

```

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

```

# RESURRECTION

The Bulletin of the Computer Conservation Society

# Computer Conservation Society

## Aims and objectives

The Computer Conservation Society (CCS) is a co-operative venture between the British Computer Society (BCS), 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 the 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 the BCS, fees from corporate membership, donations and by the free use of the facilities of both 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.

# **Resurrection**

## **The Bulletin of the Computer Conservation Society**

ISSN 0958-7403

### **Number 60**

### **Winter 2012/3**

### **Contents**

<b>Chair's Report</b>	<b>2</b>
<i>Rachel Burnett</i>	
<b>Society Activity</b>	<b>5</b>
<b>News Round-Up</b>	<b>14</b>
<b>The Tony Sale Awards 2012</b>	<b>15</b>
<i>Rachel Burnett</i>	
<b>Demonstrating Restored &amp; Replica Computers</b>	<b>18</b>
<i>Kevin Murrell</i>	
<b>Programming ENTER : Christopher Strachey's Draughts Program</b>	<b>23</b>
<i>David Link</i>	
<b>Book Review : Pegasus The Seminal Early Computer</b>	<b>32</b>
<i>Dik Leatherdale</i>	
<b>Book Review : Let IT Go</b>	<b>33</b>
<i>Hamish Carmichael</i>	
<b>Obituary: Brian Oakley</b>	<b>34</b>
<i>Martin Campbell-Kelly</i>	
<b>Forthcoming Events</b>	<b>35</b>

---

# Chair's Report

*Rachel Burnett*

---

It is sad to have to report the death in August, of Brian Oakley a former Chairman of CCS.

## **Tony Sale Award**

The inaugural Tony Sale Award, to recognise a singular engineering achievement in the area of computer conservation, sponsored by Google, was presented to Dr David Link for his reconstruction of software for text generation, the Ferranti Mark 1 LoveLetters. Nigel Sale, Tony's son, represented Google as one of the judges, and Margaret Sale, Tony's widow, made the presentation. Brian Oakley had originally been appointed the Chairman of the Judges, and Brian Oakley's widow and daughter attended the ceremony.

## **Publications, publicity and website**

Alan Thomson is retiring from the role of website editor, which he has held since 2007 in co-operation with Kevin Murrell. I would like to thank Alan for all his work. Dik Leatherdale is taking over.

Turing and his Contemporaries, the CCS book published this year, has had good reviews and was voted "best BCS book" in a recent BCS poll of members. We have had to encourage the BCS Publications Department to promote the book more, for example at Turing events and Turing-related venues.

The refurbished Our Computer Heritage website has been implemented and is attracting favourable comment from users.

We have changed printers and distributors since the summer edition of *Resurrection*, edited by Dik Leatherdale, saving costs and improving efficiency.

A photographer, Travis Hodges, has taken portrait photographs of a few CCS members, with links to relevant computer memorabilia, and has developed a display at [www.travishodges.co.uk/gallery/memory](http://www.travishodges.co.uk/gallery/memory)

## **Events Programme — Roger Johnson in London and Gordon Adshead in Manchester**

Just to mention a few of the varied programme of events held in London and Manchester, always well attended:

*A Tribute to Sir Maurice Wilkes* — David Hartley and others

*Misplaced Ingenuity, or Some Dead Ends in Computer History* (about machines towards the end of the punched card era) by Hamish Carmichael, who has retired from the Committee after 16 years, to move to Scotland, or as Hamish put it: "Archiving the Archivist". We shall miss Hamish, and we have thanked him for his work

Two events on Turing

An Atlas 50<sup>th</sup> Anniversary event at the University of Manchester in December is being organised by Simon Lavington.

## **Museums**

### **Science Museum**

Tilly Blyth is now Keeper of Technology and Engineering.

The Science Museum has received its Heritage Lottery bid money, which means that The *Making of Modern Communications* gallery is now assured for opening in 2014.

The Turing Exhibition opened in June, for a year.

### **The National Museum of Computing**

The TNMoC Trustees have appointed David Hartley as Museum Director.

The first edition of the TNMoC guidebook has been published.

The Colossus gallery has been newly refurbished to improve viewing arrangements. Static panels have been installed and the next stage of work is on interactive displays.

A scheme of joint ticketing with the Bletchley Park Trust has been introduced for visits to Bletchley Park which include the Tunny & Colossus galleries.

### **Bletchley Park**

There were 140,000 visitors last year.

Iain Standen has been appointed as CEO. Simon Greenish is to remain as consultant.

The Turing Papers Exhibition opened.

Three-year funding from English Heritage and others has been provided, primarily to repair buildings. C Block is now listed — this was the punched

card operation — and work is being carried out there, and repairs are being carried out to Hut 11A.

Bletchley Park has been nominated for an Arts Council grant.

## **Museum of Science and Industry**

The Museum of Science and Industry is now part of the Science Museum network.

## **Projects**

*Resurrection* reports regularly on the continuing work on the various CCS projects. Picking out a few highlights:

### **Bombe Rebuild — John Harper**

As part of this year's Cheltenham Science Festival, the GCHQ Historical Section challenged the Bombe Rebuild Team at Bletchley Park to intercept messages and from these intercepts to find the daily Enigma settings (*the Key of the Day*) in a manner as close as possible to that used during WWII. Once found, the Key of the Day was used to decipher messages from the general public visiting the Science Festival, who had enciphered them on a genuine German Enigma. GCHQ staff wished to emphasise to the visitors their very important and historic successes back to and before WWII. The Rebuilt Bombe was worked hard and never let itself down.

### **ICT 1301 — Rod Brown**

Events for the public have been held over the spring and summer.

The BBC recorded some interviews, which were broadcast on Radio 4.

The website has been developed. However, there is a risk for the long term future of the machine and project, as its location has recently been put up for sale, and consideration is being given as to the further software recovery and how to secure the machine.

### **Pegasus — Len Hewitt**

At the beginning of October, the first switch-on for over three years took place, to allow the Science Museum engineer to prove that the machine is safe to run. Everyone was delighted with the success.

My thanks to all Committee Officers, Committee Members and Project Leaders, and to all who attend our meetings in London and in Manchester. I have had a most interesting and enjoyable first year as Chair.

---

## Society Activity

---

### ICT 1301 — *Rod Brown*

The project has had a busy spring and summer with no less than four events held at the farm. The Annual Public Open day on the 8<sup>th</sup> of July went well and visitor counts, based on handouts distributed, exceeded 200. Although the rain dampened the day, it actually played to the project's advantage by keeping the machine cool. The machine ran for a solid nine and a half hours on the day at least eight of which were open to public access.

In August the Bromley branch of the University of the Third Age (U3A) visited with a group and the dual appeal of the 1301 system for the retired computer members and the farm's location as a backdrop for the Darling Buds of May for the non computer members worked well.

By opening to the public we find all kinds of connections are made, this time from an individual from the Forest School which took an Elliott 405 way back in 1965. He was contacted by his son and he, in turn, visited the project.

The BBC's *Saturday Live* program on 29<sup>th</sup> September included an "interview" with Flossie (more accurately with Rod Brown). Almost 45 minutes of recording was reduced to a five minute slot which can be found at [www.bbc.co.uk/programmes/b01n0sc5](http://www.bbc.co.uk/programmes/b01n0sc5) (start at 9 minutes 45 seconds).

On the hardware front the project now has an almost fully working 600 line per minute printer. Roger Holmes is working on software to allow children to produce a paper tape on a teletype. Then the machine will read the tape and print out its content as a banner for the child to take away with them. The name of the game is to capture and promote both the project and the CCS and its activities.

Work continues this year to finalise the data capture interface which needs to operate at a rate of four megabits in burst mode to secure the system's contents for the future to be used in emulation.

Work has been undertaken to port a version of the emulator to a Linux platform, with a final target of running the code on a diminutive Linux system which is popular with experimenters. The new Raspberry Pi platform seems to be able to run the code but will require adaption to allow it to load everything from an SD card along with its version of Linux.

In October the project suffered a serious blow when the farm where Flossie lives was put up for sale. Strenuous efforts are being made to find Flossie a new home where she can continue to be restored to full health.

## **Software Preservation — *David Holdsworth***

Following the October London meeting I've had two or three email follow-ups and thought that members might like to know that I have uploaded information to enable any Raspberry Pi owner to repeat my George 3 demo. It is to be found at the head of the software preservation work-in-progress website [sw.ccs.bcs.org](http://sw.ccs.bcs.org). I would like to hear from other CCS projects preserving real machines what software they have for these machines (and possibly others) so that it can be preserved along the lines suggested at the meeting.

## **Pegasus Project — *Len Hewitt & Rod Brown***

We had our first switch on of Pegasus for 3¼ years on 1<sup>st</sup> October! Peter Burton, the engineer the Science Museum has engaged to prove the machine is safe to run, needed to have the machine running for a couple of hours for him to record temperatures and voltages in various areas. Chris Burton, Peter Holland, Rod Brown and I attended from the CCS with Charlotte Connelly from the Science Museum. We had some minor problems which were overcome. One was a fuse blowing episode in the CPU caused by the back wiring having been disturbed in cleaning but this was rectified. We ran with HT on for over an hour with some packages unplugged just to check voltages and temperatures. Then, after some minor repairs, we ran for another hour with all packages in and we were able to do drum transfers and execute instructions on the hand switches. The machine appears to be 95% working.

By then end of October we had completed the testing and certification of the alternator and the supply to the machine. Secondly we have established that the majority of the power supplies are working and have removed many simple power problems from the back plane of the logic racks.

Now we are faced with several tasks to complete and prepare the machine for regular running. At the end of the last work session we had a problem involving a lack of drum clock distribution within the logic. However, we are able to fault find these and any other problems which may be identified now that the power issues are resolved. The team feels that many barriers to progress are now removed.

Pegasus may not be flying just yet, but it does look good for the future now we can work on the machine again. The target of public demonstration by the CCS team remains our principal aim. Even whilst fault-finding the public interest was very high.

We were all delighted at the progress and it speaks volumes about the initial design of the system.



**The late Harold Macmillan looks on as Pete Burton, Charlotte Connelly, Peter Holland Len Hewitt, Chris Burton (and Rod Brown behind the camera) bring Pegasus back to life**

## **EDSAC — Andrew Herbert**

The research and design phase of the project continues to make good progress. Overall our target is to progress from generating clock pulses, splitting out digit pulses, counting pulses and from that, demonstrating main store reading, writing and recirculation. With these elements in place we expect to have explored enough of the machine that the rest should fall into place fairly simply.

Via the Computer Lab and the Wilkes family, the project has acquired further images of EDSAC from one of Wilkes personal photograph albums. Several photographs have surfaced, including several of the original paper tape reader, of a half adder chassis and the control panel. John Deane, President of the Australian Computer Museum Society, has volunteered to investigate reconstruction of the tape reader with his colleagues.

Bill Purvis has continued to develop ELSIE, the EDSAC Logic Simulator, with version 3 now available offering more detailed models of several areas of the machine and also improved representation of the physical properties of EDSAC signals.

In addition, Bill has been scanning the notebooks of Ernest Lenaerts (a Lyons employee seconded to the EDSAC development team to learn about the machine prior to Lyons embarking on designing LEO I). The notebooks shed significant light on the difficulties Wilkes, Renwick and their team encountered when commissioning EDSAC and provide useful information, both direct and indirect, towards resolving a number of design questions.

Chris Burton has determined appropriate designs for all of the key circuits, documented these in an EDSAC Hardware Note and updated others where appropriate. While simply recorded in a sentence this is major progress for the project since circuit design is one of the least well-documented aspects of the original EDSAC. Chris's work provides the foundation EDSAC Replica Project volunteers need to construct prototypes that will be as authentic as possible and consistent with the logical design of the machine. It also helps us reverse engineer circuits from the photographs we have of racks and chassis since we can now identify clusters of components corresponding to specific logic elements.

A major focus has been the store address decoding circuits being developed by John Pratt, supported by Chris Burton. These are now substantially completed with circuit diagrams and chassis layouts. From this work we have been able to identify the function of most of the chassis in the front rack of the machine.

Peter Lawrence has completed a full circuit design and layout for the Clock Pulse Generator chassis and now awaits a chassis to be manufactured and parts supplied to begin construction.

A long-standing question for the project has been the reconstruction of the mercury delay lines used by the original. These are problematic for several reasons: the precision engineering required to manufacture them is demanding, the operational, maintenance and durability aspects of mercury delay lines are challenging, especially for a museum rather than laboratory environment, and the cost of buying the mercury is significant. After investigations by Peter Linington into other storage technologies we have decided that the main store will be constructed using nickel delay lines. These were the immediate successor to mercury delay lines, follow similar physical principles, and are known to be reliable and long-lived in operation. That said, the techniques used in constructing them are essentially lost so Peter has mostly recently been investigating these and has been able to demonstrate a first proof of concept prototype.

Given the importance of mercury delay lines in the history of early computers, the project has given itself the objective of, at a minimum, building a small stand-alone demonstration rig showing a mercury delay line in operation. One route to this might be to experimentally refill the surviving short delay line.

In a similar vein the project has taken the decision to use modern passive components in the replica in the hope of improving the long-term reliability of the machine. In many cases we may be able to obtain modern components of roughly similar dimensions and simply paint them to look like their ancestors, but since most of the passives are hidden inside each chassis they will not be externally visible. The transformers and chokes used throughout the machine have been specified and sample batches manufactured. We will however, use tag strips as the means to support and connect components as in the original and follow the somewhat unsatisfactory (at least to modern eyes) EDSAC approach of using the chassis as signal return. Some tests will be made to see whether we can, in fact, introduce a signal return system independent from chassis ground which could help towards electrical safety measures.

Other volunteers are getting into their stride investigating further aspects of EDSAC: John Sanderson is working on the Digit Pulse Generator and has demonstrated a TTL prototype to enable bench testing of other chassis. David Laine is exploring the half-adder and adder circuits. Andrew Herbert has built jigs to manufacture lumped delay line formers and Alan Clarke is working with a Cambridge company that has resurrected its vintage wave-winding machine to

produce the associated coils. (After a successful first trial the winder's motor burned out, so that now has to be rewound before we can go into manufacture). A new volunteer, Andrew Brown from Southampton University, is starting to investigate uniselectors and the initial orders loading system. Peter Tomlinson and a colleague Ian Pearson have done some preliminary work on an interim semiconductor store design. Looking to the longer term, Simon Moore of the Cambridge University Computer Laboratory is taking an interest in how we might use modern microelectronics to set up a non-intrusive monitoring and diagnostic capability for the replica when it becomes operational. Chris Smith has identified potential source of power supplies.

In his role as store keeper, Alan Clarke has been successful in locating a significant fraction of the valves and holders needed by the project, to the extent we generated a storage crisis at the National Museum of Computing, which Alan resolving by buying a cargo container now in place at the back of the museum. The container is equipped with racking, a bench and the growing inventory of project tools and equipment as well as the stock of components. He has also been the main link with Teversham Engineering in Cambridge which has made two sample chassis to a very high and realistic standard.

Alan has also been developing improved text for the CCS, TNMoC and EDSAC.org web pages on the project. Working with Stephen Fleming from TNMoC this content will be hosted using the TNMoC content management system and linked into their site structure as well as having its own home page and local structure.

Two further marketing/education sub-projects are underway: a display for the museum and a video blog. The design of an EDSAC display stand for TNMoC is in progress with Andy Partridge, TNMoC's design consultant. A portable tower and display case have been procured and will be covered with graphic panels giving information about EDSAC, Wilkes, the project and EDSAC artefacts (loaned from the Computer Lab) and their replicas put on display. It is also planned to include a "rolling demo" of Martin Campbell-Kelly's EDSAC emulator.

David Allen, a retired TV producer, who had worked on the original BBC Micro TV programmes, is building video history of the project. Starting from a day's worth of material obtained from the volunteers meeting on 16<sup>th</sup> August 2012 he has the first cut of a project overview video and a series of smaller videos on individual aspects of the machine that will be the basis for a project "blog".

As these activities around communication and marketing are being progressed it is increasingly clear that an important result of the project in addition to the hoped for replica machine will be the recovered design knowledge and pedagogical material on lost techniques of valve circuit design, computer

architecture and pre-transistor computer engineering. Consequently the project is ensuring that we keep a formal record of our investigations, prototypes and designs as an educational and scholarly resource.

From the project leader's perspective progress overall is good, although some areas are more ahead than others. We are on the threshold of making and testing several key parts of the replica. The next question is how we should go about "mass production" and organise the erection of the replica at TNMoC.

### **Harwell Decatron Project — *Johan Iverson***

Delwyn has been doing a sterling job and all the units have now been bench tested and re-installed in the machine. So we are in the final stages of testing the units in the machine. Various faults have been found resulting in some minor changes to the original design and component values. This has been necessary to shape pulses and to bring borderline timing into spec. We have been able enter a number from the keypad into the accumulator or to a store location and to execute a clear store instruction. There have been ongoing problems of having to replace decatron anode resistors as they go open circuit. Initially there was a high failure rate, but this has settled down to about one a week. While replacing faulty anode resistors we also discovered another unsoldered joint.

Currently work to restore the peripherals is underway. The Creed 75 has been cleaned and adjusted it seems to be working and the same for the printer but it is still resisting and needs some more work.

Also after some research the Birmingham Collection Centre was revisited to search for some missing Witch components. Two out of three items were found, a connection box for connecting the printer on the edit station directly to the machine and a Creed 7B keyboard perforator which has been modified to work in the WITCH 5-wire code. The other item, some kind of perforator possibly with ticker tape printing, could not be found.

Also now the computer is being run, seeing how many relay operations are required as it runs, it will be necessary to set up some sort of a routine maintenance schedule for the relay sets.

### **Hartree Differential Analyser — *Charles Lindsey***

The machine can now run demonstrations regularly and reliably. The setup was altered to reduce gearing down in the chain. This resulted in faster running (you can see the turntables accelerating to full speed as they approach the centre when generating a sine wave), but the gap when drawing a circle is much larger.

We now normally demonstrate drawing a sine wave, but sometimes alter the setup to draw a circle as well.

Attention has now turned to fitting an inverter to enable the drive motor speed to be altered/reversed. This is necessary before realistic use of the input table is possible. The input table has been cleaned up and lubricated and its positioning is being adjusted to line up its shafts. But no usage of it will be possible until we can access the spare gears etc. currently in offsite storage.

To help explain the distinction between analogue and digital computing, a large 3½ ft long slide rule is under construction.

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

Several small but significant organisational and historical steps have taken since the last report. We are working closely with Science Museum archivists on resolving referencing anomalies between existing listings and the new digital archive of Babbage's technical material and identifying omitted material — this in advance of wider dissemination of the archive by the Science Museum. This is detailed work covering some 7000 manuscript sheets. I have also provided input and advice for the small Science Museum exhibition of Babbage's drawings planned for later this year to coincide with the Science Museum's public announcement of the digital archive. The formalities of registering as a charity and putting in place gift-aid facilities for donors and sponsors are now complete. Fundraising will now go into full swing starting with a public announcement, agreed with the Science Museum, about *Plan 28* (the name of the project to construct an Analytical Engine).

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

#### **Elliott 803.**

After thorough testing and repair, the boards needed to add 13 bit parallel I/O to the 803 have been installed. The only problem encountered was with the very precise adjustments needed when setting up the shorter accumulator delay line. These adjustments were greatly helped by using the original 803 commissioning programme COM238. This puts random patterns into the delay line and leaves it there for a period before checking it and punching a difference tape if any bits have changed. Running COM238 for a while now forms part of the machine tests that are run when the machine is first turned on most Saturdays.

Once the delay line was correctly adjusted, a simple one bit input and output peripheral was designed to investigate any problems with using minilog gate circuits built with modern silicon PNP transistors. A few issues were found in

monostable circuits where the lower maximum reverse base voltage of silicon transistors was being exceeded. An additional isolating diode solved these problems. Currently three 13-way cables are being terminated with taper pins to make all the required data and control signal connections to the CPU.

### **Elliott 903.**

The 903 has been running well for several months now with only two minor problems since the last report. The first fault manifested itself in July. Using the spare boards in the scrap 903, this was narrowed down to a faulty A-GB microcode board in slot M18. On further investigation there was a leaky diode D6 which forms part of the data matrix. This was replaced by a diode which looked similar, from a bag named "Elliott parts", and the board has worked ever since.

Secondly, at the end of August, there was a loud "pop" from the paper tape reader and a fuse blew. The culprit was the insulation of the mains power cable into the tape reader motor. The cable was duly cut out, and a new cable and fuse were fitted.

Initial attempts to load BASIC programs into the 903 from a PC using Hyperterminal ran into timing problems. BASIC checks each line as it reads it, but there is no protocol to tell Hyperterminal to wait for this. The problem was solved using a simple fixed delay.

### **The Plotter Paper appeal.**

The appeal for Benson Lehner plotter paper in *Resurrection 59* produced the offer of one part-reel of suitable graph paper, but no vast hidden horde. However, a company has been found which is prepared to make some paper, minimum order one dozen reels, at a not too impossible price.

By coincidence, whilst that *Resurrection* was at the printers, an identical Benson Lehner plotter and six reels of plotter paper appeared on the BBC Television news on 7<sup>th</sup> August, in the obituary film clip of Sir Bernard Lovell holding a length of paper tape at Jodrell Bank.

## **CCS Website Information**

The Society has its own website, which is located at [ccs.bcs.org](http://ccs.bcs.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](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.

---

## News Round-Up

---

To Southampton University for a memorial seminar for the late David Barron at which both Martin Campbell-Kelly (“Purveyor of obituaries to the Computer Science gentry” as David used to put it) and David Hartley spoke, as did several others. What became obvious as the afternoon rolled on was not just the esteem, but the deep affection in which David was held by everybody present.

Readers will recall that David kindly volunteered to be *Resurrection’s* assistant editor a few years ago. His health prevented him from contributing as much as we both would have liked, but we conducted a lively email conversation for several years. Your editor also misses him greatly.

101010101

In *Resurrection 59* we noted the proliferation of Turing blue plaques and the absence of any such here in sunny Teddington. Now we hear of a proposal to site one of Michael Gove’s Free Schools in a redundant building at the National Physical Laboratory. Its name? *The Turing House School*. It is, however, at the opposite end of the site from the building in which ACE was conceived.

101010101

TNMoC’s upcoming software gallery aspires to assemble a collection of programs written in as many programming languages as possible. So they have issued a challenge to programmers to write a program to calculate and display or print the first 20 prime numbers. A competition is implied, albeit there is no prize beyond the glory. You need to submit a listing of the program and its output, the name and version number of the language and the platform (hardware, operating system etc). Your name and email address is required too. Jill Clarke ([swg@talkingbear.co.uk](mailto:swg@talkingbear.co.uk)) is your contact.

CCS members collectively have the knowledge and resources to submit dozens of entries written in the most obscure languages imaginable. Your Society expects great things of you. We will prevail!

101010101

We are sad to discover that after close on 40 years, AXiS, formerly the ICL 2900 User Group, has reached the end of the road and has been wound up. At its peak AXiS organised an annual two-day conference at York University attended by hundreds of delegates and which boasted over 120 one hour-long presentations run in up to six simultaneous streams.

---

## The Tony Sale Award 2012

*Rachel Burnett*

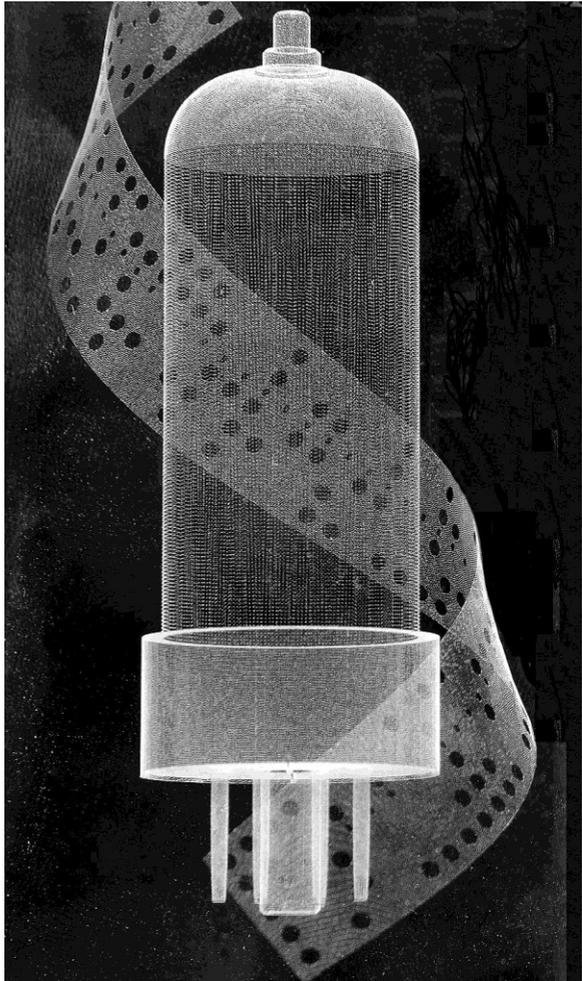
---

The Tony Sale Award, sponsored by Google, was set up to recognise singular engineering achievements in the growing area of computer conservation. The first Award was presented at a ceremony held on 11<sup>th</sup> October, at the BCS in London.

The winning project is the Ferranti Mark I LoveLetters, reconstruction of software for text generation submitted by Dr David Link who is based in Cologne.

This computer art installation is a functional replica of the 1951 Ferranti Mark 1 computer (an industrial version of the Manchester Baby, the first fully electronic universal computer controlled by software).

David Link reconstructed software developed by one of the very first software developers. In 1953-1954, using the programming system devised by Alan Turing, Christopher Strachey used the built-in random generator of the Ferranti Mark I to generate texts intended to express and arouse emotions — or, automated 'love letters'.



**The Tony Sale Award** (© ccs)

The programs used then have mostly been lost, and 50 years later the number of surviving contemporary witnesses was very small. The source code of the software survived, but the algorithm could not be read and it was impossible even to guess at the functionality of the program. To analyse it, the original machine's software had to be resurrected and the algorithm re-run.

David Link first rebuilt the Mark I in software, and then built a functional replica. In 2006 he tried to execute the source code, but a subroutine was missing, and he

coded the procedure itself for integration with the authentic files, to be able to generate the texts. Meanwhile the real routine was found and incorporated.

The installation was first on display in 2009 at the Centre for Art & Media Technology, Karlsruhe, one of the leading media art museums in Europe. It has been on display since at different centres in England and Germany. The visitor interacts with a physical functional replica of the Ferranti Mark 1, a hybrid hardware/software simulation including many original components of the time, which conveys an authentic impression of the look-and-feel of the first computer. By executing the original code of Strachey's software, this installation



**David Link** (© ccs)

continuously generates texts. The concept enables visitors to publish algorithmically generated love letters.

It is complete in itself and is also part of a larger project to safeguard the entire scientific Mark I software from 1948-1958.

The project's fusion of art, engineering and history celebrates one of the first artistic applications of the computer in a visually attractive way. It is conceptually brilliant and technically impressive in its research and reconstruction, with wide cultural appeal, originality and a touch of genius.

The runners up were, in alphabetical order:

- DEC PDP1 restoration, submitted by Dag Spicer in California. This is a computer restoration project of the first minicomputer, focusing on interactivity and affordability, launched in 1959. It is a meritorious highly professional conservation project. The restoration has been in continuous operation since 2006, for demonstration to the general public in the Computer History Museum in California.
- Time-Line Computer Archive, submitted by Sandra Hodson, George Scott and John (Binky) Armstrong on behalf of Michael Armstrong & Sandra Hodson, in Wigton, West Cumbria, UK. This project aims to collect, restore and exhibit all types of early computers and electronics. It can be viewed by the public in Wigton, and also worldwide through its website. It will be moving into purpose-built premises shortly. This is a worthy and ambitious project, run with limited resources.
- Z3 reconstruction in Hünsfeld, Germany, submitted by Professor Doctor Raúl Rojas who is based in Berlin. This is a reconstruction of Konrad Zuse's Z3 Computer and its simulation, originally built in Berlin from 1938 – 1941 and destroyed during World War II. Impressive work has been undertaken over several years to understand the architecture, design and operation. The machine was finished in 2001. It is part of the collection of the Zuse Museum in Hünsfeld, Germany and is operational. The project shows astonishing dedication from a single individual. There is original and substantial research and reconstruction. All the diagrams of the machine are available through the Zuse internet archive maintained by Professor Rojas in Germany – the first internet archive dedicated to the work of the computer pioneer.

---

## **Demonstrating Restored & Replica Computers**

*Kevin Murrell*

---

Restoring or reproducing historically important computer systems is just the first part in a process of preparing a system to be shown to the public. Choosing relevant and comprehensible software to demonstrate is just as tricky a job and we often settle for historically accurate but potentially boring routines or making our machines play parlour-tricks to entertain the public.

As members of the Computer Conservation Society, many of us have been or still are involved in the restoration of aged computer systems or in the building of replica machines. These computers are typically either one-off machines that are particularly important in the history of UK computing or commercial systems that had a particularly successful sales history.

Some early 'computer-like' systems were designed to perform a single task with only limited options to modify that task – the Colossus machine at Bletchley Park is a particular example in that its sole use was part of the process of decoding of war time encrypted messages. All we can do in these cases is to attempt to display the machine performing its original task and explain a difficult technical subject as best we can.

Once we have a working replica or have restored a machine to working order, we naturally want to demonstrate and show the system off to our colleagues and fellow CCS members. This is relatively straightforward as they will understand the operation of the computer and appreciate a detailed demonstration of a particularly special feature of the machine. With our fellow enthusiasts, we can often 'get away' with simply showing some features of the operating system, or the compilation of a COBOL program from a stack of 80 column cards. We already understand the knowledge required and the sheer amount of effort put in by the engineers just to get to this point!

However, if we are planning to show the working machine to members of the public, in a museum gallery perhaps, we need to design a demonstration to illustrate the typical original use of the machine that is also understandable by the visiting public.

For the purposes of this paper, I am solely concerned with 'manned' exhibitions. The presentation of non-working static machines is better understood and falls within the realms of the professional museum curator. However, to show the machine running we must rely on software to give the machine life.

Early pioneering stored-program computers were often designed and built to be used in education and research, and typically ran hundreds of jobs each month, submitted from a large cohort of users. A typical example is the EDSAC computer built at Cambridge University and used by many hundreds of students and researchers. Many jobs for the machine would only have been run once, albeit after several debugging test runs, particularly when used as part of a programming course. Mass-produced systems, such as the ICT 1301, were sold to a wide variety of commercial customers, but once commissioned, would typically run a limited set of programs repeatedly, or indeed continuously. The archetypal job of processing a series of stock movements and updating a stock master file would be run every day as part of a long series of scheduled operations.

Original application software may not be available for the restored machine and therefore cannot be demonstrated, or what software remains available might have been part of a long series of jobs that now makes little sense when run in isolation. It may also be that the software available has long since parted company with its documentation and its use is obscure! As an example, the limited programs we have for the Harwell Dekatron machine range from mathematically complex routines for calculating bubble sizes in the water-cooling systems of early atomic power-stations, to trivial routines designed to test the multiplication function of the computer, with precious little in-between!

This is not a new problem. When NPL demonstrated the Pilot ACE computer in 1950 they presented three programs to the assembled press and public. One was a real program that had been run in anger – calculating the various ray paths through a series of optics, and creating a table of the results. The other two were to find the highest common factor of an integer, and to calculate on which day of the week an arbitrary date fell. Whether the first program actually worked or not is a moot point as most of the audience would not have been in a position to understand the algorithm, let alone challenge the results! They would at least have had a chance with the other two programs – especially after the results had been ‘translated’ from the backwards-binary presented on the Pilot ACE console.

Given that we wish to demonstrate fundamentally static number-crunching computers, like the Harwell machine, to members of the public, are we destined to only have it run trivial programs that calculate birthdays or can we find an engaging alternative? Lest anyone think this is only a problem for a unique early 1950s valve and relay based machine, exactly the same issues apply to a working Cray Supercomputer not two hundred yards from the Harwell machine at The National Museum of Computing!



What does work well when members of the public are involved is physical movement.

Whether this is a magnetic tape deck spooling tape from one reel to the other, a pen-plotter gradually drawing an image or a fast and noisy

line printer spewing paper, the public are generally fascinated and will often watch for ages. A particularly good example at the museum is the drum graph plotter connected to the Elliott 803 computer that gradually 'draws' the results of an ALGOL program running on the machine. The visually fascinating plotter grabs the attention of the visitor and leads them into learning a little more about the machine itself. In common with similar systems, the Elliott 803 also has another party-piece in producing music (of a sort) made by connecting an amplifier and loudspeaker to a control signal in part of the CPU. Programs were written to replay a coded sequence, and these original routines can be updated to include more modern or at least more recognisable tunes.

One of the few systems at the museum that runs exactly the same software as it did during its productive life is the PDP11-based display station that was saved from the London Air Traffic Control Centre at West Drayton near



London. Pre-recorded radar signals are replayed back into a series of computers that produce a map based display of aircraft movements on an impressive

circular green CRT display. This very complete restoration project was only made possible by spending a lot of time learning about the system before it was decommissioned and ensuring that everything in terms of software, manuals and backup tapes were captured at the same time as the hardware. This required a group of volunteers with a complex set of skills combining those of museum curator, hardware engineer and systems programmer, and all having sufficient time to fully understand the system before it was moved. It was also important that as little time as possible was lost between rescuing the system and putting together the working display so that the intimate knowledge of how the system worked was not lost, and that the original engineers who supported the system were both still interested and available to offer advice.

Another iconic DEC based system has also been restored and is now regularly demonstrated using original software. The PDP-1 at the Computer History Museum in California is regularly shown running an interactive video game, developed at MIT in 1962, called 'Spacewar!' — one of the earliest digital graphical computer games. Although the PDP-1 is an honourable candidate for restoration, and the restored machine a tribute to those involved, having such an iconic and visually interesting application that positively encourages visitor interaction was surely an irresistible goal.

We must also consider that visitors to a dedicated computer museum range from those with a marginal interest in computing that have 'come along for the ride', to those with wide experience and a deep understanding of the mechanics of computer hardware and software. Some visitors will arrive to see one particular machine running and will already be quite expert in its history and operation. We therefore need to be able to present the working machine differently for the each of these different audiences. We already provide printed information in a manner that allows the visitor to dig as deep as they wish into the technical details of the machine, and we need to produce working demonstrations that can offer a similar layering of information. Many visitors will only spend a minute or two with the machine, but some may well spend the whole day chatting and getting involved.

One word of warning: it is tempting for the enthusiasts to hack together certain technically satisfying demonstrations such as using a modern single chip microprocessor to emulate a particular machine and have it connected to the original artefact. For instance, it is quite possible (and technically very satisfying) to build a cluster of OpenVMS servers using a mixture of original hardware and modern emulators running on a Raspberry Pi! I believe this only confuses visitors

and is either avoided, or at best kept 'under the counter' to be shown to the real enthusiast!

Some of the machines we display are simply impressive when running, for example the tape whirling around the bedstead on Colossus or the endless flashing lights and clicking of the Harwell machine, but often all that can be seen from a working machine is the output – be it printed or displayed on a screen. Many modern machines have no inherently fascinating features like those already mentioned and it is only the output of the machine we can use to demonstrate to the public.

We need to continue to search for better methods of displaying our restored machines to do justice to the machines themselves and their designers, and yet still entertain and educate interested visitors. We must decide on what is the key feature of a machine that we would like to convey to visitors, and understand that it is better they leave understanding and remembering one key fact about the machine rather than misunderstanding five.

When displaying many running computers throughout the museum, we also need to consider comparing the machines with each other. This is just about possible when limiting the comparison to systems of a narrowly defined era; for instance, when comparing the features of different home computers produced in the early 1980s we might contrast the resolution and colour depths of each display by showing the same colour image on each. When comparing machines designed to do similar jobs, but which were produced decades apart, like the Harwell Computer and the Cray supercomputer, we can try to demonstrate the same algorithm running on each machine, but the results are often comical as the relative performance of the machines is many orders of magnitude apart! Relying on original software is rarely the answer, and nor would I say is preparing trivial parlour-game tricks such as calculating which day of the week an arbitrary date falls. What is needed, I believe, is newly written software to succinctly demonstrate the original use of the machine but at a level that the average visitor can follow and appreciate. That new software needs to demonstrate the 'unique selling point' of the machine and explain why the machine and its use are so particularly important in the developments of computer technology.

I welcome comment and advice from our members.

*Editor's note: Kevin Murrell's day job is technical director of Savience Ltd. He is also a trustee and director of The National Museum of Computing and somehow finds time to serve as secretary of the Computer Conservation Society. He can be contacted at [kevin.murrell@tnmoc.org](mailto:kevin.murrell@tnmoc.org).*

---

# **Programming ENTER: Christopher Strachey's Draughts Program**

*David Link*

---

This article details some problems – and some solutions – encountered when resurrecting a program for the game of draughts from 1951 on an emulator of the Ferranti Mark I.

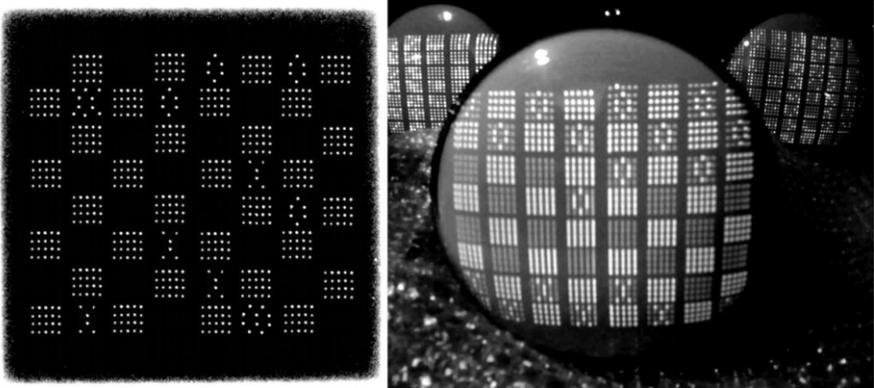
The Ferranti Mark I was the industrial version of the Manchester Mark I, whose prototype, the Manchester "Baby" (SSEM), performed its first calculation on 21<sup>st</sup> June 1948. The algorithm here described was one of the earliest complex applications authored on the pioneer computer that did not only serve system testing purposes. Christopher Strachey, an outsider to the Manchester computer laboratory and a school teacher, had developed the software in his spare time. Martin Campbell-Kelly writes, relying on the oral histories from lab personnel:

"Strachey sent his programme [draughts] for punching beforehand. The programme was about 20 pages long (over a thousand instructions), and the naiveté of a first-time user attempting a programme of such length caused not a little amusement among the programmers in the laboratory. Anyway, the day came and Strachey loaded his programme into the Mark I. After a couple of errors were fixed, the programme ran straight through and finished by playing *God Save the King* on the hooter (loudspeaker). On that day Strachey acquired a formidable reputation as a programmer that he never lost."

The material relating to the draughts program has been preserved in the Strachey papers in the Bodleian Library, Oxford. In it, there are found approximately five versions of an algorithm that is about 20 pages long, pencilled on the usual Manchester coding sheets. There are also printouts of sample games Strachey played against the machine at the time, which were recorded on the teleprinter. Dates on the papers indicate the software was mainly developed in June and July 1952. A first, undated version was probably written prior to May 1951. (In a letter dated 15th May 1951, Strachey wrote to Turing: "I have completed my first effort at the Draughts" and he was obviously talking about the Manchester Mark I. At this point, the algorithm already had "input and output arrangements"). In February 1951, the Ferranti Mark I had been installed in Manchester University. Strachey gave a lecture about Draughts at the ACM conference in Toronto in September 1952, which was published in the Proceedings.

## Game Machine User Experience

When the software started, it asked the user to **PLEASE READ THE INSTRUCTION CARD** on the teleprinter. He would then hit a key labelled "KAC" on the console to signal he had done so. The algorithm asked him to spin a coin and claimed either heads or tails. The user let the program know via a switch and KAC if it had won or not to determine who had the right to start the game. Then human and machine made moves alternately, the latter by printing them on the teletype, the former by setting the hand-switches on the console and hitting KAC. The complete game was printed out, and two consecutive situations could always be inspected in parallel graphically on cathode ray tubes 3 and 5, which were part of the working memory of the machine. The software very probably constitutes the first usage of a graphical display in a computer program.

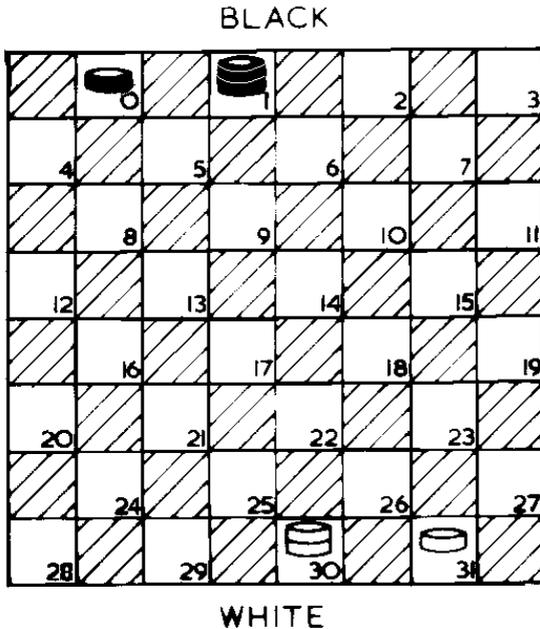


**The draughts board as shown by the storage CRT of the Ferranti Mark I, and a modern recreation by the author**

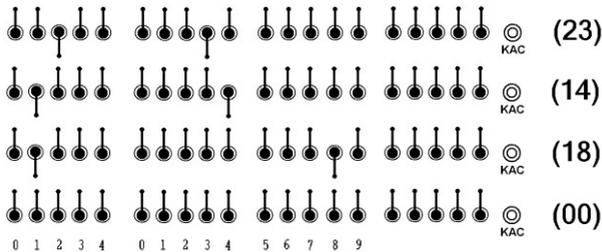
Strachey had coded an additional "preview feature": after the user had announced his move by setting it up on the switches, the machine showed the resulting position on cathode ray tube 3. If he then answered NO by composing **///T** on the console (bit 19 on), the algorithm reverted to the previous situation on the board and he could try something else. If the user input wrong information, the machine became increasingly angry, until it uttered: **I REFUSE TO WASTE ANY MORE TIME. GO AND PLAY WITH A HUMAN BEING/.** (The slash was probably used as an exclamation mark, which was missing in the teleprinter code.) A similar routine existed if the opponent took too long to reply. Strachey had apparently become fascinated with the slightly obscene theatrical effect of a machine making human-like statements and showing "emotion". His

next software was an inversion of this rather strict, impatient character, a program for the composition of love letters. Draughts already contains the complete "rhetoric" that is needed for it algorithmically, including the selection of pre-fabricated text based on random numbers.

### Coding a Game



In this way, sequences such as "23-14 (18)" could be expressed as:



(To end the move sequence and return control to the machine, the user had to hit KAC with nothing set. The first bit in each group signifies 0, the second 1, and so forth.)

The strategy implemented in the game algorithm was a heuristic one, so one could claim draughts was the first heuristic program too. Strachey wrote that the difficulty of "the machine to look ahead for a number of moves and choose its move by a valuation scheme" was of great theoretical interest and presented a typical example of a "large logical programme". Other games were less challenging, because "most of them use the complete mathematical theory so that the outcome ... is no longer uncertain". His program calculated the next move by looking ahead for a certain number of steps without being able to overview the complete game. By not trying to exhaust the endless number of combinatorial possibilities, he "succeeded in making a programme ..., which will play a complete game of Draughts at a reasonable speed." In fact, this is not true: There is no code to control the end game, to detect it is over and to announce a winner. To write a program that could handle the rather complex task of playing draughts must have been sensational at the time.

The central element in the heuristics of the algorithm was the evaluation function for future positions. In it, the machine calculated all possible moves up to a certain depth and summed up the material left on the board resulting from each, counting three for a king and one for a normal piece. Theoretically, i.e. from the perspective of storage space, the algorithm could look ahead three operations on each side (with depth = 6), but in fact, due to the much more pressing limits on time, it was in most cases only anticipating three in total (depth = 3). Quite typically, as actual program performance can be very different from the planned one, the strategy had the serious flaw that the machine started to behave suicidally: As a result of the valuation scheme, it sacrificed all its pieces when the user was about to king. Strachey met this by adopting two different depths of search in such a way that in case one of the last two moves had been a capture, the machine calculated on. After that, it kept looking ahead until the second

depth value was reached. (In the run on 10.7.1952, this value (b) had been 1, with a (normal search) = 5.)

Strachey had separated the strategic core of the algorithm from the service functions and commented: "It is rather typical of a logical programme: that this organising routine is in fact longer than the game-playing routine proper." The latter was called DRAUGHTSB or DR/B and consisted of eight pages (in the version dated 10.7.1952), while for the service part (DRAUGHTSC) occupied another ten sheets, with four containing auxiliary functions. So, 18 or 22 pages in total, depending on the method of counting — incredibly long for the time.

## Resurrecting Draughts

In the course of software reconstruction, usually parts start to work while others still malfunction and ultimately lead to a crash of the program one tries to resurrect. One technique here is to follow the algorithm through and to find the exact point where it starts to go wrong. This is usually slightly earlier in the executed code than the final crash. (It is astonishing how long programs can sometimes run on completely wrong grounds.)

When the exact position of the aberration is found, this particular place in the code can be investigated and probably be fixed, provided the situation is not too complex. The software will then continue to execute, until it encounters another crash point, or ideally run through to the end, in which case the reconstruction succeeded. This technique of debugging already existed in the 1950s and there were dedicated "check sheets" to trace or log a program at runtime, i.e. to record the memory locations that changed in the sequence of the operations of the algorithm.

In one such situation in the beginning of the resurrection of Draughts, the program was waiting for some time, and then went to a "hoot stop". This was the symbolic equivalent of a crash, by which the software signalled that something had gone fundamentally wrong.

Upon closer inspection, the algorithm was stuck in the following lines (see note below for an explanation of the notation):

```
1 - T/: Accumulator (A) = 147456
2 - TN: A = A-1
3 - /M: go to line 2 if A >= 0
4 - /I: switch M and L, the left and the right side of (A)
5 - /H: continue execution of program if A >= 0
6 - /T: go to hoot stop
```

In line 1, the 80-bit accumulator is set to a rather high number, 147456, by copying it from address VK in the working memory. It then counts this quantity down by subtracting the contents of address E: from it, which holds 1. This location is part of two pages of values that are kept in memory permanently, PERM. The third line is a conditional statement: If the accumulator is greater than 0, go to operation 2, where the number is again decremented. At one point, the value there will change from *//////// //* to *#####* *#####*, that is, from 0 to -1. Since a command takes 1.2 milliseconds to complete on the average, this will happen after approximately 5.9 minutes. The algorithm then continues in line 4. The operation here exchanges the left (L) and the right (M) 40 bits of the accumulator. Since it is set to all 1s, this produces the same number, *#####* *#####*, which is -1. In line 5, the algorithm jumps to what is obviously the continuation of the program, if and only if the quantity in the accumulator is positive! Otherwise it enters the already-mentioned hoot stop – an endless loop with no break condition, which consists of the following two lines:

```
1 - /V: hoot
2 - /P: go to 1
```

In modern notation, the algorithm we just discussed could be rewritten in the following way:

```
int i = 147456;
while(i >= 0) i--;
switchMandL(i);
if(i < 0) hootStop();
else continue program execution
```

This code seemed to make no sense at all! To understand it, it is useful to consider how signed numbers were represented in the Manchester Mark I. Generally, these were 40 or 80 “binary digits”, written with the most significant bit to the right. The handbook specified: “On the plus-minus convention the most significant digit is used to represent the sign.” To find out if the number in the 80-bit accumulator was positive, it was sufficient to have a look at bit [79]: When it was 1, the number was negative. The machine automatically copied the value of this bit to the A-sign flip flop, and in case of an A-conditional statement, it consulted the data there. So again: How could the switching of the two sides of an accumulator full of 1s result in the 79th bit becoming zero? Apparently, the algorithm expected something that could never happen, an impossible event. Formulated differently, it was waiting for a miracle. (In very much the same way, the tautology `while(true)`, which encloses the run loop in the core of most

programmes, can only be broken in the improbable event that truth is no longer truth.

In the operating instructions, Strachey wrote that the "machine gives a 'pip-pip' signal when it requires attention. It should always be restarted by operating KAC after it [the machine] has been set appropriately." He went on to give examples of what the computer would say and in which way to react to it. The KAC key was one of the several clearing switches the Manchester Mark I inherited from the "Baby" prototype and its function was to empty the accumulator. But would hitting KAC not lead to the same situation as counting it down until it reached zero? In both cases, the accumulator would first become all zeroes, = 0, and then all ones, = -1, when it was decremented in line 2. It was impossible to see any reason why the switching of the two parts would make the number positive. And yet, it was quite obvious that the code in question could do exactly this: tell the difference between counting down and hitting KAC.

## **Analogies in Logical Design**

The solution to the riddle was that Strachey relied in his programming on the logical design of the Mark I, its hardware properties. I failed to make sense out of the code fragment for a rather long time, because I was looking at the machine on a purely symbolic level, where signs were transformed into other signs instructed by signs. The emulator was only an implementation of the Mark I's operation codes and its effects on the contents of the stores. In this mode of thought, pushing a button was treated like a command, and more importantly, like a synchronous one. There was no difference between KAC and the operation code  $\mathbb{T}$ :, which also cleared the accumulator.

In writing, meaning is conveyed by material elements, the words and letters. In the same way, the data and operations in computers are represented by certain real systems with suitable properties, by a physical analogy. The function to clear the accumulator is implemented in certain electronic components, a Williams tube by the name of "A", in a way that follows the logic of this device. Since something is stored here if it is refreshed, it is sufficient to prevent recirculation to delete the data.

But it is not only the spatial physical analogy that counts, but also the temporal aspects of this simulation of thought processes. On the most basic level, computers move in cycles, which are subdivided in a number of phases in which certain predefined elementary actions take place. In the Manchester Mark I, there were seven of them: SCAN1 to SCAN3, and ACTION1 to ACTION4. The timing with respect to these also determined if an operation was synchronous or

asynchronous. In the logical design picture of the machine we are interested in here, it is important in which phase certain parts of commands are executed.

## Programming ENTER

In comparison to the activities of the algorithm when it counts down the accumulator from 147456, what happens differently on the logical design level when the user hits KAC? The 80 bits of A are set to 0 and the software subtracts 1 from it, making it negative, the A-sign flip flop is set and the program breaks from the first loop. What is important here is that the sign of A is identified after the arithmetic, but before the number re-circulated returns to the accumulator. Upon leaving the loop, the number (-1) is not re-circulated and hence A is empty again. When the algorithm switches M and L in a later cycle, the A-sign flip flop is clear and the program jumps to its continuation, not to the loop stop. The rather elaborate sequence thus simply detects if the KAC key has been pressed. In that case, the software jumps into the following code fragment:

```
1 - /J: M += //// ///E
2 - /H: go to 1 if A >= 0
```

First, a number is added to the right part of the accumulator, equivalent to adding 1 into its 75th bit. Then, if the number is not negative, the procedure is repeated. Again, it seems quite impossible that by adding a positive quantity, the result can become negative. Obviously, the user is still holding down KAC when these statements are reached, which prevents recirculation, leaving the accumulator empty. Once the key is released, it starts to increment and at the 16th addition this carries over into the sign bit. The algorithm jumps to its continuation. The code thus detects the release of KAC and waits if it stably stays in this position to prevent accidental bouncing of contacts to disturb user interaction. With the fragments described above, it formed a detection sequence for the typing (press / release) of the key.

“Phew! that was a good exercise”, wrote Christopher P. Burton after mostly he had found out what the mysterious fragments meant. So the code was actually not waiting for the impossible. Strachey had simply constructed in software what would today be called an ENTER key. He needed it because of the way in which the user should communicate with the software: He set up on the console hand-switches an answer like the next move to be played and signalled he had composed it by depressing ENTER. Interestingly, no key to “send” the carefully composed information to the machine existed on the console of the Mark I. But luckily enough, with some ingenuity, it could be programmed.

## The Manchester Mark I and its Notation

The Draughts program ran on the Ferranti version of the Manchester Mark I and Strachey used the notation established by Turing in the programming manual. The machine was based on a 20-bit word, and 20-bit numbers (and also instructions) were specified as four 5-bit elements, each element taking the name of the teleprinter code equivalent to the 5-bit value. Thus binary '00000' was expressed as '/' and '10000' (least significant on the left) as 'E'. The written form of numbers and instruction was quite opaque unless one was very familiar with all 32 of the possible teleprinter codes. The 5-bit value '11111' was written as '£'.

Most instructions contained a function (operation) number and a store address. The function number was six bits long so could be expressed as two teleprinter characters, the first of which was always '/' or 'T' ('00000' or '00001'). The instructions relevant to this article have the following meanings:

/H	Jump direct if accumulator $\geq$ 0	/T	Jump direct unconditionally
/I	Exchange most and least significant halves of accumulator	/V	Hoot (sound the loudspeaker)
/J	Add contents of a store location to most significant half of accumulator	T/	Load accumulator with contents of a store location
/M	Jump relatively if accumulator is $\geq$ 0	TN	Subtract contents of a store location from accumulator
/P	Jump relatively unconditionally	T:	Clear the accumulator

The Accumulator is 80 bits long, containing four 20-bit words. The most significant bit (bit 79) is the sign bit, 0 meaning that the number in the accumulator is zero or positive, and 1 meaning the number is negative.

*The author is indebted to Chris Burton for solving this, and other, enigmas. David Link would be very happy to hear from all readers who remember Strachey's draughts program. He would also be extremely grateful for any hint on the whereabouts of other Mark I software, especially in the areas of meteorology, nuclear physics and chess. He can be contacted at [david@khm.de](mailto:david@khm.de); his website is [alpha60.de](http://alpha60.de).*

---

# Book Review : Pegasus The Seminal Early Computer

*Dik Leatherdale*

---

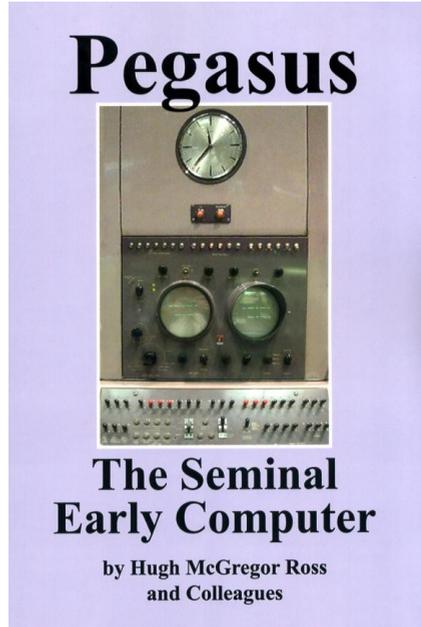
This new book by prominent CCS member Hugh McGregor Ross tells the story of perhaps the first British computer to be designed for quantity production and to be marketed to the private sector. As such Pegasus is important. The author justifies his use of "Seminal" with ease.

It is not a book for the non-expert, though anybody reading *Resurrection* will be untroubled by that. The machine, the team that built it and the internal politics within Ferranti are all covered in detail. Much is made of its facility for "self-checking" in an era when hardware reliability was shockingly poor. Ferranti's disastrous agreement with Powers-Samas comes in for much criticism as does Ferranti's unfortunate seven month-long moratorium on selling Pegasus in the marketplace which is blamed for halving likely sales from 80 to 40 machines.

The main body of the book is divided into 32 short and fairly self-contained chapters. This causes a degree of repetition which some readers may find mildly irritating. For me, the most interesting parts of the book are those dealing with the predecessors and descendants of Pegasus: a broad sweep from the Elliott Nicholas to the ICT 1900 via several others, not all of which got past the proposal stage.

The last third of the book is mainly devoted to reproductions of contemporary documents.

On the downside, for a book of this nature, the absence of an index is to be regretted and it has to be said that it might have benefited from more rigorous proof reading but that may just be the prejudice of a grumpy old editor. That said, at £9.95 for 200 pages you can't really complain, can you?



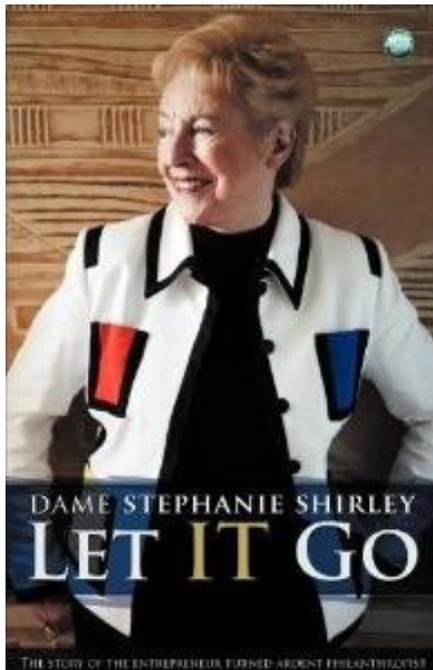
---

## Book Review : Let IT Go

*Hamish Carmichael*

---

Dame Steve Shirley, an important figure in the computing world for the past 50 years, has published a most interesting and enlightening memoir. It starts with her arriving in distress at Liverpool Street, aged five, on a Kindertransport from Austria just before the outbreak of World War Two. Then with her foster family came the gradual realisation: people have saved my life, so I'd better make sure it is a life worth saving. She tells how this led to her determination to do well at school, and her delight in mathematics. She was introduced to the possibilities of computing through her first job, with the Post Office Research Station at Dollis Hill, where she worked alongside some of the people who had done great things at Bletchley Park (though of course they never said a word about them).



Then came the foundation of F International, which started out as a means of allowing female programmers to work from home in their own time rather than being cooped up in an office and tied to a rigid daily timetable. It blossomed into one of the most serious and responsible software houses in UK, based on a principle of employee co-ownership, rather like John Lewis.

When she retired from involvement in active management, the remarkable growth and success of the FI Group led to her founder's shareholding making her fabulously wealthy. This facilitated a further career as a thoughtful philanthropist, mindful of Carnegie's principle: "a person who dies rich dies a failure". There followed honours and honorary degrees a-plenty. And in the background of all this is the tragic story of the autism of her only child, a most beloved son who died of a seizure at the age of 35.

It's a wonderful story, and very well told. I heartily recommend it.

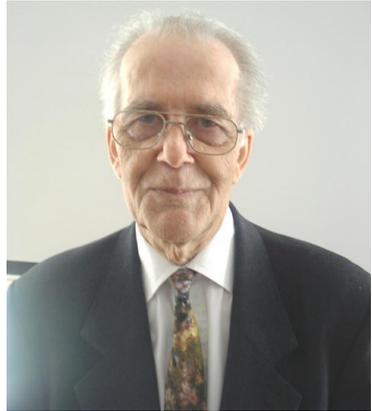
---

## **Obituary: Brian Wynne Oakley**

*Martin Campbell-Kelly*

---

After serving as a subaltern with the Royal Signals in World War 2, Brian was given a government scholarship to study physics at Exeter College, Oxford University. In 1950 he joined the Telecommunications Research Establishment (TRE). There he used the pioneering TREAC computer and its successors, and undertook research in telecommunications and civilian applications of military research. Beyond TRE, with his wife Marian, he was a keen actor and director in the Malvern amateur dramatics scene.



In 1969 Brian transferred to the Wilson Government's Ministry of Technology where he helped formulate IT industry and research policy. He successfully lobbied for the creation of a Minister of IT in the Thatcher Government. After serving as secretary of the Science and Engineering Research Council, he was appointed director of the Alvey Programme, 1983-87. The £300 million Alvey Programme was the UK's response to the Japanese Fifth Generation Project. In 1990 he wrote, with the science journalist Kenneth Owen, an important historical account of the programme *Alvey: Britain's Strategic Computing Initiative*. He was next appointed director of the follow-on ESPRIT programme. His diplomatic and administrative skills earned him huge respect in both Whitehall and academe. He was appointed CBE, and awarded honorary doctorates by two universities. He served as president of the BCS 1988-89.

In retirement he remained deeply interested in research and industrial policy. He was chairman of Logica Cambridge Ltd and a director of the European Initiative for Quantum Computing. He became an authority on the history of cryptography. In 1991 he learned that BT was planning to dispose of its Bletchley Park site for housing development and was a moving force, with Tony Sale, in establishing the Bletchley Park Trust. He served both as a director of the trust and chairman of the CCS from 1996-2000.

Brian was a popular conference chairman -- always affable, humorous, and very well informed. He last officiated at the ACE 2012 Alan Turing Centenary Conference at King's College, Cambridge, in June. He is survived by Marian, four children, and 10 grandchildren.

---

## Forthcoming Events

---

### London Seminar Programme

15 Nov 2012	History of Machine Translation	John Hitchens
13 Dec 2012	Film Show	Kevin Murrell, Dan Hayton & Roger Johnson
17 Jan 2013	The Bracknell Ferranti Computers	Peter Niblett
21 Feb 2013	A History of Information Retrieval	Keith van Rijsbergen
21 Mar 2013	The Atlas Story: 1956 to 1976	Simon Lavington et al
18 Apr 2013	EDSAC Replica	David Hartley & Andrew Herbert
16 May 2013	Babbage's Analytical Engine	Doron Swade

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](mailto:r.johnson@bcs.org.uk), or by post to Roger at Birkbeck College, Malet Street, London WC1E 7HX.

### Manchester Seminar Programme

20 Nov 2012	Advances Made by the Manchester Atlas Project	Dai Edwards
Jan 15 2013	Andrew Booth — Britain's Other Fourth Man	Roger Johnson
Feb 19 2013	The Incredible Shrinking Bit — Challenges in Computer Memory over the last 60 years and beyond	Steve Hill
Mar 19 2013	Computing Before Computers — From Counting Board to Slide Rule	David Eglin

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](mailto: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](http://www.computerconservationsociety.org/lecture.htm). Details are also published at in the events calendar at [www.bcs.org](http://www.bcs.org) and in the events diary columns of *Computing* and *Computer Weekly*.

## 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](http://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](http://www.bletchleypark.org.uk) to check details of times admission charges and special events.

**The National Museum of Computing : Thursday and Saturdays 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](http://www.tnmoc.org) for more details.

**Science Museum** : Pegasus "in steam" days have been suspended for the time being. Please refer to the society website for updates. Admission is free. See [www.sciencemuseum.org.uk](http://www.sciencemuseum.org.uk) for more details.

**Other Museums** : At [www.computerconservationsociety.org/museums.htm](http://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](mailto: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](mailto:kevin.murrell@tnmoc.org), or by post to 25 Comet Close, Ash Vale, Aldershot, Hants GU12 5SG.

### North West Group contact details

Chairman	Tom Hinchliffe:	Tel: 01663 765040.
Email:	<a href="mailto:tah25@btinternet.com">tah25@btinternet.com</a>	
Secretary	Gordon Adshead	Tel: 01625 549770.
Email:	<a href="mailto:gordon@adshead.com">gordon@adshead.com</a>	

---

## 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

### Project Leaders

SSEM: **Chris Burton CEng FIEE FBCS:**

cpb@envex.demon.co.uk

Bombe: **John Harper Hon FBCS CEng MIEE:**

bombe@jharper.demon.co.uk

Elliott: **Terry Froggatt CEng MBCS:**

ccs@tjf.org.uk

Ferranti Pegasus: **Len Hewitt MBCS:**

leonard.hewitt@ntlworld.com

Software Conservation: **Dr Dave Holdsworth CEng Hon FBCS:**

ecdhd@leeds.ac.uk

Elliott 401 & ICT 1301: **Rod Brown:**

sayhi-torod@shedlandz.co.uk

Harwell Dekatron Computer: **Johan Iversen:**

jo891979@talktalk.net

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: **Delwyn Holroyd:**

delwyn@dsl.pipex.com

Analytical Engine: **Dr Doron Swade MBE FBCS:**

doron.swade@blueyonder.co.uk

EDSAC: **Dr Andrew Herbert OBE FEng FBCS:**

andrew@herbertfamily.org.uk

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 the BCS however, need only notify their change of address to the 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

© Computer Conservation Society