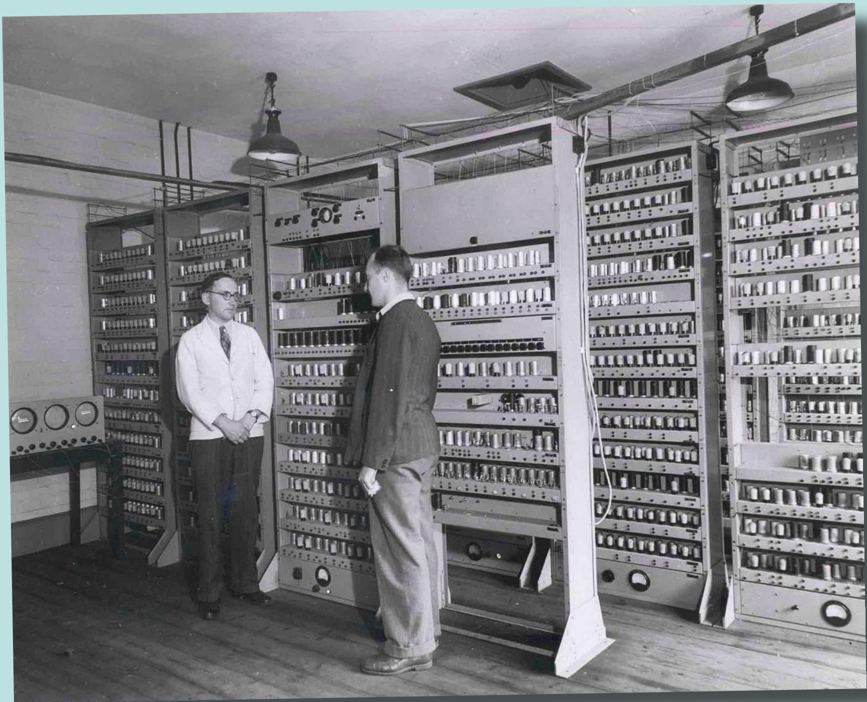
Celebrating EDSAC

Cambridge University's pioneering first computer



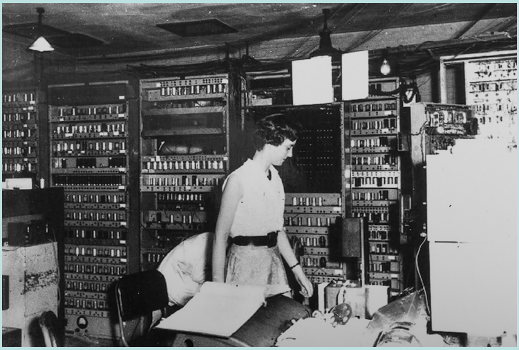
UNIVERSITY OF CAMBRIDGE

Department of Computer Science and Technology

Computing at Cambridge – driving research through to real world impact

We're delighted to be unveiling a Blue Plaque to commemorate the development of the EDSAC computer. When EDSAC ran its first program in May 1949, it did far more than prove a technical concept – it ushered in a new era of global scientific research.

Maurice Wilkes and his team built EDSAC as a practical tool rather than a mere experiment. As a result, the EDSAC became a vital workhorse for scientists across Cambridge, accelerating discoveries in multiple fields and contributing to **three Nobel Prizes**.

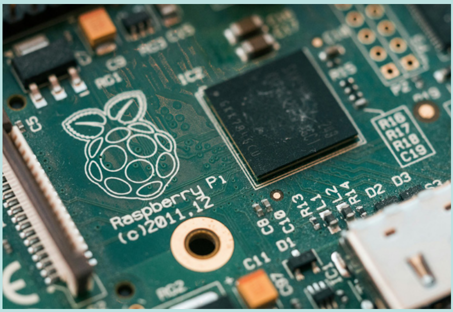


EDSAC also established the defining philosophy of the Department of Computer Science and Technology: to **undertake groundbreaking fundamental research and drive it through to significant real-world impact**. For in addition to its importance to computing and science in Cambridge, EDSAC was also a commercial success. The project was part-funded by J. Lyons & Co, a unique partnership that led directly to **the world's first business computer**, LEO I, which was based on the EDSAC design.

We proved early on that **academic brilliance and commercial enterprise can work well together**. This legacy thrives today as our staff and alumni actively collaborate with industry and launch world-changing companies. In the decades since EDSAC was first developed, our Department has continually transformed the global computing landscape.

We've made contributions to **the BBC Micro**, which brought computing into millions of UK schools and homes; **XenSource**, which created the foundations of modern cloud infrastructure; and the **Raspberry Pi**, co-founded by staff and alumni to democratise both computing and computer science education globally.

Today, our ecosystem of fundamental, disruptive research and spin-outs is as vibrant as ever. As you will see in this booklet, one of our latest projects – **CHERI** – is redesigning modern processor architecture. Working in tandem with industry leaders, we anticipate **it will significantly reduce the number of exploitable software vulnerabilities**.



The Blue Plaque for EDSAC is extremely important. It doesn't just celebrate an historic machine and an important moment in our history, but recognises EDSAC as a cornerstone of our approach to research and what our Department continues to achieve.

Professor Alastair Beresford, Head of Department

Front cover: Prof Maurice Wilkes and chief engineer Bill Renwick during the final stage of EDSAC's construction.

Left: Rosemary Hill, an early EDSAC operator. Above: a BBC Micro and a Raspberry Pi computer.

The birth of modern computing

One of the world's earliest computers was born at Cambridge University shortly after the Second World War.

EDSAC ('Electronic Delay Storage Automatic Computer') ran its first successful program on 6 May 1949. A huge machine, weighing two tons, it filled a whole room at what was then called the University Mathematical Laboratory – and is now known as the Department of Computer Science and Technology.

While another stored-program computer, 'the Manchester Baby', had been built at the University of Manchester 11 months earlier, it was experimental rather than being a practical scientific tool.

EDSAC was the first computer in the world to be fully operational and practical for general use.

Above: P Farmer, R Pigott, Maurice Wilkes & Bill Renwick with EDSAC 1947/8

Demonstrated at an open conference in November 1949, EDSAC began running a scientific computing service for the university from 1950. It was in regular day-to-day use by 1951, by which time its full programming system was in place, and was quickly adopted by the wider University for supporting research.

How EDSAC and Cambridge computing changed the world

Its development was led by computer pioneer Maurice Wilkes. He had worked in radar and electronics throughout World War Two and on his return home in September 1945, he was appointed

Head of the Mathematical Laboratory in Cambridge and tasked with providing computing expertise to the University to support its scientific research.

At the time, researchers were still using mathematical tables and mechanical calculators to carry out their calculations, which was very time-consuming. Wilkes studied the latest developments in computing machinery, especially from wartime advances made in the US, and gathered a team of experts together to form his new lab.

His vision was to use new computing technology to develop machines that would help Cambridge scientists, engineers and mathematicians in their work, equipping them with faster tools that simplified complex calculations.

Wilkes (*seen right with EDSAC memory delay lines)* also wanted the machine to be useful and accessible to as many researchers as possible. So he used simple conservative design principles in order to bring the machine into service as soon as possible and learn from having it used by real users.

A revolution in scientific research

EDSAC could execute approximately 650 instructions per second. Modern computers can compute millions of times faster, but in the late 1940s and 1950s EDSAC offered a massive improvement on what was possible at the time.

It is said to have increased productivity by 1,500 times, and it revolutionised scientific research at the University of Cambridge. Problems that had previously been impossible or impractical to attempt could now be solved.

Computers as everyday tools

The successful operation of EDSAC ushered in the use of computers as everyday tools. Prior to this, computers had only been used experimentally or to solve specific problems, such as Alan Turing's machine for breaking the Enigma code.

EDSAC was the first computer to provide general computation and practical support for mathematical and scientific calculations. It was superseded by a more advanced machine, EDSAC 2, in 1958.

A significant number of researchers benefitted from the revolutionary computing power of EDSAC which they were able to take advantage of at Cambridge. This included the winners of three Nobel Prizes: John Kendrew and Max Perutz (Chemistry, 1962), Andrew Huxley (Medicine, 1963) and Martin Ryle (Physics, 1974). They all acknowledged the role EDSAC had played in their research in their Nobel Prize acceptance speeches. EDSAC also inspired the teaching of the first Computer Science course in the UK. The Diploma in Computer Science started here in 1953 and used the EDSAC.

Above: Bill Renwick with 5 hole tape reader and Creed teleprinter.

The architecture of EDSAC

Anyone who has programmed a modern computer at the assembly language level would immediately understand how to program EDSAC.

This is because nearly all computers since then have been designed the same way, with the same memory housing program and data. (The exceptions are the so-called Harvard machines, commonly still used for DSP (digital signal processing) applications, where the code and data sit in different address spaces. These take their name from the earlier electronic computers built elsewhere.)

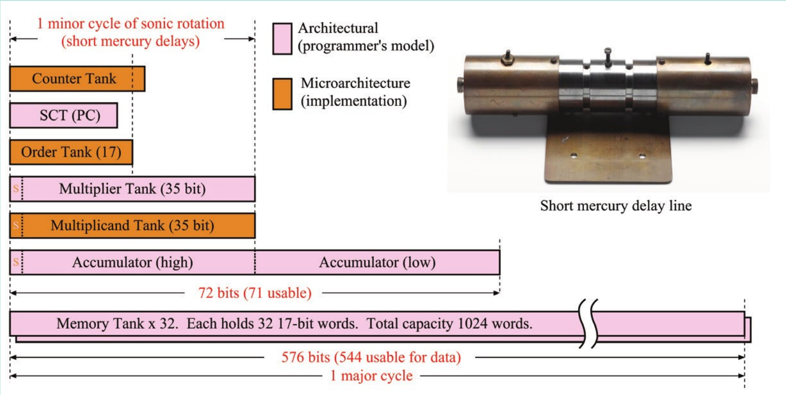
In 1946, Maurice Wilkes attended the Moore School lectures. These were held at the University of Pennsylvania's Moore School of Electrical Engineering to share then-current knowledge and progress in electronic computation, a field in which the Moore School was a leader. It was here that mathematician and polymath John von Neumann presented his now famous paper, 'First Draft of a Report on the EDVAC' – Electronic Discrete Variable Automatic Computer – which was then being designed.

After sailing back to England on the Queen Mary, Wilkes assembled a team in Cambridge that implemented the EDSAC, embodying all of the EDVAC's key principles. Remarkably, the EDSAC was working and providing a service to the University several years before the EDVAC was fully functioning in Pennsylvania.

EDSAC was an accumulator machine. This means that all arithmetic and logic operations operate on a single register, called the accumulator. The accumulator can be loaded from memory or stored to memory. The only other principal register was the program counter, which was called the Sequence Control Tank.

There were no index registers, with indexing an array being implemented using self-modifying code: in other words, using the operand field of an instruction as the index register – something that would not be possible on a Harvard machine.

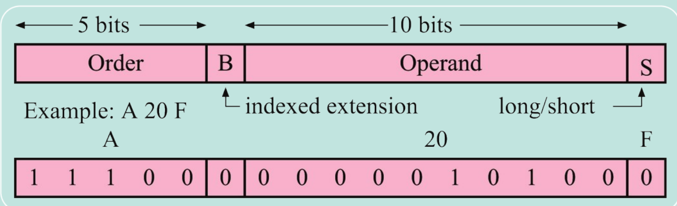
Single-precision operation treated the 17-bit word as a two's-complement number with the binary digit at the extreme right for integer purposes and just after the sign bit for fixed-point fractions. When using



Above: Programmer's Model and Microarchitecture Registers (Tanks)

double words, a further 18 bits of precision became available. Hence values consisted of a sign bit and a 34-bit value. A fraction of 34 bits was chosen to give a numerical precision of 10 base-10 digits. log(2^34)=10.235.

Those who started programming with simple microprocessors in the 1970's-80's may intrinsically think that the early computers likewise had low native precision and that long arithmetic would routinely be used. A mantissa precision of 10.2 digits is a digit better than the IEEE standard for single precision widely in use today. But it was fixed-point. The EDSAC 2 took things much further seven years later, being floating-point, but for EDSAC, calculations using large and small physical quantities, like Boltzmann's constant or Avogadro's number, had to be manually scaled at program design time. Input was from a paper tape reader and output was to a teletype or paper tape punch.



Above: the EDSAC instruction format, and an example ('add the word at location 20 to the accumulator'). Changing the suffix to D flips the last bit and a double word (two successive store locations) will be added. This provides 34-bit precision since a hidden 'sandwich' bit becomes available.

Op	Address	Short	Long
A	n	A += m[n]	AB += w[n]
S	n	A -= m[n]	AB -= w[n]
H	n	R = m[n]	RS = w[n]
V	n	AB += R*m[n]	ABC += RS*w[n]
N	n	AB -= R*m[n]	ABC -= RS*w[n]
T	n	m[n]=A; ABC=0	w[n]=AB; ABC=0
U	n	m[n]=A	w[n]=AB
C	n	AB += R&m[n]	ABC += RS&w[n]
R	$2^{(n-2)}$	ABC >>= n	
L	$2^{(n-2)}$	ABC <<= n	
E	n	if A>=0 goto n	
G	n	if A<0 goto n	
I	n	m[n]=rdch()	LS 5 bits
O	n	wrch(m[n]>>12)	MS 5 bits
F	n	Verify last output	
X		No operation	
Y		Round ABC to 34 bits	
Z		Stop and ring the warning bell	

The 1949 EDSAC Instruction Set

Microarchitecture (implementation)

The EDSAC was a bit-serial computer, so all data busses were just one-bit wide. The datapath is exactly what you'd expect. The main memory (shown in the figure over the page) was split over the 32 separate mercury delay lines that implemented it, whereas today we would show the main memory as one block, regardless of how many chips are used to implement it.

The 'order tank' contains the currently being executed instruction whereas the 'sequence control tank' is the program counter. The stored-program concept is manifest in the route from the memory to the order tank.

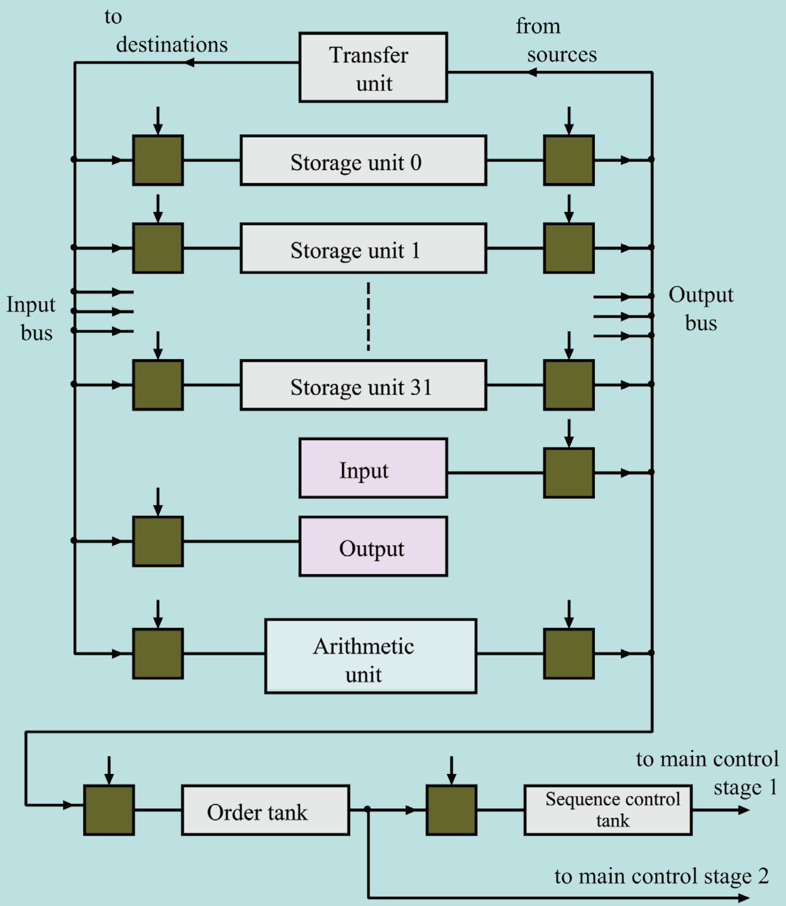


Figure 2.3 -- Numerical interconnexions in the EDSAC.

A figure from M.V. Wilkes, Automatic Digital Computers, Methuen, 1956.

Bootstrapping and Initial Orders

The ENIAC computer, which commenced operation in 1945, was not stored-program. Instead, banks of massive plugboards on wheels were manually configured and rolled around the computer room. These were bespoke pieces of hardware, whereas for EDSAC, user programs were punched on a 'perforator', of which there was one in the Mathematical Laboratory. Moreover, the punched data was human-readable, essentially being a full program listing. This was probably the most novel aspect of EDSAC – and certainly led to it being really usable and successful as a service machine.

Stored-program computers generally have a fixed program that starts up the machine, getting it ready to load and run a user program. EDSAC is no exception. It used a 'boot ROM' that was implemented using four uniselectors providing a total capacity of 42 instructions. These contained the famous Initial Orders, which were hardwired by soldering wires.

The EDSAC Initial Orders are etched into the walls of the common room in the William Gates Building.

The Initial Orders loaded the user's program from paper tape. The user appended copies of private or standard subroutines to their tape to generate a full executable image that could be loaded in one go.

The Initial Orders mapped symbols and base-ten multi-digit numbers to binary (what we would today call assembly). It also implemented relocation, where any instruction suffixed with a theta symbol was adjusted by the current load origin. A further 11 suffixes were also available for programmer or library use, enabling fixed offsets in data structures to be augmented by the base address of that construct in memory. Hence many of the concepts of subsequent link editors or segmented memory structures were present.

Since all memory addressing was absolute, and no compilers existed, this load-time relocation was the vital ingredient to making subroutines portable and reusable.

Debugging

One of the most significant issues that arose was the unexpected one of ‘getting the program correct’. Today, we call it ‘debugging’.

Many test programs were run on the hardware so that the hardware team could be confident the machine worked correctly. For hardware debugging, bit-by-bit visibility of the data in the machine was a must-have (just as it is with today’s silicon computers). For EDSAC, processor and main memory were visualised on several CRT monitor tubes, with a selector switch connecting a CRT to a given tank of memory. This gave a real-time view of execution. (Later on, computers would often have a light on the control console for each bit of every processor register, giving rise to the classic Sci-Fi style image of an early electronic brain.) Single-step via the operator’s console was supported, with one instruction being executed per button press. The CRT displays are then much more usable, not changing all the time!

Electronic technician S Barton with the display used by the operator.

Everyday users or programmers would not use this difficult and time-consuming process. They had confidence in the hardware and had to find their own software bugs. A set of standard ‘post mortem’ routines were available to print what would later be called a core dump. Since the printer was so slow, variations of these ‘peeping’ routines were available that were more selective in what they showed.

The most advanced de-bugging technique was to load EDSAC with an interpreter for itself and trace the execution of its own code on the output teleprinter. This again was made rather easy to configure, loading the interpreter as yet another library routine, but transferring control to it rather than the actual program. Today, you can try this simulator inside the EDSAC simulator!

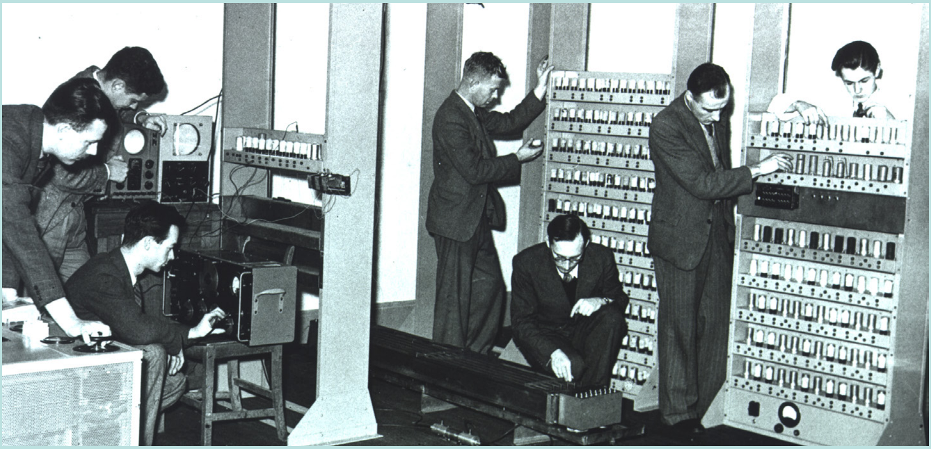
Dr David Greaves

The EDSAC Replica

EDSAC was a triumph of early British computing technology – but Maurice Wilkes and his team of co-creators had no idea how iconic it would become. So after its successor EDSAC 2 came into operation in 1958, EDSAC was simply shut down and dismantled because of a lack of space. Many of its parts were sold for scrap.

In hindsight, this was a terrible loss. So in 2011, at the suggestion of Dr David Hartley (a former director of the Cambridge University Computing Service) and Dr Hermann Hauser (the well-known Cambridge entrepreneur and inventor), a project was set up to create a replica of EDSAC. This is based at the National Museum of Computing at Bletchley Park.

Led by Dr Andrew Herbert, a former head of Microsoft Research in Cambridge, the project aims to improve our understanding of the Cambridge computing heritage, recover lost technical expertise, and learn about the challenges faced by Cambridge computing pioneers.



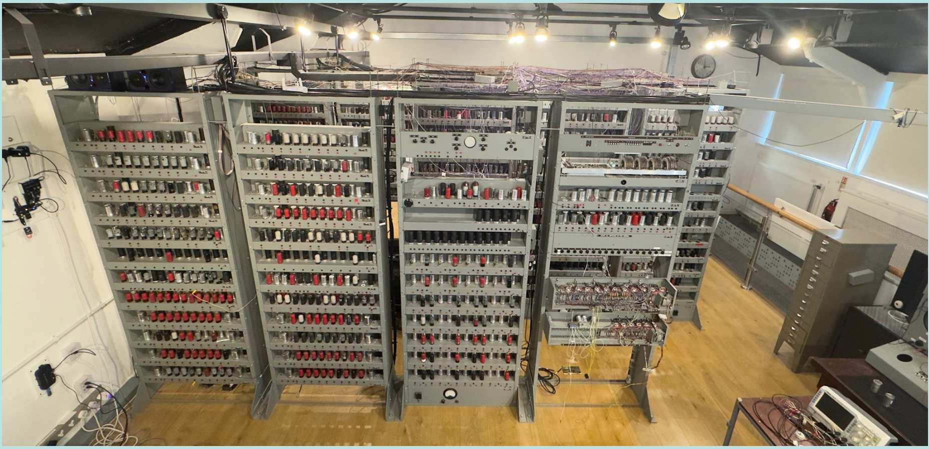
Recreating EDSAC

Reconstructing EDSAC has been a significant challenge, not least because of its size. Like the original, the replica weighs a startling two tons. It comprises three rows of tall racks containing 142 panels of electronics (see one pictured below), comprising 120 different types, using around 3,500 thermionic valves, interconnected by miles of overhead wiring. It consumes 9kW of electrical power.

EDSAC also challenges the modern electronics engineer. The machine processes data serially, and is asynchronous. Additionally, its circuits are thoroughly analogue in nature as they drew on the radio and radar experience gained by Maurice Wilkes and his team during World War II. Most circuits are A.C. coupled, so rather than fixed levels for logical 0 and 1, these vary circuit by circuit, with the convention that the voltage between them should be 20V or more.

To add to the challenge of recreating EDSAC, only a few of the original circuit diagrams have been found. And the surviving documentation is patchy and inconsistent because EDSAC was frequently updated during its life. Fortunately, a good number of high-resolution glass plate photographs of EDSAC were taken during its initial construction. Preserved in the University Library, they were able to guide those working on the project.

Dr Andrew Herbert



The EDSAC replica

Computer Architecture post EDSAC

The EDSAC design was a masterclass in building a highly functional machine given the constraints of the technologies of the day.

Maurice Wilkes was well-versed in these. While working on radar during World War Two, he'd learnt about vacuum tubes – active electronic components that were the precursor to the transistor. He also knew about mercury delay lines: they were used in radar to store the responses from one radar sweep so they could be subtracted from the next sweep, thereby separating moving objects from the background. Wilkes realised that mercury delay lines could be used as a bit-serial recirculating memory and computed on the bits as they appeared from the delay line memories.

He was also aware that these delay lines were made by the Cavendish Laboratory (the University of Cambridge Physics Department), so he knew exactly where to source them. Wilkes also made use of telecommunications equipment including teleprinters and punched paper tape.

Early days: the paper tape jobs queue

Punched paper tape was the medium on which computer programs for EDSAC were provided. When they were ready to be run, they were inserted into the job queue: a physical system of hooks and clips to hold the tapes.

Margaret Marrs, who worked with EDSAC 2, holding the paper tape used to feed data.

The priority of jobs was determined by the importance of the work or the person submitting the program, with students running programs in the middle of the night. One apocryphal story tells of a research student who fell asleep asleep while EDSAC ran his program – and woke up to find that most of his tapes had been cleared away by an over-zealous cleaning lady arriving early in the morning!

When one program finished, an operator took the next highest job off the the washing line and loaded it via a punched paper tape reader. Programs typically finished by ringing the bell, which interrupted the operator and got them to schedule the next task.

In later machines this process was automated: the human operator became a software operating system, the physical queue became an in-memory job queue, scheduling was determined algorithmically, and the paper tape reader was replaced by a linker/loader. The bell was replaced by an interrupt mechanism, and a clock was used to interrupt the running program every epoch to give the operating system an opportunity to time-slice between programs to provide concurrent operation on a single processor core.

The 1960s – time sharing machines

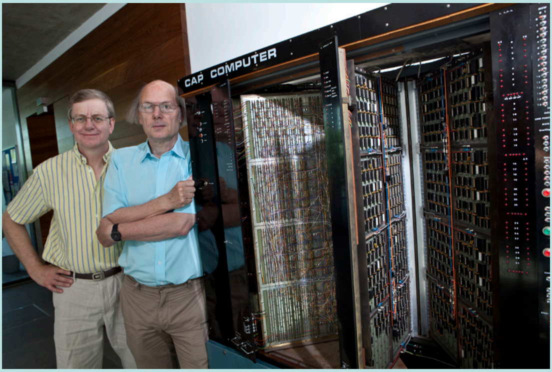
By the 1960s, computers were still very expensive so time sharing was common practice. Work at the University of Manchester on the Atlas computer introduced virtual memory: this was used to allocate and partition memory in page (3072 byte) chunks between users providing course-grained isolation between programs from different users.

Atlas was a very expensive computer and Ferranti gave parts of one to the University of Cambridge with the understanding that a less expensive version would be developed. The resulting machine – Titan (*pictured right*) – simplified the Atlas design, including removal of the complex page-based virtual memory.

Titan had many use cases, including the computation of the inverse Fourier Transforms for the One-Mile Radio Telescope at the University of Cambridge Mullard Radio Astronomy Observatory.

The 1970s – the CAP Computer

Given the limitations and complexity of virtual memory, Maurice Wilkes and Roger Needham explored the use of capability-based memory protection in work that resulted in the development of the CAP Computer in the early 1970s.



Andrew Herbert (left) and Bjarne Stroustrup with the CAP Computer, which they used extensively as research students.

The original 1960s concept of a capability was an unforgeable token of authority that could be delegated. For CAP, capabilities were access permissions to regions of memory with a program being provided a set of capabilities authorising the memory regions it could use. This proved to be a powerful security mechanism but was complex to implement in hardware. Several manufacturers (Burroughs, IBM, Intel and others) also tried building machines with capability-based memory protection – but they all proved rather complex and unsuccessful, leaving virtual memory machines as winners, despite their shortcomings.

Networking

The 1970s also saw great interest in computer-to-computer networks and the birth of the Cambridge Ring network led by Andy Hopper, David Wheeler, Maurice Wilkes and Roger Needham.

Research on local area networks transitioned to asynchronous transfer mode (ATM) switched networks under Ian Leslie and Derek McAuley's Fairisle project. Mark Hayter brought ATM networking inside the machine to form the Desk Area Network (DAN), a precursor to networks-on-chip used today. In the 2000s Andrew Moore led work in the department on the NetFPGA project, an open source hardware and software platform for network education and research in collaboration with Stanford University.

Computer Architecture Group founded
By 2000, the department had not designed a processor for decades with industry leading the charge on design. But due to technology scaling issues, Simon Moore and Robert Mullins realised that the computer industry was going to have a more difficult time. So they formed the Computer Architecture Group to explore this research space, starting with self-timed processors, networks-on-chip and many-core energy efficient processors (Loki).

This group has grown in breadth and technical depth as new professors joined: David Greaves (hardware tools, verified processors, etc), Timothy Jones (microarchitecture, automatic parallelisation, robustness, etc), Robert Watson (architecture, security, operating systems, applications, etc.), Tobias Grosser (compilers, verification, etc) and Prakash Murali (quantum computing).

CASCADE Computer Architecture research centre
And work in this area took another step forward in 2025 when a dedicated research centre was established. CASCADE, the Computer Architecture and Semiconductor Design Centre, focuses on research to address some of the grand challenges in computer architecture, design automation and semiconductors – a critical area of computing that underpins today's technologies and drives the next generation of computing systems. It has expertise across the breadth of the area, and has already garnered significant industrial support to fund a number of PhD students in this field.

The CHERI project
In 2010 Robert Watson and Simon Moore started the CHERI project with Peter Neumann from SRI International. The aim of the project, which has gone from strength to strength, is to explore how to make computers fundamentally more secure.

Like CAP, CHERI is a capability machine but reimagined for the modern era. CHERI capabilities are bounds checked pointers with provenance and permissions that provide

The Arm Morello chip (CHERI-on-Arm)

strong temporal and spatial memory safety, and highly efficient compartmentalisation. Protections are enforced by the instruction set architecture that has been formally verified by Peter Sewell's team to ensure correctness.

In collaboration with Arm – and with the support of the UK government's Digital Security by Design programme – the prototype Morello chip was produced for 1,000 desktop computers, aiding evaluation. In 2026 SCI Semiconductor, SECQAI and Codasip announced CHERI products with more to come.

From the coffee pot webcam to the Raspberry Pi
The Department has so many other hardware-related stories to tell that can only be hinted at here. The 1990s brought the famous coffee pot webcam, and the Pandora multimedia computer; location systems like the Active Badge, and numerous sensor systems including zebra net (on real zebras!). There have been new output devices, from the auto-stereo display project to more recent work on displays at the resolution, luminance and chrominance limit of the human eye. Research on collective intelligence for multi-robot systems, and explorations of human-robot interaction. Probably best known is the Raspberry Pi, which began life when Department staff and alumni decided they needed a way to promote computer science and educate a wider population about it. It is now the best-selling and most successful British computer.

Prof Simon Moore

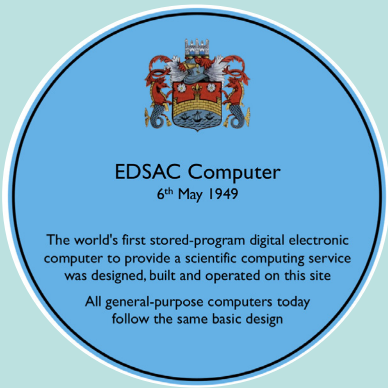
Members of the CHERI project team in spring 2026, celebrating the project's 15th birthday.

The Blue Plaque

In summer 2026, the blue plaque will be installed in the location where EDSAC was originally built. It will go on the outside wall of what is now the University of Cambridge Babbage Lecture Theatre – an entirely fitting place, as Charles Babbage is considered to be the grandfather of computing.

The plaque has been created by the charity Cambridge Past, Present and Future. Their Cambridge & District Blue Plaques mark the people and events that have shaped the city and surrounding villages, making history visible in the streets where it happened.

Mike Hakata, chief executive of the charity, says: “Every day, billions of people benefit from technologies whose origins can be traced to pioneering computing work carried out in Cambridge. The EDSAC represents a turning point in that story, when computing moved from theory into practical reality.



“Through our Blue Plaques programme, we aim to connect people with the places, people and ideas that have shaped Cambridge and changed the world. We are delighted to recognise this extraordinary achievement and ensure its story remains visible for future generations.”