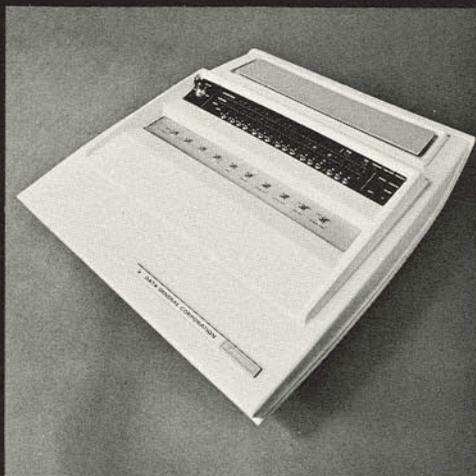


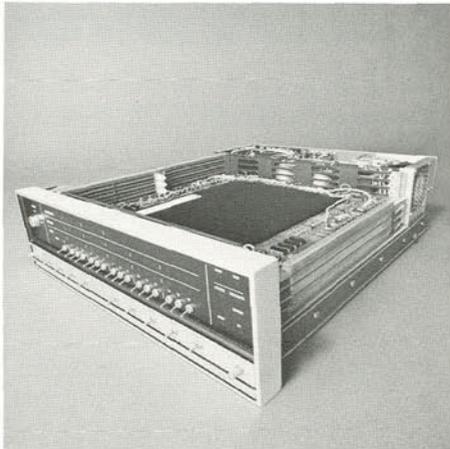
NOVA



SUPER NOVA

NOVA

Nova is a 16-bit word, small-scale, general-purpose, digital computer. It has four accumulators, two of which may be used as index registers. In its basic configuration, the Nova includes 4,096 16-bit words of core memory, Teletype interface, and input/output facilities including a high-speed data channel and automatic interrupt source identification.



NOVA, RACK MOUNTABLE

Nova is one of the most popular of the small-scale, general-purpose computers. Today there are hundreds of Novas installed.

Additional core memory is available in 2K and 4K blocks. Core can be expanded within the basic Nova cabinet to 20K and to 32K using an optional expansion chassis. Nova read-only memory is interchangeable with core and can be added in 1K blocks.

Nova comes in a desk-top model or in a rack-mountable version. The rack-mountable version is 5 $\frac{1}{4}$ " high and slides into standard 19" computer rack.

Nova is one of the most popular of all small computers. Nova number 100 was delivered a little more than 6 months after Nova number 1, and the second hundred took half as long. There are now hundreds of Novas delivered and working, and the number will be well over 1,000 before the first machine is 18 months old.

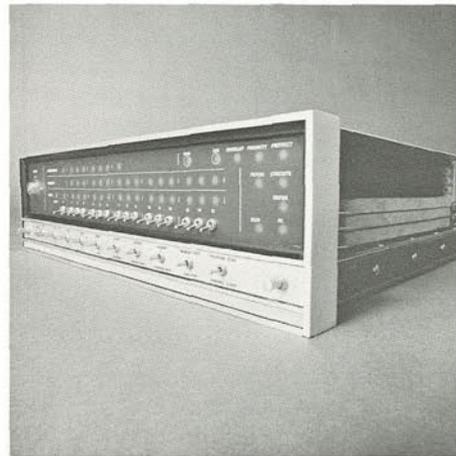
The most liberal quantity and OEM discounts available are offered with the Nova. Minimum configurations with less than 4K of core memory or read-only memory alone are available for controller applications.

SUPERNOVA

Supernova has the same basic organization as Nova: 16-bit word, 4 accumulators, flexible I/O facility, interchangeable core and read-only memories, the same packaging design. Supernova is a very fast small computer — considerably faster than any other computer at comparable cost. A full memory cycle using core memory takes 800 nanoseconds for the Supernova. Using read-only memory, cycle time is 300 nanoseconds.

Supernova overlaps the fetch and execute portions of arithmetic and logical instructions from read-only memory, so two numbers can be

added in one 300 nanosecond memory cycle. This technique has been used before in large-scale computer systems, but no other small-scale computer uses it.



SUPERNOVA, RACK MOUNTABLE

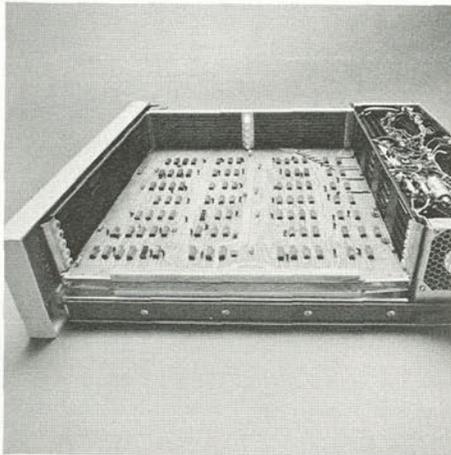
Supernova, which is completely compatible with Nova, is aimed at applications in which nanosecond speed is advantageous.

Because Supernova is considerably faster than other small computers, it is used in many applications in which other small computers have proved only marginally effective. A faster computer can monitor a high-speed phenomenon in a physics laboratory much more accurately than a slower computer, maintain closer control over a precision manufacturing process, or serve more terminals better in a retail billing system.

The basic Supernova configuration includes 4K of core memory, Teletype interface, and automatic program load. Memory allocation and protection, and multiply / divide are Supernova hardware options.

FOUR ACCUMULATORS

The Nova and Supernova, like most medium- and large-scale third-generation computer systems, have central processors organized around multiple general-purpose registers or accumulators. The logical and arithmetic instructions of these machines are performed by manipulating the contents of these accumulators. There is less need to address or access memory. And the availability of these multiple registers improves the efficiency of accumulator-to-memory operations and data flow between the computer and peripheral devices.



NOVA CPU

Multi-accumulator central processors enable Nova and Supernova to have a powerful class of arithmetic and logical instructions. These instructions combine operations for which other computers require several separate instructions.

The Nova and Supernova have four full sixteen-bit word accumulators, two of which may

be used as index registers. Data can be moved in either direction between any memory location and any accumulator. Arithmetic and logical operations are performed on operands in the accumulators, with the result appearing in an accumulator.

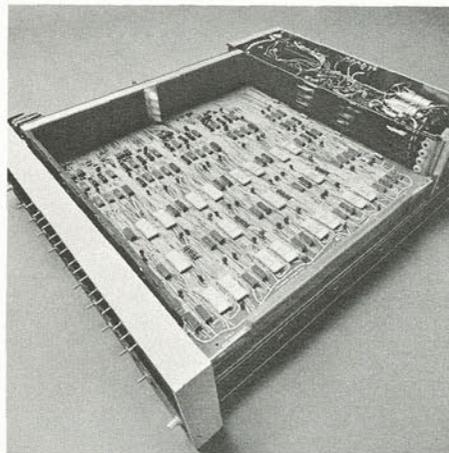
Since an arithmetic or logical instruction does not contain a memory address, there are many bits that can be used for functions other than specifying the basic operation and the operands. Arithmetic and logical instructions are frequently preceded by instructions which modify an operand and followed by a modification of the result and sometimes by a test. In the Nova and the Supernova, these operations are combined in a single instruction class. The result is a much simpler-to-use and more powerful instruction structure.

Arithmetic and logical instructions are arranged so that each bit has its own function. Thus it is unnecessary to decode most portions of the instruction word. The same instruction that adds or subtracts can also rotate right or left the 17-bit word (16-bit accumulator combined with the carry bit), or swap its right and left halves, test the result and/or carry for a skip, and specify whether or not the result shall actually be retained. The Supernova can execute any of these arithmetic and logical instructions in 300 nanoseconds.

An instruction that references memory can address two of the accumulators as index registers. A single input/output instruction can transfer a word between any accumulator and a device and at the same time control the operation of the device. Associated with the accumulators is a single carry flag, which indicates when the

magnitude of the result is too large to be accommodated in a single accumulator. The carry flag is also useful in double precision arithmetic.

This multi-accumulator organization cuts down on the number of instructions necessary to execute a program, and reduces the amount of data movement in the machine.



SUPERNOVA CPU

The Supernova uses the 24-pin dual-in-line integrated circuits shown on this central processor board.

The Nova and the Supernova are much easier to program than single accumulator machines. The results of address calculations are immediately available for index purposes to the memory reference instructions. One accumulator can be used for in-out data transmission without disturbing others being used continually for computation. Complex software routines such as multiplication, division, and floating point can be performed without constantly referencing memory.

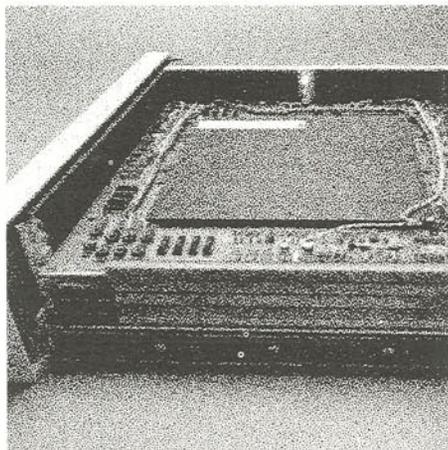


DATA GENERAL CORPORATION

NOVA

TWO KINDS OF MEMORY

Both Nova and Supernova have two compatible kinds of memory: core and read-only (ROM). The same programs run in core and read-only, and the two are physically interchangeable. Core can be expanded to 20K in the basic Nova cabinet, and to 16K in the basic Supernova cabinet. With an expansion cabinet, either machine can use 32K core.



NOVA CORE MEMORY
This Nova core memory board contains 4,096 16-bit words of storage.

Read-only memory is inherently more secure than core, since there is no way to write over or destroy information, either through an

operator error or external electrical noise. Programs that are used in very rugged environments or by untrained operators can be written and debugged in core, and then transferred to read-only memory. The read-only board is then plugged into the Nova or Supernova chassis. It can stay there permanently, or it can be plugged in only when required (as in the case of diagnostic programs).

Supernova read-only memory has the additional advantage of being very much faster than Supernova core. This is possible because the fetch and execute portions of instructions from read-only memory are overlapped. Add time is 300 nanoseconds in Supernova ROM, versus 800 nanoseconds in core.

A program written in Nova or Supernova core and then debugged and transferred to Supernova ROM will run considerably faster than it would in core. Because Supernova core and ROM are interchangeable and can be combined in the same computer, several different approaches to the use of ROM are open. The user may put all his applications packages in ROM. He may put only repetitive portions of programs in ROM, leaving variable portions of the program in core. Or frequently used mathematical routines can be stored in ROM, to be called from a program in core. The Nova floating point interpreter, for example, is available in a read-only memory module.



READ-ONLY MEMORY
Read-only memory modules are interchangeable with core in both Nova and Supernova.

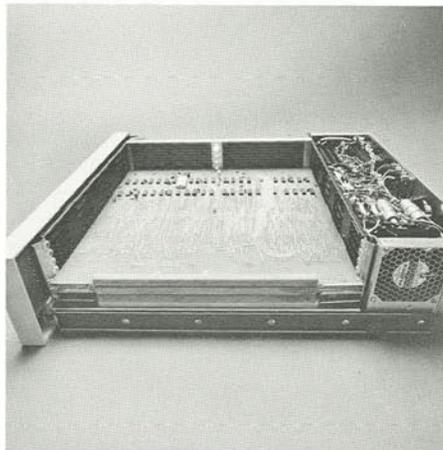
ROM can also be used to extend the instruction set of the Supernova by adding instructions that are especially useful for a specific application. The additional instructions required are constructed from several machine language instructions in core. This program is then wired into read-only memory. Supernova can often execute these extensions of its instruction set as fast as the machines in which they are implemented by hardware.

INPUT AND OUTPUT: FLEXIBLE

Small computers interface with more kinds of devices with greatly varying data rates and priority interrupt requirements than any other class of computer. Not only do they interface with the full range of computer peripherals, but they often become part of special systems.

The basic design of the Nova and Supernova computers includes a very flexible input/output facility. Included in the system are facilities for program interrupt and high speed data transfers with provision for direct access to memory. Any device can interrupt the normal program flow on a priority basis. A high-speed device such as magnetic tape or disk can gain direct access to memory through a data channel without requiring the execution of

any instructions. The data channel logic allows the transfer of data to or from memory, incrementing of a memory word, and addition of external data to a word already in memory. The latter two features facilitate such functions as pulse height analysis and signal averaging.



STANDARD I/O

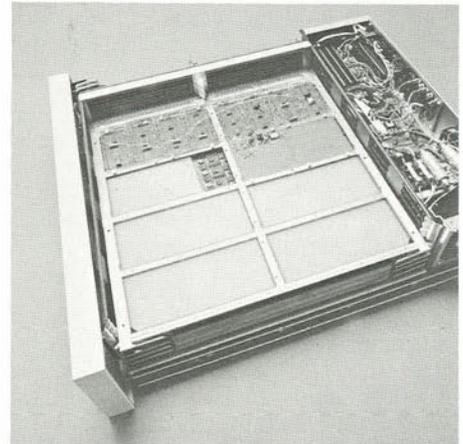
A standard printed circuit board is used to interface standard peripherals such as Teletype and paper-tape reader and punch.

The I/O facility of the Nova and the Supernova consists of 16 bi-directional data lines, 6 lines for device selection, 19 unary control lines from the central processor, and 6 control lines to the central processor. The control lines from the central processor are used to synchronize all data transfers on the data lines, to initiate and stop device functions, and to control the priority interrupt system. The control lines to the processor are used to indicate device status, and

to request priority interrupt and data channel service.

The unary control lines from the processor contain two types of information: the specific function to be performed by the device and timing information. These control lines are arranged in such a way that the device need connect only to those that correspond to the particular I/O functions that the device requires. The timing of the control lines is determined by the processor in such a fashion that the device does not require any time-dependent circuits to connect to the I/O interface.

The input/output system allows the program to address up to 62 external devices.

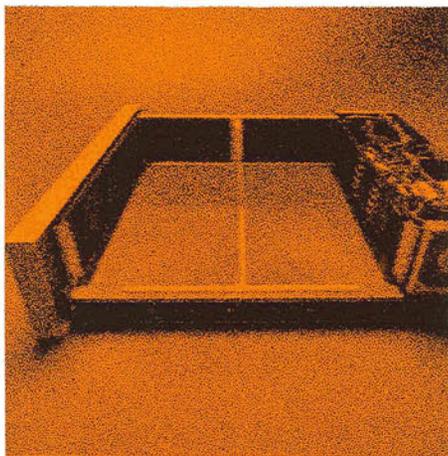


GENERAL PURPOSE INTERFACE

Customer-designed interfaces can be built on 15" x 15" general-purpose wiring boards that slide into the Nova or Supernova cabinet.

PACKAGING: SIMPLE

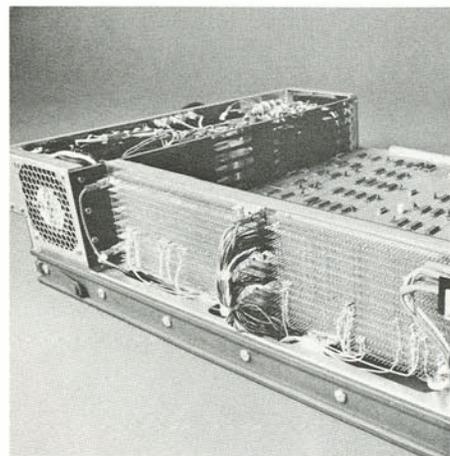
The rack-mountable versions of the Nova and Supernova computers are 5¼" high and slide into 19" racks. The chassis contains slots for seven 15" x 15" printed circuit boards. Each board has a 200-pin connector on one end and a handle on the other end. All seven slots connect into a common back panel.



NOVA / SUPERNOVA CHASSIS
The mechanical package of the Nova and the Supernova is built around seven stacked slots, into which 15" x 15" subassembly boards are plugged.

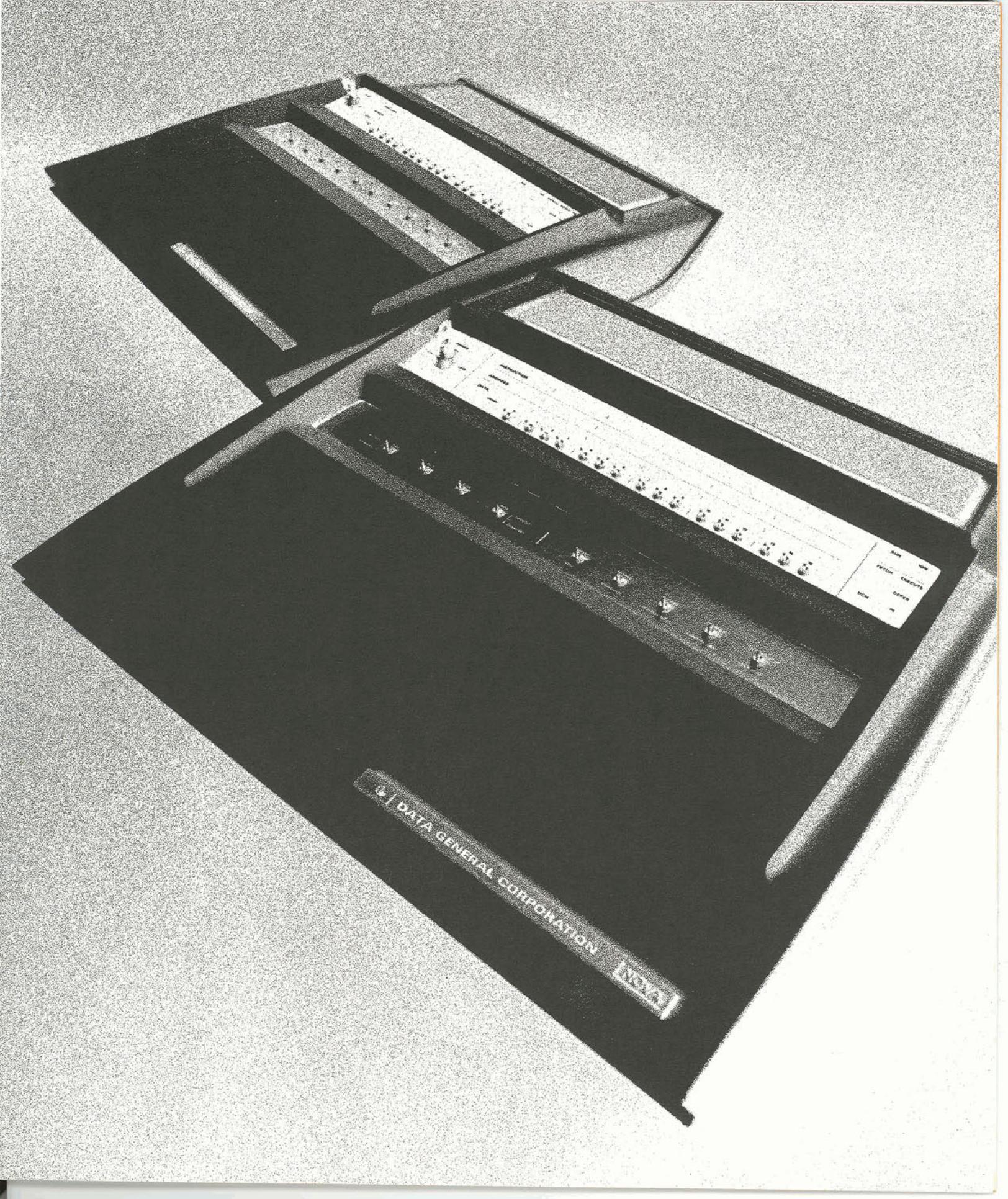
The back panel is a printed-circuit board itself, and it provides all of the interconnections between the boards. The only wire-wrapping required is for external connections — to the power supply and to I/O devices.

The Nova central processor is contained on two boards. The Supernova central processor occupies three boards. A 4,096-word core memory takes a single board in either computer, and the standard I/O interfaces fit on another board. Both Nova and Supernova have spare slots for additional memory or interfaces. Customer-designed interfaces can be built into general-purpose 15" x 15" wiring boards which slide into the chassis. Data General provides these wiring boards, plus several types of blank 3¼" x 6¾" printed circuit boards which fit on the large board. Most customer-designed interfaces will fit on one of these 15" x 15" boards, which can hold as many as 96 integrated circuit packages.



BACK PANEL
The seven slots of the Nova or Supernova chassis are interconnected through the back panel, which carries all interboard and external connections.

Packaging design contributes much to the low cost and reliability of the Nova and Supernova. Advanced printed circuit design, together with medium-scale integration, makes it possible to package large functional units of the computer on single printed circuit boards, drastically reducing internal interconnections. As the number of interconnections goes down, the cost of materials and labor drops. The overall packaging becomes simpler, less expensive, and more reliable. Service is simplified, because so few boards are involved.



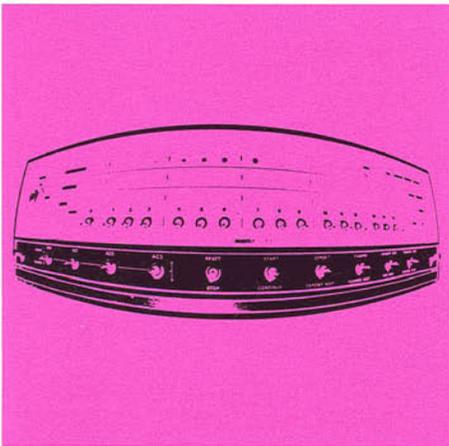
DATA GENERAL CORPORATION



MINI
SYSTEMS
SERIES
DIGITAL
DATA

SOFTWARE FOR BOTH

The same software runs on both the Nova and the Supernova. Programs developed on one machine will run, without modification, on the other.



NOVA FRONT PANEL

The multi-accumulator organization of the Nova can be seen in the functional front panel.

The Nova system software includes a standard assembler, a relocatable assembler, a relocatable linking loader, a character-oriented text editor, a completely symbolic debugger, single-user BASIC, timesharing BASIC, mathematical routines, floating point interpreter, and hardware diagnostics.

STANDARD ASSEMBLER

The standard assembler is a two-pass system producing absolute binary and an assembly listing. Pseudo commands are provided to alter assembly origin or radix and to define new operation codes. Text may also be processed and packed

into binary words. Input/output is fully buffered using the priority interrupt system, a binary search is used for the symbol table, and hence the assembly speed is I/O limited. The assembly language is free form. The input need not be precisely formatted into columns as is required by many small-computer assemblers. Control characters are used to delimit labels, comments and instruction fields. This provides greater freedom in the generation of program text as well as vastly reducing the errors due to missing spaces or blanks. The basic philosophy of the assembler has been to provide as few "default conditions" as possible. If it isn't entirely obvious what the user intended by a given line of code, the assembler will flag the line as questionable.

RELOCATABLE ASSEMBLER

The relocatable assembler includes all of the features of the standard assembler. In addition, it produces relocatable binary, allows the user to define double precision and double word floating-point constants, and has conditional assembly features.

RELOCATABLE LINKING LOADER

The relocatable linking loader, in combination with the relocatable assembler, loads programs and links global symbols using a single Teletype command. Another command initiates the printing of a loader map. The user can choose whether or not to load local symbols into a global symbol table. He can force a program to be loaded into a specific area that he has selected. The loader can also selectively load programs from a library tape.

EDITOR

The character-oriented text editor facilitates the editing of any type of text. Text editors for small computers are usually line oriented



SUPERNOVA FRONT PANEL

Automatic program load is standard on the Supernova. It is a convenience that few computers at the price level of the Supernova offer.

— all insertions and deletions are on a line basis. The Nova and Supernova editor works on a character as well as a line basis. The user changes minor text errors in comments, for example, without deleting the entire line. Single characters, character strings, whole lines and multiple lines may be inserted, deleted or replaced with single keyboard commands. Specific areas of text can be located quickly using string searches. This has eliminated the need for source line numbering (a function that otherwise must be done by the assembler or by the user himself). Eighteen commands allow the user to edit text more efficiently than is possible with editors having a limited command structure. In addition, all I/O is completely

buffered, allowing the user to input additional editor commands while output devices are still in the process of punching from their output buffers.

DEBUGGER

The debugger is completely symbolic. Any numerical value can be replaced by a user-defined or assembler-defined symbol. Memory searches and dumps, instruction examination and modification, and program patches can be made using instruction format commands, rather than octal commands. Program debugging is simpler for novices, since there is no need to know octal codes for all instructions, skip fields, I/O control fields, etc.

The symbolic debugger can accept from the relocatable loader a symbol table containing local as well as global symbols, so the user can employ symbolic names, rather than absolute addresses, while he is debugging a program. This eliminates the time-consuming calculation of absolute addresses while debugging a relocatable program.

BASIC

Two BASIC systems are available with Nova and Supernova: a single-user version and a timesharing version. BASIC is an easily-learned language that allows the programmer to solve problems using a number of common statements closely resembling simple algebra.

Single-user BASIC runs in any Nova or Supernova computer with at least 4096 words of core memory and Teletype. This version of BASIC makes it possible for a dedicated computer system to be used for general computation when it is not



PERIPHERALS

An extensive list of peripheral equipment is available for the Nova and the Supernova.

needed in its primary function.

Timesharing BASIC runs in any Nova or Supernova computer with at least 8192 words of core memory, and it can support eight users. It includes the matrix extension for the construction of matrices and the string extension, which permits the manipulation of alphanumeric

data. Timesharing BASIC is especially attractive as a very economical means of supplying a general computing capability in a technical environment. An eight-user BASIC system costs considerably less than eight desk calculators, and provides a great deal more computer power. It can also be less expensive than using a timesharing utility, and in many cases can offer comparable service with greater flexibility.

MATHEMATICAL ROUTINES

The mathematical routines provided for the Nova and Supernova computers range from input/output conversion to interpretive floating point. Representative routines include: single- and double-precision signed add, subtract, multiply, divide, absolute value, and negate; single and double-precision conversion of BCD to binary, binary to BCD, ASCII decimal to binary, and binary to ASCII decimal; single-precision conversion of ASCII octal to binary, binary to ASCII octal, Gray code to binary, and binary to Gray code; basic and extended floating point operations — conversion, arithmetic, and transcendental functions; and miscellaneous functions such as random number generation and parity generation.

FLOATING POINT INTERPRETER

An interpreter, rather than subroutines, provides floating point and related operations. The

interpreter is patterned after the Nova hardware architecture. Four floating point accumulators are implemented to manipulate floating point quantities. Register-to-register and memory-reference operations are defined to provide for complete floating arithmetic, decimal-to-floating and floating-to-decimal conversion, floating-to-fixed and fixed-to-floating conversion, plus six elementary transcendental functions, as well as square root. In addition, the manipulation of hardware accumulators and single-word memory locations is provided for in the interpretive mode. The instructions enable features such as indexing through floating point tables (two words per entry) and the execution of arbitrarily complex interpretive floating point subroutines. The assembler allows the definition of the thirty-six floating point operation codes using its symbol defining pseudo-ops, effectively doubling the programming capability of the machine. In addition, the interpreter is reentrant (therefore compatible with timesharing applications) and is available as a plug-in read-only memory module.

DIAGNOSTICS

Extremely thorough diagnostics are provided for both mainframe and peripheral diagnosing. The Nova central processor diagnostic, for example, makes a gate-by-gate check of the CPU logic, rather than a simple functional check. Faults are recognized and, in most cases, the faulty component is located. Peripheral diagnostics are equally thorough. For example, in addition to exploring the functioning of the Teletype, the Teletype diagnostic tests in detail the CPU interrupt hardware.

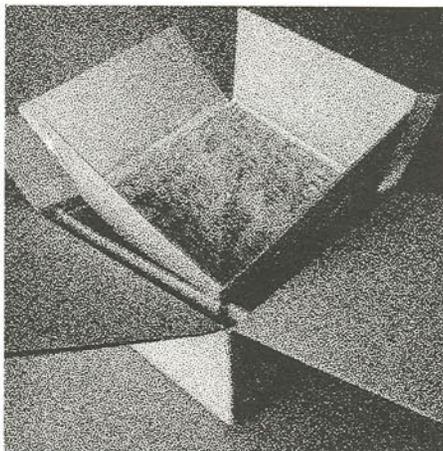
FOR CUSTOMERS

The Nova and the Supernova are supported in the field by Data General's trained service personnel. A number of service contract arrangements are available. Even more important, the computers are designed to be serviced with the least possible amount of trouble. Major functional subsystems are contained on individual printed circuit boards. Since there are at most seven boards in a single cabinet, it is usually possible for the customer himself to isolate the fault to a single board. Many quantity users and OEM's stock spare printed circuit boards, rather than purchase service contracts. They can swap spares into a

malfunctioning computer and mail the defective parts to Data General for repair without losing operating time.

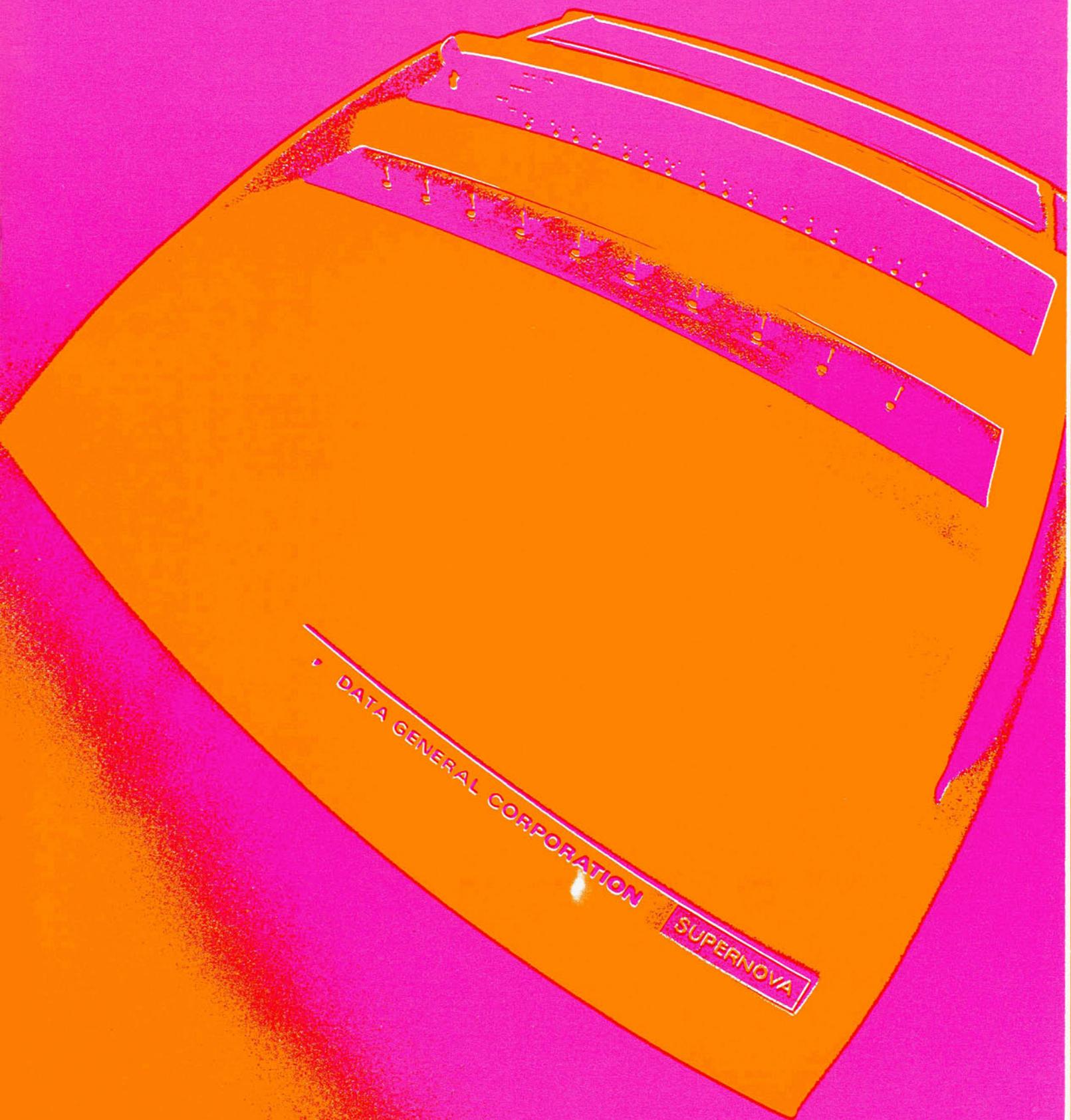
Customer training classes for Nova and Supernova maintenance and programming are held on a regular schedule at the Data General plant. Each customer also receives complete documentation, covering use of the computer, interfacing and installation, maintenance and software. Customer documentation is continuously updated, as is the expanding library of Nova and Supernova software.

Data General's regionally-based application engineers are available to support customers in the development of hardware and software for computer-based application systems.



SERVICE

A printed circuit board can be mailed to the Data General plant for diagnosis and repair, or it can be serviced in the field.



DATA GENERAL CORPORATION

SUPERNOVA

NOVA SPECIFICATIONS:

Nova is a 16-bit word, general-purpose computer. It has four accumulators, two of which may be used as index registers. It offers a choice of core or read-only memory. Read-only memory comes in blocks of 1K 16-bit words each, and core memory comes in blocks of 2 or 4K each. Nova comes in desk-top console or a 5¼" tall standard rack-mount package. Both the desk and rack versions can hold up to 20K 16-bit words of memory or interfaces for a large number of peripheral devices. Nova has the most flexible I/O facility ever built into a machine of its class. It includes a high-speed data channel and automatic interrupt source identification as standard equipment.

ELECTRICAL SPECIFICATIONS

Power Requirements

115 or 230 volts, 47 to 63 Hz single phase power capable of supplying 15 amperes (other frequencies and voltages available on special order)

Receptacle required to receive standard three wire plug

Power Dissipation

400 watts

I/O Bus Levels

Ground and +5 volts (standard TTL integrated circuit logic levels)

ENVIRONMENTAL SPECIFICATIONS

Operating Temperature

0°C to -55°C

Relative Humidity

To 90%

SUPERNOVA SPECIFICATIONS:

Supernova is also a 16-bit word, general purpose computer with four accumulators. Supernova core has a cycle time of 800 nanoseconds and comes in blocks of 4K. Supernova read-only memory has a cycle time of 300 nanoseconds and comes in blocks of 1K. Supernova comes in desk-top or 5¼" high rack-mount version. Both versions can hold up to 16K core memory or interfaces for multiple peripheral devices. Supernova has the same flexible I/O facility as the Nova, and includes all standard Nova options. In addition, automatic program load is standard on Supernova.

ELECTRICAL SPECIFICATIONS

Power Requirements

115 or 230 volts, 47 to 63 Hz single phase power capable of supplying 15 amperes (other frequencies and voltages available on special order)

Receptacle required to receive standard three wire plug

Power Dissipation

600 watts

I/O Bus Levels

Ground and +5 volts (standard TTL integrated circuit logic levels)

ENVIRONMENTAL SPECIFICATIONS

Operating Temperature

0°C to 55°C

Relative Humidity

To 90%

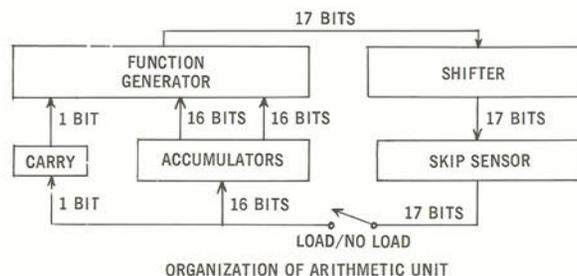
INSTRUCTIONS

ARITHMETIC AND LOGICAL INSTRUCTIONS

The structure of an arithmetic or logical instruction word is shown below.

1	AC SOURCE ADDRESS		AC DEST. ADDRESS		FUNCTION			SHIFT		CARRY	NO LD.	SKIP			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
					000	COM	01	L	01	Z		001	SKP		
					001	NEG	10	R	10	O		010	SZC		
					010	MOV	11	S	11	C		011	SNC		
					011	INC						100	SZR		
					100	ADC						101	SNR		
					101	SUB						110	SEZ		
					110	ADD						111	SBN		
					111	AND									

Each instruction in this class specifies one or two accumulators to supply operands to the function generator, which performs the function specified and produces a carry bit, whose value depends upon a base value specified by the instruction, the function performed, and the result obtained. The base value may be derived from the Carry flag, or the instruction may specify an independent value.



The 17-bit output of the function generator, comprising the carry bit and the 16-bit function result, then goes to the shifter. Here the 17-bit result can be rotated one place right or left, or the two 8-bit halves of the 16-bit function result can be swapped without affecting the carry bit. The 17-bit shifter output can then be tested for a skip; the skip sensor can test whether the carry bit or the rest of the 17-bit word is or is not equal to zero. The 17-bit shifted word can be loaded into Carry and one of the accumulators selected by the instruction. However, loading is not necessary: an instruction can perform a complicated arithmetic and shifting operation and test the result for a skip without affecting Carry or any accumulator.

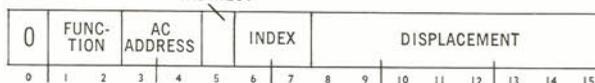
The Carry flag can be used in conjunction with the sign of a result to detect overflow in operations on signed numbers, but its primary use is as a carry out of the most significant bit in operations on unsigned numbers, such as the lower order parts in multiple precision arithmetic. For unsigned numbers, a carry is produced if addition or incrementing increases the number beyond $2^{16} - 1$. In subtraction, the condition is the same if, instead of subtracting, the complement of the subtrahend is added and 1 is added to the result (subtraction is performed by adding the twos complement). In terms of the original operands, the subtraction $A - B$ produces a carry if $A \geq B$.

The arithmetic and logical class includes eight functions: five arithmetic, three logical. In the following descriptions, ACS and ACD refer to the source and destination accumulators.

MEMORY REFERENCE INSTRUCTIONS

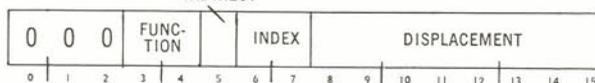
There are two formats for memory reference instructions, depending on whether an accumulator is specified.

WITH ACCUMULATOR INDIRECT



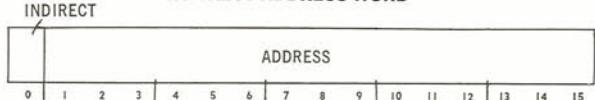
01 LDA
10 STA

WITHOUT ACCUMULATOR INDIRECT



00 JMP
01 JSR
10 ISZ
11 DSZ

INDIRECT ADDRESS WORD



Memory reference instructions must address a memory location. Each instruction word contains information for determining the effective address E, which is the actual address used to fetch or store the operand or alter program flow. The address information comprises an 8-bit displacement, a 2-bit index selection, and a single indirect bit. The displacement can directly address any location in four groups of 256 locations each. It can be an absolute address. That is, it may be used simply to address a location in page zero, the first 256 locations in memory. It can also be taken as a signed number used to compute an absolute address by adding it to a 15-bit base address supplied by an index register. The index bits can select AC2 or AC3 as the index register; either of these accumulators can thus be used as an ordinary index register to vary the address computed from a constant displacement, or as a base register for a set of different displacements. The program can also select the program counter as the index register, so any instruction can address 256 words in its own vicinity (relative addressing).

The computed absolute (15-bit) address can be the effective address. However, the instruction can use it as an indirect address. That is, it can specify a location to be used to retrieve another address. The word read from an indirectly addressed location contains an absolute address and an indirect bit; this address can be the effective address, or it can be another indirect address.

The program can make use of an automatic indexing feature by indirectly addressing any memory location from 00020 to 00037 (addresses are always octal numbers). Whenever one of these locations is specified by an indirect address, the processor retrieves its contents, increments or decrements the word retrieved, writes the altered word back into memory, and uses the altered word as the new address, direct or indirect. If the word is taken from locations 00020-00027, it is incremented by 1; if taken from locations 00030-00037, it is decremented by 1.

There are three pairs of memory reference instructions. Two instructions move data between memory and the accumulators; two modify a memory location and test the result for a skip; and two allow the

- COM Complement. Place the (logical) complement of the word from ACS in ACD.
- NEG Negate. Place the twos complement of the number from ACS into ACD. Complement Carry if ACS contains zero. (Forming the twos complement of zero generates a carry, because complementing zero produces a word containing all 1s, and adding 1 to that produces all zeros again, plus a carry.)
- MOV Move. Place the contents of ACS in ACD.
- INC Increment. Add 1 to the number from ACS and place it in ACD. If the result is 2^n , complement Carry.
- ADC Add Complement. Add the (logical) complement of the number from ACS to the number from ACD and place the result in ACD. If the original ACD = ACS, complement Carry.
- SUB Subtract. Add the twos complement of the number from ACS to the number from ACD and place the result in ACD. If the original ACD = ACS, complement Carry.
- ADD Add. Add the number from ACS to the number from ACD and place the result in ACD. If the result is $\geq 2^n$, complement Carry.
- AND And. Place the logical AND function of the word from ACS and the word from ACD in ACD.

These are the basic forms of the eight arithmetic and logical instructions in which the result is loaded, there is neither shifting nor skipping, and the present state of the Carry flag is used as a base value for carry generation (the Carry flag is complemented if a carry is generated by an arithmetic function, otherwise the original state is retained).

By appending other symbols to the basic mnemonics, the programmer can specify zero, 1, or the complement of the current state of the Carry flag as the base value for carry generation; can shift (rotate) the 17-bit function result with carry one place to the left or right, or swap halves of the 16-bit function result; and he can inhibit the loading of the 17-bit final result into ACD and Carry. He can also test the final result, whether loaded or not, for a skip as follows:

- SKP Always Skip.
- SZC Skip on Zero Carry.
- SNC Skip on Nonzero Carry.
- SZR Skip on Zero Result.
- SNR Skip on Nonzero Result.
- SEZ Skip if Either Carry or Result is Zero.
- SBN Skip if Both Carry and Result are Nonzero.

programmer to alter the normal program sequence by jumping to an arbitrary location. The modify-memory instructions are used to count loop iterations or successively modify a word for a series of operations. The jump instructions are especially useful for calling and returning from subroutines. In the following descriptions AC is the accumulator (if any) specified by the instruction, and E represents the effective address calculated from the address information given by the instruction.

- LDA Load Accumulator. Load the contents of location E into AC.
- STA Store Accumulator. Store the contents of AC in location E.
- ISZ Increment and Skip if Zero. Add 1 to the contents of location E and place the result back in E. Skip the next instruction in sequence if the result is zero.
- DSZ Decrement and Skip if Zero. Subtract 1 from the contents of location E and place the result back in E. Skip the next instruction in sequence if the result is zero.
- JMP Jump. Load E into PC. Take the next instruction from location E and continue sequential operation from there.
- JSR Jump to Subroutine. Load an address one greater than that in PC into AC3 (hence AC3 receives the address of the location following the JSR instruction). Load E into PC. Take the next instruction from location E and continue sequential operation from there.

INPUT/OUTPUT INSTRUCTIONS

The format for input/output instructions is shown below.

0	1	1	AC ADDRESS	FUNCTION TRANSFER	CONTROL	DEVICE CODE									
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
				000	NIO	01	S								
				001	DIA	10	C								
				010	DOA	11	P								
				011	DIB										
				100	DOB										
				101	DIC										
				110	DOC										
				111	SKP	00	BN								
						01	BZ								
						10	DN								
						11	DZ								

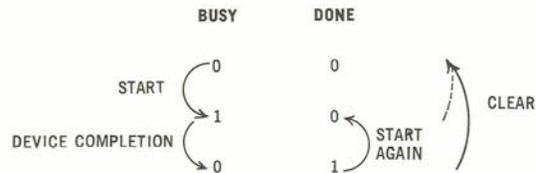
Input/output instructions govern all transfers of data to and from peripheral equipment, and perform various operations within the processor. An input/output instruction word has six bits for specifying the device. This format allows sixty-four device codes, of which sixty-two can be used to address devices (octal 01-76). The code 00 is not used, and 77 is used for special

functions, including reading the console data switches and controlling the program interrupt.

Every device has a 6-bit device selection network, an Interrupt Disable flag, and Busy and Done flags. The selection network decodes the device code part of the instruction so that only the addressed device responds to signals sent by the processor over the I/O bus. The Busy and Done flags together denote the basic state of the device. When both are clear the device is idle. To place the device in operation, the program sets Busy. If the device will be used for output, the program must give a data-out instruction that sends the first unit of data — a word or character depending on how the device handles information. When the device has processed a unit of data, it clears Busy and sets Done to indicate that it is ready to receive new data for output, or that it has data ready for input. In the former case the program would respond with a data-out instruction to send more data; in the latter with a data-in instruction to bring in the data that is ready. If the Interrupt Disable flag is clear, the setting of Done signals the program by requesting an interrupt; if the program has set Interrupt Disable, then it must keep testing Done or Busy to determine when the device is ready.

In all input/output instructions two bits either control or sense Busy and Done. In a skip instruction, the two bits specify the flag and the state on which the skip is to occur. In a transfer instruction, these bits can be used to specify a control function to be performed in addition to the transfer. Control functions are available to start the device by clearing Done and setting Busy; to clear both Busy and Done, idling the device; and to generate a special pulse whose effect, if any, depends on the device.

The overall sequence of Busy and Done states is determined by both the program and the internal operation of the device.



The data-in or data-out instruction that the program gives in response to the setting of Done can also restart the device. When all the data has been transferred the program generally clears Done so the device neither requests further interrupts nor appears to be in use, but this is not necessary. Busy and Done both set is a meaningless situation.

With a single device code the program can address up to three registers in the device. These are referred to simply as the A, B and C buffers. For each buffer there is a pair of transfer instructions, one to move data into an accumulator from the buffer, another to move data from an accumulator out to the buffer. Thus every one of these six transfer instructions must specify a device and an accumulator, and may specify a control function as well.

DIA Data In A
DOA Data Out A
DIB Data In B
DOB Data Out B
DIC Data In C
DOC Data Out C

The amount of data actually supplied or accepted by the device depends on the size of its buffer, its mode of operation, and so forth. The remaining input/output instructions specify a device and either a function or a skip condition.

NIO No IO Transfer. Perform the specified control function.
SKPBN Skip if Busy is Nonzero.
SKPBZ Skip if Busy is Zero.
SKPDN Skip if Done is Nonzero.
SKPDZ Skip if Done is Zero.

Input/output instructions with the device code 77 (CPU) perform a number of special functions rather than controlling a specific device. In a transfer instruction there may or may not actually be a transfer, but the start and clear control functions turn the interrupt on and off. The skip instructions sense the Interrupt On and Power Failure flags. In some cases the assembler recognizes a special mnemonic that includes both the instruction mnemonic and the CPU device code (these are given in the second column).

NIOS CPU INTEN
Interrupt Enable.
NIOC CPU INTDS
Interrupt Disable.
DIA AC, CPU READS
Read Switches. Read the contents of the console data switches into AC.
DIB AC, CPU INTA
Interrupt Acknowledge. Place in AC the device code of the first device on the bus that is requesting an interrupt ("first" means physically closest to the processor on the bus).
DOB AC, CPU MSKO
Mask Out. Set up the Interrupt Disable flags in the devices according to the mask in AC. For this purpose each device is connected to a given data line, and its flag is set or cleared as the corresponding bit in the mask is 1 or 0.
DIO, CPU
Clear IO Devices. Clear the control flip-flops, including Busy, Done and Interrupt Disable, in all devices connected to the Bus.
DICC O, CPU IORST
IO Reset. Clear all IO devices and disable the interrupt.
DOC O, CPU HALT
Halt the Processor.
SKPBN CPU
Skip if Interrupt On is Nonzero.
SKPBZ CPU
Skip if Interrupt On is Zero.
SKPDN CPU
Skip if Power Failure is Nonzero.
SKPDZ CPU
Skip if Power Failure is Zero.

MULTIPLY-DIVIDE INSTRUCTIONS

Hardware multiply-divide options are available for both the Nova and the Supernova. The Supernova option is built into the processor hardware and acts directly on the accumulators.

MUL Multiply. Multiply the unsigned integers in AC1 and AC2 to generate a double length product; add the product to the unsigned integer in AC0, and place the high and low order parts of the result respectively in AC0 and AC1 (in other words the result left in AC0 and AC1 is $AC0+AC1 \times AC2$).

DIV Divide. If the unsigned integer in AC0 is greater than or equal to the unsigned integer in AC2, set Carry and go immediately to the next instruction without affecting the original contents of the accumulators. Otherwise Divide the double length unsigned integer in AC0 and AC1 by the unsigned integer in AC2, producing a single length quotient including leading zeros, and then clear carry. Place the quotient in AC1 and the remainder in AC0.

In format, these instructions are actually of the input/output type, but in the Supernova they do not affect the I/O bus. The multiply-divide option for the Nova, however, is actually a device connected to the bus. This device contains A, B and C registers which correspond to accumulators 0, 1 and 2 for multiplication

and division; the registers are loaded and read by IO transfer instructions, and the arithmetic operations are triggered by the control functions that can be given with these instructions as follows:

Start Divide without overflow check.
Clear Clear the A register.
Pulse Multiply.

By following a standard procedure for loading the registers and triggering operations in the device, programming for the Nova and Supernova is compatible.

INSTRUCTION EXECUTION TIMES IN MICROSECONDS				
	Supernova		Nova	
	Core	Read-Only	Core	Read-Only
LDA	1.6	1.4*	5.2	5.0*
STA	1.6	1.4	5.5	5.3
ISZ, DSZ	1.8	1.6	5.2	5.0
JMP	.8	.6	2.6	2.4
JSR	1.4	1.2	3.5	3.3
Indirect addressing	.8	.6	2.6	2.4
Indexing	0		.3	
Autoindexing	.2		0	
COM, NEG, MOV, INC	.8†	.3†	5.6	5.4
ADC, SUB, ADD, AND	.8†	.3†	5.9	5.7
IO input (except INTA), IO skips	2.9	2.8	4.4	4.2
NIO	3.3	3.2	4.4	4.2
IO output	3.3	3.2	4.7	4.5
INTA	3.7	3.6	4.4	4.2
MUL			11.1	10.9
Average	3.8	3.7		
Maximum	5.4	5.3		
DIV	6.9	6.8	11.9	11.7
Interrupt	2.2		5.2	
Latency	9‡		12	
Data channel				
Input	2.1		3.5	
Output, increment	2.8		4.4	
Add	2.8		5.3	
Latency	9‡		12	

*If instruction and operand are both from read-only memory, Supernova time is 1.2 μ s, Nova time is 4.8 μ s.

†Time given must be doubled if a skip actually occurs.

‡5 μ s without multiply-divide.

OPTIONS

2K Core Memory (2048 16-bit words)

Includes 2K words (4K bytes) with all necessary electronics mounted on a single subassembly (15" x 15" printed circuit card) which can be plugged directly into one of the slots in the basic frame with no wiring modifications required. Available for Nova only.

4K Core Memory (4096 16-bit words)

Includes 4K words (8K bytes) of memory with all necessary electronics mounted on a single subassembly (15" x 15" printed circuit card) which can be plugged directly into one of the slots in the basic frame with no wiring modifications required.

1K Read-Only Memory Module (1024 16-bit words)

Includes 1K words (2K bytes) of read-only memory which is interchangeable with the alterable core memory with no wiring modifications required. The contents of this module can be either standard programs or special customer-specified programs.

Expansion Enclosure

This is a basic frame with a power supply and slots to mount seven subassemblies. This unit is typically mounted directly above the central processor frame and is used to mount additional memory (core or read-only), additional I/O controllers, or customer-designed hardware.

Power Monitor and Auto Restart

Provides power level detection and a flag which is attached to the program interrupt and can be sensed by the program. It allows the program to become aware of an imminent power failure so it can provide for an orderly shut down. The program automatically restarts at location 0 when power is restored.

Real-Time Clock

This option provides a flag which can be enabled by the program to provide a program interrupt at a fixed frequency. The AC line or a crystal clock may be program-selected as the time source.

Teletype Input/ Output Interface

This option provides an interface to any one of the Teletype models listed below:

Teletype ASR33

Keyboard/ printer, 8 channel reader/ punch, 10 char./sec.

Teletype KSR33

Keyboard/ printer 10 char./sec.

Teletype ASR35

Keyboard/ printer, 8 channel reader/ punch, 10 char./sec.

Teletype KSR35

Keyboard/ printer, 10 char./sec.

Teletype ASR37

Keyboard/ printer, 8 channel reader/ punch (upper/ lower case), 15 char./sec.

Teletype KSR37

Keyboard/ printer (upper/ lower case), 15 char./sec.

Teletype Modification Kit

Converts Teletype ASR 33TZ, TC or TU unit to on-line operation for use with Teletype Input/ Output Interface.

High-Speed Perforated Tape Reader and Control

Paper tape reader and control senses eight-channel, fan-fold, perforated Mylar or paper tape photoelectrically at 300 characters per second.

High-Speed Perforated Tape Punch and Control

BRPE11 Punch and Control punches eight-channel, fan-fold paper tape at 63.3 characters per second. A remote-operation modification allows power turn-on and turn-off under program control.

Card Reader and Control

Soroban SCCR card reader and control operates at 225 or 400 cards per minute.

Incremental Plotter

Plotters and control interfaces for the California Computer Products 500 Series units (drum or flat-bed) or The Houston Instruments Complot DP-1 Digital Plotter (uses Z-fold paper).

High-Speed Communications Controller

Used with high-speed full-duplex or half-duplex synchronous data sets (Bell 201 or equivalent). Allows automatic line synchronization, word assembly, and end-of-transmission recognition. (The SYNC and EOT characters may be changed under program control.) Data transfers are through the Data Channel. It accommodates character widths from 6 to 8 bits. An internal clock provides an adjustable oscillator when a modem is not actually employed. A parity option appends an even or odd parity bit to each character (6, 7, or 8 bits) on transmission and checks each character for even or odd parity on reception.

Disk and Disk Control

Control and interfaces for up to eight disks. Head-per-track disks have storage capacities of 32K, 64K, or 128K 16-bit words. Data transfer is through the data channel. Disks are rack-mountable.

Magnetic Tape Control

Controls up to eight synchronous read/write 7- or 9-track, industry-compatible tape transports.

Analog-to-Digital Converters and Interfaces

Analog-to-digital converters are available having 1 to 256 channels and word length of 8 to 14 bits. The analog-to-digital interface also runs 3½ digit panel meters.

Digital-to-Analog Converters and Interfaces

Digital-to-analog converters are available having 1 to 8 channels and word lengths of 8 to 14 bits. Digital-to-analog interfaces are provided.

Supernova Hardware Multiply/Divide

The Supernova multiply/divide unit multiplies two 16-bit numbers to produce a 32-bit product, and divides a 32-bit dividend by a 16-bit divisor to produce a quotient and a remainder.

Nova Hardware Multiply/Divide

The Nova multiply/divide option consists of three 16-bit registers that are loaded and read with I/O instructions. The unit multiplies two 16-bit numbers to form a 32-bit product and divides a 32-bit dividend by 16-bit divisor to produce a quotient and a remainder.

Supernova Memory Allocation and Protection (MAP)

The Memory Allocation and Protection option provides for instruction protection, memory mapping and memory protection, allowing a number of programs to share processor time.

Supernova High Speed Data Channel

This option allows very high speed data transfers between the Supernova and external devices by circumventing the standard Supernova data channel. Consecutive transfer rates range from 1MHz for the various data channel operations.

General Purpose Wiring Frame and Boards

Frame with 200-pin connector is 15" x 15" and slides into Nova and Supernova cabinets. Frame carries up to eight 3¼" x 6¾" general purpose wiring boards, which are available with or without wire-wrap pins and sockets.

NOVA

Nova is a 16-bit word, general-purpose computer. It has four accumulators, two of which may be used as index registers. It offers a choice of core or read-only memory. Read-only memory comes in blocks of 1K 16-bit words each, and core memory comes in blocks of 2 or 4K each. Nova comes in desk-top console or a 5¼" tall standard rack-mount package. Both the desk and rack versions can hold up to 20K 16-bit words of memory or interfaces for a large number of peripheral devices. Nova has the most flexible I/O facility ever built into a machine of its class. It includes a high-speed data channel and automatic interrupt source identification as standard equipment. The basic 4K configuration with Teletype interface costs \$7,950.

SUPERNOVA

Supernova is also a 16-bit word, general purpose computer with four accumulators. Supernova core has a cycle time of 800 nanoseconds and comes in blocks of 4K. Supernova read-only memory has a cycle time of 300 nanoseconds and comes in blocks of 1K. Supernova comes in desk-top or 5¼" high rack-mount version. Both versions can hold up to 16K core memory or interfaces for multiple peripheral devices. Supernova has the same flexible I/O facility as the Nova, and includes all standard Nova options. In addition, automatic program load is standard on Supernova. The basic 4K configuration costs \$11,700.

If you wish additional information and a chance to study "How to Use the Nova and Supernova," a rather complete reference manual, just drop the attached post card in the mail.



Southboro, Massachusetts 01772

