ELEC3020: Lecture 2-1

Some more recent history

Some Milestones

- 1964: IBM360 A dominant architecture
- 1965: DEC PDP8 First high-volume minicomputer
- 1974: INTEL8080 First general-purpose microprocessor
- 1981: IBM PC
- 1985: MIPS First commercial RISC
- 1990: IBM RS6000 First commercial superscalar

IBM 360

IBM 360

The invention of the silicon integrated circuit by Robert Noyce in 1958 allowed dozens of transistors to be put on a single chip. This packaging made it possible to build computers that were smaller, faster, and cheaper than their transistorized predecessors.

By 1964 IBM was the leading computer company and had a big problem with its two highly successful machines, the 7094 and the 1401: they were as incompatible as two machines could be. One was a high-speed number cruncher using parallel binary arithmetic on 36-bit registers, and the other was a glorified input/output processor using serial decimal arithmetic on variable-length words in memory. Many of its corporate customers had both and did not like the idea of having two separate programming departments with nothing in common.

IBM took a radical step. It introduced a single product line, the System/360, based on integrated circuits, that was designed for both scientific and commercial computing. It was a family of about a half-dozen machines with the same assembly language, and increasing size and power. Software written for one of them could, in principle, run on the other. In practice, software written for a small model would run on a large model without problems, but when moving to a smaller machine, the program might not fit in memory. Still, this was a major improvement over the situation with the 7094 and 1401. The idea of machine families caught on instantly, and within a few years most computer manufacturers had a family of common machines spanning a wide range of price and performance.

Property	Model 30	Model 40	Model 50	Model 65
Relative performance	1	3.5	10	21
Cycle time (nsec)	1000	625	500	250
Maximum memory (KB)	64	256	256	512
Bytes fetched per cycle	1	2	4	16
Maximum number of data channels	3	3	4	6

IBM360 (2)

Another major innovation in the 360 was **multiprogramming,** having several programs in memory at once, so that when one was waiting for input/output to complete, another could compute.

The 360 also was the first machine that could emulate (simulate) other computers. The smaller models could emulate the 1401, and the larger ones could emulate the 7094, so that customers could continue to run their old unmodified binary programs while converting to the 360. Some models ran 1401 programs so much faster than the 1401 itself that many customers never converted their programs.
Emulation was easy on the 360 because all the initial models and most of the later models were microprogrammed. All IBM had to do was write three micro-programs, for the native 360 instruction set, the 1401 instruction set, and the 7094 instruction set. This flexibility was one of the main reasons microprogramming was introduced.

The 360 solved the dilemma of binary-parallel versus serial decimal with a compromise: the machine had 16 32-bit registers for binary arithmetic, but its memory was byte-oriented, like that of the 1401. It also had 1401 style serial instructions for moving variable-sized records around memory.

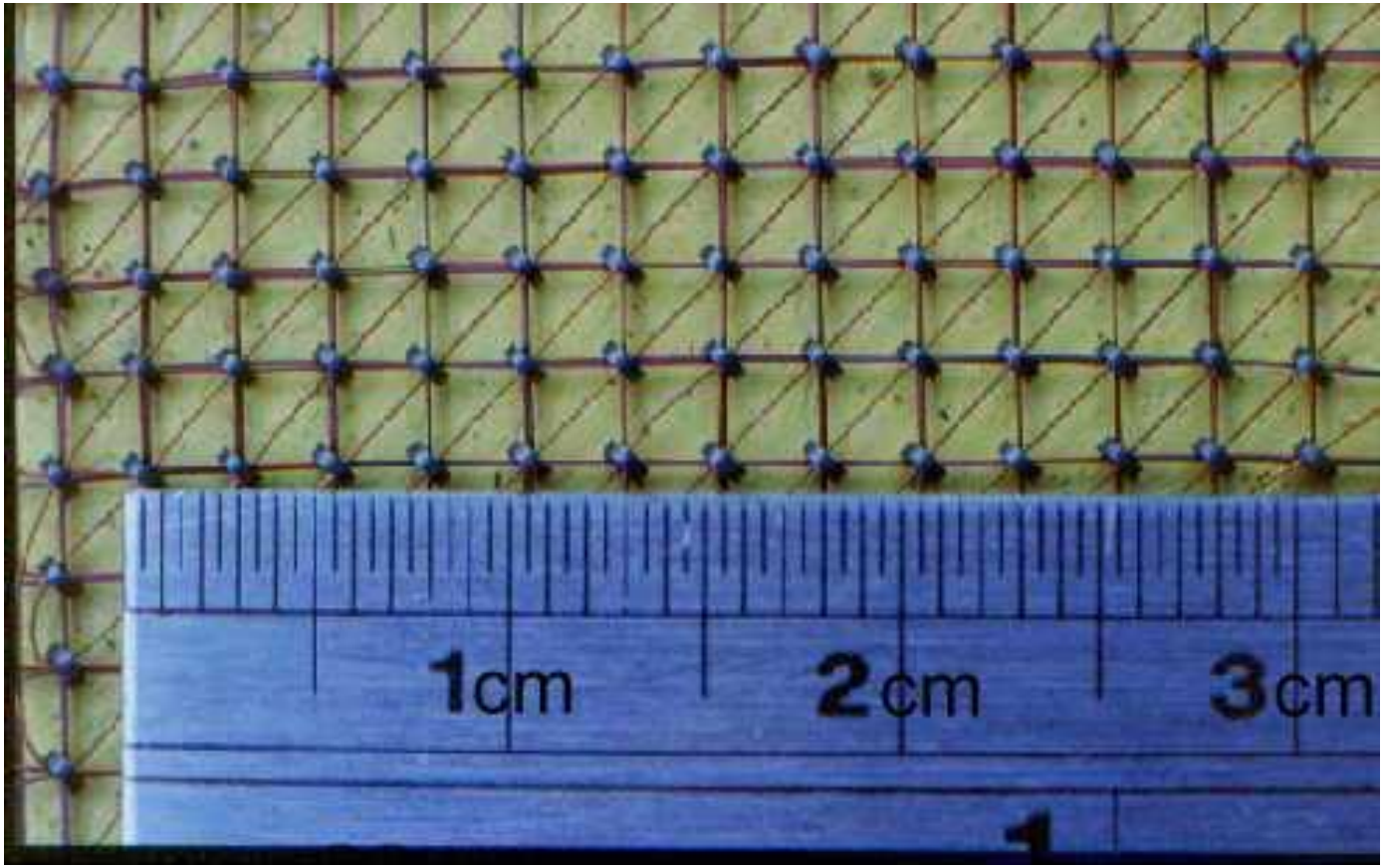
Another major feature of the 360 was a (for that time) huge address space of 2^24 bytes (16 megabytes). With memory costing several dollars per byte in those days, 16 megabytes looked very much like infinity. Unfortunately, the 360 series was later followed by the 370 series, 4300 series, 3080 series, and 3090 series, all using the same architecture. By the mid 1980s, the 16-megabyte limit became a real problem, and IBM had to partially abandon compatibility when it went to 32-bit addresses needed to address the new 2^32 byte memory.

With hindsight, it can be argued that since they had 32-bit words and registers anyway, they probably should have had 32-bit addresses as well, but at the time no one could imagine a machine with 16 megabytes. Faulting IBM for this lack of vision is like faulting a modem personal computer vendor for having only 32-bit addresses. In a few years personal computers may need far more than 4 GB of memory, at which time 32-bit addresses will become intolerably small.

A.S. Tanenbaum, 1999

PDP8

Core Store



PDP8E

Introduced in 1970. Price: $6,500. SSI and MSI TTL logic were used on these boards, and the entire CPU fit on 3 boards. Interconnection between boards was through a new bus, the OMNIBUS. This eliminated the need for a wire-wrapped backplane, since all slots in the bus were wired identically. A new line of peripheral interfaces was produced, most being single cards that could be plugged directly into the inside the main enclosure. These included a set of posibus adapters allowing use of older peripherals on the new machine.

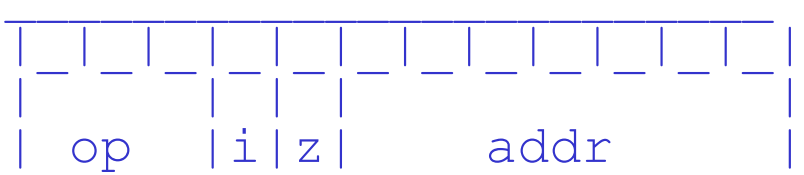
The core memory cycle time was 1.2 or 1.4 microseconds, depending on whether a read-modify-write cycle was involved (a jumper would slow all cycles to 1.4 microseconds). A 4K core plane was packaged on a single quad-wide double-high board, with most of the drive electronics packed onto two adjacent boards. Soon after the machine was introduced, an 8K core plane was released in the same format.

Standard configuration: A CPU with 4K of memory, plus 110 baud current loop teletype interface. The PDP-8/OEM had a turn-key front panel, no core, 256 words of ROM and 256 words of RAM, and was priced at $2800 in lots of 100. The standard OMNIBUS backplane had 20 slots, with no fixed assignments.

PDP8 instructions

The PDP-8 word size is 12 bits, and the basic memory is 4K words. The minimal CPU contained the following registers: PC - the program counter, 12 bits. AC - the accumulator, 12 bits. L - the link, 1 bit, commonly prefixed to AC as <L,AC>.

It is worth noting that many operations such as procedure linkage and indexing, which are usually thought of as involving registers, are done with memory on the PDP-8 family. Instruction words are organized as follows:



op the opcode.
i the indirect bit (0 = direct, 1 = indirect).
z the page bit (0 = page zero, 1 = current page).
addr the word in the page.

PDP8 instructions (2)

The top 5 bits of the 12 bit program counter give the current page, and memory addressing is also complicated by the fact that absolute memory locations 8 through 15 are incremented prior to use when used as indirect addresses. These locations are called auto-index registers (despite the fact that they are in memory); they allow the formulation of very tightly coded array operations.

The basic instructions are:

000	AND	and operand with AC.
001	TAD	add operand to L,AC (a 13 bit value).
010	ISZ	increment operand and skip if result is zero.
011	DCA	deposit AC in memory and clear AC.
100	JMS	jump to subroutine.
101	JMP	jump.
110	IOT	input/output transfer.
111	OPR	microcoded operations.

The ISZ and other skip instructions conditionally skip the next instruction in sequence. The ISZ is commonly used to increment a loop counter and skip if done, and it is also used as an general increment instruction, either followed by a no-op or in contexts where it is known that the result will never be zero. The JMS instruction stores the return address in relative word zero of the subroutine, with execution starting with relative word one. Subroutine return is done with an indirect JMP through the return address. Subroutines commonly increment their return addresses to index through inline parameter lists or to perform conditional skips over instructions following the call.

Intel 8080

The 8080 followed on from the:

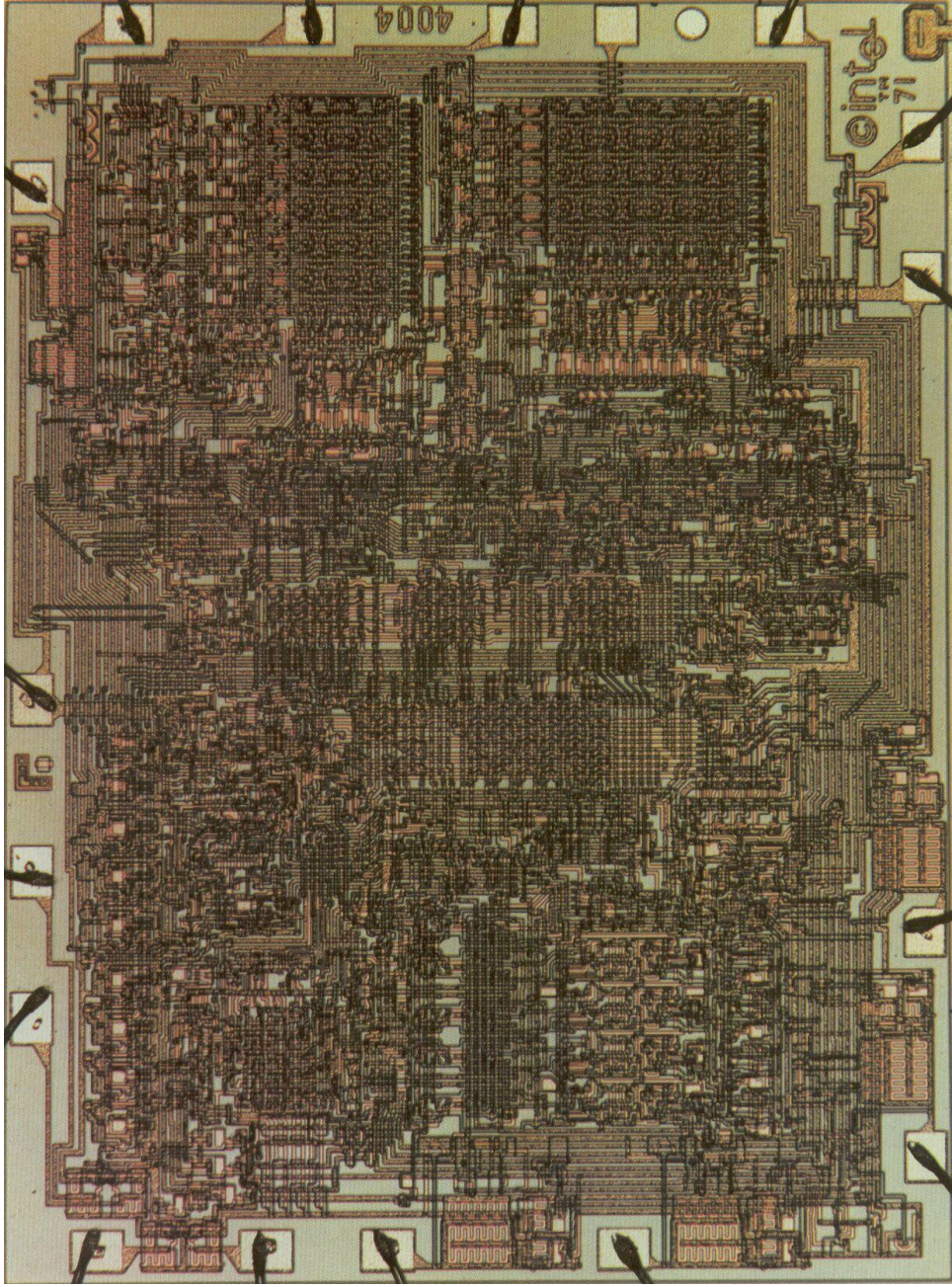
- 4004
  The world's first microprocessor, released in 1971. The 4004 contained 2300 transistors and was intended for use in a calculator. It processed data in 4 bits, but its instructions were 8 bits long. Program and Data memory were separate, it had 1 kilobyte of data memory and a 12-bit PC for 4K of program memory (in the form of a 4 level stack, used for CALL and RET instructions). There were also sixteen 4-bit (or eight 8-bit) general purpose registers. The 4004 had 46 instructions.
- 8008
  Released in 1972, with 3,500 transistors occupying 16mm^2. Intended for use in terminals.

The Intel 4004

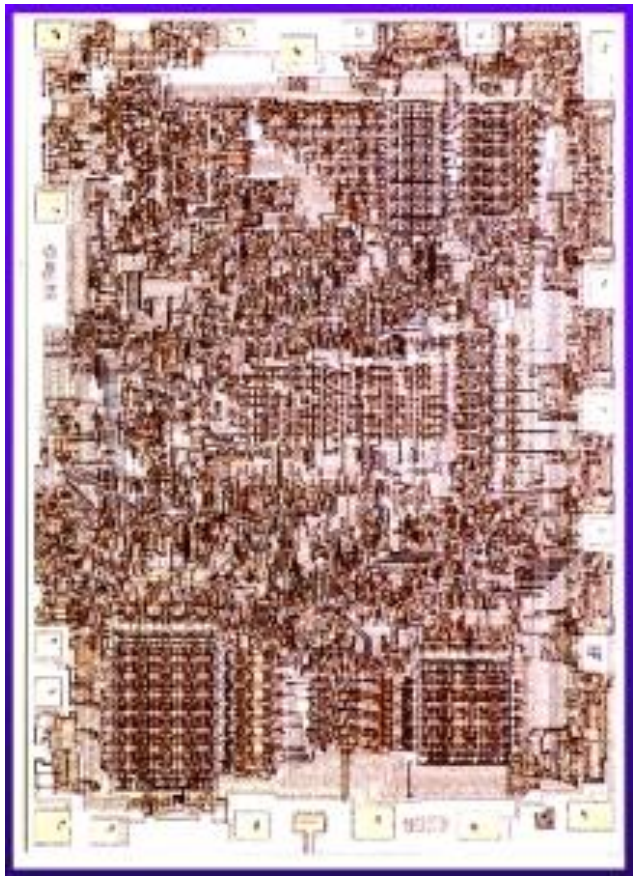
Federico Faggin

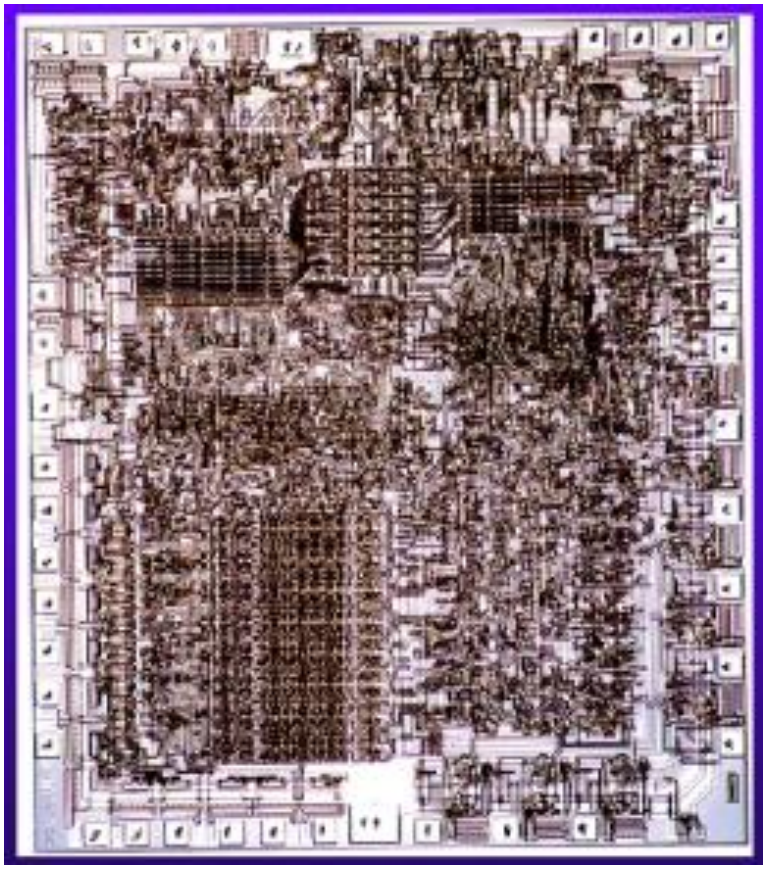
Busicom 141-PF

http://www.cpu-museum.com



8008 and 8080





The 8080

The first popular microprocessor, released in 1974 with 4,800 transistors in 16mm^2

The designer, Frederico Faggin leaves Intel the same year to start Zilog; the Z80 could directly drive DRAM.

The Intel 8080 used a register based architecture with registers AX, BX, CX, DX, and HL, all 16 bit but capable of being used as 8 bit register pairs so that the AX register could be used as two separate registers AH and AL. AH was really just the higher byte of register AX; and AL the lower byte. In this way, the AX, BX, CX, DX, and HL registers could be used as AH, AL, BH, BL, CH, CL, DH, DL, H, and L 8 bit registers.

Another feature of the 8080 was its separate I/O map. This meant that to perform byte-wide input/output to hardware, special instructions were used: IN to input from byte-wide input ports, OUT to output to byte-wide output ports. Access to memory involved access to a different memory map using typically the MOV instruction.