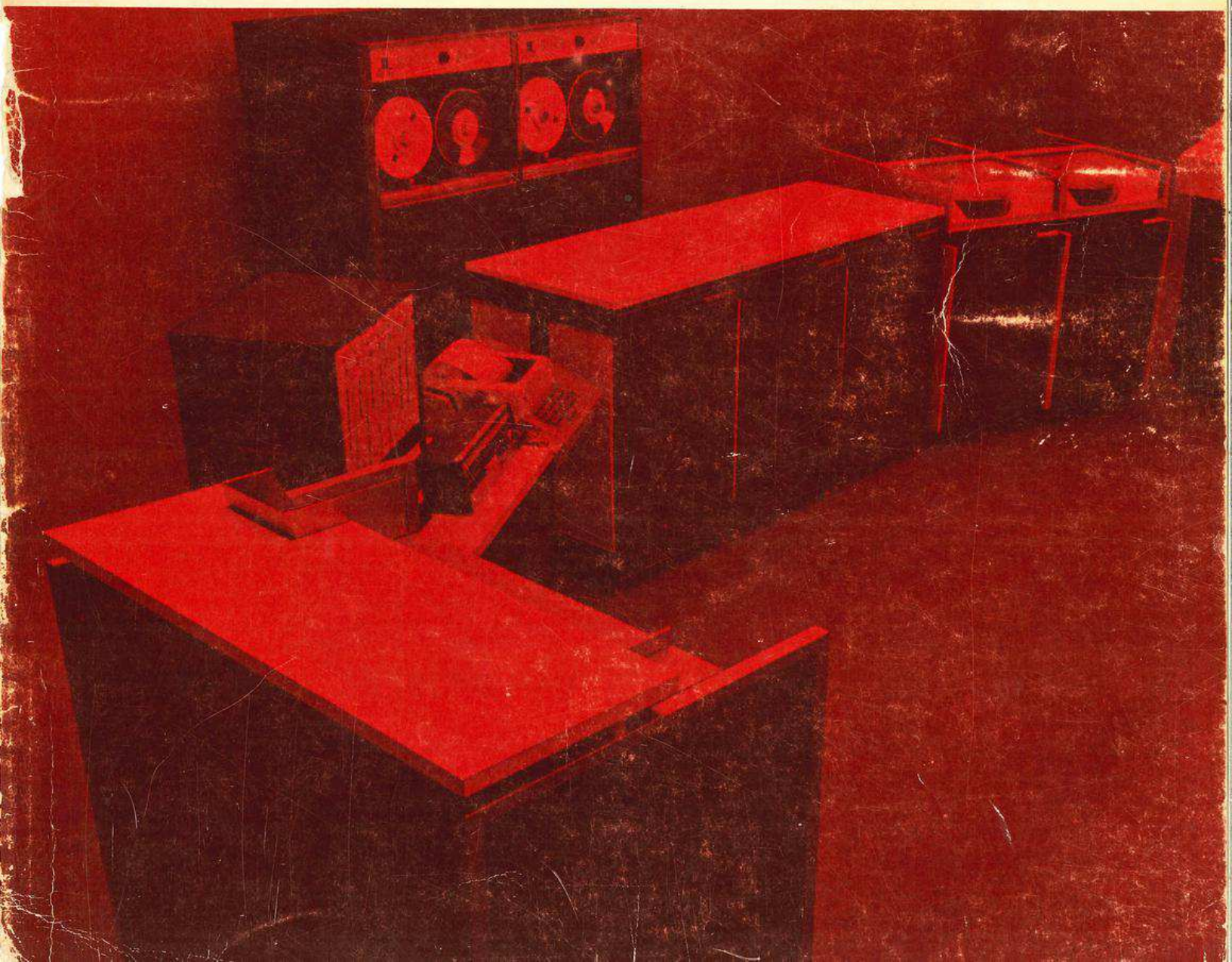
Honeywell

SERIES 16

MODELS 316 AND 516
SUMMARY DESCRIPTION



SERIES 16

MODELS 316 AND 516
SUMMARY DESCRIPTION

FOREWORD

The Series 16 Models 316 and 516 Summary Description is intended for those having a general familiarity with data processing. Machine characteristics and programming aids are described in terms that should aid comparisons between these systems and competitive equipment. The equipment characteristics reported herein remain subject to revision in order that design improvements may be incorporated.

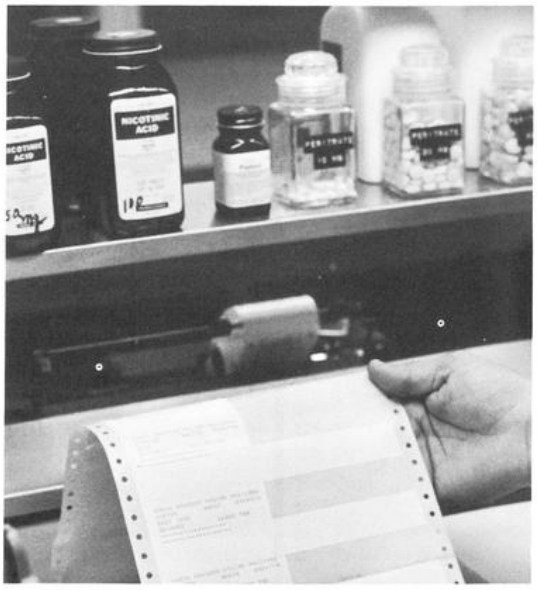
Section 1 discusses many of the problems faced by the minicomputer system designer and introduces the answers provided by the Series 16. Section 2 contains a detailed description of the Series 16, 1600, 1620, and 1640 packaged solutions. All of the individual components that comprise the Series 16 are discussed in Section 3. Finally, the appendix to this summary illustrates and describes the Series 16 instruction set, one of the primary reasons that the Series 16 is such an excellent answer to many control and communications problems.

File No. 1A01

TABLE of CONTENTS

LIST of ILLUSTRATIONS

LIST of TABLES


NICOTINIC
ACID

1

MINICOMPUTERS and INFORMATION PROCESSING

For the government, industrial, or educational analyst, consultant, and/or data processing manager, the 16-bit minicomputer is rapidly becoming a versatile, multipurpose answer to a host of information processing demands. Long recognized as an excellent tool in the process control environment (with tool the key word), its possibilities in a wider range of information processing applications (e.g., Data Communications, Computer System Timesharing, and even Batch Processing) have been recognized only recently.

Despite many claims, the "computer" is merely a tool, useful in information processing and control. Minicomputer users, from owners to operators, have long recognized that fact. Realizing that even in this era of instant answers, the rules of systems design (see insert below) are as valid as they were in the late days of patchboards and vacuum tubes, the minicomputer systems designer has consistently followed this formula:

KNOW A TOOL'S CAPABILITIES BEFORE ATTEMPTING TO USE IT.

For many reasons, quite the opposite has occurred in the data processing-oriented segment of computer system manufacturing. Analysts and manufacturers have developed many stock, and therefore reasonably economical, answers to varied business processing problems; general accounting, and inventory control, among others. For the most part, such an approach to control and communications problems was impossible. But, once the control and/or communication problems are outlined and certain general prerequisites concerning the capabilities of minicomputers are met, the solutions are available, economical, and often better, qualitatively, than those utilizing larger computers with a more general purpose.

Basically, the tools available (processors, interfaces, input/output subsystems, and the software that controls all three) merely manipulate data. Add it. Subtract it. Store it. Transfer it. To what advantage? Advantages include:

- The manipulation is fast.
- In comparison with human efforts, the manipulation is error free.
- The concept of the manipulation is easy to understand.

In fact, it is the simplicity of the 16-bit minicomputer design – its easy application to any problem – that is the minicomputer's strongest advantage. For this reason, especially in hard-to-pin-down process and communication applications, many consider the minicomputer a likely tool in implementing a solution. But remember, the tool is just one of the considerations, other factors in systems design are:

- Cost, and its cousin, time.
- Reliability among other specification requirements.
- Support provisions.
- The applications or procedures needing computerization.

Read and discuss these problem areas. Note and expand on the areas that most affect your situation. Then, discover the features and specifications that make the Honeywell Series 16 an excellent answer to many demands of the control and/or communications environment. Honeywell invites you to investigate this claim; your organization will demand it.

1. Understand the capabilities of available tools.
2. Understand those procedures requiring control or processing.
3. State the objectives of the control or processing project to be undertaken.
4. Consider changes to the entire procedure as well as the application of a computer system.
5. Design the solution.

MONEY MATTERS

Whether you are in charge of budgeting a computer implementation effort or merely hamstrung by its budgetary constraints, you know that money matters. Every solution or application is a sum of parts, components, and despite claims to the contrary, each and every component has a cost. If you have to finish a project with a budget in mind, you had better peg that cost.

In budgeting total expenses, there's a tendency to overemphasize hardware prices. While there's no doubting mainframe and peripheral charges, the key parameters in implementing a minicomputer application as budgeted, involve:

- Software – design and development
- Interfaces of all sorts.
- Support – no matter who supplies it.
- People.

As the bar graph illustrates:

1. Mainframe costs represent about one-half of total hardware costs.
2. Mainframe costs represent about one-half of software costs.
3. Mainframe costs represent only about one-fourth of total first year investment.
4. Mainframe costs represent only one-eighth of the total cost of a five-year project.

And, unlike hardware costs, which are relatively fixed by competition in the market and the cost of production, the prices charged for software, support, and personnel, can vary widely.

Time Is Money

A final monetary consideration: time is money, too. Carefully weigh project costs and benefits as they relate to a time line. For example, once the benefits of early installation and implementation are converted to the balance sheet, a solution that seems expensive on the surface is often less expensive than one requiring lengthy installation and checkout.

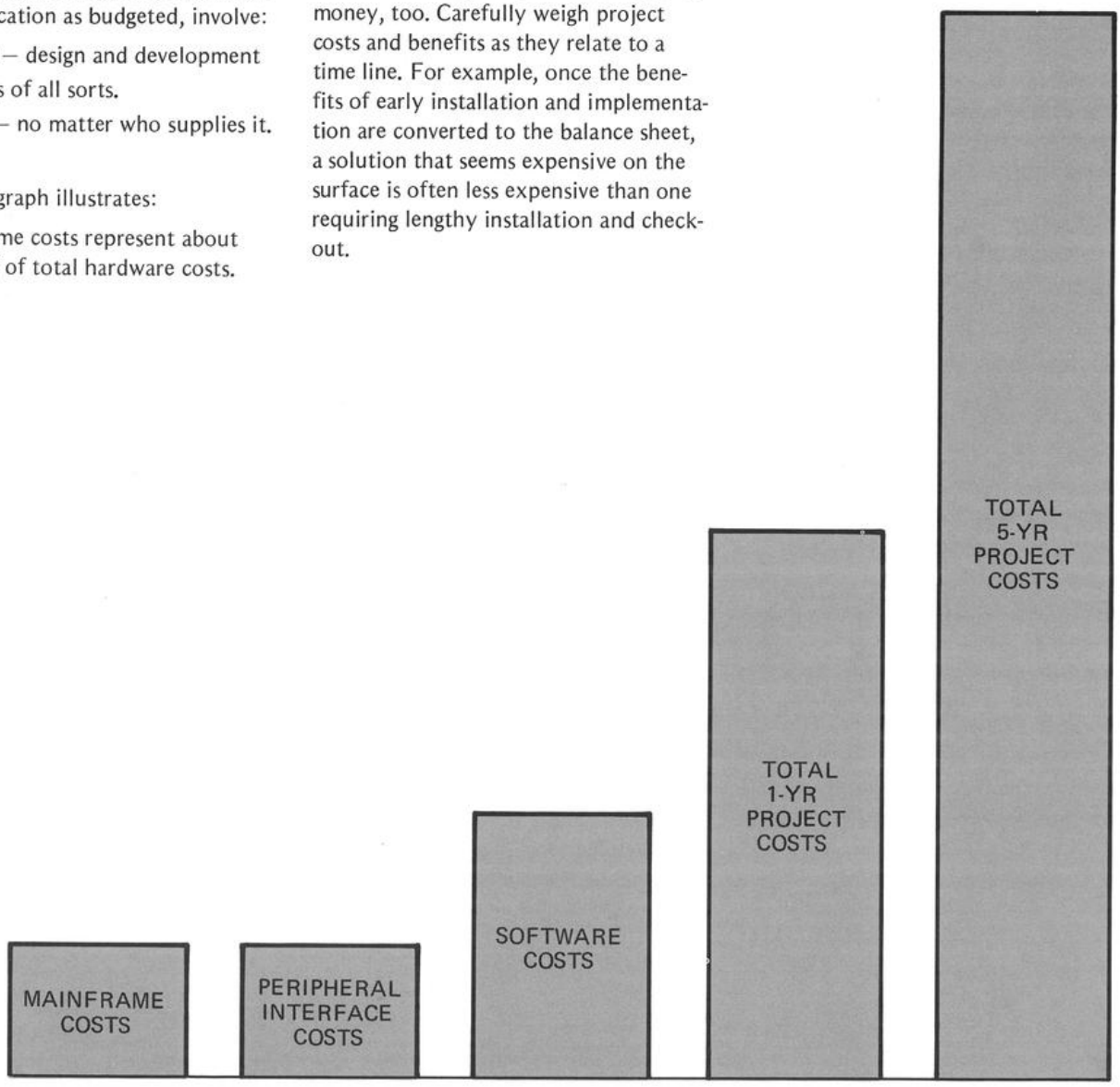


Figure 1. Ratio of Project Cost Factors

RELIABILITY; SPECIFICATIONS, IN GENERAL; THE REAL WORLD

Because of either space age technology or the trend toward consumerism, or both, product reliability is a growing demand. No where is this more true than in the computer industry. But, in a cliche-ridden industry, "reliability" has been ridden into the ground. The fact that all manufacturers advertise their equipment's reliability is no reason to assume that one product is as reliable as the next. Such an assumption could prove disastrous in real-time applications like control or communications.

Engineers, specializing in this field, define product reliability as:

> ". . . the probability that a given piece of equipment will perform a specified function, without failure, within a given period of operation, under given environmental conditions."

This is meaty material; not the stuff about which catchy jingles are generated. When, in careful systems design, objectives are stated, note the necessity and degree of required reliability. If it amazes you that there are degrees of reliability, be prepared to watch months of planning disintegrate during a brownout. In fact, to anticipate modes of failure is to increase the probability of a system's success.

Systems are a sum of many basically simple parts. They interact in a complex set of relationships in order to perform certain tasks. This is as true inside the processor and other hardware components as it is for the procedures themselves. If you're responsible for specifying these interrelationships, consider the reliability of such parts as the communications lines, the valves to be controlled, the people who'll enter control data, etc. Then, consider the reliability factor of the tools (components) to be used.

Many other specifications are involved in systems design. The analyst lists and relists the electrical, physical, and environmental conditions under which he expects his equipment to function. Again, as with reliability, there are degrees of ruggedization; remember that the ruggeder the environment specified, the higher the cost. This relationship between performance and cost can't be stressed too much.

For example, computer systems used in many large, batch-oriented applications are so sensitive that they require temperature and humidity measurements of minimal fluctuation. Other systems are tough enough to withstand great changes in temperature, humidity, altitude, barometric pressure, and even sustained shock. If you need special performance capabilities in any of these specification areas, the tools are available.

Finally, other specifications are specific to the application:

In control applications, parameters like voltage, hydraulic flow, and decibels may require measurement. Unless care is taken to match processor and interface capabilities to the capabilities of the measuring devices, even the most extensive data acquisition and control system is in for trouble.

In a minicomputer communications application, many factors affect final specifications: the terminals feeding the processor, the lines they can or must utilize, the choice of controller type, the input/output mode, and most importantly, the nature of the application. Within each one of these areas, a wide degree of performance specifications are available.

If this sounds confusing – to some degree – the confusion is intentional. By considering randomly so many of the performance/problem areas that exist in designing solutions, you can't help but be aware of the great complexity of it all. But the fact is, while there are real-world restraints like a budget and performance capabilities, there are also means of overcoming these problems. They include:

- Available support,
- Your knowledge of the applications, and
- The tools themselves.

SYSTEMS SUPPORT

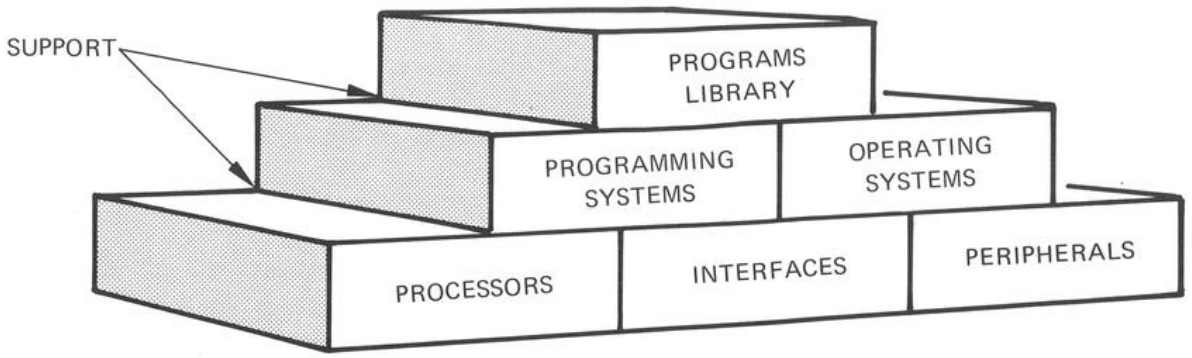


Figure 2. A System's Components.

Support can become a two-edged sword. Just as you can't confidently pretend to design, implement, and operate a minicomputer application without assistance, you must exercise full planning and implementation control over the implementation of a "turnkey" solution to one of your organization's processing problems. Probably, the greatest asset you possess is your knowledge of the procedure requiring computerization. This knowledge when combined with the kind of support you can and should seek from qualified consultants, computer manufacturers, and your own management, leads to successful application. The types of support available:

- Design support.
- Implementation support.
- Post-installation support.

All three are required in order to fully realize the potential of a project. All three are available in varying degrees and in, your systems design, the amount you expect should be specified.

Design Support

In the planning stages of a project, time is the one supporting ingredient which your management can supply. Because careful planning and the setting of objectives are so important when an organization begins to tread on new ground (either the use of a computer system for the first time or the implementation of a new application on a currently installed computer system) every objective of the system must be carefully specified. This is accomplished by:

1. Defining requirements; specifying the people needing the processed output or the device needing control.
2. Defining input methods; volumes, forms, formats – digital, analog.
3. Defining communication methods (if any); traffic, redundancy, terminals, tests.
4. Defining data storage specifications; devices, file organizations, accessing methods.
5. Defining processing methods; programming systems, operator actions, available routines.
6. Defining a control of the control system itself, and finally...
7. Defining the plan under which implementation and post-installation will proceed.

As Figure 2. illustrates, support is the non-component factor that holds the systems components (processor, input/output subsystems, interfaces, and software) together. Like an adhesive, it must be applied over the entire surface of the solution but not so heavy as to prevent their interaction. During planning the best source of support from consultants is the purchase of a study of your processing needs. When done properly, this is a worthwhile investment in your project's success.

Computer manufacturers offer extensive planning support, including many general and specific publications, and some excellent fee-charged and non-fee courses and seminars. When all three elements are combined, the designer is assured of having the proper knowledge of the problems he faces and the tools available.

Implementation Support

Implementation (or pre-installation) support, including software modules, systems analysis time, specific and often no-charge education, helps the organization over the rough period while new equipment is being installed and old systems are phasing out.

Post-Installation Support

Once installed, every system requires on-going user and manufacturer support to ensure the fulfillment of the performance and reliability objectives stated during the planning phase. Unless you're one of the "iron" organizations of the industrial community or the governmental superstructure, this type of on-going support will be a major concern. The best solution: a regular maintenance program of troubleshooting and systems test plus immediate response to down-time situations.

THE APPLICATIONS THEMSELVES

As tools, 16-bit minicomputers have special usefulness in the control and communications environments. Because of their size, cost, specification flexibility, and advantages (fast, accurate, design simplicity), minicomputer systems are limited only by the number of related problems your organization faces.

Generally, minicomputer applications are real-time, rather than batch, oriented. More specific indicators of minicomputer usefulness include:

- Problems that, though exceedingly complex in their entirety, can be reduced to small manageable components.
- The actual or possible overloading of a large-scale central computer.
- A requirement for uncomplicated, high-speed peripheral or interface throughput.
- Fast response to changing environmental conditions.
- A limited calculation application.

Remember, before going any further, that a minicomputer system is not an answer for all your information processing problems; not even for all your small information processing problems. Commercially-oriented systems designed to process large batches of related data at one time have superior characteristics (peripheral overlap, large main memory) in problem areas where the need for them exists and where the resources required for successful implementation are available. But, minicomputer systems with their special size and throughput characteristics meet the demands in the problem area (process control) for which many of them were designed. They are paradoxically, general-purpose systems that can be dedicated to one need while retaining a general nature.

Data Communications

The choice of the correct communications equipment in a particular application hinges on both cost and specifications. As in all other minicomputer configurations, the components include processor, input/output subsystems, interfaces, and software; in particular, the interfaces will be communications line controllers, and the system itself will be a remotely-located peripheral of a central computer.

In many instances, 16-bit minicomputers have proven extremely effective in accomplishing communications tasks for which only large computer-systems were suggested in the past. Message switching is one example. In the illustration on page 7 the two computers are handling the switching of messages between as many as 128 low-speed communications lines. Because their cost is so low, and the amount of actual calculation is small, the 16-bit computers used in the example offer the best solution to the budgetary problem faced in the example shown. Performance characteristics that make the minicomputer a useful tool:

- Extensive character manipulation.
- Character string checking.
- Real-time housekeeping.

All three are ideally handled by a small processor with a fast cycle time and high throughput capabilities. The use of two central processors also provides a redundancy factor that increases the systems reliability.

The peripherals used in the message switching application are also important factors in this solution's success. A proper balance between the high-access-time fixed head disks and the high-capacity moving head disks provides both immediate intransit and long term journaling and intercept storage.

The processor/peripheral array interfaces both the more versatile direct connect and the more economical switched network lines; such a capability is an important factor in holding down the costs of communicat-

ing, a cost element that is only partially affected by the central processor and peripherals chosen. These costs are fixed, according to the type of service, by the government and the common carriers who supply them.

In operation, the 128 separate lines feed partially formed characters into a multi-line controller that also offers the proper balance of performance and cost; that is, while meeting line speed specifications, they are more economical than separate data line controllers. The multi-line controller formats the characters, checks for parity, and performs other control functions before passing the characters on to the processors.

Input to both processors is simultaneous and in parallel; because of this capability, both processors build identical information on the intransit fixed head disk devices.

Interprocessor communication utilizes the ICCU (Intercomputer Communications Unit); this device enables redundant message processing. Another hardware factor, important to the reliability of the application, is the watchdog timer (WDT), an independent hardware device that monitors the performance of both processors and provides an alarm and switchover if one of the processors should fail.

Sophisticated software utilizes and controls the hardware configuration. It is this software that handles the constant tasks of blocking, error detection, code conversion, editing and routing which are the purposes of a message switcher.

Other communications applications for which the 16-bit minicomputer is an economical, reliable solution include:

- **Remote Concentration** – A system, located at a distance from the central computer, accepts data concurrently from multiple low speed lines and concentrates this data for

TABLE 1. COMMUNICATION SYSTEMS CONFIGURATION CONSIDERATIONS

Consideration	Requirements
Single-, Dual-, or Multi-Processor?	Dual processors are important where any of these factors are pre-eminent: – load sharing – reliability – maintainability Where these requirements are not pre-eminent, a single-processor control system can be specified with resulting savings.
Controllers?	Controllers available to the systems designer: – single line – high- and low-capacity multi-line – programmed multi-line Usually chosen before any other hardware component, the choice of controllers is based on the desired mix of line speeds, the need for synchronous or asynchronous transmission, the use of private line or the switched network, the types of timing and direction (simplex, full or half duplex), and finally, future plans.
Types of Communications Lines?	Usually your application and the terminals that are a part of that application will determine lines required. In general, there are three categories: – under 300 bps - low speed – between 300 and 9600 bps - medium – 9600 bps and above - high speed The important element for the designer is the configuration of an economical mix of lines within the requirements of your application.
Processor Interface?	The mode of processor interface (single word, block transfers of data, direct memory connection) is an important design consideration. These modes are normally extra cost items on a processor; first of all be sure they are available; secondly, be sure you don't have to pay extra for a capability your application doesn't need.
Line Interfaces?	The choice of line interfaces depends on the choice of terminals. Single line controllers require no interface at all; interface is integral within the controller. Multi-line controllers have a specified interface keyed to their specifications.

transmission over a single high speed line to the central site. This system is most useful where geographical distance demands a single low cost line in place of many high cost lines. The remote concentrator itself is located in the approximate center of the many low speed lines.

- **Remote Batch Terminal** – The simplest communications application, the remote batch terminal provides remote input and output of batches of data over a single communications line; depending on available software, the use of a minicomputer, rather than a hard-wired terminal, gives the user the option of off-hour data processing.
- **Multiplexer** – In many cases, the minicomputer may prove the best method of performing the same function performed by the multiline controller in the message switching illustration. Because of its software capabilities, it is able to perform intelligent character assembly/disassembly, transfer, control recognition, and other tasks associated with communications interface.
- **Message Controller** – In this application, the minicomputer or minicomputers would perform one or all of the above applications in association with a central computer. Functions include:
  - Terminal polling and addressing.
  - Line buffering.
  - Automatic logging and routing.
  - Line and network status reporting.
  - Message security and header control.

The concept of relieving the central computer of all communications functions is the goal.

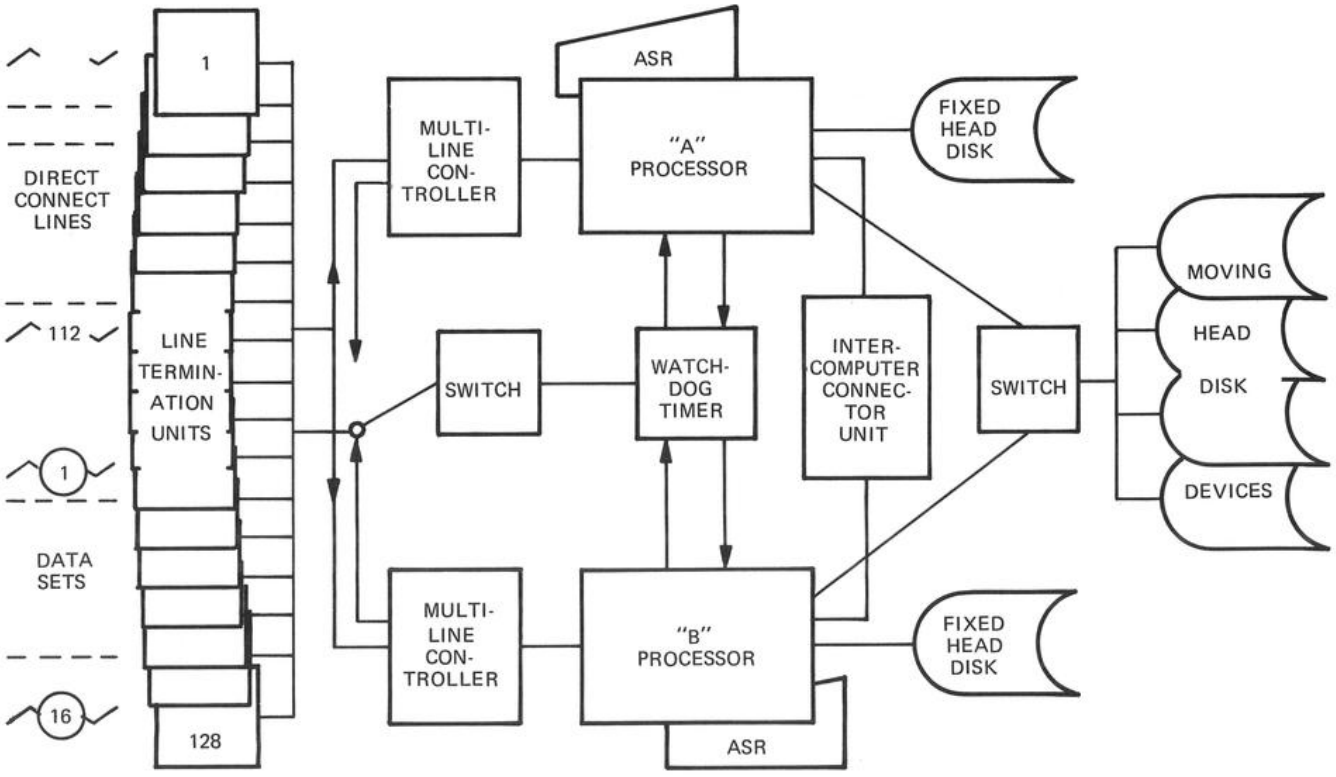


Figure 3. A Typical Message Switching System.

Data Acquisition and Control

Interface capabilities are the keys to data acquisition and controls system design. The systems themselves may vary from the simple recording of many data measurement devices to the sophisticated use of the minicomputer in order to acquire data, test its status against some norm, and branch to control instruction that would direct further actions by the procedure under computer control. The possibilities are only limited by the number of procedures that exist in industrial, and governmental applications.

TABLE 2. DATA ACQUISITION AND CONTROL SYSTEM CONFIGURATION CONSIDERATIONS

Consideration	Requirement
Processor?	In a control application, the processor should possess extended accumulator capabilities and high-speed addressing capabilities, especially indexing.
Peripherals?	The peripherals associated with a data acquisition and control system should provide economy and a good balance of bulk storage, input, and output devices in addition to the Process I/O Subsystem described below.
Bulk Storage?	With a choice between fixed and moving head disks or magnetic tape, the considerations are these: – Fixed head disks provide high speed access and relatively low capacity capabilities. – Moving head disks, on the other hand, provide a high storage capacity with a slower access time. – Useful where storage follows some fixed sequencing order, magnetic tape devices are even less expensive than disk devices.
Input?	Input peripherals include paper tape and punched card devices. Paper tape devices are usually less expensive, more difficult to change once a program tape has been prepared. The more expensive punched card devices permit the simplest program patching.
Output?	For printed output, the quantity determines an application's need for a high-speed line printer. In every case, a teletypewriter should be specified because of its usefulness as a control device.
Process Input/Output Subsystem?	The process input/output subsystem: 1. Selects and conditions input and output signals from the interfaces. 2. Buffers and times the signals. 3. Protects the central processor from excessive voltage intake from the various control devices.
Analog Inputs?	The choice of high and low level analog inputs rests on the requirements of the specific system being designed. Input modules are available to take practically any mix of thermocouple, control signal levels and outputs from process sensors such as flowmeters, pressure tranducers and others, with full scale ranges from plus or minus a few millivolts, up to 115 Vac, or current ranges.

TABLE 2. DATA ACQUISITION AND CONTROL SYSTEM CONFIGURATION CONSIDERATIONS (Cont.)

Consideration	Requirement
Digital Inputs?	The digital inputs to a process control system can be used to inform the processor of: – status (on or off, etc.) – asynchronous (a condition of many ons and offs) – count (the number of ons and offs)
Digital Output?	The processor can direct a flip/flop (set/reset) signal or a single-shot signal to the process being controlled.
Digital/Analog Converter?	Often as a cost consideration, the output of a digital/analog converter will be multiplexed so that no more than one converter will be required.

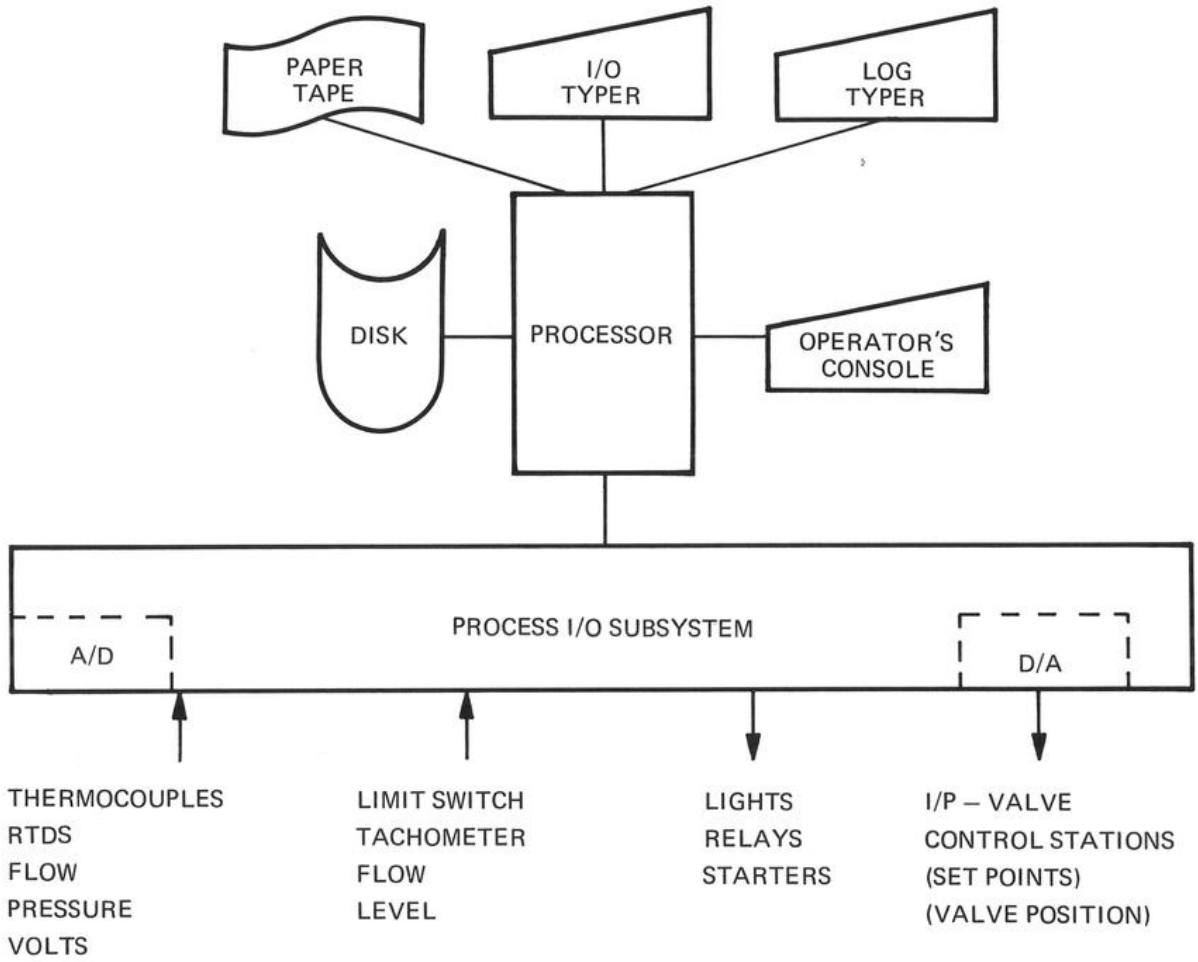


Figure 4. Typical Process Control Computer System

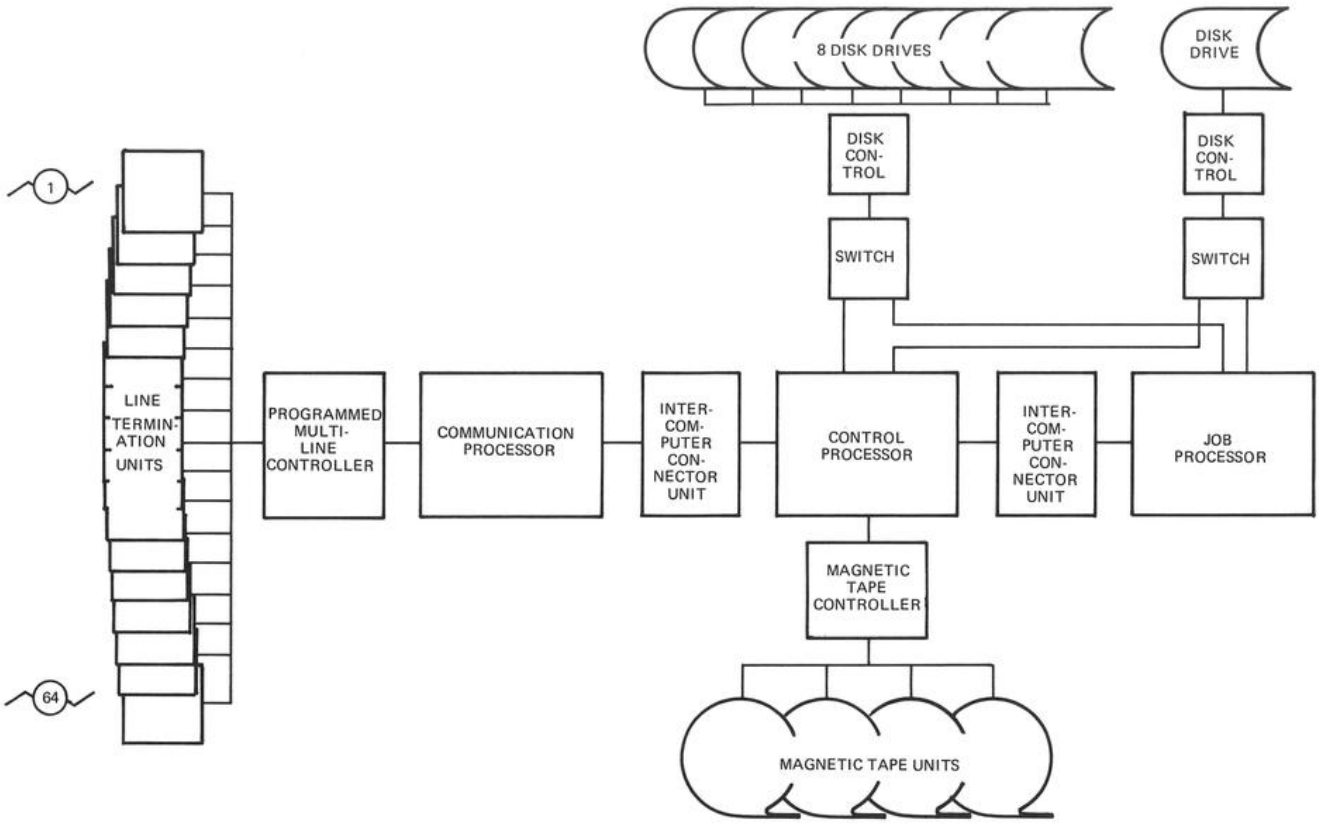


Figure 5. Typical Time Sharing System

Time sharing, both a communications and a control application because it involves the control of computer time as shared by a number of remotely located users, is a rapidly growing area of minicomputer application. In its earliest usage, time sharing involved only large-scale computer systems.

Now, this is changing. Some of the advantages these systems offer include:

- Computer access is virtually simultaneous with demand and therefore, for all intents and purposes, the computer is your own.
- Remote access is usually, but not always, performed on an easy-to-use keyboard; this replaces bulky and unfamiliar punched cards.
- Problems are solved in an interactive mode as opposed to the batch mode used by most in-house shops.
- The concept of time-sharing is ideally suited to the rapidly growing field of data base management.

Time sharing, in fact, is an already accepted solution to the problem many business-oriented organizations face when their employees interface with the computer. In many cases, the users pay for this "man/machine rapport" in ever-increasing monthly fees; sometimes these fees are higher than the cost of a wholly-owned batch system. Where this important economic factor is true and the user still desires the convenience and capabilities of time sharing, a solution even more economical than the batch system, 16-bit minicomputers, is available. With a well-designed minicomputer-based time sharing system, many manufacturing, governmental, and educational users can retain the advantages of the time sharing service while gaining:

- A system specialized to specific company needs instead of a system that satisfies the needs of an entire network.
- A wide choice of processing capabilities.
- Proprietary information never leaves the company.
- Interaction with the organization's batch system, if there is one, is easier if a time sharing system is also in-house.
- There is only one price; fees do not vary depending upon the number of time sharing services used by the organization.

Traditionally, time sharing is considered a system suited to educational and research facilities. In fact, time sharing is an economical answer to problems in any organization that needs the advantages mentioned above. Time sharing is currently utilized in a wide number of industries and applications areas, including manufacturing, printing and publishing, engineering, management science, financial management, personnel, and administration.

Many time sharing systems utilize a single minicomputer; more advanced systems use up to three separate central processors and still provide a price/performance break when compared with batch computer systems programmed, rather than designed, to provide time sharing. Again, as with control and communications applications, the determining factors, when specifying the caliber of time sharing system needed in your installation, are the specifications desired, the degree of reliability required, the number of users served, the budget restrictions, and support provisions.

Figure 5 illustrates a larger mini-computer system that services up to 64 simultaneous (others non-simultaneous) terminal users with programming capabilities limited only by the type of software the user chooses or chooses to develop.

Terminal input is handled immediately by a 16-bit processor with 4K main memory. The machine transfers characters to and from the terminals, provides echoback for transmission verification, provides some buffering, and enters characters one at a time into the control processor. The control processor and the job processor both share moving head disk files for data interchange; these two processes exchange/control information through an ICCU. The control processor, essentially the executive of the system, provides the normal interaction between the user, his programs, and data files that are held on the disks. The user may build up programs and data files upon the disks and when he requests that these programs be run, the control processor will queue his request for execution on the job processor. When the job processor is ready to execute the program, it will read the necessary files from the disks and input any required system programs from the system disk. Then, it will clock time, for a predetermined period; and if the job has not been completed, output the unfinished job to the system disk.

In this minicomputer application, the tasks have been divided evenly between the three processors. There is considerable difference in the computing power of the machines, each machine being matched reasonably well to the task required of it. The communications processor is a short cycle time, small memory, 16-bit processor, while the control and job processors are both large core capacity processors. The use of the multiple processor configuration has considerably increased the power of this system over an equally priced processor designed originally for batch data processing. Its design has simplified the task of implementing the software and of adding modifications, important factors in a growing installation.

As a simple example of the independent usage of the processors, the control and job processors can be isolated from the normal time sharing function and used for software development by the systems programmers, while the front end communications processors remains on-line to terminal users. Clearly the terminal users cannot do their normal computation, but when they attempt to sign on, the communications computer can reply with a standard message informing them of the state of the system.

THE TOOLS, THEMSELVES

The ability of the Series 16 to solve difficult control or communications problems is well documented. Aboard ship, in flight, in line, in real-time – the buzz words of tomorrow's applications – are design factors in Series 16 application today. This is no accident, the Series 16 is suited by heredity and environment to the problems of the future.

The Honeywell Series 16 includes systems components, tools, of each type:

- Central Processors
- Input/Mass Storage/Control/Output Subsystems
- On-Line Interface
- Software Systems

In addition, Honeywell supplies support that the rest of the minicomputer manufacturers can only envy. Its elements include:

- Education facilities from the only minicomputer manufacturer who is in the computer education field.
- Complete documentation, hardware and software, to support system design, installation, operation, and maintenance.
- A wide-area systems analysis and field service organization.

Most importantly, the Series 16 allows different modes of solution, including, on one hand, complete applications systems that save both time and money and, on the other, individual DATA-PAC products from which an entire system could be custom built, piece-by-piece. More specifically:

- "building blocks" – systems components, completely checked out, from which a system designer can, with confidence, assemble a solution best suited to his particular problem.

or

- "total solutions" – complete applications systems, answers to well defined and often-confronted problems for which the Series 16 is an excellent solution.

Like the selling features of the Series 16's many products, the choice of a kind of solution, associated with the Series 16, is an example of the Series 16's greatest asset: THE SERIES 16 IS A PRODUCT OF THE REAL WORLD.

What Are the Tools?

PROCESSORS

Honeywell's latest Series 16 products include two general-purpose, integrated circuit, digital central processors, the Models 316 and 516. Each features design and specification characteristics, as well as a price tag, that makes them suitable to similar, yet specifically different, minicomputer applications. The low-priced Model 316 offers the systems designer the economy features that make many applications viable; as well as offering full Honeywell support. The Model 516, with a faster cycle time and increased peripheral throughput capabilities, also offers full support capabilities and a price as attractive as its advanced performance specifications.

TABLE 3. HEREDITY/ENVIRONMENT . . . SUITED TO TODAY'S PROBLEMS

Heredity	Environment
■ Descendants of the 116, the industry's first 16-bit computer. ■ Born of the best components produced by people who began producing components. ■ Featuring technical refinement and reliability standards that are second to-none	■ "Total" Honeywell support embraces all Series 16 products. ■ Used extensively by Honeywell, the world's leading automation company, in internal applications.

INPUT/STORAGE/CONTROL/ OUTPUT

The second element, the Series 16 peripheral complement, is one of the most complete in the minicomputer field. No matter what an application demands in terms of data input control, mass storage, or output, the Series 16 features a peripheral that suits the requirements. Input devices available include: card readers, paper tape readers, and console keyboards. The console, as well as the teletypewriter and visual information projection (VIP) options featured in the Series 16, are also useful control devices. Disk devices, with both fixed and moving heads, are available for direct access mass storage and the more economical magnetic tape subsystems enable mass storage where a sequential processing mode is established. Output devices include a paper tape punch, card reader/punch, and a line printer. All Series 16 peripherals are supported by the Series 16 software systems and each features peripheral overlap and error checking controls that reflect the design status of the Series 16 processors.

INTERFACES

On-line interfaces, the modules that enable the Series 16 to be effective in all types of specific control and communications areas, including hybrid or one-time applications, also make the Series 16 a leading choice for the systems designer. Interfaces available:

- **Data Acqusition and Control Subsystems** – two complete families of compatible products permit interface with various voltage levels and signal types.
- **Communications Controllers** – three general speeds and from one to 128 lines can be controlled by an assortment of single-line, multi-line, and programmed multi-line controllers; an assortment of other features permitted.
- **DATA-PAC Family** – using the DATA-PAC family, the designer can build his own interface (as well as his own complete computer system, if he desires) from a completely compatible choice of components.

SOFTWARE

Sophisticated software, including both effective operating systems and powerful program languages, is an important Series 16 component. The operating systems feature the kind of effective control that reduces operator intervention and increases the system's efficiency. The available compilers give the Series 16 user the programming flexibility that ensures maximum utilization of the hardware. As with the other Series 16 components, software solutions are available in both general problem areas – control and communications. For programming the Series 16:

- An efficient, easy-to-use assembly language, DAP-16
- The industry standard, Fortran IV, useful in scientific applications.
- BASIC, TEACH, and SOLVE, three language systems suited to time sharing's simplicity requirements.

All software includes techniques necessary for sectorizing and loading in a small core memory processor. Unless noted, all hardware configurations are supported by the software; a Series 16 user is never forced to buy devices he otherwise doesn't need.

TABLE 4. SERIES 16 COURSE AVAILABILITY

Length	Title	Objectives
Logic/Maintenance Courses		
1 week	Basic Computer Theory	Provide "building blocks" understanding.
5 weeks	Series 16 (Models 316/516) Logic/Maintenance	Provide student with working knowledge of Series 16 hardware, diagnostic programming, and computer operations.
Programming Courses		
1 week	Introduction to Programming	Introduce programming concepts.
	Series 16 Programming	Provide working knowledge of:
		a. DAP-16 Assembler
2 weeks	(Standard)	b. Loaders
1 week	(Accelerated)	c. Utility routines
		d. Fortran IV Compiler
		e. Checkout routines
		f. Libraries
2 weeks	OLERT	Teach Fortran programmers how to use OLERT.
1 week	OP-16	Teach Series 16 user how to utilize OP-16
1 week	Series 16 Fortran	Provide sophisticated training in Fortran IV
Peripheral and Interface Courses		
2 weeks	Series 16 Real-Time Interface	Working knowledge (operation, programming, and maintenance) of RTI
8 days	Honeywell/CDC Magnetic Tape (7- and 9-track) and (VLC)	Knowledge of device and control logic and electromechanics, and programming techniques.
6 days	Honeywell Card Reader	
10 days	Honeywell Line Printer	
8 days	Honeywell Card Reader/ Card Punch Type 273 Disk Moving Head Disk File	
Other Courses		
1 week	Series 16 Interface	Present in-depth discussion of the Series 16 architecture
1 week	Series 16 Systems	In-depth discussion of the Series 16 capabilities; suited for management and design personnel
1 week	Series 16 Communications	An overview of communication techniques.

Support

The cohesive factor in systems structure, a factor that tends to hold a system together, support is one of Series 16's strongest facets. Featuring Education, Data Centers, Systems Support, Field Maintenance, Publications, Honeywell offers wide ranging and on-going support for all a system designers objectives. Specifically:

EDUCATION

To assist customers during both the planning and implementation phases of systems design, Honeywell offers a full array of programming and maintenance courses. The facing table lists the courses offered and notes their prerequisites; as always, other pertinent information is available from your Honeywell representative. Two training centers, one in Framingham, Mass., and the other in Sherman Oaks (Los Angeles), California, offer the same high quality educational atmosphere and teaching philosophy. The needs of some specific customers are met by the DSO Education Center in Arlington, Virgina. If you wish to enroll in any of the courses listed here or would merely like more information concerning them, indicate your interest on the Systems Designer Reply Card in the rear of this description.

PUBLICATIONS

A complete and up-to-date documentation package supports the installation and design of every Series 16 product or system. Both engineer-oriented and programmer-oriented documents on the following products:

- Processor and internal options.
- Peripheral devices.
- Operating systems and other software.
- Any special packages.

Most importantly, you get clear concise answers about many of the problems particular to systems design or you get only the documentation needed to support your particular installation; it's your choice.

TABLE 5. SERIES 16 FEATURES

Communications	Control
Watchdog Timer – an optional capability that causes a program controlled interrupt if occurrence is monitored.	**Cathode Ray Tube (CRT) Devices** – together with the teletypewriter, excellent man/machine interface devices.
Direct Multiplex Control – Direct Memory Access – these optional input/output modes enable processing concurrent with direct data flow to and from the controller.	**On-Line Interfaces** – a complete selection of data acquisition and control as well as custom-designed interfaces permit certain control over a wider range of analog and digital inputs and outputs.
Fast Internal Cycle Time – ensures complete message processing even with the highest throughput rates.	**High Internal Cycle Time** – ensures speedy response to external events.
Interrupt Facilities – allows branching, as necessary, between main program flow, and real-time service routine.	
Multilevel Code Handling – translates, under program control, codes of up to 8 levels.	

DATA CENTERS

Honeywell maintains Series 16 Data Centers throughout the United States. Each contains a typical Series 16 configuration and serves the needs for all of our customers for a program development system prior to system delivery. Usually, there is no charge. For more extensive work, machine time is available on a per-hour-charge basis. Either way, the design of the Series 16 assures the user that programs designed to run on one Series 16 system will run on most similar systems without any drastic changes.

SYSTEMS SUPPORT SERVICE

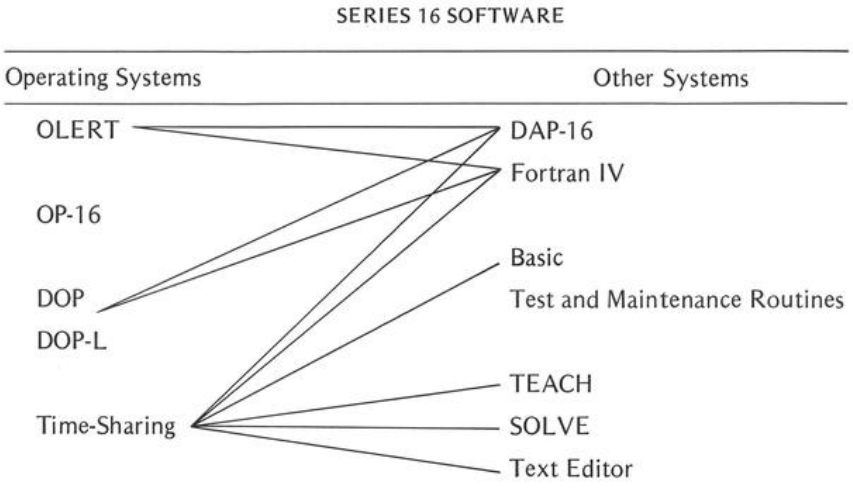
For every Honeywell salesman, more than ten other professionals are in reserve ensuring the user of complete support. Some of these support professionals, include the field service engineers who maintain installed Series 16 systems. During the design and implementation phase of systems design, support analysts and engineers are available for consultation on general and specific tasks. Or, if you desire, specific tasks will be completed by the engineers on a fee basis with or without user assistance.

FIELD MAINTENANCE

Maximum up-time is the result of Honeywell's extensive field support capability for the Series 16. This field maintenance organization allows Honeywell to insure reliability and maximum performance for the one monthly rental. And every product, whether it is a single-component or an entire system, gets exactly the amount of maintenance it requires.

What Can the Tools Do?

Several features of the Series 16 hardware complement make it especially suited to the two major minicomputer application areas. The table below defines these factors. Some, though keys to the Series 16 application in one area, are actually general parameters, important in all.



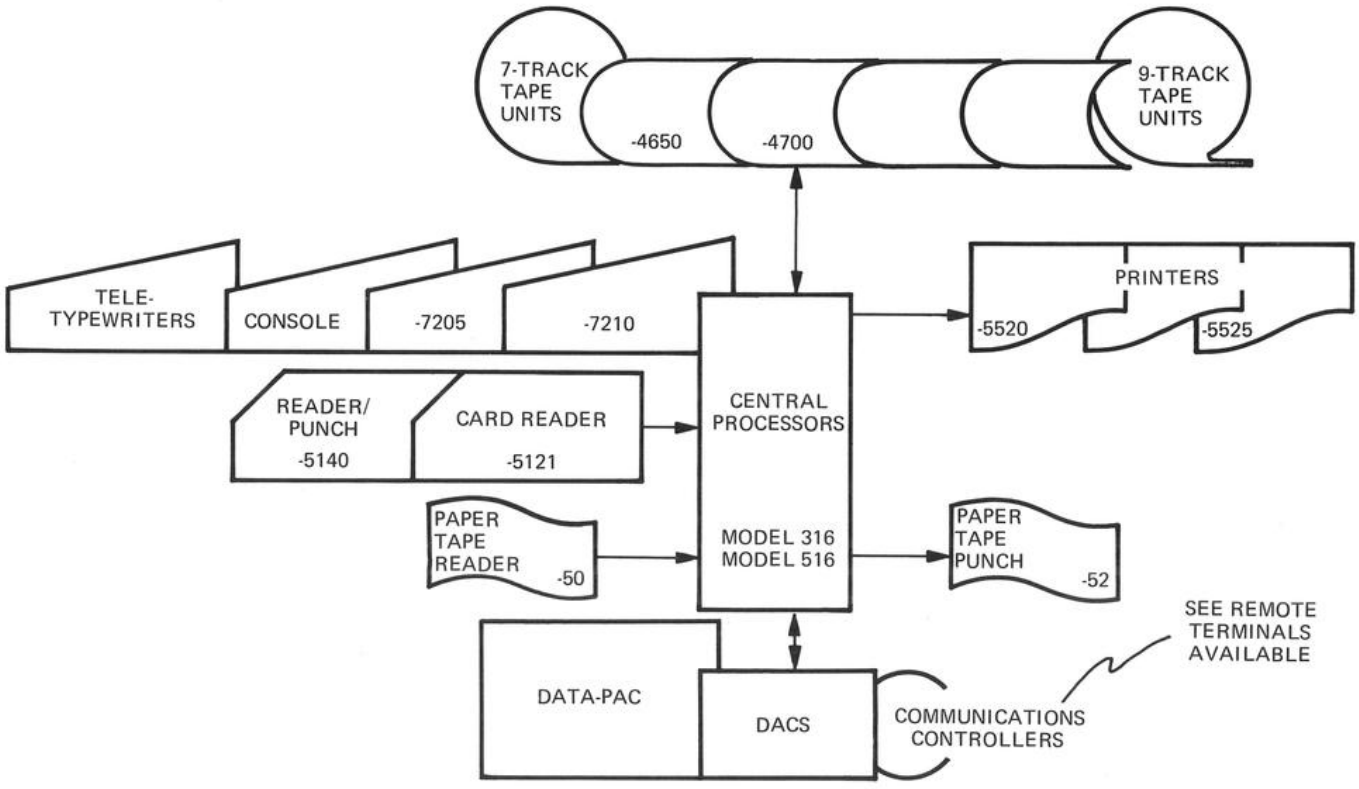


Figure 6. Series 16 System Components


TIMESHARING SYSTEMS

2

SERIES 16'S TOTAL SOLUTIONS

Total solutions save time and money. For the project or company under severe budgetary constraints, a Series 16 Application System might spell the difference between this year and next. Even for designers with the budgetary and personnel resources to implement their own solutions, the Series 16 applications systems:

- Put a project on-the-air sooner than otherwise possible.
- Free available manpower for other projects.
- Permit budget allocation flexibility in terms of both
  - present or proposed systems expansion, and
  - new systems development.

The Series 16 Application Systems are complete hardware/software/support packages combined to solve a variety of well-defined problems for which the Series 16 is an especially useful solution. Their easy and economical implementation and efficient operation combine to give a user the basic tools needed to perform the most complex functions.

Each package contains the Series 16 central processor that best does the required job in terms of computing power and economy. That is, just as an independent consultant or a staff designer would do, the Honeywell systems personnel have carefully weighed the capabilities and costs of the Models 316 and 516, and chosen, in each case, the hardware configuration best suited to the application.

Furthermore, each package utilizes the same flexible software systems available on all Series 16 processors. Not only does this factor ensure quality programming and the best application of the hardware's capabilities but it also enables the user to expand systems software at his own pace.

Finally, the complete maintenance and support structure of Honeywell backs the user of the Series 16 application systems. Included is the training required to fully implement each system and the on-going maintenance required to ensure maximum system up-time.

As discussed earlier, the Series 16 is especially suited to applications in the control and communications environments. In answering internal Honeywell (remember that the other half of the The Other Computer Company is the world's foremost automation company) or specific customer demands, Honeywell systems designers have developed applications sytems for the following environments:

- Data Acquisition and Control
  - the Models 1602, 1603, and 1605.
- Remote Message Concentrators
  - the Models 1621 and 1622.
- Time-Sharing Systems
  - the Models 1642 through 1648A.
- Clinical Laboratory Systems
  - from Honeywell Data Systems Operations.
- Ruggedized Systems
  - the Models 316R, 516R, and 1602R.

The development of systems in these and other categories is an on-going objective of the Series 16 design team.

DATA ACQUISITION AND CONTROL SYSTEMS

While the fully-developed data acquisition and control systems meet separate capability requirements, each contains three basic components:

1. *Series 16 hardware* – the central processor and peripherals needed to process and report on the acquired data.
2. *Real-Time Interface* – the specialized hardware required to easily interface with a wide variety of data acquisition devices; e.g., sensors, logic signals, control elements, displays.
3. *Series 16 software* – the operating system best suited to the application's need for data acquisition and control coordination.

Depending on the application, the data and control systems use either a standard or ruggedized 316 or 516 central processor. The Real Time Interface (RTI) regulates all signals between the monitoring and control elements of the external devices and the central processor. Each RTI:

- Decodes input and output instructions from the central processor and connects the selected input and output lines for data transfer.
- Conditions input signals.
- Converts analog input signals to digital form and digital outputs to analog form.
- Provides additional typewriters for logging data and alarm messages.
- Converts signals from computer levels to easily usable voltages and currents.
- Protects the CPU, the common or time-shared portion of the RTI, and related equipment from high-voltage faults in the field wiring between the customer plant and the RTI.

The features that apply to the Model 1602, 1603, and 1605 Data Acquisition and Control systems, design features found in few manufacturer's product lines, include:

TABLE 6. SUMMARY OF SERIES 1600 CAPABILITIES

	1602	1603	1605
Central Processor	316	316	516
Memory Sizes		4, 8, 12, 16, 24, or 32K	
Maximum Analog Inputs	64	2048	2048
Maximum Digital Inputs	256	4096	4096
Other Basic Capabilities	Real-time Clock, interface adapter, choice of cabinets		hi-boy cabinet, interface adapter, RTI master cabinet
Optional Capabilities	All Model 316 options ■ Digital and analog customer connection terminals ■ Analog input – digital input/output ■ Subsystem test modules (8, 40, 80 mV, 0.8, 4, 8 Vs) ■ Basic differential, differential with filter, and single-ended hi-level MUX PACs ■ 10, 50, 100 mVs, 1, 5, 10 Vs amplifiers, A/D converters. ■ Variable voltage modules ■ RTD Network ■ Many other digital capabilities		All Model 516 options

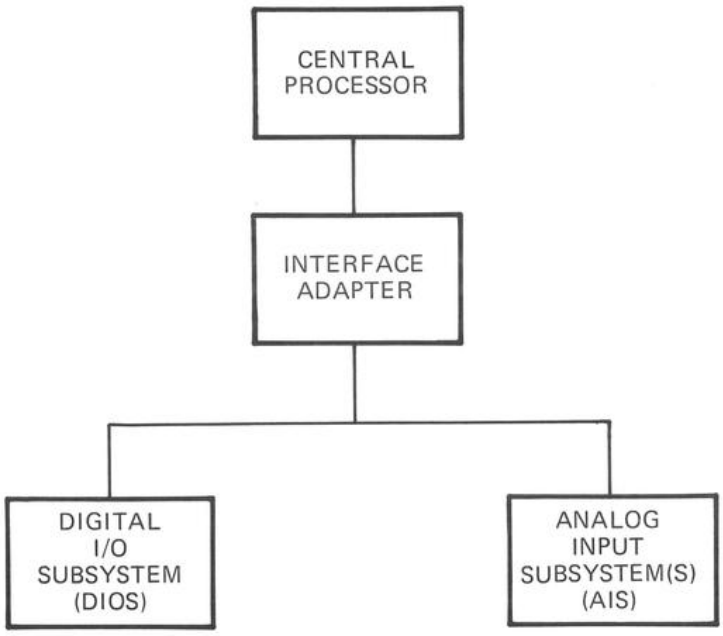


Figure 7. Data Acquisition and Control System

- 16-bit, stored-program, single-address, binary-parallel, central processor
- Up to: 125 analog inputs per second, using relay multiplexer or 20,000 analog inputs per second, with solid-state multiplexer
- Digital input and output system
- Wide variety of signal-conditioning modules
- Modular construction
- Circuit protection for signal inputs and outputs
- Maximum use of integrated circuits
- Customer terminal connectors

Organization

A data acquisition and control system normally includes:

- General-purpose central processor with 8,192 words of memory,
- Interface adapter,
- Real-time clock,
- Digital I/O and/or Analog Input Subsystems,

in a cabinet specific to the model number. With the insertion of the customer-specified plug-in, digital input/output modules, the data acquisition and control system becomes operational. System expansion requires additional digital modules and/or an analog subsystem.

DIGITAL INPUT/OUTPUT SUBSYSTEM (DIOS)

The self-contained digital input/output subsystem enables the system's central processor to accept and transfer digital data to and from external devices. Each master digital page accommodates a maximum of 16 input/output options. With additional pages (maximum of sixteen), a data acquisition and control system accommodates up to 256 input/output options. The options themselves accommodate from one to sixteen input/output points; for example, the digital/analog converter accounts for only one point while the status input option accounts for sixteen. Digital input options include:

- Status inputs used for central processor-initiated input of contact or voltage status.
- Asynchronous inputs used where an interrupt is initiated due to change in status.
- 8- or 16-bit counters to record status changes.

Digital output options available:

- Single-shot or flip/flop output capable of "sinking" current from field common and providing source current in the opposite direction.
- Power flip/flop or single-shot outputs capable of switching up to 500 mA at 48 Vdc.
- Isolated relays that can switch 100 VA under the direction of the power flip/flop or single-shot outputs.
- Digital-to-analog conversion capabilities under central processor directoin.

ANALOG INPUT SUBSYSTEM (AIS)

The AIS, also self-contained, has an input capability only. Subdivided into master pages, each with a maximum capability of 128 multiplexer points, an AIS converts analog information into digital pulses, meaningful to the processor. Expansion beyond 128 points requires the addition of slave pages, each of which also accommodates 128 multiplexer points. Total in a fully expanded analog input subsystem -- 1024 multiplexer points.

On the Models 1603 or 1605, two analog input subsystems (A and B), with a total of 2048 points, are permitted. In such a configuration, each subsystem contains independent power supplies, addressing logic, converter, and control logic. There is no time-sharing of components. As required, Subsystem B can operate under the central processor's Direct Multiplex Control facilities.

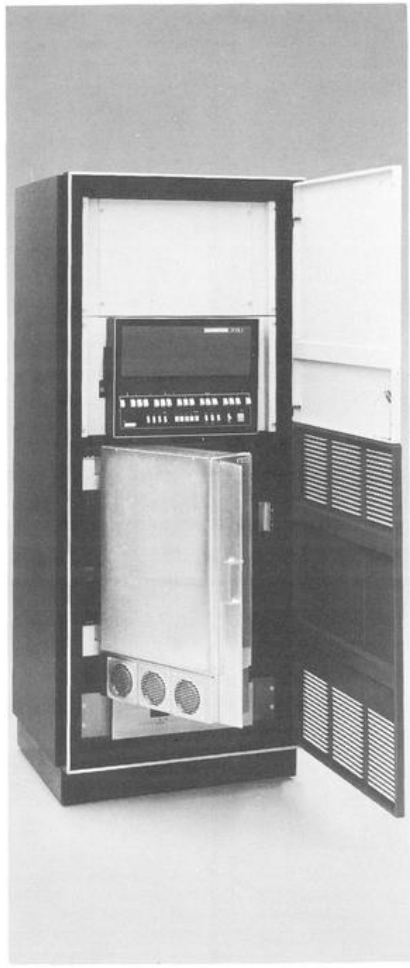
Each page of a subsystem accommodates one of three multiplexers and each multiplexer accommodates a different number of points. The only requirements are those stated above -- 128 points per page. The available options:

- a low-level relay multiplexer.
- a low-level solid-state multiplexer.
- a high-level solid-state multiplexer.

Each is provided in groups of eight channels; this submultiplexing concept minimizes analog bus and capacitance leakage.

The initiation of analog/digital conversion and the transfer of the converted value to the central processor occurs over the standard input/output facilities under the control of a stored program. In addition, where provided, the Direct Multiplex Control facilities can be utilized for block transfer of information via Subsystem B.

Standard signal-conditioning options include current-to-voltage, T/C isothermal compensation, open-circuit thermocouple detection, and resistance/thermometer bridges.



SERIES 1620 REMOTE MESSAGE CONCENTRATOR SYSTEMS

The Models 1621 and 1622 Remote Message Concentrators (RMC) are comprehensive, stored-program communication systems that reduce communications line costs and host-facility loading. A combination of hardware and software elements concentrate many low-speed data-communication lines onto a medium-speed data-communication line for transmission to a host facility.

The RMC's versatile hardware interfaces up to 128 full-duplex low-speed lines and up to 4 medium-speed output lines. Unattended operation is reinforced by the following:

- Power failure detect
- Automatic restart
- In conjunction with the software executive: down-line loading of operational programs triggered by a watchdog timer.
- High system reliability

The RMC software package provides a complete means of controlling the low-speed lines, both leased or switched. The medium-speed link to the host facility operates in a variety of line disciplines including:

1. A standard ASCII full-duplex line procedure.
2. IBM-360-compatible binary synchronous communications procedure in either ASCII or EBCDIC, both supporting full transparency.

Other outstanding features of the Models 1621 and 1622:

- Up to 128 FDX asynchronous low-speed lines
- Up to 300 baud
- Up to four medium-speed data lines
- Power failure detect/automatic restart
- Complete operational software package
- Plug-in turnkey installation
- Reconfigurable software
- Hardware bisynchronous cyclic redundancy check
- Easily expanded
- Efficient use of medium-speed line bandwidth
- Multispeed
- Message buffering
- Multicode
- Code conversion
- Down-line loading of operational programs.
- IBM bisynchronous compatibility

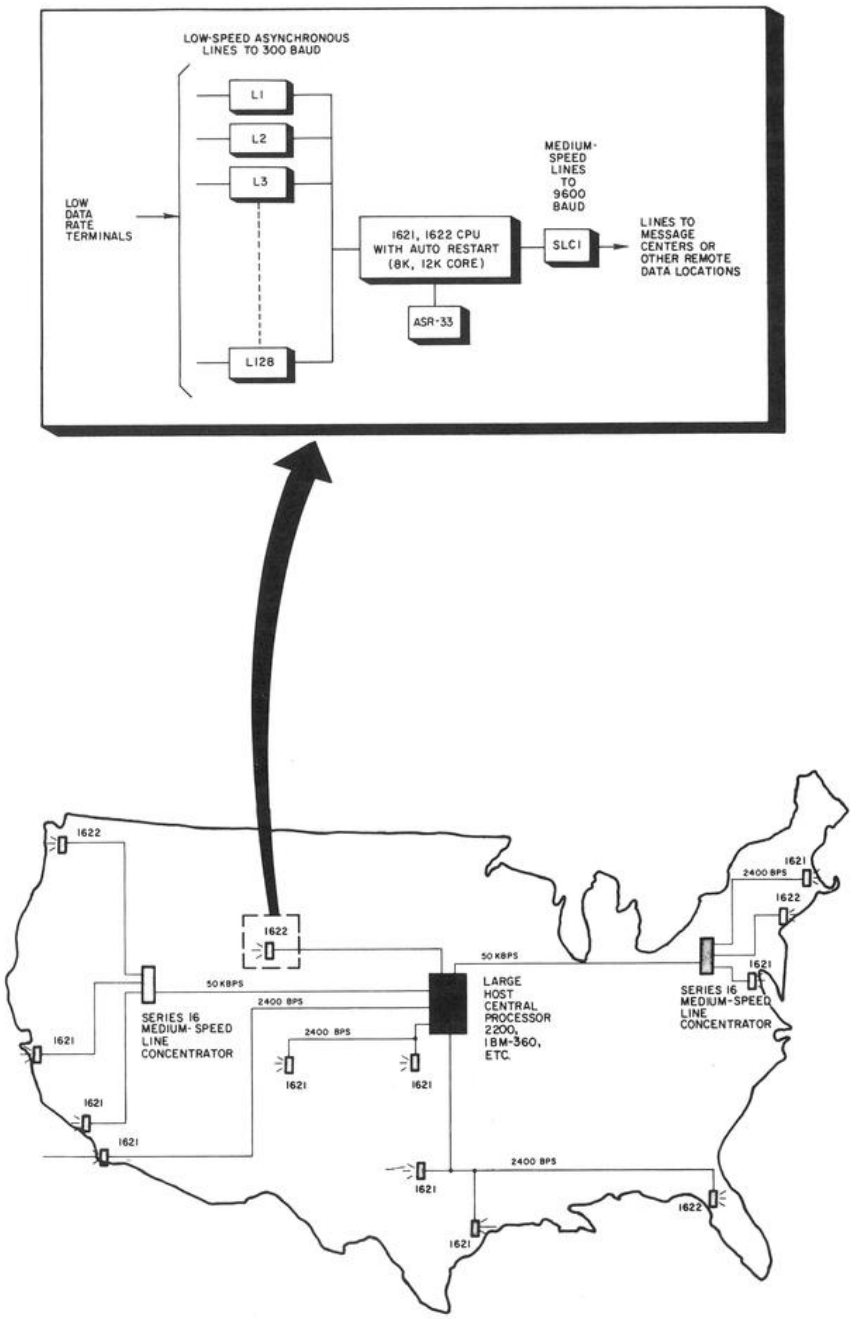


Figure 8. Typical Model 1622 Application.

TIME-SHARING SYSTEMS

The compatible members of Honeywell's new Series 1640 family of time-sharing systems are the Models 1642, 1644, 1646, and 1648A. The four are dedicated, conversational, problem-solving systems – based on the first time-sharing system ever built solely by a computer manufacturer, Honeywell. Series 16 processors and peripherals form the heart of the Series 1640. Starting with the Model 1642, a user can grow to the top of the line as needs increase.

The software, the important element of a time-sharing system, features:

Field Proven

Sixteen months were spent designing and implementing the time-sharing software – compilers, command language, and operating systems. Now there is a continually growing library of more than 200 utility, scientific, and applications packages. Countless thousands of hours have been logged in actual operation. Honeywell has invested heavily in time-sharing; and has planned a continuous software development, improvement, and enhancement program.

Easy to Use

Even users that do not understand computer concepts, compiler languages, or syntax, can solve problems with Series 1640 after only a few minutes' experience. All that is needed to compose, test, and execute solutions are 15 commands – for user identification, file maintenance, and language processor execution. For users already using a time-sharing service – don't be concerned about the move to Series 1640. The effort is straightforward and fully supported.

Many Languages

You have your pick of all these problem-solving languages: BASIC and EXTENDED BASIC – fundamental compiler languages resembling Fortran but easier to use; COBOL – an easy-to-learn language for commercial applications; full Fortran IV as specified by ASCII – designed for more sophisticated problem

TABLE 7. SUMMARY OF SERIES 1640 CAPABILITIES

	1642	1644	1646	1648A
User Program Size (without overlay)	10K	16K	16K	16K
Number simultaneous users	16-32	16-32	16-32	32-64
Maximum number of subscribers	960	960	960	960
Maximum storage available	36M	36M	93.6M	115.2M
Random files	No	No	Yes	Yes
On-line Tape Files	No	No	Yes	Yes
Floating-Point Hardware	No	No	No	Yes
Passworded Files	No	No	Yes	Yes
110 baud ASCII terminals	Yes	Yes	Yes	Yes
134.5 baud selectric	No	No	No	Yes
150 baud ASCII terminals	Yes	Yes	Yes	No
300 baud ASCII terminals	No	Yes	Yes	Yes
BASIC	Yes	Yes	Yes	Yes
Fortran IV	Yes	Yes	Yes	Yes
DAP-16	Yes	Yes	Yes	Yes
SOLVE	Yes	Yes	Yes	Yes
TEACH	Yes	Yes	Yes	Yes
EDIT	Yes	Yes	Yes	Yes
COBOL	No	Yes	Yes	Yes
EXTENDED BASIC	No	Yes	Yes	Yes
CHAINING	Yes	Yes	Yes	Yes
PROJECT ACCOUNTING	No	Yes	Yes	Yes
PUBLIC LIBRARY	Yes	Yes	Yes	Yes

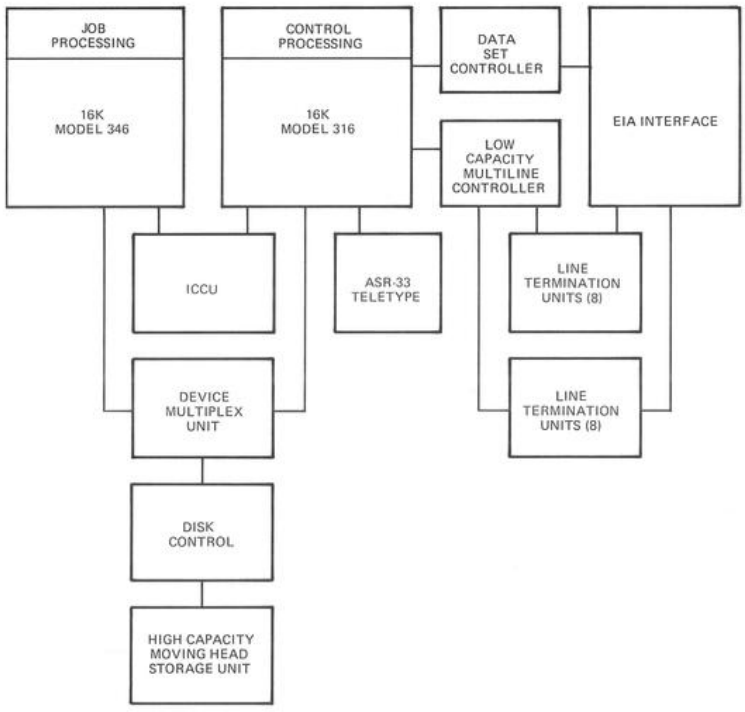


Figure 9. Model 1642 Time-Sharing System

solving; TEACH – an interpretive language that aids education; EDIT – an interactive context editor; SOLVE – a flexible, convenient language for conversational use; and DAP-16 – a symbolic assembly program.

For compatibility, the same language processors are used on all 1640 time-sharing systems. In addition, block sizes and linkages of file systems are identical for source and data files.

Advantages of an In-House System

Having your own time-sharing system is desirable in many instances – especially when you need terminals in many locations. Advantages are:

- Costs are less sensitive to the number of users accessing the system or the time they access it
- No extra charge for sophisticated applications
- No per-character or per-block storage charges
- Lower operating costs per terminal passed on to the user
- Dedicated port without an extra charge
- Configurations molded to your needs
- You control loading and operating schedules
- Your information files are protected against misuse
- No worry about access during peak loads
- Easier data transfers through on-site high-speed I/O devices
- Increased system reliability – decreased maintenance costs

Multiprocessing for Low System Overhead

The unique multiprocessor configurations of 1640 systems vastly improve performance over comparable time-sharing systems, and result in low system overhead. Instead of multiprogramming a single computer to make it a time-sharer, the tasks are divided among various Series 16 processors, allowing each to function with a simple monitor program.

All systems have a Job Processor (handles the execution of computational requests) and a Control Processor (interprets requests and schedules work). In addition, the 1648A has a Communications Processor which acts as a line handler. It receives information from the users' terminals and formats it for input to the other processors; transmits responses to the users.

Each model offers:

- High-speed arithmetic unit for more throughput
- Memory lockout permits system integrity when running unverified programs
- Restricted mode inhibits the execution of certain instructions for added protection
- Protected Section Selection protects against access of unauthorized memory
- Priority interrupt for quick response to proper service routines
- Field-proven hardware and software reliability
- Extended software: BASIC, EXTENDED BASIC, Fortran IV, COBOL, EDIT, TEACH, SOLVE, and DAP
- Off-line batch operating system
- From 16 to 64 simultaneous users
- 960 subscribers
- More than 200 utility, scientific, and applications programs
- Optional mag tapes, line printers, card reader, and card reader/punch
- Each user's time can be tracked
- Multiprocessor configurations minimize system overhead
- File storage from 7.2 million bytes to 115.2 million characters
- Magnetic tape backup storage (up to four units)
- Random access files in some configurations
- Lowest operating costs per terminal of any comparable system
- Ideal for in-house use
- Dedicated lines and/or remote connections via telephone coupling
- Lowest-cost field expandability
- Floating-point hardware at the top of the line
- Extensive training
- Thorough documentation

The Model 1642

The 1642 offers a starting point for prospects who want a small, capable, low-cost in-house time-sharing capability. A simplified, yet powerful command structure makes the 1642 easy to learn and use, and yet complete enough for experienced users who want the capabilities and features found only on larger, more expensive models. This instruction structure also serves as the basis for all other systems in the 1640 family. As the transition is made from the 1642 upwards, new commands are added, extending the facilities available to the user.

No other time-sharing system in its class is as expandable as the 1642. It has the largest data base of any system in its price range and the largest user memory. Also, it offers both Fortran and assembly language for non-time-sharing batch-processing operations, and is the smallest system offering on-line BASIC, Fortran, and editing capabilities.

The Model 1644

On the Model 1644, operating efficiency is increased by allowing up to three users with executable jobs to be resident in the job processor simultaneously. The result – overlapping of user execution, while the system is unloading the last job; greater efficiency and system utilization; more users using the system simultaneously. Features on the Model 1644, in addition to those found on the Model 1642:

- EXTENDED BASIC – string statements and matrix capabilities
- COBOL – a complete package for the development and execution of business programs in a time-sharing environment.
- Up to 16K words of memory for Fortran programs
- Project Accounting – the ability to credit portion of a terminal session against specific project numbers

Other benefits accrue to the 1644 user who has upgraded his system from a 1642:

- No user reeducation – system operations, languages, and programs learned on the 1642 apply to the 1644
- Proven software
- Greatly increased capability with a small increase in system price

The Model 1646

The 1646 offers increased power and efficiency at a minimal increase in cost. The use of larger processors permits better systems utilization and expanded capabilities. Features of the 1646 include those of the 1642 and 1644 plus:

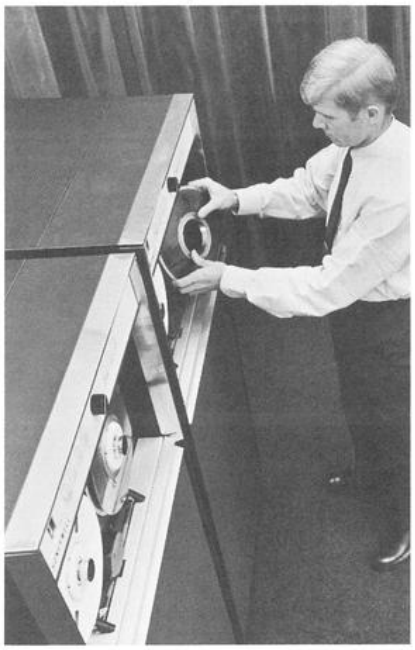
- Random file capability – for faster information access from storage or data base.
- Expanded user file storage – up to 93.6 million bytes of information may be sorted on-line
- High-speed terminal capability
- Double file protection – passwords may be added to storage file names
- Active tape files during time-sharing

The Model 1648A

The Model 1648A – fastest, most capable, most versatile member of the 1640 family, has these additional features:

- Optional floating-point arithmetic hardware – for speed and efficiency of processing
- Large storage capacity – up to 115.2 million bytes
- Up to 960 identifiable users
- Front-end communications hardware – for increased throughput

Up to 64 lines can be connected to the communications interface within the processor complex. Remote terminals are routed to the communications interface generally through low-speed, full duplex lines to multiplexers located at strategic concentration points, and from here to the communications interface via telephone lines.



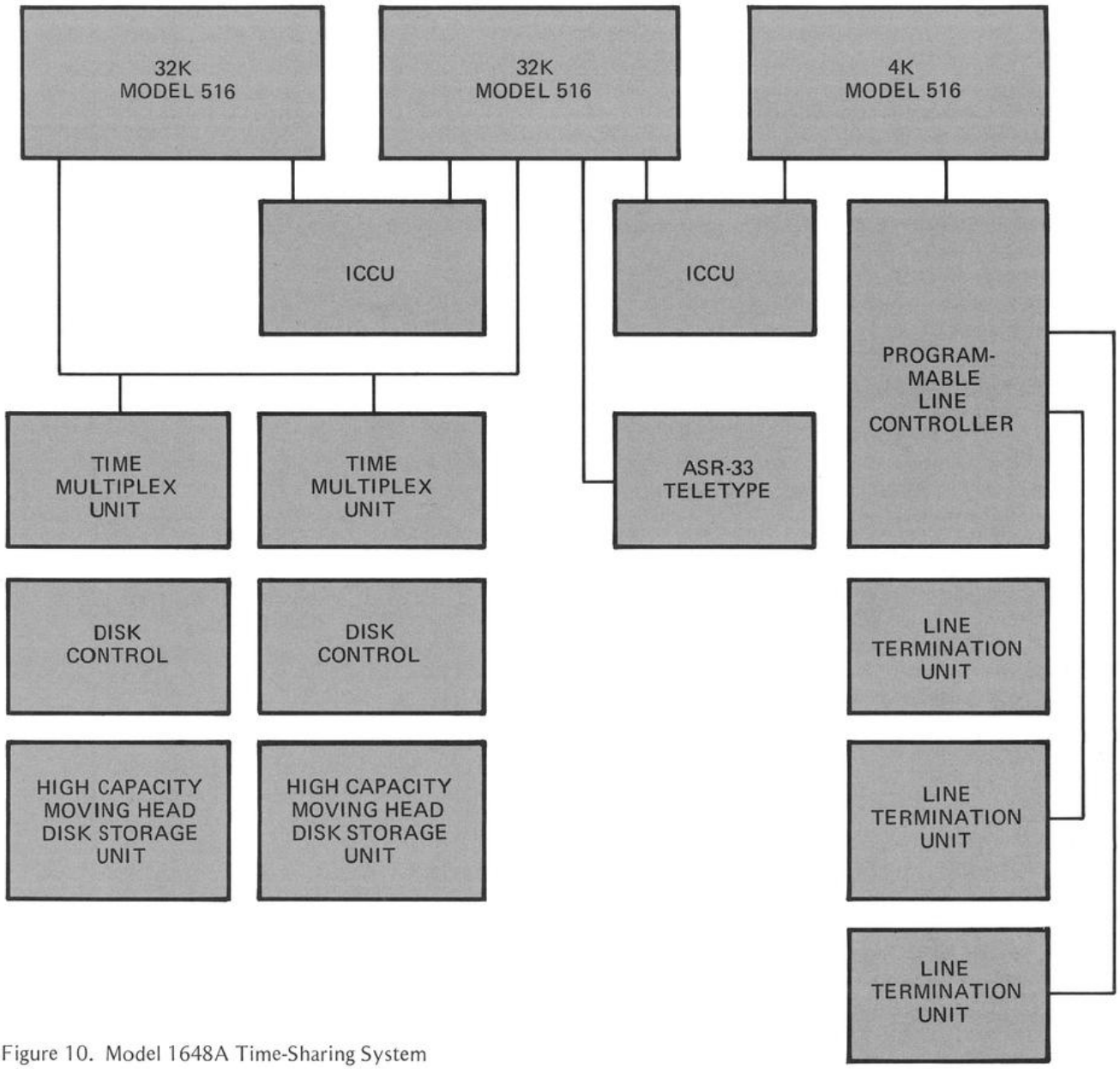


Figure 10. Model 1648A Time-Sharing System

COMPUTER-ASSISTED CLINICAL LABORATORY SYSTEM

As one of the world's oldest and largest suppliers of precision instrumentation, Honeywell is aware of the responsibility placed on medical data processing systems. All Honeywell clinical laboratory equipment has been designed with that responsibility clearly in mind, and our experience with hospital data management, biomedical systems, and many types of management information systems is our assurance of full system reliability.

In addition to a number of products designed specifically for the medical profession, Honeywell's wide selection of data processing equipment offers an almost limitless selection of system possibilities to satisfy the requirements of any clinical laboratory or research facility.

Although equipment may be supplied by any one of several specialized Honeywell divisions, the company's Data Systems Operations has sole responsibility for clinical laboratory systems. This assures the user that he is working with people who are expert in the unique requirements of the hospital laboratory, and are directly responsive to his needs.

In every case, Honeywell's objective is to implement the system to the exact requirements and specifications of the user. Honeywell provides the technical knowledge, the instrumentation, and the experience in designing systems. We provide the assurance that the system will operate efficiently and compatibly. But the user specifies what the system is expected to do, and for all intents and purposes, it is *his* system; designed for his exact requirements and no other.

Ideally, a centralized data system should have three characteristics:

- It should have complete system compatibility, with all equipment operating in harmony.
- There should be enough standardization to satisfy normal requirements at minimal cost, but with some design flexibility to handle most unusual needs when necessary.
- The system should be modular enough to allow for expansion or modification later.

In response to these requirements, Honeywell has developed systems that are compatible with either free-standing or shared-time operation, combine high-volume computer capabilities with modular "building block" techniques, and satisfy clinical laboratory data processing needs. The approach combines a variety of initial options with ample provision for revising or expanding the system later as the need arises.

Following are some characteristics of the Honeywell approach:

- Complete compatibility and efficiency of all equipment, including equipment not manufactured by Honeywell but specified as part of the Honeywell-designed system.
- Provision for later expansion or modification, incorporated in the "modular system" approach.
- Standardization of routine equipment, to reduce costs of systems that require only normal processing functions.
- On-line and real-time functions where desired, to allow moment-to-moment test readouts and immediate analog/digital monitoring information.
- Quality control capabilities, including test verification by professionals where needed, and computer-initiated alerts on out-of-limit readings.
- Ability to accept data directly, using special techniques that bypass punched card procedures where desirable.
- Information safeguard provisions, including verification codes and procedures to prevent unauthorized personnel from feeding information into, or reading it out of, the system.

A well-engineered computer-assisted system should, of course, offer the user a number of advantages over the older methods and clerical bookkeeping. Here are some of the advantages that Honeywell offers:

- Ability to handle large volumes of information, an ever-increasing necessity as hospital plans proliferate and third-party reimbursements change the older hospital work loads.
- Relate and categorize stored data. Computerized data can be broken down, categorized, annotated in an almost limitless number of ways, without upsetting the system routine. This simplifies many special studies and projects that might have been almost impossible with human recording.
- Permit more objective evaluation of data. The physician or other qualified evaluator receives data that is totally objective, never "weighted" by human data handlers.
- Provide fast, easy updating of information. Certain computer storage methods, such as optical reading forms or punched cards, provide a perfect way to handle current records that are subject to change. Magnetic tape and disk pack memory can also be revised as needed without encumbering the system.
- Reduces boredom and human error. Routine tasks cause fatigue and result in costly human mistakes. The computer frees people for the more valuable functions of supervision, critical analysis, professional training utilization and results verification.
- Reduces personnel costs. By reducing the number of clerical tasks necessary, the computerized system reduces operating costs; and by making the system more efficient, it reduces profit loss caused by duplication of effort, redundant transcriptions and repeated query responses.
- Provide a more dependable reference bank. The computer can store and retrieve information much more efficiently than clerical recording; can put entire libraries on fast recall.

RUGGEDIZED SYSTEMS

And where the application merits, ruggedized Series 16 systems have been developed to meet the most stringent military and industrial specifications. Ruggedized Series 16 systems include the Models 316R, 516R, and 1602R. Each ruggedized system features the same outstanding processing and throughput capabilities as its standard equivalent; in addition, each system has been tested to ensure its ability to withstand specified levels of punishment.

The Model 516R

The ruggedized version of the Model 516 general-purpose computer conforms, in general, with applicable military and shipboard specifications geared to meeting severe operational and environmental conditions. Certification procedures were conducted at an independent research and testing laboratory of Avco Corporation.

Certification came after the computer was shaken, shocked, rocked side to side, frozen, baked, swept to a 10,000-foot altitude, and bombarded with sudden changes in power supply, voltage, and frequency. It approaches a practically nonresonant system, (below 500 Hz) to satisfy long-term vibration specifications.

The same organization and characteristics which make the standard Model 516 so much in demand in numerous systems applications also make the ruggedized version exceptionally desirable for the wide range of configurations and uses in military and other special environments. The design approach used in the 516 closely parallels the requirements of a MIL-SPEC computer. For example, MIL-Q-9858 quality control practices are standard.

For installations where electro-magnetic interference (EMI) suppression is required, the Ruggedized 516 meets MIL-STD-826A. Modifications include an enclosure to provide shielding for the cabinet penetrations (input/output lines as well as air intake and exhaust), and incorporation of power line filters. The control panel configuration is also EMI compatible during all operational modes. These EMI suppression features are now standard in the Ruggedized 516.

The Ruggedized 516 is already in widespread use, both military and commercial. For example, it has been selected for installation on 39 U.S. Coast Guard ocean-station cutters to gather data for weather forecasting and airborne target tracking.

MANY OUTSTANDING FEATURES

All the capabilities of the standard commercial 516 are incorporated in the ruggedized system; field-proven design, reliability, flexibility, full software selection, modular options, plus spare parts provisioning and full service support. Wiring portion and logic cards, heart of the computer, remain the same.

The compact low-cost, high-performance 16-bit binary 516 has a standard memory capacity of 4,096 words (expandable to 32,768) and a 960-nanosecond cycle time. Its extremely high computation and I/O speeds ideally qualify it for real-time on-line systems applications.

Other Model 516R features:

- Front-access cabinet doors simplify maintenance.
- Controls for normal operations are externally located.
- Major assemblies are removable, and convenient test points are located in the integral backboard wiring to aid in diagnostic procedures.
- Diagnostic routines allow trained technical personnel to achieve a mean-time-to-repair of less than 20 minutes.
- Power supply is easily accessible, and replacement of any logic card can be made in less than two minutes.

Reliability and small physical size are the great advantages of this powerful machine which can be used in remote locations or where space is restricted. The Ruggedized 516 has the same MTBF of 10,000 hours as the standard computer. The compact cabinet, occupying approximately 28 cubic feet, can easily pass through most shipboard hatchways and aircraft or van doors.

- Certified to MIL specs for ruggedization, environment, and EMI/RFI.

The Model 316R

The Model 316R ruggedized general-purpose computer is a member of the growing Honeywell Series 16 family of on-line, real-time, 16-bit systems. All capabilities and features of the standard Model 316 are incorporated in the ruggedized version, which has been certified to MIL environmental and EMI specifications. Smaller than the standard 316, it is a compact, low-cost/high-performance machine which has standard memory of 4,096 words (expandable to 32,768) with 1.6 microsecond cycle time.

The Ruggedized 316 meets the severe environmental requirements demanded by military and industrial applications. A rack-mountable and a table model are available for shipboard, van, aircraft, oil and gas rig, ground station, and industrial installations. Both models provide EMI/RFI protection. The Ruggedized 316 parallels the requirements of the "MIL-SPEC" computer.

The Ruggedized 316 uses the same standard connector planes, function boards, and logic cards as the standard models. The central processing unit, integral 50-to-400 Hz power supply, 4,096 to 16,384 words of main memory in 4K modules, and several standard 316 options can be contained in the basic unit. Additional memory and options are accommodated in an expansion drawer which is mechanically identical to the basic unit. The enclosures for the basic unit and expansions are EMI/RFI-shielded structures

with removable top and bottom covers that provide access to circuit modules and power supply. A front control panel contains the operation switches and displays. A keylock switch on the control panel disables the logic functions on the panel from unauthorized operation. All connections are made at the rear panel through MIL connectors/mates. The standard rack-mountable model is supplied with slides; the optional (no charge) table model has side plates that can accommodate hard or shock mounting.

Standard 316 documentation includes much of the information required by MIL SPECS. Provisioning documentation for supply depots can also be provided.

The Model 316's characteristics and organization make it desirable for the wide range of configurations and uses in military, industrial, and other environments where ruggedization is necessary.

FEATURES

- Certified to MIL SPECS, for ruggedization, environment, and EMI/RFI.
- Compact and portable for movement between installations.
- Standard rack-mountable and optional (at no charge) table model for installation flexibility.
- EMI/RFI suppression provided as standard feature.
- Standard keylock switch disables the logic functions on the control panel from unauthorized operation.
- 22-½ inch diagonal dimension allows submarine-hatch entry.
- Occupies less than 2.9 cubic feet.
- Weighs less than 90 pounds.
- All major assemblies are removable.
- A standard ruggedized power supply allows input power frequency of 50 to 400 Hz.
- All capabilities and advantages of the standard Model 316 are maintained.
- Compatible with all Series 16 computers, peripherals, and software.
- Modular, to simplify future expansion.

The Model 1602R

The Ruggedized Model 1602 is an EMI-protected Data Acquisition and Control System consisting of a 316 central processor, peripherals, and a Real Time Interface (RTI) for digital and analog signals. The system is ideally suited for applications in hostile environments where expansion beyond 256 digital inputs/outputs and 64 analog inputs are not required. It is fully supported by the OLERT operating system (see Data Acquisition and Control). The RTI regulates all signals between the monitoring and control elements of the external devices and the 316. It is used for ground support equipment, mobile applications, and laboratory systems. The RTI:

- Decodes input and output instructions from the central processor and connects the selected input and output lines for data transfer.
- Conditions input signals.
- Converts analog input signals to digital form and digital outputs to analog form.
- Provides additional typewriters for logging data and alarm messages.
- Converts signals from computer levels to easily usable voltages and currents.
- Protects the CPU, the common or time-shared portion of the RTI, and related equipment from high-voltage faults in the field wiring between the customer plant and the RTI.

The system meets the MIL Specs listed under GENERAL SPECIFICATIONS. However for full EMI protection, optional adaptors must be used. Also, in order to meet some hostile industrial requirements, the environmental option – which provides a pressurized purged system and a heat exchanger – should be used.

FEATURES

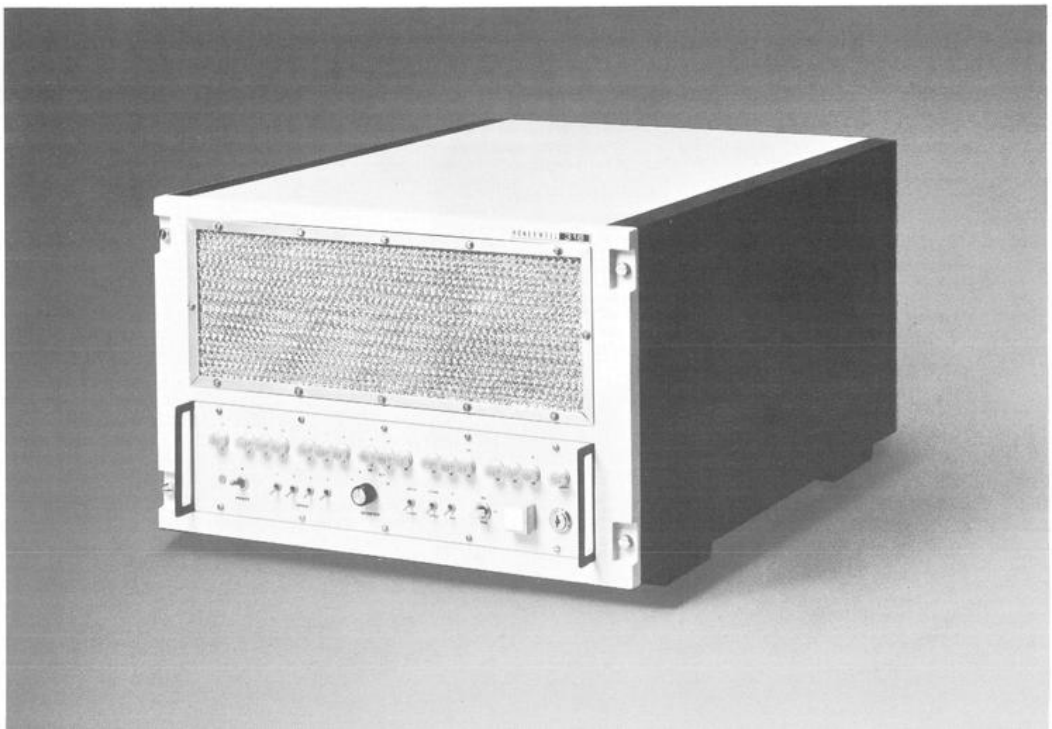
- Ruggedized system with EMI protection
- 16-bit, stored-program, single-address, binary-parallel, central processor
- 1.6-μs memory cycle time
- 16-bit series computer peripherals
- Up to: 125 analog inputs per second, using relay multiplexer or 20,000 analog inputs per second, with solid-state multiplexer
- Digital input and output system
- Wide variety of signal-conditioning modules
- Modular construction
- Circuit protection for signal inputs and outputs
- Maximum use of integrated circuits
- Terminations for real world connections (optional)

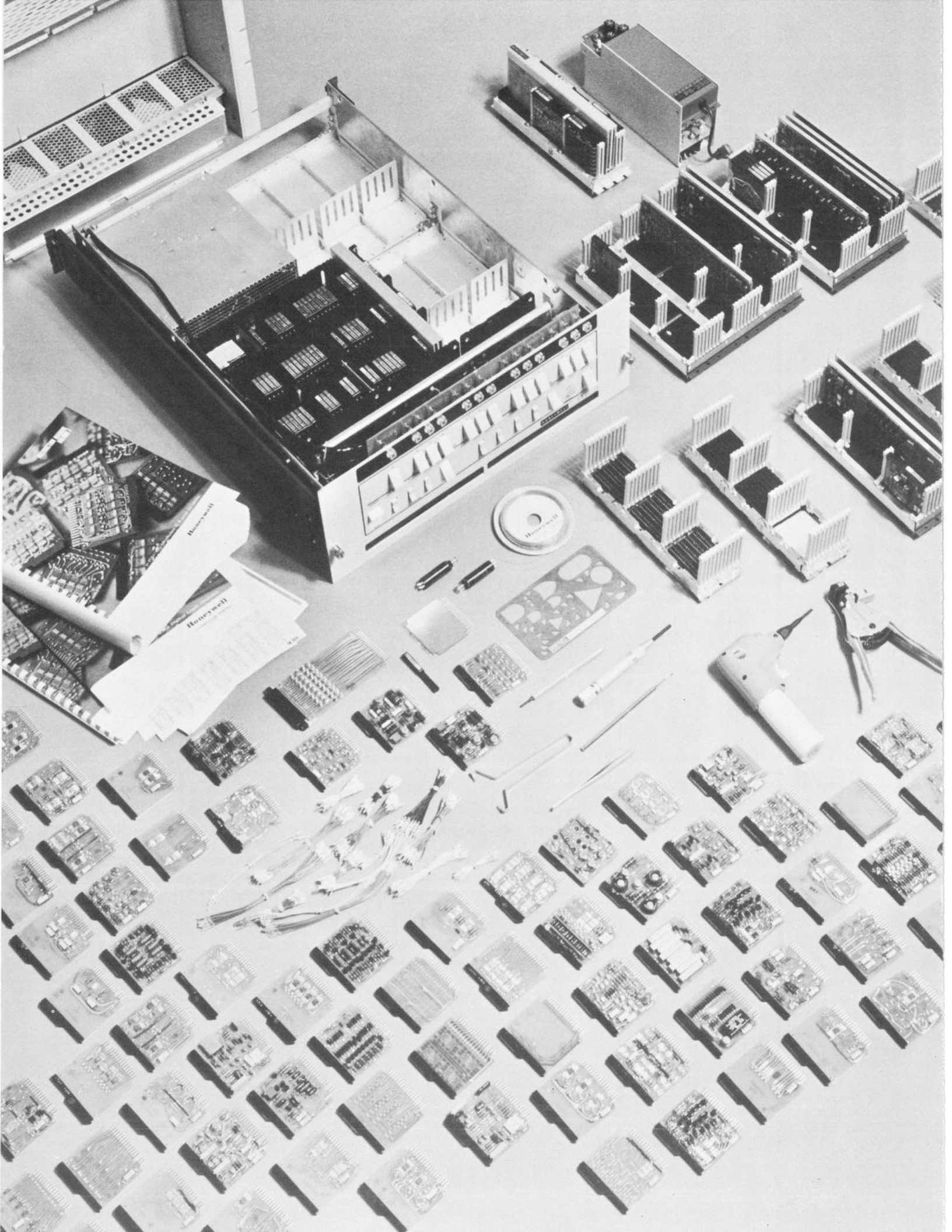
TABLE 8. SUMMARY OF RUGGEDIZED SYSTEMS SPECIFICATIONS

GENERAL	316R	516R	1602R
Dimensions	Rack-Mountable Model – 12-¼"H x 19"W (16-¾"ID) x 24-¼"D with connectors Table Model – 13-¼"H x 19"W x 24-¼"D with connectors	Vertical cabinet (hi-boy): 72"H x 24"W x 28"D Horizontal cabinet (lo-boy): 42"H x 24"W x 28"D	72"H x 24"W x 30"D
Weight	Less than 90 pounds	Vertical cabinet: 557 to 792 lbs. Horizontal cabinet: 487 to 560 lbs.	
Power	115 VAC, 50-400 Hz, 5.3 A, 475 W (max.)	Permissible voltage perturbations at 60 Hz ±5% are 115 Vac ±10%. It has a transient voltage capability of +30%, – 15%, for a maximum of two seconds. Normal power consumption is 1400 W.	115 Vac, 50-60 Hz, 5.5 A, 500 W (nominal)
Colon	Gray, Federal Standard 595, Chip #26622 Black, Federal Standard 595, Chip #27038	Finishes are to military specifications for cabinet and exposed metals.	Gray, Federal Standard 595 to match Chip #26492
Other	TEST CERTIFICATION The 316R table model, table model/isolators and rack-mountable models have been certified, to the MIL specs listed under Environmental Specifications, at the Raytheon Environmental Test Labs. Detailed information on test procedures, equipment, and results is available in Honeywell 316R Qualification Report: Volume I, Introduction and Environmental Test Report; Volume II, EMI Test Report; and Volume III, Detailed Failure Reports and Test Procedures.	MATERIALS Flame retardant printed circuit boards, nonfungus nutritive materials, nontoxic material, nonfragile materials or other materials listed in MIL-E-16400, Paragraph 3.5.3.4. ALTITUDE Operating: Up to 10,000 feet. Nonoperating: Up to 50,000 feet. EMI SUPPRESSION Per MIL-STD-826A	
ENVIRONMENTAL			
Shock	Operating: MIL-STD-810B, Method 516, Procedure I MIL-E-5400, Par. 3.2.21.6.1 MIL-T-21200, Par. 3.2.16.5.1 15g, half sine wave, 11-ms duration, 18 impact shocks, three shocks in opposite directions along each of the three perpendicular axes.	Operating: 0 to 5g to 0 in 11ms perpendicular to the base, 0 to 2g to 0 in 11ms, perpendicular to the side at the base. Nonoperating: 0 to 15g to 0 in 11ms.	Operating: MIL-STD-810B, Method 516, Procedure 1 (modified to 5g peak acceleration) (a) 5g, half sine wave, 11ms duration, perpendicular to the base. (b) 5g, half sine wave, 11ms duration perpendicular to the sides at the base. Nonoperating: MIL-STD-810B, Method 516, Procedure 1 (a) 15g, half sine wave, 11ms duration, perpendicular to the base. (b) 15g, half sine wave, 11ms duration, perpendicular to the sides at the base. (c) Three shocks in opposite directions along each of the three perpendicular axes (total, 18 shocks).
Vibration	Operating: MIL-STD-810B, Method 514, Procedure I MIL-E-16400, Par. 4.5.14.1 (MIL-STD-167, Type I) MIL-T-21200, Par. 3.2.16.4, Class II and III 5-15 Hz at .060" ± .012" D.A. 16-25 Hz at .040" ± .008" D.A. 26-33 Hz at .020" ± .044" D.A. 34-40 Hz at .010" ± .002" D.A. 41-55 Hz at .006" ± .001" D.A. Five minutes at each discrete frequency per axis. AIRCRAFT AND HELICOPTER Operating: MIL-STD-810B, Method 514, Procedure I, Curve Z, 10g MIL-E-54000, Par. 3.2.31.5.1, Curve IV, 10g MIL-T-21200, Par. 3.2.16.4, Class I, MIL-T-5422, Curve IV, 10g 5-20 Hz at .10" D. A. 20-33 Hz at ± 2g 33-74 Hz at .036" D.A. 74-500 Hz at ± 10g Sweep time 5-500-5 Hz – 15 minutes, 3 cycles (45 minutes) of vibration performed in each of three mutually perpendicular axes. Thirty minute dwell at each resonance.	Vibration per MIL-STD-167 over the following range of frequency and amplitude. *Frequency (c/s)* — *Amplitude (inches)* 5 - 15 — 0.030 ± 0.006 16 - 25 — 0.020 ± 0.004 26 - 33 — 0.010 ± 0.002	Operating: MIL-T-21200H, Par. 3.2.19.4, Class 2, 3; MIL-STD-167, Type 1 (a) Frequency 5 to 15 Hz, table amplitude (+ or -) 0.030 ± 0.006 in. (b) Frequency 16 to 25 Hz, table amplitude (+ or -) 0.020 ± 0.004 in. (c) Frequency 26 to 55 Hz, table amplitude (+ or -) 0.010 ± 0.002 in.

TABLE 8. SUMMARY OF RUGGEDIZED SYSTEMS SPECIFICATIONS (Cont.)

ENVIRONMENTAL	316R	516R	1602R
Inclination	Operating: MIL-E-16400, Par. 4.5.14.2 45° on either side of the computer vertical axis at a rate of 5 to 7 cycles per minute, for a period of 30 minutes in the horizontal and horizontal transverse axes.	Operating: Up to 45° from vertical in any direction.	Operating/Nonoperating: MIL-E-16400F, Par. 4.5.14.2. The unit must be inclinable up to 45° on either side of the computer vertical axis.
Temperature	Operating: MIL-STD-810B, Methods 501 and 502, Procedure I MIL-T-21200, Class III 0°C to +55°C Nonoperating: MIL-STD-810B, Methods 501 and 502, Procedure I MIL-T-21200, Class I MIL-E-5400, Class I -62°C to +85°C	Operating: 0°C to 50°C Nonoperating: -55°C to +80°C	Operating: MIL-STD-810B, Method 501, Procedure 1 and Method 502 – 0° to 55°C Nonoperating: MIL-STD-810B, Method 501, Procedure 1 and Method 502, -55° to +85°C
Altitude	Operating: MIL-STD-810B, Method 500, Procedure I MIL-T-21200, Class III 20,000 feet (13.8" Hg), 0°C to +55°C Nonoperating: MIL-STD-810B, Method 500, Procedure I MIL-T-21200, Class I, II, and III 50,000 feet (3.4" Hg), -62°C to +85°C		Operating: MIL-T-21200H, Par. 3.2.19.1, Class II and III, and MIL-STD-810B, Method 500, Procedure I. (a) 10,000 ft. (20.6" Hg) (b) Standard ambient temperature (23°C ± 10°) Nonoperating: MIL-T-21200H, Par. 3.2.19.1, Class I, II, and III, and MIL-STD-810B, Method 500, Procedure I. (a) 50,000 ft. (3.44" Hg) (b) Standard ambient temperature (23°C ± 10°)
Humidity	Operating: MIL-STD-810B, Method 507, Procedure I 95% RH, 30°C to 55°C, five days Nonoperating: MIL-STD-810B, Method 507, Procedure I 95% HR, 30° to 71°C, five days	Operating/Nonoperating: Up to 95% with no condensation.	Operating: MIL-STD-810B, Method 507, Procedure IV (Modified per Par. 3.4.8.2). (a) 95% RH (b) 30 to 55°C (c) No condensation
Electrical	Operating: MIL-E-16400, Par. 4.5.4, supply line voltage and frequency, transient voltage, transient frequency. Frequency (Hz): 50 / 60 / 400 — Nominal; 47-53 / 57-63 / 380-420 — Steady State Change; 1.5 Hz / 1.8 Hz / 12.0 Hz — Transient Voltage: 115 — Nominal; 103-127 — Steady State Change; ±23 VAC RMS — Transient Time (s): continuous — Nominal; continuous — Steady State Change; 2 sec. — Transient	Electrical specifications meet the following paragraphs of MIL-E-16400 (no external power conditioning equipment is required): *Paragraph 3.9.1, Primary Power Supply Line Voltage and Frequency* – The computer will meet the performance limits of operation over the steady state tolerance of ±10% in voltage and ±5% in frequency from nominal values specified. *Paragraph 3.9.1.1, Transient Voltage* – The computer is capable of standing a voltage transient of ±20% of the nominal from any point within the ±10% steady state tolerance band within two seconds. Momentary impairment of operation during the transient does not cause failure of any part nor prevent the resumption of normal operation, nor does it require the equipment to be recycled when the transient has ceased. *Paragraph 3.9.1.2, Transient Frequency* – The computer is capable of operation during frequency transients of 3% of which not more than 1% is outside the steady state tolerance band of ±5%. The transient is recoverable to a point within the steady state tolerance band within two seconds.	Operating: MIL-E-16400F, Par. 3.9.1 Frequency (Hz): 50 / 60 / 400 — Nominal; 47-53 / 57-63 / 380-420 — Steady State Change; 1.5 / 1.8 / 12.0 — Transient Voltage (Vac): 115 — Nominal; 103-127 — Steady State Change; 23 — Transient Time (s): continuous — Nominal; continuous — Steady State Change; 2 — Transient
Electromagnetic Interference	MIL-STD-461, 462, 463 (a) Conducted Emission CEO3 20 KHz to 50 MHz, power leads (b) Conducted Susceptibility CSO1 30 Hz to 50 KHz, power leads CSO2, 50 KHz to 400 MHz, power leads CSO6 Spike, power leads (c) Radiated Emission REO1 30 Hz to 30 KHz, magnetic field REO2 14 KHz to 1.0 GHz, electric field (d) Radiated Susceptibility RSO1 30 Hz to 30 KHz, magnetic field RSO2 magnetic induction field RSO3 14 KHz to 1 GHz, electric field		MIL-STD-461, 462, 463 (a) Conducted interference – 30 Hz to 50 MHz (b) Conducted susceptibility – 30 Hz to 400 MHz (c) Radiated emission – 30 Hz to 50 KHz (magnetic); 14 Hz to 1 GHz (electric) (d) Radiated susceptibility – 30 Hz to 30 KHz (magnetic); 30 Hz to 150 KHz (magnetic induction) 14 Hz to 1 GHz (electric)





3

SERIES 16'S BUILDING BLOCKS

Depending on an application's requirements, the extensive array of system components available within the Series 16 may be a better choice than one of the total solutions. Choose one or all of the components from Honeywell (Honeywell offers more of a choice than any other manufacturer) and be assured of the same kind of support that Honeywell places behind entire packages and products. Series 16's building blocks have applications in all departments and companies where a designer can succinctly state his control or communication problems and define their required solutions.

Because Honeywell designs every component as part of a complete, integral, compatible scheme, only logic design is needed before use. Therefore, the user avoids the problems that often cripple a systems implementation effort. With Honeywell building blocks serving in all, or part of a system, performance is sure, certain, and on-time. Available components:

- Central processors – the Models 316 and 516 – that have proven themselves in test after test and application after application.
- On-line interfaces – built, or build them yourself – for a variety of control, communication, and hybrid applications.
- A wide array of peripheral devices.
- Software – operating and programming systems – that ensure the efficient and on-going operation of the hardware according to the designer's specifications.

CENTRAL PROCESSORS

Four compatible central processors are included in the past and present Series 16. The Model 116, the industry's first 16-bit minicomputer, and the Model 416 both now unavailable, are predecessors of the current offerings:

- Model 316
- Model 516

Each processor offers the designer the internal speed, memory size, and throughput capability best suited to his application. Unless noted, all interfaces, peripherals, and software systems are compatible with both processors. Table 9 indicates the components available on the Series 16 central processors; their mutual capabilities include:

- The flexibility of design; for example, the processor's 16-bit architecture permits halfword, single-precision, or double-precision manipulation.
- The modular construction; for example, micro-PAC construction permits exactly the configuration that best suits a particular budget and application. Series 16 modularity extends across the processor's architecture – memory, input/output facility, arithmetic capabilities.
- Systems simplicity; for example, the completeness of the instruction repertoire permits easy application of the Series 16 processor to the given application.

Organization

A Series 16 central processor is logically, as well as functionally, divided into four parts: (see Figure 11)

- memory
- the control register
- the arithmetic registers
- the input/output facilities.

In addition, the arithmetic and control registers share an adder (not shown). Normal peripheral data transfer occurs in parallel over the input/output bus. Controls for all peripheral devices and interfaces used in most Series 16 applications are μpacs within the central processor cabinetry, but are not functionally a part of the central processor.

MEMORY

In a Series 16 processor, a memory location contains 16 bits; this 16-bit

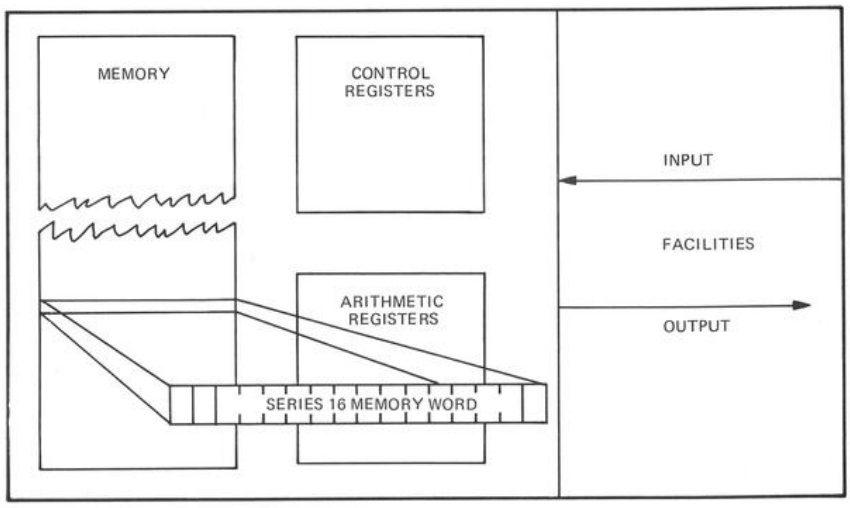


Figure 11. Logical Division of a Series 16 Central Processor

location is termed a "word." There are 4,096 words in a memory module; Series 16 central processors have a maximum storage capacity of 32,768 words. Both program instructions and the data to be manipulated by the program reside in memory during a program operation. As a key to programming capabilities, every word is identified by a unique numeric address (Figure 12). This allows the program to reference the exact memory location containing needed data.

Except for double-precision arithmetic information, each unit of data and each instruction occupies one, fixed, 16-bit location. Each of the 16 bits within a location is individually magnetized to represent a "one" or a "zero." Depending on whether the location contains an instruction or data, the format of the 16 bits varies.

Memory parity (see Table 9 for ordering information) adds a seventeenth bit to each memory location. This parity bit, used in connection with logic gating, performs an automatic error-detection operation known as parity checking ("automatic" in that it occurs without a specific program instruction). Detection of a parity error sets an indicator that can be tested under program control and used to initiate an interrupt.

The time interval required by the processor to read and restore the contents of a memory location is a basic unit of computer systems performance measurement – memory cycle time. The cycle times associated with the Series 16 processors (1.6 microseconds for the Model 316; 0.96 microseconds for the Model 516) are key elements in their ability to perform the many applications assigned them.

Priority access to memory is a final capability important to the real-time power of the Series 16. The various functions and capabilities of the processing logic are executed in a priority sequence, if two or more are accessing memory simultaneously. The priorities are:

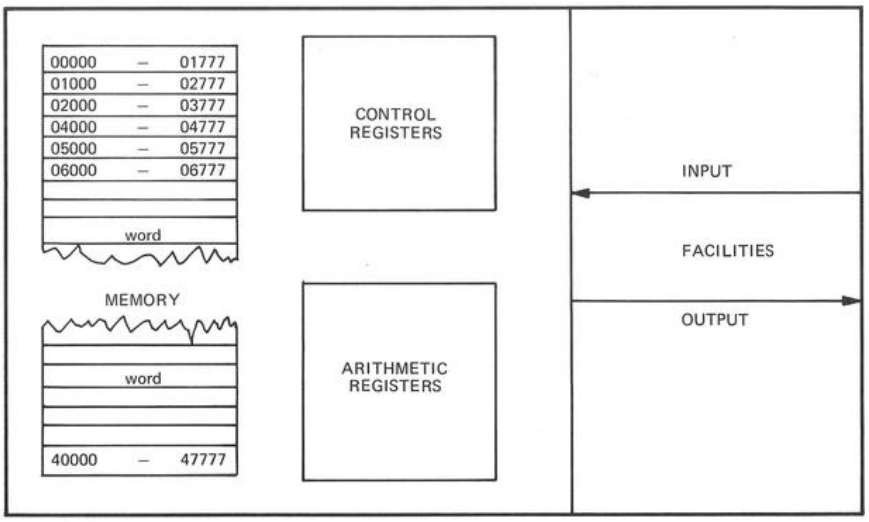


Figure 12. Address and Contents in a Series 16 Memory

1. Direct Memory Access (Feature – 21 Break
2. Direct Multiplex Control (Feature – 20) Break
3. Power Failure Interrupt
4. Real-Time Clock (Feature – 12) Break
5. Memory Lockout Violation (Feature – 08(00)) Interrupt
6. Input/Output Interrupt
7. Memory Increment Break
8. Priority Interrupt
9. Program Execution

Addressing

For addressing purposes, memory is divided into 512-word sectors of which any 1024 are directly addressable.

As illustrated below, the Flag (F), Tag (T), and Sector (S) bits of a memory reference instruction determine the addressing mode:

F	T	S	
0	0	0	Direct to base sector
0	0	1	Direct to current sector
0	1	0	Direct to base sector, indexed
0	1	1	Direct to current sector, indexed
1	0	0	Indirect through the base sector
1	0	1	Indirect through current sector
1	1	0	Indirect through base sector, pre-indexed
1	1	1	Indirect through current sector, pre-indexed

Multilevel, to 16K core, can be pre- or post-indexed. This method adds 1.6 μs per level to instruction execution time. When indirect addressing is required, the effective address is assumed to be in the location specified by the address portion of the instruction and the selected sector address. However, if this location also calls for indirect addressing, another cycle is initialed. For all instructions which permit indirect addressing, the chain can continue indefinitely.

When the index bit of an instruction is set, the contents of the index register is added to the effective address of the instruction to produce a new effective address. If indexing is specified in a given instruction, it occurs before any indirect addressing; if specified in an indirect address, it occurs before any further indirect addressing. Most importantly no additional cycles are required for instruction execution when indexing.

TABLE 9. CONFIGURING A SERIES 16 SYSTEM

Model 316		Model 516	
Type No.	Description	Type No.	Description
316-01	Model 316 computer, 4K memory, rack-mountable configuration, includes control panel and power supply	516-01	DDP-516 general-purpose digital computer with 4,096 words of core memory
316-02	Additional 4K memory	516-02	Same as 516-01 with 8,192 words of core memory
316-0601	Extended-mode operation for Model 316	516-03	Same as 516-01 with 12,288 words of core memory
316-0602	Expansion package for greater than 16K core	516-04	Same as 516-01 with 16,384 words of core memory
316-0701	Parity for first 4K of core	516-05	Same as 516-01 with 24,576 words of core memory
316-0702	Parity for each additional 4K of core	516-06	Same as 516-01 with 32,768 words of core memory
316-0800	Base sector relocation	516-07-1	Parity for 4K memory module
316-0801	Memory lockout for first 8K of core	516-07-2	Parity for 8K memory module
316-0802	Memory lockout for each additional 8K of core	516-08	Memory-lockout system for first 8K of memory
316-11	High-speed arithmetic package	516-08-1	Memory-lockout system for each additional 8K
316-12	Real-time clock	516-11	High-speed arithmetic package
316-20	Direct multiplex control (DMC) unit, controls 16 devices	516-12	Real-time clock
316-21	High-speed DMC, controls 16 devices	516-20	Direct multiplex control unit (DMC)
316-25	Group of four priority interrupt lines	516-21	Direct memory access control unit (DMA) and one channel
316-25-1	Additional group of four priority interrupt lines	516-21-1	Additional DMA channel
316-26	Memory counter modification for group of four priority interrupt lines	516-25	Group of four priority interrupt lines
316-29	16 SKS and 16 OCP lines	516-25-1	Additional groups of four priority interrupt lines
		516-26	Memory counter modification for group of four priority interrupt lines
		516-29	16 SKS and 16 OCP lines
		516-32	Parallel input channel
		516-33	Parallel output channel
		516-34	Buffered parallel I/O channel

TABLE 10. SUMMARY OF PROCESSING REGISTERS

Register	Function	Mnemonic
	Control Registers	
Program Counter	16-bit register containing the location of the next instruction to be executed.	P
Memory Register	16-bit register used to transfer information to and from memory.	M
Address Register	16-bit register containing the location in memory to or from which information is being transferred.	Y
Index Register	16-bit register used in address modification.	X
	Arithmetic Registers	
Primary Arithmetic Register	16-bit register in which all arithmetic and logical bit manipulation occurs.	A
Secondary Arithmetic Register	16-bit register used in conjunction with A when arithmetic operands exceed one word in length.	B
Overflow Bit	1-bit register used to indicate an overflow status in the arithmetic register.	C
Adder	logic gating, which performs all arithmetic operations.	—

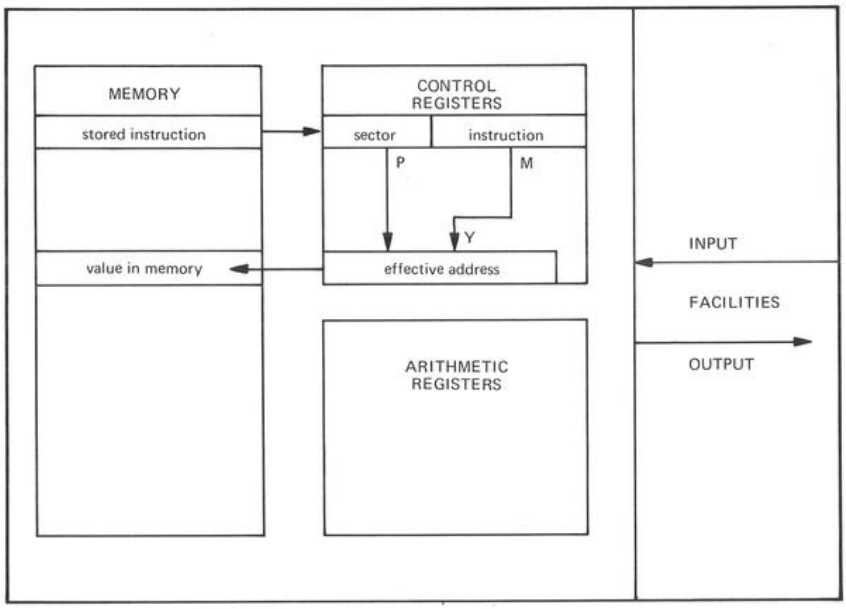


Figure 13. Typical Series 16 Control Register Activity

CONTROL REGISTERS

There are four processing control registers. These registers (see Table 10) normally contain the addresses of instructions and data being processed during a program run. For example one such register, the address register (Y register), contains the location in memory to which information is being transferred or from which it is being retrieved. The operation in which the value of this location is established is called "effective address formation." Figure 13 illustrates this typical control unit function. In the illustration, an instruction stored in memory is loaded in the memory register (M register). After the address portion of this instruction is added to the portion of the program counter (P register) that designates the memory sector, the result is stored in the Y register as the instruction is executed.

On both Series 16 processors control registers can be displayed at the operator's control console. For instance, the operator can interrogate the program counter to determine the exact location at which a program has halted. A register is addressed from the control panel by pressing the appropriate push buttons; the contents will appear in an "on" or "off" status on the 16 control panel indicators that represent a memory location.

The Index register (X register), if indexed addressing is specified, is utilized in address formation and modification.

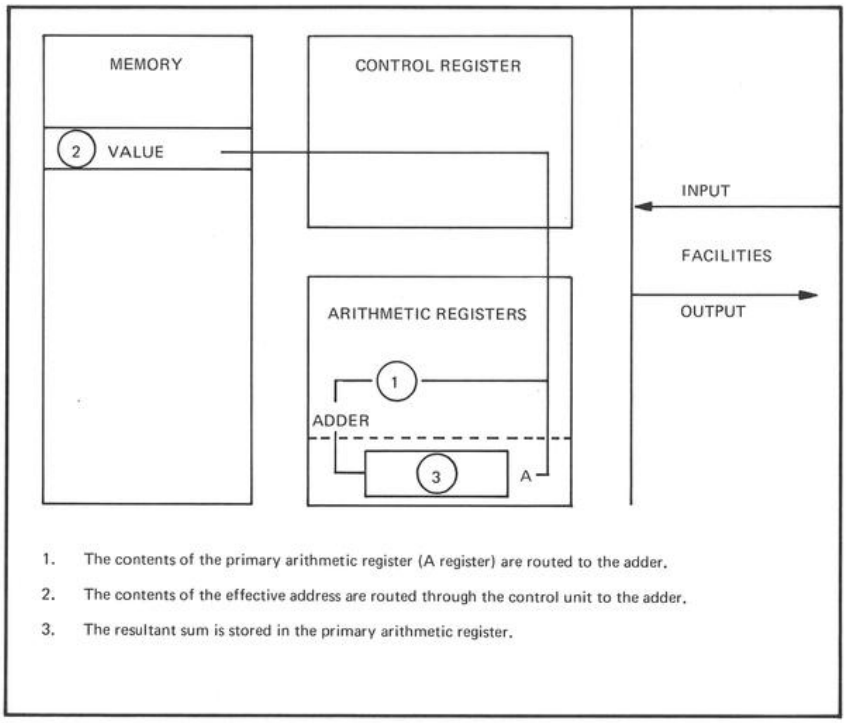


Figure 14. Typical Series 16 Arithmetic Registers Function

ARITHMETIC REGISTERS

Arithmetic and logical operations are performed in the adder and in a series of arithmetic registers (Table 10). Including a primary arithmetic register (A register), a secondary arithmetic register (B register), and an overflow indicator (C bit), the arithmetic registers give the processors the great computing flexibility so important to process and bit manipulation applications.

Figure 14 illustrates a typical arithmetic function – an add operation. Before this operation begins, the control registers have established the "effective address" in memory and the nature of the operation to be performed has been stored. Acting under these established facts, the control directs the addition operation. If the programmer desires to return the sum to a location in memory, another instruction is required.

The functioning of the other arithmetic registers, the B register and the C bit, depends on the nature of the instructions being executed.

The arithmetic registers are accessible from the control console, as well as by programmed instruction. An operator can change their contents.

Both the control and the arithmetic registers function under the control of a clocking system that enables the processor to rapidly select, interpret, and execute all instructions in a stored program. This clocking system also coordinates the various activities of receiving data from input devices and transferring data to output devices.

INPUT/OUTPUT FACILITIES

The input/output facilities consist of external control and sense lines, 16-bit input lines, 16-bit output lines, a 10-bit device address line, and an interrupt capability that utilizes these lines. The Direct Multiplexor Control (DMC) and the Direct Memory Access (DMA) features are extensions of the input/output facilities.

■ Single-Word Transfer Without Interrupt

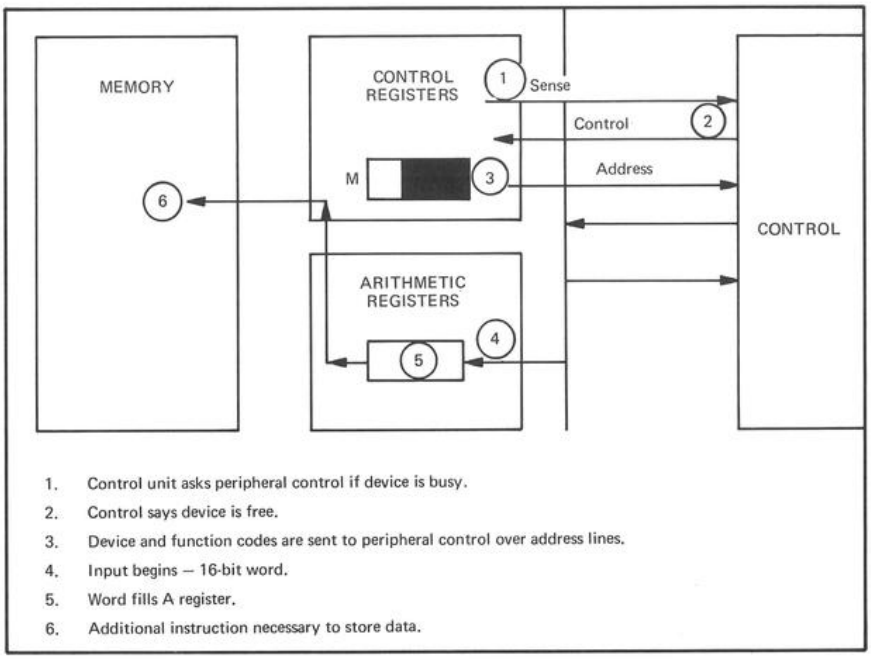
The first type of data transfer operation, single word without interrupt, is illustrated in Figure 15. In the example, the sense lines establish contact with a peripheral's control logic; then, control lines carry control characters, data, and responses to and from the control. These pulses define the meaning of the information on the input/output bus. Basically, the processor asks the peripheral device whether or not it is busy; if it is not busy, data transfer begins.

Five instructions are used in conjunction with peripheral data transfer. One of these, Skip if Ready Line Set (SKS) is the instruction that asks the control if it is busy. Two other instructions, Input Data to A (INA) and Data Output from A (OTA), are then used to transfer the data.

In the example the choice of data transfer mode (input or output) was previously established by an Output Control Pulse (OCP) instruction. A final instruction, Set Mask (SMK), is used in conjunction with the next mode of data transfer, single-word transfer with interrupt.

Once the SKS instruction has established contact with the peripheral control, the device and operation codes, stored in the memory register, proceed to the control over the address lines. Data returns over the input lines to the control registers, which route all data in this mode, one word at a time, to the primary arithmetic unit (A). Once the data is in the arithmetic registers, input is complete. The programmer must then specify a memory reference instruction in order to place the data in a memory location.

Output in this mode also involves the sensing of the peripheral control by the central processor. Once the SKS instruction has established that the peripheral control is free (in the manner described above), output from the arithmetic register can begin, one word at a time.



1. Control unit asks peripheral control if device is busy.
2. Control says device is free.
3. Device and function codes are sent to peripheral control over address lines.
4. Input begins – 16-bit word.
5. Word fills A register.
6. Additional instruction necessary to store data.

Figure 15. Single-Word Transfer Without Interrupt

■ Single-Word Transfer with Interrupt

One of the control lines of the input/output facility is an interrupt line. All peripheral devices are connected to the interrupt line. When any peripheral device is ready to transmit information, and other interrupt conditions are met, an interrupt signal reaches the central processor.

This procedure saves the processor the trouble of continually sensing the transmit status of the peripheral devices. All interrupts are stored until they are serviced, and service begins at the end of any instruction if the interrupt has priority access to memory.

An interrupt demand causes the program to branch to an interrupt routine. This routine determines the device that is causing the interrupt, re-establishes higher priority interrupts, and begins data transfer. Data transfer then occurs in the same manner as in single-word transfer without interrupt except that the single-word transfer is under the control of the interrupt routine instead of the main body of the program. Both processors have the capability for 16 individually maskable priority interrupts. Up to 48 priority interrupt lines are permitted on the Model 516.

■ DMC Data Transfer

A Direct Multiplex Control (DMC) permits the peripheral data transfer of blocks of data directly to memory; that is, the data being transferred bypasses the arithmetic register. Typical multiplexor channel input is illustrated in Figure 16. The DMC is required with high-speed devices, such as magnetic tape units and disk pack drives.

In the example, a peripheral device makes a request for service to the multiplexor channel. When the request is honored, the beginning and ending addresses of the data transmission are retrieved from their dedicated locations in memory (where they had been stored previously) and placed in the DMC.

Data transfer begins from the point indicated by the starting address and occurs until the ending address is reached. Computation can occur simultaneously with this mode of transfer; that is, an instruction can be executed in the time between the transfer of one word and the peripheral device or communication controller's next request signal. Because this transfer avoids the arithmetic register, transfer speeds reach 156K words per second.

■ Direct Memory Access (DMA)

Available only on the Model 516, this channel provides an alternate path to memory. I/O transfers are processed on a cycle stealing basis. Up to four subchannels, each with its own address and range registers, may be multiplexed into the DMA channel. In the time-shared mode, DMA interrupts processing for 0.96 μs per word. Maximum response time from data request until transfer complete is 1.92 μs on input, 2.64 on output. In the block transfer mode, with program processing suspended, DMA I/O rates exceed one million 16-bit words per second.

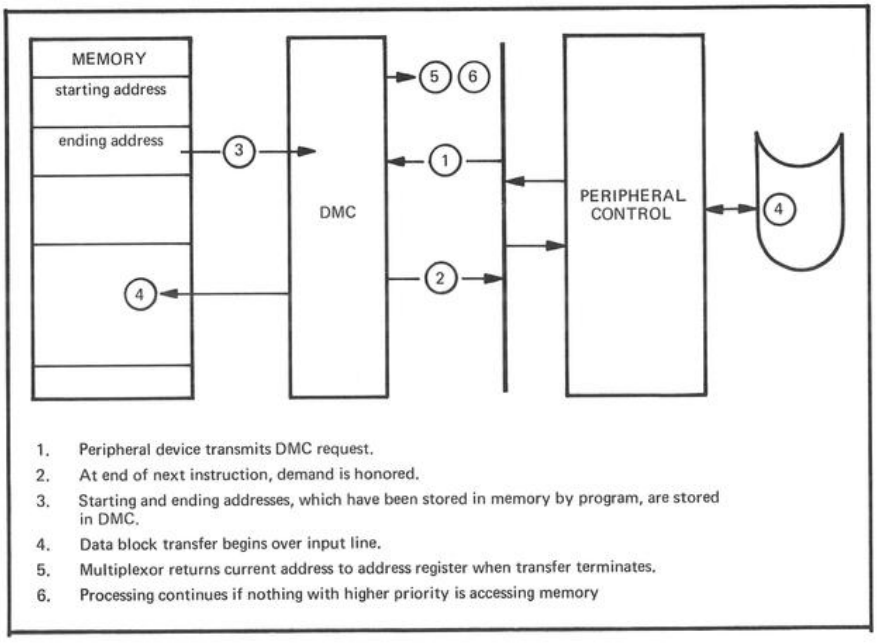


Figure 16. Direct Multiplex Control Input

Operation

Series 16 Processors provide high performance and simple, direct operation characteristics. Operator control specifications have been engineered for flexibility and are augmented by extensive information display. Design features include built-in power failure protection and enable ease of maintenance.

The standard remote console contains binary displays in octal representation, run status displays, and all operator controls. By pressing the appropriate select switch, the operator can display register contents, and memory information as well as internal counters and flip-flop status.

Registers can be cleared and/or altered from the panel and the contents of memory locations can be displayed or changed. (Memory locations 1-15 contain program load instructions and can be loaded manually only.)

Other control panel functions include selection of operation mode (memory access, single instruction, continuous run) four sense switches for control of programs, and a power failure interrupt inhibit switch.

In addition to a control panel, Series 16 processors utilize a teletypewriter keyboard unit with paper tape reader and punch. When used in conjunction with software checkout functions, this I/O device becomes the primary control for breakpointing programs, changing memory contents, displaying memory, and related functions. It can also be used off-line for preparation, duplication, or listing of program tapes.

Programs are executed sequentially with the contents of the program counter (P register) incrementing by one upon the execution of each instruction. Certain instructions (SKIPS, COMPARE, I/O) conditionally increment the program counter by an additional one or two, thereby causing a skip. Others (JUMP, JUMP-STORE) unconditionally load the program counter with a new effective address, thereby causing a branch in the program.

MEMORY-ACCESS PRIORITY CAPABILITY

Only one function can have access to memory at any given time. The various functions that the computer performs are executed in a priority sequence if two or more functions are trying to access memory simultaneously. The following shows the relative priorities between the program and breaks and interrupts.

Breaks

Certain operations may occur between instructions without affecting the contents of the program counter. When the operations are complete, the program resumes. These actions are called "breaks" and include the operation of transferring data via direct multiplex control and incrementing the real-time clock.

It may be noted that the program being executed and other operations are always computing and vying for access to memory. Only one function at any given time can have access to memory and that is either the program, the interrupt demand function, the real-time clock, or the multiplexor channel.

Interrupts

A 2-level interrupt facility provides simple, but efficient, supervision of processing involving combinations of input/output and computing. This facility allows branching as necessary between a main program and servicing routines for all input/output devices. It eliminates the need for programmed tests to detect the completion of input/output operations. A flexible interrupt capability has important applications in the field of data communications and other real-time areas, but it is equally applicable to the supervision of operations as universal as reading and punching cards or reading and writing magnetic tape.

The first interrupt level (highest) is for a power failure. The second interrupt level is for all other interrupts. The second level provides a multilevel priority interrupt structure under program control.

A program interrupt occurs whenever a peripheral device has completed an input/output operation. For example, an interrupt occurs at the end of data transmission in a tape read or write operation. Likewise the receipt of a character from a remote station by a communication control may be signalled by a program interrupt. Interrupts from a particular peripheral control can be allowed or inhibited by a program as necessary.

A program interrupt is accompanied by: (1) automatic storage of the interrupted program's location and (2) automatic branching to a routine whose address was previously loaded by program into a dedicated memory location. This routine can then proceed to determine the number and source of existing interrupts and to process the corresponding input/output demands according to whatever priority was specified by the programmer.

Series 16 processors feature a repertoire of instructions which, with tremendous flexibility and power, can handle all arithmetic, logical, control, and input/output functions necessary for control and communication processing. Also included in the processors are instructions dealing with peripherals and communications interrupts and for handling data in 5- to 8-level codes. Instructions available with the processors are shown in Table 11.

TABLE 11. SERIES 16 INSTRUCTION REPERTOIRE

Name of Operation	Mnemonic
Arithmetic Instructions	
Add	ADD
Subtract	SUB
Add One to A	AOA
Twos Complement	TCA
Add C to A	ACA
Logical Instructions	
And to A	ANA
Exclusive OR to A	ERA
Copy Sign and Set Sign Plus	CSA
Complement the Sign of A	CHS
Complement the A Register	CMA
Set Sign Minus	SSM
Set Sign Plus	SSP
Control Instructions	
Compare	CAS
Increment, Replace, and Skip	IRS
Enable Interrupt	ENB
Disable Interrupt	INH
No Operation	NOP
Halt	HLT
Unconditional Jump	JMP
Jump and Store Location	JST
Reset C bit	RCB
Set C bit	SCB
Unconditional Skip	SKP
Skip if Low order bit One	SLN
Skip if Low-order bit Zero	SLZ
Skip if A Minus	SMI
Skip if A Plus	SPL
Skip if A Not Zero	SNZ
Skip if A Zero	SZE
Skip if SENSE Switch #n is Reset:	
	SR1
	SR2
	SR3
	SR4
Skip if SENSE Switch #n is Set:	
	SS1
	SS2
	SS3
	SS4
Skip if C Set	SSC
Skip if C Reset	SRC
Skip if Any SENSE Switch Set	SSR
Skip if No SENSE Switch Set	SSS

Name of Operation	Mnemonic
Data-Handling Instructions	
Load A	LDA
Store A	STA
Clear A	CRA
Interchange A and B	IAB
Interchange Memory and A	IMA
Input Keys	INK
Output Keys	OTK
Load Index	LDX
Store Index	STX
Shift Instructions	
Logical Left Rotate	ALR
Logical Right Rotate	ARR
Arithmetic Left Shift	ALS
Arithmetic Right Shift	ARS
Logical Left Shift	LGL
Logical Right Shift	LGR
Long Left Logical Shift	LLL
Long Right Logical Shift	LRL
Long Left Rotate	LLR
Long Right Rotate	LRR
Long Arithmetic Left Shift	LLS
Long Arithmetic Right Shift	LRS
Byte-Handling Instructions	
Interchange Characters in A	ICA
Interchange and Clear Left Half of A	ICL
Interchange and Clear Right Half of A	ICR
Clear A, Left Half	CAL
Clear A, Right Half	CAR
Input/Output Instructions	
Output Control Pulse	OCP
Skip if Set	SKS
Input to A	INA
Output from A	OTA
Set Mask	SMK
Feature -11, High-Speed Arithmetic Instructions	
Double-precision Add	DAD
Double-precision Subtract	DSB
Multiply	MPY
Divide	DIV
Normalize	NRM
Shift Count to A	SCA
Enter Double-precision	DBL
Enter Single-precision	SGL
Double-precision Load A	DLD
Double-precision Store A	DST

Word Formats

The hardware and software aspect of word formatting in the Series 16 processors depends on the type of word, data or instruction. The processor transfers data between memory and the A register in units of 16 information bits. Depending on the internal timing of the processor, these bits can be interpreted as being a data word or an instruction word.

Both data- and instruction-word formats are divided into four types:

DATA WORD

Sixteen unsigned bits are used to express a logical data word; for example, the condition of sixteen binary indicators. These words are treated logically by the central processor.

BYTE WORD

Data can also be formatted as two 8-bit bytes. Such a word is used by the Series 16 central processor in half-word operations; especially, in communications routines.

SINGLE-PRECISION NUMBER

Single-precision numbers contain a sign bit plus fifteen magnitude bits. The low-order bit is the sign bit. Single-precision numbers can represent an integer with a numerical range of $\pm 2^{15}$ or ±32,768. They can also represent a fraction in which the decimal point is fixed; that is, the point is fixed by the programmer somewhere within the number.

DOUBLE-PRECISION NUMBER

A double-precision number contains a sign bit and thirty magnitude bits, encompassing two memory words. Double-precision numbers can represent integers or fixed-point fractions. As an integer, the numerical range is $\pm 2^{30}$ or ±1,073,741,824.

The first word of a double-precision number contains the sign bit plus the fifteen most significant bits. The second word contains the fifteen least significant bits of the number. The low-order bit of the second memory word is ignored.

Double-precision numbers are used by the processor to express the product of multiplication, as well as in all high-speed arithmetic operations.

MEMORY REFERENCE INSTRUCTION

Instructions that involve storing or retrieving data from memory are classified as memory reference instructions. The memory reference instructions are identified by the format shown.
Bit 1, the F bit, denotes indirect addressing; bit 2, the T bit, denotes indexing. Bits 3 through 6 contain the operation code that defines the function to be performed. For ease of communication, operation codes are generally expressed either in octal or as a mnemonic. "Subtract," for example, which has an operation code bit configuration of 0111, is referenced in machine language as (07_8) and has a mnemonic of SUB. Instructions available with the processors are shown in Table 1. The memory sector bit (S) indicates whether the base or currently executing sector is being referenced.

INPUT/OUTPUT INSTRUCTION

Instructions that involve the control of or the data transfer to input/output devices are input/output instructions. In the input/output instruction word format shown, bits 1 through 6 specify the particular input/output instruction and bits 11 through 16 specify the device that is being addressed. Bits 7 through 10 define the specific function to be performed by the instruction.

SHIFT INSTRUCTION

Instructions that involve the movement of data in the A and B registers are classified as shift instructions. The shift instruction word format is illustrated below. Bits 1 through 10 specify the type of shift and bits 11 through 16 define the number of shifts to be performed. In machine language format, the number of shifts is expressed in twos-complement form. However, the programmer codes this number in decimal and the software makes the translation.

GENERIC INSTRUCTION

All other instructions (clears, skips, halts, etc.) are referred to as generic. Generic instructions are identified by a word format as shown below. Bits 1 through 16 denote the operation code. An example of a generic instruction is Clear the A Register (CRA).

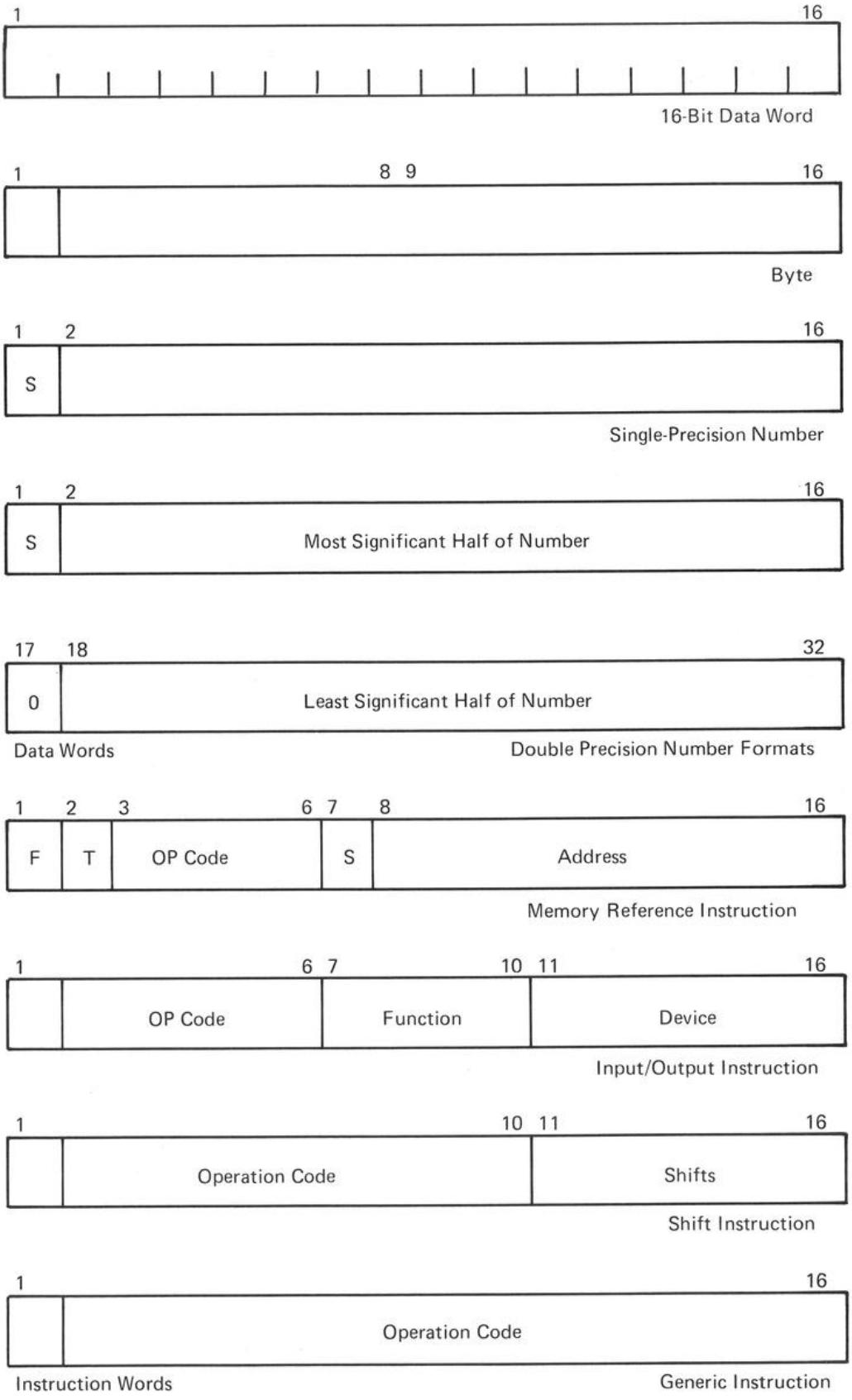


Figure 17. Series 16 Word Formats

Other Capabilities

HIGH-SPEED ARITHMETIC INSTRUCTIONS

The high-speed arithmetic instructions enhance the arithmetic capability of the processor by providing hardware implementation of multiply, divide, and normalize. They also provide a hardware double-precision load, store, add, and subtract function (see Table 11). All multiply, divide, and normalize functions are performed on data in the double-precision format.

REAL-TIME CLOCK

The real-time clock permits the programmer to keep track of real time by automatically incrementing a memory location $(00061)_8$ by one at some time increment. The increment rate can be preselected from 4 to 20 milliseconds. The increment rate is set for 16.67 milliseconds unless a different rate is selected. When memory location $(00061)_8$ overflows from $(177777)_8$ to $(00000)_8$, the real-time clock causes a program interrupt by means of the standard interrupt line. By processing a suitable interrupt subroutine, the program can keep track of time by monitoring and counting the interrupts generated by the clock. A typical application would be keeping track of elapsed time in minutes after a particular action has been initiated by the computer. Overflow from $(177777)_8$ to $(000000)_8$ does not inhibit incrementation. Incrementation can be turned on or off under program control. The real-time clock can also be used as an elapsed-time counter.

PRIORITY INTERRUPT, MEMORY INCREMENT

In addition to the standard priority interrupt system to which all peripherals and certain internal options are connected, external interrupts are available in groups of four up to a total of 48 lines. Each line can be individually enabled or disabled, and has its own cell in memory containing the address of the interrupt subroutine. As a further option, groups of four interrupt lines can be converted to memory increment lines. Here the associated location in memory is incremented by one on each interrupt request.

MEMORY LOCKOUT

Hardware protection of operating programs from undebugged programs that may be in memory at the same time is provided by this option. There are two modes of operation; normal, which permits all operations; and restricted, in which certain operations are considered illegal and cause a program interrupt. This option also enhances multiprogramming by providing the capability for private base sectors.

EXTENDED ADDRESSING

Systems with 24K or 32K word memories are equipped with bank-switching logic, whereby an extend mode is included. When in this mode, the indirect address format includes 15 address bits in order to access 32K; indexing is specified in the instruction and is applied after indirect addressing.

ON-LINE INTERFACES

Interface with the real world – in real time – is the test of a true control or communications system. With the Series 16, the facilities for capturing data or controlling data transmission are fully developed. And with the Series 16 DATAPAC Line, a systems designer can specify, from a full array of compatible components, the precise interface that's needed.

The Series 16 includes the industry's widest complement of off-the-shelf interfaces for the following applications:

- Data Acquisition and Control
- Communications Systems.
- Hybrid Systems.

Data Acquisition and Control Subsystems

Two distinctly applicable product lines are available for data acquisition interface. For small to medium-size needs (up to 96 analog input; 256-512 digital input/output), the Model 8000 Subsystem features high-performance specifications and a low price. The Real Time Interface products associated with the Models 1602, 1603, and 1605 Applications are also available separately for a wide range of small to large (up to 2048 analog input; up to 4096 digital input/output) applications. Together the two products feature:

- The flexibility that designers have consistently demanded.
- The compatibility that various applications and hardware require.
- The support that only Honeywell, among minicomputer manufacturers, can deliver.

MODEL 8000

The Model 8000 Data Acquisition and Control Subsystem (DACS) is a modular real-time interface for use with Series 16 computer systems. DACS features a line of options that allow both analog and discrete input and output points to be configured. Configurations meeting the special requirements of any user are achieved modularly with a minimum of fixed hardware overhead. Honeywell's field-proven experience in data acquisition and control systems make DACS an industry leader.

A basic analog DACS interface can handle up to 48 analog inputs; a basic digital interface accepts up to 128 digital input or output points. Expanded versions of either interface handle twice that number. By substituting a single-point analog-output module for a 16-point digital module in a digital DACS interface, the user provides an analog output capability. Such capabilities are flexibility highlights of the DACS product offering.

The maximum DACS configuration with a single Series 16 central processor is 96 analog and 256 digital points, utilizing one type each of the available expansion options. A maximum of 512 points is available in a digital-only mode.

Some of the features of DACS include:

- Its modular construction permits easy field expansion.
- Single-ended or differential analog input multiplexer units.
- Choice of 10- or 13-bit analog-to-digital (A/D) converter.
- 10-bit digital-to-analog (D/A) converter.
- Input buffer amplifier or optional sample-and-hold amplifier.
- Sampling rates up to 30 kHz.
- Sequential and random addressing modes with a configuration that includes the Direct Multiplex Control.
- External interrupt capability.
- Contact closure inputs, solid-state switching outputs, and logic level input/outputs available.
- Internal ±24V, 6A power supply for external loads.

A typical DACS configuration includes a digital input/output subsystem and an analog input subsystem.

Digital I/O Subsystem – The basic digital I/O subsystem, a 4 x 3 μ-BLOC assembly, can hold up to eight modules which may consist of any mix of digital inputs (16 points/module), digital outputs (16 points/module), or analog outputs (one point/module). The digital subsystem, like the analog unit, is available in an expanded 6 x 3 μ-BLOC version which accepts up to 16 modules providing up to 256 digital I/O points. A digital subsystem may operate alone with an analog subsystem or (unlike the analog interfaces) with a second digital interface to form a digital-only subsystem with a maximum capacity of 512 I/O points.

The digital subsystem contains all logic required to interface with the standard computer I/O bus, eliminating the need for spearate adapter logic. Priority interrupt hardware is available as an option.

Analog Input Subsystem – The basic analog input subsystem, a 4 x 3 μ-BLOC assembly, holds up to six input modules, each capable of handling eight analog points. An alternate expanded version, a 6x3 μ-BLOC, permits up to 12 input modules. Either version provides space for additional options.

The basic or expanded versions will accept either single-ended or differential multiplexer (MUX) modules up to the unit's capacity. Available plug-in options include:

- 13-bit A/D converter,
- common sample and hold amplifier,
- direct multiplex control (DMC) for either random or sequential operation, and
- automatic switching DMC.

Field connections are made via cables, with eight analog input points attached to each cable-PAC plugging into the interface.

Subsystem Interface (Basic – Model 8006, Expanded – Model 8007) – Is available with a maximum capacity of 128 or 256 I/O channels. The backplane of the interface is prewired to accept the following plug-in options and modules:

Digital Inputs (Models 8031 and 8032)– Model 8031, 16 contact-closure inputs, and Model 8032, 16 logic-level inputs, are used for the computer-initiated transfer of data from the user's devices to the computer. These modules include filtering circuits for each of the 16 inputs as well as gating logic. The input cables must be made up from twisted-pair wires to prevent interference or interaction with other circuits. Cable length from the signal source should not exceed 100 feet for Model 8032 or 1000 feet for Model 8031. Ground side of the signal or contact must have the same ground potential as the interface.

13-Bit A/D Converter (Model 8011) – Can be substituted for the 10-bit A/D converter.

Sample and Hold Amplifier (Model 8013) – Can be substituted for the buffer amplifier.

DMC Subchannel (Model 8014) – Consists of μ-PACs and the necessary I/O cables. Allows either random (two subchannels) or sequential (one subchannel) DMC control.

Automatic Switching DMC (Model 8015) – Allows either random (four subchannels) or sequential (two subchannels) auto-switching DMC control.

Single-Ended Multiplexer (Models 8041 and 8043) – Will switch eight analog channels to the common buffer

amplifier. All eight channels of a module have the same ground potential connection. Model 8043 has additional noise rejection filtering.

Differential Multiplexer (Models 8042 and 8044) – Will switch eight analog channels to the common buffer amplifier with all eight channels having different ground potential connections. Model 8044 has additional noise rejection filtering.

Cable, μ-PAC to Open-End, Eight-Pair (Models 9106 and 9108) – Consists of eight-shielded, twisted-pair wires connected to a cable-PAC which plugs into the DACS interface. The pairs on the field end are tagged and left open for connections as the system requires. Model 9106 cables are 20 feet long. Model 9108 cables are custom-cut to specified lengths up to 50 feet. One cable is required for each MUX module in the subsystem.

Digital Outputs (Models 8021 and 8022) – Model 8021, 16 logic-level outputs, and Model 8022, 16 power outputs, are used for the computer-initiated transfer of signals from the computer to the output cables to drive logic loads or operate relays, lamps, etc. These modules include gating logic, 16 flip-flops, and 16 output drivers. The user must supply the power to operate external loads. Twisted-pair wiring must be used in the output cables to prevent interference of interaction with other circuits. Cable length should not exceed 100 feet for Model 8021 or 500 feet for Model 8022. Ground side of the user's system must have the same ground potential as the digital interface.

Analog Output Model (Model 8061) – Is used to produce a single analog voltage per module, specified by the binary number from the central processor. Each analog output module contains an integral storage register, reference supply, precision switches, resistor ladder network, and output amplifier.

Priority Interrupt Option (Model 8035)– Includes two separate timing chain interrupts that cover the delay range of 1 ms to 1.0 s. Also included are two additional interrupt channels that can be activated from external signals or from digital I/O options having an interrupt-generation capability.

Cable μ-PAC to Open-End, 32-Pair (Models 9105 and 9107) – Consists of 32 twisted-pair wires connected to a double-sided cable-PAC on one end which plugs into the DACS interface to provide two digital I/O modules with a signal-carrying medium between them and the field. The pairs on the field end are tagged and left open for connections as the system requires. Model 9105 cables are 20 feet long; Model 9107 cables are custom-cut to specified lengths up to 50 feet.

Cable-PAC (Model 9103) – Is a plug-in connector card which allows the user to supply his own cables for carrying field signals between the subsystem and his devices.

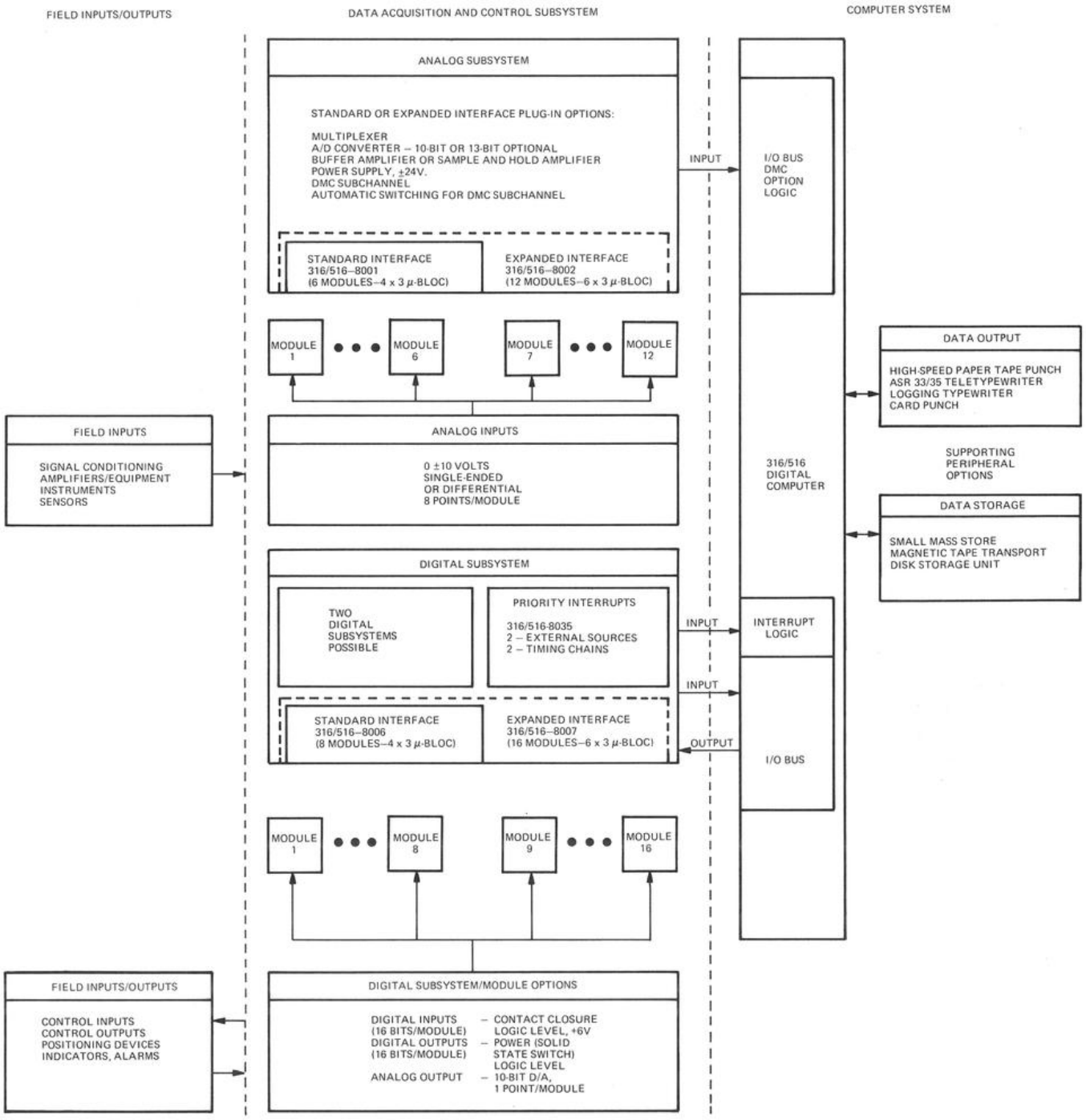


Figure 18. DACS-Model 8000

Communications Interfaces

The communications equipment offered on the Series 16 meets the specifications of any communication application. Depending on the application's size and the user's budget requirements, three types of communications interfaces are available for design:

- Data Line Controllers (DLC)
- Multi-Line Controller (MLC)
- Multi-Line Programmed Controller (MLPC)

Communication interfaces include, in turn, central processor interfaces, the controllers themselves, and line interfaces. The complexity of the communications controller varies from independent character buffers per line for DLCs to multiline programmed controllers with a single gate per receive line and a single buffer per transmit line. The different controllers may be used by themselves or in conjunction with others.

DATA-LINE CONTROLLERS

Some of the outstanding features of the data line controllers:

- Two-way simultaneous operation.
- Integrated-circuit design for reliability, compact packaging, and low cost.
- Modular construction for field upgrading and addition of options.
- Hardware parity error detection for receive characters.
- Operation in character mode on I/O bus or message mode using optional DMC subchannels.
- Less than 3% transmit distortion per character.
- Up to 45% receive distortion per character.
- Up to four controllers on one computer.
- Full range of options to meet all data speed, data codes and communications facility requirements.
- Provision for automatic dialing and automatic answering.
- Versatile software package for general applications.

Synchronous Controllers

Model 6000 provides a full-duplex (two-way simultaneous) computer interface for operation with the Bell System 201B or equivalent data set over a dedicated voice-grade data circuit. The controller handles eight bit codes at the transmission speed set by the clock in the associated data set. Odd parity checking of receive characters is standard.

Model 6020 provides computer interface for full- or half-duplex operation of Bell System 201A or 201B data sets, or Western Union 2241-B with 12275-A synchronizer data sets used in data transmission over the switched telephone network. The option provides an interface to the data set for control of signals such as Clear to Send, Data Terminal Ready, and Data Set Ready. Incoming calls from the switched network are detected and answered under computer program or manual control.

Model 6030 provides a full-or half-duplex computer interface for operation with the Bell System 301B or equivalent wide band data sets used on dedicated circuits. When used with the 301B data sets, bit speeds of 40,800 bits/second are accommodated. This option will normally be used when interfacing to the Bell System 303 data set. However, since there are several modes of operation available with the 303 data set, the user should state requirements and request technical support in selection of the proper option.

Model 6080 provides the facility for Controller Option 6020 to interface to the Bell System 801 or Western Union 12405-A Automatic Calling Unit. With the combination of options, associated data set and ACU, the computer can dial any telephone number in the switched telephone network.

Model 6091 is required when a specified synchronous controller is to be operated using six-level codes (including parity if any). The character bit pattern used for data sync is 010110. When Option 6098 is specified, the sync pattern is 110110.

Model 6092 is required when a specified synchronous controller is to be operated using seven-level codes (including parity if any). The character bit pattern used for data sync is 0010110. When Option 6098 is specified, the sync pattern is 1010110.

Model 6098 specifies even parity detection (for eight-level codes the character bit pattern used for data sync is 10010110). Standard parity error detection provided in the controller is odd.

TABLE 12. MAJOR DATA LINE CONTROLLER (DLC) MODELS

DLC Model	Prerequisite	Data Set	Speed	Terminals	Notes
Synchronous					
6000		201B	2,400 bps	Series 16 computers, Honeywell Series 200 computers. Other computers or medium speed terminals.	For medium-speed synchronous communications over dedicated two-point networks.
6020		201A	2,000 bps	Same as Model 6000	For operation over the switched telephone network.
6030		301B	40,800 bps	Same as Model 6000	
6080	6020	201A 801ACU	2,000 bps	Same as Model 6000	Provides facility for computer origination of dialed calls.
Asynchronous					
6100		103F	30-300 bps	Model 28, 33, 35, 37 teletypewriters, IBM 1050 Kleinschmidt 300 Series Other low speed terminals	For asynchronous communications over dedicated two-point networks.
6101		103F	30-300 bps	Same as Model 6100	For multipoint dedicated networks.
6110		103A	30-200 bps	Same as Model 6100	For operation over the switched telephone network.
6111		811B	45-110 bps	TWX	
6120		202D	1,800 bps	Series 16 computers, other computers, selected CRT displays	For medium-speed asynchronous communications over dedicated two-point networks.
6121		202C	1,200 bps	Same as Model 6120	For operation over the switched telephone network.
6180	6110, 6111 or 6121	801ACU	–	Any low or medium speed asynchronous terminal capable of switched network operation	Provides facility for computer origination of dialed calls.

Asynchronous Controllers

Model 6100 provides a full-duplex computer interface for operation with the Bell System 103F or equivalent data set over a two-point dedicated voice-grade circuit. The controller is designed to accommodate an eleven-unit code consisting of seven data bits, one parity bit, one start bit, and two stop bits at a transmission speed of 110 bits per second.

Model 6101 is identical to Model 6100 except that in multipoint network it is necessary for the computer to control the data set transmit/receive frequencies. This is accomplished by providing an interface to the "CY" lead of the data set.

Model 6110 provides a full-duplex or half-duplex computer interface for operation with the Bell System 103A or Western Union 1601-B switched network data set. The controller accommodates an eleven-unit code consisting of eight data bits, one start bit, and two stop bits at a transmission speed of 110 bits per second.

Model 6111 provides a half-duplex computer interface to let the computer communicate with standard Bell System teletypewriter stations on the TWX network. The Data Auxiliary Set (DAS) 811B provides the necessary interface to connect the controller to one TWX line. This model is designed for four-row (ASCII) TWX service but may be adapted for three row operation by specifying Model 6190, 6196, and 6197 specified for 45.55 bits per second. Automatic answering of incoming TWX calls is accommodated by this model. When automatic dialing of outgoing TWX calls is required, Model 6180 is also needed.

Model 6120 provides a full-duplex computer interface for operation with the Bell System 202D or equivalent data set over a dedicated voice-grade circuit. The controller accommodates an eleven-unit code consisting of eight data bits, one start bit, and two stop bits at a transmission speed of 1,800 bits per second. Half-duplex operation of the data set is accomplished by the controller generating a Request to Send signal and monitoring the data set Clear to Send signal.

Model 6121 provides a full or half-duplex computer interface for operation with the Bell System 202C, Western Union 2121-B, or Western Union 2241-B data sets over switched telephone network. The controller accommodates an eleven-unit code consisting of seven data bits, one parity bit, one start bit, and two stop bits at a transmission speed of 1,200 bits per second.

Model 6180 provides the facility for any controller designed for operation of switched network data sets to interface to the Bell System 801 Automatic Calling Unit. With this combination of options, associated data set and 801 ACU, the computer can dial any telephone number in the switched telephone network.

Model 6190 is required when a specified asynchronous controller is to be operated using five level codes (including parity if any).

Model 6191 is required when a specified asynchronous controller is to be operated using six level codes (including parity if any).

Model 6192 is required when a specified asynchronous controller is to be operated using seven level codes (including parity if any).

Model 6195 provides for asynchronous communications using one start bit and one stop bit.

Model 6196 provides for asynchronous communications using one start bit and one and one-half stop bits. This option is also used in code frames using 1.42 stop bits.

Model 6197 provides for specified data speed:

Model 6198 provides for odd parity detection. Standard parity in the asyncontroller is even.

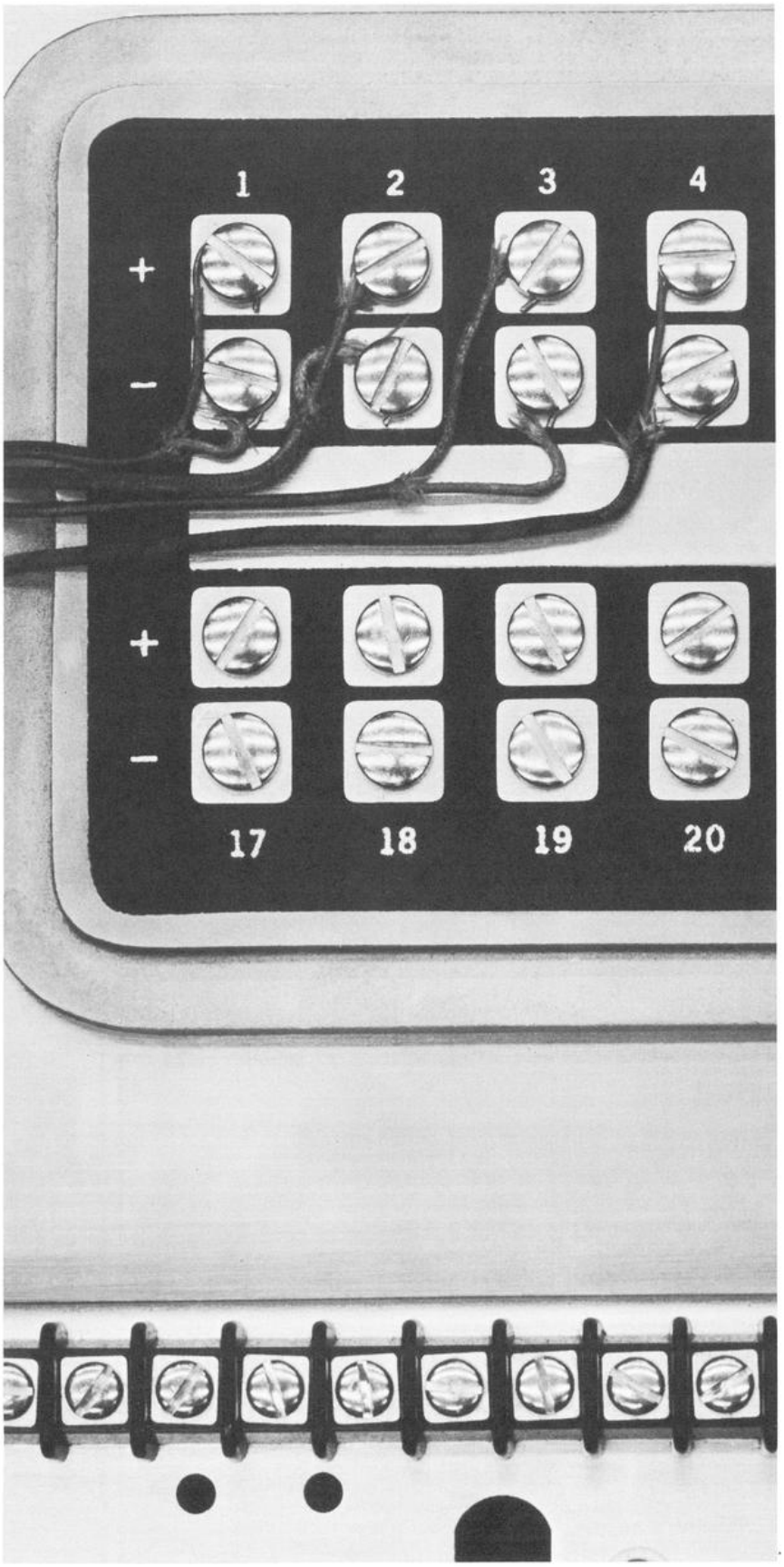
DMC Interface Options

Model 6006 (synchronous) provides the controller DMC sub-channels required for full-duplex interfacing to the Direct Multiplex Control option of the Series 16 computers. Two independent subchannels are provided, one for transmit and one for receive.

Model 6106 (asynchronous) provides the controller DMC subchannels required for full-duplex interfacing to the Direct Multiplex Control option of the Series 16 computers. Two independent subchannels are provided, one for transmit and one for receive.

6120 OR 6121 WITH 6197 (BITS PER SECOND)
4800
3600
2400
1800
1600
1200
900

ALL OTHERS WITH 6197 (BITS PER SECOND)	
4800	180
3600	150
2400	134.5
1800	110
1600	75
1200	74.2
900	66.7
800	61.1
600	56.8
400	50
300	45.5
200	30



MULTI-LINE CONTROLLERS

Systems requiring many lines become expensive and would consume a large amount of space if implemented with data line controllers. The multi-line controllers feature a lower cost per line, smaller space requirements and ease of future field expansion. This is made possible by reducing the per line loading, since character assembly and disassembly are accomplished in the control unit. The Series 16 offers several multi-line controllers capable of handling up to 128 full duplex lines. The low capacity controller handles up to 64 full duplex lines while the high capacity controllers handle up to 128 full duplex lines.

Model 670A – Each controller, Model No. 670A, contains a recirculating serial memory which provides single character buffering for each line. Also included is the timing, decoding, provision for mixed codes, and control for accommodating up to 64 full duplex line termination units. Also included in the MLC are the LTU transmit clocks. Field expansions up to 64 full duplex lines are easily accommodated. The controller interfaces to the computer via the DMCC.

Model 680A – Each controller, Model No. 680A, contains multiple synchronized serial memories which are used for bit sampling, detection, and to provide single character buffering for each line. Also included is the timing, decoding, and control for connecting up to 128 low speed full duplex line termination units. Each line interface has been simplified to a single gate for receive and a single bit buffer for transmit. All the clocking and control is located in the controller. Field expansion up to 128 full duplex lines is easily accommodated. A controller interfaces to the computer via the DMCC.

MULTI-LINE PROGRAMMED CONTROLLER

The Multi-Line Programmed Controller (MLPC) is a hardware/software system which provides an interface between a Honeywell Series 16 computer and up to 128 low-speed data communications lines.

A minimal hardware interface is used to link the data lines to the central processor. Software routines are provided to perform bit/character transformations, monitor line status, and control data flow.

The MLPC may be configured to suit the individual requirements of each system relative to different line speed, mixed dedicated/switched lines, and mixed codes. The MLPC features:

- Integrated hardware/software package.
- Interfacing for up to 128 low speed lines.
- Two way simultaneous operation over dedicated or switched lines.
- Modular construction for field upgrading and addition of lines.
- Typically less than 2% transmit distortion.
- Over 40% receive distortion.
- Data set interface including automatic answering.
- Up to 4 different line speeds.
- Choice of 3 line sampling rates to increase line handling capacity in low distortion environment.

The Multi-Line Programmed Controller (MLPC) is used for data concentrator or similar applications which interface large numbers of low-speed lines where computer loading is relatively low. By utilizing low-cost line modules and program sampling software, it is capable of controlling the flow of data between the processors and up to 128 full duplex asynchronous low-speed lines.

The MLPC provides facilities for sensing and generating the necessary control and data lines for interfacing to dedicated or switched communications facilities provided by the common carrier. The system may be configured to suit the individual requirements relative to different line speeds, mixing dedicated and switched lines, etc.

SYSTEM SPECIFICATIONS

Type of Operation	Full duplex/half duplex
Maximum Bit Rate	300 baud
Clocking	Asynchronous, supplied by software sampling function
Code	Software configurable
Interface to Computer	Input/output bus
Interface to Data Set	RS 232C
Physical Connection to Data Set	30-foot cable with EIA RS-232 compatible 25-pin connector
Maximum Number of Different Speeds	4
Maximum Number of Lines	128
Distortion	Dependent upon sampling rate which is program configurable (see text)
Receive:	
Sampling rate	7 5 3
Distortion tolerated	40.7% 38% 31.3%
Transmit	2% Typical
Program Size	1.5 sectors. An additional ½ sector per 16 line groups
Physical Characteristics:	
Size	The MLPC occupies one 2 x 3. Dedicated lines occupy one 2 x 3 for 32 lines. Switched lines occupy one 2 x 3 for 16 lines.
Power	Sufficient power is provided by the associated option bay power supply
Option	Relocatable
Environmental requirements	Same as CPU

Custom Built Interfaces

If none of the interfaces already described meets your application's needs, the Series 16 DATA-PAC Line permits custom design and construction. The designed interface can be as simple or complex as you require because DATA-PAC includes every necessary component, from simple circuit boards to core memory packages. Specifically:

- Core Memory Systems
- Digital Logic Modules
- Analog Interface Modules
- Mounting Hardware
- Power Supplies and Accessories

The DATA-PAC familty offers these benefits . . . a complete and compatible line of digital products that helps the designer produce his system – quickly, easily, and economically.

Because interfaces to other devices are expensive on a do-it-yourself basis, you will find the Honeywell DATA-PAC family a means of savings through preengineered, prepackaged, and predocumented components. They enable the newest, or most experienced designer to construct a system – practically with his bare hands.

Users benefit from continually improved, compatible DATA-PAC products. These result from an evolutionary approach to product design.

Today's DATA-PAC familty represents an enlargement in the scope of what designers have traditionally throught of as a "PAC". Honeywell has enlarged the meaning of this word to encompass modular subsystems.

Today's designers want compatible core memories, controllers, computers, analog interface modules, and other subsystems. Each family member described in this brochure is modular, expandable, flexible, compatible and reliable.

CORE MEMORY SYSTEMS

More than a decade of experience in design and production is behind Honeywell's integrated-circuit (I/C) core memory systems: the ICM-40, -42, -500, and -160. They meet the most exacting requirements – fast operating speeds and a wide range of storage capacities.

ICM-40 – First memory to use I/Cs; first of its capacity to be packaged in 5¼" high x 19" rack mounting. This compact, one-μs full cycle, magnetic core memory provides reliable operation as a high-speed random-access store. Basic ICM-40 has up to 16K words, 14 bits. ICM-40E has capacities up to 32K words, 39 bits.

ICM-42 – First buffer memory system to use I/Cs. For moderate-to-high data rates requiring buffering of 1,024 to 2,048 words of eight and 12 bits per word. Cycle times: 1.5 μs full, 900 ns half. Access time is 700 ns. Packaged for use in a drawer of digital logic circuits, or as a self-contained rack-mounted unit.

ICM-160 – Smallest (2¾" x 5" x 9"), fastest, lowest-cost memory in the DATA-PAC family. Calculated MTBF is 50,000 hours. Modular construction makes this memory easy to maintain and update. Core stack and all necessary electronic circuitry are packaged on removable printed circuit modules for ease of access. Capacity is 4K words with eight, 12, or 16 bits per wor word; cycle time is 1.6 μs.

DIGITAL LOGIC MODULES

- Over 100 types of circuit boards
- 5-, 10-, and 25-MHz switching frequencies
- High noise protection
- Low power consumption
- 100% dynamically tested
- Universally accepted NAND logic
- High packaging density

Millions of μ-PAC logic modules are now working in a variety of digital systems applications. Combining the low price, compact size, and high reliability of silicon monolithic integrated circuits, these PACs retain the straightforward logic design and flexibility of Honeywell's long established discrete-component product lines.

These static, asynchronous modules use diode transistor logic for superior noise rejection and good speed capabilities. In addition, circuit designs provide for input gate expansion (high fan-in), high fan-out, and low propagation delays.

Individual integrated-circuit assemblies, in 14-lead flat packs and dual in-line packages, are soldered to copper-etched, glass-impregnated epoxy cards.

μ-PACs are backed by more than 18 years' experience in logic modules. They are fully documented by technical bulletins, application notes, wiring and assembly notes, schematics, parts lists, and logic design aids.

ANALOG INTERFACE MODULES

- Mechanical and electrical compatibility with other DATA-PAC product lines
- Eight-, ten-, and 13-bit A/D conversion
- Eight-, and ten-bit D/A conversion
- Single-ended or differential multiplexing
- Preengineered multiplexer addressing logic
- Parallel or serial A/D output

All Analog Interface Modules (AIM) building-block units are logically, electronically, and mechanically compatible with each other, and with the other DATA-PAC family members.

This is a complete and easily applied line of preengineered and prepackaged interface modules and hardware. It allows precise configuration of analog interface systems with no additional expenditure of engineering effort for logic design or mechanical packaging.

Circuit modules, I/O compatible with all standard logic circuitry (DTL and TTL), perform analog interface functions such as A/D and D/A conversion, Sample and Hold, and Multiplexing.

Prewired, Pretested Backplanes – Single, double, or triple Omni-BLOC backplanes, in various prewired configurations, house all AIM circuit modules.

Modular Power Supply – The three regulated voltages required for the AIM line are available in a single modular power supply.

Mechanical Packaging Components – Compatible Omni-BLOC housings allow vertical or slide/tilt drawer rack mounting with natural convection or forced air cooling.

HARDWARE, POWER SUPPLIES AND ACCESSORIES

Flexibility of system design is assured through a carefully coordinated line of backplanes, mounting hardware, housings, power supplies, supplementary μ-PACs, and other accessories. With these units, precise configurations are possible without the need of custom mechanical interface design.

Backplanes – Both Omni-BLOC and μ-BLOC backplanes accept all combinations of μ-PACs, and provide for solderless-wrap, taper-pin, or TERMIPOINT interconnection wiring. In addition, they provide maximum flexibility, without sacrificing packaging density. The prewired dc power distribution system results in ease of assembly and maximum noise suppression. BLOCs can be direct-rack-mounted, or mounted in housings and slide/tilt drawer units.

Mounting Hardware and Housings – Housings and slide/tilt drawers accept any combination of Omni-BLOC backplanes and modular power supplies. A single tilt-drawer unit with integral cooling, can be used to mount up to 12 Omni-BLOCs (288 solderless-wrap μ-PAC connectors).

Power Supplies – Modular and rack-mounted power supplies are available for the required DATA-PAC operating voltages. Capacities range from 1.75A to 25A.

Accessories – A complete line of accessories, including extender PACs, blank PACs, indicator lamps, solderless-wrap tools, taper pin insertion tools, jumper lead sets and logic symbol sheets for all μ-PAC products, provides the requirements for field assembly and test.

PERIPHERALS

The Series 16 peripheral array meets all the requirements associated with a minicomputer system's likely applications.

Man/machine interface is a major requirement of control applications. The complete line of teletypewriter and graphic display subsystems available on both Series 16 central processors, answers the systems designer's demand for an effective means of implementing this important system factor. Other peripherals fullfill the need for effective reporting of control situations.

A major requirement of many communications applications, such as those involving inquiry and message switching, is the fast access to information that has been placed in storage. Of course, core memory provides the fastest access possible. But, when dealing with large files, core memory becomes too expensive. To fill this need for economical storage, Series 16 includes two magnetic tape subsystems and the direct access devices described below. The magnetic tape subsystems, 7- or 9-track, are available with the speed and recording densities listed in Table 13; this medium is sufficiently fast for many applications, especially those where files are accessed in some predetermined sequence. For faster access to randomly stored data, Series 16 offers a choice of direct access devices.

Disk Devices

The Type 4650 Dual Spindle Moving Head Disk Store option provides a flexible, low cost bulk storage system. It is a two-spindle, four-surface device, and each spindle has a capacity of 756K 16-bit words. The device can also be obtained with only a single spindle, because each spindle is functionally independent with its own control and recording electronics.

The spindle access mechanism consists of two arms, with a noncontact read/write head mounted on the end of each arm. Both arms move linearly and simultaneously in a horizontal direction to gain access to the recording tracks. One spindle module is capable of performing a seek operation while the other spindle is performing either a seek, read, or write operation.

A peripheral control provides the required interface between the computer and the storage device. It can operate up to two devices, with two spindles in each device, giving a total storage capacity of 3.02 million 16-bit words.

Features

- Removable spindle (disk) cartridges allow the disk store to operate with different packs as on-line system requirements change.

Type 4561 Disk Drive

TABLE 13. DIRECT ACCESS CAPABILITIES

Type 465X	4650 Single Spindle	4651 Dual Spindle	Type 470X	
Capacity			**Capacity**	
Words per track (1 record)	1890	1890	Words per track (1 record)	1890
Tracks per surface	200	200	Tracks per surface	200
Surfaces per disk pack	2	2	Surfaces per disk pack	20
Words per disk pack	756K	756K	Words per disk pack	7.5 million
Words per disk storage device	756K	1.51 million	Disk storage units, max.	8
Disk storage devices per controller, max.	1/option	2	Words per option, max.	60.0 million
Words per option	–	3.02 million		
Speed/Rates			**Speed/Rates**	
Seek Time			Seek Time	
1 track, max.	20 ms		1 track, max.	20 ms
Average seek	⟨100 ms		Average seek	⟨55 ms
200 tracks, max.	200 ms		200 tracks, max.	110 ms
Average rotational latency time	17.65 ms		Average rotational latency time	12.5 ms
Data transfer rate	55,312 wps		Data transfer rate	78,125 wps
Environment			**Environment**	
Temperature range	60 to 90°F		Temperature range	60 to 90°F
Relative Humidity	10 to 80%		Relative humidity	10 to 80%
Power			**Power**	
Voltage	115 ± 10% Vac, single phase, 60 ± 0.5 Hz		Voltage	115/208 ± 10% Vac, three-phase, five-wire. 60 ± 0.5 Hz
Power consumption	500 W, 0.85 pF		Power consumption	0.66 kW, 0.82 pF
Physical Characteristics			**Physical Characteristics**	
Cabinet dimensions	39" H x 35¾" W x 30" D		Cabinet dimensions	39½"H x 24"W x 33"D
Weight	300 lb.		Weight	550 lb.
Mounting			**Mounting**	
The Dual Spindle Moving Head Disk Store is mounted in a free-standing console that allows convenient access to all controls. The storage medium (disk pack) is a self-contained assembly that is removable for off-line storage. The interface logic (controller) is packaged in a relocatable 5 x 3 connector plane.			The disk storage units are mounted in a free-standing cabinet that also houses the disk storage drive, power supply, and cooling unit. The interface logic (DCU) is packaged in a relocatable 5 x 3 connector plane. Expansion to control eight disk storage units requires an extra 1 x 3 μ-BLOC. Communication between the DCU and the disc storage unit is effected via a standard twisted-pair cable that does not exceed 50 feet in length.	

- Direct seek is standard. This permits track-to-track head movement without returning to the base position.
- A seek-complete interrupt prevents loss of processing time during the seek operation.
- Up to four spindles can be connected to a single controller, permitting modular expansion of the system.
- Variable track format permits the storage of records of variable length.
- A write lockout capability protects the areas of bulk storage.
- Central-processor-finished indication permits on-line disk pack removal in multidrive systems.

The Type 4700 High Capacity Moving Head Disk Store option provides a combination of high storage capacity and low access time. High capacity is achieved through the use of disk packs containing a total of 20 recording surfaces. This capacity can be expanded even further by interchanging disk packs in a manner similar to changing magnetic tape. Low access time results from the use of 20 read/write heads mounted on a movable access arm that is positioned under program control.

Type 4700 Disk Store

The basic disk control unit (DCU) can be connected to as many as four disk storage units. The DCU can be expanded to handle a total of eight disk storage units.

Total storage capacity with a full complement of eight disk storage units is 60.0 million 16-bit words. The Type 4722 disk pack stores 7.5 million 16-bit words. Each pack is enclosed in a plastic cover which protects the pack during shelf storage and houses the mechanism for removing and installing the pack in this option.

Features

- Removable disk packs allow the disk store to operate with different packs as on line system requirements change.
- Direct seek is standard. This permits track-to-track head movement without returning to the base position.
- A single seek command positions the read/write heads to operate with up to 36K words stored on 20 tracks.
- A seek-complete interrupt prevents loss of processing time during the seek operation.
- Program-controlled tests for data transfer or transfer setup errors are available.
- Up to eight disk storage units can be connected, permitting modular expansion of the system.
- Variable track formats permit the storage of records of variable length.
- A write lockout capability protects the areas of bulk storage.
- Central-processor-finished indication permits on-line disc pack removal in multidrive systems.

Magnetic Tape Units

7-TRACK TAPE UNITS

The Series 16 Magnetic Tape systems consist of a full range of magnetic tape transports (MTT's) and associated controllers which allow operation at speeds of 26, 36, or 80 ips at standard recording densities of 200, 556, or 800 bpi. The eight tape drives offered are fully software-compatible with each other as well as IBM-code-compatible in their two-character-per-word mode of operation.

Each system can be connected to the computer through a Direct Multiplex Control (DMC) option or (except Types 4140-4142) through the input/output (I/O) bus. Types 4130-4132 and 4140-4142 can also operate with the faster Direct Memory Access (DMA).

Two tape control units (TCU's), each controlling up to four tape transports, can be included in a single Series 16 computer system.

Features

- Nonrecoverable error rate will not exceed 1 in 10^9 bits.
- Data rates up to 64,000 characters per second.
- Parity for each character written on tape.
- Multiple transport rewind.
- Program selectable BCD or binary word word formats.
- Compatible with all systems which satisfy IBM requirement for NRZI recording.
- Format and amplitudes compatible with IBM 729 series tape transports.

TABLE 14. MAGNETIC TAPE CAPABILITIES

7-Track Tape Subsystems

Program Timing

Transfer rate – The character transfer period in microseconds is as follows:

	800 bpi	556 bpi	200 bpi
80 ips	15.6	22.5	62.5
36 ips	34.7	50	139
26 ips	48.0	69.2	192.3

Data requests will occur at a nominal period of two or three times the values in the table for two- or three-character-per-word modes.

Start-Stop Timing – Where T-start is the period between the time of a "write one record" OCP and the time that writing begins, and T-stop is the period between the time of arrival from the central processor of the last word to be written and the time when the next OCP may be issued.

	T-Start (ms)	T-Stop (ms)	Rewind Time (minutes) (for full reel of tape)
80 ips	6.1	3.5	2.0
36 ips	13.6	7.7	4.5
26 ips	18.2	15.0	6.2

Power

120 Vac ±5%, single-phase, three-wire; 60 Hz ± 1%

Power Consumption

Models 4130-4132 and 4140-4142	2100 W
Model 4021 (less TCU)	750 W
Model 4022	480 W

Heat Gain

Models 4021, 4130-4132, and 4140-4142	2000 Btu/hr
Model 4022	1300 Btu/hr

Physical Characteristics

	Models 4130-4132 and 4140-4142	Model 4021	Model 4022
Height	60-¼ in.	42 in.	42 in.
Width	27 in.	31 in.	31 in.
Depth	28-¼ in.	18-½ in.	18-½ in.
Weight	900 lb	400 lb	300 lb

Mounting

The magnetic tape option is in two parts: the transport and the controller. The transport is housed in a free-standing cabinet, the dimensions of which appear above. It is connected by a 50-foot cable to its controller. The controller is a μ-BLOC connector assembly (4 x 3 for Types 4021 and 4022 and 5 x 3 for Models 4130-4132 and 4140-4142) which can be installed in an option drawer, main frame tilt-out cabinet, or I/O tilt-out cabinet as space permits.

Tape

Honeywell or IBM certified, heavy duty Mylar (1-½ mils), ½ in. wide magnetic tape on 10-½ in. reels, with IBM compatible hub (file protect ring for Models 4130-4132 and 4140-4142).

9-Track Tape Subsystems

Character Transfer Rates (bytes/second)

Type	800 bpi (density)
4150/4153	28.8K
4154/4155	64.0K

Operational Characteristics

	Model 4150/4153	Model 4154/4155
Tape speed (forward and reverse) (ips)	36	80
Start time (to within ±5% of nominal speed) (ms)	6.0	6.0
Stop time (maximum bidirectional) (ms)	6.0	6.0
Rewind time (min) and speed (ips) (full reel)	4.5, 108	2.0, 240

Power

120 Vac ± 6V, single-phase, three-wire, 60 Hz ± 0.6 Hz

Power Consumption

2.1 kVA

Heat Dissipation

2000 Btu/hr

Environmental

Temperature: 60 to 90°F
Relative Humidity: 30 to 80%

Physical Characteristics

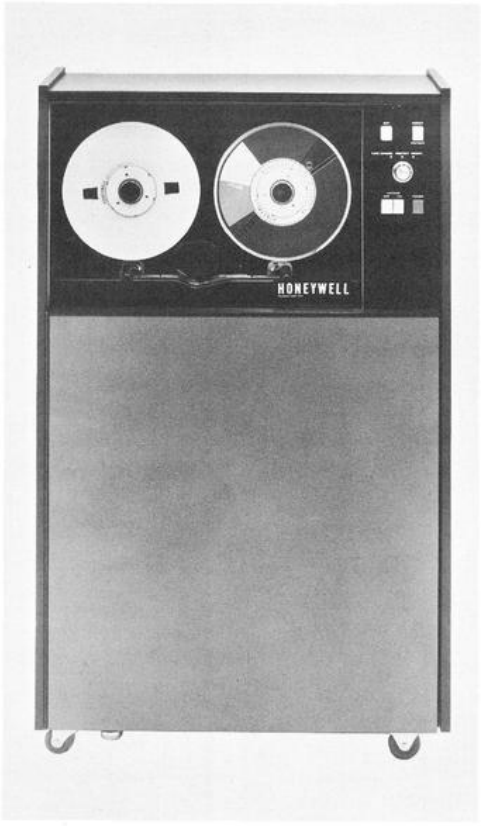
Size: 60¼" H, 27" W, 38" D
Weight: 900 lb

Mounting

The magnetic tape transports are self-contained and mounted in a free-standing cabinet containing all the necessary read/write electronics, power supplies, cooling units, tape deck, pneumatic equipment and operator controls. Interface control logic is mounted in the computer option drawer or expansion cabinet. Type 4150 and 4154 MTCUs are housed in a relocatable 6 x 3 μ-BLOC option module. Communication between the option and the control logic is effected via cable not to exceed 50 feet.

Tape

Honeywell- or IBM-certified, heavy-duty Mylar (1.5 mils), half-inch magnetic tape on 10.5" reels with IBM-compatible hub.



7-Track Tape Unit

9-TRACK TAPE UNIT

The Series 16 Nine-Track Magnetic Tape options are magnetic tape transports (MTT's) and associated controllers which allow operation at speeds of 36 or 80 ips at a standard recording density of 800 bpi. The two tape drives offered are software-compatible with each other as well as IBM-code-compatible. Data is recorded using the NRZI method.

These transports employ a motion-control concept, pioneered by Honeywell, which utilizes dual contra-rotating vacuum capstans. Also, vacuum methods are used for uniform tape-tension control and tape braking. These concepts are proven industry standards; more than 11,000 Honeywell tape transports of this type are now in use.

Types 4150 and 4153 (36 ips), and 316/516-4154 and 4155 (80 ips) are higher-performance offerings, well suited to small computer applications requiring multiple drives. See photograph for physical characteristics.

Two magnetic tape control units (MTCUs), each controlling up to four tape transports, can be included in a single Series 16 computer system.

9-Track Tape Unit

Features

- Wide range of data-transfer rates available.
- Excellent reliability.
- Compatible with all systems which satisfy IBM requirements for NRZI recording.
- High performance, with excellent start/stop times, smooth accelerations and decelerations.
- Manually selected capability for protecting hardware data.
- Write-enable ring allows writing on tape.
- Read-after-write error checking, lateral and longitudinal parity, cyclic redundancy character.
- Priority interrupt operation for real-time environment usage.

LINE PRINTERS

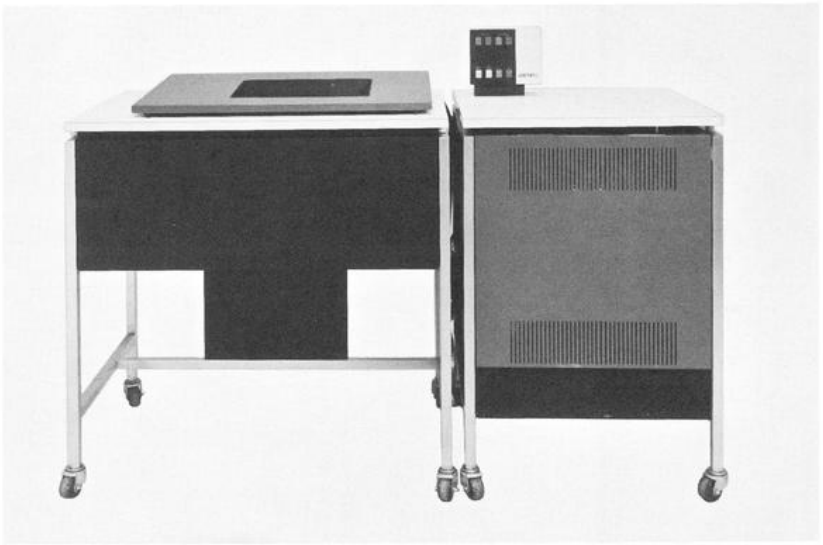
Types 5520 through 5525 Line Printers are fully buffered printed-copy, output devices. The printers operate at speeds of 300, 650, or 950 lines per minute with line widths to 120 or 132 characters. Each model is capable of operating in the I/O bus, direct multiplex control (DMC), or direct memory access (DMA) mode. They provide a wide selection of high performance/cost data printout devices, designed to satisfy a wide range of individual requirements.

Each option consists of a free-standing printing device and a controller (containing the computer interface logic and buffer memory) which is housed in the computer mainframe or expansion drawer. The buffer memory element is a solid-state, sequential access, random-time unit utilizing metal-oxide-semiconductor field-effect transistors (MOSFET) it is mounted on conventional μ-PAC modules in the controller.

Two types of printers are available: the 300 lpm units (Types 5520 and 5521) and the 650 or 950 lpm units (Types 5522-5525).

Features

- High performance – 300, 650, or 950 lines per minute.
- Rotating print drum contains full set of 63 symbols (plus blank) to be printed in any of the 120 or 132 positions in a line.
- Swingaway print drum facilitates form and ribbon loading.
- Hardened steel print drum eliminates type wear and ensures sharp, clear printing.
- Integral maintenance panels and off-line test modes ease check-out and servicing operations.
- Position servo paper feed system increases reliability by eliminating the common clutch system which is prone to wear.
- Controller utilizes 120 or 132-character MOSFET buffer memory for data transfer.
- Proven reliability with complete solid-state control circuitry.



Types 5522-5525 Printers

Types 5520, 5521 Printers

Display Devices

TYPE 7210

The Type 7210 Alphanumeric Display Terminal is a manually controlled input/output device permitting communications between an operator and a central processor. Up to 16 lines of 64 alphanumeric characters – a maximum of 1024 characters – can be presented on the 5" x 10" CRT screen. A keyboard is used to enter data or compose a retrieval request on the screen where the message may be visually verified or corrected. When prepared, the message is transmitted to the central processor via direct coupler or data modem.

Data outputs from the central processor are also displayed on the screen. Since the processor has the highest system priority, messages from it will override the keyboard mode.

The terminal contains a memory big enough to store locally generated data, received data, control data for the movable cursor, and protected field data. A movable cursor is provided to indicate the location of the next keyboard- or processor-generated character. The tabular function (TAB key) feature of the keyboard facilitates forms management. Depressing the TAB key moves the cursor to the position previously entered into the memory by the TAB SET key. The cursor controls and tab functions are linked to the protected data features of the terminal.

The central processor can be programmed to send a protected "form" to the display. The operator can enter data into the unprotected areas but cannot alter the protected characters. On transmission, only the non-protected characters are sent to the processor thereby reducing communications channel usage.

Features

- Rectangular CRT screen with P-31 phosphor for brightness
- Full display of up to 1024 characters – 64 characters in 16 lines – on a 5" x 10" screen
- Computer-program-generated formats protected against alteration
- Characters generated by digital closed-stroke technique
- Display characters continually refreshed, at line frequency, from terminal core memory
- Entire display or selected portions of data transmitted to central processor
- Movable cursor for indicating position of next character to be displayed
- Full set of 64 ASCII alphanumeric and symbol characters
- Special function, editing, and display control keys on keyboard
- Optional I/O interface for auxiliary devices such as badge reader or hard-copy printout units

TYPE 7205, 6, 7

The Type 7205 Storage Tube Display Option provides low-cost output capability for combined alphanumeric and graphic information. Visual displays can be computer-generated in a real-time experimental environment. Honeywell Types 7206 and 7207 (Tektronix Types 601 and 611) bi-stable CRT display tubes are interfaced with Honeywell Series 16 computers to accomplish this function.

These devices are used in R&D laboratories as an integral part of instrumentation and data acquisition packages. Visual presentation and data acquisition packages. Visual presentation of data facilitates iterative-type experiments and helps to optimize results. The devices are well-suited for displaying business charts, engineering curves, and geometric designs for review and analysis.

Features

- Bi-stable storage CRT eliminates refresh problem

Type 7210

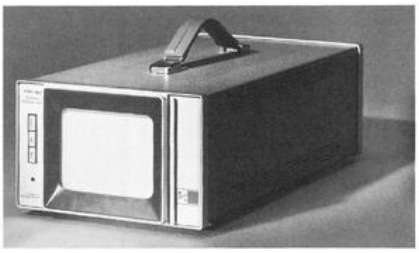
- High information density without flicker and drift
- Excellent resolution, 1024 x 1024 raster count
- Two screen sizes: 5-inch and 11-inch (diagonal)
- Convenient scaling with automatic step and deflect mode
- Hard-copy photographic equipment available
- Plotting rates up to 25K dots per second
- Alphanumeric character set, vector, and dot-plotting software

These devices are used in R&D laboratories as an integral part of instrumentation and data acquisition packages. Visual presentation of data facilitates iterative-type experiments and helps to optimize results. The devices are well-suited for displaying business charts, engineering curves, and geometric designs for review and analysis.

OPERATOR'S CONSOLE

The Honeywell Operator's Console is designed for use on data acquisition or direct digital control systems using a Model 516 computer. The console consists of (1) one of two display devices which provide a visual indication of requested information via rear-projection-type readouts, and (2) one of two keyboard devices through which a trained operator can request displays and enter information.

The Type 8182 display has 11 readouts; Type 8184, 18 readouts. The Type 8191 keyboard has 24 keys; Type 8194, 54 keys.



Type 7205, 6, 7

The basic display and keyboard are relay-rack or panel mountable, but table mounting cases are available. The operator's console communicates with the computer through the Real-Time Interface (RTI) modules.

Features

- Hermetically sealed switches give protection from corrosive atmospheres.
- Front access or display electronics facilitates maintenance.
- Driver programs are compatible with OLERT*.
- Keylock setting can be read as a digital input and used by application programs to control accessibility of console functions.
- Modularity of functions – keys are individually coded to allow frequently used functions to be initiated.
- Remote mounting – up to 500 feet from RTI.
- Multiple consoles – up to four per system.
- Oriented to process operation – familiar display, decimal indications.
- Modularity of housings – keyboard can be separate from display or placed next to it: either device can be rack mounted, table mounted, or panel mounted.

*OLERT is the On-line Executive for Real Time.

Card Equipment

TYPE 5121

Type 5121 Card Reader is a high-speed input device capable of a maximum rate of 800 cards per minute. Data sensing is accomplished by a photoelectric system which reads either alphanumeric or binary punched cards on a column-by-column basis. The reader is completely buffered with a 12-bit register which holds all data from one column in either data mode.

In the alphanumeric mode, the reader accepts cards punched in conventional Hollerith Code. Automatic code conversion produces a six-bit BCD representation of the data in each column (see Table 1). In binary mode, all 12 rows of a column are read as a 12-bit word.

Features

- Reads punched cards containing either alphanumeric or binary data upon programmed command.
- End-feed, column-by-column reading.
- Reads cards photoelectrically, 800 cards per minute maximum.
- Automatic hardware code conversion in alphanumeric mode, Hollerith to six-bit BCD code.
- Demand feeding simplifies program timing consideration.
- Error cards can be identified by offset stacking under program control.
- Interrupt operation provided for real-time environment.
- Full set of operator controls and indicators for major functions of the device.

sentation of the data in each column. In binary mode, all 12 rows of a column are read as a 12-bit word.

TYPE 5140

Type 5140 Card Reader/Punch permits the use of standard punched cards as input and output media for Series 16 computers. It reads or punches at speeds up to 400 cards per minute. The reader/punch is completely buffered with a 12-bit register which holds all data for one column in either mode.

Data sensing is accomplished by a photoelectric system which reads either alphanumeric or binary punched cards on a column-by-column basis. In the alphanumeric mode, the reader accepts cards punched in the conventional Hollerith code. Automatic hardware code conversion produces a six-bit

Type 5121

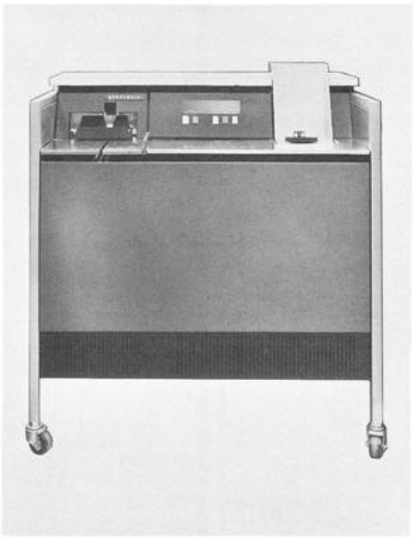
BCD representation of the data in each column. In the binary mode, all 12 rows of a column are read as a 12-bit word.

The punch mechanism operates in a binary mode to output data, column-by-column, on the cards. The punch speed, 100 to 400 cards per minute, is dependent upon the number of columns being punched.

Features

- Reading and punching capabilities in a single device.
- Data can be read and additional data punched on same card in one pass.
- Alphanumeric or binary data read on programmed command.
- Hardware code conversion from Hollerith to six-bit BCD code.
- Error cards can be identified by offset stacking under program control.
- Program controlled card punching, 100 to 400 cards per minute.
- Dual card punching mechanism increases card throughput.
- High-speed skip available to increase punching throughput.
- Interrupt operation for real-time environment.
- Error detection via interrupt, status word, and test commands.
- Full complement of operator controls and major function indicators.

Type 5140



Paper Tape Equipment

TYPE 50, PAPER TAPE READER

Type 50 Paper Tape Reader option allows the Series 16 general purpose computer to input data which has been punched on paper tape. Reading at 300 characters per second, the reader provides a high-speed input device for loading source programs, object programs, and data.

Included is a perforated tape reader and an interface control unit linking the reader to the computer. The one-inch-wide tape can be paper, mylar, or a combination of the two. Space is provided in the cabinet to house the companion Paper Tape Punch Type 52.

Features

- Maximum transfer rate of 300 characters per second.
- Separate cabinet can be located up to 50 feet from computer.
- Integrated-circuit interface logic is compatible with main frame logic.
- Priority-interrupt line permits the reader to interrupt the computer program when reader requires service, thus using less than 1% of computer operating time.
- Single-character start-stop control allows variable input block size and gapless blocks.
- Software support library.

TYPE 52 PAPER TAPE PUNCH

The Type 52 Paper Tape Punch option allows the Series 16 general purpose computer to record information at high speed on punched paper tape. The punch not only provides a means of recording data blocks for later processing, but also prepares program tapes for computer operation and maintenance. Included is an interface control unit which links the punch to the computer.

Space is provided in the electrical equipment cabinet to house the companion Paper Tape Reader Type 50.

Features

- Tape spindle capacity of 960 feet holds up to 117,000 8-bit characters.
- Maximum transfer rate is 110 characters per second.
- Uses 8-level 1-inch wide paper tape and any code combination.
- Interface logic is constructed of integrated circuits which are compatible with main frame logic.
- Priority-interrupt line permits interrupting computer program when punch requires service, thus using less than 1% of computer operating time.
- Single character start-stop control allows variable output block size and gapless blocks.
- Software support library.

TELETYPEWRITERS

Two teletypewriters are available as standard I/O options for Series 16 computer. Type 53 (ASR-33) and Type 55 (ASR-35) are similar in both operation and specifications.

The teletypewriter prints data from the computer or transmits data to the computer via the keyboard at the rate of ten characters per second. It can also read and punch paper tape at the same rate. In the local mode the unit can be used for off-line paper tape preparation, reproduction, or listing.

Features

- Versatile device can be used as reader, punch, or printer – in both on-line and off-line modes.
- Estimated service life for ASR-33 exceeds, 4,500 hours; for ASR-35 20,000 hours.
- Integrated-circuit interface logic is compatible with main frame logic.
- Priority-interrupt line enables the teletypewriter to interrupt the program only when it requires service.
- Complete software support library.

Mechanical Accessories

The Series 16 computers come with a complete assortment of mechanical support accessories. Those accessories are fully compatible and provide the systems user with the required flexibility to mount and position the system components according to his specific needs.

The accessories consist of many items, the most important of which are listed below:

- Option Cabinets Hi-Boy and Lo-Boy configuration
- Tilt Out drawer assemblies
- Mounting blocks of DATA-PACs
- Equipment tables
- I/O Cables
- Cable PACs
- Power Supplies

Terminals

Terminals, and the variety of terminals available, are the key to the success of many Series 16 applications. Interfacing the various communication services, and indirectly the communications interfaces described above, the terminals available to the Series 16 (see Figure 19 and Table 15) enable great flexibility and subsequent economy in the design of a solution.

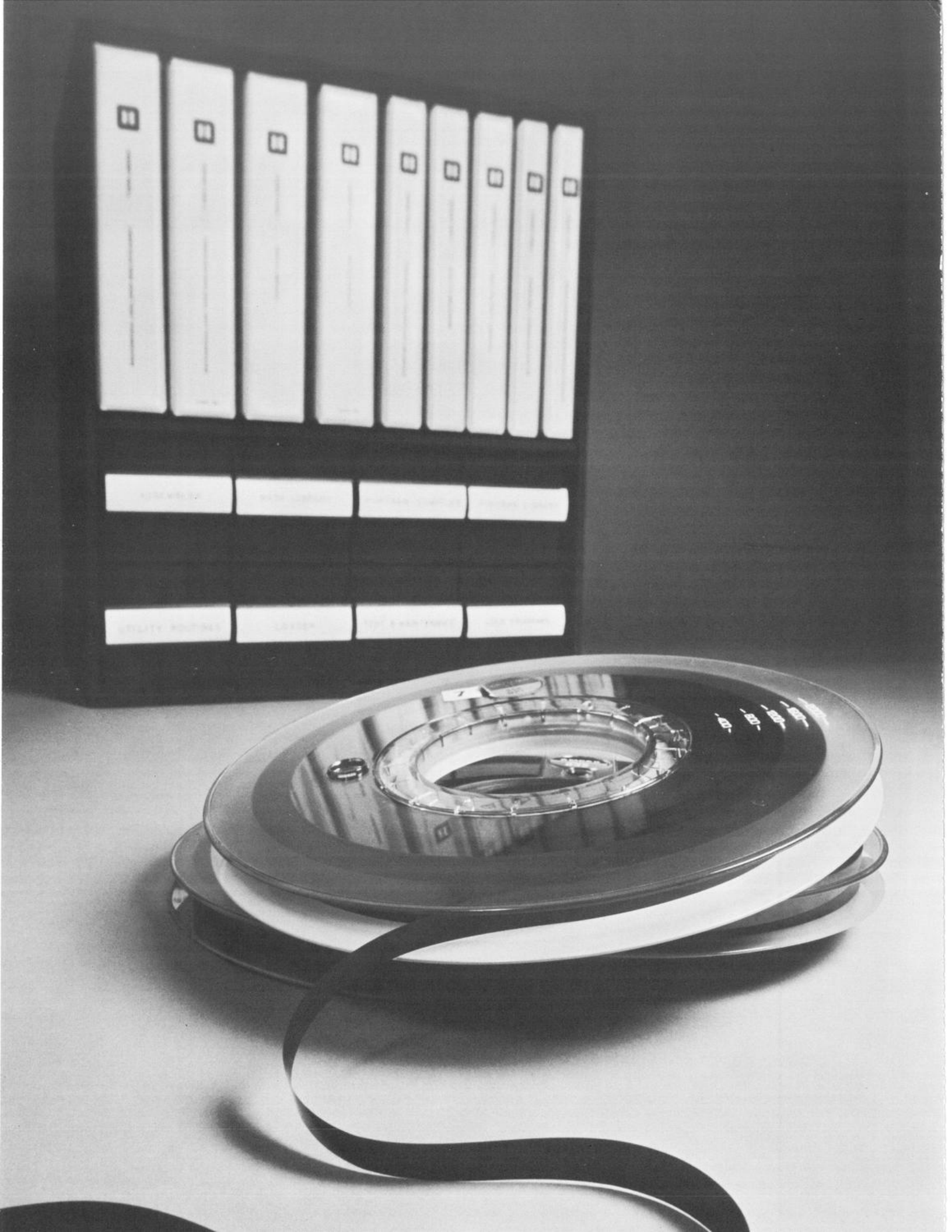
REMOTE TERMINALS, DEVICES AND/OR EQUIPMENT	SERVICE AND COMMUNICATION LINES	COMMUNICATION INTERFACE	DIGITAL PROCESSING EQUIPMENT (PROGRAMMABLE COMPUTER)
TELEPRINTER SYSTEMS Keyboards Page Printers Tape Printers Paper Tape Readers and Punches (Perforators and Reperforators) Punched Card Equipment	Common Carrier Transmission Facilities Telephone Company TWX and Dataphone Switched Circuit Services Western Union Telex Telephone Company and Western Union Leased Private Teleprinter and Narrow Bands Lines Telephone Company and Western Union Broadband Switched Circuit Services Telephone Company and Western Union Leased Private Voice-Grade Lines	Series 6000/6100 - Data Line Controller Series 6250 – Programmed Multiline Controller Series 670 – Low-Capacity Multiline Controller Series 680 – High-Capacity Single Speed Multiline Controller	Series 16 IBM Series 360
DATA SETS Bell System Data Sets Series 100, 200, 300, 400, and 800 Western Union Data Sets Common Carrier Switched Circuit Services Store and Foreword Switching Systems Auto-Polling Devices and Systems			
HONEYWELL Data Stations Processors Keytype Communicators Audio Response Systems VIP Systems Banking Teller Systems Series 200 Systems	D.C. Metallic Loop 60, 75, or 100 Speed TTY		
IBM 1009 – IBM Data Transmission Unit 1013 – Card Transmission Terminal 7701 – Mag Tape Transmission Term (75-150 Char./Sec.) 7702 – Mag Tape Transmission Term (150, 250 or 300 CPS) Paper Tape Readers and Punches Printers, Mag Tape Inquiry-Response	Voice Grade Radio Link Microwave Link		

Figure 19. Series 16 Communications Capabilities

TABLE 15. SELECTED TERMINALS USED WITH DATA LINE CONTROLLERS

Terminal	Speed	Data Set*	Operation	Controller Model	Options Required
Model 28 Teletypewriter	60-100 Words/Min	103F 103A	FDX FDX or HDX	6100 6100	6190, 6196, 6197 6190, 6196, 6197
Model 33/35 Teletypewriter	60-100 Words/Min	103F 103A	FDX FDX or HDX	6100 6110	None 6197 if other than 100 wpm
Model 37 Teletypewriter	150 Words/Min	103F 103A	FDX FDX or HDX	6100 6110	6195, 6197 6195, 6197
Kleinschmidt 311/321 (ASCII Models)	Up to 136 Words/Min Up to 272 Words/Min	103A 103F	FDX or HDX FDX	6110 6100	6197 6197
IBM 1050	14.8 Char/Sec	103A/F	HDX	6110	6192, 6195, 6197, 6198
Honeywell Data Station	120 Char/Sec 180 Char/Sec	202C 202D	HDX HDX	6121 6120	6197 None
Honeywell Keytape	120 Char/Sec 180 Char/Sec	202C 202D	HDX HDX	6121 6121	6122 None
Raytheon DIDS-400	120 Char/Sec 300 Char/Sec	202C 201B	HDX HDX	6121 6120	None None
Sanders 720 Data Display System	250 Char/Sec 300 Char/Sec	201A 201B	HDX FDX	6020 6000	None None
Honeywell Series 200 Computer with 281 Controls	250 Char/Sec 300 Char/Sec	201A 201B	HDX HDX	6020 6020	None None
IBM 360 with 2701 Synchronous Adapter Type 1	300 Char/Sec	201B	HDX	6020	6098

*Model numbers listed are Bell System. Equivalent Data Sets can be interfaced.



SYSTEMS SOFTWARE

Operating Programs

The operating programs to complete your computer automation requirements consist of monitors, compilers, assemblers, utilities, and test maintenance routines.

Honeywell offers two on-line monitors. One, OP-16, is ideal for dedicated systems. The other, OLERT, is more sophisticated and can operate in a general purpose multiprogramming environment. OP-16 is lean and efficient. OLERT is complete and easy to use.

Honeywell also offers a disk operating program, DOP; Fortran IV and BASIC compilers; an assembler, DAP-16; and a complete offering of utility, test and maintenance routines.

OLERT (ON-LINE EXECUTIVE FOR REAL-TIME)

OLERT is a software package which forms the basis for real-time multiprogramming on the Series 16 computers. OLERT provides scheduling, memory allocation input-output and other functions required for real-time multiprogramming operation. Programs can be written for OLERT using either the DAP-16 assembly language or Real-Time Fortran IV. Source programs can be assembled or compiled in the off-line mode or, as a background function, under the control of OLERT using the on-line versions of DAP-16 and Real-Time Fortran IV. The object programs are linked and converted to OLERT format and may be added to the system using either the on-line or off-line version of the OLERT system loader.

Features

Fortran Compatibility – the user can write all of his programs in Fortran, including real-time input/output interrupt response and time sequencing.

Memory Efficiency – Re-entrant programs need appear in core only once, for all users. They can be interrupted while being used by one program and then re-entered for use by other programs.

Multiprogramming Capability – OLERT can handle many concurrent real-time and background programs. The number is limited only by the memory capacity of the computer and peripheral devices.

Independent Multi-User Concept – The hardware protection features, the grouping of programs into trees, and OLERT management of core, disc, and I/O devices ensures that each user is fully protected from other users in the system.

On-Line Program Development – OLERT permits compilation, assembly, or loading and linking of programs and subroutines on-line. The operator can restructure the user software and can start and stop programs while other programs are being executed. On-line debugging is performed using the TRACE feature of Fortran IV.

Interrupt Handling – Higher priority interrupts are recognized and serviced during the servicing of lower priority interrupts. This insures rapid response to emergency conditions. Adaptability to changing process conditions is provided by the ability to change interrupt action during execution.

Executive Control – OLERT schedules the execution of programs by priority according to: interrupt response, time, program requests, operator requests. OLERT provides bulk and core memory management.

Input/Output – OLERT controls and executes all input and output operations. Editing is provided for all data formats (floating point, integer, ASCII etc.) used with Fortran IV I/O statements. Devices are referred to symbolically, permitting check-out of special I/O standard devices without reprogramming.

System Protection – OLERT uses the hardware protection features of Series 16 computer to prevent program errors from upsetting the entire computer system. Memory protection prevents a program error from altering the memory assigned to a protected program.

Modularity – Only those portions of OLERT which are necessary for a particular system need be included. This reduces memory requirements for those applications which do not require all the features of OLERT.

OP-16 OPERATING SYSTEM

OP-16 satisfies the needs of users for a small, efficient software package to implement a real-time data acquisition and control system.

On-line routines include a Real Time Executive (RTX) for computer resource management, a full complement of on-line peripheral device drivers, and a related complement of on-line device test routines. On-line and off-line trace routines along with an executive simulator provide program and system checkout capability.

Available manuals include the OP-16 Users' Guide (a manual which describes system operation, system configuration, and program formats) and a series of software support manuals which describe the operational characteristics and configuration parameters required by the device drivers and device/driver test routines.

Features

- Multiprogramming capability
- Priority scheduling of system programs on a demand and/or a time-interval basis
- Centralized control of peripheral input/output operations and associated interrupt processing
- Coordination of programs requiring common core areas, common subroutines, and input/output devices
- Communication between user programs
- Modular organization allowing configuration to suit unique application requirements
- A complete set of utility, support, debug, and peripheral-device test programs

DISK OPERATING PROGRAM (DOP)

Disk Operating Program (DOP), a powerful data processing tool for Honeywell Series 16 computers, provides maximum flexibility in programming while retaining simplicity in operation. With DOP, the user can easily and efficiently use the mass storage devices available with Honeywell Series 16 computers. Available in two versions, DOP and DOP-L, the Disk Operating Program enhances the capability of the DAP-16 assembler and the Fortran IV compiler to perform disk-to-disk language translations. The new program also supports the object-text loader to allow object programs as well as subroutine libraries to be disk resident or mixed among several I/O devices. DOP is able to store and retrieve core-image information on disk in either nonrelocatable programs or data.

Features

- Recognizes four forms of user-allocated files on disk.
- Provides keyboard interface.
- Allows user-program control.

Languages

FORTRAN IV

Fortran IV is a general-purpose, higher level programming language closely resembling the symbolic language of mathematics. It is procedure-oriented, problem-oriented, problem-solving.

Honeywell's Series 16 Fortran IV subsystem consists of a compiler, an input/output supervisor (IOS), drivers, a runtime I/O library, and a mathematical subroutine library. It conforms to ANSI Fortran standards and is substantially the same Fortran language as that available on all large computers. Thus it can draw on an extensive reservoir of existing routines.

When the characteristics of Fortran are exploited, the costs of programming can be significantly reduced. For example, Fortran enjoys several advantages over machine (assembly) languages. One Fortran statement requires translation into a number of machine language instructions. The Fortran code is more concise, more readily understood, more quickly mastered than assembly language, and to some extent self-documenting. Other advantages of Fortran over assembly language programming are a shorter elapsed time to job completion, a singificant reduction in debugging time, a doubling of machine productivity, the feasibility of investigating mathematical models, and a greatly simplified input/output procedure.

Features

- ASCII Fortran Specifications: all the advantages of a standardized language.
- One-Pass Compiler: greater operational efficiency than compilers requiring two or more passes, especially for paper tape programs.
- Run-Time Trace Capability: faster, easier debugging.
- Compiler Options: source program listing, object program listing, object text output, binary code output.
- I/O Device Selection: during compilation, all I/O is accomplished through the IOS, and devices can be selectable. I/O statements use symbolic device assignments.
- Check-Sum Generation in Object Program: guarantees correct program loading.
- Program-Chaining: an overlay capability to accommodate programs too large for the main memory of the computer. Allows the programmer to segment his program in such a way that only one segment is core-resident at any one time.
- Operating System Support: OLERT Fortran, OP-16 Fortran (Late 1971), BOS Fortran.
- Fortran Subroutines CALL from BASIC Programs: advantages of BASIC and Fortran are combined.
- Large Library of Mathematical Subroutines: many frequently used subroutines are available, so the programmer need not write them.

BASIC Interpreter

The Honeywell Series 16 BASIC Interpreter provides an interactive environment in which a user can compose, edit, and execute programs written in the BASIC language. He may assemble an entire program and call for its execution, or may enter statements to be executed immediately. Statements are only checked for errors when they are executed. Therefore, an error in a statement which is never executed, is never detected.

The Honeywell BASIC language is a superset of Dartmouth BASIC. Additional features include multiple statements per line, immediate execution of statements and commands, n-dimensional arrays, no restrictions on array subscript expressions, and the ability to call DAP or Fortran subroutines from BASIC programs.

DAP-16 MOD2 Assembler

The DAP-16 MOD2 Assembler simplifies programming and minimizes operator tasks by translating a symbolic program into machine language code. DAP-16 MOD2 enables symbolic programming while maintaining the characteristics, flexibility, speed, and precision of machine language programming while maintaining the characteristics, flexibility, speed, and precision of machine language programming. Operating in either a one- or two-pass mode, DAP-16 MOD2 assembles programs on the Model 316 and 516 computers for use on either Series 16 computer, and provides numerous pseudo-operations which conveniently offer programmer-defined assembly and loader controls, data definitions, and program linkages.

FEATURES

- Allows alphanumeric literals
- Prints and assigns storage for undefined symbols
- Flags illegal instructions and coding errors
- Permits compound expressions in the variable field

- Enables single- or double-precision, fixed- or floating-point constants
- Employs an input/output selector (IOS)
- Facilitates Fortran-like common storage in upper memory
- Provides direct addressing in source language of up to 32K memory
- Conditional assembly
- Map of symbols and addresses

PROGRAM CHARACTERISTICS

DAP-16 MOD2 operates in two basic modes: LOAD and DESECTORIZING. When operating in the LOAD mode, DAP-16 MOD2 operand addresses must be within the same sector as the instruction or in sector zero; otherwise an error flag is generated. The user must be aware of an operand's location in respect to sector boundaries. Programs assembled in the LOAD mode are always absolute.

When operating in the DESECTORIZING mode, DAP-16 MOD2 treats Series 16 computers as if all memory is directly addressable. The format of a memory reference instruction features a nine-bit address and a sector bit. Since memory is divided into sectors of 512 words, such an instruction references either the current sector containing it or the base sector of memory, depending on whether the sector bit is set or reset.

To access an arbitrary location, an indirect memory link is used. The address of the desired location is stored as the contents of a cell in either the current or base sector. This latter cell is then referenced by an instruction with its indirect-addressing bit set. Indirect memory links are supplied by the user. As an alternative, the DESECTORIZING facility of DAP-16 MOD2 provides the necessary links.

The user may ignore sector boundaries, thus permitting the assembler to produce instructions containing up to 15 address bits. When the instruction is processed by the loader, this address is reduced to nine bits and is checked to determine if either the current or the base sector is being referenced. If so, the sector bit is appropriately handled. If not, the loader constructs an indirect memory link in sector zero to provide access to the desired address.

DEBUG Program – A compact relocatable program which can:

- Type memory in octal
- Type corrections into memory
- Enter a breakpoint into memory and start at a specfied location
- Return to breakpoint and continue with program being debugged
- Clear memory to zero within limits
- Search memory for an address within specified limits
- Start at a location and print the contents of any or all of the following when one of several dynamic options is chosen: A register, B register, index register, and C bit.

Dump – This compact, completely modular relocatable program enables the user to obtain memory dumps in octal or mnemonic instruction format. Each function is provided in subroutine format; i.e., if the user desires only an octal memory dump, he need only carry the coding necessary to perform this function. The DUMP can communicate with any output equipment available in the system through the input/output selector program.

Symbolic Source Update Program (SSUP – Facilitates the deletion, insertion, or replacement of source program statements located on paper tape and whose output is on either paper or magnetic tape. A listing of the modified tape is optionally available as an output.

Input/Output Library – Made up of a set of subroutines for each I/O device, each I/O routine permits the user to specify the data format most convenient for his application. Any necessary code conversion is handled by the I/O routine. Complete error checking and, where possible, recovery procedures are included.

Verification and Test Programs – An extensive package of programs is provided with the Series 16, which includes routines for verifying the operation of the control unit, arithmetic unit, core memory, and the available input/output devices. These routines generate indicative information reflecting the operational status of the equipment being verified.

UTILITY PROGRAMS

More than 500 programs are available to Series 16 users, including the following programming software:

Input/Output Selector (IOS) – This program is used in conjunction with major programs supplied with the Series 16 to establish the input/output communication link with the input/output equipment. Users with varying complements of peripheral equipment are readily accommodated by the modular design of IOS.

Desectorizing Loader – This relocatable program loads memory with octal information in absolute or relocatable format. It is capable of loading the main program and subroutines called by it or by other subroutines, and completes the transfer vector linkage between the main program and external subroutines. Also included is the capability to generate special indirect address links in sector zero based on addressing information generated by the assembly program or compiler.

Subroutine Library – An extensive assortment of subroutines aid the programmer in performing mathematical operations and functions, conversions, and input-output operations.

Mathematical routines are available for single and double precision, fixed and floating point, and complex calculations.

Conversion routines are available for ASCII to fixed point, floating point, and complex; fixed point, floating point, and complex to ASCII; fixed point to floating point; floating point to fixed point; and complex to floating point.

Test and Maintenance Routines

With each Series 16 purchase, Honeywell provides complete backup services. These include a full set of customer-oriented Test and Maintenance (T&M) routines. They either identify problem areas and localize incorrect operation or verify correct operation of the central processor, its options, and peripheral devices. The routines exhibit actual operational characteristics so as to establish levels of performance and facilitate normal adjustments. They exercise recognized potential failure modes and furnish specific evidence of error-free completion for use in preventive maintenance.

Most of these routines apply to the entire range of available options (a few options need special routines if the memory is limited to 4K). None require an operating system. All are supplied as self-loading paper tapes (a permanent loader is stored in lower core) for use on ASR teletypewriters or high-speed readers.

PROGRAM CHARACTERISTICS

T&M routines test circuit logic, device functions, and equipment performance levels.

Programs verify the correct operation of logic circuits by isolating and exercising the logic functions they perform. A small program segment helps detect logic circuit failures by looping when it finds an error. During logic testing, mechanical support is minimized or excluded.

Tests on I/O devices require that all control functions and data paths be operational. Wherever possible, closed-loop tests are performed to verify correct operation automatically. Device tests identify failing control functions to the maintenance engineer and indicate failing data by bit position.

All tests operate for a specific period of time to establish a level of equipment performance. Tests for devices requiring periodic adjustment contain repetitive sections so that adjustments may be accomplished easily.

```
173 00036    0 11 00373        CAS   PIC2          TEST FOR E
174 00037    000004      Q4    OCT   4             NEVER CAN
175 00040    0 01 00055        JMP   SRT5          END OF BLO
176 00041    0 01 00026        JMP   SRT1          MORE SYMBO
177                       *
178 00042    0 04 00373 SRT3   STA   PIC2          PIC2 ← ADD
179 00043    0 10 00204        JST   LCHK
180 00044    0 01 00050        JMP   SRT2
181 00045    0 10 00127        JST   GBIT          FETCH CURR
182 00046    101040            SNZ                 SKIP IF SE
183 00047    0 01 00105        JMP   SWAP          INTERCHANG
184 00050    0 02 00373 SRT2   LDA   PIC2          TRY NEXT L
185 00051    0 07 00377        SUB   Q3
186 00052    0 11 00372        CAS   PIC1          TEST FOR C
187 00053    0 01 00042        JMP   SRT3          NO - TEST
188 00054    0 04 00373        STA   PIC2          YES - UPDA
189 00055    0 02 00373 SRT5   LDA   PIC2          DROP BLOCK
190 00056    1 04 00000        STA   T237,1
191 00057    1 13 00000        IMA   T136,1
192 00060    1 04 00000        STA   T137,1
193 00061    0 12 00000        IRS   0             INCREMENT
194 00062    0 01 00017        JMP   SRT7          GO SORT ON
195 00063    0 02 00402        LDA   MQ1           SEARCH FOR
196 00064    100000            SKP
197 00065    0 02 00000 SRT6   LDA   0             DECREMENT
198 00066    0 07 00401        SUB   Q1
199 00067    0 04 00000        STA   0             REPLACE PO
200 00070    0 07 00376        SUB   M36           TEST FOR C
201 00071    100400            SPL                 SKIP IF NO
202 00072    0 01 00216        JMP   DUMP          SORT COMPL
203 00073    1 02 00000        LDA   T136,1        FETCH BLOC
204 00074    100400            SPL                 SKIP IF UN
205 00075    0 01 00065        JMP   SRT6          SORTED - C
206 00076    1 04 00000        STA   T137,1        SET
207 00077    140500            SSM                 MARK IT AS
208 00100    1 04 00000        STA   T136,1        AND DROP I
209 00101    1 02 00000        LDA   T236,1
210 00102    1 04 00000        STA   T237,1
211 00103    0 12 00000        IRS   0             SET BIT IN
212 00104    0 01 00017        JMP   SRT7          GO SORT TH
213                       *
214 00105    0 02 00372 SWAP   LDA   PIC1          HERE TO IN
215 00106    141206            AOA
216 00107    0 04 00374        STA   PIC3
217 00110    0 02 00373        LDA   PIC2
218 00111    141206            AOA
219 00112    0 04 00375        STA   PIC4
220 00113   -0 02 00372        LDA*  PIC1          SWAP FIRST
221 00114   -0 13 00373        IMA*  PIC2
```

APPENDIX

PROGRAMMING the SERIES 16

Below, the actual program execution instructions found in the Series 16 assembly language are categorized by function:

- Data-handling.
- Arithmetic.
- Logic.
- Shift.
- Byte-handling.
- Control.
- Input/Output.

In addition to these standard instruction tions, the instructions that comprise the high-speed arithmetic capability and memory parity feature, are included in this section.

MNEMONIC

FORMAT:

1. As it would appear on coding form.
2. As it would appear in memory.

FUNCTION:

Description of the function directed by the instruction.

TIMING:

Total Model 316 and 516 execution time of instruction in microseconds.

NOTES:

Items that may be of interest to the programmer but are not obvious from the above information.

EXAMPLE:

Simple textual and illustrative description of a typical instruction execution. The instruction is given as it would appear on the coding form and as the involved central processor units would appear BEFORE and AFTER execution.

TABLE 16. SYMBOLOGY

SYMBOL	MEANING
	Initial (BEFORE) value has no effect on instruction execution; value will be changed by instruction.
	Final (AFTER) value of register will not be affected by instruction execution; value will be same as initial (BEFORE).
EA	Memory location specified by effective address
A	Primary arithmetic register
B	Secondary arithmetic register
C	Overflow indicator, C bit
a	Initial (BEFORE) contents of A
b	Initial (BEFORE) contents of B
c	Initial (BEFORE) contents of C: 0 = no overflow; 0 = overflow
m	Initial (BEFORE) contents of EA
X	Index register
x	Initial (BEFORE) contents of X
NSC	Normalize shift counter
nsc	Initial (BEFORE) contents of NSC
$\overline{n}$	Complement of n e.g., if c = 0, then $\overline{c}$ = 1
P	Program counter, also called P register
p	Contents of P
DP	Arithmetic precision indicator
n	Contents of DP
S	Sign of numerical value: 0 = positive; 1 = negative
( )	Programmer's coding option
*	Specifies indirect addressing
,1	Specifies indexed addressing
	Boundaries between central processor units; e.g., between memory and control unit
byte-n	Initial (BEFORE) contents of a half-word n = 1 (bits 1 - 8 of register) n = 2 (bits 9 - 16 of register)
	Register contents shifted to right
	Register contents shifted to left
	Bits discarded
	Boundaries between central processor and other system units; e.g., between arithmetic unit and device control.

DATA HANDLING

The Series 16 instructions set includes nine "data handling" instructions, so named because they manipulate the A, B, and X registers, the most frequently used in Series 16 programming. With these instructions, the Series 16 user can draw 16-bits of information from and to memory and the input/output lines as well as set up an indexing value for this important capability in memory addressing.

LDA

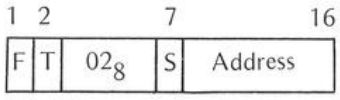
Load the A Register

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   LDA (*)

   ADDRESS, X
   Address (,1)

2. 

FUNCTION:

The control unit is directed to load the contents of the memory location, specified by the effective address, into the A register.

TIMING:

3.2 (316), 1.92 (516) microseconds.

NOTE:

If the base location, 00000_8, is addressed by an LDA instruction, the contents of the index register are loaded into the A register.

STA

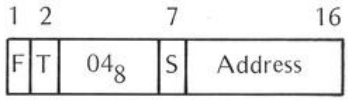
Store the A Register

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   STA (*)

   ADDRESS, X
   Address (,1)

2. 

FUNCTION:

The control unit is directed to store the contents of the A register in the memory location specified by the effective address.

TIMING:

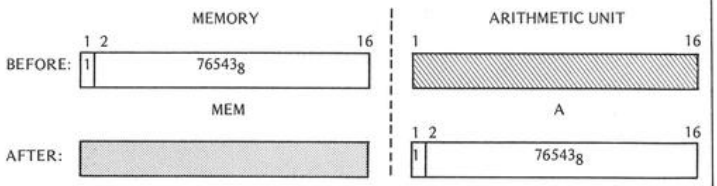
3.2 (316), 1.92 (516) microseconds.

NOTE:

If the base location, 00000_8, is addressed by the STA instruction, the contents of the A register are stored in the index register (X), as well as the base location.

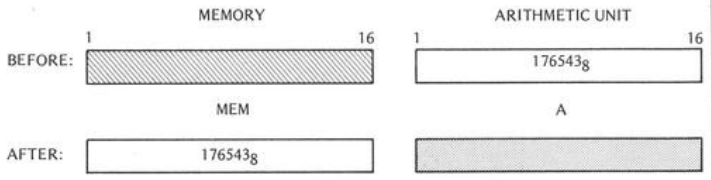
PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	LDA	MEM		

1. Assume that the value 176543_8, is stored in a location MEM.
2. The LDA instruction directs the control unit to load the contents of MEM into the A register.



PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	STA	MEM		

1. Assume that the value, 176543_8, has been loaded into the A register by an LDA instruction.
2. The STA instruction directs the arithmetic unit to store the contents of the A register in a location, MEM.



CRA

Clear the A Register

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   CRA

   ADDRESS, X
   Not Applicable

2. 1 16

FUNCTION:

The control unit is directed to replace the contents of the A register with zeros.

TIMING:

1.6 (316), 0.96 (516) microseconds.

IAB

Interchange A and B

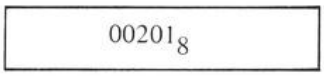
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   IAB

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to interchange the contents of the A and B arithmetic registers.

TIMING:

1.6 (316); 0.96 (516) microseconds.

IMA

Interchange Memory and A

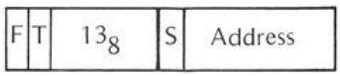
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   IMA (*)

   ADDRESS, X
   Address (,1)

2. 1 2 7 16



FUNCTION:

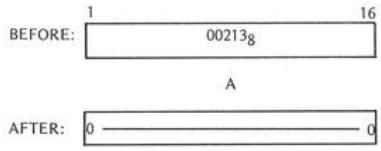
The control unit is directed to interchange the contents of the memory location specified by the effective address with the contents of the A register.

TIMING:

3.2 (316), 1.92 (516) microseconds.

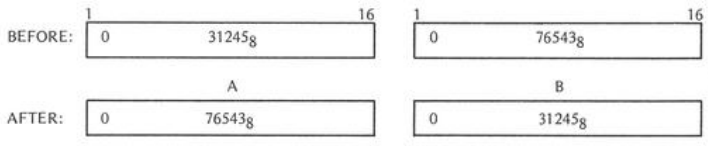
PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	CRA			

1. Assume that the value 00213_8, has been loaded into the A register by an LDA instruction.
2. The CRA instruction directs the arithmetic unit to clear the A register; that is, replace the current value with zeros. This prepares the A register for further program utilization.



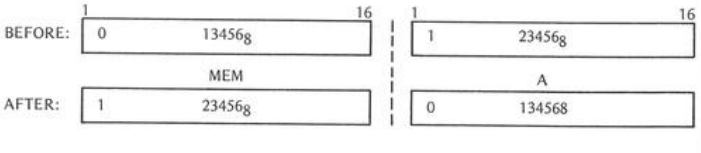
PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	IAB			

1. Assume that the contents of the A register is 031245_8, and the contents of the B register is 076543_8.
2. The IAB instruction directs the control to interchange the contents of the two registers; therefore, the A register now contains 076543_8 and the B register contains 031245_8.



PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	IMA	MEM		

1. Assume that the contents of the location MEM is 013456_8 and that the contents of A is 123456_8.
2. The IMA instruction directs the control unit to interchange the two values; therefore, MEM now contains 123456_8 and the A register is loaded with 013456_8.



INK

Input Keys

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   INK

   ADDRESS, X
   Not Applicable

2. 1 16

00043_8

FUNCTION:

The control unit is directed to store the contents of the C bit (overflow indicator) in the sign (high-order) position of A; the mode of arithmetic precision (0 = single-precision, , 1 = double-precision) is stored in the second position of A; the third through the eleventh bits contain zeros; the contents of the Normalize Shift Counter are stored in the twelfth through the sixteenth positions of A.

TIMING:

1.6 (316), 0.96 (516) microseconds

NOTES:

1. This instruction allows the programmer to store processor status during an interrupt routine; however, it should be done immediately upon entering the routine in order to be sure that the Normalize Shift Count is correct. The following example illustrates the coding of an INK instruction immediately upon entering an interrupt routine:
   This is necessary because any arithmetic or shift instruction will clear the Normalize Shift Counter.
2. C and DP are displayed on the control panel at the eighth and thirteenth lamps, respectively, if the OP push button is pressed.

OTK

Output Keys

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   OTK

   ADDRESS, X
   Not Applicable

2. 1 16

171020_8

FUNCTION:

The control unit is directed to store the contents of the sign (high-order) bit of A in the C bit; the second bit of A is stored in the DP indicator; and the twelfth through the sixteenth bits of A are stored in the Normalize Shift Counter.

TIMING:

1.6 (316), 0.96 (516) microseconds.

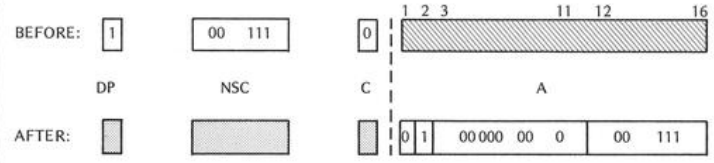
PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1 4	OPERATION 6 10	ADDRESS X 12	COMMENTS 30 72	IDENTIFICATION 73 80
S.T.R.1	S.T.A.	MEM	STORE CONTENTS OF A	
	I.N.K.		LOAD A WITH STATUS	
	S.T.A.	MEM1	STORE STATUS	

PROGRAMMER | DATE | PAGE OF

PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	INK			

1. Assume that the most recent arithmetic operation caused no overflow; that the computer is in the double-precision mode; and that the most recent normalize operation required 7 shifts.
2. The INK instruction directs the control unit to load this information in the A register.

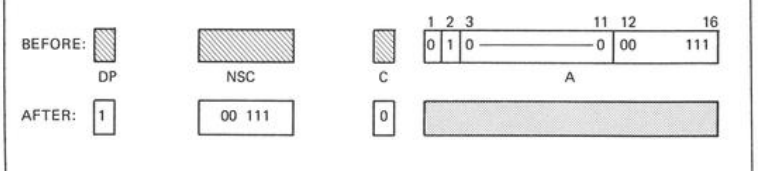


PROGRAMMER | DATE | PAGE OF

PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	OTK			

1. Assume that an interrupt routine has completed service of a given device and that an LDA instruction has retrieved processor status from memory.
2. The OTK instruction directs the control unit to store this information in its proper place in the arithmetic unit.



LDX

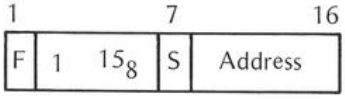
Load Index Register

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   LDX (*)

   ADDRESS, X
   Address

2. 

FUNCTION:

The control unit is directed to store the contents of the location, specified by the effective address, in the X register.

TIMING:

4.8 (316), 2.88 (516) microseconds.

NOTES:

1. Note that the operation code in this instruction is contained in bits 2 through 6; therefore, this instruction cannot be indexed. However, if indirect addressing is specified (1 bit = 1), the indirect address location can specify indexing in the usual manner.
2. Whenever an LDX instruction is executed, the value loaded into the index register (X), is also loaded into the location in memory, 00000_8.

STX

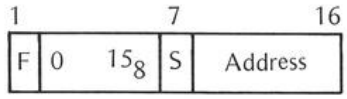
Store Index Register

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   STX (*)

   ADDRESS, X
   Address

2. 

FUNCTION:

The control unit is directed to store the contents of the X register in the memory location specified by the effective address.

TIMING:

3.2 (316), 1.92 (516) microseconds.

NOTE:

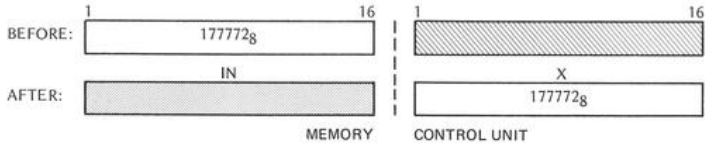
Note that the operation code is contained in bits 2 through 6; therefore, this instruction cannot be indexed. However, if indirect addressing is specified (F = 1), the indirect address location can be indexed in the usual manner.

PROGRAMMER | DATE | PAGE OF

PROGRAM | CHARGE

LOCATION (1–4)	OPERATION (6–10)	ADDRESS X (12)	COMMENTS (30–72)	IDENTIFICATION (73–80)
	LDX,	IN		

1. Assume that a location, IN, contains a value, -5, expressed as 177772_8.
2. The LDX instruction directs the control unit to load the contents of IN in the X register.

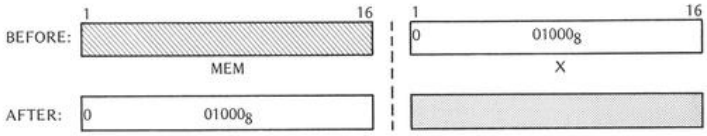


PROGRAMMER | DATE | PAGE OF

PROGRAM | CHARGE

LOCATION (1–4)	OPERATION (6–10)	ADDRESS X (12)	COMMENTS (30–72)	IDENTIFICATION (73–80)
	STX,	MEM		

1. Assume that the value 01000_8 is loaded into the X register.
2. The STX instruction directs the control unit to store the value 01000_8 in the location MEM.



ARITHMETIC

The basic Series 16 arithmetic instructions enable simple computation with overflow.

ADD

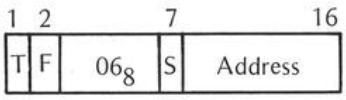
Add

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ADD (*)

   ADDRESS, X
   Address (,1)

2. 

FUNCTION:

The control unit is directed to add, algebraically, the contents of the A register to the contents of the memory location specified by the effective address. The operation takes place in the adder and the result is stored in the A register.

TIMING:

3.2 (316), 1.92 (516) microseconds.

NOTES:

1. If overflow occurs, (overflow can occur only when adding quantities with the same sign) the overflow indicator (C bit) is set.
2. Negative numbers are expressed in twos-complement notation.

TCA

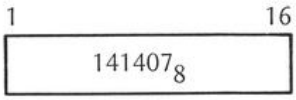
Twos-Complement A

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   TCA

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is directed to complement the value of the A register and to increase the result by the binary value 1, in order to derive the twos complement of the original contents of the A register.

TIMING:

2.4 (316), 1.44 (516) microseconds.

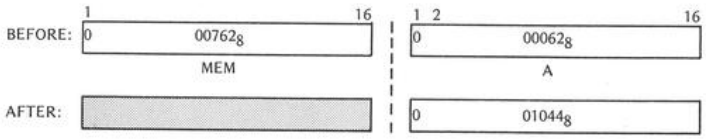
NOTE:

The twos complement of any number is obtained by deriving the ones complement (change all 0's to 1's and all 1's to 0's) and increasing the result by one (1).

PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12 30	72	73 80
	ADD	MEM		

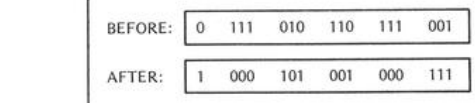
1. Assume that the decimal value 50 is stored in the A register, expressed as 00062_8 and that the value, 00762_8, is stored in location MEM.
2. The ADD instruction directs the control unit to route the two values to the adder where they are algebraically combined; then, they are stored in the A register.



PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12 30	72	73 80
	TCA			

1. Assume that the value $0\ 111\ 010\ 110\ 111\ 001_2$ is loaded in the A register.
2. The TCA instruction directs the control unit to derive its twos complement and store the result in the A register.



ACA

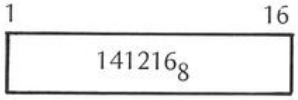
Add Overflow Bit To A

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ACA

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is directed to add, algebraically, the contents of the overflow bit (C bit) to the low-order bit of the A register with the result being stored in the A register.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTES:

1. This instruction can be used to test overflow status after an arithmetic operation.
2. This instruction can be used in double-precision program subroutines to carry overflow from the least significant word to the low-order bit of the most significant word.

AOA

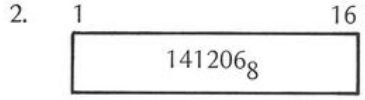
Add One to A

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   AOA

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is directed to add, algebraically, the binary value 1 to the least significant bit of the A register. The result is stored in the A register. An overflow causes the setting of the overflow indicator (C bit).

TIMING:

1.6 (316), 0.96 (516) microseconds.

SUB

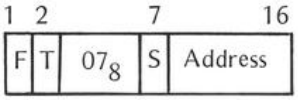
Subtract

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SUB (*)

   ADDRESS, X
   Address (,1)

2. 

FUNCTION:

The control unit is directed to subtract the contents of the memory location specified by the effective address from the contents of the A register; the result is stored in the A register.

TIMING:

3.2 (316), 1.92 (516) microseconds.

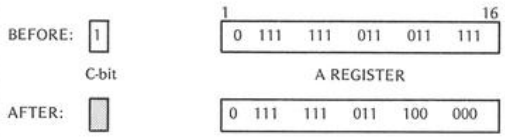
NOTES:

1. If the signs of the minuend and subtrahend are the same, the C bit is inhibited.
2. If the signs of the minuend and the subtrahend are different, overflow is allowed.

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12 30	72	73 80
	ACA			

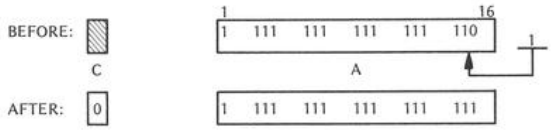
1. Assume that the program has instructed the addition of the low-order word of a multiprecision constant; this addition has caused an overflow (C bit).
2. The program then stores the low-order result and loads the A register with the high-order addend.
3. The ACA instruction directs the control unit to add the overflow bit to the low-order bit of the A register.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12 30	72	73 80
	AOA			

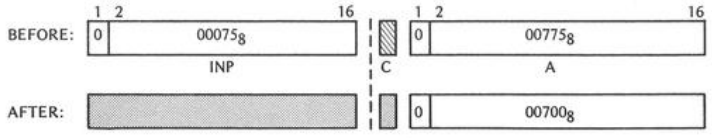
1. Assume that the value 177776_8 is loaded into the A register.
2. The AOA instruction directs the control unit to add one (1) to the A register; therefore, the A register now contains the value 177777_8.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12 30	72	73 80
	SUB	INP		

1. Assume that the value 00775_8 is stored in the A register and that the value 000075_8 is stored in a location INP.
2. The SUB instruction directs the control unit to subtract INP from A to derive the value 000700_8 in A.



LOGIC

The logical operations, key to any computer program's decision-making capability, are enabled by this set of seven Series 16 instructions. Both key operations, ANDing and ORing, are included.

ANA

Logical And to A

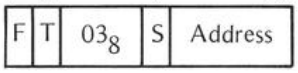
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ANA (*)

   ADDRESS, X
   Address (,1)

2.



FUNCTION:

The control unit is directed to perform a Logical AND operation on the contents of the memory location specified by the effective address and the contents of the A register, on a bit-by-bit basis, according to the AND truth table:

Bit value in "a"	0	1	0	1
Bit value in "m"	0	0	1	1
"a" AND "m"	0	0	0	1

The result of this logical comparison is stored in the A register.

TIMING:

3.2 (316), 1.92 (516) microseconds

CSA

Copy Sign and Set Sign Plus

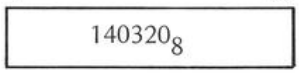
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   CSA

   ADDRESS, X
   Not Applicable

2.



FUNCTION:

The control unit is directed to store the sign (high-order bit) of the A register in the overflow indicator (C bit) and to place zero (plus) in the high-order position of A.

TIMING:

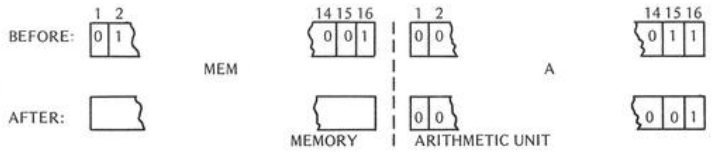
1.6 (316), 0.96 (516) microseconds.

PROGRAMMER | DATE | PAGE OF

PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	ANA	MEM		

1. Assume that you could see only the two low-order and two high-order bits of the A register and a location, MEM.
2. The ANA instruction directs the control unit to perform a logical AND operation on these two values. The result is stored in A.

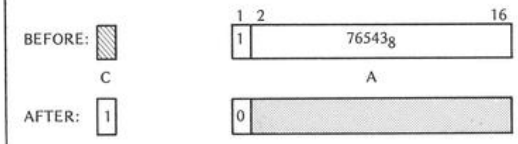


PROGRAMMER | DATE | PAGE OF

PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	CSA			

1. Assume that the value 176543_8 is loaded into the A register.
2. The CSA instruction directs the control unit to store the sign (high-order bit) of A in C and to set C plus (plus a zero in C).



CHS

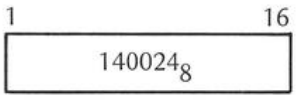
Complement the Sign of A

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   CHS

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is directed to replace the high-order bit of A with its complement.

TIMING:

1.6 (316), 0.96 (516) microseconds.

CMA

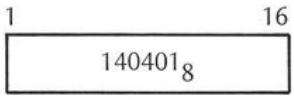
Complement the A Register

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   CMA

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is directed to replace the contents of the A register with its ones complement.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTES:

1. This operation is termed ones complementing, in contrast to the twos-complementing operation referred to earlier.
2. To form the ones complement, replace all zeros with ones, and all ones with zeros.

ERA

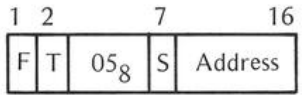
Exclusive Or to A

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ERA (*)

   ADDRESS, X
   Address (,1)

2. 

FUNCTION:

The control unit is directed to perform the Exclusive OR operation on the contents of the memory location specified by the effective address and the contents of the A register, on a bit-by-bit basis, according to the Exclusive OR truth table.

Bit Value in "a"	0	0	1	1
Bit Value in "m"	0	1	0	1
"a" ExOR "m"	0	1	1	0

The result of the logical comparison is stored in A.

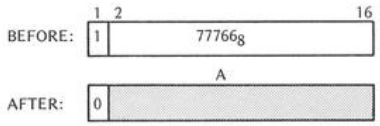
TIMING:

3.2 (316), 1.92 (516) microseconds.

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	CHS			

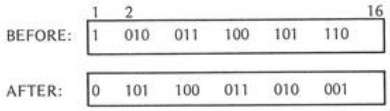
1. Assume that the value 177766_8 is loaded into the A register.
2. The CHS instruction directs the control unit to replace the sign of A (1) with its complement (0).



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	CMA			

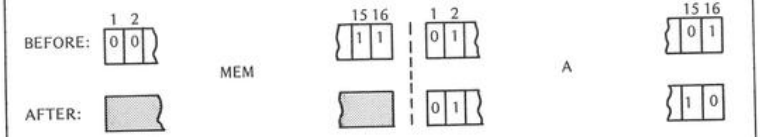
1. Assume that the value 123456_8 is loaded into the A register.
2. The CMA instruction directs the control unit to perform a ones-complement operation on the contents of the A register; therefore, the A register now contains the value, 054321_8.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	ERA	MEM		

1. Assume that the two low-order and high-order bits of the A register and a location, MEM, are visible.
2. The ERA instruction directs the control unit to perform a logical Exclusive OR operation on these two values. The result is stored in A.



SSM

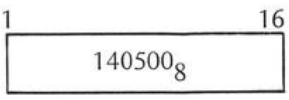
Set Sign Minus

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SSM

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is directed to replace the high-order bit of the A register with a one.

TIMING:

1.6 (316), 0.96 (516) microseconds.

SSP

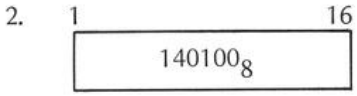
Set Sign Plus

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SSP

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

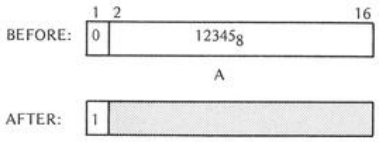
The control unit is directed to replace the high-order bit of the A register with a zero.

TIMING:

1.6 (316), 0.96 (516) microseconds.

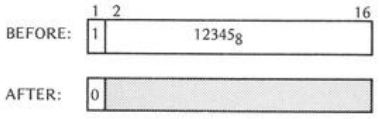
PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	SSM			

1. Assume that the value 012345_8 is loaded into the A register.
2. The SSM instruction directs the control unit to place a one in the high-order position of A; therefore, A now contains the value of 112345_8.



PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	SSP			

1. Assume that the value 112345_8 is loaded into the A register.
2. The SSP instruction directs the control unit to replace the sign of A (1) with a zero; therefore, A now contains the value 012345_8.



SHIFT

A series of twelve "shift" instructions enable data manipulation at the bit level and ensure an even more economical utilization of Series 16 memory in applications where parameters can be reduced to simple yesses and nos.

ALR

Logical Left Rotate

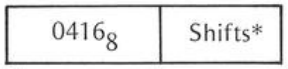
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ALR

   ADDRESS, X
   Number of Shifts

2.



*In twos-complement notation

FUNCTION:

The control unit is directed to rotate the contents of the A register to the left as many positions as specified. The high-order bit is shifted to the low-order bit and the C bit. The C bit retains the value of the last bit shifted into the low-order bit of A.

TIMING:

1.6 (316), 0.96 (516) plus 0.8 N microseconds. N = number of shifts.

NOTE:

The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.

ARR

Logical Right Rotate

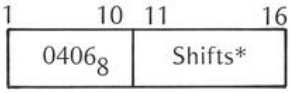
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ARR

   ADDRESS, X
   Number of Shifts

2.



*In twos-complement notation

FUNCTION:

The control unit is directed to rotate the contents of the A register to the right as many places as specified. The low-order bit is shifted to the high-order bit, and the C bit. The C bit retains the value of the last bit shifted out of the low-order bit of A.

TIMING:

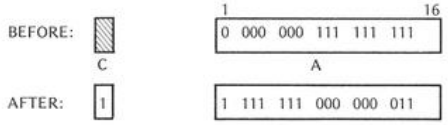
1.6 (316), 0.96 (526) plus 0.8 N microseconds. N = number of shifts.

NOTE:

The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.

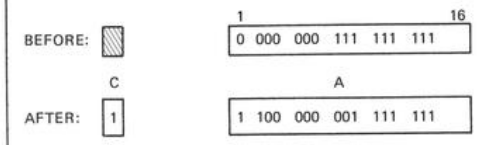
PROGRAMMER			DATE	PAGE OF
PROGRAM				CHARGE
LOCATION 1 4	OPERATION 6 10	ADDRESS X 12	COMMENTS 30 72	IDENTIFICATION 73 80
	ALR	9		

1. Assume that the value 0 000 000 111 111 111_8 is loaded into the A register.
2. The ALR instruction directs the control unit to rotate the contents of the A register 9 places to the left. The C bit retains the value of 1, the last bit shifted into the low-order of A.



PROGRAMMER			DATE	PAGE OF
PROGRAM				CHARGE
LOCATION 1 4	OPERATION 6 10	ADDRESS X 12	COMMENTS 30 72	IDENTIFICATION 73 80
	ARR	2		

1. Assume that the value 0 000 000 111 111 111_8 is loaded into the A register.
2. The ARR instruction directs the control unit to rotate the contents of the A register 2 places to the right. The C bit retains the value of 1, the last bit shifted from the low-order bit.



ALS

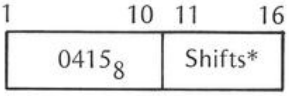
Arightmetic Left Shift

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ALS

   ADDRESS, X
   Number of Shifts

2. 

*In twos-complement notation

FUNCTION:

The control unit is directed to shift the contents of the A register to the left as many places as specified. If this shift causes the sign (high-order) bit to change, the C bit is set (C = 1). If the sign does not change, the C bit remains 0.

TIMING:

1 6 (316), 0.96 (516) plus 0.8 N microseconds. N = number of shifts.

NOTE:

The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.

ARS

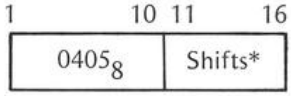
Arithmetic Right Shift

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ARS

   ADDRESS, X
   Number of Shifts

2. 

*In twos-complement notation

FUNCTION:

The control unit is directed to shift the contents of the A register to the right as many places as specified. The sign value is shifted into the vacated high-order position. The low-order bit is shifted to C; then C is discarded. C retains the value of the last bit shifted when all shifts are completed.

TIMING:

1.6 (316), 0.96 (516) plus 0.8 N microseconds. N = number of shifts.

NOTES:

The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.

LGL

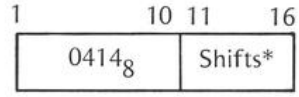
Logical Left Shift

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   LGL

   ADDRESS, X
   Number of Shifts

2. 

*In twos-complement notation

FUNCTION:

The control unit is directed to shift the contents of the A register to the left as many places as specified. Zeros replace the vacated low-order bits of A. The C bit retains the value of the last bit shifted from the high-order bit of A.

TIMING:

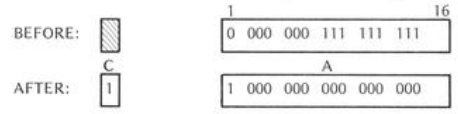
1.6 (316), 0.96 (516) plus 0.8 N microseconds. N = number of shifts.

NOTES:

1. The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.
2. After 16 or more shifts, the A register contains zeros.

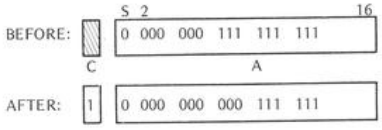
PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	ALS	15		

1. Assume that the value 0 000 000 111 111 111_8 is loaded into the A register.
2. The ALS instruction directs the control unit to shift the contents of the A register 15 places to the left. The C bit retains the value of 1, because the original sign, 0, changed value.



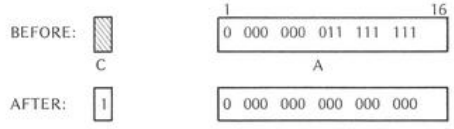
PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	ARS	3		

1. Assume that the value 0 000 000 111 111 111_8 is loaded into the A register.
2. The ARS instruction directs the control unit to shift the contents of the A register 3 places to the right. The C bit retains the value of 1, the last bit shifted from A.



PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	LGL	16		

1. Assume that the value 0 000 000 111 111 111_8 is loaded into the A register.
2. The LGL instruction directs the control unit to shift the contents of the A register 16 places to the left. The C bit retains the value of 1, the last bit shifted from A.



LGR

Logical Right Shift

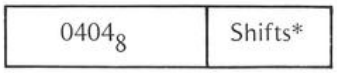
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SSP

   ADDRESS, X
   Not Applicable

2. 1 10 11 16



*In twos-complement notation

FUNCTION:

The control unit is directed to shift the contents of the A register to the right as many places as specified. Zeros replace the vacated high-order positions of A. The C bit retains the value of the last bit shifted from A.

TIMING:

1.6 (316), 0.96 (516) plus 0.8 N microseconds. N = number of shifts.

NOTES:

1. The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.
2. After 16 or more shifts, the A register contains zeros.

LRS

Long Arithmetic Right Shift

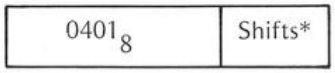
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   LRS

   ADDRESS, X
   Number of Shifts

2. 1 10 11 16



*In twos-complement notation

FUNCTION:

The control unit is directed to treat the A and B registers as one 31-bit register with A the most significant. The contents of this register are shifted to the right as many places as specified. The sign value is shifted into the vacated high-order positions. Bits are shifted from the low-order position of A to the second position of B. The C bit takes the value of the last bit shifted from the B register.

TIMING:

1.6 (316), 0.96 (516) plus 0.8 N microseconds. N = number of shifts.

NOTE:

The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.

LLS

Long Arithmetic Left Shift

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   LLS

   ADDRESS, X
   Number of Shifts

2. 1 10 11 16

*In twos-complement notation

FUNCTION:

The control unit is directed to treat the A and B registers as one 31-bit register with A the most significant. The contents of this register are shifted to the left as many places as specified. Zeros are shifted into the vacated low-order positions. Bits are shifted from the second position of B to the low-order position of A. The C bit takes the value of 1, if the "S" bit changes value. "Arithmetic" indicates that the high-order bit of B is ignored.

TIMING:

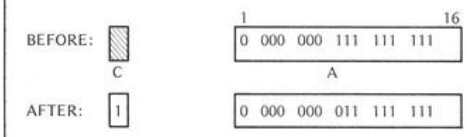
1.6 (316), 0.96 (516) plus 0.8 N microseconds. N = number of shifts.

NOTE:

The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.

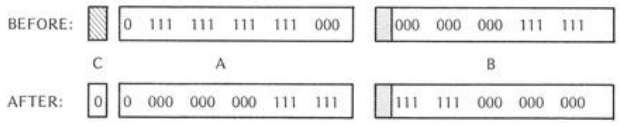
PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30 / 72	IDENTIFICATION 73–80
	LGR	1	

1. Assume that the value $0\ 000\ 000\ 111\ 111\ 111_8$ is loaded into the A register.
2. The LGR instruction directs the control unit to shift the contents of the A register 1 place to the right. The C bit retains the value of 1, the last bit shifted from A.



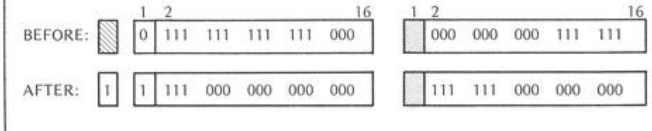
PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30 / 72	IDENTIFICATION 73–80
	LRS	9	

1. Assume that the value, $0\ 111\ 111\ 111\ 111\ 000\ 0\ 000\ 000\ 000\ 111\ 111_2$, is loaded into the arithmetic registers.
2. The LRS instruction directs the control unit to shift the contents of the A and B registers as many places as specified.



PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30 / 72	IDENTIFICATION 73–80
	LLS	9	

1. Assume that the value, $0\ 111\ 111\ 111\ 111\ 000\ 0\ 000\ 000\ 000\ 111\ 111_2$, is loaded into the arithmetic registers.
2. The LLS instruction directs the control unit to shift the contents of the A and B registers as many places as specified.



LRL

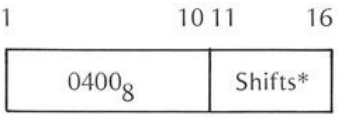
Long Right Logical Shift

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   LRL

   ADDRESS, X
   Number of Shifts

2. 

*In twos-complement notation

FUNCTION:

The control unit is directed to treat the A and B registers as one 32-bit register with A the most significant. The contents of this register are shifted to the right as many places as specified. Zeros are shifted into the vacated high-order positions. Bits are shifted from the low-order position of A to the high-order position of B. The C bit takes the value of the last bit shifted from the B register.

TIMING:

1.6 (316), 0.96 (516) plus 0.8 N microseconds. N = number of shifts.

NOTE:

The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.

LLL

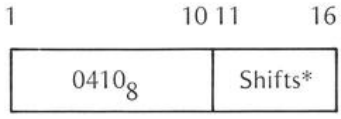
Long Left Logical Shift

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   LLL

   ADDRESS
   Number of Shifts

2. 

*In twos-complement notation

FUNCTION:

The control unit is directed to treat the A and B registers as one 32-bit register with A the most significant. The contents of this register are shifted to the left as many places as specified. Zeros are shifted into the vacated low-order positions. Bits are shifted from the high-order position of B to the low-order position of A. The C bit takes the value of the last bit shifted from A.

TIMING:

1.6 (316), 0.96 (516) plus 0.8 N microseconds. N = number of shifts.

NOTE:

The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.

LLR

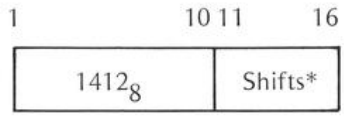
Long Left Rotate

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   LLR

   ADDRESS, X
   Number of Shifts

2. 

*In twos-complement notation

FUNCTION:

The control unit is directed to treat the A and B registers as one 32-bit register with A the most significant. The contents of this register are rotated to the left as many places as specified. Bits are shifted from the high-order position of B to the low-order position of A. The C bit takes the value of the last bit shifted into the B register.

TIMING:

1.6 (316), 0.96 (516) plus 0.8 N microseconds. N = number of shifts.

NOTE:

The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
	L,R,L,	9		

1. Assume that the value, $0\ 111\ 111\ 111\ 111\ 000\ 0\ 000\ 000\ 000\ 111\ 111_2$, is loaded into the arithmetic registers.
2. The LRL instruction directs the control unit to shift the contents of the A and B registers as many places as specified.

	C	A	B
BEFORE:		0 111 111 111 111 000	0 000 000 000 111 111
AFTER:	0	0 000 000 000 111 111	1 111 110 000 000 000

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
	L,L,L,	9		

1. Assume that the value, $0\ 111\ 111\ 111\ 111\ 000\ 0\ 000\ 000\ 000\ 111\ 111_2$, is loaded into the arithmetic registers.
2. The LLL instruction directs the control unit to shift the contents of the A and B registers as many places as specified.

	C	A	B
BEFORE:		0 111 111 111 111 000	0 000 000 000 111 111
AFTER:	1	1 111 000 000 000 000	0 111 111 000 000 000

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
	L,L,R,	9		

1. Assume that the value, $0\ 111\ 111\ 111\ 111\ 000\ 0\ 000\ 000\ 000\ 111\ 111_2$, is loaded into the arithmetic registers.
2. The LLR instruction directs the control unit to rotate the contents of the A and B registers as many places as specified.

	C	A (1 … 16)	B (1 … 16)
BEFORE:		0 111 111 111 111 000	0 000 000 000 111 111
AFTER:	1	1 111 000 000 000 000	0 111 111 011 111 111

LRR

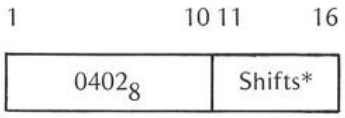
Long Right Rotate

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   LRR

   ADDRESS, X
   Number of Shifts

2. 

*In twos-complement notation

FUNCTION:

The control unit is directed to treat the A and B registers as one 32-bit register with A the most significant. The contents of this register are rotated to the right as many places as specified. Bits are shifted from the low-order position of B to the high-order position of A. The C bit takes the value of the last bit shifted from the B register.

TIMING:

1.6 (316), 0.96 (516) plus 0.8 N microseconds. N = number of shifts.

NOTE:

The number of shifts desired is expressed in decimal notation on the coding form. The assembler makes the conversion to twos-complement notation.

BYTE-HANDLING

In those cases, where data is encoded in 6- or 8-bit codes, the Series 16 instructions set provides five "byte-handling" instructions that also save on memory requirements.

CAL

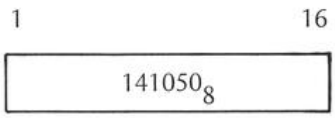
Clear Left Half of A

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   CAL

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is directed to replace the contents of the first half of the arithmetic register with zeros; the second half of the register is unchanged.

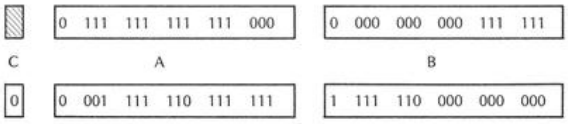
TIMING:

1.6 (316), 0.96 (516) microseconds.

PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12 30	72	73 80
	LRR	9		

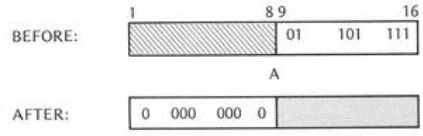
1. Assume that the value, 0 111 111 111 111 000 0 000 000 000 111 111_2, is loaded into the arithmetic registers.
2. The LRR instruction directs the control unit to rotate the contents of the A and B registers as many places as specified.



PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12 30	72	73 80
	CAL			

1. Assume the value 0 111 111 101 101 111_2 is loaded into the A register.
2. The CAL instruction directs the control unit to replace the high-order eight bits of the A register with zeros; the second half of the A register remains the same.



CAR

Clear Right Half of A

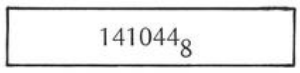
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   CAR

   ADDRESS, X
   Not Applicable

2.



FUNCTION:

The control unit is directed to replace the contents of the low-order eight bits with zeros.

TIMING:

1.6 (316), 0.96 (516) microseconds.

ICA

Interchange Characters in A

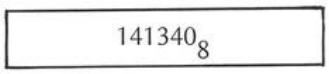
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ICA

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to interchange the high-order eight bits of the arithmetic register with the low-order eight bits of the arithmetic register. The high-order bit changes position with the ninth bit; the second bit changes position with the tenth bit; etc., until the eighth bit changes position with the low-order bit.

TIMING:

1.6 (316), 0.96 (516) microseconds.

ICL

Interchange and Clear Left Half of A

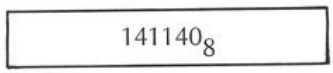
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ICL

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to store the high-order eight bits of the arithmetic register in the low-order eight positions; the low-order eight bits are discarded; the high-order eight positions are filled with zeros.

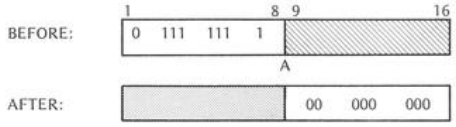
TIMING:

1.6 (316), 0.96 (516) microseconds.

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12	30	72 73 80
	CAR			

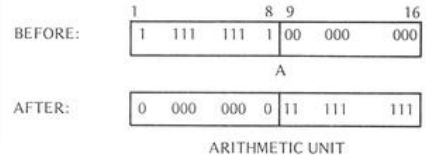
1. Assume the value $0\ 111\ 111\ 110\ 010\ 111_2$ to be loaded in the A register.
2. The CAR instruction directs the arithmetic unit to replace the low-order eight bits of the A register with zeros; the high-order bits remain the same.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12	30	72 73 80
	ICA			

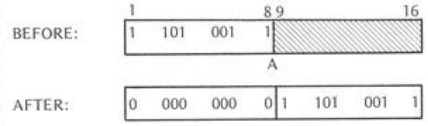
1. Assume that the value $1\ 111\ 111\ 100\ 000\ 000_2$ is loaded in the A register.
2. The ICA instruction directs the control unit to interchange the high-order eight bits with the low-order eight bits.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12	30	72 73 80
	ICL			

1. Assume that the USASCII characters "SE" are loaded into the A register in their binary form, 1 101 001 111 000 101.
2. The ICL instruction directs the control unit to store the first eight bits in the low-order eight positions and to replace the high-order eight positions with zeros. The original low-order eight bits are discarded.



ICR

Interchange and Clear Right Half of A

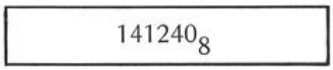
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ICR

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to store the low-order eight bits of the arithmetic register in the high-order eight positions; the low-order positions are filled with zeros; the high-order eight bits are discarded.

TIMING:

1.6 (316), 0.96 (516) microseconds.

CONTROL

The Series 16 control instructions, especially the Compare (CAS) and interrupt-related commands, give the Series 16 important real-time capabilities.

CAS

Compare

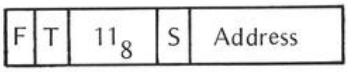
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   CAS (*)

   ADDRESS, X
   Address (,1)

2. 1 2 7 16



FUNCTION:

The control unit is directed to compare, algebraically, the contents of the A register with the contents of the location in memory specified by the effective address.

1. If the contents of the A register are greater than the contents of the memory location, the program executes the next instruction.
2. If the contents of the A register are equal to the contents of the memory location, the program skips the next instruction.
3. If the contents of the A register are less than the contents of the memory location, the program skips the next two instructions.

TIMING:

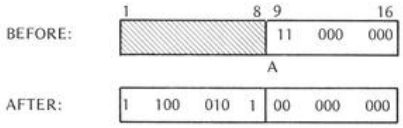
4.8 (316), 2.88 (516) microseconds.

PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
	ICR			

1. Assume that the USASCII characters "SE" are stored in the A register as $1\ 101\ 001\ 111\ 000\ 101_2$.

2. The ICR instruction directs the control unit to store the low-order eight bits (E) in the high-order eight positions. The high-order eight bits are discarded. The low-order eight positions are filled with zeros.

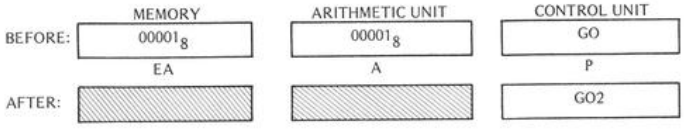


PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
GO	CAS	MEM		
GO1	***		if a > MEM	
GO2	***		if a = MEM	
GO3	***		if a < MEM	

1. Assume that the value $0\ 000\ 000\ 000\ 000\ 001_2$ is stored in the location MEM, and is loaded in the A register.

2. The CAS instruction directs the control unit to compare the two values algebraically. Because they are equal, the program counter was increased by two, and the program effectively skips the next instruction. The instruction at GO2 is executed.



IRS

Increment, Replace, and Skip

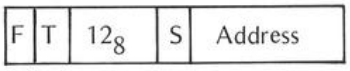
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   IRS (*)

   ADDRESS, X
   Address (,1)

2. 1 2 7 16



FUNCTION:

The control unit is directed to route the contents of the memory location specified by the effective address to the adder. The value of the contents is increased by one and returned to the specified memory location.

1. If the result is not equal to zero, execute the next instruction.
2. If the result is zero, skip the next instruction.

TIMING:

4.8 (316), 2.88 (516) microseconds.

ENB

Enable Interrupt

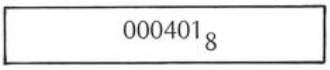
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   ENB

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to permit interruption of the program execution.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTES:

1. The permit interrupt status does not take effect until the completion of the instruction *following* the ENB instruction.
2. This has no effect on the power-failure level of interrupt.

INH

Inhibit Interrupt

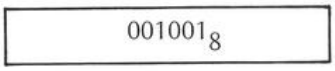
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   INH

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to prohibit the interruption of program execution. This is accomplished by resetting the System Interrupt function in the control unit.

TIMING:

1.6 (316), 0.96 (516) microseconds.

PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE

LOCATION (1–4)	OPERATION (6–10)	ADDRESS X (12–30)	COMMENTS (30–72)	IDENTIFICATION (73–80)
G01	IRS	MEM		
G02	***		if MEM+1 ≠ 0	
G03	***		if MEM+1 = 0	

1. Assume that the value $1\ 111\ 111\ 111\ 111\ 111_2$ is stored in the location MEM.
2. The IRS instruction directs the control unit to route the contents of MEM to the adder where it is increased by one. The result is stored in MEM; the result is zero, therefore the P register is increased by two; that is, the next instruction is skipped.

	MEM	P
BEFORE:	1 111 111 111 111 111	G01
AFTER:	0 000 000 000 000 000	G03

HLT

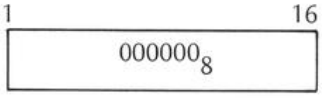
Halt

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   HLT

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is directed to stop program execution.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTES:

1. The program begins execution at the point at which it was halted only after the START button on the panel is pressed.
2. Because of the nature of this instruction, no generalized description of central processor activity is necessary.

NOP

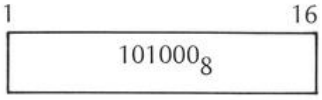
No Operation

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   NOP

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is directed to perform no program operation. The program proceeds to the next instruction.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

The programmer finds this instruction useful at the end of a logical block of instructions. The NOP instruction can be used to patch in instructions or to fulfill timing requirements.

JMP

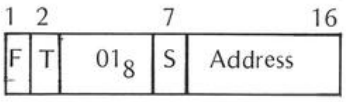
Unconditional Jump

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   JMP (*)

   ADDRESS, X
   Address (,1)

2. 

FUNCTION:

The control unit is directed to store the effective address, found in the Y register, in the P register.

TIMING:

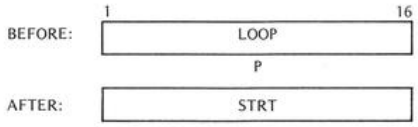
1.6 (316), 0.96 (516) microseconds.

NOTE:

In order to branch from the present or the base sector, an indirect or indexed JMP instruction can be used.

PROGRAMMER			DATE	PAGE OF
PROGRAM				CHARGE
LOCATION 1 4	OPERATION 6 10	ADDRESS X 12	COMMENTS 30 72	IDENTIFICATION 73 80
STRT	***			
LOOP	JMP	STRT		

1. Assume that the instruction located at LOOP in the program is being executed and that that location contains a JMP instruction.
2. The JMP instruction directs the control unit to store the effective address STRT in the P register; the instruction at STRT would be executed next.



JST

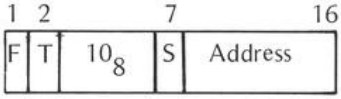
Jump and Store Location

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   JST (*)

   ADDRESS, X
   Address (,1)

2. 

FUNCTION:

The control unit is directed to store the contents of the P register in the location specified by the effective address. The effective address, found in the Y register, is increased by one and routed to the P register.

TIMING:

4.8 (316), 2.88 (516) microseconds.

NOTES:

1. The high-order bits of the P register are not affected by the operation.
2. The JST instruction is used extensively in subroutine linkages; that is, a JST instruction in the main program will cause the program to branch to some subroutine beginning at a location one higher than the effective address. The programmer stores the point in the main program to which he wishes to return at the location specified by the effective address.

RCB

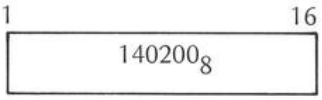
Reset C Bit

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   RCB

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is directed to replace the value in the C bit with a zero.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

Because of the nature of this instruction, no generalized description of central processor activity is necessary.

SCB

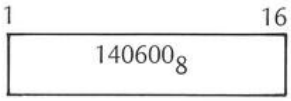
Set C Bit

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SCB

   ADDRESS, X
   Not Applicable

2. 

FUNCTION

The control unit is directed to replace the value in the C bit with a one.

TIMING:

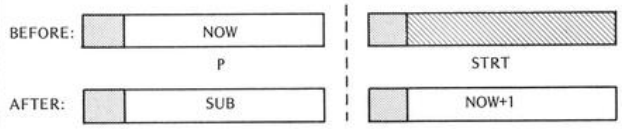
1.6 (316), 0.96 (516) microseconds.

PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12	30	72 73 80
STRT	***			
SUB				
NOW	JST	STRT		

1. Assume that in program execution, the address of an instruction located at NOW, is in the P register. The instruction is a JST instruction.

2. The JST instruction directs the control unit to store the point in the program to which it wishes to return (NOW + 1) at location STRT and to continue program execution at a point in the program one higher than the point at which NOW + 1 is stored.



SKP

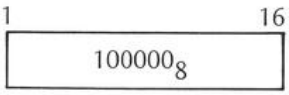
Unconditional Skip

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SKP

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is directed to skip the next instruction under all conditions.

TIMING:

1.6 (316), 0.96 (516) microseconds.

SLN

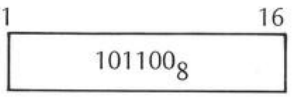
Skip if Low-order Bit One

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SLN

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is instructed to increase the P register by two if the low-order bit of the A register is one.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

Because of the nature of the instruction, no generalized description of central processor activity is necessary. Program action is described below:

Condition	Program Action
$a_{16} = 0$	Execute next instruction
$a_{16} = 1$	Skip next instruction

SLZ

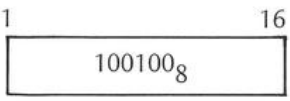
Skip if Low-order Bit Zero

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SLZ

   ADDRESS, X
   Not Applicable

2. 

FUNCTION:

The control unit is instructed to increase the P register by two if the low-order bit of the A register is zero.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

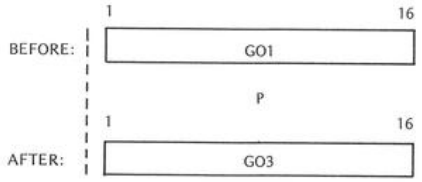
Because of the nature of the instruction, no generalized description of central processor activity is necessary. Program action is described below:

Condition	Program Action
Low-order bit = 0	Skip next instruction
Low-order bit = 1	Execute next instruction

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12 30	72	73 80
G01	SKP			
G02	***			
G03	***			

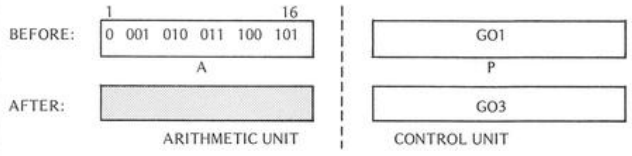
1. Assume the program to be executing an instruction located at "GO" in memory.
2. The SKIP instruction at that location directs the control unit to skip the next instruction, located at "GO2" and execute the instruction at "GO3."



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12 30	72	73 80
G01	SLN			
G02	***		if $A_{16}=0$	
G03	***		if $A_{16}=1$	

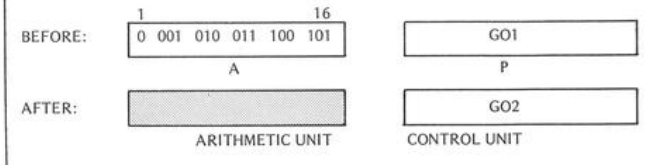
1. Assume that 012345_8 is loaded in A.
2. The SLN instruction directs the control unit to increase the P register by two; that is, to skip the next instruction.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
1 4	6 10	12 30	72	73 80
G01	SLZ			
G02	***		if $a_{16}=1$	
G03	***		if $a_{16}=0$	

1. Assume that 012345 is loaded into the A register.
2. The SLZ instruction directs the control unit to increase the P register by one ($a_{16}=1$); that is, execute next instruction.



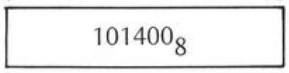
SMI

Skip if A Minus

FORMAT:

1. LOCATION
   Symbol
   OPERATION
   SMI
   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to increase the P register by two if the sign (low order bit) of A is one.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

Because of the nature of the instruction, no generalized description of central processor activity is necessary. Program action is described below:

Condition	Program Action
Sign (a_1) of A = 0	Execute next instruction
Sign (a_1) of A = 1	Skip next instruction

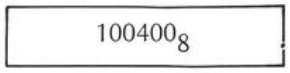
SPL

Skip if A Plus

FORMAT:

1. LOCATION
   Symbol
   OPERATION
   SPL
   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to increase the P register by two if the sign (high-order) bit of the A register is zero.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

Program action is described below:

Condition	Program Action
Sign (a_1) of A = 0	Skip next instruction
Sign (a_1) of A = 1	Execute next instruction

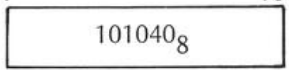
SNZ

Skip if A Not Zero

FORMAT:

1. LOCATION
   Symbol
   OPERATION
   SNZ
   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to increase the P register by two if the contents of the A register is anything other than zero (000000_8).

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

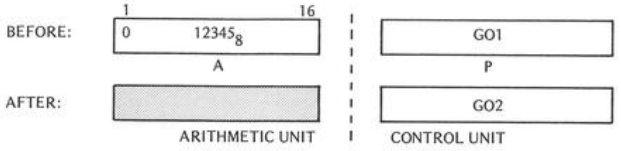
Program action is described below:

Condition	Program Action
$a = 000000_8$	Execution next instruction
$a \neq 000000_8$	Skip next instruction

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
G01	SMI			
G02	***		if A is positive	
G03	***		if A is negative	

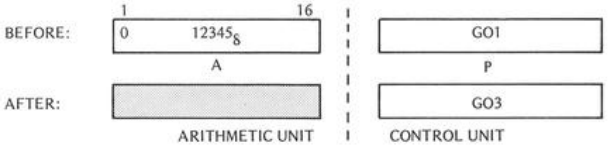
1. Assume that 012345_8 is loaded into the A register.
2. The SMI instruction directs the control unit to increase the P register by one (S = 0); that is, execute the next instruction.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
G01	SPL			
G02	***		if sign is negative	
G03	***		if sign is positive	

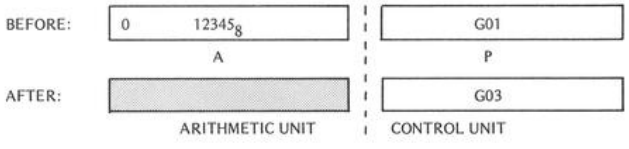
1. Assume that 012345_8 is loaded in the A register.
2. The SPL instruction directs the control unit to increase the P register by two (S = 0); that is, skip the next instruction.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
G01	SNZ			
G02	***		if a = 000000$_8$	
G03	***		if a ≠ 000000$_8$	

1. Assume that 012345_8 is loaded into the A register.
2. The SNZ instruction directs the control unit to increase the P register by two; that is, skip the next instruction.



SZE

Skip if A Zero

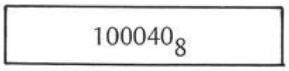
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SZE

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to increase the P register by two if the contents of the A register is 000000_8.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

Program action is described below:

Condition	Program Action
$a = 000000_8$.	Skip next instruction
$a \neq 000000_8$	Execute next instruction

SRn

Skip if Sense Switch Reset

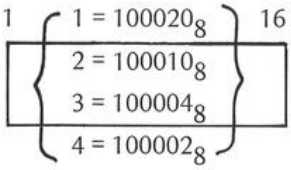
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SRn

   ADDRESS, X
   Not Applicable

2. 1 16



n = 1, 2, 3, or 4

FUNCTION:

The control unit is directed to increase the P register by two if the given SENSE switch is at the OFF position.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

Program action is described below:

Condition	Program Action
SENSE switch n is ON	Execute next instruction
SENSE switch n is OFF	Skip next instruction

SSn

Skip if Sense Switch Set

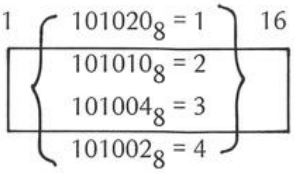
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SSN

   ADDRESS, X
   Not Applicable

2. 1 16



n = 1, 2, 3, or 4

FUNCTION:

The control unit is directed to increase the P register by two if the given SENSE switch is at the ON position.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

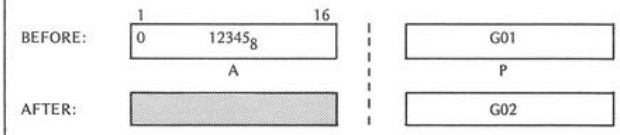
Program action is described below:

Condition	Program Action
SENSE switch is ON	Skip next instruction
SENSE switch is OFF	Execute next instruction

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
G01	SZE			
G02	***		if a ≠ 0000000_8	
G03	***		if a = 0000000_8	

1. Assume that 012345_8 is loaded into the A register.
2. The SZE instruction directs the control unit to increase the P register by one; that is, execute the next instruction.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
G01	SR2			
G02	***		if SENSE switch 2 is ON	
G03	***		if SENSE switch 2 is OFF	

1. Assume that the operator has set the No. 2 SENSE switch to the OFF position.
2. The SR2 directs the control unit to increase the P register by two; that is, skip the next instruction.

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
G01	SS2			
G02	***		if SENSE switch 2 is OFF	
G03	***		if SENSE switch 2 is ON	

1. Assume that the operator has set the No. 2 SENSE switch in the OFF position.
2. The SS2 instruction directs the control unit to increase the P register by one; that is, execute the next instruction.

SSC

Skip if C Set

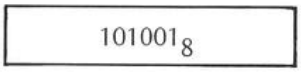
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SSC

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to increase the P register by two if the C bit is equal to one.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

Program action is described below:

Condition	Program Action
C = 0	Execute next instruction
C = 1	Skip next instruction

SRC

Skip if C Reset

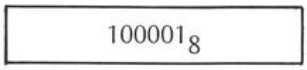
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SRC

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to increase the P register by two if the C bit is zero.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

Program action is described below:

Condition	Program Action
C = 0	Skip next instruction
C = 1	Execute next instruction

SSR

Skip if No Sense Switch Set

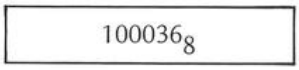
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SSR

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to increase the P register by two if no SENSE switch is at the ON position.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

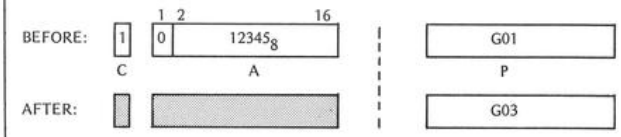
Program action is described below:

Condition	Program Action
No SENSE switch ON	Skip next instruction
Any SENSE switch ON	Execute next instruction

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
G01	SSC			
G02	***		if C = 0	
G03	***		if C = 1	

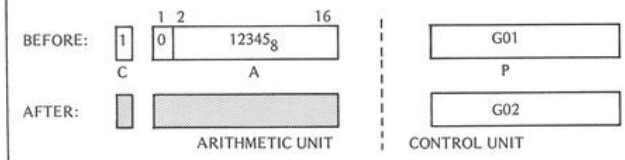
1. Assume that an addition operation has caused the C bit to be set (1).
2. The SSC instruction directs the control unit to increase the P register by two; that is, skip the next instruction.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
G01	SRC			
G02	***		if C = 1	
G03	***		if C = 0	

1. Assume that an addition operation has caused the C bit to be set (1).
2. The SRC instruction directs the control unit to increase the P register by one; that is, execute the next instruction.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
G01	SSR			
G02	***		if any SENSE switch ON	
G03	***		if no SENSE switch ON	

1. Assume that the operator has switched the Nos. 1 and 3 SENSE switches to the ON position.
2. The SSR instruction directs the control unit to increase the P register by one; that is, execute the next instruction.

SSS

Skip if Any Sense Switch Set

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SSS

   ADDRESS, X
   Not Applicable

2. 1 16

101036_8

FUNCTION:

The control unit is directed to increase the P register by two if any SENSE switch is at the ON position.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

Program action is described below:

Condition	Program Action
No SENSE switch ON	Execute next instruction
Any SENSE switch ON	Skip next instruction

INPUT/OUTPUT

Five input/output instructions control the Series 16 system's peripheral activities by both inputting and outputting data via the A register and enabling interrupts and other optional forms of data transfer (DMA and DMC).

OCP

Output Control Pulse

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   OCP

   ADDRESS, X
   I/O Code

2. 1 6 7 10 11 16

14_8	Function	Device

FUNCTION:

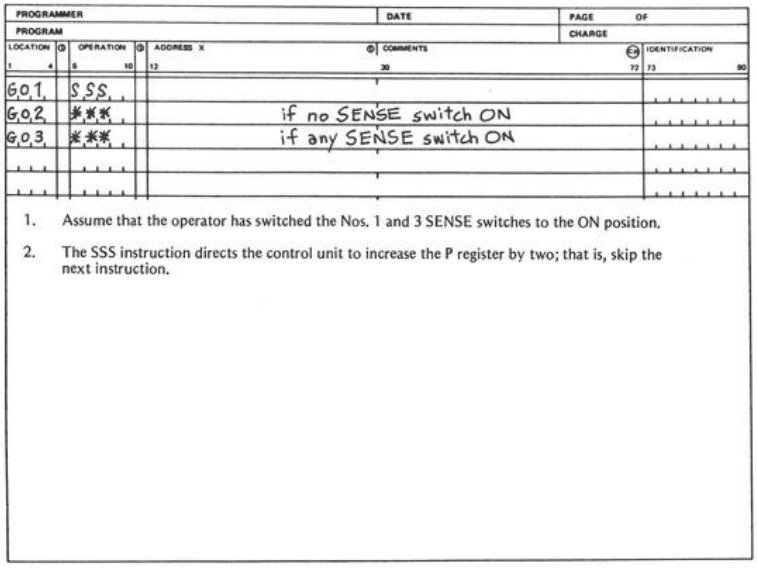
The control unit is directed to generate a control pulse to the peripheral device control or communication controller specified by the device address (bits 11 through 16 of the instruction). The specific control function is indicated by the seventh through the tenth bits of the instruction.

TIMING:

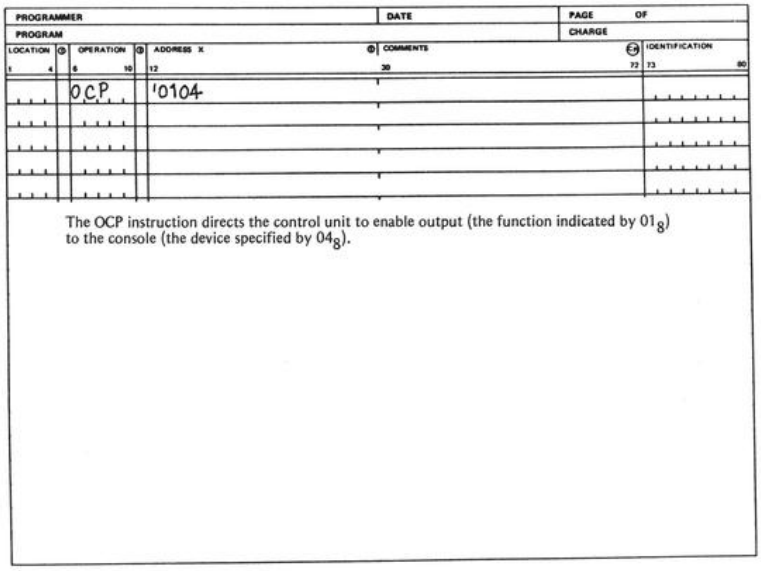
3.2 (316), 1.92 (516) microseconds.

NOTE:

The specific functions associated with an OCP instruction are outlined in the individual device manuals.



1. Assume that the operator has switched the Nos. 1 and 3 SENSE switches to the ON position.
2. The SSS instruction directs the control unit to increase the P register by two; that is, skip the next instruction.



The OCP instruction directs the control unit to enable output (the function indicated by 01_8) to the console (the device specified by 04_8).

SKS

Skip if Set

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SKS

   ADDRESS, X
   I/O Code

2. 1 6 7 10 11 16

34_8	Function	Device

FUNCTION:

The control unit is directed to test the specific function (indicated by bits 7 through 10 of the instruction) in the peripheral device or communication control (specified by bits 11 through 16 of the instruction).

If the condition is met, the next instruction is skipped.

If the condition is not met, the next instruction is executed.

TIMING:

3.2 (316), 1.92 (516) microseconds.

NOTE:

The SKS instruction associated with each individual device control or communication controller is described in the device's hardware manual.

Condition	Program Activity
Condition false	Next instruction executed
Condition true	Next instruction skipped

INA

Input to a Register

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   INA

   ADDRESS, X
   I/O Code

2. 1 6 7 10 11 16

54_8	Function	Device

FUNCTION:

The control unit is directed to test the device to determine if it is ready to transmit data; that is, if the device buffer is full.

1. If the device is busy, no operation occurs and the next instruction is executed.
2. If the device is ready, the seventh bit of the instruction is inspected by the input/output unit.
   a. If the seventh bit is zero, a logical OR operation is performed on the contents of the input line, filled from the device buffer, and the A register. The result is stored in the A register. A logical OR operation is performed according to this truth table:

Contents of Input Bit	0	1	0	1
Contents of A_n Bit	0	0	1	1
Result in A_n Bit	0	1	1	1

   The next instruction is skipped.

   b. If the seventh bit is one, the A register is cleared (its contents are replaced with zeros) and the contents of the device buffer are stored in the A register via the input line. The next instruction is skipped.

TIMING:

3.2 (316), 1.92 (516) microseconds.

NOTES:

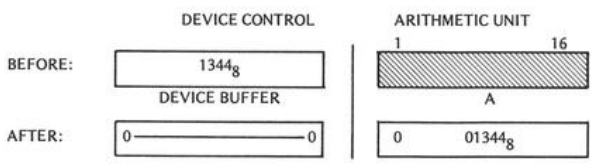
1. The size of the device buffers on the individual peripheral device controls and communication controllers vary; consult the individual hardware manuals for a more complete description of the functioning of the device buffer.
2. Because of the skip nature of this instruction, an INA instruction is often followed by a JMP instruction that either directs the program to jump back one or to jump to some other routine. On the coding form:

PROGRAMMER			DATE	PAGE OF
PROGRAM				CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	SKS	'0104		
	JMP	*-1		
	***			

1. The SKS instruction directs the control unit to test the busy status (the function indicated by 01_8) of the console (the device specified by 04_8).
2. If the console is busy, the JMP instruction will be executed; if the console is not busy, the program skips to some instruction, ***.

PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE
LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	INA	1005		

1. Assume that the value 1344_8 is loaded in the card reader's buffer.
2. The INA instruction directs that value to be loaded into the cleared A register assuming the device is ready to transmit.



OTA

Output from A Register

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   OTA

   ADDRESS, X
   I/O Code

2. 1 6 7 10 11 16

74_8	Function	Device

FUNCTION:

The control unit is directed to test the device to determine if it is ready to receive data; that is, if the device buffer is empty.

1. If the device is busy, no operation occurs and the next instruction is executed.
2. If the device is ready, the contents of the A register are transferred to the device buffer over the output line. Then, the next instruction is skipped.

TIMING:

3.2 (316), 1.92 (516) microseconds.

NOTE:

Because of the skip nature of this instruction, an OTA is often followed by a JMP instruction; on the coding form:

SMK

Set Mask

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SMK

   ADDRESS, X
   I/O Code

2. 1 6 7 10 11 16

74_8	Function	Device

FUNCTION:

The control unit is directed to store the contents of the A register, which has been previously loaded, in the appropriate mask functions. The contents of the A register direct which devices are to be enabled in the interrupt mode.

TIMING:

3.2 (316), 1.92 (516) microseconds.

NOTE:

The SMK instruction is merely the OTA instruction especially adapted for the purpose of setting the Control Interrupt function in individual peripheral device controls or communication controllers; therefore, the two instructions have the same octal value.

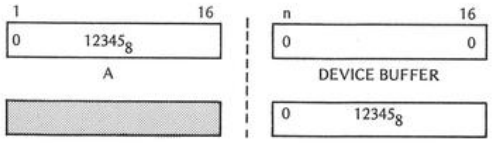
PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE

LOCATION 1 4	OPERATION 6 10	ADDRESS X 12	COMMENTS 30 72	IDENTIFICATION 73 80
	OTA	'0004		
	JMP	*-1 or ROUT	JUMP IF NOT READY	
	***		OTA AND CONTINUE, IF READY	

PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	OTA	0003		

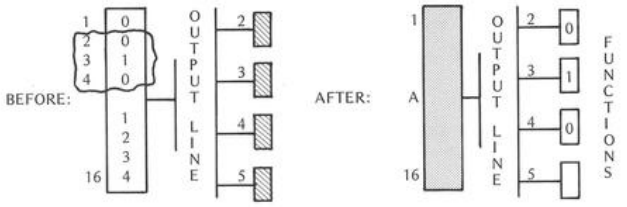
1. Assume that the value 012345_8 is loaded in the A register.
2. The OTA instruction directs the value to be loaded into the print buffer if the buffer is empty. Transfer would occur on the output line.



PROGRAMMER		DATE	PAGE OF
PROGRAM			CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	SMK	0020		

1. Assume that the A register is loaded with the value 021234_8.
2. The SMK instruction directs the control unit to store these values in the appropriate Control Interrupt functions.



FEATURED CAPABILITES

Other instructions are included with the feature available on the Series 16. These include the ten instructions associated with the Feature – High-Speed Arithmetic Package Feature – Memory Party, Feature –, Extended Addressing, and Feature –, the Restricted Mode Package.

DBL

Enter Double-Precision Mode

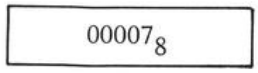
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   DBL

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

This instruction directs the control unit to execute in the double-precision mode.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTES:

1. This mode can be deactivated by either pressing the MSTR CLEAR button on the control panel or by an SGL instruction.
2. Effectively, this instruction directs the control unit to execute LDA, STA, ADD, and SUB instructions as DAD, DSB, DLD, and DST instructions.

SGL

Enter Single-Precision Mode

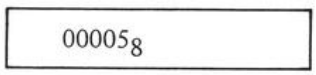
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SGL

   ADDRESS,X
   Not Applicable

2. 1 16



FUNCTION:

This instruction directs the control unit to return program execution to single-precision mode.

TIMING:

1.6 (316), 0.96 (516) microseconds.

DLD

Double-Precision Load

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   DLD (*)

   ADDRESS, X
   Address (, 1)

2.

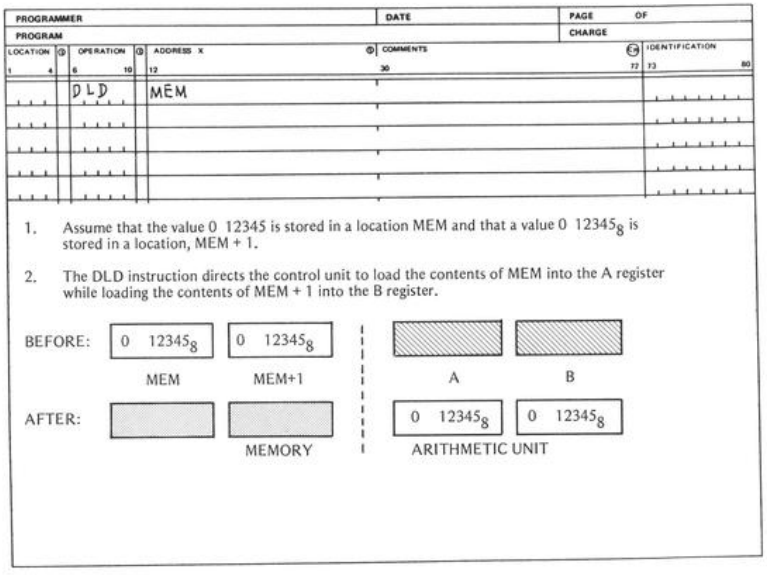
1 2	7	16
F T 02_8	S	Address

FUNCTION:

The control unit is directed to load the contents of the memory location specified by the effective address in the primary arithmetic register while loading the contents of the memory location one greater than that of the effective address in the secondary arithmetic register.

TIMING:

4.8 (316), 2.88 (516) microseconds.



DST

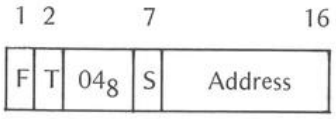
Double-Precision Store

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   DST (*)

   ADDRESS, X
   Address (, 1)

2. 

FUNCTION:

The control unit is directed to store the contents of the primary and secondary arithmetic registers in the location specified by the effective address and in the next higher location, respectively.

TIMING:

4.8 (316), 1.88 (516) microseconds.

DAD

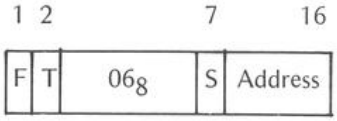
Double Precision Add

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   DAD (*)

   ADDRESS, X
   Address (, 1)

2. 

FUNCTION:

The control unit is directed to add, algebraically, the contents of the A and B registers to the contents of the locations in memory specified by the effective address. The two-word result is stored in the A and B registers with the least significant bits stored in the B register. The C bit would be set if overflow occurred as a result of this operation.

TIMING:

4.8 (316), 2.88 (516) microseconds.

NOTE:

If the low-order bit of the second memory location is not equal to the low-order bit of the B register, the result is invalid.

DSB

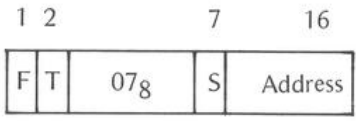
Double Precision Subtract

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   DSB (*)

   ADDRESS, X
   Address (, 1)

2. 

FUNCTION:

The control unit is directed to subtract, algebraically, the contents of the memory locations specified by the effective address, from the contents of the primary and secondary arithmetic registers; the result is stored in the arithmetic registers.

TIMING:

4.8 (316), 2.88 (516) microseconds.

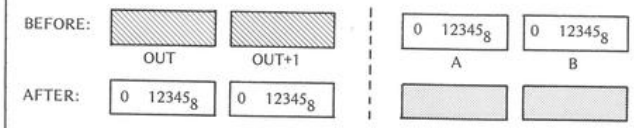
NOTE:

If the low-order bit of the memory location one higher than the location specified by the effective address is not equal to the low-order bit of the secondary arithmetic register, the difference is not correct.

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
	DST	OUT		

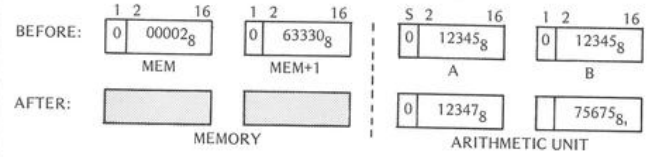
1. Assume that the value 0 12345 0 12345_8 is loaded in the A and B registers.
2. The DST instruction directs the control unit to store the contents of B in location, OUT + 1 and to store the contents of A in location, OUT.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
	DAD	MEM		

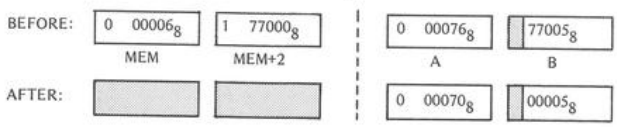
1. Assume that the value 0 12345 0 12345_8 is stored in the arithmetic registers and that the value 0 00002 0 63330_8 is stored in location, MEM and MEM + 1.
2. The ADD instruction directs the control unit to route the two values to the adder where they are algebraically added; then they are stored in the A and B registers.



PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION	OPERATION	ADDRESS X	COMMENTS	IDENTIFICATION
	DSB	MEM		

1. Assume that the value 000076 177005_8 is stored in the A and B registers and that the value 000006 177000_8 is stored in locations MEM and MEM + 1, respectively.
2. The DSB instruction directs the control unit to subtract MEM and MEM + 1 from the arithmetic registers to derive the value 0 00076 100005_8 in the arithmetic registers.



MPY

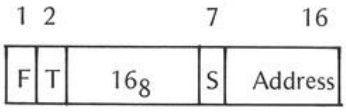
Multiply

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   MPY (*)

   ADDRESS, X
   Address (, 1)

2. 

FUNCTION:

The control unit is directed to multiply the contents of the location specified by the effective address by the contents of the arithmetic register and the result is stored in the primary and secondary arithmetic registers, as necessary.

TIMING:

8.8 (316), 5.28 (516) microseconds.

DIV

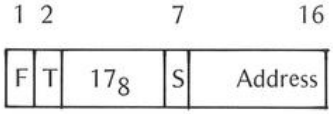
Divide

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   DIV (*)

   ADDRESS, X
   Address (, 1)

2. 

FUNCTION:

The control unit is directed to divide the contents of the primary and secondary arithmetic registers by the contents of the location specified by the effective address; the result is stored in the primary arithmetic register; the remainder is stored in the secondary arithmetic register. Overflow will occur if the initial magnitude of the dividend is greater than or equal to the magnitude of the divisor.

TIMING:

17.6 (316), 10.56 (516) microseconds.

NOTE:

The most significant bits of the dividend (the contents of A) must be less than the value in the divisor (the contents of EA) for division to take place; if this condition is not satisfied the overflow indicator is set.

NRM

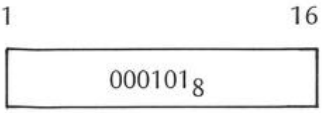
Normalize

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   NRM

   ADDRESS, X
   Not Accepted

2. 

FUNCTION:

The control unit is directed to shift the contents of the primary and secondary arithmetic registers to the left; the low-order position remains the same; the second position of the arithmetic register is shifted out and discarded; the low-order position of the secondary register is ignored. The shifting of positions continues until the value of the low-order bit of the primary arithmetic register is <u>not</u> equal to the second bit of the primary arithmetic register. The vacated high-order positions are filled with zeros. The number of shifts is stored in the normalize shift counter.

TIMING:

1.6 (316), 0.96 (516) plus 0.8 N microseconds.

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	MPY	MEM		

1. The value 77777_8 is stored in location MEM by the DEC pseudo-operation.
2. The MPY instruction directs the arithmetic unit to multiply the contents of the A register and the contents of MEM and to store the results in the A and B registers, as necessary.

	MEM	A	B
BEFORE:	0 77777_8	0 00003_8	0 00000_8
AFTER:		0 00001_8	0 77777_8

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	DIV	MEM		

1. Assume that the value 0 00003 0 03240_8 is loaded into the arithmetic registers and that the value 0 00040_8 is loaded into the location MEM.
2. The DIV instruction directs the control unit to divide the value in the arithmetic registers by the value in location MEM. The quotient is stored in A and the remainder is stored in B.

	MEM	A REG	B REG
BEFORE:	40_8	0 00001	1 03240_8
AFTER:		0 06065_8	0 00000_8

PROGRAMMER | DATE | PAGE OF
PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30	IDENTIFICATION 73–80
	NRM			

1. Assume that the value 0 00077 1 77777_8 is loaded in the arithmetic registers.
2. The NRM instruction directs the control unit to shift the contents to the left until there is a one in the second position of A. When this requires 9 shifts, a 9 is placed in the shift counter.

	A (1 2 – 16)	B (1 2 – 16)	NORMALIZE SHIFT COUNTER (1 – 5)
BEFORE:	0 00077_8	77777_8	
AFTER:	0 77777_8	77000_8	01001

SCA

Shift Count To A

FORMAT:

1. LOCATION
   Symbol

   OPERATION
   SCA

   ADDRESS, X
   Not Applicable

2. 1 16

00041_8

FUNCTION:

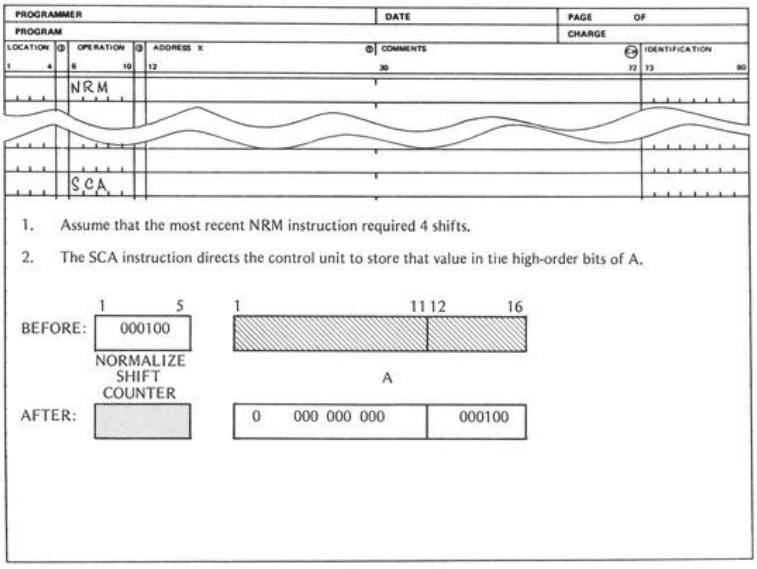
The control unit is directed to store the contents of the normalize shift counter in the high-order five positions of the A register. The low-order positions of A are filled with zeros.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

The normalize shift count is valid if no IAB, MPY, DIV, OTK, shift or double-precision instruction has been executed since the last NRM instruction.



SPN

Skip if No Parity Error

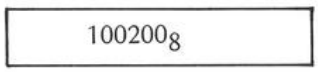
FORMAT:

1. LOCATION
   Ignored

   OPERATION
   SPN

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to skip the next program instruction if the memory parity error indicator is reset (bit = 0).

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTES:

1. The following program action is directed by this instruction.

   CONDITION/ACTION

   Error indicator set/Execute next instruction
   Error indicator reset/Skip next instruction

2. A parity error indication can be set by an error in either an instruction or a data word.
3. Memory parity is displayed on the control panel at lamp 15 if the OP mode is selected.
4. Memory parity can be attached to the interrupt facility.

SPS

Skip on Memory Parity Error

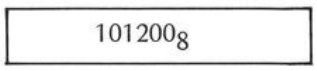
FORMAT:

1. LOCATION
   Ignored

   OPERATION
   SPS

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to skip the next program instruction if the memory parity error indicator of the central processor is set (bit = 1).

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTES:

1. The following program action is directed by this instruction.

   CONDITION/ACTION

   Error indicator set/Skip next instruction
   Error indicator reset/Execute next instruction

RMP

Reset Memory Parity Error

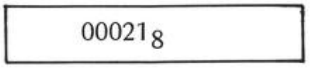
FORMAT:

1. LOCATION
   Ignored

   OPERATION
   RMP

   ADDRESS, X
   Not Applicable

2. 1 16



FUNCTION:

The control unit is directed to reset (bit = 0) the memory parity error indicator of the central processor.

TIMING:

1.6 (316), 0.96 (516) microseconds.

EXA

Enable Extended Addressing

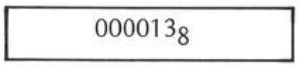
FORMAT:

1. LOCATION
   Symbol

   OPERATION
   EXA

   ADDRESS, X
   Not Accepted

2. 1 16



FUNCTION:

The control is directed to set the EXTMD (Extended Mode Indication); this mode enables the Series 16 processor to utilize memory above 16K.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTES:

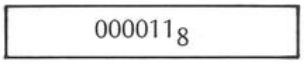
The extend mode alters the JST instruction (page 114) to allow it to store a 15-bit program counter. Bit 1 of the memory location specified by the effective operand address is left unchanged.

DXA

Disable Extended Addressing

FORMAT:

1 16



FUNCTION:

The control is directed to reset the EXTMD, this action restores the processor to normal mode. See note.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

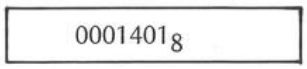
The mode change directed by the DXA instruction is not effective until a JMP (page 112) instruction, enabling proper return from an interrupt routine, has been executed. Any number of other instructions can be executed between the DXA and the JMP instruction.

ERM

Enter Restricted Mode

FORMAT:

1 16



FUNCTION:

The control is directed to enter the restricted mode of processor operation and to enable program interrupt.

TIMING:

1.6 (316), 0.96 (516) microseconds.

NOTE:

The processor remains in the restricted mode until a program interrupt occurs.

PROGRAMMER | DATE | PAGE OF

PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	EXA		ENTER EXTENDED MODE	

BEFORE: 

EXTMD

AFTER: [1]

PROGRAMMER | DATE | PAGE OF

PROGRAM | CHARGE

LOCATION 1–4	OPERATION 6–10	ADDRESS X 12	COMMENTS 30–72	IDENTIFICATION 73–80
	DXA			
END	JMP	MAIN PRGM		

1. Assume that during a given subroutine, a DXA instruction is processed.
2. Upon the return of program control to the main program flows, the Extended Mode Indicator is reset.

BEFORE: [1]

EXTMD

AFTER: [0]