

# The history and real-life stories of historic memory and storage systems

Stefan Gloutnikov  
Computer Science Department  
San Jose State University  
San Jose, CA 95192  
408-924-1000  
sgloutnikov@gmail.com

## ABSTRACT

Computers are everywhere in our society, we cannot go a day without encountering them in a given day. And when we stop to think about how each one works at a deeper level, all the way to the hardware and bit-level, it is one marvelous work of art. At least for me, when I think about how all the millions of components in a computer work together, it is mind boggling how everything works just right.

Computer systems have evolved a lot over the past decades, and we have quickly forgotten, or have never even heard about past computer technologies. We did not always have the computers we see today from the very beginning, and it is why it is good to know about the history of such historic components. At least for the very simple fact of seeing how people solved and thought about challenging problems, which we now consider tremendously easy.

## 1. INTRODUCTION

In this article I will discuss some of the most memorable memory and storage systems from the past decades. Looking at how they work was one thing, but also hearing from the people that used them, and the memories that they had was another exciting part of my research. There are a lot of technical details on how each technology works available. I wanted to take a slightly different approach, by talking to people who used the technology first-hand, and hear about their experiences. Thankfully the Computer History Museum in Mountain View is close to where I live and I had the opportunity to gather information from that amazing resource, as well as see some of the technologies live in person. I am also extremely happy that I had the chance to interview some very exciting people from the computing past. Other than basic information regarding how each technology worked and what it was, you will read about some of the stories and memories that Gordon Bell, David Macklin and Peter Capek shared with me firsthand. I was also able to take some input from a number of online users at a hardware site's message board, where I started a topic asking users about their experiences and memories, if they had some.

## 2. SYSTEM MEMORY

### 2.1 Drum Memory (1932)

Our journey begins in 1932, with Gustav Tauschek from Austria, inventing the Drum Memory. This type of memory was even used in one form or another during the 1950's and into the 1960's.

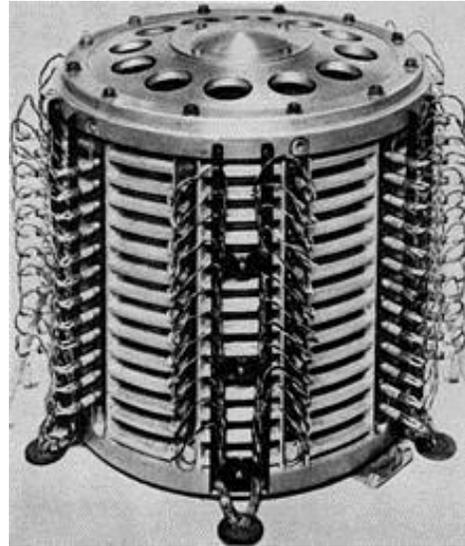


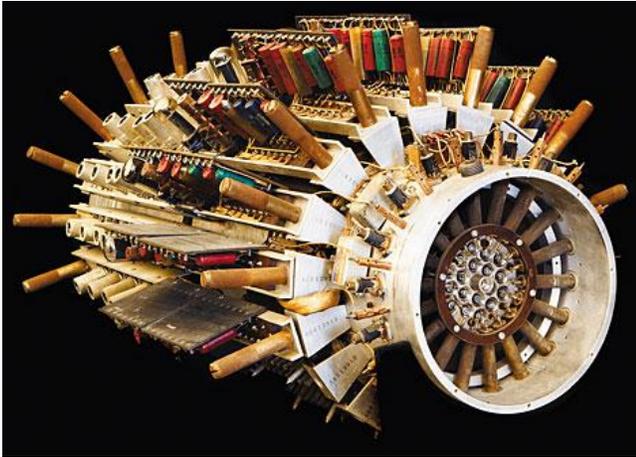
Figure 1. Drum memory.

There was actually a drum that was used for loading and reading data. This drum was a metal cylinder which was coated with recordable ferromagnetic material. The drum also consisted of a row of read-write heads that read and wrote the data. An interesting fact about drum memory that relates to today's UNIX systems is that in modern day BSD UNIX systems, `/dev/drum` is the name of the default virtual (swap) device. This comes from the historic use of drum memory as a backing storage of pages in virtual memory. More on the successor of this type of memory when we discuss magnetic drum memory below.

### 2.2 Delay Line Memory (mid-1940's)

This is the technology that fascinated me and led me to pick the topic on memory and storage in the first place. I was amazed at how far scientists back then went in order to store bits of data. Though this technology looks very unordinary to me now, it was simply a technique that worked in the 1940's, before more modern technologies were invented.

Electronic pulses (the bits or data) were transmitted through a cylindrical tube, most of the time filled with a liquid like mercury. Because of the physical properties of mercury, this formed a wave inside the tube that could be accessed again to describe the original pulse—and thus reading what was stored. This meant that in order to access a certain part of the memory, the system had to wait and use timing in order to acquire the desired part of the

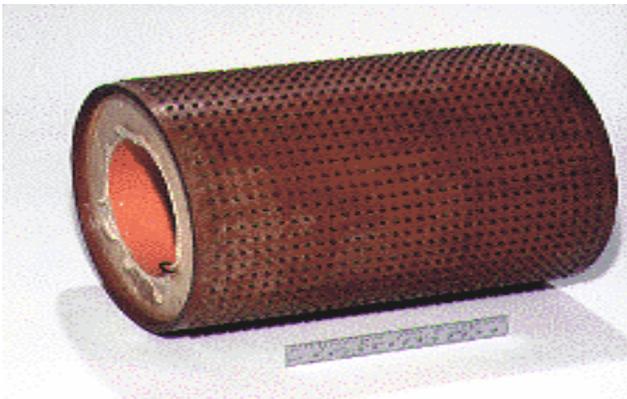


**Figure 2. Delay line memory**

pulse when it reached the end of the cylinder. This usually only took microseconds, but it was the only way to grab an exact location from memory, since the system was physically incapable of going out and directly fetching an exact location.

### 2.3 Magnetic Drum Memory (1947)

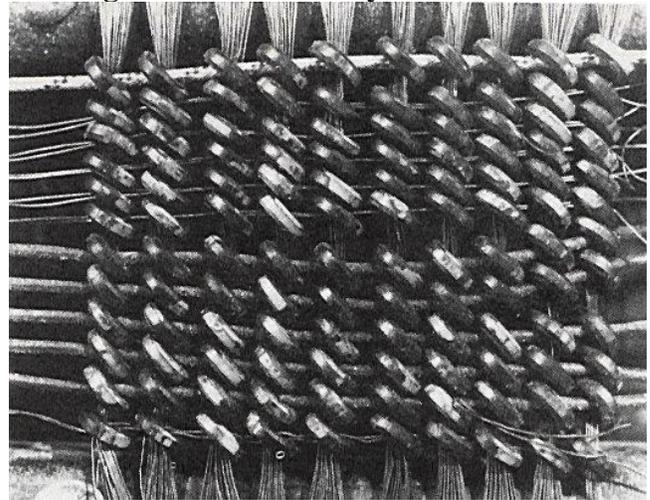
While rotating the drum, an electromagnetic pulse which represented a bit value could be stored by changing the magnetic orientation of the ferro-magnetic particle that the write head passed over at that very moment. Many machines used magnetic drum memory in the 1950's as what we call today main memory, to execute programs. David Macklin shared a story with me regarding his experience in using a magnetic drum memory machine during his employment at Republic Aviation on an IBM 650. Here is what he recalls: "The IBM 650, a drum machine which later had a disk file, RAMAC, was delivered to Republic Aviation in 1957."



**Figure 3. Magnetic drum memory.**

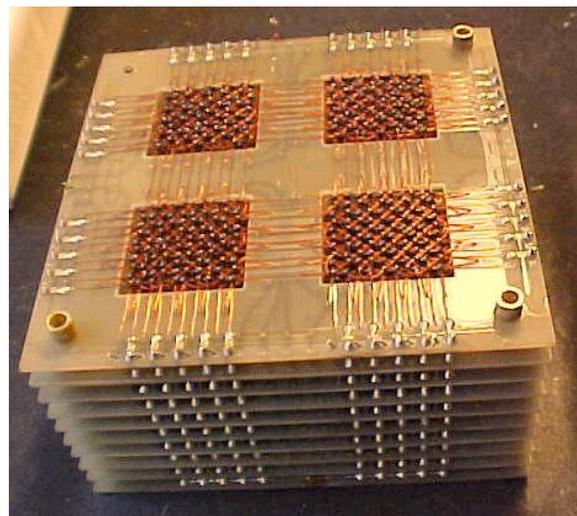
"There were 2,000 addressable memory locations in parallel tracks of fifty each. The programmer would have to provide the location of the subsequent instruction or let the SOAP assembler do that. I could average about three or four executions per revolution, depending on the earlier fixed locations of the operands. I simply added eleven to the location of the present operand. I did not use SOAP until I joined IBM later in 1957."

### 2.4 Magnetic Core Memory



**Figure 4. Magnetic core memory.**

This type of memory again, relied on magnetic fields to store information. It consisted of small rings capable of holding a magnetic charge, that were connected by wires which read or wrote bits by changing the magnetic direction. Each ring represented one bit. Magnetic core memory was the predecessor to dynamic memory that we have today, and had an enormous lifetime cycle for the time of around twenty years. Peter Capek, a long-time IBM employee shared his thoughts and memories on magnetic core memory. He began by stating that there is so much to say about it, but that also there is lots of info available so he went on about his experiences: "I did use a 7090 and 7094 with core. It was a large array perhaps 2 feet by 1.5 feet by about 9 inches, immersed in a bath of oil for cooling. The cores got progressively smaller with each generation of this memory, being barely visible at the end. But any technology which involves handling each core during manufacture can't compete with a technology that is basically a form of printing millions of bits at a time, so DRAM replaced core."



**Figure 5. Magnetic core memory stack.**

Costs in the early days of core were upward of a dollar a bit, but toward the end were under a penny a bit as manufacturing techniques got better and volumes increased tremendously.”

“One interesting aspect of core is that the magnetic fields from the wires (for writing and reading the state of the core) actually caused a mechanical force on the cores (just the way if you hold a magnet in each hand and bring them closer, both hand will feel the attraction or repulsion). This mechanical force could cause a core to vibrate or “ring”, particularly if the location in memory was repeatedly accessed at the right frequency. This caused problems with reliability in the case of Control Data’s 6x00 series machines, so much so that the memories were “potted” with a kind of resin that kept the cores from moving. This was done after the machine was delivered, but had been running unreliably. I think the oil I mentioned earlier may have avoided the problem, fortuitously.”

### 2.5. Twistor Memory & Bubble Memory (1968-1970)

Twistor memory was developed at Bell Labs around 1968, but just like bubble memory, it never made it out to large production quantities. It was built around the same concept of core memory, but did not use rings. Instead it used magnetic tape to store magnetic charges. Bubble memory was invented in 1970 by Andrew Bobeck, who was also part of the development of magnetic core memory and twistor memory. Bubble Memory used thin magnetic material to hold magnetized areas, of which each stored one bit of data. Both, Gordon Bell and Peter Capek stated that both technologies did not make it out of the lab. This was possibly because of other much better alternatives being developed at the time.

### 2.6 First Dynamic RAM (1960’s-1970’s)

The type of memory we use today began its journey in the 1960’s before it overtook every other memory technology in the 1970’s. Dynamic RAM is arranged in a square array of one capacitor and transistor per data bit. Unlike delay line memory with mercury, the system is able to exactly access an address location. Due to the physical capabilities of capacitors, accessing an address location requires that it be refreshed and rewritten back for future use. This also means that once the system is powered off, and the capacitors lose charge, the data is lost.



Figure 6. One of the first dynamic RAM modules.

Peter Capek describes his experiences with the first dynamic RAM: “My first experience was with the 370 model 145, which I

believe was the first machine with a semiconductor main memory. I recall that it had 128 bits on a chip, although I don’t recall if it was actually DRAM as we know it today. I think it was more like today’s SRAM, or static RAM, which is much less dense. It was not significantly faster than the core memory that it replaced, but by that time (ca. 1971-2) core had pretty much reached the practical end of its progress: it simply couldn’t get any faster/denser/smaller.”

“Obviously, the steadily increasing density and decreasing cost has had a huge impact on everything, though probably flash memory is equal in importance nowadays.”

## 3. SYSTEM MEMORY

Switching gears a little bit, let us examine some of the storage systems from the past. Gordon Bell, John Capek and David Macklin had even more to say about storage systems compared to memory, and they shared multiple interesting stories with me.

### 3.1 Punched Cards (1889)

Punched cards take us all the way back to 1889, before there even was an electrical computer. A punch card simply was a thicker than normal paper with holes punched though it in patterns that could then be interpreted. They were initially invented by Herman Hollerith in the 1890’s for the US census calculation. It was simply taking too much time to process data for an increasing US population. Later incorporated as an input device to computer systems, punched cards had an enormous life cycle which dated all the way up to the 1970’s. All three of my interviewees Gordon Bell, David Macklin and Peter Cepek had something to say about punched cards, and shared some interesting stories from their days. Here’s a quick excerpt from Gordon Bell, recalling his time using punched cards and how errors were corrected: “I just used these one year in while a Fulbright Scholar in Australia. I actually punched row binary data for 32 bit instructions on 12 rows. Mistakes were corrected by actually putting a bit back (1s went back to 0s), that you did very carefully and then copied that card.”

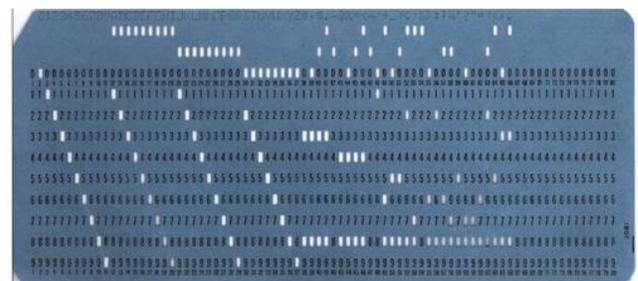


Figure 7. Example of a punched card.

Punched cards were used in more than just traditional computers, as we already noted above. David Macklin talked about using punched cards in the IBM Card Programmed Calculator (CPC). I also asked him if he had a funny story from the punched cards days, and he also shared one from the CPC days: “Starting in June of 1951 at Republic Aviation, faced with a large number of engineers demanding computing service a hundred different ways, I was tasked by Sam Pines to develop a ‘general purpose’ control panel for a CPC, not to be delivered for many months (thus not available for testing).”

“After listening to the more insistent 'users' and studying the manuals, I decided that they would need a floating decimal point. Some of their operands were fractional and tiny. Others were large integers, often mixed. Systems of linear equations with eight unknowns would have to be inverted. Logarithms and trig function would also be encountered.”

“The boards were wired and awaited the machine delivery. As I remember, now at age 93, testing went well. Errors and omissions were quickly addressed. The data cards and their input pencil forms were standardized. Education was simple. Problems such as flutter analysis were too large for the above approach. Jeanne Dupont developed a fixed, pre-punched instruction deck into which she would manually intersperse colored data cards. The decks would number many thousand. Jeanne's deck would run for hours, during which she would become more and more apprehensive. Cruelly, although harmlessly, I once put a few cards behind her monster, to spell out, on the 612 display: TILT TILT TILT...”

Peter Cepek shed more light on the punched cards days at IBM, including the business aspect, as well as more history about them and problems that he experienced. “I think it's important to keep in mind that these were invented for the purpose of storing data, long before using them for programming or a computer was on anyone's mind. They worked pretty well for that kind of application from about 1890 (when, more as paper than a card, they were used for the 1890 census in the US) until about the early 1970s, when tape and disk storage made them less than ideal. IBM used the term "unit record", meaning that each card stored the data about one unit (employee, part, line item in an order, etc.). It's also appropriate to remember that, in most case, the cards were a consumable. It depended on the application, of course, but in most realistic non-trivial applications, cards were punched (by hand, on a keypunch), additional cards may have been generated from those, they were used to print a packing list or bill, and then discarded when the transaction was complete. IBM made a lot of money selling cards—so much so that there were anti-trust implications...”

“Everyone who has ever programmed in the era of punched cards has had the experience of dropping a deck (or having someone else drop it for him!) and trying to put it back together. Some programs were literally boxes and boxes (2000 cards in a box about 14 inches long) and every time the program was run (lots of times, during debugging), it was always at risk of being dropped. It was possible, in most cases, to use the last 8 columns of each card to put sequence numbers in a deck to aid in restoring them after a drop, but this was more of a theoretical than a practical value. As the program evolved and cards were re-punched to edit the text on them, cards were inevitably inserted, or moved around (Oh! We have to do this test AFTER we set this variable, not before....), so sorting could provide, at best, a rough restoration.”

“But cards were good in that you could easily insert, delete, edit (with a keypunch), move around, pieces of your program. Of course, they were upper-case only, and with a somewhat limited character set. Key punches were very well designed and reliable machines, but a bit noisy. Key punches could be programmed (well, maybe configured is a better term) to do certain

rudimentary things, like automatic tab to a particular column, or duplicating fixed content from one card to the next, shifting automatically to numerics (rather than forcing the keyer to shift manually) for certain fields, and so on. Most programmers didn't bother, I think, although it always seemed to be of value to do it if you were going to punch more than just a couple of cards.”

“Of course, if the data on the cards was of any privacy/secretcy concern, the cards were not easily destroyed, but that was not often a big concern in my experience.”

Another interesting addition from Peter Cepek on punched cards was what happened that led to their invention. “The inspiration for storing data on cards, or at least as holes in paper, came from a train trip which Hollerith took across the US sometime in the 1880s. It was common practice for train conductors to "encode" a description of the passenger into punches in a ticket, to prevent the ticket from being reused or used by multiple people. So they might make a punch in the upper left for man and upper right for woman, somewhere else for tall/short, etc. Before about 1928, when the "standard" 80 column punched card was introduced (the size of a dollar bill at the time), cards were different for each application and really just an array of spots to punch application-specific holes. You can see this on images of the cards from the census, which shouldn't be hard to find.”

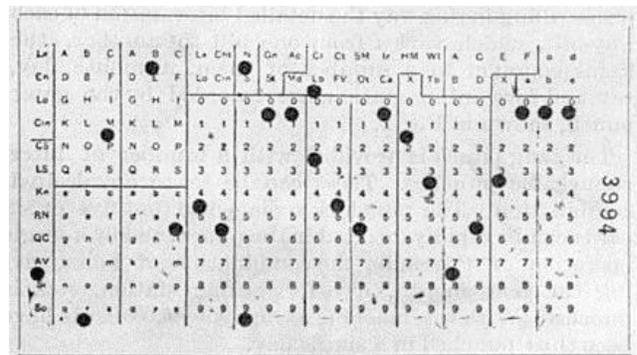


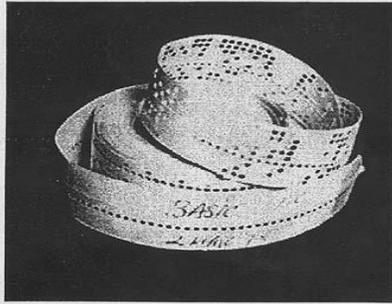
Figure 8. Example of US census punched card.

### 3.2 Paper Tape (1950's-1960's)

Moving on to another form of paper being used for data storage is the paper tape. It consisted of a long strip of paper where holes were punched at various locations to represent the data. Very much like the punched cards, paper tape worked under the same principle, except that it was possible for it to be rolled onto a drum.

Gordon Bell recalls using paper tape as the main input for the DEC computers, as well as his experience with the LINC personal computer: “We used paper tape as the main input for the DEC computers from 1960-1966. Paper tape was prepared both off line using a Flexowriter and On Line using the computer as an editor. The big change was the introduction of the Teletype 33 ASR (automatic send receive). It had a Teletype for printing, a tape reader, and a tape punch. Therefore for \$500, you got the whole I/O system for a computer.”

“LINC was the first Personal Computer and it created LINCtape, 256 KB, addressable tape that was used like modern disks or



**Figure 9. Paper tape.**

floppies in that it was useful for personal file systems. This was the dominant input until floppies became available.”

### 3.3 Magnetic Tape (1951)

After paper tape, magnetic tape followed in its footsteps. It worked much like the paper tape, except that it was able to store more information. It was spun on a roll and read/written to by a read/write head. The magnetic tape also made it into audio/video devices later on as the means of storing media. Amazingly, magnetic tape is still used throughout the world today for storing media.



**Figure 10. Magnetic tape.**

Peter Capek recalls the magnetic tape, including examples of machines that used it: “Workhorse medium from perhaps late 1950s until at least mid-80s. Evolved a lot, and still in use today. IBM and others built robust, fast, reliable tape drives. Evolved also in the way they stored data, starting with 7-track tapes (6 bits of data) and then switching to 9 tracks, around the time of System/360. There were substantial automatic tape libraries and handling systems as early as 1962 in special applications (the Tractor Tape system built for the Harvest System at NSA), and continuing well into the 90s and probably even today. I think they held their own in most environments until disk started to get really cheap and have other advantages (random access, for example). Of course, [magnetic] tape drives provided the most common image of what a computer was, because they were visually interesting.”

### 3.4 Magnetic Disk Emerges, (RAMAC) – 1952

Clearly the magnetic disk has one of the biggest impacts in terms of computing. The magnetic disk unlocked many possibilities for storing larger and larger amounts of data, which also in turn jump started the database field in computing. Previously, no one had thought of databases seriously, simply because there was no way that large amounts of data could be stored. RAMAC was the first computer to use a moving head hard disk drive. The disk drive worked very closely to what we have today, a spinning magnetic ring and a moving head used to read/write to it. David Macklin had worked on the design of the RAMAC and shared an interesting story during the design phase:

“I was directed to develop superior random addressing schemes for RAMAC. In departing from the sequential numbering of card and card-image (tape) files we had to develop means of ‘address conversion’ that were desirably direct, or, if indirect, would not wander all over the RAMAC. In a discussion with Bill Heising, I reported that no matter what technique I used, only about one third of the derived addresses were unique. He smiled and said I had discovered the reciprocal of the base of natural logarithms,  $e$ , and that Poisson had decreed the distribution I was finding:  $1/e$  of the area would never be addressed.  $1/e$  would be uniquely addressed.”

I should have known that, as a math major.”

Peter Capek also shared his experiences with the RAMAC, and an interesting story about a problem that the IBM team encountered: “My earliest experience is with a later generation, late 60s, when there were disk packs (in my case I think they held 5 megabytes!). They were a big step up from punched cards, although could still be frustratingly slow! Used, in the early days, mostly for storing operational software (OS, compilers, production programs), due to cost and the lack of adequate operating system support for any more sophisticated use.”

“I can provide one story that’s a bit funny, and which I believe to be true. At least, it came to me from a reliable source. In 1966 or so, IBM introduced the 2314. It was a system that had 9 drives (8 in use at a time, plus a spare), each capable of holding a 29+ megabyte disk pack which was removable, although quite heavy. The pack was 14 inches in diameter and held about 12 disks of aluminum (perhaps steel; don’t recall). It came with a clear plastic case to store it and remove/install it on the drive.”



**Figure 11. One storage disk showing head crash damage.**

“The first production machine was built in San Jose and installed for a customer in San Francisco. It had run in the plant for weeks without error before it was shipped. When it was delivered to the customer, they put new disk packs in and it ran fine... for about a week, when the heads started crashing and data was lost. Big disaster. They threw out all the packs, repair the crashed heads and tried it again... Again, after about a week, things started crashing again. This time they looked a bit more carefully and discovered an unexplained white powdery residue in the failed drives. They took it back to SJ and ran it again, but it ran fine for several weeks. SJ and SF are close enough so they were able to rule out environmental issues (i.e., air pollution).”

“After some time and analysis, they figured out what was happening. The white powder was dried glue... that had been used to glue onto the bottom of the customer's disk packs, labels that showed they had passed testing. After about a week of spinning, the glue was drying out and cracking. The packs they used in the plant were the same, but had never had the labels applied. Problem solved.”

Peter Capek's perspective on hard disks in the industry: “It took surprisingly long for the industry to learn how to use disks best. There were a lot of issues: reliability, protection, performance. Here's an example: When the first OS was written by Control Data's labs for the 6600 (the so-called "Chippewa" operating system), it was necessary to read through each disk file that the program had created and which was to be erased when the job ended. This was necessary because it was the only way to find out which disk blocks had been allocated to each file, so as to [de-allocate] them. The blocks were chained on disk, but that information was nowhere else.... at first. The effect of this problem, until it was fixed, was that a terminating job could hold up the entire system [worth of more than \$6M+] for tens of seconds until its file were freed.”

### **3.5 The Floppy Disk (1969)**

Finally, let's look at the floppy disk. It was invented by an IBMer to load microcode into disk controllers (3330 and 2305 disks). It provided portability to data storage, as it was easier to move floppy disks than hard drives. Originally a floppy was 8 inches, and shrunk with time as the technology improved. Peter Capek's opinion of floppies: “Not too bad a medium, though relatively bulky for the capacity. Even the 8inch size was used in a product for use directly by customers, called Displaywriter.”

## **4. CONCLUSION**

A lot more can be said about each of these technologies. But it is important to note that each one of them has had its contribution to computers as we know them today. I would like to extend a special thank you to Gordon Bell, David Macklin and Peter Capek for taking the time out of their schedules to talk to me and answer my questions. This piece would not have been the same without their real-life stories and experiences which they shared.

## **5. REFERENCES**

- [1] Mountain View Computer History Museum.
- [2] Interview from Gordon Bell.
- [3] Interview from David Macklin.
- [4] Interview from Peter Capek.
- [5] Computers Then and Now, Maurice V. Wilkes, Journal of the ACM; Volume 15 Issue 1, Jan. 1968