers 70 through 89 to locations 10 through 29 of the "User Register" memory area of the TI-405/435 PLC Register 90 contains the communication status

2000-17

Omron PLC Operator Interface Language (OIL)

P/N 96760-017B

Printed in the United States of America
Part Number 96760-017B

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORD

Revision	*Description*	*Date*
A	Appendix Released	3/91
B	Appendix Updated and Released	6/92

Address comments concerning this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number *96760-017B*

XYCOM MANUAL BUG REPORT

We have provided this form to help us track possible improvements or errors that may exist in our manuals Should you find an error in this manual or have any helpful suggestions to improve its usefulness, please describe it in the space below To return this form, fold it in half so the postage-paid* side shows, and tape or staple it closed We appreciate your input and will incorporate the changes in future revisions

Current Information

Page number(s) ____________________ Figure number(s) ________________

Information as currently printed ______________________________

____

____

____

____

Proposed Change

Information as it *should* be printed ______________________________

____

____

____

____

Any additional information ______________________________

____

____

____

____

Address (Optional)

Name ______________________________

Title ______________________________

Company ______________________________

Address ______________________________

City ____________________ State ________ ZIP ______________

Telephone (_____)__________________

Thank you for your time and effort

Xycom Use Only			Log Number
Date Received:			
Date Resolved:			
Manner Resolved:	PCN #:	Revision:	Void:

2000-17 Omron PLC Operator Interface Language (OIL), 96760-017B

**If mailing this card from outside the U S , please use an envelope with appropriate postage*

XYCOM, INC.
750 N. MAPLE ROAD
SALINE, MICHIGAN 48176

POSTAGE WILL BE PAID BY ADDRESSEE

BUSINESS REPLY MAIL

First Class Permit No 42 Saline Michigan

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

Xycom Manual Bug Report

Thank you for returning this form We appreciate your feedback

A 1 INTRODUCTION

This appendix contains information specific to the 2000-17 Omron PLC interface to a Xycom 2000 series terminal

The specific commands that allow communications are described in Section A 5 and listed below for reference

GET
GETER
INIT
PUT
PUTMODE

The 2000 Workstation supports both Single Link and Multi Link communications configurations The Single-Link configuration is for a single PLC connected to a 2000 Terminal The Multi-Link configuration allows up to 31 PLCs to be connected to a 2000 Terminal Both configurations support multiplex handshaking These configurations can be changed through the Omron Port Configuration Menu

The direct connect supports the OMRON single-frame format The PLC interface limits the maximum size of a single-frame message to 131 characters This limits the operator to a maximum of 29 channels that can be read or written at one time

A 2 CABLING THE TERMINAL TO THE PLC

The Omron PLC connects to the 9-pin primary serial port of the 2000 Workstation via a non-standard cable The 2000 Workstation can work on either RS-232 or RS-422 cable You must construct the connectors, according to the pinouts shown in Figures A-1 and A-2

If you are using an RS-232 connection, set jumpers J13 through J20 to B

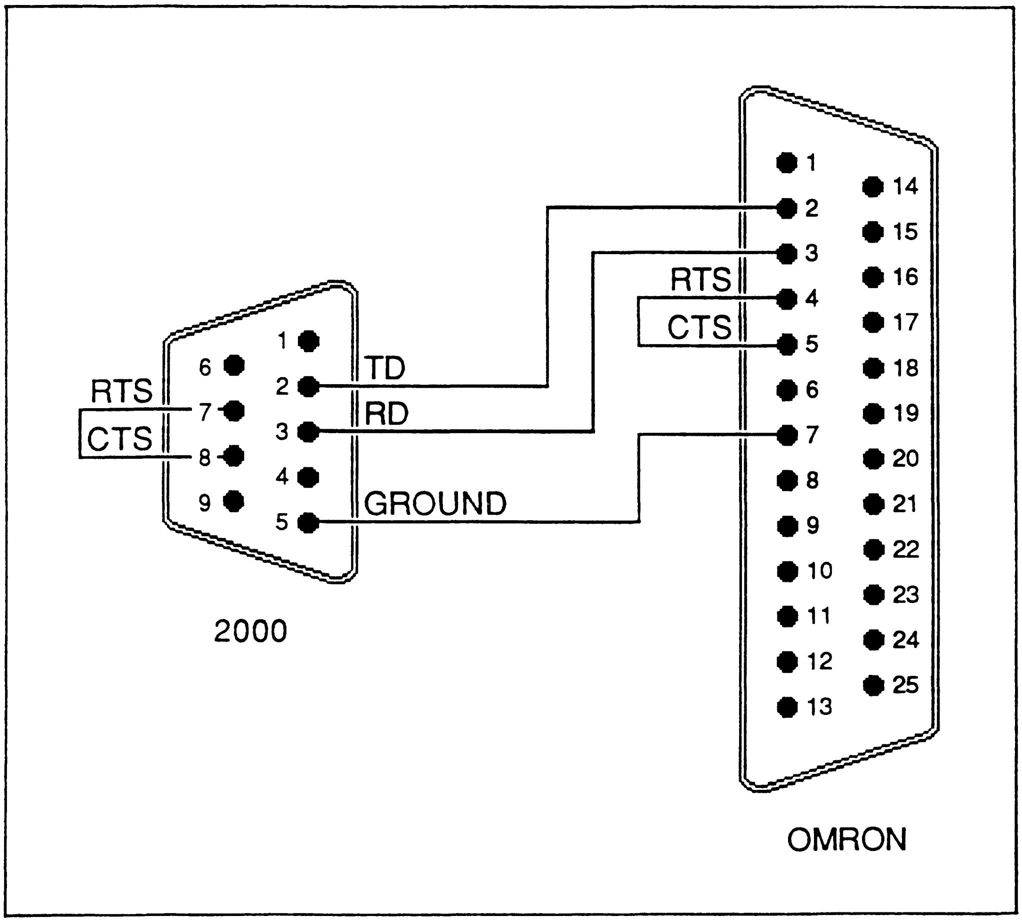


Figure A-1 RS-232 Connection to Omron PLC

If you are using an RS-422 connection, set jumpers J13 through J20 to A

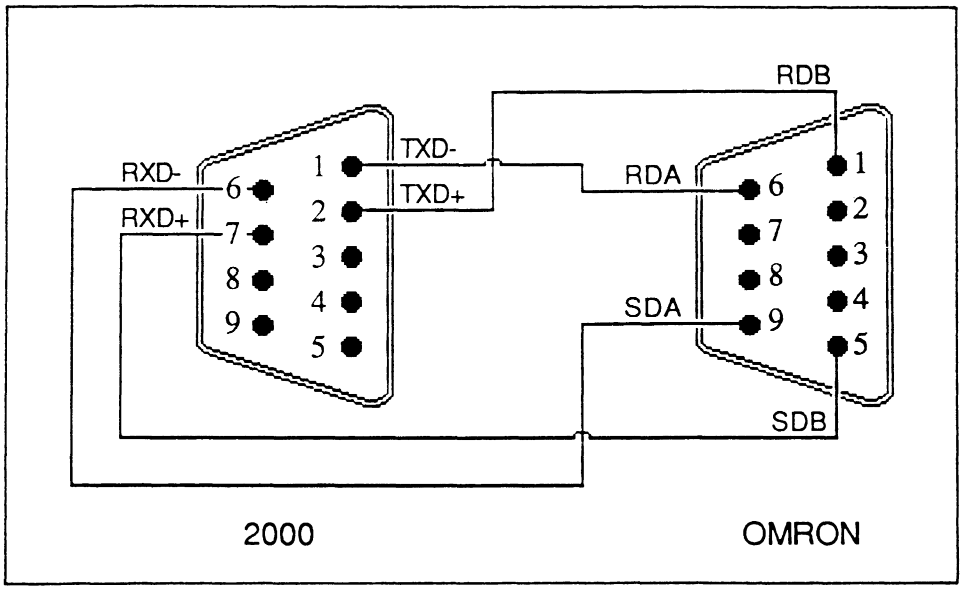


Figure A-2 RS-422 Connection to Omron PLC

A 3 CONFIGURATION MENU

The configuration menu is called up from the Main Menu (discussed in Chapter 3) The OMRON menu is shown in Figure A-3

```
— OMRON Port Configuration Menu (P) —

6  Baud       -  1=300 2=600 3=1200 4=2400 5=4800 6=9600 7=19 2K
1  Data Bits -  1=8 Data Bits    2=7 Data Bits
1  Parity     -  1=Odd Parity    2=Even Parity
1  1=Multi Link ------------------------------ 0=Single Link
0  1=Enable Multiplex Handshaking ----- 0=Disable Multiplex Handshaking



Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW>
Use values 0 through 9
"C" for next configuration menu, <RET> or <ENTER> to quit.
```

Figure A-3 Omron Port Configuration Menu

Baud. The baud rate of the channel should be set to match that of the PLC

Data Bits. The data bit option should be set to match that of the PLC

Parity. This option sets the parity used in communication to the PLC to odd or even, it should be set to match the parity of the PLC

Multi/Single Link. The Single Link configuration signifies that only one PLC is connected to your 2000 Terminal, the Multi Link configuration allows for up to 31 PLCs connected to the 2000

Multiplex Handshaking. This option enables or disables multiplex handshaking Multiplex handshaking is used when a hardware multiplexing box is connected between the primary serial port of the 2000 and the PLC The multiplex box allows multiple 2000 units to connect to a single PLC Data transmissions from the 2000 to the PLC are controlled by the RTS and CTS lines connected between the 2000 and the multiplex box The 2000 firmware automatically controls the RTS/CTS lies when this option is enabled

NOTE

If you use the multiplex handshaking, the CTS and RTS signals must be connected as specified by your particular hardware

A 4 DIAGNOSTICS MENU

The Diagnostics Menu is accessed from the Main Menu (see Chapter 3) The Diagnostics Menu specific to the 2000-17 is shown in Figure A-4 below

-- Diagnostics --

1) Complete Test
2) Continuous Test
3) RAM
4) ROM Checksum
5) EPROM Test
6) Real Time Clock Test
7) RS-232 Serial Loopbasck
8) RS-485/Multidrop Serial Loopback
9) Printer Port Test
A) Parallel Input Test
B) Matrix Keyboard Loopback Test
C) Beeper Test
D) Battery Test
E) Dipswitch Test
F) Character Attributes
G) CRT Crosshatch Pattern
H) CRT Brightness Pattern
I) Touch Screen

<ESC> or <ENTER> to quit

Figure A-4 Diagnostics Menu

A 5 COMMUNICATION STATUS REGISTER

The status value is placed in the OIL register as specified by the command line parameter Sreg The most significant byte (MSB) contains the response code returned from the OMRON HLU The least significant byte (LSB) contains Xycom internal error bits The first four bits are unused and set to zero

Table A-1 below describes the condition corresponding to each bit set in the LSB of the status register

Table A-1 Xycom Terminal Status Codes

LSB Bit #	Status From Xycom Terminal
0-3	Not Used
4	Transmit Error
5	Receive Error
6	FCS
7	Timeout

Table A-2 shows the response codes returned by the OMRON HLU

Table A-2 OMRON Status Register Response Codes

Hex Value of MSB	Omron Status Response
00	Normal completion
01	Not executable in RUN mode
02	Not executable in MONITOR mode
03	Not executable with PROM mounted
04	Address over (data overflow)
0B	Not executable in PROGRAM mode
0C	Not executable in PROM mounted
0D	Not executable in LOCAL mode
10	Parity error
11	Framing error
12	Overrun
13	FCS error
14	Format error (parameter length error)
15	Entry number data error (parameter error, data code error, data length error)
16	Instruction not found
18	Frame length error
19	Not executable (unexecutable error clear, non-registration of I/O table, etc)
20	I/O table generation impossible (unrecognized Remote I/O Unit, channel over, duplication of Optical Transmitting I/O Unit)
A0	Aborted because of parity error in transmit data
A1	Aborted because of framing error in transmit data
A2	Aborted because of overrun in transmit data
A4	Aborted because of format error in transmit data
A5	Aborted because of entry number data error in transmit data
A8	Aborted because of frame length error in transmit data
B0	Not executable because program area is not 16 Kbytes

A 6 COMMANDS

This section describes the OIL commands specific to the 2000-17 Omron interface

GETER	Read and Clear HLU error conditions
INIT	Abort current operations and re-initialize the HLU
GET	Read from PLC register areas
PUT	Write to PLC register areas
PUTMODE	Change PLC mode

Register areas include

IR	Input/Output Relay area channels
HR	Holding Relay area channels
AR	Auxiliary Relay area channels
DM	Data Memory area channels
TC	Timer/Counter area channels

> **NOTE**
> The direct connect does internal range checking, and will not allow writes to output relays

A 6 1 READ AND CLEAR HOST LINK UNIT ERROR CONDITIONS (GETER)

Syntax

GETER erclr,dreg,sreg,[unitno]

where

erclr	=	0 for Error is not cleared, 1 for Error is cleared
dreg	=	starting register in which data will be stored
sreg	=	communication status register
unitno	=	PLC's Unit Number This value is only used if a multi-link system is in use

Example

GETER 0,#90,#70

This example checks to see if an error has occurred without clearing it The result will be stored in the next two words starting at OIL register #90 Register #70 will contain the communication status A single-link system is used

Table A-3 on the following pages shows the errors associated with each bit set in the data register

Table A-3 Error Response Codes

First Word

Bit #			Error
15			FALS (CPU stops)
14			End Instruction Missing (FO)/Program Error (F3)
13			Host Link Unit transmission error
12			RTI Instruction Error
11			PC Link Error
10			I/O Bus Error (CO to 4)
9			JMP Instruction Error (F2)
8			Memory Error (F1)
7			FAL Error
6			Battery Failure (F7)
5			Duplex Bus Error
4	3	2	
0	0	0	CPU Rack
0	0	1	Expansion Rack 1
0	1	0	Expansion Rack 2
0	1	1	Expansion Rack 3
1	0	0	Expansion Rack 4
1	0	1	Expansion Rack 5
1	1	0	Expansion Rack 6
1	1	1	Expansion Rack 7
1	2		Data from I/O Bus
0	0		Group 1 (control signal error)
0	1		Group 2 (data bus failure)
1	1		Group 3 (address bus failure)

Table A-3 Error Response Codes (continued)

Second Word

Bit #	Error
15	Remote I/O Error (B0 to 3)
14	I/O setting error (E0)
13	Indirect JMP Instruction Error (F9)
12	Scan time over (F8/I/O Unit over (E1)
11	I/O verify error (E7)
10	
9	DM CH error (F8)
8	
7	FAL, FALS number 0 $0
6	
5	9 $9
4	
3	FAL, FALS number 0 $0
2	
1	9 $9
0	

A 6.2 **ABORT CURRENT OPERATIONS AND RE-INITIALIZE HOST LINK UNIT (INIT)**

Syntax

INIT

Example

INIT

This example will send an abort message to abort the process being performed by the HLU and enable reception of the next command It will then initialize the transmission control procedure of all the PLCs connected to the host computer

A 6 3 READ FROM PLC REGISTER AREAS (GET)

<u>Syntax</u>

GET area,bch,len,dreg,sreg,[unitno]

where

area = indicates which area to read Options are 1, 2, 3, 4, 5 (which stand for IR, HR, AR, DM, and TC respectively)

bch = beginning OMRON channel within area to start reading from The driver will ensure that valid ranges for each area are entered The valid ranges are listed below

IR - 000-246

HR - 00-99

AR - 00-22

DM - 0000-1999

TC - 000-511

len = the number of OMRON channels to read

dreg = starting OIL register number into which received data will be stored Data must be in hexidecimal format, except when using the TC area where the data is either a 1 or 0

sreg = communication status register

unitno = PLC's Unit Number This value is only used if a multi-link system is in use

Example

GET 4,40,2,#50,#90,2

This command will read two channels of data starting from channel 40 of the DM (Data Memory) area The data will be stored in registers #50 and #51 Register #90 will contain the communication status A multi-link system is used

A 6 4 WRITE TO PLC REGISTER AREAS (PUT)

Syntax

PUT area,bch,len,dreg,sreg,[unitno]

where

area	=	indicates which area to write Options are 1, 2, 3, 4, 5 (which stand for IR, HR, AR, DM, and TC respectively)
bch	=	beginning OMRON channel to start writing to The driver ensures that valid ranges for each area are entered The valid ranges are listed below IR - 30-49 and 232-246 Channels 0-29 and 50-231 are used as outputs HR - 00-99 AR - 07-22 DM - 0000-0999 TC - 000-511
len	=	number of OMRON data channels to write
dreg	=	starting OIL register to take data from
sreg	=	communication status register
unitno	=	PLC's Unit Number This value is only used if a multi-link system is in use

The direct connect does internal range checking and will not allow writes to output relays

Example

PUT 3,7,2,#44,#90,2

This example writes two OIL registers (#44 and #45) to the AR area beginning at channel 1 OIL register #90 will contain the communication status Multi-link system is used

> **NOTE**
> The PLC must be in monitor mode if you want to write any data into the PLC

A 6 5 **Change Mode (PUTMODE)**

Syntax

PUTMODE mode,sreg,[unitno]

where

mode	=	indicates whether to set PLC to RUN mode or Monitor mode 0 = Monitor mode 1 = Run mode
sreg	=	communication status register
unitno	=	PLC's Unit Number This value is only used if a multi-link system is in use

This command allows you to change the PLC's mode

Example

PUTMODE 0,#20,1

This example puts PLC number 1 in Monitor mode If the PLC is in Run mode, you must issue a PUTMODE command to put the PLC into Monitor mode so you can write data to it You can then use the PUTMODE command to reset the PLC to Run mode to prevent any further writes The above example assumes a MultiLink system

Rev. H

2000-18

Mitsubishi MELSEC-A PLC Interface

Printed in the United States of America
Part Number 96760-018

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORD

Revision	*Description*	*Date*
A	Appendix Released	4/91

Address comments concerning this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number ***96760-018A***

A 1 INTRODUCTION

This chapter contains information specific to the 2000-18 Mitsubishi MELSEC-A PLC interface The protocol used to communicate with the Mitsubishi MELSEC-A is full-duplex, asynchronous, RS-232C or RS-422

The specific commands that allow communication are fully described in Section A 5 and listed below for reference

GETDB	Get PLC device bits
GETDW	Get PLC device word
GETBUF	Get buffer memory values
GETCPU	Get CPU type
GETPAR	Get parameter memory
GETSFU	Get Special Function Unit buffer memory
PUTDB	Put PLC device bits
PUTDW	Put PLC device word
PUTBUF	Put buffer memory values
PUTCPU	Put CPU run/stop
PUTPAR	Put parameter memory
PUTSFU	Put Special Function Unit buffer memory

This direct connect communicates with the Mitsubishi MELSEC-A PLC using the AJ71C24 unit Because multiple PLC stations can be networked to Xycom's 2000-18, the specific station number must be specified in the direct connect commands

> **NOTE**
> AJ71C24 refers to the Mitsubishi Computer Link Unit and PC CPU refers to the Internal Mitsubishi PLC CPU (A1CPU, A2CPU, or A3CPU)

The 2000-18 direct connect can access data in the following PLC memory areas

- Device memory (bit and word devices)
- Buffer memory (AJ71C24 - PC CPU)
- PC CPU (type read, run/stop request)
- Parameter memory (PC CPU)
- Special Function Unit (SFU) buffer memory

A 2 CABLING THE TERMINAL TO THE PLC

The Mitsubishi MELSEC-A can connect to the Xycom 2000 primary port via RS-232C or RS-422 The RS-232C connection is shown in Figure A-1 below, while the RS-422 connection is shown in Figure A-2 on the following page

> **NOTE**
> Before connecting the units, configure the 2000 primary port to RS-232C by positioning jumpers J13 through J20 to B

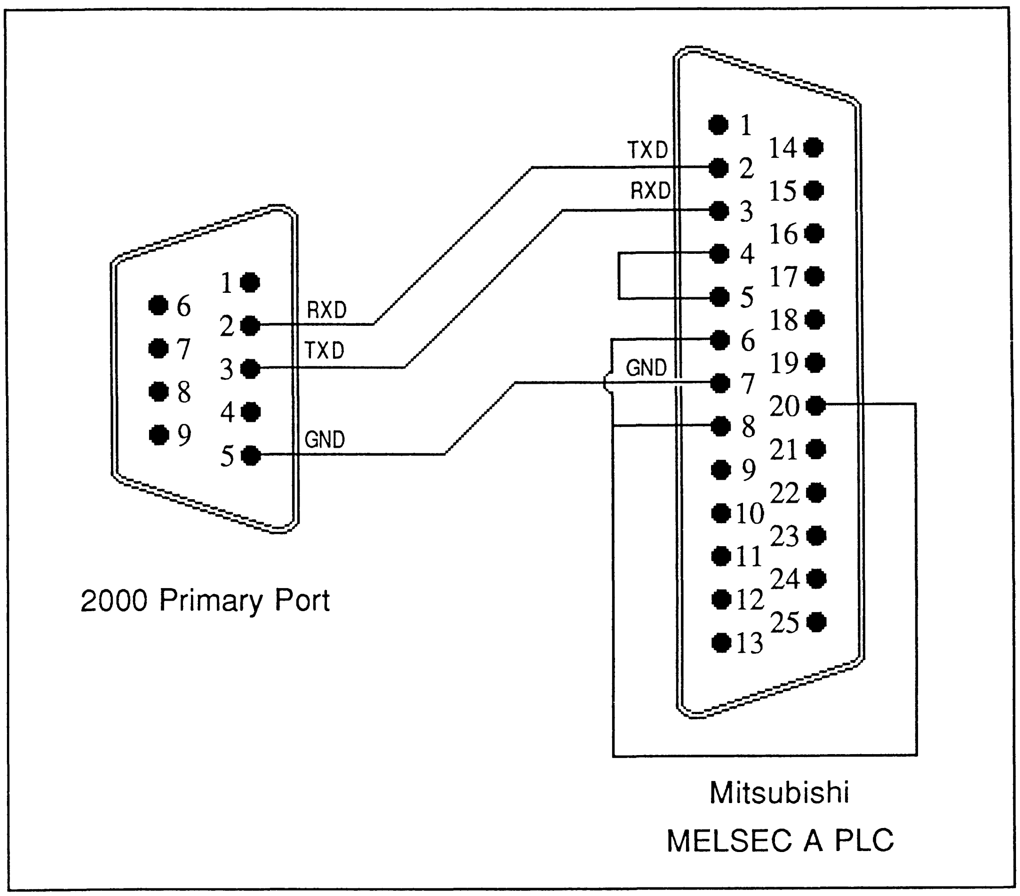


Figure A-1 RS-232C Connections

NOTE
Before connecting the units, configure the 2000 primary port to RS-422 by positioning jumpers J13 through J20 to A

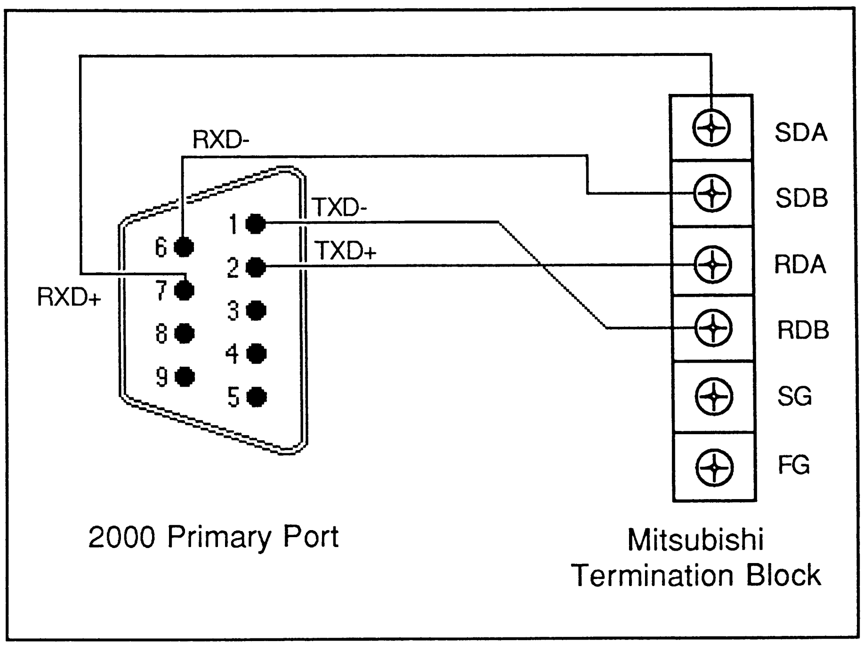


Figure A-2 RS-422 Connections

A 3 CONFIGURATION MENU

The Configuration Menu is called up from the Main Menu (discussed in chapter 3) This direct connect uses 1 start bit and 1 stop bit and supports communication with the 2000 primary serial port using RS-232C or RS-422 The Mitsubishi configuration menu is shown in Figure A-2 below

```
               -- Mitsubishi MELSEC-A PLC Configuration Menu (P) --

   6  Baud -   1=300  2=600  3=1200  4=2400  5=4800  6=9600  7=19 2K
   1  Parity -  0=No Parity   1=Odd Parity    2=Even Parity
   1  1=8 Data Bits ------- 0=7 Data Bits
   1  Format of Control Protocol (1-4)
   0  Wait for message delay (0-9, unit=10 ms)
   0  1=Enable Multiplex Handshaking -- 0=Disable Multiplex Handshaking




   Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW>
   Use values 0 through 9
   "C" for next configuration menu, <RET> or <ENTER> to quit
```

Figure A-3 Configuration Menu

Baud The baud rate of the channel should be set to match that of the PLC

Parity This option will set the parity used in communication on the Mitsubishi to none, odd, or even The type selected should match the other communication device(s)

Data Bits This option will select 7 or 8 data bits per character This should match the data bits of the other communication device(s)

Control Protocol You can set this option to the Mitsubishi AJ71C24 specific protocol format (1, 2, 3, or 4) See the Mitsubishi MELSEC-A User Manual for specific information

Message Delay This option sets the message delay which is the minimum waiting time before the PLC sends data after receiving a command from the terminal Each unit is equal to 10 ms, so selecting 1 will set the message delay to 10 ms, 2 sets the delay to 20 ms, and so on

> **NOTE**
> The 2000-18 direct connect does not support the simultaneous selection of 7 bits per character and no parity

Multiplex Handshaking. This option enables or disables multiplex handshaking Multiplex handshaking is used when a hardware multiplexing box is connect between the primary serial port of the 2000 and the PLC The multiplex box allows multiple 2000 units to connect to a single PLC Data transmissions from the 2000 to the PLC are controlled by the RTS and CTS lines connected between the 2000 and the multiplex box The 2000 firmware automatically controls the RTS/CTS lines when this option is enabled

> **NOTE**
> If using multiplex handshaking, the CTS and RTS signals must be connected as specified by your particular hardware

A 4 DIAGNOSTICS MENU

The Diagnostics Menu is accessed from the Main Menu (see Chapter 3) and lists the diagnostic tests that can be performed on the 2000 The Diagnostics Menu specific to the 2000-18 is shown in Figure A-3 below

```
-- Diagnostics --

1) Complete Test
2) Continuous Test
3) RAM
4) ROM Checksum
5) EPROM Test
6) Real Time Clock Test
7) RS-232 Serial Loopback
8) RS-485/Multidrop Serial Loopback
9) Printer Port Test
A) Parallel Input Test
B) Matrix Keyboard Loopback Test
C) Beeper Test
D) Battery Test
E) Dipswitch Test
F) Character Attributes
G) CRT Crosshatch Pattern
H) CRT Brightness Pattern
I) Touch Screen Test

<ESC> or <ENTER> to quit
```

Figure A-4 Diagnostics Menu

NOTE

Option I only appears if the optional touch screen is installed on your unit

See Chapter 3 for explanations of each of the menu choices

A 5 COMMUNICATION STATUS REGISTER

The Communication Status Register is a user-specified OIL register which contains the status about the message that was just transferred The Most Significant Byte (MSB) contains information about error codes from the 2000-18 direct connect software The Least Significant Byte (LSB) contains information about AJ71C24 NAK error codes issued

On a AJ71C24 NAK response, the MSB communication status bit 1 is reported set and the value of the LSB will clarify the reason for the NAK response

COMMUNICATION STATUS WORD		ERROR BYTE DEFINITION	
MSB	**LSB**		
Bit 0		Transmit error	(should never occur)
Bit 1		Receive error	(unexpected character or no char)
Bit 2		Time out error	(expected 5 second response)
Bit 3		Parity error	(sent/receive mismatch)
Bit 4		Enquiry error	(no AJ71C24 response)
	(Refer to AJ71C24 User Manual for details)		
	00H	Invalid access	(made during run)
	01H	Parity	(sent/receive mismatch)
	02H	Sum check	(sent/receive mismatch)
	03H	Protocol error	(communication protocol not valid)
	04H	Framing error	(data/stop bit mismatch)
	05H	Overrun error	(previous data not yet received)
	06H	Character area	(mode set/character mismatch)
	07H	Character error	(illegal character/control code)
	08H	PC access error	(buffer to PC com err)
	(Generated by AJ71C24 to PC CPU access)		
	10H	PC number error	(station # doesn't exist, use FFH)
	11H	Mode error	(invalid request sent)
	12H	SFU spec error	(no SFU at I/O location)
	18H	Remote error	(run/stop impossible)
	21H	Memory access to SFU	(control bus error/unit faulty)

A 6 COMMANDS

The 2000-18 direct connect commands communicate between the OIL registers and the PLC's device memory bits, words, buffer memory, PC CPU, parameters, and Special Function Units Bit/byte values are right justified in the 16-bit OIL registers This section contains descriptions and examples of these commands

A 6 1 GET PLC DEVICE BITS (GETDB)

Syntax

GETDB stn,type,addr,lngth,dreg,sreg

where

stn is the station number of the PLC (0-31)

type is the kind of device (bit value)

1 = X input	(000H-7FFH)
2 = Y output	(000H-7FFH)
3 = M relay	(0000-2047)
4 = L relay	(0000-2047)
5 = B relay	(000H-3FFH)
6 = F annunciator	(000-255)
7 = Special relay	(000-255)
8 = Timer contract	(000-255)
9 = Timer coil	(000-255)
10= Counter contact	(000-255)
11= Counter coil	(000-255)

addr is the device memory address (the range depends on type value above)

lngth is the length of the block (1-256)

dreg is the destination OIL register

sreg is the communication status register

This command reads PLC device memory data bits into OIL registers (1 bit per register)

Example

GETDB 0,1,0,4,#20,#19

This command reads the first four X bits from PLC station 0 into OIL registers 20 to 23 Register 19 will contain the communication status

A62 GET PLC DEVICE WORD (GETDW)

Syntax

GETDW stn,type,addr,lngth,dreg,sreg

where

stn is the station number of the PLC (0-31)

type is the kind of device (1-11 = 16 bit values/word, 12-17 = 16-bit word value)

1 = X inputs word	(00H-7FH)
2 = Y outputs word	(00H-7FH)
3 = M relays word	(000-127)
4 = L relays word	(000-127)
5 = B relays word	(00H-3FH)
6 = F annunciators word	(00-15)
8 = Timer contracts word	(00-15)
9 = Timer coils word	(00-15)
10 = Counter contacts word	(00-15)
11 = Counter coils word	(00-15)
12 = Timer word	(000-255)
13 = Counter word	(000-255)
14 = Data register D word	(0000-1023)
15 = Link register W word	(000H-3FFH)
16 = File register R word	(0000-8191)
17 = Special register word	(000-255)

addr is the device memory address (the range depends on the type value above)

lngth is the length of the block (1-32 for types 1-11, 1-64 for types 12-17)

dreg is the starting destination OIL register

sreg is the communication status register

This command will read PLC device memory values and store them in a block of OIL registers (one word per register) If you specify a length of more than 1 word, you must allow sufficient registers to contain the data After the first word is read into the starting register, data will flow into successive registers, overwriting any previous contents

Example 1

GETDW 1,1,0,3,#17,#20

This command will read 3 words from PLC station 1 into OIL registers 17, 18 and 19 Register 20 will contain the communication status

Example 2

GETDW 1,1,0,1,#17,#18

This command will read the first 16 X bits from PLC station 1 into OIL register 17 Register 18 will contain the communication status

A 6 3 GET BUFFER MEMORY VALUES (GETBUF)

Syntax

GETBUF stn,type,addr,lngth,dreg,sreg

where

stn is the station number of the PLC (0-31)

type is the type of communication
0 = Direct
1 = Indirect

addr is the memory device address
0-07FFH 16 bit values, for type = 0
1000H-1FFFH 8 bit values, for type = 1

lngth is the length of the block (1-64)

dreg is the destination OIL register

sreg is the communication status register

This command will read PLC buffer memory values and store them in a block of OIL registers (1 value per register)

Example

GETBUF 2,0,&7FF,1,#17,#18

Using direct communication, this command will read the last 16-bit value from PLC station 2 to OIL register 17 Register 18 will contain the communication status

A 6 4 GET PC CPU TYPE (GETCPU)

Syntax

GETCPU stn,dreg,sreg

where

stn is the station number of the PLC (0-31)

dreg is the destination OIL register

sreg is the communication status register

This command reads the PC CPU type (16-bit word value) from the PLC and stores the name (A1, A2, A3) in the OIL destination register

Example

GETCPU 3,#17,#18

This command reads the PC CPU type from PLC station 3 into register 17 Register 18 will contain the communication status

A 6 5 GET PARAMETER MEMORY (GETPAR)

Syntax

GETPAR stn,addr,lngth,dreg,sreg

where

stn is the station number of the PLC (0-31)

addr is the memory device address (0-BFFH)

lngth is the length of the block (1-128)

dreg is the destination OIL register

sreg is the communication status register

This command will read 8-bit values from the 3 Kbytes of PC CPU parameter memory and store them in a block of OIL registers (1 byte per register)

Example

GETPAR 4,0,128,#100,#228

This command will read 8-bit values from address 0 in PLC station 4 and store them in a 128 byte block in registers 100 to 227 Register 228 will contain the communication status

A 66 GET SPECIAL FUNCTION UNIT BUFFER MEMORY (GETSFU)

Syntax

GETSFU stn,type,addr,lngth,dreg,sreg

where

stn	is the station number of the PLC (0-31)
type	is the Special Function Unit Number (0-255, byte)
addr	is the memory device address (depends on Special Function Unit)
lngth	is the length of block (1-128 bytes)
dreg	is the destination OIL register
sreg	is the communication status register

This command will read PLC Special Function Unit buffer memory and store values in a block of OIL registers (1 byte per register) The format of the data values depends on the type and addr parameters For more information, refer to the MELSEC-A User's Manual for the Computer Link Module AJ71C24

Example

GETSFU 0,0,0,1,#20,#18

This command will read the special function unit buffer memory of PLC station 0, unit 0, address 0, in one block to OIL register 20 Register 18 will contain the communication status

A 67 PUT PLC DEVICE BITS (PUTDB)

Syntax

```
PUTDB stn,type,addr,lngth,srcreg,sreg
```

where

stn is the station number of the PLC (0-31)

type is the kind of device (bit value)

1 = X input	(000H-7FFH)
2 = Y output	(000H-7FFH)
3 = M relay	(0000-2047)
4 = L relay	(0000-2047)
5 = B relay	(000H-3FFH)
6 = F annunciator	(000-255)
7 = Special relay*	(000-255)
8 = Timer contract	(000-255)
9 = Timer coil	(000-255)
10= Counter contact	(000-255)
11= Counter coil	(000-255)

NOTE
*Special relays have pre-defined meanings and should not be written to

addr is the memory device address (the range depends on the type value above)

lngth is the length of block (1-160)

srcreg is the OIL data register to read from

sreg is the communication status register

This command writes PLC device memory data bits from OIL registers (1 bit per register)

Example

```
PUTDB 6,1,0,1,#17,#18
```

This example writes the first X bit from OIL register 17 to PLC station 6 Register 18 will contain the communication status

A 68 PUT PLC DEVICE WORD (PUTDW)

Syntax

PUTDW stn,type,addr,lngth,srcreg,sreg

where

stn	is the station number of the PLC (0-31)	
type	is the kind of device (1-11 = 16-bit values/word, 12-17 = 16-bit word value)	
	1 = X inputs word	(00H-7FH)
	2 = Y outputs word	(00H-7FH)
	3 = M relays word	(000-127)
	4 = L relays word	(000-127)
	5 = B relays word	(00H-3FH)
	6 = F annunciators word	(00-15)
	7 = Special relays word*	(00-15)
	8 = Timer contracts word	(00-15)
	9 = Timer coils word	(00-15)
	10 = Counter contacts word	(00-15)
	11 = Counter coils word	(00-15)
	12 = Timer word	(000-255)
	13 = Counter word	(000-255)
	14 = Data register D word	(0000-1023)
	15 = Link register W word	(000H-3FFH)
	16 = File register R word	(0000-8191)
	17 = Special register word*	(000-255)

NOTE

*Special relays and special registers have pre-defined meanings and should not be written to

addr	is the memory device address (the range depends on the type value above)
lngth	is the length of the block (1-10 for types 1-11, 1-64 for types 12-17)
srcreg	is the OIL data register to read from
sreg	is the communication status register

This command writes PLC device memory word values from a block of 16-bit OIL registers

Example

PUTDW 1,1,0,1,#17,#18

This command will write the 16-bit data value from OIL register 17 to the first 16 X bits in PLC station 1 Register 18 will contain the communication status

A 6 9 PUT BUFFER MEMORY VALUES (PUTBUF)

Syntax

PUTBUF stn,type,addr,lngth,srcreg,sreg

where

stn	is the station number of the PLC (0-31)
type	is the type of communication 0 = Direct 1 = Indirect
addr	is the memory device address 0-07FFH 16-bit values, for type=0 1000H-1FFFH 8-bit values, for type=1
lngth	is the length of the block (1-64)
srcreg	is the OIL data register to read from
sreg	is the communication status register

This command writes PLC buffer memory values from a block of OIL registers (1 value per register) to the target PLC

Example

PUTBUF 8,0,&7FF,1,#17,#18

Using direct communication, this command will write the last 16-bit value from a 1 Kbyte block in OIL register 17 to PLC station 8 Register 18 will contain the communication status

A 6 10 **PUT CPU RUN/STOP COMMAND (PUTCPU)**

Syntax

PUTCPU stn,com,sreg

where

stn	is the station number of the PLC (0-31)
com	is the command 1 = Run 0 = Stop
sreg	is the communication status register

This command writes a PC CPU run or stop command to the PLC station

Example

PUTCPU 9,0,#18

This command will stop PLC station 9 Register 18 will contain the communication status

A 6 11 **PUT PARAMETER MEMORY (PUTPAR)**

Syntax

PUTPAR stn,addr,lngth,srcreg,sreg

where

stn	is the station number of the PLC (0-31)
addr	is the memory device address (0-BFFH)
lngth	is the length of the block (1-128)
srcreg	is the OIL data register to read from
sreg	is the communication status register

This command will write 8-bit values to the 3 Kbytes of PC CPU parameter memory from a block of OIL registers (1 byte per register)

Example

PUTPAR 10,0,128,#100,#228

This command will write a 128 byte block from registers 100-227 to PLC station 10's parameter memory starting at address 0 Register 228 will contain the communication status

> **NOTE**
> The PC CPU must be stopped before writing to parameter memory

A 6 12 PUT SPECIAL FUNCTION UNIT BUFFER MEMORY (PUTSFU)

Syntax

PUTSFU stn,type,addr,lngth,srcreg,sreg

where

stn is the station number of the PLC (0-31)

type is the type ("Special Function Unit Number", 0-255, byte)

addr is the memory device address (depends on "Special Function Unit")

lngth is the length of the block (1-128 bytes)

srcreg is the OIL data register to read from

sreg is the communication status register

This command writes PLC Special Function Unit buffer memory with values from the block of OIL registers (1 byte per register) The format of the data values depends on the type and addr parameters For more information, refer to the User Manual for the MELSEC-A Computer Link Module AJ71C24

Example

PUTSFU 0,0,0,2,#20,#18

This command will write the special function unit buffer memory value in OIL register 20, to PLC station 0, unit 0, address 0 in a 2 Byte block Register 18 will contain the communication status

A 7 PROGRAM EXAMPLE

The following is a sample OIL program which reads and writes PLC values, using each of the 2000-18 direct connect user interface commands (This assumes the OIL registers used are loaded with the PLC values to write)

```
PUTDB 0,1,&3,1,#10,#70
" Data sent & cs     = "   tr,#10,sc(hhhh )  tr,#70,sc(hhhh)  nl
GETDB 0,1,&3,1,#11,#70
" Received data & cs = "   tr,#11,sc(hhhh )  tr,#70,sc(hhhh)  nl

PUTDW 0,1,0,1,#20,#70
" Data sent & cs     = "   tr,#20,sc(hhhh )  tr,#70,sc(hhhh)  nl
GETDW 0,1,0,1,#21,#70
" Received data & cs = "   tr,#21,sc(hhhh )  tr,#70,sc(hhhh)  nl

PUTBUF 0,0,&7ff,1,#30,#70
" Data sent & cs     = "   tr,#30,sc(hhhh )  tr,#70,sc(hhhh)  nl
GETBUF 0,0,&7ff,1,#31,#70
" Received data & cs = "   tr,#31,sc(hhhh )  tr,#70,sc(hhhh)  nl

PUTCPU 0,0,#70
" STOP command cs    = "   tr,#70,sc(hhhh)
" PLC STOP  Press any key to continue   "   tr,kb(l),#90  nl

PUTCPU 0,1,#70
" RUN command cs     = "   tr,#70,sc(hhhh)
" PLC RUN   Press any key to continue   "   tr,kb(l),#90  nl

GETCPU 0,#41,#70
" MELSEC PLC CPU type & CS  = "   tr,#41,sc(cc )  tr,#70,sc(hhhh )  nl

PUTPAR 0,0,1,#50,#70
" Data sent & cs     = "   tr,#50,sc(hh )  tr,#70,sc(hhhh)  nl
GETPAR 0,0,1,#51,#70
" Received data & cs = "   tr,#51,sc(hh )  tr,#70,sc(hhhh)  nl

PUTSFU 0,8,&80,1,#60,#70
" Data sent & cs     = "   tr,#60,sc(hh )  tr,#70,sc(hhhh)  nl
GETSFU 0,8,&80,1,#61,#70
" Received data & cs = "   tr,#61,sc(hh )  tr,#70,sc(hhhh)

stop
```

2000-19

Texas Instruments Series 305 PLC Interface

Printed in the United States of America
Part Number 96760-019

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORD

Revision	*Description*	*Date*
A	Appendix Released	2/91

Address comments concerning
this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number: ***96760-019A***

A 1 INTRODUCTION

This chapter contains information specific to the 2000-19 Texas Instruments Series 305 PLC interface To communicate with the TI 305, this module uses a full-duplex, asynchronous, point-to-point protocol that closely conforms to ANSI X3 28

The specific OIL commands that allow communication are described in Section A 6 and listed below for reference

GETR	Get register
GETIOS	Get I/O status
PUTR	Put register
PUTIOS	Put I/O status

A 2 CABLING THE WORKSTATION TO THE PLC

The PLC and/or network connects to the 9-pin primary serial port of the 2000 Workstation via a cable The electric interface between the TI 305 PLC and the 2000 Workstation is via RS-232C or RS-422 If using RS-232C, set jumpers J13 through J20 to B, and make the connections shown in Figure A-1

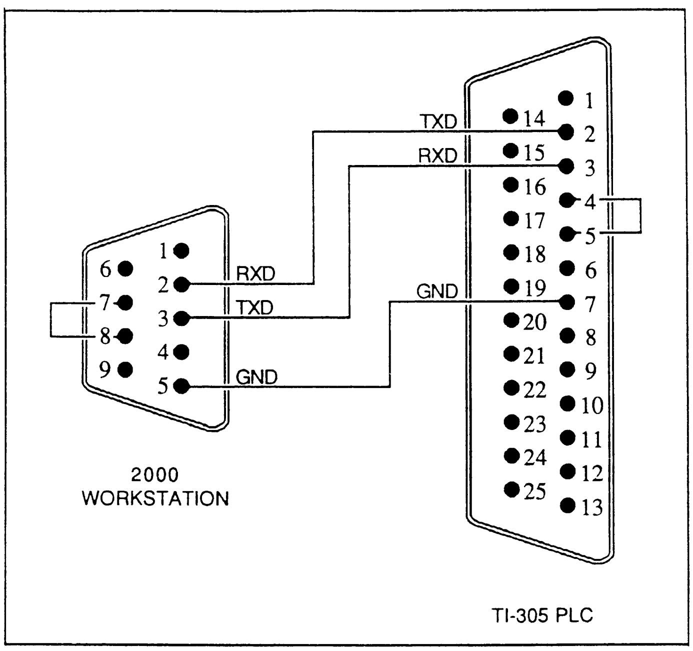


Figure A-1 Connecting to the TI 305 vis RS-232C

If using RS-422, set jumpers J13 to J20 to A, and make the connetions shown in Figure A-2 below

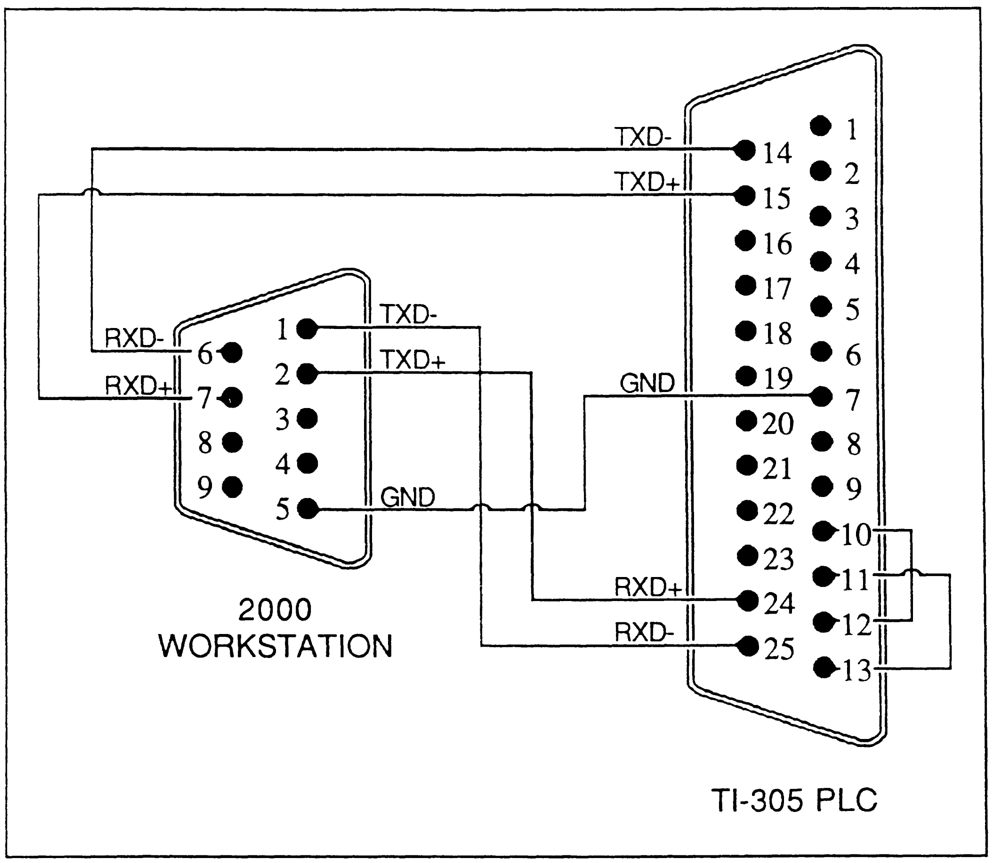


Figure A-2 Connecting to the TI 305 via RS-422

A 3 CONFIGURATION MENU

The Configuration Menu is called up from the Main Menu (see Chapter 3) and is shown below

```
          -- Texas Instruments TI-305 Configuration Menu (P) --

6     Baud - 1=300 2=600 3=1200 4=2400 5=4800 6=9600 7=19 2 Kbyte
00    0=No Parity 1=Odd Parity
01    Source ID# (1-90)
01    Timeout Value (1-20 seconds)
0     1=Enable Multiplex Handshaking  0=Disable Multiplex Handshaking


Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW>
Use values 0 through 9
"C" for next configuration menu, <ESC> or <ENTER> to quit
```

Figure A-2 Configuration Menu

Baud The baud rate of the channel should be set to match that of the PLC

Parity This option sets the parity used in communication to the Texas Instruments 305 PLC to either none or odd The type selected should match the other communication device(s)

Source ID# The source ID number setting in the TI-305 Configuration Menu **must** be identical to the station address of the TI PLC to which the terminal is directly connected Refer to the Texas Instruments user reference material for information on obtaining the station address

Timeout Value The timeout value determines how many seconds (from 1 to 20) the workstation will wait for a response before signaling a timeout

Multiplex Handshaking This option enables or disables multiplex handshaking Multiplex handshaking is used when a hardware multiplexing box is connected between the primary serial port of the 2000 and the PLC The multiplex box allows multiple 2000 units to connect to a single PLC Data transmissions from the 2000 to the PLC are controlled by the RTS and CTS lines connected between the 2000 and the multiplex box The 2000 firmware automatically controls the RTS/CTS lines when this option is enabled

> **NOTE**
> If using multiplex handshaking, the CTS and RTS signals must be connected as specified by your particular hardware

A 4 DIAGNOSTICS MENU

The Diagnostic Menu is accessed from the Main Menu (see Chapter 3 for more information) The Diagnostics Menu specific to the 2000-19 is shown in Figure A-3 below

```
-- Diagnostics --

1) Complete Test
2) Continuous Test
3) RAM Test
4) ROM Checksum
5) EPROM Test
6) Real Time Clock Test
7) RS-232 Serial Loopback
8) RS-485/Multidrop Serial Loopback Test
9) Printer Port Test
A) Parallel Input Test
B) Matrix Keyboard Loopback Test
C) Beeper Test
D) Battery Test
E) Dipswitch Test
F) Character Attributes
G) CRT Crosshatch Pattern
H) CRT Brightness Pattern
I) Touch Screen

<ESC> or <ENTER> to quit
```

Figure A-3 Diagnostics Menu

NOTE

Selection I) Touch Screen only appears on the menu if a touch screen is installed

A 5 COMMUNICATION STATUS REGISTER

The communication status register contains the status about the message that was just transferred The Most Significant Byte (MSB) is not used The Least Significant Byte (LSB) contains information about the message transfer status A zero value in the entire register indicates a successful transfer

If the bit noted is set, the condition indicated is true

Status Register

MSB **NOT USED**

LSB Message Transfer Status

Bit 0 = Transmit Error
Bit 1 = Receive Error
Bit 2 = Timeout
Bit 3 = Parity Error
Bit 4 = Enquiry Error
Bits 5-7 = **NOT USED**

A 6 COMMANDS

This section describes the commands that are specific to the 2000-19 Texas Instruments Series 305

A 6 1 Get Register (GETR)

Syntax

GETR dest,reg,lngth,dreg,sreg

where

dest	is the station address of the destination PLC
reg	is the register in the target PLC from which the data is to be read
lngth	is the number of words to read
dreg	is the destination register number to store data
sreg	is the communication status register

The command GETR reads data from a timer/counter accumulator or registers in the PLC Refer to Section A 5 for the communication status register information

Example

GETR 10,20,10,#30,#20

This command reads 10 timer/counter accumulator values starting at timer/counter 20 of the target device with station address 10 The data returned is placed into registers 30 through 39 Register 20 contains the communication status

A 6 2 **Get I/O Status (GETIOS)**

Syntax

GETIOS dest,addr,lngth,dreg,sreg

where

dest	is the station address of the destination PLC
addr	is the starting address to read
lngth	is the number of bytes to read, starting at addr
dreg	is the destination register number to store data
sreg	is the communication status register

The command GETIOS reads the I/O bits (input/output, internal relay, shift register, and timer/counter) from the PLC Refer to Section A 5 for the communication status register information

Example

GETIOS 1,1,2,#100,#70

This command reads I/O address 0 to 7 and 10 to 17 from the TI 305 and puts the bit patterns into the lower bytes of registers 100 and 101 The status information is stored in register 70

A 6 3 **Put Register (PUTR)**

Syntax

PUTR dest,reg,lngth,srcreg,sreg

where

dest	is the station address of the destination PLC
reg	is the register in the target PLC from which the data is to be written
lngth	is the number of words to write
srcreg	is the source register number to send data from
sreg	is the communication status register

The command PUTR sends data to a timer/counter accumulator or register in the PLC Refer to Section A 5 for the communication status register information

Example

PUTR 15,33,13,#30,#20

This command writes 13 words of data starting at timer/counter accumulator 33 of the target device with station address 15 The data is sent from into registers 30 through 42 Register 20 contains the communication status

A 6 3 **Put I/O Status (PUTIOS)**

Syntax

PUTIOS dest,addr,lngth,srcreg,sreg

where

dest	is the station address of the destination PLC
addr	is the starting address of input/output status to write
lngth	is the number of bytes to write, starting at addr
srcreg	is the source register number to write data from
sreg	is the communication status register

The command PUTIOS writes a byte of I/O (input/output, internal relay, shift register) information to the PLC Refer to Section A 5 for the communication status register information

Example

PUTIOS 1,1,2,#100,#40

This command writes the bit pattern from the lower byte of register 100 to I/O addresses 0 to 7, and the bit pattern from register 101 to I/O addresses 10 to 17 Register 40 contains the communication status

A

2000-20

General Electric Series 90 PLC Interface

Printed in the United States of America
Part Number 96760-020

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

4

XYCOM REVISION RECORD

Revision	*Description*	*Date*
A	Appendix Released	1/91

Address comments concerning this manual to

xycom
Technical Publications Dept
750 North Maple Road
Saline, Michigan 48176

Part Number ***96760-020A***

A 1 INTRODUCTION

This chapter contains information specific to the 2000-20 General Electric Series 90 PLC interface The Xycom OIL commands that allow communication with the General Electric Series 90 are described in Section A 6 and listed below for reference

PUT	Write data to PLC system memory
GET	Read data from PLC system memory
GETFLT	Read fault table
CLRFLT	Clear fault table

A 2 CABLING THE WORKSTATION TO THE PLC

The electrical interface between the General Electric Series 90 PLC and the 2000 Workstation is via RS-485 with separate transmit and receive circuits Because Series 90 is half-duplex, multiple PLCs can be connected to a single 2000-20 Figure A-1 shows the electrical connection from the 2000 to one PLC whereas Figure A-2 on the following page shows the connection to multiple PLCs

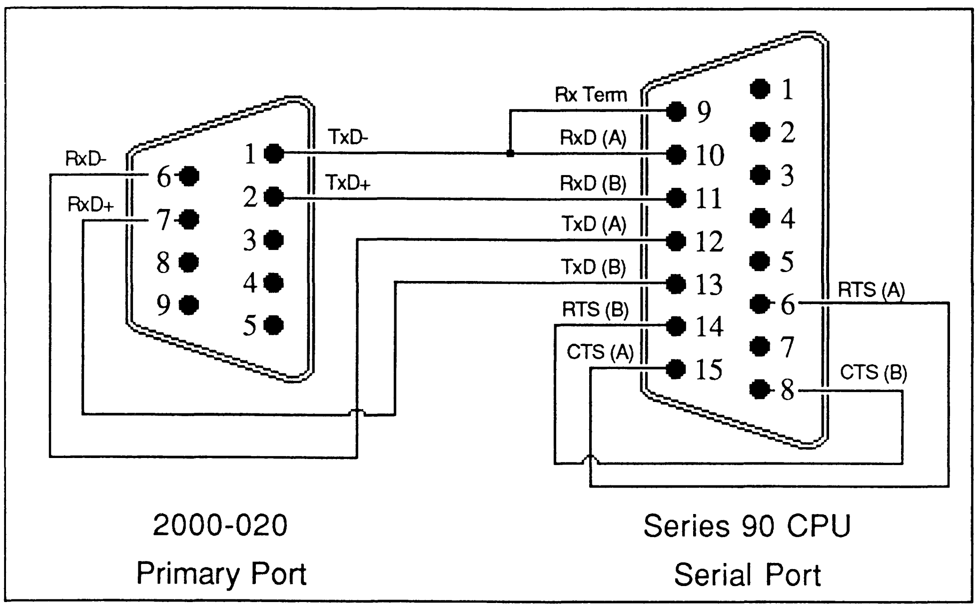


Figure A-1 Point to Point Cabling Diagram

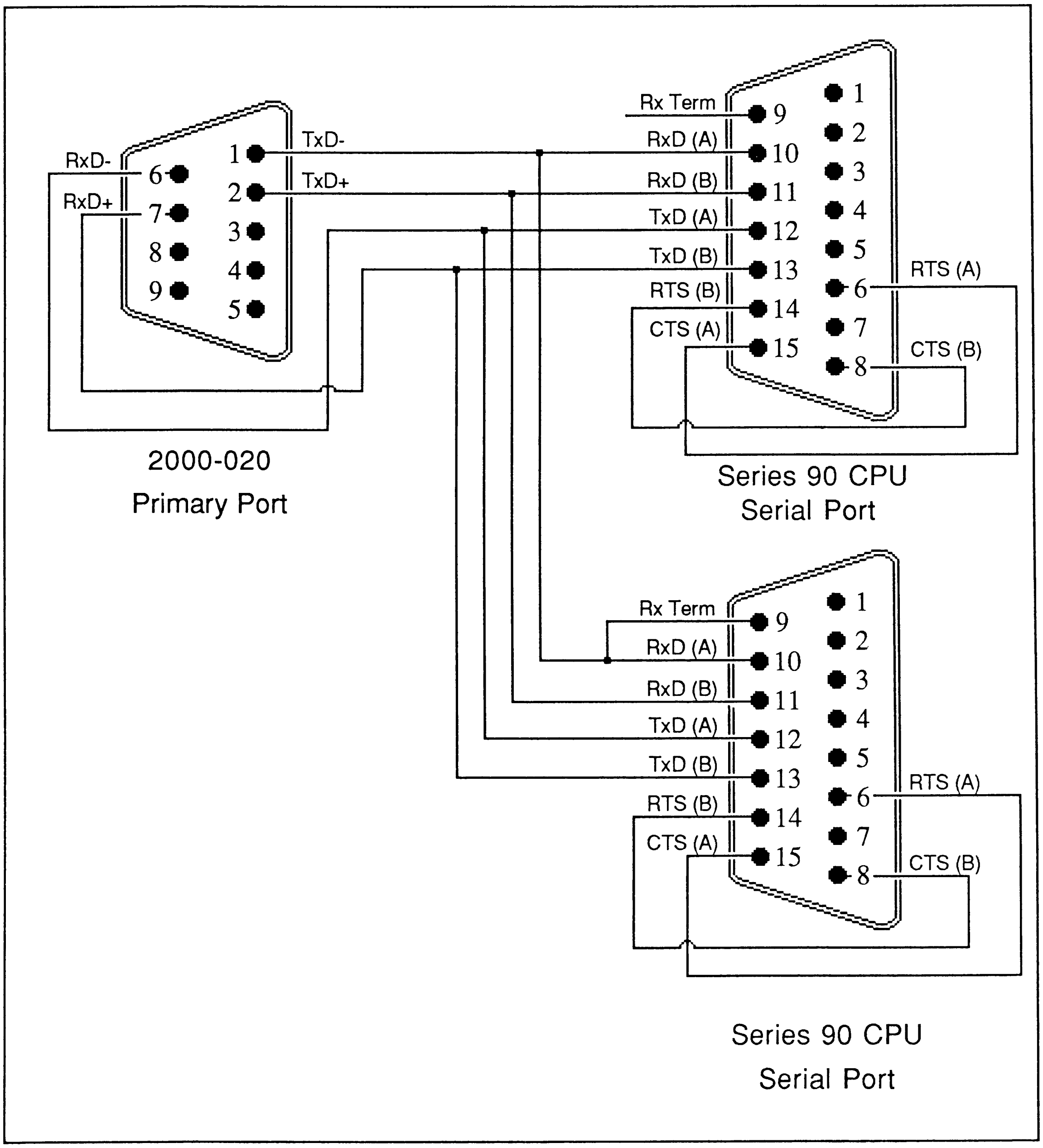


Figure A-2 Multidrop Cabling Diagram

A 3 CONFIGURATION MENU

The Configuration Menu is called up from the Main Menu (refer to Chapter 3 for a description of the Main Menu) The General Electric Series 90 Configuration Menu is shown below

```
-- GE SNP Configuration Menu (P) --

Baud - 1=300  2=600  3=1200  4=2400  5=4800  6=9600  7=19 2K
1 = Enable Parity ----------- 0 = Disable Parity
1 = Odd Parity ------------- 0 = Even Parity
Modem turn around time (0-99 milliseconds)
1 = Enable Multiplex Handshaking -- 0 = Disable Multiplex Handshaking

Use <UP ARROW>, <DOWN ARROW>, <LEFT ARROW>, <RIGHT ARROW>
Use values 0 through 9
"C" for next configuration menu, <ESC> or <ENTER> to quit
```

Baud The baud rate of the channel should be set to match that of the PLC The factory default for the General Electric Series 90 is 19 2 K

Parity This option sets the parity used in communication to the General Electric Series 90 PLC to odd or even The factory default for the General Electric Series 90 is odd

Parity Enabled/Disabled This option enables or disables the type of parity selected above The factory default for the GE Series 90 is enabled

Modem Turn Around Time The modem turn-around time determines how long (in milliseconds) the 2000 delays before beginning each transmission This delay allows using half-duplex modems by giving the modems enough time to turn the line around before the transmission begins The modem turn-around time should be set to 0 unless a half-duplex modem is used

Multiplex Handshaking This option enables or disables multiplex handshaking Multiplex handshaking is used when a hardware multiplexing box is connected between the primary serial port of the 2000 and the PLC The multiplex box allows multiple 2000 units to connect to a single PLC Data transmissions from the 2000 to the PLC are controlled by the RTS and CTS lines connected between the 2000 and the multiplex box The 2000 firmware automatically controls the RTS/CTS lines when this option is enabled

NOTE

If using multiplex handshaking, the CTS and RTS signals must be connected as specified by your particular hardware

A 4 DIAGNOSTICS MENU

The Diagnostics Menu is accessed from the Main Menu (see Chapter 3) The Diagnostics Menu specific to the 2000-20 is shown in Figure A-2 below

```
-- Diagnostics --

1) Complete Test
2) Continuous Test
3) RAM
4) ROM Checksum
5) EPROM Test
6) Real Time Clock Test
7) RS-232 Serial Loopback
8) RS-485/Multidrop Serial Loopback
9) Printer Port Test
A) Parallel Input Test
B) Matrix Keyboard Loopback Test
C) Beeper Test
D) Battery Test
E) Dipswitch Test
F) Character Attributes
G) CRT Crosshatch Pattern
H) CRT Brightness Pattern
I) Touch Screen

<ESC> or <ENTER> to quit
```

Figure A-2 Diagnostics Menu

A 5 COMMUNICATION STATUS REGISTER

The communication status register consists of two words that contain information about the message that was just transferred The first word contains 16 bits of OIL command completion information The second word contains 16 bits of PLC status information

OIL Status Register - First Word

Bit 0 1 = Attach error Could not logically connect to the PLC Usually caused by using the wrong PLC ID

Bit 1 1 = Transmit error An error occurred while attempting to send a message to the PLC

Bit 2 1 = Receive error An error occurred while attempting to receive a message from the PLC

Bit 3 1 = Timed out The PLC did not respond within the allotted time

Bit 4 1 = BCC error detected by the 2000-20

Bit 5 1 = Parity or framing error detected by the 2000-20 Usually caused by a baud rate or parity mismatch between the PLC and the 2000-20

Bit 6 1 = Protocol error The PLC responded with a message undefined by the communications protocol If this error occurs, call Xycom Customer Service

Bit 7 Not used Set to 0

Bit 8 1 = The PLC could not execute the command Usually means that the address or length was out of range

Bit 9 1 = Framing or overrun error detected by the PLC Usually caused by a baud rate, or parity mismatch between the PLC and the 2000-20

Bit 10 1 = BCC or parity error detected by the PLC Usually caused by a baud rate, or parity mismatch between the PLC and the 2000-20

Bit 11 1 = Protocol error The PLC received a message undefined by the communications protocol If this error occurs, call Xycom Customer Service

Bit 12 1 = Privilege violation The access level associated with the password was insufficient for this operation Usually caused by an incorrect password

Bits 13-15 Not used Set to zero

PLC Status Register - Second Word

Bit 0	Oversweep flag (meaningful only in constant sweep mode is active) 1 = Constant sweep value exceeded on most recent sweep 0 = No oversweep condition exists
Bit 1	Constant sweep mode 1 = Constant sweep mode is active 0 = Constant sweep mode is not active
Bit 2	PLC fault entry since last read 1 = PLC fault table has changed since last read by the 2000-20 0 = PLC fault table unchanged since last read
Bit 3	I/O fault entry since last read 1 = I/O fault table has changed since last read by the 2000-20 0 = I/O fault table unchanged since last read
Bit 4	PLC fault entry present 1 = One or more fault entries in PLC fault table 0 = PLC fault table is empty
Bit 5	I/O fault entry present 1 = One or more fault entries in I/O fault table 0 = I/O fault table is empty
Bit 6	Programmer attachment flag 1 = Programmer attachment found 0 = No programmer attachment found
Bit 7	Front panel ENABLE/DISABLE switch setting 1 = Outputs disabled 0 = Outputs enabled
Bit 8	Front panel RUN/STOP switch setting 1 = RUN 0 = STOP
Bit 9	OEM protected bit 1 = OEM protection in effect 0 = No OEM protection
Bit 10-11	Not used
Bit 12-15	PLC state 0 = Run I/O enabled 1 = Run I/O disabled 2 = Stop I/O disabled 3 = CPU stop faulted 4 = CPU halted 5 = CPU suspended 6 = Stop I/O enabled

A 6 COMMANDS

The OIL commands allow you to monitor or change areas of the PLC system memory By means of these OIL commands, the 2000-20 can read from and write to PLC system memory PLC system memory is divided into the following areas

"%I"	Discrete Inputs
"%Q"	Discrete Outputs
"%T"	Discrete Temporaries
"%M"	Discrete Internals
"%SA"	SA Discretes
"%SB"	SB Discretes
"%SC"	SC Discretes
"%S"	S Discretes
"%G"	Genius Global Data
"%AI"	Analog Inputs
"%AQ"	Analog Outputs
"%R"	Registers
"%L"	Local Subblock Data
"%P"	Program Block Data

The 2000-20 PLC commands provide remote access to all PLC system memory areas except Local Subblock Data and Program Block Data

The PLC fault table can also be read and cleared via OIL PLC commands

Each of the commands is detailed in the following sections

> **NOTE**
> All parameter strings (id, pw, and type) must be enclosed in quote marks

A 6 1 **Write Data to PLC System Memory (PUT)**

<u>Syntax.</u>

PUT "id","pw","type",addr,lngth,srcreg,sreg

where

"id" is the 0 to 7 character ASCII character PLC ID The PLC ID factory default is the null string („) All PLCs respond to their own PLC ID and to the null string Series 90-70 PLC IDs can be up to 7 characters long and can contain any printable ASCII character Series 90-30 PLC IDs can be up to 6 characters long and can only contain the characters 0-9 and A-F

"pw" is the 0 to 7 character ASCII password string that is associated with level 2 or higher PLC access If the PLC has no level 2 password, the null string („) should be used The PLC factory default password is the null string

"type" is a 2 or 3 character string always beginning with % that specifies which PLC memory area to write to The following table lists the PLC memory areas that can be written to and the data widths used

String	**Memory Area Type**	**Data Width**
"%T"	Discrete Temps	Bit
"%M"	Discrete Internals	Bit
"%SA"	SA Discretes	Bit
"%SB"	SB Discretes	Bit
"%SC"	SC Discretes	Bit
"%G"	Genius Global Data	Bit
"%R"	Registers	Word (16 bits)

addr is the number of the first discrete bit or register that is accessed in the PLC Register numbers are used for word-wide transfers Discrete bit numbers are used for bit transfers

lngth specifies the number of bits or words to be written to the PLC The total size of the data transfer cannot exceed 123 words or 1961 bits

srcreg is the OIL register that is the source of the PLC data The data written to the PLC is read from this and subsequent OIL registers

sreg is the 32-bit communication status register When command execution is complete, sreg contains 16 bits of OIL command completion status information Sreg+1 contains 16 bits of PLC status information

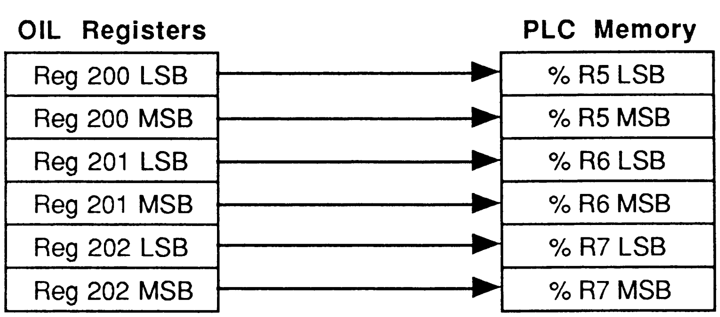
This command writes data from OIL registers to a specified memory area in the General Electric Series 90 PLC

> **NOTE**
> The PLC ID and password defaults are the null string (,,) If the null string is the first command parameter, only one comma needs to be used, e g COMMAND ,2,3 versus COMMAND 1,,3 To change the default, enter specific information (in quotes) in place of the null string

Example 1

```
PUT ,,,"%R",5,3,#200,#300
```

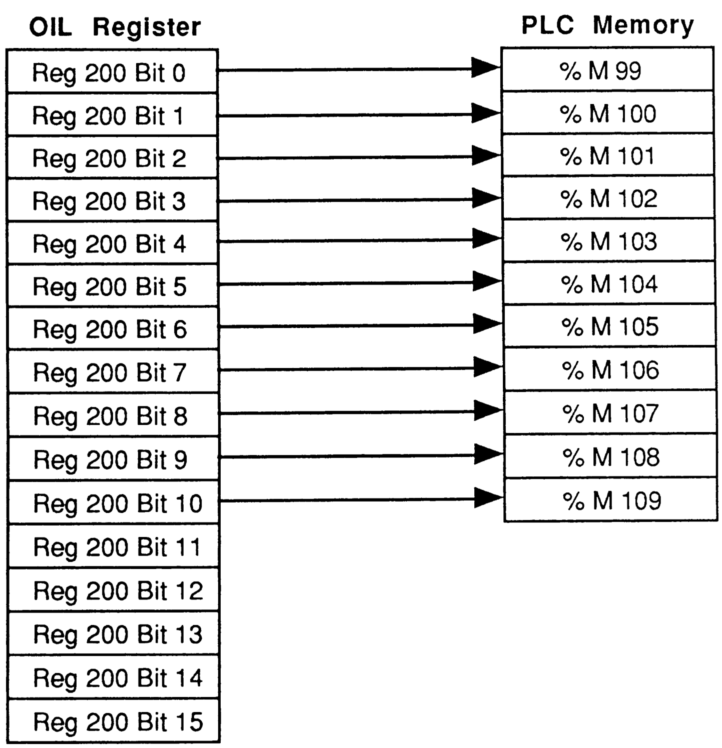
This command writes the contents of OIL registers 200 through 202 to PLC registers %R5 through %R7 The command completion status is returned in OIL register 300 The PLC status is returned in OIL register 301 (Null strings are used for the PLC ID and the password)



Example 2

PUT ,,,"%M",99,11,#200,#300

This command writes 11 bits from OIL register 200 to the PLC starting at bit address %M99 The completion status is returned in OIL register 300 The PLC status is returned in OIL register 301 (Null strings are used for the PLC ID and the password)



A 6 2 **Read Data from PLC System Memory (GET)**

Syntax

GET "id","pw","type",addr,lngth,dreg,sreg

where

"id" is the 0 to 7 character ASCII character PLC ID The PLC ID factory default is the null string („) All PLCs respond to their own PLC ID and to the null string Series 90-70 PLC IDs can be up to 7 characters long and contain any printable ASCII character Series 90-30 PLC IDs can be up to 6 characters long and can contain only the characters 0-9 and A-F

"pw" The 0 to 7 character ASCII password string that is associated with level 1 PLC access If the PLC has no level 1 password, the null string („) should be used The PLC factory default password is the null string

"type" A 2 or 3 character string always beginning with % that specifies which PLC memory area to read from The following table lists the PLC memory areas that can be read from and the data widths allowed

String	Memory Area Type	Data Width
"%I"	Discrete Inputs	Bit
"%Q"	Discrete Outputs	Bit
"%T"	Discrete Temps	Bit
"%M"	Discrete Internals	Bit
"%SA"	SA Discretes	Bit
"%SB"	SB Discretes	Bit
"%SC"	SC Discretes	Bit
"%S"	S Discretes	Bit
"%G"	Genius Global Data	Bit
"%AI"	Analog Inputs	Word (16 bits)
"%AQ"	Analog Outputs	Word (16 bits)
"%R"	Registers	Word (16 bits)

addr is the number of the first discrete bit or register that is accessed in the PLC Register numbers are used for word-wide 16-bit transfers Discrete bit numbers are used for bit transfers

lngth specifies the number of bits or words to be read from the PLC The total size of the data transfer cannot exceed 123 words or 1961 bits

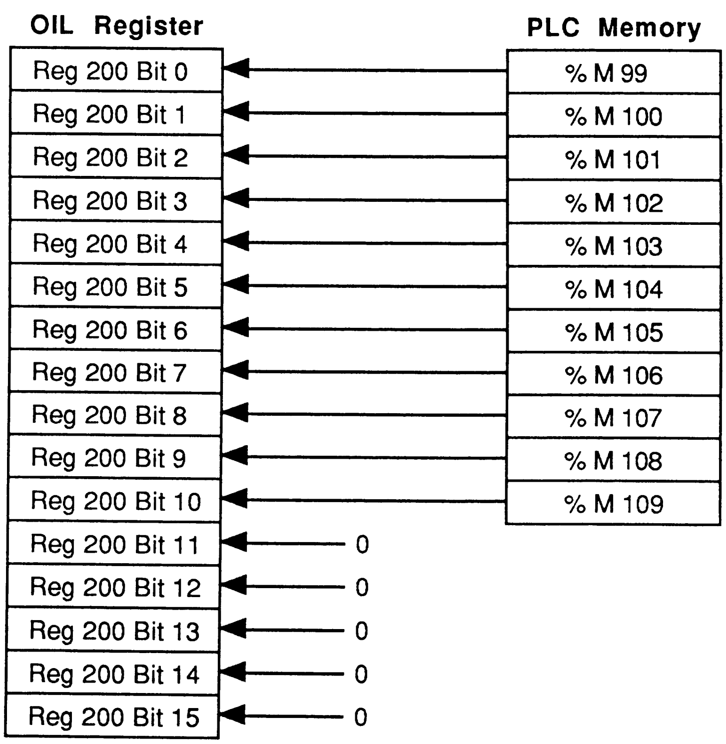
dreg is the destination register to which the PLC data is written sreg   is the 32-bit communication status register When command execution is complete, sreg contains 16 bits of OIL command completion status information Sreg+1 contains 16 bits of PLC status information

This command reads PLC data from a specified memory area in the PLC and stores the data in the specified OIL registers

Example

GET,,,"%M",99,11,#200,#300

This command reads 11 PLC discrete internal bits starting at %M99, and stores the data in OIL register 200 Unused bits in the OIL destination register are set to 0 Register 300 contains the OIL communication status Register 301 contains the communication status (Null strings are used for the PLC ID and the password)



A 6 3 **Read PLC or I/O Fault Table (GETFLT)**

Syntax

GETFLT "id","pw",table,findex,fcount,dreg,sreg

where

"id" is the 0 to 7 character ASCII character PLC ID The PLC ID factory default is the null string („) All PLCs respond to their own PLC ID and to the null string Series 90-70 PLC IDs can be up to 7 characters long and contain any printable ASCII character Series 90-30 PLC IDs can be up to 6 characters long and can contain only the characters 0-9 and A-F

"pw" is a 0 to 7 character ASCII password string associated with level 1 PLC access If the PLC has no level 1 password, the null string („) should be used The PLC factory default password is the null string

table selects either the PLC or I/O fault table
1 = I/O fault table
2 = PLC fault table

findex indicates which faults are returned from the selected fault table
0-31 for I/O fault table
0-15 for PLC fault table

fcount is the number of faults to be read The total number of bytes returned is 12 + (42 * fcount) The fcount cannot exceed 5

dreg is the OIL register to which the PLC fault data is written

sreg is the 32-bit completion status register When command execution is complete, sreg contains 16 bits of OIL command completion status information Sreg+1 contains 16 bits of PLC status information See fault errors on the following page

Fault Data Errors:

First 12 bytes is the fault header

Bytes 0-5	Time stamp of last clear
Bytes 6-7	Faults since last clear
Bytes 8-9	Faults in the table
Bytes 10-11	Number of fault entries that follow

Each PLC fault entry

Bytes 0-3	Spare bytes
Byte 4	Rack number
Byte 5	Slot number
Byte 6	Unit number
Byte 7	Spare byte
Byte 8	Fault group
Byte 9	Fault action
Bytes 10-11	Fault error code
Bytes 12-35	Spare bytes
Bytes 36-41	Time error logged

Each I/O fault entry

Byte 0	Spare byte
Bytes 1-3	Reference bytes
Byte 4	Rack number
Byte 5	Slot number
Byte 6	I/O bus number
Byte 7	Bus address
Bytes 8-9	Point address
Byte 10	Fault group
Byte 11	Fault action
Byte 12	Fault category
Byte 13	Fault type
Byte 14	Fault description
Bytes 15-35	Spare bytes
Bytes 36-41	Time error logged

This command reads a specified number of faults from a PLC or I/O fault table and stores the information in an OIL registers

Example

```
GETFLT "10",,2,12,3,#150,#300
```

This command reads 3 faults from PLC 10's PLC fault table starting at index #12, and stores this information in OIL registers 150 to 218 Communication status is placed into OIL registers 300 and 301 (A null string is used for the password)

A 6 4 **Clear PLC or I/O Fault Table (CLRFLT)**

Syntax

CLRFLT "id","pw",table,sreg

where

"id" is the 0 to 7 character ASCII character PLC ID The PLC ID factory default is the null string („) All PLCs respond to their own PLC ID and to the null string Series 90-70 PLC IDs can be up to 7 characters long and contain any printable ASCII character Series 90-30 PLC IDs can be up to 6 characters long and can contain only the characters 0-9 and A-F

"pw" is the 0 to 7 character ASCII password string that is associated with level 2 PLC access If the PLC has no level 2 password, the null string („) should be used The PLC factory default password is the null string

table selects either the PLC or I/O fault table
1 = I/O fault table
2 = PLC fault table

sreg is the 32-bit completion status register When command execution is complete, sreg contains 16 bits of OIL command completion status information Sreg+1 contains 16 bits of PLC status information

This command clears either the PLC or I/O fault table

Example

CLRFLT "6",,1,#210

This command clears PLC 6's I/O fault table The completion status is stored in OIL registers 210 and 211 (A null string is used for the password)

A 7 **RELATED DOCUMENTS**

For more information, see the following manual

Series 90-30 Programmable Controller User's Manual General Electric Fanuc Automation February, 1990 GFK-0356A

2000-30

Allen-Bradley SLC-500 Interface

96760-030A

Printed in the United States of America
Part Number 96760-030A

XYCOM
750 North Maple Road
Saline, Michigan 48176
(313) 429-4971

XYCOM REVISION RECORD

Revision	*Description*	*Date*
A	Appendix Released	1/94

Copyright Information

Address comments concerning this manual to

xycom
Technical Publications Department
750 North Maple Road
Saline, Michigan 48176

Part Number: ***96760-030A***

A 1 INTRODUCTION

This appendix describes the direct connect used to read and write data from the Allen-Bradley SLC-500 PLC using the Operator Interface Language (OIL) The OIL commands used to communicate with the SLC-500—discussed in Section A 5—are as follows

GET
PUT
PUTM
NETSTAT
GETMODE
PUTMODE

A 2 CABLING THE WORKSTATION TO THE PLC

The 2000 connects to the PLC port via the DH-485 network or through a converter directly to the PLC Figure A-1 shows the cable pinouts used to connect from the 2000 directly to the SLC-500 programming port via RS-485 (Be sure to jumper the port for RS-485)

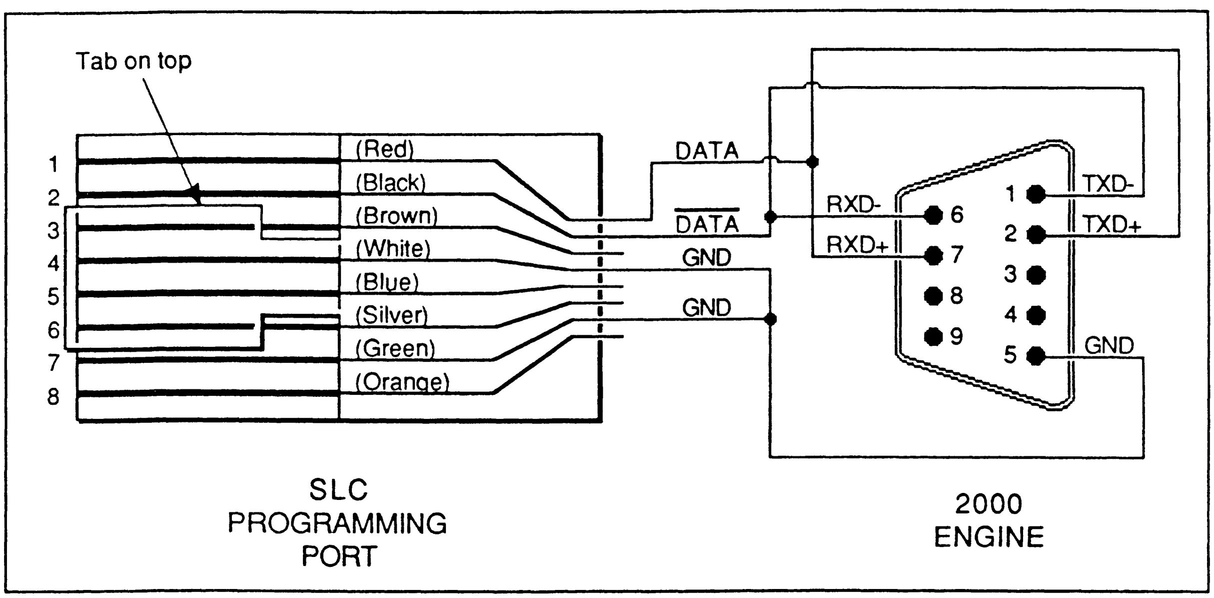


Figure A-1 Cabling to the SLC-500 Programming Port

The cable pinouts used to connect from the 2000 via RS-232 through the 1747-PIC (RS-232C to RS-485 converter) to the SLC-500 programming port are as follows

Table A-1 RS-232C and 1747-PIC Pinouts

Xycom Engine	9 Pin	1747-PIC
3	TXD	2
2	RXD	3
7	RTS	4
4	DTR	20
5	GND	7

A 3 CONFIGURATION MENU

The configuration menus are accessed from the 2000 Main Menu (refer to Chapter 3 in the 2000-OIL manual) The SLC-500 Port Configuration Menu offers the following selections

Table A-2 SLC-500 Port Configuration Menu

Baud Rate	19 2K or 9600
Station Address	0-31
Port Connect	RS-232/1747-PIC or Direct

Baud Rate Match the baud rate of the PLC Refer to the A-B SLC-5XX manual for baud rate settings

Station Address Define the address of the 2000 OIL terminal on the network Each station on the network must have a unique address and it must be different from the PLC's address

NOTE

It is recommended that you do not specify the engine as the last address on the network If it is, communications may be lost

Port Connect Specify the connection from the 2000 to the network as Direct or through the 1747-PIC When connected directly to the network, the primary port must be jumpered for RS-485 To connect to the 1747-PIC, RS-232C must be jumpered

A 4 COMMUNICATION STATUS REGISTER

The Communication Status Register contains the status of the message previously transferred The Most Significant Byte (MSB) is the status returned in the message from the communication module The Least Significant Byte (LSB) contains information about the message transfer status on the workstation

If the bit number in the specified OIL register is set, the condition corresponding to that value (as described below) is true Below is an illustration that depicts the location of the MSB and LSB

MSB								LSB							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0

MSB - Message status from PLC

LSB -
- 0 No ACK error, the command sent was not acknowledged
- 1 Receive error, the command was acknowledged but no response was returned
- 2 Polling error, the DH-485 network has been shut down
- 3 Message error, the message number received did not match the one sent
- 4 SLC error, see the MSB for the local/remote error code
- 5 SLC extended error, see the MSB for the extended error code
- 6 Not used
- 7 Not used

A 5 COMPATIBILITY ISSUES

The SLC-500 fixed I/O controller is addressed via slot numbers 0 through 2 For inputs or outputs, the element number corresponds directly to the slot number

The SLC-501, 502, and 503 are modular controllers These modular controllers can be configured with a maximum of three racks (30 total slots) from a minimum of four I/O points to a maximum of 256 I/O points This allows many different I/O configurations

Slot 0 in a modular SLC controller is reserved for the CPU module Slots 1 through 30 are available for I/O The PLC maps the physical I/O into elements in the input and output memory areas of the PLC Since there can be empty slots, and not all the slots have to have I/O, element numbers do not always correspond to the slot numbers

The PLC and its programming software know the configuration of the PLC, allowing the use of the slot number as the element number At some point, the addresses are mapped to the proper location in the I/O memory areas of the PLC

The A-B SLC-500 direct connect cannot determine the configuration of a modular PLC If a slot is empty, it ignores it and the next full slot continues in the numbering sequence For instance, if slots 1 and 2 are full, slot 3 is empty, and slot 4 is full, the direct connect will skip slot 3 as if it does not exist and will change slot 4 to be slot 3 It is also difficult for the user to determine how the element numbers are mapped to slot numbers in the PLC This makes it difficult to use the direct connect commands to reference I/O

Following are two ways to configure a modular SLC-500 PLC

> **NOTE**
> The second alternative is recommended

- The user must determine the mapping and account for it in the direct connect commands in the OIL program Any time the configuration is changed, the mapping may change and the parameters for the direct connect commands may have to change also

- The PLC program could copy the needed I/O locations to one of the other memory areas (such as N files) The OIL direct connect commands could then reference these other memory areas without worrying about the mapping

A 6 COMMANDS

This section describes the OIL commands that are specific to the Allen-Bradley SLC-500 interface Each command lists the necessary parameters and both registers (#) and indirect ($) registers can be specified

A 5 1 Supported Memory Types and Operations

The memory types and operations supported by this direct connect are listed in Table A-3 All data is one word (16 bits) in size

Memory Type	Operation	Data Files	Elements	Subelement
Outputs	Read	0	0-30	0-255
Inputs	Read/Write	1	0-30	0-255
Status	Read	2	0-15	0
Bit	Read/Write	3, 10-255	0-255	0
Timers	Read/Write	4, 10-255	0-255	0-2
Counters	Read/Write	5, 10-255	0-255	0-2
Control	Read/Write	6, 10-255	0-255	0-2
Integers	Read/Write	7, 10-255	0-255	0

For inputs and outputs, the element number corresponds to the position in the rack of the I/O module For example, the input module closest to the CPU is element 0 The next closest input module is element 1, and so forth The same is true for output modules The output module closest to the CPU is element 0, the next is element 1, and so forth

The subelement number is always zero for slots containing only 16 inputs or outputs For slots with more than 16 inputs or outputs, the subelement is used to select the word number

The status, bit, and integer memory types have one subelement (16 bits) per element Their subelement number must always be zero

The timers, counters, and control types have three subelements per element

subelement 0	=	Control
subelement 1	=	Preset (timers and counters)
	=	Length (control)
subelement 2	=	Accumulator (timers and counters)
	=	Position (control)

For data memory having a single subelement per element the memory is organized as follows

element n, element n + 1, element n + 2,

For data memory with multiple (m) subelements per element, the memory is organized as follows

element n subelement 0, element n subelement 1, ,
element n subelement (m-1), element n + 1 subelement 0,

The GET and PUT commands read and write memory in the same order as it is organized

A 5 3 **GET**

Syntax

GET addr,type,file,element,subelement,length,dest,sreg

Where

addr	=	SLC-500 address (0-31)
type	=	SLC-500 memory type 0 - Outputs 4 - Timers 1 - Inputs 5 - Counters 2 - Status 6 - Control 3 - Bit 7 - Integers
file	=	File number
element	=	Element number
subelement	=	Subelement number
length	=	Number of elements/subelements (1-40)
dest	=	Destination register for data read from the PLC (#12 through maximum number of registers configured - (count) + 1)
sreg	=	Communication status register (#12 through maximum number of registers configured)

GET reads elements/subelements from the PLC data memory into OIL registers

Example

GET 1,4,4,0,2,4,#40,#20

This reads the accumulated value of timers T4 0, T4 1, T4 2, and T4 3 from the PLC at station address 1 The data returned is placed in registers #40 through #43 Register #20 contains the communication status

A 5 4 PUT

Syntax

PUT addr,type,file,element,subelement,length,source,sreg

Where

addr	=	SLC-500 address (0-31)
type	=	SLC-500 memory type 0 - Outputs 4 - Timers 1 - Inputs 5 - Counters 2 - Status 6 - Control 3 - Bit 7 - Integers
file	=	File number
element	=	Element number
subelement	=	Subelement number
length	=	Number of elements/subelements (1-40)
source	=	Source register for data to write to the PLC (#12 through maximum number of registers - (count) + 1)
sreg	=	Communication status register (#12 through maximum number of registers configured)

PUT writes data from OIL registers into the PLC data memory elements/subelements

Example

PUT 1,7,10,3,0,5,#40,#20

This writes data into integer files N10 3, N10 4, N10 5, N10 6, and N10 7 at the PLC at station address 1 The data sent is copied from register #40 through #44 Register #20 contains the communication status

A 5 5 PUTM

Syntax

PUTM addr,type,file,element,subelement,data,mask,sreg

Where

addr	=	SLC-500 address (0-31)
type	=	SLC-500 memory type 0 - Outputs 4 - Timers 1 - Inputs 5 - Counters 2 - Status 6 - Control 3 - Bit 7 - Integers
file	=	File number
element	=	Element number
data	=	Data value to write to an element
mask	=	Bits in an element to be modified 0 = Bit is not modified 1 = Bit is set to the value specified
sreg	=	Communication status register (#12 through maximum number of registers configured)

PUTM writes data to the PLC data memory using a mask The mask specifies which bits of the element get data

Example

PUTM 1,1,1,0,0,#50,15,#20

This example writes data to the inputs in slot 0 at the PLC at station address 1 If register #50 contains a 3, and 3 = 0000 0000 0000 0011b and the mask, 15, = 0000 0000 0000 1111b, only bits 0-3 will be modified Bits 0 and 1 are both set to 1 and bits 2 and 3 are set to 0 Register #20 contains the communication status

A 5 6 NETSTAT

Syntax

NETSTAT addr,station status

Where

addr	=	SLC-500 address (0-31)
station status	=	Status of station on the network

NETSTAT returns the status of the station on the network The register specified as "station status" will contain a 1 if the station is found and is active Otherwise, the register will contain a 0

Example

NETSTAT 3,#30

This example returns the status of the PLC at station address 3 to #30

A 5 7 GETMODE

Syntax

GETMODE addr,curmode,sreg

Where

addr	=	SLC-500 address (0-31)
curmode	=	Current mode
mode	=	PLC mode 0 = Remote Program 1 = Remote Run 2 = Program (SLC-503 only) 3 = Run (SLC-503 only)
sreg	=	Communication status register (#12 through maximum number of registers configured)

GETMODE reads the current mode of a PLC and stores it in the register specified in "curmode " The register is set to 0 if the PLC is in Remote Program mode, 1 if it is in Remote Run mode, 2 if it is in Program mode (SLC-503 only), or 3 if it is in Run mode (SLC-503 only)

Example

GETMODE 10,#21,#20

This example reads the mode of the PLC at station address 10 and puts it into register #21 Register #20 contains the communication status

A 5 8 PUTMODE

Syntax

PUTMODE addr,mode,sreg

Where

addr	=	SLC-500 address (0-31)
mode	=	PLC mode 0 = Remote Program 1 = Remote Run 2 = Program (SLC-503 only) 3 = Run (SLC-503 only)
sreg	=	Communication status register (#12 through maximum number of registers configured)

PUTMODE puts the PLC into Remote Run or Remote Program mode If mode is 0, the PLC is put into Remote Program mode If mode is 1, the PLC is put into Remote Run mode

> **NOTE**
> In order for the PUTMODE command to work on a SLC-503, the key must be in the REM position To put the SLC-503 in Run or Program mode, the key must be physically switched to these positions

Example

PUTMODE 10,1,#20

This example puts the PLC at station address 10 into run mode Register #20 contains the communication status