Data General Corporation

Technical Manual

Nova 1200

015-000002-02

TECHNICAL MANUAL

FOR THE

NOVA 1200

VOLUME I

Printed in the United States of America

015-000002-02

LIST OF EFFECTIVE PAGES

Page No.	Issue	Page No.	Issue
Title	Original	6- 1 thru 6-37	Original
A	Original	6-38-6-42	Rev. 01
i thru iv	Original	6-43 thru 6-50	Original
1- 1 thru 1-17	Original	6-51 thru 6-60	Rev. 01
2- 1 thru 2-13	Original	6-61 thru 6-80	Original
3- 1 thru 3- 6	Original	7- 1 thru 7-34*	Original
4- 1 thru 4-41	Original	A- 1 thru A-41	Original
5- 1 thru 5-12	Original	B- 1 thru B-37	Original

* Section 7 contains 34 pages minimum. Additional drawings are added to this section by each selected option.

FOREWORD

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.

Users of this manual are cautioned that the specifications contained herein are subject to change by DGC at any time without prior notice. Users should refer to the latest revision before using this manual in connection with the operation of DGC equipment. DGC is not responsible for errors which may appear in the drawings and specifications in this manual.

Any comments concerning this publication should be forwarded to:

Publications Dept.
Data General Corp.
Route 9
Southboro, Mass. 01772

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION I

GENERAL DESCRIPTION

1-1 INTRODUCTION

This manual contains a detailed technical presentation of the installation, operation, and maintenance procedures for the Nova 1200 Computer. The Nova 1200 Computer, as described in this document, consists of the Nova 1200 central processor with one or more 4K core memory assemblies. The Basic I/ O Control assembly is also described in this manual. The Basic I/O Control can be configured to control three basic types of peripheral equipment, specifically the Teletype, the Paper Tape Reader, and the Paper Tape Punch. Both the memory and the Basic I/O Control with its various equipment configurations are options which may be purchased independent of the Nova 1200 central processor. This manual, however, addresses its presentation to the classical definition of a Computer and describes the central processor with 4K of core memory and a Teletype I/O facility. It should be noted that Technical Manuals for each optional peripheral device is produced by the original manufacturer and is shipped under separate cover with the corresponding equipment. This accompanying documentation should be thoroughly reviewed immediately after the peripheral device is received and prior to installation.

This manual is intended to complement Data General Reference Manual, "How to Use the Nova Computers". Operation and Programming information is provided in the Reference Manual and will not be repeated here except where necessary for expositional continuity. It is recommended that all potential users of this publication become familiar with the Reference Manual prior to reviewing this Technical Manual.

The detailed technical descriptions presented in this manual assumes the user of this document has a thorough knowledge of the operation of TTμL logic circuitry and the fundamentals of digital computer operations. The contents of this manual are divided into two individual volumes with each volume assembled and shipped under separate covers. Volume I contains Sections I through V, Appendix A, and Appendix B of this document. Volume II contains the Illustrated Parts List. These volumes are integrated into the all-inclusive Master Documentation for this equipment at the time of shipment. The input/output pin definitions of the various types of integrated circuit (IC) packages constituting the Nova 1200 and

Basic I/O Control logic are summarized in Appendix A of this manual. The illustrated parts list, presented in Section VI of this manual, can be used as a cross reference source which indexes the reference designator for the IC part (appearing on the logic diagram) with the corresponding manufacturer's part number. Once the IC manufacturer's part number has been retrieved from Section VI it can be used to reference the corresponding IC logic symbol and pin definitions listed in Appendix A.

1-2 GENERAL FUNCTIONAL DESCRIPTION

The Nova 1200 is a general purpose Computer with a 16-bit word length. The memory cycle time for the Nova 1200 is 1.2 microseconds. The Nova 1200 contains four hardware accumulators which are used for temporary data storage and data manipulation during the execution of all arithmetic and logic class (ALC) instructions. The four accumulators also perform double duty as part of the Input/Output system. Data exchanges between the interface logic of the external device and the Nova 1200 processor are performed under program control using the four accumulators.

The data paths within the Nova 1200 Processor are four bits wide and the processor is capable of performing several internal operations simultaneously. The Nova 1200 features a bi-directional Input/Output (I/O) bus which not only transmits data between the Central Processor Units (CPU) and the external I/O devices, but also allows external I/O devices to communicate directly with memory, bypassing the CPU. The speed of Data Channel operations between memory and external I/O high speed devices is greatly increased because of this direct exchange feature.

The Nova 1200 is also available in an enclosure with one central processor board assembly and 16 additional spare assembly slots, or ten more spare slots than the basic enclosure. This extra large enclosure, called the Nova 1200 "Jumbo", is available under option 8102. The other Nova 1200 enclosure configuration available is the "Table Top" cabinet (under option 8105). The technical data provided within this publication is applicable to all three Nova 1200 enclosure configurations.

Nova 1200 input/output instructions perform data transfers to and from peripheral equipment. The I/O instruction format allows 64 device code definitions, of which (octal) code 0 is not used and (octal) 77 is reserved for special functions. A 16-level programmed priority interrupt facilitates handling 16 different device speed classes within the interrupt control structure. Interrupts are enabled or disabled by a processor word, of which each

bit position exercises disabling control over (the interrupt logic) the devices assigned to that bit position. In terms of Interrupt timing, the time a device must wait depends on the number of devices capable of producing interrupts, the length of service routines for devices of higher priority, and whether the data channels are in use. Excluding the execution of indirect memory reference instructions the maximum interrupt waiting time is approximately 6 μsec. The instruction times for the Nova 1200 are listed in Table 1-1.

1-2.1 Computer Organization

The particulars covered in this paragraph are addressed primarily to the unique features of the Nova 1200 Architecture rather than to the fundamental operations characteristic of all general purpose computers. A block diagram of the Nova 1200 Computer is shown on Drawing 001-000107, bound into Section VII. Figure 1-1 is a duplication of this illustration and is included here for convenience purposes. The horizontal dash line appearing in the diagram separates the block components as to their physical relationships. The sections above the dash line are located on the CPU Printed Circuit Board (PCB) assembly. Conversely, the sections below the dash line are located on other PCB assemblies also housed in the Processor enclosure. The memory section consists of one or more memory PCB assemblies which are mounted within the Nova 1200 enclosure. The Console section is part of the Console/Enclosure Unit of the Nova 1200. The I/O Device Controller is a PCB assembly designed to interface any external I/O equipment properly which is operational compatible with the Nova 1200.

The Nova 1200 is organized around three data busses, each of which is 16 bits wide. The three busses are identified as the MEM (Memory) bus, the MBO (Memory Buffer Output) bus, and the I/O (Input/Output) bus. The primary function of the MEM bus is to carry data from the memories to the CPU. This data may be either instructions or operands. Only one memory is allowed to use the MEM bus at a time, that being the memory which is selected. The MEM bus is also used during programmed I/O input operations. Data from an I/O device Controller is loaded into the MB register of the selected memory, and transferred from the MB to the CPU via the MEM bus. The MEM bus also transmits data from the Console to the CPU. Console data may be generated either by the Console data switches, as during a READS, start, examine, etc., or by the Console functional switches. In the latter case, the Console function to be performed is encoded into bits placed on the MEM bus, from

Table 1-1. Nova 1200 Instruction Execution Times (in Microseconds)

Instruction	Execution Time*
LDA	2.55
STA	2.55
ISZ, DSZ	3.15*
JMP	1.35
JSR	1.35
Indirect addressing add	1.2
Base register addressing add	0
Autoindexing add	.6
COM, NEG, MOV, INC	1.35**
ADC, SUB, ADD, AND	1.35**
IO input (except INTA)	2.55
NIO	3.15
IO output	3.15
IO skips	2.55
INTA	2.55
MUL	3.75
DIV	
Successful	4.05
Unsuccessful	2.55
Interrupt	3.0
Latency	
With multiply-divide	8.0
Without multiply-divide	6.0
Data Channel	
Input	1.2
Output	1.2/1.8***
Increment	1.8/2.4***
Latency	
With multiply-divide	8.0
Without multiply-divide	6.0

* Times are for core; for read-only subtract .4 for LDA, STA, ISZ, DSZ if reference is to read-only memory.

** If skip occurs add 1.35 microseconds.

***When two numbers are given, the one at the left of the slash is the time for an isolated transfer, the one at the right is the minimum time between consecutive transfers.

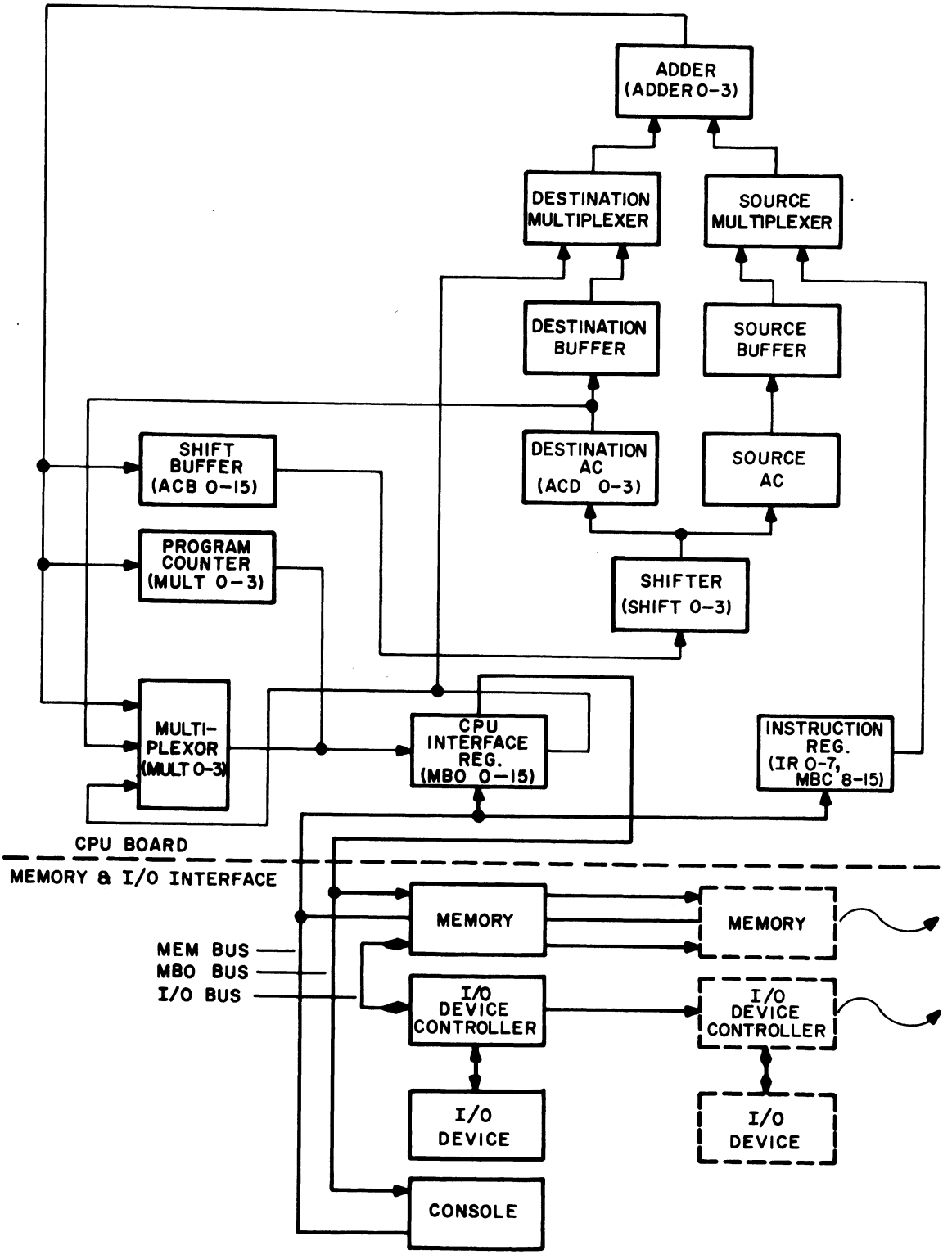


Figure 1-1. Block Diagram of Nova 1200

which it is loaded into the CPU instruction register and executed as a single instruction. As data is read from the Console onto the MEM bus the selected memory is prevented from also placing data on the bus by an $\overline{\text{INH TRANS}}$ signal generated by the CPU. The MEM bus is also used to generate a zero address function during the last cycle of a Program Load* sequence, and during the cycle preceding a PI (program interrupt cycle.)

The MBO bus is the data path for transmitting CPU data to either memory or I/O device Controllers. During a programmed I/O output operation, data is first transmitted to the MB (Memory Buffer) of the selected memory via the MBO bus. The information is then transferred from the MB out onto the I/O bus. Hence, the MBO is not physically connected directly to any I/O device Controller, but transmits output data through the MB. In the sequence of memory operations, the MBO bus transmits the address of the desired memory location to the selected memory MA register for loading. After the address data has been loaded into each MA register, the MBO bus will carry the data to be loaded into that address if the instruction requires a memory modification, as in a STA, or a console deposit operation.

The I/O bus differs from the other two bus paths in that it is a bi-directional path between memory and I/O device Controllers. The I/O bus communicates directly with memory and no logical connection is made to the CPU (although the bus is terminated at the CPU). The data flow for programmed I/O operations is the same as described previously, since the I/O bus receives data from the MB register of the selected memory. During Data Channel (DCH) operations data flow is between the I/O device Controller and memory, with the CPU completely bypassed.

1-3 PHYSICAL DESCRIPTION

The Nova 1200 Central Processor by definition consists of the Console/Enclosure Unit, Power Supply Unit, and the Central Processor printed circuit board assembly. An outline drawing of the printed circuit boards used in major Nova 1200 assemblies is provided in Appendix A of the "How to use the Nova Computers" reference manual. The Console/Enclosure Unit is so designed that seven 15 X 15 inch Printed Circuit Board (PCB) Assemblies may be plug mounted into a special printed circuit board connector in the Enclosure Chassis. The board assemblies are inserted horizontally into the Enclosure Chassis. A pair of guiding rails are built into the chassis frame (on each board level) to insure proper insertion of the board contacts into the corresponding socket of the multiple printed circuit board connector. The seven connector slots are numbered from the bottom of the chassis up to the

top with the first slot reserved for the CPU PCB assembly. The six remaining slots may be used for memory assemblies, I/O assemblies, or special control board assemblies. Figure 1-2 is a drawing showing the major components of the Nova 1200 Computer. The Power supply is mounted in the rear of the Console/Enclosure Unit and contains one fan mounted in the center of the supply for cooling. The Input/Output connector panel for the processor is mounted beside the power supply in a slightly recessed position. When optional peripheral equipment is purchased with the Nova 1200, the required I/O connector is mounted on Input/Output connector panel and wired into the multiple printed circuit board connector. (This wiring is direct from the pins of the selected PCB socket to the pins of the corresponding I/O connector.) It should be noted once a socket is wired for a specific Control PCB Assembly (used to control the optional peripheral device) that particular socket is dedicated to that purpose and must not be used for any other (different type) Control PCB Assembly. For example, a 4K Memory PCB Assembly can be mounted in any spare socket and will work properly. However, since the Device Controller PCB Assemblies are each hardwired from the selected socket position to the output I/O connector, each Controller assembly must be mounted in the (multiple printed circuit board) socket wired for it.

The wiring terminations for the major assembly components of the Nova 1200 are brought out to printed circuit edge type connectors. These connectors are mechanically aligned (on the abutting side of each major component to be assembled) so as to home with its connector mate on the adjacent assembly component. This interconnection facility replaces bulky interconnection cable runs and permits easy quick removal or replacement of any major assembly component. Figure 1-3 is a simplified diagram depicting the plugging sequence for assembling the Nova 1200. The plugging sequence for a complete processor assembly is performed as follows:

a) The Multiple PCB connector is hardware fastened to the Enclosure Chassis as the first step.
b) The plug receptacle on the end of the Console assembly is then mated with the edge connector on the end of the Multiple PCB Connector, and hardware secured to the Enclosure Chassis when properly positioned.
c) The edge connector of the Resistor Board subassembly is then plugged into the receptacle connector on the opposite end of the Multiple PCB connector and hardware secured to the Enclosure Chassis when properly positioned.
d) The edge connector of the power supply Unit is then plugged into the receptacle connector mounted on the Resistor Board subassembly and hardware secured to the Enclosure Chassis when properly positioned.

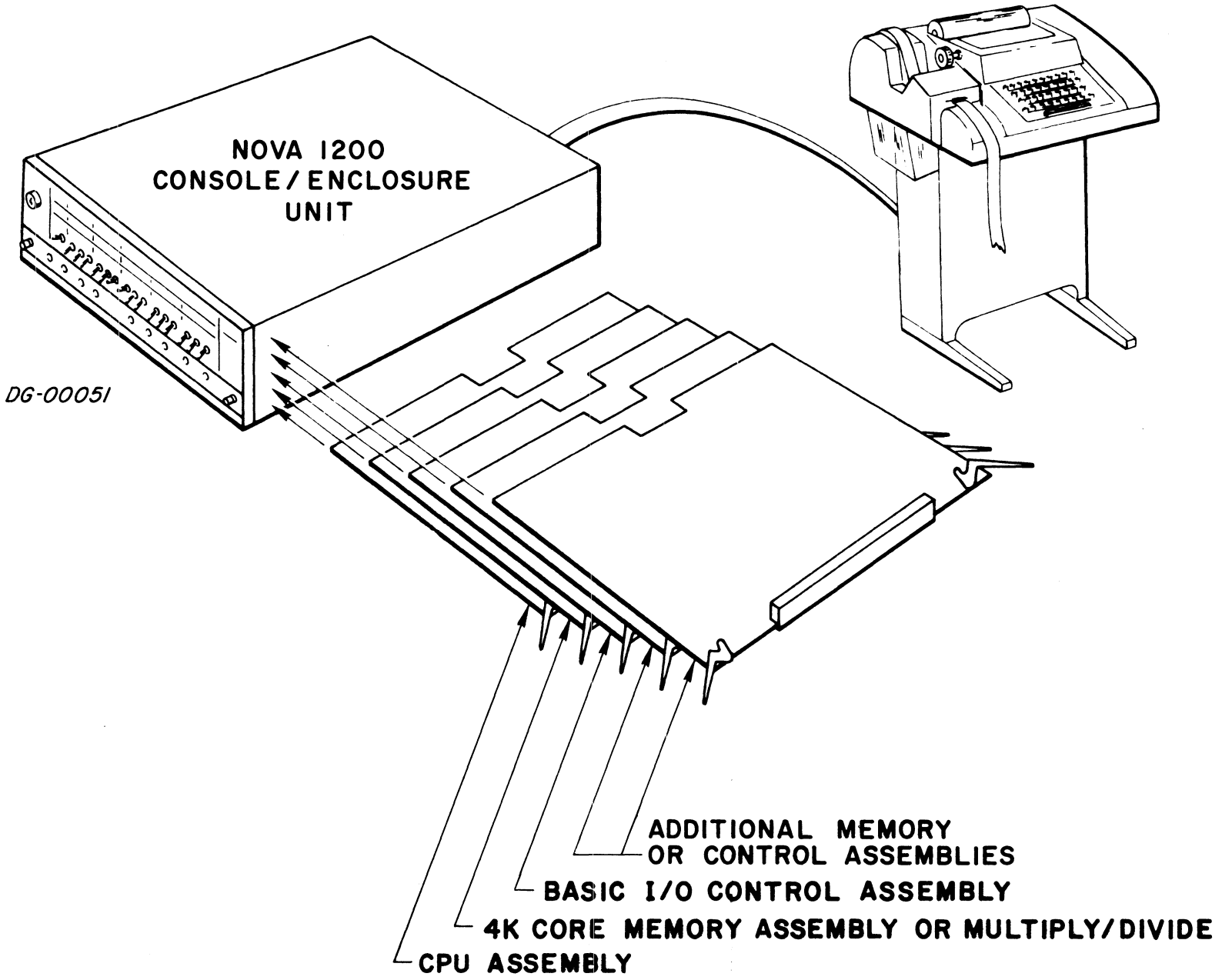


Figure 1-2. Major Components of the Nova 1200 Computer

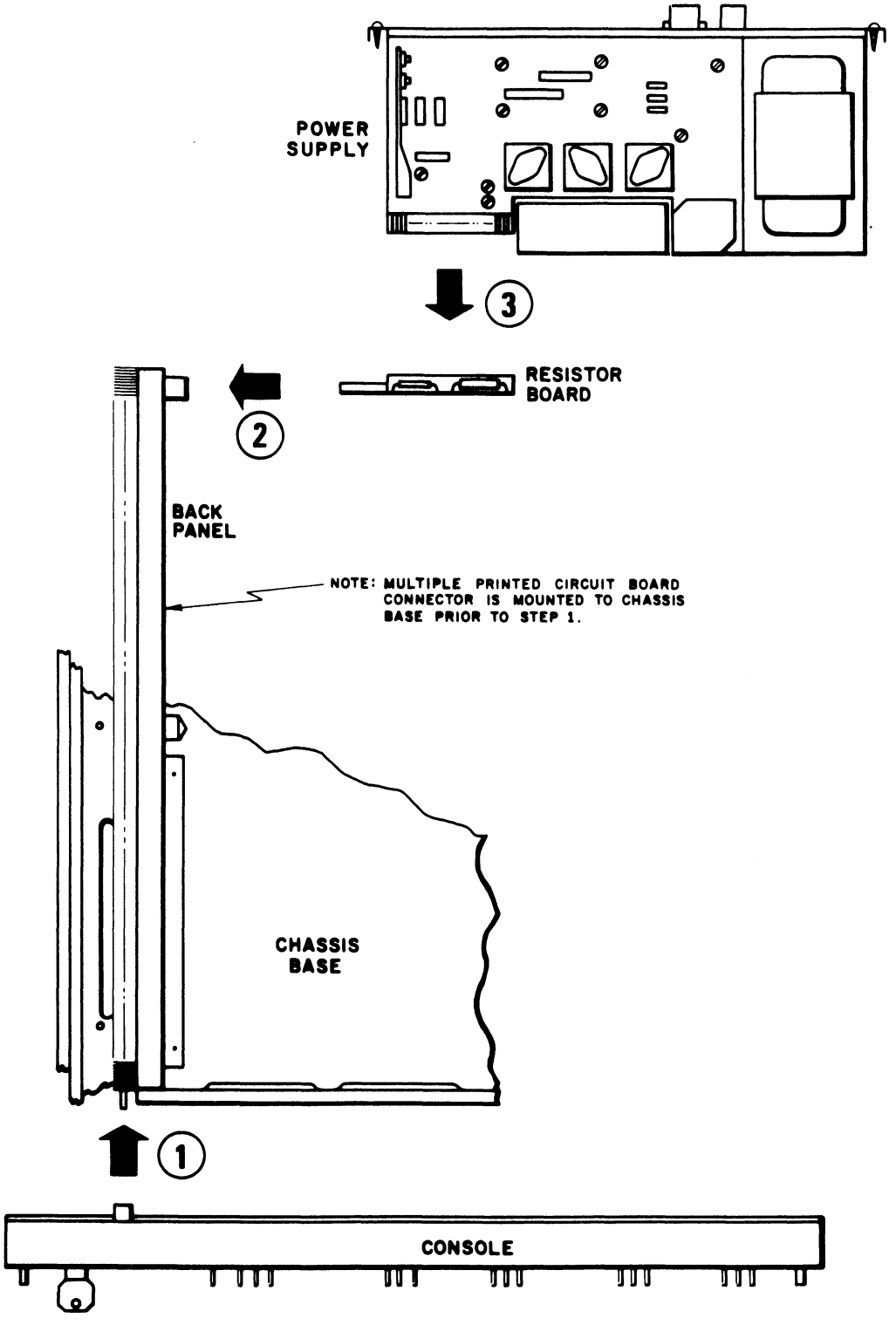


Figure 1-3. Diagram of Plugging Sequence for Assembling the Nova 1200

All that remains to make the processor operational is to plug the CPU, Memory, and I/O PCB assemblies into their respective slot locations within the Multiple PCB connector, and plug in the power cord. Table 1-2 is a summary of the Nova 1200 Physical Characteristics.

1-4 PERTINENT DOCUMENTS

The following documents serve as source material and complement the information in this manual.

How to Use the Nova Computers

Section 574-100-201 of Bulletin 272B
Volume 1, TECHNICAL MANUAL,
32 AND 33, TELETYPEWRITER SETS

Arithmetic Test
DGC Manual 097-000018

Nova 800/1200 Power Shut Down Test
DGC Manual 097-000022

Bootstrap Loader
DGC Manual 093-000002

Binary Loader
DGC Manual 093-000003

Nova 1200 Instruction Timer
DGC Manual 097-000019

Nova 1200 Teletype Test
DGC Manual 097-000021

Nova 1200 Logic Test
DGC Manual 097-000017

Memory Checkerboard III
DGC Manual 097-000014

Exerciser
DGC Manual 097-000004

1-5 ABBREVIATIONS

Listed below are the most commonly used abbreviations of registers, key operations, components, instructions, and signal names. Signal names not included in this list will be found in Appendix B Signal Origins. Appendix B contains an alphanumerical list of all signal names which appear on drawings, together with the drawing number which contains the generating circuits for the signal.

ABC0 thru ACB15	Accumulator Buffer Register Outputs 0 thru 15
ACD	Destination Accumulator
ACD OUT	Destination Accumulator Out
ACDP	Accumulator Deposit

Table 1-2. Nova 1200 Physical Characteristics

Physical Specifications

Enclosure Height:	5 1/4 inches
Enclosure Width:	19 inches
Enclosure Depth:	21 1/4 inches (23 with Console)
Weight:	50 lbs.
ASR33 Teletype Height (on stand):	45 inches
ASR33 Teletype Width:	22 inches
ASR33 Teletype Depth:	19 inches
ASR33 Teletype Weight:	56 lbs.

Electrical Specifications

Power Requirements:	115v or 230v* single phase, ±20% 47 to 63 Hz, Minimum Current** 1.5 amperes (175 watts) maximum Current 2.4 amperes (275 watts)
ASR33 Teletype:	115v single phase, 2 amperes. turn on surge 7 amperes. 92 watts.
Bus Signal	Low = 0 volts to + .4 volt nominal High = + 2.2 volts to 3.0 volts nominal
Processor Logic Signals	Low = 0 volts to + .4 volt nominal High = + 2.5 volts to 5.0 volts, + 3.5 volts nominal
Power Supply Voltages + 5 Volt Output:	+ 5 volts (nom.), + 4.7 volts low limit, + 5.45 high limit @ 25°C. Temperature variation: From + 5.1v @ 25°C to + 5.0v @ 55°C Typ.
- 5 Volt Output:	- 5 volts (nom.), -4.5 volts high limit, - 5.45 low limit @ 25°C. Temperature variation: From - 5.0V @ 25°C to - 5.1V @ 55°C Typ.

*230v on Special order

**Based on Minimum Computer Configuration of Processor, teletype interface, and 4K of memory.

Table 1-2. Nova 1200 Physical Characteristics (Continued)

Indicator Lamp Voltage ($+V_{Lamp}$)	+ 10.5 + 13.5 Volts Temperature variation: N/A
Memory Voltage (Formerly + VINH & + VMEM)	*See Memory Drive Specification.* Temperature variation: From + 15.1v setting @ 25°C to 14.5 (Max.) @ 55°C.
Power Supply Currents	
+ 5 Volt Output:	9.75 amps max. *
- 5 Volt Output:	1.0 amp max.
Memory Drive Nominal	+ 15.0
(+ VMEM) X and Y windings:	+ 15.0 Volts, 390 ma @ 25°C
(+ VINH) Inhibit windings:	+ 15.0 Volts, 740 ma @ 25°C

Functional

Memory Reference Cycle Time	
With Accumulator:	2.55 microseconds
Without Accumulator:	1.2 microseconds
Word Length:	16 bits
Core Memory Size:	4096 words, expandable to 32,768 in increments of 4096 words

* + 5 Volt current specification is based on requirements of a Nova 1200 with one 4K Memory only. Add 3/4 amp (as an approximation) for each additional 4K Memory installed.

ABBREVIATIONS. (Continued)

ACD 3 SEL	Destination Accumulator Select enable line
ACD 4 SEL	Destination Accumulator Select enable line
AC EX	Accumulator Examine
ACS	Source Accumulator
ACS 1 SEL	Source Accumulator Select enable line
ACS 2 SEL	Source Accumulator Select enable line
ACTG0, ACTG1	Accumulator Timing Generator outputs 0 & 1
ALC	Arithmetic Logic Class (instruction)
AND ENAB	AND (instruction) Enable
CLK	Clock
CLR	Clear
CLR ION	Clear Interrupt On
CON DATA	Console Data
CON INST	Console Instruction
CON RQ	Console Request
CONT	Continue switch at Console
CPU	Central Processor Unit
CPU CLK	Central Processor Unit Clock
CPU INST	Central Processor Unit Instruction
CRY ENAB	Carry Enable
CRY OUT	Carry Out
CRY SET	Carry Set
DATIA	Data In A (I/O instruction)
DATIB	Data In B (I/O instruction)
DATIC	Data In C (I/O instruction)
DATOA	Data Out A (I/O instruction)
DATOB	Data Out B (I/O instruction)
DATOC	Data Out C (I/O instruction)
DATA0 thru DATA15	I/O Data bus signals, 16 bits wide
D BUFFR	Destination (Accumulator) Buffer

ABBREVIATIONS. (Continued)

DCH	Data Channels
DCHA	Data Channel Acknowledge
DCH INC	Data Channels Increment
DCHI	Data Channel In
DCH LOOP ENAB	Data Channel Loop Enable
DCHM(0 or 1)	Data Channel Mode (0 or 1) Code type of Data Channel Cycle requested by Device
DCHO	Data Channel Out
DCHP IN	Data Channel Priority In
DCHP OUT	Data Channel Priority Out
DCHR	Data Channel Request
DEFER	Defer (instruction execution state)
DISABLE D MULT	Disable Destination Multiplexer
DIV	Divide (instruction)
DP	Deposit
DPN	Deposit Next
D MULT	Destination Multiplexer
D SET	Defer Set
DSZ	Decrement and Skip if Zero (instruction)
DS0-DS5	Device Select lines 0 thru 5
D+E SET	Defer or Execute Set
EFA	Effective Address
EX	Examine
EXN	Examine Next
E SET	Execute Set
INH GATE A	Inhibit Gate A (Memory)
INH GATE B	Inhibit Gate B (Memory)
INH TRANS	Inhibit Transmission
INH0-INH15	Inhibit Register outputs 0 thru 15 (Memory)

INTA	Interrupt Acknowledge
INTP IN	Interrupt Priority In (to Device)
INTP OUT	Interrupt Priority Out (from Device)
INTR	Interrupt (Bus Signal from Device)
IO (F+D)	IO (instruction) (Fetch or Defer state)
IO or I/O	Input/Output
ION	Interrupt On
IO PLS	Input/Output Pulse
IORST	Input/Output Reset
IO SKIP	Input/Output Skip (instruction)
IR0 thru IR7	Instruction Register outputs 0 thru 7
ISTP	Instruction Step (Console switch)
ISZ	Increment and Skip if Zero (instruction)
JMP	Jump (instruction)
JSR	Jump to Subroutine (instruction)
KEYM	Key Memory (access cycle)
LOAD AC	Load Accumulator
LOAD ACB	Load Accumulator Buffer (Shifter)
LOAD IR	Load Instruction Register
LOAD MBO	Load Memory Bus Outputs (CPU Interface Register)
LOAD PC	Load Program Counter
MA1 thru MA15	Memory Address Register outputs 1 thru 15
MA LOAD	Load Memory Address Register
MB CLEAR	Memory Buffer Clear
MBC8 thru MBC15	Memory Buffer Computer outputs 8 thru 15
MB LOAD	Load Memory Buffer Register
MBO0 thru MBO15	Memory Bus Outputs (CPU Interface Register) 0 thru 15
MD SEL1	Multiply Divide Select 1
MD1-MD15	Memory Data 1 thru 15

ABBREVIATIONS. (Continued)

MEM CLK	Memory Clock
MEM OK	Power Supply Output Memory Voltage at correct level.
MEM0 thru MEM15	Memory Bus lines 0 thru 15 (to CPU)
MSKO	Mask Out (instruction)
MSTP	Memory Step (Console switch)
MTG0 thru MTG3	Memory Timing Generator (signals) 0 thru 3
MULT0 thru MULT3	Multiplexer Output (signals) 0 thru 3
OVFLO	Signal to Device that memory location being incremented or added to (Via Data Channels) has Overflowed
PC	Program Counter
PC ENAB	Program Counter Enable
PC IN	Program Counter In
PEND	Pending, e.g., INT PEND
PI	Program Interrupt
PI SET	Program Interrupt Set
PL	Program Load
PTG5 ENAB	Processor Timing Generator 5 (pulse) Enable
PTG0 thru PTG5	Processor Timing Generator (signals) 0 thru 5
PULSE ENAB	Pulse Enable (PTG and TS3 function)
PWR FAIL	Power Fail
READ IO	Read IO (Device Controller)
RINH0 thru RINH15	(Collector) Resistor, Inhibit Driver
RQENB	Request Enable
RST	Restart (Console switch)
SARD	Selected Address
S BUFFER	Source Buffer
SELB	Selected Busy (Bus signal)
SELD	Selected Done (Bus signal)

ABBREVIATIONS. (Continued)

SET ION	Set Interrupt On
SHIFT ACB	Shift Accumulator Buffer
SHL	Shift Left
SHR	Shift Right
SKIP INC	Skip Increment
SL0 thru SL15	Sense Lines (Memory Stack) 0 thru 15
S MULT	Source Multiplexer
SNS0 thru SNS15	Sense Amplifier Outputs 0 thru 15
S0 thru S2	(Adder function) Select Control Bits 0 thru 2
STOP INH	(Processor) STOP INHIBIT
STRB A	Strobe A (Memory Stack)
STRB B	Strobe B (Memory Stack)
STRB C	Strobe C (Memory Stack)
STRB D	Strobe D (Memory Stack)
STRT	Start (Console switch)
SWP	Swap (bytes)
TS0 thru TS3	Time State 0 thru 3
TT	Teletype
TTI	Teletype In (Teletype Keyboard/Reader Buffer)
TTO	Teletype Out (Teletype Teleprinter/Punch (Buffer)
XRS	X (plane) Read Source (Memory Stack)
XWS	X (plane) Write Source (Memory Stack)
YRS	Y (plane) Read Source (Memory Stack)
YWS	Y (plane) Write Source (Memory Stack)
32 VNR	+ 32 Volts, Not Regulated
+ VINH	+ (Memory) Inhibit Voltage
+ V_{Lamp}	+ Lamp Voltage (Console indicators)
+ VMEM	+ Voltage Memory
+ 5 OK	+ 5 Volt (power) operating properly

SECTION II

INSTALLATION

2-1 GENERAL

This section provides detailed information and procedures for installing the basic Nova 1200 Computer. The Computer and Teletype are shipped in separate containers. Prior to performing any installation procedures inspect both shipping containers for any visible intransit damage such as would result from dropping or being punctured or crushed. Contact the carrier and Data General immediately if any damage is discovered, specifying the nature and extent of damage. Physical installation data and descriptions are provided in Appendix B of the "How to Use the Nova Computers" reference manual.

2-2 UNPACKING INSTRUCTIONS

The following two paragraphs describe the proper method of unpacking the Nova 1200 Computer and an ASR33 Teletype. The first paragraph describes the approved procedures for unpacking the Computer. The second paragraph describes the procedures for unpacking the Teletype. It is recommended that all shipping hardware, shims, packing and carton be saved and stored after unpacking in the event either machine is ever reshipped.

2-2.1 Unpacking the Nova 1200

After opening the Nova 1200 shipping container perform the steps of the following procedure:

a) Remove four corner pads.
b) Lift inner carton from box. (Requires two people.)
c) Cut top center seam.
d) Remove attached hardware (keys, mounting hardware, etc.).
e) Remove both U-shaped cardboard retaining frames.
f) Lift top layer of cardboard from box.
   Note: Top of Computer should now be exposed.
g) Remove cardboard shims located on sides of Computer.
h) Remove both restraining shims on the front of the Console.
i) Lift the unit from the box. (Requires two people.)
   Do not lift from the sides of the Computer. Hands should be placed on the rear and underside of the Power Supply and by the front of the Console.
   Check unit for shipping damage. Remove keys from plastic bag, insert the key, turn completely counterclockwise to the "Off" position.
j) Remove all packing material and general purpose frames from the system. Standard circuit boards should not be removed.

k) The computer is ready to apply power. It is suggested that the operator read the procedures listed under the Nova 1200 Start-up and Checkout paragraph of this Section before applying power to the machine.

2-2.2 Unpacking the ASR33 Teletype

The complete ASR33 is packaged in one carton. After opening the ASR33 shipping container perform the steps of the following procedure:

a) Remove Styrofoam pads (2). (See Figure 2-1.)
b) Remove corner braces (4) and Teletype Stand. (See Figure 2-2.) Manuals will be packed inside Teletype Stand.
c) Locate teletype bulletin 273B Vol. 1 in the manual set. Refer to page 1 of the section 574-100-201TC and read unpacking instructions.
d) Remove cardboard insets, accessory kit, and Typing Unit. (See Figure 2-3.) Typing Unit is mounted on a past board shipping pallet by seven screws.

NOTE: DO NOT USE OR ATTEMPT TO OPERATE TYPING UNIT BEFORE REMOVING THE (3) HEX HEAD BOLTS FROM THE BOTTOM OF THE SHIPPING PALLET.

e) Remove three pieces of adhesive nylon tape; two pieces are securing the paper supply and lid, the other piece is securing the paper tape supply, punch and reader.
f) Remove the Typing Unit Cover to expose the carriage. The carriage is tied to the chassis with a pipe cleaner. This securing wire must be removed before operating the Teletype.
g) Some Teletypes are equipped with a yellow spacer spring holding the reader fingers stationary. This must be removed prior to operating the Teletype.

2-3 NOVA 1200 START-UP AND CHECKOUT

The procedures listed below describe the proper methods for initial turn-on and subsequent checkout of the Nova 1200 Computer. These procedures should be performed immediately after the Nova 1200 has been unpacked. The procedures are listed below in the exact order of performance, and must be performed in the order of appearance, during initial turn-on.

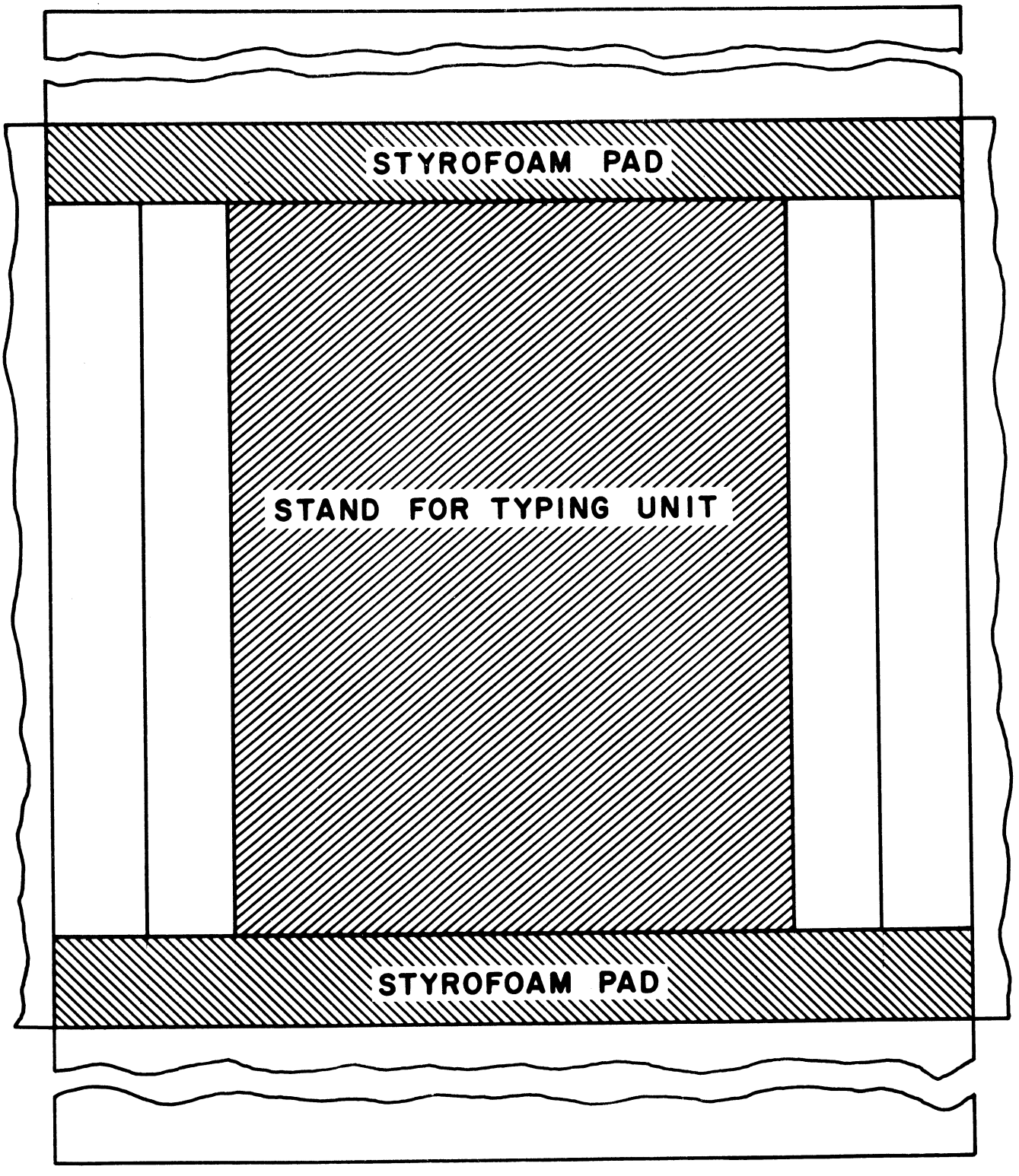


Figure 2-1. Location of Styrofoam Pads

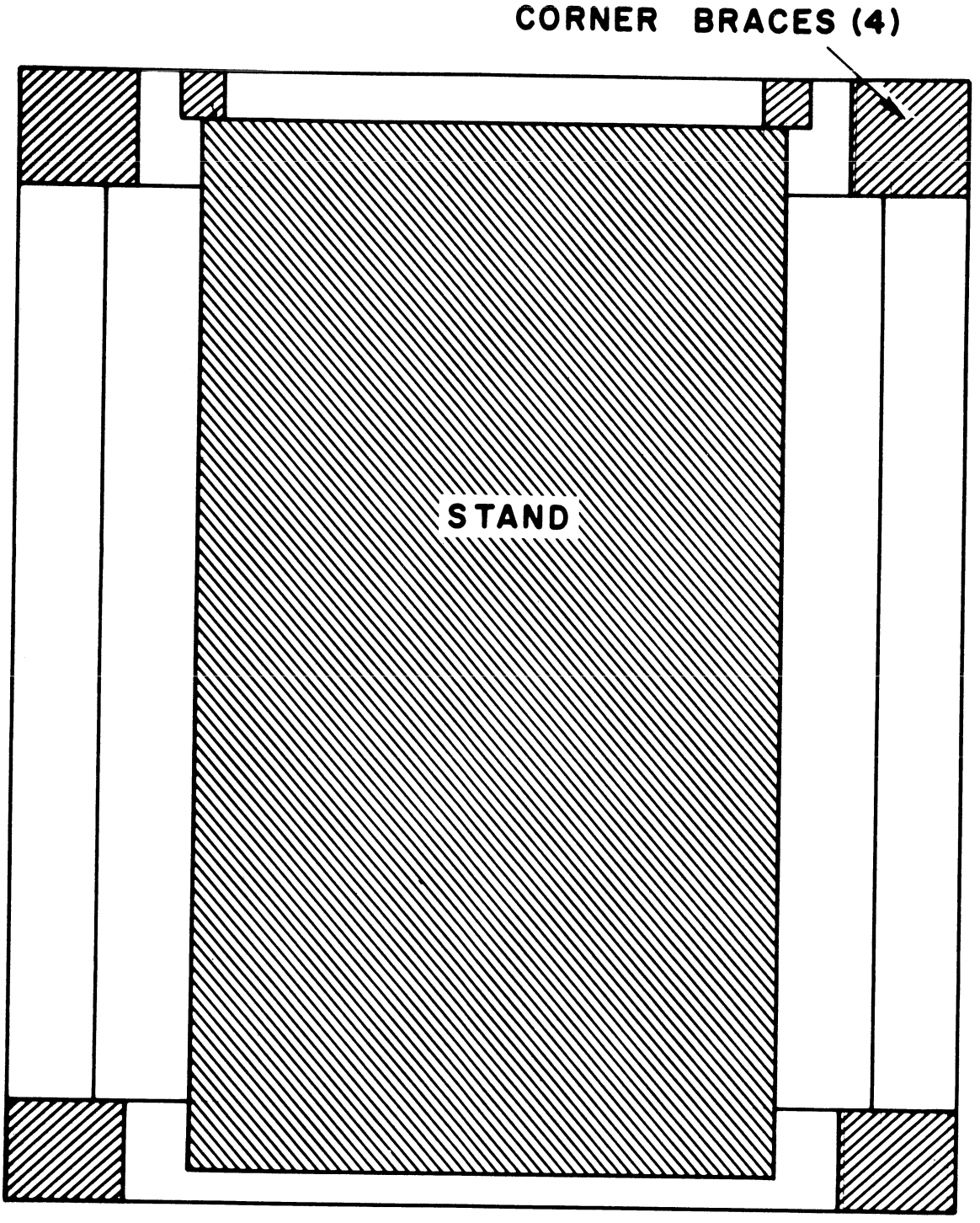


Figure 2-2. Location of Corner Braces in Teletype Carton

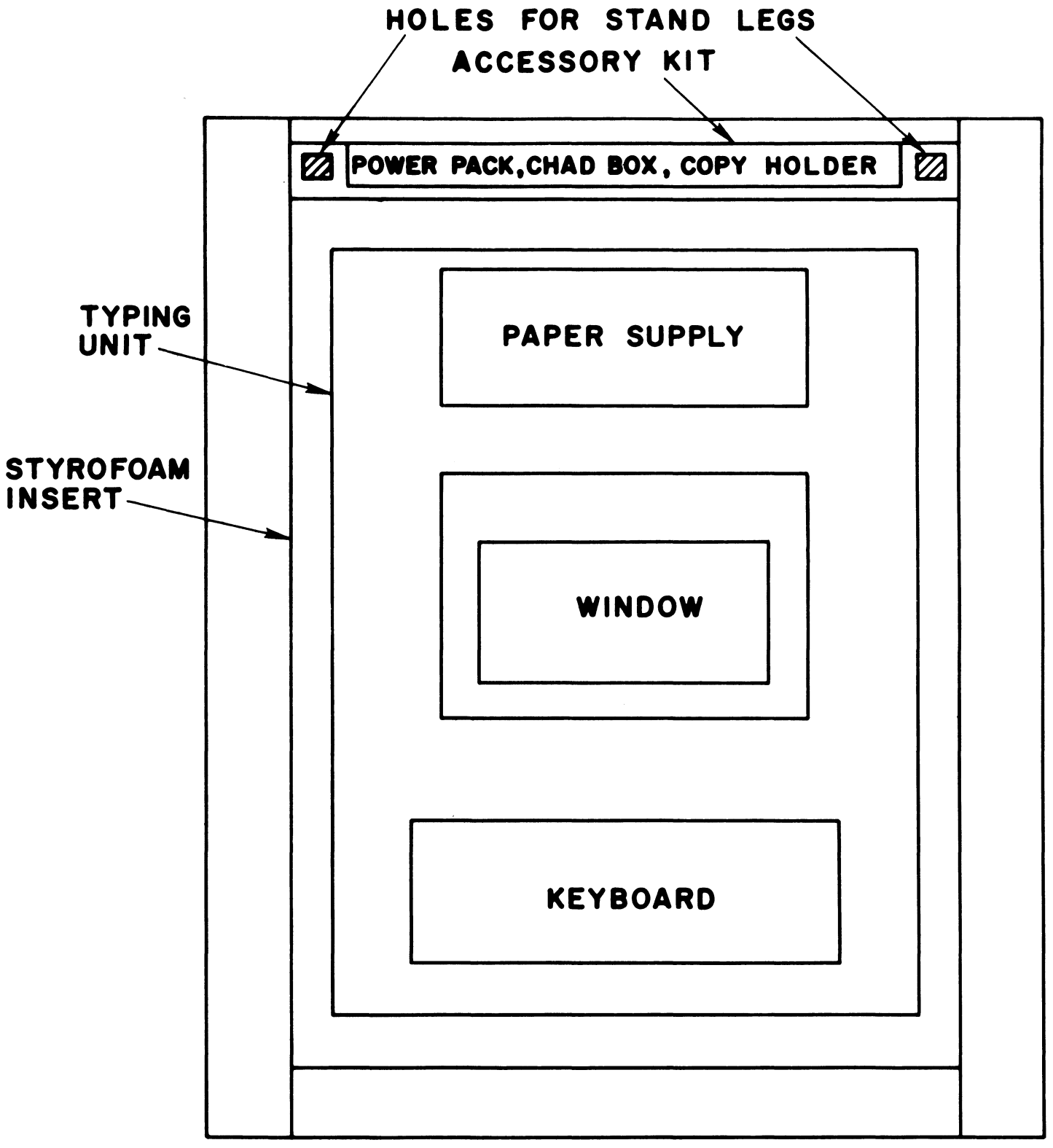


Figure 2-3. Location of Teletype Typing Unit in Carton

2-3.1 Start-Up Procedure

The general purpose Exerciser (Program Listing and Tape #095-000012, Manual #097-000004) is always the last program run in each Computer prior to shipping. The Exerciser diagnostic program checks the entire instruction repertoire and all memory locations associated with that particular processor system. This program should still be intact within the memory and available for execution. To start this program, only Console data switch 14 on the operator's console should be raised. Turn power "ON" by setting the key to the vertical position. Raise the Reset/Stop switch momentarily to the Reset position and then raise the Start switch momentarily to the Start position. Program should now be in execution with both the Fetch and Run indicators on. With typical operation of the program, an incrementing sequence from $1,000_8$ to approximately $3,400_8$ will be observed in the Data Register display. The cycle takes approximately one to two minutes for a 4K system and becomes significantly longer depending on a maximum core size of the system. Any halt of the Computer and improper indications constitutes an error. Should you encounter any difficulty with the start-up procedure, please contact the Data General representative in your area, or our Field Service Department at the Southboro factory (Area Code: 617-485-9100).

2-3.2 Check-Out Procedures

The Nova 1200 check-out sequence consists of static and dynamic tests. Static tests are performed manually at the Operator's Console. Dynamic tests, on the other hand, are a series of tests performed under program control, and either terminate successfully or halt at some specific location to indicate detection of some failure by the diagnostic. In the normal sequence of testing, the static tests are performed first to verify all of the manual controls are working properly. Once the Console is verified as operational, the dynamic testing may be performed. Successful completion of all the recommended diagnostic tests should be considered verification of the first check-out step. Complete verification is obtained when all of the diagnostic program tapes supplied with the documentation package have been run successfully.

2-3.2.1 Static Tests

a) Turn power on.
b) Verify console indicators operate properly.
c) Deposit and examine all zeros in Memory location 0.
d) Deposit and examine all ones in Memory location 0.
e) Deposit and examine all zeros in Accumulators ACC 0-3.
f) Deposit and examine all ones in Accumulators ACC 0-3.
g) Deposit 000017 in ACC 0.
h) Deposit 000360 in ACC 1.
i) Deposit 007400 in ACC 2.
j) Deposit 170000 in ACC 3.
k) Examine all accumulators and verify no data changed.
l) Deposit all zeros in Memory location 0.
m) Momentarily raise the Start/Continue switch to the Start position. Verify run indicator is on.
n) Lock computer. Verify reset/stop toggle functional inoperative.
o) Unlock and stop computer.
p) Continually depress DEPOSIT NEXT. Verify PC increments.
q) Continually depress EXAMINE NEXT. Verify PC increments.
r) Place all zeros in Memory location 0.

2-3.2.2 Dynamic Tests. At this point and prior to performing the first dynamic test it is necessary to connect the Teletype to the Nova 1200. These procedures assume all of the unpacking steps for the Teletype listed in paragraph 2-2.2 have been completed.

2-3.2.2.1 Teletype to Computer Connection Procedure

a) Turn off computer.
b) Plug in TTY to 115v outlet in the rear of the Processor Enclosure.
c) Plug in the 9 pin connector to the receptacle indicated by the connector layout diagram attached to the rear of the Nova 1200 Enclosure. The proper connector is labeled 4010 (Data General Model number for TTY).
d) Turn the line/local switch on the lower-right-front-panel of TTY to local.
e) Place roll of tape in punch and turn on punch.
f) Type all characters on keyboard. Note correct typing and also that punch is operating. It is not necessary to verify tape produced. That will be done in the sequence of tests that follow.
g) Turn punch off and return line/local switch to line. The unit is now ready for use by the computer.

The next step in the dynamic test sequence is to place the Binary Loader program into core. This may be done either manually or automatically. Automatic loading requires that the Program Load Option be included with the processor. This option operates in conjunction with the Nova 800/1200 Program Load tape supplied as part of the Program Load Option. The Program Load Read-Only hardware loads the Bootstrap program into memory location 0-37.

The Bootstrap program is executed immediately and reads a Prologue (to the Binary Loader) into locations 40 to 120. The Prologue is then executed immediately after loading, and thereby loads the Binary Loader from the last section of the Program Load Tape. Paragraph 2-3.2.2.2 describes the loading procedures for those Processors with the Automatic Program Load option. Paragraph 2-3.2.2.2 describes the manual loading procedures for Processors without the Automatic Program Load option.

2-3.2.2.2 Loading Procedures for Program Load Option

a) Turn on Computer. Verify Teletype is on-line.
b) Set Teletype reader switch to FREE.
c) Thread the Nova 800/1200 Program Load tape into Teletype reader, and set reader switch to START.
d) Set the TTI device code 010_8 (or 012_8 for High Speed Paper Tape Reader) into the six rightmost Console switches (bits 10-15).
e) Raise the Program Load switch on the Console to the Program Load position.
f) Tape should move through the Teletype reader. When the tape halts verify Console Address register displays 00120_8 as the halt location.
g) Set Teletype reader switch to FREE, and remove the Nova 800/1200 Program Load tape from Teletype reader.
h) Thread the test program tape Checkerboard III (Binary tape #095-000031, Manual #097-000014) into the Teletype reader, and set reader switch to START.
i) Press the Start/Continue switch on the Console to the Continue position.
j) Verify tape moves through the Teletype reader. When tape halts Checkerboard III will be loaded and ready for execution.
k) It should be noted that for subsequent loading of other programs after a program other than the Binary Loader has been executed, first thread the program tape in the Teletype reader, then load X7777 into the console switches and raise the Start/Continue switch to Start. This will rerun the Binary Loader and bring the new program into core.

2-3.2.2.3 Loading Procedures Without Program Load Option

a) Using the Console switches, key in the Bootstrap Loader Manual #093-000002) instructions listed below.

BOOTSTRAP LOADER

```
TTI:    ** = 10          PTR:    ** = 12
X7757      126440        GET:       SUBO    1,1
X7760      0636**                   SKPDN     X
X7761      000777                   JMP     .-1
X7762      0605**                   DIAS    0,X
X7763      127100                   ADDL    1,1
X7764      127100                   ADDL    1,1
X7765      107003                   ADD     0,1,SNC
```

```
TTI:
X7766   000772            JMP     GET+1
X7767   001400            JMP     0,3
X7770   0601**    BSTRP:  NIOS      X
X7771   004766            JSR     GET
X7772   044402            STA     1,.+2
X7773   004764            JSR     GET
                          ...
                          ...
```

b) Set the Teletype reader switch to FREE and thread the Binary Loader (Special Format #091-000004, Manual #093-000003) into the Teletype reader (or the High Speed Paper Tape Reader).
c) Verify Teletype is on-line and set the Teletype reader switch to START.
d) Set the Console data switches to X7777.
e) Momentarily raise the Start/Continue switch to the Start position
f) Verify that the tape moves through the Teletype reader. When the tape halts verify Console address register displays X7775.
g) Set Teletype reader switch to FREE, and remove the Binary Loader tape.
h) Thread the test program tape Checkerboard III (Binary tape #095-000031, Manual #097-000014) into the Teletype reader, and set reader switch to START.
i) Set the Console data switches to X7777.
j) Momentarily raise the Start/Continue switch to the Start position.
k) Verify tape moves through the Teletype reader. When the tape halts verify Console address register displays X7743.

2-3.2.2.4 Memory Test. After the memory test program, Checkerboard III has been loaded as per paragraph 2-3.2.2.2, or paragraph 2-3.2.2.3, perform the following steps to run the test program:

a) Place 000002_8 in Console switches.
b) Momentarily raise the Start/Continue switch to the Start position.
c) Verify program cycle.
d) Raise Console switch 0 = 1 to include worst case. Verify program cycle.
e) Allow program to cycle 15 minutes.
f) Reset the computer.

2-3.2.2.5 Logic Test. Load the test program (Binary #095-000036) as per paragraph 2-3.2.2.2, or paragraph 2-3.2.2.3.

a) Set the Console switches to 000400_8.
b) Momentarily raise the Start/Continue switch to the Start position.
c) Verify computer halts at location 000403_8. Press the Start/Continue switch to the Continue position.
d) Program takes less than a second for one complete pass. Allow the program to run for several minutes, then stop the computer.
e) Note the stop location of the computer, and perform a series of single instructions by toggling the Instruction Step switch. Verify PC follows the program (as listed in program documentation).
f) Toggle the Memory Step switch as in step e. Verify PC follows the program (as listed in program documentation).

2-3.3.6 Teletype Test. Load the test program (Binary #095-000041) as per paragraph 2-3.2.2.2, or paragraph 2-3.2.2.3.

a) Turn on Punch of TTY.
b) With the TTY in local, depress "Here in" to generate leader.
c) Return on-line and place tape in TTY Reader.
d) Place reader in start position.
e) Place 000040_8 in Console switches. Raise the Start/Continue switch to the Start position.
f) Program will cycle and type "PASS" on the end pass.
g) Allow program to cycle for 5 passes.
h) Reset the computer.
i) Place reader of TTY in "FREE" position.
j) Place 000045_8 in Console switches. Raise the Start/Continue switch to the Start position.
k) After teletype starts punching data, place leader of tape in TTY and push START on TTY reader.
l) Program should cycle for minimum 1 minute.

2-3.2.2.7 Instruction Timer. Load the test program (Binary #095-000038) as per paragraph 2-3.2.2.2, or paragraph 2-3.2.2.3.

a) Set Console switches to 000003_8.
b) Momentarily raise the Start/Continue switch to the Start position.
c) Allow the program to run for several minutes, and verify no teletype type-outs occur. Program has a built-in tolerance of ±20 nanoseconds for the execution time of each instruction tested and will print out the time for any instruction exceeding this limit.

d) Depress Console Reset/Stop switch to the Stop position.
e) Set 000002_8 into the Console switches.
f) Momentarily raise the Console Start/Continue switch to the Start position.
g) Starting at location 000002_8 cause the program to type out the execution time for each instruction in nanoseconds.
h) If excessive execution time was detected during step c, contact the Data General representative in your area or our Field Service Department.
i) If step c was completed successfully save the listing of instruction execution times generated during step g. This listing should be filed as part of the maintenance record for the Nova 1200. Typical instruction execution times (within ±20 ns) for the Nova 1200 are listed below.

INSTRUCTION EXECUTION TIMES

MOV 0,0	1350
ADD 0,0	1350
AND 0,0	1350
LDA 0,0	2550
STA 0,0	2550
ISZ 0	3150
DSZ 0	3150
JMP .+1	1350
JSR .+1	1350
LDA 0,@0	3750
LDA 0,@21	4345
LDA 0,@31	4345
LDA 0,@(@0)	4945
DIA 0,0	2550
DOA 0,0	3150
INTA 0	2550
SKPBN 0	2550

2-3.2.2.8 Arithmetic Test. Load the test program (Binary #095-000037) as per paragraph 2-3.2.2.2, or paragraph 2-3.2.2.3.

a) Set Console switches to 000002_8.
b) Momentarily raise the Start/Continue switch to the Start position.
c) Upon starting program should issue a message stating "Last Location in Memory is XXXXX".
d) Verify that the value printed corresponds to the actual System Memory size. If they do not agree an error has occurred.
e) The program will cycle continuously, and type out the word "PASS" on each program iteration.

2-3.2.2.9 Power Shutdown Test (No Power Monitor). Load the test program (Binary #095-000044) as per paragraph 2-3.2.2.2, or paragraph 2-3.2.2.3.

a) If Computer has Power Monitor Option perform paragraph 2-3.2.2.10 instead of these procedures.
b) Set Console switches to 000002_8.
c) Momentarily raise the Start/Continue switch to the Start position.
d) The program will request the operator to turn the computer off, on and to restart it.
e) Upon restart the program will ring the teletype bell three (3) times. The Operator should repeatedly perform the power off-restart sequence. After each restart allow 2-3 seconds for a possible error message.

2-3.2.2.10 Power Shutdown Test (with Power Monitor). Load the test program (Binary #095-000044) as per paragraph 2-3.2.2.2, or paragraph 2-3.2.2.3.

a) Set Console switches to 000002_8.
b) Momentarily raise the Start/Continue switch to the Start position.
c) The program will request the operator to turn the computer off, on, and to restart it.
d) After several restarts lock the Console and remove the AC line plug.
e) When computer power is restored the program will be restarted without operator intervention. The teletype bell will be rung three (3) times each time power is restored.
f) The operator should repeatedly remove and restore power. At each restoration of power allow 2-3 seconds for possible error messages.

2-3.2.2.11 Exerciser. Load test program (Binary #095-000012) as per paragraph 2-3.2.2.2, or paragraph 2-3.2.2.3.

a) Set Console switches to 000002_8.
b) Momentarily raise the Console Start/Continue switch to the Start position.
c) Computer will cycle. Any Halt constitutes error. After one pass raise Console switches 2 and 3.
d) Turn on TTY punch. Set reader switch to FREE.
e) After TTY starts punching data, place leader in TTY read station.
f) Push START on TTY.
g) Allow computer to cycle for five minutes.
h) Lower Console switches 2 and 3 to terminate teletype test.

This test completes the start-up checkout for the Nova 1200 Computer.

2-3.3 Repacking

In order to properly repack the Nova 1200 or the Teletype, reverse the procedures listed in paragraph 2-2. Only strict adherence to the particulars described in each step will prevent serious damage to each machine during shipment. All retaining hardware and

packing should be replaced into the original positions within the carton before the units are shipped. The following special packing considerations must be observed for the Teletype:

a) Make sure (3) hex head screws are replaced in the original position underneath the Shipping Pallet.
b) The Console front switches and keyboard must be protected with some form of resilient packing or extensive damage will occur during shipment.

SECTION III

OPERATION

3-1 GENERAL

This section identifies and describes all of the manual controls and indicators used to operate the Nova 1200 Computer. A description of the controls and indicators is also provided in paragraph 2.7 of the "How to Use the Nova Computers" reference manual. Figure 3-1 is a drawing of the Nova 1200 operator's Console showing the controls and indicators referenced throughout this section.

3-2 CONSOLE CONTROLS

Used in conjunction with a teletypewriter and peripheral devices, the control console contains all controls necessary to operate the Nova 1200 Computer system. Each console control is described briefly in the following paragraphs.

3-2.1 Power Switch

The key-operated power switch controls the ac (primary power) input to the Nova 1200 Power Supply. In the OFF position, the ac input line is removed from the power supply. In the ON position, there is ac power to the power supply and the Computer is operational. In the LOCK position there is ac power to the power supply and the Computer is operational. However, all Console Control switches are disabled except for the power switch itself. The LOCK position allows a program to run without interference from occasional or accidental "switch diddling" or any other unscheduled attempts to operate the Computer. However, the Console Data switches remain operational to allow the operator to supply information to the program (when requested by the program). It should also be noted that all of the Console indicators remain operational when the power switch is in the LOCK position. The Console key can only be removed when the power switch is in the LOCK position.

3-2.2 AC0, AC1, AC2 and AC3 Deposit/Examine Switches

These four switches are used both for depositing data into the corresponding Accumulator, and examing their contents. The DEPOSIT position of any switch operates in conjunction with the relative positions of the (16) Console Data switches. Placing any one of the four AC switches in the DEPOSIT position will load the configuration of the Console Data

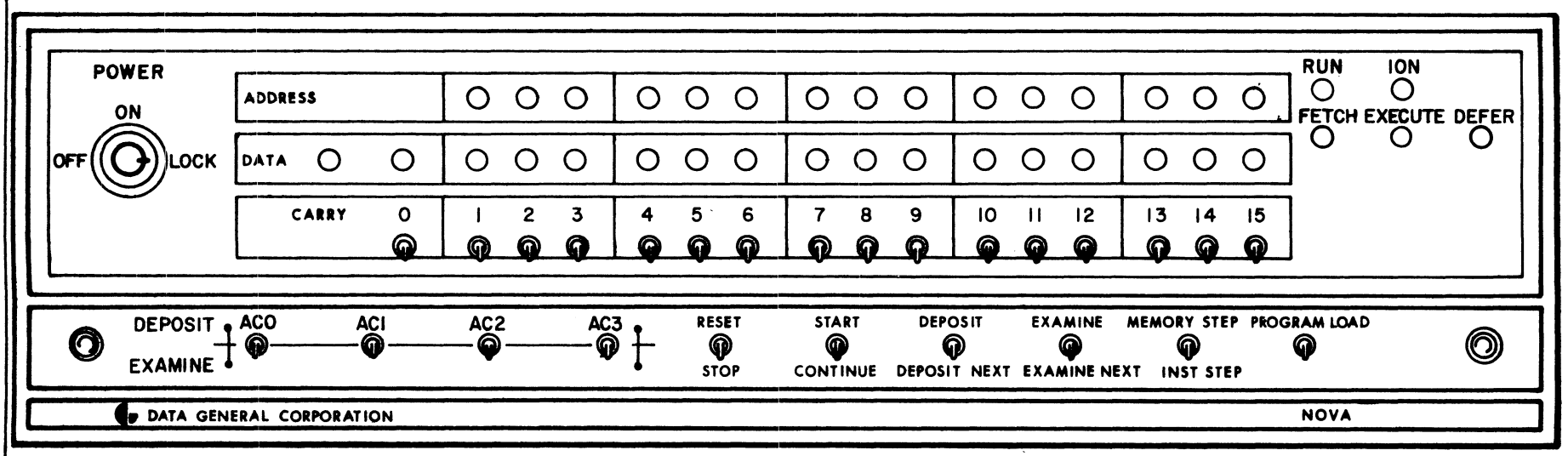


FIGURE 3-1 NOVA 1200 OPERATOR'S CONSOLE

switches into the specified Accumulator. Placing any one of the four AC switches in the EXAMINE position will display the contents of the specified Accumulator in the Console DATA lights.

3-2.3 Reset/Stop Switch

Placing the RESET/STOP switch in the RESET position causes the Nova 1200 to stop at the end of the current processor cycle. RESET also: clears flags in all I/O devices, clears Interrupt On, places the processor in supervisor mode, and sets the clock to line frequency. It should be noted that if the RESET/STOP switch is momentarily raised immediately after an AC deposit the data will not be allowed to reach the selected AC even though this data appears in the Console Data display. Pressing the RESET/STOP switch to the STOP position causes the Nova 1200 to stop before fetching the next instruction. The address indicators point to the next instruction. If the current instruction contains an infinitely long indirect addressing chain or there are continuous data channel requests, pressing STOP will not stop the Computer. Under these conditions it is necessary to momentarily raise the switch to the RESET position rather than pressing it to the STOP position.

3-2.4 Start/Continue Switch

Momentarily raising the START/CONTINUE switch to the START position causes the Nova 1200 to load the address contained in the Console Data switches into PC, light the FETCH and RUN indicators, and begin normal operation by executing the instruction at the location specified by the PC. Pressing the START/CONTINUE switch to the CONTINUE position causes the Nova 1200 to turn on the RUN indicator and begin normal operation in the state indicated by the (five) indicators on the right-hand side of the Console (i.e., RUN, ION, FETCH, etc.) It should be noted that instruction stepping can be performed by momentarily raising the START/CONTINUE switch to the position while pressing the RESET/STOP switch to the STOP position.

3-2.5 Deposit/Deposit Next

Momentarily raising the switch to the DEPOSIT position will deposit the contents of the Console Data switches into the memory location specified by the address lights. Upon completion of the deposit the Console Data lights will display the word deposited. Pressing the switch to the DEPOSIT NEXT position will add 1 to the PC address displayed in the address lights and deposit the contents of the Console Data switches into the memory location specified by the incremented address. Upon completion of the deposit next the Console Data lights will display the word deposited. This switch is generally used in conjunction with the EXAMINE/EXAMINE NEXT switch. See paragraph below for an example switching sequence utilizing both switches.

3-2.6 Examine/Examine Next Switch

Momentarily raising the switch to the EXAMINE position will load the address contained in the Console Data switches into PC (which is displayed in the address lights) and display the contents of the addressed location in the Console Data lights. Pressing the switch to the EXAMINE NEXT position will add 1 to the PC address displayed in the address lights and display the contents of the location specified by the incremented address in the Console Data lights. The DEPOSIT/DEPOSIT NEXT, EXAMINE/EXAMINE NEXT switches can be used for a sequence of operations on consecutive memory locations. The sequence must begin with EXAMINE to supply the initial address unless PC already points to the right location. Suppose the Console Data switches are set to octal 100 initially. Then the following sequence of switch settings produces the effects listed.

EXAMINE	Display location 100.
EXAMINE NEXT	Display location 101.
EXAMINE NEXT	Display location 102.
DEPOSIT	Load Data switches into 102.
EXAMINE NEXT	Display location 103.
DEPOSIT	Load Data switches into 103.
DEPOSIT NEXT	Load Data switches into 104.
EXAMINE NEXT	Display location 105.

It should be noted that the EXAMINE position can be used to load the PC for beginning any single step procedure.

3-2.7 Memory Step/Inst Step Switch

Momentarily raising the switch to the MEMORY STEP position will perform a single processor

cycle in the state indicated by the Operational indicators and then stop. Upon completion the Operational indicators will point to the next Operational state to be executed. The address lights will display the contents of the PC, the data lights will display the data fetched from the last memory location accessed. Using the AC switches between memory steps within an instruction usually destroys information (in the Accumulator) necessary for the execution of the rest of the instruction. To use the various examine and deposit switches between instruction steps, simply remember what PC is and restore it before continuing.

Pressing the switch to the INST STEP position will begin operation in the state indicated by the lights but then stop as though STOP had been pressed at the same time. If the stop occurs at the end of an instruction, the data displayed by the data lights depends on the instruction as follows.

LDA, STA	Operand
ISZ, DSZ	Operand
JMP	Direct→ Instruction
JSR	Direct→ Instruction
	Indirect→ Effective Address
Arithmetic and logical	Instruction
In /Out	Data

Note that the AC switches can be used between instruction steps without requiring any readjustment.

3-2.8 Program Load Switch

The PROGRAM LOAD option for the Nova 1200 will deposit the contents of the bootstrap read-only memory into locations 0-37, light the RUN indicator and begin normal operation at location 0.

3-3 CONSOLE INDICATORS

The Console indicators are composed of two (register driven) indicator strings and five individual (flip-flop driven) function indicators. The two indicator strings are displays for the ADDRESS (or present contents of the PC), and the DATA content of a memory location or an Accumulator. The five function indicators indicate the operation state of the Processor. A brief description of each display is presented in the following paragraphs.

3-3.1 Address Display

This section of the Console displays the present contents of the PC. When

performing an EXAMINE, this display should be identical to the Address configuration set into the Console Data switches.

3-3.2 Data Display

This section of the Console can display either the contents of any one of the four Accumulators, or display the contents of a memory location. For example, during an EXAMINE, the selected (by the Console Data switches) address will be displayed by the ADDRESS indicators, and the contents of the selected memory location will appear in the DATA display.

3-3.3 Operational Indicators

When any indicator is lit the associated flip-flop is in the 1 state verifying that the associated function is true. A few indicators display useful information while the processor is running, but most change too frequently and are therefore discussed in terms of the information they display when the processor has stopped. Each functional indicator is listed below with its indicative interpretation.

RUN	The processor is in normal operation with one instruction following another. When the light goes off, the Computer stops.
ION	The program interrupt is enabled (this is the Interrupt On flag).
FETCH	The next processor cycle will be used to fetch an instruction from memory.
DEFER	The next processor cycle will be used to fetch an address word in an indirectly addressed memory reference instruction.
EXECUTE	The next processor cycle will be used to reference memory for an operand in a move data or modify memory instruction.

FETCH, DEFER, and EXECUTE are the state indicators: they specify the state (the type of cycle) the processor will enter if operations are continued by pressing the CONTINUE or MEMORY STEP switch. At the most, only one indicator is lit at any one time. Unless otherwise indicated, use of any operating switch leaves the processor ready to enter the fetch state.

SECTION IV

THEORY OF OPERATION

4-1 INTRODUCTION

This section contains detailed information describing the functional relationships of the major logic sections comprising the Nova 1200 Computer. This section is essentially a continuation of the general function description Nova 1200 architecture provided in paragraph 1-2 of this manual. It is pointed out here that this text is intended to familiarize personnel with the functional operation of the major Processor logic sections, hence, the descriptions presented in this section are designed to provide basic conceptual information concerning the operation of the Nova 1200. In this regard it is suggested that Figure 1-1 (Nova 1200 Detailed Functional Block Diagram) be referenced along with the appropriate logic diagrams (bound in Section VII of this manual under separate cover) while reviewing the descriptions of this section.

The Integrated Circuits of the Nova 1200 are operationally synchronized with the CPU Clock (CLK) signal. The clock timing is arranged such that when the required enabling signals are present simultaneously with the negative-going edge of the CPU CLK signal, the corresponding logical operation will occur. Detailed information concerning the pin nomenclature and the signal requirements of the IC packages is provided in Appendix A of this manual. Signal origins for the Nova 1200 are listed in Appendix B.

4-2 DETAILED FUNCTIONAL DESCRIPTION

The discussion presented herein will consider the operation of the major logic sections of the Nova 1200 relative to the Console, CPU Operation, Basic Timing, Data Paths, Instruction Overlap, Instruction Timing Examples, Memory, and Power Supply. The drawing numbers of the logic diagrams of the major logic sections are listed below with their titles as a convenient reference.

NOVA 800 & 1200 CONSOLE	001-000089
CPU NOVA 1200	001-000088
Clock, Key Logic, AC & Memory Timing I/O Logic	(Sheet 1 of 4)
IR, MBC, and Major States, AC and Adder Controls	(Sheet 2 of 4)
Register Controls, Skip, Carry, EFA and Stop Logic	(Sheet 3 of 4)
AC's Adder, PC, ACB and MBO registers	(Sheet 4 of 4)

MA & MB Register & Control (Sheet 1 of 4)
Sense & Inhibit (Sheet 2 of 4)
X Drivers (Sheet 3 of 4)
Y Drivers (Sheet 4 of 4)

4-2.1 Console

The Nova 1200 Console contains the four following functional sections; the Power switch, the Console Display, the Console Data switches, and the Console Control switches. The functional operation of each section is described in the following discussion.

4-2.1.1 Console Power Switch. Power is applied to the Nova 1200 by turning the Console key to the ON position. As shown on the Console logic diagram, setting the key to the ON position switches 115 VAC into the Nova 1200 power supply. After power has been turned on, the power supply produces the + and - 5 volt logic power along with the + VMEM voltage for the memory. The power supply contains precision differential circuitry which monitors the + 5 volt and + VMEM voltage outputs. These circuits will produce a + 5 (volt) O.K. logic signal to indicate the + 5 output level is correct and a MEM O.K. logic signal to indicate the + VMEM voltage level is correct. The power supply also contains a Power Failure ($\overline{\text{PWR FAIL}}$) monitor circuit which is used in conjunction with the Power Monitor and Auto-restart option. The full-wave lamp voltage + V_{lamp} is also produced in the power supply. The + 5 O.K. and MEM O.K. lines carry power status signals from the power supply to the CPU.

When power is first turned on, the + 5 O.K. line provides a positive transition (to approximately + 5 volts) as the + 5 volt output rises to its proper level. This positive transition is gated into a differentiating capacitor to produce a pulse which in turn drives the RESET and PRESET gates. $\overline{\text{RESET}}$ and $\overline{\text{PRESET}}$ initializes the control logic of the Nova 1200 in preparation for operation. It will be noted (on the Console Logic diagram) that the $\overline{\text{RST}}$ (Reset) line from the Console is OR gated with the + 5 O.K. line from the power supply. Since $\overline{\text{RST}}$ becomes low when the Console Reset switch is actuated, the positive-going trailing edge (caused by releasing the switch) of $\overline{\text{RST}}$ is differentiated to generate the $\overline{\text{RESET}}$ and $\overline{\text{PRESET}}$ signals. Hence, the Console Reset switch may also be actuated to initialize the Nova 1200. The enabled output from the $\overline{\text{RST}}$/+ 5 O.K. OR gate also enables an inverter which in turn

enables another OR gate. This second OR gate also monitors the MEM O.K. and $\overline{\text{HALT}}$ lines. The enabled output from the second OR gate disables the RUN set gate controlling the input to the RUN flip-flop stage (of the KEY, RUN, DCH, and KEYM 4 Bit Discretes register).

It should be noted that the Nova 1200 CPU logic utilizes approximately 6 of these 4 Bit Discretes Registers to store discrete information defining the operational state of the processor. In each case each flip-flop stage can be set or reset independent of the state of any other flip-flop stage in the register. Further, conditions for either setting or resetting any stage is set up by decision gates at the stage input prior to the coincident arrival of a "load strobe" pulse together with the register clock pulse. At this time the output logic levels presented by the gates driving the input to each stage are latched into the register, with relatching occurring on each load and clock pulse combination. It is noted further that some of the 4 Bit Discrete Registers are also connected to perform right shift functions. Under these conditions a logical bit loaded into the DS input of the register will be shifted right on each clock with the Shift input enabled.

Therefore, on the next coincident occurrence of the PTG5 and MEM CLK pulses a zero will be loaded into the RUN stage of the register. Essentially this gating path will reset the RUN stage of the 4 Bit Discretes register if either a low $\overline{\text{RST}}$, + 5 O.K, MEM O.K., or $\overline{\text{HALT}}$ signal occurs. If all of these signals are in the high state, this gating path for setting the RUN flip-flop is satisfied. However, another gating path driven by selected Console logic functions is ANDed with the path described above to determine when the RUN flip-flop is to be set.

However, to return to the discussion concerning the $\overline{\text{PRESET}}$ and $\overline{\text{RESET}}$ functions, $\overline{\text{RESET}}$ also drives the Master Reset (MR) input of the previously described Discretes 4 Bit Register and unconditionally clears all of the outputs including RUN to the zero state. Hence, the main reason Reset ($\overline{\text{RST}}$) is a bi-functional (i.e., clears RUN and then generates $\overline{\text{RESET}}$ and $\overline{\text{PRESET}}$ on the zero-to-one transition) is to allow the processor to stop synchronously with the completion of the last instruction, thereby allowing the program to be continued from the last instruction when the processor is started again. It should be noted that since there is contact bounce in the switch - the $\overline{\text{PRESET}}$ action will occur before the switch is released but after the attempted synchronous stop.

4-2.1.2 Console Display. The Console displays two sets of data, address and memory data, along with data concerning the state of the machine. The address lights are driven from the MBO register in the CPU, and display the next address to be referenced when the machine is stopped. When the machine is running the address display is essentially meaningless and should be ignored. This is due to the fact that the MBO is continually shifting during machine operation. The data lights display the contents of the MB register of the memory that happens to be selected at the time. The lights are driven directly off the Memory bus. The states of the Carry, Run, Ion, Fetch, Defer, and Execute flip-flops are also displayed. All signals displayed on the Console are asserted negative. A non-inverting current driver is used to drive the lamps. In order to increase lamp brilliance, the usual current limiting resistor was not used in series with the lamps, but rather a parallel resistance was used. This provides a continuous flow of about ten milliamperes through the lamp which keeps the filament hot and eliminates large surge currents when the lamp is turned on.

4-2.1.3 Console Data Switches. The Data switches are tied directly to the Memory bus. The non-inverting buffers have open-collector (OC) type outputs and are normally in the off state. During Console operations the READS (DIA-, CPU) instruction causes the $\overline{\text{CON DATA}}$ line to be switched low, thereby causing all buffer outputs to go low. Those switches which are closed, designating a ' one', will put low levels on the bus. At this time all memories are disconnected from the bus by $\overline{\text{INH TRANS}}$.

4-2.1.4 Console Control Switches. All pull-up resistors on the Control switches, with the exceptions of STOP and RESTART, are connected to a common node. This node is connected to the base of Q1. (See the Nova 800 & 1200 Console Drawing #001-000089.) The circuit formed by Q1 and related Components performs two functions: 1) it senses current flowing through any of the pull-up resistors, and 2) it provides a delay of about 25 milliseconds from the time the switch is first actuated to the time $\overline{\text{CON RQ}}$ is asserted low. This delay guarantees that all switch bounce has subsided before the CPU attempts to perform the function requested. $\overline{\text{CON RQ}}$ must switch cleanly for proper operation. For example, bouncing on the leading edge of this signal may cause several EXAMINE NEXT operations to be done during a single actuation of the EXAMINE NEXT switch. The CPU, upon receiving the $\overline{\text{CON RQ}}$ signal drops the $\overline{\text{CON INST}}$ line. This allows the control switches to be connected to the Memory bus through OC-gates U1 and U2. Several switches may be ORed into one gate.

Also, for manual functions, the Memory bus is asserted positive. For example, if switch ACDP2 is actuated $\overline{\text{MEM0}}$, $\overline{\text{MEM1}}$, $\overline{\text{MEM4}}$, $\overline{\text{MEM5}}$ would go high. All other bits driven by U1 and U2 would go low. Bits 8 through 15 would also be high, but they are not used to encode the Console functions. Reset, Stop, Memory Step, Instruction Step, and Program Load are not encoded into the Memory bits, but rather unique lines are generated which define these functions.

4-2.2 CPU Operation

The CPU is organized around eight hardware registers, a shift buffer, (ACB), a program counter (PC), a CPU interface register (MBO), an instruction register (IR and MBC), and four accumulators. These eight registers are all 16 bits in length except for the PC which is 15 bits. All internal data paths are four bits wide. Consequently, it takes four separate operations to perform an add, or a register-to-register transfer. The overall speed of the machine is increased by allowing the CPU to perform several operations concurrently. This concurrency may be classified as either Nibble Overlap, in which several different four bit bytes (nibbles) are being operated on simultaneously, or as Instruction Overlap, in which the fetching of an instruction from memory is being performed simultaneously with the exception of the previous instruction.

It is important at this point to describe the function of each hardware register relative to the operation of the CPU.

4-2.2.1 Program Counter (PC). The 15 bit address of the next instruction to be fetched is held in the PC. During the fetch of an instruction, the PC is incremented by one causing it to point to the next sequential instruction. Certain instructions can change the contents of the PC. The PC consists of one chip, which is a 16 bit latch.

4-2.2.2 Instruction Register (IR and MBC). The Instruction Register stores the instruction currently being executed. The CPU decodes the data held in the Instruction Register in order to perform the instruction. The register is organized into two parts, the IR and the MBC. The IR consists of the eight high order bits, and the MBC of the eight low order bits. During an effective address calculation, the MBC contains the displacement and shifts through the source multiplexer into the Adder. The IR bits are not shifted at this time.

4-2.2.3 CPU Interface Register (MBO). The MBO is used in virtually every operation the CPU performs. It acts as a parallel-to-serial converter for data flowing into the machine from the MEM bus. The 16 bit data is loaded from the MEM bus into the MBO in parallel, after which the MBO shifts the data out, four bits at a time, into some other part of the machine. Conversely, data is shifted into the MBO from the Adder four bits at a time to be loaded into a Memory from the MBO bus. During effective address calculations, the MBO holds the present address used in relative addressing. During memory modify operations (such as ISZ) data is loaded into the MBO from Memory. The MBO then recirculates the data through the Adder back into the MBO, in the process modifying it. The modified data is then loaded from the MBO back into memory.

4-2.2.4 Shift Buffer (ACB). All data to be loaded into the Accumulators are passed through the ACB. The primary reason for this is that it is necessary to completely assemble the results from an ALC instruction before those results can be loaded back into the Destination Accumulator. This is due to the shifting and byte-swap operations incorporated in these instructions.

4-2.2.5 Accumulators. The accumulators and the PC are the only program accessible registers in the CPU. The accumulators may be logically and arithmetically manipulated under program control. All four accumulators are contained in a single 64 bit IC chip. During certain ALC operations it is desirable to be able to access two accumulators simultaneously. This is done by using two sets of accumulators, both sets containing identical data. For example, during an ADD, the source accumulator is fetched from one IC chip, and the destination accumulator from the other IC chip. Both accumulator chips are loaded at the same time from the same data.

4-2.3 Basic Timing

Basic CPU Timing may be grouped into three distinct levels, Major State cycles, TS cycles, and PTG cycles. Basically Major State cycles define what type of memory function is under way. The designated Major State of the machine is set at the beginning of each memory cycle, and remains set throughout that memory cycle. There are eight Major States, Fetch, Defer, Execute, PI, DCH Key, Keym, and a "dummy" state in which none of the other states are set. The Flow Charts (DGC Drawing #001-000106) detail the function of each Major State. In brief these functions are defined as follows:

a) Fetch set when the next word to be read from memory is to be treated as an instruction.

b) Defer set when the next word from memory is to be treated as the address of an operand or instruction, i.e., during indirect addressing.

c) Execute set when the next word from memory is to be treated as an operand. Programmed I/O operations also set Execute, but the memory is not allowed to run.

d) PI set during a program interrupt during which the program counter is stored at memory location 0. PI forces three conditions to occur; the next address to be 1, the next Major State to be Defer, and a "JMP" instruction to be loaded into the instruction register. Location 1 contains the address of the interrupt service routine. Location 0 contains the particular address at which the program was interrupted.

e) DCH set when the next memory cycle is to be a direct transfer between an I/O device and Memory.

f) Key set when a manual function is being requested from the Console. During Key, either all or part of the manual function is performed. The memory is not allowed to run during the Key cycle.

g) Keym set when the manual function requires a memory cycle, such as Examine or Program Load.

h) "Dummy" State set only when a machine stop is pending and the current instruction requires the skip conditions to be interrogated. During this state the machine increments the PC if the skip is successful in order that the address lights reflect the true next address.

The CPU uses two 150 ns. clocks, CPU clock and MEM clock. These two clocks are of the same phase, the only difference being that the CPU clock is gated, whereas MEM clock runs all the time. Three signals gate CPU clock, RUN, $\overline{\text{STUTTER}}$, and $\overline{\text{WHOA}}$. RUN simply stops the machine. $\overline{\text{STUTTER}}$ inhibits the clock for one clock cycle, allowing time for instruction decoding, and also allowing the ACTG signals to be clocked by the MEM clock. $\overline{\text{WHOA}}$ is used by certain options which temporarily slow the processor down. It should be noted that the memory timing generator continues to completion of either a Read or an Inhibit operation without CPU clock, as it is driven off MEM clock. All control flip-flops and registers in the machine are driven from one of the two clocks. All changes of state occur on the negative-going edge of the clock. The timing signals discussed above do not change the state of a flip-flop or register, but act to enable the device to change on the next negative-going edge of the CPU or MEM clock.

The TS levels are four clock cycles in length, and may be thought of as the time required to transfer a 16 bit word, at the rate of four bits per clock cycle, between two CPU registers. Each Major State consists of at least two TS levels, TS0 and TS3. TS0 occurs during the first half of the Major State, and TS3 during the second half. In fact, TS0 and TS3 are complementary, if the machine is not in TS0 it must be in TS3, and vice-versa. Certain operations require more time to be available in a Major State cycle than is provided by the two TS cycles. In this instance, a flip-flop, called Loop is set and forces the present TS cycle to repeat thereby providing the Major State with three TS time intervals. During TS0 the data is fetched from the memory and loaded into the MBO. At this point Loop is set and TS0 is repeated. The data in the MBO is shifted through the Adder undergoing some type of modification at this time. Finally, TS3 is set and the data is transferred from the MBO to the Memory and re-written. Individual clock cycles are designated by the Processor Timing Generator (PTG) levels. The levels are asserted for one clock period (150 ns.) only. PTG levels define the first and last clock periods during TS cycles. PTG2 is the last clock interval during TS0, and PTG5 is the last during TS3. $\overline{\text{PTG=0}\cdot\text{TS0}}$ is the first clock interval in TS0, and $\overline{\text{PTG=0}\cdot\text{TS3}}$ the first is TS3. Two typical uses for these signals are as follows: the least significant four bit nibble of the word is passed through the Adder at PTG=0 TS0 time, hence a 1 is also added to the Adder at PTG=0 $\cdot$ TS0 time to increment the word being transferred. PTG5, as another example, is used to enable the Major State flip-flops. Consequently, the Major States can change state only on the clock associated with PTG5. Four other timing signals are also important. These are PTG0, PTG1, ACTG0 and ACTG1. PTG0 and PTG1 are generated by a two bit ring counter which makes a cycle once every time state, i.e., every four clocks. It is from these two signals that the other PTG signals are decoded, PTG2, PTG5, etc. PTG0 is set during the two middle clock cycles of a time state, and PTG1 is set during the last two clock cycles of the time state. The two ACTG signals are used only to drive the two accumulator chips. They differ from the PTG signals only in that they can be set to be exactly one clock cycle ahead of the PTG signals. This is necessary because of the level of buffering (provided by the D Buffer and S Buffer) which accumulator data must pass.

4-2.4 Data Paths

As mentioned earlier, all memory and I/O data flowing to and from the CPU passes through the MBO Register. Instructions may be loaded directly from the MEM bus into the Instruction Register. All data transfers in the CPU take place in (4) four bit "nibbles." The destination AC and the source AC each drive a four bit buffer register. These buffers allow a nibble to be processed through the Adder concurrently with the fetch of the next nibble to be processed from the ACD and ACS chips. This saves time. Both the ACD (destination accumulator) and ACS (Source Accumulator) IC chips, as described earlier, each contain 64 flip-flops which are organized into four 16 bit accumulator registers. Both the ACD and ACS chips are loaded with the same data to facilitate "nibble" transfers through the Adder. For example, assume ACS and ACD are loaded and an add AC2 to AC1 (where AC1 is the destination accumulator) would cause a four bit nibble from AC1 to be passed through the Adder (and its associated buffer and multiplexer logic) with a four bit nibble from AC2. The four bit sum output from the Adder is then shifted into the ACB register. This path from ACS and ACD chips through the (buffers and multiplexer and the) Adder into the ACB register is traversed by the next three 4 bit nibbles from each accumulator. Data shifting in the ACB continues and the four serial outputs from the ACB are loaded through the (ACD, ACS) accumulator input multiplexer back into both the ACD and ACS chips at the beginning of the next fetch. The ALC ADD is described in greater detail in paragraph 4-2.4.2. The typical time needed to access an accumulator from one of the AC chips is 100 ns. The typical time needed to move a nibble through the multiplexer, the Adder, and load a register is 100 ns. The buffers allow these two operations to overlap so that they occur simultaneously. Instead of 200 ns being required to process a nibble, only 100 ns is with several different things happening during that 100 ns. Of course, the Adder is idle while the first nibble is being loaded into the buffer, which occurs during STUTTER as discussed in the previous TIMING section. Probably the easiest way to explain data flow is to go through a couple of representative instructions. Refer to the Block Diagram on Figure 1-1. The discussion will describe the flow mechanisms for the JMP and an ALC ADD instructions.

4-2.4.1 JMP Data Flow. The instruction starts at FETCH·TS0. During the first clock interval the MBO and the PC contain the present address. During TS0 the PC is incremented by shifting the MBO through the destination multiplexer, the Adder, and into the PC. The ACB is also being loaded with the same data. The MBO, rather than receiving Adder data, shifts back into itself through its input multiplexer. It is necessary for the MBO to contain the present address throughout the FETCH cycle in case an instruction calls for relative addressing. At the end of TS0 the instruction is loaded from Memory into the Instruction Register. An effective address calculation is now called for. This requires adding a signed displacement, contained in the MBC portion of the Instruction register, to a base address. The displacement is gated through the source multiplexer into the Adder. The base address may be zero, for page zero addressing, in which case the destination multiplexer is disabled, producing zeroes at the Adder. It may be the present address which requires the MBO to be gated through the destination multiplexer into the Adder, or it may be AC2 or AC3, causing the destination multiplexer to look at its other input. As the calculation proceeds, the sum is loaded into the MBO through its input multiplexer. Since the instruction is a JMP the PC is also loaded with these results. Consequently, at the end of FETCH·TS3 the MBO is ready to transfer the next instruction address to Memory.

4-2.4.2 ALC ADD Data Flow. The ALC ADD instruction data flow was described briefly above, however, the FETCH·TS0 portion of this instruction, as with all instructions, is identical to the JMP instruction just described. Immediately after loading the Instruction Register at the end of TS0, the CPU clock is inhibited for one clock cycle. At this time the low-order nibble of each referenced accumulator is loaded into its respective buffer. The CPU clock starts again, and those first nibbles are gated through the multiplexers, into the Adder, and finally into the ACB. During the addition of this first nibble, the second nibbles are being fetched from the AC chips and are loaded into the buffers at the end of the clock interval. This continues until four nibbles have been added and loaded into the ACB. In the meantime, it is necessary to get the next address, contained in the PC, into the MBO. This is done by enabling the outputs of the PC (open-collector) onto the output lines from the MBO multiplexer (which are feeding the four MBO shift inputs). Since the instruction is not a JMP or JSR, the PC is not modified. At the end of FETCH·TS3 the ACB contains the results of the addition, and the MBO contains the next address. The machine now starts another FETCH cycle. During the TS0 portion of this cycle the ACB is shifted, through the multiplexer/shifter, into both AC chips.

If the No-Load bit was set, the ACB would still shift through the multiplexer/shifter, but the AC chips would not be loaded. It is at the output of the multiplexer/shifter that the result is checked for a skip condition. At this time the MBO is being shifted through the Adder in order to increment the PC as previously discussed.

4-2.4.3 Mechanisms of Nibble Flow

As mentioned several times previously, the CPU handles data in four bit "Nibbles." When transferring data from one register to another, the lower order bits are always transferred first. The first clock interval would transfer bits 12-15, the second 8-11, the third 4-7, and the fourth 0-3. If an operation is to be performed upon a word, two things must be specified, the bit position inside the Nibble, and which Nibble is to be acted upon. For example, if it is desired to increment a word, such as during FETCH·TS0 time when the MBO is incremented, a carry is inserted into the low order bit of the Adder during the first clock interval, PTG=0·TS0, causing one to be added to that first Nibble. If a carry resulted from that first addition, it is stored in a flip-flop for the next clock interval where it is inserted into the Adder as a carry into the low order bit. This continues until all four Nibbles have passed through the Adder. The JSR is another example. During JSR it is necessary to force bit 0 to be zero as it is stored into AC3. A gate in the high order position of the Nibble forces the output of the multiplexer/shifter gate high (to load zero) during JSR and the fourth clock interval during the time state in which the PC is being loaded into AC3.

4-2.5 Instruction Overlap

Several instructions lend themselves to what may be called, Instruction Overlapping, or the execution of one instruction concurrently with the fetch of the next instruction. In the Nova 1200 the entire instruction execution is not overlapped with the next fetch, but rather only a portion of that execution. For example, during an ALC instruction the CPU operates upon the accumulator(s) and loads the result into the ACB register while the memory is rewriting the instruction into memory. Only the transfer from the ACB into the accumulators is overlapped with the next FETCH. The next major state need not necessarily be FETCH, but could be PI, DCH, or even KEY if the machine was stopped after the ALC and then restarted. Other instructions which overlap the loading of an accumulator with the next Major State are LDA, JSR, IO input, and the manual function ACDP. In fact, any operation which loads an accumulator is overlapped with the next major state. Another operation that is

overlapped with the next Major State is the interrogation of skip conditions for ALC and ISZ/DSZ instructions. The results of these instructions are loaded into the ACB, which during TS0 of the next Major State shifts through the multiplexer/shifter after which the data may or may not be loaded into the accumulators. The output of the multiplexer/shifter is checked for all zeroes which may or may not fulfill the skip conditions. If the skip conditions are met, the SKIP flip-flop is set at the end of TS0. If the next Major State was FETCH, the execution of that instruction is inhibited, effectively skipping it, even though it was fetched from memory and loaded into the instruction register. If the next Major State is PI, the PC that is loaded into address zero is incremented to reflect the skip before it is stored. If the next state is DCH and the SKIP flip-flop is left in the set state, appropriate action will be taken on the next FETCH or PI cycle. If the machine is about to be stopped from the Console by STOP, ISTP, or MSTP, a "Dummy State" is entered in which the skip conditions are interrogated, and the PC incremented is required. This is done to permit the address display of the Console to show the correct next address while the machine is in the stop state. The disadvantage of overlapping skip interrogation with the next instruction fetch is that the skip, if performed, requires the time to complete a memory cycle.

4-2.6 Instruction Timing Examples

This paragraph includes 13 timing diagrams which are provided as examples of machine instruction timing. The sample instruction functions diagrammed are as follows: Deposit Manual Function, Examine Manual Function, ADD 0, 1, SKP, MOV 0, 0, DSZ, LDA, STA, JMP @ 100, JSR @ 20, I/O Input, I/O Output, PI, DCH In and DCH Out. It is emphasized here that these instruction functions were selected as typical sample instructions only to facilitate the discussions of this Section. The selection of these examples should not be construed as being representative of the instruction complement of the Nova 1200, or restrictive in the use of such instructions of programming methods thereof. Several factors are in common to each timing diagram. The MEM CLOCK signal is 150 nanoseconds in duration and forms the basic timing source for all instructions. The CPU CLOCK signal is also 150 nanoseconds in duration and is generated simultaneously with MEM CLOCK except when the STUTTER or WHOA signals are present. STUTTER or WHOA inhibit the CPU CLOCK whenever either signal is present, but have no effect on the MEM CLOCK. A brief discussion of each timing diagram is presented in the following paragraphs.

4-2.6.1 Deposit Manual Function. Figure 4-1 is a timing diagram of the Deposit Manual Function. $\overline{\text{CON RQ}}$ or $\overline{\text{RESTART}}$ going low sets KEY SEEN provided RUN is not set. KEY SEEN generates $\overline{\text{PRESET}}$ for all manual functions except CONT, ISTP, or MSTP. KEY SEEN also directly clears the ACTG generator in order that its sequence be synchronized with the PTG generator when the machine starts. KEY SEEN enables the shift input to U23 causing a one to shift into KEY the first clock transition after it comes up. Depending upon the state of $\overline{\text{KEY ENAB}}$ another one is shifted into RUN, setting it. RUN causes KEY SEEN to be cleared and CPU CLOCK to be enabled. Two other signals, INH TRANS and CON INST come up with KEY. INH TRANS is transmitted to the memory to allow Memory Buffer data to be gated out onto the MEM bus. CON INST is a function of KEY being set and LOOP not being set. The CON INST signal (from the zero active level) gates the Console code for the actuated switch to Instruction Register via $\overline{\text{MEM}}$ lines 0 thru 7. The CPU CLOCK is inhibited by the STUTTER pulse which allows the contents of the Instruction Register to be decoded. The LOOP flip-flop is also set at PTG2 time in the KEY mode. The resetting of the KEY flip-flop occurs simultaneously with the setting of the KEYM (Key memory cycle) flip-flop on the PTG5 pulse (and MEM CLK). CON DATA also becomes true at this time and from its zero level enables the buffer driven Console switch configuration to appear on the MEM bus. A basic prerequisite for doing a Deposit is that it must be immediately preceded by an Examine (Core) Key operation. Under these circumstances the contents of the PC will also reside in the MBO at the end of the Examine KEYM cycle. Therefore, the contents of the MBO (PC address) is transferred into the MA at the beginning of the Deposit KEYM cycle. The LOAD MBO signal occurs on PTG2 (with LOOP not set) and loads the Console switch data into the MBO. A MB LOAD signal transfers the data on MBO lines 0 thru 15 into the buffer register of the selected memory.

4-2.6.2 Examine AC1 Manual Function. Figure 4-2 is a timing diagram of the Examine AC1 Manual Function. All of the CON REQ, KEY SEEN, KEY ENAB, KEY, CON INST, and RUN signals function as in the Examine (Core) or Deposit (Core) Key modes. STUTTER is reasserted to facilitate decoding the IR after the Console Instruction key code has been loaded into the IR. When the CPU CLOCK resumes the contents of the selected AC are "Nibbled" into the buffer register of the selected memory. The

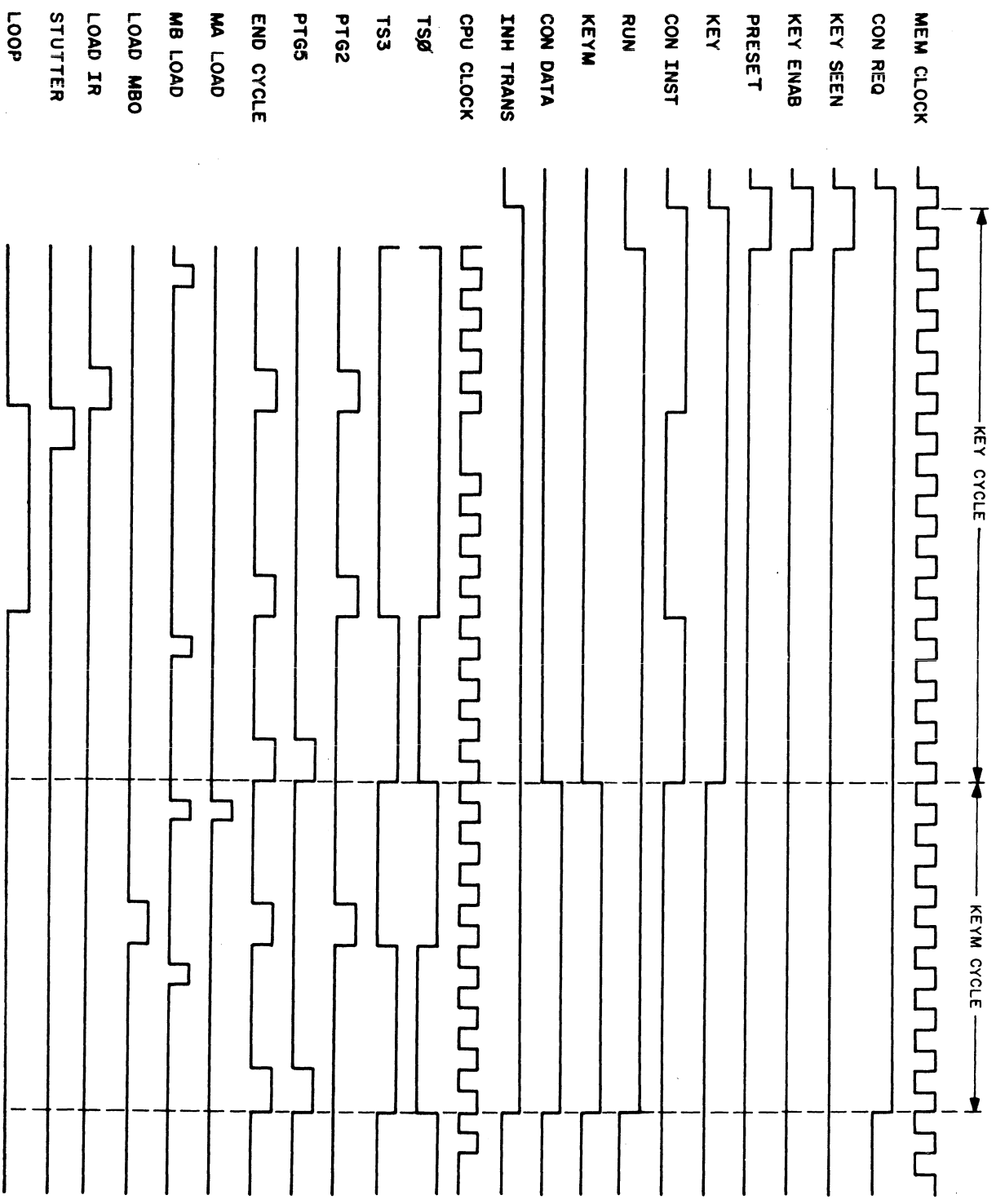


Figure 4-1. Deposit Timing Diagram

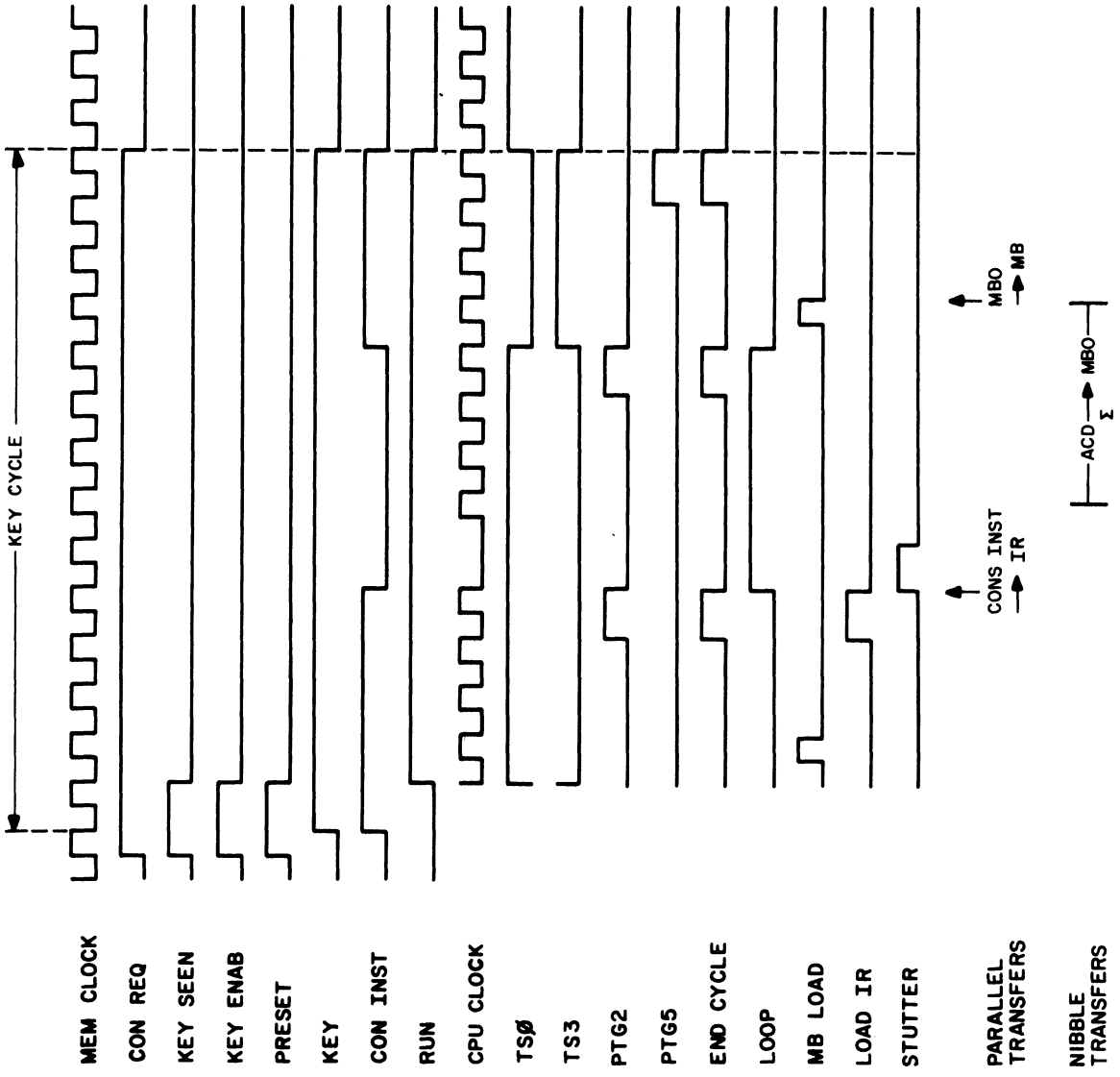


Figure 4-2. Examine AC1 Timing Diagram

outputs of this register places the data back on the MEM 0 through 15 lines. The signals on these lines in turn enable the Console lamp buffer-drivers, presenting the data in the Console data lights. As a note relating to the ACTG generator which controls the passage of each Nibble through the adder; the ACTG generator is driven by MEM CLOCK and hence is always counting, regardless of the state of RUN. During the STUTTER cycle, the PTG generator is stopped, while the ACTG generator continues to count. This causes the ACTG generator to be one clock interval ahead of the remainder of the machine. At the end of the time state, END CYCLE will clear both bits to zero putting both generators in phase again.

4-2.6.3 ADD 0, 1, SKP Instruction. Figure 4-3 is a timing diagram of the ADD0, 1 SKP instruction. As shown in the diagram MEM CLOCK and CPU CLOCK are already running with TS0 active at the start of the Fetch. PTG2 and LOAD IR occur on the trailing edge of the third CPU CLOCK, followed by STUTTER on the trailing edge of the fourth CPU CLOCK. It will also be noted that a LOAD PC (CPU CLOCK modulated) signal performs a four Nibble load of the contents of the MBO + 1 into the PC and ACB. As mentioned previously, the MBO stores the value of the PC as the last MBO operation during the previous instruction. This allows the PC to point at the next instruction. Another reason for loading the updated PC into the MBO is to allow the updated address information to be transferred into the MA at the start of the next instruction Fetch. The contents of the MBO are passed through the adder and an $\overline{\text{ADD ONE}}$ signal adds one to the LS Nibble to increment the total value. (The loading of the ACB with the updated PC is simply an offshoot of the automatic mechanism for routing adder output data and is of no consequence at this time.) A MA LOAD is also generated at the start of the Fetch cycle. This signal transfers the MBO data (updated PC) into the MA (as described above).

The instruction (which in this case is ADD0, 1, SKP) appears on MEM lines 0 through 15 and is loaded into the IR by the trailing edge of LOAD IR. As mentioned above LOAD IR is followed by STUTTER, however the ALC signal comes up on TS3, indicating an Arithmetic/Logic class instruction is to be decoded. The AC Nibble timing signals, ACTG0 and ACTG1 are derived from the MEM CLOCK signal, and hence are not effected by STUTTER's inhibit of the CPU CLOCK signal. However, the ACTG0 and ACTG1 generator logic (4 Bit Discretes register) enable is strobed by TS3. Hence, the trailing edge of the next MEM CLOCK after TS3 becomes true will produce ACTG0.

The ACTG0 and ACTG1 signals are phased together to produce the four Nibble enable

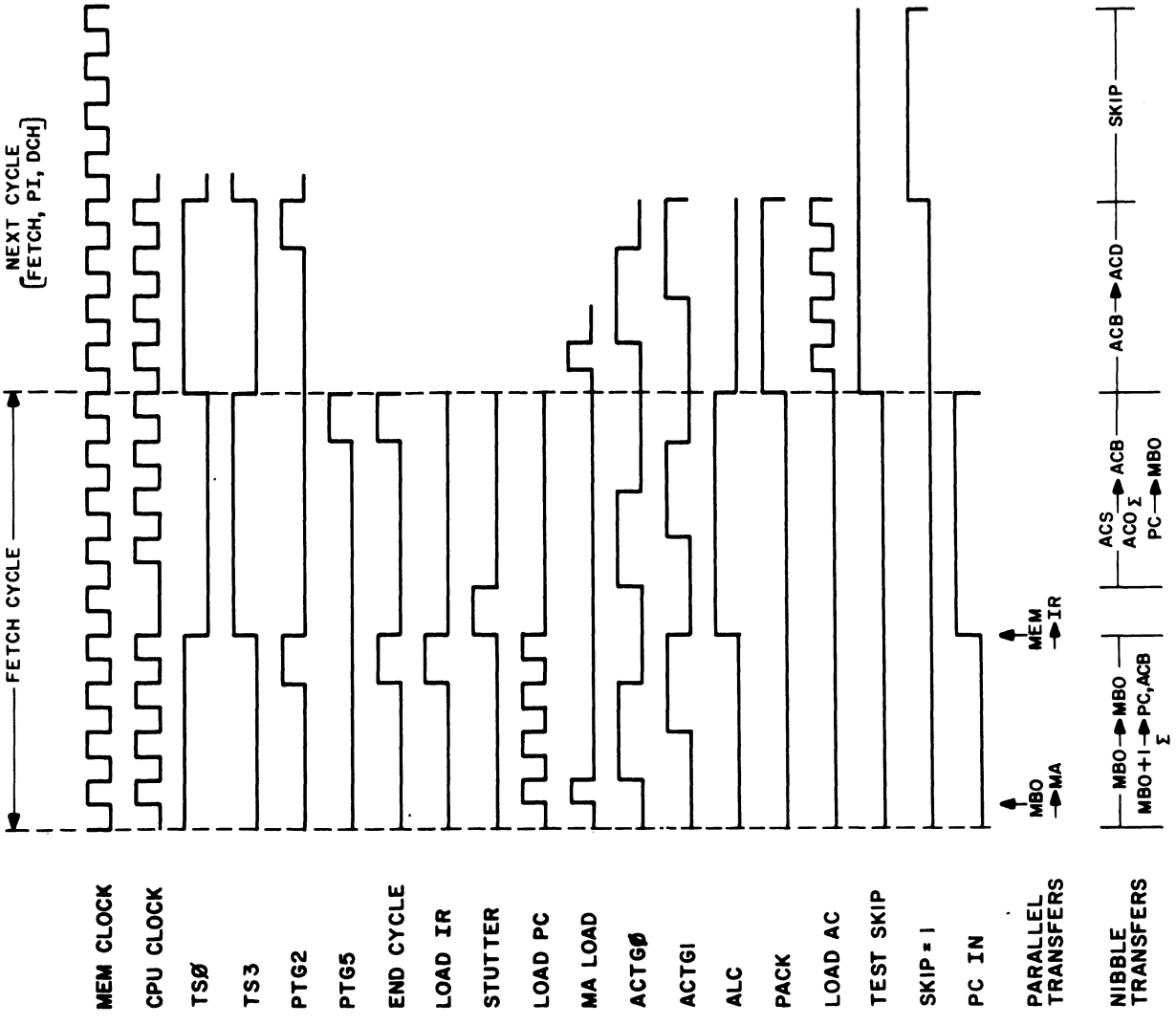


Figure 4-3. ADD0, 1, SKP Timing Diagram

signals, specifically the point where ACTG0 and ACTG1 are simultaneously low, loads or outputs the first Nibble of the Accumulator (selected by the two bit code on the $\overline{\text{ACS1 SEL}}$ and $\overline{\text{ACS2 SEL}}$ lines in the case of source AC, or $\overline{\text{ACD3 SEL}}$ and $\overline{\text{ACD4 SEL}}$ in the case of destination AC). The point where ACTG0 switches high with ACTG1 still in the low state provides input-output access to the second Nibble of the selected Accumulator. The point further along in time where ACTG1 switches high with ACTG0 now in the low state provides input-output access to the third Nibble of the selected Accumulator. The last or fourth Nibble of the selected Accumulator is accessed at the point where ACTG0 and ACTG1 both are in the high state. The appropriate accumulators are selected by the IR decode and ACS is added with ACD, the sum of which loads into the ACB. Also at this time the updated contents of the PC are loaded into the MBO. A PACK signal is produced on the trailing edge of END CYCLE allowing the AC's to be restored from the ACB at the start of the next cycle. (Actually END CYCLE and the CARRY inhibit signal $\overline{\text{LOAD CRY}}$ are used to set the PACK flip-flop.) Concurrent with PACK is the LOAD AC signal which together Nibble loads the contents of the ACB (sum of the add operation) into the ACD during the next Fetch. It should be noted that TEST SKIP set on the end of the FETCH cycle is instrumental in setting SKIP on the next Fetch cycle. The next cycle, in this case a Fetch (assuming no PI or DCH is present) will again transfer the MBO to the MA and load the ACB into ACD. The MBO will be passed through the Adder and incremented for loading into the PC as described earlier. However, since TEST SKIP was set on the end of the previous cycle SKIP will become set at this time. The updated PC is transferred back into the MBO and a new cycle will be started.

4-2.6.4 MOV0,0 Instruction. Figure 4-4 is a timing diagram of the MOV0,0 instruction. The MBO is parallel transferred to the MA at the beginning of the Fetch cycle and MBO + 1 Nibble transferred to the PC as described above. The IR is loaded and the ALC signal is produced as described previously. Up to this point the timing for this Fetch and the timing for the previous Fetch have been the same. The coincidence of bit 5 of the IR set with the ALC signal generates a positive DISABLE D MULT signal which from this level inhibits the D Multiplexer section of the Adder input. This causes the four source AC0 Nibbles to be added to zero on the next sequence of ACTG0 and ACTG1 timing signals to occur. The sum of the add operation is Nibble loaded into the ACB. The PC is also Nibble loaded into the MBO as commanded by the PC IN signal. Both operations are performed concurrently. PACK is set at the conclusion of the Fetch cycle. Assuming the next cycle is a Fetch at the start of the cycle the MA is parallel loaded with contents of

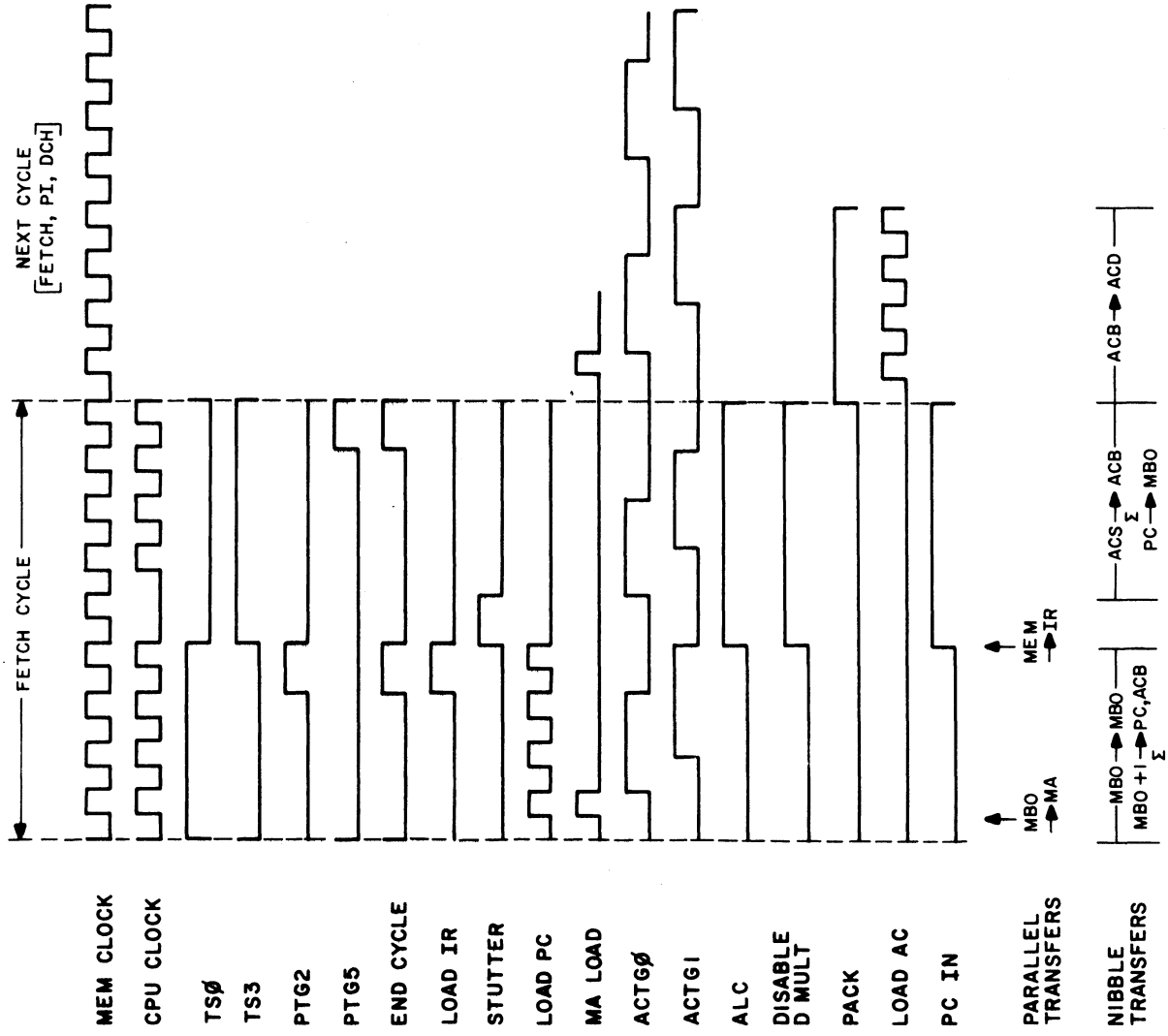


Figure 4-4. MOV 0,0 Timing Diagram

the MBO (updated PC). Since PACK is set the CPU CLOCK is gated through to modulate the LOAD AC line to accomplish the four bit Nibble load. PC updating and IR loading continues the Fetch cycle as described previously.

4-2.6.5 ISZ and DSZ Instructions. Figure 4-5 is a timing diagram for both the ISZ and the DSZ instructions. ISZ and DSZ instructions each require a Fetch cycle and an Execute cycle. The MA LOAD, LOAD PC, LOAD IR, and STUTTER signals function during the first half of the Fetch cycle as described previously. EFA becomes active on the trailing edge of END CYCLE if the instruction code for an I/O instruction ($\overline{MEM} \cdot \overline{MEM2}$ set) or the instruction code for an ALC ($\overline{MEM0}$), is not present. Hence EFA will automatically become set on each Fetch at the end of TS0 if neither instruction (I/O or ALC) code is present. The EFA signal Nibble transfers the effective memory address of the instruction from the MBC section of the IR (MBC8 through MBC15), through the S Multiplexer and the adder into the MBO register. The D Multiplexer is disable during the addition of the MBC Nibbles by the DISABLE D MULT signal (disable during TS3 time). This allows the MBC Nibbles to be added to zero. Hence the total operation is simply a data transfer from the MBC to the MBO register.

The Execute cycle starts on the active edge of TS0. The MBO data is transferred into the MA as one of the first Execute operations. The contents of the selected memory location (just read and stored in the buffer register of the selected memory) are parallel loaded into the MBO on the next Execute operation. The next operation is to pass the contents of the MBO through the adder and either ADD ONE (for ISZ) or subtract one (S0 for DSZ). Loop is set concurrent with the start of this operation so that TS0 is extended for four additional CPU Clocks. It should be noted that EFA is cleared at the start of the Execute cycle so that the D Multiplexer will no longer be disabled. Therefore, at TS0 of the Execute cycle, if the instruction is a DSZ, S0 will become active and add both the D and S Multiplexer outputs. Since EFA is not active the outputs from the S Multiplexer will be high (S Multiplexer inputs will look at the S Buffer outputs at this time, and the S Buffer outputs are shifting 1's). The effect of this addition is to add a minus one to the MBO data being Nibbled through the D Multiplexer. If the instruction is an ISZ at TS0 (and PTG = 0) of the Execute cycle ADD ONE will become active. Adder control line S0 will not be active at this time, and as a consequence the adder will look at the D Multiplexer outputs only. ADD ONE becomes active (at PTG = 0 of the TS0) on the fourth Nibble and adds one to the least significant Nibble, thereby incrementing the value of the MBO data.

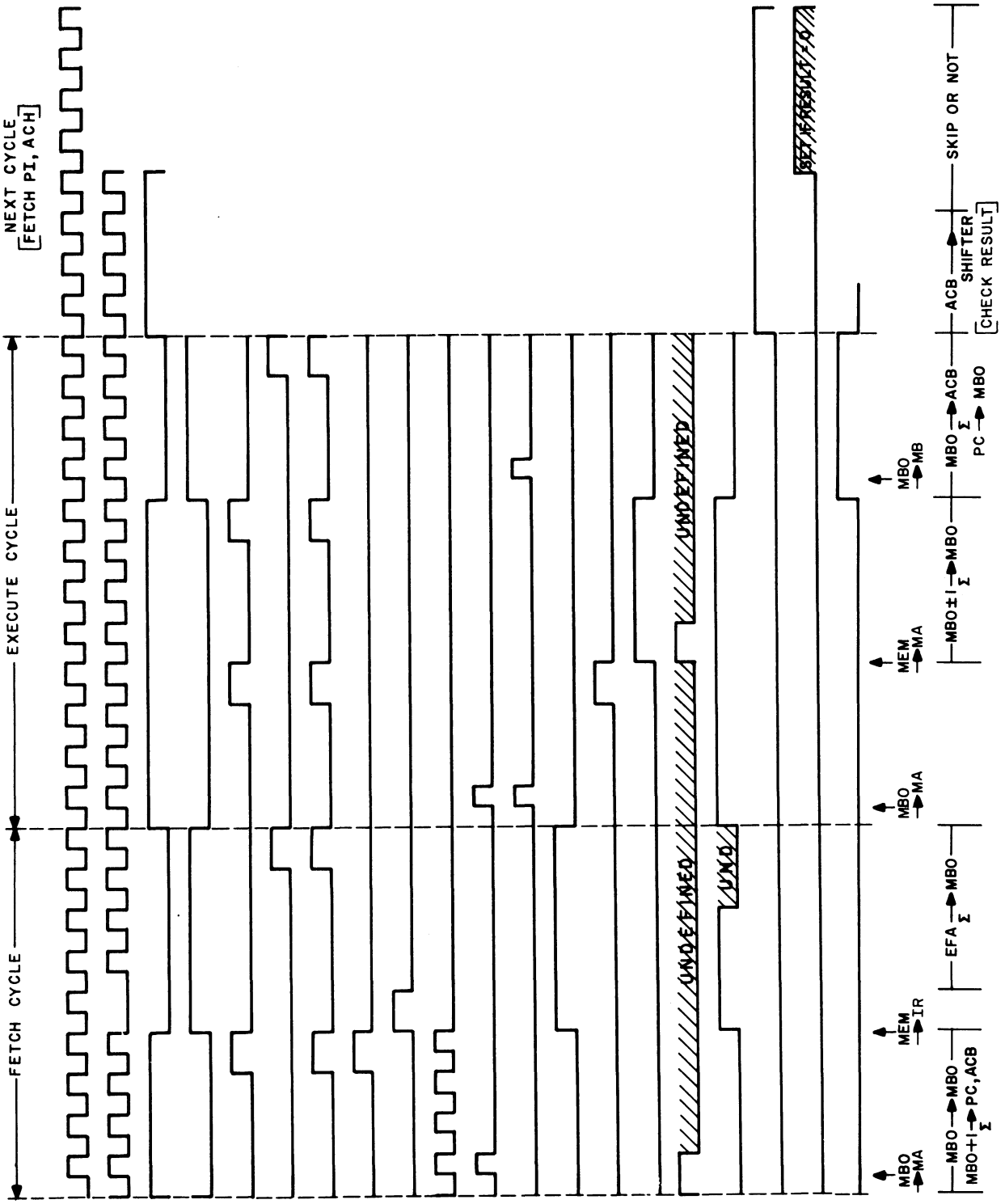


Figure 4-5. Timing Diagram for both the ISZ and DSZ Instructions

At TS3 time the MBO data is parallel loaded into the buffer register of the selected memory via the MBO bus lines MBO0 through 15. Two other Nibble transfer operations occur simultaneously with the MBO parallel transfer to the memory buffer register. One Nibble operation passes the MBO Nibbles through the (D Multiplexer and the) adder, the sum of which is Nibble loaded into the ACB. It should be noted that since the S0 control line is disabled at this time, the MBO data appears at the adder outputs. The other simultaneous Nibble operation transfers the contents of the PC into the MBO. This transfer function is enabled by the PC IN signal which becomes active at TS3 time with D + E (Defer or Execute states) SET. The ISZ or DSZ Execute cycle also sets the Test Skip flip-flop stage of a four bit discrete register. TEST SKIP enables the gating path to the Skip flip-flop for decision logic defining whether to Skip or not during the next Fetch (PI or DCH) cycle. Upon the start of the next Fetch (PI or DCH) cycle the ACB is shifted out to the AC's. (However since PACK is not set this data will not be stored in any accumulator.) The Shifter Nibble outputs are monitored (by the input gates of a storage flip-flop) for a logic 1 on any input. If a 1 occurs in any Nibble passed through the Shifter, the flip-flop will become set and remain set for the rest of the ACB data transfer. The set state of the flip-flop inhibits the gates feeding the J input of the Skip flip-flop so that the Skip flip-flop remains in the reset state. Conversely, if a logic 1 fails to appear in any Shifter Nibble, the storage flip-flop will remain reset and thereby enable the Skip flip-flop to become set. This is the basic mechanism used to determine if the ISZ or DSZ results are zero. At TS3 time the contents of the PC are loaded into the MBO by another PC IN signal. ($\overline{\text{PC IN}}$ active in the zero state is generated at TS3 of the Fetch cycle by the fact that the $\overline{\text{D + E}}$ SET and $\overline{\text{PC ENAB}}$ signals are not present.) The reset side of the Skip flip-flop provides a low active $\overline{\text{SKIP}}$ signal which inhibits the Defer section of the Major states 4 bit discrete register. SKIP also prevents effective decoding of the IR register during the present Fetch cycle. No further operations are performed and a new Fetch cycle will be started on the next TS0.

4-2.6.6 LDA Instruction. Figure 4-6 is a timing diagram for the LDA instruction. It should be noted that this timing diagram is for a basic LDA instruction only, and does not include timing for a LDA instruction with indexing provisions. All of the Fetch

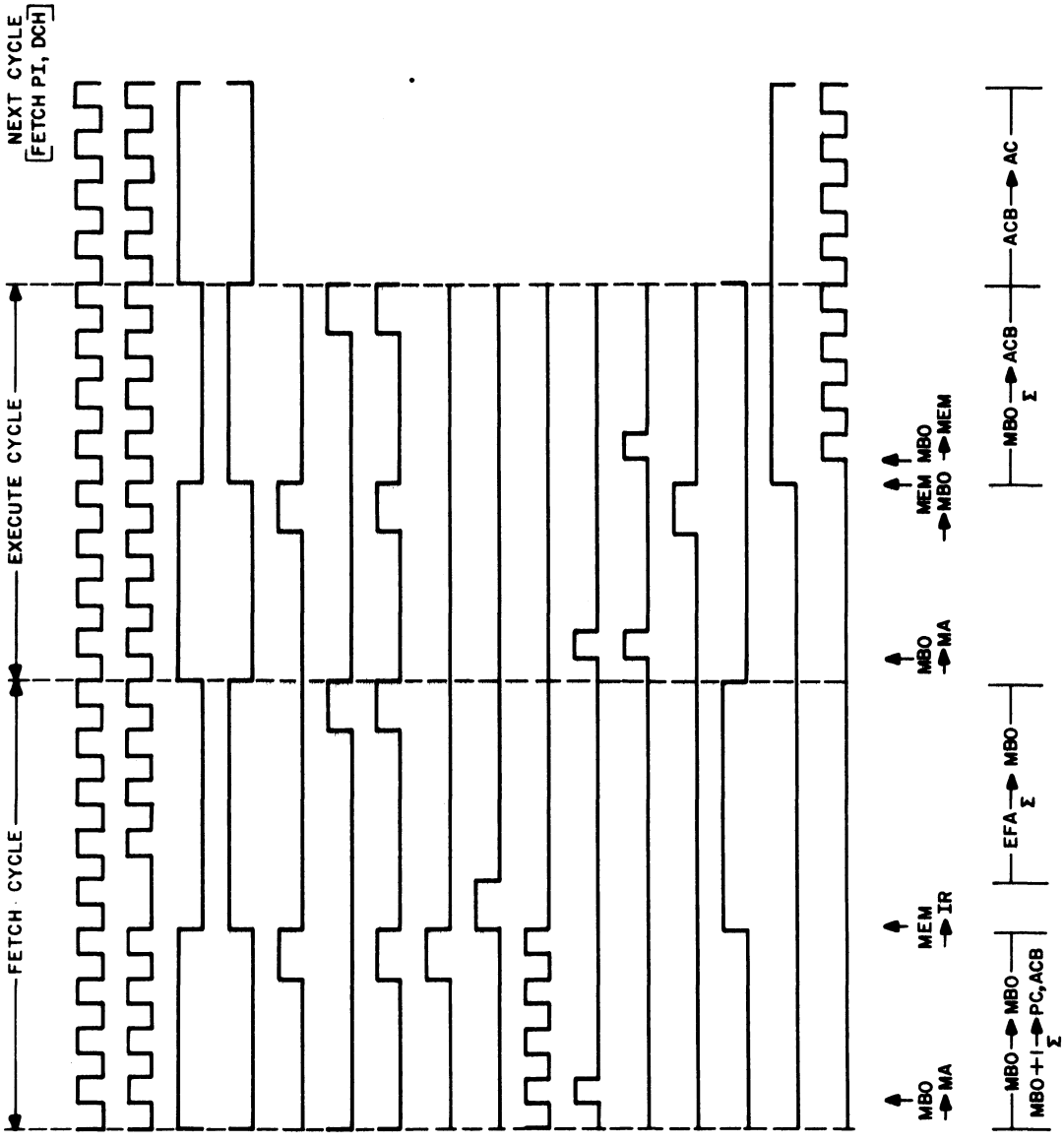


Figure 4-6. LDA Timing Diagram

timing signals previously described that appear on this diagram function in the same manner. EFA becomes set at TS3 of the Fetch. The EFA·$\overline{PTG1}$ signal becomes active at PTG1 time and enables the select input of the S Multiplexer to look at MBC input lines 12 through 15. Since it is TS3 time and if the combination of EFA, $\overline{IR6}$, $\overline{IR7}$ is present ($\overline{IR6}$ and $\overline{IR7}$ address Page 0 from the high level), a high level DISABLE D MULT signal will be produced which will disable the D Multiplexer. This causes 0 displacement (from D Multiplexer to be added to the address data on MBC 12 through 15 (through the S Multiplexer). If Page 0 is not being addressed the DISABLE D MULT signal will be at the low level to enable the data on the MBO lines 12 through 15 to be added (through D Multiplexer) to the data on MBC lines 12 through 15 (through the S Multiplexer).

Two additional controls are active in the effective address calculations. Both functions determine the state of the S0 Adder control line, but at different times during the four Nibble sequence of address calculations. The combination of EFA and PTG1 are present in their active states for the first two Nibble additions. This places the S0 line in the high state, adding the MBC displacement data (MBC8-15) to the MBO (PC) data (MBO8-15). If there is a negative displacement, and Page 0 is not being addressed, S0 will be held high for the last two addition Nibbles. This function is controlled by the state of the MBC8 line, which from the high level indicates a negative displacement and causes 1's to be added (via the MBC inputs to the D Multiplexer) during the last two Nibbles. This procedure effectively provides Sign Extension (over the eight MSB's) for negative displacement numbers. If the MBC8 line is low signifying a positive displacement, the S0 line will be allowed to switch low for the last two Nibbles, thereby causing the Adder to simply transfer the eight MSB's of the MBO back into their respective positions within the MBO register. (If Adder control lines S0-S2 are all low, the Adder will copy the A input to its outputs. The A inputs in this case is the MBO data which is output by the D Multiplexer.) A truth table for the Adder S control inputs is provided under the description of the 74181 Integrated Circuit package located in Appendix A of this manual.

After the effective address has been calculated and stored in the MBO, TS0 initiates the start of the Execute cycle. The first CPU CLOCK of this cycle parallel loads the effective address contained in the MBO into the MA. The resultant accessed memory data loaded into the buffer register of the selected memory is parallel loaded into the MBO via the MEM bus. The contents of the MBO are then parallel loaded back into the buffer register

of the selected memory by the MB LOAD signal. This allows the data just accessed from memory to be restored to the same location. PACK is set by END CYCLE by virtue of the fact $\overline{\text{LDA}}\cdot\text{E}$ is true signifying the Execute cycle of an LDA instruction, and as mentioned previously PACK enables the LOAD AC to be modulated by MEM CLK. PACK is active for eight clock cycles during the first four clocks of which the contents of the MBO are Nibble loaded into the ACB. The second four clocks allow the contents of the ACB to be Nibble loaded into the selected AC. This second group of clocks occur at beginning of the next Fetch, PI or DCH cycle. It should be recalled that the ACTG0 and ACTG1 are running continuously as described previously and are present to clock each Nibble into the selected AC, even though ACTG0 and ACTG1 are not shown on the LDA timing diagram.

4-2.6.7 STA Instruction. Figure 4-7 is a timing diagram for the STA instruction. Many of the timing signals shown on this diagram have been discussed previously. The contents of the MBO are loaded into the MA at the beginning of the Fetch. The contents of the MBO are also incrementally updated through the Adder and Nibble loaded into the PC. Since the instruction Fetched is not an I/O or ALC the EFA flip-flop will be set on END CYCLE. The effective address is calculated as previously described. The MBO is loaded into the MA of the selected memory at the start of the Execute cycle. The contents of the selected AC in the ACD chip are Nibble transferred through the MBO input multiplexer into the MBO. The accumulator is selected by decode gating of IR bits 3 and 4 which selectively enable the $\overline{\text{ACD3 SEL}}$ and $\overline{\text{ACD4 SEL}}$ destination accumulator address lines. The MBO input multiplexer selects the destination accumulator output lines ACD0 through ACD3 for Nibble loading into MBO. After loading the contents of the MBO are parallel loaded into the buffer register of the selected memory by the MB LOAD signal. The data in this register is subsequently deposited in the memory location addressed by the MA which was loaded with the effective address at the start of the Execute cycle. The PC IN signal is active during TS3 as a parallel operation during the Execute cycle, and allows the contents of PC to be Nibbled into the MBO (by timing pulses PTG0 and PTG1).

4-2.6.8 JMP @ 100 Instruction. Figure 4-8 is a timing diagram for the JMP @ 100 instruction. The JMP @ 100 instruction contains a Fetch cycle followed by a Defer cycle. The instruction is a program Jump indirect through the contents of memory location 100.

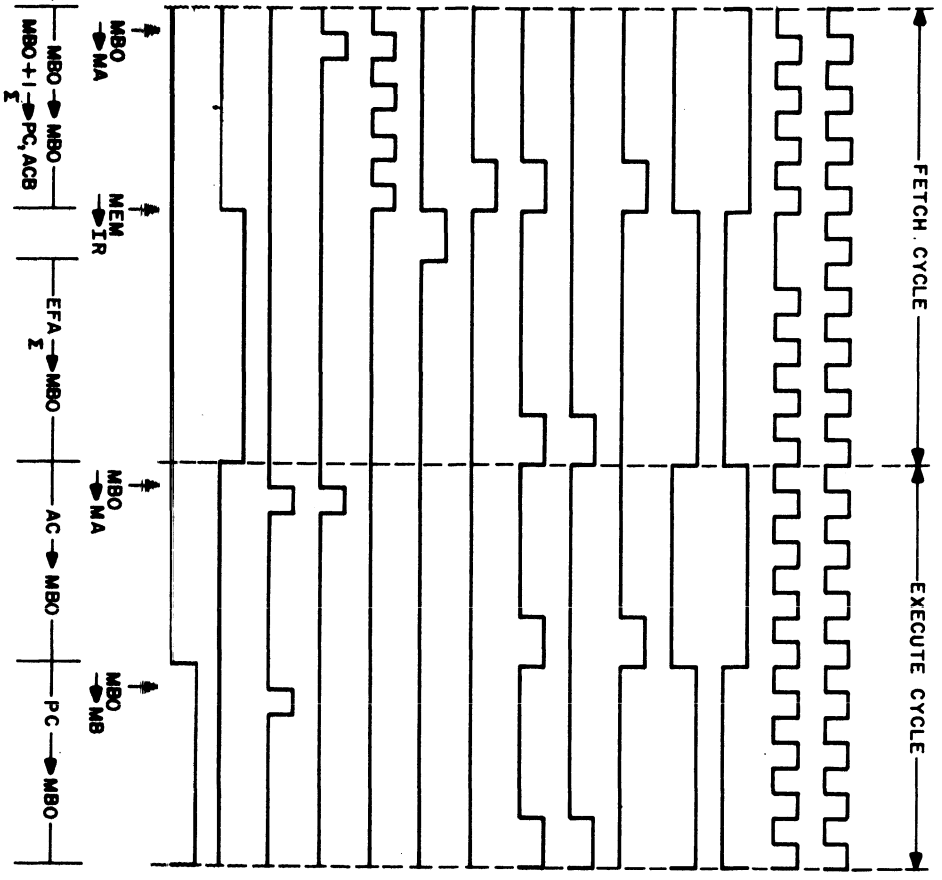


Figure 4-7. STA Timing Diagram

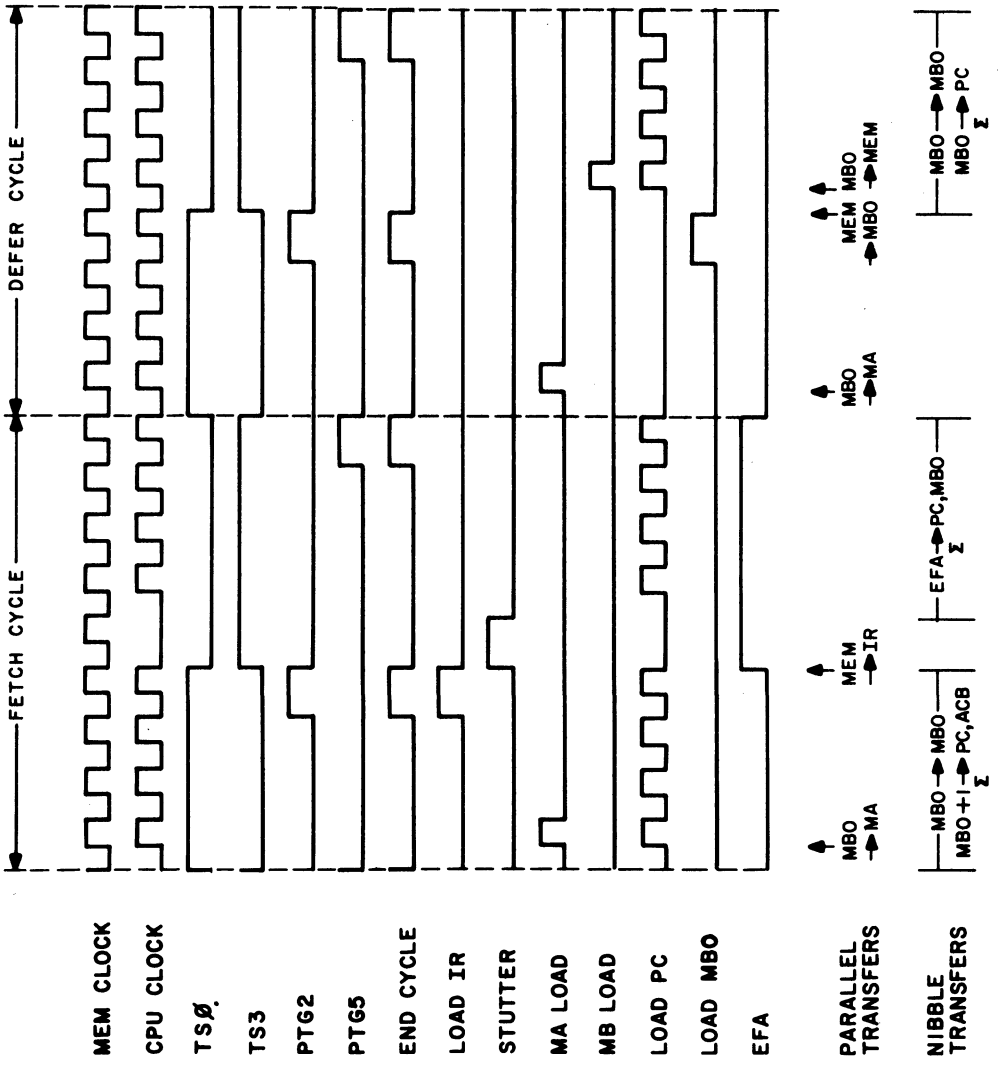


Figure 4-8. JMP @ 100 Timing Diagram

The instruction first accesses memory at the start of the Fetch in the usual manner. After the JMP indirect is decoded, the effective address between the current PC (held in the MBO) and location 100 is calculated with the new effective address sum being Nibble loaded into the PC and MBO. At the start of the Defer cycle the MBO is loaded into the MA (in this case the address data for location 100). The resulting memory data is loaded into the MBO and the new contents of the MBO are parallel loaded back into the buffer register of the selected memory while also being Nibble loaded into the PC. The contents of present MBO will be parallel loaded into the MA at the start of the next Fetch cycle.

4-2.6.9 JSR @ 20 Instruction. Figure 4-9 is a timing diagram for the JSR @ 20 instruction. The JSR @ 20 instruction contains a Fetch cycle followed by a Defer cycle. The instruction is a program jump to a subroutine indirect through the contents of memory location 20 which is defined as an autoincrement location. The instruction first accesses memory at the start of the Fetch in the usual manner. After the JSR indirect 20 is decoded, the effective address between the current PC (held in the MBO) and location 20 is calculated with the new effective address sum being Nibble loaded into the PC and MBO. At the start of the Defer cycle the MBO is loaded into the MA (in this case the address data for location 20). Also at this time the value of the updated PC, calculated during the Fetch cycle and loaded into the ACB, is Nibble transferred from the ACB into AC3 via the AC input Multiplexer and Shifter. END CYCLE sets the PACK and WAS JSR flip-flops on the end of the Fetch cycle, and PACK enables the ACB to be Nibble loaded into AC3. The Memory output data is loaded into the MBO via the MEM bus.

LOOP is set if the MBO address is one of the autoindexed locations. LOOP is set as a function of the ADDER TEST control line during the Defer Cycle. ADDER TEST is controlled by a group of gates (which are synchronously timed with the shifting of the various MBO Nibbles to mask off certain bits to detect addresses 0020 through 0037), to set LOOP (via ADDER TEST) thereby commanding an additional 600 nanoseconds for autoincrementing or autodecrementing the index location. The first gate blocks out the first or least significant Nibble (LSN) at PTG0·TS0 time. The second gate is active for all four Nibbles. This gate requires that Bit 11 of the word must be a logic 1 during the 2nd Nibble (PTG=1·TS0). For the other three Nibbles (Nibbles 1, 3 & 4) Bit 11 must be a 0. The third gate active at

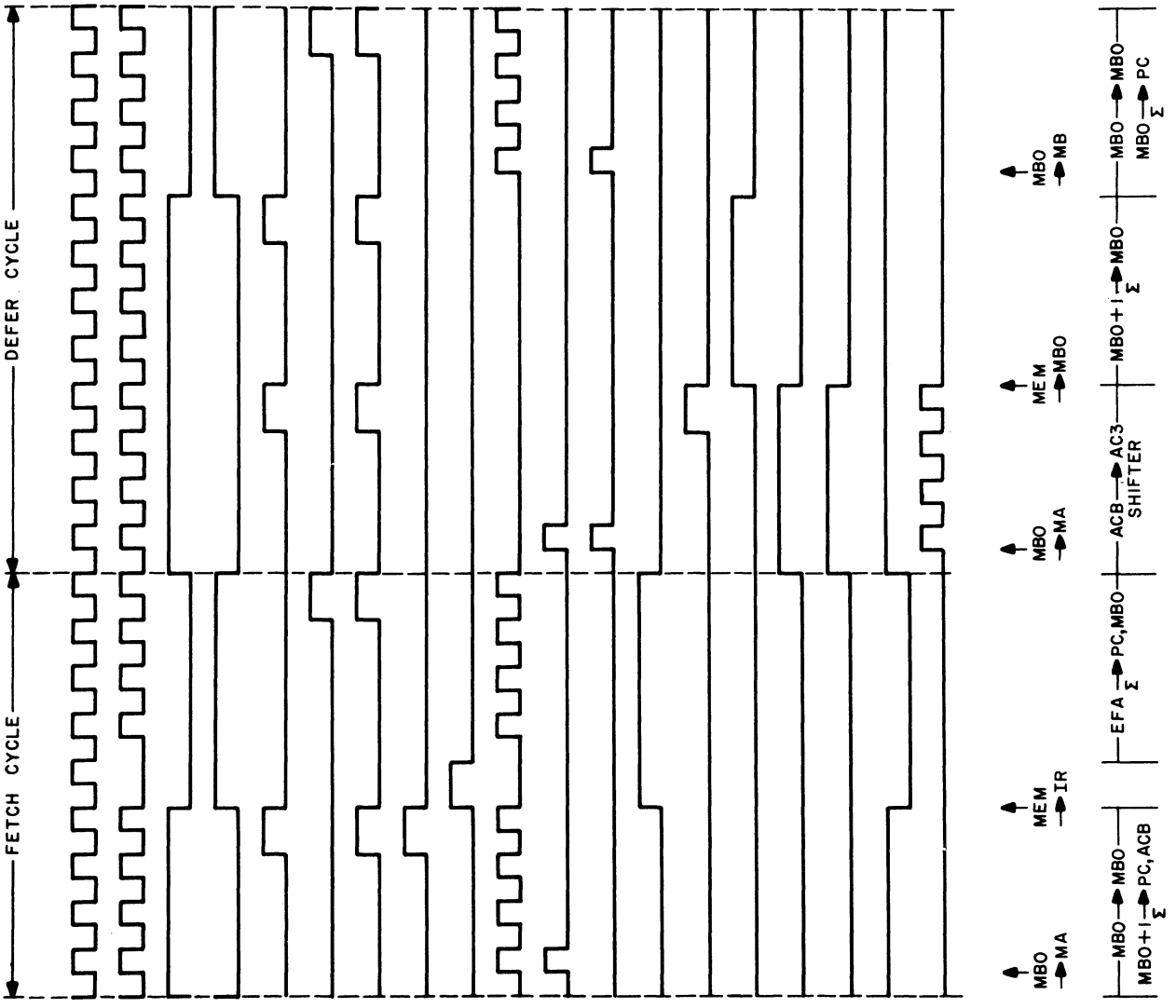


Figure 4-9. JSR @ 20 Timing Diagram

$\overline{\text{PTG2}}$ time blocks out the MSB of the Most Significant Nibble. A flip-flop is also provided to remember if the 2nd or 3rd Nibble did not meet the gating requirements. Hence at TS0 of the Defer Cycle, if an autoindex address is present the ADDER TEST line will be switched high and set the LOOP flip-flop.

The contents of the index location (presently held in the MBO are incremented through the adder and the sum is Nibble loaded into the MBO and the ACB registers. The MBO is then parallel loaded back into the buffer register of the selected memory while also being Nibble loaded into the PC. The Memory will write this incremented data back into the autoindex location. The contents of the present MBO will be parallel loaded into the MA at the start of the next Fetch cycle.

It is pointed out here that $\overline{\text{MBO12 SAVE}}$ defines autoincrementing, or autodecrementing in any selected indexing operation. MBO8 is loaded into a flip-flop stage of a 4 bit discretes register at PTG5 time. The output of this stage is identified as $\overline{\text{MBO12 SAVE}}$. At load time MBO8 holds MBO12 effectively loading MBO12 into $\overline{\text{MBO12 SAVE}}$. If MBO8 is a 1, $\overline{\text{MBO12 SAVE}}$ will also be a 1, and thereby enable the autoincrement by enabling the $\overline{\text{ADD ONE}}$ input to the adder. Conversely if MBO8 is a 0, $\overline{\text{MBO12 SAVE}}$ will also be in the 0 state, and thereby enable the autodecrement by switching the S0 adder control line to the high state.

4-2.6.10 I/O Input Instructions. Figure 4-10 is a timing diagram for the I/O input instructions. All of the timing signals relating to the first half of the Fetch cycle have been described previously. The I/O input timing is differentiated from other timing cycles by the INPUT, READ I/O, and DATIA, B, C, or I/O SKIP pulses. Each of the three pulses occur on the trailing edge of the first CPU CLOCK after the STUTTER decode.

The MB LOAD, $\overline{\text{READ IO}}$, and $\overline{\text{INH TRANS}}$ signals are concerned with moving data into the MB register in the memory. The leading edge of MB LOAD loads the data on the IO bus into the MB register. MB LOAD is generated during IO input instructions at EXECUTE· TS0 time in order to load the buffer register in the selected memory (MB) with the data on the IO bus. Asserting $\overline{\text{READ IO}}$ causes the multiplexer input to the MB in the memory to look at the IO bus, rather than the MBO bus. $\overline{\text{INH TRANS}}$ is used to turn off the MEM bus drivers in all memories. The $\overline{\text{INH TRANS}}$ function is active when it is desired to place other data on the MEM bus, as is done during certain Console manual functions or Program Load,

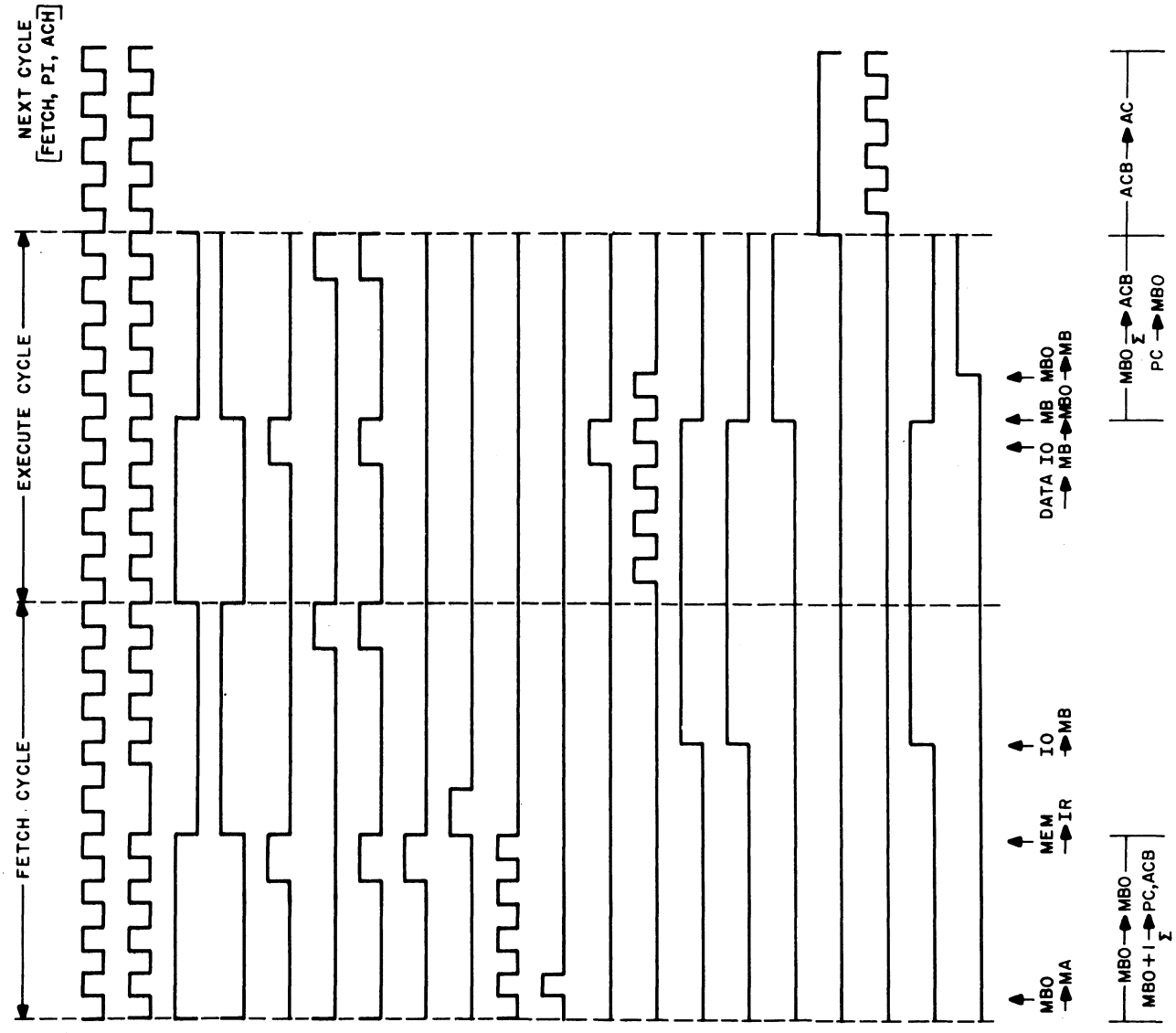


Figure 4-10. I/O Input Timing Diagram

or when it is necessary to place all zeroes on the bus. Zeroes are loaded into the MBO in this manner during the last clock interval of a Program Load sequence, and during the last clock interval of the major state preceding a PI cycle.

The INPUT flip-flop is another important control element which becomes active at PTG = 0· TS3 time if the combination of IO (F + D) and IR 7 are present. Setting INPUT enables IR bits IR 5 and IR6 to be decoded to select either the DATIA, DATIB, DATIC or IO SKIP lines. During the Execute cycle the buffer register in memory is loaded with the data present on I/O Data lines $\overline{\text{DATA0}}$ through $\overline{\text{DATA15}}$. This data is in turn transferred via the MEM bus into the MBO. The contents of the MBO are passed through the adder the sum of which is loaded into the ACB. Concurrent with this operation the PC is Nibble loaded into the MBO. On the next Fetch, PI or DCH cycle the contents of the ACB will be loaded into the selected AC. The ION, STRT, CLR, and IOPLS functions are decoded from MBC bits 8 and 9 at $\overline{\text{PTG = 1·TS3}}$ time.

4-2.6.11 I/O Output Instruction. Figure 4-11 is a timing diagram for the I/O output instructions. All of the timing signals relating to the first half of the Fetch cycle have been described previously. The I/O output timing is differentiated from other timing cycles by the DRIVE IO DATOA, DATOB, DATOC, STRT, CLR, IOPLS signals. The MB LOAD, $\overline{\text{DRIVE IO}}$, and $\overline{\text{INH TRANS}}$ signals are concerned with moving data out of the MB register in the memory. The leading edge of MB LOAD loads the data on the MBO bus into the MB register. During IO output, it is generated at the first clock interval of EXECUTE in order to load the MBO into the MB. Switching DRIVE IO low causes the selected memory to place the contents of its MB onto the IO bus.

After the STUTTER decode the contents of the selected AC is passed through the adder, the sum of which is loaded into the MBO. As mentioned previously the EFA flip-flop is not set during an I/O Fetch, therefore the S Multiplexer is disabled by the $\overline{\text{EFA·PTG1}}$ signal. The D Multiplexer is enabled by the low level DISABLE D MULT and $\overline{\text{ACD OUT}}$ signals. The effect of this condition adds the selected AC to zero and stored the sum in the MBO. The MBO is loaded into the buffer register of the selected memory (MB) at the start of the Execute cycle and since the $\overline{\text{DRIVE IO}}$ signal is active, the data is gated directly from the MB onto the I/O bus lines $\overline{\text{DATA0}}$ through $\overline{\text{DATA15}}$. LOOP is also set to allow one of the I/O output strobes (DATOA, DATOB, or DATOC) to be decoded from IR bits 5 and 6 and set up on the appropriate line. The ION, STRT, CLR and IOPLS functions are also

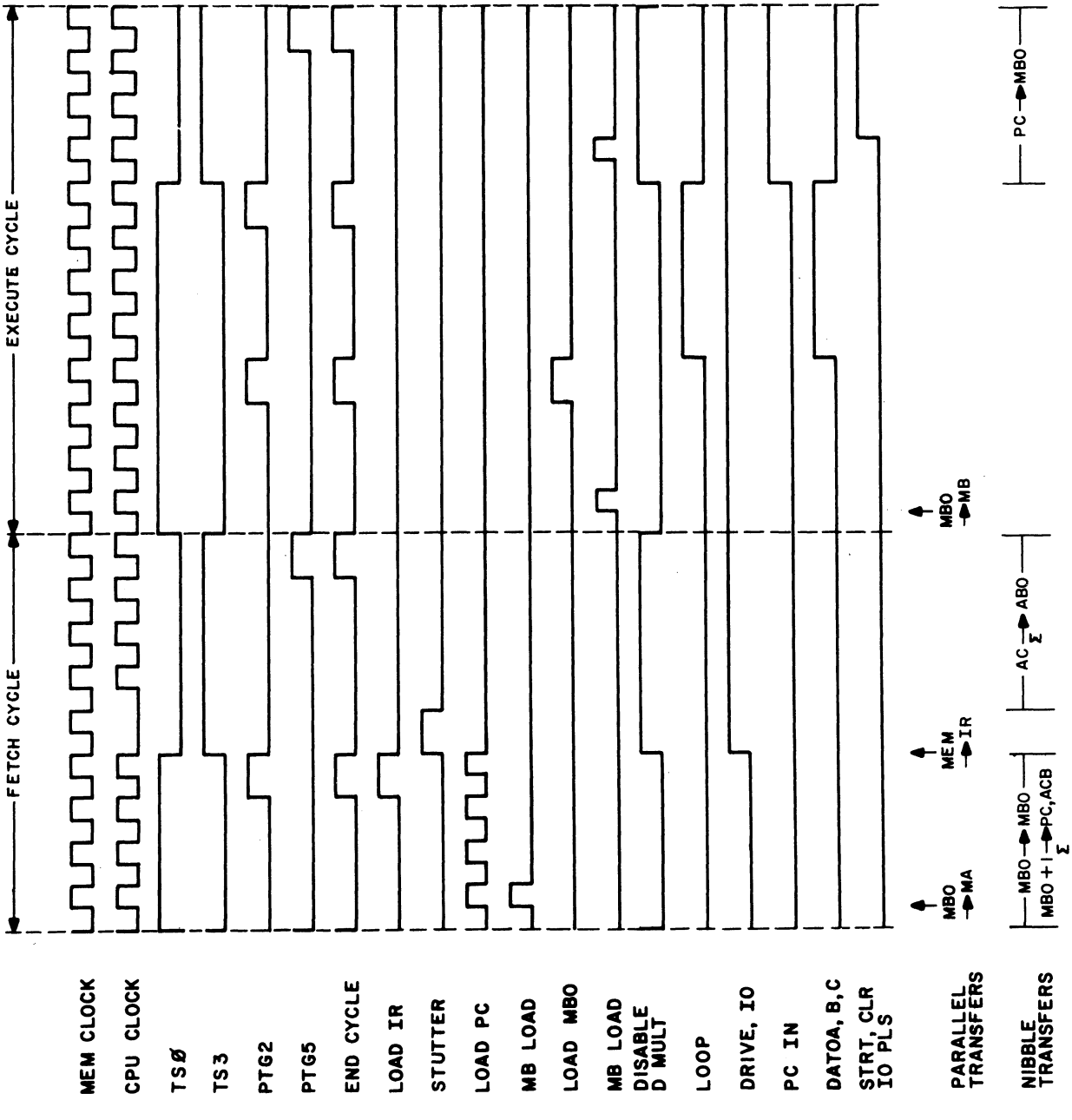


Figure 4-11. I/O Output Timing Diagram

decoded from MBC bits 8 and 9 at $\overline{PTG = 1 \cdot TS3}$ time. PC IN provides one of the final operations of the I/O output Execute cycle by transferring the updated PC into the MBO.

4-2.6.12 Program Interrupt (PI) Cycle. Figure 4-12 is a timing diagram for the PI cycle. As mentioned previously the MBO is zero at the start of a PI cycle. The contents of the MBO are loaded into the MA and the PC is Nibble loaded into the MBO. LOOP is set by END CYCLE. If the ION flip-flop has been set and an INTR signal is received, the PI stage of the major states register will become set, starting the PI cycle. The $\overline{PI}$ output at the low level clears the ION flip-flop. If TEST SKIP was set during the previous cycle the SKIP flip-flop will be set and the MBO will be incremented as it passes through the adder. Conversely if TEST SKIP was not set, the MBO will not be incremented as it passes through the adder. The contents of the MBO are transferred to the MB and zeroes are shifted into the IR. The effectively loads a JUMP to location 0 instruction into the IR. The MBO is set to a "1" at the end of the PI cycle. The MBO is transferred to the MA to access the instruction in location 1 and the MBO is loaded with the subsequent memory output. The contents of the MBO are then passed through the adder and Nibble loaded into the PC. The next cycle will be a Fetch cycle to perform the JUMP operation.

4-2.6.13 DCH IN, DCH OUT Cycles. Figure 4-13 is a timing diagram for the DCHIN, DCHOUT cycles. The DCHA stage of a four bit discretes register is set by a low level $\overline{DCHR}$ signal with either Defer or Execute set. The output from the DCHA stage in turn sets the DCH stage in another four bit discretes register. Setting DCH initiates the DCH cycle. The MTG, READ 1, READ 2, and STROBE signals occur on each memory reference instruction even though they were not shown on previous timing diagrams. RQENB is coterminous with timing signal MTG1. At the start of the DCH cycle, the IO address data is gated into the MA by the READ IO (from the low level). If the DCH operation is an Input operation, I/O data will appear on the bus as signalled by READ IO (from the low level), and written into the MB by the MB LOAD signal on the trailing edge of the fifth MEM CLOCK of the DCH CYCLE. The contents of the MB is then written into the accessed location by the INHIBIT signal (from the high level). If the DCH operation is an Output operation, READ IO will be at the high level causing the MB multiplexer to look at the MBO input lines rather than the $\overline{DATA}$ I/O bus lines. The data originally read out into the MB from the accessed memory location is placed on the I/O bus lines $\overline{DATA0}$ through $\overline{DATA15}$ by a low level DRIVE IO signal. This output data is strobed by the DCHO signal which is enabled by $MTG3 \cdot \overline{MTG1}$

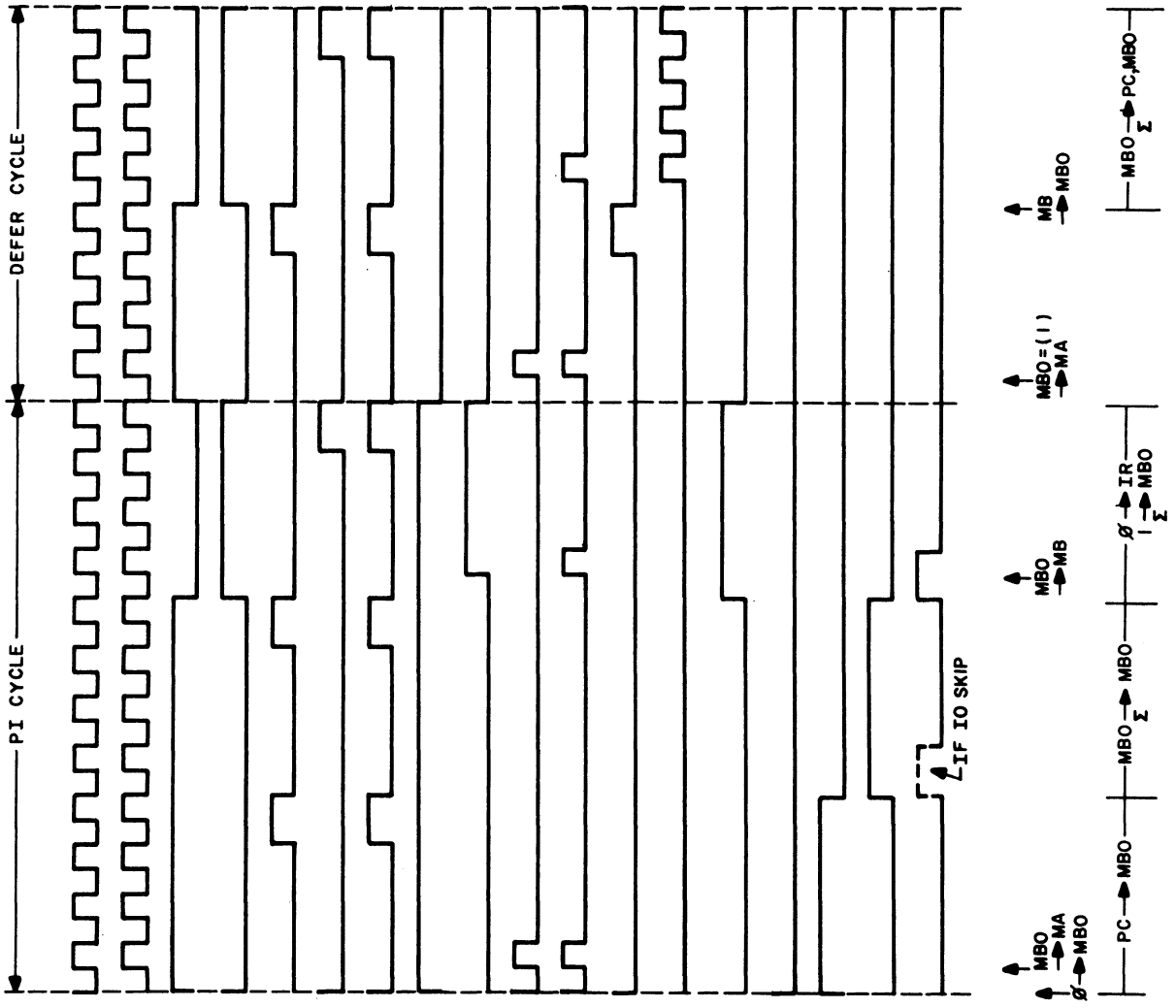


Figure 4-12. PI Timing Diagram

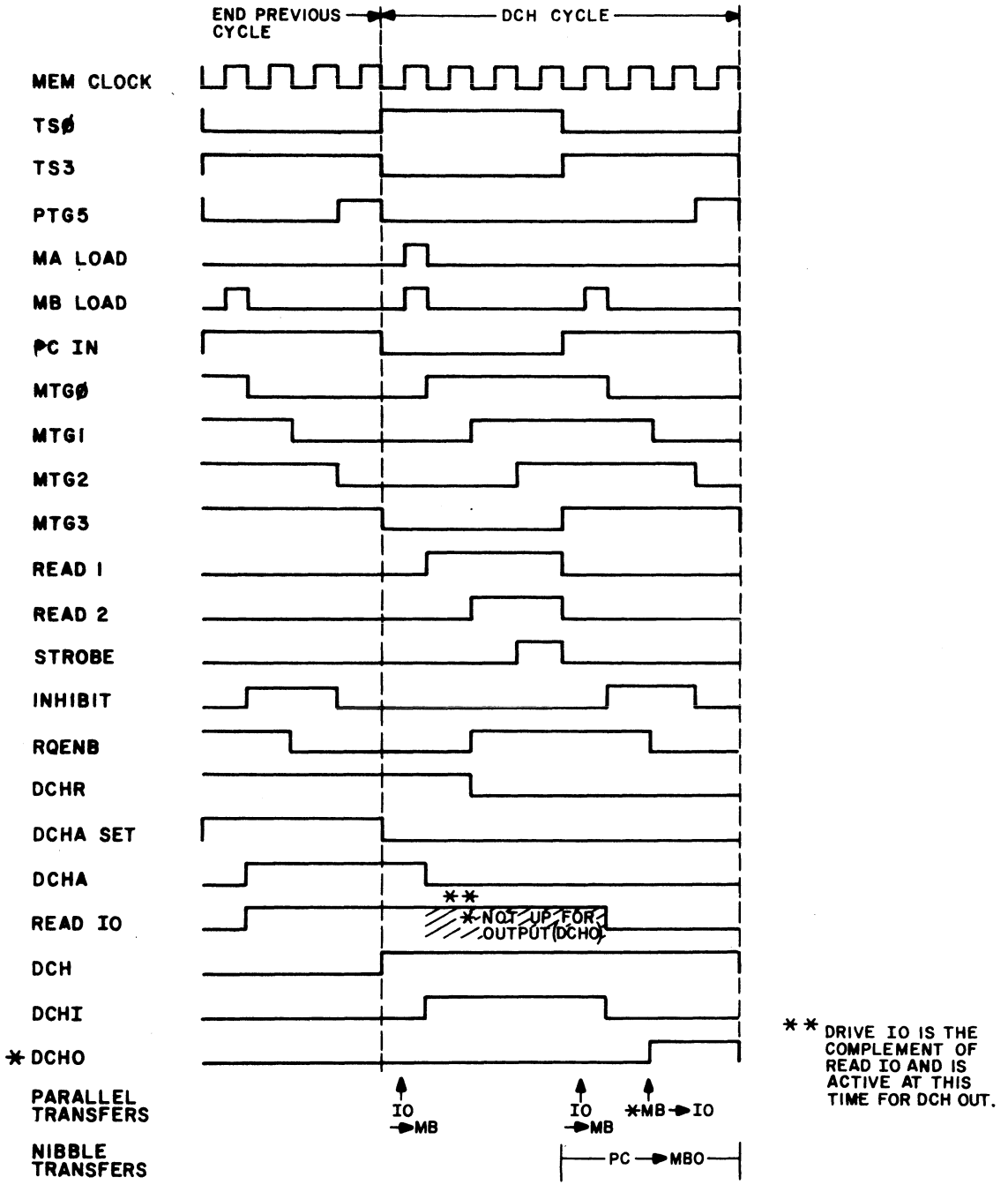


Figure 4-13. DCH IN, DCH OUT* Timing Diagram

timing signals. It is pointed out that the MBO is loaded from the MB at PTG2 time, hence with READ IO high (DRIVE IO enabled) the MB load pulse will be loading the data present in the MBO into the MB, and since this data is identical to the present contents of the MB, the contents of the MB will not be changed. Therefore, the INHIBIT signal will rewrite the original data back into the memory location accessed by the DCH OUT cycle.

4-2.7 Memory

As mentioned previously each 1K, 2K, or 4K block of memory contains a MA (memory address) register, a MB (memory buffer) register, a set of MEM bus drivers, a set of IO bus drivers, and an input multiplexer which allows the MA and the MB registers to be loaded from either the MBO bus or the IO bus. MA Bits 1-3 (4K) or 1-4 (2K) determine which memory is to be selected. (Reference Drawing 001 000103 (sheets 1-4) bound in Section VII of this manual under separate cover.) Timing signals for memory are generated in the CPU and are derived from the MTG (Memory Timing Generator) clock signals. MTG0 through MTG3 are gated together in various combinations to produce the actual memory control signals. MA bits 4 through 9 control the Y Driver current logic, while MA bits 10 through 15 control the X Driver current logic. The stack sense lines are differentiated to drive the input gates to the direct set inputs of the MB register. Therefore, with the READ 1 and READ 2 signals present, STROBE A, B, C, and D enable the sense output gates to place the memory data onto the sense lines $\overline{\text{SNS0}}$ through $\overline{\text{SNS15}}$. The configuration on these lines will be unconditionally jam transferred into the MB register. The outputs of the MB drive both the $\overline{\text{DATA0}}$ through $\overline{\text{DATA15}}$ and $\overline{\text{MEM0}}$ through $\overline{\text{MEM15}}$ bus gates. The MB is also loaded (at other times) from its input multiplexer which selects either the I/O Bus $\overline{\text{DATA0}}$ through $\overline{\text{DATA15}}$ lines as an input, or $\overline{\text{MBO0}}$ through $\overline{\text{MBO15}}$ as an input. Either source is gated into the MB by the MB LOAD signal from the CPU. The MA is loaded from the 15 least significant outputs of the same multiplexer identified as MD1 through MD15. The MA load control signal, $\overline{\text{MA LOAD}}$ is also produced in the CPU.

As mentioned previously, the memory must be SELECTED in order to operate. Memory writing requires an INHIBIT signal from the CPU together with a SELECT decode signal. These signals are gated together to enable the INH GATE A, INH GATE B, and WRITE MEM signals to be produced. WRITE MEM turns on the memory current logic, while the INH GATE A and B signals strobe the reset outputs (INH0 through INH15) from the MB register into the

inhibit drivers. The presence of inhibit current in any core winding reduces the coercive effect of the normal memory current such that the associated core will not become set. Thus, the data configuration held in the MB will be effectively written into memory via the inhibit drivers.

4-2.8 Power Supply

Reference Drawing 001-000091 bound into Section VII of this manual (under separate cover) when reviewing the descriptions presented in this paragraph. Source power for the power supply may be either standard 117 VAC lines or 220 VAC lines regulated to ±20%, and capable of supplying 325W. Power supply output voltage and currents are as follows:

Voltage	MAX. Current	Primary Use	Remarks
+15v($+V_{mem}$)	9A.	XY & Inhibit drivers in core memory.	short-circuit & over-voltage protected.
+11v($+V_{lamp}$)	2A.	Console lamps	full-wave rectified, non-filtered, non-regulated.
+5v	12A.	IC logic	short-circuit & over-voltage protected.
-5v	1A.	Sense Amplifiers in core memory	Will tolerate ground shorts up to 10 sec.
-15v	2A.	Not used in basic machine. Provided for options and customer convenience.	non-regulated.

The power supply generates four signals which are used by the processor, $\overline{\text{POWER FAIL}}$ is used by the Power Monitor and Auto-restart options to set a Power Low flag in the processor causing an interrupt to be generated when the line voltage falls to eighty percent its nominal value. MEM OK goes low when $+V_{mem}$ drops to a point the memory will no longer function reliably. +5 OK goes high when the +5volt output is approx. 4.4 volts. This edge generates a reset pulse in the processor initializing it at power turn on. 60 Cycle is a sine wave used by the real-time clock. It may be either 50 or 60 Hertz in frequency, depending on the line frequency.

The power supply is composed of five separate, functional parts, +30 VNR generation, +5 volt regulator, +15 volt regulator, -5 volt regulator and associated

circuitry, and control signal generation.

4-2.8.1 +30 VNR Generation. +30 VNR is a filtered, non-regulated voltage which is used by the +15 volt and +5 volt regulators. At nominal line voltages, +30 VNR will be between 32 volt and 35.5 volts depending upon the load. The two transformer primaries are wired in parallel for 117 VAC operation, and in series for 220 VAC operation. The fan is always wired in parallel with the BLK-BRN primary, causing it to be effectively wired into an auto-transformer during 220 VAC operation. A maximum of six amperes may be drawn from the convenience receptacle. The convenience receptacle is switched, and is protected by the two 10 ampere fuses. Consequently, improper operation of any device connected to the convenience receptacle may cause power to be lost by the CPU itself.

4-2.8.2 +5 Volt Regulator. A self-oscillating, switching regulator is used to generate the +5 volt output. (A simplified diagram of the regulator appears in Figure 4-14 below.) The operation of this type of regulator is described briefly in the following discussions. A reference voltage is compared with the output voltage. If the output voltage is less than the reference voltage, a series pass transistor is turned on. The transistor drives an LC filter. When the pass transistor turns on, the output voltage of the filter rises linearly until the output voltage equals the reference voltage. At this point the pass transistor is turned off. The field across the inductor now reverses, allowing the inductor to recover through the commutating diode. The output current is now drawn from the energy stored in the LC filter. When the output voltage falls below the reference voltage, the cycle repeats. The output voltage will be sawtooth waveform, centered around the nominal output voltage.

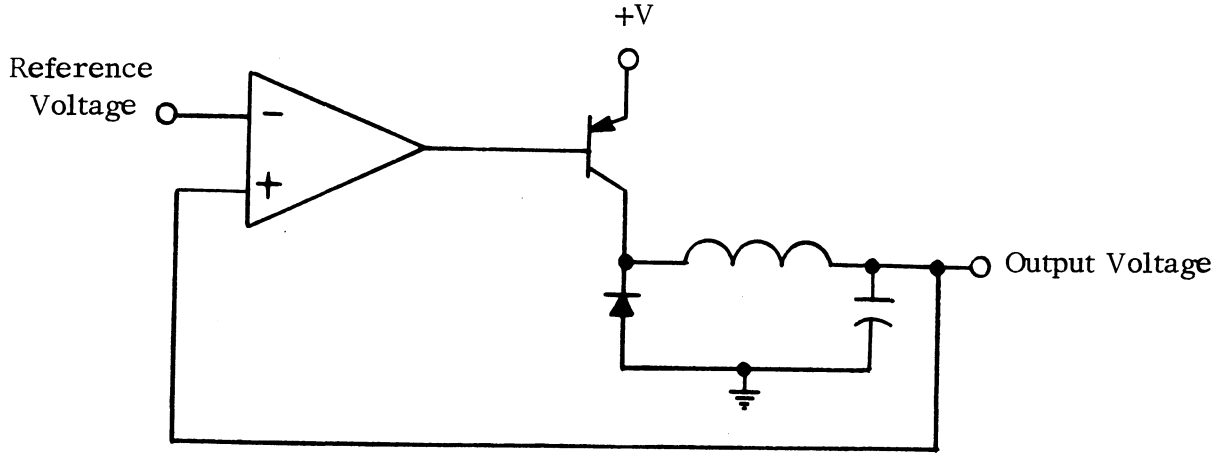


Figure 4-14. +5 Volt Regulator Functional Diagram

The +5 volt regulator consists of chip U2 and its associated circuitry. U2 contains circuits which generate a reference voltage, a voltage comparator, and disable circuits used by the short-circuit protection circuits. The reference voltage at pin 6, nominally 7.15 volts, is divided down to 5 volts and applied to one input of the comparator at pin 5. The output voltage is brought directly to the other side at pin 4. When the output voltage is less than 5.2 volts approximately, pin 11 will drop to +6 volts, turning both the pre-driver, GE D43C5, and the pass transistor, 2N4399, on. When the output voltage reaches 5.4 volts, the voltage at pin 11 switches to approximately +30 VNR, turning both transistors off. The difference in switching points is due to the hysteresis added by returning the collector of the 2N4399 through a 220K resistor to pin 5. The frequency of oscillation will vary with load. As the load increases, the frequency increases, reaching a maximum of about 25 KHz at full load.

Short-circuit protection is provided by the circuit to the right of the LC filter. The current flowing through the pass transistor generates a voltage drop across the 0.2 ohm resistor. The voltage is monitored by the 2N4403 transistor, and when the current rises above the level defined to be short-circuit current the 2N4403 turns on, applying a positive level to pin 2 of U2. This forces pin 11 to go high, turning the pass transistor off. This state is maintained by the 2N4400 transistor which turns on once the 2N4403 turns on, and holds the base of the 2N4403 negative with respect to its emitter, even after current has stopped flowing through the pass transistor. The result is that the output voltage drops to zero and stays there, even after the short is removed. To restore power, AC power should be turned off, allowing the 2N4400 and 2N4403 to unlatch, and then turned back on again.

Over-voltage protection is provided by an SCR 2N4441. When +5 volts rises above approximately 7.5 volts the SCR turns on, blowing the 15 ampere fuse which removes +30 VNR from the regulator.

4-2.8.3 <u>+15 Volt Regulator.</u> The +15 volt regulator differs from the +5 volt regulator in only two respects. The output voltage, rather than the reference voltage is divided down, before being applied to the voltage comparator. Secondly, the output voltage is reduced with rising ambient temperature. This causes the memory currents to be reduced with temperature, which is necessary to maintain good margins. At 55 degrees centigrade, the output voltage will drop to +14.4 volts ± 0.1 volts. The short-circuit protection and the over-voltage protection are

identical to the + 5 volt circuits. Notice though, that rather than using two separate over-voltage circuits, the two voltages are "OR'ed" into the SCR circuit.

4-2.8.4 -5 Volt Regulator. The -5 volt output is generated using a simple linear regulator. Since the regulator is essentially an emitter follower with no feedback involved, load regulation is marginal. Short-circuit protection is provided by a 4 ohm, 5 watt resistor. Because of the high power dissipation in the resistor during short-circuit conditions, it is only possible to guarantee shorts of ten seconds or less duration. The -15 volt output is taken directly from the rectifiers. It will of course change with the line voltage. With -15 volts and -5 volts under full load, a maximum ripple of .75 volts can be expected on the -15 volt output.

$+V_{lamp}$ is non-filtered and is used only by the console lamps. In order to avoid coupling lamp current into logic ground, a separate ground is provided between the supply and the Console along with $+V_{lamp}$ line.

4-2.8.5 Control Signal Generation. Chip U1 monitors power supply voltages to verify that all voltages are within specified limits. Power for this chip is provided from the +5 volt line. $\overline{\text{POWER FAIL}}$ will go to ground when + 30 VNR is equal to approximately 23 volts. MEM OK drops when + 30 VNR is at 21.5 volts. The time between $\overline{\text{POWER FAIL}}$ dropping and MEM OK dropping is important as this is the time in which the power fail service routine must store the state of the machine before all power is lost. The machine is forced to halt once MEM OK has gone low. The guaranteed worst time between the two signals dropping is 3 milliseconds.

SECTION V

MAINTENANCE

5-1 PERIODIC INSPECTION

The Nova 1200 has few mechanically operational components, hence it requires a negligible amount of mechanical inspection. Any lubrication schedules are confined to any mechanically operational peripherals used with the Nova 1200, e.g., card punch, line printer, tape punch or reader, etc. The Nova 1200 Console switches should be checked periodically for proper operation and switch spring tension. The Console indicators of the various displays should also be observed to detect any burned out indicators. The Static Tests described in paragraph 2-3.2.1 of this manual may be performed to check out the Console controls and indicators.

5-2 PREVENTIVE MAINTENANCE

It should be pointed out that it is impossible to compile a schedule of Preventive Maintenance routines which will satisfy the requirements of all customer applications. The routines outlined in this paragraph may be scheduled against two critical factors: the minimum down-time that can be tolerated by the installation, and the periods of least activity when these procedures may be performed. Obviously these two factors will vary from installation to installation, however, the smaller the minimum down-time becomes, the more frequent preventative maintenance is required, and this must be distributed over the periods of least activity.

IO devices, because of their general mechanical nature, benefit the most from a scheduled preventative maintenance program. In addition a certain percentage of malfunctions can be detected while in the process of occurring. Diagnostic routines should play a major role in preventive maintenance programs. Suggested items that should be included are as follows:

a) Diagnostics - Run exerciser daily for a reliability check of the entire system. All other diagnostics should be run at least once weekly.

b) IO devices - Clean daily, removing the dust that normally accumulates as the device is used. Check for excessive vibration, overheating of bearings, and signs of excessive mechanical play or wear. Check punch and teletype for wear and fraying. Empty the punch chad box and remove chad from within the device itself. Clear the type face of the teletype. Look for and remove excess oil and grease from within the devices.

c) General - Check all power and IO cables for fraying or wear. Check all plugs and connectors; tighten if necessary. Check the cooling fan in the computer power supply for proper operation.

d) Lubrication - Following the lubrication schedules as set forth in the IO device pamphlets. This requirement is perhaps the most important phase of a preventive maintenance program.

5-3 SPECIAL TOOLS AND TEST EQUIPMENT

The following is a list of special tools and test equipment recommended for efficient maintenance of the Nova 1200.

MULTIMETER	SIMPSON MODEL 260 OR EQUIVALENT
OSCILLOSCOPE	TEKTRONIX 453 OR EQUIVALENT
LONG LEAD PROBES	TEKTRONIX P6010-10X OR EQUIVALENT
CURRENT PROBE	TEKTRONIX P6022 OR EQUIVALENT
EXTENDER BOARD	DGC 107-000007-02
WIRE WRAP TOOL (24 GAUGE)	GARDNER DENVER Model 14AX2 OR EQUIVALENT
IC TEST CLIP	MANUFACTURED BY A P INC. P ainesville, Ohio (part no. 923700)
SOLDERING IRON	WELLER ISOLATED MODEL W-TCP OR EQUIVALENT

5-4 DIAGNOSTIC PROGRAMS

The Nova 1200 Diagnostics are individual programs which together test all logical operations of the computer system. Individually the programs test various logic areas of the computer and IO. The majority of the diagnostic routines are capable of diagnosing malfunctions down to the logic level. The diagnostics provide a means of measuring the performance of the system on a repeatable basis. Copies of the diagnostic tapes as well as individual program documentation are part of the software package delivered with the Nova 1200. Individual program documentation provides information as to operating procedures, error interpretation, console switch settings and logical areas tested. Certain diagnostics are normally part of the daily and weekly preventive maintenance routines.

NOVA 1200 DIAGNOSTIC PROGRAMS

PROGRAM	DESCRIPTION
Address Test	Routine to test the memory address section logic.
Checkerboard III	Worst case memory noise test. Program verifies proper operation of sense amps, inhibit drivers, and memory currents.
Nova 1200 Logic Test	Gate by gate test of CPU Logic (less IO).
Nova 1200 Instruction Timer	Routine to test CPU clock logic, prints instruction times of basic Nova 1200 instruction set.
Exerciser	Reliability test - tests CPU logic, TTY Reader, punch, high speed paper tape reader, paper tape punch and real time clock. Halts on error.
Arithmetic Test	Exercises the arithmetic and logical instructions of the Nova computers.
Nova 1200 Teletype Test	Gate by gate test of TTY logic, PI system and IO Bus logic.

PROGRAM	DESCRIPTION
Reader/Punch Test	Routine to test high speed paper tape reader and punch.
Real Time Clock Test	Routine to test Real Time Clock logic.
Nova 800/1200 Power Shut Down Test	Test retention of memory data on power loss. Tests power monitor auto restart option.

5-5 TROUBLESHOOTING PHILOSOPHY

Effective trouble shooting is accomplished in a minimum of time by following a series of logical steps. The ultimate aim is to effectively pinpoint the actual problem using all information available. Locating the malfunction is then the next logical step. The following is a suggested plan for effective casualty analysis:

a) Investigation - record the state of the machine on error occurrence. Look for obvious symtoms including operator error, loose plugs or connectors, blown fuses or tripped circuit breaker.

b) Isolation - through the use of diagnostic programs or console trouble shooting techniques attempt to isolate the malfunction to a particular board.

c) Component Isolation - Isolate the faulty component using an oscilloscope and short diagnostic loops either toggled in at the console or as part of a diagnostic. Selecting the correct external synch is of importance at this point.

d) Replace the faulty component and retest by running the diagnostic that originally failed.

e) Record for future reference, the symptoms, cause, unique trouble shooting method/s used to isolate the malfunction.

5-5.1 Memory

Address decoding and data word transfer failures are the types of memory malfunctions most frequently encountered. The inability to store or fetch a word from or into a selected core location is usually an indication of the former while storing or fetching a word which is modified by one or two bits is an indication of the latter. Address test and checkerboard are memory diagnostics designed to verify memory reliability. The two programs will

detect and, in most cases, identify the cause of a malfunction. Address test is primarily intended to test address selection logic and verifies the ability to uniquely address all core locations. Checkerboard is a worst case noise test designed to detect the picking up or dropping of bits in a data word transfer. In the case of intermittent failures it may be desirable to revert to console troubleshooting, utilizing short closed loop routines which are toggled in. Programs such as the one illustrated below are valuable in resolving failures.

SAMPLE DIAGNOSTIC LOOP PROCEDURE

1. Deposit data word in AC2
2. Deposit program in core
3. Start - Program halts - Load address in console switches and continue

LOC			
0000	063077	DOC 0, CPU	:Halt Inst.
0001	060477	DIA 1, CPU	:Reads Switches
0002	044011	STA 1, 11	:Store Addr
0003	052011	STA 2, @ 11	:Data to Addr
0004	000001	JMP. -3	:Loop

Note: The address can be varied by changing the contents of the console switches. The above routine will store the contents of AC2 (Data word) into the address in AC1. It is useful in monitoring Read/Write currents and the Inhibit current. Current loops are provided on the memory assembly to facilitate the use of a suitable current probe (Tektronix P6022 or equivalent) for current measurements. However, only one current loop is provided for measuring the Inhibit current. The physical locations of the Read/Write and Inhibit Current loops on the memory assembly may be referenced in Section VI of this manual, the Illustrated Parts List for the Nova 1200 (under separate cover.) The Nova 1200 power supply is considered a factory repairable unit only, and if the memory currents measured differ extensively from the values listed in Table 1-2 notify Data General Field Service.

Maintenance other than lubrication, minor adjustments and part changes should be performed by DGC personnel or respective manufacturer representatives. Lubrication

should be performed in accordance with the appropriate manual listed below:

Applicable Manuals

Teletype	-	Technical Manual 33 Teletype writer sets Bulletin 310B Volume I
		Technical Manual 33 Teletype writer sets Bulletin 310B Volume II
		33 Page Printer set ASR, KSR and RO Parts Bulletin 1184B
High Speed Punch	-	Technical Manual High Speed Tape Punch set (BRPE) Bulletin 215B
		High Speed Tape Punch set (BRPE) Parts Bulletin 1154B
High Speed Reader	-	(Digitronics Model) (2540EP)
		Perforated Tape Reader Operation and Maintenance Manual
		Lubricating Materials
Teletype-Keyboard		KS7470 (oil) KS7471 (grease)
Typing unit		KS7470 (oil) KS7471(grease)
Reader		KS7470 (oil) KS7471 (grease) Lupriplate 105

Applicable Manuals (Continued)

High Speed Punch	-	(BRPE11) KS7470 (oil) 145867 (grease)
High Speed Reader	-	SAELO (oil)
		Recommended spares one each
High Speed Reader	-	Lamp incandescent Digitronics TLNBF009 GE (08805) (P/N 1638)
High Speed Punch	-	Drivebelt #135097

5-6 COMPONENT REPLACEMENT

The replacement of a component requires care to prevent damage to circuit board etch. Clipping a component from the circuit board rather than unsoldering is the preferred method. Excessive heat from a soldering iron may result in damage to the component being replaced. The use of a soldering iron with an isolation transformer, a small copper alligator clip as a heat sink and a delay between the soldering of individual pins of a chip are recommended. When the extender board is used, the weight of the board under test should be supported by a non-conductive material. Replacing a Console switch or Indicator requires the removal of the Console subassembly. The following is the procedure to be followed when replacing a Console (Data) switch:

1. Remove the four 6/32 nuts attaching the Console subassembly to the enclosure frame.

2. Remove the eight 2/56 screws holding the circuit board assembly to the Console casting.

3. Replace the defective switch and reassemble in reverse order.

To replace a Console Indicator follow steps 1 & 2 above and in addition remove the three Phillips panhead screws holding the Benelex to the circuit board. Replace and reassemble in reverse order.

5-7 IC IDENTIFICATION

Table 5-1 is included in this section to facilitate any troubleshooting procedures that require identification between any Nova 1200 IC reference number (U1-Un) and the original manufacturer's part number. This list may be referenced as a bridge between the logic configurations shown on the various CPU and Memory logic diagrams and the IC (package) pin definitions summarized in Appendix A of this manual.

Table 5-1. IC Identification List

CHIP	CPU	MEMORY
U1	9601*	MC3026
U2	8828/7474	7438
U3	8889/9002	8840/9005
U4	8885	7438
U5	8H90	8840/9005
U6	8889/9002	7438
U7	8H90	8840/9005
U8	8H90	7438
U9	MC3026	8840/9005
U10	9009	7438
U11	7438	8840/9005
U12	8889/9002	7438
U13	8885	8840/9005
U14	MC3026	7438
U15	MC3026	8840/9005
U16	8H90	7438
U17	8271	8840/9005
U18	MC3026	8H90
U19	8889/9002	MC3026
U20	8828/7474	75451
U21	8889/9002	8828/7474
U22	8H90	7475
U23	8271	8828/7474
U24	8840/9005	8828/7474
U25	9321	7475
U26	8H90	MC3026
U27	8H90	8828/7474
U28	8271	8828/7474
U29	8271	7475
U30	7488	9009
U31	7488	8828/7474
U32	8271	8828/7474
U33	8271	7475

*Power Monitor Option

Table 5-1. IC Identification List (Continued)

CHIP	CPU	MEMORY
U34	8885	8828/7474
U35	9003	9009
U36	8889/9002	MC3026
U37	8271	8889/9002
U38	8271	7524
U39	8271	8889/9002
U40	8271	7524
U41	8885	MC3026
U42	8271	75451
U43	9008	75451
U44	8840/9005	8H90
U45	8840/9005	8889/9002
U46	9008	7524
U47	9008	75324
U48	8885	8889/9002
U49	7438	7524
U50	9003	75324
U51	9321	75451
U52	9321	75324
U53	8885	75451
U54	8271	75324
U55	8840/9005	8889/9002
U56	3026	7524
U57	8889/9002	75324
U58	8885	8889/9002
U59	8840/9005	7524
U60	9009	75324
U61	9008	75451
U62	9003	75324
U63	9321	75451
U64	8889/9002	8889/9002
U65	8885	7524
U66	MC3061	75324
U67	8H90	8H90
U68	9321	8889/9002
U69	8271	7524
U70	8885	75451
U71	9009	8H90
U72	MC3026	75324
U73	8H90	75324
U74	9003	75324
U75	8889/9002	75324
U76	8828/7474	75324

Table 5-1. IC Identification List (Continued)

CHIP	CPU	MEMORY
U77	8840/9005	75324
U78	MC3061	75324
U79	8885	75324
U80	7486	8H90
U81	9008	
U82	8889/9002	
U83	8H90	
U84	9009	
U85	8885	
U86	8889/9002	
U87	8889/9002	
U88	9008	
U89	9009	
U90	8840/9005	
U91	8885	
U92	8889/9002	
U93	9003	
U94	8H90	
U95	8271	
U96	8885	
U97	9009	
U98	9008	
U99	MC3026	
U100	8885	
U101	8H90	
U102	8271	
U103	8271	
U104	9008	
U105	8271	
U106	8271	
U107	8271	
U108	8271	
U109	8889/9002	
U110	9009	
U111	9003	
U112	8840/9005	
U113	MC3061	
U114	9005	
U115	8271	
U116	9322	
U117	74181	
U118	9002	
U119	74170	

Table 5-1. IC Identification List (Continued)

CHIP	CPU	MEMORY
U120	8264	
U121	9322	
U122	8271	
U123	5501C/3101	
U124	5501C/3101	
U125	8264	

5-8 POWER SUPPLY SPECIFICATION

The Nova 1200 power supply provides all of the power required to operate the Nova 1200 Processor logic and memory circuits. The electrical specification for the power supply voltages are listed in Table 1-2 and are not repeated here. Each side of the ac power line is fused by a 10 amp, 250 volt glass tube fuse (bus type). Each fuse is mounted in individual panel mounted fuseholders, with the two fuseholders mounted just above the convenience outlet in the rear of the power supply. The fuses should be checked first following any interruption of output power. Maintenance or repair beyond this point should be performed at the factory.

5-9 NOVA 1200 INTERNAL INTERCONNECTIONS

As shown in Figure 1-3, the Nova 1200 major assembly components are interconnectors which electrically join the PCB assemblies in each component together in the final assembly.

AC power is connected into the power supply fuses via the power cord. The load side of the fuses are connected to J1 (Reference Drawing #001-000091). P1 connects into J1 and electrically incorporates the Power switch (at the Console) in series with the convenience outlet and the power transformer primary.

The PCB assembly of the Power supply terminates in a 52 pin edge type connector P2, which connects into J2 on the Resistor Board subassembly. (Reference Drawing #001-000087). This subassembly also terminates in 52 pin edge type connector P3, which connects into J3 of the Back Panel (of the Multiple Printed Circuit Board Connector). The RINH signals, power monitor signals, and voltages connected into the Back Panel are routed to the various 15 inch PCB Assemblies e.g., CPU, Memory, I/O boards by the Back Panel etch (Reference Drawing #001-000090 for the Back Panel signal distribution.)

Power and Console signals are interconnected between the Back Panel and the Console by etched edge connectors P4 (of the Back Panel) and receptacle connector J4 mounted

on the Console PCB Assembly. Table 5-2 lists the various signals routed through P4 & J4 and their respective Back Panel terminations. The electrical terminations from connector J4 are shown on the Nova 800 and 1200 Console drawing #001-000089.

Table 5-2 Console/Back Panel Connections

Connector J4/P4 PIN	SIGNAL	Back Panel PIN	Connector J4/P4 PIN	SIGNAL	Back Panel PIN
1	GND	B1	27	+5	B4
2	MEM15	B18	28	MBO15	A41
3	MEM14	B76	29	MEM13	A35
4	MBO13	A37	30	MBO12	A39
5	MEM12	A36	31	MEM11	A51
6	MBO11	B5	32	MEM10	A45
7	MEM9	A53	33	LIGHTS	N/A (Bus to Pwr Supply)
8	MBO9	B9	34	MEM8	A55
9	MBO7	B14	35	MBO6	B16
10	MEM6	B22	36	MEM5	B26
11	MBO5	B32	37	MEM4	B28
12	MBO14	A43	38	MBO3	B43
13	MEM2	B47	39	MEM0	B71
14	MBO1	B77	40	LIGHT	GND
15	MBO2	B44	41	MEM1	B70
16	MBO4	B42	42	MEM7	B24
17	GND	B2	43	MEM3	B68
18	MBO8	B12	44	MBO10	B8
19	Restart Enable	A32	45	STOP	A31
20	RST	A30	46	CONT DATA	A28
21	CON RQ	A27	47	Cont+Istp+Mstp	A25
22	CON INST	A22	48	MSTP	A20
23	PL	A19	49	CARRY	A15
24	ISTP	A17	50	FETCH	A13
25	ION	A16	51	EXEC	A11
26	RUN	A14	52	DEFER	A12

SECTION VI

ILLUSTRATED PARTS LIST

The Illustrated Parts List for the Nova 1200 Central Processor, Nova 1200 4K Memory, and the Basic I/O Control are published under document No. 005-000630-01. For illustrations and descriptions of the parts complement for the equipment mentioned above reference this separate publication.

SECTION VII

REFERENCE DRAWINGS

7-1 INTRODUCTION

All of the Data General electrical reference drawings for the Nova 1200 Central Processor, Nova 1200 4K Memory, and the Basic I/O Control are contained in this section. The actual drawing complement is bound under separate cover, however all of the drawings and wire lists comprising this section are listed in Table 7-1 for reference purposes.

Table 7-1. Reference Drawings

Title	Drawing No.
Nova 1200 Block Diagram & Waveforms	001-000107
Nova 1200 Flow Chart 1	001-000106 (Sheet 1)
Nova 1200 Flow Chart 2	001-000106 (Sheet 2)
Nova 800 & 1200 Console	001-000089
CPU Nova 1200	001-000088 (Sheets 1 thru 4)
4K Memory-MA & MB Register & Control	001-000103 (Sheet 1)
4K Memory-Sense & Inhibit	001-000103 (Sheet 2)
4K Memory-X Drivers	001-000103 (Sheet 3)
4K Memory-Y Drivers	001-000103 (Sheet 4)
Nova 800, 1200 Power Supply	001-000091
Back Panel Nova 1200	001-000090
Nova 800 or 1200 Rack Installation	010-000006
Power Monitor*	001-000110

*Option drawings are supplied only with equipment which includes that particular option.

Table 7-1. Reference Drawings (Continued)

Title	Drawing No.
Program Load Nova 1200*	001-000109
Resistor Board	001-000087
I/O Bus Receivers & Common Select	001-000070
Teletype Control*	001-000071
Paper Tape Reader Control 4011*	001-000072
Paper Tape Punch Control 4012*	001-000073
Real Time Clock*	001-000074
I/O External Cable	008-000044
Nova 800/1200 Internal I/O Cable Wire List	008-000053
Nova 800/1200 Hi Speed Reader* Internal Cable Wire List	008-000054
Nova 800/1200 Punch* Internal Cable Wire List	008-000055

*Option drawings are supplied only with equipment which includes that particular option.

APPENDICES

APPENDIX A

LOGIC DIAGRAMS

AND

TRUTH TABLES

FOR

NOVA 1200 INTEGRATED

CIRCUIT PACKAGES

INTRODUCTION

This Appendix is a compilation of the Logic Diagrams and Truth Tables for the Integrated Circuit (IC) Packages used in the Nova 1200 logic. Information presented in this section is intended to supplement the electrical (assembly logic diagrams) drawings and the Integrated Circuits listed in Section V of this manual. The data presented herein is cataloged first alphabetically then by number, where the alphabetical prefixes of the number are germane to the manufacturer's identification of the part rather than defining operational parameters (e.g., temperature, case construction, etc.). Table A-1 is an index listing the types of IC's cataloged in this section with the corresponding page number location. All of the logical elements listed use positive logic, i.e., the highest voltage equals a logic 1.

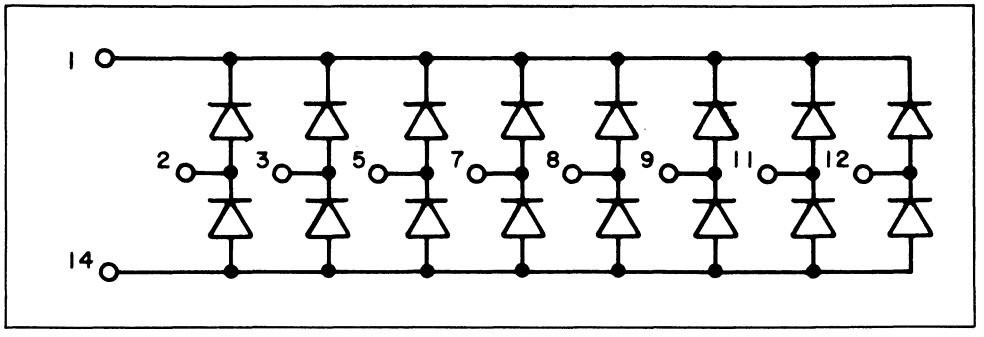
Table A-1. IC INDEX

Table A-1. IC INDEX (Continued)

BC728

16 Diode Array

LOGIC DIAGRAM



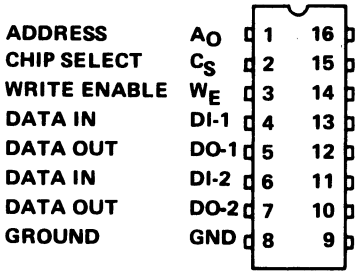
TRUTH TABLE N/A For BC728

IM5501

Bipolar Random Access Memory

LOGIC DIAGRAM/PIN DESIGNATIONS

PIN CONFIGURATION



PIN 1 DESIGNATED EITHER BY DOT OR NOTCH

The IM5501 is a High Speed Fully decoded Bipolar 64 Bit Random Access Memory organized as 16 x 4 Bit words. The READ and WRITE Controls for the IM5501 are as follows:

READ: The memory is addressed through A_0-A_3 which select one of the 16 words. The chip is enabled by placing chip select, (C_S) to logic "0". If the write enable (W_E) is at a logic "1" the four stored bits are read out of D_{01}-D_{04} in parallel.

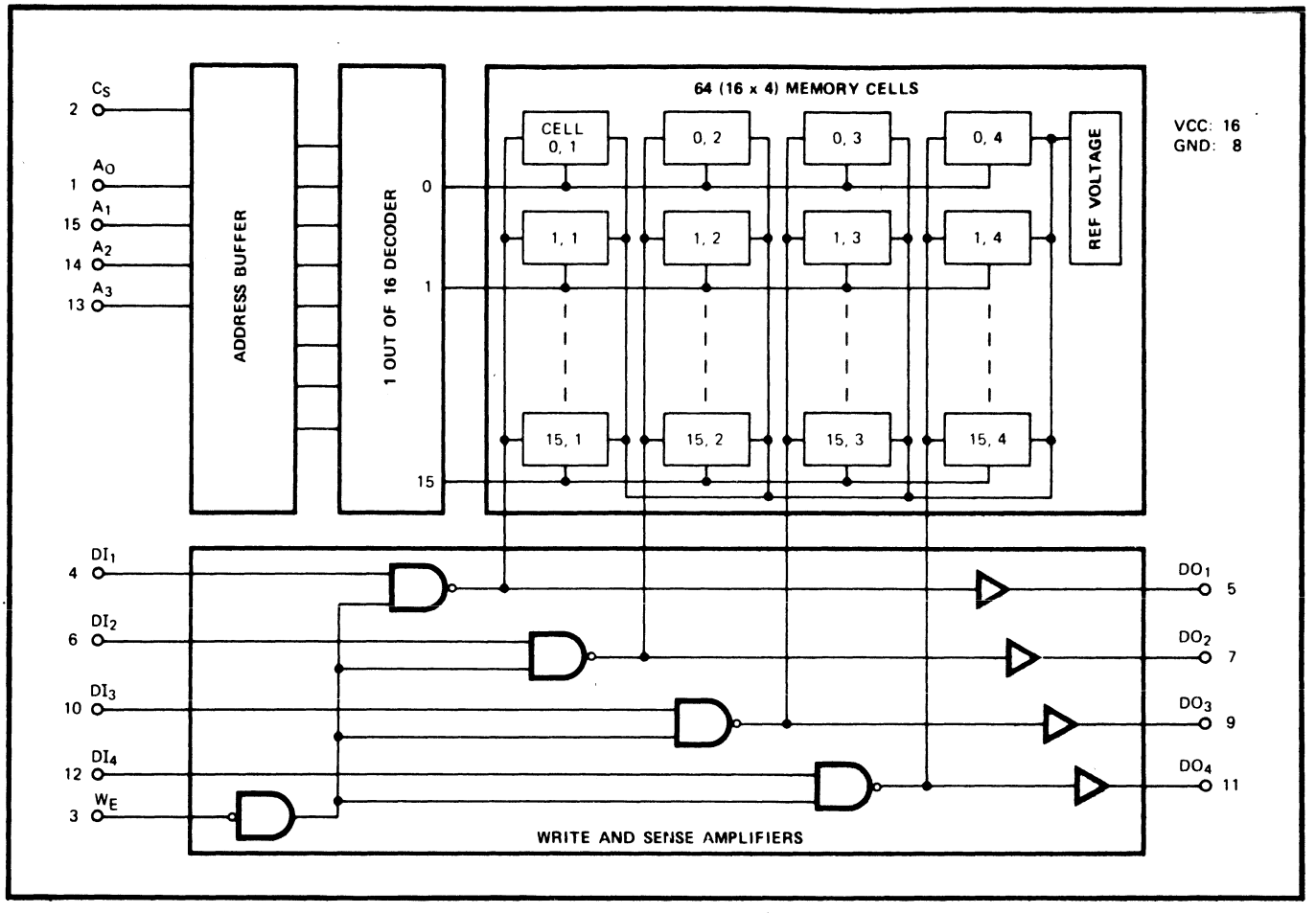
WRITE: The memory is addressed through A_0-A_3 which select one of the 16 words. The chip is enabled by placing C_S to logic "0". If the W_E is at a logic"0", the data on terminals D_{11}-D_{14} is written into the addressed work in parallel and in complementary form.

When W_E returns to logic "1", the information that was written in is now read out. However, each bit readout is the complement of what was written in.

IM5501 (cont.)

Bipolar Random Access Memory

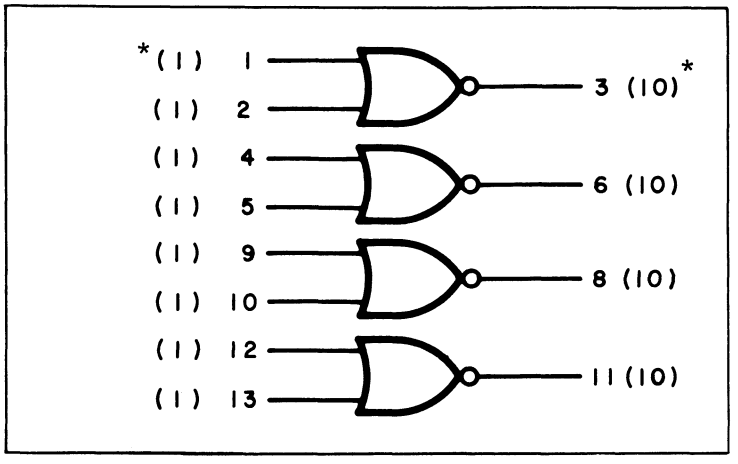
IM5501 Functional Logic Diagram



MC3002

Quad 2 - Input NOR Gate

LOGIC DIAGRAM



*Loading Max. Shown in Parenthesis

t_{pd} = 6.0 ns typ

P_D = 122 mW typ/pkg

TRUTH TABLE

$3 = \overline{1 + 2}$

$6 = \overline{4 + 5}$

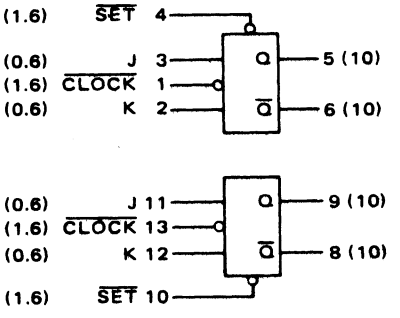
$8 = \overline{9 + 10}$

$11 = \overline{12 + 13}$

V_{CC} = Pin 14, GND = Pin 7

MC3062
Dual J-K Flip-Flop

J	K	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

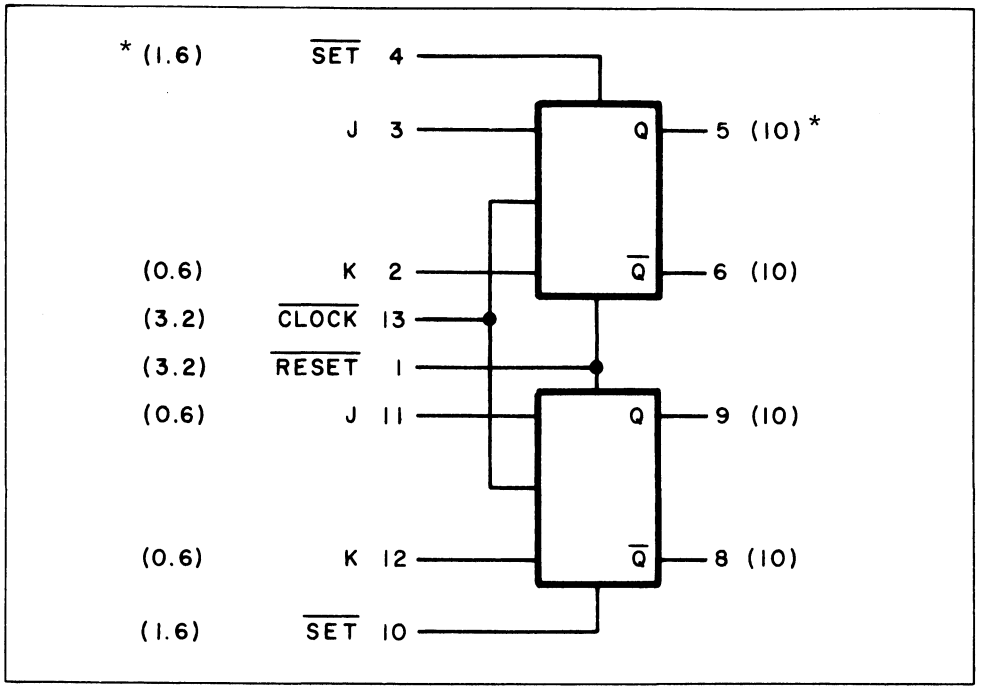


f = 50 MHz
P_D = 100 mW typ/pkg

MC3061

Dual J-K Flip-Flop

LOGIC DIAGRAM/PIN DESIGNATIONS



*Loading Max. Shown in Parenthesis

t = 50 MHz

P_D = 100 mW typ/pkg

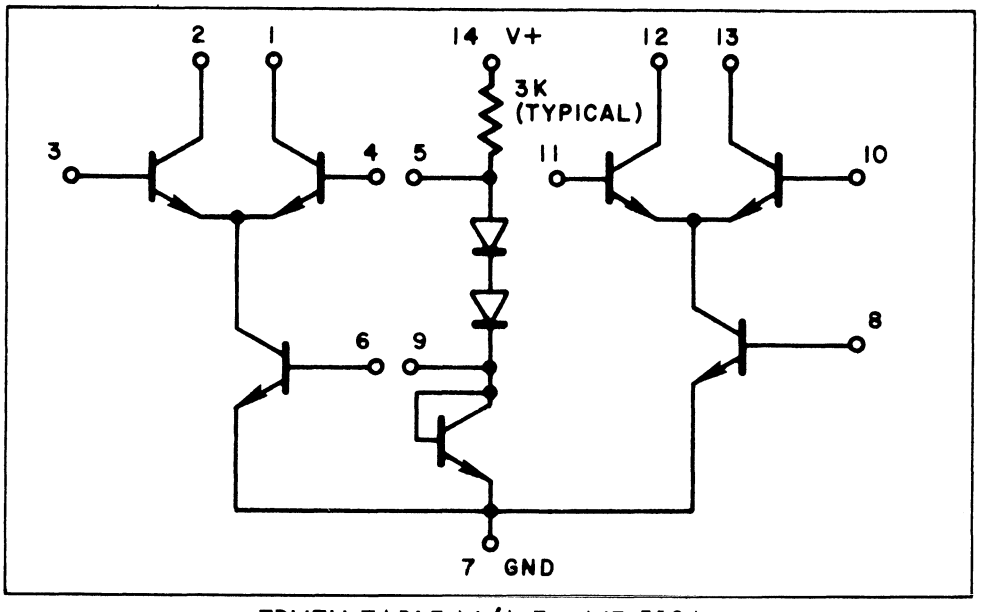
TRUTH TABLE

J	K	Q_N	Q_{N+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

V_{CC} = Pin 14, GND = Pin 7

NE510A
Amplifier

BASIC CIRCUIT SCHEMATIC



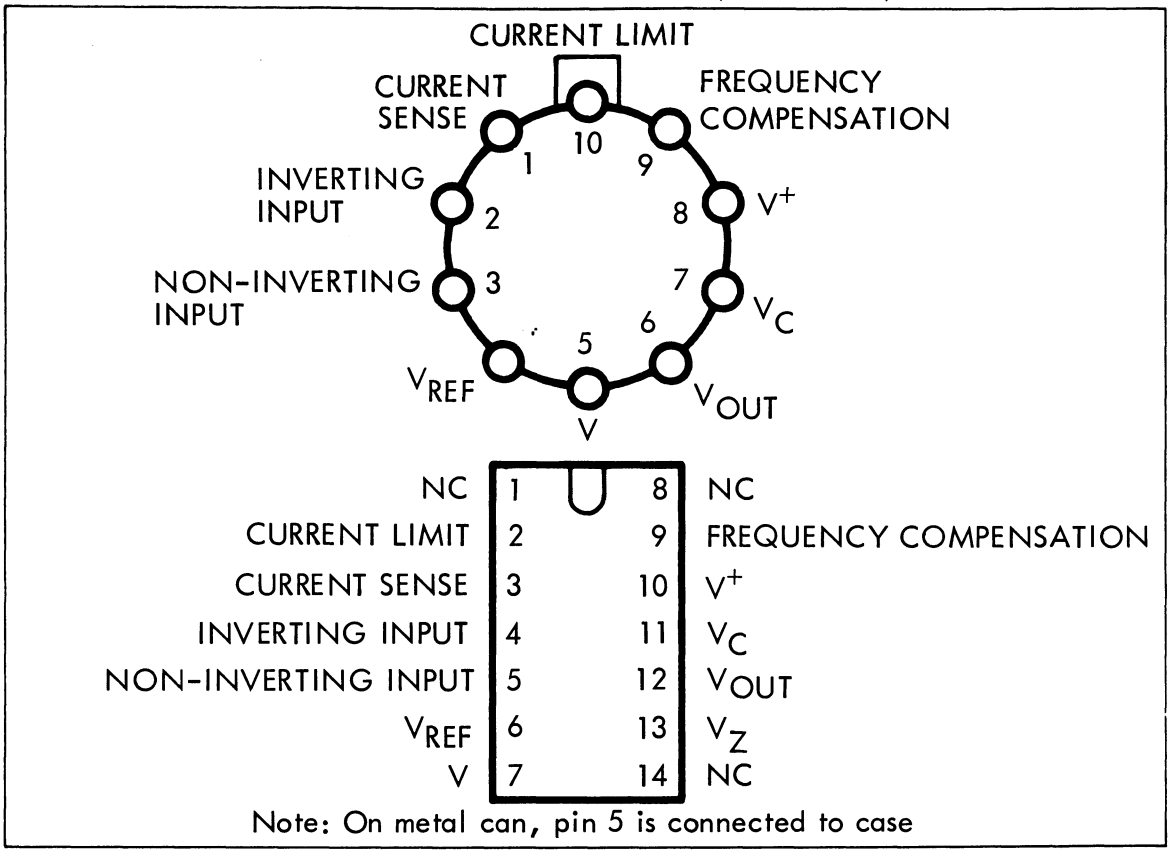
TRUTH TABLE N/A For NE 510A

NOTE: NE510 may be connected as either a high-gain, common-emitter, common-base, cascode amplifier or a common-collector, common-base, differential amplifier that is useful in critical limiter applications. Automatic gain control may be applied to either circuit.

μA723C

Precision Voltage Regulator

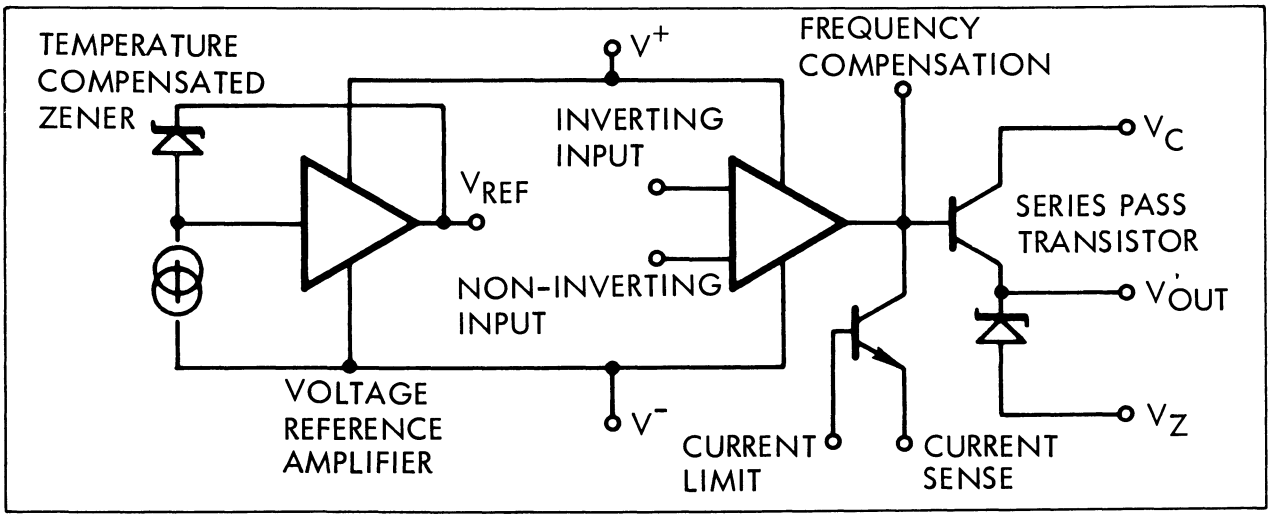
CONNECTIONS DIAGRAMS (TOP VIEWS)



NC	1	8	NC
CURRENT LIMIT	2	9	FREQUENCY COMPENSATION
CURRENT SENSE	3	10	V^+
INVERTING INPUT	4	11	V_C
NON-INVERTING INPUT	5	12	V_{OUT}
V_{REF}	6	13	V_Z
V	7	14	NC

Note: On metal can, pin 5 is connected to case

EQUIVALENT CIRCUIT

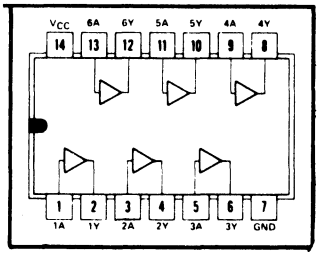


TRUTH TABLE N/A For μA723C

7407

HEX BUFFERS/DRIVERS WITH OPEN-COLLECTOR HIGH-VOLTAGE OUTPUTS

J OR N
DUAL-IN-LINE PACKAGE
(TOP VIEW)



positive logic: $Y = \overline{A}$

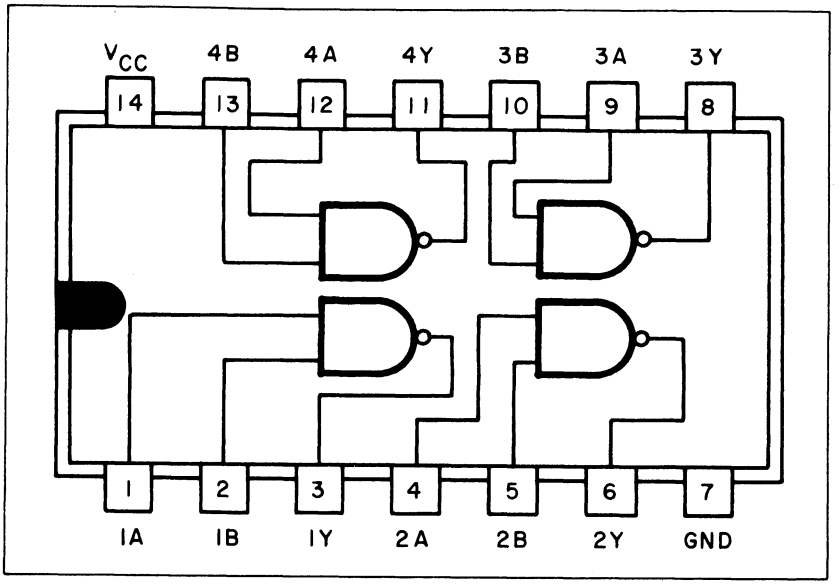
These monolithic TTL hex buffers/drivers feature high-voltage open-collector outputs for interfacing with high-level circuits (such as MOS), or for driving high-current loads (such as lamps or relays), and are also characterized for use as buffers for driving TTL inputs. For increased fan-out, several buffers in a single package may be paralleled. The SN5407 and SN7407 have minimum breakdown voltages of 30 volts and the SN5417 and SN7417 have minimum breakdown voltages of 15 volts. The maximum sink current is 30 milliamperes for the SN5407 and SN5417, and 40 milliamperes for the SN7407 and SN7417.

These circuits are completely compatible with most TTL and DTL families. Inputs are diode-clamped to minimize transmission-line effects which simplifies design. Typical power dissipation is 145 milliwatts and average propagation delay time is 14 nanoseconds.

7438

Quadruple 2 - Input Positive NAND Gates
(With Open - Collector Outputs)

LOGIC DIAGRAM/PIN DESIGNATIONS



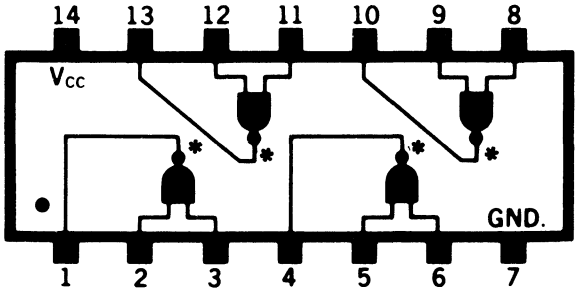
TRUTH TABLE

(positive logic) $Y = \overline{AB}$

7439 / 8881

Quad 2 - Input NAND Gate

LOGIC DIAGRAM



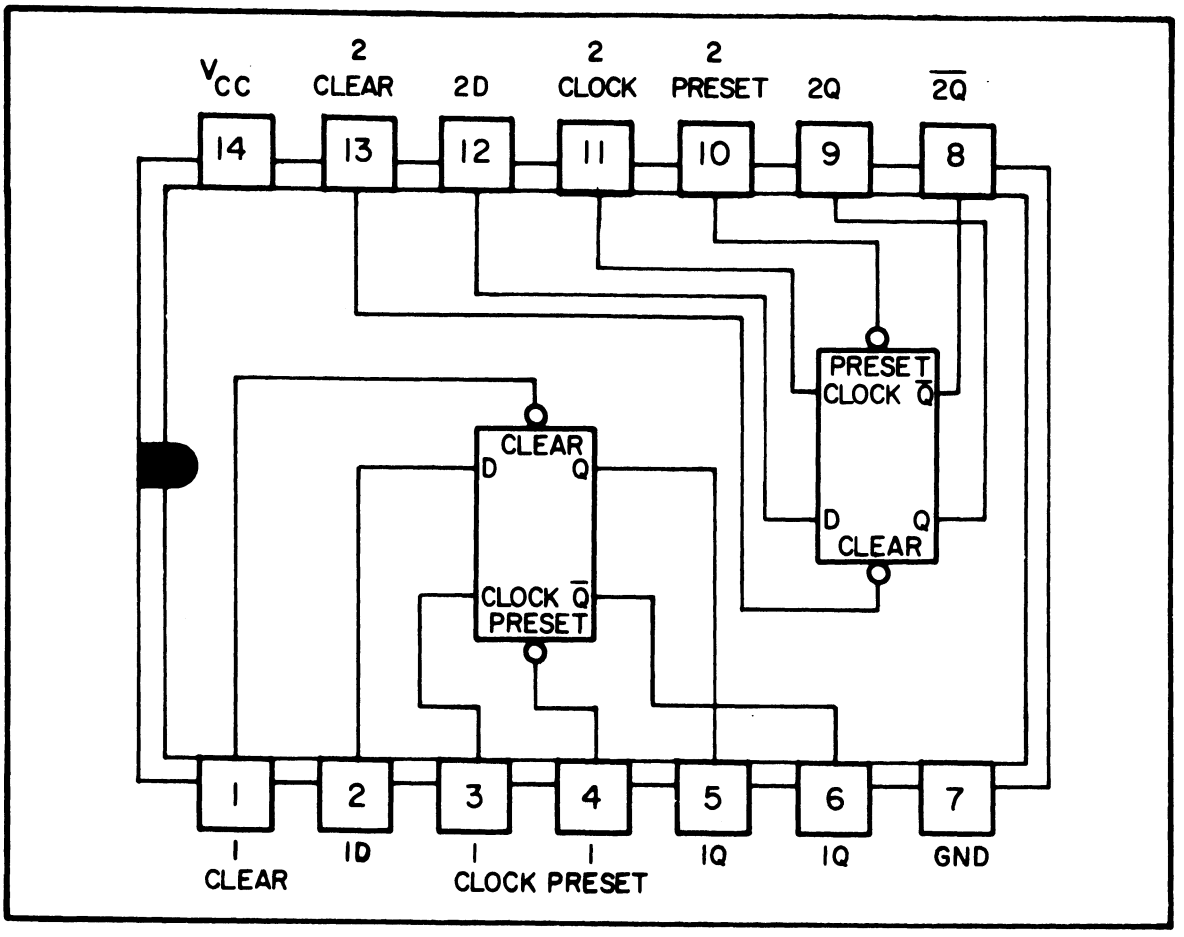
TRUTH TABLE

V_{IN}	V_{IN}	V_{OUT}
L	L	H
L	H	H
H	L	H
H	H	L

7474

Dual D-Type edge-triggered flip-flop

LOGIC DIAGRAM/PIN DESIGNATIONS



Propogation delay - 24 nsec

Power dissipation - 84 mW total for two flip-flops (42 mW per flip-flop)

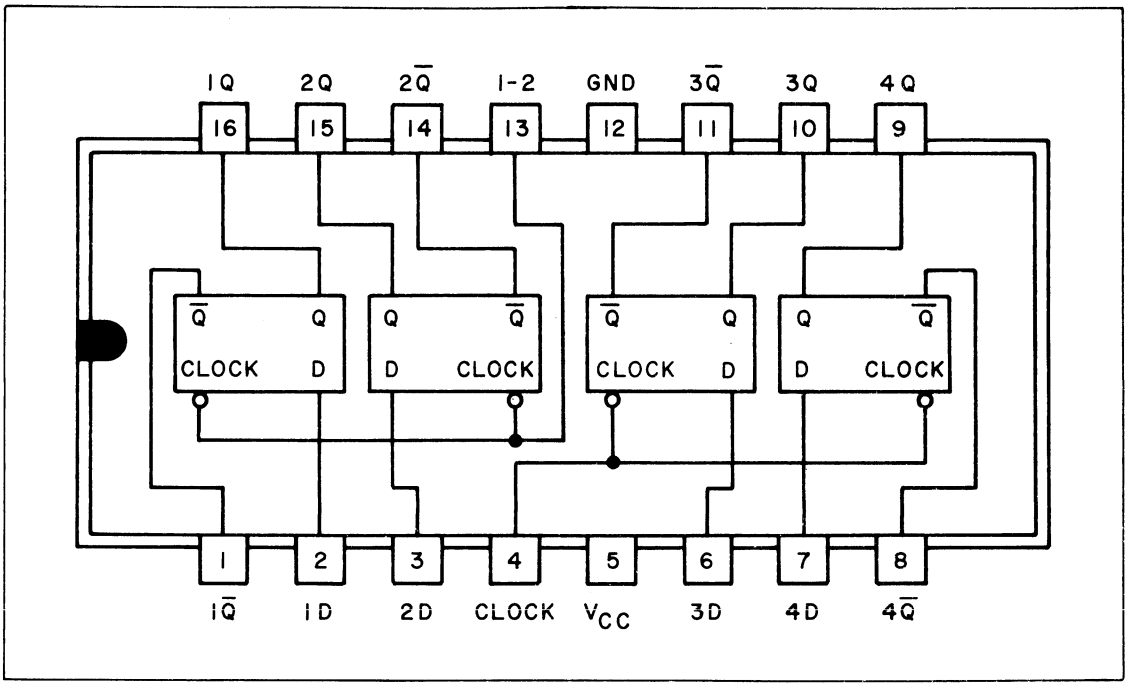
TRUTH TABLE

t_n	$t_n + 1$			
D	Q	$\overline{Q}$	PRESET	CLEAR
0 1	0 1	1 0		
	1	0	0	
	0	1		0

7475

4 - Bit Bistable Latch

LOGIC DIAGRAM/PIN DESIGNATIONS



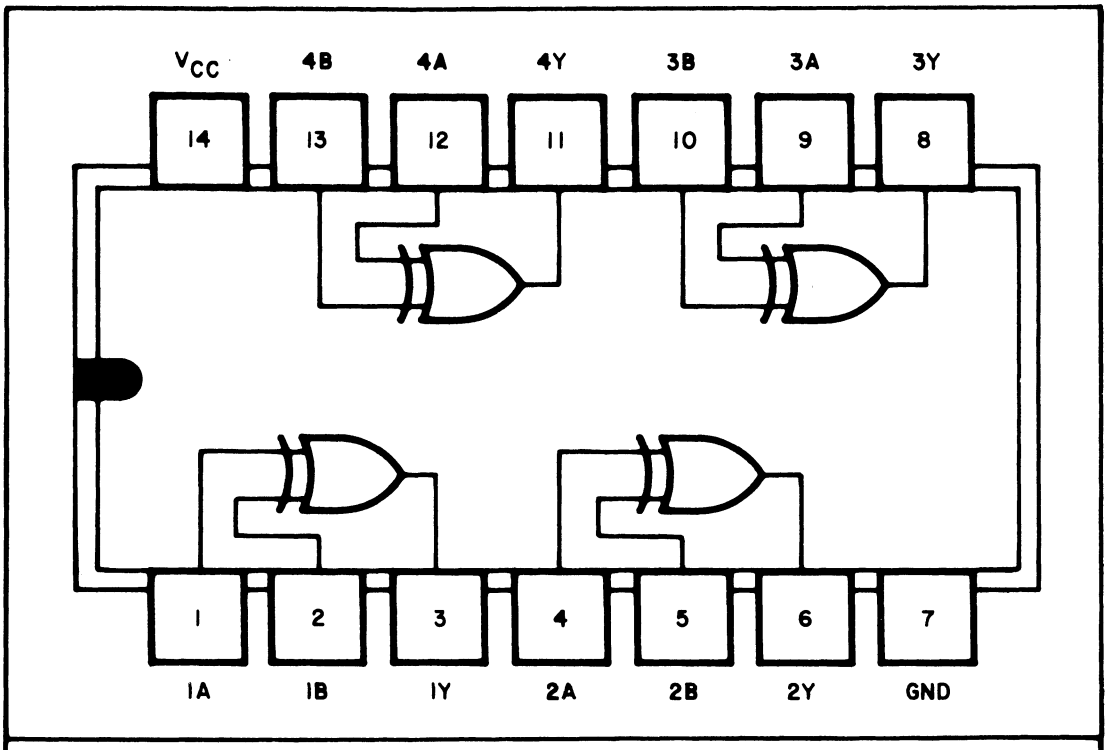
TRUTH TABLE
(EACH LATCH)

t_n	t_{n+1}
D	Q
1	1
0	0

7486

Quadruple 2-Input Exclusive-OR Gates

LOGIC DIAGRAM/PIN DESIGNATIONS



positive logic: $Y = A \oplus B$

TRUTH TABLE

INPUTS		OUTPUTS
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

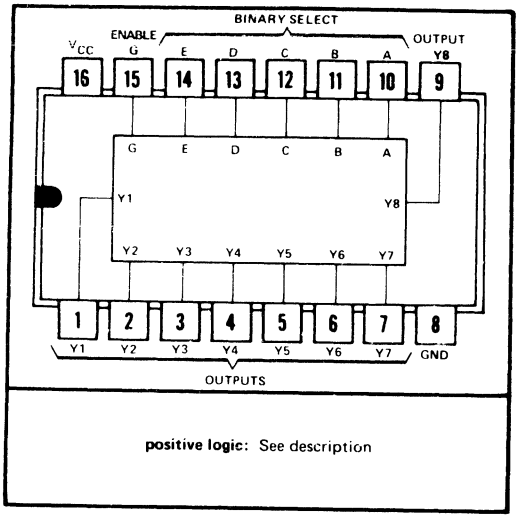
Each of these monolithic, quadruple 2-input exclusive-OR gates utilize TTL circuitry to perform the function: $Y = AB + AB$. When the input states are complementary, the output goes to a logical 1.

A full fan-out to 10 normalized series 54/74 loads is available from each of the outputs in the logical 0 state. A fan-out of 20 is provided in the logical 1 state to facilitate connection of unused inputs to used inputs. Propagation delay is 12 nanoseconds and power dissipation is 37.5 milliwatts typically for each exclusive-OR function.

7488

256-Bit Read-Only Memory

LOGIC DIAGRAM/PIN DESIGNATIONS

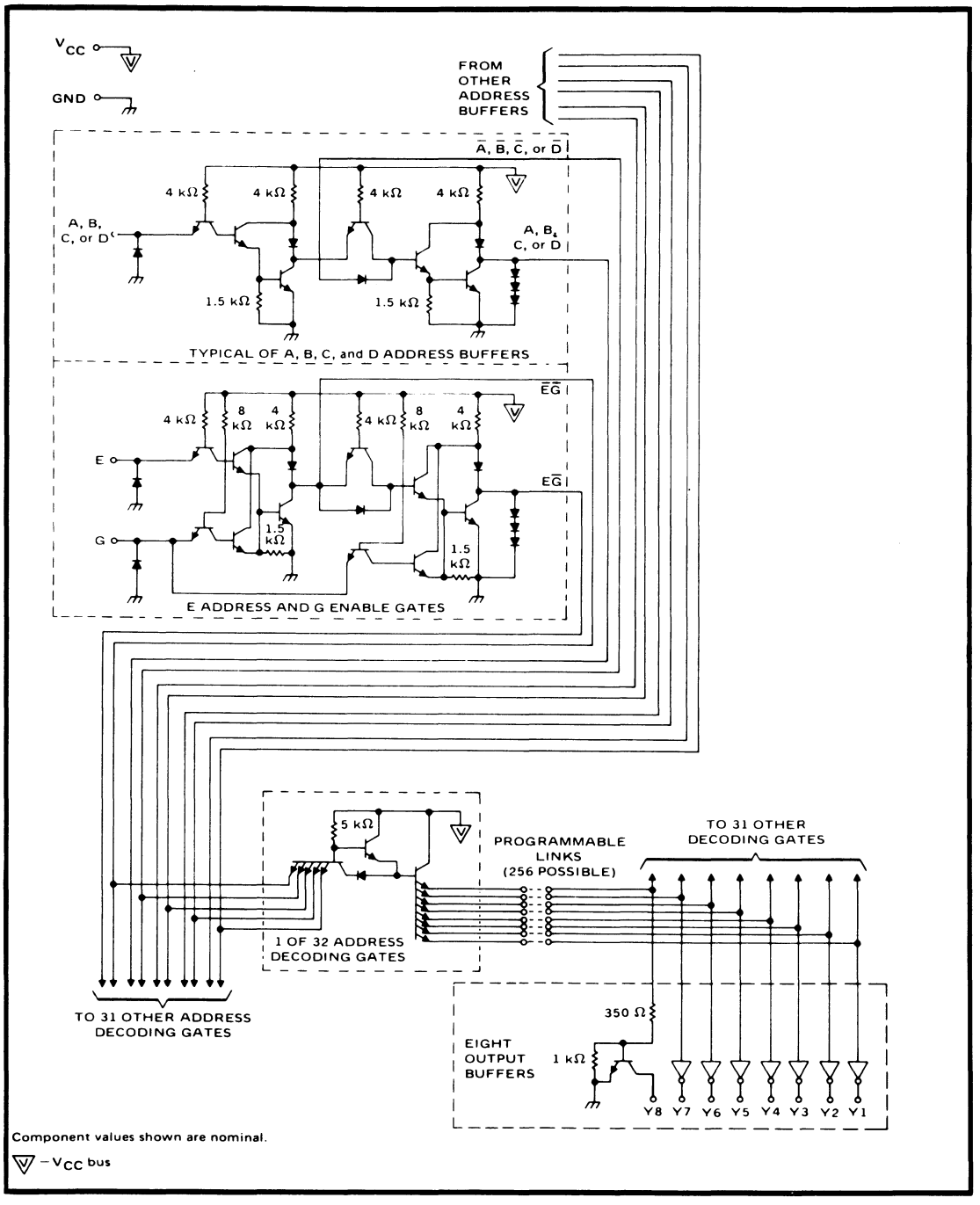


The SN7488 circuit is a custom-programmed, 256-bit, read-only memory organized as 32 words of eight bits each. This monolithic, high-speed, transistor-transistor logic (TTL), 32-word memory array is addressed in stright 5-bit binary with full on-chip decoding. An overriding memory-enable input is provided which, when taken high, will inhibit the 32 address gates and cause all eight outputs to remain high. Data, as specified by the customer are permanently programmed into the monolithic structure for the 256 bit locations. The address of an eight-bit word is accomplished through the buffered, binary select inputs which are decoded by the 32 five-input address gates. When the memory-enable input is high, all 32 gate outputs are low, turning off the eight output buffers.

7488 (cont.)

256-Bit Read-Only Memory

SIMPLIFIED SCHEMATIC DIAGRAM



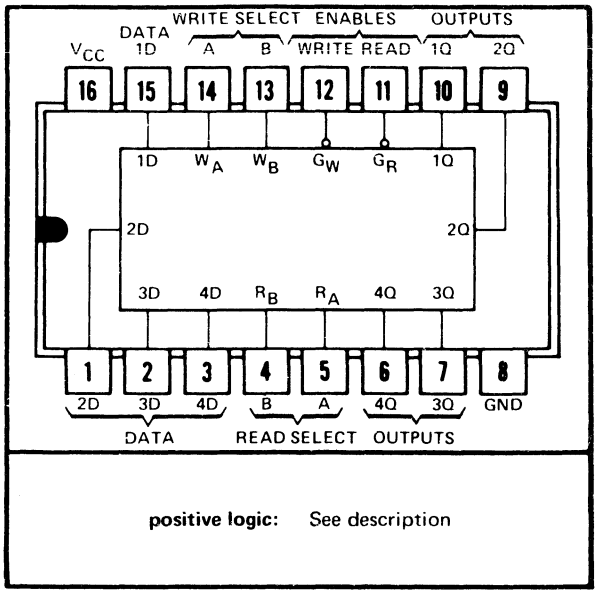
Component values shown are nominal.

▽ – V_{CC} bus

74170

4-By-4 Register Files

LOGIC DIAGRAM



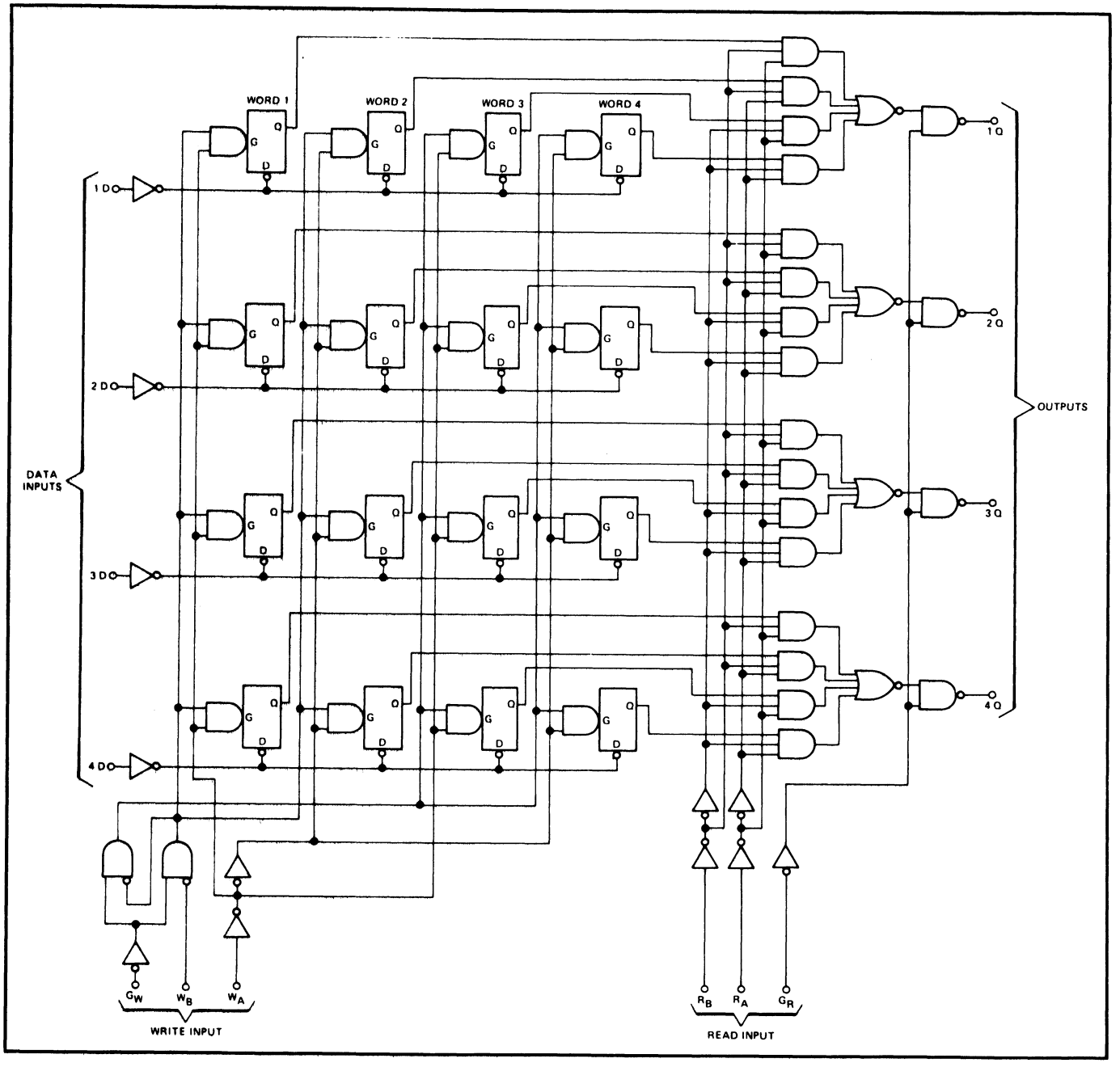
positive logic: See description

Four data inputs are available which are used to supply the 4-bit word to be stored. Location of the word is determined by the write address inputs A and B in conjunction with a write-enable signal. Data applied at the inputs should be in its true form. That is, if a high-level signal is desired from the output, a high-level is applied at the data input for that particular bit location. The latch inputs are arranged so that new data will be accepted only if both internal address (T) gate inputs are high. When this condition exists, data at the D input is transferred to the latch output. The individual address lines permit direct acquisition of data stored in any four of the latches. Four individual decoding gates are used to complete the address for teading a word. When the read address is made in conjunction with the read-enable signal, the word appears at the four outputs.

74170 (cont.)

4-By-4 Register Files

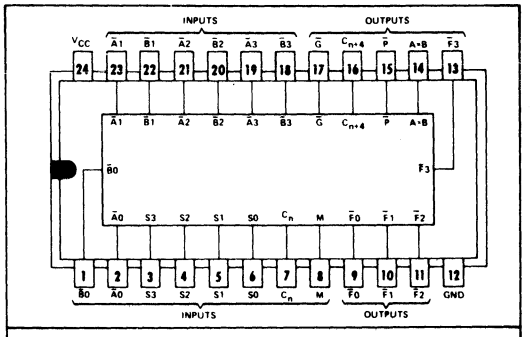
FUNCTIONAL LOGIC DIAGRAM



74181

Arithmetic Logic Units/Function Generator

PIN DESIGNATIONS



logic: see function tables

LOGIC DIAGRAM

DESIGNATION	PIN NOS.	FUNCTION
$\overline{A}3, \overline{A}2, \overline{A}1, \overline{A}0$	19, 21, 23, 2	WORD A INPUTS
$\overline{B}3, \overline{B}2, \overline{B}1, \overline{B}0$	18, 20, 22, 1	WORD B INPUTS
S3, S2, S1, S0	3, 4, 5, 6	FUNCTION-SELECT INPUTS
C_n	7	CARRY INPUT
M	8	MODE CONTROL INPUT
$\overline{F}3, \overline{F}2, \overline{F}1, \overline{F}0$	13, 11, 10, 9	FUNCTION OUTPUTS
A = B	14	COMPARATOR OUTPUT
$\overline{P}$	15	CARRY PROPAGATE OUTPUT
C_{n+4}	16	CARRY OUTPUT
$\overline{G}$	17	CARRY GENERATE OUTPUT
V_{CC}	24	SUPPLY VOLTAGE
GND	12	GROUND

TRUTH TABLES

ARITHMETIC OPERATIONS

FUNCTION SELECT				OUTPUT FUNCTION	
S3	S2	S1	S0	LOW LEVELS ACTIVE	HIGH LEVELS ACTIVE
L	L	L	L	F = A minus 1	F = A
L	L	L	H	F = AB minus 1	F = A+B
L	L	H	L	F = A$\overline{B}$ minus 1	F = A+$\overline{B}$
L	L	H	H	F = minus 1 (2's complement)	F = minus 1 (2's complement)
L	H	L	L	F = A plus [A+$\overline{B}$]	F = A plus A$\overline{B}$
L	H	L	H	F = AB plus [A+$\overline{B}$]	F = [A+B] plus A$\overline{B}$
L	H	H	L	F = A minus B minus 1	F = A minus B minus 1
L	H	H	H	F = A+$\overline{B}$	F = A$\overline{B}$ minus 1
H	L	L	L	F = A plus [A+B]	F = A plus AB
H	L	L	H	F = A plus B	F = A plus B
H	L	H	L	F = A$\overline{B}$ plus [A+B]	F = [A+$\overline{B}$] plus AB
H	L	H	H	F = A+B	F = AB minus 1
H	H	L	L	F = A plus A†	F = A plus A†
H	H	L	H	F = AB plus A	F = [A+B] plus A
H	H	H	L	F = A$\overline{B}$ plus A	F = [A+$\overline{B}$] plus A
H	H	H	H	F = A	F = A minus 1

With mode control (M) and C_n low
†Each bit is shifted to the next more significant position.

LOGIC FUNCTIONS

FUNCTION SELECT				OUTPUT FUNCTION	
S3	S2	S1	S0	NEGATIVE LOGIC	POSITIVE LOGIC
L	L	L	L	F = $\overline{A}$	F = $\overline{A}$
L	L	L	H	F = $\overline{A\overline{B}}$	F = $\overline{A+\overline{B}}$
L	L	H	L	F = $\overline{A}$+B	F = $\overline{A}$B
L	L	H	H	F = Logical 1	F = Logical 0
L	H	L	L	F = $\overline{A+\overline{B}}$	F = $\overline{A\overline{B}}$
L	H	L	H	F = $\overline{B}$	F = $\overline{B}$
L	H	H	L	F = $\overline{A \oplus B}$	F = A $\oplus$ B
L	H	H	H	F = A+$\overline{B}$	F = A$\overline{B}$
H	L	L	L	F = $\overline{A}$B	F = $\overline{A}$+B
H	L	L	H	F = A $\oplus$ B	F = $\overline{A \oplus B}$
H	L	H	L	F = B	F = B
H	L	H	H	F = A+B	F = AB
H	H	L	L	F = Logical 0	F = Logical 1
H	H	L	H	F = A$\overline{B}$	F = A+$\overline{B}$
H	H	H	L	F = AB	F = A+B
H	H	H	H	F = A	F = A

With mode control (M) high: C_n irrelevant
For positive logic: logical 1 = high voltage
logical 0 = low voltage
For negative logic: logical 1 = low voltage
logical 0 = high voltage

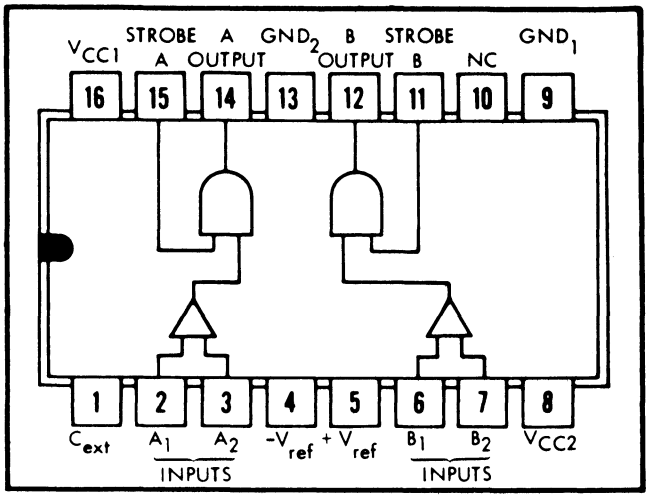
The 74181 performs 16 binary arithmetic operations on two 4-bit words as shown in the function table. These words as shown in the function

The 74181 performs 16 binary arithmetic operations on two 4-bit words as shown in the function table. These operations are selected by the four function-select lines (S0, S1, S2, S3) and include addition, subtraction, decrement, and straight transfer. When performing arithmetic operations, the internal carries must be enabled by applying a low level to the MODE Control (M) input.

7524

Dual Sense Amplifiers

LOGIC DIAGRAM/PIN DESIGNATIONS



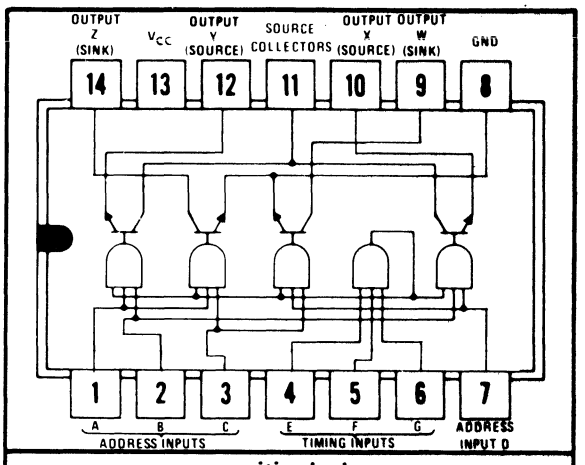
TRUTH TABLE

$IN_A \cdot$ STROBE A = OUT A
$\overline{IN_A} \cdot$ STROBE A = $\overline{OUT\ A}$
$IN_B \cdot$ STROBE B = OUT B
$\overline{IN_B} \cdot$ STROBE B = $\overline{OUT\ B}$

75324

Memory Driver with Decode Inputs (400 M A)

LOGIC DIAGRAM / PIN DESIGNATIONS



positive logic

see truth table

TRUTH TABLE

INPUTS							OUTPUTS			
ADDRESS				TIMING			SINK	SOURCES		SINK
A	B	C	D	E	F	G	W	X	Y	Z
0	0	1	1	1	1	1	ON	OFF	OFF	OFF
0	1	0	1	1	1	1	OFF	ON	OFF	OFF
1	1	0	0	1	1	1	OFF	OFF	ON	OFF
1	0	1	0	1	1	1	OFF	OFF	OFF	ON
X	X	X	X	0	X	X	OFF	OFF	OFF	OFF
X	X	X	X	X	0	X	OFF	OFF	OFF	OFF
X	X	X	X	X	X	0	OFF	OFF	OFF	OFF

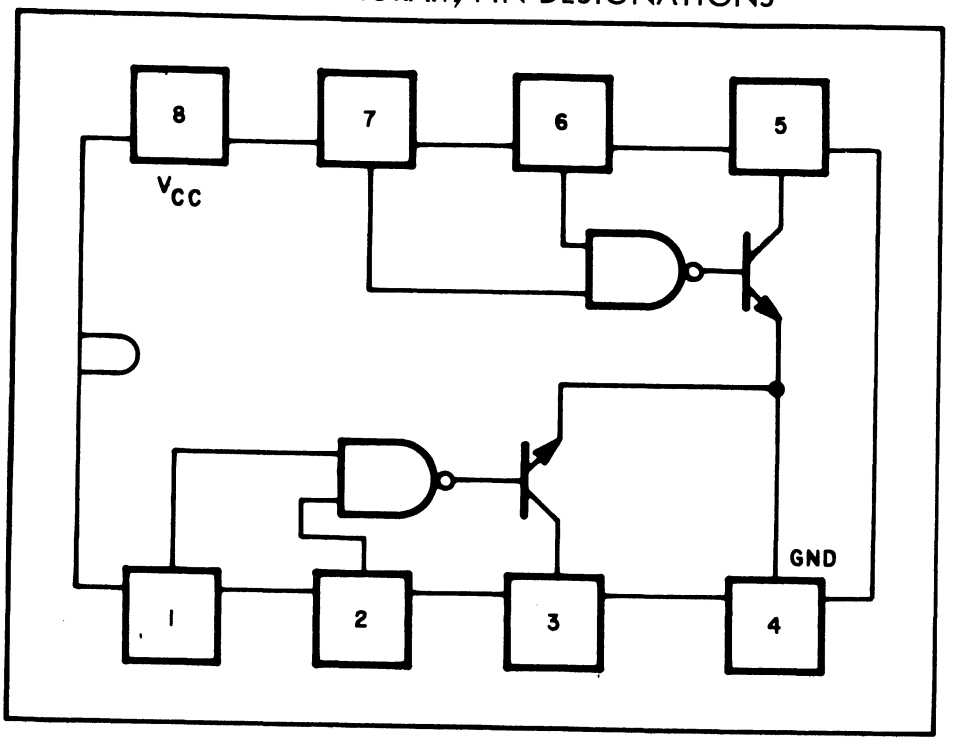
NOTES: 1. X = Logical 1 or logical 0.
2. Not more than one output is to be allowed to be ON at one time: When all timing inputs are at a logical 1, two of the address inputs must be at a logical 0.

The SN75 324 is a monolithic memory driver with decode inputs designed for use with magnetic memories. The device contains two 400-milliampere (source/sink) switch pairs, with decoding capability from four address lines. Two address inputs (B and C) are used for mode selection, i.e., source or sink. The other two address inputs (A and D) are used for switch-pair selection, i.e., output switch-pair Y/Z or W/X respectively.

75451

Interface Circuit

LOGIC DIAGRAM/PIN DESIGNATIONS



TRUTH TABLE

$3 = 1 \cdot 2$
$5 = 6 \cdot 7$

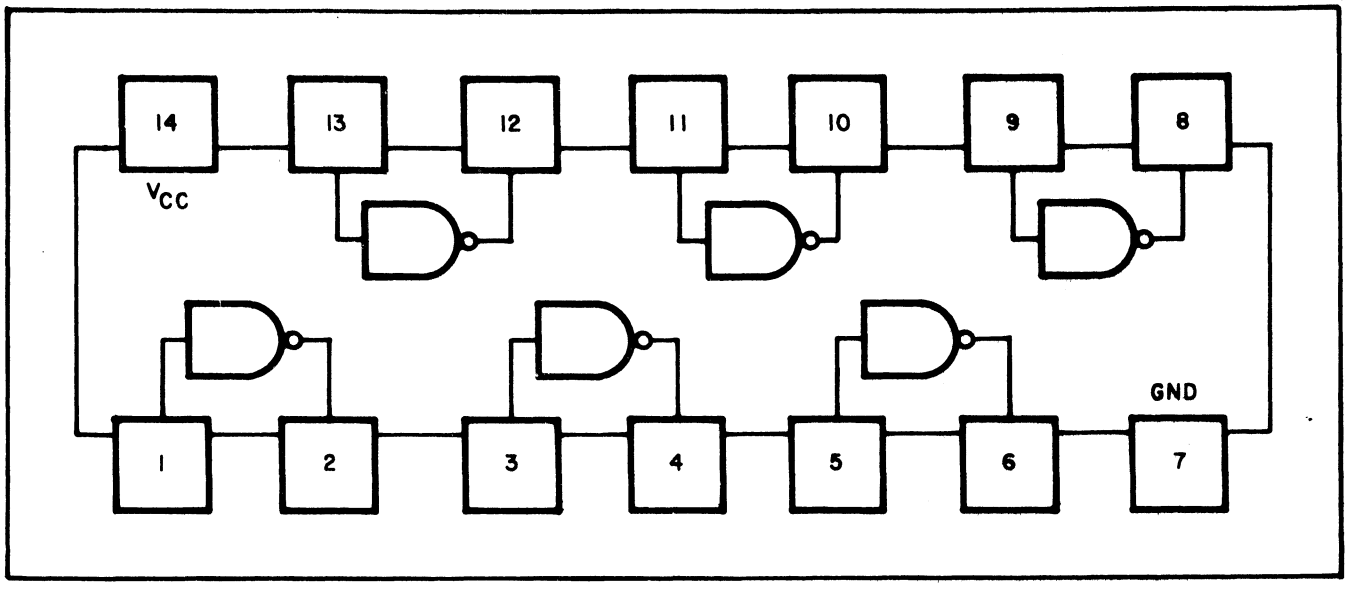
SN75 451 interface circuit - typical characteristics

Gate input current and voltage	TTL
Gate output current and voltage	TTL
Transistor collector-emitter voltage (I_c = 0.1 mA)	35V
Transistor collector substrate breakdown voltage (I_{cs} = 0.1 mA)	50V
Transistor saturated collector-emitter forward voltage (I_c = 0.1 mA)	0.3V
(I_c = 300 mA)	0.5V
Overall turn-on delay	16ns
Overall turn-off delay	17ns

8H90

Hex Inverter

LOGIC DIAGRAM/PIN DESIGNATIONS



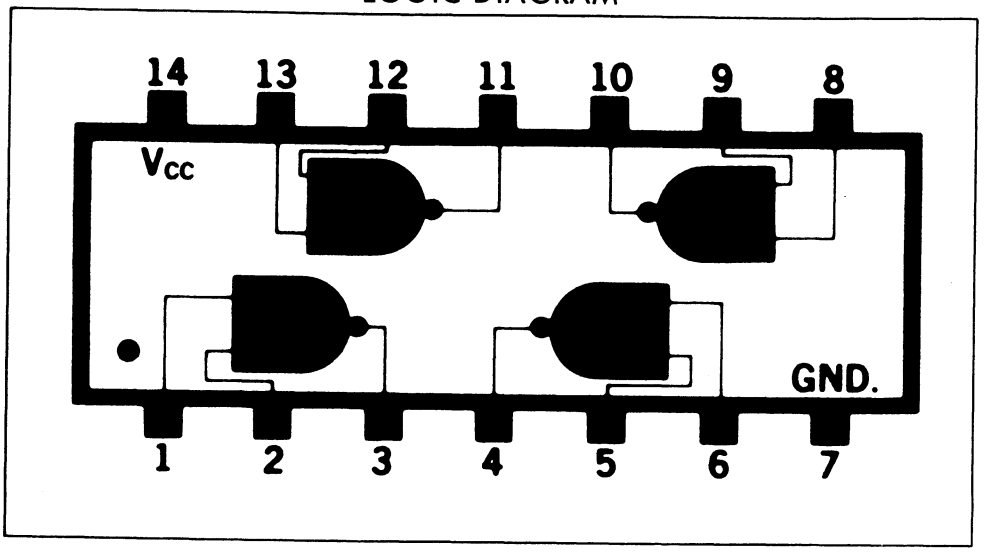
TRUTH TABLE

ANY INPUT LOW = HIGH OUT
ANY INPUT HIGH = LOW OUT

8T80

Quad 2 - Input NAND Interface Gate

LOGIC DIAGRAM

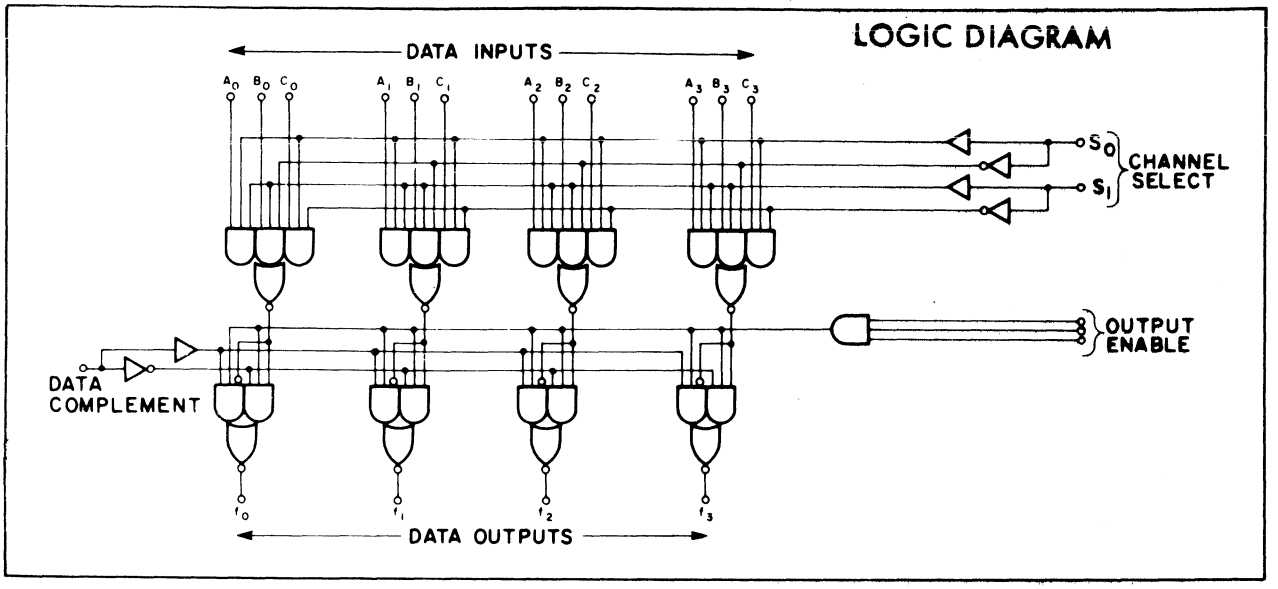


TRUTH TABLE

V_{IN}	V_{IN}	V_{OUT}
L	L	H
L	H	H
H	L	H
H	H	L

8264

3 - Input, 4-Bit Digital Multiplier

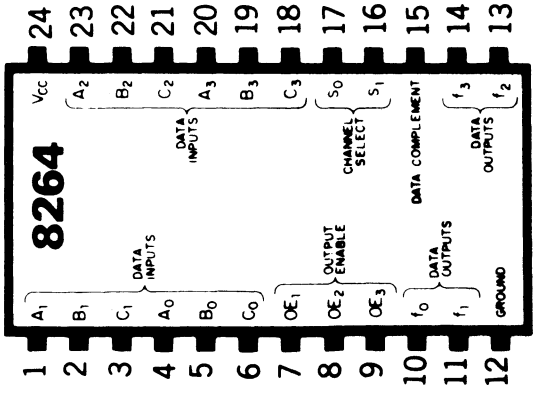


TRUTH TABLE

DATA INPUT A_n	B_n	C_n	CHANNEL SELECT S_0	S_1	DATA COMPLEMENT	OUTPUT ENABLE (8264)	DATA OUTPUTS
A_n	x	x	1	1	0	1	A_n
x	B_n	x	0	1	0	1	B_n
x	x	C_n	1	0	0	1	C_n
x	x	x	0	0	0	1	$\overline{0}$
A_n	x	x	1	1	1	1	$\overline{A_n}$
x	B_n	x	0	1	1	1	$\overline{B_n}$
x	x	C_n	1	0	1	1	$\overline{C_n}$
x	x	x	0	0	1	1	1
x	x	x	x	x	x	0	1

x = Either State

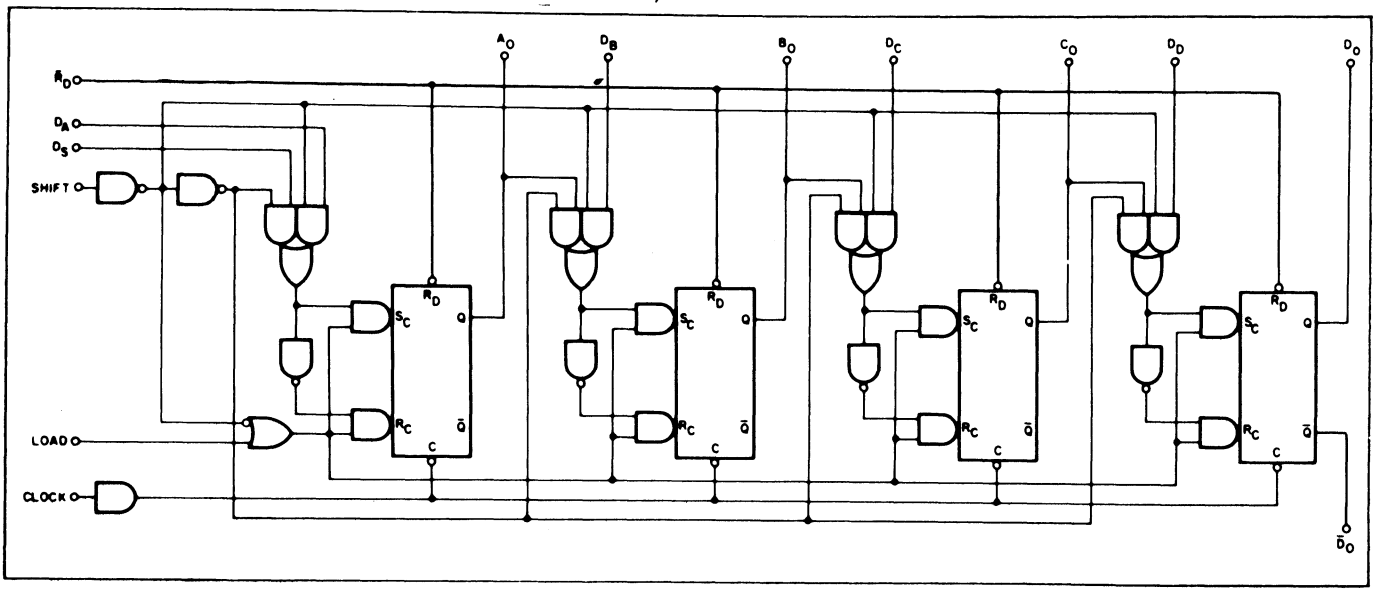
P, Y PACKAGE



8271

4 - Bit Shift Registers

LOGIC DIAGRAM/PIN DESIGNATIONS

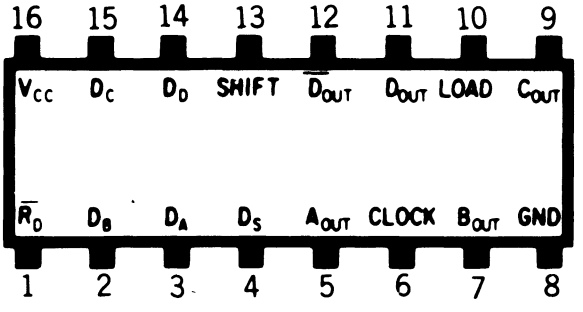


TRUTH TABLE

CONTROL STATE	LOAD	SHIFT
Hold	0	0
Parallel Entry	1	0
Shift Right	0	1
Shift Right	1	1

B PACKAGE

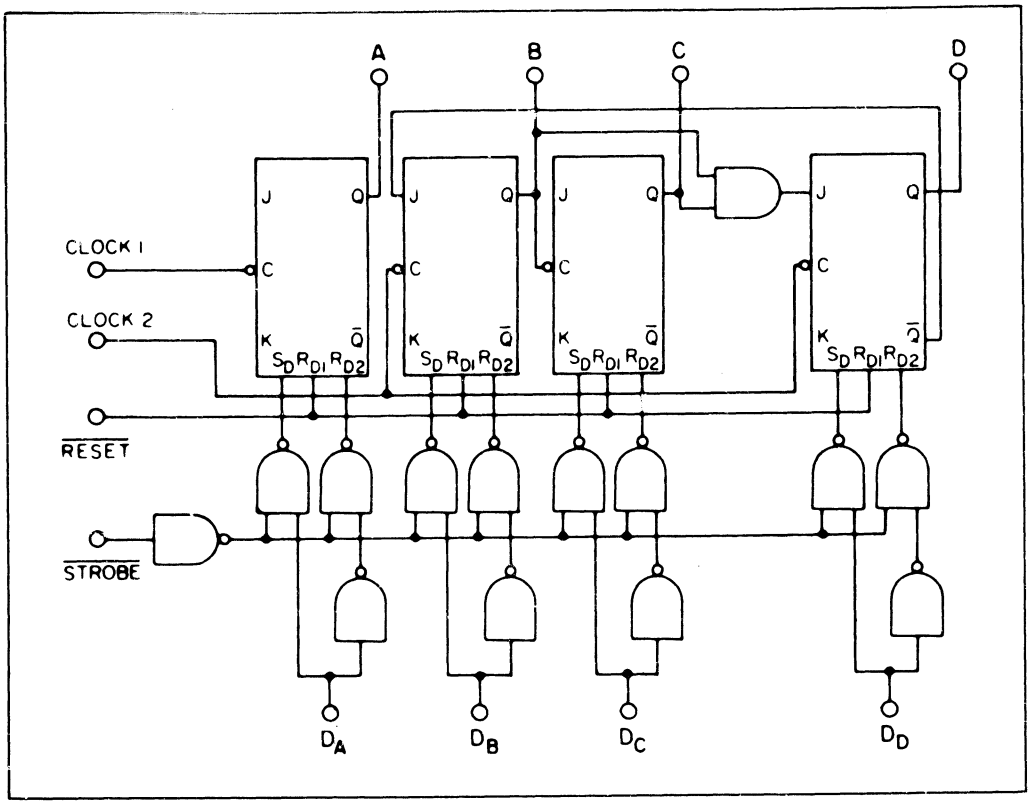
8271B



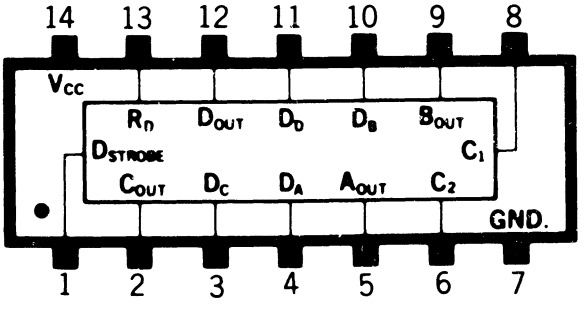
8280

BCD Decade Counter/Storage Element

LOGIC DIAGRAM



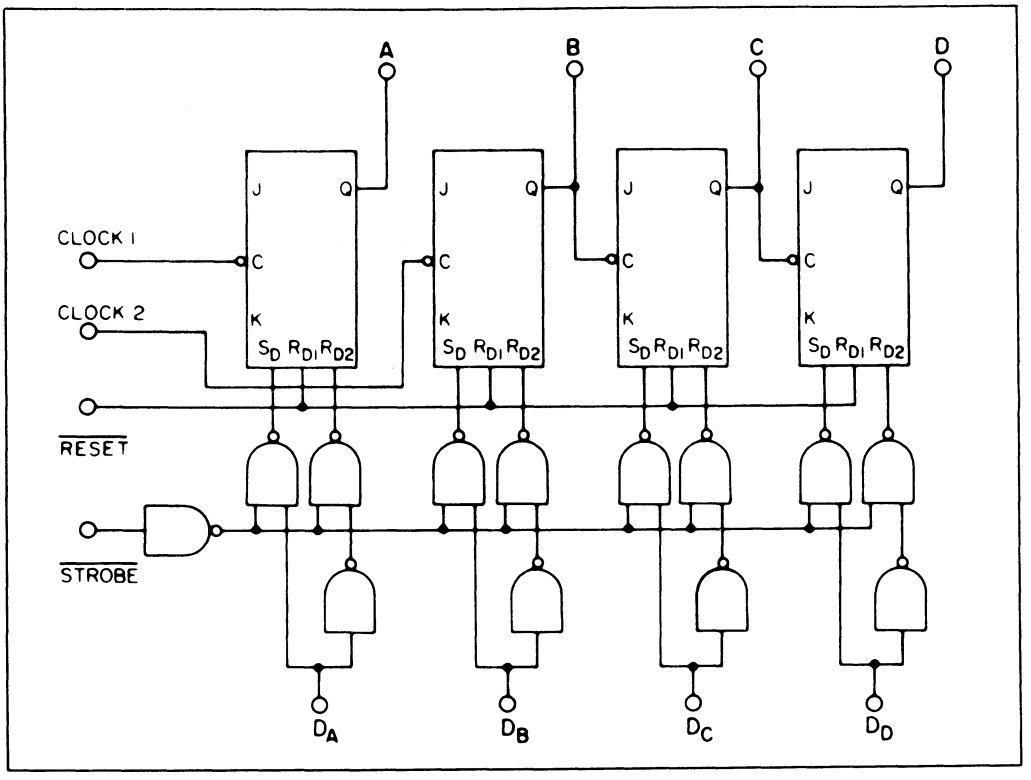
8280 has strobed parallel-entry for setting to any output state. A "1" or "0" at a data input will be transferred to the associated output when the strobe input is put at the "0" level. For additional flexibility, both units are provided with a reset input which is common to all four bits. A "0" on the reset line produces "0" at all four outputs. The counting operation is performed on the falling (negative-going) edge of the input clock pulse.



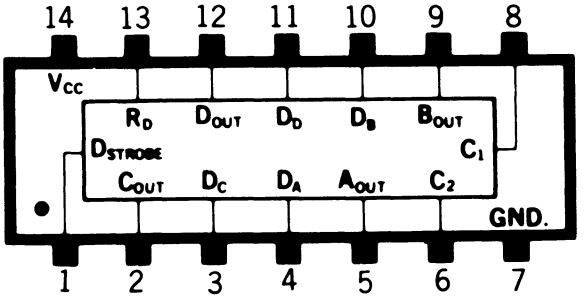
8281

4-Bit Binary Counter/Storage Element

LOGIC DIAGRAM



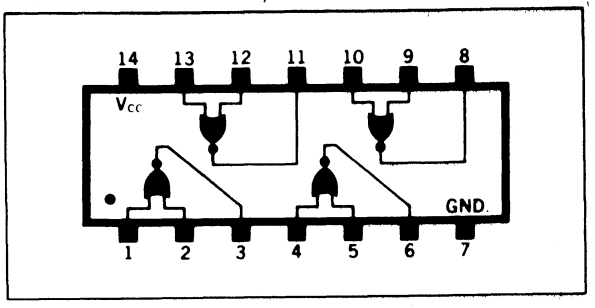
8281 has strobed parallel-entry for setting to any output state. A "1" or "0" at a data input will be transferred to the associated output when the strobe input is put at the "0" level. For additional flexibility, both units are provided with a reset input which is common to all four bits. A "0" on the reset line produces "0" at all four outputs. The counting operation is performed on the falling (negative-going) edge of the input clock pulse.



8885

Quad 2 - Input NOR Gate

LOGIC DIAGRAM/PIN DESIGNATIONS



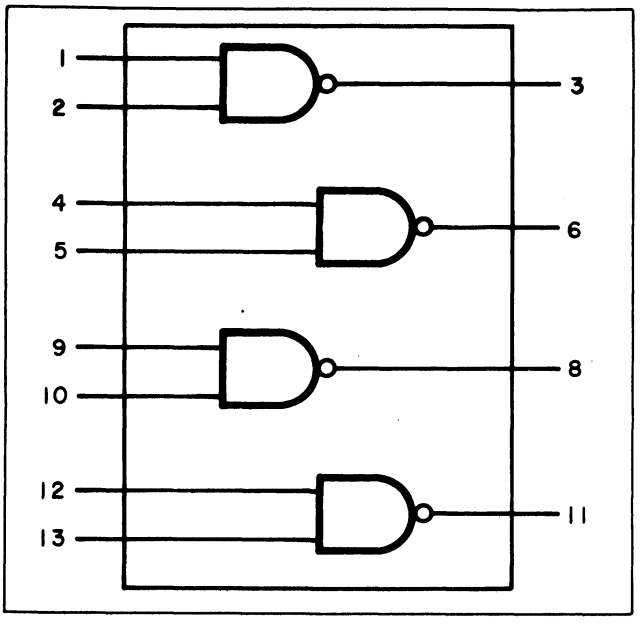
TRUTH TABLE

V_{IN}	V_{IN}	V_{OUT}
H	H	L
H	L	L
L	H	L
L	L	H

9002

Quad 2 - Input NAND Gates

LOGIC DIAGRAM/PIN DESIGNATIONS



V_{CC} = Pin 14

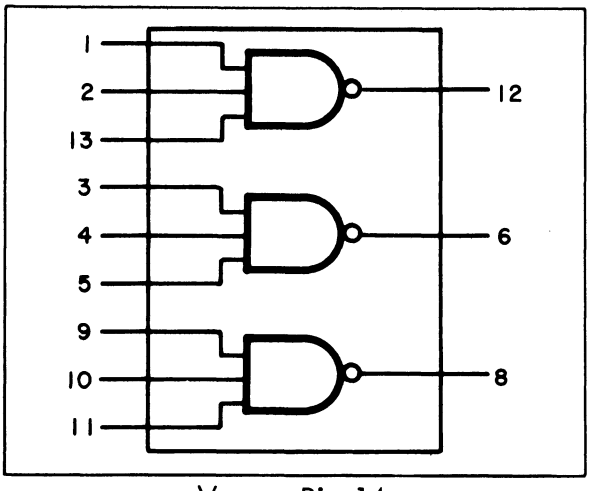
Gnd = Pin 7

TRUTH TABLE LISTED BELOW

9003

Triple 3 - Input NAND Gates

LOGIC DIAGRAM/PIN DESIGNATIONS



V_{CC} = Pin 14

Gnd = Pin 7

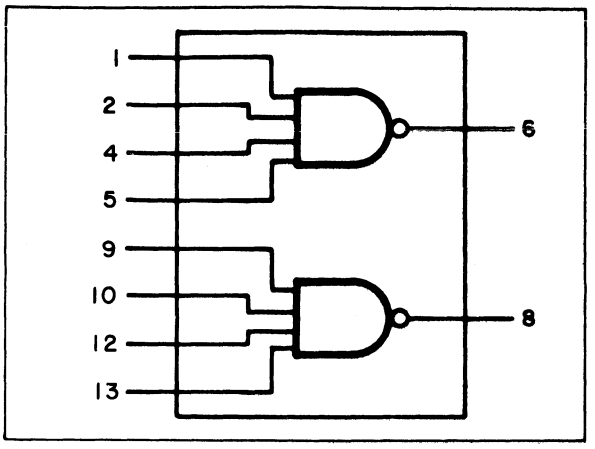
9002 & 9003 TRUTH TABLE

ALL INPUTS HIGH = LOW OUT
ALL INPUT LOW = HIGH OUT

9004/9009*

Dual 4 - Input NAND Gates

LOGIC DIAGRAM/PIN DESIGNATIONS



V_{CC} = Pin 14

Gnd = Pin 7

*9009 Has Higher Input-Output Loading Parameters Than 9004

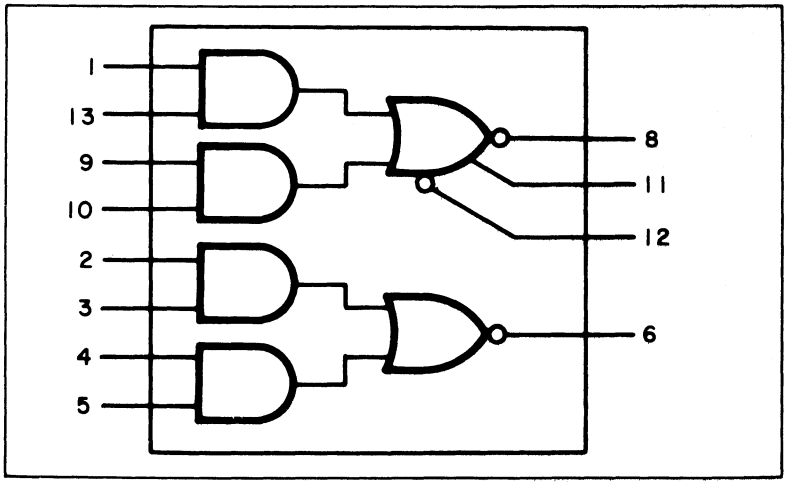
TRUTH TABLE

All Inputs High = Low Out
Any Input Low = High Out

9005

Dual Extendable AND-OR-INVERT Gates

LOGIC DIAGRAM/PIN DESIGNATIONS



*Four Extenders (9006) may be tied to these terminals

V_{CC} = Pin 14

Gnd = Pin 7

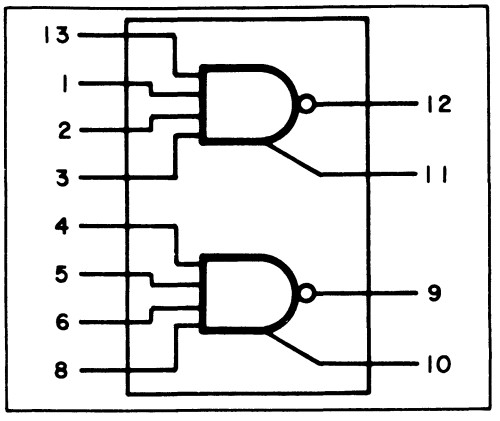
TRUTH TABLE

$(2 \cdot 3) \cdot (4 \cdot 5) = \overline{6}$
$(\overline{2} + \overline{3}) + (\overline{4} + \overline{5}) = 6$

9006

Dual Extender AND-OR-INVERT Gates

LOGIC DIAGRAM



Extender for use with 9005 & 9008

V_{CC} = Pin 14

Gnd = Pin 7

TRUTH TABLE

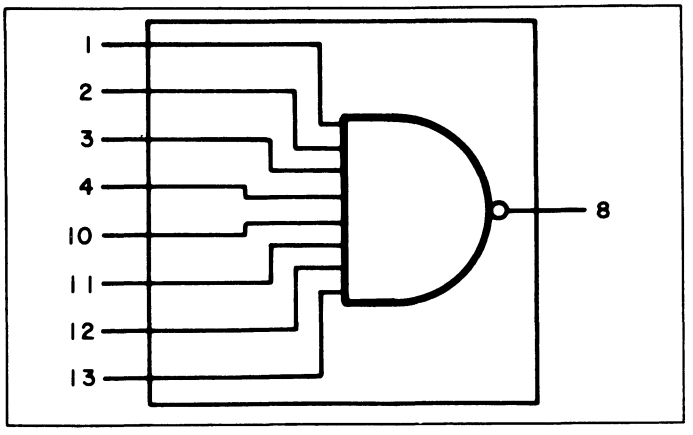
$4 \cdot 5 \cdot 6 \cdot 8 = \overline{9}$

$\overline{4} + \overline{5} + \overline{6} + \overline{8} = 9$

9007

8 - Input NAND Gate

LOGIC DIAGRAM



V_{CC} = Pin 14

Gnd = Pin 7

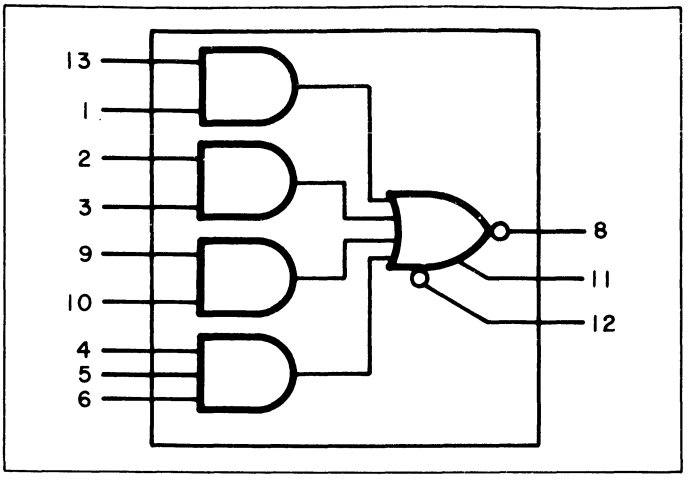
TRUTH TABLE

ALL INPUTS HIGH = LOW OUT
ANY INPUT LOW = HIGH OUT

9008

Single Extendable AND-OR-INVERT Gate

LOGIC DIAGRAM/PIN DESIGNATIONS



*Four Extenders (9006) may be tied to these terminals

V_{CC} = Pin 14
Gnd = Pin 7

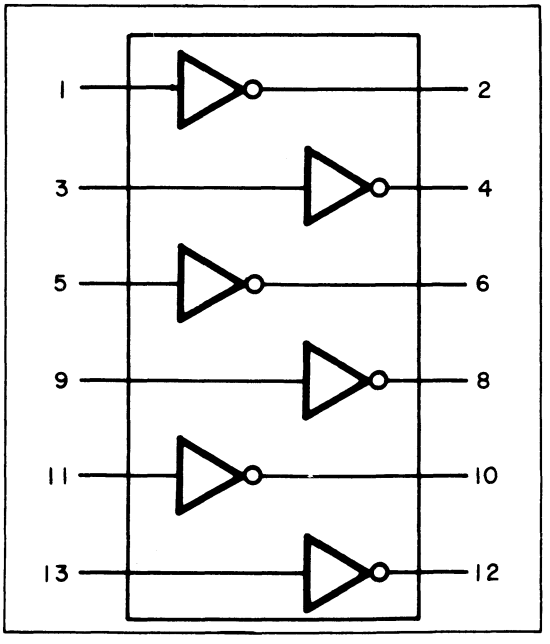
TRUTH TABLE

$(1 \cdot 13) \cdot (2 \cdot 3) \cdot (9 \cdot 10) \cdot (4 \cdot 5 \cdot 6) = \overline{8}$
$(1 + 13) + (2 + 3) + (9 + 10) + (4 + 5 + 6) = 8$

9016

Quad Hex Inverter

LOGIC DIAGRAM/PIN DESIGNATIONS



V_{CC} = Pin 14
Gnd = Pin 7

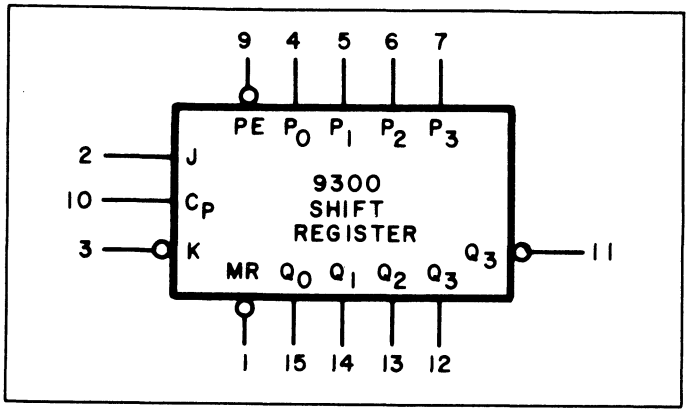
TRUTH TABLE

ANY INPUT LOW = HIGH PUT
ANY INPUT HIGH = LOW OUT

9300

4 - Bit Shift Register

LOGIC DIAGRAM



V_{CC} = Pin 14
Gnd = Pin 8

PIN NOMENCLATURE

$\overline{PE}$	Parallel Enable (Active Low) Input
P_0, P_1, P_2, P_3	Parallel Inputs
J	First Stage J (Active High) Input
$\overline{K}$	First Stage K (Active Low) Input
Cp	Clock Active High Going Edge Input
$\overline{MR}$	Master Reset (Active High) Input
Q_0, Q_1, Q_2, Q_3	Parallel Outputs
$\overline{Q_3}$	Complementary Last Stage Output

Data entry is synchronous with the registers changing state after each low to high transition of the clock. With the parallel enable low the parallel inputs determine the next condition of the shift register. When the parallel enable input is high the shift register performs a one bit shift to the right, with data entering the first stage flip-flop through $J\overline{K}$ inputs. By tying the two inputs together D type entry is obtained.

The asynchronous active low master reset when activated overrides all other input conditions and clears the register.

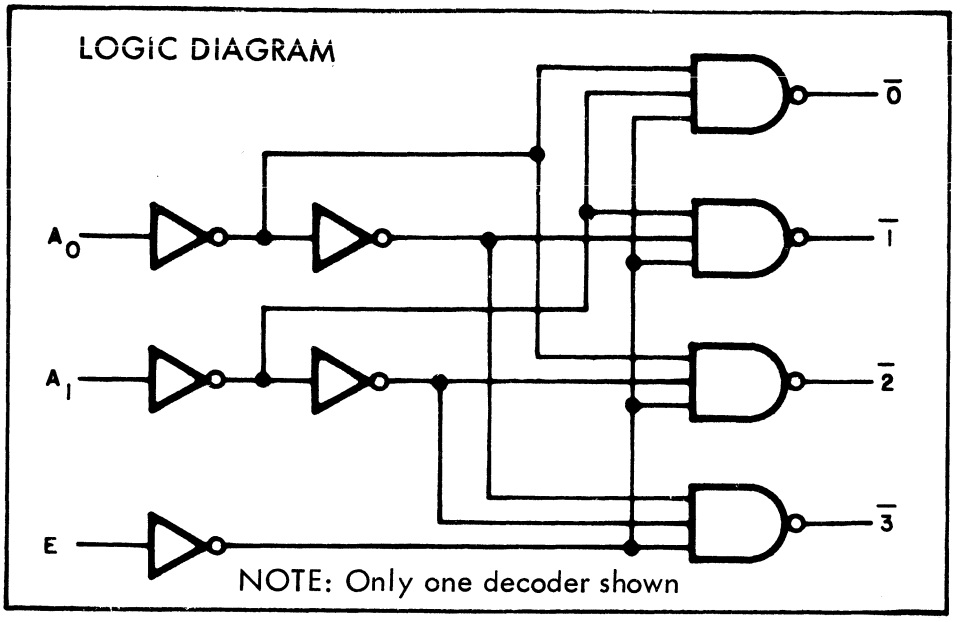
TRUTH TABLE FOR SERIAL ENTRY

J	$\overline{K}$	Q_o at t_{n+1}
L	L	L
L	H	Q_o at t_n (no change)
H	L	$\overline{Q}_o$ at t_n (toggles)
H	H	H

$\overline{PE}$ = HIGH, $\overline{MR}$ = HIGH (n + 1) indicates state after next clock

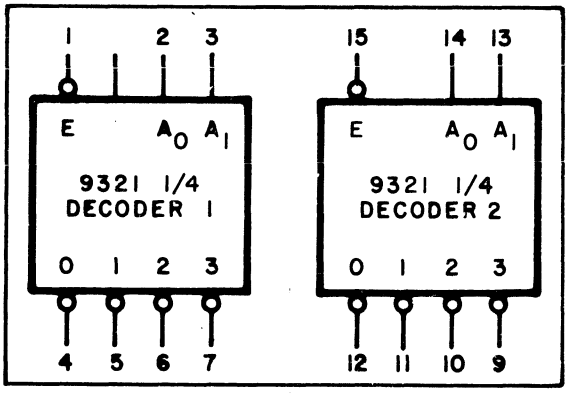
9321

Dual One-Of-Four Decoder



V_{CC} = Pin 16
GND = Pin 8

PIN DESIGNATIONS



H = High Voltage Level
L = Low Voltage Level
X = Don't Care Condition

TRUTH TABLE

$\overline{E}$	A_0	A_1	$\overline{0}$	$\overline{1}$	$\overline{2}$	$\overline{3}$
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	H	L	H	L	H	H
L	L	H	H	H	L	H
L	H	H	H	H	H	L

PIN NOMENCLATURE
Decoder 1 and 2

$\overline{E}$ Enable (Active Low) Input

A_0, A_1 Address Inputs

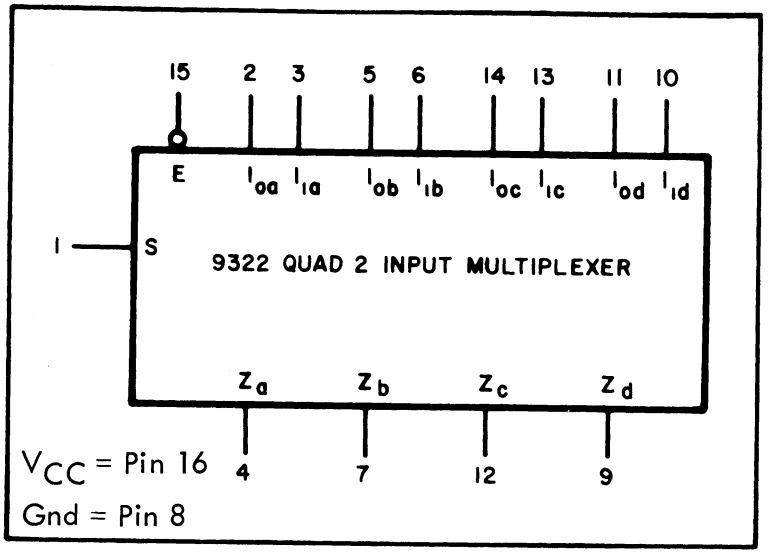
$\overline{0}$, $\overline{1}$, $\overline{2}$, $\overline{3}$ (Active Low) Outputs

The 9321 consists of two independent one-of-four decoders, each with an active low enable. Each decoder accepts two inputs and provides one of four mutually active low outputs.

9322

Quad 2-Input Multiplexer

LOGIC DIAGRAM/PIN DESIGNATIONS



TRUTH TABLE

$\overline{E}$	S	$I_{0a}, I_{0b}, I_{0c}, I_{od}$	$I_{1a}, I_{1b}, I_{1c}, I_{1d}$	Z_a, Z_b, Z_c, Z_d
H	X	X	X	L
L	L	H	X	H
L	L	L	X	L
L	H	X	H	H
L	H	X	L	L

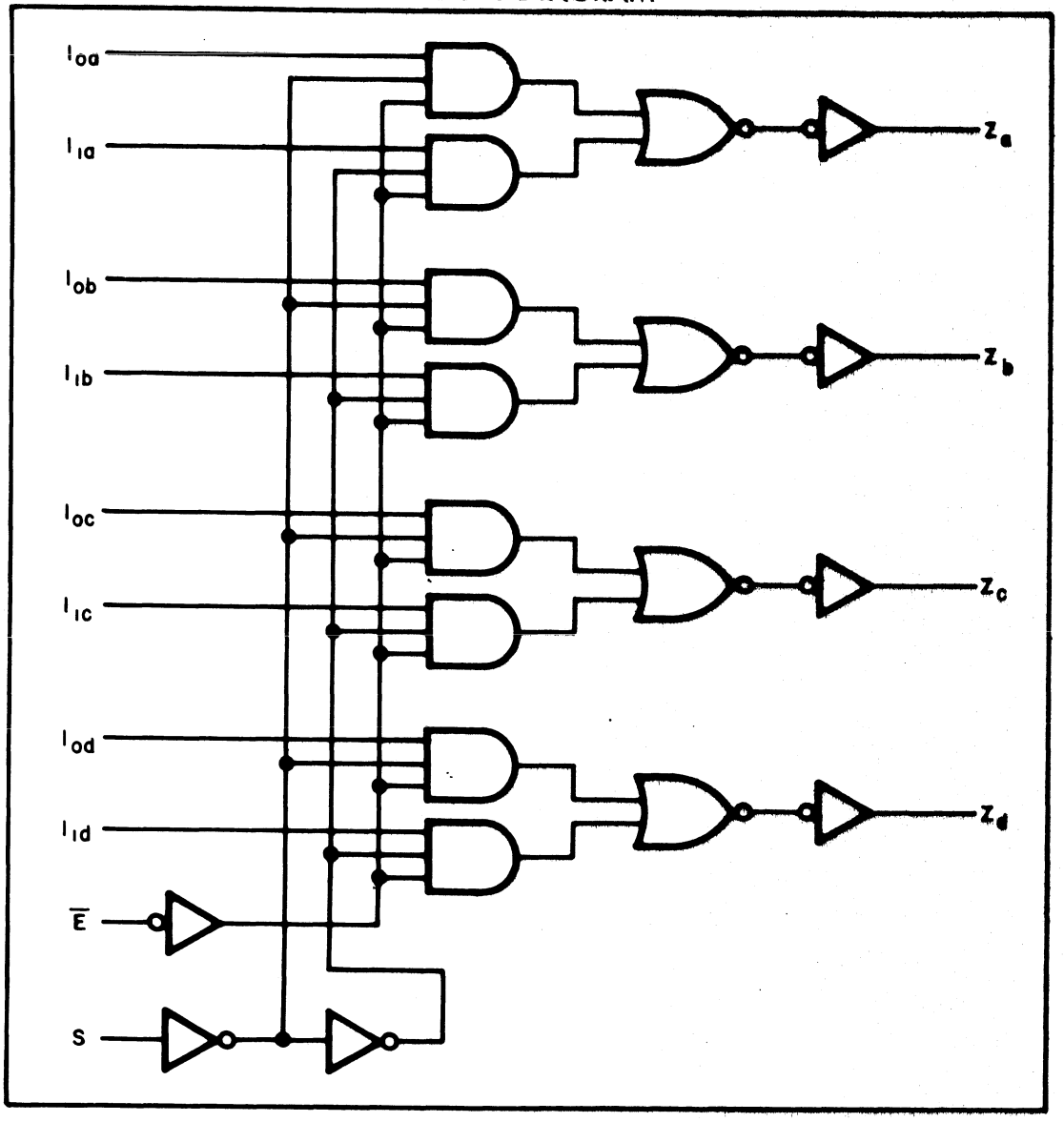
H = High Voltage Level
L = Low Voltage Level
X = Don't Care Condition

PIN NOMENCLATURE
S Common Select Input
$\overline{E}$ Enable (Active Low) Input
Multiplexers A, B, C, D
I_0, I_1 Multiplexer Inputs
Z Multiplexer Outputs

9322 (cont.)

Quad 2-Input Multiplexer

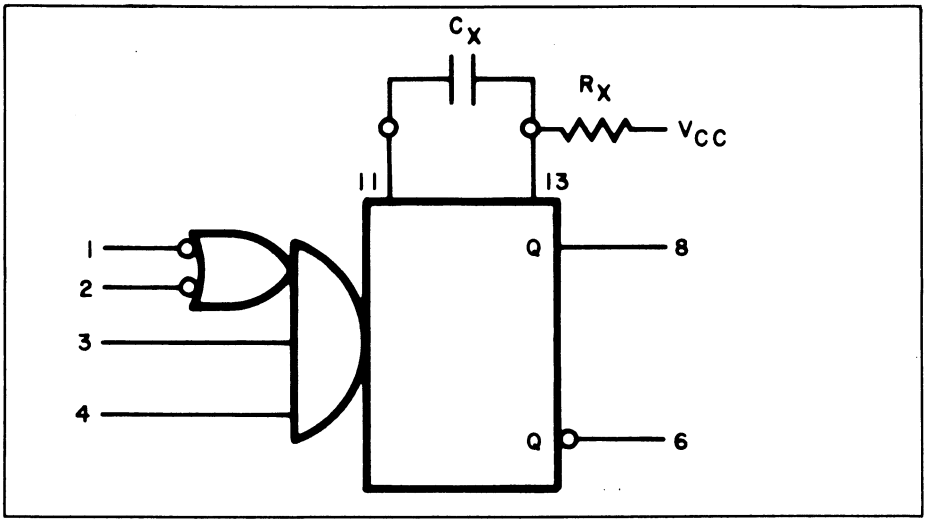
LOGIC DIAGRAM



9601

Monostable Multivibrator (One Shot)

LOGIC DIAGRAM



V_{CC} = Pin 14
Gnd = Pin 7

The inputs are dc coupled hence triggering is independent of input transition times. If the input signal is applied to an active high input, triggering will occur on the rising edge of the waveform. By applying the input signal to an active low input, triggering will occur on the falling edge of the waveform.

Each time the input conditions for triggering are met, the external capacitor is discharged and a new cycle is started. Successive inputs with a period shorter than the delay time (R_XC_X) retrigger the monostable resulting in a continuous true output. Retriggering may be inhibited by tying the negation ($\overline{Q}$) output back to an active level low input.

The formula for calculating the delay time constant is:

0.36 x R(in ohms) x C(in Farads) = T(in seconds)

APPENDIX B

NOVA 1200 SIGNAL LIST

SIGNAL ORIGIN

CENTRAL PROCESSOR

AND

MEMORY

NOTE: Blank entries in level column denote flip-flop outputs which may be in either logic state.
Accumulators, Adders, I/O DATA lines, memory INHIBIT lines MEMORY ADDRESS (MA) flip-flops, MEMORY BUFFER (MB) flip-flops, PROGRAM COUNTER (PC) flip-flops, and RINH flip-flops output levels are not defined in the level column.

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
ACBØ	1Ø5	5		88-4	ACB12	1Ø5	14		88-4
ACB1	1Ø6	5		"	ACB13	1Ø6	14		"
ACB2	1Ø7	5		"	ACB14	1Ø7	14		"
					LOAD MBO*	98	6	H	88-3
					KEYM SET*	1Ø1	9	H	88-1
ACB3	1Ø8	5		"	ACB15	1Ø8	14		88-4
ACB4	1Ø5	7		"	ACBØ	1Ø5	3		"
ACB5	1Ø6	7		"	ACB1	1Ø6	3		"
ACB6	1Ø7	7		"	ACB2	1Ø7	3		"
ACB7	1Ø8	7		"	ACB3	1Ø8	3		"
ACB8	1Ø5	9		"	ACB4	1Ø5	2		"
ACB9	1Ø6	9		"	ACB5	1Ø6	2		"
ACB1Ø	1Ø7	9		"	ACB6	1Ø7	2		"
ACB11	1Ø8	9		"	CRY SET*	81	13	H	88-3
					ACB7	1Ø8	2		88-4
					SHIFTER Logic	114	1Ø		"
ACB12	1Ø5	11		88-4	ACB12 Save	69	3		88-1
					SHIFTER Logic	1Ø9	9		88-4
ACB12*	1Ø5	12		"	SHIFTER	125	19		"
ACB13	1Ø6	11		"					
ACB13*	1Ø6	12		"	SHIFTER	125	2		88-4
					SHIFTER	125	2Ø		"
ACB14	1Ø7	11		88-4					
ACB14*	1Ø7	12		"	SHIFTER	125	1		88-4
					SHIFTER	125	5		"
					SHIFTER	125	18		"
ACB15	1Ø8	11		88-4					
ACB15*	1Ø8	12		"	SHIFTER	125	3		"
ACB12 SAVE	69	5	H	88-1	SHIFTER Logic	9Ø	1		88-4
ACDØ	123	5		88-4	MULT	12Ø	5		"
					D BUFFR	122	3		"
ACD1	123	7		88-4	MULT	12Ø	2		"
					D BUFFR	122	2		"
ACD2	123	9		88-4	MULT	12Ø	22		"
					D BUFFR	122	15		"
ACD3	123	11		88-4	MULT	12Ø	19		"
					D BUFFR	122	14		"
ACD3 SEL*	5Ø	6	L	88-2	ACD	123	1		"
ACD4 SEL*	44	8	L	"	ACD	123	15		"
ACD OUT*	45	6	L	"	D MULT(SEL)	121	1		"
[ACSØ]	124	5		88-4	S BUFFR	115	3		"
[ACS1]	124	7		"	"	115	2		"
[ACS2]	124	9		"	"	115	15		"
*Indicates "NOT"									

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
[ACS3]	124	11		88-4	S BUFFR	115	14		88-4
ACS1 SEL*	49	6,8	(OC)	88-2	ACS	124	1		"
ACS2 SEL*	49	3,11	"	88-2	ACS	124	15		"
ACTGØ	54	5		88-1	ACTG1	73	9	H	88-1
					IR (SH) Logic	111	2	"	88-2
					ACD	123	14		88-4
					ACS	124	14		"
ACTG1	54	7		88-1	ACTGØ	53	9	L	88-1
					IR(SH) Logic	111	9	L	88-1
					ACD	123			88-4
					ACS	124	13		"
ADDERØ	117	13		88-4	CRY SET*	81	3	H	88-3
					ACB (DS)	1Ø5	4		88-4
					ACB8	1Ø5	15		"
					PC Logic	118	5,4		"
					MULT	12Ø	4		"
ADDER1	117	11		88-4	ACB (DS)	1Ø6	4		"
					ACB9	1Ø6	15		"
					PC Logic	118	1,2		"
					MULT	12Ø	1		"
ADDER2	117	1Ø		88-4	ACB (DS)	1Ø7	4		"
					ACB1Ø	1Ø7	15		"
					PC Logic	118	12,13		"
					MULT	12Ø	23		"
ADDER3	117	9		88-4	ACB (DS)	1Ø8	4		88-4
					ACB11	1Ø8	15		"
					PC Logic	118	9, 1Ø		"
					MULT	12Ø	2Ø		"
ADD ONE*	88	8	L	88-2	ADDER	117	7		88-4
ADDER TEST	58	3	H	88-3	LOOP SET*	1Ø4	5	H	88-3
ALC	94	6	H	88-2	DISABLE D MULT	46	1Ø	H	88-2
					SØ	47	1	H	"
					TEST SKIP SET	86	5	H	88-3
ALC*	5Ø	8	L	88-2	ADD ONE*	44	2	L	88-2
					AND	65	5	L	"
					E SET	74	1	H	"
					S2	91	12	L	"
					ALC	94	5	L	"
					S BUFFR (SH)	115	13	L	88-4
ALC. $\overline{\text{SKIP}}$	83	1Ø	H	88-3	LOAD CRY *	97	13	H	88-3
AND	65	6	H	88-2	CRY ENAB	91	2	L	"
					S1	91	5	H	88-2
					ADDER	117	8	H	88-4
AND ENAB*	64	11	L	88-2	IO DCDR	62	13	H	88-1

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
					AND	65	4	L	88-2
					PACK	89	2	H	88-3
CARRY (F/F)	76	8		88-3	CRY ENAB	77	4	L	88-3
CARRY*(F/F)	76	9		"	CON IND(A15,P49)	6	5	L	89-1
					CRY ENAB	77	3	L	88-3
CLK FLOP	2Ø	5	H	88-1	MA LOAD*	56	1Ø	H	88-1
					CPU CLK	72	2, 12	H	"
					MEM CLK	73	3	H	"
					LOAD AC*	93	5	H	88-3
CLK FLOP*	2Ø	6	L	88-1	CLK FLOP	2Ø	2		88-1
[CLR*]	63	5	L	"	CLR	7	1	L	"
CLR	7	2	H	"	(IO CLR PLS)	(A5Ø)		H	
CLR ION*	63	11	L	"	ION	84	4	L	88-2
CLR SKIP*	99	8	L	88-3	SKIP	79	13	L	88-3
					LOAD MBO*	98	1Ø	H	"
[CONØ*](S11)	6	4	L	89-1	MEMØ*	(B71)	(391)	L	89-1
					(CON IND)	7	9	L	"
[CON1*](S12)	6	2	L	89-1	MEM1*	(B7Ø)	(P41)	L	"
					(CON IND)	7	13	L	"
[CON2*](S13)	6	8	L	89-1	MEM2*	(B47)	(P13)	L	"
					(CON IND)	7	3	L	"
[CON3*](S14)	6	12	L	89-1	MEM3*	(B68)	(P43)	L	"
					(CON IND)	7	1	L	"
[CON4*](S15)	3	8	L	89-1	MEM4*	(B28)	(P37)	L	"
					(CON IND)	8	13	L	"
[CON5*](S16)	3	1Ø	L	89-1	MEM5*	(B26)	(P36)	L	"
					(CON IND)	8	3	L	"
[CON6*](S17)	3	6	L	89-1	MEM6*	(B22)	(P1Ø)	L	"
					(CON IND)	8	1	L	"
[CON7*](S18)	3	4	L	89-1	MEM7*	(B24)	(P42)	L	"
					(CON IND)	9	13	L	"
[CON8*](S19)	3	2	L	89-1	MEM8*	(A55)	(P34)	L	"
					(CON IND)	9	3	L	"
[CON9*](S2Ø)	3	12	L	89-1	MEM9*	(A53)	(P7)	L	"
					(CON IND)	9	1	L	"
[CON1Ø*](S21)	4	8	L	89-1	MEM1Ø*	(A45)	(P32)	L	"
					(CON IND)	1Ø	13	L	"
[CON11*2(S22)	4	1Ø	L	89-1	MEM11*	(A51)	(P31)	L	89-1
					(CON IND)	1Ø	3	L	"
[CON12*](S23)	4	12	L	89-1	MEM12*	(A36)	(P5)	L	"
					(CON IND)	1Ø	1	L	"
*Indicates "NOT"									

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
[CON13*] (S24)	4	6	L	89-1	MEM13*	(A35)	(P29)	L	89-1
					(CON IND)	11	13	L	"
[CON14*] (S25)	4	4	L	89-1	MEM14*	(B76)	(P3)	L	"
					(CON IND)	11	3	L	"
[CON15*] (S26)	4	2	L	89-1	MEM15*	(B18)	(P2)	L	"
					(CON IND)	12	13	L	"
CON DATA*	4	8	L	88-1		(A28)	(P46)	L	89-1
					[CONØ*] (S11)	6	3	L	"
					[CON1*] (S12)	6	1	L	"
					[CON2*] (S13)	6	9	L	"
					[CON3*] (S14)	6	13	L	"
					[CON4*] (S15)	3	9	L	"
					[CON5*] (S16)	3	11	L	"
					[CON6*] (S17)	3	5	L	"
					[CON7*] (S18)	3	3	L	"
					[CON8*] (S19)	3	1	L	"
					[CON9*] (S2Ø)	3	13	L	"
					[CON1Ø*] (S21)	4	9	L	"
					[CON11*] (S22)	4	11	L	"
					[CON12*] (S23)	4	13	L	"
					[CON13*] (S24)	4	5	L	"
					[CON14*] (S25)	4	3	L	"
					[CON15*] (S26)	4	1	L	"
CON INST*	36	8	L	88-1		(A22)	(P22)		89-1
					[CON INST]	5	9	L	"
[CON INST]	5	8	H	89-1	MEMØ*	1	2	(OC)	"
					MEM1*	1	4	(OC)	"
					MEM2*	2	1Ø	(OC)	"
					MEM3*	1	12	(OC)	"
					MEM4*	1	1Ø	(OC)	"
					MEM5*	2	12	(OC)	"
					MEM6*	2	2	(OC)	"
					MEM7*	2	4	(OC)	"
CON RQ*	5	6	L	89-1					
	(A27)	(P21)			KEY SEEN	3	4	L	88-1
CONT+ISTP+MSTP*	(A25)		L	89-1	KEY ENAB*	3	2	H	"
CPU CLK	72	6,8		88-1	MB LOAD	14	4	H	"
					IR4-IR7	28	6	L	88-2
					MBC	32	6	L	"
					MBC	33	6	L	"
					MBO	37	6	L	88-4
					MBO	38	6	L	"
					MBO	39	6	L	"

*Indicates "NOT"

NOVA 1200 SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
CPU CLK (Contd)					MBO	40	6	L	88-4
						42	6	L	88-1
					LOAD PC*	57	10	H	88-3
					MA LOAD*	60	10	H	88-1
					INPUT	66	13	L	"
					PTG	69	6	L	"
					SKIP	78	13	L	88-3
					MAJOR STATES	95	6	L	88-2
					CARRY F/F Logic	97	9	H	88-3
						102	6	L	"
					LOOP/PACK/EFA	103	6	L	"
					ACB	105	6	L	88-4
					ACB	106	6	L	"
					ACB	107	6	L	"
					END CYCLE F/F	113	13	L	88-1
CPU INST	6	11	H	88-2	INTA	6	5	H	"
					IORST	6	10	H	"
					(SKIP LOGIC)	11	2	(OC)	88-3
					"	11	12	(OC)	"
					CON DATA*(Reads)	24	4	H	88-1
					(IO OCDR)	64	2	H	"
					HALT*	71	2	H	88-2
					PACK LOGIC	87	4	H	88-3
CPU INST*	10	6	L	88-2	MSKO*	4	13	L	88-1
					CPU INST	6	12,13	L	88-2
CRY ENAB	80	11	(XOR)	88-3	CRY SET*	81	4	H	88-3
					CRY ENAB SAVE	102	15	H	"
CRY ENAB SAVE	102	9	H	88-3	SHIFT LOGIC	90	10	H	88-4
					" "	114	13	H	"
CRY OUT*	117	16		88-4	SERIAL CRY	54	14	L	88-1
					CRY ENAB	91	1	L	88-3
CRY SET*	81	8	L	88-3	CRY SET SAVE	42	15	L	88-1
					CARRY F/F	76	12	L	88-3
CRY SET SAVE*	42	9	L	88-1	(SKIP LOGIC)	77	9	H	"
DATA0*	16	11	(OC)	103-1	TERMINATOR				88-4
	17 (B62)	1		"					
DATA1*	16	8	(OC)	103-1	TERMINATOR				88-4
	17 (B65)	3		"					
*Indicates "NOT"									

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
DATA2*	14	11	(OC)	1Ø3-1	TERMINATOR				88-4
	15	1		"					
	(B82)								
DATA3*	14	8	(OC)	1Ø3-1	TERMINATOR				88-4
	15	3		"					
	(B73)								
DATA4*	12	11	(OC)	1Ø3-1	TERMINATOR				88-4
	13	1		"					
	(B61)								
DATA5*	12	8	(OC)	1Ø3-1	TERMINATOR				88-4
	13	3		"					
	(B57)								
DATA6*	1Ø	11	(OC)	1Ø3-1	TERMINATOR				88-4
	11	1		"					
	(B95)								
DATA7*	1Ø	8	(OC)	1Ø3-1	TERMINATOR				88-4
	11	3		"					
	(B55)								
DATA8*	8	11	(OC)	1Ø3-1	TERMINATOR				88-4
	9	1		"					
	(B6Ø)								
DATA9*	8	8	(OC)	1Ø3-1	TERMINATOR				88-4
	9	3		"					
	(B63)								
DATA1Ø*	6	11	(OC)	1Ø3-1	TERMINATOR				88-4
	7	1		"					
	(B75)								
DATA11*	6	8	(OC)	1Ø3-1	TERMINATOR				88-4
	7	3		"					
	(B58)								
DATA12*	4	11	(OC)	1Ø3-1	TERMINATOR				88-4
	5	1		"					
	(B59)								
DATA13*	4	8	(OC)	1Ø3-1	TERMINATOR				88-4
	5	3		"					
	(B64)								
DATA14*	2	11	(OC)	1Ø3-1	TERMINATOR				88-4
	3	1		"					
	(B56)								
DATA15*	2	8	(OC)	1Ø3-1	TERMINATOR				88-4
	3	3		"					
	(B66)								
*Indicates "NOT"									

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
[DATOA*]	25	6	L	88-1	DATOA	7 (A58)	9	L	88-1
DATOA	7	8	H	"					
DATOB*	25	5	L	"	DATOB	7	13	L	88-1
					MSKO*	4 (A56)	12	L	"
DATOB	7	12	H	88-1					
[DATOC*]	25	4	L	"	DATOC	26 (A48)	5	L	88-1
DATOC	26	6	H	"					
[DATIA*]	25	9	L	"	DATIA	5	13	L	88-1
DATIA	5	12	H	"	CON DATA*	24 (A44)	5	H	"
[DATIB*]	25	1Ø	L	88-1	DATIB	5	11	L	88-1
DATIB	5	1Ø	H	"	INTA	6 (A42)	4	H	"
[DATIC*]	25	11	L	88-1	DATIC	7	5	L	88-1
DATIC	7	6	H	"	IORST	6 (A54)	9	H	"
[D BUFFRØ]	122	5		88-4	[D MULTØ]	121	2		88-4
[D BUFF1]	122	7		"	[D MULT1]	121	5		"
[D BUFFR2]	122	9		"	[D MULT2]	121	14		"
[D BUFFR3]	122	11		"	[D MULT3]	121	11		"
DCH	23	9	H	88-1	DCHI	14	9	H	88-1
					DCH LOOP ENAB	15	2	H	"
					ADD ONE*	41	2	H	88-2
DCHA	69	7	H	88-1	DCHA*	7	11	H	88-1
					DRIVE IO*	13	5	H	"
					DCH	23 (A6Ø)	15	H L	"
DCHA*	7	1Ø	L	88-2					
DCHA SET*	71	8	L	88-1	[DCHA SET]	67	3	L	88-1
					FETCH	97	1	H	88-2
[DCHA SET]	67	4	H	88-1	DCHA	69 (B37)	2	H H	88-1
DCHI	14	8	H	"					
					DRIVE IO*	13	4	H	88-1
DCH LOOP ENAB	15	6	H	88-1	OVFLO	15	9	H	"
					DCHO	18	12,13	H	"
					ACTG(LD)	75	1Ø	H	"
					LOOP SET*	1Ø4	1Ø,13	H	88-3
DCHMØ*	(B17)		L	88-1	DCH LOOP ENAB	15	4, 5	H	88-1
					[DCHMØ]	16	1	L	"
[DCHMØ]	16	2	H	88-1	DCHI	14	1Ø	H	88-1
DCHM1*	(B21)		L	"	"	14	12	H	"
					[DCHM1]	16	3	L	"
					LOOP SET*	34	12	L	88-3

* Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
[DCHM1]	16	4	H	88-1	OVFLO	15	1Ø	H	88-1
DCHO	18	8	H	"		(B33)		H	
DCHR*	(B35)		L	"	DCHR PEND	13	2	L	88-1
DCHR PEND	13	3	H	"	DCHA SET*	71	1Ø	H	"
					LOOP SET*	1Ø4	9	H	88-3
DEFER	95	7	H	88-2	DEFER AGAIN	76	4	L	88-2
					ADD ONE*	9Ø	4	H	"
					DEFER*	94	11	H	"
					LOOP SET*	1Ø4	6	H	88-3
DEFER*	94	1Ø	L	88-2	(CON IND)	(A12)	(P52)	L	89-1
					SØ	48	1	L	88-2
					ADDER TEST	58	12	L	88-3
					" "	59	1Ø	H	"
					FETCH +DEFER	75	2	L	88-2
DEFER AGAIN*	76	5		88-2	D SET	74	9	L	"
(D+E SET)+$\overline{TS3}$	36	11	L	"	DCHR PEND	13	1	L	88-1
					(RUN LOGIC)	24	13	L	"
					PC IN*	35	1	L	88-2
D+E SET*	96	11	L	88-2	(D+E SET)+$\overline{TS3}$	36	13	H	88-2
					(RUN LOGIC)	43	13	H	88-1
					FETCH LOGIC	97	5	H	88-2
DISABLE D MULT	53	3	H		D MULT(ENAB)	121	15	L	88-4
DIV*	(A91)		L		CARRY F/F	76	1Ø	L	88-3
[D MULTØ]	121	4		88-4	ADDER	117	19		88-4
[D MULT1]	121	7		"	"	117	21		"
[D MULT2]	121	12		"	"	117	23		"
[D MULT3]	121	9		"	"	117	2		"
DRIVE IO*	12	8	H	88-1		(B88)			
					READ IO*	12	4,5	H	88-1
					[DRIVE IO]	18	1	L	1Ø3-1
[DRIVE IO]	18	2	H	1Ø3-1	[Drive IO* Select]	26	9,1Ø,12	H	"
[DRIVE IO* Select]	26	8	H	"	DATAØ*	16	12	H	"
					DATA1*	16	1Ø	H	"
					DATA2*	14	12	H	"
					DATA3*	14	1Ø	H	"
					DATA4*	12	12	H	"
					DATA5*	12	1Ø	H	"
					DATA6*	1Ø	12	H	"
					DATA7*	1Ø	1Ø	H	"
					DATA8*	8	12	H	"

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
[DRIVE IO' Select] (Contd)					DATA9*	8	1Ø	H	1Ø3-1
					DATA1Ø*	6	12	H	"
					DATA11*	6	1Ø	H	"
					DATA12*	4	12	H	"
					DATA13*	4	1Ø	H	"
					DATA14*	2	12	H	"
					DATA15*	2	1Ø	H	"
DSØ*	8	8	L	88-1		(A72)		L	
DS1*	8	1Ø	L	"		(A68)		L	
DS2*	8	12	L	"		(A66)		L	
DS3*	22	8	L	"		(A46)		L	
DS4*	8	4	L	"		(A62)		L	
DS5*	8	2	L	"		(A64)		L	
D SET	74	8	H	88-2	DEFER	95	2	H	88-2
					E SET	96	2	L	"
					D+E SET*	96	13	H	"
DSZ. E. TSØ*	52	4	L	88-2	SØ	92	1	L	"
EFA	1Ø2	11	H	88-3	MBC(SH)	32	13	H	"
					MBC(SH)	33	13	H	"
					ACD4 SEL*	44	9	H	"
					ACD OUT*	45	1Ø	H	"
					DISABLE D MULT	46	4	H	"
					SØ	47	3	H	"
					SØ	47	4	H	"
					D SET	74	4	H	"
					JSR. EFA	93	13	H	"
EFA*	1Ø3	12	L	88-3	EFA. PTG	34	5	L	"
					ACD4 SEL*	44	1	H	"
					ACD3 SEL*	5Ø	3	H	"
EFA. PTG1	34	6	H	88-2	MBC (DS)	32	4	H	"
					S MULT(SEL)	116	1	H	88-4
END CYCLE (F/F)	113	1		88-1	ACTG	53	1Ø,12	L	88-1
					End Cycle (F/F)	113	2		"
					LOAD CRY*	97	12	H	88-3
					(LD) Test Skip	1Ø2	1Ø	H	"
					(LD) Loop/Pack	1Ø3	1Ø	H	"
					SHIFTER Logic	1Ø9	13	H	88-4
					" "	114	1	H	"
END CYCLE*(F/F)	113	6		88-1	SHIFTER Logic	114	9	H	"
					PTGØ. TSØ Logic	112		L	88-1
* Indicates "NOT"									

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
E SET	96	3	H	88-2	EXEC	95	15	H	88-2
					D+E SET*	96	12	H	"
EXEC	95	9	H	88-2	EXEC*	73	11	H	"
				"	(INST DCDR)	92	9	H	"
EXEC*	73	1Ø	L	"	(CON IND)	(A11)	(P51)		89-1
					(INST DCDR)	52	15	L	88-2
EXT LOAD*	(A47)		L		LOAD AC*	111	4	L	88-3
	(B49)		L		SHIFTER (Enab)	125	8,9		88-4
EXT Select*	(B8Ø)		L		SELECT	35	9,1Ø	L	1Ø3-1
FETCH	95	5	H	88-2	MB LOAD	13	13	H	88-1
					LOAD IR	34	9	H	88-2
					LOAD PC*	61	1Ø	H	88-3
					FETCH. TSØ*	64	9	H	88-2
					ALC*	5Ø	9	H	"
					ION	85	1	H	"
					FETCH*	94	13	H	"
					CLR SKIP*	1ØØ	4	H	88-3
FETCH*	94	12	L	88-2	(CON IND)	(A13)	(P5Ø)	L	89-1
					ACD OUT*	45	1,13	H	88-2
					FETCH+DEFER	75	1	L	"
					ADD ONE*	89	12	L	"
FETCH+DEFER	75	3	H	88-2	IRØ+SKP	5Ø	1	H	"
					E SET	74	13	H	"
FETCH. TSØ*	64	8	L	88-2	EFA	85	12	L	"
					MULT(SEL)	12Ø	16	H	88-4
FORCE LOAD IR*	(A85)				IR(LD)	12	4	L	88-2
HALT*	71	6	L	88-2	MB LOAD	14	2	H	88-1
					(RUN LOGIC)	62	3	L	"
					DCHA	71	9	H	"
INHØ	34	9		1Ø3-1	MEMØ*	16	1	H	1Ø3-1
					DATAØ*	16	13	H	"
INHØ*	34	8		1Ø3-1	(INHBØ) (Q15)	68	12	H	1Ø3-2
INH1	34	5		"	MEM1*	16	5	H	1Ø3-1
					DATA1*	16	9	H	"
INH1*	34	6		1Ø3-1	(INHB1) (Q16)	68	2	H	1Ø3-2
INH2	32	5		"	MEM2*	14	1	H	1Ø3-1
					DATA2*	14	13	H	"
INH2*	32	6		1Ø3-1	(INHB2) (Q13)	64	2	H	1Ø3-2
INH3	32	9		"	MEM3*	14	5	H	1Ø3-1
					DATA3*	14	9	H	"

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
INH3*	32	8		1Ø3-1	(INHB3) (Q14)	64	12	H	1Ø3-2
INH4	31	9		"	MEM4*	12	1	H	1Ø3-1
					DATA4*	12	13	H	"
INH4*	31	8		1Ø3-1	(INHB4)(Q11)	58	12	H	1Ø3-2
INH5	31	5		"	MEM5*	12	5	H	1Ø3-1
					DATA5*	12	9	H	"
INH5*	31	6		1Ø3-1	(INHB5) (Q12)	58	2	H	1Ø3-2
INH6	28	5		"	MEM6*	1Ø	1	H	1Ø3-1
					DATA6*	1Ø	13	H	"
INH6*	28	6		1Ø3-1	(INHB6) (Q9)	55	2	H	1Ø3-2
INH7	28	9		"	MEM7*	1Ø	5	H	1Ø3-1
					DATA7*	1Ø	9	H	"
INH7*	28	8		1Ø3-1	(INHB7) (Q1Ø)	55	12	H	1Ø3-2
INH8	27	9		"	MEM8*	8	1	H	1Ø3-1
					DATA8*	8	13	H	"
INH8*	27	8		1Ø3-1	(INHB8) (Q7)	48	12	H	1Ø3-2
INH9	27	5		"	MEM9*	8	5	H	1Ø3-1
					DATA9*	8	9	H	"
INH9*	27	6		1Ø3-1	(INHB9) (Q8)	48	2	H	1Ø3-2
INH1Ø	24	5		"	MEM1Ø*	6	1	H	1Ø3-1
					DATA1Ø*	6	13	H	"
INH1Ø*	24	6		1Ø3-1	(INHB1Ø) (Q5)	45	2	H	1Ø3-2
INH11	24	9		"	MEM11*	6	5	H	1Ø3-1
					DATA11*	6	9	H	1Ø3-1
INH11*	24	8		1Ø3-1	(INHB11) (Q6)	45	12	H	1Ø3-2
INH12	23	9		"	MEM12*	4	1	H	1Ø3-1
					DATA12*	4	13	H	"
INH12*	23	8		1Ø3-1	(INHB12) (Q3)	39	12	H	1Ø3-2
INH13	23	5		H	MEM13*	4	5	H	1Ø3-1
					DATA13*	4	9	H	"
INH13*	23	6		1Ø3-1	(INHB13)(Q4)	39	2	H	1Ø3-2
INH14	21	5		"	MEM14*	2	1	H	1Ø3-1
					DATA14*	2	13	H	"
INH14*	21	6		1Ø3-1	(INHB14) (Q1)	37	2	H	1Ø3-2
INH15	21	9		1Ø3-1	MEM15*	2	5	H	1Ø3-1
					DATA15*	2	9	H	"
INH15*	21	8		1Ø3-1	IINHB15) (Q2)	37	12	H	1Ø3-2
INHBØ	7Ø	3		1Ø3-2	Q15				
INHB1	7Ø	5		"	Q16				
INHB2	63	3		"	Q13				
INHB3	63	5		"	Q14				
INHB4	61	3		"	Q11				
INHB5	61	5		"	Q12				
*Indicates "NOT"									

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
INHB6	53	3		1Ø3-2	Q9				
INHB7	53	5		"	Q1Ø				
INHB8	51	3		"	Q7				
INHB9	51	5		"	Q8				
INHB1Ø	43	3		"	Q5				
INHB11	43	5		"	Q6				
INHB12	42	3		"	Q3				
INHB13	42	5		"	Q4				
INHB14	2Ø	3		"	Q1				
INHB15	2Ø	5		"	Q2				
INH GATE A	26	6	H	1Ø3-1	(INHBØ) (Q15)	68	13	H	1Ø3-2
					(INHB1) (Q16)	68	1	H	"
					INHB2) (Q13)	64	1	H	"
					(INHB3) (Q14)	64	13	H	"
					(INHB4) (Q11)	58	13	H	1Ø3-2
					(INHB5) (Q12)	58	1	H	"
					(INHB6) (Q9)	55	1	H	"
					(INHB7) (Q1Ø)	55	13	H	"
INH GATE B	26	6	H	1Ø3-1	(INHB8) (Q7)	48	13	H	"
					(INHB9) (Q8)	48	1	H	"
					(INHB1Ø) (Q5)	45	1	H	"
					(INHB11) (Q6)	45	13	H	"
					(INHB12) (Q3)	39	13	H	"
					(INHB13) (Q4)	39	1	H	"
					(INHB14) (Q1)	37	1	H	"
					(INHB15) (Q2)	37	13	H	"
INHIBIT	13	8	H	88-1		(B3Ø)		H	1Ø3-1
					INH GATE A, B	41	9	H	"
					WRITE MEM	41	2	H	"
INHIBIT SELECT *	(B85)				SELECT	35	5	L	1Ø3-1
INPUT* (F/F)	66	8		88-1	DRIVE IO*	12	1Ø	L	88-1
					(IO INST DCDR)	25	15	L	"
					MB LOAD	112	1,9	L	"
[INTA*]	6	6	L	88-1	INTA	5	9	L	"
INTA	5	8	H	"		(A4Ø)		H	
INTR*	(B29)		L		PI SET	75	12	L	88-2
INH TRANS*	56	6	L	88-1		(B45)		L	
					[INH TRANS.SEL]	36	2,5,4	H	1Ø3-1

* Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
[INH TRANS. SEL]	36	6	H	1Ø3-1	MEMØ*	16	2	H	1Ø3-1
					MEM1*	16	4	H	"
					MEM2*	14	2	H	"
					MEM3*	14	4	H	"
					MEM4*	12	2	H	"
					MEM5*	12	4	H	"
					MEM6*	1Ø	2	H	"
					MEM7*	1Ø	4	H	"
					MEM8*	8	2	H	"
					MEM9*	8	4	H	"
					MEM1Ø*	6	2	H	"
					MEM11*	6	4	H	"
					MEM12*	4	2	H	"
					MEM13*	4	4	H	"
					MEM14*	2	2	H	"
					MEM15*	2	4	H	"
IO. E	42	5	H	88-1	IO. E*	94	3	L	88-1
					(IO INST DCDR)	64	4	H	"
					(IO DCDR)	62	2	H	"
					HALT*	71	1	H	88-2
					LOOP SET*	86	9	H	88-3
					(PACK LOGIC)	89	4	H	88-3
IO. E*	94	4	L	88-1	MA LOAD*	6Ø	13	H	88-1
[IO(F+D)*]	51	12	L	88-2	IO(F+D)	27	5	L	88-2
IO(F+D)	27	6	H	"	INPUT F/F Logic	9	12	H	88-1
					IO. E	42	3	H	88-1
ION	82	6	H	"	(SKIP LOGIC)	11	13	(OC)	88-3
					ION*	84	5	L	88-2
ION*	84	6	L	88-2	(CON IND)	(A16)	(P26)	L	89-1
					ION	82	5	L	88-2
					(ION LOGIC)	85	5	L	"
[IO PLS*]	63	4	L	88-1	IO PLS	26	3	L	88-1
IO PLS	26	4	H	"		(A74)		H	
IORST	1Ø	8	H	"		(A7Ø)		H	
IO SKIP*	25	12	L	"	IO SKIP	26	1	L	88-1
					SKIP INC*	87	1	L	"
IO SKIP	26	2	H	88-1	(SKIP LOGIC)	59	5	H	88-3
IRØ *	28	5	L	88-2	(RUN LOGIC)	43	1Ø	H	88-1
					ACD OUT*	45	3	H	88-2
					SH/SWP DCDR	5Ø	13	H	"
					"	51	1	L	"
					PC ENAB*	53	4,5	L	88-3
					(PACK LOGIC)	92	4	H	"
* Indicates "NOT"									

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
IR1*	29	7	L	88-2	ACS 1 SEL*	49	5	H	88-2
					SH/SWP DCDR	51	13	H	"
					INST DCDR	52	13	H	"
IR2*	29	9	L	88-2	CON DATA*	24	2	H	88-1
					ACS 2 SEL*	49	1	H	88-2
					SH/SWP DCDR	51	14	H	"
					INST DCDR	52	14	H	"
IR3	29	12	H	88-2	(JMP+JSR)(F+D)	48	12	L	88-2
					ACS1 SEL*	48	9	H	"
IR3*	29	11	L	88-2	IR(DS)	12		L	88-2
IR4*	29	5	L	88-2	IR4	26	13	L	88-2
					ACD 4 SEL*	44	13	H	"
					ACS 2 SEL*	49	13	H	"
					INST DCDR	52	2	H	"
IR4	26	12	H	88-2	JSR. EFA*	93	1	H	"
IR5*	28	7	L	88-2	IO INST DCDR	25	3	H	88-1
					"	25	13	H	"
					IR5	26	11	L	88-2
					DISABLE D Mult	46	9	H	"
					AND ENAB*	65	1	L	"
					LOAD MBO*	98	1	H	88-3
IR5	26	1Ø	H	88-2	D SET	74	5	H	88-2
					(PACK LOGIC)	87	5	H	88-3
IR6*	28	9	L	88-2	IO INST DCDR	25	2	H	88-1
					"	25	14	H	"
					IR6	26	9	L	88-2
					ACD OUT*	45	9	H	88-2
					DISABLE D Mult	46	6	H	"
					KEYM SET*	55	4	H	88-1
					AND ENAB*	65	2	L	88-2
IR6	26	8	H	88-2	SØ	47	13	H	"
					S2	91	13	L	"
IR7*	28	11	L	88-2	ADD ONE*	44	4	L	88-2
					ACD 4 SEL*	44	1Ø	H	"
					DISABLE D Mult	46	5	H	"
					HALT*	71	5	H	"
					LOOP SET*	86	1Ø	H	88-3
IR7	28	12	H	88-2	INPUT (F/F)	9	13	H	88-1
					ADD ONE*	44	3	L	88-2
					AND ENAB*	64	13	H	"
					(PACK LOGIC)	89	1	H	88-3

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
IR5. IR6	65	3	H	88-2	AND ENAB*	64	12	H	88-2
					HALT*	71	4	H	"
IRØ+SKIP	5Ø	12	L	88-2	ALC*	5Ø	11	H	"
					(SH/SWP DCDR)	51	15	L	"
ISTP*	(A17)	(P24)	L	89-1	(RUN LOGIC)	24	9	L	88-1
(ISZ+DSZ)E	84	8	H	88-3	CRY SET*	81	9, 1Ø	H	88-3
					LOOP SET*	1Ø4	2	H	"
(ISZ+DSZ)E*	52	9	L	88-2	(INST DCDR)	52	1	L	88-2
					(ISZ+DSZ)E	84	13	L	88-3
					TEST SKIP SET	86	1	L	"
ISZ. E. TSØ*	52	5	L	88-2	ADD ONE*	89	9	L	88-2
(JMP+JSE)(F+D)	48	11	H	"	PC ENAB*	61	3	H	88-3
					JSR. EFA*	93	2	H	88-2
JSR. EFA	92	11	H	88-3	SHIFT ACB	1ØØ	1	L	88-3
					WAS JSR	1Ø3	3	H	"
JSR. EFA*	93	12	L	88-1	JSR. EFA	92	13	L	"
					(PACK LOGIC)	99	2	L	"
KEY	23	5	H	88-1	KEY*	6	1	L	88-1
					LOAD IR	34	1Ø	H	88-2
					CON INST*	36	9	H	88-1
					(RUN LOGIC)	43	9	H	"
					KEYM SET*	55	5	H	"
					DISABLE D Mult	46	1	H	88-2
					LOAD PC*	61	5	H	88-3
					KEYM SET*	55	5	H	88-1
KEY*	6	3	L	88-1	KEY. LOOP	4	2	L	88-1
					(DS)	23	4	H	"
					ADD ONE*	44	5	L	88-2
					INH TRANS*	56	4	L	88-1
					MA LOAD*	56	9	H	"
					(PACK LOGIC)	7Ø	13	L	88-3
					LOOP SET*	84	1Ø	L	"
					CLR SKIP*	99	1Ø	L	"
KEY ENAB*	3	3	L	88-1	PRESET*	3	12	L	88-1
						6	2	L	"
KEY. LOOP	4	3	H	88-1	CON DATA*	4	1Ø	H	88-1
					ACD OUT*	45	2	H	88-2
					LOAD MBO*	98	13	H	88-3
KEYM	23	11	H	88-1	CON DATA*	24	3	H	88-1
					(RUN LOGIC)	43	2	H	"
					ADD ONE*	41	1	H	88-2
KEYM*	23	12	L	88-1	KEYM. PL	41	1Ø	L	88-1

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
KEYM. PL	41	8	H	88-1	KEYM SET*	55	3	H	88-1
					KEYM. PL. TSØ*	57	2	H	88-3
					JSR. EFA	93	9	H	"
					LOAD MBO*	98	5	H	"
KEYM. PL. TSØ*	57	3	L	88-3	INH TRANS*	56	5	L	88-1
					LOAD PC*	57	4	L	88-3
KEYM SET*	55	6	L	88-1	[KEYM SET]	22	1	L	88-1
					FETCH	97	2	H	88-2
[KEYM SET]	22	2	H	88-1	KEYM	23	14	H	88-1
KEY SEEN*(F/F)	2	6	L	"	(RUN LOGIC)	21	1	L	"
					(MR)	54	1	L	"
					(MR)	1Ø2	1	L	88-3
KEY SEEN (F/F)	2	5	H	88-1	KEY ENAB*	3	1	H	88-1
					(SH)	23	13	H	"
LDA. E*	52	1Ø	L	88-2	(PACK LOGIC)	99	1	L	88-3
LOAD AC*	93	6	L	88-3		(A77)		L	
					ACD	123	3	L	88-4
					ACS	124	3	L	"
LOAD ACB	1ØØ	11	H	88-3	SHIFT ACB	1ØØ	2	L	88-3
					ACB (LD)	1Ø5	1Ø	H	88-4
					ACB (LD)	1Ø6	1Ø	H	"
					ACB (LD)	1Ø7	1Ø	H	"
					ACB (LD)	1Ø8	1Ø	H	"
LOAD CRY*	97	8	L	88-3	CARRY	76	11	H	88-3
					(PACK LOGIC)	99	5	L	"
LOAD IR	34	3	H	88-2		(A73)			
					IR(LD)	28	1Ø	H	88-2
					IR(LD) LOGIC	8	5	H	"
					MBC (LD)	32	1Ø	H	"
					MBC (LD)	33	1Ø	H	"
					[STUTTER]	54	15	H	88-1
LOAD MBO*	98	8	L	88-3	MBO(SH)	37	13	H	88-4
					MBO(SH)	38	13	H	"
					MBO(SH)	39	13	H	"
					MBO(SH)	4Ø	13	H	"
LOAD PC*	57	8	L	88-3	PC	119	12	L	"
LOOP	1Ø3	7	H	"	MB LOAD	13	12	H	88-1
					LOOP*	22	5	H	88-3
					SØ	47	9	H	88-2
					(IO INST DCDR)	64	5	H	88-1
					PTG2. $\overline{\text{LOOP}}$	7Ø	4	L	"
LOOP*	22	6	L	88-3	PC IN*	35	5	H	88-2

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
					CON INST*	36	1Ø	H	88-1
					MA LOAD*	56	13	H	"
LOOP SET	83	2	H	88-3	(TS 3/TS Ø)	65	9	L	"
					PTG-5	7Ø	1Ø	L	"
					LOOP	1Ø3	2	H	88-3
LOOP SET*	1Ø4	8	L	88-3	DCHA SET*	71	12	H	88-1
					LOOP SET	83	1	L	88-3
MA1	33	15		1Ø3-1	[SARD1](Jumper)	35	4		1Ø3-1
MA1*	33	14		"	[SARD1] "	35	4		"
MA2	33	1Ø		"	[SARD2] "	35	1		"
MA2*	33	11		"	[SARD2] "	35	1		"
MA3	33	9		"	[SARD2] "	35	2		"
MA3*	33	8		"	[SARD3] "	35	2		"
MA4	29	16		"	MA4 B*	67	3	H	1Ø3-4
MA4*	29	1		"					
MA4 B*	67	4		1Ø3-4	MA4 B	67	11	L	1Ø3-4
					Y ADDR DCDR	52	5,4		"
					"	66	5,4		"
MA4 B	67	1Ø		1Ø3-4	"	54	5,4		"
					"	62	5,4		"
MA5	29	15		1Ø3-1					
MA5*	29	14		"	MA5 B	67	5	H	1Ø3-4
MA5 B	67	6		1Ø3-4	MA5 B*	67	9	L	"
					Y ADDR DCDR	54	7		"
					"	62	7		"
					"	52	7		"
					"	66	7		"
MA5 B*	67	8	H	1Ø3-4	Y ADDR DCDR	54	1		1Ø3-4
					"	62	1		"
					"	52	1		"
					"	66	1		"
MA6	29	1Ø	H	1Ø3-1	MA6 B*	67	1	H	"
MA6*	29	11	L	"					
MA6 B*	67	2	L	1Ø3-4	MA6 B	67	13	L	1Ø3-4
					Y ADDR DCDR	62	6		"
					"	66	6		"
MA6 B	67	12	H	1Ø3-4	"	54	6		"
					"	52	6		"
MA7	29	9	H	1Ø3-1	MA7 B*	44	11	H	"
MA7*	29	8	L	"					
MA7 B*	44	1Ø	L	1Ø3-4	MA7 B	44	3	L	1Ø3-4
* Indicates "NOT"									

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
					Y ADDR DCDR	6Ø	5,4		1Ø3-4
					"	5Ø	5,4		"
MA7 B	44	4	H	1Ø3-4	"	57	5,4		
					"	47	5,4		
MA8	25	16	H	1Ø3-1	MA8 B*	44	9	H	1Ø3-4
MA8*	25	1	L	"					
MA8 B*	44	8	L	1Ø3-4	MA8 B	44	5	L	1Ø3-4
					Y ADDR DCDR	6Ø	7		"
					"	5Ø	7		"
					"	57	7		"
					"	47	7		"
MA8 B	44	6	H	1Ø3-4	"	6Ø	1		"
					"	5Ø	1		"
					"	57	1		"
					"	47	1		"
MA9	25	15	H	1Ø3-1	MA9 B*	44	13	H	"
MA9*	25	14	L	"					
MA9 B*	44	12	L	1Ø3-4	MA9 B	44	1	L	1Ø3-4
					Y ADDR DCDR	6Ø	6		"
					"	57	6		"
MA9 B	44	2	H	1Ø3-4	"	5Ø	6		"
					"	47	6		"
MA1Ø	25	1Ø	H	1Ø3-1	MA1Ø B*	71	3	H	"
MA1Ø*	25	11	L	"					
MA1Ø B*	71	4	L	1Ø3-3	MA1ØB	71	11	L	1Ø3-3
					X ADDR DCDR	73	5,4		"
					"	77	5,4		"
MA1Ø B	71	1Ø	H	1Ø3-3	"	72	5,4		"
					"	76	5,4		"
MA11	25	9	H	1Ø3-1					
MA11*	25	8	L	"	MA11 B	71	5	H	1Ø3-3
MA11 B	71	6	L	1Ø3-3	MA11 B*	71	9		"
					X ADDR DCDR	72	7		"
					"	76	7		"
					"	73	7		"
					"	77	7		"
MA11 B*	71	8	H	1Ø3-3	"	72	1		"
					"	76	1		"
					"	73	1		"
					"	77	1		"
MA12	22	16	H	1Ø3-1	MA12 B*	71	1	H	"
MA12*	22	1	L	"					

* Indicates "NOT"

NOVA 1200 SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
MA12 B*	71	2	L	1Ø3-3	MA12 B	71	13	L	1Ø3-3
					X ADDR DCDR	76	6		"
					"	77	6		"
MA12 B	71	12	H	1Ø3-3	"	72	6		"
					"	73	6		"
MA13	22	15	H	1Ø3-1	MA13 B*	8Ø	11	H	"
MA13*	22	14	L	"					
MA13 B*	8Ø	1Ø	L	1Ø3-3	MA13 B	8Ø	3	L	1Ø3-3
					X ADDR DCDR	79	5,4		"
					"	74	5,4		"
MA13 B	8Ø	4	H	1Ø3-3	X ADDR DCDR	78	5,4		1Ø3-3
					"	75	5,4		"
MA14	22	1Ø	H	1Ø3-1	MA14 B*	8Ø	9	H	"
MA14*	22	11	L	"					
MA14 B*	8Ø	8	L	1Ø3-3	MA14 B	8Ø	5	L	1Ø3-3
					X ADDR DCDR	79	7		"
					"	74	7		"
					"	78	7		"
					"	75	7		"
MA14 B	8Ø	6	H	1Ø3-3	"	79	1		"
					"	74	1		"
					"	78	1		"
					"	75	1		"
MA15	22	9	H	1Ø3-1	MA15 B*	84	13	H	"
MA15*	22	8	L	"					
MA15 B*	8Ø	12	L	1Ø3-3	MA15 B	8Ø	1	L	1Ø3-3
					X ADDR DCDR	79	6		"
					"	78	6		"
MA15 B	8Ø	2	H	1Ø3-3	"	74	6		"
					"	75	6		"
MA LOAD*	6Ø	8	L	88-1		(B7)		L	
					MTG (SH)	35	11	L	88-1
					[MA LOAD]	3Ø	9,1Ø	L	1Ø3-1
					"	3Ø	12,13	L	"
[MA LOAD]	3Ø	8	L	1Ø3-1	MA1-3	33	13	L	1Ø3-1
						33	4	L	"
					MA4-7	29	13	L	"
						29	4	L	"
					MA8-11	25	13	L	"
						25	4	L	"
					MA12-15	22	13	L	"
						22	4	L	"

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
MBC8*	33	5	L	88-2	(SKIP LOGIC)	11	9	L	88-3
					MBC8	27	1	L	88-2
					MBC (DS)	33	4	H	"
					(SH/SWP DCDR)	51	3		"
					(IO DCDR)	63	3		88-1
					"	63	13		"
MBC8	27	2	H	88-2	(SKIP LOGIC)	11	5	L	88-3
					SØ	47	5	H	88-2
MBC9*	32	5	L	88-2	(SH/SWP DCDR)	51	2		"
					(IO DCDR)	63	2		88-1
					"	63	14		"
					MBC9	79	9,1Ø	L	88-2
MBC9	79	8	H	88-2	(SKIP LOGIC)	8Ø	1	H	88-3
MBC1Ø*	33	9	L	88-2	MBC1Ø	27	9	L	88-2
					CRY ENAB	77	2	L	88-3
MBC1Ø	27	8	H	88-2	CPU INST*	9	5	H	88-2
					DSØ*	8	9	H	88-1
MBC11*	32	9	L	"	MBC11	27	13	L	88-2
MBC11	27	12	H	"	DS1*	8	11	H	88-1
					CPU INST*	9	4	H	88-2
					CRY ENAB	77	5	L	88-3
MBC12*	33	7	L	88-2	MBC12	27	11	L	88-2
MBC12	27	1Ø	H	"	DS2*	8	13	H	88-1
					CPU INST*	9	2	H	88-2
					LOAD CRY*	1Ø1	1	L	88-3
					S MULT	116	3		88-4
MBC13*	32	7	L	88-2	MBC13	27	3	L	88-2
MBC13	27	4	H	"	DS3*	22	9	H	88-1
					CPU INST*	9	1	H	88-2
					(SKIP LOGIC)	77	1	H	88-3
					S MULT	116	6		88-4
MBC14*	33	11	L	88-2					
MBC14	33	12	H	"	DS4*	8	3	H	88-1
					CPU INST*	1Ø	1	H	88-2
					(SKIP LOGIC)	77	1Ø	H	88-3
					S MULT	116	13		88-4
MBC15*	32	11	L	88-2	(SKIP LOGIC)	8Ø	4	H	88-3
MBC15	32	12	H	"	DS5*	8	1	H	88-1
					CPU INST*	1Ø	2	H	88-2
					S MULT	116	1Ø		88-4
MB CLR*	19	6	L	88-1		(B86)			
					[MB CLEAR]	118	9	L	1Ø3-1

* Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
[MB CLEAR]	18	8	H	1Ø3-1		3Ø	2,4,5	H	1Ø3-1
[MB CLEAR. SEL]	3Ø	6	L	"	INHØ F/F	34	13	L	"
					INH1 F/F	34	1	L	"
					INH2 F/F	32	1	L	"
					INH3 F/F	32	13	L	"
					INH4 F/F	31	13	L	"
					INH5 F/F	31	1	L	"
					INH6 F/F	28	1	L	"
					INH7 F/F	28	13	L	"
					INH8 F/F	27	13	L	"
					INH9 F/F	27	1	L	"
					INH1Ø F/F	24	1	L	"
					INH11 F/F	24	13	L	"
					INH12 F/F	23	13	L	"
					INH13 F/F	23	1	L	"
					INH14 F/F	21	1	L	"
					INH15 F/F	21	13	L	"
MB LOAD	14	6	H	88-1		(B74)		H	
						36	9	H	1Ø3-1
[MB LOAD. SEL]	36	8	H	1Ø3-1	INHØ F/F	34	11	H	"
					INH1 F/F	34	3	H	"
					INH2 F/F	32	3	H	"
					INH3 F/F	32	11	H	"
					INH4 F/F	31	11	H	"
					INH5 F/F	31	3	H	"
					INH6 F/F	28	3	H	"
					INH7 F/F	28	11	H	"
					INH8 F/F	27	11	H	"
					INH9 F/F	27	3	H	"
					INH1Ø F/F	24	3	H	"
					INH11 F/F	24	11	H	"
					INH12 F/F	23	11	H	"
					INH13 F/F	23	3	H	"
					INH14 F/F	21	3	H	"
					INH15 F/F	21	11	H	"
MBOØ*	4Ø	5		88-4		(B79)			
					[MDØ]	17	9		1Ø3-1
MBO1*	39	5		88-4		(B77)			
					MD1	17	5		1Ø3-1
					(CON IND) (P14)	7	11		89-1
MBO2*	37	5		88-4		(B44)			

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
					MD2	15	9		1Ø3-1
					(CON IND) (P15)	7	5		89-1
MBO3*	38	5		88-4		(B43)			
					MD3	15	5		1Ø3-1
					(CON IND) (P38)	8	9		89-1
MBO4*	4Ø	7		88-4		(B42)			
					MD4	13	9		1Ø3-1
					(CON IND) (P16)	8	11		89-1
MBO5*	39	7		88-4		(B32)			
					MD5	13	5		1Ø3-1
					(CON IND) (P11)	8	5		89-1
MBO6*	37	7		88-4		(B16)			
					MD6	11	9		1Ø3-1
					(CON IND) (P35)	9	9		89-1
MBO7*	38	7		88-4		(B14)			
					MD7	11	5		1Ø3-1
					(CON IND) (P9)	9	11		89-1
MBO8*	4Ø	9		88-4		(B12)			
					MBO12 SAVE*	42	2		88-1
					MD8	9	9		1Ø3-1
					(CON IND) (P18)	9	5		89-1
MBO9*	39	9		88-4		(B9)			
					MD9	9	5		1Ø3-1
					(CON IND) (P8)	1Ø	9		89-1
MBO1Ø*	37	9		88-4					
					MD1Ø	7	9		1Ø3-1
					(CON IND) (P44)	1Ø	11		89-1
MBO11*	38	9		88-4		(B5)			
					MD11	7	5		1Ø3-1
					(CON IND) (P6)	1Ø	5		89-1
MBO12*	4Ø	11		88-4		(A39)			
					MD12	5	9		1Ø3-1
					(CON IND) (P3Ø)	11	9		89-1
MBO12	4Ø	12		88-4	D MULT	121	3		88-4
					MULT	12Ø	6		"
					ADDER TEST	57	12	L	88-3
MBO13*	39	11		88-4		A37)			
					ADDER TEST	6Ø	5	L	88-3
					MD13	5	5		1Ø3-1
					(CON IND) (P4)	11	11		89-1
MBO13	39	12		88-4	D MULT	121	6		88-4
					MULT	12Ø	3		"
MBO14*	37	11		88-4		(A43)			

* Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
					ADDER TEST	6Ø	4	L	88-3
					MD14	3	9		1Ø3-1
					(CON IND) (P12)	11	5		89-1
MBO14	37	12		88-4	D MULT	121	13		88-4
					MULT	12Ø	21		"
MBO15*	38	11		88-4		(A41)			
					MD15	3	5		1Ø3-1
					(CON IND) (P28)	11	1		89-1
MBO15	38	12		88-4	ADDER TEST	84	9	H	88-3
					D MULT	12Ø	1Ø		88-4
					MULT	121	18		"
MBO12 SAVE*	42	7		88-1	SØ	48	2		88-2
					ADD ONE*	9Ø	5		"
[MDØ]	17	8			INHØ	34	12		1Ø3-1
MD1	17	6		1Ø3-1	INH1	34	2		1Ø3-1
					MA1	33	3		"
MD2	15	8		1Ø3-1	INH2	32	2		"
					MA2	33	6		"
MD3	15	6		1Ø3-1	INH3	32	12		"
					MA3	33	7		"
MD4	13	8		1Ø3-1	INH4	31	12		"
					MA4	29	2		"
MD5	13	6		1Ø3-1	INH5	31	2		"
					MA5	39	3		"
MD6	11	8		1Ø3-1	INH6	28	2		"
					MA6	29	6		"
MD7	11	6		1Ø3-1	INH7	28	12		"
					MA7	29	7		"
MD8	9	8		1Ø3-1	INH8	27	12		"
					MA8	25	2		"
MD9	9	6		1Ø3-1	INH9	27	2		"
					MA9	25	3		"
MD1Ø	7	8		1Ø3-1	INH1Ø	24	2		"
					MA1Ø	25	6		"
MD11	7	6		1Ø3-1	INH11	24	12		"
					MA11	25	7		"
MD12	5	8		1Ø3-1	INH12	23	12		"
					MA12	22	2		"
MD13	5	6		1Ø3-1	INH13	23	2		"
					MA13	22	3		"
MD14	3	8		1Ø3-1	INH14	21	2		"
					MA14	22	6		"
MD15	3	6		1Ø3-1	INH15	21	12		"
*Indicates "NOT"									

NOVA 1200 SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
					MA15	22	7		103-1
MD SEL1*	(A87)		L		ACS1 SEL*	49	6,8	(OC)	88-2
MEM0*	16	3	(OC)	103-1		(B71)			
(ACEX+ACDP)	1	3	(OC)	89-1	(CON IND) (P39)	7	9	L	89-1
					IR0*	28	3		88-2
					MBO0*	40	3		88-4
					DEFER AGAIN	76	2	L	88-2
					(EFA LOGIC)	55	10,13	L	88-3
MEM1*	16	6	(OC)	103-1		(B70)			
(ACDP)	1	6	(OC)	89-1	(CON IND) (P41)	7	13	L	89-1
					IR1*	29	2		88-2
					MBO1*	39	3		88-4
					(EFA LOGIC)	55	9	L	88-3
MEM2*	14	3	(OC)	103-1		(B47)			
(DP+DPN)	2	8	(OC)	89-1	(CON IND) (P13)	7	3	L	89-1
					IR2*	29	15		88-2
					MBO2*	37	3		88-4
					(EFA LOGIC)	55	1	L	88-3
MEM3*	14	6	(OC)	103-1					
(ACEX+ACDP)	1	11	(OC)	89-1		(B68)			
					(CON IND) (P43)	7	1	L	89-1
					IR3*	29	14		88-2
					MBO3*	38	3		88-4
MEM4*	12	3	(OC)	103-1		(B28)			
(ACEX+ACDP)	1	8	(OC)	89-1	(CON IND) (P37)	8	13	L	89-1
					IR4*	29	3		88-2
					MBO4*	40	2		88-4
MEM5*	12	6	(OC)	103-1		(B26)			
(EX+STRT+ACDP)	2	11	(OC)	89-1	(CON IND) (P36)	8	3	L	89-1
					IR5*	28	2		88-2
					MBO5*	39	2		88-4
MEM6*	10	3	(OC)	103-1		(B22)			
(EX+EXN+DP+DPN)	2	3	(OC)	89-1	(CON IND) (P10)	8	1	L	89-1
					IR6*	28	15		88-2
					MBO6*	37	2		88-4
MEM7*	10	6	(OC)	103-1		(B24)			
(EXN+DPN)	2	6	(OC)	89-1	(CON IND) (P42)	9	13	L	89-1
					IR7*	28	14		88-2
					MBO7*	38	2		88-4
MEM8*	8	3	(OC)	103-1		(A55)			
					(CON IND) (P34)	9	3	L	89-1
					MBC8*	33	3		88-2
					MBO8*	40	15		88-4

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
MEM9*	8	6	(OC)	1Ø3-1		(A53)			
					(CON IND) (P7)	9	1	L	89-1
					MBC9*	32	3		88-2
					MBO9*	39	15		88-4
MEM1Ø*	6	3	(OC)	1Ø3-1		(A45)			
					(CON IND) (P32)	1Ø	13	L	89-1
					MBC1Ø*	33	15		88-2
					MBO1Ø*	37	15		88-4
MEM11*	6	6	(OC)	1Ø3-1		(A51)			
					(CON IND) (P31)	1Ø	3	L	89-1
					MBC11*	32	15		88-2
					MBO11*	38	15		88-4
MEM12*	4	3	(OC)	1Ø3-1		(A36)			
					(CON IND) (P5)	1Ø	1	L	89-1
					MBC12*	33	2		88-2
					MBO12*	4Ø	14		88-4
MEM13*	4	6	(OC)	1Ø3-1		(A35)			
					(CON IND) (P29)	11	13	L	89-1
					MBC13*	32	2		88-2
					MBO13*	39	14		88-4
MEM14*	2	3	(OC)	1Ø3-1		(B76)			
					(CON IND) (P3)	11	3	L	89-1
					MBC14*	33	14		88-2
					MBO14*	37	14		88-4
MEM15*	2	6	(OC)	1Ø3-1		(B18)			
					(CON IND) (P2)	12	13	L	89-1
					MBC15*	32	14		88-2
					MBO15*	38	14		88-4
MEM CLK	73	6		88-1		(B48)			
					(MTG)	17	6	L	88-1
					(KEY/RUN/DCH)	23	6	L	"
					(ACTG)	54	6	L	"
					LOAD AC*	93	4	H	88-3
					S BUFF	115	6	L	88-4
					D BUFF	122	6	L	"
					IR4, IR1-3	29	6	L	88-2
MEM OK	(A9)				RUN LOGIC	62	5		88-1
					IR(SH)	83	5		88-2
					SHIFTER	125	7		88-4
					LOAD AC*	111	3	L	88-3
[MSKO]	4	11	H	88-1	MSKO*	5	3	H	88-1
MSKO*	5	4	L	"		(A38)		L	
MSTP	(A2Ø)	(P48)	L	89-1	(RUN LOGIC)	24	1, 1Ø	L	88-1
* Indicates "NOT"									

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
MTGØ	17	5		88-1	INHIBIT	13	1Ø	L	88-1
					DCHI	14	13	H	"
					MTGØ*	16	13	H	"
					MTG	17	2,15,14	H	"
					READ1*	19	1	H	"
					MB CLR*	19	4	H	"
MTGØ*	16	12		88-1	MTG(SH)(Logic)	36	4	H	"
MTG1	17	7		88-1	RQENB*	16	5	H	88-1
					MTG1*	16	11	H	"
					READ2*	19	1Ø	H	"
MTG1*	16	1Ø		88-1	MB CLR*	19	5	H	"
					DCHO	18	1Ø	H	"
MTG2	17	9		88-1	MTG2*	16	9	H	88-1
					STROBE	18	5	H	"
MTG2*	16	8		88-1	INHIBIT	13	9	L	"
MTG3	17	11		88-1	DCHO	18	9	H	"
					MTG(SH) (Logic)	36	5	H	"
MTG3*	17	12		88-1	MTG (DS)	17	4	H	"
					STROBE	18	1,2,4	H	"
					READ1*	19	2	H	"
					READ2*	19	9	H	"
MULTØ*	12Ø	1Ø		88-4	MBO (DS)	4Ø	4		88-4
MULT1*	12Ø	11		"	MBO (DS)	39	4		"
MULT2*	12Ø	13		"	MBO (DS)	37	4		"
MULT3*	12Ø	14		"	MBO (DS)	38	4		"
OVFLO	15	8	H	88-1		(B39)		H	
PACK	1Ø3	9	H	88-3	ACS1 SEL*	49	1Ø	H	88-2
					ACS2 SEL*	49	12	H	"
					PACK*	83	13	H	88-3
PACK*	83	12	L	88-3	ACS1 SEL*	49	4	H	88-2
					ACS2 SEL*	49	2	H	"
					LOAD AC*	111	5	L	88-3
[PCØ]	119	1Ø		88-4	MULTØ*	12Ø	1Ø		88-4
[PC1]	119	9		"	MULT1*	12Ø	11		"
[PC2]	119	7		"	MULT2*	12Ø	13		"
[PC3]	119	6		"	MULT3*	12Ø	14		"
PC ENAB*	61	8	L	88-3	PC IN*	36	1	H	88-2
					LOAD PC*	57	5	L	88-3
					E SET	74	2	H	88-2
PC IN*	36	3	L	88-2	PC	119	11	L	88-4

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
					MULT(ENAB)	12Ø	7,8,9	H	88-4
PI	95	11	H	88-2	PC IN*	35	4	H	88-2
					ADD ONE*	9Ø	3	H	"
					CLR SKIP*	1ØØ	5	H	88-3
					DISABLE D MULT	46	2,3	H	88-2
PI*	95	12	L	88-2	IR(SH)	114	2	L	88-2
					IR (DS)	12	13	L	"
					D SET	74	11	L	"
					ADD ONE*	82	13	L	"
					ION*	84	1	L	"
					LOOP SET*	84	12	L	88-3
PI SET	96	6	H	88-2	PI	95	14	H	"
					FETCH	96	9	L	"
					LOAD MBO*	98	2	H	88-3
PL*	(A19)	(P23)	L	89-1	KEYM. PL	41	9	L	88-1
					(RUN LOGIC)	43	3	H	"
					DISABLE D MULT	87	9	L	88-2
PRESENT*	22	1Ø	L	88-1	MTG (MR)	17	1	L	88-1
					INPUT	66	1	L	"
					PTG (MR)	69	1	L	"
					SKIP	78	1	L	88-3
					(MAJOR STATES)	95	1	L	88-2
PTGØ	69	9		88-1	PTG DCDR	68	2		88-1
					"	68	14		"
					PTG	69	14		"
					PC	119	4		88-4
					PC	119	13		"
					MB LOAD	112	13	L	88-1
PTG1	69	11		88-1	EFA. PTG1	34	4	L	88-2
					PTG DCDR	68	3		88-1
					"	68	13		"
					END CYCLE F/F	113	3		"
					PC	119	5		88-4
					PC	119	14		"
					MB LOAD	112	1Ø	L	88-1
PTG1*	69	12		88-1	SO	47	2	H	88-2
					PTG	69	15		88-1
PTG2*	68	1Ø		88-1	ADDER TEST	57	13	L	88-3
					TSØ/TS3	65	1Ø	L	88-1
					PTG2	67	9	L	"
					PTG2. LOOP	7Ø	5	L	"
PTG2	67	8	H	88-1	INPUT F/F	66	12		"

* Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
PTG5	7Ø	8	H	88-1	KEY/RUN/DCH(LD)	23	1Ø	H	88-1
					(LD)	42	1Ø	H	"
					TSØ/TS3 F/F	66	2		"
					ADDER TEST	78	12		88-3
					"	79	4	L	"
					MAJOR STATES(LD)	95	1Ø	H	88-2
					LOAD MBO*	98	3	H	88-3
					LOAD MBO*	98	4	H	"
PTG5 ENAB*	68	6	L	88-1	INH TRANS*	56	1	L	88-1
					PTG5	7Ø	9	L	"
					PACK LOGIC	7Ø	12	L	88-3
					SKIP F/F	79	12	L	"
					LOAD ACB	1ØØ	12	L	"
PTG=Ø. TSØ	113	9		88-1	ADDER TEST	58	13	L	"
					MA LOAD*	6Ø	12	H	88-1
					ADD ONE*	88	9	H	88-2
					SHIFTER Logic	9Ø	9	H	88-4
PTG=Ø. TSØ*	113	8		88-1	ADD ONE*	88	4	H	88-2
					SHIFTER Logic	9Ø	13	H	88-4
					SHIFT ACB	93	1Ø	H	88-3
PTG=Ø. TS3*	68	4	L	88-1	PTG=Ø. TS3	67	13	L	88-1
					ADD ONE*	88	5	H	88-2
PTG=Ø. TS3	67	12	H	88-1	INPUT F/F	9	9,1Ø	H	88-1
					MTG (LD)	17	1Ø	H	"
					ADD ONE*	88	1,2	H	88-2
PTG=1. TSØ*	68	11	L	88-1	ADDER TEST	8Ø	1Ø	H	88-3
					SHIFT ACB	93	11	H	"
PTG=1. TS3*	68	5	L	88-1	(IO DCDR)	1Ø9	5	L	88-1
PTG2. LOOP	7Ø	6	H	88-1	PTG2+LOOP	73	13	H	88-1
					LOAD MBO*	98	9	H	88-3
					LOOP SET*	1Ø4	3,4	H	"
PTG2+LOOP	73	12	L	88-1	LOAD IR	34	1	L	88-2
					SKIP (F/F)	79	2	L	88-3
PULSE ENAB	1Ø9	6	H	88-1	OVFLO	15	12	H	88-1
					IO DCDR	62	1	H	"
PWR FAIL*	(A5)		L		PWR LOW	86	12	L	88-3
PWR LOW	1Ø2	11	H	88-3	(SKIP Logic)	11	1	H	"
PWR LOW*	1Ø2	12	L	"	PI SET	75	13	L	88-2
					PWR LOW	86	13	L	88-3
READ1*	19	3	L	88-1		(B87)		L	1Ø3-1
					MTG(SH)	35	1Ø	L	88-1
					READ 1B	18	13	L	1Ø3-1
	18	12	H	1Ø3-1	"	19	5,4	H	"

*Indicates "NOT"

NOVA 1200 SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
					READ 2B	19	12	H	103-1
READ 1B	19	6	H	103-1	"	19	10	H	"
					(C ADDR DCDR)	72	2		103-3
					"	76	2		"
					"	73	2		"
					"	77	2		"
					"	79	3		"
					"	74	3		"
					"	78	3		"
					"	75	3		"
READ 2*	19	8	L	88-1		(B90)		L	103-1
					READ 2B	18	11	L	"
	18	10	H	103-1	"	19	9	H	"
						19	13	H	"
READ 2B	19	8	H	103-2	(Y ADDR DCDR)	54	2		103-4
					"	62	2		"
					"	52	2		"
					"	66	2		"
					"	60	3		"
					"	50	3		"
					"	57	3		"
					"	47	3		"
READ IO*	12	3	L	88-1		(B83)			103-1
					[READ IO]	18	3	L	"
	18	4	H	103-1	[MD0]	17	13	H	"
					MD1	17	2	H	"
					MD2	15	13	H	"
					MD3	15	2	H	"
					MD4	13	13	H	"
					MD5	13	2	H	"
					MD6	11	13	H	"
					MD7	11	2	H	"
					MD8	9	13	H	"
					MD9	9	2	H	"
					MD10	7	13	H	"
					MD11	7	2	H	"
					MD12	5	13	H	"
					MD13	5	2	H	103-1
					MD14	3	13	H	"
					MD15	3	2	H	"
[READ IO]	18	6	L	103-1		18	5	H	"

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
					[MDØ]	17	1Ø	L	1Ø3-1
					MD1	17	4	L	"
					MD2	15	1Ø	L	"
					MD3	15	4	L	"
					MD4	13	1Ø	L	"
					MD5	13	4	L	"
					MD6	11	1Ø	L	"
					MD7	11	4	L	"
					MD8	9	1Ø	L	"
					MD9	9	4	L	"
					MD1Ø	7	1Ø	L	"
					MD11	7	4	L	"
					MD12	5	1Ø	L	"
					MD13	5	4	L	"
					MD14	3	1Ø	L	"
					MD15	3	4	L	"
RESET*	22	4	L	88-1	PRESET*	3	13	L	88-1
					IORST	1Ø	13	L	"
						21	9	H	"
					KEY/RUN/DCH (MR)	23	1	L	"
					(MR)	42	1	L	"
					ION*	84	2	L	88-2
					LOOP/PACK (MR)	1Ø3	1	L	88-3
RESTART*				88-1	KEY SEEN F/F	3	5	L	88-1
					DISABLE D MULT	87	1Ø	L	88-2
RELOAD Disable*	(B72)					36	10,12		1Ø3-1
RESTART Enable	(A32)	(P19)		89-1					
RINHØ	(A5)			1Ø3-2					
RINH1	(A7)			"					
RINH2	(A9)			"					
RINH3	(A11)			"					
RINH4	(A13)			"					
RINH5	(A15)			"					
RINH6	(A18)			"					
RINH7	(A17)			"					
RINH8	(A19)			"					
RINH9	(A24)			"					
RINH1Ø	(A23)			"					
RINH11	(A21)			"					
RINH12	(A28)			"					
RINH13	(A25)			"					
RINH 14	(A29)			"					
*Indicates "NOT"									

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
RIN15	(A27)			1Ø3-2					
RQENB*	16	6	L	88-1		(B41)			
RST*	(A3Ø)	(P2Ø)		89-1	RESET*	21	12	L	88-1
RUN	23	7		88-1	RUN*	22	13	H	"
					CPU CLK	72	4,1Ø	H	"
RUN*	22	12	L	88-1	(CON IND) (A14)	12	1	L	89-1
					KEY SEEN F/F	2	1,2	L	88-1
SØ	92	3	H	88-2	ADDER	117	3,6		88-4
S1	91	8	H	"	"	117	5		"
S2	91	11	H	"	S1	91	4		88-1
					ADDER	117	4		88-4
[S BUFFRØ]	115	5		88-4	S MULT	116	2		"
[S BUFFR1]	115	7		"	"	116	5		"
[S BUFFR2]	115	9		"	"	116	14		"
[S BUFFR3]	115	11		"	"	116	11		"
SELB*	(A82)				SKIP Logic	11	1Ø	L	88-3
SELD*	(A8Ø)				"	11	4	L	"
SELECT	35	8	H	1Ø3-1	STRB A, B, C, D	1	1,1Ø		1Ø3-1
					"	1	12,13		"
					READ 1B	19	1, 2		"
					INH GATE A, B	26	2,4,5		"
					"	41	1Ø,12,13		"
					(DRIVE IO)	26	13		"
					(INH TRANS*)	36	1		"
					(MB LOAD)	36	13		"
					(MB CLEAR)	3Ø	1		"
SERIAL CRY	54	12	H	88-1	OVFLO	15	13	H	88-1
					ADD ONE*	88	6	H	88-2
SET ION*	63	1Ø	L	88-1	ION*	82	4	L	"
SHIFTØ*	125	13		88-4		(B94)			
					SKIP LOGIC	11Ø	12	L	88-3
					ACD	123	4		88-4
					ACS	124	4		"
SHIFT1*	125	14		88-4		(B96)			
					SKIP LOGIC	11Ø	1Ø		88-3
					ACD	123	6		88-4
					ACS	124	6		"
SHIFT2*	125	11		88-4		(B93)			
					SKIP LOGIC	11Ø	13		88-3
					ACD	123	1Ø		88-4
					ACS	124	1Ø		88-4
SHIFT3*	125	1Ø		88-4					

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
					SKIP LOGIC	11Ø	9		88-3
					ACD	123	12		88-4
					ACS	124	12		"
SHIFT ACB	1ØØ	3	H	88-3	ACB(SH)	1Ø5	13		"
					ACB(SH)	1Ø6	13		"
					ACB(SH)	1Ø7	13		"
					ACB(SH)	1Ø8	13		"
SHL*	51	6	L	88-2	CARRY F/F Logic	1Ø1	5	H	88-3
					[SHL]	1Ø1	3	L	"
					SHIFTER(SEL)	125	16		88-4
[SHL]	1Ø1	4	H	88-3	CRY SET*	81	2	H	88-3
SHR*	51	5	L	88-2	CARRY F/F Logic	81	6	H	"
					[SHR]	1Ø1	5	L	"
					SHIFTER (Sel)	125	17		88-4
[SHR]	1Ø1	6	H	88-3	CRY SET*	81	1		88-3
SKIP	78	5		"	ADD ONE*	9Ø	2	H	88-2
SKIP*	78	6		"		(B69)			
					IRØ+SKIP	5Ø	2	H	88-2
					D SET	74	3	H	"
					ADD ONE*	82	12	L	"
					TEST SKIP SET	86	4	H	88-3
SKIP INC*	42	12	L	88-1	PC IN*	35	13	L	88-2
					MA LOAD*	56	12	H	88-1
					PC ENAB*	58	4,5	L	88-3
					CLR SKIP*	99	12	L	"
+SLØ				1Ø3-2	SNSØ	69	2		1Ø3-2
-SLØ				"	"	69	3		"
+SL1				"	SNS1	69	6		"
-SL1				"	"	69	7		"
+SL2				"	SNS2	65	2		"
-SL2				"	"	65	3		"
+SL3				"	SNS3	65	6		"
-SL3				"	"	65	7		"
+SL4				"	SNS4	59	2		"
-SL4				"	"	59	3		"
+SL5				"	SNS5	59	6		"
-SL5				"	"	59	7		"
+SL6				"	SNS6	56	2		"
-SL6				"	"	56	3		"
+SL7				"	SNS7	56	6		"
-SL7				"	"	56	7		"

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
+SL8				1Ø3-2	SNS8	49	2		1Ø3-2
-SL8				"	"	49	3		"
+SL9				"	SNS9	49	6		"
-SL9					"	49	7		"
+SL1Ø					SNS1Ø	46	2		"
-SL1Ø					"	46	3		"
+SL11					SNS11	46	6		"
-SL11					"	46	7		"
+SL12					SNS12	4Ø	2		"
-SL12					"	4Ø	3		"
+SL13					SNS13	4Ø	6		"
-SL13					"	4Ø	7		"
+SL14					SNS14	38	2		"
-SL14					"	38	3		"
+SL15					SNS15	38	6		"
-SL15					"	38	7		"
[S MULTØ]	116	4		88-4	ADDER	117	18		88-4
[S MULT1]	116	7		"	ADDER	117	2Ø		"
[S MULT2]	116	12		"	"	117	22		"
[S MULT3]	116	9		"	"	117	1		"
SNSØ	69	14	H	1Ø3-2	SNSØ*	68	9	H	1Ø3-2
SNSØ*	68	8	L	"	INHØ F/F	34	1Ø	L	1Ø3-1
SNS1	69	12	H	"	SNS1*	68	5	H	1Ø3-2
SNS1*	68	6	L	"	INH1 F/F	34	4	L	1Ø3-1
SNS2	65	14	H	"	SNS2*	64	5	H	1Ø3-2
SNS2*	64	6	L	"	INH2 F/F	32	4	L	1Ø3-1
SNS3	65	12	H	"	SNS3*	64	9	H	1Ø3-2
SNS3*	64	8	L	"	INH3 F/F	32	1Ø	L	1Ø3-1
SNS4	59	14	H	"	SNS4*	58	9	H	1Ø3-2
SNS4*	58	8	H	"	INH4 F/F	31	1Ø	L	1Ø3-1
SNS5	59	12	H	"	SNS5*	58	5	H	1Ø3-1
SNS5*	58	6	L	"	INH5 F/F	31	4	L	1Ø3-1
SNS6	56	14	H	"	SNS6*	55	5	H	1Ø3-2
SNS6*	55	6	L	"	INH6 F/F	28	4	L	1Ø3-1
SNS7	56	12	H	1Ø3-2	SNS7*	55	9	H	1Ø3-2
SNS7*	55	8	L	"	INH7 F/F	28	1Ø	L	1Ø3-1
SNS8	49	14	H	"	SNS8*	48	9	H	1Ø3-2
SNS8*	48	8	L	"	INH8 F/F	27	1Ø	L	1Ø3-1
SNS9	49	12	H	"	SNS9*	48	5	H	1Ø3-2
SNS9*	48	6	L	"	INH8 F/F	27	4	L	1Ø3-1
SNS1Ø	46	14	H	"	SNS1Ø*	45	5	H	1Ø3-2
SNS1Ø*	45	6	L	"	INH1Ø F/F	24	4	L	1Ø3-1

*Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
SNS11	46	12	H	103-2	SNS11*	45	9	H	1Ø3-2
SNS11*	45	8	L	"	INH11 F/F	24	1Ø	L	1Ø3-1
SNS12	4Ø	14	H	"	SNS12*	39	9	H	1Ø3-2
SNS12*	39	8	L	"	INH12 F/F	23	1Ø	L	1Ø3-1
SNS13	4Ø	12	H	"	SNS13*	39	5	H	1Ø3-2
SNS13*	39	6	L	"	INH13 F/F	23	4	L	1Ø3-1
SNS14	38	14	H	"	SNS14*	37	5	H	1Ø3-2
SNS14*	37	6	L	"	INH14 F/F	21	4	L	1Ø3-1
SNS15	38	12	H	"	SNS15*	37	9	H	1Ø3-2
SNS15*	37	8	L	"	INH15 F/F	21	1Ø	L	1Ø3-1
STA. E	52	11	L	88-2	LOAD MBO*	99	9	L	88-3
					MULT (SEL)	12Ø	17		88-4
STOP*	(A31)	(P45)		89-1	STOP SYNC	4	4,5	L	88-3
STOP INH*	82	8	L	88-1	DCHA SET*	71	13	H	88-1
					SKIP INC*	87	2	L	"
					FETCH	97	4	H	88-2
STOP SYNC	1Ø2	5	H	88-3	RUN LOGIC	43	1	H	88-1
STROBE	18	6	H	88-1		(B2Ø)			
					STRB A, B, C, D	1	5	H	1Ø3-1
STRB A	1	6	H	1Ø3-1	SNSØ*	68	1Ø	H	1Ø3-2
					SNS1*	68	4	H	"
					SNS2*	64	4	H	"
					SNS3*	64	1Ø	H	"
STRB B	1	6	H	1Ø3-1	SNS4*	58	1Ø	H	"
					SNS5*	58	4	H	"
					SNS6*	55	4	H	"
					SNS7*	55	1Ø	H	"
STRB C	1	6	H	1Ø3-1	SNS8*	48	1Ø	H	"
					SNS9*	48	4	H	"
					SNS1Ø*	45	4	H	"
					SNS11*	45	1Ø	H	"
STRB D	1	6	H	1Ø3-1	SNS12*	39	1Ø	H	1Ø3-2
					SNS13*	39	4	H	"
					SNS14*	37	4	H	"
					SNS15*	37	1Ø	H	"
[STRT*]	63	6	L	88-1	STRT	7	3	L	88-1
STRT	7	4	H	"	(IO STRT PLS)	(A52)			
[STUTTER]	54	9	H	"	STUTTER*	73	1	H	88-1
STUTTER*	73	2	L	"	CPU CLK	72	1,13	H	"
SWP*	51	4	L	88-2	LOAD ACB	1ØØ	13	L	88-3

* Indicates "NOT"

NOVA 12ØØ SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
TSØ	66	6	H	88-1	PC IN	35	3	H	88-2
					IR(SH)	114	5	L	"
					INST DCDR	92	1Ø	H	"
					DISABLE D Mult	53	1	L	"
					KEYM. PL. TSØ*	57	1	H	88-3
					PC ENAB*	61	9	H	"
					FETCH. TSØ*	64	1Ø	H	88-2
					PTG DCDR	68	1	L	88-1
					S1	91	1Ø	L	88-2
TS3	66	5	H	88-1	LOOP SET*	34	13	L	88-3
					(D+E SET) +TS3	36	12	H	88-2
					ACD OUT*	45	4	H	"
					ALC*	5Ø	1Ø	H	"
					PC ENAB*	61	1	H	"
					PC ENAB*	61	2,4	H	88-3
					IO DCDR Logic	1Ø9	2	L	88-1
					PTG DCDR	68	15	L	88-1
					ACTG (LD)	75	9	H	"
					DEFER AGAIN(F/F)	76	3	H	88-2
TS3 SET	9	8	H	88-1	INPUT F/F	66	11	H	88-1
					PTG=Ø. TSØF/F	112		L	88-1
TEST*	(A92)				CARRY F/F	76	13	L	88-3
TEST SKIP	1Ø2	7	H	88-3	SKIP F/F Logic	59	3	H	"
TEST SKIP SET	86	3	H	"	RUN LOGIC	41	13	H	88-1
					STOP INH*	82	9	H	"
					TEST SKIP	1Ø2	2	H	88-3
WAS JSR	1Ø3	5	H	88-3	ACS1 SEL*	48	1Ø	H	88-2
					SHUTTER Logic	1Ø9	12	H	88-4
WAS JSR*	48	8	L	88-1		(A89)			
WHOA*	(B6)				CPU CLK	72	5,9	H	88-1
+5 OK	(A8)				RESET*	21	13	L	"
WRITE MEM	41	6	H	1Ø3-1	X DRIVERS	72	3	H	1Ø3-3
						76	3	H	"
						73	3	H	"
						77	3	H	"
						79	2	H	"
						74	2	H	"
						78	2	H	"
						75	2	H	"
					Y DRIVERS	54	3	H	1Ø3-4
						62	3	H	"
						52	3	H	"

*Indicates "NOT"

NOVA 1200 SIGNAL LIST-CENTRAL PROCESSOR AND MEMORY

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	LEVEL	DWG	FUNCTION	CHIP	PIN	LEVEL	DWG
					Y DRIVERS	66	3	H	103-4
					(Contd)	60	2	H	"
						50	2	H	"
XRS					X DRIVERS	72	11		103-3
					"	76	11		"
					"	73	11		"
					"	77	11		"
XWS					X DRIVERS	75	11		"
					"	78	11		"
					"	74	11		"
					"	79	11		"
YRS					Y DRIVERS	54	11		103-4
					"	62	11		"
					"	52	11		"
					"	66	11		"
YWS					Y DRIVERS	47	11		"
					"	57	11		"
					"	50	11		"
					"	60	11		"