



Where Automation Connects.



MVI69-AFC

CompactLogix Platform
Gas and Liquid Flow Computer

June 24, 2015

USER MANUAL

Your Feedback Please

We always want you to feel that you made the right decision to use our products. If you have suggestions, comments, compliments or complaints about our products, documentation, or support, please write or call us.

ProSoft Technology

5201 Truxtun Ave., 3rd Floor
Bakersfield, CA 93309
+1 (661) 716-5100
+1 (661) 716-5101 (Fax)
www.prosoft-technology.com
support@prosoft-technology.com

MVI69-AFC User Manual

June 24, 2015

ProSoft Technology[®], is a registered copyright of ProSoft Technology, Inc. All other brand or product names are or may be trademarks of, and are used to identify products and services of, their respective owners.

In an effort to conserve paper, ProSoft Technology no longer includes printed manuals with our product shipments. User Manuals, Datasheets, Sample Ladder Files, and Configuration Files are provided on the enclosed DVD and are available at no charge from our web site: <http://www.prosoft-technology.com>

Important Installation Instructions

Power, Input, and Output (I/O) wiring must be in accordance with Class I, Division 2 wiring methods, Article 501-4 (b) of the National Electrical Code, NFPA 70 for installation in the U.S., or as specified in Section 18-1J2 of the Canadian Electrical Code for installations in Canada, and in accordance with the authority having jurisdiction. The following warnings must be heeded:

WARNING - EXPLOSION HAZARD - SUBSTITUTION OF COMPONENTS MAY IMPAIR SUITABILITY FOR CLASS I, DIV. 2;

WARNING - EXPLOSION HAZARD - WHEN IN HAZARDOUS LOCATIONS, TURN OFF POWER BEFORE REPLACING OR WIRING MODULES

WARNING - EXPLOSION HAZARD - DO NOT DISCONNECT EQUIPMENT UNLESS POWER HAS BEEN SWITCHED OFF OR THE AREA IS KNOWN TO BE NON-HAZARDOUS.
THIS DEVICE SHALL BE POWERED BY CLASS 2 OUTPUTS ONLY.

MVI (Multi Vendor Interface) Modules

WARNING - EXPLOSION HAZARD - DO NOT DISCONNECT EQUIPMENT UNLESS POWER HAS BEEN SWITCHED OFF OR THE AREA IS KNOWN TO BE NON-HAZARDOUS.

AVERTISSEMENT - RISQUE D'EXPLOSION - AVANT DE DÉCONNECTER L'ÉQUIPEMENT, COUPER LE COURANT OU S'ASSURER QUE L'EMPLACEMENT EST DÉSIGNÉ NON DANGEREUX.

Warnings

North America Warnings

- A** Warning - Explosion Hazard - Substitution of components may impair suitability for Class I, Division 2.
- B** Warning - Explosion Hazard - When in hazardous locations, turn off power before replacing or rewiring modules.
Warning - Explosion Hazard - Do not disconnect equipment unless power has been switched off or the area is known to be non-hazardous.
- C** Suitable for use in Class I, Division 2 Groups A, B, C and D Hazardous Locations or Non-Hazardous Locations.

ATEX Warnings and Conditions of Safe Usage

Power, Input, and Output (I/O) wiring must be in accordance with the authority having jurisdiction.

- A** Warning - Explosion Hazard - When in hazardous locations, turn off power before replacing or wiring modules.
- B** Warning - Explosion Hazard - Do not disconnect equipment unless power has been switched off or the area is known to be non-hazardous.
- C** These products are intended to be mounted in an IP54 enclosure. The devices shall provide external means to prevent the rated voltage being exceeded by transient disturbances of more than 40%. This device must be used only with ATEX certified backplanes.
- D** DO NOT OPEN WHEN ENERGIZED.

Warning: This module is not hot-swappable! Always remove power from the rack before inserting or removing this module, or damage may result to the module, the processor, or other connected devices.

Battery Life Advisory

The MVI69 modules use a rechargeable Lithium Vanadium Pentoxide battery to backup the real-time clock and CMOS. The battery should last for the life of the module. The module must be powered for approximately twenty hours before the battery becomes fully charged. After it is fully charged, the battery provides backup power for the CMOS setup and the real-time clock for approximately 21 days. When the battery is fully discharged, the module will revert to the default BIOS and clock settings.

Note: The battery is not user replaceable.

Markings

Electrical Ratings

- Backplane Current Load: 800 mA @ 5.1 Vdc
- Power Supply Distance Rating: 2
- Operating Temperature: 0°C to 60°C (32°F to 140°F)
- Storage Temperature: -40°C to 85°C (-40°F to 185°F)
- Relative Humidity: 5% to 95% (with no condensation)
- All phase conductor sizes must be at least 1.3 mm² and earth ground conductors must be at least 4mm²

Label Markings

Class I, Division 2 Groups A, B, C, D

II 3 G

Ex nA IIC X

0°C ≤ Ta ≤ +60°C

II - Equipment intended for above ground use (not for use in mines)

3 - Category 3 equipment, investigated for normal operation only

G - Equipment protected against explosive gasses.

Agency Approvals and Certifications

Agency	Applicable Standard(s)
ATEX	EN 60079-0:2006, EN 60079-15:2005
DNV	DET NORSKE VERITAS Test 2.4
CE	EMC-EN61326-1:2006; EN61000-6-4:2007
CB Safety	CA/10533/CSA, IEC 61010-1 Ed. 2, CB 243333-2056722 (2090408)
GOST-R	EN 61010



Contents

Your Feedback Please	2
Important Installation Instructions	2
MVI (Multi Vendor Interface) Modules	2
Warnings	2
Battery Life Advisory	3
Markings.....	3
1 Introduction	9
1.1 Update Notice.....	10
1.2 MVI69-AFC Module	11
1.3 Configuration Modification Lockout and Seal	12
2 Start Here	15
2.1 System Requirements	16
2.2 Package Contents	16
2.3 Setting Jumpers	17
2.4 Installing the Module in the Rack	18
3 Ladder Logic Implementation	23
3.1 Introduction.....	25
3.2 Overview.....	25
3.3 How does the sample ladder work?	28
3.3.1 Block IDs (Block Sequence Numbers)	29
3.4 CompactLogix Sample Ladder File	37
3.4.1 Input/Output Blocks for Data Transfer.....	37
3.4.2 Input/Output Transactions	39
3.4.3 Wallclock	43
3.4.4 Site Status	44
3.4.5 Meter Profile	45
3.4.6 Enable Meter	46
3.4.7 Disable Meter	46
3.4.8 Meter Process Variables	47
3.4.9 Molar Analysis (For Gas Product Only).....	50
3.4.10 Select Stream (Firmware version 2.05.000 or later)	52
3.4.11 Meter Calculation Results	53
3.4.12 Checking Meter Alarms	56
3.4.13 Modbus Master.....	57
3.4.14 Modbus Pass-Thru	61
3.4.15 Modbus Gateway	61
3.4.16 Reset Resettable Accumulator.....	63
3.4.17 Write Hourly/Daily Archive.....	64
3.4.18 Connecting Your PC to the Processor	65
3.4.19 Downloading the Sample Program to the Processor	66
3.5 MicroLogix Sample Ladder File.....	69
3.5.1 Status	69
3.5.2 Enable/Disable Meters	72

3.5.3	WallClock.....	73
3.5.4	Process Calculation.....	73
3.5.5	Molar Analysis Update.....	76
3.5.6	Meter Signals.....	77
3.5.7	Archive Fetch.....	79
3.5.8	Modbus Pass-Thru.....	80
3.5.9	Modbus Master.....	81
3.5.10	Modbus Gateway.....	82
4	Module Configuration	85
4.1	Connect your PC to the Module.....	85
4.2	Install AFC Manager.....	86
4.2.1	System Requirements.....	86
4.3	Starting AFC Manager.....	87
4.4	Using AFC Manager.....	88
4.4.1	Starting a New Project.....	88
4.4.2	Loading an Existing project.....	89
4.4.3	Printing the Configuration Report.....	90
4.4.4	Converting a Project.....	90
4.4.5	Resetting Configuration Parameters.....	91
4.4.6	Downloading the Project to the Module.....	92
4.4.7	Verifying Correct Operation.....	93
4.5	Creating Your Application.....	94
4.6	Module Initialization.....	94
5	Diagnostics and Troubleshooting	95
5.1	User LEDs.....	96
5.1.1	App Status LED.....	96
5.1.2	BP Act and P1, P2, or P3.....	96
5.2	BBRAM LEDs.....	96
5.3	Meter Alarms.....	97
5.4	Checksum Alarms.....	101
5.5	Events.....	101
5.6	Audit Scan.....	101
6	Reference	107
6.1	General Specifications.....	107
6.1.1	On-line Communication & Configuration.....	108
6.1.2	Reports.....	108
6.1.3	Modbus Interface.....	109
6.1.4	Configurable Options.....	109
6.1.5	Sealable Parameters.....	110
6.1.6	Supported Meters.....	111
6.1.7	Hardware Specifications.....	112
6.2	Measurement Standards.....	113
6.2.1	Basic Metering According to Meter type.....	114
6.2.2	Liquid Correction Factor Details.....	116
6.3	Meter Channels.....	117
6.3.1	General Features.....	119
6.3.2	Gas Product Overview.....	122

6.3.3	Liquid Product Overview	123
6.3.4	Linear (Pulse) Meter Overview.....	124
6.3.5	Differential (Orifice) Meter Overview	125
6.3.6	Wedge Meter Applications	127
6.4	Modbus Database	127
6.4.1	AFC Modbus Address Space.....	127
6.4.2	Primary Slave	128
6.4.3	Virtual Slave	130
6.4.4	Modbus Addressing Common to Both Primary and Virtual Slaves.....	133
6.4.5	Modbus Port configuration	135
6.5	Modbus Communication.....	137
6.5.1	Communication Parameters.....	137
6.5.2	Port Options	138
6.5.3	Modbus Master.....	139
6.5.4	Modbus Pass-Through	141
6.6	Accumulators.....	142
6.6.1	Accumulator Totalizer and Residue	142
6.6.2	Accumulator Types.....	143
6.6.3	Net Accumulator Calculation	147
6.6.4	Frequently Asked Questions	148
6.7	Archives.....	148
6.7.1	Archive Overview	148
6.7.2	Archive Generation.....	148
6.7.3	Archive Monitor	150
6.7.4	Archive Types.....	155
6.7.5	Archive Order	155
6.7.6	Archive Options	158
6.7.7	Archive Locations	159
6.7.8	Configurable Archive Registers.....	160
6.7.9	Editing the Archive Structure.....	166
6.7.10	Extended Archives	167
6.7.11	Archive Reports.....	170
6.7.12	Archive Data Format	171
6.8	Events.....	178
6.8.1	The Event Log	178
6.8.2	Event Log Structures.....	179
6.8.3	Event Id Tag	180
6.8.4	Event-triggered Archives and Accumulator Resets	181
6.8.5	Period-end Events.....	181
6.8.6	Loggable Events.....	182
6.8.7	Special Events.....	183
6.8.8	Site Data Point Events	184
6.8.9	Meter Data Point Events	184
6.8.10	Stream Data Point Events	187
6.8.11	"Rkv" Notes	188
6.8.12	Event Numbers.....	189
6.9	Security (Passwords)	192
6.9.1	Hard Password.....	193
6.10	Cable Connections	196
6.10.1	RS-232 Configuration/Debug Port	196
6.10.2	RS-232 Application Port(s).....	196
6.10.3	RS-422	199
6.10.4	RS-485 Application Port(s).....	199
6.10.5	DB9 to RJ45 Adaptor (Cable 14)	200

6.11	Frequently Asked Questions.....	201
6.11.1	How does the module work?	201
6.11.2	Why should I use the AFC Manager?	201
6.11.3	Why can't the AFC Manager connect to the module?	201
6.11.4	Why do I have to enable or disable a meter?	202
6.11.5	Why does the card not calculate results, or why did it stop calculating results? ..	202
6.11.6	What is the Virtual Modbus Slave?	202
6.11.7	How does the AFC Manager transfer the configuration to the module?	203
6.11.8	What is the password used for?	203
6.11.9	Why do I receive an Illegal Data Value warning when I try to write a meter configuration or download the entire configuration to the module?	203
6.11.10	Why is the Molar Analysis button disabled?	204
6.11.11	Why does the AFC Manager show a "Communication Timeout" warning?	204
6.11.12	What is the difference between Net Accumulator and Gross Accumulator?	204
6.11.13	What are the accumulator's totalizer and residue values?	204
6.11.14	Do I have to enter all molar concentrations for the gas product?	204
6.11.15	Can I update the molar concentration values dynamically?	204
6.11.16	Why do the accumulator values not update?	205
6.11.17	What is the Wallclock?.....	205
6.11.18	Can I read the Primary (or Virtual) Slave values using the AFC Manager?	205
6.11.19	When are the archives generated?	205
7	Support, Service & Warranty	207
7.1	Contacting Technical Support.....	207
	Warranty Information	208
Index		209

1 Introduction

In This Chapter

- ❖ Update Notice..... 10
- ❖ MVI69-AFC Module..... 11
- ❖ Configuration Modification Lockout and Seal 12

The MVI69-AFC Flow Computer module performs measurement of hydrocarbon gases and liquids using currently accepted industry measurement standards. The module consists of a single-slot solution for Rockwell Automation chassis. To obtain its process inputs for calculations, the module uses the process data collected by analog and pulse I/O modules. The processor transfers this data to the AFC module, which then calculates flow rates, accumulated volumes, and accumulated mass. The results of the calculations are transferred back to the processor for use in the application logic, or for transfer to a SCADA host.

The module has two serial communication ports and one ethernet port for Modbus communication allowing easy access to a remote Modbus device. The module supports Modbus-TCP and works as a Modbus slave or master device.

As discussed later in this manual, the internal Modbus database can be accessed by a Modbus Master device and by the processor (using the Modbus Gateway Function).

The AFC Manager software is used for easy meter configuration and application monitoring.

The following section provides a sample application where input data is transferred from the transmitters to analog input cards on the Rockwell Automation rack and the values are transferred from the processor to the module (the module supports floating-point, scaled integer, or 4 to 20 mA format).

For Pulse meter applications, the pulse count and pulse frequency values are typically transmitted through high-speed counter modules in the rack.

The module performs the flow calculation based on the values transferred through the backplane. The calculation results can be read to the processor or polled from a remote Modbus master unit connected to one of the communication ports.

1.1 Update Notice

If your module measures liquids, please read this notice before upgrading from version 2.04 (or earlier) to 2.05 (or later).

For compliance with new measurement standards, the AFC version 2.05 has introduced several new liquid product groups. In particular, the two non-refined liquid product groups of version 2.04, which covered the entire density range of crudes and NGLs, have each been split into two separate product groups, one for the higher density range of crudes and the other for the lower density range of NGLs. If your module has meter channels configured for either "Crude, NGL" or "Oil-water emulsion", you should decide **before upgrading the firmware** the new product group (light or heavy) to which each such channel should be assigned. This assignment will be performed during the upgrade process and will preserve all other configuration and historical records including accumulator values and archives, in contrast to changing a product group after the upgrade which resets the meter configuration and erases all historical records. Meter channels configured for "Gas" or "Refined products" are not affected.

AFC Manager exhibits the same behavior when converting a project between versions 2.04 (or earlier) and 2.05 (or later).

The criterion for assigning the new product group depends on the density units and the Default Reference Density, as described in the following tables:

Density Units = kg/m³

Version 2.04 Product Group	Default Reference Density	Version 2.05 Product Group
Crude, NGL	= 0 OR ≥ 610.0	Crude oils, JP4
Crude, NGL	> 0 AND < 610.0	NGLs, LPGs
Oil Water Emulsion	= 0 OR ≥ 610.0	Oil-water emulsion (Crd)
Oil Water Emulsion	> 0 AND < 610.0	Oil-water emulsion (NGL)

Density Units = Rd/60

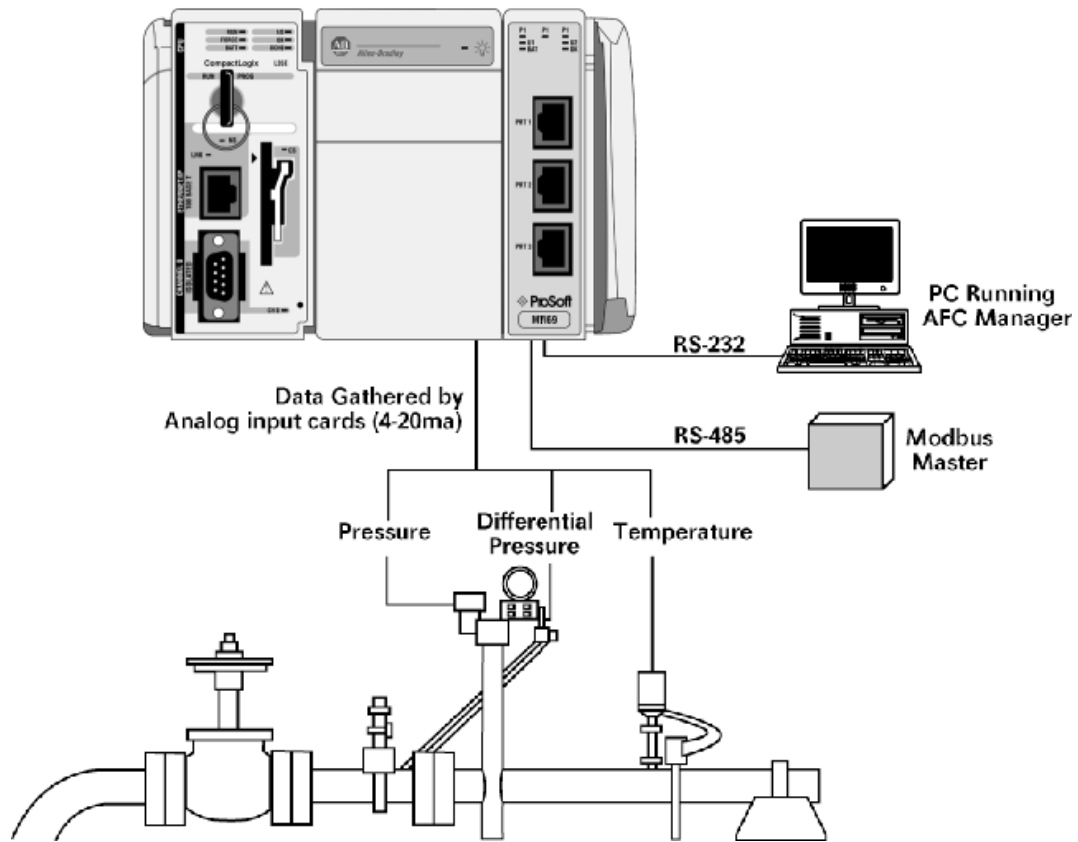
Version 2.04 Product Group	Default Reference Density	Version 2.05 Product Group
Crude, NGL	= 0 OR ≥ 0.6100	Crude oils, JP4
Crude, NGL	> 0 AND < 0.6100	NGLs, LPGs
Oil Water Emulsion	= 0 OR ≥ 0.6100	Oil-water emulsion (Crd)
Oil Water Emulsion	> 0 AND < 0.6100	Oil-water emulsion (NGL)

Due to roundoff error of numeric conversions, a Relative Density very close to the cutoff value of 0.6100 may cause the module to assign the new product group opposite to the one that was intended. Before upgrading, change the Default Reference Density to a number significantly different from 0.6100, such as 0.6110 (to target Crude) or 0.6090 (to target NGLs). You may change it back to the correct value after the upgrade.

Density Units = API Gravity

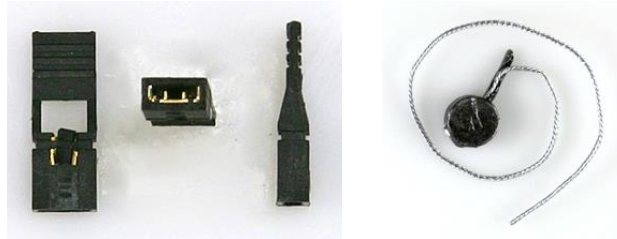
Version 2.04 Product Group	Default Reference Density	Version 2.05 Product Group
Crude, NGL	= 0 OR ≤ 100.0	Crude oils, JP4
Crude, NGL	> 0 AND > 100.0	NGLs, LPGs
Oil Water Emulsion	= 0 OR ≤ 100.0	Oil-water emulsion (Crd)
Oil Water Emulsion	> 0 AND > 100.0	Oil-water emulsion (NGL)

1.2 MVI69-AFC Module



1.3 Configuration Modification Lockout and Seal

The MVI69-AFC application configuration can be certified and sealed with a user-installable Lockout jumper and a tamper-evident lead seal. The Lockout jumper and seal are commonly required for Weights & Measures certification, or custody transfer applications.



Important: When the jumper is installed, the module will not accept configuration changes to Sealable Parameters, which are parameters that affect the accuracy of flow Sealable Parameters (page 110). Before breaking the seal to remove the jumper, you should verify the steps required to recertify the module with the appropriate regulatory agency.

For more information on sealing procedures, refer to "Sealing Provisions", on page 8 of the *MVI56-AFC Custody Transfer Certification* document, which is available from the ProSoft Technology web site at <http://www.prosoft-technology.com/content/view/full/4613>

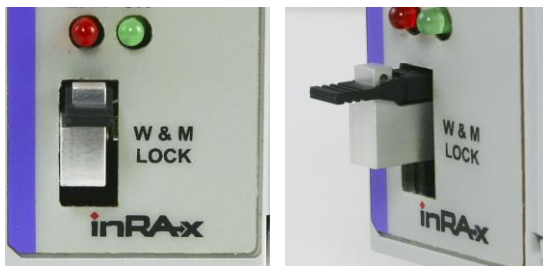
To install the Lockout jumper and seal, follow these steps.



- 1 Locate the Lockout Jumper pins and Lockout Block, labeled "W & M Lock" inside the module door, and below the BBRAM ERR and OK LEDs.



- 1 Install the provided Lockout Jumper to connect the two pins.



- 1 Carefully slide the Lockout Block up through the hole in the Lockout Jumper. Be careful not to bend or break the jumper block pins. When the Lockout Block is positioned correctly, it will expose a hole in the block, through which you may pass the seal wire.



- 1 Slide the seal wire through the hole in the Lockout Block. Pass the wire through the slot in the lead seal, and then crimp the lead seal around the wire.

Once sealed, the Lockout block and jumper cannot be removed without damaging the seal, the block, or the jumper.

2 Start Here

In This Chapter

❖ System Requirements	16
❖ Package Contents	16
❖ Setting Jumpers	17
❖ Installing the Module in the Rack.....	18

To get the most benefit from this User Manual, you should have the following skills:

- **Rockwell Automation® RSLogix™ software:** launch the program, configure ladder logic, and transfer the ladder logic to the processor
- **Microsoft Windows®:** install and launch programs, execute menu commands, navigate dialog boxes, and enter data
- **Hardware installation and wiring:** install the module, and safely connect Liquid and Gas Hydrocarbons Flow Computer and CompactLogix or MicroLogix devices to a power source and to the MVI69-AFC module's application port(s)

2.1 System Requirements

The MVI69-AFC module requires the following minimum hardware and software components:

- *Rockwell Automation CompactLogix* processors and *MicroLogix 1500 LRP* processors except 1769-L23E-QBFC1B, 1769-L16x, and 1769-L18x. Must have compatible power supply and one free slot in the rack, for the MVI69-AFC module. The module requires 800 mA of available power.

Important: The MVI69-AFC module has a power supply distance rating of 2 (L43 and L45 installations on first 2 slots of 1769 bus).

- *Rockwell Automation RSLogix 5000 (CompactLogix)* or *RSLogix 500 (MicroLogix)* programming software
- *Rockwell Automation RSLinx* communication software
- *Pentium® II* 450 MHz minimum. *Pentium III* 733 MHz (or better) recommended
- Supported operating systems:
 - *Microsoft Windows XP Professional* with *Service Pack 1* or *2*
 - *Microsoft Windows 2000 Professional* with *Service Pack 1, 2, or 3*
 - *Microsoft Windows Server 2003*
- 128 Mbytes of RAM minimum, 256 Mbytes of RAM recommended
- 100 Mbytes of free hard disk space (or more based on application requirements)
- 256-color VGA graphics adapter, 800 x 600 minimum resolution (True Color 1024 × 768 recommended)
- DVD drive
- HyperTerminal or other terminal emulator program capable of file transfers using Ymodem protocol.

Note: MVI69/PS69 modules will not work with CompactLogix L4x processors using RSLogix 5000 v17. All other processor combinations and RSLogix versions will work correctly.

2.2 Package Contents

The following components are included with your MVI69-AFC module, and are all required for installation and configuration.

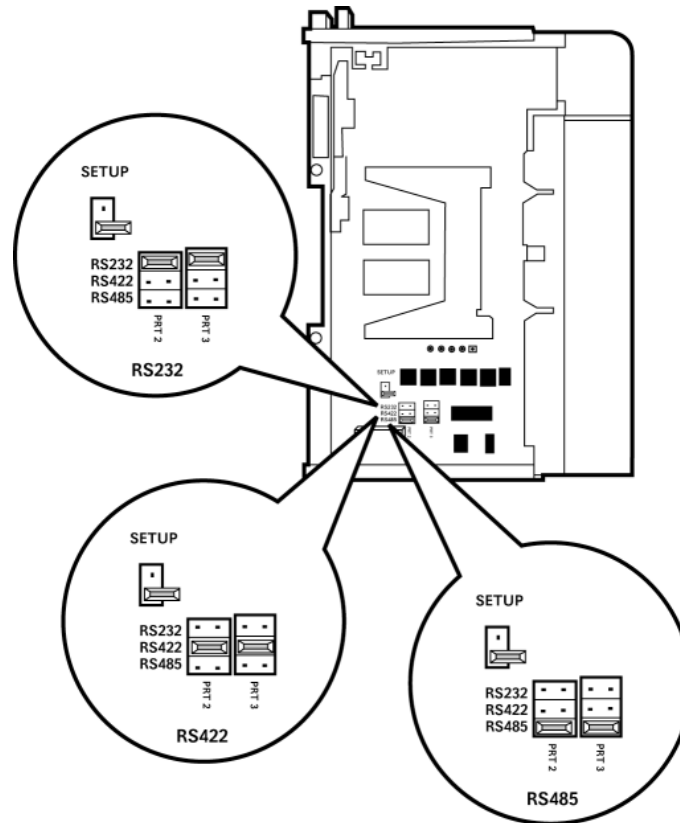
Important: Before beginning the installation, please verify that all of the following items are present.

Qty.	Part Name	Part Number	Part Description
1	MVI69-AFC Module	MVI69-AFC	Gas and Liquid Flow Computer
1	Cable	Cable #15, RS232 Null Modem	For RS232 Connection to the CFG Port
3	Cable	Cable #14, RJ45 to DB9 Male Adapter cable	For DB9 Connection to Module's Port
2	Adapter	1454-9F	Two Adapters, DB9 Female to Screw Terminal. For RS422 or RS485 Connections to Port 1 and 2 of the Module
1	ProSoft Solutions DVD	DVD-001	Contains sample programs, utilities and documentation for the MVI69-AFC module

If any of these components are missing, please contact ProSoft Technology Support for replacement parts.

2.3 Setting Jumpers

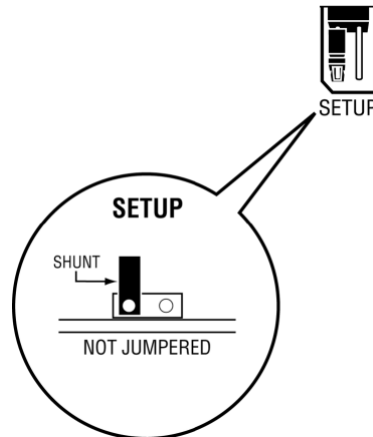
When the module is manufactured, the port selection jumpers are set to RS-232. To use RS-422 or RS-485, you must set the jumpers to the correct position. The following diagram describes the jumper settings.



Note: Jumper pin placement on the circuit board may vary.

The Setup Jumper acts as "write protection" for the module's firmware. In "write protected" mode, the Setup pins are not connected, and the module's firmware cannot be overwritten. The module is shipped with the Setup jumper OFF. If an update of the firmware is needed, apply the Setup jumper to both pins.

The following illustration shows the MVI69-AFC jumper configuration, with the Setup Jumper OFF.



2.4 Installing the Module in the Rack

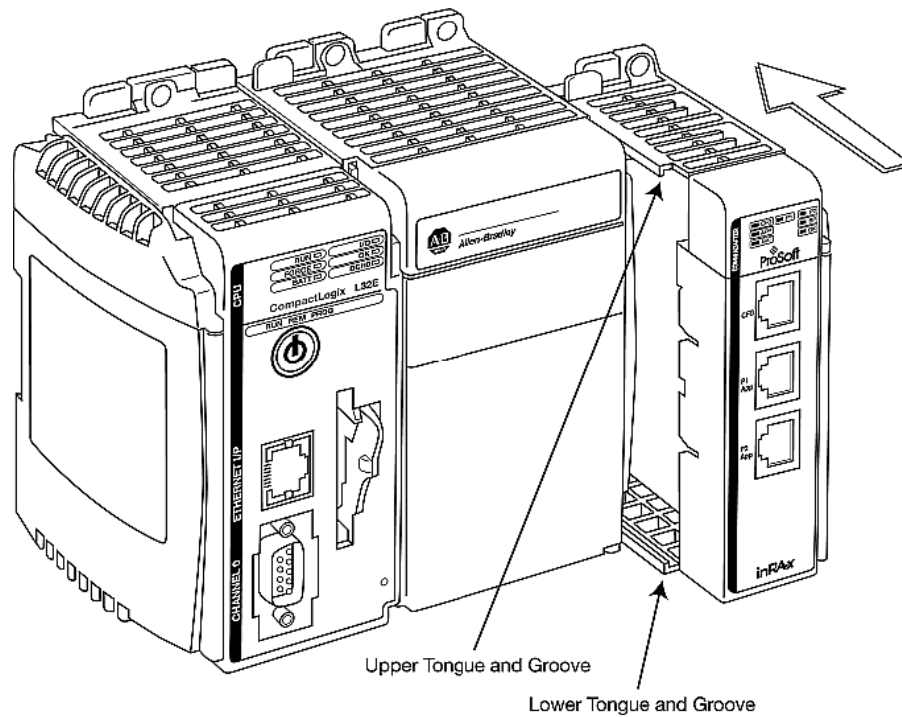
Make sure the processor and power supply are installed and configured before installing the MVI69-AFC module. Refer to the Rockwell Automation product documentation for installation instructions.

Warning: Please follow all safety instructions when installing this or any other electronic devices. Failure to follow safety procedures could result in damage to hardware or data, or even serious injury or death to personnel. Refer to the documentation for each device to be connected to verify that suitable safety procedures are in place before installing or servicing the device.

After you verify the jumper placements, insert the MVI69-AFC into the rack. Use the same technique recommended by Rockwell Automation to remove and install CompactLogix or MicroLogix modules.

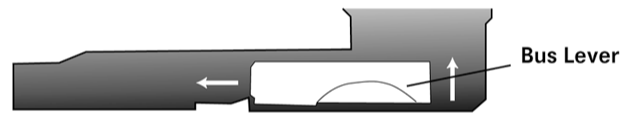
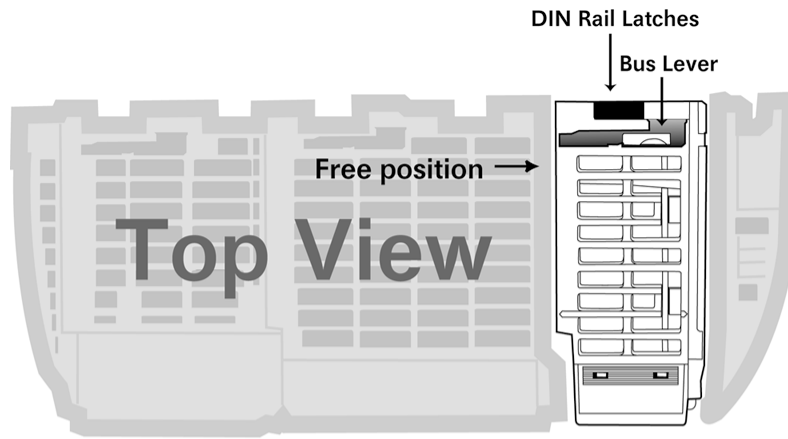
Warning: This module is not hot-swappable! Always remove power from the rack before inserting or removing this module, or damage may result to the module, the processor, or other connected devices.

- 1 Align the module using the upper and lower tongue-and-groove slots with the adjacent module and slide forward in the direction of the arrow.

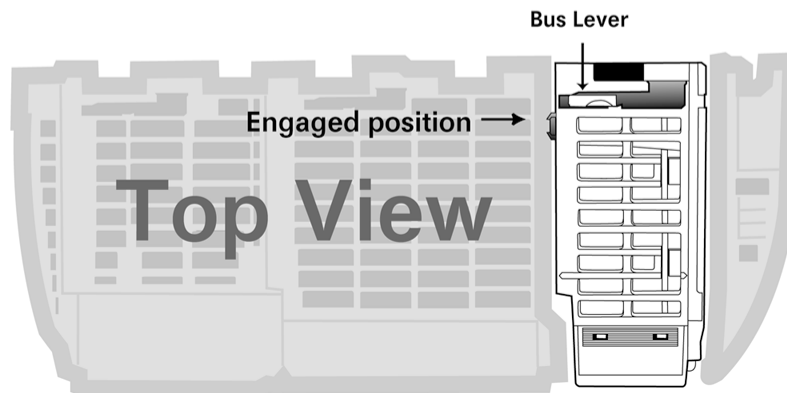


- 2 Move the module back along the tongue-and-groove slots until the bus connectors on the MVI69 module and the adjacent module line up with each other.

- 3 Push the module's bus lever back slightly to clear the positioning tab and move it firmly to the left until it clicks. Ensure that it is locked firmly in place.

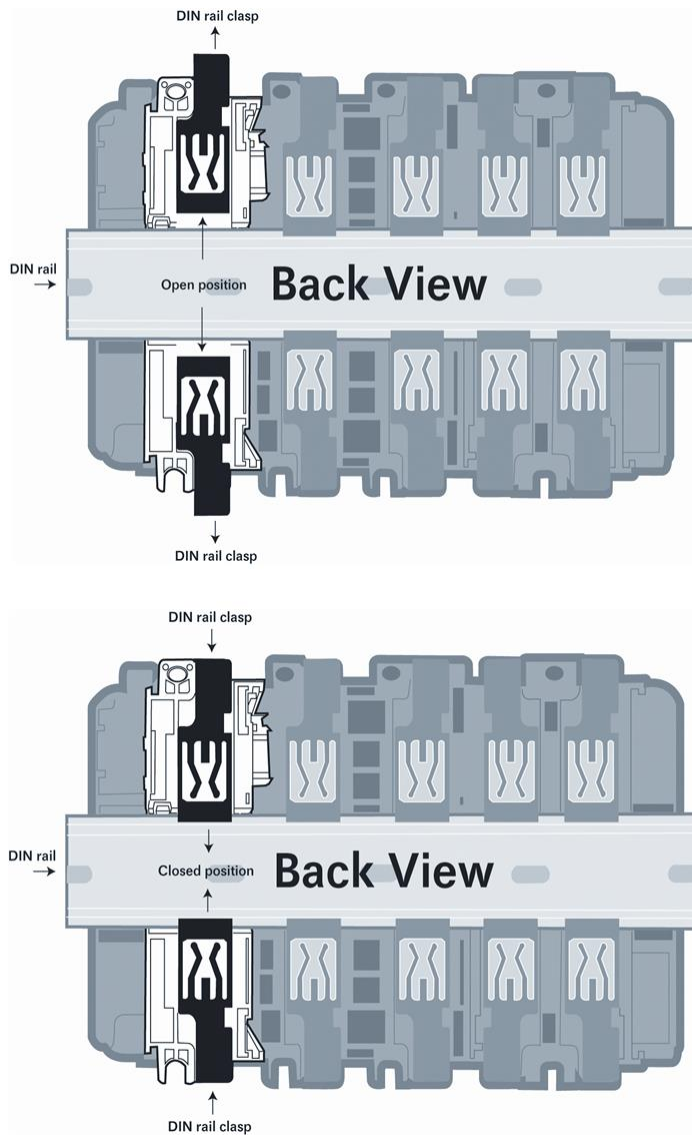


Move the Bus Lever to the left
until it clicks



- 4 Close all DIN-rail latches.

- 5 Press the DIN-rail mounting area of the controller against the DIN-rail. The latches momentarily open and lock into place.

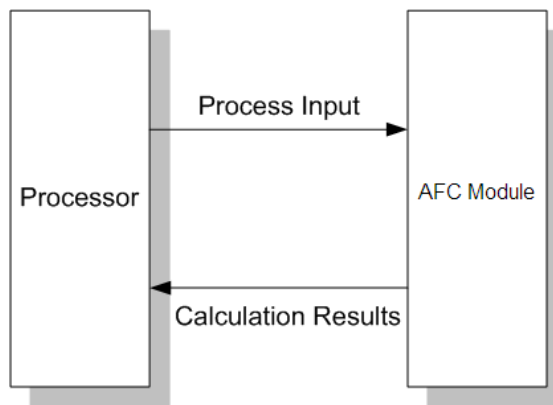


3 Ladder Logic Implementation

In This Chapter

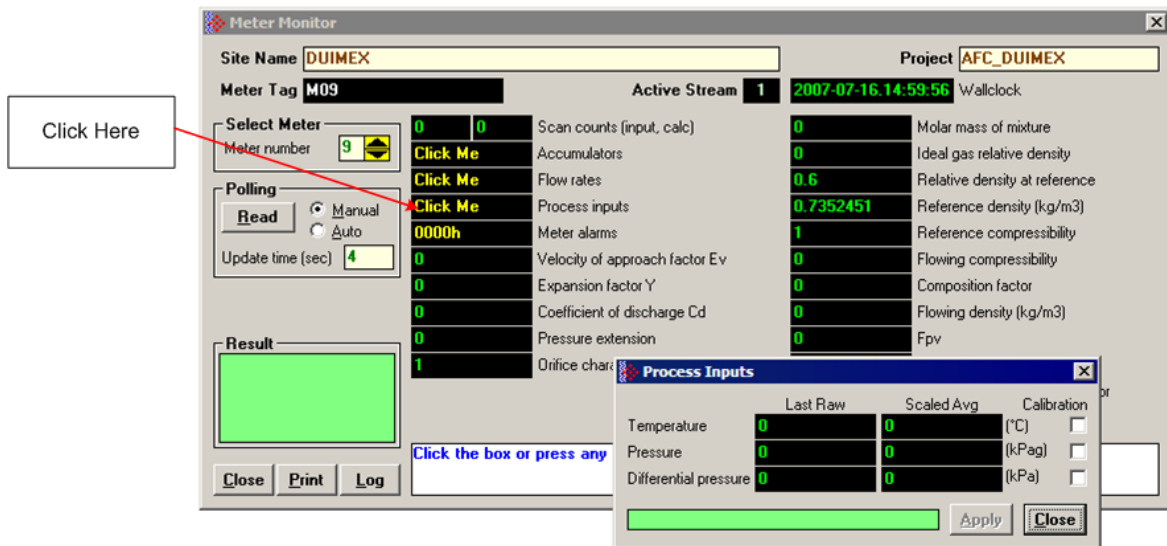
- ❖ Introduction..... 25
- ❖ Overview 25
- ❖ How does the sample ladder work? 28
- ❖ CompactLogix Sample Ladder File 37
- ❖ MicroLogix Sample Ladder File 68

The sample ladder logic performs tasks that are covered in the Ladder Logic sections of this manual. The most important task is to continuously write meter process input variables from the processor to the module, and read calculation results from the module to the processor.

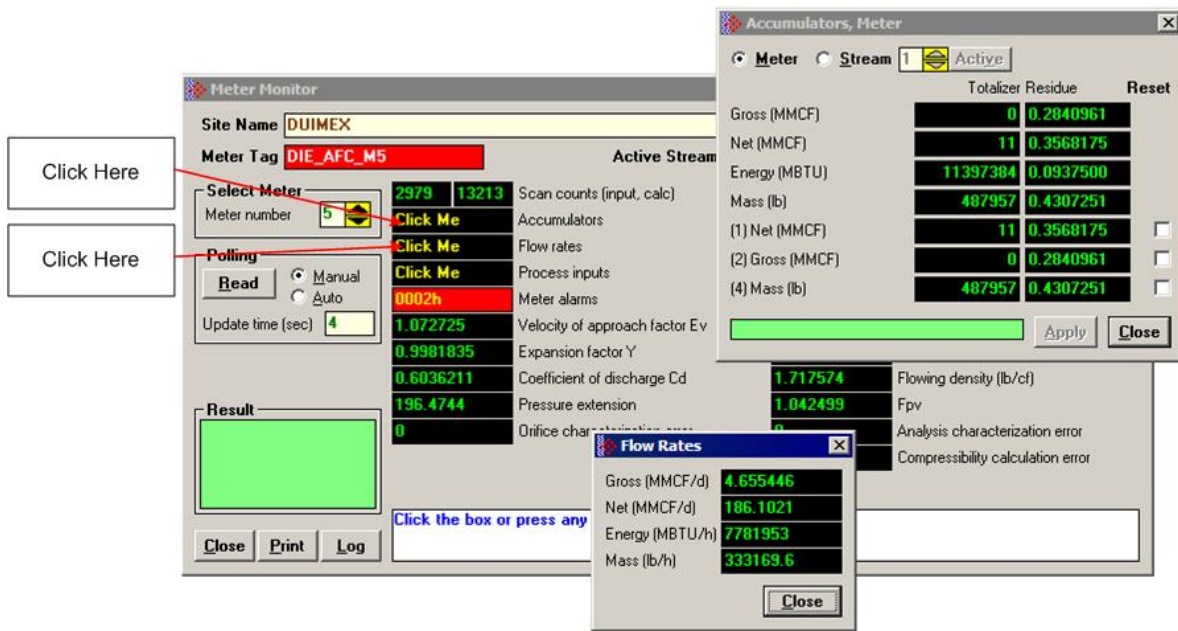


Refer to the Ladder Logic sections for instructions on how to transfer the meter process variables from the processor to the module. Ladder logic is required to move the process variables to the correct data file or controller tag in the processor.

The **Meter Monitor** window (*Process Inputs* field) displays the values that are transferred from the processor.



The values calculated by the module are continuously transferred to the processor. You can refer to the **Meter Monitor** window to verify results calculated by the module.



Refer to the Ladder Logic section for more information regarding the data files and controller tags that store the calculation results transferred from the module (for example, accumulator, flow rate, and so on).

3.1 Introduction

ProSoft strongly recommends using the sample ladder logic for all MVI69-AFC applications. The sample ladder logic configures all backplane features, and most applications will not require significant changes. The sample ladder logic implements the following tasks:

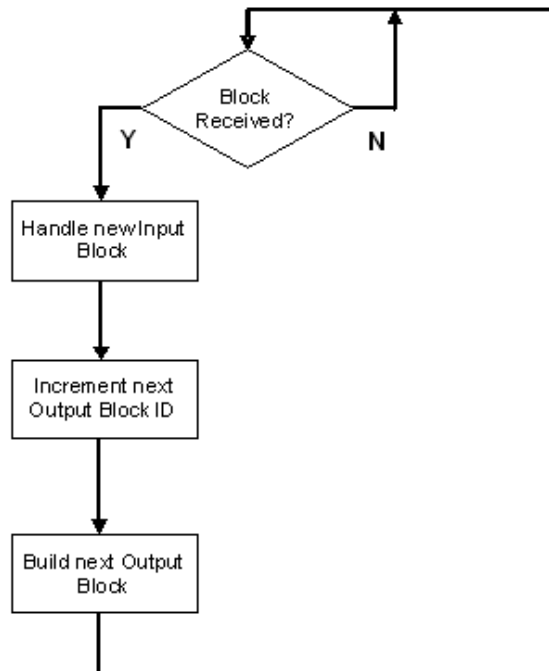
- 1 Write process variables
- 2 Read calculation results
- 3 Set the MVI69-AFC wallclock
- 4 Disable and enable the meters
- 5 Read meter profile (meter type and product group)
- 6 Request meter signals (write archives and reset accumulators)
- 7 Read meter alarms
- 8 Request Modbus master operation
- 9 Modbus Gateway transfer (read and write)
- 10 Read meter archives
- 11 Write molar analysis
- 12 Read site status

There are two sample ladder files, one for each supported platform.

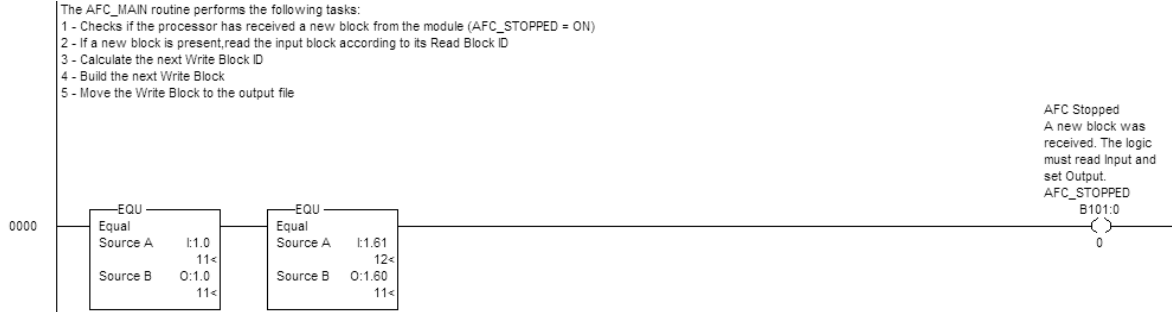
Ladder File	Platform	File Size
MVI69AFC_MICROLOGIX.RSS	MicroLogix	154KB
MVI69AFC_COMPACTLOGIX.ACD	CompactLogix	318KB

3.2 Overview

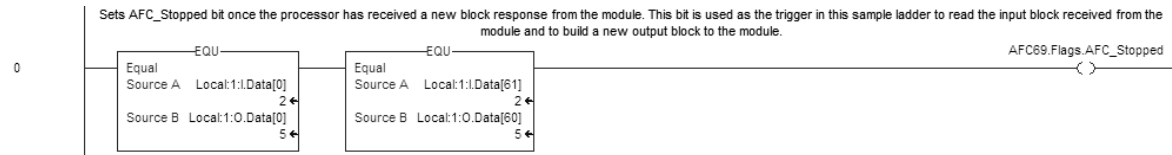
The sample ladder logic is divided into the following basic steps:



Therefore, the sample ladder logic starts by checking if the received sentinel and anchor values (from input block) contain the same value that was sent on the last output block. The following logic is used for the MicroLogix sample ladder to check if a new input block is received:



The following illustration shows the same rung for the CompactLogix processor.

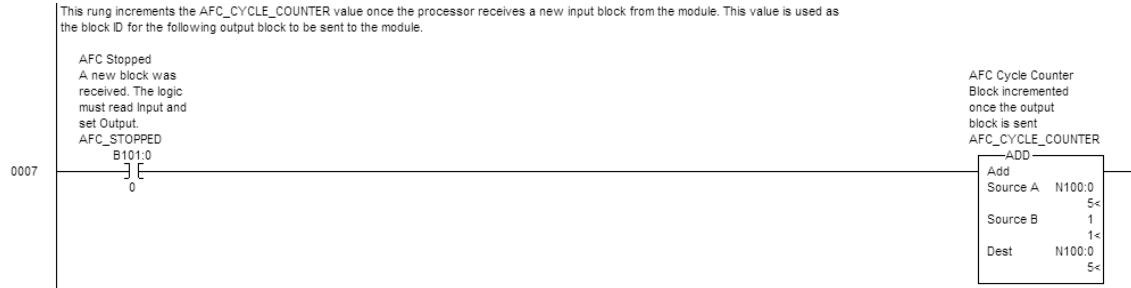


Although there are some minor differences, the overall programming concepts are similar for MicroLogix and CompactLogix.

After the logic detects that a new block was received it will handle the input block according to its block sequence number (block ID). The following rung is used in the MicroLogix sample ladder logic to copy the received block from the input file:

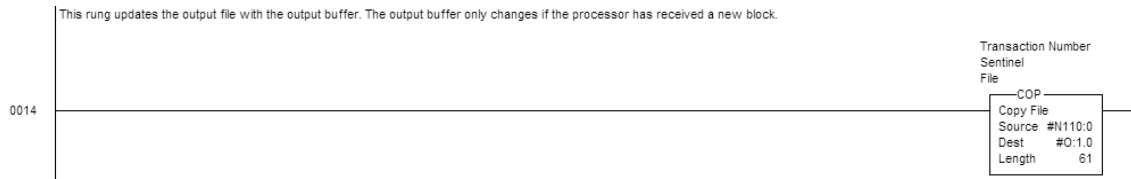


The sample ladder then will handle the received block, based on its Block ID number. Then, the sample ladder logic will calculate the block sequence number for the following output block. The following rung is used to increment the next sequence number (block ID) to be used for the next output to be sent from the processor:



For certain special blocks, the sample ladder logic might use different block IDs (as covered later in this User Manual).

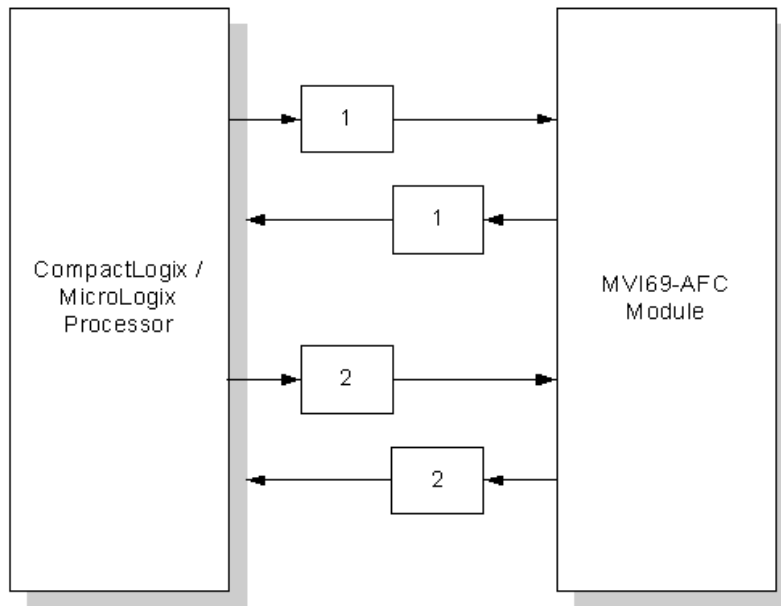
After the sample ladder logic builds the next output block, and copies the new block to the output file:



3.3 How does the sample ladder work?

This section presents a general overview on how the sample ladder logic works. For more information about specific rungs, refer to the ladder logic comments in the ladder file.

The ladder logic consists of basically sending output blocks to the MVI69-AFC and receiving the input blocks from the module. Each block contains a Block Sequence number that identifies the block. The input response block sent by the module will also contain the same Block Sequence Number as the previous output block.



The ladder logic performs the following sequence:

- 1 Receives the input block from the MVI69-AFC
- 2 Copies the input block content from the input buffer to the controller tags based on the Block Sequence Number
- 3 Increments the next Block Sequence Number
- 4 Builds the next output block with the new Sequence Number
- 5 Sends the new output block to the module.

3.3.1 Block IDs (Block Sequence Numbers)

The sample ladder logic uses specific block IDs to request certain tasks from the MVI69-AFC module. This section describes the blocks that are generated by the sample ladder logic and its block IDs. The following topics will provide detailed information on how all the features described in this section are implemented by the sample ladder logic.

The MVI69-AFC module processes blocks in the following manner:

Blocks	Without Molar Analysis	Blocks	With Molar Analysis
1	Process[1..4]	1001	Process[1..4]
2	Process[5..8]	1002	Molar[1..2]
3	Process[1..4]	1003	Process[5..8]
4	Process[5..8]	1004	Molar[3..4]
5	Process[1..4]	1005	Process[1..4]
6	Process[5..8]	1006	Molar[5..6]
7	Process[1..4]	1007	Process[5..8]
8	Process[5..8]	1008	Molar[7..8]
9	Gateway1	1009	Gateway1
10	Gateway2	1010	Gateway2
11	Gateway3	1011	Gateway3

You can update the molar analysis for each gas component from the ladder logic or from the AFC Manager. This selection is performed by a single bit as described later in this User Manual.

Without Molar Analysis

If the sample ladder logic is not configured to update the molar analysis data then the following block IDs are generated by the sample ladder logic:

Block ID	Block Description	Meters
1	Process	1, 2, 3 and 4
2	Process	5, 6, 7, and 8
3	Process	1, 2, 3 and 4
4	Process	5, 6, 7, and 8
5	Process	1, 2, 3 and 4
6	Process	5, 6, 7, and 8
7	Process	1, 2, 3 and 4
8	Process	5, 6, 7, and 8

The sample ladder logic can also be used for Modbus gateway transactions. This option is selected by moving a non-zero number of Modbus gateway blocks to a certain register as described later in this User Manual. The following block IDs are used for such application when molar analysis update is not selected (up to 10 Modbus gateway blocks are supported):

Block ID	Block Description
9	Modbus Gateway 1
10	Modbus Gateway 2
11	Modbus Gateway 3
12	Modbus Gateway 4
13	Modbus Gateway 5
14	Modbus Gateway 6
15	Modbus Gateway 7
16	Modbus Gateway 8
17	Modbus Gateway 9
18	Modbus Gateway 10

With Molar Analysis

If the sample ladder logic updates the molar analysis then the following blocks are automatically generated. Note that in this case the sample ladder logic also sends blocks to update meter process variables:

Block ID	Block Description	Meters
1001	Process	1, 2, 3 and 4
1002	Molar Analysis	1 and 2
1003	Process	5, 6, 7, and 8
1004	Molar Analysis	3 and 4
1005	Process	1, 2, 3 and 4
1006	Molar Analysis	5 and 6
1007	Process	5, 6, 7, and 8
1008	Molar Analysis	7 and 8

The following block IDs are used for such application when molar analysis update is not selected (up to 10 Modbus gateway blocks are supported):

Block ID	Block Description
1009	Modbus Gateway 1
1010	Modbus Gateway 2
1011	Modbus Gateway 3
1012	Modbus Gateway 4
1013	Modbus Gateway 5
1014	Modbus Gateway 6
1015	Modbus Gateway 7
1016	Modbus Gateway 8
1017	Modbus Gateway 9
1018	Modbus Gateway 10

The following blocks are used for special tasks that can be eventually selected during the sample ladder logic scan:

Block ID	Block Description
5000	Request Modbus Master Command
6000	Request Modbus Pass-Thru Transaction
7000	Fetch meter archive
8000	Update module wallclock

The following examples demonstrate how the sample ladder logic automatically generates blocks based on the current selections available:

Example 1:

If the following selections are performed:

Description	Value
Molar Analysis Selection Bit	OFF (no update)
Modbus Gateway Count	0 (no blocks)

The following blocks are continuously generated by the sample ladder logic:

1	2	3	4	5	6	7	8
---	---	---	---	---	---	---	---

Example 2:

If the following selections are performed:

Description	Value
Molar Analysis Selection Bit	OFF (no update)
Modbus Gateway Count	7 (blocks)

The following blocks are continuously generated by the sample ladder logic:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----

Example 3:

If the following selections are performed:

Description	Value
Molar Analysis Selection Bit	ON (update)
Modbus Gateway Count	0 (no blocks)

The following blocks are continuously generated by the sample ladder logic:

1001	1002	1003	1004	1005	1006	1007	1008
------	------	------	------	------	------	------	------

Example 4:

If the following selections are performed:

Description	Value
Molar Analysis Selection Bit	ON (no update)
Modbus Gateway Count	4 (blocks)

The following blocks are continuously generated by the sample ladder logic:

1001	1002	1003	1004	1005	1006	1007	1008	1009	10010	10011	10012
------	------	------	------	------	------	------	------	------	-------	-------	-------

The process and analysis blocks send more than one request in a single output block. These blocks also request the following tasks from the MVI69-AFC module:

- Enable Meter
- Disable Meter
- Meter Signals
- Read meter profile

Process Block

A meter process block performs the following tasks:

- Request Process Variables (for 4 meters)
- Enable

The following illustration shows the basic structure of a process block.

Block ID = X (Sentinel)
Process Variable – Meter A
Process Variable – Meter (A+1)
Process Variable – Meter (A+2)
Process Variable – Meter (A+3)
Enable/Disable Meter B
Signals – Meter B
Read Profile – Meter B
Block ID = X (Anchor)

If X = 1, 3, 5, 7 then A=1, B=X

If X = 2, 4, 6, 8 then A=5, B=X

If X = 1001, 1005, then A=1, B=(X-1000)

If X = 1003, 1007, then A=5, B=(X-1000)

The following table contains the detailed process block structure for MicroLogix processors (Block ID = 1) = 1, 3, 5, 7, 1001, 1005:

Offset	Description	Meter
0	Transaction	
1	Block Length	
2	Meter Process Variables	1
3	Meter Process Variables	1
4	Meter Process Variables	1
5	Meter Process Variables	1
6	Meter Process Variables	1
7	Meter Process Variables	1

Offset	Description	Meter
8	Meter Process Variables	1
9	Meter Process Variables	1
10	Meter Process Variables	1
11	Meter Process Variables	1
12	Meter Process Variables	1
13	Meter Process Variables	1
14	Meter Process Variables	2
15	Meter Process Variables	2
16	Meter Process Variables	2
17	Meter Process Variables	2
18	Meter Process Variables	2
19	Meter Process Variables	2
20	Meter Process Variables	2
21	Meter Process Variables	2
22	Meter Process Variables	2
23	Meter Process Variables	2
24	Meter Process Variables	2
25	Meter Process Variables	2
26	Meter Process Variables	3
27	Meter Process Variables	3
28	Meter Process Variables	3
29	Meter Process Variables	3
30	Meter Process Variables	3
31	Meter Process Variables	3
32	Meter Process Variables	3
33	Meter Process Variables	3
34	Meter Process Variables	3
35	Meter Process Variables	3
36	Meter Process Variables	3
37	Meter Process Variables	3
38	Meter Process Variables	4
39	Meter Process Variables	4
40	Meter Process Variables	4
41	Meter Process Variables	4
42	Meter Process Variables	4
43	Meter Process Variables	4
44	Meter Process Variables	4
45	Meter Process Variables	4
46	Meter Process Variables	4

Offset	Description	Meter
47	Meter Process Variables	4
48	Meter Process Variables	4
49	Meter Process Variables	4
50	RESERVED	
51	RESERVED	
52	Enable/Disable Meters*	1
53	Enable/Disable Meters*	1
54	Reset	1
55	Reset	1
56	Read Meter Profile	1
57	Read Meter Profile	1
58	Read Meter Profile	1
59	NOT USED	
60	Transaction	

Enable: 1,3,5,7

Disable: 2,4,6,8

Molar Analysis Block

A molar analysis block is basically structured as follows:

Block ID = X (Sentinel)
Process Variable – Meter A
Process Variable – Meter(A+1)
Enable/Disable Meter B
Signals – Meter B
Read Profile – Meter B
Block ID = X (Anchor)

If X = 1002 then A=1, B=(X-1000)

If X = 1004 then A=3, B=(X-1000)

If X = 1006 then A=5, B=(X-1000)

If X = 1008 then A=7, B=(X-1000)

The following table contains the detailed block structure (Block ID = 1002) for MicroLogix processors:

Offset	Description	Meter
0	Transaction	
1	Block Length	
2	Molar Analysis	1
3	Molar Analysis	1
4	Molar Analysis	1
5	Molar Analysis	1
6	Molar Analysis	1
7	Molar Analysis	1
8	Molar Analysis	1
9	Molar Analysis	1
10	Molar Analysis	1
11	Molar Analysis	1
12	Molar Analysis	1
13	Molar Analysis	1
14	Molar Analysis	1
15	Molar Analysis	1
16	Molar Analysis	1
17	Molar Analysis	1
18	Molar Analysis	1
19	Molar Analysis	1
20	Molar Analysis	1
21	Molar Analysis	1
22	Molar Analysis	1
23	Molar Analysis	1
24	Molar Analysis	1
25	Molar Analysis	1
26	Molar Analysis	1
27	Molar Analysis	2
28	Molar Analysis	2
29	Molar Analysis	2
30	Molar Analysis	2
31	Molar Analysis	2
32	Molar Analysis	2
33	Molar Analysis	2
34	Molar Analysis	2
35	Molar Analysis	2
36	Molar Analysis	2
37	Molar Analysis	2

Offset	Description	Meter
38	Molar Analysis	2
39	Molar Analysis	2
40	Molar Analysis	2
41	Molar Analysis	2
42	Molar Analysis	2
43	Molar Analysis	2
44	Molar Analysis	2
45	Molar Analysis	2
46	Molar Analysis	2
47	Molar Analysis	2
48	Molar Analysis	2
49	Molar Analysis	2
50	Molar Analysis	2
51	Molar Analysis	2
52	Enable/Disable Meters*	2
53	Enable/Disable Meters*	2
54	Reset	2
55	Reset	2
56	Read Meter Profile	2
57	Read Meter Profile	2
58	Read Meter Profile	2
59	NOT USED	
60	Transaction	

Enable: 1,3,5,7

Disable: 2,4,6,8

3.4 CompactLogix Sample Ladder File

This section shows how you can extract important information from the sample ladder without explaining the details of how the sample ladder actually works. You will realize that for most applications it will be enough to refer only to the controller tags in order to perform the tasks.

3.4.1 Input/Output Blocks for Data Transfer

When you configure the MVI69-AFC module to the I/O configuration of your CompactLogix or MicroLogix controller, the tags of the Output Block Array (OBA) and the Input Block Array (IBA) for the module are automatically created as controller tags by the RSLogix 5000 programming software.

Output Block Array

Size: 61 elements

Data type: Integer (2 bytes)

The OBA is used for transferring process variables from individual meter runs and other data that are available in the processor memory to your AFC module via the backplane. This array is made up of a block of 61 contiguous integer elements.

Element	Attribute
Local 3.O (slot number)	Local.O.Data[0]
	Local.O.Data[1]
	Local.O.Data[2]

	Local.O.Data[n]

	Local.O.Data[60]

The meter run process variables such as temperature, pressure, differential pressure, meter pulses, and so on. may come from other I/O modules in the rack or from other sources like an HMI and so on. These variables must be copied to predefined structured function blocks which transfer them to the AFC module as part of one or more OBAs.

Input Block Array

Size: 62 elements

Data type: Integer (2 bytes)

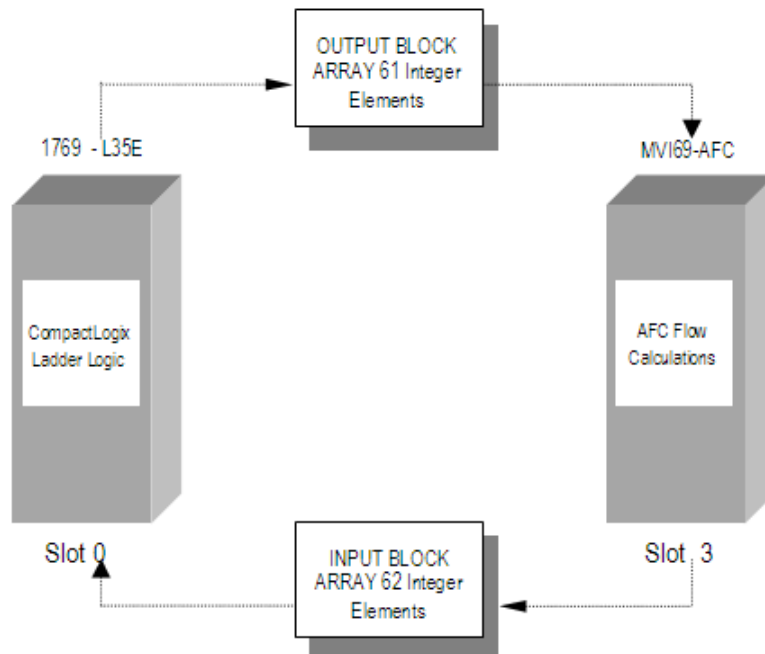
The AFC consumes the process variable inputs received in the OBAs and returns calculated variables such as process flowrate, accumulated product volume, and so on. by populating the IBA and returning it to the processor via the backplane. This array is made up of a block of 62 contiguous integer elements.

Element	Attribute
Local 3.I (slot number)	Local.I.Data[0]
	Local.I.Data[1]
	Local.I.Data[2]

	Local.I.Data[n]

	Local.I.Data[61]

The following section describes in detail how the MVI69-AFC module facilitates the two-way data transfer by conveniently partitioning the Input/Output blocks into smaller function blocks to allow flexibility in designing and implementing the ladder logic to support the AFC module.



3.4.2 *Input/Output Transactions*

The MVI69-AFC module runs automatically on power up. It receives the process variables from the CompactLogix Controller through the associated output block array and returns the calculated results back through the Input Block Array.

The data transfer to and from the MVI69-AFC module is implemented through PLC ladder logic, using the concept of Transactions to establish an interface between the AFC module and the controller. A transaction is a transfer of a pre-assembled block of data words between the controller and the AFC module. The AFC module supports a block multiplexing scheme whereby you may set up a number of Output Block Arrays (OBAs) with corresponding Input Block Arrays (IBAs) for two way data transfer. This is done by giving each OBA a unique non-zero Transaction Identifier number. The AFC module will not process an OBA unless the Transaction Identifier number of the current OBA is different from the previous one.

Output (Transaction) Block Array Definition

The 62 integer elements of the OBA that must be formatted in the controller ladder logic program are defined as follows:

Element #	Attribute
0	Sentinel (Transaction Number)
1	Output Block Array Length (≤ 245)
2	
3	
4	Space for User Allocated Sub Blocks
.	(Total Available Elements = 245)
.	
.	
56	
57	
58	
59	Reserved
60	Anchor (Transaction Number)

The AFC module recognizes an OBA as valid only if all three of the following conditions are true:

- 1 The values of the sentinel and the anchor are the same (this is called the "transaction number").
- 2 The transaction number (sentinel & anchor) is non-zero.
- 3 The transaction number is different from the preceding one.

AFC Response to an OBA

Each time the AFC module receives an OBA, it processes this array and returns a corresponding Input Block Array (IBA) with the same transaction number so that the PLC ladder logic can act upon the AFC's response to the data it received in the OBA. The OBA description follows.

Sentinel & Anchor (Transaction Number)

In order to allow the AFC to tell each Output Block Array apart, a unique Transaction Identifier number is assigned to each transmit block. This Transaction Identifier number is assigned to the top and bottom elements of the transaction block to ensure data transfer integrity. The top transaction identifier is called the *Transaction Number Sentinel* and the bottom transaction identifier is called the *Transaction Number Anchor*.

Output Block Length

The element immediately following the sentinel contains the length of the data portion, not including the sentinel, the anchor, the "reserved" word, or the Output Block Length element itself; hence its maximum value is 57. If the output data block you configured is less than 57 elements, unused elements between the end of the data block and the anchor are ignored on output (to the AFC) and returned as zero on input (from the AFC).

Output Function Blocks (OFB)

The architecture of the Output Block Array provides you with a flexible and powerful way to interface the process and other data to the AFC module. A number of pre-defined Output Function Blocks are at your disposal to implement as required. Each Output Function Block type carries out a specific function.

For example, the Meter Process Variable Output Function Block transfers the process variables (temperature, pressure, meter pulses, and so on) to a specific target meter channel in the AFC module. The calculated flow rates, product accumulators, and so on are returned to the controller by the AFC in the associated Input Function Block (IFB) as part of the Input Block Array (IBA).

In the same way a Wall Clock Function Block allows you to synchronize the AFC Wall Clock to the PLC Wall Clock. Like Lego blocks of various sizes, these function blocks can be stacked together in the data portion of the Output Block Array as shown in the following diagram:

	Element #	Attribute
	0	Sentinel (Transaction Number)
	1	Output Block Length (OBL<=57)
	2	
Global Wall Clock Function Block: 7 Elements	3	
Meter 2 Process Variable Function Block: 12 Elements	.	Space for User Allocated Function Blocks
Meter 3 Process Variable Function Block: 12 Elements	.	(Total Available Elements = 57)
Meter 3 Analysis Function Block: 25 Elements	57	
	.	
	.	
	56	
	57	
	58	
	59	Reserved
	60	Anchor (Transaction Number)

Input (Transaction) Block Array Definition

The Input Block Array is where the AFC returns the (processed and calculated data) responses to all function blocks received in the Output Block Array from the controller. It returns Input Function Blocks in locations and sizes matching those of the OFBs in the OBA. The AFC completes the actions implied by all function blocks before responding with the IBA to the controller. If the AFC is unable to determine the size of any function block, or the implied size overlaps the block array's anchor, then that and all following function blocks are not processed and a format alarm is raised.

The 62 integer elements of the Input (Transaction) Block Array (IBA) that are returned to the controller to the AFC module are defined as follows:

Element	Attribute																										
0	Sentinel Echo (Transaction Number)																										
1	Block length echo (IBL) (Negative if formatting error present)																										
2																											
3																											
4	Space for Input Image of User Allocated Function Blocks																										
.	(Total Available Elements = 57)																										
.																											
.																											
.																											
56																											
57																											
58																											
59	Site Status																										
	<table border="1"> <thead> <tr> <th>Bit #</th> <th>Site Status</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>MVI69-AFC Released and Active</td> </tr> <tr> <td>1</td> <td>Checksum Alarm</td> </tr> <tr> <td>2</td> <td>Reserved</td> </tr> <tr> <td>3</td> <td>Reserved</td> </tr> <tr> <td>4</td> <td>PLC Halted, Offline or Missing</td> </tr> <tr> <td>5</td> <td>Measurement Configuration Changed</td> </tr> <tr> <td>6</td> <td>Set upon Power-Up & cleared when the Wall Clock is set for the first time</td> </tr> <tr> <td>7</td> <td>Cold Start: remains set until first enabled meter is detected</td> </tr> <tr> <td>8</td> <td>Modbus Master poll pending</td> </tr> <tr> <td>9</td> <td>Modbus Master poll complete, results waiting</td> </tr> <tr> <td>10</td> <td>Pass-Thru input pending</td> </tr> <tr> <td>11-15</td> <td>Reserved</td> </tr> </tbody> </table>	Bit #	Site Status	0	MVI69-AFC Released and Active	1	Checksum Alarm	2	Reserved	3	Reserved	4	PLC Halted, Offline or Missing	5	Measurement Configuration Changed	6	Set upon Power-Up & cleared when the Wall Clock is set for the first time	7	Cold Start: remains set until first enabled meter is detected	8	Modbus Master poll pending	9	Modbus Master poll complete, results waiting	10	Pass-Thru input pending	11-15	Reserved
Bit #	Site Status																										
0	MVI69-AFC Released and Active																										
1	Checksum Alarm																										
2	Reserved																										
3	Reserved																										
4	PLC Halted, Offline or Missing																										
5	Measurement Configuration Changed																										
6	Set upon Power-Up & cleared when the Wall Clock is set for the first time																										
7	Cold Start: remains set until first enabled meter is detected																										
8	Modbus Master poll pending																										
9	Modbus Master poll complete, results waiting																										
10	Pass-Thru input pending																										
11-15	Reserved																										
60	Meter Alarm Map: Bit <n> set=Meter <n+1> in Alarm																										
61	Anchor Echo (Transaction Number)																										

The input block returned by the AFC module has the sentinel echo and the anchor echo in fixed locations.

Note: The anchor echo is returned as the 62nd element in the IBA and not as the 61st element as sent in the OBA.

The controller may then verify the validity of the returned data by checking that the received sentinel echo is the same as the anchor echo. If more than one OBA is being processed, the transaction number must be used to match the IBA (AFC output) with the originating OBA (AFC input, controller output).

Input Block Length and Format Alarm

The "length" element returned by the module in the IBA has the high-order (sign) bit set if any formatting error is detected in the data portion of the OBA you formatted in the ladder logic. This is referred to as a "format alarm". The remainder of the input length element contains the number of data words successfully processed before any formatting inconsistencies were detected by the AFC module and the alarm condition annunciated. If no formatting error exists, the length element will echo the output length you entered. If the output length element itself raises an alarm (for example, value > 57), then the returned length value is -1.

3.4.3 Wallclock

After the module powers up, it will not perform flow calculation until it receives valid wallclock information from the CompactLogix or MicroLogix processor. The sample ladder logic uses the processor internal clock as the source of the wallclock information.

Configure the processor time and date information:

- 1 Right-Click on Controller MVI69-AFC folder
- 2 Click on Properties
- 3 Select the Date/Time tab
- 4 Enter a valid date and time information.

After the *AFC69.Flags.AFC_Set_Clock* bit is toggled, the logic will move the date and time information from the processor to the MVI69-AFC module.

[-] AFC69.Flags	{...}
[-] AFC69.Flags.AFC_Stopped	0
▶ [-] AFC69.Flags.AFC_Set_Clock	0
[-] AFC69.Flags.AFC_OutputError	0
[+] AFC69.Flags.AFC_InputSeque...	3
[-] AFC69.Flags.plc_set_clock	0
[+] AFC69.Flags.Current_meter	0

The *AFC69.Flags.AFC_Set_Clock* bit is latched in the power up routine, in order to guarantee that the module will be up and running after power up.



After the ladder logic receives the input block back from the module it unlatches the *AFC69.Flags.AFC_Set_Clock* bit.

You may want to periodically synchronize the processor and the module's wallclock, especially when the date and time information is received from a remote station. In this case, you must write ladder logic to periodically toggle the *AFC69.Flags.AFC_Set_Clock* bit.

3.4.4 Site Status

The ladder logic continuously reads the site status from the MVI69-AFC. The following controller tags are used:

Information	Controller Tag
AFC69(8) Released	AFC69.Site_Status.AFC_ACTIVE
Checksum Alarms	AFC69.Site_Status.Checksum_Alarms
Processor Halted	AFC69.Site_Status.PLC_Halted
Cfg Changed	AFC69.Site_Status.Cfg_changed
Power Up	AFC69.Site_Status.Powerup
Cold Start	AFC69.Site_Status.ColdStart

AFC69.Site_Status		{...}
-AFC69.Site_Status.AFC_ACTIVE		1
-AFC69.Site_Status.Checksum_Alarms		0
-AFC69.Site_Status.reserved2		0
-AFC69.Site_Status.reserved3		0
-AFC69.Site_Status.PLC_HALTED		0
-AFC69.Site_Status.Cfg_changed		0
-AFC69.Site_Status.Powerup		0
-AFC69.Site_Status.ColdStart		0
-AFC69.Site_Status.ModbusMasterPollPending		0
-AFC69.Site_Status.ModbusMasterPollComplete		0
-AFC69.Site_Status.ModbusPassThruPending		0
-AFC69.Site_Status.reserved11		0
-AFC69.Site_Status.reserved12		0
-AFC69.Site_Status.reserved13		0
-AFC69.Site_Status.reserved14		0
-AFC69.Site_Status.reserved15		0

Enable/Disable Status

Each meter run will only perform flow calculation while it is enabled. However, you cannot change a meter type, product group or units while the meter is enabled. In order to accomplish this, you have to disable the meter, change meter type, product or units and then enable the meter again. The meters can be enabled or disabled from ladder logic or AFC Manager.

The ladder logic constantly reads each meter enable/disable status from the MVI69-AFC.

Refer to *AFC69.EnableStatus* data structure for each meter status. Each variable should be interpreted as:

AFC69.EnableStatus.Meterx = 0 => Meter x is Disabled

AFC69.EnableStatus.Meterx = 1 => Meter x is Enabled

Tag Name	Value
- AFC69	{ ... }
+ AFC69.Meters	{ ... }
+ AFC69.Site_Status	{ ... }
+ AFC69.Site_Alarms	{ ... }
+ AFC69.Analysis	{ ... }
+ AFC69.Modbus	{ ... }
+ AFC69.ArchiveFetch	{ ... }
+ AFC69.EnableMeter	{ ... }
- AFC69.EnableStatus	{ ... }
- AFC69.EnableStatus.Meter1	1
- AFC69.EnableStatus.Meter2	1
- AFC69.EnableStatus.Meter3	0
- AFC69.EnableStatus.Meter4	0
- AFC69.EnableStatus.Meter5	0
- AFC69.EnableStatus.Meter6	0
- AFC69.EnableStatus.Meter7	0
- AFC69.EnableStatus.Meter8	0

In the previous example, Meter 1 and Meter 2 are enabled. All other meters are disabled.

3.4.5 Meter Profile

The ladder logic constantly reads each meter profile. The meter profile informs each meter type (pulse or orifice), each product group (gas or liquid) and the active stream (for firmware version 2.05.000 or later). This information is not used by most users, but it is very important to the ladder logic in order to perform other important tasks (for example, write process variables and read calculation results).

The *AFC69.Meters[x].Profile* controller tag stores the profile information for each meter run:

AFC69.Meters[0].Profile - Meter 1 Profile

AFC69.Meters[1].Profile - Meter 2 Profile

...

AFC69.Meters[7].Profile - Meter 8 Profile

The controller tags are interpreted as follows:

Controller Tag	Value	Description
<i>AFC69.Meters[x].Profile.MeterType</i>	0	Meter x is an Orifice Meter
<i>AFC69.Meters[x].Profile.MeterType</i>	1	Meter x is a Pulse Meter
<i>AFC69.Meters[x].Profile.ProductGroup</i>	0	Meter x uses a Gas product
<i>AFC69.Meters[x].Profile.ProductGroup</i>	1	Meter x uses a Liquid product

The example below shows a situation where Meter 0 is configured as a pulse (linear) meter and uses a gas product.

- AFC69.Meters[0].Profile	{ ... }
- AFC69.Meters[0].Profile.MeterType	1
- AFC69.Meters[0].Profile.ProductGroup	0

3.4.6 Enable Meter

Each meter can be enabled through ladder logic. Refer to the *AFC69.EnableMeter* data structure. Toggle each *AFC69.EnableMeter.Meterx* controller tag in order to command the meter to be enabled.

The ladder logic will continuously analyze each meter enable status. After the logic detects that a specific meter is enabled (*AFC69.EnableStatus.Meterx = 1*) the command bit (*AFC69.EnableMeter.Meterx*) will be unlatched.

Tag Name	Value
AFC69	{...}
AFC69.Meters	{...}
AFC69.Site_Status	{...}
AFC69.Site_Alarms	{...}
AFC69.Analysis	{...}
AFC69.Modbus	{...}
AFC69.ArchiveFetch	{...}
AFC69.EnableMeter	{...}
AFC69.EnableMeter.Meter1	0
AFC69.EnableMeter.Meter2	0
AFC69.EnableMeter.Meter3	0
AFC69.EnableMeter.Meter4	0
AFC69.EnableMeter.Meter5	0
AFC69.EnableMeter.Meter6	0
AFC69.EnableMeter.Meter7	0
AFC69.EnableMeter.Meter8	0

Note: DO NOT create a rung in ladder logic to constantly enable the meter. The command bit should be toggled only once in order to enable the meter.

3.4.7 Disable Meter

Each meter can be disabled through ladder logic. Refer to the *AFC69.DisableMeter* data structure. Toggle each *AFC69.DisableMeter.Meterx* controller tag in order to command the meter to be disabled.

The ladder logic will continuously analyze each meter enable status. After the logic realizes that a specific meter is disabled (*AFC69.DisableStatus.Meterx = 1*) the command bit (*AFC69.DisableMeter.Meterx*) will be unlatched.

Tag Name	Δ	Value
[-] AFC69		{...}
[+] AFC69.Meters		{...}
[+] AFC69.Site_Status		{...}
[+] AFC69.Site_Alarms		{...}
[+] AFC69.Analysis		{...}
[+] AFC69.Modbus		{...}
[+] AFC69.ArchiveFetch		{...}
[+] AFC69.EnableMeter		{...}
[-] AFC69.EnableStatus		{...}
[-] AFC69.DisableMeter		{...}
-AFC69.DisableMeter.Meter1		0
-AFC69.DisableMeter.Meter2		0
-AFC69.DisableMeter.Meter3		0
-AFC69.DisableMeter.Meter4		0
-AFC69.DisableMeter.Meter5		0
-AFC69.DisableMeter.Meter6		0
-AFC69.DisableMeter.Meter7		0
-AFC69.DisableMeter.Meter8		0

Note: DO NOT create a rung in ladder logic to constantly disable the meter. The command bit should be toggled only once in order to disable the meter.

3.4.8 Meter Process Variables

In order to perform flow calculation the module must receive the meter process variables from the processor.

The process variables will depend on the meter type and product group. So the profile information (discussed before) decides which controller tag the variables will be copied from. The following options are used:

Meter Type	Product Group	Use this Controller Tag
Differential	Gas	<i>AFC69.Meters[x].Variables.DifferentialGas</i>
Differential	Liquid	<i>AFC69.Meters[x].Variables.DifferentialLiquid</i>
Linear	Gas	<i>AFC69.Meters[x].Variables.LinearGas</i>
Linear	Liquid	<i>AFC69.Meters[x].Variables.LinearLiquid</i>

The following illustration shows an example controller tag from RSLogix5000:

[-] AFC69.Meters[0].Variables		{...}
[+] AFC69.Meters[0].Variables.BID	2#0010_0000_0000_0001	
[+] AFC69.Meters[0].Variables.DifferentialGas		{...}
[+] AFC69.Meters[0].Variables.DifferentialLiquid		{...}
[+] AFC69.Meters[0].Variables.LinearGas		{...}
[+] AFC69.Meters[0].Variables.LinearLiquid		{...}

All you have to do is identify where the variables will be copied from (based on the meter type and product group). The sample ladder logic will automatically select the correct controller tags.

Note: In order to configure each meter type and product group, refer to the AFC Manager software tool.

For each possible combination, the following variables are used:

1) Meter Type = Differential & Product Group = Gas

Process Input	Controller Tag	Data Type
Temperature	AFC69.Meters[x].Variables.DifferentialGas.Temperature	REAL
Pressure	AFC69.Meters[x].Variables.DifferentialGas.Pressure	REAL
Differential Pressure	AFC69.Meters[x].Variables.DifferentialGas.Differential_Pressure	REAL

Where x can assume values between 0 and 7.

The following illustration shows an example for Meter 1:

- AFC69.Meters[0].Variables.DifferentialGas	{...}		Meter_Orifice_Gas_Output
+ AFC69.Meters[0].Variables.DifferentialGas.Reserved1	0	Decimal	INT
- AFC69.Meters[0].Variables.DifferentialGas.Temperature	55.97	Float	REAL
- AFC69.Meters[0].Variables.DifferentialGas.Pressure	83.75	Float	REAL
- AFC69.Meters[0].Variables.DifferentialGas.Differential_Pressure	99.45	Float	REAL
+ AFC69.Meters[0].Variables.DifferentialGas.Reserved2	0	Decimal	INT
+ AFC69.Meters[0].Variables.DifferentialGas.Reserved3	0	Decimal	INT
+ AFC69.Meters[0].Variables.DifferentialGas.Reserved4	0	Decimal	INT
+ AFC69.Meters[0].Variables.DifferentialGas.Reserved5	0	Decimal	INT

2) Meter Type = Differential & Product Group = Liquid

Process Input	Controller Tag	Data Type
Water %	AFC69.Meters[x].Variables.DifferentialLiquid.Water_Percent	INT
Temperature	AFC69.Meters[x].Variables.DifferentialLiquid.Temperature	REAL
Pressure	AFC69.Meters[x].Variables.DifferentialLiquid.Pressure	REAL
Diff Pressure	AFC69.Meters[x].Variables.DifferentialLiquid.Differential_Pressure	REAL
Density	AFC69.Meters[x].Variables.DifferentialLiquid.Density	REAL

Where x can assume values between 0 and 7.

- AFC69.Meters[0].Variables.DifferentialLiquid	{...}		Meter_Orifice_Liquid_Output
+ AFC69.Meters[0].Variables.DifferentialLiquid.Water_Percent	145	Decimal	INT
- AFC69.Meters[0].Variables.DifferentialLiquid.Temperature	150.0	Float	REAL
- AFC69.Meters[0].Variables.DifferentialLiquid.Pressure	180.0	Float	REAL
- AFC69.Meters[0].Variables.DifferentialLiquid.Differential_Pressure	24.5	Float	REAL
- AFC69.Meters[0].Variables.DifferentialLiquid.Density	990.0	Float	REAL
+ AFC69.Meters[0].Variables.DifferentialLiquid.Reserved1	0	Decimal	INT
+ AFC69.Meters[0].Variables.DifferentialLiquid.Reserved2	0	Decimal	INT

3) Meter Type = Linear & Product Group = Gas

Process Input	Controller Tag	Data Type
Temperature	AFC69.Meters[x].Variables.LinearGas.Temperature	REAL
Pressure	AFC69.Meters[x].Variables.LinearGas.Pressure	REAL
Pulse Count	AFC69.Meters[x].Variables.LinearGas.Meter_Pulses	DINT
Pulse Frequency	AFC69.Meters[x].Variables.LinearGas.Pulse_Frequency	REAL

Where x can assume values between 0 and 7.

The following illustration shows an example for Meter 1:

[-] AFC69.Meters[0].Variables.LinearGas	{...}		Meter_Pulse_Gas_Output
[-] AFC69.Meters[0].Variables.LinearGas.Reserved1	0	Decimal	INT
[-] AFC69.Meters[0].Variables.LinearGas.Temperature	12.0	Float	REAL
[-] AFC69.Meters[0].Variables.LinearGas.Pressure	43.0	Float	REAL
[+] AFC69.Meters[0].Variables.LinearGas.Meter_Pulses	333	Decimal	DINT
[+] AFC69.Meters[0].Variables.LinearGas.Reserved2	0	Decimal	INT
[+] AFC69.Meters[0].Variables.LinearGas.Reserved3	0	Decimal	INT
[-] AFC69.Meters[0].Variables.LinearGas.Pulse_Frequency	122.0	Float	REAL

4) Meter Type = Linear & Product Group = Liquid

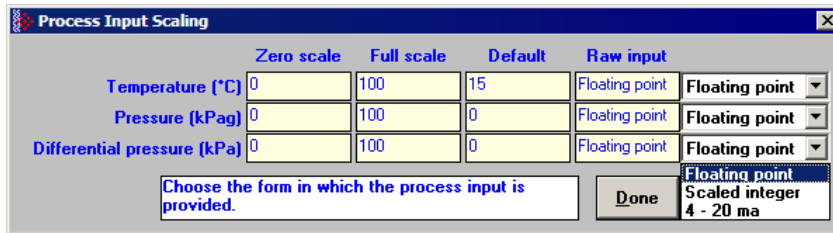
Process Input	Controller Tag	Data Type
Water Percent	AFC69.Meters[x].Variables.LinearLiquid.Water_Percent	INT
Temperature	AFC69.Meters[x].Variables.LinearLiquid.Temperature	REAL
Pressure	AFC69.Meters[x].Variables.LinearLiquid.Pressure	REAL
Pulse Count	AFC69.Meters[x].Variables.LinearLiquid.Meter_Pulses	DINT
Density	AFC69.Meters[x].Variables.LinearLiquid.Density	REAL
Pulse Frequency	AFC69.Meters[x].Variables.LinearLiquid.Pulse_Frequency	REAL

Where x can assume values between 0 and 7.

The following illustration shows an example for Meter 1:

[-] AFC69.Meters[0].Variables.LinearLiquid	{...}		Meter_Pulse_Liquid_Output
[+] AFC69.Meters[0].Variables.LinearLiquid.Water_Percent	0	Decimal	INT
[-] AFC69.Meters[0].Variables.LinearLiquid.Temperature	23.0	Float	REAL
[-] AFC69.Meters[0].Variables.LinearLiquid.Pressure	11.0	Float	REAL
[+] AFC69.Meters[0].Variables.LinearLiquid.Meter_Pulses	532	Decimal	DINT
[-] AFC69.Meters[0].Variables.LinearLiquid.Density	647.0	Float	REAL
[-] AFC69.Meters[0].Variables.LinearLiquid.Pulse_Frequency	25.0	Float	REAL

Important: The sample ladder logic is configured assuming the input variables use floating point format (default from AFC Manager). However, if Scaled Integer or 4 to 20 mA formats are used, change the meter variables format from floating point (REAL) to 32-bit long integer (DINT) in the ladder logic.



The *AFC69.Meters[0].Variables.BID* is the function block ID for each meter. This value is automatically calculated by the ladder logic, so you do not have to move any value to this controller tag.

3.4.9 Molar Analysis (For Gas Product Only)

If the application requires a chromatograph device to transfer molar concentration data to the module, the sample ladder may dynamically supply all molar concentrations to the MVI69-AFC.

Examine the "Selected" check boxes for all elements using the AFC Manager (clicking on the Analysis button in the Meter Configuration window).

Set the *AFC69.Meters[x].Analysis.Enable* bit. After that, any molar concentration configuration performed through AFC Manager will be overwritten by the ladder logic.

For modules firmware version 2.05.000 or later, select the stream number (1 to 4) to be associated to the analysis data. Select 0 to update the current active stream. If a non-active stream is selected, the molar data will only be available through AFC Manager once that stream is selected as the new active one.

[-] AFC69.Meters[0].Analysis	{ ... }		AFC_Meter_Analysis
[-] AFC69.Meters[0].Analysis.Enable	0	Decimal	BOOL
[+] AFC69.Meters[0].Analysis.Stream	0	Decimal	INT
[+] AFC69.Meters[0].Analysis.Components	{ ... }		AFC_Analysis_Components

Refer to the Meters[x].Analysis controller tag in order to move the concentrations for each meter (x assumes values between 0 and 7).

AFC69.Meters[0].Analysis.Components	{...}		AFC_Analysis_Components
+ AFC69.Meters[0].Analysis.Components.C1	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.N2	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.CO2	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.C2	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.C3	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.H2O	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.H2S	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.H2	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.CO	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.O2	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.IC4	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.NC4	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.IC5	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.NC5	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.C6	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.C7	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.C8	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.C9	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.C10	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.He	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.Ar	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.NeoC5	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.Ux_U...	0	Decimal	INT
+ AFC69.Meters[0].Analysis.Components.Uy_U...	0	Decimal	INT

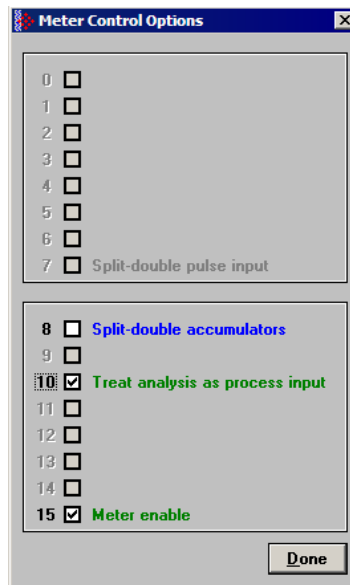
The concentrations are entered as scaled integer format where 10000 = 100%.

For example:

C1 = 9168 => all concentration should be 100%. Sometimes the chromatograph can generate values which total is slightly less (or more) than 100%. In this case you should configure the Normalization Error Tolerance parameter in the AFC Manager in order to make sure that the module will not generate any alarms.

When the module detects that a molar concentration value has changed it will generate an event. However, when the values are updated from ladder logic using a chromatograph device it is not convenient to generate an alarm every time a concentration value changes. In this case you may configure the module to not generate any alarms when a molar value is modified.

So, you should select (check) "**Meter Configuration / Ctrl Options / Treat Analysis as Process Input**" check box:



3.4.10 Select Stream (Firmware version 2.05.000 or later)

The processor can select one of the four streams as the new active stream performing flow calculation.

Command	Use this Controller Tag	Description
Enable	<i>AFC69.MeterSignals.Enable</i>	This value should be set as 1 in order to enable the command. If this value is 0 the module will ignore the commands from the ladder
Select Meter	<i>AFC69. MeterSignals.Meter</i>	Select the meter number (1 to 8) for the archive command
SelectStreamX	<i>AFC69. MeterSignals.Signals. SelectStreamX</i>	Select StreamX

[-] AFC69.MeterSignals	{...}
AFC69.MeterSignals.Enable	1
[+] AFC69.MeterSignals.BID	12289
[+] AFC69.MeterSignals.Meter	1
[+] AFC69.MeterSignals.Action	0
[-] AFC69.MeterSignals.Signals	{...}
AFC69.MeterSignals.Signals.Select_Stream1	0
AFC69.MeterSignals.Signals.Select_Stream2	0
AFC69.MeterSignals.Signals.Select_Stream3	0
AFC69.MeterSignals.Signals.Select_Stream4	0
AFC69.MeterSignals.Signals.Res_Acc1	0
AFC69.MeterSignals.Signals.Res_Acc2	0
AFC69.MeterSignals.Signals.Res_Acc3	0
AFC69.MeterSignals.Signals.Res_Acc4	0

3.4.11 Meter Calculation Results

After the module has performed the AGA/API calculation, all results are moved to the processor. The ladder logic will move the calculation results to the controller tags described in this section.

The calculation results will depend on the meter type and product group. So the profile information (discussed before) determines which controller tag the results will be copied to. The following options are used:

Meter Type	Product Group	Use this Controller Tag
Differential	Gas	<i>AFC69.Meters[x].Results.DifferentialGas</i>
Differential	Liquid	<i>AFC69.Meters[x].Results.DifferentialLiquid</i>
Linear	Gas	<i>AFC69.Meters[x].Results.LinearGas</i>
Linear	Liquid	<i>AFC69.Meters[x].Results.LinearLiquid</i>

The following shows a controller tag illustration from RSLogix5000:

[-] AFC69.Meters[0].Results	{...}
[+] AFC69.Meters[0].Results.BID	2#0010_0010_0000_0001 Binary
[+] AFC69.Meters[0].Results.Alarms	2#1000_0000_0000_0011 Binary
[+] AFC69.Meters[0].Results.DifferentialGas	{...}
[+] AFC69.Meters[0].Results.DifferentialLiquid	{...}
[+] AFC69.Meters[0].Results.LinearGas	{...}
[+] AFC69.Meters[0].Results.LinearLiquid	{...}

All you have to do is to identify where to copy the results (based on the meter type and product group). The ladder logic will automatically select the correct controller tags.

Note: In order to configure each meter type and product group, refer to the AFC manager software tool.

The following variables are used for each possible combination:

1) Meter Type = Differential & Product Group = Gas

Calculation Result	Controller Tag	Data Type
Non-Resettable Net Accumulator	AFC69.Meters[x].Results.DifferentialGas.Net_Accumulator	DINT
Net Flow Rate	AFC69.Meters[x].Results.DifferentialGas.Net_Flowrate	REAL
Gross Flow Rate	AFC69.Meters[x].Results.DifferentialGas.Gross_Flowrate	REAL
Fpv	AFC69.Meters[x].Results.DifferentialGas.Fpv	REAL
Cprime	AFC69.Meters[x].Results.DifferentialGas.Cprime	REAL

Where x can assume values between 0 and 7.

The following is an example for Meter 1:

- AFC69.Meters[0].Results.DifferentialGas	{ . . . }		Meter_Orifice_Gas_Input
+ AFC69.Meters[0].Results.DifferentialGas.Net_Accumulator	212	Decimal	DINT
- AFC69.Meters[0].Results.DifferentialGas.Net_Flowrate	22.0	Float	REAL
- AFC69.Meters[0].Results.DifferentialGas.Gross_Flowrate	21.0	Float	REAL
- AFC69.Meters[0].Results.DifferentialGas.Fpv	1.0	Float	REAL
- AFC69.Meters[0].Results.DifferentialGas.Cprime	1.0	Float	REAL

2) Meter Type = Differential & Product Group = Liquid

Calculation Result	Controller Tag	Data Type
Non-Resettable Net Accumulator	AFC69.Meters[x].Results.DifferentialLiquid.Net_Accumulator	DINT
Net Flow Rate	AFC69.Meters[x].Results.DifferentialLiquid.Net_Flowrate	REAL
Gross Accumulator	AFC69.Meters[x].Results.DifferentialLiquid.Gross_Accumulator	DINT
Gross Standard Accumulator	AFC69.Meters[x].Results.DifferentialLiquid.Standard_Accumulator	DINT
Mass Accumulator	AFC69.Meters[x].Results.DifferentialLiquid.Mass_Accumulator	DINT

Where x can assume values between 0 and 7.

The following shows an example for Meter 1:

- AFC69.Meters[0].Results.DifferentialLiquid	{ . . . }		Meter_Orifice_Liquid_Input
+ AFC69.Meters[0].Results.DifferentialLiquid.Net_Accumul...	168	Decimal	DINT
- AFC69.Meters[0].Results.DifferentialLiquid.Net_Flowrate	4.4670224	Float	REAL
+ AFC69.Meters[0].Results.DifferentialLiquid.Gross_Accum...	176	Decimal	DINT
+ AFC69.Meters[0].Results.DifferentialLiquid.Gross_Stand...	174	Decimal	DINT
+ AFC69.Meters[0].Results.DifferentialLiquid.Mass_Accum...	172527	Decimal	DINT

3) Meter Type = Linear & Product Group = Gas

Calculation Result	Controller Tag	Data Type
Non-Resettable Net Accumulator	AFC69.Meters[x].Results.LinearGas.Net_Accumulator	DINT
Net Flow Rate	AFC69.Meters[x].Results.LinearGas.Net_Flowrate	REAL
Gross Flow Rate	AFC69.Meters[x].Results.LinearGas.Gross_Flowrate	REAL
Fpv	AFC69.Meters[x].Results.LinearGas.Fpv	REAL
Cprime	AFC69.Meters[x].Results.LinearGas.Cprime	REAL

Where x can assume values between 0 and 7.

The following shows an example for Meter 1:

[-] AFC69.Meters[0].Results.LinearGas	{ ... }		Meter_Pulse_Gas_Input
+ AFC69.Meters[0].Results.LinearGas.Net_Accumulator	2925457	Decimal	DINT
- AFC69.Meters[0].Results.LinearGas.Net_Flowrate	0.0	Float	REAL
- AFC69.Meters[0].Results.LinearGas.Gross_Flowrate	0.0	Float	REAL
- AFC69.Meters[0].Results.LinearGas.Fpv	1.0005375	Float	REAL
- AFC69.Meters[0].Results.LinearGas.CPrime	1.0560492	Float	REAL

4) Meter Type = Linear & Product Group = Liquid

Calculation Result	Controller Tag	Data Type
Non-Resettable Net Accumulator	AFC69.Meters[x].Results.LinearLiquid.Net_Accumulator	DINT
Net Flow Rate	AFC69.Meters[x].Results.LinearLiquid.Net_Flowrate	REAL
Gross Accumulator	AFC69.Meters[x].Results.LinearLiquid.Gross_Accumulator	DINT
Gross Standard Accumulator	AFC69.Meters[x].Results.LinearLiquid.Gross_Standard_Accumulator	DINT
Mass Accumulator	AFC69.Meters[x].Results.LinearLiquid.Mass_Accumulator	DINT

Where x can assume values between 0 and 7.

The following is an example for Meter 1:

[-] AFC69.Meters[0].Results.LinearLiquid	{ ... }		Meter_Pulse_Liquid_Input
+ AFC69.Meters[0].Results.LinearLiquid.Net_Accumulator	777777	Decimal	DINT
- AFC69.Meters[0].Results.LinearLiquid.Net_Flowrate	0.0	Float	REAL
+ AFC69.Meters[0].Results.LinearLiquid.Gross_Accumulator	0	Decimal	DINT
+ AFC69.Meters[0].Results.LinearLiquid.Gross_Standard_A...	1065357725	Decimal	DINT
+ AFC69.Meters[0].Results.LinearLiquid.Mass_Accumulator	1065823391	Decimal	DINT

3.4.12 Checking Meter Alarms

The ladder logic continuously informs if a meter has an alarm or not. Refer to the following controller tags for the meter alarm status:

Information	Controller Tag	Values
Meter 1 Alarm Status	AFC69.Site_Alarms.Meter1	0 = Meter 1 does not have alarm 1 = Meter 1 has alarm
Meter 2 Alarm Status	AFC69.Site_Alarms.Meter2	0 = Meter 2 does not have alarm 1 = Meter 2 has alarm
Meter 3 Alarm Status	AFC69.Site_Alarms.Meter3	0 = Meter 3 does not have alarm 1 = Meter 3 has alarm
Meter 4 Alarm Status	AFC69.Site_Alarms.Meter4	0 = Meter 4 does not have alarm 1 = Meter 4 has alarm
...
Meter 8 Alarm Status	AFC69.Site_Alarms.Meter8	0 = Meter 8 does not have alarm 1 = Meter 8 has alarm

In the following example, Meters 1 and 2 have alarms. The other meters do not:

- AFC69.Site_Alarms		{ ... }
<input checked="" type="checkbox"/>	-AFC69.Site_Alarms.Meter1	1
<input checked="" type="checkbox"/>	-AFC69.Site_Alarms.Meter2	1
<input type="checkbox"/>	-AFC69.Site_Alarms.Meter3	0
<input type="checkbox"/>	-AFC69.Site_Alarms.Meter4	0
<input type="checkbox"/>	-AFC69.Site_Alarms.Meter5	0
<input type="checkbox"/>	-AFC69.Site_Alarms.Meter6	0
<input type="checkbox"/>	-AFC69.Site_Alarms.Meter7	0
<input type="checkbox"/>	-AFC69.Site_Alarms.Meter8	0

For each meter, the ladder logic also indicates which alarm was generated. Refer to the following controller tags for meter alarm information:

Information	Controller Tag	Values
Meter 1 Alarm	AFC69.Meters[0].Results.Alarms	<i>Please see following table</i>
Meter 2 Alarm	AFC69.Meters[1].Results.Alarms	<i>Please see following table</i>
Meter 3 Alarm	AFC69.Meters[2].Results.Alarms	<i>Please see following table</i>
Meter 4 Alarm	AFC69.Meters[3].Results.Alarms	<i>Please see following table</i>
...
Meter 8 Alarm	AFC69.Meters[7].Results.Alarms	<i>Please see following table</i>

Each Alarm word is interpreted as follows:

Bit Number	Description
0	Input out of range: Temperature
1	Input out of range: Pressure
2	Input out of range: Differential Pressure
3	Input out of range: Flowing Density
4	Input out of range: Water Content
5	Differential Pressure Low
6	Orifice Pressure Exception

Bit Number	Description
7	Accumulation Overflow
8	Orifice Characterization Error
9	Analysis Total Zero
10	Analysis Total Not Normalized
11	AGA 8 Calculation Error
12	API Calculation Error: Density Correction
13	API Calculation Error: Ctl
14	API Calculation Error: Vapor Pressure
15	API Calculation Error: Cpl

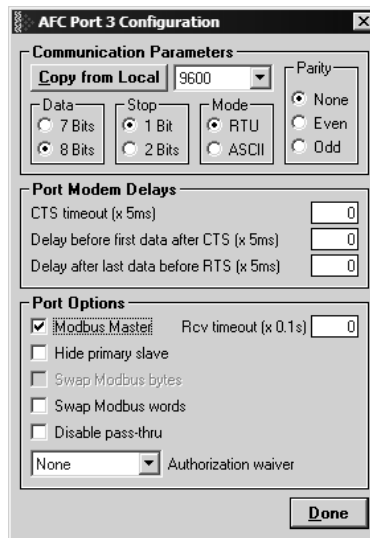
The following illustration shows an example where the Meter 1 has an "Input Out of Range: Temperature" alarm:

▶	[-] AFC69.Meters[1].Results	{...}
+	[-] AFC69.Meters[1].Results.BID	2#0010_0010_0000_0001
+	[-] AFC69.Meters[1].Results.Alarms	2#0000_0000_0000_0001
+	[-] AFC69.Meters[1].Results.OrificeGas	{...}
+	[-] AFC69.Meters[1].Results.OrificeLiquid	{...}
+	[-] AFC69.Meters[1].Results.PulseGas	{...}
+	[-] AFC69.Meters[1].Results.PulseLiquid	{...}

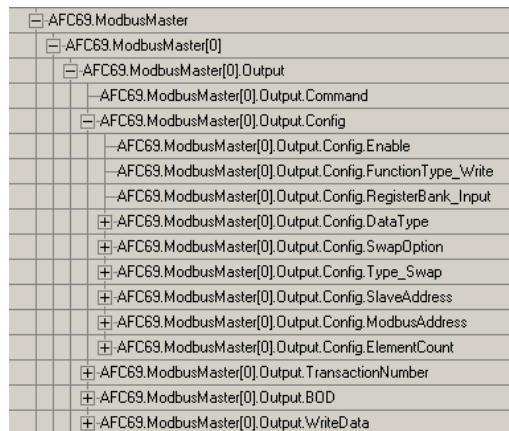
3.4.13 Modbus Master

This block performs an arbitrary data transfer between the PLC and external Modbus slaves connected to AFC port 3, provided that port 3 is configured as a Modbus master. Any data transfer to or from a slave's holding registers, input registers, output coils, or input status may be implemented using this function; equivalent Modbus function codes are 1, 2, 3, 4, 15, and 16. In addition, capability is provided for access to a slave's "long remote" (32-bit) registers where the slave implements them; in particular, Daniel-style long integer (5000 series) and floating point (7000 series) registers are accessible. Any data words not relevant to the command are ignored upon output (to the AFC) and zero upon input (from the AFC).

This feature requires port 3 to be configured as a Modbus master through AFC Manager as follows:



In order to enable this functionality set AFC69.ModbusMaster[0].Output.Command controller tag bit to 1.



The user should configure the following parameters for the command to be sent:

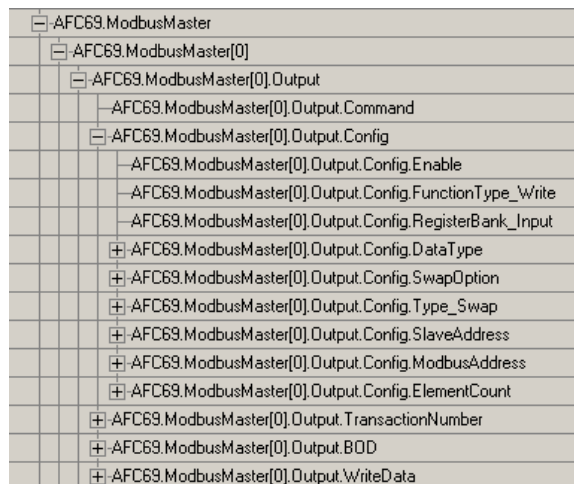
Parameter	Use this Controller Tag	Description
Enable	<i>AFC69.ModbusMaster[0].Output.Command</i>	This value should be set as 1 in order to enable the command. If this value is 0 the module will ignore the Modbus master command
Function Type	<i>AFC69.ModbusMaster[0].Output.Config.FunctionType_Write</i>	0 = Read from Modbus slave 1 = Write to Modbus slave
Register Bank	<i>AFC69.ModbusMaster[0].Output.Config.RegisterBank_Input</i>	0 = Holding Register 1 = Input Register

Parameter	Use this Controller Tag	Description
Data Type = Bit	<i>AFC69.ModbusMaster[0].Output.Config. DataType.Bit</i>	0 = Don't use bit 1 = Use bit
Data Type = Word	<i>AFC69.ModbusMaster[0].Output.Config. DataType.Word</i>	0 = Don't use word 1 = Use word
Data Type = Long	<i>AFC69.ModbusMaster[0].Output.Config. DataType.Long</i>	0 = Don't use long 1 = Use long
Data Type = Long Remote	<i>AFC69.ModbusMaster[0].Output.Config. DataType.LongRemote</i>	0 = Don't use long remote 1 = Use long remote
Swap Bytes	<i>AFC69.ModbusMaster[0].Output.Config. SwapOptions.SwapBytes</i>	0 = Don't swap bytes 1 = Swap bytes (Not valid for bits)
Swap Words	<i>AFC69.ModbusMaster[0].Output.Config. SwapOptions.SwapWords</i>	0 = Don't swap words 1 = Swap words (Not valid for bits or words)
Slave Address	<i>AFC69.ModbusMaster[0].Output.Config. SlaveAddress</i>	Modbus slave address
Modbus Address	<i>AFC69.ModbusMaster[0].Output.Config. ModbusAddress</i>	Register address in the Modbus slave
Element Count	<i>AFC69.ModbusMaster[0].Output.Config. ElementCounts</i>	Number of elements to be transferred
Transaction Number	<i>AFC69.ModbusMaster[0].Output. TransactionNumber</i>	This number will be echoed in the input block. It allows to multiplex more than one command in the same logic.

The following variables are automatically built by the ladder logic and the user does not have to modify these values:

AFC69.ModbusMaster[0].Output.Config.Type_Swap

AFC69.ModbusMaster[0].Output.BOD



- If the Modbus Master command is set as a READ function type the data will be read to the following controller tag:

AFC69.ModbusMaster[0].Input.ReadData[]

- If the Modbus Gateway command is set as a **WRITE** function type the data will be written from the following controller tag:

AFC69.ModbusMaster[0].Output.WriteData[]

- If any Modbus error occurs it will be copied to the following data tag:

AFC69.ModbusMaster[0].Input.ErrorCode

The valid Modbus error codes are listed below:

Error	Description
=0	No error
>0	Modbus exception code or communication error Modbus exception codes are issued by the responding slave and listed in commonly available Modbus protocol manuals; they lie between 1 and 127, and include:
1	Illegal function
2	Illegal address
3	Illegal data value

Communication errors are issued by the AFC:

Error	Description
500	CTS timeout
501	Receive timeout
502	Bad framing
503	Buffer overrun
504	Bad checksum/CRC
505	Wrong slave
506	Wrong function code
507	Wrong length
<0	Configuration, parameter, or logic error:
-1	Master port not configured
-2	Master port never used
-3	Bad slave address
-4	Bad direction / target
-5	Bad datum size / swap options
-6	Bad number of data items

3.4.14 Modbus Pass-Thru

In order to use pass-thru, enable the following controller tag:

AFC69.ModbusPassThru.Output.Enable (BOOL)

Once the module receives the Modbus write command the data will be copied to the following controller tag:

AFC69.ModbusPassThru.Input.ReadData_Word[] (word commands)

AFC69.ModbusPassThru.Input.ReadData_Bit[] (bit commands)

- AFC56.ModbusPassThru	{...}
- AFC56.ModbusPassThru.Output	{...}
+ AFC56.ModbusPassThru.Output.BOD	2#0001_1000_0110_0110
- AFC56.ModbusPassThru.Output.Enable	1
- AFC56.ModbusPassThru.Input	{...}
+ AFC56.ModbusPassThru.Input.BID	2#0001_1000_0000_0000
- AFC56.ModbusPassThru.Input.MessagePresent	0
- AFC56.ModbusPassThru.Input.Function_BitWrite	0
- AFC56.ModbusPassThru.Input.OverflowError	0
+ AFC56.ModbusPassThru.Input.ModbusAddress	2200
+ AFC56.ModbusPassThru.Input.RegisterCount	5
+ AFC56.ModbusPassThru.Input.ReadData_Word	{...}
+ AFC56.ModbusPassThru.Input.ReadData_Bit	{...}

3.4.15 Modbus Gateway

The ladder logic can be used to read or write data from one of the internal Modbus Slaves. Any data that is not part of the AFC69.Meters[].Results must be read or written through the Modbus Gateway blocks.

Each block can transfer up to 200 words of data and uses a specific *AFC69.Modbus.Gateway[]* controller tag. Each one of these tags must be configurable in order to read or write data between the module and the processor.

Perform the following steps to use the Modbus Gateway blocks:

- 1 Identify how many words (total) will be transferred. The sample ladder supports up to 2000 words.
- 2 Based on the number of registers to be transferred, calculate how many Modbus Gateway blocks will be necessary. Each block contains up to 200 registers. For example: if 700 registers will be used to transfer data, 4 Modbus Gateway blocks will be required.
- 3 Based on the number of Modbus Gateway blocks, configure the *AFC69.Modbus.BlockCount* controller tag.

- AFC69.ModbusGateway	{...}	Modbus
+ AFC69.ModbusGateway.CommandCount	0	Decimal INT
+ AFC69.ModbusGateway.Command	{...}	ModbusGateway[10]

For example, if you configure the number of blocks as 4, the ladder logic automatically sends the following Modbus Gateway blocks to the module:

AFC69.Modbus.Gateway[0]
AFC69.Modbus.Gateway[1]
AFC69.Modbus.Gateway[2]
AFC69.Modbus.Gateway[3]

The maximum number of blocks is 10.

If the *AFC69.Modbus.BlockCount* controller tag is configured as 0 the module will not send any Modbus Gateway blocks.

- 1 Refer to the Modbus Map document and identify the addresses of all registers in the Primary Slave.
- 2 Using the AFC Manager, re-map the registers from the Primary Slave to the Virtual Slave (refer to AFC Manager User Manual for more information about this subject).
- 3 You must also set a Virtual Slave Address greater than 0 in order to activate the Virtual Slave.
- 4 In the sample ladder logic, configure each Modbus Gateway Block using the following controller tags:

Parameter	Controller Tag	Values
Enable Transaction	AFC69.ModbusGateway.Command[0].Config. Enable	0 = The module will ignore this request 1 = The module will process this request.
Start Register	AFC69.ModbusGateway.Command[0].Config. StartRegister	Start register in the Modbus Slave to be written or read from
Register Count	AFC69.ModbusGateway.Command[0].Config. RegisterCount	Number of words to be written or read between the module and the processor.
Function Type	AFC69.ModbusGateway.Command[0].Config. FunctionType_Write	0 = Read from MVI69-AFC 1 = Write to MVI69-AFC
Register Type	AFC69.ModbusGateway.Command[0].Config. RegisterType_Input	0 = Holding Register 1 = Input Register
Slave Type	AFC69.ModbusGateway.Command[0].Config. SlaveType_Virtual	0 = Primary Slave 1 = Virtual Slave

Note: It is strongly suggested that you first configure all parameters having the Enable bit set to 0. After the configuration is finished than the Enable bit can be set to 1.

In order to write 50 words from the processor to the Primary Modbus Slave starting at holding register address 2000, the *AFC69.ModbusGateway.Command[0]* block should be configured as follows:

[-] AFC69.ModbusGateway.Command[0]	{...}		ModbusGateway
[-] AFC69.ModbusGateway.Command[0].Config	{...}		ModbusConfig
- AFC69.ModbusGateway.Command[0].Config.Enable	1	Decimal	BOOL
+ AFC69.ModbusGateway.Command[0].Config.StartRegister	2000	Decimal	INT
+ AFC69.ModbusGateway.Command[0].Config.RegisterCount	50	Decimal	INT
- AFC69.ModbusGateway.Command[0].Config.FunctionType_Write	1	Decimal	BOOL
- AFC69.ModbusGateway.Command[0].Config.RegisterType_Input	0	Decimal	BOOL
- AFC69.ModbusGateway.Command[0].Config.SlaveType_Virtual	0	Decimal	BOOL

- 1 Refer to the *AFC69.ModbusGateway.Command[x].WriteData* or *AFC69.ModbusGateway.Command[x].ReadData* controller tags depending on the configured function type:
- 2 If the Modbus Gateway block uses a **READ** function type:
The data will be read to the *AFC69.ModbusGateway.Command[x].ReadData[]* array.
If the Modbus Gateway block uses a **WRITE** function type:
The data will be written from the *AFC69.ModbusGateway.Command[x].WriteData[]* array.

Note: The BID and BOD controller tags are automatically generated by the ladder logic so you do not have to write any value to these controller tags.

3.4.16 Reset Resettable Accumulator

The sample ladder logic shows how to reset all resettable accumulators for a meter. However, most customers will prefer to reset the resettable accumulators when the archives are created. The module can be configured to automatically reset the resettable accumulators upon period end.

Refer to Accumulators for more information about this issue.

Some customers may decide to reset the Differential accumulators from the ladder logic (especially if using batch operation).

In this case you should toggle the following bit commands:

Command	Use this Controller Tag	Description
Enable	<i>AFC69.MeterSignals.Enable</i>	This value should be set as 1 in order to enable the command. If this value is 0 the module will ignore the reset commands from the ladder
Select Meter	<i>AFC69.MeterSignals.Meter</i>	Select the meter number (1 to 16) for the reset command
Reset Resettable Accumulator 1	<i>AFC69.MeterSignals.Signals.Res_Acc1</i>	If this bit is set to 1 the module will reset Acc1 for the selected meter. The ladder logic will reset this command.
Reset Resettable Accumulator 2	<i>AFC69.MeterSignals.Signals.Res_Acc2</i>	If this bit is set to 1 the module will reset Acc2 for the selected meter. The ladder logic will reset this command.

Command	Use this Controller Tag	Description
Reset Resetable Accumulator 3	<i>AFC69.MeterSignals.Signals.Res_Acc3</i>	If this bit is set to 1 the module will reset Acc3 for the selected meter. The ladder logic will reset this command.
Reset Resetable Accumulator 4	<i>AFC69.MeterSignals.Signals.Res_Acc4</i>	If this bit is set to 1 the module will reset Acc4 for the selected meter. The ladder logic will reset this command.

Note: The *AFC69.MeterSignals.BID* and *AFC69.MeterSignals.Action* tags are automatically updated by the ladder logic so you do not have to enter any value for this tag.

The following illustration shows this data structure. Note that the Sel_Stream bits are not used for this MVI69-AFC version.

- AFC69.MeterSignals	{...}
- AFC69.MeterSignals.Enable	1
+ AFC69.MeterSignals.BID	12289
+ AFC69.MeterSignals.Meter	1
+ AFC69.MeterSignals.Action	0
- AFC69.MeterSignals.Signals	{...}
- AFC69.MeterSignals.Signals.Select_Stream1	0
- AFC69.MeterSignals.Signals.Select_Stream2	0
- AFC69.MeterSignals.Signals.Select_Stream3	0
- AFC69.MeterSignals.Signals.Select_Stream4	0
- AFC69.MeterSignals.Signals.Res_Acc1	0
- AFC69.MeterSignals.Signals.Res_Acc2	0
- AFC69.MeterSignals.Signals.Res_Acc3	0
- AFC69.MeterSignals.Signals.Res_Acc4	0
- AFC69.MeterSignals.Signals.Wr_Daily_Archive	0
- AFC69.MeterSignals.Signals.Wr_Hrly_Archive	0
- AFC69.MeterSignals.ResetOnce	1
+ AFC69.MeterSignals.LastBlockID	3

3.4.17 Write Hourly/Daily Archive

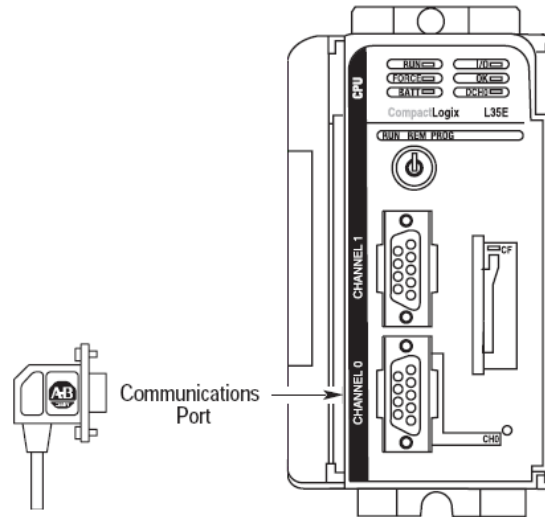
The sample ladder logic shows how to create hourly or daily archive. It is very important to notice that the sample ladder automatically creates both archives, depending on the configured **End-of-day minute** and **End-of-hour minute** parameters entered in the AFC Manager. Most users will not need to generate the archives from ladder logic.

In order to create archives from ladder logic, refer to the following data tags:

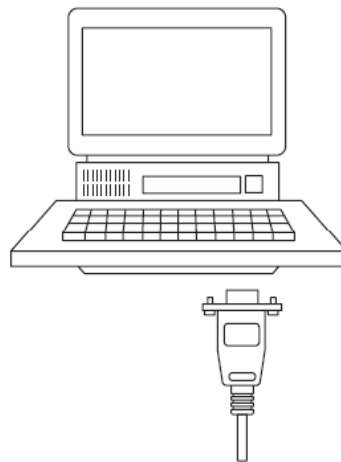
Command	Use this Controller Tag	Description
Enable	<i>AFC69.MeterSignals.Enable</i>	This value should be set as 1 in order to enable the command. If this value is 0 the module will ignore the commands from the ladder
Select Meter	<i>AFC69.MeterSignals.Meter</i>	Select the meter number (1 to 8) for the archive command
Write Daily Archive	<i>AFC69.MeterSignals.Signals.Wr_Daily_Archive</i>	If this bit is set to 1 the module will generate a daily archive
Write Hourly Archive	<i>AFC69.MeterSignals.Signals.Wr_Hrly_Archive</i>	If this bit is set to 1 the module will generate an hourly archive

3.4.18 Connecting Your PC to the Processor

- 1 Connect the right-angle connector end of the cable to your controller at the communications port.



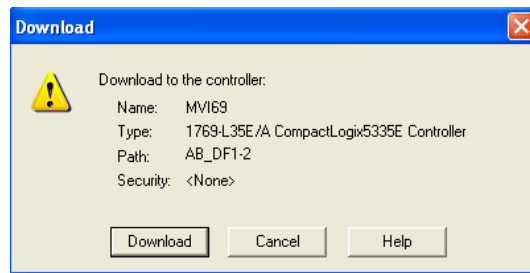
- 2 Connect the straight connector end of the cable to the serial port on your computer.



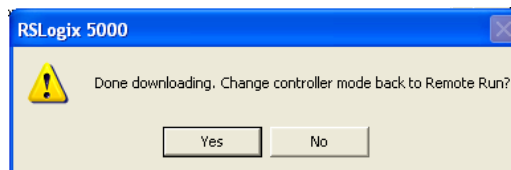
3.4.19 Downloading the Sample Program to the Processor

Note: The key switch on the front of the CompactLogix processor must be in the REM or PROG position.

- 1 If you are not already online to the processor, open the **COMMUNICATIONS** menu, and then choose **DOWNLOAD**. RSLogix will establish communication with the processor.
- 2 When communication is established, RSLogix will open a confirmation dialog box. Click the **DOWNLOAD** button to transfer the sample program to the processor.



- 3 RSLogix will compile the program and transfer it to the processor. This process may take a few minutes.
- 4 When the download is complete, RSLogix will open another confirmation dialog box. Click **OK** to switch the processor from PROGRAM mode to RUN mode.

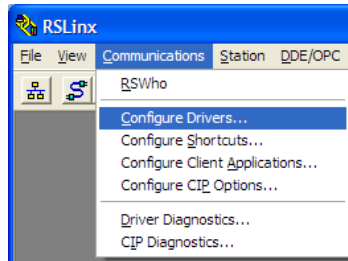


Note: If you receive an error message during these steps, refer to your RSLogix documentation to interpret and correct the error.

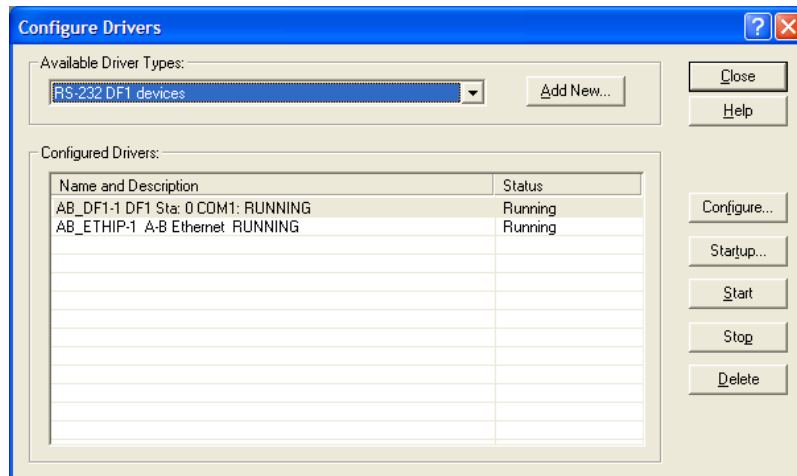
Configuring the RSLinx Driver for the PC COM Port

When trying to connect serially, if RSLogix is unable to establish communication with the processor, follow these steps.

- 1 Open *RSLinx*.
- 2 Open the **COMMUNICATIONS** menu, and click **CONFIGURE DRIVERS**.



This action opens the *Configure Drivers* dialog box.



Note: If the list of configured drivers is blank, you must first choose and configure a driver from the *Available Driver Types* list. The recommended driver type to choose for serial communication with the processor is *RS-232 DF1 Devices*.

- 3 Click to select the driver, and then click **CONFIGURE**. This action opens the *Configure RS-232 DF1 Devices* dialog box.



- 4 Click the **AUTO-CONFIGURE** button. RSLinx will attempt to configure your serial port to work with the selected driver.
- 5 When you see the message *Auto Configuration Successful*, click the **OK** button to dismiss the dialog box.

Note: If the auto-configuration procedure fails, verify that the cables are connected correctly between the processor and the serial port on your computer, and then try again. If you are still unable to auto-configure the port, refer to your RSLinx documentation for further troubleshooting steps.

3.5 MicroLogix Sample Ladder File

This section shows how the data files can be used for different tasks.

3.5.1 Status

The sample ladder logic informs the MVI69-AFC status as follows:

Site Status

The site status information is copied at every scan to the following data files:

Bit	Description
B3:4/0	Module Released
B3:4/1	Checksum Alarm
B3:4/2	Reserved
B3:4/3	Reserved
B3:4/4	Processor Offline or missing
B3:4/5	Measurement Configuration Changed
B3:4/6	Power up
B3:4/7	Cold-start
B3:4/8	Modbus master poll pending
B3:4/9	Modbus master poll complete
B3:4/9	Pass-thru input pending
B3:4/10	Reserved
B3:4/11	Reserved
B3:4/12	Reserved
B3:4/13	Reserved
B3:4/14	Reserved
B3:4/15	Reserved

Meter Status

The sample ladder logic continuously sends sub-blocks that allow the processor to enable or disable the meters at any time. The sub-block response sent by the module informs the status of each meter. This information is available as follows:

Enable/Disable status

Bit	Meter
B3:0/0	1
B3:0/1	2
B3:0/2	3
B3:0/3	4
B3:0/4	5
B3:0/5	6
B3:0/6	7
B3:0/7	8

Where:

ON = meter is enabled

OFF = meter is disabled

The meter alarm information is updated at every scan to the following processor data files:

Alarm Status

Bit	Meter
B3:1/0	1
B3:1/1	2
B3:1/2	3
B3:1/3	4
B3:1/4	5
B3:1/5	6
B3:1/6	7
B3:1/7	8

Where:

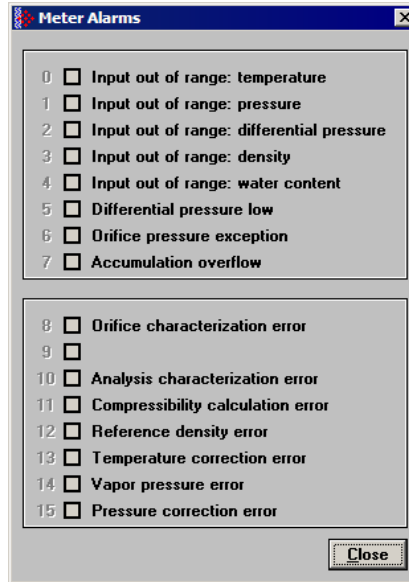
ON = meter has alarms

OFF = meter does not have alarms

The actual meter alarm is also copied to the processor at the following data files:

Register	Meter
N20:21	1
N20:22	2
N20:23	3
N20:24	4
N20:25	5
N20:26	6
N20:27	7
N20:28	8

The alarm register is a bitmapped word. The AFC Manager also shows the possible alarm values (Meter Monitor). For example, considering that the meter 1 has alarms as follows:



Register	Meter	Value
N20:21	1	0401 (hex)

The meter configuration (meter type and product group) is available at the following registers:

Meter Type

Bit	Meter
B3:2/0	1
B3:2/1	2
B3:2/2	3
B3:2/3	4
B3:2/4	5
B3:2/5	6
B3:2/6	7
B3:2/7	8

Where:

ON = meter type is configured as linear

OFF = meter type is configured as differential

Bit	Meter
B3:3/0	1
B3:3/1	2
B3:3/2	3

Bit	Meter
B3:3/3	4
B3:3/4	5
B3:3/5	6
B3:3/6	7
B3:3/7	8

Where:

ON = product group is configured as liquid

OFF = product group is configured as gas

3.5.2 Enable/Disable Meters

A meter will only perform flow calculation if enabled. There are two situations where a meter should be disabled:

- 1 The meters that are not being used should be disabled in order to improve module performance.
- 2 Some configuration parameters cannot be modified if the meters are enabled. The user should disable the meter, change the parameter and enable the meter. The parameters that require this procedure are listed below:
 - o Meter type
 - o Product Group
 - o Units
 - o Density Units
 - o Primary Input

The following bits can be used to enable the meters

Register	Enable Meter
B3:11/0	1
B3:11/1	2
B3:11/2	3
B3:11/3	4
B3:11/4	5
B3:11/5	6
B3:11/6	7
B3:11/7	8

The following bits can be used to disable the meters

Register	Disable Meter
B3:12/0	1
B3:12/1	2
B3:12/2	3
B3:12/3	4
B3:12/4	5
B3:12/5	6
B3:12/6	7
B3:12/7	8

Note: The user should not latch these command bits. These bits should be toggled once. The sample ladder logic will reset these bits After the status returned by the module indicates that the command was successful.

3.5.3 WallClock

The following registers store the date and time wallclock to be sent to the module. This logic is required to be sent at every module power up.

Register	Description
N105:0	Year
N105:1	Month
N105:2	Day
N105:3	Hour
N105:4	Minute
N105:5	Seconds

The sample ladder logic uses the processor’s wallclock (S:37) to update these registers After the block is sent. The sample ladder logic may be modified to use another wallclock data (other than the processor).

The following bit requests the wallclock update block to the module:

Bit	Description
B102:0/0	WallClock Command

3.5.4 Process Calculation

The sample ladder logic continuously writes the process variables and reads the calculation results from the module. The process variable block contains 11 words that are interpreted according to the meter type and product group.

Process Variables: Output

The following floating point data files copy process variables from the processor to the module:

Data File	Meter
F30	1
F31	2
F32	3
F33	4
F34	5
F35	6
F36	7
F37	8

The following table lists the data file description according to meter's type and product (meter 1):

	Meter Type	Differential	Differential	Linear	Linear
	Product Group	Gas	Liquid	Gas	Liquid
F30:0		Temperature	Temperature	Temperature	Temperature
F30:1		Pressure	Pressure	Pressure	Pressure
F30:2		Differential Pressure	Differential Pressure	Pulse Frequency	Density
F30:3		Not Used	Density	Not Used	Pulse Frequency

For linear meter applications it is required the transfer of the pulse count that is typically obtained from a high speed counter module. This value is assumed to use a long integer (32-bit) format. Therefore, the following registers store the pulse count values to be used by the module:

Pulse Count

Register	Meter
L10:0	1
L10:1	2
L10:2	3
L10:3	4
L10:4	5
L10:5	6
L10:6	7
L10:7	8

For liquid applications it is required the transfer of the water percentage contained in the product. This value is assumed to use a scaled integer (16-bit) format. Therefore, the following data files are used for water percentage values to be used by the module:

Water %

Register	Meter
N20:0	1
N20:1	2
N20:2	3
N20:3	4
N20:4	5
N20:5	6
N20:6	7
N20:7	8

The water percentage uses two decimal places. For example, a value of 105 is interpreted as 1.05%.

Calculation Results - Input

The following floating point data files copy process variables from the processor to the module:

Data File	Meter
F30	1
F31	2
F32	3
F33	4
F34	5
F35	6
F36	7
F37	8

The following table shows the data file description according to the meter's type and product (meter 1):

	Meter Type	Differential	Differential	Linear	Linear
	Product Group	Gas	Liquid	Gas	Liquid
F30:5		Net Flow Rate	Net Flow Rate	Net Flow Rate	Net Flow Rate
F30:6		Gross Flow Rate	Reserved	Gross Flow Rate	Reserved
F30:7		Fpv	Reserved	Fpv	Reserved
F30:8		Cprime	Reserved	Cprime	Reserved

The following table shows the long data file description according to the meter's type and product (meter 1):

	Meter Type	Differential	Differential	Linear	Linear
	Product Group	Gas	Liquid	Gas	Liquid
L10:5		Net Accumulator	Net Accumulator	Net Accumulator	Net Accumulator
L10:6		Reserved	Gross Accumulator	Reserved	Gross Accumulator
L10:7		Reserved	Gross Standard Accumulator	Reserved	Gross Standard Accumulator
L10:8		Reserved	Mass Accumulator	Reserved	Mass Accumulator

3.5.5 Molar Analysis Update

Certain applications may require the processor to update the molar analysis data using a gas chromatograph device. In order to use this feature the following bit should be latched as ON.

Register	Description
B102:0/1	Update analysis

Important: this bit should be kept as ON all the time in order for the processor to continuously update the molar analysis data.

The following data files store the molar analysis data for each meter:

Register	Description
N40	Analysis Meter 1
N41	Analysis Meter 2
N42	Analysis Meter 3
N43	Analysis Meter 4
N44	Analysis Meter 5
N45	Analysis Meter 6
N46	Analysis Meter 7
N47	Analysis Meter 8

The data structure for each data file is shown below:

Register	Code	Description
N40:0	C1	methane
N40:1	N2	nitrogen
N40:2	CO2	carbon dioxide
N40:3	C2	ethane
N40:4	C3	propane
N40:5	H2O	water
N40:6	H2S	hydrogen sulfide
N40:7	H2	hydrogen
N40:8	CO	carbon monoxide
N40:9	O2	oxygen
N40:10	IC4	iso-butane
N40:11	NC4	normal butane
N40:12	IC5	iso-pentane
N40:13	NC5	normal pentane
N40:14	C6	hexane
N40:15	C7	heptane
N40:16	C8	octane
N40:17	C9	nonane
N40:18	C10	decane
N40:19	He	helium

Register	Code	Description
N40:20	Ar	argon
N40:21	neoC5	neopentane
N40:22	Ux	user 1
N40:23	Uy	user 2

The module expects the molar concentration to be in scaled integer format, where: 10000 = 100%. For example, 9000 = 90%.

For firmware version 2.05.000 (or later) you can also select the for which stream number the molar analysis data will transferred to. Select a value of 0 to update the active stream or enter values between 1 to 4 to update specific streams.

Note: If the processor updates a non-active stream, the new molar data will not be visible through AFC Manager until that stream becomes the active one.

Register	Description
N40:30	Analysis Stream Selection - Meter 1
N41:30	Analysis Stream Selection - Meter 2
N42:30	Analysis Stream Selection - Meter 3
N43:30	Analysis Stream Selection - Meter 4
N44:30	Analysis Stream Selection - Meter 5
N45:30	Analysis Stream Selection - Meter 6
N46:30	Analysis Stream Selection - Meter 7
N47:30	Analysis Stream Selection - Meter 8

3.5.6 Meter Signals

This feature allows the processor to reset a resettable accumulator or generate an archive at any time. This application is usually required for batch applications.

The following data files are used for each meter:

Register	Description
B3:20	Signals - Meter 1
B3:21	Signals - Meter 2
B3:22	Signals - Meter 3
B3:23	Signals - Meter 4
B3:24	Signals - Meter 5
B3:25	Signals - Meter 6
B3:26	Signals - Meter 7
B3:27	Signals - Meter 8

The following table shows the data structure for meter 1. The other meters will use the exact same format:

Register	Description
B3:20/0	Select Stream 1 (for firmware versions 2.05.000 or later)
B3:20/1	Select Stream 2 (for firmware versions 2.05.000 or later)
B3:20/2	Select Stream 3 (for firmware versions 2.05.000 or later)
B3:20/3	Select Stream 4 (for firmware versions 2.05.000 or later)
B3:20/4	Reset Resettable Accumulator 1
B3:20/5	Reset Resettable Accumulator 2
B3:20/6	Reset Resettable Accumulator 3
B3:20/7	Reset Resettable Accumulator 4
B3:20/8	Write Daily Archive
B3:20/9	Write Hourly Archive
B3:20/10	Reserved
B3:20/11	Reserved
B3:20/12	Reserved
B3:20/13	Reserved
B3:20/14	Reserved
B3:20/15	Reserved

Note: After the processor receives the input block associated to the meter it will reset these bits. Therefore the logic that latches these bits should be added at the rung that copies the meter signals to the output block (routine 4/rung 6). Otherwise these bits could be cleared before being sent to the module.

For firmware version 2.05.000 or later, the processor also monitors which is the active stream through the following registers:

Register	Description
N40:30	Active stream number (1 to 4) - Meter 1
N41:30	Active stream number (1 to 4) - Meter 2
N42:30	Active stream number (1 to 4) - Meter 3
N43:30	Active stream number (1 to 4) - Meter 4
N44:30	Active stream number (1 to 4) - Meter 5
N45:30	Active stream number (1 to 4) - Meter 6
N46:30	Active stream number (1 to 4) - Meter 7
N47:30	Active stream number (1 to 4) - Meter 8

3.5.7 Archive Fetch

This feature allows the processor to read any archive from the module. The following registers must be configured in order to use this feature:

Register	Description
N90:0	Archive File Select (0=Daily Archive, 1=Hourly Archive)
N90:1	Archive Age (0=current, 1=latest, and so on.)
N90:2	Meter

The following bit should be toggled to request the meter archive information from the module:

Register	Description
B102:0/5	Archive fetch command

The following data files store the archive data:

Data File	Meter
N91	1
N92	2
N93	3
N94	4
N95	5
N96	6
N97	7
N98	8

The opening and closing timestamp are stored in a highly compressed format in the module. The sample ladder logic converts these values into a more convenient format starting at offset 40 for each data file listed in the previous table. The following table lists the registers used for meter 1:

Register	Description
N91:40	Opening timestamp - Month
N91:41	Opening timestamp - Day
N91:42	Opening timestamp - Year
N91:43	Opening timestamp - Hour
N91:44	Opening timestamp - Minute
N91:45	Opening timestamp - Second
N92:40	Closing timestamp - Month
N92:41	Closing timestamp - Day
N92:42	Closing timestamp - Year
N92:43	Closing timestamp - Hour
N92:44	Closing timestamp - Minute
N92:45	Closing timestamp - Second

This block returns an archive record to the Processor, selected according to the criteria in the Output block. Archive fetch may be controlled through the use of the "ignore output" bit of the block id; the "skip input" bit has no function and is ignored.

3.5.8 Modbus Pass-Thru

The user should configure the pass-thru region at the AFC Manager tool in order to use this feature. Refer to the Modbus section for more information about this feature. Every time the module receives a Modbus pass-thru command from a remote master it will set a specific indication bit. The sample ladder uses this information to actually poll for the Modbus pass-thru data. The module responds to this block with the Modbus pass-thru data.

The following registers contain the status of the Modbus pass-thru block:

Register	Description
N80:0/0	Command present
N80:0/1	Data format (ON = bit, OFF = word)
N80:0/2	Overflow - command too long
N80:1	Register address
N80:2	Register count

The following data files are reserved to read the Modbus pass-thru data into the processor:

Data File	Description
N81	Data from word command (Modbus function 16)
B82	Data from word command (Modbus function 15)

3.5.9 Modbus Master

In order to use this feature the port 3 must be configured for Modbus master operation through the AFC Manager tool.

The following registers must be configured prior of sending the Modbus master block to the module:

Register	Description
N70:0/0	Register Type (OFF = Holding, ON = Input)
N70:0/1	Function Type (OFF = Read, ON = Write)
N70:0/2	Data Item Size - Bit (OFF = Not Bit, ON = Bit)
N70:0/3	Data Item Size - Word (OFF = Not Word, ON = Word)
N70:0/4	Data Item Size - Long (OFF = Not Long, ON = Long)
N70:0/5	Data Item Size - Long Remote (OFF = Not Long, ON = Long)
N70:0/6	Byte Swap (OFF = No swap, ON = Swap bytes)
N70:0/7	Word Swap (OFF = no Swap, ON = swap words) - only for Long types
N70:1	Slave address
N70:2	Register Address
N70:3	Register Count
N70:4	Transaction Number

The following data file stores the data associated to the Modbus command. If it is a read command it stores the data read from the remote slave. If it is a write command it stores the data to be written to the remote slave.

Register	Description
N71:0	Modbus master data

The following bit should be toggled in order to send the Modbus master command:

Bit	Description
B102:0/2	Modbus master command

3.5.10 Modbus Gateway

This feature can be used to send Modbus gateway transactions from the processor to the module. It can be used to update the primary or virtual slave's database.

The following register must have a fixed value that shows the number of Modbus gateway blocks to be used:

Register	Description
N50:0	Modbus gateway transaction count

If this value is equal to zero then the sample ladder logic does not send Modbus gateway blocks. The sample ladder logic supports up to 10 blocks. If an application requires more than that then minor ladder logic changes must be implemented.

The following registers must be configured before sending a Modbus gateway transaction. Note that the transaction will only be performed if the *Enable Transaction* bit is set.

Register	Gateway Transaction	Description
N50:10/0	1	Slave (OFF = Primary, ON = Virtual)
N50:10/1	1	Register Type (OFF = Holding, ON = Input)
N50:10/2	1	Function Type (OFF = Read, ON = Write)
N50:10/3	1	Enable Transaction (OFF = Disable, ON = Enable)
N50:11	1	Register Address
N50:12	1	Number of registers
N50:13	1	Transaction number - this number is copied back in the input block.
N50:15/0	2	Slave (OFF = Primary, ON = Virtual)
N50:15/1	2	Register Type (OFF = Holding, ON = Input)
N50:15/2	2	Function Type (OFF = Read, ON = Write)
N50:15/3	2	Enable Transaction (OFF = Disable, ON = Enable)
N50:16	2	Register Address
N50:17	2	Number of registers
N50:18	2	Transaction number - this number is copied back in the input block.
N50:20/0	3	Slave (OFF = Primary, ON = Virtual)
N50:20/1	3	Register Type (OFF = Holding, ON = Input)
N50:20/2	3	Function Type (OFF = Read, ON = Write)
N50:20/3	3	Enable Transaction (OFF = Disable, ON = Enable)
N50:21	3	Register Address
N50:22	3	Number of registers
N50:23	3	Transaction number - this number is copied back in the input block.
N50:25/0	4	Slave (OFF = Primary, ON = Virtual)
N50: 25/1	4	Register Type (OFF = Holding, ON = Input)

Register	Gateway Transaction	Description
N50:25/2	4	Function Type (OFF = Read, ON = Write)
N50:25/3	4	Enable Transaction (OFF = Disable, ON = Enable)
N50:26	4	Register Address
N50:27	4	Number of registers
N50:28	4	Transaction number - this number is copied back in the input block.
N50:30/0	5	Slave (OFF = Primary, ON = Virtual)
N50:30/1	5	Register Type (OFF = Holding, ON = Input)
N50:30/2	5	Function Type (OFF = Read, ON = Write)
N50:30/3	5	Enable Transaction (OFF = Disable, ON = Enable)
N50:31	5	Register Address
N50:32	5	Number of registers
N50:33	5	Transaction number - this number is copied back in the input block.
N50:35/0	6	Slave (OFF = Primary, ON = Virtual)
N50:35/1	6	Register Type (OFF = Holding, ON = Input)
N50:35/2	6	Function Type (OFF = Read, ON = Write)
N50:35/3	6	Enable Transaction (OFF = Disable, ON = Enable)
N50:36	6	Register Address
N50:37	6	Number of registers
N50:38	6	Transaction number - this number is copied back in the input block.
N50:40/0	7	Slave (OFF = Primary, ON = Virtual)
N50:40/1	7	Register Type (OFF = Holding, ON = Input)
N50:40/2	7	Function Type (OFF = Read, ON = Write)
N50:40/3	7	Enable Transaction (OFF = Disable, ON = Enable)
N50:41	7	Register Address
N50:42	7	Number of registers
N50:43	7	Transaction number - this number is copied back in the input block.
N50:45/0	8	Slave (OFF = Primary, ON = Virtual)
N50:45/1	8	Register Type (OFF = Holding, ON = Input)
N50:45/2	8	Function Type (OFF = Read, ON = Write)
N50:45/3	8	Enable Transaction (OFF = Disable, ON = Enable)
N50:46	8	Register Address
N50:47	8	Number of registers
N50:48	8	Transaction number - this number is copied back in the input block.
N50:50/0	9	Slave (OFF = Primary, ON = Virtual)
N50:50/1	9	Register Type (OFF = Holding, ON = Input)

Register	Gateway Transaction	Description
N50:50/2	9	Function Type (OFF = Read, ON = Write)
N50:50/3	9	Enable Transaction (OFF = Disable, ON = Enable)
N50:51	9	Register Address
N50:52	9	Number of registers
N50:53	9	Transaction number - this number is copied back in the input block.
N50:55/0	10	Slave (OFF = Primary, ON = Virtual)
N50:55/1	10	Register Type (OFF = Holding, ON = Input)
N50:55/2	10	Function Type (OFF = Read, ON = Write)
N50:55/3	10	Enable Transaction (OFF = Disable, ON = Enable)
N50:56	10	Register Address
N50:57	10	Number of registers
N50:58	10	Transaction number - this number is copied back in the input block.

The data associated with the Modbus gateway transaction is copied to/from one of the following data files.

For a read transaction this data file contains the data that was read from the module. For a write transaction this register contains the source data to be written to the module.

Data File	Gateway Transaction
N51	1
N52	2
N53	3
N54	4
N55	5
N56	6
N57	7
N58	8
N59	9
N60	10

4 Module Configuration

In This Chapter

❖ Connect your PC to the Module	85
❖ Install AFC Manager.....	86
❖ Starting AFC Manager.....	87
❖ Using AFC Manager.....	88
❖ Creating Your Application.....	94
❖ Module Initialization.....	94

This chapter provides an overview of the most important aspects of module configuration.

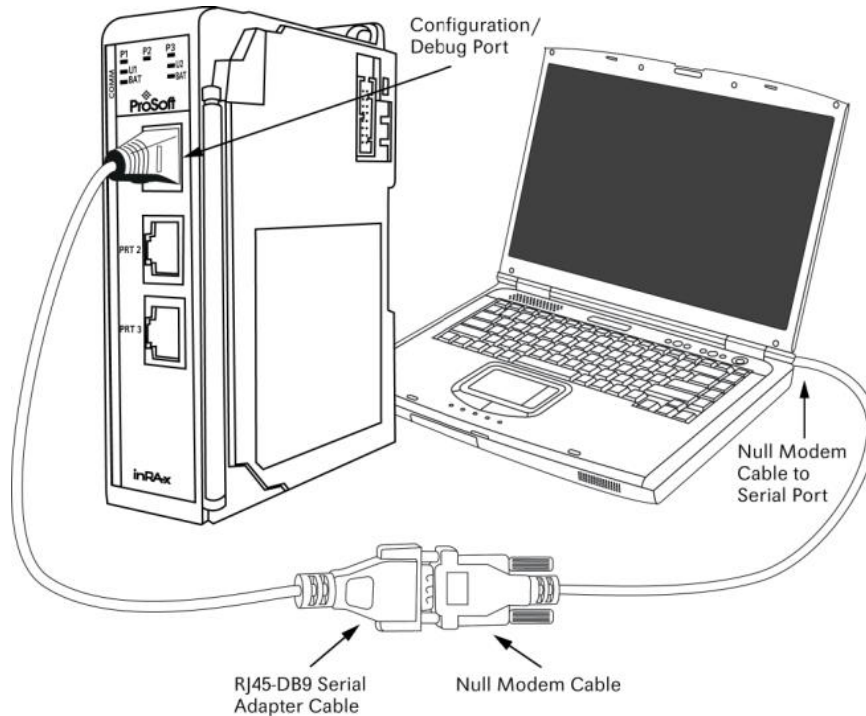
Configuration and setup for all ProSoft Technology Flow Computers are accomplished using AFC Manager software, provided with the module at no extra charge. This software has its own detailed User Manual that discusses configuration options and parameters in great detail. Please refer to the AFC Manager User Manual for additional configuration information.

4.1 Connect your PC to the Module

With the module securely mounted, connect your PC to the Configuration/Debug port using an RJ45-DB-9 Serial Adapter Cable and a Null Modem Cable.

- 1 Attach both cables as shown.
- 2 Insert the RJ45 cable connector into the Configuration/Debug port of the module.

- 3 Attach the other end to the serial port on your PC.



4.2 Install AFC Manager

The AFC Manager application is included on the Prosoft Solution DVD shipped with your module. You can also get it from the ProSoft Technology website. Download and install AFC Manager to your PC.

4.2.1 System Requirements

The following system requirements are the recommended minimum specifications to successfully install and run AFC Manager:

- Microsoft Windows compatible PC
- Windows 2000 with Service Pack 2 or higher, or Windows XP Professional with Service Pack 2 or higher, or Windows 2003.
- 300 MHz Pentium processor (or equivalent)
- 128 megabytes of RAM
- 20 megabytes of free disk space
- Available ethernet port, serial port (COM port) or USB to Serial adapter cable with necessary drivers, required for communication between AFC Manager software and the AFC module.
- DB9 adapter cable (included with module), required for connection between PC serial port and AFC module (PTQ-AFC module does not require an adapter).

To install the AFC Manager application:

- 1** Insert the ProSoft Solutions DVD in your DVD drive. On most computers, a menu screen will open automatically. If you do not see a menu within a few seconds, follow these steps:
 - a** Click the Start button, and then choose Run.
 - b** In the Run dialog box, click the Browse button.
 - c** In the Browse dialog box, click "My Computer". In the list of drives, choose the CD-ROM drive where you inserted the ProSoft Solutions DVD.
 - d** Select the file **prosoft.exe**, and then click Open.
 - e** On the Run dialog box, click OK.
- 2** On the menu, click Documentation and Tools. This action opens a Windows Explorer dialog box.
- 3** Open the Utilities folder, and then open the AFCManager folder.
- 4** Double-click the file Setup.exe. If you are prompted to restart your computer so that files can be updated, close all open applications, and then click OK. When your computer has finished restarting, begin again at Step 1.
- 5** Click OK or Yes to dismiss any confirmation dialog boxes.
- 6** It may take a few seconds for the installation wizard to start. Click OK on the AFC Manager Setup dialog box to begin installing AFC Manager.
- 7** Follow the instructions on the installation wizard to install the program with its default location and settings.
- 8** When the installation finishes, you may be prompted to restart your computer if certain files were in use during installation. The updated files will be installed during the restart process.

4.3 Starting AFC Manager

To start AFC Manager:

- 1** Click the **START** button, and then choose **PROGRAMS**.
- 2** In the Programs menu, choose ProSoft Technology.
- 3** In the ProSoft Technology menu, choose AFC Manager.

At this point, you should have the main AOI imported into your RSLogix project and you have AFC Manager running on your PC. You can now use AFC Manager to create your application.

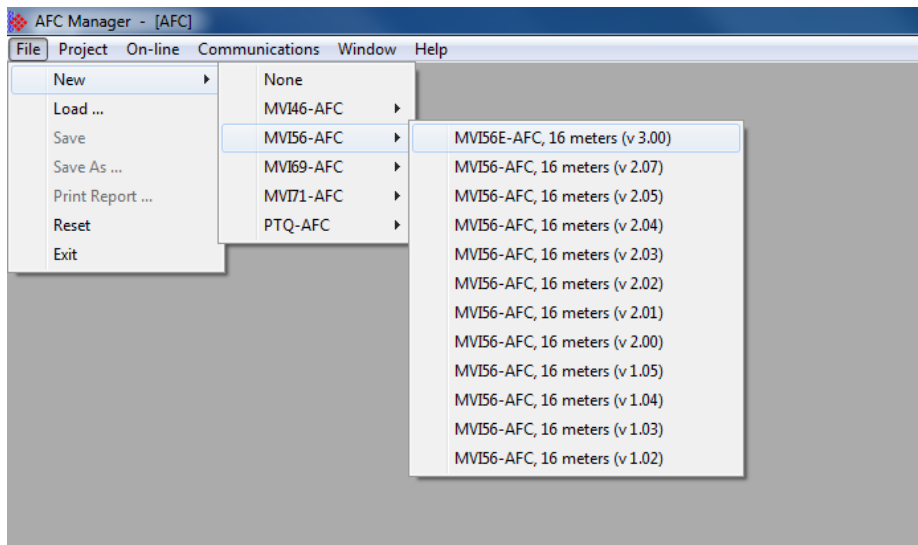
4.4 Using AFC Manager

The AFC module is configured with configuration files that you create using AFC Manager. A configuration file is called a Project.

4.4.1 Starting a New Project

To start a new project:

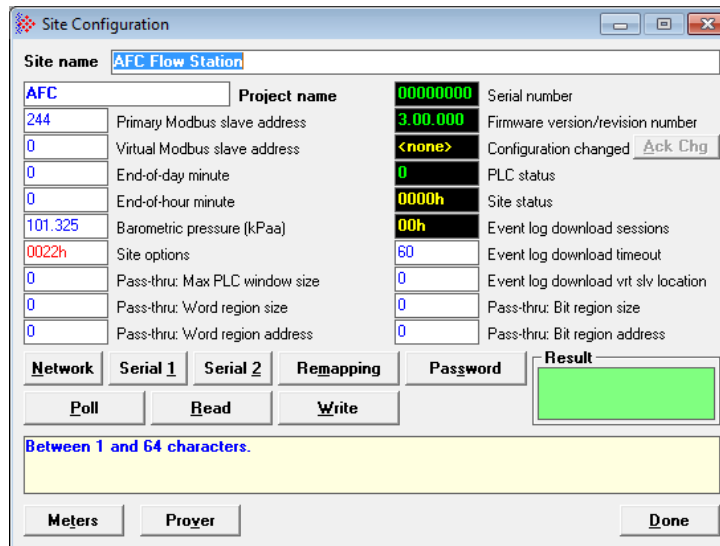
- 1 Start **AFC MANAGER**, and then open the *File* Menu.
- 2 On the *File* Menu, choose **NEW**, and then select your module and firmware version number.



The version number refers to the firmware version of your module. If you do not know the firmware version number, follow these steps:

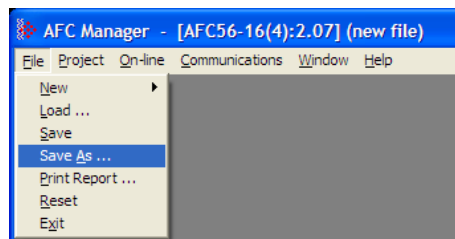
- a) Open the *Project* menu.
- b) Choose **SITE CONFIGURATION**. This action opens the *Site Configuration Dialog Box*.

- c) Click the **READ** Button. The firmware version is listed below the serial number, in the upper right part of the dialog box.



Important: You must be connected to the module and "online" to read data from the module.

- 3 Follow the steps in the remainder of this User Guide to configure your module and your AFC device.
- 4 Before closing the program, open the *File* menu and choose **SAVE AS**, to save your project so you can open it again later.



4.4.2 Loading an Existing project

You can open and edit a project you have previously saved. Do this if you have started, but not completed, the configuration of your project, or if you need to modify the settings for a project that has already been downloaded to the module.

To load an existing project:

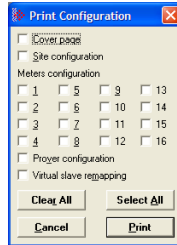
- 1 Start **AFC MANAGER**, and then open the *File* menu.
- 2 On the *File* menu, choose **LOAD**. This action opens a dialog box that shows a list of AFC Manager project files (AFC files) in the current folder.
- 3 Choose the project to load, and then click **OPEN**.

4.4.3 Printing the Configuration Report

You can print a report of your configuration for future reference, or for archival purposes.

To print the configuration report:

- 1 Open the *File* menu, and then select **PRINT REPORT**. This action opens the *Print Configuration* dialog box.



- 2 On the *Print Configuration* dialog box, select (check) the items to include in the printed report.
- 3 Click **PRINT** to send the report to your default printer.

Note: The size of the report depends on items you choose to include, and may require 75 pages or more. Consider this before printing.

4.4.4 Converting a Project

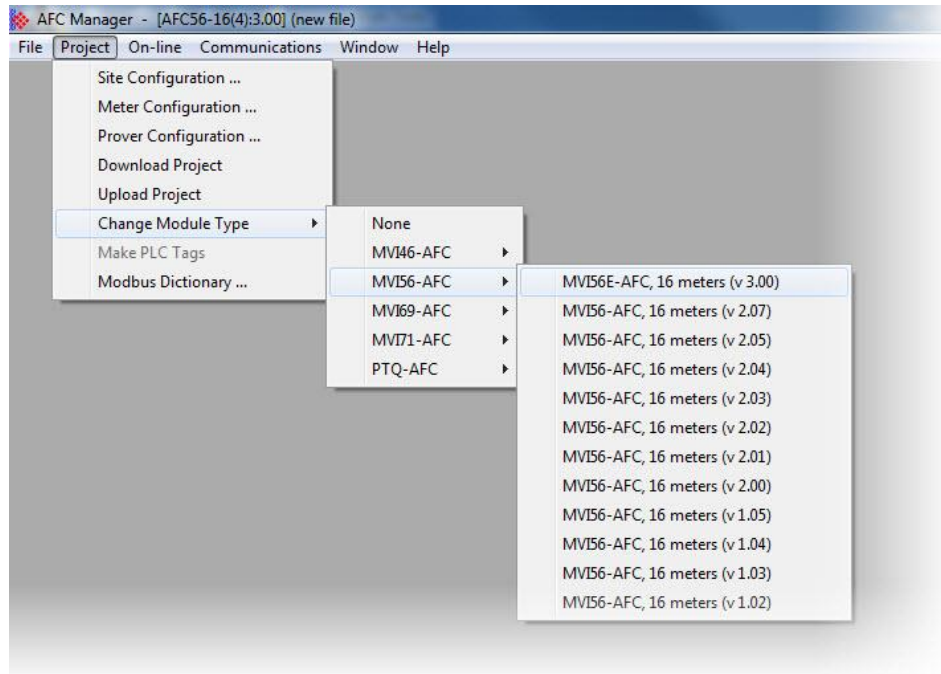
You can convert an existing project (configuration file) to use it with a different module or firmware version. Do this if:

- You want to reuse an application created for a different AFC module, for example a project that was created for a PTQ-AFC that you want to use for an MVI69-AFC.
- You apply a firmware upgrade to a module.

To convert a project:

- 1 Open the *File* menu, and then choose **OPEN**.
- 2 Open the project (configuration file) to convert.

- 3 Open the *Project* menu, and then choose **CHANGE MODULE TYPE**.



- 4 Choose the module type and firmware version from the menu.
- 5 Save your project.

Note: AFC Manager will save your updated configuration file with the same name as the file you loaded. If you need to keep your original configuration, change the file name of your updated configuration before saving.

4.4.5 Resetting Configuration Parameters

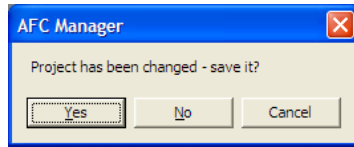
If you have modified your project (configuration file), or if you have loaded a configuration file from disk, but you want to start a new project, you can reset the configuration parameters back to their defaults without having to close and reopen the AFC Manager.

To reset configuration parameters

- 1 Close any dialog boxes that are open.
- 2 Save the configuration file you were working on, if you would like to load it again later.
- 3 On the *File* menu, choose **RESET**.

Note: This procedure has the same effect as choosing **File / New / None**.

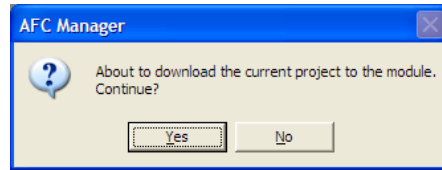
If you have made changes to the configuration that have not yet been saved, a confirmation dialog box will open.



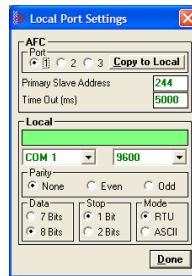
Answer Yes to save your changes, or No to discard your changes and begin working on a new configuration. Click Cancel to abandon the attempted action that caused this message.

4.4.6 Downloading the Project to the Module

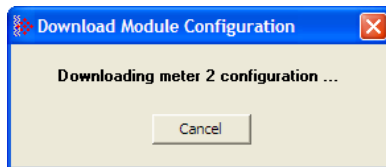
- 1 Click **PROJECT / DOWNLOAD PROJECT**.



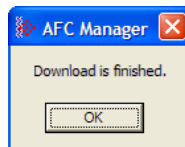
- 2 This action opens the Local Port Settings window. Enter the port parameters to use, and then click **DONE**.



- 3 During the download operation, the following progress window is displayed:



- 4 When the file transfer is complete, the following window is displayed:



Note: The Virtual Slave (page 130) is not downloaded during the procedure because it requires a separate download operation.

Troubleshooting Tip: If the AFC Manager displays an "Illegal Data Value" message, it typically indicates an invalid meter type or product group configuration. The module does not accept a configuration file that attempts to change a meter type or product group for a meter that is currently enabled. Disable all meters, change the meter types and product groups, and then enable the meters again.

4.4.7 Verifying Correct Operation

When all of the configuration steps have been completed, the module should be ready to perform measurement calculations. To verify that the module is configured correctly, follow these steps:

- 1** Enable all meters that will be used, as any meter will only perform calculations if it is enabled. Any meter can be enabled either with ladder logic (MVI69-AFC modules), function blocks (PTQ modules) or with AFC Manager.
- 2** Make sure that the wallclock is running, and that it has valid date and time information. After power-up, the wallclock will be stopped, therefore the module will not perform any time-scheduled operations, such as writing period-end archives, and will not timestamp records written to the event log until it receives a wallclock command from the ladder logic.

The sample ladder logic programs the wallclock update command upon detecting "power-up" status from the AFC. The date/time information used is the same as the processor, therefore you should use the configuration tool for your processor to verify that the processor has valid date/time data. If the processor wallclock is not valid (for example if the year = 1900), the module will not accept the command. You may easily determine if the wallclock is running by performing two consecutive read operations in the Meter Monitor.

- 3** Make sure that the meter does not have any alarms. A meter alarm may affect flow calculation. Look at the Meter Monitor dialog box for alarms.
- 4** Make sure that the input parameters transferred from the processor are correct. You can look at these values in the Meter Monitor dialog box.
- 5** When using a pulse meter, make sure that the pulse input rollover parameter in Meter Configuration matches the actual input rollover value used in the high speed counter module.

4.5 Creating Your Application

After power-up, the module must receive valid wallclock data from the ladder logic to perform time-scheduled operations and to properly timestamp historical records. The sample ladder logic automatically writes the wallclock upon detecting power-up status from the AFC using the processor's date and time information. You should ensure that the processor contains valid date and time information. If it does not, the module may not accept the wallclock block.

You can verify the wallclock information using the Meter Monitor section as shown in the following example:



2007-07-16 14:59:28 Wallclock

Refer to the Sample Ladder Logic section for more information on this topic.

4.6 Module Initialization

When the module is powered up for the first time, both the **OK** and **ERR** BBRAM LEDs are illuminated. This indicates that the module is in the *Cold Start* state and is not yet ready to perform calculations. The following steps initialize the module:

- Configure Site Parameters
- Enable at least one meter
- Set the processor to RUN mode

After these three steps are accomplished, the state is changed from *Cold Start* to *Released*. This indicates that that module is ready to perform flow calculations. When in the *Released* state, the **OK** LED is ON and the **ERR** LED is off.

When the module is ready, you will use AFC Manager to monitor meter operation, archives, and events. The *AFC Manager User Manual* contains detailed information on these tasks.

5 Diagnostics and Troubleshooting

In This Chapter

❖ User LEDs.....	96
❖ BBRAM LEDs.....	96
❖ Meter Alarms.....	97
❖ Checksum Alarms	101
❖ Events	101
❖ Audit Scan.....	101

MVI69-AFC modules have the following communication connections on the module:

- Two RS-232/422/485 Application ports
- One RS-232 Configuration/Debug port

This section provides information that will assist you during the module operation on troubleshooting issues. This section describes the following topics:

- LEDs
- Meter Alarms
- Checksum Alarms
- Events
- Audit Scan

5.1 User LEDs

There are two "user" LEDs used to indicate overall module status; App Status and BP Act (with P1, P2, or P3).

5.1.1 App Status LED

State	Description
Rapid Blinking	The processor is offline (probably in program mode).
Steady On	Some meter is indicating an alarm or no meters are enabled.
Off	The module is functioning properly.

5.1.2 BP Act and P1, P2, or P3

These LEDs indicate current Modbus traffic on any port.

State	Description
On	A Modbus command for the module is recognized. On Port 3, this LED may also indicate that a Modbus Master command was sent.
Off	No Activity

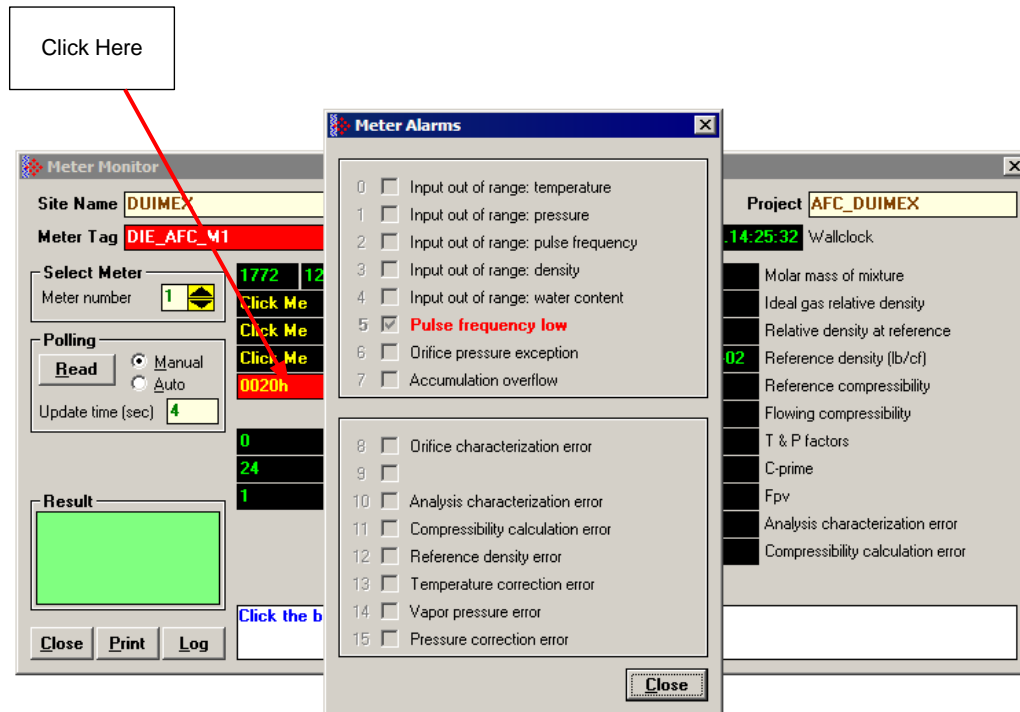
5.2 BBRAM LEDs

The BBRAM (Battery Backed RAM) LEDs inform you about the condition of the BBRAM hardware used for data storage. The following table lists the possible situations that might occur during normal operation.

OK (Green)	ERR (Red)	Description
ON	ON	The module is in a Cold Start condition that typically occurs when you power up the module for the first time. After at least one meter is enabled and the processor is in RUN mode the module starts operating.
ON	OFF	Normal Operation
Blinking	OFF	This condition is warning that a checksum flag was raised after a power cycle. If this alarm issue occurs, refer to the AFC Manager (On-line Monitor / Checksum Alarms) in order to determine the data section in which the alarm issue has occurred. After verifying that the checksum error has not affected the referred memory area you may clear the checksum alarm using the same AFC Manager interface. After the alarm is cleared the OK LED will be ON

5.3 Meter Alarms

If the module is generating unexpected data, you should verify if the meter has any alarms. Some alarms may be caused by an issue that could potentially affect the calculation results. Each archive also keeps track of the alarms that have occurred during the period (refer to the Archive section). The Meter Monitor dialog box allows you to monitor the meter alarms.



The above image shows the Meter Alarms bitmap, which gives you a quick overview of active alarms. Associated with many of these bits are Alarm Code registers which supply specific reasons for the alarms, most of which appear in the lower right corner of the main Meter Monitor window. For complete information, including which Code registers are associated with which alarm bits, use the Modbus Dictionary feature of AFC Manager.

The possible alarms are listed in the following table. Of the Alarm Codes listed, the values that can actually appear depend on both the selected Product Group and the firmware version.

Alarm Message	Description	Solution
Accumulation Overflow	The module ignores an accumulator increment of less than zero or greater than 1.000.000.000 occurring in a single meter scan.	Check your meter configuration to verify if your project is generating reasonable values.
Analysis Total Not Normalized ($v \leq 2.04$)	Absolute difference between analysis total and 1.0000 (100%) is greater than the error tolerance	Make sure that the sum of all molar concentrations is within the error tolerance of 1.0000 (100%).
Analysis Total Zero ($v \leq 2.04$)	The molar concentration sum is zero.	Make sure that the sum of all molar concentrations is within the error tolerance of 1.0000 (100%).
Analysis Characterization error ($v \geq 2.05$)	Absolute difference between analysis total and 1.0000 (100%) is greater than the error tolerance, OR the molar concentration sum is zero.	Make sure that the sum of all molar concentrations is within the error tolerance of 1.0000 (100%). Alarm Code values: 0 = No alarm 1 = Analysis total not normalized 2 = Analysis total zero
Compressibility calculation error	The compressibility calculation resulted in error based on the input values and configuration parameters used.	Check the input values and meter configuration parameters. Alarm Code values: 0 = No alarm 1 = Density exceeded reasonable maximum (warning only) 2 = Pressure maximum found 3 = Non-convergence of procedure "braket" 4 = Non-convergence of procedure "ddetail"
Differential Pressure Low	The differential pressure value transferred to the module is below the DP Alarm Threshold parameter configured in the Meter Configuration.	Check the input differential pressure value transferred to the module. If the value is correct, change the DP Alarm Threshold parameter for your project.
Flow Rate Low	The flow rate value transferred to the module is below the FR Alarm Threshold parameter configured in the Meter Configuration.	Check the input flow rate value transferred to the module. If the value is correct, change the FR Alarm Threshold parameter for your project.
Pulse Frequency Low	The pulse frequency value transferred to the module is below the Frequency Alarm Threshold parameter configured in the Meter Configuration.	Check the input pulse frequency value transferred to the module. If the value is correct, change the Frequency Alarm Threshold parameter for your project.

Alarm Message	Description	Solution
High Water error	Set if input water content is greater than 99% (less than 1% oil). For this condition, the emulsion is deemed to be all water. Both volume and mass fractions are set to zero. The module does not perform any density correction calculation, so the "default standard density" value is assumed. This alarm is applied for emulsion liquids only.	Check that the value of process input "Water %" is reasonable Alarm Code values: 0 = No alarm 1 = Emulsion is more than 99% water
Input Out of Range	The input value is not within the range specified in the meter configuration window. Applies to temperature, pressure, differential pressure, flowing density, water content, pulse frequency ($v \geq 2.05$).	Check that the input variable's ranges (Meter Configuration / Process Input button) and the process input itself have reasonable values.
Orifice Characterization error	The orifice parameters (Meter Configuration / Orifice button) are invalid.	Check the orifice and meter parameters. The following conditions should be true: <ul style="list-style-type: none"> ▪ Orifice diameter > 0 ▪ Tube diameter > 0 ▪ Orifice diameter < Tube diameter The beta ratio between the orifice and tube diameters should follow the AGA Standard. Alarm Code values: <ul style="list-style-type: none"> ▪ 0 = No alarm ▪ 1 = Orifice diameter non-positive ▪ 2 = Orifice not narrower than pipe ▪ 3 = Beta ratio less than 0.10 (adjusted by tolerance) ▪ 4 = Beta ratio greater than 0.75 (adjusted by tolerance) ▪ 5 = Pipe diameter less than 2.0 inches (adjusted by tolerance) ▪ 6 = Orifice diameter less than 0.45 inches (adjusted by tolerance) The "tolerance", fixed by the AFC firmware, allows the AGA limits to be exceeded by up to 75% towards the physical limit. For example, while AGA restricts pipe diameter to 2.0 inches or greater, the AFC allows it to be as small as 0.5 inch.
Orifice Pressure Exception	Configuration and process input for an Orifice Meter are such that the effective downstream pressure is less than vacuum. For calculation, upstream pressure is raised by the amount necessary to raise absolute downstream pressure to zero.	Check the process inputs for Gauge Pressure and Differential Pressure, and the configured Barometric Pressure and Static Pressure Tap Location. Also check any performed vapor pressure calculations to ensure that all are reasonable.

Alarm Message	Description	Solution
Pressure correction error	The pressure correction calculation resulted in an error according to the standard.	Alarm Code values: 0 = No alarm 1 = Density outside range of API Chapter 11.2 2 = Temperature above near critical limit 3 = Temperature outside range of API Chapter 11.2.1 4 = Temperature outside range of API Chapter 11.2.2 5 = Non-convergence of Cpl-density iteration
Reference density error	The density correction calculation resulted in an error according to the standard.	Alarm Code values:: 0 = No alarm 1 = Low density (NGLs), input outside API range 2 = High density (crudes & refined), input outside API range 3 = Non-convergence 4 = Zero VCF 5 = Temperature above critical point 6 = Input density outside reference fluid adjusted range 7 = Corrected density out of range 8 = Standard density input outside API range 9 = Alpha input outside API range Also check the input values and calculation parameters for your project.
Temperature Correction error	The temperature correction calculation OR the water temperature correction calculation resulted in an error according to the standard.	Alarm Code values: 0 = No alarm 1 = Low density (NGLs), input outside API range 2 = High density (crudes & refined), input outside API range 5 = Temperature above critical point 9 = Alpha input outside API range Also see the Alarm Code for Water Temperature Correction error.
Vapor pressure error	The vapor pressure calculation resulted in an error according to the standard.	Alarm Code values: 0 = No alarm 1 = Expected vapor pressure above range of TP-15 (stream's "Default Vapor Pressure" is substituted) 2 = Vapor pressure > measured static absolute pressure (vapor pressure assumed to equal static pressure) 3 = Both 1 and 2
Water Temperature error (Alarm Code only)	The water temperature correction calculation resulted in an error according to the standard. This Alarm Code sets the "Temperature Correction error" alarm bit.	Alarm Code values: 0 = No alarm 1 = Temperature < 0°C (32°F) or > 138°C (280°F)

5.4 Checksum Alarms

A checksum alarm indicates a checksum verification failure during power-up. Non-volatile information is kept in battery-backed RAM. It is partitioned into several blocks, each of which contains a checksum, and when the information is changed the checksum is updated also. During power-up, the checksum is verified, and upon failure the alarm bit is latched and the checksum corrected.

The alarm bit remains latched, even through subsequent power cycles, until it is explicitly cleared from an external source such as the AFC Manager. Refer to the AFC Manager User Manual for more information about this feature.

5.5 Events

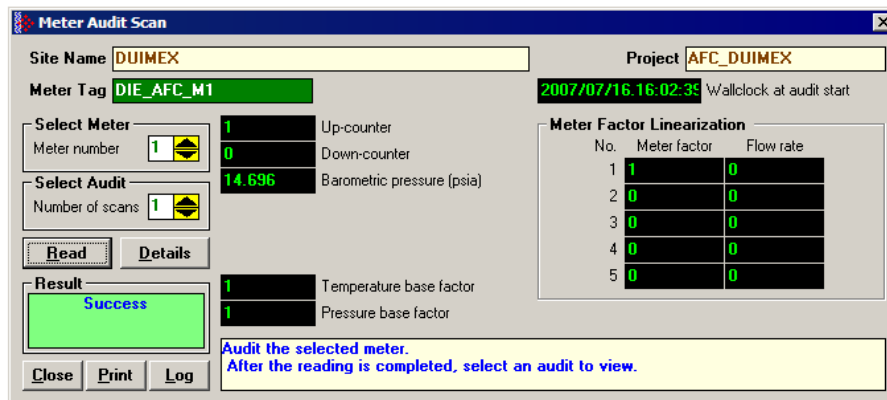
The module records up to 1999 events that have occurred during the module operation.

Important Note: Events are occurrences that may affect the results calculated by the module. This is an essential tool for troubleshooting the module.

Refer to the Events section for more information about event monitor.

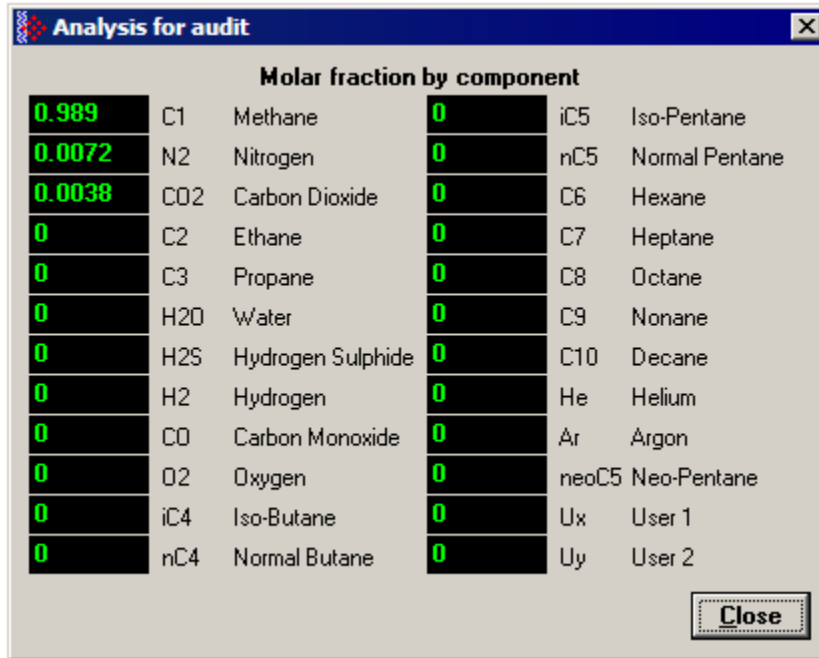
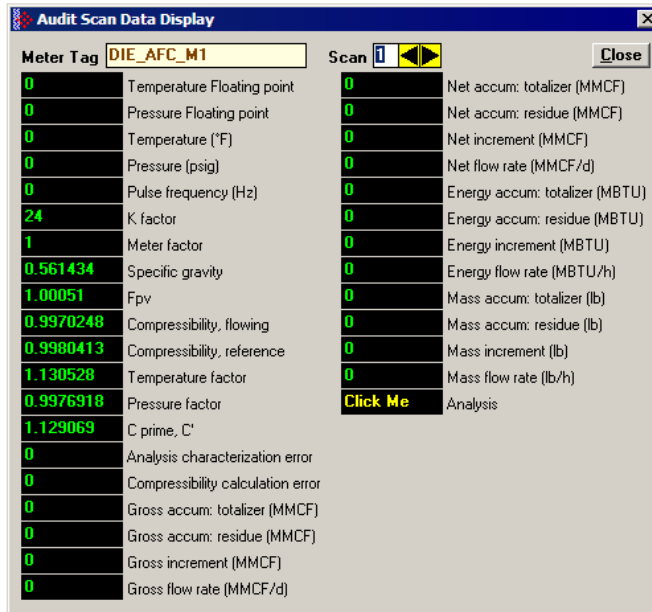
5.6 Audit Scan

An Audit Scan captures a "snapshot" of input values, intermediate calculated values, and output results for each of a short series of calculation scans for a single meter. This allows an auditor to rigorously verify the calculations performed by the AFC on live in-service production meters. The module supports eight consecutive audit scans at a time.



- 1 Select the Meter Number for the audit
- 2 Select the number of scans for the audit
- 3 Click the Read Button to begin the audit
- 4 Look at the operation result. Success = audit has been successfully completed

- When the Audit Scan is complete, click the Details Button to view the calculation and the input variables.



The following shows an example of an audit scan file report generated by the AFC Manager for 2 scans:

AFC-56(16) Audit Date: 16-09-2002 16:18:07
 Site Name: MVI Flow Station
 Project: AFC

Meter 1:

Tag		M01
Wallclock		0000/00/00.00:00:00
Barometric pressurekPaa		101,325
Viscosity		0,010268
Orifice/pipe geometric parameters		
	Orifice plate	Meter tube
Temperature	68	68
Diameter	1	2
Coefficient	9,25E-06	0,0000062

Scan		1
Temperature (Floating point)		15
Pressure (Floating point)		1000
Dif. pressure (Floating point)		22
Temperature (°F)		15
Pressure (psig)		1000
Dif. pressure (hw)		22
Scan period (second)		0,48
Specific gravity		0,7404104
Fpv		0
Compressibility flowing		0,9051347
Compressibility reference		0,9989105
Diameter at T tube		1,999343
Diameter at T orifice		0,9995098
Velocity of approach factor ev		1,032773
Pressure extension xt		149,4683
Coefficient of discharge cd		0,6042569
Expansion factor y		0,9997441
Composition factor		0,2728558
Mass flow Qh		2280,571
Orifice characterization error		0
Analysis characterization error		0
AGA8 calculation error		0
Gross accu. - totalizer (x f3)		3408
Gross accu. - residue (x f3)		0,2047686
Gross increment (x f3)		6,442598E-02
Gross flow rate (x f3/h)		483,1948
Net accu. - totalizer (x f3)		390113
Net accu. - residue (x f3)		0,8464546
Net increment (x f3)		5,3664
Net flow rate (x f3/h)		40248
Mass accu. - totalizer (x lb)		22094
Mass accu. - residue (x lb)		0,5677222
Mass increment (x lb)		0,3040761
Mass flow rate (x lb/h)		2280,571

Analysis components	
C1 methane	0,55
N2 nitrogen	0,45
CO2 carbon dioxide	0
C2 ethane	0
C3 propane	0
H2O water	0
H2S hydrogen sulphide	0
H2 hydrogen	0
CO carbon monoxide	0
O2 oxygen	0
iC4 iso-butane	0
nC4 normal butane	0
iC5 iso-pentane	0
nC5 normal pentane	0
C6 hexane	0
C7 heptane	0
C8 octane	0
C9 nonane	0
C10 decane	0
He helium	0
Ar argon	0
neoC5 neopentane	0
Ux user1	0
Uy user2	0

AFC-56(16) Audit
Site Name: MVI Flow Station
Project: AFC

Date: 16-09-2002 16:18:08

Meter 1:

Tag		M01
Wallclock		0000/00/00.00:00:00
Barometric pressurekPaa		101,325
Viscosity		0,010268
Orifice/pipe geometric parameters		
	Orifice plate	Meter tube
Temperature	68	68
Diameter	1	2
Coefficient	9,25E-06	0,0000062

Scan	2
Temperature (Floating point)	15
Pressure (Floating point)	1000
Dif. pressure (Floating point)	22
Temperature (°F)	15
Pressure (psig)	1000
Dif. pressure (hw)	22
Scan period (second)	0,495
Specific gravity	0,7404104
Fpv	0
Compressibility flowing	0,9051347
Compressibility reference	0,9989105
Diameter at T tube	1,999343

Diameter at T orifice	0,9995098
Velocity of approach factor ev	1,032773
Pressure extension xt	149,4683
Coefficient of discharge cd	0,6042569
Expansion factor y	0,9997441
Composition factor	0,2728558
Mass flow Qh	2280,571
Orifice characterization error	0
Analysis characterization error	0
AGA8 calculation error	0
Gross accu.- totalizer (x f3)	3408
Gross accu. - residue (x f3)	0,2712079
Gross increment (x f3)	6,643929E-02
Gross flow rate (x f3/h)	483,1948
Net accu. - totalizer (x f3)	390119
Net accu. - residue (x f3)	0,3805552
Net increment (x f3)	5,534101
Net flow rate (x f3/h)	40248
Mass accu. - totalizer (x lb)	22094
Mass accu. - residue (x lb)	0,8813007
Mass increment (x lb)	0,3135785
Mass flow rate (x lb/h)	2280,571
Analysis components	
C1 methane	0
N2 nitrogen	0
CO2 carbon dioxide	0
C2 ethane	0
C3 propane	0
H2O water	0
H2S hydrogen sulphide	0
H2 hydrogen	0
CO carbon monoxide	0
O2 oxygen	0
iC4 iso-butane	0
nC4 normal butane	0
iC5 iso-pentane	0
nC5 normal pentane	0
C6 hexane	0
C7 heptane	0
C8 octane	0
C9 nonane	0
C10 decane	0
He helium	0
Ar argon	0
neoC5 neopentane	0
Ux user1	0
Uy user2	0

6 Reference

In This Chapter

❖ General Specifications.....	107
❖ Measurement Standards	112
❖ Meter Channels	117
❖ Modbus Database	127
❖ Modbus Communication.....	137
❖ Accumulators.....	142
❖ Archives.....	148
❖ Events	177
❖ Security (Passwords)	192
❖ Cable Connections	196
❖ Frequently Asked Questions	201

6.1 General Specifications

- Process I/O: analog inputs (pressure, temperature, differential pressure density) from analog modules and pulse inputs from pulse/frequency input modules in rack
- Number of meter channels: 8 or 16 meters: differential (AGA3 or ISO5167) or linear (AGA7) Gas; (MPMS Ch 12.2) Liquid.

MVI46-AFC	MVI56E-AFC	MVI69-AFC	MVI71-AFC	PTQ-AFC
8 Meters	16 Meters	8 Meters	8 Meters	16 Meters

Calculation Methods

- AGA3 (1992)
- AGA7
- AGA8 (1992) Detail Characterization Method
- API MPMS Ch 21.1, 21.2
- API Tables (API MPMS Ch 11.1) 23/53 and 24/54 for Hydrocarbon Liquids
- GPA TP-25 for Hydrocarbon Liquids (Tables 23E/24E)
- API MPMS Ch 11.2
- GPA TP-15 for Vapor Pressure Correlation
- Energy (heating value) for gases according to AGA 8 Appendix C-4
- API MPMS Ch 20.1
- ISO 5167

Supports energy measurement for gas applications

Meter I/O Scan Time: Less than one second for all channels.

Product Measurement: Hydrocarbon gases and liquids including refined products

Process I/O Calibration Mode: Allows the calibration of transmitters without interfering with the process update for the module or impacting measurement.

Data Archiving

- Hourly for 2 days for each meter run (48 records per channel)
- Daily for 35 days

Note: The number of archives depends on the archive size you have configured. The default values for a 30 word archive are 48 hourly archives and 35 daily archives.

- Extended Archive feature supports up to 1440 daily archives and 1440 hourly archives stored on Compact Flash
- Each record consists of nearly 20 process and other variables. All archived data is available in the onboard Modbus memory map.
- User may configure when archives are generated
- User may configure archive content (from pre-defined list)
- Archives can be exported to an Excel spreadsheet or printed to a local printer.

Other Features

- Event Log with 1999-event buffer and timestamp.
- Virtual Slave with 20,000 re-mappable Modbus registers for contiguous SCADA polling.
- Password protection

6.1.1 On-line Communication & Configuration

The module is designed for online configuration via the configuration port. A user-friendly Windows 95/98/2000/NT/XP-based Module Configuration and Reporting/Monitoring Manager allows easy access to all configuration data for editing and saving on your computer.

Project configurations may be uploaded, downloaded, and saved to disk under user-selectable filenames. The module takes just minutes to configure using the MS Windows-based AFC Manager.

6.1.2 Reports

- **Event Log Report:** All security-sensitive configuration data (for example, orifice diameter) is date and time stamped and mapped to the local Modbus memory map. This data can be imported into any spreadsheet program and saved to disk or printed to a local printer.
- **Hourly and Daily Archive Reports:** Mapped to local Modbus memory. This data can be imported into any spreadsheet program and saved to disk, or printed as hard copy.
- **System Configuration:** May be transferred to or from the module. The configuration file can also be printed for hard reference or archiving.
- **Audit Scan:** A report can be saved to disk or printed to the local printer.

6.1.3 Modbus Interface

The two Modbus Slave ports allow the unit to be used as a SCADA interface and to broaden access to the AFC module's data table.

- Ports 2 and 3 support RS-232, RS-422 and RS-485 modes
- Supports baud rates of up to 19200 baud
- All ports may be configured for RTU or ASCII Modbus mode.
- All Modbus Slave ports provide access to all configuration and measurement data mapped to the Modbus table.
- Module contains two internal slaves (Primary and Virtual)
- Over 130,000 Modbus registers of the Primary Slave table may be re-mapped to up to 20,000 Modbus registers of the Virtual Slave for contiguous polling from a SCADA master.
- Port 3 can be configured as a Modbus master node
- Supports Modbus functions 3, 4, 5, 6, 15, and 16 as a slave (5 and 15 only on pass-thru operation)
- Supports Modbus functions 1, 2, 3, 4, 15, and 16 as a master
- Scratch Pad Modbus block of 6000 words for transfer of arbitrary data between the processor and the SCADA host via the module.

6.1.4 Configurable Options

Configurable options include:

- Gas analysis concentrations for up to 21 components
- Accumulator Rollover
- Reference temperature and pressure for both gases and liquids
- Orifice and pipe diameters, selection of type of taps, and tap locations, and so on.
- Meter K Factor and Meter Factors with 5-point linearization curve
- Temperature, Pressure, and Density Correction for liquids
- Local Atmospheric (barometric) pressure
- Default process and operating parameters such as DP Threshold for flow cutoff, and so on.
- Metric or US units
- User-selectable units for totalizers and flow rates on a per channel basis
- Resettable and non-resettable totalizers for every meter channel.

6.1.5 Sealable Parameters

Sealable parameters are Site and Meter configuration options that directly affect measurement calculations. for example, orifice diameter, or K-factor.

Prover

- Process input alarm
- Prover classification
- Prover options
- Prover run counts
- Prover input format codes
- Prover size characteristics
- Prover reference conditions
- Prover parameter value
- Prover variation limits
- Prover process input scaling

Site

- Site options
- Site parameter value
- Arbitrary event-logged value

Meter

- Process input calibration / alarm
- Meter classification
- Reference conditions
- Meter options
- Input scaling
- Analysis component selection map
- Pulse input rollover
- Units
- Accumulator rollovers
- Meter parameter value
- Analysis precision, stream assignment
- Densitometer

Stream

- Stream options
- Stream parameter value
- Meter factor curve
- Analysis mole fraction, Ushort
- Analysis mole fraction, Float

6.1.6 Supported Meters

The following meter types have been used with the MVI69-AFC module. Because of the broad range of meters available in today's market, refer to the meter's specifications and the contents of this manual to evaluate the use of the AFC modules (even if the meter is listed here). If you have questions, please contact ProSoft Technology Technical Support Group.

Meter Type	Configured As (Differential or Linear)
Turbine	Linear
Orifice	Differential
V-Cone	Differential. You must configure the meter as V-Cone type in the AFC Manager (Meter Configuration / Calculation Options)
Wedge	Differential. Refer to Wedge Meter Applications (page 127) for information about using the wedge meters.
Vortex	Linear or Differential
Ultrasonic	Linear or Differential
Coriolis	Linear or Differential

Note: For Vortex, Ultrasonic or Coriolis meters, the selection depends on the output generated by the meter.

If the meter provides a pulse train representing the volume increment, the AFC meter should be configured as Linear with Primary Input selected as Pulse Count.

If the meter provides the instantaneous flow rate, then the AFC meter should be configured as Differential with Primary Input selected as Flow Rate.

Note: The module does not support applications to measure water, because the implemented standards are applicable to hydrocarbon fluids only.

6.1.7 Hardware Specifications

These modules are designed by ProSoft Technology and incorporate licensed technology from Schneider Electric (Modbus technology) and from Rockwell Automation (backplane technology).

	MVI46-AFC	MVI56-AFC	MVI69-AFC	MVI71-AFC	PTQ-AFC
Current Loads	800mA @ 5.1 VDC (from backplane)	800mA @ 5.1 VDC (from backplane)	800 mA @ 5V (from backplane) Power supply distance rating of 2	800 mA @ 5.1 VDC (from backplane)	800 mA @ 5V (from backplane)
Operating Temperature	0 to 60°C 32 to 140°F	0 to 60°C 32 to 140°F	0 to 60°C 32 to 140°F	0 to 60°C 32 to 140°F	0 to 60°C 32 to 140° F
Storage Temperature	-40 to 85°C -40 to 185°F	-40 to 85°C -40 to 185°F	-40 to 85°C -40 to 185°F	-40 to 85°C -40 to 185°F	-40 to 85°C -40 to 185°F
Relative Humidity	5% to 95% (non- condensing)	5% to 95% (non- condensing)	5% to 95% (non- condensing)	5 to 95 % (non- condensing)	5 to 95 % (non- condensing)
Modbus Port Connector	Three RJ45 connectors (RJ45 to DB-9 cable shipped with unit) supporting RS-232, RS-422, RS-485 interfaces	Three RJ45 connectors (RJ45 to DB-9 cable shipped with unit) supporting RS-232, RS-422, RS-485 interfaces	Two RJ45 connectors (RJ45 to DB-9 cable shipped with unit (supporting RS- 232, RS-422 and RS-485 interfaces (RJ45 to DB-9 cables shipped with unit.	Three RJ45 connectors (RJ45 to DB-9 cable shipped with unit), two of which support RS-232, RS-422, and RS- 485 interfaces.	Three DB-9M connectors, two of which support RS- 232, RS-422, and RS-485 interfaces.

6.2 Measurement Standards

The module supports the following hydrocarbon (gases and liquids) measurement standards currently used in the oil and gas measurement industry:

American Petroleum Institute (API) Manual of Petroleum Measurement Standards (MPMS)

- a. Density Correction to Reference Temperature
Chapter 11.1.53, 11.1.23
Equations, Tables 53, 23 - Generalized Crude Oils, Refined Products, Lubricating Oils, Special Applications

 - b. Correction of Volume to Reference Temperature and Thermal Expansion: Ctl.
Chapter 11.1.54, 11.1.24
Equations, Tables 54, 24 - Generalized Crude Oils, Refined Products, Lubricating Oils, Special Applications

 - c. Compressibility Factors for Hydrocarbons: Cpl.
Chapter 11.2.1/Chapter 11.2.2 (Chapter 11.2.1M and 11.2.2M for SI units.)

 - d. Orifice Metering of NGLs & Crude Oils
Chapter 14.3 (AGA3)

 - e. Calculation of Liquid Petroleum Quantities Measured by Turbine or Displacement Meters
Chapter 12.2

 - f. Allocation Measurement
Chapter 20.1 (high-water-content calculations used for emulsions)

 - g. Flow Measurement Using Electronic Metering Systems
Chapter 21.1, 21.2
-

American Gas Association (AGA)

- a. Orifice Metering of Natural Gas & Other Hydrocarbon Fluids
AGA Report No. 3 (1992) (MPMS Ch 14.3)

 - b. Compressibility Factors of Natural Gas and Other Related Hydrocarbon Gases
AGA Report No. 8 (1992) - Detail Characterization Method
-

International Standards Organization (ISO)

- a. Measurement of fluid flow by means of pressure differential devices inserted in circular cross-section conduits running full - Part 2: Orifice plates
ISO 5167-2 (2003)
-

Gas Processors Association (GPA)

- a. Temperature Correction for the Volume of Light Hydrocarbons - TP-25

 - b. A Simplified Vapor Pressure Completion for Commercial NGLs
GPA Document TP-15
-

6.2.1 Basic Metering According to Meter type

Orifice (Include V-cone): Uses AGA3 1992 / ISO 5167.

A V-cone meter is like an orifice meter, except that the V-cone is an obstruction in the center of the pipe while an orifice is an aperture. V-cone calculation differs from orifice calculation in the following respects:

- 1 The orifice Beta ratio is actually the square root of the ratio of aperture cross-section to pipe cross-section hence for the V-cone it is calculated differently from the two diameters.
- 2 The V-cone Coefficient of Discharge is entered as configuration and not calculated. Expansion Factor (Y) is calculated differently.

Output of the calculation is mass flow rate, which is divided by density to get volume and then integrated over time for accumulation.

Pulse: Both Gas and Liquid

Gross Volume is (pulses) / (K-factor) * (meter factor), according to API MPMS Ch 12 sec 2 1981 and 1995. Output of the standard calculation is volume flow increment, which is then multiplied by density to get mass increment. Flow rate is calculated in parallel to flow increment by applying to (pulse frequency) process input the same calculation as is applied to (pulses); this technique is employed instead of flow increment differentiation because the pulse frequency available from the counter card in the processor is not subject to variations of timing caused by scheduling delays in processor backplane transfer and in the firmware of the module, thus yielding a smoother flow rate.

Correction Factors According to Product Phase

Gas

Compressibility is calculated according to the Detail Characterization Method of AGA8 (1992). Gas density is a byproduct of this calculation. Essential input for this calculation is molar analysis. The compressibility Z is a factor in the gas equation $PV=ZNRT$, which is the rule by which gas volumes are corrected to reference conditions.

Liquid

Temperature and pressure correction factors are calculated according to API MPMS Ch 11 and applied according to the rules given in MPMS Ch 12. Essential input for this calculation is Liquid Product Overview (page 123) at either standard or flowing conditions.

Gas Pulse Measurement

The standard applied is AGA7, which is merely a combination of the Pulse: Both Gas and Liquid (page 114) and the gas law ($PV=ZNRT$) which includes compressibility. It also specifies calculation of some intermediate factors, which are now idiosyncratic and vestigial, having been imported from an earlier AGA3 (1985 and before) which used the "factor" method to calculate gas flow and which has been superseded by the completely overhauled 1990/1992 AGA3.

Water Content of Liquids

The handling of water content in crude and NGL products depends upon whether an "emulsion" Product Group is chosen.

For emulsions, water content is removed from the mixture according to the calculations of API MPMS Chapter 20.1 before calculating and applying correction factors. In this case the volumetric quantity intermediate between "Gross" and "Net" is "Gross Clean Oil", which is the hydrocarbon component of the mixture at flowing conditions. This method is recommended for mixtures containing more than 5% water.

For non-emulsions, water content is removed from the mixture according to the rules of API MPMS Chapter 12.2 after calculating and applying correction factors. In this case the volumetric quantity intermediate between "Gross" and "Net" is "Gross Standard", which is the entire mixture including its water content corrected to standard conditions under the assumption that it is pure hydrocarbon. Because the presence of water skews the correction calculations, this method should be used only when the water content is very low.

Non-Standard Reference Conditions

For both liquids and gases, the AFC permits a range of reference conditions for volume measurement which may vary from the API/AGA standard of 15°C/101.325kPaa (SI) or 60°F/14.696psia (US) (US pressure base for gases is 14.73psia). The allowed ranges for SI units are temperature between 0°C and 25°C and pressure between 50kPaa and 110kPaa, with the allowed ranges for US units approximately equivalent.

For gases, this flexibility of reference conditions is handled automatically by the implementation of the AGA 8 (1992) standard for compressibility Z together with the "real" gas law $PV=ZNRT$.

For liquids, correction factors for non-standard reference conditions are calculated differently depending on the firmware version. For version 2.05 and later, correction factors and corrected density are calculated according to the 2004 edition of API MPMS Chapter 11.1, except for the "NGL" product groups for which the CTL and density calculations of GPA TP-27 are extended with the CPL calculations of (old) MPMS Chapter 11.2 in a manner analogous to that of the 2004 Chapter 11.1. For version 2.04 and earlier, correction factors and corrected density are calculated as described in the following paragraphs, using the calculations of the 1980 edition of MPMS Chapter 11.1. In all cases, the density input to the calculations is the density at standard API base conditions.

Temperature Correction Factor, CTL

First, the "standard" factor, $CTL(\text{Flowing} / \text{ApiBase})$, is calculated, except that the final rounding step is not performed. Then, $CTL(\text{UserBase} / \text{ApiBase})$ is calculated, also unrounded. The $CTL(\text{Flowing} / \text{UserBase})$ is then calculated as $(CTL(\text{Flowing} / \text{ApiBase}) / CTL(\text{UserBase} / \text{ApiBase}))$, to which result is applied the final rounding step of the standard CTL calculation.

Pressure Correction Factor, CPL

The CPL(Flowing / UserBase) is calculated according to the method given in MPMS Ch 12.2 1995. In order to correct "density at reference" to User Base conditions, and also when iteratively calculating corrected density for the effect of elevated pressure, the CPL(Flowing / ApiBase) (unrounded) is also calculated according to the same method.

Density Correction

The density at API Base is determined according to relevant standards, which density is used as input to the CTL and CPL calculations. The density at User Base is determined by multiplying den(ApiBase) by the term $(CTL(UserBase / ApiBase) * CPL(Flowing / ApiBase) / CPL(Flowing / UserBase))$, all unrounded factors; this density is reported only and is not used in any calculations. When density correction is not selected, or an alarm causes a default to be assumed, any default "density at reference conditions" is deemed to be at User Base, and is also corrected to API Base for input to the CTL and CPL calculations.

Archiving and Event Log

- a) Accumulation and data recording for gas-phase archives conform to the requirements of API MPMS Ch 21 sec 1, 1993. Liquid-phase archives conform to API MPMS Ch 21 sec 2.

6.2.2 Liquid Correction Factor Details

For firmware version 2.05 and later, correction factors for most liquids are calculated according to the 2004 edition of API MPMS Chapter 11.1, enhanced with additional CPL calculations if required in order to allow selection of a non-standard base (reference) pressure. For lighter liquids (NGLs and LPGs), to which the 2004 Chapter 11.1 does not apply, the CTL and density correction calculations of GPA TP-27 are enhanced with the incorporation of the CPL calculations of MPMS Chapters 11.2.1 and 11.2.2 in a manner analogous to the method of the 2004 Chapter 11.1, to permit density correction to account for the effect of pressure and to yield the combined correction factor CTPL. For all liquids the option is available to use the vapor pressure correlation of GPA TP-15 June 1988.

For firmware version 2.04 and earlier, correction factors are calculated as described in the following paragraphs.

Temperature Correction Factor CTL

(According to Several "Tables" of MPMS Ch 11.1 (1980, except E Tables 1998 = GPA TP-25) and Other Standards)

Calculation of CTL (= VCF, Volume Correction Factor) from flowing temperature and density at standard temperature depends on the measurement system (SI or US), the product type (crude or refined), and the density range (high or low).

SI units:

$D \geq 610$ kg/m³ Table 54A (Crude&NGL) or 54B (Refined Products)

$500 \leq D < 610$ (LPG) ASTM-IP-API Petroleum Measurement Tables for Light Hydrocarbon Liquids 500-653 kg/m³ 1986 ISBN 0 471 90961 0

US units:

$D \geq 0.610$ RD60 Table 24A (Crude&NGL) or 24B (Refined Products),

$0.350 \leq D < 0.610$ (LPG) Table 24E - TP25

The low density range of 0.350 RD60 in US units is considerably lower than the 500 kg/m³ in SI units, because the E Tables are available only for US units.

Correction of density from flowing temperature to standard temperature is a converging iteration which includes the calculation of the VCF (Volume Correction Factor). Standards applied are those listed above except that Tables n3x are used instead of Tables n4x.

An option is available to iteratively correct the density calculation for elevated flowing pressure according to the condition given in bold type in MPMS Ch12.2 1995 Part 1 Appendix B Section B.1 (page 21).

Compressibility Factor *F*

(According to MPMS Ch 11.2 (US) or 11.2M (SI) 1986)

- Vapor pressure correlation according to GPA TP-15 June 1988.
- Pressure Correction Factor (CPL) is calculated from *F* and pressure above equilibrium according to MPMS ch12.2 1995, where "atmospheric pressure" is read as "base pressure" and "gage pressure" is read as "pressure above base". The module considers:

Pressure process input + barometric pressure = absolute pressure

6.3 Meter Channels

The number of available meter channels depends on the platform. The MVI56E-AFC supports 16 meters.

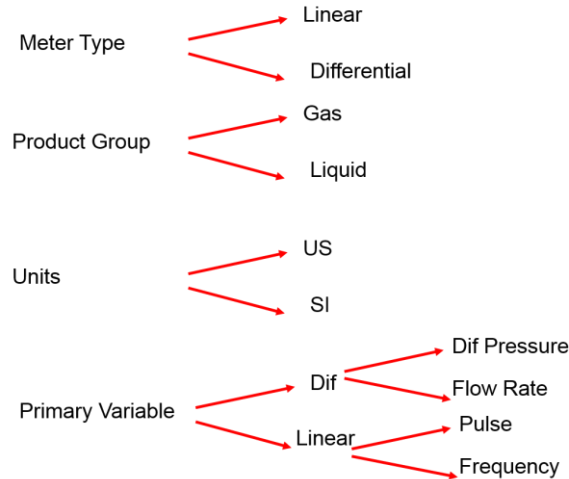
Each meter channel can be assigned as a linear meter (e.g a *pulse meter*) input or as a differential meter (e.g an *orifice meter*) input for flow measurement using either SI (metric) or US units. Selecting the differential meter causes the module to use the AGA 3 standard for flow calculation (for a gas orifice meter you may optionally choose ISO 5167-2 instead). Selecting the linear meter causes the module to use the AGA 7 standard for gas flow calculation.

Each meter channel can be configured for gas or liquid (*crude* or *refined*) product. The Product Group essentially selects the API/AGA Standards to be used in calculating flow rates/increments.

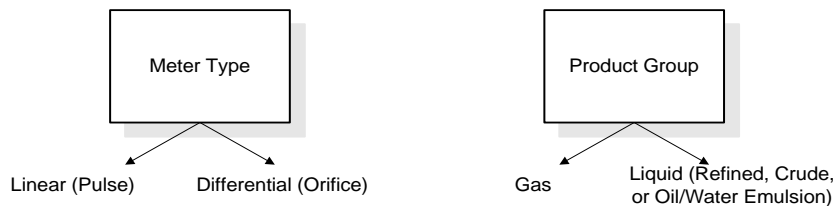
Selecting "Gas" causes use of AGA8 and either AGA3 or AGA7 Standards.

Selecting any liquid group causes use of API MPMS Chapter 11 (tables 23/24/53/54) and related Standards. "Crude oils, JP4" and "Oil-water emulsion" use the base, "A", tables. "Refined Products" use the "B" tables. "Lubricating oils" use the "D" tables, and "Special applications" use the "C" tables. "Crude oils, JP4" and "NGLs/LPG" are used for propane, butane, NGLs (natural gas liquids), and crude oils which are relatively water-free (less than 5 percent). The two "Oil-water emulsion" groups are used for crude and NGL/LPG that might have a high concentration of water for which API MPMS Chapter 20.1 is applicable. "Refined products" (xJP4) is used for lighter refined liquids such as gasolines, jet fuels (except JP4), and fuel oils. "Lubricating oils" is used for heavier refined liquids. "Special applications" is used for those liquids that cannot reasonably be assigned to one of the other groups; for this product group an explicit coefficient of thermal expansion must be supplied.

The following table provides a brief overview of the standards used according to the Meter Type and Product Group:



Meter Type	Product Group	Standards
Differential	Gas	AGA8, AGA3/ISO5167
Differential	Liquid	MPMS ch 11 AGA3/ISO5167
Linear	Gas	AGA8, AGA7
Linear	Liquid	MPMS ch 11, MPMS ch12.2



Note: The meter channel must be disabled in order to change its meter type and product group.

6.3.1 General Features

Process Variable Interface

Process variables for each of the meter channels must be produced by the controller for consumption by the AFC module. A versatile architecture for backplane transfer of process variables and other data and signals allow you to easily implement the data transfer. The sample ladder logic automatically transfers the process variables to the module and reads the calculation results to the processor.

Meter Scan Time

For good measurement, the process I/O must be sampled, and the flow calculations completed quickly in order to avoid losing process information and measurement accuracy. The process I/O scan time for the module is under one second for all meter runs.

Note: This is time-dependent on design of the ladder logic implemented to support the two-way data transfer between the AFC module and the controller. The meter calculation scan independent of the process I/O scan may take longer.

Multiple Meter Accumulators

Each meter channel supports the following set of full 32-bit accumulators that may be configured in binary or split decimal format with user-defined rollover values:

- Gross Volume
- Gross Standard Volume (liquid only)
- Net Volume
- Mass
- Water (liquid only)
- Energy (gas only)

Access to the above accumulators is available directly from the Modbus Slave communications ports.

Product Batching

Any or all of the available meter runs may be configured for field installation that requires shipping and/or receiving product batches of predetermined size. The configuration utility option of selecting resettable accumulators provides a simple way to use the power of ladder logic to design product batching, monitoring, and control tailored to suit specific field requirements.

The Meter Signals feature can be used to create an archive or reset an accumulator after the batch is concluded. Refer to the Ladder Logic section for your platform for more information on using this feature.

Data Archiving

The module supports the archiving of data for each meter channel. Each time, one record consisting of all the associated data is date and time stamped and archived. This option allows for archiving each hour for 2 days (48 records per meter run) and every day for 35 days (35 daily records per meter run) for each meter channel. Each record consists of up to 40 process and other variables. Archives are mapped to the local Modbus Table. Refer to Archives for more information about this topic.

Event Log Function

The module can log up to 1999 critical events in an Event Log File stored as a set of easily accessible Modbus registers in non-volatile RAM. Changing critical parameters, such as orifice plate size, Meter Base K factors, and Meter Correction Factors, are time stamped and logged. Refer to Events for more information about this topic.

Measurement Units

This option is provided for each meter channel to be configured with SI or US units of measurement. Units for flow totalization (*volumetric* and *mass*) and flow rate monitoring are configurable for each meter channel separately if the default configuration is not applicable. Each meter channel may be configured to use any of the standard units from liters/gallons to thousand cubic meters/barrels. The flow rate period of each meter channel may be selected from flow rate per second, per minute, per hour, or per day.

Process Input Scaling

The module allows you to either pre-scale the process inputs via ladder logic for use in the measurement calculations, or provide unscaled values from the analog input modules directly. In the second case, the scaling is done internally. You can directly enter the zero-scale, the full-scale, and the default values for each of the process variable inputs through the configuration window. Pre-scaled values may be transferred as floating point or as scaled integer.

When selecting scaled integer, the integer values are scaled as follows:

Scaled Integer		
Variable	Format	Example
Temperature	Two decimal places implied	A value of 1342 would be equivalent to 13.42°C
Pressure	No decimal places implied for SI units (kPa) and one decimal place implied for U.S. units (psi).	A value of 200 would be equivalent to 200kPag
Differential Pressure	Two decimal places implied for inches of H2O and 3 places for kPa	A value of 35142 would be equivalent to 35.142kPa
Density (kg/m3)	One implied decimal place	A value of 5137 would be equivalent to 513.7 kg/m3
Density (Relative Density)	Four implied decimal places	A value of 10023 would be equivalent to 1.0023 60F/60F.
Density (API)	Two implied decimal places	A value of 8045 would be equivalent to 80.45 °API.

When selecting the 4 to 20mA process input scaling, the module uses the following ranges:

4 to 20mA			
Processor	Module	0%	100%
SLC	MVI46-AFC	3277	16384
ControlLogix	MVI56-AFC	13107	65535
CompactLogix	MVI69-AFC	6241	31206
PLC	MVI71-AFC	819	4095
Quantum	PTQ-AFC	4000	20000

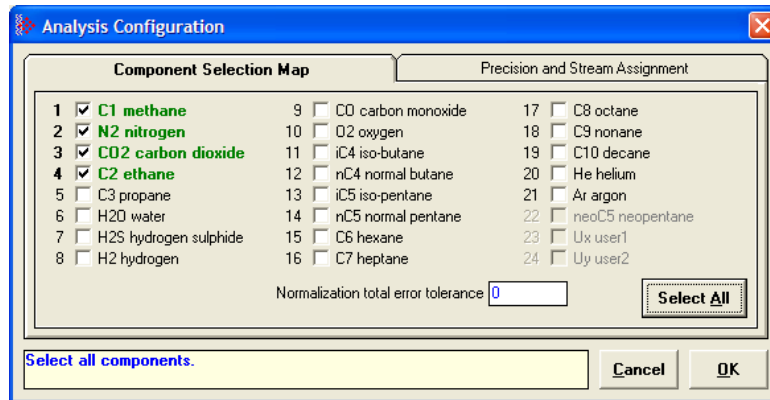
The module uses the configured values for zero and full scale to interpret the process input scaling.

In the **Meter Monitor** window, the raw values as transferred from the processor are shown at the "Last Raw" column and the converted values are shown at the "Scaled Avg" column.

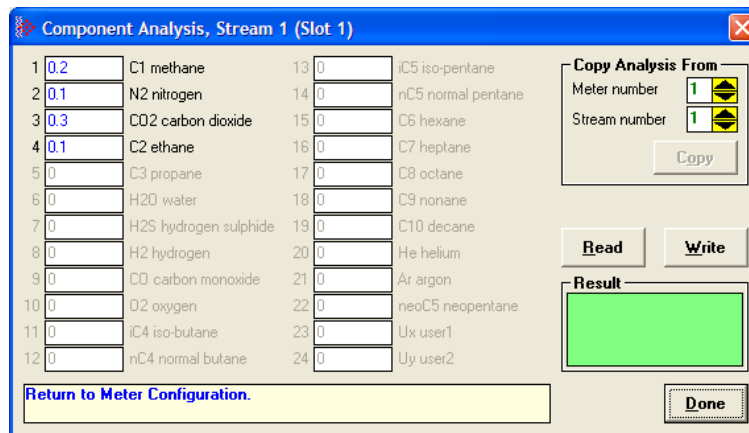
6.3.2 Gas Product Overview

The gas compressibility calculations are based on molar analysis concentrations of up to 21 components, using the Detail Characterization Method of AGA8 (1992). The module automatically generates alarms if the sum of the molar concentrations is not 100%

Configure the analysis settings using the AFC Manager (**Meter Configuration / Analysis Config**) as follows. This window allows the selection of the components(Component Selection Map) and analysis precision (Precision and Stream Assignment – version 2.06.000 or higher). The sample ladder logic assumes that all components are selected so check all components at the Component Selection Map window.



Enter the gas analysis concentrations by clicking the Analysis button. You can also update the concentrations through the backplane as will be later shown in this User Manual.



The module records events every time a molar concentration value changes. For applications that involve gas chromatograph devices, this feature might not be desirable because it is expected that the values should frequently change. You can disable this feature using AFC Manager (**Meter Configuration / Control Options / Treat Analysis as Process Input**).

6.3.3 Liquid Product Overview

The module supports applications involving crude or refined oil such as crude oil, oil/water emulsion, propane, butane, NGLs, LPGs, gasoline, jet fuels and lubricating oils.

When measuring liquids with density correction, density at flowing conditions is required. This value may be provided directly as a process input, or the module can calculate a density from the frequency provided by a densitometer device.

To use a densitometer

Follow the steps below to use a densitometer.

- 1 Configure it, entering all configuration parameters directly from the calibration data sheet supplied by the densitometer manufacturer.
- 2 Supply the frequency output from the densitometer in Hz as a floating-point value in the "Flowing density" process-input location over the backplane (refer to the Backplane Communication section for your platform in the MVI69-AFC manual to determine the correct location). The AFC then calculates a flowing density value, which is then validated by the range check mandated by the "Density" values of "Process Input Scaling" of the meter configuration. The "Scaling" sub-selection is not used against the frequency input, however; the frequency is always input as floating-point.

Note: If using the Densitometer feature, select the Density Process Input Scaling for 4 to 20mA and enter the densitometer frequency as a floating-point value.

Module Configuration

The Configuration and Debug menu for this module is arranged as a tree structure, with the Main Menu at the top of the tree, and one or more sub-menus for each menu command. The first menu you see when you connect to the module is the Main menu.

Because this is a text-based menu system, you enter commands by typing the command letter from your computer keyboard in the diagnostic window in ProSoft Configuration Builder (PCB). The module does not respond to mouse movements or clicks. The command executes as soon as you press the **[COMMAND LETTER]** — you do not need to press **[ENTER]**. When you type a command letter, a new screen is displayed in the PCB window.

Density Units

The liquid density units can be expressed as:

- Density is in kg/m³
- Relative density 60°F/60°F
- API gravity

Measuring Water Diluent

For liquid measurement applications, the optional automatic calculation of Net Oil Volume and mass based on the Sediment and Water (S&W) percent input is supported. Only provide the S&W percent value in the specified controller register. The module puts the gross standard (or gross clean oil), net oil and water accumulations in separate accumulators. Refer to Net Accumulator Calculation (page 147).

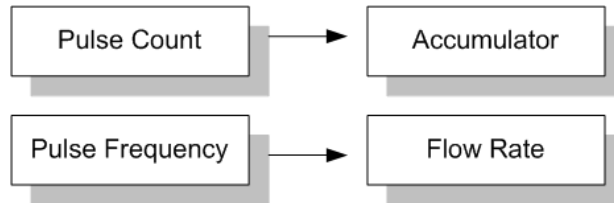
6.3.4 Linear (Pulse) Meter Overview

The module typically receives the pulse count and pulse frequency values from a high-speed counter module. The module uses these values to perform calculations.

You can configure the primary input to be used for volume calculation. You can configure it as Pulse Count or Pulse Frequency.

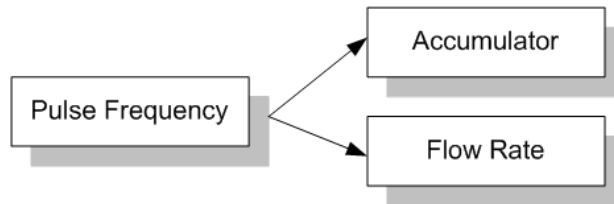
Primary Input = Pulse Count

If you select Pulse Count as the primary input, the module uses the pulse count value transferred through the backplane as the primary input for volume calculation. In this case, the pulse frequency will be used for flow rate calculation only.



Primary Input = Pulse Frequency

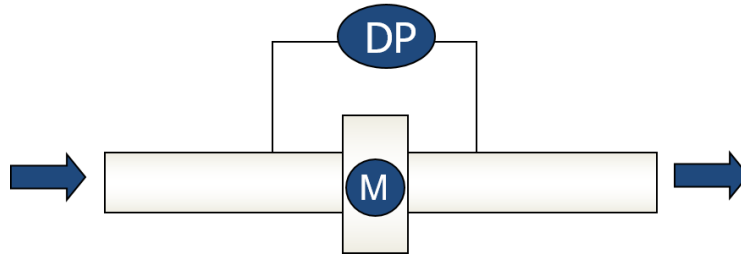
If you select Pulse Frequency as the primary input, the module uses the pulse frequency value transferred through the backplane as the primary input for both flow accumulation and flow rate calculation. The pulse count value is ignored by the module.



6.3.5 Differential (Orifice) Meter Overview

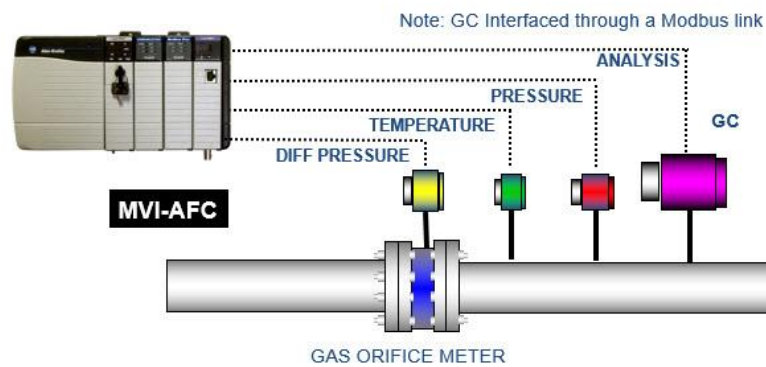
The static pressure of the gas stream can be measured either upstream of the meter (before the differential pressure drop), or downstream of the meter (after the pressure drop). Both AGA3 and AGA8 require the upstream static pressure for their calculations, where:

$$\text{upstream pressure} = \text{downstream pressure} + \text{differential pressure}$$

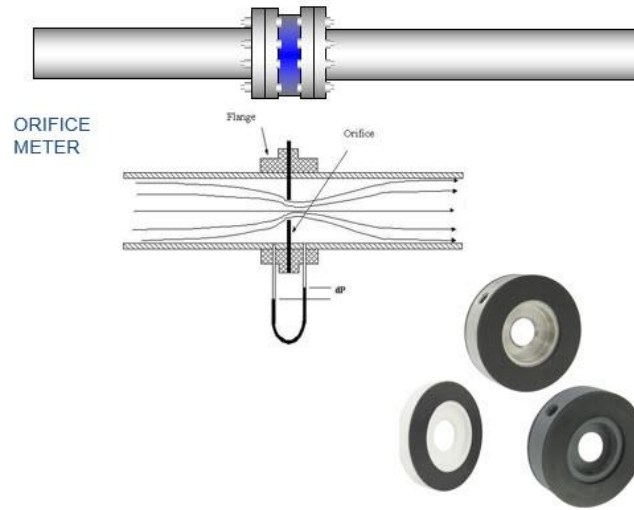


If the pressure is measured from a downstream tap (typical), the *Downstream Static Pressure* option should be set through the AFC Manager.

AGA Application with Gas



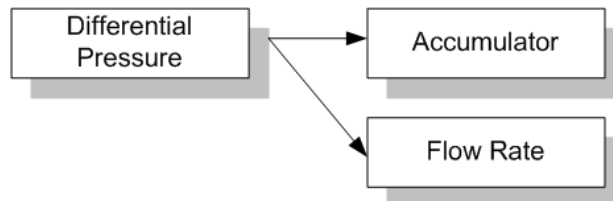
Oriface Plate



The module also supports the V-Cone device. You can configure V-Cone meters and downstream selections in AFC Manager, on the **Meter Configuration / Calculation Options** dialog box.

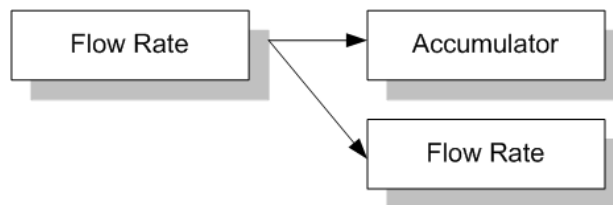
Primary Input = Differential Pressure

The primary input parameter configures the value used as source for the accumulator calculation. If the parameter is set to Differential Pressure, the module uses the differential pressure value transferred through the backplane for accumulator calculation.



Primary Input = Flow Rate

You can configure the primary input parameter as flow rate in order to use this value for the accumulator calculation.



Note: The flow rate can be converted to a different unit.

The AFC Manager software supports the following parameters:

- Orifice Plate and Meter Tube Measured Diameter
- Orifice Plate and Meter Tube Measurement Temperature
- Orifice Plate and Meter Tube, Coefficient of Thermal Expansion
- DP Flow Threshold (kPa)
- DP Alarm Threshold (kPa)

6.3.6 Wedge Meter Applications

For Wedge Meter applications you must convert some parameters from the meter manufacturer's data sheet before entering these values to the AFC Manager. The following spreadsheets can be used to calculate the AFC Manager parameters according to the meter manufacturer as follows:

Filename	Application
WEDGE_ABB.xls	ABB Wedge Meter
WEDGE_PRESO.xls	PRESO Wedge Meter

You must initially configure the meter as a differential type. Then you must configure it as a V-Cone Device (**Meter Configuration / Calculation Options**).

Refer to the spreadsheet for instructions on how to enter the correct values into AFC Manager.

6.4 Modbus Database

The module supports two individual Modbus slaves (Primary and Virtual) to optimize the polling of data from the remote SCADA system, or from the processor (through the backplane). Refer to the Modbus Dictionary dialog box in AFC Manager for information about Modbus addressing.

6.4.1 AFC Modbus Address Space

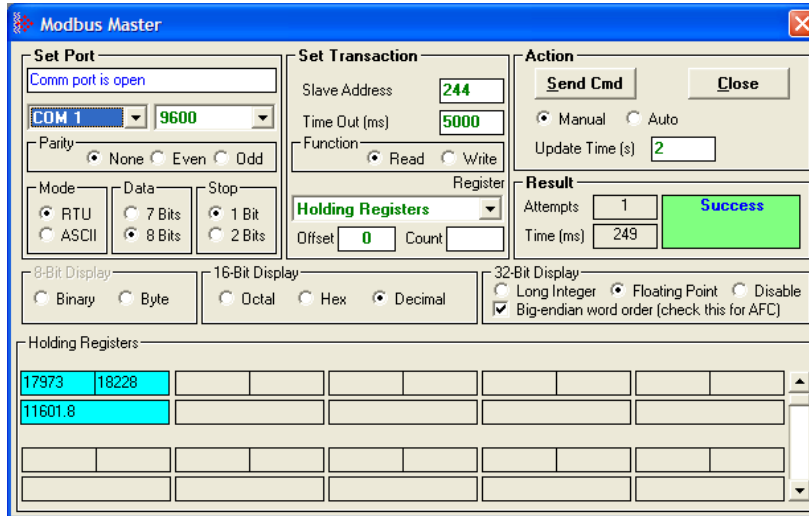
Addressable Modbus registers are divided into four banks as shown in the following table.

MODBUS Address Space Allocation: Total Modbus Registers: 131,072			
Primary Slave Banks (131072 registers)		Virtual Slave Banks (20,000 registers)	
Holding Registers	Input Registers	Holding Registers	Input Registers
From: 0	From: 0	From: 0	From: 0
To: 65535	To: 65535	To: 9999	To: 9999

The first 100 registers of the virtual slave (registers 0 through 99) are predefined to map to the first 100 registers of the primary slave. This mapping cannot be changed. Also, the Virtual Slave Input Registers can be accessed as Virtual Slave Holding Registers by adding 10000 to the Modbus register address; for example, Input Register 2386 is the same as Holding Register 12386.

Accessing the Data

The AFC Manager provides an easy way to read and write data from both slaves through the Modbus Master Interface.



6.4.2 Primary Slave

The Primary Slave contains the main AFC database that consists of 131,072 Modbus registers. The Site and Meter configuration, as well as all live process data and ongoing calculations are kept in the Primary Slave address space. This address space is divided equally between the Input Register Bank (65,536 registers) and the Holding Register Bank (65,536).

The register addressing is shown in the Modbus Dictionary dialog box in AFC Manager.

Modbus Address References

In these documents (the AFC Manager User’s Guide and the User’s Guide for your platform) you will occasionally see Modbus address references like *Ph00018* or *Mh00162*. The first two characters of such references indicate how to convert the following number into an absolute Modbus address in the module.

This table shows the possible values for the first identification character:

Address Translation ID	Description
P	Absolute Modbus address, Primary Slave
M	Meter-relative Modbus address, Primary Slave
V	Absolute Modbus address, Virtual Slave

This table shows the possible values for the second identification character:

Register Bank ID	Description
h	Holding register
i	Input register

Modbus Address Examples

Ph02000 = holding register located at address 2000 in the primary slave

Pi02000 = input register located at address 2000 in the primary slave

Mh00100 = Meter-relative holding register located at offset 100 in the block of the primary slave that contains the data for the meter

Meter-relative Data

Meter-relative data starts at absolute holding register address 8000 and occupies 2000 words of data for each meter channel.

	8000
Meter 1 Data	
	10000
Meter 2 Data	
	12000
Meter 3 Data	
	14000
Meter 4 Data	
	16000
Meter 5 Data	
	18000
Meter 6 Data	
	20000
Meter 7 Data	
	22000
Meter 8 Data	
	24000

The meter-relative addresses are offsets within each meter data area. The correct absolute address is calculated by the following formula (assumes meters are numbered starting with 1):

$$(\text{absolute address}) = (2000 * (\text{meter number}-1)) + 8000 + (\text{meter relative address})$$

In the Modbus Dictionary dialog box, addresses listed for the selected meter are absolute addresses, so you should subtract the appropriate multiple of 8000 to calculate the meter-relative address.

Example: Find the orifice diameter address for the first 5 meter channels.

The meter 1 orifice diameter registers are located at the holding register address 8162 and 8163 as follows:

8160	8161	Float	Parameter: orifice plate: measurement temperature
8162	8163	Float	Parameter: orifice plate: measured diameter
8164	8165	Float	Parameter: orifice plate: coef of thermal expansion
8166	8167	Float	Parameter: meter tube: measurement temperature
8168	8169	Float	Parameter: meter tube: measured diameter
8170	8171	Float	Parameter: meter tube: coef of thermal expansion
8172	8173	Float	Parameter: differential pressure flow threshold

The meter-relative addresses are Mh00162 and Mh00163

The addresses for meters 1 to 5 are listed on the following table.

Meter	Registers
1	8162 and 8163
2	10162 and 10163
3	12162 and 12163
4	14162 and 14163
5	16162 and 16163

Scratchpad

The Primary Modbus Slave contains a scratchpad area that can be used to store any data required by each application. This area is "empty" by default and contains 6000 words of data starting at holding register 2000 in the Primary Modbus Slave.

6.4.3 Virtual Slave

The module also provides a Virtual Address Space of 20,000 Modbus registers. This address space is divided equally between the Input Register Bank (10,000 registers) and the Holding Register Bank Holding Register Bank (10,000). This is where you can create a virtual re-map by cross-referencing any of the 130,072 Primary Slave Modbus registers to the 20,000 Modbus registers in the Virtual Slave Banks, thereby making it easy for a SCADA Master to poll only the necessary Modbus addresses in contiguous blocks. The virtual slave can also be used for data polling from the processor through the backplane.

Modbus access to the Virtual Modbus Slave is disabled by default since its Modbus address is originally set as 0. To use the Virtual Modbus Slave, you must initially configure a Modbus address greater than zero in order to enable it. Refer to Site Configuration for more information about enabling the Virtual Slave and using the remapping feature. The PLC may always access the Virtual Slave, whether or not it has a non-zero slave address and thus is available via Modbus.

A download operation will not transfer the Virtual Slave Remapping configuration. You must click on the **Write** button on the **Indirect Address Remapping** dialog box to transfer the data.

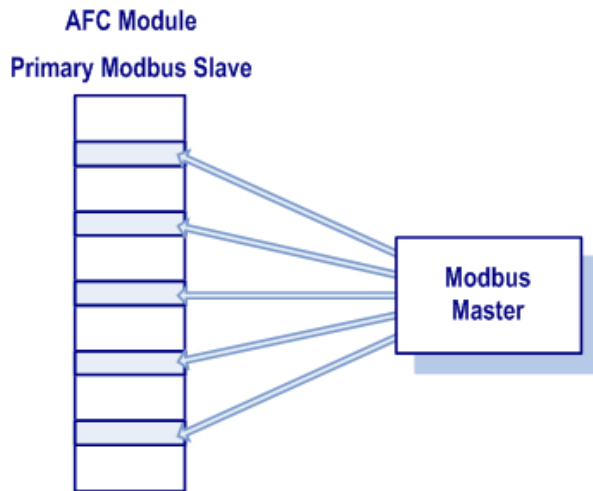
Note: The first 100 registers in the Virtual Slave Holding Register Bank have been pre-assigned and cannot be remapped. They map directly to the first 100 holding registers of the Primary Slave.

Virtual Slave Example Application

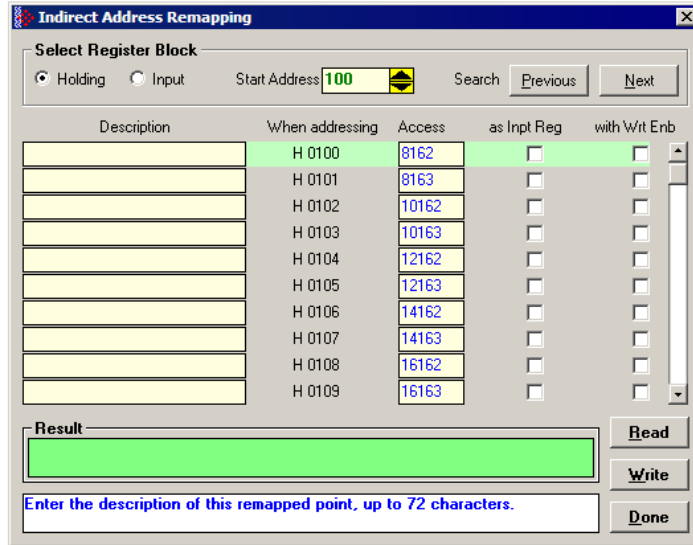
Assume that an application requires a remote Modbus Master to poll the orifice diameters for the first 5 channels. Continuing the previous example, the holding register addresses are listed again the following table.

Meter	Registers
1	8162 and 8163
2	10162 and 10163
3	12162 and 12163
4	14162 and 14163
5	16162 and 16163

Because these addresses are not contiguous, the Modbus Master would have to use five commands to poll all the data directly from the Primary Modbus Slave as follows:



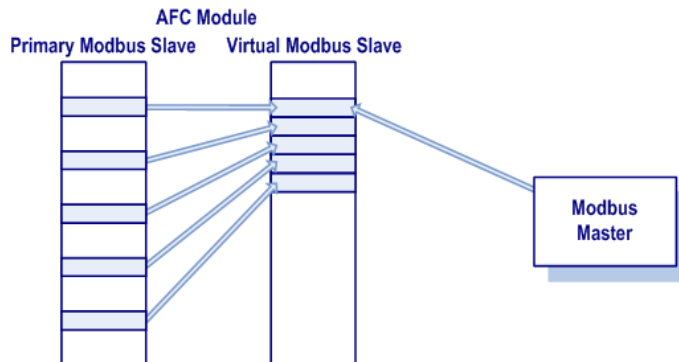
However, using the Virtual Modbus Slave optimizes the polling of data because the registers can be remapped in any order using the AFC Manager (Site Configuration window). The following illustration shows how the orifice diameter registers could be remapped to the Virtual Slave starting at address Vh00100:



The following table shows how the addresses would be remapped between both slaves:

Primary Modbus Slave Addresses	Virtual Modbus Slave Addresses
8162 and 8163	100 and 101
10162 and 10163	102 and 103
12162 and 12163	104 and 105
14162 and 14163	106 and 107
16162 and 16163	108 and 109

Therefore, instead of sending five Modbus commands (2 words each) to the Primary Modbus Slave, the Modbus Master device can now send one single Modbus command (10 words) to the Virtual Modbus Slave in order to poll the same data from the module:



This example demonstrates the benefits of using the Virtual Slave instead of accessing the data directly from the Primary Modbus Slave. The same procedure can be used when polling data from the processor (through the backplane) because the Modbus Gateway block also requires the data to be listed in a contiguous order.

6.4.4 Modbus Addressing Common to Both Primary and Virtual Slaves

Address	Type	Description
Ch00000	Char	Firmware product code, group Low byte: platform High byte: application class
Ch00001	Char	Firmware product code, item Low byte: number of streams High byte: number of meters
Ch00002	Int	Firmware version number Low byte: minor version number High byte: major version number
Ch00003	Int	Firmware revision number
Ch00004 to Ch00005	Int	Serial number

Address	Type	Description
Ch00006	Bm	<p>Site status</p> <p>bit 0 - AFC released</p> <p>Latched when both bit 15 (cold start) and bit 12 (Processor offline) first become clear, remaining so until any subsequent cold start. While this bit remains clear events are not logged, allowing an initial configuration to be fully completed without filling up the event log.</p> <p>bit 1 - Checksum alarm</p> <p>Set when any bit in the "Checksum Alarms" registers, for site and each meter, is set; clear when all such bits are clear.</p> <p>bit 2 - [reserved]</p> <p>bit 3 - [reserved]</p> <p>bit 4 - Processor halted, offline, or missing</p> <p>Set while backplane communication is faulty, which typically occurs when the Processor is switched to program mode. While set, measurement continues using the latest process input values obtained from the processor. Upon resumption of backplane communication, the AFC compensates for the downtime by computing an accumulator increment in a manner that depends on the meter type. For differential (orifice) meters, the first measurement scan acquires a scan period equal to the period of downtime as computed from the system timer, hence periods of processor downtime shorter than the rollover period of the system timer cause no loss of product. For linear (pulse) meters, the first measurement scan acquires a pulse increment equal to the difference between the processor-supplied pulse count of the current scan and that of the last scan before communication loss, hence periods of processor downtime shorter than the rollover period of the counter module cause no loss of product.</p> <p>bit 5 - Measurement configuration changed</p> <p>Set when any bit in the "Measurement Configuration Changed" registers is set; clear when all such bits are clear.</p> <p>bit 6 - Power up</p> <p>Set upon power-up, and cleared upon setting the wallclock for the first time..</p> <p>bit 7 - Cold start</p> <p>Upon power-up, AFC's non-volatile storage is checked for validity, by verifying a checksum and confirming that certain known values are present in their proper locations. If the storage is invalid, then it is initialized with a default configuration, and this bit is set. The bit remains set, even through subsequent power cycles, until at least one meter is enabled at which time the bit is cleared.</p> <p>bit 8 - A copy of the "Hard Passwords" site option, made available here so that an external application such as AFC Manager can learn all it needs to know in order to connect to the module by reading the first 20 holding registers from the Modbus table.</p> <p>bit 9 - [reserved]</p> <p>bit 10 - [reserved]</p> <p>bit 11 - [reserved]</p> <p>bit 12 - [reserved]</p> <p>bit 13 - [reserved]</p> <p>bit 14 - [reserved]</p> <p>bit 15 - [reserved]</p>
Ch00007	By	Processor offline code: 0 online, 1 offline
Ch00008	By	<p>Zero / primary slave address</p> <p>This value distinguishes the two slaves. When read from the primary slave this value is zero; when read from the virtual slave this value is the primary slave address.</p>
Ch00009	Wd	Password, write-enable

Address	Type	Description
Ch00010 to Ch00015	Wd	Wallclock (Y,M,D,h,m,s) The wallclock has a resolution of 1 second.
Ch00016 to Ch00017	Bm	Wallclock (packed) The packed wallclock has a resolution of 2 seconds.
Ch00018	Bm	accessed port and authorization bits 0- 3 - Accessed port; 0 = gateway bit 4 - Password authorization waived for read bit 5 - Password authorization waived for write bit 6 - Password authorization granted for read bit 7 - Password authorization granted for write
Ch00019	Wd	Password, read-enable
Ch00020 to Ch00089	--	[reserved] Reserved for use by diagnostic and similar procedures.
Ch00090 to Ch00099	Wd	Arbitrary event-logged registers. A Modbus master (such as the processor using Modbus Gateway) can use these to record in the Event Log changes to values unrelated to flow measurement.

6.4.5 Modbus Port configuration

Configuration of the serial ports is stored in these blocks of the Modbus table:

Address	Type	Description
Ph00102 to Ph00105	Bm	Port 1 configuration
Ph00106 to Ph00109	Bm	Port 2 configuration
Ph00110 to Ph00113	Bm	Port 3 configuration

Each group of registers specifies configuration of the corresponding serial port. The four registers of each block are interpreted as follows:

Ofs	Type	Tag	Contents
+0	Bm	Uart	UART parameters and port options
+1.L	By	TmoC	LSB: Timeout for CTS
+1.H	By	TmoR	MSB: Master mode receive timeout
+2	By	Dly1	Delay before first data after CTS
+3	By	Dly0	Delay after last data before ~RTS

The CTS timeout and both delays are in units of 5ms (200Hz system clock), with valid values from 0 thru 255, and are significant only for transmission of outgoing Modbus messages. The receive timeout is in units of 0.1 second, with valid values from 0 thru 255 (where 0 implies the default of 5, that is, one-half second), and is significant only for the last port when configured as a Modbus master. The UART parameters and port options word is a bitmap:

Bit	Parameter	Value
bits 0 to 2	Baud	000: none; see below 001: 300 baud 010: 600 baud 011: 1200 baud 100: 2400 baud 101: 4800 baud 110: 9600 baud 111: 19200 baud
bits 3 to 4	Parity	00: no parity 01: odd parity 10: even parity 11: no parity (should not be used)
bit 5	Data bits	0: 8 data bits 1: 7 data bits
bit 6	Stop bits	0: 1 stop bit 1: 2 stop bits
bit 7	Modbus mode	0: RTU mode 1: ASCII mode
bit 8	Modbus orientation	0: slave 1: master (permitted only for last port)
bit 9	Primary slave accessibility (not meaningful for master port)	0: primary slave accessible through this port 1: primary slave not accessible (not permitted for Port 1)
bit 10		Swap Modbus bytes
bit 11		Swap Modbus words
bit 12		Disable pass-thru (not meaningful for master port)
bits 13 to 15		[reserved]

A change in configuration takes effect after transmission of the response to the Modbus command that causes the change; the response is sent using the old configuration, but subsequent Modbus commands to the reconfigured port must use the new one. Writing a baud code of 0 means that the current configuration is not to be changed, and all other items are ignored. Default values are 6 for the bitmap (9600,N,8,1,RTU,slave,primary,noswap,passthru) and 0 for the timeout and both delays. The message transmission procedure is:

- Raise RTS.
- If TmoC is zero ignore CTS, else wait up to TmoC clock ticks for CTS.
- Delay for Dly1 clock ticks.
- Transmit message.
- Delay for Dly0 clock ticks.
- Drop RTS.

6.5 Modbus Communication

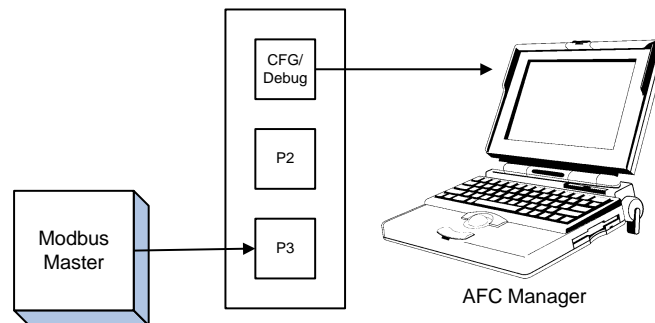
A remote Modbus Master device can be connected to any one of the communication ports for data polling. The module accepts the following Modbus command functions according to the Modbus protocol specification:

Modbus Function Code	Description
3	Read Holding Registers
4	Read Input Registers
6	Preset (Write) Single Register
16	Preset (Write) Multiple Registers

Ports 2 and 3 support RS-232, RS-422, or RS-485 communications. The Configuration/Debug port (Port 1) supports RS-232 only.

Refer to Cable Connections (page 196) for wiring instructions.

The Modbus Master command can be sent to either the Primary or Virtual Modbus Slaves in the module. Each slave has individual Modbus addresses that you can configure (**Project / Site Configuration**). The Primary Slave address is configured as 244 by default.



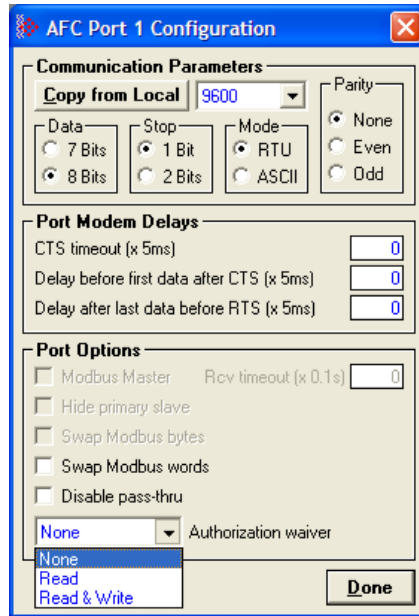
6.5.1 Communication Parameters

The module supports the following communication parameters for each communication port:

Parameter	Values
Baud Rate	300, 600, 1200, 2400, 4800, 9600 or 19200
Data Bits	7 or 8
Stop Bits	1 or 2 Bits
Mode	RTU or ASCII
Parity	None, Even or Odd

Note: Do not configure a port for both RTU mode and 7 data bits as this combination is not supported by the Modbus protocol.

You must configure the communication parameters for each communication port using the AFC Manager software (Site Configuration):



6.5.2 Port Options

The following options can be configured:

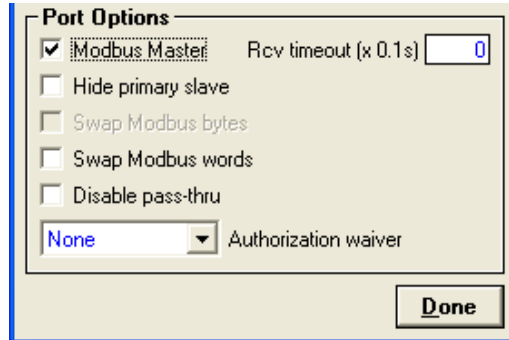
Port Options	Description
Hide Primary Slave	Protects the Primary Slave from any read or write operation from a remote master. Only the virtual slave is visible on this port.
Swap Modbus Bytes	Swap the Modbus bytes transferred through this port (Not implemented)
Swap Modbus Words	Swap the Modbus words transferred through this port. This parameter is only applicable to those data points that hold 32-bit quantities (long integers, floats, totalizers),
Disable Pass-Thru	Disables the pass-thru feature on this port
Modbus Master	Enables the Modbus Master for the port (Port 3 only)
Authorization waiver	Each port can be individually configured to waive the authorization requirement. This feature allows each port to have a different access level.

Not all options are available on every port:

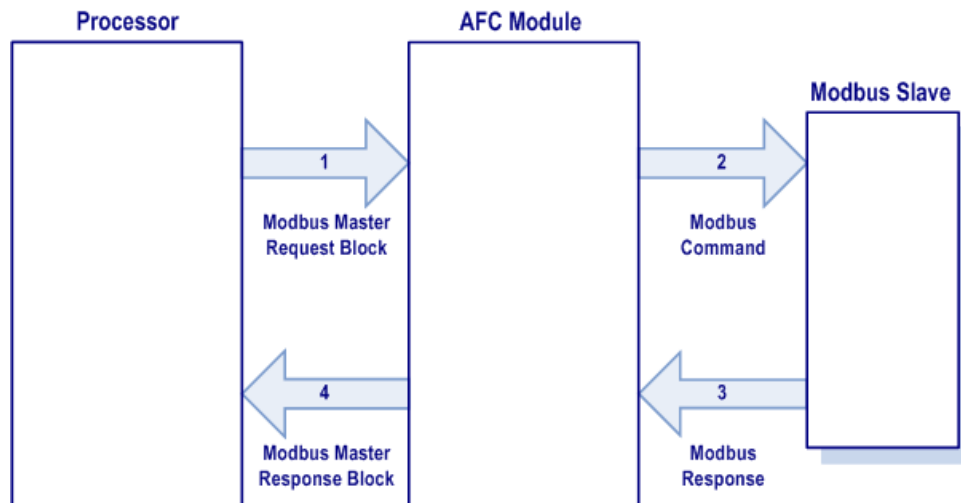
- Port 1 is restricted, so that AFC Manager can always communicate with the Primary Slave using this port.
- Modbus Master option is available only on Port 3.

6.5.3 Modbus Master

Port 3 can be configured for Modbus Master operation (**Project / Site Configuration / Port 3**).



The Modbus Master command is generated from the processor using ladder logic (Modbus master block). After the Modbus Master transaction is completed the module is ready to receive another Modbus Master request from the ladder logic:



The following Modbus functions are supported for Modbus Master operation:

Modbus Function Code	Description
1	Read Coil Status
2	Read Input Status
3	Read Holding Registers
4	Read Input Registers
15	Force (Write) Multiple Coils
16	Preset (Write) Multiple Registers

The module offers considerable flexibility for Modbus Master operation, allowing the ladder logic to select one of the following data types:

- Bit (packed 16 to a word)
- Word (16-bit register)
- Long (32-bit items as register pairs)
- Long Remote (32-bit items as single registers)

Note: Long data type implements each data unit as one pair of 16-bit registers (words). Each register contains two bytes. Long remote data type implements each data unit as one 32-bit register. Each register contains four bytes. The proper choice depends on the remote slave's Modbus implementation.

Example

The following table shows how the data types are implemented if a **write** function is selected and the item count is configured with a value of 10 (decimal):

Data Type	Register Type	Modbus Function	Number of Coils	Number of Bytes	Number of Registers	Number of words (16-bits) transferred
Bit	Coil	15	10	2	-	1
Word	Holding	16	-	20	10	10
Long	Holding	16	-	40	20	20
Long Remote	Holding	16	-	40	10	20

Note: The number of coils, bytes, and registers are part of the Modbus request (functions 15 and 16) according to the Modbus specification.

The following table shows how the data types are implemented if a **read** function is selected and the item count is configured with a value of 10 (decimal):

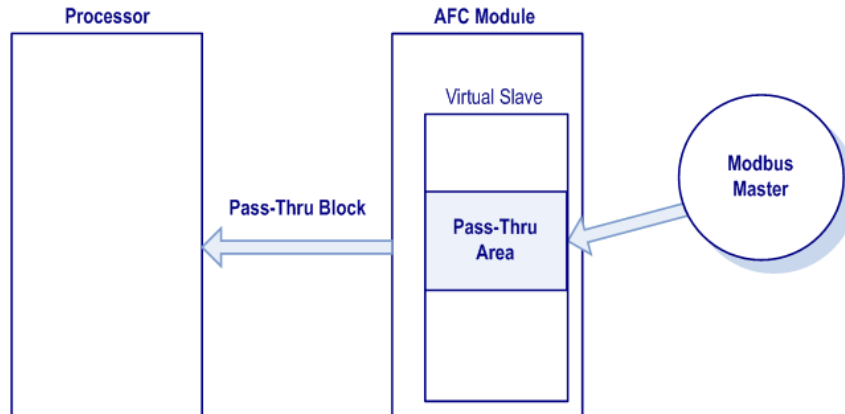
Data Type	Register Type	Modbus Function	Number of Registers
Bit	Coil	1	10
Bit	Input	2	10
Word	Holding	3	10
Word	Input	4	10
Long	Holding	3	20
Long	Input	4	20
Long Remote	Holding	3	10
Long Remote	Input	4	10

Note: The number of registers is part of the Modbus request according to the Modbus specification.

Refer to the ladder logic section for your module for more information about the Modbus Master block.

6.5.4 Modbus Pass-Through

The Modbus pass-through feature allows you to configure a Modbus pass-through region in the Virtual Slave (**Project / Site Configuration**). After the module receives a holding register write command (Modbus functions 6 or 16) or a bit write command (Modbus functions 5 or 15) to this region, it will generate a pass-through block to be sent to the processor containing the Modbus command data. You may define a word pass-through region (for Modbus functions 6 and 16) and a bit pass-through region (for Modbus functions 5 and 15).



Important: You must enable the virtual slave by configuring a Modbus address greater than 0 (**Project / Site Configuration**).

You can control which communication ports will support the pass-through (**Project / Site Configuration / Port X button**).

This feature requires ladder logic to read the pass-through block from the module to the processor. Refer to the Ladder Logic section for more information about the pass-through feature.

6.6 Accumulators

The accumulators store the current amount of measured quantity for a meter channel. This section provides detailed information about the accumulators.

6.6.1 Accumulator Totalizer and Residue

The accumulators are expressed as the totalizer and residue parts. This implementation allows the accumulation of a wide range of increments, while keeping a high precision of fractional part with an approximately constant and small round off error.

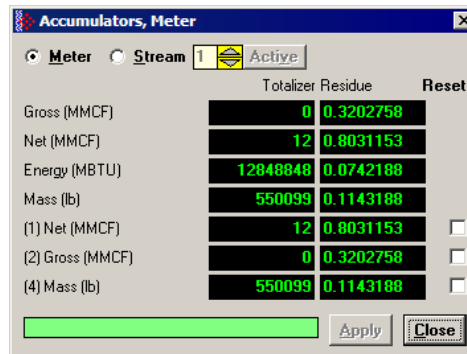
The totalizer stores the integral part of an accumulator as a 32-bit (or split) integer. The residue is the fractional part (always less than 1.0) expressed as a 32-bit IEEE floating point.

The Total Accumulator is given by the formula:

$$\text{ACCUMULATOR} = \text{TOTALIZER} + \text{RESIDUE}$$

Example

If the meter monitor window shows the following values for the accumulators:



	Totalizer	Residue	Reset
Gross (MMCF)	0	0.3202758	
Net (MMCF)	12	0.8031153	
Energy (MBTU)	12848848	0.0742188	
Mass (lb)	550099	0.1143188	
(1) Net (MMCF)	12	0.8031153	<input type="checkbox"/>
(2) Gross (MMCF)	0	0.3202758	<input type="checkbox"/>
(4) Mass (lb)	550099	0.1143188	<input type="checkbox"/>

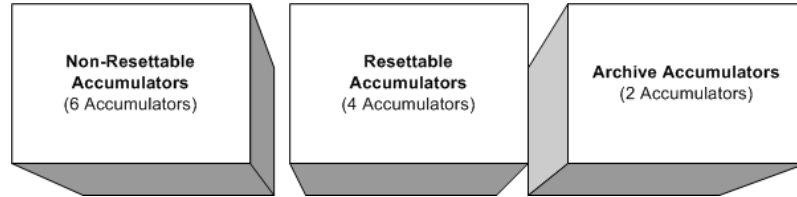
The total resettable accumulator 1 value (net) is 12.8031153.

The accumulator totalizer values can be configured to "split" with the low-order word rolling over from 9999 to 0000 at which time the high-order word is incremented. Refer to the AFC Manager (AFC Manager / Meter Configuration / Split Double Accumulators) to select this feature.

A 32-bit value is more suited to computation and has a greater range than a split value, whereas a split value is easier to read when it is represented as a pair of 16-bit numbers, as in a processor data file.

6.6.2 Accumulator Types

The module supports a total of 12 accumulators per meter channel divided into the following categories:



These 3 accumulator types are independent. For example, resetting a resettable accumulator does not affect the other accumulators.

For multiple-stream firmware (version 2.05 and later), each stream also has its own set of ten accumulators (six non-resettable and four resettable). Increments are applied both to the meter accumulators and to the accumulators for the active stream.

Non-Resetable Accumulators

The non-resettable accumulators are only reset when the accumulator rollover value is reached. The accumulator rollover value, and the accumulator unit must be configured using the AFC Manager. Refer to the AFC Manager User Manual for more information about this topic.

The module supports six non-resettable accumulators in order to show the measured quantity to be totalized:

- Non-resettable accumulator mass
- Non-resettable accumulator energy (Gas applications only)
- Non-resettable accumulator net
- Non-resettable accumulator gross
- Non-resettable accumulator gross standard (Liquid applications only). For Oil-Water Emulsion, this is non-resettable accumulator gross clean oil.
- Non-resettable accumulator water (Liquid applications only)

Refer to the Modbus Dictionary dialog box in AFC Manager for more information about the Modbus addresses for these registers.

Resetable Accumulators

The resettable accumulators are referred to as:

- Resetable Accumulator 1
- Resetable Accumulator 2
- Resetable Accumulator 3
- Resetable Accumulator 4

Configuring Resettable Accumulators

Resettable Accumulators are configured from the Resettable Accumulator Select dialog box. To open this dialog box, click the Resettable Accum button on the Meter Configuration dialog box.

Each Resettable Accumulator can be configured to represent a different quantity as follows:

Accumulator	Modbus address for accumulator select (Meter-relative)	Default Value
Resettable accumulator 1	136	Net (code 3)
Resettable accumulator 2	137	Gross (code 4)
Resettable accumulator 3	138	Gross Standard (code 5)
Resettable accumulator 4	139	Mass (code 1)

Valid Configuration Codes

The valid codes are:

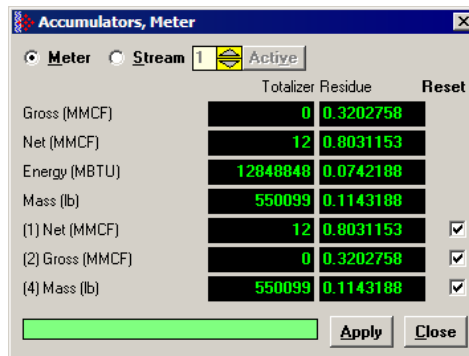
Code	Quantity
0	None
1	Mass
2	Energy (Gas Only)
3	Net
4	Gross
5	Gross Standard (Liquid Only)
6	Water (Liquid Applications Only).

For example, moving a value of 4 to holding register 8136 will configure Meter 1's resettable accumulator 1 as "Gross Volume". Moving "0" to holding register 10138 configures Meter 2's Resettable Accumulator 3 to accumulate nothing (takes it out of service).

The resettable accumulators are reset when one of the following situations occur.

Reset from AFC Manager

You may reset any of the resettable accumulators using the AFC Manager (Meter Monitor):

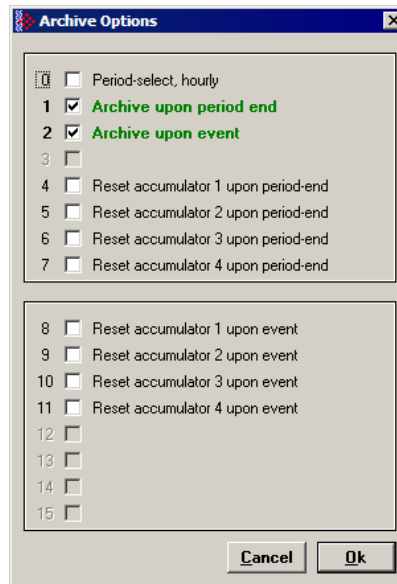


Reset from Ladder Logic

The ladder logic may send a meter signals block to command one or more resettable accumulators to be reset. This feature is especially important for applications involving field installations that require shipping and/or receiving product batches of predetermined size. Refer to the Ladder Logic section for your module type for more information.

Reset Upon Archive Period End or Reset Upon Event

Use AFC Manager to configure the resettable accumulator to be reset when the archive period ends or when an event occurs. Refer to **Event Log** in the *AFC Manager User Guide* for more information on configuring and monitoring events.



Refer to Archives for more information.

Reset When the Accumulator Rollover Value is Reached

The resettable accumulator is reset when the accumulator rollover value is reached. You must configure the accumulator rollover value using the AFC Manager software (Meter Configuration). Refer to the AFC Manager User Manual for more information about this subject.

For multiple-stream firmware (version 2.05 or later), resetting a resettable accumulator resets that accumulator for both the meter and for all its streams.

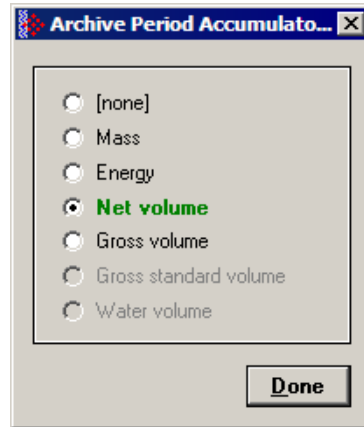
Archive Accumulators

The archive accumulators are part of the current archive (archive 0) data. These accumulators are automatically reset when a new archive is generated. The following Modbus holding registers are used:

Meter	Daily Archive		Hourly Archive	
	Accumulator: Totalizer	Accumulator: Residue	Accumulator: Totalizer	Accumulator: Residue
1	8890 to 8891	8892 to 8893	8894 to 8895	8896 to 8897
2	10890 to 10891	10892 to 10893	10894 to 10895	10896 to 10897
3	12890 to 12891	12892 to 12893	12894 to 12895	12896 to 12897
4	14890 to 14891	14892 to 14893	14894 to 14895	14896 to 14897
5	16890 to 16891	16892 to 16893	16894 to 16895	16896 to 16897
6	18890 to 18891	18892 to 18893	18894 to 18895	18896 to 18897
7	20890 to 20891	20892 to 20893	20894 to 20895	20896 to 20897
8	22890 to 22891	22892 to 22893	22894 to 22895	22896 to 22897

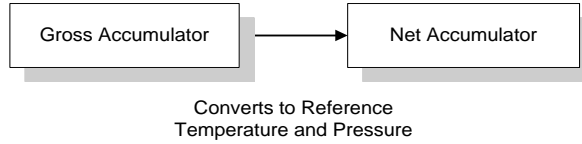
You can view the addresses, datum types and descriptions in the Modbus Dictionary dialog box.

You may configure the accumulator quantity to be used for each archive accumulator using the AFC Manager (**Meter Configuration / Archive Config / Accumulator Select**):

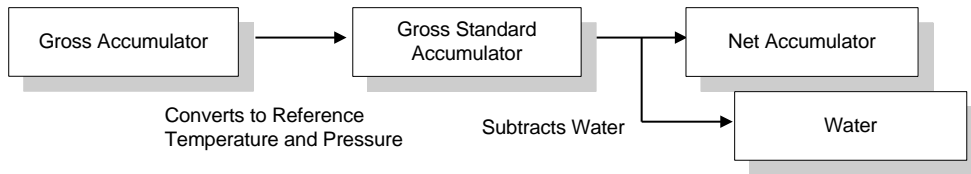


6.6.3 Net Accumulator Calculation

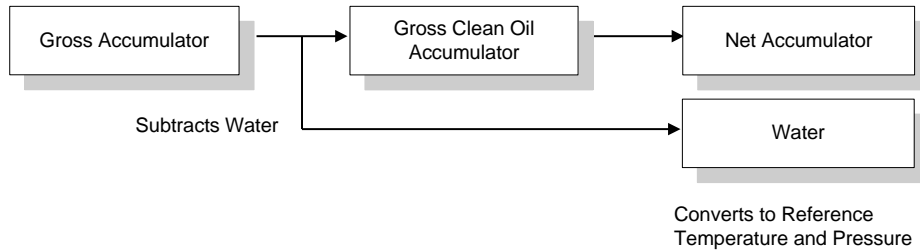
The Net Accumulator Calculation depends on the product group (gas or liquid).
For gas applications, the Net Accumulator is calculated as follows:



For liquid applications (all except Emulsion), the Net Accumulator is calculated as follows:



For liquid applications (Oil-Water Emulsion), the net accumulator is calculated as follows, using API ch 20.1:



6.6.4 Frequently Asked Questions

I need the accumulators to be reset upon period end. Which accumulator should my application use? Resettable Accumulator or Archive Accumulator?

You can use either one. The Archive Accumulators are reset every time a new archive is created and you configure whether the archive should be created upon period end and/or upon events.

There are some applications that may require the archives to be generated upon period end and upon event while the accumulators should be reset only upon period end. For these applications, you should consider the Resettable Accumulator (configured to be reset upon period end only) because the Archive Accumulators will also be reset when an event occurs.

6.7 Archives

6.7.1 Archive Overview

An archive is a set of data that records relevant process values that occurred during a certain period of time (per meter channel). The archives are automatically generated by the module and no further action is required. The process values can include:

- Net flow rate (average)
- Total accumulator
- Temperature (average)
- Alarms occurred during the period

The process values will depend on the meter type and product group as listed later in this section.

Each archive contains two values that exhibits the period of time about that archive:

- opening timestamp = starting date and time for archive
- closing timestamp = ending date and time for archive

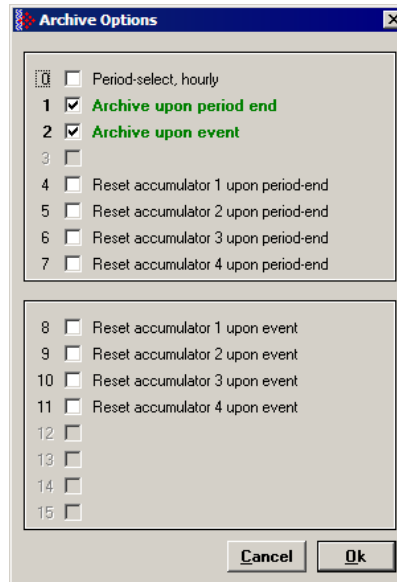
The example described in this chapter is of the default archive configuration as is present for a newly allocated meter. Version 2.01 of the firmware and AFC Manager allows the default configuration to be changed. Refer to Editing the Archive Structure.

6.7.2 Archive Generation

The archives can be generated during one of the following situations:

- Upon period end
- Upon event
- Upon processor command

You can configure if the archives should be generated upon period end and/or event using the AFC Manager (**Meter Configuration / Archive Config / Options**)



Refer to the AFC Manager User Manual for more information about this topic. By default the archives are generated upon period end and event.

If the archive is configured to be created upon period end, it will be periodically (daily or hourly) generated at the time configured by the End-of-day minute and End-of-hour minute parameters (**Project / Site Configuration**).

If the archive is configured to be created upon event, it will be generated every time an event occurs. For example, if an operator changes the orifice diameter for Meter 1, the module would automatically generate a new archive to save the relevant data to this point. Refer to this User Manual for the Events section for more information about events.

Note: Changing a meter type, product group, system of units, or primary input parameter will erase all archives for that meter.

6.7.3 Archive Monitor

The Archive Monitor dialog box opens when you open the Monitor menu, and then choose Archive.

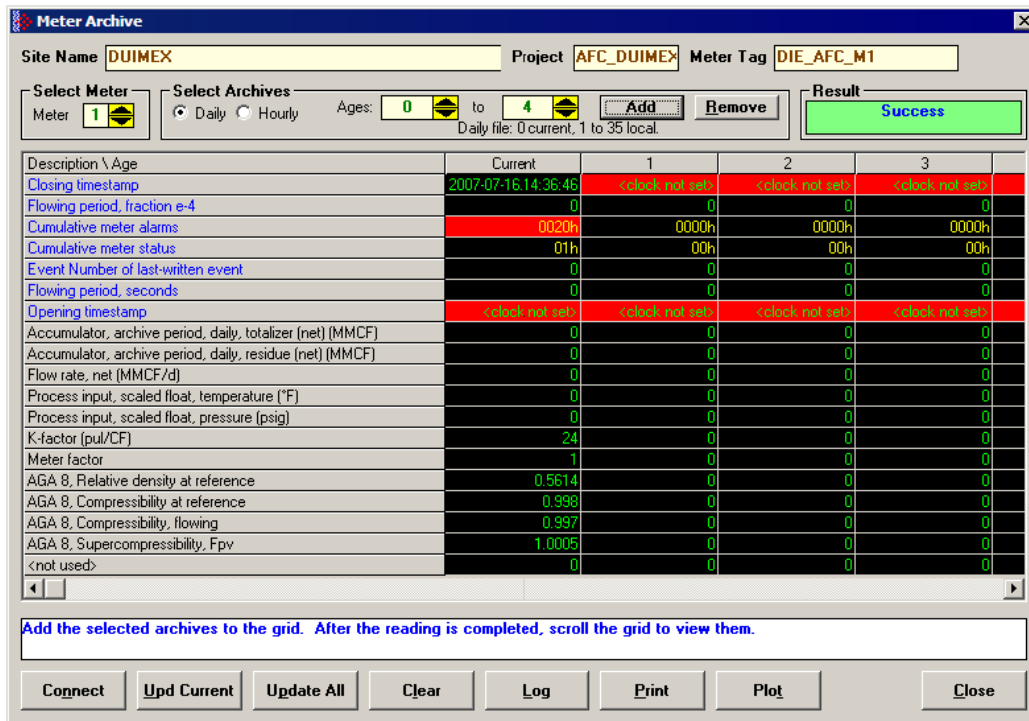
The module can archive data for each meter channel. The archives are periodically generated according to the period end defined in the Site Configuration.

There are hourly archives (48 archives) and daily archives (35 archives).

For example the daily archives will be stored as:

- Archive 0 = current archive
- Archive 1 = Archive created yesterday
- Archive 2 = Archive created 2 days ago
- Archive 3 = Archive created 3 days ago

And so on.



Control	Description
Select Meter	Select the meter number
Select Archives	Select the archive type
Ages	Select the first archive to be added or removed
To	Select the last archive to be added or removed
Add	Add the selected archives to the grid, fetching as necessary
Remove	Remove the selected archives from the grid

Control	Description
Connect	Connect to the module, if necessary
Upd Current	Update the current archive
Update All	Update all archives in the grid
Clear	Clear the grid
Log	Create a log file containing the archived data
Print	Print the archives to the local printer
Plot	Display a plot of two datum points from archives in the grid

The following shows an example of an archive report generated by the AFC Manager:

AFC-56(16) Daily Archive Date: 16-09-2002 16:26:41
 Site Name: MVI Flow Station
 Project: AFC

```

Meter 1:
Tag                M01
Archive            0

Closing timestamp of archive      2002-04-27.23:59:08
Opening timestamp of archive      2002-04-27.00:00:02
Status bitmap (details below)     00h
Alarms bitmap (details below)     0000h
Flowing period                    86346
Event counter                     53
Net accumulator (x f3)            604
Net accumulator residue (x f3)    0,6703186
Net flow rate (x f3/h)            40247,93
Temperature (°F)                  14,99997
Pressure (psig)                   999,9995
Differential pressure (hw)        21,99997
Relative density (60°F/60°F)     0,7404
Reference compressibility         0,9989
Flowing compressibility           0,9051
Fpv                               1,0505
Velocity of approach factor Ev    1,0328
Expansion factor Y                0,9997
Discharge coefficient             0,6043

Alarm Bits
bit 0  Temperature input out of range      -
bit 1  Pressure input out of range         -
bit 2  Diff. pressure input out of range   -
bit 3  Flowing density input out of range  -
bit 4  Water content input out of range    -
bit 5  Diff. pressure low                  -
bit 8  Orifice characterization error     -
bit 9  Analysis total zero                 -
  
```

```

bit 10 Analysis total not normalized -
bit 11 AGA8 calculation error -
bit 12 API calculation error, density correctio -
bit 13 API calculation error, Ctl -
bit 14 API calculation error, vapor pressure -
bit 15 API calculation error, Cpl -

```

Status Bits

```

bit 11 Meter was enabled -
bit 12 Backplane communication fault -
bit 13 Measurement configuration changed -
bit 14 Power up -
bit 15 Cold start -

```

AFC-56(16) Daily Archive
Site Name: MVI Flow Station
Project: AFC

Date: 16-09-2002 16:26:41

Meter 1:

```

Tag M01
Archive 1

Closing timestamp of archive 2002-04-27.00:00:02
Opening timestamp of archive 2002-04-26.23:59:42
Status bitmap (details below) 00h
Alarms bitmap (details below) 0000h
Flowing period 20
Event counter 53
Net accumulator (x f3) 234
Net accumulator residue (x f3) 0,1092186
Net flow rate (x f3/h) 40248,01
Temperature (°F) 15
Pressure (psig) 1000
Differential pressure (hw) 22
Relative density (60°F/60°F) 0,7404
Reference compressibility 0,9989
Flowing compressibility 0,9051
Fpv 1,0505
Velocity of approach factor Ev 1,0328
Expansion factor Y 0,9997
Discharge coefficient 0,6043

```

Alarm Bits

```

bit 0 Temperature input out of range -
bit 1 Pressure input out of range -
bit 2 Diff. pressure input out of range -
bit 3 Flowing density input out of range -
bit 4 Water content input out of range -
bit 5 Diff. pressure low -
bit 8 Orifice characterization error -
bit 9 Analysis total zero -
bit 10 Analysis total not normalized -
bit 11 AGA8 calculation error -

```



```

bit 12 API calculation error, density correctio -
bit 13 API calculation error, Ctl -
bit 14 API calculation error, vapor pressure -
bit 15 API calculation error, Cpl -
  
```

Status Bits

```

bit 11 Meter was enabled -
bit 12 Backplane communication fault -
bit 13 Measurement configuration changed -
bit 14 Power up -
bit 15 Cold start -
  
```

AFC-56(16) Daily Archive
 Site Name: MVI Flow Station
 Project: AFC

Date: 16-09-2002 16:26:44

Meter 1:

```

Tag M01
Archive 2
  
```

```

Closing timestamp of archive 2002-04-26.23:59:42
Opening timestamp of archive 2002-04-26.06:16:34
Status bitmap (details below) 60h
Alarms bitmap (details below) 0000h
Flowing period 1019877652
Event counter 53
Net accumulator (x f3) 174811
Net accumulator residue (x f3) 0,9399567
Net flow rate (x f3/h) 40247,88
Temperature (°F) 15,00736
Pressure (psig) 1000,416
Differential pressure (hw) 22,00479
Relative density (60°F/60°F) 0,7404
Reference compressibility 0,9989
Flowing compressibility 0,9053
Fpv 1,0506
Velocity of approach factor Ev 1,0331
Expansion factor Y 1,0001
Discharge coefficient 0,6045
  
```

Alarm Bits

```

bit 0 Temperature input out of range -
bit 1 Pressure input out of range -
bit 2 Diff. pressure input out of range -
bit 3 Flowing density input out of range -
bit 4 Water content input out of range -
bit 5 Diff. pressure low -
bit 8 Orifice characterization error -
bit 9 Analysis total zero -
bit 10 Analysis total not normalized -
bit 11 AGA8 calculation error -
bit 12 API calculation error, density correctio -
  
```

bit 13 API calculation error, Ctl -
bit 14 API calculation error, vapor pressure -
bit 15 API calculation error, Cpl -

Status Bits

bit 11 Meter was enabled -
bit 12 Backplane communication fault -
bit 13 Measurement configuration changed yes
bit 14 Power up yes
bit 15 Cold start -

AFC-56(16) Daily Archive
Site Name: MVI Flow Station
Project: AFC

Date: 16-09-2002 16:26:51

Meter 1:

Tag M01
Archive 3

Closing timestamp of archive 2002-04-26.06:16:34
Opening timestamp of archive 2002-04-26.06:14:08
Status bitmap (details below) 20h
Alarms bitmap (details below) 0000h
Flowing period 146
Event counter 50
Net accumulator (x f3) 1633
Net accumulator residue (x f3) 6,271362E-02
Net flow rate (x f3/h) 40248,02
Temperature (°F) 14,99999
Pressure (psig) 1000,002
Differential pressure (hw) 22,00003
Relative density (60°F/60°F) 0,7404
Reference compressibility 0,9989
Flowing compressibility 0,9051
Fpv 1,0505
Velocity of approach factor Ev 1,0328
Expansion factor Y 0,9997
Discharge coefficient 0,6043

Alarm Bits

bit 0 Temperature input out of range -
bit 1 Pressure input out of range -
bit 2 Diff. pressure input out of range -
bit 3 Flowing density input out of range -
bit 4 Water content input out of range -
bit 5 Diff. pressure low -
bit 8 Orifice characterization error -
bit 9 Analysis total zero -
bit 10 Analysis total not normalized -
bit 11 AGA8 calculation error -
bit 12 API calculation error, density correctio -
bit 13 API calculation error, Ctl -
bit 14 API calculation error, vapor pressure -

```

bit 15 API calculation error, Cpl -

Status Bits
bit 11 Meter was enabled -
bit 12 Backplane communication fault -
bit 13 Measurement configuration changed yes
bit 14 Power up -
bit 15 Cold start -
    
```

6.7.4 Archive Types

The module supports two types of archives: hourly archives and daily archives:

Archive Type	Period	Period End	Number of 30-Word Archives Stored Locally
Hourly	60 minutes (1 hour)	Set by <i>End-of-Hour Minute</i> parameter	48
Daily	1440 minutes (1 day)	Set by <i>End-of-Day Minute</i> parameter	35

The Period End parameters must be set using the AFC Manager (Site Configuration). The default value is zero for both archive types which means that:

- Daily Archives are generated every day at midnight (00:00)
- Hourly Archives are generated every hour on the hour (1:00, 2:00, 3:00, 4:00)

For example, if the parameters are configured as follows:

End-of-day minute = 480

The daily archives would be created every day at 08:00.

End-of-hour minute = 30

The hourly archives would be created every hour at 1:30, 2:30, 3:30, 4:30, and so on.

6.7.5 Archive Order

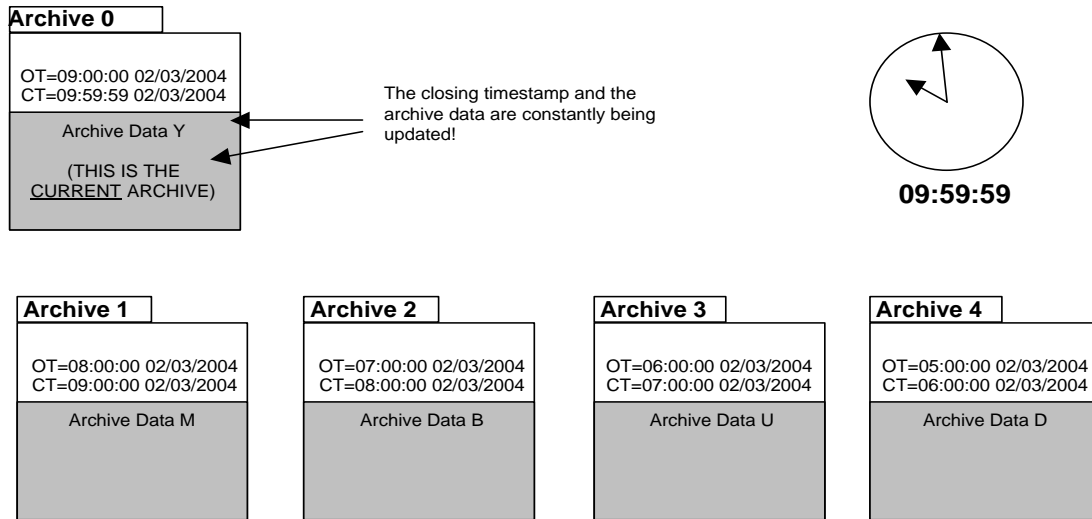
An important concept regarding this topic is the archive order. Understanding this simple concept is essential when reading archive data (through the backplane or Modbus Master). Each archive has a number (its "age") that labels its position in the archive queue. The following table shows the archive numbering scheme (both daily and hourly archives):

Archive Age	Register Types	Description
0	Holding Register	Current archive.
1	Input Register	Most recent archive
2	Input Register	Second most recent archive
3	Input Register	Third most recent archive
4	Input Register	Fourth most recent archive
...

The archive 0 is the current archive. Because its period has not been concluded its closing timestamp and values (such as accumulator, average temperature, etc...) will be continuously updated. After the period is over (or an event occurs depending on the archive configuration) the data in archive 0 will be saved as the "new" archive 1. The data in the "old" archive 1 will be saved as the new archive 2 and so forth.

The current archive is stored in the primary slave's holding register bank. The past archives are stored in the primary slave's input register bank.

The following illustration shows an example for hourly archives:



Where:

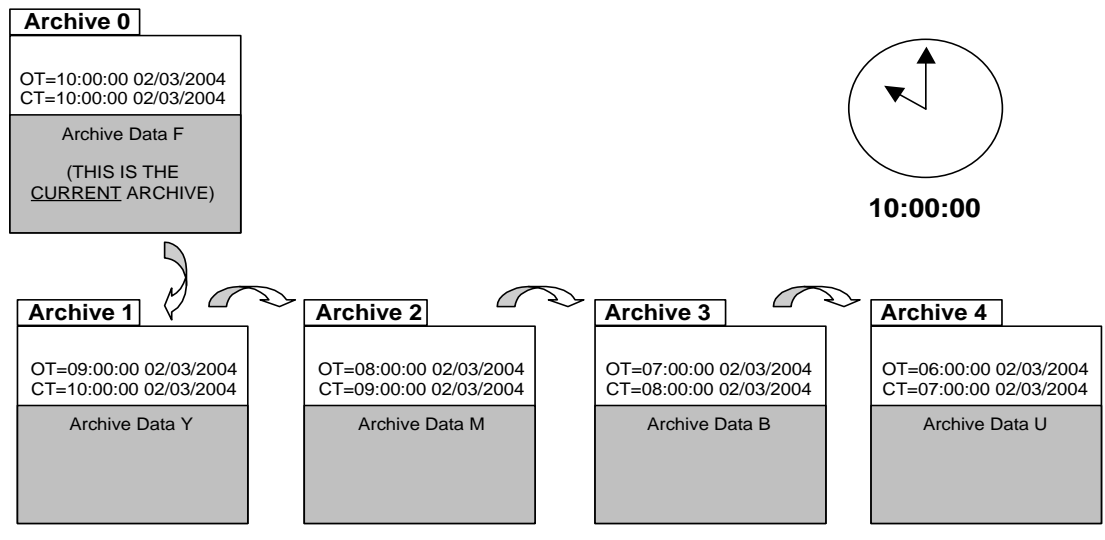
OT = Opening Time Stamp

CT = Closing Time Stamp

The previous figure shows an example where the hourly archives are configured to be generated upon period-end at the minute "0" (1:00, 2:00, 3:00, etc...).

Therefore, at 09:59:59 the archive 0 (current archive) is just about to be saved as the "new" archive 1.

When the clock changes to 10:00:00 the following illustration shows how the latest four archives are modified:



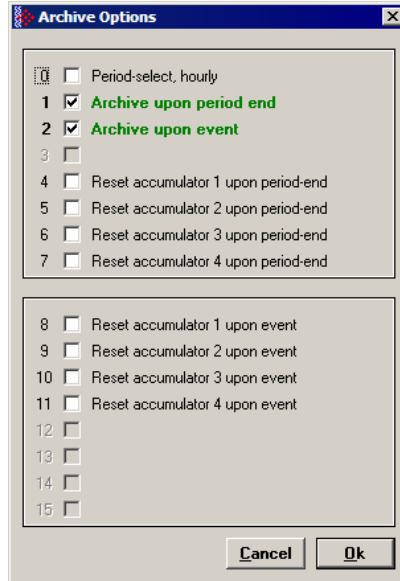
Where:

OT = Opening Time Stamp

CT = Closing Time Stamp

6.7.6 Archive Options

The module also allows you to configure whether the resettable accumulator should be reset upon period end and/or event. Most applications will require the resettable accumulators to be reset just after the archive is generated. The AFC Manager (version 2.01.000 or later) supports this feature through the archive options window as shown in the following example:



By default, the module is configured to generate archives upon period end and event. The module is not configured by default to reset the resettable accumulators upon period end.

6.7.7 Archive Locations

Click the Modbus Addresses button on the Archive Configuration dialog box to learn how to fetch an archive record of a specific age (procedure and Modbus location), and even the actual Modbus address of a specific file archived datum point (if you have highlighted the item in the archive record template).

The following table shows the current archive (Archive 0) location in the Primary Modbus Slave for each of the first 8 meters. These addresses refer to the holding register bank.

Archive 0 - Current Archives

Meter	Start Daily Archive	End Daily Archive	Start Hourly Archive	End Hourly Archive
1	9900	9939	9950	9989
2	11900	11939	11950	11989
3	13900	13939	13950	13989
4	15900	15939	15950	15989
5	17900	17939	17950	17989
6	19900	19939	19950	19989
7	21900	21939	21950	21989
8	23900	23939	23950	23989

Refer to the Modbus Dictionary dialog box for the current archive addressing.

The following table shows the past archives location in the Primary Modbus Slave for each of the first 8 meters. These addresses refer to the input register bank.

Archives 1 to n - Past Archives

Meter	Start Daily Archive	End Daily Archive	Start Hourly Archive	End Hourly Archive
1	0	1059	1060	2499
2	2500	3559	3560	4999
3	5000	6059	6060	7499
4	7500	8559	8560	9999
5	10000	11059	11060	12499
6	12500	13559	13560	14999
7	15000	16059	16060	17499
8	17500	18559	18560	19999

The default configuration sets 30 words per meter archive. For example, the Meter 1 daily archives are addressed as follows:

Daily Archive Number	Start Address	End Address
1	0	29
2	30	59
3	60	89
4	90	119
...
35	1020	1049

The Meter 1 hourly archives are addressed as follows:

Hourly Archive Number	Start Address	End Address
1	1060	1089
2	1090	1119
3	1120	1149
4	1150	1179
...
48	2470	2499

6.7.8 Configurable Archive Registers

The following table shows the possible registers that can be included in the archive definition. Use the Insert and Remove buttons on the Archive Configuration dialog box to customize the list of registers for each meter archive.

Description	Meter-Relative Address	Length
Analysis molar fraction, component 1	720	1 word
Analysis molar fraction, component 2	721	1 word
Analysis molar fraction, component 3	722	1 word
Analysis molar fraction, component 4	723	1 word
Analysis molar fraction, component 5	724	1 word
Analysis molar fraction, component 6	725	1 word
Analysis molar fraction, component 7	726	1 word
Analysis molar fraction, component 8	727	1 word
Analysis molar fraction, component 9	728	1 word
Analysis molar fraction, component 10	729	1 word
Analysis molar fraction, component 11	730	1 word
Analysis molar fraction, component 12	731	1 word
Analysis molar fraction, component 13	732	1 word
Analysis molar fraction, component 14	733	1 word
Analysis molar fraction, component 15	734	1 word
Analysis molar fraction, component 16	735	1 word
Analysis molar fraction, component 17	736	1 word
Analysis molar fraction, component 18	737	1 word

Description	Meter-Relative Address	Length
Analysis molar fraction, component 19	738	1 word
Analysis molar fraction, component 20	739	1 word
Analysis molar fraction, component 21	740	1 word
Analysis molar fraction, component 22	741	1 word
Analysis molar fraction, component 23	742	1 word
Analysis molar fraction, component 24	743	1 word
Input pulse count, archive reset, daily	840	2 words
Input pulse count, archive reset, hourly	842	2 words
Previous input pulse count	846	2 words
Current master pulse count	848	2 words
Non-resettable accumulator, mass, totalizer	850	2 words
Non-resettable accumulator, mass, residue	852	2 words
Non-resettable accumulator, energy, totalizer	854	2 words
Non-resettable accumulator, energy, residue	856	2 words
Non-resettable accumulator, net, totalizer	858	2 words
Non-resettable accumulator, net, residue	860	2 words
Non-resettable accumulator, gross, totalizer	862	2 words
Non-resettable accumulator, gross, residue	864	2 words
Non-resettable accumulator, gross standard, totalizer	866	2 words
Non-resettable accumulator, gross standard, residue	868	2 words
Non-resettable accumulator, water, totalizer	870	2 words
Non-resettable accumulator, water, residue	872	2 words
Resettable accumulator 1, totalizer	874	2 words
Resettable accumulator 1, residue	876	2 words
Resettable accumulator 2, totalizer	878	2 words
Resettable accumulator 2, residue	880	2 words
Resettable accumulator 3, totalizer	882	2 words
Resettable accumulator 3, residue	884	2 words
Resettable accumulator 4, totalizer	886	2 words
Resettable accumulator 4, residue	888	2 words
Accumulator, archive period, daily, totalizer	890	2 words
Accumulator, archive period, daily, residue	892	2 words
Accumulator, archive period, hourly, totalizer	894	2 words
Accumulator, archive period, hourly, residue	896	2 words
Process input, scaled float, temperature	1520	2 words
Process input, scaled float, pressure	1522	2 words
Process input, scaled float, dif prs / flow rate / freq	1524	2 words
Process input, scaled float, flowing density	1526	2 words
Process input, scaled float, water and sediment	1528	2 words

Description	Meter-Relative Address	Length
Process input, scaled integer, temperature	1540	1 word
Process input, scaled integer, pressure	1541	1 word
Process input, scaled integer, dif prs / flow rate / freq	1542	1 word
Process input, scaled integer, flowing density	1543	1 word
Process input, scaled integer, water and sediment	1544	1 word
Temperature, absolute	1570	2 words
Upstream pressure, absolute	1572	2 words
Densitometer frequency	1574	2 words
AGA 7 temperature base factor, Ftb	1594	2 words
AGA 7 pressure base factor, Fpb	1596	2 words
Meter alarms	1601	1 word
Orifice characterization error	1602	1 word
Analysis characterization error	1603	1 word
AGA 8 calculation error	1604	1 word
Density correction error	1605	1 word
Temperature correction error	1606	1 word
Vapor pressure error	1607	1 word
Pressure correction error	1608	1 word
Scan count, process input	1618	1 word
Scan count, calculation	1619	1 word
AGA 8, Molar mass of mixture	1620	2 words
AGA 8, Ideal gas relative density	1622	2 words
AGA 8, Compressibility at reference	1624	2 words
AGA 8, Molar density at reference	1626	2 words
AGA 8, Density at reference	1628	2 words
AGA 8, Relative density at reference	1630	2 words
AGA 8, Compressibility, flowing	1632	2 words
AGA 8, Molar density, flowing	1634	2 words
AGA 8, Density, flowing	1636	2 words
AGA 8, Supercompressibility, Fpv	1640	2 words
Previous timer tick count	1661	1 word
Scan period (seconds)	1662	2 words
AGA 3, Pressure extension	1664	2 words
AGA 3, Differential pressure in static pressure units	1666	2 words
AGA 3, Orifice bore diameter at temperature	1668	2 words
AGA 3, Meter tube internal diameter at temperature	1670	2 words
Reserved	1672	2 words
AGA 3, Density, flowing	1674	2 words
AGA 3, Mass flow rate, Qm	1678	2 words
AGA 3, Velocity of approach factor, Ev	1680	2 words

Description	Meter-Relative Address	Length
AGA 3, Expansion factor, Y	1682	2 words
AGA 3, Coefficient of discharge, Cd	1684	2 words
AGA 3, Composition factor	1686	2 words
AGA 7, Temperature factor, Ftm	1694	2 words
AGA 7, Pressure factor, Fpm	1696	2 words
AGA 7, C-prime	1698	2 words
Molar heating value, MJ/kmol	1700	2 words
Mass heating value	1702	2 words
Volumetric heating value	1704	2 words
API 2540, Density at API base temperature	1738	2 words
API 2540, Hydrometer correction factor	1740	2 words
API 2540, Density at reference	1742	2 words
API 2540, Vapor pressure	1744	2 words
API 2540, CPL low density factor A	1746	2 words
API 2540, CPL low density factor B	1748	2 words
API 2540, CPL factor F	1750	2 words
API 2540, Temperature correction factor, CTL	1752	2 words
API 2540, Pressure correction factor, CPL	1754	2 words
API 2540, Sediment and water correction factor, CSW	1756	2 words
Density calculation select	1759	1 word
AGA 8, Ideal gas relative density - scaled integer	1761	1 word
AGA 8, Compressibility at reference - scaled integer	1762	1 word
AGA 8, Relative density at reference - scaled integer	1765	1 word
AGA 8, Compressibility, flowing - scaled integer	1766	1 word
AGA 8, Supercompressibility, Fpv - scaled integer	1770	1 word
Reserved	1786	1 word
AGA 3, Velocity of approach factor - scaled integer	1790	1 word
AGA 3, Expansion factor - scaled integer	1791	1 word
AGA 3, Coefficient of discharge - scaled integer	1792	1 word
API 2540, Density at reference	1821	1 word
API 2540, Vapor pressure	1822	1 word
API 2540, Temperature correction factor, CTL	1826	1 word
API 2540, Pressure correction factor, CPL	1827	1 word
API 2540, Sediment and water correction factor, CSW	1828	1 word
Startup input pulse count	1840	2 words
Current input pulse count	1842	2 words
Pulse increment	1844	2 words
Pulse frequency	1846	2 words
Interpolated/static K-factor	1848	2 words

Description	Meter-Relative Address	Length
Interpolated/static meter factor	1850	2 words
Multiplier, mass flow rate	1864	2 words
Multiplier, energy flow rate	1866	2 words
Multiplier, volume flow rate	1868	2 words
Multiplier, mass accumulator	1870	2 words
Multiplier, energy accumulator	1872	2 words
Multiplier, volume accumulator	1874	2 words
Accumulator increment, mass	1876	2 words
Accumulator increment, energy	1878	2 words
Accumulator increment, net	1880	2 words
Accumulator increment, gross	1882	2 words
Accumulator increment, gross standard	1884	2 words
Accumulator increment, water	1886	2 words
Flow rate, mass	1888	2 words
Flow rate, energy	1890	2 words
Flow rate, net	1892	2 words
Flow rate, gross	1894	2 words
Flow rate, gross standard	1896	2 words
Flow rate, water	1898	2 words

Information for Users of AFC Manager Versions Older Than 2.01.000

If you are using AFC Manager versions older than 2.01.000, you must set these bits using the Modbus master interface in the AFC Manager. Please refer to the AFC Manager User Manual for further information about the Modbus Master interface feature.

Refer to the following words to configure the archive options directly to the Modbus database:

Address	Description
8341	Meter 1 daily archive configuration word
8421	Meter 1 hourly archive configuration word
10341	Meter 2 daily archive configuration word
10421	Meter 2 hourly archive configuration word
12341	Meter 3 daily archive configuration word
12421	Meter 3 hourly archive configuration word
14341	Meter 4 daily archive configuration word
14421	Meter 4 hourly archive configuration word
16341	Meter 5 daily archive configuration word
16421	Meter 5 hourly archive configuration word
18341	Meter 6 daily archive configuration word
18421	Meter 6 hourly archive configuration word
20341	Meter 7 daily archive configuration word

Address	Description
20421	Meter 7 hourly archive configuration word
22341	Meter 8 daily archive configuration word
22421	Meter 8 hourly archive configuration word

Each archive configuration word has the following bitmap structure:

Bit	Description
0	Period select, hourly
1	Archive upon period end
2	Archive upon event
3	Reserved
4	Reset resettable accumulator 1 upon period end
5	Reset resettable accumulator 2 upon period end
6	Reset resettable accumulator 3 upon period end
7	Reset resettable accumulator 4 upon period end
8	Reset resettable accumulator 1 upon event
9	Reset resettable accumulator 2 upon event
10	Reset resettable accumulator 3 upon event
11	Reset resettable accumulator 4 upon event
12	Reserved
13	Reserved
14	Reserved
15	Reserved

Note: Bit 0 must be set only for the hourly archives.

Changes made directly to the Modbus table in this manner are not automatically made to your open AFC configuration. To incorporate these changes into your configuration so that they may be saved in the AFC file on your hard disk, you must read back the meter configuration from the module after making the change by using the "Read Configuration" button on the Meter Configuration window.

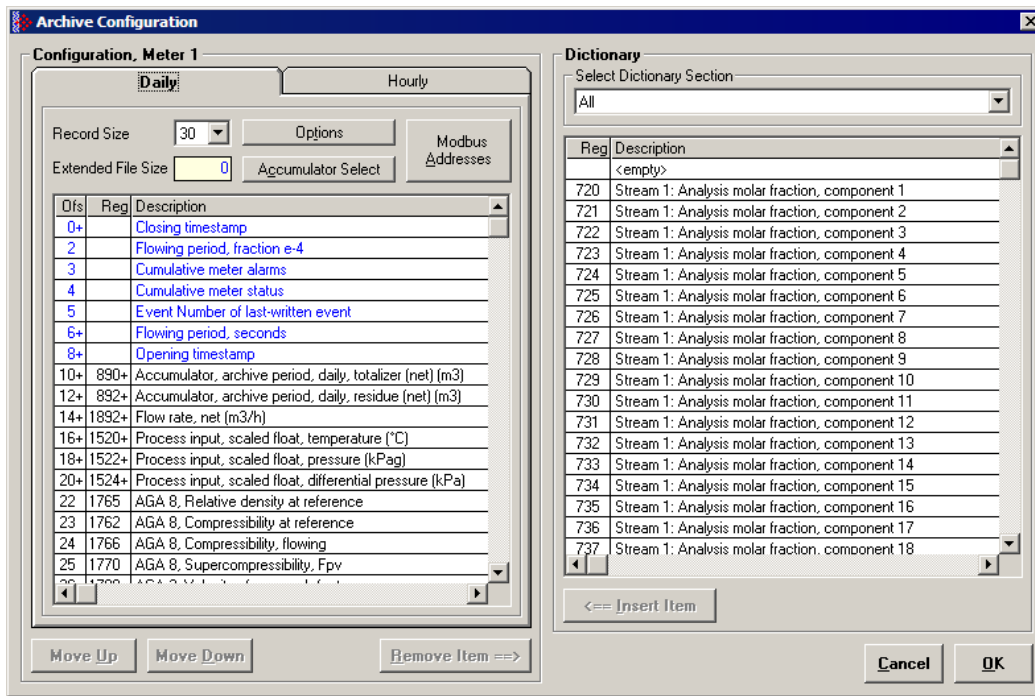
6.7.9 Editing the Archive Structure

Note: The features presented on this section are only available for AFC firmware version 2.01.000 or later. Please contact the tech support team for more information about the module upgrade.

For advanced applications, you can edit the archive contents, the record size, the order of the registers in the archive, and the archive accumulator quantity.

The Archive Configuration window (**Meter Configuration / Archive Config**) allows you to fully configure the meter archive (daily or hourly). The data to be inserted in the archive must be copied from the Dictionary Section on the right half of the window.

Refer to the AFC Manager User Manual for more information about this topic.



The module reserves 1060 words for daily archives and 1440 words for hourly archives. Because the default configuration sets the record size for 30 words, it means that the maximum (default) number of archives per meter channel is 35 daily archives and 48 hourly archives. However, because you can change the number of words per archive, the actual maximum number of archives per meter channel will depend on the configured number of words per archive as follows:

Number of Words per Archive	Number of Daily Archives	Number of Hourly Archives
10	106 daily archives	144 hourly archives
20	53 daily archives	72 hourly archives
30	35 daily archives	48 hourly archives
40	26 daily archives	36 hourly archives

You may also configure the accumulator type for each archive. You must configure one of the following options:

- Mass
- Energy (Gas product only)
- Net Volume
- Gross Volume
- Gross Standard
- Water Volume (Liquid product only)

The following topics show the default archive structure when you configure a new meter. You can edit this structure according to your own requirements.

6.7.10 Extended Archives

This feature is only supported on firmware versions 2.01.000 or newer, and requires a Compact Flash card to be installed.

The module supports the extended archive feature that allows you to configure more archives than the regular 35 daily archives and 48 hourly archives. The module supports the following number of extended archives:

	Daily Archives	Hourly Archives
Max Number of Archives	350 (version 2.04 and earlier) 1440 (version 2.05 and newer)	1260 (version 2.04 and earlier) 1440 (version 2.05 and newer)

Refer to Extended File Size entry on the **Archive Configuration** window for more information.

Note: The maximum number of extended archives is not dependent on the number of words per archive. Extended archives are stored on a Compact Flash card which must be installed for Extended Archive configuration to be effective.

Retrieving Extended Archives

The module implements an easy way to retrieve extended archives from the Modbus database. To learn how to retrieve extended archives, click Archive Config on the Meter Configuration dialog box, and then click Modbus Addresses. For each archive file the module reserves a block of 50 Input registers to hold the "selected Archive", as listed in the following table.

Meter	Daily Archive Start (Input Register)	Daily Archive End (Input Register)	Hourly Archive Start (Input Register)	Hourly Archive End (Input Register)
1	60000	60049	60050	60099
2	60100	60149	60150	60199
3	60200	60249	60250	60299
4	60300	60349	60350	60399
5	60400	60449	60450	60499
6	60500	60549	60550	60599
7	60600	60649	60650	60699
8	60700	60749	60750	60799

Meter	Daily Archive Start (Input Register)	Daily Archive End (Input Register)	Hourly Archive Start (Input Register)	Hourly Archive End (Input Register)
9	60800	60849	60850	60899
10	60900	60949	60950	60999
11	61000	61049	61050	61099
12	61100	61149	61150	61199
13	61200	61249	61250	61299
14	61300	61349	61350	61399
15	61400	61449	61450	61499
16	61500	61549	61550	61599

Note: Meters 9 through 16 are only available for the PTQ-AFC and <MODEL NUMBER> modules.

The Selected Archive start address can be calculated as (assumes meters are numbered starting at 1):

Daily Archive Start Address = $60000 + ((\text{Meter Number} - 1) * 100)$

Hourly Archive Start Address = $60000 + (((\text{Meter Number} - 1) * 100) + 50)$

Note: When using processor logic to retrieve extended archives, when possible, use unsigned 16-bit integer data type variables to hold archive addresses. Unsigned 16-bit integers display data in the range 0 to 65535.

If your programming software (such as Rockwell Automation® RSLogix™5000) does not support unsigned integer data types, there are a couple of possible alternatives. If your programming software supports signed 32-bit double integer data types, you may use that type of variable to hold the addresses above.

If you must use signed 16-bit integer data type variables to contain addresses (such as in the case of Rockwell Automation RSLogix5 or RSLogix500), you will not be able to enter the values in the previous table as positive numbers. This is because 16-bit signed integers display values only in the range -32768 to +32767. But, it is the underlying bit pattern and not the displayed decimal value that is important to the AFC module.

To enter the correct bit pattern for these addresses into a signed 16-bit integer, you will need to enter them as negative numbers. To determine the correct negative number, simply subtract 65536 from the address in the table, which will result in a negative number being displayed in the signed integer variable. This negative number (a binary twos-compliment form of the archive address) will contain the equivalent bit pattern for the value in the chart if it were held in an unsigned integer variable.

Use these modified versions of the above formulas to calculate the address values for signed 16-bit integer variables:

Daily Archive Start Address = $((60000 + ((\text{Meter Number} - 1) * 100)) - 65536)$

Hourly Archive Start Address = $((60000 + (((\text{Meter Number} - 1) * 100) + 50)) - 65536)$

The Selected Archive is continuously maintained to be a copy of the archive record having the age given in the corresponding "Archive Select" holding register, as listed in the following table. This means that the Selected Archive changes whenever either (a) the age in the Open Archive Select register is changed or (b) when the posting of a new archive causes the ages of all archives to be increased by 1.

Meter	Open Daily Archive Select Address	Open Hourly Archive Select Address
1	8300	8301
2	10300	10301
3	12300	12301
4	14300	14301
5	16300	16301
6	18300	18301
7	20300	20301
8	22300	22301
9	24300	24301
10	26300	26301
11	28300	28301
12	30300	30301
13	32300	32301
14	34300	34301
15	36300	36301
16	38300	38301

Note: Meters 9 through 16 are only available for the PTQ-AFC and <MODEL NUMBER> modules.

Use the following procedure to retrieve extended archives:

- 1 Copy the archive age to the correct Open Archive Select register.
- 2 Read the archive data from the 60000-range input addresses.

Example

To read Meter 2 Hourly Archive Number 277:

- 1 Write a value of 277 to Modbus Holding Register 10301.
- 2 Read the archive record data starting at input register 60150.

Note: This procedure can also be used to retrieve regular archives.

6.7.11 Archive Reports

Use the Archive Monitor in AFC Manager to generate an archive report or print it to a local printer. You can also save the archive report in two formats:

- Text
- Comma Separated

A report saved in **text format** (.log) contains a complete archive description. The following illustration shows an example of a text format report.

```
AFC-56(16) [2.02] Daily Archive                               Date: 4/15/2004 9:23:52 AM
Site Name: MVI Flow Station
Project: AFC
File: \\.\$AFC-56(16)
```

```
Meter 16:
  Tag                               M01
  Archive                           33

  Closing timestamp                 2004-04-17.01:49:42
  Flowing period, fraction e-4      1
  Cumulative meter alarms           0000h
  Cumulative site status            00h
  Event Number of last-written event 160
  Flowing period, seconds            16
  Opening timestamp                 2004-04-17.01:49:26
  Accumulator, archive period, daily, totalizer (m3) 0
  Accumulator, archive period, daily, residue (m3) 0.4645103
  Flow rate, net (m3/h)              101.4091
  Process input, scaled float, temperature (°C) 20
  Process input, scaled float, pressure (kPag) 50
  Process input, scaled float, dif prs / flow rate / freq (kPa) 70
  AGA 8, Relative density at reference 0.5548
  AGA 8, Compressibility at reference 0.998
  AGA 8, Compressibility, flowing 0.9959
  AGA 8, Supercompressibility, Fpv 1.001
  AGA 3, Velocity of approach factor 1
  AGA 3, Expansion factor            0.9017
  AGA 3, Coefficient of discharge    0.5975
  <not used>                        0

  Alarm Bits
  bit 0 Temperature input out of range -
  bit 1 Pressure input out of range -
  bit 2 Differential pressure input out of range -
  bit 3 Flowing density input out of range -
  bit 4 Water content input out of range -
  bit 5 Differential pressure low -
  bit 7 Accumulator overflow -
  bit 8 Orifice characterization error -
  bit 9 Analysis total zero -
  bit 10 Analysis total not normalized -
  bit 11 Compressibility calculation error -
  bit 12 API calc error - density correction -
  bit 13 API calc error - Ct1 -
  bit 14 API calc error - vapour pressure -
  bit 15 API calc error - Cp1 -

  Status Bits
  bit 11 Meter was enabled -
  bit 12 Backplane communication fault -
  bit 13 Measurement configuration changed -
  bit 14 Power up -
  bit 15 Cold start -
```

Saving the archive report in **comma-separated** (.csv) format allows it to be imported to an Excel® spreadsheet. The following example shows a portion of the .CSV report imported into Excel:

	A	B	C	D
1	AFC-71(8) [2.02] Daily Archive			
2	Date:	3/30/2004 11:21		
3	Site Name:	MVI Flow Station		
4	Project:	AFC		
5	Meter 2:			
6	Tag	M01		
7				
8	Archive	Current	1	2
9				
10	Closing timestamp	2004-03-30.08:36:54	2004-03-30.00:00:00	2004-03-29.00:00:00
11	Flowing period, fraction e-4	1	1	1
12	Cumulative meter alarms	0002h	0002h	0002h
13	Cumulative site status	00h	00h	00h
14	Event Number of last-written event	474	474	474
15	Flowing period, seconds	31014	86400	86400
16	Opening timestamp	2004-03-30.00:00:00	2004-03-29.00:00:00	2004-03-28.00:00:00
17	Accumulator, archive period, daily, totalizer (m3)	7	20	20
18	Accumulator, archive period, daily, residue (m3)	0.3965147	0.6051551	0.6052574
19	Flow rate, net (m3/h)	0.8572201	0.8571935	0.8571963
20	Process input, scaled float, temperature (°C)	49.99487	50.03679	50.03685
21	Process input, scaled float, pressure (kPag)	1	1	1
22	Process input, scaled float, dif prs / flow rate / freq (kPa)	11.00041	11.00247	11.00249
23	Process input, scaled float, flowing density (kg/m3)	700.3123	700.6372	700.6348
24	API 2540, Density at reference (kg/m3)	730.3	730.7	730.7
25	API 2540, Temperature correction factor, CTL	0.9592	0.9596	0.9596
26	API 2540, Pressure correction factor, CPL	0.9999	1.0001	1.0001
27	AGA 3, Velocity of approach factor	1	1.0003	1.0003
28	AGA 3, Expansion factor	0.9999	1.0001	1.0001
29	AGA 3, Coefficient of discharge	0.5964	0.5966	0.5966
30				

6.7.12 Archive Data Format

There are 3 columns associated with each archive data:

Column	Description
Ofs	Shows the offset location of the data in each archive. The maximum offset value will depend on the <i>Record Size</i> value you configured. If the value has a "+" value (for example 0+) it means that the data occupies 2 words of data.
Reg	Shows the Primary Modbus Slave Address of the data. This is a meter-relative address. For example: a Reg value of 890+ for meter 1 would be equivalent to Modbus addresses 8890 and 8891.
Description	Data Description.

Timestamp Date and Time Format

The date and time format used in the archives is stored in a highly compressed form in order to represent the date and time using only 2 words of data:

Word	Description
0	Date
1	Time

In order to extract the information from the date format use the following arithmetic:

Date Word

Year = ([Bits 15 thru 9] from Word 0) + 1996

Month = ([Bits 8 thru 5] from Word 0) + 1

Day = ([Bits 4 thru 0] from Word 0) + 1

Time Word

Hour = ([Bits 15 thru 11] from Word 1)

Minute = ([Bits 10 thru 5] from Word 1)

Second = ([Bits 4 thru 0] from Word 1) * 2

The first 10 words of data (archive header) are common for all archives:

Pre-defined Header

These archive areas are included in the default archive data, and cannot be reconfigured by the user.

Start Offset	End Offset	Data Format	Type	Description
0	1	Timestamp	Snapshot	Closing timestamp of archive
2		Word	Calculated	Flowing period
3		Bitmap	Calculated	Cumulative meter alarms
4		Bitmap	Calculated	Cumulative status
5		Word	Snapshot	Event counter
6	7	Double word	Calculated	Flowing period, seconds
8	9	Timestamp	snapshot	Opening timestamp of archive

Additional areas are also included in the default archive data, according to the meter type and product group associated with the meter.

The cumulative meter alarms are defined as follows:

Offset	Description
0	Current archive, daily, cumulative meter alarm: Input out of range, temperature
1	Current archive, daily, cumulative meter alarm: Input out of range: pressure
2	Current archive, daily, cumulative meter alarm: Input out of range: differential pressure
3	Current archive, daily, cumulative meter alarm: Input out of range: flowing density
4	Current archive, daily, cumulative meter alarm: Input out of range: water content
5	Current archive, daily, cumulative meter alarm: Differential Pressure Low
6	Current archive, daily, cumulative meter alarm: Orifice Pressure Exception
7	Current archive, daily, cumulative meter alarm: Accumulation overflow
8	Current archive, daily, cumulative meter alarm: Orifice characterization error
9	Not Used
10	Current archive, daily, cumulative meter alarm: Current archive, daily, cumulative meter alarm: Analysis characterization error

Offset	Description
11	Current archive, daily, cumulative meter alarm: Compressibility calculation error
12	Current archive, daily, cumulative meter alarm: Reference density error
13	Current archive, daily, cumulative meter alarm: Temperature correction error
14	Current archive, daily, cumulative meter alarm: Vapor pressure error
15	Current archive, daily, cumulative meter alarm: Pressure correction error

The cumulative status bits are defined as follows:

Offset	End Offset
00	Stream 1 active
01	Stream 2 active
02	Stream 3 active
03	Stream 4 active
11	Meter enabled
12	Backplane Communication Fault
13	Measurement Configuration Changed
14	Power up
15	Cold Start

The following 20 words (default configuration) will depend on the meter type and product group as follows:

Orifice (Differential) Meter with Gas Product

Start Offset	End Offset	Data Format	Type	Description
10	11	Accumulator	Snapshot	Accumulator totalizer, net
12	13	Floating point	Snapshot	Accumulator residue, net
14	15	Floating point	Flow weighted average	Flow rate, net
16	17	Floating point	Flow weighted average	Temperature
18	19	Floating point	Flow weighted average	Pressure
20	21	Floating point	Flow weighted average	Differential pressure
22		Word	Flow weighted average	Relative density, e-4
23		Word	Flow weighted average	Compressibility, reference, e-4
24		Word	Flow weighted average	Compressibility, flowing, e-4
25		Word	Flow weighted average	Supercompressibility, e-4
26		Word	Flow weighted average	Velocity of approach factor, Ev, e-4
27		Word	Flow weighted average	Expansion factor, Y, e-4
28		Word	Flow weighted average	Coefficient of discharge, Cd, e-4
29		Word		(available)

Pulse (Linear) Meter with Gas Product

Start Offset	End Offset	Data Format	Type	Description
10	11	Accumulator	Snapshot	Accumulator totalizer, net
12	13	Floating point	Snapshot	Accumulator residue, net
14	15	Floating point	Flow weighted average	Flow rate, net
16	17	Floating point	Flow weighted average	Temperature
18	19	Floating point	Flow weighted average	Pressure
20	21	Floating point	Flow weighted average	K-Factor
22	23	Floating point	Flow weighted average	Meter Factor
24		Word	Flow weighted average	Relative density, e-4
25		Word	Flow weighted average	Compressibility, reference, e-4
26		Word	Flow weighted average	Compressibility, flowing, e-4
27		Word	Flow weighted average	Supercompressibility, e-4
28	29	Double Word	Snapshot	Pulse Count

Orifice (Differential) Meter with Liquid Product

Start Offset	End Offset	Data Format	Type	Description
10	11	Accumulator	Snapshot	Accumulator totalizer, net
12	13	Floating point	Snapshot	Accumulator residue, net
14	15	Floating point	Flow weighted average	Flow rate, net
16	17	Floating point	Flow weighted average	Temperature
18	19	Floating point	Flow weighted average	Pressure
20	21	Floating point	Flow weighted average	Differential pressure
22	23	Floating point	Flow weighted average	Density input
24		Word	Flow weighted average	Corrected density (scaled integer)
25		Word	Flow weighted average	CTL e-4
26		Word	Flow weighted average	CPL e-4
27		Word	Flow weighted average	Velocity of approach factor, Ev, e-4
28		Word	Flow weighted average	Expansion factor, Y, e-4
29		Word	Flow weighted average	Coefficient of discharge, Cd, e-4

Pulse (Linear) Meter with Liquid Product

Start Offset	End Offset	Data Format	Type	Description
10	11	Accumulator	Snapshot	Accumulator totalizer, net
12	13	Floating point	Snapshot	Accumulator residue, net
14	15	Floating point	Flow weighted average	Flow rate, net
16	17	Floating point	Flow weighted average	Temperature
18	19	Floating point	Flow weighted average	Pressure
20	21	Floating point	Flow weighted average	K-Factor
22	23	Floating point	Flow weighted average	Meter Factor
24	25	Floating point	Flow weighted average	Density Input
26		Word	Flow weighted average	Water content, % e-2
27		Word	Flow weighted average	Corrected density (scaled integer)
28		Word	Flow weighted average	CTL e-4
29		Word	Flow weighted average	CPL e-4

Flow Rate Integration with Gas Product

Start Offset	End Offset	Data Format	Type	Description
10	11	Accumulator	Snapshot	Accumulator totalizer, net
12	13	Floating point	Snapshot	Accumulator residue, net
14	15	Floating point	Flow weighted average	Flow rate, net
16	17	Floating point	Flow weighted average	Temperature
18	19	Floating point	Flow weighted average	Pressure
20	21	Floating point	Flow weighted average	Flow Rate Input
22		Word	Flow weighted average	Relative density, e-4
23		Word	Flow weighted average	Compressibility, reference, e-4
24		Word	Flow weighted average	Compressibility, flowing, e-4
25		Word	Flow weighted average	Supercompressibility, e-4
26		Word		(available)
27		Word		(available)
28		Word		(available)
29		Word		(available)

Pulse Frequency Integration with Gas Product

Start Offset	End Offset	Data Format	Type	Description
10	11	Accumulator	Snapshot	Accumulator totalizer, net
12	13	Floating point	Snapshot	Accumulator residue, net
14	15	Floating point	Flow weighted average	Flow rate, net
16	17	Floating point	Flow weighted average	Temperature

Start Offset	End Offset	Data Format	Type	Description
18	19	Floating point	Flow weighted average	Pressure
20	21	Floating point	Flow weighted average	K-Factor
22	23	Floating point	Flow weighted average	Meter Factor
24		Word	Flow weighted average	Relative density e-4
25		Word	Flow weighted average	Compressibility, reference, e-4
26		Word	Flow weighted average	Compressibility, flowing, e-4
27		Word	Flow weighted average	Supercompressibility, e-4
28	29	Floating point	Flow weighted average	Pulse Frequency

Flow Rate Integration with Liquid Product

Start Offset	End Offset	Data Format	Type	Description
10	11	Accumulator	Snapshot	Accumulator totalizer, net
12	13	Floating point	Snapshot	Accumulator residue, net
14	15	Floating point	Flow weighted average	Flow rate, net
16	17	Floating point	Flow weighted average	Temperature
18	19	Floating point	Flow weighted average	Pressure
20	21	Floating point	Flow weighted average	Flow Rate Input
22	23	Floating point	Flow weighted average	Density Input
24		Word	Flow weighted average	Corrected density (scaled integer)
25		Word	Flow weighted average	CTL e-4
26		Word	Flow weighted average	CPL e-4
27		Word		(available)
28		Word		(available)
29		Word		(available)

Pulse Frequency Integration with Liquid Product

Start Offset	End Offset	Data Format	Type	Description
10	11	Accumulator	Snapshot	Accumulator totalizer, net
12	13	Floating point	Snapshot	Accumulator residue, net
14	15	Floating point	Flow weighted average	Flow rate, net
16	17	Floating point	Flow weighted average	Temperature
18	19	Floating point	Flow weighted average	Pressure
20	21	Floating point	Flow weighted average	K-Factor

Start Offset	End Offset	Data Format	Type	Description
22	23	Floating point	Flow weighted average	Meter Factor
24	25	Floating point	Flow weighted average	Density Input
26		Word	Flow weighted average	Water content, % e-2
27		Word	Flow weighted average	Corrected density (scaled integer)
28	29	Floating point	Flow weighted average	Pulse Frequency

Example 1

Find the Net Accumulator addresses at archive 1 (latest daily archive) for the first 4 meters.

Primary Modbus Slave Register Address	Description
10 and 11	Net Accumulator Totalizer from archive 1 - Meter 1
2510 and 2511	Net Accumulator Totalizer from archive 1 - Meter 2
5010 and 5011	Net Accumulator Totalizer from archive 1 - Meter 3
7510 and 7511	Net Accumulator Totalizer from archive 1 - Meter 4

Example 2

Find the Net Accumulator addresses at archive 0 (current daily archive) for the first 4 meters.

Primary Modbus Slave Holding Register Address	Description
9910 and 9911	Net Accumulator Totalizer from archive 0 - Meter 1
11910 and 11911	Net Accumulator Totalizer from archive 0 - Meter 2
13910 and 13911	Net Accumulator Totalizer from archive 0 - Meter 3
15910 and 15911	Net Accumulator Totalizer from archive 0 - Meter 4

6.8 Events

6.8.1 The Event Log

An "event" is any occurrence that may affect the manner in which, or whether, measurement is performed. Events include, for example:

- Any change to a sealable parameter.
- Power-up (product may have been lost during the power-down period).
- A change in PLC operating mode (programming changes may alter measurement).
- A download of the event log (for audit trail purposes).

The Event Log occupies a block of 16000 Input registers in the Modbus table starting at address 40000 and proceeding through address 55999. It consists of a 5-register "header" at address 40000 followed by 1999 8-register "event" records starting at address 40008. As they are Input registers (read with Modbus function code 4), no part of the Event Log can be written from outside the module, but it is maintained exclusively by the AFC firmware.

As events occur they are recorded in the Log, which acts as a circular file. Each new event record overwrites the oldest one, hence the log stores up to 1999 of the most recent events. As each record is written the values in the header are updated to reflect the new status of the log.

Auditors may require the Log to be "downloaded" from time to time; events are read from the module and stored in a more permanent database, and the events so copied and archived are marked in the module as "downloaded".

If all record positions contain events that have not yet been downloaded, the log is full. In this case, the handling of a new event depends on the value of the "Event log unlocked" site option:

- If the option is set, then the log-full condition is ignored and the new event overwrites the oldest one. Since the overwritten event was never downloaded, it is permanently lost.
- If the option is clear, then the Event Log is "locked", and the new event is rejected if possible and otherwise ignored. Controllable events, that is, changes to sealable parameters, are not allowed to occur; such datum points remain unchanged retaining their current values and a Modbus command that attempts such a change receives an "illegal data" exception response. Uncontrollable events, such as PLC mode change, are simply not recorded. The Log must be downloaded in order to unlock it for further events.

6.8.2 Event Log Structures

The Event Log header contains housekeeping information for maintaining the Log. Its layout is:

Address	Description
40000	Number of records maximum (= 1999)
40001	Next new record position (0 thru maximum -1)
40002	Next new event number (0 thru 65535, wrapping to 0)
40003	Oldest event number on file
40004	Oldest event number on file not yet downloaded
40005 to 40007	[reserved]

Each event record is an 8-register quantity laid out as four 32-bit items (big-endian):

Registers	Contents
0 to 1	Event Id Tag (page 180)
2 to 3	Timestamp of event In our standard "packed bit-field" format.
4 to 5	Old item value For a Datum Point event, format depends on the "datum type" field of the Event Id Tag.
6 to 7	New item value For a Datum Point event, format depends on the "datum type" field of the Event Id Tag.

Each value is right-justified in its field and sign-extended if necessary.

6.8.3 Event Id Tag

This 32-bit field has the following structure:

Bits	N	Meaning																																																			
31	1	0 Special, 1 Datum Point (e.g. sealable parameter) If this bit is clear, then bits 19-00 contain a value from the Special event tag list below; if the bit is set, then bits 19-00 have the interpretation given here.																																																			
30	1	PLC offline; timestamp may not be accurate This bit may also be set for a Special event.																																																			
29	1	[reserved]																																																			
28 to 24	5	Meter number, or 0 for Site This field may also be set for a Special event.																																																			
23 to 20	4	[Meter] Stream number or 0; [Site] 0 This field may also be set for a Special event.																																																			
19 to 16	4	Data type: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Value</th> <th>Mnemonic</th> <th>Format</th> </tr> </thead> <tbody> <tr><td>0</td><td>Ubyt</td><td>Unsigned byte</td></tr> <tr><td>1</td><td>Usht</td><td>Unsigned short integer</td></tr> <tr><td>2</td><td></td><td>[reserved]</td></tr> <tr><td>3</td><td>Ulng</td><td>Unsigned long integer</td></tr> <tr><td>4</td><td>Sbyt</td><td>Signed byte</td></tr> <tr><td>5</td><td>Ssht</td><td>Signed short integer</td></tr> <tr><td>6</td><td></td><td>[reserved]</td></tr> <tr><td>7</td><td>Slng</td><td>Signed long integer</td></tr> <tr><td>8</td><td>Bbyt</td><td>Bitmap (up to 8 bits)</td></tr> <tr><td>9</td><td>Bsht</td><td>Bitmap (up to 16 bits)</td></tr> <tr><td>10</td><td>Bm24</td><td>Bitmap (up to 24 bits)</td></tr> <tr><td>11</td><td>Blng</td><td>Bitmap (up to 32 bits)</td></tr> <tr><td>12</td><td>Bool</td><td>Boolean (value 0 or 1)</td></tr> <tr><td>13</td><td>DiBy</td><td>Dibyte (both high and low)</td></tr> <tr><td>14</td><td>B448</td><td>Bitfield nybble/nybble/byte</td></tr> <tr><td>15</td><td>Flot</td><td>Floating point</td></tr> </tbody> </table>	Value	Mnemonic	Format	0	Ubyt	Unsigned byte	1	Usht	Unsigned short integer	2		[reserved]	3	Ulng	Unsigned long integer	4	Sbyt	Signed byte	5	Ssht	Signed short integer	6		[reserved]	7	Slng	Signed long integer	8	Bbyt	Bitmap (up to 8 bits)	9	Bsht	Bitmap (up to 16 bits)	10	Bm24	Bitmap (up to 24 bits)	11	Blng	Bitmap (up to 32 bits)	12	Bool	Boolean (value 0 or 1)	13	DiBy	Dibyte (both high and low)	14	B448	Bitfield nybble/nybble/byte	15	Flot	Floating point
Value	Mnemonic	Format																																																			
0	Ubyt	Unsigned byte																																																			
1	Usht	Unsigned short integer																																																			
2		[reserved]																																																			
3	Ulng	Unsigned long integer																																																			
4	Sbyt	Signed byte																																																			
5	Ssht	Signed short integer																																																			
6		[reserved]																																																			
7	Slng	Signed long integer																																																			
8	Bbyt	Bitmap (up to 8 bits)																																																			
9	Bsht	Bitmap (up to 16 bits)																																																			
10	Bm24	Bitmap (up to 24 bits)																																																			
11	Blng	Bitmap (up to 32 bits)																																																			
12	Bool	Boolean (value 0 or 1)																																																			
13	DiBy	Dibyte (both high and low)																																																			
14	B448	Bitfield nybble/nybble/byte																																																			
15	Flot	Floating point																																																			
15 to 12	4	[reserved]																																																			
11 to 08	4	Group code This value is one of the "measurement configuration changed" bit numbers.																																																			
07 to 04	4	Subgroup code This value is the ordinal number (starting at 0) of the subgroup of parameters in the specified group.																																																			
03 to 00	4	Subgroup item code Since a parameter subgroup may contain more than one item, this value identifies the particular item; items are numbered from 0.																																																			

6.8.4 Event-triggered Archives and Accumulator Resets

Each archive file (two for each meter) contains an Archive Options bitmap whose configuration specifies the actions to be scheduled (write archive and/or reset resettable accumulator(s)) when an event occurs (daily or hourly period-end, or most loggable events). Archives and/or resets are scheduled only for enabled meters (with one important clarification; see "Rkv" Notes (page 188)). The actions to be taken upon period-end and those to be taken upon loggable events are configured separately.

Several archive/reset-triggering events can occur simultaneously. In such cases the archive or reset occurs only once (an archive is written only when archivable data has been accumulated for at least one meter scan; additional resets of already-reset accumulators have no effect).

Scheduled accumulator resets are performed at the top of the meter scan. This permits their final values to be inspected/fetched/archived while the AFC rotates its scan among the other meters.

Scheduled archives are written at the top of the meter scan, at its bottom, or between successive scans, depending on the nature of the triggering event. Archives written at the top of the scan are written before any accumulator resets.

6.8.5 Period-end Events

A "period-end" event is detected by the wallclock. There are two such:

- a) "End-of-hour" occurs when the minute of the hour steps into the "End-of-hour minute" of Site Configuration.
- b) "End-of-day" occurs when the minute of the day steps into the "End-of-day minute" of Site Configuration.

A wallclock change that skips forward over an end-of-period minute will cause that period-end to be missed, and a change that skips backward over that minute will cause that period-end to be repeated, so wallclock adjustments should be performed at times well-removed from either end-of-period minute.

Though a period-end event is not recorded in the event log, it does cause archives and resets to be scheduled for all enabled meters according to their configured "period-end" Archive Options. Archives and resets scheduled by period-end are delayed in their action until at least one meter scan has occurred after the event (the archive data accumulation that takes place at the end of the meter scan also records the latest timestamp, so the written archive then reflects the fact that the period-end has occurred).

6.8.6 Loggable Events

The tables below give full details of all events that are recorded in the Event Log. For the Special Events (page 183), columns are:

Tag	Numeric value that identifies the event.
Rkv	Effect on archives and accumulator resets (see next).
Description	Lists: The event name, identifying its triggering condition. Contents and meaning of the old and new value fields. Relevant additional information.

For the Data Point (page 184, page 184, page 186) events, columns are:

Group	Group code.
Sbgp	Subgroup code.
Item	Item code.
Dtyp	Datum type code (mnemonic).
Rkv	Effect on archives and accumulator resets (see next).
Datum point	The corresponding writable Modbus point.

In these tables, the "Rkv" columns specify how archives and accumulator resets are scheduled upon occurrence of the corresponding loggable events.

Column values are:

Value	Meaning
*	Upon this event archives and resets are scheduled according to the configured "event" Archive Options, provided that the applicable meter(s) is(are) enabled. Applicable meters depend upon the event class: (a) Special (non-meter-specific) and Site Datum Point events: All meters. (b) Meter events (including meter-specific Specials): The addressed meter. (c) Stream events: The addressed meter, provided that the addressed stream is active. Scheduled archives are always written before completing any change to data or module state implied by the event; this ensures that the data contributing to an archive is limited to that which was available before the event.
-	This event has no effect on archives and resets.
(n)	Upon this event archives and resets are scheduled as for "*", modified by the conditions and actions given in "Note (n)" in "Rkv" Notes (page 188).

6.8.7 Special Events

Tag	Rkv	Description
0	-	<p>Never Used</p> <p>Value: Always 0.</p> <p>Notes: This entry in the Event Log has never been written.</p> <p>The number of such entries starts at 1999 upon cold start and decreases as events are written until none remain, after which oldest events are overwritten with new ones.</p>
1	-	<p>Event Log Download</p> <p>Value: Number of last-downloaded event.</p> <p>Notes: Triggered by a purge of the Event Log, which marks older events as available to be overwritten by new ones.</p>
2	-	<p>Cold Start</p> <p>Value: Always 0.</p> <p>Notes: This event is obsolete and is never written.</p>
3	(1)	<p>Power-Up</p> <p>Value: "Old" value is the last-saved wallclock from the previous session; "new" value is always 0 (clock not yet set).</p> <p>Notes: The last event written upon restart of the application and before entering the meter scan. This event may be preceded by Checksum Alarm and/or PLC Mode Change events.</p>
4	-	<p>PLC Mode Change</p> <p>Value: PLC mode (0 on line, 1 off line).</p> <p>Notes: Logs changes to PLC connectivity as reported by the backplane procedures. Typically caused by switching the PLC between "run" and "program" modes.</p>
5	-	<p>Checksum Alarm</p> <p>Value: Checksum alarm word (datum type "Bsht").</p> <p>Notes: Logs changes to the checksum alarm bitmaps.</p> <p>Includes site/meter identification (bits 28-24).</p> <p>Upon power-up: Written automatically upon power up when a checksum failure is detected. In this case the event is written even if the bitmap does not change, such as when an affected bit is already set from a previous failure that was never cleared.</p> <p>Upon Modbus write to the bitmap: Records changes to the bitmap only, typically when clearing bits, though setting bits is also permitted.</p>
6	(2)	<p>Wallclock Change</p> <p>Value: Wallclock (packed bitfields).</p> <p>Notes: Triggered when the wallclock is set for the first time, or when it is reset to a value that differs from its current value by five minutes or more. These two cases can be distinguished by the "old value" in the event entry: for the initial setting this value is zero ("clock not set").</p>
7	*	<p>Stream Select</p> <p>Value: Stream number.</p> <p>Notes: Triggered by a "select active stream" meter signal.</p> <p>Includes meter identification (bits 28-24).</p>

6.8.8 Site Data Point Events

Group	Sbgp	Item	DTyp	Rkv	Data point
0	0	0	Bsht	(3)	Site options
1					Site parameter value
	0	0	Flot	*	Barometric pressure
8	n	0	Usht	-	Arbitrary event-logged value "n" ("n" = 0 thru 9)
15					PLC image address (Quantum platform only)
	0	0	Usht	*	Supervisory, get
	1	0	Usht	*	Supervisory, put
	2	0	Usht	*	Wallclock, get & put
	3	0	Usht	*	Modbus gateway, get & put
	4	0	Usht	*	Modbus pass-thru, put
	5	0	Usht	*	Modbus master, get & put

6.8.9 Meter Data Point Events

Group	Sbgp	Item	DTyp	Rkv	Data point
0	0				Process input calibration
		0	Flot	*	Temperature
		1	Flot	*	Pressure
		2	Flot	*	Primary input
		3	Flot	*	Flowing density
		4	Flot	*	Water content
0	1				Process input alarm
		0	Flot	-	Temperature range
		1	Flot	-	Pressure range
		2	Flot	-	Primary input range
		3	Flot	-	Flowing density range
		4	Flot	-	Water content range
1	0				Meter classification
		0	Bsht	*	Meter device and engineering units
		1	Usht	*	Product group
2					Reference conditions
	0	0	Flot	*	Temperature
	1	0	Flot	*	Pressure
3					Meter options
	0	0	BIng	*	Calculation options
	1	0	BIng	(4)	Control options
4					Input scaling
	0				Temperature
		0	Flot	*	Range low end
		1	Flot	*	Range high end

Group	Sbgp	Item	DTyp	Rkv	Data point
		2	Flot	*	Default
		3	Sbyt	*	Module id code
1					Pressure
		0	Flot	*	Range low end
		1	Flot	*	Range high end
		2	Flot	*	Default
		3	Sbyt	*	Module id code
2					Primary input
		0	Flot	*	Range low end
		1	Flot	*	Range high end
		2	Flot	*	Default
		3	Sbyt	*	Module id code
3					Flowing density
		0	Flot	*	Range low end
		1	Flot	*	Range high end
		2	Flot	*	Default
		3	Sbyt	*	Module id code
4					Water content
		0	Flot	*	Range low end
		1	Flot	*	Range high end
		2	Flot	*	Default
		3	Sbyt	*	Module id code
5	0	0	Bm24	*	Analysis component selection map
6	0	0	UIng	*	Pulse input rollover
7					Units
	0	0	B448	*	Primary input (period, quantity, units)
	1	0	Ubyt	*	Mass flow rate period
	2	0	Ubyt	*	Mass flow rate units
	3	0	Ubyt	*	Mass accumulator units
	4	0	Ubyt	*	Energy flow rate period
	5	0	Ubyt	*	Energy flow rate units
	6	0	Ubyt	*	Energy accumulator units
	7	0	Ubyt	*	Volume flow rates period
	8	0	Ubyt	*	Volume flow rates units
	9	0	Ubyt	*	Volume accumulators units
8					Accumulator rollovers
	0	0	UIng	*	Mass
	1	0	UIng	*	Energy
	2	0	UIng	*	Volumes

Group	Sbgp	Item	DTyp	Rkv	Data point
9					Meter parameter value
	0	0	Flot	*	Orifice plate measurement temperature
	1	0	Flot	*	Orifice plate measured diameter
	2	0	Flot	*	Orifice plate coefficient of thermal expansion
	3	0	Flot	*	Meter tube measurement temperature
	4	0	Flot	*	Meter tube measured diameter
	5	0	Flot	*	Meter tube coefficient of thermal expansion
	6	0	Flot	*	Primary input flow threshold
	7	0	Flot	*	Primary input alarm threshold
	8	0	Flot	*	V-cone/Wedge coefficient of discharge
10					[reserved]
11	0				Densitometer
		0	Usht	*	Densitometer type
		1	Flot	*	Calibration temperature
		2	Flot	*	Calibration pressure
		3	Flot	*	Calibration constant K0
		4	Flot	*	Calibration constant K1
		5	Flot	*	Calibration constant K2
		6	Flot	*	Calibration constant 6
		7	Flot	*	Calibration constant 7
		8	Flot	*	Calibration constant 8
		9	Flot	*	Calibration constant 9
		10	Flot	*	Calibration constant 10
		11	Flot	*	Calibration constant 11
		12	Flot	*	Calibration constant 12
		13	Flot	*	Calibration constant 13
		14	Flot	*	Calibration constant 14
		15			PLC image address (Quantum platform only)
	0	0	Usht	*	Meter process input &c, get
	1	0	Usht	*	Meter results, put
	2	0	Usht	*	Meter archive fetch, put

6.8.10 Stream Data Point Events

Group	Sbgp	Item	DTyp	Rkv	Data point
0	0	0	Bsht	*	Stream options
1					Stream parameter value
	0	0	Flot	*	Default relative density (gas) at reference
	1	0	Flot	*	Viscosity
	2	0	Flot	*	Isentropic exponent
	3	0	Flot	*	Default Fpv
	4	0	Flot	*	K/meter factor
	5	0	Flot	*	Default energy content
	6	0	Flot	*	Default reference density (liquid)
	7	0	Flot	*	Default vapor pressure
	8	0	Flot	*	Water density at API reference
	9	0	Flot	*	Default Ctl
	10	0	Flot	*	Default Cpl
	11	0	Flot	*	Shrinkage factor
	12	0	Flot	*	Precalculated alpha
2	0				Meter factor curve
		0	Flot	*	Datum point 1, meter factor
		1	Flot	*	Datum point 1, flow rate
		2	Flot	*	Datum point 2, meter factor
		3	Flot	*	Datum point 2, flow rate
		4	Flot	*	Datum point 3, meter factor
		5	Flot	*	Datum point 3, flow rate
		6	Flot	*	Datum point 4, meter factor
		7	Flot	*	Datum point 4, flow rate
		8	Flot	*	Datum point 5, meter factor
		9	Flot	*	Datum point 5, flow rate
3	0				Analysis mole fraction
					** Because the item code extends into the subgroup field, this can be the only subgroup of group 3! (Pending any future reformat of the Event Id Tag)
		0	Usht	(5)	Component 1, scaled molar fraction
		1	Usht	(5)	Component 2, scaled molar fraction
		2	Usht	(5)	Component 3, scaled molar fraction
		3	Usht	(5)	Component 4, scaled molar fraction
		4	Usht	(5)	Component 5, scaled molar fraction
		5	Usht	(5)	Component 6, scaled molar fraction
		6	Usht	(5)	Component 7, scaled molar fraction
		7	Usht	(5)	Component 8, scaled molar fraction
		8	Usht	(5)	Component 9, scaled molar fraction

Group	Sbgp	Item	DTyp	Rkv	Data point
		9	Usht	(5)	Component 10, scaled molar fraction
		10	Usht	(5)	Component 11, scaled molar fraction
		11	Usht	(5)	Component 12, scaled molar fraction
		12	Usht	(5)	Component 13, scaled molar fraction
		13	Usht	(5)	Component 14, scaled molar fraction
		14	Usht	(5)	Component 15, scaled molar fraction
		15	Usht	(5)	Component 16, scaled molar fraction
		16	Usht	(5)	Component 17, scaled molar fraction
		17	Usht	(5)	Component 18, scaled molar fraction
		18	Usht	(5)	Component 19, scaled molar fraction
		19	Usht	(5)	Component 20, scaled molar fraction
		20	Usht	(5)	Component 21, scaled molar fraction
		21	Usht	(5)	Component 22, scaled molar fraction
		22	Usht	(5)	Component 23, scaled molar fraction
		23	Usht	(5)	Component 24, scaled molar fraction

6.8.11 "Rkv" Notes

- 1 Archives (only, not resets) are forced regardless of configuration, capturing any unarchived data from the previous session.
- 2 Archives and resets are scheduled (immediately, without a "period-end" delay) only for the initial setting of the wallclock; a "five-minute" event causes no scheduling. This ensures capture of any flow that has occurred prior to the initial clock-set.
- 3 Event occurs only when one or more of the following bits are changed:
 - o Bit 2, "Barometric pressure units"
 - o Bit 5, "Process input out of range use last good"
 - o Bit 12, "Analysis is packed in module"
 - o Bit 13, "Analysis is packed over backplane" (1756 and 1769 platforms only)
- 4 A change to Meter Control Options bit 15, "Meter enable", imposes these adjustments to the normally-scheduled archives/resets:
 - o Upon meter enable, cancel any scheduled archives (no data yet to be archived), but leave in place any scheduled resets.
 - o Upon meter disable, cancel any resets (for inspection and so on.; reset will be rescheduled upon subsequent enable), and force archiving of both files regardless of configuration (so that a disabled meter never has any pending unarchived data).
- 5 Events occur only if Meter Control Options bit 10, "Treat analysis as process input", is clear.

6.8.12 Event Numbers

For auditing purposes, each event has a "number" assigned sequentially, starting at 0 for the first event written and increasing up through 65535 then wrapping to 0 again.

An event record properly includes its event number along with the information listed in the preceding sections. To conserve space, and to make transmittal more efficient, the event number is not stored as part of the event record. Instead, the Event Log header contains sufficient information to calculate for any event its event number from the position of its record in the Log and vice versa.

The following procedures use these terms:

Term	Meaning
my_record	Known record position. Input to procedures (A) and (C)
event_number	Desired event number. Output from procedure (A).
Modbus_address	Desired Modbus address. Output from procedure (C).
my_event	Known event number. Input to procedure (B).
record_position	Desired record position. Output from procedure (B).
number_of_records	Maximum number of records. Contents of register 40000. In this version of the AFC "number_of_records" is 1999; however, to be compatible with future versions that may store a different number of events, an application should use the value from the header instead of a constant 1999.
next_record	Next new record position. Contents of register 40001.
next_event	Next new event number. Contents of register 40002.
oldest_event	Oldest event number on file. Contents of register 40003.
oldest_not_downloaded	Oldest event number not yet downloaded. Contents of register 40004.
events_on_file	Total number of events on file. Calculated. This value starts at 0 upon cold start, then, as events are logged, it rises to a maximum of "number_of_records" and stays there.
downloadable_event	Event number of event being downloaded. Calculated.
event_age	The age of the event in question. Calculated. The next event to be written (which of course is not yet on file) has age 0; the newest event already on file has age 1, the next older event has age 2, and so on up to age "number_of_records".

Also in these procedures:

- a) The expression "AND 0x0000FFFF" means "take the low-order 16 bits of the result, discarding all other higher-order bits"; it is equivalent to "(non-negative) remainder upon dividing by 65536" (A traditionally negative remainder that would result from dividing a negative dividend by 65536 must be made positive by subtracting its absolute value from 65536)
- b) The operator ":= " means "assignment"; that is, "assign" the expression on the right to the object on the left by calculating the value of the expression on the right and making the object on the left assume that value. The operator "==" means "is equal to".
- c) Words in all caps and the other arithmetic operators have their expected meanings.
- d) Text enclosed in brackets ("["]") are comments only.

Procedure (A): Calculate event number from record position.

1 Calculate number of events on file.

```
events_on_file := ( next_event - oldest_event ) AND 0x0000FFFF
```

2 Determine whether desired record is on file.

```
IF ( my_record < 0 OR my_record ≥ events_on_file ) THEN  
  [record is not on file]  
  EXIT this procedure
```

3 Calculate age of desired record.

```
event_age := ( next_record - my_record )  
IF ( event_age ≤ 0 ) THEN  
  event_age := event_age + number_of_records
```

4 Calculate event number of desired record.

```
event_number := ( next_event - event_age ) AND 0x0000FFFF
```

Procedure (B): Calculate record position from event number.

1 Calculate number of events on file.

```
events_on_file := ( next_event - oldest_event ) AND 0x0000FFFF
```

2 Calculate age of desired event.

```
event_age := ( next_event - my_event ) AND 0x0000FFFF
```

3 Determine whether desired event is on file.

```
IF ( event_age == 0 OR event_age > events_on_file ) THEN  
  [event is not on file]  
  EXIT this procedure
```

4 Calculate record position of desired event.

```
record_position := ( next_position - event_age )  
IF ( record_position < 0 ) THEN  
  record_position := record_position + number_of_records
```

Procedure (C): Calculate Modbus address of record from record position.

1 Calculate number of events on file.

```
events_on_file := ( next_event - oldest_event ) AND 0x0000FFFF
```

2 Determine whether desired record is on file.

```
IF ( my_record < 0 OR my_record ≥ events_on_file ) THEN  
  [record is not on file]  
  EXIT this procedure
```

3 Calculate Modbus address.

```
Modbus_address := ( my_record * 8 ) + 40008
```

Procedure (D): Download all events not yet downloaded.

The downloading application should download the entire Log, starting at the oldest event not yet downloaded and extending through all newer events.

1 Fetch event number of oldest event not yet downloaded.

```
downloadable_event := oldest_not_downloaded
```

2 Determine whether any more events remain to be downloaded.

```
IF ( downloadable_event == next_event ) THEN  
  [all events have been downloaded]  
  EXIT this procedure
```

3 Download this event.

a) Calculate record number.

```
my_event := downloadable_event  
record_position := { via Procedure (B) }
```

b) Calculate Modbus address.

```
my_record := record_position  
Modbus_address := { via Procedure (C) }
```

c) Download the event with Modbus.

```
Set Modbus Function Code := 4, Read Input Registers  
Set Modbus Number of Registers := 8  
Set Modbus Register Address := Modbus_address  
Execute
```

Copy the returned data to permanent storage

4 Step to next event and loop.

```
downloadable_event := ( downloadable_event + 1 ) AND 0x0000FFFF  
GOTO step 2.
```

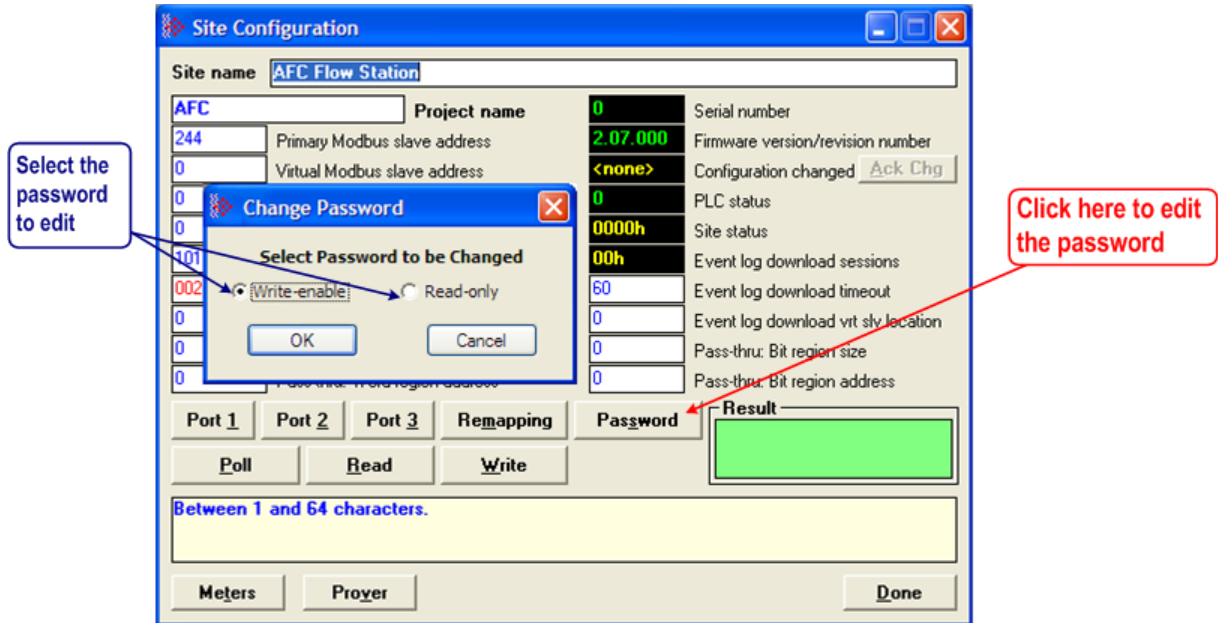
When the download is complete, and the downloaded events have been logged to disk, the AFC should be told of this fact by issuing the "download complete" Site Signal. This signal updates the header to show that all records have been downloaded, unlocking the Log for further events, and (if "Event log unlocked" is clear) posts a "download" event. A download may be performed at any time; it is not necessary to wait for the log-full condition in order to download.

An application that downloads the event log should explicitly include the event number in any copy of the event that it stores in its own database.

6.9 Security (Passwords)

The passwords are intended for interrogation by application software in order to verify an operator's authorization to make configuration changes and to view measurement results. The passwords are resident in the module so that different operators using different copies of the application software must use the same password. Passwords cannot be retrieved in "Hard Password" mode. The password protection is not used by default.

Passwords can be numbers between -32768 and 32767. For example, 1234. A password of 0 (zero) is interpreted as "No password present".



The module supports two passwords: Write-Enable and Read-Only. Each password is enabled when you write a non-zero value to the corresponding register.

Password	Holding Register Address	Description
Write-Enable	9	Protects the module from write operations from the AFC Manager
Read-Only	19	Protects the module from read or write operations from the AFC Manager

The following table shows how the passwords affect the AFC Manager operation depending on the values that you configure:

Protection Level	Read-Only Password	Write-Enable Password	Read Operation: Requires Authorization?	Write Operation: Requires Authorization?
No protection	Zero	Zero	No	No
Write Protection	Zero	Non-zero	No	Yes (Use Write-Enable password)
Read and Write Protection	Non-zero	Zero	Yes (Use Read-Only password)	Yes (Use Read Only password)
Read and Write Protection	Non-zero	Non-zero	Yes (Use Read-Only or Write-Enable password)	Yes (Use Write-Enable password)

Each port can be assigned to different password protection levels. Refer to the Port Configuration Section for more information about this topic.

6.9.1 Hard Password

The hard password feature offers further protection against unauthorized access to the module.

If the Hard Password option is cleared, these registers can be read either from an external Modbus device, from the processor or using the Modbus Master interface in the AFC Manager. This operation mode is called "Soft Password" mode. It is then the responsibility of a compatible application (such as AFC Manager) to verify the password given by the operator against those fetched from the module in order to determine the access granted.

If the Hard Password option is selected, a read of a password register will return zero regardless of the password's actual value. In this case, read or write access is obtained by writing a candidate password to the Password Test register (register 18), the module itself verifies the password, and the access granted is determined by reading back that same register 18 (called the Accessed Port and Authorization register when read) and examining its contents. The access is granted to the port over which the request was made; other ports remain unaffected. If the port remains idle with no Modbus activity for two minutes, then the granted access is removed and can be regained only by writing a new password to the test register. For highest security, you can explicitly revoke your own password-obtained authorization before it times out by writing zero to the Password Test register.

Access granted by password, whether Soft or Hard, is to the module as a whole, including the password registers themselves. That is, in order to change a stored Hard password you must first obtain write access to the module by giving the correct Write-Enable password. However, some registers are exempt from authorization. There are a very few registers that are exempt from write authorization and are always writable; the Password Test register 18 is one such for the obvious reason. Similarly, some registers are exempt from read authorization and are always readable; they include most of the first 20 holding registers, including the Firmware Product and Group codes in registers 0 and 1 (so an application like AFC Manager can learn whether it is talking to an AFC without being trapped in a catch-22), the Site Status in register 6 (so the application can learn whether the password mode is Soft or Hard and verify the operator's password entry using the proper method), and the Accessed Port and Authorization register 18 (so the application can learn whether access was granted in Hard-password mode even if the wrong read password was entered).

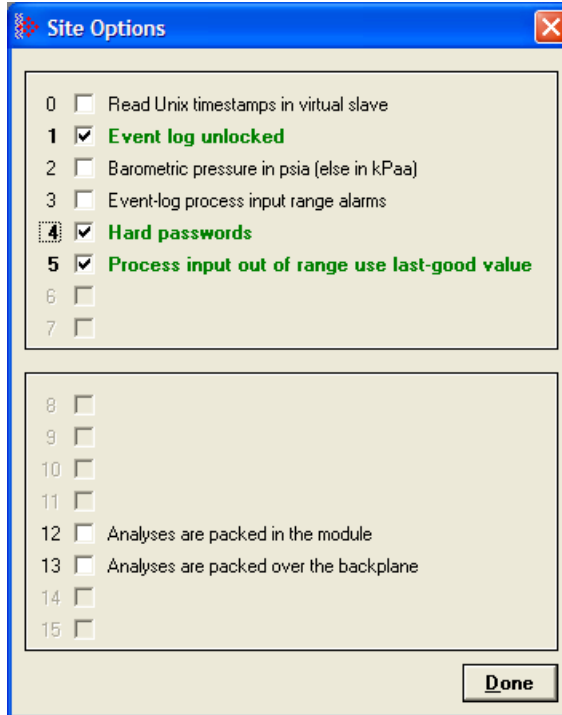
The Accessed Port and Authorization register is a bit-mapped word defined as follows:

Bits	Description
0 to 3	The number of the accessing port (0 for Modbus Gateway)
4	Read Authorization Waived
5	Write Authorization Waived
6	Read Access Granted
7	Write Access Granted
8 to 15	Reserved

A waived authorization means that password entry is not required for this action even if a non-zero password has been configured. Authorization waivers are configured separately for each port, so, for example, a SCADA system connected to port 2 can be allowed to read measurement results without having to supply a password while an operator connecting AFC Manager to port 1 still must enter the correct password. The backplane is always given both waivers, so the PLC never has to supply a password.

To set a hard password in AFC Manager:

- 1 Open the Site Configuration Dialog box
- 2 Click in the Site Options field. This action opens the Site Options dialog box
- 3 Select (check) option 4, Hard Passwords



When this option is selected, any authorization granted using Hard Passwords times out after two minutes of inactivity, and the user will be required to re-enter the password to continue.

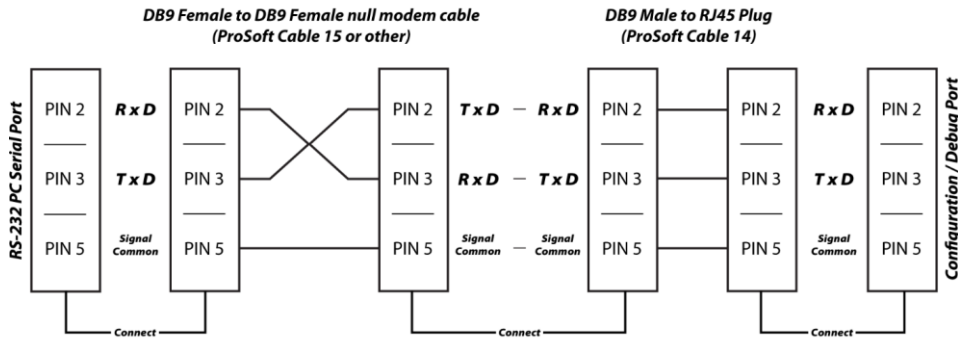
6.10 Cable Connections

The application ports on the MVI69-AFC module support RS-232, RS-422, and RS-485 interfaces. Please inspect the module to ensure that the jumpers are set correctly to correspond with the type of interface you are using.

Note: When using RS-232 with radio modem applications, some radios or modems require hardware handshaking (control and monitoring of modem signal lines). Enable this in the configuration of the module by setting the *UseCTS* parameter to 1.

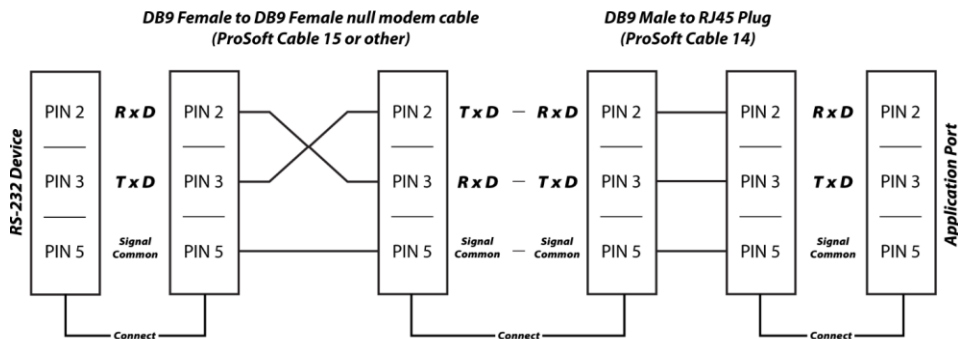
6.10.1 RS-232 Configuration/Debug Port

This port is physically an RJ45 connection. An RJ45 to DB-9 adapter cable is included with the module. This port permits a PC-based terminal emulation program to view configuration and status data in the module and to control the module. The cable pinout for communications on this port is shown in the following diagram.



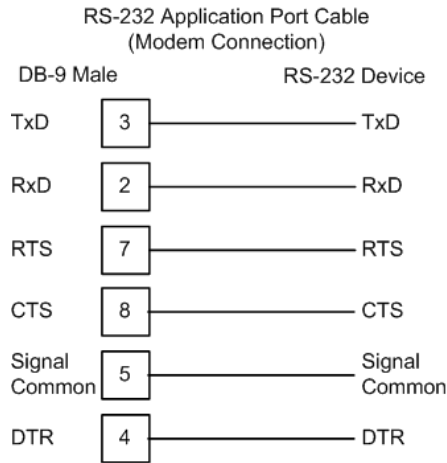
6.10.2 RS-232 Application Port(s)

When the RS-232 interface is selected, the use of hardware handshaking (control and monitoring of modem signal lines) is user definable. If no hardware handshaking will be used, here are the cable pinouts to connect to the port.



RS-232: Modem Connection (Hardware Handshaking Required)

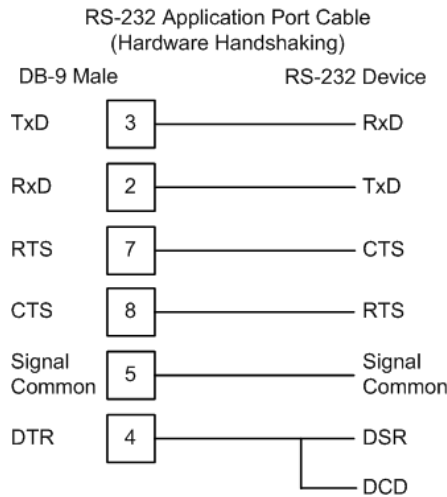
This type of connection is required between the module and a modem or other communication device.



The *Use CTS Line* parameter for the port configuration should be set to **Y** for most modem applications.

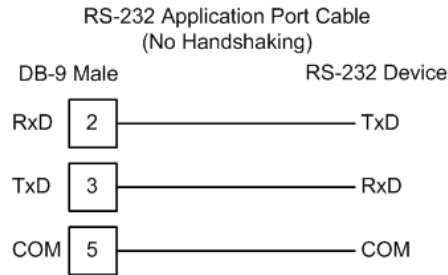
RS-232: Null Modem Connection (Hardware Handshaking)

This type of connection is used when the device connected to the module requires hardware handshaking (control and monitoring of modem signal lines).

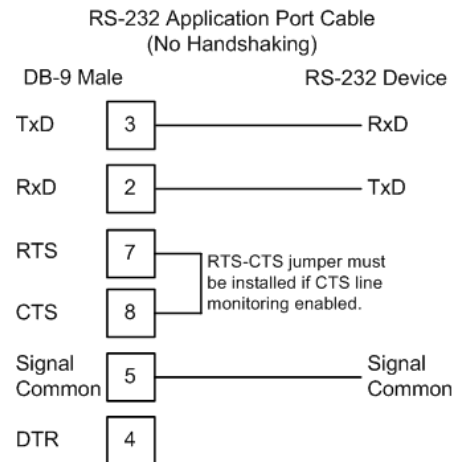


RS-232: Null Modem Connection (No Hardware Handshaking)

This type of connection can be used to connect the module to a computer or field device communication port.

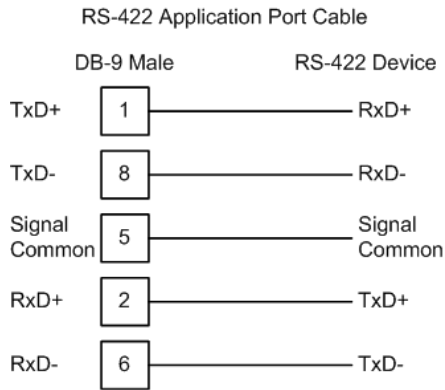


Note: For most null modem connections where hardware handshaking is not required, the *Use CTS Line* parameter should be set to **N** and no jumper will be required between Pins 7 (RTS) and 8 (CTS) on the connector. If the port is configured with the *Use CTS Line* set to **Y**, then a jumper is required between the RTS and the CTS lines on the port connection.



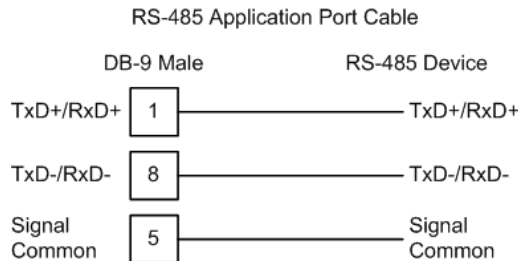
6.10.3 RS-422

The RS-422 interface requires a single four or five wire cable. The Common connection is optional, depending on the RS-422 network devices used. The cable required for this interface is shown below:



6.10.4 RS-485 Application Port(s)

The RS-485 interface requires a single two or three wire cable. The Common connection is optional, depending on the RS-485 network devices used. The cable required for this interface is shown below:

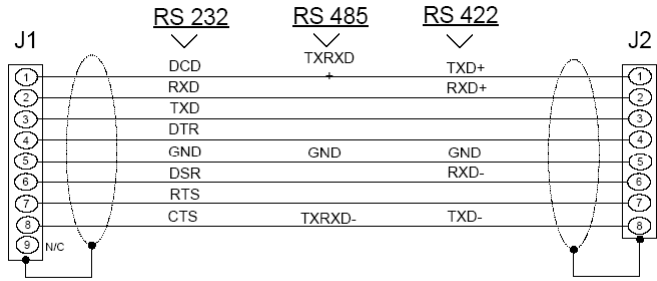
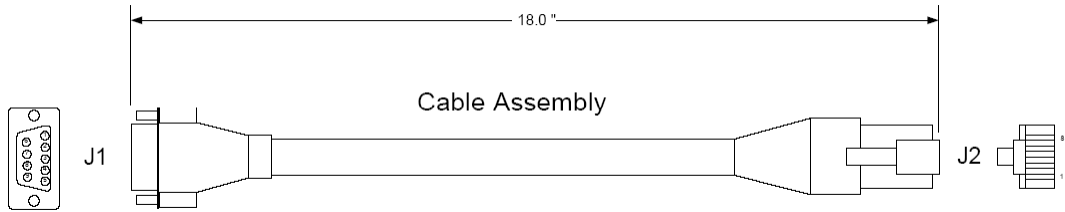


Note: Terminating resistors are generally not required on the RS-485 network, unless you are experiencing communication problems that can be attributed to signal echoes or reflections. In these cases, installing a 120-ohm terminating resistor between pins 1 and 8 on the module connector end of the RS-485 line may improve communication quality.

RS-485 and RS-422 Tip

If communication in the RS-422 or RS-485 mode does not work at first, despite all attempts, try switching termination polarities. Some manufacturers interpret + and -, or A and B, polarities differently.

6.10.5 DB9 to RJ45 Adaptor (Cable 14)



Wiring Diagram

6.11 Frequently Asked Questions

The Automatic Flow Computer (AFC) is a powerful rack flow computer solution for PLC platforms. The design intent of the module is to simplify the setup and maintenance of a meter installation. With this in mind, the sample ladder logic was created to accomplish the following:

- Pass meter run variables to the module.
- Return meter results to the processor.
- Allow individual meters to be enabled or disabled.
- Allow resets of individual meter runs.
- Allow transfer of a new gas analysis to an individual meter run.

Actual meter setup includes units of measure setup, range checking for input variables, and the type of meter being used. This setup is handled by the AFC Manager software. The intended design is to have the processor only handle the variables of an actual process and the AFC Manager handle the setup and configuration of necessary meter variables.

The sample ladder logic included with the system is intended to fulfill this requirement and works for many applications. Should you feel that your application requires more than this, then a very intimate knowledge of the operations of the module are required to be successful in the implementation of the application. It is highly recommended that the sample be used as a starting point for any application.

6.11.1 How does the module work?

Ignoring the fundamentals of a meter run, the module's operation is very simply divided into two operations, those being the transfer of data from the Processor to the module (variables as a rule) and the second being the transfer of data from the module to the Processor (results).

Refer to the Backplane section of the AFC User Manual for your module for more information on backplane operation.

6.11.2 Why should I use the AFC Manager?

The AFC Manager should be used to configure the module project parameters (Site Configuration) and each meter (Meter Configuration).

Once your project is up and running, you can also use the AFC Manager to monitor each meter run (Meter Monitor), archives, and events.

6.11.3 Why can't the AFC Manager connect to the module?

Check the cable used in your project: a null-modem cable should be used to connect the module with the local PC serial port. Make sure that the baud rate, data bits, mode, parity and primary slave address are the same (both in PC and module).

If you change the primary slave address and later forget the new address, the module will not establish communications. You must read the primary slave address value (address 100) over the backplane using the Modbus Gateway Transaction Block.

6.11.4 Why do I have to enable or disable a meter?

A meter channel will only perform flow calculation if it is enabled. For performance reasons you should disable all meter channels that are not being used. You cannot change a meter type and/or product group for a meter channel that is currently enabled.

6.11.5 Why does the card not calculate results, or why did it stop calculating results?

This could be caused by a couple of things.

- 1 The first thing to check is that the module actually received a clock. If the card does not get a clock it will not be able to schedule storage of historical records.
- 2 The next possibility is that the meter is not enabled or some parameter for the run is not correct. Check to see if the run is enabled and that no errors exist in configuration or data for the run in question. Check for alarms arising from the calculations. The AFC Manager software can be a great help with this as it will highlight problem areas.

6.11.6 What is the Virtual Modbus Slave?

The AFC Modbus database can be accessed using the Primary Modbus Slave address. More than 100,000 registers may be accessed using this slave.

You may want to use certain values from the Modbus database in a different order than the one presented in the Primary Modbus Slave. One example is if you want to poll certain values from the Modbus database using a Modbus master device in the field. Instead of using several commands to poll from different locations in the Modbus database, it is better to remap these values to other locations in order to optimize the master polling.

This is the reason the AFC module offers a second slave: the Virtual Modbus Slave. Using the AFC Manager software, you can remap up to 20,000 registers from the Primary Modbus Slave in any order. The Virtual Modbus Slave Address must be configured using the AFC Manager software (Site Config dialog box).

The Virtual Modbus Slave is also used when using the Modbus Pass-Thru function block.

6.11.7 How does the AFC Manager transfer the configuration to the module?

You can configure the site and meter parameters at the local PC saving the project as a .AFC file. You may then download the configuration by clicking on **Project / Download Configuration**. In this case, all configuration will be downloaded from the local PC to the module, except for the Virtual Slave Re-mapping (must be written separately).

Once you download the entire configuration, you may perform smaller adjustments (Site Configuration and Meter Configuration) by clicking on the Write button.

6.11.8 What is the password used for?

The password protects the module from any changes to "sealable" parameters. Sealable parameters directly affect measurement calculations (for example, orifice diameter, or K-factor).

The password is stored in the module so different computers should always use the same password.

6.11.9 Why do I receive an Illegal Data Value warning when I try to write a meter configuration or download the entire configuration to the module?

Follow these steps:

- Ensure that any parameters you had changed (from the default configuration) are acceptable according to applicable standards. The white rectangle (Site Configuration and Meter Configuration) shows the correct range of values for each parameter.
- The module will not accept a downloaded configuration that changes the meter type and/or the product group of a meter that is currently enabled. Disable the meter first, then proceed with the meter download.
- Look at the number of events currently stored in the module. You can check this using *Monitor / Event Log* and then click on the Read button. If the *number of events not yet downloaded* is 1999 it means that the event log is full. In this case, if the project also has the *event log unlocked* option clear, the module will not accept any further configuration downloads generating the *Illegal Data Value* at any attempt. Delete all events from the module event buffer (refer to the Event Log section). You may want to select (check) the *Event Log Unlocked* check box. This setting allows the module to overwrite the oldest event from the buffer when the buffer is full.

6.11.10 Why is the Molar Analysis button disabled?

In order to transfer the molar analysis values between the module and the local computer, it is required that the module's configuration and the configuration at the local computer should match. In order to accomplish this, you can perform either a **Meter Configuration / Read** or a **Meter Configuration / Write** operation.

6.11.11 Why does the AFC Manager show a "Communication Timeout" warning?

The communication parameters for the AFC Manager and the module should match. Look at the communication parameters and cables (RS-232 null-modem). Also ensure that the setup jumper on the module is OFF.

6.11.12 What is the difference between Net Accumulator and Gross Accumulator?

The module initially calculates the Gross Accumulator value. It then uses the Gross Accumulator value and corrects it for pressure and temperature before calculating the Gross Standard Accumulator value.

For Gases, Gross Standard Accumulator = Net Accumulator

For Liquids, Gross Standard Accumulator - Water = Net Accumulator

6.11.13 What are the accumulator's totalizer and residue values?

The totalizer is the integer part and the residue is the fractional part. The accumulator will be calculated by:

Accumulator = Totalizer + Residue

6.11.14 Do I have to enter all molar concentrations for the gas product?

Yes, the module uses the Detail Characterization Method that requires all molar concentration values.

6.11.15 Can I update the molar concentration values dynamically?

Yes, if the values are generated from a gas chromatograph you can update these values from the processor to the module (via backplane). Refer to the module's user manual for more information about this subject.

6.11.16 Why do the accumulator values not update?

Follow these steps:

- 1 Check if the Wallclock is running. The Wallclock should be set every time the module powers up by ladder logic. If the Wallclock is not running, some very early versions of the AFC will not perform the applicable calculation.
- 2 Determine if the meter has an alarm using the Meter Monitor dialog box. If the alarm field is red, it indicates that the meter has at least one alarm.
- 3 Determine if the meter is enabled. If the meter is not enabled, it will not perform the applicable calculation.
- 4 Look at the input variables in the AFC Manager. Make sure the values that are being copied from the processor match the input variables displayed at the AFC Manager Meter Monitor dialog box.

6.11.17 What is the Wallclock?

The Wallclock is the internal module clock that is used by the module to perform the applicable calculation. Typically, the Wallclock will be copied from the processor at every power up operation, otherwise the module will not perform time-of-day-dependent calculations.

6.11.18 Can I read the Primary (or Virtual) Slave values using the AFC Manager?

Yes, the Modbus Master interface (**Communications / Modbus Master**) allows you to easily read (or write) to any register in both slaves.

6.11.19 When are the archives generated?

There are two types of archives: the *daily* archives (which are generated once a day) and the *hourly* archives (which are generated once a hour). The Site Configuration dialog box has two parameters that allow you to configure when the archives will be generated:

- End-of-Day minute = the minute of the day when the daily archives will be written
- End-of-Hour minute = the minute of the hour when the hourly archives will be written

7 Support, Service & Warranty

In This Chapter

- ❖ Contacting Technical Support207

7.1 Contacting Technical Support

ProSoft Technology, Inc. is committed to providing the most efficient and effective support possible. Before calling, please gather the following information to assist in expediting this process:

- 1 Product Version Number
- 2 System architecture
- 3 Network details

If the issue is hardware related, we will also need information regarding:

- 1 Module configuration and associated ladder files, if any
- 2 Module operation and any unusual behavior
- 3 Configuration/Debug status information
- 4 LED patterns
- 5 Details about the serial, Ethernet or Fieldbus devices interfaced to the module, if any.

Note: For technical support calls within the United States, an emergency after-hours answering system allows 24-hour/7-days-a-week pager access to one of our qualified Technical and/or Application Support Engineers. Detailed contact information for all our worldwide locations is available on the following page.

Internet	Web Site: www.prosoft-technology.com/support E-mail address: support@prosoft-technology.com
Asia Pacific (location in Malaysia)	Tel: +603.7724.2080 E-mail: asiapc@prosoft-technology.com Languages spoken include: Chinese, English
Asia Pacific (location in China)	Tel: +86.21.5187.7337 x888 E-mail: asiapc@prosoft-technology.com Languages spoken include: Chinese, English
Europe (location in Toulouse, France)	Tel: +33 (0) 5.34.36.87.20 E-mail: support.EMEA@prosoft-technology.com Languages spoken include: French, English
Europe (location in Dubai, UAE)	Tel: +971-4-214-6911 E-mail: mea@prosoft-technology.com Languages spoken include: English, Hindi
North America (location in California)	Tel: +1.661.716.5100 E-mail: support@prosoft-technology.com Languages spoken include: English, Spanish
Latin America (Oficina Regional)	Tel: +1-281-2989109 E-Mail: latinam@prosoft-technology.com Languages spoken include: Spanish, English
Latin America (location in Puebla, Mexico)	Tel: +52-222-3-99-6565 E-mail: soporte@prosoft-technology.com Languages spoken include: Spanish
Brasil (location in Sao Paulo)	Tel: +55-11-5083-3776 E-mail: brasil@prosoft-technology.com Languages spoken include: Portuguese, English

Warranty Information

For complete details regarding ProSoft Technology's TERMS & CONDITIONS OF SALE, WARRANTY, SUPPORT, SERVICE AND RETURN MATERIAL AUTHORIZATION INSTRUCTIONS please see the documents on the Product DVD or go to www.prosoft-technology.com/warranty

Documentation is subject to change without notice.

Index

1

- 1) Meter Type = Differential & Product Group = Gas • 48, 54

2

- 2) Meter Type = Differential & Product Group = Liquid • 48, 54

3

- 3) Meter Type = Linear & Product Group = Gas • 49, 55

4

- 4) Meter Type = Linear & Product Group = Liquid • 49, 55

A

Accessing the Data • 128
Accumulator Totalizer and Residue • 142
Accumulator Types • 143
Accumulators • 142
AFC Modbus Address Space • 127
AFC Response to an OBA • 40
Alarm Status • 70
App Status LED • 96
Archive Accumulators • 146
Archive Data Format • 171
Archive Fetch • 79
Archive Generation • 148
Archive Locations • 159
Archive Monitor • 150
Archive Options • 158
Archive Order • 155
Archive Overview • 148
Archive Reports • 170
Archive Types • 155
Archives • 148
Archiving and Event Log • 116
Audit Scan • 101

B

Basic Metering According to Meter type • 114
Battery Life Advisory • 3
BBRAM LEDs • 96
Block IDs (Block Sequence Numbers) • 29
BP Act and P1, P2, or P3 • 96

C

Cable Connections • 137, 196

Calculation Results - Input • 75
Can I read the Primary (or Virtual) Slave values using the AFC Manager? • 205
Can I update the molar concentration values dynamically? • 204
Checking Meter Alarms • 56
Checksum Alarms • 101
Communication Parameters • 137
CompactLogix Sample Ladder File • 37
Compressibility Factor F • 117
Configurable Archive Registers • 160
Configurable Options • 109
Configuration Modification Lockout and Seal • 12
Configuring the RSLinx Driver for the PC COM Port • 67
Connect your PC to the Module • 85
Connecting Your PC to the Processor • 65
Contacting Technical Support • 207
Converting a Project • 90
Correction Factors According to Product Phase • 114
Creating Your Application • 94

D

Data Archiving • 120
Date Word • 172
DB9 to RJ45 Adaptor (Cable 14) • 200
Density Correction • 116
Density Units • 123
Diagnostics and Troubleshooting • 95
Differential (Orifice) Meter Overview • 125
Disable Meter • 46
Do I have to enter all molar concentrations for the gas product? • 204
Downloading the Project to the Module • 92
Downloading the Sample Program to the Processor • 66

E

Editing the Archive Structure • 166
Enable Meter • 46
Enable/Disable Meters • 72
Enable/Disable status • 69
Enable/Disable Status • 44
Event Id Tag • 179, 180
Event Log Function • 120
Event Log Structures • 179
Event Numbers • 189
Events • 101, 178
Event-triggered Archives and Accumulator Resets • 181
Example • 140, 142
Example 1 • 177
Example 2 • 177
Extended Archives • 167

F

Flow Rate Integration with Gas Product • 175

Flow Rate Integration with Liquid Product • 176
Frequently Asked Questions • 148, 201

G

Gas • 114
Gas Product Overview • 122
Gas Pulse Measurement • 114
General Features • 119
General Specifications • 107

H

Hard Password • 193
Hardware Specifications • 112
How does the AFC Manager transfer the configuration to the module? • 203
How does the module work? • 201
How does the sample ladder work? • 28

I

Important Installation Instructions • 2
Information for Users of AFC Manager Versions Older Than 2.01.000 • 164
Input (Transaction) Block Array Definition • 41
Input Block Array • 38
Input Block Length and Format Alarm • 43
Input/Output Blocks for Data Transfer • 37
Input/Output Transactions • 39
Install AFC Manager • 86
Installing the Module in the Rack • 18
Introduction • 9, 25

L

Ladder Logic Implementation • 23
Linear (Pulse) Meter Overview • 124
Liquid • 114
Liquid Correction Factor Details • 116
Liquid Product Overview • 114, 123
Loading an Existing project • 89
Loggable Events • 182

M

Markings • 3
Measurement Standards • 113
Measurement Units • 120
Measuring Water Diluent • 124
Meter Alarms • 97
Meter Calculation Results • 53
Meter Channels • 117
Meter Data Point Events • 182, 184
Meter Process Variables • 47
Meter Profile • 45
Meter Scan Time • 119
Meter Signals • 77
Meter Status • 69
Meter Type • 71
Meter-relative Data • 129
MicroLogix Sample Ladder File • 69
Modbus Address Examples • 129

Modbus Address References • 128
Modbus Addressing Common to Both Primary and Virtual Slaves • 133
Modbus Communication • 137
Modbus Database • 127
Modbus Gateway • 61, 82
Modbus Interface • 109
Modbus Master • 57, 81, 139
Modbus Pass-Through • 141
Modbus Pass-Thru • 61, 80
Modbus Port configuration • 135
Module Configuration • 85, 123
Module Initialization • 94
Molar Analysis (For Gas Product Only) • 50
Molar Analysis Block • 34
Molar Analysis Update • 76
Multiple Meter Accumulators • 119
MVI (Multi Vendor Interface) Modules • 2
MVI69-AFC Module • 11

N

Net Accumulator Calculation • 124, 147
Non-Resettable Accumulators • 143
Non-Standard Reference Conditions • 115

O

On-line Communication & Configuration • 108
Orifice (Differential) Meter with Gas Product • 173
Orifice (Differential) Meter with Liquid Product • 174
Orifice (Include V-cone)
Uses AGA3 1992 / ISO 5167. • 114
Output (Transaction) Block Array Definition • 39
Output Block Array • 37
Output Block Length • 40
Output Function Blocks (OFB) • 40
Overview • 25

P

Package Contents • 16
Period-end Events • 181
Pinouts • 2, 196, 200
Port Options • 138
Pre-defined Header • 172
Pressure Correction Factor, CPL • 116
Primary Input = Differential Pressure • 126
Primary Input = Flow Rate • 126
Primary Input = Pulse Count • 124
Primary Input = Pulse Frequency • 124
Primary Slave • 128
Printing the Configuration Report • 90
Process Block • 32
Process Calculation • 73
Process Input Scaling • 120
Process Variable Interface • 119
Process Variables
Output • 73
Product Batching • 119
Pulse
Both Gas and Liquid • 114

Pulse (Linear) Meter with Gas Product • 174
Pulse (Linear) Meter with Liquid Product • 175
Pulse Count • 74
Pulse Frequency Integration with Gas Product • 175
Pulse Frequency Integration with Liquid Product • 176

R

Reference • 107
Reports • 108
Reset from AFC Manager • 144
Reset from Ladder Logic • 145
Reset Resettable Accumulator • 63
Reset Upon Archive Period End or Reset Upon Event • 145
Reset When the Accumulator Rollover Value is Reached • 145
Resettable Accumulators • 143
Resetting Configuration Parameters • 91
Retrieving Extended Archives • 167
RS-232
 Modem Connection (Hardware Handshaking Required) • 197
 Null Modem Connection (Hardware Handshaking) • 197
 Null Modem Connection (No Hardware Handshaking) • 198
RS-232 Application Port(s) • 196
RS-232 Configuration/Debug Port • 196
RS-422 • 199
RS-485 and RS-422 Tip • 199
RS-485 Application Port(s) • 199

S

Scratchpad • 130
Sealable Parameters • 12, 110
Security (Passwords) • 192
Select Stream (Firmware version 2.05.000 or later) • 52
Sentinel & Anchor (Transaction Number) • 40
Setting Jumpers • 17
SI units: • 117
Site Data Point Events • 182, 184
Site Status • 44, 69
Special Events • 182, 183
Start Here • 15
Starting a New Project • 88
Starting AFC Manager • 87
Status • 69
Stream Data Point Events • 182, 187
Support, Service & Warranty • 207
Supported Meters • 111
System Requirements • 16, 86

T

Temperature Correction Factor CTL • 116
Temperature Correction Factor, CTL • 115
The Event Log • 178

Time Word • 172
Timestamp Date and Time Format • 171
To use a densitometer • 123

U

Update Notice • 10
US units: • 117
User LEDs • 96
Using AFC Manager • 88

V

Verifying Correct Operation • 93
Virtual Slave • 93, 130
Virtual Slave Example Application • 131

W

Wallclock • 43
WallClock • 73
Warnings • 2
Water % • 74
Water Content of Liquids • 115
Wedge Meter Applications • 111, 127
What are the accumulator's totalizer and residue values? • 204
What is the difference between Net Accumulator and Gross Accumulator? • 204
What is the password used for? • 203
What is the Virtual Modbus Slave? • 202
What is the Wallclock? • 205
When are the archives generated? • 205
Why can't the AFC Manager connect to the module? • 201
Why do I have to enable or disable a meter? • 202
Why do I receive an Illegal Data Value warning when I try to write a meter configuration or download the entire configuration to the module? • 203
Why do the accumulator values not update? • 205
Why does the AFC Manager show a • 204
Why does the card not calculate results, or why did it stop calculating results? • 202
Why is the Molar Analysis button disabled? • 204
Why should I use the AFC Manager? • 201
Write Hourly/Daily Archive • 64

Y

Your Feedback Please • 2