DESIGN OF ITT 525 "VADE" REAL-TIME PROCESSOR

Dr. D. R. Helman, E. E. Barrett, R. Hayum and F. O. Williams
ITT Federal Laboratories
Nutley, New Jersey

SUMMARY

The ITT 525 VADE (Versatile Automatic Data Exchange) is a medium-scale communications processor capable of handling 128 duplexed teletype lines and 16 high speed data lines. The processor is of the single-address, parallel binary type utilizing a two-microsecond-cycle-time core memory and operating at a single-phase clock rate of four megacycles. The fundamental design approach of the machine is to trade the intrinsic speed of high performance hardware for a reduction in total equipment, through time-sharing. The memory is shared between stored program and input/output functions without the use of a complicated "interrupt" feature. Serial data transfers between the memory and communication lines are performed on a "bit-at-a-time" basis requiring a minimum of per-line buffering. The central processor hardware is largely conventional but has been reduced as much as possible without impairing the power of a basic communications processing instruction repertoire—which includes indexing and character mode operations but not, as yet, multiplication or division. Instruction time is six microseconds and the number of instructions performed per second varies from 63,500 to 81,000 depending on the existing input/output traffic load. Duplexing of the machine is accomplished by a "shadow" system whereby the off-line processor is continuously updated by the on-line processor through one of the normal high speed data links.

INTRODUCTION

At the present time the Design of Real-Time Processors is following the multiprogramming or multisequencing philosophy. Multiprogramming is usually defined as the time sharing of a single central processing unit. The processor responds to the real-time channels by activating a stored program which in many cases is unique to that particular channel. The memory must store not only the individual programs, but also the addresses of the programs for the corresponding channels. Furthermore, the processor must provide some type of priority-interrupt system which will respond to the various service requests of the real-time channels.

Real-time processors designed upon the multiprogramming basis usually provide per-line equipment which converts the serial binary stream to a parallel character. After this conversion, the character enters the Input-Output system where character or word buffering occurs. Finally, the data enters the Main Memory for processing, after the channel service—request has been recognized. To implement such a processing system much special purpose hardware and programming is required. Program, index and supervisory memories may be utilized in conjunction with special purpose priority interrupt hardware.

ITT 525 (Versatile Automatic Data Exchange) is a real-time processor designed upon a radically new philosophy. The objective of the design is to trade-off high internal processing speed with hardware, such that
effective utilization is made of the machine capability. The ITT 525 processor serves as a real-time store and forward message processor, which may serve as many as 16 high speed duplexed data lines and 128 teletype lines operating at a 100% line utilization.

The unique features of the ITT 525 include the sharing of one core memory for input, output and processing functions; the serial bit at a time assembly and disassembly of messages in the shared core memory using some simple in-out hardware; the storage of instruction micro-function logic instead of the standard operation decoder and logic; a minimum register central processor utilizing direct data transfers and providing the facilities for indexing and character mode; a powerful instruction repertoire for the implementation of the operational, utility and diagnostic programs.

System Design

The objective of the ITT 525 design was to produce a versatile message processor, at a minimum cost per line, to perform the function of a local area center handling a reasonable amount of data and teletype lines. It was decided to implement a system capable of interfacing with 128 duplexed teletype lines plus 16 duplexed data lines.

In order to achieve minimum cost per line the first design philosophy established was to make optimum use of the common equipment. Thus, it was decided to time-share one core memory and the major control circuits, between the In-Out unit and the central processor. This was made possible because of the high speed core memory, operating at a 2 microsecond read-write speed, and 4 megacycle logic circuits. If it is assumed that two memory cycles are required to perform a machine instruction, a maximum of 250,000 instructions per second may be executed by the ITT 525.

Since the ITT 525 has such a high internal speed, it was further decided to deviate from conventions and accept data from the real-time lines a bit at a time per line into the one core memory with no per line buffering. The messages are, thus, taken directly from the serial bit stream into the core memory where they are completely assembled. The message remains in this storage area while it is being processed and analyzed by the stored program and finally becomes disassembled one bit at a time for the output line transmittal. This line scanning or bit sampling of the input and output lines requires a total of 62% of the total machine time for the 128 teletype and 16 data line configuration. Thus, a total of 95,000 instructions per second are available for the central processing functions.

The system analysis of the ITT 525 determined that to completely process the assembled messages, with an input line utilization of 100%, would require between 35,000 to 45,000 instructions per second. This processing consists of message validity checking, message decoding for destination and priority, message filing, message journaling, message code conversion and finally output queuing. Approximately 1500 instructions per message are needed to perform the complete processing functions. This processing estimate of 45,000 instructions/sec is rather conservative, since the probability of continuous 100% line utilization is very remote. Thus, the average processing time will be much smaller than the 45,000 instructions per second. However, the total available central processing time for the ITT 525 is 95,000 instructions per second so that, obviously, 50,000 instructions per second remains for future expansion or a further trade-off of time for hardware or flexibility.

To make further use of the extra machine time, it was decided to employ the concept of stored micro-operations or microfunctions. A reserved area of memory contains the microoperations for each machine instruction. This word is retrieved for each instruction before the execution of the instruction can proceed. This extra memory retrieval per instruction uses an equivalent of 31,500 instructions per second, so that a total of 63,500 instructions per second remain for message processing. The stored microfunction logic replaces the conventional wired logic operation decoder and some corresponding microoperation logic. However, the primary advantage of this approach is not the reduction of hardware obtained, but in increased instruction flexibility and speed of machine check-out. Each microfunction may be tested independently either by a diagnostic program or from the operator's console. This facility greatly reduces the time required to isolate and repair machine failure.

In summary, the ITT 525 system design has resulted in the development of a stored program processor in which the memory is...
time-shared between Input/Output wired logic and the program control logic. The processor operates internally on parallel binary words each consisting of thirty-two bits. The instruction cycle, consisting of 6 microseconds, performs single address instructions with an available rate of at least 63,000 instructions per second.

Machine Organization

The block diagram of ITT 525 VADE (Figure 1) illustrates the machine configuration consisting of a Central Processor and In-Out time-sharing the core memory.

Central Processor

The central processor of the ITT 525 is a single address, binary, one's complement processor employing stored logic for instruction decoding, one index register and character mode operation. The design is based upon a minimum register configuration, with maximum time sharing, and direct transfer between registers.

The instruction cycle of the processor is six microseconds. This cycle is broken down into three memory accesses: one unload-load to fetch the instruction; one unload-load to obtain the stored microfunction control word; and finally one unload-load to obtain the operand and execute the instruction. The processor word consists of thirty-two bits which may take the form of 4 eight bit characters or an instruction word divided into six fields (Figure 2).
The accumulator register of the ITT 525 is the heart of the arithmetic unit. This register in conjunction with the memory buffer performs a parallel, two-step addition and subtraction. One of the unique features of the Accumulator is the carry chain configuration, which has a maximum delay of 425 nanoseconds.

The standard, simple, carry chain configuration consists of a single gate per flip-flop stage. The delay encountered for this arrangement is the number of stages times the gate delay, which for the ITT 525 would have been 32 x 35 or 1120 nanoseconds. In order to take full advantage of the four megacycle clock it was determined that a carry chain delay of less than 500 nanoseconds would be desirable for the ITT 525. One technique available to speed the carry chain is the pass-carry or grouping-carry idea. In this case, several stages are combined to form one large carry gate, thus, reducing the overall carry chain delay. However, in the ITT 525 Accumulator the carry chain design is based upon the group hierarchy principle. This concept makes optimum use of the recursive nature of the carry equation by first combining flip-flops into groups and groups into sections. In this way, if a carry has to be passed for 32 bits, it will avoid not only the groups of flip-flops, but also the section of groups.

The functions that the accumulator may perform upon data are as follows:

1. Partial Add (Exclusive Or)
2. Carry
3. Inclusive Or
4. Reset
5. Complement
6. And
7. Cycle Left

The accumulator may be sensed by program for the following conditions:

1. Minus Zero
2. Plus Zero
3. Overflow
4. Any bit of Character 3 (24-31)
5. Plus or Minus Zero

The Control Buffer contains the Instruction Micro-operations obtained during the Processor's "Stored Logic" Memory cycle. Each bit of this register is assigned a specific microfunction, such as "Reset Accumulator," "Transfer Index Register to Memory Buffer," etc. If a particular instruction requires that microfunction a "One" appears in that bit position. High fan-out drivers distribute these microfunctions to the various register input gates. In addition to the increased flexibility and some cost reduction, the use of the stored logic technique provides a powerful tool for checkout and maintenance.

**INPUT/OUTPUT UNIT**

With the single exception of a direct input from a paper tape reader on the console, all processor inputs and outputs are handled by the Input/Output Unit, including transfers between core memory and secondary storage devices. The initial implementation of the ITT 525 system has the following traffic-handling capability:

1. 16 duplexed high speed data lines operating at any speeds up to 2400 bits per second (8-bit code).
2. 128 duplexed teletype lines operating at speeds of 60, 75 or 100 words per minute (5-bit code).
3. Block transfers of computer words to one of eight magnetic tape units operating at a transfer rate of 2500 computer words per second.

This is a maximum capability configuration with regard to teletype and data lines. Smaller machine capabilities are implemented in any combination of modular blocks of 4 data or 16 teletype lines. Also, individual line speeds are completely independent and may be changed without incurring hardware changes.

Although the stated capabilities conform only to the task of communications processing, the unique features of the Input/Output Unit are applicable to other tasks and configuration requirements with a moderate amount of hardware change. The "bit-at-a-time" technique is easily adapted to various forms of serial bit streams, regardless of framing or synchronization details, and the method used for tape word transfers is directly applicable to any block transfer process, even if the "blocks" are degenerate ones of only a few words of characters.

**Teletype and High Speed Data Lines**

Incoming serial bits on these lines are transferred directly into core memory. Outgoing serial bits are transferred directly from the memory to one or two per-line output flip-flops. Each output line requires one
flip-flop for pulse-stretching and data output lines require an additional flip-flop to reduce bit jitter. The total line storage required is 160 flip-flops, which compares favorably with the 1536 flip-flops required if each line (input and output) were to terminate in a one-character buffer.

Several fixed core memory locations are permanently assigned to each input and each output line which are used by the input/output logic. These per-line locations contain space for character assembly/disassembly, program flag bits, control bits and timing information. The stored program exercises control over the Input/Output Unit by performing regular scans of these words and changing their contents when necessary, thus modifying the operations of the wired logic of the Input/Output Unit. Specifically, in addition to noting the end of incoming messages or initiating output for outgoing messages, the program must make "bin" assignments to active lines. It is unfeasible to reserve for each line a space in memory adequate for the largest possible message. Alternatively, "bins" of 75 words, or 300 characters, are assigned to active lines as they are required. The Input/Output Unit logic notifies the program of such needs by flag bits and can store temporarily, in the fixed memory locations, as many as twelve incoming characters during the interim between bin assignments. In normal input operation, after a bin assignment is received, characters are transferred to memory soon after completion, independent of the stored program.

In reference to the block diagram of Figure 1, the basic operation of the I/O Unit is rather simple. A "Scan Generator" controls line selection and memory addressing (for control words) according to a fixed cycle of operation. Then, for each line scan, the most important control word for the line, the "status word," is unloaded to the "Status Word Buffer" where it remains for one or two more memory cycles to control operations on the line information through use of the Memory Buffer for examination and modification of other words. Finally, two counters are used for timing purposes indicated below.

The Input/Output Unit obtains control of the memory and performs a "scan cycle" every 280 microseconds. This interval is compatible in two different ways, with the bit periods of the lines. A 2400 bit-per-second data line has a bit length of 417 microseconds and a 100 word-per-minute teletype line has a bit length of 13.46 milliseconds. By scanning all data input and output lines each scan cycle but only one-fourth the teletype input lines and one-sixteenth the teletype output lines, the following rates are obtained:

1. data lines are scanned at least 1.49 times per bit.
2. teletype input lines are scanned 12, 16, or 20 times per bit for 100, 75 and 60 word-per-minute lines, respectively.
3. teletype output lines are scanned 3, 4, or 5 times per bit for 100, 75 and 60 word-per-minute lines, respectively.

These rates permit the sampling of teletype input lines within ±8% of the nominal bit-center, or better, to minimize the effects of distortion. Actual sampling is accomplished on the basis of predicted bit-center sampling times established when the "stop" pulse to "start" pulse transition is detected and stored in the fixed memory space for the line for later coincidence comparison with a real-time (based) counter. Data input lines are sampled on the basis of timing provided by their associated synchronizing signals. The synchronizing signal used has a frequency of one-half the bit rate of the line. The value of this signal (zero or one) is stored on each scan and the line value is not sampled unless the stored and present values of the synchronizing signal differ. Output lines are handled in exactly the same manner as input lines except that teletype output lines do not need the high scan rate provided for input teletype lines since the output process itself controls the waveform distortion.

During the I/O scan operations, one memory cycle is required to scan each line and two additional memory cycles are required for each character transfer between the fixed memory locations for the line and the message bin elsewhere in memory. By limiting the number of character transfers allowed in each scan, minimum and maximum I/O scan time requirements of 144 and 173 microseconds, respectively, are obtained. Since the interval between scans is 280 microseconds, 52% to 62% of total processor time is spent in input/output operations and 38% to 48% remains for stored program use (63,500 to 81,000 instructions per second).

**Magnetic Tape Block Transfers**

Block transfers of computer words between the memory and the Magnetic Tape Module...
of the 525 are initiated by the stored program and executed in detail by the Input/Output Unit. A single fixed location in memory holds a block address and a count of the number of words to be transferred. During a block transfer, the MTM sends requests for word transfers to the I/O Unit at approximate 400 microsecond intervals and these requests must result in a word transfer within 67 microseconds. It is possible, then, that a word transfer must be made when the I/O Unit is not in control of the memory. In this case, the I/O Unit gains control of the memory only for the transfer time and then relinquishes control until the next regular line scan cycle time. While it is in control of the memory, the I/O Unit uses the fixed location MTM word to control the transfer of a word between the specified address and the MTM buffer. Then the address is incremented, the block transfer count decremented and the MTM word is transferred back to its fixed location. Much of the logic performing these operations within the I/O Unit is common to the logic required for teletype and data line operations, since character transfers and bin counts for these lines are handled on the same general basis. The I/O Unit is easily expanded to include a similar block transfer provisions for magnetic drums, card readers, punches, printers and displays.

Except for the implementation of the block transfer process, there is nothing unusual about the operation of the magnetic tapes. One tape at a time may be selected to read, write, backspace one record, advance one record, write end-of-file or rewind, the rewind operation being performed in a quasi-off-line state so that other units may be selected during this operation.

**Input/Output Program Requirements**

Since the ITT 525 has no interrupt feature, the stored program—input/output interface is an unusual one. Regular scanning of the fixed—location input/output control words is essential to the bin assignment task of the program. Input teletype words must be scanned at least once every 800 milliseconds and input data words at least once every 40 milliseconds. These figures represent the amount of time required for incoming information to fill the twelve-character per-line temporary storage space. Output words are scanned (for bin assignment needs) at whatever speeds the programmer desires since no information can be lost and the only consideration is for efficiency in transmitting messages which are more than one bin in length.

A more complicated problem arises when the program must exert control over I/O Unit operations by modifying the contents of fixed location control words. An input/output line scan cycle may interrupt the program and change the contents of a control word at the same time that the program is preparing to modify the word. Since the program has no natural means of knowing an interruption has occurred, it would tend to force obsolete data into the control word. To circumvent this problem, a flip-flop is provided which can be sensed by the program and which, if on, guarantees the program that the six instruction times immediately following the sense instruction will be free of I/O Unit interruption—this number of instructions being sufficient to perform the control word modification.

**Duplexing**

To meet the reliability of many real-time problems, the ITT 525 may be utilized in a duplexed configuration. The duplexing design of the ITT 525 has been selected on the basis of maximum reliability and minimum special purpose duplexing hardware.

The Duplexed System configuration is illustrated in Figure 3. System A is in control of the magnetic tape and communication links, all input lines and output lines are accepting and sending data. The standby machine B also has the input lines connected to it and accepts all input messages. Furthermore, machine B assembles, processes the message and sets up the output queue. Machine A regularly sends data to machine B via a

![Figure 3. Duplex Configuration.](From the collection of the Computer History Museum (www.computerhistory.org))
normal high speed data line concerning the disposition of messages. Once machine A has outputted the message, machine B erases the message and updates its own output queue. Machine B does not file, journal or overflow, or output any messages. Using one of the high speed data links for the regular communication between machines A and B insures a smooth cutover with no loss of data. The worst condition that might occur is that after cutover machine B might output a given message again since it had not received the last disposition data.

Machine A, if in control may by program relinquish control to Machine B and vice versa. Also, manual means are available to establish the duplex configuration via the Operator's Console. Output lines and the magnetic tapes are automatically switched into the proper machine for each configuration.

CONCLUSION

The ITT 525 VADE is intended to do a medium-scale job using only a small-scale amount of hardware. Although testing and program debugging will not be complete for another two or three months, there is little doubt that the system will satisfy this aim. Further extensions of the VADE approach have been planned which will improve the speed, line-handling capacity and versatility of the machine by modular additions of hardware at selectively increased cost.