Chapter 3

Computer-Aided Design's Strong Roots at MIT

During World War II, the Massachusetts Institute of Technology became a significant research and development partner with the United States military establishment in Washington. One such activity was the Servomechanism Laboratory which was founded in 1940 by Professor Gordon Brown who would eventually become dean of the Institute's school of engineering. Jay Forrester, a key participant in what follows, enrolled at MIT in 1939 as a graduate student was soon involved in the Lab's efforts to develop feedback control mechanisms for military equipment such as shipboard radars and gun mounts. During the war, this activity took Forrester to the South Pacific where he spent time on the aircraft carrier Lexington helping to repair prototype radar systems. While Forrester was on board the Lexington, it was torpedoed but not sunk.

Whirlwind - test bed for key technologies

In 1944, the Servomechanism Laboratory or the Servo Lab as it was typically referred to, began working on the development of computer systems in support of the Navy's Airplane Stability and Control Analyzer (ASCA) project under a $75,000 contract. The objective was to create a general purpose flight simulator as compared to the then current practice of building custom flight simulators for each aircraft type. At the time, most simulation work involved analog computers and this was the initial plan for ASCA. By 1946, Forrester became convinced that digital computers along the lines of the ENIAC machine which had recently been completed at the University of Pennsylvania, would provide a better platform for aircraft simulation. This led to the establishment of Project Whirlwind with the objective of building MIT's first digital computer, Whirlwind I. There never was a Whirlwind II and most references to the I were dropped by 1950 or so.[1]

Within a couple of years, work on ASCA was shelved and the group's focus was to build the world fastest and most reliable digital computer. Whirlwind is important to the development of Computer-Aided Design (CAD) technology for several reasons. Because of the original intent to use this computer as the control element for a flight simulator, it was designed from the start to be capable of real-time operations. In turn, this led to the utilization of interactive devices for operator communication with the computer. One such device was a Cathode Ray Tube (CRT) display console as shown in Figure 3.1. This early interest in display technology would eventually lead to the development of more advanced graphics terminals by MIT's Lincoln Laboratory, a successor to the Institute's Digital Computer Laboratory, as part of the TX-0 and TX-2 computer systems described below.

In developing Whirlwind, Forrester and his assistant, Robert Everett, had to tackle two major issues. The first was how to achieve substantially better system reliability with

---

3.1   [1] *Funding a Revolution: Government Support for Computing Research* - Computer Science and Telecommunications Board, 1999

a computer that utilized many thousands of relatively unreliable vacuum tubes. The concept they came up with was to use a marginal checking technique where engineers could vary the system's voltages to detect vacuum tubes that were on the verge of failing. It added substantial complexity to the computer but enabled Whirlwind to operate for long periods without problems.

The second issue was what to use for the computer's main memory. ENIAC had just 20 words of memory that was fashioned out of the same vacuum tubes used for the rest of the system. This was totally inadequate for Whirlwind's intended purposes. The design goal was to have 2,048 words of 16-bit memory. The initial system configuration utilized electrostatic storage tubes. These proved to be difficult to fabricate and Whirlwind initially began operation with just 256 words of internal memory. The memory was later increased to 1,024 words. In early 1949, Forrester began experimenting with magnetic core memories. While the concept was relatively straightforward, finding the appropriate material that had the magnetic properties they were looking for took several years. In retrospect, the development of magnetic core memory by the Whirlwind design team was probably its most significant accomplishment even though we are more interested in its graphics capabilities.
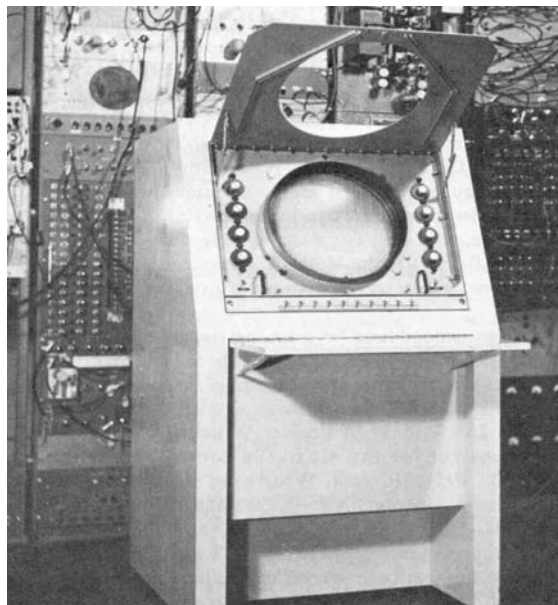

Figure 3.1
16-inch Whirlwind Display Console[2]

In 1947, the group within the Servomechanisms Laboratory working on Project Whirlwind became the Lab's Electronic Computer Division and in 1951 it became the MIT Digital Computer Laboratory, independent of the Servo Lab. During these years Forrester continued as both the Whirlwind project director and director of the Computer

---

[2] Redmond, Kent C. and Smith, Thomas M. – *Project Whirlwind: The History of a Computer Pioneer* – Digital Press, 1980 pg. 192

Laboratory with Everett as the associate director. Other key players who eventually played important roles in this story included Ken Olsen (founder of Digital Equipment Corporation), Norm Taylor (vice president of ITEK and leader of that company's development of the Electronic Drafting Machine – EDM described in Chapter 6), and Charles Adams and Jack Gilmore (co-founders of Adams Associates and instrumental in the development of the EDM). Forrester would go on to become a professor at MIT's Sloan School of Management while Everett would eventually become president of MITRE Corporation.

Whirlwind was a physically imposing machines as illustrated in Figure 3.2. It took up 2,500 square feet of space on the third floor of the Barta Building, just north of the main MIT campus. The machine's circuitry consisted of 12,500 vacuum tubes which consumed 150,000 watts of electricity. Whirlwind was assembled in a series of large racks which provided easy access to the machine's circuitry, facilitating maintenance and modification of what was essentially a research machine. Most of the physical equipment was fabricated by Sylvania Electric Products Company which was located in Boston at
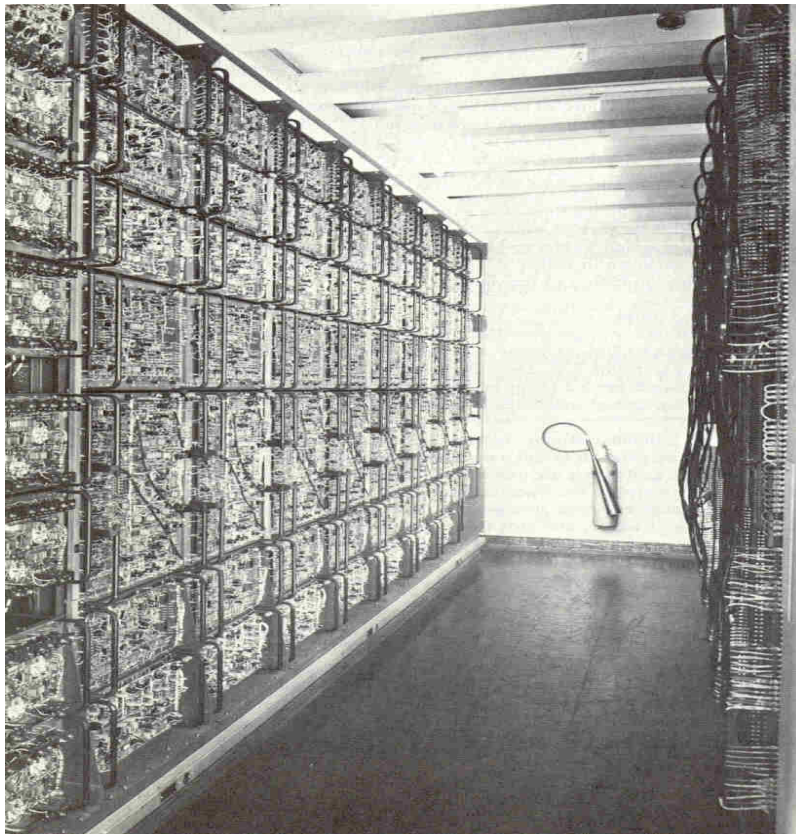

Figure 3.2
Whirlwind Arithmetic Unit (Approximately one tenth of the total computer system)[3]

---

[3] Ibid pg. 248

that time. It is hard to determine exactly when Whirlwind became operational. For practical purposes the best date is probably March 1951 although some productive work was done as early as the third quarter of 1949.

Computational performance was impressive for the time. The system had two clock speeds, 1 MHz and 2 MHz, which resulted in 20,000 operations per second. While many early computers handled data in a serial manner, Whirlwind transferred data and instruction words internally in a parallel fashion – all 16 bits at the same time. The computer was capable of supporting up to 32 different commands but only 27 were implemented.[4] While slow by today's standards, Whirlwind continued to provide valuable computational services to the MIT community until it was decommissioned in 1959.

After sitting idle for several years, it was moved in 1963 to Wolf Research & Development in West Concord, Massachusetts where it continued in operation until the early 1970s when it was permanently shut down. Overall, the Navy and Air Force spent about $5 million developing Whirlwind. As described in detail in *Project Whirlwind: The History of a Computer Pioneer* by Redmond and Smith, the relationship between the Whirlwind development team and the Navy was often quite rocky and there were several points in time when the Navy was close to shutting off the funding for this project. If it had done so, the history of the CAD industry might have turned out far different than it did.

During most of the 1950s, efforts surrounding Whirlwind focused on supporting the Air Force's SAGE (Semiautomatic Ground Environment) air defense system. The actual SAGE computers, which began deployment in 1958, were built by IBM utilizing design concepts derived from Whirlwind. A byproduct of the SAGE project was the establishment of MIT's Lincoln Laboratory, located in Bedford, MA, about 15 miles west of the Institute's Cambridge campus. Lincoln Lab is important to this story in that it was the initial home of the TX-0 computer, a second generation Whirlwind system, and the permanent home of the TX-2 computer. The latter computer was used extensively as a graphics research platform throughout the 1960s. The Computer Lab became Division 6 of Lincoln Lab with Forrester as its director from 1951 to 1956. Everett took over the division at that point and managed it until 1959 when he left to help start MITRE Corporation.

Forrester was very perceptive in seeing where the development of computer technology was leading. As early as 1948 he described remote access of computers similar to the time-sharing methods of the 1960s and the use of the Internet today.[5]

Graphics programming with Whirlwind

Programming early computers was done at the lowest level of the machine's binary command and memory addressing structure. Several research centers realized that better techniques could be developed. One of these groups was a Whirlwind programming team led by Charles Adams. Adams delivered a paper at the 1952 meeting of the Association of Computing Machinery where he said "Ideally, one would like a procedure in which the mathematical formulation together with the initial conditions can

---

[4] Ibid pg. 240
[5] Ibid pg. 232

simply be set down in words and symbols and then solved directly by a computer without further programming."[6]

According to Herman Goldstine:

> "At the same meeting John Carr described the programming work at the Massachusetts Institute of Technology. It is clear that the Whirlwind group there was very alive to the needs of the programmer. Adams and J. T. Gilmore[7] extended the ideas of Wilkes, Wheeler, and Gill, and there evolved from this symbolic address procedure, an idea that seems to have been independently created by Rochester and his colleagues at IBM. The Whirlwind group also pioneered in the development of a so-called interpretive algebraic coding system. ..."[8]

From its earliest existence, Whirlwind had some form of graphics display attached. The earliest device could display just 256 points of light. This was subsequently expanded to 4,096 or a 64 by 64 matrix. Substantially less than the 1280 by 1024 or greater resolution used by contemporary graphics systems. Adams wrote a short program that displayed a bouncing ball on the display. This was done by solving three simultaneous differential equations. A little later, probably in late 1950, Adams and Gilmore wrote the first computer game. It consisted of trying to get the ball to go through a hole in the floor by changing the frequency of the calculations.

Meanwhile, Everett developed the first version of the light gun, the predecessor to the light pen, that could be used to identify specific displayed points on the CRT. The first engineering design application on Whirlwind was probably done by a masters thesis student, Dom Combelec, who wrote a program to help design antenna arrays.[9]

Development of NC machine tools and APT

The wartime control feedback work of the Servomechanisms Laboratory involved military equipment that utilized continuous or analog signals to control devices such as radar antennas and anti-aircraft guns. By 1949, the theoretical foundation was being developed to control servo mechanisms using digital or pulse data. In the spring of that year, Gordon Brown received a telephone call from John T. Parsons, Jr. of the Parsons Corporation's Aircraft Division in Traverse City, Michigan. Parsons was looking for a device that could accept digital data and drive a machine tool.[10]

---

[6] Adams, Charles W. – *Small Problems on Large Computers*- Proceeding of the ACM, 1952 pg.101

[7] Adams and Jack Gilmore would go on to establish one of the first computer software consulting firms in the country in 1959. In 1961, they initiated the development of a prototype CAD systems under contract to Itek Corporation working with Norm Taylor who had been instrumental in the early Whirlwind activity and subsequently was a senior manager on the SAGE project. See Chapter 6.

[8] Goldstine, Herman – *The Computer: From Pascal to von Neumann* – Princeton University Press, 1972 pg. 338. In a private communication, Gilmore told me that he and Adams briefed IBM's Nat Rochester on the concept shortly after Gilmore came up with it and that the concept was not developed independently at IBM.

[9] Taylor, Norman H., *Retrospectives I: The Early Years in Computer Graphics*, SIGGRAPH '88 Panel Proceedings

[10] Reintjes, J. Francis – *Numerical Control: Making a New Technology* – Oxford University Press, 1991 pg. 16

The common practice at the time for machining complex surfaces was to drill closely spaced holes to a predetermined depth and then to finish the part either by carefully machining the surface or manually filing. The Air Force was beginning to develop a series of high-speed fighter aircraft and needed better methods to machine parts to high tolerances. Parsons had negotiated a contract with the Air Force to develop a machine to do this but he lacked the technical resources to accomplish the project. The result was a three-way arrangement between Parsons, the Air Force and MIT.

The initial Servo Lab project leader was William Pease assisted by James McDonough. They quickly recognized that a comprehensive multi-axis machining technique was preferred over the technique of simply drilling closely spaced holes. The initial idea was that a data record (the plan was to use punch cards for inputting this data) would be read for each increment the tool would move. With a positional resolution of 0.0005 inches, this would have required a card reader far faster than anything then available and a huge volume of punch cards for even the simplest part. The concept that evolved at MIT was to provide a command for the machine to go to a specific point in three-dimensional space and then have a controller mechanism that would feed pulse data to the motors that controlled each axis of the machine tool.[11] The basic ideas developed by Pease and McDonough is fundamentally how it is still done today.

Actual implementation of these concepts began in earnest in July, 1950. One of the new Servo Lab staff members working under McDonough was Herbert Grossimon.[12] The Air Force provided the Servo Lab with a three-axis Cincinnati Milling Hydro-Tel milling machine which was subsequently equipped with the appropriate control devices for each axis and interfaced to a vacuum tube NC controller. Machining commands were read from punched paper tape instead of punch cards. The experimental machine tool was up and running in March 1952.[13] A formal introduction of the first NC machine tool was held at MIT on September 15-17, 1952. The total cost to the Air Force for the Servo Lab's work on developing NC to this point came to just $360,000.[14]

Preparing control tapes for early NC machine tools was a time-consuming manual process. By the time the first NC machine tool was introduced, it was recognized that unless the process of producing these tapes could be substantially improved, the overall economics of using NC would suffer. The first work in trying to automate the process was done by John H. Runyon on Whirlwind using a subroutine technique. This was followed by Arnold Siegel's efforts in developing a procedural language for programming machine tools. Using Segal's software, a part that had taken eight hours to program by hand was done on Whirlwind in less than 15 minutes.[15]

A major effort to develop a more advanced part programming solution was undertaken in 1956 by Doug Ross who had joined the Lab's staff several years earlier. A

---

[11] Reintjes, J. Francis – *Numerical Control: Making a New Technology* – Oxford University Press, 1991 pg. 34

[12] In 1956, McDonough and Grossimon left MIT and founded Concord Control to build NC controllers. The company subsequently developed a series of specialized plotters and digitizers. One such device was a plot-back digitizer in 1968. The Concord Control project manager was Philippe Villers who shortly thereafter co-founded Computervision. See Chapter 12.

[13] Reintjes, J. Francis – *Numerical Control: Making a New Technology* – Oxford University Press, 1991 pg. 43

[14] Ibid p. 47

[15] Ibid p. 78

Computer Applications Group was established within the Lab with Ross as its head and with John Ward assigned the task of acting project engineer for the development of what eventually became the Automatically Programmed Tool (APT) system. Ward was subsequently replaced by Donald Clements as permanent APT project administrator.

APT was a fairly significant undertaking that involved substantial participation by a number of large industrial companies, particularly those in the aerospace industry under the auspices of the Aircraft Industries Association (AIA is now known as the Aerospace Industries Association). These companies did not simply provide funding support - rather they actively participated in the development of the APT software.

One method they developed that is still used was the concept of preparing NC information in a generalized form and then post-processing that information for specific combinations of machine tools and controllers. By the mid-1950s it was obvious that the machine tool manufacturers were not interested in building the controllers themselves and that these would be built by companies such as Bendix and General Electric that had more expertise in the field of electronics.

Ross felt that Siegel's approach did not go far enough and that a more comprehensive part programming language was needed. He wanted something where the part programming could be done using English-like statements and could be done by someone who was not a computer programmer. His work on APT resulted in a long-term personal commitment to the development of tools that would bridge the gap between engineers and the computer. According to J. Francis Reintjes:

> "He envisioned programming in general as an interrelational process between human and computer in which a general purpose computer would, in effect, be turned into a special purpose machine through use of a set of special programs developed to deal with whatever problem was at hand. The human would then work back and forth with the computer, in a conversation-like mode."[16]

The first version of APT developed by the Servo Lab team under Ross and its industry partners was a two-dimensional implementation that utilized curved boundary lines. It was called 2D-APT-II. Nine aircraft companies and IBM worked with MIT on this implementation. Most of the early programming work was done on IBM 704 computers which all the aircraft manufacturers had access to. MIT had not yet installed the 704 it had on order and its work continued to be done on Whirlwind. 2D-APT-II was ready for initial field testing on April 30, 1958. This was a rather complex application with the documentation taking six volumes and running to 517 pages.[17]

Once 2D-APT-II was completed, the Servo Lab began phasing out of APT development and moving on to more generalized CAD development. Ross and his people continued to work on some of the core mathematical routines but the bulk of the development initially moved to the AIA's Central Project Group in San Diego and then, in 1962, to the Illinois Institute of Technology Research Institute (known today as Alion Science and Technology) which took over the long term support of APT - subsequently referred to as APT-III. IITRI, under the leadership of Dr. S. Hori, continued to provide

---

[16] Ibid pg. 81
[17] Ibid pg. 87

APT support through the mid-1970s. An early aluminum alloy forging machined using APT III to program the NC milling machine is shown in Figure 3.3. Not only did this single machined piece replaced an assembly of over 250 sheet metal parts but the resulting aircraft component was stronger, more reliable and cost less to produce.[18]

Work on establishing APT as an international standard began in 1963 under the auspices of the USASI X3.4.7 Subcommittee. Since APT had been implemented on the assumption that user organizations would be able to add company or project-specific functionality to the core software, establishing a standard was a complex task. If every conceivable function was included, the standard would have been unwieldy and possibly impossible for many computer companies to implement. On the other hand, an overly
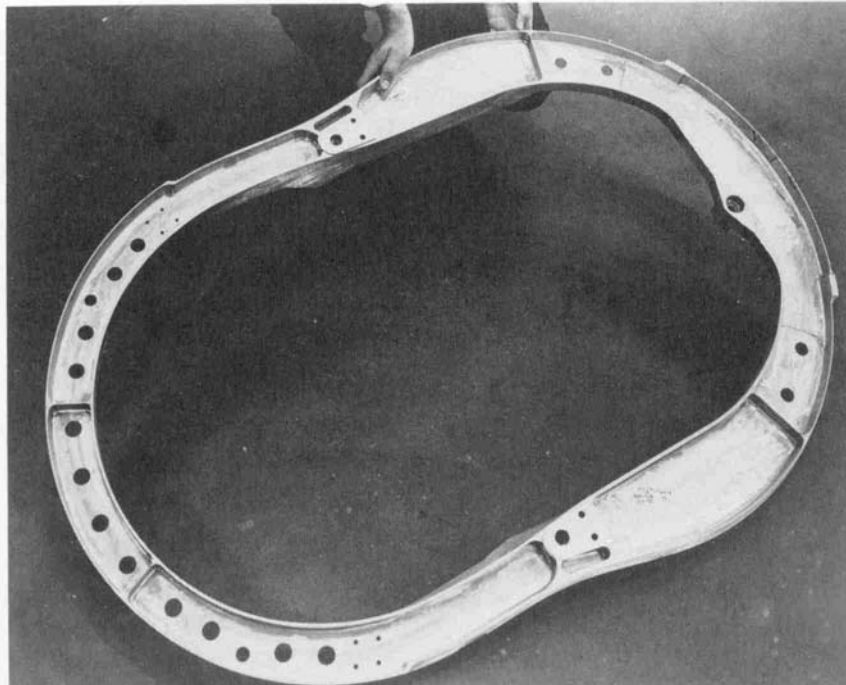

Figure 3.3
Aluminum Alloy Forging Machined Using APT III[19]

limited subset would have equally impractical. The solution was to define five subsets where each subset incorporated all the features of lower level subsets. The five levels were :

1. Multi-axis contouring system
2. 3D contouring system
3. Minimum contouring system
4. Advanced point-to-point system
5. Minimum system

Specific sets of special-purpose features implemented by various groups were categorized as "modular features" with the ability of a user to plug the modular feature into a core

---

[18] Ibid pg. 89
[19] Ibid pg. 90

subset. As of 1968 there were 25 such modular features being considered by the X3.4.7 Subcommittee.[20]

APT was eventually implemented on a number of large mainframe computer systems. Among the many different versions were APT 360 developed by IBM, 6400/6600 APT developed by Control Data Corporation, and UCC APT developed by University Computing Corporation. Some companies including General Electric developed their own in-house implementations of APT. While these different implementations all basically conformed to the APT standard, each version had its own idiosyncrasies. In the late-1970s, IBM launched a new version of APT for the 370 series of computers called APT-AC where AC stood for Advanced Contouring. Today, there are probably still several hundred organizations, mostly aerospace companies and large machine shops, using APT with the primary source of support coming from Austin NC Corporation, the group that previously was the APT support team at University Computing. The latest versions run under the UNIX and Windows operating systems.[21]

Moving on to new research

An offspring of the original Servomechanisms Laboratory at MIT in the early 1950s was the Dynamic Analysis and Control Laboratory (DACL) which worked on the research and development of components for air-to-air missiles. In 1952, Robert W. Mann joined DACL as the head of its Design Division. Mann had worked as a draftsman before the war for Bell Telephone Laboratories and had returned there after serving in the Army Signal Corps. Under the G.I. Bill he was able to attend MIT and earn both a Bachelor's and Master's degree in mechanical engineering in four years. Shortly after taking on the design management role with DACL, Mann was asked to join the MIT Mechanical Engineering faculty to teach design.

As readers will see in succeeding pages, the history of the CAD industry is replete with personal relationships and chance meetings that took place at MIT. Mann's undergraduate roommate was Kenneth Olsen, the future founder of Digital Equipment Corporation, who worked with Forrester on the development of Whirlwind's magnetic core memory. Olsen was responsible for the development of a specialized variant of Whirlwind called the Memory Test Computer. Mann's future wife, Margaret Florencourt, was also part of the Whirlwind team.

Within MIT's Mechanical Engineering Department, a major change took place in 1953. At that time the Institute restructured its undergraduate curriculum, eliminating the requirement that all freshman take a required course in drawing and descriptive geometry. The faculty teaching this course, including Steven Anson Coons, were merged into the department's Design Division. Coons had received his bachelor's degree from MIT in 1932 and had worked for a number of years as a mathematician and designer for Chance-Vought Aircraft. Coons would eventually become world-known for his work in developing surface definition concepts used throughout the CAD industry.[22]

---

[20] Feldman, Clarence G. – *Subsets and modular features of standard APT* - Proceedings of the 1968 Fall Joint Computer Conference, San Francisco, California. 1968 Volume 33, Thompson Books, pg. 67

[21] Telephone conversation with Ven Sudhaker on June 20, 2003. Sudhaker had been involved for many years with UCC's activities in this area and was currently with Austin NC.

[22] Robert W. Mann - *Fundamental Developments of Computer-Aided Geometric Modeling* – Chapter 16 *Compute-Aided Design – 1959 through 1965 – In the Design and Graphics Division of MIT's Mechanical Engineering Department,* 1993

Getting the CAD ball rolling at MIT

In January 1959, an informal meeting was held involving individuals from both the Electronic Systems Laboratory (the new name for the former Servomechanisms Laboratory) and the Mechanical Engineering Department's Design Division. They concluded that there was a significant opportunity for digital computers to be used to automate engineering design activities. The ESL contract with the Air Force for the development and support of APT was scheduled to terminate in about a year and Reintjes and his staff were eager to find a new area in which to channel their resources.[23]

It is interesting to note that as early as 1959 the question about whether or not mechanical drawings would be needed in the future was already being discussed by ESL personnel. It would take over 30 years before design engineers would feel comfortable transferring three-dimensional models of parts to manufacturing organizations for production without the necessity of accompanying those electronic files with paper drawings.

The January meeting led to a series of seminars, the first of which was held in April, 1959. Among those in attendance were Reintjes and Ross representing ESL and Mann and Coons representing the Mechanical Engineering Department. Also involved on the Mechanical Engineering Department side was Dwight Baumann who had joined DACL after receiving his bachelor's degree from North Dakota State University and was then working as an instructor in the Design Division. A second seminar was held in May. Perhaps the most significant development resulting from these seminars was the definition of the effort as Computer-Aided Design rather than Computerized Design or Computer Automated Design. Mann in particular thought that the addition of the hyphen was particularly significant.[24] The point being that the computer can assist engineers in creating designs but cannot replace them.

According to Reintjes, by the spring of 1959 they had a fairly good idea of what the general tasks should initially be in developing a viable Computer-Aided Design system.

> "…a critical analysis of what roles the computer should play for creative design; the development of a system of programs for the creative design process; and an investigation of requirements for a workstation that would serve as the interface between the designer and the design tool – the general purpose computer."[25]

At this early stage of development, these tasks were rephrased as:
- How does a design engineer enter operational commands and spatial information into the computer?
- How is the design information presented back to the engineer?
- How is the information stored internally in the computer system?
- What design and drafting functions need to be implemented?

---

[23] J. Francis Reintjes – *Numerical Control- Making a New Technology,* 1991
[24] Robert Mann (item cited)
[25] J. Francis Reintjes – *Numerical Control- Making a New Technology,* 1991

We sometimes forget that modern computer systems and applications are built on top of the technology that has come before. Today, a personal computer costing less than $1,000 comes with a mouse for input, a high performance color display, a functionally rich operating system, communication tools, a document scanner, a large quantity of main memory and disk storage, a graphical printer and a wide variety of application software. Virtually none of this existed in 1959 and what did exist was expensive and difficult to access.

In December 1959, the Air Force issued a one year contract to ESL in the amount of $223,000 to fund what became known as the Computer-Aided Design Project. Included in the contract was $20,800 for 104 hours of computer time at $200 per hour. In 1959 that was a lot of money. Newly graduated engineers were making perhaps $500 to $600 per month at the time.

Much of the work on the Air Force contract was done by graduate students working on masters and Ph.D. theses. One of these students was Richard Parmelee who focused on using the computer for stress analysis. According to Reintjes, Mann, Coons and Parmelee saw excellent possibilities for the computer in stress analysis:

> "…..because it was hoped that, through the use of a computer, many more checkpoints on stress could be analyzed quickly and economically. Since hand calculation was long, tedious, and often only approximate, the conventional design tendency was to limit the number of calculations to locations deemed critical in light of the designer's judgment and experience."[26]

One conclusion Parmelee reached in preparing his master's thesis was that a stress analysis capability would need to be compatible with the many other functions that would be required by a complete CAD system. This was the start of the concept of a fully integrated design capability where individual programs would feed data to other modules and use the data they created by still other modules.

The then high cost of computer resources limited the flexibility the project team had in exploring avenues of research and required them to use whatever resources they did have as efficiently as possible. Another graduate student, Joe Purvis, Jr. looked at the issue of storing and retrieving standard component data. One major problem he faced was the high cost of data storage at that time. A 10MB storage unit cost about $60,000 per year to lease (most large-scale computer systems were leased at the time rather than purchased outright) as compared to a 300GB disk drive which costs less than $100 to purchase today.

The key point to be made here is that in 1959 and 1960 the researchers at MIT were already looking at the right issues even if the computer technology had not yet advanced sufficiently to support their ideas.

Focus on large computers

The Computer-Aided Design Project being run by ESL and the Mechanical Engineering Department focused, for the most part, on using large mainframe computers. Typically these systems were run by professional operators and programmers had little

---

[26] Ibid, Pg. 103

opportunity to gain hands-on experience with them. This was a different approach than what was being followed across the campus in the Civil Engineering Department where, under the leadership of Professor Charles Miller, software development for highway design applications was initially being done on an IBM 650 located in Boston followed by an IBM 1620 installed on campus. With these machines, the programmer was the operator. On the other-hand, these machines provided no on-line graphics capability.

Initially, ESL used Whirlwind extensively for tasks such as developing the APT NC part programming language. By the late 1950s, that machine was becoming costly to support and it was phased out for general work by 1958. This forced the Lab to shift its activity to the Institute's Computation Center run by Dr. Phillip Morse, Professor of Physics. The center had an IBM 709[27] mainframe which was only available eight hours per day. The other 16 hours were split between other New England universities and IBM's own work. In 1962, the vacuum tube 709 was replaced by a transistorized mainframe computer, an IBM 7090. Although somewhat faster than the 709, the 7090 still could not keep up with the workload.

The demand for computer resources was so great that instead of returning the 709 to IBM when the 7090 was installed, the Institute purchased it outright and installed it in a separate facility run by Professor John Slater, Professor of Physics. This machine was operated on a fee-for-service basis and was used extensively by the CAD Project. Probably more important was the fact that the TX-0 computer at Lincoln Laboratory had been replaced by the far larger TX-2 in 1959 and was basically surplus. The TX-0 was transferred to the Research Laboratory of Electronics and was subsequently used extensively by the CAD Project, especially for work on interactive workstations.

TX-0 was originally built by Lincoln Lab as a test bed for building transistorized computers. The team responsible for this machine was led by Ken Olsen. It was an 18-bit machine that contained 3,500 transistors at a time when they cost $80 each. Initially, TX-0 had a 64K word magnetic core memory but when it was turned over to ESL, this memory was removed to be used in the TX-2 and a smaller 4K memory was installed. The machine was capable of 83,000 add/subtract type operations per second. Because the machine was intended primarily as a test bed for transistorized design, it had a fairly limited instruction repertoire.

The TX-0 had a CRT display designed by Ben Gurley and a light pen designed by Wesley Clark. The availability of this machine and its graphics terminal complemented the CAD Project's use of the MIT Computation Center's mainframe machine. Olsen went on to start Digital Equipment Corporation in 1957 and the company's first computer, the PDP-1, had many similarities to the TX-0 but a more extensive set of commands.

In 1963, MIT's IBM 7090 was replaced by a 7094 system. For the first time, the Computation Center was able to offer time-sharing services, greatly improving access to the Center's resources. Development of this system was under the direction of Dr. Fernando Corbató. The software, the development of which had started on the 709 and continued on the 7090, was called the Compatible Time Sharing System (CTSS) and it could support 30 users at one time.[28]

---

[27] The IBM 709 replaced an IBM 704 which was in use at MIT for a relatively short period of time.

[28] Corbató, Fernando J. et al – *An Experimental Time-Sharing System* - Proceedings of the Spring Joint Computer Conference, San Francisco, California 1962 Vol. 21 – Spartan Books, pg. 335

Corbató's time-sharing work on the 7090 at the Computation Center led, in 1963, to the establishment of Project MAC, funded by the Department of Defense's Advanced Research Projects Agency (ARPA). Under Dr. Robert Fano, Professor of Electrical Engineering, Project MAC implemented an advanced version of CTSS called MULTICS (Multiplexed Information and Computing Service). The software ran on a modified General Electric 635 computer subsequently marketed as the GE 645. It took until 1968 before MULTICS became available on the GE 645 and it was eventually used extensively for CAD research in the late-1960s and during the 1970s.

Formalizing data structures for CAD systems

After completing the 2D-APT-II system, Ross and his team of programmers continued working on improving the mathematical calculation portion of the APT system. This software was called ARELEM (Arithmetic Element) and was key to future versions of APT. The work covered the period from 1960 to 1962. During the same time frame, Ross began establishing a theoretical framework for CAD software. One of the key issues was how to store data so that one geometric element could link to one or more other elements in the design database. Some of his ideas started germinating as early as April 1959 when Ross presented a paper at an MIT Symbolic Manipulation Conference. He subsequently published an extensive paper on this subject at the 1963 Spring Joint Computer Conference with Jorge E. Rodriquez as co-author that proposed a data structure that would be insensitive to where in the CAD model an operator selected an item for further action. Ross and Rodriquez emphasized that users of CAD applications need not be aware of the internal data structure of the applications they would be using.[29] An expanded version of this architecture was used by Ivan Sutherland in developing the Sketchpad system described below.

Ross' work on APT along with his initial design of a CAD system data architecture led him to conceptualize a comprehensive software environment for implementing engineering design software. Called AED for Automated Engineering Design, it was intended to be a generalized problem solving system that would require a new and non-traditional programming language. Ross settled on using a modified form of ALGOL-60 for AED programming. It is interesting to note that Ross was focused on "automated design" while Mann and Reintjes had earlier made an issue of the fact that it was to be computer-aided design. As mentioned above, Mann was adamant about the hyphen in the term.

AED eventually involved tens of man-years of programming effort. In typical Ross style, AED was described as a generalized methodology that consisted of four primary components; a lexical processor that would break down input statements into their basic components, a parsing processor that grouped these basic components into meaningful operations, a modeling processor that extracted meaning from these statements and created relevant data and an analysis processor that actually carried out the solution to the problem as defined.[30] As the reader can see, AED was conceptualized

---

[29] Ross, Douglas T. and Rodriquez, Jorge E. – *Theoretical Foundations for the Computer-Aided Design System* - Proceedings of the Spring Joint Computer Conference, Detroit, Michigan 1963 Vol. 23 – Spartan Books pg. 305
[30] Reintjes, J. Francis – *Numerical Control- Making a New Technology,* 1991 p. 118

as a general problem solution methodology and was not necessarily limited to mechanical engineering problems.

This was an ambitious undertaking, perhaps too ambitious. The generalization of AED was driven by tables – changing the nature of a problem to be solved was expected to simply require the formulation of new tables. A initial version of AED called AED-0 was up and running on the MIT Project MAC time-sharing system sometime in 1963. In addition to using the software for mechanical design applications funded by the Air Force, ESL also undertook the development of several electrical and electronic design applications under a $40,000 research grant from NASA.

Ross had plans for a more powerful version of AED called AED-1 but lacked the financial wherewithal to accomplish the task with just MIT's resources. ESL took the same approach it had with the development of APT and solicited support from industry participants in December, 1963. By the following March, six experienced programmers were on board from a group of aerospace firms for one year assignments at MIT. A seventh from IBM joined later that year. Instead of working on AED-1, however, the team spent most of its efforts enhancing AED-0 and implementing batch versions for IBM 709, 7090 and 7094 mainframe computers as well as expanding the time-sharing version. This development effort took about a year and the upgraded version of AED-0 was released to interested companies in March 1965.

Development of the light pen and display terminals

Obviously, if a computer system were to be developed that would enable engineers to interactively create technical designs, a method was needed to input graphical information and to view that data. The first reference to the need for a visual display device that I found was in a memorandum written in 1945 by John von Neuman.

> "In many cases the output really desired is not digital (presumably printed) but pictorial (graphed). In such situations the machine should graph it directly, especially because graphing can be done electronically and hence more quickly than printing. The natural output in such a case is an oscilloscope, i.e. a picture on its florescent screen. In some cases these pictures are wanted for permanent storage (i.e. they should be photographed); in others only visual inspection is desired. Both alternatives should be provided for."[31]

The first practical use of an oscilloscope or CRT as a graphical output device was likely in conjunction with MIT's Whirlwind in the early 1950s. At one point it was shown on an Edward R. Morrow television show, *See It Now*, with Jack Gilmore participating in the demonstration.[32]

There were two types of CRT displays in use at this time. One type displayed an image in the form of a pattern of dots while the other type created an image by drawing straight line segments between pairs of points. With the latter type of device, often

---

[31] von Neumann, John – *Memorandum on the Program of the High-Speed Computer Project*, November 8, 1945 as quoted by Herman Goldstine in *The Computer- from Pascal to von Neumann,* 1972 Princeton University Press pg. 242
[32] *The Story of Computer Graphics*, ACM SIGGRAPH Video 1999.

referred to as a stroked display, curves and circular arcs were displayed as series of short vectors. Most CRT displays were not persistent, meaning that unless the image was refreshed continuously, it would quickly fade away. Typically, the image had to be regenerated 30 or more times per second in order to avoid having the image flicker. The data for refreshing the image was stored either in the computer memory, on a secondary storage device such as a magnetic drum or disk or in a dedicated memory built into the display terminal itself. It fairly quickly became obvious that stroked displays were preferable for CAD applications since they could create higher quality and more complex images, but they were more expensive than point displays.

Figure 3.4
Use of a Light Gun by a SAGE Operator[33]

Given that a CRT was to be used for graphical output, the next issue was how to input commands and graphical information. Typewriters, pushbuttons and rheostat dials were all in use for various applications by the early 1960s and they were utilized by many early CAD developers. None of these devices were effective tools for inputting graphical information or for selecting previously created graphical elements. SAGE used a light gun, invented by Bob Everett, for this purpose. It sensed light on the CRT screen and caused a computer interrupt to occur. The device sensed the initial burst of light caused when the electrons first stuck the phosphor on the back of the CRT screen. This process occurred in just a few microseconds, but it was enough time that the computer could identify the specific graphical item that had been pointed to. A trigger mechanism was used to activate the light gun. Otherwise, the photosensitive pick-up element in the gun would sense random light and would cause false signals to be sent to the computer. Light

[33] Redmond, Kent C. – *From Whirlwind to Mitre: The R&D Story of the SAGE Air Defense System* – 2000, MIT Press pg. 436

guns, which were the size and shape of an automatic pistol, were used extensively in the SAGE system as shown in Figure 3.4.

The light gun was an effective device when used as a component in an air defense system such as SAGE but was rather awkward to use for scientific and engineering applications. An ergonomically more attractive device called a light pen was designed at MIT's Lincoln Laboratory for use with the TX-0 and later TX-2 computers. When the TX-0 was transferred to the MIT campus in 1958, a CRT display and light pen were part of the configuration. The light pen was the size and shape of a large fountain pen with a shutter button replacing the light gun's trigger. The initial version, which had been designed by Ben Gurley at Lincoln Lab, was improved upon by the ESL staff under the direction of John Ward. The major problem with the original light pen was that CRT tubes at the time were substantially curved (the flat screens we are familiar with today are a fairly recent development) and were covered by a protective plate. The result was that the distance from the tip of the light pen to the actual CRT screen (the side with the phosphor) was as much as two inches. This made it difficult to resolve small items displayed on the screen.

The original work done at Lincoln Lab on both light pens and CRT displays was well documented by Robert Stotz in a paper presented at the 1963 Spring Joint Computer Conference. Stotz, who worked with Ross at MIT, pointed out that a key graphics design issue was what work should be done by special purpose graphics hardware and what work should be done by the computer the device is attached to.[34]

Ward redesigned the light pen with a small focusing lens in front of the photo diode sensing element. The redesigned pen was about 5/8 inch in diameter and seven inches long. According to Reintjes, the new light pen was used on the ESL Display Console associated with Project MAC and Ward was asked to produce similar pens by more than a dozen other research organizations, requests which ESL responded to. A diagram of the redesigned light pen is shown in Figure 3.5. One interesting connection is the fact that the member of ESL's staff responsible for supervising the operation of TX-0 from 1958 to 1959 was Earl W. Pughe, Jr. Pughe subsequently went to work for Itek Corporation and was responsible for much of the hardware engineering on that company's Electronic Drafting Machine including its light pen.
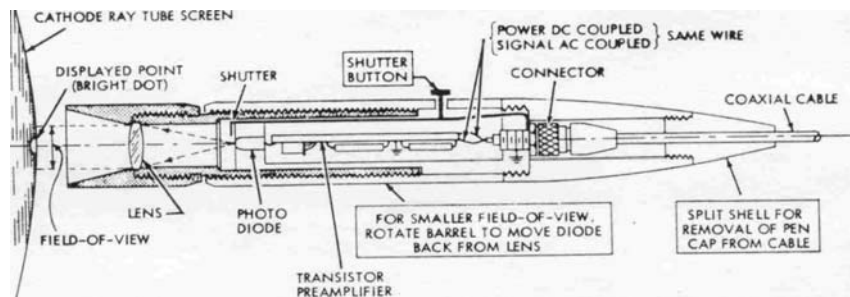

Figure 3.5
ESL's Advanced Light Pen[35]

---

[34] Stotz, Robert – *Man-Machine Console Facilities for Computer-Aided Design* - Proceedings of the Spring Joint Computer Conference, Detroit, Michigan 1963 Vol. 23 – Spartan Books pg. 327
[35] Reintjes, J. Francis – *Numerical Control – Making a New Technology* – Oxford, 1991 pg. 113

One of the key aspects of the TX-0 computer once it had been transferred to the MIT campus was that students and research associates could sign up to use the machine for a maximum of one hour at a time. This was one of the few places where someone working on a software problem could have an entire machine for their own use. Nearly everyone else on the MIT campus had to work with batch mainframes – not particularly conducive for developing interactive software.

Early graphics activity at Lincoln Laboratory

In 1955, MIT Lincoln Laboratory established an Advanced Computer Research Group headed by William Papian with Wes Clark responsible for logical design and software and Ken Olsen responsible for circuitry and computer production. One of the group's first projects was the previously mentioned TX-0 computer. The first computer to use transistor circuitry, TX-0 was built primarily to test the circuitry and core memory destined to be used in the TX-2 computer. This machine was equipped with a CRT display and light-pen.

Jack Gilmore, who had left the Whirlwind project in 1952 to become a Navy pilot, returned in 1956 and joined Clark's logical design team. He subsequently developed an on-line operating system that provided programmers with the ability to sit at the computer's console and debug and modify programs using an IBM typewriter and CRT display. It was a new concept in that most on-line program debugging until then was done at the machine code level using console lights and switches.

A combination of CRT display, on-line keyboard and light pen was used by Gilmore to build a utility program so that programmers could debug programs using flow charts on the display with logical switches that could be opened and closed during the debugging session. One of the first TX-0 graphic applications was a pattern recognition algorithm that examined EEG brain wave data fed into the TX-0 via an analog to digital converter.

The demand for utilizing scientific symbols in TX-0 applications led Clark to initiate a project in 1957 to design and build a work station (subsequently called the MIT Lincoln Writer) that would enable programmer scientists to use scientific symbols in their programs and output results to the CRT display as well as a printer.

Ironically, this project led to the first use of interactive text editing and graphics. The need to design a scientific set of a symbols that could be used to represent a complete set of alpha numeric symbols and characters led the team to build a program to simulate a scientific work station. The light pen was used to design individual combination of spots that formed the characters and a simulated keyboard consisting of 200 keys was displayed on the lower half of the display as 200 individual points. A plastic overlay was used with the anticipated symbols labeled above each key point. The result was a program, primarily written by Gilmore, called "Scopewriter."

One task Gilmore undertook was to simulate a typewriter which resulted in what was perhaps the first interactive text editor. In the process of building symbols, they found that they could create electronic symbols and geometric figures and they were used to create simple circuit diagrams and flow charts. Gilmore summarized this effort at a 1988 SIGGRAPH conference.

"….what I believe the TX-0 really contributed was an online interactive man machine communication environment. We didn't really think about the fact that we were doing anything particularly impressive in the area of graphics, but we were trying to switch from a batch processing orientation to an interactive situation where programmers actually worked at a console and made changes right there on the spot."[36]

In 1959, Gilmore wrote an assembly program on the TX-2 for a programming language based on the alphanumeric and scientific character set of the Scopewriter. He left Lincoln Lab in October of 1959 and with Charles W. Adams, co-founded one of the earliest software consulting firms, Charles W. Adams Associates Inc. Gilmore's Lincoln Lab staff position was filled by Ivan Sutherland.[37]

Sketchpad – The project that got the world excited about interactive design

The activity that has received more credit than any other for launching the use of interactive graphics for engineering design and drafting was Ivan Sutherland's Sketchpad project developed using Lincoln Laboratory's TX-2 computer. The terminal consisted of a CRT display, a light pen, a set of push buttons, a panel of toggle switches and four dials that were used to change the size and position of the displayed image. See Figure 3.6. The TX-2 was also interfaced to a PACE plotter built by Electronic Associates.[38]

TX-2 was a powerful 36-bit computer that had earlier served as the prototype for the IBM machines built for the Air Force's SAGE project. Images were displayed on the graphics display in the form of a series of dots with a resolution of ten bits in each axis. These dots were displayed at the rate of 100,000 per second. To avoid flicker which occurred at 30 frames per second or less, an individual image was limited to about 3,000 dots. The coordinates of the dots to be displayed were stored in a table in the main memory of the TX-2 computer, enabling the Sketchpad application software to proceed independently from display operations.[39]

A light pen was used to select elements already being displayed or to indicate a new location on the screen. In the later case, the user would point to a symbol constantly displayed on the screen, in the case of Sketchpad it was the word "INK," and then drag a displayed cursor to the desired location. Light pen operations were terminated by flicking the pen. Since the computer could no longer determine that the light pen was sensing light, it would conclude that the operation was completed. Rotary dials allowed the user to display images at a wide range of scales.

Functional commands were entered using the terminals pushbuttons. To draw a line from an existing point to a new location, the user would aim the light pen at the existing line, press the "LINE" pushbutton, move the light pen to the location of the line's second endpoint and then flick the light pen to indicate that the operation was completed. The line ended up as a straight line irrespective of the path the user followed

---

[36] Gilmore, John T., *Retrospectives II: The Early Years in Computer Graphics*, SIGGRAPH '88 Panel Proceedings

[37] See Chapter 6 for a discussion of graphics work undertaken by Adams Associates in partnership with Itek Corporation.

[38] Sutherland, Ivan E. – *Sketchpad: A Man-Machine Graphical Communication System* - Proceedings of the Spring Joint Computer Conference, Detroit, Michigan, 1963 Vol. 23 – Spartan Books pg. 329

[39] Ibid pg. 334

to go from the initial point to the final point. The toggle switches were used to set specific modes of operation such as whether or not to display certain types of information.

Figure 3.6
Ivan Sutherland seated at the TX-2 display terminal[40]

A key aspect of Sketchpad was the use of the previously mentioned data file structure developed by Ross, somewhat extended by Sutherland. In his 1963 Spring Joint Computer Conference paper, Sutherland described it as a ring structure since the string of pointers eventually closes back on itself. This data structure enabled a Sketchpad user to insert new elements arbitrarily anywhere in the displayed drawing, to remove elements and have the software close the logical gap created by that operation, to merge several data files and to perform auxiliary operations on the data in either forward or reverse order.[41]

Each spot displayed on the CRT screen was stored in the previously mention display table in the form of a 36-bit word. Twenty bits were used to define the X and Y coordinates of the spot while the other 16 bits of the 36-bit word were used to point to the element that caused that point to be added to the display table. When the user pointed to a displayed line, circle or arc on the CRT screen, the computer could tell which spot in the display table was being sensed by the light pen. Since the display table word contained a pointer back into the ring-organized data table, the software could immediately identify the graphical element the user was pointing to.

The two features of Sketchpad that most impressed me were its ability to support nested symbols and its ability to support geometric constraints. Any group of geometric elements could be combined together to form a symbol (over the years these groupings would be called by many other names such as cells and blocks). This symbol could then be placed wherever the user desired at different sizes and different orientations. Specific locations of a symbol could be defined as an attachment point for other elements. This was particularly useful when using the system to create electrical schematic diagrams.

---

[40] Ibid pg. 329
[41] Ibid pg. 333

The software allowed constraints to be applied or removed from geometric elements after the elements had been initially constructed. As an example, lines could be made parallel to each other, defined to be vertical or horizontal, set to the same length and locked, one element to another. If two lines were defined as being parallel and then one line was defined as being horizontal or at some other angle, the second line would move accordingly, maintaining the parallel constraint. This was well before the concept of parametric design became popular.

While Sketchpad was a significant technical accomplishment, this success needs to be put in the context of the computer system available to support the software. TX-2 was a huge machine (I remember literally walking through it on a visit to Lincoln Lab) that cost the Air Force millions to build. Sketchpad used a substantial amount of the computer's resources when operating. As a consequence, it was not viewed at the time as putting legions of drafters out of work in the near future. Sutherland summed it up fairly well:

> "For drawings where motion of the drawing, or analysis of a drawn problem is of value to the user, Sketchpad excels. For highly repetitive drawings or drawings where accuracy is required, Sketchpad is sufficiently faster than conventional techniques to be worthwhile. For drawings which merely communicate with shops, it is probably better to use conventional paper and pencil."[42]

In his 1963 Spring Joint Computer Conference article Sutherland goes on to describe several applications where he felt the value of using interactive graphics was particularly significant. Large repetitive patterns that would have taken two days to do manually were done in less than an hour including the time it took to plot the results. One example was the use of Sketchpad to produce a binary coded decimal encoder where the layout had to be plotted to high accuracy. Sutherland liked the idea of using Sketchpad to analyze mechanical linkages, an application which would not become widespread for several decades. He also demonstrated using Sketchpad to do bridge stress analysis on TX-2 and suggested that the analysis of electrical circuits was also an attractive application. Finally, he stated that the system's ability to make moving drawings opened up the possibility of using it for making animated cartoons. Very prescient but somewhat ahead of its time.

Sutherland concluded that:

> "The circuit experience points out the most important fact about Sketchpad drawings. It is only worthwhile to make drawings on the computer if you need something more out of the drawing than just a drawing….. We are as yet a long way from being able to produce routine drawings economically with the computer."[43]

It would be another six years before Applicon and Computervision would begin delivering commercial CAD systems that did in fact produce drawing economically.

---

[42] Ibid pg. 341
[43] Ibid pg. 344

© 2008 David E. Weisberg

Sutherland eventually moved to the University of Utah where he worked with David Evans and the two co-founded Evans& Sutherland, an important computer graphics firm. In 1989, he was awarded the "Turing Award" by the Association of Computing Machinery for his work on Sketchpad.

Tim Johnson gives Sketchpad three dimensions

Timothy Johnson was a research assistant working for Doug Ross on the Air Force-sponsored CAD Project being undertaken by the MIT Mechanical Engineering department and the Institute's Electronic Systems Laboratory. His work, as reported in the Proceedings of the 1963 Spring Joint Computer Conference, was to extend the two-dimensional Sketchpad system to three dimensions. The resulting software, also developed on Lincoln Laboratory's TX-2 computer, is usually referred to as Sketchpad III.

Sketchpad III was the first computer-based graphics system to implement the traditional three orthogonal views of a three-dimensional object together with a perspective view of that object. The perspective view could be at a different scale from the other views. This layout was a result of Johnson's determination that users were uncomfortable sketching in perspective and that a more traditional drafting methodology using orthogonal views was needed.[44] Johnson used the same graphics display and light pen as did Sutherland. As with Sketchpad, pushbuttons were used to indicate what action was to be taken with the graphical element the user pointed to with the light pen such as erasing the item, moving it or using that location to start a new line or arc.

Rotary dials were used somewhat differently than with the initial Sketchpad program. In addition to magnifying and rotating the drawing, one dial was used to change the perspective view by modifying the point at which lines converged. Graphical elements created in one view of the object were immediately displayed in the other views. The model was rotated by selecting one of the views with the light pen and then turning one of the rotary dials on the display console. New graphic images of all three orthographic views and the perspective views were generated continuously as the dial was rotated and the resulting image displayed on the CRT screen. [45]

To create a new element in other than one of the orthogonal planes simply required the user to rotate the model until the plane of interest was parallel to the plane of the CRT display. This technique was subsequently used by many commercial CAD systems. Sketchpad III also started moving this technology more towards being a fairly complete design solution. Utility routines were developed for storing three-dimensional data on magnetic tape and plotting hard copy drawings. These routines were written by Leonard Hantman who subsequently went to work at Adams Associates where he managed a number of computer graphics projects.

The technique for displaying perspective views with hidden lines removed was developed by Larry Roberts[46] as part of his Ph.D. thesis. Initially Roberts wrote software

---

[44] Johnson, Timothy – *Sketchpad III: A Computer Program for Drawing in Three Dimensions* - Proceedings of the Spring Joint Computer Conference, Detroit, Michigan, 1963 Vol. 23 – Spartan Books pg. 348
[45] Ibid pg. 349
[46] Roberts is best known for his work in developing ARPANET, the forerunner to the Internet as we know it today. Roberts was also the founder and CEO of Telenet, the first packet switching company as well as CEO of a number of other communications companies.

on the TX-2 that took digitized photographs of three dimension objects and detected the edges of the object in three-dimension space. This led to the first software that did hidden line elimination. Roberts wanted to create perspective views without hidden lines but could find no documented techniques for combining matrix techniques with generating perspective views. He actually went back to some of the early German textbooks on descriptive geometry in order to solve this problem.[47]

   I remember visiting Lincoln Lab sometime in the latter part of 1964 or early 1965 for a demonstration of Sketchpad III where the operator (probably Tim Johnson) retrieved a model of a building and then demonstrated how the model could be rotated (with hidden lines removed) to the point where the viewer was inside the model looking out the front door. In his paper on Sketchpad III, Johnson listed as future research subjects the ability to define arbitrary surfaces, determine the intersection of surface in three-dimensional space, determine edges hidden by visual surfaces and satisfy general graphical constraints.[48]

General Motor's DAC-1

   Like most large automobile manufacturers, General Motors was extremely interested in determining the extent to which computer graphics could be used to improve vehicle design. Starting in the late 1950s General Motors Research (GMR) began work on a research project called DAC-1 where DAC stood for Design Augmented by Computers.[49] Prototype work involved the use of an IBM 704 mainframe computer equipped with a Model 780 display. The DAC-1 project leader at GM's Research Laboratories (GMR) was Edwin Jacks. Fred Krull, who was involved in CAD-related activity at GM for over 30 years, was also a key member of the team.

   Contrary to what has often been published, DAC-1 was not initially conceived to be a computer-based design and drafting system along the lines of Sutherland's Sketchpad or Itek's Electronic Drafting Machine although it eventually evolved to have many of the same capabilities. In 1964, when a series of papers were presented at that year's Fall Joint Computer Conference, the system was really a hardware test facility used to support research in computer graphics. Jacks outlined four areas of concern regarding using computer graphics in automotive design:

1. The need to be able to work with existing drawings. The computer system had to be able to read existing drawings as well as produce some form of graphical output. This led to the incorporation of a device to scan 35 millimeter film as well as a high-resolution CRT to produce film output.
2. Once data was scanned into the system, tools were needed to manipulate the data. The thinking at GM revolved around the need for several individuals to view the data and collaboratively decide

---

[47] Roberts, Lawrence G., *Retrospectives II: The Early Years in Computer Graphics*, SIGGRAPH '88 Panel Proceedings
[48] Johnson, Timothy – *Sketchpad III: A Computer Program for Drawing in Three Dimensions -* Proceedings of the Spring Joint Computer Conference, Detroit, Michigan, 1963 Vol. 23 – Spartan Books pg. 353
[49] The project was initially simply called "Digital Design."

what changes should be made and then to graphically make those changes to the data.

3. There was also a need to compare graphical data. A typical example would be comparing the roof line of a new model to the prior year's model.

4. In addition to graphical information, an engineering design system had to support textual information.[50]

The general philosophy at GM was summed up in 1994 in an article on the history of DAC-1 by Krull. "It was felt that, if a computer could read sketches and drawings, then it could be programmed to produce further drawings, engineering data, and control tapes for numerically controlled machine tools."[51] Initial development of these techniques was done on the IBM 704 computer using simple cubic polynomials to describe the outlines of components such as automobile hoods. GMR personnel developed surface interpolation techniques that preceded Steven Coon's work at MIT by several years. The resulting surface outline was output to the 780 display device.

Subsequent to proving the feasibility of these concepts on the 704 computer, GMR and IBM entered into a multi-million dollar joint development project in November 1960 to create the DAC-1 hardware configuration. The system, based on GMR's specifications, took 30 months to complete. It was built around an IBM 7090 mainframe with 32K words[52] (each word was 36 bits in length) of main memory. This was subsequently upgraded to a 7094 Model II with a 64K memory. The DAC-1 configuration required the design and construction of several specialized hardware components including a special data channel, a display adapter, a display unit, a photo-recorder-projector and a photo scanner. Some of this equipment was based on work IBM had previously undertaken as part of the SAGE air defense project described earlier.

Substantial programming effort was spent in modifying IBM's standard operating system so that the 7094 could handle both batch and real time programs at the same time. In order to understand what GMR was attempting to do, the reader needs to appreciate that this expensive computer configuration was supporting just a single graphics terminal.

While most real time systems were being programmed in assembly language in the early 1960s, the DAC-1 software was written using NOMAD, a customized version of MAD, the Michigan Algorithmic Decoder, which in turn, was based on ALGOL 58. FORTRAN was rejected as a programming language since at that time it could not handle bit manipulations nor could it handle interactive selection of subroutines for execution.

---

[50] Jacks, Edwin L., – *A Laboratory for the Study of Graphical Man-Machine Communications* - Proceedings of the Fall Joint Computer Conference, San Francisco, California 1964 Vol. 26 – Spartan Books pg. 343

[51] Krull, Fred N., *The Origin of Computer Graphics within General Motors*, IEEE Annals of the History of Computing, Vol 16, No.3, 1994

[52] Krull's paper in the *Annals of the History of Computing* repeatedly uses the term 32-Kbyte memory. I believe this statement is inaccurate and that it was a 32 K word memory.
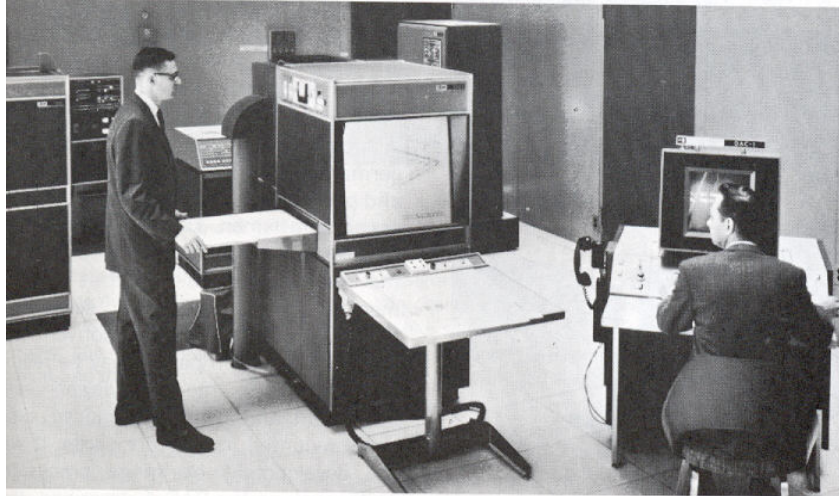
Figure 3.7
DAC-1 Hardware Configuration with Photo Scanner/Recorder at Center

The primary application implemented by 1964 was the scanning of drawing images from film, viewing and manipulated the scanned images on a CRT display and then producing copies of the revised images on film. The display terminal had a position-indicating pencil that functioned quite differently than the light pen which was used with other interactive systems of that era. Rather than sensing light, it utilized a conductive surface on the face of the display monitor to determine the location the user was pointing to.

Figure 3.8
DAC-1 Console

GMR programmers developed software that could take curve data or coordinate values describing surfaces and create an internal model of the surfaces. They also produced software that would take this three-dimensional data and prepare control tapes for driving NC machine tools.

It became apparent fairly quickly that scanning hard copy design data was not an effective approach to entering this information into the computer. GMR programmers began experimenting with an approach that involved using the console's function box with overlays indicating the task that a particular button would initiate. If the action required the selection of an existing item on the display, the operator would pick it with the electronic pencil. Contrary to subsequent experience at Lockheed with its CADAM software, GM engineers objected to the ergonomics involved with pointing to the screen with the pencil.

The DAC-1 hardware was the precursor to the Alpine system subsequently developed by IBM which resulted in the quite successful Model 2250 graphic terminal. The DAC-1 software, from its inception, was three-dimensional oriented. GM did some significant work in creating and manipulating fairly complex surface geometry at a time when other graphics developers were focused on straight-forward two-dimensional orthographic drawings. In 1967, GM's president, Edward Cole, decided that the DAC-1 project had become too expensive and transferred future responsibility for this type of technology to the company's Manufacturing Development Staff.

Over the next several years, work on applying computer graphics to design problem continued at GM, although at a somewhat lower level of intensity. By 1970, an experimental design system utilizing an IBM System 360/67 with several 2250 III display terminals was in use.[53] GM subsequently developed advanced design software that eventually was called the Corporate Graphic System or CGS. It was used for automotive body design until well into the 1990s.

---

[53] Beckermeyer, Robert L. – *Interactive graphic consoles – Environment and software* - Proceedings of the Fall Joint Computer Conference, Houston, Texas 1970 Vol. 37 – AFIPS Press, pg. 315