Model PE-730

Memory Expansion Unit

Technical Manual

Document Number: 600-265-00
Revision: A
Date: 3/4/86
Serial No.:

NOTICE

ZETACO, Inc. has prepared this Technical Manual for use by ZETACO personnel and its customers to assist in the installation, operation and maintenance of the Model PE-730 Memory Expansion Unit. The information contained herein is the property of ZETACO, Inc. and shall not be reproduced in whole nor in part without prior written approval of ZETACO, Inc.

ZETACO, Inc. makes every effort to produce quality products and documentation, however, the reader must be cautioned that changes may be made to the product not reflected in the supplied documentation. If you find errors or omissions, please notify ZETACO, Inc. to remedy the problem. ZETACO, Inc. shall not be responsible for any damages (including consequential) caused by reliance on this material as presented.

If installation problems arise after you thoroughly review the manual, please contact the ZETACO Customer Support Hotline at (612) 941-9480.

Sector Slip™ and Virtual Mapping™ are tradenames of ZETACO, Inc.

Eclipse®, Nova®, and MV® are registered trademarks of Data General Corporation.

FAST™ is a trademark of Fairchild Camera & Instrument Corporation.

REVISION HISTORY

ECO No.	Date	Description	Pages
0328	6/28/84	New ZETACO Cover	
0523	3/4/86	New Cover	

CUSTOMER SUPPORT HOTLINE

ZETACO, Inc. provides a Customer Support Hotline (612-941-9480) to answer technical questions and to assist with installation and trouble-shooting problems. The Hotline is manned by a technical team from 8:00 a.m. to 5:00 p.m. (Central Time) Monday through Friday.

WARRANTY INFORMATION

All ZETACO products are warranted free from manufacturing and material defects, when used in a normal and proper manner, for a period of up to two years from date of shipment. Except for the express warranties stated above, ZETACO disclaims all warranties including all implied warranties of merchantability and fitness. The stated express warranties are in lieu of all obligations of liabilities on the part of ZETACO for damages, including but not limited to, special, indirect or consequential arising out of or in connection with the use or performance of ZETACO's products.

PRODUCT RETURN AUTHORIZATION

All possible effort to test a suspected malfunctioning controller should be made before returning the controller to ZETACO for repair. However, if controller or module malfunction has been confirmed, you should return the part to ZETACO. If the part is no longer under warranty, or if the problem is not warranted, then repair will be on a time-and-material basis. A Return Material Authorization (RMA) number is required before shipment and should be referenced on all packaging and correspondence.

To ensure prompt response, the information outlined in the Material Return Information form on the following page should be gathered before calling the ZETACO Hotline for the RMA number. Please include a completed copy of the Material Return Information form with the product. Each product to be returned requires a separate RMA number and Material Return Information form.

To safeguard the product during shipment, please use packaging that is adequate to protect it from damage. Mark the box "Delicate Instrument" and indicate the RMA number(s) on the shipping label.

MATERIAL RETURN INFORMATION

All possible effort to test a suspected malfunctioning controller should be made before returning the controller to ZETACO, Inc. for repair. The speed and accuracy of a product's repair is often dependent upon a complete understanding of the user's checkout test results, problem characteristics, and the user system configuration. Use the form below to record the results of your trouble-shooting procedures. If more space is needed, use additional sheets.

FUNCTION	TEST	RESULT
Expansion	DG N3MMU	_____
Multiply/Divide	DG MDV TST	_____
Parity	DG N3 PRTY	_____
Protect	DG MMPU	_____
All	DG NMORTL	_____

Other tests performed:

Please allow our service department to do the best job possible by answering the following questions thoroughly and returning this information with the malfunctioning board.

1. Does the problem appear to be intermittent or heat sensitive? (If yes, explain.)

2. Under what operating system are you running? (AOS, AOS/VS, RDOS, etc.)

3. Describe the system configuration (i.e.; peripherals, controllers, model of computer, etc.)

4. Has the unit been returned before? Same problem?

To be filled out by CUSTOMER:

Model #: _____
Serial #: _____
RMA #: _____ (Call ZETACO to obtain an RMA number.)

Returned by:

Your name: _____
Firm: _____
Address: _____
_____
Phone: _____

TABLE OF CONTENTS

1.0 General Description

The Custom Systems Model 730 Memory Expansion unit contains any combination of the following NOVA 3 processor features:

Feature 732	Memory Expansion
Feature 734	Memory Expansion and Protection
Feature 736	Memory Parity Generator/Checker
Feature 738	Hardware Multiply/Divide

1.1 Memory Expansion

The Memory Expansion feature is required to support memory sizes above 32K words to a maximum of 512K words. The Memory Expansion feature translates the normal 15-bit memory logical address (32K) to a 19 bit physical address (512K). The 19 bit physical address is grouped in four, thirty two 9 bit registers called maps. The four maps are named Program Map A, Program Map B, Data Channel Map A, and Data Channel Map B.

Each time the CPU accesses memory, the Memory Expansion feature is either not active (in which case the physical address corresponds to the logical address) or the Memory Expansion unit is performing logical to physical address translation according to one of the four maps.

NOTE: D.G. Standard Software will only address 128K words of memory. To use the 512K words capability the customer must write special Software.

1.2 Memory Expansion and Protection

The prerequisite for the Protection feature is the Memory Expansion feature. The Protection feature detects as many as five varieties.

of map violations and affords systems protection from these violations. The five types of violations are:

(1) Validity Violation
(2) Write Protect Violation
(3) I/O Violation
(4) Defer Violation
(5) Autoindex Violation

When any map violation is detected, the Protect option and the CPU trap the violation.

NOTE: When Memory Protect option is installed on the board the memory capacity is limited to 128K words.

1.3 Memory Parity Generator/Checker

The Memory Parity feature detects an odd number of bit errors between a word written into a Parity Memory and a word read from a Parity Memory. It performs this by generating a parity bit when a word is written into a Parity Memory and then checking the parity bit when the word is read from the Parity Memory. The parity bit generated can correspond to either odd or even parity selectable under program control. When a parity error occurs the parity feature will either (1) initiate a program interrupt sequence, or (2) light the parity console light and place the computer in an idle state until the console Reset switch is activated.

Both Parity and non Parity (Core) Memory can be intermixed within the same system. The Parity Generator/Checker feature will not be active for non Parity Memory Boards.

1.4 Hardware Multiply/Divide

The Multiply/Divide feature provides for high speed arithmetic operations on unsigned integers. Significant improvements in CPU efficiency and utilization over software routines can be achieved, with the addition of this feature, expecially when system applications involve a large number of computations.

2.0 Installation

This section provides detailed information for installing the ZETACO Series 730 Memory Expansion Unit.

Inspect the Controller Boards for any intransit damage. Contact the involved carrier and ZETACO, Inc. if any damage is discovered, specifying the nature and extent of the damage.

Before installing the board remove power from the NOVA 3 and return power when the Memory Expansion Board is secured in Slot 2.

Installation of the Memory Expansion Board involves inserting the board into Slot 2 of the NOVA 3 and locking it into place with the release levers.

3.1 MEMORY EXPANSION (DATA GENERAL MMU)

The memory expansion option expands the memory addressing capabilities from 32K words to 512K words. It accomplishes this by translating every 15 bit address (32K logical) to an 19 bit address (512K physical) with no change in memory access time.

The memory expansion option performs this logical-to-physical address translation by replacing the five most significant bits of a logical address with 9 bits of a corresponding physical address. No logical to physical translation occurs on the ten least significant bits. This logical to physical address translation is performed such that for every logical address there is one and only one physical address.

The relation between the five most significant bits of a logical address and the nine most significant bits of a physical address is determined by the contents of four groups of thirty-two nine bit registers called maps.

The memory expansion option has four maps:

1. Program Map A
2. Program Map B
3. Data Channel Map A
4. Data Channel Map B

The selection of which data channel map is to be used is under control of the peripheral controllers. Those peripheral controllers not equipped to make this distinction will use data channel map "A" by default.

The two program maps and the two data channel maps are completely independent. Only one program map may be enabled at a time, but both data channel maps are enabled at the same time. The mapping of program addresses and the mapping of data channel addresses may or may not be enabled at the same time depending upon the wishes of supervisor program. If either program mapping or data channel mapping is disabled then, for that function, the physical address space is equal to the logical address space and only the lowest 32K words of memory are accessible.

When power is first turned on, or after a Clear command to device code 3, both the program map and data channel map portions of the MMU are disabled. The physical address space is equal to the logical address space and only the lowest 32K words of memory are accessible.

PROGRAMMING NOTES

The instructions for the MMU are in the standard I/O format. The MMU takes two device codes: 2 and 3. The mnemonic for device code 2 is MMU. The mnemonic for device code 3 is MMU1.

Device code 2 has a Done flag which is set to 1 by the MMU any time address translation is enabled and not inhibited. Device code 3 does not have a Busy or Done flag.

The flag control commands for device code 2 are as follows:

f = S Reserved for future use.

f = C Reserved for future use.

f = P The second non-data channel memory address after the issuance of this command is mapped using the map indicated by the Single Cycle Select bit in the MMU status word.

The flag control commands for device code 3 are as follows:

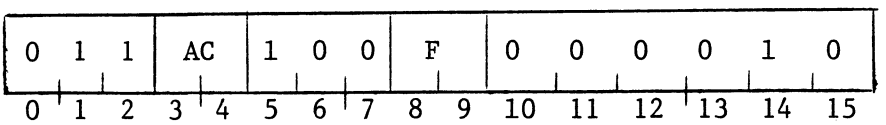
f = S Reserved for future use.

f = C The program map and data channel map portions of the MMU are disabled. All internal MMU logic is initialized.

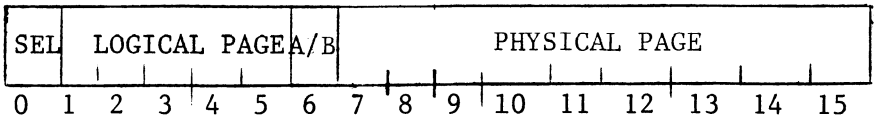
f = P Reserved for future use.

LOAD MAP

DOB <f> ac,MMU



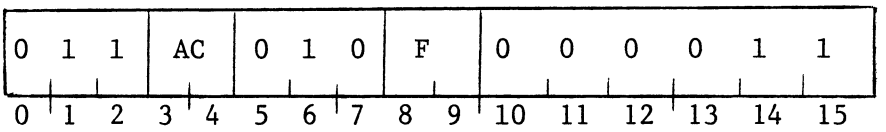
The contents of the specified AC are transferred to the MMU. The contents of the specified AC remain unchanged. The format of the AC is as follows:



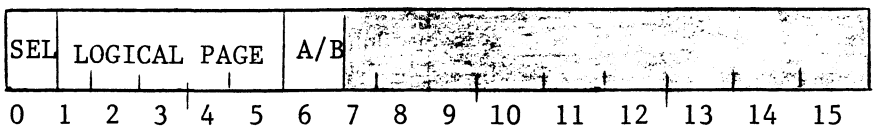
BITS	CONTENTS
0	0 = this instruction gives an address translation for the CPU (program map). 1 = this instruction gives an address translation for the data channel (data channel map).
1-5	Logical page number. This is an octal number in the range 0-37.
6	0 = this instruction gives an address translation for map "A" of the map indicated by bit 0. 1 = this instruction gives an address translation for map "B" of the map indicated by bit 0.
7	Physical page number. This is an octal number in the range 0-777.

INITIATE PAGE CHECK

DOA <f> ac,MMU1



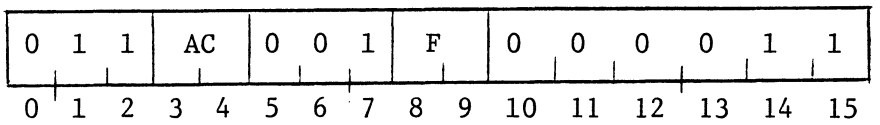
The contents of the specified AC are transferred to MMU for later use by the PAGE CHECK instruction. The contents of the specified AC remain unchanged. The format of the AC is as follows:



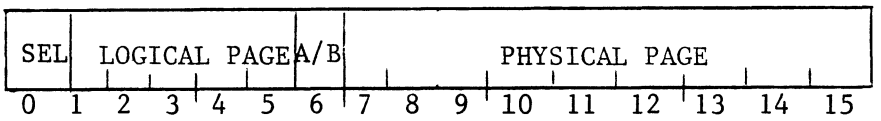
BITS	CONTENTS
0	0 = page check is for a program map. 1 = page check is for a DCH map.
1-5	Logical page. This is an octal number in the range 0-37 and is the number of the logical page for which the check is requested.
6	0 = page check is for map "A" of the map indicated by bit 0. 1 = page check is for map "B" of the map indicated by bit 0.
7-15	Reserved for future use. Should be 0.

PAGE CHECK

DIA <f> ac,MMU1



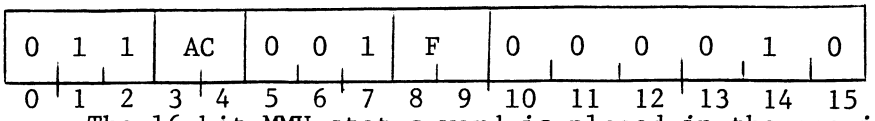
The number of the physical page which corresponds to the logical page number given in the last INITIATE PAGE CHECK instruction is placed in bits 7-15 of the specified AC. The format of the specified AC is as follows:



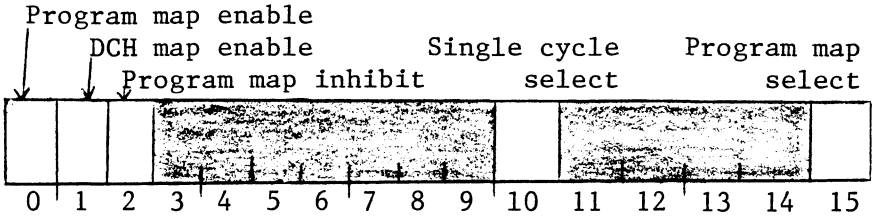
BITS	CONTENTS
0-6	Bits 0-6 from the last INITIATE PAGE CHECK instruction.
7-15	Physical page. This is an octal number in the range 0-177 and is the number of the physical page which corresponds to the logical page given in the last INITIATE PAGE CHECK instruction.

READ MMU STATUS

DIA <f> ac,MMU



The 16-bit MMU status word is placed in the specified AC. The format of the AC is as follows:

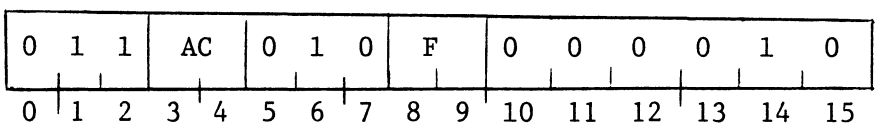


BITS	CONTENTS
0	0 = program mapping is disabled.
	1 = program mapping is enabled.
1	0 = data channel mapping is disabled.
	1 = data channel mapping is enabled.
2	0 = program mapping is not inhibited.
	1 = program mapping is inhibited. If set, this bit takes precedence over bit 0.
3-9	Reserved for future use. Set to 0.
10	0 = single cycle mapping will use program map "A".
	1 = single cycle mapping will use program map "B".
11-14	Reserved for future use. Set to 0.
15	0 = program mapping will be done with program map "A".
	1 = program mapping will be done with program map "B".

NOTE: The Program Map Inhibit bit is set by a stack overflow, I/O interrupt, or execution of a TRAP instruction.

WRITE MMU STATUS

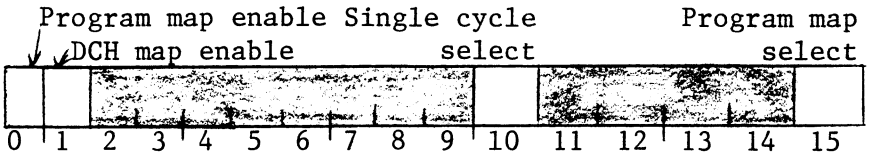
DOA <f> ac,MMU



The contents of the specified accumulator are placed in the MMU status word. The Program Map Inhibit bit in the MMU status word is set to 0.

The new settings of the Program Map Enable bit, the Program Map Inhibit bit, and the Program Map Select bit are compared to the settings of these bits before the instruction was issued. If any of these has changed, none of them takes effect until the memory cycle after the next defer cycle. All three of the bits take effect at that time. This allows the program to change the settings of these bits and then transfer control to the new environment in an orderly manner.

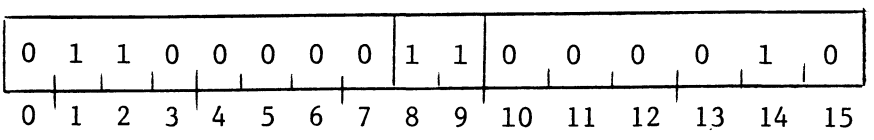
The format of the specified AC is as follows:



BITS	CONTENTS
0	0 = program mapping will be disabled.
	1 = program mapping will be enabled.
1	0 = data channel mapping will be disabled.
	1 = data channel mapping will be enabled.
2-9	Reserved for future use. Should be 0.
10	0 = single cycle mapping will use program map "A".
	1 = single cycle mapping will use program map "B".
11-14	Reserved for future use. Should be 0.
15	0 = program mapping will be done with program map "A".
	1 = program mapping will be done with program map "B".

MAP SINGLE CYCLE

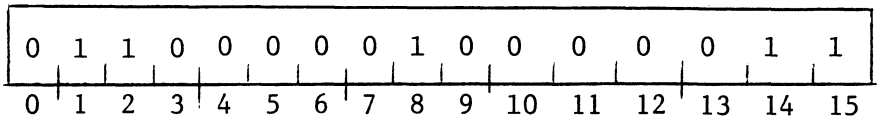
NIOP MMU



The second non-data channel memory reference after this instruction is issued is mapped with the user map indicated by the Single Cycle Select bit in the MMU status word.

CLEAR MAP

NIOC MMU1



The program and data channel maps are disabled, and all internal MMU logic is initialized.

3.2 MEMORY PROTECT (DATA GENERAL MMPU)

The prerequisite for the Memory Protection option is the Memory Expansion option. With the Memory Protection option five types of protection features are available and must be enabled via the software (See Programming Notes).

1. I/O Protection

   If I/O protection is enabled it protects all I/O devices except those using device Codes 1, 74, 75 and 76. Device Code 1 is generally assigned to the multiply/divide option, and device Codes 74-76 are generally assigned to the floating point option. The I/O devices using these device codes are not protected by I/O protection under any circumstances.

   When I/O protection is enabled, the system decodes all I/O instructions to see if the referenced device is user protected. If it is, the protect option does not allow the execution of the instruction. Instead it stores the logical address of the instruction and the system enters the supervisor mode.

2. Validity Protection

   By convention, validity protection cannot be disabled. A logical page is validity protected by mapping the page to physical page 177_8 and setting the validity protect and write protect bits.

   NOTE: It is not necessary for physical page 177_8 to exist. Validity protection is indicated by setting the physcial page bits to 177_8 and setting the validity protect and write protect bits. Since validity protection prevents the writing of the page, the existence of the physical page is not required.

   The protect option checks all CPU request for invalid addresses. If the address is valid the required translation proceeds. If the address is invalid the protect option stores the logical address of the instruction, and the system enters the supervisor mode.

3. Runaway Defer Protection

   If runaway defer protection is enabled, the protect option checks memory references to see if they are part of a defer cycle. If the protect option detects 15 consecutive defer cycle memory requests, it traps. Upon receiving the 15 requests, the protect option stores the address of the instruction that started the defer loop, and the system enters the supervisor mode.

4. Write Protection

   If write protection is enabled, the protect option monitors all modify memory requests and determines whether or not that logical page is write protected. If the page is not write protected, the protect option allows the operation to proceed. If the page is write protected, the protect option stores the instruction address, and the system enters the supervisor mode. Single cycle write protection works in the same way as normal write protection, but it can be enabled separately.

5. Auto - Increment/Decrement Protection

   If Auto-Increment/Decrement protection is enabled, and indirect reference to memory locations $20\text{-}37_8$ will be considered a violation and will therefore trap. The system then stores the logical address, of the instruction that caused the violation, and the system enters the supervisor mode.

PROGRAMMING NOTES

The instructions for the MMPU are in the standard I/O format. The MMPU takes two device codes: 2 and 3. The mnemonic for device code 2 is MAP. The mnemonic for device code 3 is MAP1.

Device code 2 has a Done Flag which is set to 1 by the MMPU any time address translation is enabled and not inhibited. Device code 2 also has a Busy Flag which is set when a Data Channel error occurs. Device code 3 does not have a Busy or a Done Flag.

The flag control commands for device code 2 are as follows:

f = S Reserved for future use.

f = C Clear violation status word.

f = P The second non-data channel memory address after the issuance of this command is mapped using the map indicated by the Single Cycle Select bit in the MMPU status word.

The flag control commands for device code 3 are as follows:

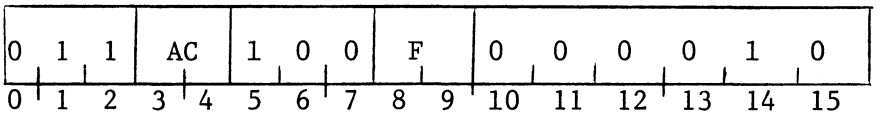
f = S Reserved for future use.

f = C The program map and data channel map portions of the MMPU are disabled. All internal MMPU logic is initialized.

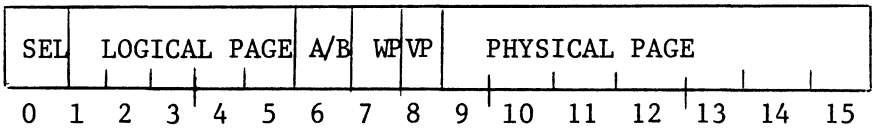
f = P Reserved for future use.

LOAD MAP

DOB <f> ac,MAP



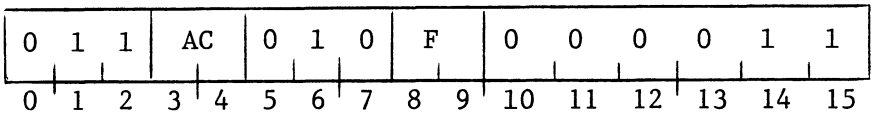
The contents of the specified AC are transferred to the MMPU. The contents of the specified AC remain unchanged. The format of the AC is as follows:



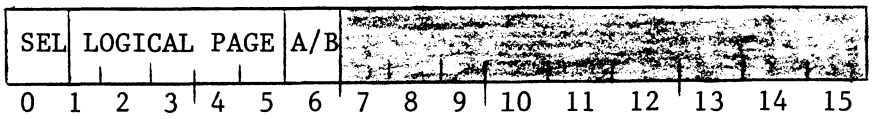
BITS	CONTENTS
0	0 = this instruction gives an address translation for the CPU (program map). 1 = this instruction gives an address translation for the data channel (data channel map).
1-5	Logical page number. This is an octal number in the range 0-37.
6	0 = this instruction gives an address translation for map "A" of the map indicated by bit 0. 1 = this instruction gives an address translation for map "B" of the map indicated by bit 0.
7	0 = write protect disabled. 1 - write protect enabled.
8	0 = validity protect disabled. 1 = validity protect enabled (if bits 7 and 9-15=1).
9-15	Physical page number. This is an octal number in the range 0-177.

INITIATE PAGE CHECK

DOA <f> ac,MAP1



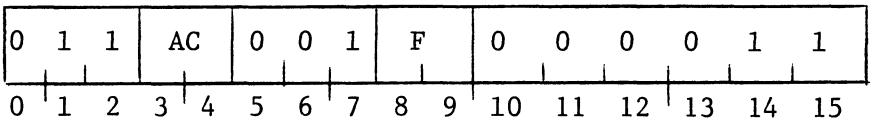
The contents of the specified AC are transferred to the MMPU for later use by the PAGE CHECK instruction. The contents of the specified AC remain unchanged. The format of the AC is as follows:



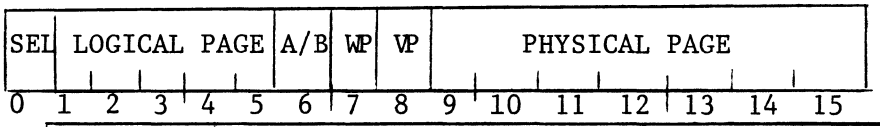
BITS	CONTENTS
0	0 = page check is for a program map. 1 = page check is for a DCH map.
1-5	Logical page. This is an octal number in the range 0-37 and is the number of the logical page for which the check is requested.
6	0 = page check is for map "A" of the map indicated by bit 0. 1 = page check is for map "B" of the map indicated by bit 0.
7-15	Reserved for future use. Should be 0.

PAGE CHECK

DIA <f> ac,MAP1



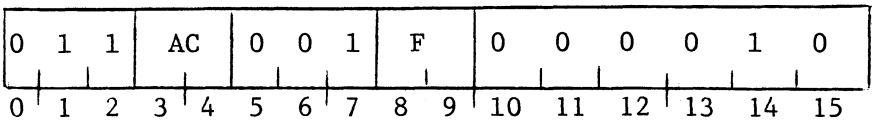
The number of the physical page which corresponds to the logical page number given in the last INITIATE PAGE CHECK instruction is placed in bits 9-15 of the specified AC. The format of the specified AC is as follows:



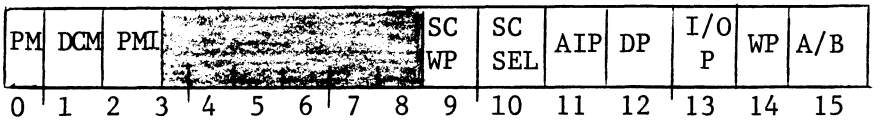
BITS	CONTENTS
0-6	Bits 0-6 from the last INITIATE PAGE CHECK instruction.
7	0 = write protect disabled. 1 = write protect enabled.
8	0 = validity protect disabled. 1 = validity protect enabled (if bits 7 and 9-15=1).
9-15	Physical page. This is an octal number in the range 0-177 and is the number of the physical page which corresponds to the logical page given in the LAST INITIATE PAGE CHECK instruction.

READ MMPU STATUS

DIA <f> ac,MAP



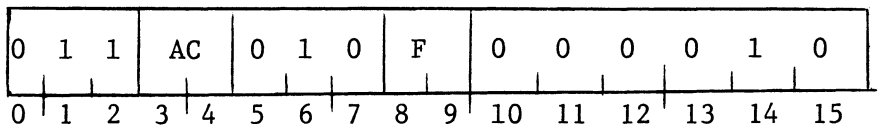
The 16-bit MMPU status word is placed in the specified AC. The format of the AC is as follows:



BITS	MEANING WHEN 1
0	Program mapping enabled.
1	Data channel mapping enabled.
2	Program map inhibited. Takes precedence over bit 0.
3-8	Reserved for future use. Set to 0.
9	Single cycle write protect enabled.
10	Single cycle select. 0 = A, 1 = B.
11	Auto increment/decrement protect enabled.
12	Defer protect enabled.
13	I/O protect enabled.
14	Write protect enabled.
15	Program map select. 0 = A, 1 = B.

WRITE MMPU STATUS

DOA <f> ac,MAP

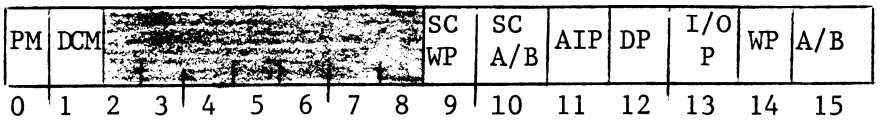


The contents of the specified accumulator is placed in the MMPU status word. The Program Map Inhibit bit in the MMPU status word is set to 0.

The new settings of the Program Map Enable bit, the Program Map Inhibit

bit, and the Program Map Select bit are compared to the settings of these bits before the instruction was issued. If any of these has changed, none of them takes effect until the memory cycle after the next defer cycle. All three of the bits take effect at that time. This allows the program to change the settings of these bits and then transfer control to the new environment in an orderly manner.

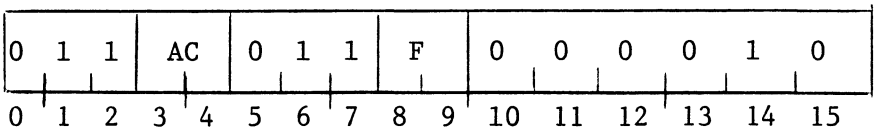
The format of the specified AC is as follows:



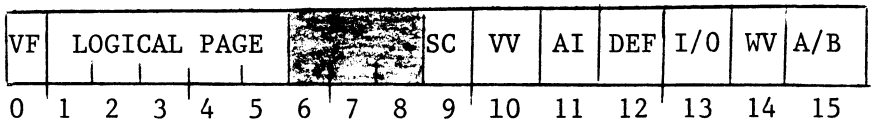
BITS	MEANING WHEN 1
0	Program mapping enabled.
1	Data channel mapping enabled.
2-8	Reserved for future use. Set to 0.
9	Single cycle write protect enabled.
10	Single cycle select. 0 = A, 1 = B.
11	Auto-increment/decrement protect enabled.
12	Defer protect enabled.
13	I/O protect enabled.
14	Write protect enabled.
15	Program map select. 0 = A, 1 = B.

READ VIOLATION DATA

DIB <f> ac,MAP



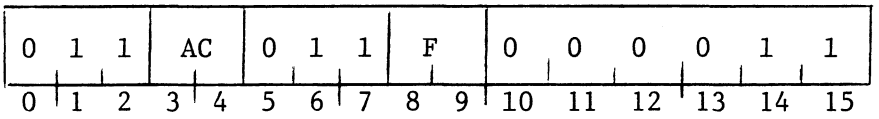
The violation status bits for the MMPU are placed in the specified AC, along with the logical page in which the violation occurred. The format of the data placed in the specified AC is as follows:



BITS	CONTENTS
0	Violation flag. Set to 1 if any of bits 10-14 = 1.
1-5	Logical page number. This is an octal number in the range 0-37.
6-8	Reserved for future use. Set to 0.
9	1 = single cycle was enabled during violation.
10	1 = validity violation.
11	1 = auto-increment/decrement violation.
12	1 = defer violation.
13	1 = I/O violation.
14	1 = write violation.
15	0 = violation occurred in map "A". 1 = violation occurred in map "B".

READ VIOLATION ADDRESS

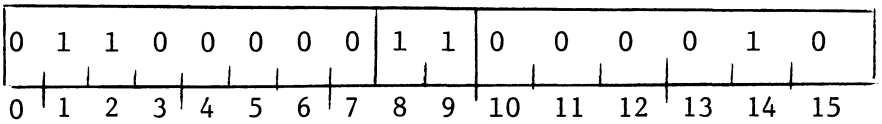
DIB <f> ac,MAP1



The logical address of the instruction that caused the violation is placed in the specified AC. After the instruction, bit 0 of the specified AC is cleared and bits 1-15 contain the address as an octal number in the range 0-77777.

MAP SINGLE CYCLE

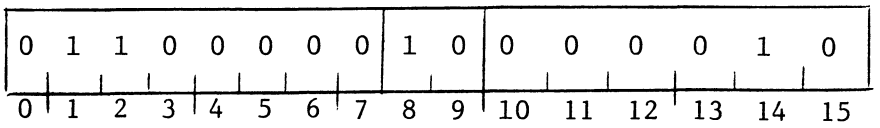
NIOP MAP



The second non-data channel memory reference after this instruction is issued is mapped with the user map indicated by the Single Cycle Select bit in the MMPU status word.

CLEAR VIOLATION

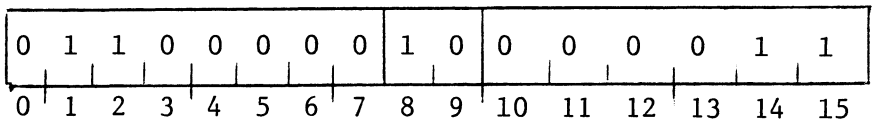
NIOC MAP



The violation status word for the MMPU is cleared. The Busy flag for device code 2, which indicates data channel errors, is also cleared.

CLEAR MAP

NIOC MAP1



The program and data channel maps are disabled, and all internal MMPU logic is initialized.

3.3 PARITY OPTION (DATA GENERAL PAR)

Description

The ZETACO Parity option, in conjunction with Parity Memory, provides a means of detecting Memory Data errors as they occur. This is accomplished by adding a seventeenth bit (Parity) to the Memory data bus. The Parity bit is the exclusive or of the 16 bit Memory data bus, and depending upon whether odd or even Parity is selected, is coded such that the total number of bits set to a "one" in each word is odd or even.

Each time a Memory location is written into, the Parity option computes the Parity bit and passes it to the Memory along with the data bits. Each time a Memory location is read, the Parity option computes the Parity and checks this Parity bit against the Parity bit read from Memory. If the bit Parity bits are identical, no error has occurrred and the 16 Memory data bits are passed along with no delay. If the two Parity bits are not identical an error has occurred and the Parity option either executes a system reset or initiates a program interrupt request. The address at which the Parity error occurred can then be read by using the DIA and DIB instructions.

The choice of whether the Parity option executes a system reset or initiates a program interrupt request, when a Memory Parity error occurs, is selectable by jumpers.

Condition	Jumpers
Program Interrupt	J1000 & J1001 installed J1002 not installed
System Reset	J1000 & J1001 not installed J1002 installed

Both Parity and Non-Parity Memory Boards can be intermixed in the same system. Parity Memory Boards provide a $\overline{\text{paren}}$ signal to the Parity option logic which indicates a Parity Memory is driving the Memory data

bus. For Non-Parity Memory Boards this signal is not present and Parity checking is disabled.

PROGRAMMING NOTES

Device code 4 and the mnemonic PAR are assigned to the parity option. The parity option has a Busy and Done flag, with the following meanings:

Done: 1 = Parity error
0 = No parity error

Busy: 1 = Even parity
0 = Odd parity

The flag control commands for the parity option are as follows:

f = S Set odd parity, enable parity interrupts.

f = C Clear parity error, disable parity interrupts.

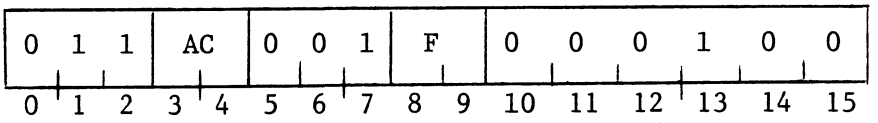
f = P Set even parity, enable parity interrupts.

In addition, an I/O RESET instruction will clear parity error, set odd parity, and disable parity interrupts. This is also the state that the machine is in when power is first turned on.

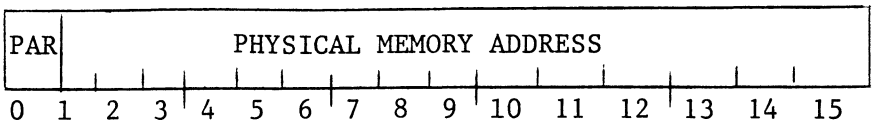
Two instructions are used to read the address which caused the parity error:

READ PARITY ERROR ADDRESS

DIA <f> ac,PAR



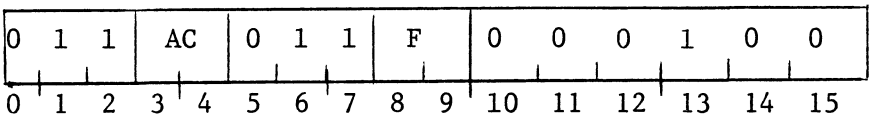
The parity bit associated with the most recent memory error is placed in bit 0 of the specified AC. The low-order 15 bits of the physical memory address where the parity error was found are placed in bits 1-15. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



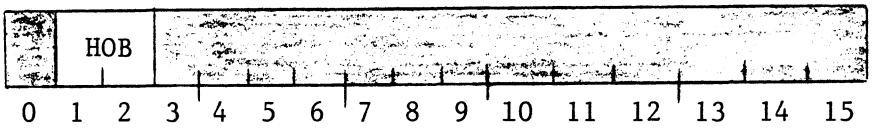
BITS	CONTENTS
0	Parity bit associated with memory error.
1-15	Physical memory address where parity error was found.

READ EXTENDED ERROR ADDRESS

DIB <f> ac,PAR



The two high-order bits of the physical memory address where the most recent parity error was found are placed in bits 1 and 2 of the specified AC. Bits 0 and 3-15 are set to 0. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



BITS	CONTENTS
0	Reserved for future use.
1-2	High-order bits of the physical memory address where the parity error was found.
3-15	Reserved for future use.

3.4 MULTIPLY/DIVIDE (DATA GENERALS MDV)

Description

The multiply/divide option provides hardware implementation of the software multiply/divide instructions. The multiply/divide option accomplishes this by being an extension to logic contained in the CPU.

A multiply or divide operation is enabled during the CPU fetch major state when the signal "fetch" is asserted and the memory data bus is 073301_8 (multiply) or 073101_8 (divide). The "fetch" signal sets the DP1 f/f within the CPU and the multiply/divide option asserts set $\overline{\text{DP1}}$ to maintain the set condition of the DP1 f/f until the multiply or divide operation is completed.

While the multiply or divide instruction is being executed the multiply/divide option directs the multiply or divide operation. If a data channel break is required during the execution of a multiply or divide instruction, the instruction is suspended to allow the CPU to execute DCH major states. At the completion of the data channel break, the state of the CPU, prior to the data channel break, is restored and the multiply/divide algorithm resumes. When the execution of the multiply or divide instruction is completed, the multiply/divide option relinquishes its control to the CPU, and ceases to assert $\overline{\text{Set DP1}}$.

Programming Notes

The multiply and divide operations follow fixed algorithms implemented with hardware subroutines instead of software subroutines.

4.0 Pre-Return Test

If a controller malfunction is expected, the use of the following Data General Diagnostics should be run and results recorded. These results should be sent with the returning controller board.

Option 732	Load and Run	N3MMU	25 passes
		N3MORT S	1/2 hour
		N3MORT L	1 hour
Option 734	Load and Run	N3MMU	25 passes
		N3MMPU	25 passes
		N3MORT S	1/2 hour
		N3MORT L	1 hour
Option 736	Load and Run	N3PRTY	25 passes
Option 738	Load and Run	MDV	25 passes

These protrams are available on Data General DTOS Diagnostic Tapes.

ZETA

Please give us your comments.

Please use this form to send us your comments regarding this Technical Manual. Your input is greatly appreciated! Problems will be promptly addressed and action taken as necessary. If you wish a written reply, please furnish your name and mailing address. Thank you.

Date_____

Name_____Title_____

Firm_____

Address_____

City/State/ZIP_____

TECHNICAL MANUAL TITLE_____

DOCUMENT NUMBER_____REVISION_____

ERRORS IN MANUAL:

SUGGESTIONS FOR IMPROVING EITHER THE MANUAL OR THE PRODUCT:

NO POSTAGE
STAMP
NECESSARY
IF MAILED IN
UNITED STATES

Business Reply Mail

FIRST CLASS PERMIT NO. 939 HOPKINS, MN 55343

POSTAGE WILL BE PAID BY ADDRESSEE

ZETA

6850 Shady Oak Road
Eden Prairie, MN 55344