DataGeneral

Technical Reference

4206 SERIES MULTIPROCESSOR COMMUNICATIONS ADAPTER

014-000072-00

Technical Reference

4206 SERIES MULTIPROCESSOR COMMUNICATIONS ADAPTER

014-000072-00

NOTICE

Data General Corporation (DGC) has prepared this manual for use by DGC personnel and customers as a guide to the proper installation, operation, and maintenance of DGC's equipment and software. The drawings and specifications contained herein are the property of DGC and shall neither be reproduced in whole or in part without DGC prior written approval nor be implied to grant any license to make, use, or sell equipment manufactured in accordance herewith.

DGC reserves the right to make changes without notice in the specifications and materials contained herein and shall not be responsible for any damages (including consequential) caused by reliance on the materials presented, including but not limited to typographical or arithmetic errors, company policy and pricing information. The information contained herein on DGC software is summary in nature. More detailed information on DGC software is available in current released publications.

NOVA, SUPERNOVA, ECLIPSE and **NOVADISC** are registered trademarks of Data General Corporation, Southboro, Mass.

Ordering No. 014-000072

Printed in the United States of America
Rev. 00, July 1976

TABLE OF CONTENTS

SECTION I
GENERAL DESCRIPTION

SECTION II
LOGICAL ORGANIZATION

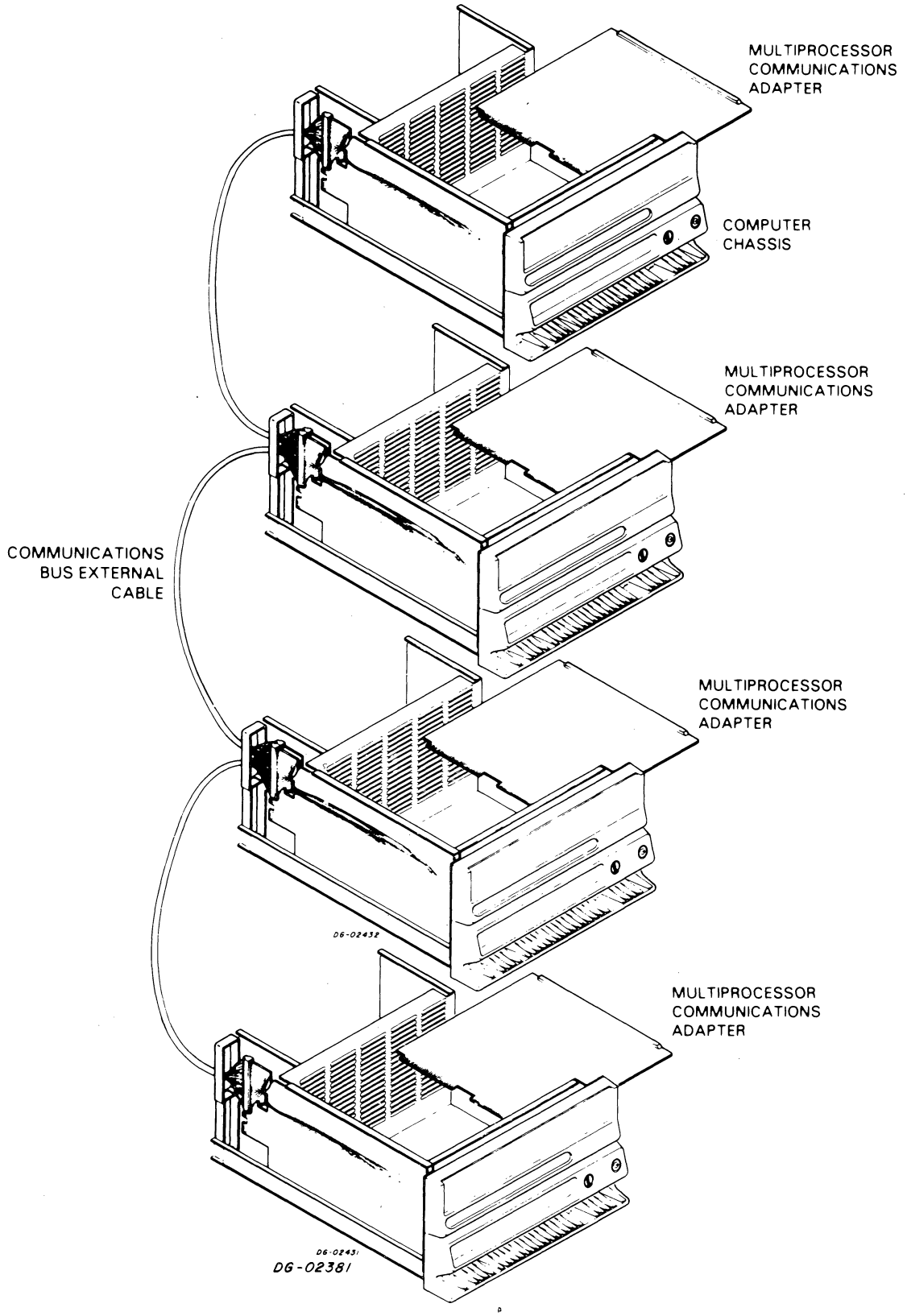
SECTION III
PROGRAMMING

TABLE OF CONTENTS (Continued)

SECTION IV
INSTALLATION

APPENDICES

This page intentionally left blank.


MULTIPROCESSOR
COMMUNICATIONS
ADAPTER
COMPUTER
CHASSIS
MULTIPROCESSOR
COMMUNICATIONS
ADAPTER
COMMUNICATIONS
BUS EXTERNAL
CABLE
MULTIPROCESSOR
COMMUNICATIONS
ADAPTER
DG-02432
MULTIPROCESSOR
COMMUNICATIONS
ADAPTER
DG-02431
DG-02381

DG-02381

SECTION I
GENERAL DESCRIPTION

INTRODUCTION

The 4206 series multiprocessor communications adapter (MCA) subsystem allows the connection of up to fifteen NOVA and/or ECLIPSE line computers to form a multiprocessor system. Blocks of data are transferred at high-speed from one computer to another through the computer's data channel facilities. Such a system composed of many small computers is a powerful, highly flexible alternative to a single large computer in many applications.

An MCA subsystem consists of a common communications bus and an adapter for each computer in the system.

The communications bus provides a daisy-chain connection to the adapters for each computer in the system. The communications bus is time-division multiplexed among the adapters.

Each adapter connects a single computer to the communications bus. The adapter is a 15" square printed circuit board that mounts in any I/O slot of the computer chassis. Each adapter has a unique jumper-selectable number from 1 to 15 which is used to identify it in data transfers to and from other computers in the multiprocessor system.

The multiprocessor communications adapter contains two distinct devices, a transmitter and a receiver, which allow for the independent transmission and reception of data. Each device is connected separately to the data channel. The program need only set up a device for sending or receiving, and all transfers to and from memory are then handled automatically by the data channel hardware.

A processor with an MCA can establish a link between its transmitter and any receiver it designates, provided that the receiving adapter has been initialized for reception. Upon receipt of the first data word from a transmitting adapter, the receiver will "lock" onto that transmitter and will subsequently accept further data only from that transmitter. An I/O instruction must be issued by the receiving processor to "unlock" the receiver so that it can receive data from another transmitter.

The MCA makes special provision for graceful system degradation in the case of hardware or software failure. The communications bus is designed in such a way that any one of the interconnected computers can be stopped or have its power switched off without affecting the other computers still in operation. If a transmitter attempts to transmit data to an unavailable receiver, a timeout interrupt will occur approximately 10 milliseconds later, after the transmitter has made several thousand attempts to transmit the data. If the receiver is unavailable because it is locked to another transmitter, a lockout indicator will be set when the timeout interrupt occurs. The transmitter may be restarted for further attempts or the data may be routed to a different receiver.

The size and nature of the data transmission can follow any convention established by the user; no particular structure is forced by the hardware design. In a relatively simple system in which the size and nature of the data blocks to be transferred is always known in advance, the receiver can simply initialize itself to accept the next block at the completion of the previous transmission.

If the exact size and nature of the data blocks is determined dynamically, a control block specifying the nature of a transfer can be transmitted before the actual data block. With such a convention, the receiver initializes itself to accept a control block of standard format and unlocks itself. The first word transferred to the receiver locks it to the sending transmitter until explicitly unlocked by the program. Thus, once the first word in a control block is received, the receiver is locked to that transmitter and can be initialized to accept subsequent data blocks from that transmitter.

Alternatively, the control block from adapter A to adapter B can be a request for data. Adapter B's transmitter can start sending the desired data while its receiver is reinitialized to accept a new control block. The hardware in no way distinguishes between data and control blocks.

An MCA network can operate in one of two jumper-selectable modes: either normal mode or fast mode. A network operating in normal mode can transmit up to 312,500 words per second, and may contain up to 15 processors with a maximum communications bus cable length of 150 feet. A network operating in fast mode can transmit up to 500,000 words per second, and may contain up to 4 processors with a maximum communications bus cable length of 40 feet. For either type of network, the communications bus is shared equally among the active logical links. If there are 4 logical links established and communications are proceeding on all, each link receives 1/4 of the communications bus time. Data transfer rates are primarily determined by the response time of the processors' data channel facilities. The maximum data transfer rate will be obtained only when a large number of links are active concurrently. A detailed analysis of the data transfer rates supported by the MCA is contained in Section II.

The multiprocessor communications adapter may be used in a wide variety of applications where it is convenient to partition the job. In a typical multiprocessor system, one computer might handle a number of data communications lines while another preprocesses data, and a third performs the actual computations. Such a system is often less expensive and far more flexible than any single medium- to large-scale computer capable of meeting all the job requirements. In addition, because several compatible processors are used, the system provides its own backup in case of failure.

SECTION II
LOGICAL ORGANIZATION

INTRODUCTION

The multiprocessor communications adapter subsystem provides a facility for data to be transferred from one computer to another, a block at a time, using the computers' data channel facilities. Each MCA consists of two separate devices, a transmitter and a receiver, which allow for the independent transmission and reception of data.

Each transmitter and receiver contains an address register, a word count register, and a status register. The address register is initialized to contain the memory location of the first word to be transferred, and the word count register is initialized to contain the two's complement of the number of words to be transferred. The transmitter registers must be initialized by the transmitting processor, and the receiver registers must be initialized by the receiving processor.

When a transmitter and the receiver with which it wants to communicate have been properly initialized, data transmission begins. Upon receipt of the first data word, the receiver locks onto the transmitter, and will accept data only from that transmitter until the receiver is explicitly unlocked. Data transmission continues until the word count register of either the transmitter or receiver increments to 0.

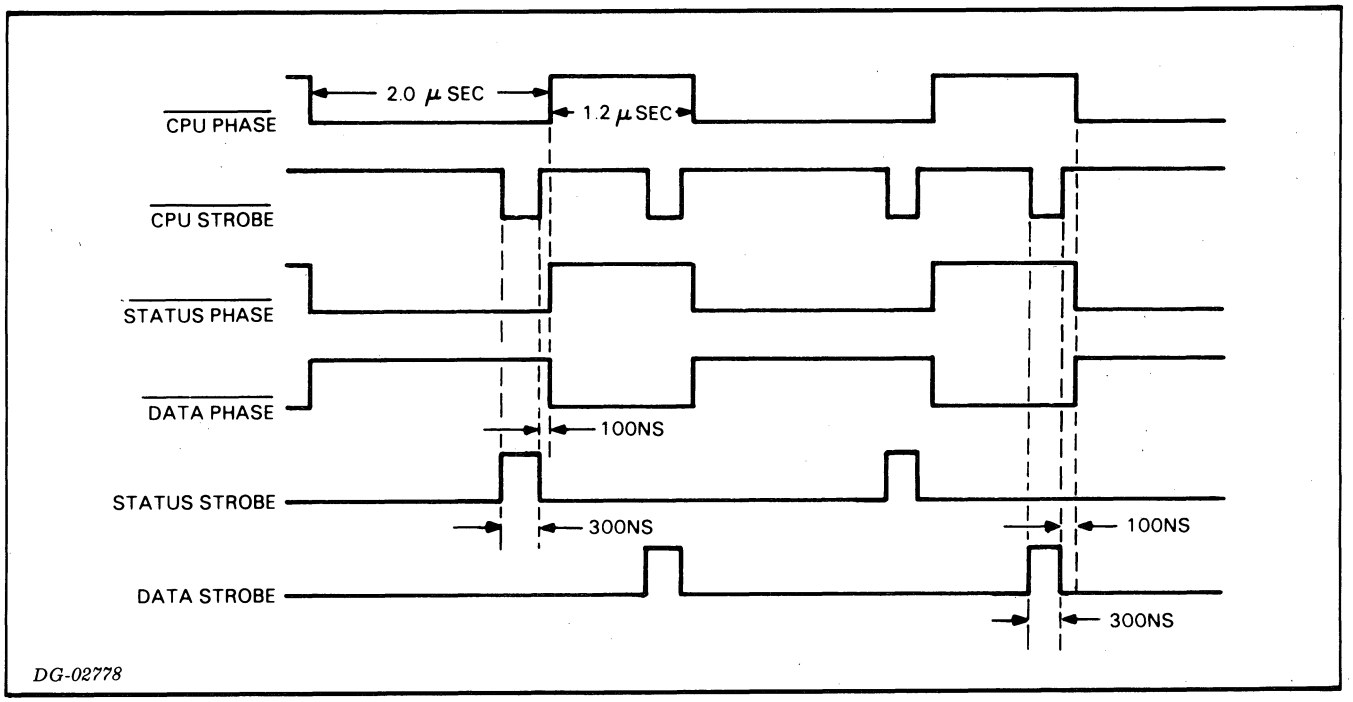
The computers in an MCA network are connected, in daisy-chain fashion, to a single communications bus. Each MCA has two ports, OUT and IN, for connection to the communications bus. The OUT port of one MCA is connected to the IN port of the next MCA in the daisy chain. The MCA which does not have another MCA connected to its IN port is referred to as the "leftmost" adapter, and the MCA which does not have another MCA connected to its OUT port is referred to as the "rightmost" adapter.

The signals on the communications bus can be grouped into four categories based on the type of information they carry. The four categories are status signals, data signals, timing signals, and priority control signals. The status signals ($\overline{\text{S-CPU [0-7,12-15]}}$) are used to establish transmitter-receiver pairings and to transfer status information such as the states of the transmitter and receiver word count registers from one computer to another. The data portion of the communications bus ($\overline{\text{CPU [0-15]}}$) is used to transmit 16 data bits in parallel from one computer to another. The timing signals are used to ensure that only one transmitter-receiver pair is using the communications bus at any given time and to control the transfer of status information and data between that transmitter-receiver pair. The priority control signals determine which of the active transmitters will be given access to the bus, using a round-robin priority scheme.

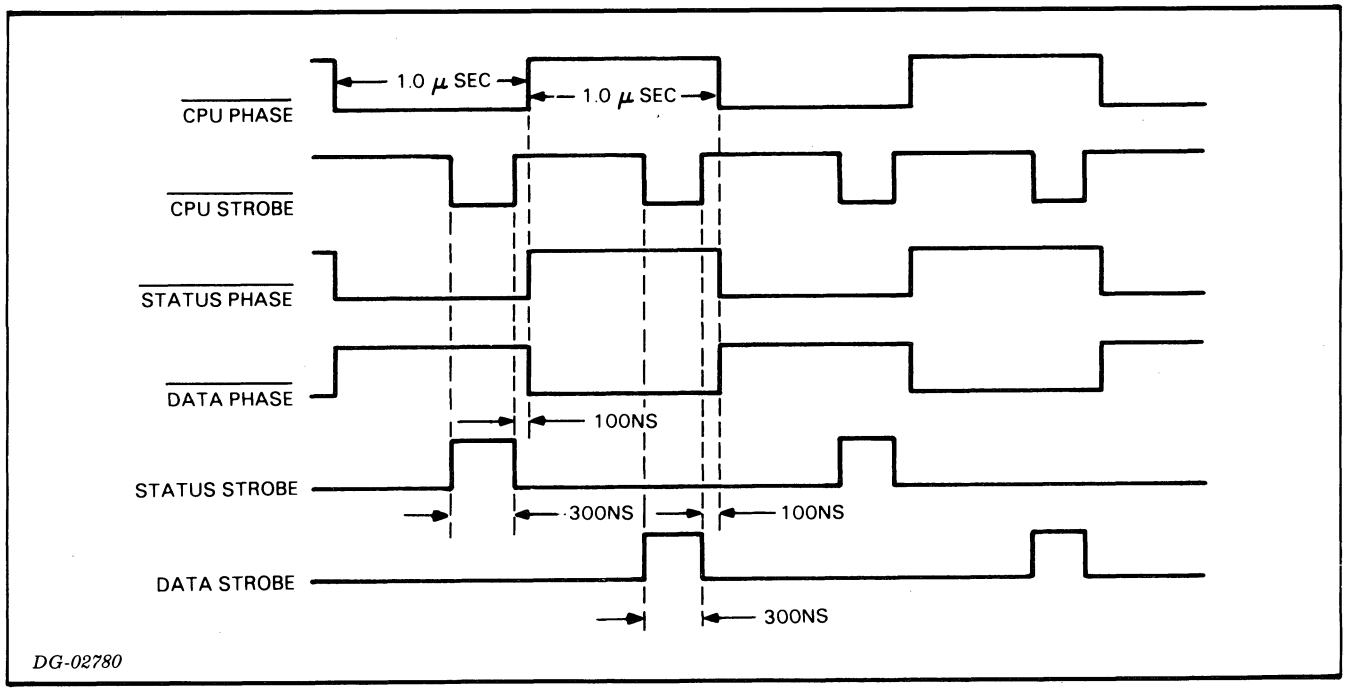
TIMING STATES

An MCA network can operate in one of two modes: normal mode or fast mode. A network operating in normal mode has a basic cycle time of 3.2 microseconds and a maximum data transfer rate of 312,500 words per second. A network operating in fast mode has a basic cycle time of 2.0 microseconds and a maximum data transfer rate of 500,000 words per second.

The "leftmost" adapter in an MCA subsystem determines the operating mode of the network, based on its jumper settings, and generates the timing signals for the communications bus. The timing signals are **CPU PHASE** and **CPU STROBE**. Each MCA further divides these signals into two phases, **STATUS PHASE** and **DATA PHASE**, and two strobes, **STATUS STROBE** and **DATA STROBE**. The relationship of the timing signals for normal mode and fast mode is as follows:



DG-02778



DG-02780

During the status phase the transmitter which is currently using the communications bus places its identifying number, the identifying number of the receiver to which it wishes to transmit data, and an indicator of the status of the transmitter's word count register on the status portion of the communications bus. The specified receiver then places a positive or negative acknowledgment, an indicator of whether or not the receiver is locked to a transmitter, and the status of the receiver's word count register on the status portion of the communications bus. The status indicators for the transmitter's and receiver's word count registers will be set to 1 if the next data word is the last word to be transmitted or received before that word count increments to 0.

The status strobe is used to clock the status of the receiver's word count register and lock into the transmitter's status register. In addition, if the receiver issued a positive acknowledgment, the status strobe is used to clock the transmitter's identifying number and the status of the transmitter's word count register into the receiver's status register.

During the data phase, the transmitter places the data bits of the word to be transmitted on the data portion of the communications bus. If the receiver issued a positive acknowledgment, the receiver clocks the data bits on the communications bus into its receive data register with the data strobe and initiates a data channel request. The receiver must transfer the data word to its processor via the data channel facility before it can receive another word from the transmitter. The data strobe is also used by the transmitter to initiate a data channel request to its processor for the next data word, if any, to be transmitted.

PRIORITY SYSTEM

When a transmitter is ready to transmit a data word after retrieving the word from its processor via the data channel facility, the transmitter issues a request for the communications bus. This request is issued at the beginning of a data phase. During the data phase, the priority ripples from the transmitter currently using the communications bus to the first transmitter requesting the bus to the "right" of the current transmitter, using the signal **REQP**. If the "rightmost" transmitter is not requesting the bus, priority returns to the "leftmost" transmitter. The transmitter currently using the communications bus cannot request the bus again during this data phase. If no transmitter is requesting the communications bus, the "leftmost" transmitter has highest priority when a request for the bus is made.

The highest priority transmitter requesting the communications bus becomes active to use the bus at the beginning of the next status phase.

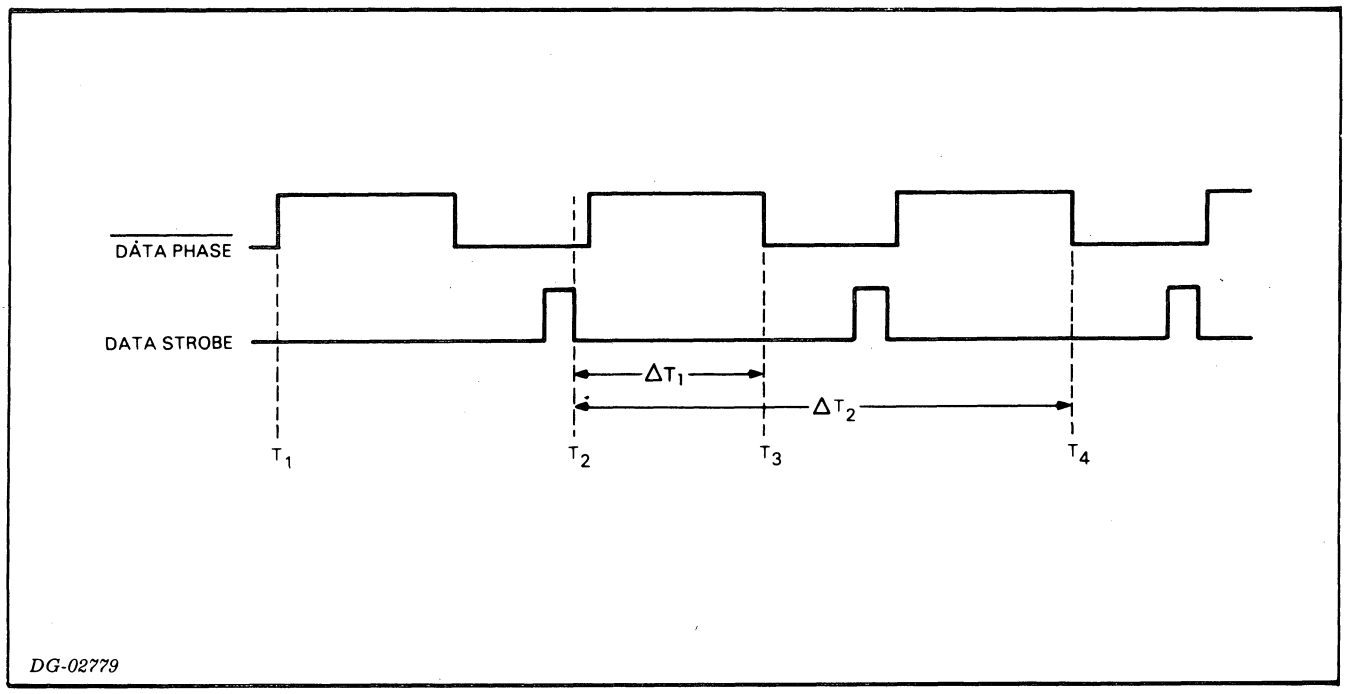
Within a single MCA, the receiver has higher priority than the transmitter for both data channel service and program interrupt service.

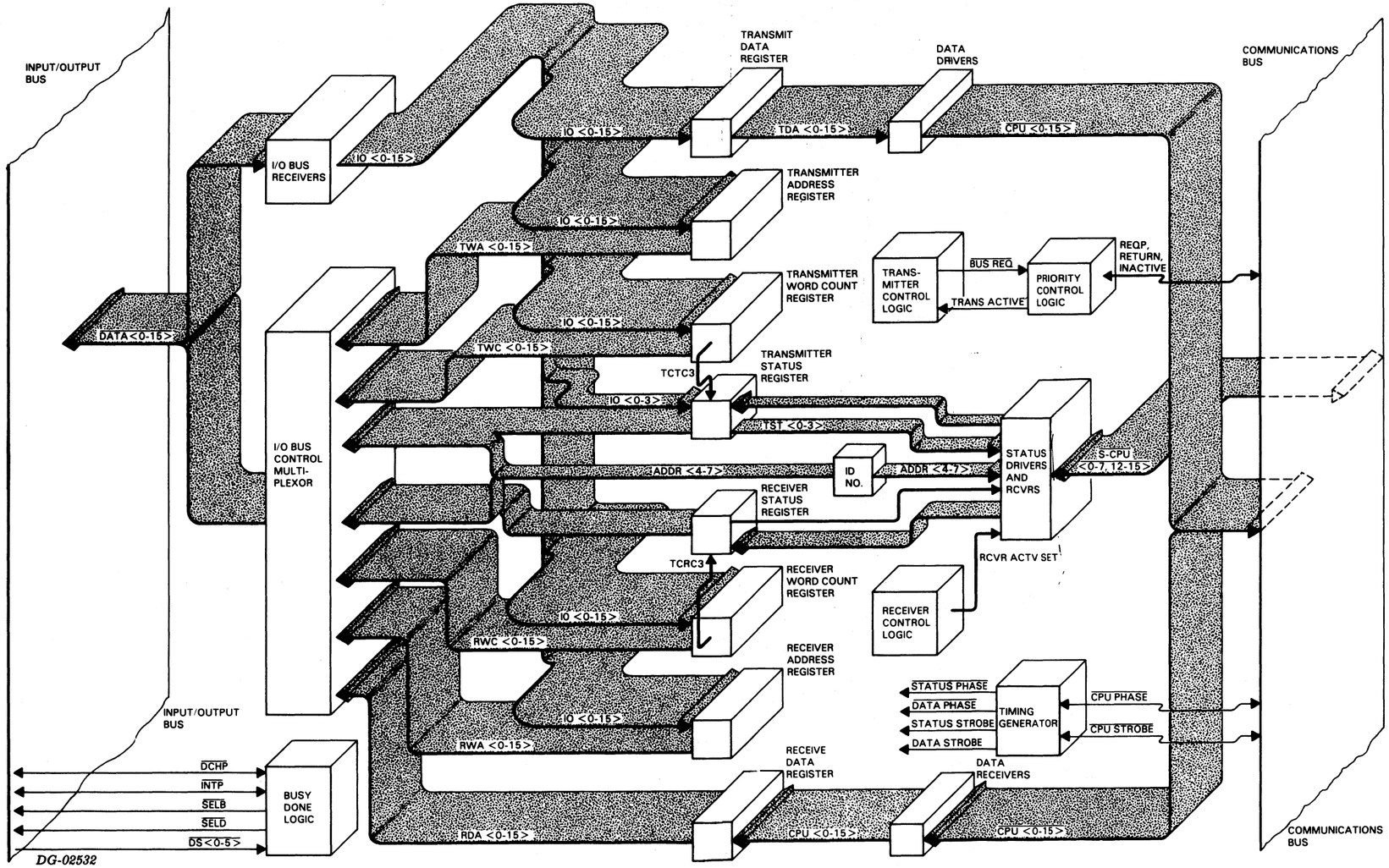
DATA THROUGHPUT

If several transmitter-receiver pairs are competing for the use of the communications bus, the maximum data transfer rate of 312,500 words per second in normal mode or 500,000 words per second in fast mode may be obtained. However, in general the data transfer rates are primarily determined by the speed of the processor's data channel facilities.

Consider an MCA subsystem with a single active transmitter-receiver pair. Assume that the transmitter gets the bus at time T_1, and the receiver issues a positive acknowledgment. At time T_2, the receiver requests a data channel cycle to store the data word, and the transmitter requests a data channel cycle to obtain the next data word.

The first time that the transmitter could request the bus again is T_3. However, the transmitter can request the bus at T_3 only if it has completed a data channel access in the time interval ΔT_1. For an MCA subsystem operating in normal mode, ΔT_1 is 2.1 microseconds, and for an MCA subsystem operating in fast mode, ΔT_1 is 1.1 microseconds. With no data channel latency, a data channel cycle for a typical processor takes 1.0 microseconds using the high-speed data channel facility and 2.0 microseconds using the standard data channel facility. In any case, it is unlikely that the data channel cycle can be completed by T_3. In general, the transmitter must wait until time T_4 or later to request the communications bus, where ΔT_2 is 5.3 microseconds for an MCA subsystem operating in normal mode and ΔT_2 is 3.1 microseconds for an MCA subsystem operating in fast mode. The respective data transfer rates are 1/9.6 microseconds $\cong$ 100,000 words per second in normal mode and 1/6.0 microseconds $\cong$ 165,000 words per second in fast mode.





DG-02532

THE MULTIPROCESSOR COMMUNICATIONS ADAPTER (MCA) subsystem allows the connection of up to fifteen NOVA and/or ECLIPSE line computers to form a multiprocessor system. Each MCA consists of two independent devices, a transmitter and a receiver, which are separately connected to the computer's I/O bus. A single timing generator located on the "leftmost" MCA in the system coordinates the activities of the transmitters and receivers in the MCA network. The priority control logic determines which transmitter in the MCA network has access to the shared communications bus at any given time.

The transmitter transfers 16-bit data words in parallel from the I/O bus via the data channel to the transmit data register. When a data word has been loaded into the transmit data register, the transmitter requests the use of the shared communications bus. Subsequently, when the transmitter is given access to the communications bus, the transmitter places its identifying number, the identifying number of the receiver to which it wants to transmit data, and the status of its word count register on the status portion of the communications bus. The transmitter then reads the status of the specified receiver from the status portion of the communications bus, and places the contents of its transmit data register on the data portion of the communications bus. If the receiver status included a positive acknowledgment, the transmitter then requests another data word from the computer via the data channel; otherwise, the transmitter requests to use the communications bus again on a later cycle to try to transmit the same data. The transmitter control logic maintains the transmitter's address, word count, and status registers, including a timeout indicator, coordinates the transmitter's data channel activities, and generates requests to use the communications bus.

The receiver control logic maintains the receiver's address, word count, and status registers, including a lock and a timeout indicator, coordinates the receiver's data channel activities, and monitors the status portion of the communications bus for requests to transmit data to this receiver. When such a transmission request is detected, the receiver reads the status of the transmitter from the status portion of the communications bus into its receiver status register. The receiver responds by placing the status of its lock, the status of its word count register, and a positive or negative acknowledgment on the status portion of the communications bus. The receiver will issue a positive acknowledgment if the receiver is not locked, or if it is locked to the transmitter that initiated the request, and if the receive data register is not in use as a result of a previous transmission. Subsequently, if the receiver issued a positive acknowledgment, the receiver will transfer a 16-bit data word in parallel from the data portion of the communications bus to the receive data register. The receiver will then issue a data channel request to transfer the data from the receive data register through the I/O bus control multiplexor to the data channel of the I/O bus.

The Busy and Done logic and the address, word count, and status registers of the transmitter and receiver allow the program to determine in detail the status of the MCA subsystem.

SECTION III
PROGRAMMING

INTRODUCTION

The multiprocessor communications adapter provides a facility for data to be transferred, a block at a time, from one computer to another in a network consisting of up to fifteen NOVA and/or ECLIPSE line computers. One MCA is installed in each computer. The MCA communicates with that computer over its I/O bus and with adapters in the other computers in the network over a common communications bus.

A data transfer from one computer to another must be initiated by both the transmitting and receiving processors using programmed I/O instructions. The actual data transfer uses the data channel facilities of the respective computers.

Each MCA consists of two distinct devices, a transmitter and a receiver, which allow for the independent transmission and reception of data. Each device has its own device code and separate data channel connection, with the receiver having higher priority than the transmitter. The transmitter and receiver share a common identifying number, selected with jumpers, which is used to distinguish that MCA and associated computer in communications with other computers in the network.

Each transmitter and receiver contains an address register, a word count register, and a status register. The address register is initialized to contain the memory location of the first word to be transferred, and the word count register is initialized to contain the two's complement of the number of words to be transferred. The transmitter registers must be initialized by the transmitting processor, and the receiver registers must be initialized by the receiving processor.

The transmitting processor specifies the identifying number of the receiver to which it wishes to transmit data. Then, if both the transmitter and receiver have been turned on and their address and word count registers initialized, data transmission begins. Upon receipt of the first data word, the receiver locks onto the transmitter, and will accept data only from that transmitter until the receiver is explicitly unlocked with an I/O instruction. Data transmission continues until the number of words transmitted equals the smaller of the transmitter word count and the receiver word count.

Both the transmitter and receiver have a timeout indicator which is set to 1, generating an interrupt, if data transmission has been initiated but no new data has been transferred for approximately 10 milliseconds. If the specified receiver has not been initialized or is locked to another transmitter, the transmitter will generate a timeout interrupt approximately 10 milliseconds after the transmitter is turned on. In all other cases, a timeout interrupt indicates some form of hardware or software failure.

The size and nature of the data transmission can follow any convention established by the software of the multiprocessor system, from messages composed of only standard-length blocks to messages composed of standard-length control blocks and arbitararily-sized data blocks, where the sizes of the data blocks are specified in the control blocks.

INSTRUCTIONS

The multiprocessor communications adapter contains two independent devices, a transmitter and a receiver. The transmitter contains a 16-bit address register, a 16-bit word count register, a 14-bit status register, and a 16-bit transmit data register. The receiver contains a 16-bit address register, a 16-bit word count register, a 12-bit status register, and a 16-bit receive data register.

Eleven instructions are used to program the MCA. Eight of these instructions are used to load and read the contents of the transmitter's and receiver's address and word count registers. Two instructions are used to read the transmitter's and receiver's status registers. The remaining instruction is used to specify the receiver to which the transmitter wishes to transmit data.

The transmitter and receiver have separate Busy and Done flags. The transmitter's Busy flag is set to 1 when the transmitter is turned on and attempting to transmit data. The transmitter's Done flag is set to 1 when transmission is terminated as a result of a timeout condition or the word count increasing to 0 for either the transmitter or the receiver to which it was transmitting data. An I/O RESET will turn off the transmitter, remove the transmitter from diagnostic mode, and set the Transmitter Count Done and Timeout indicators in the transmitter's status register to 0. The three transmitter device flag commands are used as follows:

f = **S** Set the Busy flag to 1 and the Done flag to 0, set the Transmitter Count Done and Timeout indicators in the transmitter status register to 0, and turn on the transmitter.

f = **C** Set the Busy and Done flags to 0, set the Transmitter Count Done and Timeout indicators in the transmitter status register to 0, and turn off the transmitter.

f = **P** Step the internal clock (**CLK**) in diagnostic mode.

The receiver's Busy flag is set to 1 when the receiver is turned on and enabled to receive data. The receiver's Done flag is set to 1 when reception is terminated as a result of a timeout condition or the word count increasing to 0 for either the receiver or the transmitter from which it was receiving data. An I/O RESET will turn off the receiver, set the contents of the receiver's address and word count registers to 0, and set the Transmitter Count Done, Receiver Count Done, Receiver Lock, and Timeout indicators in the receiver status register to 0. The receiver device flag commands are used as follows:

f = **S** Set the Busy flag to 1 and the Done flag to 0, set the Transmitter Count Done, Receiver Count Done, and Timeout indicators in the receiver status register to 0, and enable the receiver to accept data.

f = **C** Set the Busy and Done flags to 0, set the Transmitter Count Done, Receiver Count Done, Receiver Lock, and Timeout indicators in the receiver status register to 0, and turn off the receiver.

f = **P** No effect.

Primary Mnemonic
- Transmitter . . . . . MCAT
- Receiver . . . . . MCAR

Primary Device Code
- Transmitter 6
- Receiver 7

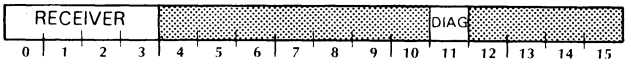
Secondary Mnemonic
- Transmitter MCAT1
- Receiver MCAR1

Secondary Device Code
- Transmitter 46
- Receiver 47

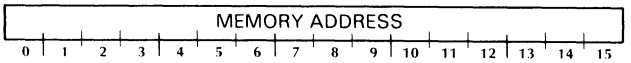
Priority Mask Bit
- Transmitter 12
- Receiver 12

ACCUMULATOR FORMATS

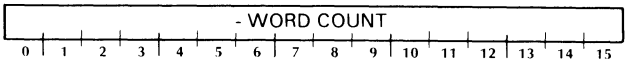
Specify Receiver **(DOC to MCAT)**



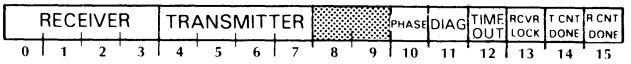
Load Transmitter Address Register **(DOA to MCAT)**



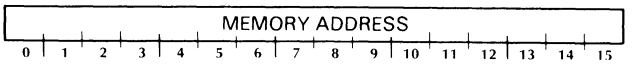
Load Transmitter Word Count **(DOB to MCAT)**



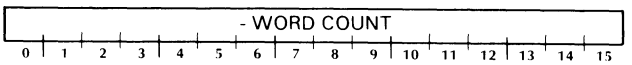
Read Transmitter Status **(DIC to MCAT)**



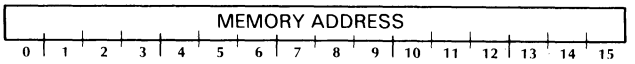
Read Transmitter Address Register **(DIA to MCAT)**



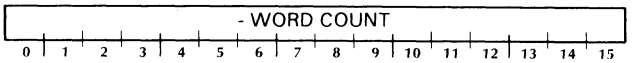
Read Transmitter Word Count **(DIB to MCAT)**



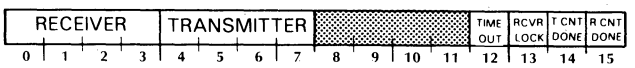
Load Receiver Address Register **(DOA to MCAR)**



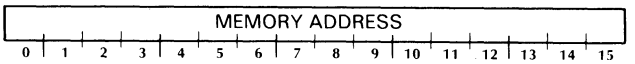
Load Receiver Word Count **(DOB to MCAR)**



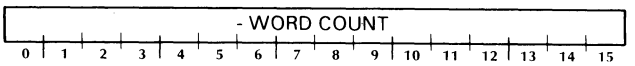
Read Receiver Status **(DIC to MCAR)**



Read Receiver Address Register **(DIA to MCAR)**



Read Receiver Word Count **(DIB to MCAR)**



S, C, AND P FUNCTIONS

MCAT

S Set the Busy flag to 1 and the Done flag to 0, set the Transmitter Count Done and Timeout indicators in the transmitter status register to 0, and turn on the transmitter.

C Set the Busy and Done flags to 0, set the Transmitter Count Done and Timeout indicators in the transmitter status register to 0, and turn off the transmitter.

P Step the internal clock (CLK) in diagnostic mode.

MCAR

S Set the Busy flag to 1 and the Done flag to 0, set the Transmitter Count Done, Receiver Count Done, and Timeout indicators in the receiver status register to 0, and enable the receiver to accept data.

C Set the Busy and Done flags to 0, set the Transmitter Count Done, Receiver Count Done, Receiver Lock, and Timeout indicators in the receiver status register to 0, and turn off the receiver.

P No effect.

CODING AIDS

In the descriptions of the separate instructions, the general form in which the instruction is coded in assembly language is given along with the instruction format and the description of the instruction. The general form in which an instruction may be coded has the following format:

MNEMONIC *[optional mnemonic]* *ac,* **DEVICE**

The mnemonic must be coded exactly as shown in the instruction description. The optional mnemonic may be appended to the main mnemonic if the option is desired.

Square brackets "[ ]" or *"[ ]"* along with boldface- and italic-printed symbols are used in this manual to aid in defining the instructions. These conventions are used to help describe how an instruction should be written so that it can be recognized by the assembler and translated into the correct binary, or machine language, representation. Their general definition is given below.

[ ], *[ ]* Square brackets indicate that the enclosed symbol is an optional operand or mnemonic. The operand enclosed in the brackets (e.g., *[f]*) may be coded or not, depending on whether or not the associated option is desired.

BOLD Operands or mnemonics printed in boldface must be coded exactly as shown. For example, the mnemonic for the DATA IN A instruction is coded **DIA**.

italic Operands or mnemonics printed in italics require a specific substitution. Replace the symbol with the number of a desired accumulator, or address, or with a user-defined symbol that the assembler recognizes as a specific name, address, number, or mnemonic.

The following abbreviations are used throughout this manual:

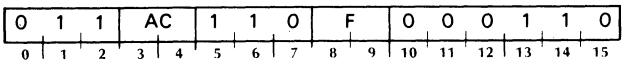
AC = Accumulator

F = Flag Control Function

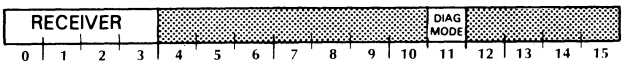
When describing the format of a word involved in an information transfer between the computer and the MCA, the various fields and bits in the word are labeled with names descriptive of their functions. Bits in the word which are not used by the MCA are shaded. Shaded bits are ignored on output and set to 0 on input.

Specify Receiver

DOC *[f]* *ac,* **MCAT**



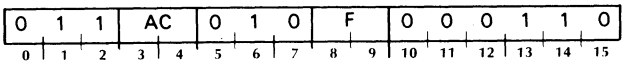
Bits 0-3 of the specified accumulator are used to select the receiver to which the transmission will be directed. Bit 11 of the specified AC controls whether the MCA is in diagnostic mode. If bit 11 is 1, the MCA is placed in diagnostic mode with the internal clock stopped. If bit 11 is 0, the MCA is placed in online mode with the internal clock running. Bits 4-10 and bits 12-15 of the AC are ignored. After the receiver and mode of operation have been selected, the function specified by F is performed. The contents of the specified AC remain unchanged. The format of the specified AC is as follows:



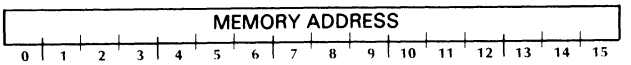
Bits	Name	Contents
0-3	Receiver	The identifying number of the receiver to which any subsequent transmissions are to be directed.
4-10	----	Reserved for future use.
11	Diagnostic Mode	The mode of operation of the MCA as follows: 0 Online mode 1 Diagnostic mode
12-15	----	Reserved for future use.

Load Transmitter Address Register

DOA *[f]* *ac,* **MCAT**



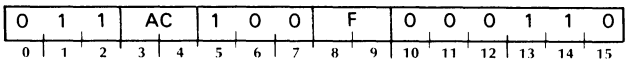
Bits 0-15 of the specified accumulator are placed in the transmitter's address register. After the transfer, the function specified by F is performed. The contents of the specified AC remain unchanged. The format of the specified AC is as follows:



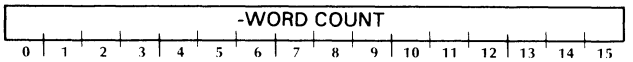
Bits	Name	Contents
0-15	Memory Address	The location of the next word in memory to be transmitted.

Load Transmitter Word Count

DOB *[f]* *ac*, **MCAT**



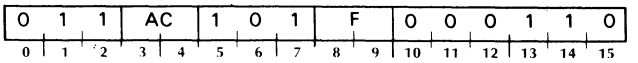
Bits 0-15 of the specified accumulator are placed in the transmitter's word count register. After the transfer, the function specified by F is performed. The contents of the specified AC remain unchanged. The format of the specified AC is as follows:



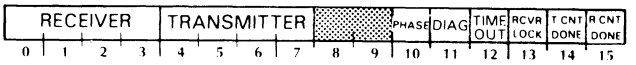
Bits	Name	Contents
0-15	- Word Count	The two's complement of the number of words to be transmitted.

Read Transmitter Status

DIC *[f]* *ac*, **MCAT**



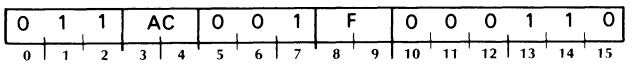
The contents of the transmitter's status register are placed in bits 0-7 and bits 10-15 of the specified accumulator. Bits 8-9 of the AC are set to 0. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



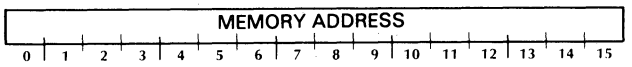
Bits	Name	Contents
0-3	Receiver	The identifying number of the receiver to which transmissions are directed.
4-7	Transmitter	The identifying number of this transmitter.
8-9	----	Reserved for future use.
10	Phase	If in online mode, zero. If in diagnostic mode, the state of the phase clock (DATA PHASE) as follows: 0 Low or Data phase 1 High or Status phase
11	Diagnostic Mode	The mode of operation of the MCA as follows: 0 Online mode 1 Diagnostic mode
12	Timeout	If the bit is 1, data transmisson has been attempted, but the selected receiver has not accepted data for 10 milliseconds. At the beginning of a block transfer, this bit set to 1 indicates that the processor at the receiving end has not initialized the receiver, or that the receiver is locked to another transmitter. If a block transfer is already in progress, this bit set to 1 indicates hardware or software failure.
13	Receiver Lock	If the bit is 1, the receiver identified in bits 0-3 is locked to either this transmitter or some other transmitter.
14	Transmitter Count Done	If the bit is 1, the transmitter has sent the last word of a block as specified by its word count register.
15	Receiver Count Done	If the bit is 1, the receiver has received the last word of a block as specified by its word count register.

Read Transmitter Address Register

DIA *[f]* *ac*, **MCAT**



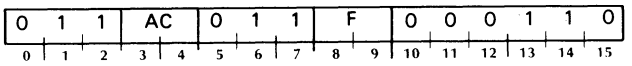
The contents of the transmitter's address register are placed in bits 0-15 of the specified accumulator. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



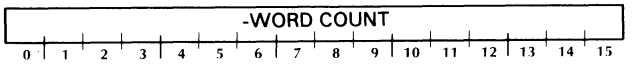
Bits	Name	Contents
0-15	Memory Address	The location of the next word in memory to be transmitted.

Read Transmitter Word Count

DIB *[f]* *ac*, **MCAT**



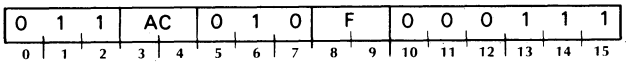
The contents of the transmitter's word count register are placed in bits 0-15 of the specified accumulator. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



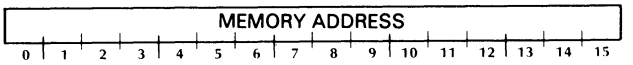
Bits	Name	Contents
0-15	- Word Count	The two's complement of the number of words yet to be transmitted.

Load Receiver Address Register

DOA *[f]* *ac*, **MCAR**



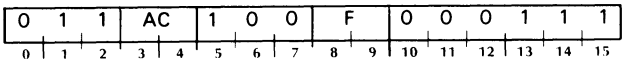
Bits 0-15 of the specified accumulator are placed in the receiver's address register. After the transfer, the function specified by F is performed. The contents of the specified AC remain unchanged. The format of the specified AC is as follows:



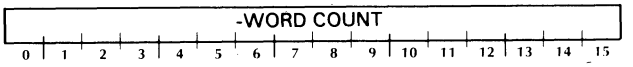
Bits	Name	Contents
0-15	Memory Address	The location in memory where the next word received is to be stored.

Load Receiver Word Count

DOB *[f]* *ac*, **MCAR**



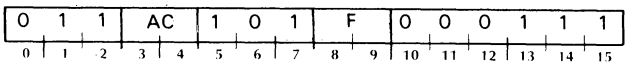
Bits 0-15 of the specified accumulator are plced in the receiver's word count register. After the transfer, the function specified by F is performed. The contents of the specified AC remain unchanged. The format of the specified AC is as follows:



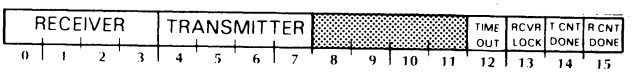
Bits	Name	Contents
0-15	- Word Count	The two's complement of the number of words to be received.

Read Receiver Status

DIC *[f]* *ac*, **MCAR**



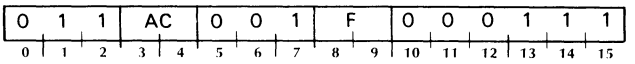
The contents of the receiver's status register are placed in bits 0-7 and bits 12-15 of the specified accumulator. Bits 8-11 of the AC are set to 0. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



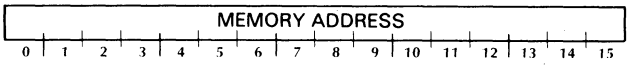
Bits	Name	Contents
0-3	Receiver	The identifying number of this receiver.
4-7	Transmitter	The identifying number of the transmitter to which the receiver is or was most recently locked.
8-11	----	Reserved for future use.
12	Timeout	If the bit is 1, data transmission to this receiver has been in progress, but no new data has been received for 10 milliseconds, indicating a hardware or software failure.
13	Receiver Lock	If the bit is 1, the receiver is locked to the transmitter identified in bits 4-7.
14	Transmitter Count Done	If the bit is 1, the transmitter has sent the last word of a block as specified by its word count register.
15	Receiver Count Done	If the bit is 1, the receiver has received the last word of a block as specified by its word count register.

Read Receiver Address Register

DIA *[f]* *ac*, **MCAR**



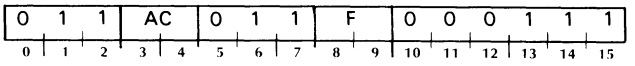
The contents of the receiver's address register are placed in bits 0-15 of the specified accumulator. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



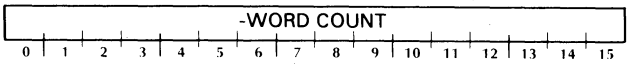
Bits	Name	Contents
0-15	Memory Address	The location in memory where the next word received is to be stored.

Read Receiver Word Count

DIB *[f]* *ac*, **MCAR**



The contents of the receiver's word count register are placed in bits 0-15 of the specified accumulator. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



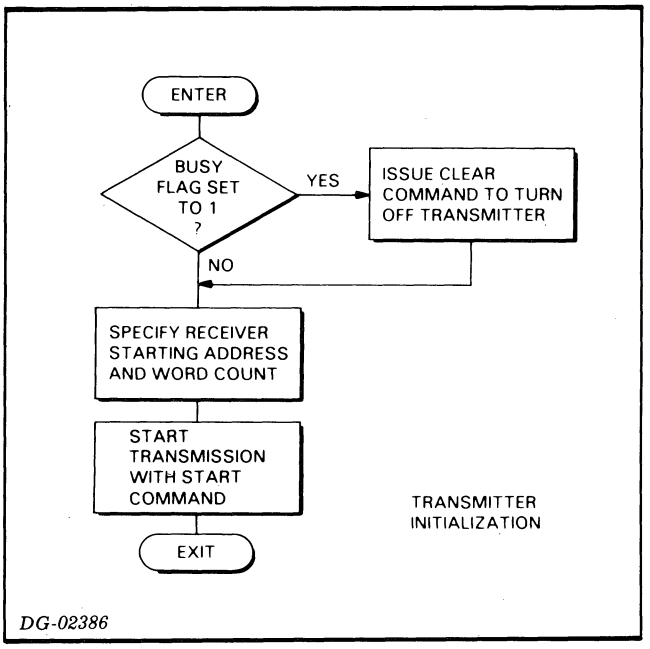
Bits	Name	Contents
0-15	- Word Count	The two's complement of the number of words yet to be received.

PROGRAMMING

Since the MCA is actually two separate devices, the transmitter and receiver are discussed separately.

Transmitter

The transmitter is initialized by specifying the receiver to which data is to be transmitted, the location in memory which contains the first word to be transmitted, and the two's complement of the number of words to be transmitted. The receiver to which data is to be transmitted is specified with a SPECIFY RECEIVER instruction (**DOC**). The location in memory which contains the first word to be transmitted is specified with a LOAD TRANSMITTER ADDRESS REGISTER instruction (**DOA**). The two's complement of the number of words to be transmitted is specified with a LOAD TRANSMITTER WORD COUNT instruction (**DOB**). The transmitter then can be turned on by setting its Busy flag to 1 with a Start command. The Start command can be appended to the last of the initializing instructions.



After the transmitter is turned on, it retrieves a word from the memory location specified by its address register and attempts to send this word to the receiver specified by the last SPECIFY RECEIVER instruction. If the receiver is locked to some other transmitter, the Receiver Lock bit in the transmitter's status register is set to 1; otherwise the Receiver Lock bit is set to 0. The transmitter continues to attempt to send the data word to the receiver until the transmission is successful or until a timeout occurs. A timeout will occur, and is indicated by the Timeout bit in the transmitter's status register being set to 1, if the receiver has refused to accept the word for at least 10 milliseconds. If the receiver accepts the word before a timeout occurs, the receiver locks onto the transmitter and the Receiver Lock bit in the transmitter's status register is set to 1.

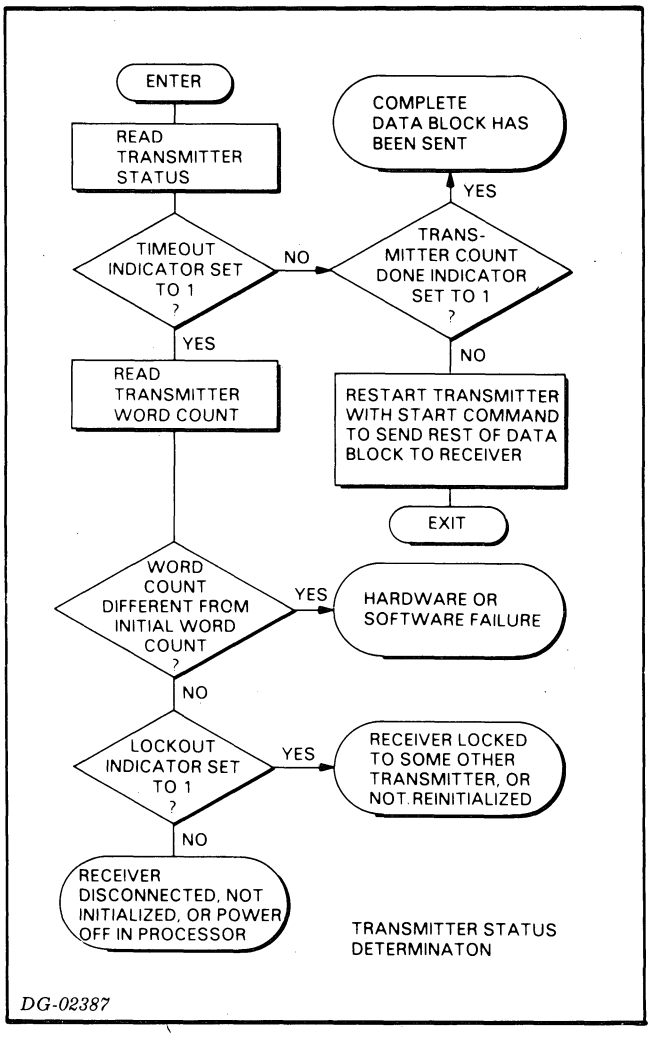
After the receiver locks onto the transmitter, the transmitter sends the block of words retrieved from the locations specified by its address register, and increments both the address and word count registers after sending each word.

When the word count of either the transmitter or the receiver increases to 0 or a timeout occurs, the data transmission terminates. In addition, the transmitter's Busy flag is set to 0, its Done flag is set to 1, and a program interrupt request is initiated.

The READ TRANSMITTER STATUS instruction (**DIC**) can be used to determine the condition which caused the transmission to terminate. If the Timeout indicator is set to 1, the complete data block has not been tranmitted. The READ TRANSMITTER ADDRESS REGISTER instruction (**DIA**) or the READ TRANSMITTER WORD COUNT instruction (**DIB**) can be used to determine if any data words have been transmitted. If the contents of either the address or word count register is different from the value with which it was initialized, some data has been transferred, and a hardware or software failure at the receiving processor resulted in the transmission being terminated. If no data has been transferred, the Receiver Lock bit in the transmitter's status register will be set to 1 if the specified receiver is locked to another transmitter, and will be set to 0 if the specified receiver has not been initialized or is otherwise unable to accept the data. In either case, the transmitter may make additional attempts to transmit the block. The transmitter can be restarted using a NO I/O TRANSFER instruction with a Start command (**NIOS MCAT**); the contents of the address, word count, and status registers remain unchanged.

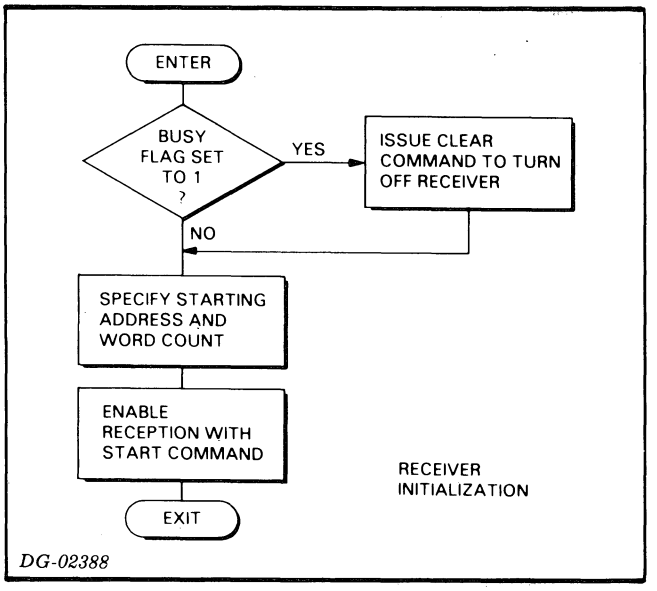
If the Timeout indicator is set to 0, the transmission terminated naturally with either the transmitter's or the specified receiver's word count increasing to 0. If the Transmitter Count Done indicator in the transmitter's status register is set to 1, the entire data block was sent to the receiver. If the Receiver Count Done indicator is set to 1, the number of words in the current block remaining to be sent to the receiver can be determined with a READ TRANSMITTER WORD COUNT instruction (**DIB**).

The Clear command will terminate the current block transfer, if any is in progress, and turn off the transmitter. In addition, the Transmitter Count Done and Timeout indicators in the transmitter status register are set to 0. Issuing a Clear command when a block transfer is in progress can cause data loss.



Receiver

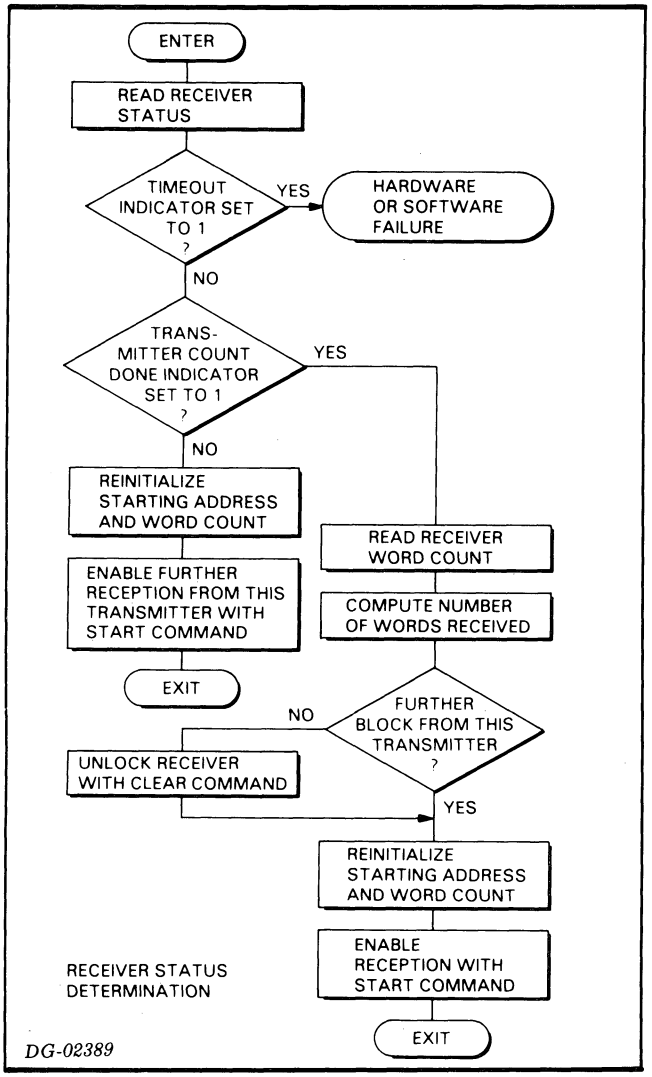
The receiver is initialized by specifying the location in memory where the first word received is to be stored, and the two's complement of the maximum number of words to be received. The location in memory where the first word received is to be stored is specified with a LOAD RECEIVER ADDRESS REGISTER instruction (**DOA**). The two's complement of the maximum number of words to be received is specified with a LOAD RECEIVER WORD COUNT instruction (**DOB**). The receiver then can be turned on and enabled to accept data by setting its Busy flag to 1 with a Start command. The Start command can be appended to the second initializing instruction.



As soon as some transmitter sends a word to the receiver, the receiver sets its Receiver Lock indicator to 1, and will only accept data from that transmitter until it is explicitly unlocked with a Clear command. As each word is received, the receiver stores the word in the memory location specified by its address register, and increments both the address and word count registers. The receiver will not accept a new data word until the previous word has been stored.

When the word count of either the transmitter or the receiver increases to 0 or a timeout occurs, the data transmission terminates. In addition, the receiver's Busy flag is set to 0, its Done flag is set to 1, and a program interrupt request is initiated.

The READ RECEIVER STATUS instruction (**DIC**) can be used to determine the condition which caused the transmission to terminate. If the Timeout indicator is set to 1, the receiver has not received any new data from the transmitter for at least 10 milliseconds, indicating a hardware or software failure at the transmitting processor. If the Timeout indicator is set to 0, the transmission terminated naturally with either the receiver's or transmitter's word count increasing to 0. If the Transmitter Count Done indicator is set to 1, an entire data block has been received. The number of words received can be calculated by subtracting the initial contents of the receiver word count register from the present contents of the receiver word count register, as determined with a READ RECEIVER WORD COUNT instruction (**DIB**). If the Receiver Count Done indicator is set to 1, the transmission terminated when the receiver's buffer filled. The transmission can be continued, with the receiver still locked to the transmitter, by reinitializing the receiver's address and word count registers and issuing a Start command.



The Clear command will terminate the current block transfer, if any is in progress, and turn off the receiver. In addition, the receiver is unlocked and freed to accept data from any transmitter once the next Start command is issued. Issuing a Clear command when a block transfer is in progress can cause data loss.

Automatic Program Load

The MCA can be used for automatic program loading from any other computer in the MCA network. An I/O RESET instruction followed by an **NIOS MCAR** will enable the receiver to accept up to 65,536 words of data from any transmitter beginning at memory location 0. If the computer is equipped with the standard data channel program load option, the following sequence of console operations will enable the receiver to accept program load data from any transmitter and to begin instruction execution at location 377_8:

1. Turn on power to the computer.
2. Press RESET.
3. Set data switches 10-15 to device code 7.
4. Set data switch 0 to 1 to indicate a data channel load.
5. Press PROGRAM LOAD.

Diagnostic Mode

To facilitate maintenance, the MCA can be placed in diagnostic mode with its internal clocks stopped. The MCA logic can be tested under program control by supplying diagnostic control inputs, examining appropriate outputs, and stepping the internal clocks.

The SPECIFY RECEIVER instruction (**DOC**) can be used to place both the transmitter and receiver of an MCA in diagnostic mode. The Pulse command can be used to step the internal clock **CLK** in diagnostic mode. The READ TRANSMITTER STATUS instruction (**DIC**) can be used to determine the state of the phase clock **DATA PHASE** in diagnostic mode.

Typically, an adapter is removed from an MCA network before it is placed in diagnostic mode. Placing the "leftmost" adapter in diagnostic mode without removing it from the network will, in effect, place the entire network in diagnostic mode since all the timing signals within an MCA network are supplied by the "leftmost" adapter. Placing any other adapter in diagnostic mode without removing it from the network will set the Diagnostic Mode indicator in that MCA's transmitter status register to 1, but will not affect the operating mode of the network.

CONSIDERATIONS

Due to delays in the transmitter status logic, the READ TRANSMITTER STATUS instruction (**DIC**) should not be issued for 6 microseconds after the transmitter's Done flag is set to 1. This delay is usually inherent in the interrupt service routine entry procedure.

Both the transmitter address register and the receiver address register are sixteen bits wide. For a standard NOVA 3 computer, the high-order bit in the address registers can be used to select the data channel map to be used in memory references. For a modified NOVA 3 computer with 64K memory, the high-order bit can be used to select the upper or lower 32K of memory to be used in memory references. For all other Data General computers, the high-order bit in the address registers is ignored.

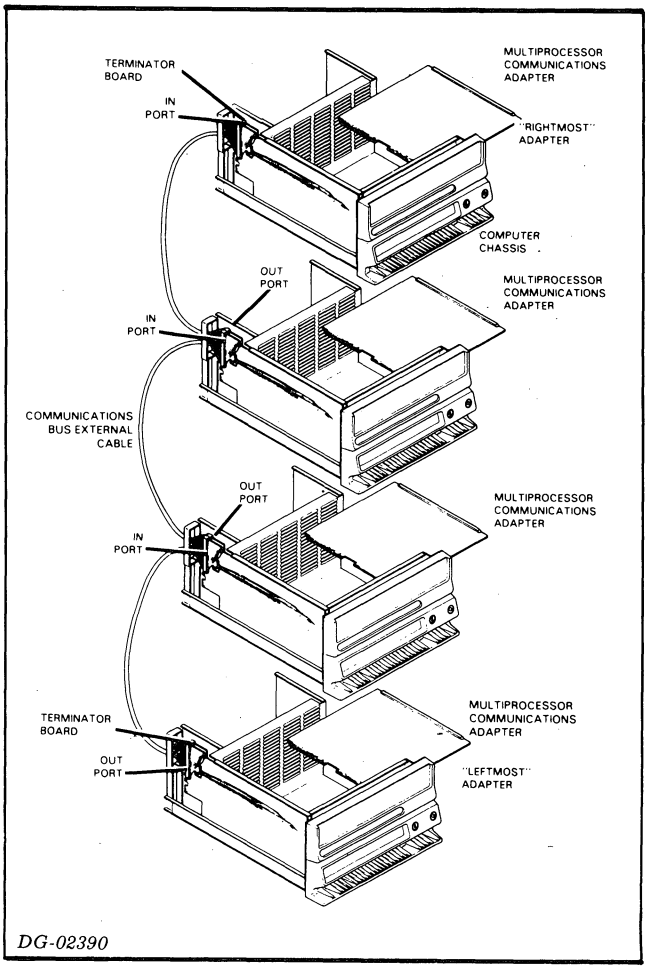
This page intentionally left blank.

SECTION IV
INSTALLATION

INTRODUCTION

An MCA subsystem consists of an adapter for each computer in the network and a common communications bus. The communications bus provides a daisy-chain connection to each adapter, connecting the OUT port of one adapter to the IN port of the next adapter in the chain. The MCA which does not have another MCA connected to its IN port is referred to as the "leftmost" adapter, and the MCA which does not have another MCA connected to its OUT port is referred to as the "rightmost" adapter. Communications bus terminator boards are connected to the IN port of the "leftmost" adapter and the OUT port of the "rightmost" adapter to provide terminator logic for the communications bus.

CAUTION *MCA's from the 4206 series should not be used in the same network with MCA's from the 4038 series.*



There are three situations which can accompany the installation of a multiprocessor communications adapter system:

1. The MCA equipment was purchased together with several computers and other equipment as a complete multiprocessor computer system already installed and shipped in equipment cabinets.

2. An MCA subsystem, including several adapters and communications bus cables, was purchased for on site installation in an existing computer system.

3. An additional MCA was purchased for insertion into an existing MCA subsystem.

In the first case, the complete multiprocessor system will be assembled and tested at the factory and then shipped in the equipment cabinets. In general, such a system needs only to be placed in position and can be operated after all the padding is removed from inside and from around the equipment.

In the second and third cases, however, the on site installation of an MCA subsystem often includes a whole spectrum of steps ranging from straightforward tasks like inserting the adapters in their slots and cabling them together to the more subtle task of assigning system priorities at the hardware level.

Procedures for the on site installation of an MCA subsystem are contained in this chapter and are arranged in an orderly installation sequence. Some sections, notably the section on choosing a slot for the MCA, are meant only to be a guide and cannot give hard, fast answers to questions that will be unique for each MCA subsystem.

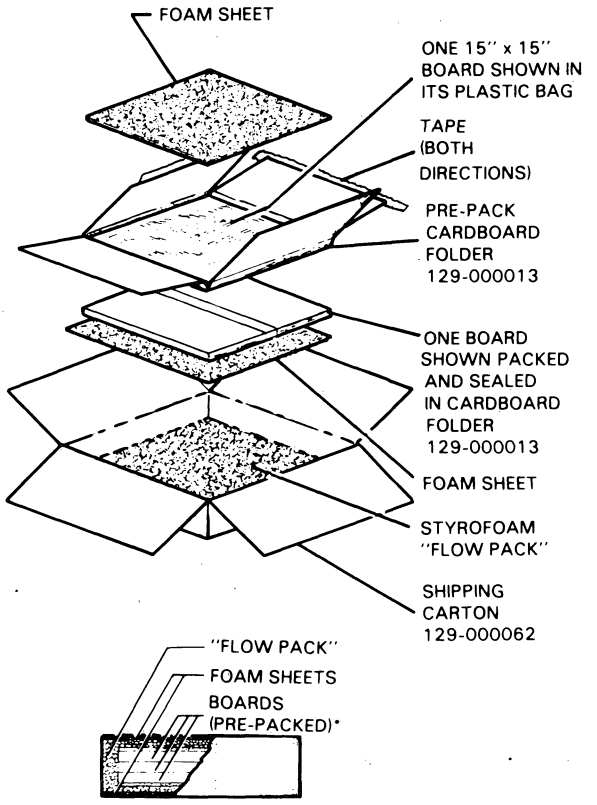
INSTALLING AN MCA SUBSYSTEM

Installing an MCA subsystem involves unpacking the boards and their internal cables, installing jumpers on each board, installing the boards in the computers, and cabling the system together.

Unpacking

NOTE *Do NOT unpack cartons damaged in shipping unless specifically directed to do so by a representative of the carrier or of Data General Corporation.*

The MCA is shipped in the packing kit shown below. Save all the packing materials. To unpack the MCA, slit the sealing tapes of the packing container in a way that does not endanger the printed circuit board inside. Handle the board carefully and particularly avoid flexing it.



MULTIPLE PACKING--UP TO THREE (3) 15 IN.X 15 IN. BOARDS, ENCLOSED IN PLASTIC BAGS AND SEALED IN PRE-PACKED FOLDERS AS SHOWN, CAN BE PUT IN SHIPPING CARTON NUMBER 129-000062. FOR FOUR (4) TO SEVEN (7) BOARDS, USE SHIPPING CARTON NUMBER 129-000012.

Power Requirements

Each MCA draws 3.4 amps at +5 volts dc. Before installing an MCA in a computer chassis, determine the total +5 volt current drawn by the boards in that chassis to ensure that sufficient +5 volt current is available for the MCA. In addition, if the computer chassis has separate fuses for groups of adjacent slots, ensure that the +5 volt current drawn by the boards in the group of slots in which the MCA is placed does not exceed the rating of the fuse for that group of slots.

Choosing a Slot

The MCA can be placed in any available I/O slot of a Data General computer. Its position with respect to other I/O controllers mounted in the computer will, in general, determine the priority of the MCA in the data channel and the program interrupt priority chains. Unless modifications have been made on the backpanel to change the priority structures, highest priority in both the data channel and program interrupt chains is assigned to the controller mounted lowest in the chassis. Each other controller is assigned a lower priority in turn than the controller mounted immediately underneath it.

When boards are installed or removed from the computer chassis, it is important to preserve the integrity of the data channel and program interrupt systems. Memories and other boards that do not use either or both of the enabling signals, $\overline{\text{DCHP}}$ and $\overline{\text{INTP}}$, pass them across their slots with no interference. It is important to note that an open slot can prevent any controllers further along the chain from requesting service. And so the priority signals for both the data channel and the interrupt chains should be jumpered across the empty slots using the wire wrap pins on the backpanel. Empty slots located physically above the top controller of the main chassis do not normally require the jumpers unless an expansion chassis or other I/O bus extension is being used.

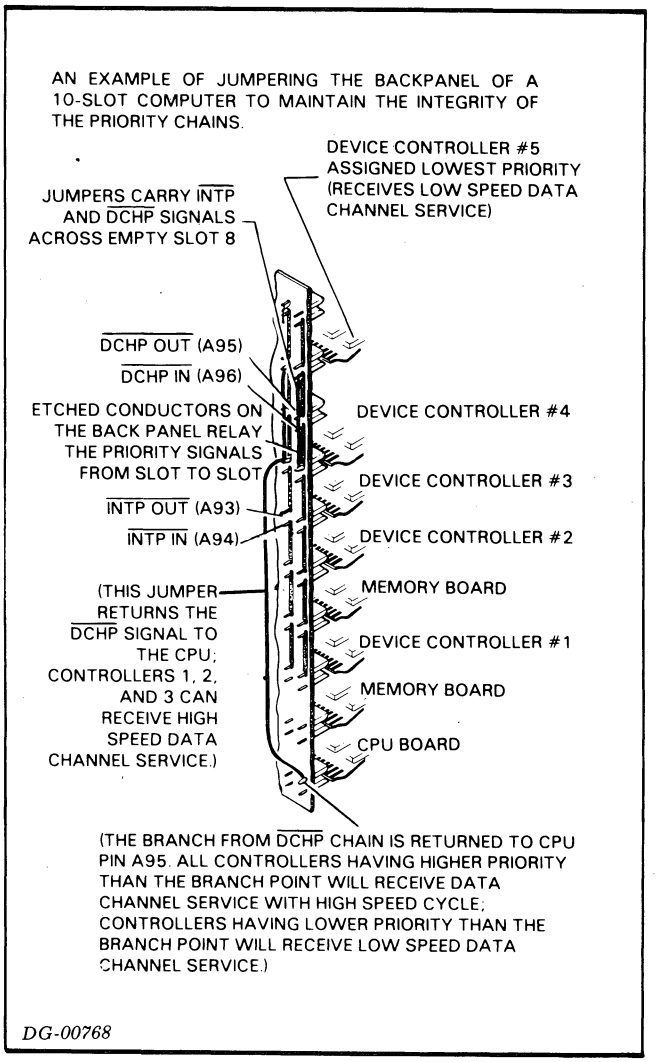
In the NOVA 2, NOVA 3, and NOVA 800 series computers, the enabling signal in the data channel priority chain is used by the central processor to determine whether data channel transfer is to be implemented with a high-speed cycle or a low-speed cycle. A branch in the priority chain returns the signal to the processor, and any controller requesting data channel service that is connected further out the priority chain than the branch will receive low-speed data channel service. Normally the $\overline{\text{DCHP}}$ signal is returned to the processor on CPU pin A95 through the backpanel etch from $\overline{\text{DCHP OUT}}$ of the top slot of the chassis.

The priority signals and their backpanel positions are tabulated below. The following figure shows how the priority chains are implemented.

CAUTION *The pins on the backpanel are fragile and damaging any of them may lead to particularly expensive repairs or replacements. The jumpers should be connected to the backpanel pins by wire wrapping only, and absolutely NOT by soldering. Data General strongly recommends that its field service personnel install all wire wrap connections to the backpanel.*

JUMPER CONNECTIONS FOR EMPTY I/O SLOT

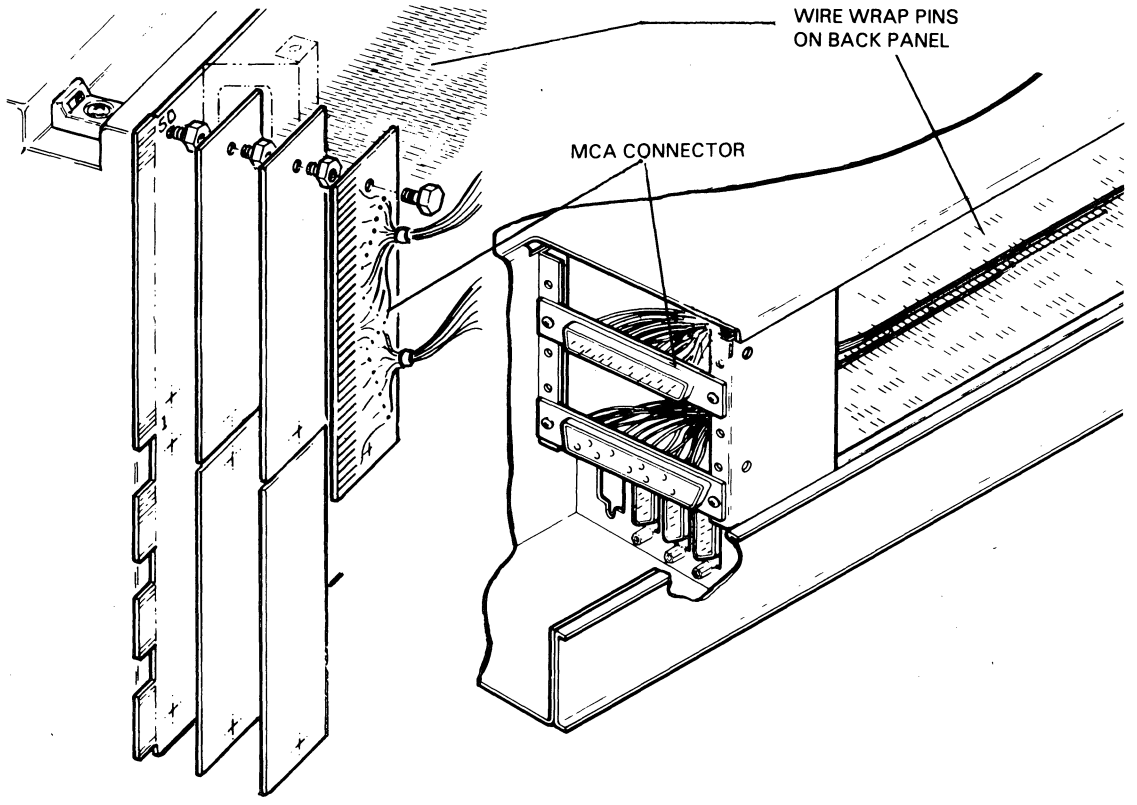
SIGNAL	ON PIN	TO SIGNAL	ON PIN
$\overline{\text{INTP IN}}$	A96	$\overline{\text{INTP OUT}}$	A95
$\overline{\text{DCHP IN}}$	A94	$\overline{\text{DCHP OUT}}$	A93



Internal Cables for the Communications Bus

While the signals from the CPU that are used by the MCA are bussed with etched connectors on the backpanel and are available at each slot, the signals which pass from one MCA to another on the communications bus appear only at isolated pin positions for the slot occupied by the MCA. These signals must be brought to IN port and OUT port communications bus connectors by internal cables. The internal cables consist of wire wrap connections from the designated pins on the backpanel to the communications bus connectors. Certain Data General computers have an internal cable composed of etched conductors; however, this internal cable cannot be used for the MCA. The following tables show the signals that compose the internal cables for the IN port connector and the OUT port connector, and the backpanel pin numbers and edge connector or socket connector pin numbers associated with each signal. The figure below shows the proper method for mounting a communications bus connector at the back of the computer chassis.

CAUTION *The pins on the backpanel are fragile and damaging any of them may lead to particularly expensive repairs or replacements. The internal cable leads should be connected to the backpanel pins by wire wrapping only, and absolutely NOT by soldering. Data General strongly recommends that its field service personnel install all wire wrap connections to the backpanel.*



DG-01942

INTERNAL CABLE CONNECTION FOR MCA IN PORT

Signal Name	Backpanel Pin Number	Edge Connector Pin Number	Socket Connector Pin Number
GND	A1	A	1
+V TERM	A89	C	3
+V TERM	A90	D	4
+V TERM	A91	E	5
RETURN	A92	F	6
INACTIVE	A61	H	7
GND	A2	J	8
$\overline{\text{CPU STROBE}}$	A49	K	9
GND	A99	L	10
$\overline{\text{CPU PHASE}}$	A47	M	11
GND	A100	N	12
SPARE	A63	P	13
REQP	A59	R	14
GND	B1	S	15
GND	B2	T	16
$\overline{\text{CPU 12}}$	A65	U	17
$\overline{\text{CPU 0}}$	A67	V	18
$\overline{\text{CPU 4}}$	A69	W	19
$\overline{\text{CPU 8}}$	A71	X	20
$\overline{\text{CPU 9}}$	A73	Y	21
$\overline{\text{CPU 5}}$	A75	Z	22
$\overline{\text{CPU 6}}$	A76	a	23
GND	B100	b	24
$\overline{\text{CPU 1}}$	A77	c	25
$\overline{\text{CPU 2}}$	A78	d	26
$\overline{\text{CPU 10}}$	A79	e	27
$\overline{\text{CPU 13}}$	A81	f	28
$\overline{\text{CPU 7}}$	A83	h	29
$\overline{\text{CPU 3}}$	A84	j	30
$\overline{\text{CPU 15}}$	A85	k	31
$\overline{\text{CPU 11}}$	A86	l	32
$\overline{\text{CPU 14}}$	A88	m	33
$\overline{\text{S-CPU 0}}$	B11	p	35
$\overline{\text{S-CPU 3}}$	B13	r	36
$\overline{\text{S-CPU 1}}$	B15	s	37
$\overline{\text{S-CPU 2}}$	B23	t	38
$\overline{\text{S-CPU 15}}$	B25	u	39
$\overline{\text{S-CPU 12}}$	B27	v	40
$\overline{\text{S-CPU 13}}$	B34	w	41
$\overline{\text{S-CPU 6}}$	B36	x	42
$\overline{\text{S-CPU 7}}$	B38	y	43
$\overline{\text{S-CPU 14}}$	B40	z	44
GND	B48	AA	45
$\overline{\text{S-CPU 4}}$	B52	AB	46
$\overline{\text{S-CPU 5}}$	B54	AC	47
GND	B50	AD	48
GND	B99	AF	50

INTERNAL CABLE CONNECTION FOR MCA OUT PORT

Signal Name	Backpanel Pin Number	Edge Connector Pin Number	Socket Connector Pin Number
GND	A1	A	1
+V TERM	A89	C	3
+V TERM	A90	D	4
+V TERM	A91	E	5
RETURN	A92	F	6
INACTIVE	A61	H	7
GND	A2	J	8
$\overline{\text{CPU STROBE}}$	A49	K	9
GND	A99	L	10
$\overline{\text{CPU PHASE}}$	A47	M	11
GND	A100	N	12
SPARE	A63	P	13
REQP OUT	A57	R	14
GND	B1	S	15
GND	B2	T	16
$\overline{\text{CPU 12}}$	A65	U	17
$\overline{\text{CPU 0}}$	A67	V	18
$\overline{\text{CPU 4}}$	A69	W	19
$\overline{\text{CPU 8}}$	A71	X	20
$\overline{\text{CPU 9}}$	A73	Y	21
$\overline{\text{CPU 5}}$	A75	Z	22
$\overline{\text{CPU 6}}$	A76	a	23
GND	B100	b	24
$\overline{\text{CPU 1}}$	A77	c	25
$\overline{\text{CPU 2}}$	A78	d	26
$\overline{\text{CPU 10}}$	A79	e	27
$\overline{\text{CPU 13}}$	A81	f	28
$\overline{\text{CPU 7}}$	A83	h	29
$\overline{\text{CPU 3}}$	A84	j	30
$\overline{\text{CPU 15}}$	A85	k	31
$\overline{\text{CPU 11}}$	A86	l	32
$\overline{\text{CPU 14}}$	A88	m	33
$\overline{\text{S-CPU 0}}$	B11	p	35
$\overline{\text{S-CPU 3}}$	B13	r	36
$\overline{\text{S-CPU 1}}$	B15	s	37
$\overline{\text{S-CPU 2}}$	B23	t	38
$\overline{\text{S-CPU 15}}$	B25	u	39
$\overline{\text{S-CPU 12}}$	B27	v	40
$\overline{\text{S-CPU 13}}$	B34	w	41
$\overline{\text{S-CPU 6}}$	B36	x	42
$\overline{\text{S-CPU 7}}$	B38	y	43
$\overline{\text{S-CPU 14}}$	B40	z	44
GND	B48	AA	45
$\overline{\text{S-CPU 4}}$	B52	AB	46
$\overline{\text{S-CPU 5}}$	B54	AC	47
GND	B50	AD	48
GND	B99	AF	50

Jumper Locations

The illustration below shows the location of the various jumpers on the MCA board.

Device Select Jumpers

Jumpers are used to select the device codes for the MCA. Device codes 6 and 7 are used for the first MCA in a computer. The secondary device codes 46_8 and 47_8 are used when a second MCA is installed in the computer. Insert the jumpers as follows:

DEVICE CODE JUMPERS

DEVICE CODE	INSERT JUMPERS
6/7 (MCAT/MCAR) 46/47 (MCAT1/MCAR)	W1 W2,W17

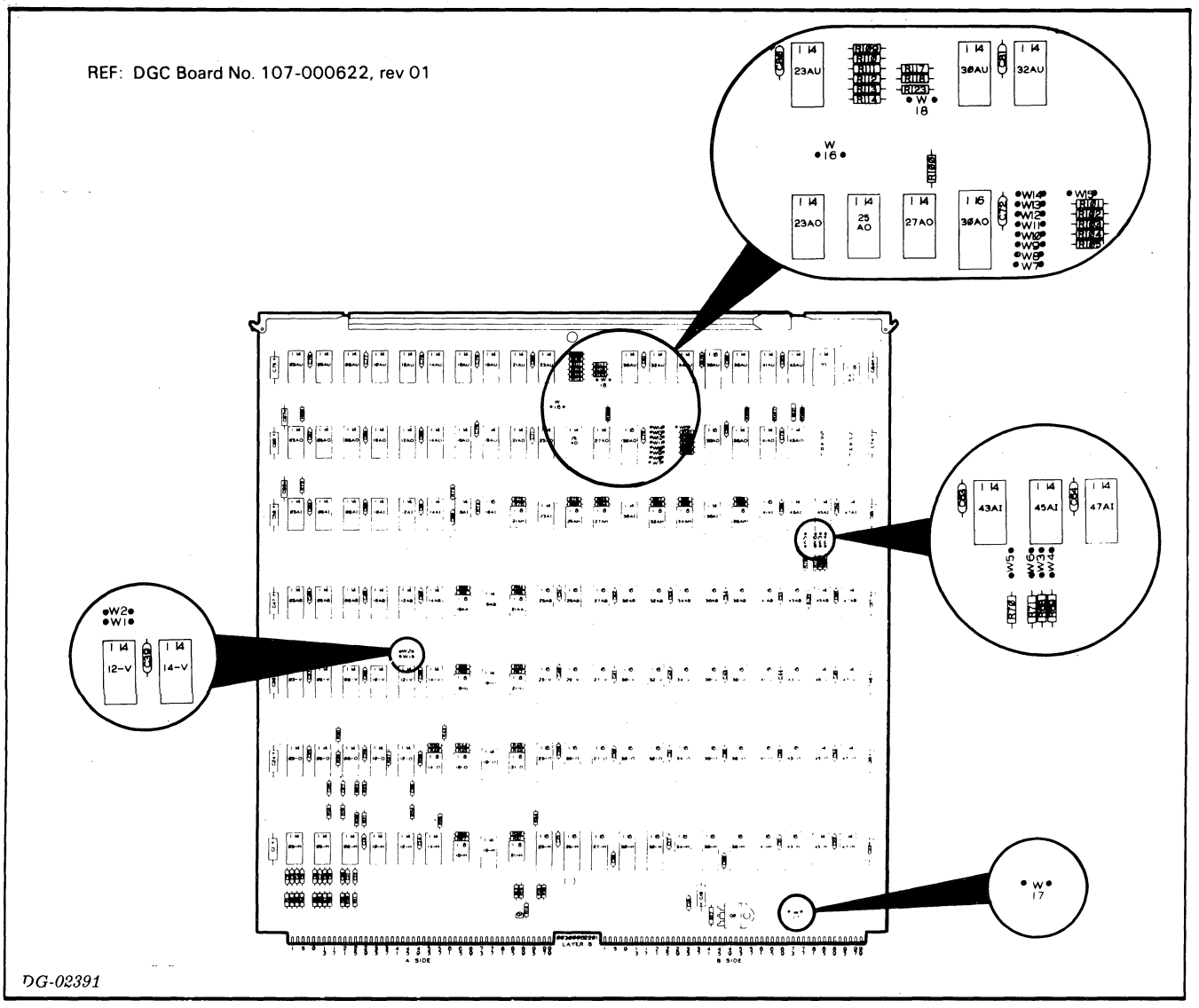
Identifying Number Jumpers

Jumpers are used to select the identifying number which is used to distinguish this MCA and associated computer in communications with other computers in the MCA network. Four jumpers are used to specify the binary representation of the selected identifying number. Insert the jumpers as follows:

IDENTIFYING NUMBER JUMPERS

BIT POSITIONS OF IDENTIFYING NUMBER	0	1	2	3
INSERT JUMPER TO SPECIFY 1	W5	W6	W3	W4

NOTE *Although 0 is a valid identifying number within an MCA network, 0 is not a valid identifying number with Data General's Real-time Disc Operating System.*



Operating Mode Jumpers

Jumpers on the "leftmost" adapter are used to select whether the MCA network operates in normal mode with a maximum data transfer rate of 312,500 words per second or in fast mode with a maximum data transfer rate of 500,000 words per second. An MCA network operating in fast mode can contain a maximum of four computers. The operating mode jumpers on all but the "leftmost" adapters are used to select the operating mode of the adapter when it is removed from the MCA network and placed in diagnostic mode. Insert the jumpers as follows:

OPERATING MODE JUMPERS

MODE OF OPERATION	INSERT JUMPERS
NORMAL MODE	W8, W10, W12, W14, W15
FAST MODE	W7, W9, W11, W13

"Leftmost" Processor Jumper

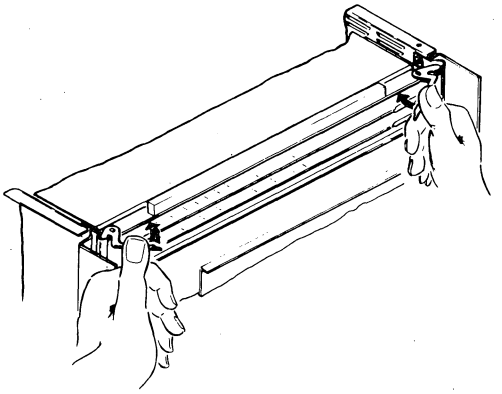
The "leftmost" processor jumper, W16, disables the internal clocks of the MCA. Insert jumper W16 in all the adapters in the network except the "leftmost" adapter. Omit jumper W16 on the "leftmost" adapter.

"Rightmost" Processor Jumper

The "rightmost" processor jumper, W18, disables the return of communications bus priority to the "leftmost" adapter from all adapters in the network except the "rightmost" adapter. Insert jumper W18 in all adapters in the network except the "rightmost" adapter. Omit jumper W18 on the "rightmost" adapter.

Inserting the MCA in its Slot

The MCA is placed in the slot by carefully sliding it into the guides on each side of the chassis. It should slide smoothly and not bind. Use the lock tabs to provide the leverage needed to seat the circuit board connectors completely as shown below.



DG-00626

CABLING AN MCA SUBSYSTEM

Cabling an MCA subsystem consists of installing a series of communications bus cables from one computer to the next in a network. The communications bus cables connect the OUT port connector of one adapter to the IN port connector of the next adapter, starting with the OUT port of the "leftmost" adapter in the network and ending with the IN port of the "rightmost" adapter in the network.

There are three types of communications bus cables available, where the choice of communications bus cable is governed by the types of the two computers to which the communications cable will be attached. Consult the table of ordering information in Appendix A to order the correct cables.

Installing Communications Bus Terminator Boards

To provide the necessary terminator logic for the communications bus, communications bus terminator boards are plugged onto both the IN port connector of the "leftmost" adapter and the OUT port connector of the "rightmost" adapter in the MCA network. The figure below shows the proper method for attaching a terminator board to a communications bus internal cable connector.

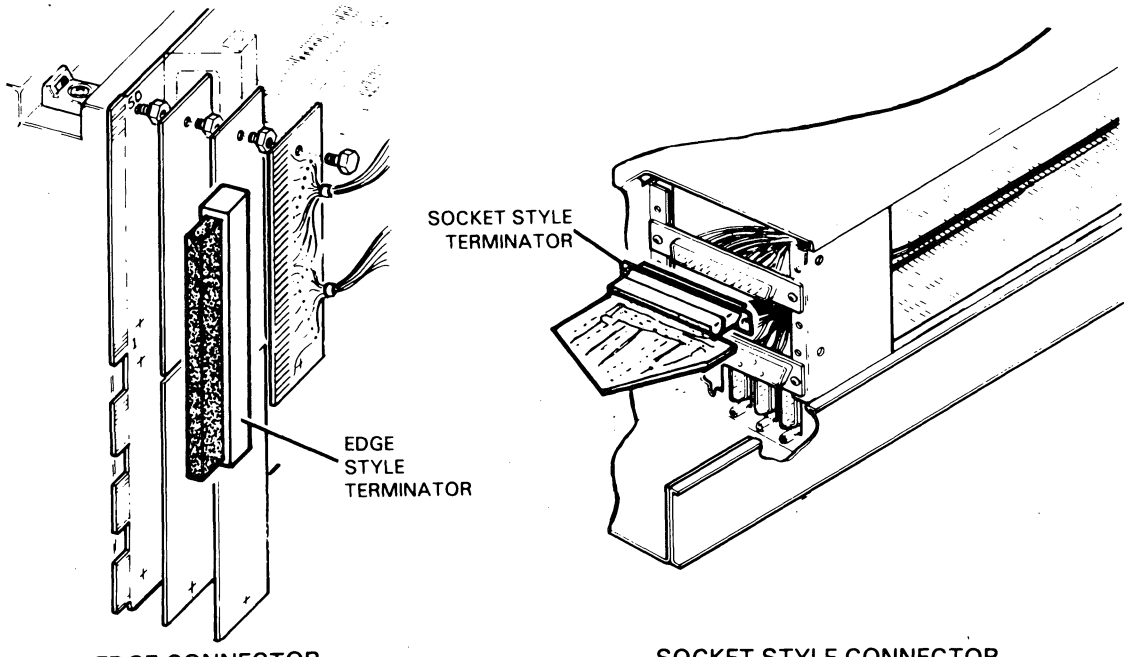
SYSTEM TESTING

A complete reliability and testing program is included with the documentation accompanying the MCA. Before running the diagnostic program, the particular MCA to be tested should be disconnected from the MCA network. The "leftmost" processor jumper, W16, should be removed to enable the internal clocks of the MCA. In addition, the communications bus terminator board should be plugged onto either the IN port or OUT port connector of the MCA as shown in the figure below.

The table below lists the reliability and diagnostic programs provided and the part numbers associated with the absolute binary tapes and the listings of the programs.

DIAGNOSTIC ROUTINES

NAME	PART NUMBER ABSOLUTE BINARY TAPE	PART NUMBER LISTING
4206 MCA RELIABILITY	095-342	095-342
4206 MCA DIAGNOSTIC	095-343	095-343



DG-02392

APPENDIX A
ORDERING INFORMATION

The table below summarizes model numbers and information pertinent to ordering the various components of an MCA subsystem.

4206 SERIES MCA SUBSYSTEM

MODEL	DESCRIPTION
4206	Multiprocessor communications adapter and communications bus terminator board.
1106-AA	Multiprocessor communications adapter cable for connecting two 4206 series MCA's installed in NOVA, SUPERNOVA, NOVA 800, 830, 840, or 1200 computers.
1106-AB	Multiprocessor communications adapter cable for connecting a 4206 series MCA installed in a NOVA, SUPERNOVA, NOVA 800, 830, 840, or 1200 computer and a 4206 series MCA installed in an ECLIPSE, NOVA 2, 3, 1210, 1220, or 820 computer.
1106-BB	Multiprocessor communications adapter cable for connecting two 4206 series MCA's installed in ECLIPSE, NOVA 2, 3, 1210, 1220, or 820 computers.

This page intentionally left blank.

READERS COMMENT FORM

DOCUMENT TITLE: .....

Your comments, accompanied by answers to the following questions, help us improve the quality and usefulness of our publications. If your answer to a question is "no" or requires qualification, please explain.

How did you use this publication?

( ) As an introduction to the subject.
( ) As an aid for advanced knowledge.
( ) For information about operating procedures.
( ) To instruct in a class.
( ) As a student in a class.
( ) As a reference manual.
( ) Other.....

Did you find the material:

- Useful..... YES ( ) NO ( )
- Complete..... YES ( ) NO ( )
- Accurate..... YES ( ) NO ( )
- Well organized..... YES ( ) NO ( )
- Well written..... YES ( ) NO ( )
- Well illustrated..... YES ( ) NO ( )
- Well indexed..... YES ( ) NO ( )
- Easy to read..... YES ( ) NO ( )
- Easy to understand..... YES ( ) NO ( )

We would appreciate any other comments; please label each comment as an addition, deletion, change, or error and reference page numbers where applicable.

COMMENTS

PAGE	COL	PARA	LINE	FROM	TO

From:

NAME..... TITLE.....
FIRM..... DIV.....
ADDRESS.....
CITY..... STATE..... ZIP.....
TELEPHONE..... DATE.....

Data General Corporation
ENGINEERING PUBLICATIONS
COMMENT FORM
DG-00935

CUT ALONG DOTTED LINE

FOLD DOWN FIRST FOLD DOWN

FIRST CLASS
PERMIT NO. 26
SOUTHBORO
MASS. 01772

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL

Postage will be paid by:

DataGeneral
Southboro, Massachusetts 01772

ATTENTION: Engineering Publications

FOLD UP SECOND FOLD UP

STAPLE