The Dawn of the Computer Age

by Irving S. Reed

Northrop's Snark, armed with a four-megaton thermonuclear warhead, was in production from 1959 to 1961, when the Atlas intercontinental ballistic missile made it obsolete.

More than a half century ago, I was lucky enough to witness perhaps the fastest mathematical mind of the 20th century going head to head with one of the earliest digital computers—a machine that I had helped develop. Like the epic contest between John Henry and the steam drill, the event symbolized the changes of the coming era.

My part in the story began when I was discharged from the Navy after World War II and returned to Caltech—where I had received a BS in mathematics in 1944—to attend graduate school. My wife and I lived in Caltech's grad-student housing in Temple City, and money was tight, forcing us to economize wherever possible. To that end, I usually biked the four and a half miles to the Caltech campus, where as a teaching assistant I taught freshman math. I also graded graduate students' papers on advanced mathematical analysis.

In the spring of 1947, I began to look in earnest for a better-paying summer job. Eric Ackerlind, an amiable, heavyset electrical engineer in charge of what was later to become the Northrop computer group, hired me to work on the company's Project MX-775, the Snark. This was what would now be called a cruise missile—a pilotless subsonic jet airplane designed to fly itself 5,000 miles and deliver a warhead inside a circle 200 yards in diameter. I was hired to help design the navigation system.

The woes of high-tech workers toiling in tiny cubicles have now been the subject of books, movies, and comic strips. At the risk of sounding like the person who claims he walked 10 miles uphill to school—both ways!—I say, at least the present-day personnel have cubicles. On my first day at Northrop, I beheld a large rectangular room, approximately 100 feet by 50 feet, filled with a sea of aircraft draftsmen, row upon row upon row, all working away at their drafting tables. Dr. Ackerlind sat in one corner of the room like a schoolteacher.

Management later moved the Snark navigation group to another room, not quite as enormous as the first, but still without partitions. Among my new colleagues was Floyd Steele [MS '41], with whom I would work intimately in coming years. Others included George Fenn [BS '45, MS '46], Chester Stone [BS '45], and Herman Kahn [MS '47], who later made a name for himself as a physicist at the RAND Corporation before emerging as a world-famous expert on thermonuclear warfare. (He was rumored to have been one of the models for the title character in the film *Dr. Strangelove*.)

The guidance problem was difficult. The Snark had to fly for hours at an elevation of 30,000 to 50,000 feet. It was not just a matter of installing a simple autopilot and entering a compass heading; this is what the Germans had done with their notoriously inaccurate V-1.

The initial thought was that the Snark would find its way by celestial navigation: star trackers would lock onto two or three stars, and by constantly calculating the azimuth and elevation of those stars, a more sophisticated autopilot (the Greek word for "helmsman" is *kybernetes*, the source of the word "cybernetics") would determine the missile's position and orientation and make the steering adjustments needed to stay on course. But upon analyzing the problem, we decided that instead of stargazing, inertial guidance—which required continuously solving a set of differential equations—should to be used to control the servos that guided the missile.

We were vaguely aware that on the East Coast, John Mauchly and J. A. Presper "Pres" Eckert Jr. were building an electronic computing device named ENIAC—but it weighed 30 tons, took up 1,800 square feet of floor space, and could not possibly go airborne.

It would not be easy to create a machine that could solve those equations in real time. We were vaguely aware that on the East Coast, John Mauchly and J. A. Presper "Pres" Eckert Jr. were building an electronic computing device named ENIAC—but it weighed 30 tons, took up 1,800 square feet of floor space, and could not possibly go airborne.

Steele, who worked full-time on the project, instead took his cues from the mechanical differential analyzers that had been developed in the 1930s by Vannevar Bush at MIT, based on 19th-century work by William Thomson, the first Baron Kelvin. These machines used wheels, gears, and cams to create a mechanical analogue of the equations. (A drafting compass can be thought of as a simple analog computer, programmed to solve the equation that describes a circle of a given radius.) The famous Norden bombsight used in World War II was a small, mechanical, very clever Kelvin-inspired differential analyzer.

Steele had a brilliantly simple idea. He would model the actions of the gears and cogs in the mechanism with what he called a DIgital Differential Analyzer, or DIDA. The linkages between the moving parts would be encoded in the arrangement of the wires between digital integrators—assemblies of vacuum tubes that added ones and zeroes.

But it was not clear that DIDA's mathematical representation was in fact a full and accurate statement of the problem—that is, whether its digital shorthand faithfully modeled the differential equations it was designed to solve. I was given the assignment of proving that it did. This was more than a summer project, so when classes resumed, I left Northrop but remained on the payroll as a consultant.

My approach to the problem was heavily influenced by another, quite different, line of "research." At that time, Chester Stone and I and another classmate, Leonard Abrams [BS '44], were attempting to turn our mathematical skills into tangible financial rewards. Leonard was fascinated with probability theory—specifically, he wanted to find the probability that a given horse in a race would finish first. We used a year's worth of the *Racing Form* and a model we developed based on assigning a Gaussian probability distribution to the speeds of each horse in different races.

We performed the intensive calculations by hand, using mechanical calculators, tables, and slide rules. The best solution required running all the horses in the race simultaneously, which was totally out of reach of our computational abilities. But Leonard and I found a simplified method, where we imagined running each horse against what we called a "standard superhorse" that we constructed from the *Racing Form* records.

Then Leonard, sometimes with Chester and me, took his slide rule and notes to the local tracks—Santa Anita or Hollywood Park—and watched the tote board. He'd multiply our calculated probability of winning (say, 0.1) by the payoff shown on the board for a one-dollar bet (which might be $5.00), and when the product was more than a dollar, he'd bet. And he won—not spectacularly, but consistently. Alas, none of us had enough capital to take advantage of the small margin the system gave us, and Leonard failed to convince well-to-do friends to invest in a racetrack system—even though this one was sound and genuinely worked.

But my immersion in probability study paid off: instead of a hard proof that DIDA produced a completely accurate representation of a differential analyzer, I found a soft one that showed that DIDA could be trusted to find the correct answer to within a specified margin of error.

My rising enthusiasm for the possibilities of this electronic digital computer dovetailed neatly with a mathematical logic course I had just completed, taught by my hero and mentor, Eric Temple Bell. I had read Bell's *Men of Mathematics* as a 15-year-old in Fairbanks, Alaska. He had kindly replied to my fan letter, recommending the books he thought I needed to continue my studies, and was in large part responsible for my coming to Caltech.

Bell was tweedy, Scottish, very professorial, and seldom without a large unlit cigar. He lived, with a housekeeper, across the street from Caltech in a modest home that was overgrown with bamboo and weeds, much to the dismay of his house-proud neighbors. He seemed to find time for everything but gardening, including writing science fiction—under the pen name of John Taine—in his gazebo.

As I worked on the DIDA problem, Bell's course and my shipboard experience with radar led me to think about how configurations of switches and relays could represent mathematical logic. Conversations with grad student John Harris [BS '48, MS '49 and '50], who, like me, had spent the war working with electronics (Harris had been in the Signal Corps), led me to the idea of hooking two switches together in series to represent an AND statement, and two switches in parallel to represent an OR statement. I also had the idea of utilizing "gang switches"—switches and relays with several contacts—to simulate *N*-valued logic.

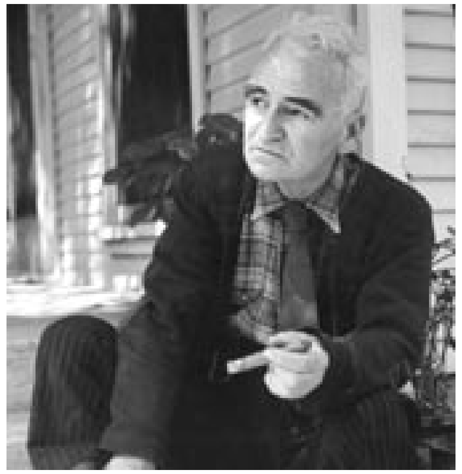
It was a eureka moment for me—a sudden rush of insight: a whole world to open up. Like Archimedes rushing from his bathtub, I raced to Bell's office, confident that I had found a truly original application of mathematical logic.

"I could do a dissertation on logic," I told him, "by making electro-mechanical devices or possibly even electronic computing devices, which would be, in effect, logic machines." I anxiously awaited his affirmation.

"That's a great idea," he said. "But you've been preceded by almost a decade."

He then went on to tell me about an obscure paper, "A Symbolic Analysis of Relay and Switching Circuits," published in the *Transactions of the American Institute of Electrical Engineers* in 1938 by a then-unknown researcher named Claude Shannon, who had been working under Bush at MIT. (The paper was actually a version of Shannon's master's thesis, written in 1937.) Today, Shannon is recognized as one of the seminal mathematical thinkers of the 20th century and the founder of the science of information theory—its premier prize, the Shannon Award (which I much later was privileged to receive), was established in his honor. But at the time he had not yet published "A Mathematical Theory of Communication," and at Caltech he was not a household word. Bell knew of him from a presentation a student had made in his class three years before.

I raced to the library to read the paper, and digested it carefully. I at once recognized its power as a guide for digital computer design. While I was chagrined that someone else had discovered the existence of this exciting new trail, I nonetheless realized that an enormous amount of practical work had to be done for anyone to actually use it. I was in a perfect position to help make this happen.



Professor of Mathematics Eric Temple Bell in a 1953 photograph.

Shannon's paper had been written when the switches and relays to embody his circuits were slow and primitive. But World War II and a decade of intensive, forced-draft research had brought a revolution in electronic hardware. It was now far more possible to actually build a machine based on his ideas, and my Navy experience with practical electronics gave me the requisite technical background.

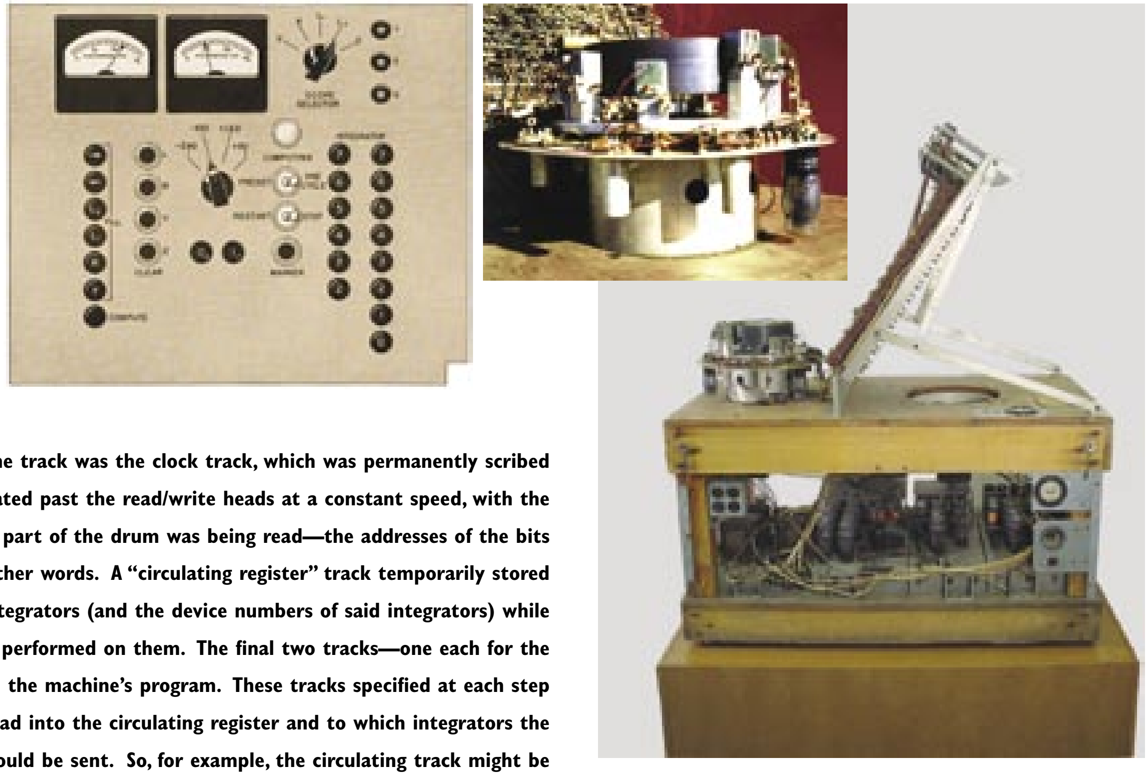
I could hardly wait to share Shannon's paper with Steele. Excitedly, I told him it offered a clear, well-developed intellectual framework that could be applied to the Snark guidance problem. Steele took the information and diligently pursued it. He quickly found that others had already set out on a similar trail, notably the researchers at the Harvard Computation Laboratory under Howard Aiken, whom he contacted to gather more information on electronic digital logic.

I joined Northrop as a full-time employee after finishing my PhD in the spring of 1949, and Steele and I plunged into development of the digital computer. Ackerlind's group was now working on a new and much more powerful type of DIDA: the MADDIDA (MAgnetic Drum DIgital Differential Analyzer).

We made outstanding progress on the design, building, and testing of MADDIDA. Thus we were quite shocked that summer when Northrop management revealed to us that they had let a contract to another company to design and build a ground-based, general-purpose guidance computer for the Snark. Because of this, we were told, MADDIDA was no longer a major priority.

Our competition was a company founded by ENIAC's Eckert and Mauchly, and the agreement to build what would become known as BINAC had been forged in December 1947, when Robert Rawlins, our former project manager, had contracted with them for an *airborne* digital computer. I've since learned that Eckert and Mauchly, strapped

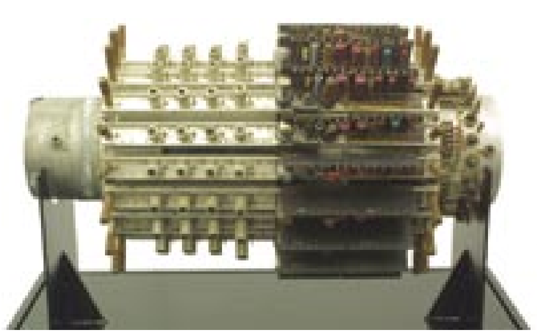
Courtesy of the Computer History Museum, Mountain View, CA, www.computerhistory.org; photos by Dag Spicer, drawing from a Northrop brochure.



The original MADDIDA (far right) had 22 integrators that performed the actual mathematics. Each integrator had two registers, called Y and R, to hold the two numbers to be added or subtracted. At MADDIDA's heart was a primitive hard drive (inset). About the size of a brake drum, it had only four tracks around its circumference. One track was the clock track, which was permanently scribed on the drum. The drum rotated past the read/write heads at a constant speed, with the clock track identifying what part of the drum was being read—the addresses of the bits on the other tracks, in other words. A "circulating register" track temporarily stored the contents of the various integrators (and the device numbers of said integrators) while the computation was being performed on them. The final two tracks—one each for the Y and R registers—contained the machine's program. These tracks specified at each step which integrators should be read into the circulating register and to which integrators the circulating track's outputs should be sent. So, for example, the circulating track might be holding the contents of integrator 5 and might be instructed to send the results to the Y register of integrator 19 and the R register of integrator 6. The output would be read off an oscilloscope (note the "scope selector" switch on the control panel, above). MADDIDA was programmed—in machine language, of course!—by pushing the "0" and "1" buttons in the middle of the panel.

for cash, had undertaken to build BINAC for an absurdly low figure—$100,000. Furthermore, in the course of negotiations, a requirement that the resulting system actually be installable in the Snark (or any other aircraft!) was essentially dropped. (These and many other details can be found in *ENIAC: The Triumphs and Tragedies of the World's First Computer*, by Scott McCartney, published by Walker and Company in 1999.)

Given how far along we were, our group was agonized by the news. Team member Richard Sprague would later write of us "swallowing our resentment" and agreeing to serve as liaison engineers with the Eckert and Mauchly Computer Company, whose executives worked with us and tutored us in ways that some members of our team found condescending.



Courtesy of the Computer History Museum

A mercury delay-line memory from UNIVAC I—high technology in 1951. The 18 long, thin, rectangular boxes running the length of the unit each contain a mercury-filled tube capable of storing 120 bits. This chunk of hardware weighs about 1,000 pounds.

The pill was bitter, but it came with a sweetening of additional information: we got a chance to look at BINAC and learn what another team faced with a similar problem had done. Two separate units of Northrop employees were flown to Philadelphia to watch the E&M staff conduct acceptance demonstrations of the new machine during a very hot August.

Of the E&M staff, BINAC's logic designer Bob Shaw especially stands out in my memory. A nearly blind, severely handicapped albino, he may have needed assistance drawing the computer's intricately detailed circuit diagrams, but he was gifted with a phenomenal recall. To compensate for his poor eyesight, he had simply memorized the diagrams!

I tried consciously to keep an open mind, and perhaps because of this, I was more impressed than my colleagues. BINAC proved to be a fine, though quite complex and not very reliable, machine that had at least the potential to do the job, given considerable modifications. In modern terminology, it used 32-bit words, each word being made up of two 16-bit instruction/address sets. With its 512 words of memory, one could enter a relatively sophisticated but limited program into it. It used mercury acoustic delay lines for high-speed memory. (These ingenious devices were mercury-filled tubes some five feet long, down which data was propagated in the form of sound waves. A transducer at the tube's front end converted the incoming electrical impulses into sound, and a transducer at the other end converted the sound back to electricity. Since sound in mercury travels much slower than electrons in copper, this allowed the data to be stored for a fraction of a second while the computation proceeded apace.) BINAC had, by the standards of the day, an extremely high clock rate—four megabits per second—and was, in fact, a pacesetter for all general-purpose computers to come. It was essentially the prototype for the famous UNIVAC, which was built by the firm that ultimately absorbed E&M.

Even though BINAC was not designed using Shannon's insights, it could be programmed to solve any type of problem, including a very small fraction of the system of differential equations needed to guide the Snark. By contrast, MADDIDA was a special-purpose machine, designed specifically to solve those equations efficiently.

And while BINAC—which consisted of two large racks of electronic equipment, all that heavy mercury plus the apparatus needed to heat it to its optimal operating temperature of 40° C, and an air conditioner to cool everything else—might possibly someday have flown in a large cargo plane, MADDIDA could actually fit into the fighter-sized Snark. Including its power supply, MADDIDA was a self-contained box about the size of a small four-cylinder gasoline engine.

Courtesy of the Computer History Museum

BINAC was a room-filling machine. The mercury memory is just visible in the background on the far left.

We lobbied Northrop's management to pursue our approach while still allowing for, and even encouraging, the use of BINAC as what we would now call a mainframe computer to do engineering calculations. In February 1950, Steele and I approached Jack Northrop, whose door was open to all employees, and suggested that we have mathematician John von Neumann of Princeton University's Institute for Advanced Study evaluate MADDIDA. Northrop agreed to this, but since von Neumann wasn't able to travel west at the time, MADDIDA was put on board a commercial airliner and flown east.

Lee Ohlinger, representing Northrop management, joined Steele, engineer Don Eckdahl, and me on the journey. We installed the machine in a fourth-floor room of the Princeton Inn—and immediately realized that the hotel didn't have the three-phase power MADDIDA needed. Eckdahl noticed that an electric company was located directly across the street, and that evening we implored them to string the required three cables to MADDIDA's hotel room. They somehow agreed, and we were in business.

Courtesy of the Computer History Museum

John von Neumann

The father of game theory, von Neumann was a legend in his own time, which is why we had proposed him as the judge of our project's worth: his voice carried unique weight. He was renowned not just for the depth and originality of his thought, but for the sheer speed with which his mind worked. He was a lightning calculator with a photographic memory. We met him the next morning in his office, where he told us that he had been carefully examining any reports he could find about the machine. He said he had felt that a differential analyzer such as ours could be extremely helpful in setting up aircraft control simulators—an important need at the time, given that the first jet airliners were being designed and that jets flew too fast to be controlled in the same manner as propeller aircraft.

He asked to see a diagram of the machine; the diagram was a large set of logic equations. This seemed to impress him in no small way. All the flip-flops of the machine were set or reset by ZERO-ONE logic statements or Boolean algebraic equations. "I always felt one should design a machine this way," he told us.

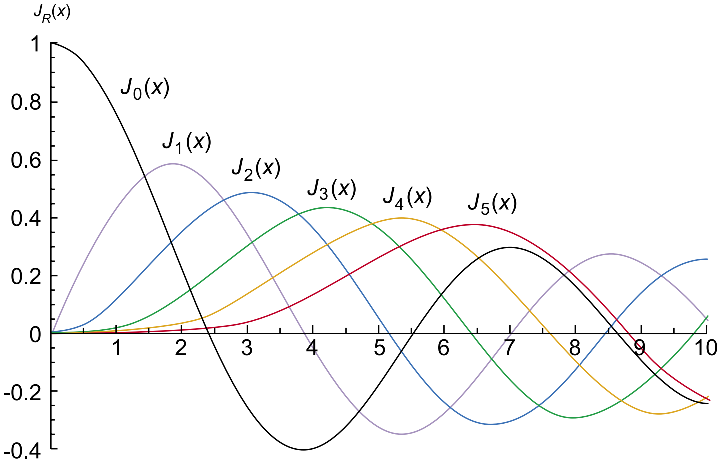
He asked us about the programming of the machine, so I moved to the blackboard to show him how we did it. This was a moment that was both exhilarating and daunting, as I now was demonstrating our programming techniques to a man I had long considered to be omniscient in mathematics.

Then Eckdahl explained how we performed logic design and computed the resistor values associated with the design process. This had to be done carefully, due to the considerable forward impedance in the germanium point-contact crystal diodes of the time. We then discussed what today would be called MADDIDA's architecture.

Our actual demonstration was set for the next day. Our tests that afternoon were complicated by a misbehaving DC power supply. Steele found the problem—a bad solder joint—and the next morning we programmed the machine with the Bessel differential equation.

This wasn't easy. Everything had to be entered using two pushbuttons for ONE and ZERO—we had no keyboard—and Eckdahl was the only person in the room, and probably one of three people in the world, who knew how to both encode the program and operate the machine.

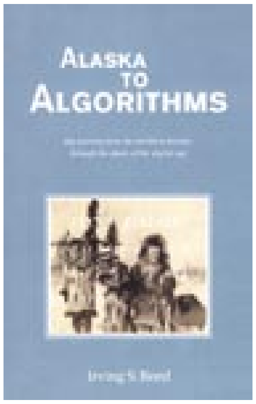
We programmed MADDIDA to calculate $J_{1/2}(x)$, a simple program that was relatively easy to enter. Eckdahl checked the program, started the machine, and launched the program as a test. Just then von



This set of Bessel functions, $J_0(x)$, $J_1(x)$, $J_2(x)$, $J_3(x)$, $J_4(x)$, and $J_5(x)$, are the solutions to the differential equation

$$x^2 = \frac{d^2 y}{d x^2} + x \frac{d y}{d x} + (x^2 - n^2) y = 0 \;;$$

John von Neumann kept pace with pencil and paper as MADDIDA solved the equation for $x = 1$, $x = 2$, $x = 3$, and so forth.



Alaska to Algorithms

By Irving S. Reed

Xlibris Corporation

2005

209 pages; $31.99 hardcover, $21.99 paperback

Neumann arrived unexpectedly and asked us what we were computing. He immediately sat down with a pencil and paper to compute the series himself. The machine calculated to the point x = 1, then stopped so the output could be read, then went to x = 2, stopped again, and so forth. Von Neumann kept pace.

He had no printed formulas or tables available. He had an almost instantaneous method for calculating the cosine of t and square roots—perhaps he had them all in memory—and was stunningly rapid at making estimates. (I've since learned from *A Beautiful Mind*, Sylvia Nasar's biography of John Nash, that von Neumann did similar feats versus other early computers.)

After the demonstration, von Neumann was visibly excited by what we had accomplished and said it was a genuine privilege to meet us. That was my last encounter with this mathematical hero, although I have kept with me a copy of the letter he wrote on March 14 to Jack Northrop, in which he called MADDIDA "a most remarkable and promising instrument" and suggested various applications for it, including determining molecular wave functions for quantum-theoretical chemistry—an application that remained beyond the reach of general-purpose machines until the supercomputers of the last decade. He concluded, "The fact that your machine could be transported by airplane and by truck from Los Angeles to Princeton and be satisfactorily running within 24 hours after its delivery is one of the most impressive engineering feats I have ever observed in this field. One has to be familiar with the great difficulties of running equipment of this type under even the most ideal laboratory conditions in order to appreciate the extraordinary tour de force of your group."

I've also kept a story from the March 28 *Newark Evening News* reporting on our demonstration (on that same trip) at the very first computer conference, cosponsored by Rutgers University's College of Engineering and the Association of Computing Machinery. We stole the show as the only full-scale, working electronic computing machine there—the others, behemoths that were very far from portable, were represented as models, photographs, or pieces of hardware. Headlined "Brainy 'MADDIDA' Makes First Appearance: Machine Can Operate Whole Factories," the article began with the sensational question: "Will people become obsolete?" and went on to say "A mechanical brain was placed on public display here last night which makes possible production of goods without help from the human hand or human brain. Its inventor called it the forerunner of the automatic factory. . . . Until last week, the awesome 'MADDIDA' was classified secret by the Air Force, for which it was built. Last night its inventor, Floyd Steele, an earnest, 31-year-old physicist and aeronautical engineer of Manhattan Beach, Cal., said of his handiwork: 'It will make a big impact on the American economy—bigger than the industrial revolution.'"

In an illustration of how the English language had not yet found words to describe the new electronic frontier, the accompanying photo had a caption that read: "That round thing in the foreground is MADDIDA's memory gadget."

The trip home was one of the happiest and most hopeful journeys of my life. I was only 26 years old, and had not only met a personal hero, but had had a hand in introducing him to a world-changing technology that I had helped bring forth. I had reason to be excited. ☐

Irving S. Reed (BS '44, PhD '49, both in mathematics) left Northrop in May 1950 to found the Computer Research Corporation with Steele, Harold Sarkissian, Sprague, and Eckdahl. This small start-up company was acquired by NCR in 1952, but by then Reed had already left for MIT's Lincoln Laboratory, which was working on a semi-automatic radar system to detect, identify, and track enemy aircraft that might be carrying atomic bombs before they got within striking distance of the United States. There he developed the first register-transfer-design language, which made it possible to translate large, complex logic statements into computer hardware almost automatically. In 1960, he and Gustave Solomon, a former grad student of John Nash, developed the Reed-Solomon error-correcting codes that allowed the Voyager missions to return images of the outer planets in unprecedented detail. Revolutionary at the time, Reed-Solomon codes are now used in everything from CDs to fax machines. (For more on Reed-Solomon codes, see E&S, *Summer 1989.) Reed returned to Southern California shortly thereafter, joining the senior staff of the Rand Corporation in Santa Monica. In 1963, he left industry for academe, joining the University of Southern California as an associate professor of electrical engineering. He is now the Powell Professor of Electrical Engineering and Computer Science, Emeritus. His awards are too numerous to mention.*

This article is condensed from Chapter Three of his autobiography, Alaska to Algorithms.