

Issue Number 81

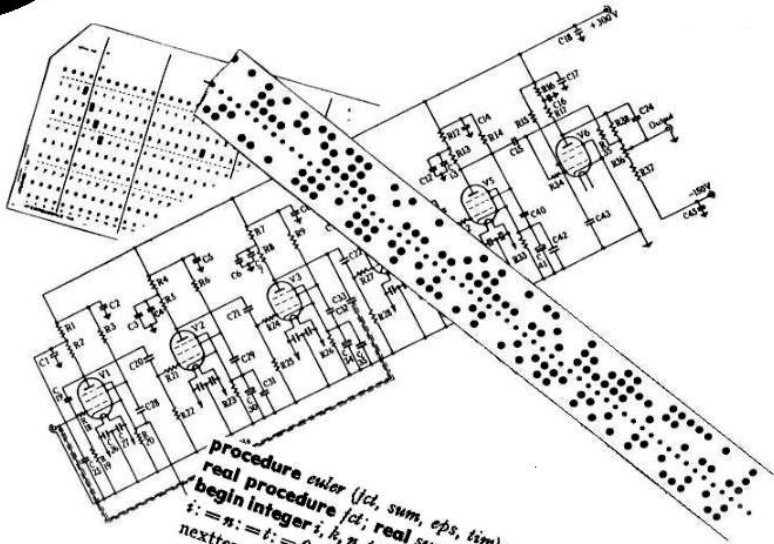
Spring 2018

bcs

The
Chartered
Institute
for IT

RESURRECTION

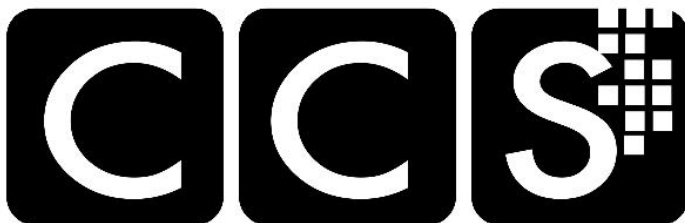
The Journal of the Computer Conservation Society

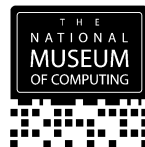


```

procedure euler (fcn, sum, eps, tim); value eps, tim; integer tim;
begin procedure fcn; real sum, eps;
begin integer i, k, n, t; array m[0:15]; real mn, mp, ds;
i := n; t := 0; m[0] := fcn(0); sum := m[0]/2;
nextterm: i := i + 1; mn := fcn(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]) ^ (n < 15) then
    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 (MSI) 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 Journal of the Computer Conservation Society

ISSN 0958-7403

Number 81 Spring 2018 Contents

Society Activity	2
News Round-Up	9
North West Group <i>David Morriss</i>	11
Language H: An Informal Overview (Part 1) <i>Andrew Herbert</i>	12
Emulation, Simulation and Replicas — Bringing New Life and new Users to the PDP-8 <i>Kevin Murrell</i>	20
A Life in Computing <i>John Dennis</i>	28
<u>Correspondence</u> <i>Ann Moffatt and Others</i>	30
50 Years AgoFrom the Pages of Computer Weekly <i>Brian Aldous</i>	32
Forthcoming Events	35

Society Activity

EDSAC Replica — *Andrew Herbert*

The good news is that we now have a working Counter once more, and a reliable Trig0 signal.

Work has now moved on to checking the interface between the sequence control tank (i.e., the program counter) and the coincidence unit so that main control can fetch the next order from store. Following on from this we will need to check out coincidence on each order's operand address so that orders can read or write their operands. At that point we will be able to test the interface between main control and the computer (i.e., arithmetic unit). We know that order decoding and arithmetic work in standalone mode, so we are hopeful that we will fairly quickly then be able to demonstrate an "EndP" pulse coming back from the computer to tell main control to increment the sequence control tank. Then we can start to run test programs to verify the correct implementation of the entire order code.

We have yet to commission Input, Output and Initial Orders — these remain in development. For testing purposes we are using a "Signal Sequence Injector" developed by Tom Toth to prime either the order tank in main control with test patterns or (main) store tank 0 as required.

Progress is being made on other fronts. All of the boxes for the main store delay lines have been delivered by the manufacturer and construction of the units has begun. Cabling between the storage regeneration chassis and the delay line "coffins" is substantially complete. A short video describing the machine and the project has been made and is available on the TNMoC website: www.tnmoc.org/news/edsac/edsac-reflecting-vision. We have installed a touch screen and large monitor at the front of the EDSAC gallery to show videos about the original EDSAC, the EDSAC pioneers and the reconstruction project.

Analytical Engine — *Doron Swade*

Work continues compiling the searchable database for Babbage's manuscript Notebooks. This work is being done by Tim Robinson in the US. Until now it has been impossible for Babbage scholars to come to definitive conclusions about aspects of the Analytical Engine design because of uncertainty as to what the Notebooks contained: we could not know whether what had already been researched was all that Babbage had to say on any particular topic, nor could we assess the degree of

completeness of the designs in ignorance of what else there might be in the some two dozen volumes of his manuscript Notebooks. Digging in the final cracks has been rewarding though we have resisted spending too much time interpreting the content given that the major immediate priority is data capture and cross-indexing.

The question of the levels of completeness of the various designs is critical to a prospective build and new clues have emerged from the recent work. A manuscript in the Cambridge University Library contains a disconcerting observation by Babbage: that "when some great improvement arose I only worked out enough to satisfy myself of its truth. I reserved the enquiry into many of its consequences as a treat when I otherwise felt indisposed to work" (1860). This has bleak implications for a definitive detailed design. Tim Robinson's mining of the Notebooks revealed an entry in which Babbage refers to Plan 13 as the "most complete ever made" (1849). So we now have a datum set by Babbage himself by which to judge the best expectation of completeness and this promises to give us a first approximation of the size and nature of design gaps we might need to fill in the specification of a meaningfully buildable machine. Progress on the database is slightly behind schedule with completion now estimated a few months hence.

Harwell Dekatron/WITCH — *Delwyn Holroyd*

The light bulbs in the power supply valve anode circuits have been removed and re-tightened after one worked loose. The relative brightness of the bulbs shows that the two parallel EL360 valves in each of the two main series regulators are sharing the load equally.

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.

Members who move house or change email address should notify Membership Secretary Dave Goodwin dave.goodwin@gmail.com of their new address or go to

www.computerconservationsociety.org/membership/membership_general.htm.

Those who are also members of BCS, however, need only notify their change of address to BCS, separate notification to the CCS being unnecessary.

Queries about all other CCS matters should be addressed to the Secretary, Roger Johnson at r.johnson@bcs.org.uk, or by post to 9 Chipstead Park Close, Sevenoaks, TN13 2SJ.

Elliott 903 — Terry Froggatt

I'm sad to report that Oliver Harlow, who donated the Elliott 903 to TNMoC, died on New Year's eve, aged 56.

Elliott Brothers of Borehamwood had given the 903 to nearby Aldenham school, where Olly used



it to run Algol 60 programs for his examinations in 1979. When the school switched to BBC computers, the 903 was destined for the tip, so Olly offered it a home. I met Olly at Elliott Computer Users Association meetings, in York in 1982 and Oxford in 1984. Then, in 2000, I received an email from him saying that he was considering donating his 903 to TNMoC. For various reasons it didn't happen at that time. But by November 2010, the 903 was suffering badly from being on a wet carpet under a leaking flat roof, and it was duly transported in four estate car loads, complete with software and a stock of blank tape, from Olly's home to TNMoC. By the end of March 2011, we (Olly, Peter Williamson & myself) had a working 8K CPU, and by September we had added the extra 8K store.

With the 903 working and regularly displayed, Olly turned to other tasks at TNMoC. He sorted out the TeleTypes, attaching one to the 903. He sorted out the stocks of spares and components (especially valves) held around TNMoC. He became the volunteers association treasurer, accessions team lead, co-ordinator of the 8-bit home computer and home-gaming systems (especially his favourites, the Ataris and Commodores), keen supporter of robotics projects for Bytes festivals, often creating his own mini-robots, and the logistical brains behind the acquisition of the Royal Navy Combined Tactical Trainer simulation system. Always helpful, he could turn his hand to most things (although he told me that "he didn't do analogue"). But the relationship was mutually beneficial: TNMoC gave Olly a home-from-home and a purpose to life, which I think helped him through his frequent illnesses. He will be missed.



The museum has seen a busy month repairing damaged Oslo machines and doing some major reorganisation. The 416 tabulator is back on four wheels and the 550 interpreter now has a prosthetic leg to take the weight while the cast iron original is superglued in place. One of the two master clocks is now back to working order and is on display in the "C R" origins room. This now only needs minor tweaks to the pendulum length to regulate timekeeping. The clock only lost about 1½ minutes over the Christmas period.

We have now decided how best to display the System/370 console, which will go out in the next week or two. The rear is nearly as interesting as the front, so we've done a poster size photograph to sit next to it. Hanging from the ceiling could allow 360 degree viewing, but at around 100Kg we thought that maybe might not be best solution.

We have replaced the top shelf of many of the two storey display tables with two narrower shelves. The personal computer room has been reorganised into timeline format, starting with System/23 and the IBM PC. Some PS/2s have been removed and a selection of ThinkPads moved from their previous location in their place.

The old video control room now houses the ASCII terminals, token ring / network workstation and point of sale in place of the ThinkPads. The AS/400 display now occupies the previous RISC room, with the RISC machines displayed alongside PCs. The space freed up by the AS/400s has allowed us to dedicate most of that room now to Office Products, with Copier 1, typewriters, word processors, dictating machines and DisplayWriter.



Rear view of the System/370 operator console

This console is from the S/370 Model 135. Its smaller brother, the Model 135, was developed in Hurley.

The Model 135 was introduced in 1971, with four main storage sizes from 96,000 to 240,000 bytes, four times that of the S/280-30. The Model 135 pioneered the use of erasable microcode, using the new 9" floppy disk code named OSAR.

The System/370 range used monolithic logic and memory technology, giving it a high internal operating speed from 275 to 1,400 nanoseconds per micro-instruction.

OS/VS compatibility ensured customers could continue to run their S/360 applications.

Software — *David Holdsworth*

There has been quite a bit of activity during 2017.

David Huxtable and I have made strides with Kidsgrove Algol. You can run the compiler on-line at kdf9.settle.dtdns.net. The server is very modest, so please do not try mining bitcoins.

If you want to experiment running it on your own machine there is info here: kdf9.settle.dtdns.net/KDF9/kalgol/DavidHo/nonout70.htm.

Bill Findlay has scanned the Egdon Manuals (very heroic of him — for which thanks) www.findlayw.plus.com/KDF9/Documents/?C=M;O=D.

I have nearly finished scanning the Service Routine Library Manual — kdf9.settle.dtdns.net/KDF9/SRLM.

We have failed to trace any other copies of these manuals, and so they have been scanned page by page on flatbed scanners to avoid possible page wrecks.

Brian Wichmann is working on running his test suite, but now that we know we do not have a finished production version of kAlgol or Usercode this exercise loses some of its impact.

I have a new year resolution to work on presentation in 2018, to which end I have already contacted Dik and Simon.

I intend to transfer stuff to more convincing URLs than settle.dtdns.net.

Also, I would like to get an online capability for Elliott Algol and KDF9 assembly languages. I have hopes of capturing some interest from modern school kids as IT teaching once again includes coding.

If you know Algol 60, do please try kdf9.settle.dtdns.net

ICT/ICL 1900 - *Delwyn Holroyd, Brian Spoor, Bill Gallagher*

Since the last report all of our efforts have been on the 1901A sub-project, to get this to a natural break-point; so that we can concentrate on polishing the 1904S emulator ready for release.

ICL 1901A

The 1901A emulator is now running both E1HS and the fixed part of E1DS successfully.

We are still missing the second paper tape for E1DS (the overlays), but have recreated more of the missing overlays, so that we can now use magnetic tape in addition to basic peripherals (cards, paper tape and

printer). We have started looking at discs, but the UDAS implementation, with the little information we have, is proving difficult.

We can compile/consolidate PLAN programs under E1HS (#XPLC/#XPCC) in a basic peripheral environment, while under E1DS we have used magnetic tape based compilers to compile PLAN (#XPLH), COBOL (#XEKB) and Algol (#XALM) programs.

We have also written and run a small COBOL-based suite of programs:-

- Create initial MT master file
- Validate card input and write to unsorted update MT
- MT sort of validated data (#XSME)
- Classic 3 tape master file update (Old, New & Updates)
- Print master file

Again, the same plea as the last time - Has anybody any information, listings etc. of Stevenage Executives – we would welcome it.

ICL 2966 — *Delwyn Holroyd*

One Saturday in December the machine crashed whilst running George 3 and could not be restarted. The symptoms were a hang during CME microcode loading, but not always at the same point. In further testing it was found that the engineers' loader also crashed after a short time.

Normally when faults occur the first recourse is the CUTS diagnostic test programs, but in this case all tests passed except for the store test MHR7 which has been failing for some years due to a Hamming logic fault on one of the sub-store control boards. We hadn't corrected this fault because it didn't cause any problems during normal running, and we are down to our last spare sub-store control board (there are four in the machine).

Following some more diagnostic work it became clear the crashes were due to a timed out SCU transaction to the store. The spare sub-store control board was fitted to allow the full store test to be run, but it so happened that this also fixed the issue! Evidently this board has developed a second fault. Unfortunately we now have no spares, so the task of constructing a test jig for diagnosing faults on the more complex boards has got a bit more urgent. The existing test jig only drives half the board pins, which was sufficient for a lot of the DCU boards but not for the SCU.

Meanwhile the intermittent OCP -5.2V problem has continued to occur. Before Christmas we narrowed down possible causes as a bogus under-voltage comparator trip, a control board logic fault, or a bogus signal from one of the opto-isolators involved in over-voltage detection. The former two were then ruled out by board swapping, leaving only what was

considered to be the most unlikely cause! An actual over-voltage signal from one of the power supplies was thought to be impossible because it latches in the supply and thus the over-voltage LED would remain lit after the OCP control circuits shut off all the supplies.

To resolve the issue once and for all the four channel scope was wheeled out and connected to the -5.2V and 5V supplies, the under-voltage comparator output and the wire OR of the over-voltage opto-isolator outputs. After an hour the supplies tripped off and the scope triggered, showing a large spike on the over-voltage line. It was clear from the other traces that no over-voltage had actually occurred.

Three scope channels were then moved to the other side of the opto-isolators for each of the three parallel -5.2V power supply bricks. After another hour or so the scope triggered again and this time showed a clear spike on the over-voltage line coming from one of the supplies.

So it appears that either the signal is actually coming from the supply, there is a wiring fault somewhere, or there is an intermittent internal short in the opto-isolator in question. The suspect supply was swapped with the spare (after suitable running in of the spare on the bench) to return the machine to operation, and then attention was turned to testing the suspect supply on the bench. The spike didn't occur in an hour or so of running, and the over-voltage detection and latch appeared to work just fine.

Bombe Rebuild — *John Harper*

As reported some time ago the Bombe Rebuild website was lost to readers when the organisation that was hosting it withdrew its facilities at short notice. The original has now been updated and incorporated into a new site with much wider coverage: not just as before the ten year or so rebuild phase.

The plan is that the site will be expanded over time with new Bombe related subjects added so viewers might see place holders appear from time to time or only partial subjects be covered as a topic is developed. For a while "page not found" might messages appear. Hopefully, these will be short-lived.

The site is at *bombe.org.uk*.

If any major errors are noted, please let me know.

News Round-Up

Following the Meccano D.A. article in *Resurrection 80*, reader Roger Bolton has reported that “a few years later John Barton, working at the University of the West Indies, made a Meccano computer in which certain of its motions were controlled by pulses from a Geiger counter, making a sort of Monte-Carlo integrating engine “. Any more?

101010101

After 191 years, Belgium has finally got around to abolishing its telegram system. Latterly used by a tiny number of organisations, largely for legal reasons, it was switched off in December. The UK took this step in 1982, the USA in 2006. But a surprising number of national telegram systems remain in use including in New Zealand which abolished its service in 1999 only to reinstate it four years later. Whether the telegrams are still delivered by a callow youth on a low-powered motor cycle is not known.

Legend has it that the late Cary Grant once received a telegram enquiry “HOW OLD CARY GRANT” to which he replied “OLD CARY GRANT FINE HOW YOU”. This story is thought to be a fiction invented by Mr Grant.

101010101

It is our unhappy duty to report the passing of a distinguished CCS member and software pioneer, Mary Lee Berners-Lee. As Mary Lee Woods she was one of Ferranti’s very earliest recruits to work on software for the Mark 1 computer. She led a successful pioneering campaign for equal pay when it was discovered that Ferranti paid women less than men for the same work. After marrying Conway Berners-Lee she became Mary Berners-Lee Squared according to the Ferranti wags.

As the mother of Sir Tim B-L she claimed to be “Grandmother of the Web”.

101010101

On page 29 Ann Moffatt writes concerning the early days of computing exhibits at the Science Museum in London. But she also told us that in 1975 the Museum’s Pilot ACE computer of 1951 was working well enough to add two numbers together. It seems unlikely that power has been applied from that day to this. What a glorious achievement it would be if ...

101010101

Exciting news from Bletchley Park! Agreement has been reached to transfer the Turing-Welchman Bombe Rebuild from the premises of the Bletchley Park Trust to those of the National Museum of Computing, still within the Bletchley Park estate.



John Harper, leader of the Bombe Rebuild team said "After careful consideration of the options, The Bombe Trustees approached TMNoC, which agreed to host the Bombe exhibit. We are delighted with this solution and welcome the opportunity to remain part of the overall visitor attraction at Bletchley Park. Our team of volunteers is looking forward to continuing to demonstrate how the Bombes made their vital contribution to Bletchley Park's wartime role in the new venue. We thank the Bletchley Park Trust for their co-operation over the years and are pleased that the story of the Bombe will remain very much part of the story that it tells."

Andrew Herbert, TNMoC chair responded "To house the reconstructed Bombe close to the Colossus Rebuild makes a lot of sense from many perspectives. As a pre-computing electro-mechanical device, the Bombe will help our visitors better understand the beginnings of computing and the general thought processes that led to the development of Colossus and subsequent computers. The story of the design of the Bombe by Alan Turing, the father of computer science, leads very appropriately into the eight decades of computing that we curate. Even the manufacture of the Bombes leads directly to British computing history — the originals were built by the British Tabulating Machine company (BTM) in Letchworth, which later became part of ICT, then ICL and now Fujitsu"

The Bombe will be housed near Colossus in a new gallery. A crowdfunding campaign can be found via the TNMoC website www.tnmoc.org.

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.

At www.computerconservationsociety.org/software/software-index.htm can be found emulators for historic machines together with associated software and related documents all of which may be downloaded.

North West Group

David Morriss

At the end of the current season of events in Manchester, Tom Hinchliffe and Gordon Adshead will be stepping down as Chair and Secretary (respectively) of the North West Group of the Society.

Tom has been Chair for an outstanding 18 years taking over the job from the late Prof. Frank Sumner in 2000. During a distinguished career with ICL he rose to become head of the Mainframe Development Division responsible for all development aspects of ICL's flagship large computers both hardware and software. He has been Chair of Council and Pro-Chancellor of the University of Manchester Institute of Science and Technology (UMIST) and was a director of the Museum of Science and Industry in Manchester (MSI). Quite how he found time to support the CCS is something of a mystery.

Gordon assumed the position of secretary in 2010 following a career in ICL during which he built up the Design Automation Division providing the tools needed to support the design and manufacture of several generations of Mainframes.

The Society owes these two gentlemen a huge debt. Not just for keeping the show on the road, but for managing and developing a vibrant and active group over a long period. Most of all perhaps, for the calm and resourceful way in which the group's involuntary exit from MOSI was overcome and a new venue was found without any disturbance to the group's programme of events.

Happily, volunteers have been found to fill the vacant posts. We welcome Bob Geatrell and Alan Pickwick as the new Chair and Secretary who, by the time this edition of *Resurrection* is published, will be taking up the reins.

North West Group contact details

Chair	Bob Geatrell:	Tel: 01457-868700.
Email:	<i>bob@geatrell.co.uk</i>	
Secretary	Alan Pickwick:	Tel: 0161 973 6796.
Email:	<i>alan_c_pickwick@btinternet.com</i>	

Language H: An Informal Overview (Part 1)

Andrew Herbert

Simon Lavington has written that Language H was a “business processing language ‘designed to permit COBOL style operations in a simpler language, and suitable for running on smaller machines.’ The first language H compiler for the NCR/Elliott 405M ‘was completed in August 1961 and checked out to be effective by December 1961’. ‘January 1962 saw the start of a project to develop a language H compiler that ran on the 405M but produced object code for an NCR-Elliott 803B.’ In time, Language H compilers were produced for the Elliott 803 and the 4120/4130 and the NCR 315.”

In 2014 Mr Tony Gough presented me with a box of Elliott documentation dating back to the time he had been employed by Elliotts at Borehamwood. In the collection was a Language H (1962) Manual and a Language H Facts Book published by NCR (The National Cash Register Company Limited). The manual describes a version of the language for the NCR-Elliott 803, succeeding earlier versions for the 405M (1960, 1961) and a 405M to 803B cross compiler (1961).

Language H is a cousin to COBOL. Program commands similar to COBOL are written in a stylised English form intended to make programs readable by non-programmers. A significant difference is that Language H does not have a direct equivalent of the COBOL Data Division to describe the structure of input data, output data and working storage, but instead derives this information from how data is used in program commands.

Design Philosophy

The Language H manual sets out the design philosophy of H:

“...a simple language in which many data processing problems can be expressed”;

“...simplicity of programming...a layman’s language having comparatively few rules and providing flexibility of expression”;

“...the smallest number of effective phrases should be provided”;

“...the strain on the memory of the user should be kept as low as possible”;

“...any program should be unambiguous”;

“...the structure of the language should be independent of any class of computer”;

"...the user should have close control over the operation of the program and the positioning of the data";

"...the basic unit of information handled should be the size of a number, a name or short comment; only exceptionally need it be as small as a single character or as large as a record";

"...the wealth of possible expressions ... has been deliberately reduced in order to make learning and accurate use easier."

The author leaves it to readers to make their own judgement as to how successful the language H designers were in meeting these goals.

Arithmetic Commands

Like COBOL, H has arithmetic commands such as:

H Command	Effect
ADD a TO b	$(b)' := (b) + (a)$
ADD a TO b AND c ...	$(b)' := (b) + (a);$ $(c)' := (c) + (a)$
SUBTRACT a FROM b	$(b)' := (b) - (a)$
SUBTRACT a FROM b AND c ..	$(b)' := (b) - (a);$ $(c)' := (c) - (a)$
MULTIPLY a AND b BY z	$(a)' := (a) \times (z);$ $(b)' := (b) \times (z)$
DIVIDE a BY z	$(a)' := (a) \div (z)$ - integer division ¹

Each arithmetic command implicitly sets an "auxiliary operand," denoted by '@' which can be referenced by subsequent commands, as in:

ADD a TO b	$(@)' := (b)' := (b) + (a)$
ADD c	$(@)' := (@) + (c)$

¹ (**x**)' stands for the value of **x** after the command is executed, (**x**) stands for contents of **x** before the command is executed, where **x** is an "operand". The destination operand can be a labelled operand (i.e. variable) or an indirectly specified operand (i.e. array element). The source operands can be labelled operands, indirectly specified operands or literals. These are explained later in the text.

The auxiliary operand can be set:

```
TAKE a (a)' := (a)
```

and interrogated:

```
LOAD a (a)' := (a)
```

Multiple assignment is possible:

```
LOAD a AND b WITH z (a)' := (a)' := (b)' := (z)
```

This repertoire of arithmetic functions is extended by the **CALCULATE** command, as in:

```
CALCULATE a (FUNCTION n) FROM b AND c
```

Where 'n' denotes the specific function to be applied, so that, for example,

```
CALCULATE a (FUNCTION 11) FROM b AND c
```

computes

```
(a)' := (b) × 10 (c).
```

and

```
CALCULATE a AND b (FUNCTION 24) FROM c AND d
```

computes

```
(a)' := quotient of (c ÷ d)
```

```
(b)' := remainder of (c ÷ d)
```

The manual lists some 45 functions including rounded and unrounded division, remaindering and percentages.

CALCULATE is also used for text processing:

CALCULATE a (FUNCTION 40) FROM b AND c — extracts the (b)th character from the alphanumeric string held in (c) and stores it as a numeric value in (a') (a six-bit character code is assumed). Function 41 extracts the character as an alphanumeric operand and functions 42 and 43 allow a character to be replaced by a numeric code or alphanumeric value respectively.

BRANCH TO a OR b is in effect a computed **GOTO**. If (**x**) is out of range control passes to the next statement.
OR c OR ...
ACCORDING TO x

HALT stops the program.

WAIT pause the program awaiting operator intervention via a "**WAIT**" switch on the computer console. The action required of the operator is as follows: if the **WAIT** switch is **ON** it must be turned **OFF**; if the wait switch is **OFF** it must be turned **ON** and then **OFF** again. (This is to force the initial **WAIT** in a program pause.)

There is an additional **DO SUBROUTINE** command concerned with calling machine code subroutines that falls beyond the scope of this article.

Conditions following an **IF** can be arithmetical, or else reference flag and switch/or operands. Switches represent the state of readable switches on the operator's console; flags are Boolean variables.

IF b IS GREATER THAN a also **EQUAL TO, LESS THAN, NOT GREATER THAN, NOT EQUAL TO, NOT LESS THAN**

IF LESS THAN b i.e. if $(\theta) < (b)$

IF a AND b ARE POSITIVE also **ZERO, NEGATIVE, NOT POSITIVE, NOT ZERO, NOT NEGATIVE**
IF a OR b OR c IS NOT POSITIVE

IF SWITCH 1 AND SWITCH 2 ARE ON
IF SWITCH 5 OR SWITCH 6 ARE OFF
IF a FLAG IS ON
IF a FLAG AND b FLAG ARE OFF
IF c FLAG OR d FLAG IS OFF
IF a FLAG AND SWITCH 1 ARE ON

Flags are set using the **TURN** command:

TURN ON a FLAG AND b FLAG

TURN OFF c FLAG IF ...

TURN OVER d FLAG IF ... flip the state of *d* if the condition holds.

Operands

Most commands take parameters, which H calls “operands”. The simplest operands are literals. These can be numeric: e.g., **1234**, **-99** (but not **+99**); sterling, e.g., **£20/19/11**, **-£0.0.2** (and the stored value is held as pence), or alphanumeric as in **'ABCD012'**.

There are no facilities for floating point numbers.

The equivalent to variables in other languages are described as “labelled operands” in H. They have alphabetic names and can be made up of multiple words. Strictly, in H, variables are called “labelled operands”. Moreover, all names whether used as variables or as labels to tag statements in programs are examples of “labels”. Labels must be distinct from any of the 81 reserved words in the syntax of H (e.g., command names like **ADD** or **BRANCH**).² The same name can be used as a variable and to label a statement since it is obvious from the context (flow command or otherwise) which use is intended: indeed, by the same means, a given label can be used to name operands of different types (numeric, alphanumeric, flag) and the compiler will work out which operand instance to use in each command where the name is referenced using on the implied type.

The first use of an alphanumeric variable must define its length:

name (UP TO m CHARACTERS) — *m* is the maximum number of six bit characters the operand name is required to hold.

The contents of an alphanumeric variable can be accessed and modified using the **CALCULATE** command.

² Internally an operand label is represented by a six character string, comprising the first four letters of its first word, followed by the first letter of its second word (if any), followed by the first letter of its third word (if any). Thus the label **ACCOUNT** is an invalid operand label since it clashes with the reserved word **ACCORDING**.

H also provides the equivalent of arrays in other languages. H calls the array elements "unlabelled operands". A one-dimensional array is specified thus:

**STOCK (AND UP TO n where n must be a numeric literal.
OPERANDS)**

This creates a vector of ($n+1$) elements, the first of which is a labelled operand named 'STOCK', followed in store by n unlabelled operands. The elements of the vector can be accessed indirectly using the **FOR** construct:

STOCK FOR 0 accesses (**STOCK**) i.e., 0th operand following **STOCK**

STOCK FOR 3 accesses the third unlabelled operand following **STOCK**

STOCK FOR PERIOD accesses the (**PERIOD**)th unlabelled operand where **PERIOD** is a numeric operand.

By default, memory is allocated to variables in the order in which they are textually encountered when compiling the program. Thus, suppose immediately before the first use of **STOCK** there was a first use of a variable named **DATE**, and following it a first use of a variable named **TOTAL**. Then **STOCK FOR -1** would access **DATE** and **STOCK FOR 1** would access **TOTAL**. By the same token **DATE FOR 1** would access **STOCK**.

The programmer can explicitly state how operands are to be laid out in store using a **NOTE** statement, as in:

**NOTE: CODE, CUSTOMER NAME (UP TO 29 CHARACTERS),
CUSTOMER ADDRESS (UP TO 59 CHARACTERS), CREDIT
RATING, WEEKLY PURCHASES (AND UP TO 51 OPERANDS)
AND TOTAL**

This defines a list of 57 operands, 51 of which are the unlabelled numeric operands following **WEEKLY PURCHASES**.

Subsequently in the same program, the statement:

FILE CODE (AND UP TO 56 OPERANDS) ON CHANNEL 1

would result in all the operands in the list following **NOTE** being filed on Channel 1 in the sequence in which they were named in the note. (This is an example of a using **NOTE** to create a "record specification". The commands for handling files and records will be explained later).

Usually **NOTE** statements are found at the head of a program, in any case the use of an operand label in a **NOTE** statement must precede the first use of that operand label in any other statement.

Putting these concepts together the following code adds up the individual total sales in each of 52 weeks of a year to provide the year's total sales:

```

NOTE:                SALES WEEK TOTAL (AND UP TO
                    51 OPERANDS)
                    . . . .
                    LOAD CUMULATIVE SALES WITH 0.
ACCUMULATE:         ADD SALES WEEK TOTAL FOR WEEK
                    TO CUMULATIVE SALES
                    REPEAT 51 TIMES FROM
                    ACCUMULATE COUNTING IN WEEK
                    . . . .

```

The **REPEAT** statement takes care of setting **WEEK** to 0 at the start, then loops around **ACCUMULATE** 51 additional times to sum the totals for 52 weeks numbered 0 to 51.

Vectors of alphanumeric values are specified thus:

name (UP TO *m* CHARACTERS) (AND UP TO *n* OPERANDS)

n+1 strings each up to *m* characters in length labelled name.

H permits up to five levels of indirection:

Indirect Operand Specification	Interpretation
a FOR b	a [b]
a FOR b FOR c	a [b[c]]
a FOR b FOR c FOR d	a [b [c]+ d]
a FOR b FOR c FOR d FOR e	a [b [c]+ d [e]]
a FOR b FOR c FOR d FOR e FOR f	a [b [c]+ d [e]+ f]

Andrew Herbert is a leading member of the Society, Chair of The National Museum of Computing and leader of the EDSAC Replica Project. He can be contacted at andrew@herbertfamily.org.uk. Part 2 of this article will appear in the next edition of Resurrection.

Emulation, Simulation and Replicas — Bringing New Life and new Users to the PDP-8

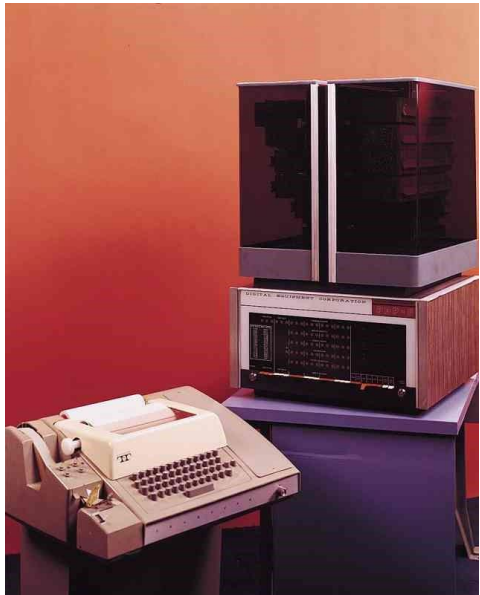
Kevin Murrell

Despite being designed more than 50 years ago, the PDP-8 is now attracting new programmers and users encouraged by a series of replicas and emulators. The question remains why in particular has the PDP-8 has sparked this interest?

A Little History

More than 50 years ago, the Digital Equipment Company (DEC) released the PDP-8 computer which quickly became the best-selling computer of its time. Contemporaneously, a devoted fan-base for the machine began, which continues today with new machines being built and development of software for the computer continuing.

The PDP-8 in 1965 was a transistorised desktop computer (albeit a sturdy desk is required) with 4k of 12-bit word core memory, an easily understandable order code, extensive



Desktop "Straight-8" with ASR33

documentation and practical system software. Often paired with an ASR33 teletype, the machine was personal in the sense it was affordable by a small team and operated by one person at a time but was also very self-contained. Editors, assemblers and linkers were available from the start and several high-level languages including BASIC, FOCAL and FORTRAN IV were readily available. Various operating systems were available, the most popular being OS/8, but the list also included a multi-user system called TSS-8 which supported up to 17 users. DECUS, the DEC user group, circulated regular bulletins with both hardware and software fixes and suggestions.

The PDP-8 used DEC's modularised "flip-chip" circuit boards, each about 3" by 7" which were standardised components already in wide use in other DEC computers. Each flip-chip had a specific function and typical examples included a flip-flop (12 of which could be linked to form a register), inverters and standard logic gates. To satisfy their initial investors Digital did not ostensibly sell computers in their early days but sold components such as flip-chips instead. Indeed, PDP stands for Parallel Data Processor — carefully avoiding the term computer as those early investors thought few would be willing to buy a computer from this small start-up company. By 1965 DEC had given up the pretence and were solely in the market selling computers, but the PDP name remained.



Two models from the DecMate series

of the PDP-8 known as the CMOS-8 which was used in the VT78 video terminal. The CMOS-8 was also used in a series of small personal computers known as the DecMate series. The new machines were mostly compatible with the previous generations and system software was often transferrable. However DEC regarded the machines as platforms for their new office software and discouraged their use as a general PDP-8. The DecMate series was manufactured until 1990. In all, the PDP-8 was reimplemented with new technology ten times over a period of 15 years.

The PDP-8 series was continually developed as semiconductor technology improved until the last recognisable machine, the PDP-8/A was released in 1974. The bestselling model was the PDP-8/E with a performance similar to an IBM 360 Model 40 — possibly the best performance/cost ratio ever achieved.

In 1976, under licence from DEC, Intersil produced a single-chip implementation

What's happening now

Software Emulation

In 1993, Bob Supnik, formerly an engineer at DEC, produced a general multi-system and multi-platform emulator called SIMH, which was based on his previous work dating back to the 1960s. SIMH supports many machines from different manufacturers but supports DEC machines, including the PDP-8, very well.

SIMH is a text based programme that lets the user define the machine they wish to emulate and build virtual devices and media packs. While real peripherals such as disc drives and tape drives were generally more affordable from DEC than those from the traditional mainframe suppliers, users were of course limited by space, funds and available power. There is no such limitation with virtual devices and it is easy to add dozens of RK05 disc drives to a virtual PDP-8. This occasionally brings out some shortcomings in the system generation scripts where the original programmer assumed no one would ever have more than eight disc drives!

As SIMH is designed to emulate many different systems, it does not attempt to simulate the target system from the logic level upwards, but rather emulating instruction by instruction.

The primary loop to emulate instructions is:

```
while (!halted) {
    if (interrupt_pending)
        handle_interrupt();
    form_current_program_counter();
    get_word_from_memory();
    increment_program_counter();
    decode_instruction();
    switch(instructionType) {
        case TAD: // Twos Complement add to Accumulator
            switch (addressingMode) {
                case IMMEDIATE: do_some_stuff();
                    break;
                case ZEROPAGE : do_some_different_stuff();
                    break;
                // more instruction types here
            }
    }
}
```


Once the basic structures of the processor have been defined, the action of each instruction type can be handled individually. Having used SIMH to build an emulator for a quite different processor it is possible to generate a HALT whenever a new unhandled instruction is found, and then keep adding code for each new instruction until the machine finally runs.

As an example of running the OS/8 operating system on a simulated PDP-8 using SIMH, the following shows the start-up of SIMH by attaching a disc image and then booting that image. The DIR command is entered, then the run command for the FORTRAN Run Time System (FRTS), then ADVENT to start the Adventure program.

```
$. /pdp8
Set cpu 32k
Set cpu idle
Attach rko advent.rk05
Boot rko

PDP-8 simulator V3.8-1

.DIR
DIRECT.SV 7          FRTS .SV 26          BASIC .WS 11
PIP .SV 11          PASS2 .SV 20         SET .SV 20
FOTP .SV 8          PASS20.SV 5         RTFLOP.SV 15
[... ]
BRTS .SV 15         LCSYS .BI 3         ADVENT.LD 145
F4 .SV 20           UCSYS .BI 3         LOAD .LS 8

162 FILES IN 2648 BLOCKS - 544 FREE BLOCKS

.R FRTS
*ADVENT
*$

WELCOME TO ADVENTURE!! WOULD YOU LIKE INSTRUCTIONS?

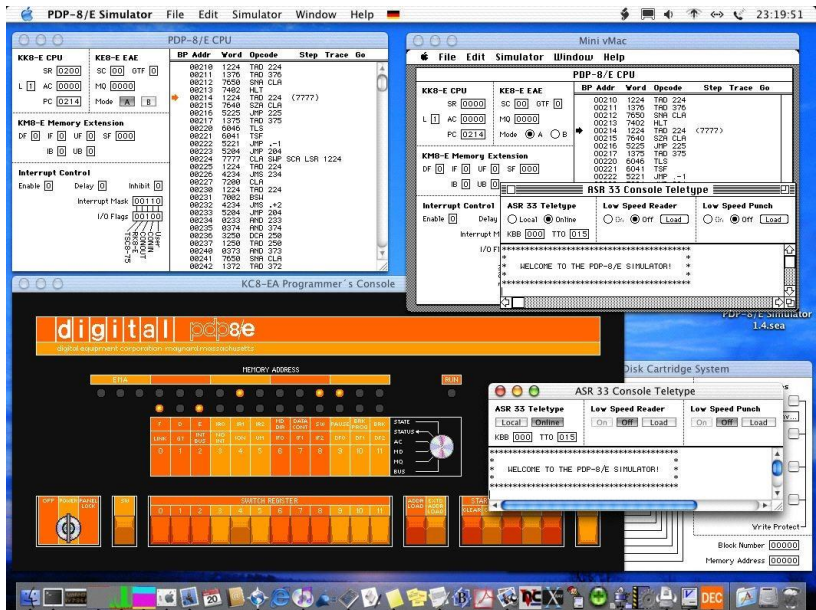
> NO
YOU ARE STANDING AT THE END OF A ROAD BEFORE A SMALL BRICK BUILDING.
AROUND YOU IS A FOREST. A SMALL STREAM FLOWS OUT OF THE BUILDING AND
DOWN A GULLY.

> ENTER
YOU ARE INSIDE A BUILDING, A WELL HOUSE FOR A LARGE SPRING.
```

There is little new under the sun of course, and in the 1970s DEC produced something similar for their own development teams: a PDP-11/60 emulating OS/8. The /60 was the only model in the series with a writeable microcode store that made this possible, however this system was only used internally at DEC.

While SIMH gives a very accurate emulation of the PDP-8, it does of course lack the visual appeal of a real system.

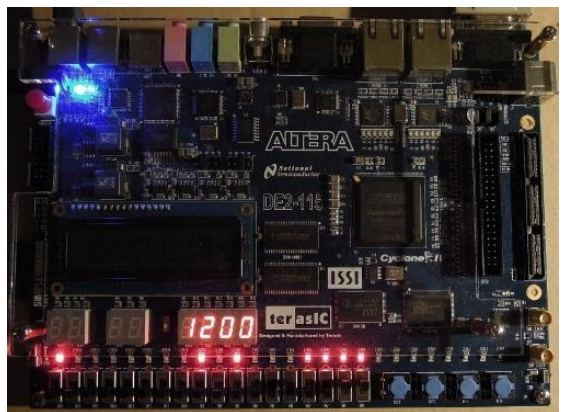
Bernhard Baehr has produced a GUI emulator that is shown below. As well as an interactive front panel, this emulator allows the internal processor state to be continually monitored. It supports visual hard disc drives and DecTapes that spin! The virtual teletype even makes the (quickly annoying) noise that an original ASR-33 makes!



Bernhard Baehr's MAC-based PDP-8 emulator

Hardware emulation

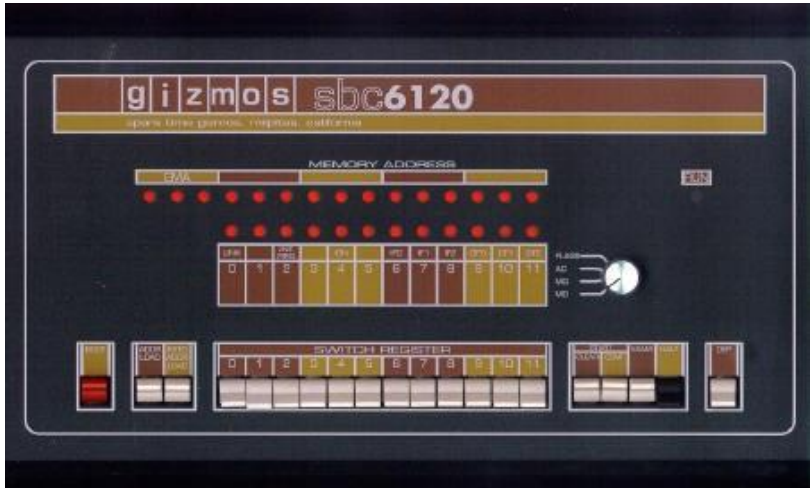
Several projects are either completed or underway to implement the PDP-8 using FPGA (Field Programmable Gate Array) technology. The FPGA chip is an array of programmable logic blocks which can be programmed as simple logic gates and flip-flops through to registers and memory arrays. The open-cores project includes a complete HDL (Hardware Description Language)



Altera DE2-115 prototyping board

definition of the PDP-8, which is typically downloaded to a development system that includes sufficient switches and lights to represent the PDP control panel. The latest version of this HDL emulates the complete range of PDP-8 machines including memory and a disc drive. One popular FPGA development system which has been used to emulate the PDP-8 is the Altera DE2-115 prototyping system which incidentally Bill Purvis has also used to emulate the logical structure of EDSAC.

Once connected to a serial terminal, this development system is quite indistinguishable from an original fully stacked PDP-8/E running OS/8.



SpareTimeGizmos 6120-based system

The single chip PDP-8 (the 6120 series of Harris microprocessors used in the DecMate machines) is still available from various second-hand sources, which led Bob Armstrong (of SpareTimeGizmos) to develop the SBC6120 — a new single board system using the original microprocessor but with new circuitry to support modern hard discs. The most popular disc drive for the PDP-8/E was an RK05 drive that had removable disc packs, each pack supporting 2M x 12bit words. A modern hard disc on the SBC6120 can be logically divided into hundreds of RK05 drives.

The single board computer was followed by a front panel that replicates the PDP-8/E front panel. The design accurately represents the original front panel, albeit only around 60% of the original size. The difficulty of obtaining original CMOS-8 microprocessors sadly limits the use of this machine.



Oscar Vermeulen's PiPDP-8

Most recently, Oscar Vermeulen in the Netherlands has taken the SIMH emulator further by designing a front panel complete with LEDs and switches that 'attaches' to the virtual PDP-8 and allows it to be operated from the switches and to have the lights represent what is happening inside the virtual machine. The new front panel is an accurate version of the PDP-8/L (albeit $\frac{2}{3}$ of the original size) which was the last PDP-8 model to show all the processor registers and state simultaneously.

In Vermeulen's machine, the SIMH emulation of the PDP-8 is run on a Raspberry Pi mounted on the back on the front panel, and with very few modifications to it to expose the register and processor state values. Additional code on the Pi reads those values and updates the front panel display. Given a sufficiently powerful variant of the Pi, the emulation also modulates the brightness of the LEDs to mimic the behaviour of the original incandescent lamps!

It is this new realisation of the PDP-8 that has generated the most interest in those new to the machine, partly because of the ease of building the kit and partly because of the very attractive design. (Vermeulen is now designing a new front panel replica of the PDP-11/70 — again using SIMH and a Raspberry Pi to do the heavy lifting.)

Why?

The question that remains is why is there all this activity and enthusiasm for a computer designed more than 50 years ago? Yes, nostalgia for a machine many of us used in our early days in computing, but why the new generation of enthusiasts for a machine that came out of production after many of them were born?

The PDP-8 is very well documented; DEC produced detailed descriptions of exactly how the machines worked and of course all the circuit diagrams, and both DEC and DECUS provided a wealth of program examples and useful routines as well as the source-code for all the utilities. All this information is now readily available on the web and includes scans of all the original DEC handbooks. Designing new emulators and hardware replicas is therefore made much easier with this information available.

Programming the PDP-8 is quickly learnt; the order code is uncomplicated, and for those not keen on assembly programming, high-level languages such as BASIC, FORTRAN and FOCAL are freely available and are well documented. Development of at least one operating system for the PDP-8 continues today and some of those new to the PDP-8 are already submitting changes and fixes.

Many of these new PDP-8 programmers have full-time jobs in IT, but are typically constrained by modern development frameworks and the continual need to learn new techniques and programming paradigms. The simpler world of the PDP-8 is an attractive alternative during their downtime. A recent posting from a young convert to the PDP-8 said "Wow, truly amazing how these were created with such limited knowledge at that time. It shows how inventive people were, something I think today's generation is losing." That probably makes most of us feel very old, but it also seems to assume that modern machines are somehow created by magic, and not by 'inventive' people at all. It does highlight the disconnect between users of modern off-the-shelf solutions and the research and technology that went into building them.

There is also the not-inconsiderable fact that the PDP-8 is visually interesting, with a full complement of flashing lights and switches — everything that you would expect from a computer from the 1960s. Having something to 'show off' when you have completed construction of the kit is of course important. Very highly specified machines are available off the shelf for anyone today, but are largely "black boxes" that discourage any modifications or investigation, while the PDP-8 remains a design that is readily understandable at all levels in both hardware and software, is easily configured, and remains endlessly tweakable.

Long may it continue!

Kevin Murrell is a trustee and co-founder of the National Museum of Computing and a former secretary of the CCS as well as being the current London Meetings Secretary. Contact at kevin.murrell@tnmoc.org.

A Life in Computing

John Dennis

During my national service I was trained as a radio engineer. At the end of two years, (in August 1959) as I prepared to leave the army I heard that my previous employer, Road Research Laboratory, was about to purchase an electronic computer. I was offered the job as one of the maintenance engineers. I had to buy a book to find out what an electronic computer was and this made me decide to take the opportunity.

Road Research was collecting a large amount of data on road accidents and holding it on Hollerith cards. In this form analysis was limited and it was decided that a new general purpose computer would overcome this problem.

The computer that had been ordered was a Ferranti Pegasus with two magnetic tape decks. It had about 2,500 electronic valves which produced a lot of heat. The room where it would be installed had to have air conditioning to get rid of the heat. I think the computer cost £80,000 but the air conditioning and suite of rooms cost about £100,000. There were 48 registers, each five bytes long which gave it a memory of 240 bytes, not much compared with today's PCs! The equivalent of today's hard disc was a magnetic drum of 8196 words. Again each word had 5 bytes so this was a magnetic drum/disc of 40980 bytes. All the electronics were housed in large grey painted metal cabinets which were manufactured by a firm called Mulliner who also made the bodywork for Rolls Royce cars. It was a large impressive piece of equipment.

Before it was delivered, two of us were sent on a two-week course at West Gorton in Manchester. This was all theory, there was no practical training as there were no spare machines available on which to hone one's skill. Before it was handed over, Ferranti engineers had to commission it, running special programs for various parts of the machine.

Each new machine had incorporated some new aspect which was designed to improve the operation or reliability so these unique aspects took up the time in testing. One of these special features was the replacement of a small part of the magnetic drum with acoustic delay lines. The commissioning engineer spent a long time writing some software for testing this part of the machine and when it was finished he tested it and proclaimed that the delay lines were working perfectly. The other Ferranti engineer said "Yes, so well that it still works when I have taken one of the circuit boards out!"

When it had successfully completed its acceptance trials it was handed over to Road Research Laboratory and was available to researchers on a booking system (one user at a time). However these early machines had to be programmed by the users themselves in the actual machine code. There were no spreadsheets, databases or other packages so there was a steep learning curve for those who had the vision of the potential of an electronic computer.

At the beginning there was a lot of spare time available so we engineers could hone our fault finding skills. One of us would go out of the room and the other would choose a valve at random, remove it and snip one of the pins off and then replace it. The other engineer would then return and attempt to find the fault. He had to work logically as you couldn't take each one of 2500 valves out one at a time and inspect the base. We were also able to try out our own programming skills but at this time we did not spend long on this task. I can remember the first one I wrote was to sum the terms of a series of $1/n$ and using some spare time I reached 500 terms. No use but a bit of pure mathematics!

The machine we had was a Pegasus II which had some extra instructions for character manipulation associated with Hollerith cards. We did not bother too much with these instructions as we did not have any card equipment. However one day a user came to me and said 'The 37 instruction is not working.' My reaction was 'It has passed all the engineer's test programs so there can't be a problem.' However I discovered that we had older engineer test programs and these did not test the newer instructions.

Road Research Laboratory was collecting a large amount of data automatically, mainly on various types of magnetic tape so there was a requirement for direct input. SR39 (special register 39) was developed in-house to satisfy this requirement. Then an analogue to digital converter was interfaced and a graph plotter. Our Pegasus computer started to look like a Christmas tree but I found a great deal of satisfaction when a particular piece of equipment was shown to operate reliably after several months of my effort.

We had the Pegasus computer for ten years and it gave me my grounding in computing for which I am grateful.

Correspondence

Ann Moffatt and Others

Ann Moffatt has commented on Nick Enticknap's *Resurrection 79* report of the Making IT Work Conference held last summer —“In the early 1970s the Science Museum asked the BCS if they would help establish a computer gallery. I was the first woman elected to the BCS council and was appointed liaison person. The gallery keeper was Jane Pugh.

The Science Museum had a Mathematics Gallery and so already had some interesting computer artefacts including early Napier's bones, early slide rules, Pilot Ace, parts of Babbage's prototype engines, some punched card sorters and an early computer to name just a few. Many of these were not exhibited because so little was known about them. Also, they were all hardware and in the early 1970s software was king.

We also wanted to demonstrate an interactive application that would be of use to visitors and decided to develop a system to show visitors the route on the London rail system to get from one station to another destination showing which line to take and where to change lines etc. We worked with Imperial College to set up a terminal to their computer.

We had interactive computer generated music and art in an especially constructed 'dark' cave.

In a letter dated July 1975, after I had left for Australia, Jane Pugh informed me that 'another triumph was that the Manchester University differential analyser was brought back to life on Tuesday. After six and a half hours of adjusting it finally drew quite a passable circle to cries of triumph from the weary group around it'.

Of the railway map project Jane wrote 'The terminal is also online for public operation now, in a temporary position in the central aisle. It is usually 10 deep in people at all times, working or not.'

The gallery was opened on 17th December 1975 by Grace Hopper.

It is sad that this sort of history is now in danger of being rewritten. Nicholas writes 'The Science Museum was involved from the start — but its own exhibits of those days were static objects, not working machines'. You can see that just isn't true. We worked hard to move from things in glass cases to working examples of computing as it was in the 1970s. I have press cuttings and letters of this early work. It would be nice to see them preserved."

Nick Enticknap replies —

"I used to go to the Science Museum regularly in the late '70s and the '80s. On those occasions I never once came across a working exhibit in the historic computing section. I recall the Differential Analyser clearly — it was a fascinating object! — but I never saw anyone demonstrating it. Nor did I with see anyone operating or demonstrating the Pilot Ace, the unit record equipment or any of the other artefacts on display. So, I believe my comment in *Resurrection 79*, made from experience as a regularly visiting member of the public on random occasions before the CCS was formed and with no privileged access to special events, is a fair one. I note Ann does not mention any working machines apart from Hartree's, and that on only one occasion. I would argue that, while computer generated art and music and the railway map project showed what computers in general were capable of in the seventies, they did not demonstrate the capabilities or the technology of historic computers."

Doron Swade added —

"Ann is correct when she says that there were working exhibits in the 1975 gallery. However, Nick was referring to 1989, some 14 years after the opening of the gallery and by that stage most of the working exhibits in the gallery were either defunct or had been removed. There is another more oblique sense in which Nick is correct: while there were electronic educational interactive and non-interactive exhibits in the computer gallery, there were no working general purpose electronic computers, predating as it did the tsunami of personal computers that was to follow. Nick is also correct in respect of historic electronic computing machines. While the Science Museum at that time had well-staffed workshops, the specialist expertise to restore vintage electronic computers to working order was not available in-house and large historic computers were by default displayed as static exhibits. A major motive for founding the CCS was to marshal volunteer effort to restore machines to working order for public demonstration and display. So, happily, both Nick and Ann are correct"

[Honours even, I think. But the reference in Ann's letter to having got the Science Museum's half of the Differential Analyser working again, raises an intriguing notion. Now that both halves are in the same ownership (the other half is at MSI in Manchester and is also known to work thanks to Charles Lindsey and his team) wouldn't it be wonderful if? Too much to ask? Probably.] [ed]

Ann Moffatt is Director of the Silicon Coast Extracurricular Code School, Doron Swade a former curator at the Science Museum and Nick Enticknap the emeritus editor of this very publication.

50 Years Ago

From the Pages of Computer Weekly

Brian Aldous – TNMoC Archivist

Keeping Track of Space 'Junk': The first of three British earth stations in the Anglo/American Intermediate Defence Communications Satellite Programme (IDCSP) is using an English Electric Elliott 4130 computer to analyse reception data and for general research in a wide range of satellite communication experiments. (CW77 p5)

Boost for Big System 4 Machines: An enhanced version of English Electric Computers' System 4/70 and 4/75 machines is being offered to a number of universities and research establishments. The improved model offers a general boost in performance under Gibson Mix conditions of about 20% and costs approximately £10,000 more than the standard model. Only single processor models are being offered. (CW78 p1)

DEC Expand for PDP Machines: With the addition of a further 6,000 square feet to their factory at Arkwright Road, Reading, almost doubling previous capacity, Digital Equipment Corporation have announced that they are to start a full manufacturing programme for the PDP computers in this country. (CW78 p16)

SPL and US Firm Study Heathrow Customs System: A first contract for a feasibility study of the £2 million plan to speed Customs at London Airport, Heathrow, has been placed by the GPO's National Data Processing Service with Planning Research Corp of Los Angeles, in conjunction with Systems Programming Ltd of London. (CW79 p1)

More Terminals on Titan Multi-Access: The Titan multi-access system at Cambridge University Mathematical Laboratory will soon be able to handle 24 remote terminals at one time. This will become possible when the core store is enlarged from 64K to 128K. (CW79 p20)

Fighting Chance for UK Industry: The long drawn out speculation about the constitution of a merged UK computer industry came to an end last week with the announcement of the formation of International Computers Ltd. The new company will combine the commercial and scientific computer business of ICT and English Electric Computers, with both Plessey and the government taking a financial stake. (CW80 p1)

Data Logging System from Solartron: A new modular, programmable data logging system has been introduced by Solartron Electronics Group Ltd, Farnborough, Hants. IDAS, or Industrial Data Acquisition System, has a

fully industrial specification and the modular design allows a broad range of system configurations to be manufactured on production line principles. The flexibility of IDAS is passed on to the user who can expand and alter his system simply by adding new modules or programs as required. (CW80 p9)

Automated Grocery Warehouse: Groceries supplied to 1,000 Co-operative Society shops in the North-East of England will, from 1970, be distributed from a single, highly automated warehouse now being constructed in Birtley, County Durham. Most local societies within the area, which embraces North Yorkshire, Cumberland, Durham and Northumberland, have agreed to the Co-operative Wholesale Society handling the purchase and distribution of their supplies rather than handling them themselves through small and relatively inefficient warehouses, which would always need to rely on a high degree of manual labour. (CW82 p8)

Oil Depot Controlled by 1130 System: The British subsidiary of the Compagnie Francaise de Petrol, Total Oil, who opened their first petrol station in 1960, in the sixth year of their existence and now have 600, have opened an IBM 1130-controlled terminal at Leeds at a cost of £500,000, claiming it to be the most advanced in the UK. (CW82 p16)

Myriad to Control Network: The first major hardware step in the establishment of a computer network at English Electric's Mechanical Engineering Laboratory, at Whetstone, near Leicester, took place this week with the delivery of a Marconi Myriad II with 16K of core store. This machine will be the network controller. (CW83 p1)

Titan Aids Balance of Payments Evaluation: Despite devaluation many experts feel that the balance of payments problem currently being experienced in the UK will recur and grow worse, unless more fundamental changes are made in the economy. In an attempt to see how these changes could be made and to evaluate their possible effects a model of the economy is being constructed using the Titan computer at Cambridge University. (CW83 p2)

Compact Machine Launched: Innovation not research is the big point in the computer industry, said Mr Iann Barron, managing director of Computer Technology, yesterday when he announced the volume production of the Modular One system at the new Hemel Hempstead factory. (CW85 p1)

On-Line Use for Elliott Display Units: Commercial quantities of a new family of alpha-numeric computer displays, with associated keyboard and light pen input facilities, are now being manufactured at Borehamwood by

Elliott Automation. The family has been developed for on-line communication applications in all environments and includes units specially designed for use in military aircraft and armoured vehicles as well as commercial and industrial operations. (CW85 p16)

CEGB get Giant Hybrid System: A massive hybrid computing system is to be among the equipment installed at the CEGB's new Bankside computer centre. Almost certainly the most powerful system of this kind in the UK, it will comprise three EAL 8800 analogue-hybrid computers, an EAL 8400 digital computer, and an EAL 8930 data interface and linkage rack. (CW85 p16)

Argus 400 Ordered for Stock Exchange System: A service which will provide City stockbrokers with continuously updated data on changes in the prices of about 700 quoted securities, is to be established in July next year by the London Stock Exchange. Based on a £95,000 12K Ferranti Argus 400 system, the service is expected to be used by about 250 stockbrokers involving GPO data transmission facilities between the Stock Exchange and well over 100 office buildings where stockbrokers are located. (CW86 p1)

LEO III Used for Mailing List: A computerised mailing list, containing the names and addresses of over 150,000 companies, has been completed by the English Electric Computers Bureau, for the Financial Times. (CW87 p3)

1901 Operational for Newspaper Typesetting: First live operations have been started by Portsmouth and Sunderland Newspapers Ltd on an ICT 1901 computer, which is destined to become the heart of one of the most sophisticated computer typesetting projects in the UK. One weekly newspaper — the West Sussex Gazette — is already being produced by the computerised system, and by the end of the year this will have been joined by six other weekly papers and one evening paper. (CW88 p3)

ICT's Optical Memory Unveiled: First fruits of ICT's work into optical stores were shown last week at the Research and Development Centre, Stevenage. The research model shown was a read-only memory with a capacity of 65,000 68-bit words. The memory plate currently being used is about 10 inches in diameter and it is expected that the packing density will be improved at least four times. (CW89 p1)

[Given the obvious resemblance between this last item and the later Compact Disc (1982) and Phillips Laserdisc (1972) one is bound to wonder what became of this development. Did it ever come to market? Was the 1966 James Russell patent a problem, perhaps? In the absence of the telegram service, answers on a postcard please.][ed]

Forthcoming Events

London Seminar Programme

15 th Mar 2018	Founding of the Numerical Algorithms Group (NAG), its Early Days and its Rôle Today	Brian Ford and colleagues
19 th Apr 2018	Constructing the EDSAC replica — Lessons Learned	Andrew Herbert
17 th May 2018	Programmed Inequality	Marie Hicks

London meetings take place at the BCS in Southampton Street, Covent Garden starting at 14:30. Southampton Street is immediately south of (downhill from) Covent Garden market. The door can be found under an ornate Victorian clock.

You are strongly advised to use the BCS event booking service to reserve a place at CCS London seminars. Web links can be found at www.computerconservationsociety.org/lecture.htm. It is important to pre-register because if too few registrations are made, the meeting may be assigned a lecture room too small to accommodate unregistered members.

For queries about London meetings please contact The CCS secretary Roger Johnson (contact details inside back cover).

Manchester Seminar Programme

20 th Mar 2018	Restoring the 2966	Delwyn Holroyd
18 th Sep 2018	To be announced	
16 th Oct 2018	To be announced	

North West Group meetings take place in the conference room of the Royal Northern College of Music, Booth St East M13 9RD: 17:00 for 17:30.

For queries about Manchester meetings please contact Alan Pickwick at alan_c_pickwick@btinternet.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.

Museums

MSI : Demonstrations of the replica Small-Scale Experimental Machine at the Museum of Science and Industry in Manchester are run every Tuesday, Wednesday and Sunday between 12:00 and 14:00. Admission is free. See www.msimanchester.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 : Colossus Galleries open daily 10.30-17.00; full Museum open Thursday, Saturday and Sunday 12.00-17.00. Situated on the Bletchley Park campus, TNMoC covers the development of computing from the "rebuilt" Colossus codebreaking machine via the Harwell Dekatron (the world's oldest working computer) to the present day. From ICL mainframes to hand-held computers. Please note that TNMoC is independent of Bletchley Park Trust and there is a separate admission charge. Visitors do not need to visit Bletchley Park Trust to visit TNMoC. See www.tnmoc.org for more details.

Science Museum :

There is an excellent display of computing and mathematics machines on the second floor. The *Information Age* gallery explores "Six Networks which Changed the World" and includes a CDC 6600 computer and its Russian equivalent the BESM-6 as well as Pilot ACE, arguably the world's third oldest surviving computer.

The new Mathematics Gallery has the Elliott 401 and the Julius Totalisator, both of which were the subject of CCS projects in years past, and much else besides.

Other galleries include displays ranging from ICT card-sorters to Cray supercomputers. 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.

Committee of the Society

Chair: **David Morriss** FBCS CEng CITP: *dmorriss@btinternet.com*
Secretary: **Dr Roger Johnson** FBCS: *r.johnson@bcs.org.uk*
Treasurer: **Dr David Hartley** FBCS CEng CITP: *david.hartley@clare.cam.ac.uk*
Chair, North West Group: **Bob Geatrell:** *bob@geatrell.co.uk*
Secretary, North West Group: **Alan Pickwick:** *alan_c_pickwick@btinternet.com*
Resurrection Editor: **Dik Leatherdale** MBCS: *dik@leatherdale.net*
Website Editor: **Dik Leatherdale** MBCS: *dik@leatherdale.net*
London Meetings Secretary: **Kevin Murrell** FBCS: *kevin.murrell@tnmoc.org*
Membership Secretary: **Dr Dave Goodwin** MBCS: *dave.goodwin@gmail.com*
Media Officer: **Dan Hayton** MBCS FRSA: *daniel@newcomen.demon.co.uk*
Digital Archivist: **Prof. Simon Lavington** FBCS FIEE CEng: *lavis@essex.ac.uk*
Awards Sub-Committee Co-ordinator: **Peta Walmisley:** *peta@pwcepis.demon.co.uk*

Awards Sub-Committee

Rachel Burnett, Roger Johnson, David Hartley

Museum Representatives

Bletchley Park Trust: **Peronel Craddock:** *pcraddock@bletchleypark.org.uk*
National Museum of Computing: **Andrew Herbert** *andrew@herbertfamily.org.uk*

Project Leaders

SSEM: **Chris Burton** CEng FIEE FBCS: *cpb@envex.plus.com*
Bombe: **John Harper** Hon FBCS CEng MIEE: *bombeebm@gmail.com*
Elliott: **Terry Froggatt** CEng MBCS: *ccs2@tjf.org.uk*
Software Conservation: **Dr David Holdsworth** 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*
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: *andrew@herbertfamily.org.uk*
Bloodhound Missile/Argus: **Peter Harry:** *pdh@imtex.co.uk*
IBM Group: **Peter Short** MBCS: *peters@slx-online.biz*

Co-opted Members

Prof. Martin Campbell-Kelly FBCS CITP FLSW: *m.campbell-kelly@warwick.ac.uk*
Past Chair **Rachel Burnett** FBCS CITP Hon D. Tech: *rb@burnett.uk.net*

Resurrection is the journal of the Computer Conservation Society.

Editor — Dik Leatherdale

Printed by — BCS the Chartered Institute for IT

© Computer Conservation Society