DataGeneral

Technical Reference

4240 INTER-PROCESSOR BUS

014-000056-01

Technical Reference

4240 INTER-PROCESSOR BUS

014-000056-01

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 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** and **ECLIPSE** are registered trademarks of Data General Corporation, Westboro, Massachusetts. **DASHER** is a trademark of Data General Corporation, Westboro, Massachusetts.

Ordering No. 014-000056

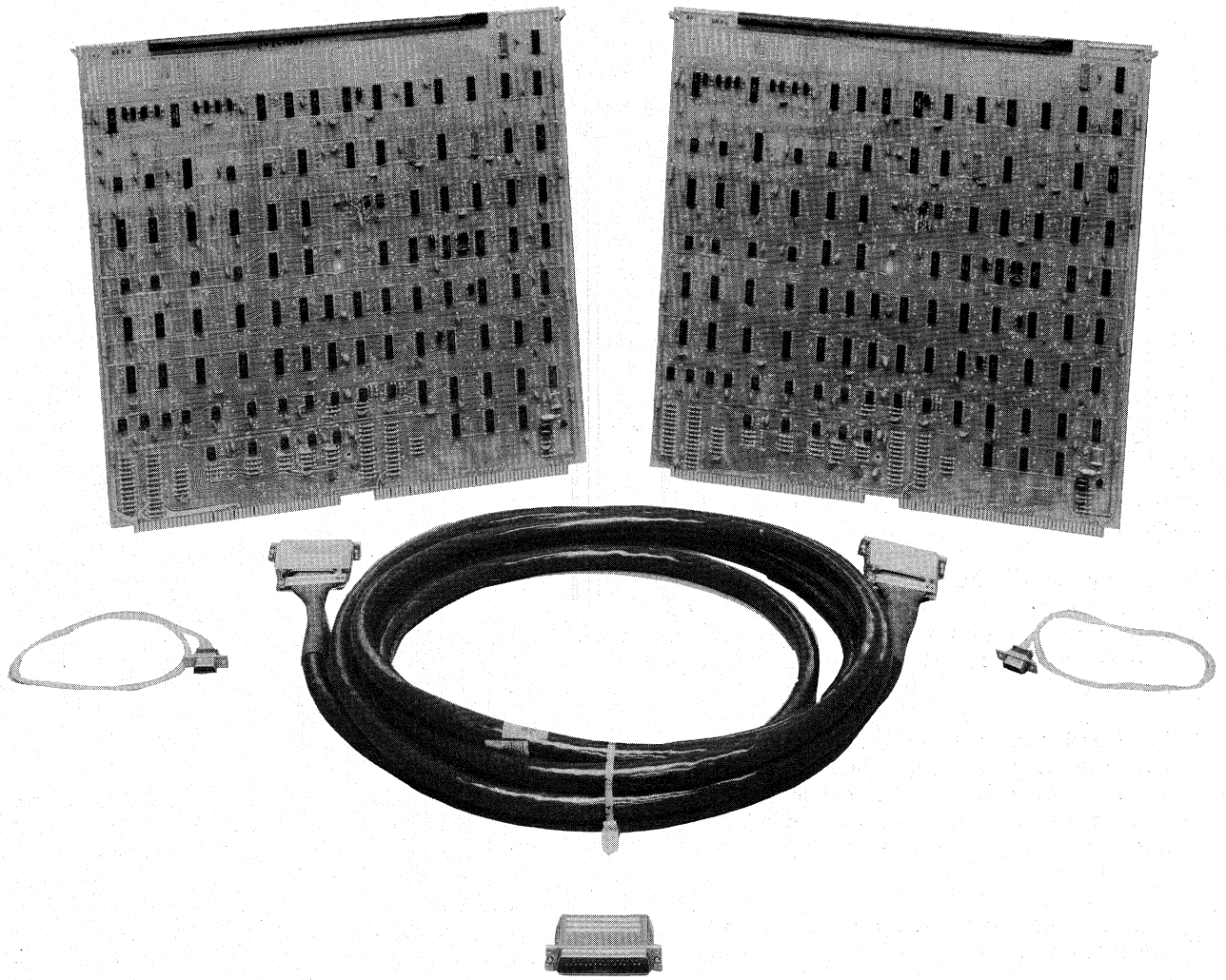
Printed in the United States of America
Rev. 01, August 1977

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

This page intentionally left blank.

INTER-PROCESSOR BUS PRINTED CIRCUIT BOARDS, EXTERNAL CABLE, INTERNAL CABLES, AND TEST PLUG.



CHAPTER 1
OVERVIEW

INTRODUCTION

The Data General Corporation 4240 Inter-Processor Bus (IPB) subsystem provides the facilities for asynchronous communications between any two NOVA or ECLIPSE line computers without using an intermediate storage device. The IPB is under the direct program control of both computers and provides the computers with the following distinct capabilities:

- Bi-directional, full-duplex communications between computers. This feature allows dual processor communication to be programmed in a manner similar to full-duplex communication with a Teletype® .

- Bi-directional, half-duplex communications between computers, which may be *interlocked* with suitable software. The interlock feature allows only one processor to transmit data when the appropriate software protocol is honored by both computers.

- Program interrupt request in either computer when the other computer:

    1) fails to perform as expected; or,

    2) issues a signal indicating a probable *power-fail* condition. This capability is implemented by a facility called the watchdog timer.

The IPB is useful in applications where direct program control of transferred information is needed and in applications where information must be transferred quickly between computers, but neither the amount of information to be transferred nor the speed with which it is to be transferred requires the use of the DGC Multiprocessor Communications Adapter. Using routines solely dedicated to transmitting and receiving information, two NOVA or ECLIPSE line computers can achieve transfer rates exceeding 100,000 16-bit words per second, using the IPB. During normal program operation the information transfer rate is limited by the length of the interrupt handling routine in each computer.

Typical applications of the IPB include: shared disc environments, parallel processing arrangements, communications concentration systems, and back-up systems. In a shared disc environment, two computers share a common system's disc pack. The *interlock* feature of the half-duplex communications link provides a means of communicating disc space reservations and actions performed on disc files.

In a parallel processing arrangement, the IPB allows the two computers to exchange data rapidly, providing a check on the results of computation. In a communications concentration system, one computer is dedicated to the handling of a large number of I/O communications to and from the system, freeing the other computer to perform calculations. The IPB allows a rapid exchange of information, under program control, in such a system. In a back-up system, computer failure is guarded against by having a back-up computer ready to assume the duties of the failing computer. The watchdog timer facility of the IPB can be used to notify the back-up computer of the failure of the other computer.

The applications mentioned above are implemented in the software of the two computers, using the hardware available in the IPB. This reference gives a brief description of the functioning of the IPB hardware and presents possible schemes of programming the computers, to implement the functions of the IPB. The design of the IPB hardware is flexible enough to allow programming schemes other than those described in this reference to alter the IPB functions.

Teletype® is a registered trademark of the Teletype Corporation, Skokie, Illinois. All references to Teletypes in this manual shall apply to this mark.

ARCHITECTURE

The IPB Communications link between two NOVA or ECLIPSE line computers consists of two 15-inch square IPB printed circuit boards, one in each computer, connected by a 15-foot cable. Each IPB board contains four separate devices: a 16-bit full-duplex (non-interlocked) transmitter, a 16-bit full-duplex (non-interlocked) receiver, a 16-bit half-duplex (interlocked) transmitter/receiver, and a watchdog timer.

The block diagram of an IPB communications link illustrates the operation of the IPB in terms of data and signal flow. Two features basic to the understanding of the IPB will be discussed in the following paragraphs. These features are the independent transfer of data and status information and the distinction between the *left* and *right* IPB board.

Two types of information are (independently) transferred through the IPB, data words and status information. Data words are transferred to and from accumulators in the two CPUs by the appropriate instructions. The data paths in the IPB are shown in the upper part of the block diagram. Data enters from the DATA[0-15] lines of the I/O bus of one computer, passes through a transmitter into the CPB[0-15] lines between the two IPB boards, and finally passes through a receiver onto the DATA[0-15] lines of the I/O bus of the other computer. A 16-bit storage buffer is associated with each transmitter and receiver.

Status information tells whether a computer has transmitted a data word, received a data word, experienced a failure, etc. Each device in the IPB has its own flags (Busy and Done) which can interact with the programs of the two computers to transfer status information. Flag states are affected by commands from both computers and by the status of flags in both IPB boards. A computer receives status information by means of the program interrupt facility or by reading the flags. The block diagram illustrates the hardware separation of data and status information; all signals used in data transfer are in the upper part of the figure, all signals used in status transfer are in the lower part. The hardware independence of these two types of information transfer give a degree of flexibility to the programming of the IPB.

The IPB boards in each computer are identical, but they do not have identical roles in the transmission of information. The wiring in the IPB external cable assigns the IPB board in one computer (at the end of the cable labeled *right*) to serve as a data storage area and the other board (at the *left* end of the cable) to serve only in a gating capacity in the transmission of data in both full- and half-duplex communications. This means that in any data transfer, the transmitting computer writes data into one of the four buffers in the *right* IPB board and the receiving computer reads the data from that buffer. The four data storage buffers in the right IPB board latch data for the different types of data transmission in the IPB as follows:

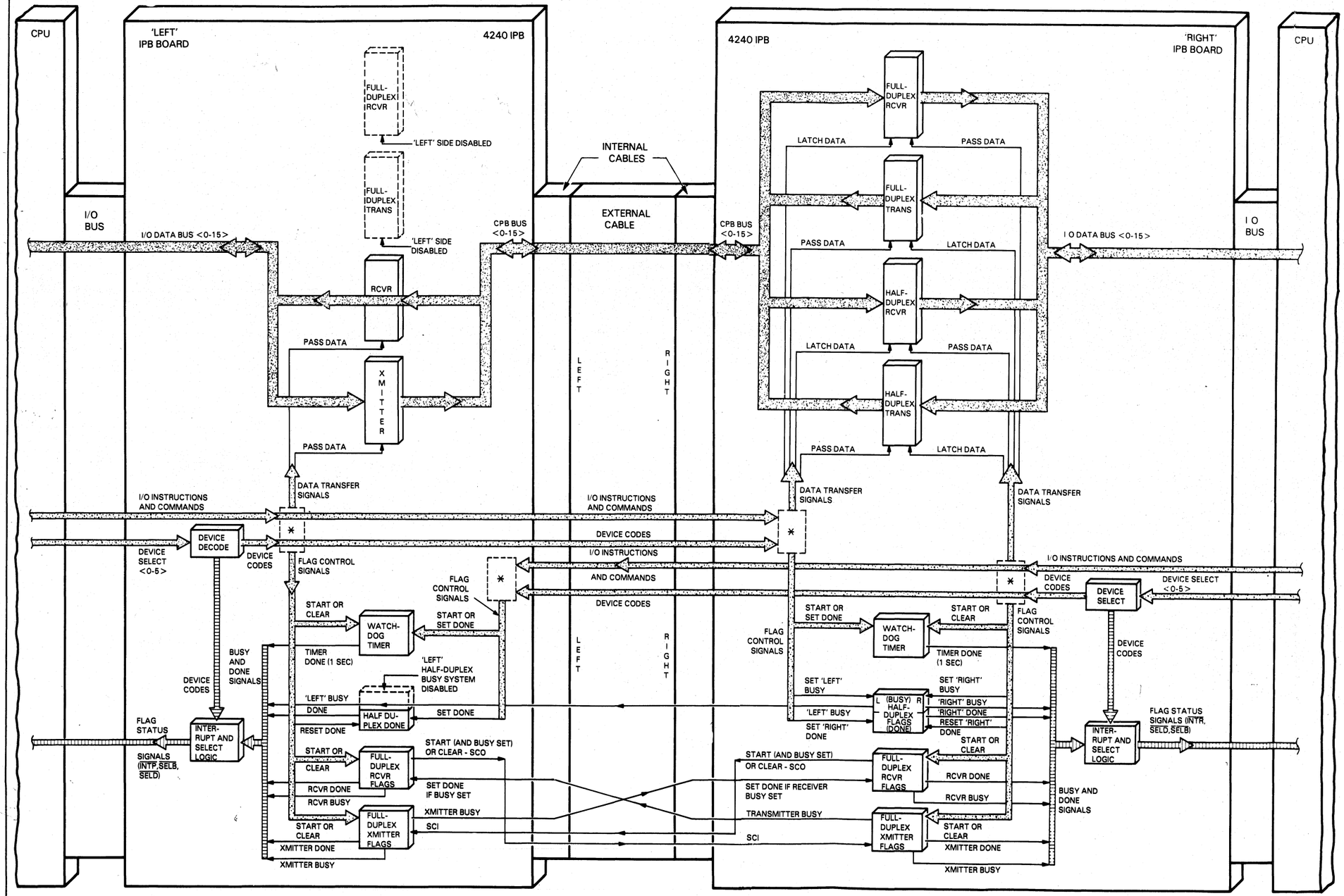
- The buffers labeled full-duplex latch data transmitted in the full-duplex communications link.
- The buffers labeled half-duplex latch data transmitted in the half-duplex communications link.
- The receiver buffers latch data transmitted by the *left* computer to the *right* computer.
- The transmitter buffers latch data transmitted by the *right* computer to the *left* computer.

Since the *left* IPB board is meant to serve only in a gating capacity in data transfers, only one receiver and one transmitter are necessary; the full-duplex transmitter and receiver that would be used if the board were on the *right* side of the cable are disabled.

Another distinction between the *left* and *right* IPB boards is in the half-duplex Busy flags. The *right* IPB board has a half-duplex Busy flag system containing the half-duplex Busy flags for both *left* and *right* computers. The corresponding logic on the *left* IPB is disabled.

The distinction between *right* and *left* IPB boards is a design distinction, allowing the two IPB boards to be identical, and does not effect the programming, operation, or communications capabilities of either computer in the system.

IPB COMMUNICATIONS LINK BLOCK DIAGRAM



DG-00826

The IPB transfers data using signals derived from the I/O instructions (top). Status information is transferred through each device by its flags (bottom) which respond to signals originating in both computers and the flags in both IPB boards. The external cable determines which board is the *left* board and which is the *right* board. The *right* board stores data passing between the two computers and contains the half-duplex Busy flags for both IPB boards.

SPECIFICATIONS

4240 Inter-Processor Bus

Size:	15" X 15"
Power Requirements:	Approximately 2.5 amps, 5Vdc
Space Requirements:	1 slot in each computer
Items Supplied on Purchase:	1 15" X 15" IPB Printed Circuit Board 1 Internal Cable 1 Test Plug 1 Documentation Package

1065 External Cable for the 4240 Inter-Processor Bus

(This item must be purchased separately; specify the two computer models.)

Length:	15 feet
Contents:	78 lines 39 signal levels 39 grounds

ORDERING INFORMATION

When purchasing the 4240 Inter-Processor Bus, the DGC computer model using the IPB must be specified in order for the correct internal cable and test plug to be supplied. When purchasing an external cable for the 4240 IPB, the models of the two DGC computers to be linked must be specified. If an item normally supplied with the 4240 IPB must be purchased separately, i.e., an internal cable or test plug, the order should specify the computer with which it is to be used and the fact that the item is to be used with the 4240 Inter-Processor Bus.

DGC PART NUMBERS

Item	DGC Computers	DGC Part Number
4240 IPB	All	005-001961
Internal Cable	NOVA 820, 1210, 1200	005-000901
	NOVA 2	005-001802
	NOVA 800, 840, 1200, 1230	005-001965
Test Plug	NOVA 800, 840, 1200, 1230	005-001963
	NOVA 820, 1210, 1230	
	NOVA 2	005-001964
Documentation Package for 4240 IPB	All (Specify computer to receive correct wire lists.)	005-001969
External Cable	Group A to Group A*	005-001966
	Group A to Group B**	005-001967
	Group B to Group B**	005-001968

* Group A computers are NOVA 800, 830, 840, 1200 and 1230.

** Group B computers are NOVA 820, 1210, 1220, NOVA 2, and ECLIPSE line.

CHAPTER II
PROGRAMMING

The Data General Corporation 4240 Inter-Processor Bus (IPB) allows any two NOVA or ECLIPSE line computers to transfer information back and forth without using an intermediate storage device. Information transfer is accomplished via programmed I/O in either full- or half-duplex communications. In addition, the IPB provides a *watchdog timer* that will interrupt one processor if the other processor stops functioning.

The IPB is useful in applications where information must be transferred quickly between processors, but neither the amount of information to be transferred nor the speed with which it is to be transferred requires the use of the Multiprocessor Communications Adapter. The IPB is also useful in applications such as a shared disc environment where agreements must be made as to which processor has control over which files at any one point in time. The watchdog timer facility of the IPB allows this device to be used in applications where one processor must know about the failure of the other processor.

The IPB is made up of four separate devices for each computer: a 16-bit transmitter, a 16-bit receiver, a 16-bit transmitter/receiver, and a watchdog timer. The 16-bit transmitter and receiver operate together to provide a full-duplex communications link between the processors. This is functionally equivalent to a Teletype, but is much faster. The only limit to transmission speed is the speed of the I/O interrupt handler. The 16-bit transmitter/receiver provides a half-duplex communication link between the processors. This link also provides a mechanism whereby the two processors can ensure that each one knows who is *master* of the system at any time. The watchdog timer is a one-second timer that allows either processor to be informed of a hardware or software failure in the other processor.

FULL-DUPLEX COMMUNICATIONS

Full-duplex communication between processors is accomplished with four devices: two transmitters and two receivers. Each processor in the system has a transmitter/receiver pair. Associated with each receiver is a receiver buffer. These devices are asynchronous and correspond functionally to the transmitter and receiver associated with a Teletype. Each computer looks like a high-speed Teletype to the other computer.

If there were not some way to prevent it, it would be possible for one computer to transmit words faster than the other computer could receive them. So that this problem will not occur, the interaction between the transmitter of one computer and the receiver of the other computer is similar to the interaction between a computer and a Teletype controller. When one computer transmits a word, the word is placed in the holding buffer of the other computer's receiver. The transmitting computer is not signaled that the transmission is complete until the other computer has read the word from its holding buffer into an accumulator.

Another possible problem that could occur is that a word could be lost due to the fact that the receiving computer has no way to know whether or not the other computer has already transmitted a word. Again, the interaction of the DPI Busy and Done flags of one computer with the DPO Busy and Done flags of the other computer ensures that no data will be lost due to this uncertainty. The following paragraphs describe a typical information transfer between two computers. The computer *transmitting* information is *computer B* and the computer *receiving* information is *computer A*.

Before the transfer begins, computer A has no way of knowing whether or not his receiver (DPI) buffer contains meaningful information. Therefore, computer A issues an INITIATE RECEIVER instruction (NIOS DPI). The DPI Busy and Done flags of computer A will be set to 1 and 0, respectively. If computer B has not already transmitted a word, these flags will remain in this state until computer B transmits a word. When computer B finally does transmit a word, the DPI Done flag of computer A will be set to 1, signaling computer A that there is a meaningful word in its DPI buffer waiting to be read. If computer B has already transmitted a word, the DPI Done flag of computer A will be set to 1 immediately after computer A issues the INITIATE RECEIVER instruction and computer A will know that there is a meaningful word in its DPI buffer waiting to be read. Note that in the above sequence, computer A's DPI Busy flag has not been set to 0; it is still 1.

PROGRAMMING SUMMARY

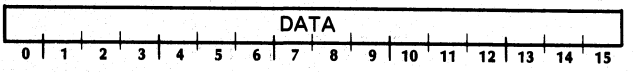
FULL-DUPLEX

Mnemonic (Input)	DPI
Device Code (Input)	40_8
Priority Mask Bit (Input)	8
Mnemonic (Output)	DPO
Device Code (Output)	41_8
Priority Mask Bit (Output)	8

ACCUMULATOR FORMATS

READ DATA **(DIA)**

WRITE DATA **(DOA)**



S, C, AND P FUNCTIONS

Input

f = S In the computer executing the instruction, the DPI Busy flag is set to 1, and the DPI Done flag is set to 0. If the DPI Busy flag was 1 before this instruction was issued, then, in the other computer, the DPO Busy flag is set to 0, and the DPO Done flag is set to the prior value of DPO Busy flag.

f = C In the computer executing the instruction, the DPI Busy and Done flags are both set to 0. In the other computer, the DPO Busy flag is set to 0 and the DPO Done flag is set to the prior value of the DPO Busy flag.

f = P This command has no effect.

Output

f = S In the computer executing the instruction, the DPO Busy flag is set to 1, and the DPO Done flag is set to 0. In the other computer, if the DPI Busy flag is 1, the DPI Done flag is set to 1 and the DPI Busy flag remains unchanged.

f = C In the computer executing the instruction, the DPO Busy and Done flags are both set to 0.

f = P This command has no effect.

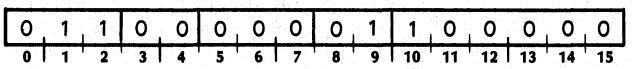
Upon learning that there is a word in its DPI buffer, computer A issues a READ DATA instruction with a Clear (DIAC *ac*, DPI). The word will be read into computer A's specified AC, and the DPI Busy and Done flags will both be set to 0. The DPO Busy and Done flags of computer B will be set to 0 and 1, respectively, and computer B will know that computer A received the word.

Computer B then issues a WRITE DATA instruction with a Start (DOAS *ac*, DPO). this instruction transfers the word to computer A's DPI buffer and sets computer B's DPO Busy and Done flags to 1 and 0, respectively. When computer A is ready to receive another word, it issues another INITIATE RECEIVER instruction, and the cycle repeats. In this way, data will not be lost due to computer B transferring data so fast that computer A cannot handle it or to uncertainty.

INSTRUCTIONS

Initiate Receiver

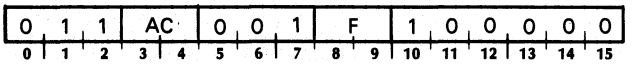
NIOS DPI



In the computer executing the instruction, the DPI Busy and Done flags are set to 1 and 0, respectively. If the DPO Busy flag of the other computer is 1, the DPI Done flag of the computer executing the instruction is set to 1 immediately. If the DPO Busy flag of the other computer is 0, the DPI Busy and Done flags of the computer executing the instruction will remain set to 1 and 0 until the DPO Busy flag of the other computer is set to 1.

Read Data

DIA *[f]* *ac*, DPI

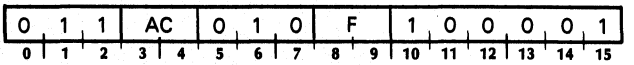


The data contained in the DPI buffer of the computer executing the instruction is placed in bits 0-15 of its specified AC. The previous contents of the specified AC are lost. The contents of the DPI buffer remain unchanged. After the data transfer, the Busy and Done flags are set according to the function specified by f.

NOTE *A series of DIAS instructions may be used to read a multiple word message.*

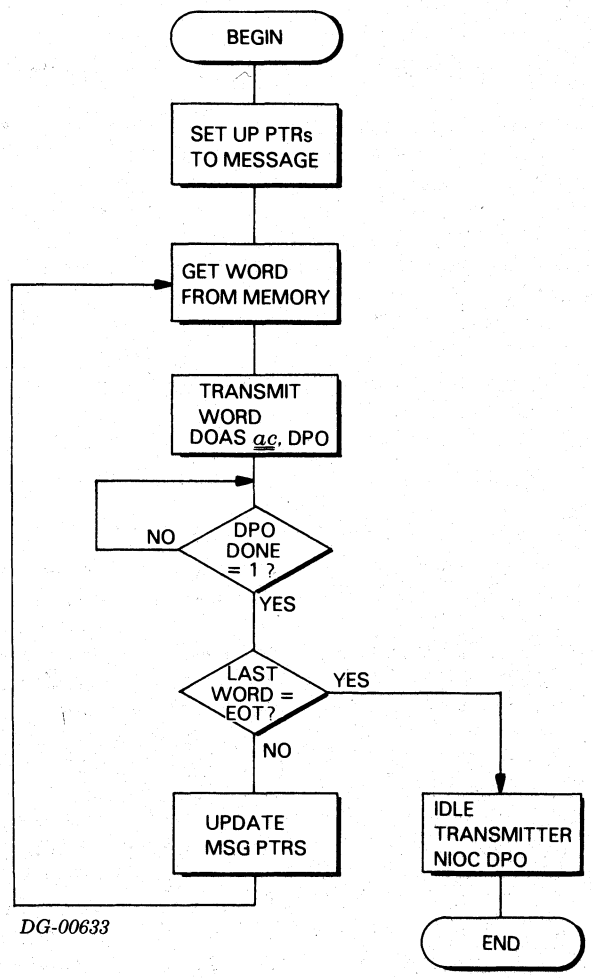
Write Data

DOA *[f]* *ac*, DPO

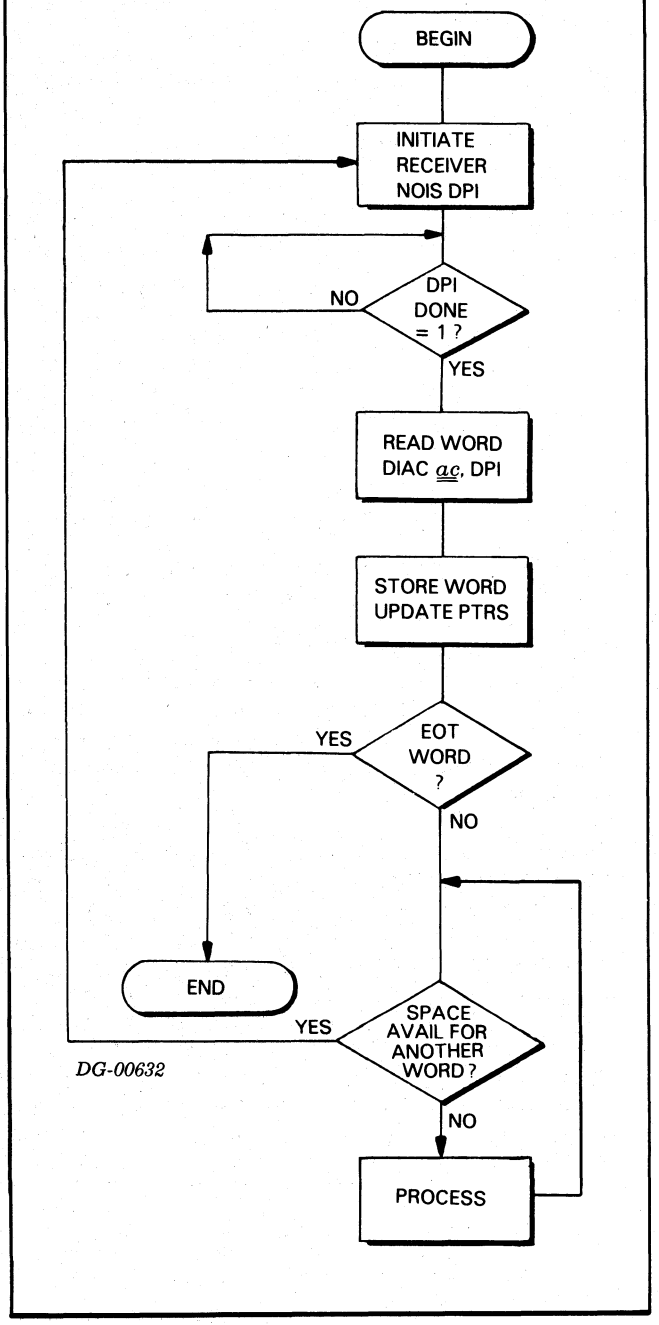


The data contained in bits 0-15 of this specified AC are placed in the DPI buffer of the other computer in the system. The previous contents of that DPI buffer are lost. The contents of the specified AC remain unchanged. After the data transfer, the Busy and Done flags are set according to the function specified by f.

TRANSMISSION FLOWCHART



RECEPTION FLOWCHART



HALF-DUPLEX COMMUNICATION

Half-duplex communications between processors is accomplished with two devices: one transmitter/receiver in each computer. Associated with each transmitter/receiver is a receiver buffer. The devices are asynchronous and operate under program control. In addition to the communication capability, the Busy and Done flags of these devices interact with each other in such a way that an *interlock* system can be implemented with suitable software.

In a system with two processors, it is advantageous, in terms of disc space, to have the operating systems of these processors share the system's disc pack. This is known as a *shared-disc environment.* It is conceivable that the systems would share not only system disc space, but also user disc space. In this way, only one copy of a file would be on disc and either processor could access it. This sharing presents a problem, however. If both processors were to read the same record of the same file at the same time, update the record, and then write it back, information would be lost. The updating performed by one processor would be overlaid by the updating performed by the other processor. This is clearly an undesirable situation.

Another version of the problem has to do with disc allocation. If both systems wish to allocate new disc space on the shared disc at the same time, it is conceivable that they would both allocate the same space. The result of this would be both systems writing on the same space of the disc, thinking that this space belonged to them.

In order to prevent these events from occurring, some sort of *interlock* facility is required. Using either the full-duplex or half-duplex communication capabilities of the IPB, some sort of interlock could be implemented, but it would be complex and require much code passing for the two processors to determine which one should have custody of which disc file. The interaction of the IPB Busy flags of the two processors gives a simple way to accomplish this interlock. One of the many ways to perform interlocking is described below. Let us call the computer wishing to establish a lock on a file *computer A* and the other computer in the system *computer B.*

PROGRAMMING SUMMARY

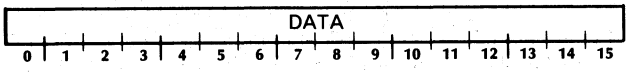
HALF-DUPLEX

Mnemonic	IPB
Device Code	36_8
Priority Mask Bit	6

ACCUMULATOR FORMATS

READ DATA (DIA)

WRITE DATA (DOA)



S, C, AND P FUNCTIONS

Input

f = S	In the computer executing the instruction, the IPB Busy flag is set to 1 if the IPB Busy flag of the other computer is 0. Even if both computers issue a Start at exactly the same time, only one IPB Busy flag will be set to 1.
f = C	In the computer executing the instruction, the IPB Done flag is set to 0. In the other computer, the IPB Busy flag is set to 0.
f = P	In the computer executing the instruction, the IPB Done flag is set to 0. In the other computer, the IPB Done flag is set to 1.

To start the lock procedure, computer A tries to set its IPB Busy flag to 1. Computer A does this by issuing a REQUEST BUS instruction (NIOS IPB). Computer A then checks to see if its IPB Busy flag is 1. If the flag is 1, computer A can continue with the procedure of establishing the desired lock. If the IPB Busy flag of computer A is 0, it means that the IPB Busy flag of computer B is 1, and computer B is about to do something in connection with either locking or unlocking a file. Computer A cannot continue the lock procedure until computer B completes its procedure.

Each computer keeps a table containing information about files in which it is interested. Each entry in the table has indicators which say whether or not this file has a lock on it and which computer has the lock. Once a computer succeeds in setting its IPB Busy flag to 1, computer A looks in its table to see whether or not computer B has a lock on the file that computer A wants. If the name of the desired file appears in this table, computer A must wait until computer B gives up its lock on the file. Note that computer A does not look in this *lock table* until it is successful in setting its IPB Busy flag to 1. If computer A looked in the table before attempting to set its IPB Busy flag to 1, it would be possible for computer B to establish a lock on the desired file between the time that computer A finished looking in the table and the time that computer A succeeded in setting its IPB Busy flag to 1.

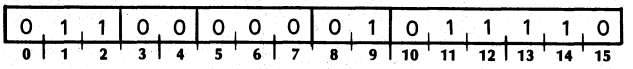
After computer A has determined that computer B does not have a lock on the desired file, computer A sends a code to computer B that means, "I want to establish a lock." Computer A does this by issuing a WRITE DATA instruction with a Pulse (DOAP AC, IPB). The code word must be in the specified AC. When computer A issues this instruction, the IPB Done flag of computer B will be set to 1, signaling computer B that there is a word in its transmitter/receiver buffer waiting to be read.

Computer B reads this word by issuing a READ DATA instruction with a Pulse (DIAP *ac*, IPB). This instruction sets the IPB Done flag of computer B to 0. The instruction also sets the IPB Done flag of computer A to 1, signaling computer A that computer B has read the word and is ready to receive another. This sequence continues until all information pertaining to the lock desired by computer A has been transferred to computer B. After the last word has been transferred, computer A issues a CLEAR FLAGS instruction (DIB) to set its IPB Busy and Done flags to 0. The system is now ready for another lock or unlock procedure to begin. The procedure for unlocking a file is the same as the procedure described above, except that the code word sent is the code word for unlock.

INSTRUCTIONS

Request Bus

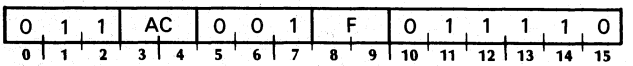
NIOS IPB



In the computer executing the instruction, the IPB Busy flag is set to 1 if the IPB Busy flag of the other computer is 0.

Read Data

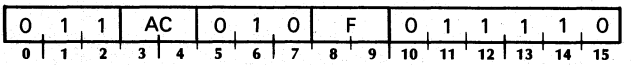
DIA *[f]* *ac*, IPB



The contents of the IPB buffer are placed in bits 0-15 of the specified AC in the computer executing the instruction. The previous contents of the specified AC are lost. The contents of the IPB buffer remain unchanged. After the data transfer, the Busy and Done flags are set according to the function specified by f.

Write Data

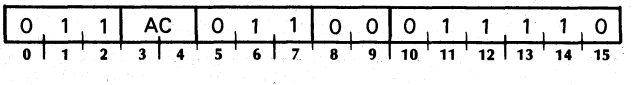
DOA *[f]* *ac*, IPB



Bits 0-15 of the specified AC in the computer executing the instruction are placed in the IPB buffer. The previous contents of the IPB buffer are lost. The contents of the specified AC remain unchanged. After the data transfer, the Busy and Done flags are set according to the function specified by f.

Clear Flags

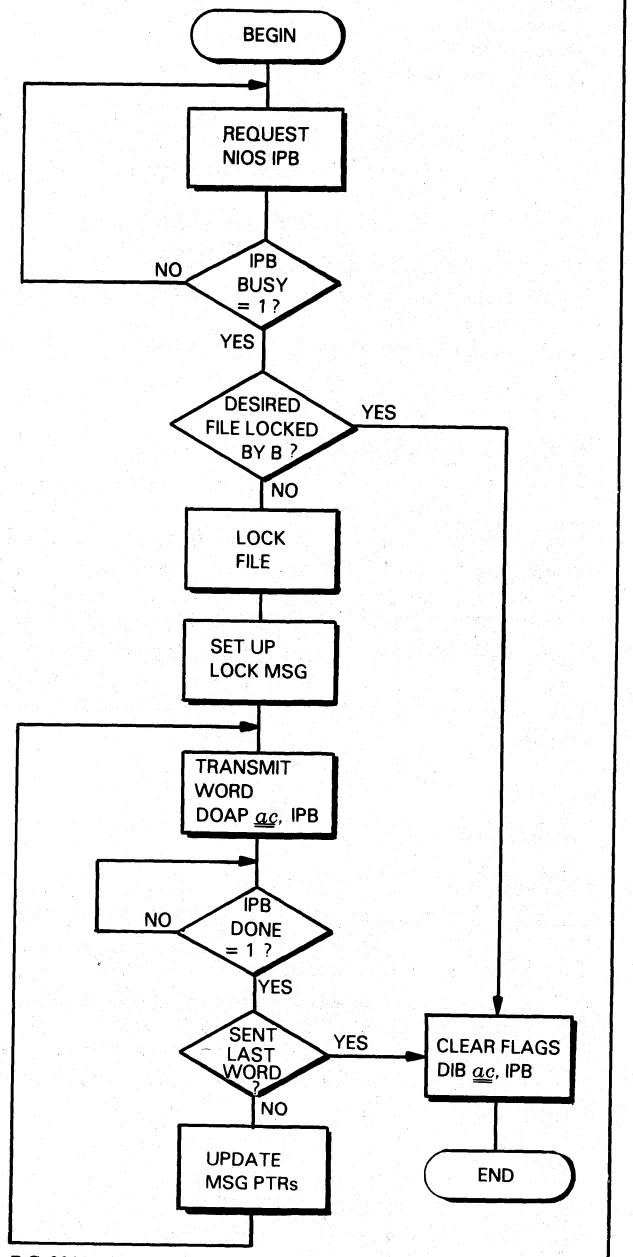
DIB *[f]* *ac*, IPB



The IPB Busy and Done flags in the computer executing the instruction are both set to 0. The contents of the specified AC are lost.

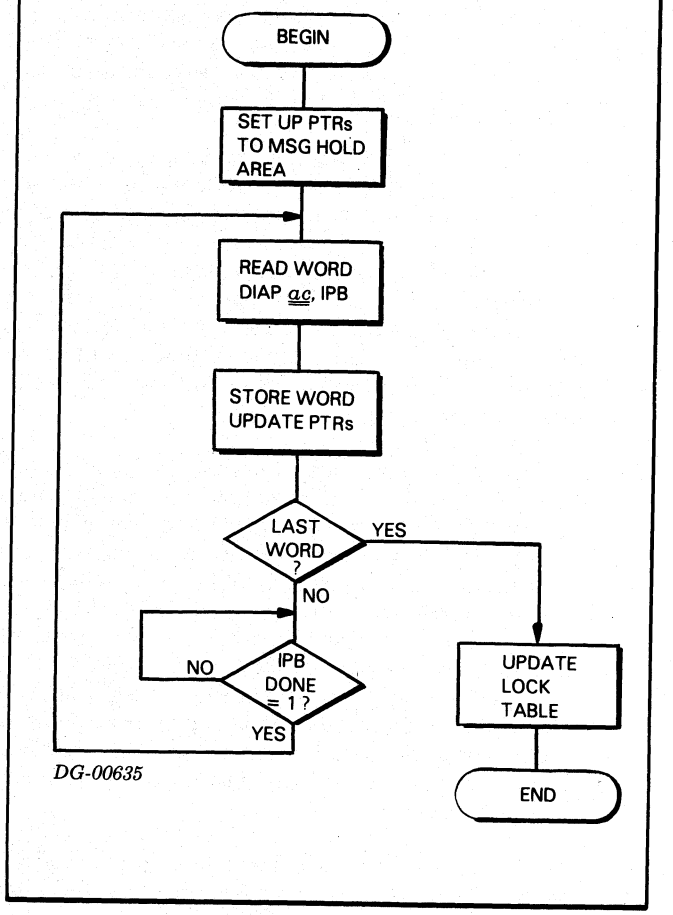
NOTE *None of the optional mnemonics specified by f should be coded with this instruction. If any of the three optional menmonics are coded, results will be unpredictable.*

TRANSMISSION FLOWCHART COMPUTER A



DG-00634

RECEPTION FLOWCHART COMPUTER B



DG-00635

WATCHDOG TIMER

The watchdog timer facility of the IPB consists of two timers. One timer is placed in each computer in the system. The timers are so constructed that if they are allowed to run for approximately one second without being restarted, they will initiate a program interrupt request in the computer in which they are installed.

The watchdog timer facility is useful in multi-processor configurations for indicating to one processor whether or not the other processor is still functioning. Each timer is normally restarted by the execution of the appropriate I/O instruction in the other computer at regular intervals of one second, or less. When one computer fails to restart the timer in the other computer within the specified time period, it indicates a probable hardware or software failure in the initiating processor.

To use the watchdog timer facility, each processor must issue a START TIMER instruction at least once per second. This instruction starts the timer in the other processor. If a timer is allowed to run for one second without being restarted, its Done flg will be set to 1. This indicates to the processor in which the timer is installed that the other processor has ceased to function. If a processor receives this indication, it should take appropriate action and then issue a CLEAR TIMER instruction. If the processor that is still functioning does not issue a CLEAR TIMER instruction, then it will continue to report the failure of the other processor.

In the event of one processor knowing that it is about to cease functioning, (i.e., a Power Fail situation) that processor can indicate this situation to the other processor by issuing a SET TIMER instruction. This instruction will immediately set the IVT Done flag in the other processor to 1.

PROGRAMMING SUMMARY

Mnemonic	IVT
Device Code	37_8
Priority Mask Bit	6
Timer Period (Second)	1

ACCUMULATOR FORMATS
(NONE)

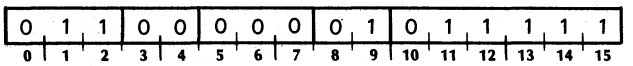
S, C, AND P FUNCTIONS

- f = S The timer in the other computer is started.
- f = C In the computer executing the instruction, the IVT Done flag is set to 0.
- f = P In the other computer, the IVT Done flag is set to 1.

INSTRUCTIONS

Start Timer

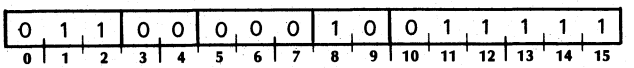
NIOS IVT



The one-second timer in the other processor is started.

Clear Timer

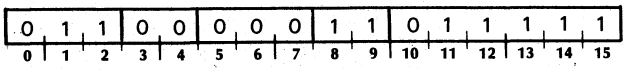
NOIC IVT



In the computer executing the instruction, the IVT Done flag is set to 0.

Set Timer

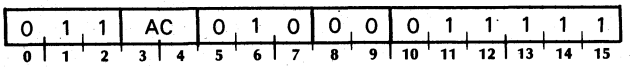
NIOP IVT



The IVT Done flag in the other processor is set to 1.

Start Own Timer

DOA *[f]* *ac*, IVT



In the computer executing the instruction, the timer is started. The contents of the specified AC remain unchanged.

CHAPTER III
INSTALLATION

INTRODUCTION

A dual DGC computer system requires the following items for an IPB communications link: two NOVA or ECLIPSE line computers, two 15-inch square 4240 Inter-Processor Bus printed circuit boards (one in each computer), two internal cables (one for each IPB board in each computer) and one external cable. These items can be purchased separately (an internal cable is supplied when a 4240 IPB printed circuit board is purchased), or as part of a complete minicomputer system. When items are purchased separately, the installation procedure will include all the tasks outlined in this chapter. In the case where the IPB is purchased as part of a complete system, the system will be shipped in a partially or completely assembled state, eliminating some of the tasks outlined in this section. The considerations involved in the installation of the IPB can be divided into inspection, unpacking, positioning of boards and cabling.

INSPECTION

Inspection is performed in three stages: before unpacking, after unpacking, and through the diagnostic and reliability programs. Before unpacking, all shipping containers should be inspected for signs of damage incurred in transit. After unpacking, items should be inspected for damage incurred in transit; if damage has occurred in either of these cases, both Data General Corporation and the carrier should be notified immediately. After installation, the IPB diagnostic and reliability programs can be run as a final inspection. The diagnostic program can be run as described in the diagnostic listing whenever a single IPB board is ready to be installed. The reliability program is run when the two IPB boards, internal cables, and external cables have all been installed.

UNPACKING

CAUTION *The primary consideration in unpacking is to avoid damage to the equipment being unpacked. Tapes sealing a container or folder should be cut with scissors or slit in such a manner as to avoid damage to the items inside. Avoid flexing or bending printed circuit boards. Save all packing materials.*

The unpacking procedure for a computer containing boards and internal cables is given in the technical manual for that computer. Printed circuit boards shipped separately are packed in cardboard folders within a carton. Cables are shipped in several different ways: in position when part of an assembled system, in the same carton as the printed cicuit boards when part of a small shipment, or in a package containing several cables. The unpacking procedure for a package containing printed circuit boards or cables is as follows:

1. Open the cardboard carton, avoiding damage to its contents, and remove the loose *flow-pack* and any cables shipped in the package.

2. Remove the folders containing printed circuit boards. Avoid bending the boards.

3. Open each folder using scissors to cut the tape. Avoid stabbing or bending the folder.

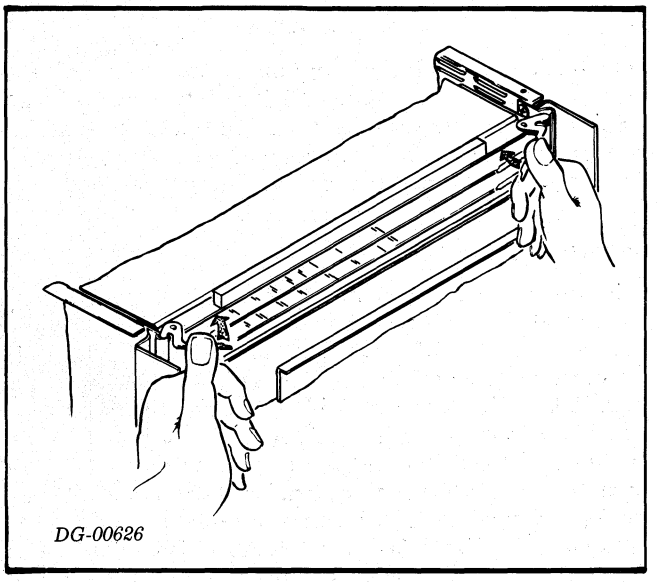
4. Remove the printed circuit board from the folder and open the plastic bag containing the board.

POSITIONING THE IPB BOARD

The slots available in any NOVA or ECLIPSE line computer for the IPB will depend on the computer, its options, and the other I/O devices in use. The installation section of the computer technical manual (and some of the I/O controller manuals) should be consulted for specific rules and some recommendations on board placement. Some general rules concerning the occupation of slots in DGC computers are: slot 1 always has a CPU board; slot 2 has either a CPU board, a memory, or a multiply/divide option, depending on the computer and its options; slot 3 is wired for the basic I/O devices (Teletype or CRT display, paper tape punch and paper tape reader) using the 4007 I/O interface subassembly, but can also have a memory board. Also, the Data General convention is to install memory boards from lower numbered slots up and I/O controllers from the highest slot down.

In spite of the rules given above, the task of choosing a slot in the computer chassis in which to place the IPB board is not necessarily trivial. The major user criterion for choosing a slot in which to place the IPB board is the priority for interrupt service relative to other controllers he wishes to assign the IPB. The program interrupt chain is hardwired so that device controllers in lower slots have a higher priority in responding to INTERRUPT ACKNOWLEDGE instructions. Within the IPB board itself there is an internal interrupt priority chain that goes from highest priority to lowest as follows: half-duplex line, full-duplex receiver, full-duplex transmitter, watchdog timer.

A related consideration in choosing the slot of the IPB board is the continuity of the program interrupt and data channel priority chains. These chains are independent of each other and operate by passing an enabling signal from slot to slot through each board in the chain. A controller can be recognized for data channel service or interrupt service only when the enabling signal, $\overline{\text{DCHP}}$ or $\overline{\text{INTP}}$, respectively, is low. The lowest controller requesting service is recognized and removes the enabling signal level from other controllers further along the chain. All DGC boards maintain the continuity of both chains by either participating in the chain or having internal jumpers passing the enabling signals through the board. If a slot is empty (or contains a board not manufactured by Data General Corporation which does not maintain data channel and interrupt continuity) controllers located in higher slots will not receive the program interrupt and data channel enabling signals, unless the continuity is maintained by jumpering the enabling signals over the empty slot(s) to the next DGC board.



DG-00626

The continuity of the program interrupt enabling signal ($\overline{\text{INTP}}$) is maintained by placing a jumper from pin A96 of the lowest slot in a sequence of empty slots (or slots containing boards which do not pass the signal) to pin A96 of the next slot containing a DGC board. This jumpering is required from each set of slots which do not pass $\overline{\text{INTP}}$. Similarly, for the data channel enabling signal ($\overline{\text{DCHP}}$), a jumper is placed from pin A94 of the lowest slot in a sequence of slots that do not pass the $\overline{\text{DCHP}}$ signal to pin A94 of the next slot containing a DGC board.

When the IPB board is placed in a previously empty slot that has been correctly jumpered over to maintain the data channel and program interrupt priority chain continuity, the jumpers must be removed or rewired to reflect the new arrangement.

Once the slot has been chosen for the IPB board, the remaining tasks are cabling the system and inserting the IPB board in the slot. The IPB board is placed in a slot by carefully sliding it, component side up, into the guides on each side of the chassis. The locking tabs at the corners of the board are used to provide leverage to completely seat the board in the two connectors in the slot. The board is plugged into the connectors by pressing the locking tabs towards the board and is unplugged by releasing the locking tabs as shown above.

CABLING

The cables in the IPB communications link pass data and signals between the two IPB boards. Two internal cables are needed (one for each computer) to link the back panel pins of the slot occupied by an IPB board to an external cable connecting the two computers.

Internal Cables

The types of cables required by different systems are determined by the internal cable connector mounting schemes used in different DGC computer models. DGC computers having 7 or 17 slots (i.e., NOVA computer, SUPERNOVA computer, NOVA 800, 840, 1200, and 1230 computers) are designated group A; these computers have a cable mounting area on the chassis which is suitable for internal cables using female socket connectors. NOVA line computers having 4 or 10 slots (i.e, NOVA 820, 1210, 1220, and NOVA 2) and ECLIPSE line computers are designated group B; these computers mount paddleboard connectors for internal cables on standoffs next to other paddleboard connectors on the chassis.

When an IPB board is purchased, the DGC model computer it is to be used in must be specified so that the correct internal cable can be shipped with the board. The internal cable is installed by mounting the connector in the cable mounting area of the chassis and wire wrapping the cable wires to the appropriate back panel pins of the slot holding the IPB board. The one exception to this procdedure is on slot 9 of all 10-slot NOVA line computers (NOVA 820, 1200 and NOVA 2/10) where an internal cable is etch wired onto the back panel in the factory. The correct back panel pin for each wire is indicated by a tag on the wire and also in the wire list (see below). Pins with an A label are on the left side of the back panel; pins with a B label are on the right side; odd pins are on the upper row; even pins are on the lower row; pin numbering is from left to right.

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

External Cable

When an IPB cable is purchased, the model numbers of the DGC computers it is to connect must be specified so that the correct external cable will be sent. There are three types of external cable: cables with two male socket connectors for connecting two group A computers, cables with two female edge connectors for connecting two group B computers, and cables with one male socket connector and one female edge connector for connecting a group A computer to a group B computer. The cabling of the IPB communications link is completed by the plugging of the external cable connectors into the connectors of the internal cables going into the slots containing the IPB boards in the two computers.

CAUTION *The external cable and all device cables should be secured and strain relieved at each computer. Failure to do so may result in damage to the back panel pins.*

DIAGNOSTIC AND RELIABILITY PROGRAMS

The IPB documentation package includes a diagnostic program tape, a reliability program tape and listings giving operating procedures for these tapes. The diagnostic program is used to test a single IPB board; it requires a special test plug supplied with the board. The reliability program tests the various types of transfers between two computers carried out through the IPB.

REFERENCES

Wire Lists

008-000194 Wire list for internal cable (general purpose) for NOVA 1210, 1220 and 820 computers

008-000426 Wire list for internal cable (general purpose) for NOVA 2 and ECLIPSE line computers

008-000467 Wire list for internal cable for NOVA 1200, 1230, 800, 840 computers

008-000468 Wire list for external IPB cable (group A to group A)

008-000469 Wire list for external IPB cable (group A to group B)

008-000470 Wire list for external IPB cable (group B to group B)

Diagnostics

095-000177 Absolute binary IPB diagnostic tape

095-000178 Absolute binary IPB reliability tape

096-000156 Listing for the IPB diagnostic tape

096-000157 Listing for the IPB reliability tape

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

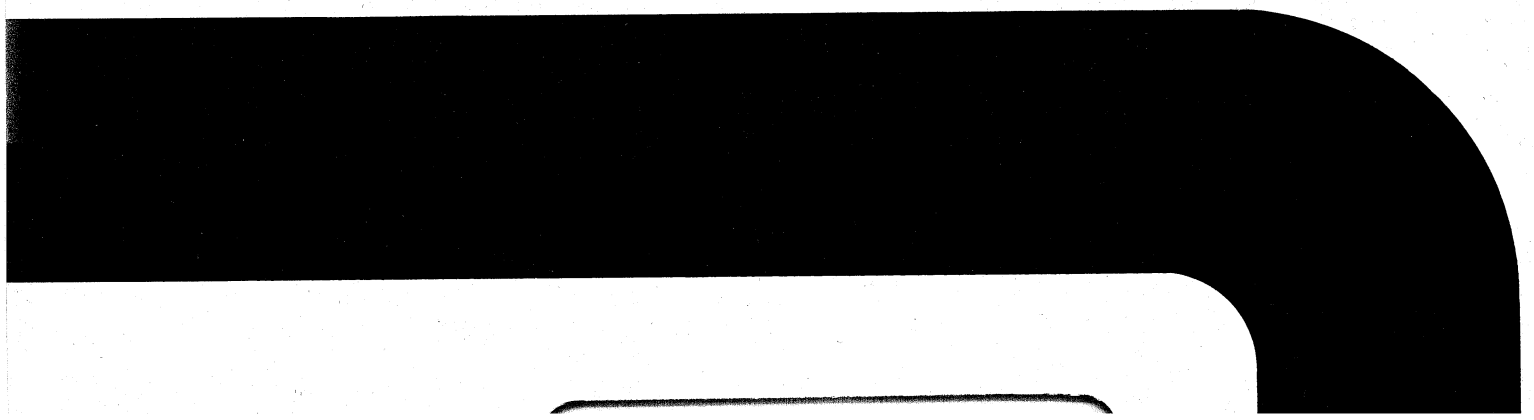
ATTENTION: Engineering Publications

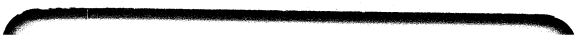
FOLD UP SECOND FOLD UP

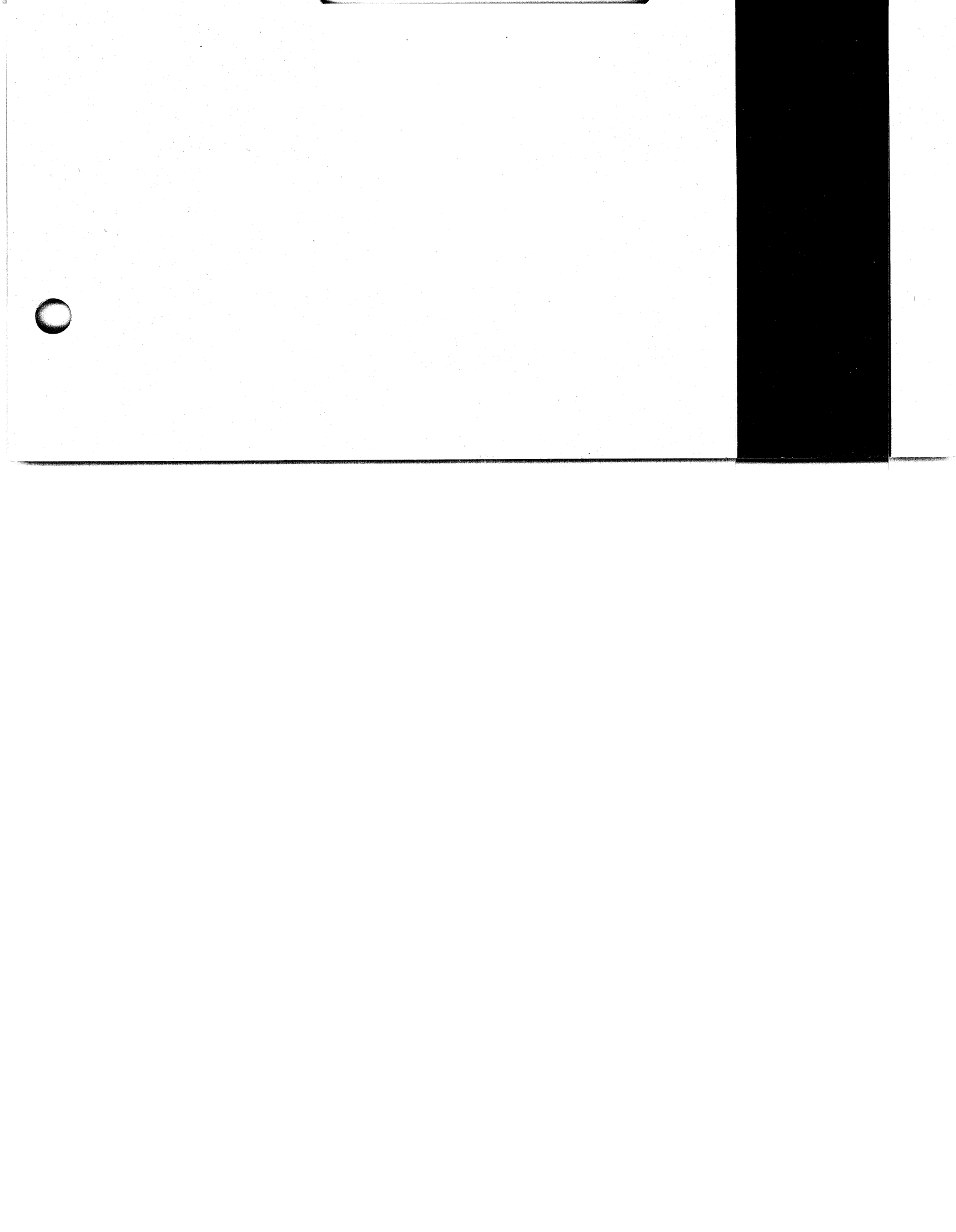
STAPLE

[redacted]

DataGeneral







Technical Manual

4240 INTERPROCESSOR BUS SUBSYSTEM

015-000033-00

NOTICE

Data General Corporation (DGC) has prepared this maual for use by DGC personnel and customers as a guide to the proper installation, operation, and maintenance of DGC 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 informatin 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. 015-000033

Printed in the United States of America
Revision 00, September 1976

Model 4240

INTERPROCESSOR BUS SUBSYSTEM

DataGeneral

TABLE OF CONTENTS

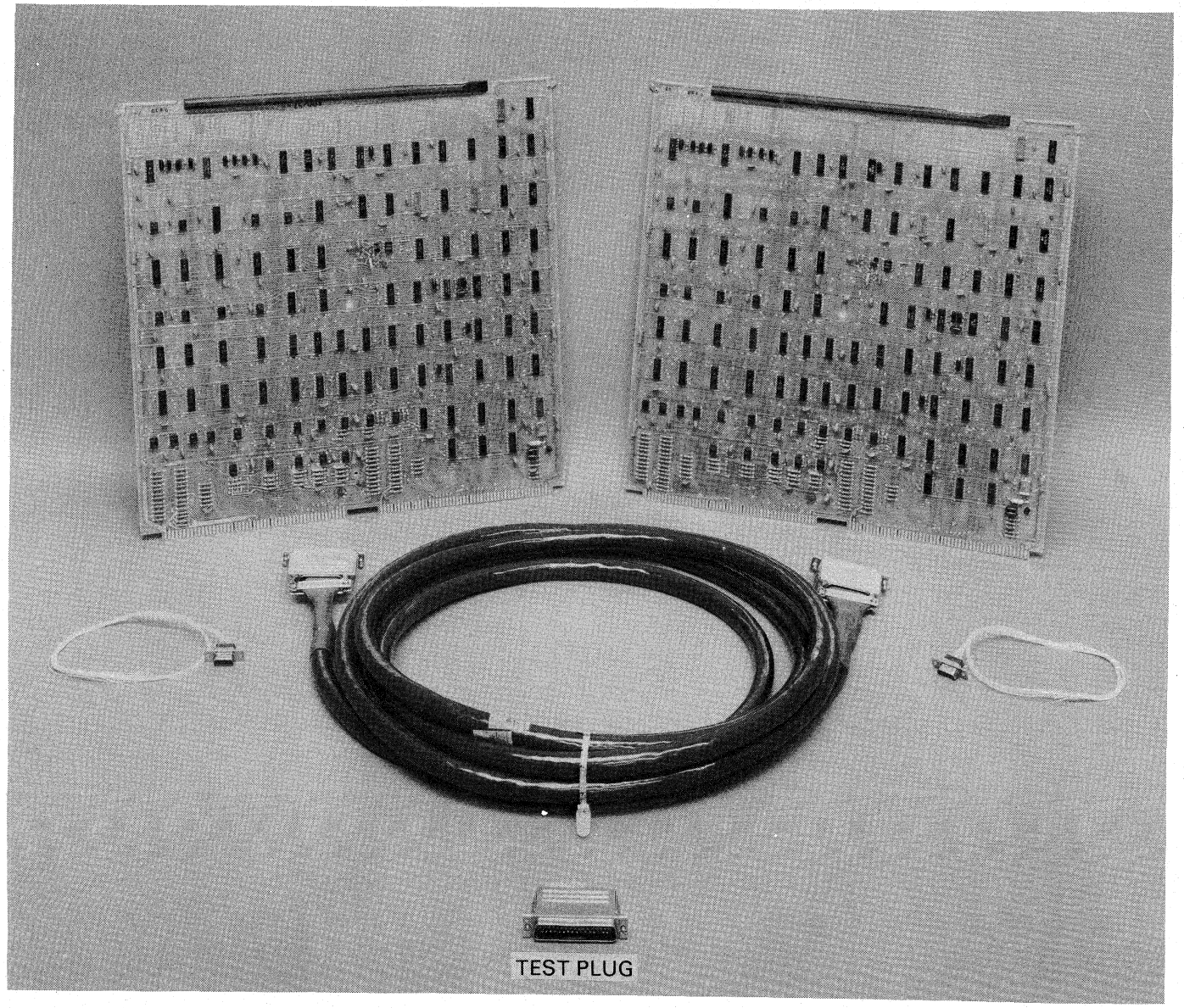
SECTION I
OVERVIEW

TABLE OF CONTENTS (Continued)

SECTION II
THEORY OF LOGICAL OPERATION

APPENDICES

This page intentionally left blank.



4240 SUBSYSTEM

SECTION I
OVERVIEW

INTRODUCTION

The Data General Corporation 4240 Inter-Processor Bus (IPB) subsystem provides the facilities for asynchronous communications between any two NOVA or ECLIPSE line computers. It consists of two, identical, 15" square, IPB printed circuit boards (one board installed in an I/O slot of each computer); one connecting 15' IPB external cable; and two internal cables (one cable installed in each computer) connecting the printed circuit boards to the external cable.

The IPB subsystem is under the direct program control of both computers and provides the computers with the following distinct capabilities:

- Bi-directional, full-duplex communications between computers.
- Bi-directional, half-duplex communications between computers, which may be "interlocked" with suitable software. The "interlock", consisting of interconnected status flags, allows only one processor to transmit data when the appropriate software protocol is honored by both computers.
- Program interrupt request in either computer when the other computer: 1) fails to perform as expected; or, 2) generates a signal indicating a probable "power fail" condition. This capability is implemented by missing pulse detectors and is called the Watchdog Timer facility.

The IPB transfers two types of information between processors: status information and data. Status information is transferred by the setting of Busy and Done flags. These flags are set both by programmed I/O instructions and by flag interactions between boards. Each processor is notified of status information either by means of the program interrupt facility or by flag checking software routines. Data is transferred between an accumulator in one processor and an accumulator in the other processor under direct program control, one 16-bit word at a time.

Using software routines solely dedicated to the operation of the IPB, two NOVA or ECLIPSE line computers can achieve data transfer rates exceeding 100,000 16-bit words per second. Using the program interrupt facility, the data transfer rate is limited by the time required by the interrupt handling routine in each computer.

The IPB is useful in applications where direct program control over the transfer of information is needed and where the information must be transferred rapidly. Typical applications of the IPB include: parallel processing arrangements, communications concentration systems, shared disc environments, and back-up systems. In a parallel processing arrangement or in a communications concentration system, the full-duplex communications link allows two computers to exchange data rapidly. In a shared disc environment, the "interlock" capability of the half-duplex communications link provides a means of communicating disc space reservations and actions performed on disc files. In a back-up system, the "Watchdog Timer" facility can be used to notify one computer of a possible software or hardware failure in the other computer.

Architecture

The communications capabilities of the IPB subsystem are implemented by the two IPB printed circuit boards and the IPB external cable. Although the two boards are identical, when they are connected to the IPB external cable they assume different roles. These roles are assigned by asymmetrical wiring of the "Left" and "Right" connectors of the cable. In this manual, each IPB board is referred to by its connection to the external cable; i.e., "Left" IPB board, "Right" IPB board. In data communications, the "Right" IPB board serves as the data holding area; the "Left" IPB board serves in a gating capacity. This distinction between "Left" and "Right" IPB boards does not affect the programming, operation, or communications abilities of either computer in the system.

The block diagram shows the data paths, data holding areas, missing pulse detectors, and status flags used in IPB communications.

Data Paths and Holding Areas

Data is transferred between a specified accumulator in the computer and that computer's IPB board via the I/O data bus. The data path between each processor's I/O data bus and the data holding areas of the IPB differs. On the "Right" IPB board, data is transferred directly between the "Right" processor's I/O data bus and the data holding areas. On the "Left" IPB board, data is transferred between the "Left" processor's I/O data bus and the data holding areas on the "Right" IPB board via a 16-bit, bi-directional, half-duplex, data bus in the external cable. A general purpose transmitter and a general-purpose receiver on the "Left" IPB board, which function as repeaters, form the communications link between the external cable data bus and the I/O data bus of the "Left" processor.

The data holding areas on the "Right" IPB board consist of four 16-bit storage buffers. Associated with each storage buffer is a set of 16 gates which function as a receiver. The storage buffer of this storage buffer/receiver combination is under the direct program control of one of the processors while its associated receiver is under the direct program control of the other processor. Each processor controls one full-duplex and one half-duplex storage buffer and one full-duplex and one half-duplex receiver. Each storage buffer and each receiver responds to a device code.

In full-duplex communications, the processor's full-duplex storage buffer is referred to as a 16-bit full-duplex transmitter and responds to device code 41; the processor's full-duplex receiver is referred to as a 16-bit full-duplex receiver and responds to device code 40.

In half-duplex communications, the processor's half-duplex storage buffer and half-duplex receiver are referred to as a 16-bit half-duplex transmitter/receiver and respond to the same device code, 36.

Data is loaded into a storage buffer by the execution of the appropriate I/O instruction in the transmitting processor. This data is latched at the buffer's output and is available to the receiving processor. When the receiving processor executes the appropriate I/O instruction, the associated set of receiver gates is enabled, allowing the contents of that storage buffer to pass to the receiving processor's I/O data bus.

This separate control of the storage buffer and its associated receiver allows asynchronous data communications between processors. The arrangement of the storage buffer/receiver combinations allows data to be transferred from the left processor to the right processor and from the right processor to the left processor in both full- and half-duplex communications. In full-duplex communications, the true, half-duplex character of the external cable data bus is transparent to the programmer because of the high-speed data transfer rate capability of the IPB.

Missing Pulse Detectors

The missing pulse detectors, one on each IPB board, respond to device code 37. Each detector consists of a one-second timer which is normally restarted by the execution of the appropriate I/O instruction in the other computer at regular intervals of one second or less. When one computer fails to restart the timer in the other computer within the specified time period, that timer's Done flag is set, generating a program interrupt request in the computer in which the timer is installed. Additionally, when either computer knows that it is about to cease functioning (e.g., a power fail condition), that computer may set the other computer's timer Done flag directly by the execution of the appropriate I/O instruction.

Status Flags

Each IPB communications element, or device, has its own set of status flags. The status flags for the devices under the control of the "Left" processor are located on the "Left" IPB board; the status flags for the devices under the control of the "Right" processor are located on the "Right" IPB board. In addition, the "Right" IPB board contains the Busy flag circuitry for both half-duplex transmitter/receiver devices; this circuitry provides the software "interlock" in half-duplex communications.

When the appropriate software is used, these flags provide the processors with useful information; such as, a word is in a storage buffer, the half-duplex communications link is or is not available for transmitting data, the other processor is experiencing a probable software or hardware malfunction.

THE IPB TECHNICAL MANUAL

The purpose of this manual is to describe the logical operation of the IPB subsystem. The information is presented in two sections:

- Section I introduces the 4240 Inter-Processor Bus subsystem and the conventions used to describe it.
- Section II describes the logical operation of the IPB.

Print References

To assist the reader in understanding the IPB subsystem, frequent references are made to the IPB logic schematic, DGC No. 001-000614. Portions of this schematic are referred to by grid reference; e.g., 2A5. The grid reference 2A5 is interpreted as follows: the first number of the reference, 2, is the page number of the schematic; the letter A refers to the row on that page; and the last number, 5, refers to the column on that page.

Nomenclature

The title blocks on each page of the IPB logic schematic refer to the communications link described on that page as follows: IPB Interlocked Comm. refers to the half-duplex communications link; IPB Non-Interlocked Chan. refers to the full-duplex communications link; and Interval Timer refers to the Watchdog Timer.

Logic Conventions

Drawings

Data General logic prints are drawn in close accordance with Mil-Std-806C. With this convention, logical functions are drawn as physically implemented. That is, where discrete gates are used to implement a function, these gates are shown. On the other hand, where a more complex integrated circuit is used, for instance a multiplexor, that function is shown as a rectangular box instead of the gates comprising the function.

Signal Levels

Throughout this manual, a distinction is frequently made between electrical levels and logical values. To minimize confusion, electrical levels are always indicated by an "H" or "L", and logical values by a "1" or "0". As an electrical level, an "H" indicates that the signal is high (greater than +2.0 volts) and an "L" indicates that it is low (less than +0.7 volts). An asserted, or true, signal is indicated by a logical "1" and a false signal by a "0".

Signal Names

The voltage level at which a signal is said to be "asserted" (true) is a matter of definition. To distinguish between signals that are asserted high (0=L, 1=H) and those that are asserted low (0=H, 1=L), a naming convention has been adopted in Data General's documentation which defines the relationship between the logical value and electrical level of a signal. If the signal name includes a horizontal bar over the name, as **WRITE**, then that signal is asserted when it is at a low electrical level; conversely, a signal without the bar, **WRITE** is asserted when high. Logical functions may often require more than one binary signal. For instance, three lines are required to express an octal digit. Generally, these closely related signals are individually identified by effectively subscripting a common label. For instance, suppose that **BUS0** through **BUS5** are all required to completely specify a function. All or part of such a group of signals is identified by placing brackets around the range of subscripts included, as **BUS[0-5]** In this case, the suffix carries the information that there are six **BUS** lines under discussion, from **BUS0** through **BUS5**, inclusive.

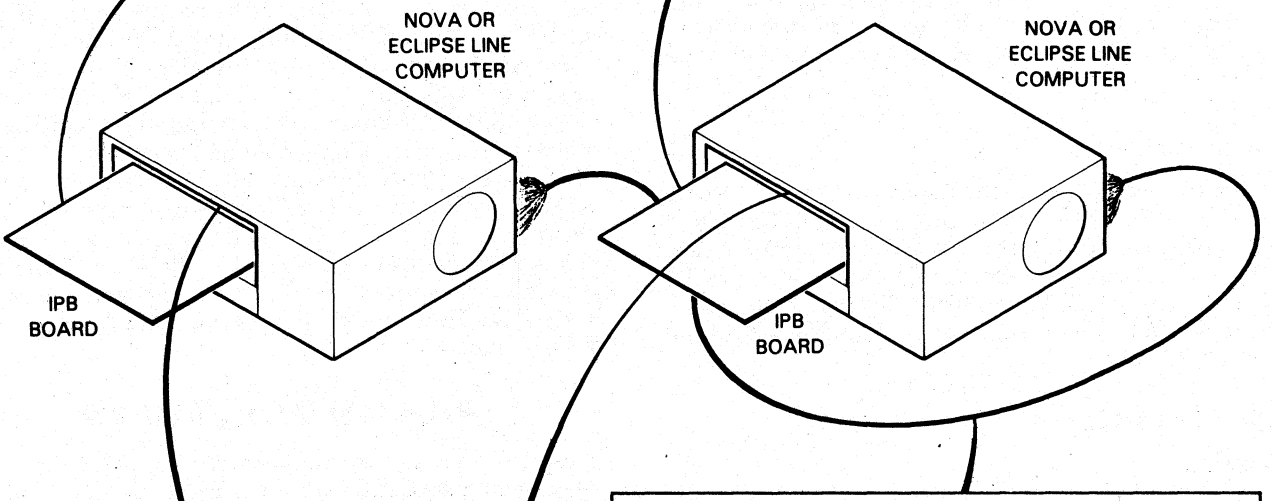
RELATED DOCUMENTS

The IPB Related Documentation diagram lists the engineering drawings and the manuals pertinent to the IPB. The ordering number of each document is listed in the diagram. The Components Guide gives logic diagrams, pin designations and truth tables for the I.C.'s used in Data General equipment. The Interface Designer's Reference, NOVA and ECLIPSE Line Computers, provides useful information concerning the processor's I/O bus and the program interrupt system. The Installation Data Sheet explains how to install the 4240 Inter-Processor Bus subsystem. The Technical Reference, 4240 Inter-Processor Bus, explains how to program the subsystem.

The documentation diagram also lists the diagnostic and reliability program tapes and listings together with the test plug wire lists.

IPB RELATED DOCUMENTATION

IPB PRINTED CIRCUIT BOARD	
IPB LOGIC SCHEMATICS	001-000614-03
IPB ILLUSTRATED PARTS LIST	016-000152-00
COMPONENTS GUIDE	015-000028
USER'S MANUAL, INTERFACE DESIGNER'S REFERENCE, NOVA AND ECLIPSE LINE COMPUTERS	015-000031
INSTALLATION DATA SHEET	010-000091
TECHNICAL REFERENCE, 4240 INTER-PROCESSOR BUS	014-000056-01



IPB EXTERNAL CABLE, MODEL	
IPB EXTERNAL CABLE, A TO A	008-000468-02
IPB EXTERNAL CABLE, A TO B	008-000469-00
IPB EXTERNAL CABLE, B TO B	008-000470-00

IPB INTERNAL CABLES	
NOVA 820, 1210, 1220	008-000194-03
NOVA 2 AND ECLIPSE LINE	008-000426-01
NOVA 800, 830, 840, 1200	008-000467-00

TEST PLUG (required to run diagnostics)	
NOVA 800, 840, 1200	008-000465-00
NOVA 820, NOVA 2, 830, 1210 1220 AND ECLIPSE LINE	008-000466-00

DIAGNOSTIC TAPES AND LISTINGS	
AB IPB DIAGNOSTIC	095-000177-01
AB IPB RELIABILITY	095-000178-02
LS IPB DIAGNOSTIC	096-000156-01
LS IPB RELIABILITY	096-000157-02

DG-02568

SECTION II
THEORY OF LOGICAL OPERATION

INTRODUCTION

The IPB subsystem consists of two sets of four communications elements (one set for each computer), which are referred to as devices. These devices provide the computers with three distinct communications links:

- The 16-bit full-duplex transmitter and the 16-bit full-duplex receiver of each processor provide the full-duplex communications link between processors.
- The 16-bit half-duplex transmitter/receiver of each processor provides the half-duplex communications link between processors.
- The missing pulse detector of each processor provides a functional status communications link between processors. This is called the Watchdog Timer facility.

Since each of these communications links functions independently when used with the appropriate software, this section of the manual is presented in four subsections: the first is concerned with that information which is common to all elements of the IPB subsystem; the second is concerned with full-duplex communications between processors; the third is concerned with half-duplex communications between processors; and the fourth is concerned with the Watchdog Timer facility.

THE SUBSYSTEM

The communications capabilities of the IPB are implemented by the two IPB printed circuit boards and the IPB external cable. The internal cables, one installed in each computer, serve only as an interface between each computer's IPB printed circuit board and the external cable.

"Left" and "Right" IPB Boards

When the identical IPB printed circuit boards are connected, through the internal cables, to the external cable, the "Left" connector of the external cable partially disables the IPB board connected to it. This is accomplished by the asymmetrical wiring of the $\overline{\text{LEFT FINDER}}$ signal at the "Left" and "Right" connectors of the cable. On the "Left" IPB board, $\overline{\text{LEFT FINDER}}$ is always low; on the "Right" IPB board, $\overline{\text{LEFT FINDER}}$ is always high.

As shown in the block diagram, on the "Left" IPB board the $\overline{\text{LEFT FINDER}}$ signal disables the full-duplex storage buffer/receiver combinations. Additionally, this signal partially disables the half-duplex storage buffer/receiver combinations, allowing them to function as repeaters. In both full- and half-duplex communications, one half-duplex storage buffer/receiver combination serves as a general purpose transmitter while the other half-duplex combination serves as a general purpose receiver. The $\overline{\text{LEFT FINDER}}$ signal also disables the half-duplex Busy flag circuitry on the "Left" IPB board, which provides the software "interlock" on the "Right" IPB board.

On the "Right" IPB board, the $\overline{\text{LEFT FINDER}}$ signal enables the full-duplex storage buffer/receiver combinations and those portions of the half-duplex storage buffer/receiver combinations which are disabled on the "Left" IPB board. This arrangement allows the "Right" IPB board to serve as the data holding area for the subsystem.

The missing pulse detectors and the remaining status flags are not affected by the $\overline{\text{LEFT FINDER}}$ signal.

IPB Signals

Each processor's IPB devices are controlled by signals sent to the processor's IPB board via the I/O bus. These signals are generated by the execution of the appropriate I/O instructions in the initiating processor. Three types of signals are generated by the fields of an I/O instruction which affect the IPB devices and their respective status flags:

- The device code (36, 37, 40, 41)
- The instruction (DOA, DIA, DIB)
- The flag command (Start, Clear, I/O Pulse)

When these signals are asserted and received by the initiating processor's IPB board, they generate two types of signals: first, those which affect the circuitry of the initiating processor's IPB board; and, second, those which affect the circuitry of the other processor's IPB board. Refer to the diagram showing the paths of IPB signals.

The signals which affect the circuitry of the initiating processor's IPB board have no suffix added to them; e.g., **DS40.**

The signals which affect the circuitry of the other processor's IPB board have the suffix "O" added after the last letter in the signal name; e.g., **$\overline{\text{DSO40}}$**. These signals are sent to the other processor's IPB board via the IPB external cable. When these signals are asserted on the other processor's IPB board, the suffix changes from "O" (out) to "I" (in); thus, the **$\overline{\text{DSO40}}$** signal on the initiating processor's IPB board becomes the **$\overline{\text{DSI40}}$** signal on the other processor's IPB board.

On both IPB boards, the device code is ANDed with the instruction and/or flag command. For example, the **DS40** and **DIA** signals generate the **$\overline{\text{DIA40}}$** signal on the initiating processor's IPB board; the **$\overline{\text{DSI40}}$** and **DIAI** signals generate the **$\overline{\text{DIAI40}}$** signal on the other processor's IPB board.

In addition to signals which are generated directly by the execution of an I/O instruction, three signals are generated when the appropriate status flags are set. As shown in the diagram, these signals also contain the "O" and "I" suffix.

Status Information

Status information concerning IPB operations is provided by the state (0 or 1) of each IPB device's Busy and Done flags. This information can be determined by the processor during flag checking software routines or by means of the program interrupt facility.

Flag Checking

Flag checking allows the program to ascertain the status of the Busy and Done flags of the IPB devices under the control of the processor at any time. This is accomplished by the execution of the appropriate I/O Skip instructions, which test the state of two I/O bus lines, **$\overline{\text{SELB}}$** and **$\overline{\text{SELD}}$**. Whenever the CPU issues an IPB device code, the **$\overline{\text{SELB}}$** line will be asserted if that device's Busy flag is set to 1 and the **$\overline{\text{SELD}}$** line will be asserted if that device's Done flag is set to 1.

Program Interrupt

The program interrupt facility provides the means for notifying the processor when one of its peripheral devices requires service. An IPB device requests a program interrupt by asserting the **$\overline{\text{INTR}}$** line of the I/O bus. When more than one IPB device is requesting a program interrupt, the devices are serviced according to the following priority scheme:

PRIORITY	IPB DEVICE
1	HALF-DUPLEX TRANSMITTER/RECEIVER
2	FULL-DUPLEX RECEIVER
3	FULL-DUPLEX TRANSMITTER
4	TIMER (MISSING PULSE DETECTOR)

The program interrupt system is explained in detail in the User's Manual, Interface Designer's Reference, NOVA AND ECLIPSE LINE COMPUTERS, DGC No. 015-000031.

Timing

Each IPB operation is governed by the timing of the signals received from the CPU initiating the operation. Since timing varies between computer models, it is necessary to refer to the technical manual for the particular NOVA or ECLIPSE line computers in use to determine exact timing information.

Typical timing of signals received from the CPU and typical timing of resulting IPB signals are shown in the timing diagrams. In this manual, five relative IPB time periods are defined, IPB0 through IPB4. These time periods are used only for illustrative purposes and do not relate to any timing signals.

Time Periods

IPB0 is the time period starting when the device code only is asserted (e.g., **DS40**) and ending after an instruction signal, if any, is asserted (e.g., **DIA**).

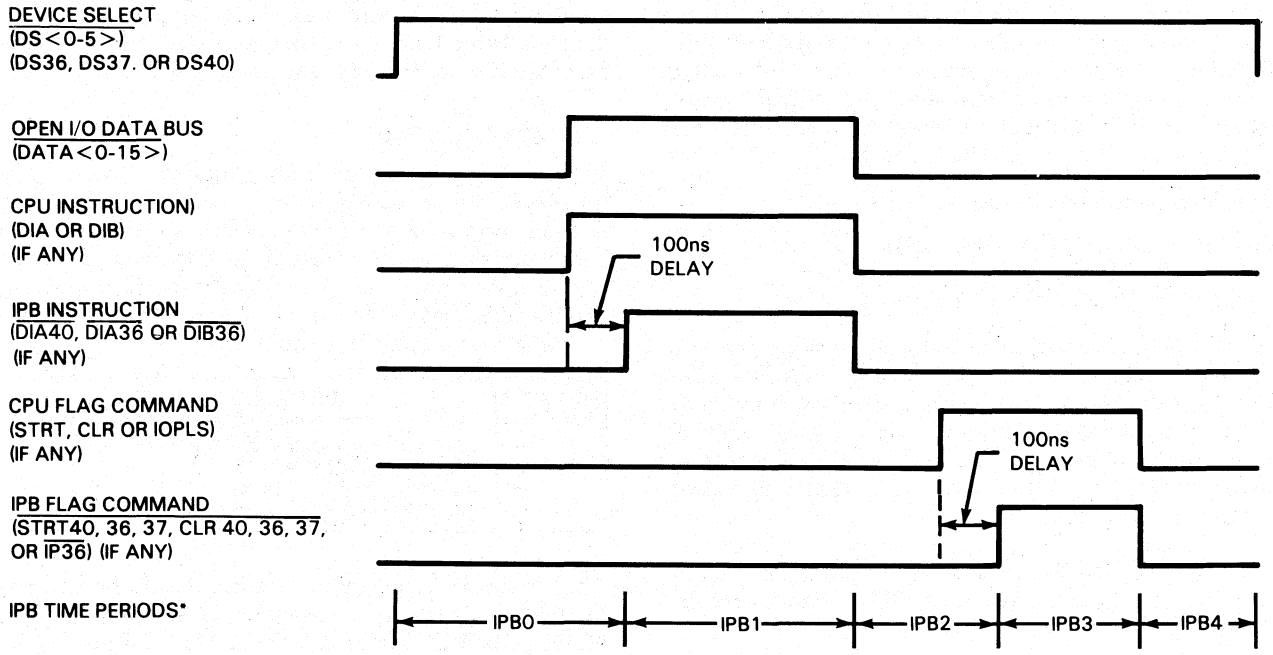
IPB1 is the time period starting when an IPB instruction signal, if any, is asserted (e.g., **$\overline{\text{DIA40}}$**) and ending when the IPB instruction ends.

IPB2 is the time period starting after the end of the IPB instruction signal, if any, and ending after a flag command signal, if any, is asserted (e.g., **CLR**).

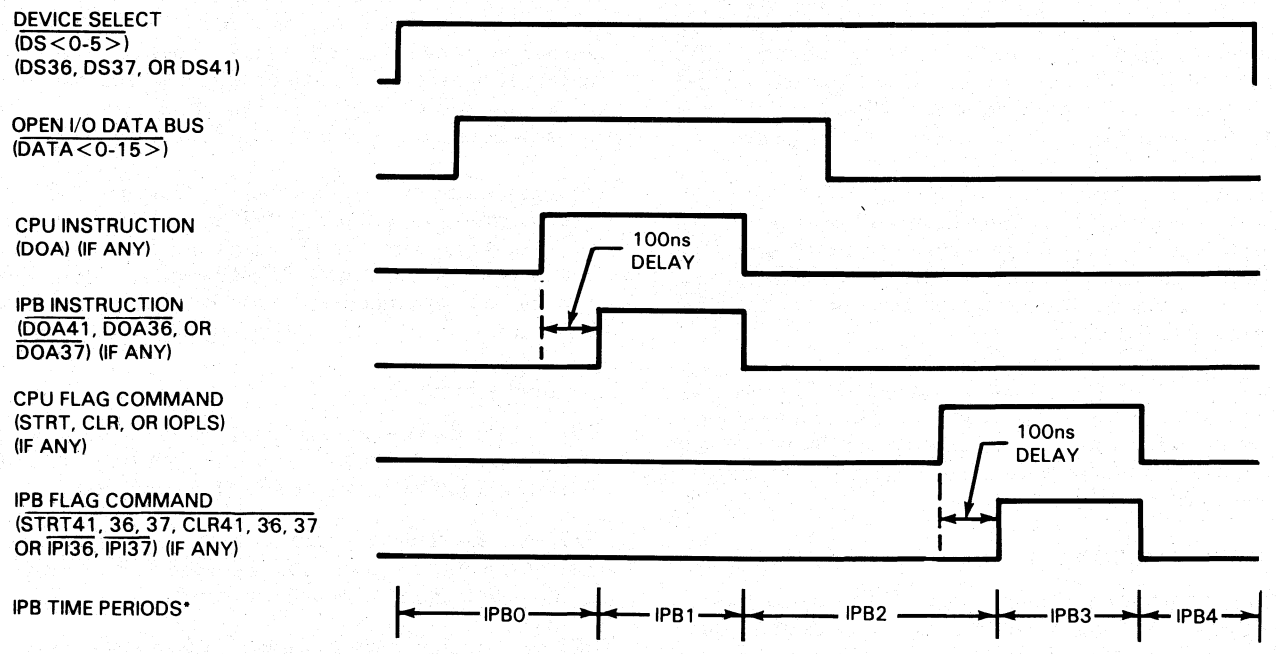
IPB3 is the time period starting when an IPB flag command signal, if any, is asserted (e.g., **$\overline{\text{CLR40}}$**) and ending when the IPB flag command signal ends.

IPB4 is the time period starting after the end of the IPB flag command signal, if any, and ending when the instruction ends.

TYPICAL PROGRAMMED IPB INPUT AND FLAG COMMAND TIMING



TYPICAL PROGRAMMED IPB OUTPUT AND FLAG COMMAND TIMING



*THESE TIME PERIODS ARE USED ONLY FOR ILLUSTRATIVE PURPOSES AND DO NOT RELATE TO ANY TIMING SIGNALS.

DG-02571

FULL-DUPLEX COMMUNICATIONS

The full-duplex communications link provides the facilities for high-speed, bi-directional communications between processors. Each processor controls one full-duplex transmitter (storage buffer) and one full-duplex receiver (set of receiver gates), which allows data to be transferred from the left processor to the right processor and from the right processor to the left processor. Because of the high-speed data transfer rate capability of the IPB, this arrangement provides what appears to be a full-duplex channel to the programmer. Data is actually transferred over the half-duplex, bi-directional data bus ($\overline{\text{CPB[0-15]}}$) in the IPB external cable.

Data Transfer

Two operations are required to transfer data between processors: first, the transmitting processor sends data to the full-duplex storage buffer by executing a WRITE DATA instruction. Second, the receiving processor retrieves the data stored at the buffer's output by executing a READ DATA instruction.

Transmit

When a WRITE DATA instruction (DOA ac, DPO) is executed, the transmitting processor transfers data from the specified accumulator to the IPB board, via the I/O data bus. If the instruction was initiated by the "Right" processor, the data is loaded directly from the I/O data bus into the storage buffer. If the instruction was initiated by the "Left" processor, the data passes from the I/O data bus, through the general purpose (half-duplex) transmitter, over the **CPB** bus, and into the storage buffer.

Receive

When a READ DATA instruction (DIA ac, DPI) is executed, the receiver passes the contents of the storage buffer to the I/O data bus, and the receiving processor transfers the data into the specified accumulator. If the instruction was initiated by the "Right" processor, the receiver passes the data directly to the I/O data bus. If the instruction was initiated by the "Left" processor, the receiver passes the data to the **CPB** bus. The data passes over the **CPB** bus, through the general purpose (half-duplex) receiver, to the I/O data bus.

PROGRAMMING SUMMARY

FULL-DUPLEX

MNEMONIC (INPUT)	DPI
DEVICE CODE (INPUT)	40_8
PRIORITY MASK BIT (INPUT)	8
MNEMONIC (OUTPUT)	DPO
DEVICE CODE (OUTPUT)	41_8
PRIORITY MASK BIT (OUTPUT)	8

INSTRUCTIONS

READ DATA . . . . . DIA ac, DPI
WRITE DATA . . . . . DOA ac, DPO

FLAG COMMANDS S, C, AND P FUNCTIONS

Input

f = **S** In the computer executing the instruction, the DPI Busy flag is set to 1 and the DPI Done flag is set to 0. If the DPI Busy flag was 1 before this instruction was issued, then, in the other computer, the DPO Busy flag is set to 0 and the DPO Done flag is set to the prior value of the DPO Busy flag.

f = **C** In the computer executing the instruction, the DPI Busy and Done flags are both set to 0. In the other computer, the DPO Busy flag is set to 0 and the DPO Done flag is set to the prior value of the DPO Busy flag.

f = **P** This command has no effect.

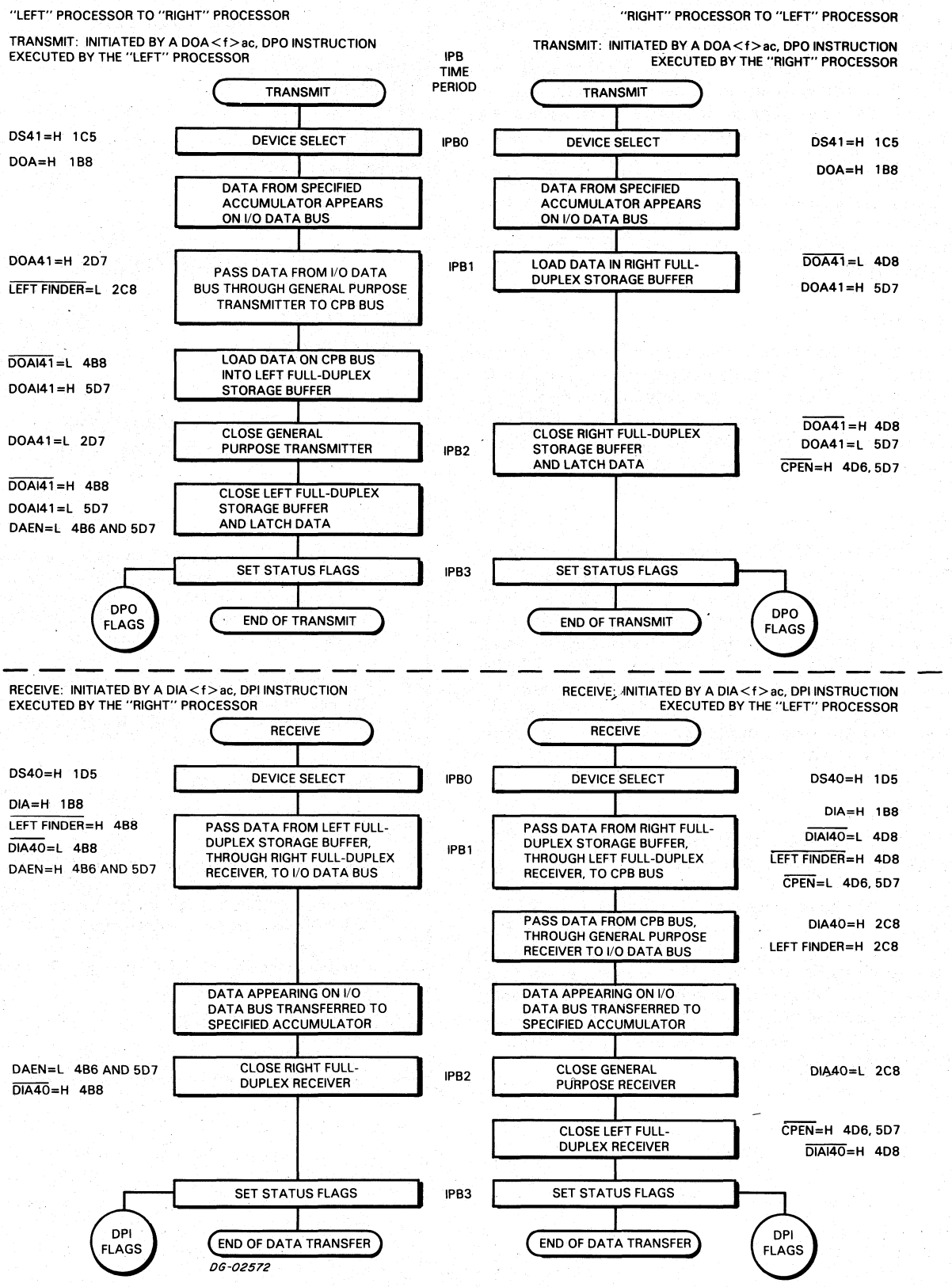
Output

f = **S** In the computer executing the instruction, the DPO Busy flag is set to 1 and the DPO Done flag is set to 0. In the other computer, if the DPI Busy flag is 1, the DPI Done flag is set to 1 and the DPI Busy flag remains unchanged.

f = **C** In the computer executing the instruction, the DPO Busy and Done flags are both set to 0.

f = **P** This command has no effect.

FULL-DUPLEX DATA TRANSFER



NOTE: <f> = OPTIONAL FLAG COMMAND

Status Information

Since all data transfers require the participation of both processors, each processor must be notified of the status of the data transfers. This is accomplished by manipulating the full-duplex transmitter Busy and Done flags in the transmitting processor and the full-duplex receiver Busy and Done flags in the receiving processor. The processors receive status information by means of the program interrupt facility and flag checking software routines.

The receiving processor is notified that the other processor has initiated a data transfer when the receiver Done flag is set to 1. The transmitting processor is notified that the other processor has received the data when the transmitter Done flag is set to 1.

The Busy and Done flags are set by IPB device flag commands, Start and Clear. An I/O pulse has no effect on the Busy and Done flags of the full-duplex devices.

The flags are set in the following manner:

Transmit

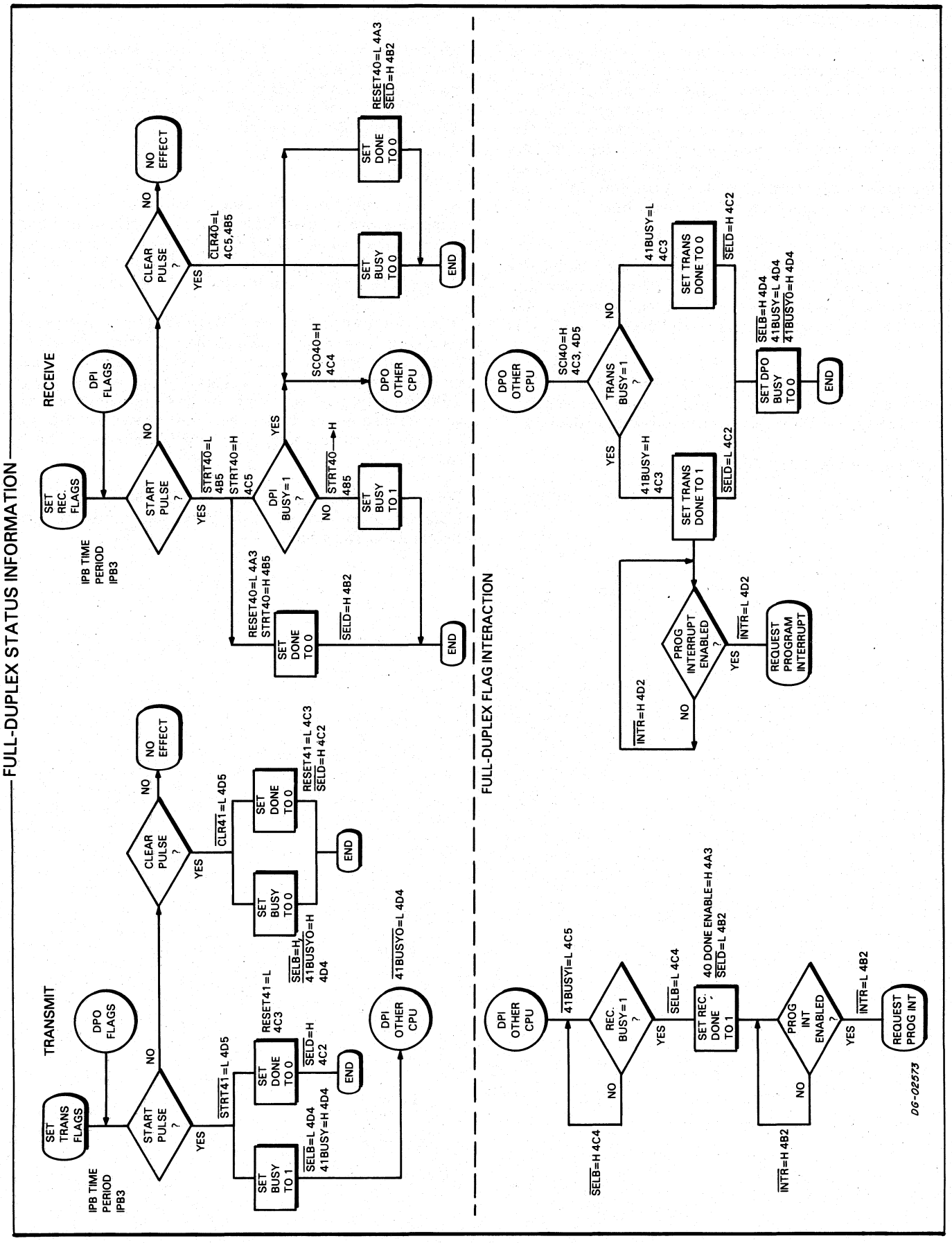
After data is loaded into a full-duplex storage buffer, the transmitting processor issues a Start pulse. This sets the transmitter Busy flag to 1, the transmitter Done flag to 0, and generates the **41BUSYI** signal on the other processor's IPB board.

Receive

When the receiving processor is ready to receive data, it issues a Start pulse. This sets the receiver Busy flag to 1 and the receiver Done flag to 0. When the receiver Busy flag is set to 1 and the **41BUSYI** signal is asserted, the receiver Done flag is immediately set to 1.

After the data in the storage buffer is retrieved, the receiving processor issues either a Start or Clear pulse. This generates the **SCI40** signal on the other processor's IPB board. If the receiving processor issues a Start pulse, the receiver Busy and Done flags are set as described above. If the receiving processor issues a Clear pulse, the receiver Busy and Done flags are both set to 0.

When the **SCI40** signal is asserted, the transmitter Done flag is set to the previous value of the transmitter Busy flag.



HALF-DUPLEX COMMUNICATIONS/INTERLOCKED

The half-duplex communications link provides the facilities for bi-directional, half-duplex communications between processors. Additionally, when the appropriate software is used, the circuitry which governs the setting of the Busy flags of both half-duplex devices allows the program to establish an "interlock".

Each processor controls one half-duplex transmitter/receiver device. While the transmitter/receiver is actually two separate devices, a transmitter (half-duplex storage buffer) and a receiver (half-duplex set of receiver gates), it responds to one device code and can assume only one role at a time. This arrangement allows information to be transferred either from the "Left" processor to the "Right" processor or from the "Right" processor to the "Left" processor.

Data Transfer

Two operations are required to transfer data between processors: first, the transmitting processor sends data to the half-duplex storage buffer by executing a WRITE DATA instruction. Second, the receiving processor retrieves the data stored at the buffer's output by executing a READ DATA instruction.

Transmit

When a WRITE DATA instruction (DOA ac, IPB) is executed, the transmitting processor transfers data from the specified accumulator to the IPB board, via the I/O data bus. If the instruction was initiated by the "Right" processor, the data is loaded directly from the I/O data bus into the storage buffer. If the instruction was initiated by the "Left" processor, the data passes from the I/O data bus, through the general purpose (half-duplex) transmitter, over the **CPB** bus, and into the storage buffer.

Receive

When a READ DATA instruction (DIA ac, IPB) is executed, the receiver passes the contents of the storage buffer to the I/O data bus, and the receiving processor transfers the data into the specified accumulator. If the instruction was initiated by the "Right" processor, the receiver passes the data directly to the I/O data bus. If the instruction was initiated by the "Left" processor, the receiver passes the data to the **CPB** bus. The data passes over the **CPB** bus, through the general purpose (half-duplex) receiver, to the I/O data bus.

PROGRAMMING SUMMARY

HALF-DUPLEX

MNEMONIC	IPB
DEVICE CODE	36_8
PRIORITY MASK BIT	6

INSTRUCTIONS

READ DATA	**DIA ac, IPB**
WRITE DATA	**DOA ac, IPB**
CLEAR FLAGS	**DIB ac, IPB**

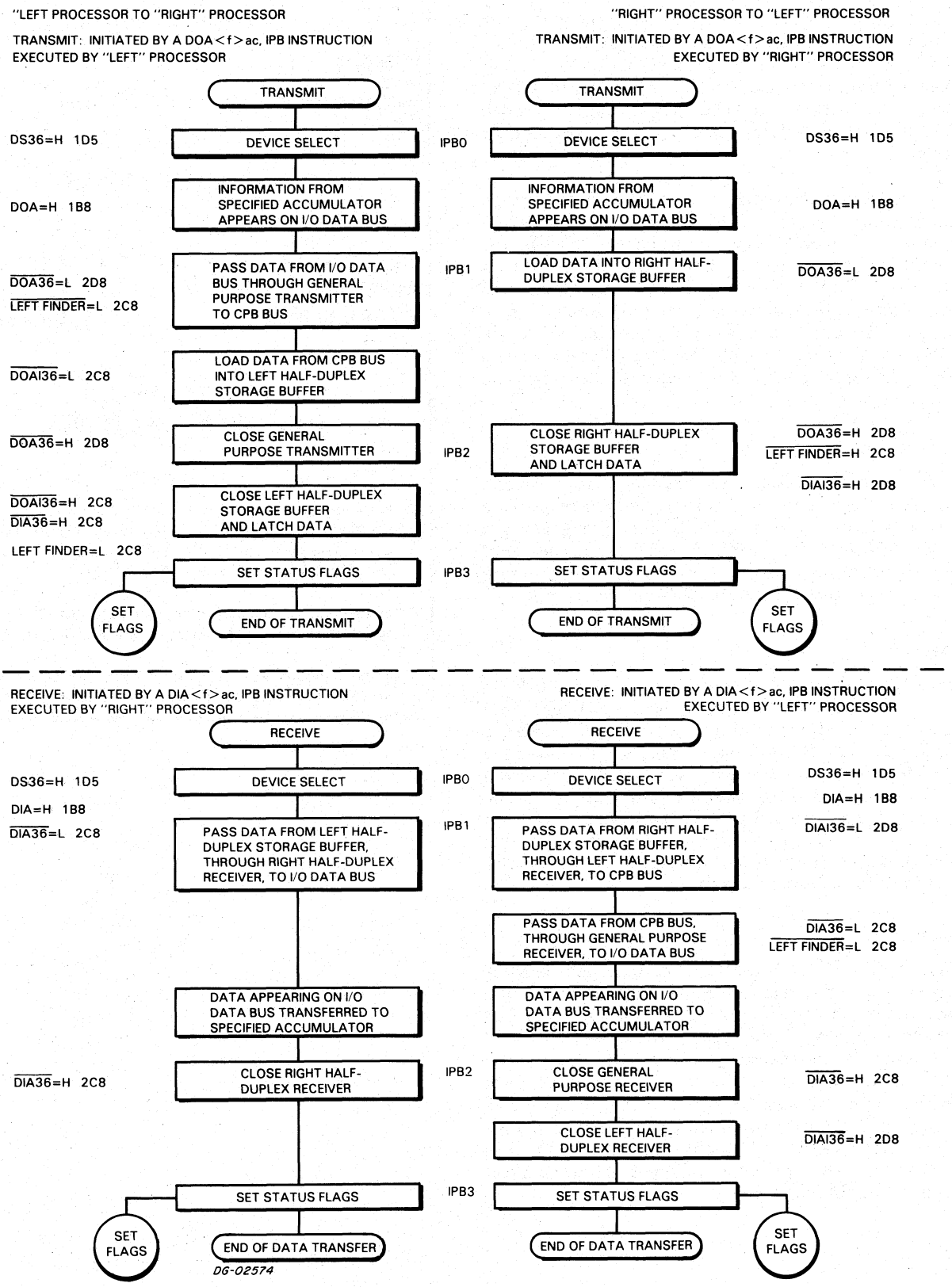
FLAG COMMANDS S, C, AND P FUNCTIONS

S In the computer executing the instruction, the IPB Busy flag is set to 1 if the IPB Busy flag of the other computer is 0. Even if both computers issue a Start at exactly the same time, only one IPB Busy flag will be set to 1.

C In the computer executing the instruction, the IPB Done flag is set to 0. In the other computer, the IPB Busy flag is set to 0.

P In the computer executing the instruction, the IPB Done flag is set to 0. In the other computer, the IPB Done flag is set to 1.

HALF-DUPLEX DATA TRANSFER



NOTE: <f> = OPTIONAL FLAG COMMAND

Status Information

Since all data transfers require the participation of both processors, each processor must be notified of the status of the data transfer. This is accomplished by manipulating the Busy and Done flags of the half-duplex devices of both processors. The processors receive status information by means of the program interrupt facility and flag checking software routines.

The receiving processor is notified that the other processor has initiated a data transfer when its half-duplex device Done flag is set to 1. The transmitting processor is notified that the other processor has received the data when its half-duplex device Done flag is set to 1.

The Busy and Done flags are set by IPB device flag commands, Start, Clear, and I/O Pulse. In addition, a CLEAR FLAGS instruction (DIB ac, IPB) functions as an IPB device flag command in half-duplex communications.

Interlock

The interlock allows only one processor to transmit data when the appropriate software is used, since it allows only one processor's half-duplex Busy flag to be set to 1 at any one time. This is accomplished by double-flag circuitry on the "Right" IPB board which governs the setting of the half-duplex device Busy flag of both processors. This circuitry is disabled on the "Left" IPB board by the $\overline{\text{LEFT FINDER}}$ signal.

The double-flag circuitry contains two gates; only one of these gates can be enabled at any one time. When one gate is enabled, it sets the "Left" processor's Busy flag to 1; when the other gate is enabled, it sets the "Right" processor's Busy flag to 1. A gate is enabled when the following conditions are satisfied.

- A Start pulse is issued by the appropriate processor.
- The other processor's Busy flag is set to 0.
- The gate is enabled by the output of the "W" and "X" flip-flops. These flip-flops are clocked by a 20MHz oscillator and their outputs form a 4-phase, 5MHz oscillator. The outputs of the flip-flops alternately enable the two Busy flag gates.

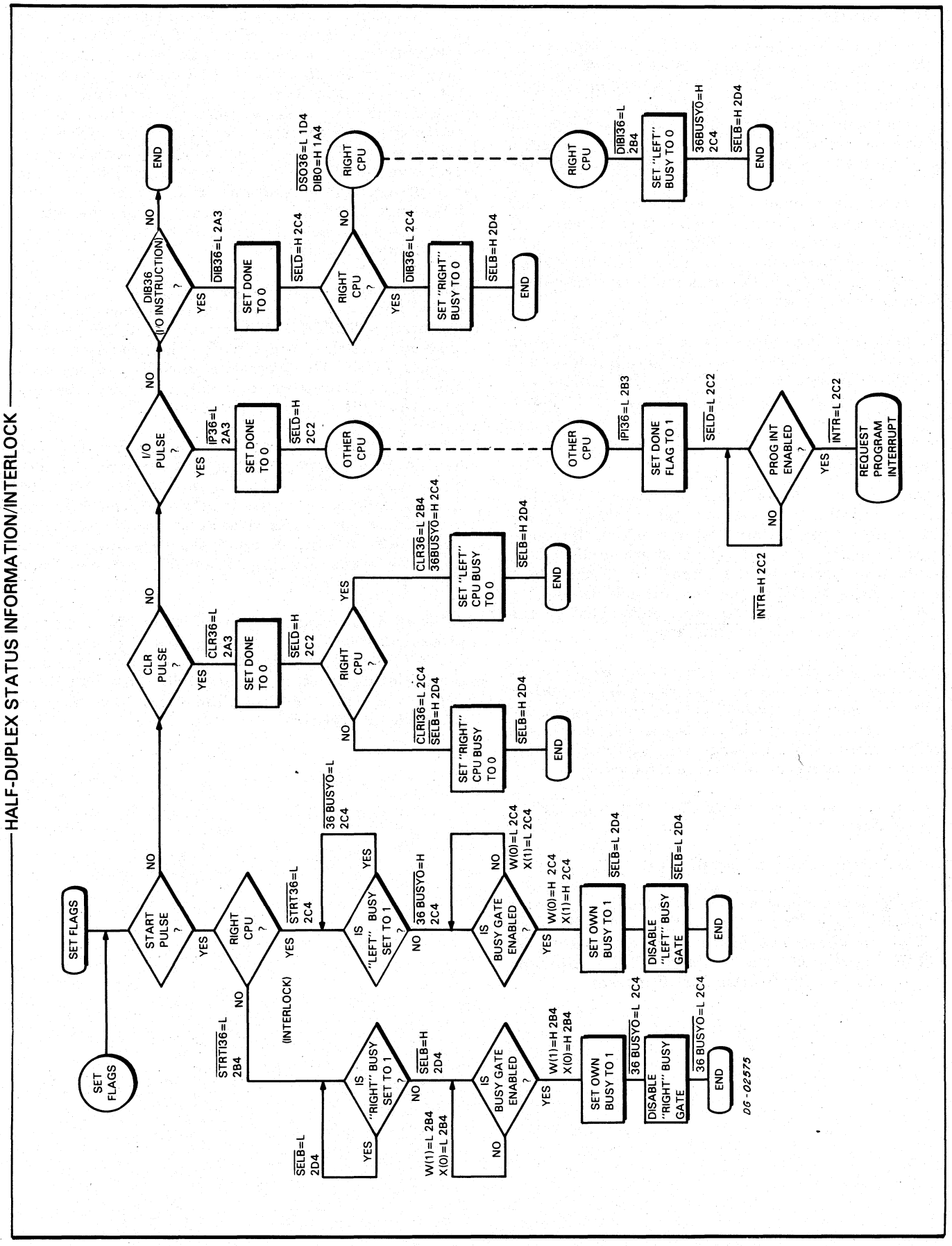
To establish an interlock, the initiating processor must first execute a REQUEST BUS instruction (NIOS IPB), which generates a Start pulse. Next, the processor must execute an I/O Skip instruction to test the state of the $\overline{\text{SELB}}$ line of the I/O bus. If the half-duplex Busy flag is set to 1, the interlock is established and the data transfer can be initiated. If the Busy flag is set to 0, it indicates that the other processor has established an interlock in the half-duplex communications link. In this case, the initiating processor must wait until the other processor releases the interlock by setting its half-duplex Busy flag to 0.

After the initiating processor establishes an interlock, it transfers data to the half-duplex storage buffer by executing a WRITE DATA instruction. After the data is loaded into the storage buffer, the processor issues an I/O Pulse. This sets the initiating processor's half-duplex Done flag to 0 and sets the receiving processor' half-duplex Done flag to 1.

When the receiving processor is notified that its half-duplex Done flag is set to 1, it retrieves the data by executing a READ DATA instruction. After the data in the storage buffer is retrieved, the receiving processor issues an I/O Pulse. This sets the receiving processor's half-duplex Done flag to 0 and sets the initiating (transmitting) processor's half-duplex Done flag to 1.

After all data transfers are completed, the initiating processor releases the interlock by executing a CLEAR FLAGS instruction (DIB ac, IPB). This sets the initiating processor's half-duplex Busy and Done flags to 0; thus, allowing the other processor to establish an interlock.

HALF-DUPLEX STATUS INFORMATION/INTERLOCK



WATCHDOG TIMER

The Watchdog Timer provides the facilities for generating a program interrupt in either computer when the other computer: 1) fails to execute a specific instruction; or, 2) generates a signal indicating a probable "power fail" condition. This capability is implemented by two missing pulse detectors, one located on each IPB board, and each detector's Timer Done flag.

Each detector consists of a one-second timer which is normally restarted by the execution of a START TIMER instruction (NIOS IVT) in the other computer at regular intervals of one second, or less. When one computer fails to restart the timer in the other computer within the specified time period, that detector's Timer Done flag is set to 1, generating a program interrupt request in the computer in which the detector is installed.

When either computer knows that it is about to cease functioning (e.g., a power fail situation), that computer may set the Timer Done flag in the other computer directly by executing a SET TIMER instruction (NIOP IVT).

When the processor is notified that the Timer Done flag is set to 1, it should take appropriate action and then execute a CLEAR TIMER instruction (NIOC IVT), setting the Timer Done flag to 0. If this instruction or an I/O Reset is not executed, the processor will continue to be notified of the failure.

PROGRAMMING SUMMARY

MNEMONIC	IVT
DEVICE CODE	37_8
PRIORITY MASK BIT	6
TIMER PERIOD (SEC)	1

INSTRUCTION

START OWN TIMER . . . . . DOA ac, IVT

FLAG COMMANDS S, C, AND P FUNCTIONS

S The timer in the other computer is started.

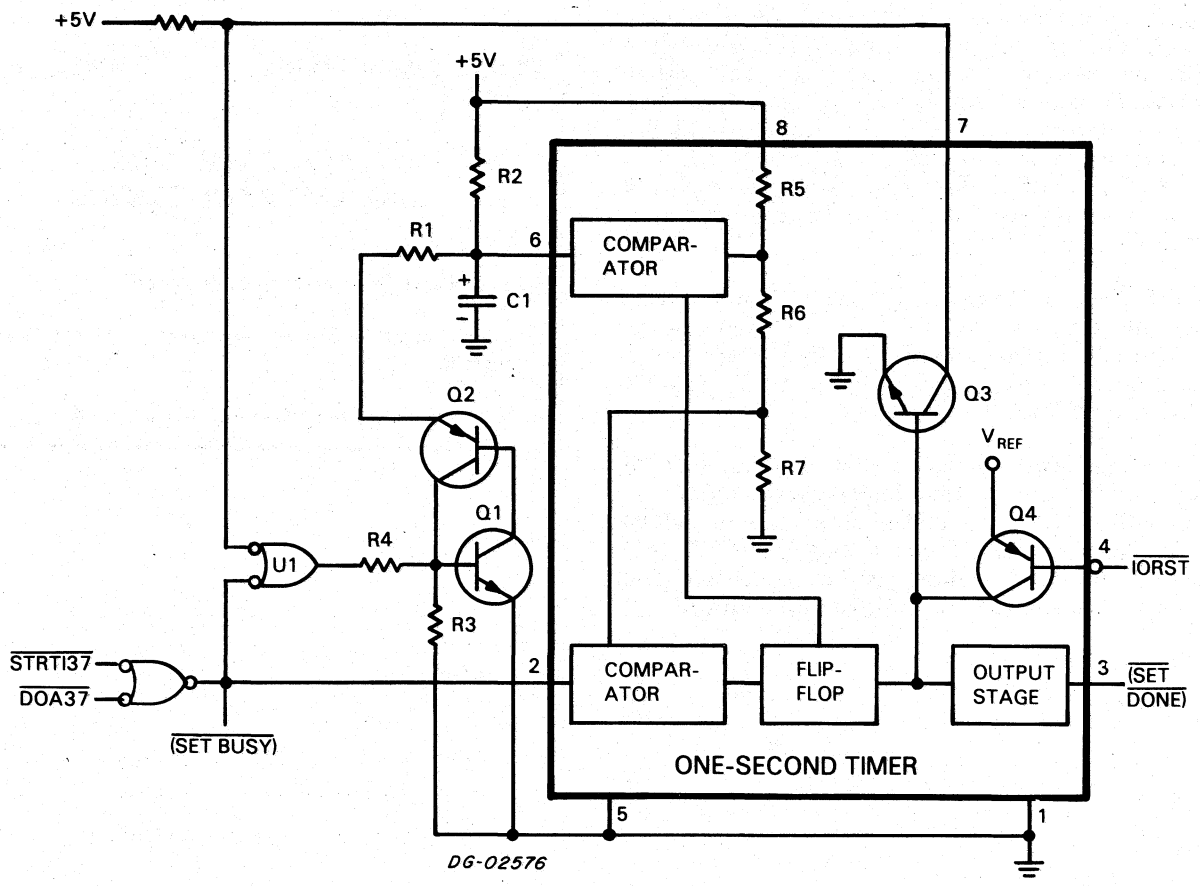
C In the computer executing the instruction, the IVT Done flag is set to 0.

P In the other computer, the IVT Done flag is set to 1.

The Timer

The one-second timer functions as a missing pulse detector. After the timer is initially started, it must be restarted at intervals of one second, or less, in order to keep its output (normally high) from changing. If the timer is allowed to run, its output changes when a comparator in the timer chip senses a predefined voltage which is stored in an external capacitor. When this occurs, the Timer Done flag is set to 1.

TIMER CIRCUIT



TIMER CIRCUIT - Either the **STRTI37** signal, generated by the execution of a START TIMER instruction in the other processor, or a **DOA37** signal, generated by the execution of a START OWN TIMER instruction (DOA ac, IVT) in the timer's processor, starts the timer and sets its Busy flag to 1. (The Timer Busy flag does not assert the **SELB** line of the I/O bus and cannot be tested by flag checking.) Either of these signals enables the timer's external gate U1, passing current to the transistor Q1. The transistor Q1 turns on transistor Q2, allowing the external capacitor C1 to discharge through the resistor R1 to ground. This keeps the output of the timer high and the Timer Done flag set to 0. (The Done flag is set to 0 after a Clear pulse or an I/O reset is issued by the timer's processor.)

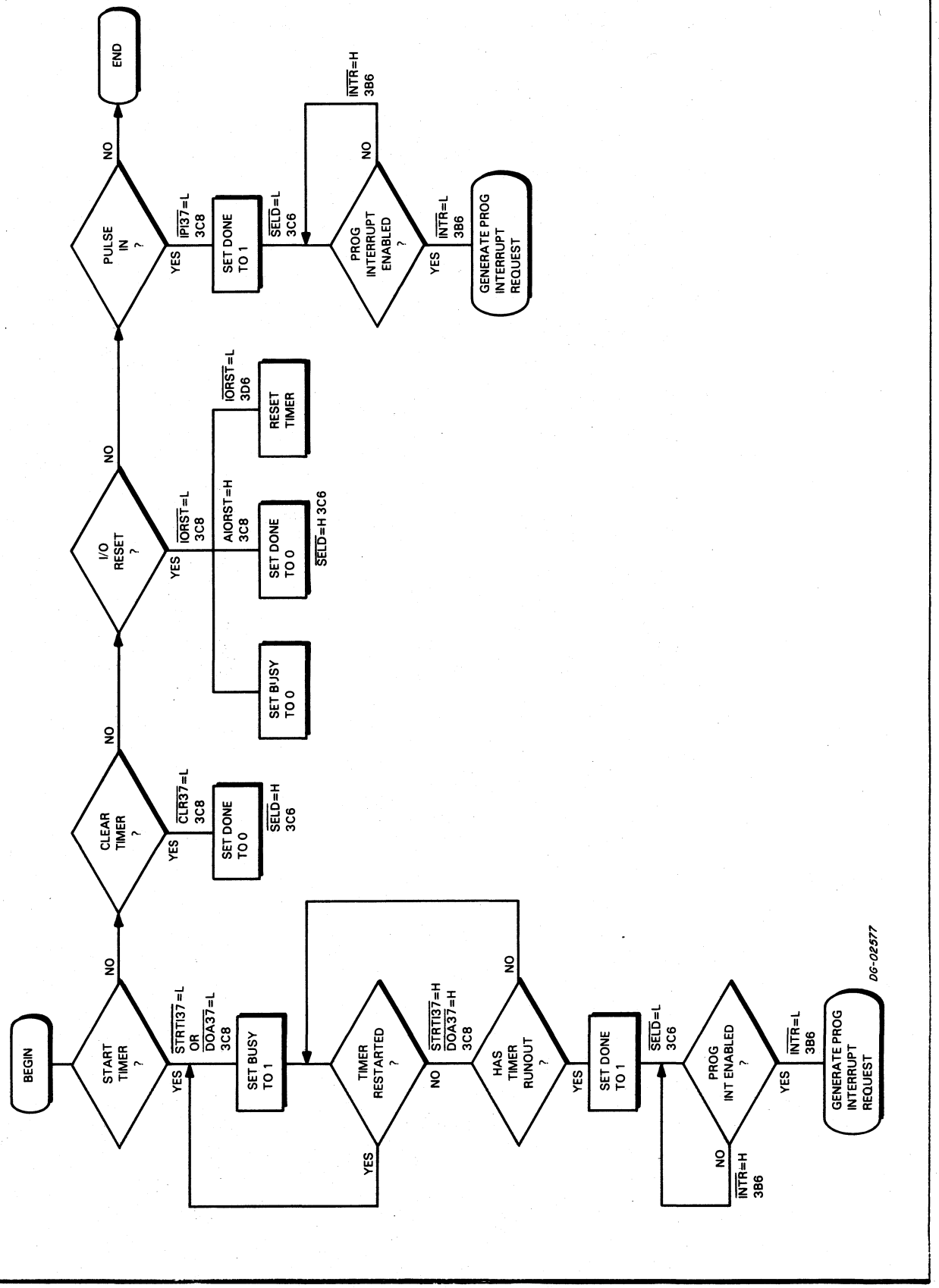
When the instruction ends, the transistors are turned off and the capacitor C1 charges through resistor R2.

If the timer is not restarted in approximately one second, the timing cycle is allowed to run and the external capacitor continues to charge. When the capacitor C1 reaches a predefined limit, the comparator senses the voltage across the capacitor, causing a flip-flop in the timer chip to toggle, and the output of the timer goes low. When this occurs, the value of the Busy flag (Busy is set to 1 after either a **STRTI37** or **DOA37** signal is asserted) is clocked into the Timer Done flip-flop, setting the Done flag to 1.

In order for the Timer Done flag to reflect the functional status of the other processor, the timer must be restarted by the **STRTI37** signal.

The **IORST** signal directly sets the Timer Done flag to 0, sets the Busy flag to 0, and resets the timer. When the timer is reset, the timer output goes high. Additionally, the **IORST** signal enables transistor Q3 in the timer chip, causing the +5 line to go low. When this occurs, the external gate U1 is enabled, allowing the capacitor C1 to discharge to ground.

WATCHDOG TIMER



This page intentionally left blank.

APPENDIX A
SPECIFICATIONS

4240 Inter-Processor Bus Printed Circuit Board

Size: 15" x 15"

Power Requirements: approximately 2.5 amps, 5Vdc

Space Requirements: one I/O slot in a NOVA or ECLIPSE line computer

Items Supplied on Purchase: one 15" x 15" IPB printed circuit board, one internal cable, one test plug, and one documentation package. (When ordering, specify the two computer models.)

1065 External Cable for the 4240 Inter-Processor Bus

Length: 15'

Contains 78 lines: 39 signal levels and 39 grounds

When ordering, specify the two computer models.

This page intentionally left blank.

CUT ALONG DOTTED LINE

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

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

[redacted]