X6818.2013

GRI Computer

gri-909 COMPUTER

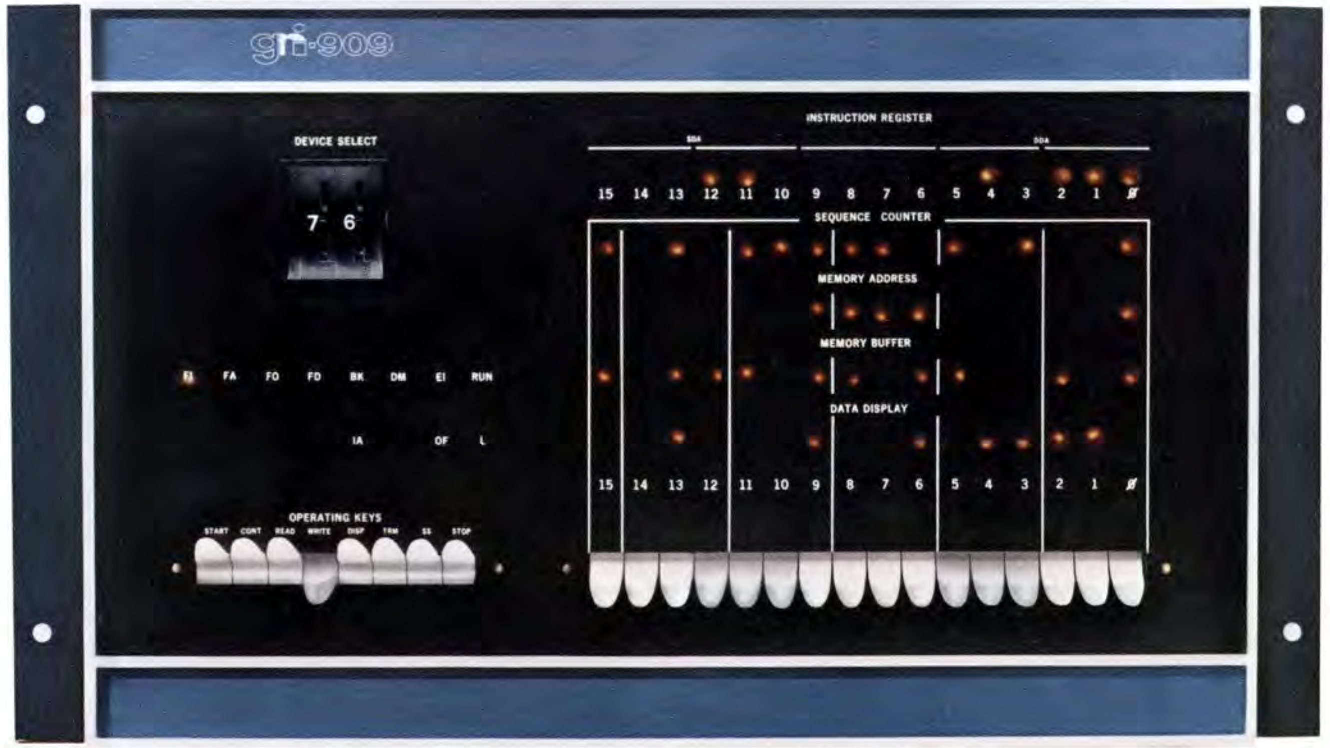
A DRAMATIC NEW APPROACH TO SYSTEMS-ORIENTED COMPUTER DESIGN

INTRODUCING THE GRI-909* A DIRECT FUNCTION PROCESSOR

The GRI 909 Computer is the first of a new class of general-purpose digital computers. Called a Direct Function Processor, it represents the next logical step in the use of computers as wholly-dedicated system controllers. The internal architecture and the machine language of the Direct Function Processor greatly enhance the system designer's ability to utilize computer control effectively and economically. Its programming language is oriented functionally, not arithmetically. The GRI 909 can be programmed in the kind of functional terms a systems engineer must use to define system operation. It keeps the responsibility of the system program in the hands of the designer by providing a means of relating systems functions directly to computer instructions.

The intrinsic modular design of the Direct Function Processor permits a variety of machine configurations ranging from highly economic, minimal processors for systems requiring simple data manipulations, to large configurations combining powerful computing instructions with a variety of peripheral devices. Hardwired firmware operators can be added in the form of plug-in modules to provide virtually thousands of computer instructions. These modules, or firmware operators, provide a flexibility and expandibility unequaled by conventional computer designs. Data transfer between system or computer devices is accomplished directly, in a single operation, without temporarily storing the data in special input/output registers or accumulators.

Essentially, Direct Function Processing provides expanded flexibility, considerable potential economy and systems-oriented programming, without sacrificing any of the capabilities inherent in the general purpose digital computer.



*PATENTS APPLIED FOR

GRI-909 FEATURES

- 16-bit fully parallel processor with 1.76 microsecond cycle time.
- 32,768 words of random access core memory, directly addressable — not page oriented. Minimum core size is 1024 words.
- Functional organization permits simple assembly language programming in user system language.
- Every device in the system, both inside and outside the computer, is directly addressable by programmed instructions, allowing direct data transfer between devices without special accumulators or need for temporary storage.
- Firmware options can expand hardwired instruction set to provide system flexibility unequaled by more conventional computer designs. These include: multiple arithmetic units, multiply options, byte pack, byte swap, square root, etc.
- Direct memory access channel is available on the same data and control lines as the programmed input/output channel (I/O rate: 1.76 microseconds per 16-bit word). No DMA multiplexer is required for multiple DMA devices.
- Priority interrupt system has full capability to be used as a single channel interrupt or as a full hardware interrupt at the option of the system designer.
- Core memory and read-only memory are interchangeable.
- Basic chassis occupies 10½ inches of standard 19-inch cabinet with space available for 8192 words of memory, three major firmware option modules and 16 firmware or device interface modules.
- Memory power fail protection and automatic restart are standard. Also, remote start and stop available for use in system interface.
- TTL integrated circuit and medium scale integrated circuit modules used for maximum economy.
- All system registers, both internal and external to the computer, can be displayed on console. Data can be transferred from console switches to all system devices.
- Peripheral options include mass memory media, input/output devices, communication interfaces, display and digital system devices.

The GRI 909 is the ideal system control computer. Its flexibility, modularity and ease of programming are intended to provide the original equipment or system manufacturer with a system control center which minimizes many of the problems inherent in conventional computer designs. The Direct Function Processing technique, implemented by the GRI 909, is the product of many years of experience in both the design of computers and the use of computers in systems. In the GRI 909, the internal computer devices are treated like functional elements of the overall system. The modular firmware capability of the GRI 909 provides the system designer with a flexibility to meet changing system requirements, the ability to incorporate his own proprietary and unique control features, and a hedge against obsolescence by the introduction of new system devices or circuitry. Its mechanical design simplifies its system integration, and its styling is compatible with virtually any system decor.

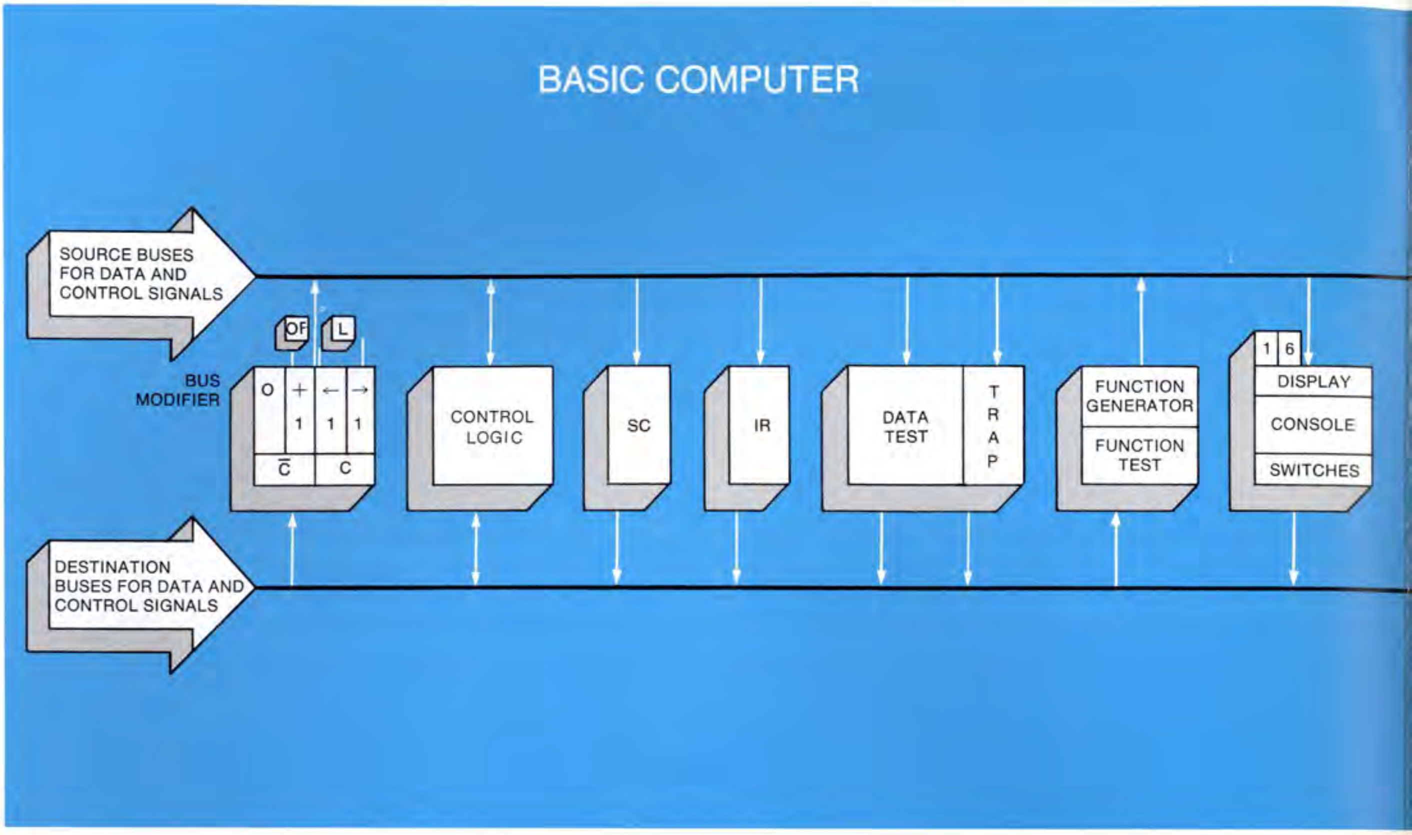
GRI Computer Corp.
76 ROWE STREET, NEWTON, MASSACHUSETTS 02166
(617) 969-7346

A NEW APPROACH TO COMPUTER ARCHITECTURE

The GRI 909 computer differs from the conventional computer in that the input/output data and control buses (groups of parallel wires) on which peripheral equipments reside, *extend inside the computer*. All of the internal elements of the computer, including the Instruction Register, Sequence Counter, Arithmetic Unit and Memory, are connected across these buses.

This type of structure is made effective by the addition of a programmable link between the input (Destination) and output (Source) buses. This programmable link provides a controlled path for transferring data directly, or with certain optional "on-the-fly" modifications between the buses. Since all functional components of the system are connected between the buses, any device can send a word in parallel directly to any other device, regardless of whether either device is internal or external to the computer. The result is a computer in which every processor component or input/output element is directly addressable, as if it were part of the internal processor logic. It simplifies the writing of programs and minimizes much of the "bookkeeping" associated with the manipulation of data.

Following is a more detailed examination of the arrangement and components of the GRI 909.



BASIC COMPUTER ELEMENTS

Bus Modifier — The bus modifier is designed to take information from an input device and move it to an output device, while performing simple operations on the data as it passes through.

This programmable path provides a means of getting any data word, from any functional component of the system, appearing on the destination buses back to the source buses to be delivered to any other functional component in the system.

It can transfer a data word from one device to another, complimented or not complimented, and can modify the data in one of the following ways: (1) no modification, (2) incremented by one, (3) shifted left one bit, (4) shifted right one bit. A two's complement (negative) number can be obtained, on the fly, by combining the one's complement operation with the increment by one operation. Associated with the Bus Modifier is a Link Bit through which data can be shifted for testing one bit at a time, and an Overflow Bit for tests involving incrementing data.

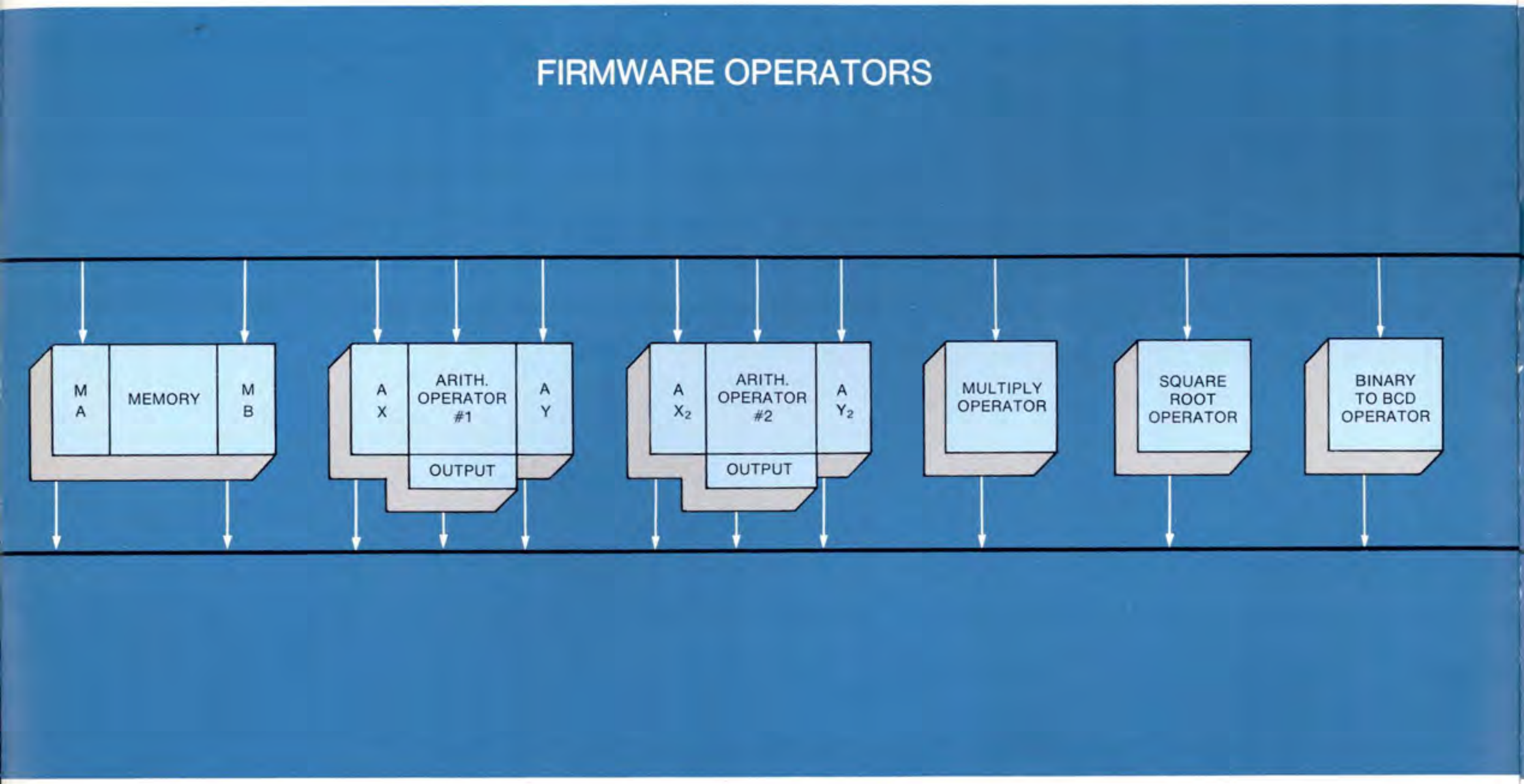
Sequence Counter — A device is provided to keep track of the program information. This program or sequence counter is common to all computers and indicates the address of the next instruction.

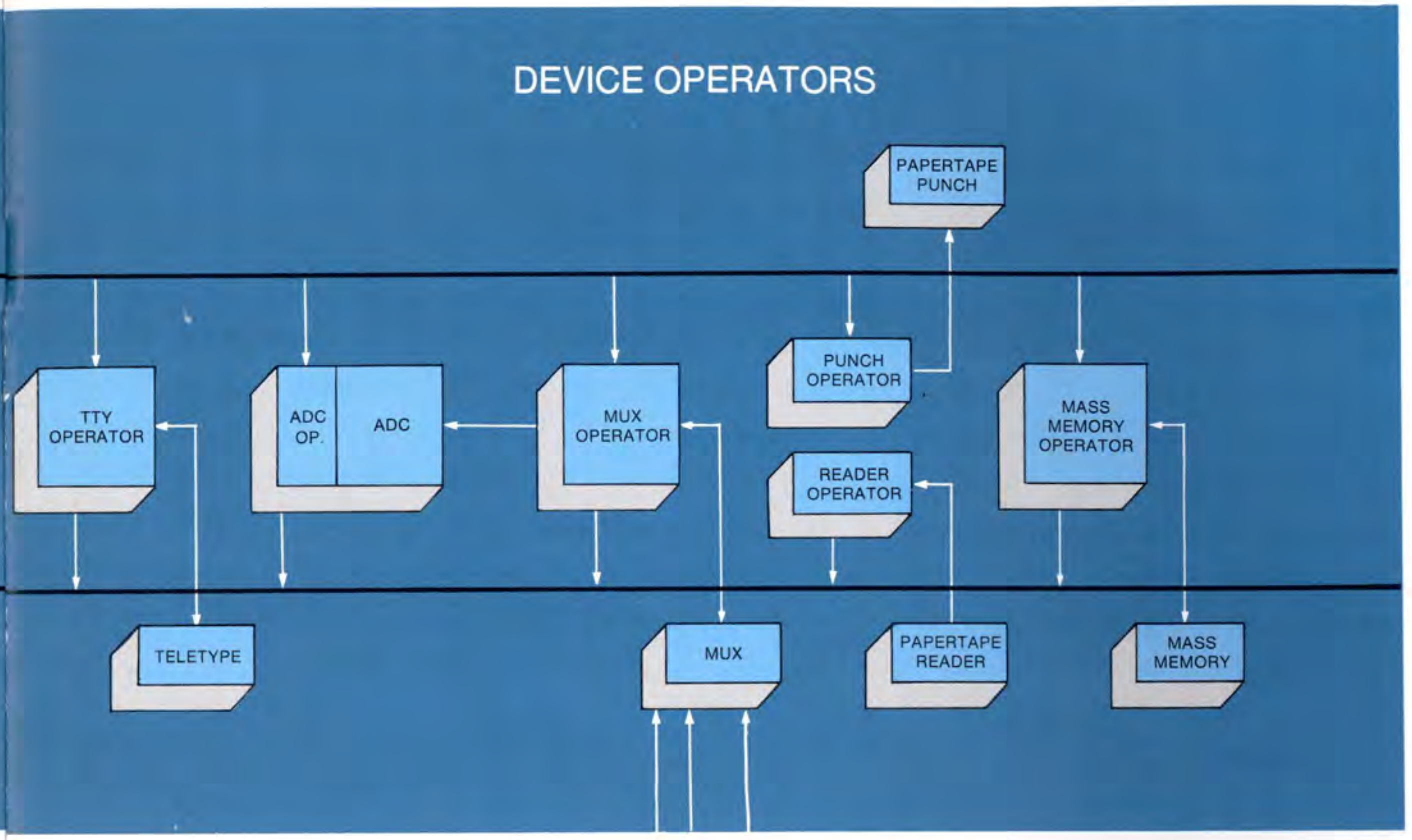
In the GRI 909, the sequence counter is connected across the buses, as are all other elements in the system, providing direct access from device to device. An external device can cause the program to go to some special subroutine in memory by transmitting an address word directly to the sequence counter.

Instruction Register — The Instruction Register contains the current instruction in the computer to be executed. Like other elements in this system organization, it is connected across the Source and Destination buses.

Data Test — A thought process that a computer performs is to decide on the paths that it will follow, based on the value of the data that it receives. In the GRI 909, Data Test determines whether the value of the information it receives is less than zero, equal to zero, or any combination thereof (including the negation). This tester is connected between the Source and Destination buses, and is programmed to accept data directly from any source. A positive response to a data test results in a jump instruction. The contents of the Sequence Counter are automatically stored in a trap register associated with Data Test when a jump is executed.

Function Generator — Most peripheral devices require control pulses to perform such functions as Start, Stop, Clear, etc. Up to sixteen different control commands can be issued to each system device by generating the address of the device and control pulses on four control lines provided for this purpose. A typical instruction to be issued by the function generator might be, "Start reader".





Function Test — Some devices produce status signals which indicate certain conditions to the computer. The GRI 909* contains a Function Test Operator that looks at these status signals and acts upon them.

Three control lines are provided for this purpose, plus a fourth which provides logical negation of the other three. A positive response by the Function Test Operator to the sense lines results in a skip instruction.

Console — If a peripheral device is added to a system, a set of lights may be required to indicate what is going on within the device.

When the switches on the console are set to a device address, any data that is delivered to that device will be displayed. This is useful for maintenance and debugging purposes. In some systems, this can eliminate the development of large display panels and their attendant cost. The contents of any register in the system can be displayed on the computer console.

A programmers console is optionally available which simultaneously displays major internal registers in the computer in addition to the selectable data display. Both consoles are pluggable and interchangeable.

Data may also be transmitted to any device in the system from the console switches by selecting its address and activating the Transmit Key. Control switches on the console are: Start, Continue, Read, Write, Display, Transmit, Single step and Stop.

Minimum Configuration — The system described so far is the minimum configuration in which the GRI 909 is available. With the addition of some kind of stored program, it can provide all the control capabilities of a general purpose computer. It can test data, transmit and receive control signals and perform arithmetic and logic operations on data, one bit at a time, as it passes through the Bus Modifier. Although limited in the execution times of its arithmetic operations, it can be used as a special purpose controller with read-only memory or a general purpose controller with random access memory for those applications requiring little or no arithmetic, or where the execution times of arithmetic are not a critical factor. This provides a very inexpensive approach to system control, with the capacity for expansion to full computer capability if required.

Core Memory — The memory of the GRI 909 is a 16-bit, random access, ferrite core memory in 1024 and 4096 word plug-in modules. It can be expanded to 8K in the basic processor frame without additional wiring. The total expansion capability is 32,768 directly addressed words.

There is a multiple channel, single cycle, direct memory access system in the basic processor which permits direct, single-word access to the memory by a system function. The complexity of multiplexing multiple devices on many different data channels is relegated to each device as it is added by a simple priority allocation system. Similarly, a range of complexity in implementation is available to the system designer. The data channels may be used to transfer data in or out of memory, or simply to increment a specified memory location. A replace operation is also permissible, i.e., read/modify/write, all within the one memory cycle.

*PATENTS APPLIED FOR

Read-only memory can be substituted for and intermixed with core memory.

Arithmetic Operator — The arithmetic and logic manipulations that can be performed in the functional Arithmetic Operator are "ADD," "AND," "OR," and "EXCLUSIVE OR." This arithmetic operator operates somewhat differently than that of a typical computer. Instructions are not issued that say "ADD", which in a conventional computer says, "one number is in the accumulator and the other number is in memory. Pull the number out of memory, add the two together, and put the sum back into the accumulator."

In the GRI 909, the Function Generator is used to generate the "ADD" function. What the instruction looks like is: "Function Output 'ADD' to the arithmetic operator." This element will always perform the ADD function between the current value of X and Y accumulators until the user issues another command changing the state. When either one of those registers is changed, a new sum appears, immediately available for transfer to any point in the system. New values can be presented from a system register with a new result obtained in a single cycle time, 1.76 usec. The result, contained in a separate register, always reflects the instantaneous output generated by the contents of the X and Y accumulators as controlled by the function selected. It can be stored in memory by a single instruction. The introduction of new values to one accumulator does not alter the contents of the other accumulator.

FIRMWARE OPERATORS

Since the bus structure of the GRI 909 is open ended, other functional operators can be added on a modular basis. Each time a functional operator is added to the system at least one instruction is added, plus all the variations on that instruction provided by the Bus Modifier. These functional operators can be selected from among the standard options offered or can be developed by the system engineer to incorporate special or proprietary instructions. The standard options include: Multiply, Square root, Byte pack, Byte swap, additional Arithmetic Operators, general purpose Registers, BCD/binary conversion, BCD comparators and others.

DEVICE OPERATORS

Device operators provide the interface circuitry between system devices and the computer. Direct memory access and priority interrupt circuitry are provided for each device, as required, in the device operator. The high speed signals related to the internal operation of the computer are terminated at the device operators so that external devices see only relatively slow signals that can be cabled to the device. In some cases, if the device is simple, the entire device can be provided on the device operator plug-in module. This module is approximately 3½ inches by 9 inches and provides an output cable termination.

Standard software routines provided with the GRI 909 include:

Programming Support Software:

FAST: an assembly program supporting a functionally-oriented language.

BASE: an assembly program supporting a basic assembly language.

BASE 360: a version of BASE written in PL/1 for the IBM 360 series computers.

Source Tape Editor: an editing program combining line oriented and content oriented commands.

Debugging Aids

Loaders

Math Routines:

Fixed-point single-precision arithmetic

Fixed-point double-precision arithmetic

Floating-point arithmetic and mathematical functions (an interpretive package)

Utility Routines:

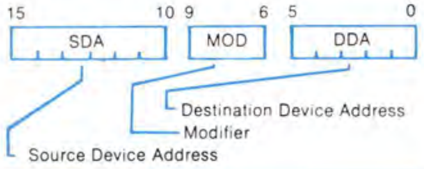
Input/output for standard devices

Data conversion

Diagnostics

PROGRAMMING

Machine Instructions:
GRI 909 instructions are of the general form DEVICE X TO DEVICE Y, and are described by a single machine format:

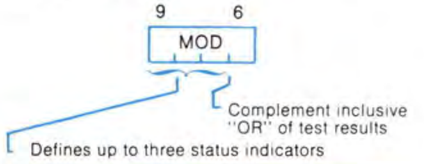


The actual operation performed by an instruction is dependent upon the unique combination of SDA, MOD and DDA.

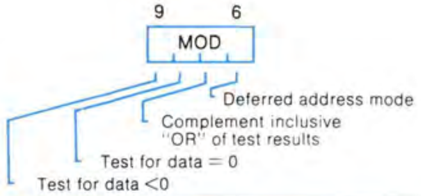
The four functional instruction classes are outlined below. Each class is described with examples of machine and corresponding functional language instructions shown in Table 1. In machine language SDA and DDA are represented as two octal digits each, and MOD is represented as four binary digits.

Function Generator: The source address of the Function Generator is 02. It causes control signals (rather than data) to be transmitted to any system device specified by the DDA. The Modifier defines up to four pulses in combination to be transmitted in parallel.

Function Test: The destination address of Function Test is 02. It senses the status indicators associated with any system device as specified by the source address. If the status test defined by the modifier is true, a SKIP instruction is performed. The format for the modifier is:


9
6
MOD
Complement inclusive "OR" of test results
Defines up to three status indicators

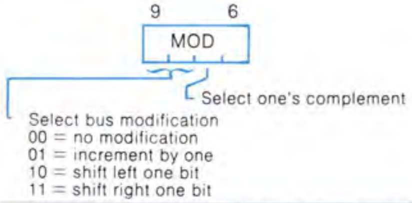
Data Test: The destination address of Data Test is 03. It can test data from any source address for less than zero, equal to zero or any combinations thereof. If the test defined by the modifier is true, the next word is taken as a jump address; otherwise, the next word is skipped. The format for the modifier is:



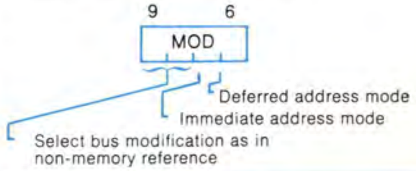
If a jump is performed, the current value of the Sequence Counter is automatically stored in a trap register associated with the Data Test. If the jump is to a subroutine, the trap register provides the link back to the calling program.

Data Transmission: Any instruction not in one of the classes defined above implies the transmission of data from a source device to a destination device as specified by their addresses. The source and destination may be the same device. Data transmission instructions are of two forms:

Non-Memory reference: The Modifier format is



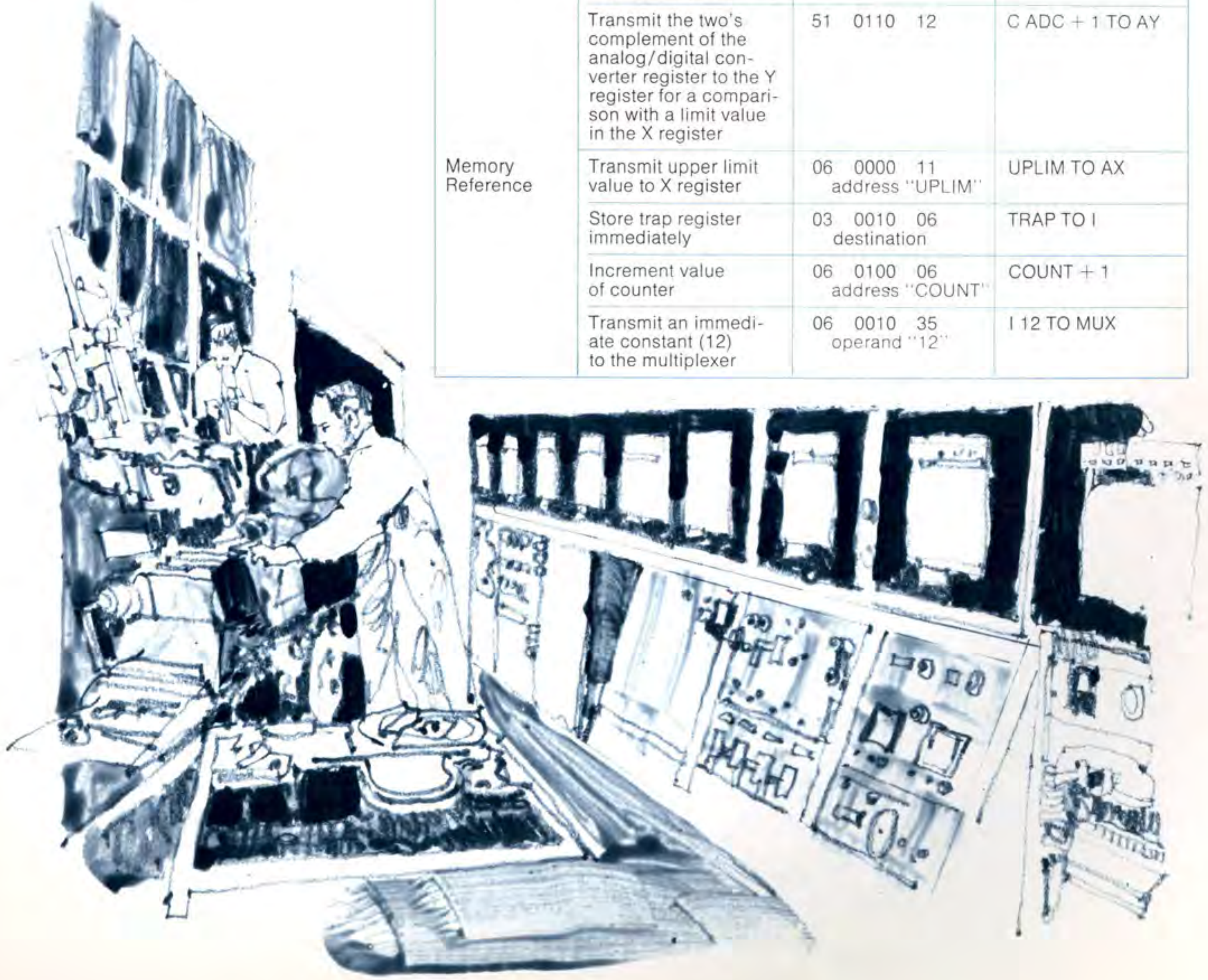
Memory reference: A memory reference instruction is designated if the source or destination address is the Memory Buffer Register (06). The next word is taken as a memory address or an operand. The format for the Modifier is:



INSTRUCTION EXAMPLES:

The GRI-909 user-oriented assembly language is easily comprehended by engineers and programmers. Each assembly language functional statement is translated by the assembler into a single machine instruction. Examples of both assembly statements and corresponding machine instruction are shown.

Class	Description	Machine Instruction	Assembler Instruction
Function Generator	Start teletype paper tape input	02 0001 77	START TO TTI
	Select Arithmetic Operator "OR" function	02 1100 13	OR TO AO
Function Test	Skip if teletype input not ready	77 0011 02	SKIP IF TTI NOT READY
	Skip if arithmetic operation caused overflow	13 0010 02	SKIP IF AO OVFLO
Data Test	If teletype input equal to zero, go to "done"	77 0100 03 jump address "DONE"	IF TTI ETZ GO TO DONE
	If arithmetic result greater than zero, go to "alarm"	13 1110 03 jump address "ALARM"	IF AO GTZ GO TO ALARM
Data Transmission Non-memory Reference	Increment the multiplexer address register	35 0100 35	MUX + 1
	Transmit teletype input character to the high speed punch	77 0000 76	TTI TO HSP
	Transmit the two's complement of the analog/digital converter register to the Y register for a comparison with a limit value in the X register	51 0110 12	C ADC + 1 TO AY
Memory Reference	Transmit upper limit value to X register	06 0000 11 address "UPLIM"	UPLIM TO AX
	Store trap register immediately	03 0010 06 destination	TRAP TO I
	Increment value of counter	06 0100 06 address "COUNT"	COUNT + 1
	Transmit an immediate constant (12) to the multiplexer	06 0010 35 operand "12"	I 12 TO MUX

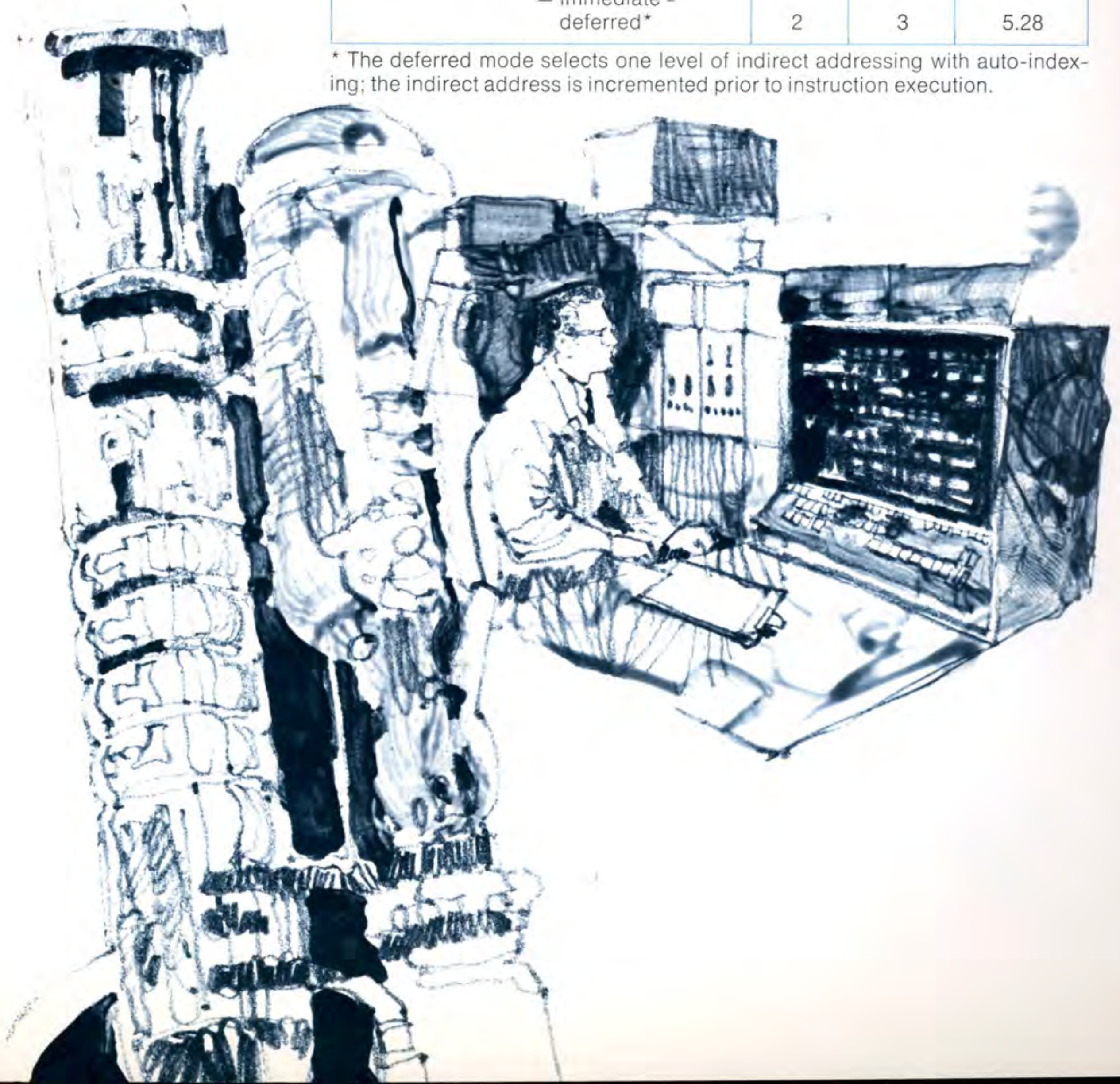


SOFTWARE

INSTRUCTION SUMMARY

Class	Length (words)	Memory Cycles	Time (usec.)
Function Generation	1	1	1.76
Function Testing – skip	1	1	1.76
– no skip	1	1	1.76
Data Testing – no jump	2	1	1.76
– jump direct	2	2	3.52
– jump deferred*	2	3	5.28
Data Transmission			
Non-Memory reference	1	1	1.76
Memory reference – direct	2	3	5.28
– immediate	2	2	3.52
– deferred*	2	4	7.04
– immediate - deferred*	2	3	5.28

* The deferred mode selects one level of indirect addressing with auto-indexing; the indirect address is incremented prior to instruction execution.



FIRMWARE OPERATORS

GRI 909* capabilities can be expanded by adding firmware operators to the system. A firmware operator presents at least one register to the bus system. Each functional operator added to the computer system adds one or more hardware instructions to the computer plus the variations provided by the bus modifier. The development of these options is a continuing corporate effort providing greater flexibility and computing power with software compatibility, on a modular basis. Examples of some of the firmware options currently available are described as follows:

Multiplier – A multiply can be executed either through subroutine instructions or by the Multiply Operator, a firmware option. The subroutine for multiply has a maximum execution time of 360 microseconds for a full 31-bit signed product and occupies 42 memory locations.

The Multiply Operator performs a 16-bit unsigned multiplication. It uses the Arithmetic Operator and issues external instruction requests to the processor. It uses a single address and contains a 16 bit register (MPR) which may also be used as a temporary storage register.

The 31-bit product is available in the X-accumulator of the Arithmetic Operator (most significant) and MPR (least significant) 56.32 microseconds after starting the operator.

The execution time might be extended if direct memory access requests arrive during the processing of this instruction. A higher speed multiply operator is also available with an execution time under 10 microseconds.

Byte Swap – The Byte Swap subroutine requires 20 memory locations and executes in 124 microseconds.

The Byte Swap Operator is a 16-bit register (SWAP) used to interchange the halves of a computer word. It executes the interchange in a single cycle time, 1.76 microseconds, exchanging bits 15-8 with bits 7-0. A Byte Pack Operator is also available to form a 16 bit word from two 8 bit characters that are loaded sequentially.

These two examples are indicative of the flexibility inherent in the GRI 909 computer. The ability is intrinsic to the Direct Function Processing technique permitting virtually any operation to be incorporated in the instruction repertoire of the machine.

*PATENTS APPLIED FOR

INPUT/OUTPUT OPERATIONS

The contents of any register connected to the computer bus structure can be transferred to any other register in the system in a single instruction. The registers can be any devices in the compuer or in the system, including memory. The GRI 909 recognizes and services interrupt requests and direct memory requests generated by system devices. Devices may, however, cause traps to unique memory locations. The cost of generating a unique address is borne by the device interface itself. In a multi-level interrupt environment, priority is enforced by manipulations of an interrupt status register by the respective interrupt routines. Each device that is interfaced to operate in an interrupt mode contributes a bit to the Interrupt Status Register so that it may be selectively enabled and disabled. Interrupts may cause a trap to memory location zero. If two or more devices trap to the same memory location, a standard SKIP sequence is used by the interrupt executive routine to enforce priorities and isolate the device.

Input/output devices may be driven in a programmed synchronized mode, normal interrupt mode, or direct memory access mode. The permissible mode(s) for a device are determined by its interface with the system.

A device to be operated in the direct memory mode presents at least two registers to the bus system (memory starting address and block length). A typical direct-access device interface will appear to the user as follows:

3 addresses – DSKA: starting address on disc (prime register); DSKM: starting address in memory; DSKL: block length to be transferred.

Example: read from disc

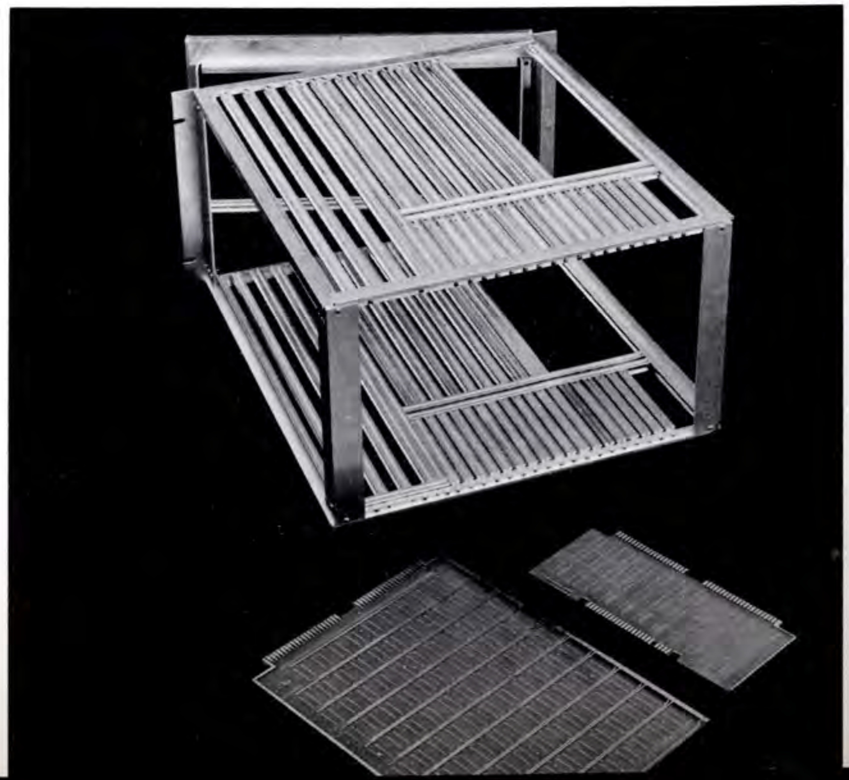
```
TRACK TO DSKA   : SET DISK ADDRESS
BUFFA TO DSKM   : SET CORE ADDRESS
LENG TO DSKL    : SET LENGTH
READ TO DSKA    : INITIATE READ
```

The device will steal one machine cycle (1.76 usec) for the transfer of each data word. When the block transfer is completed, the device generates an end-of-range interrupt on the priority interrupt channel. In addition, the disc contributes one bit to the machine status word to indicate its active state and sense flags to indicate parity error and up-to-speed.

Another interrupt mode, External Instruction Request, is provided. This interrupt mode is used to exercise the computer with hardware diagnostic modules. It can also be used in conjunction with a read-only memory. When read-only memory is used in this interrupt mode an instruction is executed in 880 nanoseconds.

SPECIFICATIONS

Physical	Mounts from front in standard 19-inch cabinet with provision for rack slides. **Size:** 10½ inches high, 20 inches deep. **Weight:** 50 pounds, approximately Space is provided in basic frame for up to three major functional firmware options and up to 16 firmware or interface modules. Memory capacity of the basic frame is 8,192 16-bit words. Extender frames can be provided for additional 24,576 words of core memory or additional firmware or interface modules. Read-only memory modules can be interchanged with core memory modules.
Electrical	**Power:** 100-130 vac, 60 Hz ± 3% or 200-240 vac, 50 Hz ± 3%, single phase, 4 amperes. **Power Dissipation:** 150-250 watts **Logic Levels:** ground and +4 vdc, DTL and TTL compatible
Functional	**Machine Cycle Time:** 1.76 microseconds, when executing instructions from main memory. 880 nanoseconds, when executing instructions in External Instruction Request mode. **Word Length:** 16 bits **Core Memory Size:** 1024 words, expandable to 32,768 words, directly addressable. **Instructions:** The number of machine instructions is modular and depends upon the firmware and device options used. The theoretical limit is in excess of 50,000 instructions. **Interrupts:** 16 levels, more than one device can be operated at each level. **Input/Output Rate:** Direct memory access or device to device at 568,000 16-bit words per second, maximum. **Registers:** 11 in basic computer including arithmetic operator and memory plus Link and Overflow status bits. Registers are also provided with direct memory access and priority interrupt devices. Additional general purpose registers and/or accumulators available as options.
Environmental	**Temperature:** 0° to 50° C, ambient **Relative Humidity:** to 90% (operating) **Cooling:** Convection, no fans required for operation over ambient temperature range. **Switches:** All console data and control switches are photo-optical without mechanical contacts subject to wear or arcing.



GRI Computer Corp.

76 ROWE STREET, NEWTON, MASSACHUSETTS 02166
(617) 969-7346