The MOS Silicon Gate Technology and the First Microprocessors

Federico Faggin

This is a preprint of the article published in La Rivista del Nuovo Cimento, Società Italiana di Fisica, Vol. 38, No. 12, 2015

1. – Introduction

There are a few key technological inventions in human history that have come to characterize an era. For example, the *animal-pulled plow* was the invention that enabled efficient agriculture, thus gradually ending nomadic culture and creating a new social order that in time produced another seminal invention. This was the *steam engine* which gave rise to the industrial revolution and created the environment out of which the *electronic computer* emerged – the third seminal invention that started the information revolution that is now defining our time.

All such inventions have deep roots and a long evolution. Engines powered by water or wind were used for many centuries before being replaced by steam engines. Steam engines were then replaced by internal combustion engines, and finally electric motors became prevalent; each generation of engines being more powerful, more efficient, more versatile, and more convenient than the preceding one. Similarly, the origin of our present computers dates back to the abacus, a computational tool that was used for several millennia before being replaced by mechanical calculators in the 19th century, by electronic computers in the 1950's, and by microchips toward the end of the 20th century.

The invention of the electronic computer was originally motivated by the need for a much faster computational tool than was possible with electromechanical calculators operated by human beings. This improvement was accomplished not only by performing the four elementary operations considerably faster than previously possible with calculators, but even more importantly, by adding the ability to program a long sequence of arithmetic operations that could be executed *automatically*, without human intervention. The addition of *programmability*, proved to be an immensely fruitful and versatile capability, allowing computers to soon become *universal symbol manipulators*, with countless new applications, far exceeding what even programmable calculators were originally intended to do.

Toward the end of the 20th century, the relentless progress in microelectronics combined with the digitization of all types of information, made computers so powerful, small, low-cost, and pervasive that many functions that previously required separate, single-function devices were subsumed by a single programmable mobile computer capable of handling the individual's needs for communication, computation, control, and storage of all kinds of information, be it numbers, text, images, sound, or video.

This paper will describe the history of two key inventions: the MOS silicon gate technology, and the microprocessor. These were the two developments that made it possible to replace huge and costly machines with pocket-size devices many thousands of times less expensive, and thousands of times more powerful than the early computers. This remarkable progress was due to the power and flexibility of microelectronics, the technology that provided the early transistors used in the second-generation computers.

What's remarkable is that while in the late 1950's transistors were just one of the *many* key components necessary to build a computer, less than 20 years later an entire monolithic computer could be built in a single chip of silicon, in the same physical volume that previously housed a single transistor!

This progress is unprecedented because in less than 30 years, a single chip weighting less than one gram, occupying a volume smaller than a cubic centimeter, dissipating less than one Watt, and selling for less than ten dollars could do more information processing than the UNIVAC I, the first commercial computer, which used 5200 vacuum tubes, dissipated 125 kW, weighted 13 metric tons, occupied more than 35 m^2 of space, and sold for more than one million dollars per unit.

Viewed from today's perspective, the early giant computers of the 1950's and '60's provided precisely the architectural blueprints of the type of symbol manipulator that the world needed. The microelectronics industry then did the rest, bringing the cost, size, and energy requirement down to the point where a computer could fit inside an electric toothbrush, a hearing aid, or an inexpensive toy – applications that were not only unimaginable, but even *incongruous* when computers were room-sized and cost millions of dollars.

After a brief history of microelectronics and computers, this paper will describe, from a first-person basis, the development of the silicon gate technology (SGT) at Fairchild Semiconductor, and the development of the early microprocessors (MP) at Intel, the two inventions that gave new life and impetus to the information revolution that began in the mid 1940's with the development of the first mainframe electronic computers using vacuum tubes.

2. – A brief history of microelectronics

Microelectronics officially started with the invention of the first transistor at Bell Laboratories in 1947 by John Bardeen, Walter Brattain and William Shockley, just one year after the first electronic computer, the ENIAC, was realized at the University of Pennsylvania. No one ever imagined at that time that these two inventions would merge, less than 30 years later, into a microchip, changing society in a fundamental way.

By replacing the electro-mechanical relays used in previous generations of calculating machines with vacuum tubes, the operating speed of ENIAC was increased by more than a factor of 1,000. This increase in speed came at a cost — vacuum tubes were bulky, power-hungry, and most of all, *unreliable.* The short meantime between failures in a single vacuum tube was a crippling problem for a computer needing thousands of them. Considering the reliability issues that then plagued electronic equipment, vacuum tube computers were over 10 times more likely to malfunction when compared to the most complex electronic equipment of that time. The search for a viable vacuum tube replacement with a more reliable, smaller, power-efficient, and lower cost solid state device had been ongoing since the 1920's. The first patent for a semiconductor "triode" was filed in 1925 by Julius Edgar Lilienfeld, and a more advanced device was patented by Oskar Heil in 1934. Both devices were *field-effect devices,* similar in principle to the current MOS transistors, but no commercial devices were ever produced.

The first Bell Labs transistor was a point-contact device, commercialized in 1948 by Raytheon with model CK703. However, point-contact transistors were too difficult to

build and too fragile to be useful, since even a modest mechanical shock could put an end to their operation. A new operating principle was needed. It was the breakthrough work of W. Shockley on diffusion transistors that introduced the new operating principles used in all modern bipolar transistors. This original work led to the first commercial *alloy-junction* transistors introduced by GE and RCA in 1951. From that point on, new applications for bipolar transistors started to grow rapidly – among them, the first hearing aids and portable radios – and in 1953, one million transistors were produced in the US alone.

All early transistors used a tiny single-crystal of germanium as their starting semiconductor material, and were built *one at a time*, just like the vacuum tubes. It was soon realized, however, that a better semiconductor material was necessary, since the temperature effects on germanium transistors could cause a thermal runaway with the self-destruction of the transistor. This unwanted effect was due to the rather narrow bandgap (.66 eV at 300K) between the conduction band and the valence band of germanium crystals. Furthermore, the operating frequency of germanium transistors was rather limited. A better material, silicon, with a bandgap of 1.11 eV at 300K, was soon identified, and silicon junction transistors were introduced commercially for the first time by Texas Instruments (TI) in 1954.

In 1957, a new company, Fairchild Semiconductor, was started in the San Francisco Bay Area by eight key engineers abruptly leaving Shockley Semiconductor. Among them were Robert Noyce, Gordon Moore, Jean Hoerni, and Jay Last. Fairchild's mission was to develop advanced bipolar junction transistors made with silicon to serve the needs of the emergent aerospace industry. Just for size, in 1957 the total production of transistors in the US was 29 million units. Today, a single chip costing less than $1 may easily contain more than 30 million transistors, including all their interconnections.

Before long, Fairchild Semiconductor became the leading company in the nascent microelectronics industry due to the seminal invention of the *planar process* by Jean Hoerni, a Swiss engineer. Up until that time, transistors had been fabricated one at a time. With the planar process, *many* transistors could be simultaneously fabricated, one next to the other, on the surface of a thin slice (called wafer) of a single-crystal silicon ingot. The planar process truly revolutionized the industry because it not only dramatically reduced the size and cost of transistors, but far more importantly, it made possible the monolithic integrated circuit (IC).

Hoerni's process consisted in fabricating an array of identical transistors on the surface of a silicon wafer with the diameter of less than one inch. This was done by recognizing that silicon dioxide could mask the diffusion of dopants in silicon. Therefore, the silicon doping necessary to create semiconductor junctions in specific places, could be achieved by first thermally growing a layer of silicon dioxide on the surface of a silicon wafer, and then opening windows in the oxide where the junctions were needed. The windows were defined by using photolithography followed by etching, a technique already in use to make printed circuit boards.

After the removal of the oxide by chemical etching in the areas not protected by the developed photoresist, the junctions could then be created by thermal diffusion of the appropriate dopants into the silicon. By using a sequence of thermal, chemical, and masking steps simultaneously affecting all transistors, it was possible to produce a batch of many identical transistors on the surface of a wafer. After the wafer was completed, it

was then cut into the individual transistors that were packaged into discrete components in the final step of the process.

With the invention of the planar process, it was straightforward to recognize that the transistors sitting next to each other could also be directly interconnected on the wafer itself, rather than being separated and individually packaged, only to be re-assembled and interconnected in a printed circuit board. What was missing was a process to directly *insulate* each transistor in the wafer so that the transistors would not interfere with each other – a process that was not needed when the transistors were physically cut and individually packaged. Bob Noyce, the head of Fairchild Semiconductor, invented a process to do so, making it possible for the first time to fabricate a truly *monolithic* integrated circuit (IC). It was 1961, and the first commercial IC, a simple resistor-transistor logic gate (RTL), was designed by Fairchild's Jay Last, and was first sold in 1962. A short ten years later, the microprocessor was born.

The history of the integrated circuit is another example clearly showing how most seminal ideas have roots that go back in time much beyond what is normally acknowledged when the history of the invention is written by the victors. The *idea* or conception of the integrated circuit was first presented at the US Electronic Components Symposium on May 7, 1952 by Geoffrey Dummer, who then asserted: "it now seems possible to envisage electronic equipment in a solid block with no connecting wires." This occurred at the same time when the industry had just managed to successfully fabricate the first diffused-junction bipolar transistors. But Dummer was never able to bring his idea into a working device, despite trying for many years.

TI's Jack Kilby, who won the 2000 Nobel Prize in Physics for his contributions to the invention of the IC, is credited with making the first commercial integrated circuit in 1959. This circuit, however, was *not a monolithic* IC, because it required the individual positioning of *separate* solid-state components inside a small package followed by manually wiring them in situ. The interconnection of the separate components was supposed to eventually be performed in a single operation. However, that idea was never commercially realized. It was Hoerni's planar process combined with Noyce's transistor insulation process that provided the first *practical* method to fabricate monolithic ICs; a method that was soon universally adopted by the world's microelectronics industry.

Planar transistors and ICs were initially very expensive, limiting their early applicability primarily to the US missile and space program where size, power dissipation, and reliability were of paramount importance. However, the inherent potential of the planar process was huge because over time, the cost of a single transistor within an integrated circuit could be dramatically reduced by increasing the wafer size and reducing the physical dimensions of the transistor. Thus more ICs, each integrating more and more transistors, could be simultaneously batch-fabricated in a single silicon wafer.

The advantages of using ICs over discrete transistors were many and compelling, as long as it was possible to *standardize* the functions being performed by the ICs (the cost of an IC depends dramatically on the unit volume produced). However, the circuit designers that previously used discrete components to create a large variety of different circuits, resisted the idea of standardization, slowing down the early adoption of ICs. As the cost of ICs diminished and more standard building blocks became available, rationality prevailed and these early difficulties were gradually overcome. With the

development of the microprocessor, the boundary of standardization was pushed beyond the individual circuits to an entire system.

By the mid-sixties the cost of ICs had become low enough to enable many new industrial and commercial applications, leading to a market expansion and thus to lower cost-per-function. This in turn led to the emergence of new *consumer* applications, fueling the virtuous cycle responsible for the establishment of a vigorous and rapidly-growing semiconductor industry.

In 1965, Gordon Moore noticed that the number of transistors in an IC for which the cost per transistor was nearly minimized, had been doubling every year since 1962. He boldly predicted that this trend would continue for the following 10 years. This observation became known as Moore's law, forecasting an exponential growth of the number of transistors in an IC. Of course Moore's law is not a physical law, and therefore the doubling-time must increase over time, as more and more physical limitations are progressively reached. And eventually the process must halt when the transistor size approaches the atomic size. In fact, the doubling time has gradually gone from one year in 1963 to three years in 2013, averaging two years for the period 1970-2014. But since the doubling-time increases slowly with time, Moore's Law has been a good predictor of the number of transistors that can fit in future generations of ICs, and thus it has served very well as a valuable planning and forecasting tool.

All early ICs used *bipolar* transistors whose principle of operation is essentially identical to the early discrete diffusion transistors made with germanium or silicon. In 1959, the same year the planar process was invented at Fairchild Semiconductor, the MOSFET (Metal Oxide Semiconductor Field Effect Transistor) was invented by Dawon Kahng and John Atalla at Bell Labs. This was another seminal invention. The MOSFET, or simply MOS transistor, works by using a different operating principle than bipolar transistors. In the MOS transistor, the signal amplification is achieved by controlling the conduction of charge carriers at the *interface* between silicon and silicon-dioxide. This is accomplished by changing the voltage applied to a *gate* electrode sitting atop a thin insulating layer of silicon dioxide that bridges two back-to-back junctions (called *source* and *drain)*. In a bipolar transistor instead, the conduction depends on the *bulk diffusion* of carriers within the *volume* of silicon (the base) in common with two back-to-back junctions. The three terminals connecting the back-to-back junctions are called emitter, base, and collector; with the emitter-base junction forward biased and the base-collector junction backward biased.

MOS transistors have the important feature of being self-isolating. Therefore, they are simpler to manufacture and quite smaller than bipolar transistors. Even more importantly, they are *surface-effect* devices, therefore their physical size can be proportionally scaled down much more easily than bipolar transistors, a capability that was not recognized at the time of their invention, but later became the basis for the dimensional scaling of transistors that fueled Moore's law for the last 50 years.

In the early 1960's, however, MOS transistors had speed and reliability characteristics that were much worse than bipolar, and were generally considered a second-best alternative to make ICs, except for a class of less-demanding applications where cost more than performance or reliability were of paramount importance. It took many years for the industry to learn how to fabricate reliable MOS devices. The key development to

bring MOS technology to maturity was the silicon gate technology (SGT) invented at Fairchild Semiconductor in 1968 by Federico Faggin and Tom Klein.

With SGT, it became possible to make reliable MOS ICs that were 5 times faster and had twice the number of transistors per unit area than MOS ICs made with aluminum gate (for random logic circuits). With SGT new categories of devices could also be produced, such as dynamic random access memories (RAMs) (difficult to fabricate with metal gate due to the high junction leakage of metal-gate transistors), image sensors using charged coupled devices (CCD), non-volatile memories, sophisticated analog ICs, and microprocessors.

For the first time in the history of computers, all the key components necessary to implement a general-purpose computer could be made with the same technology, leading naturally to a *monolithic computer*. SGT was also the core technology that allowed the scaling of MOS transistors for the following 40 years, eventually replacing bipolar technology in nearly all applications.

3. – A brief history of computers

As mentioned in the introduction, the roots of modern computers go back to the abacus. Therefore, they are far deeper than the roots of microelectronics which are based on electromagnetism and quantum physics: unknown physics before the XIX and XX centuries respectively. The abacus dates back at least 4600 years, and is the first known instrument to aid human beings in performing arithmetic calculations. We have to go forward about 4200 years to find the first example of the next milestone: a mechanical calculator invented by Blaise Pascal in 1642, called Pascaline. Since this machine could only perform additions and subtractions, however, its usefulness was quite limited. In fact, a skilled abacus user could compute faster than a skilled Pascaline user.

In 1673, Gottfried Leibnitz devised a conceptual method for mechanically performing multiplications and divisions, but the technology of his day did not support the fabrication of such devices. We have to wait another 178 years before the first four-function mechanical calculator entered production. This mechanical calculator was called Arithmomètre. It was invented by Tomas de Colmar and it was made commercially available for the first time in 1851, exactly 100 years before the first commercial electronic computer was sold. This machine was sturdy and practical enough to be used routinely, launching an industry that lasted until the 1970's when the electromechanical calculator was superseded by electronic calculators based on microchips.

The first *programmable* machine was the Jacquard loom, invented by Joseph Marie Jacquard in 1801 to automatically weave complex patterns in textiles. The loom was controlled by punched cards that stored the "program" directing the step-by-step operation of the loom. The Jacquard loom was a highly successful invention, and is also the first example of an automatic system controlled by a program; a procedure that can be changed without changing the machine itself. It is a forerunner of modern computers, though on the surface, it appears to have little in common with computers.

The idea of using punched cards was later employed by Herman Hollerith to produce an electromechanical tabulator that could rapidly sort data. This device was used to compile the data of the 1890 US census, demonstrating great improvements over manual

sorting. Interestingly, Hollerith was the founder of Tabulating Machine Company, which later changed its name into IBM.

Another important area of development was associated with automatic telephone exchanges based on relay commutators (relays were invented in 1835 by J. Henry and improved over the years). This application area formed another major thread that was later woven into the modern computer design. Victor Shestakov in Russia and Claude Shannon at Bell Labs in the US, in the period between 1935 and 1937, independently "discovered" that Boolean logic – a binary logic developed in the 1840's by Charles Boole – was the perfect mathematical formalism to describe switching systems and calculating machines.

In an unrelated development, Alan Turing in England invented the Turing Machine in 1936. It was a mental experiment that he used to falsify Hilbert's Decision Problem, posed by the mathematician David Hilbert in 1928. His Universal Turing Machine provided an abstract model to describe a class of machines capable of executing any general algorithm, thus giving birth to theoretical information science.

All the threads I mentioned earlier came together in 1941 with the realization of the first fully functioning Turing-complete electromechanical computer, the Z3, designed and built by Konrad Zuse in Germany. This machine used a binary floating-point architecture based on 22-bit words and a CPU made with about 2300 relays. The read-write memory (write once) to store programs and data was ingeniously implemented by using 35 mm punched film (the same film used in cameras). The clock frequency was about 5 Hertz, which makes us smile now, considering that microchips today have clock frequencies of several GHz. The Z3 takes us to the threshold of the electronic computer era.

Electronic computers started in 1943 with a secret project financed by the US Army to develop a computer capable of rapidly calculating ballistic trajectories. The major new idea was to replace relays with vacuum tubes to increase the computational speed by at least a factor of 1000. The result was ENIAC, the first fully functional electronic computer designed and built by J. Mauchly and P. Eckert, and completed in 1946. ENIAC had an instruction cycle of 200 microseconds and the program was provided by plugboards and switches, a fairly rudimentary and laborious method. It employed 17,468 vacuum tubes, occupied an area of 167 m^2, dissipated 150 kW of power, and weighted 30 tons. The meantime between failures was a few hours, due to the poor reliability of the vacuum tubes.

ENIAC however was not yet a complete solution because it lacked the ability to store a program in its electronic memory. Therefore, the first electronic computer to have all the essential features of the modern machines was the EDSAC, the first *stored-program computer* developed at Cambridge University by M. Wilkes, with the collaboration of the famous mathematician John von Neumann, who was the one suggesting to use the same memory that stored data to also store the programs. The EDSAC was completed in 1949 and its program and data memory was realized with a serial memory using a mercury delay line with 1024 17-bit words. The modern random access memory (RAM) had not yet been invented.

All early computers were one-of-a-kind research machines, until the introduction of the first commercial electronic computer in 1951. This machine was the UNIVAC I, which was a stored-program machine with a *serial* main memory of 1024 12-bit words. The UNIVAC I used for the first time a magnetic tape secondary memory to increase the

overall memory. Capable of executing 500 multiplications per second, UNIVAC I used 5200 vacuum tubes dissipating 125 kW, and sold a total of 46 units at more than one million dollar per unit, demonstrating for the first time the commercial viability of computers. Twenty years later, the first single-board computer using the Intel 4004 microprocessor had similar performance to the UNIVAC I in a single printed circuit board of 25x25 cm^2 dissipating approximately 10 W and costing a few hundred dollars! Less than ten years later, that single-board computer could be integrated into a single chip – a *monolithic computer*.

All early mainframe computers used vacuum tubes until 1957, the year when the first commercial transistorized computer, the Philco Transac S-2000, was introduced. Two years later, the Olivetti Elea 9003, and the IBM 7090 transistorized computers were also commercialized. From that point on, all new computer models used transistors. With transistors, the size, power dissipation, speed, and especially the reliability of computers were drastically improved. This was the coming of age of computers, and from this point on computers showed their worth and versatility in a rapidly growing variety of applications.

The 1960's marked a period of rapid evolution for computers, from very large to very small, from minicomputers to supercomputers. For example, in 1963 the SAGE system began operation. Designed by IBM in collaboration with the US Air Force to coordinate the operations of 24 radar stations in Northern America, SAGE became the largest computer ever built (area: 2000 m^2; weight: 275 ton; power consumption: 3MW) and created the world's first real-time computer network. In 1964 IBM introduced the System/360, a large family of compatible and scalable computers capable of covering a wide range of applications. It became a highly successful product line with novel software, like the first sophisticated operating system. That same year, Control Data Corporation introduced the CDC 6600 the world's first supercomputer. The CDC 6600 was designed by Seymour Cray who later founded Cray Computers, a trailblazing supercomputer company for many years. The CDC 6600 cost more than $8 million and was 10 times faster than the fastest computer of that time.

By 1965, the commercialization of a family of logic ICs allowed the reduction of the size of computers to that of a small cabinet, or a large box, giving birth to the *minicomputer*. The minicomputer was a scaled down version of a mainframe computer, intended for applications where mainframes were too big and too costly to be usable. First introduced by Digital Equipment Corporation (DEC) with the model PDP-8, minicomputers opened up new application areas for computers, particularly for communication and for all types of control systems, further expanding their reach.

The versatility of the computer is obviously due to its programmability. Therefore, the hardware alone is not sufficient to solve any particular problem; one also needs a program that once loaded in the memory of the computer, makes the hardware-software combination perform the desired function. The creation of software necessary to handle specific functions started a parallel development to the computer hardware, marking the beginning of the software industry. Furthermore, since a computer is a *universal* symbol manipulator, as computers became faster and more affordable, more and more applications could be performed with appropriate software. In several cases the applications appeared initially incongruous with the expectation of what a computer should be able to do; for example, speech recognition, playing chess, or processing video.

Over time, the percentage of the total information processing investment dedicated to software grew, while the cost of the hardware decreased. Today the preponderant cost resides in the software.

The art of programming relies on the ability to conceptualize the solution to a problem, or task, into a series of algorithms – procedures that a computer can efficiently execute. The importance of programming has grown dramatically with far reaching consequences.

Although it is beyond the scope of this paper to explore this new field of human endeavor, I wish to at least underscore that the information revolution that is sweeping society has brought to the forefront the deep and unsuspected relationship between the nature of information and the nature of reality. This is a particularly fascinating subject that was completely unknown to physicists and philosophers until the 1960's; a subject that in my opinion will shape a new sense of self and a new sense of reality in human society, with major social consequences.

We have now reached the state of the art at the time when the inventions of the SGT and the microprocessor occurred, as will be described in the rest of the paper.

4. – The invention of the MOS silicon gate technology

4.1 *The state-of-the-art in the mid-Sixties*

As mentioned earlier, more than 95% of all commercial integrated circuits manufactured in 1967 used bipolar transistors. Compared to bipolar, MOS transistors were surface-effect devices of simpler construction, fabricated using similar processing steps, but in a different sequence. However, that simplicity did hide a major difficulty: MOS devices were very sensitive to the presence of impurities in the manufacturing process. In particular, even an infinitesimal amount of sodium could contaminate the IC causing a substantial threshold voltage drift over time that could render the IC useless in the field.

The MOS technology had two key advantages over bipolar: (1) higher circuit density by a factor of 5 to 10; and (2) lower power dissipation by a factor of 5 to 10. But its speed in the mid 1960's was about 50 times lower than bipolar, limiting its usefulness to a short list of applications. Overall, it was generally believed that MOS circuits were too slow and too unreliable to ever challenge the market dominance of bipolar technology.

The prevalent MOS process technology of that time used enhancement mode, P-channel MOS transistors with gates made of aluminum, the same metal that was used to interconnect the transistors within an IC. The standard MOS fabrication process started by thermally growing the field oxide in a [111]-orientation silicon wafer, followed by masking and etching the field oxide in the regions where the source and drain of the MOS transistors had to be located. The process continued with the following steps: (1) doping the source and drain regions with boron, (2) growing some additional thermal oxide over the exposed source and drain regions, with the concomitant diffusion of the boron, (3) removing the field oxide in the gate region of the transistor, and growing a thin oxide, the gate oxide. After the gate oxide growth, contact areas to the source and drain junctions were defined and etched, followed by aluminum deposition, masking and etching to create the gate electrodes and the interconnections. The final passivation step consisted in the vapor deposition of a layer of silicon dioxide (called *vapox*) over the entire structure, followed by etching the vapox over the aluminum pads. The pads were large aluminum areas located at the edges of the chip where thin aluminum wires were ultrasonically

bonded to the package.

Due to the inevitable misalignment of the gate mask with respect to the source and drain mask, it was necessary to have a fairly large overlap area between the gate region and the source and drain regions of the transistor, to insure that the channel controlled by the gate would positively bridge both source and drain under worst-case misalignment of the gate mask. This requirement resulted in a significant increase in the gate-to-source and gate-to-drain parasitic capacitances, over and above the amount that would be strictly necessary if perfect alignment was possible, causing a major performance degradation.

The threshold voltage of the transistors was in the range of -4 to -9 volt, dictated primarily by the crystal orientation and the resistivity of the silicon wafer, the gate oxide thickness, and the work-function difference between aluminum and silicon. To maintain the isolation of the various transistors within the IC, it was essential that the threshold voltage of the parasitic MOS transistors be higher than the supply voltage[1]. This requirement was accomplished by using a sufficiently thick field-oxide; and the higher the supply voltage, the higher the thickness of the field oxide had to be.

However, increasing the thickness of the field oxide interfered with the integrity of the aluminum interconnections going over the oxide steps, due to the thinning out of the aluminum at those steps. This effect could potentially create severe reliability problems in the field due to electro-migration[2], and also cause major yield problems if the metal did break. The high threshold voltage MOS technology being used at that time achieved a delicate balance between the various conflicting requirements with only a small safety margin.

The ideal conditions for MOS technology would have been to use wafers with [100] crystal orientation, instead of the [111] orientation used in the high-threshold-voltage process, because that change would have caused the threshold voltage of the MOS transistors to be in the range of -2 to -5 volt, allowing the supply voltage to go from -24 to -15 volt, with the major benefit of halving the power dissipation for the same speed. Unfortunately, with [100] starting material, the field threshold voltage was lower than the supply voltage for the maximum allowed field-oxide thickness. The only solution then was to use N+ “channel-stoppers,” which were diffusions underneath the field oxide to raise the field-oxide threshold voltage. However, this solution was undesirable because it required an additional masking step, and even more importantly, it drastically reduced the circuit density, increasing the cost of the IC; a hardly worthwhile tradeoff. In the mid-Sixties, the industry was still trying to find a way to reduce the MOS transistor threshold voltage without sacrificing the circuit density.

([1]) A parasitic MOS transistor is an unintended transistor obtained when a metal line over the field oxide crosses two junctions. In this case the junctions act as the source and the drain of a parasitic MOS device, with the metal line acting as the gate of such device. If the voltage on the metal line is higher than the threshold voltage of the parasitic MOS (called the field-oxide threshold voltage), a stray conduction path between the two junctions is created. Therefore, if there is a voltage difference between the two junctions, a stray current will flow between them, rather than the junctions being isolated. This possibility is particularly damaging in the case of dynamic circuits because electrical charges stored in MOS transistor gates may rapidly leak away through this unwanted stray paths, causing malfunctions.

([2]) Electro-migration causes the aluminum interconnects to open before the expected lifetime of the device if the current density in the aluminum exceeds a certain limit. The thinning of metal lines over the oxide steps caused by self-shadowing during the aluminum vacuum deposition process could lead to such problem in the field, creating a major reliability hazard.

The other major limitation of MOS technology was the very high parasitic gate-to-source and gate-to-drain overlap capacitances – a widely recognized problem with no known solution. The overlap capacitance with the most adverse consequences on circuit performance was the gate-to-drain parasitic capacitance, C_{gd}. Due to the well-known Miller effect, the effective gate capacitance of any given transistor is increased by its C_{gd} multiplied by the gain of the circuit of which the transistor is a part. Since the gain is generally much greater than one, the impact of C_{gd} on the switching speed of the transistor is considerable. Furthermore, since the variability of C_{gd} due to the unpredictable direction of the misalignment is very high, some wafers would be impacted very little and some other wafers would be impacted very much, producing a large and undesirable spread in the speed of the ICs produced.

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 in 1968, and with aluminum-gate devices through the use of ion implantation, in 1969-1970. Ion implantation was a new technology that allowed doping silicon in highly controllable amounts, far and beyond what was possible with thermal doping, particularly for low doping concentrations, which were nearly impossible to achieve with thermal doping. Ion implantation was the cure to increase the field-oxide threshold voltage with [100] oriented silicon without channel stoppers. With SGT, instead, the MOS threshold voltage was reduced without having to use [100] material or ion implantation. But most importantly, the parasitic overlap capacitance problem and the reliability problems were also permanently solved, as we will see below.

4.2 The self-aligned gate

In 1966 Robert Bower realized that if the gate electrode was defined first, the channel boundaries of the source and drain regions would be "*self-aligned*," thus avoiding the overlap between the source-drain mask and the gate mask. This method would not only minimize the parasitic overlap capacitances, but it would also make them insensitive to misalignment. He proposed a method in which the aluminum gate itself was used as a mask to define the source and drain regions of the transistor at the gate-region boundaries. However, since aluminum could not withstand the high temperature required for the conventional thermal doping of the source and drain junctions, Bower proposed to use ion implantation, a new doping technique still in development at Hughes Aircraft, his employer, and not yet available at other labs.

While Bower's idea of using the aluminum gate as a mask to define the source and drain regions was conceptually sound, in practice it *did not work* because aluminum could not survive the follow-on high temperatures steps necessary to complete the process, *after* the aluminum had been deposited. In particular, it was impossible to repair the radiation damage done to the silicon crystal structure by the ion implantation. Thus, Bower's idea was good in principle, but a more refractory gate material than aluminum was needed. Bower's process was described in [1] but it was never used to produce commercial integrated circuits, and his process was unknown before the publication date of his patent. I independently developed in early 1968 a process architecture similar to his, but using polycrystalline silicon instead.

In 1967 John C. Sarace and collaborators at Bell Labs fabricated discrete transistors with gate electrodes made of vacuum-evaporated amorphous silicon, successfully

building working self-aligned gate MOS transistors [2]. Their experiment started with a wafer in which they grew a thin oxide, followed by vacuum deposition of amorphous silicon. They then masked the silicon to define the gates, which were shaped like annular regions. They then etched away the thin gate oxide, except where it was protected by the silicon gates. Finally they doped the wafer with boron to create the source and drain junctions. After a thin layer of oxide was grown over the entire structure, the following sequence of steps was performed: contact mask, aluminum evaporation, and metal mask; thus completing the process in the usual way.

The ring structure of the gate created the drain electrodes inside the ring, while the source electrodes of *all* the transistors were connected together, since it was the common diffusion outside all the rings. Therefore, the process they described was useless for the fabrication of integrated circuits; it was just a proof of principle, suitable only for the fabrication of *discrete transistors*, and was not pursued further by its investigators.

In late 1967, Tom Klein of Fairchild Semiconductor, experimented with MOS capacitors, called CV dots[3], where the aluminum was replaced with heavily-doped P-type amorphous silicon. He observed that the work function difference between the doped amorphous silicon and the lightly doped single-crystal N-type silicon was 1.1 volt *lower* than the work function difference between aluminum and the same N-type silicon. This meant that the threshold voltage of MOS transistors built with silicon gate could be 1.1 volt lower than the threshold voltage of MOS transistors with aluminum gate fabricated from the same starting material. Therefore, starting with [111] orientation silicon, it was possible to simultaneously achieve both low-threshold-voltage MOS transistors and high field-oxide threshold voltage.

Tom Klein thus established that by using P-type doped silicon gate it would be *potentially* possible not only to create self-aligned gate transistors, but also to achieve a low threshold voltage process, using the same crystal orientation employed in the high threshold voltage MOS process. However, Klein could not figure out how to *architect* the process to make the isolated transistors necessary for the fabrication of self-aligned-gate integrated circuits. At this point I enter the picture, and I will continue the story with a first-person narrative, after a brief description of my background.

4.3 ***Before Silicon Valley***

Born, raised and educated in Northern Italy, I graduated in 1960 in radio technology from the A. Rossi Technical Institute in Vicenza at the age of 18. During my senior year in school, I became interested in computers, reading everything I could find about them. My first job in 1960, was assistant engineer at the Olivetti Electronic R&D Laboratory near Milan, Italy, where Olivetti was developing their early electronic computers. Due to a series of fortunate circumstances, I ended up co-designing and building a small experimental electronic computer with 4k 12-bit words of magnetic core memory. I was 19 years old and I had four technicians working for me, helping me with the construction

---

([3]) CV-dots was a technique used in the 1960's to test the cleanliness of silicon dioxide. It consisted in evaporating small aluminum dots through a metal mask with small holes in contact with oxidized silicon. This method allowed to quickly measure the capacitance-voltage relationship of small capacitors, and to also measure the threshold voltage drift with temperature, without having to perform a photolithographic and etching process to define the capacitors.

of that computer. The computer used approximately 1000 logic gates made with germanium transistors (fabricated in Italy by SGS), requiring about 200 small PC boards, housed in several card cages that were mounted on a single equipment rack. This early experience turned out to be absolutely invaluable, setting the stage for much of my future career.

At the end of that successful project I decided to return to school and study physics at Padua University, because I wanted to deepen my understanding of solid state physics. Since I needed to complete my Olivetti project, I began university in January 1962, a few months after the academic year had already started, forcing me to study incessantly to catch up. I studied with much pleasure and I graduated in October 1965, receiving a *laurea* degree in physics (the only university degree in Italy in those days), summa cum laude, with an experimental thesis on flying spot scanners. I was given a job at the University as soon as I graduated, and I taught Electronics Laboratory to 3rd-year physics students during the 1965-1966 academic year.

In the summer of 1966, I decided to leave the university and join CERES, a startup company in the Milan area, to work for my old boss at Olivetti, who now was the head of this company. CERES was developing thin film circuits, and was also the Italian agent of GMe, General Micro-electronics, Inc., the world's first MOS company. GMe had started in Silicon Valley a couple of years earlier with founders coming from Fairchild, and its first commercial MOS product was a 20-bit shift register. Unbeknownst to me, GMe was struggling to make reliable MOS ICs.

My first assignment at CERES was to take a one-week course at GMe in Sunnyvale, California, on MOS technology. This was necessary to be able to properly explain the GMe product line to potential customers in Italy. My trip to Silicon Valley was my first intercontinental journey, and I found the San Francisco Bay Area immensely energizing. After my return to Italy, CERES received an order for a few 100-bit MOS shift registers – the most recent product of GMe – from the University of Rome. Unfortunately, GMe could never deliver, and before long it was sold to RCA, thus ending the relationship with CERES.

That one-week course on MOS technology, however, landed me a job at SGS-Fairchild in Agrate Brianza, Italy in 1967. SGS-Fairchild was at that time the only Italian semiconductor company, 30% owned by Fairchild Semiconductor, and a licensee of Fairchild's bipolar technology. I worked in the newly-formed R&D department of SGS, and given my brief exposure to MOS, I was assigned the job of developing their first MOS process technology. When the process was done, I also designed the first two commercial MOS IC products for SGS, which I successfully completed in early 1968 [3]. A few months later I was promoted to the position of MOS group leader.

In September, 1967 I married Elvia Sardei and settled in an apartment in Agrate Brianza. Elvia grew up in Vicenza, like me, and attended the same university in Padua where we had met several years before. Toward the end of 1967, SGS asked me if I was interested in going to the US for 6 months, as part of an engineer exchange program between SGS and Fairchild. I jumped at the opportunity to return to Silicon Valley, and Elvia was excited as well. We arrived in the San Francisco Bay Area in February, 1968, rented a small apartment in nearby Mountain View, and I worked in the R&D Laboratory of Fairchild Semiconductor in Palo Alto. It was the start of a great adventure!

I was supposed to stay for six months and then return to Italy, but while I was in the US, Fairchild sold its interest in SGS-Fairchild, and they asked me to stay. That's how Elvia and I decided to make Silicon Valley our new home.

4.4 Developing the SGT and the first commercial IC with SGT

In February 1968, I joined the MOS process development group of the Fairchild Semiconductor R&D Laboratory in Palo Alto, CA, then directed by Les Vadasz. I was a guest engineer, and my US boss was Les Vadasz, who immediately gave me the 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 since I was well aware of the deleterious effects of the parasitic drain to gate capacitance in MOS transistors. I was not aware, however, of the previous work done at Bell Labs and at Hughes Aircraft. I learned about the Bell Labs work only after I had successfully completed the project, and about the Hughes Aircraft work many years later. But I was told, of course, about the experiments of Tom Klein with CV dots, and about the concept of self-alignment, which I had already heard before. What no one at Fairchild had yet devised was the necessary process architecture to make integrated circuits with silicon gate.

When I started at Fairchild, it wasn't even known how to precision-etch thin films of silicon. Therefore, 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. To be able to characterize the process, I also had to design a suitable test pattern that would allow me to measure all critical parameters. If all worked well, I would then have to design an appropriate commercial integrated circuit to convincingly prove the superior performance and reliability of the new technology.

For many days I struggled trying to figure out how to architect the process. I asked all the local experts how they would do it, but I got nowhere. After a week or so it donned 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. Only then did I realize how much I had been conditioned by the old way of making MOS transistors where the first step was to define *only* the source and drain of the device. 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 to the source and drain junctions in the most efficient manner. Obviously one could use an aluminum strip to connect the silicon gate to the appropriate junction, but that would require a lot of area. Desiring a better solution, I came up with the idea of making a *buried contact*, which consisted in directly connecting the amorphous silicon to a junction without using metal. This method required an additional masking step, but it would considerably increase the circuit density because metal could now run over the buried contact. Having already some experience with chip layout, I appreciated this possibility since the density of conventional MOS circuits was severely limited by the metal interconnections. The addition of the buried contacts to the silicon gates that were also "buried" under oxide, would afford the same interconnection density achievable with two layers of metal, at the cost of only one additional masking step.

The alternative way of making the smallest possible silicon-to-junction contact was to

overlap the amorphous silicon onto a portion of the tub, then opening a contact large enough to overlap both the silicon and the junction area. The contact area would be covered with a dash of metal, completing the connection. This type of contact was called "butting contact." It was relatively small, but it penalized the interconnection density, particularly for random logic circuits where the interconnections dominated the overall circuit density. For example, using the SGT with only butting contacts, the chip area of the first microprocessor could have been reduced by about 30% over the area required with metal-gate MOS. Using SGT with *buried* contact, the chip area could be further reduced by about 40% over the SGT with butting contact; this amounted to half the size of the metal-gate version!

These were impressive numbers because the chip cost is proportional to the chip area divided by the yield (the wafer yield is the number of good chips divided by the total number of chips in a wafer). Now, the yield decreases with increasing chip size, slowly at first for small chips, but rapidly as the chip size approaches a limit, called maximum chip size, beyond which the yield is too small for the chip to be manufacturable[4]. Given this yield behavior with growing chip size, saving 40% of the area when a chip is approaching the maximum chip size, reduces the cost far more than 40%.

In summary, the process architecture I devised was to first open the areas in the initial oxide where the source, drain and gate of each transistor were to be located (tub mask). This step was then followed by the growth of the gate oxide, and followed by the vacuum-deposition, masking and etching of the silicon layer, thus defining the gates and the silicon interconnection layer. Next, the thin oxide was etched inside the tubs where it wasn't protected by the poly-silicon, thus defining the source and drain regions of the transistors. Notice here that the process is properly called *self-aligned* because a misalignment between the tub mask and the gate mask would only slightly change the geometry of the source and drain regions, but would not change the gate overlap capacitances of the transistors.

After the removal of the thin oxide, the source, drain and amorphous silicon would be doped with boron, and here the silicon gate would act as a mask against the doping occurring in the gate region. After doping, a thin layer of the best possible oxide would be thermally grown to protect the exposed source, drain and gate areas, followed by the deposition of a thicker layer of good quality vapox (which required a higher temperature than could be tolerated by aluminum). Contact mask, contact etching, aluminum deposition, metal mask, and passivation would then complete the process [4], [5], [6].

I also proposed a variant of the above process to make buried contacts, as follows: After the gate oxide was grown in the tubs, the gate oxide was removed in the areas where the buried contacts between silicon and junctions were to occur. Then the processing steps following the buried contact mask continued as previously described. The idea here was that when the boron doping would be performed, boron atoms would completely diffuse *through* the thin layer of deposited silicon in contact with the single-crystal silicon of the wafer, forming a junction in the single-crystal silicon itself, thus

---

([4]) The maximum chip size depends on the defect density achievable with any particular process, at any particular time. One of the key improvements responsible for Moore's law has been a relentless reduction of the defect density, between 1968 and today, by more than a factor of 10^6.

creating an *isolated* contact that would later be protected by oxide. Therefore, aluminum could go right above the buried contact, allowing a greater circuit density to be achieved.

When ten days after my arrival I described to Vadasz how I proposed to make silicon gate ICs, he approved my proposed process architecture, but said that the buried contact would never work and when I respectfully disagreed, he didn't even want me to try it out. Vadasz had a forceful management style and I decided not to argue with him, but when a few weeks later I designed the test pattern necessary to properly characterize the new technology, I decided to place a couple of structures in it that would allow me to verify if the buried contact would actually work, and also to characterize its properties. I disobeyed because it cost nothing to do so, and I strongly believed in my idea.

The next step was to develop a suitable silicon etching solution. Again, nobody knew how to do that. I finally found a chemical engineer, and I asked him what he would do. He suggested I read a paper that described how to chemically attack silicon with a mixture of nitric acid and hydrofluoric acid. Therefore I decided to experiment with a *dilution* in deionized water of different proportions of the two acids. By trial and error, I found the best ratios to give me an optimal differential etching rate between silicon and silicon dioxide, while achieving a negligible undercut under the photoresist. 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. Fortunately the etching of my right shoe ended just before reaching my skin.

In parallel with that task I designed a test pattern, called XTPG, containing all kinds of structures suitable to determining the various parameters of the MOS transistors; the resistivity of various thin films, the contact resistance of various contact geometries, and so on. The last step was to define and calculate all the detailed processing conditions to complete the "run sheet." The run sheet was the list of materials, operations, equipment, and conditions that accompanied a batch of wafers, called a *run*, going through the wafer fab. It described in a step-by-step sequence all the operations to be performed on a batch of wafers, from the very beginning to the end of the process.

By April, 1968 I was able to fabricate the first working MOS transistors with silicon gate, suitable for making integrated circuits, going through the entire silicon gate process. Since the characteristics of these first devices were quite promising, it was now time to design the first integrated circuit using the new self-aligned process, and compare it with a similar production metal-gate device. Vadasz asked the production people what was their most difficult IC to produce. They said it was the Fairchild 3705, an 8-bit analog multiplexer with decoding logic – a chip with stringent processing requirements.

The 3705 consisted of 8 very large transistors that had to behave as close as possible to ideal switches. This meant that when a transistor was turned on, it had to have a very low on-resistance; and when it was turned off, it had to have a very low leakage current. The chip also contained a decoder, so that each "switch" could be selected by simply using a 3-bit address. In addition, the switching speed had to be fairly high; which was another difficult requirement to meet. The worst parameter to control, however, was the low leakage current, made worse by the large size of the transistors.

The idea was to design a chip that was functionally identical to the 3705, but used the SGT instead. In this manner, the characteristics of the SGT chip could be easily compared one-for-one to the characteristics of the metal-gate version, facilitating the evaluation of the new technology. The new chip was called 3708 and it was hoped that, if

everything worked well, the 3708 could replace the 3705, given the production difficulties of the latter. The 3708 also provided the platform to further evaluate the SGT and to improve the process, if necessary, during the following months.

It took me a couple of weeks to complete the 3708 design and the composite layout of the chip, with the help of a layout draftsman. Then the laborious process of creating the masks needed to process the wafers started. That entire process took several weeks, and finally, by early July, I had the first wafers of the 3708. When I tested it, it worked immediately, and I could also see that its performance was much superior to the 3705. That made me very happy.

With a fully functioning 3708 (See Fig. 1), I could now start the characterization of the 3708 and also complete the process characterization. This last task was accomplished by having a few XTPG test patterns interspersed in the same masks used for the 3708. Therefore, both process and chip characteristics could be measured. Additionally, the characterization required that special runs be performed by purposefully manipulating certain critical parameters to generate a set of worst-case conditions able to simulate the expected variations in the manufacturing process over the lifetime of the product.

It was during this characterization process that I found to my dismay that the amorphous silicon tended to break at the large oxide steps created by the tub mask. This discovery should not have been surprising since the amorphous silicon was vacuum-evaporated using the same equipment and methods used for the aluminum evaporation, where such problems were already well known. Therefore, the same self-shadowing responsible for aluminum breakage was also at work for the silicon film, except that the situation was worse for silicon.

Fortunately, there was another possibility for creating silicon films at Fairchild. This method used the chemical decomposition of silane (SiH_4), at low pressure and at temperatures ranging from 650^o to 750^oC. This method produced *polycrystalline* silicon rather than amorphous silicon, with grain sizes up to 10nm, dependent on the growth conditions. Because the new deposition method created near-chaotic atomic trajectories, rather than the directional *rays* of aluminum particles responsible for the self-shadowing in the vacuum deposition process, the step coverage was excellent, and the problem was elegantly solved.

Sometime later, I found out that the bipolar group had developed a process to soak up, so to speak, and eliminate much of the highly mobile impurities present in the wafer prior to the contact mask and metal deposition. This process consisted in the deposition of a layer of vapox, followed by heavy doping with phosphorous of both the topside oxide and the bottom side silicon, followed by a thermal treatment at temperatures between 800^o and 900^oC. Under those conditions, the phosphorous acted like a *sponge,* locally segregating impurities which would then diffuse from the wafers toward the phosphorous layers and be segregated there. The impurities could then be eliminated by etching the highly doped layer of silicon in the backside of the wafer and the doped layer on the topside oxide. This process was called phosphorous gettering and resulted in much lower junction leakage currents and higher long-term reliability.

Phosphorous gettering, however, could not be used with the metal-gate MOS process because the aluminum would short the junctions at temperatures above 600^oC. Therefore, the junction leakage current of MOS devices was dominated by the effect of the impurities already present in the starting material, plus the impurities accumulated during

the manufacturing process. Since the silicon gate could easily withstand the phosphorous gettering temperature, this process could be easily added to the existing process flow, *after* the MOS transistors had been fabricated, and after they had been sealed in the best possible material. This was silicon dioxide (quartz), thermally grown at 1200°C. The results were outstanding: the junction leakage current was reduced by more than a factor of 100, and the threshold voltage drift that still occasionally plagued MOS devices made with aluminum gates, was eliminated.

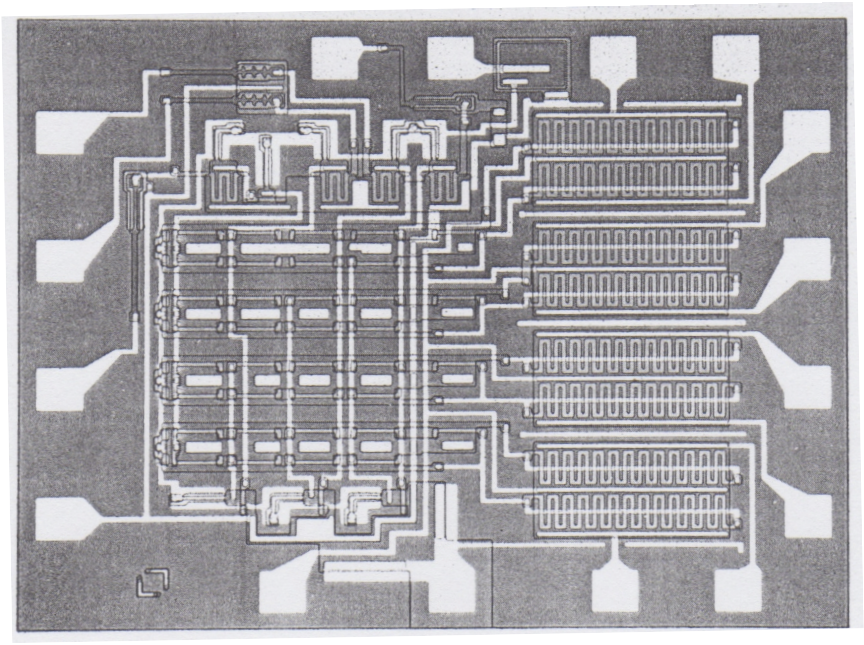

Fig. 1. – The Fairchild 3708, the world's first commercial self-aligned gate MOS IC with SGT, first sold toward the end of 1968.

At last, the MOS ICs long-term reliability achieved the same level enjoyed by bipolar ICs, removing another major obstacle to the broad market adoption of MOS technology. Reducing the leakage current to this low level proved of fundamental importance in the development and acceptance of dynamic random access memories (DRAM), in which the information is stored as electrical charges in the gate capacitances of MOS transistors. In this case, since the gate is inevitably connected to a junction, the leakage current of that junction will discharge the electrical charge that represents the informational bit.

After the characterization was completed, the silicon gate technology had achieved impressive results: Compared with the 3705, the 3708 was 5 times faster, it had about 100 times less leakage current, and the on resistance of the analog switches was 3 times lower [4], [5], [6]. The Fairchild marketing department decided to sell the 3708 chips that were produced in the R&D Lab, after their reliability was proven by the production QA engineers. The first sale occurred toward the end of 1968, and continued with R&D wafers until the process was transferred to the MOS division in Mountain View. In the interim, I did provide the wafers to the production department.

Remarkably, the cost of setting up a fab in the 1960's was a very small fraction of what it cost today. Therefore, in the R&D division there were several independent fabs. One of them was the fab of the digital integrated electronics department, the department headed by Bob Seeds, to which our MOS group belonged.

4.5 New Developments

June, 1968 was a memorable month. First, the SGT development was progressing very well. Then, in the middle of June, I was told that my project had been selected for presentation at the upcoming IEDM Conference to be held in Washington D.C. the following October, and I gave a dry-run presentation to Gordon Moore, the head of the R&D Laboratory. This would have been my first public presentation to an international conference, and I was quite happy that my work had been chosen. But I was also surprised that management wanted to disclose this project so soon, before it was completed. I was 26 year old, still learning to be proficient in English and trying to figure out the American ways; I reckoned that this was the way it's done in America. That same month Fairchild also decided to sell their interest in SGS-Fairchild, and I was offered a job at the R&D Lab to continue my work, which I gladly accepted since by that time I had no desire to return to Italy. Elvia and I both felt that life in Silicon Valley was far more exciting than in Agrate Brianza!

My official starting date as an employee of Fairchild, was July 1st. And that day the laboratory was abuzz with the rumor that Robert Noyce, the head of Fairchild Semiconductor, and Gordon Moore who reported to him, were leaving Fairchild to start a new company. Soon after Noyce and Moore left, many other Fairchild employees quit to join the new company. Andy Grove, who was Moore's assistant, left one week after his boss, and Vadasz, a friend of Grove, left within two weeks after Grove's departure. The name of the new company was Intel. I suspected immediately that Intel would use the SGT because of the strong interest shown by Vadasz, Grove and Moore in my project. That suspicion was reinforced when, about one month later, Intel also hired the technician who was doing the amorphous silicon evaporation for my experiments.

Tom Klein took over the position vacated by Vadasz, and became my new boss. Klein, Vadasz, and Grove were Hungarian born, and they all had fled Hungary in the 1956 uprising, though they didn't know each other in Hungary. Klein made no secret to me that he expected to be asked to join Intel sometime soon. But this never happened.

4.6 Designing Circuits with the SGT

Toward the end of 1968, Fairchild's management made the decision to transfer the SGT process to the production fab that belonged to the MOS operating division. My job was essentially done, though I had to assist the production team to explain the fine points if necessary, and in case of problems. I was also writing a voluminous report on the SGT that was part of the internal Technical Report Series so that other departments within Fairchild R&D could use the process. I was expecting that the production engineers and the professional chip designers of the MOS division would be excited to have such a superior technology. But that was not the case. Despite the success of the 3708, the division was resisting the adoption of the SGT. My contact in the division was telling me that the design engineers were complaining that the circuit layout with SGT used more area than metal gate did.

This was my first encounter with what is called the *NIH syndrome*. NIH stands for not-invented-here, meaning that often engineers refuse to consider worthwhile anything that isn't invented in their own group – even if it is invented in the same company. It is a typical malaise of large companies, a prevalent cultural problem that Fairchild had, even if its size was not yet very large. In fact, at Fairchild there was real animosity between the R&D people and the operating divisions, with finger-pointing in both directions. The division guys complained that they were paying the bills while the haughty R&D guys thought they were so much smarter than them, and only cared about their own theories and projects. For their part, the R&D guys thought that the division guys had no vision and only cared about getting their immediate problem solved.

I began to understand why people leave companies to start new businesses, and why startup companies are necessary to bring new ideas into the world. I realized, if only faintly at that time, that if a company doesn't have a strong culture of innovation, it will resist new ideas because new ideas mean *change*, and change always brings new risks, requires extra effort, and opposes *routine,* which is what many people want.

When I got the feedback that the layout with SGT took more area than metal gate, I asked to be shown the trial layouts done by the division chip designers. I was flabbergasted when I saw what the engineers had produced: they had automatically translated into silicon-gate the old aluminum-gate circuit topologies, without the necessary rethinking, given the substantial differences between the two technologies. In other words, their layouts were done by following the same *implicit rules and style* they used with metal gate technology, when SGT required instead a different style. No wonder they found no advantage with SGT!

Those poor results demonstrated to me that I should not have taken for granted that engineers would understand how to take advantage of the new technology without being given examples of how to do it. I was so close to my creation that for me it was obvious what needed to be done; but not to them. Therefore, I showed them how I would layout their same circuits, proving my contention that SGT was much denser than metal gate, particularly if buried contacts were used.

I didn't hear much else for a while, until I was told that the SGT was not good because it didn't allow the fabrication of isolated capacitors. When I asked why they needed those capacitors, I was told that their main application was to make bootstrap load devices. True, SGT didn't allow making isolated capacitors for the simple reason that the polysilicon over thin oxide would block the formation of a junction underneath. That was in fact the entire point of the self-aligned gate. Their objection was valid, and I fully understood the implications, even though I personally believed that the advantages of the SGT were big enough to warrant the cost of an additional masking step. But that was not the conclusion of the managers. Finally, I got to the bottom of why the design engineers resisted using the SGT.

The importance of the so-called *bootstrap load* was to allow a logic gate to achieve an output voltage swing equal to the supply voltage, V_{DD}, rather than ($V_{DD} - V_t$), which is the maximum output voltage produced by a conventional MOS transistor load with threshold voltage V_t. Bootstrap loads empowered a very efficient design technique widely used in those days, called two-phase-clock dynamic logic. This technique was also called quasi-static design because it allowed mixing static with dynamic logic circuits.

Now, to make quasi static circuits it was necessary to use pass transistors, i.e.

transistors connected to the gate of other transistors. This allowed the temporary storage of information in the gate capacitance of the transistor driven by the pass transistor in the form of an electrical charge. This capability dramatically reduced the number of transistors needed for random logic circuits, compared to a fully static design. It was also a key reason why MOS technology required far fewer transistors than bipolar technology to realize logic functions. For example, with a pass transistor, a dynamic D flip-flop could be made with three transistors, while a *static* D flip-flop required 10 transistors. A D flip-flop made with bipolar technology required more than 15 transistors.

When $(V_{DD} - V_t)$ is applied to the gate of a pass transistor, however, the maximum voltage on the gate connected to the pass transistor is one *additional* threshold voltage drop below $(V_{DD} - V_t)$. Unfortunately, this new signal is generally not sufficient, in worst-case conditions, to turn on that transistor. And this was why the bootstrap load was indispensable to the realization of complex dynamic logic ICs: it allowed a logic gate to drive the gate of a pass transistor with a voltage equal to V_{DD} when that gate had a bootstrap load rather than a conventional load.[5] Without bootstrap loads, it was impossible to design the complex logic circuits where the output of a logic gate could drive pass transistors, although two-phase-clock shift registers or other simple logic gates could be made, as long as the clock voltage swing was equal to V_{DD}.

The other indispensable circuit where bootstrap loads were required was the push-pull buffer. Push-pull buffers were necessary whenever a logic gate had to drive a large capacitive load without dissipating an inordinate amount of power (the data-bus drivers had to drive several hundred pF of capacitive load). And this was absolutely necessary in any complex random logic circuit. The bottom line was that to make complex MOS random logic circuits in 1970 there were only three practical approaches: (1) using 2-phase dynamic logic circuits with bootstrap loads for push-pull and logic; (2) using 4-phase dynamic logic circuits with bootstrap loads for push-pull; and (3) using fully static circuits, which required many more transistors and much more power dissipation than dynamic circuits, for any given speed.

Without bootstrap loads, the world's first microprocessor, the Intel 4004, would not have been feasible in 1970 because it would have required a fully static design, resulting in a chip too large to be made with acceptable yield, and too slow to be useful, unless the allowed power dissipation was much greater than available commercial packages could handle. For example, military applications could use proprietary packages with high

---

([5]) To fully understand the situation, one needs to know the so-called *body-effect*, where the body is the common substrate of the chip whose electrical potential is shared by all the transistors in the IC. The threshold voltage of a transistor whose source is at a different potential with respect to the body, or substrate, is augmented by an amount proportional to the square root of the source-to-body voltage, V_{SB}. In other words, V_{t1}, is function of V_{SB}, which is also the output voltage of the logic gate, V_{O1}. This is the body-effect. Therefore, $V_{t1} = V_{T1} + k\sqrt{V_{SB}} = V_{T1} + k\sqrt{V_{O1}}$, where V_{T1} is the threshold voltage of the load transistor with $V_{SB} = 0$; and k is the proportionality constant of the body effect. When a logic gate drives the silicon gate of a pass transistor, the maximum voltage at the output (the source) of the pass transistor is $V_{O2} = V_{DD} - (V_{T1} + k\sqrt{V_{O1}}) - (V_{T2} + k\sqrt{V_{O2}})$, where V_{T2} is the threshold voltage of the pass transistor with its $V_{SB} = 0$. Therefore, V_{O2}, under worst case conditions, can be insufficient to drive the transistor connected with the pass transistor. However, if the output of the logic gate is V_{DD}, then $V_{O2} = V_{DD} - (V_{T2} + k\sqrt{V_{O2}})$, which is exactly like the output voltage of any other logic gate.

power dissipation and large chip size, which meant very high cost due to low yield; prohibitive conditions for non-military applications.

When I heard about the bootstrap load, I couldn't let that difficulty thwart the unconditional use of the SGT. Therefore I kept on thinking about how I could solve this thorny problem. It took me about nine months to figure out how to design bootstrap loads without adding another mask. The solution, when I founded it, was actually very simple. This of course is very typical of most engineering problems.

Then one day I noticed that the metal electrode of the capacitor in a metal-gate bootstrap load was always biased in such a way that it would create an *inversion layer* in the silicon below *even if there was no diffusion* under the gate oxide. In other words, the operating conditions of the bootstrap load were such that there was a "virtual" diffusion below the metal electrode at all times. Therefore, if I replaced the metal with a silicon electrode, I would have as good a capacitor as if there was a real diffusion underneath. All I needed to do was to create a large area in the tub mask corresponding to the drain of the bootstrap load transistor, and create a poly-silicon region, *inside* this area so that the inversion layer beneath the poly-silicon would be surrounded by a junction area connected with the drain of the load transistor.

This was another key insight, just like when I got the idea of the tub mask – the missing idea to making self-aligned *isolated* transistors. And it felt exactly the same way: a rush of exultation; the "Aha!" of a deeper comprehension. Something like, "I got it! I cracked the nut!" Finally there was nothing more standing between me and the success of the SGT! I proceeded to design a test chip with different geometries of bootstrap loads to find out how well the idea worked, and also to characterize the circuit. I verified that the bootstrap load worked perfectly, shortly before deciding to leave Fairchild. Now my job had been done to the very end.

After Vadasz left Fairchild to join Intel, I also fabricated and tested the buried contact, the invention that Vadasz did not approve of. I verified that the idea worked perfectly, and I made a number of test layouts to prove to myself that with buried contacts I could make significantly more compact layouts, particularly for random logic circuits. With the addition of the buried contact and the bootstrap load, the SGT was now in all respects better than the incumbent metal gate technology, allowing a designer to integrate in the same chip size about twice the number of random logic transistors and achieve about 5 times the speed of the incumbent technology – when using two-phase clock designs with the same power dissipation of an equivalent metal-gate design.

The SGT with buried contacts, bootstrap loads and two-phase clocks was also the technology that made the microprocessor feasible in 1970. As mentioned earlier, the only other viable method to realize a microprocessor was to use fully dynamic circuits with four-phase clocks. This was a relatively complex technique, requiring computer-assisted design. It was successfully used by Rockwell Semiconductor and Four-Phase Systems to produce calculator and computer chips respectively. The SGT, however, using simpler two-phase design was also 3 to 5 times faster and had twice the circuit density of the best four-phase random logic designs.

4.7 Patenting the SGT

In October 1968, I presented the SGT at the IEDM Conference in Washington D.C. [4], after Gordon Moore made the decision to disclose the new technology in June 1968, a

few weeks before he left Fairchild to found Intel. The patenting of the SGT was an important matter since Fairchild had one-year grace period to apply for a patent after such disclosure. If not, the invention would become public domain. Tom Klein, my boss, was the person directly responsible for patenting all the inventions made in his group, and he told me that he was pursuing the patent. I personally knew next to nothing in matters of US patent law, and I entrusted Klein with that task. In particular, I wasn't aware of the existence of the grace period.

Klein delayed the application of the SGT patent until the one-year grace period had lapsed, and the most important claim, which was the original process architecture I invented, could no longer be patented. A patent was eventually applied for in October 1970, two years after public disclosure, and 6 months after I had left Fairchild to join Intel. This patent (No. 3,673,471) was granted on June 27, 1972 with inventors Thomas Klein and Federico Faggin. It contained only a single claim: the use of poly-silicon in self-aligned silicon gates obtained from thermal decomposition of silane in a hydrogen atmosphere. This idea was originally suggested by Klein (though that process already existed at Fairchild), thus justifying his name being first. The embodiment of this invention was of course the process I invented, but could no longer be claimed.

Klein also decided not to patent the buried contact, and another important invention of mine that was later independently invented by someone else and became widely used in the industry under the name LOCOS. This last invention consisted in starting the silicon gate process by first growing a very thin silicon dioxide layer, followed by the deposition of a layer of silicon nitride. The nitride was then removed on the field, leaving it only in the tub areas (using the "negative" of the tub mask used in the normal SGT process). The field oxide would then be grown. Since the nitride doesn't let oxygen through, the silicon dioxide would only grow into the field consuming the silicon in the field areas, and thus moving its inner boundary almost 50% below the original surface. In this manner, the field-oxide height with respect to the tub surface where the active circuits would later be built, was about half of the total oxide thickness, completely eliminating any possible breakage of aluminum at oxide steps.

The buried contact was first used in a commercial product in 1970, embodied in the Intel 1103, the first 1024-bit dynamic RAM, and in the Intel 4004, the world's first microprocessor. Leslie Vadasz applied for a patent for the buried contact on December 28, 1970 without even mentioning to me what he was doing, and the patent was granted on October 24, 1972 with number 3,699,646. When I found out about this patent, sometime in 1973 or 1974, I confronted Vadasz, reminding him that I originally disclosed this invention to him in the presence of Tom Klein in early March, 1968.

Vadasz was taken aback and claimed not to remember about it, and said: "Somehow this idea was stuck in my mind. Let's go talk to Andy Grove," and didn't say another word. Vadasz let Grove do the talking, and Grove tried to convince me that this sort of thing happens all the time in the industry, and it wasn't such a big deal. I did not like the way this incident was handled, and certainly this was one of the reasons that contributed to my leaving Intel about one year later to start my first company, Zilog, Inc.

In 1997 Robert W. Bower was inducted in the National Inventors Hall of Fame for the invention of the self-aligned gate MOS transistor embodied in patent number 3,472,712 [1]. As I mentioned earlier, his invention was never used, though his process architecture was similar to mine, using aluminum and ion implantation instead of poly-silicon and

conventional thermal doping. Since my process architecture was never patented he eventually became the official "inventor" of the self-aligned silicon gate technology. Vadasz became the "inventor" of the buried contact, and someone else independently invented the LOCOS process a couple of years later. I learned my lesson that patents are important.

4.8 Impact of the SGT

Where Fairchild was slow in adopting the SGT, Intel embraced it and made it the core technology around which to build the company. When Vadasz left Fairchild to join Intel as manager of the MOS design department, he had already seen the 3708 working, and he knew all the details about the SGT. Therefore, Intel started with a major advantage, even if they had not yet seen that the amorphous silicon had to be replaced with poly-silicon.

Many years later, the success of Intel was acknowledged by Gordon Moore, one of its founders, to be largely due to the invention of the SGT. He said that the SGT was difficult enough to make that it took a while for the industry to copy it, and yet it was not so difficult that a startup company couldn't do it. What G. Moore did not acknowledge was that Intel had privileged knowledge of the work done at Fairchild which made it possible for them to develop the SGT in a relatively short time. But for sure they believed in the SGT, and that was the missing ingredient at Fairchild.

Intel presented itself as the inventor of the SGT, claiming that the 1101 was the first IC to use the SGT, and never acknowledged that the Fairchild 3708 was actually the first silicon-gate commercial IC. They claimed that the Fairchild process still had problems of metal breakage, when that had been solved, and never acknowledged my role and the role of Tom Klein as the real inventors. In April, 1970 I joined Intel where I designed the 4000 family of chips that included the Intel 4004, the world's first microprocessor. In addition to the buried contact, I used my most recent invention, the bootstrap load that Intel did not yet know about. Both inventions were *indispensable* to design a random logic circuit with the complexity of the 4004 with the necessary speed, power dissipation and chip size to be commercially viable.

When I told Intel's Dov Frohman, who had left Fairchild one month or so before me, about the bootstrap load, he said with a scolding air: "don't you know that you cannot make a capacitor with silicon gate?!" I explained how it worked and said that I had actually built and verified that it worked at Fairchild. He left without saying a word, and the next day I saw Bob Abbott, the 1103 engineer working for Vadasz, placing bootstrap capacitors on the 1103 layout. I asked what he was doing and he said the Vadasz told him to do so. Vadasz never acknowledged that I invented the bootstrap load. It was like nothing happened. Nonetheless, the buried contact and the bootstrap load were the last two pieces of the puzzle to make the SGT the absolute winner of the competition with metal gate.

Around 1969-1970, the availability of ion implantation equipment made possible low threshold voltage, *metal gate* MOS technology, allowing metal gate MOS to narrow the speed gap with SGT, by a factor of 2. But the SGT still held all the other advantages. The 3708 was the cover story in the September, 1969 issue of Electronics magazine [5], and in 1970 the SGT was featured in a scientific article in the journal Solid-State Electronics [6]. Intel also featured the 1101 in a cover article on IEEE Spectrum magazine in 1970.

With the SGT, the silicon gate was entirely surrounded by top quality thermal oxide,

one of the best insulators known, making possible the creation of new device types, not feasible with metal-gate technology. For example, in 1969-1970, Dov Frohman at Fairchild was experimenting with *floating-gate* devices made with poly-silicon (MOS transistors whose gates are not connected to anything) to make non-volatile memory devices. He joined Intel in 1970 where he developed the first electrically programmable and UV erasable read only memory, EEPROM, opening the way to a vast class of non-volatile memories that include flash memories and many other types.

Another major invention made practical by the SGT was charge coupled devices (CCD). Originally invented at Bell Labs, CCDs could be successfully manufactured in 1970 at Fairchild with the SGT to produce the first solid-state image sensors that revolutionized the entire field of photography. These new classes of devices dramatically enlarged the range of functions that could be made with solid state electronics.

By the mid-seventies, the SGT had been adopted by the entire industry, replacing the metal gate technology and eventually allowing MOS technology to also replace the incumbent bipolar technology, still dominant in the early 1970's, for nearly all ICs produced in the world. Only recently the industry has been forced to use materials other than silicon dioxide and poly-silicon for the gate stack, in order to continue the scaling of MOS transistor sizes at or below 45 nm lithography, without too much loss of performance (due to the low dielectric constant of the silicon dioxide). Nonetheless, SGT remains one of the most influential technologies that have fueled the stunning progress of microelectronics during the last 50 years.

5. – The invention of the first microprocessor

5.1 Joining Intel Corporation

Toward the end of 1969, Fairchild had become a slow-moving company, crippled by its own success and especially by the defection of many key executives and engineers to Intel and to several other startup companies. I was frustrated by the slow adoption of the silicon gate technology by the Fairchild MOS division and when Intel announced its first silicon gate MOS product in the fall of 1969 – a 256-bit static RAM (the Intel 1101) – I felt angry and resentful at both Fairchild and Intel, and for very different reasons. By that time the Intel mission had become clear as well: They wanted to become the leading company in the emergent market for semiconductor memories, and particularly for random access memories (RAM), to replace the magnetic core memory that was universally used as the main memory of all mainframe computers and minicomputers.

Magnetic core memories had been in use since the mid-Fifties and were well suited for relatively large storage systems. However, for lesser RAM systems, the large fixed overhead cost (independent of the number of bits) of core memory made their cost per bit prohibitively large. This alternative market was growing rapidly and Intel intended to also sell RAM memory chips to those customers for which the only practical solution was serial memory made with MOS shift registers. MOS shift registers, for many years, had been the only viable method to make small read-write memories. However, they were a barely acceptable solution only for those applications, like calculators and display terminals, where the data flow was naturally serial. For all other applications, RAM was a far superior choice.

My resentment led me to consider leaving Fairchild. Contributing to this idea was also the desire to become an LSI chip designer using SGT, the very technology that I felt would empower this new trend. I confess that a part of me wanted to demonstrate that SGT was truly a superior technology, despite what the Fairchild design engineers had been saying. I wanted to be able to experience a kind of liberating "See? I told you so!" I also felt that chip design would become the new frontier of microelectronics, since the industry was at the threshold of LSI (large scale integration) – chips containing more than 1000 gates. Finally, I wanted to return to my first love, building sophisticated computer systems, but this time in a chip instead of in a rack of printed circuit boards like I did at Olivetti nine years earlier.

A few months later, I was approached by National Semiconductor and I was interviewed by Pierre Lamond, its executive vice president. Lamond wanted me to develop the silicon gate technology for them, and they offered me a big raise over what I was earning at Fairchild. But I wanted to design complex integrated circuits, not repeat the work I had already done. Furthermore, it didn't seem ethical to me to transfer to National the SGT I had developed at Fairchild, even if Fairchild did not seem to take advantage of my work. That offer convinced me that I had to leave Fairchild, however, and gave me the incentive to look elsewhere. I picked up the phone and called Les Vadasz, asking him for a design job at Intel.

So, in April, 1970, less than one month after our first daughter was born, I joined Intel working for my very first boss in the US, Les Vadasz, who now was heading the Intel MOS Design department, reporting to Andy Grove. During my interview process, Vadasz had been very vague and secretive about the project I was supposed to lead, but he assured me that it would satisfy my hunger for a challenging chip-design project.

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." He told me the story of how it evolved from a Busicom proposal of a "dozen" custom LSI chips, to an Intel proposal, spearheaded by Ted Hoff, of a set of four chips where a general-purpose CPU, a ROM chip, a RAM chip and an I/O chip would allow to build a general purpose computer to solve Busicom's problem: making a desktop printing calculator.

Mazor also gave me the basic specifications of the four chips, developed over a period of a few months between Intel's Hoff and Mazor, and Busicom engineers, among them Masatoshi Shima, who was the calculator lead engineer. Mazor also told me with a nervous smile that Shima was going to arrive in a few days to check on the progress, expecting to find the logic design of the CPU completed, and the other three chips in an advanced state of design. The problem was that no work had been done on the project since late 1969, and Busicom was not told about it…

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. Furthermore, my boss was consumed with the key project going on at that time, the 1103, and made it clear to me that he had little time for me.

The Intel 1103 was to be the first 1024-bit MOS dynamic RAM, the product that was intended to make Intel successful in the semiconductor memory market. This was all the more important given the lukewarm market response to the 1101 and to the 3101 (64-bit static bipolar RAM). Both Vadasz and Grove considered my project a diversion dreamed up by the marketing guys, with the tacit approval of Bob Noyce, then the CEO of Intel, to get some revenues while waiting for the memory business to mature. The only products that had any market traction were a family of dynamic shift registers that were a second-source to a family of metal-gate MOS shift registers designed and produced by National Semiconductor. Thanks to the SGT, the Intel shift registers had superior characteristics to National chips, allowing Intel to successfully compete with them. Now I also understood why National was eager to adopt the SGT, and had given me such a generous offer a few months before. Incidentally, Tom Klein left Fairchild after I did, sometime in 1970, to join National Semiconductor where he did what National wanted me to do, i.e. he "transferred" to them the SGT developed at Fairchild.

It was clear to me that the schedule for the Busicom project had been put together without much understanding of what was involved in making these chips, particularly for the CPU, for which a layout time of 7 weeks, only two weeks more than a simple memory chip, had been forecasted by Vadasz. A memory chip is a repetitive design (each memory cell is identical) whose layout is substantially faster to plan and draw than random logic where almost every circuit is unique and has to be custom designed and fitted. Therefore, not only was the project starting about 5 months later than promised to the customer, but also the duration of each project phase had been underestimated, particularly for the CPU.

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.

Given the absence of any random logic design methodology at Intel, my knowledge of the silicon gate technology gave me the opportunity to develop a new methodology for random logic that could take advantage of the strengths, and avoid the weakness of the SGT. That methodology was indeed very successful and was used for all the early Intel and Zilog microprocessors – the company that I founded in late 1974.

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.

5.2 The 4000 family takes shape

Designing a production integrated circuit required many steps, starting with the definition of the chip architecture and its basic specifications. In this case, the first step had been led by Ted Hoff, the head of the Application Research department, with the assistance of Stan Mazor and the Busicom team. The task of the Application group was finished with the completion of the specifications. The actual design and development of the chips, however, was done in another department, the MOS Design department, and from that point on was entirely led by me, without any further contributions by Hoff and Mazor, who were not chip designers.

The design and development steps followed the sequence: logic design; circuit design; composite layout design; ruby-cutting; mask generation; wafer processing – these last two steps were typically done outside the design group – first silicon; chip verification, debugging and characterization; production test-pattern development; and transfer to manufacturing. Generally this process, starting from chip specifications to first silicon, would take at least six to nine months. From first silicon to transfer-to-production – at which point the responsibility for the product would move from the MOS Design department to Manufacturing – it 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.

My job was to lead the design and development of the four Busicom chips, and take them all the way to the transfer to manufacturing. However, the lack of infrastructure meant that I needed to carry out many more tasks than a typical project engineer working for a company already in the custom chip business had to do. In particular, the logic design of a typical custom chip was normally done and verified by the customer, and the task of the chip designer was then to translate that logic design into silicon.

In my case, 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 first, followed by the 4003, 4002, 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. This sequence would also allow Intel to regain Busicom's confidence by showing early success with the simpler chips working at first silicon.

The 4001 was a state-of-the-art 2048-bit metal mask programmable ROM with four metal-mask programmable I/O lines. I did the logic and circuit design of the 4001 in two-three weeks and gave it to Shima to check. Shima was an excellent logic designer and was also the engineer that had to develop the firmware of the Busicom desktop calculator – the first intended application of the 4000 family. Just like Hoff and Mazor, Shima was not a chip designer and didn't know much about MOS technology, but he was eager to learn and was very detail-oriented – a terrific quality given the lack of verification tools at Intel.

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 flip-flop 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. I came up with a circuit which I later patented for Intel [7].

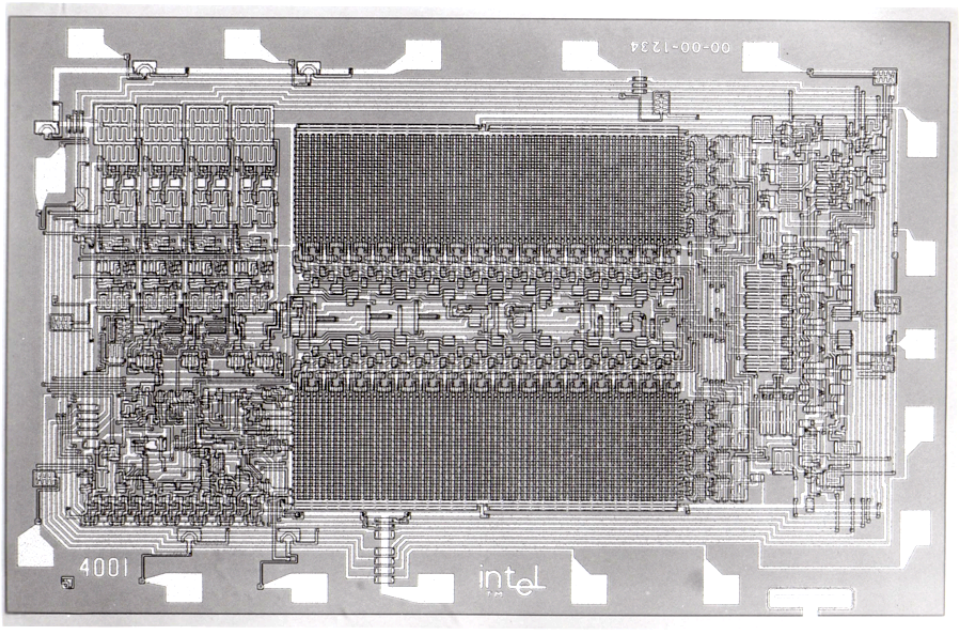

Fig. 2. – The Intel 4001. This chip was a 2048-bit, metal-mask programmable ROM (read only memory), used to store the computer program. The chip also contained a section of metal-mask-programmable logic for its four programmable I/O lines.

The 4001 layout started the day my first layout draftsman, Rod Sayre, showed up for work. He was hired from Lockheed where he was a mechanical draftsman, and he didn't even know what a chip was, never mind having laid out one… In those days, layout draftsmen were harder to find than engineers, and all the experienced Intel draftsmen were busy with memory projects and I could not use any of them. I trained Rod, and in time he became a very good draftsman, but at the beginning I had to draw myself all the building blocks free hand and Rod would copy them properly in the composite layout.

After the 4001 layout was completed (see Fig. 2), Rod laid out the 4003, which was the only really simple chip of the 4000 family, and only took 2-3 weeks to layout. The 4003

was a 10-bit static shift register with serial input, serial output and gated parallel outputs (see Fig. 3). For its design I used a novel static flip-flop that I had co-invented and patented in Italy while working for SGS; a design that had been successfully used in my second commercial chip design at SGS [3]. This same circuit was also used for many of the counters in the 4002 and 4004 because it reduced substantially the transistor count needed for static counters.

The next chip to be started was the 4002, the data RAM of the family. The 4002 was organized as four registers of 16+4 nibbles each (1 nibble is equal to 4 bits), for a total of 320 bits. Furthermore, it had a 4-bit addressable output port. Again, I did the logic and circuit design in a couple of weeks and Shima checked my work. It was good to have somebody else check my work given that there was no time to do any logic or circuit simulation, and Shima was very thorough.

In the 4002 design I used a three-transistor dynamic RAM cell, similar to the one that was being used in the 1103. The chip included also a fair amount of logic for the memory refresh, the decoding of some instructions, timing and control circuitry, and for the 4-bit

Fig. 3. – The Intel 4003. This chip was the only MSI (medium scale integration) component of the 4000 family, containing a 10-bit static shift register with serial-in, serial-out and gated parallel outputs to be used for I/O line expansion. The 4003 was quite helpful to expand the number of I/O lines available and for keyboard scanning, given that each 4000 family chip was packaged in a 16-pin DIP package, with only 4 I/O lines available in each 4001 and 4002 chips.

output register. The layout of the 4002 was done by a new draftsman just hired from Fairchild, Julie Hendricks, the same draftsman who had laid out the Fairchild 3708 a few years before, when she was a trainee. Fortunately Julie was experienced, though mostly in laying out bipolar ICs, and that experience reduced my workload. When Rod finished the 4003 layout, he joined Julie to help speed up the 4002 layout (see Fig. 4).

Finally I could start the logic design of the 4004, though I was slowed down considerably by having to keep the other 3 chips moving, all at different stages of the design process. I also needed the debugging and characterization tester in a couple of

months when I expected to receive the first silicon of the 4001. Fortunately Hal Feeney, a design engineer, and Paul Metrovich, an electronic technician, were assigned to me to help with the design and construction of such tester. We started with a discarded memory system and we built a programmable pattern generator by adding new electronics, and a paper tape reader. We also designed adjustable pin electronics. The entire contraption was ready only days before I received the first 4001 wafers.

After I designed a good portion of the logic of the 4004, Shima offered to complete the logic design, particularly the control section of the CPU. At this point I felt very comfortable that he could do that task after the learning he acquired by assisting me with the checking of the prior three chips. By now I had perfected the methodology, and especially the method of combining logic and circuit design in a single document that also contained the notion of how the chip would have to be laid out. This method avoided the potential translation errors when going from the logic diagram to the circuit diagram. One could focus on the critical circuits, estimating the layout capacitances, and thus the sizing of the transistors, based on the same document. It also speeded up the layout by streamlining the translation from circuit design to layout, again reducing the potential for mistakes.

Fig. 4. The Intel 4002. This chip was a 320-bit dynamic RAM (random access memory) used to store the data for the computer. It also contained its own memory refresh circuitry and four addressable output lines with relative control logic.

Pressed for time, I had to start and closely supervise the 4004 layout before the logic design was completed. Therefore, I was coordinating with Shima so that I could keep the draftsmen busy and achieve an excellent layout density, despite the fact that the design

was in progress. It was like “fast-tracking” a building, i.e. starting the building before the plan was completed. For a chip that was at the limit of what could be economically produced, I could not afford to waste any precious silicon real estate (see Fig. 5). Joining Julie and Rod in the 4004 layout team was Barbara Manness, an experienced memory layout draftsman who had been at Intel nearly from the beginning.

However, no Intel draftsman had ever laid out a complex random logic chip before, requiring a close supervision and coordination on my part. In particular, since each draftsman had their own drafting table, and each was working on a separate sheet of mylar (using colored lead pencils), it was quite a challenge for me to maintain a good sense of how the total chip would come together when the three separate pieces would be merged. The 4004 layout lasted about 14 weeks, for a total of 42 man-weeks, compared with the 5 man weeks of the 4001 layout. The original 4004 schedule prepared by Vadasz had predicted 7 weeks with two draftsmen, for a total of 14 man-weeks.

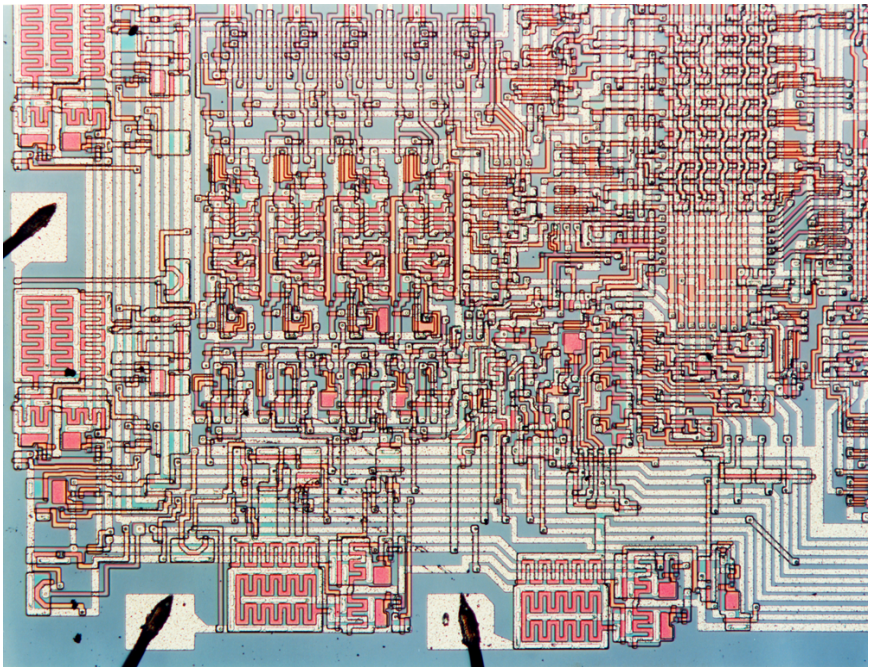

Fig. 5. – This image shows a portion of the 4004 chip layout with two large data bus drivers at the bottom of the image (the large MOS transistors with the orange wavy lines). The external 4-bit data bus was the main highway connecting all the chips together. For a system with many 4001’s and 4002’s, the capacitance of each data-bus line could be several hundred pF, requiring powerful drivers. Moving from right to left on the image, one can see a portion of the control logic of the arithmetic unit, followed by a portion of the 4-bit arithmetic unit. Notice the higher random logic layout density of the 4004 compared with the other three chips of the family since the logic circuits with their local interconnections could be tucked under the more global signal lines. The 4004 was the only one of the four chips to use buried contacts.

When the 4004 layout was completed (see Fig. 6), I followed my impulse to sign my initials, F.F., on the metal mask, as an artist autographs his creations. I felt the layout was like a work of art, where each stroke was not only functional and meaningful, but

contributed to an overall esthetic effect. This impulse proved to be very helpful later when Intel tried to disown me of the paternity of the 4004. It was like the smoking gun that could not be erased

5.3 The mask–making process

At this point it may be helpful to describe how the tooling of the masks was accomplished in 1970 since the methods we used were quite crude compared to what is done today. Today an engineer sits in front of the screen of a CAD workstation and plays

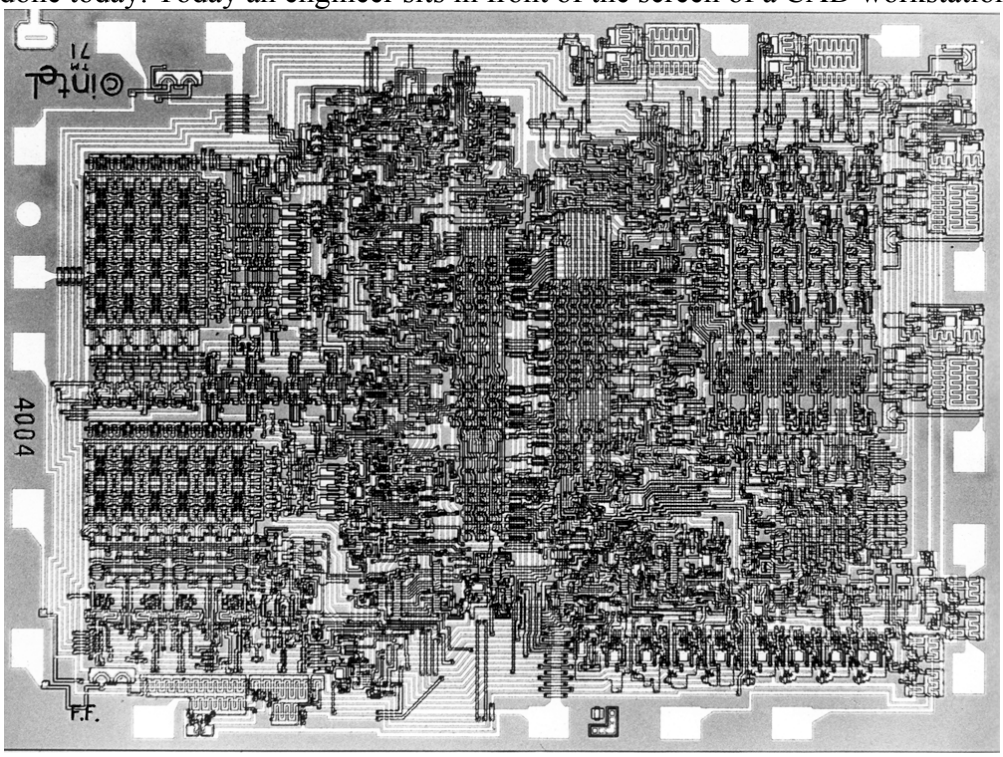

Fig. 6. The Intel 4004, the world's first CPU-on-a-chip. This 4-bit microprocessor contained approximately 2300 random logic transistors. The 4004 basic instruction cycle used 8 clock periods of a 2-phase clock running at 750 kHz, lasting 10.7 microseconds. This time was longer than strictly necessary (by approximately a factor of 2.5) due to the heavy use of multiplexing to send the 12-bit address, the 8-bit instructions and the 4-bit data onto the same 4 data bus lines. The typical power dissipation of the 4004 was 750 mW. Notice my initials (F.F.) in the lower left corner of the chip.

with layout building blocks, assisted by powerful tools. When he is finished with the composite layout, he pushes a button, and everything is done automatically from that point on. Inside the workstation there is a powerful microprocessor. So, how was the first microprocessor created before there were microprocessors?

Like today, the chip composite layout was the key design and tooling document. But unlike today, the composite was drawn by hand in a large, reclining drafting table with a straight edge and colored lead pencils, at 400 to 500 times the actual scale. Instead of paper, we used a quadrille mylar sheet for dimensional stability, since the composite layout was to be later placed underneath the rubylith, and serve as a guide for the cutting

process. The name *composite* was given because all the masking layers of the chip were superimposed in the same drawing. The composite, however, could not be used directly to generate the masks necessary for the manufacturing process. To do so, it was necessary to prepare a separate rubylith layer for each mask.

The rubylith, or ruby, was a transparent mylar sheet covered with a film of semitransparent red material. The ruby was placed on a precision cutting table on top of the composite drawing that served as a guide. Thereafter, the film was cut and peeled to expose the pattern of the mask. The ruby was then photo-reduced into a "reticle," i.e. a 10x version of one of the masks needed to make the chip, and the reticle was then further photo-reduced to real size and "stepped" into the "master plate".

In other words, the red film on the mylar was cut and peeled off with tweezers in correspondence with the areas to be etched on the chip, thus producing a rubylith of the same size of the composite drawing, but showing only one of its layers. The large ruby was then photographed in a gigantic camera and reduced to a black and white reticle at 10x magnification. The reticle was then mounted on a special "step and repeat" camera that reduced the image to actual size, and repeatedly exposed it onto a photographic glass plate until the entire surface of the plate was covered with an array of the same pattern. This process produced the "master," out of which "sub-masters" and then "working plates" were copied by contact photography.

The working plates were then mounted in the mask aligners, the lithographic equipment used to align a mask to the previous layers and expose the photoresist covering the wafer with ultraviolet light. After developing the photoresist, the wafer was ready for the next processing step necessary to produce the silicon wafers – typically etching the material below the photoresist that was not exposed to the UV light because it was protected by a non-transparent area on the mask. The working plates were used only a limited number of times before they were damaged by inevitable small scratches, due to the contact printing process.

Cutting the rubylith was a tedious, long, and error prone operation that required careful and time consuming checking before sending the rubies to the mask making service provider. Since each ruby represented only one layer of the chip, it was necessary to check its integrity and alignment by superimposing the rubies of the other layers, in various combinations, over a large light table. When the ruby-cutting was finished, many people were "recruited" from the lab to help spot potential errors. To complete the checking of the 4004 rubies it took several weeks, and this operation could only be done at the end of the cutting process. Checking the composite layout was far less stressful because it could be partially done during its drawing, and not just at the end.

The ruby-cutting of the 4004 was a challenge because the entire composite was larger than the cutting table and had to be done in two pieces. Shima and I carried the brunt of this process – this was the most complex chip ever done at Intel – but Hal Feeney and others also cheerfully joined forces to help speed up the process.

Sometime in October, 1970, after the 4004 ruby checking was finished, Shima returned to Japan with a brief detour to Egypt for a well-deserved vacation. I continued working 70-80 hours a week for many more months before I could take a brief break. Given how busy I was on this project, Elvia turned to her Italian family for support with our three months old daughter. She went to Italy where she stayed for several months, allowing me to work very hard without feeling too guilty about being missing in action.

5.4 The moment of truth

Shortly after Shima returned to Japan, 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!

After a few days of checking and verification, everything was found to work as expected, not only functionally, but also the clock speed at high temperature, and all the critical signals and the power supply margins were exceeding the design targets. It was a great relief! My methodology passed the litmus test and now I couldn't see any show-stopper for the rest of the family. Busicom was also relieved that their first chip worked as expected.

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.

Elvia had been waiting to hear the news. She woke up from a light sleep as soon as she heard my steps and immediately asked, "How did it go?" Still in a state of excitement I exclaimed, "It works!" And we shared feelings of exhilaration and happiness, knowing that something very significant had happened. That was the night the first microprocessor was born, almost 9 months after I had started the project. I had just turned 29 the month before and I realized that nine years earlier, at about this same time of the year, I had just completed another computer, made with germanium transistors, that had about the same capabilities of this one, except the new one could all fit into a single PC board, instead of a few hundred boards; had about ten times higher speed and consumed almost one thousand times less power. What a difference nine years make!

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. In the meantime, after Shima's return in Japan, Busicom had finished building a 4000 system simulator with a full 4004 simulator and with RAM replacing the 4001's so that the calculator firmware could be conveniently developed, verified, and easily modified, in parallel with the development of the 4004. This was necessary since the 4001 was a metal mask programmable ROM, taking several weeks to fabricate and therefore its use was appropriate only when the firmware was fully debugged.

Soon after Shima heard the news that the 4004 was working, he sent me the four verified ROM codes. We could now fabricate the 4001's in parallel with the correction of the few errors in the 4004. 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.

That engineering prototype was later gifted to me by Yoshio Kojima in recognition for my leadership and contributions to the success of the 4000 family. In the 1990's I gifted the engineering prototype of the world's first product to use a microprocessor to the Computer History Museum, Mountain View, CA, where it is now on display.

5.5 The 4004 Story Revisited

The following account is intended to add some perspective to the history of the microprocessor. It took me many years to paste the pieces together about the background of the "invention" of the microprocessor. I wasn't present at the business discussions between Busicom and Intel, and when the microprocessor became widely known I had already left Intel to found Zilog, a company entirely dedicated to microprocessors. I was therefore declared *persona non grata* by Intel, and my contributions were disowned and attributed to others. Intel attributed to itself and to Hoff many contributions that were instead Busicom's or mine.

When Busicom visited Intel in June 1969, Busicom already wanted to make a family of calculating machines based on the *same chips* they were asking Intel to design for them as custom chips. They went to Intel by following the recommendation of a consultant they retained, Jim Imai, who was familiar with MOS technology and the MOS companies

of that time. Surprisingly, Jim Imai, was the engineer who taught the MOS course I took in 1966 at GMe! Imai told me many years later that the design Busicom wanted exceeded the speed and complexity capabilities of the metal gate technology, but he felt that it could possibly be done with SGT. He correctly told Busicom that only Intel was producing *LSI chips* with SGT at that time, though they were memories. Therefore, Busicom had no alternative when Intel did not want to pursue their design, and also when Intel was 5 months late in starting the project. They could not run to another competitor, as any other customer would have done under those circumstances.

Busicom's design, however, was not just a "dozen custom chips to make a desktop calculator" as it was represented by Intel, but it consisted in 7 custom chips of which 3 chips were intended to be a special purpose CPU, the fourth chip was a shift register read-write memory (SRM), the fifth was a ROM, and the last two were I/O chips to control keyboard, printer, switches, lights, and so on. The CPU used macroinstructions designed to facilitate making calculating machines, and reduce ROM usage. It addressed ROM like any computer does, but it addressed SRM as I/O. The "baroque" way the 4004 addressed RAM in the 4000 family was a vestige of the Busicom design.

Therefore, Busicom's design already included a CPU, though it was not a general-purpose single-chip CPU, and the family of chips was *programmable*. Intel's claim that they came up with a programmable solution to do what a supposedly non-programmable set of custom chips was intended to do is not correct. However, even the Busicom solution was not as revolutionary as it might appear, without knowing a bit more of the history. It was actually Olivetti that introduced the world's first desktop *programmable* calculator in 1965. Conceived by Giangiorgio Perotto, the product was called Programma 101, and it was a small computer packaged as a desktop programmable calculator with a keyboard, printer and a magnetic card reader. It was made with discrete components cleverly packaged in very compact modules, and it had a *serial* read-write memory made with magnetostrictive wires.

That was the *revolutionary* product that started a new trend. It was a very successful product, with more than 40,000 units sold – a large number in those days – and that success drew much competition. From that point on, all high-end calculators were built with a small computer at their heart. Unfortunately, Olivetti sat on its laurels rather than quickly following up with a more advanced product, and Hewlett Packard in 1968 introduced the HP 9100, a desktop programmable calculator much more advanced than the Programma 101, taking over the market from Olivetti.

Busicom therefore, had been following the market trend. But their *technological* vision was much more farsighted than Olivetti because they realized that if they could make a family of calculating machines by *reusing the same building blocks*, they would be ahead of the game. Of course, these building blocks had to construct a computer with various amounts of memory and I/O facilities. In other words, Busicom, saw that, if their CPU was fast enough, they could make several different models of calculating machines using the same set of chips for all of them, by simply employing different amounts of memory and different software (now called firmware). That was a sensible decision because the chip development cost would then be amortized over several products, and the production cost reduced by the *combined* production volume of the *same* set of chips.

Busicom was unaware that Intel was already developing a dynamic RAM that could replace their shift register memory with much advantage (the HP 9100 already used

RAM made with miniature magnetic cores; an engineering feat), and they were looking at Intel like any other MOS supplier making custom circuits. Intel, however, was not a custom MOS supplier, and their business plan was unlike most MOS companies of the day. Intel was *opportunistically* interested in entertaining some custom projects only because they realized that it would take some time before its potential RAM customers would design-in their RAM chips for volume production. Therefore, custom chips could help them ramp up revenues sooner. Intel's vision was more farsighted than their peers.

When Ted Hoff saw the Busicom proposal, he immediately realized that their CPU was much more complicated than it needed to be, given its reliance on shift register memory (SRM). Hoff could simplify the architecture by replacing the SRM and the static RAM (SRAM) used for the CPU internal memory, with the new dynamic RAM (DRAM). This was beneficial because a DRAM-bit used three-transistors while a SRM-bit and a SRAM-bit used 6 transistors. With the encouragement of Bob Noyce, Hoff then simplified the Busicom design by using more traditional computer architecture, made possible by the elimination of the SRM. The result was the 4-chip architecture of the 4000 family that I described earlier. That architecture, however, was no engineering feat because any small general purpose CPU would have resulted in a similar complexity. The clearest example of this statement was the CTC architecture that resulted in the 8008 microprocessor described in the next section.

5.6 The Intel 8008 microprocessor

Toward the end of 1969, Computer Terminal Corporation (CTC, later renamed Datapoint, Corp.) visited Intel with another custom circuit proposal. CTC was already a customer of Intel. They purchased MOS shift registers used for the memory of their computer terminals – a typical use of shift registers at that time. CTC had plans to build a new *intelligent* terminal, called Datapoint 2200, at the heart of which was a simple CPU of their own design. They intended to construct this CPU with bipolar SSI and MSI TTL components, as was typical in those days. CTC wanted Intel to design a special custom *bipolar* RAM chip to be used as the stack register for their CPU.

When Stan Mazor found out the purpose of that custom chip – fresh from his participation to the Busicom project – he ventured to CTC that Intel had the technology to put their entire CPU on a chip, not just the stack memory – if they used RAM instead of shift registers. This was a pretty bold statement since the 4004 had not yet been designed and he was not a chip designer! Eventually CTC was convinced that Intel could integrate their 8-bit CPU into a single MOS chip and they signed a contract with Intel for the development of a custom CPU chip, called the 1201. Hal Feeney was hired to lead the 1201 project just a few weeks before I did. Before Intel, Hal had worked for General Instruments where he had designed a number of custom MOS random logic chips.

I learned about the 1201 shortly after I joined Intel, and I was disappointed that there was another microprocessor in development at Intel. This one was an 8-bit microprocessor, more advanced and with a cleaner, RAM-based architecture than the 4004. I figured that the 1201 would be finished before the 4004, since Hal had to design only one chip while I had four, and the 4004 was going to be my last one. I was so busy with my many challenges, however, that I soon forgot about it. The 1201 project, however, dragged along for several months but never got into high gear and eventually was mothballed; Hal was reassigned to a memory project and then he was assigned to me

to help with the characterization tester. Fundamentally, Hal was overwhelmed by the magnitude of the project; not only by the complexity of the job, but also because he had never designed a chip with silicon gate technology before, and the lack of design methodology and support available at Intel made the task daunting.

In January, 1971, Young Feng, a new engineer, joined my team to help me with the extensive characterization and the transfer to production of the 4000 family, giving me some breathing room. The success of the 4000 family brought another surprise: I was given the job of designing the 1201, supervising Hal Feeney who had been helping me with the 4000 family testing since August, 1970. I inherited the 1201 project after the 4004 was essentially completed and my experience, combined with the now proven methodology, allowed me to lead the project to its successful conclusion with Hal doing the detailed design. The 1201 took the entire 1971 to be designed, with first silicon out in December, and became commercially available in April, 1972 with the name 8008. The 8008 was the world's first 8-bit microprocessor and the "founding father" of the spectacularly successful x86-family of Intel microprocessors that are powering most of the personal computers in use today.

An interesting development occurred in April, 1971 in connection with the 8008. Texas Instruments (TI) announced with much fanfare of having successfully designed a CPU-on-a-chip, as they called it. This announcement was made one or two months *after* the 4004 was fully functional and already sold to Busicom. In other words, TI claimed to have designed what they believed to be the first microprocessor. We later found out that such development started as a custom project for CTC who wanted a second source for their CPU. The specification of this chip that CTC gave to TI was of course *identical* to the 8008, except TI used metal-gate technology for its design. The TI chip size was twice that of the 8008, a size Intel would have considered prohibitive to produce, clearly showing the advantages of the silicon gate technology with buried contacts. I later surmised that the highly competitive nature of the semiconductor business, stirred up by CTC self-interest, convinced TI that, "if Intel can do a CPU on one chip, so can we!"

Many years later I was told by Vic Poor – CTC's VP of Engineering in 1970 – that the TI chip never functioned and of course it was never used. TI also never made that chip available in the market, even after Intel's announcement of the 4004 and the 8008. It was only used for PR purposes. This simple fact serves to prove that the *implementation* of the microprocessor was not a routine design job and was the essential ingredient to be able to claim the paternity of the microprocessor; not the idea, and not the architecture.

If TI, then the leader in MOS custom-chip development, with many powerful design tools and much more experience than Intel in random logic design could not make their first microprocessor work, then Intel did something special! And it wasn't the CPU architecture that was special either. It was the masterful use of a new technology, the SGT with buried contacts and bootstrap loads, that provided the speed and density advantages to make the microprocessor a reality.

If the TI's chip had worked, Intel would still have the microprocessor paternity because the 4004 was first sold in March 1971, prior to TI. In this case execution was the key, given that the 4004 and the TI's 8008 started at about the same time. If the TI's chip did not work (as Vic Poor claimed), the point that execution was the key is even more strongly proven. Since the authors of the 8008 architecture were not Hoff and Mazor,

they could not claim that idea (Busicom, Four-Phase Systems and others had the same idea) and/or architecture were fundamental in the creation of the microprocessor.

At that time, the path to a microprocessor was a race among the best in the business, not a revolutionary idea that occurred to Ted Hoff. The microprocessor was inevitable and it would have happened in time. Its realization needed a sufficiently powerful technology to occur. It was the SGT that had the right features to make it the front runner of the race, and I also contributed another vital ingredient: a flawless execution done in record time under difficult circumstances. The game was about circuit design and dense layout with a technology that had substantially better speed and density potential that metal-gate MOS technology. The essence of the microprocessor was the translation of existing ideas into a single chip of silicon. That was the crucial step that had not been done before.

In 1970, many people knew how to architect simple CPUs, or how to do logic design, and Lee Boysel had already promoted the idea of making a CPU in a few chips. In fact, he had already designed a CPU in a few MOS chips before the 4004, as the founder and CEO of Four-Phase Systems. What remained to be done was: (1) to make the CPU into a single chip so that the many benefits of cost and performance possible only with a single chip implementation would accrue to the user community, and (2) make the microprocessor available to the general market so that the engineering community could take advantage of it, rather than using it only for a captive customer, as was the case for Four-Phase Systems, Busicom, and CTC.

5.7 Announcing the microprocessor to the world

During the design of the 4000 family, I found out that Intel had entered into a contract with Busicom that gave them exclusive rights to use the 4000 family. I was upset because I saw a great market potential for the microprocessor, and I wanted my work to have a much bigger impact than just being a custom job for Busicom. With the project nearing completion, I started lobbying with Intel's management to get out of the exclusivity agreement and sell the 4000 family in the open market. Hoff and Mazor believed that Busicom would at most give up their rights for non-calculator applications, and since they felt that the 4000 family was primarily good for calculator-like applications, they were not initially convinced that selling the 4000 family in the open market for non-calculator applications was a good idea.

They felt, however, that the more general-purpose architecture of the 8008 made it more suitable for general use than the 4000 family, even if introducing the 8008 in the market was also problematic since it was bound by a similar exclusivity arrangement with CTC. They were also concerned about how to market a microprocessor, a product unlike any other general-purpose component that had been sold before by the semiconductor industry.

I thought that there were many *control* applications where the 4000 family would do well, despite its limitations, and I set out to find out for myself. The opportunity came with a new project I needed to start: a wafer-sort tester for the 4004. I decided to use the 4004 to perform the *control logic* for the tester, instead of using random logic as was common practice. I figured this was the best way to find out first hand, whether or not the 4004 was appropriate to the task. I would also gain insights into what a customer would have to do to use the 4000 family to solve his problems. Finally, I was very interested in programming the 4004, a task I had never done before.

Since there was no programming tool for the 4004 and I was pressed for time, I wrote the tester control program using the instruction mnemonics, and then I had to literally translate the program by hand into machine language – using the ones and the zeros that needed to be stored in ROM, the 4001. However, the 4001 required a metal mask to store the program. That could only be done if I had fully debugged my program and if I needed many copies of it. But if I only needed one chip, that process was too costly and too long. Therefore, I decided to use a product that had just been developed by Dov Frohman at Intel: the 1702, the world's first electrically programmable, UV erasable ROM.

The 1702 was intended to aid the development, debugging and prototyping of ROM codes that, after a thorough engineering and customer testing, would be translated into conventional, much lower cost, mask-programmable and pin-compatible ROM chips. The ROM chip Intel was developing was called 1302, intended to replace the 1702 for production, fitting in exactly the same socket previously used by the 1702. All I had to do, then, was to design an appropriate interface between the 4004 and the 1702 to make the 1702 behave like a field-programmable 4001.

The tester project was very successful and convinced me that the 4004 was effective when applied to many control applications like the one I had. I used that experience to lobby with management to broadly market the 4000 family, building a more convincing case for it, particularly with Ed Gelbach, the new Intel VP of marketing. Finally, during a phone conversation with Shima, around the middle of 1971, I found out that Busicom was not doing well in the market and could not compete effectively with their desktop calculators against competitors that had a traditional design based on MOS custom chips; the price they were paying to Intel for the 4000 family chips was too high. Shima also told me that Bob Noyce and Ed Gelbach were going to visit Busicom shortly.

That information gave me the break I needed: I told Bob Noyce about my conversation, suggesting that he might be able to get a release from exclusivity by Busicom, for non-calculator applications, in exchange for a lower price. Of course I also pushed once more the case that the 4004 was very good for control applications based on my experience with the tester project. Shortly after Bob Noyce's visit to Busicom, I learned that he had been successful in negotiating a release from exclusivity and had decided to introduce the 4000 family in the market. I was delighted.

Ed Gelbach soon appointed Hank Smith to lead the marketing effort for the new microprocessor products, and Feeney and I from MOS R&D together with Hoff and Mazor from Application Research, helped the new marketing group prepare the literature and strategy for the 4000 family market launch. Smith coined a new name for the 4000 family: MCS-4, standing for micro computer system 4-bit. The MCS-4 was soon to be followed in early 1972 by the introduction of the MCS-8, with the 8008 as its centerpiece. The rest of the MCS-8 family were primarily standard Intel memories renamed as if they had been intended to be components of the MCS-8 family.

In November, 1971 the official birth announcement of the microprocessor to the world finally happened. A two-page spread in the well-read *Electronic News* magazine proclaimed: "Announcing a new era of integrated electronics," (see Fig. 7) and briefly described the microprocessor and its availability. This turned out to be a prophetic statement, a rare occurrence in advertising, since the impact of the microprocessor on our lives has been truly extraordinary, a claim only a handful of other inventions in the last 100 years can make [8], [9], [10].

5.8 The 8080 Microprocessor

In the late summer of 1971, in view of the imminent public announcement of the microprocessor, I went to Europe with Hank Smith to visit potential microprocessor customers. I described both the MCS-4 family and the 8008 which was in an advanced layout stage at that time. The most interesting aspect of that visit was finding out that the companies in the computer business were quite critical about our products, whereas companies with problems that our chips could help resolve were very receptive and glad about the new possibilities offered by microprocessors. On the other hand, the most useful feedback about improving our products came from the computer companies, such as Nixdorf Computer and ICL (International Computers Ltd.), even though much of their feedback was delivered with some scorn, like saying, "What do you know about computers anyway."

Fig. 7. – This was the first microprocessor advertisement: A two-page spread that appeared in *Electronic News* in November, 1971.

I made treasure of some of those "suggestions" and by the end of 1971 I began to conceive and architect what was to become the 8080, a second-generation 8-bit microprocessor that solved all the known limitations inherent in the 4004 and 8008, and added a bounty of new features. The 8080 was intended to take full advantage of the N-channel SGT process that Intel was developing for the new 4k-bit DRAM. It would also have a new bus architecture and a new interrupt structure, which were major limitations in the 8008. The 8008 was packaged in an 18-pin DIP, and required two-dozen other support chips to interface with memory and I/O. Almost all of those components could be eliminated by adopting a 40-pin package, but Vadasz didn't want to hear about it. It

didn't seem to matter to him that the 8008 was 2-3 times slower than it needed to be only because it was forced to be packaged in 18-pins.

I had to pull all the stops to convince Vadasz to let me use a 40-pin package, since it was indispensable to optimize the performance of the 8080. But Vadasz was resisting. Fortunately at that time I was also developing a single-chip calculator that had to use a 40-pin package. For some reason that was fine with Vadasz because our chip had to compete with all other single chip calculators that were housed in 40-pin packages, but for the 8080, 40-pins were not acceptable somehow. That made no sense to me. I was finally able to get through to Vadasz and have his permission to go with 40 pins. Therefore, in April, 1972 I wrote a proposal to Intel's management describing the 8080 project and soliciting approval. The 8080 was conceived to be machine-code compatible with the 8008 and have many additional instructions and features that would turn a very limited 8008 into a good microprocessor.

Speed was essential, and my goal was to get a minimum instruction cycle of 2 microseconds, six times faster than the 8008, within striking distance of the speed of many contemporary minicomputers. Therefore, many applications that were off limits to the first generation MPs would become possible with the 8080. Despite my urgings, however, it took about 9 months before I was given the green light to start the 8080 project. I hired Shima from Japan to work with me because he had witnessed how to design the 4004, and with some additional training, I felt he could become a good project leader.

The chip design started in November 1972, and after a few months of working closely with Shima, he became independent enough that a weekly meeting and a few short interactions in between meetings were generally sufficient to stay on top of the project. The first silicon of the 8080 came out in December, 1973, and it worked completely, but for a few minor problems. The 8080 was announced and shipped in March, 1974 [11], [12], and Motorola became the first worthy competitor to Intel by introducing their first SGT 8-bit microprocessor, the 40-pin 6800, six months after the 8080.

I always regretted the loss of 9 months of market leadership. I felt that Intel mindlessly squandered a hard-won competitive advantage that had cost me a lot of energy. Fortunately the 8080 was good enough to withstand the competition, even though Intel's lead was irreversibly shortened.

The 8080 was an instant success. With the 8080, the microprocessor had come of age, and the market started expanding rapidly; an exponential expansion that hasn't yet subsided. Today, almost any non-memory chip integrates one or more microprocessors. Surprisingly, even flash memories contain microprocessors. In fact each chip has a 32-bit MP inside just to handle the management of the data stored within.

As Intel grew, I also grew, gradually taking over more responsibility and more projects. Toward the end of 1973 I was given a memory project by Vadasz. He needed my help because Intel could not deliver 500 nsec access-time 2102 RAM chips to Burroughs, a big computer customer. Burroughs needed a large numbers of 2102's, but Intel's yield at 500 nsec was so small that it was left with a large number of unsold, slower chips. Being under pressure, and having nobody available, Vadasz turned to me for help. I was surprised because Vadasz had told me more than once that memory chips were much harder to design than microprocessors, as if the expertise required to design memories was a more refined art than the one required to design microprocessors.

The 2102 was a 1024-bit static RAM that used for the first time a single 5-volt supply voltage, the same supply voltage used by bipolar logic chips. This choice finally made MOS signals compatible with bipolar signals, when historically that had not been the case. However, operating at 5 volt with regular MOS transistor loads was barely possible because the worst-case output voltage of a gate, (*VDD – Vt*), was not much more than the worst-care threshold voltage. This created a large spread in the 2102 speed distribution with process variations, contributing to the current difficulty of meeting the 500 nsec speed-selection.

Vadasz asked me to redesign the 2102 by reducing the gate oxide thickness of the transistors, claiming that such a change would solve the problem. After thinking for a while, it was clear to me that his suggestion would only buy a small improvement. On the other hand, if we used depletion load devices[6], the improvement would be great. Vadasz didn't want to go in the direction I proposed, even if depletion load transistors had been successfully used before. The excuse was that they required another masking step.

We got into a stalemate, and I told Vadasz that I didn't want his project because I had no confidence that it would succeed. If he wanted his way, I said, he should give it to somebody else. He finally agreed to go with depletion loads, and six months later we got the first 2102A chips (the new part number). They were 5 to 8 times faster than the 2102! I even found chips with 80 nsec access time, close to the access time of the much more expensive bipolar RAMs. I told Vadasz that now, with a few refinements, we could go after the lucrative fast bipolar RAM market. This was the start of the process technology that carried the industry for the following 10 years before being supplanted by CMOS (complementary MOS), the ultimate technology that is still in use today. This project launched Intel into the highly profitable fast static RAM business that successfully competed with bipolar memories, eventually taking over one of the last bastions of resistance of bipolar technology.

6. Zilog and the Z80 microprocessor

In early 1974, I was promoted to department manager in charge of all MOS chip designs, except for dynamic memories. My group had about 80 people, mostly engineers, and the microprocessor business was finally expanding, given the success of the 8080. But Intel

---

([6]) An enhancement mode MOS transistor is a MOSFET where the threshold voltage is of the same sign of the supply voltage. A depletion mode MOS transistor is a MOSFET where the threshold voltage is of the *opposite* sign of the supply voltage. For example, with $V_{GS} = 0$ an N-channel enhancement-mode transistor has a positive threshold voltage and there is no current flowing until $V_{GS} > V_T$. An N-channel depletion-mode device has a *negative* threshold voltage and therefore there is current flowing with $V_{GS} = 0$. To turn the device off, V_{GS} has to be made more negative than V_T; i.e. $V_{GS} < V_T$. A depletion load MOS transistor is a depletion-mode transistor with its gate and source connected to the output of a logic gate, and the drain connected to the supply voltage. In this configuration, the depletion load behaves like a bootstrap load, allowing the output voltage to be equal to the supply voltage rather than (*VDD – Vt*). In other words, the current available to charge a load capacitor is initially nearly constant since the transistor starts in its saturation region, eventually decreasing to zero when the output voltage is equal to the supply voltage. For the same power dissipation, the switching speed is 5 to 10 times better than conventional enhancement load devices. See note ([5]).

was still a memory company making microprocessors to sell more memories. Despite my success, I was getting tired of having to fight for anything that I wanted to do. In the summer of 1974 I decided that it was time for me to leave Intel and become my own boss. If Intel didn't want to be a microprocessor company, I would create a company that would.

There were many other reasons to leave Intel, including the ascendant of Andy Grove whose management style didn't agree with mine. When I told Vadasz that I was leaving, he could not convince me to stay. Therefore, he asked me to talk to Grove. Grove was quite cordial at the beginning, but when he saw my resolve to leave, he turned ugly and said that if I left Intel I would leave no legacy to my children and grandchildren, and that, furthermore, I would never be successful! It felt like a curse mixed with blackmail, and that made my resolve to leave even stronger.

I asked Ralph Ungermann, one of my managers, to join me, and by the end of 1974 Zilog was born. Zilog was the first company entirely dedicated to the emergent microprocessor market. The first product I conceived was a powerful single-chip microcontroller, but then I opted for another idea of mine: a third-generation 8-bit microprocessor family, the Z80 family, centered around the Z80-CPU and several other peripheral chips intended to work seamlessly together. The entire family would use the new 5-volt N-channel depletion load process technology that I also used in the 2102A, and all the chips shared a bus and interrupt structure similar to the one used by contemporary minicomputers.

The Z80-CPU was machine-code compatible with the 8080 and had a long list of improvements that included the ability to automatically refresh the main memory, typically realized with dynamic RAM, without any external components. The instruction cycle time of the Z80-CPU was 1 microsecond, at parity with many minicomputers of the day that were built with bipolar technology. Introduced in mid-1976, the Z80-CPU became wildly successful, powering many of the early personal computers and literally thousands of other applications [12], [13]. The Z80 is still in high volume production in 2015, with several billion units sold over its lifetime. My first idea, the chip that I set aside in favor of the Z80 became a couple of years later the Z8 microcontroller. Introduced in 1978, the Z8 was also very successful and is still in production today.

7.0 Conclusions

The Z80-CPU was the last engineering project I directed. It was also the end of my technical career and the beginning of my entrepreneurial career. I became a founder, CEO of a few startup companies, as well as an angel investor in many other startups. I believed then, and I still believe now, that startups are the most effective way to bring new ideas into the world because they *catalyze* the extraordinary passion, energy and focus needed to innovate. When I left Zilog to start another company, I had logged more than 10 years of my life to bringing 4 generations of microprocessor into the world (the last one was the Z8000). I felt it was time for me to do something else. The microprocessor was like a grown up son to me; a son that did very well in the world and could now take care of itself.

I started a company in the personal communication area that I recognized as the new frontier, and the new company, called Cygnet Technologies, Inc., introduced a personal communication system that worked in concert with a PC to improve voice and data communications. I then started Synaptics, Inc. to do research in the field of neural networks and intelligent systems. Eventually Synaptics introduced the early touchpads and touchscreens that have revolutionized the way we interface with our mobile devices.

In 2011 I started Federico and Elvia Faggin Foundation, a non-profit organization dedicated to the scientific study of consciousness. This is another "startup," but this time in the world of ideas rather than in the world of technology. Here the purpose is not to bring a new product or technology into the world, but to help bring fundamental new ideas about the nature of reality [14], [15]. Specifically, the Foundation supports the development of a mathematical theory of reality based on *cognitive* principles rather than materialistic principles.

The information revolution of the last 75 years has begun to highlight the deep and unsuspected relationships between the nature of information and the nature of reality, the nature of the observer, and the nature of life. This is a fascinating subject that has the potential to shape a new worldview with major consequences for human society. The famous physicist John Wheeler in the 1970's coined the phrase "it from bit" to stress his belief that information is primary and matter "derives" from *information*. I believe that life and consciousness can be more profitably seen as informational rather than biochemical or physical phenomena. But I believe that the concept of information we need is far vaster than Shannon information, which only expresses a small part of what I believe information is.

My life has been a long journey of discovery. I feel very fortunate to have been able to contribute to the information revolution, but I also feel that my job is not finished. I now want to understand more clearly the nature of consciousness, the nature of what gives meaning and purpose to our existence, *the nature of experience itself*.

REFERENCES

[1] BOWER, R. W., *Field-effect device with insulated gate*, "US Patent No. 3,472,712," to Hughes Aircraft Co., filed October 27, 1966, issued October 14, 1969

[2] SARACE, J. C., KERWIN, R. E., KLEIN, D. L., and EDWARDS, R., "Solid State Electronics" 1968, Vol. 11, p.653

[3] FAGGIN, F., and CAPOCACCIA, F., *A New Integrated MOS Shift Register*, "Atti XV Congresso Scientifico Internazionale per l'Elettronica," Roma, April 1968, pp.143-152.

[4] FAGGIN, F., KLEIN, T., and VADASZ, L. *Insulated Gate Field Effect Transistor Integrated Circuits with Silicon Gates*, "International Electron Devices Meeting," Washington, D.C., October 1968, p. 22.

[5] FAGGIN, F., and KLEIN, T., *A Faster Generation of MOS Devices with Low Threshold Is Riding the Crest of the New Wave, Silicon Gate IC's*, "Electronics," Sept. 29, 1969.

[6] FAGGIN, F., and KLEIN, T., *Silicon Gate Technology*, "Solid-State Electronics," 1970. Vol. 13, pp. 1125-1144.

[7] FAGGIN, F., *Power supply settable bi-stable circuit,* "US Patent 3,753,011," to Intel Corp., Aug. 14, 1973.

[8] HOFF, M., MAZOR, S., and FAGGIN, F., *Memory System for Multi-Chip Digital Computer*, "US Patent 3,821,715," to Intel Corp., June 28, 1974.
[9] Faggin, F., and Hoff, Jr. M., *Standard Parts and Custom Design Merge in Four-Chip Processor Kit,* "Electronics," Apr. 24, 1972, pp. 112-116.
[10] FAGGIN, F. et al., *The MCS-4 – An LSI Micro Computer System*, "Proc. IEEE Region Six Conference," IEEE, 1972.
[11] SHIMA, M., FAGGIN, F., MAZOR, S., *An N-channel 8-bit single-chip microprocessor,* "IEEE ISSCC," February, 1974, pp. 56-57.
[12] FAGGIN, F., SHIMA, M., MAZOR, S., *MOS computer employing a plurality of separate chips,* "US Patent 4,010,449," to Intel Corp., filed 12/31/1974, granted 3/1/77
[13] SHIMA, M., FAGGIN, F., UNGERMANN, R., *Z80 Chip Set Heralds Third Microprocessor's Generation*, "Electronics," vol. 49, n. 17, 19 August, 1976, pp. 89-93.
[14] ANGELO GALLIPPI, *Federico Faggin – Il padre del microprocessore,* tecniche nuove, Milano, 2011
[15] Fagginfoundation.org