



A Sierra Monitor Company

Driver Manual
(Supplement to the FieldServer Instruction Manual)

FS-8704-16 Omron FINS

APPLICABILITY & EFFECTIVITY

Effective for all systems manufactured after May 1, 2004

TABLE OF CONTENTS

1. Omron FINS Description.....	1
2. Driver Scope of Supply	2
1.01 Supplied by FieldServer Technologies for this driver	2
1.02 Provided by the Supplier of 3 rd Party Equipment.....	2
2.1.1. <i>Required 3rd Party Hardware</i>	2
2.1.2. <i>Required 3rd Party Software</i>	2
2.1.3. <i>Required 3rd Party Configuration</i>	2
3. Hardware Connections	3
1.03 Hardware Connection Tips / Hints.....	3
1.04 Example of Omron PLC Configuration using Omron ETN11 Module.....	4
4. Configuring the FieldServer as a FINS Client	6
1.05 Data Arrays/Descriptors	6
1.06 Driver Specific FieldServer Parameters	7
1.07 Client Side Connection Descriptors.....	7
1.08 Client Side Node Descriptors	8
1.09 Client Side Map Descriptors.....	9
4.1.1. <i>FieldServer Related Map Descriptor Parameters</i>	9
4.1.2. <i>Driver Related Map Descriptor Parameters</i>	9
4.1.3. <i>Timing Parameters</i>	10
4.1.4. <i>Map Descriptor Example 1 – IO Read</i>	11
4.1.5. <i>Map Descriptor Example 2 – IO Write</i>	12
4.1.6. <i>Map Descriptor Example 3: Clock Read:</i>	13
4.1.7. <i>Map Descriptor Example 4: Clock Write:</i>	13
4.1.8. <i>Map Descriptor Example 5: Read CPU Cycle Times:</i>	13
4.1.9. <i>Map Descriptor Example 6: Run-Stop PLC:</i>	14
4.1.10. <i>Map Descriptor Example 7: Read CPU Status:</i>	14
5. Configuring the FieldServer as a FINS Server	15
1.010 Server Side Connection Descriptors	15
1.011 Server Side Node Descriptors.....	16
1.012 Server Side Map Descriptors.....	17
5.1.1. <i>FieldServer Specific Map Descriptor Parameters</i>	17
5.1.2. <i>Driver Specific Map Descriptor Parameters</i>	17
5.1.3. <i>Map Descriptor Example 1: IO Read</i>	18
5.1.4. <i>Map Descriptor Example 2: IO Write</i>	19
5.1.5. <i>Map Descriptor Example 3: Clock Read:</i>	19
5.1.6. <i>Map Descriptor Example 4: Clock Write:</i>	19
5.1.7. <i>Map Descriptor Example 5: CPU Cycle Times:</i>	19
5.1.8. <i>Map Descriptor Example 6: Run-Stop PLC:</i>	20
5.1.9. <i>Map Descriptor Example 7: CPU Status:</i>	20
Appendix A. Advanced Topics	21
Appendix A.1. Driver Map Descriptor Parameter's Bounds	21

Appendix A.2. PLC status to execute commands	22
Appendix A.3. End Codes.....	23
Appendix B. Driver Notes	30
Appendix B.1. Data Storage	30
Appendix B.2. Driver stats	32
Appendix B.3. Driver Error Messages.....	33
Appendix C. Troubleshooting tips	35
Appendix C.1. Connection Tips & Hints	35
Appendix D. Revision History	37

1. Omron FINS Description

The Ethernet Omron FINS driver allows the FieldServer to transfer data to and from devices over Ethernet using Omron FINS protocol. The FieldServer can emulate either a Server or Client.

FINS is an Omron protocol which can be used by a PLC program to transfer data and perform other services with a remote PLC connected on an Ethernet Network. It can also be used by remote devices such as PC's and FieldServer's to transfer data and perform other services.

The protocol uses the Ethernet protocol called UDP to carry the FINS messages back and forth. The UDP protocol is not connection based and reliability is achieved by using confirmation messages.

This Ethernet Driver can be used to transfer data to and from the Nodes supporting FINS communications.

Max Nodes Supported

FieldServer Mode	Nodes	Comments
Client	126	Omron limit the set of permitted nodes to 126. They are numbered 1 to 126 corresponding to the last byte of the remote node IP address.
Server	20	The FieldServer can emulate a maximum of 20 Omron FINS servers.

2. Driver Scope of Supply

1.01 Supplied by FieldServer Technologies for this driver

FieldServer Technologies PART #	Description
FS-8915-10	UTP cable (7 foot) for Ethernet connection
FS-8704-16	Driver Manual.

1.02 Provided by the Supplier of 3rd Party Equipment

2.1.1. Required 3rd Party Hardware

Part #	Description

2.1.2. Required 3rd Party Software

CX-Programmer Software or any other compatible Software by Omron to setup the PLC

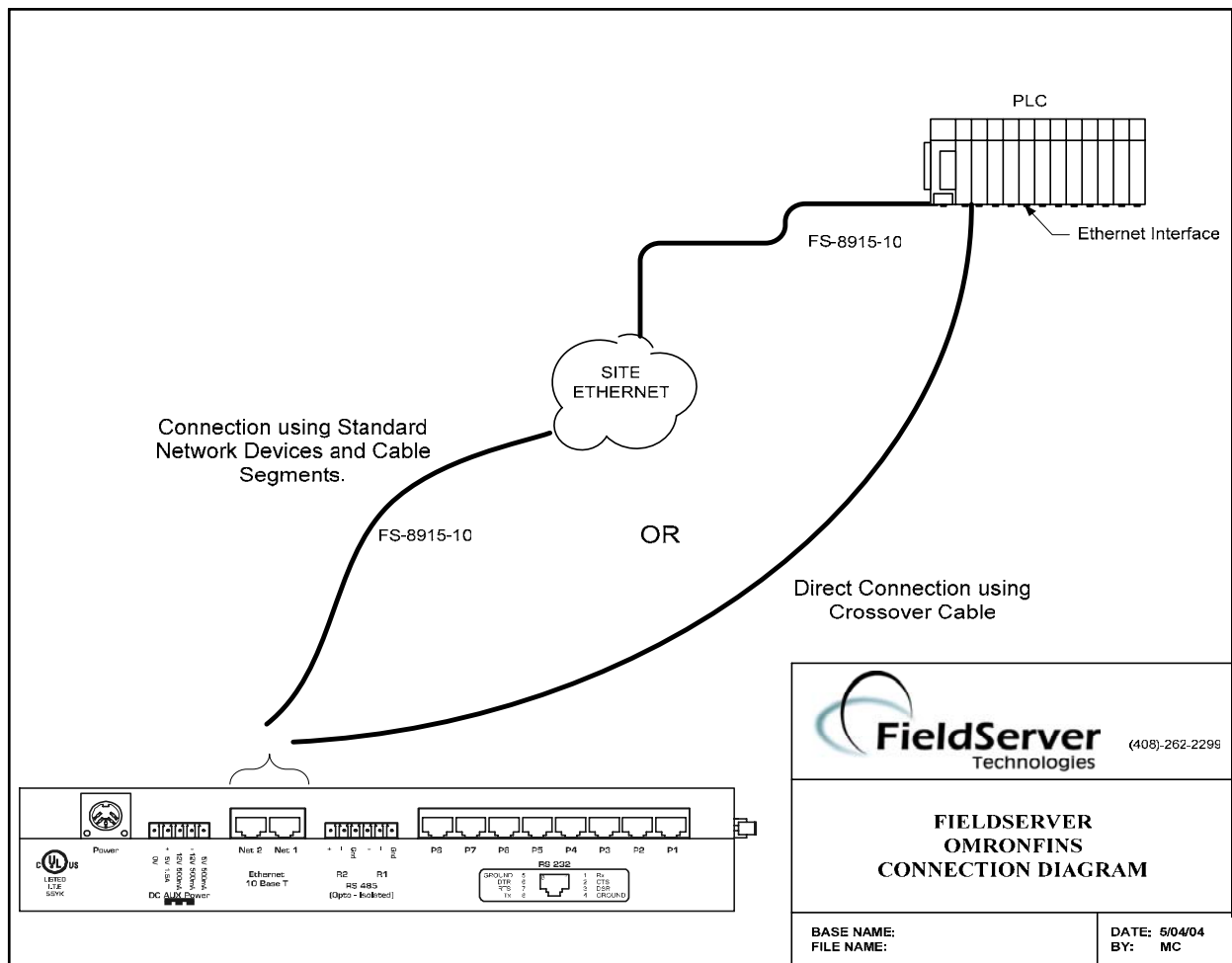
2.1.3. Required 3rd Party Configuration

The Omron PLC device needs to be set into the Automatic Address Generation mode using the CX-Programmer or any other compatible Software.

3. Hardware Connections

The FieldServer is connected to the PLC as shown in connection drawing.

Configure the PLC according to manufacturer’s instructions to work with other FINS supported device.



1.03 Hardware Connection Tips / Hints

If communication doesn't start check the following.

1. Are the FieldServer and PLC on the same network?
2. Are all intended Nodes configured to communicate on FINS?
3. Are all FINS Nodes configured to use the same Port Number?

4. Is the Network healthy?

1.04 Example of Omron PLC Configuration using Omron ETN11 Module.¹

Setting the Node Number



With the FINS communications service, when there are multiple Ethernet Units connected to the Ethernet network, the Ethernet Units are identified by node numbers. Use the node number switches to set the node number between 01 and 7E hexadecimal (1 to 126 decimal). Do not set a number that has already been set for another node on the same network.

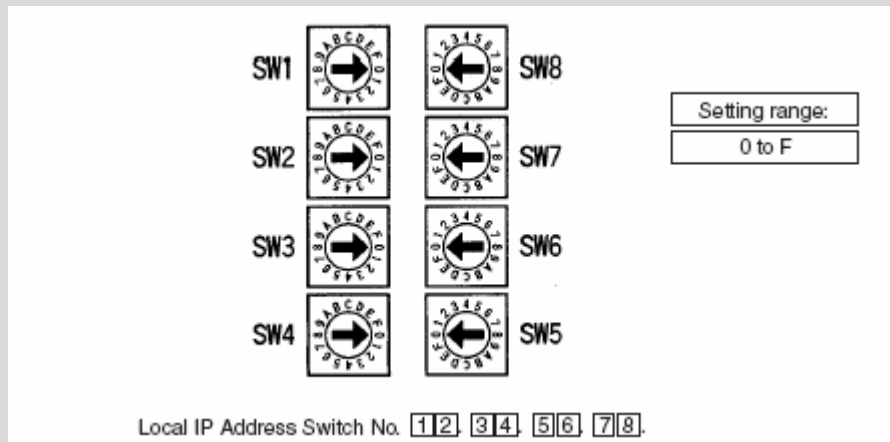
The left switch sets the sixteens digit (most significant digit) and the right switch sets the ones digit (least significant digit). The node number is factory set to 01. When using the automatic generation method for address conversion, set the node number to the same value as that of the local IP address switches. If this is not possible, then either the IP address table method or the combined method must be used for address conversion. For details, refer to 4-2 CPU

Bus Unit System Setup. If the FINS communications service is not being used over the Ethernet network, then there is no problem if the node number duplicates that of another Ethernet Unit. The node number must still be set from 01 to 7E, however, or the ERC indicator will light.

Note Turn OFF the power supply before setting the node number.

Setting the Local IP Address

The nodes on an Ethernet network are identified by IP addresses. Each IP address is set with 32 bits of binary data. These 32 bits are divided into four 8-bit fields called octets, and each octet is expressed as four decimal numbers. At CS-series Ethernet Units, four bits are expressed as a hexadecimal digit, and the eight hexadecimal rotary switches (the local IP address switches) on the back of the Unit are used to set the local IP address. Set the eight switches in hexadecimal as shown below, combining SW1 and SW2, SW3 and SW4, SW5 and SW6, and SW7 and SW8. Each switch can be set from 0 to F.



¹ The following notes are extracted from the Omron Manual “W343-E1-05 9/03”

The switches are all factory-set to 0 (00.00.00.00). The Ethernet Unit cannot be used with this setting; a proper IP address must be set.

The following settings cannot be made for the IP address, or the ERC indicator will flash.

All bits in the network number field set to 0 or 1.

All bits in the host number field set to 0 or 1.

All bits in the subnet number field set to 1.

The beginning of the IP address set to 127 (7F Hex) Example: 127.35.21.16

Note 1. When using the automatic generation method for address conversion, set switches 7 and 8 to the same values as the node number setting, and set the rest of the host number to zeroes. For details regarding the host number, refer to *1-7 IP Addresses*. The value for the host number field in the IP address must match the value for the node number or the ERC indicator will flash.

2. If a subnet mask is to be set, use the CX-Programmer to set it in the CPU Bus Unit System Setup. For details, refer to *4-2 CPU Bus Unit System Setup*.

4. Configuring the FieldServer as a FINS Client

For a detailed discussion on FieldServer configuration, please refer to the FieldServer Configuration Manual. The information that follows describes how to expand upon the factory defaults provided in the configuration files included with the FieldServer (See “.csv” sample files provided with the FieldServer).

This section documents and describes the parameters necessary for configuring the FieldServer to communicate with a FINS Server. As a Client this driver reads and writes data to Server Nodes. Server nodes should be FINS capable and be configured to communicate over FINS

1.05 Data Arrays/Descriptors

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for FINS communications, the driver independent FieldServer buffers need to be declared in the “Data Arrays” section, the destination device addresses need to be declared in the “Client Side Nodes” section, and the data required from the servers needs to be mapped in the “Client Side Map Descriptors” section. Details on how to do this can be found below.

Note that in the tables, * indicates an optional parameter, with the bold legal value being the default.

Section Title		
Data Arrays		
Column Title	Function	Legal Values
Data_Array_Name	Provide name for Data Array	Up to 15 alphanumeric characters
Data_Array_Format	Provide data format. Each Data Array can only take on one format.	Float, Bit, UInt16, SInt16, Packed_Bit, Byte, Packed_Byte, Swapped_Byte
Data_Array_Length	Number of Data Objects. Must be larger than the data storage area required by the Map Descriptors for the data being placed in this array.	1-32767

Example

```
// Data Arrays
Data_Arrays
Data_Array_Name,          Data_Format,          Data_Array_Length,
DA_CIO,                   UInt16,               5143
DA_WR,                    UInt16,               511
DA_HR,                    UInt16,               511
DA_DM,                    UInt16,               32767
```

1.06 Driver Specific FieldServer Parameters

Section Title		
FieldServer		
Column Title	Function	Legal Values
Title	Name for FieldServer	Text
System_Node_Id	Specify physical node Id on network. This is the last byte of the IP Address of the FieldServer. eg. If FieldServer's IP Address is 192.168.1.81 then this parameter should be set to 81	1-126

Example

```
// FieldServer Driver specific parameters
FieldServer
Title,                      System_Node_Id
Fins Client,                81
```

1.07 Client Side Connection Descriptors

Section Title		
Adapter		
Column Title	Function	Legal Values
Adapter	Adapter Name	N1, N2 ²
Protocol	Specify protocol used	Fins, omn_fins or fins_udp
Udp_port_number	Specify UDP port number	0, 9600 etc

² Not all ports shown are necessarily supported by the hardware. Consult the appropriate Instruction manual for details of the ports available on specific hardware.

Example

// Client Side Connections		
Adapters		
Adapter, N1,	Protocol, Fins,	Udp_port_number 9600

1.08 Client Side Node Descriptors

Section Title		
Nodes		
Column Title	Function	Legal Values
Node_Name	Provide name for node	Up to 32 alphanumeric characters
IP_Address	Provide IP Address of PLC	Eg. 192.168.1.105
Node_ID	Node number set at PLC The node number should correspond to the last byte of the IP address. Eg. 105 corresponds to the example IP address above.	1-126
Protocol	Specify protocol used	Fins, omn_fins or fins_udp
Adapter	Specify which port the device is connected to the FieldServer	N1, N2 ³
*Net_Number	Provide the network number of PLC	1-255

Example

// Client Side Nodes					
Nodes					
Node_Name,	IP_Address,	Node_ID,	Protocol,	Adapter,	Net_Number
PLC 1,	192.168.1.5,	1,	Fins,	N1,	1

³ Not all ports shown are necessarily supported by the hardware. Consult the appropriate Instruction manual for details of the ports available on specific hardware.

1.09 Client Side Map Descriptors

4.1.1. FieldServer Related Map Descriptor Parameters

Column Title	Function	Legal Values
Map_Descriptor_Name	Name of this Map Descriptor	Up to 32 alphanumeric characters
Data_Array_Name	Name of Data Array where data is to be stored in the FieldServer	One of the Data Array names from "Data Array" section above
Data_Array_Offset	Starting location in Data Array	0 to maximum specified in "Data Array" section above
Function	Function of Client Map Descriptor	RDBC, WRBC, WRBX

4.1.2. Driver Related Map Descriptor Parameters

Column Title	Function	Legal Values
Node_Name	Name of Node to fetch data from	One of the node names specified in "Client Node Descriptor" above
*Data_Type (see Note 1 Appendix A.1)	Data type	CIO-WORD, WR-WORD, HR-WORD, AR-WORD, EM-WORD
Length	Length of Map Descriptor	1- 729
Address	Starting address of read block	0, 1,...100, etc see Appendix A.1 for details
*Memory_Code (see Note 1 Appendix A.1)	Memory code for PLC memory type	B0, B1, B2, B3, 82, 98 and A0 ---- AC
Command_Name (see Note 2 section 6.1)	Name of the command	MEMORY AREA READ MEMORY AREA WRITE RUN-STOP CLOCK READ CLOCK WRITE CYCLE TIME READ CPU UNIT STATUS READ
*MRC (see Note 2 Appendix A.1)	Main Request Code	1,4,6,7
*SRC (see Note 2 Appendix A.1)	Sub Request Code	1,2,20

*Unit_Number	Unit number of CPU at PLC	0,1,2 etc
--------------	---------------------------	-----------

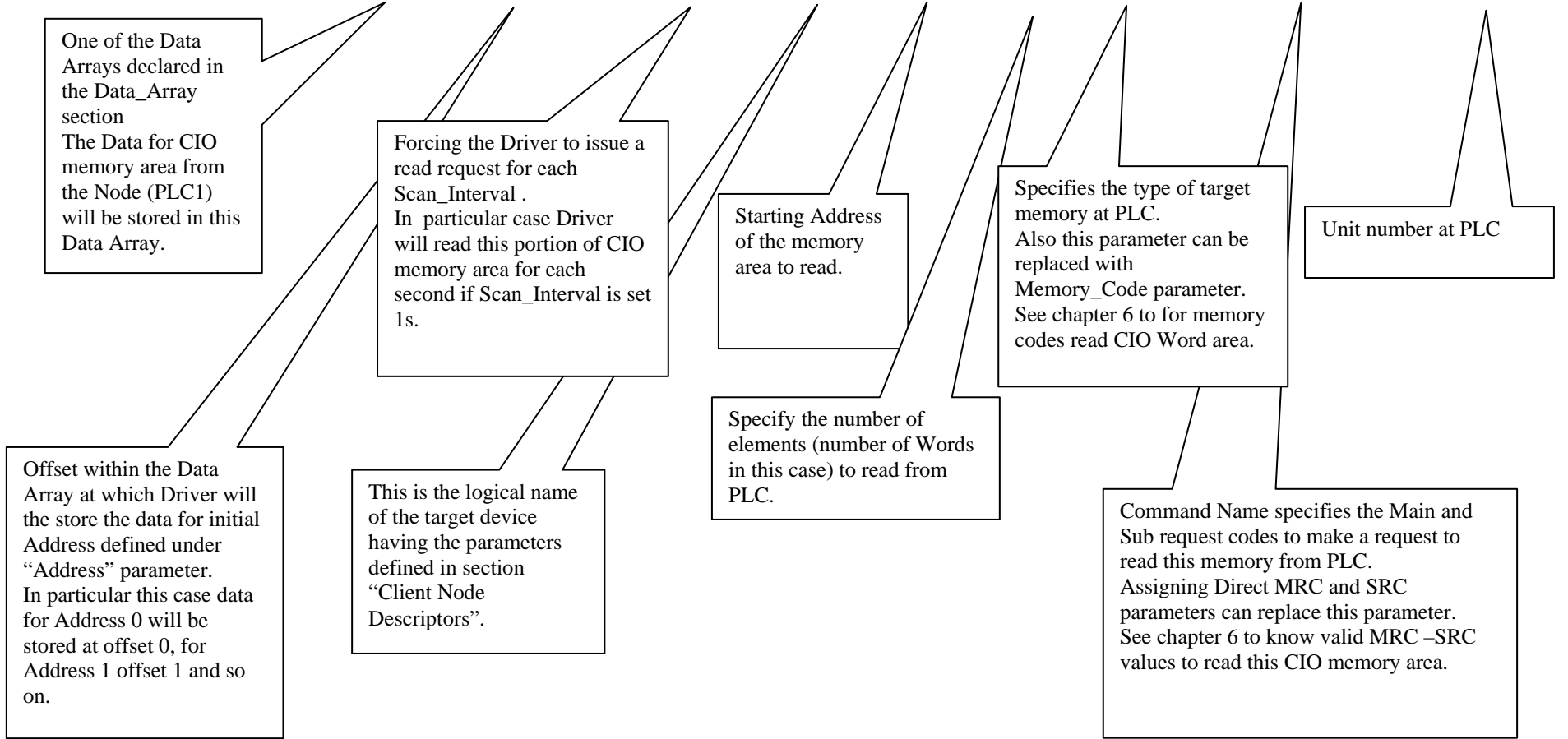
4.1.3. Timing Parameters

Column Title	Function	Legal Values
Scan_Interval	Rate at which data is polled	$\geq 0.001s$

4.1.4. Map Descriptor Example 1 – IO Read

This example provides all the required information to read and write to the IO memory area of the PLC. The following Map Descriptor creates a task for the driver to read the first 20 Words from the CIO memory area and store them in the Data Array DA_CIO. The first word from the PLC will be stored as the first element in the Data Array. Whenever an upstream device writes any element in the Data Array, the Driver will write the same value to the PLC at the corresponding address. This scheme is known as Write-thru. In this example the Driver can write only one value at a time.

Map_Descriptor_Name, CMD_CIO,	Data_Array_Name, DA_CIO,	Data_Array_Offset, 0,	Function, RDBC,	Node_Name PLC1,	Address, 0,	Length, 20,	Data_Type, CIO-WORD,	Command_Name, MEMORY AREA READ,	Unit_number 0
----------------------------------	-----------------------------	--------------------------	--------------------	--------------------	----------------	----------------	-------------------------	------------------------------------	------------------



4.1.5. Map Descriptor Example 2 – IO Write

This example is used to write a value(s) to the PLC. The write is done when the contents of the Data Array are updated (written to by a remote device.). In the previous example, it was shown how a 'read' Map Descriptor can also be used to write by using FieldServer's Write-Through technology.

Map_Descriptor_Name, CMD_CIOw,	Data_Array_Name, DA_CIOw,	Data_Array_Offset, 0,	Function, WRBX,	Node_name PLC1,	Address, 0,	Length, 20,	Data_type, CIO-WORD,	Command_Name, MEMORY AREA WRITE,	Unit_Number 0
-----------------------------------	------------------------------	--------------------------	--------------------	--------------------	----------------	----------------	-------------------------	-------------------------------------	------------------

One of the Data Arrays declared in the Data_Array section (See section 1.05)
The Data in this Data Array will be written to the PLC1.

Driver will fetch consecutive 20 (Length) elements starting from this offset to be written at Node PLC1

Forcing the Driver to issue a write request upon updating this dedicated portion of Data Array. In this case first 20 elements comes under dedicated portion for this map descriptor.

Note : If WRBC , It will Force the Driver to issue a write request for each Scan_Interval
In this particular case Driver will write this portion of CIO memory area for each second.

Specify the number of elements (number of Words in this case) to read from PLC.

Specifies the type of target memory at PLC.
Also this parameter can be replaced with Memory_Code parameter.
See Appendix A to for memory codes read CIO Word area.

Command Name specifies the Main and Sub request codes to make a request to write this memory from PLC.
Assigning Direct MRC and SRC parameters can replace this parameter.
See Appendix A to know valid MRC –SRC values to read this CIO memory area.

Unit number at PLC (Keep mostly zero, or undefined keep it zero)

4.1.6. Map Descriptor Example 3: Clock Read:

This Map Descriptor reads the PLC clock and stores the information in a Data Array DA_CLK. For detail on how the Driver stores clock information see Appendix B.1.

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_Name,	Length,	Command_Name,	Unit_Number
CMD_CLKr,	DA_CLKr,	0,	RDBC,	PLC1,	7,	CLOCK READ,	0

4.1.7. Map Descriptor Example 4: Clock Write:

Write-thru is not possible for Clock write. A dedicated Map Descriptor is required to overwrite the PLC clock. This Map Descriptor overwrites the clock whenever an upstream device updates the DA_CLKw Data Array. See Appendix B.1 for details.

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_Name,	Length,	Command_Name,	Unit_Number
CMD_CLKw,	DA_CLKw,	0,	WRBX,	PLC1,	7,	CLOCK WRITE,	0

Note: All elements must be updated in DA_CLKw to set the clock as expected. Whenever any element updates, the Driver will write all seven values to the PLC along with the one updated value. If all elements are updated by the upstream device in a single operation then the Driver will also set the all elements at the PLC in a single operation.

4.1.8. Map Descriptor Example 5: Read CPU Cycle Times:

This Map Descriptor reads the CPU cycle time at the PLC and stores the data in a Data Array DA_CYCT. Average, maximum and minimum cycle times will be stored in three consecutive locations starting with the location indicated by the Data_Array_Offset parameter. See Appendix B.1

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_Name,	Length,	Command_Name,	Unit_Number
CMD_CYCT,	DA_CYCT,	0,	RDBC,	PLC1,	3,	CLOCK READ,	0s

The Driver stores these values as it gets them from the PLC. Scaling can provided to determine the unit in which the value is stored. Default is 10 times milliseconds.

4.1.9. Map Descriptor Example 6: Run-Stop PLC:

Defining this Map Descriptor Driver can change the PLC mode to STOP (Program), MONITOR or RUN mode. The Driver issues a change mode command to the PLC whenever the value at the declared offset is updated.

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_name	Length	Command_Name	Unit_Number
CMD_RUN_ST,	DA_RUN_ST,	0,	WRBX,	PLC1,	1,	RUN-STOP	0s

The Driver will change PLC modes depending upon the value poked by the upstream device at offset (0 in this case). See section Appendix B.1 for values corresponding to PLC modes.

4.1.10. Map Descriptor Example 7: Read CPU Status:

Defining this Map Descriptor Driver reads the CPU status and stores it in the Data Array DA_STATUS.

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_name	Length	Command_Name	Unit_Number
CMD_STATUS,	DA_STATUS,	0,	REBC,	PLC1,	67,	CPU UNIT STATUS READ	0s

The Driver stores the status information in the named Data Array at consecutive locations starting with Data_Array_Offset.

See Appendix B.1 for further information.

5. Configuring the FieldServer as a FINS Server

For a detailed discussion on FieldServer configuration, please refer to the FieldServer Configuration Manual. The information that follows describes how to expand upon the factory defaults provided in the configuration files included with the FieldServer (See “.csv” files on the driver CD).

This section documents and describes the parameters necessary for configuring the FieldServer to communicate with a FINS Client.

As a Server, the Driver responses to read requests and updates the FieldServer Data Arrays with write requests from the Client.

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for FINS communications, the driver independent FieldServer buffers need to be declared in the “Data Arrays” section, the FieldServer virtual node(s) needs to be declared in the “Server Side Nodes” section, and the data to be provided to the clients needs to be mapped in the “Server Side Map Descriptors” section. Details on how to do this can be found below.

Note that in the tables, * indicates an optional parameter, with the bold legal value being the default.

1.010 Server Side Connection Descriptors

Section Title		
Adapter		
Column Title	Function	Legal Values
Adapter	Adapter Name	N1, N2 ⁴
Protocol	Specify protocol used	Fins, omn_fins or fins_udp
Udp_port_number	Specify UDP port number	0 ,9600 etc

Example

// Server Side Connections		
Adapters		
Adapter, N1,	Protocol, Fins,	Udp_port_number 9600

⁴ Not all ports shown are necessarily supported by the hardware. Consult the appropriate Instruction manual for details of the ports available on specific hardware.

1.011 Server Side Node Descriptors

Section Title		
Nodes		
Column Title	Function	Legal Values
Node_Name	Provide name for node	Up to 32 alphanumeric characters
Node_ID	Virtual Node number of FINS server.	1-126
Protocol	Specify protocol used	Fins, omn_fins or fins_udp
Server_Hold_Timeout*	Specifies time FieldServer will reserve server side connection while waiting for the Client side to update data in Data_Array (if necessary)	>1.0s
Net_Number	Provide the network number of FINS Server network	1-255

Example

```
// Server Side Nodes
Nodes
Node_Name,      Node_ID,      Protocol,      Net_Numer
PLC 1,          1,           Modbus_RTU,   1
```

1.012 Server Side Map Descriptors

5.1.1. FieldServer Specific Map Descriptor Parameters

Column Title	Function	Legal Values
Map_Descriptor_Name	Name of this Map Descriptor	Up to 32 alphanumeric characters
Data_Array_Name	Name of Data Array where data is to be stored in the FieldServer	One of the Data Array names from "Data Array" section above
Data_Array_Offset	Starting location in Data Array	0 to maximum specified in "Data Array" section above
Function	Function of Server Map Descriptor	Server

5.1.2. Driver Specific Map Descriptor Parameters

Column Title	Function	Legal Values
Node_Name	Name of Node to fetch data from	One of the node names specified in "Server Node Descriptor" above
*Data_Type (see Note1 Appendix A.1)	Data type	CIO-WORD, WR-WORD, HR-WORD, AR-WORD, EM-WORD
Length	Length of Map Descriptor	1 to maximum specified in Data Array section above
*Address	Starting address of read block	0,100, 32767 etc see section 6.1 for detail
*Memory_Code (see Note1 Appendix A.1)	Memory code for PLC memory type	B0, B1, B2, B3, 82, 98 and A0 ---- AC
*Command_Name (see Note2 Appendix A.1)	Name of the command	MEMORY AREA READ MEMORY AREA WRITE RUN-STOP CLOCK READ CLOCK WRITE CYCLE TIME READ CPU UNIT STATUS READ
*MRC (see Note2 Appendix A.1)	Main Request Code	1,4,6,7
*SRC (see Note2 Appendix A.1)	Sub Request Code	1,2,20
*Unit_Number	Unit number of CPU at PLC	0,1,2 etc

5.1.3. Map Descriptor Example 1: IO Read

The following Map Descriptor enables the Driver to serve the clients for CIO memory operations. The Command_Name “MEMORY AREA READ” makes this memory readable.

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_Name	Address	Length	Data_Type	Command_Name	Unit_Number
SRV_CIOr,	DA_CIO,	0,	SERVER,	PLC1,	0,	6143,	CIO-WORD	MEMORY AREA READ	0

One of the Data Arrays declared in the Data_Array section. Driver will serve client for CIO operations using this Data Array.

Starting location for data within Data Array

Specify the Driver to Serve Clients.

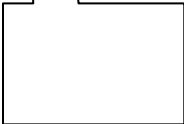
This is the logical name of the server device having the parameters defined in section “Server Node Descriptors”.

Starting Address of the memory area to serve.

Specify the number of elements (number of Words in this case) that this Map Descriptor can serve.

Specifies the type of memory to serve. Also this parameter can be replaced with Memory_Code parameter. See chapter 6 to for memory codes

Command Name specifies the Main and Sub request codes to make a request to read this memory. Assigning Direct MRC and SRC parameters can replace this parameter. See chapter 6 to for MRC -SRC values to enable to read this CIO memory area.



5.1.4. Map Descriptor Example 2: IO Write

This Map Descriptor makes the memory area writable which was made readable by the previous Map Descriptor. Thus memory area can be made read only, write only or read and write enabled.

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_Name	Address	Length	Data_Type	Command_Name	Unit_number
SRV_CIOw,	DA_CIO,	0,	SERVER,	PLC1,	0,	6143,	CIO-WORD	MEMORY AREA WRITE	0

5.1.5. Map Descriptor Example 3: Clock Read:

This Map Descriptor enables the Driver to serve the client with a Clock read request. See Appendix B.1 for further information

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_Name	Length	Command_Name	Unit_Number
SRV_CLKr,	DA_CLKr,	0,	SERVER,	PLC1,	7,	CLOCK READ	0s

5.1.6. Map Descriptor Example 4: Clock Write:

This Map Descriptor enables the Driver to update Clock information when the Client makes a Clock Write request. See section Appendix B.1 for further information.

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_name	Length	Command_Name	Unit_Number
SRV_CLKr,	DA_CLKr,	0,	SERVER,	PLC1,	7,	CLOCK WRITE	0

5.1.7. Map Descriptor Example 5: CPU Cycle Times:

This Map Descriptor enables the Driver to serve Clients with CPU cycle time information upon request. See Appendix B.1 for further information.

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_Name	Length	Command_Name	Unit_Number
SRV_CYCT,	DA_CYCT,	0,	SERVER,	PLC1,	7,	CLOCK READ	0

5.1.8. Map Descriptor Example 6: Run-Stop PLC:

This Map Descriptor enables the Driver to give access to the Client to change the Server’s Operating Mode. See Appendix B.1 for stored values corresponding to PLC modes.

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_Name	Length	Command_Name	Unit_Number
SRV_RUN_ST,	DA_RUN_ST,	0,	SERVER,	PLC1,	1,	RUN-STOP	0

5.1.9. Map Descriptor Example 7: CPU Status:

This Map Descriptor enables the Driver to respond to clients requesting CPU status. See Appendix B.1for further information.

Map_Descriptor_Name,	Data_Array_Name,	Data_Array_Offset,	Function,	Node_Name	Length	Command_Name	Unit_Number
SRV_STATUS,	DA_STATUS,	0,	SERVER,	PLC1,	67,	CPU UNIT STATUS READ	0

Appendix A. Advanced Topics

Appendix A.1. Driver Map Descriptor Parameter's Bounds

Most Map Descriptors need to know the following

- The Data Type to be read or written: Specify Either
 a) Data Type or
 b) Memory Code.
 This is how the driver determines which memory area of the PLC must be processed. (For EM Banks you can only specify the Memory_Code.)
- The Command to be executed: Specify either
 a) The command name or
 b) The MRC/SRC Pair.

Memory Type	Data_Type	Memory_Code	Command_Name	MRC	SRC	Address
I/O Memory	CIO-WORD	B0	MEMORY AREA READ	1	1	0-6143
			MEMORY AREA WRITE	1	2	0-6143
	WR-WORD	B1	MEMORY AREA READ	1	1	0-511
			MEMORY AREA WRITE	1	2	0-511
	HR-WORD	B2	MEMORY AREA READ	1	1	0-511
			MEMORY AREA WRITE	1	2	0-511
	AR-WORD	B3	MEMORY AREA READ	1	1	0-959
			MEMORY AREA WRITE	1	2	448-959
	DM-WORD	82	MEMORY AREA READ	1	1	0-32767
			MEMORY AREA WRITE	1	2	0-32767
	EM-WORD	98 Present Current EM Bank	MEMORY AREA READ	1	1	0-32767
			MEMORY AREA WRITE	1	2	0-32767
	Not Available (Use only memory_code)	A0-AC (Any other EM Bank)	MEMORY AREA READ	1	1	0-32767
			MEMORY AREA WRITE	1	2	0-32767

Appendix A.2. PLC status to execute commands

PLC Type	Memory Type	Command name	Run Mode	Monitor Mode	Program Mode	Access right ⁵	UM Read Protection ⁶	DIP UM Protection ⁷
CS1/CJ1 Series	I/O Memory	MEMORY AREA READ	OK	OK	OK	OK	OK	OK
		MEMORY AREA WRITE	OK	OK	OK	OK	OK	OK
	Operating Mode Changes	RUN	OK	OK	OK	Disabled	OK	OK
		STOP	OK	OK	OK	Disabled	OK	OK
	Time Data Access	CLOCK READ	OK	OK	OK	OK	OK	OK
		CLOCK WRITE	OK	OK	OK	Disabled	OK	OK
	Status Reading	CPU UNIT STATUS READ	OK	OK	OK	OK	OK	OK
		CYCLE TIME READ	OK	OK	Disabled	OK	OK	OK

⁵ Access Right at Other Device: The Access right at other device column tells whether the CPU Unit can or cannot receive a command when another device has the access right to the CPU Unit.

⁶ UM Read Protection: The UM read protection column tells whether the CPU Unit can or cannot receive the command when UM (user memory) is protected from a Peripheral Device.

⁷ DIP Switch UM Protection: The DIP switch UM protection column tells whether the CPU Unit can or cannot receive a command when UM is write-protected by turning ON pin 1 of the DIP switch on the CPU Unit’s front panel.

Appendix A.3. End Codes⁸

The following table lists the main codes and the sub-codes, which combine to form the end code (response code) returned for a FINS command. The probable cause and corrections for each error code are also given. Depending on the command, the destination code will sometimes make a request of another node on a network. The other node is referred to as the third Node.

⁸ This section is a reproduction of Section 5-1-3 End Codes from Omron Communication Commands Reference Manual (M11W342e160603.pdf)

Main code	Subcode	Check point	Probable cause	Correction
00: Normal completion	00: Normal completion	---	---	---
	01: Service canceled	---	Service was cancelled.	Check the capacity of the destination area in the third node.
Data link status		---	Service was cancelled.	Check the status of the data link.
01: Local node error	01: Local node not in network	Network status of local node	Local node is not participating in the network.	Connect the node to the network.
	02: Token timeout	Maximum node address	Token doesn't arrive.	Set the local node to within the maximum node address.
	03: Retries failed	---	Send was not possible during the specified number of retries.	Execute a communications test between the nodes and re-examine the system environment if it fails.
	04: Too many send frames	Number of enabled send frames	Cannot send because maximum number of event frames exceeded.	Check event execution on the network and reduce the number of events per cycle. Increase the maximum number of event frames.
	05: Node address range error	Node address	Node address setting error occurred.	Check the settings of the rotary switches to be sure that the address is within range and that each address is set only once in the same network.
	06: Node address duplication	Node addresses	The same node address has been set twice in the same network.	Change the address of one of the nodes with the same address.
02: Destination node error	01: Destination node not in network	INS indicator on Unit	The destination node is not in the network.	Add the destination node to the network.
	02: Unit missing	Instruction control data	There is no Unit with the specified unit address.	Check the destination unit address.
		03: Third node missing	Instruction control data	The third node does not exist.
	Command data		Broadcasting was specified.	Specify only one node for the third node.
	04: Destination node busy	---	The destination node is busy.	Increase the number of retries or review the system so that the destination node does not receive so many messages.
	05: Response timeout	---	The message was destroyed by noise.	Increase the number of retries or test communications between nodes to see if there is too much noise.
		Instruction control data	The response monitor time is too short.	Increase the length of the response monitor time.
		Error history	The send/receive frame was discarded.	Take appropriate measures based on the error history.

Main code	Subcode	Check point	Probable cause	Correction
03: Controller error	01: Communications controller error	Unit/Board indicators	An error occurred in the communications controller.	Take appropriate measures based on the operation manuals for the related Units/Boards.
	02: CPU Unit error	CPU Unit indicators at destination node	A CPU error occurred in the destination CPU Unit.	Clear the error from the CPU Unit based on its operation manuals.
	03: Controller error	Board indicators	A response was not returned because an error occurred in the Board.	Check network communications status and restart the Board. If the problem persists, replace the Board.
	04: Unit number error	Unit number	The unit number was set incorrectly.	Set the rotary switches correctly, being sure the unit numbers are within range and that each number is used only once.
04: Service unsupported	01: Undefined command	Command code	The Unit/Board does not support the specified command code.	Check the command code.
	02: Not supported by model/version	Unit model and version	The command cannot be executed because the model or version is incorrect.	Check the model number and version.
05: Routing table error	01: Destination address setting error	Routing table	The destination network or node address is not set in the routing tables.	Register the destination network and node in the routing tables.
	02: No routing tables	Routing table	Relaying is not possible because there are no routing tables.	Set routing tables in the source node, designation node, and relay nodes.
	03: Routing table error	Routing table	There is an error in the routing tables.	Set the routing tables correctly.
	04: Too many relays	Network configuration	An attempt was made to send to a network that was over 3 networks away	Reconstruct the networks or change the routing tables so that commands are sent within a range of 3 networks or less.
10: Command format error	01: Command too long	Command data	The command is longer than the maximum permissible length.	Check the command format and correct the command data.
	02: Command too short	Command data	The command is shorter than the minimum permissible length.	Check the command format and correct the command data.
	03: Elements/data don't match	Command data	The designated number of elements differs from the number of write data items.	Check the number of elements and set data for each element.
	04: Command format error	Command data	An incorrect format was used.	Check the command format and correct the command data.
	05: Header error	Routing table	Either the relay table in the local node or the local network table in the relay node is incorrect.	Set the routing tables correctly.

Main code	Subcode	Check point	Probable cause	Correction
11: Parameter error	01: Area classification missing	Memory area code in command data	The specified word does not exist in the memory area or there is no EM Area.	Check the memory areas and parameter codes in the command and correct the command data.
	02: Access size error	Access size specification in command data	The access size specification is incorrect or an odd word address is specified.	Check the memory areas and access size and correct the access size.
	03: Address range error	Starting address in command data	The start address in command process is beyond the accessible area.	Check the area being processed and set the correct range.
	04: Address range exceeded	Starting address and number of elements in command data	The end address in command process is beyond the accessible area.	Check the area being processed and set the correct range.
		Data link tables	The total number of words is beyond the limit.	Correct the data link tables.
	06: Program missing	Program number in command data	FFFF Hex was not specified.	Specify FFFF Hex.
	09: Relational error	Command data	A large–small relationship in the elements in the command data is incorrect.	Check the command data and correct the relationship between the elements.
		Data link table	A node not set in the common link parameters is set as a refresh parameter.	Correct the data link tables.
	0A: Duplicate data access	I/O access in CPU Unit	Differential monitoring was specified during data tracing or data tracing was specified during differential monitoring.	Abort the current process or wait until it ends before executing the command.
		Data link tables	The same node address is specified more than once.	Correct the data link tables.
	0B: Response too long	Number of elements in command data	The response format is longer than the maximum permissible length.	Check the command format and correct the number of elements.
	0C: Parameter error	Parameters in command data	There is an error in one of the parameter settings.	Check the command data and correct the parameters.
Data link table file		There is an error in the file.	Check the contents of the file.	
20: Read not possible	02: Protected	---	The program area is protected.	Release protection from a Programming Device and then execute the command.
	03: Table missing	Table	A table has not been registered.	Register a table.
			There is an error in the table.	Correct the table.
	04: Data missing	---	The search data does not exist.	---
	05: Program missing	Program number in command data	A non-existing program number has been specified.	Check the program numbers and specify a valid one.
	06: File missing	File name and file device	The file does not exist at the specified file device.	Check the path and file name, and correct them.
	07: Data mismatch	Contents of memory being compared	A data being compared is not the same.	Check memory contents and use the correct data.
---			A file read operation failed.	Check the contents of the file.

Main code	Subcode	Check point	Probable cause	Correction
21: Write not possible	01: Read-only	---	The specified area is read-only.	If the area is protected using a switch setting, release protection and then execute the command. If the area is permanently read-only, the command cannot be executed.
	02: Protected Cannot write data link table	---	The program area is protected.	Release protection from a Programming Device and then execute the command.
		PLC Setup	Writing is not possible because automatic data link table generation has been specified.	Change the PLC Setup so that the data link tables can be manually written.
	03: Cannot register	Number of files in file device	The file cannot be created because the limit has been exceeded.	Delete any unnecessary files or create more file memory.
		Number of files open	The maximum number of files has already been opened for the system limit.	Close one or more files and then execute the command.
	05: Program missing	Program number in command data	A non-existing program number has been specified.	Check the program numbers and specify a valid one.
	06: File missing	File name	The file does not exist at the specified file device.	Correct the file name and then execute the command.
	07: File name already exists	File name	A file with the same name already exists in the specified file device.	Change the name of the file being written and then execute the command.
08: Cannot change	Contents of memory being changed	The change cannot be made because doing so would create a problem.	---	
22: Not executable in current mode	01: Not possible during execution	---	The mode is incorrect.	Check the mode.
		Data link status	The data link is operating.	Check the status of the data links.
	02: Not possible while running	---	The mode is incorrect.	Check the mode.
		Data link status	The data links are active.	Check the status of the data links.
	03: Wrong PLC mode	---	The PLC is in PROGRAM mode.	Check the modes of the PLC and computer.
	04: Wrong PLC mode	---	The PLC is in DEBUG mode.	Check the modes of the PLC and computer.
	05: Wrong PLC mode	---	The PLC is in MONITOR mode.	Check the modes of the PLC and computer.
	06: Wrong PLC mode	---	The PLC is in RUN mode.	Check the modes of the PLC and computer.
07: Specified node not polling node	---	The specified node is not the polling node.	Check node functioning as the polling node for the network.	
08: Step cannot be executed	---	The mode is incorrect.	Check step status.	
23: No such device	01: File device missing	Unit configuration	The specified memory does not exist as a file device.	Mount memory or format EM as file memory.
	02: Memory missing	---	There is no file memory.	Check the file memory to see if it is mounted.
	03: Clock missing	---	There is no clock.	Check the model.
24: Cannot start/stop	01: Table missing	Data link tables	The data link tables have not been registered or they contain an error.	Set the data link tables.

Main code	Subcode	Check point	Probable cause	Correction
25: Unit error	02: Memory error	Contents of memory being processed	The contents of memory contains an error.	Transfer the correct contents to memory.
	03: I/O setting error	I/O Unit configuration	The registered I/O tables do not agree with the actual I/O configuration.	Correct the I/O tables or the I/O configuration.
	04: Too many I/O points	Number of I/O in registered I/O tables	There are too many I/O points and remote I/O points registered.	Change the registered I/O table so that it is within the limit.
	05: CPU bus error	CPU bus line	An error occurred in data transfer between the CPU and a CPU Bus Unit.	Check Units, Boards, and cables to be sure they are connected correctly and then execute the ERROR CLEAR command.
	06: I/O duplication	Rack numbers, Unit numbers, and I/O addresses in PLC Setup	The same number/address was set more than once.	Check the PLC Setup and correct the numbers/addresses so that each is used only once.
	07: I/O bus error	I/O bus line	An error occurred in data transfer between the CPU and an I/O Unit.	Check Units, Boards, and cables to be sure they are connected correctly and then execute the ERROR CLEAR command.
	09: SYSMAC BUS/2 error	SYSMAC BUS/2 transmission path	An error occurred in data transfer on the SYSMAC BUS/2 line.	Check Units, Boards, and cables to be sure they are connected correctly and then execute the ERROR CLEAR command.
	0A: CPU Bus Unit error	CPU Bus Unit transmission path	An error occurred in data transfer for a CPU Bus Unit.	Check Units, Boards, and cables to be sure they are connected correctly and then execute the ERROR CLEAR command.
	0D: SYSMAC BUS No. duplication	Word settings	The same word is allocated more than once.	Check the I/O tables and correct the allocations.
	0F: Memory error	Status of memory being processed	A memory error has occurred in internal memory, a memory card, or EM file memory.	For internal memory, write the correct data and then execute the command. For a memory card or EM file memory, the file data has been destroyed. Execute the FILE MEMORY FORMAT command. If the problem persists, replace the memory.
10: SYSMAC BUS terminator missing	---	Terminators have not been set.	Set the terminators correctly.	

Main code	Subcode	Check point	Probable cause	Correction
26: Command error	01: No protection	Command protection for program area	The specified area is not protected.	An attempt was made to clear protection on an area that is not protected, i.e., there is no reason to clear protection.
	02: Incorrect password	---	An incorrect password has been specified.	Specify the correct password.
	04: Protected	---	The specified area is protected.	Clear protection from a Programming Device and then execute the command.
		Number of commands being executed	The node receiving the command is already processing 5 commands.	Wait for current processing to end or force the end of a current process and then execute the command.
	05: Service already executing	---	The service is being executed.	Wait for the service to end or force the end of the service and then execute the command.
	06: Service stopped	---	The service is not being executed.	If necessary, start the service.
	07: No execution right	LNK indicator on Unit/Board	The right to execute the service has not been obtained.	The local node is not in the data link. Execute the command from a node that is participating in the data link.
		---	A response was not returned because a buffer error occurred.	Restart the Board. If the problem persists, replace the Board.
	08: Settings not complete	Settings required before execution	The settings required before executing the service have not been made.	Make the required settings.
	09: Necessary items not set	Command data	The required elements have not been set in the command data.	Check the command format and set the required elements in the command data.
0A: Number already defined	Action numbers and transition numbers of program in program area	The specified action/ transition number has already been registered in a previous program.	Check the action/transition numbers to ones that are not being used and then execute the command.	
	0B: Error will not clear	Cause of error being cleared	The cause of the error has not been removed.	Remove the cause of the error and then execute ERROR CLEAR.
30: Access right error	01: No access right	---	The access right is held by another device. (Online editing is being executed from another node or ACCESS RIGHT ACQUIRE or ACCESS RIGHT FORCE ACQUIRE has been executed by another node.)	Wait until the access right is released and then execute the command. ACCESS RIGHT ACQUIRE or ACCESS RIGHT FORCE ACQUIRE can be executed to obtain the access right, but this may adversely affect processing by the node that previously held the access right.
40: Abort	01: Service aborted	---	Service was aborted with ABORT command.	---

Appendix B. Driver Notes

Appendix B.1. Data Storage

Clock Information

Element	Contents
1	Year (4 for 2004, 12 for 2012)
2	Month
3	Day of Month
4	Hours
5	Minutes
6	Seconds
7	Day of the week (Sunday = 0, Saturday=6)

CPU Cycle Time

Element	Contents
1	Average Cycle Time
2	Maximum Cycle Time
3	Minimum Cycle Time

PLC Mode

Stored Value	PLC Mode
1	STOP
2	MONITOR
3	RUN

CPU Status

ELEMENT NUMBER	CONTENTS	DESCRIPTION
1	1	CPU Status ⁹ 1= Stop 2= Standby (waiting for signal from another Device) 3= Run
	2	
	3	
2	1	CPU Mode 1= PROGRAM 2= MONITOR 3= RUN
	2	
	3	
3	1/0	1: Memory Error
4	1/0	1: I/O Bus Error
5	1/0	1: Duplication Error
6	1/0	1: Fatal Inner Board Error
7	1/0	1: I/O Point overflow
8	1/0	1: I/O Setting Error
9	1/0	1: Program Error
10	1/0	1: Cycle Time Over
11	0	-
12	1/0	1: FALS Error
13-18	0	-
19	1/0	1: FAL Error
20	Unknown	Reserved for System
21	1/0	1: Interrupt Task Error
22	1/0	1: Basic I/O Unit Error
23	Unknown	Reserved for System
24	1/0	1: PLC Setup Error
25	1/0	1: I/O Verification Error
26	1/0	1: Inner Board Error
27	1/0	1: CPU Bus Unit Error
28	1/0	1: Special I/O Unit Error
29	1/0	1: Sysmac Bus Error
30	1/0	1: Battery Error
31	1/0	1: CPU Bus Unit Setting Error
32	1/0	1: Special I/O Unit-Setting Error
33-34	Unknown	Reserved for System
35-42	0	-

9

PLC Value	Driver Value	Meaning
0x00	1	Stop
0x80	2	Standby
0x01	3	Run

Note: During testing it has been found that the PLC Value corresponds with this book value. Hence the Driver stores or sends any other value as it is. User has to do some experiment on it.

ELEMENT NUMBER	CONTENTS	DESCRIPTION
43	1/0	1:Message # 7 Present
44	1/0	1:Message # 6 Present
45	1/0	1:Message # 5 Present
46	1/0	1:Message # 4 Present
47	1/0	1:Message # 3 Present
48	1/0	1:Message # 2 Present
49	1/0	1:Message # 1 Present
50	1/0	1:Message # 0 Present
51	0-65535	Error Code
52-67	A-Z	16 Character Text

Appendix B.2. Driver stats

In addition to the standard FieldServer operating statistics the driver exposes certain key stats in a Data Array if required. An upstream device can then monitor these stats.

Add the following to your configuration file to activate these stats.

```
// Expose Driver Operating Stats.

Data_Arrays
Data_Array_Name,          Data_Format,      Data_Array_Length
fins-stats,              UINT32,          200
```

Stat Number	Stats	Description
0	FINS_BAD_START	Number of Messages received with bad start byte.
1	FINS_NET	Number of Messages received with bad or unsupported Network Number.
2	FINS_STATION	Number of Messages received with bad or unsupported Node Number.
3	FINS_UNIT	Number of Messages received with bad or unsupported Unit Number.
4	FINS_MRC	Number of Messages received with bad or unsupported Main Request Code
5	FINS_SRC	Number of Messages received with bad or unsupported Sub Request Code
6	FINS_SID	Number of Messages received with wrong or Sequence Id.

7	FINS_ECODE	Total number of Messages ignored by PLC due to error.
8	FINS_EMRC	Latest Main Error code returned by PLC.
9	FINS_ESC	Latest Sub Error code returned by PLC.
10	FINS_EMRC	Main and Sub Request codes for which PLC returned Error Code.
11	FINS_ESRC	
12	FINS_STOR_SP	Number of times Driver ignored messages because of insufficient storage space.
13	FINS_NO_WR_THU	Number of blocked attempts to write data via write-thru operation.

Appendix B.3. Driver Error Messages

Some configuration errors might produce an error every time a poll is generated. This will fill the error buffer quickly and not add any clarity. For this reason the driver suppresses subsequent similar messages on the System Error Screen. Thus it is possible for the same error produced by multiple Map Descriptors to produce only one error message on the System Error screen. The driver displays subsequent error messages on the Driver Messages screen.

Note : In the actual message you will see that %d has been replaced by an integer, %s by text indicating a data array name or map descriptor name and %x by two hex characters.

Error Message	Description and Action Required
FINS_UDP#1: Err. Not Enough Space for <%d> items. When offset <%d> DA <%s> MD <%s> MRC <%2X> SRC<%2X>	Data array length needs to be increased. ¹⁰
FINS_UDP#2: Err. MD <%s>. Error Returned by PLC. MRC <%2X> SRC<%2X> Main End Code <%2X> Sub End Code <%2X> See Driver Manual for End Codes detail...	This message shows a request that produced this error at the PLC. Check Appendix A.1 to see the detail for error and action.
FINS_UDP#3: FYI. MRC<%2X> SRC<%2X> Not Supported	This command is not supported by the Driver.
FINS_UDP#11: Err. Station Req/Ext <%d/%d>	Response from PLC is not addressed to this Client.

¹⁰ Some error messages require that the user correct a problem in the configuration. This is done by editing the configuration CSV file, downloading the modified file and resetting the FieldServer to have the changes take effect.

Error Message	Description and Action Required
FINS_UDP#12: Err. Mrc-Src Req/Ext <%2X-%2X/%2X-%2X>	Response from PLC not for current request.
FINS_UDP#21: Err. Message from PLC is Not a Response MRC <%2X> SRC <%2x> MD <%s>	Message from PLC for shown request is not a response.
FINS_UDP#22: Err. Unknown Device with Parameters... Req/Ext dna <%d/%d> node <%d/%d> unit <%d/%d>	Response from PLC ignored because mismatching for any shown parameters.
FINS_UDP#23: Err. Message Sequence Not Matched.Req/Ext <%d/%d> MRC <%2X> SRC <%2X> MD <%s>	Response from PLC ignored because of mismatching request-response sequence Id.
FINS_UDP#24: FYI. Bad Start<%2X>	The message was ignored because the first byte was bad.
FINS_UDP#31 : FYI Net_Number set to 1 Node <%s>	The Net_Number parameter is 0 or not specified. The Driver will automatically change it to the default of 1
FINS_UDP#32 : FYI Udp Port is <%d>	Information about UDP Port number in use.
FINS_UDP#41: FYI. Write-thru not Possible On MD <%s>	Upstream device tried Write-thru operation on other than IO memory area.
FINS_UDP:#51 FYI. You could have used an Array called <%s> to expose diagnostic info. Read Manual.	Define an array to expose stats if required. See Appendix B.1

Appendix C. Troubleshooting tips

Appendix C.1. Connection Tips & Hints

1. Each transaction must be completed in one UDP message fragment. The maximum length of a UDP fragment is 1500 bytes. Thus, if you wanted to read 730 words of PLC memory you will need to configure two MD's. The one should have a length of 729 and the other a length of 1. The reason is that when the length is set to 729, then 729 words (or 1500 bytes including FINS and UDP header) of data are read and this is the maximum for one message fragment. Similarly as a Server, the Driver can handle transactions composed of a maximum 1500 bytes (or 729 Words).
2. Ensure that the IP Address of the PLC provided under IP_Address parameter in configuration is correct
3. Ensure the Node number of the FieldServer on the network is mentioned under System_Node_Id parameter in configuration file
4. Ensure that the "Udp_port_number" parameter has the correct value in the configuration file
5. Omron PLC's respond to remote FINS polls by sending UDP response messages on the ethernet network. **The PLC does not use the IP address of the incoming poll to determine the IP address it must respond to.** The PLC builds the IP address it will send the response to using
 - 1) Its own IP address and the Node ID of the polling device or
 - 2) A routing table built using Omron software. To find out more information consult the Omron Manual W343-E1-3, Chapter 3 provides more information.

When the PLC uses its own IP address to build the IP address it will send the response to, then special consideration should be taken if the PLC IP address is Class A or B.

On a Class B network, the PLC uses the 1st two bytes of its own IP address, sets the 3rd byte to zero and sets the 4th byte equal to the polling station's Node_Id.

Example: - Class B Addressing

A FieldServer with IP address=192.168.1.81 and system_node_id=34 polls for data from a PLC with IP address =192.168.2.33 and Subnet mask = 255.255.0.0

The PLC responds to 192.168.2.33
 255.255.0.0
 192.168.x.y

Where x is always set to zero and
y is equal to 34 (The PLC obtains the node ID of 34 by inspecting the contents of the poll message.)

Therefore the PLC responds to [192.168.0.34](#) which is clearly the wrong address and the FieldServer will not see the response.

Example: - Class A Addressing

A FieldServer with IP address=192.168.1.81 and system_node_id=34 polls for data from a PLC with IP address =192.168.2.[33](#) and Subnet mask = 255.255.0.0

The PLC responds to 192.168.2.[33](#)
 255.0.0.0
 192.x.x.y

Where x is always set to zero and
y is equal to 34 (The PLC obtains the node ID of 34 by inspecting the contents of the poll message.)

Therefore the PLC responds to [192.0.0.34](#) which is clearly the wrong address and the FieldServer will not see the response.

Thus, if the PLC is using Class A/B IP addressing then the 3rd (Class B) or the 2nd and 3rd (Class A) bytes of the address must be zero as must the FieldServer's if communications are to work.

There are ways around this problem, we believe. The solution requires usage of the Omron PLC routing table. For more help consult with Omron's Tech Support.

Appendix D. Revision History

Date	Resp	Format	Driver Ver.	Doc. Rev.	Comment
4/17/04	SSS		0.00	0	Issued for PMC review.
4/20/04	PMC		0.00	1	Reviewed and made some reference changes, some type changes, changed some wording and included Omron manual excerpts for IP address setting.
4/20/04	SSS		0.00	2	Changed section 7.1.4, Value from PLC does not agree with book value. Updated section 8.1 Maximum words are 729 Not 750. This is the limit for one UDP message fragment because of FINS and UDP header.
4/20/04	SSS		0.00	2	Issued for Release.
5/20/04	Meg	Meg	0.00	3	Reformatted document. Changed language and grammar. Improved readability of document. Updated according to DUR0356
6/14/04	JD		0.00	4	Releasing
8/26/04	Meg	Meg	0.00	5	Updated according to DUR0372. Changed Appendix numbers to letters and updated cross referencing.
12/29/04	JD		0.00	6	Releasing