


[Home](#)
[Prehistoric Times](#)
[The 9100 Project](#)
[Prologues](#)
[The 9825 Projects](#)
[Epilogues](#)

Revised 11/11/2004

[Home](#)

Tom Osborne's Story in His Own Words

In 1994, Dave Packard was writing a book about the world-famous HP way and the founding of HP. Tom Osborne wrote up his recollections concerning his original calculator design (the Green Machine) and the HP 9100 and HP 35 calculator projects in a letter he sent to Barney Oliver, but the technical details in Osborne's letter didn't make their way into Packard's more business-oriented book. Below is the letter that Tom Osborne sent to Barney Oliver in 1994. The letter provides a wonderful record of Osborne's thoughts and experiences as he made history with the HP 9100 and HP 35 calculators.

Many thanks to Tom Osborne for permission to reproduce this letter.

[Note: I've made a few editorial insertions to improve the letter's clarity. These appear in square brackets.]

Barney:

Hopefully this is what you need to expand on Mr. Packard's history of HP. I am intentionally adding more detail than you need (and probably more than you want). One can always toss unwanted information, but the absence of facts can cause a bit of difficulty.

Most of the support for what follows is now in the archives of the Smithsonian. I kept reasonably good records, which they asked for about 20 years ago. By now they should carry enough dust to be credible. Without the hard facts, I will test my long term memory bank.



Tom Osborne shipped this box of documentation covering the development of the prototype calculator that would become HP's first desktop calculator, the HP 9100, to the Smithsonian along with other bits and pieces from the project.

Photo courtesy of Tom Osborne

The Green Machine had its beginnings in 1962 while I was a student at UC Berkeley. (It is called the Green Machine because of the Cadillac Green Metallic paint I used on it's balsa wood case.) At UCB, I had independently (and belatedly) re-discovered what is now known a Finite State Machine (FSM), but I moved the art from the province of mathematicians into the realm of engineering by discovering a set of transformations that could reduce the description of an FSM directly into very efficient digital circuitry. Later, we at HP called the entire process the Algorithmic State machine procedure.

I first used my transformations to produce digital circuits when I was at UC Berkeley. There I found they produced such elegant circuitry that it took me hours to reverse engineer the results to see why they worked so well. Most of the efficiency comes from the ASM's uncanny ability to perform operations that seem to be redundant (like setting an already set flip-flop) but careful examination shows that these seemingly ineffective actions greatly reduce the gate count required to implement the function as a whole. The technique also produces some actions that at first seem inefficient

but in fact set up conditions that greatly simplify later operations.

In 1962, after leaving UCB, I worked for SCM Corp (Smith-Corona Marchant), which was a major manufacturer of mechanical rotary calculators. They were trying to establish an electronics base. It was incredibly frustrating to watch them struggle with electronic digital design. I was a junior employee and totally unable to convince them that there was a better way to design things. In the fall of 1963 I told them that I could no longer help them produce a calculator that, in my opinion, was doomed to failure (it was, and it did). I offered to design a machine for them at no cost if they would give me lab space. Later, if they liked what they saw, they could pay me for the time I had spent in the design and construction. I do not know if they fired me or I quit, but they did get excited about me having not given them the plans for the calculator I must have designed. In their opinion, they could not see how I could so confidently state that I could build a calculator without having designed it. Nothing I could say would convince them that there were more efficient ways to design digital circuits.

In 1963 I was unemployed, miffed, but well armed with good design techniques. Carol [Osborne's wife] had a good job, so we were not about to starve. We discussed the options and decided to go ahead and build a calculator, but only after having sought legal help to ensure that SCM would not have any claims to my design. I went to one of the biggest law firms in San Francisco who helped me write a crackerjack of a note telling SCM that I had given them everything they had asked for except for the design of calculator, which at that time simply did not exist. The note politely but firmly said that if they wanted anything else they must tell me about it. They did not reply.

In January of 1964 I began the design of a calculator that was going to have several attributes not found in contemporary machines. It was going to be about 100 times faster, take about one tenth of the power, be about a third of the size and weight of the then existing calculators, and have a floating point arithmetic unit that produced 10 significant digits of accuracy over a dynamic range spanning 200 decades. Most of the speed came from a Random Access Memory (everyone else used delay line memory). The savings in power and size came from good engineering.

Things were a bit slow at the start. The calculator would be about 50 times larger than anything I had previously designed. Conventions and standards had to be developed; procedures had to be refined; and more importantly a ferrite-core Random Access Memory had to be designed. Ferrite-core RAMs had been around for a long time, but I had to build one costing several orders of magnitude less than anything then in existence.

Osborne shrewdly built the logic and memory for his prototype calculator into a separate box to allow the prototype keyboard and display unit to assume the size he believed would fairly represent the final production product. Otherwise, managers from prospective companies would get the impression that the calculator would be huge and unsellable.

Photo Courtesy of Tom Osborne



Because I had no laboratory equipment my circuit designs relied heavily on good modeling and careful calculations (it would have been nice to have had the calculator being designed to help me with those computations). I used less sophisticated equipment to achieve my goals. I calibrated the on-off periods of the calculator's asymmetric clock oscillator by listening to its fundamental harmonics on a radio. I built a sophisticated digital tester to debug the calculator's algorithms. Without that tester the project would have failed because of the enormous amount of time required to simultaneously monitor the outputs of 40 flip-flops, to set their initial conditions, and to stop processing at a predetermined condition previously set into the tester.

I remember an especially poignant day shortly after my arrival at HP when I was able to see, for the first time, most of the actual waveforms of the circuits I had designed. Once in the early stages of the calculator's design, I visited the UCB engineering lab late one evening to use an oscilloscope to tell me why my RAM sense amplifier responded so well to signals whose polarity was opposite to those I wanted to capture. That time was well spent because I had had stumbled onto a circuit that used the very low impedance of the RAM's sense windings to drive a transistor out of saturation. The circuit had superior noise immunity and performed far better than the one I had initially designed. Later, that same circuit would be used to build the HP 9100's ROM sense amplifiers.

The ASM process and the tester I had designed allowed me to debug the calculator's routines much as one debugs computer programs. Although the first routine I designed was a simple mathematical function (a shift operation), it was obvious that even with the tester, it took an enormous amount of time enter arguments and see the results. I decided to abandon the mathematical routines and concentrate on the keyboard and display routines and their associated hardware. Oddly enough, when the entry and display routines were completed, the calculator looked as if it was complete. However, it was several months before the calculator would be able to perform arithmetic.



Osborne housed his prototype calculator in a case made of balsa wood and Elmer's glue, which he then painted a metallic green selected from the 1960s paint palette of GM's Cadillac automotive division.

Photo courtesy of Tom Osborne

Finally, on Christmas Eve afternoon in 1964 the calculator was totally functional. I remember the overwhelming realization that sitting in front of me on a red card table in the corner of our bedroom/ workshop, sat more computing power per unit volume than had ever existed on this planet. I felt more like the discoverer of the object before me than its creator. I thought of things to come. If I could do this alone in my tiny apartment, then there were some big changes in store for the world.

To protect my inventions in the calculator, I wrote up a patent application and spent more money than I thought existed in getting it ready to file with the patent department. I did not file because I wanted the eventual buyer of the inventions to be able to tailor the claims and to redo the patent's specification. I also thought that the un-filed patent was as good as one that was filed (I had up to a year from the date of my first public disclosure to file, and all of the demonstrations I gave were on a confidential basis.)

I never considered manufacturing the calculator myself, so my task was to find a manufacturer for the Green Machine. I had already failed at selling my concept to SCM, but now that I could blow the socks off of anything on the market (and probably anything under development) I had no trouble in getting attention. I had discussions with more than 30 companies, all of which turned me down (HP was among those). Although no one wanted to buy the rights to the machine, I was given some incredibly good job offers.

I remember many humorous events surrounding the disclosures to various companies. The IBM people were not slowed down a bit by their inability to find the apartment's slightly hidden entry. They climbed the fire escape and knocked on the fire exit window. I opened it and they entered by climbing over the hi-fi set which partially blocked their entry. Through it all they retained their composure.

Most of the demonstrations were made in the presence of Richard T. Lacoss, a classmate at UC Berkeley whom I paid \$5.00 per demonstration. I had no idea of the impact he had on the people who came to our apartment for a demonstration. Apparently they were not expecting to have any witness at all, and especially one who was a final year PhD candidate. One time while waiting for the Monroe Calculator representative to arrive, Richard told me that he wished he had been named Thad (a contraction of his middle name). When Al Jensen, Monroe's chief engineer, arrived I introduced him to Thad Lacoss. That overloaded Richard's composure gland. He deteriorated one level every time Al called him Thad. One especially deep downturn occurred when Al asked "Thad" where he thought we should go for lunch. I knew Richard had totally missed hearing his new name and was lost in thought having nothing to do with my demonstration. Richard almost used the fire escape exit when I said "Well Thad, Al would like your recommendation on a place for lunch".



Tom Osborne demonstrated his calculator to several companies in his apartment, where he'd built the machine. Here, the calculator prototype (right) is readied for the showing, along with the logic module (center box), the power supply (on top of the logic module), and a specialized tester Osborne built (to the left of the logic module) that allowed him to debug the prototype without the expensive test equipment he'd otherwise need for such a project. The white mockup calculator on the left shows how large Osborne expected a finished calculator to be, once it housed the logic and power supply in addition to the keyboard and display.

Photo courtesy of Tom Osborne

Things were pretty low a year later when I was about at my rope's end in trying to market the calculator. Then, I received a phone call from HP's Paul Stoft. Paul told me that Tony Lukes, an HP employee and a co-worker at SCM with me, had told Paul that I was working on a calculator and that I had a reputation for making things that worked. Paul asked me to bring the Green machine down to HP. I showed it to Paul. He then introduced me to Barney Oliver who watched the demo and asked me if I could make it do the transcendental functions and if I had ever heard of the Cordic Algorithm. Yes and No to those. Barney then showed me a 64-bit ROM that needed help, but showed promise. He wanted to know if I could implement things in ROM. Again, I knew I could, but I was a bit worried about increasing its bit capacity by a factor of 500. To me it appeared that each of us thought that there was no reason not to proceed. Barney then asked me if I could come back the next day to meet with Dave and Bill. That was OK with me, but I asked, "Bill and Dave who?" (that was the first time I had heard their first names).

The meeting the next day was memorable. Nothing like it had occurred with the previous companies. It appeared to me that while other companies were looking for a weakness that might preclude them from success, HP was looking for the opportunity to might lead to a success. We discussed the project's good points, its weak points, and the risks involved to both parties. We agreed to give it a try for six weeks during which I would explain my design processes to HP's engineers and perform a total evaluation of the project. The meeting was about over when Mr. Packard said, "Oh Tom, we won't take the project without you coming along with it." I said, "You can't have it without me." In those few words it was clear to me that one of my main tasks was to transfer the information that existed only in my mind into the minds of the people with whom I would be working for a couple of years. It was also clear that I was among people who were open minded and trained to advance the state of the art.

The six week investigation went well. I found the HP employees to be enthusiastic and very competent. I had some difficulty in explaining some of my processes because they had not been named. I had never explained them to anyone except a few people at SCM and the process had been greatly enhanced since I had left them. Although the processes were clear in my mind, I had not given many of them names, so I found that I had to name these processes before I could explain them effectively.

It was a busy time. I spent a few weeks with Dave Cochran (HP) and Malcolm MacMillan, a consulting mathematician who had reduced the Cordic Algorithm into hardware. I did not see his machine, and from all accounts it did not work very well. None the less, Malcolm knew his mathematics, and explained the Cordic algorithm well. It was the basis of the transcendental functions in the calculator we would build. Malcolm had a quality I have not seen in anyone else. Late one evening he told Dave and me that he had perfect pitch. With that Dave hauled out an audio oscillator, shorted the output (we had no speaker, but the short caused the output transformer's laminations to hum enough to be heard). Dave spun the oscillator's dial to some random frequency and looked at Malcolm, who winced and told us that the music world does not recognize the continuous spectrum. It only considers the notes to which a piano is tuned. None-the-less, he would give it a try. He hummed a bit, closed his eyes a lot and absolutely nailed the frequency. I was then, and now am even more, impressed with his performance.

At the end of the six week evaluation process, Al Smith of HP's legal department dropped by and gave me a check which meant that the project was a go. I was excited. The check was nice, but the opportunity to do the project was everything I had hoped for. At that time, I decided that if the project was to get into serious trouble, I would be the first one to know, and I would immediately tell Barney that we should cancel the project. The HP people had placed a great deal of faith and trust in me and I was not going to misuse it.

For the next two years I spent some long hours keeping the various aspects of the HP 9100 project on course. I was barely able to stay ahead of the alligators on my tail. The ASM aspects of project were the easiest part of the project to manage. The HP engineers, as good as they are, were behind the art in digital design in my opinion. They mastered the ASM design technique, used it well, and soon were at the leading edge designing ASMs.

At the other end of the spectrum was the HP 9100's inductive ROM. Very few, if any, other companies had the technical and manufacturing skill to produce a ROM of such complexity. It could not have been built without a fundamental knowledge of physics, good engineering, and hard work. The ROM's sense amplifiers were, by far, the most difficult circuits I have ever had to design. The sense amplifiers detected a millivolt signal lasting but nanoseconds, which were imbedded in common-mode noise. From these feeble signals, they generated a 15 volt signal, capable of driving more than 100 milliamperes for more than a microsecond. The signal was also gated so that it could be selected on or off, and it was able to recover for a new cycle that would occur a few hundred nanoseconds later. All of this was done with three transistors and a tiny ferrite transformer.

When we began designing the multilayer printed-circuit ROM, I discovered a quality that characterizes many of the HP engineers. The ROM's bit density required a pc [printed-circuit] line width that was smaller than the then-existing HP standards. I met with Clyde Coombs and Bob Perricone to see how much they could reduce their limit. As I recall, the standards were a 50-mil line width with 50-mil line separation. Try as I might, I could not get them to go down as far as we needed. Maybe they could go down 10 or 20 percent, but no more. Then I told them we needed a 70% reduction (to 15 mils). My requirement was taken seriously. The next day, Bob Perricone called to tell me that he thought of a different process that might produce a 15-mil line width [eventually, the pc-board shop produced 10-mil lines]. He had to abandon many of the things that set the 50-mil line limit. To me, HP Engineers are not always self limited [and do not always] sail in safe territory. They understand the fundamentals, think well, and take calculated risks.

The HP 9100 project had one major philosophical change. We decided to make it into a programmable calculator, but that required a different architecture than the one embedded into the Green Machine. I changed the architecture to use RPN (Reverse Polish Notation), which is the ideal notation for programming environment in which coding efficiency is critical. In the beginning, that change was not well received, and without Barney Oliver's persuasive powers, the project probably would have not have survived.

I had a unique relationship with the people of HP. I had the freedom to talk with a broad spectrum of people. I was not pinned between hierarchical levels. The fact that I was given the freedom to discuss problems with those who could best solve them was critical to the success of the HP 9100. I believe I did not step on too many toes while we were developing projects. I was not after anyone's job and only wanted to help bring products to life. From my point of view, I could not have been given more support or freedom. I tried not to misuse my status. Most of the time I could give people ample warning of an upcoming critical event and they made room for the event, even though it was an R&D environment, and subject to unknown delays and functional changes. A couple of times, we called for outside help. Because HP had not yet invented their first computer, Carol wrote the IBM assembly program that programmed the HP 9100 ROM.

The project moved from HP Labs to HP's Loveland Division where it was manufactured. The transition went smoothly, mainly because of the high standards and professionalism of the Colorado Crew.

We introduced the HP 9100 in March of 1968, a bit over 2 years from the time I first demonstrated the Green Machine to Paul Stoft. It was well received. Although there is always the danger that one's self perception can be over elevated, I was brought into calibration by a customer named Steve Medwadowski. Steve had an Olivetti programmable calculator for which he had written several elegant structural-engineering programs. He was curious about the HP 9100 so, during a demonstration, I programmed the HP 9100 to solve a problem he had previously programmed. I was lucky enough to have the program run on the first pass. Steve was impressed too. Whereas his program took about a minute to run, mine took about a second. I asked him if he liked the 60 to 1 improvement. Steve let most of the air out of my balloon when he said that before programming the Olivetti, his hand solution took about an hour, so that from his point of view, the Olivetti saved him 59 minutes and the HP 9100 would only save him an additional minute.

The HP 9100's architecture was very similar to that in the Green Machine. Unlike conventional computer architectures that perform but one computer instruction or branch instruction per clock cycle, the HP 9100 performed several instructions AND could branch to any of several locations on each clock cycle.

The high degree of parallelism accounted for the HP 9100's high speed. Even though it used a ferrite-core RAM (which required a restore cycle every time it was read), the HP 9100 was faster than the next two generations of HP calculators (which were based on integrated circuits and conventional computer architecture). Both the Green Machine and the HP 9100 had "power gating," meaning that the power that drives a logic gate could be also be used as a logic signal. This technique had two significant benefits. First it reduced the gate count by about 25%, but more importantly, it reduced the power consumption by about 90%. We received a patent on that process and applied it broadly to calculating devices using integrated circuit memories with "chip select." (Virtually all RAMS have chip select. HP did extremely well by collecting royalties on this invention).

The HP 9100 had no integrated circuits because they were just coming into existence and consumed huge amounts of power. In the beginning, I received a bit of pressure from Paul Stoft who wanted to make the HP 9100 an IC-based machine. I asked Paul to make an estimate of the number of ICs that would be required to build the machine and then compute the power required to supply those circuits. If he could build a power supply that would fit into the case we were considering, then I would consider the ICs. (At that time HP was developing the HP 2116 computer whose architecture was much simpler than the calculator's. The HP 2116's power supply was about 5 or 10 times the size of the HP 9100's case.)

Two major peripherals, a printer and a plotter, were built for the HP 9100. Barney Oliver wrote a program to show the 3-dimensional plot of an optical diffraction pattern, which has since become a de facto standard for numerous manufacturer's plotter and graphic terminal advertisements.

At the introduction the HP 9100 at the IEEE show [Electro] in March of 1968, Bill Hewlett, Barney Oliver and I gave a private showing to An Wang. Wang Laboratories had one of the best calculators on the market and the HP 9100 was obviously going to seriously affect their position.

Dr. Wang's parting words were, "You have a good machine. We had better get busy." After he had gone, Bill Hewlett said, "I think the next machine should be a tenth the cost, a tenth the size, and be ten times faster than the HP 9100." I knew he was serious, but we were kept hostage by the lack of low-power integrated circuits. A couple of companies were close to a solution, but would be three years before I read in a trade magazine that a relatively small upstart (MOSTEK) had an IC process using ion-implantation that would reduce the power levels of ICs by a couple of orders of magnitude. During the time between the introduction of the HP 9100 and the ion-implantation article, I was visited regularly by Bill Hewlett who wanted to know why we were not working on the calculator he had prescribed. The pressure cooker would have been hotter had Mr. Packard not been in Washington at the Department of Defense. None-the-less, when he did visit the labs, he asked the same question.

The day I read about ion-implantation, I went to Bill Hewlett and told him that I thought we could build the machine he wanted. I knew I would have a bit of trouble getting the project staffed because Paul Stoft, Tom Whitney, and Tom's engineers were hard at work on a briefcase-sized something-or-other. This was one of a very few times that I used whatever power I had to pressure Paul and Tom to change their minds about what they wanted to be doing for the next couple of years. I told them that if they were sure we could not do a shirt pocket calculator, then they would have to explain their reasons to Bill Hewlett because I had already told him that we could do it. During the next few days, they decided that it just might be possible. In another few days the project became exothermic, and we began some serious design.

My role in the HP 35 was quite different than that in the HP 9100. Except for the card reader and the power supply, I did most of the circuit design in the HP 9100. I did none of it in the HP 35. Instead, almost all of my effort went into prescribing functional characteristics. This time, I knew that someone would want the calculator that followed the HP 35 to be programmable. A couple of times, I dug in and argued for features that would grease the slides for the follow-on product, the HP 65 (which, I think, was the best product I ever worked on).



As with Osborne's HP 9100 desktop calculator, the physical prototype of the HP 35 pocket scientific calculator contained only the keyboard and display. Another case contained the prototype electronics. This design approach allowed the prototype to take on the appearance of the final design without requiring that the prototype electronics fit into the target case.

Photo courtesy of Tom Osborne

The HP 35 project was just plain fun. We knew it would work (the algorithms were similar to those in the HP 9100) so we spent a lot of time deciding whether the arithmetic keys should be on the right or the left, and whether the "+" key should follow the convention of adding machines and be placed in the lower right corner, or whether it should be located where it is most convenient. As I recall, we did not seem to be the least bit worried about the fact that we were going to be the first people to have non-standard key spacing. After all, if the thing was to fit in a shirt pocket, the keys had to be crunched together.

I was a bit concerned about the stack architecture and RPN notation, but that all went away when I demonstrated one of the first prototypes to my mother-in-law, who is anything but a mathematician. After I had done some fairly complex operations, Fran said, "How many things can you stack up in that machine?" With that comment, my worries disappeared.

Looking back, it seems as if the HP 35 had a life of its own. It simply chose HP as its birthplace. It had no name until we were about to introduce it. I asked Bill if he had any preferences. We had kicked around a few somewhat wild names but none seemed quite right. Bill looked at the machine for a minute or so and said, "Let's call it the HP 35." It sounded OK to me, but why the 35? He smiled and said, "Well, it has 35 keys".

The HP 35's design-to-production cycle was incredibly short for a product of that complexity. We got the official go ahead on Ground Hog's day (2 Feb) and demonstrated working machines to the Board of Directors in August. I still do not know how it happened in such a short time. I remember being busy, but I do not recall the HP35 having had any major hang-ups.

We had no idea whether the HP 35 would be a success or a dud. (Before it was introduced, a market analysis by a major consulting firm had determined that it would fail because of the tiny keys and the RPN notation. In my opinion, it succeeded for those reasons.) Anyway, we gave it our all and found that it was so well received that overnight, it made the slide rule a relic.

As I said, the slides were greased for the HP 65 (the world's first programmable shirt pocket calculator). That too was a super project. By that time, everyone in the calculator business was aiming at HP. I knew they would be second guessing what we were doing, so at a convention at which I was a speaker, I somewhat intentionally misled the competition by telling them that to just have a programmable calculator was inadequate. The programs had to be easily loaded into the machine. Keying them in once is OK, but successive loadings should come from a magnetic media, like an external tape reader (at the time we were designing an internal card reader into the HP 65). When we introduced the HP 65 with its tiny internal mag reader, I could hear the competitor's projects fall by the wayside.

Both the HP 35 and the HP 65 are on permanent exhibit at the Smithsonian. They changed the world we live in. I am truly honored to have worked with and for a company whose underlying principle and business is that of providing the world with products that allow it to be a better place.

HP9825.COM is in no way associated with the Hewlett-Packard Company.

All text Copyright 2004 to 2010 - Steve Leibson

Hewlett-Packard, HP, the HP logos, the HP 9825, and all other product model numbers starting with HP are all trademarks of the Hewlett-Packard Company and don't let nobody tell you different!

This site exists strictly for educational purposes. Nothing is for sale here. Substantial material from various issues of the incomparable Hewlett-Packard Journal appears on this site, by permission of the Hewlett-Packard Company.

[\[Home\]](#) [\[Prehistoric Times\]](#) [\[The 9100 Project\]](#) [\[Proloques\]](#) [\[The 9825 Projects\]](#) [\[Epiloques\]](#)