Data General Corporation

Technical Manual

Nova 1220

015-000011-02

DATA GENERAL TECHNICAL MANUAL

NOVA 1220 COMPUTER

MODELS

8151, 8152, 8153, 8154
8155, 8156, 8157, 8158

Ordering No. 015-000011-02

rinted in the United States of America
Rev. 02 March 1973

NOTICE

DATA GENERAL CORPORATION (DGC) HAS PREPARED THIS MANUAL FOR INFORMATION PURPOSES ONLY. DGC RESERVES THE RIGHT TO MAKE CHANGES WITHOUT NOTICE IN THE SPECIFICATIONS AND MATERIALS CONTAINED HEREIN AND SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES CAUSED BY RELIANCE ON THE MATERIALS PRESENTED, INCLUDING BUT NOT LIMITED TO TYPOGRAPHICAL OR ARITHMETIC ERRORS, COMPANY POLICY AND PRICING INFORMATION.

TABLE OF CONTENTS

SECTION O

INTRODUCTION

SECTION C

THE CENTRAL PROCESSOR UNIT

SECTION K

THE OPERATOR'S CONSOLE

Page

SECTION P

THE POWER SUPPLY

SECTION M

THE MEMORY

SECTION I

INSTALLING THE COMPUTER

SECTION N

MAINTAINING THE COMPUTER

REFERENCE TABLES

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Continued)

LIST OF TABLES

SECTION O

INTRODUCTION

THE NOVA 1220 COMPUTER

The Nova 1220 computer shown in Figure O-1 consists of a power supply-backpanel assembly and a console assembly mounted on a chassis into which plug up to ten 15" by 15" PC boards. The chassis includes a frame, two fans, a filter, a power transformer and a power switch assembly; the power supply-backpanel includes the power supply and ten sets of edge connectors mounted on an etched PC board. The console includes a frame, front panel and PC board which holds the switches, lights and associated logic. Each basic Nova 1220 includes a Central Processor module, and any one of four types of memory modules; 1K, 2K, 4K or 8K. A table top assembly is also available but not shown.

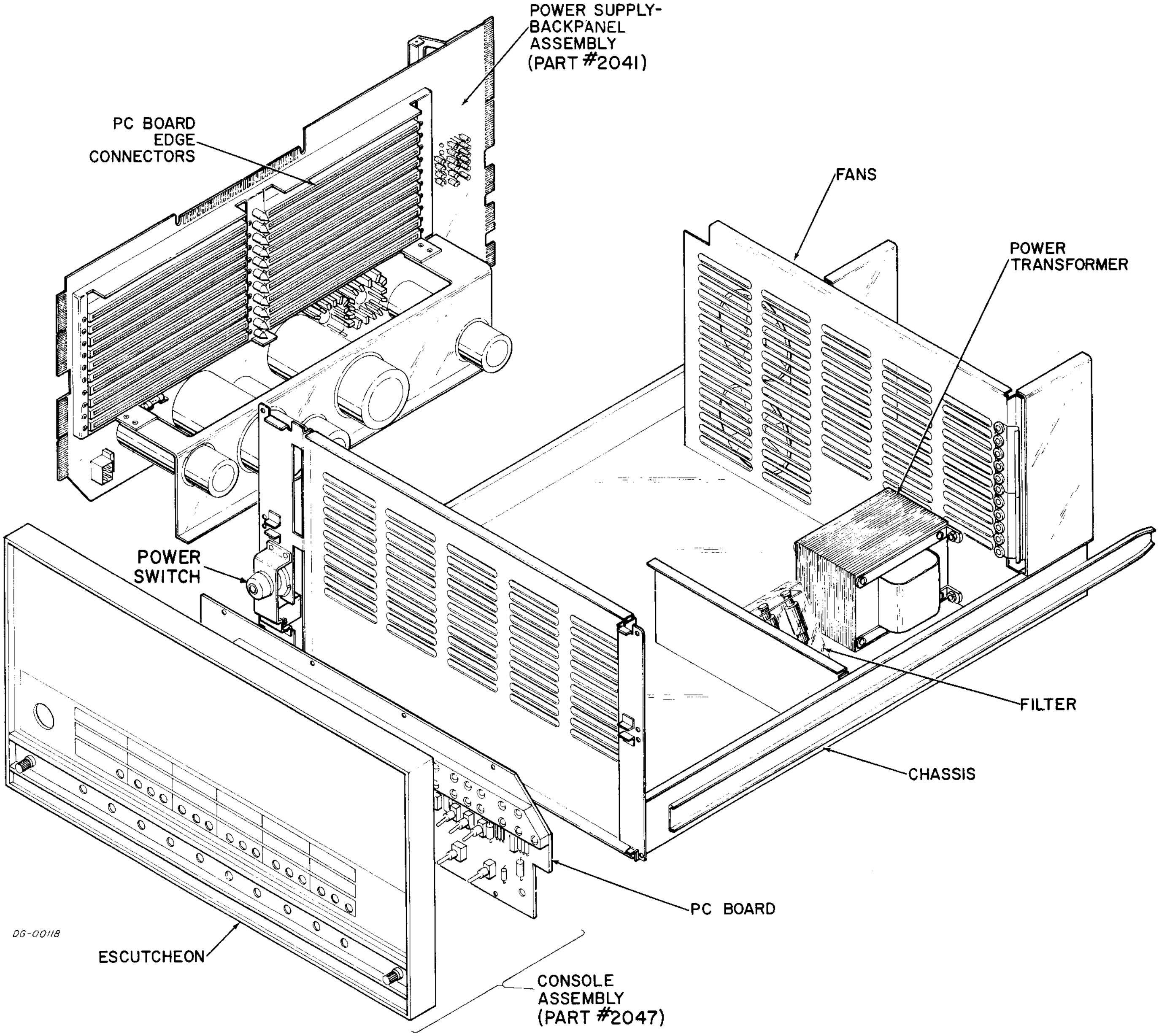


Figure O-1 Exploded View of The Nova 1220 Computer With Central Processor and Memory Cards Removed

The Central Processor, Console, Memories and Controllers communicate with each other along 16 bit buses called MEM, MBO and IN-OUT as shown in Figure O-2. MEM transfers information from Memory or the Console to the MBO or Instruction registers; MBO transfers information from the MBO register to the Console and Memories, and IN-OUT transfers information between the Memory's MB register and peripheral controllers. In the Nova 1220 proper all these data paths and their associated control signals travel along etched tracks on the backpanel to the board's edge connectors and to a plug in the console's PC board.

THIS MANUAL

This manual explains how the basic Nova 1220 works, how it is installed and how it is maintained. It is divided into 8 sections:

Section O introduces the machine and this manual;

Section C explains how the Central Processor works;

Section K explains how the operator's Console works;

Section P explains how the Power Supply works;

Section M explains how the Memories work;

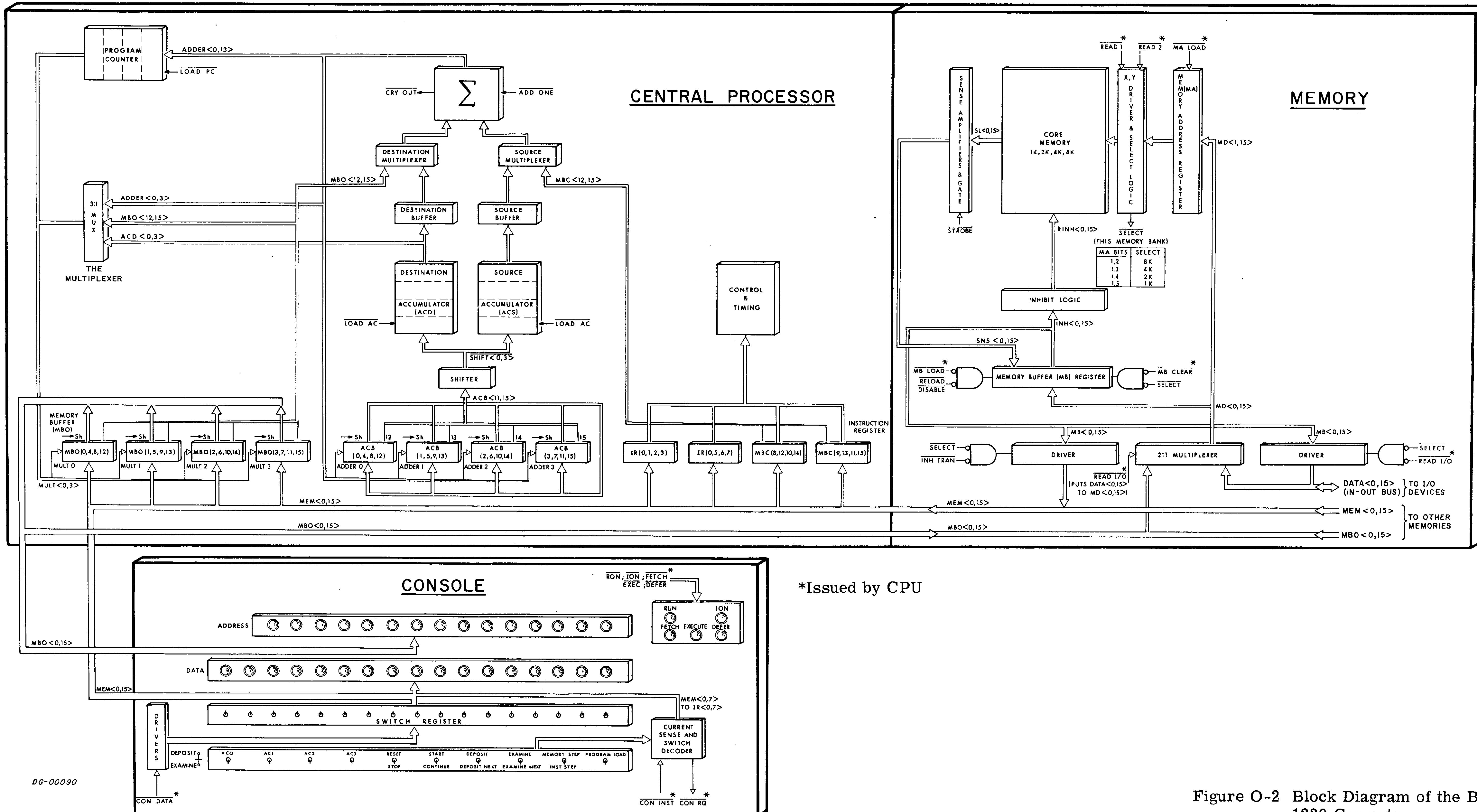
Section I explains how to install the computer;

Section N explains how to maintain the computer;

Section T has two reference tables - a signal list and a list of expanded abbreviations. The signal list traces the source and destination of each signal in the Central Processor and the Memory. Source signals are listed alphanumerically by name. Each source signal originates at the output pin (PIN) of an integrated circuit (CHIP) which is called out on a drawing (DWG) at a grid reference (GRID). Each signal is wired to one or more ICs which themselves originate more signals, or (FUNCTIONS), whose names and locations are listed in the DESTINATION column beside their originating signal. Drawing numbers are identified by the last two numbers of the print followed by a hyphen followed by their sheet number(s).

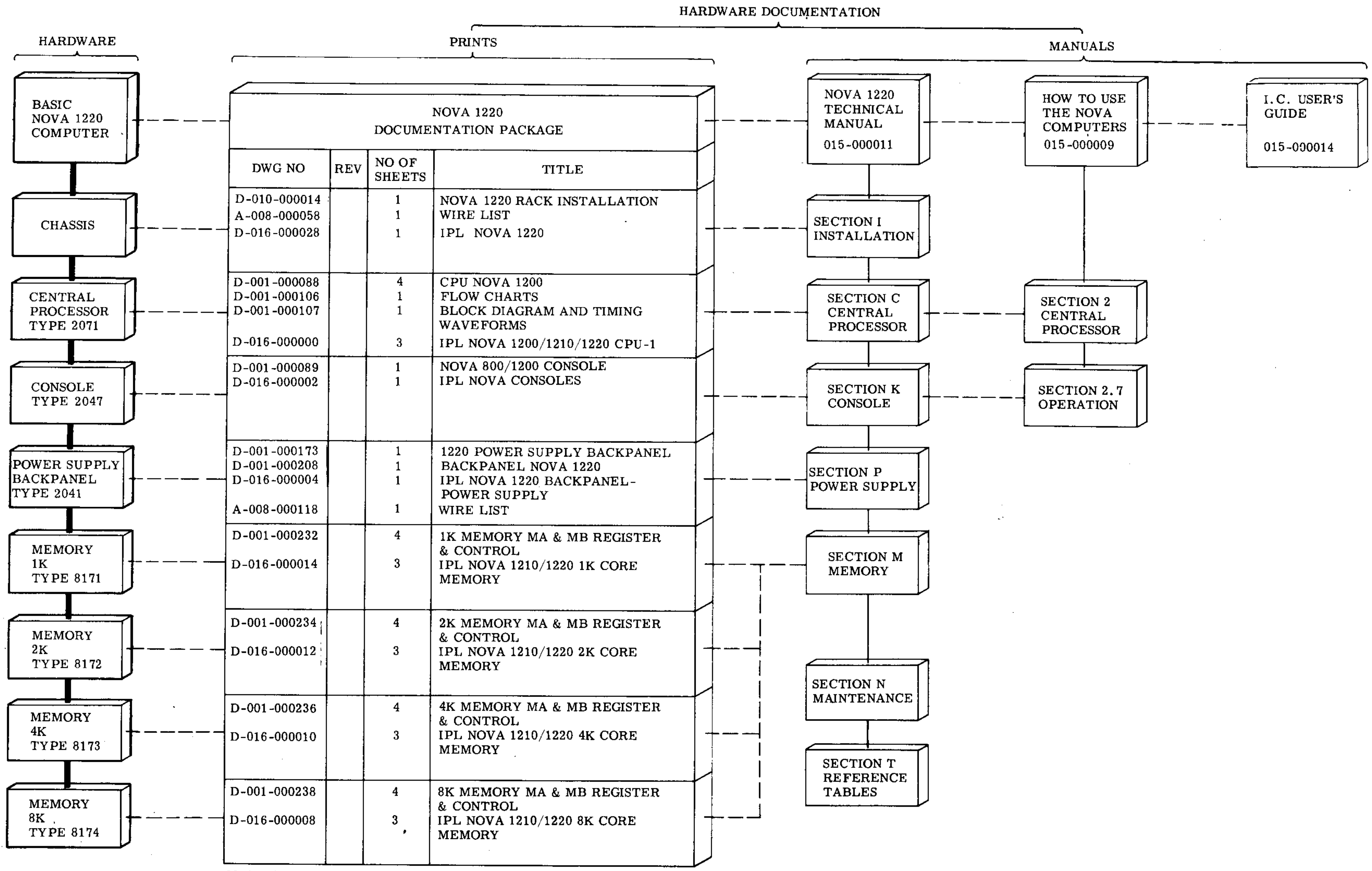
RELATED DOCUMENTS

Figure O-3 lists the engineering prints and manuals which describe the basic computer. The manual "How To Use The Nova Computers" explains how to program the machine. The manual "The I.C. User's Guide" gives logic diagrams and truth tables for the I.C.s used in Data General's machines.



*Issued by CPU

Figure O-2 Block Diagram of the Basic 1220 Computer



DWG NO	REV	NO OF SHEETS	TITLE
D-010-000014 A-008-000058 D-016-000028		1 1 1	NOVA 1220 RACK INSTALLATION WIRE LIST IPL NOVA 1220
D-001-000088 D-001-000106 D-001-000107 D-016-000000		4 1 1 3	CPU NOVA 1200 FLOW CHARTS BLOCK DIAGRAM AND TIMING WAVEFORMS IPL NOVA 1200/1210/1220 CPU-1
D-001-000089 D-016-000002		1 1	NOVA 800/1200 CONSOLE IPL NOVA CONSOLES
D-001-000173 D-001-000208 D-016-000004 A-008-000118		1 1 1 1	1220 POWER SUPPLY BACKPANEL BACKPANEL NOVA 1220 IPL NOVA 1220 BACKPANEL- POWER SUPPLY WIRE LIST
D-001-000232 D-016-000014		4 3	1K MEMORY MA & MB REGISTER & CONTROL IPL NOVA 1210/1220 1K CORE MEMORY
D-001-000234 D-016-000012		4 3	2K MEMORY MA & MB REGISTER & CONTROL IPL NOVA 1210/1220 2K CORE MEMORY
D-001-000236 D-016-000010		4 3	4K MEMORY MA & MB REGISTER & CONTROL IPL NOVA 1210/1220 4K CORE MEMORY
D-001-000238 D-016-000008		4 3	8K MEMORY MA & MB REGISTER & CONTROL IPL NOVA 1210/1220 8K CORE MEMORY

Figure O-3 Nova 1220 Hardware Documentation

This Page Left Blank

Intentionally

SECTION C

THE CENTRAL PROCESSOR UNIT

INTRODUCTION

The central processor unit (CPU) used in this computer is a binary, 2's complement, fixed word length, parallel/serial, digital, automatic processor. It takes up to 32K words of 1.2μsec co-ordinate-addressed core memory of 16 bits per word. It has 7 sixteen bit hardware registers: four accumulators (AC0, AC1, AC2 and AC3); a program-transparent shift buffer (ACB); a program-transparent memory buffer (MBO); and one 15 bit program counter (PC). All internal data paths are four bits (or one "nibble") wide, so each internal transfer takes four steps; all three external data paths or buses, (MEM, MBO and IN-OUT) are 16 bits wide so each external transfer takes one step.

There are three classes of instructions; memory reference (EFA), input-output (I/O) and arithmetic and logic (ALC). There are three modes of addressing; absolute, index (to AC2 or AC3) and relative (to PC).

Peripheral devices can interrupt the processor and transfer data to or from its accumulators via the I/O instruction set, or simply use the processor's high speed data channel directly to memory.

The CPU is contained on a single 15" by 15" PC board which is inserted into the first slot of the computer's chassis. Power is supplied by the chassis' power supply.

THE CONTROL UNIT

The CPU is a synchronous processor for which time is broken up by two clocks into discrete, fixed periods. The two clocks are derived from a 13.333Mhz crystal oscillator which is divided by two. One clock, called MEM CLK is always running; the other, called CPU CLK is gated by three signals RUN, $\overline{\text{STUTTER}}$ and $\overline{\text{WHOA}}$. RUN is a control flip-flop which stops the processor when it resets; $\overline{\text{STUTTER}}$ inhibits the clock for one cycle and $\overline{\text{WHOA}}$ is used by certain options like the multiply divide to slow the machine down. With these clocks the Control generates eight major states and two levels of minor states called timing state (TS) cycles and timing generator (TG) cycles.

Major States

Major states 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 during which none of the other states are set.

1. Fetch occurs when the next word to be read from memory is to be treated as an instruction.
2. Defer occurs when the next word from memory is to be treated as the address of an operand or instruction, i.e., during indirect addressing.
3. Execute occurs 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.
4. PI occurs during a program interrupt when:
   - the contents of the PC are stored in location 0
   - the next major state is set to Defer
   - A JMP instruction is forced into the Instruction Register
   - the next address executed is in location 1, which should be set to the starting address of the service routine.
5. DCH occurs when the next memory cycle is to be a direct transfer between an I/O device and Memory.
6. Key occurs 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.
7. Keym occurs when the manual function requires a memory cycle, such as Examine or Program Load.
8. "Dummy" State occurs 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.

TS Cycles

The TS cycles are four clock pulses long, and may be thought of as the time required to transfer a 16 bit word between two CPU registers at the rate of four bits per clock cycle. Each Major State consists of at least two complementary TS levels, called TS0 and TS3. TS0 occurs during the first half of the Major State, and TS3 occurs during the second half. Certain operations require more time than that provided by the two TS cycles, so a flip-flop called Loop is set to force the TS0 cycle to repeat and give the Major State three TS time intervals. During TS0 of this operation the data is fetched from the memory and loaded into the MBO; then Loop is set, TS0 is repeated, and the data in the MBO is shifted through the Adder. Finally, TS3 is set and the data is transferred from the MBO to the Memory and re-written.

Timing Generator Cycles

There are three timing generators, called the processor timing generator (PTG); the accumulator timing generator (ACTG) and the memory timing generator (MTG). These timing generators effectively designate the clock pulses for specific functions in the processor, accumulator and memory respectively.

The Processor Timing Generator. This two bit counter, designated, PTG0 and PTG1, cycles every four clock pulses. PTG0 is set during the two middle clock cycles of a TS cycle, and PTG1 is set during the last two cycles of a TS cycle. These two levels are decoded into two others called PTG2 and PTG5. PTG2 is the last clock interval during TS0, and PTG5 is the last clock interval during TS3. PTG5 is used, for example, to enable the major state flip-flops. PTG0 "anded" with TS0 to form $\overline{PTG0 \cdot TS0}$, the first clock interval during TS0, is used to increment the Adder as the least significant four bit nibble is passed through it. Figures C-1 and C-2 show the timing for the PTG during FETCH or KEY major states, and all other states.

The Accumulator Timing Generator. This two bit counter, designated ACTG0 and ACTG1, is always one clock state ahead of the PTG counter. Its two signals are used to drive the accumulator chips. Their timing is given in Figure C-3.

The Memory Timing Generator. This four bit counter, designated MTG0, MTG2, MTG3, is used to form the control signals for memory. Its timing is given in Figure C-4.

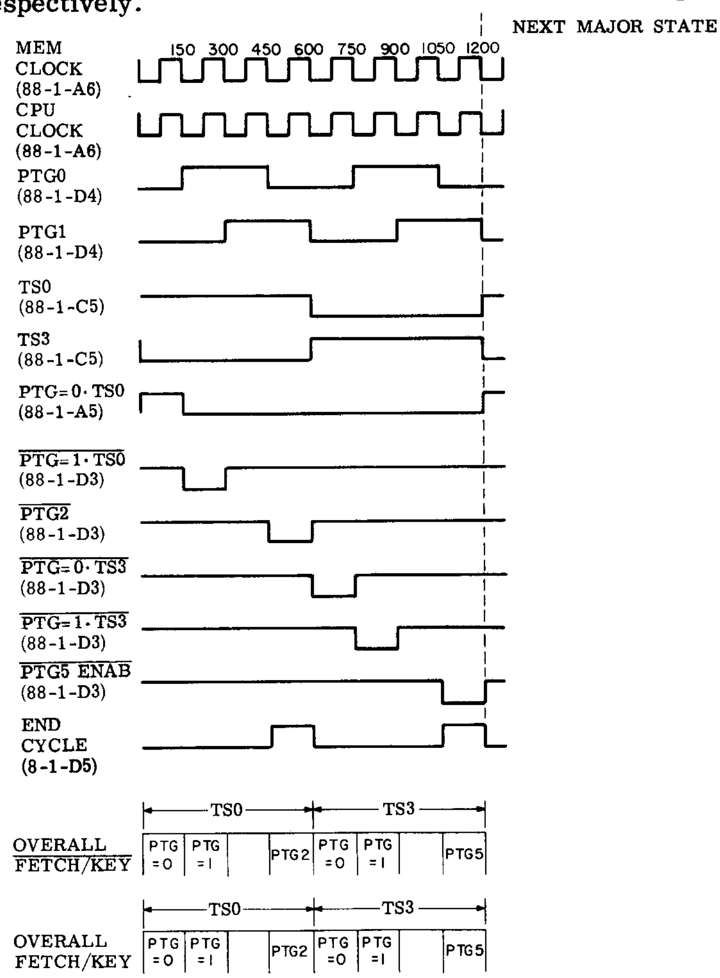


Figure C-1 Timing For The Processor Timing Generator During All Major States Except Fetch or Key

DG-00045

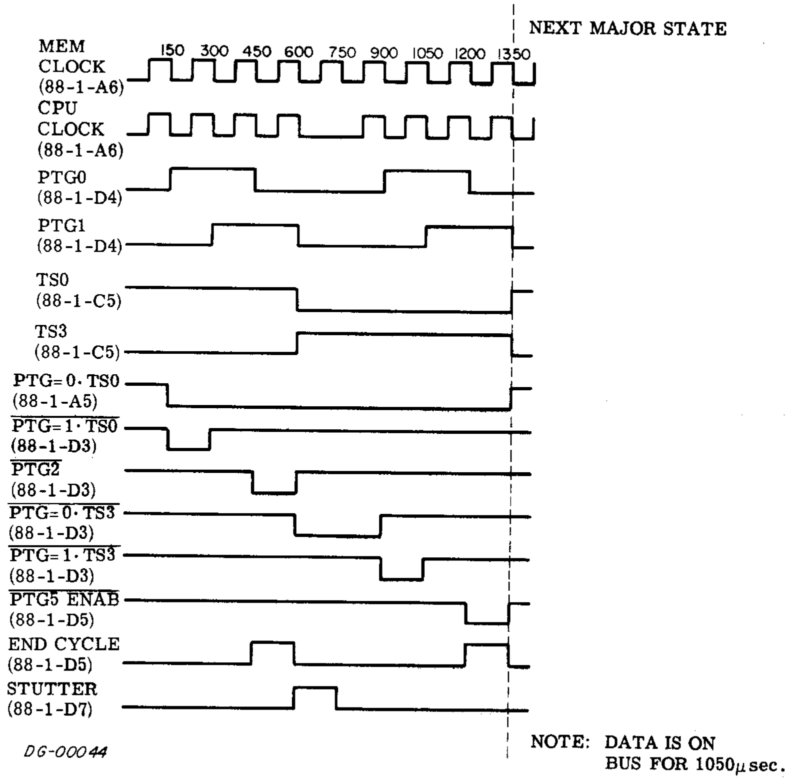


Figure C-2 Timing For The Processor Timing Generator During Fetch or Key

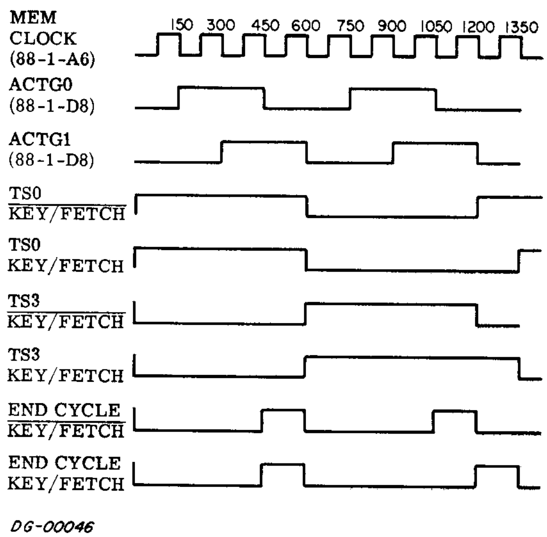
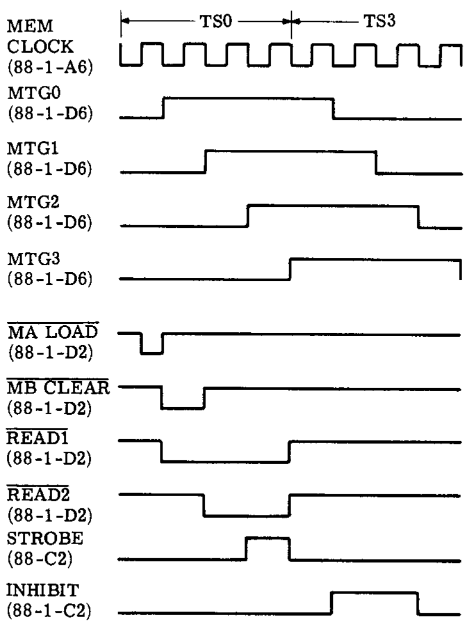


Figure C-3 Timing For The Accumulator Timing Generator

ACCUMULATOR TRUTH TABLE (88-4-B6 & B7 U124 & U123)

ACTG0	ACTG1	
0	0	BITS 12-15
1	0	BITS 8-11
1	1	BITS 4-7
0	1	BITS 0-3



MEMORY TIME GEN. COUNTS

	MTG0	MTG1	MTG2	MTG3
TS0	0	0	0	0
1st CLOCK	1	0	0	0
2nd CLOCK	1	1	0	0
3rd CLOCK	1	1	1	0
4th CLOCK	1	1	1	1
		SEE NOTE		
TS3	1	1	1	1
1st CLOCK	0	1	1	1
2nd CLOCK	0	0	1	1
3rd CLOCK	0	0	0	1
4th CLOCK	0	0	0	0

NOTE - IF LOOPING TS0, CLOCK FREEZES WITH ALL ONES UNTIL FIRST CLOCK IN TS3.

DG-00047

Figure C-4 Timing For The Memory Timing Generator

CPU DATA PATHS

Registers

The CPU is organized around eight hardware registers as shown in Figure C-5; a shift buffer (ACB); a program counter (PC); a CPU interface register (MBO); an instruction register (IR and MBC); and four accumulators, (AC0, AC1, AC2, AC3). These eight registers are all 16 bits long except for the PC which is 15 bits. All internal data paths are four bits wide, so it takes four separate operations to perform an add, or a register-to-register transfer.

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 so that it points to the next sequential instruction. Certain instructions, such as JMP can change the contents of the PC. The PC consists of one 16 bit latch.

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 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 and the IR bits remain static.

CPU Interface Register (MBO). The MBO is used in every operation the CPU performs. It acts as a parallel-to-serial converter for 16 bit data flowing into the machine from the MEM bus. This data is loaded from the MEM bus into the MBO in parallel, and shifted 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 Memory. The MBO then modifies the data by recirculating it through the Adder and back into the MBO. The modified data is then loaded from the MBO back into Memory.

Shift Buffer (ACB). All data to be loaded into the Accumulators are passed through the ACB, where the results of an ALC instruction are assembled before they are loaded back into the Destination Accumulator.

Accumulators (AC0, AC1, AC2, AC3.) There are two identical sets of four - 16bit accumulators all of which can be logically and arithmetically manipulated under program control. Each set of accumulators is contained in a single 64 bit chip; (only one accumulator- nibble per chip can be addressed at any one time). Since it is necessary to be able to access two accumulators simultaneously, two sets are available, called source (S) and destination (D), each set containing the same information as the other. For example, two accumulators can be added together by simultaneously fetching the source data from one chip and the destination data from the other and then adding the two. The accumulators are buffered by four bit registers (source and destination) so that the next nibble can be selected while the current nibble is being processed. It takes 100 ns to access a nibble in the accumulator, and 100 ns to move a nibble through the Adder and Multiplexer, so by overlapping the two, the total time to process a nibble is 100 ns.

During the first nibble, the Adder is idle and a flag called STUTTER inhibits the clock until data is ready.

Data Flow

Nibble Transfers. 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 the nibble to be acted upon. For example, to increment a word 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, so a "one" is 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. 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.

Instruction Overlapping. Certain instructions are carried out at the same time as parts of other instructions. For example, any operation which loads an accumulator is overlapped with the next major state. Such is the case with the ALC instruction when the CPU first operates upon the accumulator(s), loads the result into the ACB register while memory is re-writing the instruction, and then waits until the next state to transfer the result from the ACB back into the accumulator. The next state could be FETCH, PI, DCH or even KEY. 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 shifts through the multiplexer/shifter during TS0 of the next major state, after which the data may or may not be loaded into the accumulators. The output of the multiplexer/shifter is checked for all zeroes to see if it fulfills the skip conditions. If it does, 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 as required to permit the ADDRESS lights on the Console to show the correct next address when the machine is stopped.

Data Buses

Data is transferred between memory and the central processor or an I/O device along three data buses called:

- $\overline{\text{MEM}}$ which transfers data from memory to the Central Processor;
- $\overline{\text{MBO}}$ which transfers data from the Central Processor to Memory;
- $\overline{\text{DATA}}$ which transfers data in either direction between memory and I/O devices.

During an output I/O instruction, data moves from the source AC into the MBO and on to the MBO bus. From the bus it is strobed into the memory MB register and on through the IN-OUT bus to the destination device. During an output I/O instruction the destination device outputs to the IN-OUT bus into the memory's MB register, which dumps into the MEM bus. The MEM bus is strobed into the MBO which moves it through the Adder to the ACB and into the destination AC.

THE FLOW AND TIMING DIAGRAMS

The following diagrams illustrate each step in the sequence of functions carried out by the central processor and memory. Each block of a flow diagram describes an operation, its data path and the location of critical logic. For example, this block means that the ACB register was transferred to an AC register via the

$ACB \xrightarrow{S} AC$
88-4-A7

shifter (ACB) which is located on print 001-000088, sheet 4, in grid A7. The symbol Σ means Adder, M means Multiplexer, and S means Shifter. Supporting notes near the blocks give the current time state, relevant figures and the status of important signals.

REFERENCES

1. Nova 1200 CPU — Print D-001-000088-13
2. Flow Charts — Print D-001-000106-00
3. Waveforms — Print D-001-000107-00

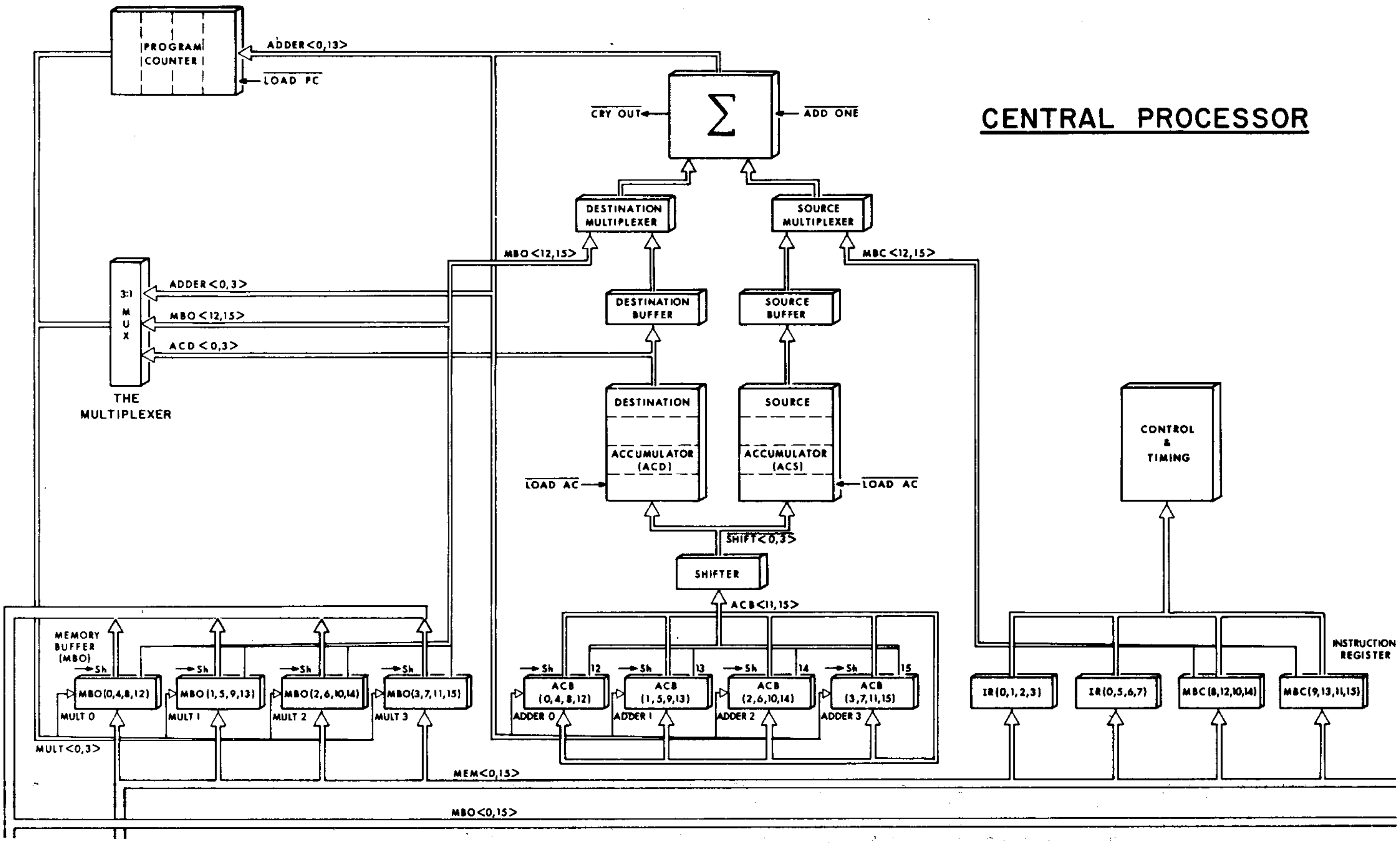
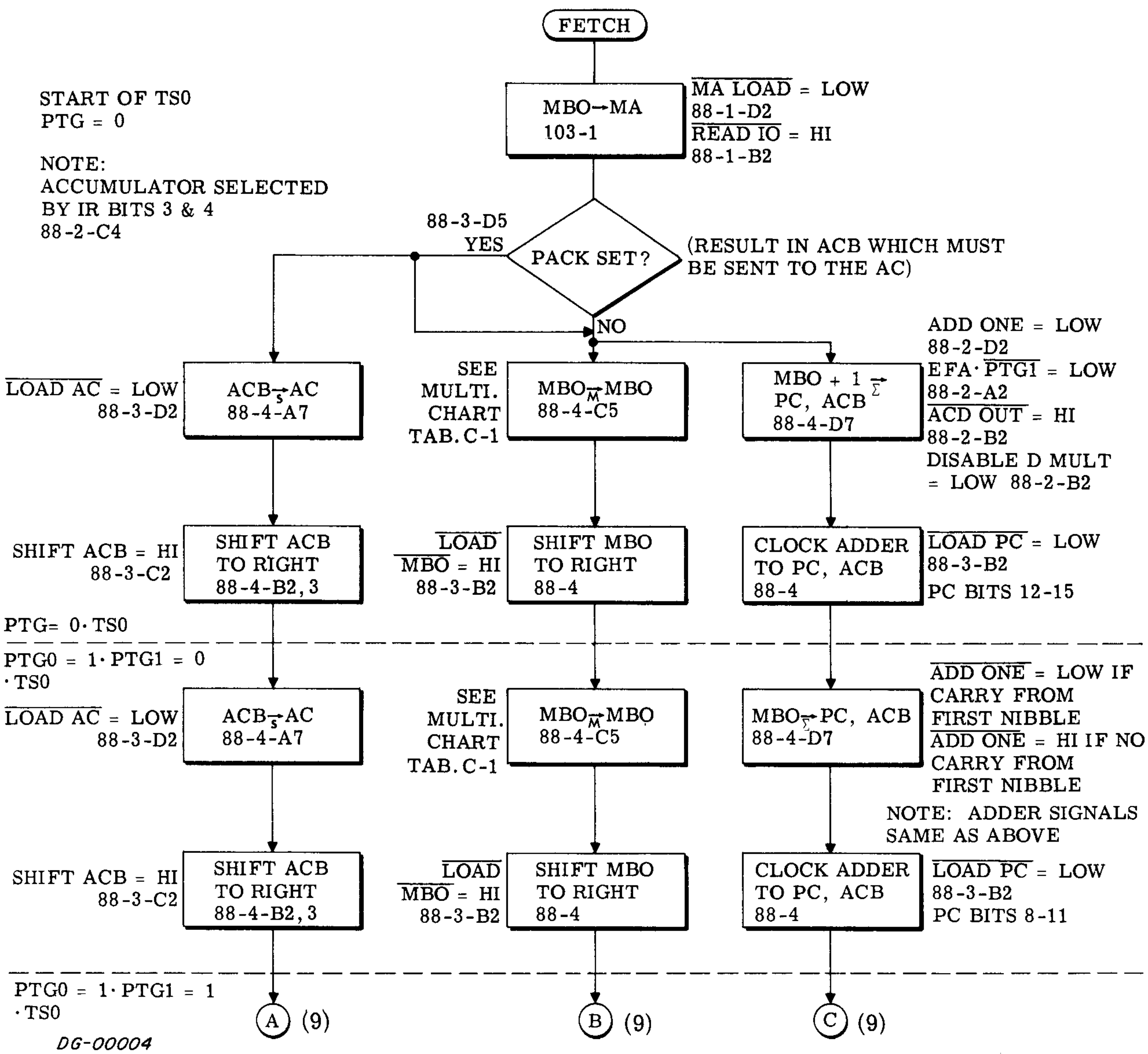
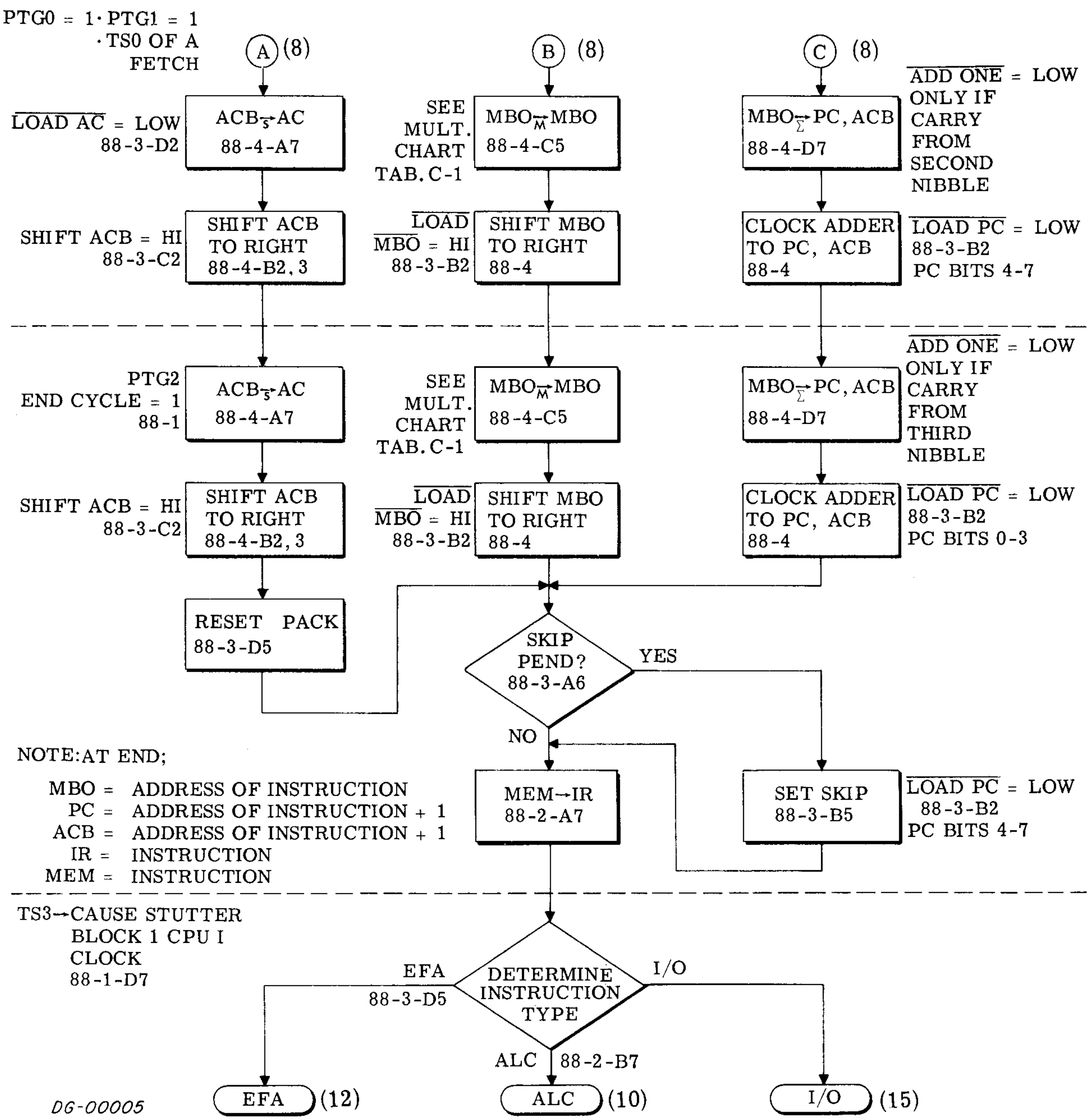
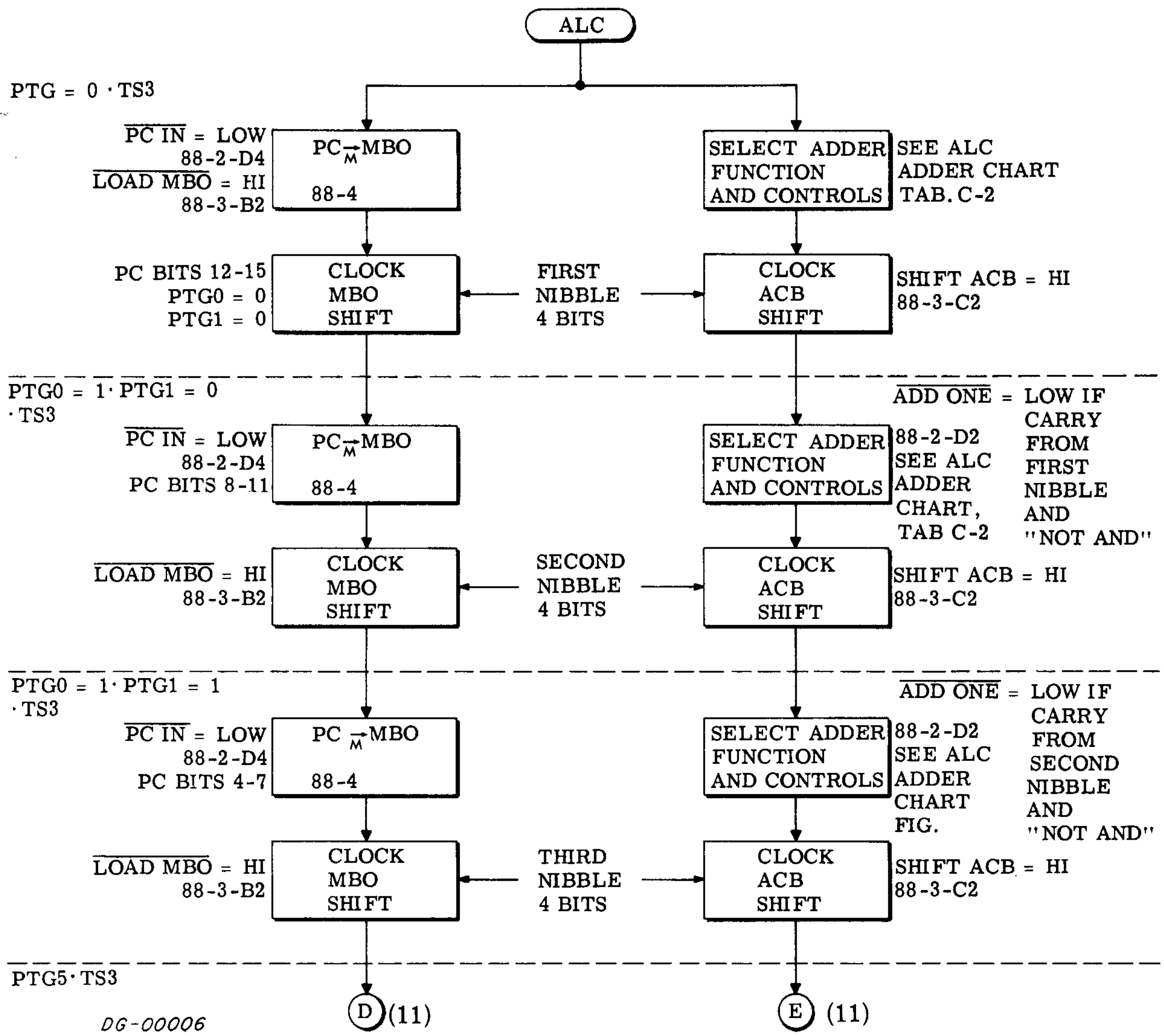


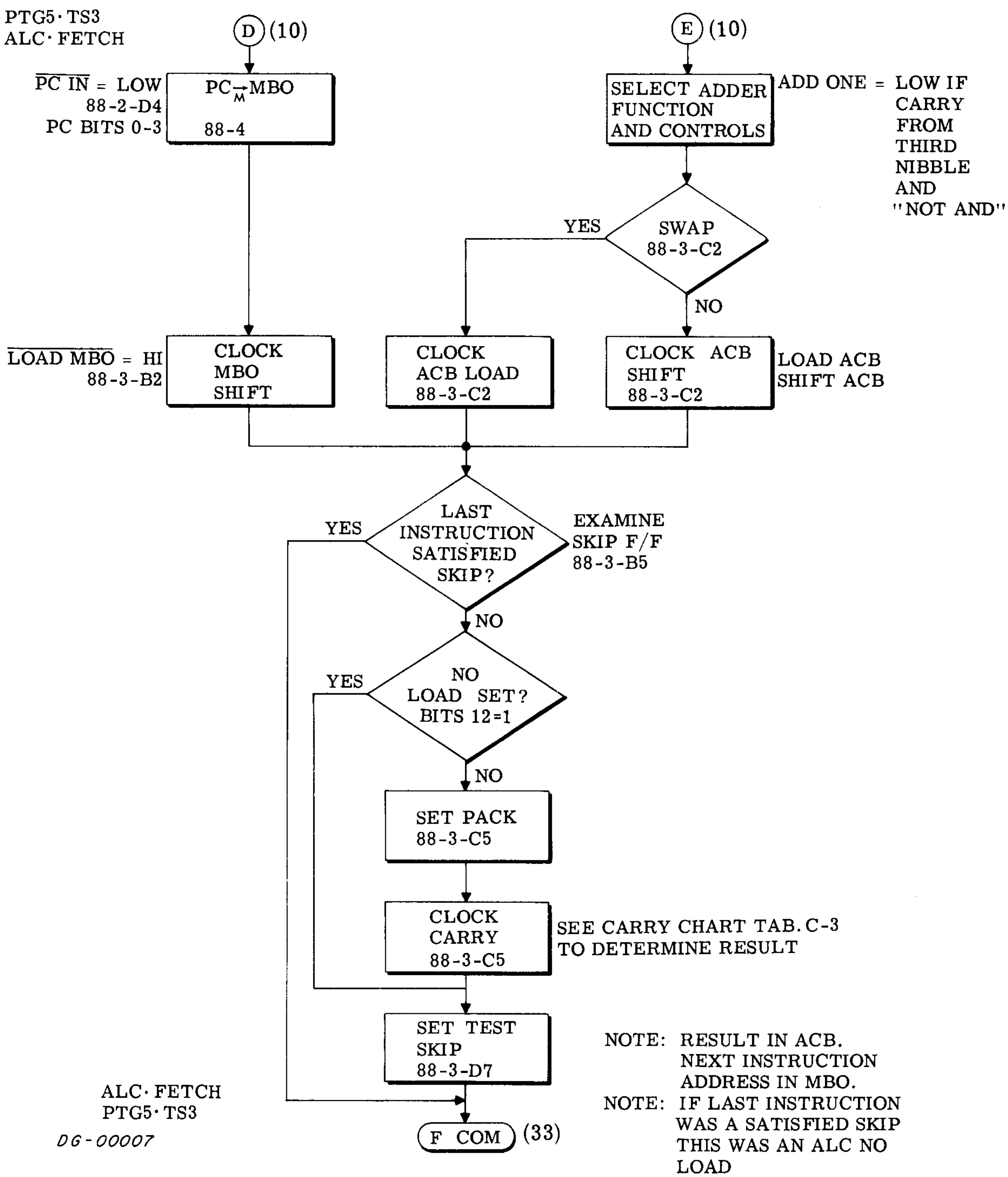
Figure C-5 The Nova 1220 Central Processor


FETCH
START OF TS0
PTG = 0
NOTE:
ACCUMULATOR SELECTED
BY IR BITS 3 & 4
88-2-C4
MBO→MA
103-1
MA LOAD = LOW
88-1-D2
READ IO = HI
88-1-B2
88-3-D5
YES
PACK SET?
(RESULT IN ACB WHICH MUST
BE SENT TO THE AC)
NO
LOAD AC = LOW
88-3-D2
ACB→AC
88-4-A7
SEE
MULTI.
CHART
TAB. C-1
MBO→MBO
88-4-C5
MBO + 1 →
PC, ACB
88-4-D7
ADD ONE = LOW
88-2-D2
EFA·PTG1 = LOW
88-2-A2
ACD OUT = HI
88-2-B2
DISABLE D MULT
= LOW 88-2-B2
SHIFT ACB = HI
88-3-C2
SHIFT ACB
TO RIGHT
88-4-B2, 3
LOAD
MBO = HI
88-3-B2
SHIFT MBO
TO RIGHT
88-4
CLOCK ADDER
TO PC, ACB
88-4
LOAD PC = LOW
88-3-B2
PC BITS 12-15
PTG= 0·TS0
PTG0 = 1·PTG1 = 0
·TS0
LOAD AC = LOW
88-3-D2
ACB→AC
88-4-A7
SEE
MULTI.
CHART
TAB. C-1
MBO→MBO
88-4-C5
MBO→PC, ACB
88-4-D7
ADD ONE = LOW IF
CARRY FROM
FIRST NIBBLE
ADD ONE = HI IF NO
CARRY FROM
FIRST NIBBLE
NOTE: ADDER SIGNALS
SAME AS ABOVE
SHIFT ACB = HI
88-3-C2
SHIFT ACB
TO RIGHT
88-4-B2, 3
LOAD
MBO = HI
88-3-B2
SHIFT MBO
TO RIGHT
88-4
CLOCK ADDER
TO PC, ACB
88-4
LOAD PC = LOW
88-3-B2
PC BITS 8-11
PTG0 = 1·PTG1 = 1
·TS0
A (9)
B (9)
C (9)
DG-00004

Rev. 01

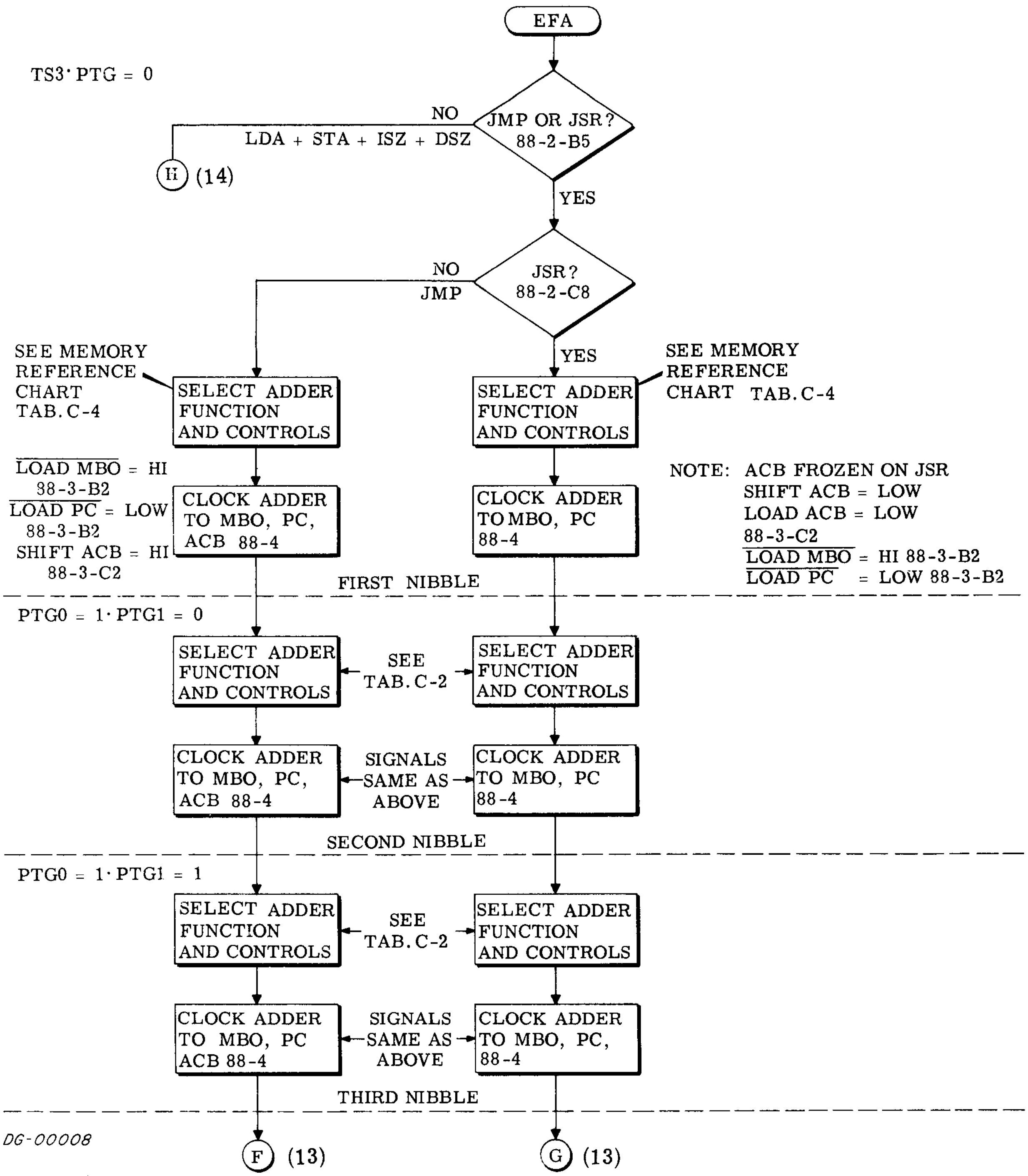

PTG0 = 1 · PTG1 = 1 · TS0 OF A FETCH
A (8)
B (8)
C (8)
LOAD AC = LOW 88-3-D2
ACB→AC 88-4-A7
SEE MULT. CHART TAB. C-1
MBO→MBO 88-4-C5
MBO→PC, ACB 88-4-D7
ADD ONE = LOW ONLY IF CARRY FROM SECOND NIBBLE
SHIFT ACB = HI 88-3-C2
SHIFT ACB TO RIGHT 88-4-B2, 3
LOAD MBO = HI 88-3-B2
SHIFT MBO TO RIGHT 88-4
CLOCK ADDER TO PC, ACB 88-4
LOAD PC = LOW 88-3-B2 PC BITS 4-7
PTG2 END CYCLE = 1 88-1
ACB→AC 88-4-A7
SEE MULT. CHART TAB. C-1
MBO→MBO 88-4-C5
MBO→PC, ACB 88-4-D7
ADD ONE = LOW ONLY IF CARRY FROM THIRD NIBBLE
SHIFT ACB = HI 88-3-C2
SHIFT ACB TO RIGHT 88-4-B2, 3
LOAD MBO = HI 88-3-B2
SHIFT MBO TO RIGHT 88-4
CLOCK ADDER TO PC, ACB 88-4
LOAD PC = LOW 88-3-B2 PC BITS 0-3
RESET PACK 88-3-D5
SKIP PEND? 88-3-A6
YES
NO
NOTE:AT END;
MBO = ADDRESS OF INSTRUCTION
PC = ADDRESS OF INSTRUCTION + 1
ACB = ADDRESS OF INSTRUCTION + 1
IR = INSTRUCTION
MEM = INSTRUCTION
MEM→IR 88-2-A7
SET SKIP 88-3-B5
LOAD PC = LOW 88-3-B2 PC BITS 4-7
TS3→CAUSE STUTTER BLOCK 1 CPU I CLOCK 88-1-D7
EFA 88-3-D5
DETERMINE INSTRUCTION TYPE
I/O
ALC 88-2-B7
EFA (12)
ALC (10)
I/O (15)
DG-00005


ALC
PTG = 0 · TS3
PC IN = LOW
88-2-D4
LOAD MBO = HI
88-3-B2
PC→MBO
M
88-4
SELECT ADDER FUNCTION AND CONTROLS
SEE ALC ADDER CHART TAB. C-2
PC BITS 12-15
PTG0 = 0
PTG1 = 0
CLOCK MBO SHIFT
FIRST NIBBLE 4 BITS
CLOCK ACB SHIFT
SHIFT ACB = HI
88-3-C2
PTG0 = 1· PTG1 = 0
· TS3
ADD ONE = LOW IF CARRY FROM FIRST NIBBLE AND "NOT AND"
PC IN = LOW
88-2-D4
PC BITS 8-11
PC→MBO
M
88-4
SELECT ADDER FUNCTION AND CONTROLS
88-2-D2
SEE ALC ADDER CHART, TAB C-2
LOAD MBO = HI
88-3-B2
CLOCK MBO SHIFT
SECOND NIBBLE 4 BITS
CLOCK ACB SHIFT
SHIFT ACB = HI
88-3-C2
PTG0 = 1· PTG1 = 1
· TS3
ADD ONE = LOW IF CARRY FROM SECOND NIBBLE AND "NOT AND"
PC IN = LOW
88-2-D4
PC BITS 4-7
PC → MBO
M
88-4
SELECT ADDER FUNCTION AND CONTROLS
88-2-D2
SEE ALC ADDER CHART FIG.
LOAD MBO = HI
88-3-B2
CLOCK MBO SHIFT
THIRD NIBBLE 4 BITS
CLOCK ACB SHIFT
SHIFT ACB = HI
88-3-C2
PTG5· TS3
D (11)
E (11)
DG-00006

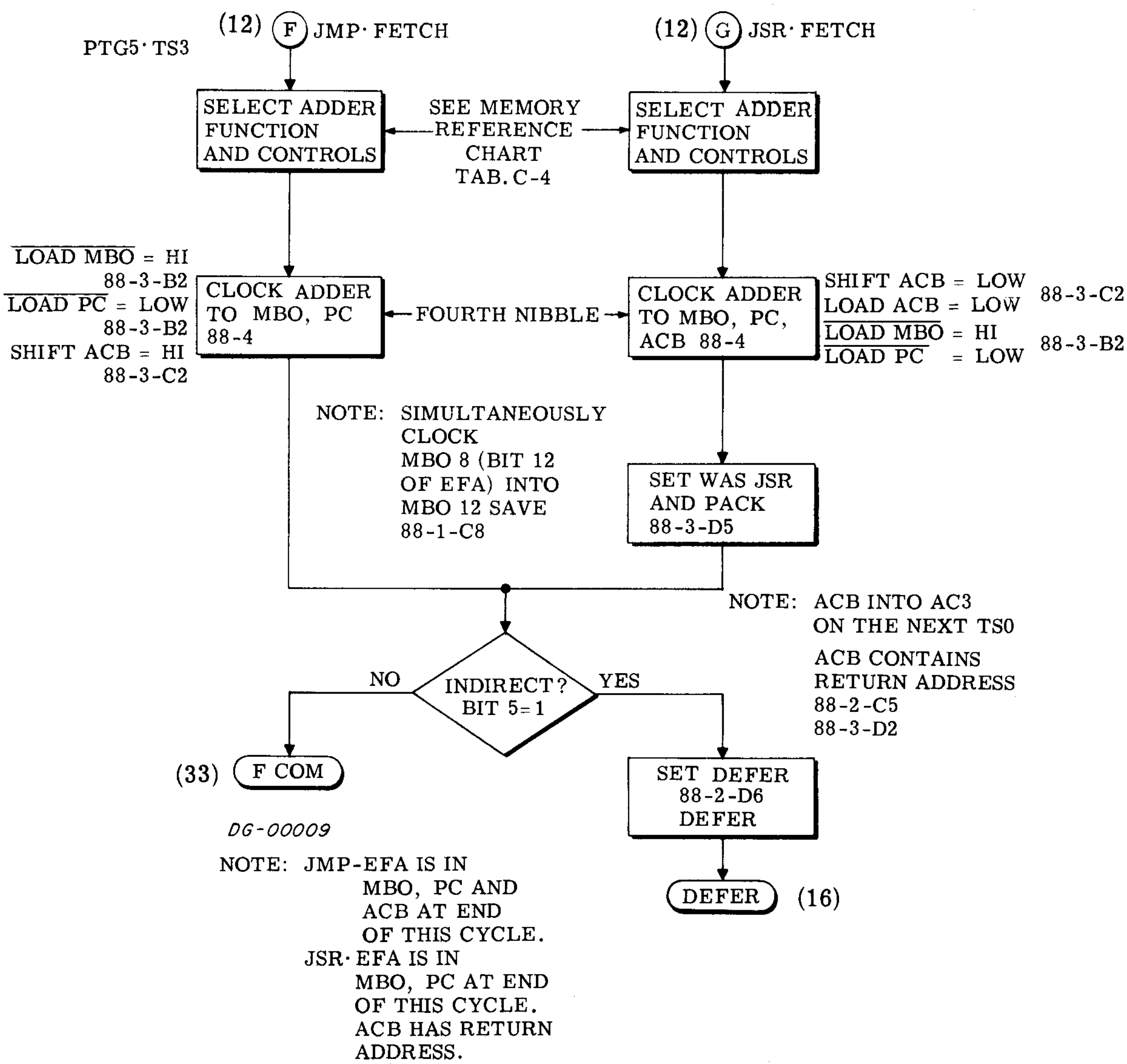


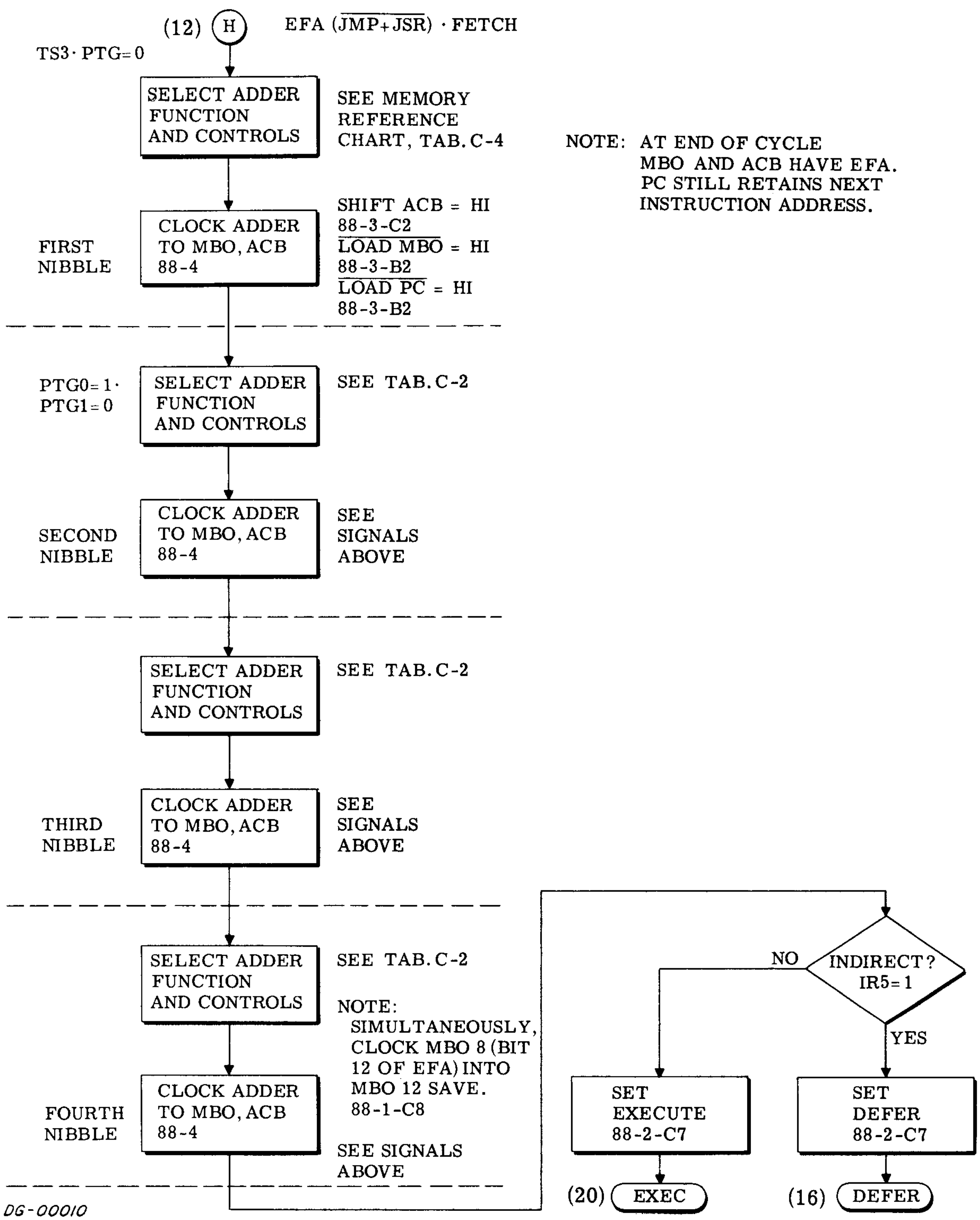
PTG5· TS3
ALC· FETCH
D (10)
E (10)
PC IN = LOW
88-2-D4
PC BITS 0-3
PC→MBO
M
88-4
SELECT ADDER FUNCTION AND CONTROLS
ADD ONE = LOW IF CARRY FROM THIRD NIBBLE AND "NOT AND"
YES
SWAP
88-3-C2
NO
LOAD MBO = HI
88-3-B2
CLOCK MBO SHIFT
CLOCK ACB LOAD
88-3-C2
CLOCK ACB SHIFT
88-3-C2
LOAD ACB
SHIFT ACB
YES
LAST INSTRUCTION SATISFIED SKIP?
EXAMINE SKIP F/F
88-3-B5
NO
YES
NO LOAD SET?
BITS 12=1
NO
SET PACK
88-3-C5
CLOCK CARRY
88-3-C5
SEE CARRY CHART TAB. C-3
TO DETERMINE RESULT
SET TEST SKIP
88-3-D7
NOTE: RESULT IN ACB.
NEXT INSTRUCTION ADDRESS IN MBO.
NOTE: IF LAST INSTRUCTION WAS A SATISFIED SKIP THIS WAS AN ALC NO LOAD
ALC· FETCH
PTG5· TS3
DG-00007
F COM (33)

Rev. 01

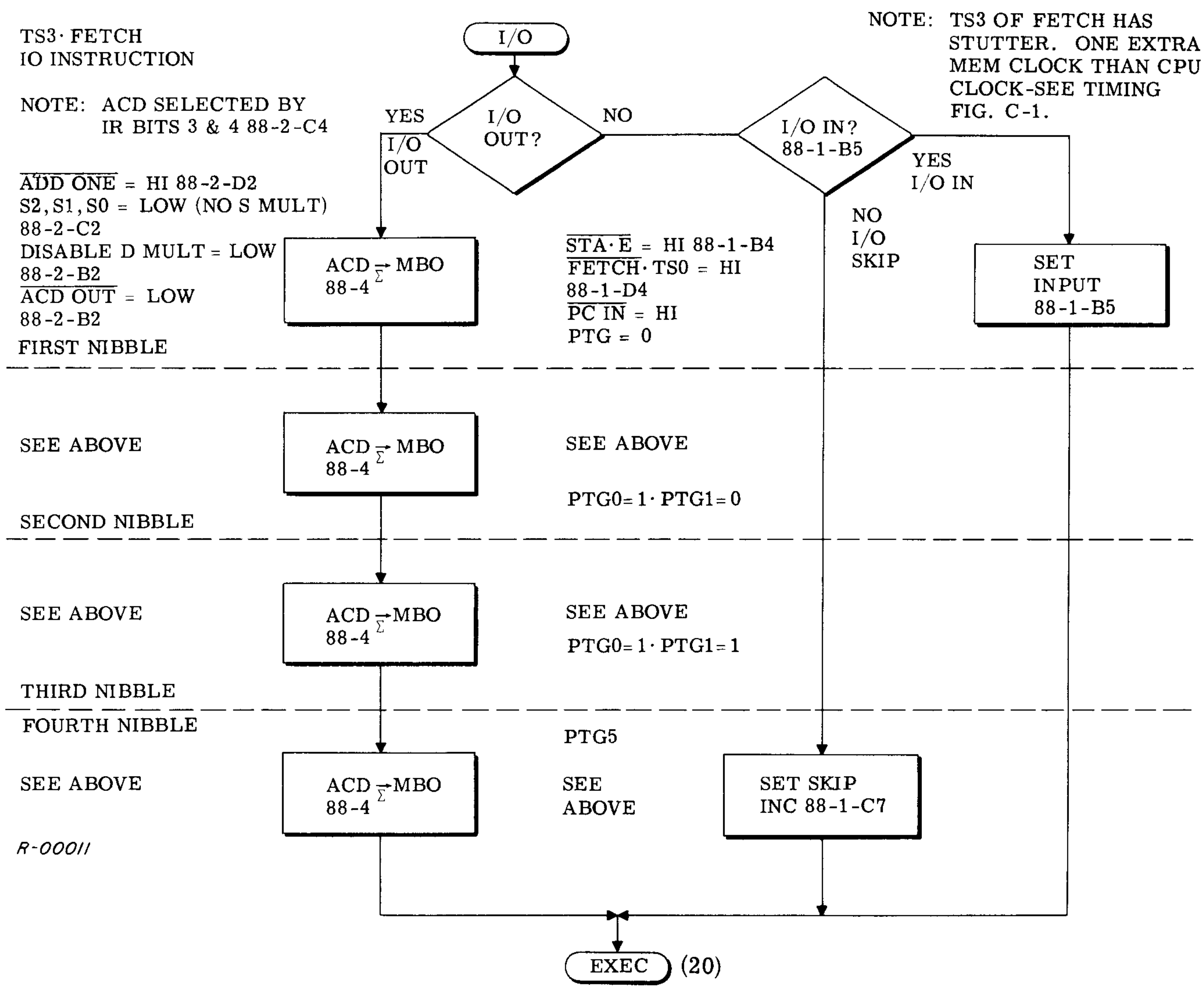

EFA
TS3·PTG = 0
JMP OR JSR? 88-2-B5
NO
LDA + STA + ISZ + DSZ
H (14)
YES
JSR? 88-2-C8
NO
JMP
YES
SEE MEMORY REFERENCE CHART TAB.C-4
SELECT ADDER FUNCTION AND CONTROLS
SEE MEMORY REFERENCE CHART TAB.C-4
SELECT ADDER FUNCTION AND CONTROLS
LOAD MBO = HI 88-3-B2
LOAD PC = LOW 88-3-B2
SHIFT ACB = HI 88-3-C2
CLOCK ADDER TO MBO, PC, ACB 88-4
CLOCK ADDER TO MBO, PC 88-4
NOTE: ACB FROZEN ON JSR
SHIFT ACB = LOW
LOAD ACB = LOW 88-3-C2
LOAD MBO = HI 88-3-B2
LOAD PC = LOW 88-3-B2
FIRST NIBBLE
PTG0 = 1·PTG1 = 0
SELECT ADDER FUNCTION AND CONTROLS
SEE TAB.C-2
SELECT ADDER FUNCTION AND CONTROLS
CLOCK ADDER TO MBO, PC, ACB 88-4
SIGNALS SAME AS ABOVE
CLOCK ADDER TO MBO, PC 88-4
SECOND NIBBLE
PTG0 = 1·PTG1 = 1
SELECT ADDER FUNCTION AND CONTROLS
SEE TAB.C-2
SELECT ADDER FUNCTION AND CONTROLS
CLOCK ADDER TO MBO, PC ACB 88-4
SIGNALS SAME AS ABOVE
CLOCK ADDER TO MBO, PC, 88-4
THIRD NIBBLE
F (13)
G (13)

DG-00008

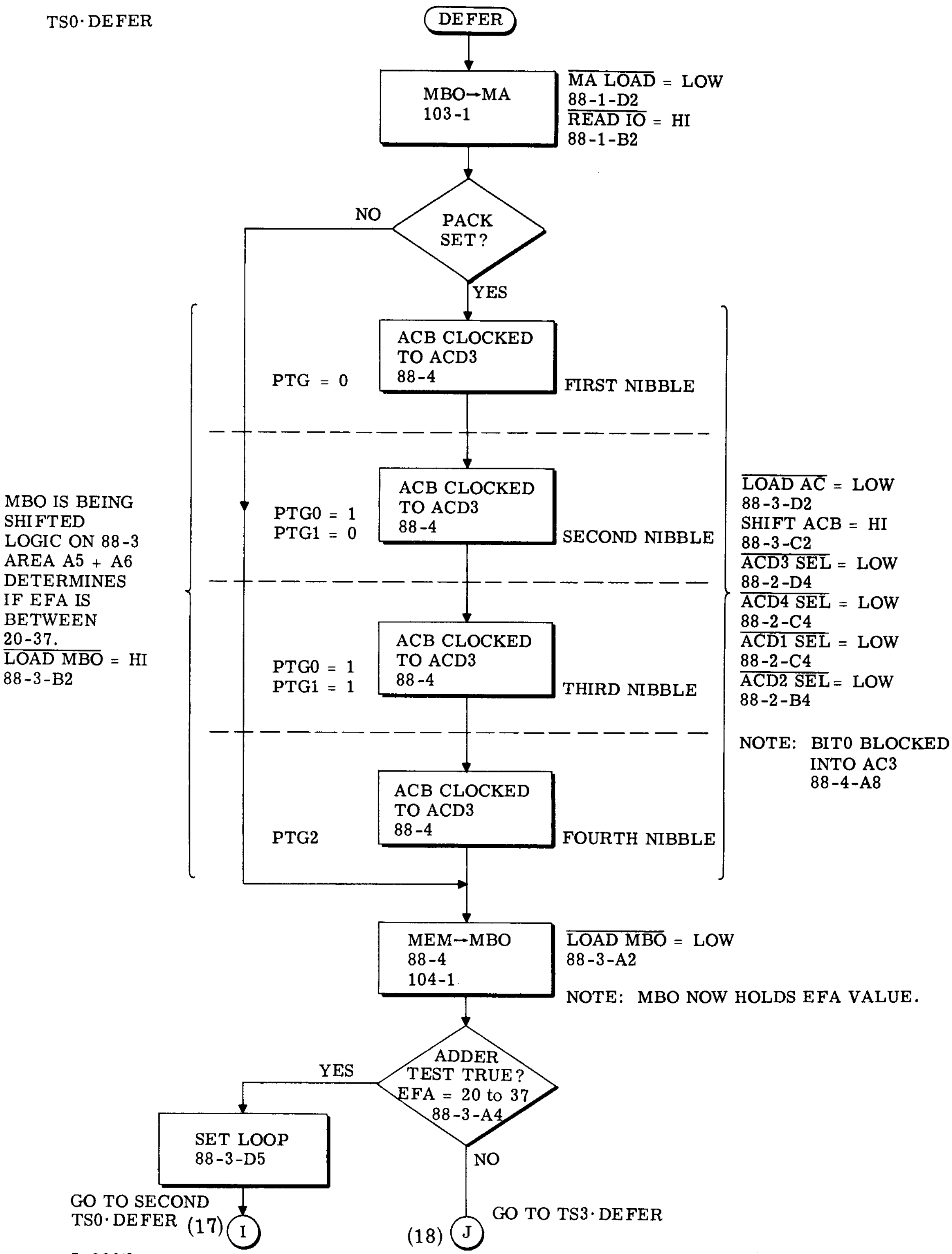

PTG5·TS3
(12) F JMP·FETCH
(12) G JSR·FETCH
SELECT ADDER FUNCTION AND CONTROLS
SEE MEMORY REFERENCE CHART TAB. C-4
SELECT ADDER FUNCTION AND CONTROLS
LOAD MBO = HI 88-3-B2
LOAD PC = LOW 88-3-B2
SHIFT ACB = HI 88-3-C2
CLOCK ADDER TO MBO, PC 88-4
FOURTH NIBBLE
CLOCK ADDER TO MBO, PC, ACB 88-4
SHIFT ACB = LOW
LOAD ACB = LOW
88-3-C2
LOAD MBO = HI
LOAD PC = LOW
88-3-B2
NOTE: SIMULTANEOUSLY CLOCK MBO 8 (BIT 12 OF EFA) INTO MBO 12 SAVE 88-1-C8
SET WAS JSR AND PACK 88-3-D5
NOTE: ACB INTO AC3 ON THE NEXT TS0
ACB CONTAINS RETURN ADDRESS 88-2-C5 88-3-D2
NO
INDIRECT? BIT 5=1
YES
(33) F COM
SET DEFER 88-2-D6 DEFER
DEFER (16)
DG-00009
NOTE: JMP-EFA IS IN MBO, PC AND ACB AT END OF THIS CYCLE.
JSR·EFA IS IN MBO, PC AT END OF THIS CYCLE. ACB HAS RETURN ADDRESS.


(12) H EFA (JMP+JSR) · FETCH
TS3 · PTG=0
SELECT ADDER FUNCTION AND CONTROLS
SEE MEMORY REFERENCE CHART, TAB. C-4
NOTE: AT END OF CYCLE MBO AND ACB HAVE EFA. PC STILL RETAINS NEXT INSTRUCTION ADDRESS.
FIRST NIBBLE
CLOCK ADDER TO MBO, ACB 88-4
SHIFT ACB = HI
88-3-C2
LOAD MBO = HI
88-3-B2
LOAD PC = HI
88-3-B2
PTG0=1 · PTG1=0
SELECT ADDER FUNCTION AND CONTROLS
SEE TAB. C-2
SECOND NIBBLE
CLOCK ADDER TO MBO, ACB 88-4
SEE SIGNALS ABOVE
SELECT ADDER FUNCTION AND CONTROLS
SEE TAB. C-2
THIRD NIBBLE
CLOCK ADDER TO MBO, ACB 88-4
SEE SIGNALS ABOVE
SELECT ADDER FUNCTION AND CONTROLS
SEE TAB. C-2
NOTE: SIMULTANEOUSLY, CLOCK MBO 8 (BIT 12 OF EFA) INTO MBO 12 SAVE. 88-1-C8
FOURTH NIBBLE
CLOCK ADDER TO MBO, ACB 88-4
SEE SIGNALS ABOVE
INDIRECT? IR5=1
NO
YES
SET EXECUTE 88-2-C7
SET DEFER 88-2-C7
(20) EXEC
(16) DEFER

DG-00010

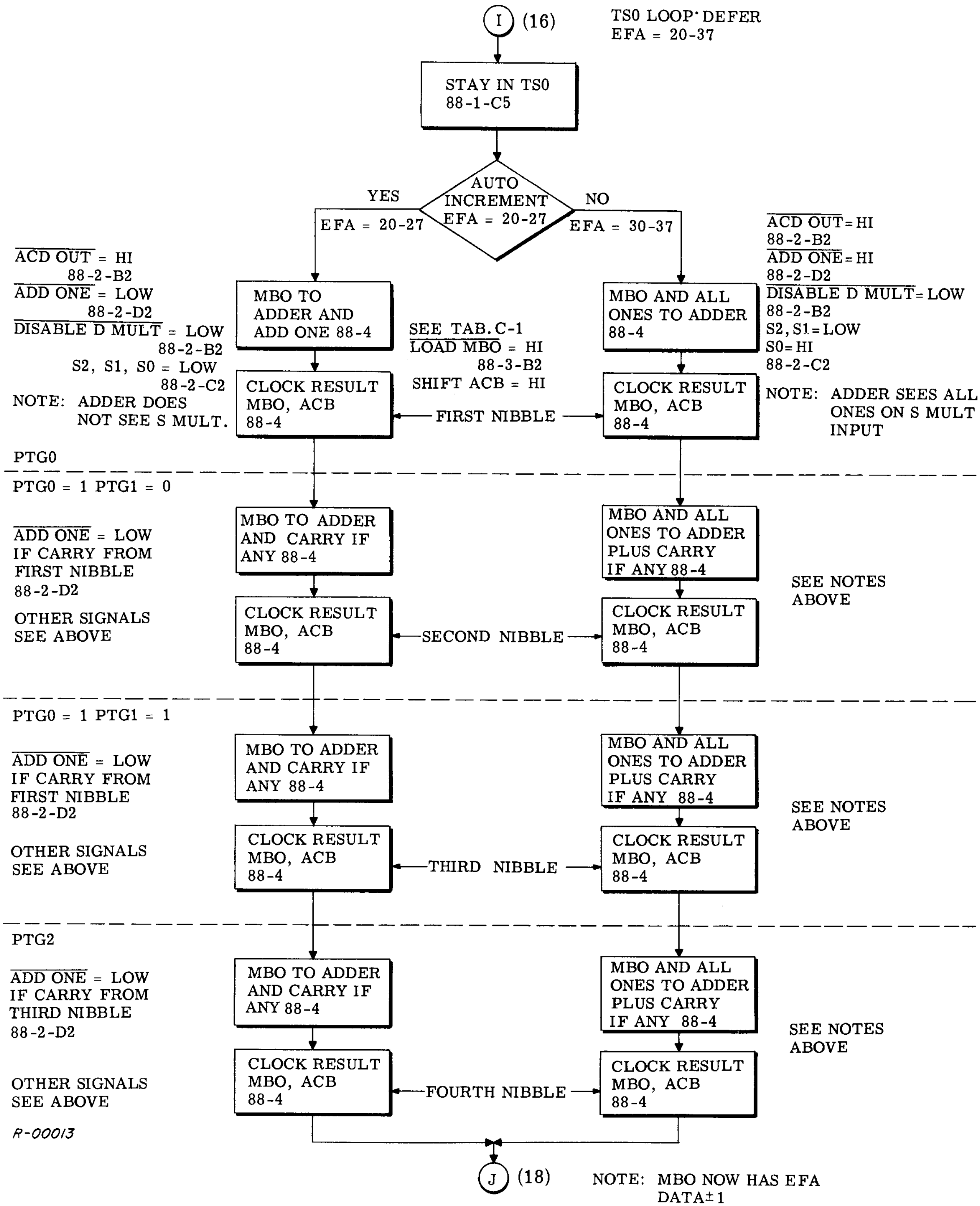

TS3· FETCH
IO INSTRUCTION
NOTE: ACD SELECTED BY IR BITS 3 & 4 88-2-C4
ADD ONE = HI 88-2-D2
S2, S1, S0 = LOW (NO S MULT) 88-2-C2
DISABLE D MULT = LOW 88-2-B2
ACD OUT = LOW 88-2-B2
FIRST NIBBLE
I/O
I/O OUT?
YES
I/O OUT
NO
I/O IN? 88-1-B5
YES
I/O IN
NO
I/O
SKIP
NOTE: TS3 OF FETCH HAS STUTTER. ONE EXTRA MEM CLOCK THAN CPU CLOCK-SEE TIMING FIG. C-1.
ACD → MBO 88-4
STA·E = HI 88-1-B4
FETCH· TS0 = HI 88-1-D4
PC IN = HI
PTG = 0
SET INPUT 88-1-B5
SEE ABOVE
ACD → MBO 88-4
SEE ABOVE
PTG0=1· PTG1=0
SECOND NIBBLE
SEE ABOVE
ACD → MBO 88-4
SEE ABOVE
PTG0=1· PTG1=1
THIRD NIBBLE
FOURTH NIBBLE
PTG5
SEE ABOVE
ACD → MBO 88-4
SEE ABOVE
SET SKIP INC 88-1-C7
R-00011
EXEC
(20)

TS0·DEFER

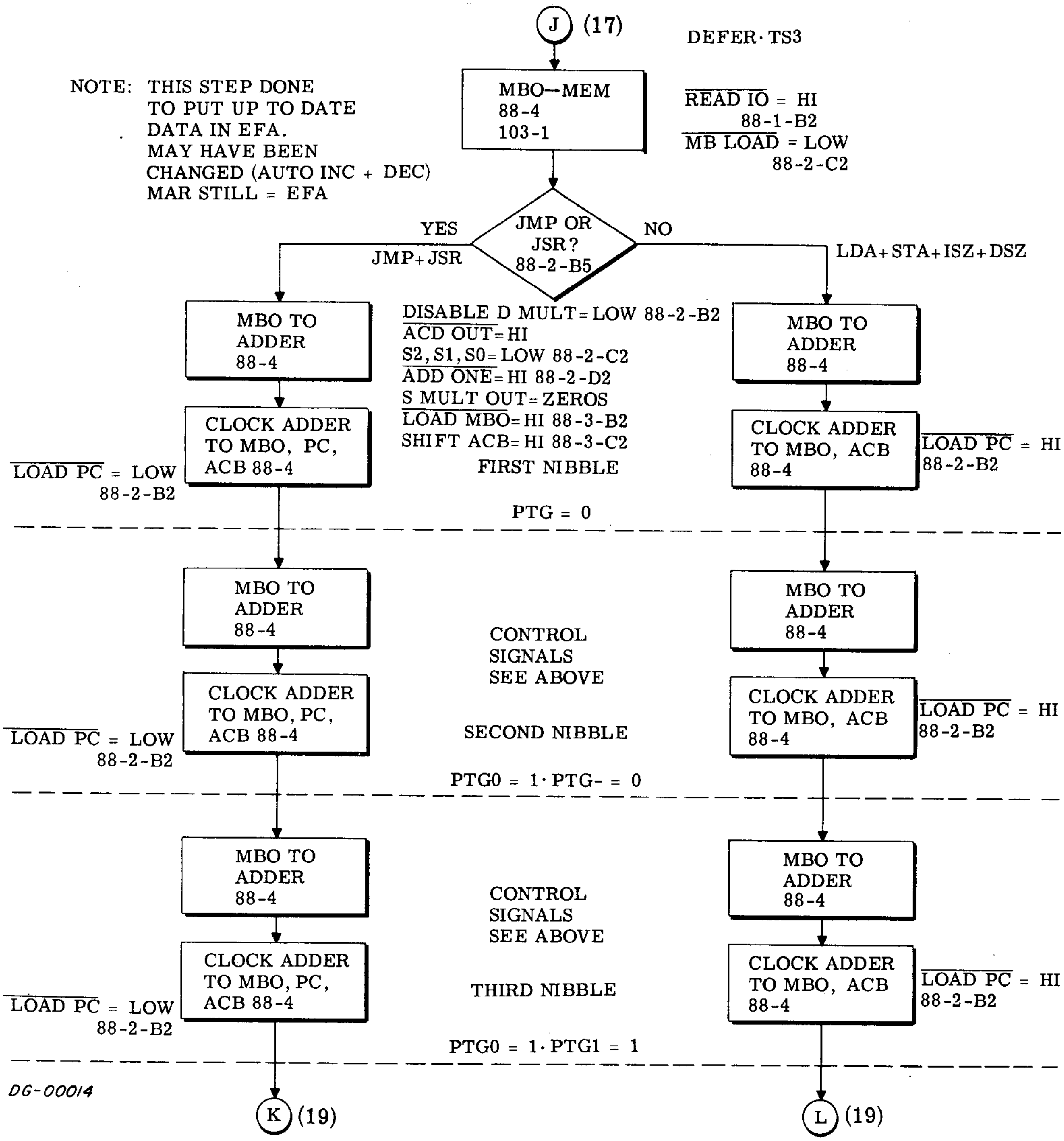


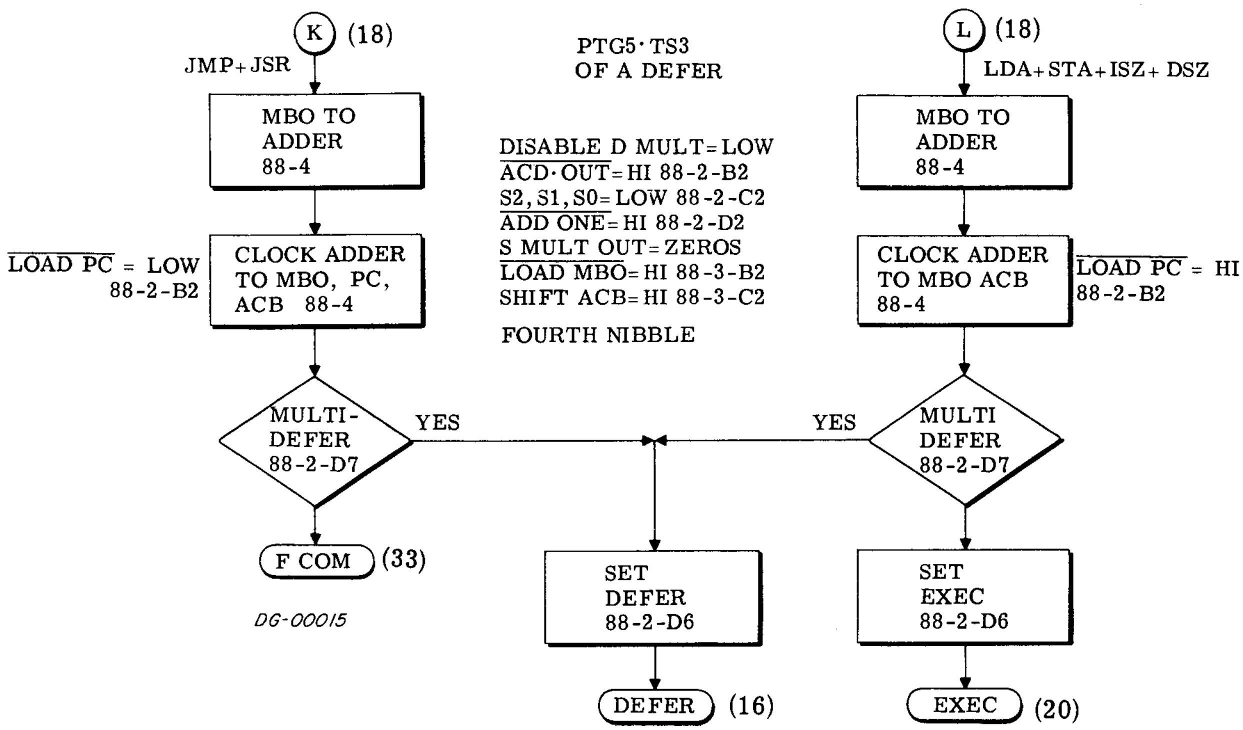
R-00012

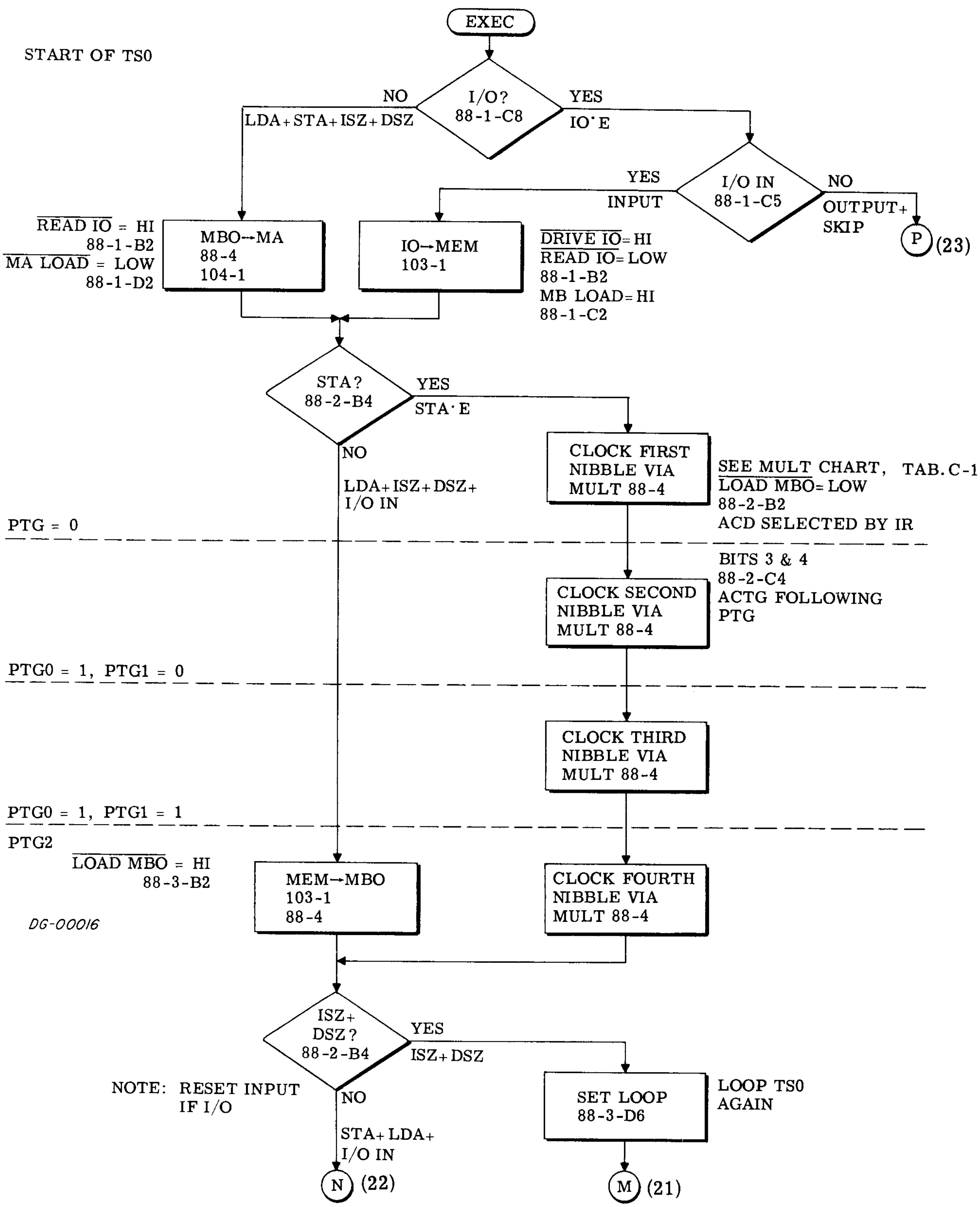
Rev. 01



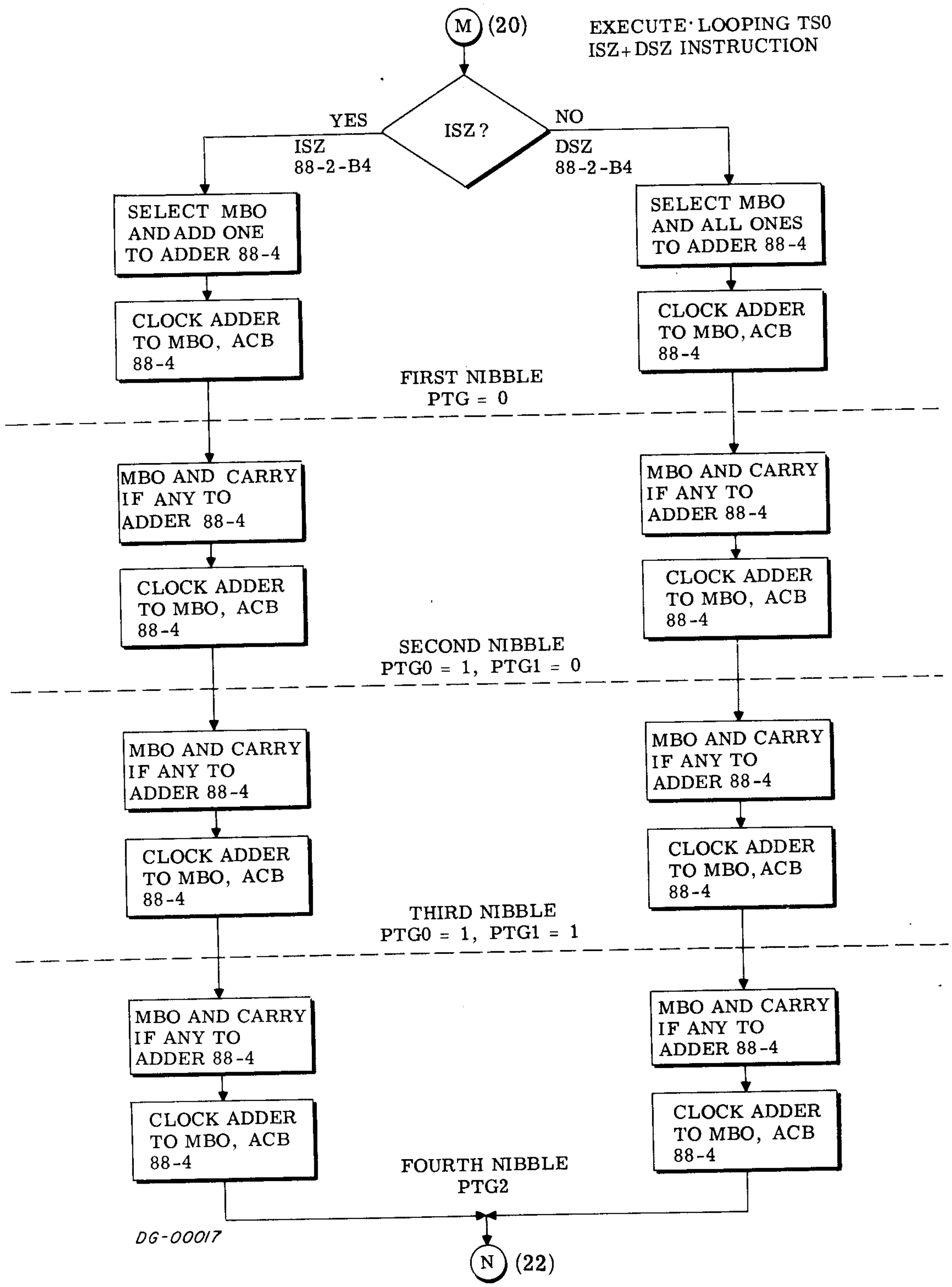
I (16)
TS0 LOOP·DEFER
EFA = 20-37
STAY IN TS0
88-1-C5
AUTO INCREMENT EFA = 20-27
YES
NO
EFA = 20-27
EFA = 30-37
ACD OUT = HI
88-2-B2
ADD ONE = LOW
88-2-D2
DISABLE D MULT = LOW
88-2-B2
S2, S1, S0 = LOW
88-2-C2
NOTE: ADDER DOES NOT SEE S MULT.
MBO TO ADDER AND ADD ONE 88-4
SEE TAB. C-1
LOAD MBO = HI
88-3-B2
SHIFT ACB = HI
MBO AND ALL ONES TO ADDER 88-4
ACD OUT = HI
88-2-B2
ADD ONE = HI
88-2-D2
DISABLE D MULT = LOW
88-2-B2
S2, S1 = LOW
S0 = HI
88-2-C2
NOTE: ADDER SEES ALL ONES ON S MULT INPUT
CLOCK RESULT MBO, ACB 88-4
FIRST NIBBLE
CLOCK RESULT MBO, ACB 88-4
PTG0
PTG0 = 1 PTG1 = 0
ADD ONE = LOW
IF CARRY FROM FIRST NIBBLE
88-2-D2
OTHER SIGNALS SEE ABOVE
MBO TO ADDER AND CARRY IF ANY 88-4
MBO AND ALL ONES TO ADDER PLUS CARRY IF ANY 88-4
SEE NOTES ABOVE
CLOCK RESULT MBO, ACB 88-4
SECOND NIBBLE
CLOCK RESULT MBO, ACB 88-4
PTG0 = 1 PTG1 = 1
ADD ONE = LOW
IF CARRY FROM FIRST NIBBLE
88-2-D2
OTHER SIGNALS SEE ABOVE
MBO TO ADDER AND CARRY IF ANY 88-4
MBO AND ALL ONES TO ADDER PLUS CARRY IF ANY 88-4
SEE NOTES ABOVE
CLOCK RESULT MBO, ACB 88-4
THIRD NIBBLE
CLOCK RESULT MBO, ACB 88-4
PTG2
ADD ONE = LOW
IF CARRY FROM THIRD NIBBLE
88-2-D2
OTHER SIGNALS SEE ABOVE
MBO TO ADDER AND CARRY IF ANY 88-4
MBO AND ALL ONES TO ADDER PLUS CARRY IF ANY 88-4
SEE NOTES ABOVE
CLOCK RESULT MBO, ACB 88-4
FOURTH NIBBLE
CLOCK RESULT MBO, ACB 88-4
R-00013
J (18)
NOTE: MBO NOW HAS EFA DATA±1


J (17)
DEFER·TS3
NOTE: THIS STEP DONE TO PUT UP TO DATE DATA IN EFA. MAY HAVE BEEN CHANGED (AUTO INC + DEC) MAR STILL = EFA
MBO→MEM 88-4 103-1
READ IO = HI 88-1-B2
MB LOAD = LOW 88-2-C2
JMP OR JSR? 88-2-B5
YES
JMP+JSR
NO
LDA+STA+ISZ+DSZ
MBO TO ADDER 88-4
DISABLE D MULT = LOW 88-2-B2
ACD OUT = HI
S2, S1, S0 = LOW 88-2-C2
ADD ONE = HI 88-2-D2
S MULT OUT = ZEROS
LOAD MBO = HI 88-3-B2
SHIFT ACB = HI 88-3-C2
FIRST NIBBLE
CLOCK ADDER TO MBO, PC, ACB 88-4
LOAD PC = LOW 88-2-B2
CLOCK ADDER TO MBO, ACB 88-4
LOAD PC = HI 88-2-B2
PTG = 0
CONTROL SIGNALS SEE ABOVE
SECOND NIBBLE
PTG0 = 1·PTG- = 0
THIRD NIBBLE
PTG0 = 1·PTG1 = 1
K (19)
L (19)
DG-00014

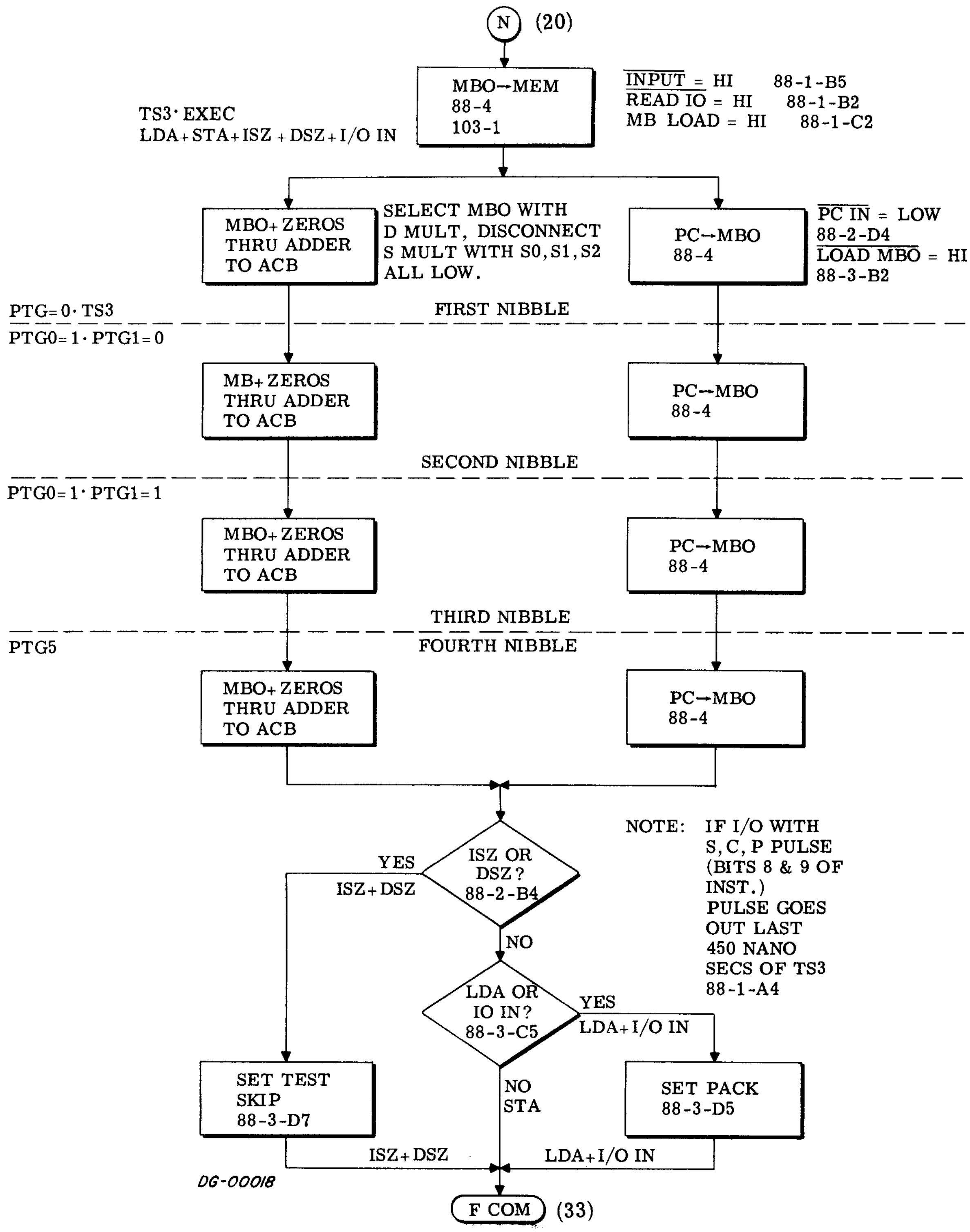

K (18)
JMP+JSR
MBO TO ADDER 88-4
LOAD PC = LOW 88-2-B2
CLOCK ADDER TO MBO, PC, ACB 88-4
MULTI-DEFER 88-2-D7
YES
F COM (33)
DG-00015
PTG5·TS3
OF A DEFER
DISABLE D MULT=LOW
ACD·OUT=HI 88-2-B2
S2, S1, S0=LOW 88-2-C2
ADD ONE=HI 88-2-D2
S MULT OUT=ZEROS
LOAD MBO=HI 88-3-B2
SHIFT ACB=HI 88-3-C2
FOURTH NIBBLE
SET DEFER 88-2-D6
DEFER (16)
L (18)
LDA+STA+ISZ+DSZ
MBO TO ADDER 88-4
CLOCK ADDER TO MBO ACB 88-4
LOAD PC = HI 88-2-B2
YES
MULTI DEFER 88-2-D7
SET EXEC 88-2-D6
EXEC (20)

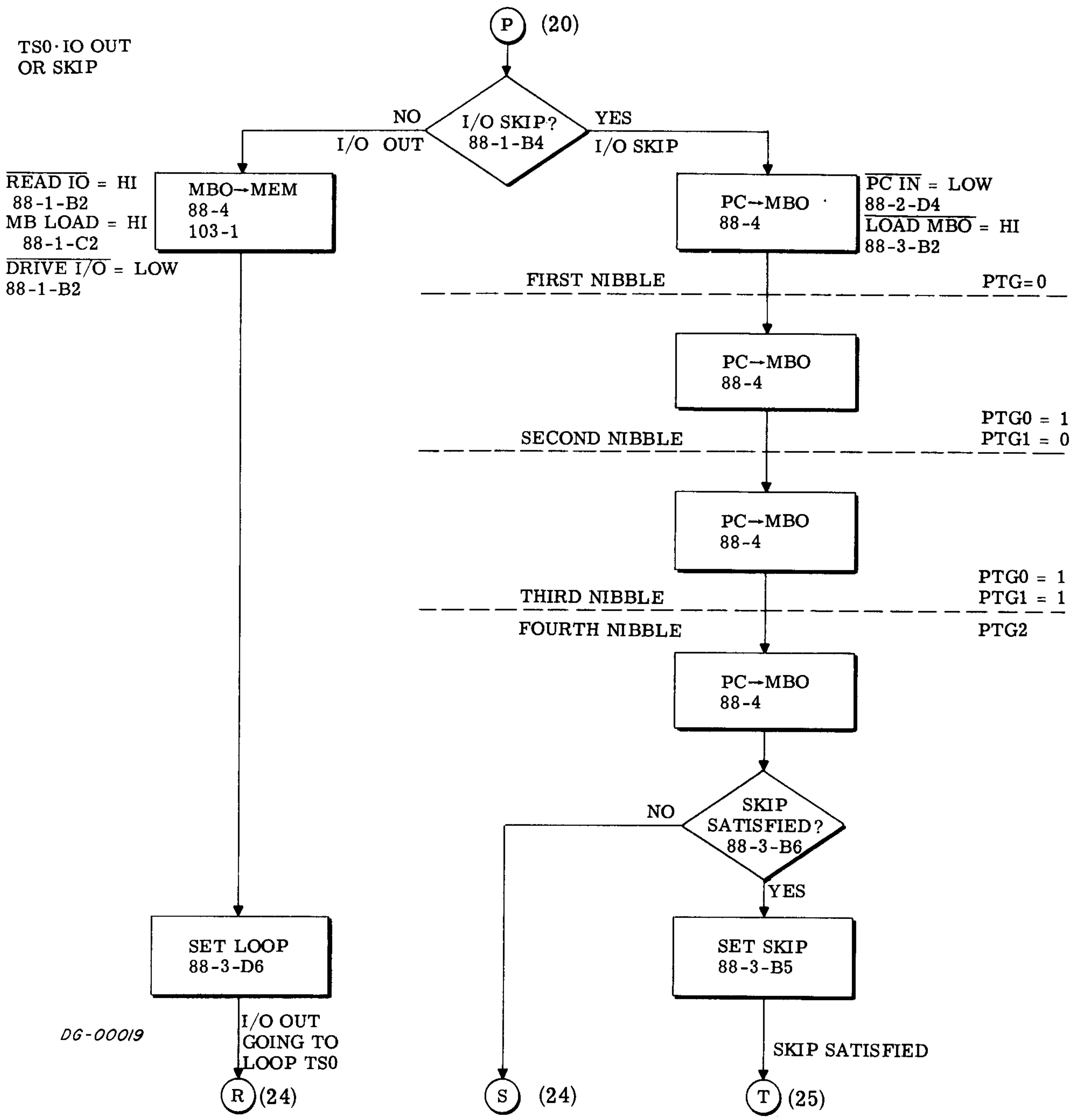

START OF TS0
EXEC
I/O?
88-1-C8
NO
LDA+STA+ISZ+DSZ
YES
IO·E
I/O IN
88-1-C5
YES
INPUT
NO
OUTPUT+
SKIP
P (23)
READ IO = HI
88-1-B2
MA LOAD = LOW
88-1-D2
MBO→MA
88-4
104-1
IO→MEM
103-1
DRIVE IO= HI
READ IO= LOW
88-1-B2
MB LOAD= HI
88-1-C2
STA?
88-2-B4
YES
STA·E
NO
LDA+ISZ+DSZ+
I/O IN
CLOCK FIRST
NIBBLE VIA
MULT 88-4
SEE MULT CHART, TAB. C-1
LOAD MBO= LOW
88-2-B2
ACD SELECTED BY IR
BITS 3 & 4
88-2-C4
ACTG FOLLOWING
PTG
PTG = 0
CLOCK SECOND
NIBBLE VIA
MULT 88-4
PTG0 = 1, PTG1 = 0
CLOCK THIRD
NIBBLE VIA
MULT 88-4
PTG0 = 1, PTG1 = 1
PTG2
LOAD MBO = HI
88-3-B2
MEM→MBO
103-1
88-4
CLOCK FOURTH
NIBBLE VIA
MULT 88-4
DG-00016
ISZ+
DSZ?
88-2-B4
YES
ISZ+DSZ
NOTE: RESET INPUT
IF I/O
NO
STA+LDA+
I/O IN
N (22)
SET LOOP
88-3-D6
LOOP TS0
AGAIN
M (21)

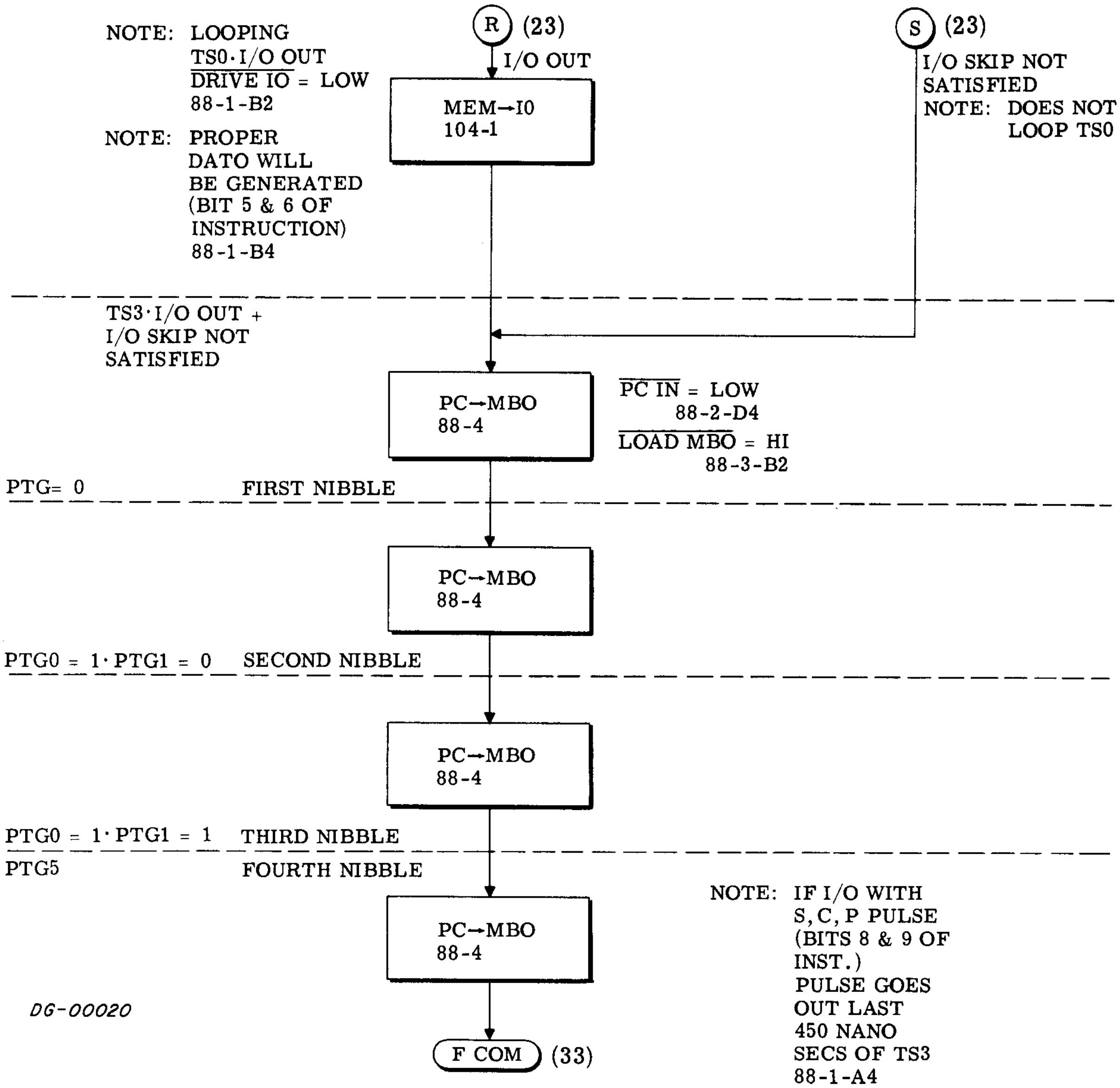
Rev. 01

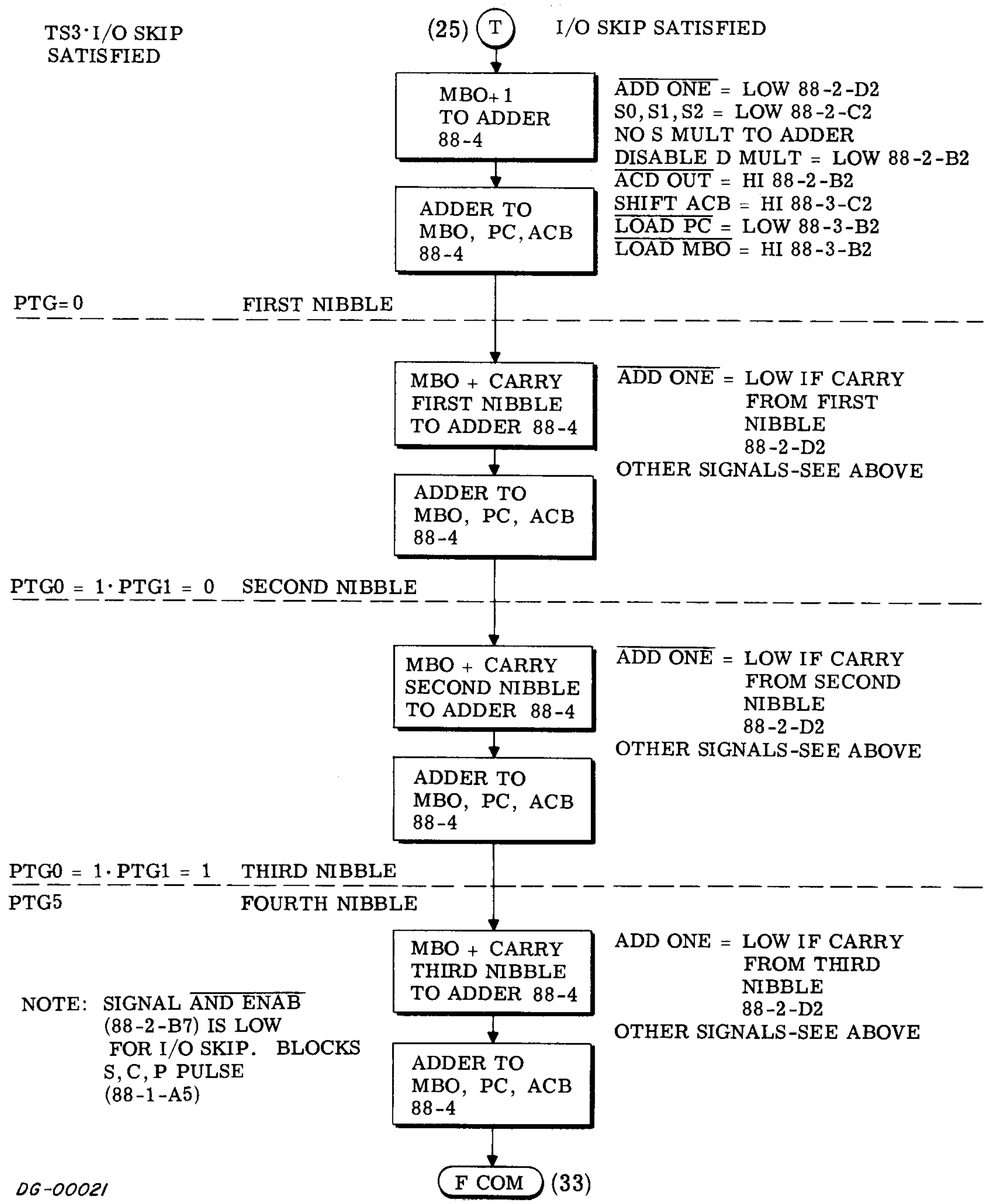

M (20)
EXECUTE· LOOPING TS0
ISZ+DSZ INSTRUCTION
ISZ?
YES
ISZ
88-2-B4
NO
DSZ
88-2-B4
SELECT MBO AND ADD ONE TO ADDER 88-4
SELECT MBO AND ALL ONES TO ADDER 88-4
CLOCK ADDER TO MBO, ACB 88-4
CLOCK ADDER TO MBO, ACB 88-4
FIRST NIBBLE
PTG = 0
MBO AND CARRY IF ANY TO ADDER 88-4
MBO AND CARRY IF ANY TO ADDER 88-4
CLOCK ADDER TO MBO, ACB 88-4
CLOCK ADDER TO MBO, ACB 88-4
SECOND NIBBLE
PTG0 = 1, PTG1 = 0
MBO AND CARRY IF ANY TO ADDER 88-4
MBO AND CARRY IF ANY TO ADDER 88-4
CLOCK ADDER TO MBO, ACB 88-4
CLOCK ADDER TO MBO, ACB 88-4
THIRD NIBBLE
PTG0 = 1, PTG1 = 1
MBO AND CARRY IF ANY TO ADDER 88-4
MBO AND CARRY IF ANY TO ADDER 88-4
CLOCK ADDER TO MBO, ACB 88-4
CLOCK ADDER TO MBO, ACB 88-4
FOURTH NIBBLE
PTG2
DG-00017
N (22)

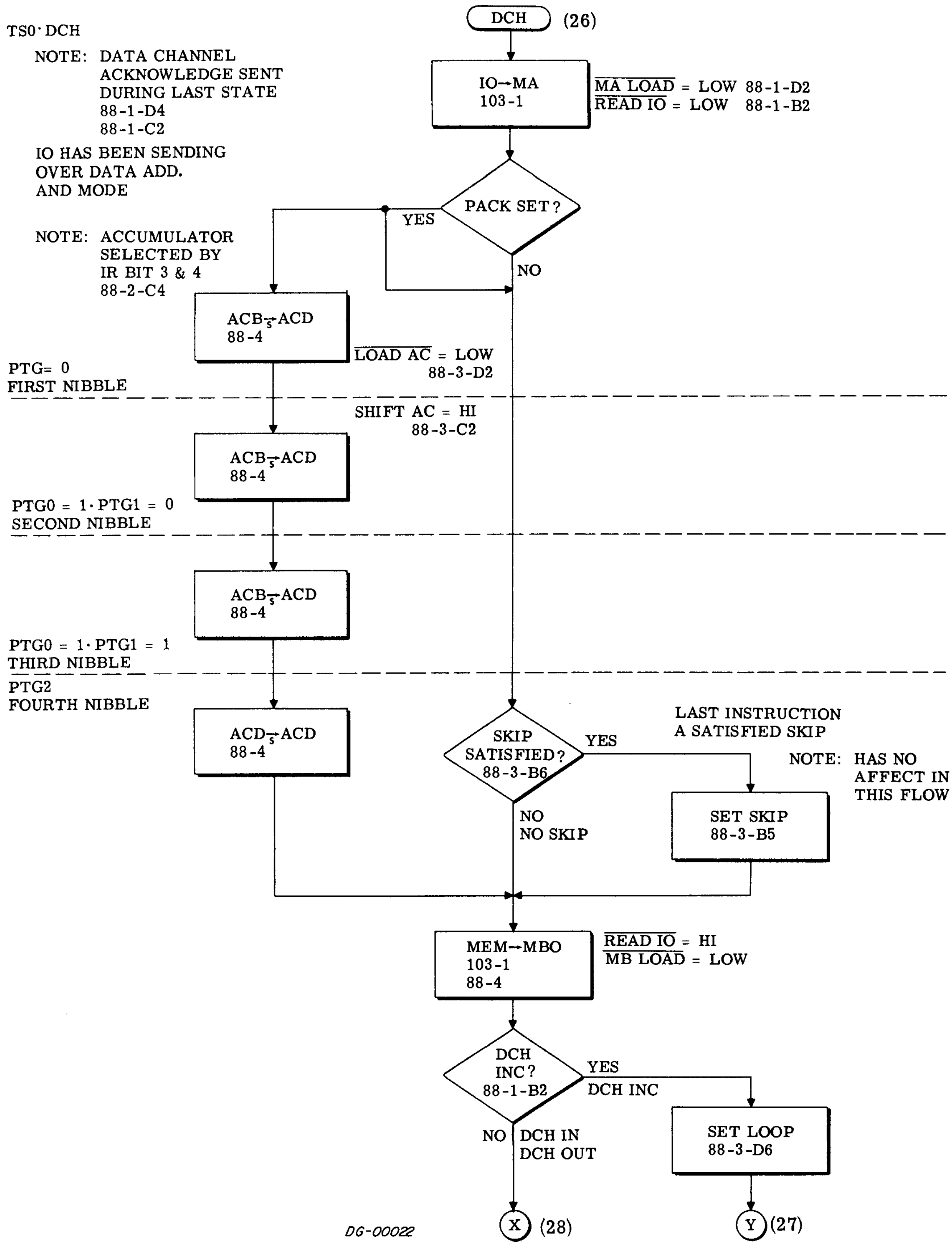
Rev. 02

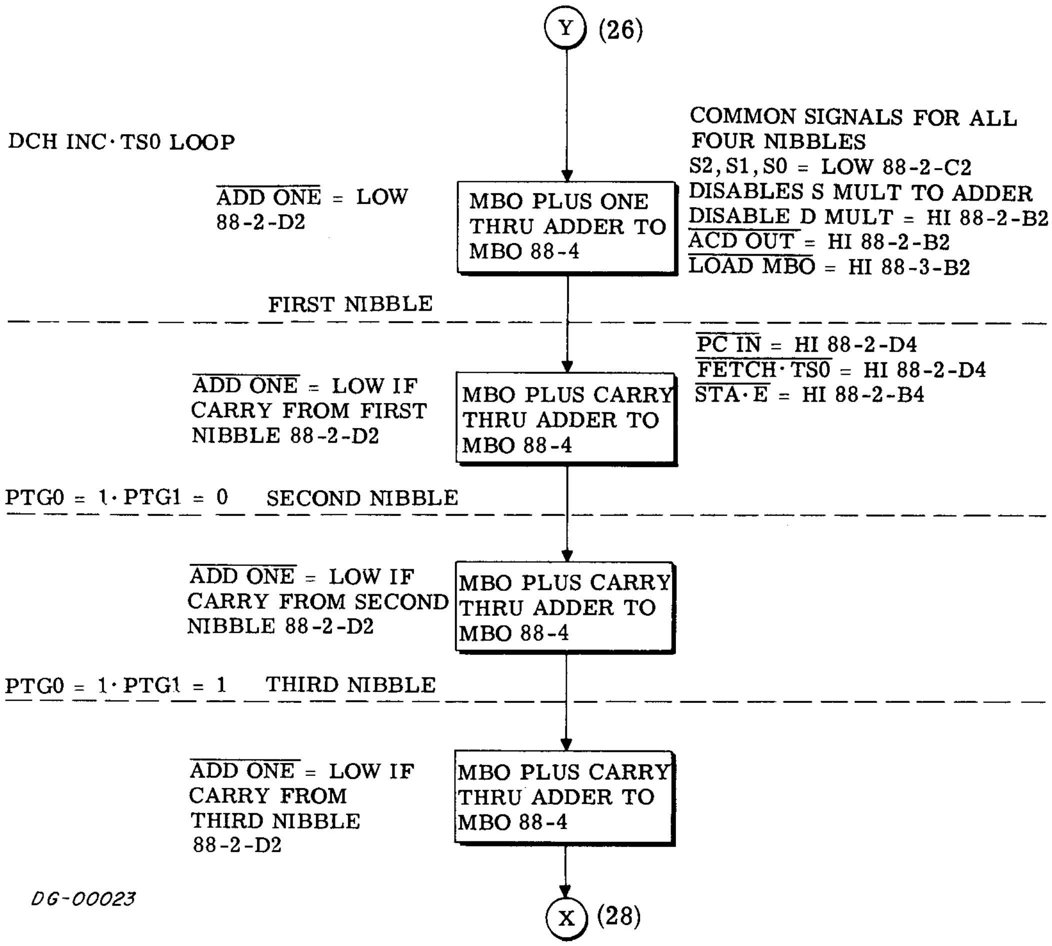

N
(20)
MBO→MEM
88-4
103-1
TS3· EXEC
LDA+STA+ISZ+DSZ+I/O IN
INPUT = HI 88-1-B5
READ IO = HI 88-1-B2
MB LOAD = HI 88-1-C2
MBO+ZEROS
THRU ADDER
TO ACB
SELECT MBO WITH
D MULT, DISCONNECT
S MULT WITH S0,S1,S2
ALL LOW.
PC→MBO
88-4
PC IN = LOW
88-2-D4
LOAD MBO = HI
88-3-B2
PTG=0· TS3
PTG0=1· PTG1=0
FIRST NIBBLE
MB+ZEROS
THRU ADDER
TO ACB
PC→MBO
88-4
SECOND NIBBLE
PTG0=1· PTG1=1
MBO+ZEROS
THRU ADDER
TO ACB
PC→MBO
88-4
THIRD NIBBLE
PTG5
FOURTH NIBBLE
MBO+ZEROS
THRU ADDER
TO ACB
PC→MBO
88-4
ISZ OR
DSZ?
88-2-B4
YES
ISZ+DSZ
NO
NOTE: IF I/O WITH
S,C,P PULSE
(BITS 8 & 9 OF
INST.)
PULSE GOES
OUT LAST
450 NANO
SECS OF TS3
88-1-A4
LDA OR
IO IN?
88-3-C5
YES
LDA+I/O IN
NO
STA
SET TEST
SKIP
88-3-D7
SET PACK
88-3-D5
ISZ+DSZ
LDA+I/O IN
DG-00018
F COM
(33)

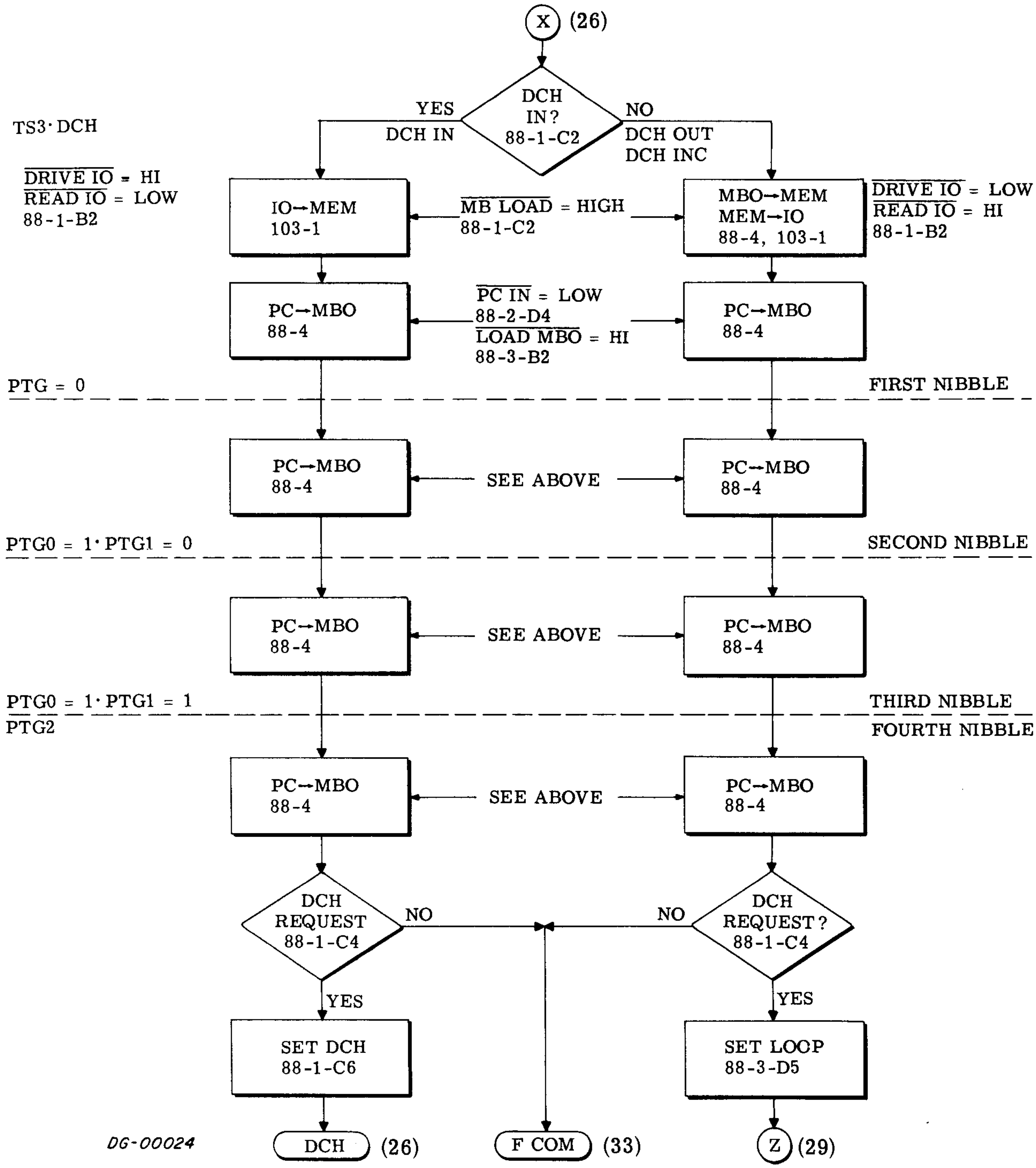

P (20)
TS0·IO OUT
OR SKIP
I/O SKIP?
88-1-B4
NO
I/O OUT
YES
I/O SKIP
READ IO = HI
88-1-B2
MB LOAD = HI
88-1-C2
DRIVE I/O = LOW
88-1-B2
MBO→MEM
88-4
103-1
PC→MBO
88-4
PC IN = LOW
88-2-D4
LOAD MBO = HI
88-3-B2
FIRST NIBBLE
PTG=0
PC→MBO
88-4
SECOND NIBBLE
PTG0 = 1
PTG1 = 0
PC→MBO
88-4
THIRD NIBBLE
PTG0 = 1
PTG1 = 1
FOURTH NIBBLE
PTG2
PC→MBO
88-4
SKIP
SATISFIED?
88-3-B6
NO
YES
SET LOOP
88-3-D6
SET SKIP
88-3-B5
I/O OUT
GOING TO
LOOP TS0
SKIP SATISFIED
R (24)
S (24)
T (25)
DG-00019


NOTE: LOOPING
TS0·I/O OUT
DRIVE IO = LOW
88-1-B2
NOTE: PROPER
DATO WILL
BE GENERATED
(BIT 5 & 6 OF
INSTRUCTION)
88-1-B4
R (23)
I/O OUT
MEM→I0
104-1
S (23)
I/O SKIP NOT
SATISFIED
NOTE: DOES NOT
LOOP TS0
TS3·I/O OUT +
I/O SKIP NOT
SATISFIED
PC→MBO
88-4
PC IN = LOW
88-2-D4
LOAD MBO = HI
88-3-B2
PTG= 0
FIRST NIBBLE
PC→MBO
88-4
PTG0 = 1·PTG1 = 0
SECOND NIBBLE
PC→MBO
88-4
PTG0 = 1·PTG1 = 1
THIRD NIBBLE
PTG5
FOURTH NIBBLE
PC→MBO
88-4
NOTE: IF I/O WITH
S, C, P PULSE
(BITS 8 & 9 OF
INST.)
PULSE GOES
OUT LAST
450 NANO
SECS OF TS3
88-1-A4
DG-00020
F COM (33)


TS3·I/O SKIP SATISFIED
(25) T
I/O SKIP SATISFIED
MBO+1 TO ADDER 88-4
ADD ONE = LOW 88-2-D2
S0, S1, S2 = LOW 88-2-C2
NO S MULT TO ADDER
DISABLE D MULT = LOW 88-2-B2
ACD OUT = HI 88-2-B2
SHIFT ACB = HI 88-3-C2
LOAD PC = LOW 88-3-B2
LOAD MBO = HI 88-3-B2
ADDER TO MBO, PC, ACB 88-4
PTG=0
FIRST NIBBLE
MBO + CARRY FIRST NIBBLE TO ADDER 88-4
ADD ONE = LOW IF CARRY FROM FIRST NIBBLE 88-2-D2
OTHER SIGNALS-SEE ABOVE
ADDER TO MBO, PC, ACB 88-4
PTG0 = 1·PTG1 = 0
SECOND NIBBLE
MBO + CARRY SECOND NIBBLE TO ADDER 88-4
ADD ONE = LOW IF CARRY FROM SECOND NIBBLE 88-2-D2
OTHER SIGNALS-SEE ABOVE
ADDER TO MBO, PC, ACB 88-4
PTG0 = 1·PTG1 = 1
THIRD NIBBLE
PTG5
FOURTH NIBBLE
MBO + CARRY THIRD NIBBLE TO ADDER 88-4
ADD ONE = LOW IF CARRY FROM THIRD NIBBLE 88-2-D2
OTHER SIGNALS-SEE ABOVE
NOTE: SIGNAL AND ENAB (88-2-B7) IS LOW FOR I/O SKIP. BLOCKS S, C, P PULSE (88-1-A5)
ADDER TO MBO, PC, ACB 88-4
F COM (33)
DG-00021


DCH (26)
TS0·DCH
NOTE: DATA CHANNEL ACKNOWLEDGE SENT DURING LAST STATE 88-1-D4 88-1-C2
IO HAS BEEN SENDING OVER DATA ADD. AND MODE
IO→MA 103-1
MA LOAD = LOW 88-1-D2
READ IO = LOW 88-1-B2
PACK SET?
YES
NO
NOTE: ACCUMULATOR SELECTED BY IR BIT 3 & 4 88-2-C4
ACB→ACD 88-4
LOAD AC = LOW 88-3-D2
PTG= 0 FIRST NIBBLE
SHIFT AC = HI 88-3-C2
ACB→ACD 88-4
PTG0 = 1·PTG1 = 0 SECOND NIBBLE
ACB→ACD 88-4
PTG0 = 1·PTG1 = 1 THIRD NIBBLE
PTG2 FOURTH NIBBLE
ACD→ACD 88-4
SKIP SATISFIED? 88-3-B6
YES
LAST INSTRUCTION A SATISFIED SKIP
NOTE: HAS NO AFFECT IN THIS FLOW
SET SKIP 88-3-B5
NO NO SKIP
MEM→MBO 103-1 88-4
READ IO = HI
MB LOAD = LOW
DCH INC? 88-1-B2
YES DCH INC
SET LOOP 88-3-D6
NO
DCH IN DCH OUT
X (28)
Y (27)
DG-00022


Y (26)
DCH INC·TS0 LOOP
COMMON SIGNALS FOR ALL FOUR NIBBLES
S2, S1, S0 = LOW 88-2-C2
DISABLES S MULT TO ADDER
DISABLE D MULT = HI 88-2-B2
ACD OUT = HI 88-2-B2
LOAD MBO = HI 88-3-B2
ADD ONE = LOW 88-2-D2
MBO PLUS ONE THRU ADDER TO MBO 88-4
FIRST NIBBLE
PC IN = HI 88-2-D4
FETCH·TS0 = HI 88-2-D4
STA·E = HI 88-2-B4
ADD ONE = LOW IF CARRY FROM FIRST NIBBLE 88-2-D2
MBO PLUS CARRY THRU ADDER TO MBO 88-4
PTG0 = 1·PTG1 = 0
SECOND NIBBLE
ADD ONE = LOW IF CARRY FROM SECOND NIBBLE 88-2-D2
MBO PLUS CARRY THRU ADDER TO MBO 88-4
PTG0 = 1·PTG1 = 1
THIRD NIBBLE
ADD ONE = LOW IF CARRY FROM THIRD NIBBLE 88-2-D2
MBO PLUS CARRY THRU ADDER TO MBO 88-4
DG-00023
X (28)


X (26)
DCH IN? 88-1-C2
YES
DCH IN
NO
DCH OUT
DCH INC
TS3·DCH
DRIVE IO = HI
READ IO = LOW
88-1-B2
IO→MEM 103-1
MB LOAD = HIGH
88-1-C2
MBO→MEM
MEM→IO
88-4, 103-1
DRIVE IO = LOW
READ IO = HI
88-1-B2
PC→MBO 88-4
PC IN = LOW
88-2-D4
LOAD MBO = HI
88-3-B2
PC→MBO 88-4
PTG = 0
FIRST NIBBLE
PC→MBO 88-4
SEE ABOVE
PC→MBO 88-4
PTG0 = 1·PTG1 = 0
SECOND NIBBLE
PC→MBO 88-4
SEE ABOVE
PC→MBO 88-4
PTG0 = 1·PTG1 = 1
PTG2
THIRD NIBBLE
FOURTH NIBBLE
PC→MBO 88-4
SEE ABOVE
PC→MBO 88-4
DCH REQUEST 88-1-C4
NO
NO
DCH REQUEST? 88-1-C4
YES
YES
SET DCH 88-1-C6
SET LOOP 88-3-D5
DG-00024
DCH (26)
F COM (33)
Z (29)

Z (28)

DCH · LOOPING TS3

PTG = 0

600 NANO SECS. DEAD TIME

PTG5

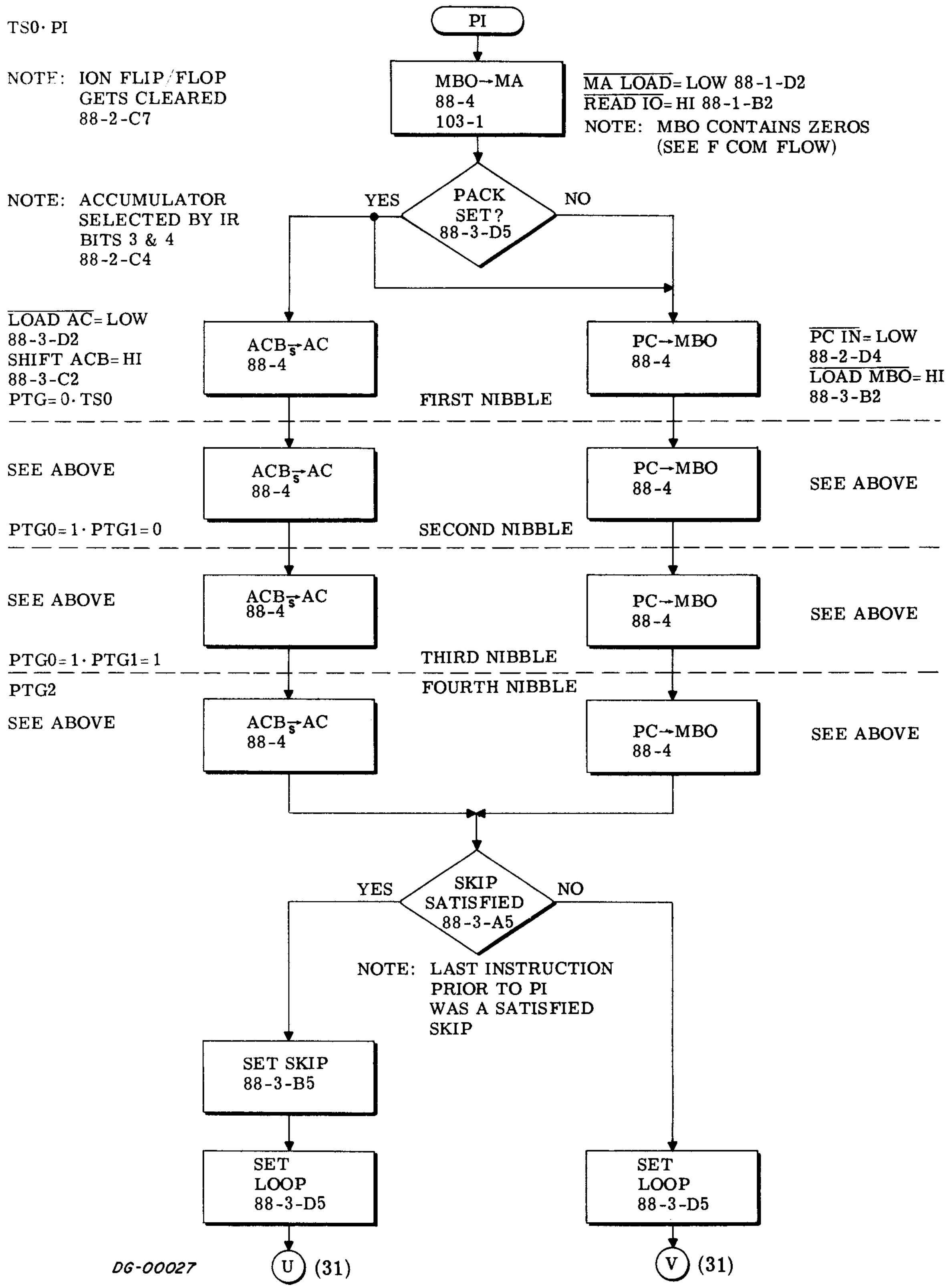
THIS PAGE COULD BE ENTERED ONE OF TWO WAYS.

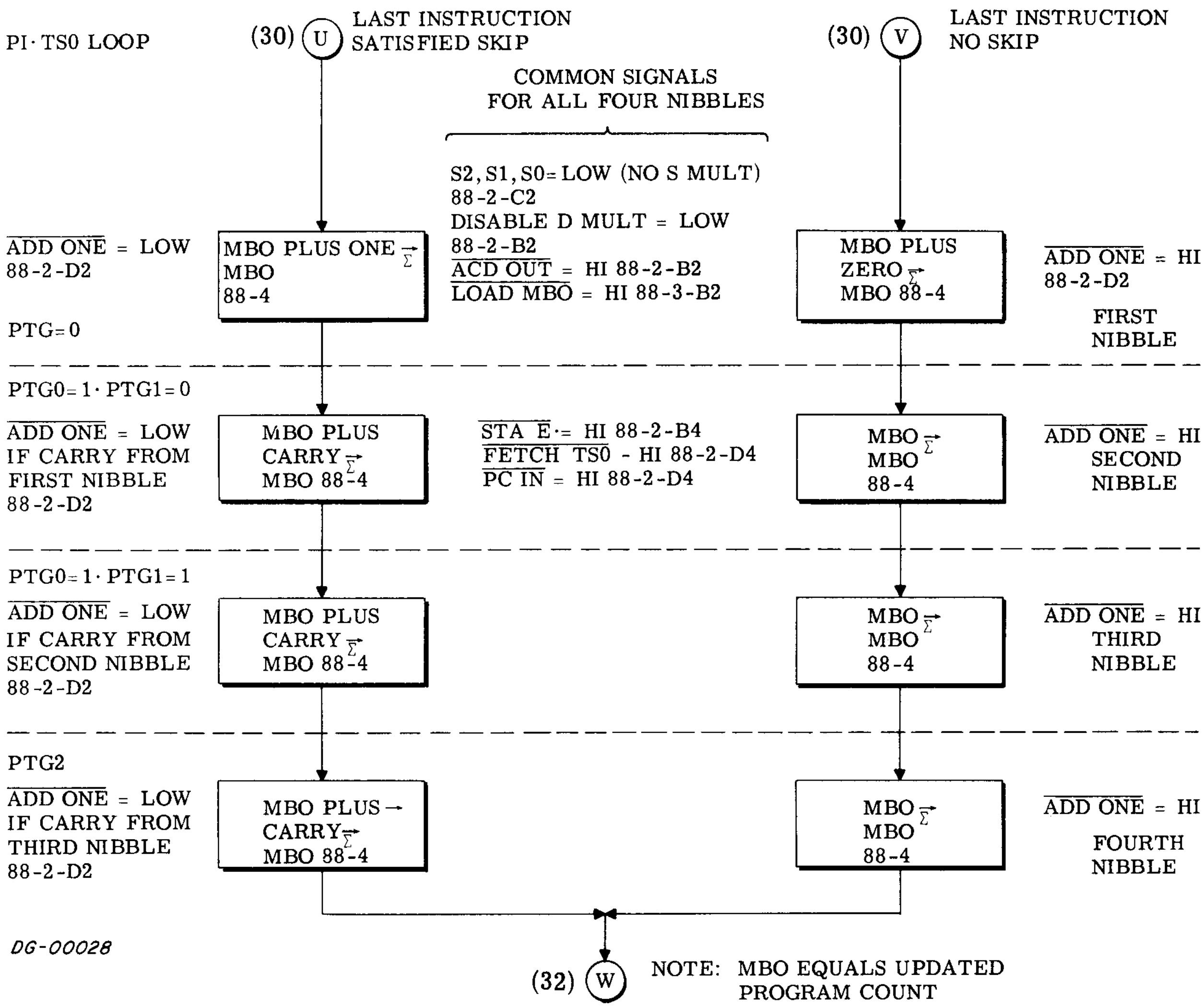
1. A DATA CHANNEL OUT FOLLOWED BY ANY TYPE OF DATA CHANNEL REQUEST: LOOPING TS3 ALLOWS THE NEW MAIN MEMORY ADDRESS TO TAKE THE PLACE OF THE DATA OUT ON THE DATA BUS.

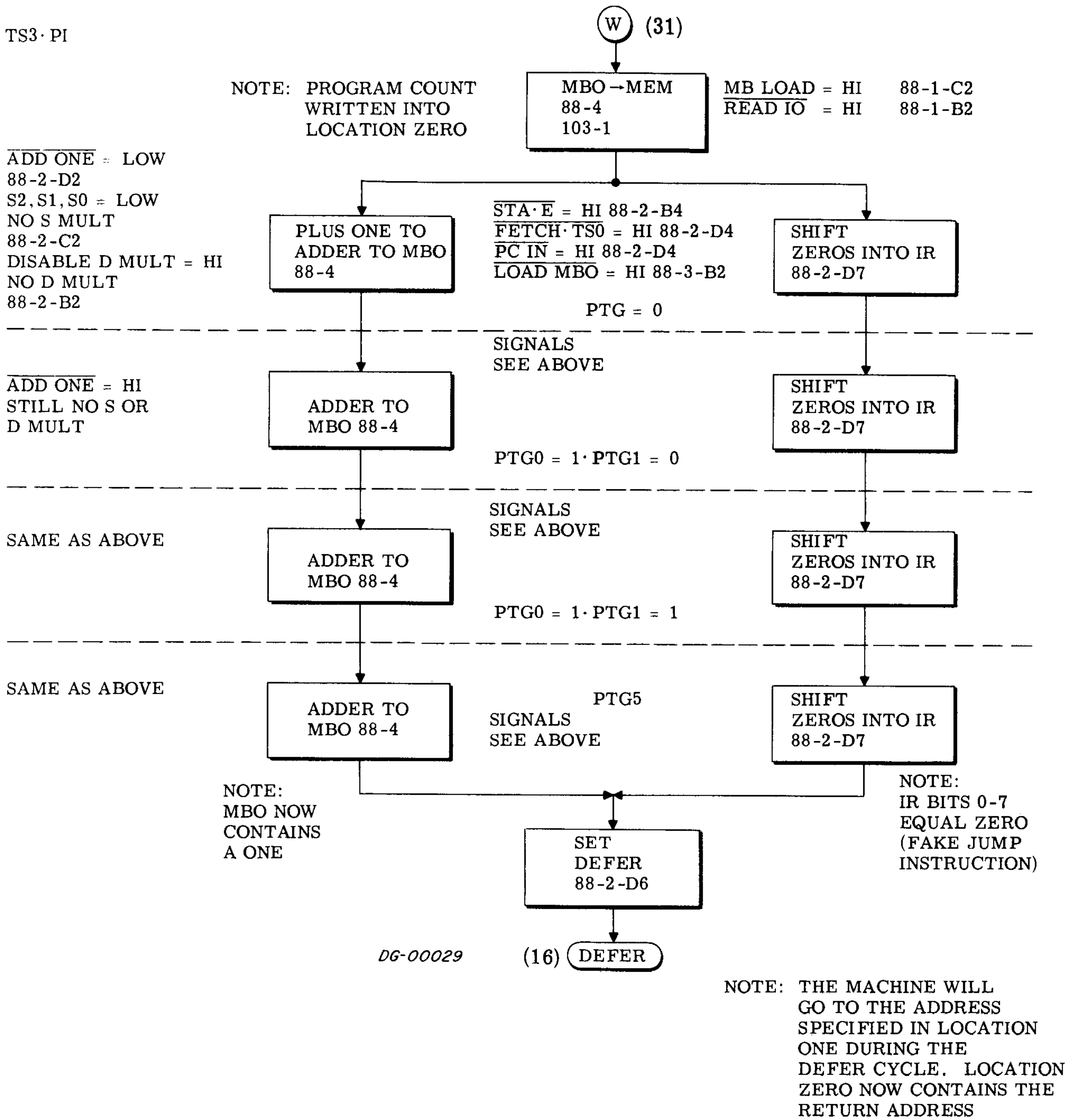
2. A DATA CHANNEL INCREMENT FOLLOWED BY ANY TYPE OF DATA CHANNEL REQUEST: LOOPING TS3 ALLOWS THE NEW MAIN MEMORY ADDRESS TO TAKE THE PLACE OF THE DATA OUT ON THE DATA BUS.

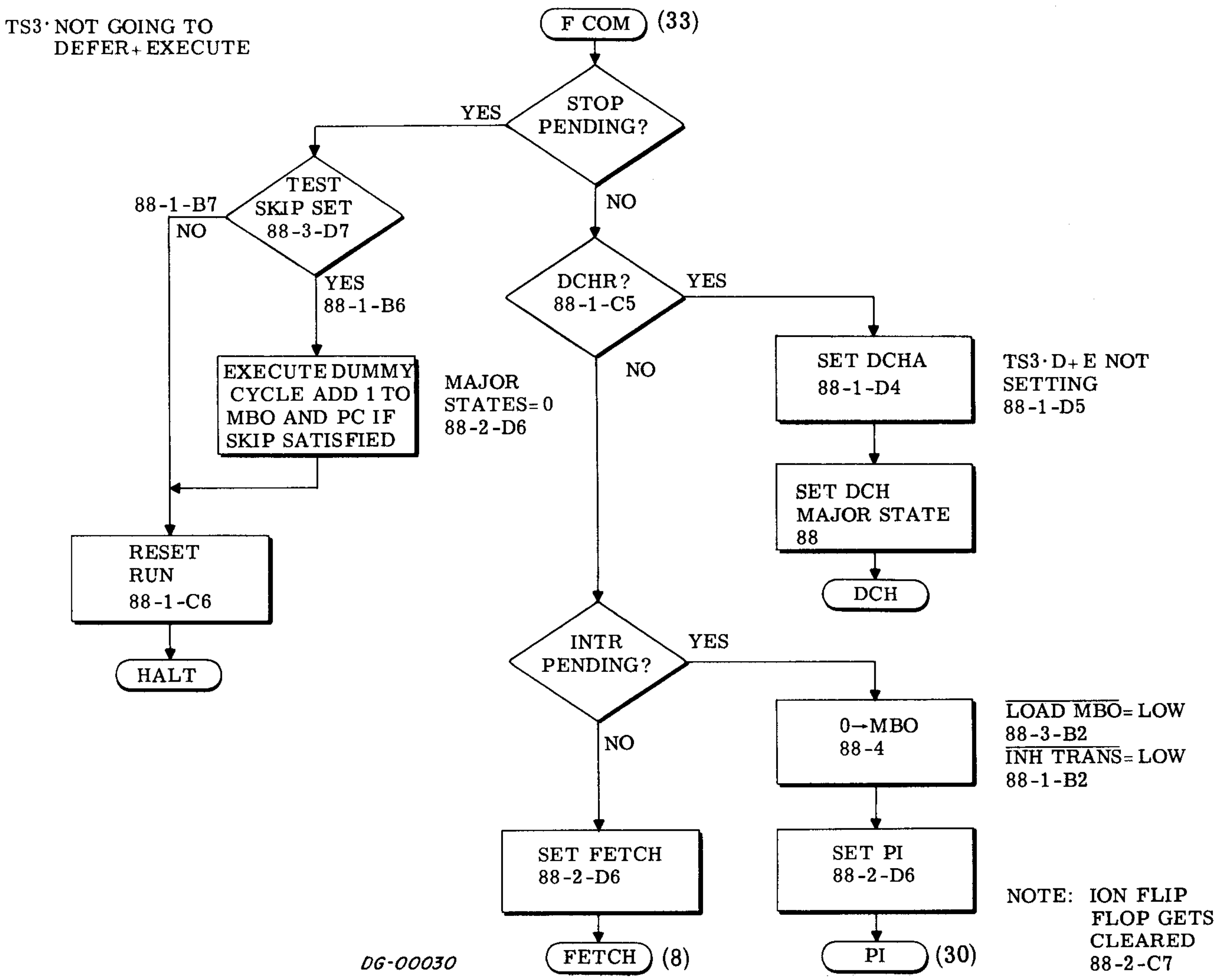
DCH (26)

DG-00026


TS0· PI
PI
NOTE: ION FLIP/FLOP GETS CLEARED 88-2-C7
MBO→MA 88-4 103-1
MA LOAD= LOW 88-1-D2
READ IO= HI 88-1-B2
NOTE: MBO CONTAINS ZEROS (SEE F COM FLOW)
NOTE: ACCUMULATOR SELECTED BY IR BITS 3 & 4 88-2-C4
YES
PACK SET? 88-3-D5
NO
LOAD AC= LOW 88-3-D2
SHIFT ACB= HI 88-3-C2
PTG= 0· TS0
ACB→AC 88-4
PC→MBO 88-4
PC IN= LOW 88-2-D4
LOAD MBO= HI 88-3-B2
FIRST NIBBLE
SEE ABOVE
ACB→AC 88-4
PC→MBO 88-4
SEE ABOVE
PTG0= 1· PTG1= 0
SECOND NIBBLE
SEE ABOVE
ACB→AC 88-4
PC→MBO 88-4
SEE ABOVE
PTG0= 1· PTG1= 1
THIRD NIBBLE
PTG2
FOURTH NIBBLE
SEE ABOVE
ACB→AC 88-4
PC→MBO 88-4
SEE ABOVE
YES
SKIP SATISFIED 88-3-A5
NO
NOTE: LAST INSTRUCTION PRIOR TO PI WAS A SATISFIED SKIP
SET SKIP 88-3-B5
SET LOOP 88-3-D5
SET LOOP 88-3-D5
DG-00027
U (31)
V (31)


PI·TS0 LOOP
(30) U LAST INSTRUCTION SATISFIED SKIP
(30) V LAST INSTRUCTION NO SKIP
COMMON SIGNALS FOR ALL FOUR NIBBLES
S2, S1, S0 = LOW (NO S MULT) 88-2-C2
DISABLE D MULT = LOW 88-2-B2
ACD OUT = HI 88-2-B2
LOAD MBO = HI 88-3-B2
ADD ONE = LOW 88-2-D2
PTG=0
MBO PLUS ONE → Σ MBO 88-4
MBO PLUS ZERO → Σ MBO 88-4
ADD ONE = HI 88-2-D2
FIRST NIBBLE
PTG0=1·PTG1=0
ADD ONE = LOW IF CARRY FROM FIRST NIBBLE 88-2-D2
MBO PLUS CARRY → Σ MBO 88-4
STA E· = HI 88-2-B4
FETCH TS0 - HI 88-2-D4
PC IN = HI 88-2-D4
MBO → Σ MBO 88-4
ADD ONE = HI SECOND NIBBLE
PTG0=1·PTG1=1
ADD ONE = LOW IF CARRY FROM SECOND NIBBLE 88-2-D2
MBO PLUS CARRY → Σ MBO 88-4
MBO → Σ MBO 88-4
ADD ONE = HI THIRD NIBBLE
PTG2
ADD ONE = LOW IF CARRY FROM THIRD NIBBLE 88-2-D2
MBO PLUS → CARRY → Σ MBO 88-4
MBO → Σ MBO 88-4
ADD ONE = HI FOURTH NIBBLE
DG-00028
(32) W
NOTE: MBO EQUALS UPDATED PROGRAM COUNT


TS3 · PI
W (31)
NOTE: PROGRAM COUNT WRITTEN INTO LOCATION ZERO
MBO → MEM 88-4 103-1
MB LOAD = HI 88-1-C2
READ IO = HI 88-1-B2
ADD ONE = LOW 88-2-D2
S2, S1, S0 = LOW
NO S MULT 88-2-C2
DISABLE D MULT = HI
NO D MULT 88-2-B2
PLUS ONE TO ADDER TO MBO 88-4
STA · E = HI 88-2-B4
FETCH · TS0 = HI 88-2-D4
PC IN = HI 88-2-D4
LOAD MBO = HI 88-3-B2
SHIFT ZEROS INTO IR 88-2-D7
PTG = 0
SIGNALS SEE ABOVE
ADD ONE = HI
STILL NO S OR D MULT
ADDER TO MBO 88-4
SHIFT ZEROS INTO IR 88-2-D7
PTG0 = 1 · PTG1 = 0
SAME AS ABOVE
SIGNALS SEE ABOVE
ADDER TO MBO 88-4
SHIFT ZEROS INTO IR 88-2-D7
PTG0 = 1 · PTG1 = 1
SAME AS ABOVE
ADDER TO MBO 88-4
PTG5
SIGNALS SEE ABOVE
SHIFT ZEROS INTO IR 88-2-D7
NOTE: MBO NOW CONTAINS A ONE
NOTE: IR BITS 0-7 EQUAL ZERO (FAKE JUMP INSTRUCTION)
SET DEFER 88-2-D6
DG-00029
(16) DEFER
NOTE: THE MACHINE WILL GO TO THE ADDRESS SPECIFIED IN LOCATION ONE DURING THE DEFER CYCLE. LOCATION ZERO NOW CONTAINS THE RETURN ADDRESS


TS3· NOT GOING TO DEFER+ EXECUTE
F COM (33)
STOP PENDING?
YES
NO
TEST SKIP SET 88-3-D7
88-1-B7
NO
YES 88-1-B6
EXECUTE DUMMY CYCLE ADD 1 TO MBO AND PC IF SKIP SATISFIED
MAJOR STATES=0 88-2-D6
RESET RUN 88-1-C6
HALT
DCHR? 88-1-C5
YES
NO
SET DCHA 88-1-D4
TS3·D+E NOT SETTING 88-1-D5
SET DCH MAJOR STATE 88
DCH
INTR PENDING?
YES
NO
0→MBO 88-4
LOAD MBO= LOW 88-3-B2
INH TRANS= LOW 88-1-B2
SET FETCH 88-2-D6
SET PI 88-2-D6
NOTE: ION FLIP FLOP GETS CLEARED 88-2-C7
FETCH (8)
PI (30)
DG-00030

Rev. 01

Table C-1

Adder and Multiplexer Control Signals During EFA Instructions

	S0 *	S1	S2	DISABLE D MULT	EFA· PTG1 *	ACD OUT
REL· +(P6)	H/L	L	L	L	H/L	H
REL· -(P6)	H/H	L	L	L	H/L	H
BASE +(AC2)(AC3)	H/L	L	L	L	H/L	L
BASE -(AC2)(AC3)	H/H	L	L	L	H/L	L
PAGE ZERO	H/L	L	L	H	H/L	H (circled: DON'T CARE)

* H for FIRST TWO NIBBLES; L for LAST TWO NIBBLES

DG-00049

Table C-2

Adder Control Signals During ALC Instructions (TS3)

IR BITS 5 6 7	FUNCTION	IR5(1)=LOW DISABLE D MULT	ACD OUT (overbar)	EFA · PTG1 (overbar)	IR6(1) =HI S0	S1	IR6(0) =HI S2	IR7(1) =LOW ADD ONE (overbar)
0 0 0	COMPLEMENT	H	L	L	L	H	H	H
0 0 1	NEGATE	H	L	L	L	H	H	L
0 1 0	MOVE	H	L	L	H	L	L	H
0 1 1	INCREMENT	H	L	L	H	L	L	L
1 0 0	ADD COMPLEMENT	L	L	L	L	H	H	H
1 0 1	SUBTRACT	L	L	L	L	H	H	L
1 1 0	ADD	L	L	L	H	L	L	H
1 1 1	AND	L	L	L	H	H	L	L
88-2 A7 & 6		88-2-B2	88-2 B2	88-2 A2	88-2 C2	88-2 C2	88-2 C2	88-2 D2

DG-00048

Table C-3

Carry Chart For ALC Instruction

PRIOR TO INSTRUCTION	IR 10	BITS 11	OVERFLOW OCCURRED?	CARRY AT COMPLETION
CARRY RESET	0	0	NO	RESET
CARRY RESET	0	0	YES	SET
CARRY SET	0	0	NO	SET
CARRY SET	0	0	YES	RESET
CARRY RESET	0	1	NO	RESET
CARRY RESET	0	1	YES	SET
CARRY SET	0	1	NO	RESET
CARRY SET	0	1	YES	SET
CARRY RESET	1	0	NO	SET
CARRY RESET	1	0	YES	RESET
CARRY SET	1	0	NO	SET
CARRY SET	1	0	YES	RESET
CARRY RESET	1	1	NO	SET
CARRY RESET	1	1	YES	RESET
CARRY SET	1	1	NO	RESET
CARRY SET	1	1	YES	SET

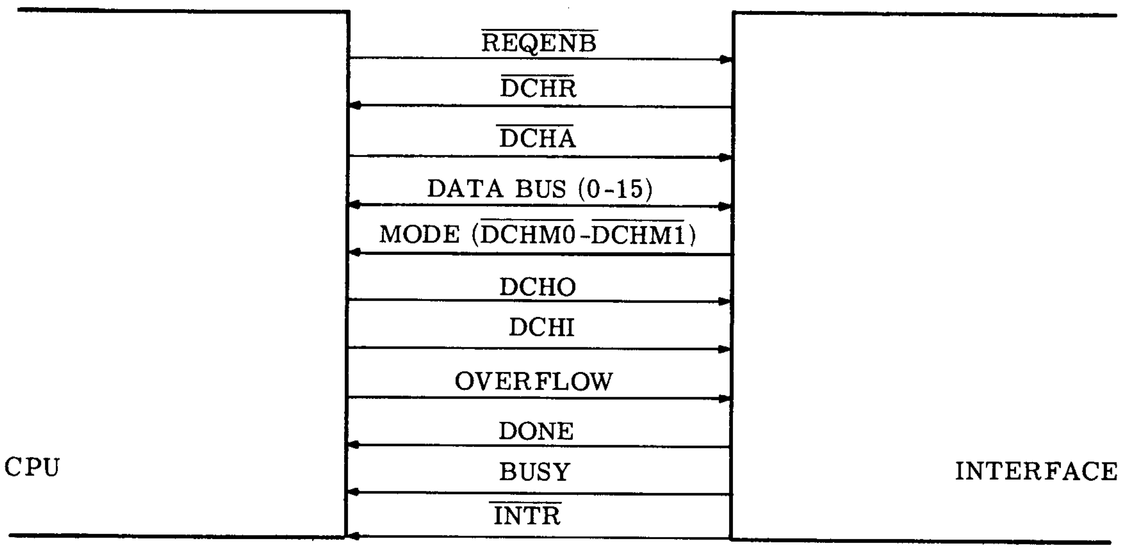
DG-00050

Table C-4

Memory Reference Instruction Decoding Chart

IR	0	1	2	3	4		
NO AC	0	0	0	0	0	JMP	SINGLE CYCLE(FETCH) EXCEPT DEFER(BIT5=1)
NO AC	0	0	0	0	1	JSR	SINGLE CYCLE(FETCH) EXCEPT DEFER(BIT5=1)
NO AC	0	0	0	1	0	ISZ	TWO CYCLE(FETCH & EXEC) EXCEPT DEFER(BIT5=1)
NO AC	0	0	0	1	1	DSZ	TWO CYCLE(FETCH & EXEC) EXCEPT DEFER(BIT5=1)
AC	0	0	1	ACD		LDA	TWO CYCLE(FETCH & EXEC) EXCEPT DEFER(BIT5=1)
AC	0	1	0	ACD		STA	TWO CYCLE(FETCH & EXEC) EXCEPT DEFER(BIT5=1)

DATA CHANNEL SIGNALS



SEQUENCE:

1. $\overline{REQENB}$ TO I/O
2. $\overline{DCHR}$ TO CPU
3. $\overline{DCHA}$ TO I/O
4. a. MAIN MEMORY ADDRESS ON DATA BUS TO CPU
   b. MODE BITS TO CPU (SEE TABLE)
5. DATA ON DATA BUS DIRECTION DETERMINED BY TYPE OF OPERATION.
6. DCHO OR DCHI TO INTERFACE

A. OVERFLOW LINE APPLIES ON TO INCREMENT MODE
B. DONE, BUSY AND $\overline{INTR}$ SAME AS NORMAL I/O

MODE BIT TABLE

$\overline{DCHM0}$	$\overline{DCHM1}$	FUNCTION
H	H	OUT (WRITE)
H	L	INCREMENT
L	H	IN (READ)
L	L	NOT USED

DG-00031

Figure C-6 Data Channel Signals

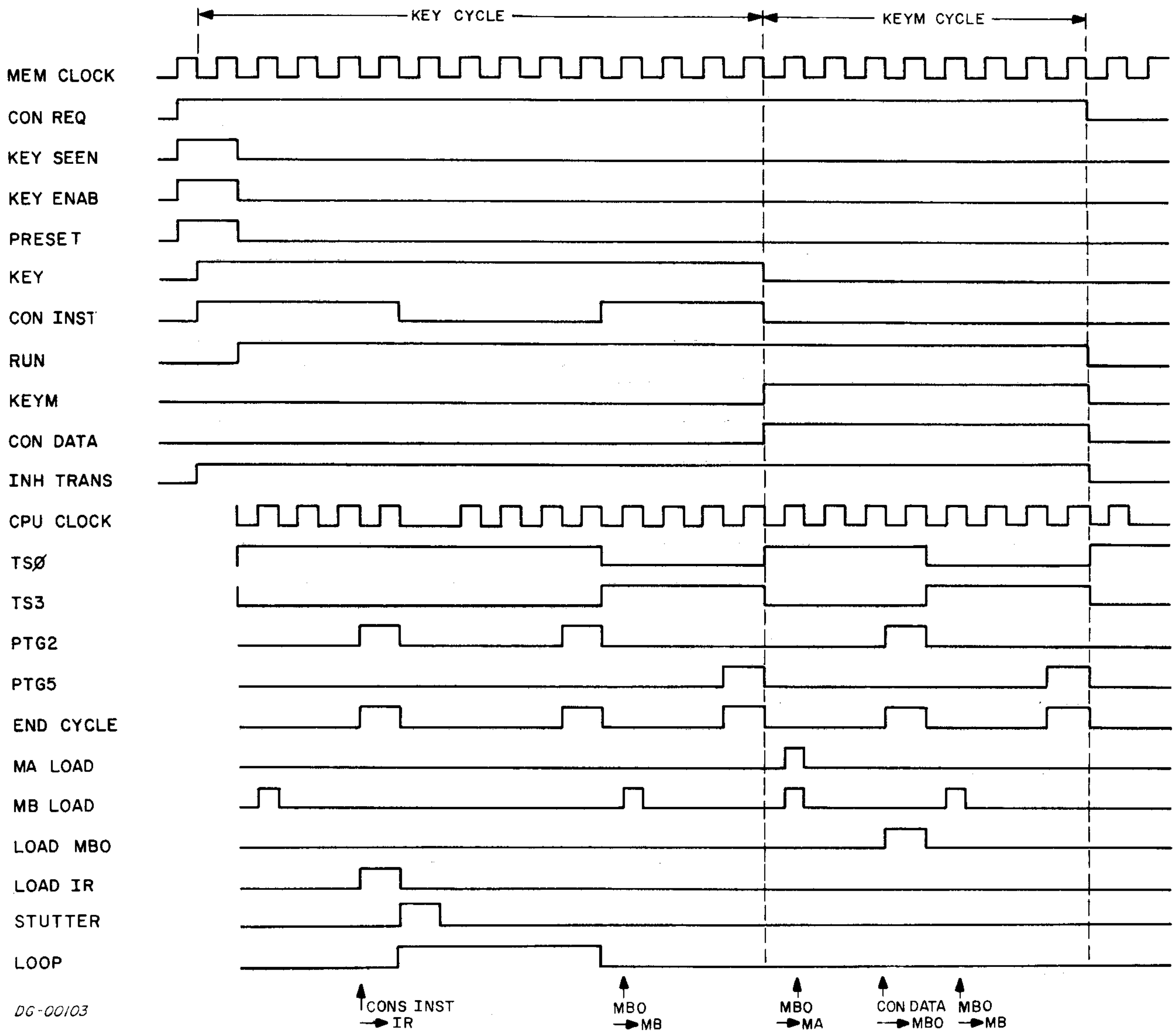


Figure C-7 Deposit Timing Diagram

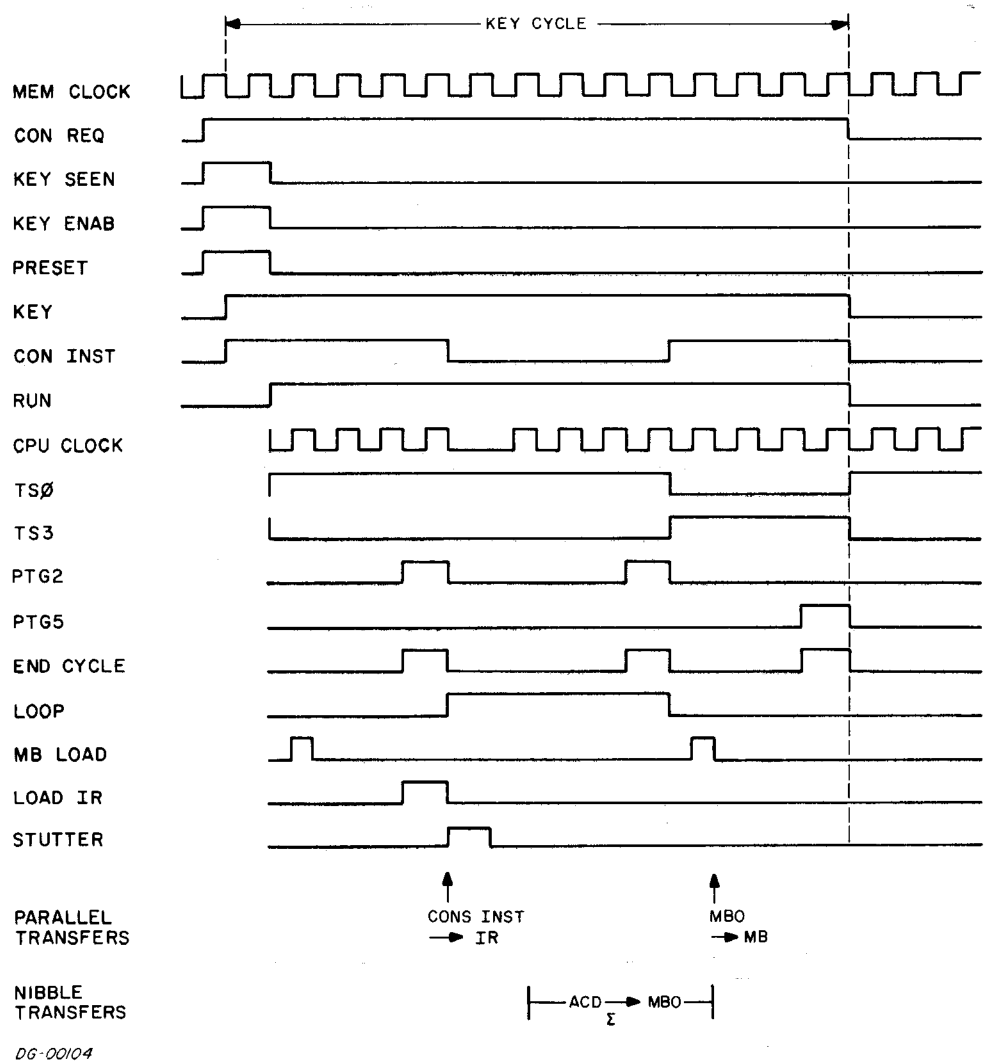


Figure C-8 Examine AC1 Timing Diagram

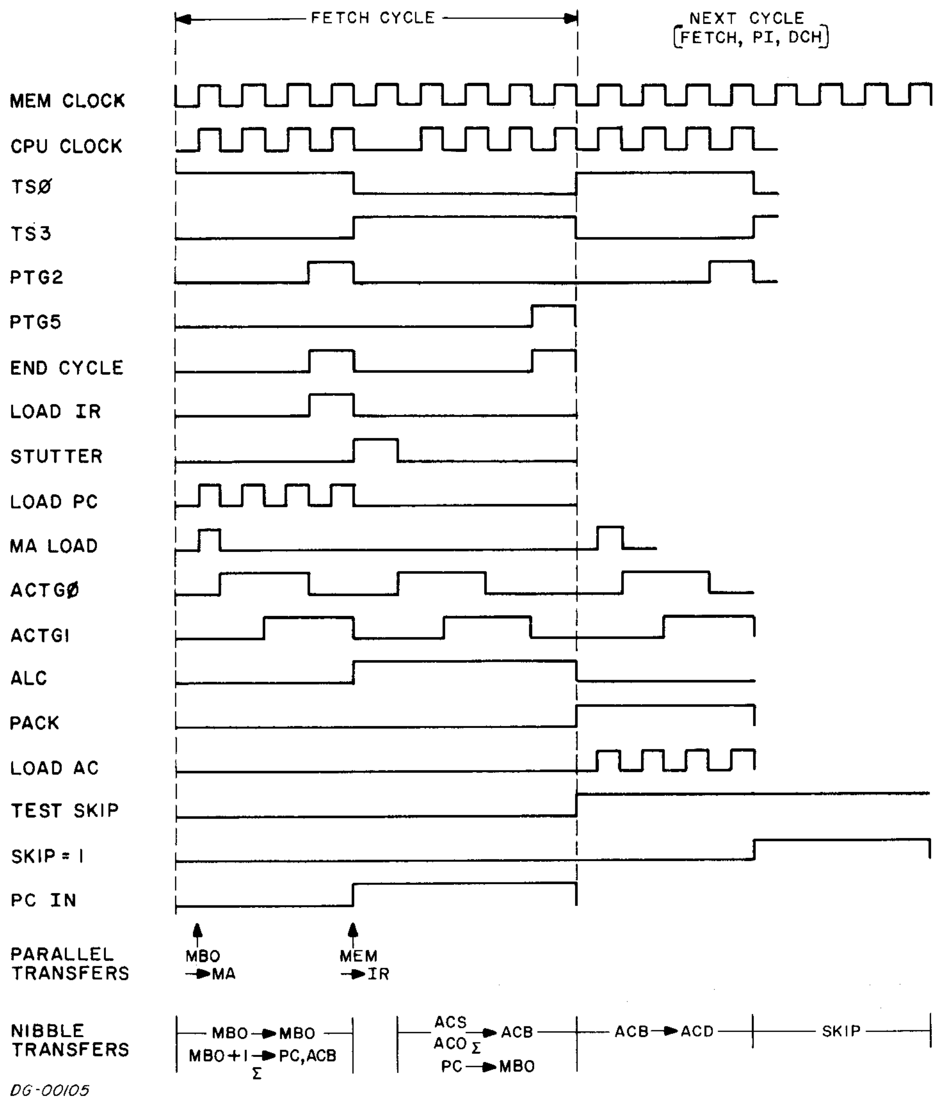


Figure C-9 ADD0, 1, SKP Timing Diagram

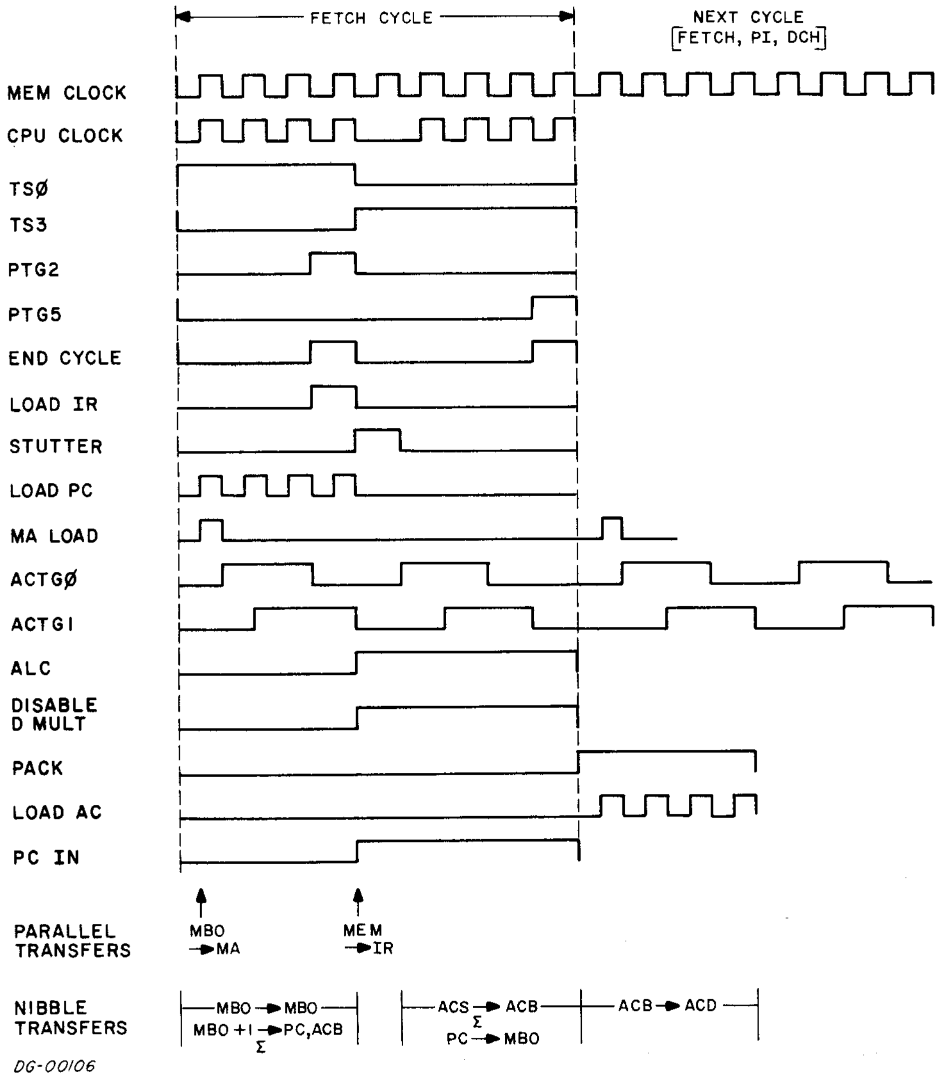


Figure C-10 MOV 0, 0 Timing Diagram

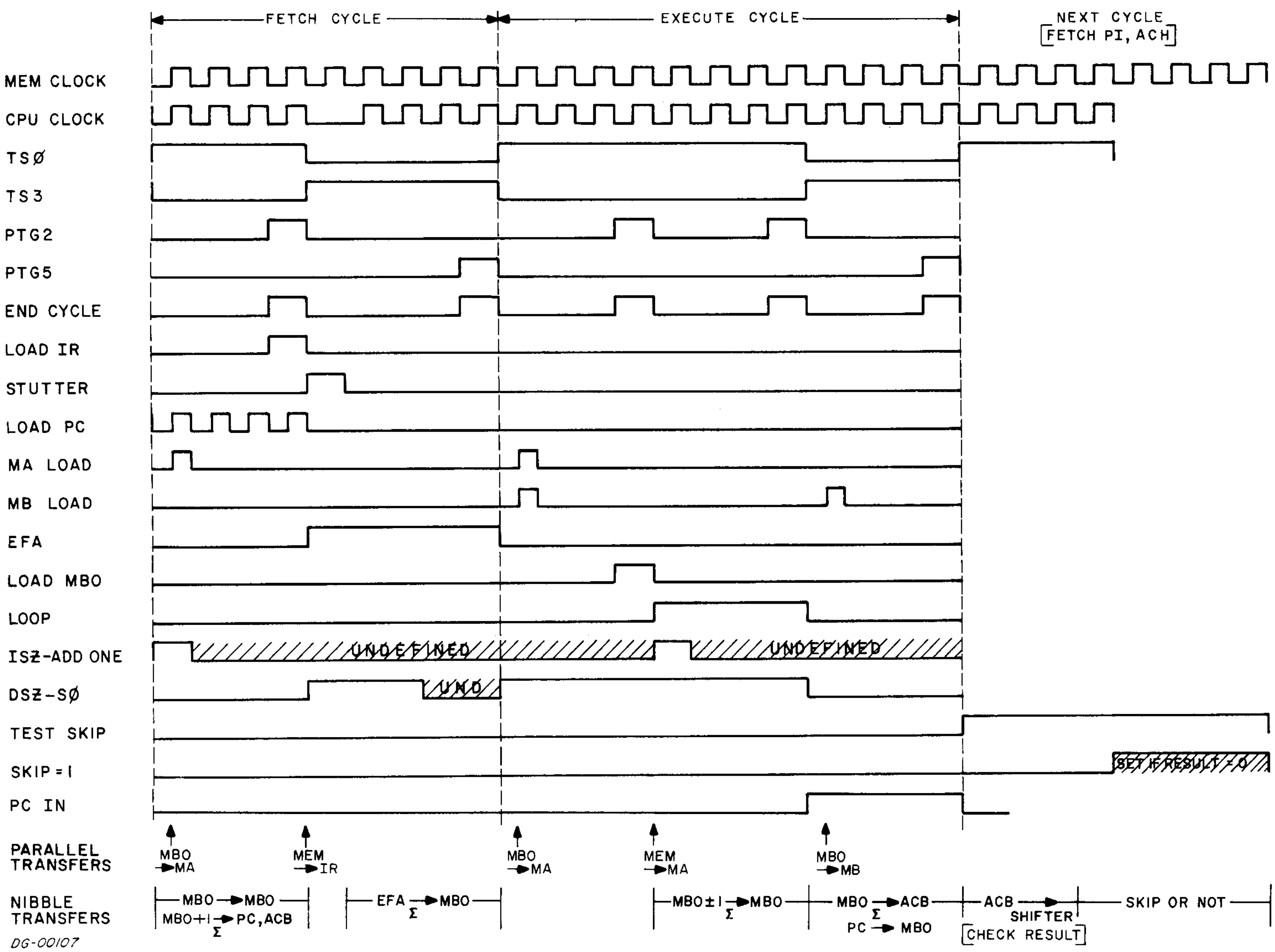


Figure C-11 Timing Diagram For Both The ISZ And DSZ Instructions

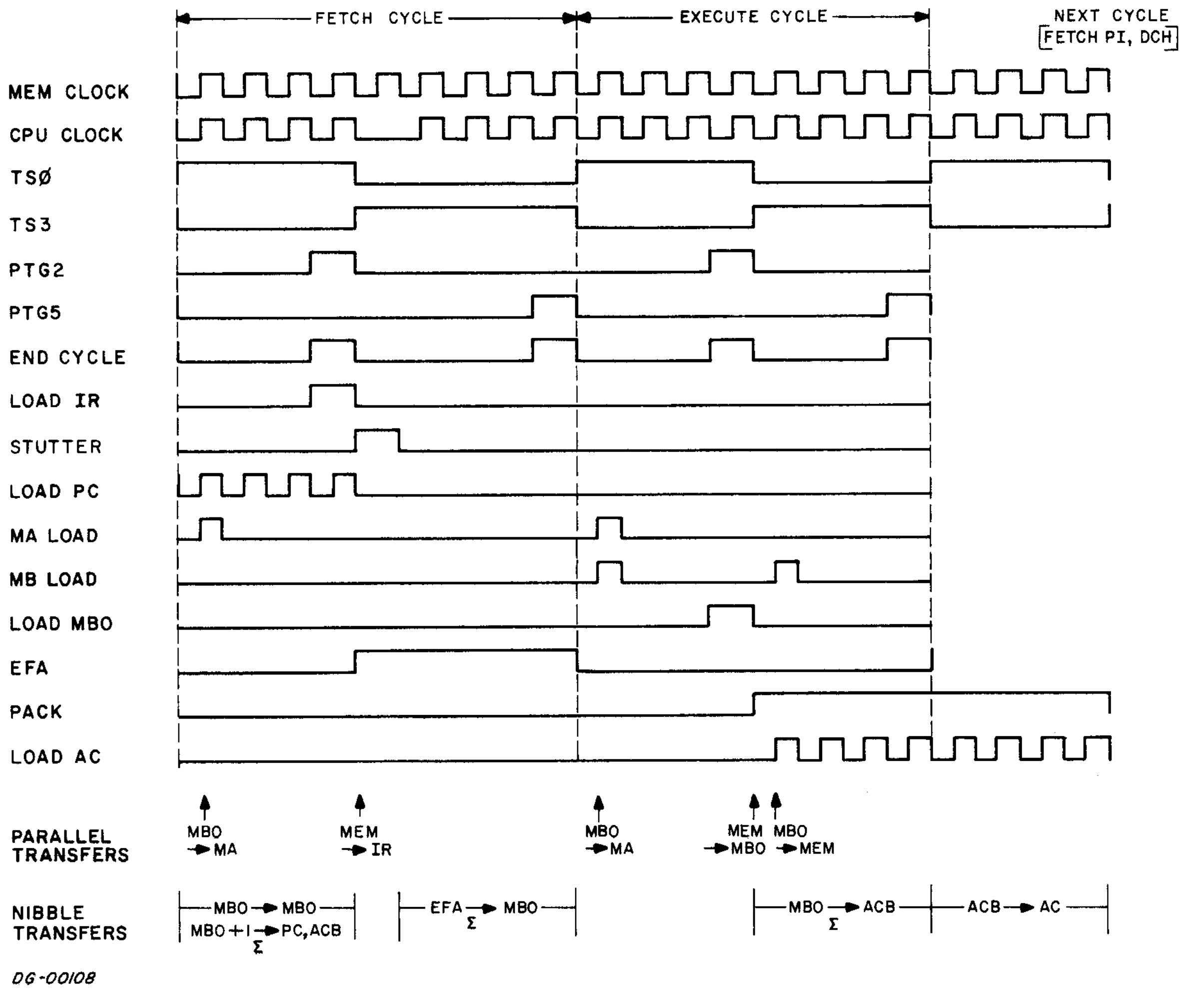


Figure C-12 LDA Timing Diagram

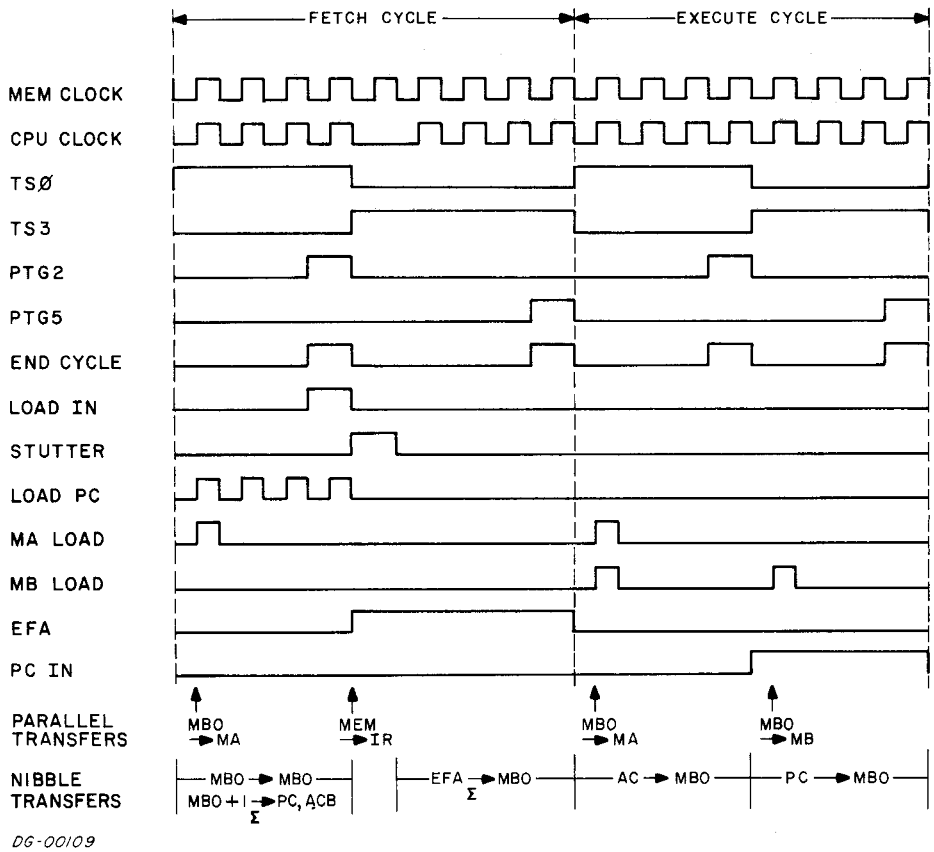


Figure C-13 STA Timing Diagram

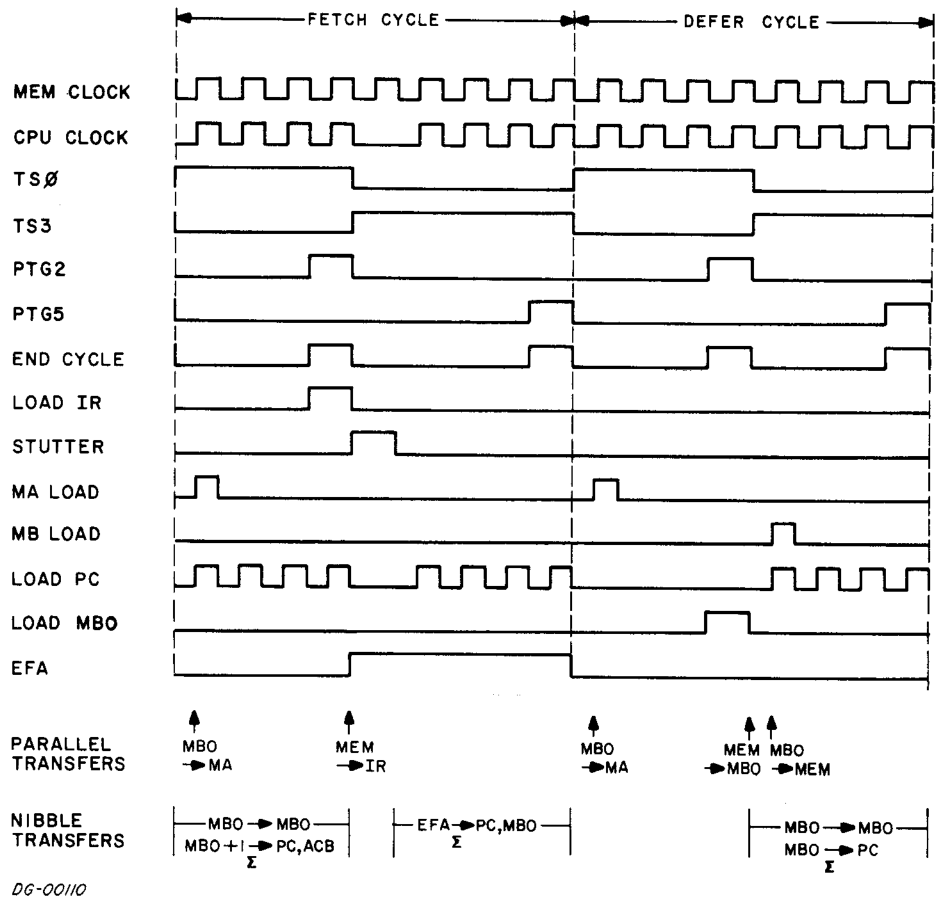


Figure C-14 JMP @ 100 Timing Diagram

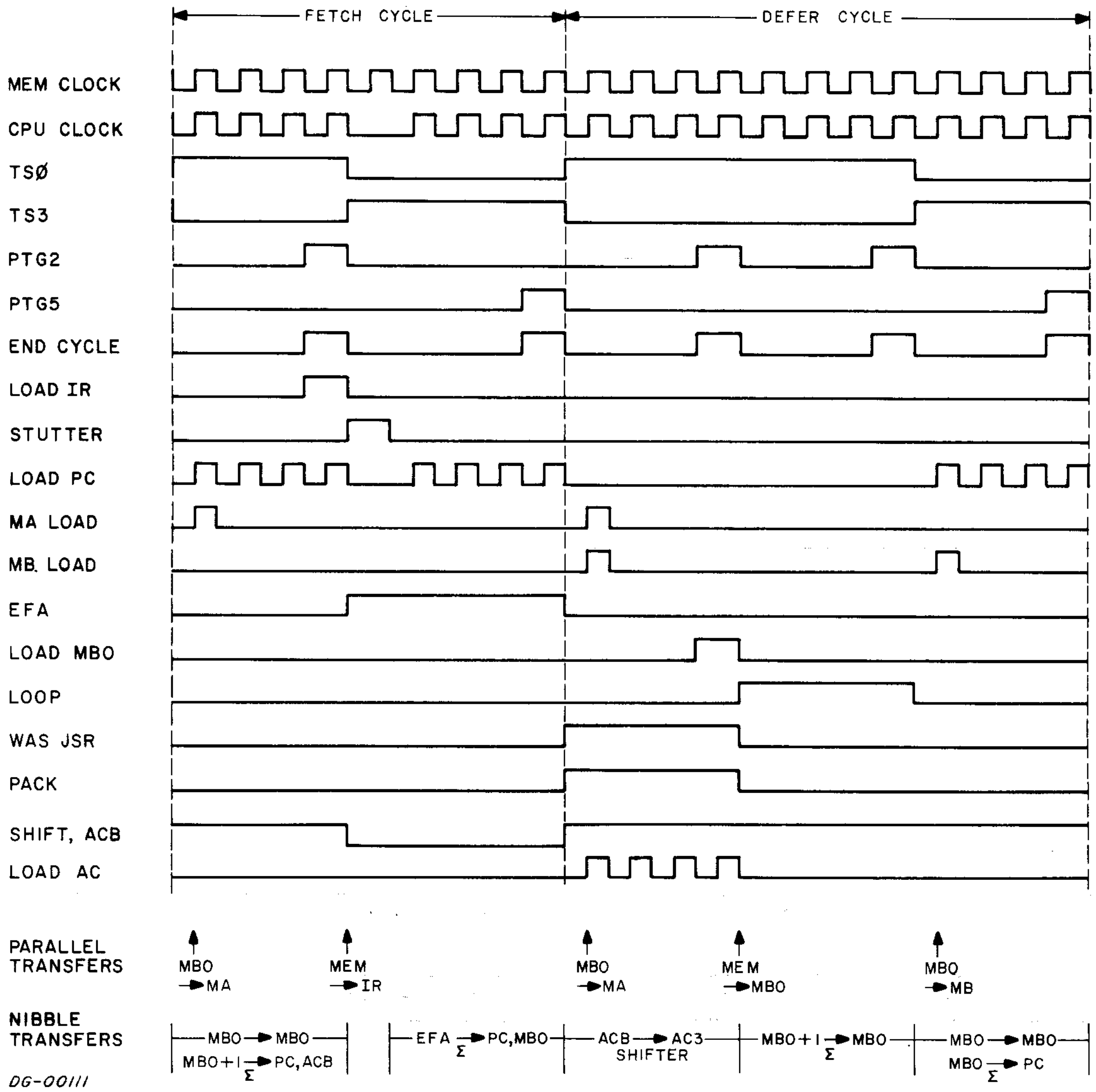


Figure C-15 JSR @ 20 Timing Diagram

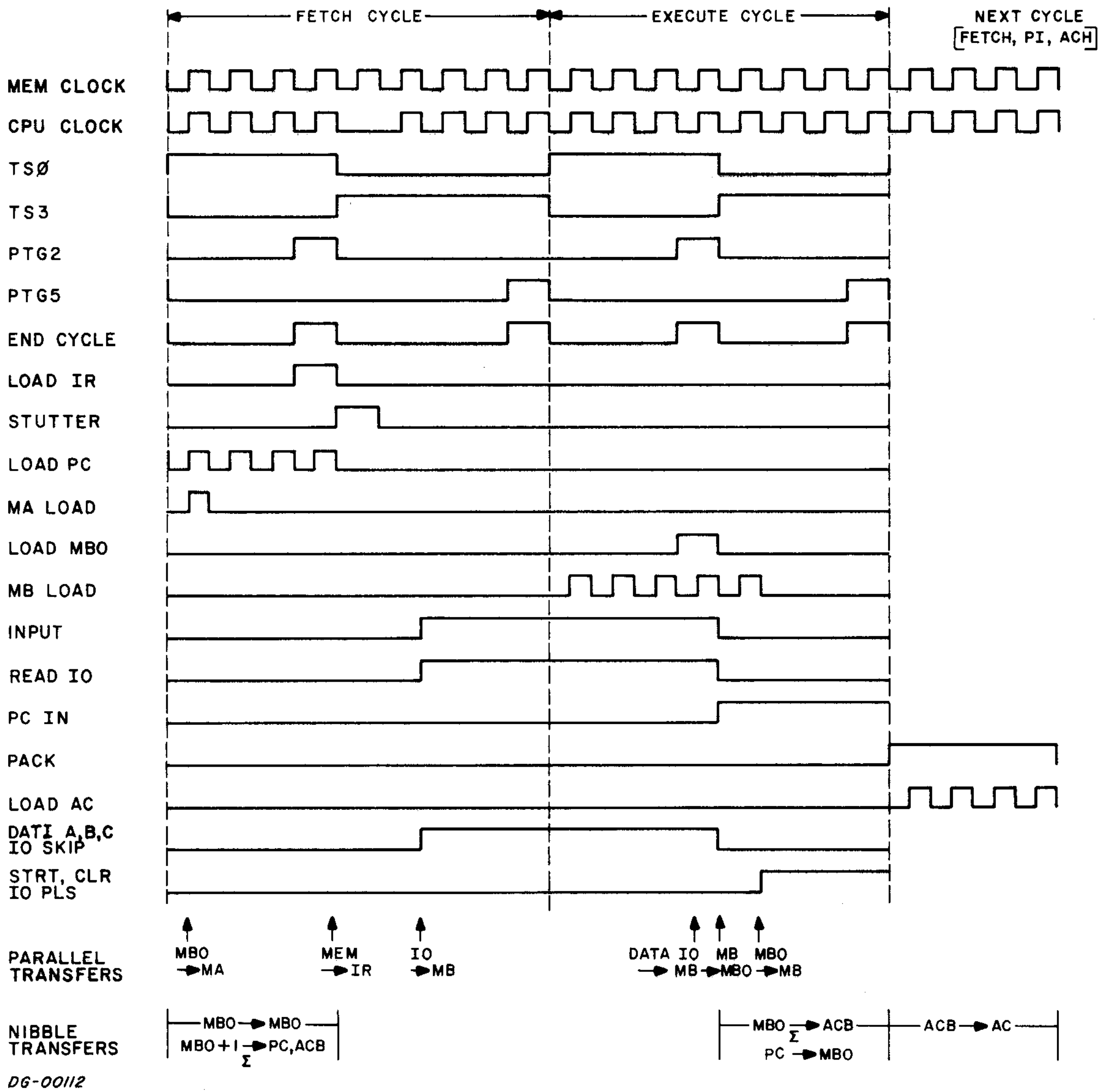


Figure C-16 I/O Input Timing Diagram

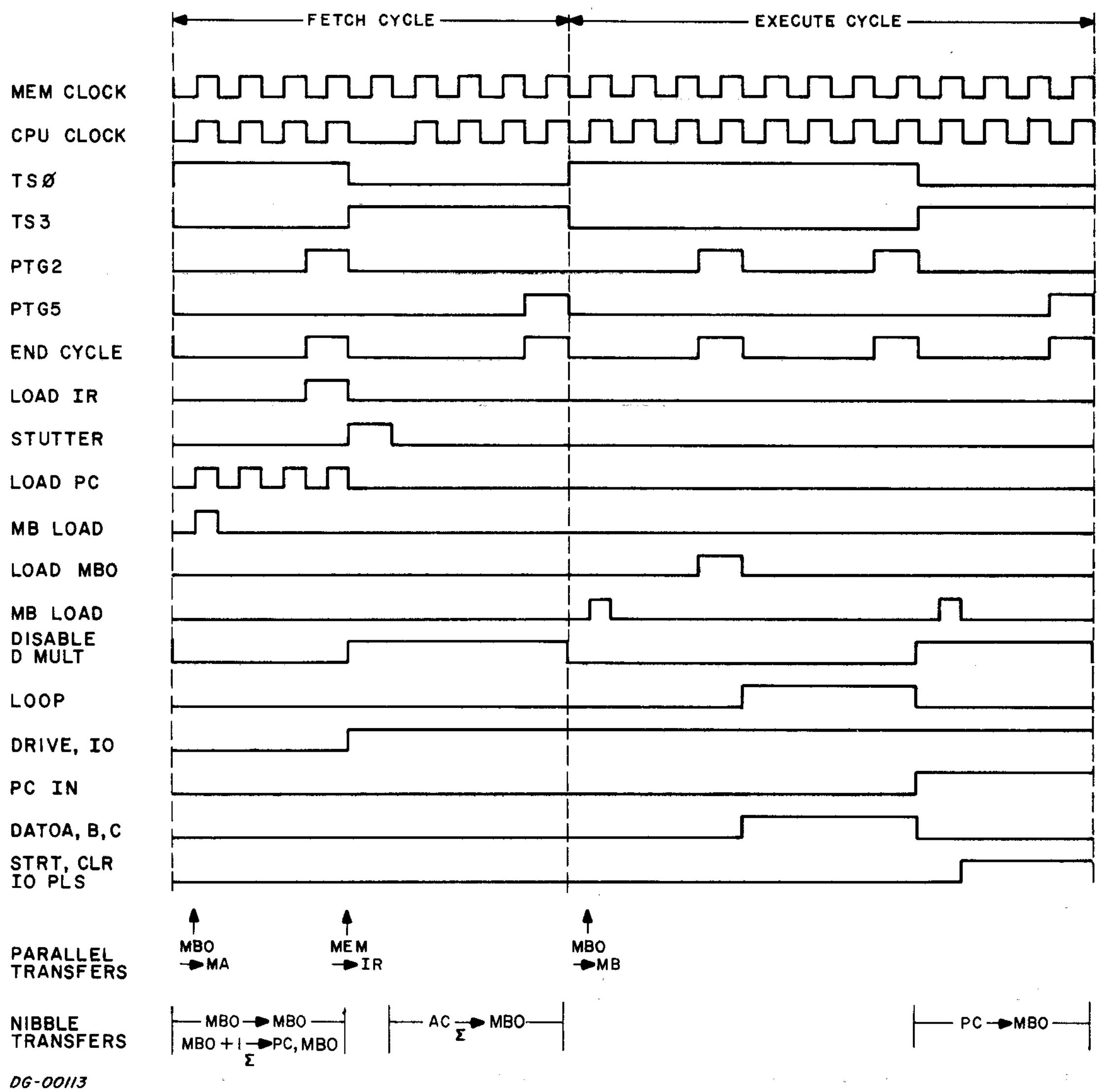


Figure C-17 I/O Output Timing Diagram

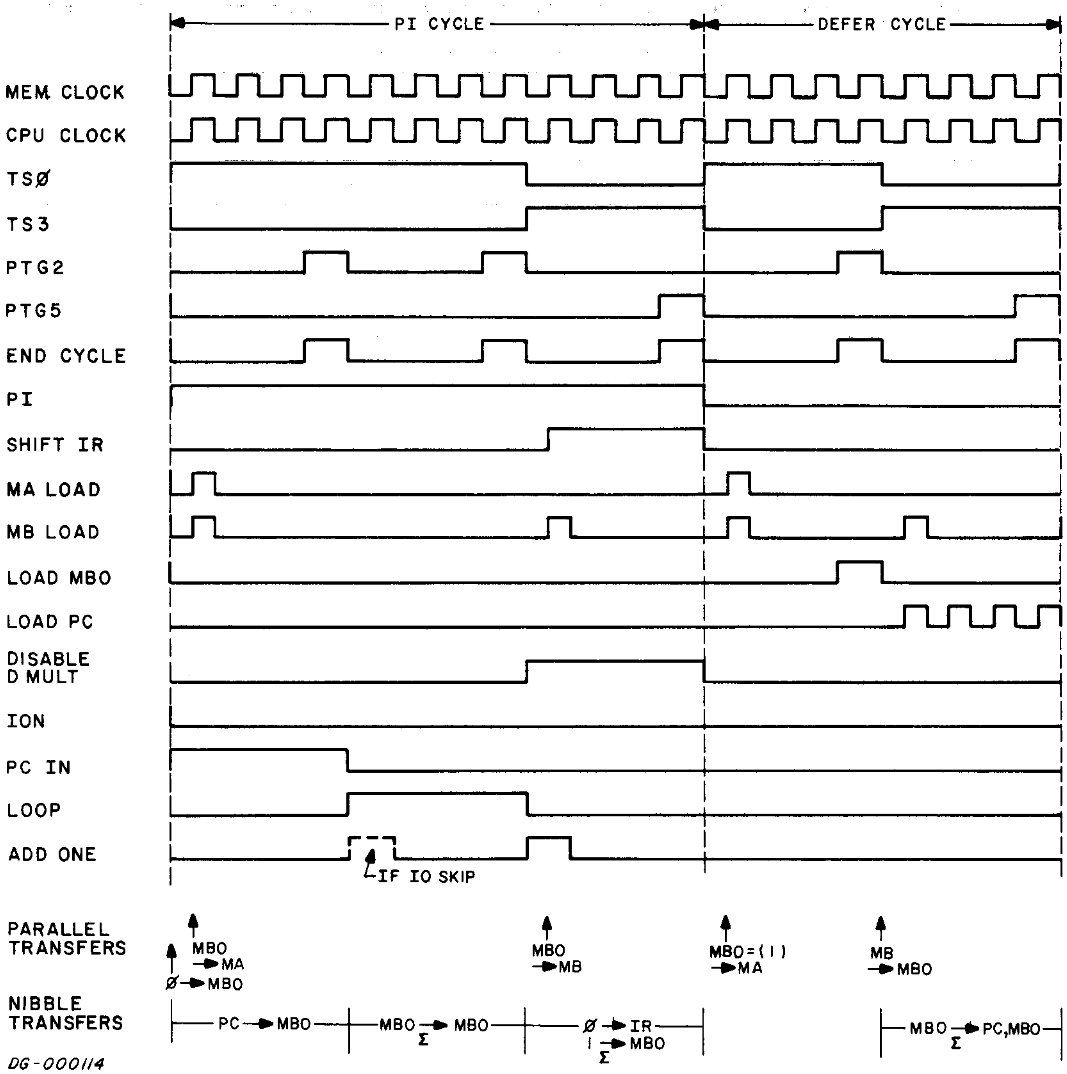


Figure C-18 PI Timing Diagram

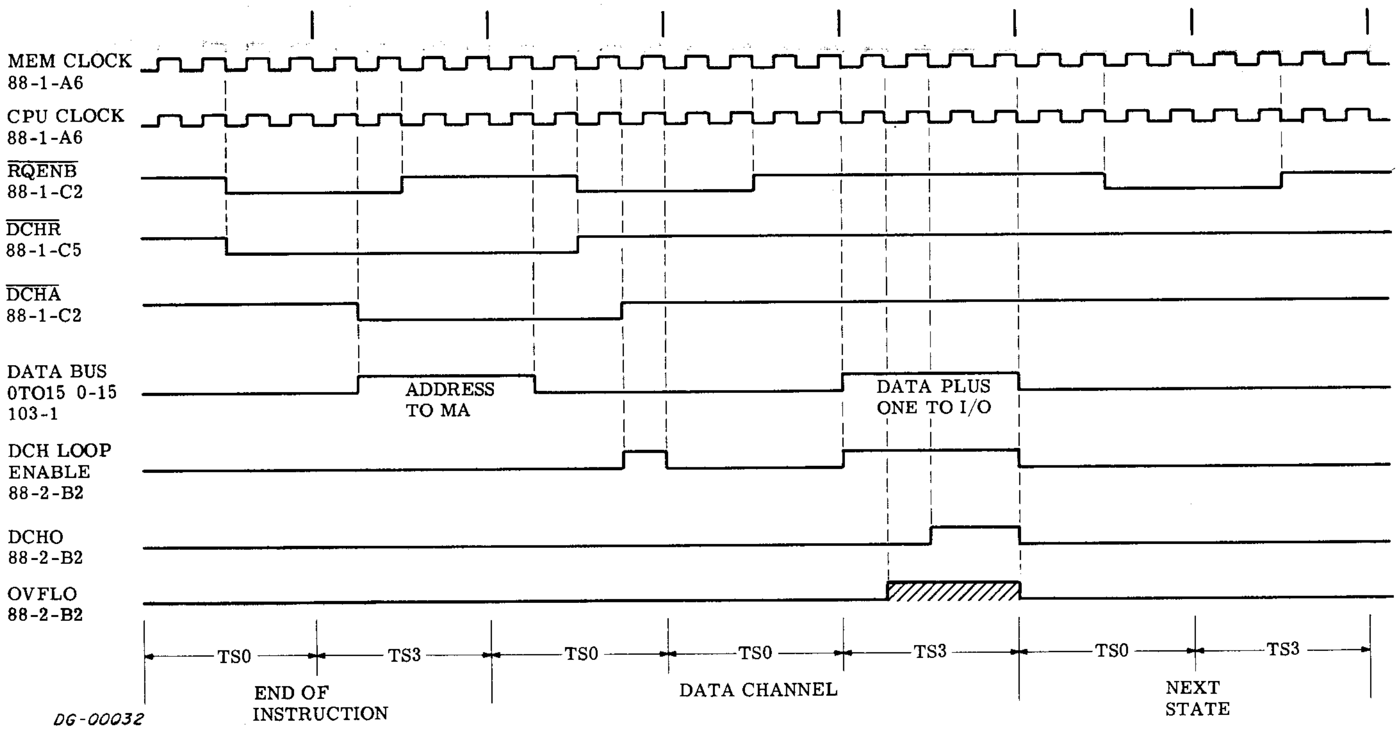


Figure C-19 Data Channel Increment Timing

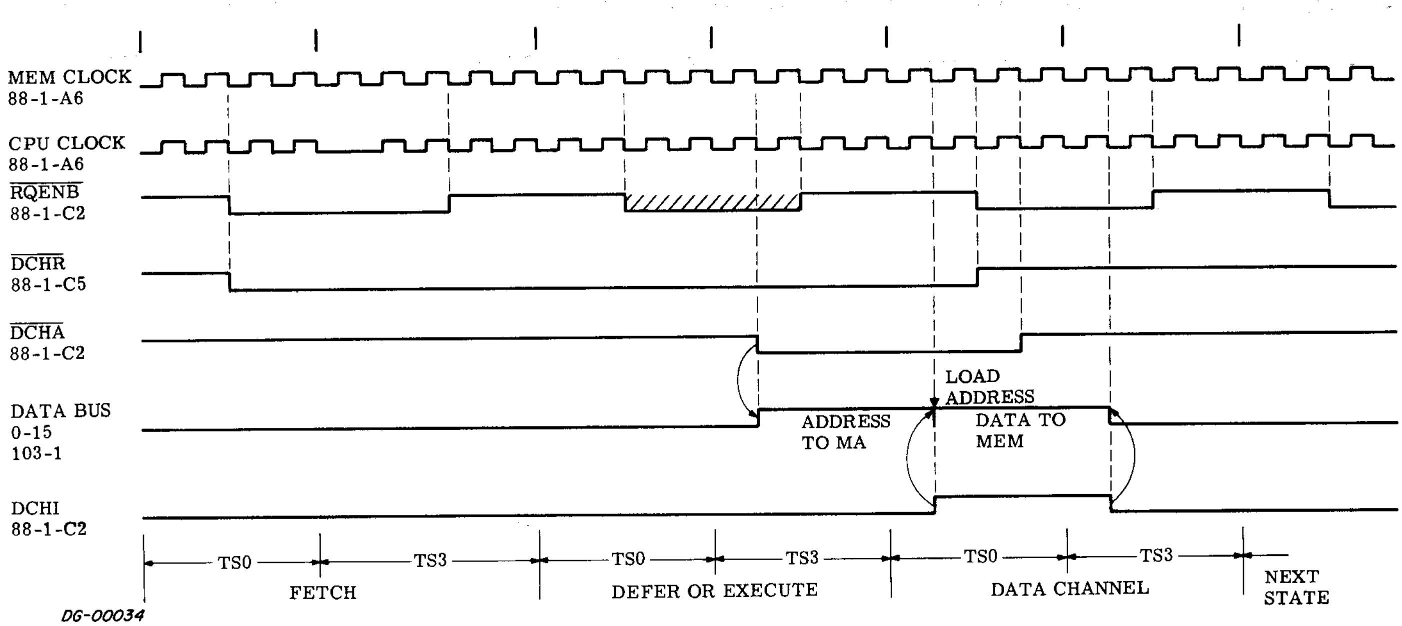


Figure C-20 Data Channel In Timing

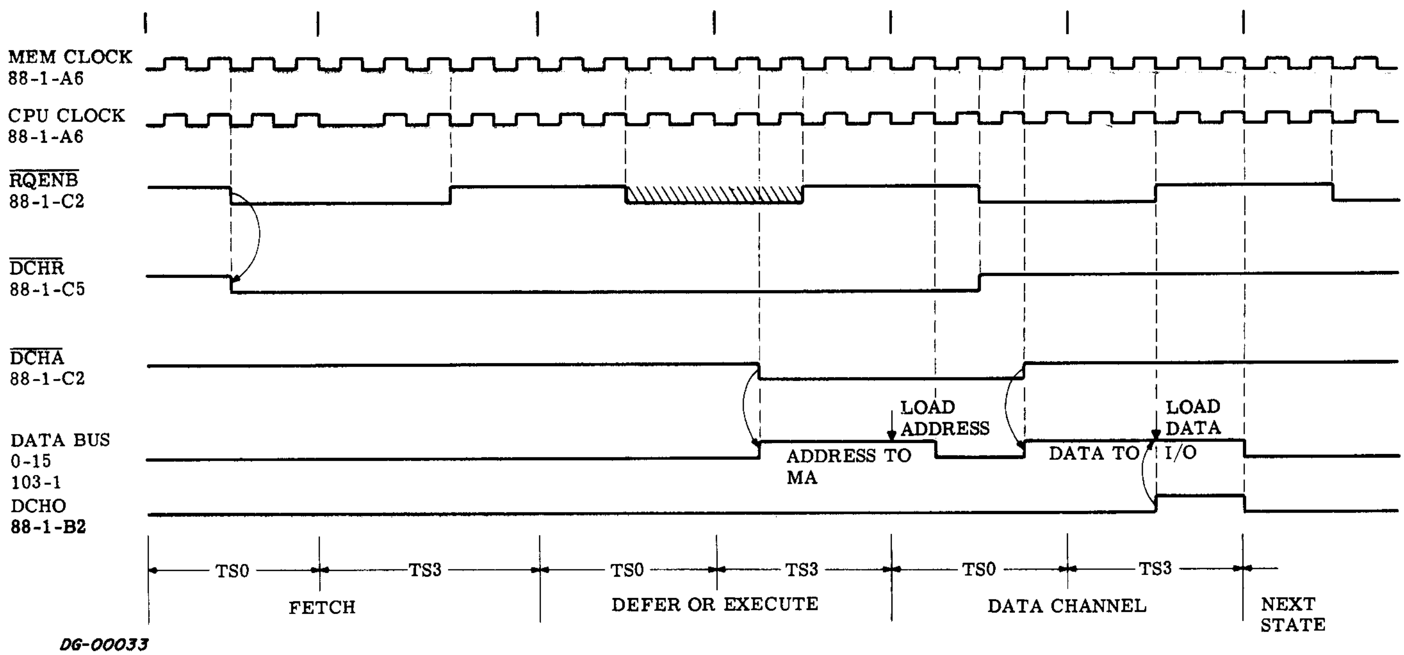


Figure C-21 Data Channel Out Timing

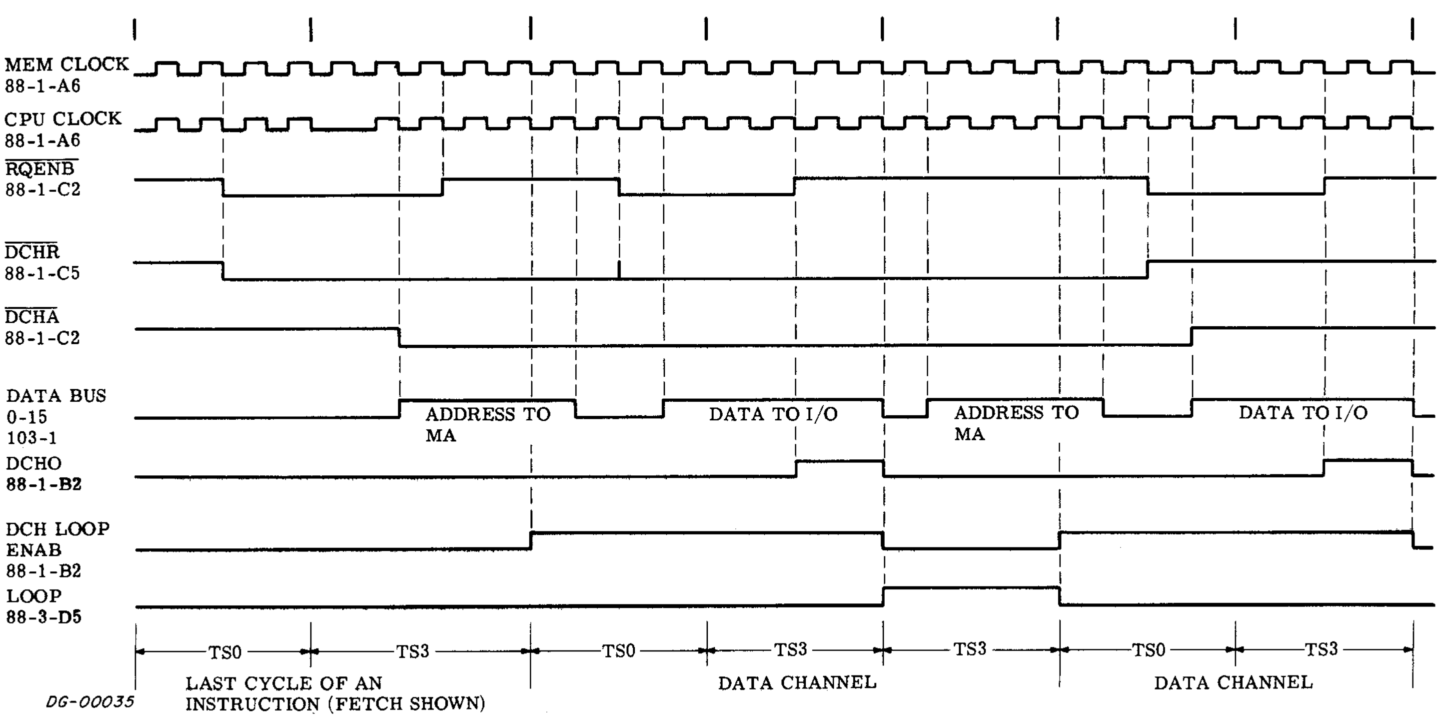


Figure C-22 Data Channel Out Followed By Data Channel In Timing

SECTION K

THE OPERATOR'S CONSOLE

INTRODUCTION

The console illustrated in Figure K-1, has a set of ADDRESS lights which display the contents of the MBO bus; a set of DATA lights which display the contents of the MEM bus; a register of toggle switches which will output to the MEM bus; a row of control switches at the bottom of the panel which instruct the computer on what to display in the lights, what to do with the information in the toggle switches, where to start or stop and how. The console also has a three position keyed rotary switch which turns power on and off and locks some of the operating switches.

CONSOLE LIGHTS AND SWITCHES

All the lights in the console are continually drawing about 10ma each through series resistors, so their filaments are always hot (but not glowing) and large surge currents are avoided when the filaments are driven on.

The Console ADDRESS Lights

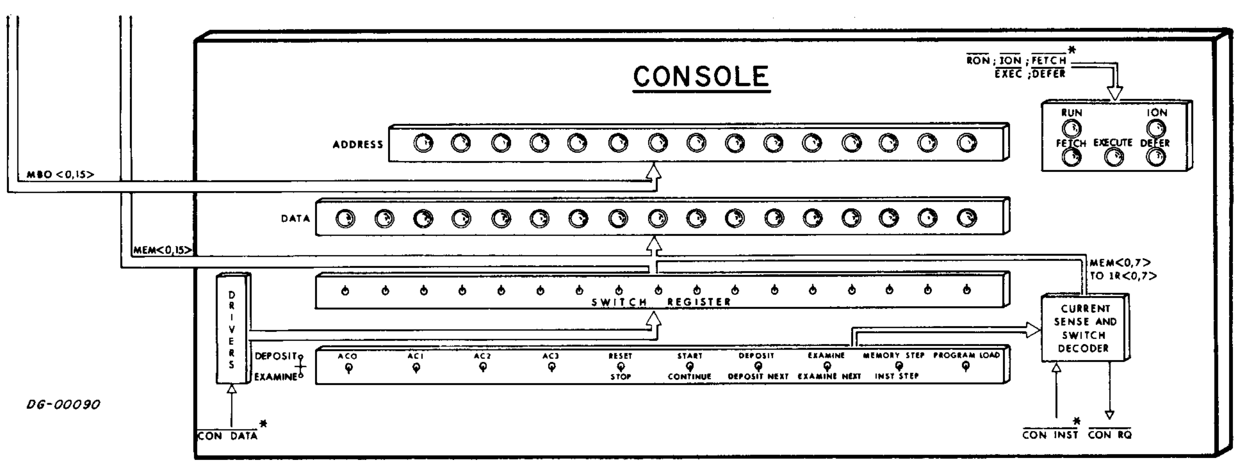
These lights are always showing the state of the MBO bus which is driven directly from the MBO register. When the machine is running, the MBO register is continually shifting, so the display is meaningless; when the machine is stopped, the MBO register shows the contents of the PC, i.e., the next address.

The Console DATA Lights

These lights are always showing the state of the MEM bus. When the machine is running this bus carries data from memory to the instruction and MBO registers; when the machine is stopped this bus contains the contents of the memory buffer of the last memory selected.

The Console Operational Indicators

These lights are driven directly from their corresponding flip-flops in the central processor.



* Issued by CPU

Figure K-1 The Console

The Console Switch Register

These switches connect non-inverting open collector buffers directly to the MEM bus. All Drivers go low when the $\overline{\text{CON DATA}}$ level goes low; $\overline{\text{CON DATA}}$ is issued by the CPU during the READS instruction or during a console operation that requires input from these switches, such as EXAMINE.

The Console Control Switches

All the control switches except STOP and RESET are wired through pull-up resistors to a common circuit which detects when current is flowing through a switch, initiates a delay to suppress contact bounce and then issues the signal $\overline{\text{CON REQ}}$ to the CPU. This signal forces the CPU into the key sequence shown in Figure K-2 which returns the signal $\overline{\text{CON INST}}$ to the console. $\overline{\text{CON INST}}$ connects switches AC0, AC1, AC2, AC3, DEPOSIT, DEPOSIT NEXT, EXAMINE and EXAMINE NEXT through a decoder to the MEM <0, 7> lines, which are input to the Instruction Register and interpreted as shown in Table K-1. The computer then goes into either the KEY or KEYM major state and follows the flows of Figure K-3.

The switches RESET, STOP, MEMORY STEP, INSTRUCTION STEP and PROGRAM LOAD are wired separately to the CPU. RESET stops the computer at the end of the current cycle, issues the IORST pulse to all I/O devices, clears ION and sets the real time clock to the line frequency. STOP simply stops the computer at the end of the current instruction.

MEMORY STEP takes the processor through the current state and then stops. INST STEP takes the processor through the current state and on to the end of the current instruction. Both signals force a $\overline{\text{CON}}$ $\overline{\text{RQ}}$ to the CPU and output $\overline{\text{MSTP}}$ and $\overline{\text{ISTP}}$ respectively. PROGRAM LOAD deposits the contents of the bootstrap ROM into locations 0-37 and the machine at location 0. It outputs the signal $\overline{\text{PL}}$ to the CPU.

The Console Rotary Switch

This switch controls the primary power to the power supply. It has three positions:

OFF - the primary power is removed from the power supply

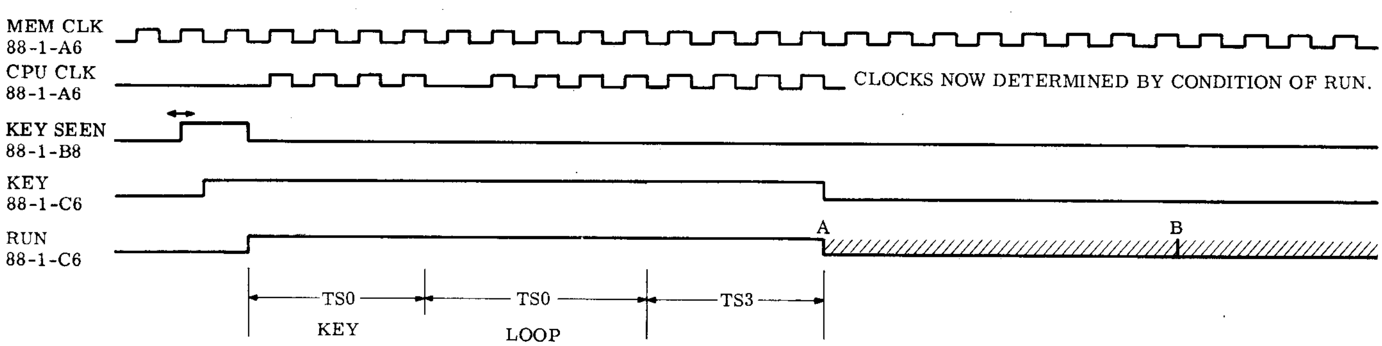
ON - the primary power is applied to the power supply

LOCK - the primary power is applied to the power supply but the STOP RESET switch is disabled

REFERENCES

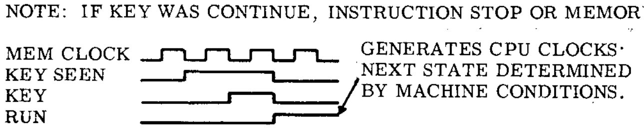
1. "How To Use The Nova Computers" 015-000009-00.

2. Nova 800/1200 Console Print D-001-000089-05.



A. RUN RESETS IF KEY WAS AC EXAMINE OR AC DEPOSIT.
NEXT STATE IF RUN DOES NOT RESET;
KEYM-IF KEY WAS DEPOSIT, DEPOSIT NEXT, EXAMINE, EXAMINE NEXT OR PROGRAM LOAD.
FETCH-IF KEY WAS START
B. RUN RESETS IF KEY WAS DEPOSIT, DEPOSIT NEXT, EXAMINE OR EXAMINE NEXT.

NOTE: IF KEY WAS CONTINUE, INSTRUCTION STOP OR MEMORY STOP



DG-00037

Figure K-2 The CPU Key Sequence Timing Diagram

Rev. 02

Table K-1

Control Switch Decoding To The Instruction Register

CONSOLE INSTRUCTION		IR0	IR1	IR2	IR3	IR4	IR5	IR6	IR7	IR8 TO 15
AC DEP.	AC0	0	0	1	0	0	0	1	1	0
	AC1	0	0	1	0	1	0	1	1	0
	AC2	0	0	1	1	0	0	1	1	0
	AC3	0	0	1	1	1	0	1	1	0
AC EXAM.	AC0	0	1	1	0	0	1	1	1	0
	AC1	0	1	1	0	1	1	1	1	0
	AC2	0	1	1	1	0	1	1	1	0
	AC3	0	1	1	1	1	1	1	1	0
DEPOSIT		1	1	0	1	1	1	0	1	0
DEPOSIT NEXT		1	1	0	1	1	1	0	0	0
EXAMINE		1	1	1	1	1	0	0	1	0
EXAMINE NEXT		1	1	1	1	1	1	0	0	0
MEMORY STEP		1	1	1	1	1	1	1	1	0
INSTRUCTION STEP		1	1	1	1	1	1	1	1	0
PROGRAM LOAD		1	1	1	1	1	1	0	1	0
START		1	1	1	1	1	0	1	1	0
WHEN BIT GOES FALSE		ACD+ACEX	ACD	DEP.+DEP. NEXT	AC SELECT ON ACD+ACEX		DATA TAKEN FROM SWITCHES	NEXT STATE KEYM	DEPOSIT NEXT + EXAMINE NEXT	ALWAYS FALSE ON KEY

DG-00036

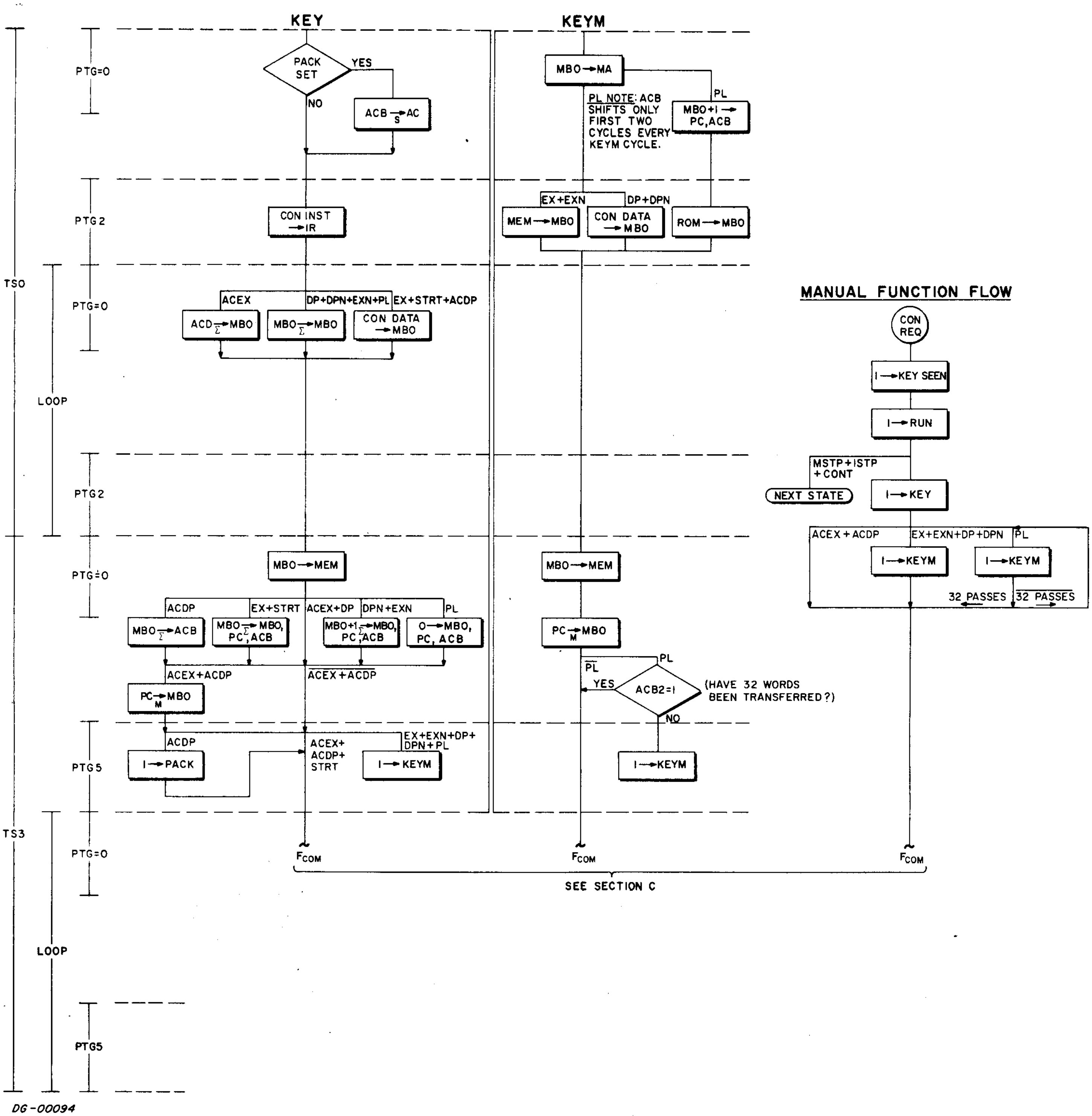


Figure K-3 Key, KEYM and Manual Flow Diagrams

Table K-2
Backpanel Connections To The Console
Through POA

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

This Page Left Blank

Intentionally

SECTION P

POWER SUPPLY

INTRODUCTION

The Nova 1220 power supply is mounted on the backpanel below the circuit boards where it converts either 110Vac at 60Hz or 220Vac at 50Hz to regulated, current limited 5Vdc, -5Vdc, +15Vdc for the logic and memories, and to unregulated 6.3Vac for the real time clock. With the power monitor and restart option, the power supply interrupts the computer when it detects a line voltage failure (less than 90% of nominal) stops the computer when the voltage gets too low for reliable operation, and issues a start pulse to the computer when the line voltage recovers.

POWER SUPPLY CIRCUITS

The 30V Unregulated Supply

110Vac or 220Vac are input through the power cord to a switch on the console S1, then on to transformer T1. The two primaries of T1 are wired in parallel for 110Vac, and in series for 220Vac. Note that the cooling fan operates on 110Vac only.

The secondary of the transformer is wired to two full wave bridge rectifiers which output approximately 30V and -15V into RC filters. The 30V is applied to two series pass switching regulators which supply the regulated +5Vdc and +15Vdc. The 15V is applied to a simple linear regulator for the -5Vdc.

The Series Pass Switching Regulators

A series pass switching regulator acts like a multivibrator which sets when it detects a low output voltage and resets when it detects a high output voltage. When the regulator is set, it gates current from the 30V supply into an LC circuit and the load; when the regulator is reset, the load draws all of its power from the LC circuit until the circuit is sufficiently exhausted to be recharged by the regulator. The frequency at which the regulator sets and resets varies from 0 to 25KHz depending on the load.

There are two such regulators in the 1220 power supply, one for the +15Vdc (Figure P-1) and the other for the +5Vdc (Figure P-2). The -5Vdc is controlled by a linear regulator.

Note that the outputs of these circuits are DC levels with about .15V ripple at frequencies which vary with the loads.

The Fuses

The 1220 power supply has two fuses, a 10 amp between the power cord and the switch S1, and a 15 amp just after the bridge rectifier. The 10 amp will blow if there is a short in the cabling to S1, or if the convenience receptacle is overdrawing; the 15 amp will blow if the +15Vdc or +5Vds levels rise high enough to trigger an SCR, which then creates a short between the 30V supply and ground.

The Power Fail Module

This module detects a line voltage failure and outputs the signals shown in Table 2.

REFERENCES:

1. Fairchild Semiconductor Integrated Circuit Data Catalog - Fairchild Semiconductor 1970

2. Backpanel Nova 1220 print No. D-001-000208-00

3. Backpanel 1220 Power Supply print No. D-001-000173-02.

Table P-1

Nova 1220 Power Supply Specifications

Output Voltage Level Name	Output Voltage	Maximum Current	Used On	Remarks
+15V	14.5-15.1Vdc (.15V ripple)	9A	XY Drivers	Full wave rectified; Short Circuit & Over-voltage Protection Regulated
5V	-5→-7Vdc	1A	Sense Amplifiers	Full wave rectified; Current limited by a resistor, regulated
+5V	5.2→5.4Vdc	20A	IC Logic	Full wave rectified; Short Circuit & Over-voltage Protection Regulated
TTY	-5→-7Vdc (.15V ripple)		Teletypewriter	Full wave rectified; Current limited by a resistor, regulated
RINH<0,15>	14.5-15.1Vdc	760mA each	Inhibit Driver	Full wave rectified; Short Circuit & Over-voltage Protection, Regulated
60Hz	6.3Vac	500mAc	Real Time Clock	This signal has the same frequency as the line (input) voltage
A10(VINH)	14.5-15.1Vdc (.15V ripple)	6Adc	Memory Inhibit Logic	Current Limited
B84(VINH)	14.5-15.1Vdc (.15V ripple)		Memory Drivers	Turns off memory drivers at about +12Vdc
$+V_{LAMP}$	≈14-16Vdc	2Adc	Console Lamps	Unfiltered, Unregulated

Table P-2

Output Signals of the Nova 1220 Power Fail Module

SIGNAL NAME	SIGNAL FUNCTION
$\overline{\text{PWR FAIL}}$	-sets the PWR LOW flag in the processor when the line voltage drops to 90% of nominal voltage.
MEM OK	-resets the RUN flag and stops the computer when the + Vmem (+15Vdc) voltage goes too low for the memory to function reliably.
+5OK	-sets the RUN flag and starts the computer when the +5Vdc has risen to 4.4 Vdc.

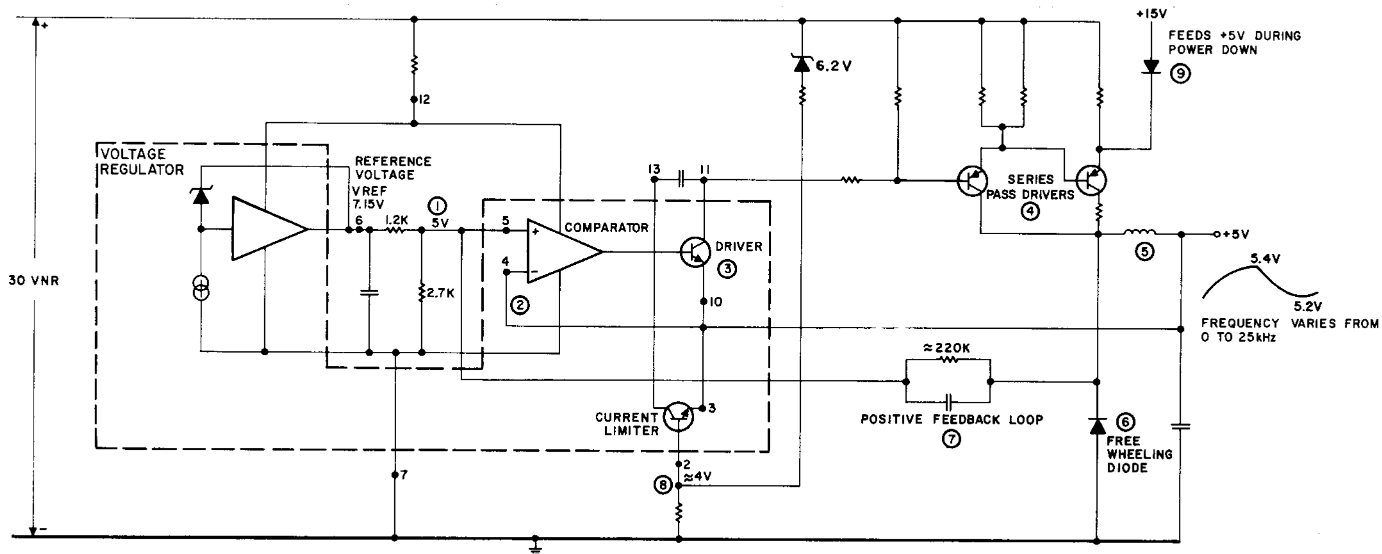


Figure P-1 Simplified Schematic of the +5Vdc Series Switching Regulator. When the comparator senses a difference between the (divided) reference voltage (1) and the output voltage (2) it switches, turning on the driver transistor (3) and consequently the series pass transistors (4). Current is shunted through the series pass transistors to the coil, output capacitor and load (5). The output voltage rises, reducing the error voltage to the comparator, which resets, turning off the driver (3) and consequently the series pass transistors. Now the load is supplied from power stored in the LC circuit. The back emf developed across the coil as a result of this switching is dropped across the free wheeling diode (6). Note that each time the comparator is forced to switch it is driven into saturation by the positive feedback loop which includes the 220K resistors (7).

The current limiter (8) turns on if the output voltage drops below about 4V, turning the driver (3) and subsequently the series pass transistors (4) off. The supply is latched in this state until power is removed and then returned.

The diode (9) feeds current from the 15V supply to +5V during power-down, driving the memory supply off early and the logic supply off later.

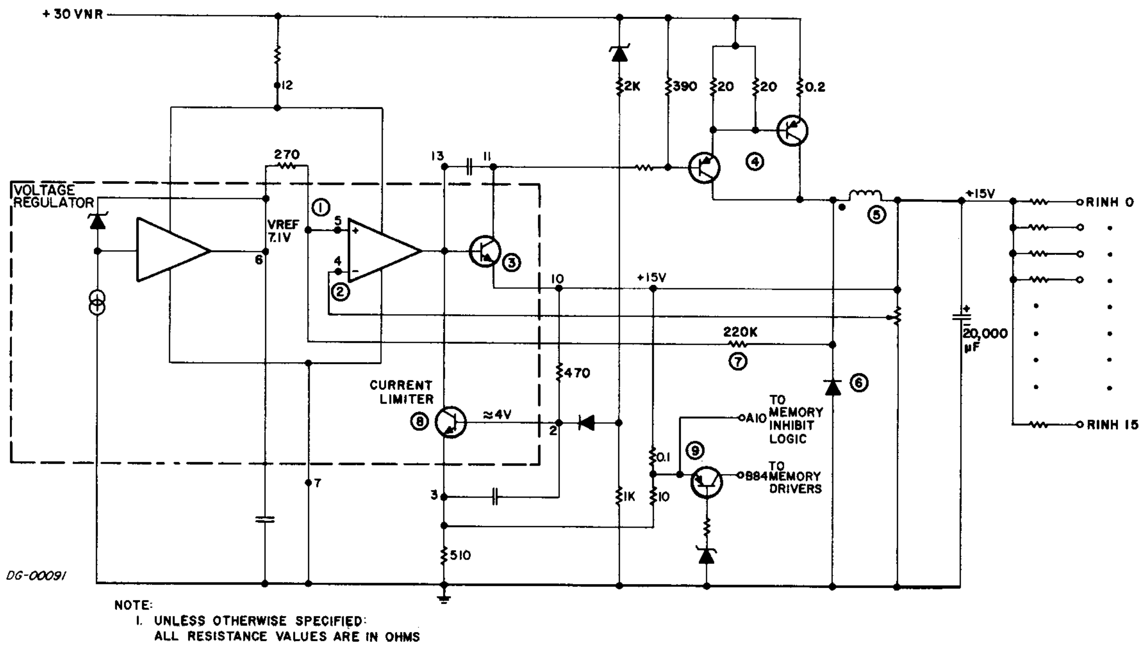


Figure P-2 Simplified Schematic of the +15Vdc Series Switching Regulator. When the comparator senses a difference between the reference voltage (1) and the divided output voltage (2), it switches, turning on the driver transistor (3) and consequently the series pass transistors (4). Current is shunted through the series pass transistors to the coil, output capacitor and load (5). The output voltage rises, reducing the error voltage to the comparator, which resets, turning off the driver (3) and consequently the series pass transistors. Now the load is supplied from power stored in the LC circuit. The back emf developed across the coil as a result of this switching is dropped across the free wheeling diode (6). Note that each time the comparator is forced to switch it is driven into saturation by the positive feedback loop which includes the 220K resistor (7).

The current limiter (8) turns on if the output voltage V MEM drops too low, or if the current at either terminal of (9) (memory inhibit and memory drive) is too high. When on, the current limiter turns off the driver and subsequently the series pass transistors, latching the supply into this mode until power is removed and then returned.

The transistor at (9) will switch off when the +15V drops too low for memory to function properly, thus removing power to the memory drivers.

SECTION M

THE MEMORY

A REVIEW OF CORE MEMORIES

A "bit" of information can be stored in a ferrite core by magnetizing the core in one of two possible directions or "states" and then calling one state a "1" and the other state a "0", similar to a flip-flop. Unlike a flip-flop, however, a core cannot be read simply by examining its output voltages; a core is read by forcing it into the "0" state and then watching for the current pulse which is always generated when a core changes state. If the pulse occurs, then the core must have been in the "1" state before it was excited; if no pulse occurs then the core must already have been in the "0" state because no transition took place.

Reading a core, then, always leaves it in the "0" state and although the information that it contained has probably been transferred to some register which was set by the current pulse, that information is no longer in the core, and it usually has to be restored with what is called a "write cycle". Writing means setting the core to a one or a zero, depending on the state of the memory register that usually contains core bound information.

Reading or writing into a core is a matter of sending current pulses along wires into the core; the direction of current relative to the core determines into which state the core will move.

Data General's core memories contain many thousands of these ferrite cores strung together like beads on wire. Each core has three wires passing through it, and these wires carry the currents to magnetize them and the pulses which occur when they change state. The memories are wired so that the computer can select any group of 16 bits at once, and read or write a complete 16 bit word "in parallel". A group of 16 cores, called an "address" is picked by passing current down two selected wires called X and Y, which are strung into the cores so that they both pass through only one address. The combined effect of current in these two wires is enough to flip the core into the zero state if it is not already there. Each core that flips sends a pulse down its own third wire called the sense wire which is then fed into one flip-flop of a 16 bit Memory Buffer. The flip-flop sets if it sees a pulse, and remains static if it does not. The register which selects the X Y wire or "lines" is called the Address Register.

Restoring the contents of the address involves resetting those core bits that set ones into the Memory Buffer. This is done by sending reverse currents down all the X and Y lines of that address, and inhibit currents to these bits which should remain in the "0" state. The contents of the memory buffer could be changed before this write-cycle so that new information is entered into the address.

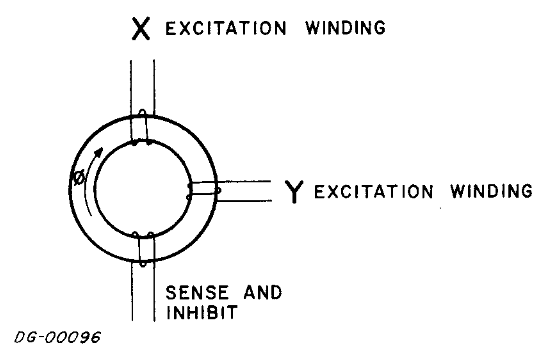


Figure M-1 Simplified Schematic of a Memory Core

A core will remain in the "one" state until currents pass through the X and Y excitation windings and force it into the "zero" state. The transition causes a pulse to travel down the sense winding to the detection logic. The core can be reset to the "one" state by reversing the currents in the X and Y windings. The transition will still cause a pulse to be generated in the sense and inhibit winding, but the sense logic is disabled at this point.

SECTION N

MAINTAINING THE COMPUTER

INTRODUCTION

The Data General Corporation supports its equipment with a large field service organization, customer training programs and technical documentation. This section summarizes these services and includes tips on preventive maintenance, recommended tools and trouble shooting.

FIELD SERVICE ORGANIZATION

Field Service Programs

Data General's Field Service Organization currently offers its users a choice of three maintenance services. These services are subject to change without notice.

1. On Call Service Contract under which DGC will repair equipment at the installation when DGC is notified of a problem by the user. DGC also provides preventive maintenance on a regular schedule under this contract. Parts, labor and travel are included in the monthly payment schedule which is determined by the type and amount of equipment to be serviced and the distance between the installation and the nearest DGC service center.

2. Factory Service Contract under which DGC will:

   (1) repair equipment when it is returned to the DGC factory in Southboro, Mass. The user assumes full responsibility for freight and insurance charges to and from the plant. Parts and labor are included in the monthly payment schedule.

   (2) repair equipment at the installation when notified of a problem by the user. Parts are included in the monthly maintenance schedule, labor is charged at reduced rates and travel is charged at the prevailing standard rates.

3. Hourly Service under which parts, labor and travel are charged as needed at prevailing rates. No contract is signed for this service.

Field Service will also generate on request a complete spare parts list for any installation, and rent or sell replacement and loaner boards.

General Terms and Conditions (Subject to change without notice).

1. Equipment which is not under a DGC service contract or normal warranty is subject to an inspection by DGC Field Service before it is eligible for a service contract. All costs for this inspection are borne by the user.

2. The user must bear all maintenance costs incurred as a result of unauthorized changes to DGC equipment. These costs will be charged as Hourly Service, regardless of the type of service contract existing between DGC and the user.

3. No additional service charge will be added for new (add-on) equipment until the warranty period of that equipment has expired.

4. All services are offered between 9 a.m. and 5 p.m. Monday through Friday excluding DGC holidays.

5. The minimum contract period is 6 months.

6. Field Service price schedules are available on request from Data General Field Service, Southboro, Mass. 01772, Telephone 617-485-9100.

TRAINING ORGANIZATION

Data General's Training Organization currently offers its users four types of training courses. These courses are subject to change without notice.

Mainframe Maintenance Course. This course covers the logical structure of the central processor, memory, operator's console and power supply. Students must have experience with digital logic, integrated circuits and computer principles.

Fundamentals of Mini-Computer Programming. This course covers number systems, logic, flow charts and computer architecture. Students should have an aptitude for mathematics.

Basic Programming. This course covers Data General's assembly language utility software including loaders, editors, debuggers and assemblers. Students should have experience in programming.

Advanced Programming. This course covers Data General's Operating Systems, DOS, RTOS and SOS. Students must have experience in programming.

Courses are scheduled regularly in the training department at Southboro, Mass., and occasionally in field offices. Special courses can be arranged.

For more information call or write

Training Department
Data General Corporation
Southboro, Mass. 01772

Tel. 617-485-9100

PREVENTIVE MAINTENANCE

Periodically carry out the checks listed in Table, N-1, and remember the following points:

1. It is very poor practice to use the equipment as a counter top, particularly for liquids like coffee or soft drinks.
2. Always check the line voltage before plugging an expensive piece of equipment into an unknown socket. (see Section I).
3. Be careful not to get metal filings into the equipment; for example never let the equipment room be cleaned with steel wool.
4. Never clean the equipment with a vacuum cleaner that has a metal (conducting) nozzle.
5. Always be aware that too much heat, moisture or contaminants can do much to harm the equipment (see Section I).
6. Be very careful how cables are routed; they should never be strained, cramped or crushed (underfoot).

Table N-1

Preventive Maintenance Check List	
Item	Check
Mechanical Connections	1. that all screws are tight and that all mechanical assemblies are secure. 2. that all crimped lugs are secure and properly inserted onto their mating connectors.
Wiring and Cables	1. all wiring and cables for breaks, cuts, frayed leads, or missing lugs. 2. wire wraps for broken or missing pins. 3. that no wires or cables are strained or cramped. 4. that cables do not interfere with doors, and that they do not chafe when doors are opened and closed.
Air Filters	all air filters for cleanliness and for normal air movement through cabinets.
Modules and Components	1. that all modules are properly seated. Look for areas of discoloration on all exposed surfaces. 2. all exposed capacitors for signs of discoloration, leakage, or corrosion. 3. power supply capacitors for bulges.
Indicators and Switches	all indicators and switches for tightness; check for cracks, discoloration, or other visual defects.
Fans	for broken fan blades.
Diagnostics	Run all diagnostics periodically

Table N-2

Recommended Maintenance Tool Kit			
ITEM	QTY	DESCRIPTION	MFG. & PART No.
1	1	6" combination slip joint pliers	Utica # 5-6
2	2	5 1/2" needle nose pliers	Utica # 654-5 1/2
3	1	4" needle nose pliers	Utica # 23-4
4	1	5" diagonal wire cutters	Utica # 44-5
5	1	4" diagonal wire cutters	Utica # 347-4 CFJS
6	1	5" ignition pliers	Utica # 517-5
7	1	Screwdriver kit including handle, 3/16", 1/4", 5/16" slotted #1, #2 phillips blades, each 4" long	Xcelite # 99 PV-6
8	1	3/32 slotter screwdriver with 2" blade	Xcelite # R3322
9	1	1/8" #0 phillips screwdriver	Xcelite # P12S
10	1	Magnetic pick up tool	Bonney # K26
11	1	3/32 through 3/8, 10 pc nut driver set	Xcelite # PS120
12	1	Xacto knife	
13	1	6" adjustable wrench	Utica # 91-6
14	1	Ignition wrench	Bonney # N24R
15	1	Set of 25 feeler gauges with 3" blades	Bonney # K53
16	1	Set of 15 hex keys	Bonney # N6R
17	1	Slotter 5" screw starter	Bonney # 5527
18	1	Phillips 6 1/4" screw starter	Bonney # 556
19	1	5" adjustable wire strippers	Utica # 110-5
20	1	Set of 4 cut needle files	Hunter # F228A
21	1	4 1/2" electrical tweezers	Hunter # B3M3
22	1	flash light	
23	1	Can Quick Freez (circuit cooler)	

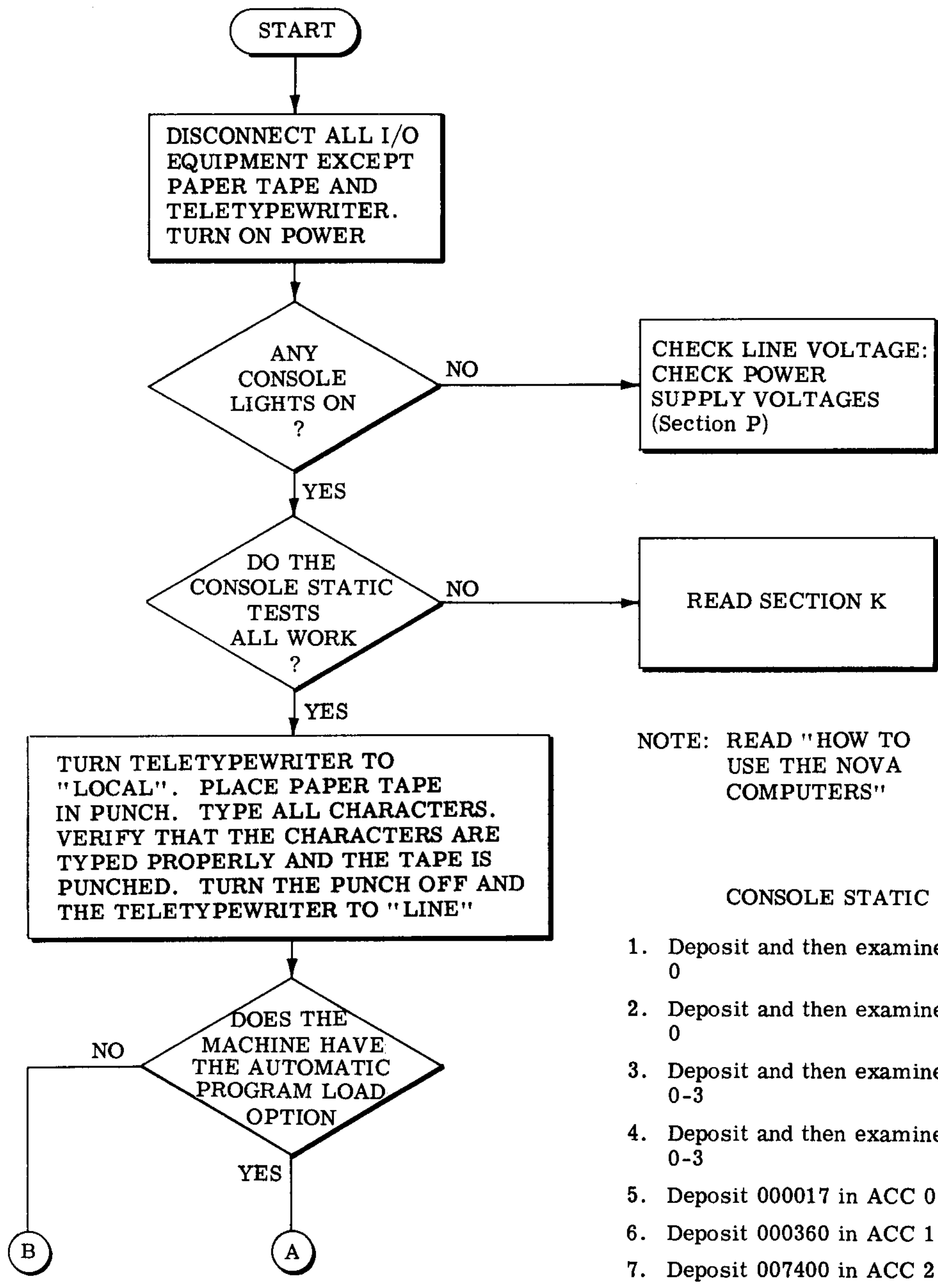
Table N-2 (Continued)

Recommended Maintenance Tool Kit			
ITEM	QTY	DESCRIPTION	MFG & PART No.
24	1	Can degreaser (flux remover)	
25	2	16P I/C test clip	
26	1	23 1/2 watt soldering iron with iron plated chisel tip	Ungar
27	1	47 1/2 watt soldering iron element	
28	1	1lb, 60/40 resin core solder	Kester
29	3	Spools of solder wick	
30	2	Acid brushes	
31	1	Vacuum solder removal tool	
32	1	Multimeter	Simpson # 260
33	1	Tool carrying case	
34	1	Oscilloscope	Tektronics # 453
35	1	Current probes	Tektronics # P60-22

Table N-3

The Nova 1220 Diagnostics			
Diagnostic	Part No.	Binary Tape No.	Description
Address Test	097-000007	095-000005	checks memory address selection logic
Checkerboard III	097-000014	095-000031	tests memory sense amplifiers and inhibit logic
Nova 1220 Logic Test	097-000017	095-000036	tests CPU logic other than I/O
Nova 1220 Instruction Timer	097-000019	095-000038	tests CPU clock logic and outputs time-to-complete for each instruction
Exerciser	097-000004	095-000012	tests CPU logic, teletypewriter, reader, punch and real-time clock;
Arithmetic Test	097-000018	095-000037	exercises arithmetic and logical instructions in CPU

HOW TO TEST THE COMPUTER

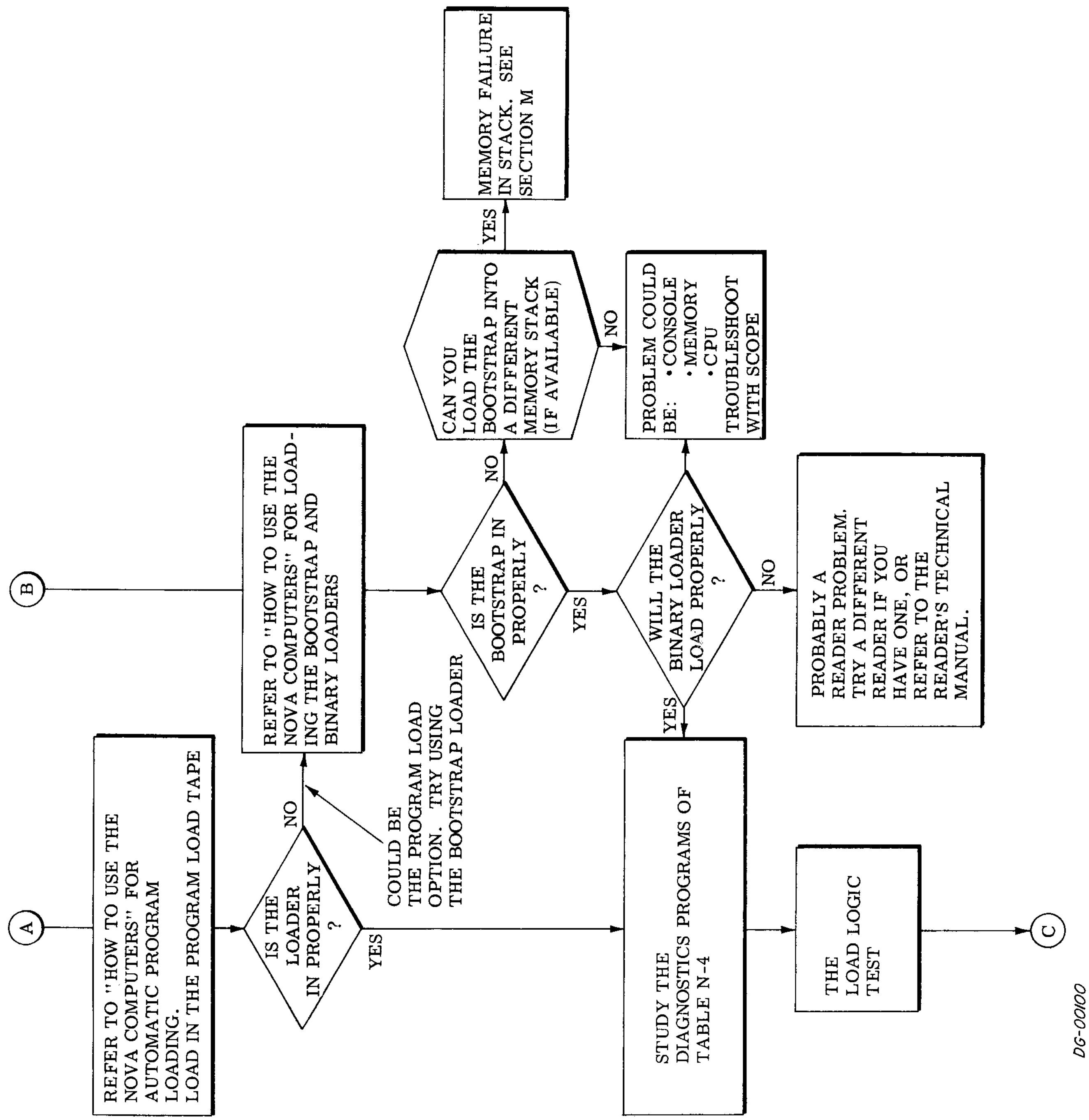


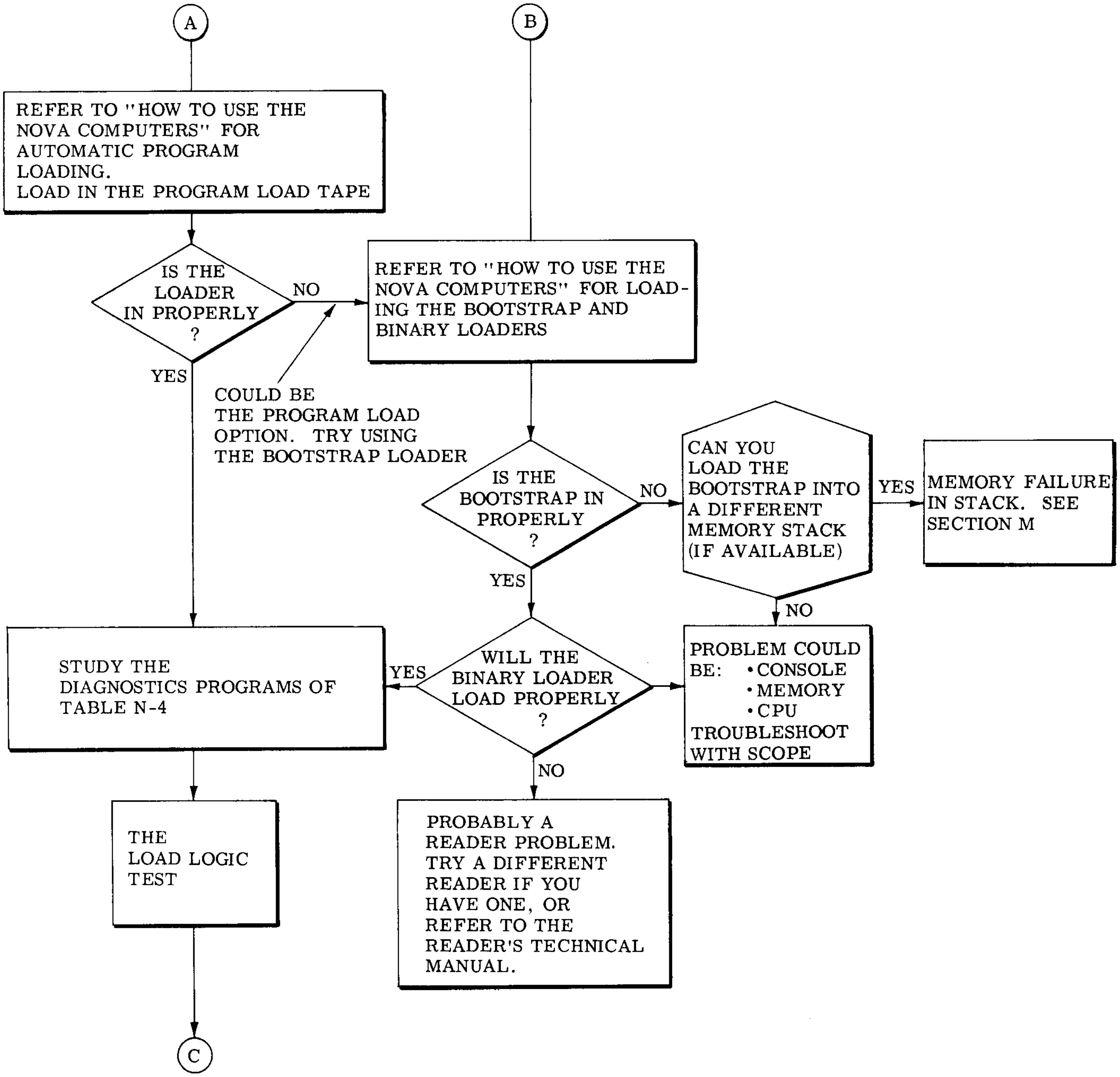
DG-00102

NOTE: READ "HOW TO USE THE NOVA COMPUTERS"

CONSOLE STATIC TESTS

1. Deposit and then examine 000000 in location 0
2. Deposit and then examine 111111 in location 0
3. Deposit and then examine 000000 in ACC 0-3
4. Deposit and then examine 111111 in ACC 0-3
5. Deposit 000017 in ACC 0 and verify
6. Deposit 000360 in ACC 1 and verify
7. Deposit 007400 in ACC 2 and verify
8. Deposit 170000 in ACC 3 and verify
9. Deposit 000000 in location 000000
10. Raise START/CONTINUE switch to START. RUN indicator should be lit
11. Lock the computer and try pushing the RESET/STOP switch. Computer should keep running.
12. Unlock the computer and hit STOP.
13. Depress DEPOSIT NEXT several times and check that the PC increments
14. Depress EXAMINE NEXT several times and check that the PC increments


A
REFER TO "HOW TO USE THE NOVA COMPUTERS" FOR AUTOMATIC PROGRAM LOADING. LOAD IN THE PROGRAM LOAD TAPE
IS THE LOADER IN PROPERLY ?
YES
NO
COULD BE THE PROGRAM LOAD OPTION. TRY USING THE BOOTSTRAP LOADER
B
REFER TO "HOW TO USE THE NOVA COMPUTERS" FOR LOADING THE BOOTSTRAP AND BINARY LOADERS
IS THE BOOTSTRAP IN PROPERLY ?
NO
YES
CAN YOU LOAD THE BOOTSTRAP INTO A DIFFERENT MEMORY STACK (IF AVAILABLE)
YES
NO
MEMORY FAILURE IN STACK. SEE SECTION M
PROBLEM COULD BE: •CONSOLE •MEMORY •CPU TROUBLESHOOT WITH SCOPE
WILL THE BINARY LOADER LOAD PROPERLY ?
NO
YES
PROBABLY A READER PROBLEM. TRY A DIFFERENT READER IF YOU HAVE ONE, OR REFER TO THE READER'S TECHNICAL MANUAL.
STUDY THE DIAGNOSTICS PROGRAMS OF TABLE N-4
THE LOAD LOGIC TEST
C
DG-00100


A
B
REFER TO "HOW TO USE THE NOVA COMPUTERS" FOR AUTOMATIC PROGRAM LOADING. LOAD IN THE PROGRAM LOAD TAPE
IS THE LOADER IN PROPERLY ?
NO
YES
REFER TO "HOW TO USE THE NOVA COMPUTERS" FOR LOADING THE BOOTSTRAP AND BINARY LOADERS
COULD BE THE PROGRAM LOAD OPTION. TRY USING THE BOOTSTRAP LOADER
IS THE BOOTSTRAP IN PROPERLY ?
NO
YES
CAN YOU LOAD THE BOOTSTRAP INTO A DIFFERENT MEMORY STACK (IF AVAILABLE)
YES
MEMORY FAILURE IN STACK. SEE SECTION M
NO
STUDY THE DIAGNOSTICS PROGRAMS OF TABLE N-4
YES
WILL THE BINARY LOADER LOAD PROPERLY ?
PROBLEM COULD BE: • CONSOLE • MEMORY • CPU TROUBLESHOOT WITH SCOPE
NO
THE LOAD LOGIC TEST
PROBABLY A READER PROBLEM. TRY A DIFFERENT READER IF YOU HAVE ONE, OR REFER TO THE READER'S TECHNICAL MANUAL.
C

DG-00100

SECTION M

THE MEMORY

A REVIEW OF CORE MEMORIES

A "bit" of information can be stored in a ferrite core by magnetizing the core in one of two possible directions or "states" and then calling one state a "1" and the other state a "0", similar to a flip-flop. Unlike a flip-flop, however, a core cannot be read simply by examining its output voltages; a core is read by forcing it into the "0" state and then watching for the current pulse which is always generated when a core changes state. If the pulse occurs, then the core must have been in the "1" state before it was excited; if no pulse occurs then the core must already have been in the "0" state because no transition took place.

Reading a core, then, always leaves it in the "0" state and although the information that it contained has probably been transferred to some register which was set by the current pulse, that information is no longer in the core, and it usually has to be restored with what is called a "write cycle". Writing means setting the core to a one or a zero, depending on the state of the memory register that usually contains core bound information.

Reading or writing into a core is a matter of sending current pulses along wires into the core; the direction of current relative to the core determines into which state the core will move.

Data General's core memories contain many thousands of these ferrite cores strung together like beads on wire. Each core has three wires passing through it, and these wires carry the currents to magnetize them and the pulses which occur when they change state. The memories are wired so that the computer can select any group of 16 bits at once, and read or write a complete 16 bit word "in parallel". A group of 16 cores, called an "address" is picked by passing current down two selected wires called X and Y, which are strung into the cores so that they both pass through only one address. The combined effect of current in these two wires is enough to flip the core into the zero state if it is not already there. Each core that flips sends a pulse down its own third wire called the sense wire which is then fed into one flip-flop of a 16 bit Memory Buffer. The flip-flop sets if it sees a pulse, and remains static if it does not. The register which selects the X Y wire or "lines" is called the Address Register.

Restoring the contents of the address involves resetting those core bits that set ones into the Memory Buffer. This is done by sending reverse currents down all the X and Y lines of that address, and inhibit currents to these bits which should remain in the "0" state. The contents of the memory buffer could be changed before this write-cycle so that new information is entered into the address.

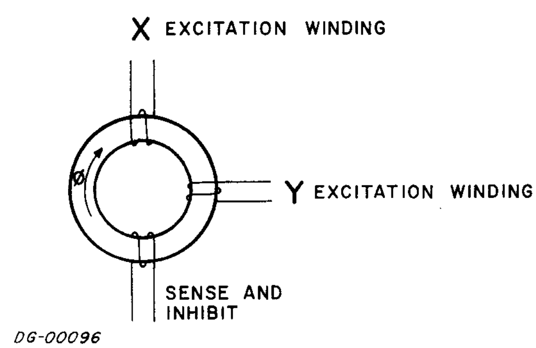


Figure M-1 Simplified Schematic of a Memory Core

A core will remain in the "one" state until currents pass through the X and Y excitation windings and force it into the "zero" state. The transition causes a pulse to travel down the sense winding to the detection logic. The core can be reset to the "one" state by reversing the currents in the X and Y windings. The transition will still cause a pulse to be generated in the sense and inhibit winding, but the sense logic is disabled at this point.

DATA GENERAL'S CORE MEMORIES

The memories used on the basic computer consist of cores arranged in a three wire 3D scheme in which the sense and inhibit functions share the same wire. The cores are laid out in a single plane in mats, and wired together in the bow tie pattern shown in Figure M-2. There are four core planes available; 1K, 2K, 4K, and 8K. Each plane is assembled on a "daughter" board which is mounted on a 15" by 15" "mother" board, where most of the memory logic sits. Power is supplied by the chassis supply.

The memory logic on any board consists of drivers, sense amplifiers, a Memory Address Register, a Memory Buffer Register, Multiplexers, and Memory select logic shown in Figure M-3.

Data is transferred between memory and the central processor or an I/O device along three data buses called:

$\overline{\text{MEM}}$	which transfers data from memory to the Central Processor;
$\overline{\text{MBO}}$	which transfers data from the Central Processor to Memory
$\overline{\text{DATA}}$	which transfers data between memory and I/O devices in either direction.

The Memory Select Logic

When a memory board is plugged into a computer, its select logic must be wired to respond to the correct code in the MA register, since the MA registers of all boards are loaded with the same address at the same time. This wiring is done with a set of jumpers that connect either the 0 or 1 side of the high order MA bits to an "and" gate. The output of this "and" gate will be true only if the code for which it is wired is in the MA register, and only when this output is true can the memory respond. This code must be unique to that memory board.

The jumpers are forced into points on the board. These points are located on the logic side of the board at the lower right hand corner when its fingers are pointing at you. If there is a mixture of boards, i.e., 1K, 2K, 4K or 8K, it is a good policy to wire the largest board for low core, the second largest above it and so on. This way there will not be any gaps in the system's core map.

Figures M-4 and M-5 show how the select logic of the four types of boards are jumpered.

REFERENCES:

8K	Memory Prints	#001-000238-00
4K	Memory Prints	#001-000236-00
2K	Memory Prints	#001-000234-00
1K	Memory Prints	#001-000232-00

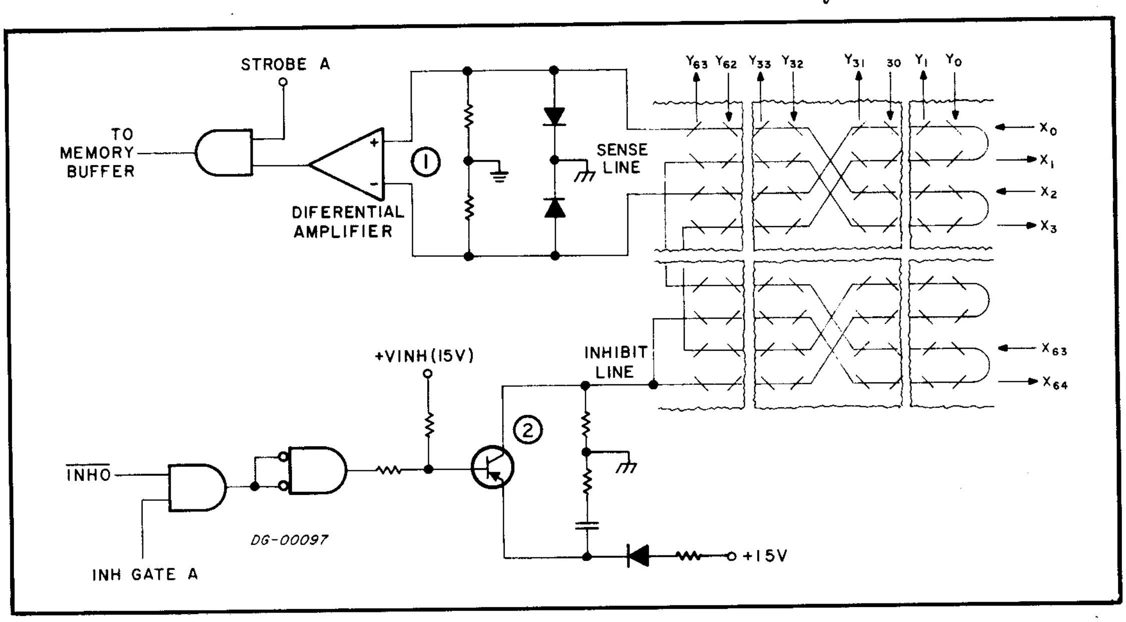
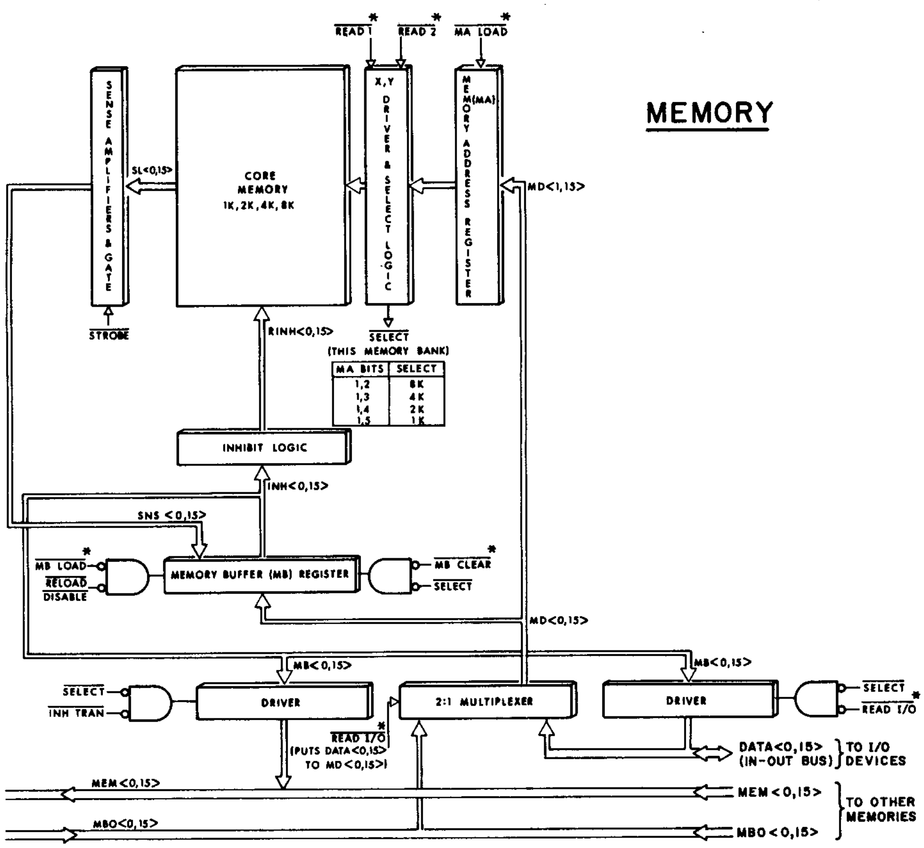


Figure M-2 Simplified Schematic of The Core Memory's Sense and Inhibit Lines

The sense and inhibit functions share the same wire. The sense circuitry, (1), sees both ends of the wire, and detects negative pulses with a differential amplifier. The output of this amplifier is examined at STROBE time.

The inhibit logic, (2), drives +15Vdc level into the middle of the same wire at INHIBIT time. The current is divided and passes through all cores to ground through the diodes at the other end.

MEMORY

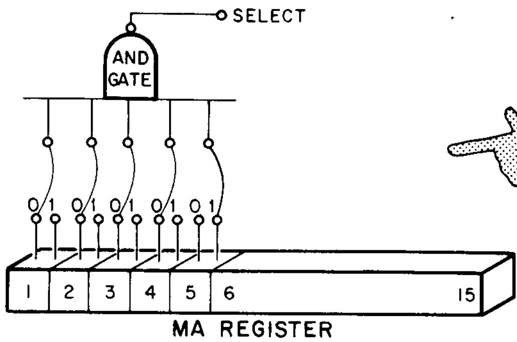


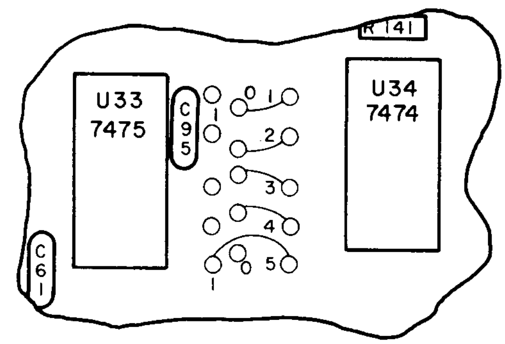
MA BITS	SELECT
1,2	8K
1,3	4K
1,4	2K
1,5	1K

* Issued by CPU

Figure M-3 Core Memory

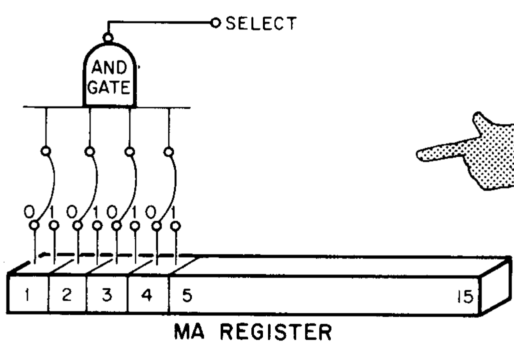
During a typical FETCH instruction, the CPU outputs the memory address on the MBO <0, 15> data lines and then issues $\overline{\text{MA LOAD}}$. READ I/O is high, so the address is strobed into the Memory Address register and output to the driver select logic. Then, READ 1 and READ 2 are issued,gating the X and Y currents to the selected address. A little later,STROBE is output by the CPU and it gates all core pulses into their corresponding Memory Buffer bits. The Memory Buffer is then re-read back into core by reversing all the driver currents and gating the INHIBIT signal issued by the CPU to those bits which are not to be reset. If the contents of the address are to change, the Memory Buffer is loaded with the new word before the address is re-written.

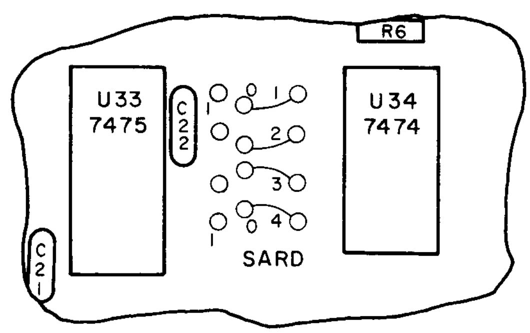




1K BOARDS						
MA BITS JUMPERED					BOARD NUMBER	ADDRESSES ENABLED (OCTAL)
1	2	3	4	5		
0	0	0	0	0	1	00000-01777
0	0	0	0	1	2	02000-03777
0	0	0	1	0	3	04000-05777
0	0	0	1	1	4	06000-07777
0	0	1	0	0	5	10000-11777
0	0	1	0	1	6	12000-13777
0	0	1	1	0	7	14000-15777
0	0	1	1	1	8	16000-17777

Selecting 1K Memory Boards. On the lower right hand side of the board between U33 and U34 there are 3 sets of 5 points. The first two sets are wired to MA <1, 5> on the 1 and 0 side respectively; the last set of points is wired to the "and" gate. The board of this figure is wired for 00001, board #2.



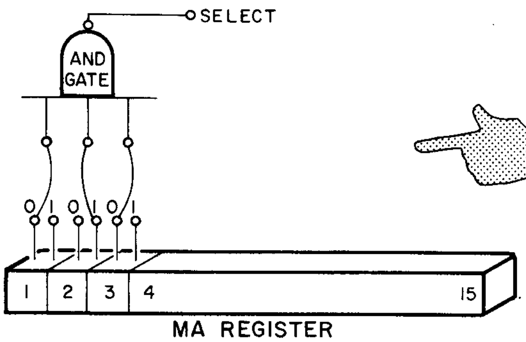


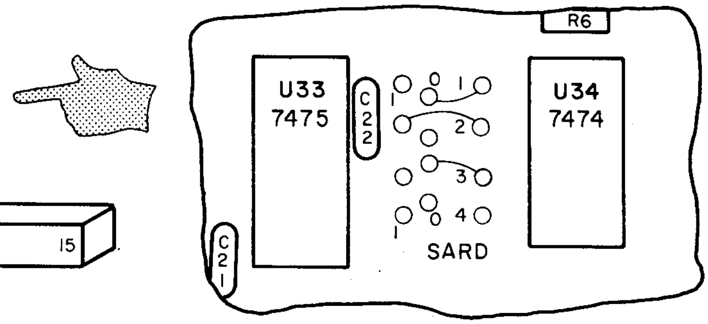
2K BOARDS					
MA BITS JUMPERED				BOARD NUMBER	ADDRESSES ENABLED (OCTAL)
1	2	3	4		
0	0	0	0	1	00000-03777
0	0	0	1	2	04000-07777
0	0	1	0	3	10000-13777
0	0	1	1	4	14000-17777
0	1	0	0	5	20000-23777
0	1	0	1	6	24000-27777
0	1	1	0	7	30000-33777
0	1	1	1	8	34000-37777

DG-00093A

Selecting 2K Memory Boards. On the lower right hand side of the board between U33 and U34 there are 3 sets of 4 points. The first two sets are wired to MA <1, 4> on the 0 and 1 side of each flip-flop; the last four points are wired to the "and" gate. The board of this figure is wired for 0000, board #1.

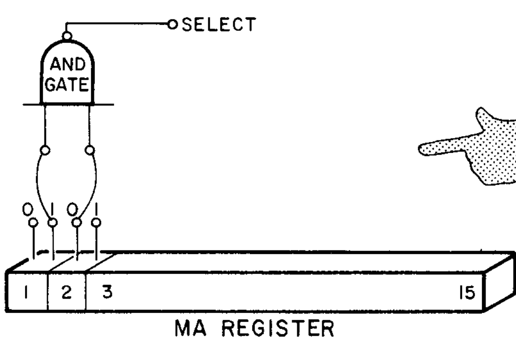
Figure M-4 Wiring Up The Select Logic of 1K and 2K Boards

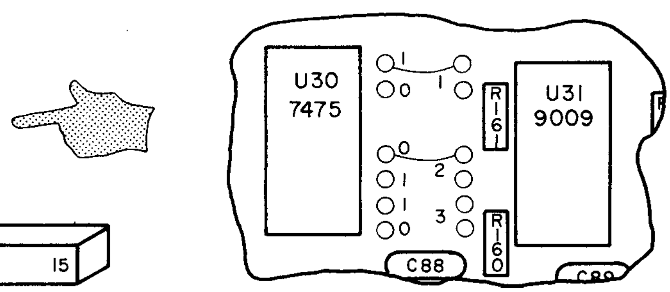




4K BOARDS						
MA BITS JUMPERED					BOARD NUMBER	ADDRESSES ENABLED (OCTAL)
1	2	3				
0	0	0			1	00000-07777
0	0	1			2	10000-17777
0	1	0			3	20000-27777
0	1	1			4	30000-37777
1	0	0			5	40000-47777
1	0	1			6	50000-57777
1	1	0			7	60000-67777
1	1	1			8	70000-77777

Selecting 4K Memory Boards. On the lower right hand side of the board between U33 and U34 there are 3 sets of 4 points. The first two sets are wired to MA <1, 3> on the 1 and 0 sides respectively, the last set is wired to the "and" gate. The board of this figure is wired for 010, board #3. Sard 4 should NOT be jumpered.





8K BOARDS						
MA BITS JUMPERED					BOARD NUMBER	ADDRESSES ENABLED (OCTAL)
1	2					
0	0				1	00000 - 17777
0	1				2	20000 -37777
1	0				3	40000 -57777
1	1				4	60000 -77777

DG-00095B

Selecting 8K Memory Boards. On the lower right hand side of the board between U30 and U31 there are 2 sets of 6 points. The first set is wired to MA <1,3> on the 1 and 0 sides; the second set is wired to the "and" gate. The board of this figure is wired for 10, board #3. Position 3 should NOT be jumpered.

Figure M-5 Wiring Up The Select Logic of 4K and 8K Boards

Table M-1

External Memory Signals

SIGNAL NAME	FUNCTION
DATA <0, 15>	16 bidirectional lines which carry information to and from devices on the IN-OUT bus.
DRIVE I/O	Issued by CPU-1 to strobe the MB register onto DATA <0, 15> lines.
INH TRAN	Issued by CPU-1 to prevent the MB register from outputting to the MEM <0, 15> bus during a data transfer from the console.
INHIBIT SELECT	Issued by CPU-1 to prevent the memory from being selected.
MA LOAD	Issued by CPU-1 to load the MA register.
MEM <0, 15>	16 lines which carry information from the memory to CPU-1.
MB CLEAR	Issued by CPU-1 to clear the MB register.
MB LOAD	Issued by CPU-1 to load the MB register.
READ 1	Issued by CPU-1 to select the memory drivers.
READ 2	Issued by CPU-1 to select memory drivers.
READ I/O	Issued by CPU-1 to enable the DATA <0, 15> lines into the MD <1-15> lines.
RELOAD DISABLE	Issued by CPU-1 to inhibit MB Load.
STROBE	Issued by CPU-1 to strobe core pulses into the Memory Buffer.
MBO <0, 15>	16 lines which carry information from CPU-1 to memory.

SECTION I

NOVA 1220 INSTALLATION

INTRODUCTION

This section explains how to unpack, assemble and cable the computer.

PLACING THE COMPUTER

The computer room must be large enough to accommodate the equipment, operating personnel, tables and chairs, storage space (for tapes, manuals and listings), service clearances and possible future expansion. The room should be well lit and clean, with adequate primary power. The temperature and humidity must fall within acceptable tolerances of the most sensitive peripheral.

Overhead sprinklers should be "dry pipe" systems that remove primary power from the room and turn on a battery operated light source before opening the master valve. If power connections are made under the floor, use waterproof receptacles and connections. Any carpeting should be of the type that minimizes static electricity, and metal flooring should be well grounded.

UNPACKING THE COMPUTER

The computer is shipped in the kit shown in Figure I-1.

1. Open the top of the outer carton; remove all cables, manuals, packing filler, etc.
2. Remove the styrofoam container (it and contents weigh about 50 pounds) and place it on a flat surface right side up.
3. Unstrap the container and remove the cover and styrofoam spacers.
4. Carefully remove the styrofoam block from the back of the computer.
5. Remove the computer, placing your hands under the chassis front and back.
6. The computer is sometimes shipped with cardboard spacers in spare slots to keep the boards from vibrating during shipment. Remove these.

Table I-1

The Nova 1220 Electrical, Mechanical and Environmental Specifications

Voltage (AC)	Current (A) NOMINAL @ 115V	Power Dissipation (W)	Heat Dissipation (Btu/hr)	Operating Temperature (min-max F)	Storage Temperature (min-max F)	Humidity (Rel) (min - max)	Maximum Wet Bulb	Maximum Cable Length	Dimensions (inches)	Service Clearance (inches)	Weight (lbs)
110	9	1000	3400	32-130	-30-+160	20% 90%	78°F	IN-OUT 50FT	HEIGHT 10½" WIDTH 17½" LENGTH 22¼"	BACK 3" FRONT 36"	PACKED 65 UN-PACKED 45

The Nova 1220 operates from a single-phase source at 115V 60Hz or ±50Hz all ±20%. This device has a separate 4.5 foot power cord terminating in a standard 3 wire single-phase male connector. An earth ground connection must be supplied through the power cord.

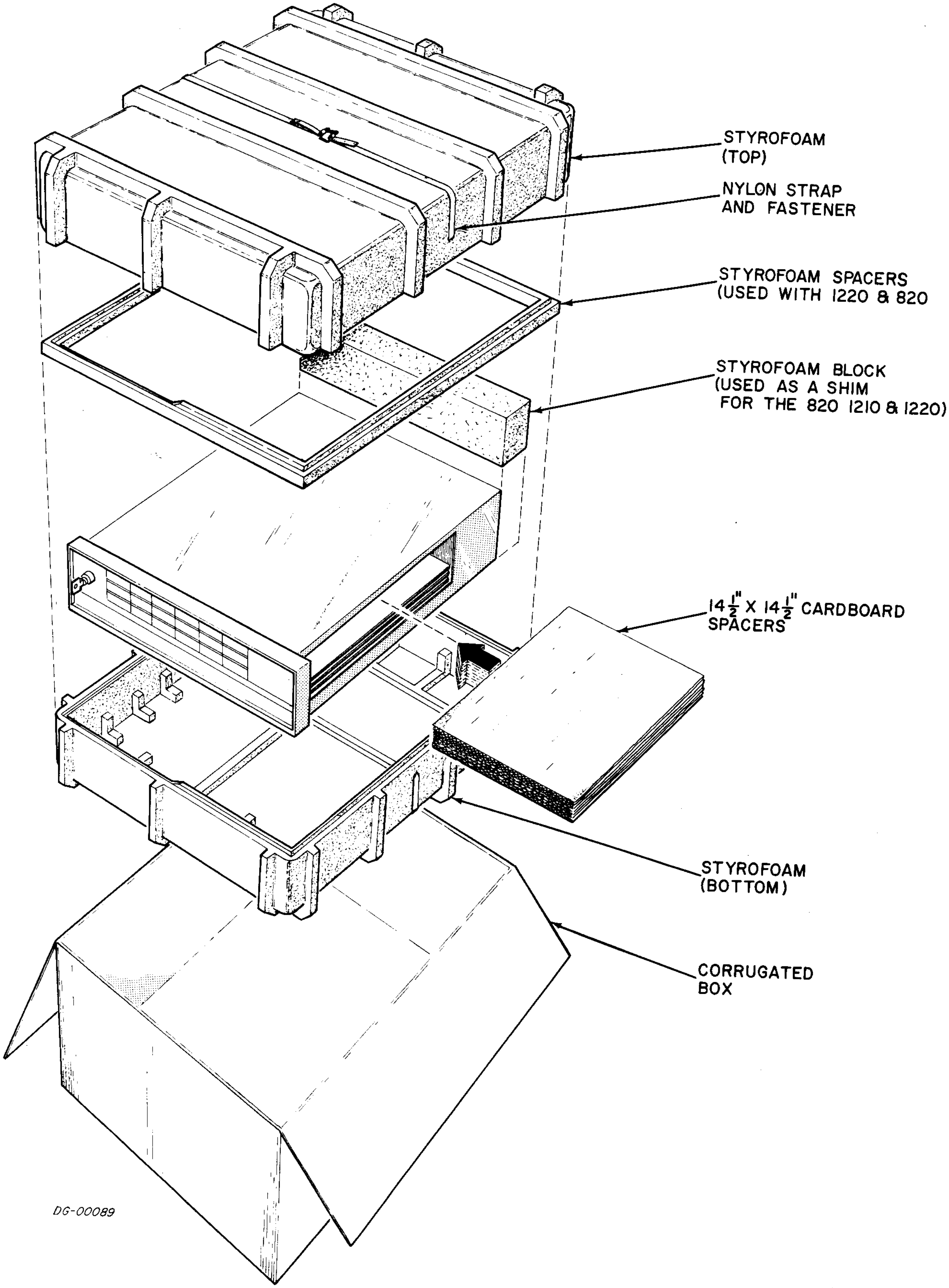


Figure I-1 The Nova 1220 Shipping Kit

PACKING THE COMPUTER

1. Locate the original shipping container and packing material. If it is not available, order a shipping kit from Data General Corporation. DO NOT SHIP THE COMPUTER IN ANY OTHER CONTAINER.
2. Fill any large spaces inside the chassis with just enough cardboard spacers so the boards cannot vibrate.
3. Place the computer in the bottom half of styrofoam container "front justified" with the back end on top of the extra rib. Pack the power cord into the hollow area at the back. Fill in the space at the back with the styrofoam block to prevent the computer from moving during shipment.
4. Add the styrofoam spacers as needed.
5. Put on the cover of the styrofoam container and strap the pieces together.
6. Put the styrofoam container into the cardboard box. Place any odds and ends on top of the container, and fill in any empty spaces with cardboard or pieces of styrofoam.
7. Close and seal the cardboard box.
8. Call your local Field Service representative for the correct address if the equipment is to be shipped to Data General Corporation.

ASSEMBLING THE COMPUTER

Assembling the computer outside the factory involves installing memory or controller boards or mounting the chassis into a 19" rack.

Installing or Removing Boards

The Nova 1220 computer has slots for ten 15 X 15 inch circuit boards which plug into ten sets of 100 pin connectors on the PC backpanel. The slots are numbered from the bottom up and assigned as follows:

Slot Number	Boards Accepted
1	CPU-1 Only
2	Any 1220 Memory or the Multiply Divide option (8107)
3	Any 1220 Memory or the I/O Interface Assembly (4007)
4-8	Any 1220 Memory or Controller
9,10	Any 1220 Controller

Note that slot 3 has special wiring for the 4007.

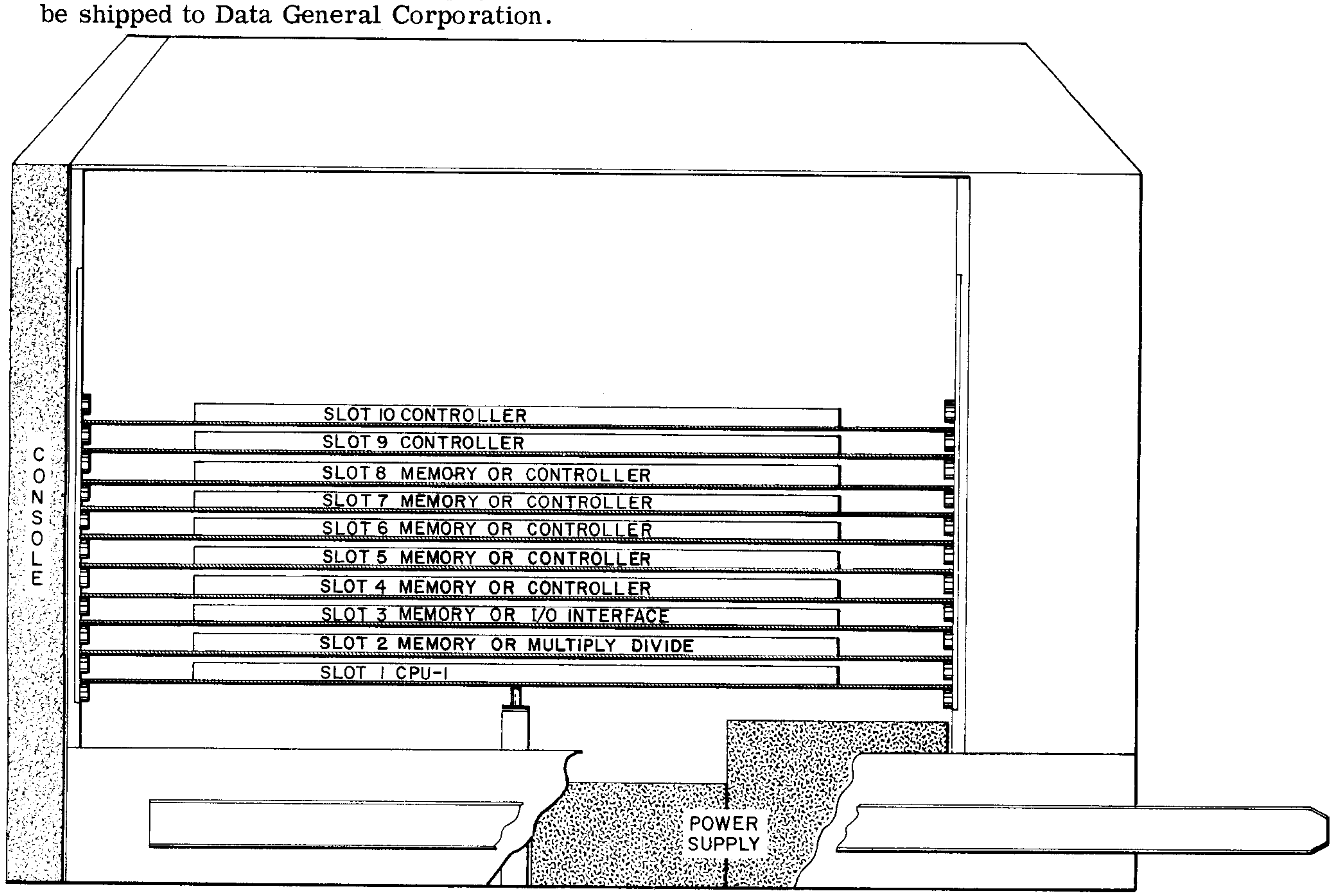


Figure I-2 Nova 1220 Board Slots

1. Fasten the two slide brackets to the rail using one #8 screw at the back and two #8 screws at the front. Note carefully which holes in the rail the screws go into.
2. Fasten the two nut plates to their vertical rails with six #10 screws, but do not tighten the screws. Note that the holes in the vertical rail and those in the nut plate are not evenly spaced but they can and must line up.
3. Insert the slide brackets between their vertical rails and nut plates and tighten the #10 screws.
4. Fasten the strike plate to the vertical rail using two #10 screws.

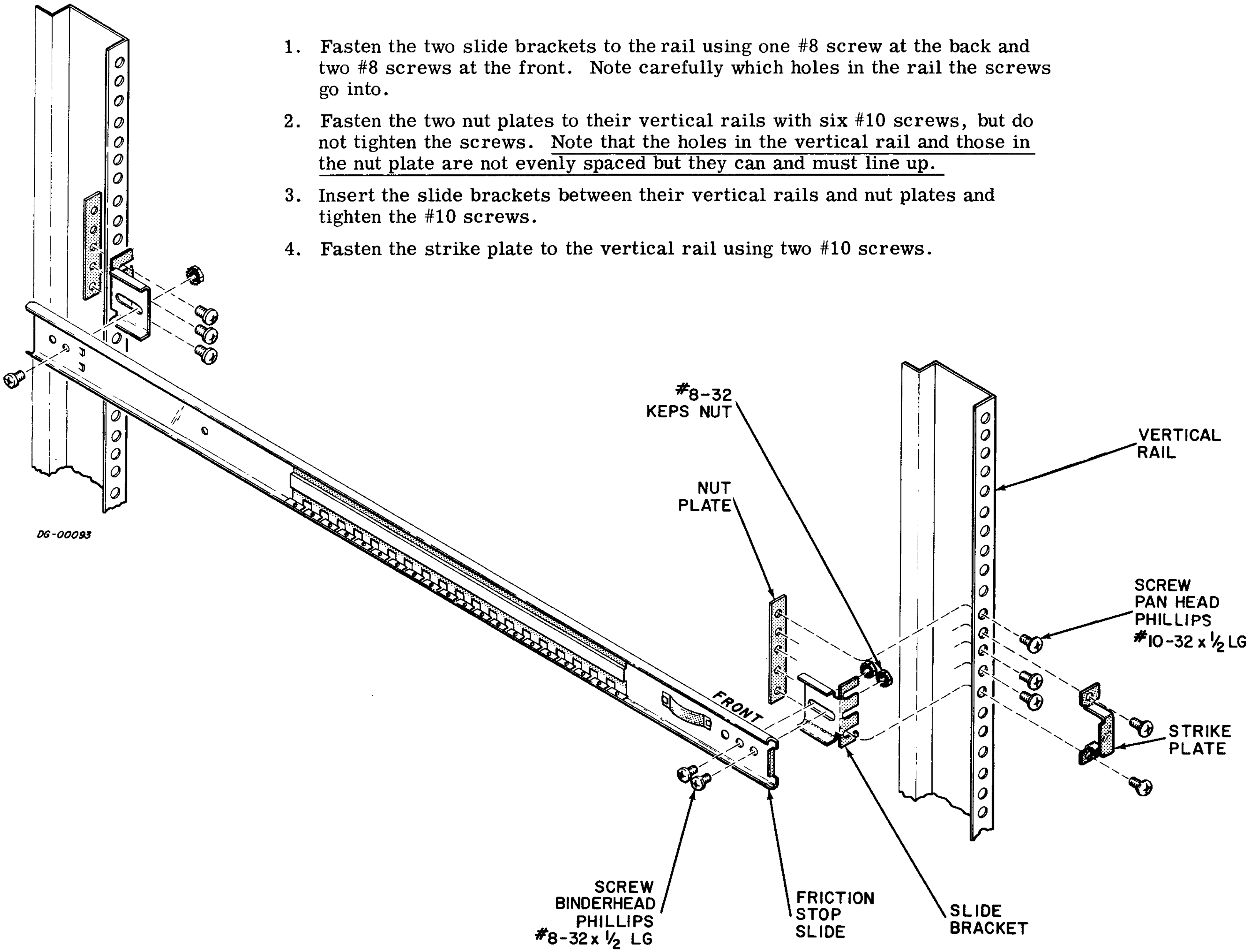


Figure I-3 Rack Mounting Hardware For The Nova 1220

Note that if the Multiply Divide option 8107 is used, it must go into slot 2, and if the I/O Interface Assembly is used it must go into slot 3. If a new memory board is installed, check that the select logic jumpers are correct (See Section M).

If boards are installed or removed from the computer chassis, it is important that the integrity of the Program Interrupt and Data Channel priority systems be preserved. The Priority systems of the Program Interrupt and Data Channel facilities each use a scheme in which a wire is chained through every controller, one after the other, in such a way that only when there is an enabling level on that wire can a controller effectively request service of the facility. The enabling level on the wire will appear at any given controller only if all controllers closer to the computer on the chain are not requesting service themselves; i.e., whenever a controller requests service it removes the enabling level from all devices below it on the chain. There are two chains, one for the Program Interrupt and the other for the Data Channel.

The program interrupt chain enters a board slot at pin A96 and leaves at pin A95; the data channel chain enters at pin A94 and leaves at pin A93. (See "How to Use the Nova Computers" for more details.)

Here are the rules:

1. Memories take Data Channel and Program Interrupt signals and pass them through their slots.
2. All controllers that use the interrupt system must be included in the interrupt chain; all controllers that use the data channel must be included in the data channel chain.
3. The Data Channel and Program Interrupt chains are completely independent and must not cross. Each chain must run through the controllers in series, NEVER in parallel.
4. Controllers that use the Program Interrupt system but do not use the Data Channel system do not need a jumper for the unused line. The only jumpering required is on unused slots or the user's manufactured boards.

Rack Mounting The Computer

The Nova 1220 can be mounted in a standard 19 inch rack, so each unit is shipped with rack slides attached and all of the necessary mounting hardware included. Figure I-3 shows how the right side of the rack slide is assembled in a cabinet; the other side uses identical hardware.

Leave at least two inches open at the back for cables and about 36" open at the front for servicing. The console protrudes 1 3/4 inches out of the front of the rack.

CABLING ASSEMBLIES TOGETHER

Types of Cables

There are five types of cables used on a typical installation; I/O cables, device cables, internal cables, interdevice cables, and adapter cables. The correct cables are supplied with the equipment unless otherwise specified in the price list.

<u>I/O Cables</u> which connect peripheral controllers mounted outside the computer chassis, to the computer IN-OUT bus. The cables form a daisy chain, from controller to controller and finally to the computer chassis, where the first cable must terminate in a female connector compatible with the 100 finger male called P3 shown in Figure I-4. Controllers mounted inside the chassis are connected to the IN-OUT bus through backpanel etching, and therefore do not need an I/O cable.

<u>Device Cables</u> which connect each peripheral controller to the device it is controlling. When such a controller is inserted into the Nova 1220 chassis, an internal cable is run from the appropriate backpanel pins to a male connector, such as P3 of Figure I-4. The device cable must then run between the male paddle board on the 1220 chassis and the device.

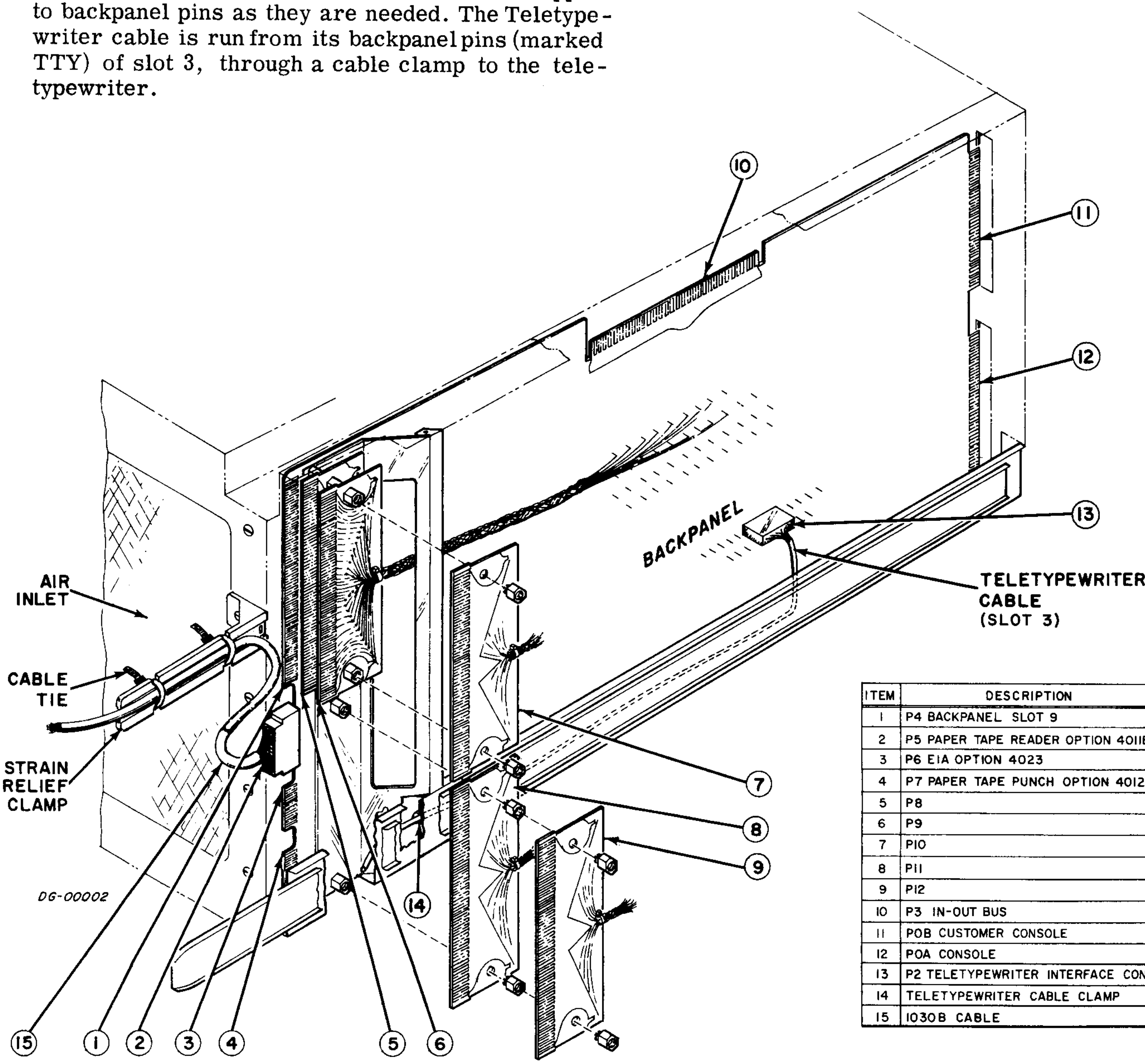
<u>Internal Cables</u> are added when the controller is added, whether in the factory or in the field, so each shipment includes a wire list for the internal cable, and the internal cable itself. Figure I-4 shows how the paddle boards are mounted on the chassis.

<u>Interdevice Cables</u> interconnect peripheral devices. Some controllers will drive more than one device of the same kind, such as industry compatible tape controllers. In this case the device cables are daisy chained from device to device in the same way that the I/O cables are chained between controllers. The cables which interconnect the devices are not always the same as the device cable that runs from the controller to the first device, however, so these cables are called "interdevice cables".

<u>Adapter Cables</u> reconcile different cabling schemes. The Nova, Supernova, Nova 1200 and Nova 800 series computers use Cannon connectors instead of paddle boards for their device and I/O cables, and Data General supplies adapters so that peripherals used on these machines can also be used on the new models, or the other way around.

Figure I-4 Sketch of the Nova 1220 Cabling Schemes

Signals from the backpanel pins are connected to edge connectors called P3-P12, which are mounted parallel to the backpanel. The fingers of P3 are permanently connected to the IN-OUT Bus signals according to Table I-2 via etched tracks on the backpanel PC board. The fingers of P4 are permanently connected to pins of slot 9 according to Table I-3. P5, P6 and P7 are all part of a three plug 60 finger paddle board which is permanently connected, but used only when the paper tape reader, the paper tape punch or the EAI options are installed in slot 3. P8-P12, 100 finger paddle boards which accept 48 signal wires and two ground wires, can be mounted on standoffs next to P4 or P5, P6, P7 and wire wrapped to backpanel pins as they are needed. The Teletypewriter cable is run from its backpanel pins (marked TTY) of slot 3, through a cable clamp to the teletypewriter.



ITEM	DESCRIPTION
1	P4 BACKPANEL SLOT 9
2	P5 PAPER TAPE READER OPTION 4011B
3	P6 EIA OPTION 4023
4	P7 PAPER TAPE PUNCH OPTION 4012A
5	P8
6	P9
7	P10
8	P11
9	P12
10	P3 IN-OUT BUS
11	P0B CUSTOMER CONSOLE
12	P0A CONSOLE
13	P2 TELETYPEWRITER INTERFACE CONNECTOR
14	TELETYPEWRITER CABLE CLAMP
15	1030B CABLE

Table I-2

P3 Interconnections for Nova 1220

P3 LETTER SIDE	P3 NUMBER SIDE	SIGNAL NAME
	1 THRU 50	GND
A	— — — — — —	GND
B		PWR ON (+5V)
C		$\overline{\text{MSKO}}$
D		INTA
E		DATIB
F	— — — — — —	DATIA
H		$\overline{\text{DS3}}$
J		DATOC
K		CLR
L		STRT
M	— — — — — —	DATIC
N		DATO B
P		DATO A
R		$\overline{\text{DCHA}}$
S		$\overline{\text{DS4}}$
T	— — — — — —	$\overline{\text{DS5}}$
U		$\overline{\text{DS2}}$
V		$\overline{\text{DS1}}$
W		IORST
X		$\overline{\text{DS0}}$
Y	— — — — — —	IO PLS
Z		$\overline{\text{SELD}}$
a		$\overline{\text{SELB}}$
b		$\overline{\text{DCHP OUT}}$
c		$\overline{\text{INTP OUT}}$
d	— — — — — —	$\overline{\text{DCHM0}}$
e		$\overline{\text{DCHM1}}$
f		$\overline{\text{INTR}}$
h		DCH0
j		DCHR
k	— — — — — —	DCH1
l		$\overline{\text{OVFLO}}$
m		$\overline{\text{RQENB}}$
n		$\overline{\text{DATA7}}$
p		$\overline{\text{DATA14}}$
r	— — — — — —	$\overline{\text{DATA5}}$
s		$\overline{\text{DATA11}}$
t		$\overline{\text{DATA12}}$
u		$\overline{\text{DATA8}}$
v		$\overline{\text{DATA4}}$
w	— — — — — —	$\overline{\text{DATA0}}$
x		$\overline{\text{DATA9}}$
y		$\overline{\text{DATA13}}$
z		$\overline{\text{DATA1}}$
AA		$\overline{\text{DATA15}}$
AB	— — — — — —	$\overline{\text{DATA3}}$
AC		$\overline{\text{DATA10}}$
AD		$\overline{\text{DATA2}}$
AE		$\overline{\text{DATA6}}$
AF		GND

Table I-3

P4 Interconnections for Nova 1220

P4		BACKPANEL
NUMBER SIDE	LETTER SIDE A THRU AF GND	SLOT-SIDE-PIN No.
1		GND
2		9 A 92
3		9 A 91
4		9 A 78
5		9 A 77
6		9 A 76
7		9 A 75
8		9 A 73
9		9 A 71
10		9 A 69
11		9 A 67
12		9 A 65
13		9 A 63
14		9 A 61
15		9 A 59
16		9 A 57
17		9 A 47
18		9 A 49
19		9 A 79
20		9 A 81
21		9 A 84
22		9 A 83
23		9 A 86
24		9 A 85
25		9 A 88
26		9 A 87
27		9 A 89
28		9 A 90
29		9 B 6
30		9 B 11
31		9 B 13
32		9 B 15
33		9 B 19
34		9 B 23
35		9 B 25
36		9 B 27
37		9 B 31
38		9 B 34
39		9 B 36
40		9 B 38
41		9 B 40
42		9 B 48
43		9 B 49
44		9 B 51
45		9 B 52
46		9 B 53
47		9 B 54
48		9 B 67
49		9 B 69
50		RESERVED

Cabling The System

Turn all systems off, do not plug in any power cords, then:

1. install all internal cables not factory installed, following the instructions in the appropriate controller's manual.

2. install all device cables, remembering not to exceed the maximum length in each case. Be careful to protect each cable from wear and tear.

3. install the teletypewriter cable as shown in Figure I-4.

4. measure the line voltage of each service outlet, and check that it is correct for the computer.

5. measure the voltage between the ac return line and the frame ground at each outlet. THIS MUST BE ZERO

6. plug the power cord of each device into its service outlet.

REFERENCES:

Nova 1220 Rack Installation Print D-010-000014-01.

This Page Left Blank

Intentionally

SECTION N

MAINTAINING THE COMPUTER

INTRODUCTION

The Data General Corporation supports its equipment with a large field service organization, customer training programs and technical documentation. This section summarizes these services and includes tips on preventive maintenance, recommended tools and trouble shooting.

FIELD SERVICE ORGANIZATION

Field Service Programs

Data General's Field Service Organization currently offers its users a choice of three maintenance services. These services are subject to change without notice.

1. On Call Service Contract under which DGC will repair equipment at the installation when DGC is notified of a problem by the user. DGC also provides preventive maintenance on a regular schedule under this contract. Parts, labor and travel are included in the monthly payment schedule which is determined by the type and amount of equipment to be serviced and the distance between the installation and the nearest DGC service center.

2. Factory Service Contract under which DGC will:

   (1) repair equipment when it is returned to the DGC factory in Southboro, Mass. The user assumes full responsibility for freight and insurance charges to and from the plant. Parts and labor are included in the monthly payment schedule.

   (2) repair equipment at the installation when notified of a problem by the user. Parts are included in the monthly maintenance schedule, labor is charged at reduced rates and travel is charged at the prevailing standard rates.

3. Hourly Service under which parts, labor and travel are charged as needed at prevailing rates. No contract is signed for this service.

Field Service will also generate on request a complete spare parts list for any installation, and rent or sell replacement and loaner boards.

General Terms and Conditions (Subject to change without notice).

1. Equipment which is not under a DGC service contract or normal warranty is subject to an inspection by DGC Field Service before it is eligible for a service contract. All costs for this inspection are borne by the user.

2. The user must bear all maintenance costs incurred as a result of unauthorized changes to DGC equipment. These costs will be charged as Hourly Service, regardless of the type of service contract existing between DGC and the user.

3. No additional service charge will be added for new (add-on) equipment until the warranty period of that equipment has expired.

4. All services are offered between 9 a.m. and 5 p.m. Monday through Friday excluding DGC holidays.

5. The minimum contract period is 6 months.

6. Field Service price schedules are available on request from Data General Field Service, Southboro, Mass. 01772, Telephone 617-485-9100.

TRAINING ORGANIZATION

Data General's Training Organization currently offers its users four types of training courses. These courses are subject to change without notice.

Mainframe Maintenance Course. This course covers the logical structure of the central processor, memory, operator's console and power supply. Students must have experience with digital logic, integrated circuits and computer principles.

Fundamentals of Mini-Computer Programming. This course covers number systems, logic, flow charts and computer architecture. Students should have an aptitude for mathematics.

Basic Programming. This course covers Data General's assembly language utility software including loaders, editors, debuggers and assemblers. Students should have experience in programming.

Advanced Programming. This course covers Data General's Operating Systems, DOS, RTOS and SOS. Students must have experience in programming.

Courses are scheduled regularly in the training department at Southboro, Mass., and occasionally in field offices. Special courses can be arranged.

For more information call or write

Training Department
Data General Corporation
Southboro, Mass. 01772

Tel. 617-485-9100

PREVENTIVE MAINTENANCE

Periodically carry out the checks listed in Table, N-1, and remember the following points:

1. It is very poor practice to use the equipment as a counter top, particularly for liquids like coffee or soft drinks.
2. Always check the line voltage before plugging an expensive piece of equipment into an unknown socket. (see Section I).
3. Be careful not to get metal filings into the equipment; for example never let the equipment room be cleaned with steel wool.
4. Never clean the equipment with a vacuum cleaner that has a metal (conducting) nozzle.
5. Always be aware that too much heat, moisture or contaminants can do much to harm the equipment (see Section I).
6. Be very careful how cables are routed; they should never be strained, cramped or crushed (underfoot).

Table N-1

Preventive Maintenance Check List	
Item	Check
Mechanical Connections	1. that all screws are tight and that all mechanical assemblies are secure. 2. that all crimped lugs are secure and properly inserted onto their mating connectors.
Wiring and Cables	1. all wiring and cables for breaks, cuts, frayed leads, or missing lugs. 2. wire wraps for broken or missing pins. 3. that no wires or cables are strained or cramped. 4. that cables do not interfere with doors, and that they do not chafe when doors are opened and closed.
Air Filters	all air filters for cleanliness and for normal air movement through cabinets.
Modules and Components	1. that all modules are properly seated. Look for areas of discoloration on all exposed surfaces. 2. all exposed capacitors for signs of discoloration, leakage, or corrosion. 3. power supply capacitors for bulges.
Indicators and Switches	all indicators and switches for tightness; check for cracks, discoloration, or other visual defects.
Fans	for broken fan blades.
Diagnostics	Run all diagnostics periodically

Table N-2

Recommended Maintenance Tool Kit

ITEM	QTY	DESCRIPTION	MFG. & PART No.
1	1	6" combination slip joint pliers	Utica # 5-6
2	2	5 1/2" needle nose pliers	Utica # 654-5 1/2
3	1	4" needle nose pliers	Utica # 23-4
4	1	5" diagonal wire cutters	Utica # 44-5
5	1	4" diagonal wire cutters	Utica # 347-4 CFJS
6	1	5" ignition pliers	Utica # 517-5
7	1	Screwdriver kit including handle, 3/16", 1/4", 5/16" slotted #1, #2 phillips blades, each 4" long	Xcelite # 99 PV-6
8	1	3/32 slotter screwdriver with 2" blade	Xcelite # R3322
9	1	1/8" #0 phillips screwdriver	Xcelite # P12S
10	1	Magnetic pick up tool	Bonney # K26
11	1	3/32 through 3/8, 10 pc nut driver set	Xcelite # PS120
12	1	Xacto knife	
13	1	6" adjustable wrench	Utica # 91-6
14	1	Ignition wrench	Bonney # N24R
15	1	Set of 25 feeler gauges with 3" blades	Bonney # K53
16	1	Set of 15 hex keys	Bonney # N6R
17	1	Slotter 5" screw starter	Bonney # 5527
18	1	Phillips 6 1/4" screw starter	Bonney # 556
19	1	5" adjustable wire strippers	Utica # 110-5
20	1	Set of 4 cut needle files	Hunter # F228A
21	1	4 1/2" electrical tweezers	Hunter # B3M3
22	1	flash light	
23	1	Can Quick Freez (circuit cooler)	

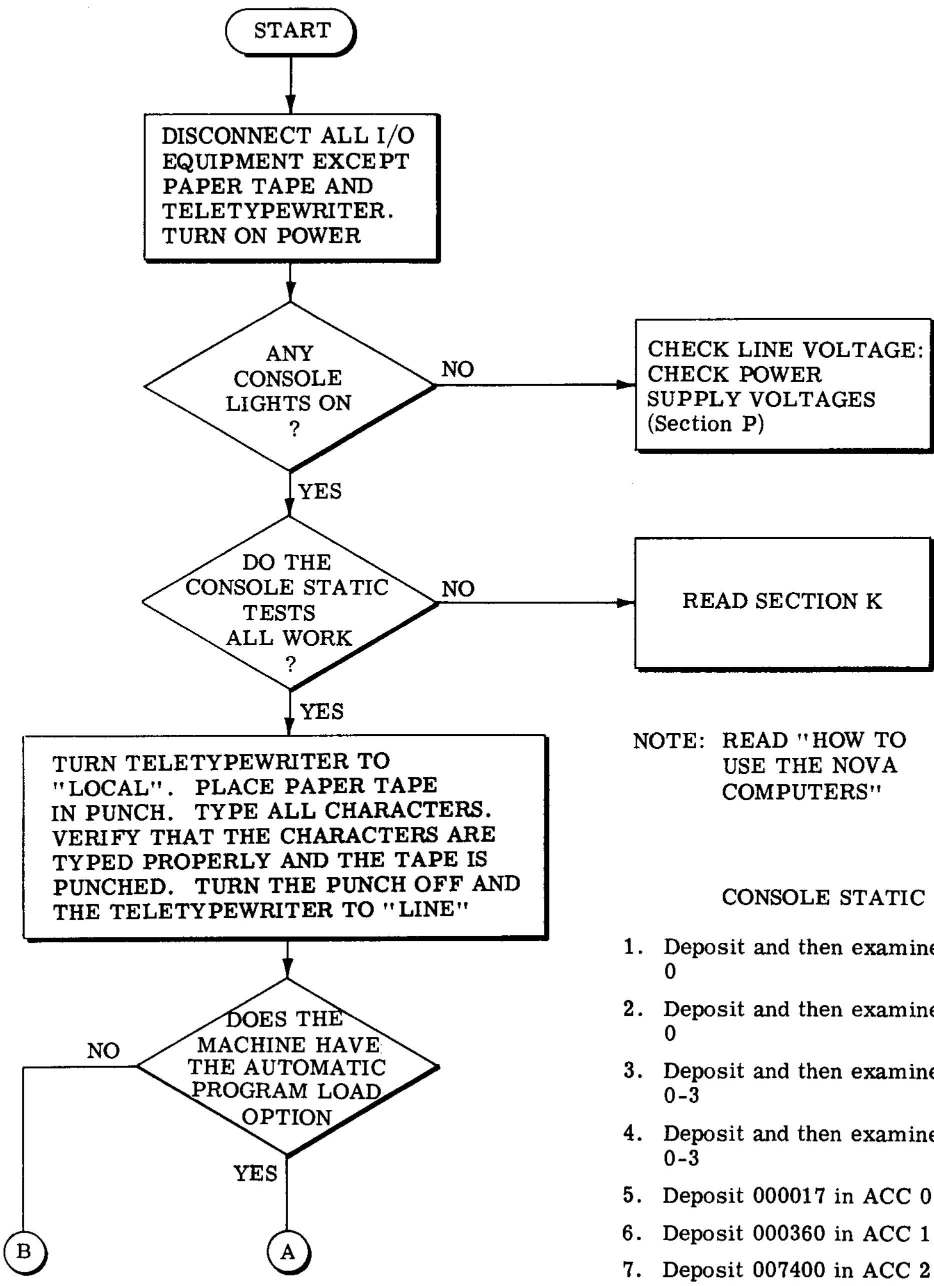
Table N-2 (Continued)

Recommended Maintenance Tool Kit			
ITEM	QTY	DESCRIPTION	MFG & PART No.
24	1	Can degreaser (flux remover)	
25	2	16P I/C test clip	
26	1	23 1/2 watt soldering iron with iron plated chisel tip	Ungar
27	1	47 1/2 watt soldering iron element	
28	1	1lb, 60/40 resin core solder	Kester
29	3	Spools of solder wick	
30	2	Acid brushes	
31	1	Vacuum solder removal tool	
32	1	Multimeter	Simpson # 260
33	1	Tool carrying case	
34	1	Oscilloscope	Tektronics # 453
35	1	Current probes	Tektronics # P60-22

Table N-3

The Nova 1220 Diagnostics			
Diagnostic	Part No.	Binary Tape No.	Description
Address Test	097-000007	095-000005	checks memory address selection logic
Checkerboard III	097-000014	095-000031	tests memory sense amplifiers and inhibit logic
Nova 1220 Logic Test	097-000017	095-000036	tests CPU logic other than I/O
Nova 1220 Instruction Timer	097-000019	095-000038	tests CPU clock logic and outputs time-to-complete for each instruction
Exerciser	097-000004	095-000012	tests CPU logic, teletypewriter, reader, punch and real-time clock;
Arithmetic Test	097-000018	095-000037	exercises arithmetic and logical instructions in CPU

HOW TO TEST THE COMPUTER

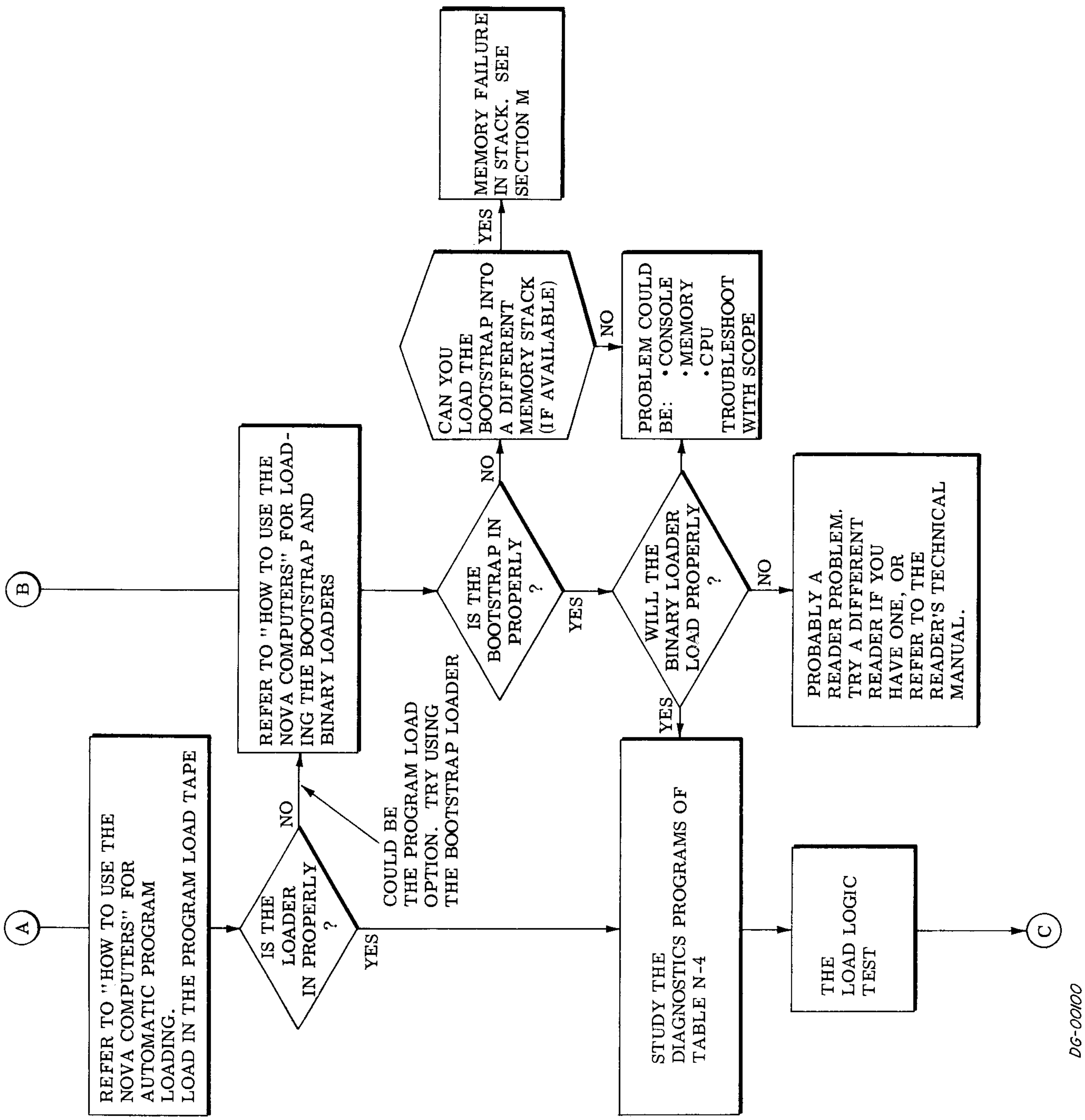


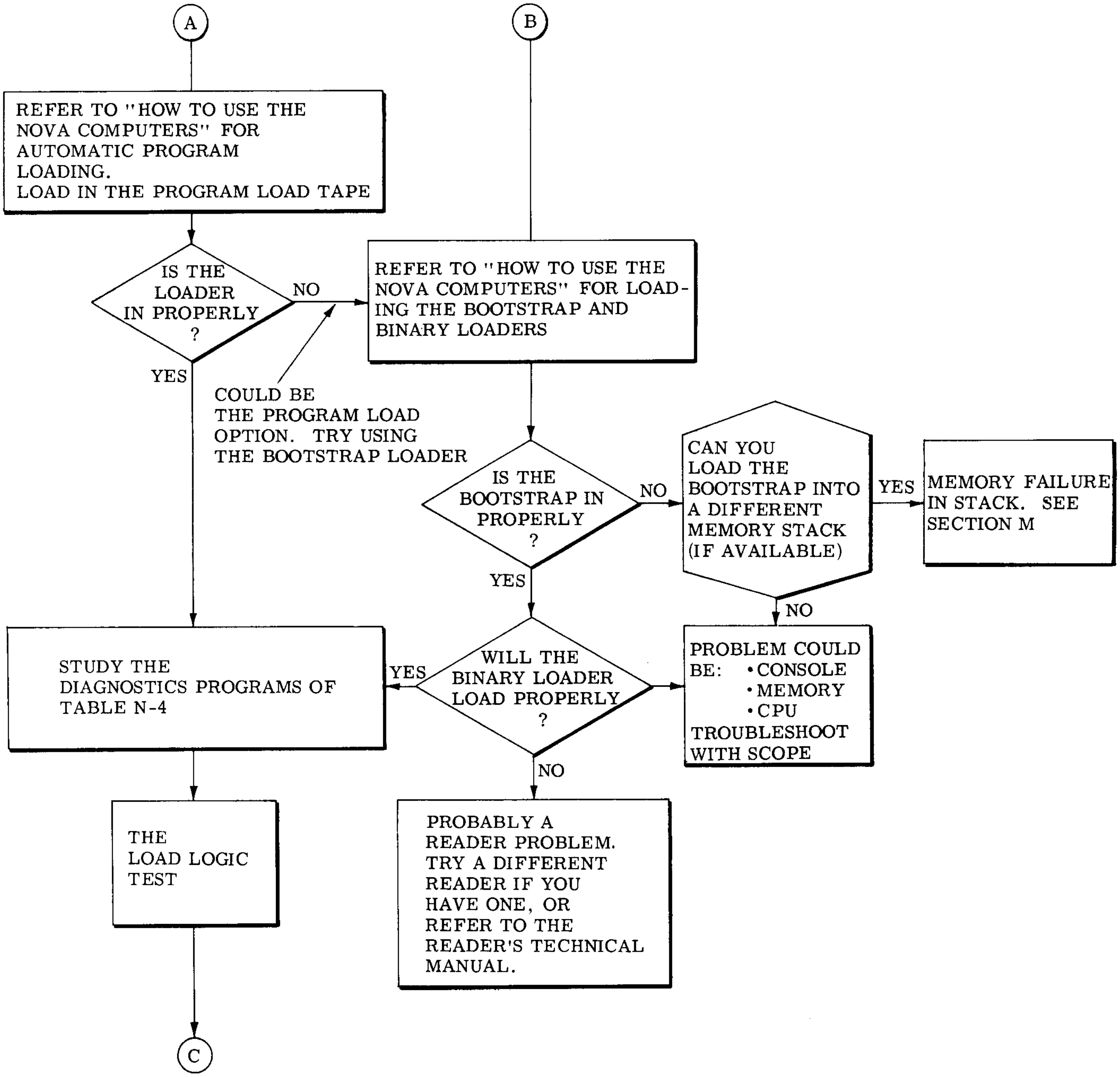
DG-00102

NOTE: READ "HOW TO USE THE NOVA COMPUTERS"

CONSOLE STATIC TESTS

1. Deposit and then examine 000000 in location 0
2. Deposit and then examine 111111 in location 0
3. Deposit and then examine 000000 in ACC 0-3
4. Deposit and then examine 111111 in ACC 0-3
5. Deposit 000017 in ACC 0 and verify
6. Deposit 000360 in ACC 1 and verify
7. Deposit 007400 in ACC 2 and verify
8. Deposit 170000 in ACC 3 and verify
9. Deposit 000000 in location 000000
10. Raise START/CONTINUE switch to START. RUN indicator should be lit
11. Lock the computer and try pushing the RESET/STOP switch. Computer should keep running.
12. Unlock the computer and hit STOP.
13. Depress DEPOSIT NEXT several times and check that the PC increments
14. Depress EXAMINE NEXT several times and check that the PC increments


A
REFER TO "HOW TO USE THE NOVA COMPUTERS" FOR AUTOMATIC PROGRAM LOADING. LOAD IN THE PROGRAM LOAD TAPE
IS THE LOADER IN PROPERLY ?
YES
NO
COULD BE THE PROGRAM LOAD OPTION. TRY USING THE BOOTSTRAP LOADER
STUDY THE DIAGNOSTICS PROGRAMS OF TABLE N-4
THE LOAD LOGIC TEST
C
B
REFER TO "HOW TO USE THE NOVA COMPUTERS" FOR LOADING THE BOOTSTRAP AND BINARY LOADERS
IS THE BOOTSTRAP IN PROPERLY ?
NO
YES
CAN YOU LOAD THE BOOTSTRAP INTO A DIFFERENT MEMORY STACK (IF AVAILABLE)
YES
MEMORY FAILURE IN STACK. SEE SECTION M
NO
PROBLEM COULD BE: •CONSOLE •MEMORY •CPU TROUBLESHOOT WITH SCOPE
WILL THE BINARY LOADER LOAD PROPERLY ?
YES
NO
PROBABLY A READER PROBLEM. TRY A DIFFERENT READER IF YOU HAVE ONE, OR REFER TO THE READER'S TECHNICAL MANUAL.
DG-00100


A
B
REFER TO "HOW TO USE THE NOVA COMPUTERS" FOR AUTOMATIC PROGRAM LOADING. LOAD IN THE PROGRAM LOAD TAPE
IS THE LOADER IN PROPERLY ?
NO
YES
REFER TO "HOW TO USE THE NOVA COMPUTERS" FOR LOADING THE BOOTSTRAP AND BINARY LOADERS
COULD BE THE PROGRAM LOAD OPTION. TRY USING THE BOOTSTRAP LOADER
IS THE BOOTSTRAP IN PROPERLY ?
NO
CAN YOU LOAD THE BOOTSTRAP INTO A DIFFERENT MEMORY STACK (IF AVAILABLE)
YES
MEMORY FAILURE IN STACK. SEE SECTION M
YES
NO
STUDY THE DIAGNOSTICS PROGRAMS OF TABLE N-4
YES
WILL THE BINARY LOADER LOAD PROPERLY ?
PROBLEM COULD BE: • CONSOLE • MEMORY • CPU TROUBLESHOOT WITH SCOPE
NO
THE LOAD LOGIC TEST
PROBABLY A READER PROBLEM. TRY A DIFFERENT READER IF YOU HAVE ONE, OR REFER TO THE READER'S TECHNICAL MANUAL.
C

DG-00100

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
ACB0	105	5	88-4	B4	ACB12	105	14	88-4	B3
ACB1	106	5	"	B3	ACB13	106	14	"	B2
ACB2	107	5	"	A4	ACB14	107	14	"	A4
					LOAD MBO*	98	6	88-3	A3
					KEYM SET*	101	9	88-1	B7
ACB3	108	5	"	A3	ACB15	108	14	88-4	A2
ACB4	105	7	"	B4	ACB0	105	3	"	B4
ACB5	106	7	"	B3	ACB1	106	3	"	B3
ACB6	107	7	"	A4	ACB2	107	3	"	A4
ACB7	108	7	"	A3	ACB3	108	3	"	A3
ACB8	105	9	"	B4	ACB4	105	2	"	B4
ACB9	106	9	"	B3	ACB5	106	2	"	B3
ACB10	107	9	"	A4	ACB6	107	2	"	A4
ACB11	108	9	"	A3	CRY SET	81	13	88-3	C6
					ACB7	108	2	88-4	A3
					SHIFTER Logic	114	10	"	A8
ACB12	105	11	"	B4	ACB12 SAVE	69	3	88-1	D5
					SHIFTER Logic	109	9	88-4	A8
ACB12*	105	12	"	B3	SHIFTER	125	19	"	A7
ACB13	106	11	"	B2					
ACB13*	106	12	"	"	SHIFTER	125	2	"	A7
					SHIFTER	125	20	"	A7
ACB14	107	11	"	A4					
ACB14*	107	12	"	A3	SHIFTER	125	1	"	A7
					SHIFTER	125	5	"	A6
					SHIFTER	125	18	"	A7
ACB15	108	11	"	A2					
ACB15*	108	12	"	A2	SHIFTER	125	3	"	A7
ACB12 SAVE	69	5	88-1	D4	SHIFTER Logic	90	1	88-4	A7
AC CLR	20	9	"	A6	IR(SH)	83	5	88-2	B8
					SHIFTER	125	7	88-4	A8
					LOAD AC*	111	3	88-3	D3
ACD0	123	5	88-4	B8	MULT	120	5	88-4	D5
					D BUFFR	122	3	"	C8
ACD1	123	7	"	B8	MULT	120	2	"	D5

*Indicates "Not"

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
					D BUFFER	122	2	88-4	C8
ACD2	123	9	88-4	B8	MULT	120	22	"	C5
					D BUFFER	122	15	"	C7
ACD3	123	11	"	B8	MULT	120	19	"	C5
					D BUFFER	122	14	"	C7
ACD3 SEL*	50	6	88-2	D4	ACD	123	1	"	B8
ACD4 SEL*	44	8	"	C4	ACD	123	15	"	B8
ACD OUT*	45	6	"	B3	D MULT(SEL)	121	1	"	C8
[ACS0]	124	5	88-4	B7	S BUFFER	115	3	"	C7
[ACS1]	124	7	"	B7	"	115	2	"	C7
[ACS2]	124	9	"	B6	"	115	15	"	C6
[ACS3]	124	11	"	B6	"	115	14	"	C6
ACS1 SEL*	49	6,8	88-2	C4	ACS	124	1	"	B7
ACS2 SEL*	49	3,11	"	B4	ACS	124	15	"	B7
ACTG0	54	5	88-1	D8	ACTG1	73	9	88-1	C8
					IR(SH) LOGIC	111	2	88-2	B8
					ACD	123	14	88-4	B8
					ACS	124	14	"	B7
ACTG1	54	7	88-1	D8	ACTG0	53	9	88-1	D8
					IR(SH) LOGIC	111	9	88-2	A8
					ACD	123		88-4	B8
					ACS	124	13	"	B7
ADDER0	117	13	88-4	D7	CRY SET*	81	3	88-3	C6
					ACB (DS)	105	4	88-4	B4
					ACB8	105	15	"	B4
					PC LOGIC	118	5,4	"	B6
					MULT	120	4	"	D5
ADDER1	117	11	88-4	D7	ACB(DS)	106	4	"	B3
					ACB9	106	15	"	B3
					PC LOGIC	118	1,2	"	B6
					MULT	120	1	"	D5
ADDER2	117	10	88-4	D7	ACB(DS)	107	4	"	A4
					ACB10	107	15	"	B2
					PC LOGIC	118	12,		
							13	"	A6
					MULT	120	23	"	C5
ADDER3	117	9	88-4	D7	ACB(DS)	108	4	"	A3
					ACB11	108	15	"	A2

Indicates "Not"

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
					PC LOGIC	118	9,		
							10	88-4	A6
					MULT	120	20	"	C5
ADD ONE*	88	8	88-2	D2	ADDER	117	7	"	D6
ADDER									
TEST	58	3	88-3	A4	LOOP SET*	104	5	88-3	D6
ALC	94	6	88-2	B7	DISABLE D				
					MULT	46	10	88-2	B3
					S0	47	1	"	C3
					TEST SKIP				
					SET	86	5	88-3	D8
ALC*	50	8	88-2	B7	ADD ONE*	44	2	88-2	D3
					AND	65	5	"	B7
					E SET	74	1	"	C7
					S2	91	12	"	C3
					ALC	94	5	"	B7
					S BUFFER				
					(SH)	115	13	"	C7
ALC·$\overline{SKIP}$	83	10	88-3	D8	LOAD CRY*	97	13	88-3	C5
AND	65	6	88-2	B7	CRY ENAB	91	2	"	C6
					S1	91	5	88-2	C3
					ADDER	117	8	88-4	D8
AND ENAB*	64	11	88-2	B7	IO DCDR	62	13	88-1	A5
					AND	65	4	88-2	B7
					PACK	89	2	88-3	C5
CARRY									
(F/F)	76	8	88-3	C5	CRY ENAB	77	4	88-3	C7
CARRY*									
(F/F)	76	9	"	C5	CON IND				
					(A15, P49)	6	5	89-1	C8
					CRY ENAB	77	3	88-3	C7
CLK FLOP	20	5	88-1	A6	MA LOAD*	56	10	88-1	D3
					CPU CLK	72	2,		
							12	"	A7
					MEM CLK	73	3	"	A7
					LOAD AC*	93	5	88-3	D3
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
CLK FLOP*	20	6	88-1	A7	CLK FLOP	20	2	88-1	A7
[CLR*]	63	5	"	A4	CLR	7	1	"	A4
CLR	7	2	"	A4	(IO CLR PLS)	(A50)		90-1	
CLR ION*	63	11	"	B4	ION	84	4	88-2	C7
CLR SKIP*	99	8	88-3	B3	SKIP	79	13	88-3	B5
					LOAD MBO*	98	10	"	B3
[CON0*](S11)	6	4	89-1	C8	MEM0*	(B71)	(39)	89-1	C8
					(CON IND)	7	9	"	C8
[CON1*](S12)	6	2	89-1	C7	MEM1*	(B70)	(P41)	"	C7
					(CON IND)	7	13	"	C7
[CON2*](S13)	6	8	89-1	C7	MEM2*	(B47)	(P13)	"	C7
					(CON IND)	7	3	"	C7
[CON3*](S14)	6	12	89-1	C7	MEM3*	(B68)	(P43)	"	C7
					(CON IND)	7	1	"	C7
[CON4*](S15)	3	8	89-1	C6	MEM4*	(B28)	(P37)	"	C6
					(CON IND)	8	13	"	C6
[CON5*](S16)	3	10	89-1	C6	MEM5*	(B26)	(P36)	"	C6
					(CON IND)	8	3	"	C6
[CON6*](S17)	3	6	89-1	C6	MEM6*	(B22)	(P10)	"	C6
					(CON IND)	8	1	"	C6
[CON7*](S18)	3	4	89-1	C5	MEM7*	(B24)	(P42)	"	C5
					(CON IND)	9	13	"	C5
[CON8*](S19)	3	2	89-1	C5	MEM8*	(A55)	(P34)	"	C5
					(CON IND)	9	3	"	C5
[CON9*](S20)	3	12	89-1	C5	MEM9*	(A53)	(P7)	"	C5
					(CON IND)	9	1	"	C5
[CON10*](S21)	4	8	89-1	C4	MEM10*	(A45)	(P32)	"	C4
					(CON IND)	10	13	"	C4
[CON11*](S22)	4	10	89-1	C4	MEM11*	(A51)	(P31)	"	C4
					(CON IND)	10	3	"	C4
[CON12*](S23)	4	12	89-1	C3	MEM12*	(A36)	(P5)	"	C3
					(CON IND)	10	1	"	C3
[CON13*](S24)	4	6	89-1	C3	MEM13*	(A35)	(P29)	"	C3
					(CON IND)	11	13	"	C3
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
[CON14*](S25)	4	4	89-1	C3	MEM14*	(B76)	(P3)	89-1	C3
					(CON IND)	11	3	"	C3
[CON15*](S26)	4	2	89-1	C2	MEM15*	(B18)	(P2)	"	C2
					(CON IND)	12	13	"	C2
CON DATA*	4	8	88-1	A2		(A28)	(P46)	"	
					[CON0*](S11)	6	3	"	C8
					[CON1*](S12)	6	1	"	C7
					[CON2*](S13)	6	9	"	C7
					[CON3*](S14)	6	13	"	C7
					[CON4*](S15)	3	9	"	C6
					[CON5*](S16)	3	11	"	C6
					[CON6*](S17)	3	5	"	C6
					[CON7*](S18)	3	3	"	C5
					[CON8*](S19)	3	1	"	C5
					[CON9*](S20)	3	13	"	C5
					[CON10*]				C4
					(S21)	4	9	"	C4
					[CON11*]				
					(S22)	4	11	"	C4
					[CON12*]				
					(S23)	4	13	"	C3
					[CON13*]				
					(S24)	4	5	"	C3
					[CON14*]				
					(S25)	4	3	"	
					[CON15*]				C2
					(S26)	4	1	"	
CON INST*	36	8	88-1	A2		(A22)	(P22)	"	
					[CON INST]	5	9	"	A8
[CON INST]	5	8	89-1		MEM0*	1	2	"	C8
					MEM1*	1	4	"	C7
					MEM2*	2	10	"	C7
					MEM3*	1	12	"	C7
					MEM4*	1	10	"	C6
					MEM5*	2	12	"	C6
					MEM6*	2	2	"	C6
					MEM7*	2	4	"	C5
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
CON RQ*	5	6	89-1	C8					
	(A27)	(P21)			KEY SEEN	3	4	88-1	B8
CONT+ISTP+ MSTP*	(A20)		89-1	B3	KEY ENAB*	3	2	88-1	B8
CPU CLK	72	6,8	88-1	A6	MB LOAD	14	4	"	C2
					IR4-IR7	28	6	88-2	A6
					MBC	32	6	"	A4
					MBC	33	6	"	A5
					MBO	37	6	88-4	C4
					MBO	38	6	"	C3
					MBO	39	6	"	D3
					MBO	40	6	"	D4
						42	6	88-1	C8
					LOAD PC*	57	10	88-3	B3
					MA LOAD*	60	10	88-1	D2
					INPUT	66	13	"	C5
					PTG	69	6	"	D4
					SKIP	78	13	88-3	B5
					MAJOR STATES	95	6	88-2	D6
					CARRY F/F Logic	97	9	88-3	C5
						102	6	"	D8
					LOOP/PACK /EFA	103	6	"	D5
					ACB	105	6	88-4	B4
					ACB	106	6	"	B3
					ACB	107	6	"	A4
					ACB	108	6	"	A3
					END CYCLE F/F	113	13	88-1	D5
CPU INST	6	11	88-2	B7	INTA	6	5	"	B5
					IORST	6	10	"	A4
					(SKIP Logic)	11	2	88-3	B7
					"	11	12	"	B7
					CON DATA* (Reads)	24	4	88-1	A3
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
					(IO OCDR)	64	2	88-1	B5
					HALT*	71	2	88-2	C8
					PACK Logic	87	4	88-3	C6
CPU INST*	10	6	88-2	B7	MSKO*	4	13	88-1	A4
					CPU INST	6	12,		
							13	88-2	B7
CRY ENAB	80	11	88-3	C6	CRY SET*	81	4	88-3	C6
					CRY ENAB				
					SAVE	102	15	"	D7
CRY ENAB									
SAVE	102	9	88-3	D7	SHIFT Logic	90	10	88-4	A7
					" "	114	13	"	A8
CRY OUT*	117	16	88-4	D8	SERIAL CRY	54	14	88-1	D7
					CRY ENAB	91	1	88-3	C6
CRY SET*	81	8	88-3	C5	CRY SET				
					SAVE	42	15	88-1	C7
					CARRY F/F	76	12	88-3	C5
CRY SET SAVE*	42	9	88-1	C7	(SKIP Logic)	77	9	"	B7
DATA0*	16	11	103-1	C	Terminator			88-3	C8
	17	1	"						
	(B62)								
DATA1*	16	8	103-1	C	Terminator			88-3	C8
	17	3	"						
	(B65)								
DATA2*	14	11	103-1	C	Terminator			88-3	C8
	15	1	"						
	(B82)								
DATA3*	14	8	103-1	C	Terminator			88-3	C8
	15	3	"						
	(B73)								
DATA4*	12	11	103-1	C	Terminator			88-3	C8
	13	1	"						
	(B61)								
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
DATA5*	12	8	103-1	C	Terminator			88-3	C8
	13	3	"						
	(B57)								
DATA6*	10	11	103-1	C	Terminator			88-3	C8
	11	1	"						
	(B95)								
DATA7*	10	8	"	C	Terminator			88-3	C8
	11	3	"						
	(B55)								
DATA8*	8	11	103-1	C	Terminator			88-3	C8
	9	1	"						
	(B60)								
DATA9*	8	8	103-1	C	Terminator			88-3	B8
	9	3	"						
	(B63)								
DATA10*	6	11	103-1	C	Terminator			88-3	B8
	7	1	"						
	(B75)								
DATA11*	6	8	103-1	C	Terminator			88-3	B8
	7	3	"						
	(B58)								
DATA12*	4	11	103-1	C	Terminator			88- 3	B8
	5	1	"						
	(B59)								
DATA13*	4	8	103-1	C	Terminator			88-3	B8
	5	3	"						
	(B64)								
DATA14*	2	11	103-1	C	Terminator			88-3	B8
	3	1	"						
	(B56)								
DATA15*	2	8	103-1	C	Terminator			88-3	B8
	3	3	"						
	(B66)								
[DATOA*]	25	6	88-1	B4	DATOA	7	9	88-1	B4
DATOA	7	8	"	B4		(A58)		90-1	
DATOB*	25	5	"	B4	DATOB	7	13	88-1	B4
					MSKO*	4	12	"	B4
DATOB	7	12	88-1	B4		(A56)		90-1	
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
[DATOC*]	25	4	88-1	B4	DATOC	26	5	88-1	B4
DATOC	26	6	"	B4		(A48)		90-1	
[DATIA*]	25	9	"	B4	DATIA	5	13	88-1	B4
DATIA	5	12	"	B4	CON DATA*	24	5	"	A3
						(A44)			
[DATIB*]	25	10	"	B4	DATIB	5	11	88-1	B4
DATIB	5	10	"	B4	INTA	6	4	"	A4
						(A42)			
[DATIC*]	25	11	"	B4	DATIC	7	5	88-1	B4
DATIC	7	6	"	B4	IORST	6	9	"	A4
						(A54)			
[D BUFFR0]	122	5	88-4	C8	[D MULT0]	121	2	88-4	C8
[D BUFF1]	122	7	"	C8	[D MULT1]	121	5	"	C8
[D BUFFR2]	122	9	"	C8	[D MULT2]	121	14	"	C8
[D BUFFR3]	122	11	"	C8	[D MULT3]	121	11		C8
DCH	23	9	88-1	C6	DCHI	14	9	88-1	C2
					DCH LOOP				
					ENAB	15	2	88-1	B3
					ADD ONE*	41	2	88-2	D4
DCHA	69	7	88-1	D4	DCHA*	7	11	88-1	C2
					DRIVE IO*	13	5	"	B3
					DCH	23	15	"	C6
DCHA*	7	10	88-2	C2		(A60)		90-1	
DCHA SET*	71	8	88-1	C4	[DCHA SET]	67	3	88-1	C4
					FETCH	97	1	88-2	D7
[DCHA SET]	67	4	88-1	C4	DCHA	69	2	88-1	C4
DCHI	14	8	"	C2		(B37)		90-1	
					DRIVE IO*	13	4	88-1	B3
DCH LOOP									
ENAB	15	6	88-1	B2	OVFLO	15	9	88-1	B2
					DCHO	18	12,		
							13	88-1	B2
					ACTG(LD)	75	10	"	D8
					LOOP SET*	104	10,		
							13	88-3	C6
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
DCHM0*	(B17)		88-1	B3	DCH LOOP ENAB	15	4, 5	88-1	B3
					[DCHM0]	16	1	"	B3
[DCHM0]	16	2	88-1	B3	DCHI	14	10	"	B2
DCHM1*	(B21)		"	B3	"	14	12	"	B2
					[DCHM1]	16	3	"	B3
					LOOP SET*	34	12	88-3	C6
[DCHM1]	16	4	88-1	B3	OVFLO	15	10	88-1	B2
DCHO	18	8	"	B2		(B33)		90-1	
DCHR*	(B35)		"	C5	DCHR PEND	13	2	88-1	C5
DCHR PEND	13	3	"	C5	DCHA SET*	71	10	"	C5
					LOOP SET*	104	9	88-3	D6
DEFER	95	7	88-2	D6	DEFER AGAIN	76	4	88-2	C7
					ADD ONE*	90	4	"	D4
					DEFER*	94	11	"	D6
					LOOP SET*	104	6	88-3	D6
DEFER*	94	10	88-2	D6	(CON IND)	(A12)	(P52)	89-1	C2
					S0	48	1	88-2	C4
					ADDER TEST	58	12	88-3	A6
					ADDER TEST	59	10	"	A6
					FETCH + DEFER	75	2	88-2	C7
DEFER AGAIN*	76	5	88-2	C7	D SET	74	9	"	C7
(D+ E SET) + $\overline{TS3}$	36	11	"	D5	DCHR PEND	13	1	88-1	C5
					(RUN LOGIC)	24	13	"	B7
					PC IN*	35	1	88-2	D5
D+ E SET*	96	11	88-2	D7	(D+ E SET)+ $\overline{TS3}$	36	13	88-2	D5
					(RUN LOGIC)	43	13	88-1	B7
					FETCH LOGIC	97	5	88-2	C7

*Indicates "Not"

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
Disable D Mult	53	3		B2	D Mult (Enab)	121	15	88-4	C8
DIV*	(A91)			C5	Carry F/F	76	10	88-3	C5
[D MULT0]	121	4	88-4	C7	ADDER	117	19	88-4	D7
[D MULT1]	121	7	"	C7	"	117	21	"	D7
[D MULT2]	121	12	"	C7	"	117	23	"	D7
[D MULT3]	121	9	"	C7	"	117	2	"	D7
DRIVE IO*	12	8	88-1	B2		(B88)		90-1	
					READ IO*	12	4, 5	88-1	B2
					[DRIVE IO]	18	1	103-1	C8
[DRIVE IO]	18	2	103-1	C8	[Drive IO· Select]	26	9, 10, 12	103-1	C8
[DRIVE IO· Select]	26	8	103-1	C8	DATA0*	16	12	"	C
					DATA1*	16	10	"	C
					DATA2*	14	12	"	C
					DATA3*	14	10	"	C
					DATA4*	12	12	"	C
					DATA5*	12	10	"	C
					DATA6*	10	12	"	C
					DATA7*	10	10	"	C
					DATA8*	8	12	"	C
					DATA9*	8	10	"	C
					DATA10*	6	12	"	C
					DATA11*	6	10	"	C
					DATA12*	4	12	"	C
					DATA13*	4	10	"	C
					DATA14*	2	12	"	C
					DATA15*	2	10	"	C
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
DS0*	8	8	88-1	C4		(A72)		90-1	
DS1*	8	10	"	C4		(A68)		"	
DS2*	8	12	"	C4		(A66)		"	
DS3*	22	8	"	C4		(A46)		"	
DS4*	8	4	"	C4		(A62)		"	
DS5*	8	2	"	C4		(A64)		"	
D SET	74	8	88-2	C6	DEFER	95	2	88-2	C6
					E SET	96	2	"	C6
					D+E SET*	96	13	"	D7
DSZ·E·TS0*	52	4	88-2	B4	S0	92	1	"	C3
EFA	103	11	88-3	D5	MBC(SH)	32	13	88-2	A5
					MBC(SH)	33	13	"	A4
					ACD4 SEL*	44	9	"	C5
					ACD OUT*	45	10	"	B3
					Disable D Mult	46	4	"	B3
					S0	47	3	"	C3
					S0	47	4	"	B3
					D SET	74	4	"	C7
					JSR·EFA	93	13	"	C8
EFA*	103	12	88-3	D5	EFA·$\overline{PTG1}$	34	5	"	A3
					ACD4 SEL*	44	1	"	C5
					ACD3 SEL*	50	3	"	C5
EFA·$\overline{PTG1}$	34	6	88-2	A2	MBC (DS)	32	4	"	A4
					S Mult (SEL)	116	1	88-4	C7
End Cycle (F/F)	113	1	88-1	D5	ACTG	53	10, 12	88-1	C8
					End Cycle(F/F)	113	2	"	C5
					LOAD CRY*	97	12	88-3	C5
					(LD) Test Skip	102	10	"	D8
					(LD) Loop/ Pack	103	10	"	D5
					Shifter Logic	109	13	88-4	A8
					" "	114	1	"	A8
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
End Cycle*(F/F)	113	6	88-1	C5	Shifter Logic	114	9	88-4	A8
					PTG0·TS0 Logic	112		88-1	A6
E SET	96	3	88-2	C6	EXEC	95	15	88-2	D6
					D+E SET*	96	12	"	D7
EXEC	95	9	88-2	D6	EXEC*	73	11	"	D6
					(INST DCDR)	92	9	"	B5
EXEC*	73	10	"	D6	(CON IND)	(A11)	(P51)	89-1	C1
					(INST DCDR)	52	15	88-2	B5
EXT LOAD*	(A47)			A3	LOAD AC*	111	4	88-3	D3
	(B49)			A8	Shifter (Enab)	125	8, 9	88-4	A8
EXT Select*	(B80)				SELECT	35	9, 10	103-1	
FETCH	95	5	88-2	D6	MB LOAD	13	13	88-1	C3
					LOAD IR	34	9	88-2	A7
					LOAD PC*	61	10	88-3	B4
					FETCH·TS0*	64	9	88-2	D5
					ALC*	50	9	"	B8
					ION	85	1	"	C6
					FETCH*	94	13	"	D6
					CLR SKIP*	100	4	88-3	B4
FETCH*	94	12	88-2	D6	(CON IND)	(A13)	(P50)	89-1	C2
					ACD OUT*	45	1, 13	88-2	B4
					FETCH+ DEFER	75	1	"	C7
					ADD ONE*	89	12	"	D3
Fetch+Defer	75	3	88-2	C7	IR0+SKP	50	1	"	B6
					E SET	74	13	"	C7
FETCH·TS0*	64	8	88-2	D4	EFA	85	12	"	C5
					Mult (SEL)	120	16	88-4	C5
Force Load IR*	(A85)		88-2	A8	IR(LD)	12	4	88-2	A8
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
HALT*	71	6	88-2	C7	MB LOAD	14	2	88-1	C2
					(RUN LOGIC)	62	3	"	B7
					DCHA	71	9	"	C5
INH0	34	9	103-1	B	MEM0*	16	1	103-1	C
					DATA0*	16	13	"	C
INH0*	34	8	103-1	B	(INHB0) (Q15)	68	12	103-2	7
INH1	34	5	"	B	MEM1*	16	5	103-1	C
					DATA1*	16	9	"	C
INH1*	34	6	103-1	B	(INHB1) (Q16)	68	2	103-2	7
INH2	32	5	"	B	MEM2*	14	1	103-1	C
					DATA2*	14	13	"	C
INH2*	32	6	103-1	B	(INHB2) (Q13)	64	2	103-2	7
INH3	32	9	"	B	MEM3*	14	5	103-1	C
					DATA3*	14	9	"	C
INH3*	32	8	103-1	B	INHB3) (Q14)	64	12	103-2	7
INH4	31	9	"	B	MEM4*	12	1	103-1	C
					DATA4*	12	13	"	C
INH4*	31	8	"	B	(INHB4) (Q11)	58	12	103-2	7
INH5	31	5	"	B	MEM5*	12	5	103-1	C
					DATA5*	12	9	"	C
INH5*	31	6	103-1	B	(INHB5) (Q12)	58	2	103-2	7
INH6	28	5	"	B	MEM6*	10	1	103-1	C
					DATA6*	10	13	"	C
INH6*	28	6	103-1	B	(INHB6) (Q9)	55	2	103-2	7
INH7	28	9	"	B	MEM7*	10	5	103-1	C
					DATA7*	10	9	"	C
INH7*	28	8	103-1	B	(INHB7) (Q10)	55	12	103-2	7
INH8	27	9	"	B	MEM8*	8	1	103-1	C
					DATA8*	8	13	"	C
INH8*	27	8	103-1	B	(INHB8) (Q7)	48	12	103-2	4
INH9	27	5	"	B	MEM9*	8	5	103-1	C
					DATA9*	8	9	"	C
INH9*	27	6	103-1	B	(INHB9) (Q8)	48	2	103-2	4
INH10	24	5	"	B	MEM10*	6	1	103-1	C
					DATA10*	6	13	"	C
INH10*	24	6	103-1	B	(INHB10)(Q5)	45	2	103-2	4
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
INH11	24	9	103-1	B	MEM11*	6	5	103-1	C
					DATA11*	6	9	"	C
INH11*	24	8	103-1	B	(INHB11) (Q6)	45	12	103-2	5
INH12	23	9	"	B	MEM12*	4	1	103-1	C
					DATA12*	4	13	"	C
INH12*	23	8	"	B	(INHB12) (Q3)	39	12	103-2	5
INH13	23	5	"	B	MEM13*	4	5	103-1	C
					DATA13*	4	9	"	C
INH13*	23	6	103-1	B	(INHB13) (Q4)	39	2	103-2	5
INH14	21	5	"	B	MEM14*	2	1	103-1	C
					DATA14*	2	13	"	C
INH14*	21	6	103-1	B	(INHB14) (Q1)	37	2	103-2	5
INH15	21	9	103-1	B	MEM15*	2	5	103-1	C
					DATA15*	2	9	"	C
INH15*	21	8	103-1	B	(INHB15) (Q2)	37	12	103-2	5
INHB0	70	3	103-2	7	Q15				7
INHB1	70	5	"	7	Q16				7
INHB2	63	3	"	7	Q13				7
INHB3	63	5	"	7	Q14				7
INHB4	61	3	"	7	Q11				7
INHB5	61	5	"	7	Q12				7
INHB6	53	3	"	7	Q9				7
INHB7	53	5	"	7	Q10				7
INHB8	51	3	"	4	Q7				7
INHB9	51	5	"	4	Q8				4
INHB10	43	3	"	4	Q5				4
INHB11	43	5	"	4	Q6				4
INHB12	42	3	"	4	Q3				4
INHB13	42	5	"	4	Q4				4
INHB14	20	3	"	4	Q1				4
INHB15	20	5	"	4	Q2				4
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
INH GATE B	26	6	103-1	D2	(INHB8) (Q7)	48	13	103-2	4
					(INHB9) (Q8)	48	1	"	4
					(INHB10) (Q5)	45	1	"	4
					(INHB11) (Q6)	45	13	"	4
					(INHB12) (Q3)	39	13	"	4
					(INHB13) (Q4)	39	1	"	4
					(INHB14) (Q1)	37	1	"	4
					(INHB15) (Q2)	37	13	"	4
INHIBIT	13	8	88-1	C2		(B30)		103-1	D3
					INH GATE A, B	41	9	103-1	D3
					WRITE MEM	41	2	"	D3
INHIBIT SELECT*	(B85)		103-1	D8	SELECT	35	5	103-1	D8
INPUT*(F/F)	66	8	88-1	B5	DRIVE IO*	12	10	88-1	B3
					(IO INST DCDR)	25	15	"	B4
					MB LOAD	112	1,9	"	C3
[INTA*]	6	6	88-1	A4	INTA	5	9	"	A4
INTA	5	8	"	A4		(A40)		90-1	
INTR*	(B29)				PI SET	75	12	88-2	C7
INH TRANS*	56	6	88-1	B2		(B45)		90-1	
					[$\overline{\text{INH TRANS}\cdot}$ SEL]	36	2,5,4	103-1	C8
[$\overline{\text{INH TRANS}\cdot}$ SEL]	36	6	103-1	C8	MEM0*	16	2	103-1	C
					MEM1*	16	4	"	C
					MEM2*	14	2	"	C
					MEM3*	14	4	"	C
					MEM4*	12	2	"	C
					MEM5*	12	4	"	C
					MEM6*	10	2	"	C
					MEM7*	10	4	"	C
					MEM8*	8	2	"	C
					MEM9*	8	4	"	C
					MEM10*	6	2	"	C
					MEM11*	6	4	"	C
					MEM12*	4	2	"	C

*Indicates "Not"

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
					MEM13*	4	4	103-1	C
					MEM14*	2	2	"	C
					MEM15*	2	4	"	C
IO·E	42	5	88-1	C8	IO·E*	94	3	88-1	C7
					(IO Inst DCDR)	64	4	"	D5
					(IO DCDR)	62	2	"	A5
					HALT*	71	1	88-2	C8
					LOOP SET*	86	9	88-3	C7
					(Pack Logic)	89	4	88-3	C5
IO·E*	94	4	88-1	C7	MA LOAD*	60	13	88-1	D2
[IO(F+D*)]	51	12	88-2	B6	IO(F+D)	27	5	88-2	B6
IO(F+D)	27	6	"	B5	INPUT F/F				
					Logic	9	12	88-1	C5
					IO·E	42	3	88-1	C8
ION	82	6	"	C7	(SKIP Logic)	11	13	88-3	B7
					ION*	84	5	88-2	C7
ION*	84	6	88-2	C7	(CON IND)	(A16)	(P26)	89-1	D2
					ION	82	5	88-2	C7
					(ION LOGIC)	85	5	"	C7
[IO PLS*]	63	4	88-1	A7	IO PLS	26	3	88-1	A4
IO PLS	26	4	"	A4		(A74)		90-1	
IORST	10	8	"	A4		(A70)		90-1	
IO SKIP*	25	12	"	B4	IO SKIP	26	1	88-1	B4
					SKIP INC*	87	1	"	B8
IO SKIP	26	2	88-1	B4	(Skip Logic)	59	5	88-3	B6
IR0*	28	5	88-2	A6	(RUN LOGIC)	43	10	88-1	B7
					ACD OUT*	45	3	88-2	B3
					SH/SWP DCDR	50	13	88-2	B6
					"	51	1	"	B6
					PC ENAB*	53	4, 5	88-3	B4
					(Pack Logic)	92	4	88-3	C6
IR5·IR6	65	3	88-2	B8	AND ENAB*	64	12	88-2	B8
					HALT*	71	4	"	D8
IR0+SKIP	50	12	88-2	B6	ALC*	50	11	"	B8
					(SH/SWP				
					DCDR)	51	15	"	D6
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
ISTP*	(A17)	(P24	89-1	B8	(RUN LOGIC)	24	9	88-1	B7
(ISZ+DSZ)E	84	8	88-3	D6	CRY SET*	81	9, 10	88-3	C6
					LOOP SET*	104	2	"	D6
(ISZ+DSZ)E*	52	9	88-2	B4	(INST DCDR)	52	1	88-2	B5
					(ISZ+DSZ)E	84	13	88-3	D6
					Test Skip Set	86	1	"	D8
ISZ·E·TS0*	52	5	88-2	B4	ADD ONE*	89	9	88-2	D3
(JMP+JSE) (F+D)	48	11	"	B5	PC ENAB*	61	3	88-3	B4
					JSR·EFA*	93	2	88-2	C7
JSR.EFA	92	11	88-3	C3	SHIFT ACB	100	1	88-3	C3
					WAS JSR	103	3	"	D5
JSR·EFA*	93	12	88-2	C7	JSR·EFA	92	13	"	C3
					(Pack Logic)	99	2	"	C5
KEY	23	5	88-1	C6	KEY*	6	1	88-1	C7
					LOAD IR	34	10	88-2	A7
					CON INST*	36	9	88-1	A2
					(RUN LOGIC)	43	9	"	B7
					KEYM SET*	55	5	"	B6
					Disable D Mult	46	1	88-2	B3
					LOAD PC*	61	5	88-3	B4
KEY*	6	3	88-1	C6	KEY·LOOP	4	2	88-1	C6
					(DS)	23	4	"	C6
					ADD ONE*	44	5	88-2	D3
					INH TRANS*	56	4	88-1	B2
					MA LOAD*	56	9	"	D3
					(Pack Logic)	70	13	88-3	C6
					LOOP SET*	84	10	"	C6
					CLR SKIP*	99	10	"	B3
KEY ENAB*	3	3	88-1	B8	PRESET*	3	12	88-1	B7
					$\overline{KEY}$	6	2	"	C7
KEY·LOOP	4	3	88-1	C6	CON DATA*	4	10	"	A2
					ACD OUT*	45	2	88-2	A3
					LOAD MBO*	98	13	88-3	B3
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
KEYM	23	11	88-1	C6	CON DATA*	24	3	88-1	A3
					(RUN LOGIC)	43	2	"	B7
					ADD ONE*	41	1	88-2	D4
KEYM*	23	12	88-1	C6					
KEYM· PL	41	8	88-1	C5	KEYM SET*	55	3	88-1	B6
					KEYM· PL· TS0*	57	2	88-3	C4
					JSR· EFA	93	9	"	C4
					LOAD MBO*	98	5	"	A3
KEYM· PL· TS0*	57	3	88-3	C3	INH TRANS*	56	5	88-1	B2
					LOAD PC*	57	4	88-3	B3
KEYM SET*	55	6	88-1	B6	[KEY M SET]	22	1	88-1	B6
					FETCH	97	2	88-2	D7
[KEYM SET]	22	2	88-1	B6	KEYM	23	14	88-1	A6
KEY SEEN* (F/F)	2	6	"	B8	(RUN LOGIC)	21	1	"	B6
					(MR)	54	1	"	D8
					(MR)	102	1	88-3	D8
KEY SEEN (F/F)	2	5	88-1	B8	KEY ENAB*	3	1	88-1	D8
					(SH)	23	13	"	C6
LDA· E*	52	10	88-2	B4	(Pack Logic)	99	1	88-3	D5
LOAD AC*	93	6	88-3	D2		(A77)		90-1	
					ACD	123	3	88-4	B8
					ACS	124	3	"	B7
LOAD ACB	100	11	88-3	C3	SHIFT ACB	100	2	88-3	C3
					ACB(LD)	105	10	88-4	B4
					ACB(LD)	107	10	"	B4
					ACB(LD)	108	10	"	B4
LOAD CRY*	97	8	88-3	C5	CARRY	76	11	88-3	C5
					(Pack Logic)	99	5	"	C5
LOAD IR	34	3	88-2	A6		(A73)		90-1	
					IR(LD)	28	10	88-2	A6
					IR(LD) Logic	8	5	"	A8
					MBC(LD)	32	10	"	A4
					MBC(LD)	33	10	"	A5
					[STUTTER]	54	15	88-1	D7
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
LOAD MBO*	98	8	88-3	A2	MBO(SH)	37	13	88-4	C4
					MBO(SH)	38	13	"	C4
					MBO(SH)	39	13	"	C4
					MBO(SH)	40	13	"	C4
LOAD PC*	57	8	88-3	A2	PC	119	12	"	A6
LOOP	103	7	"	D5	MB LOAD	13	12	88-1	C3
					LOOP*	22	5	88-3	D5
					S0	47	9	88-2	C3
					(IO Inst DCDR)	64	5	88-1	B5
					PTG2·$\overline{\text{LOOP}}$	70	4	"	D5
LOOP*	22	6	88-3	D5	PC IN*	35	5	88-2	D5
					CON INST*	36	10	88-1	A2
					MA LOAD*	56	13	"	D3
LOOP SET	83	2	88-3	D5	(TS3/TS0)	65	9	"	C5
					PTG-5	70	10	"	D5
					LOOP	103	2	88-3	D5
LOOP SET*	104	8	88-3	D5	DCHA SET*	71	12	88-1	C5
					LOOP SET	83	1	88-3	D5
MA1	33	15	103-1	C7	[SARD1] (Jumper)	35	4	103-1	D8
MA1*	33	14	"	C7	[SARD1] "	35	4	"	D8
MA2	33	10	"	C7	[SARD2] "	35	1	"	D8
MA2*	33	11	"	C7	[SARD2] "	35	1	"	D8
MA3	33	9	"	C7	[SARD2] "	35	2	"	D8
MA3*	33	8	"	C7	[SARD3] "	35	2	"	D8
MA4	29	16	"	C7	MA4B*	67	3	103-4	D8
MA4*	29	1	"	C7					
MA4B*	67	4	103-4	D8	MA4B	67	11	"	D8
					Y ADDR DCDR	52	5,4	"	7
					"	66	5,4	"	7
MA4B	67	10	103-4	D8	"	54	5,4	"	7
					"	62	5,4	"	7
MA5	29	15	103-1	C6					
MA5*	29	14	"	C6	MA5B	67	5	103-4	C8
MA5B	67	6	103-4	C8	MA5B*	67	9	"	C8
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
					Y ADDR DCDR	54	7	103-4	7
					"	62	7	"	7
					"	52	7	"	7
					"	66	7	"	7
MA5B*	67	8	103-4	C8	Y ADDR DCDR	54	1	103-4	7
					"	62	1	"	7
					"	52	1	"	7
					"	66	1	"	7
MA6	29	10	103-1	C5	MA6B*	67	1	"	B8
MA6*	29	11	"	C5					
MA6B*	67	2	103-4	B8	MA6B	67	13	103-4	B8
					Y ADDR DCDR	62	6	"	7
					"	66	6	"	7
MA6B	67	12	103-4	B8	"	54	6	"	7
					"	52	6	"	7
MA7	29	9	103-1	C5	MA7B*	44	11	"	A8
MA7*	29	8	"	C5					
MA7B*	44	10	103-4	A8	MA7B	44	3	103-4	A8
					Y ADDR DCDR	60	5,4	"	A
					"	50	5,4	"	A
MA7B	44	4	103-4	A8	"	57	5,4	"	A
					"	47	5,4	"	A
MA8	25	16	103-1	C4	MA8B*	44	9	103-4	A8
MA8*	25	1	"	C4					
MA8B*	44	8	103-4	A8	MA8B	44	5	103-4	A8
					Y ADDR DCDR	60	7	"	A
					"	50	7	"	A
					"	57	7	"	A
					"	47	7	"	A
MA8B	44	6	103-4	A8	"	60	1	"	A
					"	50	1	"	A
					"	57	1	"	A
					"	47	1	"	A
MA9	25	15	103-1	C4	MA9B*	44	13	"	A8
MA9*	25	14	"	C4					
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
MA9B*	44	12	103-4	A8	MA9B	44	1	103-4	A8
					Y ADDR DCDR	60	6	"	A
					"	57	6	"	A
MA9B	44	2	103-4	A8	"	50	6	"	A
					"	47	6	"	A
MA10	25	10	103-1	C4	MA10B*	71	3	"	D8
MA10*	25	11	"	C4					
MA10B*	71	4	103-3	D8	MA10B	71	11	103-3	D8
					X ADDR DCDR	73	5,4	"	7
					"	77	5,4	"	7
MA10B	71	10	103-3	D8	"	72	5,4	"	7
					"	76	5,4	"	7
MA11	25	9	103-1	C4					
MA11*	25	8	"	C4	MA11B	71	5	103-3	C8
MA11B	71	6	103-3	C8	MA11B*	71	9	"	C8
					X ADDR DCDR	72	7	"	7
					"	76	7	"	7
					"	73	7	"	7
					"	77	7	"	7
MA11B*	71	8	103-3	C8	"	72	1	"	7
					"	76	1	"	7
					"	73	1	"	7
					"	77	1	"	7
MA12	22	16	103-1	C3	MA12B*	71	1	103-3	B8
MA12*	22	1	"	C3					
MA12B*	71	2	103-3	B8	MA12B	71	13	103-3	B8
					X ADDR DCDR	76	6	"	7
					"	77	6	"	7
MA12B	71	12	"	B8	"	72	6	"	7
					"	73	6	"	7
MA13	22	15	103-1	C3	MA13B*	80	11	"	A8
MA13*	22	14	"	C3					
MA13B*	80	10	103-3	A8	MA13B	80	3	103-3	A8
					X ADDR DCDR	79	5,4	"	A
					"	74	5,4	"	A
MA13B	80	4	103-3	A8	X ADDR DCDR	78	5,4	"	A
					"	75	5,4	"	A
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
MA14	22	10	103-1	C2	MA14B*	80	9	103-3	A8
MA14*	22	11	"	C2					
MA14B*	80	8	103-3	A8	MA14B	80	5	103-3	A8
					X ADDR DCDR	79	7	"	A
					"	74	7	"	A
					"	78	7	"	A
					"	75	7	"	A
MA14B	80	6	103-3	A8	"	79	1	"	A
					"	74	1	"	A
					"	78	1	"	A
					"	75	1	"	A
MA15	22	9	103-1	C2	MA15B*	84	13	"	A8
MA15*	22	8	"	C2					
MA15B*	80	12	103-3	A8	MA15B	80	1	103-3	A8
					X ADDR DCDR	79	6	"	A
					"	78	6	"	A
MA15B	80	2	103-3	A8	"	74	6	"	A
					"	75	6	"	A
MA LOAD*	60	8	88-1	D2		(B7)		90-1	
					MTG(SH)	35	11	88-1	C7
					[MA LOAD]	30	9,10	103-1	C8
					"	30	12, 13	"	C8
[MA LOAD]	30	8	103-1		MA1-3	33	13	103-1	
						33	4	"	
					MA4-7	29	13	"	
						29	4	"	
					MA8-11	25	13	"	
						25	4	"	
					MA12-15	22	13	"	
MBC8*	33	5	88-2	A5	(SKIP LOGIC)	11	9	88-3	B7
					MBC8	27	1	88-2	A5
					MBC(DS)	33	4	"	A5
					(SH/SWP DCDR)	51	3	"	B6
					(IO DCDR)	63	3	88-1	A4
*Indicates "Not"					"	63	13	"	A4

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
MBC8	27	2	88-2	A4	(SKIP LOGIC)	11	5	88-3	B7
					S0	47	5	88-2	C3
MBC9*	32	5	88-2	A4	(SH/SWP DCDR)	51	2	"	B6
					(IO DCDR)	63	2	88-1	A4
					"	63	14	"	A4
					MBC9	79	9, 10	88-2	A4
MBC9	79	8	88-2	A3	(SKIP LOGIC)	80	1	88-3	B6
MBC10*	33	9	88-2	A5	MBC10	27	9	88-2	A4
					CRY ENAB	77	2	88-3	C7
MBC10	27	8	88-2	A4	CPU INST*	9	5	88-2	B8
					DS0*	8	9	88-1	C4
MBC11*	32	9	"	A4	MBC11	27	13	88-2	A3
MBC11	27	12	"	A3	DS1*	8	11	88-1	C4
					CPU INST*	9	4	88-2	B8
					CRY ENAB	77	5	88-3	C7
MBC12*	33	7	88-2	A5	MBC12	27	11	88-2	A4
MBC12	27	10	88-2	A4	DS2*	8	13	88-1	C4
					CPU INST*	9	2	88-2	D8
					LOAD CRY*	101	1	88-3	C6
					S MULT	116	3	88-4	C7
MBC13*	32	7	88-2	A4	MBC13	27	3	88-2	A3
MBC13	27	4	"	A3	DS3*	22	9	88-1	C4
					CPU INST*	9	1	88-2	B8
					(SKIP LOGIC)	77	1	88-3	B7
					S MULT	116	6	88-4	C7
MBC14* (NOT USED)	33	11	88-2						
MBC14	33	12	88-2	A5	DS4*	8	3	88-1	C4
					CPU INST*	10	1	88-2	B8
					(SKIP LOGIC)	77	10	88-3	B7
					S MULT	116	13	88-4	C6
MBC15*	32	11	88-2	A4	(SKIP LOGIC)	80	4	88-3	B6
MBC15	32	12	"	A3	DS5*	8	1	88-1	C4
					CPU INST*	10	2	88-2	B8
					S MULT	116	10	88-4	C6
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
MB CLR*	19	6	88-1	D2		(B86)		103-1	B8
[MB CLEAR]	18	8	103-1	B8		30	2,4,5	103-1	B8
[MB CLEAR· SEL]	30	6	"	B8	INH0 F/F	34	13	103-1	B
					INH1 F/F	34	1	"	B
					INH2 F/F	32	1	"	B
					INH3 F/F	32	13	"	B
					INH4 F/F	31	13	"	B
					INH5 F/F	31	1	"	B
					INH6 F/F	28	1	"	B
					INH7 F/F	28	13	"	B
					INH8 F/F	27	13	"	B
					INH9 F/F	27	1	"	B
					INH10 F/F	24	1	"	B
					INH11 F/F	24	13	"	B
					INH12 F/F	23	13	"	B
					INH13 F/F	23	1	"	B
					INH14 F/F	21	1	"	B
					INH15 F/F	21	13	"	B
MB LOAD	14	6	88-1	C2		(B74)		90-1	
						36	9	103-1	B8
[MB LOAD· SEL]	36	8	103-1	B8	INH0 F/F	34	11	103-1	B
					INH1 F/F	34	3	"	B
					INH2 F/F	32	3	103-1	B
					INH3 F/F	32	11	"	B
					INH4 F/F	31	11	"	B
					INH5 F/F	31	3	"	B
					INH6 F/F	28	3	"	B
					INH7 F/F	28	11	"	B
					INH8 F/F	27	11	"	B
					INH9 F/F	27	3	"	B
					INH10 F/F	24	3	"	B
					INH11 F/F	24	11	"	B
					INH12 F/F	23	11	"	B
					INH13 F/F	23	3	"	B
					INH14 F/F	21	3	"	B
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
					INH15 F/F	21	11	103-1	B
MBO0*	40	5	88-4	D4		(B79)			
					[MD0]	17	9	103-1	A7
MBO1*	39	5	88-4	D3		(B77)			
					MB1	17	5	103-1	A7
					(CON IND) (P14)	7	11	89-1	D7
MBO2*	37	5	88-4	C4		(B44)			
					MD2	15	9	103-1	A7
					(CON IND) (P15)	7	5	89-1	D7
MBO3*	38	5	88-4	C3		(B43)			
					MD3	15	5	103-1	A6
					(CON IND) (P38)	8	9	89-1	D7
MBO4*	40	7	88-4	D4		(B42)			
					MD4	13	9	103-1	A6
					(CON IND) (P16)	8	11	89-1	D6
MBO5*	39	7	88-4	D3		(B32)			
					MD5	13	5	103-1	A6
					(CON IND) (P11)	8	5	89-1	D6
MBO6*	37	7	88-4	C4		(B16)			
					MD6	11	9	103-1	A5
					(CON IND) (P35)	9	9	89-1	D6
MBO7*	38	7	88-4	C3		(B14)			
					MD7	11	5	103-1	A5
					(CON IND) (P9)	9	11	89-1	D5
MBO8*	40	9	88-4	D4		(B12)			
					MBO12 SAVE*	42	2	88-1	C8
					MD8	9	9	103-1	A5
					(CON IND) (P18)	9	5	89-1	D5
MBO9*	39	9	88-4	D3		(B9)			
					MD9	9	5	103-1	C4
					(CON IND) (P8)	10	9	89-1	D5
MBO10*	37	9	88-4	C4					
					MD10	7	9	103-1	C4
					(CON IND) (P44)	10	11	89-1	D4
MBO11*	38	9	88-4	C3		(B5)			
					MD11	7	5	103-1	C4
					(CON IND) (P6)	10	5	89-1	D4
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
MBO12*	40	11	88-4	D4		(A39)			
					MD12	5	9	103-1	A3
					(CON IND) (P30)	11	9	89-1	D4
MBO12	40	12	88-4	D3	D MULT	121	3	88-4	C8
					MULT	120	6	"	D6
					ADDER TEST	57	12	88-3	A4
MBO13*	39	11	88-4	D2		(A37)			
					ADDER TEST	60	5	88-3	A4
					MD13	5	5	103-1	A3
					(CON IND) (P4)	11	11	89-1	D3
MBO13	39	12	88-4	D2	D MULT	121	6	88-4	C8
					MULT	120	3	"	D6
MBO14*	37	11	88-4	C4		(A43)			
					ADDER TEST	60	4	88-3	A4
					MD14	3	9	103-1	A3
					(CON IND) (P12)	11	5	89-1	D3
MBO14	37	12	88-4	C3	D MULT	121	13	88-4	C7
					MULT	120	21	"	C6
MBO15*	38	11	88-4	C2		(A41)			
					MD15	3	5	103-1	A2
					(CON IND) (P28)	11	1	89-1	D3
MBO15	38	12	88-4	C2	ADDER TEST	84	9	88-3	A4
					D MULT	120	10	88-4	C7
					MULT	121	18	"	C6
MBO12 SAVE*	42	7	88-1	C7	S0	48	2	88-2	C4
					ADD ONE*	90	5	"	D4
[MD0]	17	8	103-1	B7	INH0	34	12	103-1	B7
MD1	17	6	103-1	B7	INH1	34	2	103-1	B7
					MA1	33	3	"	C7
MD2	15	8	103-1	B7	INH2	32	2	"	B7
					MA2	33	6	"	C7
MD3	15	6	103-1	B6	INH3	32	12	"	B7
					MA3	33	7	"	C7
MD4	13	8	103-1	B6	INH4	31	12	"	B7
					MA4	29	2	"	C7

*Indicates "Not"

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
MD5	13	6	103-1	B6	INH5	31	2	103-1	B6
					MA5	39	3	"	C6
MD6	11	8	103-1	B5	INH6	28	2	"	B5
					MA6	29	6	"	C5
MD7	11	6	103-1	B5	INH7	28	12	"	B5
					MA7	29	7	"	C5
MD8	9	8	103-1	B5	INH8	27	12	"	B5
					MA8	25	2	"	C5
MD9	9	6	103-1	B4	INH9	27	2	"	B4
					MA9	25	3	"	C4
MD10	7	8	103-1	B4	INH10	24	2	"	B4
					MA10	25	6	"	C4
MD11	7	6	103-1	B4	INH11	24	12	"	B4
					MA11	25	7	"	C4
MD12	5	8	103-1	B3	INH12	23	12	"	B3
					MA12	22	2	"	C3
MD13	5	6	103-1	B3	INH13	23	2	"	B3
					MA13	22	3	"	C3
MD14	3	8	103-1	B3	INH14	21	2	"	B3
					MA14	22	6	"	C3
MD15	3	6	103-1	B2	INH15	21	12	"	B2
					MA15	22	7	103-1	C2
MULTIPLY/ DIVIDE MD SEL1*	SEL (A87)		88-2	C5	ACS1 SEL*	49 (B71)	6,8	88-2	C4
MEM0*	16	3	103-1	B7					
(ACEX+ ACDP)	1	3	89-1	A5	(CON IND) (P39)	7	9	89-1	C8
					IR0*	28	3	88-2	A6
					MBO0*	40	3	88-4	D4
					Defer Again	76	2	88-2	C7
					(EFA LOGIC)	55	10, 13	88-3	C6
MEM1*	16	6	103-1	B7		(B70)			
(ACDP)	1	6	89-1	A5	(CON IND) (P41)	7	13	89-1	C7
					IR1*	29	2	88-2	A7
					MBO1*	39	3	88-4	D3
					(EFA LOGIC)	55	9	88-3	C6
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
MEM2*	14	3	103-1	B7		(B47)			
(DP+DPN)	2	8	89-1	A4	(CON IND) (P13)	7	3	89-1	C7
					IR2*	29	15	88-2	A7
					MBO2*	37	3	88-4	C4
					(EFA LOGIC)	55	1	88-3	C6
MEM3*	14	6	103-1	B6					
(ACEX+ACDP)	1	11	89-1	A7		(B68)			
					(CON IND) (P43)	7	1	89-1	C7
					IR3*	29	14	88-2	A7
					MBO3*	38	3	88-4	C3
MEM4*	12	3	103-1	B6		(B28)			
(ACEX+ACDP)	1	8	89-1	A6	(CON IND) (P37)	8	13	89-1	C6
					IR4*	29	3	88-2	A7
					MBO4*	40	2	88-4	D4
MEM5*	12	6	103-1	B6		(B26)			
(EX+STRT+ ACDP)	2	11	89-1	A3	(CON IND) (P36)	8	3	89-1	C6
					IR5*	28	2	88-2	A6
					MBO5*	39	2	88-4	D3
MEM6*	10	3	103-1	B5		(B22)			
(EX+EXN+DP+ DPN)	2	3	89-1	A3	(CON IND) (P10)	8	1	89-1	C6
					IR6*	28	15	88-2	A6
					MBO6*	37	2	88-4	C4
MEM7*	10	6	103-1	B5		(B24)			
(EXN+DPN)	2	6	89-1		(CON IND) (P42)	9	13	89-1	C5
					IR7*	28	14	88-2	A6
					MBO7*	38	2	88-4	C3
MEM8*	8	3	103-1	B5		(A55)			
					(CON IND) (P34)	9	3	89-1	C5
					MBC8*	33	3	88-2	A5
					MBO8*	40	15	88-4	D4
MEM9*	8	6	103-1	B4		(A53)			
					(CON IND) (P7)	9	1	89-1	C5
					MBC9*	32	3	88-2	A4
					MBO9*	39	15	88-4	C3
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
MEM10*	6	3	103-1	B4		(A45)			
					(CON IND) (P32)	10	13	89-1	C4
					MBC10*	33	15	88-2	A5
					MBO10*	37	15	88-4	C4
MEM11*	6	6	103-1	B4		(A51)			
					(CON IND) (P31)	10	3	89-1	C4
					MBC11*	32	15	88-2	A4
					MBO11*	38	15	88-4	C3
MEM12*	4	3	103-1	B3		(A36)			
					(CON IND) (P5)	10	1	89-1	C4
					MBC12*	33	2	88-2	A5
					MBO12*	40	14	88-4	D4
MEM13*	4	6	103-1	B3		(A35)			
					(CON IND) (P29)	11	13	89-1	C3
					MBC13*	32	2	88-2	A4
					MBO13*	39	14	88-4	D2
MEM14*	2	3	103-1	B2		(B76)			
					(CON IND) (P3)	11	3	89-1	C3
					MBC14*	33	14	88-2	A5
					MBO14*	37	14	88-4	C4
MEM15*	2	6	103-1	B2		(B18)			
					(CON IND) (P2)	12	13	89-1	C3
					MBC15*	32	14	88-2	A4
					MBO15*	38	14	88-4	C2
MEM CLK	73	6	88-1	A6		(B48)		90-1	
					(MTG)	17	6	88-1	D6
					(KEY/RUN/DCH)	23	6	"	C6
					(ACTG)	54	6	"	D8
					LOAD AC*	93	4	88-3	D3
					S BUFF	115	6	88-4	C7
					D BUFF	122	6	"	C8
					IR4, IR1-3	29	6	88-2	A8
MEM OK	(A9)		91	B2	RUN LOGIC	62	5	88-1	C7
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
[MSKO]	4	11	88-1	B4	MSKO*	5	3	88-1	A4
MSKO*	5	4	"	A4		(A38)		90-1	
MSTP	(A20)	(P48)	89-1		(RUN LOGIC)	24	1, 10	88-1	B7
MTG0	17	5	88-1	D6	INHIBIT	13	10	"	C2
					DCHI	14	13	"	C2
					MTG0*	16	13	"	D6
					MTG	17	2, 15, 14	"	D6
					READ1*	19	1	"	D2
					MB CLR*	19	4	"	D2
MTG0*	16	12	88-1	D6	MTG(SH)(Logic)	36	4	"	C7
MTG1	17	7	88-1	D6	RQENB*	16	5	"	C2
					MTG1*	16	11	"	D6
					READ2*	19	10	"	D2
MTG1*	16	10	88-1	D6	MB CLR*	19	5	"	D2
					DCHO	18	10	"	B2
					MTG(SH)(Logic)	36	5	"	D6
MTG3*	17	12	88-1	D6	MTG(DS)	17	4	"	D6
					STROBE	18	1, 2, 4	"	D2
					READ1*	19	2	"	D2
					READ2*	19	9	"	D2
MULT0*	120	10	88-4	CD 5	MBO(DS)	40	4	88-4	CD34
MULT1*	120	11	"	"	MBO(DS)	39	4	"	"
MULT2*	120	13	"	"	MBO(DS)	37	4	"	"
MULT3*	120	14	"	"	MBO(DS)	38	4	"	"
OVFLO	15	8	88-1	B2		(B39)		90-1	
PACK	103	9	88-3	D5	ACS1 SEL*	49	10	88-2	C5
					ACS2 SEL*	49	12	"	B5
					PACK*	83	13	88-3	D5
PACK*	83	12	88-3	D5	ACS1 SEL*	49	4	88-2	C5
					ACS2 SEL*	49	2	"	B5
					LOAD AC*	111	5	88-3	D3
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
[PC0]	119	10	88-4	A5	MULT0*	120	10	88-4	CD5
[PC1]	119	9	"	A5	MULT1*	120	11	"	CD5
[PC2]	119	7	"	A5	MULT2*	120	13	"	CD5
[PC3]	119	6	"	A5	MULT3*	120	14	"	CD5
PC ENAB*	61	8	88-3	B3	PC IN*	36	1	88-2	D5
					LOAD PC*	57	5	88-3	B3
					E SET	74	2	88-2	C7
PC IN*	36	3	88-2	D4	PC	119	11	88-4	A5
					Multiplexer	120	7,8,		
							9	88-4	C5
					MULT(ENAB)	120	7,8,		
							9	88-4	C5
PI	95	11	88-2	D6	PC IN*	35	4	88-2	D5
					ADD ONE*	90	3	"	D4
					CLR SKIP*	100	5	88-3	A4
					Disable D Mult	46	2,3	88-2	B3
PI*	95	12	88-2	D6	IR(SH)	114	2	88-2	A8
					IR(DS)	12	13	"	A8
					D SET	74	11	"	C7
					ADD ONE*	82	13	"	D3
					ION*	84	1	"	C7
					LOOP SET*	84	12	88-3	D6
PI SET	96	6	88-2	C6	PI	95	14	"	
					FETCH	96	9	"	D6
					LOAD MBO*	98	2	88-3	A3
PL*	(A19)	(P23)	89-1	B2	KEYM· PL	41	9	88-1	C6
					(RUN Logic)	43	3	"	B7
					Disable D Mult	87	9	88-2	B4
PRESET*	22	10	88-1	B7	MTG(MR)	17	1	88-1	D7
					INPUT	66	1	"	B5
					PTG(MR)	69	1	"	D5
					SKIP	78	1	88-3	B5
					(Major States)	95	1	88-2	D7
PTG0	69	9	88-1	D4	PTG DCDR	68	2	88-1	D3
					"	68	14	"	D3
					PTG	69	14	"	D4
					PC	119	4	88-4	A5
*Indicates "Not"					PC	119	13	"	A5

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
					MB LOAD	112	13	88-1	C3
PTG1	69	11	88-1	D4	EFA· $\overline{PTG1}$	34	4	88-2	A3
					PTG DCDR	68	3	88-1	D3
					"	68	13	"	D3
					End Cycle F/F	113	3	"	D5
					PC	119	5	88-4	A5
					PC	119	14	"	A5
					MB LOAD	112	10	88-1	C3
PTG1*	69	12	88-1	D4	SO	47	2	88-2	C3
					PTG	69	15	88-1	D4
PTG2*	68	10	88-1	D4	ADDER Test	57	13	88-3	A6
					TS0/TS3	65	10	88-1	C5
					PTG2	67	9	"	D3
					PTG2· $\overline{LOOP}$	70	5	"	D5
PTG2	67	8	88-1	D3	INPUT F/F	66	12	"	B5
PTG5	70	8	88-1	D4	Key/Run/DCH/				
					(LD)	23	10	88-1	C6
					(LD)	42	10	"	C8
					TS0/TS F/F	66	2	"	C5
					Adder Test	78	12	88-3	A5
					"	79	4	"	A5
					Major States				
					(LD)	95	10	88-2	D7
					LOAD MBO*	98	3	88-3	A3
					LOAD MBO*	98	4	88-3	A3
PTG5 ENAB*	68	6	88-1	D3	INH TRANS*	56	1	88-1	B2
					PTG5	70	9	"	D5
					Pack Logic	70	12	88-3	C6
					SKIP F/F	79	12	"	B5
					LOAD ACB	100	12	"	C3
PTG=0· TS0	113	9	88-1	A5	Adder Test	58	13	"	A6
					MA LOAD*	60	12	88-1	D2
					ADD ONE*	88	9	88-2	D3
					Shifter Logic	90	9	88-4	A7
PTG=0· TS0*	113	8	88-1	A5	ADD ONE*	88	4	88-2	D3
					Shifter Logic	90	13	88-4	A7
					SHIFT ACB	93	10	88-3	C4
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
PTG=0·TS3*	68	4	88-1	D3	PTG=0·TS3	67	13	88-1	D3
					ADD ONE*	88	5	88-2	D3
PTG=0·TS3	67	12	88-1	D3	INPUT F/F	9	9, 10	88-1	C5
					MTG(LD)	17	10	"	D7
					ADD ONE*	88	1,2	88-2	D3
PTG=1·TS0*	68	11	88-1	D3	ADDER Test	80	10	88-3	A6
					SHIFT ACB	93	11	"	C4
PTG=1·TS3*	68	5	88-1	D3	(IO DCDR)	109	5	88-1	A5
PTG2·$\overline{\text{LOOP}}$	70	6	88-1	D4	$\overline{\text{PTG2}}$+LOOP	73	13	88-1	D4
					LOAD MBO*	98	9	88-3	B3
					LOOP SET*	104	3,4	88-3	D6
$\overline{\text{PTG2}}$+LOOP	73	12	88-1	D4	LOAD IR	34	1	88-2	A7
					SKIP (F/F)	79	2	88-3	B5
PULSE ENAB	109	6	88-1	A5	OVFLO	15	12	88-1	B2
					IO DCDR	62	1	"	A5
PWR FAIL*	(A5)		91-1	C2	PWR LOW	86	12	88-3	D8
					AC CLR	20	12	88-1	A6
PWR LOW	102	11	88-3	D7	(SKIP Logic)	11	1	88-3	B1
PWR LOG*	102	12	"	D7	PI SET	75	13	88-2	C7
					PWR LOW	86	13	88-3	D8
READ1*	19	3	88-1	D2		(B87)		103-1	
					MTG(SH)	35	10	88-1	D6
					READ 1B	18	13	103-1	D6
	18	12	103-1	D6	"	19	5,4	103-1	D6
					READ2B	19	12	103-1	D6
READ 1B	19	6	103-1	D5	"	19	10	"	
					(X ADDR DCDR)	72	2	103-3	A7
					"	76	2	"	A7
					"	73	2	"	A7
					"	77	2	"	A7
					"	79	3	"	A7
					"	74	3	"	A7
					"	78	3	"	A7
					"	75	3	"	A7

*Indicates "Not"

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
READ 2*	19	8	88-1	D2		(B90)		103-1	D6
					READ 2B	18	11	"	D6
	18	10	103-1	D6	"	19	9	"	D6
						19	13	"	D6
READ 2B	19	8	103-2	D5	(Y ADDR DCDR)	54	2	103-4	A7
					"	62	2	"	A7
					"	52	2	"	A7
					"	66	2	"	A7
					"	60	3	"	A7
					"	50	3	"	A7
					"	57	3	"	A7
					"	47	3	"	A7
READ IO*	12	3	88-1	B2		(B83)		103-1	A8
					[READ IO]	18	3	"	A8
	18	4	103-1	A8	[MD0]	17	13	"	A
					MD1	17	2	"	A
					MD2	15	13	"	A
					MD3	15	2	"	A
					MD4	13	13	"	A
					MD5	13	2	"	A
					MD6	11	13	"	A
					MD7	11	2	"	A
					MD8	9	13	"	A
					MD9	9	2	"	A
					MD10	7	13	"	A
					MD11	7	2	"	A
					MD12	5	13	"	A
					MD13	5	2	103-1	A
					MD14	3	13	"	A
					MD15	3	2	"	A
[READ IO]	18	6	103-1	A8		18	5	"	A8
					[MD0]	17	10	"	A
					MD1	17	4	"	A
					MD2	15	10	"	A
					MD3	15	4	"	A
					MD4	13	10	"	A
					MD5	13	4	"	A
*Indicates "Not"					MD6	11	10	"	A

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
					MD7	11	4	103-1	A
					MD8	9	10	"	A
					MD9	9	4	"	A
					MD10	7	10	"	A
					MD11	7	4	"	A
					MD12	5	10	"	A
					MD13	5	4	"	A
					MD14	3	10	"	A
					MD15	3	4	"	A
RESET*	22	4	88-1	B7	PRESET*	3	13	88-1	B7
					IORST	10	13	"	A4
						21	9	"	B8
					KEY/RUN/DCH				
					(MR)	23	1	"	C6
					(MR)	42	1	"	C8
					ION*	84	2	88-2	C7
					LOOP/PACK				
					(MR)	103	1	88-3	D5
RESTART*			88-1	A8	KEY SEEN F/F	3	5	88-1	B8
					Disable D Mult	87	10	88-2	B4
RELOAD Disable*	(B72)					36	10, 12	103-1	B8
RESTART Enable	(A32)	(P19)	89-1	B7					
RINH0	(A5)		103-2	7					
RINH1	(A7)		"	7					
RINH2	(A9)		"	7					
RINH3	(A11)		"	7					
RINH4	(A13)		"	7					
RINH5	(A15)		"	7					
RINH6	(A18)		"	7					
RINH7	(A17)		"	7					
RINH8	(A19)		"	4					
RINH9	(A24)		"	4					
RINH10	(A23)		"	4					
RINH11	(A21)		"	4					
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
RINH12	(A28)		103-2	4					
RINH13	(A25)		"	4					
RINH14	(A29)		"	4					
RIN15	(A27)		"	4					
RQENB*	16	6	88-1	C2		(B41)		90-1	B8
RST*	(A30)	(P20)	89-1	B6	RESET*	21	12	88-1	B8
RUN	23	7	88-1	C6	RUN*	22	13	"	C6
					CPU CLK	72	4, 10	"	A7
RUN*	22	12	88-1	C6	(CON IND) (A14)	12	1	89-1	D2
					KEY SEEN F/F	2	1,2	88-1	B8
S0	92	3	88-2	C2	ADDER	117	3,6	88-4	D8
S1	91	8	"	C2	"	117	5	"	D8
S2	91	11	"	C2	S1	91	4	88-2	C3
					ADDER	117	4	88-4	D8
[S BUFFR0]	115	5	88-4	C7	S MULT	116	2	"	C7
[S BUFFR1]	115	7	"	C7	"	116	5	"	C7
[S BUFFR2]	115	9	"	C7	"	116	14	"	C7
[S BUFFR3]	115	11	"	C7	"	116	11	"	C7
SELB*	(A82)		90-1		SKIP Logic	11	10	88-3	B6
SELD*	(A80)		90-1		"	11	4	"	B6
SELECT	35	8	103-1	D7	STRB A,B, C,D	1	1, 10	103-1	D4
					"	1	12, 13	"	D5
					READ 1B	19	1,2	"	D6
					INH GATE A,B	26	2,4, 5	"	D3
					"	41	10, 12, 13	"	D3
					(DRIVE IO)	26	13	"	C8
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
					(INH TRANS*)	36	1	103-1	B8
					(MB LOAD)	36	13	"	B8
					(MB CLEAR)	30	1	"	B8
SERIAL CRY	54	12	88-1	D7	OVFLO	15	13	88-1	B2
					ADD ONE*	88	6	88-2	D3
SET ION*	63	10	"	B8	ION*	82	4	"	C7
SHIFT0*	125	13	88-4	A 678		(B94)			
					SKIP Logic	110	12	88-3	A6
					ACD	123	4	88-4	A 678
					ACS	124	4	"	"
SHIFT1*	125	14	88-4	"		(B96)			
					SKIP Logic	110	10	88-3	A6
					ACD	123	6	88-4	A 678
					ACS	124	6	"	"
SHIFT2*	125	11	88-4	"		(B93)			
					SKIP Logic	110	13	88-3	A6
					ACD	123	10	88-4	A 678
					ACS	124	10	"	"
SHIFT3*	125	10	88-4	"					
					SKIP Logic	110	9	88-3	A6
					ACD	123	12	88-4	A 678
					ACS	124	12	"	"
SHIFT ACB	100	3	88-3	C2	ACB(SH)	105	13	"	B4
					ACB(SH)	106	13	"	B4
					ACB(SH)	107	13	"	B4
					ACB(SH)	108	13	"	B4
SHL*	51	6	88-2	B6	Carry F/F Logic	101	5	88-3	C6
					[SHL]	101	3	"	C6
					SHIFTER(SEL)	125	16	88-4	A8
[SHL]	101	4	88-3	C6	CRY SET*	81	2	88-3	C6
SHR*	51	5	88-2	B6	Carry F/F Logic	81	6	"	C6
					[SHR]	101	5	"	C6
					SHIFTER (SEL)	125	17	88-4	A8
[SHR]	101	6	88-3	C6	CRY SET*	81	1	88-3	C6

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
SKIP	78	5	88-3	B5	ADD ONE*	90	2	88-2	D4
SKIP*	78	6	"	B5		(B69)		90-1	
					IR0+SKIP	50	2	88-2	B6
					D SET	74	3	"	C7
					ADD ONE*	82	12	"	D3
					Test Skip Set	86	4	88-3	A8
SKIP INC*	42	12	88-1	C7	PC IN*	35	13	88-2	D5
					MA LOAD*	56	12	88-1	D3
					PC ENAB*	58	4,5	88-3	B4
					CLR SKIP*	99	12	"	B3
+SL0			103-2	7	SNS0	69	2	103-2	6
-SL0			"	7	"	69	3	"	6
+SL1			"	7	SNS1	69	6	"	6
-SL1			"	7	"	69	7	"	6
+sl2			"	7	SNS2	65	2	"	6
-SL2			"	7	"	65	3	"	6
+SL3			"	7	SNS3	65	6	"	6
-SL3			"	7	"	65	7	"	6
+SL4			"	7	SNS4	59	2	"	6
-SL4			"	7	"	59	3	"	6
+SL5			"	7	SNS5	59	6	"	6
-SL5			"	7	"	59	7	"	6
+SL6			"	7	SNS6	56	2	"	6
-SL6			"	7	"	56	3	"	6
+SL7			"	7	SNS7	56	6	"	6
-SL7			"	7	"	56	7	"	6
+SL8			"	4	SNS8	49	2	"	3
-SL8			"	4	"	49	3	"	3
+SL9			"	4	SNS9	49	6	"	3
-SL9			"	4	"	49	7	"	3
+SL10			"	4	SNS10	46	2	"	3
-SL10			"	4	"	46	3	"	3
+SL11			"	4	SNS11	46	6	"	3
-SL11			"	4	"	46	7	"	3
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
+SL12			103-2	4	SNS12	40	2	103-2	3
-SL12			"	4	"	40	3	"	3
+SL13			"	4	SNS13	40	6	"	3
-SL13			"	4	"	40	7	"	3
+SL14			"	4	SNS14	38	2	"	3
-SL14			"	4	"	38	3	"	3
+SL15			"	4	SNS15	38	6	"	3
-SL15			"	4	SNS15	38	7	"	3
[S MULT0]	116	4	88-4	C 6, 7	ADDER	117	18	88-4	D,78
[S MULT1]	116	7	"	"	ADDER	117	20	"	"
[S MULT2]	116	12	"	"	"	117	22	"	"
[S MULT3]	116	9	"	"	"	117	1	"	"
SNS0	69	14	103-2	6	SNS0*	68	9	103-2	D6
SNS0*	68	8	"	6	INH0 F/F	34	10	103-1	B7
SNS1	69	12	"	6	SNS1*	68	5	103-2	D6
SNS1*	68	6	"	6	INH1 F/F	34	4	103-1	B7
SNS2	65	14	"	6	SNS2*	64	5	103-2	C6
SNS2*	64	6	"	6	INH2 F/F	32	4	103-1	B7
SNS3	65	12	"	6	SNS3*	64	9	103-2	C6
SNS3*	64	8	"	6	INH3 F/F	32	10	103-1	B6
SNS4	59	14	"	6	SNS4*	58	9	103-2	C6
SNS4*	58	8	"	6	INH4 F/F	31	10	103-1	B6
SNS5	59	12	"	6	SNS5*	58	5	103-2	B6
SNS5*	58	6	"	6	INH5 F/F	31	4	103-1	B6
SNS6	56	14	"	6	SNS6*	55	5	103-2	B6
SNS6*	55	6	"	6	INH6 F/F	28	4	103-1	B5
SNS7	56	12	"	6	SNS7*	55	9	103-2	A6
SNS7*	55	8	"	6	INH7 F/F	28	10	103-1	B5
SNS8	49	14	"	3	SNS8*	48	9	103-2	D3
SNS8*	48	8	"	3	INH8 F/F	27	10	103-1	B5
SNS9	49	12	"	3	SNS9*	48	5	103-2	D3
SNS9*	48	6	"	3	INH9 F/F	27	4	103-1	B4
SNS10	46	14	"	3	SNS10*	45	5	103-2	C3
SNS10*	45	6	"	3	INH10 F/F	24	4	103-1	B4
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
SNS11	46	12	103-2	3	SNS11*	45	9	103-2	C3
SNS11*	45	8	"	3	INH11 F/F	24	10	103-1	B3
SNS12	40	14	"	3	SNS12*	39	9	103-2	C3
SNS12*	39	8	"	3	INH12 F/F	23	10	103-1	B3
SNS13	40	12	"	3	SNS13*	39	5	103-2	B3
SNS13*	39	6	"	3	INH13 F/F	23	4	103-1	B3
SNS14	38	14	"	3	SNS14*	37	5	103-2	B3
SNS14*	37	6	"	3	INH14 F/F	21	4	103-1	B2
SNS15	38	12	"	3	SNS15*	37	9	103-2	A3
SNS15*	37	8	"	3	INH15 F/F	21	10	103-1	B2
STA·E*	52	11	88-2	B5	LOAD MBO*	99	9	88-3	B3
					MULT (SEL)	120	17	88-4	
STOP*	(A31)	(P45)	89-1	B5	STOP SYNC	4	4,5	88-3	D8
STOP INH*	82	8	88-1	B6	DCHA SET*	71	13	88-1	C5
					SKIP INC*	87	2	"	C8
					FETCH	97	4	88-2	D7
STOP SYNC	102	5	88-3	D7	RUN Logic	43	1	88-1	B7
STROBE	18	6	88-1	C2		(B20)			
					STRB A, B, C, D	1	5	103-1	D5
STRB A	1	6	103-1	D4	SNS0*	68	10	103-2	C6
					SNS1*	68	4	"	C6
					SNS2*	64	4	"	C6
					SNS3*	64	10	"	C6
STRB B	1	6	103-1	D4	SNS4*	58	10	"	A6
					SNS5*	58	4	"	A6
					SNS6*	55	4	"	A6
					SNS7*	55	10	"	A6
STRB C	1	6	103-1	D4	SNS8*	48	10	"	C3
					SNS9*	48	4	"	C3
					SNS10*	45	4	"	C3
					SNS11*	45	10	"	C3
STRB D	1	6	103-1	D4	SNS12*	39	10	103-2	A3
					SNS13*	39	4	"	A3
					SNS14*	37	4	"	A3
					SNS15*	37	10	"	A3
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
[STRT*]	63	6	88-1	A4	STRT	7	3	88-1	A4
STRT	7	4	"	A4	(IO STRT PLS)	(A52)		90-1	
[STUTTER]	54	9	"	D7	STUTTER*	73	1	88-1	D7
STUTTER*	73	2	"	D7	CPU CLK	72	1,13	88-1	A7
SWP*	51	4	88-2	B6	LOAD ACB	100	13	88-3	C3
TS0	66	6	88-1	C5	PC IN	35	3	88-2	D5
					IR(SH)	114	5	"	B8
					INST DCDR	92	10	"	B5
					Disable D Mult	53	1	"	B3
					KEYM· PL· TS0*	57	1	88-3	C4
					PC ENAB*	61	9	"	B4
					FETCH· TS0*	64	10	88-2	D5
					PTG DCDR	68	1	88-1	D3
					S1	91	10	88-2	C3
TS3	66	5	88-1	C5	LOOP SET*	34	13	88-3	C6
					(D+E SET)+$\overline{\text{TS3}}$	36	12	88-2	D5
					ACD OUT*	45	4	"	B3
					ALC*	50	10	"	B8
					PC ENAB*	61	1	"	B4
					PC ENAB*	61	2,4	88-3	B4
					IO DCDR Logic	109	2	88-1	A5
					PTG DCDR	68	15	"	D3
					ACTG(LD)	75	9	"	D8
					Defer Again (F/F)	76	3	88-2	D7
TS3 SET	65	8	88-1	C5	PTG=0· TS0 "	112	3	88-1	A6
TEST*	(A92)		90-1		CARRY F/F	76	13	88-3	C5
TEST SKIP	102	7	88-3	D7	SKIP F/F Logic	59	3	"	B6
Test Skip Set	86	3	"	D7	RUN LOGIC	41	13	88-1	B6
					STOP INH*	82	9	"	B6
					TEST SKIP	102	2	88-3	D7
WAS JSR	103	5	88-3	D5	ACS1 SEL*	48	10	88-2	C5
					SHIFTER Logic	109	12	88-4	A8
WAS JSR*	48	8	88-2	C5		(A89)		90-1	
WHOA*	(B6)		90-1		CPU CLK	72	5,9	88-1	A7
+5 OK	(A8)		91-1	B2	RESET*	21	13	"	B8
*Indicates "Not"									

SIGNAL LIST

Table 1 - Nova 1210/1220

ORIGIN					DESTINATION				
SIGNAL	CHIP	PIN	DWG	GRID	FUNCTION	CHIP	PIN	DWG	GRID
WRITE MEM	41	6	103-1	D2	X DRIVERS	72	3	103-3	A7
						76	3	"	A7
						73	3	"	A7
						77	3	"	A7
						79	2	"	A7
						74	2	"	A7
						78	2	"	A7
						75	2	"	A7
					Y DRIVERS	54	3	103-4	A7
						62	3	"	A7
						52	3	"	A7
						66	3	"	A7
						60	2	"	A7
						50	2	"	A7
XRS			103-3	B2	X DRIVERS	72	11	103-3	B7
					"	76	11	"	B7
					"	73	11	"	B7
					"	77	11	"	B7
XWS			103-3	B2	X DRIVERS	75	11	"	B3
					"	78	11	"	B3
					"	74	11	"	B3
					"	79	11	"	B3
YRS			103-4	B2	Y DRIVERS	54	11	103-4	B7
					"	62	11	"	B7
					"	52	11	"	B7
					"	66	11	"	B7
YWS			103-4	B2	Y DRIVERS	47	11	"	B3
					"	57	11	"	B3
					"	50	11	"	B3
					"	60	11	"	

This Page Left Blank

Intentionally

ABBREVIATIONS

CENTRAL PROCESSOR AND MEMORY

NOVA 1210/1220

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

ABBREVIATIONS (Continued)

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
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
+VLamp	+ Lamp Voltage (Console indicators)
+ VMEM	+ Voltage Memory
+ 5 OK	+ 5 Volt (power) operating properly