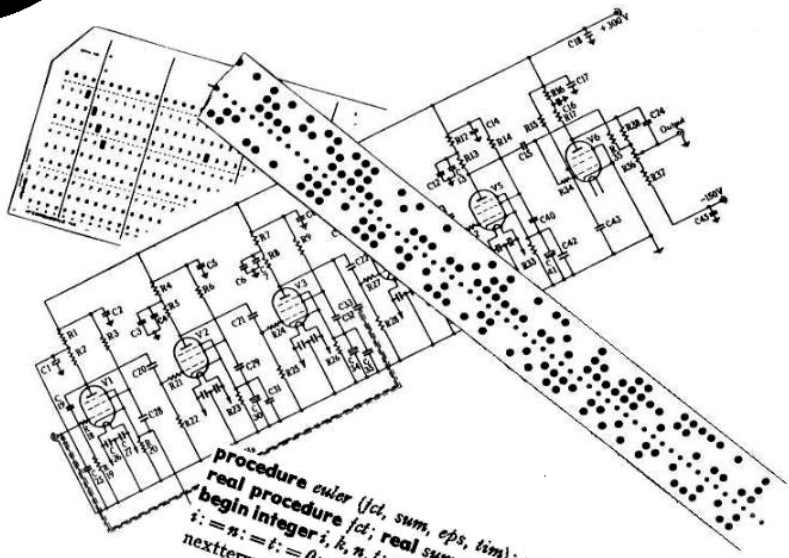
**Issue Number 65** **Spring 2014**

The Chartered Institute for IT

# RESURRECTION

The Bulletin of the Computer Conservation Society



```
procedure euler (fct, sum, eps, tim); value eps, tim; integer tim;
real procedure fct; real sum, eps;
begin integer i, k, n, t; array m[0:15]; real mn, mp, ds;
i:=n:=t:=0; m[0]:=fct(0); sum:=m[0]/2;
nextterm: i:=i+1; mn:=fct(i);
for k:=0 step 1 until n do
begin mp:=(mn+m[k])/2; m[k]:=mn; mn:=mp end means;
if (abs(mn) < abs(m[n]))∧(n < 15) then
begin ds:=mn/2; n:=n+1; m[n]:=mn end accept
else ds:=mn;
sum:=sum+ds;
if abs(ds) < eps then t:=t+1 else t:=0;
if t < tim then go to nextterm
end euler
```

# Computer Conservation Society

# Aims and objectives

The Computer Conservation Society (CCS) is a co-operative venture between BCS, The Chartered Institute for IT, the Science Museum of London and the Museum of Science and Industry (MOSI) in Manchester.

The CCS was constituted in September 1989 as a Specialist Group of the British Computer Society. It is thus covered by the Royal Charter and charitable status of BCS.

The aims of the CCS are:

◊ To promote the conservation of historic computers and to identify existing computers which may need to be archived in the future,

◊ To develop awareness of the importance of historic computers,

◊ To develop expertise in the conservation and restoration of historic computers,

◊ To represent the interests of Computer Conservation Society members with other bodies,

◊ To promote the study of historic computers, their use and the history of the computer industry,

◊ To publish information of relevance to these objectives for the information of Computer Conservation Society members and the wider public.

Membership is open to anyone interested in computer conservation and the history of computing.

The CCS is funded and supported by voluntary subscriptions from members, a grant from BCS, fees from corporate membership, donations and by the free use of the facilities of our founding museums. Some charges may be made for publications and attendance at seminars and conferences.

There are a number of active projects on specific computer restorations and early computer technologies and software. Younger people are especially encouraged to take part in order to achieve skills transfer.

The CCS also enjoys a close relationship with the National Museum of Computing.

# Resurrection

## The Bulletin of the Computer Conservation Society

ISSN 0958-7403

## Number 65
## Spring 2014
## Contents

# Society Activity

## National Museum of Computing — *David Hartley*

### Storage - Bilton Road

The acquisition of storage at Bilton Road, situated one and a half miles from the Museum, has proved to be a substantial asset and has enabled us to house much from H Block and from other substandard space elsewhere on the Park.

### Refurbishment

Considerable refurbishment is taking place in the Large Systems Room (LSR) and the Multipurpose Room (MPR). In the LSR, the WITCH has been temporarily moved in order to improve its display facilities. A new workshop area has been created, whilst the space for the EDSAC Replica is being prepared in the other part of the room. The IBM 1130 has been relocated to join the Elliott machines in the adjoining room, and the Air Traffic Control display has been moved to the MPR.

### Opening times

TNMoC is now open to visitors on Sundays, Thursdays and Saturdays; this has been the de facto position through most of 2013; it is now official.

### New acquisitions

Plans are being laid to scan the library of ICL Aperture Cards which are a valuable archive resource. Suitable equipment is being investigated, for which it may be necessary to seek funds.

Consideration is being given to whether we can accept a donation of a Cray 1, and possibly the HEC1 currently in Birmingham. The latter would not be a working exhibit but might sit alongside the EDSAC and the WITCH.

### Stop Press *[ed]*

At the end of January a dispute between TNMoC and the Bletchley Park Trust which has been brewing for some time burst into the open as a result of a BBC News report which can be viewed at *www.bbc.co.uk/news/uk-25916048*. A comprehensive summary of the dispute may be found at *tinyurl.com/tnmocvsbpt*. CCS is closely related to both TNMoC and BPT and feels that it would be inappropriate at this stage to comment. We hope the dispute can be resolved without further adverse effects to both parties. In the meantime, we feel obliged to let members know what is going on.

## **EDSAC Replica —** *Andrew Herbert*

The project has secured an additional £50,000 of funding from Mike Lynch and his wife, bringing the amount raised to 75% of the total budget of £250,000. Project expenditure is currently less than budget.

Construction of EDSAC chassis continues apace — approximately 55 chassis are completed or under construction out of a total of approximately 142 required.

A volunteers' meeting took place at TNMoC on 18th December. Two new volunteers have joined the project, Martin Evans and David Milway, both based in Cambridge and experienced in electronics design and construction.

Mass production was started on chassis type 01 — storage regeneration — but had to be halted following the discovery of short circuit problems and difficulties with electrical noise. Chris Burton is investigating as this is a priority item to enable progress in other areas, since delay lines are used for machine registers as well as the main store.

Peter Lawrence and John Sanderson assisted by Andrew Herbert have completed commissioning of the clock pulse generator and five type 1 digit pulse generator chassis. John is now building the remaining two type 2 digit pulse generators required to complete the clock and digit pulse generation system. Peter Lawrence is investigating clock pulse distribution across EDSAC as a whole. Nigel Bennée has a working adder and accumulator shifting unit: his attention is now turning to the rest of the arithmetic unit design (known in EDSAC terminology as "the computer"). Andrew Brown has been investigating uniselectors and has a design prototype for the initial instruction unit. There are still open design questions relating to how the orders delivered by the unit are injected into the store and the engineers' panel for injecting fixed patterns into the order decoding system. James Barr continues to develop a design for main control —- that is order fetch and decode. The decoder is now substantially complete. John Pratt has been investigating the coincidence system for synchronising the computer to the store. Peter Linington has robust working prototypes for the short delay lines used as registers and a design for the wooden "coffins" to hold them. Long delay lines for the main store remain a topic of investigation — a different composition for the wires is being tried to see if this improves performance. Alex Passmore has completed the design of the AC and HT DC power distribution system.

A design has been prepared for the layout of the EDSAC gallery at TNMoC and as this report is being written contractors are clearing the space as part of TNMoC's overall refurbishment of the Large System Gallery. In the space available it will

be possible to set the replica up in the same configuration as the original machine in 1949.

Bill Purvis and Simon Moore have developed a hardware implementation of the EDSAC logical design, developed by Bill using his ELSIE simulator. The ELSIE input is converted to Verilog which in turn is compiled to configure a Field Programmable Gate Array (FPGA). The FPGA implementation runs EDSAC programs successfully and has the potential for development as a test harness for replica chassis.

Andrew Herbert and Martin Campbell-Kelly have been investigating published EDSAC programs. From the EDSAC99 celebrations we have reconstructions of the first "table of squares" program and the "primes" program from the November 1949 Cambridge conference. Martin has reconstructed a partially debugged version of Wilkes' Airy Integral program. Andrew has found an improved version due to Don Hunter that produces identical results to those published by Wilkes. Also from Don Hunter have come programs for Chapman's Grazing Incidence Integral, also published by Wilkes, a program to solve simultaneous different equations from Dodd and Glennie's Armament Research Establishment report on EDSAC, and versions of several of the example programs in Wilkes, Wheeler and Gill. We therefore have a considerable body of software ready to run when the machine is working and have recovered many of the useful routines from the EDSAC library. Using various Elliott 903 computers we have the means to punch EDSAC five-hole tape.

## **Harwell Dekatron —** *Delwyn Holroyd*

The machine has continued to work well with no major problems to report. We have experienced one or two baffling issues which cleared up of their own accord, such as the Add and Clear order skipping through the arithmetic stage, and thus acting as a No-Op instead. The Creed reperforator is occasionally mispunching and skipping rows and will require some attention.

At the time of writing the Large Systems Gallery at TNMoC which houses the machine is being refurbished. The machine has been dismantled and packed away whilst this takes place. It was re-assembled in its new home on the weekend of 18th January. The new display provides much improved lighting, a new wall mounted arrangement for the tape readers and keyboard perforator, more space for display materials, and greater visibility of the output table peripherals. A new dedicated workshop is under construction in the area behind the HDC.

## **Analytical Engine —** *Doron Swade*

The major development since the last report has been the award of a Leverhulme research grant for a study of Babbage's use of his Mechanical Notation in his design process. The grant is for three years and has been awarded to Professors Adrian Johnstone and Elizabeth Scott, Centre for Software Language Engineering, Computer Science Department, Royal Holloway, University of London. The grant will in part support Doron Swade's study of Babbage's Mechanical Notation, a symbolic descriptive language of Babbage's own devising that describes the mechanisms of his computing engines. See *tinyurl.com/rhcbabb*.

The research project, 'Notions and notations: Babbage's Language of Thought', addresses a central question: how Babbage was able, in the 19th century, to devise systems with levels of complexity that were not matched for 100 years – until the early integrated circuit era. The project aims to test the hypothesis that Babbage's Notation was key to the design process allowing him to manage high levels of complexity independently of the mechanical details. Software tools will be developed to examine the Notation and its effectiveness in driving simulations of the designs — this as an intriguing alternative to traditional modelling i.e. as an alternative to creating virtual mechanisms as direct equivalents of their physical counterparts.

The Leverhulme project relates specifically to Difference Engine No. 2, not the Analytical Engine so its relevance to the Analytical Engine (AE) project requires some clarification. Difference Engine No. 2 (DE2) was built using Babbage's mechanical drawings as the primary informational source i.e. without substantial reference to the Notational description. The physical DE2 is being used as a kind of Rosetta Stone to decode the Notation as used by Babbage in the late 1840s. The relevance of the Leverhulme project to the attempt to build an Analytical Engine is twofold: the knowledge it will yield of the Notational language will allow us to more fully understand and interpret the AE designs from the Notational descriptions — this to mine logical information that might not be included in or recovered from the mechanical drawings; secondly, if the Notation can drive a simulation of a mechanism then the techniques developed to do this will yield an invaluable and possibly revolutionary tool to simulate the Analytical Engine, an essential staging post to an eventual physical build,

The Leverhulme project makes provision for the recruitment of a post-doc researcher. In the broader context of Babbage studies the recruitment of a post-doc is seen as a welcome first move towards succession planning and the retention of expertise — especially relevant given that the build project is likely to be a longish haul.

## **IBM Hursley Museum —** *Peter Short*

Over the last two months we have acquired quite a bit of old IBM hardware from two former IBM offices in Basingstoke and Weybridge. We have also received donations from the Hursley DP Machine Room of equipment made redundant in recent weeks, together with a few private donations.

Hardware from the former offices includes several different iterations of the AS/400, including a rack containing three tape library machines, a 9309 and two 3590s. We also have some RS/6000 and PC / PS2 boxes, printers, miscellaneous cables and adapter cards, a couple of racks and eServer & RS/6000 rack mounted units. We also acquired a Sequent-branded rack containing as yet unidentified servers.

The Hursley DPMR kit is not particularly old, but we figure it will be one day. Included is a Power5+ server, comprising 8 × 2 way 2.2 GHz CPU cards, 16 × 4Gb DIMM, 16 × 73Gb 10K RPM SCSI discs and 8 × 2GB Fibre Channel PCI-X cards. The curators have been kept very busy with this new influx, but have also had time to continue work on bringing old kit back to life. Focus is currently on the AS/400s and building a Token Ring Network. Meanwhile, cataloguing and filing of the paper archive continues apace.

Over the last few months we have been putting together a list of products developed at Hursley. There are still many gaps, but we do now have a reasonably comprehensive list on the website. The hardware list is at *tinyurl.com/hurshw* and the software at *tinyurl.com/hursw*. There is a comprehensive list of IBM products on Wikipedia, which we find useful from time to time.

The collection still has some gaps we would like to fill. For example, we need the handset from an old rotary dial telephone to display on our acoustic coupler. We are also sadly lacking any 3270 displays, either 3277 or 3278, and are also missing an IBM PC Enhanced Graphics Adapter (EGA) and 5154 Advanced Colour Display.

Whilst we don't expect anyone to have an IBM 2984 Lloyds Cashpoint in their attic, we've still not managed to track one down. This was a significant product developed in Hursley for Lloyds, and heralded the beginning of the ATM market as we know it today. If we could only find one, we could perhaps use it in a display covering both hardware and software - especially CICS. Even Lloyds don't seem to have one stashed away. If anyone has any leads that might help us find one, please let us know!

## **Our Computer Heritage —** *Simon Lavington*

Six volunteers are active in compiling information for the post-1970 group of machines, Dave Goodwin (GEC 2050 series, 4000 series & System 63) and Alan Wakefield (ICL System 10, System 25 and DRS/20) being new recruits. In addition Bill Findlay has provided a wealth of information about the Information Computer Systems: ALP1/2/3 range. We still, however, lack a volunteer for Digico Computers Ltd. including Digiac and Digico Micro 16-S, 16-P, 16-V.

With Rod Brown's help, the plan is to create a separate, temporary website for OCH minicomputer development. No material is yet ready to upload, though Lisa Robinson's material for the HLH Orion looks like being the first.

## **ICL 2966 —** *Delwyn Holroyd*

The machine has not been operational for most of the period since the last report. However I am pleased to report it is now working again. Following removal of the OCP platters to clean the unreliable C connectors, a whole new set of problems appeared culminating in the diagnostic software crashing on the SCP without any indication of what the problem may be. Having ruled out SCP and DCU faults through board swapping, we realised that at the time of the crash the software might be having a look down the OCP diagnostic interface to verify that the configuration number provided matched the attached hardware. Accordingly we swapped the OCP diagnostic interface board and the fault cleared. Not only that but all of the other issues that CUTS was previously reporting have also vanished! Evidently these apparent faults were being caused by some issue in the diagnostic system itself. We haven't yet dared touch this board again to verify whether the problem really was down to a board fault, or the more likely scenario of the off-card connections. Since then the machine has run on two whole days without any crashes (a record), but did crash twice on another day.

A working ICL Trimetra DY server (great-grandson of 2966 [ed]) has recently been donated to the museum by Gresham, and put on display in the 2966 area. A licence to run the VME software is being negotiated with Fujitsu.

## **Tony Sale Award**

Entries for the 2014 Tony Sale Award are invited. The award is to recognise achievements in computer conservation or restoration. The award was established in memory of computer conservation pioneer Tony Sale and is sponsored by Google UK.

Martin Campbell-Kelly said: "Computer history may be relatively short, but progress has been incredibly fast and it is vital that this history is not lost. We can gain important insights and perspective by understanding the context and the challenges faced by computing pioneers over the past decades."

*www.sale-award.org* gives full details

## **Bombe Replica —** *John Harper*

The most recent activity to report is the screening of the first episode of the second series of *Bletchley Circle*. This was broadcast by ITV on Monday 6th January 2014 at 9 p.m.

The introduction sequences were mostly filmed in our Bombe Rebuild area but you would be hard to recognise this although the Bombe was seen running and operated along with the Checking Machine and Typex.

The actual filming took part mainly in the early hours of the morning with a great deal of technical support being provided by our team members. It took place during the third week in May. It is such a long time between filming and broadcast that one almost forgot what went on when it was eventually shown.

Filming caused us tremendous disruption and the cancellation of demonstrations for three weeks. In order to give the whole area a completely different overall image the walls were painted a darker colour, the display cabinets that we were not prepared to empty were covered and made to look like cupboards and numerous desks and other props brought into the area. One of the old props Bombes from the original film 'Enigma' was brought into the area and this involved using a very large crane.

This was quite an experience but not one that we would wish to repeat very often.

## **Bloodhound Missile Preservation Group —** *Peter Harry*

The past few weeks have seen further work on the I/O and Display Console in preparation for re installing the Argus 700.

Some email correspondence with the Swiss has taken place regarding addressing used by the Argus 700 GX to discriminate between its simulator and operational role. Details were exchanged on the differences between the classroom trainer and operational versions of the Bloodhound Argus 700 system. (The BMPG has the operational variant). The Swiss have sent instructions and commands to run tests on specific I/O addresses from the Argus 700 with the system monitor (FT81). This information relates to the Argus 700 GX test rig where a single

monitor is being driven, but not successfully to date, by the BMPG. The person responsible for running the Swiss Bloodhound simulator has recently built an Argus 700 GX test rig so notes on this set-up are being exchanged.

## **Elliott 803/903 —** *Terry Froggatt*

Both the Elliott 903 and Elliott 905 have been moved recently to make way for that young upstart, the EDSAC Replica.

The 903 has moved to the other side of the room, against the windows. No longer will we be able to sit cosily behind it, facing the public and chatting with them as they pass by. When I wrote my previous report, only half of the 903's 16K memory was working. On the day I visited TNMoC with a set of spare cards, all 16K was working so I was unable to locate the fault. But since being moved, only 8K is working again.

The 905 has moved to Bilton Road. The separation of the 903 & 905, and the splitting of the Elliott spares holding & workshop facilities between sites, is likely to cause some inconvenience, although it may lead to better facilities for working on the 905 in due course. And volunteers tinkering in Bilton Road can no longer be included in the head-count for opening the Museum.

Before the move, I spent a day surveying the 905. The main desk contains two processors, one on cold standby, so effectively we have a complete set of spare cards. We only have two of the three original memory cabinets. These two contain in total three 16K store modules, each with its own power supply, so by using just one cabinet we could have a 32K system, with the other 16K as spare. Sadly the store cable terminators were in the third cabinet, although do we have the specification and some spare plugs. With so many duplicated units available, it may be that we only need to make some terminators to get the 905 working.

On the software front, Peter Onion suggested some months ago that a program to solve the Channel 4 TV "Countdown" Numbers Game might be implemented on various TNMoC machines, to show how computer speeds have increased over the years. A Pentium can find the answer before the 30-second clock starts. We now have a version which we can demonstrate on the 903. This is not quite fast enough to play the game in real time, but it would be on the 905. We also have versions of Tic-Tac-Toe written in BASIC and in 903 assembly code, to illustrate the speed difference.

The divide instruction on the Elliott 900-series is not well suited to the exact integer arithmetic needed for "Countdown". It gives the nearest odd number to the right answer, and it discards the remainder. When the answer should be

exactly even, which way it rounds depends on the sign of the denominator. Writing "Countdown" prompted me to check how the various issues of the 903 Algol interpreter performed integer division. I discovered that NONE of them has ever given the right result for all legitimate input values.

Finally on the documentation front, Andrew Herbert has kindly arranged for the large (15" x 19") circuit diagrams to be scanned.

## **ICT 1301 —** *Rod Brown*

The 1301 project has formulated a plan to restart work and resume the software capture early in 2014. It only remains to set dates and start the delivery of the key items held in safe keeping to the TNMOC store at Bilton Road to get work on this project restarted in 2014.

A list of newer volunteers will be revisited and individuals contacted during the rest of January to set the new team in place. It is hoped that the closeness of Milton Keynes to the old ICT/ICL locations of Stevenage and Letchworth will work to the new project's advantage and a good response is expected.

## **Manchester Baby (SSEM) Replica —** *Chris Burton*

The replica continues to be routinely operated and demonstrated at MOSI, on four days per week including Sundays. The volunteers use the number of conversations they have with visitors in a session as a measure of public interest. Typically this number is in the range 20 to 40.

We constructed some cathode ray tube test equipment some years ago. It is now being used to develop a method of determining which CRTs are suitable to use as stores and which are not. They are making progress in developing techniques which were not developed at the time of the build of the replica. There is some concern that CRTs are ageing quickly (loss of vacuum?) and that it is becoming harder to keep the main store CRT reliable. The Accumulator (32 bits) and Control (64 bits) CRTs are much more reliable and can normally be left switched-in for demonstrations.

We have addressed and mitigated a significant number of safety issues over the last year, such as emergency shut-down buttons and isolation plastic panels over high voltages. In the same category is the complete implementation of the volunteer grading scheme. It is gratifying that occasional demonstration session reports will say that it was not possible to switch-on the machine due to a shortage of the right grade of volunteer on that day. In such event the demonstrators have recourse to using simulators on a large screen to engage visitors.

## **Software —** *David Holdsworth*

### Leo III Intercode

Shortly before Christmas, we achieved a clean bootstrap. (Remember that the Intercode Translator is written in Intercode.) We now have a source code of the Translator as a plain text file, which is an image of a hypothetical paper tape that would have translated on a real Leo III to produce the binary program of the Intercode Translator. We have an image of a magnetic tape that holds both the source and binary versions, and we can run the binary version to translate the source and generate the same binary version.

Members of our team of volunteer veterans are checking out some of the more esoteric facilities that are not actually invoked by the bootstrap process. This means that it is currently possible to write an Intercode program that translates correctly but generates instructions that our emulator does not handle correctly. We know that there are some Intercode instructions that are never used in the translator itself, so there is always the possibility that we have errors in the parts of the translator that translates these instructions.

But, the big message is that it works. Having now rescued the translator from the printer listing, there remains the job of presenting it to a modern audience in a relevant way, and of ensuring its retention into the indefinite future.

We also hope to be able to preserve the Master Routine in a form that can be executed. As this is written in Intercode, we have now got a vital tool towards achieving this goal.

### Relevance

We see our on-line versions of the manuals as a vital part of making the preservation relevant to today's audience. There have been only very minor corrections of OCR imperfections in recent times, but these on-line manuals are in frequent use by members of our volunteer team.

Our interim translation produced a listing in HTML that has hot links to enable the following of subroutines and other non-local references in the code. A post processor for the real translator listing to achieve similar hot-links is a gleam in the eye.

Our on-line facility for execution of the translator has undergone enhancement, and is now to be found at: *tinyurl.com/simleo.*

Please do not ask too much of this facility by giving it enormous programs. It is only running on a Raspberry Pi.

We also have Windows binary versions of the emulator and sort-of-loader that extracts the relocatable binary program from the magnetic tape image and generates a file holding a memory image for the emulator. These are all that are needed to run the Intercode Translator, and to execute the resulting program. We have also a Windows version of our magnetic tape diagnostic print program. Each of these is written in C.

**Retention**

In the 1960s the Leo III Intercode system held both source code and binary program on magnetic tape. In our emulation environment we have files which are images of these tapes, and these are perhaps the closest material that we have to the originals. Our "paper tape" version of the source is 570610 bytes. Had it existed in real life, it would have been almost a mile long (actually 1585 yds 1 inch, plus any blanks).

**Brooker-Morris Compiler Compiler for Atlas**

The Atlas Emulator continues to develop, and in particular the project to resuscitate the Brooker-Morris Compiler Compiler marches briskly forward. By the end of November, the Compiler Compiler source code was sufficiently clean to be able to write a copy of the compiled program to (simulated) tape albeit in a format which was not (quite) compatible with the Supervisor (operating system). But not before it had solemnly printed the date on the (simulated) lineprinter – "**30 NOV 1913**" — possibly the last millennium bug ever to be discovered.

We have experienced a number of difficulties along the way in that the source code we have is dated December 1963 whereas the emulator tries to reproduce the system as it was in January 1965 when the main documentation was published and much further development had taken place. The Supervisor tape problem mentioned above is just one example. Work was suspended for Christmas, but has been restarted by making some changes to the emulator and the source code (carefully documented, of course) to address the issues.

Meanwhile Bill Purvis has joined the Compiler Compiler project. The Atlas emulator runs under Windows, but Bill is implementing a version under Unix and is making spectacular progress. This is not a port of the Windows version, but a new emulator written in Java, implementing just enough of the Atlas to run the Compiler Compiler.

We currently lack detailed information on the format of the Supervisor Tape and on certain aspects of the unstandardised floating point instructions. If there is anybody out there who can help, please contact the editor.

# News Round-Up

Just before Christmas, our esteemed chair Rachel Burnett received a tentative enquiry from one Jeff Opt of the NCR Corporate Archive in Dayton Ohio. Jeff had heard a rumour that somewhere in the UK there was an NCR 315; a type of machine which has not survived elsewhere. Rachel forwarded the email to the usual suspects and, within an hour or so Kevin Murrell had replied with not only the location of said computer but some photographs. Such is the time zone difference that the question had apparently been answered before it was asked! The ability of the Internet to facilitate this sort of information exchange never ceases to amaze!

101010101

On Christmas Eve the UK Government announced a Royal pardon for Alan Turing. This follows a long campaign to have his 1952 conviction for gross indecency set aside.

Reaction was generally positive. Iain Standen, head of the Bletchley Park Trust opined "Turing was a visionary mathematician and genius whose work contributed enormously both to the outcome of the war and the computer age"

However, Turing's distinguished biographer, Andrew Hodges was less effusive —

"Alan Turing suffered appalling treatment 60 years ago and there has been a very well intended and deeply felt campaign to remedy it in some way. Unfortunately, I cannot feel that such a 'pardon' embodies any good legal principle. If anything, it suggests that a sufficiently valuable individual should be above the law which applies to everyone else. ..... For me, this symbolic action adds nothing.

A more substantial action would be the release of files on Turing's secret work for GCHQ in the cold war. Loss of security clearance, state distrust and surveillance may have been crucial factors in the two years leading up to his death in 1954."

101010101

In *Resurrection 57* we reported that Leonardo DiCaprio was to star in a new film of Turing's life. It now appears that this report may have been premature. Benedict Cumberbatch has now assumed the role which seems more appropriate. CCS member Tony Brooker, once Turing's assistant, reports that he has been approached by Cumberbatch for advice on the voice.

101010101

At the start of 2014 it seems appropriate to record that we are now 50 years on from the launch of the ICT 1900 Series, the IBM 360 and the CDC 6600 (in that order). An eventful year, 1964. And the ICL 2900 was launched 10 years later. 50/40 years on, the IBM 360 and ICL 2900 ranges still soldier on albeit re-named and re-implemented several times each.

101010101


In December, to commemorate the 70th anniversary of the first Colossus code-breaking machine, a bust of Tommy Flowers was unveiled at BT's Adastral Park research facility at Martlesham Heath in Suffolk. Martlesham Heath is the successor to the GPO's Dollis Hill labs where Flowers worked and where Colossus was built.

Fellow inventor Trevor (clockwork radio) Bayliss, the Sage of Twickenham, did the unveiling and is pictured here with the bust.

101010101

## CCS Website Information

The Society has its own website, which is located at *www.computerconservationsociety.org.* It contains news items, details of forthcoming events and also electronic copies of all past issues of *Resurrection*, in both HTML and PDF formats, which can be downloaded for printing. We also have an FTP site at *ftp.cs.man.ac.uk/pub/CCS-Archive*, where there is other material for downloading including simulators for historic machines. Please note that the latter URL is case sensitive.

# Remembering the IBM 1620

*Andrew Colin*

> As part of the Atlas group in the University of London Computer Unit in 1965, I had enjoyed the luxury of working with Atlas, the most advanced computer in the UK. The IBM 1620, provided when I moved to the University of Lancaster later that year, came as something of a shock. IBM had given it the name "CADET" which, for reasons which will become apparent, wags decided stood for "Can't Add, Doesn't Even Try".

This was the period of major university expansion, and I was given the chance to be in charge of computing at the newly founded University of Lancaster. The offer was attractive; not only would my family and I be able to exchange our town flat for a house in the country, but I would have unlimited access to a computer. It's hard, now, to imagine the significance of such a privilege. This was the era before the invention of personal computers. Programming was an immensely frustrating experience; you punched a paper tape or a pack of cards and handed it to "Program Reception". With luck, you might get the result the next day. At Lancaster I could sit at the computer all night if I wanted to, and frequently did.

The long-term plan at Lancaster was to buy an ICL 1909 computer. But in 1965 the campus at Bailrigg was only half-built, and the university was temporarily housed in a warehouse in the city centre. Lectures were given in a nearby church. As a stop-gap, the University had arranged to rent a small cheap computer — an IBM 1620.

**The IBM 1620 computer**

The room prepared for the machine was on the top floor of the warehouse. The lift didn't go that far, and the delivery men spent some twelve hours using ropes and planks to haul the computer (which was solidly built and extremely heavy) up the last flight of stairs. All this time I hovered about like an expectant father.

An IBM 1620

Eventually the machine was fully installed and commissioned. It had 20,000 decimal digits of immediate access storage, a card reader and punch, an exchangeable disc drive of about 4 million

digits, and a 10 characters/second typewriter. The software provided included a FORTRAN 2 compiler and a symbolic assembler.

In its favour, I must record that the computer always worked when I switched it on in the morning (a difference from Atlas!). On the other hand, some of its characteristics, when I had absorbed them, surprised me considerably, as I was then used to working on binary machines with useful, orthogonal order codes.

Each decimal digit of an operand was stored in a 6-bit word. One of these was a parity digit. Another was a special flag used only to mark the last digit in a number, and the remaining four bits held the decimal digits in an 8421 code. One of the other combinations was an "end of record" marker, and five of the 16 possible combinations were never used.

All arithmetic was decimal, with numbers of arbitrary length. There was no arithmetic unit as such. Instead, addition, subtraction and multiplication were done by tables held in the memory. Printable characters were coded as pairs of digits, with all the numerous undefined combinations appearing as the Russian letter Ж.

The machine had no accessible registers as such. All data handling instructions used a two-address format, memory to memory. There was a 'rich' order code, so that almost everything that could be done could be done in several ways. You could simply ignore most of the instructions and just use a few of them.

The computer had a basic cycle time of 20 microseconds. Arithmetic times for fixed point numbers of five digits, and floating point numbers with two-digit exponents and eight-digit mantissae were:

| OPERATION | TIME (MICROSECONDS) |
|---|---|
| Fixed point addition and subtraction | 560 |
| Fixed point multiplication | 5,000 |
| Fixed point division | 16,900 |
| Floating point addition and subtraction | 1,200 |
| Floating point multiplication | 12,500 |
| Floating point division | 41,700 |

Hierarchy is a key concept in computer science, but one which was evidently unfamiliar to the designers of the IBM 1620. Thus the machine had no concept of a stack, in any form. It had one dedicated "return address" register, which could only be loaded by a "call subroutine" instruction and read by a "return" instruction. This made it awkward to write code with multiple nested subroutine calls.

The "end of record" character was always recognised as such, and would terminate transfers. This made it impossible to analyse arbitrary data, or even to copy sets of nested records. E.W.Dijkstra wrote *A review of the IBM 1620 data processing system* at the time the machine was widely used. He identified several unfortunate aspects of the design, and says,

> "As the reader will understand, my recent study of the IBM 1620 has been a shocking experience. I knew it was rather a small machine but I had never suspected that it would embody so many basic blunders."

**An IBM 1620 at Stanford with grown-up peripherals**

My task at Lancaster was a dual one; I had to supply a computing service to the nascent Physics and Chemistry departments, and also teach something about computing to Mathematics undergraduates.

The first part of the job was undemanding; most users were happy with the FORTRAN compiler and seemed to accept the glacial speed of the machine without question (that's all they knew, the poor innocents!). The second part was more challenging. I felt that as part of the Computer Science course I should inform the students about the architecture they would meet on a more conventional machine.

In my days at the University of London I had already devised a virtual computer called the IMDAC. It had 4096 words of 16 binary digits, and two accumulators which also served as index registers. I had written a simulator on the Atlas, as well as extensive user manuals. I now had the problem of making the IBM 1620 behave like a conventional computer. Here's what I did:

I decided to store one bit per word, and to use a number length of 16 digits. I adjusted the arithmetic tables to do binary arithmetic. This made the 1620 into a highly inefficient binary computer. This did not leave enough space in the memory for the IMDAC store, so I used a virtual storage system, built entirely in software, but based on the Atlas virtual memory. The result was painfully slow,

but entirely adequate for students to run simple sort programs and prime number tabulations — which was as far as my course went.

At this time direct interaction with the computer was a hot topic and I now had the opportunity, for the first time, to experiment with user/machine dialogues. I wrote a simple program that let the user enter a "flat" data set from cards, and then ask for any of several statistics with an immediate response. The most sophisticated output was a crude scatter diagram drawn by typing X's at roughly the right place on the page.

When I invited potential users to try the program, initial results were discouraging. The users were so overawed at sitting in front of a real computer, so terrified of damaging it, that they were overcome by paralysis and seemed unable to understand the simple instructions on the page before them.

I improved matters somewhat by offering a gentler introduction. Users could start by typing a single letter, and so they did (with a little encouragement from me):

```
Computer:   Please press any key on the typewriter. Do it now.
User:       (after some hesitation)  J
Computer:   Well done! You have typed a J. Now type your first
            name, followed by the ENTER key
User:       MARMADUKE
Computer:   Hello, Marmaduke. Your data set has 243 records,
            with 7 numbers in each.
            What statistic would you like?
```

And we were off ….

Eventually the building at Bailrigg was completed, and the ICL 1909 installed. It was time to say goodbye to the 1620. Our IBM engineer was very conscientious. He had even replaced the entire disc unit when the rental had only six weeks to run. On his last visit he told me that when we gave up the computer it would be scrapped. I asked him whether IBM would consider donating it to the University instead. "No" he replied. "It will be broken up and the circuit boards sent to the local jail, where the prisoners will occupy themselves in scraping the two micron gold plating off the edge connectors."

*Andrew Colin has held senior computing positions at the Universities of London, Lancaster and Strathclyde. He can be contacted at* Andrew@crm.scotnet.co.uk.

# 100 Years of IBM (part 2)

*Terry Muldoon*

Part 2 of the article started in *Resurrection 64*.

**The 1980s**

The Reduced Instruction Set Computer

The first Reduced Instruction Set Computer (RISC) was the IBM 801 which was begun in 1975 by John Cooke at the IBM Thomas J. Watson Research Centre and was completed in 1980. The 801 was named after building 801 on the site.

The 801 was productised as the ROMP (Research [Division] Office Products [Division] Micro Processor). Originally designed for use in the follow-on product to Office System/6 (which became the Intel-powered Displaywriter System). ROMP was actually used in the IBM RT PC (IBM 6150 in the UK).

The 801 also lead to the highly successful POWER (Performance Optimization with Enhanced RISC) architecture which is used in IBM RISC System/6000 systems and most of the World's Gaming Consoles.

The IBM Personal Computer

In August 1981 IBM announced the IBM Personal Computer — and the rest was history.

IBM United Kingdom's plant in Spango Valley, Greenock, Scotland was chosen to manufacture the PC for EMEA (Europe, Middle East, Africa).

PC/IX

In 1984, IBM announced its first UNIX based Operating System: Personal Computer Interactive Executive (PC/IX). It came on about twenty discettes and ran on an IBM Personal Computer XT (PC XT).

Personal System/2 (PS/2)

The IBM Personal System/2 family of personal computers was announced on April 2nd 1987. Other than the Model 30, they were all based on a new bus architecture called Micro Channel Architecture. Micro Channel Architecture was a 32-bit (with a 16-bit subset) bus architecture that was designed to overcome the restrictions of the IBM PC/PC XT/PC AT buses. It was a very advanced personal computer architecture based on the expertise that IBM had accumulated over many years of designing mini-computers.

The new PS/2s had new built-in mouse ports and video adapters. The video adapters, the built-in Video Graphics Array (VGA) and 8514/a & Image Adapter/a adapter cards were all designed at IBM Hursley as were the monitors.

Micro Channel Architecture

No one at the time or since has criticised the technical quality of Micro Channel Architecture. In fact, after the announcement, some of IBM's competitors changed their views as to what was actually "wrong" with Micro Channel over a period of days, weeks, and months.

One of the major criticisms was that Micro Channel was not compatible with the IBM PC/PC XT/PC AT design. At the time, IBM failed to counter that with the fact that Micro Channel was designed NOT to be compatible in order to correct some of the deficiencies of the old machines. Additionally, other than some special adapter cards, most personal computers were coming with features built-in to the motherboard that previously had been on adapter cards.

One of the accusations levelled at IBM at the time was that, unlike the preceding PC/PC XT/PC AT designs, Micro Channel was not "Open Architecture". In fact the Micro Channel Technical Reference Manuals contained MORE detail on the new bus than the previous IBM Personal Computer documentation had. Experts at the time gave IBM eighteen months before it was competing with "PS/2 Clones", which proved to be pretty close to what happened.

Another criticism was that manufacturers would have to pay royalties to IBM for Micro Channel Architecture, ignoring the fact that royalties were payable to IBM on the old PC/PC XT/PC AT design. This comment seemed to be particularly from those manufacturers who, at the time, had not got around to paying IBM the royalties that they currently owed on PC/PC XT/PC AT design.

In the long term, Micro Channel Architecture failed in the marketplace and it failed because of a lack of marketing skills in the IBM Corporation rather than for

any technical reasons. The highly successful Peripheral Component Interconnect (PCI) bus of 1992 is remarkably similar to Micro Channel.

Operating System/2

Operating System/2 (OS/2) announced in 1987, as an eventual replacement for Personal Computer DOS, was a joint development with Microsoft and was intended to make better use of the more powerful processors that were capable of multi-tasking being developed by the Intel Corporation at that time — such as the i386.

IBM Hursley was the lead for OS/2 graphics, based on its previous work on mainframe-based Graphical Data Display Manager.

OS/2 was a very advanced Operating System but took a while to ship. This was partly due to differences in culture between IBM and Microsoft. Additionally, IBM and Microsoft had different views about some of the technical aspects of the product.

So IBM and Microsoft parted company with IBM taking over the development of OS/2 completely. Microsoft continued to market MS DOS/Windows while at the same time taking over the development of OS/2 Version 3 – which was renamed Windows NT.

AS/400

The AS/400 (the AS standing for Application System) was announced by IBM in July 1988. Its architecture was based on the IBM System/38 (announced 1979) which itself used many of the features of the cancelled Future System project.

Like System/38 before it AS/400 quickly acquired a deeply loyal fan base amongst its customers. In 2000, the AS/400 was renamed eServer iSeries. In 2006 it was again renamed as System i. (Among its proponents, it is often still called AS/400 however.)

**The 1990s**

RISC System/6000 and POWER Architecture

Following on from the IBM 801 and ROMP, with the announcement of the RISC System/6000 in 1990, IBM announced a new RISC based architecture: POWER (Performance Optimization with Enhanced RISC) POWER came in both multi-chip and a single chip versions. The latter, called the RISC Single-Chip (RSC) was the basis of the later POWER-PC used in many products including Apple Mackintosh.

POWER has gone through many iterations but is still the architecture used in the IBM Power Systems, IBM Supercomputers, and most of the world's games consoles.

The Operating system for the RISC System/6000 was a new version of the UNIX based IBM Advanced Interactive eXecutive (AIX) — Version 3.

Magneto Resistive Hard Disc Heads

Always in the vanguard of disc technology, IBM introduced a three and half inch, one gigabyte disc ("Corsair") in 1991 using Magneto Restive (MR) heads.

In 1993, IBM won many awards for the first three and half inch, one inch high hard disc code named "Spitfire".

At this time, IBM was becoming highly competitive in the OEM (Original Equipment Manufacturers) marketplace. Previously IBM had made all its discs for internal consumption. IBM United Kingdom was manufacturing a large percentage of these products for customers in EMEA at its plant in Havant, Hampshire, England.

ThinkPad

In 1992, IBM announced the highly successful ThinkPad brand of mobile computers. The product family was named after a small flip-open notebook that many IBMers carried in their pockets. The small pad was printed with "IBM" on one side and "Think" on the other. They were universally known in IBM as "Think Pads".

Gerstner

In 1993, following some of the largest corporate losses in history, IBM appointed Louis V Gerstner Jr. as CEO. Uniquely (other than Thomas J Watson Snr. in 1914), Gerstner did not come from within but was an outside hire. Gerstner is widely credited for "saving" IBM.

Lotus

As part of the transformation of IBM from a hardware-led company, in 1995 IBM bought Lotus Development Corporation. Adding the Lotus portfolio to the in-house products such as DB2, CICS, IBM became a major player in software.

Tivoli

In 1996 IBM acquired Tivoli Systems Inc. in order to strengthen its portfolio in systems management software. IBM is now the market leader in systems management.

Deep Blue

Deep Blue was a chess playing computer. The hardware was a massively parallel IBM RS/6000 SP.

The name Deep Blue was a play on the fictional Deep Thought computer in Douglas Adams' *The Hitch-hikers Guide to the Galaxy* and IBM's nick name – "Big Blue".

Two PhD students at Carnegie Mellon University, Murray Campbell and Feng-hsiung Hsu, were hired by IBM Research and given the resources to build Deep Blue.

In 1997, Deep Blue became the first computer to defeat a reigning world chess champion – Gary Kasparov.

GMR Disc Heads

Following on from introducing Magneto Restive (MR) heads in 1991, IBM introduced heads using the Giant Magneto Resistive (GMR) effect on the DeskStar product line in 1998. GMR initially allowed a capacity of ten gigabytes on a three and a half inch hard disc. Later, even greater density was achieved.

Although the first company to make a magnetic disc and a pioneer of the technology, with "Winchester", thin film heads, Magneto Resistive heads and Giant Magneto Resistive heads, IBM merged its disc business with Hitachi in 2003, although the company carried on making storage subsystems.

e-business

When Lou Gerstner joined IBM in 1993, he was asked by a journalist for his vision for IBM. He replied: "The last thing that IBM needs right now is a vision."

By 1995, IBM and Gerstner were starting to see light at the end of the tunnel and having "saved" the company, Gerstner could concentrate on where IBM was going next. (Some might call this a "vision".)

Gerstner saw that IBM was the company that could help its customers realise the huge potential of the Internet by making their businesses "Network Centric".

"E-business" was one of IBM's most successful announcement and it wasn't even a product.

**Recently**

PwC

In October 2002 IBM bought the entire consultancy business of PricewaterhouseCoopers (PwC). The acquisition almost doubled the number of consultants in IBM Global Business Services. IBM Global Business Services is the largest IT consultancy business in the world.

Rational

In December 2002 IBM bought Rational Software. The Rational acquisition strengthened IBM's portfolio of software development products.

PC Division Sold

In 2005 IBM sold the PC Division, including the iconic ThinkPad brand, to Lenovo of China.

Watson

In 2011 an IBM Supercomputer based on IBM POWER 750 servers called "Watson" won the US television game show "Jeopardy". The computer, named after Thomas J Watson is capable of understanding questions posed in natural language. The technology developed for Watson is now available commercially.

**IBM Today**

The IBM of today is not the IBM of the past. In fact the IBM of Thomas J Watson Jr. was not the IBM of Thomas J. Watson Snr. Companies, like organic life forms are said to have a finite life — however long or short that life may be. IBM is one of those unique companies that appear to buck that theory by constantly reinventing itself. Something that it continues to do to the present day.

*Terry Muldoon started with IBM as a Customer Engineer in 1970. He is now a volunteer curator of the IBM Hursley Museum. His email address is* terry_muldoon@btinternet.com

## North West Group contact details

Chairman Tom Hinchliffe: Tel: 01663 765040.
Email: *tah25@btinternet.com*
Secretary Gordon Adshead Tel: 01625 549770.
Email: *gordon@adshead.com*

# Building a Soft Atlas

*Dik Leatherdale*

> It all began in 2002: on the day of Tom Kilburn's memorial service. In the morning I took the opportunity to visit MOSI and, in particular the replica Manchester Baby. Happily there was somebody (and I know not who) on hand demonstrating — that seemed appropriate. I'd not seen the Baby before and I was, to say the least, deeply impressed. That such a thing should be thought possible using 50 year-old technology was amazing enough. That the effort had been successful seemed little short of miraculous. We chewed the fat for a while before I ventured that a replica Atlas sounded like a project worth doing. A cloud came over our host. "That might be a bit beyond us." "A soft Atlas then." I ventured. And somewhere deep inside my head a light bulb came on.

I filed the idea away; a retirement project, perhaps. Well, it didn't turn out quite like that. A couple of years later I found myself "between engagements". You can spend a fair amount of time applying for jobs in the space of a day, but there's a limit and I loathe being bored. In previous "fallow periods" I'd taught myself C, C++ and Java on my Windows PC. I'd heard good things about C# so I put the two ideas together and made a start. C#, it turned out, was a good choice. I'd not read David Holdsworth's paper on using a C subset (C--) for emulation and if I had I might have gone that way but that would have been harder I suspect.

My initial source material (beside my own unreliable recollection) was the manual for the main Atlas assembly language — Atlas Basic Language (ABL) — a document so detailed it left very little to guesswork. Later the ABL manual would be supplemented by the so-called *Atlas Bible*: a collection of working papers which circulated amongst the members of the Atlas development team before there was any formal documentation.

I set myself a limited but achievable objective (or so I thought) of being able to emulate a single process, rather than the complete machine. I wanted to run Atlas programs. I was dimly aware that Chris Burton had a supervisor tape containing many of the Atlas compilers and a 1" tape deck. To date Chris has been far too busy doing much more important projects to progress matters.

After much thought I realised that an emulation of the store was the place to make a start. Atlas was the very first paged machine which made it more interesting. The fundamental building block was a C# **`int`** (32 bits) onto which I mapped a 24-bit Atlas halfword. An array of two halfwords made up a word, an array of 512 words to a block (nowadays called a page) and then a list (rather

than an array) of store blocks comprising the complete virtual store of a process. So far so good, but I had only limited ability to test what I'd done. The host program had a temporary test harness which attempted to write and read words and halfwords in specific locations. That seemed fine. I put a display of the store block numbers which were in use on the host window and that too worked well. Then I allowed the user (me) to click a block number which would bring up a display of the contents of the block in octal.

All very well, but testing was laborious to say the least. So I started to implement a crude compiler which would read source code from a text file. Only two statement types — a pair of eight digit octal numbers defined the content of the next two halfwords and * followed by an eight digit octal number to say "start from this address". What I didn't realise at the time was that this was almost exactly the capability of the very first Atlas compiler — which formed the bottom layer of the bootstrap for the famous Brooker-Morris "Compiler Compiler".

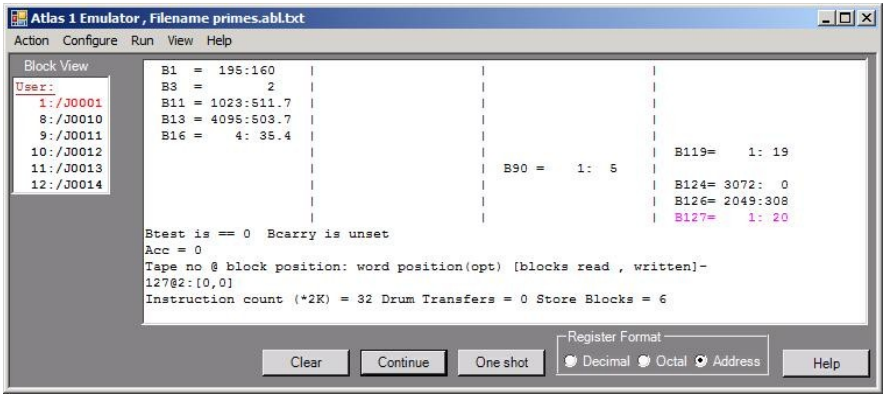
I enhanced the store display to allow the user to view the store contents in decimal, octal, character and instruction formats in any combination. Then I tried writing an Atlas program in octal. This turned out to be harder than I anticipated. Very error-prone. What I needed was an assembler. Why invent a new assembler when I had the ABL manual to hand? Though ABL looks simple enough at first glance, it wasn't the first Atlas assembler and it had lots of fairly sophisticated facilities (for its day) the result of early experience. But if there's one thing I know about it's writing compilers. Rather than attempting to write an ABL compiler in octal I took the cowards' way out and wrote it in C# within the emulator host. Even then it took so long that by the time it was almost finished I was back at work.

Work which, it turned out, was rather undemanding, but I did briefly get to use my newly acquired skills in C# – perhaps the only useful work I did in three years. The job did however, afford the opportunity to spend a bit more time working on the Atlas project, and it was during this period that I set to work on the next strand of the exercise, that of the emulator itself.

The problem with writing an emulator for a machine like Atlas is that, even in the original machine, there wasn't a lot to see. To be sure paper tapes and cards went in at one end, magnetic tapes whirled, and lines were printed at the other, but to the casual observer, none of this bore much relationship to the work going on and, in any event, the emulator wouldn't have proper peripherals so there would be even less to see. So I hatched the idea of basing the emulator on Microsoft's Visual Studio debugger. The notion of allowing the user to halt

execution of the program and then proceed one instruction at a time would allow the observer to see the detail of how the machine functioned at an order code level. And it would allow me to debug not only the ABL programs but the emulator itself with rather more ease than in the real machine. So the ability to set a "break point" at which the program would halt awaiting further instructions came next.

The next step was to implement the 128 index registers and a floating point accumulator and build in the capability of displaying them in the host window. Then an execution engine which would analyse an instruction into its component parts and then access and modify the store and registers as necessary. Finally the beginnings of a proper emulator.




**Displaying the register values etc. in a running program.**
**The contents of each block (page) of store can also be displayed.**
**Formats can be varied but conform to ABL conventions.**

Now Atlas had no less than 100 hardware instructions, so this, together with the necessary test programs took some time. Indeed, I've only recently implemented the last of them and I'm not completely satisfied that I have some of the more obscure floating point operations correct. One day, one day!

If that was all it took then it was achievable. But there were another 262 extracodes — pseudo-instructions implemented as subroutines in the operating system — anything from input/output operations to arithmetic functions like square root. In the original machine the supporting code was located in the "fixed store" — what we would now call read-only memory. Some of these instructions were rather important. "**`1117 — Stop`**" was an obvious necessity. But as I worked through the first few extracodes I noticed that a few of them would have been quite simple to implement as genuine extracodes. "**`1362 — Subroutine Call`**" was an obvious candidate. Just two hardware instructions

which cried out to be implemented properly. So it was that I set about implementing a fixed store to support a few of the extracodes for the purposes of demonstrating the mechanism. Not difficult because I just had to make a special case of the store components I already had. But how to set up the content of the fixed store? Easy! I already had an ABL compiler, so all I had to do was compile it from source code, develop a method for saving it and restoring it each time the emulator ran.

Having achieved that I set about implementing more extracodes in the fixed store and replacing some of those I'd already implemented as hardware instructions. At which point I had a stroke of luck because I discovered a decompiled listing of the fixed store in the John Rylands Library at Manchester University. They also had a listing of the ABL run-time library and the ABL compiler itself. The Library generously supplied a photocopy of the first two. The ABL compiler I decided was beyond me and besides, I already had one, hadn't I?

So I started to implement the extracodes one or two at a time a process which, several years later is still ongoing. At the time of writing I have 150 extracodes "properly" implemented in the fixed store, 70 extracodes implemented directly as if they were hardware instructions — mainly concerned with I/O and deep operating system functions such as store management. Just 42 still to go — many of which such as the multithreading and ½" tape subsystems will probably never be attempted.

It was while I was working my way through the extracodes that I developed a huge respect for the original implementers. Nowadays we prize readability and maintainability above mere efficiency. But 50 years ago, things were different. I found some deeply obscure methods aimed at saving a few words or store here and a few words there. Unhygienic to be sure, but impressively ingenious.

In 2008 I finally retired from the world of paid work. At last surely, I could devote myself to this seemingly endless project. Wrong! Foolishly perhaps, I took on the job of editing *Resurrection*. Progress slowed to a snail's pace.

My first attempt at input/output was to implement output functions by writing the characters (after translation from Atlas internal code to ASCII) to a window so that my test programs could display their results instead of having to examine the results in store using the debugger. Later I built on this mechanism to allow the option of writing to a text file and implemented the opposite capability of reading <u>from</u> a text file.

But a moment's thought will suggest that although this maps well onto punched cards, paper tape and lineprinter I/O is less well suited. For example, program

text such as "**begin**" is difficult to convey in a text file. My first attempt was to use **** (backslash) to represent backspace as in "**begin_____**" to mean "**begin**" but this is not only tedious but very error prone. I conceived the idea of using MS Word to prepare simulated paper tapes. That would allow me to use underlining to represent the most common forms of compound characters such as **integer array** which might be found in programming languages. Word's "strikethrough" facility would also emulate paper tape erases without difficulty. Other, rarer cases of character compounding can often be emulated by using the rich repertoire of Unicode characters available to the Word user such as "**≠**".

MS-Word's internal format is frightening in its complexity as well as being a tad unstable, so I bought a book on Rich Text Format (RTF). Output was simple enough. I had to implement a form of line reconstruction in the emulator but once that was done generating simple RTF was easy enough. Soon I could read the output with Word or Wordpad!

Input was more challenging. Both Word and, to a lesser extent, Wordpad generate fearsomely complex RTF for the simplest documents — much of which can be safely ignored for our purposes. I found an API for RTF — undocumented and commented in Italian. But it was very similar to an XML parser I'd used for my last unfortunate employer so after a lot of trial and error I had a full I/O subsystem.

Atlas 1" magnetic tapes were mapped onto binary-mode files 3072 bytes to a 512 Atlas word block. The files are unreadable for any practical purposes except by the emulator. But that is as it should be. Atlas tapes were unreadable on any other machine.

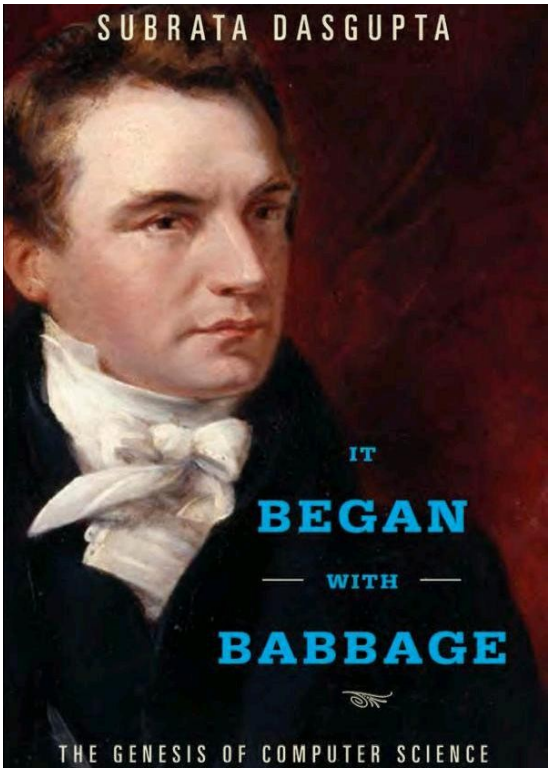
In December 2012 I had the pleasure of demonstrating the emulator at the Atlas @ 50 event in Manchester. I was not alone! Professor Roland Ibbett also has an "Atlas Simulator". His is a rather different beast animating the information path through the computer from component to component as each instruction is obeyed. Fascinating to watch but not at all the same thing.

Most recently, with the emulator incomplete but substantially working, I've been blown off course again. Iain MacCallum and I are trying to resurrect (if I may use the word here) the Brooker-Morris Compiler Compiler from old source listings. Iain was a member of the original Compiler Compiler implementation team. But that'll be another story.

*Dik Leatherdale is an obsessive programmer who when he's not wasting his time on nonsense like this edits this very publication. He has only himself to blame. He can be contacted at dik@leatherdale.net.*

# Book Review: It Began with Babbage

*Simon Lavington*




This book is a fascinating reflection on a new academic discipline. It is an intellectual and cultural story woven round the history of automatic computation from 1819 to 1969. Centre-stage is the stored-program paradigm, which emerged between 1945 and 1949. Before this, the story takes in Babbage, Leonardo Torres y Quevedo, Percy Ludgate and several other thoughtful people who occupy what has been unfairly described as the 'dark ages' of computing history. After 1949 the author Subrata Dasgupta, resists the temptation to use computer manufacturers and their products as milestones. Instead he concentrates on concepts and what he terms the sub-paradigms of Computer Science. He explains that, though the theoretical abstractions created by Alan Turing in 1936 may have shaped the paradigm, Computer Science did not come into its own until after Turing's death, with the gradual accretion of sub-paradigms such as finite automata, systems architecture, declarative programming and artificial intelligence.

We are introduced to the ideas of luminaries such as Knuth, McCarthy and Dijkstra. Closer to home, the spotlight is trained on Maurice Wilkes. Some readers may feel that the book's emphasis on Cambridge has tended to push other UK players into the shadows. Nevertheless, overall this book takes an even-handed view of developments on both sides of the Atlantic. It is thus a valuable counter-point to shallower histories of the subject. Read it for its philosophical reflections as well as for its history of ideas.

# Obituary : Chris Hipwell

*Nicholas Enticknap*

Chris Hipwell will forever be remembered as the man who conceived and drove through the creation of *Computer Weekly*, the first IT weekly publication anywhere in the world. *CW* rapidly established itself as both the main source of information about industry developments and as the best forum for computer professionals to find jobs, and remained the heartbeat of the computer community in the UK from its launch in 1966 till the rise of the Internet in the late 1990s. Chris was deeply involved with the publication throughout this period as founder, editor, publisher, publishing director, editorial director and finally as consultant editor.

Chris has himself recounted how the idea for *Computer Weekly* germinated in a talk he gave to the Society in 2005 (subsequently reprinted in *Resurrection 39*). In 1965 he was editor of one of only two UK computer publications of the day, *Data Processing*. He was being overwhelmed with more news than he could cope with, and by the time he was able to get it published it was long out of date. He saw the need for a publication which came out rather more often than Data Processing's once every two months, and thus *CW* was born.

Chris was in his element during this period as the job called on all his many talents. He had the essentials — an eye for a story, a wide range of industry contacts, and an in-depth knowledge of the IT business developed over his eight previous years in journalism. He also had the necessary personal skills — the ability to select quality staff (many of his recruits went on to distinguished careers in IT journalism and PR) and to provide them with appropriate guidance while allowing their own news-gathering and story-telling instincts free rein. He also had the ability to communicate his own and *CW*'s needs effectively to the management he served.

Chris was an enthusiastic supporter of the CCS from its foundation. He joined the Committee in 1991, and three years later accepted responsibility for organising the Society's London meetings, which task he performed with characteristic efficiency and good humour till 1996. He continued to attend our meetings regularly thereafter. Chris Hipwell was a charming and singularly wise individual whose opinion on anything was always worth hearing, and who was quite astonishingly easy to work with — the writer cannot recall ever having a cross word with him over four decades as an employee and colleague. Chris will be greatly missed.

# 40 Years Ago ….
# From the Pages of *Computer Weekly*

*Brian Aldous – TNMoC Archivist*

**A new microprocessor**, said to be the first using Complementary Metal Oxide Silicon (CMOS), has been developed by RCA at its New Jersey Solid State Technology Centre. (CW 383 7 March p18)

**The prestige contract to supply the hardware** for the SWIFT international banking network has been won by Burroughs. Worth £2,600,000, it involves two duplex B3700 mainframes, four DC140 communications processors and 14 B700 series data concentrators. (CW 384 14 March p1)

**New dual processor time sharing systems** in the 2000F series have been announced by Hewlett Packard, who have significantly reduced the cost of the systems by the provision of moving head disc drives instead of the earlier fixed-head units. (CW 384 14 March p20)

**NPL to work on EPSS interfacing protocol**: Interfacing methods to link users' computers and terminals to packet-switching exchanges will be one of the principal areas to be studied under a contract placed by the Telecommunications Division of the Post Office with the National Physical Laboratory. (CW 385 21 March p1)

**Coral 66 compiler contract**: Following the government announcement of an increased effort to standardise and promote the Coral 66 language the National Computing Centre has retained SDL of Frimley, Surrey, to design a 'portable' Coral 66 compiler. (CW 385 21 March p8)

**DEC launches first microprocessor**: The question which the entire minicomputer industry has been asking, "When will DEC go into microprocessors?" has been answered by the launch in the US of an 8-bit LSI microprocessor based on the Intel 8008 microprocessor chip. (CW 386 28 March p1)

**Bar code readers for goods routeing**: Automatic routeing of goods in warehouses and factories are the main applications for a range of laser beam-based bar code reading equipment now available in the UK from Techex of Poole, Dorset. (CW 387 4 April p3)

**A Floppy disc drive system** is being tested by the McBee division of Litton Industries in Canada as a potential main memory for the 1600 series business computer. A spokesman for Litton said the tests had been 'reasonably encouraging', but added it was much too early to make any definitive statement. It is hoped to use it to replace paper tape and magnetic tape cassettes. (CW 388 11 April p21)

**Babbage gives high level boost to 4080**: The tendency among mainframe manufacturers to write systems software for their newer machines in high-level languages has now been extended to the minicomputer area, with the advent of the Babbage language for the GEC 4080. Conceived as the lowest level of coding for the 4080, it is termed a 'high-level assembler'. (CW 389 18 April p2)

**Motorola has joined the battle** for the increasingly competitive microcomputer market with a range of integrated circuit systems components called MC6800. The five chips in the series are an 8-bit CPU, an 8K ROM, a 1K RAM, a peripheral interface adaptor and an asynchronous communications interface adaptor. (CW 389 Int. Supplement 18 April p11)

**CDC has announced a double-density IBM 3330 replacement disc storage system**. Called the CDC 33302, the multiple disc system attaches to an IBM mainframe's block-multiplexer channel on any of the 370 series or larger units of the 360 series. (CW 390 25 April p12)

**NCR wins John Lewis PoS contract** with an order for a 100-terminal system, giving a considerable boost to NCR's standing in the point-of-sale market. Commenting on the order, NCR chairman Gordon Shingleton told CW that it represented a major breakthrough for NCR. (CW 391 2 May p1)

**ICL announces that it is considering licensing software**, a totally novel method of discouraging independent competition in the area of 2903 software. The plan, involving the issuing of 'licences' to market such software, possibly throws some light on the confused question of New Range policy in this field. (CW 391 2 May p1)

**Leading microprocessor manufacturer Intel** has announced a new product, the 8080, which is compatible with the established 8008 but is about10 times faster. The new microprocessor is a fully parallel 8-bit unit with an instruction cycle time of two microseconds and a repertoire of 78 instructions. (CW 391 2 May p7)

**RTL/2 launched to challenge Coral 66**: Seen as a possible challenge to the government's efforts to establish Coral 66 as the standard language for real time processing, the newly released language is regarded by many sources as being more explicitly oriented to real time operation than Coral. (CW 391 2 May p8)

**A programmable, emitter-coupled read-only memory** called the 10139 has been developed by Signetics Corporation for use in high-speed data communications equipment, large mainframe computers, minis, add-on memories for computers and in high-speed oscilloscopes. (CW 391 2 May p12)

**Plans for test runs of TV data transmission** using a unified national system of data transmission for domestic television receivers have now been agreed between the BBC and the IBA and experimental transmissions are planned for the autumn. The system incorporates features of both the BBC's Ceefax and the IBA's Oracle. The principle of both systems is to make use of suppressed lines in the video transmission to carry data which can be displayed on the television screen by means of a special adaptor. (CW 392 9 May p23)

**GEC equipment for Underground project**: A £700,000 project, financed by the government and the Greater London Council, to improve services on the Northern and Victoria lines of London's Underground railway system has been announced by London Transport. (CW 393 16 May p3)

**Plastics firm first to install 2903**: After a period of difficulties with component supplies ICL has started deliveries of the 2903 small business system announced at the Hanover Fair last year. It was there that Thermo Plastics of Dunstable, placed the first order, and the company has now become the first to install a 2903. (CW 394 23 May p46)

**NCR cash terminals on trial at Barclays**: The prospect of round-the-clock banking facilities for Barclays Bank customers looms nearer with the announcement of the NCR 770 Self Service Financial Terminal. (CW 394 23 May p46)

**Success of ICL 2903 rolls on**: Orders continue to flow in for what must be one of ICL's most successful machines, the 2903. Since its launch at Hanover Fair just over a year ago, more than 500 systems have been sold world-wide and more European orders have now been announced. With a money spinner like this in the product range and the continued demand for 1900 systems, the delayed announcement of the New Range falls into perspective. (CW 395 30 May p1)

# Forthcoming Events

## London Seminar Programme

| | | |
|---|---|---|
| 13th Mar 2014 | Structured Software and the Break with Electrical Engineers | David Grier |
| 10th Apr 2014 | The History of Computing in Colour | Martin Campbell-Kelly |
| 15th May 2014 | Computing Before Computers — From Counting Board to Slide Rule | David Elgin |

London meetings normally take place in the Fellows' Library of the Science Museum, starting at 14:30. The entrance is in Exhibition Road, next to the exit from the tunnel from South Kensington Station, on the left as you come up the steps. For queries about London meetings please contact Roger Johnson at *r.johnson@bcs.org.uk*, or by post to Roger at 9 Chipstead Park Close, Sevenoaks, TN13 2SJ.

## Manchester Seminar Programme

| | | |
|---|---|---|
| 18th Mar 2014 | Babbage's Analytical Engine | Doron Swade |

North West Group meetings take place in the Conference Centre at MOSI — the Museum of Science and Industry in Manchester — usually starting at 17:30; tea is served from 17:00. For queries about Manchester meetings please contact Gordon Adshead at *gordon@adshead.com.*

Details are subject to change. Members wishing to attend any meeting are advised to check the events page on the Society website at *www.computerconservationsociety.org/lecture.htm*. Details are also published at in the events calendar at *www.bcs.org.*

## Museums

**MOSI :** Demonstrations of the replica Small-Scale Experimental Machine at the Museum of Science and Industry in Manchester are run each Tuesday between 12:00 and 14:00. Admission is free. See *www.mosi.org.uk* for more details.

**Bletchley Park :** daily. Exhibition of wartime code-breaking equipment and procedures, including the replica Bombe, plus tours of the wartime buildings. Go to *www.bletchleypark.org.uk* to check details of times, admission charges and special events.

**The National Museum of Computing : Thursday, Saturday and Sunday from 13:00.** Situated within Bletchley Park, the Museum covers the development of computing from the wartime Tunny machine and replica Colossus computer to the present day and from ICL mainframes to hand-held computers. Note that there is a separate admission charge to TNMoC which is either standalone or can be combined with the charge for Bletchley Park. See *www.tnmoc.org* for more details.

**Science Museum :**

There is an excellent display of computing and mathematics machines on the second floor. Other galleries include displays of ICT card-sorters and Cray supercomputers. Highlights include Pilot ACE, arguably the world's oldest surviving computer. Admission is free. See *www.sciencemuseum.org.uk* for more details.

**Other Museums :** At *www.computerconservationsociety.org/museums.htm* can be found brief descriptions of various UK computing museums which may be of interest to members.

## Contact details

Readers wishing to contact the Editor may do so by email to dik@leatherdale.net, or by post to 124 Stanley Road, Teddington, TW11 8TX. Queries about all other CCS matters should be addressed to the Secretary, Kevin Murrell, at kevin.murrell@tnmoc.org, or by post to 25 Comet Close, Ash Vale, Aldershot, Hants GU12 5SG.

## Committee of the Society

*Chair:* **Rachel Burnett FBCS**: *rb@burnett.uk.net*
*Secretary:* **Kevin Murrell MBCS**: *kevin.murrell@tnmoc.org*
*Treasurer:* **Dan Hayton MBCS**: *daniel@newcomen.demon.co.uk*
*Chairman, North West Group:* **Tom Hinchliffe:** *tah25@btinternet.com*
*Secretary, North West Group:* **Gordon Adshead MBCS:** *gordon@adshead.com*
*Resurrection Editor:* **Dik Leatherdale MBCS**: *dik@leatherdale.net*
*Website Editor:* **Dik Leatherdale MBCS**: *dik@leatherdale.net*
*Meetings Secretary:* **Dr Roger Johnson FBCS**: *r.johnson@bcs.org.uk*
*Digital Archivist:* **Prof. Simon Lavington FBCS FIEE CEng**: *lavis@essex.ac.uk*

### Museum Representatives

*Science Museum:* **Dr Tilly Blyth**: *tilly.blyth@nmsi.ac.uk*
*Bletchley Park Trust:* **Kelsey Griffin**: *kgriffin@bletchleypark.org.uk*
*TNMoC:* **Dr David Hartley FBCS CEng**: *david.hartley@clare.cam.ac.uk*
*MOSI:* **Sarah Baines**: *s.bBaines@mosi.org.uk*

### Project Leaders

*SSEM:* **Chris Burton CEng FIEE FBCS**: *cpb@envex.demon.co.uk*
*Bombe:* **John Harper Hon FBCS CEng MIEE**: *bombeebm@gmail.com*
*Elliott:* **Terry Froggatt CEng MBCS**: *ccs2@tjf.org.uk*
*Software Conservation:* **Dr Dave Holdsworth CEng Hon FBCS**: *ecldh@leeds.ac.uk*
*Elliott 401 & ICT 1301:* **Rod Brown**: *sayhi-torod@shedlandz.co.uk*
*Harwell Dekatron Computer:* **Delwyn Holroyd**: *delwyn@dsl.pipex.com*
*Computer Heritage:* **Prof. Simon Lavington FBCS FIEE CEng**: *lavis@essex.ac.uk*
*DEC:* **Kevin Murrell MBCS**: *kevin.murrell@tnmoc.org*
*Differential Analyser:* **Dr Charles Lindsey FBCS**: *chl@clerew.man.ac.uk*
*ICL 2966/1900:* **Delwyn Holroyd**: *delwyn@dsl.pipex.com*
*Analytical Engine:* **Dr Doron Swade MBE FBCS**: *doron.swade@blueyonder.co.uk*
*EDSAC:* **Dr Andrew Herbert OBE FREng FBCS**: *andrew@herbertfamily.org.uk*
*Bloodhound Missile/Argus:* **Peter Harry**: *pdh@imtex.co.uk*
*IBM Hursley Museum:* **Peter Short**: *peters@slx-online.biz*
*Tony Sale Award:* **Peta Walmisley**: *peta@pwcepis.demon.co.uk*

### Others

**Prof. Martin Campbell-Kelly FBCS**: *m.campbell-kelly@warwick.ac.uk*
**Peter Holland MBCS**: *p.holland@talktalk.net*
**Pete Chilvers**: *pete@pchilvers.plus.com*

## Point of Contact

Readers who have general queries to put to the Society should address them to the Secretary (see page 36 for contact details). Members who move house should notify Kevin Murrell of their new address to ensure that they continue to receive copies of *Resurrection*. Those who are also members of BCS, however, need only notify their change of address to BCS, separate notification to the CCS being unnecessary.

***Resurrection*** is the bulletin of the Computer Conservation Society.
Editor — Dik Leatherdale
Printed by — BCS the Chartered Institute for IT