C. J. Bashe
W. Buchholz
G. V. Hawkins
J. J. Ingram
N. Rochester

The Architecture of IBM's Early Computers

Most of the early computers made by IBM for commercial production are briefly described with an emphasis on architecture and performance. The description covers a period of fifteen years, starting with the design of an experimental machine in 1949 and extending to, but not including, the announcement in 1964 of System/360.

Introduction

This is a technical account of IBM's principal computers during the fifteen-year period beginning in late 1949 and leading up to the announcement in 1964 of System/360. They were all electronic stored-program computers. Each had a random-access memory that was reasonably large by the standards of the time, could do arithmetic and logic, could call subroutines, could—and did—translate from various symbolic languages to its own machine language, and each had a substantial set of input/output (I/O) equipment. This account emphasizes architecture and performance, that is, the programmer's view at the level of machine language.

The paper begins with a section on machines which were precursors of the computer era. This is followed by descriptions of several series of related computers, each named after its first member: 701, 702, 650, and 1401. Next is an essay on the IBM Stretch system, and the final section concerns I/O techniques for all these machines.

Not all of IBM's stored-program computers of this period are included; because of space limitations, several interesting and important machines are omitted. The selection is based on a subjective judgment of which machines had technically the most impact during those fifteen years and beyond.

Table 1 illustrates the chronology of the machines discussed in this paper. The members of a family or series of machines appear vertically in a column; the later entries are more or less upward compatible with earlier ones. (There is no family relationship, however, among the machines in the column labeled "Others.") All machines up to 1957 used vacuum-tube electronics; from 1958 on, all of IBM's computers were built with transistors. Every machine listed is discussed or referred to in the text.

Except for a discussion of IBM's precursor machines, no attempt is made here to acknowledge adequately the many pioneering projects and individuals in IBM and elsewhere which made these developments possible. The 701 clearly owed much to the machine at the Institute for Advanced Studies, to prior work at the University of Pennsylvania, and to Whirlwind at MIT; the 702 was largely stimulated by the Eckert-Mauchly UNIVAC; and the 650 had early roots in work at Engineering Research Associates. Cathode-ray tube (CRT) memory, as used in the 701 and 702, and index registers are well known to have originated at the University of Manchester. The names of J. P. Eckert and J. W. Mauchly, of J. von Neumann, who was a consultant on the 701, and of F. C. Williams readily come to mind. To give sufficient credit beyond the obvious, however, would go outside the scope of this paper.

Copyright 1981 by International Business Machines Corporation. Copying is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract may be used without further permission in computer-based and other information-service systems. Permission to *republish* other excerpts should be obtained from the Editor.

Table 1 Chronology of principal IBM computers before System/360.

701 Series	*702 Series*	*650 Series*	*1401 Series*	*Others*	*Date*
		650			1948 start
Test Assembly					1949 start
TPM					12/49 start
701					5/21/52 announced
		650 Card			7/2/53 announced
	702				9/25/53 announced
704					5/7/54 announced
	705				10/1/54 announced
				NORC	12/1/54 dedicated
				SAGE	1/55 delivered
		650 Tape			5/3/55 announced
		650 Disk		305 RAMAC	9/4/56 announced
709					1/2/57 announced
	705 III				9/3/57 announced
					VACUUM TUBES
					TRANSISTORS
		7070			9/2/58 announced
7090					12/30/58 announced
			1401		10/5/59 announced
	7080				1/18/60 announced
		7074			6/27/60 announced
			1410		9/12/60 announced
		7072			4/17/61 announced
				Stretch	5/61 delivered
7040, 7044					12/15/61 announced
7094					1/15/62 announced
			1440		10/11/62 announced
			7010		10/29/62 announced
			1460		2/18/63 announced
7094 II					5/16/63 announced
		System/360			4/7/64 announced

Precursors

The decade of the 1950s was a significant one in the history of IBM computers. All of the electronic stored-program computers designed for regular production were announced and first delivered after 1950. All earlier devices that were more than electromechanical accounting machines were either one-of-a-kind machines or combinations of electromechanical and electronic calculating units, which did not incorporate an electronic form of stored program [1].

There were, to be sure, earlier examples of the greater logical power obtained from large numbers of calculating elements operating under elaborate sequence controls and of the increased speed resulting from substitution of electronic circuits for electromechanical relays and counters. The IBM Automatic Sequence Controlled Calculator, which was presented by IBM to Harvard University in 1944 and became known as the Harvard Mark I, was based on electromechanical devices for both calculation and sequence control. The 604 electronic calculating punch, which IBM started shipping in 1948, used electronic counters for calculation and was controlled by a plugboard program of up to 60 three-address instructions.

A major step was the dedication, also in 1948, of the Selective Sequence Electronic Calculator (SSEC) at IBM's headquarters in New York City. It had a hierarchical memory system consisting of 20 000 words of paper-tape storage, 150 words of relay memory, and eight words of electronic memory, each word consisting of 19 digits and a sign. Its electronic arithmetic unit could add, subtract, divide, and multiply, the latter in 20 ms. The paper-tape storage consisted of 66 readers, each holding a loop of tape as wide and thick as an IBM card. These tapes could hold tables, data to be operated on, or sequences of 78-bit instructions. The SSEC was the first operating computer capable of treating its own stored instructions exactly like data, modifying them, and acting on the result.

In 1949 IBM introduced the Card Programmed Calculator (CPC), the concept for which originated at Northrop Aircraft. The CPC had a 604 as its arithmetic unit, a card reader to read the program from punched cards, a 480-digit electromechanical memory, a card punch, and a tabulating machine to print the output. It required a sorter, a collator, and a cart to move decks of cards around. This machine provided the means to do computing on a hitherto impractical scale, enabled users to develop computing methods and expertise, and increased the awareness of and interest in the potential of fully electronic, stored-program computers, which were just coming into existence.

The final precursor, the Test Assembly, was such a computer. This experimental unit was built in the IBM Poughkeepsie laboratory starting in 1949. It used a 604 as its electronic arithmetic unit, a second machine frame with 604-type circuits as logical elements to exercise control, a 250-word CRT memory that was a prototype of the 701 and 702 memories, a magnetic drum memory to store both data and instructions, and a card reader-punch for I/O. By solving a differential equation in 1950, the Test Assembly demonstrated that these components could be used to build a full-fledged, stored-program computer.

The 701 series

The 701 was the first large-scale electronic computer that IBM put into production. To make the machine quicker and easier to design, implement, and maintain, the architecture was kept simple and spartan. Many desirable features, such as floating-point arithmetic, were left to be done by programming. One quarter of the central processing unit (CPU) remained empty to give flexibility for correcting errors or adding function. The successor of the 701, the 704, provided the added function and filled up the space. Then there followed a succession of larger and faster, upward-compatible machines with new features as they were invented, culminating in the 7094 II and its smaller, simplified offshoots, the 7040 and 7044.

• *The 701*

The 701 was a parallel, binary, single-address, stored-program computer with a CRT memory of 2048 (later 4096) words of 36 bits [2–5]. It had 33 18-bit instructions with which to perform arithmetic and logic operations, sense and control its environment, and direct, word by word, the flow of information to and from

- two pairs of 1250-word-per-second magnetic-tape drives, adapted from designs developed for the TPM (702) project,
- four 2048-word magnetic drums, adapted from early designs for the 650 computer,
- a 150-line-per-minute printer,
- a 150-card-per-minute card reader, and
- a 100-card-per-minute card punch.

Instructions could address 18-bit half words or 36-bit full words. The instruction set was simple but carefully polished for efficiency and consistency. A 36-bit binary addition took five 12-μs cycles, while multiply and divide instructions each took 38 cycles.

The 701 was made binary, instead of decimal, because fast binary arithmetic was simpler and therefore less costly, yet it did not mean giving up the advantages of decimal and alphabetic input and output. Conversion between external decimal and internal binary representations and between punched-card and magnetic-tape codes could readily be handled by programming in a binary machine and did not require special translation circuits. The choice of binary arithmetic with its superior characteristics right at the start also set an important precedent: It might have been very difficult to introduce a binary machine later on if all of IBM's early computers had been decimal.

• *The 704*

In the 704, magnetic-core memory with from 4096 to 32 768 words replaced the CRT memory of the 701, thereby eliminating the most difficult maintenance problem and providing users with a more efficient way to run large programs. In the early 1950s, there was a widely held belief among theorists that 1000 words or so of memory should suffice. However, practical programmers seemed to need more than that. Also, the cost of providing larger memories kept going down as engineering and production methods advanced.

To accommodate the larger address space and other requirements, the instruction length was expanded to 36 bits. The instructions were similar in function to the 701 instructions but were incompatible, and their number was increased to 91. Floating-point arithmetic was provided. Three index registers were added for automatic address modification and for programming iteration loops. The combination of built-in floating point and indexing greatly improved performance.

An instruction for entering a subroutine put its location in an index register before branching. The subroutine could use indexed instructions to access parameters in the main routine and an indexed branch to return. This hardware implementation of a programming technique invented at the University of Cambridge made subroutines much more efficient and has become widely used. Many other instructions, too detailed to name here, contributed to the overall efficiency.

A CRT display was available, primarily as a graphic output device.

- *The 709 and 7090*

The 709 was basically a 704 with several additional architectural functions [6]. The most important of these were its data-synchronizer units (now called channels), which relieved the CPU of the need to control the transmission of individual words between memory and I/O units.

The 7090 was a transistorized version of the 709 with some added facilities. Like the change from CRT to magnetic-core memory, the change from vacuum tubes to transistors was one of technology and not of architecture. However, the effect of these two changes was so profound that it must be mentioned in the context of architecture. The 7090 was not just less costly to maintain and more reliable than the 701; it was so much better that it was a qualitatively different kind of machine.

The set of more than 180 instructions of the 709 and 7090 provided various other improvements. Among these were three convert instructions, which speeded up the translation from one number base to another and other kinds of access to tables of information.

- *The 7094*

In the 7094 the multiply and divide circuits were reworked to increase the speed of fixed- and floating-point operations and to provide a full set of double-precision floating-point arithmetic instructions. New instructions were added, bringing the total to 274.

The performance of the 701 series culminated in the 7094 II, which had a shorter cycle time, fewer cycles to multiply and divide, and memory interleaving. Fixed-point multiplication was 108 times faster than in the 701, and the speed on a problem using double-precision floating point had increased by a much larger factor.

- *The 7040 and 7044*

These machines provided a less costly alternative to the 7090 and 7094 for users whose requirements could be met by a simpler machine. They had a basic subset of 49 instructions as well as six sets of optional instructions that could bring the total up to 120 as compared to 180 on the 7090 and 274 on the 7094. The machines differed only in speed, with an 8-μs cycle on the 7040 and a 2.5-μs cycle on the 7044. A few 7040/7044 instructions were not part of the 7094 instruction set; thus, writing compatible programs required avoiding these improvements.

A novel combination, the Direct-Coupled System (DCS), was an early asymmetrical multiprocessing system that was very successful. It used a 7040 or 7044 as a front-end unit to handle I/O and to schedule jobs, and a 7090 or 7094 to do the computing. Combinations similar to DCS are still in common use with System/370 computers.

- *Special system for CTSS*

A particularly significant modified 7090, and later 7094, made it possible to create the first major time-sharing system: the Compatible Time-Sharing System (CTSS) at M.I.T. A second 32 768-word memory bank was provided. User programs occupied one bank while the operating system resided in the other. The user memory was divided into 128 memory-protected, 256-word blocks, and a user could not refer to a block outside of his allocation. The contents of a 7-bit relocation register were added to the most significant bits of each address, so that each user programmed as if his allocation started at address zero. There was an interrupt clock that enabled the system to control and record the time allocated to each user.

The 702 series

The 702 project started in late 1949, when design was begun in the IBM Poughkeepsie Laboratory on an engineering model, then called the Tape Processing Machine (TPM). Work was slowed considerably in 1951, when much of the TPM engineering staff was assigned to the higher-priority "Defense Calculator" (701) project because of the Korean War. The TPM model became operational in 1952; it was revised and announced as the 702 in 1953. Thus, the 702 project started earlier than the 701 but finished later.

- *The TPM and 702*

As its name implied, the Tape Processing Machine was intended to provide a data processing system with the speed and functional advantages of a new storage medium, magnetic tape, replacing punched cards. Its decimal arithmetic and logical unit, CRT memory, and magnetic-tape units are described in detail in a patent filed in 1954 and issued in 1966 [7].

The TPM was substantially redesigned to achieve greater performance and reliability, its instruction set was revised while retaining much of the basic architecture, and it became the 702 [8, 9]. Serial-by-bit, serial-by-character operation was changed to parallel-by-bit, serial-by-character operation, which turned out to provide higher performance at lower cost. The single-address instructions operated on one operand in CRT memory and another operand in a second CRT storage device called an accumulator. Instruction execution times of the 702 varied greatly depending on the number of characters involved. Numbers of five decimal digits could be added in 0.25 ms, whereas multiplying ten by ten digits took 3.36 ms.

Several significant innovations especially useful for business data processing were introduced by the TPM and 702. These are now briefly discussed.

Variable field and record length In contrast to the 36-bit word orientation of the machines that formed the 701 series, the architecture of the 702 series was oriented towards alphanumeric characters. A character was encoded as six bits and an even-parity check bit that was used to detect data errors going to and from memory and on tape. An instruction could directly address any character in memory and process a field of arbitrary length; both the memory operand and the operand in an accumulator could be up to 511 characters long. Likewise, records on magnetic tape could be of arbitrary length. One disadvantage was that fields and records were delimited by codes within the data, which imposed restrictions on the character set. This data-marking technique was subsequently replaced, in Stretch and in System/360, by having the program specify lengths explicitly.

Collating sequence Sorting and merging of files of alphanumeric data require an appropriate "collating sequence," which governs the low-to-high sequencing of alphabetic, numeric, and other characters. A compare instruction was designed to provide a particular collating sequence which was already well established for punched-card collators, so that existing file sequences could be maintained when converting from punched cards to magnetic tape.

Editing instructions Ease of data editing was a major criterion for selecting the 702 instruction set in general and variable field length in particular. One special instruction inserted commas and periods and suppressed leading zeros in numeric fields; another instruction facilitated the insertion of floating dollar signs and check-protection asterisks. Although the card reading, punching, and printing units were derived from standard punched-card equipment on which editing was controlled by plugboard wiring, their plugboard facilities were omitted on the 702 devices. All editing was to be done by programming so as to minimize operator intervention.

Competent I/O Most early non-IBM computers had only primitive I/O equipment, such as paper-tape readers and punches, or magnetic-tape devices connected to keyboards and typewriters. It was considered essential for business applications to provide more power and flexibility: high-speed magnetic tape for file storage, card readers for efficient data entry, card punches for producing machine-readable output documents, and high-quality line printers at least equal to the best in use in punched-card installations. The variety of I/O needed for different applications required flexibility in attaching the I/O devices; this aspect is discussed in a later section.

It soon became evident that CPU speed was more important in a data-processing environment than had been thought at first. The "housekeeping" required to keep just the I/O running at full speed consumed a great deal of CPU time. Improved handling of large tape files was also a concern, because much machine time was being spent on sorting, merging, and searching of files. The need for more speed and the desire to replace the CRT memory and accumulator storage with faster and more reliable magnetic cores prompted an early development of a 702 successor. At the same time, considerable effort was spent on developing an auxiliary tape sorting and collating machine, tentatively called the 703, which was intended to relieve the main computer of much of its tape-handling burden.

• *The 705 models*

The first 702 successor was the 705, announced in 1954, which provided the following significant improvements:

- The memory size was increased from 10 000 characters to 20 000 (705 I) and later to 40 000 (705 II). The 702 addressing scheme had to be modified to accommodate the larger memory.
- All the variable-field-length instructions with addressing of individual characters were retained, but instruction fetching and data moving within memory by means of some new instructions were done five characters at a time.
- Tape reading could be overlapped with tape writing.

The redesign and the nonvolatile core memory provided substantial performance improvements. Addition of five digits took 119 μs, and ten-by-ten multiplication used 2.41 ms. The 705 also proved to be sufficiently better than the 702 at handling tape that the plan to announce the specialized 703 tape sorter-collator was dropped.

Although the 705 instruction set was largely the same as that of the 702, it was different enough that 702 programs could not be run on the 705. With the successors to the first 705 model, however, compatibility became an important consideration because of the inconvenience to users of having to convert existing programs. Even minor differences, such as new functions which only made use of previously unassigned bits of an instruction, were found to cause problems, and a ''compatibility mode'' was needed to be able to run programs that had used these bits for their own purposes. The conversion problem was more significant than it would be today, because high-level languages were still in their infancy, and most programming was done at or close to the machine-language level.

The 705 III, announced in 1957, provided up to 80 000 characters of core memory, indirect addressing, and one or more ''data-synchronizer'' units to permit overlap of tape reading and writing with computing, using main memory as the buffer. It added five digits in 86.5 μs and did ten-by-ten multiplication in 1.62 ms. Several instructions were added to do bit handling, to simplify address computations, and to blank a memory area. The 705 III was capable of running 705 I and 705 II programs with only minor exceptions.

- *The 7080*

The 7080 was a transistorized version of the 705 III. It provided up to 160 000 characters of memory, a set of I/O channels to replace multiple data-synchronizer units, a single-level interrupt facility to handle I/O, and compatibility modes to run 705 I-II or 705 III programs. It could perform a five-digit addition in 11 μs and a ten-by-ten multiplication in 265 μs.

The 650 series

- *The 650*

In 1948, when the engineers from the IBM Endicott laboratory who designed the SSEC had completed that project, they turned their attention to the design of a new and much smaller computer that would be suitable for volume production. From the outset, the emphasis was on reliability and moderate cost, which led to the choice of a magnetic drum as the central storage medium and to a high degree of checking throughout. The result was the 650, which was announced in July 1953 [10, 11].

The 650 was a serial, decimal, stored-program computer. A word, which had a fixed length of ten decimal digits and a sign, could contain one instruction or one decimal number. Decimal digits were represented internally in a seven-bit (biquinary) self-checking code, and on the drum in a five-bit (2-out-of-5) self-checking code.

The magnetic drum, which had a capacity of 2000 words, rotated at the unusually high speed of 12 500 rpm. The drum also served as an I/O buffer for the card reader and punch, the only I/O units provided initially. The high speed of the drum kept the maximum time of access to instructions or data to 4.8 ms. To allow still further reduction of the effective access time, a two-address instruction format was chosen, one address for the operand and the other for the next instruction to be executed. With the aid of the Symbolic Optimal Assembly Program (SOAP) developed for the 650, a program could be optimized so that the data or the next instruction would be close to the read-write heads at just the time they were needed, thus avoiding a long wait of up to one whole revolution for each. Such optimum programming could bring the time for an add instruction from an average of 5.2 ms down towards the minimum of 0.8 ms.

The basic 650 with its card equipment was very successful. Later it was expanded with alphabetic capability, using two digits to represent one alphanumeric character, and with on-line printing and magnetic tape. Also, a 600-digit core storage was added as a high-speed I/O buffer.

The most important addition was a magnetic disk unit, the high-capacity storage device that had been developed at the IBM San Jose laboratory as an integral part of a smaller machine, the 305 RAMAC [12]. The 650, whose version of the disk unit had a storage capacity of six million digits, thus led the way in attaching disks to general-purpose computer systems. The 650 also acquired terminals, primarily for inquiry and response applications. This combination of magnetic disks and terminals was important as it constituted the beginning of a major shift from tape to disk for secondary storage and from batch to transaction-oriented processing.

With the addition of these I/O capabilities, the 650 was no longer solely a card-processing system but had become an intermediate data-processing system with considerable flexibility.

- *The 7070*

The successor to the 650 was the 7070, which was a transistorized machine with up to 9990 ten-digit words of core memory [13, 14]. Because of the random-access core memory, the two-address instruction format of the 650 was no longer needed and was replaced by a single-address format. Decimal digits were represented in a five-bit (2-out-of-5) code throughout. To make these basic changes possible, the 7070 was not constrained to be compatible with the 650.

By omitting the second address in an instruction, space was freed up for two additional functions. Two digits were used in indexing to select one of 99 index words located at the low end of memory. Two more digits constituted the field definition, which allowed a portion of a ten-digit word to be processed by a single instruction.

The 7070 transferred words in parallel between memory and the CPU, but it processed them serially, digit by digit. Addition of two ten-digit numbers required 72 μs, multiplication 924 μs. Floating-point decimal arithmetic was also available.

To permit overlap of CPU processing with multiple input and output operations, the 7070 had an I/O interruption capability called priority processing. It allowed a main program to be interrupted when an I/O device required service, which was provided by switching to a "priority routine" that was not interruptible. Priority processing was used to perform peripheral operations, such as card-to-tape and tape-to-printer conversion, on the main computer instead of an auxiliary system. The term SPOOLing (Simultaneous Peripheral Operation On Line) was coined for this function, which represented an early form of multiprogramming.

A block-transmission facility provided for movement of blocks of data, both within memory and between memory and I/O, in a single operation. The blocks, which did not have to be in contiguous memory locations, were described by a chain of record-definition words.

- *The 7074 and 7072*

The 7074 was a faster version of the 7070, with storage increased up to 30 000 words [15]. It had the same instruction set except that, when using the additional storage, the 7074 had to be switched to a different addressing mode. Parallel arithmetic allowed the 7074 to add ten-digit numbers in 10 μs and multiply them in 56 μs, considerably faster than the 7070. The 7072 was a lower-cost, tape-only version of the 7074 oriented towards numerical applications.

The 1401 series

The upward growth of the 650 series and the rapid progress in semiconductor technology during the mid-1950s left a void at the low end, which was filled by the 1401 series. The 1401 was initially intended as a stand-alone stored-program computer to provide faster card reading, card punching, and printing equipment than did earlier electromechanical punched-card machines. Its chain printer, which combined a high speed of 600 lines per minute with high print quality, was a technological breakthrough. This printer, with some modifications, is still a mainstay of the IBM product line. Since the high-speed reading, punching, and printing devices of the 1401 were also valuable to users of the larger 7000-series computers, it was decided to provide a magnetic-tape version as an auxiliary system for off-line card-to-tape, tape-to-card, and tape-to-printer operations. Both versions of the 1401, which were announced at the same time, became eminently successful.

- *The 1401*

The architecture of the 1401 was character-oriented and resembled that of the 702 in several respects, but there were important differences. The 1401 used two-address, memory-to-memory instructions; data flow was serial-by-character, with memory cycles alternating between the two operands. Both data and instructions were variable in length. Characters were represented by eight bits: six for data, one for odd-parity checking, and one for a "word mark" used to define the high-order position of each word. Data movement and manipulation were checked on all operations: thus numeric data were converted to a self-checking biquinary code before any arithmetic operation.

The 1401 had 34 different operation codes. An additional character could modify many of these operations, so that the total number was effectively much larger than 34. Operation codes were one character in length and addresses three; a typical two-address instruction was seven characters long. Not all instructions had two addresses: Some had no addresses and others one, so there were also one- and four-character instructions. Similarly, instructions with a modifier character could be two, five, or eight characters in length. "Chaining" of instructions was an interesting capability: Where consecutive instructions were used to add or move words located sequentially in storage, the first instruction with the usual two addresses could be followed by instructions consisting of only the operation codes.

Editing of numeric fields for printing was accomplished in a manner similar to the editing facility of the 702, but additional function and flexibility allowed each field to be edited completely with only one instruction.

The word mark, probably the most distinctive architectural feature of the 1401, in conjunction with the memory-to-memory logic, minimized the electronic circuitry requirements. Only two single-character data registers and three memory-address registers were used in the basic machine. Variable word length for both data and instructions reduced storage requirements very significantly.

Core memory was initially offered in capacities of 1400 (shown by studies to be adequate for card-accounting jobs) to 4000 characters. For more complex applications, the memory capacity was later increased to 16 000 characters. Addressing of 16 000 characters with a three-character address was accomplished by binary encoding of previously unused bits of the characters. Two other available address bits were used for indexing; that is, they could select one of three storage locations set aside as index registers.

The memory cycle time was 11.5 μs. Addition of five-digit fields took 230 μs, and multiplying two ten-digit fields required 6.9 ms.

The 1401 tape system was widely used, both as an auxiliary system for the larger computers and as an independent tape processor. Several special features were added to handle tapes for these different computers, thus bringing into focus the inconsistencies among them. Operation with even- and odd-parity tape was required. A column-binary feature allowed binary cards to be read or punched, and converted to or from tapes, for use on the 704/9 and 7090/4 machines. Similarly, special provisions were made to handle 7070 tapes. On the 1401 itself, the unique eighth word-mark bit could not be written directly on the seven-track tape that was common to all these machines, so this bit was translated to a word-separator character, which created another special tape format.

As applications for the 1401 expanded, numerous devices and optional features were added. Attachment of magnetic disk provided the third major 1401 version, the disk system, which was as well received as the card and tape systems. Processing could be overlapped with I/O, which gave significant performance improvement. Paper tapes, magnetic and optical character readers, and numerous other I/O media were attached via a serial input/output adapter, which also allowed channel-to-channel communication with other computers.

- *The 1440 and 1460*

The 1440 was a smaller, and the 1460 a larger, successor machine with the same architecture as the 1401 and with the same memory capacity of up to 16 000 characters. The 1440 and 1460 used newer circuits and had much of their electronics in common. The 1440 had an 11.1-μs memory cycle, giving it approximately the same internal speed as the 1401, but simpler and slower I/O. The 1460 with a 6-μs memory cycle had almost twice the processing speed, and up to three 1100-line-per-minute printers could be attached.

- *The 1410 and 7010*

The 1410 was a considerably larger and more powerful machine than the 1401. The memory cycle was reduced to 4.5 μs (4.0 μs optionally). The architecture was based on that of the 1401, but addresses were expanded to five characters to allow the memory capacity to be increased. Up to 80 000 characters of memory were provided. Zone bits were used to select one of 15 index registers. Two address registers were added to speed up arithmetic operations. The instruction set was enhanced considerably by expanding some instructions and adding other functions, such as table look-up. An optional I/O interruption feature, similar to that of the 7070, was available. The 1410 could run 1401 programs by switching to a 1401 compatibility mode.

At the top of the 1401 series was the 7010; it was fully compatible with the 1410. It had a memory of up to 100 000 characters, a 2.4-μs memory cycle, and two-character access to storage.

Stretch

The Stretch system, formally known as the 7030, played a special role in the evolution of IBM's computers [16]. It is best known as a "super-computer" developed for the Los Alamos Laboratory of the U.S. Atomic Energy Commission. A "Harvest" extension (the 7951) was designed for the National Security Agency [17]. Unlike IBM's earlier super-computer, the one-of-a-kind NORC (Naval Ordnance Research Computer) [18], the Stretch system went into production, though in limited numbers. More significant, however, was that much of the transistor, magnetic-core, circuit, and packaging technology of the Stretch project was carried forward to other 7000-series and 1401-series computers, and many of the architecture innovations were later applied to System/360.

In late 1954, a project was started at the IBM Poughkeepsie laboratory to develop a new, all-transistor computer, which was to benefit from the experience gained with the 701, 702, and 650 series that were marketed to all customers and with the SAGE real-time control computer [19, 20] that was marketed to the military. Subsequently it was decided to set the very ambitious goal of one hundred times the overall performance of the 704. The need for such performance was clear, as was the fact that the technology available at the time was not adequate. New transistors, circuits, magnetic cores, and design and manufacturing techniques would require the utmost technical effort, hence the name *Project Stretch*.

The technology which was developed especially for Stretch included drift-transistor circuits with a 20-ns delay time, multiple 128K-byte memory units (where K = 1024) with a 2.1-μs cycle time, and high-speed magnetic disks. This was combined with new organizational techniques, such as instruction look-ahead and memory interleaving. Error correction was used in memory and on the disks, and errors were checked extensively throughout the system.

Project Stretch introduced a number of new or substantially enhanced functions into computer architecture, and it brought new terms, such as "byte," "I/O interface," and the word "architecture" itself into the language of computers.

Addressing The architecture took advantage of binary addressing to provide storage subdivisions (words, bytes, and bits) of lengths that are powers of two. This choice greatly facilitated the design of variable field length, indexing, address arithmetic, and instruction formats. Addresses were up to 24 bits in length and permitted direct addressing of 262 144 or 2^{18} words of 64 bits (2048K bytes) in storage, and of any bit within those words.

Storage protection Protection against incorrect storage accesses by the program was provided by a pair of address-boundary registers.

Binary and decimal arithmetic Both binary and decimal arithmetic were included: binary arithmetic for speed and for address computation, and decimal for efficient data handling.

Variable field length Fixed-point arithmetic, comparison, and logical operations were provided for fields from 1 to 64 bits in length. They facilitated operations on decimal numbers, character strings, logical bit strings, single-bit indicators, and parts of instruction formats of various lengths, including binary addresses.

Variable byte size To operate on subdivisions (bytes) within a field, such as 4-bit decimal digits and 6-bit or 8-bit alphanumeric characters, variable-field-length instructions could specify a byte size of from 1 to 8 bits. For other than variable-length fields, the byte size was fixed at 8, a power of 2. The 8-bit byte also permitted a larger character set that included upper and lower case.

Floating-point arithmetic Much care was taken in the specification of the binary floating-point arithmetic to provide a complete, consistent, and easily programmed instruction set. Particular attention was paid to the control of precision loss and round-off errors, and to instructions which simplified the programming of multiple-precision arithmetic.

Indexing Each of the 16 index registers contained an index word of 64 bits, which provided not only the index value but also a loop count and a chain field. The combination could be used, among other things, to specify a chain of memory areas. I/O operations employed the same format for their control words. Consequently the same index or control word could be used to control reading of input data, then to index through the data during processing, and finally to control writing of the modified data to an output device.

Interrupts A single interrupt scheme provided a systematic method for responding to asynchronous I/O requests, time signals, and machine errors, as well as to synchronous program exceptions. The interruption conditions and other conditions which could not cause interruptions were all collected in one 64-bit indicator register. These bits could either be interrogated by a conditional branch instruction or, under the control of mask bits in a second register, cause execution of an instruction in an interrupt table. That instruction might be a branch to an interrupt handler or simply a one-instruction fix-up.

Clocks An interval timer provided an interrupt condition when a time interval had elapsed. A separate, continuously running clock permitted the programming of time-of-day indications.

Input/Output Up to 32 I/O devices could be operated simultaneously through the "exchange," a unit which corresponded to a 32-way byte-multiplexer channel in later terminology. Hardware to control data transfer was shared among many I/O devices. The exchange was completely device-independent, as were the controlling I/O instructions.

Consoles The operator console was connected via a channel and treated as an I/O device. Switches and lights were just so many bits of input and output, the meanings of which were determined entirely by the program. More than one console could be attached.

Multiprogramming Several of these architectural facilities—storage protection, interrupts, clocks, and program-interpreted multiple consoles—were intended specifically to make it possible and practical to use a large computer not only on single large problems, but also on multiple smaller programs simultaneously.

From the beginning the intent was to develop a very general architecture that would be well-suited to many different types of applications. It had been planned to make the high-speed, fixed-length arithmetic unit separable from the variable-length processing unit, so that the latter could subsequently be marketed as a lower-performance commercial product using the same instruction set. Indeed, some aspects of the architecture, such as variable field length, were primarily aimed at manipulation of alphanumeric data, since numeric computing had less need for them. Thus, it was thought that a single line of compatible machines, tentatively called the 7000 Series, would replace IBM's various earlier incompatible computers. This part of the architectural plan was not realized because of subsequent engineering and marketing decisions. Such a single line of architecturally compatible machines did not make its appearance until System/360 arrived.

For typical 704 programs, Stretch fell short of its performance target of one hundred times the 704, perhaps by a factor of two. In applications requiring the larger storage capacity and word length of Stretch, the performance factor probably exceeded one hundred, but comparisons are difficult because such problems were not often tackled on the 704.

It was decided to limit production to the Los Alamos machine (delivered in 1961), the Harvest system (delivered in 1962), and seven more systems that were on order at the time. Development of additional compatible versions was discontinued. Instead, the 7000 Series came to include various machines other than Stretch (7030); these used its architectural ideas and high-speed technology but not the plan for a line of machines with compatible architecture.

Early input/output architecture

The primary challenges of I/O architecture in IBM's early computers were quite similar to those which have faced system designers and architects up to the present day. These are

- To provide for efficient data flow between a very high-speed, precisely clocked processor/memory complex and a number of larger-capacity but (usually) slower and (usually) less precisely clocked I/O devices.
- To provide the necessary control for those devices, allowing for a wide variety of individual characteristics such as start/stop times, data formats, and media-handling requirements.
- To achieve these data and control functions with, as nearly as possible, a common set of computer instructions for all I/O devices.
- To synchronize these operations with the internal processes of the computer in a manner which is acceptably economical in its use of both hardware and programming.

- *The 701: Minimizing hardware*

The arrangements in the 701 for handling I/O data were rather spartan, and the stored program in the CPU was involved in the transfer of every word (36 bits in this case) between CRT memory and an input or output device. Several instruction types in the 701 repertoire were provided specifically for I/O operations, and three of these will be described.

The instruction PREPARE TO READ (READ, for short) caused an input device, as specified by the address part of the instruction, to be selected for reading. The selected device could be one of four magnetic-tape drives, or four logical magnetic drums (portions of physical drums), or the single card reader. PREPARE TO WRITE (WRITE) followed similar rules with some rather obvious differences. COPY AND SKIP (COPY) was the instruction which caused each word of input or output data to be transmitted between the multiplier-quotient (MQ) register of the CPU and the location in memory specified by the address part of the instruction. In general, the program had to have a COPY instruction waiting to be executed whenever the input or output device, depending on its timing, required access to a memory location as a destination or source for a word of data.

Depending on the input device (and these same comments apply to output), varying amounts of time between successive COPY instructions were available to the CPU for computing. A COPY instruction, encountered after the last word of the record had been placed in memory, would cause an end-of-record skip of the next two instructions. This skip permitted the program to initiate an instruction sequence appropriate to the end-of-record condition. Similarly, an end-of-file skip of just a single instruction permitted branching to an end-of-file sequence. It can be seen that the necessary synchronization between operation of the CPU and of the I/O system depended to a great extent upon the proper use of the COPY instruction.

The card reader operated in row-by-row fashion, but on only 72 of the 80 columns, so that a card row could be treated as two 36-bit words. The buffering between a card and memory consisted of 72 vacuum-tube flip-flops and the MQ register, all in the CPU, which was a highly CPU-integrated I/O attachment scheme. The card punch and the printer (which electrically resembled a card punch) were attached, and output data supplied to them, in a very similar manner.

The magnetic drums provided data at a rate of one word every 1.28 ms after an initial access time of several milliseconds. Thirty-six tracks were involved simultaneously in the transfer of a word. The sequential data rate of the drums could have been much higher; only one word out of every 128 was read or written. This deliberate slow-down gave the CPU time to perform the calculations needed to deal with the data, including generating subsequent COPY addresses. It also provided time for the regeneration cycles required by the CRT memory of the CPU. One of the most interesting aspects of the drum unit of the 701 is that it was included at all. This was an early, if not the earliest, use of magnetic drums as auxiliary storage in a large-scale computer whose main memory was of a different, faster type.

To summarize, the I/O organization of the 701 was physically simple. It was uncommitted to the transfer of fixed blocks of data or even to dealing with sequential memory locations within an I/O operation. Whenever possible, existing mechanisms were used with a minimum of additional electronics, relying on the CPU program and hardware to carry out housekeeping functions that would later be taken over by such system components as control units and channels.

- *The 702: Simplifying programming*

The I/O capabilities of the 702 illustrate an alternative I/O architecture to that of the 701, in a machine of comparable size and capability. Its planning reflected the assumption that potential users would be most attracted to a computer that dealt with data in the familiar form of existing business records, and one that required the fewest program steps to perform a typical data processing job. An architectural feature which clearly illustrated this orientation was the manner in which READ and WRITE instructions operated. Having previously specified the desired I/O device with a SELECT instruction, the programmer used the address part of the READ or WRITE instruction to designate the location in memory to or from which the first character of an input or output record was to be transferred. Thereafter, as soon as the selected device was available, an entire record was read or written without requiring additional instructions such as the COPY instruction of the 701.

This technique would have left the CPU and memory idle much of the time unless the I/O data rate was high enough. The magnetic-tape data rate, at 15 000 characters per second, came fairly close, being a little over one-third the internal memory speed. The magnetic drum at 25 000 characters per second was a better match, as it used more than half of the available 23-μs memory cycles. For slower I/O units, which included card readers, card punches, and line printers, magnetic-core buffer storage was provided to increase the data-transfer rate and to avoid delaying the CPU while waiting for one of these devices to go through a substantial portion of its mechanical cycle. The rather modest-sized, magnetic-core buffer memory units of the 702, which were delivered starting in February 1955, were among the first coincident-current, magnetic-core storage units produced by IBM.

- *Buffer storage for tape*

As magnetic-tape applications grew in importance, it became clear that keeping the CPU of the 702 busy one-third to one-half of the time was not enough and that some provision for overlapping among tape input, tape output, and processing was needed. At first a small number of magnetic-core buffers were designed and provided for use with 702 magnetic-tape units on a special-engineering basis. Following the 702, the tape buffer idea was kept alive for some years in record storage units made available as regular products for use with the 705. Initially the 705 also provided instructions which overlapped tape reading with writing, but not with processing.

Overlapping of all three was first obtained by means of the Tape Record Coordinator, announced in 1955 for use with the 705. It contained a 1024-character buffer storage, which allowed it to read and simultaneously write a master file on tape while searching for "active" records in the file. Only the active records were passed to the 705 CPU for processing; the rest of the time, the 705 program could proceed independently.

- *Flexibility of attachment*

Right from the start it was considered important to permit a system like the 702 to be tailored to the needs of each installation and to allow for the use of new devices whose characteristics could not be anticipated. Thus any reasonable number of each type of device could be attached, including none. All devices regardless of type were attached via their control units to the CPU and memory by a common cable connection. Circuits peculiar to a particular device type were contained in the appropriate control unit rather than in the processor. This attachment method was the first version of what later became known as a standard I/O interface.

Other systems contributed to the evolving standard interface. The serial input/output adapter of the 1401 permitted attachment of any of (eventually) a wide variety of serial-by-character I/O devices. A major contribution was the Stretch I/O interface, which further generalized the attachability of devices to the system through its "exchange," and included operator consoles, typewriter inquiry stations, and communication terminals in the list of devices attached in this standard fashion.

In the 702 system, the control units for the card reader, punch, and printer were equipped for either direct attachment to the CPU (on-line) or attachment to a tape drive (off-line). Thus the user had a choice of the greater computational efficiency of off-line operation, at the expense of additional delay and tape handling, or the better error control and shorter turnaround time of on-line operation, at the expense of throughput.

The concept of off-line control units underwent an interesting evolution. If a new printer, for example, was equipped with a tape-to-printer control unit, the printer could be used in conjunction with any computer which could furnish data on magnetic tape in a compatible format. This approach avoided both the engineering expense of attaching the printer to the computer and the inefficient use of the computer in feeding data directly to a printer.

But, as mentioned before, when the 1401 was announced in 1959 as a low-end computer system, it was found economical to use the 1401 also as an off-line control unit (with considerable editing capability) to make its superior printing and card-handling equipment available to users of the larger systems of the 7000 series. Thus the 1401 was IBM's first intelligent control unit, an alternative to the "spooling" mentioned in the 7070 description.

- *I/O channels*

As late as 1957, computers were seriously limited in their ability to utilize their processing capabilities during I/O operations, or to perform more than one input or output operation at a time. In the 701 or 704, instructions could be executed during the transfer of a record between memory and the I/O device, but the involvement of the CPU program and the MQ register in the transfer of every word severely restricted the amount of such processing. In the 702, processing was halted during the entire transfer of an I/O record between memory and the device or its buffer storage. Later refinements, such as the Tape Record Coordinator and the read-while-writing feature of the 705, made possible concurrent input and output operations.

Although the 709, one of the last of the vacuum-tube computers announced by IBM, had a rather short technological life, it made an important contribution to concurrent CPU and I/O operations by the introduction of I/O channels [21]. These channels acted individually as I/O processors with a specialized instruction repertoire. Each channel could address and access memory to store or retrieve data quite independently of the CPU program. Coordination between the CPU and channel programs was achieved by CPU instructions for (1) conditional branching depending on whether a channel was in operation, (2) delaying the CPU until completion of a channel command, followed by loading of channel control registers, and (3) storing the channel control register contents in a specified memory location.

For the 709, two channels were packaged in a unit called the data synchronizer, and three such units could be attached to a 709 CPU. Each data synchronizer could handle up to 16 magnetic-tape drives and one card reader, card punch, and printer. With a full complement of six channels, six of these I/O devices could be operating at the same time. (Magnetic-drum and CRT-display units operated independently of the channels.)

It was the parallelism provided by the channels of the 709, as much as any other factor, that led to the development of a supervisory program called the I/O control system (IOCS). It would have been inefficient, when programming each application, to redesign the detailed channel programs and the routines for synchronizing the CPU and channels. The IOCS introduced with the 709 represented one of the early steps in the evolution of the operating system.

Channels similar to those of the 709 were later provided in the 705 III and in most subsequent IBM computers. An equivalent function was introduced through the processing-overlap feature on the 1410, a simplified version of which became available on the 1401. These 1400-series "channels" were integrated into the CPU by means of special registers for holding I/O data and addresses. They required special instructions, but they were not controlled by separate channel programs as in the 700 series and its successors.

All of these channels or equivalent features involved "cycle stealing," a technique for accessing memory in a manner transparent to the CPU program, which had been employed several years earlier in the SAGE computer.

- *I/O interruption*

The instruction loops required on the 709 to test for the status of overlapped I/O operations, and thus synchronize CPU and channel programs, were found to be generally cumbersome and inefficient. A much more direct method is to interrupt the CPU at the end of an I/O operation, or upon the occurrence of exceptional conditions, and to cause the CPU program to branch to an instruction sequence designed to take the desired action. Early versions, such as the data-channel trap of the 7090, were later replaced by more general program-interruption schemes. All of the 7000-series systems, announced between 1958 and 1963, included both I/O channels and program interruption in varying degrees.

The most sophisticated interruption scheme was developed for Stretch, which combined interruptions for I/O and other external causes with the management of program exceptions. It was a direct precursor of the I/O channel and interruption architecture of System/360.

Conclusion

By the end of the fifteen-year period covered by this paper, the computer industry had grown from almost nothing to one of the most important, IBM had changed from being principally involved in punched-card machines to being mainly a supplier of computers, and computers had begun to change the way man dealt with the world.

IBM had several successful lines of computers, but this success also presented a problem. The total effort required to provide systems and application programming for such different computers had become large and could be predicted to become unmanageably so. In the early 1960s, computers were understood much better than in the early 1950s, yet these lines of upward-compatible computers were still constrained to architectural decisions that had been made up to fifteen years earlier. The time was now ripe to replace these multiple older lines of computers with System/360: a group of machines built to a single architecture, that took much better advantage of what had been learned since the beginning of the computer era.

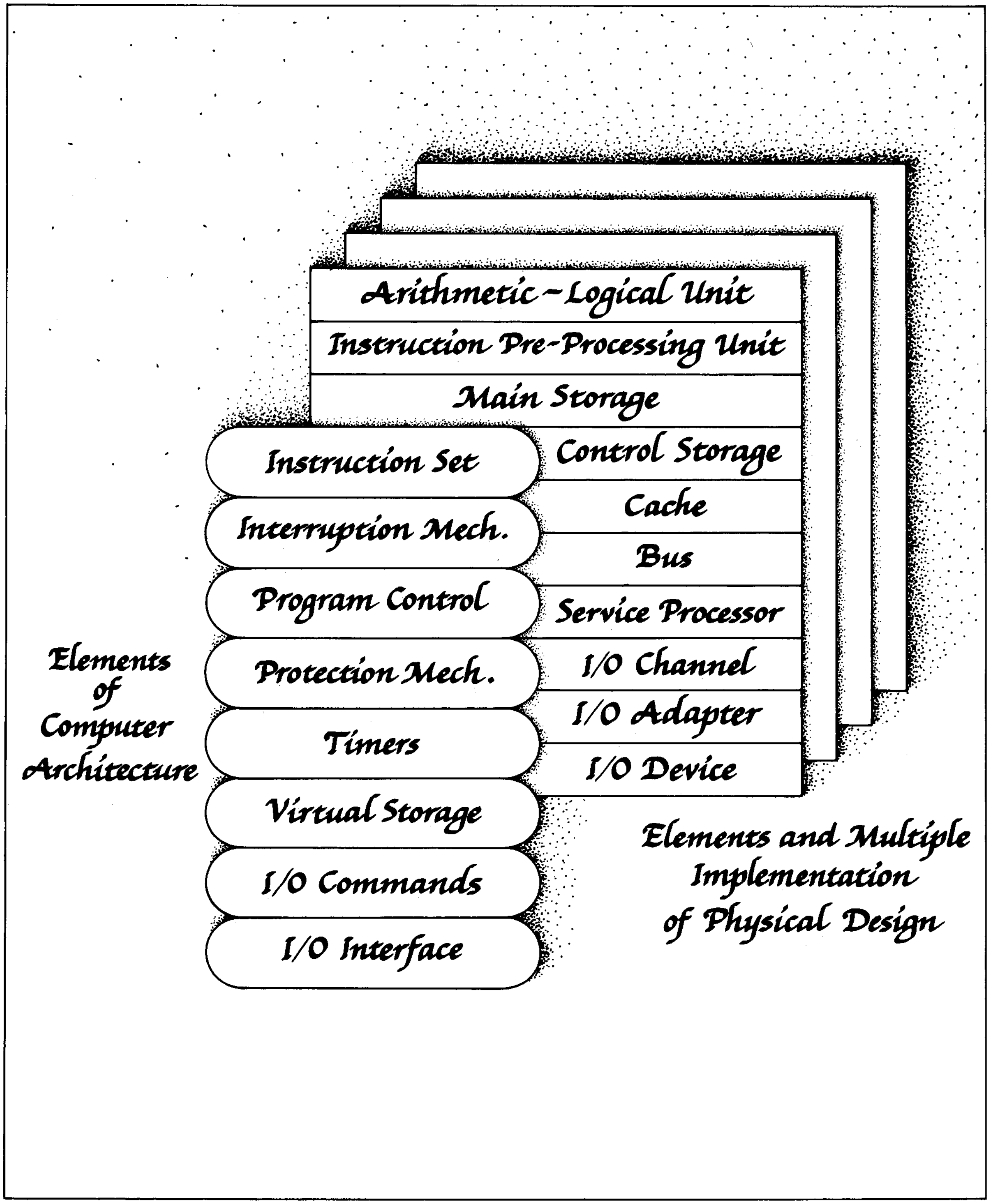
A key factor in finalizing the transition from the older series to System/360 was the advent of emulation. Emulation was a combination of microprogram and software simulation of the various instruction sets, which allowed users to continue to run existing programs efficiently on the new hardware. Thus, at the end of IBM's first fifteen years of developing computers for the marketplace, a new evolution was starting, one which has not ended to this day.

References

1. For additional information concerning precursor machines, see Byron E. Phelps, "Early Electronic Computer Developments at IBM," *Annals of the History of Computing* **2,** 253-267 (1980).
2. L. D. Stevens, "Engineering Organization of Input and Output for the IBM 701 Electronic Data Processing Machine," *Proceedings,* Joint AIEE-IRE-ACM Computer Conference, New York, December 1952, pp. 81-85.
3. M. M. Astrahan and N. Rochester, "The Logical Organization of the New IBM Scientific Calculator," *Proceedings of the Electronic Computer Symposium* (IRE), Los Angeles, April 30-May 2, 1952, paper XIX (7 pp.).
4. W. Buchholz, "The System Design of the IBM Type 701 Computer," *Proc. IRE* **41,** 1262-1275 (1953).
5. C. E. Frizzell, "Engineering Description of the IBM Type 701 Computer," *Proc. IRE* **41,** 1275-1287 (1953).
6. J. L. Greenstadt, "The IBM 709 Computer," *Proceedings of the Symposium: New Computers, a Report from the Manufacturers* (ACM), Los Angeles, March 1957, pp. 92-96.
7. N. Rochester, C. J. Bashe, W. Buchholz, R. P. Crago, P. E. Fox, J. A. Haddad, and B. E. Phelps, "Electronic Data Processing Machine," U.S. Patent 3,245,039, April 5, 1966 (448 pp.).
8. C. J. Bashe, W. Buchholz, and N. Rochester, "The IBM 702, an Electronic Data Processing Machine for Business," *J. ACM* **1,** 149-169 (1954).
9. C. J. Bashe, P. W. Jackson, H. A. Mussell, and W. D. Winger, "The Design of the IBM Type 702 System," paper no. 55-719, *AIEE Transactions* **74** (Part I, Communication and Electronics), 695-704 (1956).
10. E. S. Hughes, Jr., "The IBM Magnetic Drum Calculator Type 650, Engineering and Design Considerations," *Proceedings of the Western Computer Conference*, Los Angeles, February 1954, pp. 140-154.
11. F. E. Hamilton and E. C. Kubie, "The IBM Magnetic Drum Calculator Type 650," *J. ACM* **1,** 13-20 (1954).
12. M. L. Lesser and J. W. Haanstra, "The Random-Access Memory Accounting Machine—I. System Organization of the IBM 305," *IBM J. Res. Develop.* **1,** 62-71 (1957).
13. R. W. Avery, S. H. Blackford, and J. A. McDonnell, "The IBM 7070 Data Processing System," *Proceedings of the Eastern Joint Computer Conference*, Philadelphia, December 1958, pp. 165-168.
14. J. Svigals, "IBM 7070 Data Processing System," *Proceedings of the Western Joint Computer Conference*, San Francisco, March 1959, pp. 222-231.
15. R. R. Bender, D. T. Doody, and P. N. Stoughton, "A Description of the IBM 7074 System," *Proceedings of the Eastern Joint Computer Conference*, New York, December 1960, pp. 161-171.
16. W. Buchholz, Ed., *Planning a Computer System (Project Stretch)*, McGraw-Hill Book Co., Inc., New York, 1962.
17. S. S. Snyder, "Computer Advances Pioneered by Cryptologic Organizations," *Annals of the History of Computing* **2,** 60-70 (1980).
18. W. J. Eckert and R. Jones, *Faster, Faster*, McGraw-Hill Book Co., Inc., New York, 1955.
19. R. R. Everett, C. A. Zraket, and H. D. Benington, "SAGE—A Data-Processing System for Air Defense," *Proceedings of the Eastern Joint Computer Conference*, Washington, DC, 1957, pp. 148-155.
20. M. M. Astrahan, B. Housman, J. F. Jacobs, R. P. Mayer, and W. H. Thomas, "Logical Design of the Digital Computer for the SAGE System," *IBM J. Res. Develop.* **1,** 76-83 (1957).
21. C. L. Christiansen, L. E. Kanter, and G. R. Monroe, "Data Synchronizer," U.S. Patent 3,812,475, May 21, 1974.

Received April 11, 1980; revised September 25, 1980

C. J. Bashe is located at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598. W. Buchholz is with the Data Processing Products Group at the IBM laboratory in Poughkeepsie, New York 12602. G. V. Hawkins is located at 445 Hamilton Avenue, White Plains, New York 10601. J. J. Ingram is located at the IBM System Communications Division laboratory, Research Triangle Park, North Carolina 27709. N. Rochester is located at the IBM Cambridge Scientific Center, 545 Technology Square, Cambridge, Massachusetts 02139.



Computer architecture and its implementation