THE WANG LABORATORIES

Programmer

Published Monthly At Tewksbury, Massachusetts, U. S. A.

JUNE 1970 Vol. 4 No. 6

SPECIAL ISSUE

THE WANG 3300 TIME SHARING MINI-COMPUTER

The PROGRAMMER

Published monthly by Wang Laboratories, Inc. at Tewksbury, Massachusetts, as a service of information to users of its calculating equipment and computing systems. Users who have a program, application or story which may be shared with other readers, are invited to write to the Editor.

TABLE OF CONTENTS

THE WANG 3300 TIME SHARING MINI-COMPUTER

PREFACE: *The 3300 is a time-sharing mini-computer system for only one-fourth the cost of subscription services or most other in-house time-sharing systems. It features the popular BASIC computer language which minimizes computer communication problems for both beginners and experts. The system can be initiated with a single terminal, and hardware can be added as required to accommodate up to 16 users simultaneously.*

The following sections have been included to provide an integrated historical description of the 3300, from inception to development.

Section 1 introduces the reader to time-sharing and the reasons for its creation. Many fundamental but interesting points describing this concept of computer use are presented.

Section 2 explains Wang Laboratories' specific reasons for entering the computer market. In this section, the economic advantages of the 3300 mini-computer as well as its power as a computing tool are shown to fill a large gap between the calculator and the large computer.

Section 3 describes the development of the hardware for the 3300. The central processor, programmable registers, memory organization and addressing are only some of the hardware features described in this section.

Section 4 contains a comprehensive discussion of the extensive software development for the 3300. The many details and features of the BASIC computer language are described. In addition, the program organization and math package are discussed. A glossary of computer terminology is included for the reader's convenience.

Section 5 is a survey of the overall capability of the 3300. This section is composed of reprints of eight different articles which appeared in major magazines and summarizes the overall features of the 3300 from several different viewpoints.

The five sections described above are intended to supply the reader with a brief overview of Wang Laboratories' recent entrance into the computer field.

1. WHAT IS TIME-SHARING?

Peter Seymour
Sales Support Analyst, Wang Laboratories

A great deal of work is being done today on what are termed small computer time-sharing systems; i.e. systems which fall into the general price range of $15,000 – $200,000. In the near future, it will be possible for any company to afford a complete time-sharing system, dedicated to its own needs.

Inexpensive mini-computers are fast becoming a popular means of supplying small companies with capabilities previously not available. When supplied with an operating system with time-sharing capability, mini-computers offer an extremely efficient service in educational and industrial applications at a cost which is well within the associated limited budgets. At this point, certain questions come to mind. What are mini-computers? What does time-sharing actually mean? What is conversational programming?

IN PARTICULAR, DO YOU KNOW?

In the early 1960's most companies that owned computers had their machines installed in a single, large room. The environment of the computer room had to be carefully controlled to keep the temperature and humidity within certain limits, which would not adversely effect the computer's operation. The computer itself generally consisted of several components. The central processing unit, or cpu, was the brains of the system; it was here that all programs were actually executed. Connected to the cpu were several types of input/output and storage devices such as card readers, high speed line printers, magnetic tape drives, and disk storage units. Even a medium sized system might cost nearly a million dollars; obviously not every company could afford their own computer.

A user wishing to execute a program would first have it punched onto a deck of cards. He would then present his program deck to the computer operator. Several individual decks would be stacked, one behind the other, in a card reader, and the operator would instruct the cpu to execute all programs in that particular stack. Residing inside the cpu's core memory was a package of machine language programs called an operating system, which contained an assembler or a compiler program. It was the task of the operating system to read each incoming program, to translate it into machine language using the assembler or the compiler, to check it for errors, and to execute it. This type of operating procedure is known as batch processing.

The speed with which computers could execute a program was tremendously fast compared with the time required to accept an incoming program or to print the results of a completed job. Most computer systems in those days were designed in such a way that they could execute only a single program at a time, and much of the system would be idle while these jobs were entered or while the results were returned to the operator (Figure 1.1). This represented a

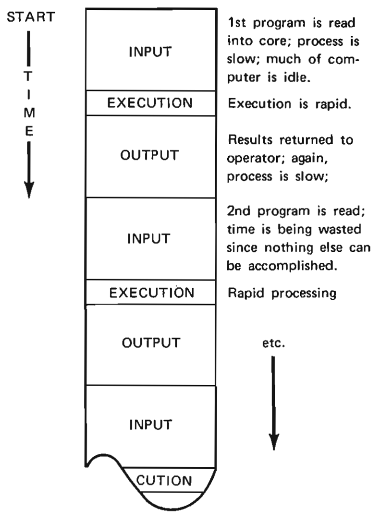


Figure 1.1 *Single Program Execution Schematic*

serious loss of time and money to the company that owned and operated the installation. Occasionally, people would write programs which would cause the computer to stop and wait for them to enter additional instructions through the cpu's console before continuing. Obviously, this was not an efficient method of operating, since the entire computer would lie idle, its speed limited by the typing speed of a single person. When an error was found in a job, the entire program would be discarded by the machine and returned to the programmer. During periods when the computer was extremely busy, it was not uncommon for a person to wait several days before receiving the results of his program.

If the entire system could be redesigned in such a way that it would execute some programs at the same time that it returned results from, or entered several others, then little or none of the machine's potential would be wasted. Also, if each user had his own input device to the computer, he could receive his results almost instantaneously. The entire operation would be far more efficient, since he would have the opportunity to rerun the program immediately if necessary. The concept of time-sharing (T-S) grew out of a concerted effort to develop such a system. The first fully successful T-S system was project MAC, put into operation at the Massachusetts Institute of Technology in 1963.

Figure 1.2 is a schematic diagram of a possible four-user time-sharing system. The central portion of the figure represents core memory inside the cpu. Attached to the cpu are four input/output devices, known as terminals. There are several types of terminal devices on the market today, but in general they all have the following capabilities:

i) allowing a user to transmit his programs and data to the cpu, and
ii) allowing the computer to relay its results back to the user.

The most common terminal in use is the Teletype, which is similar in many respects to a standard typewriter. A person may enter information into the computer by typing it on the Teletype's keyboard; as the information is sent to the cpu, a printed copy is produced at the terminal. In returning the results of the inputted program, the computer assumes the role of the typist. Some terminals are equipped with a unit which allows the user to have his program or data punched on a strip of paper tape. These tapes may then be saved and read back into the computer at a later date. Saving programs and data in this fashion can be inconvenient, but it is less expensive than using high speed disk or magnetic tape storage devices at the computer site. Terminals may reside either near the computer or at great distances from it, e.g. communicating via telephone lines (see Terminal 4 in Figure

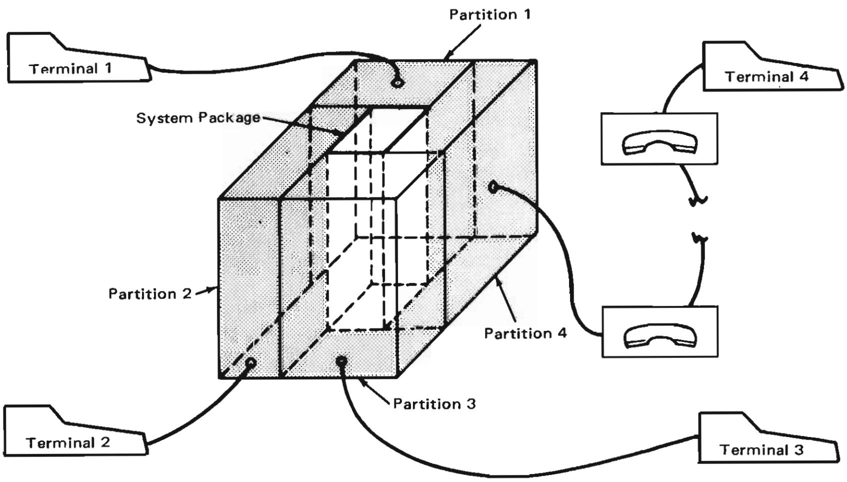


Figure 1.2 *Time-Sharing Schematic*

1.1). No longer need a person bring his work to the computer, in effect, the computer has gone to him.

Initially, when a T-S system is installed, a suitable operating system, including a compiler and a controller is read into a predetermined block of core memory. In the figure, this block is shown as a column which runs through the center of core, although in an actual system its location is irrelevant. A controller may be written in such a way that it divides the remainder of core into equal segments, called partitions, and assigns a unique partition to each terminal. In figure 1.1, these partitions are shown surrounding the system package.

In order to use the system, a user turns his terminal on, thereby establishing a link with the main computer, and types his program into the terminal. Each individual program is stored and executed only in the user's own partition; *in this way, a person's job cannot be altered or destroyed by anyone but himself.* It is the controller's job to continually check each partition, keeping track of which users have jobs that are awaiting execution. (See Figure 1.3). If user number 1 executes a program which has already been entered into core, the compiler executes a fixed portion of that program (Figure 1.3a) and passes on to the next partition. Input and output are handled by a separate portion of the operating system under what is termed "interrupt control". User number 2 may be entering a new line of his program. As he begins to enter the line, an interrupt signal is turned on and control of his partition is handed over to the special input/output program (Figure 1.3b). As long as that signal remains on, partition 2 will be bypassed by the controller. The I/O program simply stores each character as it is entered, in a small area called a buffer. The person using terminal 3 may have just completed entering a line of his program, so his interrupt signal is turned off (Figure 1.3c). The compiler checks his entire line for errors and passes on to partition number 4 if no errors are found. If an error had been found, then the interrupt signal would again be switched on and the I/O program would be charged with transferring the characters of an error message to terminal 3. This partition would be skipped by the controller until the message was completed and the interrupt turned off. If no one is using terminal 4, the controller simply returns to the first partition and continues to execute that program (Figure 1.3d). This process continues until everyone's needs have been satisfied. *The system is truly sharing its time among all persons who wish to utilize its services.*

In order that a time-sharing system of this kind be most effective, it must actually converse with each user. Approximately forty percent of all time-sharing computers utilize a conversational programming language which was developed at Dartmouth College. The language is called BASIC, which is an acronym for Beginner's All-Purpose Symbolic Instruction Code. As each person initially enters a line of his Basic program, the compiler checks it for syntax errors. These are grammatical mistakes such as misspelled words, missing punctuation, etc., which violate rules of the Basic language. If an error has been made, the system types out an error message to the user and permits him to correct his mistake. When the user requests that his program be executed, the system then checks for logic errors; if any are found, the same procedure is followed. There are also provisions in Basic for writing programs which will stop execution and request the user to type in extra information. It is worth repeating here that *the computer does not actually stop* in a case like this While it waits for, and then accepts the new data, it is performing tasks for all other persons who are using the system.

Modern time-sharing computers come in an assortment of sizes, shapes, colors, and prices. For a company contemplating the use of a time-sharing system, however, there are only two roads open: owning its own complete system, or purchasing a terminal and leasing time on someone else's computer. Until recently, the prohibitive cost of computer systems (see Section 2) has persuaded many people to adopt the second alternative. In establishing a commercial T-S service bureau, a limited number of partitions are generally offered to serve a much larger

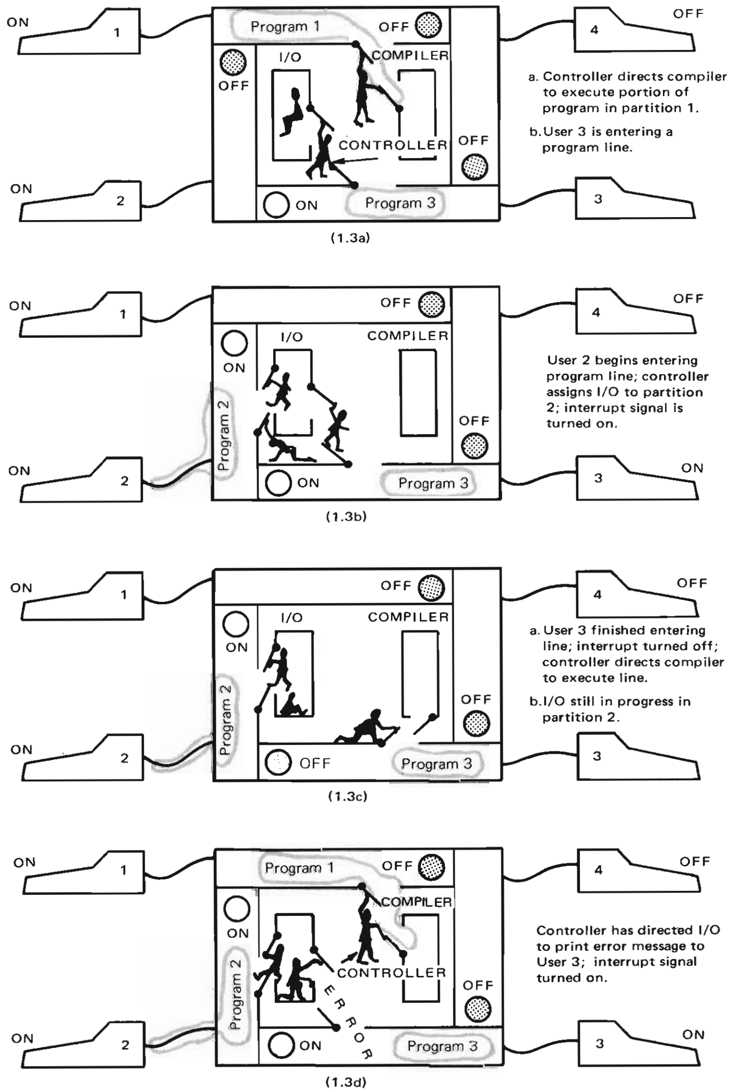


Figure 1.3 *Typical Operation of a Four User Time-Sharing System*

number of customers. Many services assign specific time slots to each of their users, and no one is allowed on the system outside the bounds of their allotted schedule. In effect the T-S vendors play a game of percentages: each user is given a decent chance that his partition will not be busy when he wishes to use it.

Ideally, a company would like to control its own in-house system if a suitable combination of low cost and high power are available. In place of high monthly rental charges, limited time slots, and busy partitions a company would purchase the system for a fixed total price and be guaranteed unlimited usage and private core parititons for each terminal. In addition, most small computers operate under a wide range of temperature and humidity levels, which means that no specialized computer room need be constructed. The time is indeed just around the corner, when it will be possible for any company to afford a complete time-sharing system dedicated to its own needs.

2. WANG LABORATORIES ENTERS THE COMPUTER MARKET

John Cunningham
3300 Marketing Manager

I thought that Wang Laboratories made Calculators. Wang does, and does it well. But, Wang Laboratories does not have a defined objective of selling calculators – rather the objective has been to provide a low cost problem solving tool that fulfills the calculational need that exists in all areas of business, industry, education and the sciences.

This was the thought behind Dr. Wang's development of the LOCI* and both the 300 and 700 Series Calculators – and is the same impetus for the development and introduction of the 3300 BASIC System. Commercial Time Sharing systems, most in the $1,000,000 range, were also designed to fill a need – the need that existed when engineers required quick answers to problems, and were forced to waste days of turn around time on a batch processing computer system. In 1963, as today, a computer for each engineer is an economic impossibility. Time sharing was conceived to improve this situation. The results of the efforts on project MAC and at Dartmouth discussed in Section 1 provided one fast central processor which could service many individuals at prices that were becoming justifiable. The GE235 computer system development at Dartmouth went one step further – the development of a special time sharing language called BASIC. BASIC is a language designed for ease of operation. It also incorporates special problem solving features that make it simple for engineers and scientists to obtain answers quickly and easily.

Even with the introduction of time sharing and BASIC in 1964, there was a large price gap in problem solving tools. (See figure 2.1). During the four years from 1966 to 1970, there have been significant changes in the design and development of problem solving tools. New advances in design, such as Wang's Model 700 Calculator, have increased the capability of the single problem

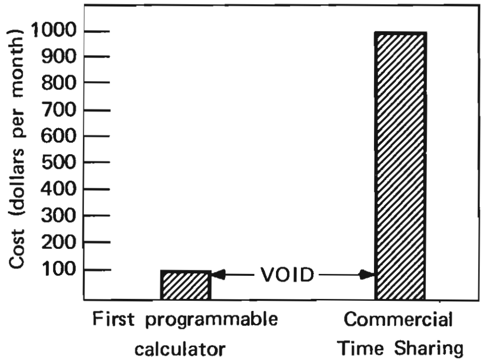


Figure 2.1 *1966–Costs for Problem Solving Tools*

solving tool. To purchase a machine such as the 700 Calculator only 15 years ago, would have cost approximately $100,000! A parallel development has taken place in the time sharing industry. Faster and less expensive central processors have cut average monthly time sharing terminal costs in half. These costs have decreased $500-$600 per month (Figure 2.2) and the associated purchase price has been reduced to the $150,000 range.

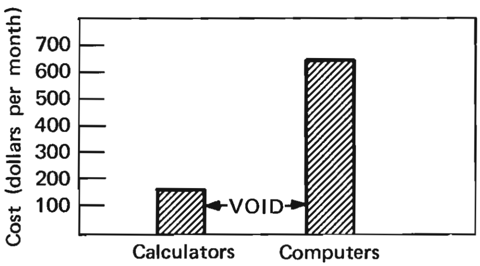


Figure 2.2 *Typical Time-Sharing Costs – January 1, 1970*

Today there still remains a significant price gap between time sharing calculation and large time sharing computers. This gap, however, is not as significant as it was only five years ago. The 3300 BASIC Time Shar-

*A desk-top computer developed by Wang Laboratories in 1964.

ing Computer System fills the void shown in figure 2.2. It is a modular type of system that neatly fills the need for performance and price between the most expensive calculators and least expensive rented terminals.

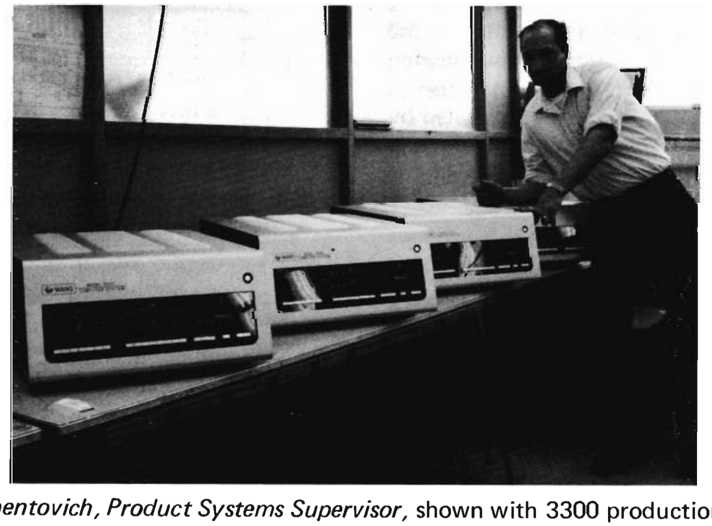
The result is that Wang Laboratories is in the unique position of not selling calculators or time sharing systems specifically, but of providing a 'variety' of problem solving tools from $900 calculators to complete time sharing systems.

Operator features incorporated from inception:

From inception, Wang was able to incorporate features into the 3300 for operator convenience that make the system perform more efficiently than many others costing 25 to 30 times as much! The error correction procedures, debugging, syntax correction, plotting, (see sections 4 and 5) are all designed for the individual that is not a professional 8 hour a day programmer. By extending the already simple BASIC language to build in these special features, Wang Laboratories has provided a system with excellent performance at a reasonable price. By desiging the 3300 computer central processor in a special way to handle multiple terminals, has allowed the complete time sharing system to be priced at only a fraction of other systems.

The growth that Wang has experienced over the past ten years in products, salesmen, service centers, software support (PHI Computer Services) and home office backup puts the Company in a unique position to support an integrated hardware/software system such as 3300 BASIC. *Lack of support has been a major problem in other low cost time sharing systems.* Wang Laboratories is one of the first companies with the manpower to solve the problem, and to put out a time sharing system with a firm commitment that it will continually improve and become more efficient. To solve the computational and computing needs of individuals at a reasonable price is Wang's business.

There is a significant difference between the 3300 System and other systems. The 3300 may be employed as a one user BASIC system which can be economically justified by creating substantial savings over alternative methods. This single user system can readily be expanded to a $200,000 time sharing system with the simple addition of plug-in peripherals. This is a unique feature of 3300 BASIC, and makes it the first in-house system that a user can justify in the $15,000 purchase range, still employ the system for 7 to 8 years, and expand it to 16 on line terminals and a full range of peripheral options.



Joe Klementovich, Product Systems Supervisor, shown with 3300 production models.

3. DEVELOPMENT OF HARDWARE

Ned Chang
Vice President, Corporate Planning, Wang Laboratories

3.1 Introduction

The Wang 3300 is a mini-computer designed for efficient operation in a multi-user system. The low-cost BASIC time sharing system is a good example of such a system. Some of the general hardware-related requirements for BASIC are:

1. Several (1 to 16) terminals must have access to the computer in a time sharing mode. Each terminal requires independent processing and a relatively short response time.
2. Sophisticated systems softwares, including a BASIC language processor, a dynamic resource (core memory) allocator and an I/O controller (IOC) must reside in memory. Particular attention was paid to stack operations and decimal arithmetic.
3. Flexibility in handling alpha-numeric data as well as strictly numeric data is necessary.
4. The system must be open ended so that expansion (such as a disc) is not only possible but reasonably painless.
5. The system could be serviced easily by the existing field service organization.
6. The cost must be low, starting below $15,000. (The generally accepted definition of a mini-computer is $25,000.)

These goals are most readily illustrated by certain features of the computer. They shall be discussed after a general outline of the machine is presented.

3.2 General Outline of the Machine:

Central Processor:

The central processor organization is very simple. The design philosophy was to provide a maximum number of general instructions, (which can reference memory, for example). There are five programmable registers, each register consists of eight bits*, except for the S-Register which has seven bits.

B-C Registers:

The two B-C registers hold the current instruction address. B contains the page address and C contains the address with the page.

A-Register:

The A-Register is a general purpose register. It serves both as the accumulator for single-precision addition and as the low-order accumulator for double-precision addition.

Z-Register:

The Z-Register is a second general purpose register. It serves as the high order accumulator for double-precision arithmetic Z and also holds the device address during an I/O command.

S-Register:

The S-Register consists of seven individual status bits . These include two general bits for program switches, add last carry bit, add zero last result bit, a valid decimal bit, an absolute page designating bit, and a decimal mode bit.

The Instruction Word:

An instruction word consists of two bytes*. There are three types of instructions: full memory reference commands, memory reference commands, and mini-commands.

Full memory reference format:

O_p	D	M
5 bits	3 bits	8 bits

Memory reference format:

O_p	D	M
5 bits	2 bits	8 bits

Mini-command format:

O_p	M
8 bits	8 bits

*A byte is the basic unit of core memory for storing program commands and data. Each byte is composed of 8 bits. The bit is a small ferromagnet used as an on-off indicator.

Where, O_p is the operation code field,
D is the address type designation field, and
M is the address or operand field

Memory Organization and Addressing:

Core memory consists of 4096 bytes per module. A Wang 3300 can include between one and sixteen 3301 memory modules in the system. Each module is organized into 16 pages of 256 bytes each. The core cycle time is 1.6 μ sec.

There are a total of eight types of addressing:

1. Current direct addressing—Operand address given by:

B-Register	M-field

2. Absolute direct addressing – Operand address given by:

Page bit in S	M-field

3. Current indirect addressing – Operational address found in two consecutive bytes whose address is given by:

B-Register	M-field

4. Absolute indirect addressing – Operand address found in two consecutive bytes whose address is given by:

Page bit in S	M-field

5. Current indirect addressing, auto-increment-identical to current indirect addressing, but the 16-bit operand address is automatically incremented *after* fetching the operand.
6. Absolute indirect addressing, auto-increment-identical to absolute indirect addressing, but with automatic increment of address *after* instruction fetch.
7. Current indirect addressing, auto-decrement – Identical to current indirect addressing, except the 16-bit address is automatically decremented *before* fetch of the operand.
8. Absolute indirect addressing, auto-decrement – Identical to absolute indirect addressing, except with automatic decrement of the address *before* fetching the operand.

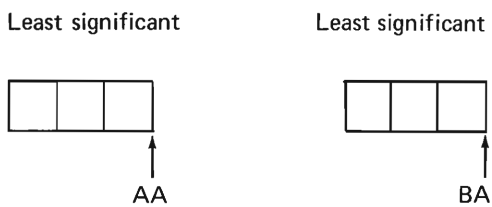
Auto-increment and auto-decrement occur with a count of one or two, depending on whether a single-precision or a double-precision command is executed. These auto-indexing commands provide considerable flexibility in stack processing, and table handling.

The Instruction Set:

The 3300 instruction set is designed to take full advantage of the addressing capability of the machine. The idea is to design the maximum number of instructions which have the complete addressing logic. *Accordingly, there are 21 instructions which can address memory. This is considerably more than the traditional mini-computer.*

The result is a computer whose memory is "open", making it easy to operate on data in the memory. *The programmer does not have to rely on intricate sequencing and manipulation. A relatively simple, generalized instruction sequence will do the job efficiently.*

Consider now an example of addition. Suppose AA and BA are addresses and each points to the least significant byte positions of two 32-bit numbers:



To add the two operands, leaving the result in place of the BA operand, a simple four-instruction sequence suffices:

```
DL  — AA (double load Z and A, auto-
           decrement)
DAM — BA (double add to memory,
           auto-decrement)
DL  — AA
DAM — BA
```

The carry is automatically linked. An instruction to turn off the carry bit may be needed at the beginning if the carry is not initially zero. For data of greater precision, a loop results in another four instruction sequence.

```
LOOP  DL   — AA
      DAM  — BA
      INC    COUNT (Increment the
                    count)
      JNE    LOOP  (Jump if count
                    ≠ 0).
```

We assume that COUNT initially contains the complement of the count.

By adding directly into memory with the DAM (Double Add to Memory), an instruction is saved. This is one example of the powerful memory reference commands available on the machine.

A second example of the flexibility of the instruction set is found in the hardware decimal arithmetic feature on the computer. In fact, the instruction sequences above can perform either binary addition or BCD (binary coded decimal) addition, as determined by a program controlled status bit. Accordingly, the 3300 BASIC uses decimal arithmetic in its user computations. The advantages are twofold. For the user, it eliminates the annoying phenomena of answers such as 2 x 2 = 3.999 . . . For the compiler, the system input and output conversion routines are simplified considerably.

Stack Operations:

Availability of auto-increment and auto-decrement on the Wang 3300 provides a practically unlimited number of push/pop stack pointers. The use of the stack has become a proven technique in many types of software design. The 3300 auto-decrement/auto-increment features facilitate the use of the stack techniques. Since any memory location can be designed as a stack pointer, any number of stacks can be maintained simultaneously. For example, there are 30 stack pointers in the 3300 BASIC compiler.

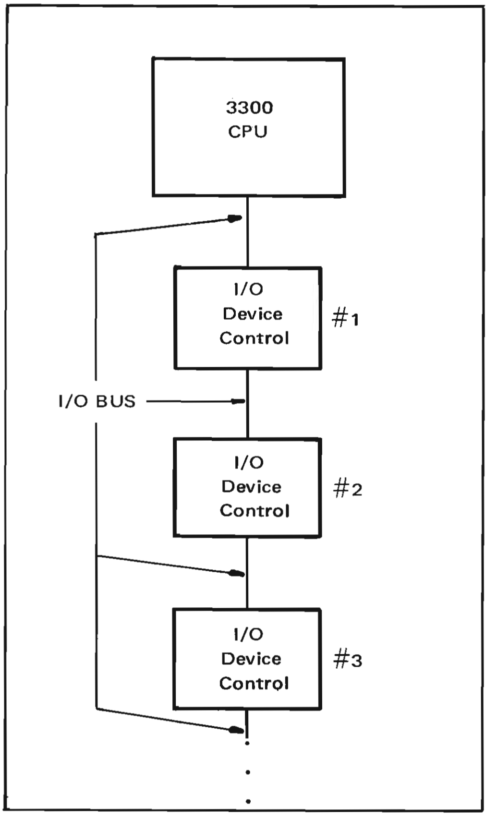
System Communication:

In a reasonably large software system, a common area for communicating between different routines is needed. Naturally, the larger the core area available for this purpose, the more convenient it is to write the software, up to a point. A page oriented minicomputer is restricted to a single page for communication purposes, with the absolute page zero.

The 3300 system provides two pages for this purpose. Page 0 and page 1. The page is selected by setting a status bit. Thus, *512 bytes of core become available as the communication area.*

Input/Output:

The Wang 3300 Input/Output Bus is designed to accommodate both low-speed devices such as the Wang 3315 Teletype and the 3310 I/O writer, and high-speed devices such as a disc. There is a single bus, going into the device control units in a "daisy chain" fashion as indicated below:



Up to 120 devices can be attached to the I/O Bus.

Low-speed devices operate under programmed interrogation by the Central Processor, since it can generate interrupts to the central processor. Interrupt priority is determined by device position on the I/O Bus. The device closer to the CPU has higher priority. Masking commands are provided to disable all device interrupts or to disable selected devices.

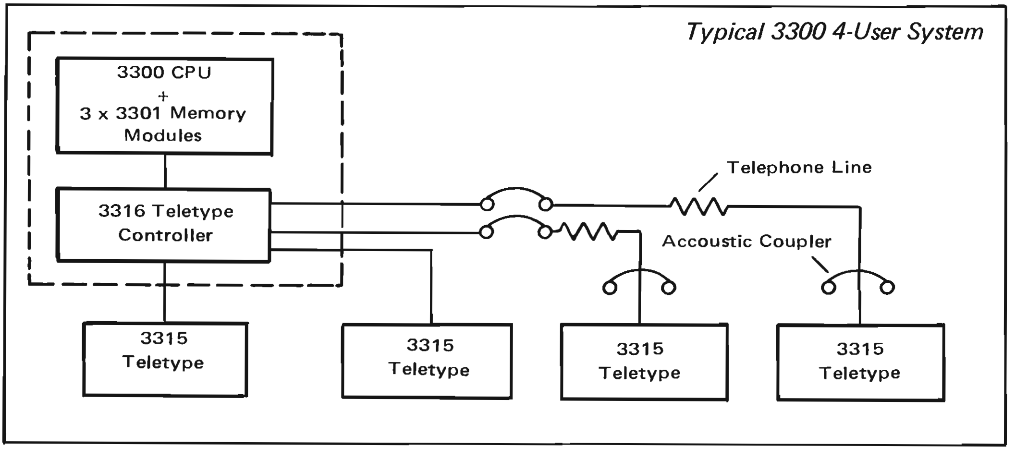
The I/O Bus provides direct memory access to high speed devices at 1.6 μ sec. intervals, *regardless* of the type of instruction being executed. If necessary, a high-speed device may access core memory in between cycles of the same instruction. The programmer no longer needs to avoid the use of long instructions while operating a high speed device.

A feature which results in significant savings to the user is the multi-device control unit. For example, the 3316 teletype control unit can handle up to four teletypes simultaneously. In a multi-terminal system, this approach significantly reduces physical package size as well.

3.3 Summary

There are significant 3300 hardware features which simplify the design of compilers and multi-user systems. These include general features such as an open memory design, as well as stack operations and decimal arithmetic. There are many others such as a flexible set of shift commands, a set of immediate operand commands, and an efficient I/O structure.

These features yield direct benefits to the user in the way of an inexpensive system. There are more indirect benefits in the form of an efficient total system.



SUMMARY OF 3300 HARDWARE	
3300	Central Processor Unit (4K core)
3301	4K Memory Module
3302	Memory Extension Chassis
3310	I/O Typewriter
3311	I/O Typewriter Controller (Controls four 3310's)
3315	33-ASR Teletype
3316	Teletype Controller (Controls four 3316's)
3319	I/O Extension Chassis
3320	Dual Magnetic Tape Cassette Drives
3330	30 CPS Paper Tape Reader

4. DEVELOPMENT OF SOFTWARE

Robert Kolk
Manager, System Programming
Wang Laboratories

Introduction:

With the announcement of the 3300, Wang Laboratories has moved rapidly into the computer field. Although technically the differences between programmable calculators and computers are sometimes quite subtle, the terminology can be dramatically different. Reference is made to "software" instead of programs in the title of this section.

Included at the end of the section is a glossary of terminology which is commonly used in the discussion of computer software. If terms such as compiler, simulator and interactive quickly bring a frown to your face, a short detour to the glossary on pages 20 and 21 will provide a useful aid to reading the following paragraphs.

The development of the System and applications programs for the 3300 has involved an interesting and sophisticated approach. Rather challenging requirements were placed on both the software and hardware development. The initial system program requirements for most mini-computers include such standard items as an assembler, loader, debug package, symbolic tape editor and diagnostics. Since the 3300 system was to be marketed as an integrated hardware/software time-sharing system, the initial requirements also included an interactive Basic Time-Sharing Compiler and an applications program library. It was also important that this programming development effort be done in parallel with hardware development described in section 3 since both were required to meet initial system delivery schedules. Another consideration was to insure that that hardware and software design would meet the performance requirements of a time-sharing system.

IBM-360 at PHI Inc. used as unique simulator:

A combined software development group was organized from programming staff members of both Wang Laboratories and PHI Computer Services – Wang's computing consultant subsidiary. The first effort was to write an assembler and a simulator for the 3300 on PHI's IBM-360 model 65 (see Figure 4.1). *This permitted 3300 programs to be written and checked out under simulation while the 3300 was being developed.* Although an important consideration for going to assembly and simulation on the 360-65 was gaining six months in the software development schedule, a number of other planned benefits were achieved by this approach. The simulator was used as a major design tool for the 3300 computer. A number of bench mark programs were run to evaluate the performance of the 3300 design.

Figure 4.1 *Computer Room at PHI Computer Services, Inc., where IBM-360-65 was used as 3300 simulator in software development.*

As important segments of the time sharing executive system, Basic Compiler, input/output controller and math package were developed and simulated, they proved to be a valuable aid in evaluating the performance of the computer design. Since the evaluation was available at the earlier stages of hardware development a number of important design modifications could be made. For example, the auto-indexing features in the computer were modified to improve the list handling capability for compiler tables. The character comparison capability was extended to handle syntax analysis more efficiently. Modifications were made to the arithmetic and shift instruction sets to improve the arithmetic floating point software performance. The input/output logic was modified to both reduce component logic and improve performance. Conversely, some instructions which did not contribute to overall efficiency were eliminated. The net result was a computer design with an improved cost/performance ratio, and tested operational performance.

Another advantage in using the 360 assembler/simulator was efficiency. With the use of card input, programs could be easily modified and reassembled. A further advantage was gained by storing assembled object programs on disk. By assembling the program once and storing it on an IBM 360 disk, a number of simulation runs could be made with minimum object loading time.

Checkout of a complex program is often more efficient under simulation. When checkout is done in real time on a smaller computer, it is common for program errors to cause the destruction of other parts of the program. In many instances, it is more difficult to find those types of errors, since program execution does not reach planned program halts or checkpoints. When using a simulator however, each instruction is interpretively executed by the simulator program which operates in a protected environment. Since the simulator cannot be destroyed, accurate tracing printouts and snap dumps of memory can be made during and after the errors occur. Accurate information is available to pinpoint the error.

Design Goals:

The design goals for the 3300 Basic system software fall into three general areas:

1. *To provide an in-house time-sharing system with languages and procedures that are relatively easy to use for customers with minimum programming experience.*
2. *To provide programming languages for the system that have a large degree of acceptance in the computer field. and a large amount of software applications already written.*
3. *To provide software for a small computer time-sharing system that can be offered in a variety of configurations, to best meet each customer's need at minimum cost.*

Easy-to-use system:

The first goal, an easy-to-use system, established a number of design parameters. The Basic language was selected for the initial system. In addition to being a popular time-sharing language, Basic is a language which is easy to learn and use. The standard features of original Dartmouth Basic were maintained. Features added to the system and language were strongly oriented toward improving user convenience and learning. An immediate execution mode feature was added to the system. This permits users to enter and immediately execute one line Basic programs. In this mode, the terminal is used essentially as a very powerful calculator. *A novice user, in this mode, can begin to effectively use the system in a very short period. For example, if the user enters the following line:*

```
FOR I = .01 TO 1 STEP .01 : PRINT SIN(I),
COS(I): NEXT I
```

The system will immediately type out 100 values of SINE and COSINE.

The system was designed to perform a syntax check of each program line as it is entered. If an error was made, the system immediately points out an error message pointing to the exact spot in the line where the language format was incorrect. Many systems do not output error messages until the program is run, and then only one at a time.

A typical program line entry and error response is illustrated below:

```
: 200 LET A = B + SIN(C
                        ↑ERROR 12 COL 22
```

The error message indicates that a right parenthesis is missing from the statement and points to the place in the line where the error occurred. An immediate error printout prevents duplication of an error and provides an effective learning tool.

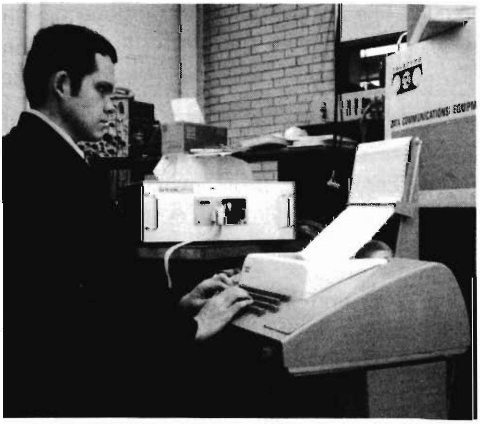
The design of the Basic Compiler is consistent with the overall goal of providing an easy to use and reliable system. The Basic Compiler was designed to be interpretive. That is, instead of actually converting a higher level language like Basic to 3300 computer object instructions and then executing them, each source line is scanned, interpreted, and the required operations are performed directly without conversion. One of the number of advantages to this approach is efficiency in the use of available memory. An important advantage from a user's point is system protection. Since user program execution is always directly under the control of the interpreter, *errors can be detected without possibly damaging other parts of the system program, and thus saving reloading the system.*

Universally accepted program language:

The second goal of providing universally accepted program languages is reflected in both currently available 3300 systems and systems under development. A standard Basic language system is currently developed. An extended basic language which incorporates matrix operations, string variables and extended printing and file operations will be available shortly. In both systems an attempt was made to conform to standard specifications of the Basic language. A Fortran system is currently under development. By providing these languages, a large number of application programs already developed as well as a vast amount of programming experience is applicable to the system. The 3300 Basic system is also provided with an extensive collection of application programs in many fields, and is constantly being expanded by Wang's programming group.

Configuration Flexibility:

The third goal, flexibility in configuration, has also provided a number of design considerations in the software. The available hardware inherently provides the user with a variety of choices as discussed in Section 3. The 3300 system is available with both, as a memory time sharing system and as a disk system. Configurations of one to 16 users and memory configurations of from 12,000 to 65,000 bytes are available. User partition sizes can be varied. Both teletype and selectric terminals are available and can be mixed.



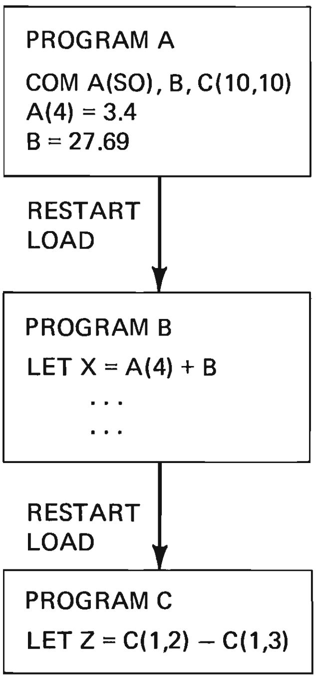
Peter Seymour testing teletype.

Up to 16 cassette tapes can also be added to the system. To support this great variety of configurations a highly modular program design was required. The Input/Output control section of the system was desiged to utilize a device table structure which provides the flexibility to modify configurations. Special terminal characteristics such as a separate line feed character in teletypes are handled in the I/O controller section, and are isolated from the operating system. The time-share swapping procedures are isolated in a highly modular executive section. Partition size extents are set up in an easily modifyable table structure. By utilizing the 3300 assembler written on Wang's IBM 360-65, a new system configuration can be assembled rapidly with merely the replacement of tables in the source card decks.

Additional language and control features:

In addition to program design considerations to provide flexibility, a number of system language and control features also reflect this requirement. For example, *formats for program save/load and file read/write commands in the Basic language have been set up to perform these operations on teletype paper tape, cassettes, and disk. A feature is provided to reallocate partition size dynamically in the system. This permits a user to obtain a large partition at times when other terminals are not being used, and thus run more complex problems.*

Another useful advantage of the Wang 3300 Basic language is the common data feature. This permits data that is defined in one program to be automatically passed on to subsequent programs. For example, in the program flow illustrated below, program A originates several data variables and defines them as common data. These variables are passed on to programs B and C when they are loaded and run.



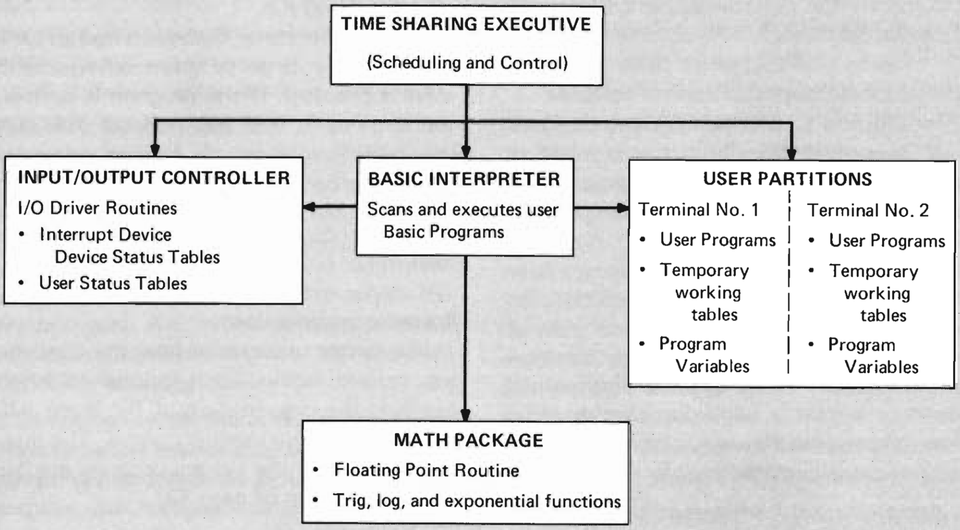
The common data feature is quite useful when a very large program is required to solve a problem. If the program is normally too large to fit into the available user memory partition, it can be broken down into smaller program segments. Pertinent data can be passed between these program segments by using the common feature.

Program organization:

To better understand how the time-sharing system works, it is helpful to briefly examine the organization of the Wang 3300 Basic system operating programs. The major components of the system program are illustrated at the top of page 18.

The Time-Sharing executive provides overall control of the processing. In this program section, determination is made of the user partition to be executed next, and the duration of the execution. Theoretically each active partition will receive an equal portion of execution time on a cyclic basis. However in practice, at any given time, a number of terminals will be temporarily busy receiving or entering programs or data. The information that they are "busy with I/O" is stored in the user status table in the Input/Output controller section. The Time-Sharing executive can therefore look into these tables and observe that terminals are temporarily "busy doing I/O". They will then be bypassed when their turn comes up in the execution cycle until their Input/Output operation is completed. *This bypassing of busy terminals is the most significant advantage of time-sharing systems.*

An area of computer memory or auxillary storage such as disk is reserved for user partitions. Each partition is an area that is assigned to a specific terminal in the system. The partition contains the users program, an area reserved for the program variable data, and areas reserved for temporary working tables generated by the Interpreter when the program is entered and executed. In a



"memory" time-sharing system all partitions reside in the computer memory. In a "disk" time-sharing system all partitions reside on disk and each one is transferred to computer memory by the Time-Sharing Executive section when its "turn" to be executed in the cycle comes up.

The Basic Interpreter is the program section that actually processes and executes each user program. It operates directly on the program that is stored in a user partition. It employs the user partition to store all temporary information necessary for its processing. Therefore, when the Time-Sharing Executive program selects a particular user partition to be executed, it simply tells the Basic Interpreter where the partition is located and transfers to the Interpreter which begins its processing. The Basic interpreter has three major functions. (1) Processing the system commands which are entered by the user terminal, (Such as print the program, load a new program, etc.). (2) Checking each new program and program line for ERRORS, and (3) Executing each program by scanning and interpreting each program line and performing the required operations. The Interpreter stops processing a particular user partition and returns to the Time-Sharing Executive when either a given period of processing time is exceeded, or when the executed program initiates an input or output operation, (such as READ, PRINT). In the later case, execution can no longer continue until Input/Output is completed.

The Input/Output Controller handles all input/output operations, (such as printing, receiving typed input, reading tapes, etc.). When the Basic interpreter encounters an I/O operation while executing a user program, or when it requires another program line to be entered, it transfers to an I/O Driver routine in the I/O Controller. Here the required operation is set up and initiated. Appropriate data is entered into status tables for both the device and user. Information such as what operation is being performed, how it is completed, and what user is temporarily busy with I/O is entered. Two examples are: (1) print a message starting at a specified location and containing a specified number of characters, and (2) receive typed characters for a specified terminal, and save them until a carriage return character is received. Once a requested input or output operation is initiated, system control is transferred back to the Time-Sharing

Executive, to start the execution of another partition. The input/output operation is then completed by the Interrupt Routine. For most devices each time an Input/Output device completes a single operation, (such as printing a character), a hardware computer interrupt occurs. This causes the current computer processing to be interrupted. Transfer is then made to the interrupt routine. The interrupt routine determines what device causes the interrupt. It can then refer to that device's status table to find out what operation is being performed and how it should be continued, (for example print the next character). If the I/O operation is complete, the user status table associated with the current Input/Output function can be reset to active, thus allowing the user to be put back into the time sharing execution cycle. The interrupt routine returns to the interrupted processing after setting up the next I/O operation. Therefore with this mode operation, user programs are executed most of the time. The execution is interrupted, occasionally for short periods, when input/output interrupts occur. During these short periods, an additional read or write is set up to continue requested Input/Output operations.

Math package:

The final topic to be covered in this discussion of the 3300 software development is the math package. *The 3300 computer provides both decimal and binary arithmetic operations.* This is a significant advantage over purely binary computers where conversion between binary and decimal and binary coefficient roundoff causes both inaccuracy and inefficiency. A prime consideration for the math package was the specification of a standard floating point number format. The final selection was a format with 8 digits of mantissa which are contained in decimal form in four computer bytes and a one byte exponent which is maintained in binary to express an exponent of from 10^{-64} to 10^{+64}. This format provides sufficient accuracy in a wide variety of applications and also is relatively efficient to use. Another interesting decision to be made, was the method to be used in the calculation of the various trig, log and exponential functions. For the 3300, one of the most accurate methods, digit by digit, also turned out to be one of the most efficient since it results in series of additions and subtractions. Using this method, a full eight digits of accuracy is generally obtained for the functions. A fascinating sidelight that resulted in the math package was that it was extremely modular for extended precision. The 3300 with decimal arithmetic is inherently multi-precision. The techniques used for trig and log function were also multiprecision in nature. Therefore if 10 or 20 digits of precision instead of 8 were desired, changes would be relatively straightforward.

Janette Weber, dwarfed by mountain of teletypes.

ASSEMBLER

A computer program which reads and converts programs that are written in assembly language to their machine code equivalent. For all computers, the final format of instruction is a combination of alphanumeric codes. One part of each instruction is a code which defines the instruction (add, load, store, jump); another part defines the address of where the number of data to be operated on is located.

To make programming less tedious, and enable improved error checking, programs are written in assembly language. In this language, alpha-numeric mnemonics are used in place of machine instruction codes and addresses. Mnemonics are assembled into machine code by an assembler. Generally, each line of an assembly language program represents one computer instruction.

For example:

Function	Address	
LA	NUMB1	(load Accumulator with Number 1)
ADD	NUMB2	(Add Number 2 to it)
UA	NUMB3	(Unload result into Number 3)

Might be assembled into:

Instruction Code	Address	
C2	90	(where C2 is the machine
82	92	code equivalent of the
CA	94	load accumulator instruction, and 90 is the address of NUMB1, etc.)

COMPILER

A computer program which converts higher level language programs into a computer machine code program which can be subsequently loaded into the computer and run. Generally each line of the higher level language results in a number of machine code instructions (whereas in an assembler, each assembly line produced one computer instruction.). Languages such as Basic, Fortran, and Cobol are typical higher level languages. For example the following Basic program line might be compiled as illustrated.

100 LET X = Y + SIN(Z)

Compiled Program:

Code		Assembly Language Equivalent	
C2	84	LA Z	(load accumulator with Z)
4E	90	JST FSIN	(call sine routine to calc sine of Z)
82	AO	ADD Y	(Add Y to result, (sine of Z))
CA	86	UA X	(Unload final result in X)

SYMBOLIC TAPE EDITOR

A computer program that facilitates the preparation and punching of a source computer program on tape. It allows the user to enter and hold a number of program lines in the computer. They can then be listed and individually corrected. When they are all correct, they can be punched or written onto tape in a block.

INTERPRETER

A computer program which performs a combined compilation/execution operation on programs which are generally written in a higher level programming language. A true compiler converts a higher level language program to a sequence of machine code instructions called an object program which can be subsequently loaded and run at any time. An interpreter never makes the conversion to machine code. Instead it operates directly on the program in its higher level language form, interprets it, and performs the required operations via a series of subroutine calls. For example, with the Basic program line:

100 LET X = Y + SIN(Z)

The interpreter would first scan the entire program and save space for the variables such as X, Y, and Z. Then, during execution, when it encountered this line, it would retrieve the current value of Z, evaluate its sine, add the current value of Y to the result, and store this result in the location saved for X. The net result of the line would be a new value of X.

SOURCE LANGUAGE

The form a program is written in before it is assembled, or compiled. For example the following source line of a Basic program would be considered to be written in a source language; it would be part of a source program:

100 LET A = B + C * SQR(D)

OBJECT LANGUAGE

An object language is the computer machine code that a program is converted to after it is assembled or compiled. Typically an object program is produced on paper tape, magnetic tape, or cards. It will also contain information specifying where the program is to be loaded in computer memory. Object language format might appear as follows:

0100	(starting address to load program)
C290	(1st machine code instruction)
8292	(2nd machine code instruction)
CA94	(3rd machine code instruction)

DIAGNOSTIC

A diagnostic is a sequence of programmed instructions which tests a hardware or software component and verifies it is operating properly. Examples of diagnostics are tests of the computer instructions, memory, paper tape reader, or typewriter.

A diagnostic normally has two operating elements: (1) a test phase and (2) a failure mode. The test phase exercises a component either once or repetitively. If the results achieved differ from those expected, the program enters the failure mode.

SIMULATOR

A computer program which is written to run in a computer and completely simulate or duplicate the operations of another computer. The simulator program would input and operate on an object program which would normally be loaded into the computer that it simulated. The simulator program analyzes each machine code instruction and performs the operation normally performed when the instruction is actually executed in the simulated computer. In addition to simulating the computer, many simulators also simulate input/output devices by using I/O devices on the computer they are being run on to perform similar operations.

REAL-TIME (ON-LINE)

The use of a computer in a manner whereby the programs it is executing produce an immediate result or action for the user. For example, a computer and programs controlling processes in a chemical plant would continuously monitor various input signals, analyze them, and output necessary control signals as the process continued. This would be real-time computer operation. A calculator or time-sharing terminal can be considered to be running in real-time mode, since program results are displayed or printed back immediately. Non-real time operation of a computer occurs when a number of computer program runs are grouped together, loaded and run at some later time. This is called batch processing.

TIME-SHARING

A mode of computer operation in which a number of independent computer programs can be run or operated in the same time period (although not at the same instant) in a single computer. Each individual program is stored in a different section of the computer memory, or on auxillary storage such as a disk. Each program is only allowed to execute for a limited period of time, and then a segment of another program is executed. This sequencial cycle continues for all programs currently active. In most computers, input/output processes are much slower than internal computer execution speeds.

DEBUG PACKAGE

A debug package is a set of programs that facilitates checkout of an applications program. Debug elements include capability to examine and modify portions of the applications program, trace instruction logic, test different program paths, and print portions of the memory and registers.

PARTITION SIZE

In a time sharing system, partition size refers to the maximum allowable space allocated in the computer to hold each user's program and data. If a user's program exceeds this limit, it cannot be run and his processing is usually terminated with an error message.

INTERACTIVE

A term which in computer terminology refers to a manner of operating a computer. When a computer system operates in an interactive mode, the user can enter information directly into the computer and receive an immediate response to most information he enters. The computer "interacts" directly with the user. For example if a user has just entered a program line or a system command, the computer will immediately type back a response acknowledging that it has received the information and it is correct. If it is in error, an error message would be typed back immediately.

INSTRUCTION

An instruction is a single internal computer command. For example, ADD, LOAD A REGISTER, JUMP.

SUBROUTINE

A subroutine is a set of computer instructions which together perform a given task. For example, calculate a SINE or COSINE. The subroutine may be called at any point or time during the execution of a program. The general purpose of the subroutine is to efficiently perform repetitive sequences of calculations.

PROGRAM

A program is that set of internal computer instructions and subroutines which together comprise a definable computer job or task.

TIME-SHARING EXECUTIVE

The program that controls the overall processing within the computer is called the executive. The executive schedules tasks to be performed based on planned constraints such as priority, allocation of space, and data availability.

In time-sharing, several tasks are performed at the same time so that overall computer utilization is maximized. A time-sharing executive controls the operation of tasks which may be running concurrently in different parts of the computer, and determines what is to be done next, and for how long.

INPUT/OUTPUT CONTROLLER

The input/output controller program operates subordinate to the executive handling all input and output actions. By centralizing input/output handling the efficiency of peripheral device processing and thus overall computer utilization can be optimized.

The input/output controller handles all commands for input and output transfers as well as device peculiar code conversions and hardware interface considerations.

LOADER

Computer programs that read an object program into computer memory from tape or cards, and verify that it properly loaded.

5. SURVEY OF OVERALL CAPABILITY OF THE 3300

The following twelve pages contain reprints of eight articles which appeared in recent publications describing Wang Laboratories' new 3300 Basic Timesharing System. Although these articles contain some repetition, they supply the reader with a description of the 3300 system from several different viewpoints and thereby stress different features of the system. We hope they will be of use to our readers who are interested in an overview of this new Wang product.

THE WANG 3300 BASIC SYSTEM

Will 1970 be "The Year of the In-House Time-Sharing System?"

Having initiated 1970 by extolling the virtues of shared in-house BASIC systems, we were fully prepared to wager that another half-dozen or so of these systems (presently ranging from $40K to $60K for a 4-terminal configuration; providing for up to 12 additional terminals at $1500–$3000 per) were already "on the pike" and scheduled for announcement by the fall (FJCC) of this year. Our expectations were based on our beliefs that **1.** the market is "clear and present"; **2.** the field is wide-open competitively (at the time we knew of only two systems which were being actively marketed); **3.** the state-of-the-art is well within realization; and **4.** $40K–$60K is not unreasonable for a flexible BASIC system geared to handle up to 16 user terminals.

So much for our expectations. As of this month, Wang Laboratories, Inc. will offer an up-to-16-terminal BASIC system that will sell in the $25K range for a typical 4-user configuration, including Selectric terminals and cassette drives. As a result, while we are even more convinced that 1970 will be "The Year of the In-House Time-Sharing System," we have revised downwards our expectations as to the number of companies that will be offering competing systems in the near (12 month) future.

HARDWARE

The heart of Wang's Time-Sharing System is a spanking-new 4K–64K (bytes), 1.6μsec. processor designated the Wang 3300. In its minimum 4K configuration, the TTL/MSI 3300 sells for $4,950, making it the first g-p mini priced under $5K. Add-on memory modules are available in 4K increments for $2,500.

These low prices in large part reflect Wang's choice of an 8-bit word, but the decision to use a smaller word size has had proportionately little effect on performance. An extensive repertoire of 72 instructions includes 20 for memory reference alone and five single and double arithmetic instructions which operate in either binary or decimal mode. Instruction logic is therefore equivalent to many 16-bit machines. Additional hardware instruction features include a complete and compatible set of bus and channel I/O, interrupt, and skip instructions; the ability to permit I/O, the ability to permit I/O direct memory access cycle-stealing between instruction execution sub-cycles; and a unique, push-pop auto-incrementing and decrementing indirect addressing feature throughout memory for both single and double instructions.

A standard 3300 time-sharing configuration consists of one 4K processor ($4,950); two additional 4K core units ($5,000); two Selectric I/O terminals ($4,200); a terminal control unit ($500 – controls up to four Selectrics); a pair of magnetic tape cassette drives ($1,400); and the BASIC software package which includes a one-time set-up and initialization fee ($1,500). Total: $17,550. Assuming individual terminal users are content with 1400-byte partitions (partition size is determined at set-up time), a 4-user Selectric/cassette system would cost $25,650. Two- and four-user TTY systems are priced at $15,250 and $21,250. Goodies include an acoustic coupler ($795) for use with the TTYs, and still-to-be-priced 65K and ½-megabyte disks.

SOFTWARE

Since the BASIC compiler was simulated on an S/360-65 while the hardware was still in development, the resulting system has all the advantages of a happy hardware/software marriage. Significant features of Wang BASIC are:

- **An "Immediate Mode"** – unnumbered BASIC statements are immediately executed in calculator fashion.

- **A "Diagnostic Pointer" and "Trace Mode"** – for debugging and error checking.

- **Common Data Allocation** – variables can be retained between chained programs to allow more efficient use of allocated core.

- **Basic Prime and Tab Extensions** – for plotting functions.

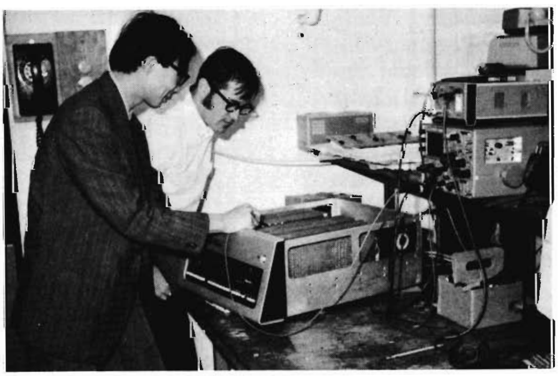
- **Terminal Convenience Features** – such as the ability to save and load programs via high-speed cassette tapes, and to modify and copy statements on a character basis.

These features, along with Boolean logic functions, unlimited subroutines, and a full 8 digits of decimal accuracy (data may have exponents between 10^{-64} and 10^{-64}) should be more than enough to handle the needs of most present time-sharing users.

SUPPORT

Wang is certainly geared to produce, sell, and support these turnkey systems. The company is a most efficient manufacturer (last year Wang generated over $23 million in revenue with only 55,000 sq. ft. of plant space; an additional 85,000 sq. ft. facility is under construction), and has an excellent sales and marketing track record in the desk-top calculator field. In addition to what the company already provides in its extensive BASIC program library, Wang can draw upon the resources of Philip Hankins Inc., acquired in 1968, for systems and programming support.

Dr. An Wang, the company's president and treasurer, summed up Wang Laboratories' capabilities and support philosophy very neatly: "As the leader in sophisticated calculators, it was logical to enter the small computer market. We have our traditional strength in hardware, an excellent software group, and an extensive field sales and service organization. Not only can we develop a total system, we can also support the user." We don't doubt it.



Ned Chang (left) and Paul Congo testing a 3300 prototype.

Wang Labs enters mini-computer market

Wang Laboratories, Inc., Tewksbury, Mass., has gone into the computer business. First product of the new venture is a time-sharing mini-computer, the Wang 3300. Wang, a major U.S. maker of programable electronic calculators, says that its Wang 3300 BASIC system—which can accommodate as many as 16 users simultaneously—was designed to fill "the growing gap between expensive time-sharing terminals and our most powerful desk top calculators."

The firm claims that the 3300 BASIC, with prices starting at $15,250, is "the lowest cost and most easily operated mini-computer time-sharing system available." The system uses modified IBM Selectric typewriters as primary input/output (I/O) devices, but can also work with standard teletypewriter terminals. Programs are stored on standard magnetic tape cassettes, with high-speed disk storage soon to be available as an extra-cost option.

Heart of the system is the Wang 3300 central processing unit, an eight-bit general-purpose computer which uses TTL (transistor-to-transistor logic) integrated circuits and MSI (medium scale integration) circuitry techniques. It has a standard 4096-word, 1.6-microsecond full-cycle memory, expandable to 65,536 words.

Most mini-computers have only three to 10 index registers, Wang says, but the 3300's core memory incorporates a push-up/pop-down indirect addressing system which allows the entire core to serve as index registers. This mode of address not only enhances programing efficiency, the maker states, but also enables the 3300 to execute programs previously considered impossible for eight-bit computers.

Repertoire. The Wang 3300 has a repertoire of 72 instructions, including 21 memory reference instructions with both single and double byte operands. There are also five single and double arithmetic instructions which operate in either binary or decimal mode. Although the 3300 is nominally an eight-bit computer, use of double memory reference and arithmetic instructions provides instruction logic equivalent to that of many 16-bit machines, Wang claims.

Wang 3300 uses BASIC software

Verb repertoire:

COM	GOSUB	PRINT
DATA	GOTO	READ
DEF	IF	REM
DIM	INPUT	RESTORE
END	LET	RETURN
FOR	NEXT	STOP
TRACE		

Function library:

SIN	LOG	SGN
COS	ABS	AND
TAN	SQR	OR
ATN	INT	BOOL
EXP	RND	

Command repertoire:

LIST	RERUN	START
¢*	LOAD	RESTART
RUN	SAVE	

* Partial statement correction.

Wang offers a choice of I/O devices. Standard teletypewriter terminals are the least expensive. Or, for higher speed (15 characters per second) and quieter operation, Wang will furnish its own I/O writer consisting of a modified IBM Selectric typewriter with character buffer and BASIC-compatible character set. As many as 16 I/O terminals can be connected to the system. With teletypewriter terminal and an acoustic coupler, the computer can be operated from remote locations over standard telephone lines.

Cassettes. For secondary storage with the I/O writer, Wang has developed a high-speed (300 characters per second) tape cassette driver capable of operating two magnetic tape cassettes (the same kind used for audio purposes). Wang says that the magnetic tape cassettes offer a combination of compactness, ease of storage and handling, low cost, and durability unmatched by any other storage device. A single system may contain multiple cassette drivers.

For users who require still greater storage, several disk storage devices, including a low-cost 65,536-word disk and a high-speed 0.5-megabyte disk, will be available soon. There are few restrictions on the use of these high transfer rate peripherals, the firm says, since the central processor design permits I/O direct memory access cycle stealing between instruction execution subcycles.

Software for the 3300 uses the BASIC language (see box). BASIC was chosen, the company says, after marketing field studies indicated that BASIC was the most popular conversational time-sharing language. The similarities of BASIC to everyday English make it simple, Wang asserts, for even the beginner to accomplish significant programing results during his first hour at the terminal.

Simulator. Wang engineers developed the software with an IBM 360/65 simulator while still working on the hardware.

"By using simulators," software applications manager Bob Kolk says, "we not only completed software development on an accelerated schedule, we also noted some unique characteristics that later were incorporated into the hardware design."

The company claims several advantages for its version of BASIC:

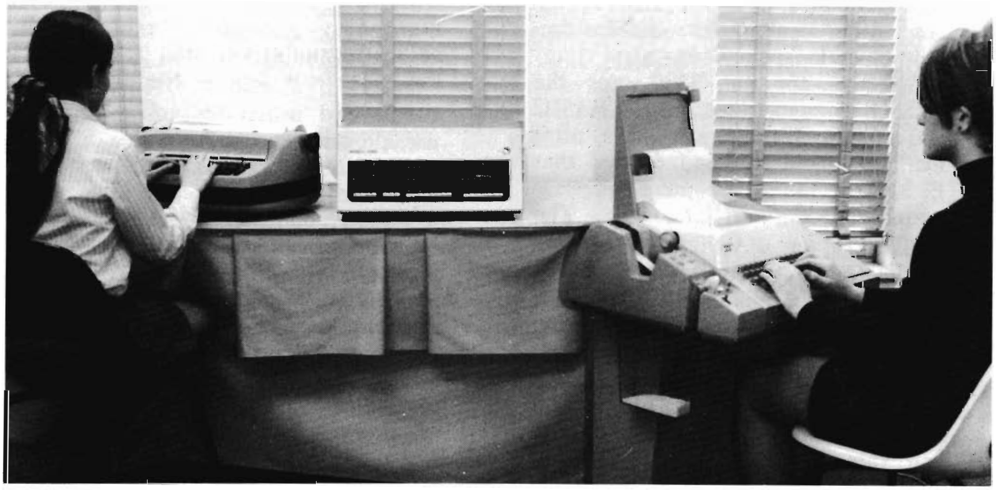
- An "immediate" mode converts the BASIC system to a conversational calculator. Entry of an unnumbered BASIC statement causes immediate execution, even during a program operation.
- System save and load commands allow the user to store and load BASIC programs conveniently on cassette tapes, thus permit him to accumulate library programs without costly disk allocations.
- Floating point format is used for all variables, with eight digits of rounded decimal precision. Data may have exponents between 10^{-63} and 10^{63}. Arithmetic calculations are performed in decimal mode. Boolean logic functions are included.
- Several terminal entry and diagnostic features make for convenient operation. Diagnostic error messages point to the exact column in the BASIC statement where the error was detected. Syntax diagnostic messages are printed immediately after each line entry. A small change command (¢) permits correction of one or more columns in a statement without retyping the entire statement, and also permits copying existing statements with new statement numbers. The user may stop program execution and system output at any time by pressing the ATTENTION key on the terminal.
- Several provisions are made for user program checkout. Trace ON and OFF are incorporated as BASIC statement verbs to permit selected traces. A RERUN command permits program execution starting in the middle of the program, with current variable values maintained.
- Compiler design makes optimum use of user partition area for text and tables. Except for ultimate partition size, there are no physical restrictions on the number of allowable nested loops, nested subroutine calls, or nested functions. Partition size is modularly expandable, as are the verb, command, and function repertoires.
- Provision for common data allocation (COM) allows each user to retain variables between chained programs and make more efficient use of his allocated core space.

Standard support software such as assembler, loader, and diagnostics are available without extra charge to the 3300 purchaser. Complete applications systems software will be furnished for an initial charge and a monthly use and maintenance charge.

Library. A standard accessory of the 3300 BASIC system is a program library abstract service, which lists programs written and maintained for the system. System users can buy these library programs on tape cassettes "at a nominal cost," Wang says. Programs are available in the fields of mathematics, statistics, business, finance, physics, chemistry, mechanics, electronics, thermodynamics, structures, surveying, and games. An extra dividend, Wang points out, is the extensive BASIC software developed in the past five years.

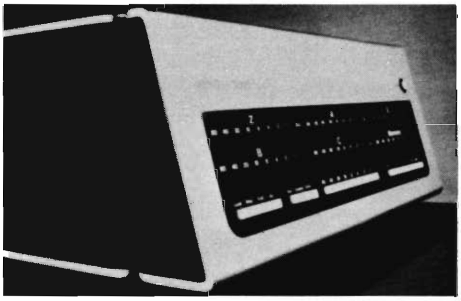
System prices begin at $15,250 for a central processor and two teletypewriter terminals. Most comparable in-house time-sharing systems would cost about $65,000, Wang says. For less than $20,000, the firm adds, a user can purchase the central processor, several terminals, cassettes, and "all the software needed to get started." Or, he could lease the same package for less than $600 a month.

"The 3300 system was designed to be used for computing answers, not process control," says Wang's marketing manager John Cunningham. "Initially, our market will be people who otherwise would have to use a calculator or a terminal." According to Wang, marketing studies have indicated potential customers in high schools, junior colleges and small universities, individual departments of larger universities, and engineering and research departments of both small and large corporations. Primary areas of concentration will be in statistics, mathematics, engineering, finance, biomedicine, and the sciences.



Wang 3300 BASIC uses either standard teletypewriters or modified IBM Selectric typewriters for input/output, accepts as many as 16 simultaneous users.

Time-Sharing Minicomputer Designed Around Software



Wang 3300 Basic

By Frank Piasta
CW Staff Writer

TEWKSBURY, Mass. – The Wang 3300 Basic, a low-cost time-sharing computer system, was tailored during development to accommodate software requirements, according to Wang Laboratories Inc.

The software packages for the system were developed by PHI Computer Services, a subsidiary of Wang, by simulating the 3300 on an IBM 360/65. Some of the problems encountered were solved by the novel approach of changing the hardware design, the firm said.

As its name implies, the 3300 Basic uses the Basic time-sharing language as the prime means of communications. This results in what Wang calls "the most easily operated minicomputer time-sharing system available."

The central processor, an eight-bit computer called the Wang 3300, features 1.6 μsec cycle memory. From a minimum of 12K words, the 3300 is expandable in 4K increments to a maximum of 65K words. A repertoire of 70 instructions and over 20 memory reference commands include decimal and binary arithmetic capabilities.

Also featured in the processing unit are a push-down stack addressing system that permits every core position to be used as an index register, priority interrupt, or both character bus and direct memory access I/O, according to Wang.

At the present time, peripheral units are limited to terminals, of which several are offered. The most economical unit is the teletypewriter terminal. A 15 char/sec I/O writer, a modified Selectric typewriter with character buffer and Basic-compatible character set, is also available.

For secondary storage with the I/O writer, Wang developed a high-speed tape cassette drive capable of operating two magnetic tape cassettes. The Wang 1103A acoustic coupler may be used with the Wang 3315 Teletype terminal over standard telephone lines. A total of 16 terminals can be supported by one 3300 system.

Wang has indicated that a disk-storage device will be available for the 3300 in the near future. CW has learned that this will be a five-million byte system to be built by Digital Information Storage Corp. Wang is a part owner of this supplier. The disk units will probably be announced in about six months, according to Wang.

Other software available for the 3300 includes an assembler, and utility programs such as debugging aids; and printing and plotting routines.

A typical 3300 system, including a 12K processor, two teletypwriter terminals, and the Basic language, costs $15,250.

Wang Laboratories Inc. are located at 836 North St.

Computer has calculator

Eight-bit time-sharing system aimed at in-house applications serves 16 users simultaneously, performs decimal arithmetic

By James Brinton *Electronics* staff

When the price and capability of a company's products bump the bottom of the next higher order of machine, there's a strong likelihood that a new line will evolve. Wang Laboratories Inc., long a major competitor in the calculator field, pushed its programable machines upward into the minicomputer market, both in price and capability. Its model 700 was less a calculator than a microprogramed, time-sharing, special-purpose computer.

So it should surprise no one that the firm is now going to market with an 8-bit computer system capable of handling up to 16 users simultaneously. And, though the model 3300 is aimed at engineering, scientific, and industrial applications, its software library includes programs that make it a true general-purpose computer.

Though the move was a simple extension of its market, Wang looked before it leaped into design. The company's market researchers found that:

▶ Almost 75% of the tasks demanded of a costly time-sharing system could be performed with a calculator; but users were willing to pay for that extra 25% flexibility gained by time-sharing.

▶ Despite all the programing languages available, more than 40% of users adopted Basic because of its simplicity and similarity to mathematical expression.

Time-sharer. Computer for in-house systems uses Basic language for scientific, engineering, and commercial applications.

▶ Most time-sharing users were paying an average of $9 to $11 per hour for port time—plus costs for telephone lines, and storage of programs and data bases. The lowest rate was about $5 per hour.

The 3300 system is designed to sell at a price low enough to fit the in-house time-sharing market, competing effectively with service bureaus. The 3300 costs $15,250 up. It also competes in price with other in-house time-sharing systems like those built around the PDP-8L, which at about $20,000 minimum is the lowest priced of Wang's competitors, though it serves fewer users.

Beyond this point, costs escalate; the time sharing PDP-8I runs from about $40,000 upward, and typical systems built around Hewlett-Packard's HP-2000 cost about $80,000 minimum.

In a four-user configuration, the 3300 costs $21,250. This, in effect, prices each of its teletypewriter terminals at $5,312.50. Hewlett-Packard's HP-9100 programable calculator costs $4,400—plus $975 for a companion printer—for a total of $5,375. Wang's model 700 costs a basic $4,900. Thus a time-sharing general-purpose computer system can cost less per terminal than a set of calculators.

Leasing a 3300 system for less than $600 monthly, a user will pay only about 75 to 80 cents per hour for port time, estimates John F. Cunningham, 3300 marketing manager. "That's a tenth the cost of service bureau time," he says, "and there are no telephone or other charges to add." Thus, Cunningham claims, the 3300 is a good buy even though its data storage capabilities are more limited than those of a service bureau, and its overall capabilities—like those of other in-house systems—encompass only 50-60% of a service bureau operations.

Finally, the 3300 system uses a modified Basic that permits the user to do mathematical operations as easily as with a calculator, to use a teletypewriter or Selectric output as a stand-in for a plotter, and perform business-oriented operations, as well. Wang already has on the shelf about 200 calculation programs and expects to add another 200 to 250 soon, covering 50 "unique disciplines," Cunningham says.

Wang's software is "unbundled." The company charges a $1,500 fee for the Basic compiler and for system setup. Thereafter, users pay $10 per month per terminal to cover the cost of a program library maintenance and updating program. Robert S. Kolk, software applications manager, says that the availability of up-to-date, relevant programs has been a shortcoming of other in-house time sharing systems, and the library policy should circumvent it. Users are to be mailed a list of the library's contents regularly, and the programs themselves are to be available at from $5 to $10 per cassette. "We just want to ask enough to pay for labor and materials," says Kolk.

Unique? The 3300 can do both binary and decimal arithmetic, something perhaps unique in minicomputers. Kolk says the decimal capability allows computations more accurate than those possible with 16-bit computers operating at double precision.

Wang stores numbers in five byte locations, one giving binary notation, and the other four bytes each storing two decimal digits. "We're willing to pay the premium in core space this costs us in return for the increased accuracy," says Kolk.

But Wang's use of memory probably makes the penalty easy to bear. Core expands from the basic system's 12,288 words in 4,096 increments to a total of 65,536—twice the amount possible with most minicomputers.

For second level storage, Wang plans to offer disks as well as its present cassette deck line. A dual cassette deck is offered for use with the 3300 at $1,400; pricing hasn't been set for the disk system, although it is said to be patterned after the fluidically controlled DDR-1 of the Digital Information Storage Corp. of Berlin, Mass. [*Electronics*, Oct. 13, 1969, p. 149].

Close fit. The 3300's cpu is a simple-seeming mainframe which manages to accomplish fairly impressive tasks through its close fit with the Basic software package.

The processor appears to be small; it has only one accumulator register, a single accumulator extension register, and one addressable status register, for example. But, in its handling of lists, loops, and nested subroutines, it is more like DEC's PDP-11 in that all its core addresses are open for use as temporary registers—and up to 65K of core can be accommodated.

Also, though the processor is an 8-bit machine, it is said that use of double memory reference and arithmetic instructions provide instruction logic equivalent to that found in many 16-bit machines. Its addressing modes include immediate addressing that lets the cpu shift into a calculator emulation mode when the Basic line number is omitted, and indirect addressing, which allows auto-incrementing, auto-decrementing, and push-down stack uses with either 1- or 2-byte instructions.

Accenting the simplicity of software operation are features like automatic error detection: if an error in syntax, or an illegal operation or an overflow appears, the next line printed out points to the exact location of the error in the statement above and tells its column number and the type of error made. Also, the user can put more than one Basic statement on a line; the 3300 saves them as entered for ultimate execution. In addition, statements can be easily corrected: if the user catches his own error, he can backspace to erase it, or—using the appropriate symbols—selectively correct individual characters within a statement.

Built-in check. Perhaps most important to the new or inexpert user is the Trace feature built into the 3300 system. The Trace mode allows the computer to follow the programer through his entries. Also, whenever a variable receives a new value during program execution, the Tracer mode reads back data. This read-back ability proves useful when a program transfer is made to another sequence of statements, such as when the computer is diverted to a new subroutine by Fortran-like IF-THEN, or GO-TO branching commands.

Delivery is 90 to 120 days.

Wang Laboratories Inc., 836 North St., Tewksbury, Mass. 01876

New general purpose microcircuit digital computer developed by Wang Laboratories, Inc., is the heart of a low-cost time-sharing system designed to fill the gap between desk-top calculators and large-scale time-sharing systems. The central processor designated the Wang 3300, is an 8-bit machine, offering 1.6-microsec. full cycle memory in 4K units, expandable to 65K, binary and decimal arithmetic, a repertoire of 70 instructions and a choice of input/output terminals. Standard Teletype 33ASR units may be used for input/output stations, and Wang also offers a 15 character/sec. modified typewriter with character buffer and basic language compatible character set.

EQUIPMENT & EDP

Edited by PHILLIP BROOKE

Wang Labs Offers Low-Cost Mini-Computer Time Sharing System—Model 3300 BASIC

TEWKSBURY, Mass. — Wang Laboratories Inc. has introduced what it calls the lowest cost and most easily operated mini-computer time sharing system on the market today.

Billed as the Model 3300 BASIC, the Wang system begins at $15,250 for a central processor and two teletype terminals, including BASIC, compared to approximately $50,000 for other low cost time-sharing systems.

Extending the features of the highly popular and simple BASIC language, the Wang system accommodates any number of terminals up to sixteen. With the Wang Model 1103A acoustic coupler, the user may operate from remote locations using the Wang 3315 Teletype terminal and standard telephone lines.

The central processor, an 8-bit computer called the Wang 3300, features 1.6 microsecond full cycle memory in 4K units expandable to 65K, and decimal arithmetic, a repertoire of 70 instructions, over 20 memory reference instructions, push/pop addressing system, both character bus and direct memory access 1/0, and priority interrupt, the company says.

Unlike most manufacturers, Wang offers a choice of terminals. For greatest economy, the user may select standard teletype terminals as input-output devices, but Wang also developed a 15 character per second 1/0 writer; this is a modified typewriter with character buffer and BASIC compatible character set. The high speed, excellent reliability and low noise levels of typewriters, combine efficiency and simplicity in the Wang terminals. For secondary storage with the 1/0 writer, Wang developed a special high tape cassette driver capable of operating two magnetic tape cassettes. Magnetic tape cassettes combine compactness, ease of storage and handling, low cost and durability unmatched by any other storage device. A single system may combine multiple cassette drivers. Anyone requiring still greater storage, however, may expect several disk storage devices from Wang in the near future.

Aside from the uniqueness of the total system, the Wang 3300 possesses several features that make it stand alone even as a mini-computer, the company states.

For example, the Wang 3300 incoorporates a push up/pop down addressing system in the core memory. This means that, unlike most mini-computers with only 3 to 10 index registers, the entire 3300 core serves as index register, thus conserving program steps, eliminating saves and stores, and increasing programming efficiency.

The addition of double memory reference commands lets the Wang 3300 readily execute programs previously considered impossible for an 8-bit computer. The Wang 3300 offers both the standard binary arithmetic for internal computations and a decimal mode for eliminating base conversions, again enhancing program efficiency. Most systems utilize only binary mode.

A list of special memory instructions represents another hardware feature of the Wang 3300. With over 20 memory reference instructions, the inexpensive 3300 central processor handles sophisticated software such as the BASIC language. Most similar systems have only 5 to 10 memory reference instructions.

Software for the 3300 is the popular and easily learned BASIC language. Wang engineers developed the software with a 360/65 simulator while still perfecting the hardware. By using the simulator, Wang not only completed the software on an accelerated schedule, but also realized major engineering cost savings. The savings, in turn, are passed on to customers in a singularly low cost system that offers full computer power. Other software features include powerful debugging and plotting commands.

The Immediate Mode feature of the software enables the user to have the computer function as a powerful desk-top calculator. By simply omitting a line number from the BASIC command, the operator automatically switches the computer into this mode of instant execution and generates an immediate response. Since the Wang 3300 system allows multiple statements on a line, the user may run sophisticated calculations without writing an entire program.

The Wang 3300 not only incorporates state of the art features found in most BASIC systems, but introduces several others.

"Aside from the actual hardware and software features, the choice of terminals, expandability of the system, ease of operation and computing power make the Wang 3300 BASIC the most sophisticated and least expensive time-sharing system available today," a Wang spokesman says.

For further information contact: Wang Laboratories, Inc., 836 North Street, Tewksbury, Mass. 01876.

MINI TIME-SHARING

Now one of the big calculator manufacturers has crossed over the thin line and built a mini time-sharing system based on a real live computer, the Model 3300. A byte machine, the 3300 uses an 8-bit word size which can be expanded by double precision software, a core memory with a 1.6 usec cycle time which can be ordered in sizes from 4K bytes to 65K, built-in binary and decimal arithmetic, a 72-instruction set with 21 memory reference commands, a push-down/pop-up address scheme, and a direct memory access channel.

Input stations are IBM Selectric I/O writers or ASR 33 Teletypes, and the cpu and its time-sharing executive can handle up to 16 of them. (Response times are expected to be in the two to five second range for up to eight simultaneous users running "average" t-s programs.) A standard part of each terminal is a dual magnetic tape cassette drive rated at 300 cps. Acoustic couplers are offered as options, and two discs, one 65K low cost unit and one .5 megabyte high speed unit, will soon make an appearance in the product line.

The 3300 is set up with an 8-bit accumulator, an 8-bit accumulator extension, an 8-bit addressable status register, and the claimed ability to use every byte of core as an index register. The designers also claim that double memory reference and arithmetic instructions give the 3300 instruction logic equivalent to "many" 16-bit computers. Five modes of addressing are used, including: in page, absolute (page 0 or page 1), immediate, and indirect.

The system's speeds are given as 6.4 usec for a 16-bit binary or BCD add (4.8 usec for 8-bit words), better than 300KC for the transfer rate of the dma.

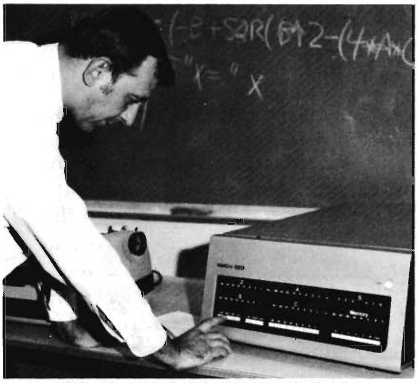
The 3300 is intended for commercial dp use. Its primary language is BASIC. Extensions to the compiler have been incorporated to provide for an "Immediate Execution" mode (giving one user complete control, this is also called "Calculator" mode), multiple instructions per line, COMMON data areas, 8-digit floating-point accuracy (with exponents from 10^{-63} to 10^{63}), diagnostics, a program trace, and unlimited nesting.

Other software furnished includes an assembler, a source tape editor, a debug package, loader and IOCS, and diagnostic routines. There is a charge for BASIC and other applications systems like it, and for applications programs such as the BASIC statistical package.

Like anything else, the cost of a time-sharing system should be figured by how much work is done per dollar. Because of its relatively low cost, the 3300 should show up fairly well in such calculations. A two terminal Selectric/cassette system runs $17,550. If the buyer chooses the Teletype-based configuration without cassettes, the price falls to $15,250. Additional Selectrics run $2100, tty's are $1750, and the cassette drives (which will be Newell's in the first systems) will go for $1400. Couplers go for $795 each, and there is a $1500 charge for the BASIC compiler and system set-up (plus a maintenance cost for the compiler). WANG LABORATORIES, INC., Tewksbury, Mass. For information:

COMPUTERS

low-cost minicomputer provides time-sharing



SOMETHING TO FILL THE GAP
the Wang 3300 central processor

THAT THERE IS A GAP in pricing and performance between programable desk-top calculators and expensive time-sharing systems is known to most researchers looking for a modest system. One of the leaders in the calculator field may have filled that gap by designing the first time-sharing minicomputer.

The step up to a minicomputer is a small step for Wang Laboratories Inc. (Tewksbury, Mass.) but could be an important jump for many research facilities with this type of intermediate need in computing power.

Industrial Research has learned that the central processor and two teletype terminals will cost $15,250, which compares favorably with most systems costing $65,000. Up to 16 in-house or telephone-remote users may operate simultaneously using the popular BASIC language.

Heart of the system is the 3300 digital computer that offers a sophisticated instruction and input/output capability. An 8-bit machine, expandable in 4,000 units up to 65,000 of 1.6-μsec core, the 3300 has a repertoire of 70 instructions, over 20 memory reference instructions, and several other features common only to more expensive systems.

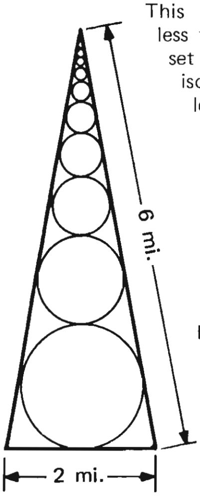
The BASIC language was chosen after Wang market research showed it to be the most popular conversational time-sharing language. Included in the software package are: an immediate mode that can give each user the capability of a desk-top calculator, a "diagnostic pointer" that debugs and provides error checking, a common data allocation that allows each user to retain variables between chained programs, BASIC print and tap extensions for plotting functions, and terminal convenience features that can save and load programs via high-speed cassette tapes.

Current terminal equipment for the 3300 system includes a 15 CPS modified typewriter with character buffer and BASIC compatible character set. In addition, there is an acoustic couple modem for phone line terminal hookup. In the future, Wang expects to add a low-cost 65,000 disk and a high-speed 0.5 megabyte disk.

Among options that can further aid users are the program library abstract service, which lists programs written and maintained for the system and complete applications systems software. □

JUNE PROBLEM OF THE MONTH

— inscribed circles —



This is another quickie requiring less than 30 seconds. The infinite set of circles inscribed in the isosceles triangle shown at the left are given. What is the sum of their circumferences if the legs of the triangle are 6 miles long and the base is 2 miles long?

Contributed by

Dr. T. Skinner, Tech Ops., Inc.

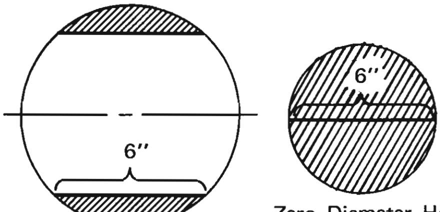
Readers are invited to send their solutions to the Editor.

For solution:
see July issue of PROGRAMMER

SOLUTION TO MAY PROBLEM OF THE MONTH

— drilled sphere —

The major point to note is that *neither the radius of the sphere nor the diameter of the hole* were given. In addition, the problem was stated to have a unique solution. With this information one can assume that the volume remaining after drilling is *independent* of both radii of the hole and sphere. Thus, by choosing the most simplifying situation of a zero diameter hole, the remaining volume is equal to the entire volume. Thus the remaining volume for any situation is 36π cubic inches. This result may be rigorously proven using techniques of solid geometry or calculus.



Large Diameter Hole — Zero Diameter Hole

SAME REMAINING VOLUME

The first five correct solutions were received from:

Greg Bethards — Wang Laboratories, Tewksbury, Massachusetts.
D. F. Patterson — The M. W. Kellog Co., Piscataway, New Jersey
Edward Barrett — U. S. Bureau of Mines, U. S. Department of Interior, Pittsburgh, Pennsylvania.
M. L. Seaman and E. L. Blakslee — American Can Co., Barrington, Illinois.
Peter A. Mora — Shell Development Co., Oakland, California.

CORRECTION TO APRIL SOLUTION

Several readers correctly pointed out an error in the published solution of the April Problem of the Month, the town election. Ken Bernstein has included the following correction:

The analysis leading to the probability of a *tie* was correct. The probability that a tie would occur, given that Smith received 301 votes and Jones received 200 votes, is $2 \times \frac{200}{501}$. The probability that a tie occurs and that Smith received the first vote under the above circumstances is $\frac{200}{501}$ since it was previously shown that if a tie occurs the probabilities that either candidate received the first vote were equal. This probability can also be expressed as:

Probability (Smith received 301 votes, Jones received 200 votes, Smith received the first vote, and a tie occurred) = (Probability Smith received the first vote) x (Probability In the succeeding 500 votes Jones was ahead at some point given that Smith received the first vote.)

This last probability is the quantity we wish — i.e. the probability that (given Smith received 300 votes and Jones received 200 votes) Jones was ever ahead.

Using the information we have already determined in the above equation:

$$\frac{200}{501} = \frac{301}{501} \times \text{desired probability} \quad \text{or} \quad \text{probability} = \frac{200}{301} = 0.6644518272$$

This method is similar to that proposed by Prof. Ricardo A.R. Palmeira, of the University of Texas who started with a Lemma first demonstrated by Bertrand in 1887. The same solution was also derived by Marvin E. Whatley, Oak Ridge National Laboratory.

Sales and Service Locations by City and State

ALABAMA
Huntsville 35801
3322 Memorial Parkway, S.W.
Suite 46
(205) 881-5360
Mobile 36608
3304 A Old Shell Road

ALASKA
Anchorage 99501
P. O. Box 3-3828
(907) 279-3223

ARIZONA
Phoenix 85015
1600 W. Camelback Road, Suite E
(602) 277-4719

ARKANSAS
Little Rock 72205
5904 West Markham
(501) 666-7115

CALIFORNIA
Burlingame 94010
1799 Old Bayshore Highway
(415) 692-0584
(415) 697-7103
Telex: 34-9395
Los Angeles 90045
5250 West Century Boulevard
Suite 407
(213) 776-2985
Telex: 67-4572
Sacramento 95825
1601 Fulton Ave. Suite 4
(916) 489-7326
San Diego 92109
4438 Ingrahm Blvd.
Room 202-203
(714) 276-8464
Telex: 695-445
Santa Barbara
Answering Service
(805) 962-6112

COLORADO
Denver 80223
1309 South Inca Street
(303) 744-1301
Telex: 45-4419

CONNECTICUT
Hamden 06514
60 Connolly Parkway, Bldg. 11
(203) 288-8481
(203) 288-8482

DELAWARE
Via Haverford, Pennsylvania 19041
354 Lancaster Avenue
(215) 642-4321

DISTRICT OF COLUMBIA
Via Silver Spring, Maryland 20910
7961 Eastern Avenue
(301) 588-3711
Telex: 89-8432

FLORIDA
Fort Lauderdale 33304
2455 East Sunrise Blvd.
International Bldg. Suite 608
(305) 563-8458
(305) 563-8450
Telex: 51-4480
Jacksonville 32207
Suite 1824, Prudential Building
Miami Road
(904) 396-1415
Orlando 32803
3165 McCrory Place
(305) 841-3691
Tampa 33609
5445 Mariner Street
(813) 872-7347

GEORGIA
Atlanta 30329
Suite 111, Building 3
Dunwoody Park
(404) 451-0178
(404) 451-0179
(404) 451-2000
(404) 451-2115

HAWAII
Honolulu 96813
49 South Hotel Street
Empire Building, Room 203
(808) 536-5359

IDAHO
Via Denver, Colorado 80223
1309 South Inca Street
(303) 744-1301
Telex: 45-4419

ILLINOIS
DesPlaines 60018
2200 East Devon Avenue
(312) 297-4323
Telex: 72-6426
Morton 61550
c/o Neil B. Graf
356 East Hazelwood Street
(309) 674-8931

INDIANA
Indianapolis 46220
2825 East 56th Street
(317) 257-5381

IOWA
Des Moines 50310
2725 Douglas Avenue
Thompson Plaza Building
(515) 277-2691

KANSAS
Wichita 67211
227 Ida Street
(316) 262-1388

KENTUCKY
Louisville 40207
142 Chenoweth Lane
Rooms 107 & 109
(502) 897-5143

LOUISIANA
Metairie 70003
7809 Airline Highway
Suite 305D
(504) 729-0581
Telex: 58-7493

MAINE
Portland 04101
Warren Building
500 Forest Avenue
(207) 774-2180

MARYLAND
Baltimore 21204
6800 Loch Raven Blvd.
(301) 821-8212
Telex: 87-636
Silver Spring 20910
7961 Eastern Avenue
(301) 588-3711
Telex: 89-8432

MASSACHUSETTS
Northampton 01060
P. O. Box 242
(413) 734-2230
Tewksbury 01876
836 North Street
(617) 851-7311
Telex: 94-7421
TWX: 710-343-6769
Waltham 02154
395 Totten Pond Road
(617) 891-6621
(617) 891-6622
(617) 891-6623

MICHIGAN
East Lansing 48823
541 East Grand River
(517) 351-6783
Grand Rapids 49503
2215 Oak Industrial Drive, N.E.
(616) 458-6595
Southfield 48075
24632 Northwestern Highway
(313) 352-7112

MINNESOTA
Minneapolis 55420
9001 Bloomington Freeway
(612) 884-7328 Telex: 29-0285

MISSOURI
Kansas City 64130
1815 East 63rd Street
Suite 309
(816) 444-8388
Telex: 42-6309
St. Louis 63124
9909 Clayton Rd.
Suite 2
(314) 997-5866
Telex: 44-2319

NEBRASKA
Omaha 68131
33 Kiewit Plaza
3555 Farnam Street
(402) 341-6042

NEVADA
Reno 89502
John Lakey Company
1685 Chaska Place
(702) 322-4692
Via Los Angeles, California 90045
5250 West Century Blvd., Suite 407
(213) 776-2985
Telex: 67-4572

NEW HAMPSHIRE
Manchester 03101
22 Bridge Street
(603) 669-0404

NEW JERSEY
Cranford 07016
530 South Avenue, East
(201) 272-7160

NEW MEXICO
Albuquerque 87110
1805 San Pedro, North East
(505) 265-5608
Telex: 660438

NEW YORK
Binghamton 13901
O'Neil Building, Suite 408
(607) 722-7616
Lake Success, 11040
3000 Marcus Avenue
(516) 437-2500 Telex: 96-1326
Latham 12100
Latham Corner Shopping Center
Suite 206
(518) 785-3234
New York City 10001
875 Avenue of the Americas
Suite 1104
(212) 736-0084
Rochester 14619
474 Thurston Road
Rooms 7 & 8
(716) 328-2510
Telex: 97-8349
Syracuse 13206
3522 James Street
(315) 437-7279
Williamsville 14221
P. O. Box 5
(716) 634-5448

NORTH CAROLINA
Charlotte 28202
112 East Morehead Street
(704) 372-9660
Greensboro 27405
1025 East Wendover Avenue
(919) 272-5683

NORTH DAKOTA
Via DesPlaines, Illinois 60018
2200 East Devon Avenue
(312) 297-4323
Telex: 72-6426

OHIO
Cincinnati 45246
11750 Chesterdale Road
Suite 3111, Atkinson Square
(513) 771-3445 Telex: 21-4211
Cleveland 44126
4402 West 215th Street
(216) 331-6525 Telex: 98-0480
Columbus 43224
909 Oakland Park Ave., Suite A
(614) 268-3511

OKLAHOMA
Oklahoma City 73106
2720 North Classen Blvd.
Suite 113
(405) 524-4489
Tulsa 74105
Post Office Box 7338
4920 South Lewis Ave. Suite 101
(918) 743-2571

OREGON
Portland 97225
3162 Southwest 87th Street
Post Office Box 25183
(503) 297-2501

PENNSYLVANIA
Harrisburg 17105
Post Office Box 3526
(717) 236-4782
Haverford 19041
354 Lancaster Avenue
(215) 642-4321
Telex: 83-4797
Pittsburgh 15235
11620 Frankstown Road
(412) 243-3642
Telex: 81-2359
Wilkes Barre 18703
P.O. Box 1361
(717) 675-3879

RHODE ISLAND
Providence 02904
849 North Main Street
(401) 421-0710

SOUTH CAROLINA
Charleston 29407
1255B Savannah Highway
Parkwood Professional Building
(803) 766-0851

SOUTH DAKOTA
Via DesPlaines, Illinois 60018
2200 East Devon Avenue
(312) 297-4323

TENNESSEE
Chattanooga 37402
124 East Tenth Avenue
(615) 266-5055
Knoxville 37902
313 Commerce Avenue
(615) 546-8881
Telex: 55-7468
Memphis 38111
80 North Tillman
(901) 327-8241
Nashville 37215
2402 Twelfth Ave., South
and Caruthers Avenue
(615) 385-1262

TEXAS
Austin 78752
406 East Anderson Lane
(512) 454-4324
Dallas 75206
6400 North Central Expressway at
University Blvd.
Suite 414, Noel Page Bldg.
(214) 361-7156
Telex: 73-0324
El Paso 79931
P. O. Box 31040
(915) 565-9927 & 8
Fort Worth 76111
Sylvania Building, Suite 7
403 North Sylvania Street
(817) 834-1611
Houston 77005
2370 Rice Blvd.
Suite 108 Telex: 76-2379
(713) 522-1011
Longview 75601
Post Office Box 5038
(214) 758-1810
Midland 79701
Post Office Box 1751
2209 N. Big Spring, Suite D
(915) 683-3304

UTAH
Salt Lake City 84115
2285 South Main Street
Suite 15
(801) 487-5791

VERMONT
Montpelier 05602
38 Elm Street
Post Office Box 481
(802) 223-6398

VIRGINIA
Arlington 22202
775 So. 23rd Street
Cor. Hayes Street
(703) 684-6254
Newport News 23601
9314 Warwick Boulevard
(703) 595-6777
Richmond 23221
3318 West Cary Street
(703) 359-6320

WASHINGTON
Seattle 98103
8610 Aurora, North
Suite 1
(206) 525-2000
Telex: 32-0190
Spokane 99208
P. O. Box 7676
(509) 326-1920

WEST VIRGINIA
Charleston 25302
110 West Lee Street
(304) 346-0345

WISCONSIN
Madison 53703
222 South Hamilton
Suite 23
(608) 255-4411
Milwaukee 53222
7729 West Burleigh Street
(414) 442-0160

WYOMING
Via Denver, Colorado 80223
1309 South Inca Street
(303) 744-1301
Telex: 45-4419

PUERTO RICO
Hato Rey 00919
Techsale Corporation
P. O. Box 1632
(809) 761-0772

CANADA
Don Mills
Wang Laboratories (Canada) Ltd.
180 Duncan Mill Road
(416) 449-7890
(416) 449-7894
Telex: 06217549

EUROPE
Belgium
Wang Europe, S. A.
68 Rue du Moulin
Brussels 1030,
(02) 19.25.34
Telex: 84623783 (WUI)

TAIWAN
Taipei
Wang Laboratories (Taiwan) Ltd.
9, Sec. 5, Nanking East Road
Taipei
Tel. 77.96.19

UNITED STATES AMBASSADOR VISITS WANG LABORATORIES EXHIBIT IN LONDON

The United States' Ambassador to the United Kingdom, Walter H. Annenberg, recently visited Wang Laboratories' exhibit at Instruments Electronics Automation Show in Olympia Hall, London. The show was held during the week of May 11 to 16. Ambassador Annenberg (left foreground), and Robert Mills (right foreground), Manager of Operations for Wang Electronics, Ltd. in London, are shown discussing a Wang calculator. With the ambassador are Stanley M. Cleveland, minister for Economic and Commercial Affairs, and Harold E. Hall, Counselor for Commercial Affairs.

The Programmer

WANG LABORATORIES, INC.
836 NORTH STREET
TEWKSBURY, MASS. 01876

BULK RATE
U. S. POSTAGE
PAID
Permit No. 1000
Boston, Mass.

CENTRAL AUTO SPRINKLER CO
ENG DEPT
4TH ST + CANNON AVE
LANSDALE PA 19446

Printed in U.S.A.
700-0148AI