TRENDS IN DESIGN OF LARGE COMPUTER SYSTEMS

Charles W. Adams
Charles W. Adams Associates, Inc.
Bedford, Massachusetts

Summary

New developments in computer design are reported and trends are analyzed -- first with regard to physical devices with emphasis on fixed and high-speed storage systems; then with regard to logical techniques including various logical organization schemes, stored logic, concurrent autonomous operation, and new approaches to modularity in design.

Introduction

When the WJCC Program Committee invited me to prepare a survey of newer, larger computer systems, they suggested that I include both an indication of what are the characteristics of the specific systems and an interpretation of what these imply for the design of future systems. It was also their intention that computer manufacturers be encouraged to give five-minute discussions of their new systems.

I have taken the liberty of changing that format slightly, by considering different aspects of computer design and using specific systems as illustrations, and have sought to find "interlude" speakers who would give broad, unbiased, expert coverage to important design aspects.

In preparation for the session, a rough but fairly comprehensive outline was prepared and distributed to a dozen or so friends who are well-known both for their awareness of new developments and trends in the computer field and for their willingness to express themselves freely. Based on their thoughtful replies, as well as on material received from a number of people representing the computer manufacturers who had new systems and devices to report, I came to an almost obvious conclusion: the keynote in the design of new systems, whether large or small, is not hardware but logic. Consequently, new developments in physical hardware can be rather quickly summarized, with principal emphasis given to new approaches to internal logical operation.

New Devices

The hardware development which caused so much excitement only a few years ago -- solid-state circuitry, two-microsecond core storage cycles, magnetic tapes operating reliably at upwards of a hundred thousand characters per second, magnetic juke-boxes for "random" access to files of ten or twenty million characters, scanners for reading and interpreting printed numbers optically -- are now taken for granted. Today the attention of the system designer is turned toward facilitating the effective utilization of these now almost humdrum achievements in hardware design.

This is not to say that new developments are not being made in the hardware area. For one thing, rather spectacular improvements are being made in the reliability and costs associated with all aspects of computer componentry. Most notable perhaps are the continually improving performance of the large drums, disc files, optical scanners, and magnetic-ink character recognition devices (witness the new checks the banks are supplying to all their customers as part of the nation-wide adoption of MICR).

A new approach to random-access files has been announced by NCR, to wit the CRAM system involving automatically selected magnetizable sheets which are then wrapped around a drum. IBM demonstrated last year a "tractor" system for selecting tape reels automatically. The really new hardware seems, however, to be appearing primarily in the area of high-speed storage, both erasable and non-erasable.

Non-erasable Storage

Increased attention is being given to various forms of non-erasable (i.e., read-only) storage systems, large and small. Though new in detail, such devices are far from new in principle. Interest in them stems in part from increasing recognition of the potentialities of using "stored logic" in system design, and in part from
the growing popularity of special-purpose stored program and/or stored data file systems.

One example of non-erasable storage is the 0.2 microsecond magnetic slug memory which will be used in the Ferranti Atlas computer to contain an executive routine for multi-programming control as well as such things as macro-program subroutines. Speed is the most important aspect of this device, but the fact that it can be altered only at the factory puts the Ferranti programmers in an enviable position indeed since no user can tamper with their programming package once it leaves the plant.

Various control computers use non-erasable memories such as drums with mechanically locked-out recording circuitry, twisters biased with permanent magnets, and non-destructive reading from thin-film storage. For example, Remington Rand announced some time ago a dual-film device in which an erasable cobalt film biases a read-only permalloy film, and is actually using Bell Labs twisters with magnets mounted on removable cards. In the M490 and 1206 they provide a small diode matrix, the contents of which can be mechanically altered only by inserting plugs pre-assembled with wires soldered in place, for use in loading error routines, etc. And, of course, more than ten years ago M.I.T.'s Whirlwind I was performing 125,000 instructions per second, working from a storage of 80 flip-flops and 432 diodes controlled by individual toggle switches.

A third class of non-erasable storage is photographic storage, optically scanned by cathode ray tubes, which yield a very large high bit-rate file -- e.g., the IBM photoscopic disc memory for use in language translation and the ETL random-access photographic storage used in electronic telephone exchanges. In these cases, the emphasis is large volume data storage at reasonable costs and/or rather high speeds, while the fixed drums, twisters and diodes are aimed at protection against accidental loss of information (usually of a real-time control program).

High-speed Storage Devices

Reading or writing in a random-access storage in less than a microsecond is being accomplished in thin-film magnetic memories on which the M.I.T. Lincoln Laboratory, Remington Rand, and Honeywell have announced some results. Small (128 word in the UNIVAC 1107 case) thin film storages operating at 0.6 microseconds are now functioning satisfactorily in the laboratory. A ten-thousand-word storage at 0.1 microseconds is the present objective at Lincoln Laboratory.

The National Cash Register Company has announced a magnetic rod memory storing a thousand bits per cubic inch with switching times of 0.05 microseconds.

Aside from film and rod systems, work is apparently still continuing in the cryogenic area (involving superconductivity phenomena at temperatures near absolute zero), but no real breakthroughs have been made public.

Increasing Effective Storage Speed

Techniques for increasing the effective speed of a conventional magnetic core storage device include using two or more independent banks alternately, providing asynchronous operation in which rewriting is delayed if possible until storage access is not otherwise needed, and implementing lookahead schemes by which any potentially idle time is used to read information in anticipation of its later use. The gain from such techniques is limited to a reasonably small percentage increase (under 100%) and certainly in practice has not always been as great as has been expected. In any event, these are basically logical rather than physical means of increasing speed, akin to the various forms of autonomous operation discussed later.

New Logical Designs

The biggest design improvements in large digital computers during the next few years seems likely to come in the logical organization of the systems rather than in the componentry involved. The techniques employed and the objectives gained are many and various -- so much so that what follows cannot lay claim to being even a comprehensive catalog, quite aside from not containing any appreciable degree of detail. What has been attempted, rather, is to make mention of a number of techniques and to assess their apparent purpose and merit.

Arithmetic

The age-old problem of choosing among binary, decimal, and alphanumerical operation has not been completely resolved, but the natural tendency is toward compromise: binary computers which have convenient facilities for conversion to decimal and even machines with two or more forms of operation built directly into the hardware (e.g., the Honeywell 800 and the RCA 601).

Effective storage capacity is sometimes increased by use of a short word length with built-in double-precision arithmetic to be used when needed without an excessive speed penalty (e.g., the Ramo-Wooldridge AN/UYK-1 and the Packard Bell 250). Alternatively,
provision is made for dealing with half-words as in the Remington Rand 1107 and the IBM 7030.

Most large machines of course have built-in floating point arithmetic at least as an optional feature, but the Bendix G20 is the only one being promoted as having no fixed point operations. (Provision is made for unnormalized floating operations which then are essentially fixed point). However, one sometimes encounters businessmen who will not consider the G20 because they definitely want to be able to do fixed point arithmetic.

Address Logic

Merely numbering storage locations consecutively from 0 to N is now old hat. Indirect addressing and "literals" permit instructions to operate on the contents of the location whose address is contained in "x" or, much more directly, on "x" itself. Both are convenient in certain cases, but designers who haven't bits to burn in their instruction words sometimes omit these features.

On the other end of the scale, when there are more storage registers than there are bits to distinguish between them in the instructions, bank addressing (Honeywell 800) or relative addressing (CDC 160) are sometimes used. The programmer often has little patience with these --- to him they are merely a nuisance. But the obvious economy of storage they permit (since most instructions do, or can be made to, refer to nearby locations) has perhaps not been as widely recognized as it should.

Engineering problems occasionally deal with short numbers, but usually with fairly long ones, and never with very, very long ones. The business user, however, concerns himself with "fields," not variables, and his fields can and do run the gamut from a single bit (male or female?) to several hundred bits (home address). Character-addressable machines like the IBM 705 and RCA 501 (and many others) help dispose of this problem, but a more honest way of dealing with the situation appears to be "field addressing," as in the IBM 7070, in which the existence of words is admitted (the IBM 705 has of course a 5-character word length, but evidently is ashamed to mention it). Semantics aside, character addressability would be a very desirable feature if it could be accomplished in a parallel manner to preserve speed.

A novel addressing scheme is planned for the Ferranti Atlas computer. It is a "one-level store" in which a large drum logically comprises the main memory, each drum location being separately addressed, yet the programs actually operate from a large magnetic-core working storage. "Pages" of 512 words are "automatically" brought into core as needed, and copied back to the drum when the space is more urgently needed for another page. Every reference to storage requires that the page number be first processed against a list of all the pages already in core, and the appropriate core page used where possible. An executive routine in the "fixed store" (mentioned earlier) is used to decide which page to replace if the reference is to a page not already in core. The scanning of the list is done in a parallel fashion in a fraction of a microsecond.

Instruction Format and Repertoire

Of the 35 commercially announced solid-state general purpose computers, 24 are single address and only the Honeywell 800 and 460 and the NCR 304 are three-address. The IBM 1401, 1410, and 1620, all the RCA systems, and the RW400 are basically two-address machines but in a few cases four, three, two, one or no addresses are used in different operations.

More startling yet is the logic of the Burroughs B5000 which, they assert, "can well be considered the first non-vonNeumann computer." Operations are grouped into classes and the computer operates in arithmetic mode, subroutine mode, data-manipulation mode, or control mode, the last involving an executive routine permanently recorded on a magnetic drum. In arithmetic mode, the computer interprets a string of intermingled addresses and operations written in "Polish notation" (the expression \( y = (w + i + t) \) becomes \( yw + t + pq - z/ = \) in Polish notation).

A number of interesting operation codes appear in some of the new computers, with the Remington Rand 1107 certainly among the leaders in the aspect of design. It is the stored-logic approach that holds the greatest fascination, however. The Ferranti Atlas, for example, uses one bit to make operation codes which are used as any built-in code but is executed by the computer as a subroutine stored in the fixed store. The Ramo-Wooldridge AN/UYK-1 on the other hand has no elaborate built-in operation at all and uses "logrannes" of simple instructions to carry out more complex operations. Means are provided to go from logram to logram automatically, so that in use they behave much the same as conventional built-in instructions.
Autonomous Operation

To make effective use of each of the expensive high-performance magnetic tape units, core storage banks and arithmetic-logical elements in a large computer, each must be kept as busy as possible. Since different problems make different demands on the various units, proper balance can only be obtained by operating several programs at one time in the hope that the combination will provide a better distribution of demand. If a balanced load is to be thus achieved, each unit must be designed to operate autonomously, under its own control, and there must be an executive or scheduling procedure that keeps everything running as smoothly as possible.

The approaches to the control of a large system of autonomous units are several. In the IBM 7090, Bendix G20, and others, for example, elaborate input-output control units work out of the same storage as the central control, and means are usually provided to "trap" or interrupt the main program when the input-output control needs a new set of instructions. All of the scheduling is done by a part of the regularly stored program, although most users do not have to concern themselves with the question of this "driver" or "executive" routine.

The Honeywell 800 uses a highly-publicized multi-programming arrangement, in which each of up to eight different programs are performed virtually a step at a time in sequence. While a program is waiting for an input-output unit to function, it is by-passed so that there is no idle computer time.

The Ferranti Atlas fixed store is used to hold an executive routine which is automatically called in whenever any autonomous unit needs attention and whenever the central computer is faced with any delay for input-output in the program currently being processed. Thus, the executive routine carries on moment-by-moment control of the scheduling, intermixing programs as required to give the best possible utilization of the system.

The Atlas also provides multiple consoles so that the separate programs may have separate operators. Furthermore, the executive routine can alter the page reference list (see above) according to what program is in use, so that the same "drum" location can be referred to by more than one program yet have no interference between them (the executive routine keeps track of the actual drum pages corresponding to the page numbers used in each program --- the absolute addresses are in effect treated symbolically during operation).

The multiple consoles of the Atlas or the Bendix G20 make possible concurrent "on-line" program debugging by several programmers at one time. The Atlas scheme provides fool-proof protection of each program against accidental alteration by another sharing the machine at the same time. The potentialities and techniques of concurrent (autonomous) debugging would be a worthy subject for considerably more discussion than can be provided here.

Modularity, Graceful Degradation

Intimately associated with the provision for and control of autonomous operations of control computers, storage banks, drums, tapes, in-out devices, consoles, etc., is the question of modularity and of providing for graceful degradation. This latter term is unfortunately not familiar to many computer designers -- as will be seen, it refers to the behavior of the system in the face of malfunctioning of one or more of its parts.

Most present-day large computers are modular in that the user has a wide selection of usable configurations and correspondingly of system rental costs. In most cases, failure of a peripheral device affects only that part of the system, and the rest of the computer can continue in operation. But when any part of the central computer fails, the system is usually completely out of action.

Graceful degradation is a design objective and for large systems is an important one. It implies that multiple central control units and storage banks are provided and are treated as autonomous units along with the tapes, etc. Using the philosophy of a telephone exchange, components are assigned to jobs as needed, and malfunctioning units are by-passed.

Conclusion

It seems safe to predict that computer hardware will continue to gain in speed and particularly in reliability as the years go on. Almost equally certain is a continued trend toward stored logic and toward more autonomous operation in a multiprogrammed modular computer capable of some degree of graceful degradation. The results in terms of improved performance per dollar should be impressive.