

Allen-Bradley SLC 500 Application Note

Because of the variety of uses for this information, the user of and those responsible for applying this information must satisfy themselves as to the acceptability of each application and use of the program. In no event will Allen-Bradley Company be responsible or liable for indirect or consequential damages resulting from the use or application of this information.

The illustrations, charts, and examples shown in this document are intended solely to illustrate the principles of programmable controllers and some of the methods used to apply them. Particularly because of the many requirements associated with any particular installation, Allen-Bradley Company cannot assume responsibility or liability for actual use based upon the illustrative uses and applications.

No patent liability is assumed by Allen-Bradley Company with respect to use of information, circuits, equipment, or software described in this text.

Reproduction of the contents of this document, in whole or in part, without written permission of Allen-Bradley Company is prohibited.

1.0 Introduction

PLC and SLC software products are divided into 3 categories; programming software, monitoring and control software and configuration file management software. Programming software allows you to create and edit PLC and SLC application programs. An example of programming software is 6200 Series Software. Monitoring and control software allows read and write of data to and from the PLC and SLC. An example of monitoring and control software is ControlView. Configuration file management software allows PLC and SLC programs to be uploaded and downloaded from a host machine. The host machine is responsible for PLC/SLC file backup and revision history. An example of configuration file management software is Control Guardian.

The purpose of this document is help those who write monitoring and control software to understand the communication differences between PLC-5 and the SLC-5/04. Also, this document is a SLC communication driver guide. This document contains information intended for publication in 1770 6.5.16 **Data Highway/Data Highway Plus (TM)/DH-485 Communication Protocol and Command Set**. Until 1770-6.5.16 can be published, this document is available. Also, Allen-Bradley reserves the right not to make this document public if it so desires.

This document assumes the reader is familiar with DH/DH+/DH-485 message packet formats and addressing as described in commercial publication 1770 6.5.16 **Data Highway/Data Highway Plus (TM)/DH-485 Communication Protocol and Command Set**, 5000-6.4.23 **SLC 500 Family of Programmable Controllers Addressing Reference** and 5000-6.4.4 **1785 PLC-5 Programmable Controllers Addressing Reference**. Topics in this document include:

- 1.0 Introduction
- 2.0 Recommended Commercial Publications
- 3.0 Top Level PLC-5 and SLC-5/04 Comparison
- 4.0 PLC-5 and SLC Communication Command Set Comparison
- 5.0 PLC-5 and SLC Data Type Comparison
- 6.0 Addressing Modes-Logical ASCII, Logical Binary, Typed Data
- 7.0 SLC Command Descriptions
 - 7.1 PCCC or DH+ ID Codes
 - 7.2 New File Types and I/O File Types
 - 7.3 A command to write bits in the SLC
 - 7.4 Typed Read Command and Typed Write Command
 - 7.5 Read File Info for SLC
 - 7.6 Set Processor Mode for SLC
 - 7.7 Read Diagnostic Counters

2.0 Recommended Commercial Publications

1770 6.5.16 **Data Highway/Data Highway Plus (TM)/DH-485 Communication Protocol and Command Set**

- 5000-6.4.23 SLC 500 Family of Programmable Controllers Addressing Reference.
- 1747-2.39 SLC 500 Chassis-based Processors Product Data
- 1747-6.11 Advanced Programming Software (APS) (Cat. No. 1747-PA2E) Reference Manual
- 1747-6.4 Advanced Programming Software (APS) (Cat. No. 1747-PA2E) User Manual
- 1785-6.2.1 1785 PLC-5 Programmable Controllers Design Manual
- 1785-7.1 1785 PLC-5 Programmable Controllers Quick Reference
- 1785-6.6.1 1785 PLC-5 Family Programmable Controllers Hardware Installation Manual
- 6200-6.4.6 PLC-5 Programming Software Configuration and Maintenance User Manual
- 6200-6.4.11 PLC-5 Programming Software Instruction Set Reference
- 5000-6.4.4 1785 PLC-5 Controllers Addressing Reference Manual

3.0 Top Level PLC-5 and SLC-5/04 Comparison

The PLC-5 family uses the 1771 form factor and the SLC-5/04 uses the 1746 form factor. The PLC is larger in size, memory, number of communication channels and price than the SLC. The table below shows some product differences.

PLC-5/40	protocol	SLC-5/04	protocol
channel 0	RS-232/422/423 serial port	channel 0	RS-232 serial port
	ASCII,		DH485,
	DF1 full duplex point to point		ASCII,
	DF1 half duplex slave		DF1 full duplex point to point
	DF1 half duplex master		DF1 half duplex slave
channel 1a	RIO or DH+	channel 1	DH+
channel 1b	RIO or DH+		
channel 2a	RIO or DH+		
channel 2b	RIO or DH+		
channel 3	coprocessor port		
user memory	48K		20K
I/O points	2048		960 discrete
form factor	1771		1746

This comparison is not complete and certainly does not detail differences that exist within each of the product families (i.e.-SLC-5/03 does not have a DH+ channel). Nor does this list include differences in instruction set and data table formats.

Another major difference is that the (1785) PLC-5 is an interpreted machine whereas the SLC is a compiled machine. This means anyone writing programming software must be aware that a compiler/decompiler step is required in the upload and download process. This also means any on-line edit functionality, if available, will be much harder to write.

It is also important to know that at this time (Jan 1995), the SLC-5/04 does not support DH+/RIO passthru technology. DH+/RIO passthru technology allows the upload and download of application programs of certain RIO devices (PanelView 1200, DataLiner) by initiating software commands on DH+ to the SLC and through the RIO connection. DH+/RIO passthru technology exists for the PLC-5. Changes to both the SLC-5/04 and the 1747-SN are required before this technology is available. The passthru in SLC-5/04 that is discussed in the user manual is DH485 passthru. DH485 passthru allows a "1 hop bridge" on DH485 so that the PanelView 550 can be up/downloaded through the SLC.

4.0 PLC-5 and SLC Communication Command Set Comparison

Initially DH+ was not supported in the SLC family. To keep cost down in the SLC, DH485 was implemented rather than DH+. As the product grew in function and popularity, the SLC-5/04 added a DH+ channel. Likewise as the PLC-5 family expanded it added a RS-232 port. Each PLC and SLC implements its own command set for communications. Never assume that a communication command in one product (SLC-5/04) will work in another product (PLC-5/40). For instance a PLC-5 processor implements the command Read-Modify-Write (CMD=0F hex FNC=26) and the SLC-5/04 does not. The Read-Modify-Write command is used when bit manipulation is needed. To accomplish the same thing in a SLC-5/04 you would need to use the command, Protected Typed Logical Write with Mask (CMD=0F hex FNC=AB hex). It is important to read and compare the family processor command tables given in 1770 6.5.16 **Data Highway/Data Highway Plus (TM)/DH-485 Communication Protocol and Command Set** so you are not surprised by what appear to be "incompatibilities". The comparison of PLC-5 and SLC-5/04 public domain commands is found below. Note that six new commands are introduced into the public domain;

- | | |
|---|---------------------------|
| • read section size | for PLC-5 |
| • set processor mode | for all SLC's |
| • typed read, you can not read inputs | for SLC-5/04 & 5/03 OS301 |
| • typed write, you can not write outputs | for SLC-5/04 & 5/03 OS301 |
| • read file info | for SLC-5/04 & 5/03 all |
| • protected typed logical write with mask | for all SLC's |

Command	CMD	FNC	PLC5	SLC	General Description/Use
Diagnostic Counter Reset	06	07	X	X	clear network diagnostics, debug
Diagnostic Loop	06	00	X	X	echo data sent, debug
Diagnostic Read	06	01	X	X	report network diagnostics, debug
Diagnostic Status	06	03	X	X	identify host & show some status
Protected Bit Write	02	N/A	X		write w/o regard to file type
Protected Write	00	N/A	X		write w/o regard to file type
Unprotected Bit Write	05	N/A	X		write w/o regard to file type
Unprotected Read	01	N/A	X	X LIM	read w/o regard to file type
Unprotected Write	08	N/A	X	X LIM	write w/o regard to file type
Download All	0F	50	X	only	download programs
				5/03 & 5/04	
Download Complete	0F	52	X	X	download programs
Modify PLC-2 Compatibility File	0F	5E	X		for plc-2 compatibility
Read Bytes Physical	0F	17	X		read w/o regard to file type for physical upload
Read-Modify-Write	0F	26	X		to write a bit
Set Processor Mode	0F	3A	X		to change mode; run, program, test
Set Processor Mode	0F	80		X	to change mode; run, program, test
Typed Read	0F	68	X	only	to read a word or group of words w/ 5/03 OS301 file type returned. No access to & 5/04 inputs in SLC.
Typed Write	0F	67	X	only	to write a word or group of words 5/03 OS301 and data type must match. No access & 5/04 to outputs in SLC
Upload All	0F	53	X		upload program
Upload Complete	0F	55	X		upload program
Write Bytes Physical	0F	18	X		write w/o regard to file type for physical upload
Word Range Read	0F	01	X		to read a word or group of words w/o any file type returned
Word Range Write	0F	00	X		to write a word or group of words & data type is not checked
Read Link Parameters	06	09		X	report DH485 characteristics
Set Link Parameters	06	0A		X	set DH485 characteristics
Protected Typed Logical Read with 3 Address Fields	0F	A2		X	read word or group of words
Protected Typed Logical Write with 3 Address Fields	0F	AA		X	write word or group of words
Protected Typed Logical Write with Mask	0F	AB		X	to write a bit rather than a word
Read Section Size			X		to determine file type and size
Read File Info	0F	94		only	to determine file type and size
				5/03 & 5/04	

Typed Write: Note, in a PLC-5 the data types must match. No data type conversion occurs. Assume the same for the SLC.

Set Processor Mode: Note, there are 2 distinct commands; one for the SLC family and one for PLC-5 family.

Note on "LIM":

Unprotected Read and Unprotected Write: SLC implementation different than PLC-5. See 1770-6.5.16. Also note that in the SLC version, you can use any node number but they only file that can be read or written is file 9. In the PLC-5 file version, the file read or written is the node number making the request.

In general, the commands most likely to be implemented in monitoring and control software for the PLC-5 and SLC-5/04 are

- Typed Read...to read a word or group of words in PLC-5 or SLC-5/04... in the SLC you can not read the I/O image with this command
- Typed Write...to write a word or group of words in PLC-5 or SLC-5/04...in the SLC you can not write to I/O image with this command
- Protected Typed Logical Write with 3 Address Fields...to write a word or group of words in the SLC-5/04
- Protected Typed Logical Read with 3 Address Fields...to read a word or group of words in the SLC-5/04...and to read to I/O image
- Read-Modify-Write...to write a bit in the PLC-5
- Protected Typed Logical Write with Mask...to write a bit in the SLC-5/04
- Read Section Size...to determine file type and size in the PLC-5
- Read File Info...to determine file type and size in the SLC-5/04

This command set, implemented with logical binary addressing for typed read and typed write commands and separate encode/decode algorithms and commands for I/O image, will give you the "biggest overlap" of commands between the PLC-5 and SLC-5/04.

Typed Read and Typed Write commands are more robust than Word Range Read and Word Range Write commands in that the data type must match (else error). Since data type must be encoded into the packet, packet code and decoding algorithms are more complex. The benefit of code complexity is that there is less of a chance of the wrong data being read or written.

If you are writing a general SLC family driver, you will want to include

- Protected Typed Logical Write with 3 Address Fields...to write a word or group of words in the SLC
- Protected Typed Logical Read with 3 Address Fields...to read a word or group of words in the SLC...and to read the I/O image in the SLC-5/03 and SLC-5/04

- Protected Typed Logical Write with Mask...to write a bit in the SLC
- Read File Info...to determine file type and size in the SLC-5/04

Access to the I/O image in the SLC 500 fixed controller and SLC-5/01 and SLC-5/02 requires a license from Allen-Bradley. Complex decode schemes are required to correctly address this address space. This information is considered too complex to become part of the public domain.

Also note that "Read File Info" command was ONLY implemented in the SLC-5/03 and SLC-5/04. This will be a limitation you will have to keep in mind if you are writing a generic SLC communication driver.

5.0 PLC-5 and SLC Data Type Comparison

The SLC has less data types than the PLC-5. The number of files is limited to 256 rather than 1000 as in the PLC-5. Also, the number of elements per file is limited to 256 rather than 1000 as in PLC-5. The I/O image mapping in the SLC family is different from a PLC-5.

Writing to the SLC I/O image table is much more restricted than in the PLC-5. This is because I/O mapping in an SLC product is more complex than the PLC-5. In latter products (SLC-5/03 and SLC-5/04) an additional I/O file type (8Bh and 8Ch) was added to make access easier. However, the basic addressing difference of rack, group, slot vs slot, word still exists.

	PLC-5	Data Type ID for 5	SLC-5/04	Data Type ID for SLC	File Type Code for SLC
B Binary	x	1	x	1	85h
F Float	x	8	%%	8	8ah
N Integer	x	4	x	4	89h
R Control	x	7	x	7	88h
T Timer	x	5	x	5	86h
C Counter	x	6	x	6	87h
I Input	x	4	x	4	8ch logical by slot ***
O Output	x	4	x	4	8bh logical by slot ***
S Status	x	4	x	4	84h
D BCD	x	10h			
A ASCII	x	3	%%	3h	8eh
BT Block Transfer	x	20h			
MG Message	x	16h			
PD PID	x	15h			
SC SFC Status	x	1dh			
ST ASCII String	x	1eh	%%	1eh	8dh
An array of data	x	9h	x	9h	

*** only available in SLC-5/03 and SLC-5/04. These file type codes will not work with typed write and typed read commands.

%% Float, ASCII and ASCII String types are only available in SLC-5/03 OS301 and SLC-5/04.

All other listed types under SLC-5/04 are valid in all SLC's.

Note any data type code greater than 7h will be preceded with a 9xh field where the '9' indicates that the type information will follow in the next byte. See Appendix E of 1770-6.5.16.

6.0 Addressing Modes-Logical ASCII, Logical Binary, Typed Data

PLC-5 supports 2 addressing modes; logical ASCII and logical binary addressing. The SLC-5/03 OS301 and SLC-5/04 only supports logical binary addressing.

Addressing modes are described in Appendix B of document 1770-6.5.16. Logical ASCII addressing lets you use the same format that you use in the programming software. It also frees you from knowing the internal structure of the PLC/SLC.

In logical ASCII addressing, the format starts with NUL and \$ characters and ends with a NUL. For example if you wish to read element 1 of file 7 in a PLC-5, you would code it as

	ASCII	HEX
byte 1	NUL	00
byte 2	\$	24
byte 3	N	4e
byte 4	7	37
byte 5	:	3a
byte 6	1	31
byte 7	NUL	00

Logical binary addressing as supported in the PLC-5, SLC-5/03 OS301 and SLC-5/04 uses the format of section, file number, element number and sub-element number. The first byte is always the mask byte indicating the number of addressing levels. Section is always 0. Below is an example of the logical binary address for T4:2.ACC for a PLC-5:

```

00001111  4 levels of addressing
00000000  section=0
00000100  file=4
00000010  element=2
00000010  subelement=2 which is ACC (see Appendix B).

```

For data types that consist of a structure (more than 1 type of data/element) like a timer, subelements will exist. The numbering scheme used is its word number within the structure. For example, a timer has its control bits in word 0, the preset in word 1 and its accumulated value in word 2. Definitions are found in Appendix B in 1770-6.5.16. The SLC-5/03 OS301, SLC-5/04 and PLC-5 share the same subelement numbering scheme for timer, counters and control file types.

However, the typed read and typed write command takes a type/data parameter (see Appendix E in 1770-6.5.16) which is a bit more complicated than the above example. The type/data parameter specifies file type and the format of the data. This is best done by example:

where processor type:

1A hex= SLC 500 Fixed processors

18 hex= SLC 5/01

25 hex= SLC 5/02

49 hex= SLC 5/03

5b hex= SLC 5/04

where series/revision:

bits 0-4=0 for FRN1, 1 for FRN2, etc

bits 5-7=0 for Series A, 1 for Series B, etc

7.2 New File Types and I/O File Types

In the SLC-5/03 Series C and the SLC-5/04, several new file types have been introduced: floating point, string and ASCII. These are the file type codes that would be used in Protected Typed Logical Read/Write with Three Address Fields (CMD=0F hex FNC= A2 and CMD=0F hex FNC=AA hex) commands. The file type codes are found below:

File Type	File Type Code	Processor Support
Floating Point	8A hex	SLC-5/03 Ser C and SLC-5/04
ASCII	8E hex	SLC-5/03 Ser C and SLC-5/04
String	8D hex	SLC-5/03 Ser C and SLC-5/04
Output Logical by Slot	8B hex	all SLC-5/03 and 5/04
Input Logical by Slot	8C hex	all SLC-5/03 and 5/04

To read the output image use the "Protected Typed Logical Read w/3 address fields" command with the following parameters (only in 5/03 and 5/04):

byte size	=number of bytes to read
file number	=0
file type	=8b hex
element number	=slot number of the target module
subelement number	=0

To read the input image use the "Protected Typed Logical Read w/3 address fields" command with the following parameters (only in 5/03 and 5/04):

byte size	=number of bytes to read
file number	=1
file type	=8c hex
element number	=slot number of the target module
subelement number	=0

Note that not all bits in the highest word may be used by the slot; e.g. a 24 bit input module would have two 16 bit words, but would only use 24 bits.

CAUTION: AVOID DIRECT WRITING TO THE I/O IMAGE. Accesses to the input and output image should be "read-only" i.e. I/O data should be monitored, not changed. The SLC operating system automatically overwrites the I/O image as the I/O scan collects input data from the field devices and the user ladder program scan updates output data.

Writing to the image of live I/O may create a temporary I/O "glitch" of unknown duration.

If it is necessary to remotely modify data in the I/O image, the preferred method is to write to an integer file and use the data to modify I/O via the user ladder program. The indirect approach has the benefit that it requires a decision on the part of the ladder program developer to "expose" specific I/O points to remote modification, and the ability to condition the remote modification to the state of the system.

Please note that remote writing to the I/O table is prevented in RUN mode in the SLC-500, 5/01, and in the default configuration of the 5/02, 5/03 and 5/04. The user can choose to configure a 5/02, 5/03 and 5/04 to allow remote writing. See User Manual for details.

Access to the I/O image in the SLC 500 fixed controller and SLC-5/01 and SLC-5/02 requires a license from Allen-Bradley. Complex decode schemes are required to correctly address this address space. This information is considered too complex to become part of the public domain.

7.3 A command to write bits in the SLC

1770 6.5.16 Data Highway/Data Highway Plus (TM)/DH-485 Communication Protocol and Command Set does not define a command for the SLC-500 that allows you to write a bit of information. In the PLC-5 you can accomplish this with the Read-Modify-Write command. In the SLC-500 family, you would use the command, Protected Typed Logical Write with Mask (CMD=0F hex FNC=AB hex). The four fields are file, element, sub-element and bit mask. This command is used to write data to a logical address. The file number, element number and sub-element fields are normally one byte unless the value is > FE hex, then a three byte field is used with an ff hex in the first byte and the actual value in the next two bytes in low byte then high byte format. The bit mask has a "1" for each bit that can be modified by this command and a "0" for each bit that is to remain unaffected by this command. There is exactly one 16-bit mask that is used repeatedly if the data is longer than two bytes. If the size byte is odd, then the low byte of the bit mask will be used for the low byte.

Note: Use this command to ONLY write to the file types listed below.

CAUTION: THE CONTROLLER MAY CHANGE THE STATES OF THE ORIGINAL BITS IN MEMORY BEFORE THIS COMMAND CAN WRITE THE WORD BACK TO MEMORY. Therefore, some data bits may unintentionally be overwritten. To help prevent this, we suggest that you use this

command to write into the storage area of the controller's data table, and have the controller read the word only, not control it.

Command Parameters:

cmd 0F hex	1 byte
sts	1 byte
tnsw AB ₁₆	1 word
fnr AH hex	1 byte
byte size	1 byte, size of "data" field to be written
file number	1 byte or 3 bytes, 1 if 0-254 or 3 if 255 or higher
file type	1 byte, see below
element number	1 byte or 3 bytes, 1 if 0-254 or 3 if 255 or higher
subelement number	1 byte or 3 bytes, 1 if 0-254 or 3 if 255 or higher
bit mask	1 word; low byte then high byte
data	determined by byte size; low byte then high byte

Byte size value can range from 0 to the maximum allowed by the target device.

For the SLC-500, 5/01 and 5/02, valid range is 0 to 88 bytes (odd or even).

For the SLC-5/03 and 5/04 with Internet protocol used, valid range is 0-217 bytes (odd or even).

For the SLC-5/03 and 5/04 without Internet protocol used, valid range is 0-228 bytes (odd or even).

The maximum number of allowed data bytes is reduced by 2 bytes for each field that is expanded.

File number, element number and sub-element number are expanded if the field contains an FF hex.

Example, 257 decimal expands to FF hex, 01 hex, 01 hex (low byte then high byte).

Note: Use this command to ONLY write to the file types listed below.

File type codes are	ONLY in SLC-5/03 OS301 and 5/04	
84 hex	status	
85 hex	bit	
86 hex	timer	
87 hex	counter	
88 hex	control	
89 hex	integer	
8A hex	floating point	xxxxx
8B hex	output logical by slot	xxxxx
8C hex	input logical by slot	xxxxx
8D hex	string	xxxxx
8E hex	ASCII	xxxxx

Bit mask has a "1" for each bit that can be modified by this command and a "0" for each bit that is to remain unaffected by this command.

Reply Parameters:

cmd 4f hex	1 byte
------------	--------

sts	1 byte
tnsw	1 word
ext sts	1 byte

Sts and ext sts value:

00 hex	success
10 hex	command format is wrong
50 hex	address problem
F0 hex 0B hex	access denied, improper privilege
F0 hex 1A hex	file already open + byte indicating node address of file owner
F0 hex 1B hex	processor owned + byte indicating node address of program owner

7.4 Typed Read Command and Typed Write Command

Two PLC-5 commands have now been added to the SLC-5/03 OS301 and SLC-5/04; Typed Read and Typed Write (CMD=0F hex FNC=68 hex and CMD=0F hex and FNC=67 hex). See page 3.6.11 in 1770 6.5.16 **Data Highway/Data Highway Plus (TM)/DH-485 Communication Protocol and Command Set** for details. Please note that the size parameter is 2 bytes and is specified low byte and high byte. Size is in units of number of elements. Only logical binary addressing is supported. The format for the logical binary addressing follows the PLC-5 format. See Appendix B of 1770 6.5.16 **Data Highway/Data Highway Plus (TM)/DH-485 Communication Protocol and Command Set** for details. For addressing details see section 5.0 **Addressing Modes-Logical ASCII, Logical Binary, Typed Data**, of this document.

Both the PLC-5 and SLC will return long words (32 bit) for the accumulated and preset values of a timer.

3/6/95

Sub-element addressing is not supported in SLC in typed read and typed write command.

7/27/95

Other differences between PLC-5 and SLC typed read and typed write commands exist! Documentation is in process!

6.5 Read File Info for SLC

To determine a file's type and size in the SLC use cmd=0fh, fnc=94h. The format used will specify the address to the "file number" level by using mask 06h.

This command is only available in the SLC-5/03 and SLC-5/04.

Command Format:

cmd, sts, tns, fnc, 06h, 80h, xxh

8/21/95 Command not supported by Micrologix(TM) 1000 because data table is fixed.

where,

06h =mask means 2 bytes to follow, major file type and file number

80h =major file type=80h=data table file

xxh =file number (0 to 255 decimal)

Reply Format:

sts + reply data

or sts, ext sts (+ node address of owner)

STS Ext STS

00h success + data

10h illegal format

50h bad address; file doesn't exist

F0h 1Bh file protection error + byte with node address of owner

Reply Data

Size; word low, byte low where size is size of file in bytes

Size; word low, byte high

Size; word high, byte low

Size; word high, byte high

Count word; byte low where count word= number of elements in file (2 bytes/element)

Count word; byte high

Count word; byte high

Reserved Byte

Data Type Byte

where data type byte,

84h status

85h bit

86h timer

87h counter

88h control

89h integer

- 8ah floating point
- 8bh output logical by slot
- 8ch input logical by slot
- 8dh string
- 8eh ASCII
- 8fh BCD

7.6 Set Processor Mode for the SLC

Set processor mode (cmd=0fh, fnc=80h) is different than the PLC-5. The command is only allowed when keyswitch must be in "REM" position. There is only 1 command parameter and it is the byte that follows the fnc byte. This is the "mode" byte. Values for mode are found below:

- "mode" =01h, change to Program mode (REM Prog)
- =06h, change to Run mode (REM Run)
- =07h, change to Test-continuous scan mode (REM Test)
- =08h, change to Test-single scan mode (REM Test)
- =09h, change to Test-debug single step (REM Test)

SLC500 & 5/01 do not support mode 09h.

The reply can have the following STS and Ext STS values:

STS	Ext STS	
00h		Success
10h		Command format is wrong
F0h	0Bh	Access Denied: Channel Read Protect
F0h	0Ch	Condition cannot be generated (keyswitch)
F0h	1Ah	File already open + byte containing DH485 node number of file owner
F0h	1Bh	Processor owned + byte containing DH485 node number of program owner

7.7 Read Diagnostic Counters

The response to this command will be determined by the protocol used ie- DH485, DF1 Full-Duplex, DF1 Half-Duplex or DH+. The format and responses are found below:

Command Format

- CMD byte = 06h
- FNC byte = 01h
- cmd param,address = 0000h (2 bytes) This is the only valid value for SLC
- cmd param,size = 0Ah This is the only valid value for SLC

Possible Responses

STS	Ext STS
00	Success + Data (low byte, high byte)
10h	Illegal command or size too large

Reply for Data for DH485 size <=10 bytes

Total Packets Received, low byte
Total Packets Received, high byte
Total Messages Sent, low byte
Total Messages Sent, high byte
Retries
Retry Limit Exceeded
NAK, no memoy sent
NAK, no memory received
Bad Messages Received
Reserved

Reply for Data for DF1 Full-Duplex size <=40 bytes

RS232 Modem Line Status, bit 0=CTS input line sts
bit 1=RTS output line sts
bit 2=DSR input line sts
bit 3=DCD input line sts
bit 4=DTR output line sts

Total message packets sent
Total message packets received
Undelivered message packets
ENQuiry packets sent
NAK packets received
ENQ packets received
Bad Message packets received & NAK'ed
No buffer space and NAK'ed
Duplicate message packets received
00h (unused)
DCD recover field
Lost modem field
00h (unused)
00h (unused)
00h (unused)
00h (unused)
00h (unused)
00h (unused)
00h (unused)
00h (unused)

Reply for Data for DF1 Half-Duplex size <=40 bytes

RS232 Modem Line Status, bit 0=CTS input line sts
 bit 1=RTS output line sts
 bit 2=DSR input line sts
 bit 3=DCD input line sts
 bit 4=DTR output line sts

Total message packets sent
Total message packets received
Undelivered message packets
Message packets retried
NAK packets received
Polls received
Bad Message packets rec's & NAK'ed
No buffer space
Duplicate message packets rec'd
00h (unused)
DCD recover field
Lost modem filed
00h (unused)
00h (unused)
00h (unused)
00h (unused)
00h (unused)
00h (unused)
00h (unused)

Reply Data for DH+ size <=64 bytes

Messages received
Messages sent
Messages received with error
Messages sent with errors
Messages Unable to receive
Network dead
Claims won
Claims lost
New successor
Token retry
Token failed
Started Linear scan
Linear scan failed
Duplicate node
Dropped token
Received SDA
Received bad packets
Received SDA retransmission

Received SDA but full
 Received SDA SAP off
 Transmit SDA confirm
 Transmit SDA NAK misc
 Transmit SDA timeout
 Transmit SDA not acked
 Transmist SDX retry
 Transmit SDA failed
 Transmit SDA Naked full
 Transmit SDA Naked inactive SAP
 Transmit SDN confirm
 Transmit SDN failed
 Solicit rotations
 Receive SDN

500 CPU Write - LOCAL

	File # < 254 Element # < 254	either > 254	File # > 254 Element # > 254
Data bytes available	230	228	226
Input, Output, Status, Bit, Integer, ASCII	115	114	113
Float	57	57	56
Timer, Counter, Control	38	38	37
String	2	2	2

500 CPU Write - INTERNET

	File # < 254 Element # < 254	either > 254	File # > 254 Element # > 254
Data bytes available	219	217	215
Input, Output, Status, Bit, Integer, ASCII	109	108	107
Float	54	54	53
Timer, Counter, Control	36	36	35
String	2	2	2

500 CPU Read & 485 CIF Read

	LOCAL	INTERNET
Data bytes available	236	225
Input, Output, Status, Bit, Integer, ASCII	118	112
Float	59	56
Timer, Counter, Control	39	37
String	2	2

485 CIF Write

	LOCAL	INTERNET
Data bytes available	234	223
Input, Output, Status, Bit, Integer, ASCII	117	111
Float	58	55
Timer, Counter, Control	39	37
String	2	2

PLC5 Write - LOCAL

	File # < 254 Element # < 254	either > 254	File # > 254 Element # > 254
Data bytes available	222	220	218
Input, Output, Status, Bit, Integer	111	110	109
Float	55	55	54
Timer	22	22	21
Counter, Control	37	36	36
ASCII	222	220	218
String	1	1	1

PLC5 Write - INTERNET

	File # < 254 Element # < 254	either > 254	File # > 254 Element # > 254
Data bytes available	211	209	207
Input, Output, Status, Bit, Integer, ASCII	105	104	103
Float	52	52	51
Timer	21	20	20
Counter, Control	35	34	34
String	1	1	1

PLC5 Read

	LOCAL	INTERNET
Data bytes available	230	219
Input, Output, Status, Bit, Integer, ASCII	115	109
Float	57	54
Timer	23	21
Counter, Control	38	36
String	1	1

7/27/95

Important type/data parameter note:

The PCCC specifications says that the type/data parameter is a variable length field. The most significant bit of each nibble determine if additional bytes are used. This field value can be extended to a 7 byte unsigned integer. The order of the bytes is least to most significant. All zero most significant bytes are permitted, but the fields thus generated are not different than those that omit these insignificant bytes. Therefore the following three descriptors all are permitted and equivalent:

case 1:	case 2:	case 3:
0100 0011	0100 1001	0100 1010
	0000 0011	0000 0011
		0000 0000

These all have an "ID" value of four and a "size" value of three.

Your software must be able to handle all 3 cases!

4.36 Protected Typed Logical Write with Mask (CMD = 0x0F, FNC = 0xAB)

- (a) CMD Byte - 0x0F
- (b) FNC Byte - 0xAB
- (c) Command Parameters -

- Size, one byte, size of data to write

SLC Product	Valid Range
500, 5/01, 5/02	0 - 82 bytes
5/03, 5/04, 5/05 Channel 0 (RS232 Port) and 5/03 Channel 1 (DH485)	0 - 217 bytes with Internet network layer field ¹ 0 - 228 bytes without Internet field ¹
5/04 Channel 1 (DH+)	0 - 236 bytes with or without Internet field ¹
5/05 Channel 1 (Ethernet)	0 - 2096 bytes ¹ (typically file size limits reached first ²)

- File Number, one or three bytes³
- File Type, one byte, see Appendix A for valid SLC File Types
- Element Number, one or three bytes¹
- Sub-Element Number, one or three bytes³
- Bit Mask, two bytes

The bit mask has a "1" for each bit that can be modified by this command, and a "0" for each bit that is to remain unaffected by this command. There is exactly one 16-bit bit mask. It is used repeatedly if the data is longer than two bytes. If the size byte is odd, then the low byte of the bit mask will be used for the odd byte.

- Data, number of bytes to write, low byte first

¹ Deduct 2 bytes from maximum valid range for every field that is in three byte format rather than one byte format (see Note 3). EXAMPLE: 5/03 receives command without Internet field, File Number is 3 bytes, Element Number is 1 byte, Sub-Element Number is 3 bytes; valid range of data that can be written is then 0 - 224 bytes (not 0 - 228 bytes as tabulated).

² With an SLC limit of 256 elements per data file, this command can be used to write an entire maximum size data file via the SLC 5/05's Ethernet port with the exception of a String data file type where only 24 elements may be written.

³ Note: The File Number, Element Number, and Sub-Element Number values are normally one byte; however, these fields may be expanded to three bytes to allow a word. The format of the expanded field is 0xFF, low byte of value, high byte of value.

(d) Possible Reply Error Codes -

STS	Ext STS	Reason
0x00	none	Success
0x10	none	Illegal format
0x50	none	Address problem
0xF0	0x0B	Access Denied: Channel Protected or Improper Privilege
0xF0	0x1A	File already open + node address of file owner, 1 byte (for SLC 5/05, value of 0xFF if ownership via Ethernet port)
0xF0	0x1B	Processor owned + node address of program owner, 1 byte (for SLC 5/05, value of 0xFF if ownership via Ethernet port)

(e) Operation -

This command is used to write bit data to a logical address in an SLC processor by specifying a bit mask three address fields: file number, element number and sub-element number. If this command is used to write to I/O Modules (i.e. "G" files, M0 & M1 files), then the data is written WITHOUT the Mask.

(f) Command Format -

COMMAND BLOCK	
FNC Byte (0xAB)	
Byte Size	
File Number	
File Type	
Element Number	
Sub-Element Number	
Bit Mask, low	
Bit Mask, hi	
1st Data Word, low	
1st Data Word, hi	
.	
.	
Last Data Word, low	
Last Data Word, hi	

REPLY BLOCK
Ext Status if STS = 0xF0
Owner Address if Ext STS = 0x1A or 0x1B