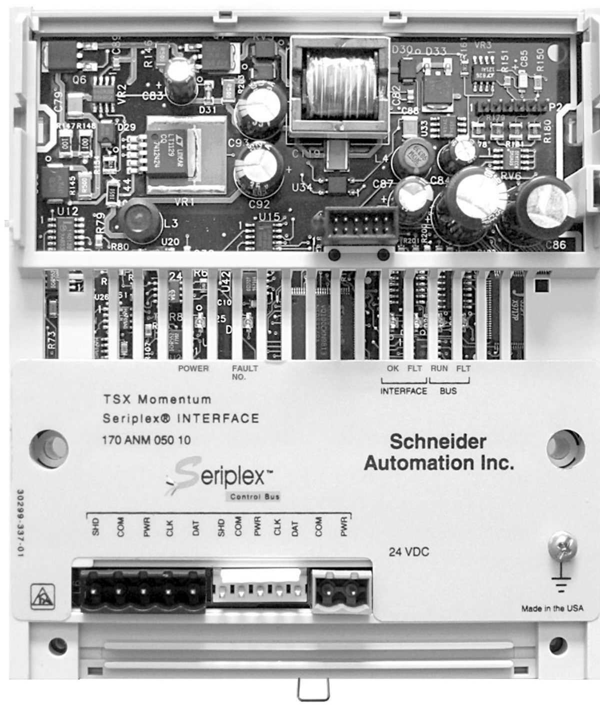
MOMENTUM™ SERIPLEX®
Interface Base Unit
170 ANM 050 10



Merlin Gerin
Modicon
Square D
Telemecanique

Schneider Electric

Data, Illustrations, Alterations

Data and illustrations are not binding. We reserve the right to alter products in line with our policy of continuous product development. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us by e-mail at techcomm@modicon.com.

Training

Schneider Automation Inc. offers suitable further training on the system.

Hotline

See addresses for Technical Support Centers at the end of this publication.

Trademarks

All terms used in this publication to denote Schneider Automation Inc. products are trademarks of Schneider Automation Inc.

All other terms used in this publication to denote products may be registered trademarks and/or trademarks of the corresponding corporations. Microsoft and MS-DOS are registered trademarks of Microsoft Corporation. Windows is a brand name of Microsoft Corporation in the USA and other countries. IBM is a registered trademark of International Business Machines Corporation. Intel is a registered trademark of Intel Corporation.

MOMENTUM™ SERIPLEX®
Interface Base Unit
170 ANM 050 10

March 1999

Schneider Electric

Preface

The data and illustrations found in this book are not binding. We reserve the right to modify our products in line with our policy of continuous product development. The information in this document is subject to change without notice and should not be construed as a commitment by Schneider Electric, Inc.

Schneider Electric, Inc assumes no responsibility for any errors that may appear in this document. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us through your distributor or local Square D office.

CAUTION

All pertinent state, regional, and local safety regulations must be observed when installing and using this product.

For reasons of safety and to assure compliance with documented system data, repairs to components should be performed only by the manufacturer.

Failure to observe this precaution can result in injury or equipment damage.

Printed in U.S.A.

Contents

List of Illustrations

List of Tables

MOMENTUM™ SERIPLEX®
Interface Base Unit
170 ANM 050 10

At a Glance

Purpose

This book describes the 170 ANM 050 10 Seriplex interface base unit.

Document Scope

This manual describes the features, installation, operating behavior, and configuration and programming methods of the MOMENTUM-SERIPLEX interface base unit. For information about SERIPLEX system design and operation, refer to *SERIPLEX Design, Installation, and Troubleshooting,* document number 30298-035-01_, included on the CD-ROM accompanying this unit. For general information about the nature and purpose of SERIPLEX control bus technology, see the SERIPLEX Technology Organization's *White Paper: The SERIPLEX Control Bus,* available from Square D Company as document number 8310PD9501.

Refer to the appropriate MODICON® product manuals for information about MOMENTUM processors and communication adapters, MODSOFT™ PLC programming software, and CONCEPT PLC programming software.

For a hard copy of this manual, call the Literature Fulfillment Center at (800)888-2448. Ask for document number 870 USE 002 00 V.2.

Module Description

The MOMENTUM™-SERIPLEX® interface base unit (170 ANM 050 10) enables the exchange of I/O data between a MOMENTUM processor adapter or communications adapter and a SERIPLEX bus. The interface takes the form of a MOMENTUM I/O base unit, on which a MOMENTUM processor or communications adapter can be mounted.

The interface base unit acts as a SERIPLEX bus master device, performing all functions necessary to host a SERIPLEX control bus. The SERIPLEX bus data appears as normal I/O memory registers to the host PLC or controller.

An M1 processor adapter can mount on the interface base unit, to control and monitor SERIPLEX I/O devices directly. A MOMENTUM option adapter can also be used, so that multiple M1 processors can be networked to create a distributed control application.

MOMENTUM communications adapters can also mount on the interface, creating a gateway between the SERIPLEX control bus and a higher-level communications network. In this case, a remote host could monitor and control the SERIPLEX bus data through the communications adapter. The communications adapter performs the translation of the I/O register data in the interface to the data format of the other network.

The MOMENTUM-SERIPLEX interface base unit uses 32 input and 32 output registers. The first *output* register is used for interface control and configuration, and the first *input* register is used to report status information. The remaining registers can be used for SERIPLEX bus input and output data.

Important User Information

It is possible that corrupted information in the control system described in this document can cause the system to report an incorrect input, or incorrectly operate an output. Where personnel and/or equipment hazards exist, use appropriate interlocking. Those responsible for the application, implementation, and use of the control system must ensure that the necessary design considerations have been incorporated into each application, completely adhering to applicable laws, performance and safety requirements, regulations, codes, and standards.

This system offers significant advantages in communication throughput, I/O number, and control bus length. Proper application will result in a reliable, high-performance control system.

WARNING

Hazard of unexpected output actuation

Where personnel and/or equipment hazards exist, use the appropriate hard-wired safety interlocks.

Failure to follow this instruction can result in death or serious injury.

WARNING

Unintentional Equipment Operation

The application of this product requires expertise in the design and programming of control systems. Only persons with such expertise should be allowed to program, install, alter, and apply this product

Failure to follow this instruction can result in death or serious injury.

WARNING

Equipment Damage Hazard

To avoid improper handling of equipment:
- Never insert/remove the host adapter while the interface base unit power is on.
- Do not expose to electrostatic discharge (ESD). The control bus modules contain electronic components that are susceptible to damage from ESD.

Failure to observe this instruction can result in death or serious injury.

Continued on next page

Module Overview

Introduction

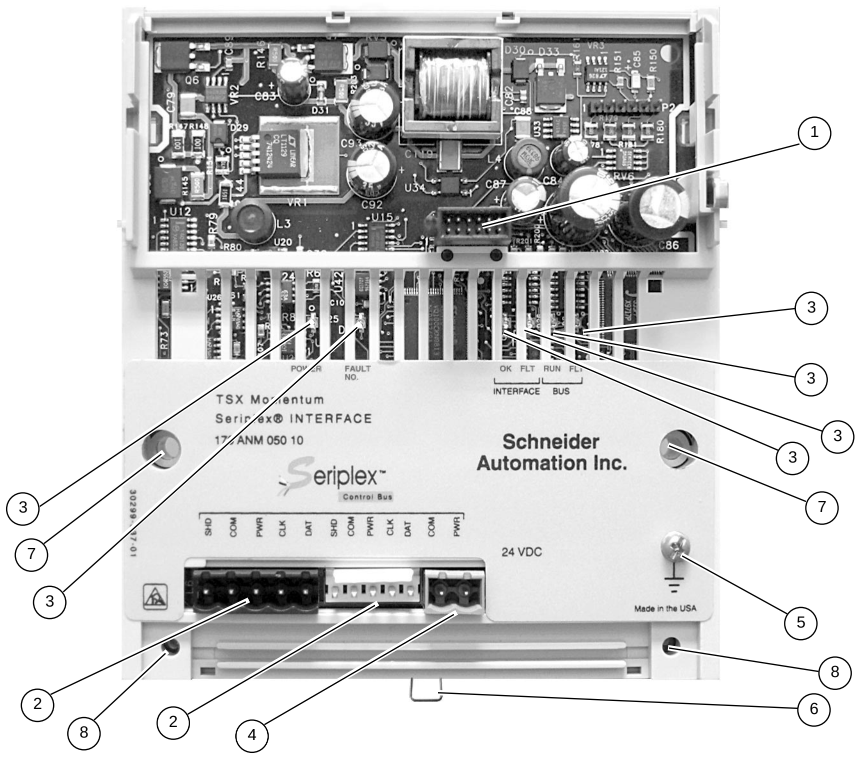
Purpose

This section describes the front panel components of the 170 ANM 050 10 Momentum Seripex interface base and provides specifications.

In This Section

This section contains the following topics:

Front Panel Components



Label	Description (See “Wiring Diagram” on page 9.)
1	Internal interface (ATI) connector
2	External Seriplex bus connections
3	LED status display (See Table 17 on page 71)
4	24Vdc power supply connector
5	Grounding screw
6	Locking tab for DIN rail mount
7	Mounting holes and ground contact for panel mount
8	Mounting holes for panel mount

Product Features

General features and capabilities

General features and capabilities of the MOMENTUM-SERIPLEX interface base unit are listed below. These features are described in more detail later in the document.

- Includes all necessary circuitry and logic provided to perform SERIPLEX control bus clock source functions while in master/slave operating mode
- Compatible with MOMENTUM M1 processor, communication, and option adapters — logic power provided
- Input and output data exchanged with the host processor/adapter through 32 16-bit input and 32 output registers
- All configuration performed through the first output register, status reported through first input register
- SERIPLEX version 2 bus input and output devices supported at 24 Vdc bus voltage
- Common SERIPLEX bus fault conditions detected and reported to host
- SERIPLEX bus data capacity of 992 discrete I/O devices, or 62 16-bit analog I/O signals
- SERIPLEX bus clock rates from 10 to 192 kHz supported
- Optional data error detection methods available
- Digital debounce
- Complementary data retransmission (CDR)
- Status indicator lamps visible through the front of the base unit
- User-selectable auto restart capability

Package Contents

The carton for the MOMENTUM-SERIPLEX interface base unit contains:

- one MOMENTUM-SERIPLEX interface base unit in a sealed, anti-static bag
- one *MOMENTUM™-SERIPLEX® Interface Base Unit* instruction booklet, numbered 30298-085-01.
- one MOMENTUM-SERIPLEX Interface Base Unit Accessory Disk, in CD-ROM form

Physical Installation

Installation To install the interface base unit, perform the following steps:

Din Rail

1. To mount the interface base unit on a 35-mm DIN rail:

Step	Action
a	Pull out the DIN rail's locking tab at the bottom of the unit
b	Position the unit over the DIN rail with the label reading right-side up (see Figure).
c	Snap the unit onto the DIN rail by hooking the top of the channel found on the back of the unit on the rail, then pushing the bottom of the unit against the rail.
d	Push in the locking tab to secure the unit.

Mounting the Interface Base Unit

2. To mount the interface base unit directly to a panel, insert mounting screws and washers through the two mounting holes. See Figure 1, below.

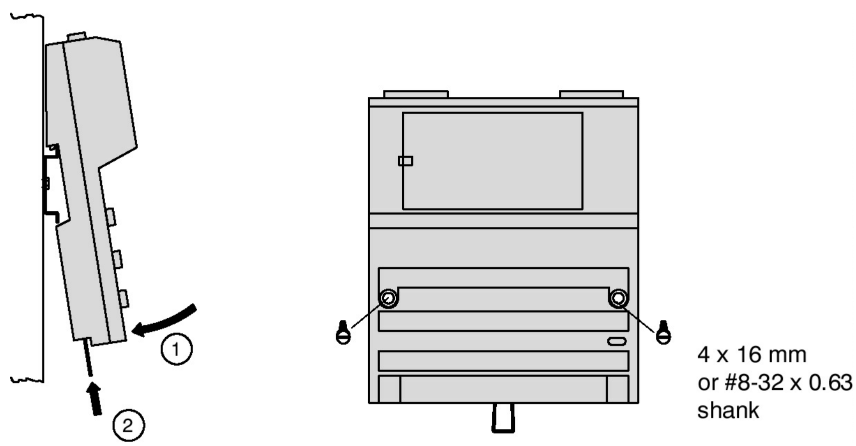


Figure 1. Mounting the Interface Base Unit

3. Attach a MOMENTUM communications adapter or processor adapter, as well as an option adapter (if desired), to the front of the interface base unit.

Continued on next page

Physical Installation, Continued

4. Connect the grounding screw on the front of the interface base unit to a sound earth ground. Grounding the screw greatly increases the unit's immunity to electrical noise, therefore reducing the chance of erroneous data transmission/ reception.

WARNING

Unintentional Equipment Operation

Connect the interface base unit's grounding screw to earth ground. Do not electrically connect the DIN rail or mounting screws to any point other than earth ground.

Failure to follow this instruction can result in death or serious injury.

5. See Figure 2. *Wiring Diagram* on page 9. Connect a SERIPLEX bus cable to one or both of the SERIPLEX wiring connectors on the front of the interface base unit, as directed by the diagram and label on the unit. Inside the base unit, like conductors (power and power, shield and shield, data and data, etc.) are connected to each other (a parallel connection).
   The screw-terminal connector will accept conductors from 22 to 16 AWG. Wire the insulation-displacement connector as instructed in the SERIPLEX version 2 product catalog from Square D (document number 8330CT9601).

6. Connect a 24 Vdc supply to the interface base unit. The power supply should be located close to the base unit, to reduce voltage drops. Other network power supplies may also be required. Refer to the *SERIPLEX Design, Installation, and Troubleshooting* (30298-035-01_). Power can be supplied through any of the three interface connectors, since these connectors are electrically connected inside the unit.

7. Apply power and test network operation, as instructed in *SERIPLEX Design, Installation, and Troubleshooting* (30298-035-01_).

Continued on next page

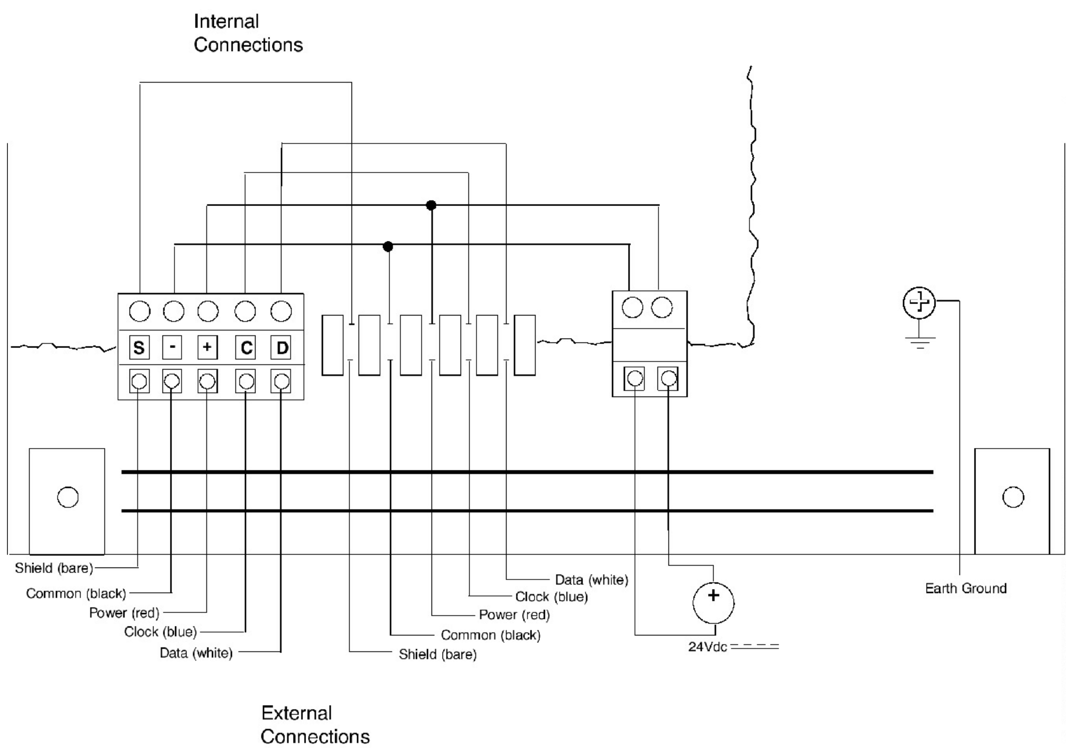


Figure 2. Wiring Diagram

Feature Descriptions

Overview

The following sections provide users a general understanding of the features and operation of the MOMENTUM-SERIPLEX interface base unit, with enough detail to help the user select the appropriate operating parameters for the base unit.

Operating the SERIPLEX Bus

The SERIPLEX bus starts and halts under the control of the host processor. These commands are given by writing to the least significant bit (bit 1) within the interface base unit's control register.

When the bus halts, the interface base unit does not generate the SERIPLEX bus clock signal, so no data is exchanged between bus I/O devices. Bus output devices will therefore assume their vendor-determined default states, usually corresponding to a signal state of 0, or off. The bus halts upon any interface base unit reset condition, including power-up. The bus also halts when a bus fault or internal card fault is detected, or when commanded by the host software.

When the bus is running, the interface base unit generates the SERIPLEX bus clock signal. Input and output data is exchanged normally between bus devices. The bus runs when commanded by the host software, or upon a successful auto restart that follows detection of a bus fault.

Reading and Writing SERIPLEX Bus Data

Host software can read SERIPLEX bus input data by reading the interface base unit's input data registers. The input data in the registers is the most recent data validated by the interface base unit. If digital debounce or CDR data validation is enabled for an input signal, the data in the registers might not be identical with that of the last data frame received by the interface base unit.

Host software can write bus output data by writing the intended data into the interface base unit's output data registers. The card transmits the output data through the SERIPLEX bus the next time the corresponding address(es) and multiplex channel are scanned by the interface base unit. Host software can also read bus output data from the card's output data registers.

The location of SERIPLEX bus data within the interface base unit's input and output data registers varies according to the card's selected operating parameters. Refer to the section "Programming" on page 17 for details of the register memory map.

Continued on next page

Feature Descriptions, Continued

Status Monitoring

The interface base unit provides indication directly to the user of both its internal status and bus status by means of indicator lights, found on the front of the unit. The base unit also provides extensive status information to the host processor through the status register.

The base unit provides an *interface-OK* indicator light to indicate that it has successfully passed the internal diagnostic tests. The base unit also has an *interface fault* indicator light to show that the card has detected an internal fault, such as a configuration error or hardware fault.

The unit also indicates that it is operating by toggling its *heartbeat* bit. At all times when operating properly, the interface base unit changes the value of this bit in the status register from 0 to 1 to 0, at a rate of one to two times per second.

SERIPLEX Bus Operating Parameters

The interface base unit allows the user to select the SERIPLEX bus clock rate and frame length through the control register. These operating parameters can be adjusted to match bus performance to the exact needs of a particular application.

Clock rate is the speed at which the bus operates. Faster clock rates result in more frequent updates of bus signals, but decrease the allowable length of the bus cable (see *SERIPLEX Design, Installation, and Troubleshooting*, document 30298-035-01_, for more information). The interface base unit supports clock rates of 10, 16, 25, 32, 64, 100, 147, and 192 kHz.

Frame length is the number of SERIPLEX bus addresses scanned by the interface base unit. For the fastest possible update of bus signals, specify a frame length no greater than the number of I/O points actually used by the application (see *SERIPLEX Design, Installation, and Troubleshooting*, document 30298-035-01_, for more information). The base unit supports frame lengths of 64, 128, 192, and 256 addresses.

Address Multiplexing

Address multiplexing is a means of expanding the data capacity of a SERIPLEX bus, by allowing multiple devices to share the same bus addresses. With address multiplexing, the interface base unit can exchange up to 31 16-bit analog input signals and 31 analog output signals (or 496 discrete input and 496 discrete output signals) with the SERIPLEX bus.

Continued on next page

Feature Descriptions, Continued

Address Multiplexing, Continued

Multiplexed signals that share the same bus address are distinguished from one another by assigning each signal to one of 16 multiplex channels. The base unit scans one multiplex channel per data frame; devices assigned to the scanned channel communicate with the bus normally, while the other devices do not communicate with the bus until their assigned channels are scanned.

Address multiplexing is performed on data words (16 consecutive bus addresses). Each word corresponds to one data register. The base unit allows multiplexing of 50%, 75%, or 100% of available bus addresses. The multiplexing selection applies to both input and output words at the same bus address. See Figure 3, below.

Figure 3. Multiplex Channels

The interface base unit can be configured to scan four or eight multiplex channels. The number of channels is called the multiplex channel depth. The time required to update each multiplexed signal is calculated by multiplying the duration of one data frame by the number of scanned channels. However, non-multiplexed data on the same bus is still updated every data frame; therefore, the signal update time of non-multiplexed data is not affected by the use of address multiplexing.

Digital Debounce

Digital signal debounce stabilizes a discrete (one-bit) SERIPLEX input signal before reporting any change in its state to the host processor. Digital debounce can be used to filter both electrical noise and mechanical switch bounce — like that generated from an electromechanical relay — from bus input signals.

Continued on next page

Feature Descriptions, Continued

Digital Debounce, Continued

When digital debounce is enabled for an input signal, the interface base unit retains the value present in its memory until a new signal value has been reported for three consecutive data frames. This is not a voting function — the signal must remain at a new state continuously for the three consecutive frames in order for the base unit to report the changed signal state to the host.See Figure 4, below.

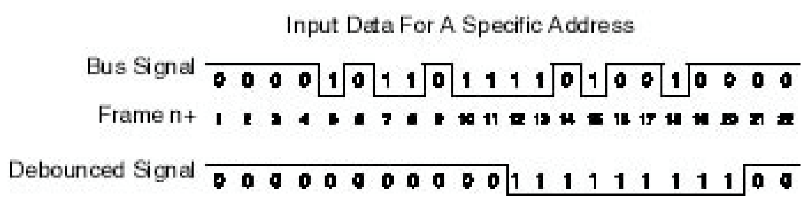


Figure 4. Digital Debounce Example

Since digital debounce is a function of the device that receives the bus signal, the base unit supports the debouncing of bus input signals only. Some bus output devices support debouncing of output signals.

When enabled, digital debounce is applied to all input signals that do not have address multiplexing or multi-bit data validation (using CDR) enabled. Digital debounce should be applied only to discrete input signals. Multi-bit and/or multiplexed input signals can be verified through complementary data retransmission (CDR).

Multi-Bit Data Validation with CDR

Multi-bit data validation is used to verify that signals are transmitted correctly from one SERIPLEX bus device to another. This is accomplished by a technique known as complementary data retransmission, or CDR. CDR is used for multi-bit data such as analog I/O signals, as well as for multiplexed signals. Discrete (one-bit) signals are generally verified through digital debounce or data echo (provided by some I/O devices), rather than CDR.

Continued on next page

Feature Descriptions, Continued

Multi-Bit Data Validation, Continued

When CDR is enabled for a signal, that signal's transmitting device transmits the signal data normally, and also transmits an encoded version of the data. The encoding mechanism incorporates the signal value, its SERIPLEX bus address, its multiplex channel number, and whether it is input or output data. The receiving device then compares the actual data to its encoded value, to verify that it has received the signal correctly.

CDR validation is performed on bytes (eight bits) of data. The signal data is contained in data words (16 bits), which begin at bus addresses that are multiples of 32 (32, 64, 96, etc.). The encoded data for each signal is contained in the following word, as shown in Table 1, below.

Table 1: CDR Signal Address Assignments

Address	0	8	16	24	32	40	48	56
Content	—	—	Data 16	Data 24	Data 32	Data 40	CDR 32	CDR 40
Address	64	72	80	88	96	104	112	120
Content	Data 64	Data 72	CDR 64	CDR 72	Data 96	Data 104	CDR 96	CDR 104
Address	128	136	144	152	160	168	176	184
Content	Data 128	Data 136	CDR 128	CDR 136	Data 160	Data 168	CDR 160	CDR 168
Address	192	200	208	216	224	232	240	248
Content	Data 192	Data 200	CDR 192	CDR 200	Data 224	Data 232	CDR 224	CDR 232

The interface base unit can enable CDR for 1) all bus addresses, 2) only the multiplexed portion of the bus data, or 3) all the multiplexed data, plus the last half of the non-multiplexed addresses. Any CDR selection applies to both input and output bytes at the same bus address.

When CDR data validation is enabled for a particular input byte, the interface base unit compares the signal's value with its encoded CDR value. If the two values match, the input data is placed in the memory of the base unit, for use by the host processor. If the encoded data does not match the actual data, the interface base unit rejects the data and leaves that signal's last validated value in the memory.

Continued on next page

Feature Descriptions, Continued

Multi-Bit Data Validation, Continued

By default, the interface base unit declares an input CDR fault for a signal immediately upon sensing a data mismatch. The base unit also provides an input CDR ride-through option, which instructs the card to wait for three consecutive mismatches before declaring a CDR fault. The ride-through option applies to all input signals — it is not applied selectively to individual signals.

Since it is possible that physical input devices might not be assigned to all SERIPLEX bus input bytes within the CDR address range, the base unit performs a CDR test at bus initialization. The base unit counts the number of CDR-enabled input bytes that have valid CDR check bytes, then reports the value in its status register. If this count is less than the proper number for a given application, the host control program should take appropriate action. Thereafter, the interface base unit ignores CDR errors for input bytes that were not validated during the initial test.

If, during the initial test, the base unit receives valid CDR check bytes for both bytes of a data word, the unit treats the entire word as a single data value. When a CDR error is detected for either byte in that word, both bytes are rejected. However, if the base unit receives a valid check byte for only one byte in a word, the two bytes are treated independently. A CDR error in one byte does not affect the value of the other byte in the word.

The interface base unit transmits CDR check values for all output signals in the CDR address range, regardless of whether a corresponding physical output device is connected to the SERIPLEX bus. Output CDR faults are detected by SERIPLEX bus output devices, not by the MOMENTUM-SERIPLEX interface base unit. However, output devices can inform the base unit that they have detected a fault by turning on SERIPLEX bus input 9. However, this method does not identify the particular output signal that has experienced the CDR fault. The base unit applies the CDR ride-through selection to output CDR fault detection, as well as to input CDR fault detection.

In addition to global CDR fault indication by means of indicator lights, the base unit reports the individual word addresses that have experienced input CDR faults to the status register.

Continued on next page

Feature Descriptions, Continued

Multi-Bit Data Validation, Continued

By default, the base unit halts the bus when a CDR fault is declared. The base unit also provides the option of ignoring CDR faults and continuing bus operation. If this option is selected, CDR faults are still detected and reflected in the unit's valid CDR byte count, but the bus continues normal operation.

Auto Restart

When the interface base unit detects a SERIPLEX bus fault, it halts the bus. By default, the base unit remains halted until commanded to restart by the host software. However, the auto restart feature, when enabled, instructs the base unit to attempt to restart the bus without a command from the host.

When auto restart is enabled and a bus fault is detected, the base unit waits 250 milliseconds, then tests the bus for faults. If a fault is still found, the base unit waits another 250 milliseconds before testing again, until all bus faults are cleared. Once no faults are found, the base unit starts the bus normally, activating all normal bus initialization procedures.

Auto restart is attempted continually until 1) the bus is able to restart, 2) the base unit receives a software command to halt the bus, or 3) the base unit is reset.

Only use the auto restart feature if you can assure proper and safe system operation when starting the SERIPLEX bus without direct, intentional intervention by an operator and/or by control software.

Programming

Operation

The general procedure to operate a MOMENTUM-SERIPLEX interface base unit through a control program is:

Step	Action
1	Verify that the base unit is in proper operating condition by reading its status register.
2	Set the initial values for all SERIPLEX bus output signals by writing the desired values to the base unit's output data registers.
3	Write the desired operating parameter values to the unit's control register.
4	Activate the operating parameters and start bus operation by setting the unit's *run/halt* bit in the control register.
5	After the bus has initialized, read the status code in the base unit to determine whether all CDR-enabled input devices are connected and communicating, as described in the section "Multi-Bit Data Validation with CDR" on page 13. Depending on frame length and clock rate, the CDR count may take several seconds.
6	During normal operation, read the bus input values and write the bus output values according to the requirements of the application.
7	Monitor for bus fault conditions by reading the status register in the interface base unit, and respond to faults, as appropriate for the application.
8	Halt operation of the bus by clearing the base unit's *run/halt* bit.

To help perform these tasks, user-loadable program function blocks are provided with the interface base unit. These instructions can be embedded in ladder-logic programs within either MODSOFT™ or CONCEPT™ PLC programming software, to help select operating parameters and determine the status of the interface base unit.

The following sections provide more detail about programming methods for the base unit. This manual presumes that the user is familiar with the PLC programming software used in his/her application. Refer to the applicable software user's manual for complete instructions covering the use of the programming software.

Continued on next page

Programming, Continued

Memory Map

The MOMENTUM-SERIPLEX interface base unit communicates with a host processor by means of 32 input registers and 32 output registers. The output registers are used to configure the base unit, start and stop the SERIPLEX bus, and transmit bus output data from the host to the base unit. The input registers are used to report the status of the base unit and the bus, and to transmit bus input data to the host. The usage of these input and output registers varies, depending on the interface configuration, as explained in the following paragraphs.

In all cases, the first output register is used to configure and control the interface base unit. The first input register is used to report the status of the base unit and the status of the bus to the host. The remaining registers are used to exchange bus input and output data with the host.

The following sections explain the detailed memory map and the operation of the control register, status register, and I/O data registers.

Control Register

The control register is used to set the base unit's operating parameters, and to start and stop the SERIPLEX bus.

Operating parameters become effective when the base unit's *run/halt* bit is set, instructing the base unit to start the SERIPLEX bus. Do not change operating parameters while the bus is running. If an attempt is made to write to the control register while the *run/halt* bit is set, the base unit will declare a configuration fault and the bus will halt.

Continued on next page

Programming, Continued

See Table 2 for definitions of individual bits in the control register.

Table 2: Control Register Bit Descriptions

Bit	Description
1	Run/halt
2	Auto restart
3	Debounce enable
4	Clock rate
5	
6	
7	Multiplex address range
8	
9	Frame length
10	
11	Multiplex channel depth
12	CDR ride-through
13	Ignore CDR errors
14	CDR address range
15	
16	Reserved

Bit 1: Run/Halt

This bit controls the operation of the SERIPLEX bus. Its values are:

0: Halt the bus. The clock signal is not transmitted, and all bus devices' output signals revert to their default states.

1: Run the bus. Under this condition, the base unit transmits the clock signal and exchanges data between the I/O data registers and the bus, unless the unit detects an error.

All base unit operating parameters are activated at the time the *run/halt* bit transitions from 0 to 1.

Continued on next page

Programming, Continued

Bit 2: Auto Restart

This bit enables or disables the base unit's auto restart feature. Its values are:

0: Disable auto restart

1: Enable auto restart

Refer to the section "Auto Restart" on page 16 for details covering this feature's operation.

Bit 3: Debounce Enable

This bit enables or disables the base unit's digital debounce feature. Its values are:

0: Disable digital debounce

1: Enable digital debounce

When enabled, digital debounce is applied to all input signals that do not have address multiplexing or CDR enabled. The debounced input signal addresses begin at input 16 and continue up to the starting address for either address multiplexing or CDR.

Refer to the section "Digital Debounce" on page 12 for details covering this feature's operation.

Continued on next page

Programming, Continued

Bits 4 – 6: Clock Rate

The value of these bits determines the bus clock rate. The clock rate is the frequency of the SERIPLEX bus clock signal, which the interface base unit transmits to the bus. The available values are shown in Table 3.

Table 3: Clock Rate Bit Values

Bit 6	Bit 5	Bit 4	Clock Rate
0	0	0	10.0 kHz
0	0	1	15.9 kHz
0	1	0	25.0 kHz
0	1	1	32.9 kHz
1	0	0	64.1 kHz
1	0	1	100.0 kHz
1	1	0	147.1 kHz
1	1	1	192.3 kHz

Bits 7 – 8: Multiplex Address Range

The value of these bits determines the bus addresses that have address multiplexing enabled. The address range is expressed as the percentage of the designated bus frame length. Available values are listed in Table 4 and Table 5.

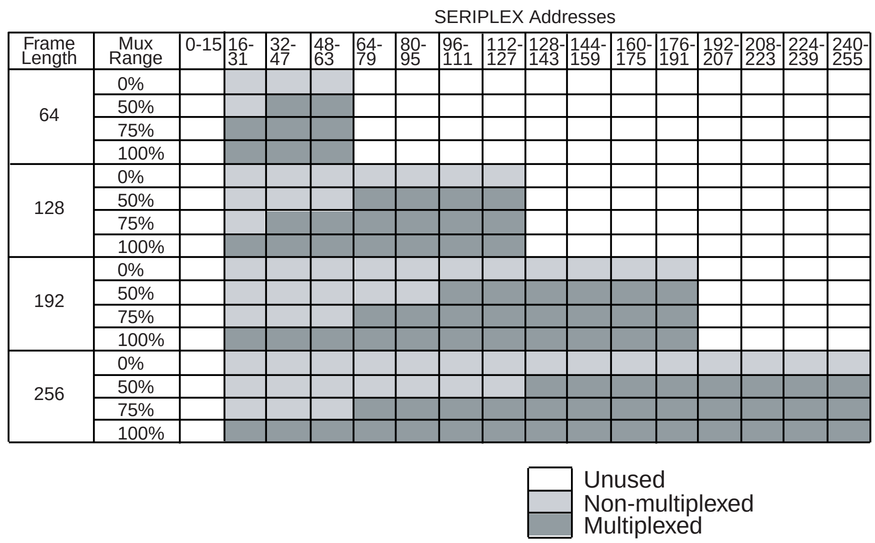
Table 4: Multiplex Address Range Bit Values

Bit 8	Bit 7	Multiplex Address Range
0	0	0% (disabled)
0	1	50%
1	0	75%*
1	1	100%
* The number of multiplexed addresses is not equal to 75% of frame length in all cases. Refer to Table 5, below.		

Continued on next page

Programming, Continued

Table 5: Multiplex Range Selections



Bits 7 – 8: Multiplex Address Range, Continued

The multiplexed address range occupies the highest numbered bus addresses within the frame. Once multiplexing begins at a designated address, it continues through the end of the data frame. Address multiplexing is performed on entire 16-bit words (registers) of bus data.

The range selected for address multiplexing affects the memory map of the I/O data registers. The base unit assigns registers to all multiplex channels of a data word before assigning registers to the next word. Examples of multiplex range selections and the resulting I/O register maps are shown in the section “Input and Output Data Registers” on page 32.

Continued on next page

Programming, Continued

Bits 9 – 10: Frame Length

The value of these bits determines the bus frame length. This is the number of SERIPLEX addresses that will be scanned by the interface. Available values are listed in Table 6.

Table 6: Frame Length Bit Values

Bit 10	Bit 9	Frame Length (Number of Addresses)
0	0	64
0	1	128
1	0	192
1	1	256

Since SERIPLEX bus addresses begin with address 0, the highest address scanned by the base unit is one less than the designated frame length. For example, if the frame length is set to 128, the highest address scanned by the base unit is address 127.

The MOMENTUM-SERIPLEX interface base unit does not provide access to addresses 0 through 15 in its I/O data registers. Therefore, the actual number of input or output signals available to the host is 16 less than the frame length.

The selected frame length affects the number of I/O data registers used by the base unit. Examples of frame length selections and the resulting I/O register maps are given in the section “Input and Output Data Registers” on page 32.

Bit 11: Multiplex Channel Depth

If multiplexing is disabled, the unit ignores this bit. When address multiplexing is enabled, the *multiplex channel depth* bit determines the number of multiplex channels scanned by the interface base unit. Its values are:

0: Scan 4 multiplex channels

1: Scan 8 multiplex channels

Continued on next page

Programming, Continued

Bit 12: CDR Ride-Through

When CDR data validation is enabled for any SERIPLEX bus data, this bit determines whether the interface base unit will respond immediately to any detected CDR error. Its values are:

0: Respond immediately to any detected input CDR error or output CDR error

1: Do not respond to any detected input CDR error or output CDR error, unless a CDR error is detected for three consecutive data frames

Bit 13: Ignore CDR Errors

When CDR data validation is enabled for any SERIPLEX bus data, this bit determines whether the base unit will halt the bus in response to any detected CDR error. Its values are:

0: Halt the bus upon detection of any input CDR error or output CDR error

1: Do not halt the bus upon detection of any input CDR error or output CDR error

Bits 14 – 15: CDR Address Range

The value of these bits determines the bus addresses that will have CDR multi-bit data validation enabled. Available values are listed in Table 7 below:

Table 7: CDR Address Range Bit Values

Bit 15	Bit 14	CDR Address Range	Explanation
0	0	CDR disabled	CDR not transmitted or evaluated
0	1	50% non-multiplexed	CDR enabled for all multiplexed bus addresses, plus half of the non-multiplexed addresses
1	0	All multiplexed	CDR enabled for all multiplexed bus addresses, but no non-multiplexed addresses
1	1	All	CDR enabled for all bus addresses

The CDR address range occupies the highest-numbered bus addresses within the frame. Once CDR begins at a designated address, it continues through the end of the data frame.

Continued on next page

Programming, Continued

Bits 14 – 15: CDR Address Range, Continued

The selected CDR address range affects the memory map of the I/O data registers. The designation of a given data word for CDR data validation removes the following data word from the register map. Examples of CDR range selections and the resulting I/O register maps are described in "Input and Output Data Registers" on page 32.

Bit 16: *Reserved*

This bit is reserved for a specific use by future versions of the interface base unit. Always write a value of 0 to this bit. If a value of 1 is written to the bit, the base unit will declare a configuration fault.

Status Register

The status register reports the condition of the interface base unit and the SERIPLEX bus. The register consists of several individual status bits, plus an 8-bit status code. Definitions of the bits are given in Table 8.

Table 8: Status Register Bit Definitions

Bit	Description
1	Bus running
2	Configuration fault
3	Bus fault
4	MOMENTUM fault
5	CDR fault
6	Heartbeat
7	Firmware version
8	Reserved
9	Status code
10	
11	
12	
13	
14	
15	
16	

Continued on next page

Programming, Continued

Bit 1: Bus Running

This bit indicates whether the SERIPLEX bus is running, that is, whether the base unit is transmitting the clock signal. Its values are:

0: Bus halted (no clock signal transmission)

1: Bus running (base unit transmitting the clock signal)

The value of this bit normally corresponds to the value of the control register's run/halt bit, unless the base unit has detected an error.

Bit 2: Configuration Fault

This bit indicates whether the base unit has detected a configuration fault. Its values are:

0: No configuration fault detected

1: Configuration fault detected

The specific type of configuration fault can be determined from the value of the status code. The two types of configuration faults detected by the base unit are 1) an attempt was made to change the value of the control register while the bus was running and 2) an attempt was made to set a value of 1 to control register bit 16.

The configuration fault bit is cleared when a value of 0 is written to the base unit's *run/halt* bit, to allow re-initialization of the interface base unit. Auto restart will not clear a configuration fault.

Continued on next page

Programming, Continued

Bit 3: Bus Fault

This bit indicates whether the base unit has detected a Seriplex bus fault. Its values are:

0: No SERIPLEX bus fault detected

1: SERIPLEX bus fault detected

The specific type of bus fault can be determined from the value of the status code. The base unit can detect any of the following fault conditions, listed in order of highest to lowest priority:

1. undervoltage
2. data line stuck low
3. data line stuck high
4. clock fault
5. data line undercurrent
6. data line overcurrent
7. excess data line capacitance
8. input CDR error
9. output CDR error

If the base unit detects more than one type of fault at a time, the status code and fault type light indicates the highest priority fault. Refer to “Troubleshooting” on page 75 for possible causes of each bus fault.

The *bus fault* bit is cleared when 1) a value of 0 is written to the base unit’s *run/halt* bit, to allow re-initialization of the unit, or 2) the base unit successfully accomplishes an auto restart without detecting a bus fault.

Continued on next page

Programming, Continued

Bit 4: MOMENTUM Fault

This bit indicates whether the base unit has detected a problem in the exchange of data between the unit and its host processor or communications adapter. Its values are:

0: No MOMENTUM fault detected

1: MOMENTUM fault detected

The specific type of MOMENTUM fault can be determined from the status code value. The base unit can detect data flow errors. Refer to Table 9: *Status Code Contents* on page 30 for possible causes of the fault.

The *MOMENTUM fault* bit is normally cleared by cycling power to the base unit. However, if the host adapter is able to re-establish communications with the interface base unit without a power cycle, the *MOMENTUM fault* bit will clear when the base unit's *run/halt* bit is set to 1.

Bit 5: CDR Fault

This bit indicates whether the base unit has declared an input or output CDR fault. Its values are:

0: No CDR fault

1: CDR fault

The status code indicates whether the detected fault is an input or an output CDR fault. The *CDR fault* bit itself only indicates whether the base unit has declared a CDR fault and halted the bus. However, in some instances, the base unit may detect a CDR error without declaring a CDR fault. The unit's response to CDR errors is determined by the *CDR ride-through* bit and the *ignore CDR errors* bit in the control register.

Continued on next page

Programming, Continued

Bit 5: CDR Fault, Continued

The *CDR fault* bit is cleared when the base unit scans a data frame without detecting a CDR error, or when a value of 0 is written to the *run/halt* bit to stop the bus and allow the unit to re-initialize.

Bit 6: Heartbeat

When the base unit is operating properly, this bit toggles between values of 0 and 1 continuously, at a rate between one and two times per second. If one second elapses without a change in state of this bit, the host software should assume that the interface is not operating properly and take action appropriate for the application.

Bit 7: Firmware Version

This bit indicates that the interface is displaying its firmware version number in the status code bits. Its values are:

0: Firmware version number not indicated by status code

1: Firmware version number indicated by status code

The interface base unit displays its firmware version number at all times when the *run/halt* bit is cleared to 0, as long as there are no detected fault conditions. See “Bits 9 – 16: Status Code” for an explanation of the data format of the firmware version number.

Bit 8: Reserved

This bit is reserved for a specific use by future versions of the interface base unit. Its value will always be 0.

Bits 9 – 16: Status Code

These bits indicate detailed status information about the interface or the SERIPLEX bus. The meaning of this eight-bit code varies according to the conditions of other bits within the status register.

Continued on next page

Programming, Continued

Table 9: Status Code Contents

Bus Run	Config Fault	Bus Fault	Mom. Fault	CDR Fault	Firm Vers	Status Code Meaning
0	0	0	0	0	1	Firmware version number
1	0	0	0	0	0	Valid CDR byte count
0	1	x	x	0	0	Fault code
0	x	1	x	0	0	
0	x	x	1	0	0	
0	0	0	0	1	0	CDR fault location(s)
x = don't care						

Under internal hardware or firmware fault conditions, the base unit will not communicate with the host adapter, and therefore cannot report its status code. Status code values and their meanings under various conditions are described below.

Normal Bus Halt Condition: FirmwareVersion Number

While the SERIPLEX bus is halted under normal conditions, the status code indicates the interface base unit's firmware version number. This number is a two-digit BCD (binary coded decimal) value, with the most significant nibble (four bits) indicating the major version number, and the least significant nibble indicating the minor version number. For example, firmware version 1.3 would be indicated as 0x13 hexadecimal, or 00010011 binary.

The *firmware version* bit is set under this condition, to indicate that the status code is indicating the firmware version number.

Normal Bus Run Condition: Valid CDR Byte Count

While the bus is operating under normal conditions, the status code indicates the number of bytes of input data for which the base unit has detected valid CDR check codes. Host software can use this number to verify that the proper number of bus input devices are connected and communicating through the bus, as described in the section "Multi-Bit Data Validation with CDR" on page 13.

Continued on next page

Programming, Continued

Normal Bus Run Condition: Valid CDR Byte Count, Continued

When determining the valid CDR byte count, the base unit counts only bytes that have CDR enabled. The CDR byte count is presented as an eight-bit, binary number, with bit 9 as the least significant bit. The byte count varies as devices are connected to or disconnected from the bus, or if input CDR errors are detected. Depending on frame length and clock rate, the CDR count may take several seconds.

Non-CDR Fault Conditions: Fault Codes

If the *configuration fault*, *bus fault*, or *MOMENTUM fault* bit value is 1, the status code indicates a fault code value. These fault conditions are not indicated simultaneously — only one bit is set to 1 at a given time. If more than one bus fault condition exists at a time, the lowest applicable fault code value is indicated.

The fault code is an eight-bit, binary number, with bit 9 as the least significant bit. Fault code values and corresponding error conditions are listed in Table 10.

Table 10: Fault Code Values

Decimal	Hexadecimal	Description
0	0x00	Bus undervoltage
1	0x01	Bus data stuck low
2	0x02	Bus data stuck high
3	0x03	Clock fault
4	0x04	Bus data undercurrent
5	0x05	Bus data overcurrent
6	0x06	Bus excess capacitance
7	0x07	Bus input CDR fault
8	0x08	Bus output CDR fault
100	0x64	Configuration changed while bus was running
150	0x96	Data flow error
200	0xC8	Attempt to set control register bit 16 to value of 1

Continued on next page

Programming, Continued

CDR Fault Conditions: CDR Fault Location Table

If the *CDR fault* bit is set to 1 and all other fault bits are cleared to 0, the status code indicates the address of the CDR fault. Multiple CDR faults can be indicated simultaneously. The status code does not represent a single numeric value; instead, the meaning of each bit within the status code is independent of the other bits. The CDR fault code is defined in Table 11 below.

Table 11: CDR Fault Code

Bit	**Location of CDR Error**
8	Output CDR error
9	Input CDR error at addresses 32 to 47
10	Input CDR error at addresses 64 to 79
11	Input CDR error at addresses 96 to 111
12	Input CDR error at addresses 128 to 143
13	Input CDR error at addresses 160 to 175
14	Input CDR error at addresses 192 to 207
15	Input CDR error at addresses 224 to 239

Note: The CDR fault code indicates only the SERIPLEX bus addresses that experienced an input CDR error, and not the specific multiplex channel(s) in which the error occurred.

Input and Output Data Registers

The MOMENTUM-SERIPLEX interface base unit uses 31 input data registers and 31 output data registers. These registers are assigned immediately following the control register and status register. Input registers report bus input data to the host adapter, while output registers receive bus output data from the host.

The number of registers that actually contain bus data, and the sequence in which the data appears, varies according to the interface base unit's operating parameters. The number and sequence of the registers is the same for both input and output data.

Registers are allocated until either all SERIPLEX bus data is available, or the last assigned register has been used. If fewer than 31 data registers are required, the unused input registers report data values of 0, and the unused output registers are ignored by the base unit. If the configuration requires more than 31 registers, some bus data is not accessible to the host.

Continued on next page

Programming, Continued

Input and Output Data Registers, Continued

Bus addresses 0 through 15 are reserved for system use and are not available to the host processor.

Within each data register, bit 1 corresponds to the lowest bus address, and bit 16 corresponds to the highest address, as shown in Table 12.

Table 12: Register Map

Register Bit	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
SERIPLEX Bus Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	79	78	77	76	75	74	73	72	71	70	69	68	67	66	65	64
	95	94	93	92	91	90	89	88	87	86	85	84	83	82	81	80
	111	110	109	108	107	106	105	104	103	102	101	100	99	98	97	96
	127	126	125	124	123	122	121	120	119	118	117	116	115	114	113	112
	143	142	141	140	139	138	137	136	135	134	133	132	131	130	129	128
	159	158	157	156	155	154	153	152	151	150	149	148	147	146	145	144
	175	174	173	172	171	170	169	168	167	166	165	164	163	162	161	160
	191	190	189	188	187	186	185	184	183	182	181	180	179	178	177	176
	207	206	205	204	203	202	201	200	199	198	197	196	195	194	193	192
	223	222	221	220	219	218	217	216	215	214	213	212	211	210	209	208
	239	238	237	236	235	234	233	232	231	230	229	228	227	226	225	224
	255	254	253	252	251	250	249	248	247	246	245	244	243	242	241	240

The rules for register allocation are illustrated through the examples in the next section. These examples begin with the simplest case, a non-multiplexed system with no CDR data validation. Examples are also given for multiplexed systems and systems that enable CDR.

Continued on next page

Programming, Continued

Register Allocation with Address Multiplexing Disabled and CDR Disabled

Register allocation is simplest when neither address multiplexing nor CDR data validation is enabled. Registers are allocated in ascending numerical order, with each register corresponding to 16 SERIPLEX addresses. The number of registers used is determined by the base unit's selected frame length.

Examples of register allocation for non-multiplexed systems without CDR are given in Table 13 on page 35.

Continued on next page

Table 13: Register Allocation — Non-Multiplexed Systems

MOMENTUM Register Numbers	SERIPLEX Bus Addresses	
	Frame Length 96	**Frame Length 256**
3xxxx/4xxxx + 0	status/control	status/control
3xxxx/4xxxx + 1	16 - 31	16 - 31
3xxxx/4xxxx + 2	32 - 47	32 - 47
3xxxx/4xxxx + 3	48 - 63	48 - 63
3xxxx/4xxxx + 4	64 - 79	64 - 79
3xxxx/4xxxx + 5	80 - 95	80 - 95
3xxxx/4xxxx + 6	unused	96 - 111
3xxxx/4xxxx + 7	unused	112 - 127
3xxxx/4xxxx + 8	unused	128 - 143
3xxxx/4xxxx + 9	unused	144 - 159
3xxxx/4xxxx + 10	unused	160 - 175
3xxxx/4xxxx + 11	unused	176 - 191
3xxxx/4xxxx + 12	unused	192 - 207
3xxxx/4xxxx + 13	unused	208 - 223
3xxxx/4xxxx + 14	unused	224 - 239
3xxxx/4xxxx + 15	unused	240 - 255
3xxxx/4xxxx + 16	unused	unused
3xxxx/4xxxx + 17	unused	unused
3xxxx/4xxxx + 18	unused	unused
3xxxx/4xxxx + 19	unused	unused
3xxxx/4xxxx + 20	unused	unused
3xxxx/4xxxx + 21	unused	unused
3xxxx/4xxxx + 22	unused	unused
3xxxx/4xxxx + 23	unused	unused
3xxxx/4xxxx + 24	unused	unused
3xxxx/4xxxx + 25	unused	unused
3xxxx/4xxxx + 26	unused	unused
3xxxx/4xxxx + 27	unused	unused
3xxxx/4xxxx + 28	unused	unused
3xxxx/4xxxx + 29	unused	unused
3xxxx/4xxxx + 30	unused	unused
3xxxx/4xxxx + 31	unused	unused
For all addresses, mux range = 0%; channels = don’t care; CDR disabled.		

Continued on next page

Programming, Continued

Register Allocation with Address Multiplexing Enabled and CDR Disabled

When address multiplexing is enabled, registers are allocated for all multiplex channels in a data word in ascending numerical order, then for all channels of the next data word. Examples of register allocation for multiplexed systems without CDR are given in Table 14.

Continued on next page

Table 14: Register Allocation for Multiplexed Systems

MOMENTUM Register Numbers	**SERIPLEX Bus Addresses**		
	Frame Length 192 4 Channels, 50% Mux Range	**Frame Length 128 4 Channels, 100% Mux Range**	**Frame Length 64 8 Channels, 100% Mux Range**
3xxxx/4xxxx + 0	**status/control**	**status/control**	**status/control**
3xxxx/4xxxx + 1	16 - 31	16 - 31, channel 0	16 - 31, channel 0
3xxxx/4xxxx + 2	32 - 47	16 - 31, channel 1	16 - 31, channel 1
3xxxx/4xxxx + 3	48 - 63	16 - 31, channel 2	16 - 31, channel 2
3xxxx/4xxxx + 4	64 - 79	16 - 31, channel 3	16 - 31, channel 3
3xxxx/4xxxx + 5	80 - 95	32 - 47, channel 0	16 - 31, channel 4
3xxxx/4xxxx + 6	96 - 111, channel 0	32 - 47, channel 1	16 - 31, channel 5
3xxxx/4xxxx + 7	96 - 111, channel 1	32 - 47, channel 2	16 - 31, channel 6
3xxxx/4xxxx + 8	96 - 111, channel 2	32 - 47, channel 3	16 - 31, channel 7
3xxxx/4xxxx + 9	96 - 111, channel 3	48 - 63, channel 0	32 - 47, channel 0
3xxxx/4xxxx + 10	112 - 127, channel 0	48 - 63, channel 1	32 - 47, channel 1
3xxxx/4xxxx + 11	112 - 127, channel 1	48 - 63, channel 2	32 - 47, channel 2
3xxxx/4xxxx + 12	112 - 127, channel 2	48 - 63, channel 3	32 - 47, channel 3
3xxxx/4xxxx + 13	112 - 127, channel 3	64 - 79, channel 0	32 - 47, channel 4
3xxxx/4xxxx + 14	128 - 143, channel 0	64 - 79, channel 1	32 - 47, channel 5
3xxxx/4xxxx + 15	128 - 143, channel 1	64 - 79, channel 2	32 - 47, channel 6
3xxxx/4xxxx + 16	128 - 143, channel 2	64 - 79, channel 3	32 - 47, channel 7
3xxxx/4xxxx + 17	128 - 143, channel 3	80 - 95, channel 0	48 - 63, channel 0
3xxxx/4xxxx + 18	144 - 159, channel 0	80 - 95, channel 1	48 - 63, channel 1
3xxxx/4xxxx + 19	144 - 159, channel 1	80 - 95, channel 2	48 - 63, channel 2
3xxxx/4xxxx + 20	144 - 159, channel 2	80 - 95, channel 3	48 - 63, channel 3
3xxxx/4xxxx + 21	144 - 159, channel 3	96 - 111, channel 0	48 - 63, channel 4
3xxxx/4xxxx + 22	160 - 175, channel 0	96 - 111, channel 1	48 - 63, channel 5
3xxxx/4xxxx + 23	160 - 175, channel 1	96 - 111, channel 2	48 - 63, channel 6
3xxxx/4xxxx + 24	160 - 175, channel 2	96 - 111, channel 3	48 - 63, channel 7
3xxxx/4xxxx + 25	160 - 175, channel 3	112 - 127, channel 0	unused
3xxxx/4xxxx + 26	176 - 191, channel 0	112 - 127, channel 1	unused
3xxxx/4xxxx + 27	176 - 191, channel 1	112 - 127, channel 2	unused
3xxxx/4xxxx + 28	176 - 191, channel 2	112 - 127, channel 3	unused
3xxxx/4xxxx + 29	176 - 191, channel 3	unused	unused
3xxxx/4xxxx + 30	unused	unused	unused
3xxxx/4xxxx + 31	unused	unused	unused
For all addresses, CDR is disabled.			

Programming, Continued

Register Allocation With CDR Enabled

When CDR data validation is enabled, the interface base unit conserves register memory by allocating registers only to actual SERIPLEX bus data. SERIPLEX bus addresses that contain CDR check codes do not appear in the register map. Examples of register allocation for multiplexed systems with CDR are given Table 15 on page 39.

Continued on next page

Table 15: Register Allocation for Multiplexed Systems

MOMENTUM Register Numbers	SERIPLEX Bus Addresses		
	Frame Length 128 8 Channels 50% Mux Range CDR @ Half Non-Mux	**Frame Length 192 4 Channels 75% Mux Range CDR @ All Mux**	**Frame Length 256 4 Channels 100% Mux Range CDR @ All**
3xxxx/4xxxx + 0	status/control	status/control	status/control
3xxxx/4xxxx + 1	16 - 31	16 - 31	16 - 31, channel 0[1]
3xxxx/4xxxx + 2	32 - 47	32 - 47	16 - 31, channel 1[1]
3xxxx/4xxxx + 3	64 - 79, channel 0	48 - 63	16 - 31, channel 2[1]
3xxxx/4xxxx + 4	64 - 79, channel 1	64 - 79, channel 0	16 - 31, channel 3[1]
3xxxx/4xxxx + 5	64 - 79, channel 2	64 - 79, channel 1	32 - 47, channel 0
3xxxx/4xxxx + 6	64 - 79, channel 3	64 - 79, channel 2	32 - 47, channel 1
3xxxx/4xxxx + 7	64 - 79, channel 4	64 - 79, channel 3	32 - 47, channel 2
3xxxx/4xxxx + 8	64 - 79, channel 5	96 - 111, channel 0	32 - 47, channel 3
3xxxx/4xxxx + 9	64 - 79, channel 6	96 - 111, channel 1	64 - 79, channel 0
3xxxx/4xxxx + 10	64 - 79, channel 7	96 - 111, channel 2	64 - 79, channel 1
3xxxx/4xxxx + 11	96 - 111, channel 0	96 - 111, channel 3	64 - 79, channel 2
3xxxx/4xxxx + 12	96 - 111, channel 1	128 - 143, channel 0	64 - 79, channel 3
3xxxx/4xxxx + 13	96 - 111, channel 2	128 - 143, channel 1	96 - 111, channel 0
3xxxx/4xxxx + 14	96 - 111, channel 3	128 - 143, channel 2	96 - 111, channel 1
3xxxx/4xxxx + 15	96 - 111, channel 4	128 - 143, channel 3	96 - 111, channel 2
3xxxx/4xxxx + 16	96 - 111, channel 5	160 - 175, channel 0	96 - 111, channel 3
3xxxx/4xxxx + 17	96 - 111, channel 6	160 - 175, channel 1	128 - 143, channel 0
3xxxx/4xxxx + 18	96 - 111, channel 7	160 - 175, channel 2	128 - 143, channel 1
3xxxx/4xxxx + 19	unused	160 - 175, channel 3	128 - 143, channel 2
3xxxx/4xxxx + 20	unused	unused	128 - 143, channel 3
3xxxx/4xxxx + 21	unused	unused	160 - 175, channel 0
3xxxx/4xxxx + 22	unused	unused	160 - 175, channel 1
3xxxx/4xxxx + 23	unused	unused	160 - 175, channel 2
3xxxx/4xxxx + 24	unused	unused	160 - 175, channel 3
3xxxx/4xxxx + 25	unused	unused	192 - 207, channel 0
3xxxx/4xxxx + 26	unused	unused	192 - 207, channel 1
3xxxx/4xxxx + 27	unused	unused	192 - 207, channel 2
3xxxx/4xxxx + 28	unused	unused	192 - 207, channel 3
3xxxx/4xxxx + 29	unused	unused	224 - 239, channel 0
3xxxx/4xxxx + 30	unused	unused	224 - 239, channel 1
3xxxx/4xxxx + 31	unused	unused	224 - 239, channel 2

1. These addresses do not have CDR.

Programming Through MODSOFT

MSPX Block

The MOMENTUM-SERIPLEX interface base unit provides a custom-loadable block, named MSPX, for use in MODSOFT PLC programming software. The MSPX block can be incorporated into a MOMENTUM M1 processor program for easier configuration and monitoring of the interface base unit.

The MSPX block works with MODSOFT software version 2.51 and later versions. The installation procedure for the block varies slightly with different versions of MODSOFT software, but the use of the block is the same.

The interface base unit can be used with MODSOFT software without using the MSPX block, by directly reading and writing bits within the base unit's registers.

Installing the MODSOFT Custom-Loadable Block

Installation of the MSPX custom-loadable block differs slightly, depending on the version of MODSOFT software used. For all versions, the files are found in the directory of the CD-ROM disk shipped with the interface base unit.

Copy the MSPX block files onto your local hard disk, so they are available to the MODSOFT application. From the DOS prompt, use the copy command, or from Windows™ or DOS, use a file-management tool to perform the instructions listed under the appropriate heading below. In these instructions, "C:" represents the computer hard disk drive where MODSOFT programming software is installed; "D:" represents the CD-ROM drive that is reading the MOMENTUM-SERIPLEX CD-ROM disk.

MODSOFT PLC Programming Software Version 2.51

Step	Action
1	Copy the file D:\MOD251\mspx.dat to the C:\Modsoft\Loadable directory.
2	Rename the files C:\Modsoft\Runtime\dxfdt.sys to dxfdt.bak; m1tcop.sys to m1tcop.bak; and momentum.sys to momentum.bak.
3	Copy the files D:\MOD251\dxfdt.sys, m1tcop.sys, and momentum.sys to the C:\Modsoft\Runtime directory.

Continued on next page

Programming Through MODSOFT, Continued

Version 2.6

Step	Action
1	Copy the files D:\MOD26\mspx.zmm, mspx.dat, mspx 1938.dat, and mspx 193m.dat to the C:\Modsoft\Programs directory.
2	Rename the files C:\Modsoft\Runtime\m1tcop.sys to m1tcop.bak, and momentum.sys to momentum.bak.
3	Copy the files D:\MOD26\m1tcop.sys and momentum.sys to the C:\Modsoft\Runtime directory.

Version 2.61 or Later

Copy the files D:\MOD261\mspx.zmm, mspx.dat, mspx1938.dat, and mspx193m.dat to the C:\Modsoft\Programs directory.

Note: For Modsoft versions 2.6 and 2.61, it may be necessary to repeat installation step 1 in the following cases:

- A processor other than an M1 processor is selected, followed by selecting an M1 processor.
- The block is deleted from the list and then added back.

Completing Installation

If the files C:\Modsoft\Runtime\dxfdt.sys, m1tcop.sys, and/or momentum.sys have been modified since the original installation of the MODSOFT software, some loadable instructions may no longer be available after this installation. Contact the technical support group at Square D at (919)217-6509 for information about how to edit these files so that all instructions are available.
Once the files are copied onto your hard disk, install the MSPX block' instruction into the MODSOFT program:

Step	Action
1	At the segment status screen, press the <F5> key (config) to get to the configuration overview screen.
2	At the configuration overview screen, press <F7> (loadable) to get to the DX loadable configuration screen.
3	In the DX loadable configuration screen, press <F3> (dir) to view the directory of available, custom-loadable blocks.

Continued on next page

Programming Through MODSOFT, Continued

Step	Action
4	Within the directory window, press <F1> (load), enter the appropriate file name (\Modsoft\loadable\mspx.dat for version 2.51 and \Modsoft\programs\mspx.dat for versions 2.6 and later), press <enter> one time to load the block, then press <enter> again to view the updated list of loadable blocks

The MSPX block's instructions are now available for use within a MODSOFT program.

Using the MODSOFT Custom-Loadable Block

Allocate input and output registers to the MOMENTUM-SERIPLEX base unit before using the MSPX custom-loadable block. While at the I/O map screen in the MODSOFT software, press <shift> and <?> to view the list of available MOMENTUM I/O base units. Within the list, select the MOMENTUM-SERIPLEX interface base unit (170ANM05010). While the interface base unit is highlighted in the list, you can view a help screen by pressing <alt> and <h>.

Once the interface base unit has been selected, assign to it 32 input registers (from the 3xxxx series) and 32 output registers (from the 4xxxx series). Then be sure to reserve the next output register in the processor's memory, since the MSPX block actually uses 33 output registers (the base unit's 32 registers, plus one more for working storage).

Once a minimum of one interface base unit has been mapped within the processor, an instance of the block can be placed within the control program.

Step	Action
1	In the ladder segment programming screen, press <F3> (elements) to select different types of program elements.
2	Press <F5> (loadable) to view the list of available, loadable blocks.
3	Select the MSPX block.

When placed within the ladder diagram, the block appears similar to Figure 5. on page 43.

Continued on next page

Programming Through MODSOFT, Continued

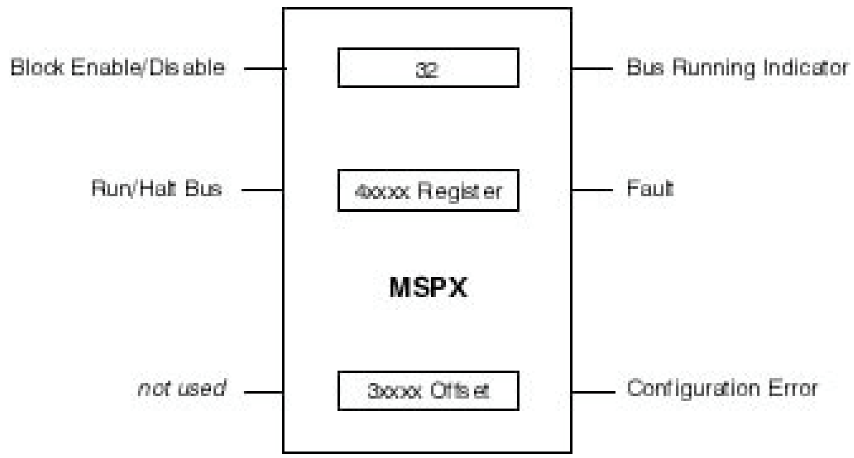


Figure 5. MSPX Block Ladder Logic Diagram

Using the MODSOFT Custom-Loadable Block, Continued

To view the help screen, place the cursor over the block, then press <alt> and <h>. The help screen describes the block's inputs, outputs, and node values.

Once the block is placed within the ladder diagram, the general procedure for using the MSPX block is:

Step	Action
1	Assign node values to the block (see the descriptions of inputs, outputs, and node values described on the following paragraphs).
2	Connect control logic to the block's inputs to enable operation of the block and start/ stop the SERIPLEX bus.
3	Connect logic to the block's outputs so the software can take action when it reads the status of the interface base unit
4	Use the block's associated zoom screen to assign values for the interface base unit's operating parameters, as appropriate for the application. While the ladder program is running, the zoom screen can also be used to monitor the status of the interface base unit and the SERIPLEX bus.

Continued on next page

Programming Through MODSOFT, Continued

MSPX Block Inputs

The MSPX block has three input signals, described below.

Block Enable/ Disable

This input enables and disables operation of the MSPX block. When the connected logic is true, the block's instructions are executed within the ladder program. The SERIPLEX bus can then be started or stopped, and the block's outputs report their status conditions to the interface base unit. When this input is false, the block's instructions are not executed.

Enabling or disabling the block does not affect the values of any of the base unit's input or output registers, including the control register. Therefore:

- The values of the base unit's operating parameters, and their activation, are not affected by the block enable/disable input.
- The base unit will not receive a command to start or stop the bus while the block is disabled. If the bus is running when the block is disabled, the bus will continue to run. If the block is re-enabled, the run/stop bus input determines whether the bus will run.

Disabling the block enable/disable input leaves the block's outputs in their current states. Actual changes in the interface base unit's status are not reported by the block.

Run/Stop Bus

This input controls the operation of the SERIPLEX bus through the *run/halt* bit within the base unit's control register. When the connected logic is true, the *run/halt* bit is set to 1. When the logic is false, the *run/halt* bit is cleared to 0. The bus will run while the run/stop input is true, unless a bus fault is present.

The operating parameters of the interface base unit become effective when the run/stop bus input becomes true. Parameters cannot be changed while the run/stop input is enabled. Any attempt to change the parameters while the *run/halt* bit is set will result in a configuration fault.

Continued on next page

Programming Through MODSOFT, Continued

Not Used

This input has no effect on the operation of the MSPX block. The block ignores any logic connections and the state of the interface base unit is not affected.

MSPX Block Outputs

The MSPX block has 3 output signals, described below.

Bus Running Indicator

This output reports whether the Seriplex bus is running, that is, whether the interface base unit is transmitting the SERIPLEX bus clock signal. The output directly reflects the state of the *bus running* bit in the base unit's status register. The output is true when the *bus running* bit is on (1), and false when the bit is off (0).

When the bus-running indicator output is true and the block is enabled, the SERIPLEX bus is operating normally and bus input and output data is being exchanged with the PLC.

Fault

This output reports whether the base unit is experiencing a fault condition other than a configuration fault. The output is true when any of the following bits in the base unit's status register are on (1):

Bus fault (bit 3), MOMENTUM fault (bit 4), CDR error (bit 5)

The interface base unit halts the bus under any of these fault conditions.
The exact nature of the detected fault can be determined by reading the base unit's status register, as explained within the status code description in the section "Memory Map" on page 18. A fault can also be identified by examining the MSPX block's zoom screen.

Continued on next page

Programming Through MODSOFT, Continued

Config Error

This output indicates that the interface base unit has detected a configuration error. The output directly reflects the state of the *config fault* bit in the base unit's status register. The output is true when the *config fault* bit is on (1), and false when the bit is off (0). When true, the output indicates that an improper attempt was made to write to the base unit's control register (see "Memory Map" on page 18). The base unit halts the bus under a config fault condition.

MSPX Block Node Values

The MSPX block has 3 node values, described below.

Top Node

This node value represents the internal sub-function number of the block. *This node must be assigned the value 32*, so the ladder program can identify the correct MSPX block logic.

This node represents the internal sub-function number. The top node value can either be assigned the decimal constant 32 directly, or be assigned a 4xxxx-series register containing the decimal value 32. In the latter case, a change to the 4xxxx register's value prevents the MSPX block from executing.

Middle Node

This node value represents the starting 4xxxx-series register location for the SERIPLEX-MOMENTUM interface base unit. This is the starting location of the base unit's output registers, therefore representing the location of the base unit's control register.
The MSPX block uses 33 output registers, starting at the middle node value. The MSPX block places a copy of the current, basic SERIPLEX status register in the last 4xxxx register (middle node value + 32), so the individual bits of the status register can be interpreted for use by the zoom screen.

Continued on next page

Programming Through MODSOFT, Continued

Bottom Node

The MSPX block interprets this node value as a numeric offset from 300000, to indicate the first 3xxxx-series input register assigned to the interface base unit. The value indicates the location of the base unit's status register.

Note: This value is an offset, with a value between 1 and 9999, and not the 3xxxx-series register number itself. For example, if the starting register address is 300033, the bottom node value would be 33.

Setting Interface Parameters

The interface base unit's operating parameters can be set through the MSPX block's DX zoom screen. Access the DX zoom screen by pressing <ctrl> and <z> while the cursor is on the MSPX block.

You must select parameters before commanding the base unit to run the SERIPLEX bus. Typically, parameters are selected when the ladder program is commissioned. Once the parameters are selected, they are stored along with the ladder program and are used by the interface base unit until they are changed.

The zoom screen has two pages: one for selecting parameter values and one for viewing the status of the base unit. The status page is described later in this manual. Select parameters on the first page, shown in Figure 6, below.

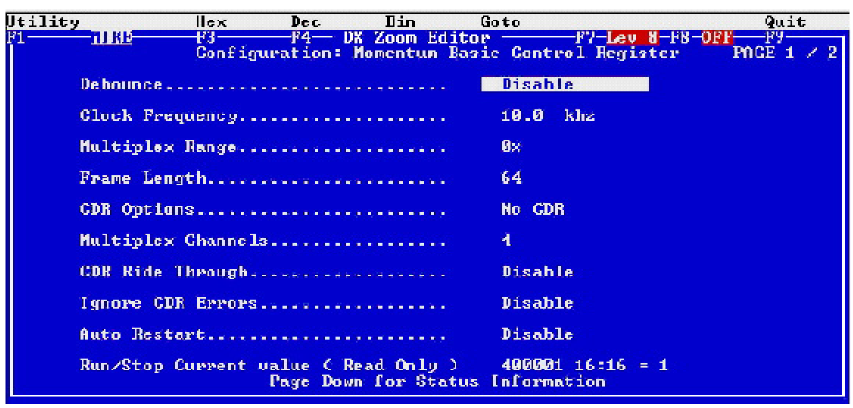


Figure 6. Zoom Screen Parameter Selection

Continued on next page

Programming Through MODSOFT, Continued

Setting Interface Parameters, Continued

The zoom screen shows the parameters and their current values. To select a parameter for editing, use the arrow keys until the value of the desired parameter is highlighted. Press <enter> to edit the value.

If the selected parameter has only two possible values, a small window will appear in the screen, as shown in Figure 7. on page 48.

Figure 7. Pop-up Menu

Use the arrow keys to select your choice, then press <enter> to accept the value and close the pop-up window.

If the selected parameter has several possible values, a larger pop-up window appears. The window displays all possible values for the selected parameter, as shown in Figure 8. on page 49.

Continued on next page

Programming Through MODSOFT, Continued

Setting Interface Parameters, Continued

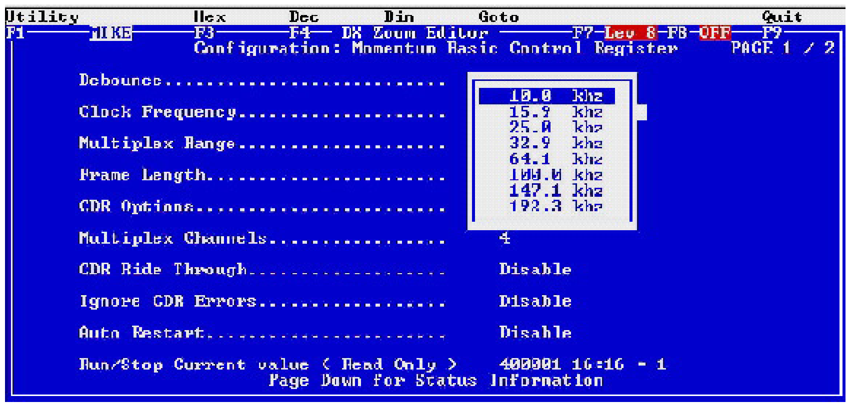


Figure 8. Multiple Choice Pop-up Menu

Again, use the arrow keys to select the desired value, then press <enter> to accept the value and close the pop-up window.

Check that all parameters have the correct values selected before closing the zoom screen. The zoom screen does not supply any default values; instead, values are directly read from and written to the base unit's control register. When you are finished setting values, press the <esc> key to exit the zoom screen.

Note: Do not change parameter values while the bus is running. If you attempt to change parameters, the base unit declares a configuration fault and halts the bus.

Viewing Interface Base Unit Status

The second page of the MSPX block's DX zoom screen displays the interface base unit's status. The display is based on a copy of the base unit's status register, stored in the thirty-third 4xxxx-series output register assigned to the base unit (middle node value + 32). The status screen displays the base unit's current status only when the MSPX block is enabled and executing.

Continued on next page

Programming Through MODSOFT, Continued

Viewing Interface Base Unit Status, Continued

To view the status page, first view the zoom screen for the MSPX block by positioning the cursor on the image of the block and pressing <ctrl> and <z>. Press the <page down> key to view the status page, as shown in Figure 9, below.

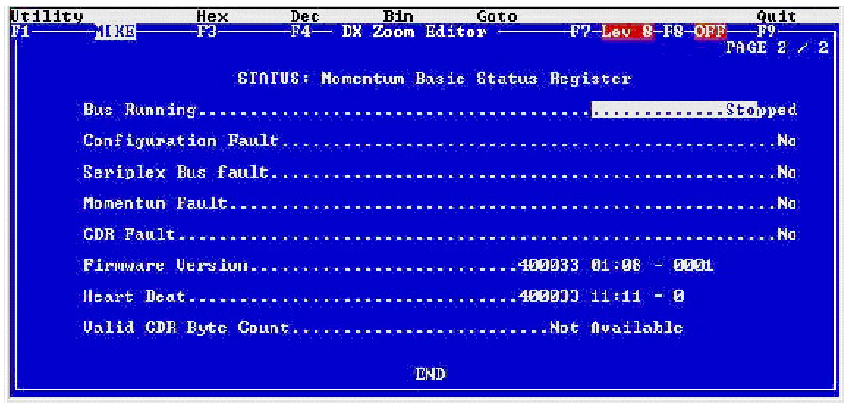


Figure 9. DX Zoom Screen, Status Page

If the SERIPLEX bus is halted and no faults are present, the screen displays the base unit's firmware version number. If the bus is running, the screen displays the valid CDR byte count (the number of bus input data bytes that have CDR enabled and were received with a valid check byte). If there is a fault, the "Valid CDR Byte Count" displays "Not Available", as shown in Figure 10. on page 51.

Continued on next page

Programming Through MODSOFT, Continued

Viewing Interface Base Unit Status, Continued

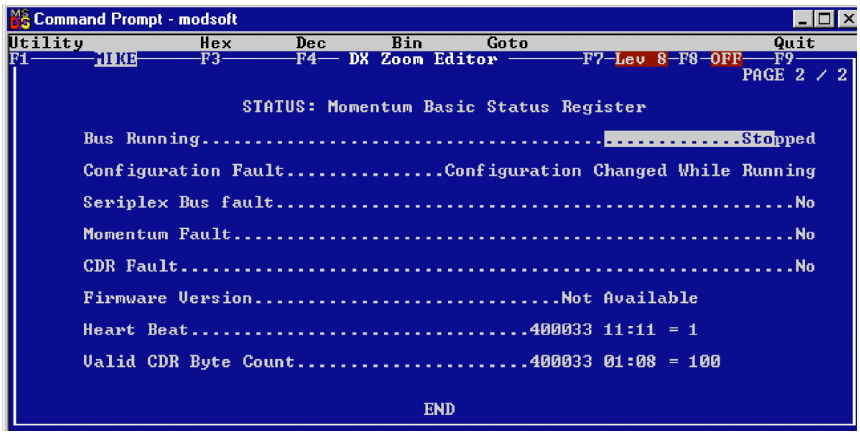


Figure 10. Fault Status

To return to the parameter-selection page (page 1 of the zoom screen), press the <page up> key. To exit the zoom screen, press the <esc> key.

Programming Without The MSPX Custom-Loadable Block

The MOMENTUM-SERIPLEX base unit can be used with MODSOFT software without using the MSPX custom-loadable block. In fact, the MSPX block cannot be used when using any MODICON processor other than a MOMENTUM M1 processor.

To use the interface base unit without the MSPX block:

When using MODSOFT software versions 2.51 and 2.60, copy the file m1tcop.sys from the CD-ROM to your hard drive, as described in “Installing the MODSOFT Custom-Loadable Block” on page 40. Skip this step if you are using MODSOFT software version 2.61 or later.

Continued on next page

Programming Through Concept

Programming Without The MSPX Custom-Loadable Block, Continued

Assign 32 input registers and 32 output registers to the base unit, as described in the section "Using the MODSOFT Custom-Loadable Block" on page 42. You do not need to reserve a thirty-third register within the processor if you are not using the MSPX block.

In the ladder program, write the desired parameter settings directly to bits within the base unit's control register and read the base unit's status register directly to detect fault conditions.

Programming Through CONCEPT PLC Software

The MOMENTUM-SERIPLEX interface base unit provides a derived function block (DFB), named MSPX, for use in CONCEPT™ PLC programming software. This block can be incorporated into a MOMENTUM M1 processor's IEC ladder logic program for easier configuration and monitoring of the interface base unit.

The base unit can also be used with CONCEPT software without using the MSPX block by directly reading and writing bits within the base unit's registers.

Installing the CONCEPT DFB

An installation program for the CONCEPT DFB is stored on the CD-ROM disk provided with the SERIPLEX-MOMENTUM interface base unit. The installation program is found at D:\Concept\English\setup.exe, where D: is the name of the CD-ROM drive. Run this file from the Windows Start menu, or use a file-management tool to locate and execute the file.

In addition to the DFB files, the installation program also installs the SERIPLEX Calculator program. The Calculator is used to help define base unit parameter values, as described in the section "Using the SERIPLEX Calculator Program" on page 61.

Continued on next page

Programming Through Concept, Continued

Installing the CONCEPT DFB, Continued

The installation program places the following files into the \Concept\dfb directory on your computer hard disk:

mspx.dfb

mspx.dsk

mspx.p1

mspx.p2

mspx.p3

mspx.p4

mspx.q1

mspx.q2

The installation program also places a shortcut to the SERIPLEX Calculator program on the Windows desktop.

Installing the Interface Base Unit into the CONCEPT Program

To make the interface base unit available within the CONCEPT software, install the unit's MDC file through the CONCEPT ModConnect tool (Modconf.exe).
To do this:

Step	Action
1	Open the Modconnect tool by clicking on its icon, found in the CONCEPT program directory.
2	Select "Add Module" from the Module menu.
3	Select and open the file \Concept\Seriplex\Seriplex.mdc.
4	Select the MOMENTUM-SERIPLEX interface base unit from the list in the dialog box that appears, then click on the Add Module button.
5	Close the dialog box. The base unit should now be installed and available for use within the CONCEPT software.
6	Once installed into the CONCEPT software, assign 32 input (3xxxx) and 32 output (4xxxx) registers to the base unit.

Continued on next page

Programming Through Concept, Continued

Using the CONCEPT DFB

To place an instance of the MSPX block into a program:

Step	Action
1	Within a CONCEPT project, select New Section from the File menu.
2	Select the FBD editor type using the radio buttons on the screen.
3	Select the FFB icon on the toolbar, or select FFB Selection from the Objects menu.
4	In the pop-up window that appears, click on the DFB button.
5	For the DFB type, select MSPX.
6	Place the function block at the desired location in the program window.

The appearance of the DFB is similar to the diagram in Figure 11, below.

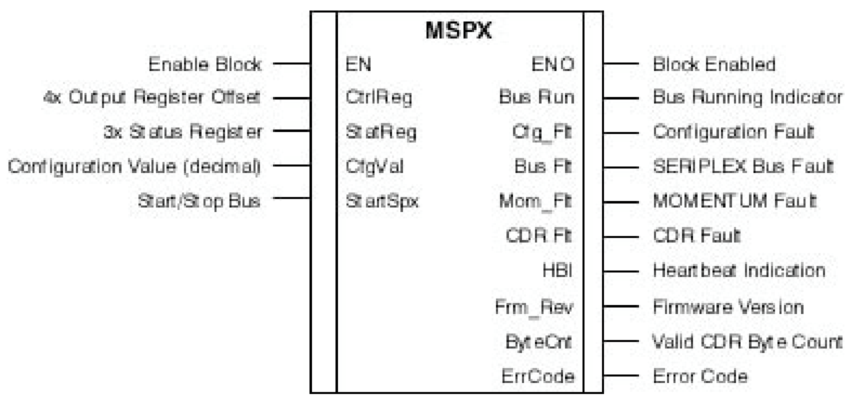


Figure 11. DFB Block

Once the block is placed within the program, connect control logic to the block's inputs to identify the assigned registers, set operating parameters, and start/stop the SERIPLEX bus. Connect logic to the block's outputs to take action based on the status of the interface base unit.
The MSPX block has four input signals and ten output signals, described below.

Continued on next page

Programming Through Concept, Continued

DFB Inputs

The MSPX DFB has 5 input signals, described below.

EN (Enable Block)

This input enables and disables the operation of the MSPX block. When the connected logic is true, the block's instructions are executed by the control program. The SERIPLEX bus can then be started or halted, and the block's outputs will report their status conditions. When the EN input is false, the block's instructions are not executed.

Enabling or disabling the block does not affect the values of any of the interface base unit's input or output registers, including the control register. Therefore:

- The values of the base unit's operating parameters, and their activation, are not affected by the EN input.
- The base unit will not receive a command to start or stop the SERIPLEX bus while the block is disabled. If the bus is running when the block is disabled, the bus will continue to run. If the block is re-enabled, the *StartSpx* input determines whether the bus will run.

Disabling the EN input leaves the block's outputs in their current states. The block will not report actual changes in the base unit's status.

Note: You can choose whether or not to display the EN input and ENO output on the block. When these items are not displayed, the block is continuously enabled.

CtrlReg (Output Register Offset)

This input designates the register address of the interface base unit's control register, which is the first output register (4xxxx series) assigned to the base unit. The value is a numeric offset from the start of the 4xxxx holding registers (400000). For example, if the control register address is 400033, then CtrlReg = 33. The value must correspond to the register assignment within the processor.

Continued on next page

Programming Through Concept, Continued

StatReg (Status Register)

This input designates the register address of the status register in the interface base unit, which is the first input register (3xxxx series) assigned to the base unit. The value of the input must be a 3xxxx register address (such as %3:00001), not a numeric offset. The value must correspond to the register assignment in the processor.

CfgVal (Configuration Value)

This input represents the value to be written to the base unit's control register. This value designates all settings for the control register, except for the *run/halt* bit, which is controlled by the *StartSpx* input.

The input value is a decimal number with an even value between 2 and 32766, inclusive. The SERIPLEX Calculator program can be used to determine this value, as described later in this manual. However, there is no direct means to transfer the Calculator's value to the DFB input.

StartSpx (Start/ Stop Bus)

This input controls the operation of the SERIPLEX bus via the *run/ halt* bit in the interface base unit's control register. When the connected logic is true, the *run/halt* bit is set to 1; when the logic is false, the *run/halt* bit is cleared to 0. The bus runs while this input is true, unless a bus fault is present.

The operating parameters for the interface base unit become effective when the *StartSpx* input becomes true. Do not change parameters while the input is enabled. Any attempt to change the parameters while the *run/halt* bit is set and the *StartSpx* input is true will result in a configuration fault.

Continued on next page

Programming Through Concept, Continued

Derived Function Block (DFB) Outputs

The MSPX DFB has 10 output signals, described below.

ENO (Block Enabled)

This output indicates that the DFB is enabled. The output directly reflects the value of the block's EN input. By itself, the output does not indicate any information about the state of the interface base unit.

Bus_Run (Bus Running Indicator)

This output reports whether the bus is running, which indicates whether the interface base unit is transmitting the bus clock signal. The output directly reflects the state of the *bus running* bit in the base unit's status register. The output is true when the *bus running* bit is on (1) and false when the bit is off (0).

When the *bus_run* output is true and the DFB is enabled, the SERIPLEX bus is operating normally and bus input and output data is being exchanged with the PLC.

Cfg_Fault (Configuration Fault)

This output indicates that the interface has detected a configuration fault. The output directly reflects the state of the *configuration fault* bit in the base unit's status register. The output is true when the *configuration fault* bit is on (1) and false when the bit is off (0).

When true, the *cfg_fault* output indicates that an improper attempt has been made to write to the base unit's control register. See the section "Memory Map" on page 18 for details.

Continued on next page

Programming Through Concept, Continued

Bus_Fault (SERIPLEX Bus Fault)

This output indicates that the interface base unit has detected a bus fault. The output directly reflects the state of the *bus fault* bit in the base unit's status register. The output is true when the *bus fault* bit is on (1) and false when the bit is off (0). The particular type of detected bus fault is indicated in the *ErrCode* output.

Mom_Fault (MOMENTUM Fault)

This output indicates that the interface base unit has detected a problem in the communications between the base unit and its host adapter. The output directly reflects the state of the *MOMENTUM fault* bit in the base unit's status register. The output is true when the *MOMENTUM fault* bit is on (1) and false when the bit is off (0). The particular type of detected MOMENTUM fault is indicated in the *ErrCode* output.

Since the base unit is not able to report MOMENTUM errors through its host adapter, the *mom_fault* output is true only in cases where the host adapter has re-established communications to the base unit, without a power cycle.

CDR_Fault (CDR Fault)

This output indicates that the base unit has detected either an input or output CDR fault. The output directly reflects the state of the *CDR fault* bit in the base unit's status register. The output is true when the *CDR fault* bit is on (1) and false when the bit is off (0). The SERIPLEX bus address that experienced the CDR fault(s) is indicated in the *ErrCode* output.

HBI (Heartbeat Indication)

This output toggles between true and false at a rate between one and two times per second, indicating that the base unit is operating. The HBI output directly reflects the state of the *heartbeat* bit in the base unit's status register. The output is true when the *heartbeat* bit is on (1) and false when the bit is off (0).

Continued on next page

Programming Through Concept, Continued

Frm_Rev (Firmware Version)

This output reports the interface base unit's firmware version. The version is given as a two-digit BCD number, as described in the section "Bits 9 – 16: Status Code" on page 29.

The output only reports the firmware version when the SERIPLEX bus is not running and the base unit is free of fault conditions. If the bus is running or a fault is present, the reported firmware value is 0.

ByteCnt (Valid CDR Byte Count)

This output reports the number of bytes of CDR-enabled input data for which the interface base unit detected valid CDR check codes. The host software can use this number to verify that the proper number of bus input devices are connected and communicating. See "Bits 9 – 16: Status Code" on page 29 for information about how the byte count is calculated.

The interface base unit reports the valid CDR byte count only when the bus is running and no fault conditions are present. If the bus is halted or a fault is present, the reported byte count value is 0.

ErrCode (Error Code)

When the interface base unit detects a fault, this output reports the associated error code value. The value is obtained directly from the base unit's status register. Possible error code values are listed under "Bits 9 – 16: Status Code" on page 29.

The DFB retains the error code value after the StartSpx input clears the base unit's *run/halt* bit to 0. However, when the SERIPLEX bus is restarted — either through a command like the StartSpx input or through the base unit's auto restart feature — the error code value clears to 0.

Continued on next page

Programming Through Concept, Continued

Viewing and Editing the DFB Code

The MSPX block is constructed from normal CONCEPT program elements. The underlying code can be viewed and edited, if desired. To view the DFB's underlying code, double-click on the MSPX DFB diagram to bring up the Properties dialog box, then click on the <refine> button within the box. The code will be displayed, but cannot be edited.

Before editing the underlying code in the DFB, save a copy of the original DFB code by copying and renaming all of the files listed on page 52. For instance, rename the file \Concept\dfb\mspx.dfb to mspxdfb.bak (making the original a back-up file). Run the CONCEPT DFB editor and open the file \Concept\dfb\mspx.dfb to edit the code. To recover the original code, rename mspxdfb.bak to mspx.dfb, and so on for the other files.

Programming Without the DFB

The MOMENTUM-SERIPLEX interface base unit can be used with the CONCEPT software without using the MSPX derived function block. In fact, the MSPX block cannot be used with any MODICON processor other than a MOMENTUM M1 processor.

To use the interface base unit without the MSPX DFB:

Step	Action
1	Use the ModConnect tool to install the interface base unit into the CONCEPT software, as described on page 53.
2	Assign 32 input (3xxxx) registers and 32 output (4xxxx) registers to the base unit.
3	In the CONCEPT software, write the desired parameter settings directly to the bits within the base unit's control register, and read the base unit's status register directly to detect faults.

Continued on next page

Programming Through Concept, Continued

Using the SERIPLEX Calculator Program

The MOMENTUM-SERIPLEX interface base unit is packaged with the SERIPLEX Calculator program, which performs the following tasks:

- Calculates values to be written to the base unit's control register, based on the user's selection of desired parameter settings
- Shows the base unit's register memory map, which stores the values of the control register
- Saves parameter settings and the register map as a text file, which can be imported into other programs

The Calculator can be used with Microsoft® Windows™ 95, 98, or NT® operating systems. The program is installed with the CONCEPT derived function block. Once installed, an icon labeled SPXCalc appears on the Windows desktop.

To start the Calculator program, double-click on the icon on the desktop. The Calculator main window is shown in Figure 12. on page 62.

Continued on next page

Programming Through Concept, Continued

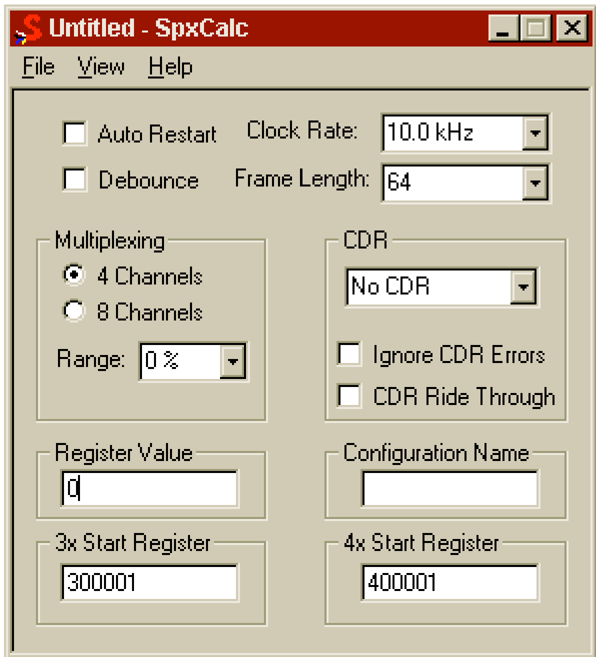


Figure 12. SERIPLEX Calculator

Using the SERIPLEX Calculator Program, continued

Select the desired parameter values for the interface base unit by using the Calculator's check boxes, radio buttons, and drop-down lists. For documentation purposes, you may also enter starting register addresses and a name for the configuration in the provided text boxes.

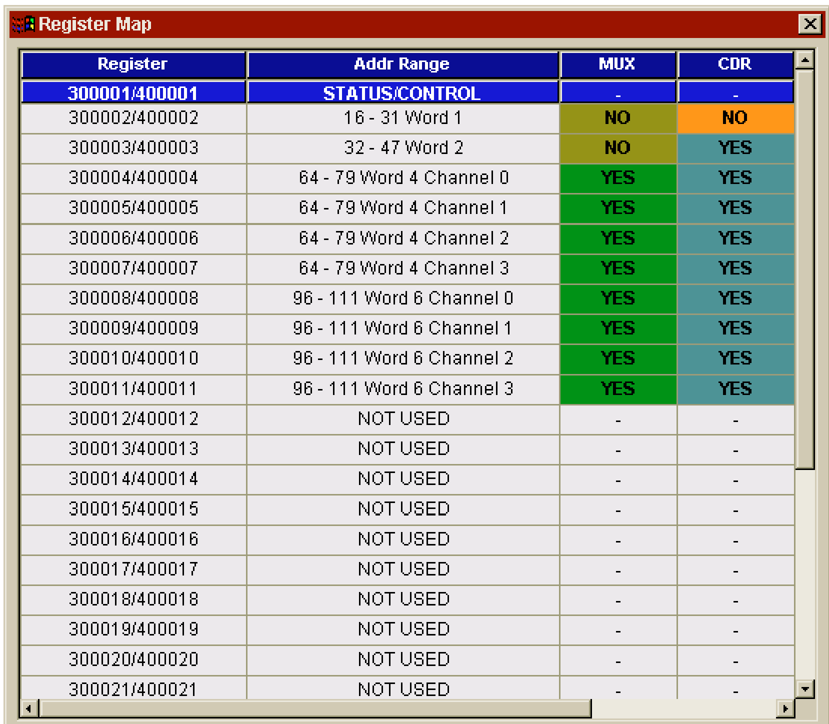
The Calculator calculates a value for the base unit's control register, based on the selected parameter values, and displays it in the text box labeled "register value". Alternately, you can enter a register value directly in the text box; the Calculator then determines and displays the parameter values, based on the register value. The program accepts the register value and calculates parameter values once you type in the value and press the <enter> key.

Continued on next page

Programming Through Concept, Continued

Using the SERIPLEX Calculator Program, continued

Once you select all desired parameters, view the associated register map by selecting Register Map in the View menu. The map will display in a second window, as shown in Figure 13, below.



Register	Addr Range	MUX	CDR
300001/400001	STATUS/CONTROL	-	-
300002/400002	16 - 31 Word 1	NO	NO
300003/400003	32 - 47 Word 2	NO	YES
300004/400004	64 - 79 Word 4 Channel 0	YES	YES
300005/400005	64 - 79 Word 4 Channel 1	YES	YES
300006/400006	64 - 79 Word 4 Channel 2	YES	YES
300007/400007	64 - 79 Word 4 Channel 3	YES	YES
300008/400008	96 - 111 Word 6 Channel 0	YES	YES
300009/400009	96 - 111 Word 6 Channel 1	YES	YES
300010/400010	96 - 111 Word 6 Channel 2	YES	YES
300011/400011	96 - 111 Word 6 Channel 3	YES	YES
300012/400012	NOT USED	-	-
300013/400013	NOT USED	-	-
300014/400014	NOT USED	-	-
300015/400015	NOT USED	-	-
300016/400016	NOT USED	-	-
300017/400017	NOT USED	-	-
300018/400018	NOT USED	-	-
300019/400019	NOT USED	-	-
300020/400020	NOT USED	-	-
300021/400021	NOT USED	-	-

Figure 13. View of Register Map

The register map shows the SERIPLEX bus addresses and multiplex channel numbers associated with each of the base unit's 64 input and output registers. In addition, the map shows which registers have address multiplexing and/or CDR data validation enabled. To close the register map window and return to the main window, click on the <X> button in the upper right corner of the window.

Continued on next page

Programming Through Concept, Continued

Using the SERIPLEX Calculator Program, continued

To print the configuration data and the register map, select Print from the File menu in the main window. To save the configuration data and register map as a text file, choose Save from the File menu, then select a name and directory location for the file. By default, the file saves with the extension ".txt". The text file can be used for documentation purposes or imported into another program.

The SERIPLEX Calculator cannot open a file. However, all parameter and register map information can be recovered by entering the register value into the calculator. See Figure 12. on page 62.

The control register value produced by the Calculator does not include the *run/halt* bit. The calculated value clears the *run/halt* bit to 0, to provide the proper format for the *CtrlReg* input of the MSPX block. When writing directly to the base unit's control register, add a value of 1 to the Calculator's register value to activate the selected parameters and start the bus.

Using the Communications Adapter

Using Communication Adapters

A MOMENTUM communications adapter can be used as a host adapter, instead of an M1 processor adapter, for the MOMENTUM-SERIPLEX interface base unit. The base unit still appears as 32 input registers and 32 output registers to the host device. The operation of the interface base unit and its data registers remains the same.

In almost all cases, you cannot use either the base unit's MODSOFT custom-loadable block or its CONCEPT derived function block, since the host adapter is not an M1 processor. The only exception is when the base unit is used as a remote I/O drop to a MOMENTUM M1 processor through an IOBUS network.

With some types of control networks, using the interface base unit requires a unique profile file, or some other type of special consideration. See the following five sections covering considerations for the use of the base unit with several types of MOMENTUM communication adapters.

MODBUS PLUS: 170PNT11020, 170PNT16020, 170NEF11021, or 170NEF16021

To use the interface base unit with any TSX MODICON™ processor through a MODBUS PLUS™ communications adapter, peer cop (cross-reference) the base unit's registers within the master. Use an adapter with the proper data format (IEC, 984) to avoid mixing the order of the SERIPLEX data bits within the base unit's data registers.

I/OBUS, INTERBUS-S: 170 INT 110 00

To use the interface base unit with a TSX MODICON processor through IOBUS™ or INTERBUS-S™ networks using CONCEPT programming software, use the Modconnect tool to install the file Seriplex.mdc, found on the CD-ROM provided with the base unit. For directions on using the ModConnect tool, see "Installing the Interface Base Unit into the CONCEPT Program" on page 53. Once the file is installed, the base unit becomes available as a remote I/O drop for the host processor.

To use the base unit with a TSX MODICON processor using MODSOFT programming software, first follow the instructions found in the section "Installing the MODSOFT Custom-Loadable Block" on page 40. Next, use the MODSOFT software to perform an add/drop procedure, choosing the interface base unit from the list of available modules.

Continued on next page

Using the Communications Adapter, Continued

DEVICENET: 170LNT71000

To use the interface base unit with any host controller through a DEVICENET™ communications adapter, configure the master device to view the base unit as 32 input words and 32 output words of DEVICENET I/O data, which may require the use of a profile generator for the master device. An electronic data sheet (.eds) is provided on the CD ROM which comes with the base unit.

ETHERNET: 170ENT11000

When using the interface base unit with a TSX MODICON processor through an ethernet communications adapter, you cannot peer cop the base unit's data registers. Use MSTR instructions to read and write the base unit's input and output registers. Configure the master device to view the base unit as 32 input words and 32 output words of data.

PROFIBUS DP: 170DNT11000

To use the interface base unit with a host controller using a PROFIBUS DP™ communications adapter, configure the host to view the base unit as 32 input words and 32 output words of PROFIBUS data. You may need to generate a profile for use by the master device.

The host's .gsd file may need to be modified to recognize the interface base unit. Refer to the CD ROM file D:\Comm Adapter Support\Profibus\Readme.txt for further information.

Operating Characteristics

The following sections describe the behavior of the interface base unit during start-up, normal operation, and fault conditions.

Initialization

When released from any reset condition, including initial application of power, the interface base unit performs the sequence of actions listed below. The entire initialization procedure is typically accomplished in approximately two seconds.

Step	Action
1	To begin, the base unit's MOMENTUM interface circuitry is disabled, preventing the unit from communicating with its processor or communications adapter. The *interface fault* light is illuminated. The *fault number* light is continuously lit, showing that the base unit has not yet successfully completed its internal tests.
2	After performing some internal initialization and self-testing, the interface base unit enables its MOMENTUM interface circuitry. The unit can now perform read and write operations between the host and the base unit's register memory. At this point, the base unit reports its status through the status register. The *interface fault* and *fault number* lights remain lit.
3	If the internal self-tests are not completed successfully, the *interface fault* and *fault number* lights remain lit. The interface base unit will not accept or process software commands. Remove and re-apply power to make the base unit attempt re-initialization. In some cases, the interface base unit is able to report the nature of the internal fault through the status register. When internal self-tests are successfully completed, the interface base unit turns off the *interface fault* and *fault number* lights, then illuminates the *interface-OK* light. The base unit is ready to accept and process software commands through the control register and will accept base unit and bus operating parameters.
4	The Seriplex bus remains halted until the base unit receives a command to start the bus.

SERIPLEX Bus Initialization

The interface base unit typically starts the SERIPLEX bus when the unit receives a software command. However, the base unit can also start the bus with the auto restart feature, after detecting a bus fault.

The interface base unit performs the following sequence of actions when starting the SERIPLEX bus:

Continued on next page

Operating Characteristics, Continued

- All bus operating parameters in the base unit's control register become effective when the base unit receives the command to start the bus. The parameters include bus frame length, clock rate, address multiplexing selections, CDR selections, the digital debounce selection, and the auto restart selection.
- All bus input data in the base unit's input registers is cleared to zero.
- The base unit tests the bus for the presence of faults. If a bus fault is found, the clock signal is not transmitted, and the unit does not start the bus; the *bus run* indicator will not light.
  Once the bus fault tests are completed successfully, the base unit starts transmitting the clock signal at the rate selected during configuration, and illuminates the *bus run* light.
- The base unit updates each bus input signal in its memory as soon as digital signal debounce is satisfied for that input. Inputs that do not have digital signal debounce enabled are updated at the end of the data frame. However, most bus input devices report signal values of 0 for at least four data frames after the start of the clock signal. Therefore, actual input signal states might not be available to the host for a short period after the bus has started.
- The base unit transmits values of 0 for all bus outputs, until bus initialization is complete.
- The base unit tests for valid CDR check bytes, as described in *Multi-Bit Data Validation with CDR* on page 13.

Continued on next page

Operating Characteristics, Continued

Normal Operation

During normal bus operation:

- The base unit reads bus input data from the bus and reports the data to the input data registers for use by the host.
- The base unit continually evaluates digital signal debounce for each input signal that has debounce enabled. The base unit updates the input register when debounce conditions are satisfied for a particular input.
- The base unit continually evaluates CDR for input signals that have the feature enabled. The unit reports valid data to the input registers and rejects invalid data. Input CDR faults are detected and reported according to the selections made during configuration. The base unit updates input data in the input registers at the end of the data frame in which the data is received, following data validation and bus fault checks.
- The base unit reads bus output data from the output data registers, and transmits the data through the bus as the appropriate bus output signals. The unit continuously transmits CDR check values for each bus output signal that has CDR enabled. Output CDR faults are detected and reported according to the selections made during parameter set-up.
- The base unit tests the bus for faults at the end of each data frame. If it detects a fault, the base unit halts the bus and responds as described in the section “Fault Responses” on page 73.
- The base unit reports internal status and bus operating status to the host software through the status register. The unit also reports operating status visually with the indicator lights.
- The base unit broadcasts the current multiplex channel information to the bus, according to selections made during parameter set-up.
- While the bus is operating normally, the base unit does not respond to any software commands except a bus-halt command. If an attempt is made to change the base unit’s operating parameters while the bus is running, the base unit declares a configuration fault and stops the bus.
- Normal bus operation begins approximately 0.75 seconds after the clock signal starts transmitting. This time period allows for proper initialization and settling of all bus devices, including those that use CDR. After the delay, the interface base unit transmits bus output data normally from its output registers.

Continued on next page

Operating Characteristics, Continued

Signal Response Time

The latest edition of *SERIPLEX Design, Installation and Troubleshooting* (bulletin number 30298-035-01A) presents instructions for calculating SERIPLEX signal response times. The following table shows quantities to be used in these calculations for a system that incorporates a MOMENTUM-SERIPLEX interface base unit.

Table 16: Response Time Parameters

Parameter	Symbol	Value	Unit
Sync length	sync	8 when clock rate is <= 100 kHz 16 when clock rate is >100 kHz	clock cycles
Host input propagation delay	t_{hip}	32	clock cycles
Host output propagation delay	t_{hop}	32	clock cycles

Continued on next page

Operating Characteristics, Continued

Indicator Lights

The interface base unit provides six indicator lights to give users direct visual indication of the status of the base unit and the bus. These lights are visible from the front of the base unit. See Figure 14, below.

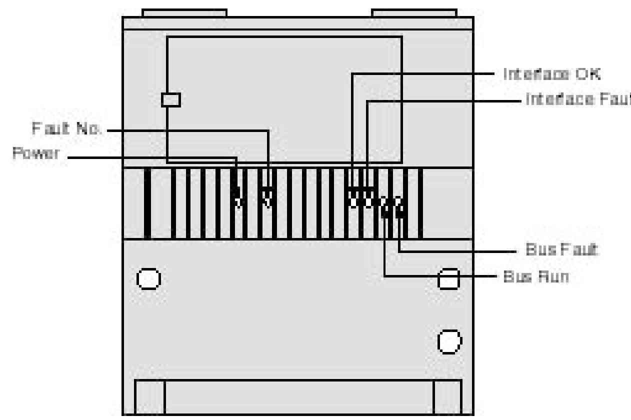


Figure 14. LED Indicators

LED Indicators

Table 17 describes the appearance and meaning of each light.

Table 17: Base Unit Indicator Lights

Name	Color	Description
Interface-OK	green	The base unit is receiving power and has passed its internal self-tests.
Interface fault	red	The base unit is receiving power and has detected an internal fault.
Bus run	green	The base unit is actively transmitting the clock signal.
Bus fault	red	The base unit has detected a bus fault.
Fault number	red	The light flashes a number of times to indicate specific types of base unit and bus faults.
Power	green	The base unit is receiving enough power to operate.

Continued on next page

Operating Characteristics, Continued

Fault Number Light

Table 18 lists the specific faults identified by the number of flashes of the *fault number* light. The *fault number* light only indicates one type of fault at a time. Faults are prioritized in the order given in the table, with the first fault indicated in the event of multiple, simultaneous faults.

Table 18: Fault Type Indication

# of Flashes	Fault Type	Source
0 (continuously lit)	Internal CPU or program fault	Interface base unit
1	Internal RAM fault	
2	External RAM fault	
3	FPGA load fault	
4	Interface personality fault	
5	Internal watchdog fault	
6	Data flow fault	
9	Other internal fault	
0 (continuously lit)	Bus undervoltage	SERIPLEX bus
1	Data line stuck low	
2	Data line stuck high	
3	Clock fault	
4	Data line undercurrent	
5	Data line overcurrent	
6	Excess data line capacitance	
7	Input CDR fault	
8	Output CDR fault	

Continued on next page

Operating Characteristics, Continued

Fault Number Light, Continued

The light flashes at a rate of once every 0.4 seconds until the flash sequence is completed, then remains unlit for three seconds before repeating the flash sequence.

If auto restart is selected, the fault type flash sequence is not valid. The fault number light flashes continuously, because the flash sequence is interrupted and restarted every 0.25 seconds as the base unit attempts to restart the bus.

Fault Responses

The following sections describe the base unit's response to internal faults and bus faults.

Interface Base Unit Faults

The base unit tests for internal faults during the initial power-up sequence. Therefore, most internal faults are detected before the base unit begins communicating with either the MOMENTUM host or the SERIPLEX bus. When a fault is detected, the base unit activates the *interface fault* and *fault number* lights and will not accept or process software commands. Remove power from the unit and re-apply the power to make the unit attempt re-initialization. In general, the base unit does not notify the host of an error, although in some cases the base unit may report the nature of the fault through the status register.

Configuration Faults

The base unit declares a configuration fault if it detects either of the following conditions:

- an attempt was made to change a value in the control register while the *run/halt* bit is set to 1
- an attempt was made to set the reserved bit 16 in the control register to 1

After declaring a configuration fault, the base unit clears all bits in the control register to 0, reports the configuration fault to the host via the status register, activates the *interface fault* light, and flashes the *fault number* light the proper number of times. To recover from a configuration fault, write a 0 to the *run/halt* bit. The base unit's operating parameters can then be re-initialized, and the unit can resume normal operation.

Continued on next page

Operating Characteristics, Continued

SERIPLEX Bus Faults

When the interface base unit detects any bus fault (except a CDR fault), the unit halts the bus at the end of the current data frame. The internal timers on all bus devices expire and their output signals revert to their default states, without reacting to the bus data received during that frame. The interface base unit reports the specific bus fault(s) to the host via the status register, as well as activating the *bus fault* light and flashing the *fault number* light as described in Table 18 on page 72.

Unless the base unit's auto restart feature is enabled, the bus does not restart until the unit receives software commands first to stop, then to start the bus. The bus remains halted for at least 250 milliseconds. If auto restart is enabled, the base unit periodically attempts to restart the bus, as described in the section "Auto Restart" on page 16.

The base unit's response to a detected CDR fault depends on the CDR ride-through selection, as described in the section "Multi-Bit Data Validation with CDR" on page 13.

Troubleshooting Guide

Troubleshooting

Table 19 below suggests possible solutions when troubleshooting the SERIPLEX system, based on faults detected by the interface base unit. For more information about system troubleshooting, consult *SERIPLEX Design, Installation and Troubleshooting* (bulletin number 30298-035-01_).

Table 19: Troubleshooting Guide

Fault Type	Possible Cause	Corrective Action(s)
Internal CPU or program fault	Internal hardware fault Corrupt firmware memory	Reset unit by cycling power If error persists after reset, replace unit.
Internal RAM fault	Internal microprocessor RAM failure	Reset unit by cycling power If error persists after reset, replace unit.
External RAM fault	External RAM failure	Reset unit by cycling power If error persists after reset, replace unit.
FPGA load fault	Backplane circuitry on unit failed to initialize properly Possible corruption of unit's program memory	Reset unit by cycling power If error persists after reset, replace unit.
Interface person-ality fault	Internal unit identification file is corrupted	Reset unit by cycling power If error persists after reset, replace unit.
Internal watch-dog fault	Internal hardware fault Corrupt firmware memory	Reset unit by cycling power If error persists after reset, replace unit
Data flow fault	Unit failed to process bus data within the time limits Possible internal hardware failure	Reset unit by cycling power Reduce bus clock rate. If error persists after reset, replace unit.
Other internal fault	Unrecognized internal error	Reset unit by cycling power If error persists after reset, replace unit.

Table 19: Troubleshooting Guide (Cont.)

Fault Type	Possible Cause	Corrective Action(s)
Bus undervoltage	Bus power missing	Verify that the bus power supply is turned on Check the bus wiring for an open circuit between the power supply and the base unit.
	Bus power voltage too low	Check bus power supply for the proper output voltage, adjust if necessary. Check the SERIPLEX system for an excessive power load, add power supplies if necessary.
Data line stuck low	SERIPLEX data line shorted to SERIPLEX common line	• Check bus wiring for shorts between data and common lines, or between data and ground. • Especially check wiring at the device connectors, entrances to junction boxes, and bends in conduit and wiring – a single stray strand of wire can create a fault. • Check bus devices for shorts by disconnecting bus cable.
Data line stuck high [1]	SERIPLEX data line shorted to SERIPLEX power line [1]	• Check bus wiring for shorts between data and power lines. • Especially check wiring at the device connectors, entrances to junction boxes, and bends in conduit and wiring – a single stray strand of wire can create a fault. • Check bus devices for shorts by disconnecting bus cable.

[1] At cable distances of more than 500 ft., a data line stuck high condition may not be detected.

Table 19: Troubleshooting Guide (Cont.)

Fault Type	Possible Cause	Corrective Action(s)
Clock fault	SERIPLEX clock line shorted to another bus conductor	• Check bus wiring for shorts between the clock line and other bus conductors. • Especially check wiring at the device connectors, entrances to junction boxes, and bends in conduit and wiring – a single stray strand of wire can create a fault. • Check bus devices for shorts by disconnecting bus cable.
	Electrical noise disrupting the clock signal	Consult *SERIPLEX Design, Installation and Trouble-shooting* (30298-035-01_).
Data line undercurrent	SERIPLEX data line shorted to SERIPLEX common line	Check indicator lights or host software for the presence of a data line stuck low condi-tion (see instructions for data line stuck low condition).
	Base unit's data current source is out of adjustment or working incorrectly	Replace base unit.

Table 19: Troubleshooting Guide (Cont.)

Fault Type	Possible Cause	Corrective Action(s)
Data line overcurrent	SERIPLEX clock line shorted to SERIPLEX data line	• Check bus wiring for shorts between clock and data lines. • Especially check wiring at the device connectors, entrances to junction boxes, and bends in conduit and wiring – a single stray strand of wire can create a fault. • Check bus devices for shorts by disconnecting bus cable.
	SERIPLEX data line shorted to SERIPLEX power line [1]	• Check indicator lights or host software for the presence of a data line stuck high condition (see instructions for data line stuck high condition). [1]
	Base unit's data current source is out of adjustment or working incorrectly	• Replace base unit.
	SERIPLEX common line is open or the voltage drops on the common line are too large (>1.0 V)	• Check the continuity of the common line throughout the installation. • Increase the size of the common line or add a second common line in parallel, to reduce voltage drops. • Do not run control load power through the bus cable.
Excess data line capacitance	Bus cable length and/or load too great for the bus clock rate	• Decrease the bus clock rate. • Re-route the bus cable to reduce total cable length.

[1] At cable distances of more than 500 ft., a data line stuck high condition may not be detected.

Table 19: Troubleshooting Guide (Cont.)

Fault Type	Possible Cause	Corrective Action(s)
Input CDR fault	A SERIPLEX bus input device is not transmitting the proper CDR data	• Verify that the input device is connected to the bus and operating properly. • Verify that the input device is assigned to the proper bus addresses. • Verify that the input device is configured to produce CDR data.
	Multiple bus input devices are assigned to the same bus address(es)	• Verify that the input devices are properly addressed. • Especially verify that no input signal addresses conflict with another signal's CDR data.
	Bus cable open or shorted	• Check the bus wiring for shorts between clock and data lines. • Especially check wiring at the device connectors, entrances to junction boxes, and bends in conduit and wiring – a single stray strand of wire can create a fault.
	Excess bus capacitance is causing invalid signals	• See instructions for the excess data line capacitance condition.
	Electrical noise is disrupting the bus signals	• Consult *SERIPLEX Design, Installation and Troubleshooting* (30298-035-01_).

Table 19: Troubleshooting Guide (Cont.)

Fault Type	Possible Cause	Corrective Action(s)
Output CDR fault	The base unit is not transmitting CDR data	• Verify that the unit is properly configured to produce CDR data for the desired output signals
	A SERIPLEX bus output device is not processing CDR data properly	• Verify that the output device is connected to the bus and operating properly • Verify that the output device is assigned to the proper bus addresses
	Bus cable open or shorted	• Check the bus wiring for shorted or open wires • Especially check wiring at the device connectors, entrances to junction boxes, and bends in conduit and wiring – a single stray strand of wire can create a fault
	A bus input device is inadvertently indicating an output CDR fault	• Verify that no bus input device is assigned to and producing a signal at bus input 9
	Excess bus capacitance is causing invalid signals	• See instructions for the excess data line capacitance condition
	Electrical noise is disrupting the bus signals	• Consult *SERIPLEX Design, Installation and Troubleshooting* (30298-035-01_)

Specifications

Physical Characteristics

Table 20: Physical Characteristics (Figure 15. on page 83)

Parameter	Value
SERIPLEX bus connection [1]	• One five-pin header: 0.200 inches centers with the mating screw terminal connector, Phoenix SMSTB 2,5/5-ST-5,08 or Square D SPX Termbus • One five-pin header: 0.156 inches centers with the mating insulation-displacement connector, AMP 641229-5
Power connection [2]	2-pin, removable screw terminal connector Phoenix SMSTB 2,5/2-ST-5,08
Width	4.9 inches
Height	5.6 inches
Depth	1.6 inches
Operating temperature	0 to 60 °C
Storage temperature	-25 to 80 °C
Relative humidity	5 to 95%, non-condensing
Vibration	IEC 6B-2-6 10 - 150 HZ, 1g
Shock	IEC 6B-2-27 15g, 11ms, half-sine

1. Both connectors are wired in parallel.
2. The power connection is wired in parallel to the bus connections.

Continued on next page

Specifications, Continued

Electrical Characteristics

Table 21: Electrical Characteristics

Parameter	Value
Nominal power voltage	24 Vdc
Power voltage range	19.2 to 30 Vdc
Current draw	450 mA maximum, at 24 Vdc
Power Hold up Time	1ms
Electrostatic Discharge (ESD) IEC 1000-4-2, level 3	8 kV air, 6 kV contact
Radiated RF (IEC 1000-4-3, level 3)	10 V/m, 80 MHz to 1 GHz
Electronic Fast Transient Burst (EFTB) (IEC 1000-4-4, level 3)	1000 V – SERIPLEX cabling 2000 V – power lines
Surge (IEC 1000-4-5, level 2)	500 V @ 2 Ω, power to common 1 kV @ 12 Ω, power/common to shield 2 kV @ 2 Ω, shield to shield (at a distance of 500 ft.)

Bus Operating Parameters

Table 22: Bus Operating Parameters

Parameter	Value
Clock rate	10, 16, 25, 32, 64, 100, 147, 192 kHz
Frame length	64, 128, 192, 256 addresses
Bus operating mode	Master/slave only

Continued on next page

Specifications, Continued

Physical Dimensions and Mounting Hole Locations

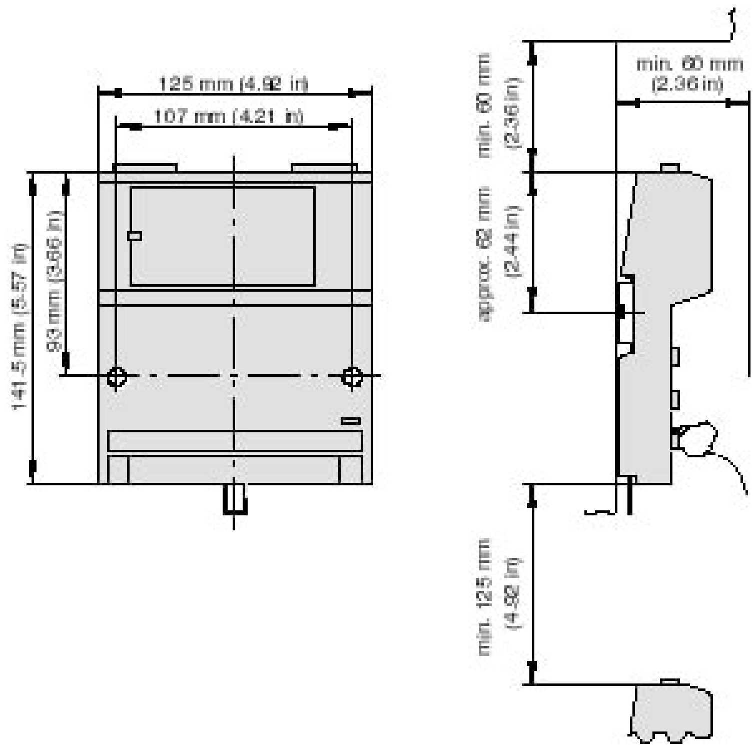


Figure 15. Unit Dimensions

Index

Modicon, Square D and Telemecanique are PLC brand names from Schneider. These products are sold in the US by Square D; in Canada, Latin America, Europe, Africa, Asia/Pacific and Middle East by Schneider; in Germany by AEG Schneider Electric; in China and Persian Gulf by Schneider Electric; in South Africa by ASA Systems Automation; in Austria by Online.

Schneider Electric, Inc.
One High Street
North Andover, MA 01845
Tel: (1) 508-794-0800
Fax: (1) 508-975-9400

Schneider Electric GmbH
Steinheimer Strasse 117
D-63500 Seligenstadt
Tel: (49) 6182 81-2584
Fax: (49) 6182 81-2860

Schneider Electric S.A.
245, Route des Lucioles-BP147
F-06903 Sophia-Antipolis Cedex
Tel: (33) 92 96 20 00
Fax: (33) 93 65 37 15

30298-086-01