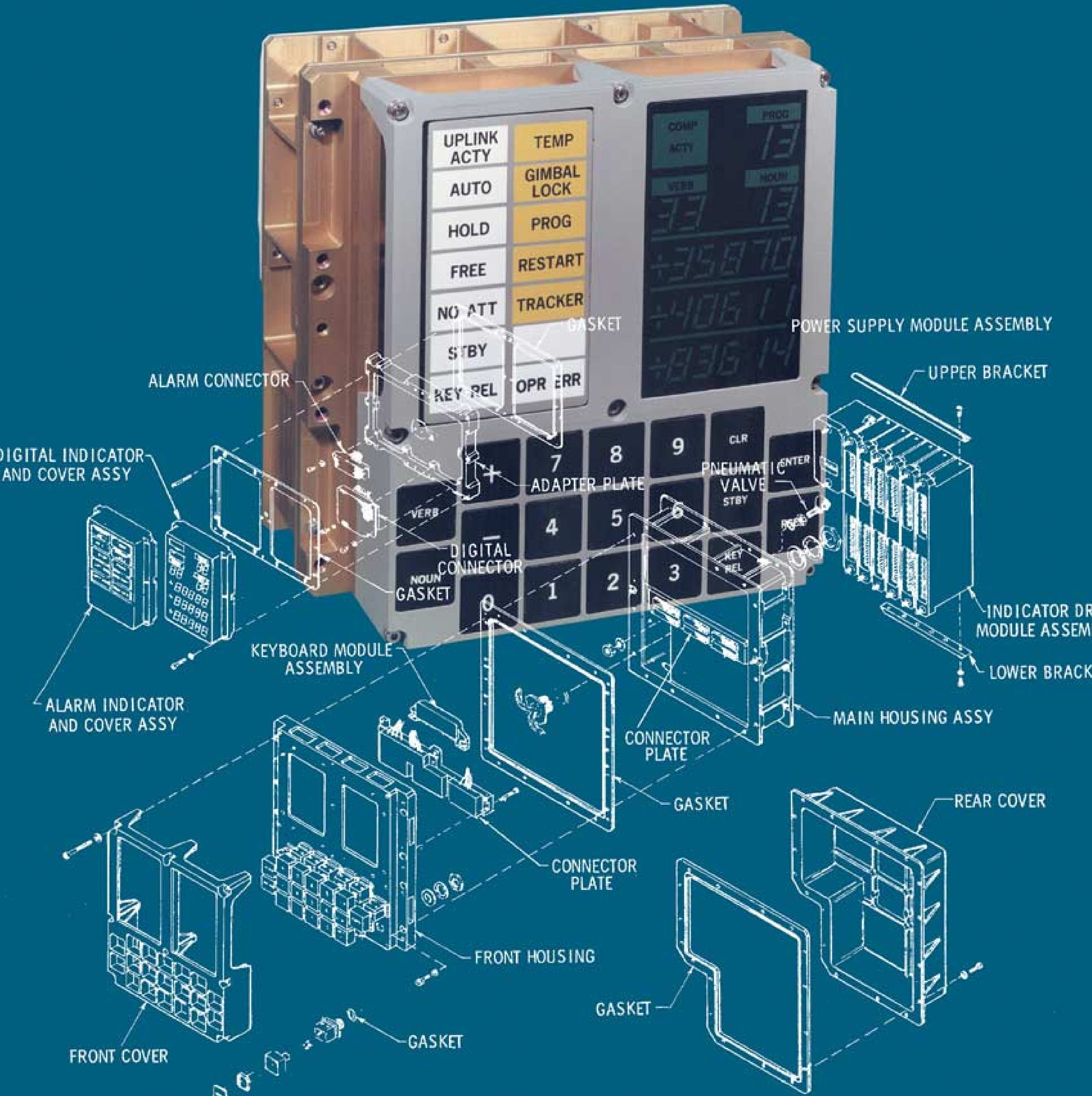
MAY 2002

CORE 3.2

A PUBLICATION OF THE COMPUTER HISTORY MUSEUM
WWW.COMPUTERHISTORY.ORG



FUNDAMENTALS IN CHANGING TIMES

As our fiscal year ends in June, it's natural to look at the Museum's accomplishments and future plans. It is also a time to reflect on how amazing our annual fundraising support has been during a difficult year in the U.S. and around the world. Thank you to everyone who has contributed to our expanding programs and enabled us to grow in stature, capability, and professionalism! It is critically important to operate in the black, and I am happy to report that our audited 2001 financial statements show exactly that. With your continued support, we expect to do the same this year and in the upcoming fiscal year that starts on July 1.

The economy, the war on terrorism, and the corresponding impacts on local climates have been extraordinary challenges for all non-profits, but the Museum has remained strong with your help. This is an important testament to our base of support, which has helped this organization through good times and bad. The mission of preserving the stories and artifacts of the information age strikes a fundamental note in many people's minds, which makes our organization solid even in challenging times. If you have not already donated to our annual campaign, please consider this mission and what we are trying to accomplish, and become a contributor—we have included an insert in this issue to make it as easy as possible.

Look carefully at all the activities reported in this issue, and you will see how our organization is growing. The free lecture series has been a tremendous success. Our curatorial staff is doing an outstanding job in organizing the collections, focusing on future exhibits, and working with an impressive list of volunteers who are helping as docents, greeters, and enthusiastic helpers. We are also finding ourselves much more prominent in the press. Tours of our Visible Storage Exhibit Area (with expanded Saturday hours twice a month) provide visitor access to our collection and demonstrate our emphasis on content in the fulfillment of our mission. Finally, the new building architecture team, led by EHDD, completed their schematic design phase, and delivered an amazing set of great ideas for our permanent home. The schematic design phase of exhibit design will continue through early fall.

While our public presence has continued to increase during this economic downturn, the Trustees and staff have also considered the challenges, opportunities, and risks at every stage. In fact, we have been constantly evaluating our long-term plans, and have developed new insights into the future. Although it's too early to publicly address any emerging options, we are continually challenging our assumptions as we search for the best investments of our resources. The changing economy poses some unique opportunities today, but also challenges us to project our next 10 years very carefully. We also are getting much more information on the costs and timelines for our plan of record with NASA, which becomes important to our analysis. The "Beta Building" that will provide additional room for us to grow is still a major priority, but will be delayed several months in this calendar year as we refine our plans. Stay tuned for more information.

Although, over time, plans and details may evolve to meet opportunities and to address challenges, the building blocks of our organization—the people, the collection, and the mission—are fundamentally strong and the basis of a great institution. Help us make this year the best ever!

John C Toole

JOHN C TOOLE
EXECUTIVE DIRECTOR & CEO

May 2002
A publication of the Computer History Museum
CORE 3.2

MISSION
TO PRESERVE AND PRESENT FOR POSTERITY THE ARTIFACTS AND STORIES OF THE INFORMATION AGE

VISION
TO EXPLORE THE COMPUTING REVOLUTION AND ITS IMPACT ON THE HUMAN EXPERIENCE

EXECUTIVE STAFF

John C Toole
EXECUTIVE DIRECTOR & CEO

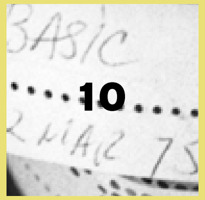
Karen Mathews
EXECUTIVE VICE PRESIDENT

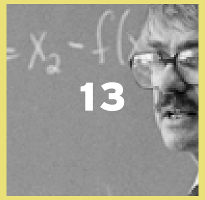
David A Miller
VICE PRESIDENT OF DEVELOPMENT

Michael R Williams
HEAD CURATOR

PUBLICATION STAFF

Karyn Wolfe
EDITOR





IN THIS ISSUE

The Museum is an independent 501(c)(3) organization, FID #77-0507525. PO Box 367, Moffett Field, CA 94035, USA.

Computer History Museum
Building T12-A
Moffett Field, CA 94035, USA
+1 650 604 2579
+1 650 604 2594 (fax)

WWW.COMPUTERHISTORY.ORG

Submission guidelines for technical articles can be found at www.computerhistory.org/core, or contact the editor at core@computerhistory.org.

Cover: Photo and exploded-view diagram of the Apollo Guidance Computer Display Keyboard (DSKY)

THE APOLLO GUIDANCE COMPUTER

BY ELDON HALL AND DAVID SCOTT

INTRODUCTION

The following article is drawn from a lecture given by Apollo Guidance Computer (AGC) lead designer Eldon Hall on June 10, 1982 at The Computer Museum in Boston. It was first printed in *The Computer Museum Report* in Fall, 1982 and provides some insight into the development of a major component that allowed "a giant leap for mankind."

The Computer History Museum collection contains several items and prototypes comprising the AGC, including logic modules, a DSKY, and rope memory; as well as lecture videotape; photos of the units in use and under test; and various paper documents that provide us with further details.

Eldon Hall led the hardware design effort throughout the development of the AGC and pioneered the use of integrated circuits in this design. His group at the MIT Instrumentation Laboratory (MIT/IL) was awarded the contract in 1961 to begin work on the Apollo Guidance Computer after their successful work on the Polaris missile project, in which Hall was responsible for encouraging the Navy to use digital guidance computers. Hall received his AB in Mathematics at Eastern Nazarene College, his AM in Physics at Boston University, and had completed much of a PhD in Physics from Harvard when he took a position at MIT/IL in 1952.

DESIGNING THE AGC

BY ELDON HALL

In the early sixties the so-called mini-computer had not emerged and there was no commercial computer suitable for use in the Apollo mission. Most of the technologies that were eventually used in the Apollo computer were ones just emerging from research and development efforts. The "design" was mainly a task of fitting the components together in order to meet the mission requirements for computational capacity and miniaturization.

FROM POLARIS TO APOLLO

Previous aerospace computers greatly influenced the development of the Apollo Guidance Computer. The demands placed on these computers provided the motivation to miniaturize and develop semiconductors. The MIT Instrumentation Lab, now called Charles Stark Draper Laboratory, had responsibility for the design of the computers used in the Polaris, Poseidon, and Apollo programs.

The lab's first significant venture into the field of digital computing was, for the Polaris program, a very small ballistic missile launched from a submarine. A special-purpose digital computer was designed to solve the specific equations required for the guidance and control system based on analog techniques originally developed by the Navy. With a need for increased accuracy, the Navy decided to use digital techniques for the Polaris program, resulting in the construction of a wired-program, special-purpose computer to solve the guidance and control equations. In 1959 the first version of this system, called the Mark 1, flew in a Polaris missile. It was the first ballistic missile flown with an on-board digital computer providing the guidance and control computations. The computer occupied about four-tenths of a cubic foot, weighed 26 pounds, and consumed 80 watts. Even before this first guided flight succeeded, designs were already being explored that would reduce the size and improve the maintainability of the system. The new design, eventually designated Mark 2, repeated the architecture and logic design with improvements in circuits and packaging.

In August 1961, when NASA contracted the laboratory to develop the Apollo guidance, navigation, and control system, the mission and its hardware were defined in only very broad terms. A general-purpose digital computer would be required to handle the data and computational needs of the spacecraft. Therefore a special arrangement of display and controls would be necessary for in-flight operation. The boost phase of the mission, which was the Saturn system, had its own internal guidance system to put the command and service module in translunar trajectory. Then the Apollo system took over to guide the mission to the moon.

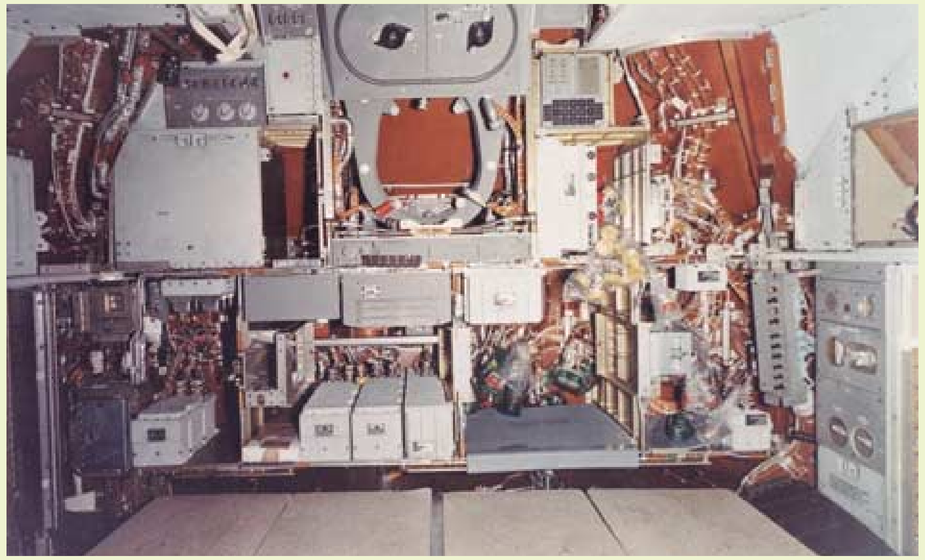
In effect, navigating in space is the same as navigating on Earth. One might take a star sighting with a sextant. That information is put into the computer and from it the state vector, i.e. the position and velocity of the missile at any point of time, is computed. The computer orients the missile such that the change in velocity will cause the state vector to be updated so the missile will free-fall into the targeted point. While it is thrusting, the guidance system must control the attitude of the vehicle, the magnitude of the thrust in the case of the Lunar Excursion Module (LEM), and the direction of the thrust in the case of the command and service module.

DESIGN CONSTRAINTS

Initially the need for a very reliable computer with significant computational capacity and speed was clear. The design constraints included very limited size, weight, and power consumption. If the designers had known then what they learned later, or had a complete set of specifications been available as might be expected in today's environment, they would probably have concluded that there was no solution with the technology of the early sixties.

Establishing interface requirements was a monumental task. The astronaut interface was one of these. In 1962, computers were not considered user-friendly. Heated debates arose over the nature of the computer displays. One faction, which usually included the astronauts, argued that meters and dials were necessary. Logically, the pressure for digital displays won most of the arguments because of their greater flexibility in the limited area allowed for a control panel. In late 1963, as the requirements for the LEM were being firmed up, NASA decided to use identical guidance computers in both the command module and the LEM.

In the early manned orbital missions before Apollo, NASA learned that the human animal, confined in a spacecraft for a week or so, was not as clean as might be expected from observations on Earth. This additional constraint had



Inside the Apollo capsule

Assemblers at Raytheon testing and building AGC modules

Lead designer Eldon Hall testing the Apollo Guidance Computer

The interface with the astronauts was the DSKY (for display keyboard). It used digital displays and communicated with the astronauts using verb and noun patterns and two-digit operation and operand codes. A set of status and caution lights is shown in the top left corner.

a rather interesting and far-reaching impact on the mechanical design of the computers and other hardware. All electrical connections and metallic surfaces had to be corrosion resistant and even though the computer was designed to have pluggable modules, everything had to be hermetically sealed.

THE SUPPLIERS

By the end of 1962, NASA selected contractors: General Motors' AC Sparkplug Division for the inertial systems and system integration; Raytheon, Sudbury Division, for the computer and computer testing equipment; Kollsman Instrument for the optical systems; North American Aviation for the command and service module; and Grumman Aircraft for the Lunar Excursion Module.

In late 1959 and 1960 the lab began evaluating semiconductors, purchased at $1,000 each from Texas Instruments. Reliability, power consumption, noise generation, and noise susceptibility were the prime subjects of concern in the use of integrated circuits in the AGC. The performance of these units under evaluation was sufficient to justify their exclusive use in place of the core transistor logic proposed initially for the Apollo project design. The micrologic version of the Apollo computer was constructed and tested in mid-1962 to discover the problems that the circuits might exhibit when used in large numbers. Finally, in 1964, Philco-Ford was chosen to supply the integrated circuits used in the prototype computer that operated in February 1965. These cost approximately $25 each.

SPECIFICATIONS

Approximately one cubic foot had been allocated in the command module for the computer. The first prototype was operating in the spring of 1964 and utilized the wire wrap and modular welded cordwood construction that had been produced for the Polaris program. It was designed to have pluggable trays with room for spare trays.

Since the clock in the computer was the prime source of time, it had to be accurate to within a few parts per million. The data and instruction words in the memory were 15 bits plus parity. Data was represented as 14-bit binary words plus the sign bit. Double-precision operations were provided to supply 28-bit computations. The instruction word contained the address and operation codes for the computer operation. The memory address field was extended by organizing the memory in banks.

The AGC had 2,000 15-bit words of erasable core memory and started with 12,000 words of read-only memory, called rope memory. It was quickly upgraded to 24,000 words. Then by mid-1964, when the first mission program requirements had been conceived and documented, there was increasing concern about the possible insufficiency of the memory. This prompted a further expansion to 36,000 words.

DESIGN AND USE OF THE CONSOLE

A display and keyboard was developed for the astronauts and had the designation DSKY (pronounced "Diskey"). Functionally, the DSKY was an integral part of the computer, and two were mounted remotely and operated through the discrete interface circuits. One was for a sitting position and another one near the entry to the LEM, convenient for a reclining position.

The principle part of the DSKY display was a set of three numeric light registers. Each register contained five decimal digits consisting of segmented electro-luminescent lights. Five decimal digits were used so that a computer word of 15 bits could be displayed in either decimal or octal. In addition, three two-digit numeric displays indicated the major program in progress, the verb code, and the noun code. The verb/noun format permitted communication in a language whose syntax was similar to that of spoken language. Examples of verbs were display, monitor, load, and proceed. Examples of nouns were time, gimbal angles, error indications, and star identifications. Commands and requests were made in a form of sentences, each with a noun and a verb, such as "display velocity" or "load desired angle." To command the computer, the operator pressed the Verb key followed by a two-digit code. This entered the desired verb into the computer. The operator then pressed the Noun key and a corresponding code. When the enter key was pressed, the computer carried out the operation that had been commanded. The computer requested action from the operator by displaying a verb and noun in flashing lights to attract the astronauts' attention.

IN-FLIGHT USE

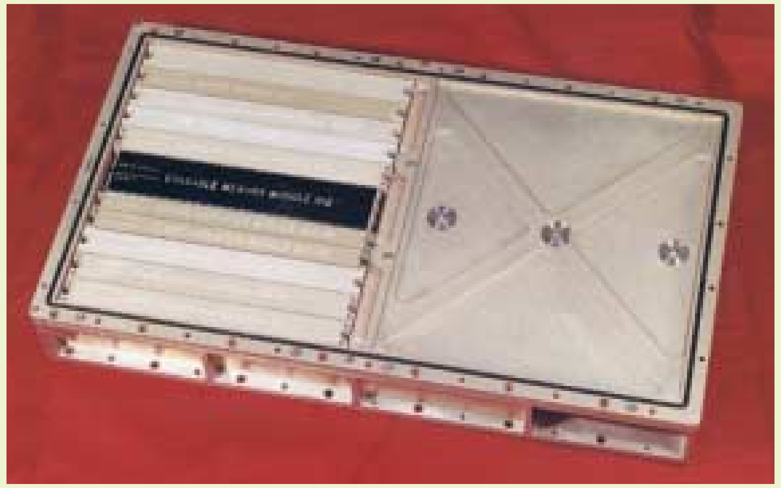
Shortly after the lift-off of Apollo 12, two lightning bolts struck the spacecraft. The current passed through the command module and induced temporary power failure in the fuel cells supplying power to the AGC. During the incident, the voltage fail circuits in the computer detected a series of power trenches and triggered several restarts. The computer withstood these without interruption of the mission programs or loss of data.

The read-only memory of the computer consisted of six rope memory modules, each containing 6,000 words of memory. This special type of core memory depended on the patterns set at the time of manufacture. Its sensing wires were woven into a set pattern. It had five times the density and was far more reliable than the coincident current core memory used for erasable storage in the computer. Being unalterable, it also provided a greater incentive for error-free software development.

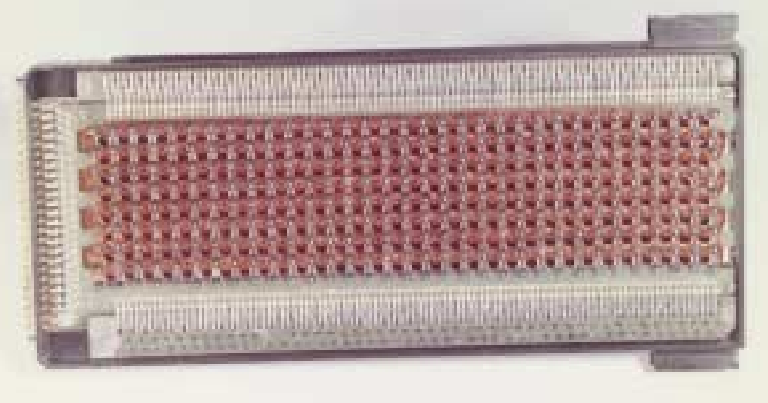
The Apollo 11 lunar landing had an anomaly that attracted public attention. The computer in the LEM signaled a restart alarm condition several times during a very critical period prior to touchdown. This fact was broadcast to the public and those who knew its significance were close to a state of panic.

After analysis, it was determined that the alarms were an indication to the astronauts that the computer was overloaded and was eliminating low priority tasks from the waitlist. The overload resulted from the rendezvous radar being set in the wrong mode during the lunar landing phase, wasting computer memory cycles. The computer software was responding to overloads as designed.

This incident triggered a news brief in Datamation in October, 1969, faulting the computer design for being too slow. It rightfully claimed that there were a number of minicomputers, including the PDP-11, that were at least an order of magnitude faster. In the eight years since the initiation of the Apollo program, commercial technology had far surpassed that of the Apollo design and capacity. However, no commercial computer could claim to match the power consumption and space characteristics of the AGC.



The Apollo Guidance Computer was responsible for the guidance, navigation, and control computations in the Apollo space capsule. The AGC was the first computer to use integrated circuit logic and occupied less than one cubic foot of the spacecraft. It stored data in 15-bit words plus a parity bit and had a memory cycle time of 11.7 microseconds, utilizing 2,000 words of erasable core memory and 36,000 words of read-only memory. The frame is made of magnesium for lightness and designed to hermetically seal the components.



The read-only memory of the computer consisted of six rope memory modules, each containing 6,000 words of memory. This unique type of core memory treated each core as a transformer within a matrix of discrete "rope-like" wires and depended on the patterns set at the time of manufacture. Wires running through the core stored a "1," and those bypassing the core represented a "0." It had five times the density and was far more reliable than the coincident current core memory used for erasable storage in the computer. Being unalterable, it also provided a greater incentive for error-free software development.

The module in the collection has been used only on Earth. The Museum's prototype computer ran at Draper Labs and was used to test the routines for in-flight machines. However, in space, all of the components had to be completely "potted" to insure that all the parts would stay firmly in place and remain uncontaminated.

Photo courtesy of NASA.

The Apollo 9 prime crew from left to right: Commander James A McDivitt, Command Module Pilot David R Scott, and Lunar Module Pilot Russell L Schweickart. The Apollo 9 mission was designed to test the Apollo Command/Service Module (CSM) and Lunar Module (LM) in Earth orbit to verify that the CSM could successfully dock with the LM, and to test the LM systems in a “free flying” attitude to ensure that it performed as per specifications.

MISSIONS WITH THE AGC

BY DAVID SCOTT

In 1963, when NASA was conducting the selection of the third group of astronauts for the U.S. space program, I had just received a graduate degree at MIT and finished test pilots school. My interests and the program’s need for a user to interact with the design of the guidance computer at the MIT Instrumentation Lab were a good fit. I was part of the discussions whether to use analog or digital controls.

THE MIT INTERFACE

When I was studying at MIT, the ability to rendezvous in space was an issue for debate. It wasn’t clear whether it was possible to develop the mathematics and speed of computation necessary to bring two vehicles together at a precise point in space and time—a critical issue for the Apollo mission’s successful landing on the moon and return to Earth. Between 1963 and 1969, with the flight of Apollo 9, this was accomplished. I stayed in the spacecraft while Rusty Schweickart and Jim McDivitt got in the lunar module and went out about 60 miles away. The computer behaved flawlessly during our first successful rendezvous in space.

Another assignment for Apollo 9 was to take the first infrared photographs of the Earth from space. To do this, a large rack of four cameras was mounted on the spacecraft. Since they were fixed to the spacecraft, the vehicle itself had to track a perfect orbit such that the cameras were precisely vertical with respect to the surface that they were photographing. During simulations it was determined that manual orbit procedures would be inaccurate. We were at a loss.

About two weeks before the flight, I called up MIT and asked if they could program the computer to give the vehicle a satisfactory orbit rate. They answered, “Of course. Which way do you want to go and how fast?” In a matter of a couple of days we had a program and a simulator that automatically drove a spacecraft at perfect orbit rate. We got into flight with very little chance to practice or verify, but we put on the cameras and the results were perfect.

POTENTIAL COMPUTER FAILURE

During the development process we ran many simulations of in-flight computer operations with particular concern for in-flight failure. But in the 10 years that I spent in the program there was never a real computer failure. Yet people often wonder what a computer failure would have meant on a mission. It would have depended on the situation and the manner in which the computer failed.

We probably would not have expired, but there were some parts of the mission in which a computer failure would have been especially compromising. Navigation was not necessarily time critical but the lunar landing was very time critical. You could have a situation during a lunar landing in which, if the computer failed, the engine would be driven into the ground. Unless the astronaut could react quickly enough to stop it, the Lunar Module could have been flung on its side. Chances are that the astronaut could prevent such an event by switching to manual control of the vehicle. It must be remembered that the computer had been designed to be as reliable as possible and the astronauts had a great amount of confidence in the machine.

PROBLEMS OF SUCCESS

We had a backup called the entry monitor system, which had a graphic display based on the accelerometers in the spacecraft. With this display the vehicle could be flown manually using pre-drawn curves to be followed for attitude, g-loading, and velocity. It was reassuring to know that we were still able to return to Earth even if the Apollo Guidance Computer failed. During re-entry there was a scroll in the entry monitor system and we could see the computer tracking the predetermined curves all the way to the landing site. As our skills and the computer programs improved over the years of the Apollo program, we came down closer and closer to the carrier waiting to meet us. Finally, by the last Apollo mission, they didn’t park the carrier directly on the landing point. ■

Excerpted by Ben Goldberg from remarks made by David Scott on June 10, 1982 at The Computer Museum in Boston. Reprinted from *The Computer Museum Report*, Fall 1982.

USAF Colonel David Scott flew on the Gemini 8, Apollo 9, and was spacecraft commander on Apollo 15. On the Gemini 8 mission in 1966, Scott and Command Pilot Neil Armstrong performed the first successful docking of two vehicles in space. As Command Module Pilot for Apollo 9 in 1969, Scott helped complete the first comprehensive Earth orbital qualification and verification test of a fully configured Apollo spacecraft. In 1971 Scott commanded Apollo 15, the first extended scientific exploration of the Moon, doubling the lunar stay time of previous flights and using the first Lunar Roving Vehicle to explore the Hadley Rille and the Apennine Mountains. Scott received an MS and an Engineer's Degree in Aeronautics and Astronautics from MIT in 1962.

AGC SPECIFICATIONS

Instruction Set
Approximately 20 instructions;
100 noun-verb pairs, data up to triple-precision

Word Length
16 bits (14 bits + sign + parity)

Memory
ROM (rope core) 36K words;
RAM (core) 2K words

Disk
None

I/O
DSKY (two per spacecraft)

Performance
Approx. Add time: 20µs

Basic machine cycle
2.048 MHz

Technology
RTL bipolar logic (flat pack)

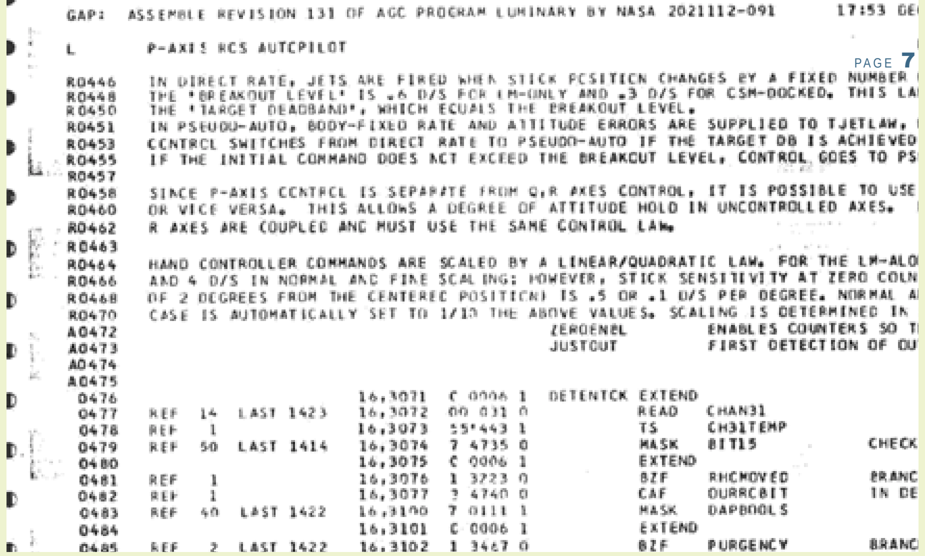
Size
AGC: 24" x 12.5" x 6" (HWD)
DSKY: 8" x 8" x 7" (HWD)

Weight
AGC: 70 lbs; DSKY: 17.5 lbs

Number produced
AGC: 75; DSKY: 138

Cost
Unknown

Power consumption
Operating: 70W @ 28VDC
Standby 15.0 watts



```
GAP:  ASSEMBLE REVISION 131 OF AGC PROGRAM LUMINARY BY NASA 2021112-091      17:53 GE

L        P-AXIS RCS AUTCPILOT

R0446    IN DIRECT RATE, JETS ARE FIRED WHEN STICK PCSITICN CHANGES BY A FIXED NUMBER
R0448    THE 'BREAKOUT LEVEL' IS .6 D/S FOR LM-ONLY AND .3 D/S FOR CSM-DOCKED. THIS LA
R0450    THE 'TARGET DEADBAND', WHICH ECUALS THE BREAKOUT LEVEL.
R0451    IN PSEUDO-AUTO, BODY-FIXED RATE AND ATTITUDE ERRORS ARE SUPPLIED TO TJETLAW,
R0453    CCNTRCL SWITCHES FROM DIRECT RATE TO PSEUDO-AUTO IF THE TARGET DB IS ACHIEVED
R0455    IF THE INITIAL COMMAND DOES ACT EXCEED THE BREAKOUT LEVEL, CONTROL GOES TO PS
R0457
R0458    SINCE P-AXIS CCNTRCL IS SEPARATE FROM Q,R AXES CONTROL, IT IS POSSIBLE TO USE
R0460    OR VICE VERSA.  THIS ALLOWS A DEGREE OF ATTITUDE HOLD IN UNCONTROLLED AXES.
R0462    R AXES ARE COUPLED ANC MUST USE THE SAME CONTROL LAW.
R0463
R0464    HAND CONTROLLER COMMANDS ARE SCALED BY A LINEAR/QUADRATIC LAW. FOR THE LM-ALO
R0466    AND 4 D/S IN NORMAL ANC FINE SCALING; HOWEVER, STICK SENSITIVITY AT ZERO COUN
R0468    OF 2 DEGREES FROM THE CENTEREC POSITICN) IS .5 OR .1 D/S PER DEGREE. NORMAL A
R0470    CASE IS AUTOMATICALLY SET TO 1/10 THE ABOVE VALUES. SCALING IS DETERMINED IN
A0472                                                        ZEROENEL      ENABLES COUNTERS SO T
A0473                                                        JUSTOUT       FIRST DETECTION OF OU
A0474
A0475
 0476                               16,3071  C 0006 1  DETENTCK EXTEND
 0477    REF  14  LAST 1423         16,3072  00 031 0           READ    CHAN31
 0478    REF   1                    16,3073  55*443 1           TS      CH31TEMP
 0479    REF  50  LAST 1414         16,3074  7 4735 0           MASK    BIT15              CHECK
 0480                               16,3075  C 0006 1           EXTEND
 0481    REF   1                    16,3076  1 3723 0           BZF     RHCMOVED           BRANC
 0482    REF   1                    16,3077  3 4740 0           CAF     OURRCBIT           IN DE
 0483    REF  40  LAST 1422         16,3100  7 0111 1           MASK    DAPBOOLS
 0484                               16,3101  C 0006 1           EXTEND
 0485    REF   2  LAST 1422         16,3102  1 3467 0           BZF     PURGENCY           BRANC
```

Even as the Apollo 11 crew—Armstrong, Aldrin, and Collins—were sitting on the launch pad, the only “documentation” on the AGC program was the listing itself, part of which is shown here.

IN THE COLLECTION

Burroughs Corporation Apollo Guidance Computer read only rope memory (1963), XD115.76, Gift of Charles Stark Draper Laboratory

Draper Laboratories Apollo Guidance Computer block 1 components: 3 logic prototypes, 1 finished logic module (1962), X1067.91, Gift of Eldon Hall

Draper Laboratories Apollo Guidance Computer block 2 prototype components: 1 sense amplifier, 2 logic modules (year unknown), X1068.91, Gift of Eldon Hall

MIT Instrumentation Laboratory Apollo memory stack module (1962), X186.83, Gift of Boguslaw Frankiewicz

MIT Instrumentation Laboratory, Raytheon Company, Charles Stark Draper Laboratory Apollo Guidance Computer Prototype Processor-Logic-Interface-Memory modules (1962), X37.81B, Gift of Charles Stark Draper Laboratory

MIT Instrumentation Laboratory, Raytheon Company, Charles Stark Draper Laboratory Apollo Guidance Computer Prototype Universal DSKY Input/Output array (1962), X37.81A, Gift of Charles Stark Draper Laboratory

FURTHER READING

Apollo Operations Handbook, GUIDANCE AND NAVIGATION SYSTEM (G&N), Basic Date: 12 November 1966, http://users.primary.net/~pebecker/apollogc.htm

For a summary of NASA flight computers and software reliability, see: http://www.dfrc.nasa.gov/History/Publications/f8ctf/chap3.html

Hall, Eldon. *Journey to the Moon: The History of the Apollo Guidance Computer*, Washington: American Institute of Aeronautics, 1996.

For an Apollo 8 mission journal, see: http://history.nasa.gov/ap08fj/index.htm

An online version of *Chariots for Apollo: A History of Manned Lunar Spacecraft*, by Courtney G Brooks, James M Grimwood, Loyd S Swenson, published as NASA Special Publication-4205 in the NASA History Series, 1979 can be found at: http://www.hq.nasa.gov/office/pao/History/SP-4205/contents.html

A thorough history of the Apollo Guidance Computer is located at: http://hrst.mit.edu/hrs/apollo/public/

HISTORY MATTERS

BY MICHAEL R WILLIAMS

Michael R Williams is Head Curator at the Computer History Museum.

BUILDING A COLLECTION IN A COMPUTER MUSEUM

The challenges encountered in creating a computer history collection are often different from those found in creating, say, a collection of rare historical science books. For the latter, wide agreement exists as to what constitutes an historic breakthrough and which authors are the fundamental authorities. Computers are, of course, a modern invention and we often do not have the insight to say, with any real confidence, what are the real advances and what are simply derivative embellishments. Additionally, many of the people who worked in the early days of computing are still alive, which makes documenting history both easier and harder. It is only human nature to consider one's own accomplishments to be fundamentally important, which may or may not be the case.

When it comes to creating a collection of relatively modern artifacts, a museum has two basic choices, both of which have advantages. The first is to simply collect everything possible (within certain parameters) and hope that another 15 or 20 years will bring some perspective, allowing curators to weed out unimportant items over time. However, unless the subject is something the size of a postage stamp, storage space simply runs out too soon. The second methodology is to use the best knowledge and intuition in deciding what is or will be important in the future and from the start to limit the items brought into the collection. In this case, some important items will undoubtedly be rejected and impossible to obtain at a later date.

At the Computer History Museum, we have striven over the past twenty years to collect items according to a process of curatorial review centered around the Museum's Collections Committee. We have been fortunate to have generous storage space during this time. However, since the institution's move West, the collection has doubled in size, thanks to an aggressive policy of rescuing important artifacts. Coupled with the storage requirements demanded by the Museum's preservation mission, space is becoming an increasing challenge and will continue to be so, even with a new facility.

When we accept a donation and properly "accession" it through documents that transfer ownership rights to us, we are legally obligated to keep it for a specified period of time. Legislation in this regard was enacted to prevent various unethical groups from accepting potentially valuable donations and selling them on the open market. Here at the CHM we have an additional policy that requires us to keep each item in our collection until the Board of Trustees specifically authorizes the Museum to "de-accession" it, preventing staff members from simply cleaning house on a whim.

Many other considerations arise when evaluating a potential donation. The question of whether an item looks good and would make an interesting exhibit must be balanced against its usefulness in illustrating a particular technology or its status as something of such importance that it must be obtained regardless of its exhibiting potential. One such item would be the Apollo spacecraft guidance computer (see page two), which may not much to look at. But, who wouldn't agree that a device that helped humans get to the moon deserves a place in the Museum?

Another approach can be to compromise—perhaps "hedge our bet" is a better term—by accepting illustrative pieces of something big. For example, we recently decided that we could not accept an entire Fujitsu/Amdahl 5995A (a system 390 class of computer). Instead, we arranged for the donation of sample boards from the CPU and memory sections as well as the fundamental design documentation. This gives us visually and technically interesting items to exhibit as well as information that future historians might want. Additionally, the donor is now investigating the possibility of producing a family tree of all 390 systems—something historians will certainly find interesting.

What the Museum has attempted to do is to develop a philosophy to guide our decision on any particular donation. In essence it states, "we want to have as many of the home runs as possible, and a representative sample of the doubles, base hits, and strike-outs." To accomplish this, the collections department meets once a week to discuss items offered for donation. If the decision is obvious, we make it there and then; for further advice, we consult the Collections Committee, which is composed of members of our Board of Trustees and other experts in the field.

Everyone has a favorite machine and sometimes we must be very diplomatic in declining an offer. However, if anyone knows of an IBM 650 or one of their 700 series of machines we will be happy to consider it at our next weekly collections meeting! ■

To find out how to donate an item, please visit our web page at http://www.computerhistory.org/collections/donateArtifact/ or call Chris Garcia at +1 650 604 2572 for more information.

RECENT DONATIONS

TO THE COMPUTER HISTORY MUSEUM COLLECTION

1940s-era slide rule documentation collection (various dates) X2389.2002, Gift of Herbert F. Spirer

A Computer Perspective (1973), *The Personal Computer Lilith* (1981), X2386.2002, Gift of Ron Mak

APL documentation and ephemera collection (1963-1995), X2393.2002, Gift of Curtis Jones

Apple Macintosh PowerBook 165c and Color StyleWriter 2200 (1993), X2384.2002, Gift of Lynne Engelbert

Atanasoff-Berry Add-Shift Module replica (c. 1995), X2446.2002, Gift of John Gustafson

Bound firing tables for a 155mm M1/M1A1 gun (1942), X2395.2002, Gift of the United States Department of the Army, Aberdeen Proving Ground

Commodore SX-64 Executive portable computer (1985), X2367.2002, Gift of Lee and Mary Long

"Compu-mug" coffee mug (c. 1980), X2364.2002, Gift of Jim Gross

Computer Logic (1964) and Charting Courses (1931), X2392.2002, Gift of Steven Golson

Computer Simulation Applications (1971), X2397.2002, Gift of Julian Reitman

Digital Equipment Corporation document collection, including many *Pocket Service Guide* handbooks (1964-1983), X2394.2002, Gift of Petar Sredojevic

Early computing manuals collection (c. 1960-1980), X2381.2002, Gift of Charles Jortberg

Epson PX-8 laptop computer (1983), X2451.2002, Gift of Chris Illes

Guide to the IBM pavilion, 1964 World's Fair, X2382.2002, Gift of Dag Spicer

Hewlett-Packard Integral Personal Computer (1985), X2369.2002, Gift of Peter Gulotta

IBM 1403 printer music audio tape (1970), X2386.2002, Gift of Ron Mak

IBM advertisements (c. 1950), X2450.2002, Gift of Robert Garner

IBM manual collection (c. 1964-1969), X2398.2002, Gift of Donald Keegan

IBM Models 3494 and 3590 Tape Library Subsystems and Drives (c. 1998), X2399.2002, Gift of University of California, Berkeley, Computer Science Division

IBM software and documentation (various dates), X2391.2002, Gift of Richardson Data Services

Illiac I drum image (CD-ROM) (1952), X2447.2002, Gift of Al Kossow

Inside NETBIOS (1986), X2383.2002, Gift of NASA Ames Library

Laser Computer Inc. pc3 portable computer, software, and manuals (1989), X2390.2002, Gift of Bobby Greenberg

"Laws of Computer Programming" coffee mug (1982), X2365.2002, Gift of Jim Gross

MACTEP (MASTER) personal computer, documentation, and software (c. 1993), X2452.2002, Gift of Serguei Nikolaev

Manual and documentation collection (various dates), X2388.2002, Anonymous Donor

Palm Pilot VII (c. 1998), X2385.2002, Gift of Andrea Butter

Promotional button collection (1970s-1980s), X2451.2002, Gift of Chris Illes

Ricochet Model 21062 wireless modem (1992), X2448.2002, Gift of Karen Mathews

Tano AVT2 Personal/Business Computer, manuals, and software (c. 1985), X2396.2002, Gift of Mark Possof

The Portable Companion collection and related Osborne documentation (1982-1984), X2445.2002, Gift of Leslie Blackwell

Two TRS-80 computer cassettes (c. 1982), X2366.2002, Gift of Jim Gross

Tutorial Description of the Hewlett-Packard Interface Bus (1980), X2387.2002, Gift of T J Forsyth

Various computer science manuals and supercomputer documentation collection (various dates), X2449.2002, Gift of Eugene Miya

Xerox 860 Information Processing System printer wheels and ribbons, documentation, and software library (c. 1980), X2453.20002, Gift of Kenneth G Lehmann

GIFTS OF DAVID BELKNAP

Apple Newton Message Pad (1993), X2357.2002

Apple Newton Message Pad 110 with GPS docking port (1994), X2358.2002

Casio Z-7000 personal digital assistant (1993), X2355.2002

GRiD System Corporation 2260 "Convertible" personal digital assistant (c. 1992), X2359.2002

GRiD System Corporation 2260 "Convertible" personal digital assistant (c. 1992), X2360.2002

GRiD System Corporation Model 2352 PalmPad (1992), X2361.2002

GRiD System Corporation Model 2352 PalmPad (1992), X2362.2002

MicroSlate Datellite 300L personal digital assistant (1991), X2356.2002

NCR Safari 3115 CommStation docking port (c. 1992), X2363.2002

NCR Safari 3115 portable computer (c. 1992), X2363.2002

GIFTS OF MICHAEL PLITKINS

Apple GLM computer system (c. 1984), X2435.2002

Apple IIc Plus computer system (1988), X2433.2002

Apple III computer system (1980), X2437.2002

Apple LISA I prototype computer system (1983), X2436.2002

Apple LISA II personal computer (c. 1984), X2442.2002

Apple Lisa NOS cathode ray tube (c. 1983), X2438.2002

Apple/Franklin floppy disk drive (c. 1978), X2441.2002

Atari 520 ST personal computer system (c. 1985), X2439.2002

Atari 520 ST personal computer system (c. 1985), X2440.2002

Atari 520 STFM personal computer (c. 1985), X2443.2002

IBM 320 POWERserver (c. 1996), X2444.2002

Pixar Image computer in Symbolics SCOPE cabinet (c. 1987), X2434.2002

Sony HB-75AS Hit Bit Home Computer (c. 1985), X2432.2002

(Dates represent dates of introduction and not necessarily dates of manufacture.)

If you would like to update the Museum regarding your artifact donation, please contact Registrar Jeremy Clark at +1 650 604 1524 or clark@computerhistory.org.

BASIC

BY CHRISTOPHER GARCIA

BASIC paper tape. Written by Bill Gates for the Altair 8800, BASIC quickly became the language of choice among hobbyists, and was the first piece of software to be heavily pirated.


Copyright ©2002 True BASIC Inc.

Thomas Kurtz and John Kemeny, co-inventors of BASIC


Students using a PDP-8 based timesharing system

Batch processing dominated the earliest days of computing. A programmer would take a deck of cards he or she had punched off-line, give them to a system operator, and wait, sometimes days, for the results. Obviously, this meant large delays in analyzing and adjusting code, since iterations could not be tested immediately.

The need for systems where multiple users could function as individual operators helped bring about the BASIC language. BASIC, the "Beginners All-Purpose Symbolic Instruction Code," was invented in the early 1960s by two Dartmouth mathematics professors, Thomas Kurtz and John Kemeny, and various Dartmouth students. They wanted to create an easy-to-learn language that could be used on the GE225 timesharing system that Dartmouth was about to launch. This time-sharing system would allow many users to log in at the same time, running programs remotely via terminals in the mathematics and science departments.

Kurtz and Kemeny thought that the most popular languages of the day, including Fortran and ALGOL, were too complex for non-technical users. Using elements from several languages, and adding features such as line numbering that made troubleshooting easier, the two developed BASIC. With just 14 commands in the beginning—including the famous "GOTO"—BASIC could be learned in as little as two learning sessions, creating a tremendous advantage over other languages that could take months to learn.
BASIC may have been the first programming language written for use by non-computer professionals. Many early timesharing systems used BASIC, including those powered by GE machines and DEC PDP-11 systems. BASIC began to show up in many elementary schools around the country, particularly in cities where school districts could use teletypes to get at university mainframe timesharing systems. Children as young as seven years old learned BASIC as part of their curriculum. This early introduction made sure that BASIC would continue to evolve.

When the microprocessor was introduced in the early 1970s, some of the young people introduced to BASIC in elementary schools started building computers from kits and went on to start companies. It should be no surprise that many early microcomputing systems chose BASIC, especially since Kemeny and Kurtz never patented or copyrighted the language. The first BASIC considered to be a full language implemented on a microprocessor was Li Chen Wang's Tiny Basic, which appeared in *Dr Dobbs* magazine in early 1975.

Bill Gates, then a student at Harvard, wrote a BASIC interpreter for the Altair in March, 1975. Microsoft (then Micro-Soft) released their own version on paper tape later in the year, once delivery of Altairs had started. A paper tape was easy to pirate, because it could be run into the computer and a copy could then be punched out.

After this had been occurring for awhile, Bill Gates wrote an open letter to hobbyists (see page 12) claiming that software copying was theft. He stated that this theft had resulted in an income of two US dollars per hour for all the work he and his team had put into BASIC for the Altair. The letter was published in many computer hobby magazines and was the first time people began to contemplate the idea that software sharing was piracy. Some hobbyists believed passionately in free sharing of software, and Gates' letter began to turn some of them against Gates and Microsoft—an attitude that persists even today.

In 1983, BASIC designers Kemeny and Kurtz released their own polished version of BASIC called True BASIC. The two originators claimed that the variants of BASIC released by multiple companies were altering the premises of BASIC, and the "true" BASIC was to be the definitive version. However, it did not sell as well as the other versions on the market, especially those made by Microsoft.

Many new systems used BASIC to introduce people to computing. In the 1980s, the British Broadcasting Corporation (BBC) used a version of BASIC called BBCBASIC (occasionally called BBasiC by the few Americans who knew anything about it) to be used on the BBCMicro, later the Archimedes, and many other British micros. The BBC Micro had been designed as part of a BBC plan to introduce computers to the general population (since to a degree Britain had been lagging behind the US in the percentage of homes and classrooms with computers). The machine and the variant of BASIC are almost unknown in America, though some believe that it could have caught on in the US with a proper introduction. There continues to be a strong group of users who proclaim BBCBASIC to be "the best, most powerful BASIC ever written."

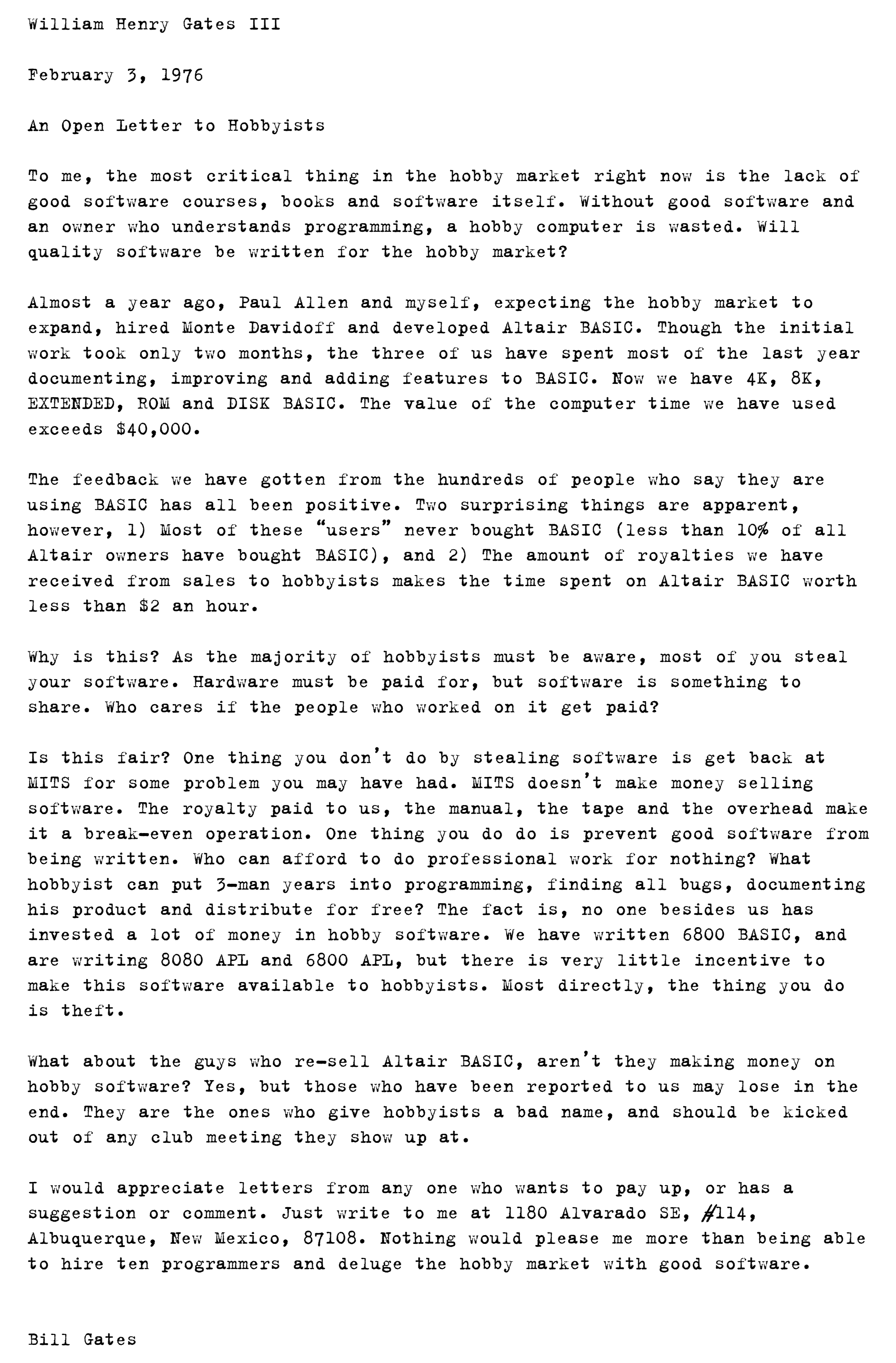
BASIC began to fade from the limelight when languages like C and Pascal were implemented for small machines. The beginning of object-oriented programming and languages like C++ brought a close to BASIC's glory days. The language still exists today in Microsoft's QBASIC and a few other products, and also as Visual BASIC, an object-oriented language developed by Microsoft, though it is less popular than many of the other object-orientated programming languages.

Some people point to BASIC as the "gateway" programming language: it was the first real language to enable the common person to program computers and it ultimately helped to make computer science a discipline of its own. Kemeny passed away in the early 1990s, but Kurtz continues to speak and write about the early days of BASIC. Recently, Kurtz denied the claim that BASIC was the single-most important advancement in the history of programming, commenting, "I'm sorry to say, but I don't think we had much effect…"

Christopher Garcia is Historical Collections Coordinator at the Computer History Museum.

FURTHER READING

Wexelblat, Richard L. *History of Programming Languages*, Academic Press, New York, 1981.



William Henry Gates III

February 3, 1976

An Open Letter to Hobbyists

To me, the most critical thing in the hobby market right now is the lack of good software courses, books and software itself. Without good software and an owner who understands programming, a hobby computer is wasted. Will quality software be written for the hobby market?

Almost a year ago, Paul Allen and myself, expecting the hobby market to expand, hired Monte Davidoff and developed Altair BASIC. Though the initial work took only two months, the three of us have spent most of the last year documenting, improving and adding features to BASIC. Now we have 4K, 8K, EXTENDED, ROM and DISK BASIC. The value of the computer time we have used exceeds $40,000.

The feedback we have gotten from the hundreds of people who say they are using BASIC has all been positive. Two surprising things are apparent, however, 1) Most of these "users" never bought BASIC (less than 10% of all Altair owners have bought BASIC), and 2) The amount of royalties we have received from sales to hobbyists makes the time spent on Altair BASIC worth less than $2 an hour.

Why is this? As the majority of hobbyists must be aware, most of you steal your software. Hardware must be paid for, but software is something to share. Who cares if the people who worked on it get paid?

Is this fair? One thing you don't do by stealing software is get back at MITS for some problem you may have had. MITS doesn't make money selling software. The royalty paid to us, the manual, the tape and the overhead make it a break-even operation. One thing you do do is prevent good software from being written. Who can afford to do professional work for nothing? What hobbyist can put 3-man years into programming, finding all bugs, documenting his product and distribute for free? The fact is, no one besides us has invested a lot of money in hobby software. We have written 6800 BASIC, and are writing 8080 APL and 6800 APL, but there is very little incentive to make this software available to hobbyists. Most directly, the thing you do is theft.

What about the guys who re-sell Altair BASIC, aren't they making money on hobby software? Yes, but those who have been reported to us may lose in the end. They are the ones who give hobbyists a bad name, and should be kicked out of any club meeting they show up at.

I would appreciate letters from any one who wants to pay up, or has a suggestion or comment. Just write to me at 1180 Alvarado SE, #114, Albuquerque, New Mexico, 87108. Nothing would please me more than being able to hire ten programmers and deluge the hobby market with good software.

Bill Gates

Representation of Bill Gates' open letter to hobbyists claiming that software copy was theft

THOMAS KURTZ ON BASIC

INTRODUCTION

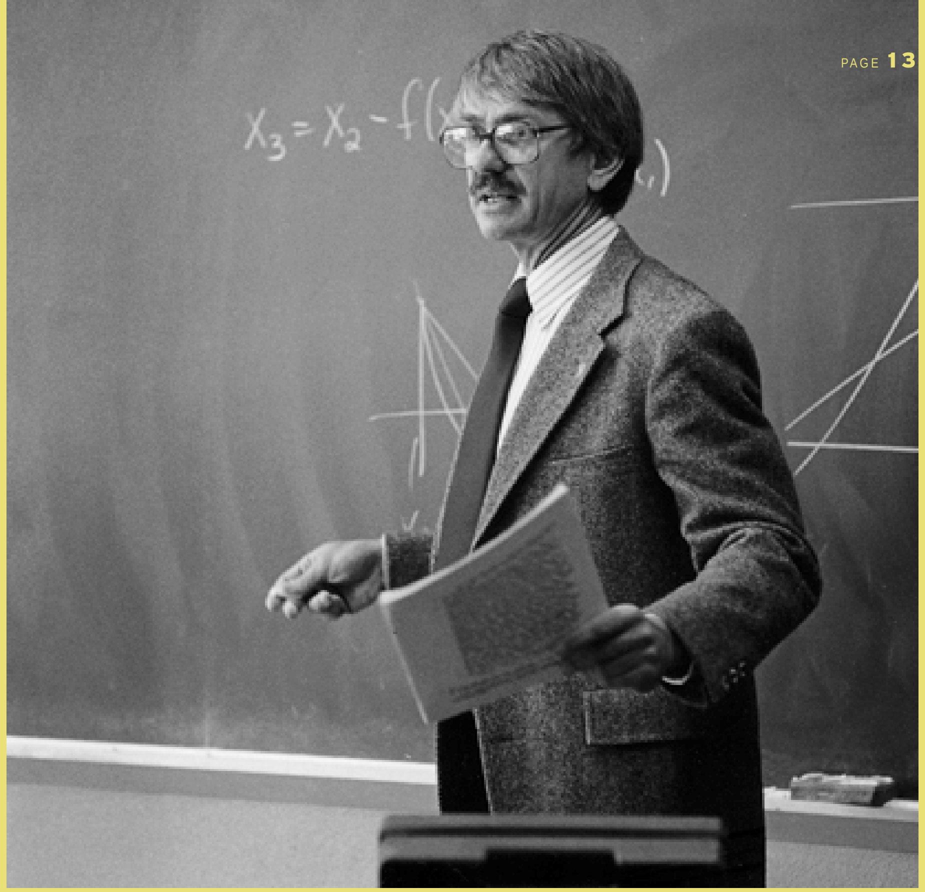
In an email exchange with Computer History Museum Curator of Exhibits Dag Spicer, Thomas Kurtz graciously responded to several questions regarding his experiences with BASIC. Thomas Kurtz and John Kemeny, along with many students at Dartmouth, invented BASIC in the 1960s. Kurtz and Kemeny later wrote a version called True BASIC.

Dag Spicer: In your opinion, what was required to transition from a single-user paradigm to a timeshared paradigm in computing? How did you observe this happen and what, in retrospect, is striking about how it occurred?

Thomas Kurtz: Timesharing was a way to provide many persons with a small amount of computing resources from a single, expensive main frame, each user having the impression that he/she "owned" the whole computer. Remember, 1964 was long before personal computers or microcomputers, and there were only mainframes. Timesharing was a fantastic improvement over punched cards! In other words, there was no paradigm; it was a matter of economics, plus wanting to allow thousands of students at the computer.

DS: Before the arrival of the GE225/235 BASIC timesharing system in 1964, Dartmouth students had access to the school's LGP-30 machine. In your Wexelblat paper you observe that by using this machine, "a good undergraduate could achieve what at that time was a professional-level accomplishment, namely, the design and writing of a compiler." What was Dartmouth's policy regarding getting machine time on the LGP-30? Did these high-level accomplishments surprise you? Why?

TK: Remember that what arrived in 1964 were two machines, the GE-225 (later the 235) and the Datanet-30. Dartmouth undergraduate students built the entire timesharing system with their bare hands! Regarding the LGP-30, at the time we acquired the machine in 1959, there was a crude interpreter called "24.1." What our students did over the next few years was: build a genuine algebraic language processor (in one summer); build a compiler for Algol-60 (actually, a subset of Algol-60); build a load-and-go Algol-like processor for student use (we called it SCALP for Self Contained ALgol Processor); prove a number theory result about the tenth Fermat number; construct a concordance of the works of Wallace Stevens; and on and on. All this was done by undergraduate students in their spare time. I observed that the work done by our students was superior in sophistication and quality to the work done by the industrial users of the same LGP-30.



Professor Thomas Kurtz lectures to his class.

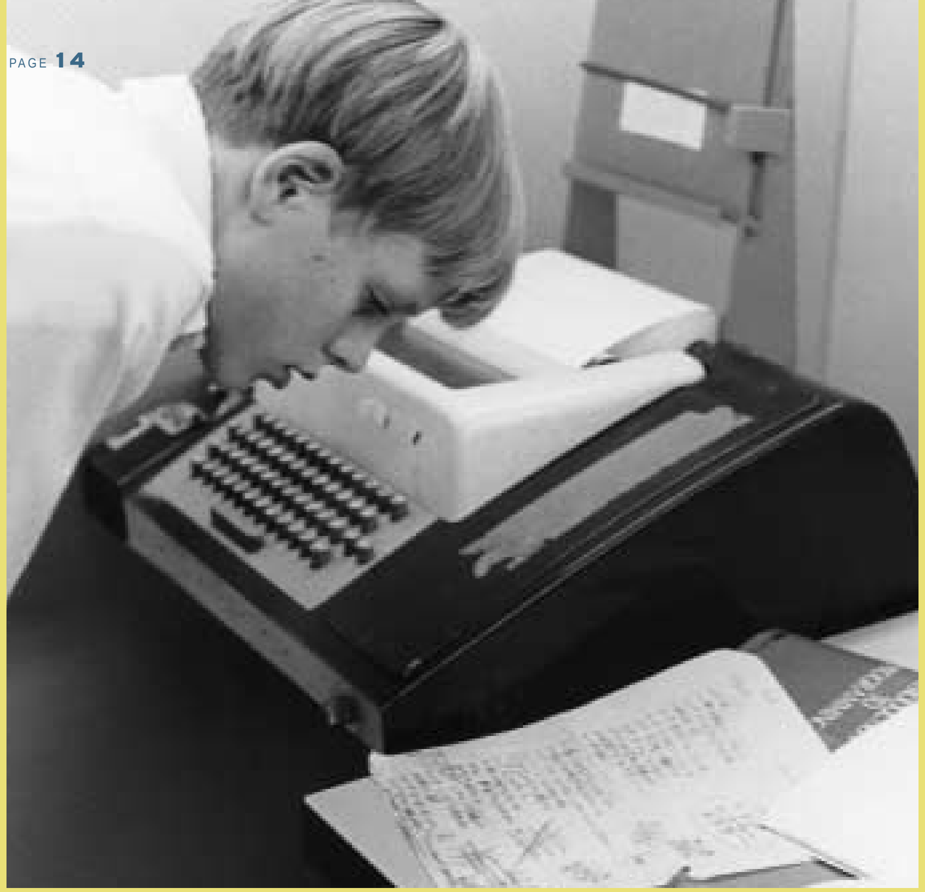
DS: You once said that, "Lecturing about computing doesn't make any sense, any more than lecturing on how to drive a car makes sense." How important was the timesharing metaphor (in contradistinction to batch punch card processing) to your goals for BASIC as a language "for the rest of us?"

TK: Punched cards could not do the job. They were okay for professionals working full time on huge projects, but students (with few exceptions) wouldn't stand for the messing around with keypunches, waiting in line for their job to run, and grappling with the completely unintelligible error messages that came back. They just wouldn't do it. And recall, we were trying to educate ALL Dartmouth students, especially those having major interests in the humanities and social sciences.

Therefore, at the time, timesharing was the only way. BASIC was a part of the solution, being far simpler to understand and use than Fortran or Algol.

DS: Can you explain the relationship between BASIC and GE's Mark I timesharing system and how the relationship helped promulgate BASIC as a standard?

TK: In the fall of 1964 or thereabouts, the GE Service Bureau decided to add Dartmouth timesharing to their existing offerings, which were restricted to punched-card type services. So they hired the two students who wrote the



A student goes over his program in the mid-1970s

One of millions of young students who learned BASIC at an early age

timesharing executive to go to Phoenix to install the Dartmouth timesharing system on similar hardware at the service bureau. Of course, they renamed it the GE timesharing system, Mark I. It was with our blessing, as (1) they had provided a slight bit more than their usual educational discount plus several other non-monetary benefits, and (2) we had no interest in marketing what we had built through a commercial operation. The timesharing system, also called the GE-265, was the basis of the GE service bureau operation for the next ten or so years, and eventually provided them with $100,000,000 in annual revenue. I seem to recall that the GE-265 was replicated in over 50 locations, some of them in the GE Service Bureau, the others in various corporations and in a few school districts.

Thus, BASIC became the most widely used language in the timesharing world, as other vendors “copied” the GE approach on different computers. At one time, there were over 100 companies in the world offering timesharing services, and the vast majority offered some form of BASIC. Thus, when (finally) microcomputers began to appear, the vendors adopted BASIC as being (a) simple, (b) easy to learn, and (c) able to fit in the teeny memories at the time. This then motivated an effort to standardize BASIC in 1974. (It failed, as coming too little, too late.) Gates and Allen wrote one of the first (not the first) BASIC interpreters in 1975. In short, the GE connection was the vehicle that popularized BASIC, which was then picked up by the emerging personal computer industry.

DS: What principles of BASIC do you believe still remain fundamentally important or true? What ideas are so ubiquitous today they no longer feel like BASIC, but nevertheless are/were?

TK: Since most of the users would be casual and occasional users, the language had to be simple and easy to remember. Error messages should be in English and also be suggestive. Beginners should not have to learn fancy stuff of use mainly to experts. These aspects are not present today in most computer applications. The majority of applications are so huge that a casual user must take a course to figure out how to use them. Their desktops are cluttered with so much junk that it is almost impossible to figure things out without studying the manual. And most manuals are atrocious. The computer industry is out for the quick buck, and puts little effort into creating reliable and safe products with readable and useful manuals. The whole strategy is to bring out upgrades on a regular basis in order to establish a revenue stream, each upgrade making the product ever more complicated. The whole virtue of simplicity has been lost!

DS: Why do you think there has been such a proliferation of programming languages since the invention of the stored-program computer some 50 years ago?

TK: My opinion is that all (well, almost all) programming languages are the same, differing only in the spelling of the words, and the clientele for which they are intended. Each new systems programmer that comes along feels he can improve things by inventing a new language. In some cases, a new language was needed because it was intended for a different environment.

Early users of the Dartmouth timesharing system on the GE-225

For example, C was invented as a higher-level improvement for assembly language on Unix machines.

DS: How did RAND’s JOSS (Johnniac Open Shop System) influence you?

TK: John Kemeny had used JOSS at the Rand Corporation, and so had experience with timesharing systems. But we did not adopt JOSS as there were details that we preferred not to use. For example, each JOSS statement ended with a period. Well, periods are the way most folks represent decimal numbers. Also, we wanted to make all internal calculations in double-precision floating point to (a) provide enough accuracy for serious computations, and (b) isolate our users from having to learn about the internal number formats. Other than that, I cannot recall our discussions about JOSS.

DS: You are part of a project to reconstruct the Dartmouth timesharing system. Can you say more about that?

TK: Oddly enough, I didn’t write any of the code. John Kemeny had written a BASIC compiler for the GE-225 using punched cards during the summer of 1963, but didn’t do any coding once the hardware arrived in 1964. (I had written much code for the LGP-30, as he did as well.) It was clear that our students were better at coding than we were. All we did was to supervise the project. Kemeny was 1/12 time as the supervisor of the programming group, but interfered little in their work, except to maintain the main goals, such as simplicity. I was the director of the “center.” We collaborated on the original design of BASIC, and on the additions and improvements that were subsequently made.

DS: You and Dr. Kemeny are heroes to many for your invention of BASIC. Do you have any heroes?

TK: Anyone who makes significant progress toward world peace.

DS: Is there anything you’d like to say about the role of BASIC in the history of computing?

TK: Dartmouth BASIC will be celebrating its fortieth birthday in 2004. It is still around; its current incarnation is True BASIC, which is used in schools and some colleges. While we have used True BASIC to build many serious applications, its chief appeal is that it is simple and easy to use. Plus, there have been no major language changes in the last decades; teachers much prefer continuity, as they don’t want to have to change their teaching materials every year.

We are hot on this project of recreating the Dartmouth timesharing system, circa 1965. One of the then student programmers, Steve Hobbs (formerly of DEC and Compaq, now of Intel), has located assembly language listings of the BASIC compiler and runtime, the Algol compiler and runtime, the 235 exec, and the D-30 exec. We are now in the process of hand transcribing these listings into a machine-readable form. (We tried scanning but that didn’t work. Plus, we have to proofread very carefully anyhow.) As of the moment, the D-30 exec has been transcribed and proofread. The Algol compiler and runtime has been transcribed, but not proofread. Once the code is thus finished, someone will write emulators for the 235 and D-30. When completed, we will actually have a working model of the original (well, one year later) system.

Others who are directly involved in the project are: John McGeachie, who wrote the original GE-235 exec to DTSS and Ron Martin, who took over the code for the D-30 exec (which had been originally written by Mike Busch.) As we progress, I am sure more people will become involved. A start of a website for this project can be found at: http://www.dtss.org. ■

For more information about True BASIC, visit the company website at www.truebasic.com.

REPORT ON MUSEUM ACTIVITIES

BY KAREN MATHEWS

Karen Mathews is Executive Vice President at the Computer History Museum

It is clear—when we think about it—that computer history is created every day. The challenge of preserving and presenting important artifacts and stories of that history is our abiding passion at the Computer History Museum. Much of what we do on a daily basis relates to education and serving the public; researching and planning for the future building and the physical and CyberMuseum exhibits; processing artifact donations; cataloguing and caring for the existing collection; planning and holding programs and events; and of course, raising the funds to continue and advance this important work. In my opinion, it is our great privilege to both facilitate and observe the process of preservation in action.

CHARLIE SPORCK

PUTTING THE SILICON IN SILICON VALLEY

With SEMI (www.semi.org) as our co-host, Charlie Sporck kicked off the Museum's Spring 2002 lecture series on January 16 with his talk, "Putting the Silicon in Silicon Valley: The Birth of the Semiconductor Industry in Silicon Valley," where he relayed fascinating and sometimes surprising personal observations and stories about the people and personalities who brought the semiconductor industry in Silicon Valley into being. Recruited by Fairchild Semiconductor in Mountain View, Calif., Sporck began as a production manager and rose to vice president and general manager. It was during this period at Fairchild that Jean Hoerni developed the planar process and Bob Noyce, the integrated circuit. These innovations, together with the manufacturing equipment and organization, became

Charlie Sporck (left) autographs his book, *Spinoff: A Personal History of the Industry that Changed the World*, after his Museum lecture on January 16.

After leaving Fairchild in the late 1960s, Sporck distinguished himself as CEO of National Semiconductor, where, under his leadership, the company became a multi-billion-dollar giant. With Richard L Molay, Sporck recently co-authored *Spinoff: A Personal History of the Industry that Changed the World*, a book about the Silicon Valley semiconductor industry. Lecture attendee Mike Cheponis remarked, "I really appreciated Charlie Sporck's talk and book. I wish more computer old-timers would do what he's done! It is very nice to see someone like him 'giving back' to the community of preserved history."

JEFF HAWKINS, DONNA DUBINSKY, AND ED COLLIGAN

THE PALMPILOT STORY

Pioneers (left to right) Jeff Hawkins, Donna Dubinsky, and Ed Colligan discuss the roots and challenges of the handheld computing industry.

Andrea Butter, former Palm Computing marketing executive, facilitated a panel discussion with Hawkins, Dubinsky, and Colligan on February 26.

The late 1980s and early 1990s buzzed with corporations and startups trying to develop portable computers that used pens as the means of interaction. By late 1993, every one of these efforts had failed. Though running out of funding, one of these startups, Palm Computing, introduced the Pilot organizer and Palm operating system, which, in turn, launched the handheld computing industry. Last February 26, to an audience of 250, Jeff Hawkins, Donna Dubinsky, and Ed Colligan discussed the roots of handheld computing, how Palm learned from failure, and the challenges of battling conventional technology wisdom. Andrea Butter, former Palm marketing executive and co-author of *Piloting Palm: The Inside Story of Palm, Handspring, and the Birth of the Billion Dollar Handheld Industry*, facilitated the discussion.

In 1994, Hawkins invented the original PalmPilot products and founded Palm Computing. He is often credited as the designer who reinvented the handheld market. As president and CEO of Palm Computing, Dubinsky helped make the PalmPilot the best-selling handheld computer and the most rapidly adopted new computing product ever produced.

Handspring President and CEO Donna Dubinsky autographs Butter's book, *Piloting Palm*, before the panel discussion with colleagues Hawkins and Colligan.

Handspring Chairman and Chief Product Officer Jeff Hawkins (right) shows off the Treo.

Doug Engelbart (left) and Pierluigi Zappacosta prepare for Engelbart's talk in which he reminisced about his lifetime of invention and research.

Attendees record their thoughts during Engelbart's lecture.

It is incredible how much tenacity and determination it took to make this happen. As the vice president of marketing for Palm Computing, Colligan worked with Hawkins and Dubinsky to lead the product marketing and communications efforts for Palm. After their successful run together at Palm Computing, Hawkins and Dubinsky co-founded Handspring in July of 1998 to create a new breed of handheld computers for consumers. Colligan joined Handspring to lead the development and marketing efforts.

Be sure to visit our Visible Storage Exhibit Area and view the PalmPilot prototype on display.

DOUG ENGELBART

OUTRACING THE FIRE: 50 YEARS (AND COUNTING) OF TECHNOLOGY AND CHANGE

Hosted at Microsoft's Silicon Valley Campus on March 26, Doug Engelbart—thinker, inventor, and humanitarian—shared with an audience of 250 some of the influences and struggles behind his life of research. Pierluigi Zappacosta, founder of Logitech and chairman of Digital Persona, facilitated the dialogue.

Although he may be best known for his tangible evidence of productivity—the computer mouse, display editing, outline processing, multiple remote online users of a networked processor, hyperlinking and in-file object processing, multiple windows, hypermedia, context-sensitive help—Engelbart's drive has been to maximize his professional contributions toward helping humankind cope with complex and urgent problems.

Since 1989, he has become the recipient of an extraordinarily long string of awards, including the Lemelson-MIT Prize of $500,000, and the National Medal of Technology in 2000. Still to be recognized is that Engelbart's technological accomplishments are but part of his humanitarian career. Said lecture attendee Susan Nycum, "My impressions are that Doug is, as always, looking ahead and impatient with looking behind—even at his own accomplishments. [This is] something he shares with all the 'young for their age' senior superstars I know."

250 people attended the Engelbart event.

Our host and Microsoft's general manager of cable services, Colin Dixon, said, "I think the most magical moment for me… was when Doug mentioned, almost offhandedly, an invention he made during the war. He described how he held a tube of electro-luminescent gas up against an antenna he was trying to tune. When he had the power set just right, the gas in the tube glowed most intensely. It was a fascinating glimpse into the mind of a consummate inventor." Engelbart continues to propagate his ideas through his Bootstrap Institute. Additional background information is available at www.bootstrap.org.

CHARLIE BACHMAN

ASSEMBLING THE INTEGRATED DATA STORE (IDS)

On April 16, Charlie Bachman, winner of the ACM Turing Award and Distinguished Fellow of the British Computer Society, described the circumstances under which the first database management system (DBMS) came into being. In 1960, General Electric was desperate to computerize their manufacturing systems, without each of 100 departments inventing their own solution. Bachman and others at GE set out to solve the problem. By 1964 they had created and put into production a generic manufacturing system (MIACS), a transaction-oriented operating system, and the first database management system (Integrated Data Store, or IDS), all running on an 8K GE 225 computer. IDS was a unique combination of existing software technologies: virtual

Charlie Bachman discussed his experiences in developing the first database management system, the Integrated Data Store.

memory, blocked records, list processing, data descriptions, self identifying records, data manipulation language, recovery and restart, etc., and was the first disk-based database management system used in everyday production. Among other things, Bachman was also responsible for developing data structure diagrams (ER diagrams), commonly known as Bachman diagrams, as graphical representations of semantic structures within the data.

In April 1983, Bachman Information Systems, Inc. was created to commercialize Computer Aided Software Engineering (CASE) concepts, which he developed while at Honeywell and Cullinet. In 1991 the company went public, and in 1996, merged with Cadre Technology, Inc., to form Cayenne Software, Inc. Bachman's IDS and CASE products are still alive under the CA banner. Today, Bachman is a consultant and is currently working on a book about the story of the development of IDS.

STEVE RUSSELL AND NOLAN BUSHNELL

SHALL WE PLAY A GAME? THE EARLY YEARS OF COMPUTER GAMING

From their humble beginnings in the 1960s as demonstrations of computer interactivity, computer video games have become a major part of popular culture in America, Japan, Europe, and elsewhere. On May 7, Stephen "Slug" Russell, inventor of the early computer game SpaceWar!, and Nolan Bushnell, designer of Computer Space and founder of Atari, shared their personal stories, starting from the days when computer games were played on mainframes. Stewart Brand, publisher of the original *Whole Earth Catalog* and president of The Long Now Foundation, moderated this fascinating discussion about the advent of the modern gaming age.

Video game fans gathered to celebrate the 40th birthday of Spacewar! and the 30th birthday of PONG.

Slug Russell, Bill Pitts, Steve Golson, and Nolan Bushnell (left to right) enjoyed the rare opportunity to play the Galaxy game, which was developed by Pitts and based on Spacewar! Find Spacewar! online at: http://agents.www.media.mit.edu/groups/el/projects/spacewar/

Hanging out together at the model railroad club and inspired by the writings of sci-fi author E.E. "Doc" Smith, Russell and his team of programmers at MIT worked to create SpaceWar! in 1962. "The space program was peaking at the time and people didn't have much sense of what it might be like to steer the spacecraft," said Russell. "I was into realism and really trying to teach people what flying in space was all about."

SpaceWar! was created on a Digital Equipment Corporation (DEC) PDP-1, an early "interactive" mini-computer that used a cathode-ray tube display and keyboard input. The computer was a donation to MIT from DEC, which hoped MIT's think tank would be able to do something remarkable with its product. A game was possibly the last thing the company expected. But Russell's SpaceWar! showed that fun could be a driving force in the advancement of computer technology. It influenced companies like Atari and others in creating a powerful new entertainment medium.

Delighted fans Cassidy Nolen and Nicole Servais with Nolan Bushnell's autograph.

As a youth in Salt Lake City, Bushnell worked in the games department of an arcade. He first encountered SpaceWar! on an IBM machine in the mid 1960s and describes himself at the time as "truly obsessed with the game." Bushnell co-founded Atari in 1972 and after four years of financial struggles, the company was purchased by Warner Communications. It had become "part of the Atari culture to get to the bank first with your paycheck," Bushnell admitted. Having brought PONG to the masses, Bushnell is justifiably revered as the "Father of the Video Game Industry."

TOURS AT THE MUSEUM BRING PEOPLE TOGETHER

You never know whom you will run into at the Museum's Visible Storage Exhibit area—nor what you will learn about them. For example, Jamis MacNiven, owner of the famed Buck's Restaurant of Woodside, California (where hundreds of businesses have been founded over breakfast), recently organized a tour for some of his friends. His guests included: Brian Carlisle, founder of the robotics firm, Adept Technology, where the Milano Cookies are assembled; venture capitalist Paul Dali; Reid Dennis, founder of Institutional Venture Partners and pilot of a 50-year-old airplane that he restored and flew around the world; Kevin Kelly, co-founder of *WIRED* Magazine and outspoken optimist for the coming new age of interconnectivity; Jacques Littlefield, who has an impressive operation in Woodside to collect and restore army tanks from around the world; Bill Peacock, venture capitalist and former assistant secretary of the United States Army; networking pioneer and entrepreneur Larry Roberts; Dennis Taylor, managing editor of *Silicon Valley Biz Ink*; Meihong Xu, venture capitalist with Möbius and formerly an intelligence officer in China; and Steve Zelencik, senior vice president at Advanced Micro Devices and a great finder of computer artifacts himself.

(left to right) Jacques Littlefield, Brian Carlisle, Steve Zelencik, Reid Dennis, Meihong Xu, Bill Peacock (behind), Len Shustek (behind), Kevin Kelly, and Larry Roberts converse in the Museum's Visible Storage Exhibit Area.

Bill Peacock, Jacques Littlefield, and Jamis MacNiven with Museum Curator of Exhibits Dag Spicer at a special tour arranged by Buck's Restaurant owner MacNiven.

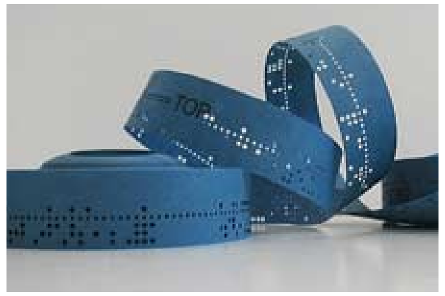
Why not organize a tour for your friends? Contact Kelly Geiger at +1 650 604 0345 to make arrangements.

COLLECTION CONTINUES TO GROW

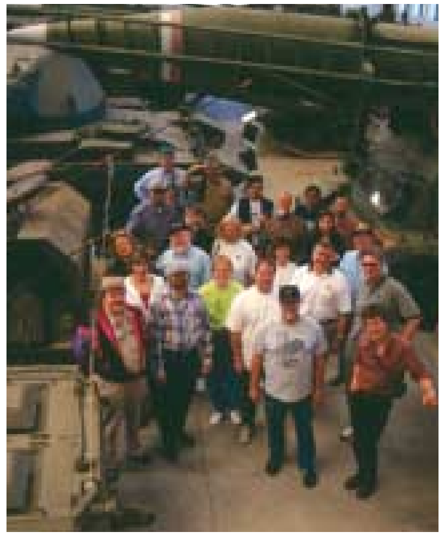
Among the many items recently donated to the Museum's collection (see page nine), the following are particularly noteworthy. A replica of an add-shift module from the Atanasoff-Berry Computer (ABC) replicates in exact detail the circuitry and components used in the original ABC from 1937. While the machine was not a direct progenitor of the modern stored program digital computer, it played a key role in a decades-long lawsuit over the official "inventor" of the digital computer, a legal battle that Atanasoff eventually won.

Secondly, the U.S. Army's Aberdeen Proving Ground donated an original World War II Artillery Firing Table, precisely the type of table the production of which was the impetus for the design and construction of the ENIAC, the United States' first electronic computer. Gunners used the 1942 booklet of tables to properly guide their artillery shells to their targets. It was the long process of calculating these tables by rooms full of human "computers" that led the Army to consider an automated method of production. ENIAC, though completed after the war, was still used to calculate firing tables but also played a major role in the development of the hydrogen bomb.

Finally, Al Kossow donated an ILLIAC I drum image: a snapshot of the actual bit patterns stored on the computer's drum memory (delivered on paper tape). The ILLIAC I, a vacuum tube machine completed in about 1952, was a direct descendant of the famous IAS (Institute for Advanced Study) machine designed by John von Neumann—the prototype of the modern stored-program, binary, parallel, digital computer. This acquisition helps the Museum fulfill its mission of preserving not just hardware, but software as well, and is an exciting find from the "prehistoric" era of the modern computer.



A paper tape of the ILLIAC I drum memory was recently donated to the Museum by Al Kossow.



On March 30, Museum volunteers and staff visited Jacques Littlefield's Tank Farm in Portola Valley.

VOLUNTEERS VISIT TANK FARM

About 30 Museum volunteers and staff went on a field trip on March 30 to Pony Tracks Ranch in Portola Valley to see Jacques Littlefield's tanks and the Military Vehicle Technology Foundation organization. Curator Roy Robertson showed us 150 of the nearly 200 tanks held on the site. Most of them are operable and many have been restored to combat-ready appearance and operating condition. We are always interested in seeing how other organizations collect, restore, preserve and present their collections. The foundation is doing an impressive job. ■

THANKS TO OUR ANNUAL DONORS

We acknowledge with deep appreciation the individuals and organizations that have given to the Annual Fund.

CORE BENEFACTORS
64K (65,536)+
Gwen & C Gordon Bell
L John Doerr & Ann Doerr
Donna Dubinsky
Elaine & Eric Hahn
Jeff Hawkins & Janet Strauss
Gardner Hendrie & Karen Johansen
David & Karla House
Janice & Bob Lisbonne
Leonard J Shustek

32K ($32,768)+
Eric Brewer
Bill & Roberta Campbell
Grant & Dorrit Saviers
John & Sheree Shoch

16K ($16,384)+
Steve Blank & Alison Elliott
Mike & Kristina Homer
The Krause Foundation
John Mashey & Angela Hey
Eric Schmidt

CORE INVESTORS
8K ($8,192)+
Vinton & Sigrid Cerf
Andrea Cunningham
Dave Cutler
Steve & Michele Kirsch Foundation
Ike & Ronee Nassi
Bernard L Peuto
Allen Rudolph
Sigma Partners

4K ($4,096)+
Sally M Abel & Mogens Lauritzen
Peggy Burke
Eric D & Marilyn G Carlson
Christine Hughes & Abe Ostrovsky
Shawn & Doug MacKenzie
Gordon E & Betty I Moore
John & Elizabeth Toole

2K ($2,048)+
Allan Alcorn
Bruce G & Leona D Baumgart
Yogen & Peggy Dalal
Carol & Chris Espinosa
Federico & Elvia Faggin
Robert B Garner
The Bill & Melinda Gates Foundation
Charles & Nancy Geschke
Rob & Yukari Haitani
Donald & Jill Knuth
Jim & Stephanie Nisbet
James N Porter
The Rockwell International Corporation Trust
Dave & Jan Rossetti
Jean E Sammet
Alvy Ray & Zu Smith
Rich & Cindy Tennant

1k ($1,024)
Adobe Systems - Matching Gift Program
Frances Allen
Gene & Marian Amdahl
David & Robin Anderson
Austrade
The Avram Miller Family Foundation
John & Sheila Banning
Craig & Barbara Barrett
Allen Baum & Donya White
Barry & Sharla Boehm
Gary Boone
Jack & Casey Carsten
Ned Chapin
Bruce & Gail Chizen
Cisco Foundation
Richard J Clayton
The Computer Language Company Inc
R T Coslet
Lori Kulvin Crawford
Davidow Foundation
Eleanor & Lloyd Dickman
Whitfield Diffie & Mary Lynn Fischer
Les & Marian Earnest
David Emerson
Judy Estrin & Bill Carrico
Irwin & Concepción Federman
Edward A Feigenbaum & Penny Nii
Bruce Fram
Frank-Ratchye Family Foundation
Fujitsu Laboratories of America, Inc
Forrest Gunnison
Trip Hawkins
John & Andrea Hennessy
Mary Henry & Rajpal Sandhu
Chuck & Jenny House
Tim & Nancy Howes
IBM International Foundation
IEEE - Hot Chips Symposium 2001
Matthew B Ives
Dr & Mrs Leonard Kleinrock
Tom Kopec & Leah Carneiro
Richard & Ellen Lowenthal
Ron & Deb Marianetti
James A Markevitch
Frank & Judith Marshall
Karen Mathews
James McElwee
Malachy Moynihan
Joan & Stanley Myers
Donald & Helen Nielson
Jeffrey Berg & Debra Paget
Max Palevsky
Paul Pierce
Susan Poduska & John W Poduska
Bill & Shelly Pratt
Dennis Ritchie
Toni & Arthur Rock
Peter & Valerie Samson
Robert Shaw
Virginia Shuler
Lee & Robert Sproull
Stephen Squires & Ann Marmor-Squires
Edward Taft
Edward Thelen
Larry & Dawn Weber
Kayton & Steven Weinstein
Raymond & Carol Williams
Paul Winalski
Anthony J Wood
Warren Yogi
Robert & Carrie Zeidman
Cindy & Peter Ziebelman

GENERAL SUPPORTERS
Anonymous
Donald & Nancy Alpert
Saul & Irene Amarel
John Amos
Judith & Curt Anderson
Melissa Anderson & Howard Look
Paul & Joan Armer
Mary Artibee
The Dennis & Janet Austin Fund
Connie & Charlie Bachman
John Backus
Joseph Barrera
Sandy & Ann Benett
Leslie Berlin
Paul Berry
Doris & Alfred Bertocchi
Mohan Bethur
Lyle Bickley
Philip Blanchar
Michael & Sharon Blasgen
Blumberg Capital Management
Philippe Bouissou
Stuart Bowen
Richard Brand
Ron & Margaret Brender
Frederick & Nancy Brooks
John & Doris Brown
Werner Buchholz
Jack Burness
Bruce & Janet Burns
Andrea Butter
Ruth Carranza & Pamela Walton
John Chang
Pamela Cleveland
Richard & Dorene Cohen
Nancy & Thomas Colatosti
Gordon Collins
George Comstock
Michael Coulter
Deborah & Michael Cussen
William Danielson
Naren Dasu
Alistair Davidson
Peter & Dorothy Denning
Lena M Diethelm
Michael diVittorio
Lee & Daniel Drake
Mark Duncan
John Dykstra
John Ehrman
Stephen Ellis & Panos Aglaia
David Ellsworth
Jack Esbin
Thelma & Gerald Estrin
Maria & Bob Evans
Douglas G Fairbairn
Alan Scott Fitz
Rita Foley
Shawn Ford
Jim Forster
John & Wendy-Ann Francis
Barbara & Joseph Fredrick
Jim Fruchterman
Chris Garcia
George Glaser & Karen Duncan
Bob & Dee Glorioso
Gary M Goelkel
Arlene & Earl Goetze
Robert E Goldberg
Bert Graeve
Mark Graham
Philip Gregory
Douglas Greig
Matthew Hamrick
Rollin C Harding
Ann Hardy
Norman Hardy
Roy & Virginia Harrington
Alys Hay
Glen B Haydon
Dan Hill
Winston Hindle
Thea Hodge
James Hurd
Joseph Impellizeri
Joanne & Irwin Jacobs
Dina & Neil Jacobson
Luanne Johnson
Curtis Jones & Lucille Boone
Chuck Kaekel
Brewster Kahle
Robert Kahn & Patrice Lyons
Marlene & Jeffrey Kalb
Laurel & Ray Kaleda
Mark Kaminsky
Christopher A Kantarjiev
Randy Katz
Yumi & Tom Kelley
Tabinda Khan
Tracy Holloway King
Thomas & Mary Kornei
Daniel Kottke
Ed Kramer
Winston Kriger
Philip Kurjan
Richard & Joanne Kurkowski
Thomas Kurtz
Larry Kwicinski
David & Grayson Lane
Cecilia A Larsen
Kenneth Larsen
John L Larson
Karl Lautman
David A Laws
Roy Kwok Ming Lee
John V Levy
Jefferson Lilly
Joyce Currie Little
Carl & Claudia Lowenstein
Slava & Hana Mach
Walt Main
Michael Malcolm
Milt Mallory
John Maloney & Roxanne Guilhamet Maloney
Julius Marcus
Bill & Sandra Martin
Connie Martinez
George Maul
Terry Mayer
Stanley & Maurine Mazor
Frank McConnell
Russell McHugh
Stuart McHugh
William D & Dianne Mensch, Jr
Philip Menzies
David Miller
Celia & Gary Miller
Charlene Miyashita
W E & Sharon Moerner
Michael Morganstern
Edward Munyak
Ronald Nicholson
Marilee J Niemi
Landon Noll
Mike & Betsy Noonen
Arthur Norberg
David Novak
Dave Olson
Donn B Parker
Jeff Parker & Barbara Waddy
Rich Pasco
Doug & Shirley Pearson
Jerry Lee Perkins
S Michael Perlmutter
Michael Pique
Arati Prabhakar & Patrick Windham
Robert Praetorius
Jane & Bob Puffer
Donald & Sandie Pugh
James Quinn
Sandhya Ramanathan
Carol Randall
Glenn Ricart
David Richey
Annie Roe-Rever
Heidi Roizen & David Mohler
Lynn & George Rossmann
Dick Rubinstein
Phillip Rupp
Kathleen L Rydar
June & David Rynne
Paul Saffo III
John & Linda Sailors
Rita Seplowitz Saltz
Rex Sanders
John & Christine Sanguinetti
Marisa & Werner Schaer
Michael & Wyn Schuh
Gail Schure
Matthew & Melissa Shafer
Chris Sheedy & Marsha Brewer
Dick Shoup
Thomas Siekman & Pamela Kenney
Dan & Karon Siewiorek
Michael P Simon
Chris & Jade Simonson
Rebecca Elizabeth Skinner
Eric Smith
Sally & Dick Smith
Joseph & Sally Smith
Frank Snow
Matt Chew Spence
Daryl Spitzer
David & Shirley Stackpole
Larry Staley
David Stearns
Steven Stepanek
Peter Stewart
Studio Mobius
Richard Swan & Claudia Mazzetti
Dr Bradley S Tice
Ilene Chester & Frank Tobin
Fritz & Nomi Trapnell
Joseph Traub & Pamela McCorduck
Stephen Trimberger
Richard & Pamela Tucker
United Way of King County
Teruo Utsumi
The Vanguard Group Foundation
Daniel Wade
Duane & Lorna Wadsworth
Floy & Willis Ware
Michael Weaver
John Weirich
Carol Welsh
Marguerite & James Wengler
Gio & Voy Wiederhold
W Roger B Willis
Duane Wise
Jon & Marsha Witkin
Jim & Sylvia Work
Ko Yamamoto
Wai Chee Yee
Robert Yeh
Bill Yundt
John G Zabolitzky
Martin Zam
Maria D Zorsky
Maureen & John Zuk
George D Zuras

This information is current as of May 1, 2002. Please notify us of any changes to your listing (liska@computerhistory.org). Thank you.

COMPANIES PLAY CRITICAL ROLE IN PRESERVATION

Your company has played a critical role in the computer industry; you spent nights sleeping underneath your desk and an 80-hour work week was average. Now it's time for you to help preserve the history you created by becoming a corporate member of the Computer History Museum.

Corporate members join the Museum on an annual basis, and enjoy many advantages and exclusive privileges for the critical support they provide.

Through this program, your company will be associated with the Museum's most visible and significant activities.

Contributions play an essential role in guaranteeing the future success of the Computer History Museum, and helping us to continue our work collecting the artifacts and human stories of computing history.

The items we seek and the pioneers of the industry are disappearing; we need your help to preserve this piece of history now.

For further information please contact David Miller, vice president of development, at 650.604.2575 or miller@computerhistory.org.

MUSEUM SEEKS DIRECTOR OF INDIVIDUAL GIVING AND MAJOR GIFTS

The Computer History Museum has an immediate opening for a director of individual giving and major gifts. As a member of the development team, the director is responsible for the Museum's annual fund program and goals and serves as major gifts officer for the Museum's capital campaign.

For more information please visit www.computerhistory.org/jobs.

UPCOMING EVENTS

Please RSVP for all events and activities by calling +1 650 604 2714 or visiting www.computerhistory.org/events. Thank you!

TUE, MAY 21
THE HISTORY AND FUTURE OF ELECTRONIC PHOTOGRAPHY
Carver Mead, Foveon, Inc.
MEMBER RECEPTION: 6:00 PM
LECTURE: 7:00 PM
AMD, Commons Building
Sunnyvale, California

TUE, JUNE 4
EARLY TECHNOLOGY MARKETING EFFORTS: AN EVENING WITH REGIS MCKENNA
Regis McKenna, The McKenna Group
MEMBER RECEPTION: 6:00 PM
LECTURE: 7:00 PM
Xerox PARC Auditorium
Palo Alto, California

THU, SEPTEMBER 5
HALF A CENTURY OF DISK DRIVES AND PHILOSOPHY: FROM IBM TO SEAGATE
Al Shugart, Al Shugart International
MEMBER RECEPTION: 6:00 PM
LECTURE: 7:00 PM
Xerox PARC Auditorium
Palo Alto, California

TUE, OCTOBER 22
FELLOW AWARDS BANQUET
Fairmont Hotel, Imperial Ballroom
San Jose, California

TUE, NOVEMBER 12
JOHN WARNOCK AND CHUCK GESHKE
Adobe Systems, Inc.
MEMBER RECEPTION: 6:00 PM
Building 126
LECTURE: 7:00 PM
Moffett Training and Conference Center
Building 3
Moffett Field, California

TUE, DECEMBER 10
STEVE WOZNIAK
MEMBER RECEPTION: 6:00 PM
Building 126
LECTURE: 7:00 PM
Moffett Training and Conference Center
Building 3
Moffett Field, California

TOUR THE MUSEUM

Tours of the Museum's Visible Storage Exhibit Area are normally held on Wednesdays and Fridays at 1:00 p.m. and the first and third Saturdays of each month at 1:00 p.m. and 2:00 p.m. For tour registration call +1 650 604 2579.

VOLUNTEER OPPORTUNITIES

The Museum tries to match its needs with the skill and interests of its volunteers and relies on regular volunteer support for events and projects. In addition to special projects, monthly work parties generally occur on the second Saturday of each month, including:

JUNE 8, JULY 13, AUGUST 10, SEPTEMBER 14, OCTOBER 12

Please RSVP at least 48 hours in advance to Betsy Toole for work parties, and contact us if you are interested in lending a hand in other ways!

For more information, please visit our volunteer web page at www.computerhistory.org/volunteers

CONTACT INFORMATION

EXECUTIVE STAFF

JOHN TOOLE
Executive Director & CEO
+1 650 604 2581
toole@computerhistory.org

KAREN MATHEWS
Executive Vice President
+1 650 604 2568
mathews@computerhistory.org

DAVID A MILLER
Vice President of Development
+1 650 604 2575
miller@computerhistory.org

MICHAEL R WILLIAMS
Head Curator
+1 650 604 3516
williams@computerhistory.org

FULL-TIME STAFF

JEREMY CLARK
Registrar
+1 650 604 1524
clark@computerhistory.org

PAM CLEVELAND
Event Manager
+1 650 604 2062
cleveland@computerhistory.org

WENDY-ANN FRANCIS
Office Administrator
+1 650 604 5205
francis@computerhistory.org

CHRIS GARCIA
Historical Collections Coordinator
+1 650 604 2572
garcia@computerhistory.org

DAPHNE LISKA
Development Associate
+1 650 604 3470
liska@computerhistory.org

JACKIE McCRIMMON
Executive Assistant
+1 650 604 5145
mccrimmon@computerhistory.org

DAG SPICER
Curator of Exhibits
+1 650 604 2160
spicer@computerhistory.org

CATRIONA SWEENEY
Development & PR Associate
+1 650 604 5133
sweeney@computerhistory.org

KIRSTEN TASHEV
Building & Exhibits Project Manager
+1 650 604 2580
tashev@computerhistory.org

MIKE WALTON
Director of Cyber Exhibits
+1 650 604 1662
walton@computerhistory.org

KARYN WOLFE
Special Projects Manager
+1 650 604 2570
wolfe@computerhistory.org

PART-TIME STAFF

JENNIFER CHENG
Event and PR Intern
+1 650 604 2714
cheng@computerhistory.org

LEE COURTNEY
Volunteer Coordinator
courtney@computerhistory.org

KELLY GEIGER
Administrative and Accounting Intern
+1 650 604 2579
geiger@computerhistory.org

SOWMYA KRISHNASWAMY
Database Services Intern
+1 650 604 2579
krishnaswamy@computerhistory.org

TANYA PODCHIYSKA
Web Services Intern
+1 650 604 2070
podchiyska@omputerhistory.org

KATHY VO JOZEFOWICZ
E-Commerce Intern
+1 650 604 2577
jozefowicz@computerhistory.org

BETSY TOOLE
Hospitality & Facilities Support
+1 650 604 2567
etoole@computerhistory.org

ROBERT YEH
Administration and Accounting Intern
+1 650 604 2067
yeh@computerhistory.org

JOHN J VILAIKEO
Tech Suport Intern
+1 650 604 4962
vilaikeo@computerhistory.org

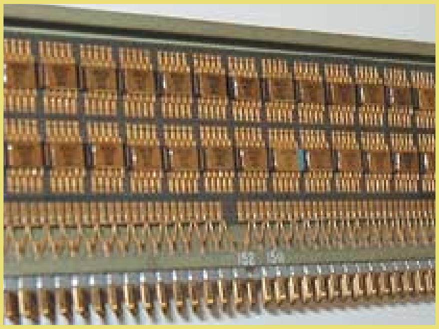
COMPUTER HISTORY MUSEUM
Building T12-A
Moffett Field, CA 94035, USA
+1 650 604 2579
+1 650 604 2594 (fax)
or
COMPUTER HISTORY MUSEUM
PO Box 367
Moffett Field, CA 94035, USA

WWW.COMPUTERHISTORY.ORG

Current staff openings can be found at www.computerhistory.org/jobs.

MYSTERY ITEMS

FROM THE COLLECTION OF
THE COMPUTER HISTORY MUSEUM



MIT Instrumentation Laboratory, Raytheon Company, Charles Stark Draper Laboratory Apollo Guidance Computer Prototype Processor-Logic-Interface-Memory modules (1962), X37.81B, Gift of Charles Stark Draper Laboratory

Explained from CORE 3.1

APOLLO GUIDANCE COMPUTER LOGIC MODULE PROTOTYPE

Shown here is a prototype logic module from the Apollo Guidance Computer (AGC) currently on display at the Computer History Museum. The AGC was a 70 lb. box of integrated circuitry (with attached control panel) that performed real-time guidance and control and served as a lifeline to American astronauts descending to the lunar surface in 1969.

Spanning nearly a decade of development, the AGC began in about 1961 as a research project at the MIT Instrumentation Lab in Cambridge, Massachusetts. It was built by Raytheon and used approximately 4,000 discrete integrated circuits from Fairchild Semiconductor.

The Apollo Guidance Computer program was a landmark both in terms of hardware design and software management and laid the foundation for SpaceLab and shuttle computer systems development. The speed, power, and size requirements for the AGC pushed along an entire industry that was just taking its first steps along the breathtaking curve of Moore's Law.

See page two for more information about the AGC. ■

WHAT IS THIS?

THIS ITEM WILL BE EXPLAINED IN THE NEXT ISSUE OF CORE.

Please send your best guess to mystery@computerhistory.org before 07/15/02 along with your name, shipping address, and t-shirt size. The first three correct entries will each receive a free t-shirt with the new Museum logo and name.

COMPUTER HISTORY MUSEUM

PO Box 367, Moffett Field, CA 94035 USA

Address Service Requested

NONPROFIT ORG
U.S. POSTAGE
PAID
MOUNTAIN VIEW, CA
PERMIT NO. 50