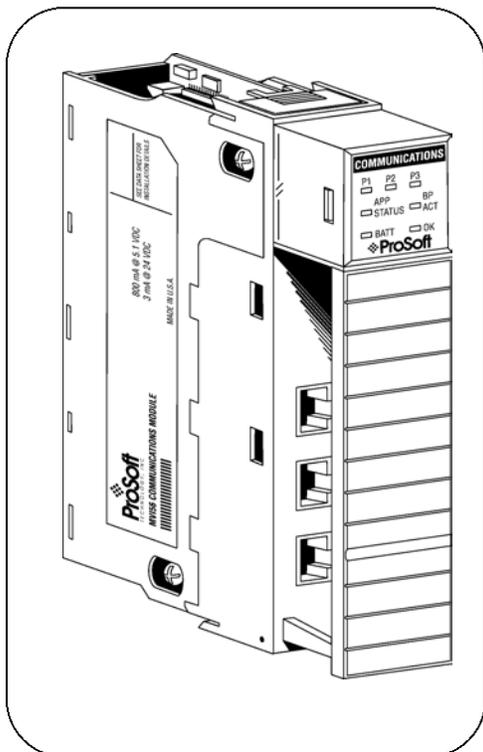


MVI56



MVI56-MCM Plataforma ControlLogix Módulo de Comunicación Modbus

“This is not the latest revision.
Please refer to the English version
of this manual.”

**MANUAL
DEL USUARIO**
Julio 1, 2004

ProSoft
TECHNOLOGY

Por Favor, lea la Siguiete Notificación

Para la aplicación exitosa de éste módulo, se requiere un conocimiento práctico razonable del funcionamiento de PLC's y SLC's Allen Bradley, su hardware y la aplicación para la cual será utilizado el módulo MVI56-MCM. Por esta razón, es importante que quienes sean responsables de su implementación, comprendan que combinando ésta información se cubrirán las necesidades de la aplicación sin exponer al personal o a los equipos a condiciones de trabajo inseguras o inapropiadas.

Este manual se ha provisto para asistir al usuario. Cada tentativa ha sido hecha para asegurar que la información provista es precisa y refleja de manera cierta los requerimientos de instalación de éste producto. De manera de asegurar un completo entendimiento de la operación de éste producto, el usuario debería leer toda la documentación Allen Bradley aplicable a la operación del hardware Allen Bradley.

Bajo ninguna condición, Prosoft Technology, Inc. será responsable o estará obligado por daños directos o indirectos que resulten del uso o aplicación de éste producto.

Queda terminantemente prohibida la reproducción completa o parcial del contenido de éste manual sin la autorización por escrito de Prosoft Technology, Inc.

La información contenida en éste manual está sujeta a cambios sin notificación y no representa ningún acuerdo por parte de Prosoft Technology, Inc. Mejoras y/o cambios a éste manual o el producto pueden ser realizadas en cualquier momento y sin previo aviso. Estos cambios serán hechos periódicamente para corregir imprecisiones técnicas o errores tipográficos.

ProSoft Technology, Inc.

1675 Chester Avenue

Fourth Floor

Bakersfield, CA 93301

(661) 716-5100

(661) 716-5101 Fax

www.prosoft-technology.com

Tabla de Contenido

Tabla de Contenido	iii
1 Introducción	1
1.1 Conceptos Generales	1
1.2 Inicializando el Módulo	1
2 Entendiendo la Arquitectura	3
2.1 Lazo Lógico Principal	4
2.2 Procesador ControlLogix fuera del Modo Run	4
2.3 Transferencia de Datos en el Backplane	4
2.4 Transferencia Normal de Datos	6
2.4.1 Bloques de Lectura	6
2.4.2 Bloques de Escritura	8
2.5 Configuración de la Transferencia de Datos	8
2.5.1 Datos de Configuración del Módulo	9
2.6 Lista Maestra de Datos de Comando	10
2.7 Bloques de Status del Dispositivo Esclavo	11
2.8 Bloques de Comandos de Control	14
2.8.1 Comandos de Eventos	14
2.8.2 Comandos de Control	15
2.8.3 Configuración de Escritura	16
2.8.4 Arranque en Caliente	17
2.8.5 Arranque en Frío	17
2.9 Bloques de Control Pass-Through	17
2.9.1 Bloques de Control Pass-Through no Formateados	17
2.9.2 Bloques de Control Pass-Through Formateados	19
2.9.2.1 Código de Función 5	19
2.9.2.2 Códigos de Función 6 y 16	19
2.9.2.3 Código de Función 15	20
Flujo de Datos Entre el Módulo MVI56-MCM y el Procesador ControlLogix	21
2.9.3 Modo de Driver Esclavo	21
2.9.4 Modo de Driver Maestro	23
2.9.4.1 Lista Maestra de Comandos	24
3 Modificando la Configuración del Módulo	27
3.1 Encendido	27
3.2 Cambiando Parámetros Durante la Operación	27
3.3 Inicialización del Módulo	28
3.4 Objetos de Datos del Módulo (MCMóduloDef)	34
3.4.1 Objetos de Configuración	35
3.4.1.1 Parámetros de Transferencia de Datos (MCMódulo)	36
3.4.1.2 Parámetros del Puerto Modbus (MCMPort)	37
3.4.1.3 Comandos Maestros Modbus (MCMCmd)	38
3.4.2 Objetos de Status (MCMInStat)	39
3.5 Objetos de Datos del Usuario	40
3.6 Control y Status para Interrogar al Esclavo	41

Table of Contents

3.7	Datos de Mensaje Modbus.....	41
4	Modificando la Lógica de Escalera Ejemplo.....	43
4.1	Energización (Power Up).....	43
4.2	Rutina Principal.....	44
4.3	Tarea de Datos de Lectura.....	45
4.4	Tarea de Datos de Escritura.....	50
5	Diagnóstico y Troubleshooting.....	57
5.1	Leyendo los Datos de Status del Módulo.....	57
5.1.1	Hardware Requerido.....	57
5.1.2	Software Requerido.....	58
5.1.3	Utilizando el Puerto.....	58
5.1.4	Opciones de Menú.....	59
5.1.4.1	A=Analizador de Datos.....	59
5.1.4.1.1	1=Selección del Puerto 1.....	60
5.1.4.1.2	2= Selección del Puerto 2.....	60
5.1.4.1.3	5=Ticks de 1 mSeg.....	60
5.1.4.1.4	6= Ticks de 5 mSeg.....	60
5.1.4.1.5	7= Ticks de 10 mSeg.....	60
5.1.4.1.6	8= Ticks de 50 mSeg.....	60
5.1.4.1.7	9= Ticks de 100 mSeg.....	60
5.1.4.1.8	0= Ticks de 0 mSeg.....	60
5.1.4.1.9	H=Format Hexadecimal.....	60
5.1.4.1.10	A=Formato ASCII.....	60
5.1.4.1.11	B=Start.....	60
5.1.4.1.12	S=Stop.....	61
5.1.4.1.13	M = Menú Principal.....	61
5.1.4.2	B=Estadística de Transferencia de Bloques.....	61
5.1.4.3	C=Configuración del Módulo.....	62
5.1.4.4	D=Visualizació de la Base de Datos Modbus.....	62
5.1.4.5	0-9=Páginas de Registro 0-9000.....	62
5.1.4.6	S=Mostrar de Nuevo.....	63
5.1.4.6.1	- = Retroceder 5 Páginas.....	63
5.1.4.6.2	P = Página Anterior.....	63
5.1.4.6.3	+ = Avanzar 5 Póginas.....	63
5.1.4.6.4	N = Próxima Página.....	63
5.1.4.6.5	D = Display Decimal.....	63
5.1.4.6.6	H = Display Hexadecimal.....	63
5.1.4.6.7	F = Display en Coma Flotante.....	64
5.1.4.6.8	A = Display ASCII.....	64
5.1.4.6.9	M = Menú Principal.....	64
5.1.4.7	E y F=Comandos Maestros de Error (Puerto 1 y 2).....	64
5.1.4.7.1	S = Mostrar de Nuevo.....	64
5.1.4.7.2	- = Retroceder 2 Páginas.....	64
5.1.4.7.3	P = Página Anterior.....	64
5.1.4.7.4	+ = Avanzar 2 Páginas.....	65
5.1.4.7.5	N = Próxima Página.....	65

5.1.4.7.6	D = Display Decimal	65
5.1.4.7.7	H = Display Hexadecimal	65
5.1.4.7.8	M = Menú Principal	65
5.1.4.8	I y J=Lista maestra de comandos (Puertos 1 y 2)	65
5.1.4.8.1	S = Mostrar de Nuevo	65
5.1.4.8.2	- = Retroceder 5 Páginas	65
5.1.4.8.3	P = Página Anterior	66
5.1.4.8.4	+ = Avanzar 5 Páginas	67
5.1.4.8.5	N = Próxima Página	67
5.1.4.8.6	M = Menú Principal	67
5.1.4.9	O y P=Lista de Status del Esclavo (Puerto 1 y 2)	67
5.1.4.10	V=Información de la Versión	67
5.1.4.11	W=Módulo deArranque en Caliente	68
5.1.4.12	Y=Configuración de Transferencia de Módulo al Procesador	68
5.1.4.13	1 y 2=Status de Comunicación (Puertos 1 y 2)	69
5.1.4.14	6 y 7=Configuración de Puertos (Puertos 1 y 2)	69
5.1.4.15	Esc=Salir del Programa	69
5.2	LED's Indicadores de Status	70
5.2.1	Borrando una Condición de Falla	72
5.2.2	Troubleshooting	73
6	Connexiones de Cableado	75
6.1	Puertos de Comunicación Modbus	75
6.1.1	Conectando el Cableado al Conector	75
6.1.1.1	RS-232	76
6.1.1.2	RS-485	76
6.1.1.3	RS-422	76
6.2	Puerto de Configuration/Depuración RS-232	77
Appendix A	– Definición de la Base de Datos MVI56-MCM	79
Appendix B	– Definición de Datos de Status MVI56-MCM	81
Appendix C	– Definición de Datos de Configuración MVI56-MCM	83
	Inicialización del Backplane	83
	Inicialización del Puerto 1	84
	Inicialización del Puerto 2	86
	Comandos del Puerto 1	89
	Comandos del Puerto 2	90
	Status Misceláneos	90
	Control de Comando	92
Appendix D	– Control de Comando MVI56-MCM	93
Appendix E	– Especificaciones del Producto	95
	Especificaciones Generales	95
	Especificaciones Funcionales del Esclavo Modbus	96
	Especificaciones Funcionales del Maestro Modbus	96
	Características Físicas	96
	Interfaz ControlLogix	96
	Especificaciones de Hardware	96
	Soporte, Servicio y Garantía	99

1 Introducción

El producto MVI56-MCM (“Módulo de Comunicación Modbus”) facilita la interfaz entre procesadores ControlLogix de Allen Bradley con otros dispositivos compatibles con el protocolo Modbus. Estos dispositivos compatibles incluyen no sólo PLC’s Modicon (todos los cuales soportan el protocolo Modbus) sino también un amplio surtido de dispositivos terminales.

El módulo MVI56-MCM actúa como un gateway entre la red Modbus y el backplane Allen-Bradley. Los datos son transferidos asincrónicamente desde el procesador ControlLogix en las acciones de la red Modbus. El módulo utiliza un espacio de registros de 5000 palabras para el intercambio de datos entre el procesador y la red Modbus.

1.1 Conceptos Generales

La siguiente discusión cubre diversos conceptos que son claves para entender la operación del módulo MVI56-MCM.

Una vez energizado, el módulo comienza a ejecutar las siguientes funciones lógicas:

1. Componentes de hardware que se inicializan:
 - a. Se inicializa el driver del backplane del ControlLogix.
 - b. Prueba y borrado de toda la RAM.
 - c. Se inicializan los puertos de comunicación serial.
2. El módulo espera por su configuración desde el procesador ControlLogix.
3. Se inicializa el espacio de registros del módulo.
4. Se habilita el driver Esclavo de los puertos seleccionados.
5. Se habilita el driver Maestro de los puertos seleccionados.

Una vez que el módulo recibe el bloque de configuración del Módulo desde el procesador, el módulo comenzará la comunicación con los otros nodos de la red, dependiendo de la configuración.

1.2 Inicializando el Módulo

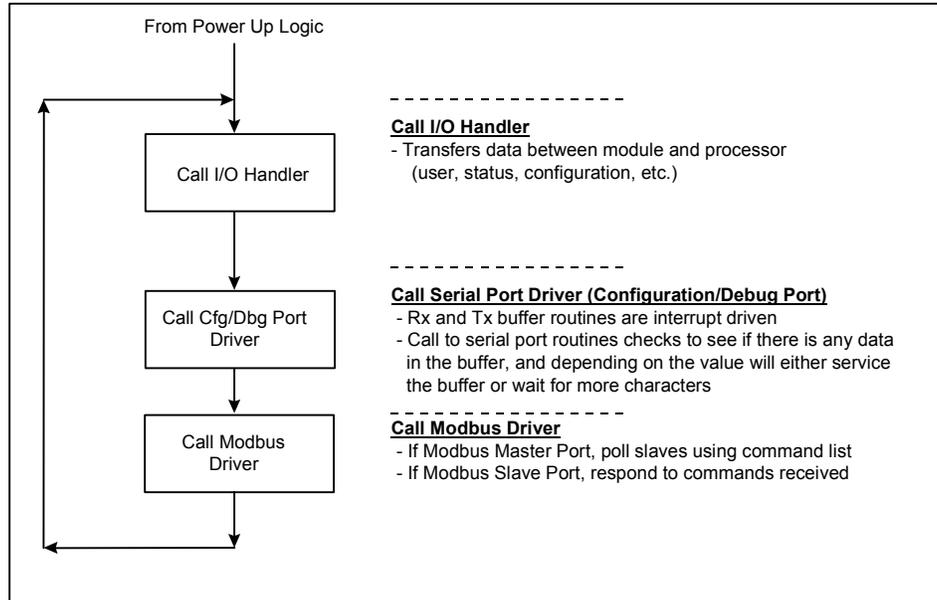
Una vez que su módulo está instalado, usted puede comenzar el proceso de modificación de la configuración de su módulo y de la lógica de escalera. Antes de comenzar éste proceso, usted debería comprender la arquitectura, la cual es mostrada en la próxima sección. Las secciones restantes explican cómo realizar modificaciones al archivo .cfg existente y a la lógica de escalera de ejemplo.

2 Entendiendo la Arquitectura

Esta sección presenta al lector una visión funcional del módulo MVI56-MCM. Los detalles asociados con la lógica de escalera y el mapa de memoria no son cubiertos en ésta sección (para ello refiérase a la sección **Inicialización del Módulo**). Se requiere comprender completamente la información contenida en éste documento para la implementación exitosa de éste módulo en una aplicación del usuario. Si ya usted entiende el contenido de esta sección, refiérase a la sección **Inicialización del Módulo** para el encendido y arranque del módulo. Si usted no está familiarizado con la transferencia de datos y las operaciones del protocolo Modbus, lea éste documento antes de inicializar el módulo.

2.1 Lazo Lógico Principal

Una vez completado el proceso de configuración de encendido, el módulo entra en un lazo infinito que ejecuta las siguientes funciones:



2.2 Procesador ControlLogix Fuera del Modo Run

Cada vez que el módulo detecta que el procesador ControlLogix está fuera del modo Run (por ejemplo, en modo falla (Fault) o en modo programación (PGM)), el puerto Modbus puede ser detenido como está prescrito en la configuración del usuario. Cuando el procesador retorna al estado de ejecución (Run), el módulo reestablece la comunicación con la red.

2.3 Transferencia de Datos en el Backplane

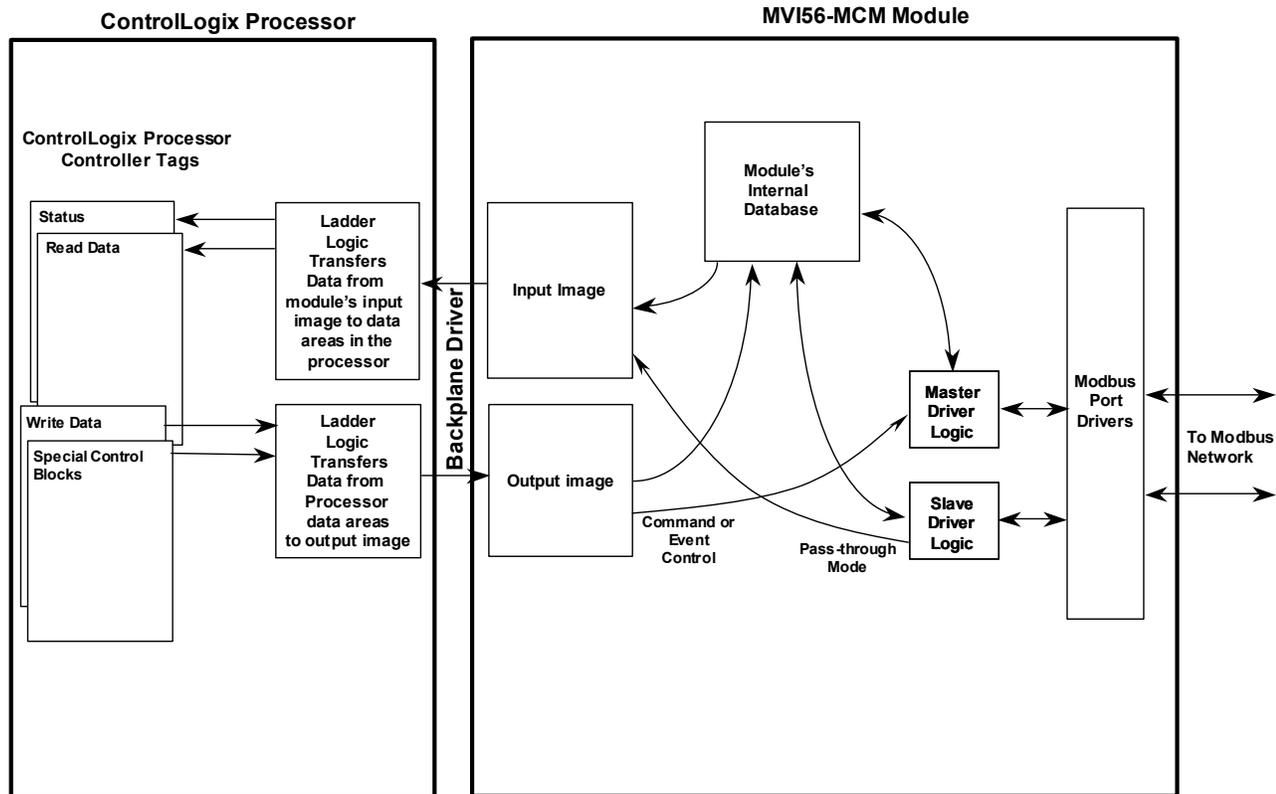
El módulo MVI56-MCM es único en el camino que utiliza el backplane del ControlLogix. Los datos son compaginados entre el módulo y el procesador ControlLogix a través del backplane utilizando las imágenes de entrada y salida del módulo. La frecuencia de actualización de éstas imágenes está determinada por la rata de scan programada y definida por el usuario para el módulo y la carga de comunicación del módulo. Típicamente la actualización está en el orden de 2 a 10 milisegundos.

Esta transferencia bidireccional de datos es efectuada por el módulo llenando de datos la imagen de entrada del módulo para enviarla luego al procesador. Los datos que llegan a la imagen de entrada son colocados por la lógica de escalera en los Tags Controladores en el procesador. La imagen de entrada para el módulo está ajustada a 250 palabras. Esta extensa área de datos permite una transferencia rápida de los mismos entre el módulo y el procesador.

El procesador inserta los datos en la imagen de salida del módulo a ser transferidos al módulo. El programa del módulo extrae los datos y los coloca en la base de datos interna del mismo.

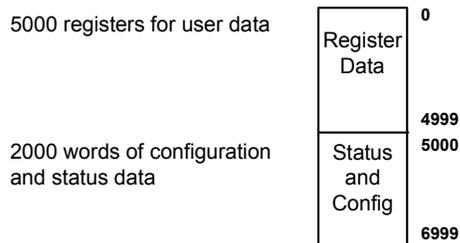
La imagen de salida para el módulo está ajustada a 248 palabras. Esta extensa área de datos permite la transferencia rápida de datos desde el procesador hasta el módulo.

El siguiente diagrama muestra el método utilizado para transferir datos entre el procesador ControlLogix, el módulo MVI56-MCM y la red Modbus.



Como se muestra en el diagrama superior, todos los datos transferidos entre el módulo y el procesador sobre el backplane es a través de las imágenes de entrada y salida. La lógica de escalera debe ser escrita en el procesador ControlLogix para hacer interfaz entre los datos de las imágenes de entrada y salida con los datos definidos en los Tags Controladores. Todos los datos utilizados por el módulo son almacenados en su base de datos interna. Esta base de datos está definida como una tabla de datos Modbus virtual direccionada desde 0 (40001 Modbus) hasta 6999 (47000 Modbus). El diagrama siguiente muestra el arreglo de la base de datos:

Module's Internal Database Structure



Los datos contenidos en esta base de datos están compaginados a través de las imágenes de entrada y salida por la coordinación de la lógica de escalera del ControlLogix y el programa del módulo MVI56-MCM. Pueden transferirse hasta 248 palabras de datos por vez desde el módulo al procesador. Pueden ser transferidas hasta 247 palabras de datos desde el procesador hasta el módulo. Cada imagen tiene una estructura definida dependiendo del contenido de los datos y de la función de transferencia de datos como se define más abajo.

2.4 Transferencia Normal de Datos

La transferencia normal de datos incluye la compaginación de los datos de usuario que se hallan en la base de datos interna del módulo en los registros del 0 al 4999 y los datos de status. Estos datos son transferidos a través de bloques de lectura (imagen de entrada) y escritura (imagen de salida). Refiérase a la sección de **Inicialización del Módulo** para una descripción de los objetos de datos utilizados con los bloques y la lógica de escalera requerida. La estructura y función de cada bloque se discute a continuación.

2.4.1 Bloques de Lectura

Estos bloques de datos son utilizados para transferir información desde el módulo hasta el procesador ControlLogix. La estructura de la imagen de entrada utilizada para transferir éstos datos se muestra en la siguiente tabla:

Offset	Descripción	Longitud
0	Reservado	1
1	ID de Bloque de Escritura	1
2 – 201	Datos de Lectura	200
202	Contador de Scan de Programa	1
203 – 204	Código del Producto	2
205 – 206	Versión del Producto	2
207 – 208	Sistema Operativo	2
209 – 210	Número de Ejecución	2
211 – 217	Status de Error Puerto 1	7
218 – 224	Status de Error Puerto 2	7
225 – 230	Status de Transferencia de Datos	6
231	Error/Índice Actual Puerto 1	1
232	Error/Índice Último Puerto 1	1
233	Error/Índice Actual Puerto 2	1
234	Error/Índice Último Puerto 2	1
235 – 248	Reserva	14
249	ID de Bloque de Lectura	1

El ID (identificador) del bloque de lectura es un valor de índice utilizado para determinar la localización donde los datos serán colocados en el arreglo de datos de lectura del módulo de Tags Controladores del procesador ControlLogix. En cada transferencia se pueden mover hasta 200 palabras (el block hace offset del 2 al 201) de datos. Adicionalmente para mover datos de usuario, el bloque también contiene datos de estado para el módulo. Este último juego de datos es transferido con cada nuevo bloque de datos utilizado para el movimiento de datos a alta velocidad. El ID del bloque de escritura asociado con el bloque es utilizado para requerir datos del procesador ControlLogix. Bajo la operación normal del programa, el módulo envía secuencialmente

bloques de lectura y solicita bloques de escritura. Por ejemplo, si tres bloques de lectura y 2 bloques de escritura son utilizado con la aplicación, la secuencia será la siguiente:

R1W1-->R2W2-->R3W1-->R1W2-->R2W1-->R3W2-->R1W1-->....

Esta secuencia continuará hasta que sea interrumpida por otros números de bloques de escritura enviados por el controlador ó por la solicitud de un comando de un nodo en la red Modbus o el control de operador a través del puerto de Configuración/Depuración del módulo.

2.4.2 Bloques de Escritura

Estos bloques de datos son utilizados para transferir información desde el procesador ControlLogix hacia el módulo. La estructura de la imagen de salida utilizada para transferir éstos datos se muestra en la siguiente tabla:

Offset	Descripción	Longitud
0	ID de Bloque de Escritura	1
1 – 200	Datos de Escritura	200
201 – 247	Reserva	47

El ID del bloque de escritura es un valor índice utilizado para determinar la localización en la base de datos del módulo donde los datos son colocados. Cada transferencia puede mover hasta 200 palabras (el bloque hace offset desde 1 hasta 200) de datos.

2.5 Configuración de la Transferencia de Datos

Cuando el módulo ejecuta una operación de reinicio, requerirá información de configuración del procesador ControlLogix. Estos datos son transferidos al módulo en bloques de escritura formateados especialmente (imagen de salida). El módulo preguntará por cada bloque ajustando el número de bloques de escritura requeridos en un bloque de lectura (imagen de entrada). Refiérase a la sección de **Inicialización del Módulo** para una descripción de los objetos de datos utilizados con los bloques y la lógica de escalera requerida. El formato de los bloques para la configuración es indicado en las siguientes secciones.

2.5.1 Datos de Configuración del Módulo

Este bloque es utilizado para enviar información de configuración general desde el procesador hacia el módulo. Los datos son transferidos en un bloque con un código de identificación de 9000. La estructura del bloque se muestra en la siguiente tabla:

Offset	Descripción	Longitud
0	9000	1
1 – 6	Inicialización del Backplane	6
7 – 31	Configuración del Puerto 1	25
32 – 56	Configuración del Puerto 2	25
57 – 59	Configuración Aux. del Puerto 1	3
60 – 62	Configuración Aux. del Puerto 2	3
63 – 247	Reserva	185

El bloque de lectura utilizado para requerir la configuración tiene la siguiente estructura:

Offset	Descripción	Longitud
0	Reservado	1
1	9000	1
2	Errores de Configuración del Módulo	1
3	Errores de Configuración Puerto 1	1
4	Errores de Configuración Puerto 1	1
5 – 248	Reserva	244
249	-2 ó -3	1

Si hay errores en la configuración, el bit asociado con el error se ajustará en una de las tres palabras de error de configuración. El error debe ser corregido antes de que el módulo inicie su modo de operación normal.

2.6 Lista Maestra de Datos de Comando

Cada puerto en el módulo puede ser configurado como un dispositivo Maestro Modbus que contiene su propia lista de cien comandos. Los comandos son leídos desde el procesador utilizando los siguientes ID's de bloques de escritura: Puerto Modbus 1 – 6000 hasta 6003 y Puerto Modbus 2 – 6100 hasta 6103. El módulo preguntará secuencialmente por cada bloque desde el procesador. La lógica de escalera se debe escribir para manejar uno y cada uno de los datos transferidos. La estructura de cada bloque esta indicada en la siguiente tabla:

Offset	Descripción	Longitud
0	6000 hasta 6003 y 6100 hasta 6103	1
1 – 8	Definición de Comando	8
9 – 16	Definición de Comando	8
17 – 24	Definición de Comando	8
25 – 32	Definición de Comando	8
33 – 40	Definición de Comando	8
41 – 48	Definición de Comando	8
49 – 56	Definición de Comando	8
57 – 64	Definición de Comando	8
65 – 72	Definición de Comando	8
73 – 80	Definición de Comando	8
81 – 88	Definición de Comando	8
89 – 96	Definición de Comando	8
97 – 104	Definición de Comando	8
105 – 112	Definición de Comando	8
113 – 120	Definición de Comando	8
121 – 128	Definición de Comando	8
129 – 136	Definición de Comando	8
137 – 144	Definición de Comando	8
145 – 152	Definición de Comando	8
153 – 160	Definición de Comando	8
161 – 168	Definición de Comando	8
169 – 176	Definición de Comando	8
177 – 184	Definición de Comando	8
185 – 192	Definición de Comando	8
193 – 200	Definición de Comando	8

2.7 Bloques de Status del Dispositivo Esclavo

Los bloques de status del Esclavo son utilizados para enviar información de status (estado) de cada dispositivo Esclavo en el puerto Maestro. Los Esclavos conectados al puerto Maestro pueden tener uno de los siguientes estados:

0	El Esclavo está inactivo y no está definido en la lista de comandos para el puerto Maestro.
1	El Esclavo está siendo accesado activamente o está controlado por el puerto Maestro y la comunicación es exitosa.
2	El puerto Maestro ha fallado su comunicación con el dispositivo Esclavo. La comunicación con el Esclavo está suspendida durante un período definido por el usuario basado en el muestreo de la lista de comandos.
3	La comunicación con el Esclavo está siendo deshabilitada por la lógica de escalera. Ninguna comunicación ocurrirá con el Esclavo hasta que éste estado sea borrado por la lógica de escalera.

Los Esclavos son definidos hacia el sistema cuando el módulo inicializa la lista maestra de comandos. Cada Esclavo definido se ajustará a un estado de uno (1) en este paso inicial. Si el puerto Maestro falla, la comunicación con un dispositivo Esclavo (cuenta de reintento finalizada en un comando), el Maestro ajustará el estado del Esclavo a un valor de 2 en la tabla de status. Esto suspenderá la comunicación con el dispositivo Esclavo para la cuenta de scan de un usuario específico (valor **Error Delay Cntr** en el objeto **MCMPort** para cada puerto). Cada vez que es escaneado un comando en esta lista que tiene la dirección de un Esclavo suspendido, el valor del contador de retardo será decrementado. Cuando el valor llegue a cero, el estado del Esclavo se ajustara a 1. Esto habilitara el que se pregunte al Esclavo.

BLOCK ID	DESCRIPCIÓN
3002	Solicita los primeros 128 valores de status del Esclavo para el Puerto Modbus 1
3003	Solicita los últimos 128 valores de status del Esclavo para el Puerto Modbus 1
3102	Solicita los primeros 128 valores de status del Esclavo para el Puerto Modbus 2
3103	Solicita los últimos 128 valores de status del Esclavo para el Puerto Modbus 2

El formato de éstos bloques es como se muestra en la siguiente tabla:

Offset	Descripción	Longitud
0	3002 – 3003 or 3102 – 3103	1
1 – 247	Reserva	246

El módulo reconocerá la solicitud recibiendo un código de bloque de escritura especial y respondiendo con un bloque de lectura con el siguiente formato:

Offset	Descripción	Longitud
0	Reservado	1
1	ID del Bloque de Escritura	1
2 – 129	Data de Estado de Acceso Esclavo	128

Understanding the Architecture

130 – 248	Reserva	119
249	3002 - 3003 ó 3102 – 3103	1

La lógica de escalera puede ser escrita para prevalecer sobre el valor en la tabla de estado del Esclavo. Puede inhabilitarse (valor de estado igual a 3) enviando un bloque especial de datos desde el procesador hacia el Esclavo. Los Esclavos del Puerto 1 son inhabilitados utilizando el bloque 3000 y los Esclavos del Puerto 2 son inhabilitados utilizando el bloque 3100. Cada bloque contiene el nodo Esclavo direccionado a inhabilitado (disable). La estructura del bloque es mostrada en la siguiente tabla:

Offset	Descripción	Longitud
0	3000 o 3100	1
1	Número de Esclavos en el bloque	1
2 – 201	Esclavos Indexados	200
202 – 247	Reserva	46

El módulo responderá con un bloque con el mismo código de identificación recibido e indicando el número de Esclavos activados con el bloque. El formato de éste bloque de respuesta es mostrado en la siguiente tabla:

Offset	Descripción	Longitud
0	Reservado	1
1	ID del Bloque de Escritura	1
2	Número de Esclavos Procesados	1
3 – 248	Reserva	246
249	3001 o 3101	1

La lógica de escalera puede ser escrita para prevalecer sobre el valor en la tabla de estado del Esclavo para habilitar el Esclavo (valor de estado igual a 1) enviando un bloque especial. Los Esclavos del Puerto 1 son habilitados utilizando el bloque 3001 y los Esclavos del Puerto 2 son habilitados utilizando el bloque 3101. Cada bloque contiene el nodo Esclavo direccionado a habilitado. El formato del bloque es mostrado en la siguiente tabla:

Offset	Descripción	Longitud
0	3001 or 3101	1
1	Número de Esclavos en el Bloque	1
2 – 201	Esclavos Indexados	200
202 – 247	Reserva	46

El módulo responderá con un bloque con el mismo código de identificación recibido e indicando el número de Esclavos activados con el bloque. El formato de éste bloque de respuesta es mostrado en la siguiente tabla:

Offset	Descripción	Longitud
0	Reservado	1
1	ID del Bloque de Escritura	1
2	Número de Esclavos Procesados	1
3 – 248	Reserva	246
249	3001 o 3101	1

2.8 Bloques de Comandos de Control

Los bloques de Comandos de Control son bloques especiales utilizados para controlar el módulo ó solicitar datos especiales del módulo. La versión actual del software soporta cinco bloques de comandos de control: Comandos de control de eventos, Comandos de control, configuración de escritura, arranque en caliente y arranque en frío.

2.8.1 Comandos de Eventos

Los bloques de comandos de eventos de control son utilizados para enviar comandos Modbus directamente desde la lógica de escalera a uno de los puertos Maestro. El formato para éstos bloques es mostrado en la siguiente tabla:

Offset	Descripción	Longitud
0	1000 –1255 ó 2000 – 2255	1
1	Dirección BD interna	1
2	Cuenta de Punto	1
3	Código de Inversión	1
4	Código de Función Modbus	1
5	Dirección de BD del dispositivo	1
6 – 247	Reserva	242

El número del bloque define el puerto Modbus a ser considerado y el nodo Esclavo a ser accesado. Los bloques en el rango de 1000 están dirigidos al puerto Modbus 1 y los bloques en el rango de 2000 están dirigidos al puerto Modbus 2. La dirección del Esclavo está representada en el número del bloque en el rango de 0 a 255. La suma de éstos dos valores determina el número del bloque. Los otros parámetros pasados con el bloque son utilizados para construir el comando. El comando **Internal/DataBase Address** especifica la localización de la base de datos del módulo asociada con el comando. El parámetro **Point Count** define el número de puertos ó registros para los comandos. El código de inversión (**Swap Code**) es utilizado con la función Modbus 3 requerida para invertir el orden de la palabra o bit. El código de función Modbus (**Modbus Function Code**) tiene uno de los siguientes valores: 1, 2, 3, 4, 5, 6, 15 o 16. La dirección de base de datos de dispositivo (**Device Database Address**) es el registro ó punto Modbus en el dispositivo Esclavo remoto a ser asociado con el comando. Cuando el comando recibe el bloque, lo procesará y colocará en la cola del comando. El módulo responderá a cada bloque de comando de eventos con un bloque de lectura con el siguiente formato:

Offset	Descripción	Longitud
0	Reservado	1
1	ID del Bloque de Escritura	1
2	0=Fail (fallo); 1=Success (acierto)	1
3 – 248	Reserva	246
249	1000 – 1255 o 2000 – 2255	1

La segunda palabra del bloque puede ser usada por la lógica de escalera para determinar si el comando fue añadido a la cola de comando del módulo. El comando sólo fallará si la cola de comando para el puerto está llena (100 comandos por cada cola).

2.8.2 Comandos de Control

Los bloques de comandos de control son utilizados para colocar comandos en la lista de comandos dentro de la cola de comandos. Cada puerto tiene una cola de comandos de hasta 100 comandos. El módulo sirve comandos en la cola antes de la lista maestra de comandos. Esto da alta prioridad a los comandos en la cola. Los comandos colocados en la cola a través de éste mecanismo deben ser definidos en la lista maestra de comandos. Bajo la ejecución de la lista de comandos normal, el módulo solo ejecutará comandos con el parámetro Enable ajustado a 1 ó 2. Si el valor está ajustado a 0, el comando es ignorado. Los comandos pueden ser colocados en la lista de comandos con el parámetro Enable ajustado a cero. Estos comandos pueden ser ejecutados utilizando los bloques de control de comando.

De uno a seis comandos pueden ser colocados en la cola de comando con una solicitud sencilla. El formato de éste bloque es mostrado en la siguiente tabla.

Offset	Descripción	Longitud
0	5001 – 5006 ó 5101 – 5106	1
1	Índice de Comando (MCM.P1.CMD [valor índice de comando])	1
2	Índice de Comando (MCM.P1.CMD [valor índice de comando])	1
3	Índice de Comando (MCM.P1.CMD [valor índice de comando])	1
4	Índice de Comando (MCM.P1.CMD [valor índice de comando])	1
5	Índice de Comando (MCM.P1.CMD [valor índice de comando])	1
6	Índice de Comando (MCM.P1.CMD [valor índice de comando])	1
7 – 247	Spare	241

Los bloques en el rango de 5001 hasta 5006 son usados por el puerto Modbus 1 y los bloques en el rango de 5101 hasta 5106 son usados por el puerto Modbus 2. El último dígito en el código del bloque define el número de comandos a procesar en ése bloque. Por ejemplo, un código de bloque 5003 contiene 3 índices comandos que están siendo usados con el puerto Modbus 1. Los parámetros de índice de comando en el bloque tienen un rango de 0 a 99 y corresponden a las entradas de la lista maestra de comandos.

El módulo responde a un bloque de control de comando con un bloque que contiene el número de comandos añadidos a la cola de comandos para el puerto. El formato del bloques es mostrado en la siguiente tabla:

Offset	Descripción	Longitud
0	Reservado	1
1	ID del Bloque de Escritura	1
2	Número de Comandos Añadidos a la cola de comandos	1
3 – 248	Reserva	246
249	5000 – 5006 ó 5100 - 5106	1

2.8.3 Configuración de Escritura

Este bloque es enviado desde el procesador ControlLogix hacia el módulo para forzar al módulo a escribir su configuración actual de regreso al procesador. Esta función es utilizada cuando la configuración del módulo ha sido alterada remotamente utilizando operaciones de escritura en la base de datos. El bloque de escritura tiene un valor de -9000 en la primera palabra. El módulo responderá con bloques que contienen los datos de configuración del módulo. La lógica de escalera debe ser escrita para manejar la recepción de éstos bloques. Los bloques transferidos desde el módulo son como sigue:

Bloque -9000, Datos de Configuración General:

Offset	Descripción	Longitud
0	Reservado	1
1	-9000	1
2 – 7	Inicialización del Backplane	6
8 – 32	Configuración del Puerto 1	25
33 – 57	Configuración del Puerto 2	25
58 – 60	Configuración Aux. del Puerto 1	3
61 – 63	Configuración Aux. del Puerto 2	3
64 – 248	Reserva	185
249	-9000	1

Bloques desde -6000 hasta -6003 y -6100 hasta 6103, datos de lista maestra de comandos para los puertos 1 y 2 respectivamente:

Offset	Descripción	Longitud
0	Reservado	1
1	-6000 a -6003 y -6100 a 6103	1
2 – 9	Definición de Comando	8
10 – 17	Definición de Comando	8
18 – 25	Definición de Comando	8
26 – 33	Definición de Comando	8
34 – 41	Definición de Comando	8
42 – 49	Definición de Comando	8
50 – 57	Definición de Comando	8
58 – 65	Definición de Comando	8
66 – 73	Definición de Comando	8
74 – 81	Definición de Comando	8
82 – 89	Definición de Comando	8
90 – 97	Definición de Comando	8
98 – 105	Definición de Comando	8
106 – 113	Definición de Comando	8
114 – 121	Definición de Comando	8
122 – 129	Definición de Comando	8
130 – 137	Definición de Comando	8
138 – 145	Definición de Comando	8
146 – 153	Definición de Comando	8
154 – 161	Definición de Comando	8
162 – 169	Definición de Comando	8
170 – 177	Definición de Comando	8
178 – 185	Definición de Comando	8
186 – 193	Definición de Comando	8
194 – 201	Definición de Comando	8

202 – 248	Spare	47
249	-6000 to -6003 and -6100 to -6103	1

Cada uno de éstos bloques debe ser manejado por la lógica de escalera para la apropiada operación del módulo.

2.8.4 Arranque en Caliente

Este bloque es enviado desde el procesador ControlLogix hacia el módulo (imagen de salida) cuando se requiere que el módulo ejecute una operación de arranque en caliente (reset de software).

Este bloque se envía comunmente al módulo cada vez que se realizan modificaciones en la data de configuración en el área de datos de Tags del Controlador. Esto fuerza al módulo a leer la nueva información de configuración y al reinicio. La estructura de los bloques de control es mostrada en la siguiente tabla:

Offset	Descripción	Longitud
0	9998	1
1 – 247	Reserva	247

2.8.5 Arranque en Frío

Este bloque es enviado desde el procesador ControlLogix hacia el módulo (imagen de salida) cuando se requiere que el módulo ejecute una operación de arranque en frío (reset de hardware). Este bloque es enviado hacia el módulo cuando un problema de hardware es detectado por la lógica de escalera que requiere un reset de hardware. La estructura de los bloques de control es mostrada en la siguiente tabla:

Offset	Descripción	Longitud
0	9999	1
1 – 247	Reserva	247

2.9 Bloques de Control Pass-Through

2.9.1 Bloques de Control Pass-Through no Formateados

Si uno ó mas de los puertos Esclavo del módulo son configurados para el modo de operación Pass-through sin formato, el módulo pasará bloques con códigos de identificación de 9996 hacia el procesador por cada comando de escritura recibido. Para cualquier función Modbus 5, 6, 15 y 16 los comandos serán pasados desde el puerto al procesador usando éste número de identificación de bloque. La lógica de escalera debe ser escrita para manejar todas las funciones de escritura Modbus hacia el procesador y para responder como se espera a comandos enviados por el dispositivo Maestro Modbus remoto. La estructura de los bloques de control Pass-through sin formato es mostrada en la siguiente tabla:

Offset	Descripción	Longitud
0	0	1
1	9996	1
2	Número de Bytes en Mensaje Modbus	1
3 – 248	Mensaje Modbus Recibido	246
249	9996	1

La lógica de escalera debería analizar la copia del mensaje recibido y controlar el procesador como es esperado por el dispositivo Maestro. El procesador deberá responder al bloque de control Pass-Through con un bloque de escritura con el siguiente formato:

Offset	Descripción	Longitud
0	9996	1
1 – 247	Reserva	247

Esto informará al módulo que el comando está siendo procesado y puede ser borrado desde la cola Pass-Through.

2.9.2 Bloques de Control Pass-Through Formateados

Si uno ó más de los puertos Esclavos del módulo son configurados para el modo de operación Pass-Through formateado, el módulo pasará bloques con código de identificación 9996 hacia el procesador por cada comando de escritura recibido. Cualquier función de comandos Modbus 5, 6, 15 y 16 pasará desde el puerto hacia el procesador utilizando éste número de identificación de bloque. La lógica de escalera debe ser escrita para manejar la recepción de todas las funciones de escritura Modbus hacia el procesador y responder como se espera a los comandos enviados por el dispositivo Maestro Modbus remoto. La estructura del bloque de control pass-through formateado es mostrada en las siguientes tablas:

2.9.2.1 Código de Función 5

Offset	Descripción	Longitud
0	0	1
1	9958	1
2	1	1
3	Bit de Direccion	1
4	Bit de Datos	1
5 – 248	Mensaje Modbus Recibido	244
249	9958	1

La lógica de escalera debería analizar la copia del mensaje recibido y controlar el procesador como es esperado por el dispositivo Maestro. El procesador debería responder al bloque de control Pass-Through con un bloque de escritura con el siguiente formato:

Offset	Descripción	Longitud
0	9958	1
1 – 247	Reserva	247

Esto informará al módulo que el comando está siendo procesado y puede ser borrado desde la cola Pass-Through.

2.9.2.2 Códigos de Función 6 y 16

Offset	Descripción	Longitud
--------	-------------	----------

0	0	1
1	9956 / 9957 (Coma Flotante)	1
2	Número de Palabras de Datos	1
3	Dirección de Datos	1
4 – 248	Datos	244
249	9956 / 9957	1

La lógica de escalera debería copiar el análisis del mensaje recibido y controlar el procesador como es esperado por el dispositivo Maestro. El procesador debería responder al bloque de control Pass-Through con un bloque de escritura con el siguiente formato:

Offset	Descripción	Longitud
0	9956/9957	1
1 – 247	Reserva	247

Esto informará al módulo que el comando está siendo procesado y puede ser borrado desde la cola Pass-Through.

2.9.2.3 Código de Función 15

Cuando el módulo recibe un código de función 15 cuando en el modo Pass-Through, el módulo escribirá los datos utilizando el bloque ID 9959 para datos de multiple-bit. Primero el bit de máscara es utilizado para limpiar los bits a ser actualizados. Esto es efectuado haciendo AND la máscara invertida con los datos existentes. Luego los nuevos datos hechos AND con la máscara son hechos OR con los datos existentes. Esto protege los otros bits en los registros INT desde donde son afectados.

Offset	Descripción	Longitud
0	0	1
1	9959	1
2	Número de Palabras	1
3	Dirección de la Palabra	1
4 – 53	Datos	50
54 – 103	Máscara	50
104 – 248	Reserva	145
249	9959	1

La lógica de escalera debería copiar el análisis del mensaje recibido y controlar el procesador como es esperado por el dispositivo Maestro. El procesador debería responder al bloque de control Pass-Through con un bloque de escritura con el siguiente formato:

Offset	Descripción	Longitud
0	9959	1
1 – 247	Reserva	247

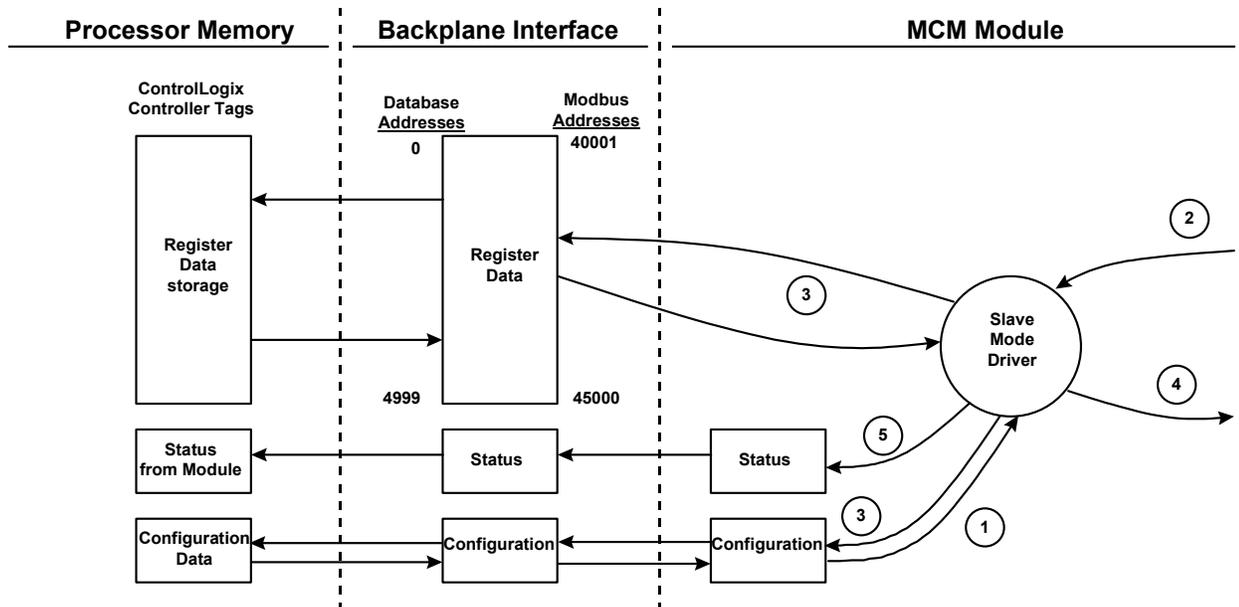
Esto informará al módulo que el comando está siendo procesado y puede ser borrado desde la cola Pass-Through.

Flujo de Datos entre el Módulo MVI56-MCM y el Procesador ControlLogix

La siguiente discusión detalla el flujo de datos entre dos equipos de hardware (el procesador ControlLogix y el módulo MVI56-MCM) y otros nodos sobre la red Modbus bajo diferentes modos de operación del módulo. Cada puerto en el módulo está configurado para emular un dispositivo Maestro Modbus o un dispositivo Esclavo Modbus. La operación de cada puerto es dependiente de ésta configuración. Esta sección que sigue discute la operación de cada modo.

2.9.3 Modo de Driver Esclavo

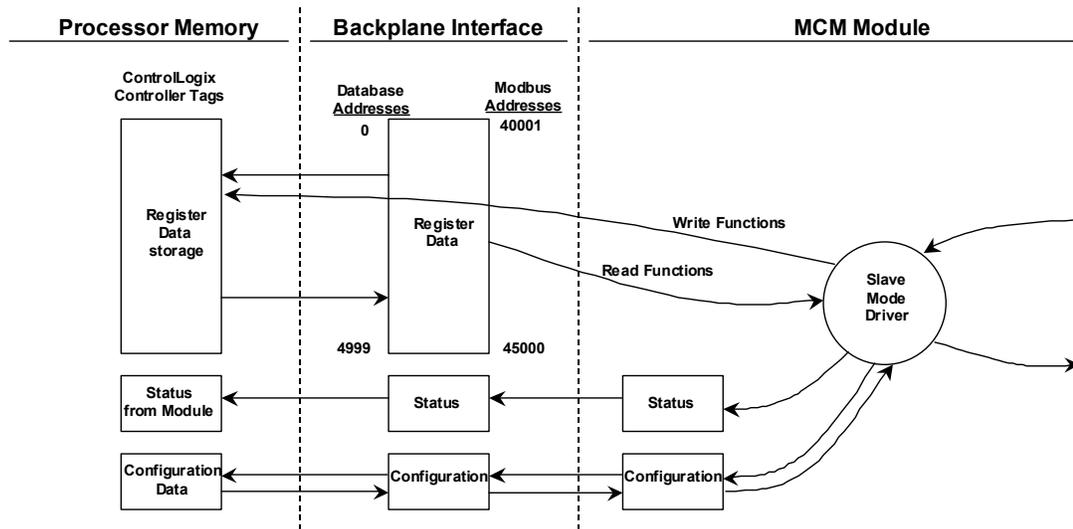
El modo de driver Esclavo permite al módulo MVI56-MCM responder a los datos leídos y escribir comandos enviados por un Maestro sobre la red Modbus. El siguiente flujograma con su tabla asociada, detalla el flujo de datos dentro y fuera del módulo.



Paso	Descripción
1	El driver del puerto Esclavo Modbus recibe la información de configuración desde el procesador ControlLogix. Esta información es utilizada para configurar el puerto serial y define las características del nodo Esclavo. Adicionalmente, la información de configuración contiene datos que pueden ser usados para hacer offset los datos en la base de datos a la dirección requerida en mensajes recibidos desde las unidades maestras.
2	Un dispositivo Host tal como un PLC Modicon ó un paquete MMI, emite un comando de lectura o escritura hacia la dirección del nodo del módulo. El driver del puerto cualifica el mensaje antes de ser aceptado dentro del módulo.
3	Una vez que el módulo acepta el comando, los datos son transferidos inmediatamente hacia o desde la base de datos interna en el módulo. Si el comando es un comando de lectura, los datos son leídos fuera de la base de datos y se construye un mensaje de respuesta.
4	Una vez que el procesamiento de datos de datos esta siendo completado en el paso 2, la respuesta es enviada al nodo Maestro original.
5	Los contadores están disponibles en el bloque de estados que permite a la lógica de escalera del programa determinar el nivel de actividad del driver Esclavo.

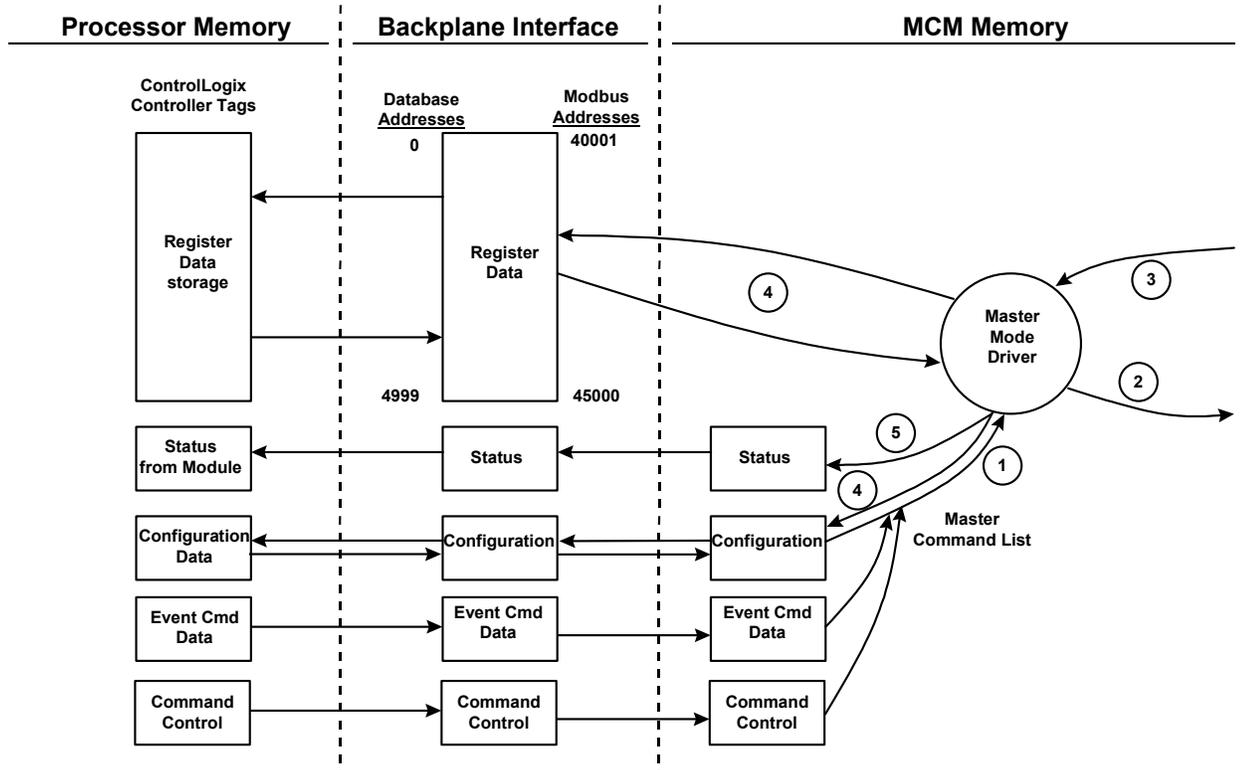
Revise la sección de **Inicialización del Módulo** para una lista completa de los parámetros que deben ser definidos para un puerto Esclavo.

Una excepción a éste modo normal de operación ocurre cuando el modo Pass-through se implementa. En este modo, todos los requerimientos de escritura pasarán directamente hacia el procesador y no serán colocados en la base de datos. Esto permite directamente el control remoto del procesador sin una base de datos intermedia. Este modo es especialmente útil para dispositivos Maestros que no envían ambos estados de control. Por ejemplo, un sistema SCADA puede enviar solo un comando ON a un punto de control digital y nunca enviar el estado de borrado. El sistema SCADA espera que la lógica local resetee el bit de control Pass-Through debe ser usado para simular este modo de operación. El siguiente diagrama ilustra el flujo de datos para un puerto Esclavo con el modo Pass-through habilitado:



2.9.4 Modo de Driver Maestro

En el modo de operación Maestro, el módulo MVI56-MCM es responsable del envío de comandos de lectura o escritura hacia los dispositivos Esclavos sobre la red Modbus. Estos comandos son configurados por el usuario en el módulo vía la lista maestra de comandos recibida desde el procesador ControlLogix o enviada directamente desde el procesador ControlLogix (Control de Comandos de Evento). El estado de comandos es retornado hacia el procesador por cada comando individual en el bloque de estados de lista de comando. La localización de este bloque de status en la base de datos interna del módulo, es definida por el usuario. El siguiente flujograma y tabla asociados, detallan el flujo de datos dentro y fuera del módulo.



Step	Descripción
1	El driver del puerto Maestro obtiene los datos de configuración desde el procesador ControlLogix. Los datos de configuración obtenidos incluyen el número de comandos y la lista maestra de comandos. Estos valores son usados por el driver Maestro para determinar el tipo de comandos a ser enviados a los otros nodos sobre la red Modbus (Ver la Guía de Inicialización del Módulo MVI56-MCM).
2	Una vez configurado el driver, el Maestro comienza a transmitir los comandos de lectura y/o escritura a los otros nodos sobre la red. Si se están escribiendo datos a otro nodo, los datos para los comandos de escritura son obtenidos desde la base de datos interna del módulo para construir el comando.
3	Presumiendo el procesamiento exitoso por el nodo especificado en el comando, un mensaje de respuesta es recibido dentro del driver Maestro para ser procesado..
4	Los datos recibidos desde el nodo sobre la red son pasados dentro de la base de datos interna del módulo, asumiendo un comando de lectura.
5	El estado es retornado al procesador ControlLogix para cada comando en la lista maestra de comandos. Refiérase a la sección de Inicialización de Módulo para una descripción completa de los parámetros requeridos para definir el puerto Maestro Modbus virtual. Refiérase a la documentación del driver MCM para una discusión completa de la estructura y contenido de cada comando. Se debe tener mucho cuidado en la construcción de cada comando en la lista para la operación predecible del módulo. Si dos comandos escriben a la misma dirección de la base de datos interna del módulo, el resultado no sería el deseado. Todos los comandos conteniendo datos inválidos serán ignorados por el módulo.

Refiérase a la sección de **Inicialización del Módulo** para una descripción completa de los parámetros requeridos para definir el puerto Maestro virtual Modbus. Refiérase a la documentación del driver MCM para un discusión completa de la estructura y contenido de cada comando. Tenga cuidado en la construcción de cada comando en la lista para la operación predecible del módulo. Si dos comandos escribe la misma dirección de base de datos interna del módulo, los resultados no serán los deseados. Todos los comandos conteniendo datos inválidos serán ignorados por el módulo.

2.9.4.1 Lista de Maestra de Comandos

De manera de hacer funcionar el módulo Maestro, la lista maestra de comandos del módulo debe ser definida. Esta lista contiene hasta 100 entradas conteniendo la información requerida para construir un comando válido. Esto incluye lo siguiente:

Modo de habilitación de comando ((0) Inhabilitado, (1) Continuo o (2) Condicional).

Dirección de Nodo Esclavo.

Comando tipo lectura o escritura hasta 125 palabras (2000 bits) por comando.

Direcciones de registro fuente y destino de la base de datos – Determina dónde serán colocados y obtenidos los datos.

Cuenta-Selecciona el número de palabras a ser transferidas – 1 a 125 en FC, 3, 4 ó 16. Selecciona el número de bits en FC 1, 2, 15.

Asi como la lista es leída desde el procesador y los comandos son procesados, un valor de error es mantenido en el módulo para cada comando. Esta lista de error puede ser transferida hacia el procesador. Los errores generados por el módulo son mostrados en las siguientes tablas.

Errores de Protocolo Modbus Estándar

Código	Descripción
1	Función Ilegal
2	Direccion de Datos Ilegal
3	Valor de Datos Ilegal
4	Falla en Dispositivo Asociado

5	Reconocimiento
6	Ocupado, mensaje rechazado

Códigos de Errores de Comunicación del Módulo

Código	Descripción
-1	Línea de Control de Modem CTS ajustada antes de transmitir
-2	Ocurrencia de Timeout mientras el mensaje es transmitido
-11	Ocurrencia de Timeout esperando por respuesta a una solicitud
253	Dirección de Esclavo incorrecta en respuesta
254	Código de función incorrecto en respuesta
255	Valor CRC/LRC inválido en respuesta

Código de Errores de Entrada de Lista de Comandos

Código	Descripción
-41	Código habilitado inválido
-42	Dirección Interna > Dirección Máxima
-43	Dirección de Nodo Inválida (<0 o >255)
-44	Parámetro de cuenta ajustado a 0
-45	Código de función inválido
-46	Código de inversión inválido

3 Modificando la Configuración del Módulo

De manera de hacer funcionar el módulo MVI56-MCM, una cantidad mínima de datos de configuración debe ser transferida hasta el módulo. La tabla siguiente provee una visión de los diferentes tipos de datos de configuración que el módulo requiere, dependiendo de los modos de operación a ser soportados.

Dirección de Registro del Módulo	Funcional Modes Affected	Name	Descripción
5000-5009	Modos Funcionales Afectados	Nombre	Descripción
5010-5039 y 5040-5069	Transferencia de Datos	Configuración General del Módulo	Esta sección de los datos de configuración contiene los datos de configuración del módulo que definen la transferencia de datos entre el módulo y el procesador ControlLogix
5070-5869 y 5870-6669	Maestro y Esclavo	Configuración del Puerto	Estas secciones son usadas para definir las características de cada puerto de comunicación serial Modbus en el módulo. Estos parámetros deben ser ajustados correctamente para la apropiada operación del módulo.
6750-6770	Maestro	Lista maestra de comandos	Si la funcionalidad del modo Maestro del módulo esta siendo soportada sobre un puerto, la lista maestra de comandos debe ser inicializada.

Refiérase a la sección **Inicialización del Módulo** para una descripción de la configuración del módulo. El módulo MVI56-MCM debe ser configurado al menos una vez cuando la tarjeta es energizada primero y en cualquier momento después cuando los parámetros deban ser cambiados.

3.1 Encendido

Durante su encendido, el módulo entra en un lazo lógico esperando recibir los datos de configuración desde el procesador. Una vez que ésta es recibida, el módulo comenzará a ejecutar la lista de comandos si está presente.

3.2 Cambiando Parámetros Durante la Operación

Una copia de los datos de configuración del módulo es mapeada en la base de datos del módulo como se muestra en la tabla superior. Estos valores son inicializados cuando el primer módulo recibe su configuración del procesador ControlLogix. Cualquier nodo sobre la red puede cambiar éstos datos. Un puerto Maestro sobre el módulo puede solicitar datos a un Esclavo o un puerto Esclavo puede recibir los datos de una unidad maestra remota. El módulo no usará estos datos hasta que es comandado. La lógica de escalera puede ser escrita para enviar un bloque de comandos de configuración de escritura (9997) hacia el módulo. Un dispositivo remoto puede ejecutar un valor de 9997 a la dirección 6800 en el módulo para descargar la configuración al procesador. Alternativamente, el puerto de Configuración/Depuración del módulo puede ser usado para enviar el comando directamente al módulo. Todos estos tres métodos forzarán al módulo a descargar la configuración al procesador ControlLogix. La lógica de escalera debe existir en el procesador para aceptar los bloques enviados por el módulo. Si todo está configurado correctamente, el módulo podrá recibir su configuración de un dispositivo remoto.

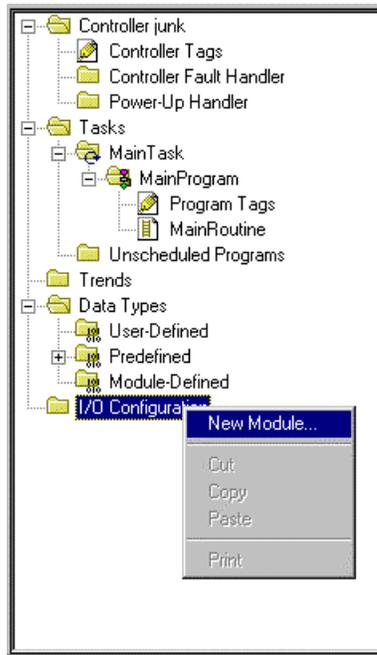
3.3 Inicialización del Módulo

Para inicializar el módulo MVI56-MCM solo se requiere la configuración de software utilizando el programa RSLogix5000. El método más fácil a implementar en el módulo es comenzar con el programa de ejemplo provisto con el módulo (MVI56_MCM_EX1.ACD).

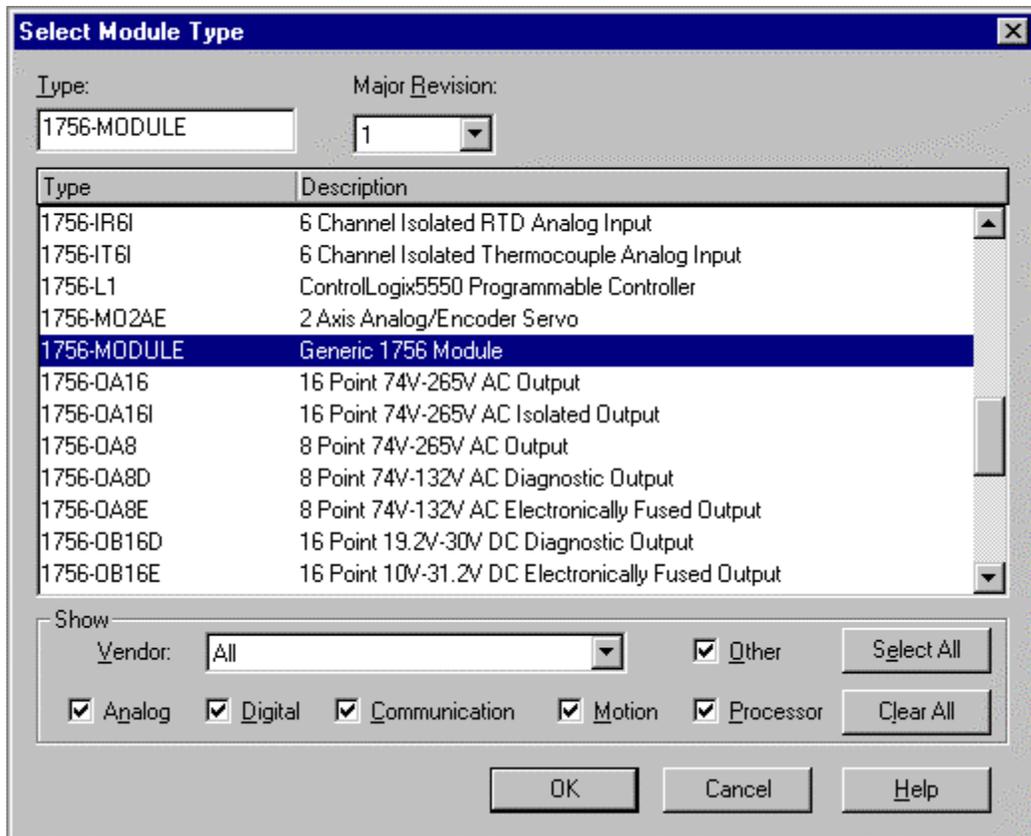
Si usted esta instalando éste módulo en una aplicación existente, usted puede simplemente copiar los elementos requeridos de la lógica de escalera del ejemplo a su aplicación.

Nota: El módulo puede ser añadido solamente a un proyecto utilizando el software en el modo offline.

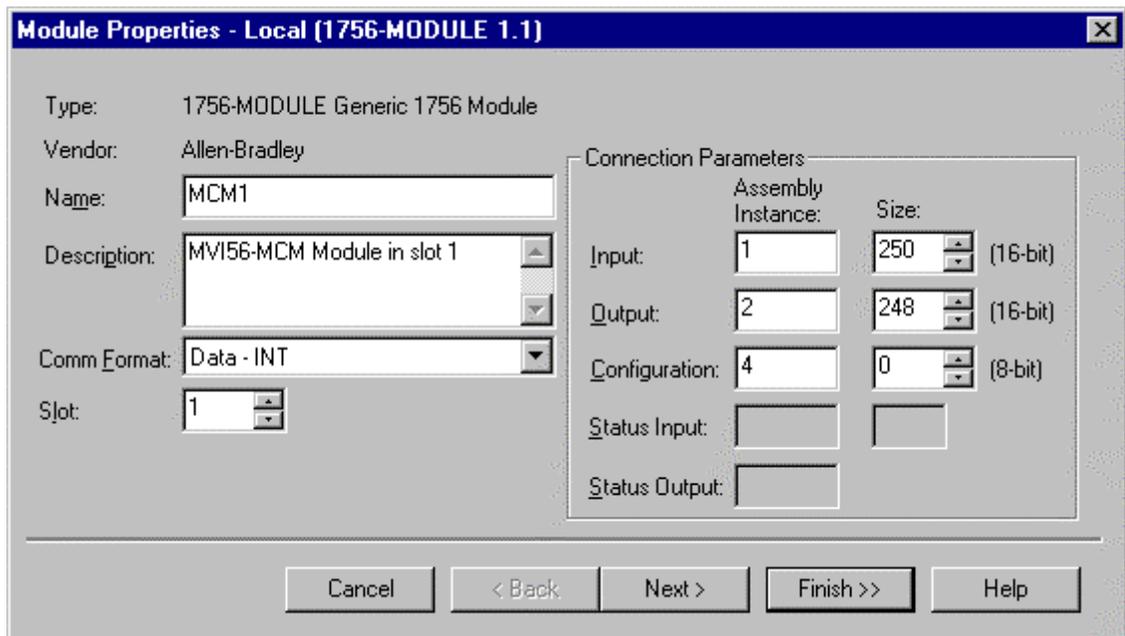
El primer paso en la inicialización del módulo es definir el módulo en el sistema. Presionando el boton derecho del ratón sobre la opción de configuración E/S (entrada/salida (E/S)) en la ventana de Organización del Controlador (Controller Organization) mostrará un menú pop-up. Seleccione la opción de módulo nuevo (New Module) desde el menú de configuración E/S:



Esto hace que el programa muestre la siguiente caja de diálogo:



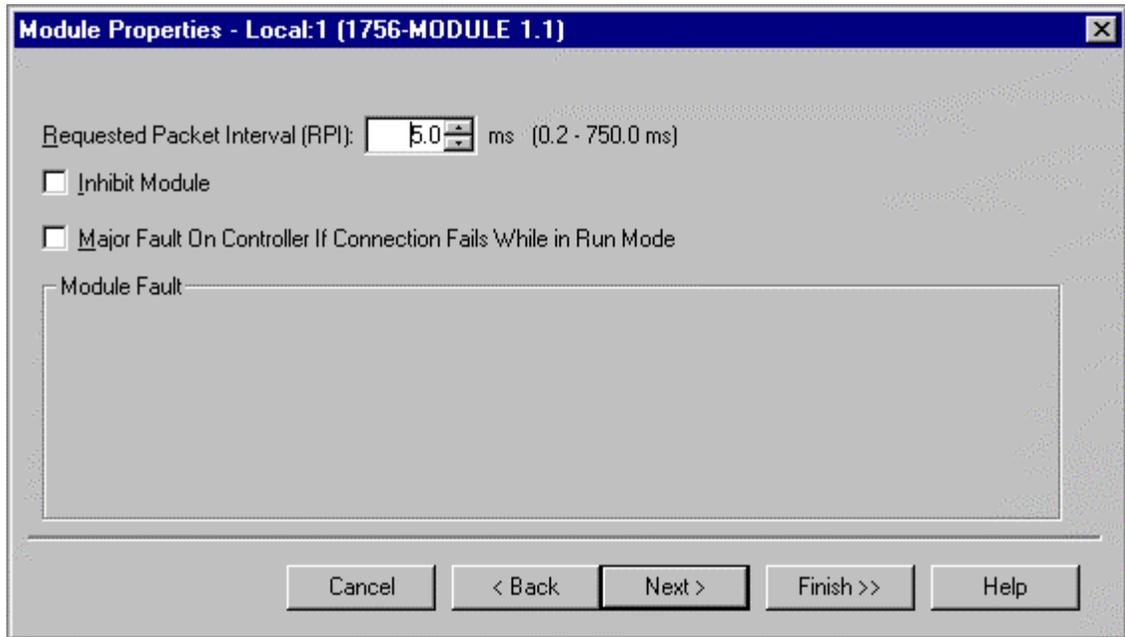
Seleccione el módulo 1756 (Módulo 1756 genérico) de la lista y seleccione el botón OK. La siguiente caja de diálogo es mostrada:



Llene las cajas de diálogo como se muestra llenado las opciones de Nombre, Descripción y Slot para su aplicación. Asegurese de seleccionar el formato de comunicación **Comm Format** como **Data - INT** en la caja de diálogo. Un fallo en el

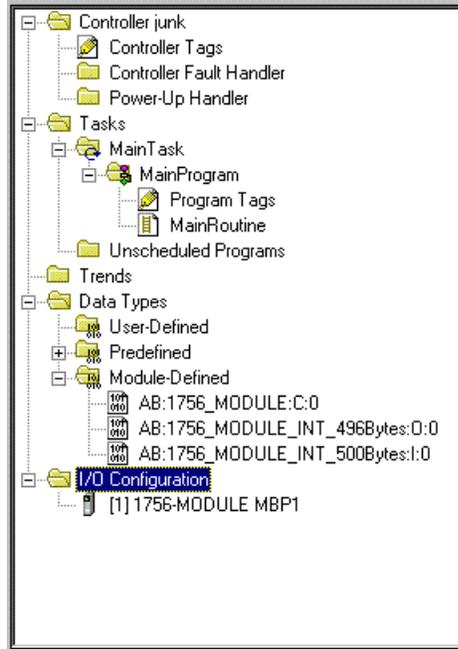
Modifying the Módulo Configuration

ajuste de los valores de **Assembly Instance** y **Size** resultara en un módulo que no se comunicará sobre el backplane del rack del ControlLogix. Seleccione el botón de comando Next para mostrar la siguiente caja de diálogo.

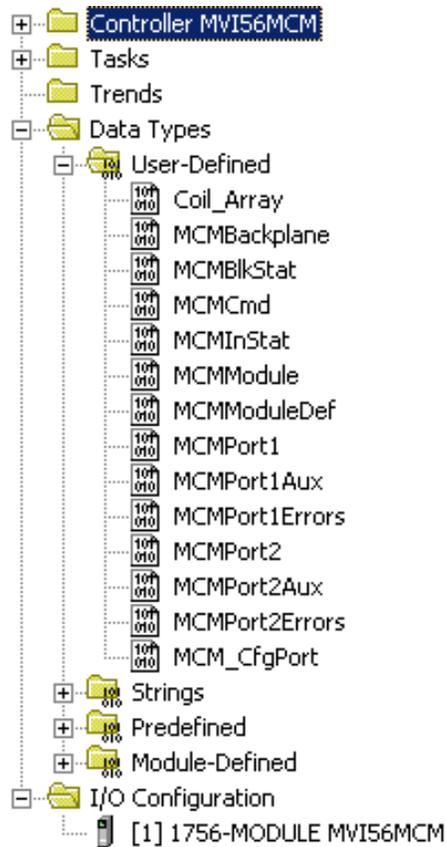


Seleccione el valor de intervalo del Paquete Requerido (Request Packet) para el scanning del E/S sobre el módulo. Este valor representa la mínima frecuencia con que el módulo manejará eventos programados. Este valor no debería ser ajustado por debajo de 1 milisegundo. Valores entre 1 y 10 milisegundos podrían trabajar con la mayoría de las aplicaciones.

Una vez completada la inicialización del módulo, la ventana de Organización del Controlador mostrara la presencia del módulo. Los datos requeridos para el módulo son entonces definidos para la aplicación, y los objetos son localizados en el área de datos de Tags del Controlador. Un ejemplo de la ventana de Organización del Controlador se muestra en el siguiente ejemplo:



El próximo paso en la inicialización del módulo es definir los tipos de datos a ser usados para la interfaz con el módulo. Copie éstos tipos de datos desde la lógica de escalera de ejemplo si usted no está utilizando el ejemplo. Están definidos si usted está comenzando con la lógica de escalera de ejemplo. La ventana de Organización del Controlador debería mostrar los Tipos de Datos Definidos por el Usuario mostrados en el siguiente ejemplo:



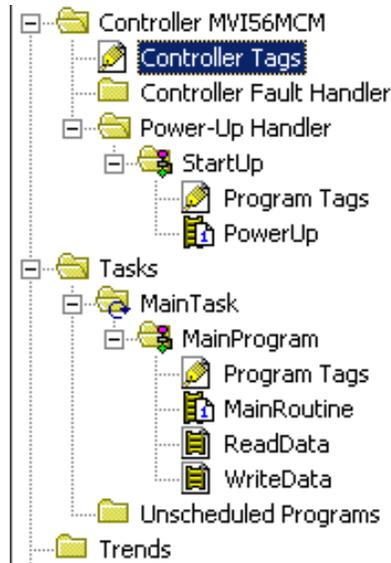
El próximo paso en la inicialización del módulo es definir los datos a ser usados para la interfaz entre el módulo y la lógica de escalera.

Abra la caja de diálogo de edición de Tags del Controlador e introduzca los valores mostrados en el siguiente ejemplo. El módulo MVI56-MCM está definido en el ejemplo como MCM1. Usted puede ajustar el nombre del tag a cualquier nombre del tag válido que usted desee. Si usted está utilizando la lógica de escalera de ejemplo, éste paso ya ha sido realizado.

Tag Name	Value	Force Mask	Style	Type	Description
CmdControl	0		Decimal	BOOL	
ColdBoot	2#0000_0000		Binary	BOOL	
EventCmd	0		Decimal	BOOL	
Local1:C	{...}	{...}		AB:1756_MODULE:C:0	
Local1:I	{...}	{...}		AB:1756_MODULE_INT...	
Local1:O	{...}	{...}		AB:1756_MODULE_INT...	
MBCoil	{...}	{...}		Coil_Array	
MBControl1	{...}	{...}		CONTROL	
MBControl2	{...}	{...}		CONTROL	
MBMsg	{...}	{...}	Decimal	SINT[500]	
MBMsgLen	200		Decimal	INT	
MBOffset	0		Decimal	INT	
MBOffsetBit	0		Decimal	INT	
MBScratch	{...}	{...}	Decimal	INT[3]	
MCM	{...}	{...}		MCMModuleDef	
MJFAULTS	{...}	{...}	Decimal	DINT[12]	
Port1Slave0Read	2#0000_0000		Binary	BOOL	
Port1Slave128Read	2#0000_0000		Binary	BOOL	
Port2Slave0Read	2#0000_0000		Binary	BOOL	
Port2Slave128Read	2#0000_0000		Binary	BOOL	
WarmBoot	2#0000_0000		Binary	BOOL	

En este punto, tómese el tiempo para llenar los valores de configuración en la tabla de datos MCM1 y ajustar los tamaños del arreglo. Refiérase a la sección de objetos de datos del módulo de este documento para obtener información sobre la configuración del módulo.

El ultimo paso en la inicialización del módulo es añadir la lógica de escalera. Si usted está utilizando la lógica de escalera del ejemplo, copie la lógica de escalera mostrada en la ventana de Organización del Controlador debajo de su aplicación.



El módulo está ahora inicializado y listo para ser utilizado con su aplicación.

Descargue la nueva aplicación en el procesador y colóquelo en el modo Run. Si todos los parámetros de configuración están ajustados correctamente y el módulo está conectado a la red Modbus, el LED de la aplicación del módulo (LED APP) debería permanecer apagado y el LED de actividad del backplane (BP ACT) debería titilar muy rápidamente. Refiérase a la sección de **Diagnostico y Troubleshooting** de éste manual si usted encuentra errores. Conecte un computador o terminal al puerto de Configuración/Depuración en el módulo y chequee el status del módulo utilizando el depurador existente en el módulo.

3.4 Objetos de Datos del Módulo (MCM MóduloDef)

Todos los datos relativos al MVI56-MCM son almacenados en tipos de datos definidos por el usuario. Una instancia de los tipos de datos es requerida antes de que el módulo pueda ser utilizado. Esto se hace simplemente declarando una variable tipo dato en la caja de diálogo de Tags de Edición de Tags del Controlador. La estructura del objeto está mostrada en la siguiente figura:

Warning: This structure is being referenced. Modifications will result in loss of data.

Name:

Description:

Members: Data Type Size: ?? byte(s)

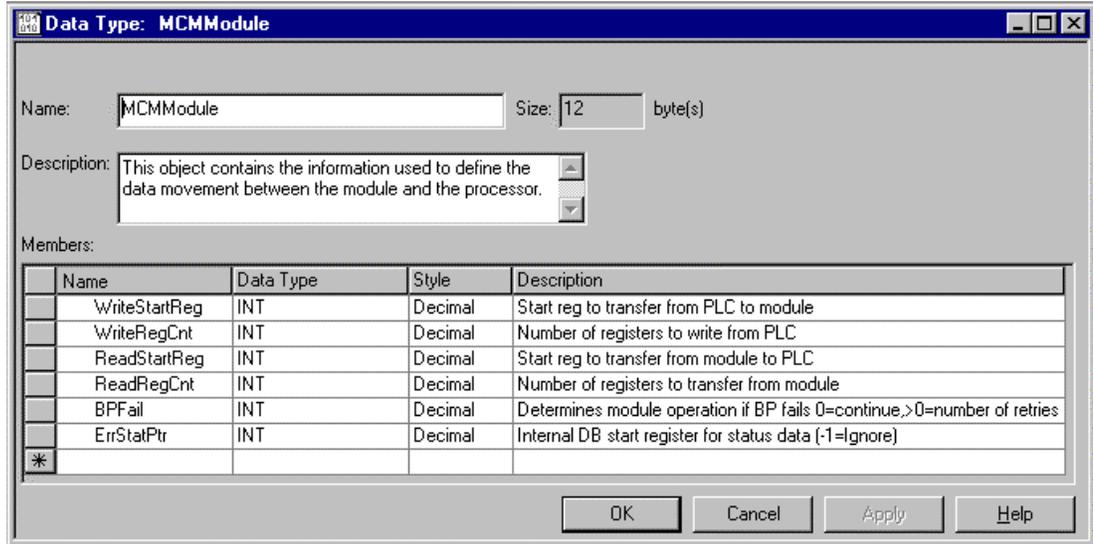
	Name	Data Type	Style	Description
<input type="checkbox"/>	ModDef	MCMModule		Settings for data to be read and written to the module
<input type="checkbox"/>	CFG_Port	MCM_CfgPort		Settings for debug port
<input type="checkbox"/>	Port1	MCMPort1		Modbus settings for P1
<input type="checkbox"/>	Port2	MCMPort2		Modbus settings for P2
<input type="checkbox"/>	P1Cmd	MCMCmd[100]		Master Command List for Port 1
<input type="checkbox"/>	P2Cmd	MCMCmd[100]		Master Command List for Port 2
<input type="checkbox"/>	InStat	MCMInStat		Status information in each read block
<input type="checkbox"/>	ReadData	INT[600]	Decimal	Data read from module
<input type="checkbox"/>	WriteData	INT[600]	Decimal	Data to write to module
<input type="checkbox"/>	BP	MCMBackplane		Data to handle backplane logic
<input type="checkbox"/>	P1Slaves	INT[256]	Decimal	Port 1 slave status values
<input type="checkbox"/>	P2Slaves	INT[256]	Decimal	Port 2 slave status values
<input type="checkbox"/>	Port1Aux	MCMPort1Aux		MB auxilliary settings for Port 1
<input type="checkbox"/>	Port2Aux	MCMPort2Aux		MB auxilliary settings for Port 2
<input type="checkbox"/>	*			

Este objeto contiene objetos que definen la configuración, los datos del usuario, los estados y datos de control de comandos relativos al módulo. Cada uno de estos tipos de objetos es discutido en las siguientes secciones del documento.

3.4.1 Objetos de Configuración

La configuración del módulo es ejecutada simplemente llenando en los valores del objeto del módulo definido en la caja de diálogo de Tags de Edición de Tags del Controlador. Cada parámetro requerido por el módulo tiene una localización definida en el objeto. Las siguiente tablas y discusiones describen los parámetros ajustados en la caja de diálogo. Usted puede ver éstas tablas abriendo los tipos de datos bajo la opción de Tipos de Datos Definidos por el Usuario en la ventana de Organización del Controlador.

3.4.1.1 Parámetros de Transferencia de Datos (MCMódulo)



Este objeto es usado para definir los parámetros para el movimiento de datos entre el módulo y el procesador. Los valores introducidos determinan la lógica de escalera y el tamaño de datos requerido en la aplicación. Los arreglos de escritura y lectura de datos deben ser dimensionados mucho más grandes que los valores de cuenta introducidos. La lógica de escalera debe ser escrita para procesar el número de bloques de datos a ser transferidos. El número de bloques es computado como sigue:

$$\text{BlockCnt} = \text{INT}(\text{Reg Cnt} / 200) + \text{if}(\text{MOD}(\text{Reg Cnt}, 200), 1, 0)$$

Si la cuenta del registro es exactamente divisible por 200, el número de bloques es fácil de computar y la escalera es mucho más simple de escribir. Si el número no es exactamente divisible por 200, se debe desarrollar un manejo especial del ultimo bloque de datos, así como se deben transferir menos de 200 palabras. Se recomienda que los valores de cuenta siempre se ajusten a valores exactamente divisibles por 200.

El parámetro BPFail es utilizado para determinar si el módulo debería continuar sobre la red Modbus cuando la operación de transferencia del backplane falla. Un valor de cero indica que el módulo debería continuar comunicando cuando el backplane no está operando. Si el valor es mayor que cero, el backplane será interrogado un número de veces (entrada) antes que una falla sea reportada y cese la comunicación en los puertos. Cuando la comunicación del backplane es reestablecida, el módulo iniciará la comunicación con la red. Por ejemplo, si usted introdujo un valor de 10 para el parámetro, el módulo detendrá todas las comunicaciones Modbus si 10 errores sucesivos del backplane son reconocidos. Cuando es reconocida una transferencia exitosa, el módulo reestablecerá la comunicación con la red.

El parámetro ErrStatPtr es utilizado para definir la localización en la base de datos del módulo donde los datos de error/status serán almacenados. Si el valor es ajustado a -1, el dato no será almacenado en el área de datos del usuario. Un valor entre 0 y 4939 causará que el programa del módulo almacene los datos en la localización específica.

3.4.1.2 parámetros del Puerto Modbus (MCMPort)

Name:

Description:

Members: Data Type Size: 12 byte(s)

Name	Data Type	Style	Description
Baudrate	DINT	Decimal	Baudrate for P1 debug port. Fixed value and cannot be changed with this setting.
Parity	INT	Decimal	0=None. Fixed value
DataBits	INT	Decimal	8 data bits. Fixed value
StopBits	INT	Decimal	1 stop bits. Fixed value

Name:

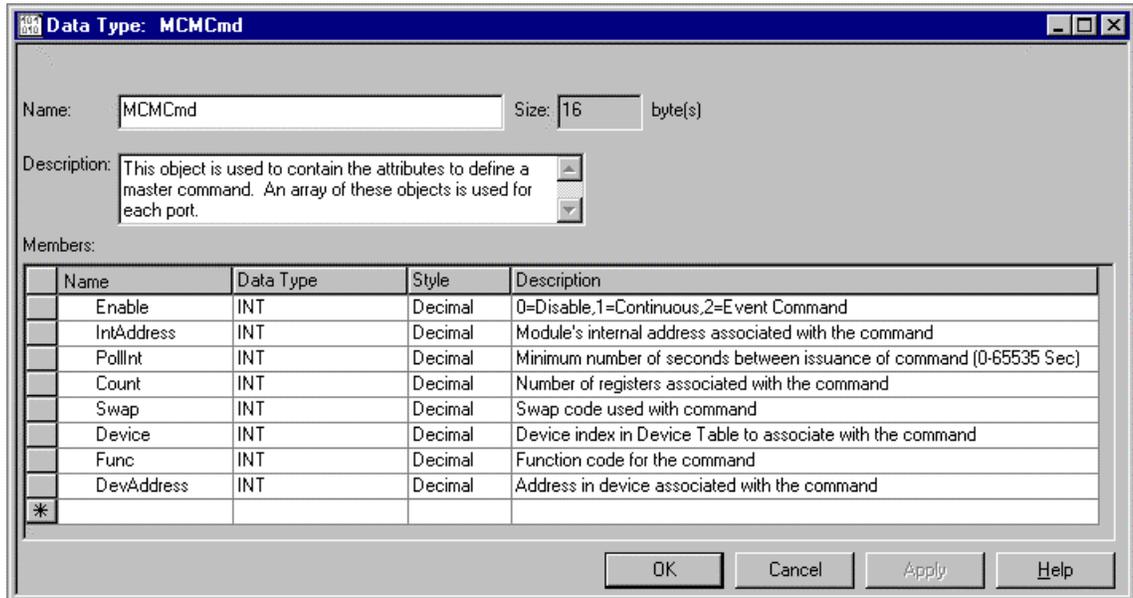
Description:

Members: Data Type Size: 52 byte(s)

Name	Data Type	Style	Description
Enabled	INT	Decimal	0=Port Disabled,1=Port Enabled
Type	INT	Decimal	0=Master, 1=Slave, 2=Slave: pass-through, 3=Slave: formatted pass-through/data swapped, 4=Slave: form. pass-through
FloatFlag	INT	Decimal	0=No floating-point data, 1=Use floating-point data
FloatStart	INT	Decimal	Register offset in message for floating-point data
FloatOffset	INT	Decimal	Internal DB offset to start of floating-point data
Protocol	INT	Decimal	0=Modbus RTU, 1=Modbus ASCII
Baudrate	INT	Decimal	Baudrate for port (110 to 115.2K)
Parity	INT	Decimal	0=None, 1=Odd, 2=Even, 3=Mark, 4=Space
DataBits	INT	Decimal	5 to 8 data bits
StopBits	INT	Decimal	1 or 2 stop bits
RTSDn	INT	Decimal	0-65535 mSec delay before data
RTSDf	INT	Decimal	0-65535 mSec delay after data
MinResp	INT	Decimal	0-65535 mSec minimum time before response to request
UseCTS	INT	Decimal	0=No, 1=Yes to use CTS modem line
SlaveID	INT	Decimal	1-255 Modbus Node Address (Slave)
BitInOffset	INT	Decimal	Internal DB offset to bit input data (Slave)
WordInOffset	INT	Decimal	Internal DB offset to word input data (Slave)
OutOffset	INT	Decimal	Internal DB offset to bit output data (Slave)
HoldOffset	INT	Decimal	Internal DB offset to holding register data (Slave)
CmdCount	INT	Decimal	Command list count (Master)
MinCmdDelay	INT	Decimal	0-65535 mSec minimum time between each command (Master)
CmdErrPtr	INT	Decimal	Internal DB location to place command error list (Master)
RespTD	INT	Decimal	0-65535 mSec response timeout for command (Master)
Retry_Count	INT	Decimal	Retry count for failed request (Master)
ErrorDelayCntr	INT	Decimal	0-65535 Command cycle count if error (Master)

Este objeto es usado para definir los parámetros para la operación de cada uno de los puertos Modbus del módulo. Refiérase al Apendice C para la definición de cada parámetro.

3.4.1.3 Comandos Maestros Modbus (MCMCmd)



Este objeto es utilizado para definir los parámetros para cada comando en la lista maestra de comandos. El objeto **MCMÓduloDef** contiene un arreglo de esos objetos que definen la lista completa para cada puerto. La definición de cada parámetro requerido para cada comando se da a continuación:

Parámetro	Descripción
Enable	Este parámetro es utilizado para definir si el comando será ejecutado ó será descartado. Los siguientes valores son válidos: 0=Inhabilita el comando y no lo ejecuta. 1=El comando será considerado por ejecución de cada scan de la lista de comandos y sera controlado por el parámetro PollInt 2=El comando solamente se ejecutará si los datos asociados con el comando han cambiado desde que el comando fue el ultimo enviado. Esta opción está sólo disponible para comandos de escritura.
IntAddress	Este parámetro especifica la dirección de registro interna de arranque a ser asociada con el comando. La entrada válida para este parámetro es desde los registros 0 hasta 4999 ó desde 0 hasta 65535 bits cuando se direcciona el comando de nivel de bit.
PollInt	Este parámetro define el número mínimo de segundos a esperar entre la ejecución de comandos continuos (Enable=1). Este comando de obtención de intervalo puede ser usado para aligerar la carga de la comunicación en una red ocupada. La entrada válida para este parámetro es 0 a 65535.
Count	Este parámetro define el número de registros a ser considerados por el comando. Una entrada válida para éste parámetro es de 1 a 125 palabras ó 2000 bits.
Swap	Este parámetro es usado para especificar si los datos utilizados en el comando deben ser alterados cuando se usa el código de función 3 Modbus para leer datos de un nodo de la red. Los valores que pueden ser asignados son los siguientes:

Parámetro	Descripción
	0=No hay inversión de los datos 1=Valores de palabra invertida 2=Palabra invertida y valores de byte 3=Valores de bite invertidos. Esta opción es utilizada cuando hay interfaz entre el módulo con datos ASCII y de coma flotante en otros dispositivos.
Device	Este parámetro es utilizado para asignar la dirección del nodo Esclavo Modbus para el módulo para alcanzar con el comando en la red Modbus. A este parámetro se le pueden asignar valores desde 0 hasta 255. La mayoría de las redes Modbus limitan el valor más alto a 247.
Func	Este parámetro especifica la función Modbus a ser ejecutada por el comando. Las entradas válidas son 1, 2, 3, 4, 5, 6, 15 y 16
DevAddress	Este parámetro define la dirección de arranque en el dispositivo que esta siendo considerado por el comando. Los valores introducidos en este campo son dependientes de la definición de la base de datos del nodo. Refiérase a la definición de la base de datos del fabricante para el dispositivo para determinar la localización de los datos a formar la interfaz.

3.4.2 Objetos de Status (MCMInStat)

Este objeto es utilizado para visualizar el status del módulo. El objeto **MCMInStat** mostrado a continuación es actualizado cada vez que es recibido un bloque de lectura. Estos datos pueden ser utilizados para monitorear el status del módulo a una "Rata de Tiempo Real".

Modifying the Módulo Configuration

Name:

Description:

Members: Data Type Size: 72 byte(s)

Name	Data Type	Style	Description
PassCnt	INT	Decimal	Program cycle counter
Product	INT[2]	Hex	Product Name
Rev	INT[2]	Hex	Revision Level Number
QP	INT[2]	Hex	Operating Level Number
Run	INT[2]	Hex	Run Number
[-] Prt1Errs	MCMPort1Errors		Port error statistics
[-] CmdReq	INT	Decimal	Total number of command list requests sent
[-] CmdResp	INT	Decimal	Total number of command list responses received
[-] CmdErr	INT	Decimal	Total number of command list errors
[-] Requests	INT	Decimal	Total number of requests for port
[-] Responses	INT	Decimal	Total number of responses for port
[-] ErrSent	INT	Decimal	Total number of errors sent
[-] ErrRec	INT	Decimal	Total number of errors received
[-] Prt2Errs	MCMPort2Errors		
[-] CmdReq	INT	Decimal	Total number of command list requests sent
[-] CmdResp	INT	Decimal	Total number of command list responses received
[-] CmdErr	INT	Decimal	Total number of command list errors
[-] Requests	INT	Decimal	Total number of requests for port
[-] Responses	INT	Decimal	Total number of responses for port
[-] ErrSent	INT	Decimal	Total number of errors sent
[-] ErrRec	INT	Decimal	Total number of errors received
[-] BlkErrs	MCMBlkStat		Block transfer statistics
[-] Read	INT	Decimal	Total number of read block transfers
[-] Write	INT	Decimal	Total number of write block transfers
[-] Parse	INT	Decimal	Total number of blocks parsed
[-] Event	INT	Decimal	Total number of event blocks received
[-] Cmd	INT	Decimal	Total number of command blocks received
[-] Err	INT	Decimal	Total number of block transfer errors
Port1CurErr	INT	Decimal	Current error/index for Port 1
Port1LErr	INT	Decimal	Last error/index for Port 1
Port2CurErr	INT	Decimal	Current error/index for Port 2
Port2LErr	INT	Decimal	Last error/index for Port 2

*

Refiérase al Apéndice B para un listado completo de los datos almacenados en este objeto.

3.5 Objetos de Datos del Usuario

Estos objetos son utilizados para mantener los datos a ser transferidos entre el procesador y el módulo MVI56-MCM. Los datos del usuario son los datos de lectura y escritura transferidos entre el procesador y el módulo como “paginas” de datos de hasta 200 palabras de longitud.

ReadData	INT[600]	Decimal	Data read from module
WriteData	INT[600]	Decimal	Data to write to module

Los datos de lectura (ReadData) son un arreglo ajustado para igualar el valor introducido en el parámetro ReadRegCnt del objeto MCM Módulo. Para su uso facil, este arreglo

debería dimensionarse como unos incrementos exactos de 200 palabras. Estos datos son paginados hasta 200 palabras por vez desde el módulo hasta el procesador.

Los datos de escritura (**WriteData**) son un arreglo ajustado para igualar el valor introducido en el parámetro **WriteRegCnt** del objeto **MCM**Módulo. Para su uso fácil, este arreglo debería dimensionarse como unos incrementos exactos de 200 palabras. Estos datos son paginados hasta 200 palabras por vez desde el procesador hasta el módulo.

El trabajo de Datos de Escritura es el responsable por la colocación de los datos escritos en la imagen de salida para ser transferidos al módulo. Estos datos son pasados desde el procesador hacia el módulo para la información de control y status a ser usados por otros nodos en la red. Si este arreglo es > 600 registros, cambie el valor High LIM en el rung (peldaño) 2 de ReadData y el rung 10 de WriteData del archivo de escalera.

3.6 Control y Status para Interrogar al Esclavo

Dos arreglos están localizados en el objeto primario del módulo para mantener el status de interrogaciones de cada Esclavo en los puertos Maestros. Estos datos de status pueden ser utilizados para determinar cuales de los Esclavos están actualmente activos en el puerto están en error de comunicación o tienen sus interrogaciones suspendidas e inhabilitadas. La lógica de escalera en el procesador puede ser escrita para monitorear y controlar el status de cada Esclavo en un puerto Maestro. El objeto utilizado es mostrado en el siguiente diagrama:

P1Slaves	INT[256]	Decimal	Port 1 slave status values
P2Slaves	INT[256]	Decimal	Port 2 slave status values

Utilizando bloques especiales, el procesador puede solicitar los datos actuales para los Esclavos. A través del uso de otros bloques, el procesador puede habilitar o deshabilitar la interrogación de los Esclavos seleccionados.

3.7 Datos de Mensaje Modbus

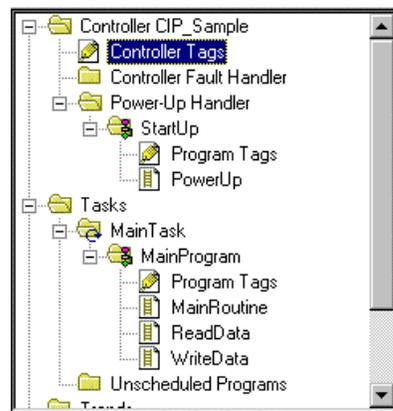
Esta nueva versión del programa del módulo incluye el modo de operación Pass-through. En este modo, los mensajes escritos se envían a un puerto Esclavo y son pasados directamente hacia el procesador. Es responsabilidad de la lógica de escalera procesar el mensaje recibido utilizando esta característica. Se requieren dos objetos de datos para este modo de operación: una variable para mantener la longitud del mensaje y un buffer para mantener el mensaje.

Esta información es pasada desde el módulo hacia el procesador utilizando el código de identificación de bloque 9996. La segunda palabra de este bloque contiene la longitud del mensaje y el mensaje comienza en la tercera palabra. Otros Tags del Controlador se requieren para almacenar los valores controlados contenidos en estos mensajes. El protocolo Modbus soporta un controlador de salida binaria (bobinas-funciones 5 y 15) y registros (funciones 6 y 16).

4 Modificando la Lógica de Escalera Ejemplo

La lógica de escalera requerida para la aplicación del módulo MVI56-MCM. Las tareas que deben ser manejadas por la lógica de escalera son la configuración del módulo, la transferencia de datos, el manejo especial de bloque y la recepción de datos de estado. Esta sección discute cada aspecto de la lógica de escalera como es requerida por el módulo. Adicionalmente un manejador de alimentación debe ser escrito para la inicialización de los datos del módulo y para borrar cualquier condición de falla del procesador.

La ventana de Organización del Controlador para lógica de escalera de ejemplo para el módulo MVI56-MCM se muestra en el siguiente ejemplo:



4.1 Energización (PowerUp)

La lógica de escalera de energización es utilizada para inicializar los objetos de datos utilizados por el módulo MVI56-MCM y para reestablecer las fallas del controlador en la energización inicial del procesador. La lógica de escalera requerida para ejecutar éstas tareas se muestran en los siguientes parrafos:

Este rung es utilizado para reestablecer una condición de falla del procesador debido a una pérdida de energía y reinicio cuando el procesador esta modo Run. Usted puede tener que manejar también otras condiciones de falla. Adicionalmente un manejador de falla puede ser escrito para el procesador para manejar otras fallas. El objeto MJFaults debe ser definido en los Tags de Controlador antes de que pueda ser usado en esta lógica:

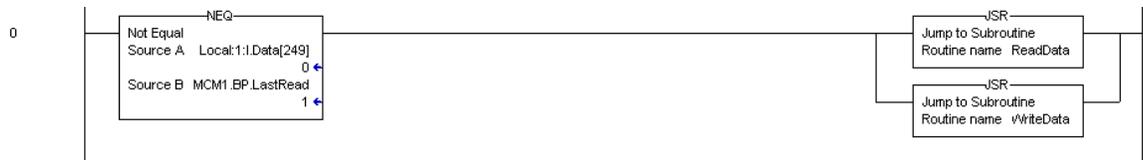


Este rung es usado para inicializar en cero los ultimos valores de lectura y escritura, la imagen de salida para el módulo MVI56-MCM y el área de datos de escritura. Los ultimos valores de lectura (**MCM.BP.LastRead**) y escritura (**MCM.BP.LastWrite**) son usados en la lógica de transferencia de datos. La imagen de salida para el módulo MVI56-MCM (**Local:1:O.Data[]**) es utilizada para transferir datos desde el procesador hasta el módulo. El area de datos de escritura (**MCM.WriteData[]**) es utilizada para almacenar los datos del procesador a ser escritos en el módulo utilizando la imagen de salida.



4.2 Rutina Principal

La rutina principal es usada para reconocer la presencia de nuevos datos de lectura desde el módulo por el procesador. El módulo hará un ciclo a través de su lista de bloques de lectura para transferir datos desde el módulo hasta el procesador. Cada vez que un nuevo dato está disponible, el módulo ajustará el valor para el bloque en la imagen de entrada del módulo (**Local:1:I.Data[249]**). La lógica de escalera debe escanear constantemente ésta palabra de entrada para un valor nuevo. Cuando un nuevo valor está presente, la lógica de escalera debería ejecutar las tareas ReadData y WriteData en ese orden.

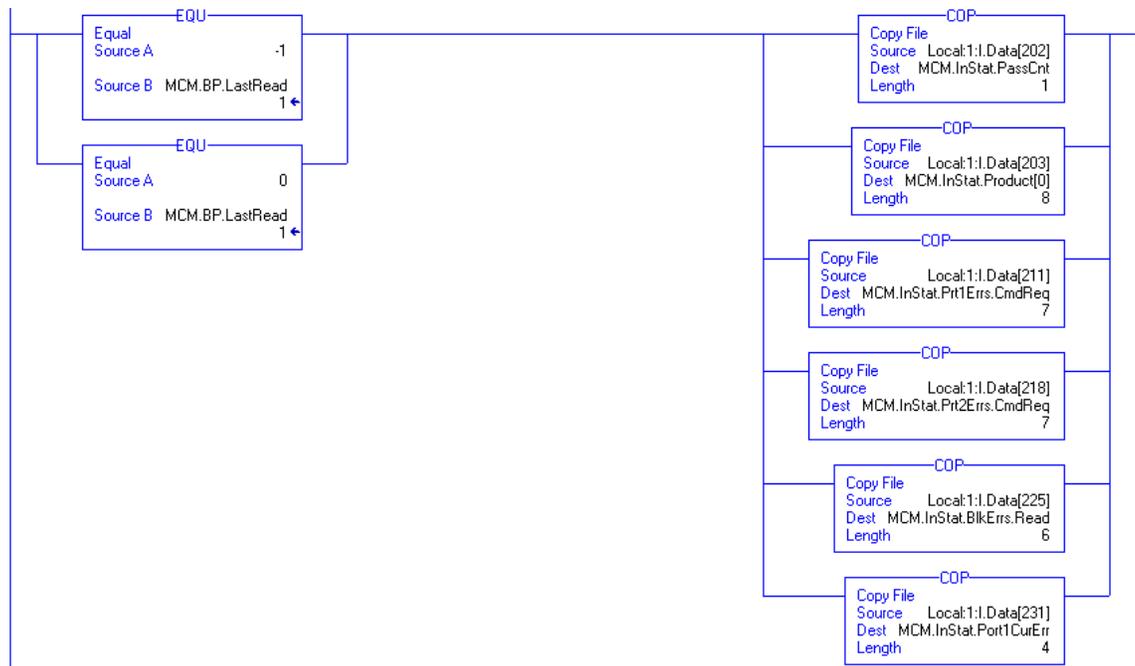


4.3 Tarea de Datos de Lectura (ReadData)

La tarea de lectura de datos (ReadData) es la responsable de manejar todos los nuevos datos recibidos desde el módulo y colocados en la locación apropiada en el procesador. Los datos son transferidos desde el módulo al procesador utilizando la imagen de entrada del módulo (**Local:1:I.Data[]**). El primer rung de la tarea ajusta el ultimo número de bloque leído (**MCM1.BP.LastRead**) al número de bloque actual enviado desde el módulo (**Local:1:I.Data[249]**).



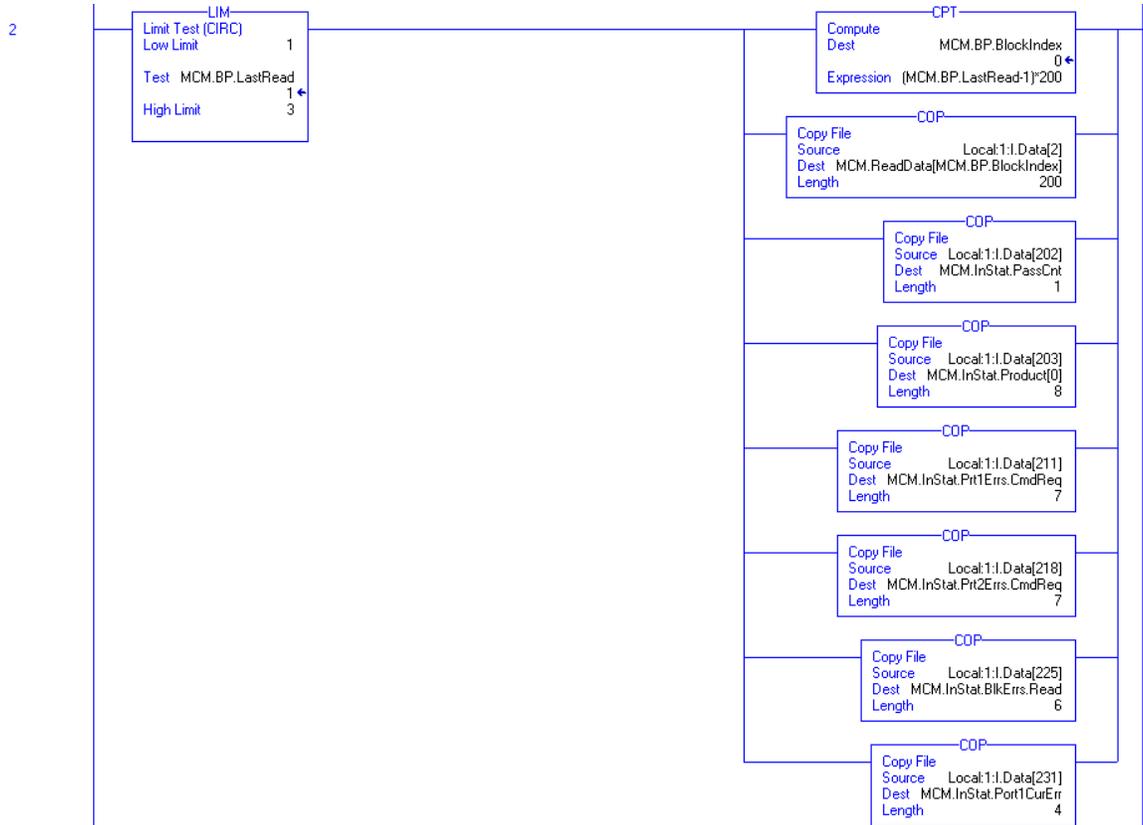
Si el módulo está configurado para los bloques cero o uno, enviará bloques con código de identificación de 0 y -1. Ninguno de los datos del usuario será incluido en éstos bloques. Ellos sólo contendrán el estado de los datos. El rung presentado abajo muestra la lógica que maneja esos bloques.



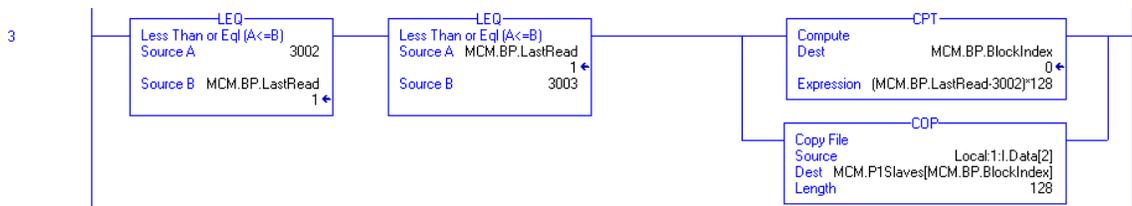
El próximo rung de la lógica de escalera determina si los nuevos datos recibidos en la imagen de entrada son datos de usuario. Si los datos de usuario están presentes, la lógica de escalera colocará los datos en la localización correcta en el área de datos de lectura del procesador (**MCM.ReadData[]**).

Modifying the Sample Ladder Logic

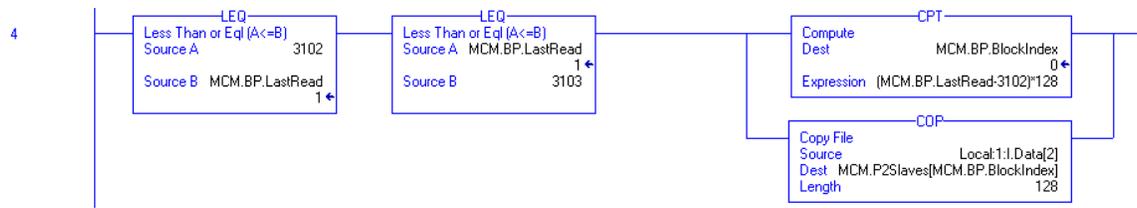
Se pueden transferir hasta 200 palabras de datos en cada transferencia de bloques. En adición a los datos de usuario, el bloque contiene también importantes datos de estado. Estos datos deberían ser copiados al área de datos correcta en el módulo (**MCM.InStat**). Estos datos de estado pueden ser utilizados para determinar la “salud” del módulo MVI56-MCM.



Los siguientes dos rungs de la lógica de escalera son usados para manejar el recibo de datos de estado del nodo Esclavo. Estos bloques son requeridos por el procesador en la tarea WriteData y enviados desde el módulo hasta el procesador. Los dos rung a continuación muestran la lógica requerida para procesar estos bloques.

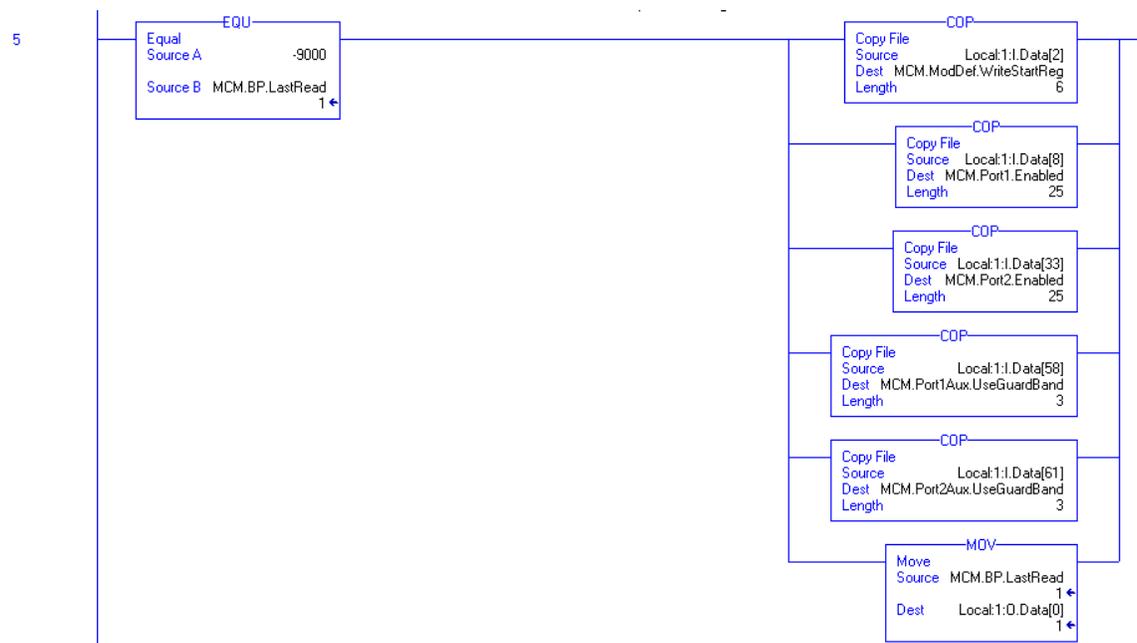


Este rung es utilizado para manejar Esclavos conectados al puerto Modbus 1. Dos bloques de 128 Esclavos cada uno son procesados por el rung y los datos son almacenados en la localización de arreglo apropiada.

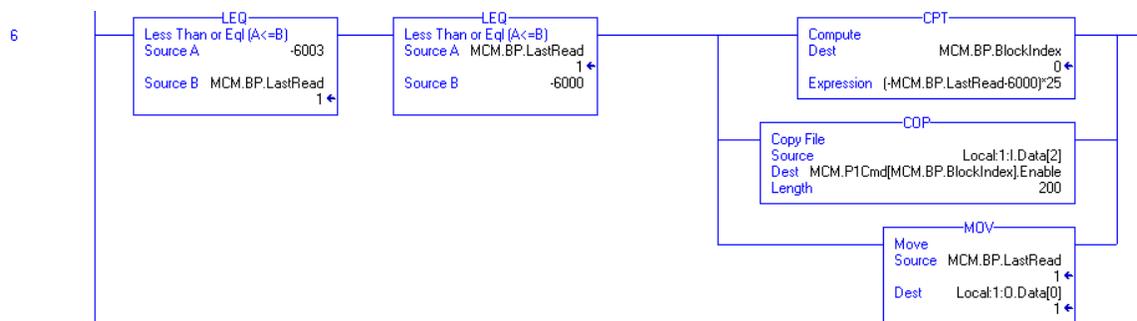


Este rung es utilizado para manejar Esclavos conectados al puerto Modbus 2.

Si el procesador está recibiendo la configuración del módulo desde una fuente remota a través de la base de datos del módulo, debería ser programado para manejar bloques especiales. La información de configuración es transferida desde el módulo hacia el procesador a través de los bloques -9000, -6000 hasta -6003 y -6100 hasta -6103. La lógica de escalera para manejar esta función es mostrada en el ejemplo siguiente:

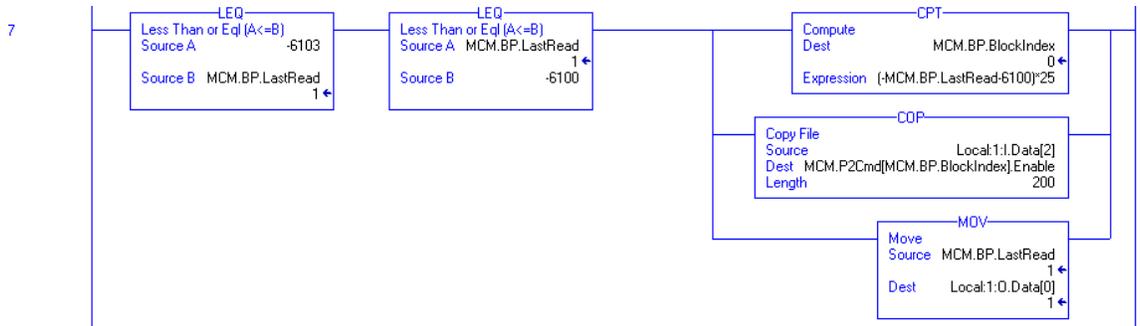


Este rung es utilizado para procesar el recibo de información de configuración general por el módulo.



Este rung es utilizado para manejar la recepción de datos de la lista maestra de comandos por el puerto Modbus 2.

Modifying the Sample Ladder Logic



Este rung es utilizado para manejar los datos de la lista maestra de comandos por el puerto Modbus 3. Otros bloques pueden ser manejados en futuras actualizaciones del producto.

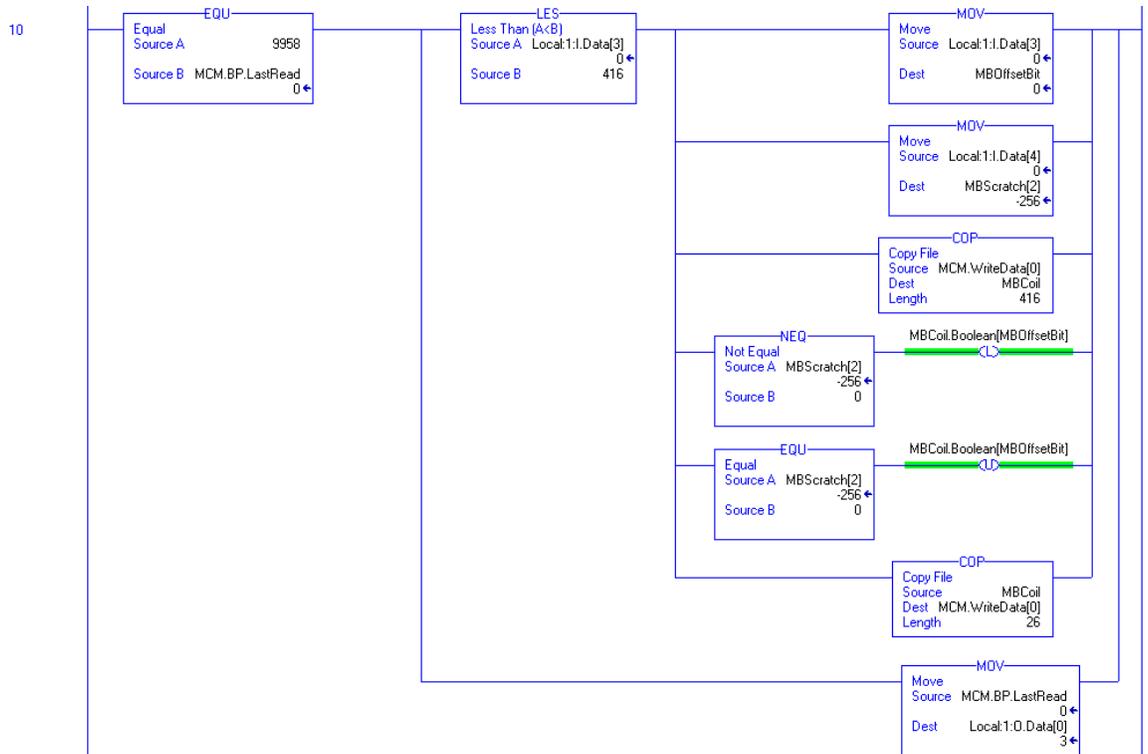
El siguiente rung es utilizado para copiar un mensaje pasado al procesador directamente desde un host remoto a través del módulo (modo Pass-Through).



Este rung es utilizado para manejar los códigos de función 6 y 16 requeridos cuando el módulo esta siendo usado en el modo Pass-Through formateado.



Este rung es usado para manejar código de función 5 requerido cuando el módulo está siendo utilizado como un Esclavo en el modo Pass-Through formateado:



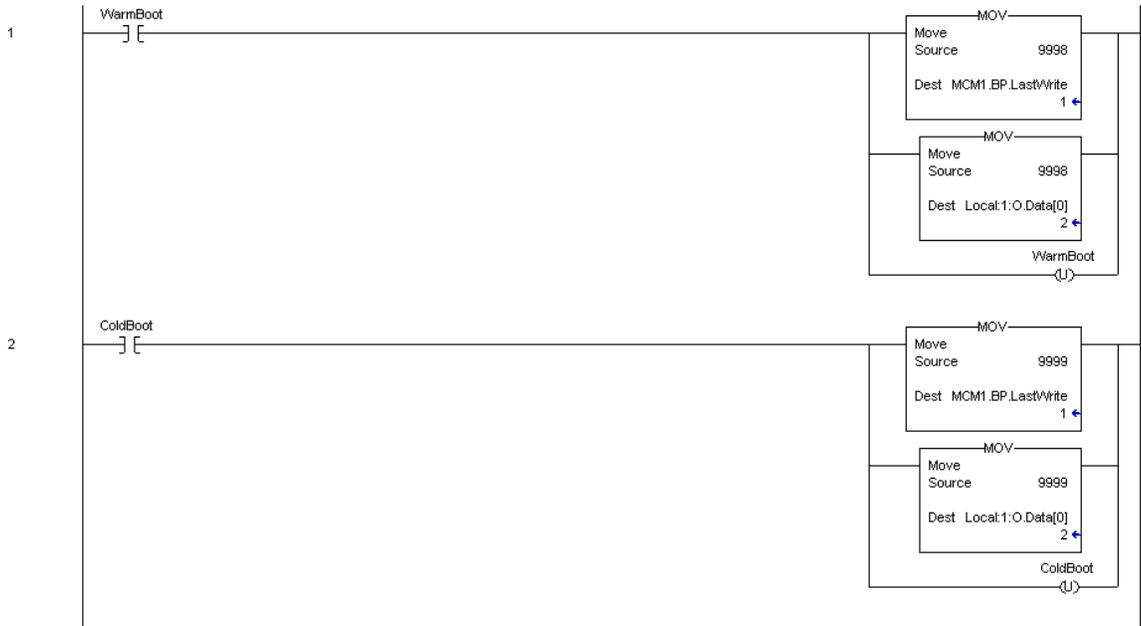
Note: Nota: El rung 11 (no mostrado) es utilizado por la lógica Pass-Through para el código de función 15. Este rung no debería ser alterado.

4.4 Tarea de Datos de Escritura (WriteData)

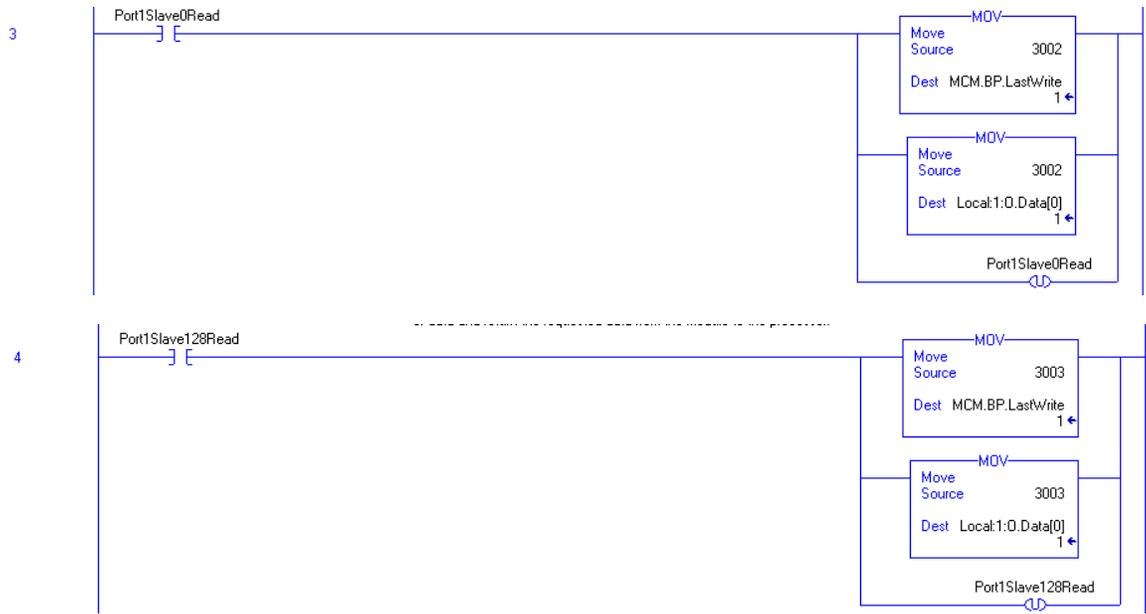
La tarea WriteData es la responsable por el envío de datos desde el procesador hacia el módulo MVI56-MCM. Los datos son transferidos desde el procesador hacia el módulo utilizando la imagen de salida del módulo (**Local:1:O:Data[]**). El primer rung es utilizado para almacenar los datos actuales requeridos ajustados en el objetos de datos **MCM.BP.LastWrite** del módulo. Este objeto es utilizado en toda la lógica de escalera subsecuente en caso de cambios en la palabra de entrada (**Local:1:I:Data[1]**) durante el procesamiento.



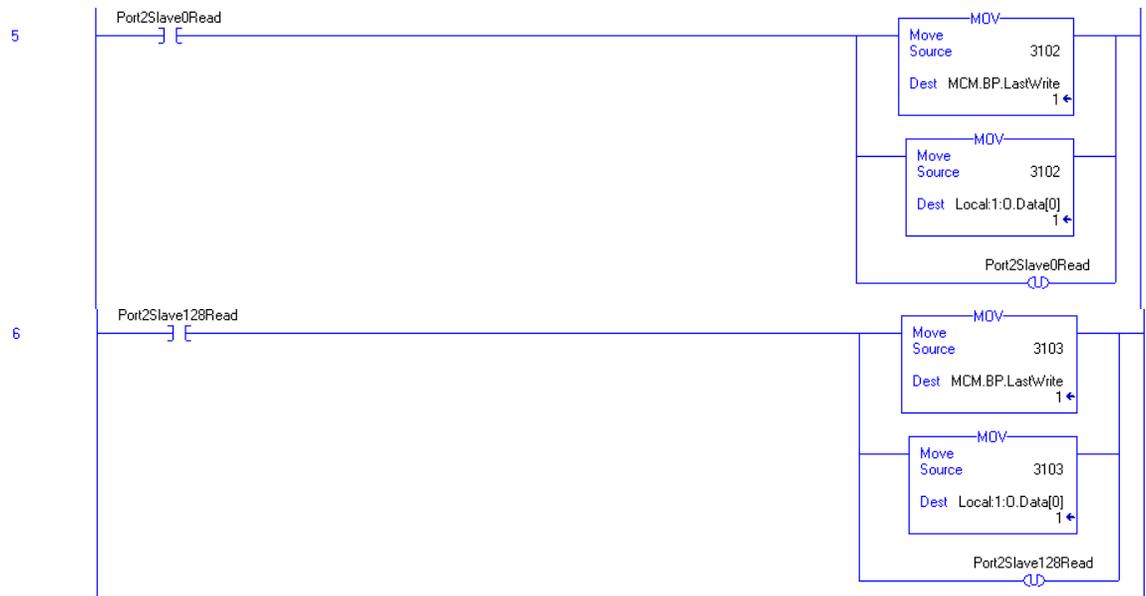
Los próximos dos rungs son utilizados para manejar el control del procesador del módulo utilizando los números de bloque de control de arranque en frio y caliente. Cuando el procesador requiere que el módulo ejecute una de éstas operaciones, simplemente copia el número del bloque dentro de la imagen de salida del módulo y el módulo ejecutará la operación. Asegurese de ajustar el número de bloques requerido en el ultimo objeto de escritura para prevenir procesamientos posteriores en la tarea WriteData. Ejemplos de cada bloque de control se presentan en los siguientes rungs.



Los siguientes cuatro rungs son utilizados para requerir los datos de estado del nodo Esclavo asociado con cada puerto Maestro. Dos solicitudes son requeridas para cada puerto de manera de obtener los datos para los potenciales 256 – Esclavos direccionados en el puerto. La siguiente lógica de escalera muestra lo que se requiere para obtener los datos del puerto Modbus 1.



Los próximos dos rungs muestran la lógica de datos de control/status para el puerto Modbus 2 Esclavo.



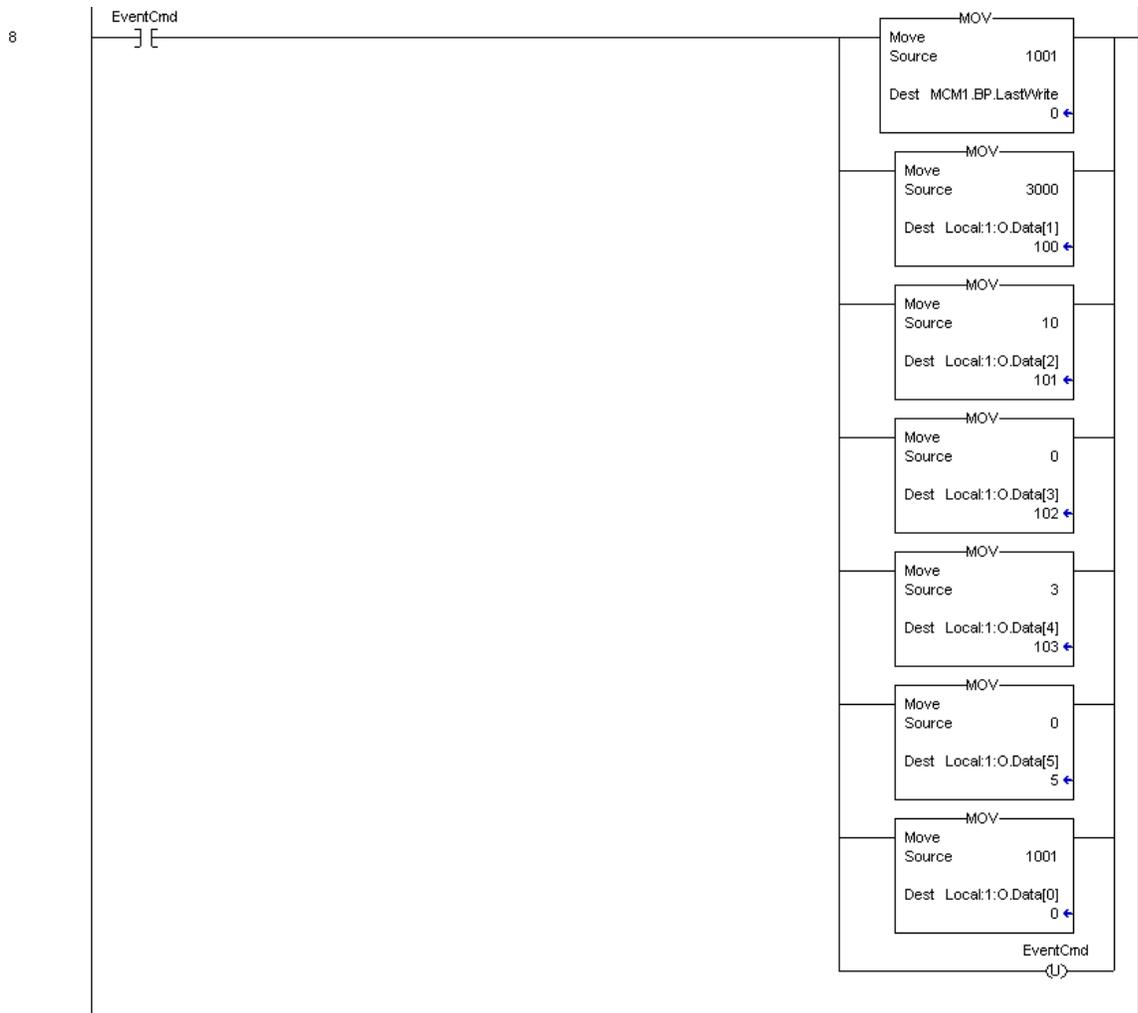
Modifying the Sample Ladder Logic

El siguiente rung muestra un ejemplo de control de comando. Este bloque de datos es pasado desde el procesador hasta el módulo para ejecutar un comando en una lista maestra de comandos del puerto.



Cuando el bit CmdControl es ajustado, el puerto comando Maestro 1 (indice 0) será colocado en la cola comando y ejecutado. Hasta 6 comandos pueden ser transferidos con una solicitud desde la lista de comandos hasta la cola de comandos.

El próximo rung es usado para enviar un mensaje de evento (mensaje construido por el usuario) en el puerto 1.



Cuando el bit EventCmd es ajustado, el rung será ejecutado. Coloca el comando contenido en la cola comando para su ejecución. Esta técnica puede ser usada para enviar comandos a un puerto sin tener que construir una lista maestra de comandos o ejecutar comandos que son enviados bajo condición especial (por ejemplo, un comando de reset que debería ser ejecutado una vez al día, semanalmente, etc.).

Si el módulo es configurado para que no transfiera bloques o para que transfiera un solo bloque, se requiere un proceso especial. El módulo debe observar la primera palabra de la imagen de salida del módulo cambiando de manera de reconocer la recepción de los nuevos datos. Si el valor nunca cambia, el módulo no procesará los datos. Esto representa un problema cuando menos de dos bloques son transferidos al módulo desde el procesador. Para resolver este problema el módulo enviara -1 y 0 en la palabra de entrada. Cuando el módulo es configurado para cero bloques de escritura, la siguiente secuencia de bloques requerida presentara: -1, 0, -1, 0, -1, 0...

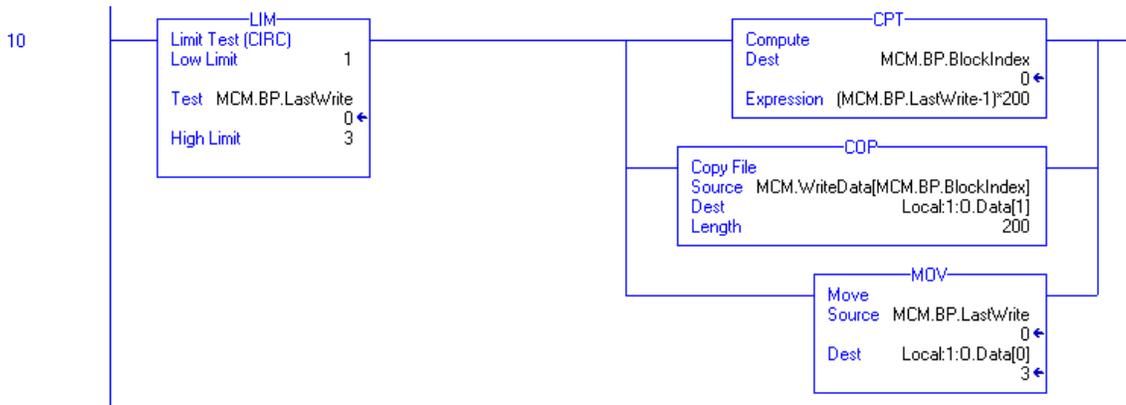
Cuando el módulo es configurado para un bloque de escritura, la siguiente secuencia de solicitud de bloques presentara: 1, 0, 1, 0, 1, 0...

El siguiente rung es requerido para manejar éstas condiciones.

Modifying the Sample Ladder Logic

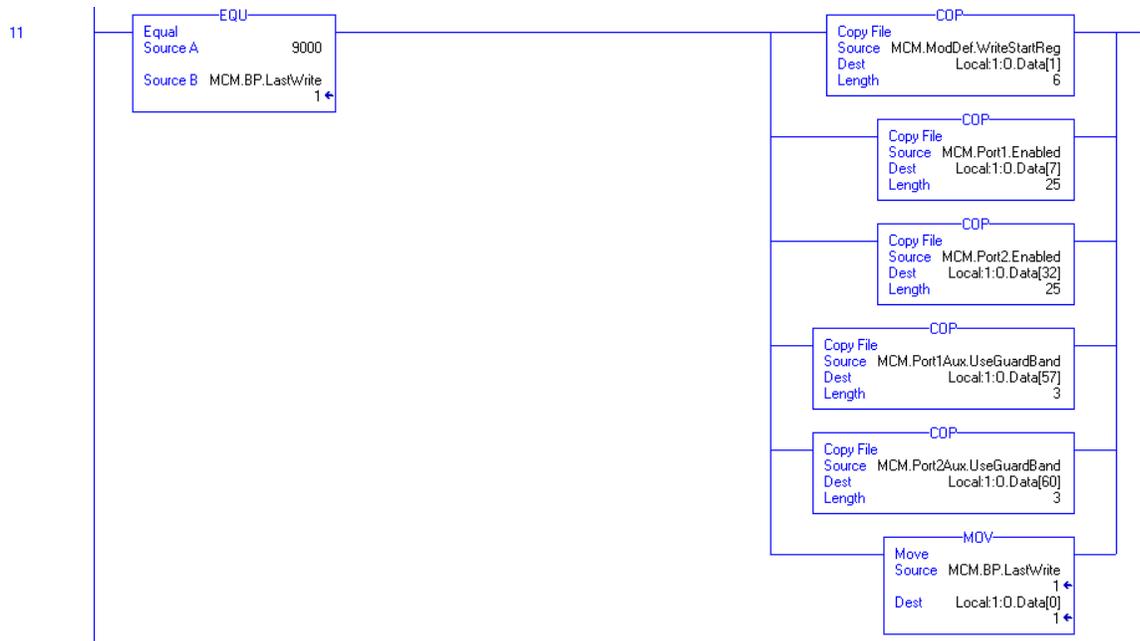


El siguiente rung en la lógica de escalera es el más importante. Maneja la transferencia de datos del procesador hacia el módulo. Hasta 200 palabras de datos del usuario son mantenidas en el procesador (**MCM.WriteData[]**) pueden ser transferidas al módulo cada vez.

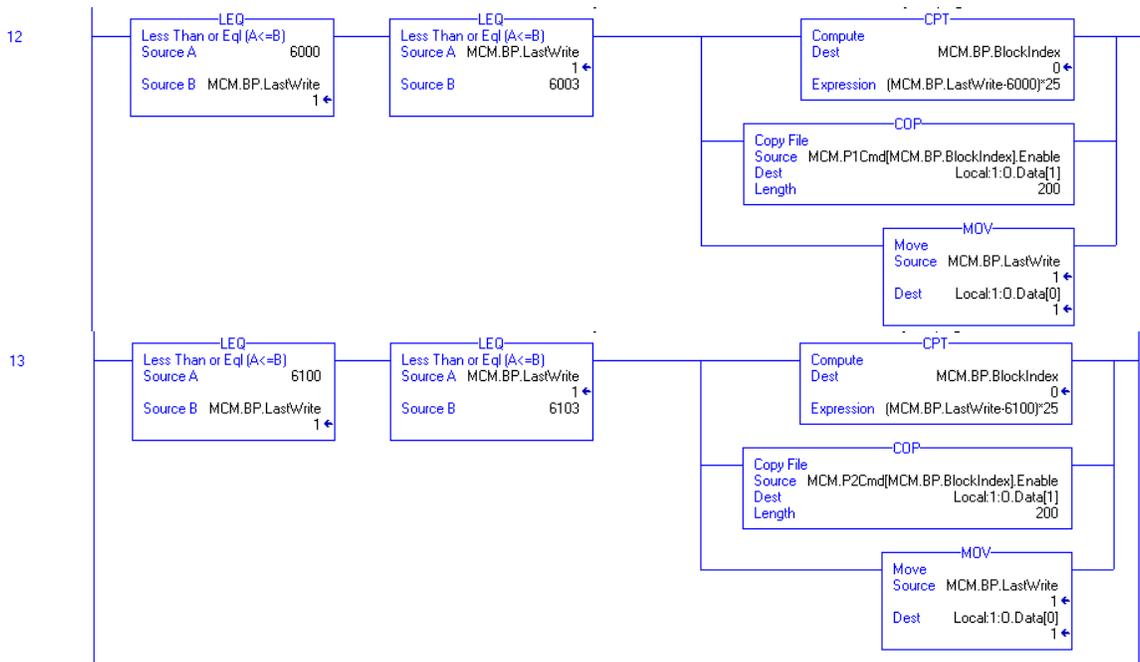


De forma de configurar el módulo, la configuración debe ser transferida desde la tabla de datos del procesador hacia el módulo. Se requieren varios bloques para transferir toda la información requerida por el módulo. Cada uno de éstos bloques debe ser programado y manejado por el módulo para ser ejecutado.

El primer bloque de configuración tiene un valor de código de 9000. Este bloque es usado para transferir la información del tamaño de los bloques de datos (MCM.ModDef) y la información de configuración del puerto Modbus (MCM.Port[]). Este es el primer juego de datos requerido por el módulo cuando es iniciado.



El último juego de información de configuración requerida desde el módulo es la lista maestra de comandos para cada puerto. Esta lista es transferida a los 25 comandos del módulo a un mismo tiempo. La lógica de escalera para transferir la lista de comandos hacia el módulo es mostrada en los siguientes rungs:



5 Diagnóstico y Troubleshooting

El módulo provee información de diagnóstico en tres formas para el usuario: 1) Los valores de datos de estado son transferidos desde el módulo hacia los archivos de datos definidos en el procesador ControlLogix. 2) Todos los datos contenidos en el módulo pueden ser visualizados a través del puerto de Configuración/Depuración en un terminal emulador conectado. 3) Los LED's indicadores de estado al frente del módulo proveen información sobre el estado de los módulos.

Las siguientes secciones explican como obtener datos de estado desde el módulo y el significado de los LED's individuales en el módulo.

5.1 Leyendo los Datos de Status desde el Módulo

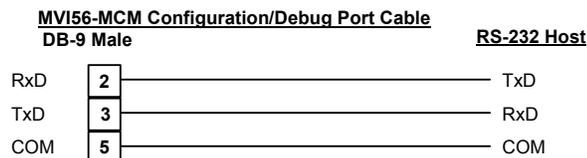
El módulo MVI56-MCM retorna un bloque de datos de estado de 29 palabras que puede ser usado para determinar los estados de operación del módulo. Estos datos están localizados en la base de datos del módulo entre los registros 6670 y 6698 y en las localidades especificadas en la configuración. Estos datos son transferidos de forma continua hacia el procesador ControlLogix con cada bloque leído. Para un listado completo de los objetos de datos de Esclavo, refiérase a la sección de **Inicialización del Módulo**.

5.1.1 Hardware Requerido

Los requerimientos de hardware para hacer interfaz con el puerto de Configuración/Depuración no son tan estrictos. Un computador personal con un puerto serial estándar sería suficiente. Para un desempeño óptimo, lo mínimo que se requiere es:

- Procesador Basado en 80486 (Pentium Preferiblemente)
- 1 Megabyte de memoria
- Al menos un puerto de comunicación serial disponible

Adicionalmente, un cable null-modem es suministrado con el módulo para proveer la comunicación entre su PC y el puerto. El puerto del módulo tiene un conector macho DB-9 al final de un conector convertidor RJ-45 a DB-9. El terminal RJ-45 del cable es colocado en el conector del puerto 1 del MVI56-MCM (Puerto Superior). El cable requerido es mostrado en el siguiente diagrama:



5.1.2 Software Requerido

El software requerido en su computador personal para hacer interfaz con el puerto de Configuración/Depuración es independiente del sistema operativo. El software probado incluye lo siguiente:

DOS	Procomm, PS-Term y otros programas de emulación para terminal
Windows 3.1	Terminal
Windows 95/98	HyperTerminal y PS-Term
Windows NT	HyperTerminal
Linux	Minicom

Cualquier paquete de software de emulación de terminal provisto con su sistema operativo debería funcionar siempre y cuando pueda ser configurado como sigue:

Rata de Baudios	57,600
Paridad	Ninguna
Bits de Datos	8
Bits de Parada	1
Protocolo de Transferencia de Archivo	Zmodem

5.1.3 Utilizando el Puerto

Los siguientes pasos son requeridos para la interfaz en el puerto de Configuración/Depuración:

1. Conecte su computador al puerto del módulo utilizando un cable null-modem.
2. Inicie el programa de emulación de terminal en su computador y configure los parámetros de comunicación como los mostrados en la sección de software requerido (57600K, N, 8, 1).
3. Introduzca el carácter '?' en su computador. Si todo esta configurado correctamente, se mostrará el menú del puerto.

Si no hay respuesta desde el módulo, verifique la configuración de comunicación y el cableado. Adicionalmente, asegurese de que se ha conectado al puerto correcto tanto en su computador como en el módulo.

5.1.4 Opciones de Menú

Las características disponibles a través del uso del puerto de Configuración/Depuración en el módulo MVI56-MCM se obtienen utilizando sencillamente teclas de función en su computador. Existe un menú principal sencillo y varios submenús presentados en el puerto. Para ver las selecciones actuales disponibles, presione la tecla '?' en su computador. Si usted esta en el modo de menú principal, aparecerá el siguiente menú:

```

MODBUS MASTER/SLAVE COMMUNICATION MODULE (MVI56-MCM) MENU
?=Display Menu
A=Data Analyzer
B=Block Transfer Statistics
C=Module Configuration
D=Modbus Database View
Master Command Errors : E=Port 1   F=Port 2
Master Command List  : I=Port 1   J=Port 2
Slave Status List    : O=Port 1   P=Port 2
U=Version Information
W=Warm Boot Module
Y=Transfer Module Cfg to Processor
Communication Status : 1=Port 1   2=Port 2
Port Configuration   : 6=Port 1   7=Port 2

Esc=Exit Program

```

Si éste menú no es presentado, presione la tecla 'M' para mostrar el menú principal. Todas las facilidades ofrecidas para la configuración/depuración se muestran en el menú principal. Cada opción es discutida en las siguientes secciones.

5.1.4.1 A=Analizador de Datos

La selección de esta opción de menú coloca el programa en el modo de menú analizador. Este modo de operación es usado para mostrar los mensajes Modbus generados y recibidos por el módulo. Para visualizar las opciones de menú disponible en este modo, presione la tecla '?' y el siguiente menú aparecerá:

```

Data Analyzer Mode Selected

MODBUS DATA ANALYZER VIEW MENU
?=Display Menu
1=Select Port 1
2=Select Port 2
5=1 mSec Ticks
6=5 mSec Ticks
7=10 mSec Ticks
8=50 mSec Ticks
9=100 mSec Ticks
0=No mSec Ticks
H=Hex Format
A=ASCII Format
B=Start
S=Stop
M=Main Menu

Port = 1, Format=HEX, Tick=10

```

Esta herramienta es extremadamente útil en la determinación de la operación del módulo y los nodos sobre la red de cada puerto. Los parámetros en la parte baja del display muestran los ajustes actuales del analizador. Cada una de las opciones de este menú son discutidas en las siguientes secciones.

5.1.4.1.1 1=Selección del Puerto 1

Esta opción selecciona el Puerto Modbus 1 para su análisis. Los datos mostrados en el modo analizador serán los relativos a este puerto.

5.1.4.1.2 2= Selección del Puerto 2

Esta opción selecciona el Puerto Modbus 2 para su análisis. Los datos mostrados en el modo analizador serán los relativos a este puerto.

5.1.4.1.3 5=Ticks de 1 mSeg

Esta opción genera marcas de tiempo de 1-milisegundo en el display. Esto puede ayudar cuando se están determinando las características de tiempos de comunicación.

5.1.4.1.4 6= Ticks de 5 mSeg

Esta opción genera marcas de tiempo de 5-milisegundo en el display. Esto puede ayudar cuando se están determinando las características de tiempos de comunicación.

5.1.4.1.5 7= Ticks de 10 mSeg

Esta opción genera marcas de tiempo (Ticks) de 10-milisegundo en el display. Esto puede ayudar cuando se están determinando las características de tiempos de comunicación.

5.1.4.1.6 8= Ticks de 50 mSeg

Esta opción genera marcas de tiempo de 50-milisegundo en el display. Esto puede ayudar cuando se están determinando las características de tiempos de comunicación.

5.1.4.1.7 9= Ticks de 100 mSeg

Esta opción genera marcas de tiempo de 100-milisegundo en el display. Esto puede ayudar cuando se están determinando las características de tiempos de comunicación.

5.1.4.1.8 0= Ticks de 0 mSeg

Esta opción hace inactivo el display de marcas de tiempo.

5.1.4.1.9 H=Formato Hexadecimal

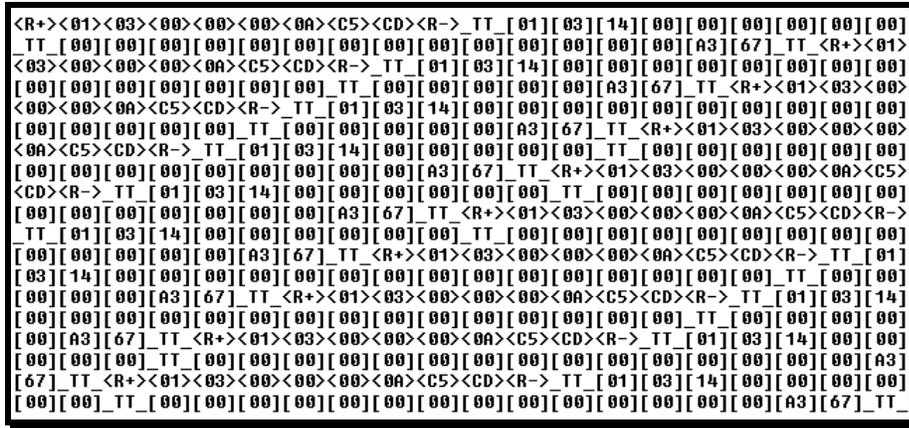
Esta opción selecciona el display de datos en formato hexadecimal. Este formato es mas util cuando se visualizan mensajes en protocolo Modbus RTU.

5.1.4.1.10 A=Formato ASCII

Esta opción selecciona el display de datos en formato ASCIIhexadecimal. Este formato es mas util cuando se visualizan mensajes en protocolo Modbus ASCII.

5.1.4.1.11 B=Start

Esta opción inicia el analizador de datos. Despues que la tecla es presionada, todos los datos transmitidos y recibidos sobre el puerto actual seleccionado son mostrados. Un display de ejemplo es mostrado a continuacion:



La siguiente tabla describe los caracteres especiales utilizados en el display:

[]	Los datos encerrados en corchetes representan datos recibidos por el puerto.
< >	Los datos encerrados en éstos caracteres representan datos transmitidos por el puerto.
<R+>	Estos caracteres son insertados cuando una línea RTS tiene estado alto en el puerto.
<R->	Estos caracteres son insertados cuando una línea RTS pasa a estado bajo en el puerto.
<CS>	Estos caracteres son mostrados cuando una línea CTS es reconocida como alta.
TT	Estos caracteres son mostrados cuando el intervalo de marca de tiempo ha sido alcanzado. Este parámetro es definido por el usuario.

5.1.4.1.12 S=Stop

Esta opción detiene el analizador. Utilice esta opción para congelar el display de forma que los datos puedan ser analizados. Para reiniciar el analizador, presione la tecla 'B'.

Advertencia -- Cuando se retoma el modo analizador, la ejecución del programa será lenta (slow down). Únicamente utilice ésta herramienta durante sesiones de troubleshooting. Inhabilite el analizador antes de dejar que el módulo corra en su modo normal.

5.1.4.1.13 M = Menú Principal

Esta opción de menú es utilizada para retornar al modo de menú principal.

5.1.4.2 B=Estadísticas de Transferencia de Bloque

Esta opción de menú muestra la configuración y estadísticas de las operaciones de transferencia de datos del backplane. Después de seleccionada esta opción, es mostrado lo siguiente.

Seleccionando esta opción a intervalos de un segundo puede ser usada para determinar el número de bloques transferidos cada segundo.

```
BACKPLANE STATISTICS:
DATA TRANSFER CONFIGURATION:
WRITE DATA TRANSFER:
  Start : 0      Count : 400      Max Blocks : 2      Last : 2
READ DATA TRANSFER:
  Start : 0      Count : 600      Max Blocks : 3      Last : 2
BLOCK COUNTS:
  Retry : 20     Failed: 0      Fail Cnt: 0
  Read  : 43009  Write : 43021  Parsing : 43008
  Error : 59833  Event : 0      Command : 0
```

5.1.4.3 C=Configuración del Módulo

Esta opción de menú muestra información de configuración general del módulo para el módulo MVI56-MCM. Después de seleccionada esta opción, aparece la siguiente pantalla.

```
MODULE CONFIGURATION:
MVI56-MCM, ProSoft Technology, Inc.
DATABASE:
  Err/Stat Blk Pointer : 600
BLOCK TRANSFER:
  READ -- Start : 0      Count : 600      Max : 3
  WRITE -- Start : 0     Count : 400      Max : 2
  FAIL COUNT : 20
```

5.1.4.4 D=Visualización de la Base de Datos Modbus

Seleccionando esta opción se coloca el programa en el modo de menú de visualización de la base de datos. Este modo de operación muestra los valores de la base de datos interna del módulo. Para visualizar las opciones de menú disponible en este modo, presione la tecla '?' y el siguiente menú aparece.

```
MODBUS DATABASE VIEW MENU
?-Display Menu
0-6=Display 0-6000
S=Show Again
--=Back 5 Pages
P=Previous Page
+=Skip 5 Pages
N=Next Page
D=Decimal Display
H=Hexadecimal Display
F=Float Display
A=ASCII Display
M=Main Menu
```

Todos los datos contenidos en la base de datos del módulo están disponibles para su visualización utilizando las opciones de menú. Cada opción de menú disponible es discutida en las siguientes secciones.

5.1.4.5 0-9 Páginas de Registro 0-9000

Esta opción de menú salta a un juego específico de registros en la base de datos y muestra los datos. Las teclas ejecutan las siguientes funciones:

Tecla	FUNCIÓN
0	Muestra Registros del 0 al 99
1	Muestra Registros del 1000 al 1099
2	Muestra Registros del 2000 al 2099
3	Muestra Registros del 3000 al 3099
4	Muestra Registros del 4000 al 4099
5	Muestra Registros del 5000 al 5099
6	Muestra Registros del 6000 al 6099
7	Muestra Registros del 7000 al 7099
8	Muestra Registros del 8000 al 8099
9	Muestra Registros del 9000 al 9099

5.1.4.6 S=Mostrar de Nuevo

Esta opción de menú muestra la página actual de 100 registros en la base de datos. Una salida de ejemplo de la base de datos es mostrada en el siguiente ejemplo:

```

MODBUS DATABASE DISPLAY 0 TO 99 (DECIMAL)

  0  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0  0
  0  0  0  0  0  0  0  0  0  0
  
```

5.1.4.6.1 - = Retroceder 5 Páginas

Esta opción de menú salta los anteriores 500 registros de datos para visualizar y mostrar los datos.

5.1.4.6.2 P = Página Anterior

Esta opción de menú selecciona y muestra los 100 registros de datos previos.

5.1.4.6.3 + = Avanzar 5 Páginas

Esta opción de menú salta los siguientes 500 registros de datos para visualizar y visualiza la nueva pagina de datos.

5.1.4.6.4 N = Próxima Página

Esta opción de menú selecciona los siguientes 100 registros de datos para mostrar y visualizar los datos.

5.1.4.6.5 D = Display Decimal

Esta opción de menú muestra los datos de la pagina actual en formato decimal.

5.1.4.6.6 H = Display Hexadecimal

Esta opción de menú muestra los datos de la pagina actual en formato hexadecimal.

5.1.4.6.7 F = Display en Coma Flotante

Esta opción de menú muestra los datos de la página actual en formato de Coma Flotante. El programa asume que los valores están alineados en campos de registro exactos. Si los valores en coma flotante no están alineados de tal forma, no aparecerán apropiadamente.

5.1.4.6.8 A = Display ASCII

Esta opción de menú muestra los datos de la página actual en formato ASCII. Esto es útil para regiones de la base de datos que contienen caracteres ASCII.

5.1.4.6.9 M = Main Menú

Esta opción de menú retornará al nodo de menú principal.

5.1.4.7 E and F= Comandos Maestros de Error (Puertos 1 y 2)

Seleccionando estas opciones de menú se coloca el programa en el modo de menú de errores de comando Maestro para el puerto específico. Este modo de operación muestra múltiples páginas de datos de error/status de la lista de comandos Maestros. Para visualizar las opciones de menú disponible en este modo, presione la tecla '?' y el siguiente menú aparecerá.

```
COMMAND ERROR LIST MENU (MASTER Port 1)
?=Display Menu
S=Show Again
-=Back 2 Pages
P=Previous Page
+=Skip 2 Pages
N=Next Page
D=Decimal Display
H=Hexadecimal Display
M=Main Menu
```

Cada opción de menú es discutida en las siguientes secciones:

5.1.4.7.1 S = Mostrar de Nuevo

Esta opción muestra la página actual de datos de error/status de comando Maestro. Luego de seleccionada la opción, aparecerá la siguiente pantalla.

```
COMMAND ERROR LIST FOR PORT 1, COMMANDS 0 TO 19 (DECIMAL)
  0      0      0      0      0      0      0      0      0      0
  0      0      0      0      0      0      0      0      0      0
```

Cada valor mostrado en la pantalla corresponde al código de error/status para el índice de lista de comando Maestro asociado. Refiérase a la sección de **Inicialización del Módulo** para un listado completo y la interpretación de los códigos mostrados.

5.1.4.7.2 - = Retroceder 2 Páginas

Esta opción retrocede 20 comandos y muestra los datos.

5.1.4.7.3 P = Página Anterior

Esta opción muestra la página previa de datos.

5.1.4.7.4 + = Avanzar 2 Páginas

Esta opción avanza 20 comandos y muestra los datos.

5.1.4.7.5 N = Próxima Página

Esta opción muestra la siguiente página de datos de errores/status de la lista maestra de comandos.

5.1.4.7.6 D = Display Decimal

Esta opción cambia la presentación de los datos al formato decimal.

5.1.4.7.7 H = Display Hexadecimal

Esta opción cambia la presentación de los datos al formato hexadecimal.

5.1.4.7.8 M = Menú Principal

Esta opción retorna al programa al modo de menú principal.

5.1.4.8 I y J=Lista maestra de comandos (Puertos 1 y 2)

Seleccionando estas opciones de menú se coloca el programa en el modo de menú de lista maestra de comandos para el puerto específico. Este modo de operación muestra múltiples páginas de datos de la lista maestra de comandos. Para visualizar las opciones de menú disponible en este modo, presione la tecla '?' y el siguiente menú aparecerá.

```

MASTER COMMAND LIST MENU (Port 1)
?=Display Menu
S=Show Again
-=Back 5 Pages
P=Previous Page
+=Skip 5 Pages
N=Next Page
M=Main Menu
    
```

Cada opción de menú es discutida en las siguientes secciones:

5.1.4.8.1 S = Mostrar de Nuevo

Esta opción muestra la página actual de comandos maestros. Diez comandos son mostrados en cada página como se muestra en el siguiente ejemplo:

COMMAND LIST FOR PORT 1-- COMMANDS 0 TO 9									
EN	MBREG	POLLINT	COUNT	SWAP	NODE	FUNC	ADDRS	LASTERR	
1	400	0	10	0	1	3	0	0X0000	
0	0	0	0	0	0	0	0	0X0000	
0	0	0	0	0	0	0	0	0X0000	
0	0	0	0	0	0	0	0	0X0000	
0	0	0	0	0	0	0	0	0X0000	
0	0	0	0	0	0	0	0	0X0000	
0	0	0	0	0	0	0	0	0X0000	
0	0	0	0	0	0	0	0	0X0000	
0	0	0	0	0	0	0	0	0X0000	
0	0	0	0	0	0	0	0	0X0000	

5.1.4.8.2 - = Retroceder 5 Páginas

Esta opción de menú muestra los datos de lista maestra de comandos después de haber saltado los 50 comandos previos.

5.1.4.8.3 P = Página Anterior

Esta opción de menú muestra la página previa de los datos de la lista maestra de comandos.

5.1.4.11 W=Módulo de Arranque en Caliente

Esta opción es utilizada cuando una operación de arranque en caliente es requerida en el módulo. Esta solicitud se hace usualmente después de haber realizado cambios de configuración en el área de datos de Tags del Controlador del procesador ControlLogix. Luego de seleccionar la opción, aparecerá la siguiente pantalla.

```
Press 'Y' key to confirm warm boot!

Warm booting module....

Reloading Program Values....

Read Configuration...
Read Port 1 Command Configuration...
MODBUS MASTER/SLAVE COMMUNICATION MODULE (MVI56-MCM)
(c) 1999, ProSoft Technology, Inc.

      PRODUCT NAME CODE       : MCM
      SOFTWARE REVISION LEVEL  : 0.00
      OPERATING SYSTEM REVISION : 0100
      RUN NUMBER               : 2602

Press ? for menu help.
```

5.1.4.12 Y= Configuración de Transferencia de Módulo al Procesador

Esta opción transfiere los datos de configuración actuales en el módulo procesador ControlLogix. La lógica de escalera debe existir en el procesador para la implementación exitosa de esta opción. Lo siguiente es mostrado indicando la operación exitosa:

```
Press 'Y' key to confirm transfer option!

Sending configuration to processor....

Send complete....Error code = 0.

Warm booting module....

Reloading Program Values....

Read Configuration...
Read Port 1 Command Configuration...
MODBUS MASTER/SLAVE COMMUNICATION MODULE (MVI56-MCM)
(c) 1999, ProSoft Technology, Inc.

      PRODUCT NAME CODE       : MCM
      SOFTWARE REVISION LEVEL  : 0.00
      OPERATING SYSTEM REVISION : 0100
      RUN NUMBER               : 2602

Press ? for menu help.
```

Si la operación no es exitosa, un código de error es retornado. Los errores de código retornados son los siguientes:

Código	Descripción
0	Transferencia Exitosa
-1	Error transfiriendo datos de configuración del módulo (bloque – 9000)
-2	Error transfiriendo datos de la lista maestra de comandos por el puerto 1 (bloques –6000 al –6003).

-3	Error transfiriendo datos de la lista maestra de comandos por el puerto 2 (bloques -6100 al -6103).
----	---

Luego de transferir exitosamente los datos, el módulo ejecuta una operación de arranque en caliente para leer los nuevos datos.

5.1.4.13 1 and 2=Status de Comunicación (Puertos 1 y 2)

Estas opciones muestran el estado de comunicación y las estadísticas del puerto Modbus específico. Esta información puede ser informativa cuando se hace troubleshooting a problemas de la red. Después de seleccionar la opción, la siguiente información es mostrada.

```

PORT 1 MODBUS STATUS:
Enabled : Y
Retries : 0      Cur Cmd : 0      State : 100
ComState: 0

Number of Command Requests: 5431
Number of Cmd Responses : 5430
Number of Command Errors : 0
Number of Requests : 5430
Number of Responses : 5430
Number of Errors Received : 0
Number of Errors Sent : 0

```

5.1.4.14 6 y 7=Configuración de Puertos (Puertos 1 y 2)

Estas opciones son usadas para mostrar la información de configuración para el puerto Modbus seleccionado. Después de seleccionar la opción, la siguiente información sera mostrada:

```

PORT 1 CONFIGURATION:
Enabled : Y      Port Type : (0) - MASTER
SLAVE SETUP:
Modbus Slave ID: 0      Pass-Through = DISABLED
Offsets:
  BitIn: 0      WordIn: 0      Output: 0      Holding: 0
Floating-point Data:
  Flag: N      Start: 0      Offset: 0
Route Count : 0
Function 99 Offset : 0
Use Packet Gap : N      Packet Gap Delay : 0
MASTER SETUP:
Command Count : 3      Cmd Delay: 0      Cmd Offset : 350
Response Timeout: 1000      Retries : 0      Delay Count: 0
COMMUNICATION PARAMETERS:
Protocol: 1 (Modbus ASCII)
Baud: 38400      Parity: NONE      Databits: 8      Stopbits: 1
RTS On: 0      RTS Off: 0      Use CTS Line: N

```

5.1.4.15 Esc=Salir del Programa

Esta opción termina el programa y muestra el prompt del sistema operativo. Esta opción debería solamente seleccionarse si es instruida por el grupo de soporte técnico de ProSoft. Si usted selecciona ésta opción, el módulo cesa su operación de operar. Los

datos no son más transferidos entre los puertos Modbus y el módulo y entre el procesador ControlLogix y el módulo. Esto pudiese causar un trastorno al proceso que actualmente se este ejecutando.

5.2 LED's Indicadores de Status

Los LED's indicaran el estado de operación del módulo como sigue:

Módulo ProSoft	Color	Status	Indicación
P1	Verde Verde	On	Los datos están siendo transferidos entre el módulo y un terminal remoto utilizando el puerto de configuración/depuración
		Off	Ningun dato está siendo transferido en el puerto de configuración/depuración
P2	Verde Verde	On	Datos están siendo transferidos entre el módulo y la red Modbus por su Puerto Modbus #2
		Off	Ningun dato está siendo transferido en el puerto
P3	Verde Verde	On	Datos están siendo transferidos entre el módulo y la red Modbus por su Puerto Modbus #3
		Off	Ningun dato está siendo transferido en el puerto
APP	Ambar Ambar	On	El MVI56-MCM está trabajando normalmente
		Off	El programa del módulo MVI56-MCM ha reconocido un error de comunicación en uno de sus puerto Modbus
BP ACT	Ambar	On	El LED está encendido cuando el módulo está ejecutando una operación de escritura en el backplane
		Off	El LED está apagado cuando el módulo está ejecutando una operación de lectura en el backplane. Bajo operación normal. El LED debería titilar rapidamente.
OK	Rojo/ Verde	Off	La tarjeta no está recibiendo ninguna energía y no ha sido insertada de forma segura en el rack
		Green	El módulo opera normalmente
		Red	El programa ha detectado un error o está siendo configurado. Si el LED permanece rojo por más de 10 segundos, el programa probablemente se ha detenido. Retire la tarjeta del rack y reinsertela para reiniciar el programa del módulo.
BAT	Rojo	Off	El voltaje de la batería esta ok y está funcionando
		On	El voltaje de la batería está bajo o la batería no está presente. Reemplace la batería en el módulo.

Durante la configuración del módulo, el LED OK debe estar en rojo y los LED's APP y BP ACT deben estar encendidos. Si los LED's están mantenidos (latcheados) en este modo por un largo periodo de tiempo, chequee las palabras de error de configuración en el bloque de solicitud de configuración. La estructura del bloque es mostrada en la siguiente tabla:

Offset	Descripción	Longitud
0	Reservado	1
1	9000	1
2	Errores de Configuración del Módulo	1
3	Errores de Configuración del Puerto 1	1
4	Errores de Configuración del Puerto 2	1
5 – 248	Reserva	244
249	-2 o -3	1

Los bits en cada palabra de configuración se muestran en la siguiente tabla. La palabra de error de configuración del módulo tiene la siguiente definición:

Bit	Descripción	Valor
0	El valor de inicio del bloque de lectura es mayor que el tamaño de la base de datos.	0x0001
1	El valor de inicio del bloque de lectura es menor de cero	0x0002
2	El valor de cuenta del bloque de lectura es menor de cero	0x0004
3	La cuenta del bloque de lectura + Start es mayor que el tamaño de la base de datos	0x0008
4	El valor de inicio del bloque de escritura es mayor que el tamaño de la base de datos.	0x0010
5	El valor de inicio del bloque de escritura es menor de cero	0x0020
6	El valor de cuenta del bloque de escritura es menor de cero	0x0040
7	La cuenta del bloque de escritura + Start es mayor que el tamaño de la base de datos	0x0080
8		0x0100
9		0x0200
10		0x0400
11		0x0800
12		0x1000
13		0x2000
14		0x4000
15		0x8000

Las palabras de error de configuración del puerto tienen las siguientes definiciones:

Bit	Descripción	Value
0	Tipo de Código no válido. Entre un valor de 0 (Maestro) y 1 (Esclavo)	0x0001
1	El parámetro de bandera flotante no es válido	0x0002
2	El parámetro de inicio flotante no es válido	0x0004
3	El parámetro de offset flotante no es válido	0x0008
4	Parámetro de Protocolo no válido	0x0010
5	Parámetro de Rata de Baudios no válido	0x0020
6	Parámetro de paridad no válido	0x0040
7	Parámetro de Bit de Datos no válido	0x0080
8	Parámetro de Bits de Parada no válido	0x0100
9	ID Esclavo no válido	0x0200
10	Palabra o Bit de Entrada, Palabra de Salida y/o Registros Sostenidos de Offset no válidos	0x0400
11	Parámetro de Cuentas de Comandos no válidos	0x0800
12	Reserva	0x1000
13	Reserva	0x2000
14	Reserva	0x4000
15	Reserva	0x8000

Corrija cualquier dato inválido en la configuración para la apropiada operación del módulo. Cuando la configuración contiene un set de parámetros válidos, todos los bits en las palabras de configuración serán borrados. Esto no indica que la configuración es

válida para la aplicación del usuario. Asegurese de que cada parámetro esta ajustado correctamente para la aplicación específica.

Si los LED's APP, BP ACT y OK titilan a una rata de 1 segundo, llame al personal de soporte de ProSoft Technology, Inc. Hay un problema serio en el módulo y el mismo tendrá que ser enviado de regreso a ProSoft.

5.2.1 Borrando una Condición de Falla

Típicamente, si el LED de OK en la parte frontal del módulo se ilumina de rojo por más de 10 segundos, un problema de hardware está siendo detectado en el módulo o el programa se ha terminado. Para intentar limpiar ésta condición proceda de la siguiente manera:

1. Remueva la tarjeta del rack y reinsertela de nuevo en el mismo.
2. Verifique que la configuración de los datos esta siendo transferida hacia el módulo desde el procesador ControlLogix.

Si el LED OK del módulo no se torna verde, asegurese de que el módulo está insertado completamente en el rack y se ha descargado un programa de escalera válido dentro del procesador (el procesador debe estar en modo Run). Si ésto no resuelve el problema, contacte al fabricante.

5.2.2 Troubleshooting

La siguiente tabla está diseñada para asistirlo en el troubleshooting del módulo. Por favor, use ésta tabla para intentar corregir el problema. De cualquier forma, si usted tiene preguntas adicionales ó algún problema, por favor no dude en contactarnos.

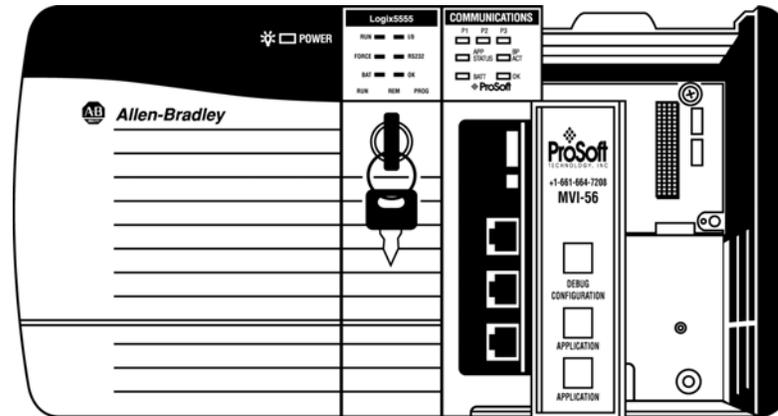
Las entradas en esta sección han sido colocadas de manera de resolver los problemas que ocurren con mayor frecuencia después de encender el módulo.

Descripción del Problema	Pasos a Seguir
Falla del Procesador	<ol style="list-style-type: none"> 1) Asegurese de que el módulo está insertado en el slot que está siendo configurado para el módulo MVI56-MCM 2) Asegurese de que el slot en la configuración del rack ha sido inicializado correctamente.
El LED E/S del procesador se ilumina (titila)	Esto indica que hay un problema con la comunicación con el backplane. Asegurese de esto y de que todos los módulos en el rack están configurados en el procesador.
El LED BP ACT permanece apagado o parpadea lentamente	<p>Esto indica que las operaciones de transferencia del backplane están fallando. Utilice la facilidad del puerto de Configuración/Depuración para chequear esto. Para reestablecer la comunicación con el backplane asegurese de lo siguiente:</p> <ol style="list-style-type: none"> 1) El driver del backplane está cargado en el módulo. 2) El módulo está configurado para leer y escribir un bloque de transferencia de datos. 3) La lógica de escalera maneja todas las situaciones de lectura y escritura de bloques 4) El módulo está configurado en el procesador.
El LED OK permanece rojo	El programa se ha detenido o ha ocurrido un error crítico. Conéctese al puerto de Configuración/Depuración para ver si el módulo está en ejecución. Si el programa está detenido, retire la tarjeta del rack y reinsértela de nuevo.

6 Conexiones de Cableado

El módulo MVI56-MCM tiene las siguientes conexiones de comunicación en el módulo:

- Dos (2) Puertos de Comunicación Modbus (Conector RJ-45)
- Un (1) Puerto RS-232 de Configuración/Depuración (Conector RJ-45)



6.1 Puertos de Comunicación Modbus

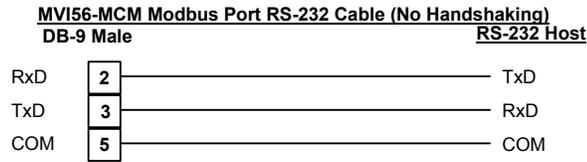
El módulo MVI56-MCM posee físicamente dos conectores Modbus con una conexión RJ-45 localizada en la parte frontal del módulo.

6.1.1 Conectando el Cableado al Conector

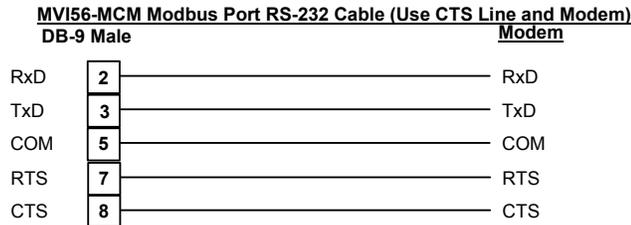
ProSoft provee dos (2) convertidores RJ-45 a DB-9 macho para hacer más simple la interfaz con otros dispositivos. Los puertos Modbus del módulo se pueden configurar para operar en el modo RS-232, RS-422 y RS-485. Esta interfaz está asociada con un puerto que está ajustado con jumpers en el módulo. Existe un jumper para cada uno de los puertos. Adicionalmente el uso de líneas de control de módem es definido por el usuario por la configuración de las variables MCM.PORT1.UseCTS y MCM.PORT2.UseCTS al valor de uno (1). Las siguientes secciones describen cada una de las interfazs.

6.1.1.1 RS-232

Cuando se selecciona la interfaz RS-232, el uso de líneas de control de módem es definible por el usuario. Si no se usan líneas de control de módem, el cableado que se debe conectar al puerto es mostrado en el siguiente ejemplo:

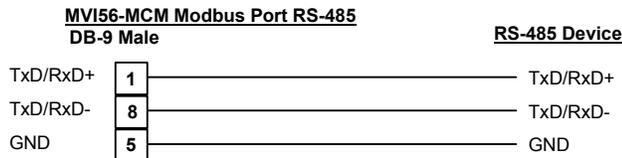


La línea RTS está controlada por los parámetros RTS on y RTS off ajustados en el puerto. Si la línea CTS es utilizada (usualmente sólo se requieren modems half duplex), las líneas RTS y CTS se deben unir entre sí ó se conectan al módem. El siguiente diagrama muestra el cableado que se requiere cuando se conecta el puerto al módem.



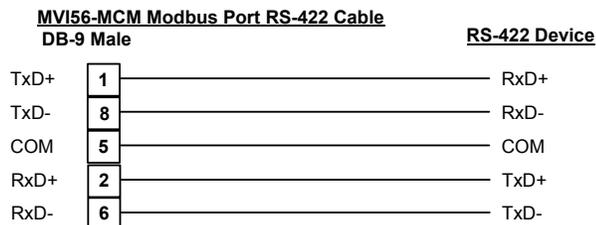
6.1.1.2 RS-485

Cuando se selecciona la interfaz RS-485, sólo se requieren dos o tres cables. El uso de una tierra es opcional y depende de la red RS-485. El cableado requerido para ésta interfaz se muestra en el siguiente diagrama:



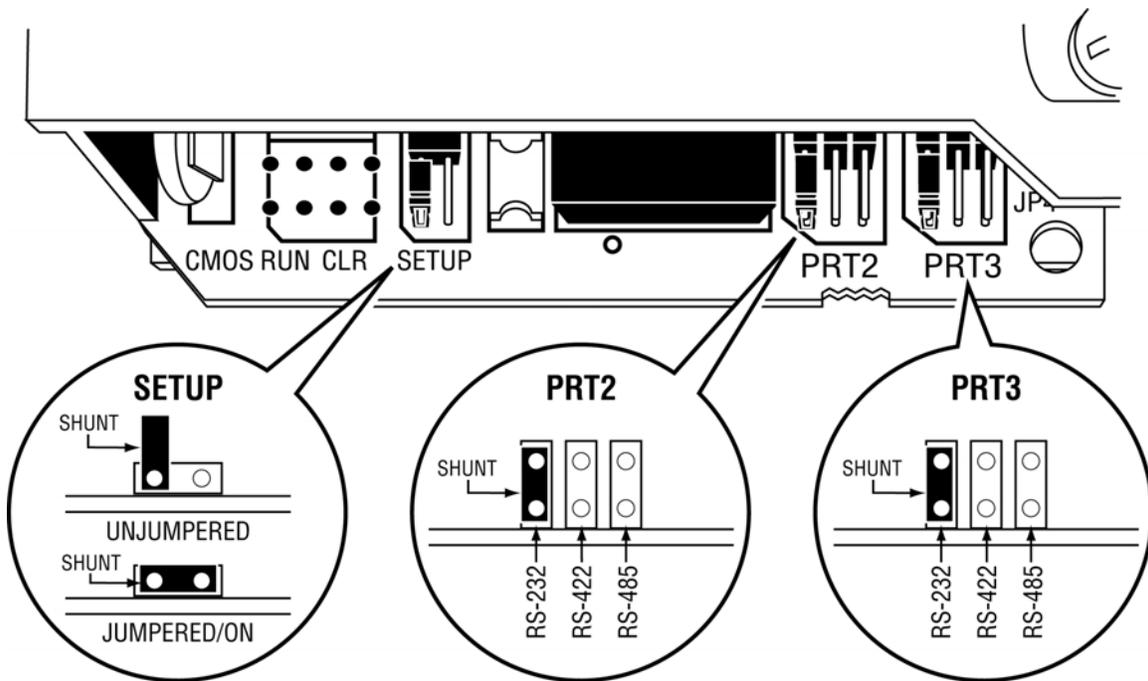
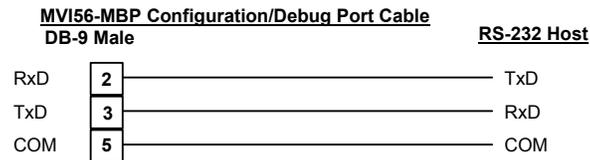
6.1.1.3 RS-422

Cuando se selecciona la interfaz RS-422, sólo se requieren cuatro o cinco cables. El uso de una tierra es opcional y depende de la red RS-422. El cableado requerido para ésta interfaz se muestra en el siguiente diagrama:



6.2 Puerto de Configuración/Depuración RS-232

Este puerto es físicamente una conexión RJ-45. Un convertidor RJ-45 a DB-9 es entregado con el módulo. Este puerto permite a un computador con un programa basado en emulación de terminal para ver la configuración y los datos de estado en el módulo y controlar el mismo. El cableado para la comunicación de éste puerto se muestra en el siguiente diagrama:



Apéndice A – Definición de la Base de Datos MVI56-MCM

Este apéndice contiene un listado de la base de datos interna del módulo MVI56-MCM. Esta información puede ser utilizada para realizar interfaz entre otros dispositivos con los datos contenidos en el módulo.

Rango de Registro	Modbus Bajo	Modbus Alto	Contenido	Tamaño
0-4999	40001	45000	Data de Usuario	5000
5000-5009	45001	45010	Configuración del Backplane	10
5010-5039	45011	45040	Inicializacion Puerto 1	30
5040-5069	45041	45070	Inicializacion Puerto 1	30
5070-5869	45071	46070	Comandos del Puerto 1	800
5870-6669	46071	47070	Comandos del Puerto 1	800
6750-6752	46751	46753	Inicializacion Puerto Auxiliar 1	3
6760-6762	46761	46763	Inicializacion Puerto Auxiliar 1	3
7600-7632	47601	47633	Datos de Estado Misceláneo	33
7200-7232	47801	7999	Control de Comandos	200

El área de datos de usuario es usada para contener los datos recolectados de otros nodos en la red (comandos de lectura Maestros) o datos recibidos del procesador (bloques de escritura).

Adicionalmente ésta área de datos es utilizada como fuente de datos para el procesador (bloques de lectura) u otros nodos de la red (comandos de escritura).

La definición detallada del área de datos de estados misceláneos se puede encontrar en el apéndice B.

La definición de las áreas de los datos de configuración puede ser encontrada en la sección de definición de datos de éste documento en el apéndice C.

El apéndice D contiene una discusión de la sección de control de comandos de la base de datos.

Appendix B – Definición de Datos de Status del MVI56-MCM

Est apéndice contiene una descripción de los miembros presentes en el objeto **MCMInStat**. Estos datos son transferidos desde el módulo hacia el procesador como parte de cada bloque leído.

Offset	Contenido	Descripción
202	Cuenta de Scan del Programa	Este valor se incrementa cada vez que ocurre un ciclo completo del programa en el módulo.
203 –204	Código del Producto	Estos dos registros contienen el Código del Producto de “MCM”
205 – 206	Versión del Producto	Estos dos registros contienen la Versión del Producto para el software que se ejecuta actualmente
207 –208	Sistema Operativo	Estos dos registros contienen los valores de mes y año del sistema operativo del programa
209 – 210	Número de Ejecucion	Estos dos registros contienen el valor del número de ejecución para el software que se ejecuta actualmente
211	Lista de Comandos del Puerto 1 solicitada	Este campo contiene el número de solicitudes hechas desde este puerto a los dispositivos Esclavos en la red.
212	Lista de Comandos de Respuesta del Puerto 1	Este campo contiene el número de mensajes de respuesta recibidas de los Esclavos en la red
213	Errores de Lista de Comandos del Puerto 1	Este campo contiene el número de errores de comando procesados en el puerto. Estos errores pudiesen deberse a un comando o respuesta malo.
214	Solicitudes del Puerto 1	Este campo contiene el número total de mensajes enviados desde el puerto
215	Respuestas del Puerto 1	Este campo contiene el número total de mensajes recibidos desde el puerto
216	Errores Enviados Puerto 1	Este campo contiene el número total de errores de mensajes enviados desde el puerto
217	Errores Recibidos Puerto 1	Este campo contiene el número total de errores recibidos desde el puerto
218	Lista de Comandos del Puerto 2 solicitada	Este campo contiene el número de solicitudes hechas desde este puerto a los dispositivos Esclavos en la red.
219	Lista de Comandos de Respuesta del Puerto 2	Este campo contiene el número de mensajes de respuesta recibidas de los Esclavos en la red
220	Errores de Lista de Comandos del Puerto 2	Este campo contiene el número de errores de comando procesados en el puerto. Estos errores pudiesen deberse a un comando o respuesta errado.
221	Solicitudes del Puerto 2	Este campo contiene el número total de mensajes enviados desde el puerto
222	Respuestas del Puerto 2	Este campo contiene el número total de mensajes recibidos desde el puerto
223	Errores Enviados	Este campo contiene el número total de errores de

Appendix B - MVI56-MCM Status Data Definition

	Puerto 2	mensajes enviados desde el puerto
224	Errores Recibidos Puerto 2	Este campo contiene el número total de errores recibidos desde el puerto
225	Cuentas de bloques de Lectura	Este campo contiene el número total de bloques de lectura transferidos desde el módulo hasta el procesador
226	Cuentas de bloques de Escritura	Este campo contiene el número total de bloques de escritura transferidos desde el módulo hasta el procesador
227	Cuenta de Bloques Analizados	Este campo contiene el número total de bloques analizados exitosamente que fueron recibidos desde el procesador
228	Cuentas de Bloques de Eventos de Comando	Este campo contiene el número total de bloques de evento de comandos que fueron recibidos desde el procesador
229	Cuentas de Bloques de Comandos	Este campo contiene el número total de bloques de comandos que fueron recibidos desde el procesador
230	Cuentas de Errores de Bloques	Este campo contiene el número total de errores de bloque reconocidos por el módulo.
231	Errores Actuales del Puerto 1	Para un puerto Esclavo, éste campo contiene el valor de código de errores actual retornado. Para un Maestro, este campo contiene el índice de comando ejecutado actual
232	Ultimos Errores del Puerto 1	Para un puerto Esclavo, éste campo contiene el valor de código de los ultimos errores retornado. Para un Maestro, este campo contiene el índice de ultimos comandos ejecutado
233	Errores Actuales del Puerto 2	Para un puerto Esclavo, éste campo contiene el valor de código de errores actual retornado. Para un Maestro, este campo contiene el índice de comando ejecutado actual
234	Ultimos Errores del Puerto2	Para un puerto Esclavo, éste campo contiene el valor de código de los ultimos errores retornado. Para un Maestro, este campo contiene el índice de ultimos comandos ejecutado

Appendix C – Definición de Datos de Configuración MVI56-MCM

Este apéndice contiene los listados de la base de datos del módulo MVI56-MCM relativos a la configuración del módulo. Estos datos están disponibles para cualquier nodo en la red y son leídos desde el procesador ControlLogix cuando el módulo se inicializa por primera vez.

Inicialización del Backplane

Registro	Contenido	Descripción
5000	Registro de Inicio de Escritura	Este parámetro especifica el registro de inicio en el módulo donde los datos transferidos desde el procesador serán colocados. El rango válido para este parámetro es de 0 a 4999
5001	Cuenta de Registro de Escritura	Este parámetro especifica el número de registros transferidos desde el procesador al módulo. Los valores de entrada para éste parámetro van desde 0 a 5000
5002	Registro de Inicio de Lectura	Este parámetro especifica el registro de inicio en el módulo donde los datos serán transferidos desde el módulo al procesador. El rango válido para éste parámetro es de 0 a 4999
5003	Cuenta de Registro de Lectura	Este parámetro especifica el número de registros a ser transferidos desde el módulo al procesador. Los valores de entrada para éste parámetro van desde 0 a 5000
5004	Falla del Backplane	Este parámetro especifica el número de errores transferidos sucesivamente que pueden ocurrir antes de que los puertos de comunicación sean detenidos. Si el parámetro es ajustado a cero, los puertos de comunicación continuarán operando bajo todas las condiciones. Si el valor es mayor de 0 (1-65535), la comunicación cesará si el número especificado de fallas ocurre.
5005	Puntero de Estado de Error	Este parámetro especifica la localización de registro en la base de datos del módulo donde los datos de estado del módulo serán almacenados. Si un valor menor de cero es introducido, los datos no se almacenarán en la base de datos. Si el valor es especificado en el rango de 0 a 4940, los datos serán colocados en el área de datos del usuario.
5006	Reserva	
5007	Reserva	
5008	Reserva	
5009	Reserva	

Inicialización del Puerto 1

Registro	Contenido	Descripción
5010	Enable	Este parámetro es usado para definir si éste puerto Modbus será utilizado. Si el parámetro es ajustado a cero, el puerto está deshabilitado. Un valor de 1 habilitará el puerto.
5011	Type	Este parámetro especifica si el puerto emulará un dispositivo Maestro Modbus (0), un dispositivo Esclavo Modbus sin modo Pass-Trough (1), un dispositivo Esclavo Modbus en modo Pass-Trough no formateado (2), un dispositivo Esclavo Modbus en modo Pass-Trough formateado y alternación de datos (3).
5012	Float Flag	Esta bandera especifica si el acceso funcional de datos en Coma flotante está siendo implementado. Si la bandera flotante está ajustada a 1, las funciones Modbus 3, 6 y 16 serán interpretadas como valores en coma flotante para registros según se especifique en los dos parámetros siguientes.
5013	Float Start	Este parámetro define el primer registro de datos en coma flotante. Todas las solicitudes con valores de registro mayores o iguales a este valor serán consideradas como solicitudes en coma flotante. Este parámetro solo es utilizado si la bandera flotante (Float Flag) esta habilitada.
5014	Float Offset	Este parámetro define el registro de inicio para datos en coma flotante en la base de datos interna. Este parámetro solo es utilizado si la bandera flotante (Float Flag) está habilitada.
5015	Protocol	Este parámetro especifica el protocolo Modbus a ser usado en el puerto. Los protocolos válidos son: 0=Modbus RTU y 1=Modbus ASCII
5016	Baud Rate	Esta es la rata de baudios a ser usada en el puerto. Introduzca la rata de baudios como un valor. Por ejemplo, para seleccionar 19K baudios, introduzca 19200. Entradas válidas son: 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 28800, 38400, 57600 y 115200
5017	Parity	Este es el código de paridad a ser usado por el puerto. Los valores son: None (ninguna paridad), Odd (paridad impar) y Even (paridad par)
5018	Data Bits	Este parámetro ajusta el número de bits de datos para cada palabra usada por el protocolo. Las entradas válidas para este campo son 5, 6, 7 u 8
5019	Stop Bits	Este parámetro ajusta el número de bits de parada a ser usado con cada valor de datos enviado. Los valores de entrada son 1 y 2.
5020	RTS On	Este parámetro ajusta el número de milisegundos de retardo después que RTS es acertado antes de que los datos sean transmitidos. Valores válidos están en el rango

Registro	Contenido	Descripción
		de 0 a 65535 milisegundos.
5021	RTS Off	Este parámetro ajusta el número de milisegundos de retardo después que el ultimo byte de datos es enviado antes de la señal RTS del modem que sera ajustada a su valor bajo Valores válidos están en el rango de 0 a 65535 milisegundos.
5022	Tiempo de Respuesta Minimo	Este parámetro ajusta el número de milisegundos de retardo antes de responder a una solicitud de mensaje. Este retardo pre-enviado es aplicado antes de RTS a tiempo. Esto puede requerirse cuando se tenga comunicación con dispositivos lentos.
5023	Uso de Linea CTS	Este parámetro especifica si la línea de control del módem CTS es usada. Si el parámetro es ajustado a 0, la línea CTS no será monitoreada. Si el parámetro es ajustado a 1, La línea CTS será monitoreada y deberá ser alta antes de que el modem envíe datos. Este parámetro es requerido normalmente cuando se utilizan módems half-duplex para la comunicación (2 hilos).
5024	ID Esclavo	Este parámetro define la dirección del Esclavo Modbus virtual para la base de datos interna. Todas las solicitudes recibidas por el puerto con ésta dirección son procesadas por el módulo. Asegurese de que cada dispositivo tiene una dirección unica en la red. El rango válido para este parámetro es de 1 a 255 (247 en algunas redes).
5025	Bit en Offset	Este parámetro especifica la dirección offset en la base de datos Modbus interna que esta siendo usada con solicitudes de red para los comandos Modbus de función 2. Por ejemplo, si el valor es ajustado a 150, un requerimiento de dirección de 0 retornara el valor al registro 150 en la base de datos.
5026	Palabra en Offset	Este parámetro especifica la dirección offset en la base de datos Modbus interna que esta siendo usada con solicitudes de red para los comandos Modbus de función 4. Por ejemplo, si el valor es ajustado a 150, un requerimiento de dirección de 0 retornará el valor al registro 150 en la base de datos.
5027	Salida en Offset	Este parámetro especifica la dirección offset en la base de datos Modbus interna que esta siendo usada con solicitudes de red para los comandos Modbus de función 1, 5 y 15. Por ejemplo, si el valor es ajustado a 150, un requerimiento de dirección de 0 retornara el valor al registro 150 en la base de datos.
5028	Offset de Registro Mantenido	Este parámetro especifica la dirección offset en la base de datos Modbus interna que está siendo usada con solicitudes de red para los comandos

Registro	Contenido	Descripción
		Modbus de función 3, 6 o 16. Por ejemplo, si el valor es ajustado a 150, un requerimiento de dirección de 0 retornara el valor al registro 150 en la base de datos.
5029	Contador de Comandos	Este parámetro especifica el número de comandos a ser procesados por el puerto Modbus Maestro.
5030	Retardo Minimo de Comando	Este parámetro especifica el número de milisegundos a esperar entre entregas de cada comando. Este valor de retardo no aplica para reintentos.
5031	Puntero de Error de Comando	Este parámetro ajusta la dirección en la base de datos Modbus interna donde el error de comando sera colocado. Si el valor es ajustado a -1, los datos no serán transferidos a la base de datos. El rango válido de valores para este parámetro es de -1 a 4999.
5032	Time Out de Respuesta	Este parámetro representa el periodo de timeout de respuesta de mensaje en incrementos de 1 milisegundo. Este es el tiempo que un puerto configurado como Maestro espera antes de retransmitir un comando si ninguna respuesta es recibida del Esclavo direccionado. El valor es ajustado dependiendo de la red de comunicación usada y del tiempo de respuesta esperado del dispositivo mas lento de la red.
5033	Contador de Reintento	Este parámetro especifica el número de veces que un comando será reintentado si falla. Si el puerto Maestro no recibe una respuesta después del ultimo reintento, la comunicación de los dispositivos Esclavos se suspenderá en el puerto por escaneos de contador de retardo con error.
5034	Contador de Retardo de Error	Este parámetro especifica el número de interrogaciones para saltar sobre el Esclavo antes de tratar de reestablecer la comunicación. Después que el Esclavo falla para responder, el Maestro saltara los comandos a ser enviados al Esclavo el número de veces entrado en este parámetro.
5035	Reserva	
5036	Reserva	
5037	Reserva	
5038	Reserva	
5039	Reserva	

Inicialización del Puerto 2

Registro	Contenido	Descripción
5040	Enable	Este parámetro es usado para definir si éste puerto Modbus será utilizado. Si el parámetro es ajustado a cero, el puerto está deshabilitado. Un valor de 1 habilitará el puerto.

Registro	Contenido	Descripción
5041	Type	Este parámetro especifica si el puerto emulará un dispositivo Maestro Modbus (0), un dispositivo Esclavo Modbus sin modo Pass-Trough (1), un dispositivo Esclavo Modbus en modo Pass-Trough no formateado (2), un dispositivo Esclavo Modbus en modo Pass-Trough formateado y alternación de datos (3).
5042	Float Flag	Esta bandera especifica si el acceso funcional de datos en Coma flotante está siendo implementado. Si la bandera flotante está ajustada a 1, las funciones Modbus 3, 6 y 16 serán interpretadas como valores en coma flotante para registros según se especifique en los dos parámetros siguientes.
5043	Float Start	Este parámetro define el primer registro de datos en coma flotante. Todas las solicitudes con valores de registro mayores o iguales a este valor serán consideradas como solicitudes en coma flotante. Este parámetro solo es utilizado si la bandera flotante (Float Flag) esta habilitada.
5044	Float Offset	Este parámetro define el registro de inicio para datos en coma flotante en la base de datos interna. Este parámetro solo es utilizado si la bandera flotante (Float Flag) está habilitada.
5045	Protocol	Este parámetro especifica el protocolo Modbus a ser usado en el puerto. Los protocolos válidos son: 0=Modbus RTU y 1=Modbus ASCII
5046	Baud Rate	Esta es la rata de baudios a ser usada en el puerto. Introduzca la rata de baudios como un valor. Por ejemplo, para seleccionar 19K baudios, introduzca 19200. Entradas válidas son: 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 28800, 38400, 57600 y 115200
5047	Parity	Este es el código de paridad a ser usado por el puerto. Los valores son: None (ninguna paridad), Odd (paridad impar) y Even (paridad par)
5048	Data Bits	Este parámetro ajusta el número de bits de datos para cada palabra usada por el protocolo. Las entradas válidas para este campo son 5, 6, 7 u 8
5049	Stop Bits	Este parámetro ajusta el número de bits de parada a ser usado con cada valor de datos enviado. Los valores de entrada son 1 y 2.
5050	RTS On	Este parámetro ajusta el número de milisegundos de retardo después que RTS es acertado antes de que los datos sean transmitidos. Valores válidos están en el rango de 0 a 65535 milisegundos.
5051	RTS Off	Este parámetro ajusta el número de milisegundos de retardo después que el ultimo byte de datos es enviado antes de la señal RTS del modem que sera ajustada a su valor bajo Valores válidos están en el rango de 0 a 65535 milisegundos.
5052	Tiempo de	Este parámetro ajusta el número de

Appendix C – MVI56-MCM Configuration Data Definition

Registro	Contenido	Descripción
	Respuesta Minimo	milisegundos de retardo antes de responder a una solicitud de mensaje. Este retardo pre-enviado es aplicado antes de RTS a tiempo. Esto puede requerirse cuando se tenga comunicación con dispositivos lentos.
5053	Uso de Linea CTS	Este parámetro especifica si la línea de control del módem CTS es usada. Si el parámetro es ajustado a 0, la línea CTS no será monitoreada. Si el parámetro es ajustado a 1, La línea CTS será monitoreada y deberá ser alta antes de que el modem envíe datos. Este parámetro es requerido normalmente cuando se utilizan módems half-duplex para la comunicación (2 hilos).
5054	ID Esclavo	Este parámetro define la dirección del Esclavo Modbus virtual para la base de datos interna. Todas las solicitudes recibidas por el puerto con ésta dirección son procesadas por el módulo. Asegurese de que cada dispositivo tiene una dirección unica en la red. El rango válido para este parámetro es de 1 a 255 (247 en algunas redes).
5055	Bit en Offset	Este parámetro especifica la dirección offset en la base de datos Modbus interna que esta siendo usada con solicitudes de red para los comandos Modbus de función 2. Por ejemplo, si el valor es ajustado a 150, un requerimiento de dirección de 0 retornara el valor al registro 150 en la base de datos.
5056	Palabra en Offset	Este parámetro especifica la dirección offset en la base de datos Modbus interna que esta siendo usada con solicitudes de red para los comandos Modbus de función 4. Por ejemplo, si el valor es ajustado a 150, un requerimiento de dirección de 0 retornará el valor al registro 150 en la base de datos.
5057	Salida en Offset	Este parámetro especifica la dirección offset en la base de datos Modbus interna que esta siendo usada con solicitudes de red para los comandos Modbus de función 1, 5 y 15. Por ejemplo, si el valor es ajustado a 150, un requerimiento de dirección de 0 retornara el valor al registro 150 en la base de datos.
5058	Offset de Registro Mantenido	Este parámetro especifica la dirección offset en la base de datos Modbus interna que está siendo usada con solicitudes de red para los comandos Modbus de función 3, 6 o 16. Por ejemplo, si el valor es ajustado a 150, un requerimiento de dirección de 0 retornara el valor al registro 150 en la base de datos.
5059	Contador de Comandos	Este parámetro especifica el número de comandos a ser procesados por el puerto Modbus Maestro.
5060	Retardo Minimo	Este parámetro especifica el número de

Registro	Contenido	Descripción
	de Comando	milisegundos a esperar entre entregas de cada comando. Este valor de retardo no aplica para reintentos.
5061	Puntero de Error de Comando	Este parámetro ajusta la dirección en la base de datos Modbus interna donde el error de comando sera colocado. Si el valor es ajustado a -1, los datos no serán transferidos a la base de datos. El rango válido de valores para este parámetro es de -1 a 4999.
5062	Time Out de Respuesta	Este parámetro representa el periodo de timeout de respuesta de mensaje en incrementos de 1 milisegundo. Este es el tiempo que un puerto configurado como Maestro espera antes de retransmitir un comando si ninguna respuesta es recibida del Esclavo direccionado. El valor es ajustado dependiendo de la red de comunicación usada y del tiempo de respuesta esperado del dispositivo mas lento de la red.
5063	Contador de Reintento	Este parámetro especifica el número de veces que un comando será reintentado si falla. Si el puerto Maestro no recibe una respuesta después del ultimo reintento, la comunicación de los dispositivos Esclavos se suspenderá en el puerto por escaneos de contador de retardo con error.
5064	Contador de Retardo de Error	Este parámetro especifica el número de interrogaciones para saltar sobre el Esclavo antes de tratar de reestablecer la comunicación. Después que el Esclavo falla para responder, el Maestro saltara los comandos a ser enviados al Esclavo el número de veces entrado en este parámetro.
5065	Reserva	
5066	Reserva	
5067	Reserva	
5068	Reserva	
5069	Reserva	

Comandos del Puerto 1

Registros	Contenido	Descripción
5070 – 5777	Comando #1	Este juego de registros contiene los parámetros para el primer comando en la lista maestra de comandos. La estructura de ésta área de datos es como se describe en la sección de objeto de datos de la documentación
5078 – 5085	Comando #2	Ajuste de Datos del Comando #2
----	----	----
5852 – 5859	Comando #100	Ajuste de Datos del Comando #100

Comandos del Puerto 2

Registros	Contenido	Descripción
5870 – 5877	Comando #1	Este juego de registros contiene los parámetros para el primer comando en la lista maestra de comandos. La estructura de ésta área de datos es como se describe en la sección de objeto de datos de la documentación
5878 – 5885	Comando #2	Ajuste de Datos del Comando #2
----	----	----
6662 – 6669	Comando #100	Ajuste de Datos del Comando #100

Status Misceláneos

Registro	Contenido	Descripción
6670	Cuenta de Scan del Programa	Este valor se incrementa cada vez que ocurre un ciclo completo del programa en el módulo.
6671 - 6672	Código del Producto	Estos dos registros contienen el Código del Producto de "MCM"
6673 - 6674	Versión del Producto	Estos dos registros contienen la Versión del Producto para el software que se ejecuta actualmente
6675 - 6676	Sistema Operativo	Estos dos registros contienen los valores de mes y año del sistema operativo del programa
6677 - 6678	Número de Ejecución	Estos dos registros contienen el valor del número de ejecución para el software que se ejecuta actualmente
6679	Lista de Comandos del Puerto 1 solicitada	Este campo contiene el número de solicitudes hechas desde éste puerto a los dispositivos Esclavos en la red.
6680	Lista de Comandos de Respuesta del Puerto 1	Este campo contiene el número de mensajes de respuesta recibidas de los Esclavos en la red
6681	Errores de Lista de Comandos del Puerto 1	Este campo contiene el número de errores de comando procesados en el puerto. Estos errores pudiesen deberse a un comando o respuesta malo.
6682	Solicitudes del Puerto 1	Este campo contiene el número total de mensajes enviados desde el puerto
6683	Respuestas del Puerto 1	Este campo contiene el número total de mensajes recibidos desde el puerto
6684	Errores Enviados Puerto 1	Este campo contiene el número total de errores de mensajes enviados desde el puerto
6685	Errores Recibidos Puerto 1	Este campo contiene el número total de errores recibidos desde el puerto
6686	Lista de Comandos del Puerto 2 solicitada	Este campo contiene el número de solicitudes hechas desde este puerto a los dispositivos Esclavos en la red.
6687	Lista de Comandos de Respuesta del Puerto 2	Este campo contiene el número de mensajes de respuesta recibidas de los Esclavos en la red

Registro	Contenido	Descripción
6688	Errores de Lista de Comandos del Puerto 2	Este campo contiene el número de errores de comando procesados en el puerto. Estos errores pudiesen deberse a un comando o respuesta malo.
6689	Solicitudes del Puerto 2	Este campo contiene el número total de mensajes enviados desde el puerto
6690	Respuestas del Puerto 2	Este campo contiene el número total de mensajes recibidos desde el puerto
6691	Errores Enviados Puerto 2	Este campo contiene el número total de errores de mensajes enviados desde el puerto
6692	Errores Recibidos Puerto 2	Este campo contiene el número total de errores recibidos desde el puerto
6693	Cuentas de bloques de Lectura	Este campo contiene el número total de bloques de lectura transferidos desde el módulo hasta el procesador
6694	Cuentas de bloques de Escritura	Este campo contiene el número total de bloques de lectura transferidos desde el módulo hasta el procesador
6695	Cuenta de Bloques Analizados	Este campo contiene el número total de bloques analizados exitosamente que fueron recibidos desde el procesador
6696	Cuentas de Bloques de Eventos de Comando	Este campo contiene el número total de bloques de evento de comandos que fueron recibidos desde el procesador
6697	Cuentas de Bloques de Comandos	Este campo contiene el número total de bloques de comandos que fueron recibidos desde el procesador
6698	Cuentas de Errores de Bloques	Este campo contiene el número total de errores de bloque reconocidos por el módulo.
6699	Errores Actuales del Puerto 1	Para un puerto Esclavo, este campo contiene el valor de código de errores actual retornado. Para un Maestro, este campo contiene el índice de comando ejecutado actual
6700	Ultimos Errores del Puerto 1	Para un puerto Esclavo, este campo contiene el valor de código de los ultimos errores retornado. Para un Maestro, este campo contiene el índice de ultimos comandos ejecutado
6701	Errores Actuales del Puerto 2	Para un puerto Esclavo, este campo contiene el valor de código de errores actual retornado. Para un Maestro, este campo contiene el índice de comando ejecutado actual
6702	Ultimos Errores del Puerto2	Para un puerto Esclavo, este campo contiene el valor de código de los ultimos errores retornado. Para un Maestro, este campo contiene el índice de ultimos comandos ejecutado
6703	Reserva	
-	-----	
6749	Reserva	
6750	Banda de Guarda de Uso de Puerto 1	Utiliza timeout packet gap para mensajes (Si o No). Se usa solo en aplicaciones multi-drop
6751	Tiempo de bande	Un valor de cero se usa como rata de baudios

Appendix C – MVI56-MCM Configuration Data Definition

Registro	Contenido	Descripción
	de Guarda de Puerto 1	por defecto o usted puede ajustar un valor de timeout en milisegundos.
6752	Offset de Función 99 de Puerto 1	Offset de base de datos interna para contador de función 99
6753	Reserva	
-	-----	
6759	Reserva	
6760	Banda de Guarda de Uso de Puerto 1	Utiliza timeout packet gap para mensajes (Si o No). Se usa solo en aplicaciones multi-drop
6761	Tiempo de bande de Guarda de Puerto 1	Un valor de cero se usa como rata de baudios por defecto o usted puede ajustar un valor de timeout en milisegundos.
6762	Offset de Función 99 de Puerto 1	Offset de base de datos interna para contador de función 99
6763	Reserva	
-	-----	
6799	Reserva	

Control de Comando

Registro	Contenido	Descripción
6800	Código de Comando	Introduzca uno de los códigos de comando de control en este registro para controlar el módulo (9997, 9998 o 9999). Refiérase al apéndice D para mas información
6801	Datos de Comando	No usado
-	-----	-----
6999	Datos de Comando	No usado

Apéndice D – Control de Comando MVI56-MCM

Los datos de control de comando son recibidos desde otros nodos en la red que pueden controlar el módulo MVI56-MCM. Valores específicos son escritos a regiones de éste bloque para controlar el módulo. Corrientemente el módulo está programado para manejar el recibo de las siguientes solicitudes: Escribir la configuración al procesador, arranque en caliente y arranque en frío.

El nodo remoto controla el módulo escribiendo uno de los siguientes valores al registro 7800 (Direccion Modbus 47801):

9997	Escribe la configuración en la base de datos al procesador y da un arranque en caliente al módulo.
9998	Arranque en caliente al módulo
9999	Arranque en frío al módulo

El registro de control es borrado (con un valor de 0) después que la operación es ejecutada, con la excepción del comando 9997. Si el módulo falla en transferir exitosamente la configuración hacia el procesador, entonces será retornado un código de error al registro de control tal como sigue:

0	No hay errores. Transferencia exitosa.
-1	Error transfiriendo información de configuración general
-2	Error transfiriendo lista maestra de comandos del puerto Modbus 1
-3	Error transfiriendo lista maestra de comandos del puerto Modbus 2

La lógica de escalera debe estar escrita para manejar el comando 9997. No se requiere lógica de escalera cuando se usan comandos de arranque en caliente ó frío.

Apéndice E – Especificaciones del Producto

Especificaciones Generales

El módulo MVI56-MCM actúa como un gateway entre la red Modbus y el Backplane Allen-Bradley. La transferencia de datos desde el procesador ControlLogix es asíncrona desde las acciones sobre la red Modbus. Un espacio de registro de 5000 palabras en el módulo es usado para intercambiar datos entre el procesador y la red Modbus.

Algunas de las especificaciones generales incluyen:

Soporte para el almacenamiento y transferencia de hasta 5000 registros hacia y desde los archivos de datos del procesador ControlLogix.

Uso de memoria del módulo que es completamente definible por el usuario.

Dos (2) puertos para emular cualquier combinación de dispositivos Modbus Maestros y/o Esclavos.

Parámetros configurables que incluyen:

Protocolo	:	RTU or ASCII
Rata de Baudios	:	110 to 115,200
Paridad	:	None, Odd and Even
Bits de Datos	:	5 to 8
Bits de Parada	:	1 or 2
Temporización RTS On y Off	:	0 to 65535 milliseconds
Retardo Mínimo de Respuesta	:	0 to 65535 milliseconds
Uso de Línea Modem CTS	:	Yes or No
Camino de Ruta de Dispositivo	:	1 to 64
Soporta Coma Flotante	:	

Especificaciones Funcionales del Esclavo Modbus

El módulo MVI56-MCM acepta comandos de código de función Modbus 1, 2, 3, 4, 5, 6, 8, 15, 16, 17, 22 y 23 desde una unidad maestra Modbus conectada. Un puerto configurado como Esclavo Modbus permite un manejo remoto para interactuar con todos los datos contenidos en el módulo. Estos datos pueden ser entregados desde otro dispositivo Esclavo Modbus en la red a través de un puerto Maestro o desde el procesador ControlLogix. Este módulo puede ser configurado para pasar comandos de escritura (funciones 5, 6, 15, 16, 22 y 23) directamente desde el host remoto hacia el procesador. Este modo de operación es referido como un modo Pass-Through.

Especificaciones Funcionales del Maestro Modbus

Un puerto configurado como un dispositivo Modbus Maestro virtual sobre el módulo MVI56-MCM activará entregas de comandos Modbus a otros nodos de red Modbus. Cien comandos pueden ser soportados por cada puerto. Adicionalmente, los puertos Maestros presentan una característica de interrogación optimizada que interroga a los Esclavos con problemas de comunicación menos frecuentes. El procesador ControlLogix puede ser programado para controlar la actividad en el puerto por la activación de comandos de selección desde la lista de comandos para ejecutar o entregar comandos directamente desde la lógica de escalera. El procesador ControlLogix también tiene la habilidad de controlar el scanning de Esclavos en el puerto. La interrogación individual de los Esclavos puede ser controlada selectivamente (habilitación/deshabilitación) a través de la lógica de escalera.

Características Físicas

El módulo MVI56-MCM está diseñado por ProSoft Technology Inc. e incorpora tecnología licenciada de Allen-Bradley (tecnología de “Backplane” ControlLogix). Factor de forma y pendiente sencilla ControlLogix.

Conexiones:

- 2-Conectores RJ-45 para soportar interfazs RS-232, RS-422 y RS-485.
- 1-Conector herramienta de configuración RJ-45/RS-232

Interfaz ControlLogix

Operación vía simple lógica de escalera.

Completa inicialización y monitoreo del módulo a través de software RSLogix5000.

Interfaz con backplane ControlLogix vía acceso E/S.

Todos los datos relativos al módulo están contenidos en un Tag Controlador sencillo con objetos definidos para facilitar la configuración, el monitoreo y la interfaz con el módulo.

La configuración del módulo y la configuración de la comunicación de datos es transferida al MVI56-MCM via tipos de datos predefinidos por el usuario en el procesador.

Especificaciones de Hardware

El módulo MVI56-MCM esta diseñado por ProSoft Technology Inc. e incorpora tecnología licenciada de Allen-Bradley (tecnología de “Backplane” de ControlLogix).

Cargas de Corriente: 800 ma @ 5V (from backplane)

Temperatura de Operación: 0 to 60 Deg C (32 to 140 Deg F)
Temp. de Almacenamiento: -40 to 85 Deg C (-40 to 185 Def F)
Humedad Relativa: 5-95% (w/o condensation)
Conector Puerto Modbus: Dos Conectores RJ45 (Cable RJ45 to DB9
Entregado con la unidad)
Soporta interfazs RS-232, RS-422
E interfazs RS-485 (cable RJ45 to DB9
entregados con la unidad)
Conector de Configuración: Conector RJ45
RS-232 (cable RJ45 to DB9 entregados con la
unidad)

Soporte, Servicio y Garantía

Soporte Técnico

ProSoft Technology Inc. provee soporte a sus compradores. Si alguna pregunta o problema se presenta, favor siéntase libre de contactarnos a:

Factory/Technical Support
ProSoft Technology, Inc.
1675 Chester Avenue, 4th Floor
Bakersfield, CA 93301
(661) 716-5100
(661) 716-5101 (fax)

E-mail address: prosoft@prosoft-technology.com

Web Site: <http://www.prosoft-technology.com>

Antes de llamarnos para que le brindemos soporte, por favor prepárese para llamarnos. De manera de proveerle el mejor y más rápido soporte posible, nos gustaría preguntarle la siguiente información:

(Usted podría desear enviarnosla por fax antes de llamarnos):

1. Número de Versión del Producto
2. Jerarquía del Sistema
3. Configuración del Módulo y Contenido del Archivo MCM.CFG.
4. Operación del Módulo
 - Información de status de configuración/depuración
 - Patrones de Leds
5. Información acerca del procesador y las áreas de datos como se visualizan a través del RSLogix500 y el patrón de LED's del procesador.
6. Detalles sobre la red serial.

Un sistema de respuesta fuera de horario (en el número Baskensfield) le permitirá compágnar el acceso a uno de cuatro técnicos calificados y/o ingenieros de soporte de aplicación en cualquier momento para responder preguntas que son importantes para usted.

Servicio y Reparación del Módulo

La tarjeta MVI56-MCM es un producto electrónico diseñado y elaborado para funcionar bajo algunas condiciones adversas. Como cualquier producto a través de los años, malas aplicaciones o cualquiera de muchos posibles problemas, la tarjeta pudiera requerir reparación. Una vez comprado el módulo a ProSoft Technology, el módulo tiene un año de garantía en partes y labor de acuerdo a los límites especificados en la garantía. El reemplazo y/o retorno debería ser dirigido al distribuidor al cual el producto fue comprado. Si usted necesita retornar la tarjeta para su reparación, obtenga un número RMA de ProSoft Technology. Por favor, llame a la fábrica por éste número y muestre el número prominentemente en la parte externa del cartón de envío usado para retornar la tarjeta.

Política General de Garantía

ProSoft Technology, Inc. (en adelante referido como ProSoft) garantiza que el producto conforme y de acuerdo en las especificaciones técnicas publicadas y los materiales escritos que le acompañan y libre de defectos en materiales y mano de obra, por el período de tiempo aquí indicado, tal periodo de garantía comenzando desde el recibo del producto.

Esta garantía esta limitada a reparación y/o reemplazo, a elección de ProSoft no se responsabilizará por la falla del producto por desarrollar funciones específicas, o cualquier otra causa no conforme por o atribuible a: a) Cualquier desaplicación o desuso del producto; b) Falla del comprador de adherirse a cualquiera de las especificaciones o instrucciones de ProSoft. C) Negligencia, abuso o accidente con este producto o d) Cualquier asociación o equipo complementario o software no aceptado por ProSoft.

El servicio de garantía limitada puede ser obtenido por entrega del producto a ProSoft y proveyendo pruebas de compra o fecha de recepción. El comprador está de acuerdo en asegurar el producto o asumir el riesgo de pérdida o daños en tránsito para prepagar cargos de envío a ProSoft y para usar el contenedor original de envío o equivalente. Contacte a servicio al cliente de ProSoft para futura información.

Limitación de Responsabilidad

EXCEPTO COMO QUEDA EXPRESAMENTE PROBADO, PROSOFT NO GARANTIZA NINGUNA CLASE, EXPRESA O IMPLICITA, CON RESPECTO A CUALQUIER EQUIPO, PARTES O SERVICIOS PROVISTOS CONFORME A ESTE ACUERDO, INCLUYENDO PERO NO LIMITADO A LAS GARANTÍAS IMPLICADAS DE HABILIDAD MERCANTIL Y FITNES PARA SU PROPOSITO PARTICULAR. NI PROSOFT O SUS SOCIOS SERÁN RESPONSABLES POR CUALQUIERA OTROS DAÑOS, INCLUIDOS PERO NO LIMITADOS DIRECTA, INDIRECTA, ACCIDENTAL O ESPECIALMENTE O DAÑOS CONSECUENCIALES, QUE TENGAN UNA ACCION EN CONTRATO O (TORT) (INCLUYENDO NEGLIGENCIA Y RESPONSABILIDAD ESTRICTA), TALES COMO, PERO NO LIMITADO A, PERDIDAS ANTICIPADAS DE GANANCIAS O BENEFICIOS QUE RESULTAREN, O , O EN CONEXIÓN CON EL USO DE MUEBLES O EQUIPOS, PARTES, SERVICIOS O LA EJECUCION, USO O INHABILIDAD PARA USAR EL MISMO, SI LA RESPONSABILIDAD TOTAL DE PROSOFT O SUS EXCEDIO EL PRECIO PAGADO POR EL PRODUCTO.

Donde dirigidos por Leyes de Estado, algunas de las exclusiones hechas anteriormente o limitaciones pueden no ser aplicables en algunos estados. Esta garantía provee derechos legales específicos; otros derechos que varían de estado en estado también pueden existir. Esta garantía no será aplicable a la extensión que cualquier provisión de esta garantía son prohibidas por cualquier Ley Federal, Estatal o Municipal que no pueda ser.

Detalles de Garantía y Producto Software

Período de Garantía: ProSoft garantiza el Hardware del producto por un periodo de un (1) año.

Procedimiento de Garantía: Sobre el retorno del producto hardware, ProSoft, como una opción, reparará o reemplazará el producto sin cargo adicional, carga prepagada,

excepto como se estableció anteriormente. Las partes de reparación y productos de reemplazo no serán suplidos sobre una base de intercambio y serán ya sea reacondicionados o nuevos. Todo producto reemplazado y partes se convertirán en propiedad de ProSoft. Si ProSoft determina que el producto no esta bajo garantía, será opción del comprador reparar el producto usando rangos estándar actuales de ProSoft para las partes y labor y retornar el producto de carga por cobrar.

-----**FIN DEL MANUAL**-----