DataGeneral

Technical Manual

NOVA 1220 COMPUTER

015-000011-04

Technical Manual

NOVA 1220 COMPUTER

015-000011-04

DATA GENERAL TECHNICAL MANUAL

NOVA 1220 COMPUTER

Models

8151, 8152, 8153, 8154

8155, 8156, 8157, 8158

Ordering No. 015-000011

Printed in the United States of America

Rev. 04 April 1974

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

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.

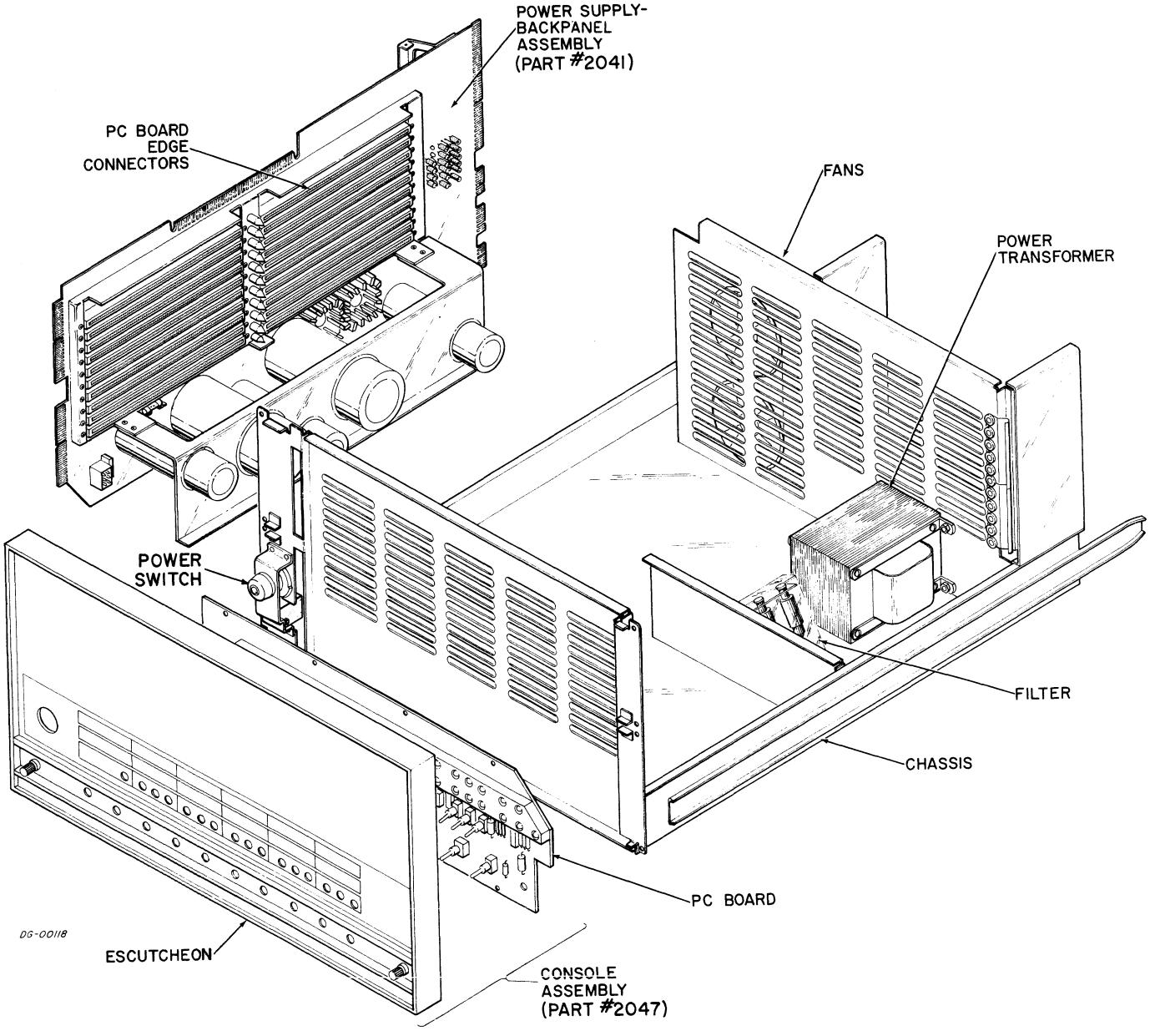


Figure O-1 Exploded View of The Nova 1220 Computer With Central Processor and Memory Cards Removed

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;

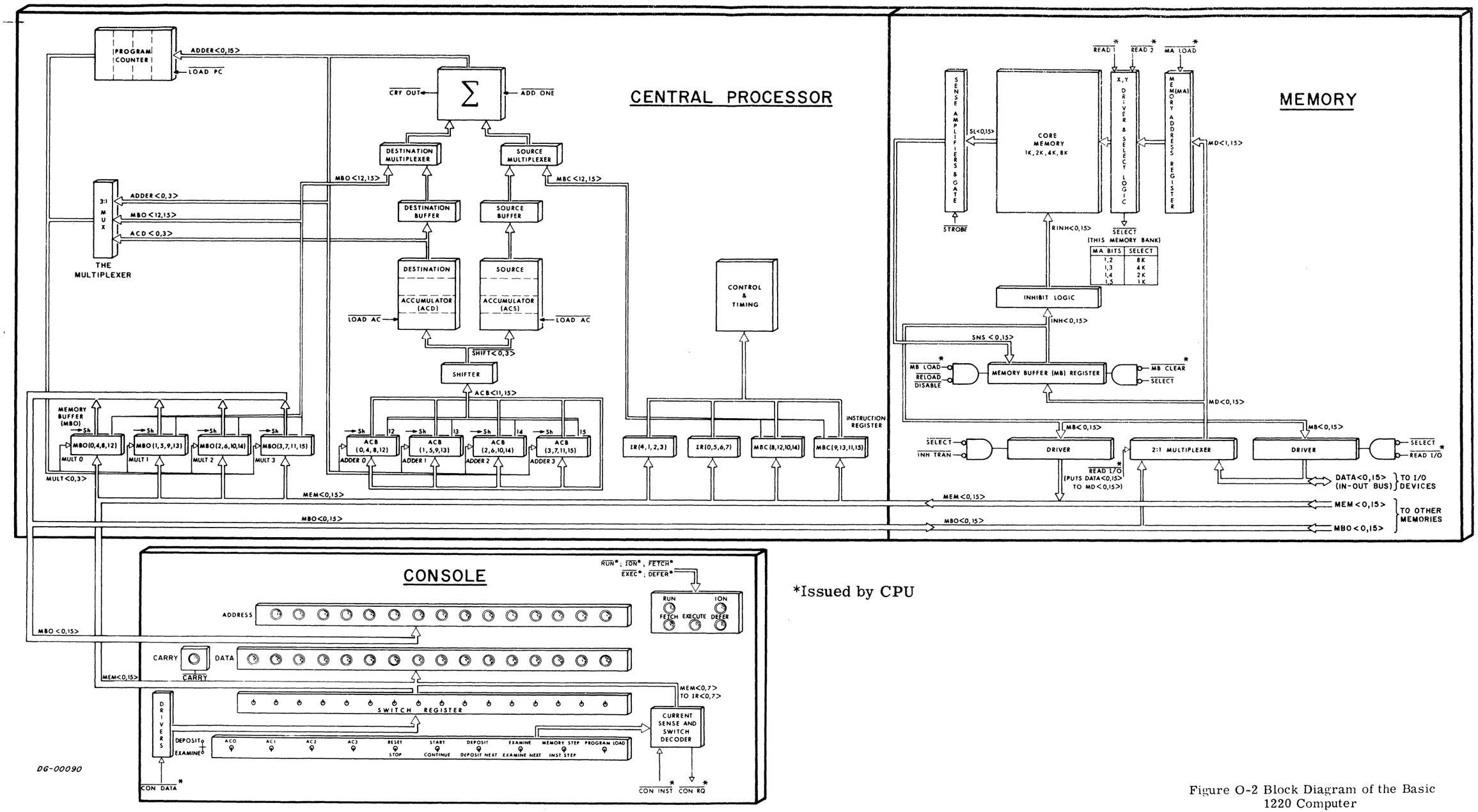
Section I explains how to install the computer;

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

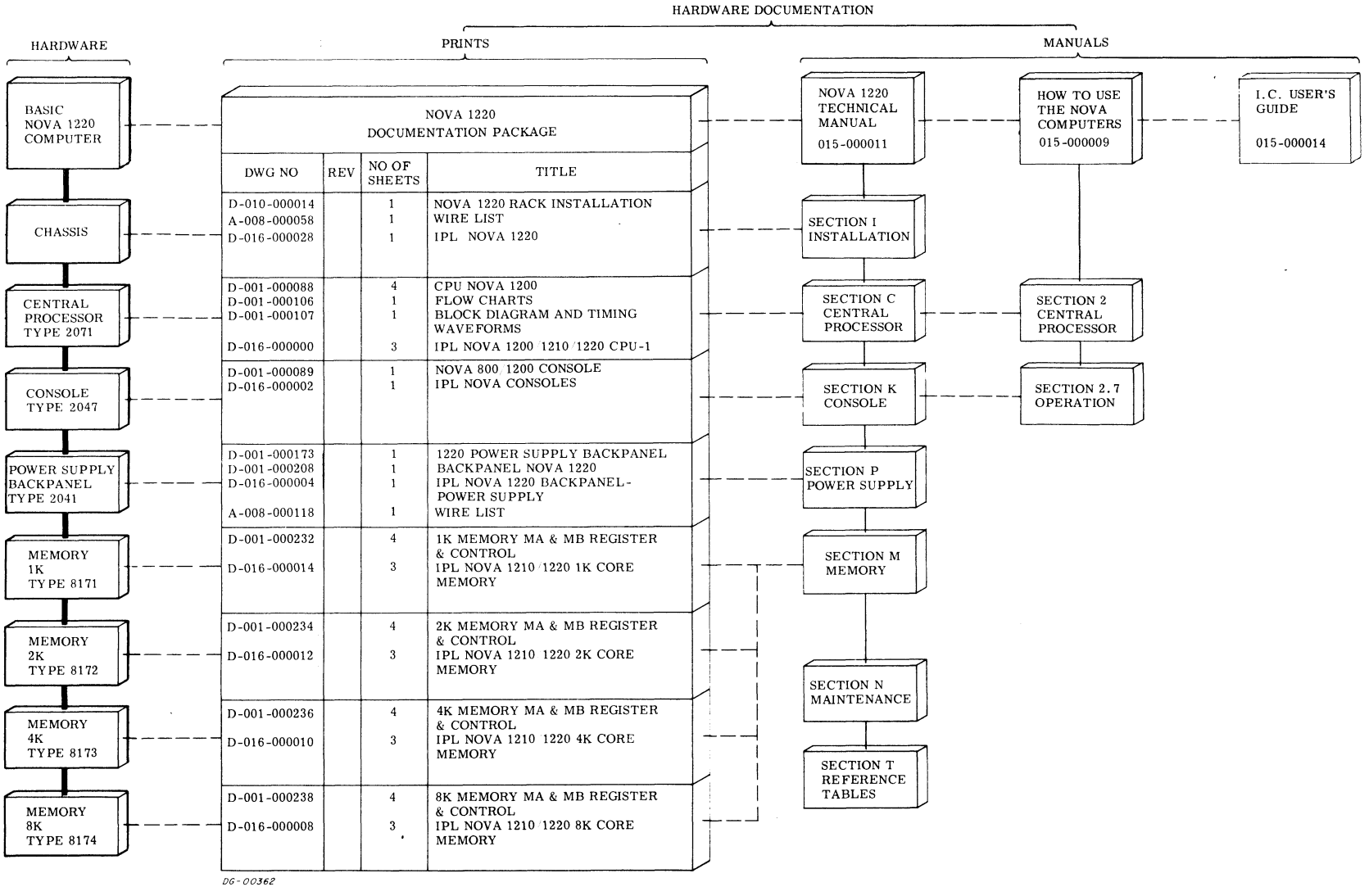


Figure O-3 Nova 1220 Hardware Documentation

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.

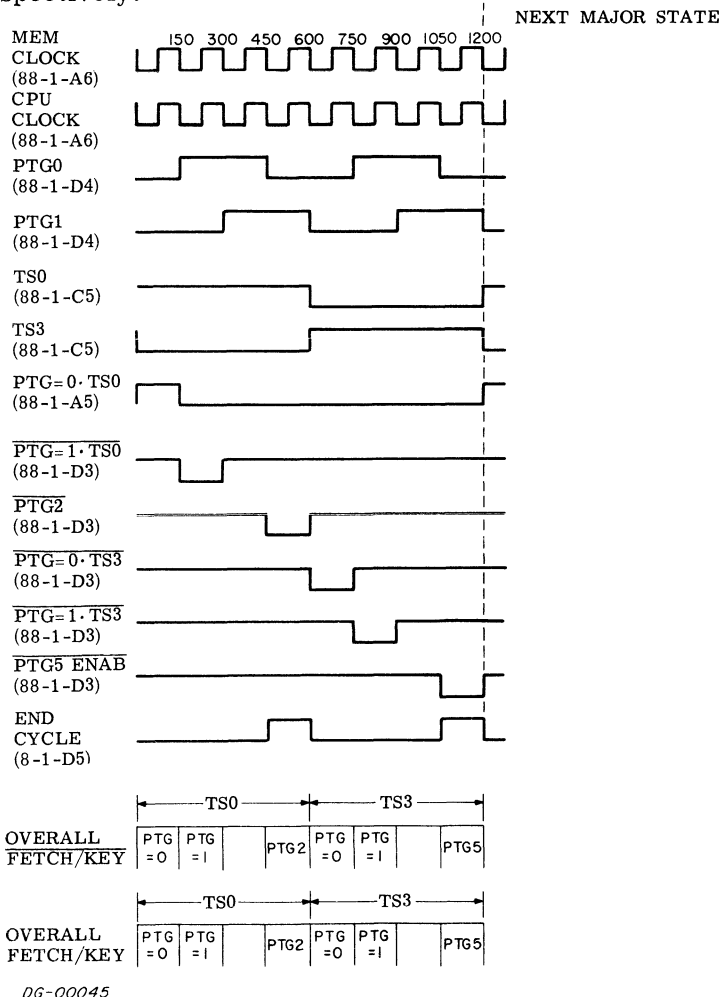


Figure C-1 Timing For The Processor Timing Generator During All Major States Except Fetch or Key

DG-00045

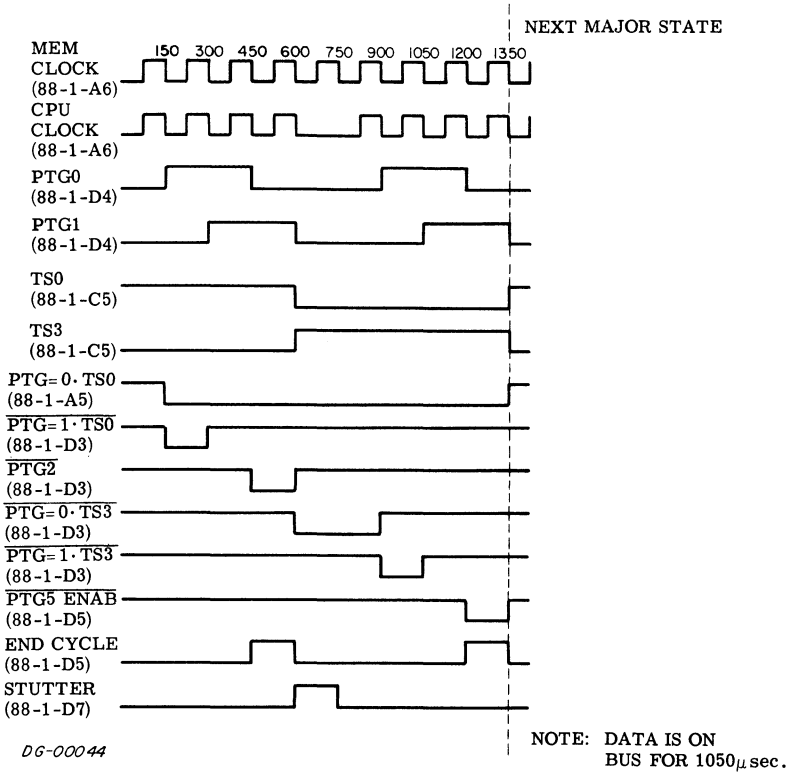


Figure C-2 Timing For The Processor Timing Generator During Fetch or Key

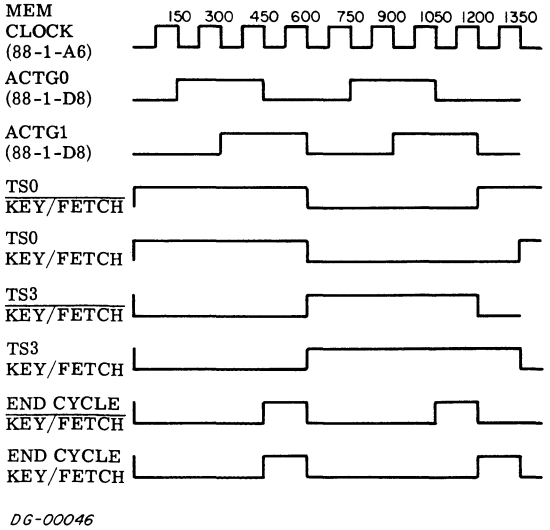
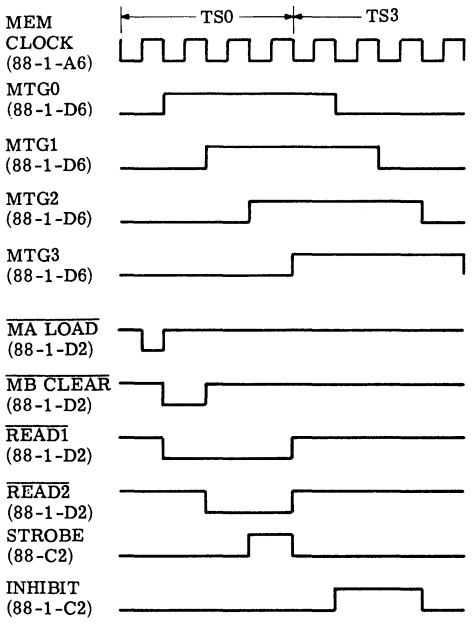


Figure C-3 Timing For The Accumulator Timing Generator

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

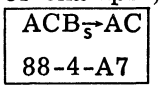
Data is transferred between memory and the central processor or an I/O device along three data buses called:

- $\overline{\text{MEM}}$ 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



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

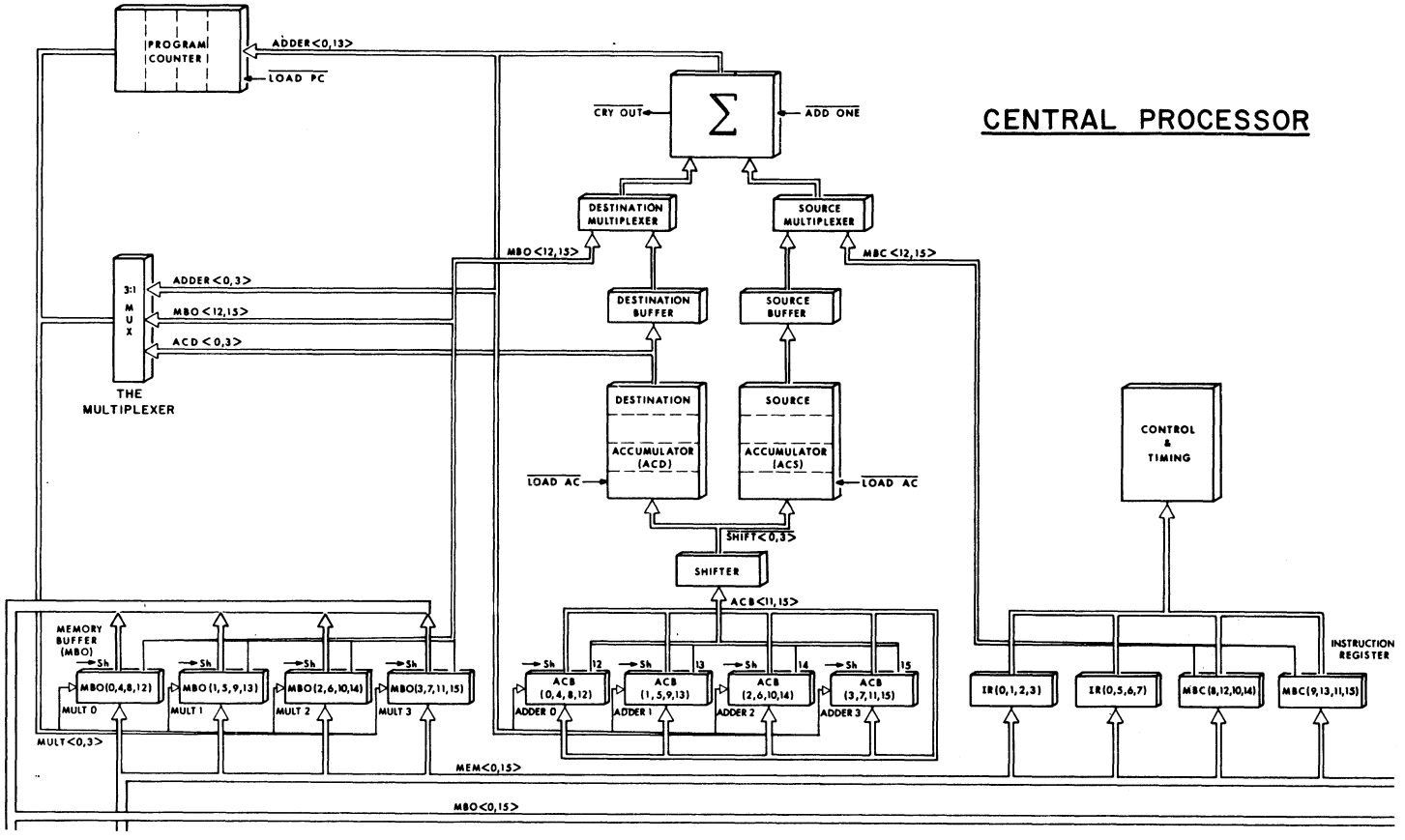
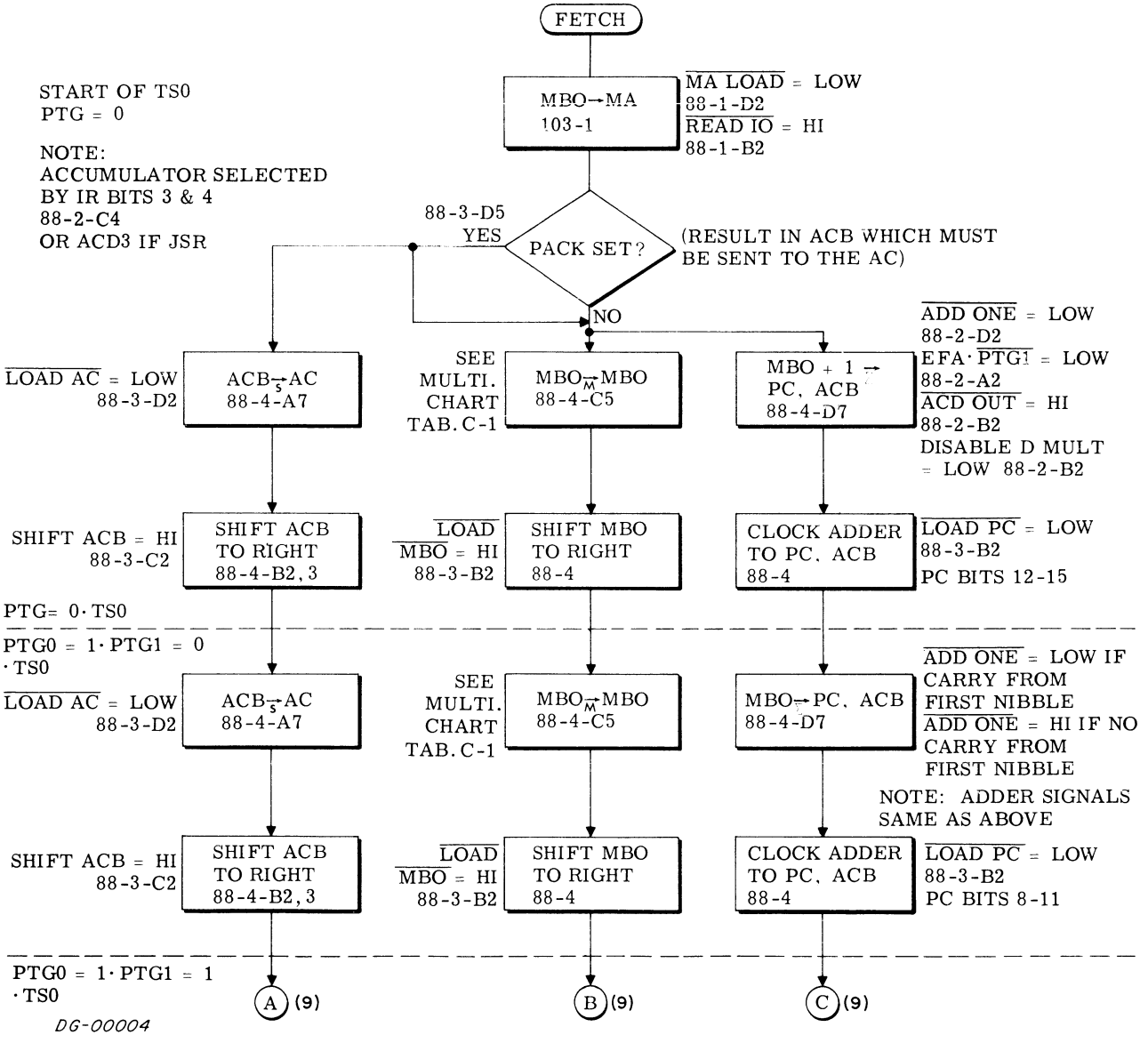
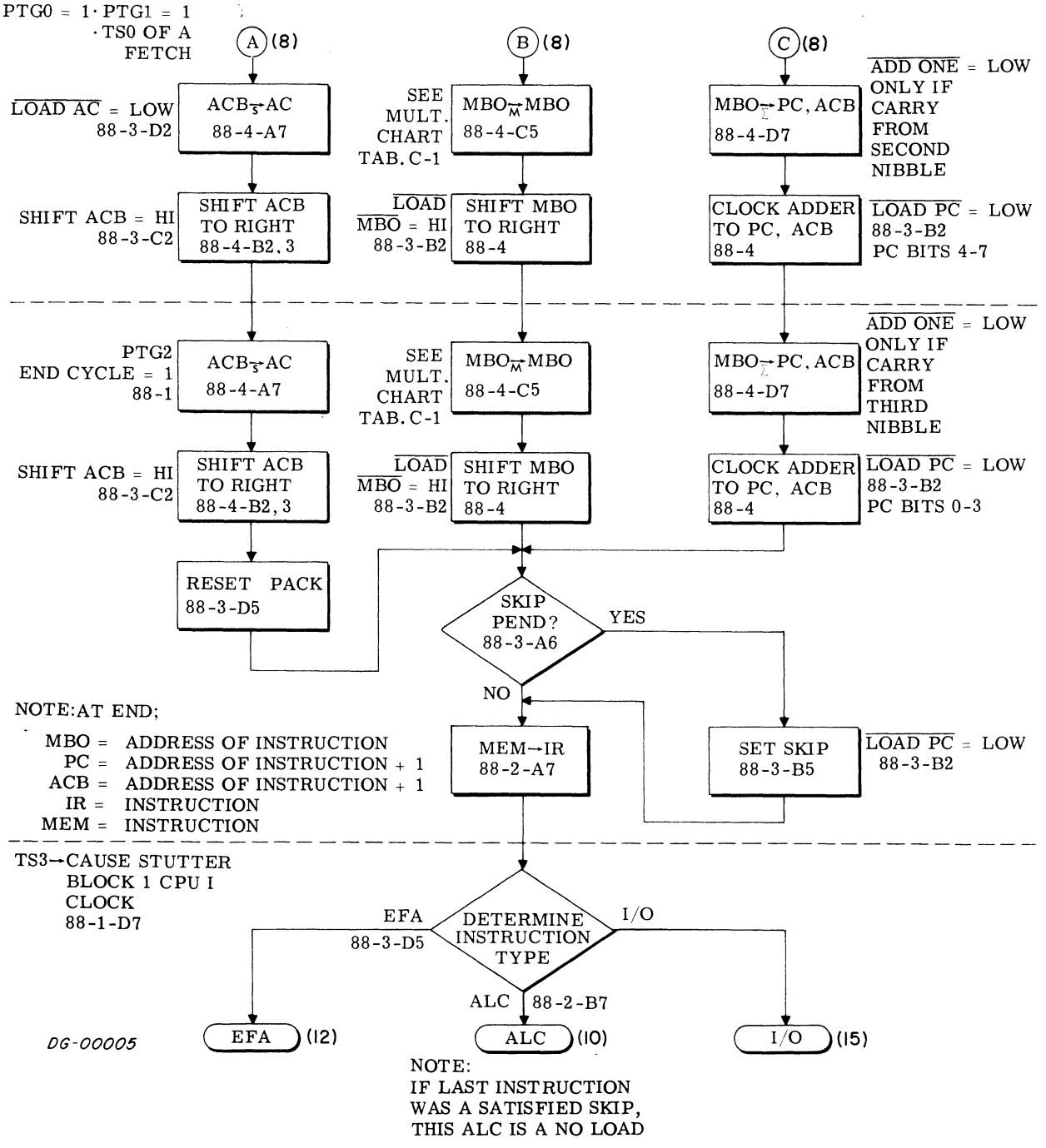
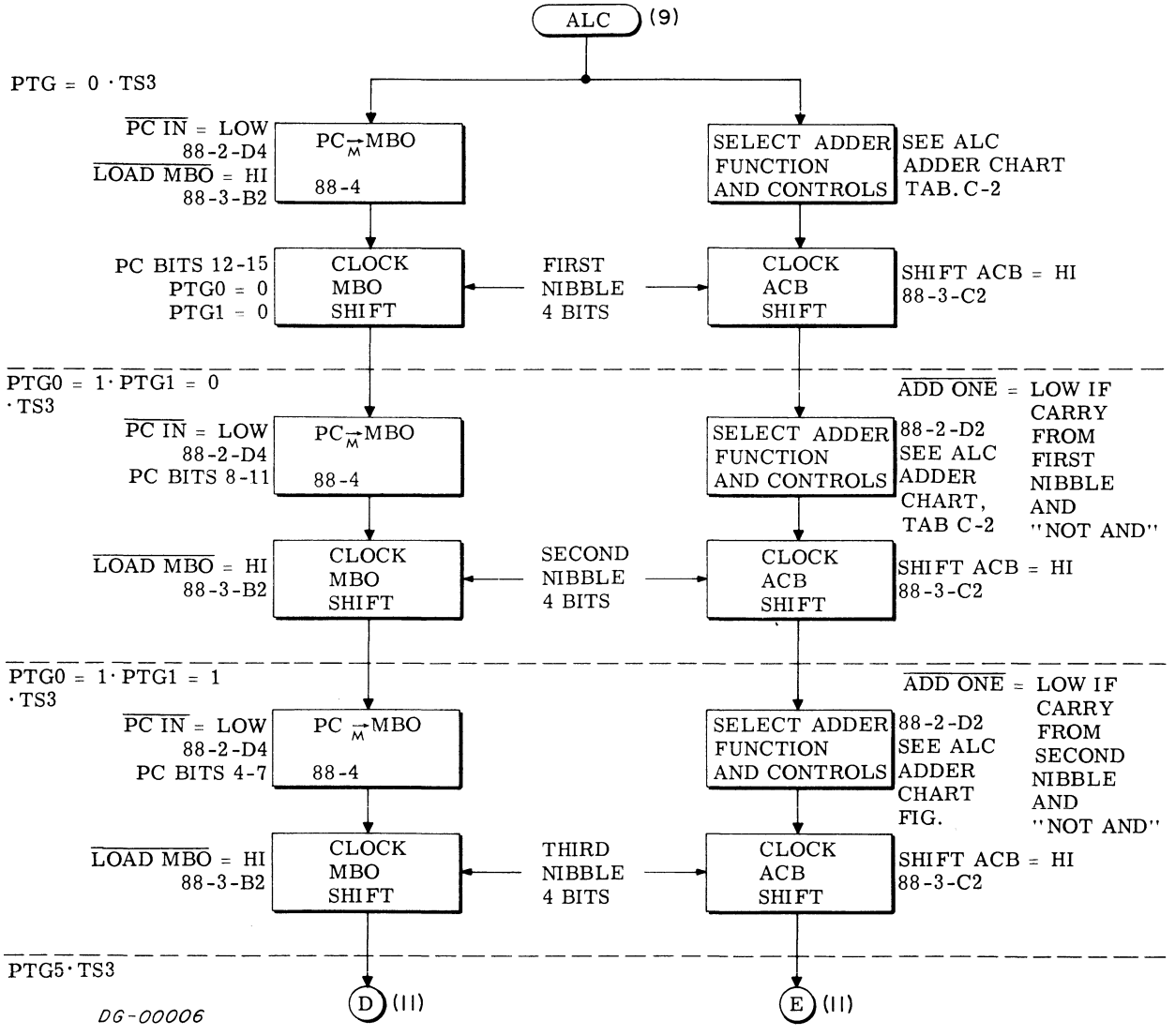
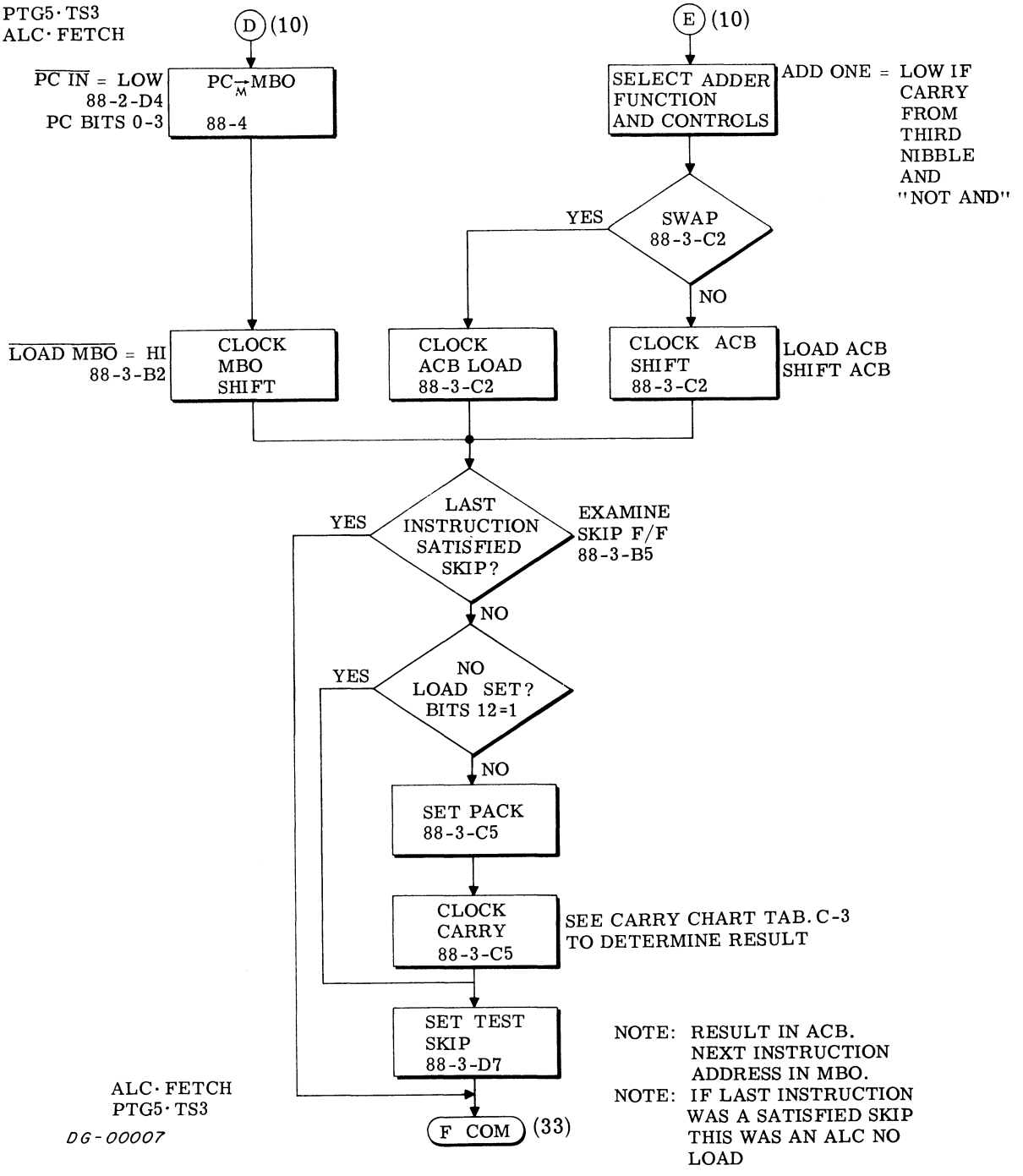


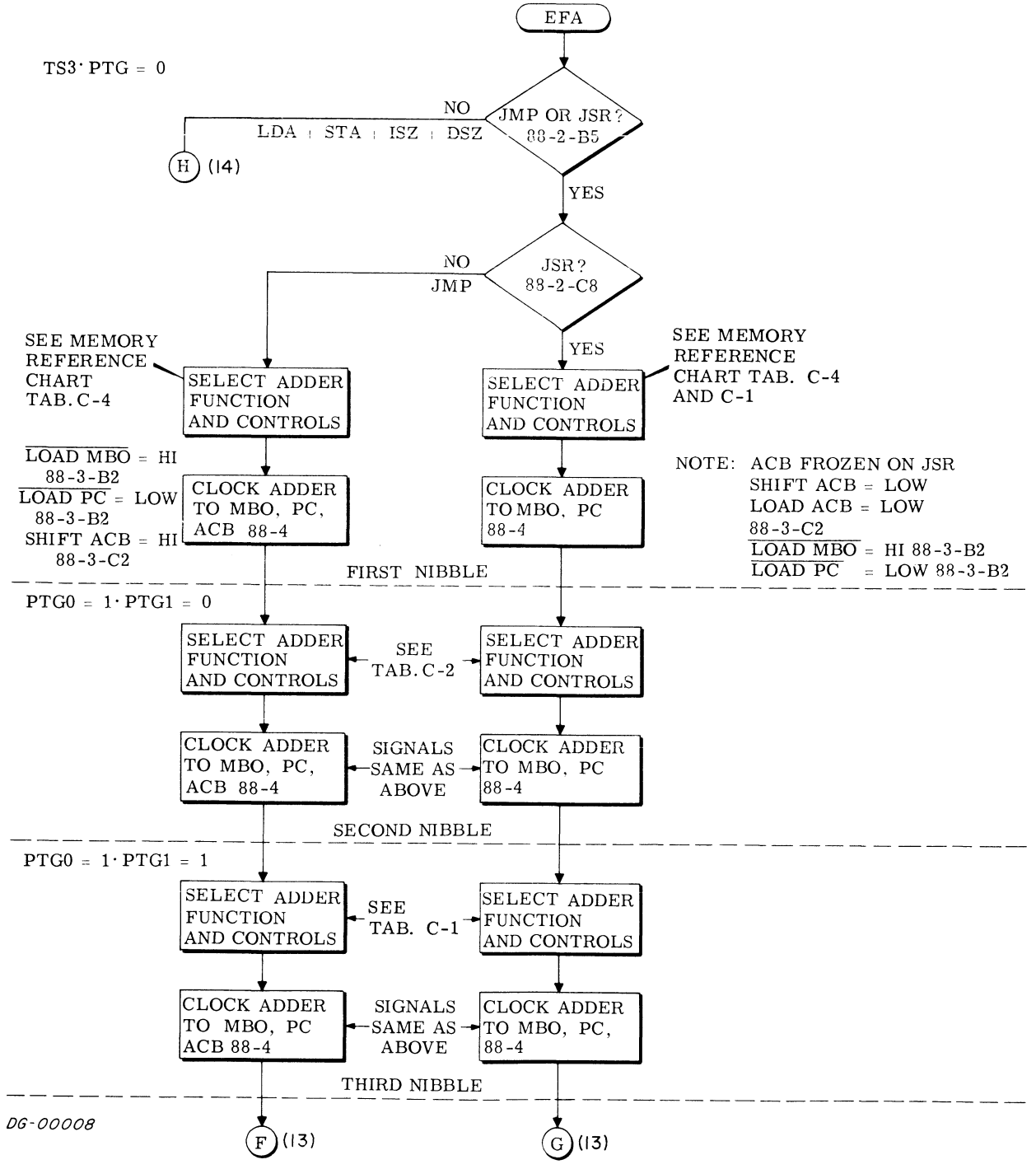
Figure C-5 The Nova 1220 Central Processor

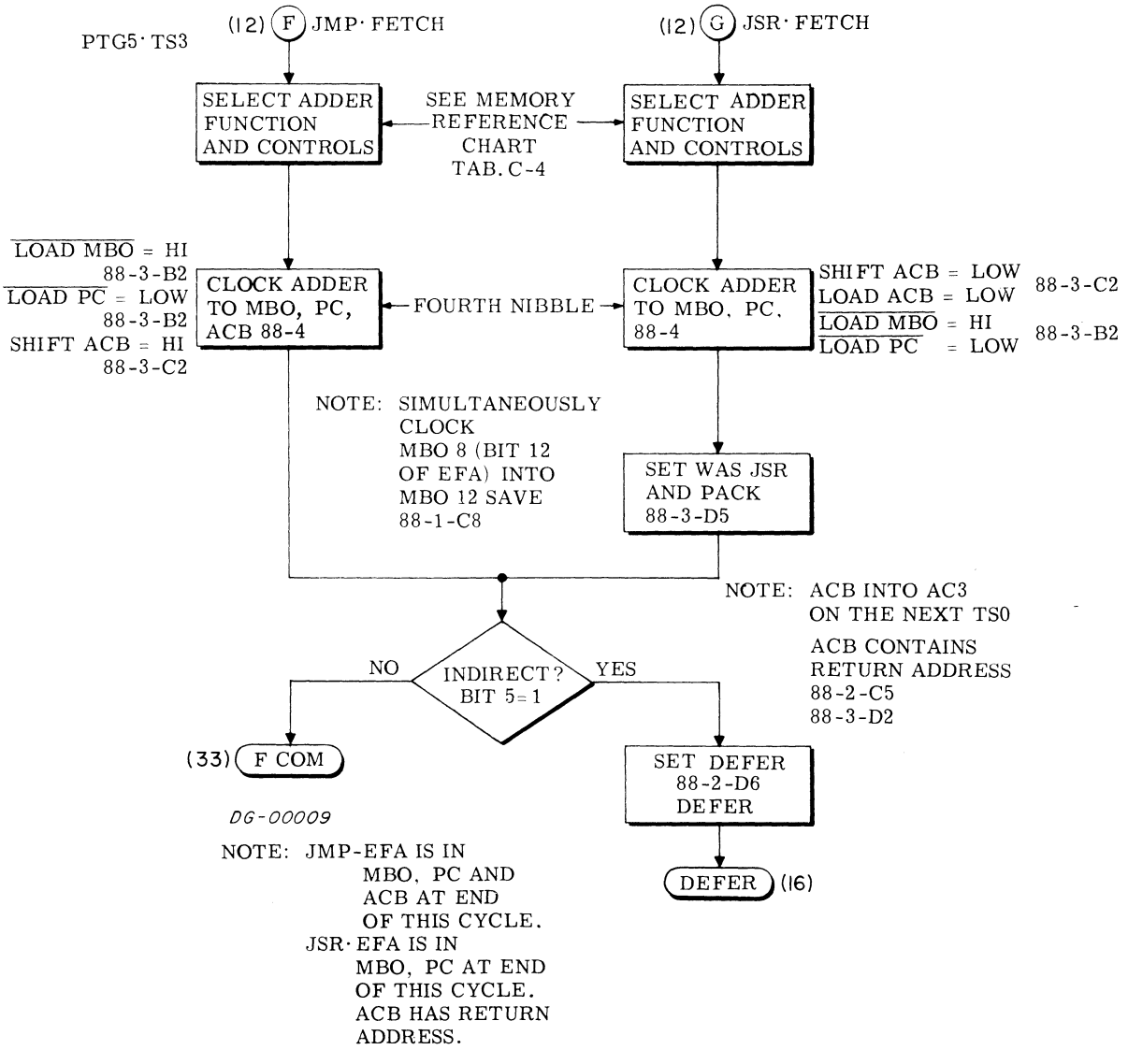


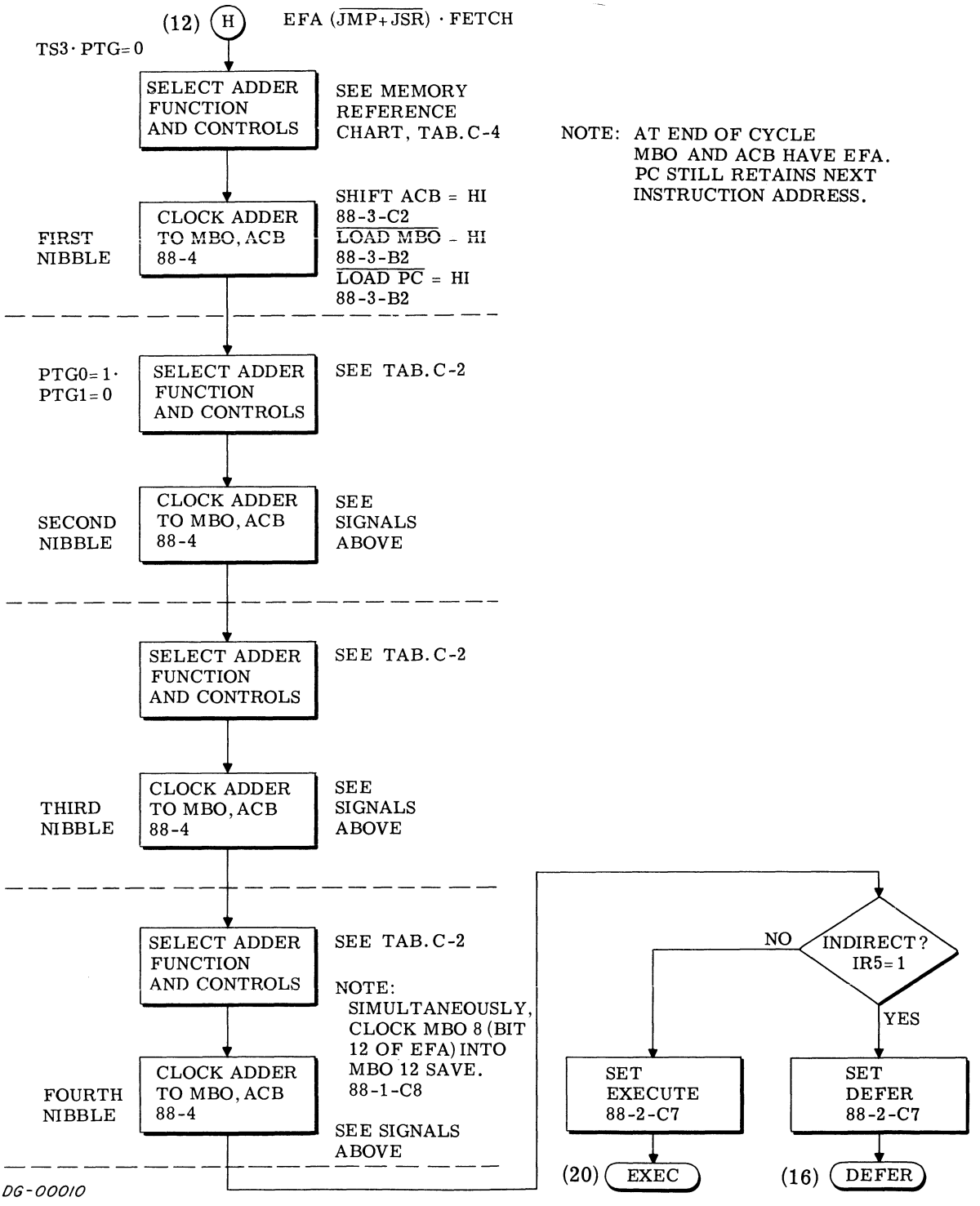


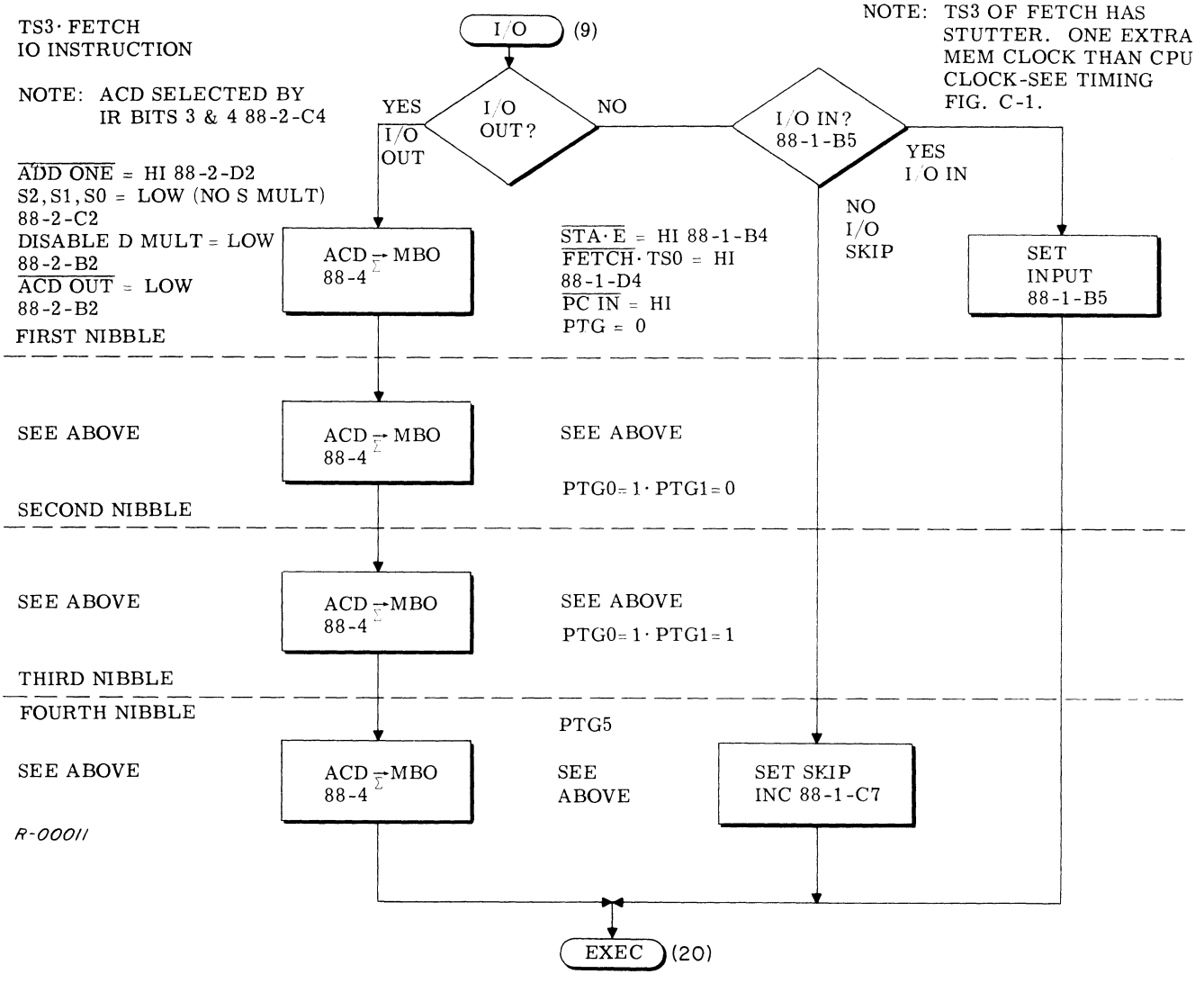


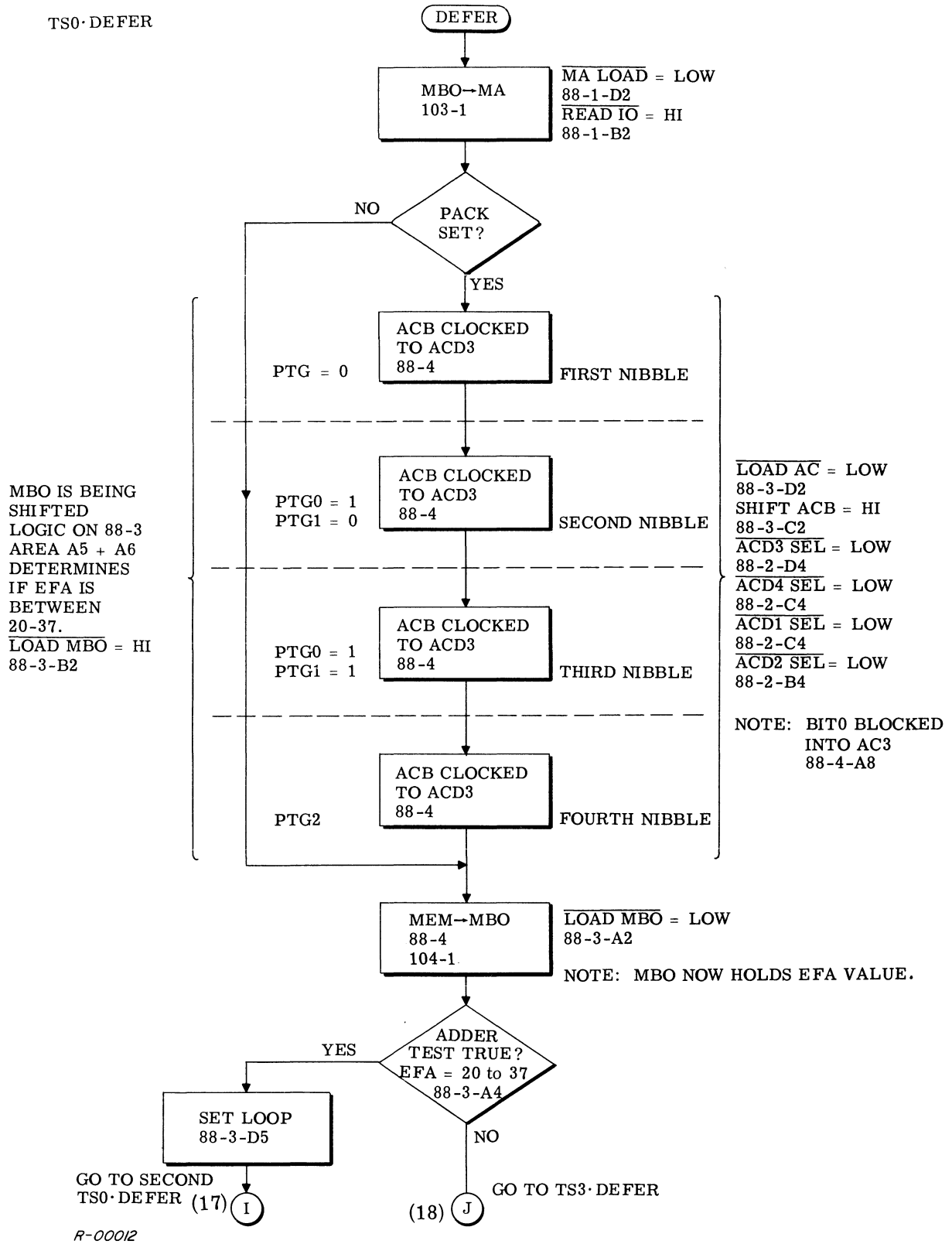


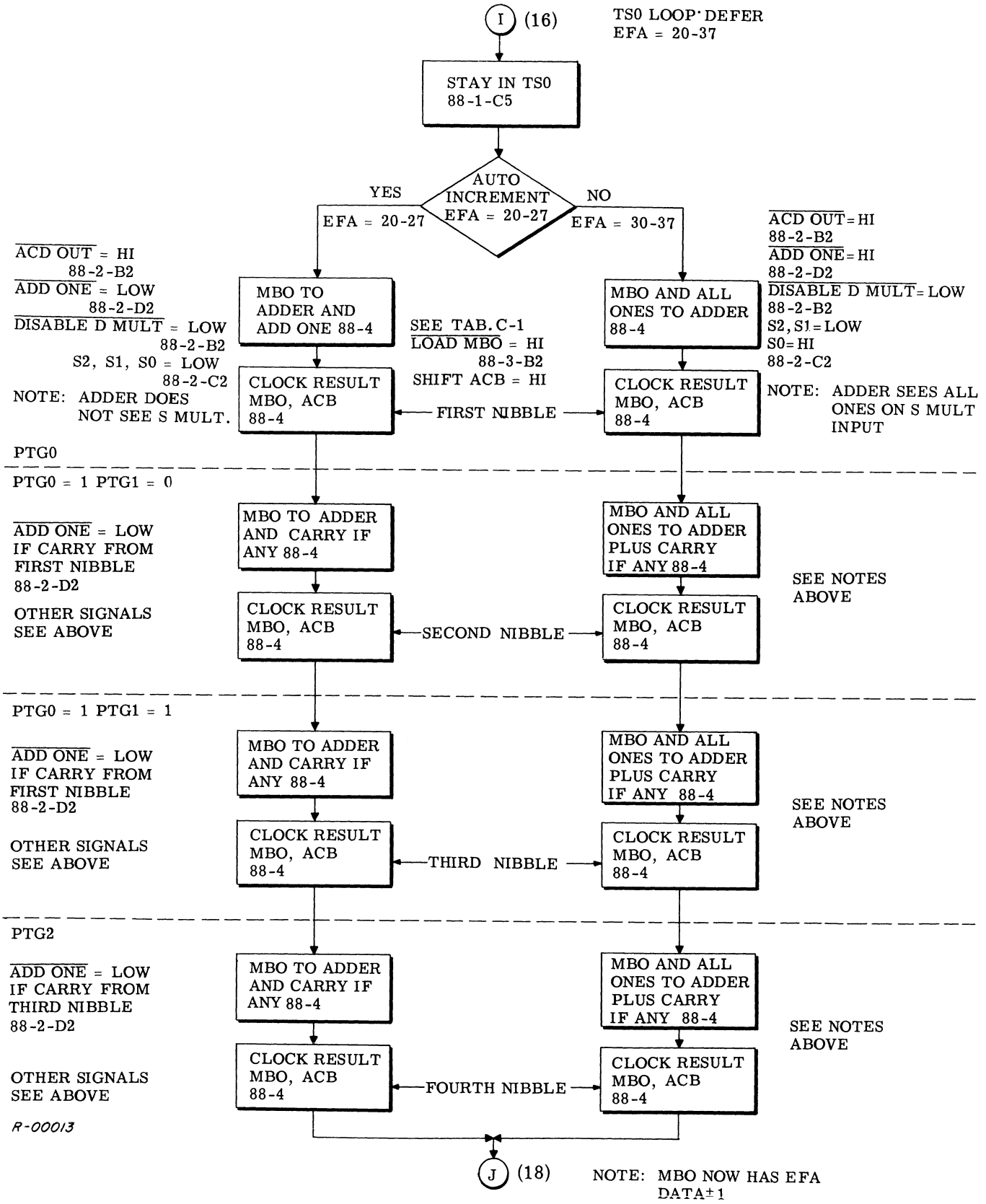


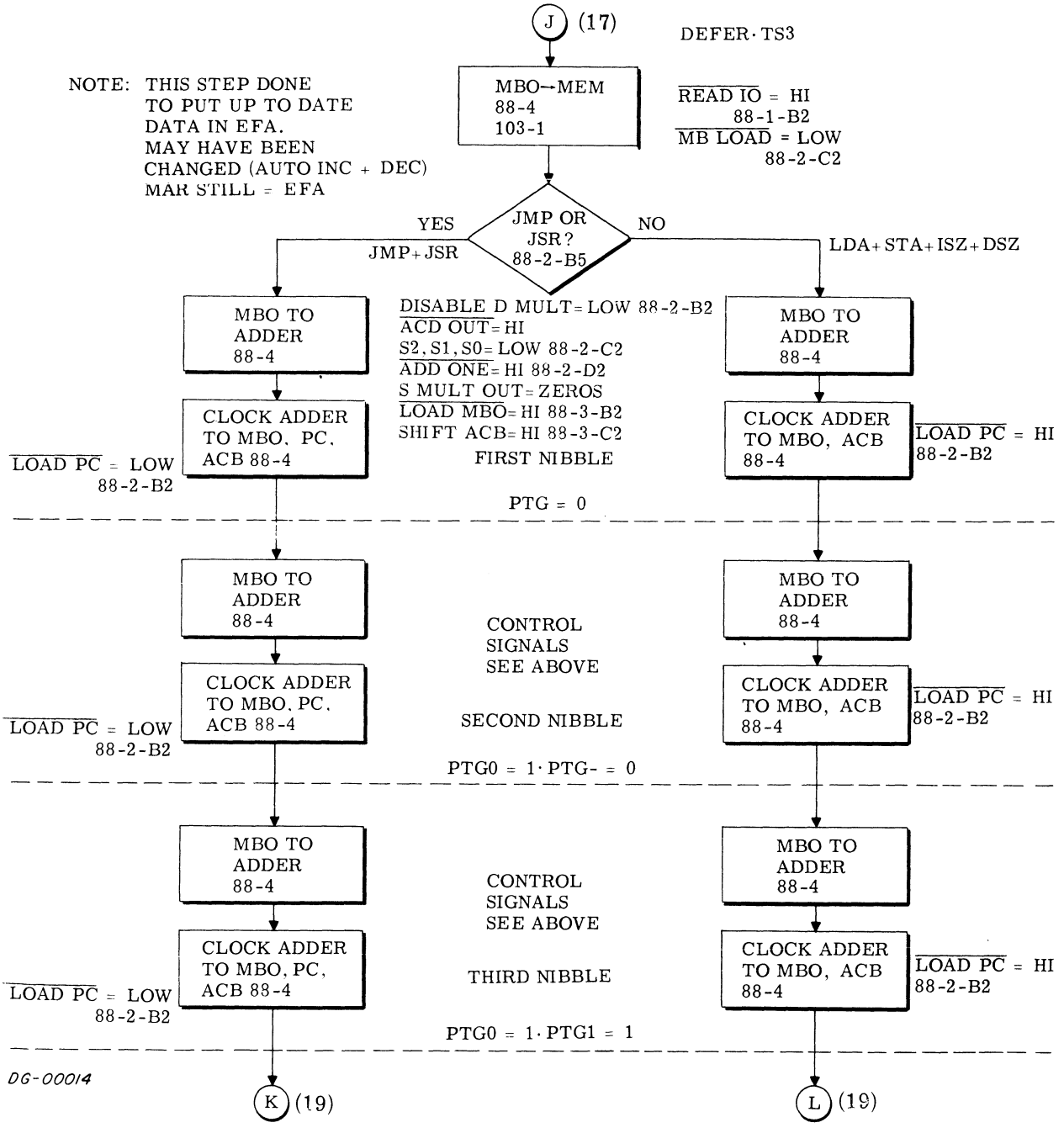


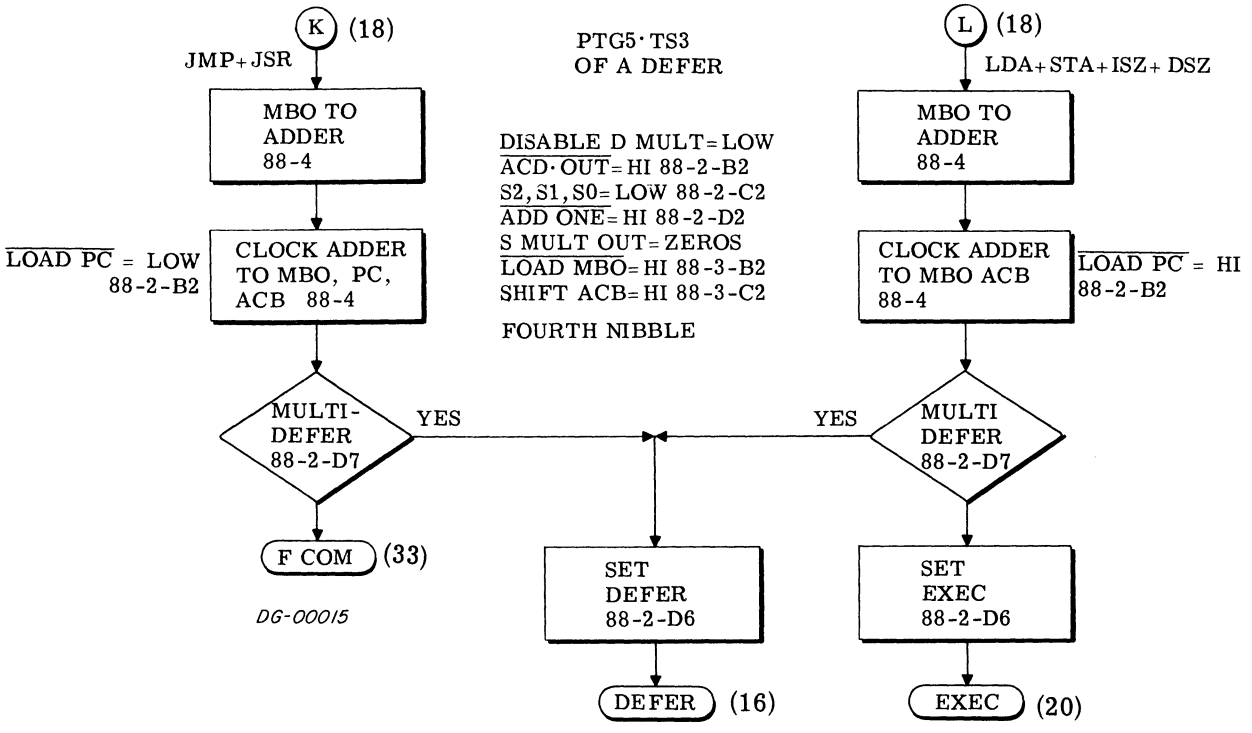


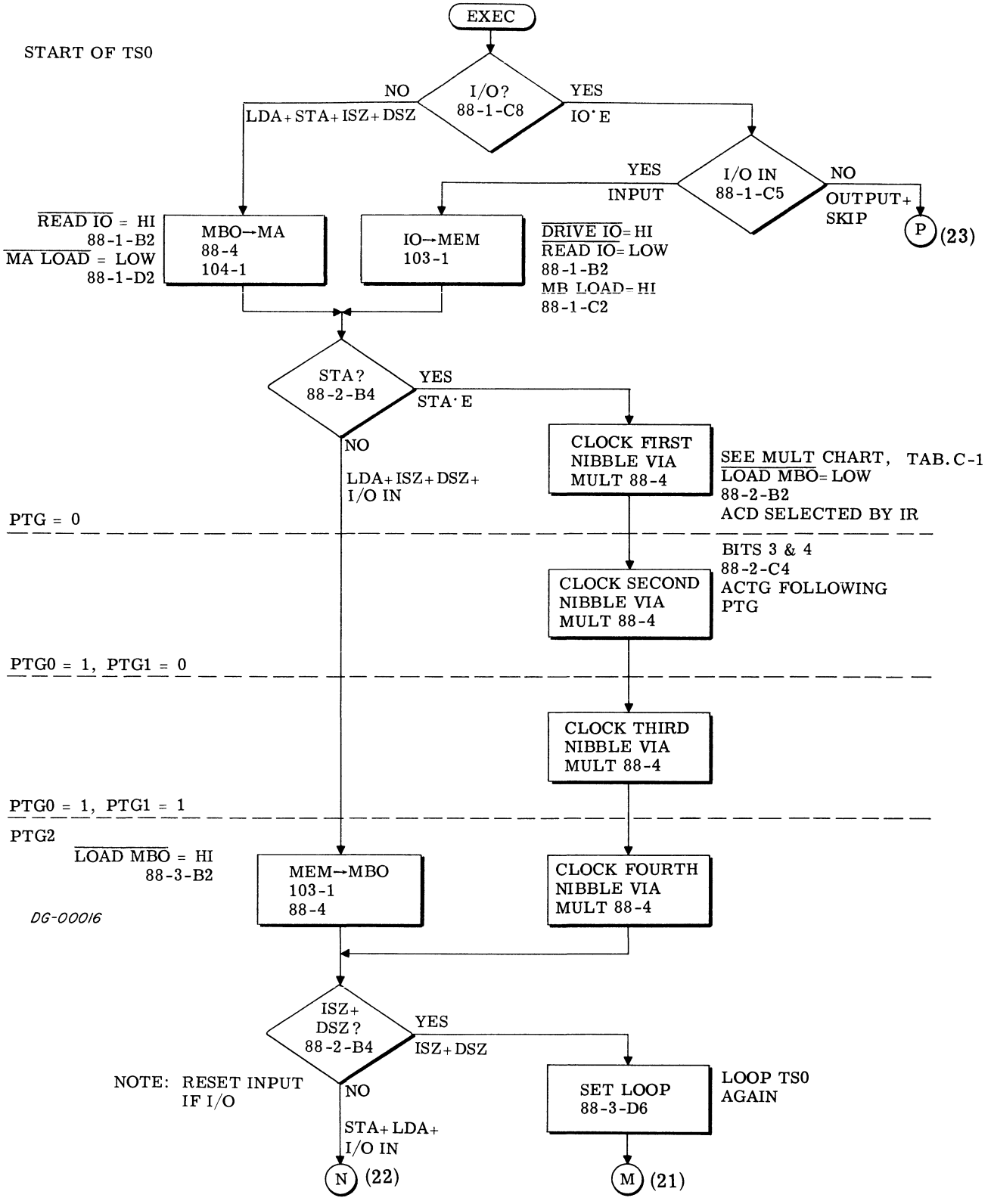


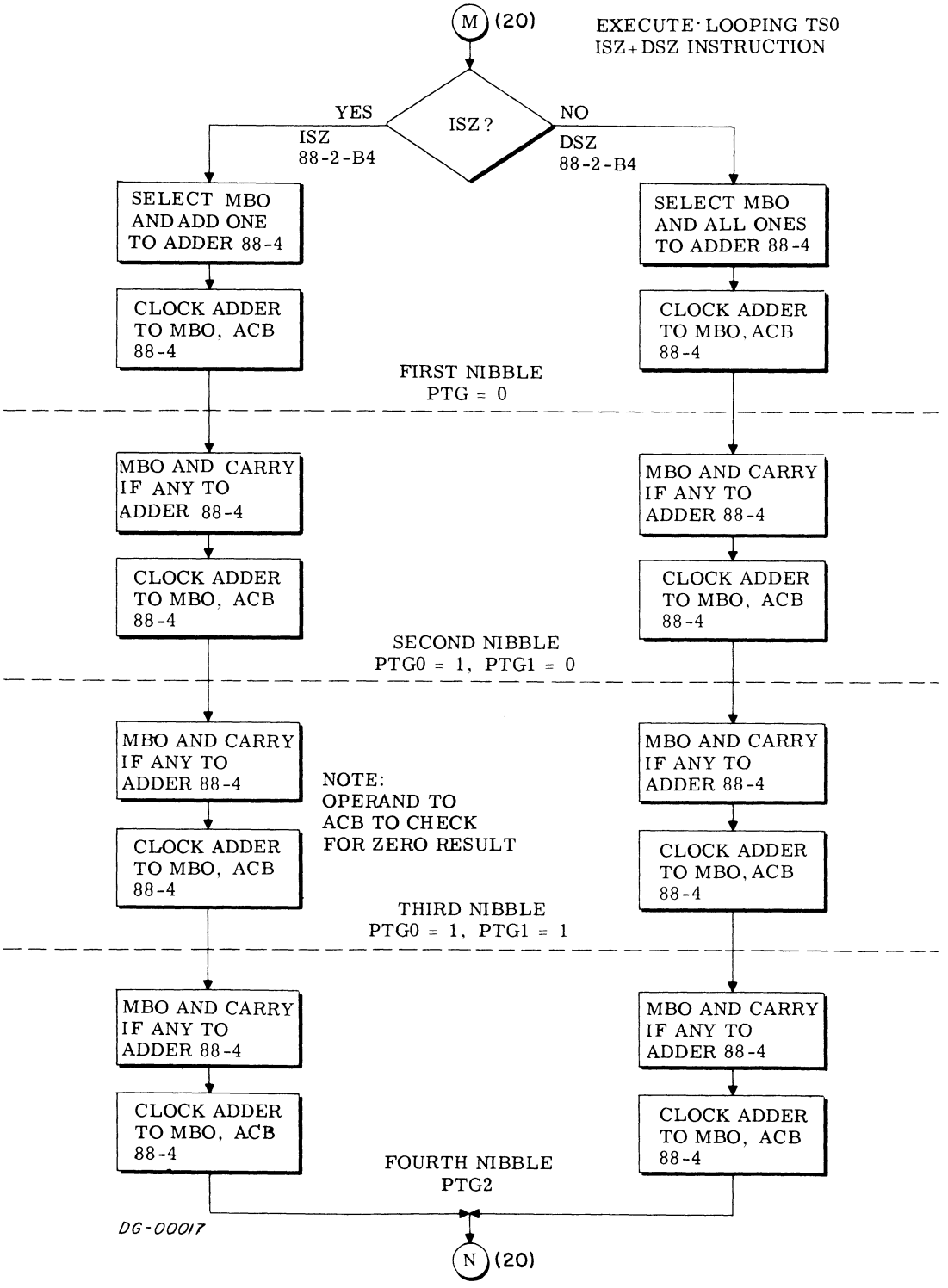


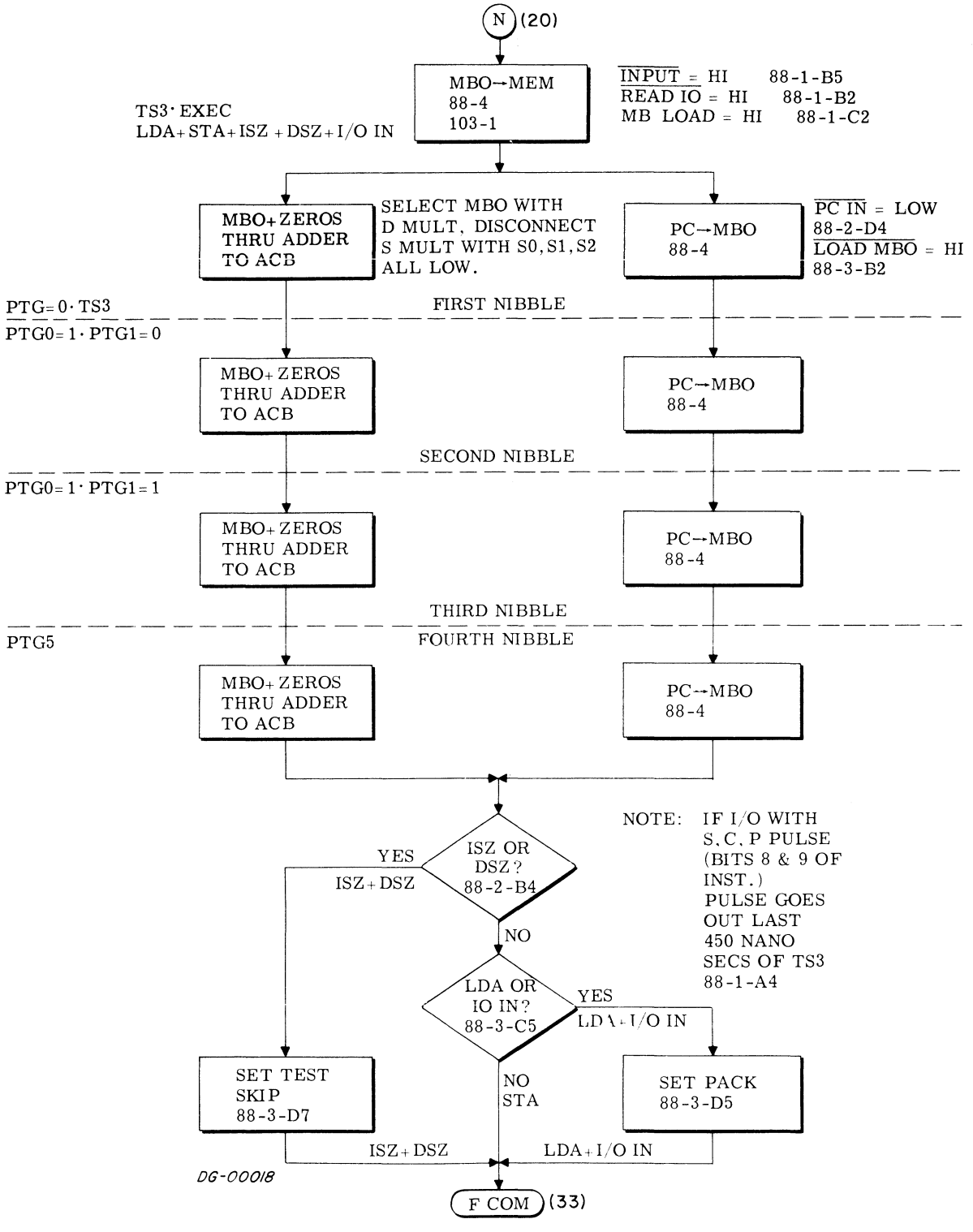


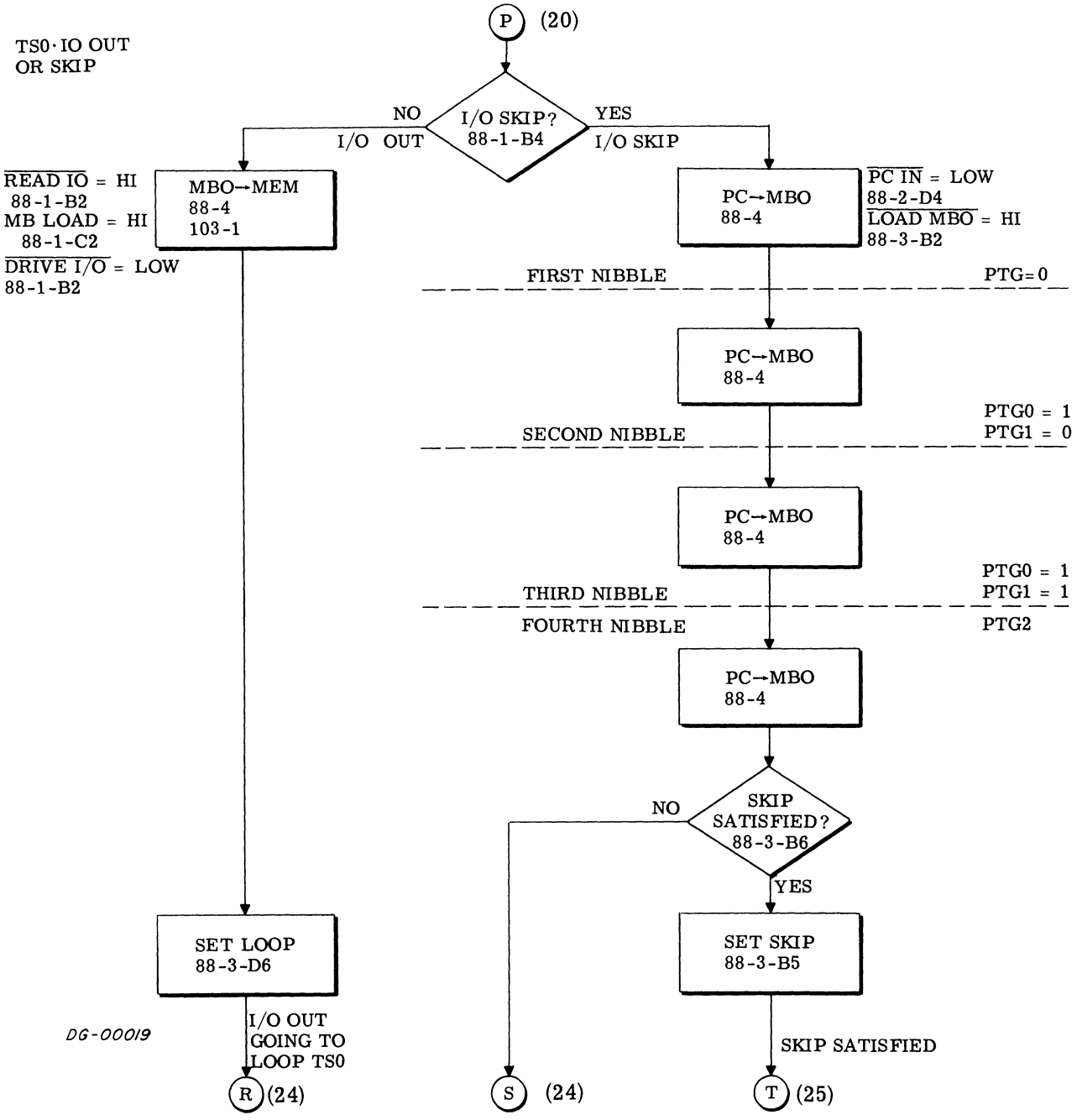


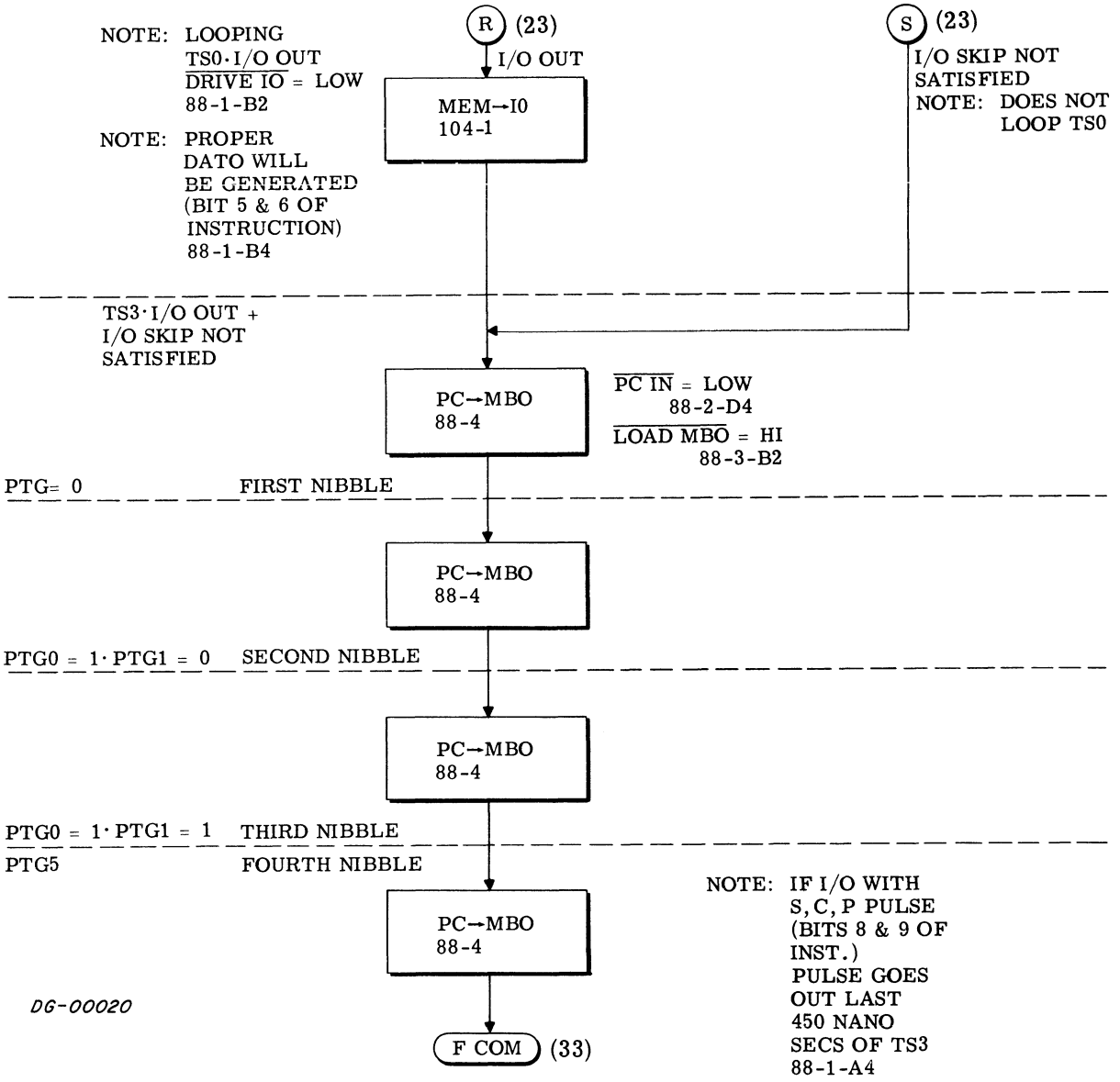


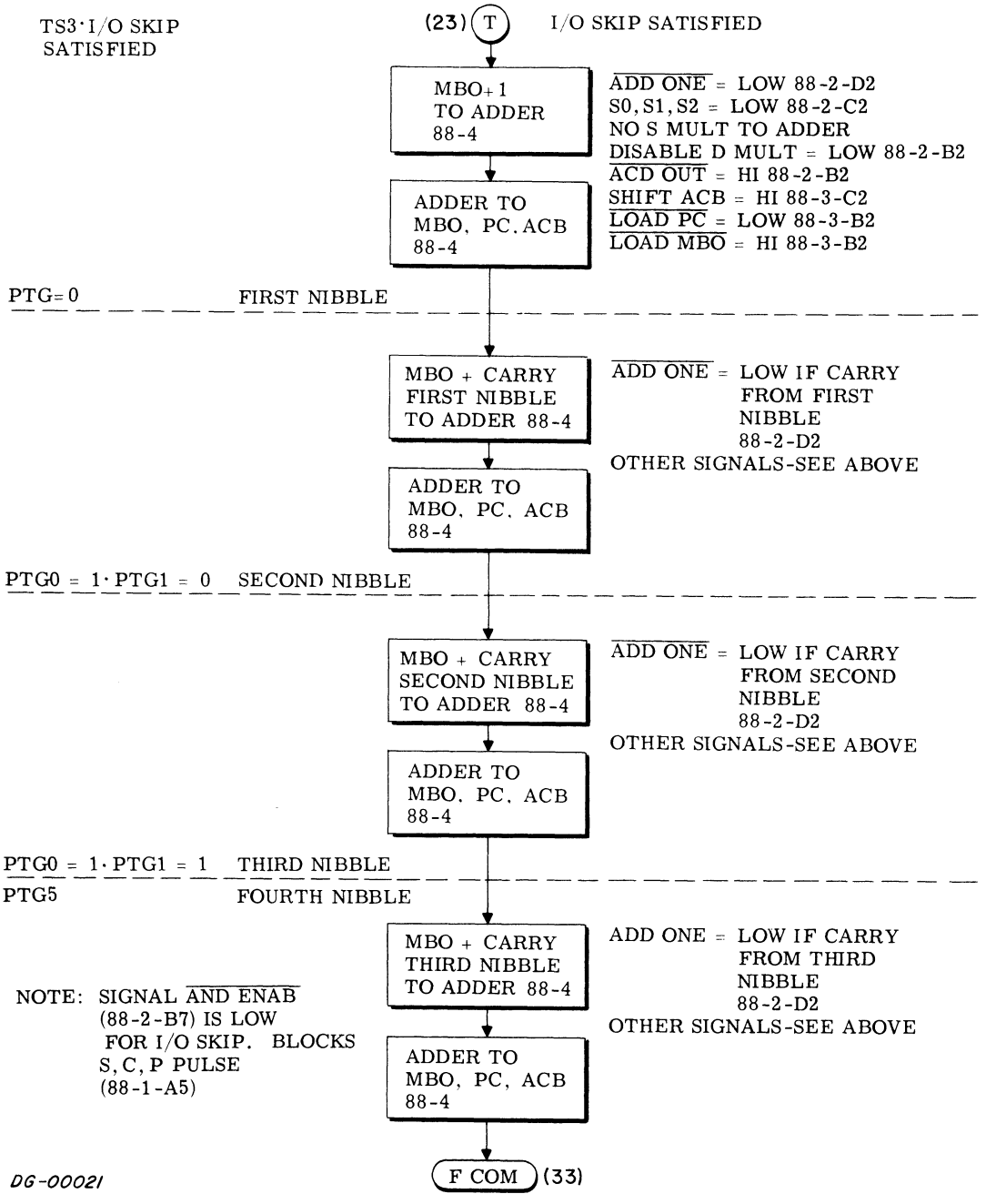


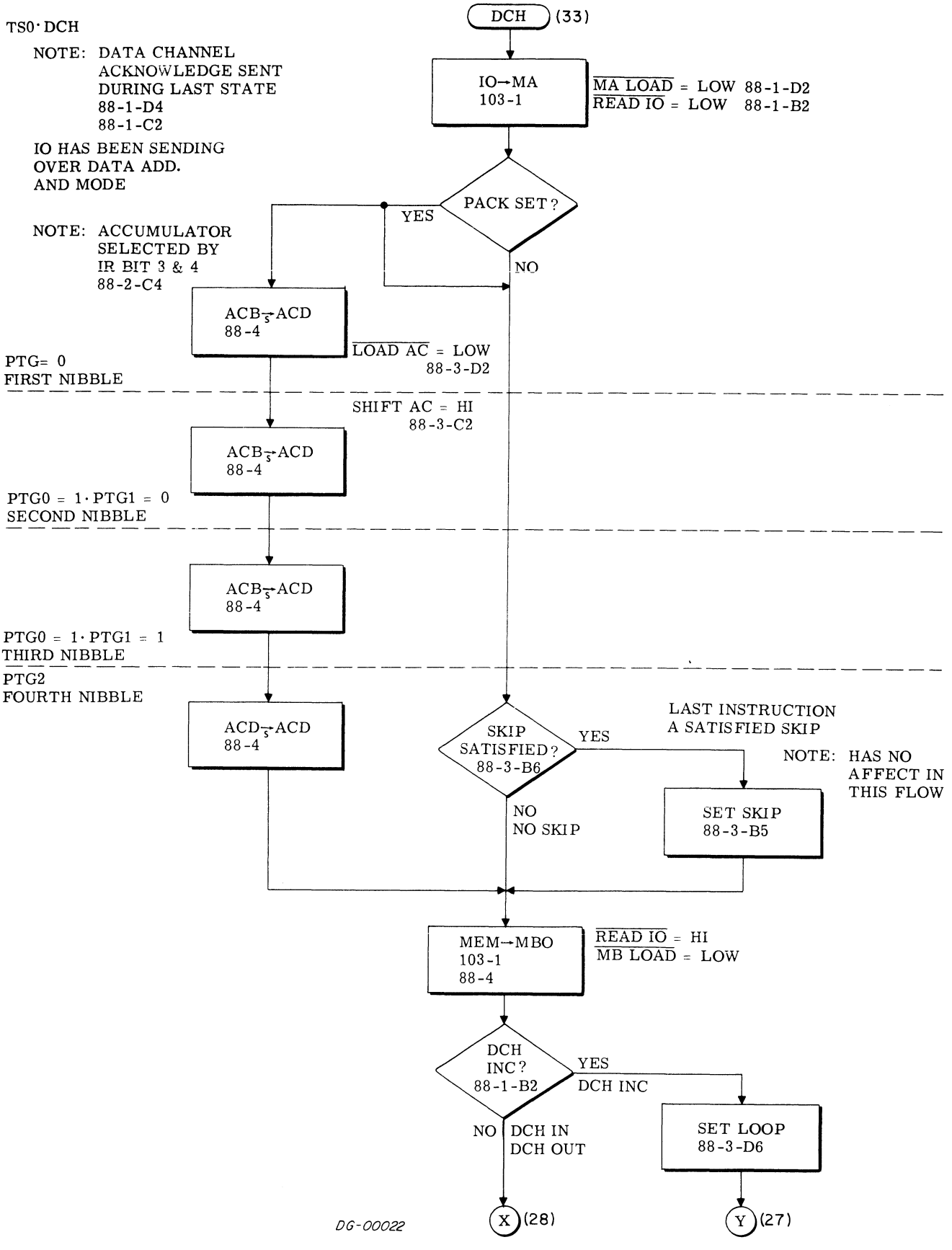


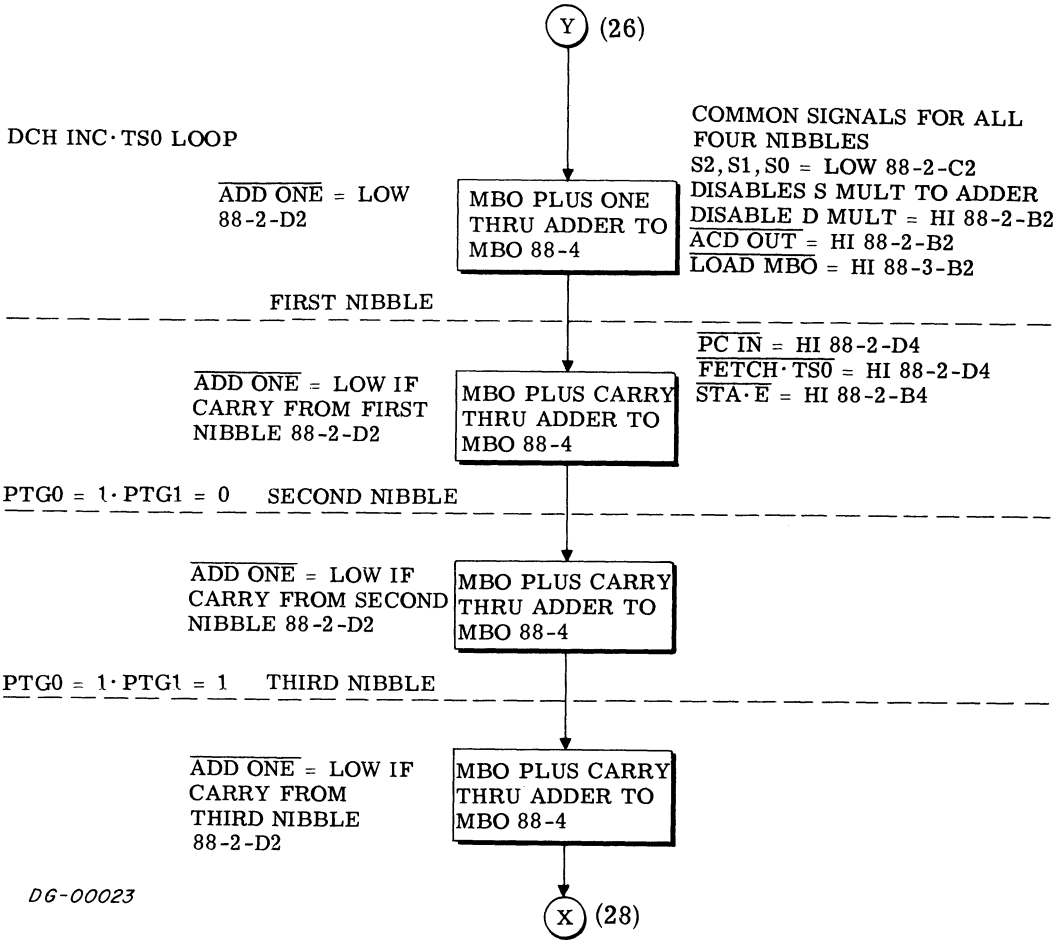


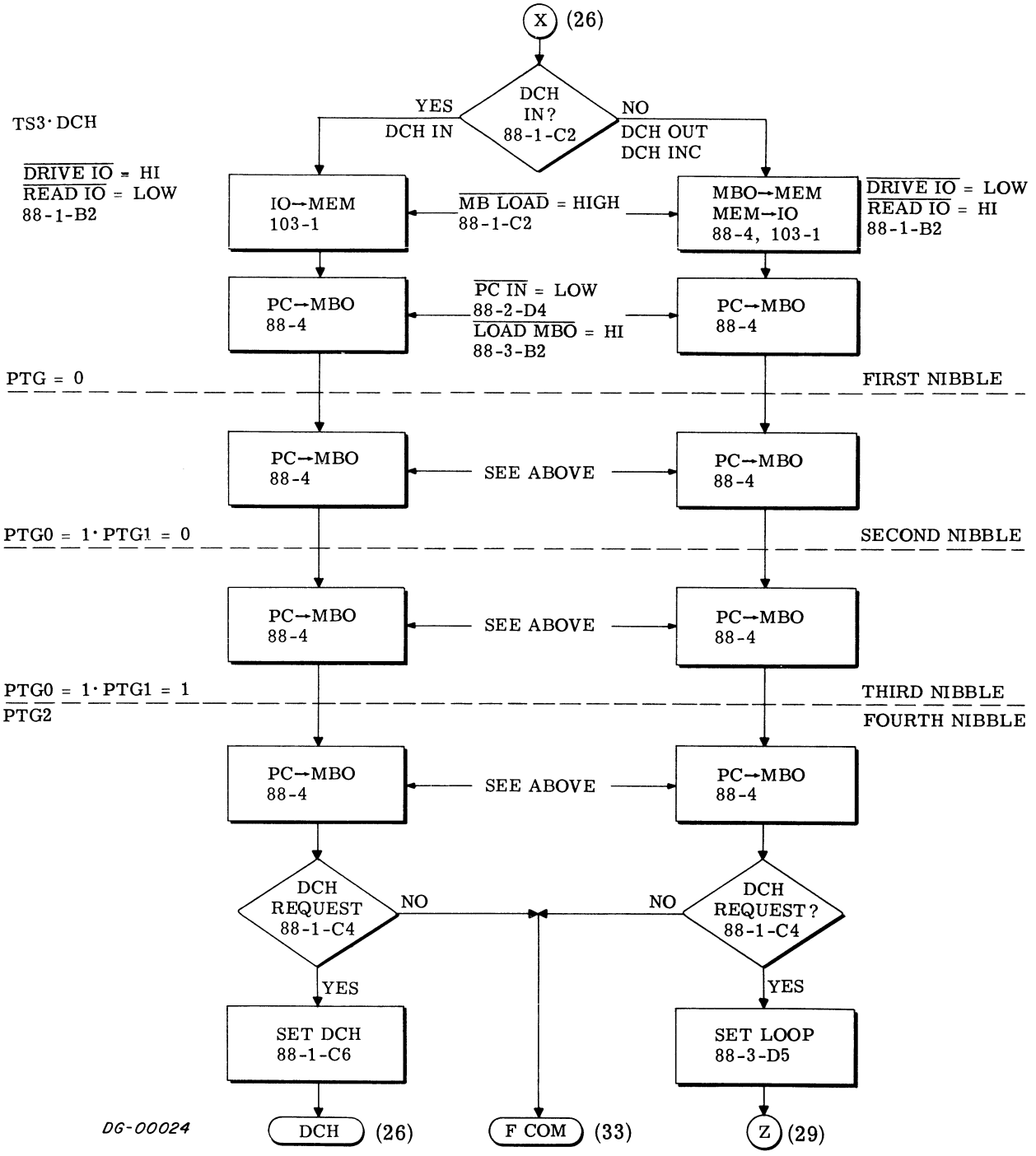


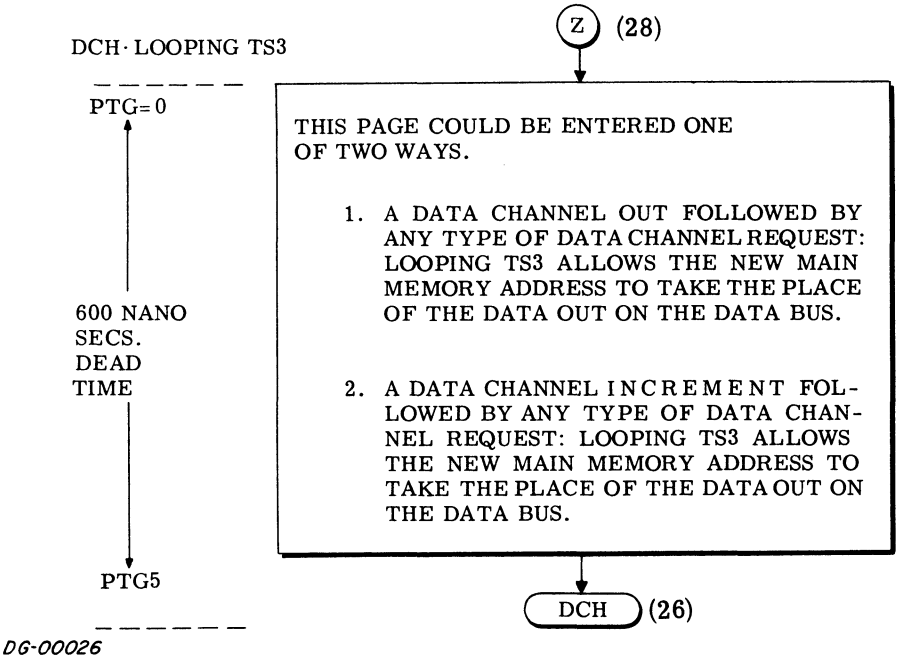


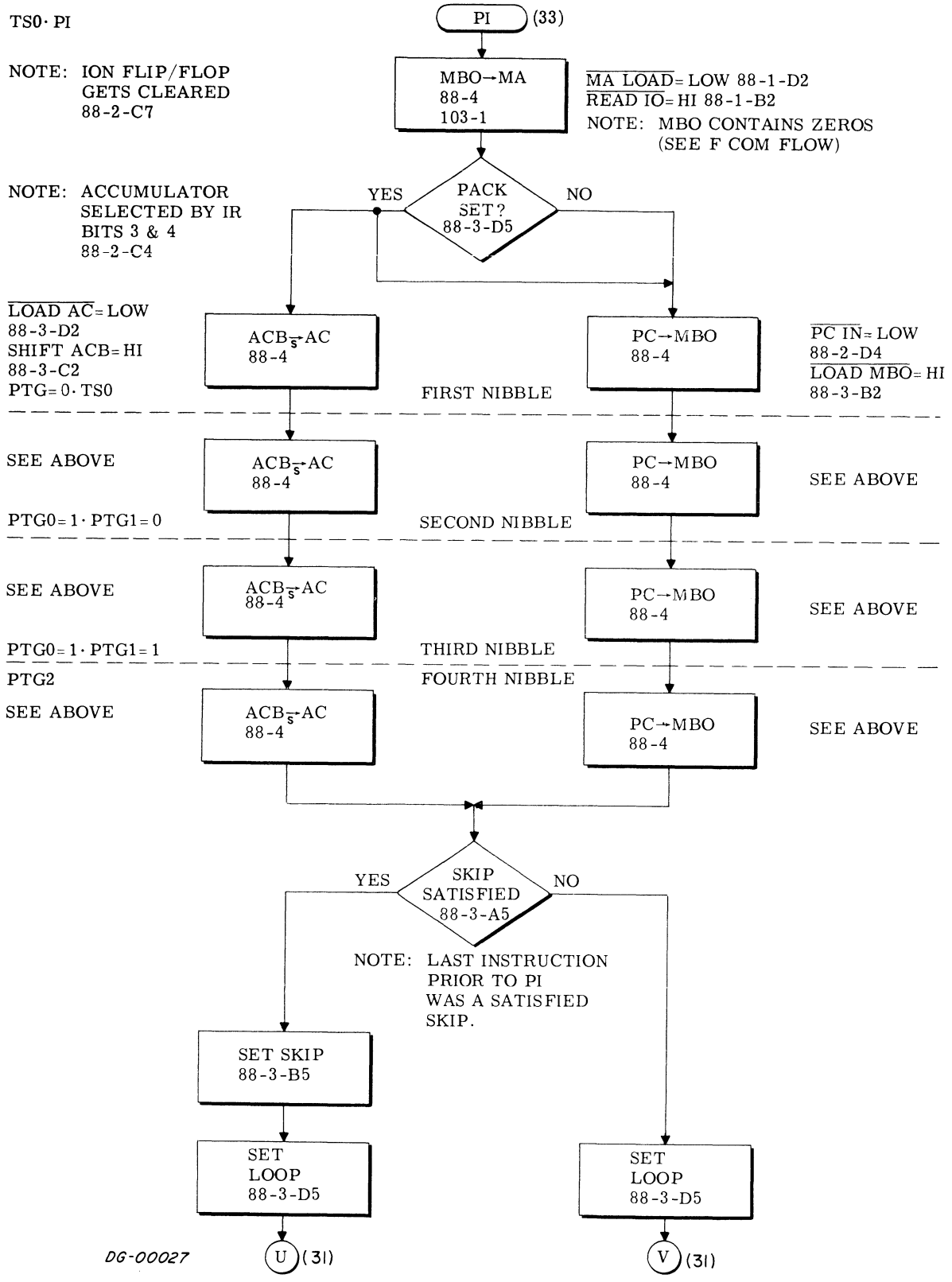


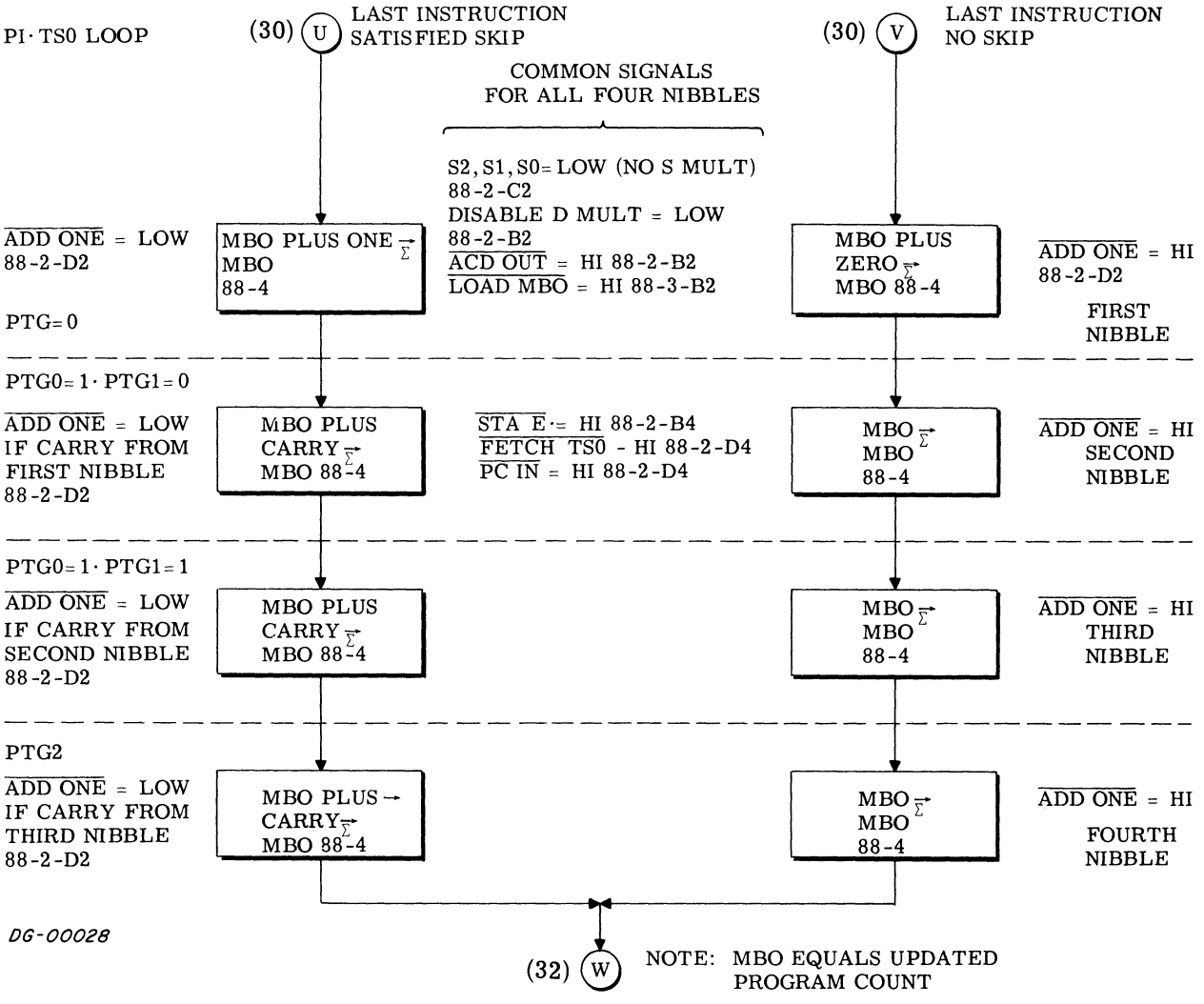


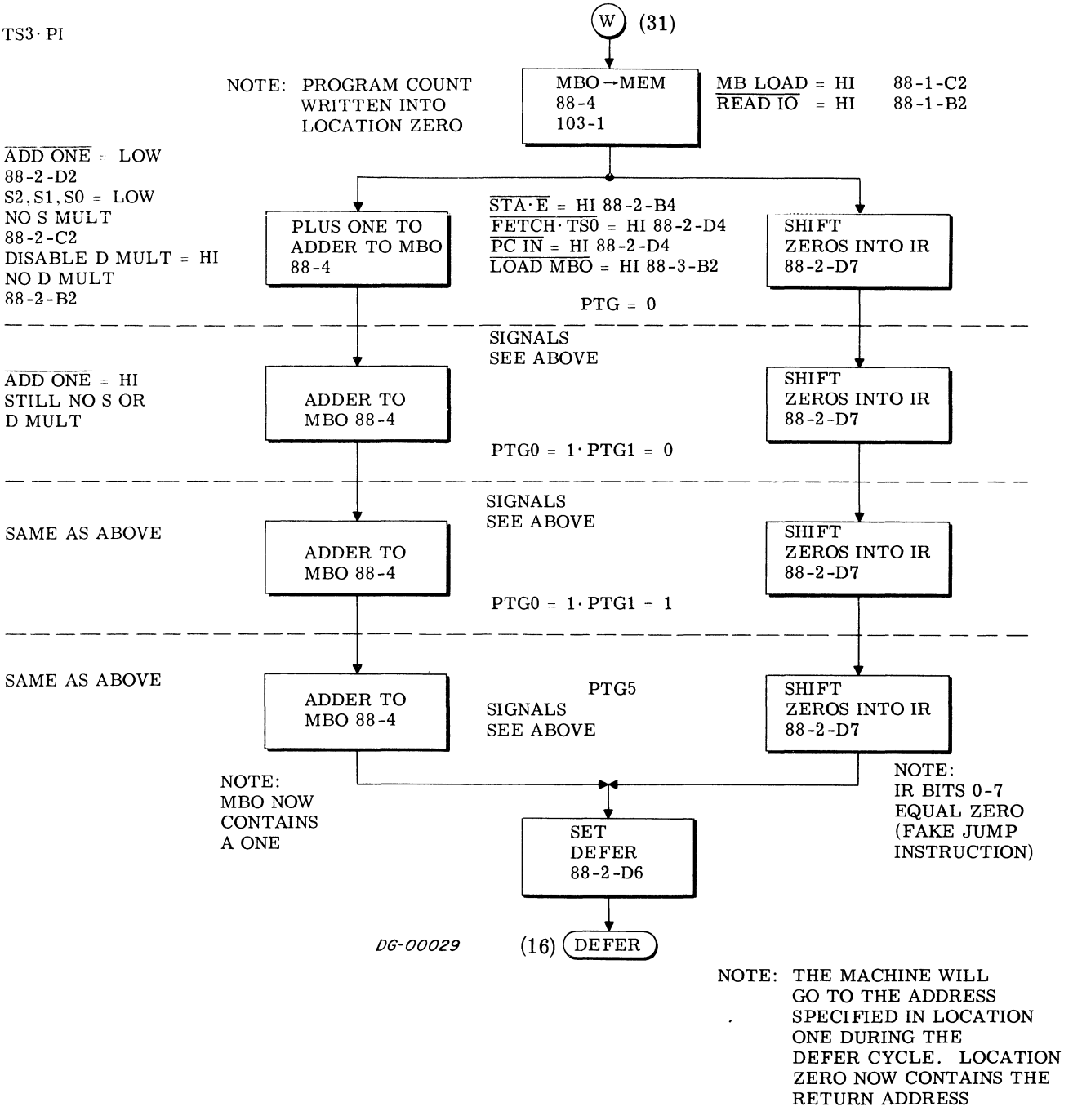












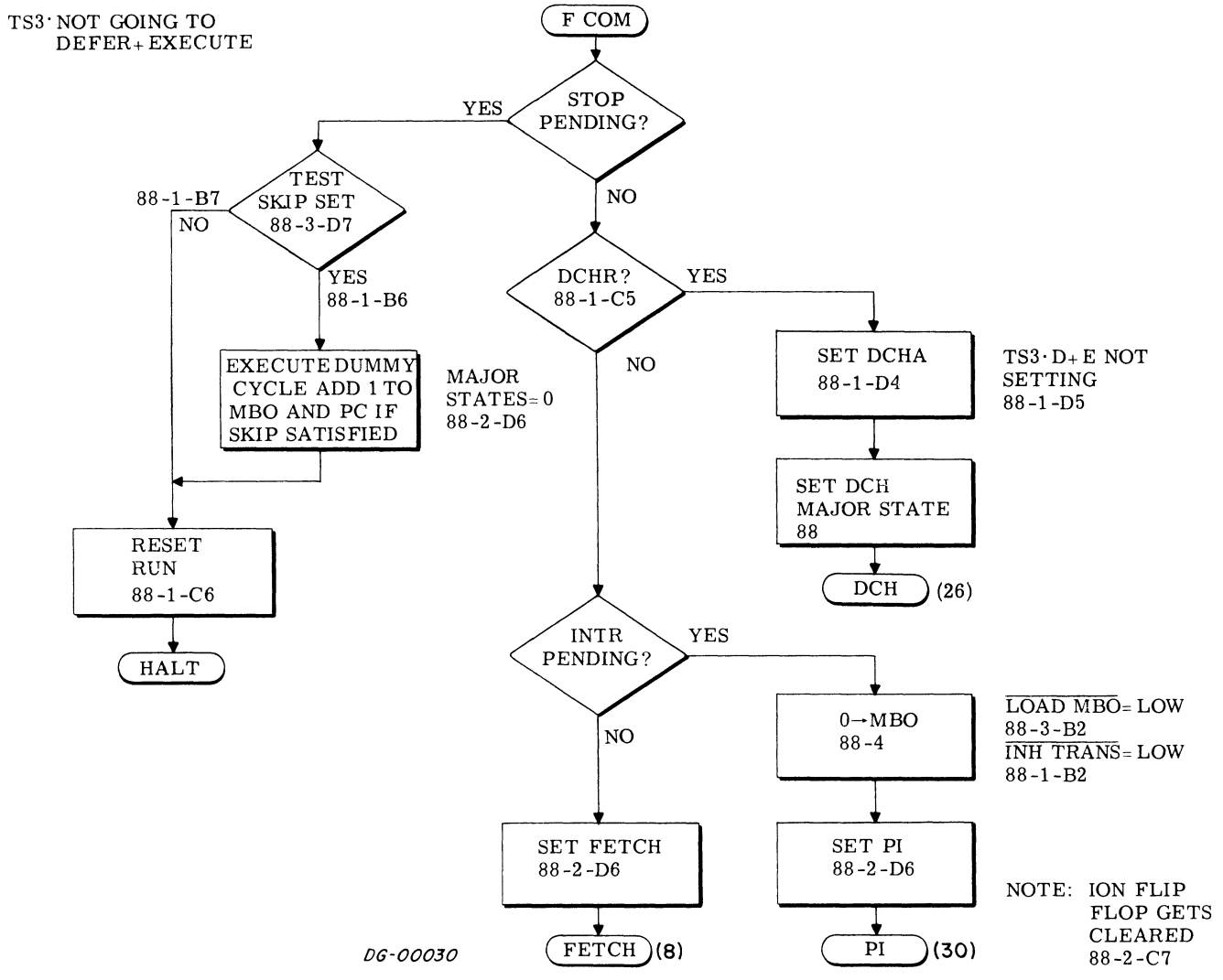


Table C-1

Adder and Multiplexer Control Signals During EFA Instructions

	S0 *	S1	S2	DISABLE D MULT	EFA· $\overline{\text{PTG1}}$ *	$\overline{\text{ACD}}$ $\overline{\text{OUT}}$
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) 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	$\overline{\text{ACD}}$ $\overline{\text{OUT}}$	EFA · $\overline{\text{PTG1}}$	IR6(1) =HI S0	S1	IR6(0) =HI S2	IR7(1) =LOW $\overline{\text{ADD}}$ $\overline{\text{ONE}}$
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

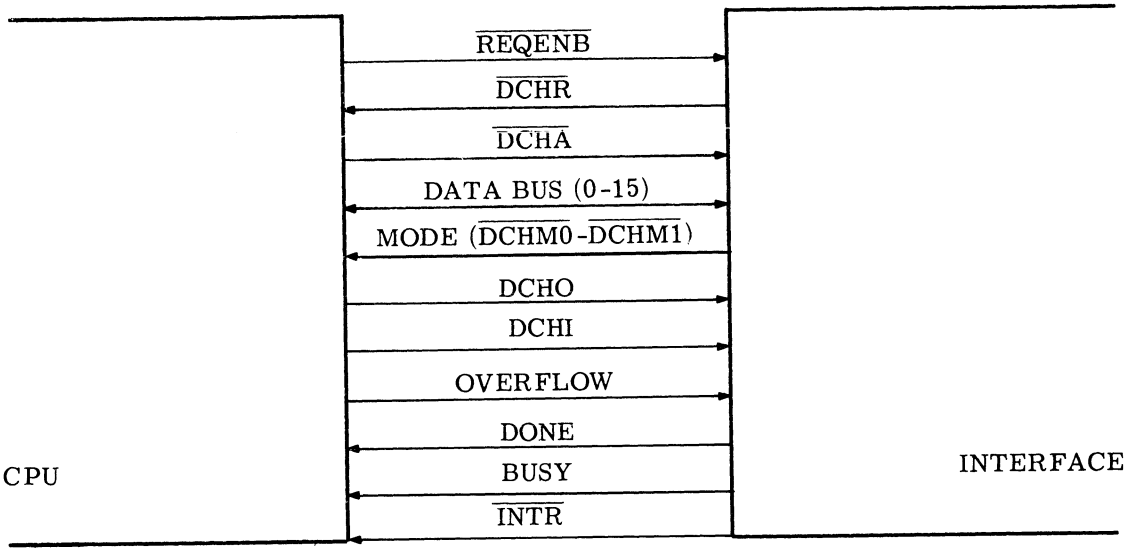
DG-00050

Table C-4

Memory Reference Instruction Decoding Chart

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

DATA CHANNEL SIGNALS



SEQUENCE:

1. REQENB TO I/O
2. DCHR TO CPU
3. 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 INTR SAME AS NORMAL I/O

DG-00031

MODE BIT TABLE

DCHM0	DCHM1	FUNCTION
H	H	OUT (WRITE)
H	L	INCREMENT
L	H	IN (READ)
L	L	NOT USED

Figure C-6 Data Channel Signals

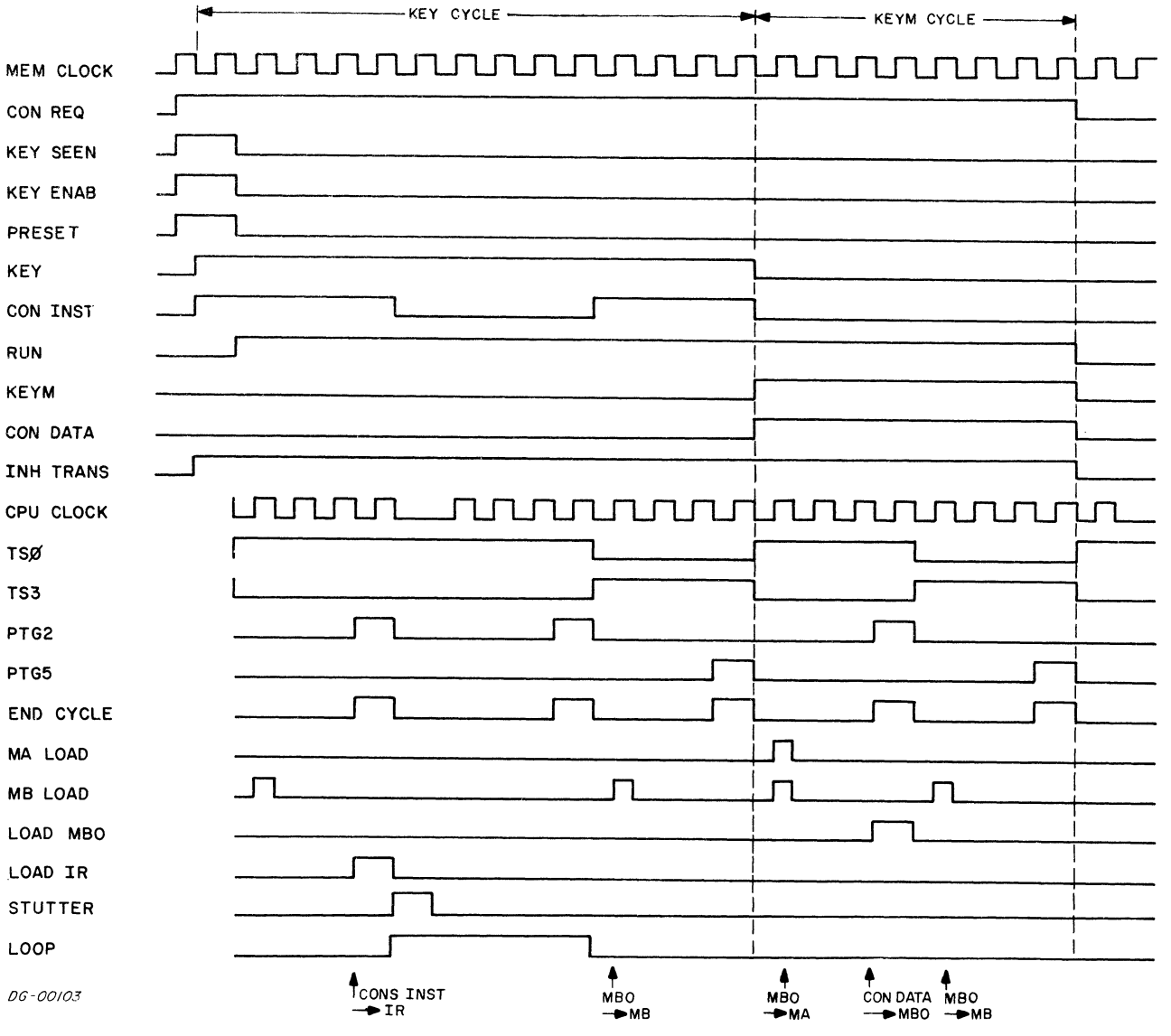


Figure C-7 Deposit Timing Diagram

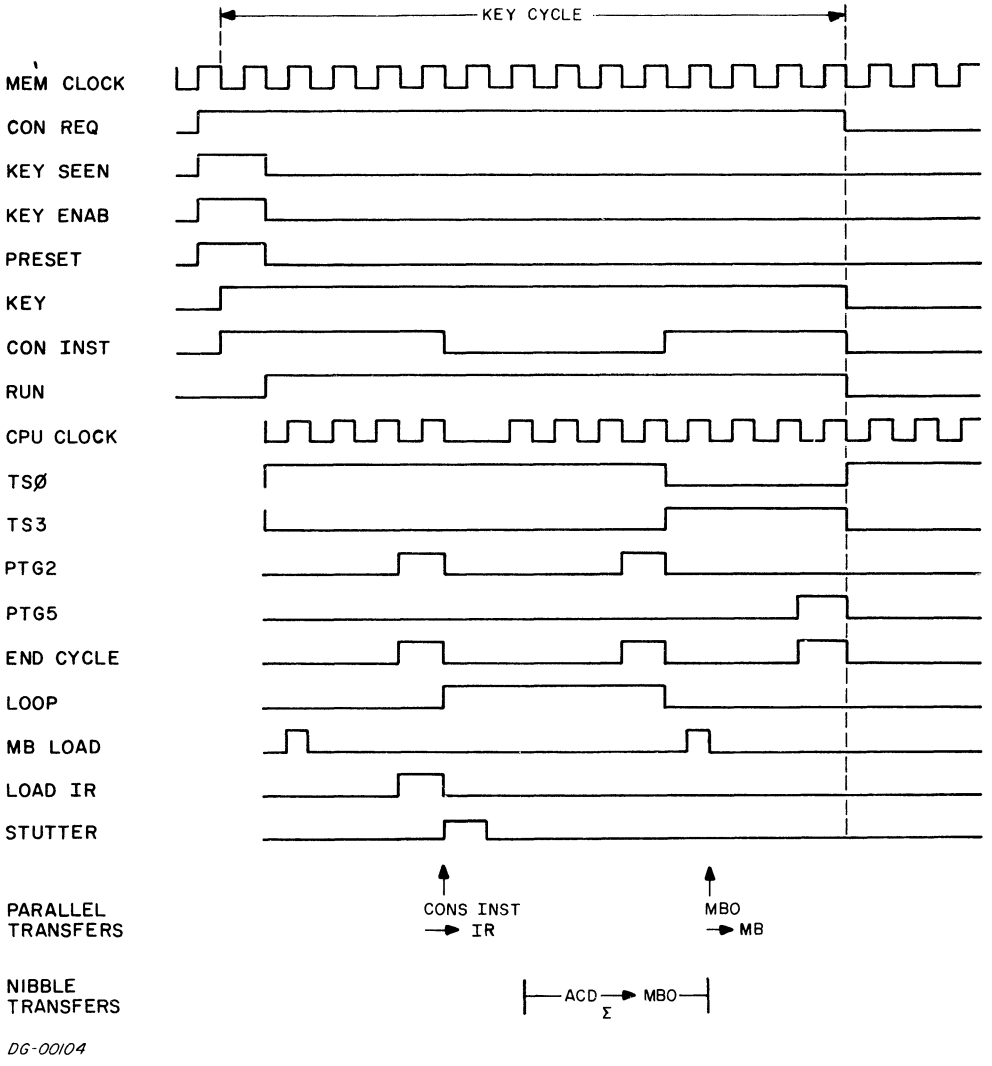


Figure C-8 Examine AC1 Timing Diagram

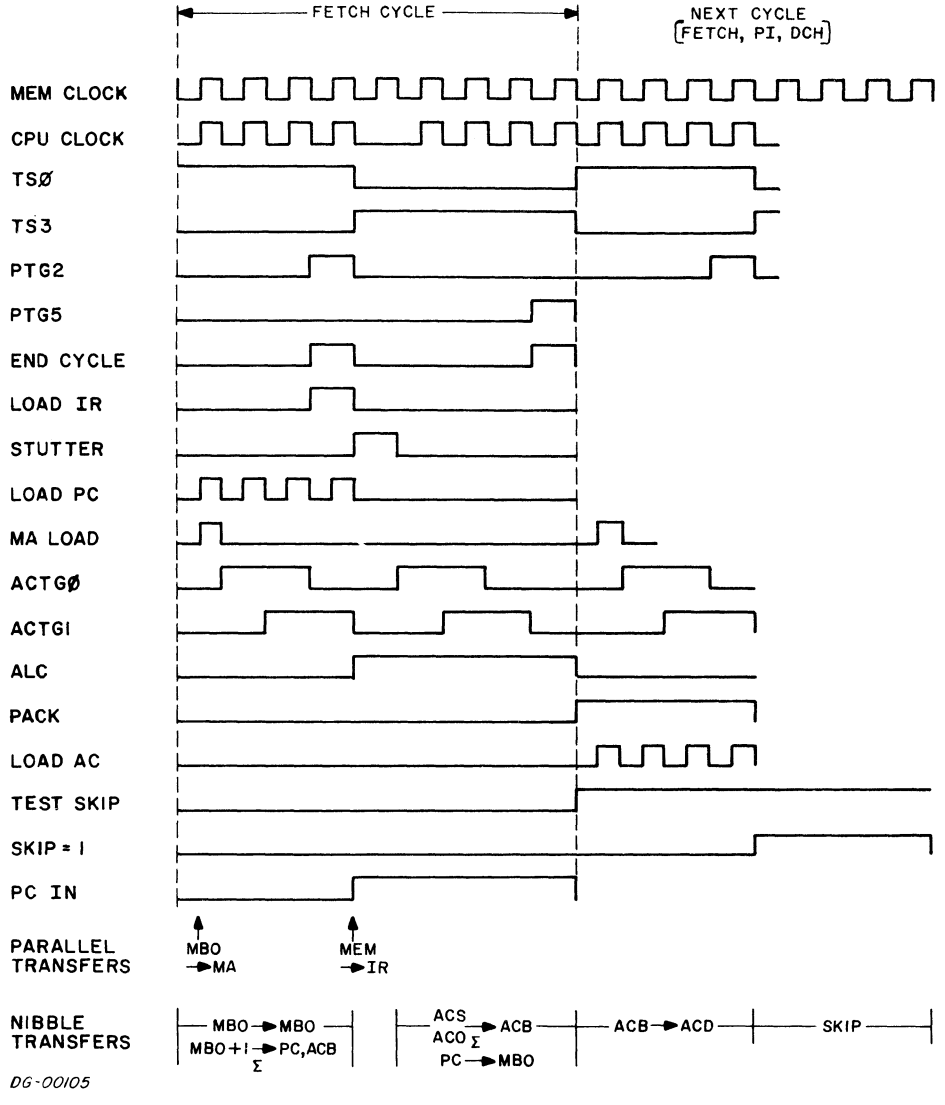


Figure C-9 ADD0, 1, SKP Timing Diagram

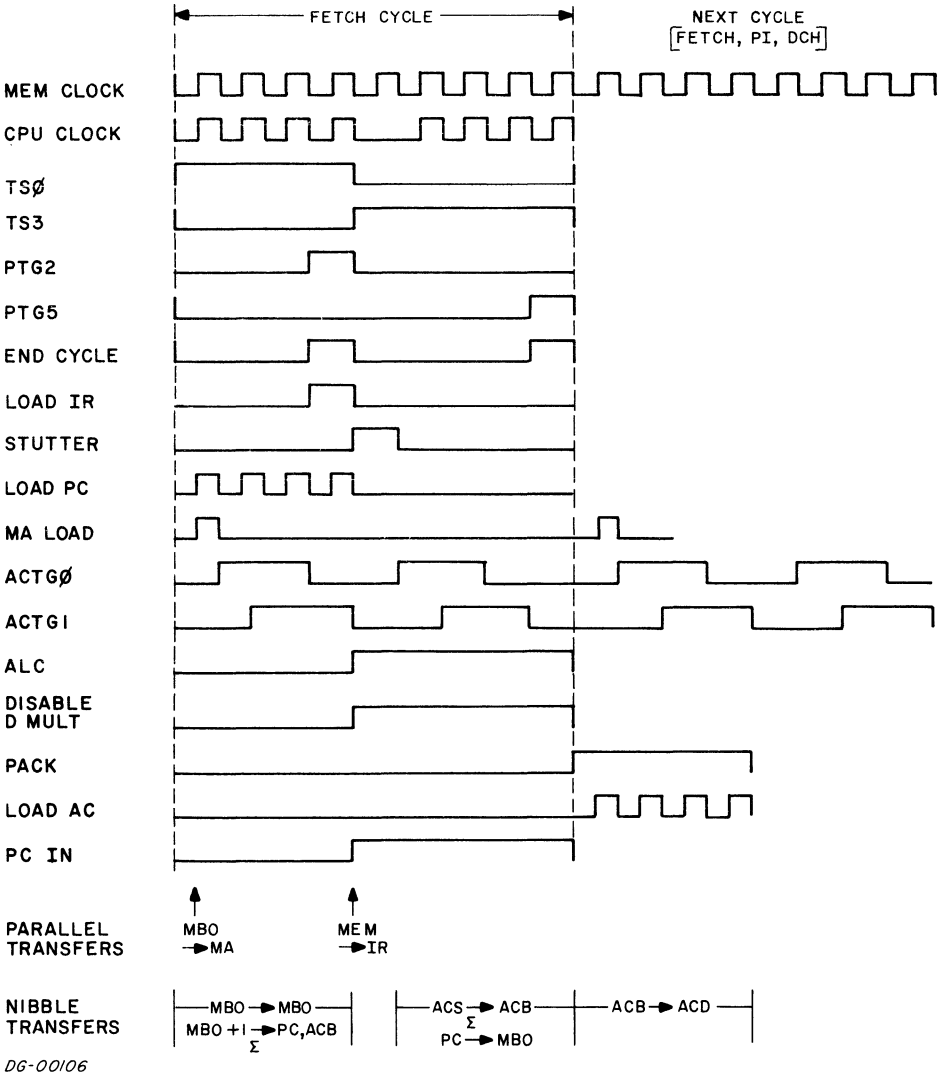


Figure C-10 MOV 0, 0 Timing Diagram

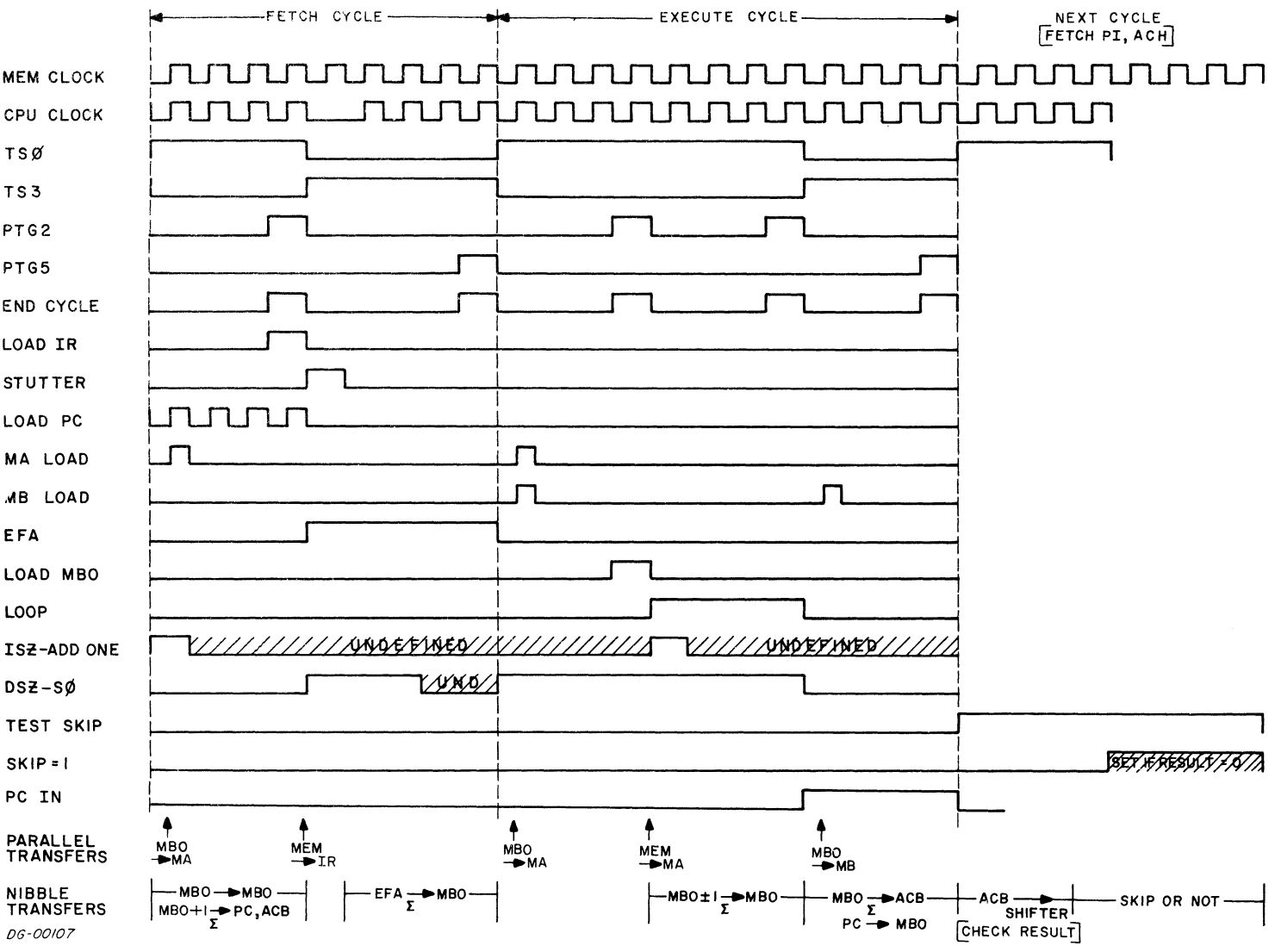


Figure C-11 Timing Diagram For Both The ISZ And DSZ Instructions

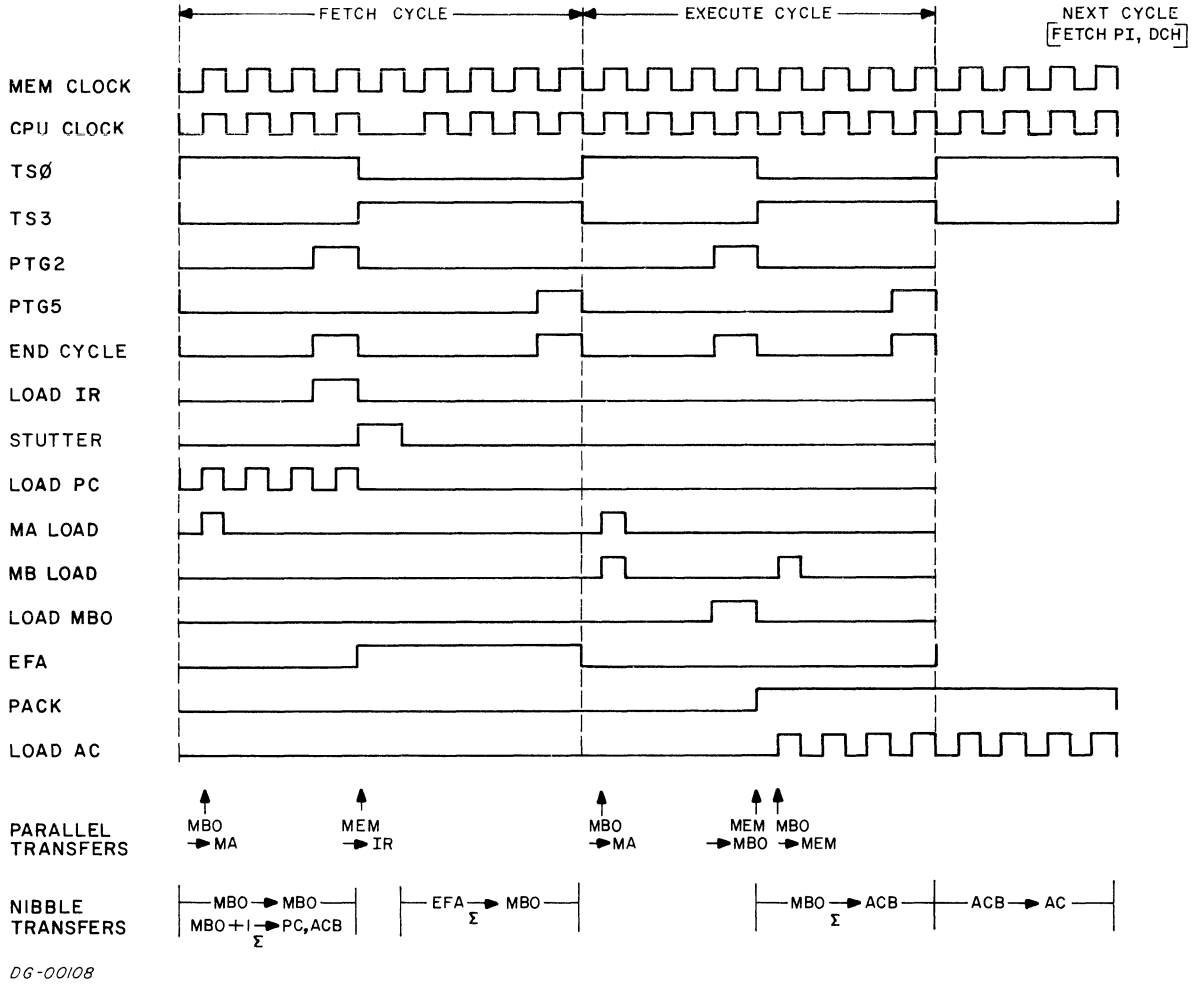


Figure C-12 LDA Timing Diagram

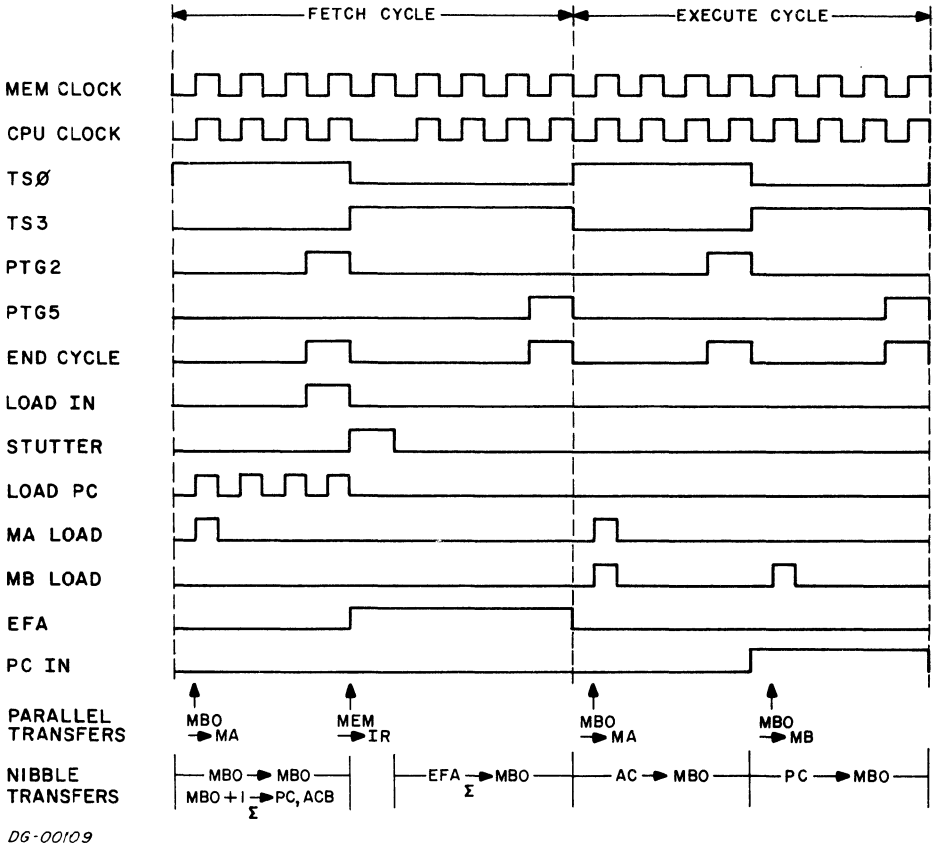


Figure C-13 STA Timing Diagram

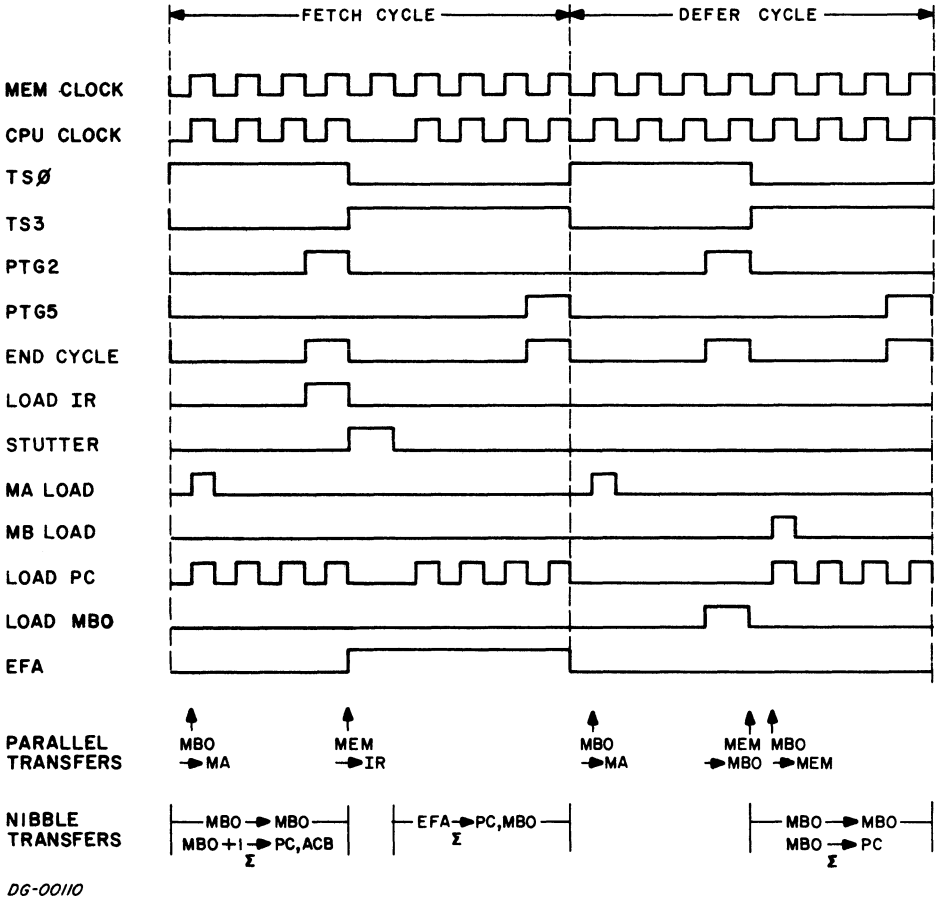


Figure C-14 JMP @ 100 Timing Diagram

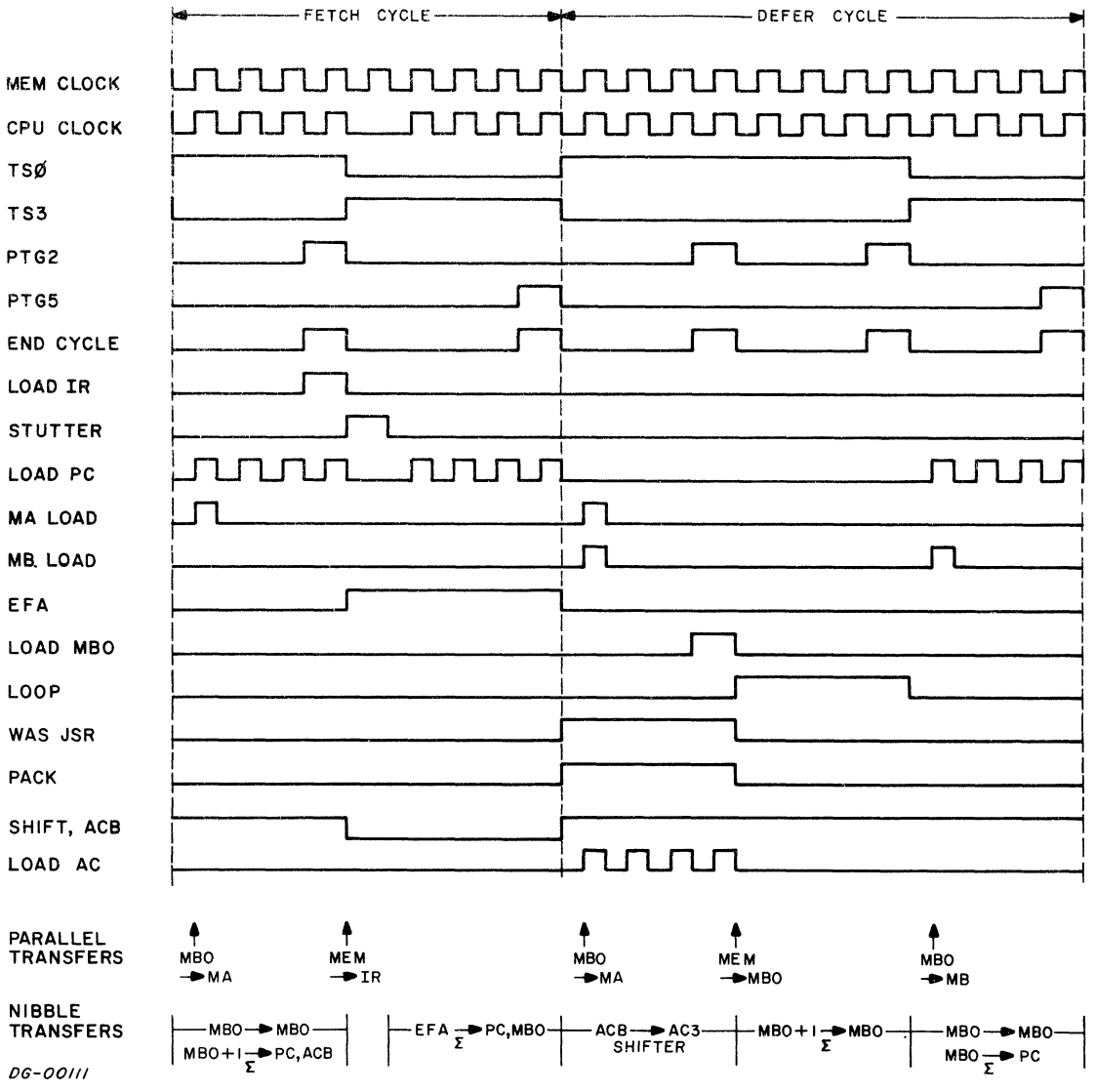


Figure C-15 JSR @ 20 Timing Diagram

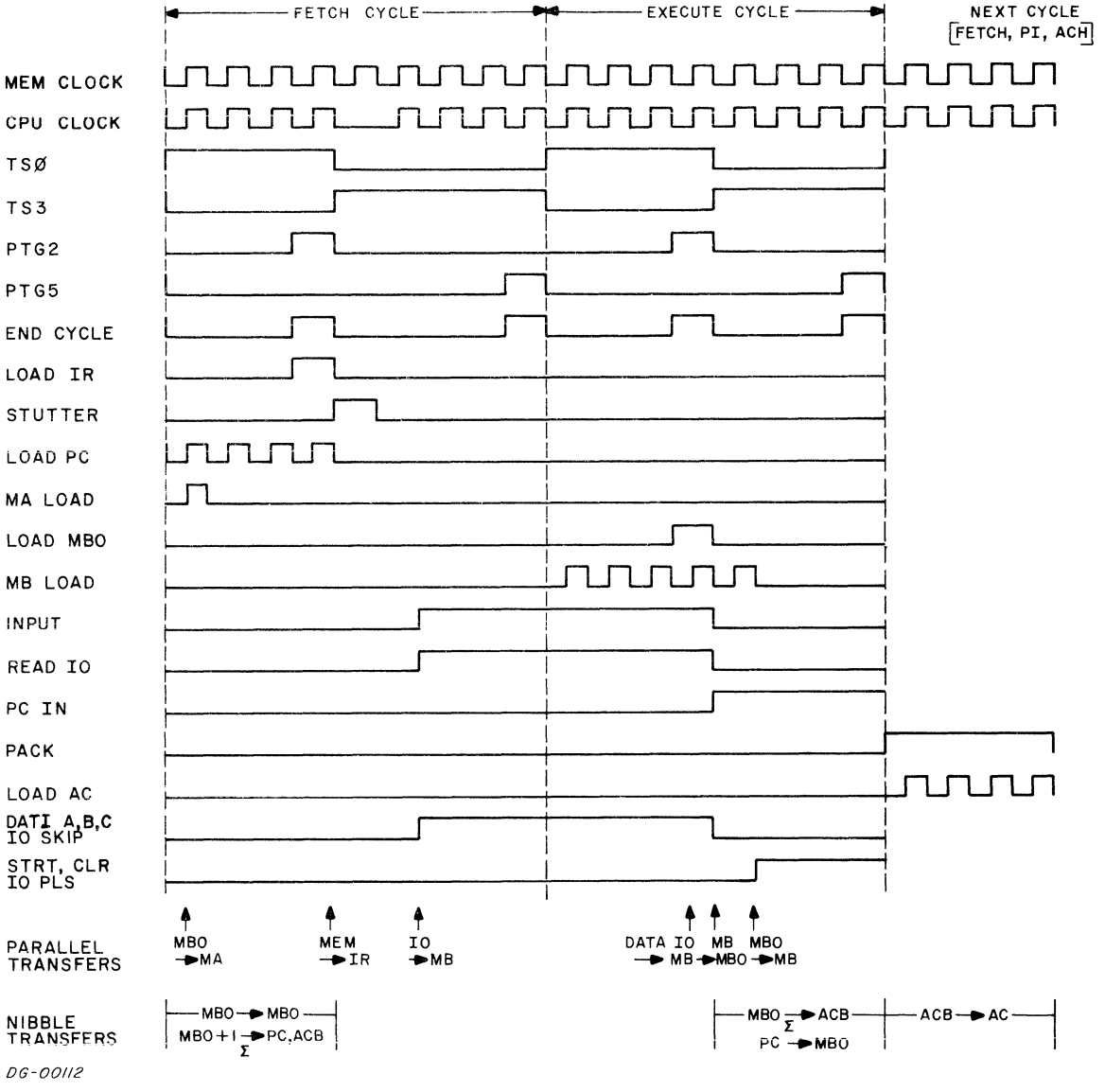


Figure C-16 I/O Input Timing Diagram

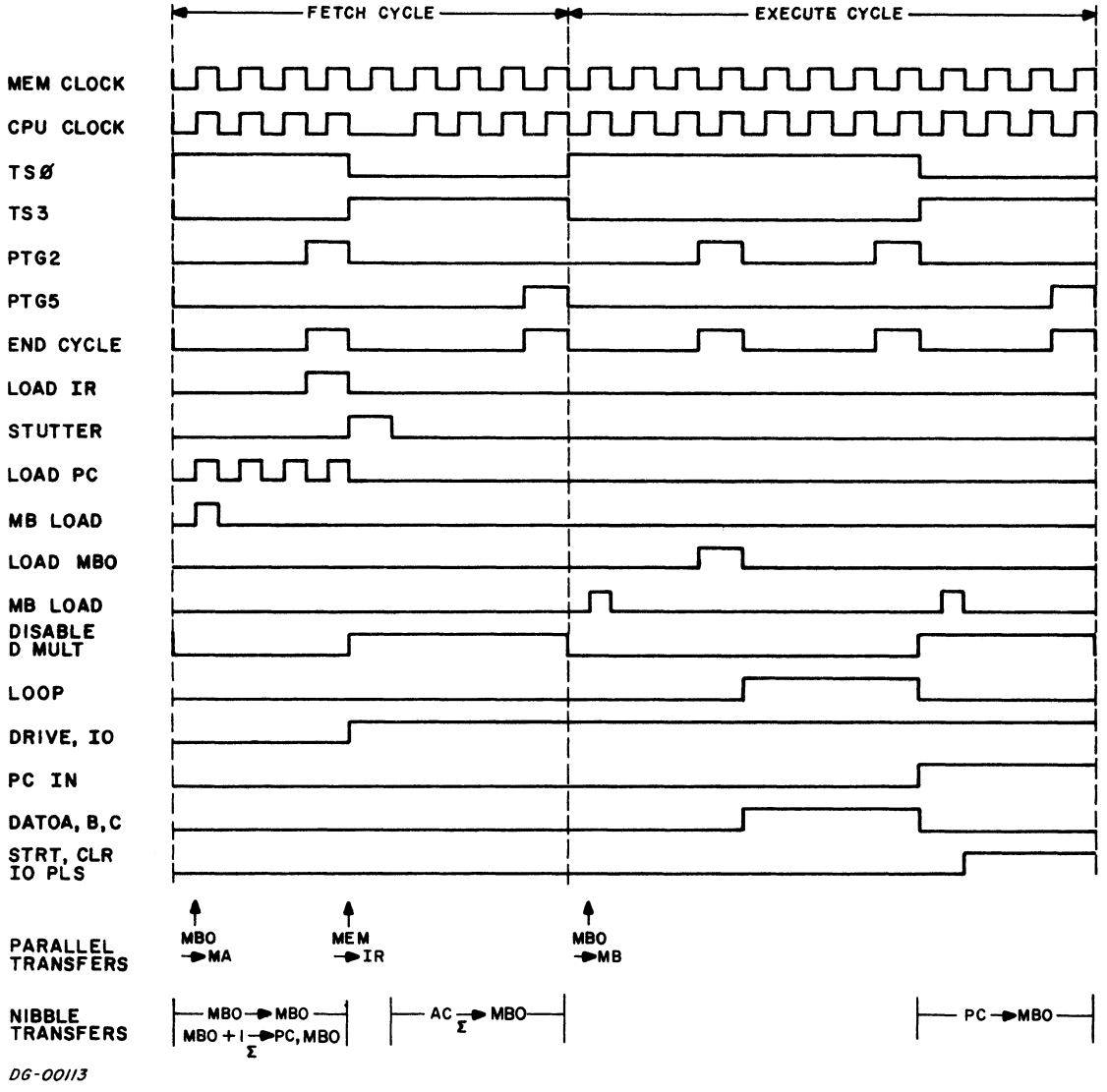


Figure C-17 I/O Output Timing Diagram

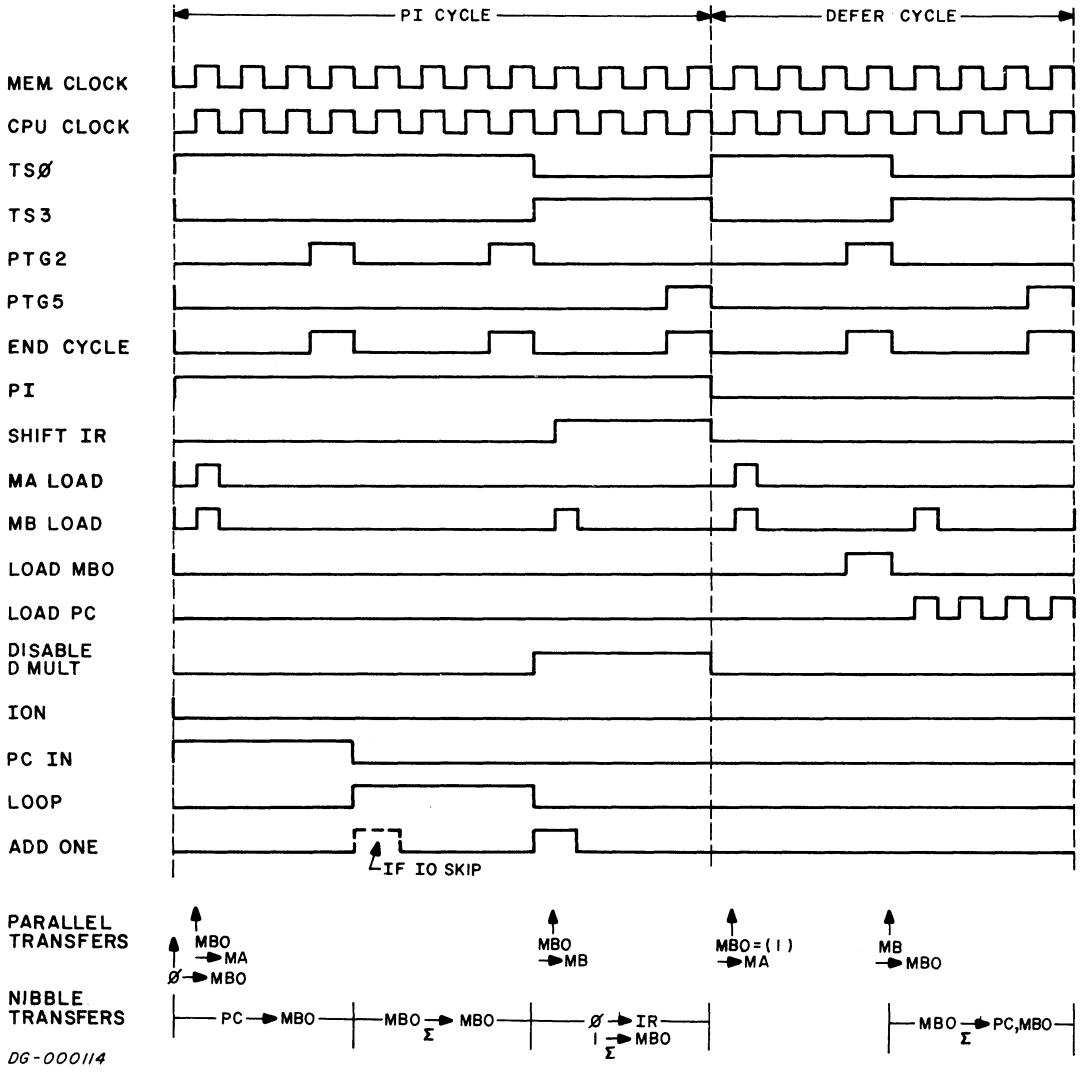


Figure C-18 PI Timing Diagram

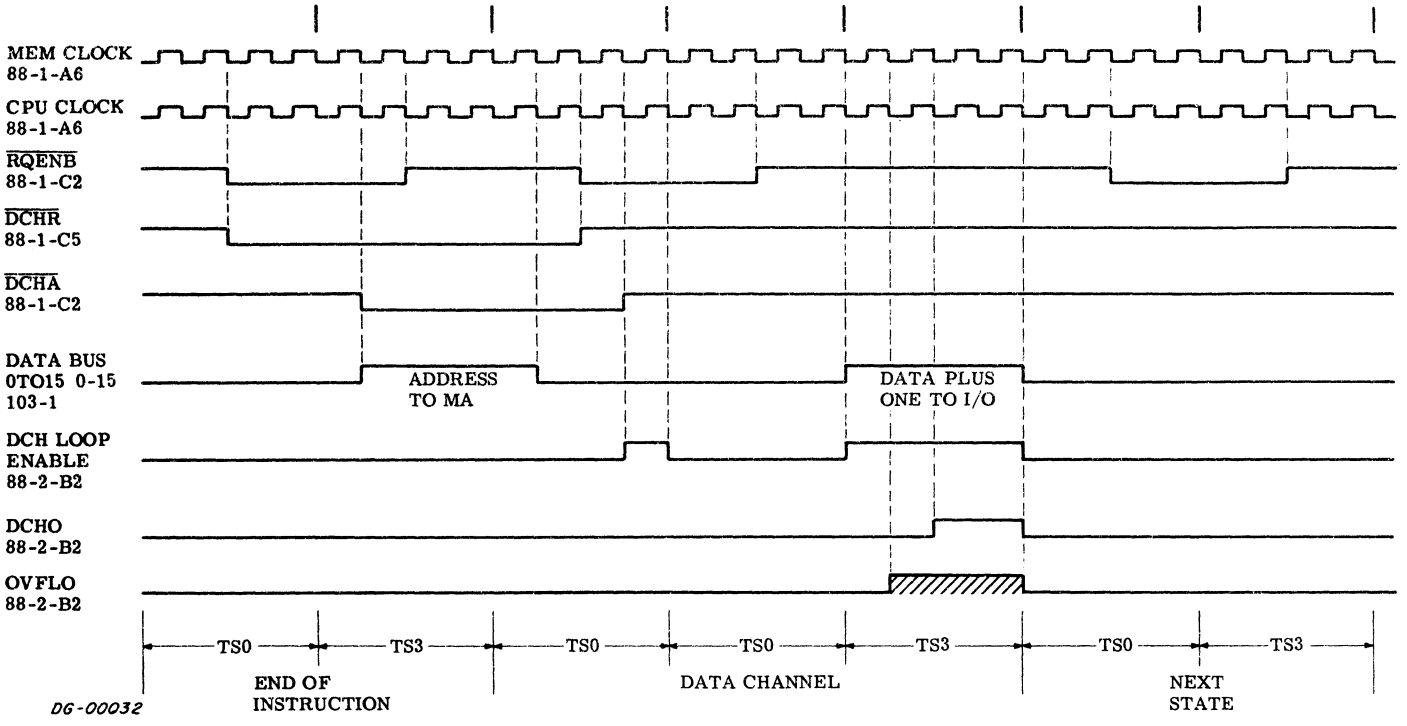


Figure C-19 Data Channel Increment Timing

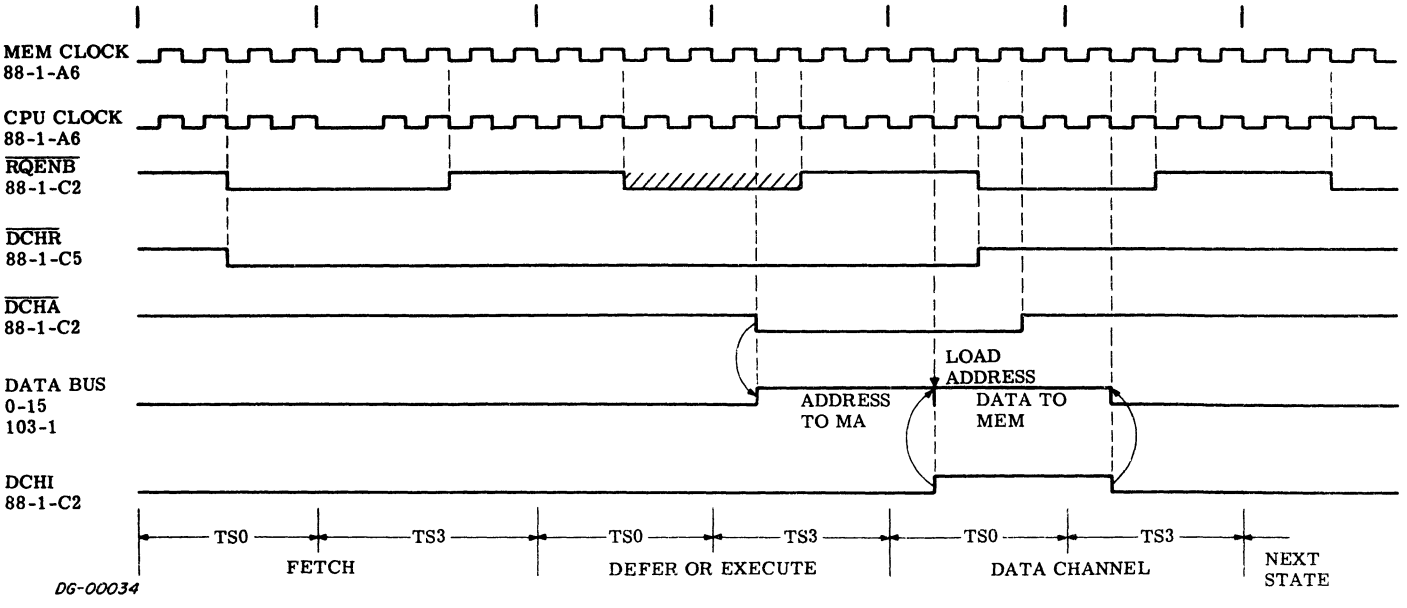


Figure C-20 Data Channel In Timing

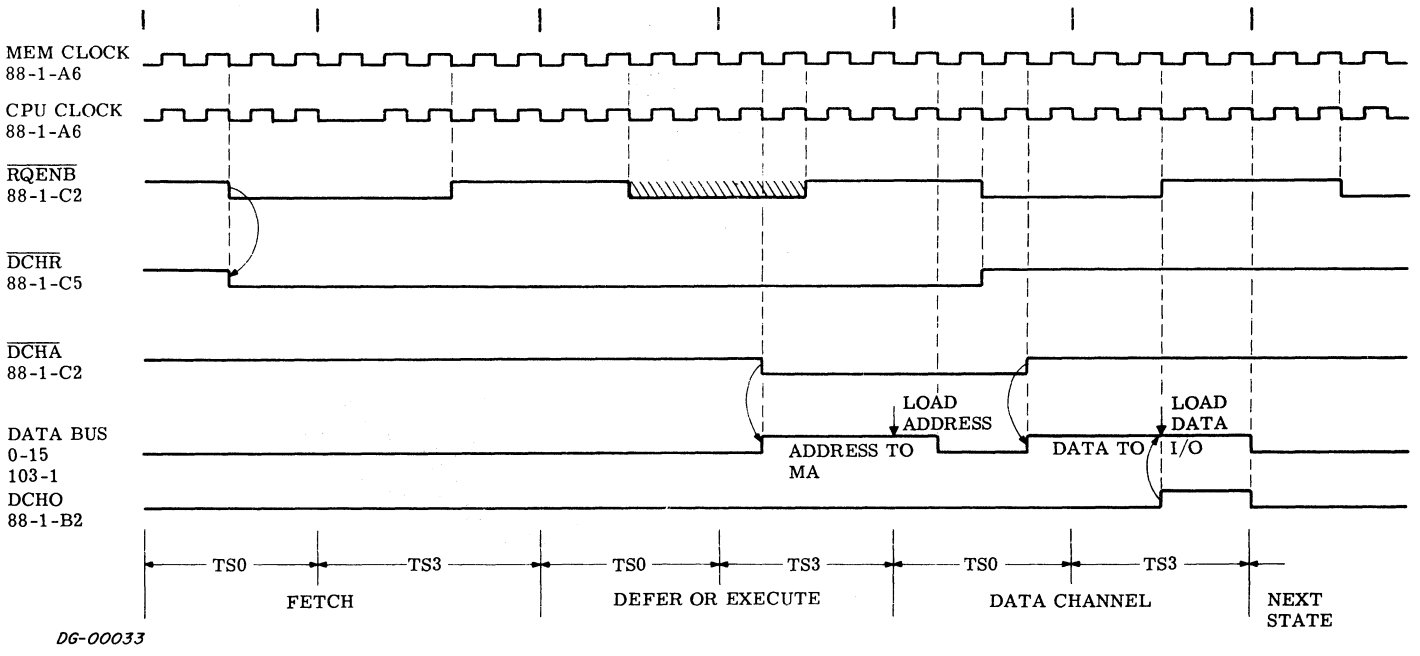


Figure C-21 Data Channel Out Timing

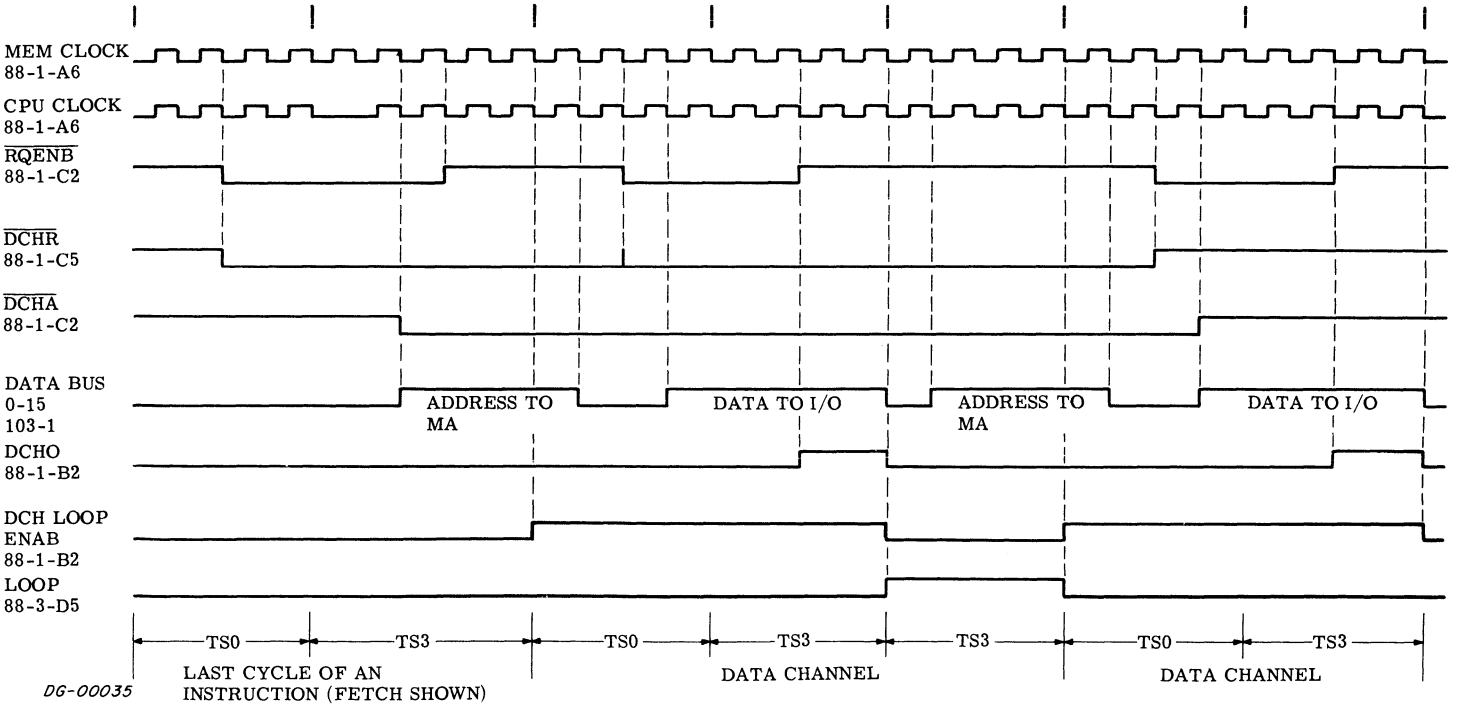


Figure C-22 Data Channel Out Followed By Data Channel In Timing

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

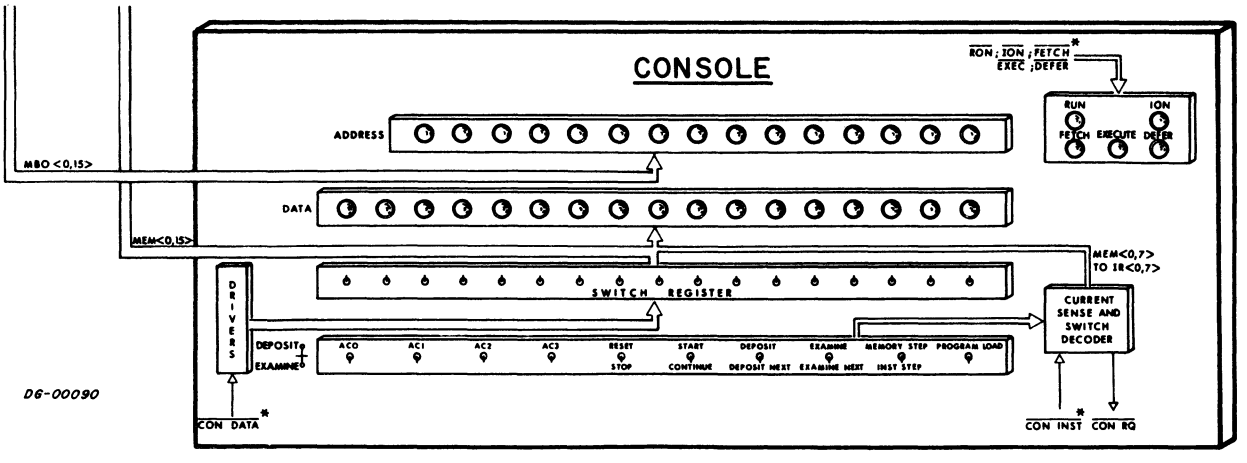
These lights are always showing the state of the MBO bus which is driven directly from the MBO register. When the machine is running, the MBO register is continually shifting, so the display is meaningless; when the machine is stopped, the MBO register shows the contents of the PC, i.e., the next address.

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 $\overline{\text{CON RQ}}$ 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.

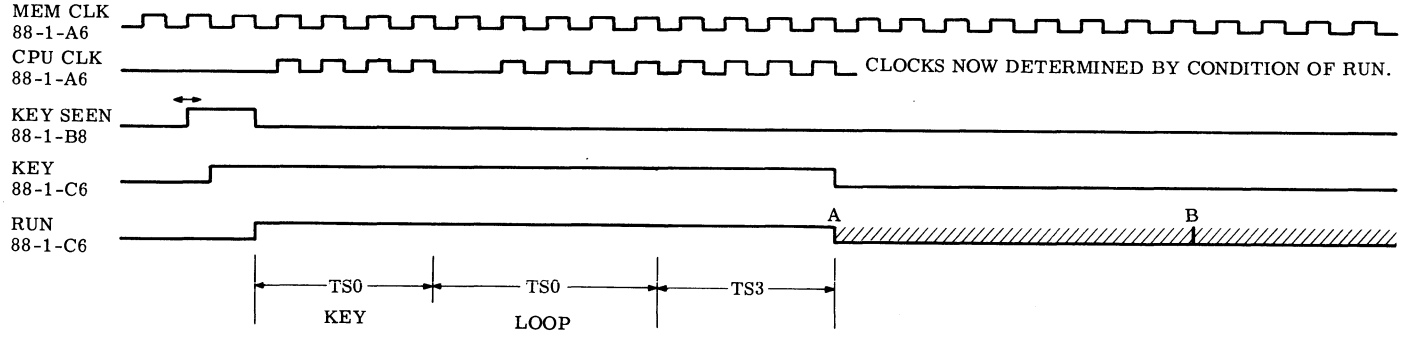
The Console Rotary Switch

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

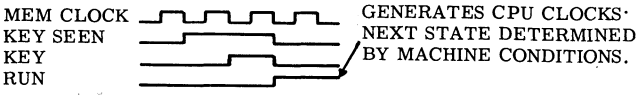
REFERENCES

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

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

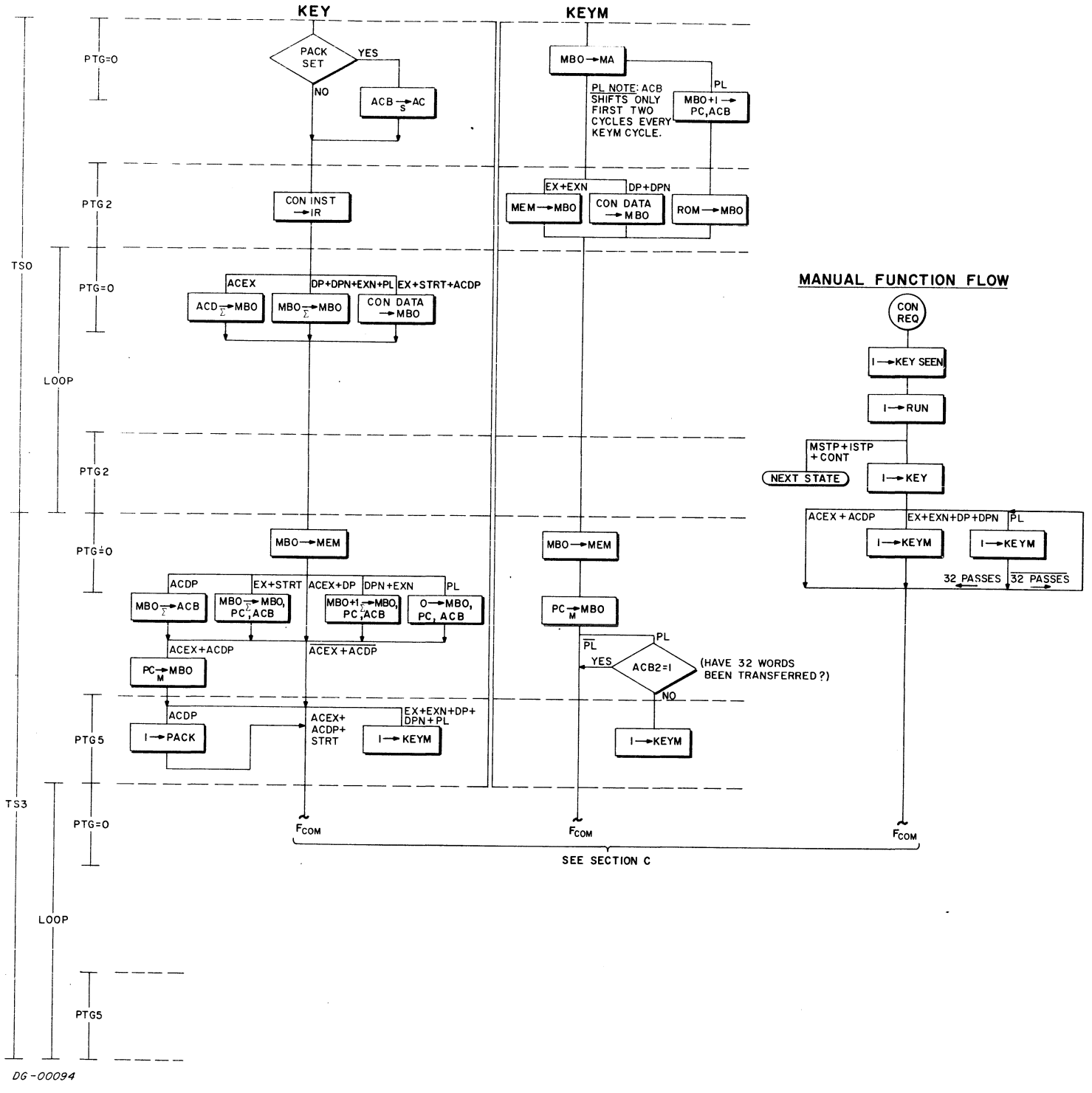


Figure K-3 Key, KEYM and Manual Flow Diagrams

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 1210 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 failure in the dc power supply (10-20% lower than normal), 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.

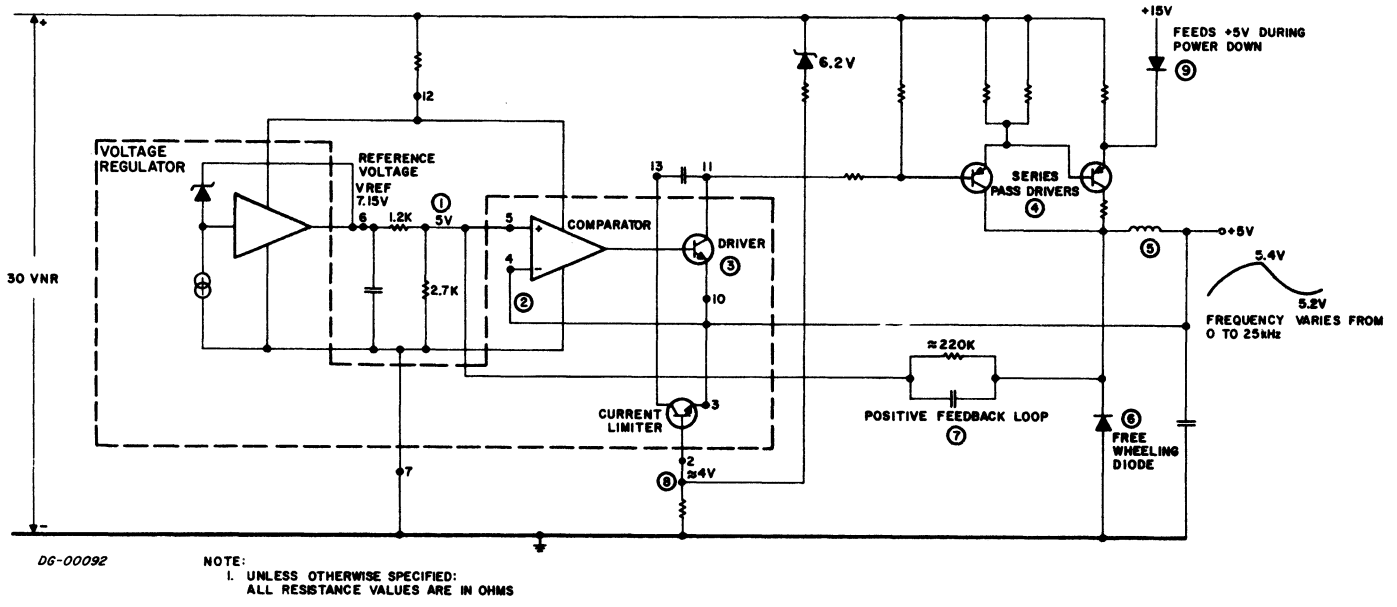


Figure P-1 Simplified Schematic of the +5Vdc Series Switching Regulator. When the comparator senses a difference between the (divided) reference voltage (1) and the output voltage (2) it switches, turning on the driver transistor (3) and consequently the series pass transistors (4). Current is shunted 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.

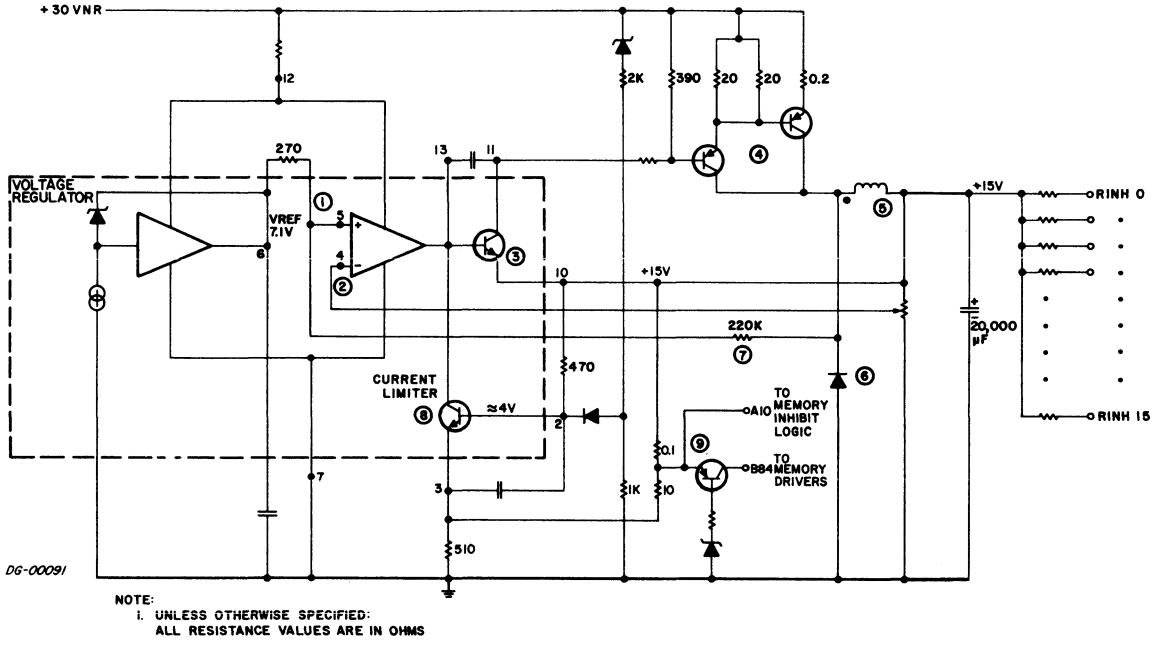


Figure P-2 Simplified Schematic of the +15Vdc Series Switching Regulator. When the comparator senses a difference between the reference voltage (1) and the divided output voltage (2), it switches, turning on the driver transistor (3) and consequently the series pass transistors (4). Current is shunted 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 wire-cycle so that new information is entered into the address.

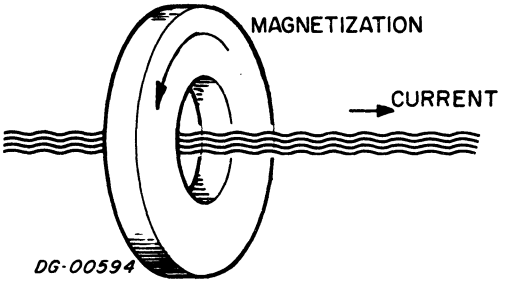


Figure M-1 Simplified Schematic of a Memory Core

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

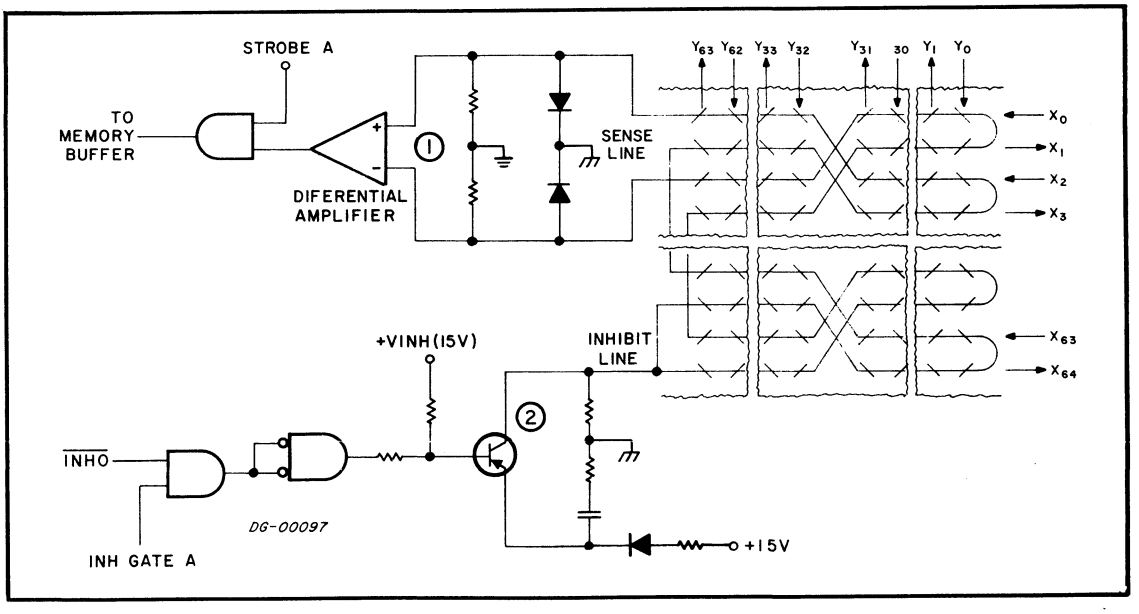


Figure M-2 Simplified Schematic of The Core Memory's Sense and Inhibit Lines

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.

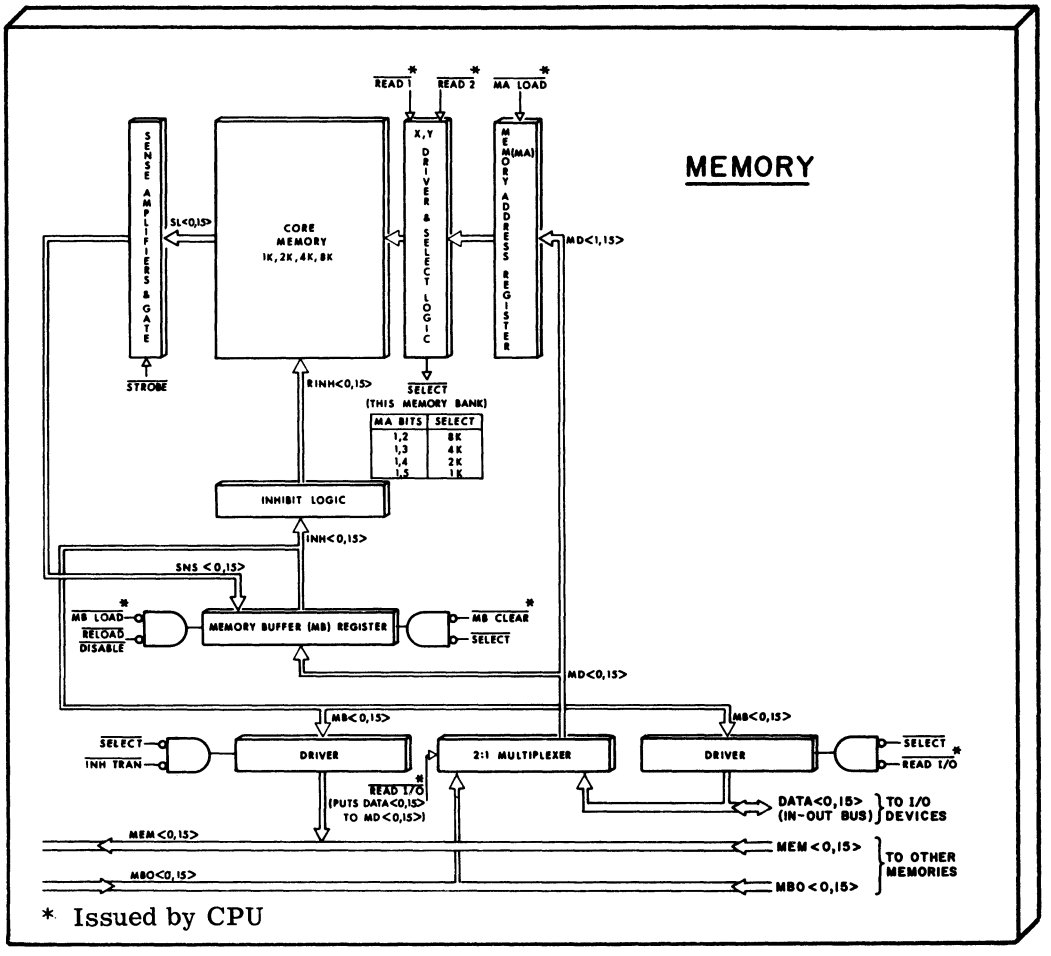
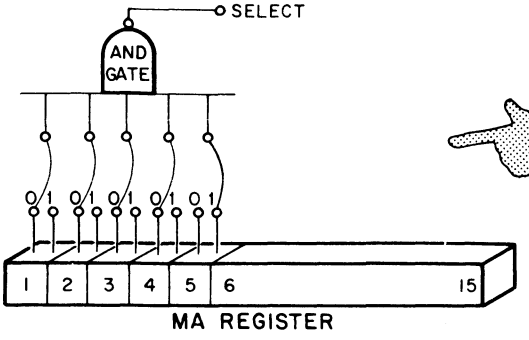
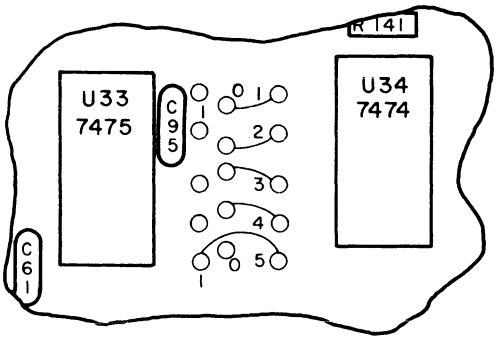


Figure M-3 Core Memory

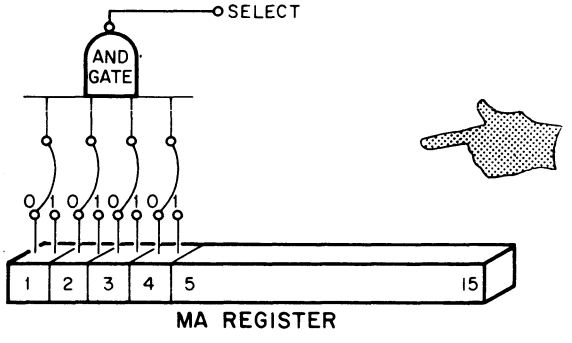
During a typical FETCH instruction, the CPU outputs the memory address on the MBO <0, 15> data lines and then issues MA LOAD. READ I/O is high, so the address is strobed into the Memory Address register and output to the driver select logic. Then, READ 1 and READ 2 are issued, gating the X and Y currents 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.

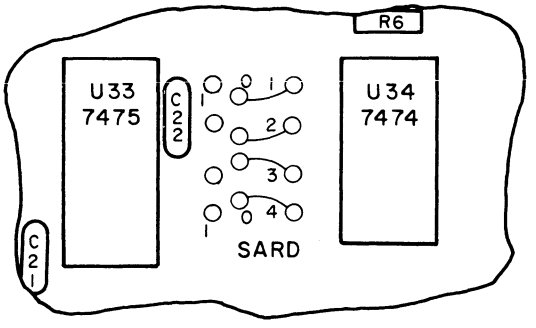




1K BOARDS						
MA BITS JUMPERED					BOARD NUMBER	ADDRESSES ENABLED (OCTAL)
1	2	3	4	5		
0	0	0	0	0	1	00000-01777
0	0	0	0	1	2	02000-03777
0	0	0	1	0	3	04000-05777
0	0	0	1	1	4	06000-07777
0	0	1	0	0	5	10000-11777
0	0	1	0	1	6	12000-13777
0	0	1	1	0	7	14000-15777
0	0	1	1	1	8	16000-17777

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.



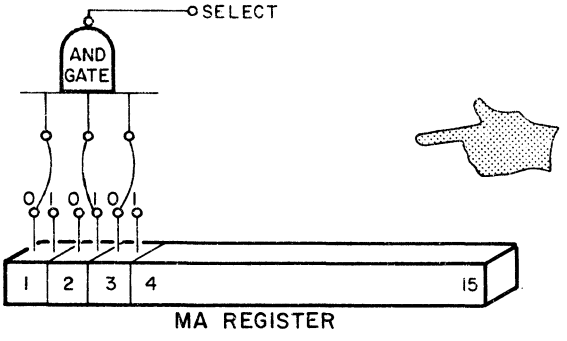


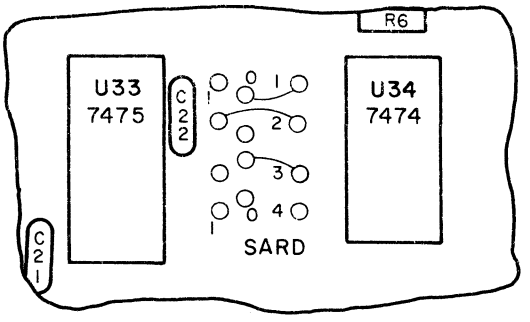
2K BOARDS						
MA BITS JUMPERED					BOARD NUMBER	ADDRESSES ENABLED (OCTAL)
1	2	3	4			
0	0	0	0		1	00000 - 03777
0	0	0	1		2	04000 - 07777
0	0	1	0		3	10000 - 13777
0	0	1	1		4	14000 - 17777
0	1	0	0		5	20000 - 23777
0	1	0	1		6	24000 - 27777
0	1	1	0		7	30000 - 33777
0	1	1	1		8	34000 - 37777

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.

DG-00095A

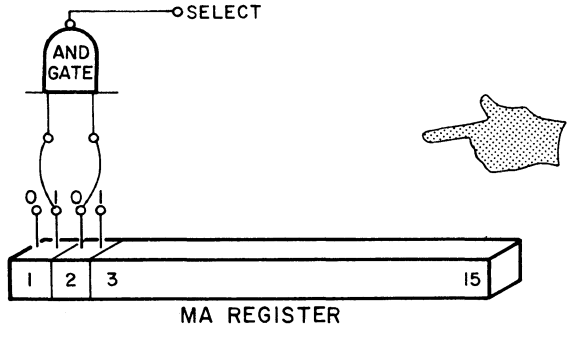
Figure M-4 Wiring Up The Select Logic of 1K and 2K Boards

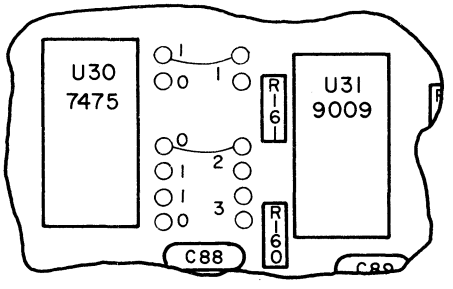




4K BOARDS						
MA BITS JUMPERED					BOARD NUMBER	ADDRESSES ENABLED (OCTAL)
1	2	3				
0	0	0			1	00000-07777
0	0	1			2	10000-17777
0	1	0			3	20000-27777
0	1	1			4	30000-37777
1	0	0			5	40000-47777
1	0	1			6	50000-57777
1	1	0			7	60000-67777
1	1	1			8	70000-77777

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-000958

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

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	520	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 1210 operates from a single phase source at 115V 60Hz or 220V 50Hz a11 ±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.

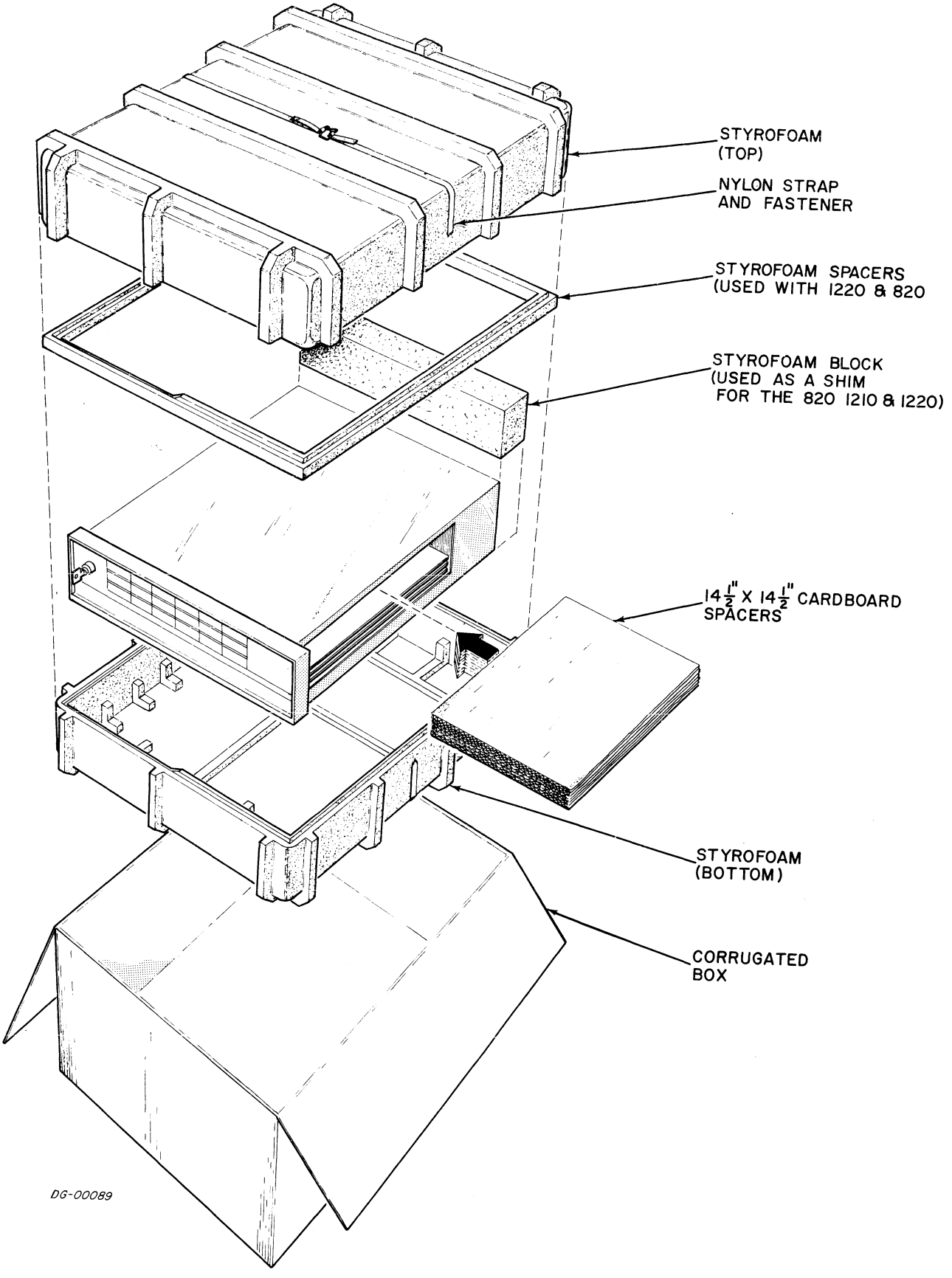


Figure I-1 The Nova 1220 Shipping Kit

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

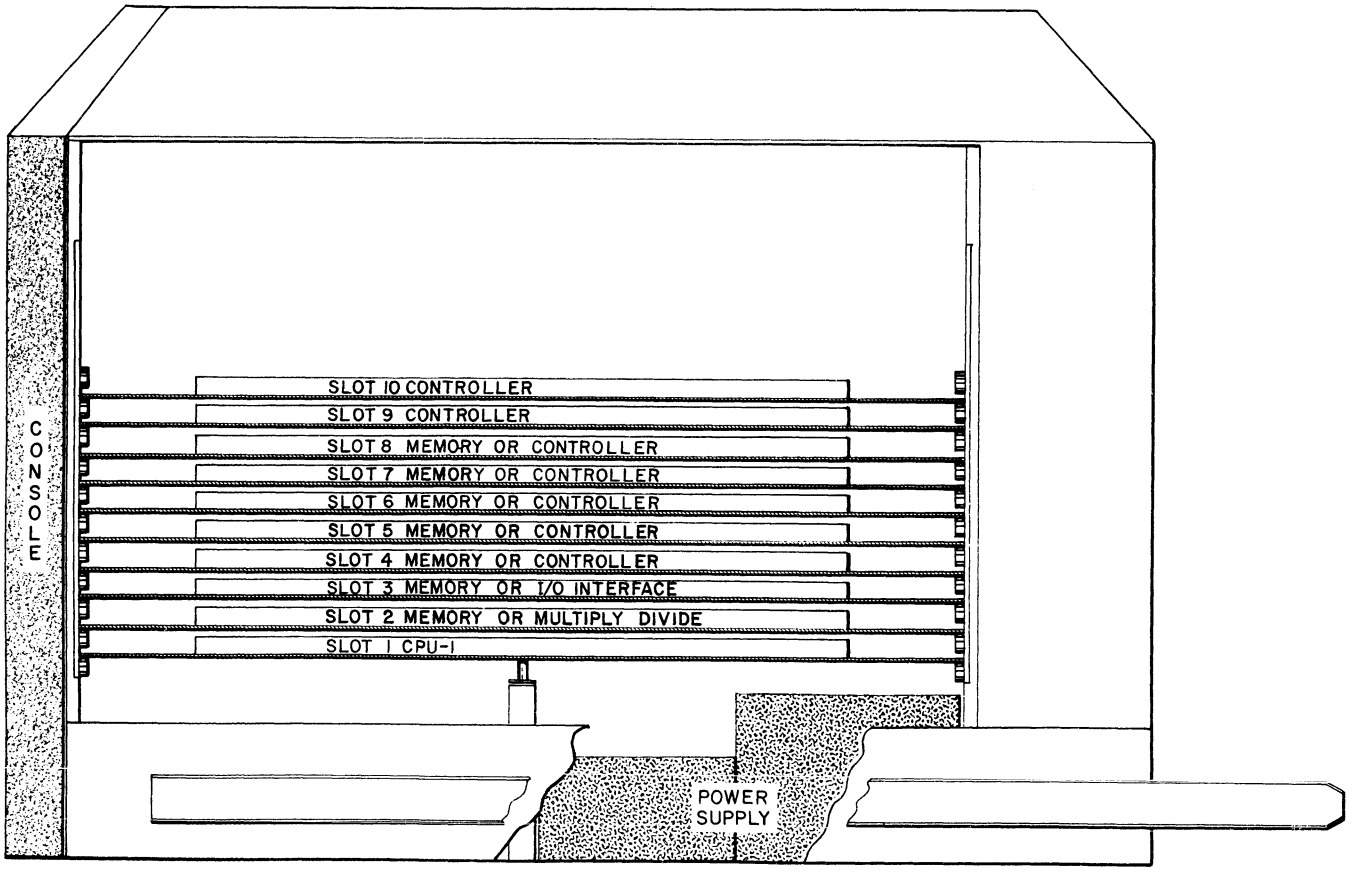
Assembling the computer outside the factory involves installing memory or controller boards or mounting the chassis into a 19" rack.

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.



DG-00099

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.

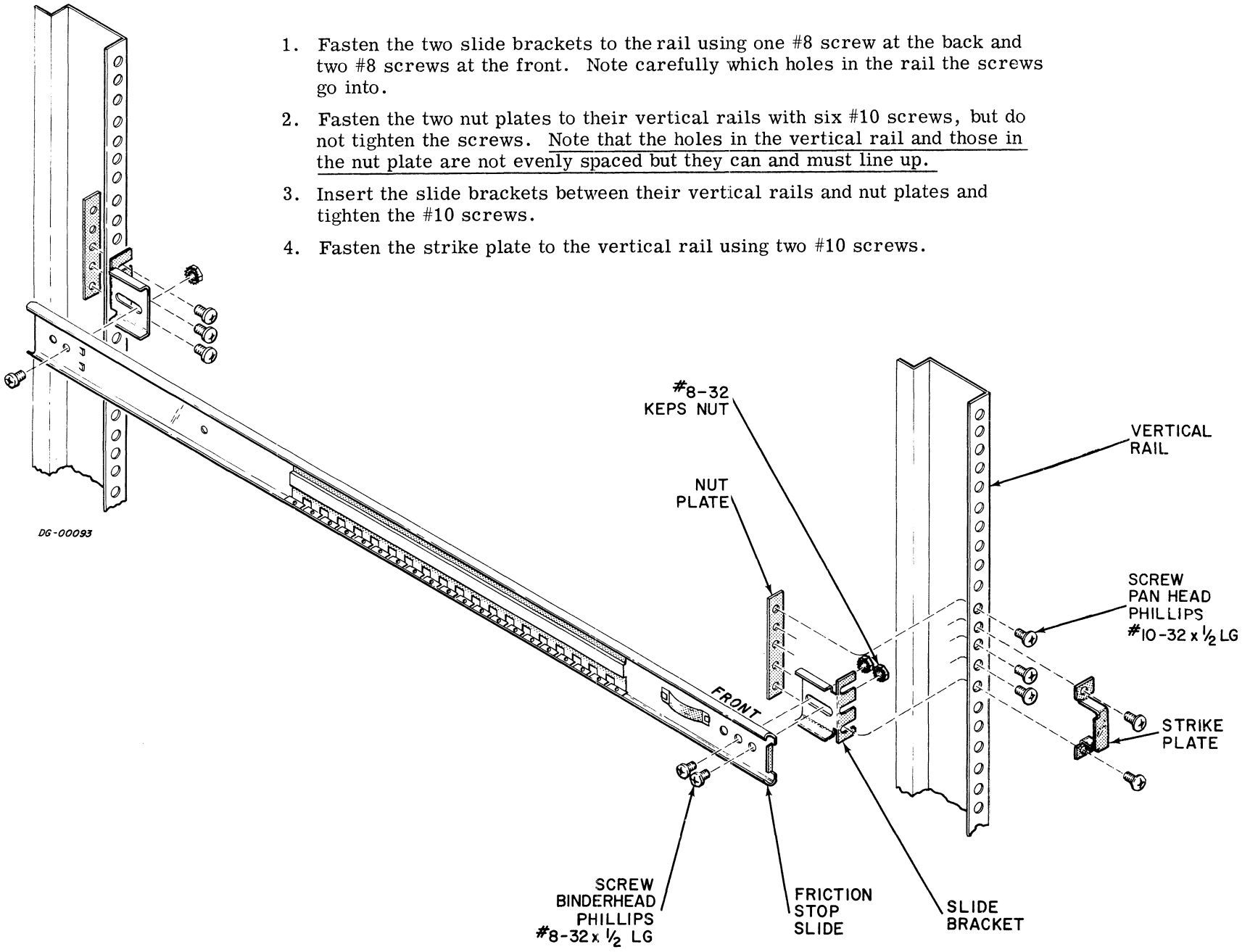


Figure I-3 Rack Mounting Hardware For The Nova 1220

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.

I/O Cables 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.

Device Cables 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.

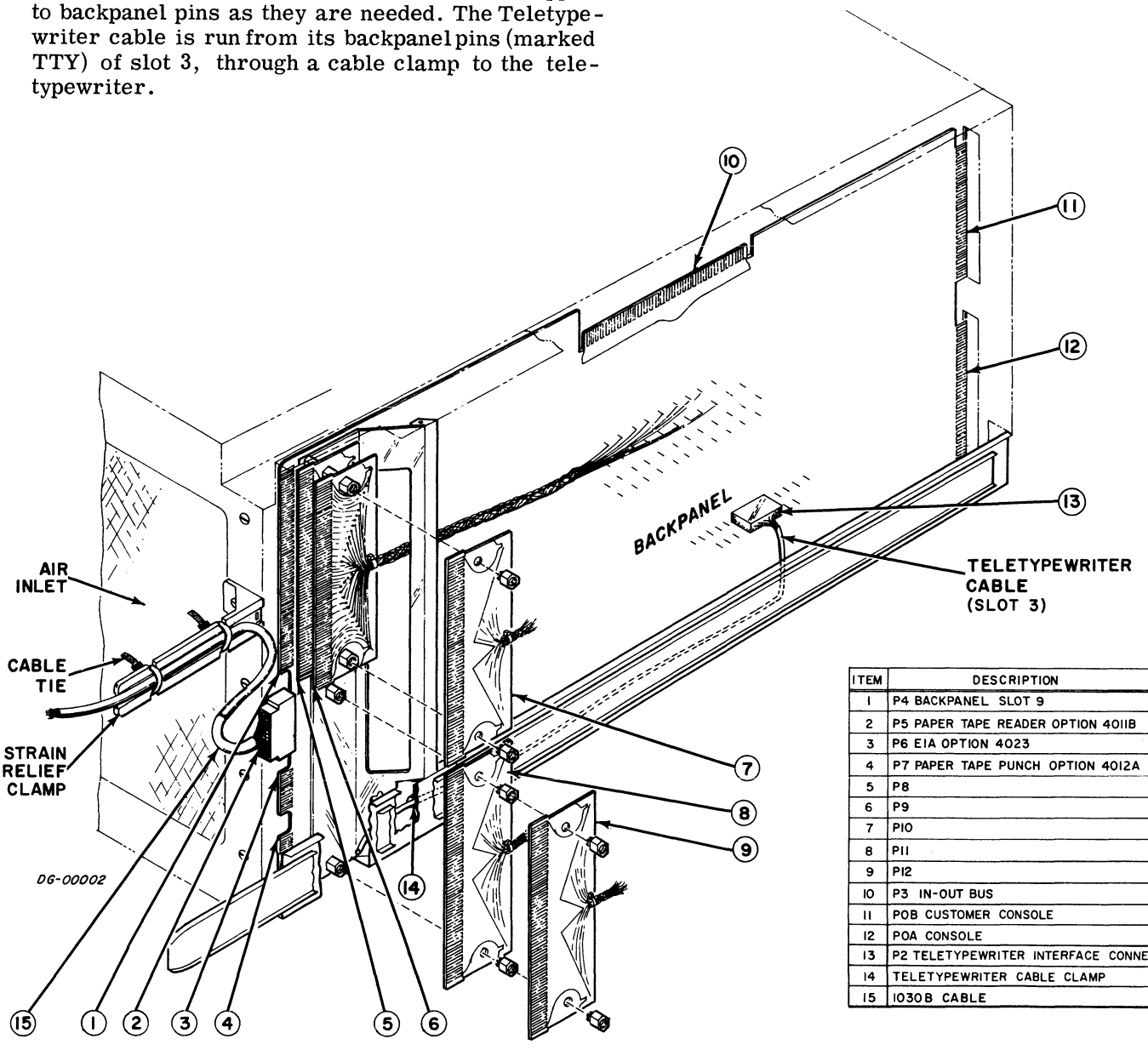
Internal Cables are added when the controller is added, whether in the factory or in the field, so each shipment includes a wire list for the internal cable, and the internal cable itself. Figure I-4 shows how the paddle boards are mounted on the chassis.

Interdevice Cables 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".

Adapter Cables 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	POB CUSTOMER CONSOLE
12	POA 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		$\overline{\text{DATA5}}$
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)	

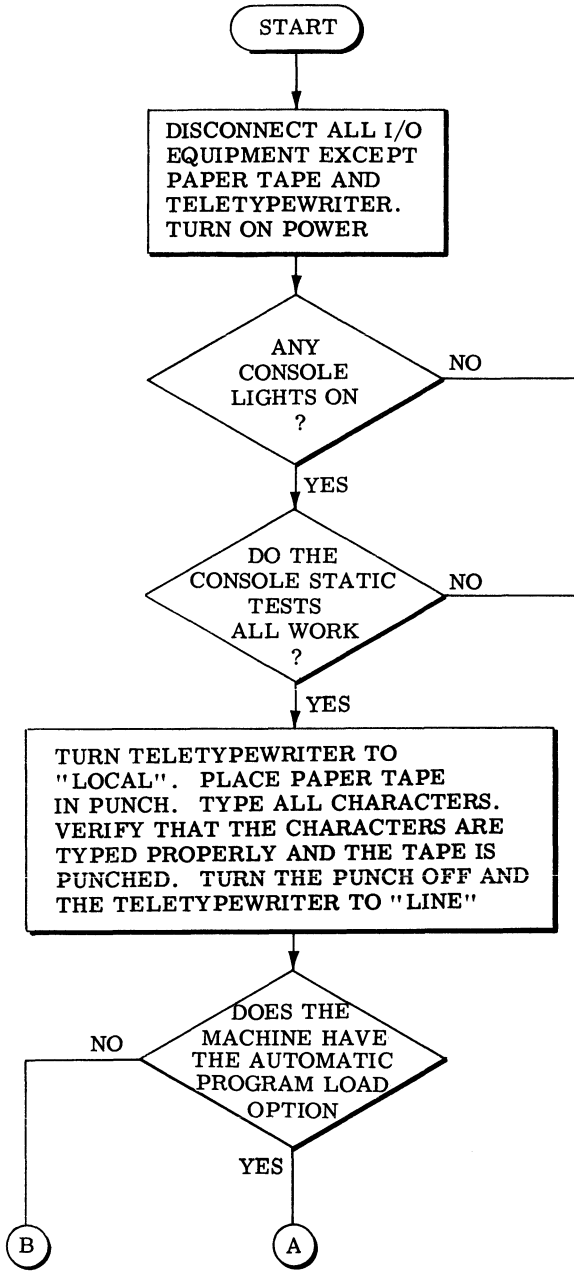
Table N-2 (Continued)

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	11b, 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

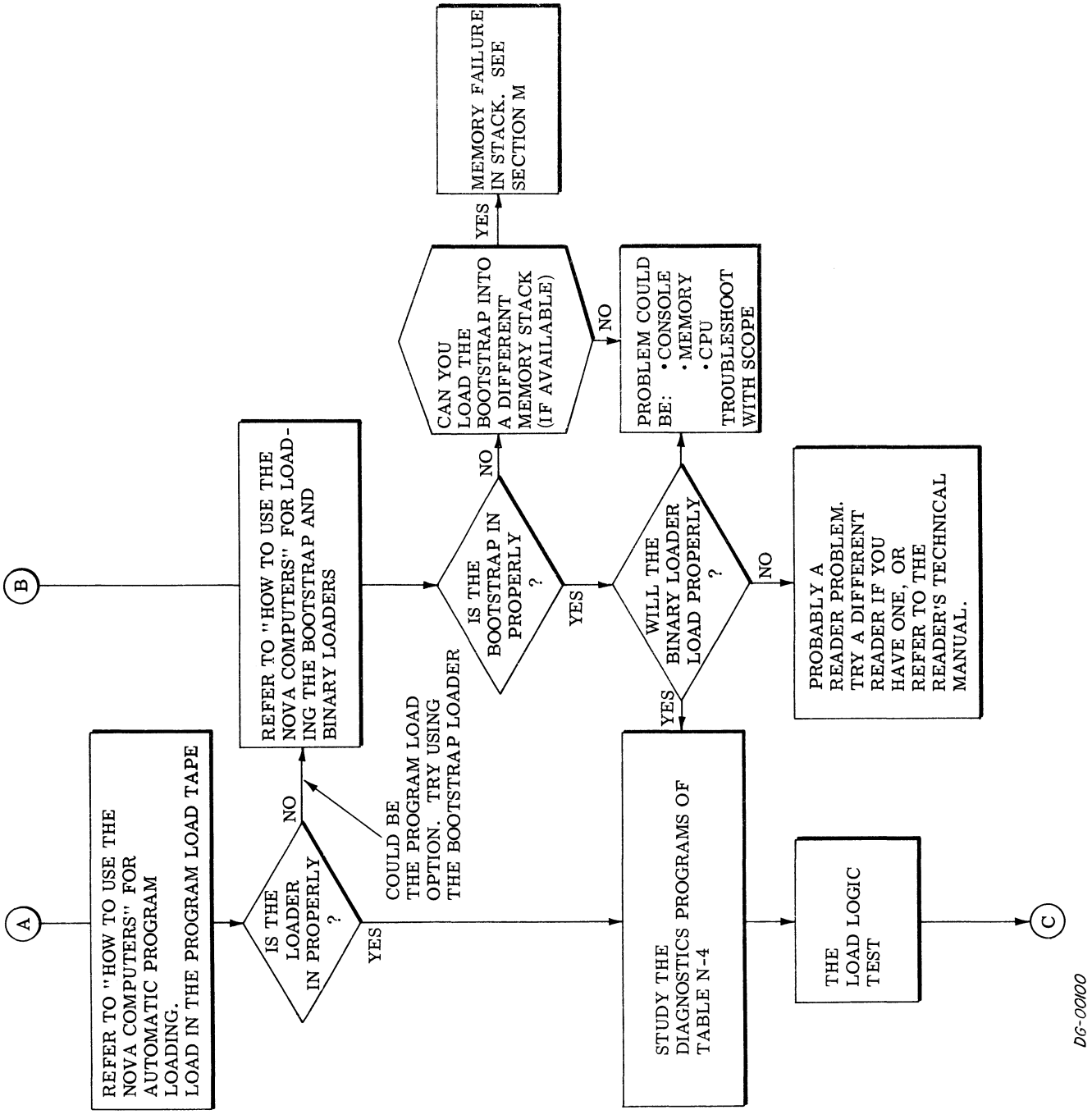


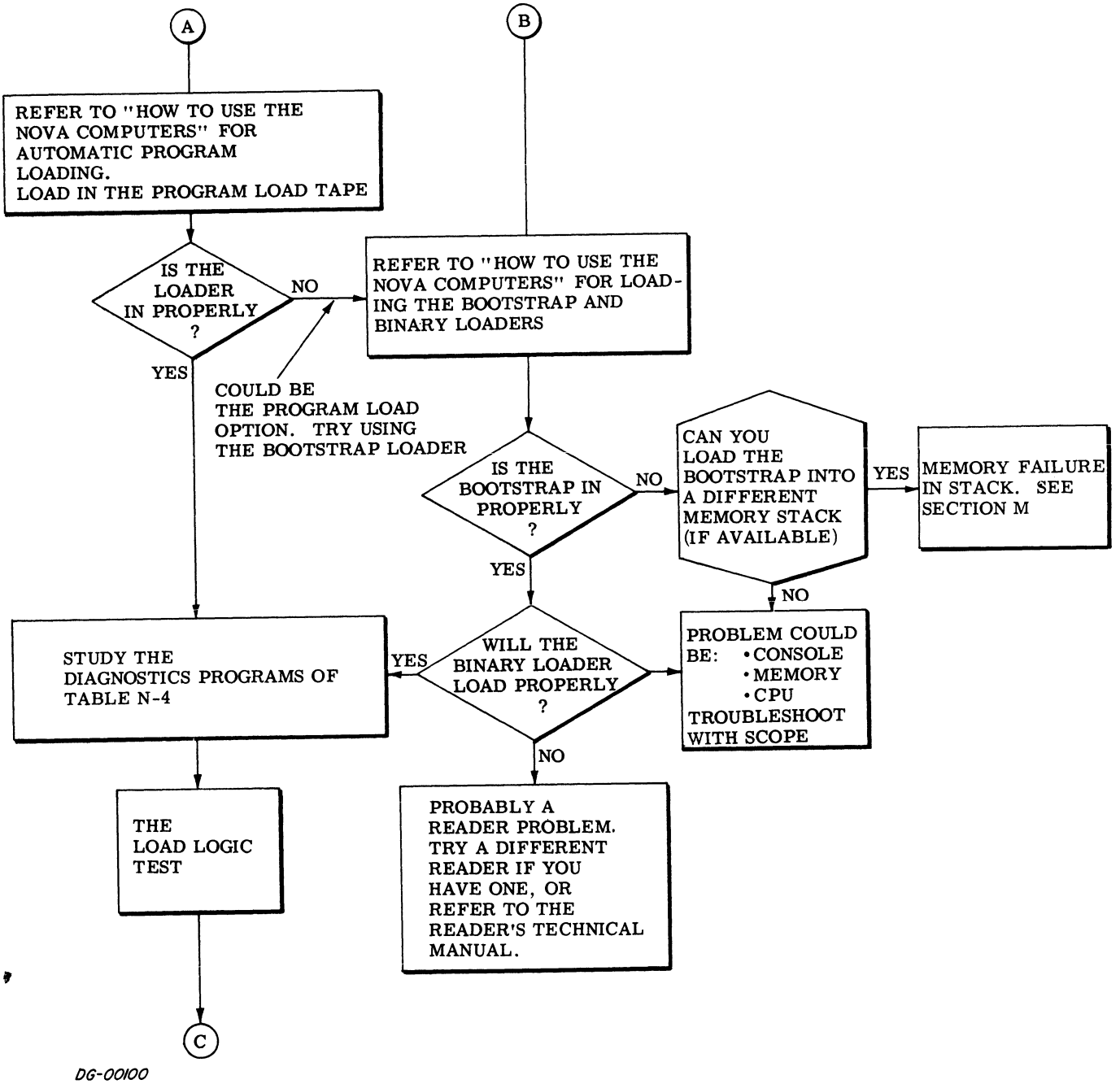
DG-00102

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





NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 &MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
(D+E SET)+(TS3)'	88-2	D5		
(D+E)SET+(TS3)'			88-1	C5
(ISZ+DSZ)' E	88-2	B4		
(ISZ+DSZ)' E			88-3	D6, D8
(ISZ+DSZ)E			88-3	C6
(JMP+JSR)(F+D)	88-2	B5		
(JMP+JSR)(F+D)			88-2	C8
			88-3	B4
(PTG2)' +LOOP			88-2	A7
			88-3	B5
(PTG2)(+LOOP)	88-1	D4		
(TSZ+DSZ)E			88-3	D6
+5OK	88-1	B8		
+SL1			38-2	D7
+SL10			38-2	C4
+SL11			38-2	C4
+SL12			38-2	C4
+SL13			38-2	B4
+SL14			38-2	B4
+SL15			38-2	B4
+SL2			38-2	C7
+SL3			38-2	C7
+SL4			38-2	C7
+SL5			38-2	B7
+SL6			38-2	B7
+SL7			38-2	A7
+SL8			38-2	D4
+SL9			38-2	D7
+SL0			38-2	D7
+V BIAS	38-1	D4	38-2	A5
+VINH	38-2	A8		
+VINHP			38-1	D3
+VLAMP			89-1	C8
-SL1			38-2	C7
-SL10			38-2	C4
-SL11			38-2	C4
-SL12			38-2	B4
-SL13			38-2	B4
-SL14			38-2	B4
-SL15			38-2	A4
-SL2			38-2	C7
-SL3			38-2	C7
-SL4			38-2	B7
-SL5			38-2	B7
-SL6			38-2	B7
-SL7			38-2	A7
-SL8			38-2	D4

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
-SL9			38-2	D4
-SL0			38-2	D7
753SET			88-1	A6
A10	38-2	A8		
A11	38-2	C7, D6		
A12	88-2	D6		
A13	88-2	D6		
A14	38-2	C7		
A15	38-1	B4		
	38-2	B7, C7		
	88-3	C5		
A16	38-2	B7		
	88-2	C7		
A17	38-2	A7	88-1	B7
A18	38-2	B7		
A19	38-2	D4	88-1	C6
A20	38-2	A7, B7	88-1	A7
A21	38-2	C4		
A22	38-2	D4		
	88-1	A2		
A23	38-2	D4		
A24	38-2	D4		
A25	38-2	B4		
	88-1	B8		
A26	38-2	C4		
A27	38-2	A4		
	88-1	B8		
A28	38-2	C4		
	88-1	A2		
A29	38-2	B4		
A30	38-2	B4, C4		
	88-1	B8		
A31	38-2	A4, B4	88-3	D8
A35	38-1	B3		
A36	38-1	B3		
A37			38-1	A3
A38	88-1	A4		
A39			38-1	A3
A40	88-1	A4		
A41			38-1	A2
A42	88-1	B4		
A43			38-1	A2
A44	88-1	B4		
A46	88-1	C4		
A47			88-3	D3
A48	88-1	B4		
A49	88-2	D6		

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
A5	38-2	D7		
A50	88-1	A4		
A51	38-1	B3		
A52	88-1	A4		
A53	38-1	B4		
A54	88-1	B4		
A55	38-1	B4		
A56	88-1	B4		
A58	88-1	B4		
A60	88-1	C2		
A62	88-1	C4		
A64	88-1	C4		
A66	88-1	C4		
A68	88-1	C4		
A7	38-2	D7		
A70	88-1	A4		
A72	88-1	C4		
A73	88-2	A6		
A74	88-1	A4		
A77	88-3	D2		
A8	38-2	D7		
	88-1	B8		
A80			88-3	B7
A82			88-3	B7
A85			88-2	A8
A89			88-1	C5
A9	38-2	C7		
A91			88-3	C5
A92	88-3	C5		
A9MEMOK			88-1	C7
AC CLR	88-1	A6	88-3	D3
AC CLR'			88-2	B8
ACB/SAVE	88-1	D4		
ACB11			88-3	C6
ACB12			88-1	D4
ACB2			88-1	B7
			88-2	A3
ACD OUT'	88-2	B2		
ACD3 SEL'	88-2	D4		
ACD4 SEL'	88-2	C4		
ACS1SEL'	88-2	C4		
ACS2SEL'	88-2	B4		
ACTG0	88-1	D8	88-1	C8
			88-2	A8
ACTG1	88-1	D7	88-1	C8
			88-2	A8

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
ADD ONE'	88-2	D2		
ADDER TEST	88-3	A4	88-3	D6
ADDER0			88-3	C6
ALC	88-2	B7	88-2	B3, C3, C7
			88-3	D8
ALC'	88-2	B7	88-2	B8, C3, D3
ALC·(SKIP)'			88-3	C5, D7
AND	88-2	B7	88-2	C3
			88-3	C6
AND ENAB			88-3	D5
AND ENAB'	88-2	B7	88-1	A5
B12			38-1	A5
B14			38-1	A5
B16			38-1	A5
B17	88-1	C3		
B18	38-1	B2		
B20	88-1	C2	38-1	D5
B21	88-1	B3		
B22	38-1	B5		
B24	38-1	B5		
B25			88-3	B8
B26	38-1	B6		
B28	38-1	B6		
B29			88-2	C8
B30	88-1	C2		
B32			38-1	A6
B33	88-1	B2		
B35	88-1	C6		
B37	88-1	C2		
B39	88-1	B2		
B41	88-1	C2		
B42			38-1	A6
B43			38-1	A6
B44			38-1	A7
B45	88-1	B2	38-1	C8
B47	38-1	B7		
B48	88-1	A6		
B5			38-1	A4
B55			88-3	C8
B56	88-3	B8	38-1	A2, A5
B57			38-1	A6
			88-3	C8
B58			88-3	B8
B59			38-1	A3
			88-3	B8

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
B6			88-1	A7
B60			38-1	A4
			88-3	C8
B61			38-1	A6
			88-3	C8
B62			38-1	A7
			88-3	C8
B63			88-3	B8
B64	88-3	B8	38-1	A3
B65			38-1	A4, A7
			88-3	C8
B66	88-3	B8	38-1	A2
B68	38-1	B6		
B69	88-3	B4		
B7	38-1	C8		
	88-1	D2		
B70	38-1	B7		
B71	38-1	B7		
B73			38-1	A6
			88-3	C8
B74	88-1	C2	38-1	B8
B75			38-1	A4
B76	38-1	B2		
B77			38-1	A7
B79			38-1	A7
B8			38-1	A4
B80			38-1	D7
B82			38-1	A7
			88-3	C8
B83	88-1	B2		
B85			38-1	A8, D8
B86			38-1	B8
B87	88-1	D2	38-1	D6
B88	88-1	B2		
B9			38-1	A4
B90	88-1	D2		
B95			38-1	A5
			88-3	C8
CARRY	88-3	C5	88-3	C7
CARRY'	88-3	C5	88-3	C7
			89-1	C8
CLK FLOP	88-1	A6, A7	88-1	D3
			88-3	D3
CLR	88-1	A4		
CLR ION			88-2	C7
CLR ION'	88-1	B4		

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
CLR SKIP'	88-3	B3	88-3	B5
CON DATA'	88-1	A2	89-1	C8
CON RQ'	88-1	B8	89-1	C8
CON INST'	88-1	A2		
CONT+ISTP+MSTP'	89-1	B3	88-1	B8
CPU			88-2	A4
CPU CLK	88-1	A6	88-1	A5, C2, C5, C8, D2, D4 D5
			88-2	A5, A6
			88-3	A5, B5, C5, D5, D8
CPU INST	88-2	B7	88-1	A3, A4, A5, B5
			88-2	C8
			88-3	B7, C6
CPU INST'	88-2	B7	88-1	A4
CRY ENAB			88-3	C6, D7
CRY ENAB SAVE			88-3	D7
CRY OUT'			88-1	D7
			88-3	C6
CRY SET SAVE'	88-1	C7	88-3	B7
CRY SET			88-1	C7
			88-3	C5
D SET	88-2	C6	88-2	D7
D+E SET'	88-2	D7	88-1	B7
			88-2	D5
D+E SET+(TS3)'			88-1	B7
DATA0'	38-1	C7	38-1	A7
			88-3	C8
DATA1'	38-1	C7	38-1	A7
			88-3	C8
DATA10'	38-1	C4	38-1	A4
			88-3	B8
DATA11'	38-1	C4	38-1	A3
			88-3	B8
DATA12'	38-1	C3	38-1	A3
			88-3	B8
DATA13'	38-1	C3	38-1	A3
	88-3	B8		
DATA14'	38-1	C2	38-1	A2
	88-3	B8		
DATA15'	38-1	C2	38-1	A2
	88-3	B8		
DATA2'	38-1	C7	38-1	A7
			88-3	C8
DATA3'	38-1	C6		
	88-3	C8		
DATA4'	38-1	C6	38-1	A6
			88-3	C8

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
DATA5'	38-1	C6	38-1	A6
			88-3	C8
DATA6'	38-1	C5	38-1	A5
			88-3	C8
DATA7'	38-1	C5	38-1	A5
			88-3	C8
DATA8'	38-1	C5	38-1	A4
			88-3	C8
DATA9'	38-1	C4	38-1	A4
			88-3	B8
DATIA	88-1	B4	88-1	B3
DATIB	88-1	B4	88-1	A4
DATIC	88-1	B4	88-1	A4
DATOA	88-1	B4		
DATOB	88-1	B4		
DATOB'	88-1	B4	88-1	A4
DATOC	88-1	B4		
DCH	88-1	C6	88-1	C2
			88-2	D4
DCH LOOP ENAB	88-1	B2	88-1	D8
			88-3	D6
DCH LOOP'			88-1	C3
DCHA	88-1	D4	88-1	B3, C2, C6
DCHA SET'	88-1	C4	88-2	D7
DCHA'	88-1	C2		
DCHI	88-1	C2	88-1	B3
DCHM0'	88-1	C3		
DCHM1'	88-1	B3	88-1	C2
			88-3	D6
DCHO	88-1	C2		
DCHR PEND	88-1	C5	88-3	D6
DCHR'	88-1	C6		
DEFER	88-2	D6	88-2	C7, D4
			88-3	D6
DEFER AGAIN	88-2	C7		
DEFER'	88-2	D6	88-2	C4, C7
			88-3	A6
			89-1	C2
DISABLE D MULT	88-2	B2		
DIV'			88-3	B5
DRIVE IO'	88-1	B2	38-1	C8
DS0'	88-1	C4		
DS1'	88-1	C4		
DS2'	88-1	C4		
DS3'	88-1	C4		

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
DS4'	88-1	C4		
DS5'	88-1	C4		
DSZ·E·TS0	88-2	B4		
DSZ·E·TS01			88-2	C3
E SET	88-2	C6	88-2	D7
EFA	88-3	D5	88-2	A4, A5, B3, C3, C8, D5
				D7
EFA'	88-3	D4	88-2	A3, C5, D5
EFA·(PTG1)'	88-2	A2	88-2	A4
END CYCLE	88-1	D5	88-1	C8
			88-3	C5, D5, D8
END CYCLE			88-1	A6
EXEC	88-2	D6	88-2	B5
EXEC'			88-2	B5
			89-1	C1
EXT SELECT'			38-1	D7
EXT·LOAD'			88-3	D3
FETCH	88-2	D6	88-2	A7, B8, C6, D5
			88-3	B4
FETCH'	88-2	D5	88-2	B3, C7, D3
			89-1	C2
FETCH+DEFER	88-2	C7	88-2	B6
FETCH·TS01	88-2	D4	88-3	C5
FETCH3			88-1	C3
FORCE LOAD IR'			88-2	A8
GND LAMP			89-1	C8
HALT'	88-2	C7	88-1	B7, C2, D5
INH GATE A'	38-1	D2	38-2	A8
INH GATE B'	38-1	D2	38-2	A5
INH TRANS'	88-1	B2	38-1	C8
INH0	38-1	B7	38-2	D8
INH1	38-1	B7	38-2	D8
INH10	38-1	B4	38-2	C5
INH11	38-1	B3	38-2	C5
INH12	38-1	B3	38-2	C5
INH13	38-1	B3	38-2	B5
INH14	38-1	B2	38-2	B5
INH15	38-1	B2	38-2	B5
INH2	38-1	B7	38-2	C8
INH3	38-1	B6	38-2	C8
INH4	38-1	B6	38-2	B8
INH5	38-1	B6	38-2	B8
INH6	38-1	B5	38-2	B8

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
INH7	38-1	B5	38-2	A8
INH8	38-1	B4	38-2	D5
INH9	38-1	B4	38-2	D5
INHB0	38-2	D7		
INHB1	38-2	D7		
INHB10	38-2	C4		
INHB11	38-2	C4		
INHB12	38-2	B4		
INHB13	38-2	B4		
INHB14	38-2	B4		
INHB15	38-2	A4		
INHB2	38-2	C7		
INHB3	38-2	C7		
INHB4	38-2	B7		
INHB5	38-2	B7		
INHB6	38-2	B7		
INHB7	38-2	A7		
INHB8	38-2	D4		
INHB9	38-2	D4		
INHIBIT	88-1	C2	38-1	D2, D3
INHIBIT SELECT'			38-1	D8
INPUT	88-1	C5		
INPUT'	88-1	C4	88-1	B3, C3
INTA	88-1	A4		
INTR'	88-2	C8		
IO PLA	88-1	A4		
IO RST	88-1	C6		
IO SKIP	88-1	B4	88-3	B6
IO SKIP'	88-1	B4	88-1	C8
IO(F+D)	88-2	B5	88-1	C5, C8
IO.'			88-1	D2
IO.E	88-1	C8	88-1	A5, B5,
			88-2	C8
			88-3	D5, D7
IO.E'	88-1	C7		
ION	88-2	C7	88-3	B7
ION'	88-2	C7	89-1	D2
IR0'	88-2	A6	88-1	B7
			88-2	B3, B6
			88-3	B4, C6
IR0+SKIP	88-2	B6	88-2	B8
IR1'			88-2	B5, B6, C5
IR17	88-2	A7		
IR2'	88-2	A7	88-1	A3
			88-2	B5, B6

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
IR3	88-2	A7	88-2	C5
IR3'	88-2	A7	88-2	A8
IR4	88-2	A7	88-2	C8
IR4'			88-2	A7, B5, C5
IR5	88-2	A5	88-2	C7
			88-3	D6
IR5'	88-2	A6	88-1	B4
			88-2	B3, B8
			88-3	B3
IR5·IR6	88-2	B8	88-2	C8
IR6	88-2	A5	88-2	C3
IR6'	88-2	A6	88-1	B4, B6
			88-2	B3, B8
IR7	88-2	A5	88-1	C5
			88-2	B8, D3, C5
IR7'	88-2	A6	88-2	B3, C5, C8, D3, D7
ISTP'	89-1	B3	88-1	B7
ISZ.E·TS0			88-2	D3
ISZ·E·TS0'	88-2	B4		
JSR·EFA	88-3	C3	88-3	D5
JSR·EFA'	88-2	C7	88-3	C3, C5
KEY	88-1	C6	88-1	A2, B6, B7
			88-2	A7, B3
			88-3	B4, D6
KEY ENAB'	88-1	B8	88-1	C7
KEY SEEN	88-1	B8	88-1	B8, C6
KEY SEEN'			88-1	D8
			88-3	D8
KEY'	88-1	C6	88-1	B2, D3
			88-2	D3
			88-3	B3, C6
KEY-LOOP			88-1	A2
KEY·LOOP	88-1	C6	88-2	B3
			88-3	B3
KEYM	88-1	C6	88-1	A3, B7
			88-2	D4
KEYM SET'	88-1	B6	88-2	D7
KEYM.PL	88-1	C5	88-1	B6
			88-3	A3, C4
KEYM·PL.TS0'	88-3	C3	88-1	B2
LDA·'	88-2	B4		
LDAE'			88-3	C5
LOAD AC'	88-3	D2		
LOAD ACB	88-3	C2		
LOAD CRY'	88-3	C5	88-3	C5
LOAD IR	88-2	A6	88-1	D7
			88-2	A8
LOAD MBO'	88-3	B2		

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
LOAD PC'	88-3	B3		
LOOP	88-3	D5	88-1	B5, C3, D5
			88-2	C3
LOOP 2 SET			88-3	D5
LOOP 8 SET'			88-3	D5
LOOP SET			88-1	C5, D5
LOOP SET'			88-1	C5
LOOP'	88-3	D4	88-1	A2, C6, D3
			88-2	D5
MA LOAD'	88-1	D2	38-1	C8
			88-1	C7
MA1	38-1	C7	38-1	D8
MA10	38-1	C4	38-3	B7, D7
MA10'			38-3	B7, C7, C8, D7
MA10B	38-3	C8	38-3	C7
MA11	38-1	C4	38-3	B7, D7, D8
MA11'			38-3	B7
MA11B	38-3	D7		
MA11B'	38-3	D7		
MA12	38-1	C3	38-3	B7, B8, D7
MA12'			38-3	B7
MA12B	38-3	B7	38-3	B8
MA13	38-1	C3	38-3	A8
MA13B	38-3	A8	38-3	A3, A4
MA13B'	38-3	A8	38-3	A5, A6
MA14	38-1	C2	38-3	A8
MA14B	38-3	A7	38-3	A3, A4, A5, A6
MA14B'	38-3	A8	38-3	A3, A4, A5, A6
MA15	38-1	C2	38-3	A8
MA15B	38-3	A7	38-3	A3, A5
MA15B'	38-3	A8	38-3	A4, A6
MA2	38-1	C7	38-1	D8
MA3	38-1	C6	38-1	D8
			38-4	A7
MA3B	38-4	A6	38-5	A4, A5, A6
MA3B'	38-4	A6	38-4	A3, A4, A5, A6
			38-5	A6
MA4	38-1	C6	38-4	A8
MA4B	38-4	A7	38-4	A3, A4
			38-5	A3, A4
MA4B'			38-4	A5, A8
			38-5	A5, A6
MA5	38-1	C6	38-4	A4, A8
MA5B	38-4	A8	38-4	A3, A5, A6
			38-5	A3, A4, A5, A6

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
MA5B'	38-4	A8	38-4	A3, A4, A5, B6
			38-5	A3, A4, A5, A6
MA6	38-1	C5	38-4	A5, A8
MA6B	38-4	A7	38-5	A3
MA6B'	38-4	A8	38-4	A3, A4, A6
			38-5	A4, A5, A6
MA7	38-1	C5	38-4	D8
MA7B	38-4	D8	38-4	C7, D7
MA7B'	38-4	D8	38-4	B7, C7
MA8	38-1	C4		
MA8'			38-4	C8
MA8B			38-4	B7, C7, C8, D7
MA8B'	38-4	C8	38-4	B7, C7, D7
MA9	38-1	C4		
MA9B	38-4	B7	38-4	B8, C7, D7
MA9B'	38-4	B8	38-4	B7, C7
MAHB'			38-4	B6
MB CLEAR'	88-1	D2	38-1	B8
MB LOAD	88-1	C2	38-1	B8
MBC			88-2	C3
MBC10	88-2	A4	88-1	C4
			88-2	B8
MBC10'	88-2	A5	88-3	C7
MBC11	88-2	A3	88-1	C4
			88-2	B8
			88-3	C7
MBC11'	88-2	A4		
MBC12	88-2	A4	88-1	C4
			88-2	B8
			88-3	C5
MBC12'	88-2	A5		
MBC13	88-2	A3	88-1	C4
			88-2	B8
			88-3	B7
MBC13'	88-2	A4		
MBC14	88-2	A4	88-1	C4
			88-2	B8
			88-3	B7
MBC15	88-2	A3	88-1	C4
			88-2	B8
MBC15'	88-2	A3	88-3	B6
MBC8	88-2	A4	88-3	B7
MBC8'	88-2	A5	88-1	A4, B4
			88-2	B6
			88-3	B7
MBC9	88-2	A3	88-3	B6
MBC9'	88-2	A4	88-1	A4, B4
			88-2	B6

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
MBO 12 SAVE'	88-1	C7	88-2	C4, D4
MBO0'			38-1	A7
MBO1'			38-1	A7
MBO10'			38-1	A4
			89-1	D4
MBO11'			38-1	A4
			89-1	D4
MBO12			88-3	A6
MBO12'			38-1	A3
			89-1	D4
MBO13'			38-1	A3
			88-3	A6
			89-1	D3
MBO14'			38-1	A2
			88-3	A6
			89-1	D3
MBO15			88-3	A6
MBO15'			38-1	A2
			89-1	D3
MBO2'			38-1	A7
			89-1	D7
MBO3'			38-1	A6
			89-1	D7
MBO4'			38-1	A6
			89-1	D6
MBO5'			38-1	A6
			89-1	D6
MBO6'			38-1	A6
			89-1	D6
MBO7'			38-1	A5
			89-1	D5
MBO8'			38-1	A5
			88-1	C8
			89-1	D5
MBO9'			38-1	A4
			89-1	D5
MD SEL1'			88-2	C5
MD1	38-1	B7	38-1	C7
MD10	38-1	B4	38-1	C4
MD11	38-1	B4	38-1	C3
MD12	38-1	B3	38-1	C3
MD13	38-1	B3	38-1	C3
MD14	38-1	B2	38-1	C2
MD15	38-1	B2	38-1	C2
MD2	38-1	B7	38-1	C7
MD3	38-1	B6	38-1	C6
MD4	38-1	B6	38-1	C6
MD5	38-1	B6	38-1	C5
MD6	38-1	B5	38-1	C5
MD7	38-1	B5	38-1	C5
MD8	38-1	B5	38-1	C4
MD9	38-1	B4	38-1	C4

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
MDO1'			89-1	D7
MEM CLK	88-1	A6	88-1	C6, D6, D7, D8
			88-2	A8, C8
			88-3	D3
MEM'			88-2	A7
MEM0'	89-1	A5	88-2	A6, C7
			88-3	C6
			89-1	C8
MEM1'	89-1	A5	88-2	A7
			88-3	C6
			89-1	C7
MEM10'			88-2	A5
			89-1	C4
MEM11'			88-2	A4
			89-1	C4
MEM12'			88-2	A5
			89-1	C4
MEM13'			88-2	A4
			89-1	C4
MEM14'			88-2	A5
			89-1	C3
MEM15'			88-2	A3
			89-1	C3
MEM2'	89-1	A4	88-2	A7
			88-3	C6
			89-1	C7
MEM3'	89-1	A7	89-1	C7
MEM4'	89-1	A6	88-2	A7
			89-1	C6
MEM5'	89-1	A3	88-2	A6
			89-1	C6
MEM6'	89-1	A3	88-2	A6
			89-1	C6
MEM7'	89-1	A4	88-2	A6
			89-1	C5
MEM8'			88-2	A5
			89-1	C5
MEM9'			88-2	A4
			89-1	C5
MSKO'	88-1	A4		
MSTP'	89-1	B3	88-1	B7
MTG0	88-1	D6	88-1	C2, D2, D6
MTG0'	88-1	D6	88-1	C7
MTG1	88-1	D6	88-1	C2, D2
MTG1'	88-1	D6	88-1	B2
MTG2	88-1	D6	88-1	C2
MTG2'	88-1	D6	88-1	C2, D2
			88-2	B8
MTG3	88-1	D6	88-1	B2, C6
MTG3'	88-1	D6	88-1	C2, D2, D7

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
OVFLO	88-1	B2		
PACK	88-3	D5	88-2	C5
PACK'	88-3	D4	88-2	C5
			88-3	D3
PC ENAB'	88-3	B3	88-2	C7, D5
PC IN'	88-2	D4		
PI	88-2	D6	88-2	B3, D4, D5
			88-3	A4
PI SET	88-2	C6	88-3	B3
PI'	88-2	D6	88-2	A8, C7, D3
			88-3	D6
PL'	89-1	B2	88-1	B7, C6
			88-2	B4
PRESET'	88-1	B7	88-1	A5, B5, C5, D5, D7
			88-2	D7,
			88-3	B5
PTG-1·TS3'	88-1	D3		
PTG0	88-1	D4	88-1	C3, C4, D4
PTG0·TS0	88-1	A5		
PTG0·TS0'	88-1	A5		
PTG1	88-1	D4	88-1	C3, C4, D4, D5
			88-2	A3
PTG1'	88-1	D4	88-2	C3
PTG2	88-1	D3	88-1	B5
PTG2'	88-1	D3	88-1	C5, D4, D5
			88-3	A6
PTG2·(LOOP)'	88-1	D4	88-3	C3, D6
PTG5	88-1	D4	88-1	C5, C6, C8
			88-2	D7
			88-3	A3, A5
PTG5 ENAB'	88-1	D3	88-1	D5
			88-3	B5, C6
PTG5'			88-1	B2
			88-3	C3
PTG=0·TS0			88-1	D2
			88-2	D3
PTG=0·TS0'			88-2	D3
			88-3	C4
PTG=0·TS3	88-1	D3	88-1	C5, D7
			88-2	D3
PTG=0·TS3'	88-1	D3	88-1	D4
			88-2	D3
PTG=0·TS9			88-3	A6
PTG=1·TS0'	88-1	D3	88-3	A6
PTG=1·TS3'			88-1	A5
PULSE ENAB	88-1	A5	88-1	B2
PWR FAIL			88-3	D8
PWR FAIL'			88-1	A7
PWR LOW			88-3	B7, D7
PWR LOW'			88-2	C7
			88-3	C8, D7

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
R2B			38-4	A8
			38-5	A7
READ IO'	88-1	B2	38-1	A8
READ1'	88-1	D2	38-1	D6
			88-1	C7
READ1B	38-1	D5	38-3	A7
READ2'	88-1	D2	38-1	D6
READ2B	38-1	D5	38-4	A7, B7
RELOAD DISABLE'			38-1	B8
RESET'	88-1	B7	88-1	A4, C7, C8, D5
			88-2	C7
			88-3	D5
RESTART			88-2	A7
RESTART ENABLE	89-1	B7		
RESTART'			88-1	A8
			88-2	B4
RESTART·KEY'	88-2	A7		
RINH B	38-2	D4		
RINH0	38-2	D7		
RINH1	38-2	D7		
RINH10	38-2	D4		
RINH11	38-2	C4		
RINH12	38-2	C4		
RINH13	38-2	B4		
RINH14	38-2	B4		
RINH15	38-2	B4, B7		
RINH2	38-2	C7		
RINH3	38-2	C7		
RINH4	38-2	C7		
RINH6	38-2	B7		
RINH7	38-2	A7		
RINH9	38-2	D4		
RQENB'	88-1	C2		
RST'	88-1	B8		
RUN	88-1	C6	88-1	A7
RUN'	88-1	C6	88-1	A6, B8, D2
S0	88-2	C2		
S1	88-2	C2		
S2	88-2	C2		
SARD			38-1	D8
SELB'			88-3	B7
SELD'			88-3	B7
SELECT	38-1	D7	38-1	B6, B8, C8, D3, D4, D5, D6
SERIAL CRY	88-1	D7	88-1	B2
			88-2	D3
SET ION'	88-1	B4	88-2	C7
SHIFT ACB	88-3	C2		

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
SHIFT10'			88-3	A6
SHIFT11'			88-3	A6
SHIFT12'			88-3	A6
SHIFT13'			88-3	A6
SHL'	88-2	B6	88-3	C6
SHR'	88-2	B6	88-3	C6
SKIP	88-3	B5	88-2	D4
SKIP INC'	88-1	C7	88-1	D3
			88-2	D5
			88-3	B3, B4
SKIP'	88-3	B4	88-2	B6, C7, D3
			88-3	D8
SNS0'	38-2	D6	38-1	B7
SNS1	38-2	D6		
SNS1'			38-1	B7
SNS10'	38-2	A7	38-1	B4
SNS11'	38-2	C3		
SNS12'	38-2	C3	38-1	B3
SNS13'	38-2	B3	38-1	B3
SNS14'	38-2	B3	38-1	B2
SNS15	38-2	A3	38-1	B3
SNS2'	38-2	C6	38-1	B7
SNS3'	38-2	C6	38-1	B6
SNS4'	38-2	C6	38-1	B6
SNS5'	38-2	B6	38-1	B5
SNS6	38-2	B6		
SNS6'			38-1	B5
SNS7'	38-2	A6		
SNS8'	38-2	D3	38-1	B4
SNS9'	38-2	D3	38-1	B4
STA·E'	88-2	B4	88-3	B3
STOP INH			88-2	D7
STOP INH'	88-1	B6	88-1	C5, C8
STOP SYNC			88-1	B7
			88-3	D7
STOP'			88-3	D8
STRB A	38-1	D4		
STRB B	38-1	D4	38-2	A6
STRB C	38-1	D4		
STRB D	38-1	D4	38-2	A3
STROBE	88-1	D2	38-1	D5
STRT	88-1	A4		
STUTTER'	88-1	D7	88-1	A7
SWP'	88-2	B6		
TEST SKIP			88-3	B6, D7
TEST SKIP 3 SET			88-3	D7
TEST SKIP SET			88-1	B6
TEST'	88-3	C5		

' Indicates "NOT"

NOVA 1200/1210/1220 SIGNAL LIST

CPU#001-000088-17, CONSOLE#001-000089-08 & MEMORY#001-000238-07

	SOURCE		DESTINATION	
	PAGE	GRID	PAGE	GRID
TS0	88-1	C5	88-1	D4
			88-2	A8, B3, C3, D5
			88-3	B4, C4
TS3	88-1	C5	88-1	B5, D4, D8
			88-2	B3, B8, D5, D7
			88-3	B4, D6
TS3SET			88-1	C5
WAS JSR	88-3	D5	88-2	C5
WAS JSR'			88-2	C5
WHOA'			88-1	A7
WRITE MEM	38-1	D2	38-3	A7
			38-4	A7, B7
WRM			38-4	A3
			38-5	A7
XRS	38-3	B3	38-3	B7
XWS	38-3	B3	38-3	A3
Y00	38-4	B3	38-5	B7
Y01	38-4	B3	38-5	B7
Y02	38-4	C3	38-5	C7
Y03	38-4	C3	38-5	C7
Y04	38-4	C3	38-5	C7
Y05	38-4	C3, D3	38-5	D7
Y06	38-4	D3	38-5	D7
Y07	38-4	D3	38-5	D7
YRS	38-4	B3	38-4	B7
YWS	38-4	B3	38-4	A3
			38-5	B7

' 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"

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
DCH	Data Channels
DCHA	Data Channel Acknowledge
DCH INC	Data Channels Increment
DCHI	Data Channel In
DCH LOOP ENAB	Data Channel Loop Enable
DCHM(0 or 1)	Data Channel Mode (0 or 1) Code type of Data Channel Cycle requested by Device
DCHO	Data Channel Out
DCHP IN	Data Channel Priority In
DCHP OUT	Data Channel Priority Out
DCHR	Data Channel Request
DEFER	Defer (instruction execution state)
DISABLE D MULT	Disable Destination Multiplexer
DIV	Divide (instruction)
DP	Deposit
DPN	Deposit Next
D MULT	Destination Multiplexer
D SET	Defer Set
DSZ	Decrement and Skip if Zero (instruction)

ABBREVIATIONS (Continued)

DS0-DS5	Device Select lines 0 thru 5
D+E SET	Defer or Execute Set
EFA	Effective Address
EX	Examine
EXN	Examine Next
E SET	Execute Set
INH GATE A	Inhibit Gate A (Memory)
INH GATE B	Inhibit Gate B (Memory)
INH TRANS	Inhibit Transmission
INH0-INH15	Inhibit Register outputs 0 thru 15 (Memory)
INTA	Interrupt Acknowledge
INTP IN	Interrupt Priority In (to Device)
INTP OUT	Interrupt Priority Out (from Device)
INTR	Interrupt (Bus Signal from Device)
IO (F+D)	IO (instruction) (Fetch or Defer state)
IO or I/O	Input/Output
ION	Interrupt On
IO PLS	Input/Output Pulse
IORST	Input/Output Reset
IO SKIP	Input/Output Skip (instruction)
IR0 thru IR7	Instruction Register outputs 0 thru 7
ISTP	Instruction Step (Console switch)
ISZ	Increment and Skip if Zero (instruction)
JMP	Jump (instruction)
JSR	Jump to Subroutine (instruction)
KEYM	Key Memory (access cycle)
LOAD AC	Load Accumulator
LOAD ACB	Load Accumulator Buffer (Shifter)
LOAD IR	Load Instruction Register
LOAD MBO	Load Memory Bus Outputs (CPU Interface Register)
LOAD PC	Load Program Counter
MA1 thru MA15	Memory Address Register outputs 1 thru 15
MA LOAD	Load Memory Address Register
MB CLEAR	Memory Buffer Clear
MBC8 thru MBC15	Memory Buffer Computer outputs 8 thru 15
MB LOAD	Load Memory Buffer Register
MBO0 thru MBO15	Memory Bus Outputs (CPU Interface Register) 0 thru 15
MD SEL1	Multiply Divide Select 1
MD1-MD15	Memory Data 1 thru 15
MEM CLK	Memory Clock
MEM OK	Power Supply Output Memory Voltage at correct level
MEM 0 thru MEM15	Memory Bus lines 0 thru 15 (to CPU)
MSKO	Mask Out (instruction)
MSTP	Memory Step (Console switch)
MTG0 thru MTG3	Memory Timing Generator (signals) 0 thru 3
MULT0 thru MULT3	Multiplexer Output (signals) 0 thru 3
OVFLO	Signal to Device that memory location being incremented or added to (Via Data Channels) has Overflowed
PC	Program Counter
PC ENAB	Program Counter Enable
PC IN	Program Counter In
PEND	Pending, e.g., INT PEND
PI	Program Interrupt
PI SET	Program Interrupt Set
PL	Program Load
PTG5 ENAB	Processor Timing Generator 5 (pulse) Enable
PTG0 thru PTG5	Processor Timing Generator (signals) 0 thru 5

ABBREVIATIONS (Continued)

PULSE ENAB	Pulse Enable (PTG and TS3 function)
PWR FAIL	Power Fail
READ IO	Read IO (Device Controller)
RINH0 thru RINH15	(Collector) Resistor, Inhibit Driver
RQENB	Request Enable
RST	Restart (Console switch)
SARD	Selected Address
S BUFFER	Source Buffer
SELB	Selected Busy (Bus signal)
SELD	Selected Done (Bus signal)
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	Teletypewriter
TTI	Teletypewriter In (Teletype Keyboard/Reader Buffer)
TTO	Teletypewriter Out (Teletype Teleprinter/ Punch (Buffer)
XRS	X (plane) Read Source (Memory Stack)
XWS	X (plane) Write Source (Memory Stack)
YRS	Y (plane) Read Source (Memory Stack)
YWS	Y (plane) Write Source (Memory Stack)
32 VNR	+32 Volts, Not Regulated
+VINH	+ (Memory) Inhibit Voltage
$+V_{Lamp}$	+ Lamp Voltage (Console indicators)
+VMEM	+ Voltage Memory
+5 OK	+5 Volt (power) operating properly

CUT ALONG DOTTED LINE

READERS COMMENT FORM

DOCUMENT TITLE: .....

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

How did you use this publication?

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

Did you find the material:

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

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

COMMENTS

PAGE	COL	PARA	LINE	FROM	TO

From:

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

Data General Corporation
ENGINEERING PUBLICATIONS
COMMENT FORM
DG-00935

FOLD DOWN FIRST FOLD DOWN

FIRST CLASS
PERMIT NO. 26
SOUTHBORO
MASS. 01772

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL

Postage will be paid by:

DataGeneral
Southboro, Massachusetts 01772

ATTENTION: Engineering Publications

FOLD UP SECOND FOLD UP

STAPLE