AccessionIndex: TCD-SCSS-T. 20250916.001
Accession Date: 16-Sep-2025
Accession By: Dr.Brian Coghlan and Erturk Kocalar
Object name: Intel 4004 and 4040 microprocessors and associated chips
Vintage: 1971
Synopsis: The first commercially successful microprocessors.

Description:
Over the years there was debate over who invented the first microprocessor. Fig.4 of the History of the Early Ideas of Computers [1] shows the first six microprocessor designs as:

(1) ***Four Phase Systems' AL1:*** designed by Lee Boysel, first working samples April 1969, but only available within the company.
(2) ***Garret AiResearch's MP944:*** designed by Ray Holt and Steve Geller, first working 1970, designed for and used in the F-14 fighter aircraft, but not publicly disclosed until 1998.
(3) ***Fairchild's PPS-25:*** designed by Rich Whicker, first working samples May 1971, available for sale from late-1972.
(4) ***Texas Instruments' TMX1795:*** designed by Gary Boone and Michael Cochran, first working samples June/July 1971, but only taken as far as the prototype stage.
(5) ***Intel's 4004:*** designed by Federico Faggin, Masatoshi Shima, Ted Hoff and Stan Mazor, first working samples January 1971, available for sale from November 1971.
(6) ***Pico / General Instrument's PICO1 / GI250:*** designed by George Stevenson, David Campbell, Harry McLennan and Les Leech, first working samples 1971, but only sold within a calculator from January 1972.

Of these first six, only two became standard commercially available products: the Intel 4004 and the Fairchild PPS-25. It seems Fairchild began design before Intel, but with Herculean effort by Federico Faggin, Intel achieved working samples in January 1971 just before Fairchild in May 1971. Texas Instruments and Intel registered patents, and Intel invested in subsequent developments, while Fairchild neither patented nor invested. Thereafter, disputes relating to the Intel and Texas Instruments patents eventually led to a court case that resolved that these patents be invalidated and Lee Boysel be recognised as the inventor, a decision that is now widely accepted [2]. Despite this, it is undeniable that Intel's 4004 [3][4][5] was the first commercially successful microprocessor, the heart of a 4-bit multi-chip pMOS family.

The design of the 4004 started as follows [6]:

> Starting in 1968 a young engineer at Busicom, Masatoshi Shima, worked on the design of Busicom's first calculator with printed output, the Busicom 141-PF. Due to the pressure for rapid development, Shima's supervisor, Tadashi Tanba, using his experience while working at the advanced computer manufacturer Control Data Corporation, had started a design based on a programmed approach using computer software technology with desktop calculator hardware. This would allow easy changes to be applied to alter the specification of the calculator or control other products in their business product line.
>
> Shima, who had some programming experience, took this "computer system" approach further and produced a design for the 141-PF incorporating arithmetic units (adders), multiplier units, registers, read-only memory, and he even defined the macro-instruction set to control this decimal computer system.

Tadashi Sasaki was an executive for Sharp Corporation, a major manufacturer of electronic calculators, and had had a good working relationship with Bob Noyce of Fairchild Semiconductor Corporation, the U.S. integrated circuit manufacturer. In 1968 Noyce left Fairchild to start up Intel Corporation with Gordon Moore, and shortly afterwards visited Sasaki at Sharp in Japan to try to sell Intel's integrated circuits. Unfortunately, Sharp was contracted to Rockwell for the exclusive supply of calculator integrated circuits so no business with Intel was possible.

However, Sasaki had graduated from the same university department as Yoshio Kojima, the president of the up and coming Busicom Corporation, and so felt empathy for Kojima and his company, which was struggling financially. This resulted in Sasaki, as an executive of the large Sharp Corporation, offering some technical assistance to the much smaller Busicom, a business situation that was allowed in Japan. However, working behind the scenes, Sasaki went a step further and offered finance to Busicom on the condition that it contracted Intel Corporation to develop the integrated circuits for Masatoshi Shima's "computer system" design for the 141-PF calculator.

In 1969 Masatoshi Shima was one of three Busicom employees who traveled from Japan to Intel in the U.S.A. with details of the proposal for the integrated circuits for the 141-PF and other advanced calculators, which would differ just by the contents of the instructions in the ROM (Read-Only Memory) chips.

Intel put Marcian E. *"Ted"* Hoff in charge of the project with the assistance of Stanley Mazor, another engineer.

There was little progress in the first year of the project. Intel knew little about calculator electronics and how to use the proposed line printer, and was not very enthusiastic about developing the proposed integrated circuits. Also it considered that for its small development staff there were too many chips required, the designs were too complicated and required non-standard packaging with large numbers of pins, and they had enough to do working on the RAM (Random-Access Memory) chips that were Intel's principal product and were starting to sell well.

Hoff had been working with a DEC PDP-8 computer and appreciated its RISC (Reduced Instruction Set Computer) architecture which simplified the electronics at the expense of a bigger memory for the larger program required. And, happily, Intel was a memory chip manufacturer.

Hoff proposed a much simplified system using a few standardised chips employing a limited instruction set so that different combinations of the simple instructions could be used for different actions such as reading the keyboard and driving the printer. Use of a small chipset to produce a computing system was actually something that had been foreseen by many in the computing and semiconductor industries for some time, but Intel had the manufacturing capability and here was a specific proposal.

Intel was in two minds. On the positive side there would be much less chip development required for Hoff's proposal than Busicom's proposal, but it still did not really have the development staff available, and would it make any money after the development costs for a small production run of specialised chips for one company?

For some months Shima and Hoff continued to work on their different designs at Intel. However, in October 1969, Busicom executives visited Intel and they were given presentations on the two proposals. The Busicom executives decided to go with Hoff's proposal, perhaps somewhat surprisingly since they would have been expected to favour their in-house proposal.

Since Hoff's design was to go ahead Shima dropped the work he had been doing and started to work on Hoff's design, and then went back to Japan at the end of 1969 to finish the programming and produce the documentation. When Shima returned to Intel in April 1970 to check on how the development was progressing he was aghast to find that nothing further had been done. Hoff had been moved from this job to the development of another central processing unit for an intelligent terminal for Computer Terminal Corporation. This cpu would become the very successful Intel 8008.

This period of architectural and logic design posed many interesting issues that required inventive solutions. These events are perhaps best portrayed in a reflective IEEE paper by Stanley Mazor [7] {lightly reformatted}:

With only two designers, Intel didn’t have the manpower to do that many custom chips. We needed to solve their problem with fewer chip designs. Ted Hoff chose a programmed computer solution using only one complex logic chip (CPU) and two memory chips; memory chips are repetitive and easier to design. Intel was a memory chip company, so we found a way to solve our problem using memory chips! … We used the 16-pin package, because it was the only one available in our company. … However, the multiplexing logic increased chip area of the specialized ROM/RAM memory chips, which then had to have built-in address registers. Increasing the transistor count to save chip connections was a novel idea. … The use of three-transistor dynamic memory cells made the RAM chip feasible. A built-in refresh counter was used to maintain data integrity. Refresh took place during instruction fetch cycles, when the RAM data was not being accessed. Dynamic RAM memory cells were also used inside the CPU for the 64-bit index register array and 48-bit Program counter/stack array. Intel expertise in dynamic memory was an enabling factor for the MCS4!

…

We wrote many subroutines which operated on 16-digit numbers stored in RAM. As an example, a 10-byte loop for digit serial addition took about 80µs/digit (similar speed as IBM 1620 computer sold in 1960 for $100,000). … One major difference compared to most computers, was the MCS-4's separate program and data memories. Conventional computers ran programs from RAM (core) memory. However, our application firmware needed to be permanently stored in ROM. A major change was needed for subroutine linkage. Normally, as part of a minicomputer subroutine call instruction execution (PDP-8, HP 2114) the calling program's return address would be saved at the top of the subroutine in RAM. Since MCS-4 routines were in ROM (can't write into it) we could not use this method. Instead, we used a push down stack inside the CPU for saving up to three return addresses. This was not a new idea. Stacks had been used in Burrough's computers and the IBM 1620, which Ted Hoff and I had programmed - we used our experience with large scale computers.

…

The time division multiplexing of the 4-bit bus, the on-chip dynamic RAM memories, and the CPU's address stack are the highlights of the MCS-4 architecture. However, there is another interesting feature - distributed decoding of instructions. The ROM/RAM chips watched the bus, and locally decoded port instructions, as they were sent from the ROM. This eliminated the need for the CPU to have separate signal lines to the I/O ports, and also saved CPU logic. This is not a feature used in conventional computers.

…

Ted Hoff and I made the original proposal for the MCS-4 and did the feasibility study for the first calculator. Federico Faggin did all of the logic and circuit design and implemented the layout; Busicom's M. Shima wrote most of Busicom's firmware. (Later Shima joined Intel as the 8080 designer.) The Intel patent on the MCS-4 (Hoff, Faggin, Mazor) has 17 claims, but the single chip processor is not claimed as an invention.

…

One difficulty implementing any system on a set of LSI chips is partitioning into pieces with a reasonable number of I/O pins on each. It was very expensive to get more than 20 pins. … Optimization consists of maximizing the number of gates inside compared to the number of pins outside - the gate to pin ratio. Memory chips with 1kb in an 18-pin package gave an excellent gate/pin ratio of about 100:1. … If a CPU were to be built of LSI chips it was not obvious how to break it into pieces with a small number of I/O pin connections and a high gate/pin ratio. Simply put, if you cut an ordinary CPU into two pieces you would have hundreds of signals which would need to cross the chip boundaries. Each package pin also required a lot of MOS chip "real estate" for amplifiers to drive the heavy off chip capacitive loads and for the wire bonding pads which go from the chip to the package. Besides the cost, placing more pins on an LSI chip also lowered the reliability. Hence, most commercial LSI applications were constrained by the few leads available on IC packages. This is why the early microprocessors were in 16 and 18 pin packages.

…

A CPU chip contains "random logic" requiring many interconnection wires. Prior to 1980 most semiconductor chips had only one layer of metal. This metal was used for global connections such as power, ground, clocks, and major busses. Local connections were made using poorer quality, higher resistance, lines of poly-silicon or diffusion. The silicon gate process, developed originally at Fairchild Semiconductor in around 1967 {by Faggin et al}, provided slightly better local interconnections and crossovers. This technology also offered lower capacitance, smaller size (self-aligned structures) and lower voltage operation. This was a key technology enabler for microprocessor development at Intel. … To reduce the overall power dissipation most of the circuits were operated dynamically in a two phase operation. First a circuit was precharged using an on-chip amplifier, and then the circuit was conditionally discharged, based upon logic decisions. Previously, "bootstrap" amplifiers were built using the gate "overlap capacitance" as part of the circuit. However, silicon gate self-aligned geometry eliminated this capacitor. F. Faggin innovated a new and efficient bootstrap amplifier as part of his early circuit design of Intel's chips.

Faggin became well known as the inventor of the important silicon gate technology (SGT). From Faggin's memoir [8] {lightly reformatted}:

By 1967 much of the MOS industry was engaged in developing a low threshold voltage MOS technology to replace the incumbent technology. This objective was eventually achieved with the SGT {Silicon Gate Technology} in 1968.

…

In 1966 Robert Bower {had} realized that if the {MOS transistor} gate electrode was defined first, the {transistors} would be "self-aligned" … In 1967 John C. Sarace {et al created} working self-aligned gate MOS transistors. … In late 1967, Tom Klein of Fairchild Semiconductor {established it was} possible not only to create self-aligned gate transistors, but also to achieve a low threshold voltage process … However, Klein could not figure out how to architect the process.

…

In February 1968, I joined the MOS process development group of the Fairchild Semiconductor … choice between two projects: (1) designing a special shift register chip or (2) developing a self-aligned-gate MOS technology using silicon gates. I chose the latter … I was told, of course, about the experiments of Tom Klein. … my first tasks were: (1) to invent the process architecture for self-aligned silicon gates, (2) to develop a method to precision-etch the amorphous silicon, and (3) to design the detailed processing steps to fabricate MOS ICs with silicon gates … also had to design a suitable test pattern … If all worked well, I would then have to design an appropriate commercial integrated circuit … After a week or so it {dawned} on me that if I started with etching a tub into the initial oxide where the entire MOS device was supposed to be, I could then solve the problem. … The "tub" was the missing insight about how to make self-aligned gate MOS ICs. … The next problem was how to connect the silicon gates … I came up with the idea of making a buried contact … required an additional masking step, but it would considerably increase {by ~40%} the circuit density because metal could now run over the buried contact.

…

When ten days after my arrival … approved my proposed process architecture, but said that the buried contact would never work and … didn't even want me to try it out. … I decided to place a couple of structures {in the test pattern} that would allow me to verify if the buried contact would actually work. … The next step was to develop a suitable silicon etching solution … I decided to experiment with {nitric and hydrofluoric acids}. By trial and error, I found the best ratios … It took about ten days and a new pair of shoes to reach the goal; the shoes being the victims of a drop of the mixture falling on my right foot. … By April, 1968 I was able to fabricate the first working MOS transistors with silicon gate … {my supervisor asked} what was their most difficult IC to produce. They said it was the Fairchild 3705 … The idea was to design a {new chip called 3708} that was functionally identical to the 3705, but used the SGT … by early July, I had the first wafers of the 3708 … it worked immediately. {But during characterization} I found to my dismay that the amorphous silicon tended to break {at the tub} … Fortunately, there was another possibility … the chemical decomposition of silane (SiH4) … and the problem was elegantly solved … the 3708 was 5 times faster, it had about 100 times less leakage current, and the on resistance of {transistors} was 3 times lower.

SGT was presented at IEDM in October 1968 [9]. For the 3708 see [10]. Patents were granted for aspects of SGT, but not to Faggin, in what seems a failure of duty of care by both Fairchild and its leaders who were soon to leave Fairchild to become the founders of Intel, who then exploited SGT. In March 1970, knowing the origins of SGT and the status of the Busicom project, Intel poached Faggin from Fairchild to design the chips by year-end. Again from Faggin's memoir [8] {lightly reformatted}:

In April 1970 … I joined Intel. My first day of work, I met Stan Mazor, an engineer working for Ted Hoff, the manager of the Application Research department, who described to me the "Busicom Project." … When I saw the project schedule that was promised to Busicom, my jaws dropped: I had less than six months left to design four chips, one of which, the CPU, was at the boundary of what was humanly possible because a chip of that complexity had never been done before. I had nobody working for me to share the workload; Intel had never done random logic custom chips before and, contrary to companies in that business, had no methodology and no design tools for their speedy and error-free design. … What did I get myself into? I thought. Fortunately I was young and eager to prove myself in my new chosen field of endeavor. I understood computers, I could design both logic and circuits and I had experience in both MOS IC design and in MOS process development. Most importantly, I knew intimately the capabilities of the MOS SGT, a process only a few engineers knew about. This was a very rare combination indeed, even in those days. Therefore I felt that if I couldn't do it, nobody could.

Within a few days of joining Intel, Stan Mazor and I met Shima at the San Francisco airport arriving from Tokyo. Shima was eager to check the progress of the Busicom project since his last visit in the Fall of 1969. In particular, he wanted to check the logic design of the CPU and make sure that it would perform according to the agreed upon specification. We drove directly to the company and when Shima asked me about the progress, I innocently gave him the material I was given by Stan a couple of days earlier. Shima impatiently said that he had already seen that material months before, and became furious when he found out that that's all I had since no additional work had been done during the previous 5 months. He became very angry at me, the project leader, literally calling me names. I could not convince him that, having joined Intel only a few days before, I could not have done the work he expected to find.

He repeatedly said, "I came here to check, and there is nothing to check! This is only idea!" He said that his project was irreparably compromised and that he had to call his management to find out what to do. It took almost one week for Shima to calm down and accept what happened. During that time I resolved the remaining architectural issues; I started working on the missing design methodology; and prepared a new schedule that would give Busicom first silicon of all four chips by the end of December, assuming I could get one engineer and a couple of draftsmen on time to help me. This new schedule was extremely aggressive and would require me to work 70-80 hours per week to make up for the previous unrealistic schedule, and to recover part of the incurred delay. I also told Shima that if he'd help me there would be a chance of meeting the new schedule, since it would take time to hire the people I needed. Finally, the difficulties were resolved; Busicom accepted the new schedule; Shima got permission to stay for six months to help me; and I could concentrate on designing what by now I had named the 4000 family.

> I should also mention here that my initial impression of the Busicom project was mixed. I liked the idea of making a CPU on a chip, something that had been in the air for some time. In fact, Lee Boysel, the head of the Fairchild MOS design group, had been advocating this idea since 1968, saying that with MOS technology it would become possible to make a CPU in a few chips. He left Fairchild in 1969 to start Four-Phase Systems, a company that successfully developed and sold small computers with CPUs made with a few MOS chips.
>
> I liked the idea of a family of chips that seamlessly worked together, and I was excited at the prospect of designing a CPU on a chip, but I had some misgivings as well. For example, I found the use of 16-pin packages, particularly for the CPU, incomprehensible, since a lot of performance would be lost by the need to multiplex address and data into a single 4-bit bus. But in those days, using only 16-pin packages was a religion at Intel, despite the fact that 40–pin packages had been standard in the industry for many years. I found the architecture of the RAM, and the way it was addressed by the CPU quite strange, to say the least. RAM was addressed as if it was an I/O operation, requiring a complicated and long setup. I couldn't understand why it had to be so difficult to address RAM just like any other CPU did. There had to be a better way to accomplish this task, but the last thing anybody would have wanted, given the enormous delay of the project, was to make any changes to the architecture that had been blessed by the customer. I had enough to worry about, and I concentrated on checking that the architecture was sound. I found a couple of errors that fortunately could be easily fixed, and I set my heart in peace for the long haul required to make the 4000 family a reality.

From Faggin's article [11] he recalls his subsequent Herculean 12-month adventure to create 4 chips, which for each chip he states: *"{normally} starting from chip specifications to first silicon, would take at least 6 to 9 months. From first silicon to transfer-to-production … would normally take from 3 to 8 months"*:

> Since Intel had only designed memory chips up to that point, they had no expertise in random logic design as it existed in companies like Fairchild, Texas Instruments, AMI and others. Those companies were in the business of designing custom random logic chips – the major application of MOS technology in those days. These companies had extensive libraries of circuits and circuit blocks, with layouts known to work and characterized. They had computer simulation tools for logic and circuit design, and for test program generation. They also had characterization tools; random logic testers; and, most importantly, random logic designers expert in the entire process. Furthermore, the SGT was new and had never been used before for random logic circuits. It required quite a different layout style than metal gate, particularly when the buried contact was used. … I had to also do the logic design for all four chips in the family. And above all, I had to figure out and create the random logic design methodology for silicon gate technology that didn't exist. I even had to design and build a debugging and characterization tester that wasn't available at Intel; for Intel had characterization equipment for only memory chips.
>
> Since I had promised the customer – under duress I might add – to deliver samples of all four chips by December, 1970, less than 9 months from start; and since the CPU alone would take almost 8 months, I had to work practically on all four chips simultaneously, staggering them so that the critical layout resources would be kept continually busy. I decided to design the 4001 {ROM} first, followed by the 4003 {output shift register}, 4002 {RAM}, and 4004 – the CPU. This sequence allowed me to incrementally develop the methodology and all the necessary building blocks I needed to use for the most complex chip, the 4004.
>
> …
>
> To avoid having to do circuit simulation, except for exceptional situations, I had prepared a set of normalized MOS characteristics based on measuring worse-case transistors fabricated at Intel. Using those graphs I could rapidly calculate the transistor sizes necessary to achieve the required speed for the expected capacitive load. This was a graphic calculation method similar to the one I had learned at the technical institute to size up vacuum tube circuits.
>
> One of the early challenges encountered during the 4001 design was to invent a flipflop that was guaranteed to come up in a known state after turning the power supply on. This was necessary because there were no extra pins in some of the 400x chips to dedicate to a reset signal (each chip was packaged in a 16-pin DIP!). This flip-flop was to be used in the critical control of the tristate external data bus that connected all the chips, to avoid bus-contention immediately after the power supply was turned on.
>
> …
>
> Sometime in October, 1970 … I received the first silicon of the 4001. This was my first LSI chip design and I was very nervous because it was the first real test of my methodology: If the 4001 didn't work, all the other chips would have the same problems because the same worst-case design rules I had developed were used in all of them. The characterization tester had been partially completed, enough to verify the 4001 operation and I was delighted when the oscilloscope displayed the familiar waveforms I had drawn so many times on paper and now were painted live on the display! I was stunned by the fact that the chip was doing exactly what it was supposed to do, after so much work and so many error-prone steps. The miracle of technology!
>
> …

> A few weeks after receiving the first 4001 wafers I also got the first silicon wafers of the 4003. That chip also worked the first time, adding to my confidence level. In late November I received the first silicon of the 4002 which also was fully functional, but for one minor mistake that was quickly identified and fixed.
>
> Finally came the big day when I was given the first wafers of the 4004. The climactic moment of truth had arrived. It was the end of the work day, few days before New Year eve, and most people had already left the lab. If the 4004 did work, I would have met the schedule I committed to Busicom nearly nine months before. Luckily nobody was around to see how nervous I was. My trembling hands placed the first wafer on the wafer prober. I lowered the probes onto the first chip expecting to see the now familiar activity in the data bus, but instead nothing happened. "Oh, well," I said to myself, "that one must be a bad chip." I lowered the probe onto another chip with the same outcome, and then probed several more chips, always with the same symptoms. "Maybe this is a bad wafer," I thought. I tested another wafer, and got exactly the same results.
>
> By this time I was profusely sweating thinking, "Nothing works! How could I have screwed up so badly?" I decided to look at the chips under the microscope, and sure enough, the problem was obvious: during the manufacturing process the buried contact mask was left out by a technician's mistake, therefore most of the transistor gates were not connected anywhere, hence no life. Now my chance to meet the schedule had been irreparably blown away by a trivial mistake in manufacturing that was going to cost me about 3 weeks of delay. What a disappointment!
>
> About three weeks later I received a new run of 4004 chips. This time nothing was left out, and I made sure of that by checking the wafers under the microscope before loading the first one on the probe station. Like the previous time, I received the wafers at the end of the work day with the lab nearly deserted, and I set out to spend most of the night probing the 4004. I breathed much easier after the familiar signals in the data bus appeared in the oscilloscope. Now I was in business! I probed until 3-4 am, finding that everything was working as expected until, exhausted, I left for home. … In the following couple of weeks I continued to check the 4004 and found a few minor problems that were relatively easy to diagnose and fix.
>
> …
>
> And by mid-March, when the revised silicon of the 4004 was received, we also had the completed 4001's. Busicom could then test the entire calculator using an engineering prototype that had sockets in the pre-production printed circuit board ready to receive the components. When the calculator was turned on, it worked perfectly with all the final 4000 family components! The production of chips and calculators could now start. Finally I could take a giant sight of relief; and Intel could start selling components to Busicom.

To Faggin must go much of the plaudits, for using the silicon gate technology (SGT) he'd invented at Fairchild, and for inventing how to use buried contacts and bootstrap loads with SGT, halving chip size and increasing yields (although the 4004 was the only one of the four chips to use buried contacts). Astonishingly in view of the challenges evident above, delivery to Busicom began in March 1971.

Busicom then found its 4004-based 141-PF calculator progressively less competitive against designs using medium-scale (e.g. TTL) chips, which were falling in price. This was overcome by negotiating a reduced price per chipset in exchange for allowing Intel to sell the chipset to the general market. Thus on the 15th November, 1971, Intel published the now-famous advertisement in the Electronic News, see Fig.1, for many the accepted beginning of the microprocessor age [8][12].

The chipset, marketed as *MCS-4* (see [13] and Figs.2-4) consisted of:

4001	256 x 8 ROM plus 4-bit I/O port
4002	80 x 4 RAM plus 4-bit output port
4003	10-bit parallel output shift register
4004	4-bit microprocessor
4008	8-bit memory address latch plus I/O port
4009	memory and I/O converter

The MCS-4 design details are well known and will not be repeated here. Suffice to say that the 4-bit 4004 contained just 2300 transistors and executed instructions in 10.7µS when fed with a 750kHz two-phase clock.

Some aspects bear a resemblance to the Fairchild PPS-25 (see elsewhere in this catalog) that may reflect the prevailing state of technology, e.g. the serialization, the size of nibbles, the Harvard architecture, and the use of dynamic memory forms. However whereas the 4004 used DRAM, an Intel speciality, the PPS-25 used equally efficient shift registers for memory, and whereas the 4004 had a single unified bus and traditional computer architecture, the PPS-25 had three separate busses yielding the heavily overlapped "racetrack" architecture that Cochran so admired [14]. Both used Federico Faggin's invention at Fairchild of silicon gate technology, but his later inventions of buried contacts and bootstrap loads for SGT led to a greatly superior application of that technology in the 4004.

To improve on the 4004, and in order to overcome Intel's preference for small/cheap dual-inline packages (DIP), which had constrained the 4004 to 16-pins, Faggin proposed an expansion of the 4004 to a modest 24-pins, formulated the architecture and led the project, supervising the detailed design by Tom Innes of what became the 4040 [15], see Figs.5-6. The extra pins enabled a doubling of the ROM space, interrupts (absent on the 4004), and halt/stop/single-step. The register set was roughly doubled and, the instruction set expanded from 46 to 60. Even so, it retained backwards compatibility with the 4004 support chips. The 4040 was introduced in 1974, with an extended chipset, and then marketed as *MCS-40* [16], comprising:

4040	4-bit microprocessor
4001	256 x 8 ROM plus 4-bit I/O port
4002	80 x 4 RAM plus 4-bit output port
4003	10-bit parallel output shift register
4008	8-bit memory address latch plus I/O port
4009	memory and I/O converter
3216	4-bit parallel bus driver
3226	4-bit parallel bus driver
4101	256 × 4-bit Static RAM
4201	clock generator
4207	general purpose 8-bit output port
4209	general purpose 8-bit input port
4211	general purpose 8-bit I/O port
4265	programmable general-purpose I/O
4269	keyboard/display interface
4289	memory interface (merged 4008 + 4009)
4308	1K × 8-bit ROM
4316	2K × 8-bit ROM
4702	256 × 8-bit EPROM

Faggin's contribution was sadly downplayed by Intel after he left to set up Zilog to produce the Z80 as a worthy competitor to the Intel 8080, ironic since Intel's founders left Shottky Semiconductor Labs to set up Fairchild, and then left Fairchild to set up Intel. Moreover, Faggin's worth to Intel was clear from their 1973 review [17], and later Gordon Moore said: "*we essentially had a monopoly on silicon gate for seven* *years, and that was extremely important"* [18]{at 75mins:05secs}. Fortunately, later those differences were resolved.

Trivia1: The 4004's 2,300 transistors on its 0.01 sq.cm die (1.8 sq.cm packaged) had as much power as ENIAC's 18,000 vacuum tubes over its 2.8 million sq.cm.

Trivia2: There is an unverified claim the Intel 4004 was the first microprocessor to fly on a spacecraft (the Pioneer Venus probe launched on 8th August, 1978) [19].

There were and are many demonstration/development boards or systems for the 4004 and 4040, and especially see [20]. In this Collection such boards are represented:

(1) for the 4004 by a replica of the *Linux/4004* board designed by Dmitry Grinberg that hosted the epic first successful boot of Linux on the 4004, which took nearly one week.

(2) for the 4040 by a *Retroshield* to be designed by Erturk Kocalar, which is to use an Arduino host to emulate the original 4040 external environment, such that it can execute original 4040 software.

See elsewhere in this catalog for descriptions of these items.

Very many thanks to Brian Coghlan and Erturk Kocalar for donating these chips, and to Dmitry Grinberg and Erturk Kocalar for providing the demonstration boards.

The homepage for this catalog is at: https://www.scss.tcd.ie/SCSSTreasuresCatalog/
Click *'Accession Index'* (1st column listed) for related folder, or *'About'* for further guidance.
Some of the items below may be more properly part of other categories of this catalog,
but are listed here for convenience.

Accession Index	Object with Identification
TCD-SCSS-T.20250916.001	Intel 4004 and 4040 microprocessors and associated chips. The first commercially successful microprocessors. 1971.
TCD-SCSS-T.20250916.001.01	1 x Intel 4004 microprocessor. [Erturk Kocalar]
TCD-SCSS-T.20250916.001.02	2 x Intel 4003 I/O interface.
TCD-SCSS-T.20250916.001.03	2 x Intel D4265 programmable general-purpose I/O.
TCD-SCSS-T.20250916.001.04	1 x Intel P4269 keyboard/display interface.
TCD-SCSS-T.20250916.001.05	1 x Intel P4289 memory interface.
TCD-SCSS-T.20250916.002	Linux/4004 board. A replica of the board that hosted the epic first successful boot of Linux on the 4004, the first commercially successful microprocessor, 2025.
TCD-SCSS-T.20251216.004	Arduino shield for Intel 4040. A board that enables execution of software by the 4040, 2025.
TCD-SCSS-X.20250916.001	Dr.Brian Coghlan's Collection of Early Microprocessors. An extensive and nearly complete set of unused 1970s microprocessor chips, most accompanied with documentation, some with demonstration boards. 1971.

References:

1. Trinity College Dublin, *History of the early ideas of computers*, see:
https://www.scss.tcd.ie/SCSSTreasuresCatalog/miscellany/TCD-SCSS-X.20121208.009/TCD-SCSS-X.20121208.009.pdf
Last browsed to on 5-Oct-2025.

2. Wikipedia, *Intel/4004*, see:
https://en.wikipedia.org/wiki/Intel_4004
Last browsed to on 16-Sep-2025.

3. David Laws, *Who invented the computer?*, Computer History Museum, 20th September, 2018, see:
https://www.scss.tcd.ie/SCSSTreasuresCatalog/hardware/TCD-SCSS-T.20250916.001/Who-invented-the-microprocessor-DavidLaws-CHM-20250916.pdf
Last browsed to on 16-Sep-2025.

4. Computer History Museum, 1971: *Microprocessor integrates CPU function onto a single chip*, see:
https://www.computerhistory.org/siliconengine/microprocessor-integrates-cpu-function-onto-a-single-chip/
Last browsed to on 16-Sep-2025.

5. IEEE, *Milestone Proposal: Intel 4004 Microprocessor*, 2020, see:
https://ieeemilestones.ethw.org/Milestone-Proposal:Intel_4004_Microprocessor

Last browsed to on 16-Sep-2025.

6. Vintagecalculators, *The Calculator that spawned the Microprocessor: The Busicom 141-PF calculator and the Intel 4004 microprocessor*, see: http://www.vintagecalculators.com/html/busicom_141-pf_and_intel_4004.html
Last browsed to on 16-Sep-2025.

7. IEEE, *The History of the Microcomputer - Invention and Evolution*, Vol.83, No.12, Proc.IEEE, December 1995, see: https://pages.cs.wisc.edu/~markhill/restricted/MKreadings2000percaitlin/ieee_Proceedings_1995_mazor.pdf
Last browsed to on 16-Sep-2025.

8. Federico Faggin, *Silicon, From the Invention of the Microprocessor to the New Science of Consciousness*, pp.304, ISBN: 978-1949003413, Waterside Productions, USA, 8th February, 2021.

9. Faggin, F., Klein, T., and Vadasz, L., *Insulated gate field effect transistor integrated circuits with silicon gates*, p.236, Vol.16, No.2, IEEE Transactions on Electron Devices, February 1969.

10. Federico Faggin and Thomas Klein, *The New Wave: Silicon Gate ICs bring on a faster generation of MOS with low thresholds*, Electronics, 29th September, 1969. Also see: https://www.scss.tcd.ie/SCSSTreasuresCatalog/hardware/TCD-SCSS-T.20250916.001/NewWave-SiliconGate-FagginKlein-Electronics-29Sep1969.pdf

11. Federico Faggin, *The MOS Silicon Gate Technology and the First Microprocessors*, see: http://intel4004.com/The_MOS_Silicon_Gate_Technology_and_the_First_Microprocessors.pdf
This is a preprint of: *La Rivista del Nuovo Cimento, Società Italiana di Fisica*, pp.575-62, Vol.38, No.12, 14th December, 2015, see: https://link.springer.com/article/10.1393/ncr/i2015-10119-7
Last browsed to on 16-Sep-2025.

12. RobWalker, *Interview of Federico Faggin*, 9th June, 1998, see: https://www.scss.tcd.ie/SCSSTreasuresCatalog/hardware/TCD-SCSS-T.20250916.001/Interview-of-FedericoFaggin-byRobWalker-9Jun1998.pdf
Last browsed to on 16-Sep-2025.

13. Intel, *Intel MCS-4 User's Manual*, see: https://www.scss.tcd.ie/SCSSTreasuresCatalog/hardware/TCD-SCSS-T.20250916.001/Intel-MCS4-UsersManual-OCR.pdf
Last browsed to on 16-Sep-2025.

14. Lisa A. Cochran, *A Mind, A Man, A Mystery: The Complex Life of David Shelton Cochran*, Masters Thesis, University of Idaho, December, 2022, see: https://objects.lib.uidaho.edu/etd/pdf/Cochran_idaho_0089N_12523.pdf

Last browsed to on 5-Oct-2025.

15. Wikipedia, *Intel 4040*, see: https://en.wikipedia.org/wiki/Intel_4040
Last browsed to on 16-Sep-2025.

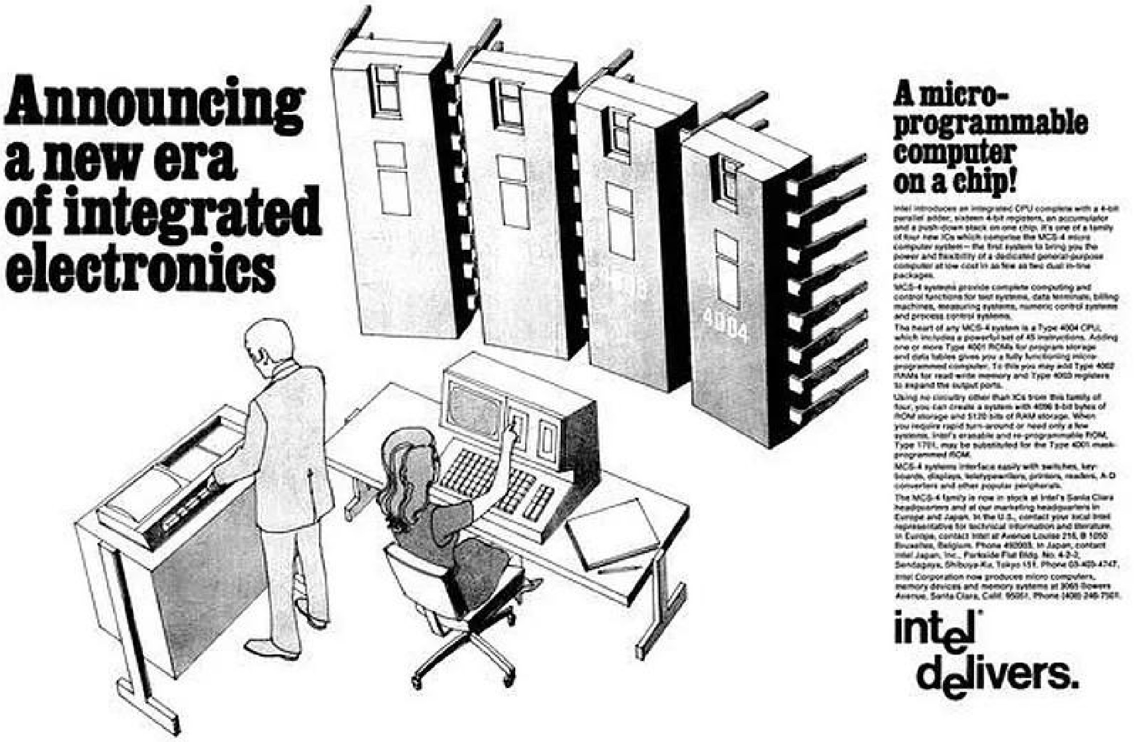
16. Intel, *Intel MCS-40 User's Manual*, see:
https://www.scss.tcd.ie/SCSSTreasuresCatalog/hardware/TCD-SCSS-T.20250916.001/MCS-40-Users-Manual-Nov74.pdf
Last browsed to on 16-Sep-2025.

17. Intel, *Exempt Performance and Salary Review: Federico Faggin*, 12th March, 1973, see: http://www.intel4004.com/images/8080_review.jpg
Last browsed to on 16-Sep-2025.

18. Computer History Museum, *Fairchild 50th Anniversary Celebration*, 2007, see:
https://www.youtube.com/watch?v=arIXj8xpf6Q
Last browsed to on 5-Oct-2025.

19. Herb Johnson, *Pioneer Venus and the Intel 4004*, 19th July, 2019, see:
https://ieeemilestones.ethw.org/Milestone-Proposal:Intel_4004_Microprocessor
Last browsed to on 16-Sep-2025.

20. 4004.com, *Intel 4004 — 50th Anniversary Project*, see:
https://4004.com/
Last browsed to on 5-Oct-2025.

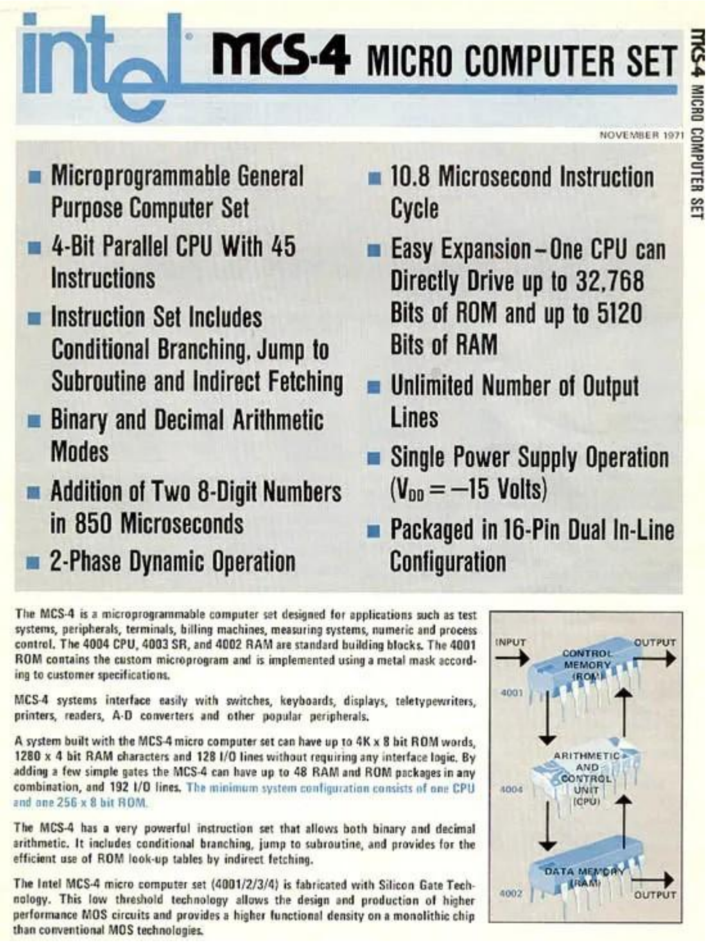


Announcing a new era of integrated electronics

A micro-programmable computer on a chip!

intel delivers.

Figure 1: Intel 4004 advertisement, Electronic News 15-Nov-1971



intel MCS-4 MICRO COMPUTER SET

MCS-4 MICRO COMPUTER SET

NOVEMBER 1971

- Microprogrammable General Purpose Computer Set
- 4-Bit Parallel CPU With 45 Instructions
- Instruction Set Includes Conditional Branching, Jump to Subroutine and Indirect Fetching
- Binary and Decimal Arithmetic Modes
- Addition of Two 8-Digit Numbers in 850 Microseconds
- 2-Phase Dynamic Operation
- 10.8 Microsecond Instruction Cycle
- Easy Expansion – One CPU can Directly Drive up to 32,768 Bits of ROM and up to 5120 Bits of RAM
- Unlimited Number of Output Lines
- Single Power Supply Operation ($V_{DD} = -15$ Volts)
- Packaged in 16-Pin Dual In-Line Configuration

The MCS-4 is a microprogrammable computer set designed for applications such as test systems, peripherals, terminals, billing machines, measuring systems, numeric and process control. The 4004 CPU, 4003 SR, and 4002 RAM are standard building blocks. The 4001 ROM contains the custom microprogram and is implemented using a metal mask according to customer specifications.

MCS-4 systems interface easily with switches, keyboards, displays, teletypewriters, printers, readers, A-D converters and other popular peripherals.

A system built with the MCS-4 micro computer set can have up to 4K x 8 bit ROM words, 1280 x 4 bit RAM characters and 128 I/O lines without requiring any interface logic. By adding a few simple gates the MCS-4 can have up to 48 RAM and ROM packages in any combination, and 192 I/O lines. The minimum system configuration consists of one CPU and one 256 x 8 bit ROM.

The MCS-4 has a very powerful instruction set that allows both binary and decimal arithmetic. It includes conditional branching, jump to subroutine, and provides for the efficient use of ROM look-up tables by indirect fetching.

The Intel MCS-4 micro computer set (4001/2/3/4) is fabricated with Silicon Gate Technology. This low threshold technology allows the design and production of higher performance MOS circuits and provides a higher functional density on a monolithic chip than conventional MOS technologies.

Figure 2: Intel MCS-4

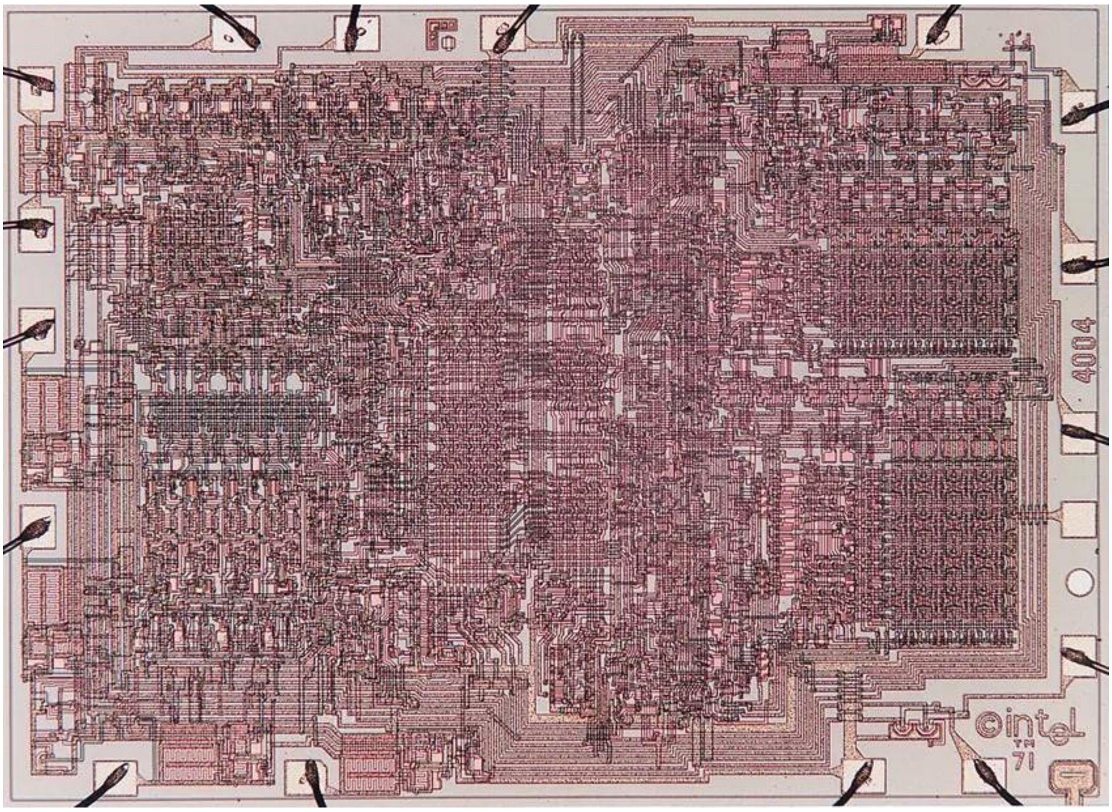


Figure 3: Intel 4004 chip die micrograph (from WikiChip)

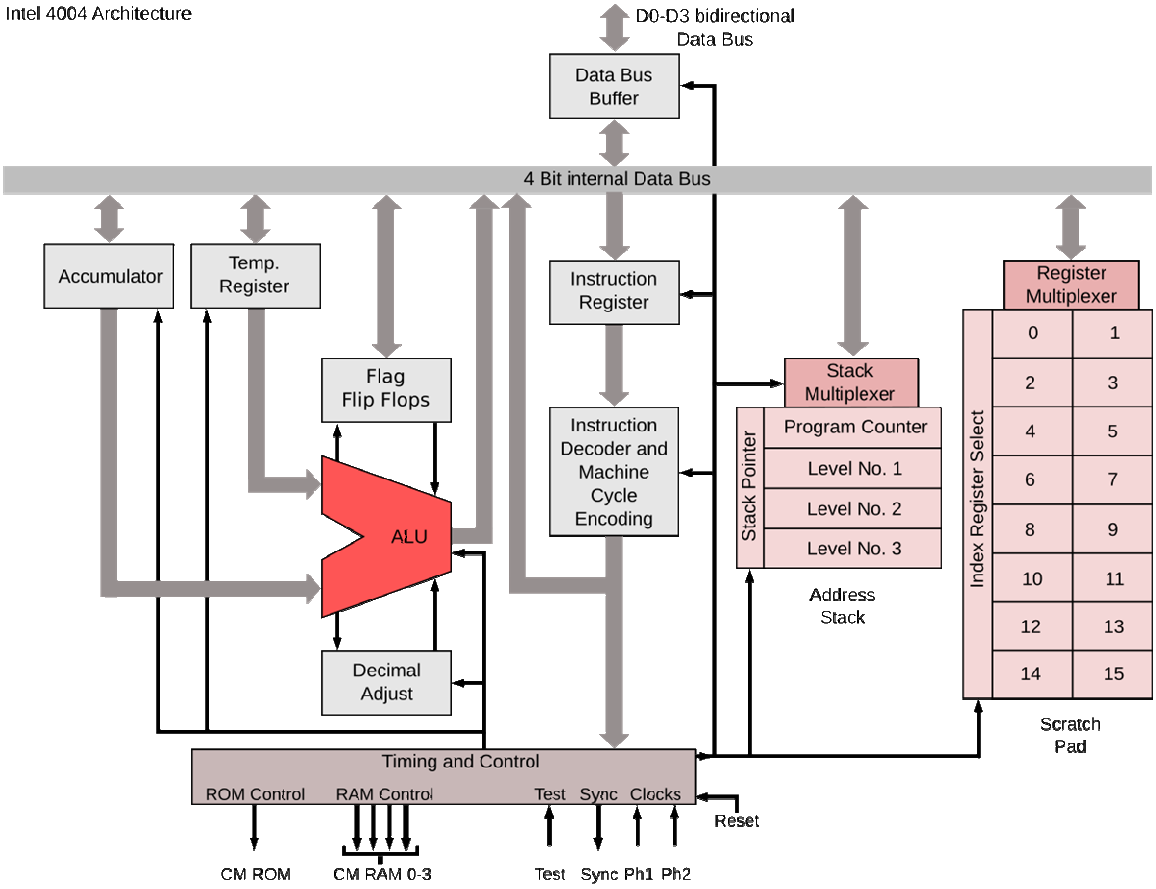


Figure 4: Intel 4004 architecture (from Wikipedia)

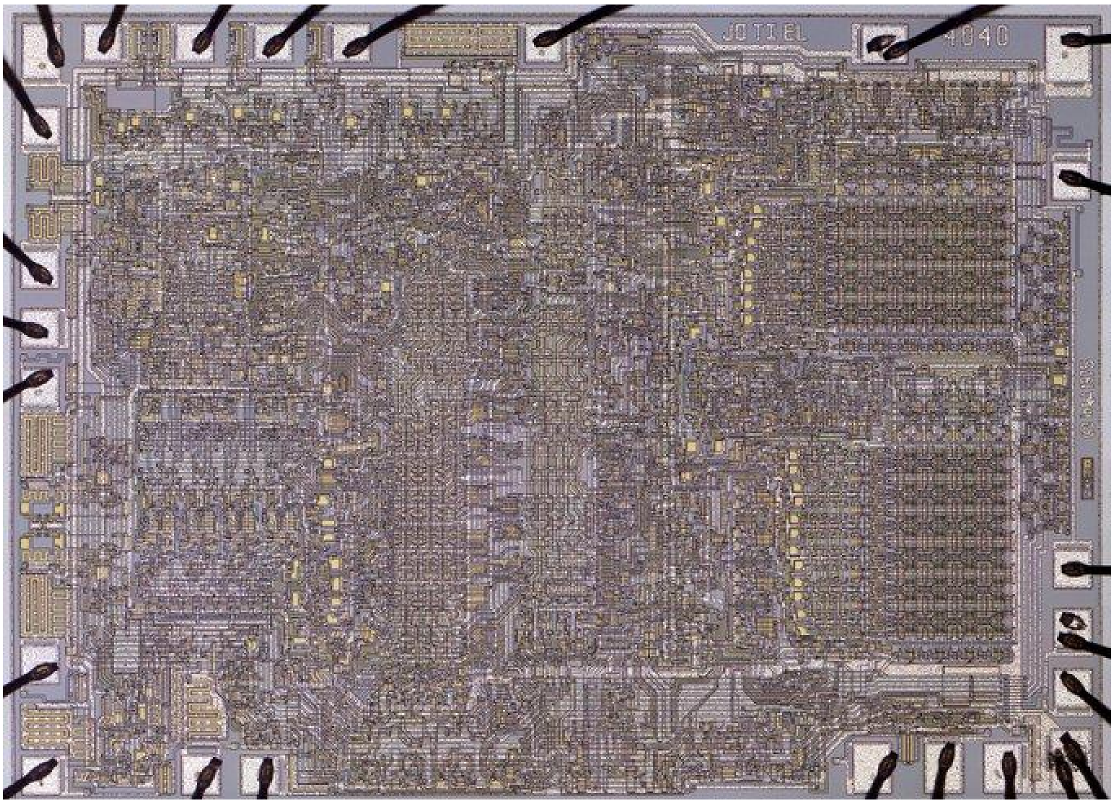


Figure 5: Intel 4040 chip die micrograph (from WikiChip)

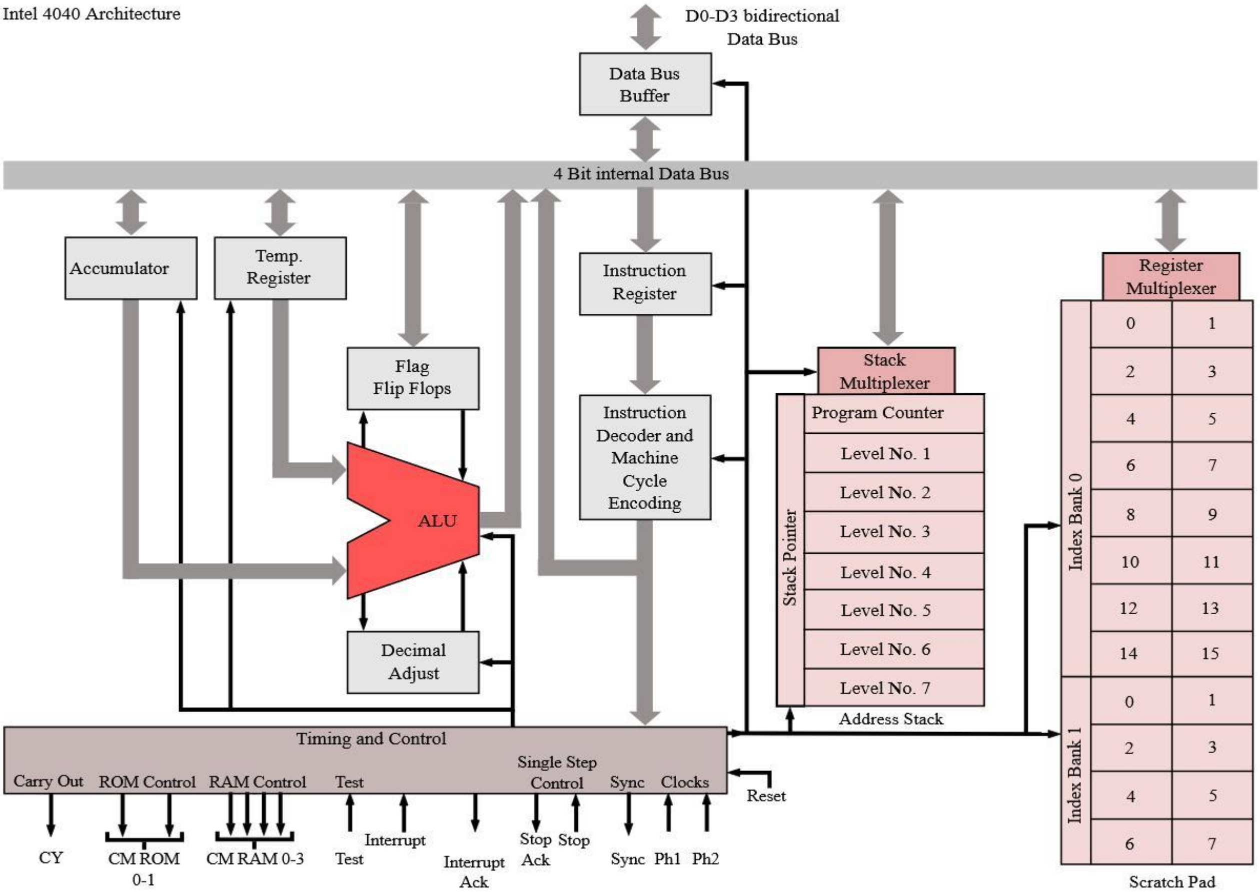


Figure 6: Intel 4040 architecture (from Wikipedia)

Figure 7: Intel 4004 microprocessor and 2 x 4003 10-bit parallel output shift register

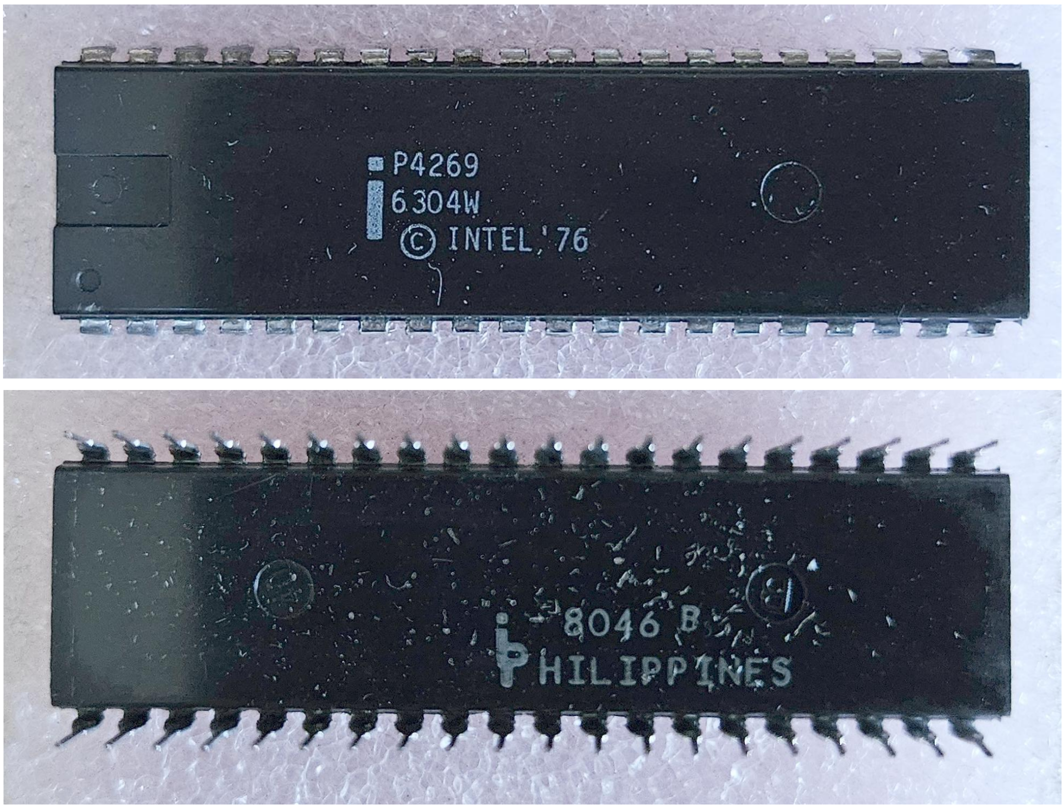


Figure 8: Intel 4269 keyboard/display interface, front and rear views