AMATEUR COMPUTER CLUB NEWSLETTER

VOLUME 5 ISSUE 2 JUNE 1977

DIY COMP CONF

The photograph on the right shows the ACC stand during one of the rare quiet moments at the conference on May 14th.
This conference, mentioned in Vol 4 Iss 6, was held in the august surroundings of the IEE headquarters at Savoy Place. As far as attendance was concerned it was a complete sell-out, over 600 delegates paid the £8 fee and on the morning of the conference the organisers were turning away late arrivals without tickets. More than fifty ACC members took advantage of the reduced offer of £6.50, and in general seemed to have felt that it was good value for money.

In conjunction with the symposium (chaired by Dr. Chris Evans of the NPL) there was an exhibition of equipment manufacturers, publishers' representatives etc., and this included a small stand manned by ACC committee members. We showed a WB1 built by Tony Cassera and the prototype 7768. The gospel concerning ACC was spread amongst the multitude , forty plus new members were signed on and many potential members chatted up. Interest in personal computing (or whatever you like to call it !) was highest amongst people who had attended the conference out of curiosity.

As the first gathering of this sort organised by a professional conference company, it was undoubtedly a financial success and also did a great deal of good for the ACC. Perhaps members who attended might care to tell ACCN what they thought of it all.

Bob Warren

IN THIS ISSUE

ACC CONVENTION

The ever increasing ACC membership, the growing awareness by electronic hobbyists that home built computers are possible, and the evident success of the first computer constructors conference has encouraged the committee to decide to hold an ACC Autumn Convention.

The date and rendezvous etc. are still exceedingly tentative, but it is anticipated that mid - late October in the London area will probably be adopted.

It is hoped that the Convention will consist of displays of members' computers, kit computers, and systems from manufacturers, surplus dealers and a general 'bring & buy'. Naturally the format of the Convention will largely depend upon what members feel they want. So please, comments ASAP concerning how you feel this event should be organised.

GAMES

ACC members are invited to a 'games evening' from 6.00pm on Wednesday 13th July at the Computer Unit of City University, St. John's St. EC1 (nearest underground stations are Angel, Barbican and Farringdon).
Available will be 10 terminals with BASIC/FORTRAN/ Algol 60 & other languages, and games including Star Trek & Moon Landing.
Unfortunately there are only 10 terminals, so the attendance will have to be limited to the first 30 or so to turn up.

GALDOR

An ACC meeting will be held at Galdor on Sunday 31 July from 2.00pm. All members are invited, to look, make admiring noises, and play with the new machine.

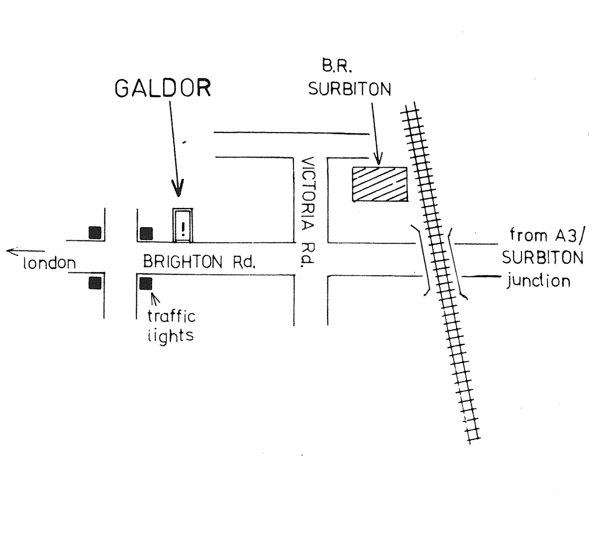
One of the ACC's largest members, the Galdor Centre is now running an ICL 1903 (32k) to give a cheap service for clubs' mailing lists, student projects, fringe research etc. This is made possible by using discarded 1900 parts which are beginning to appear on the scrap market.

Galdor used to have an ICT 1301, which responded to hardware improvements that its designers never dreampt of in the 1950's.

The 1900 series has less scope for tinkering about inside the processor, but all sorts of gadgets can be plugged in; Galdor hope to provide a timesharing service soon.

The Galdor Centre, 52 Brighton Rd.,Surbiton KT6 5PL
01-399 1300

Andy Keene



NU BUS

SUMMARY OF SUGGESTED BUS SPECIFICATION

Pat Crowe & Dave Howland

AIM OF SPECIFICATION

The rapid growth in the availability of hobbyist computer systems and system components in the USA has been largely dependent on a considerable degree of standardisation. The S-100 bus supported by MITS, Polymorphic Systems, Imsai and Vector, is the most popular there but for several reasons is not suitable as a standard in this country. This document is an attempt to produce a specification for a general purpose 8-bit microprocessor bus system suitable for adoption by British (or even European) Hobbyists. The main principles guiding the choices were that it should be a system;

1) Using readily available equipment practice (boards, card-frames, connectors etc.) and allowing easy manufacture of printed circuit boards by the individual if required.

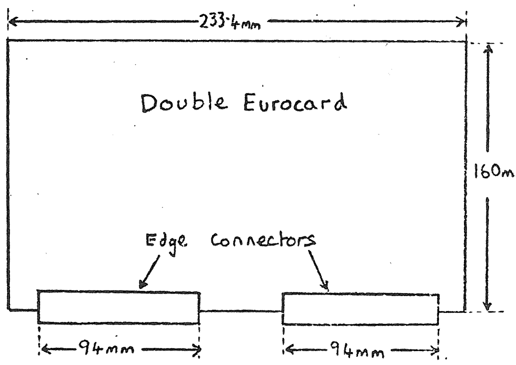
2) Sufficiently flexible in definition to allow its use with any existing (and hopefully future) 8-bit microprocessor without penalising any particular one by requiring complex logic to interface with the bus.

3) Sufficiently flexible to allow good possibilities for implementing personal experimental or non standard systems without departing from the framework of the standard.

4) Flexible enough to make it possible to produce equipment based on the system to full professional standards, while at the same time not penalising the experimenter with a low budget.

EQUIPMENT PRACTICE

The equipment practice decided upon is the double sized Eurocard (with a possible small system option of a single sized Eurocard). The Eurocard has gained a good deal of acceptance in Britain and Europe and is gaining in popularity all the time. It is supported in this country by Vero and the practice is therefore freely available. The double sized Eurocard is shown below.



The card has 2 64-way connectors. The proposal is that one of the connectors should be connected as a bus for the full length of the card-frame. The other connector is optional and is generally to be used for point to point wiring of special signals relating to a particular card. This would mean that a RAM card would have only one connector, whereas an I/O card would use the other connector for, among other functions, communicating with the outside world. As the non-bussed connector connections are unspecified it represents one of the methods by which flexibility is ensured.

The connector is a two-part 64-way type. This is one of the standard Eurocard connectors and is available from many different manufacturers. One part is attached to the card and mates with the other part which is attached to the card frame.The reasons for choosing a two-part connector are as follows;

1) They are reliable and replaceable.

2) They allow the production by an amateur of a professional quality PCB without the need to have access to gold-plating equipment.

3) Although they are not cheap, neither is any other type of reliable edge-connector. On a prototyping card one has the option of fitting only one connector if required.

4) The largest manufacturer of prototyping boards in this country, Vero, produces double-sized Eurocards using this connector . They do not intend to support integral connectors at this time.

Card frames for this system are also available from Vero in a variety of sizes and styles.

DEFINITION OF BUS SIGNALS

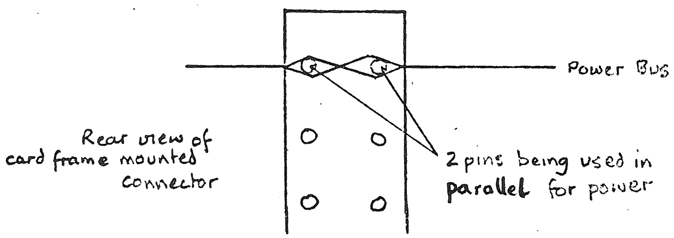
ELECTRICAL 1) All signals will be asserted low.

2) All drivers should be capable of sinking at least 45mA and driving into at least 50pF.

SIGNAL DESCRIPTIONS

Power Supplies 12 pins are used for power. They are connected as 6 pairs of pins to allow reasonable current (max 4A) to pass through the connector to each card and to simplify the bus connection which will have to pass a considerable current.

The supplies provided for are +12V, +5V, 0V, -5V, -12V and a spare voltage; PUD (Power, User Definable)


Power Bus
Rear view of card frame mounted connector
2 pins being used in parallel for power

Address 16 pins $\overline{\text{A0}}$ - $\overline{\text{A15}}$

Data 8 pins $\overline{\text{D0}}$ - $\overline{\text{D7}}$

Memory and Peripheral Control 10 pins. A set of signals, a subset of which can be used by any microprocessor to access memory or peripherals;

$\overline{\text{M}}$/P	(Memory /Peripheral)	A set of signals used, in particular,by 6800,6500,2650,F8
$\overline{\text{R}}$/W	(Read / Write)	
$\overline{\text{MADD}}$	(Memory Address)	
$\overline{\text{PADD}}$	(Peripheral Address)	
$\overline{\text{MRD}}$	(Memory Read)	Used in particular by Z80,8080,8085,SC/MP
$\overline{\text{MWR}}$	(Memory Write)	
$\overline{\text{PRD}}$	(Peripheral Read)	
$\overline{\text{PWR}}$	(Peripheral Write)	

Note; Some microprocessors from each of the above groups may be able to provide signals for the other group. Also all microprocessors in group B having a 'Wait' facility for memory or peripheral access will use signals in group A as a cue to return a Wait request.

$\overline{\text{WAIT}}$ The signal is returned by an addressed memory or peripheral device if it requires extra time to perform the data transfer.

$\overline{\text{REFRESH}}$ This signal is sent (probably) from the CPU card and is an indication that the 7 low order address lines contain a refresh address for dynamic memory and that a refresh cycle should be performed.

Note; The selection of which of the above signals to use is a user function. Codes of pratice will exist to explain the system significance of using or discarding a given facility.

System Initialisation 2 pins

$\overline{\text{RESET}}$ A clean pulse generated by the CPU card to initialise the system

$\overline{\text{INHR}}$ (Inhibit Read) A signal used by all RAM boards to inhibit reads from them. Its purpose is to allow 'shadow' PROM's to be temporarily placed at the same memory locations as RAM during system initialisation, for bootstrapping purposes. The

PROM will be able to write into the RAM, if required although it occupies the same address space.

Memory Mapped I/O Block Decoding   1 pin

$\overline{MNIOB}$ A single signal which is a partially decoded address indicating that the address presently on $\overline{A0}$ - $\overline{A15}$ lies within a pre-defined block of 256 addresses. This is to reduce the amount of decoding required by I/O boards in a system using memory-mapped I/O.

DMA Protocol   2 pins

$\overline{DMAR}$ (Direct Memory Access Request)
DMAG (Direct Memory Access Grant)

Interrupt   6 pins

$\overline{INT}$ (Interrupt)
$\overline{NMI}$ (Non Maskable Interrupt)
$\overline{IUD1}$, $\overline{IUD2}$, $\overline{IUD3}$, $\overline{IUD4}$ } User definable interrupt wires, subject to codes of practice for 8080, Z80 etc.

User Options   7 pins

$\overline{OUD1}$ to $\overline{OUD7}$ User definable options, subject of codes of practice which suggest such functions as system clock etc.

Any comments or suggestions concerning the proposed bus or its application to specific microprocessors would be welcomed by;

Pat Crowe, 43 Cypress Rd., Woodley, Reading RG5 4BD
tel Reading 693911 (home)

Dave Howland, 5 Bay Tree Rise, Langley Hill, Calcot, Reading. tel Reading 411677 (home)

New microcomputer kit

HARDWARE BITS

MOSTEK is second-sourcing Fairchild's F8 as well as the Zilog Z80.

Meanwhile, Zilog have announced the Z80A, a 4MHz clock version of the original Z80 (2.5MHz clock).

Intersil have cut the prices of their IM6100 range, for example the IM6100CCPL MPU is now £12 for 100 up.

Fairchild are to second-source the 6800 family, and have announced the 9940 MPU, a 16 bit micro which emulates the Nova 1200 mini, and which will be constructed using IIIL bipolar technology.

ITT are to second-source GI's 1600 16 bit MPU.

Rockwell are to produce the MOS Technology's 6500 microprocessor family.

NSC are now selling the SC/MP Mark II, a faster, lower power, single (5V) supply version.They also make an 8080 MPU and are selling CPU and memory boards compatible with Intel's SBC 80/10 .Finally, they will be second-sourcing the Signetics 2650.

Signetics will be second-sourcing NSC's SC/MP II.

Texas Instruments continue to exploit their 9900 16 bit MPU; the 9980 is a stripped down version which uses an 8 bit data bus, runs much slower, and fits into a 40 (instead of 64) pin package. The newly announced 9940 is a single chip computer having 2K bytes EPROM and 128 bytes RAM on the chip, and an instruction set basically similar to the 9900. Peripheral chips now available for the 9900 family include the 9901 parallel I/O , 9902 UART, 9903 synchronous data controller and 9904 4 phase clock generator. TI are also selling a single board computer based on the 9900 for around £300.

Intel announced the 8085, a modern version of the 8080 having an on-chip oscillator & reset circuitry, as well as a serial I/O port, and needing only a single (5V) supply. The 8048 and 8748 are single chip microcomputers having 1K bytes of ROM and 64 bytes of RAM on-chip as well as 27 I/O lines (the 8748 has a UV eraseable ROM, the 8048 uses a fixed, metallised ROM). Intel have also cut the price of their 2708 1K x 8 EPROM to £36.70 (1 off) and have announced the 2716 2k x 8 EPROM.

LOCAL GROUPS

MANCHESTER & Co.

Would any member in the North West interested in forming a local group centred on Manchester please drop me a line or give me a ring on 061-980-2755 (home) or 061-236-9432 ext 211 (work)
P Wade 6 Mossgrove Rd., Timperley, Cheshire

MIDLANDS MEETING 17th April

This meeting proved to be a gala event, as Dave Goadby brought along his 'Computer Workshop' M6800 system. This is available in kit form with very clear instructions for construction and Dave thoroughly recommends it to anyone familiar with soldering, the main precaution is to avoid solder bridges.

Dave gave us a rundown on the general system, and also his reason for choosing a 6800 system. His previous efforts had been with an 8008 and 8080, both of which require a considerable amount of support circuitry, and, more important still, the software is very expensive. Whereas the M6800 requires relatively little support circuitry and there is plenty of relatively cheap software available.

He keeps his software on standard audio cassettes played through an ordinary portable cassette recorder via a very simple interface circuit from 'Micro Trek' magazine. This is not full CUTS standard, but it uses the same frequencies as CUTS.

Direct communication with the computer is at present through a KSR33 Teletype, but plans are afoot to change over to VDU.

A BASIC translator was loaded to demonstrate the operation for direct BASIC programming, and then Star Trek was loaded, providing considerable antertainment.

Another meeting is planned for June 26 when, work permitting, Dave will bring his system again. All members welcome, contact Roy Diamond, 27 Loweswater Rd., Coventry tel; Coventry 454061

Two new data books from Texas Instruments;

'Memory and Microprocessor Data Book' £3.00
'TTL Data Book Second Edition' £5.00

NETHERLANDS CLUB FORMED

An amateur computer enthusiasts' club is being formed in the Netherlands. Details from Dick Barnhoorn, Delftsekade 12, Leidschendam.

HIGH CLASS COMPUTER COMICS

EUROMICRO NEWSLETTERS

Contains scientific papers & conference announcements. Quarterly. Details from Prof. R Hartenstein Universitat Karlsruhe, Institut fur Informatik IV Postfach 6380, D-7500 Karlsruhe 1, West Germany

NEW LOGIC NOTEBOOK

Subscription for 12 monthly issues $US 95
Microcomputer Technique, 1120 Centre Office Bldg., RESTON, VA 2209L USA

MICROCOMPUTER DIGEST

Monthly. Details from Microcomputer Associates PO Box 1167, CUPERTINO, CA 95014 USA

COMPUTER MAGAZINE

IEEE Computer Society, 5855 Naples Plaza, Long Beach, CA 90803 USA

COMPUTER DESIGN

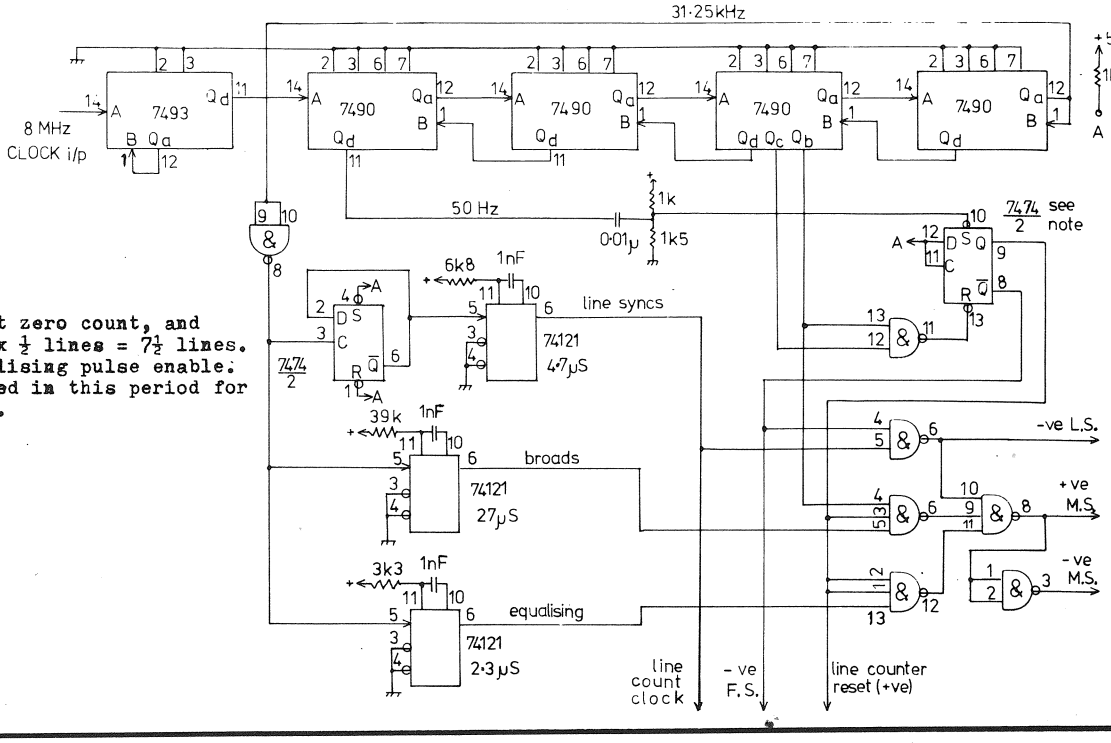
Computer Design Publishing, PO Box A, Winchester MA 01890 USA

MINIS ET MICROS

zero-un informatique, 41 rue de la Grange aux Belles, 75010 Paris, France

MIXED SYNCH GENERATOR

Bob Warren



D-type set at zero count, and reset at 15 x ½ lines = 7½ lines. This is equalising pulse enable. Broads enabled in this period for 2½ - 5 lines.

FOR US

For the benefit of new members, a summary of the services provided by the ACC for its members;

ACC GENERAL LIBRARY

Is now being run by Frank Cato, 3 Rykneld Way, Derby DE3 7AT (tel Derby 53769).

Items currently available for loan to UK ACC members are;

- ACC Newsletter Vol 1 (20p P&P)
- " " " 2 " "
- " " " 3 " "
- " " " 4 " "
- 'Kansas City' cassette interface circuit folder, contains designs from Byte March '76 (2 circuits) and December '76. (20p p&p)
- RS-232-C / V24 folder, contains RS232C, V1,V24 & V28 interface specifications. (25p p&p)
- 'Bear Microcomputer Systems' reprint of Weeny-Bitter articles. (35p p&p)

Members wishing to borrow any of these items must send stamps or cash to cover the cost of p&p as indicated.

8080/Z80 LIBRARY

The library has grown considerably since the first list was published earlier this year and now offers over fifty items for 8080 and Z80 users. The Software section includes a number of monitors, an Assembler, Editor and two BASIC interpreters. If you are building rather than programming, the Hardware section can help by providing product specs, technical manuals, circuits for CPU boards, A/D interfacing etc. For an up to date listing send a 9" x 4" SAE to the address below and of course if you have anything that may be of use to the Library send it along.

INTEL have kindly allowed the Library access to their own software library, INSITE. The arrangements for this have yet to be finalised, but will result in a large number of programs becoming available to ACC members, anything from floating point maths packs to NIM and other games.

I am hoping to organise an ACC meeting with Zilog for members interested in the Z80. More details in the next Newsletter.

Neil Harrison 15 Hill Rd.,Watchfield,Swindon,Wilts

ACC 6800 LIBRARY

The Library now holds well over 100 items of Hardware information and 50 items of Software, including Editors, Assemblers, and BASIC interpreters. For an up to date index send a SAE A4 size envelope and 32p in stamps (any surplus will be rufunded).

Tim Moore 24 College Rd., Maidenhead,Berks
tel 0628 29073

CHIC

New readers start here ...

CHIC (Cheap Hardware Information Centre) was introduced about a year ago . The intention is to provide a service to members who wish to dispose of or acquire computer equipment in the following categories;

1) Paper tape readers & punches - 8 hole
2) Paper tape readers & punches - less than 8 hole
3) Teleprinters & VDU's - ASCII code
4) Teleprinters & VDU's - non-ASCII code
5) Magnetic tape equipment
6) Magnetic disc & drum equipment
7) Computers
8) Core stores

Anyone who wishes to sell equipment sends the details to

Bob Warren, 90 Tudor Rd., Hampton, Middx
tel 01-979-4193

who will circulate them to subscribers.

To become a subscriber send three SAE's, indicating the categories in which you are interested.

So far, the system has worked well when used, but both vendors & subscribers have been disappointingly low in numbers.

MU BUS

The following information was taken, with the kind permission of J.D.Nicoud, from the April 1977 issue of 'Microscope'. For those interested in following this subject up, the April issue of 'Microscope' is recommended reading, as it contains much more information than that given here.

The objective of MUBUS is to provide a simple yet powerful set of signals allowing the design of modular microprocessor systems, able to accept microprocessors of various manufacturers. The set of control signals has been defined in such a way that only a few natural and easy to interface signals are required in a simple, minimum system.

Interrupt, DMA, multiprocessor features can be added by using additional control lines. Buffered buses are used, but not required for minimum systems implemented on few cards (an existing 'minimum' system implemented on 4 cards includes however 32k byte RAM, 8k byte ROM, I/O and a powerful display controller.)

MUBUS SIGNALS

A 62 Line Bus

The proposed set of MUBUS signals has been defined starting from the general bus consideration. The lines on the bus can be classified into three groups;

- 16 address and 16 data lines, active high
- 20 control signal lines, active low
- 10 power supply and ground lines

In each group, the number of lines really required for a given application varies, and minimum MUBUS systems can use as few as 12 address lines,8 data lines, 5 control lines and two power supply lines, that is 27 lines total. (*Note; although the MUBUS definition allows for 16 data lines, most implementations use only 8. ed.*)

The complete set of control lines is only required for complex multiprocessor systems. Some systems highly dedicated to one processor or one application may require additional control lines. The 8 free data lines in 8-bit systems can be used, but this is not recommended. An additional connector and flexible cables should **preferably** be used.

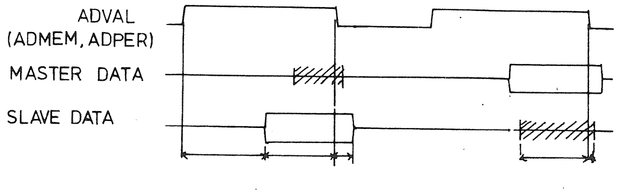
The timing constraints proposed in this section are indicative. They may change with the evolution of the technology and MUBUS should allow the assembly of fast or slow systems using either fast or slow processor and memories, or slow units. Careful checking of the timing of the cards is necessary before putting the system together; the drawings of this section name all the important timings and give typical minimum values, valid for the present 1uS cycle time processors.

ADDRESS AND DATA LINES

These lines are three-state, active high onthe bus, and usually buffered.

ADDRESS 16 lines issued by the master (processor or DMA unit).The timing on the address lines is related to the signals ADMEM,ADPER and REFRESH.

DATA 16 lines issued by the master or by the selected slave. The timing of the data lines depends on the signals ADMEM,ADPER,WRITE , NODA and INTACK.
The data access and write times of slaves (e.g. memory) should match those of the master (e.g. processor)

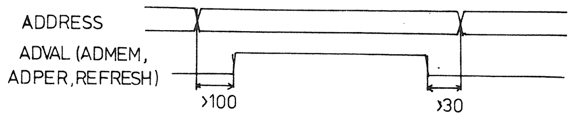


CONTROL LINES

All the control signals, except for two daisy chained lines, are active low, controlled by open-collector gates or, except for a few lines, by three-state lines.

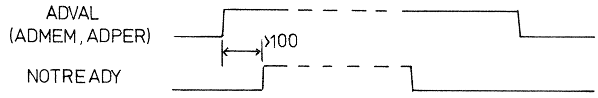
Control signals are always thought of as functional signals, active high, and are represented as such on timing diagrams. The physical inverted signal name on the bus is characterised by a suffix -LOW and it is recommended to set the 'invert' button of the scope when looking at the bus control signals. Experience has shown that a more easy understanding of the complex bus operation is reached if only functionsl signals are considered.

ADMEM(LOW)	Memory address valid (16 bits)
ADPER(LOW)	Peripheral address valid (7 bits)
REFRESH(LOW)	Dynamic memory refresh address valid (7 bits)

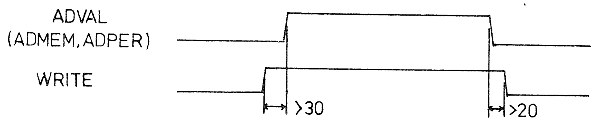


NOTREADY(LOW)Request for the suspension of the processor cycle,also called NOTRDY,NOTYET or STOP.

The NOTREADY signal should be valid when sampled by the processor. Its duration may be limited to 4uS (6800). The relation between the NOTREADY duration and the resulting increase of processor cycle time (and required memory access time) depends on the processor.

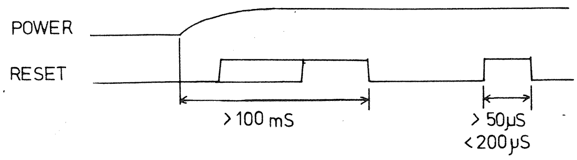


WRITE(LOW) Defined if a read (WRITE = 0) or a write cycle is being performed.



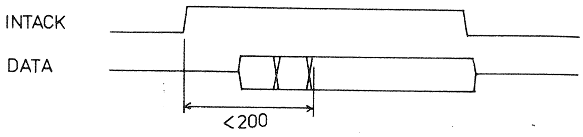
NODA(LOW) The data on the bus is not valid yet. This signal is required only with dynamic memories, and provides a better control of the data bus, especially on processors with a multiplexed data bus.

RESET(LOW) Initialize the processor and the peripherals. Must be a pulse of limited duration in systems including dynamic memory refreshed by the processor.



INTREQ(LOW) Interrupt request line. The interrupting device can be located by software or with the help of the next one or three following control signals.

INTACK(LOW) Read an interrupt vector address from the requesting device of highest priority. No new request should be accepted when INTACK is active.



INTIN Interrupt priority chain input.

INTOUT Interrupt priority chain output.

INTIN and INTOUT are connected together on cards not using interrupts. A pull-up resistor is connected to INTIN on cards using interrupt chain.

NMI(LOW) Non maskable interrupt request, usually reserved for power fail. No priority feature is planned for NMI requests.

HOLDREQ(LOW) Request for the bus by a vice-master (secondary processor or DMA device).

HOLDACK(LOW) Bus granted by the master; HOLDACK is the only bus line still controlled by the master.

HOLDIN Hold request priority chain input.

HOLDOUT Hold request priority chain output.

HOLDIN and HOLDOUT are connected together on slave cards. A pull-up resistor is connected to HOLDIN on cards using DMA chain.

PROCREQ(LOW) Processor request line for multi processor systems. Corresponding priority chain is given by HOLDIN/HOLDOUT lines.

SYSTEM CLOCK Processor clock for application in special multiprocessor systems or for special peripheral interface handling. This signal should not be used in standard systems.

USERSCLOCK 1 Hz or 50 Hz for timing and synchronisation.

POWER SUPPLY LINES

Four voltages are used by the most frequently met components in microprocessor systems. The use of on card regulators and DC-DC converters allows to reduce the number of power supply lines, but increases the system cost.

In addition to the GND and power line for the processor, memory and interface, a separate GND and + 15 V line is required for a minimum noise in A/D converter. The two grounds are usually connected together inside the A/D converter only.

If optical couplers are used, a separate ground and power supply line should be used for the external isolated part.

MUBUS suggested power lines are hence the following

GND (OV) Logic ground for MUBUS signals and power supplies.
+5V Main power supply.
+12V Second main supply for some MOS circuits and DC-DC converters.
-5V
-12V] Additional supplies used in some MOS ccts.
GND(A) Ground for A/D converters.
+15V(A)
-15V(A)] Supplies for A/D converters only.
GND(X) Separate ground for external opto-isolated circuits.
+5V(X) Power for external opto-isolated circuits.

MUBUS IMPLEMENTATIONS

We propose to restrict the developments to two physical standards;

1) EUROCARDS with direct 74 pin connectors. Double size cards are possible for special applications, but the drawback of the smallness of EUROCARDS is decreasing with newly available components.

2) EUROCARDS with indirect 31-pin connectors for simplified low cost systems. Since the bus is not buffered, I/O expansion is provided through a specialised bus using also a 31-pin indirect connector.

The Schools Committee of the BCS has established a Working party which is particularly interested in the use of micros in schools. Would anyone interested please get in touch with J.J.Turnbull, The National Computing Centre, Oxford Rd.,Manchester M1.

MUBUS DIRECT CONNECTOR

component side A		B solder side
+15V(A) □	1	□-15V(A)
GND(A) □		□GND(A)
GND(X) □		□GND(X)
+12V □		□+12V
-12V □	5	□-12V
50Hz/1Hz □		□+5V(X)
■		■
AD0 □		□AD8
AD1 □		□AD9
AD2 □	10	□AD10
AD3 □		□AD11
AD4 □		□AD12
AD5 □		□AD13
AD6 □		□AD14
AD7 □	15	□AD15
REFRESHLOW □		□(SYSTEM CLOCK)
NMILOW □		□PROCREQLOW
INTREQLOW □		□INTACKLOW
HOLDREQLOW □		□HOLDACKLOW
RESETLOW □	20	□NODALOW
WRITELOW □		□ADMEMLOW
NOTYETLOW □		□ADPERLOW
DA0 □		□(DA8)
DA1 □		□(DA9)
DA2 □	25	□(DA10)
DA3 □		□(DA11)
DA4 □		□(DA12)
DA5 □		□(DA13)
DA6 □		□(DA14)
DA7 □	30	□(DA15)
INTOUT □		□INTIN
HOLDOUT □		□HOLDIN
-5V □		□-5V
+5V □		□+5V
+5V □	35	□+5V
GND □		□GND
GND □		□GND

Back panel from rear
Connector AMP 163635-9 or
CONNECTRAL 641-74-Y-001

component side		
GND ○	1	
		○ +5V
+12V ○		
		○ -12V
AD0 ○	5	
		○ AD1
AD2 ○		
		○ AD3
AD4 ○		
	10	○ AD5
AD6 ○		
		○ AD7
AD8 ○		
		○ AD9
AD10 ○	15	
		○ AD11
DA0 ○		
		○ DA1
DA2 ○		
	20	○ DA3
DA4 ○		
		○ DA5
DA6 ○		
		○ DA7
WRITELOW ○	25	
		○ ADMEMLOW
ADPERLOW ○		
		○ NOTREADYLOW
RESETLOW ○		
	30	○ INTRELOW
GND ○	31	

DIN 41617 indirect connector

MUBUS 31-pin indirect

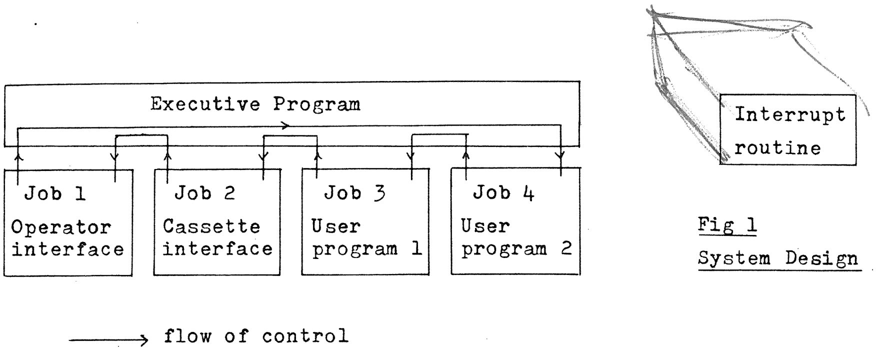


Fig 1
System Design

JOB^N EXEC

MULTIJOBBING EXECUTIVES

Tom Gardner

These are supervisory programs which allow several independant programs (jobs) to be run concurrently. They are not neccessarily large programs only suitable for minicomputer and mainframe systems, but they can be very useful to amateurs using microcomputers. The executive approach to writing system software offers many advantages, the main one being that it is extremely easy to expand the system.

The executive works as follows; At some convenient time the job being run hands control back to the executive which then hands control to the next job in the sequence. Provision must be made for any particular job to remove itself from this sequence (suspend itself). To this end each job has a status word which contains information on whether a job is, for example, waiting to be run or waiting for input/output or is suspended by the operator.

The system that I intend to implement on my 6300 is shown in Fig 1. Note that the interrupt routine is completely separate from the executive and is common to all jobs. Another system program, such as a BASIC interpreter, can be added with very little modification to the system.

The detailed operation of the executive is given in the form of a PASCAL program which performs the following algorithm;

The job being run executes a jump to subroutine instruction which places the return address on the stack associated with this job. The 'subroutine' is actually the executive, which stores the stack pointer in an array location associated with this job. It checks the status word' of the next job in the sequence to see if it can be run. If it can then the stackpointer is loaded from the array location associated with this new job. A return-from-subroutine instruction is then executed which picks the return address from the new job's stack. If the status word had indicated that the new job was suspended, then the executive would have ignored it and looked at the next job.

NUMJOBS is the total number of jobs in the sequence.
JOBNUM is the number of the job currently being run.
SPSTORE is the array in which the stack pointers are stored.
STATUSWORD is the array of status words associated with each job. If the status word is non-zero then the job is suspended.

```
PROCEDURE EXEC (VAR JOBNUM:INTEGER);
          CONST NUMJOBS = 4;(* FOR EXAMPLE *)
          VAR SPSTORE,STATUSWORD:ARRAY[0..NUMJOBS]
             OF INTEGER;
BEGIN SPSTORE [JOBNUM]:= STACKPOINTER;
REPEAT JOBNUM := JOBNUM -1 ;
       IF JOBNUM=0 THEN JOBNUM:=NUMJOBS;
UNTIL  STATUSWORD [JOBNUM] =0; (* UNTIL JOB WAITING
       TO BE RUN *)
STACKPOINTER:=SPSTORE [JOBNUM] ;
END;
```

The following 6800 version of the program is 23 + 3*NUMJOBS words long and takes about 60uS to execute. In order to simplify the addressing the arrays are interleaved as shown below. JOBNUM takes the values 3,6,9,12,.. and NUMJOBS = 3*N where N is the number of jobs in the sequence. The memory map is;

```
address  contents
  0,1    NUMJOBS
  2,3    JOBNUM        16 bit integers
  4      status word
  5,6    stack pointer    Job No. 1
  7      status word
  8,9    stack pointer    Job No. 2
   etc.
```

```
Address Code
 0100   DE02   EXEC     LDX JOBNUM   current job no.
 0102   AF02            STS 2,X      store stackpointer
 0104   09     MISSJOB  DEX          move onto next job
 0105   09              DEX
 0106   09              DEX
 0107   2602            BNE OK       jump if JOBNUM=0
 0109   DE00            LDX NUMJOBS  reset JOBNUM
 010B   E601   OK       LDAB 1,X     test status word
 010D   26F5            BNE MISSJOB  jump if suspended
 010F   DF02            STX JOBNUM   save JOBNUM
 0111   AE02            LDS 2,X      new stackpointer
 0113   39              RTS
```

note; the addresses are purely arbitrary

The instruction LDAB 1,X is used in preference to TST 1,X since the latter takes 40% longer to execute.

The program is fully relocatable, but the data area is not.

One modification to the executive is worth noting, as given above the jobs are self suspending, but this means that if a job gets into an infinite loop the whole system would be locked up. This can be avoided by having a real-time clock interrupt the job and give control to the executive. The return from subroutine instruction would then become a return from interrupt instruction. The disadvantage is that the details of the system become considerably more complex.

If anybody has any queries then I will try to answer them if they send an SAE to 46 Woodcote Green Rd., Epsom, Surrey

SC/MP'R

SC/MP INTROKIT

Having recently put together a SC/MP Introkit, I offer the following pieces of advice to save others' time!

Watch the addressing - the board as set up has the RAM able to respond to multiple addresses, for bits 10,11 are ignored. I connected a second 256 of RAM in 300 - 3FF, with a selection scheme using a 74154 to cover all 4K. Kitbug therefore completely ignored the actual RAM memory, and got completely lost. Removing the top two address lines to the '154 cured the problem. Note also the remark about the top 20 words being required for the Kitbug stack. Your pointer must go at nnEB, no higher.

I wanted to use a Teletype set up for RS232C interface to a Datel 200 modem, so removed the 20mA loop circuitry and used a pair of DM1488/9 chips instead. These invert, so I thought I had to re-invert input but leave output alone. In fact its the other way about.

The fact that Kitbug is listed in the manual is very helpful. I developed a small dump program and loader via the teletype reader/punch. These use PUTC and GECO, which saved a lot of effort, once I had realised that the eighth bit of a character read in by GECO can be retrieved off its stack. Return addr for Kitbug is 001F effective - why does the manual not state that it can be found ?

The board is well made, and you really don't need a lot of extra bits to get it going !

T M Spencer.

SC/MP DUMP   ** Loader on page 10 ** ed.

```
C4 01       LDI 01
37          XPAH P3
CA FB       ST -5(P2)
C4 C4       LDI C4
33          XPAL P3
CA FA       ST -6(P2)     ;PUTC addr in P3,KITBUG return
                          ; on stack
C4 21       LDI 21        ;prepare to put out loader
                          ; blank first
CA FD       ST -3(P2)     ;put count on stack
BA FD Ø1:   DLD -3(P2)    ;decrement & load count
98 C5       JZ Ø2         ;jump out if done
C4 00       LDI 00
3F          XPPC P3       ;output header blank tape
90 F7       JMP Ø1
C5 01 Ø2:   LD @1(P1)     ;get word from memory, note @
9C 07       JNZ Ø3        ;4 blanks is signal for end
                          ; of dump
01          XAE
BA FC       DLD -4(P2)
98 0B       JZ Ø5         ;if we've four zeroes,finish
90 05       JMP Ø6
01    Ø3:   XAE
C4 05       LDI 05
CA FC       ST -4(P2)     ;reset '4 zeroes' counter
01    Ø6:   XAE           ;retrieve char & output it
3F          XPPC P3
90 EC       JMP Ø2        ;repeat for next character
C4 21 Ø5:   LDI 21        ;prepare to o/p trailer
CA FD       ST -3(P2)
BA FD Ø4:   DLD -3(P2)
98 05       JZ Ø7         ;return KITBUG if done
C4 00       LDI 00        ;output
3F          XPPC P3       ; a blank
90 F7       JMP Ø4
C2 FB Ø7:   LD -5(P2)     ;restore KITBUG address
37          XPAH P3
C2 FA       LD -6(P2)
33          XPAL P3
3F          XPPC P3
```

INSTRUCTIONS: Load PC with starting address of dumper, P1 with start address for dump. Ensure four words beyond last word to be dumped are zero. P2 stack address. Hit G & C/R with punch on. 'Disable TTY send' during header if TTY KBD not suppressed.

SIMPLE SC/MP'R

This is a description of a simple microcomputer I am building, using the National SC/MP as a basis. Although the SC/MP is slow and lacks the large instruction set of, say, the 8080, it is cheap and does have several facilities of its own. In my own case, the need to keep costs down outweighed any requirements for speed or sophistication, and indeed, the challenge of making a useful machine from the SC/MP is proving worthwhile.

One of the main requirements was that I should be able to expand the system at any time with the minimum of alterations to the existing circuits. Hence Tri-state buffers (74126) have been used extensively for controlling the flow of data onto the system buses. The boards used are Vero 4½" x 8" DIP boards with 42 way edge-connectors, with the exception of the panel buffer and processor boards which are double-sided with 84 edge connections. These all plug into a Vero 19" rack frame.

Front Panel

This contains; 16 address switches, 8 data switches, 8 data monitor LED's, a single-step mode switch, RUN button, RESET button, DMA WRITE button, LOAD button, power rail monitor LED's and a mains power switch.

Direct Memory Access (DMA)

While the processor is in a reset condition, data from the store location specified by the address switches is displayed on the monitor LED's. Data set up on the data switches may be stored in the addressed location by pressing the WRITE button. DMA is disabled when the processor is running. However, the data switches and LED's are treated as store locations by the processor and data may be strobed from the former and written into the latter under program control. The data switches are given address F000 in hexadecimal, the LED's E000. Data is held in the LED latches until changed.

Single Step

A front panel switch enables a program to be stepped through an instruction at a time, by repeated pressing of the RUN button. With the switch in the 'free run' position, pressing the RUN button takes the machine out of reset and the program is executed at normal speed beginning with the instruction in store location 001. Pressing the RESET button returns the processor to its reset state.

Program Halt

As the extra hardware was minimal, the facility for program halt of the processor was added. This is suggested in the SC/MP manual, and makes use of the status word put out on the data bus by the microprocessor. When a HALT instruction is encountered in the program, the processor is stopped. By pressing the RUN button program execution continues with the next instruction following. This type of halt is useful when pressing RESET and subsequent loss of pointer register contents needs to be avoided.

Memory Board

The memory addressing scheme is such that expansion of the store is carried out by adding 2k x 8 bits of RAM at a time, on a single board module. All modules are identical to the one shown in the diagram, except for different enable signals from the processor board. The RAM's used are 2102s because they are cheaper than most. The input and output lines have been combined onto the system bi-directional data bus by using TTL Tri-state buffers - 74126s.

Memory Addressing

The top four bits of the address bus are decoded via 74154 on the processor card to give sixteen memory 'page' enable signals, $\overline{E0}$ - $\overline{E15}$. The four next most significant address bits are decoded by another 74154 to provide sixteen memory 'block' enable signals $\overline{E16}$ - $\overline{E31}$. A page thus contains 4k bytes of store and a block 256 bytes. As my store is made up of 1k byte sections, I have OR-gated the block enable signals together, to give four 1k select signals; Ea,Eb,Ec & ED. Each memory board contains two 1k sections, so it requires two 1k select signals and one page select signal.

Power Requirements

For the basic processor described, consisting of; Front panel with LED's, panel buffer board, microprocessor board and one 2k byte memory board, the power requirements are +5V @ 1.5A and -7V @ 0.1A.

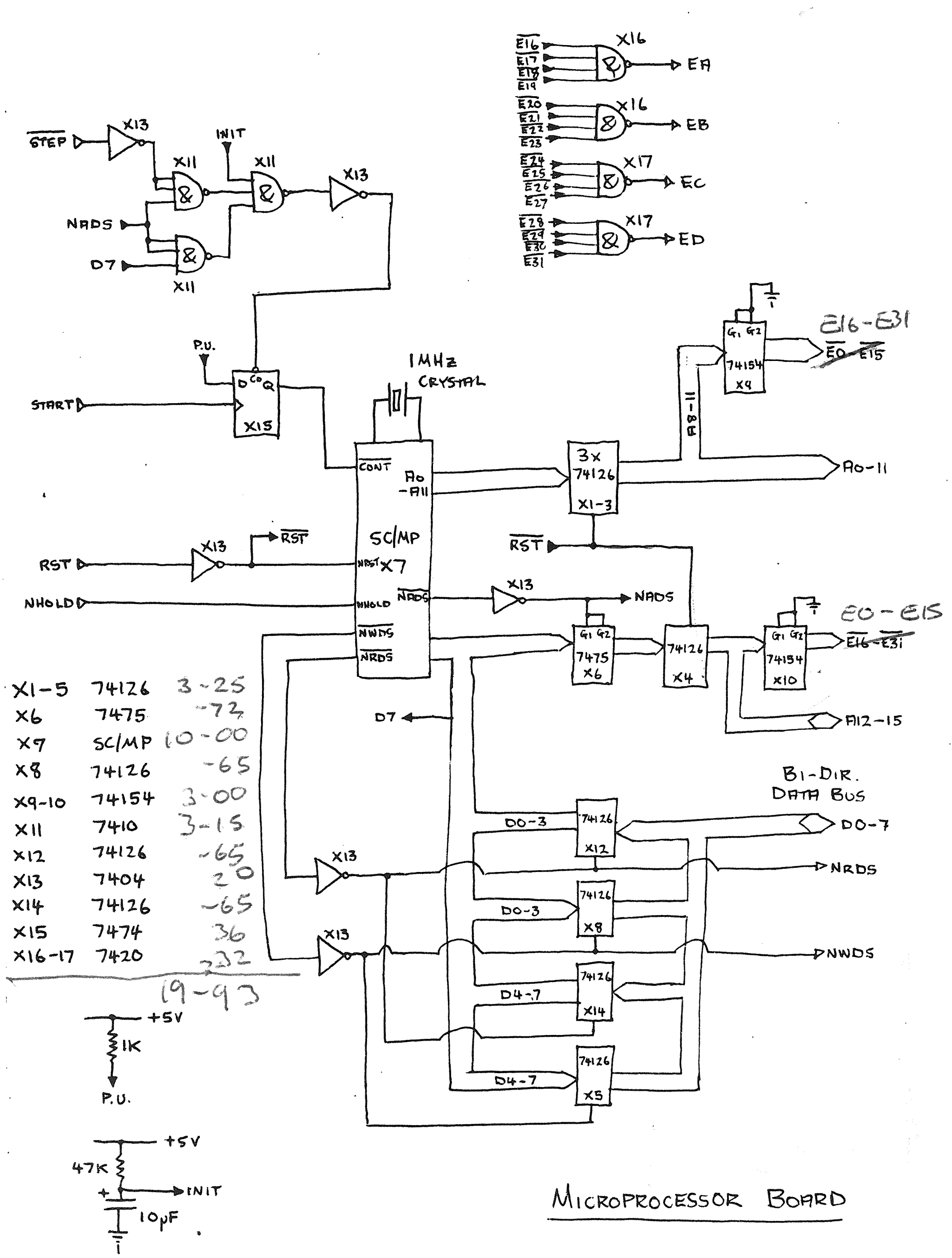
Peripherals

A fourth card is being built containing a paper tape interface and hardware binary loader, the latter acuated by the LOAD button. A fifth card will carry a cassette interface, with a keyboard and VDU coming next.

Conclusion

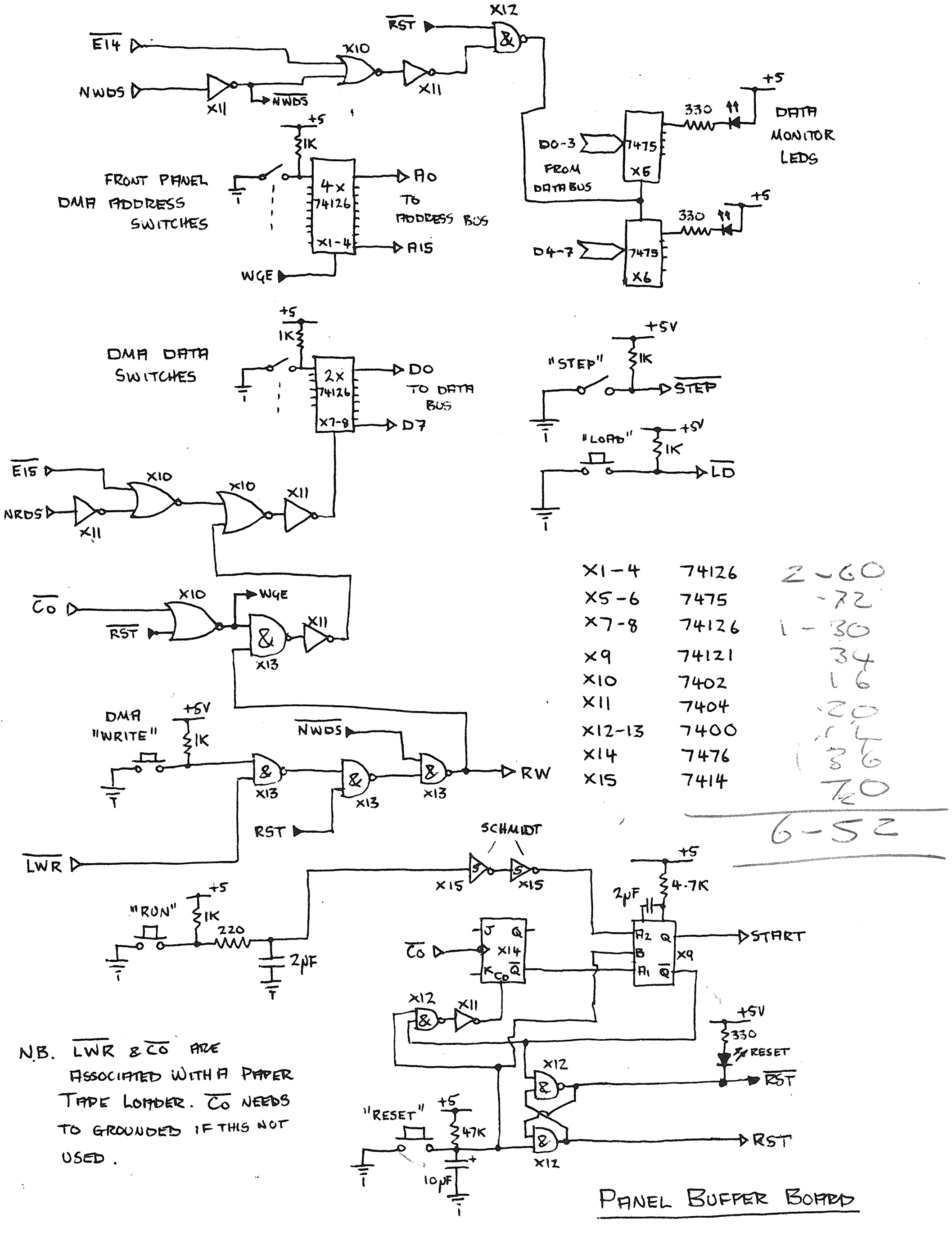
Although I hope these notes will be of interest, I do not pretend that there isn't much room for improvement. However, it does work! Further development should show that the SC/MP is more versatile than the rather limited Intro-Kit indicates.

W.G.Marshall



X1-5	74126	3-25
X6	7475	-72
X7	SC/MP	10-00
X8	74126	-65
X9-10	74154	3-00
X11	7410	3-15
X12	74126	-65
X13	7404	20
X14	74126	-65
X15	7474	36
X16-17	7420	232
		19-93

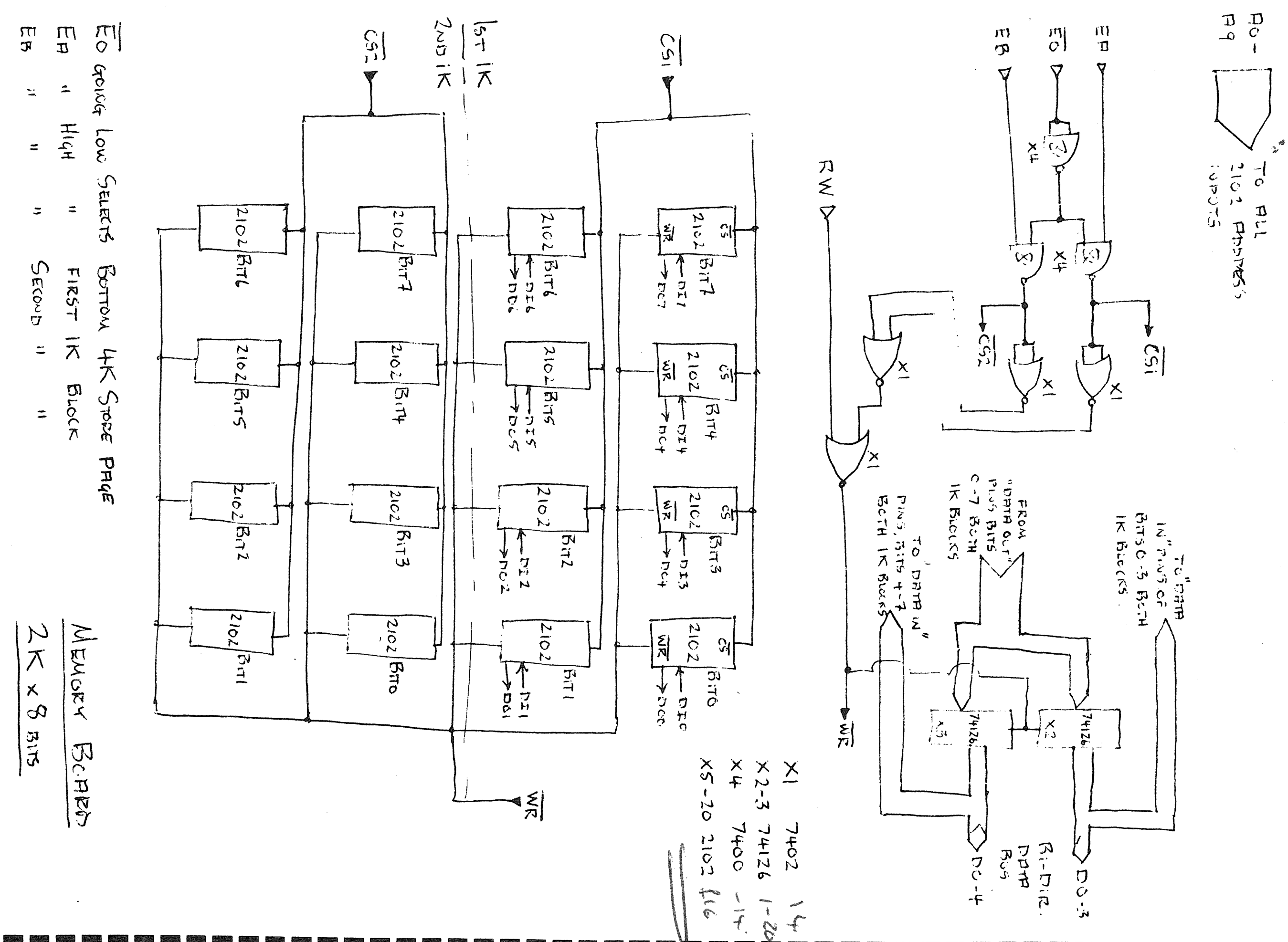
MICROPROCESSOR BOARD



X1-4	74126	2-60
X5-6	7475	-72
X7-8	74126	1-30
X9	74121	34
X10	7402	16
X11	7404	20
X12-13	7400	64
X14	7476	36
X15	7414	70
		6-52

N.B. LWR & C0 ARE ASSOCIATED WITH A PAPER TAPE LOADER. C0 NEEDS TO GROUNDED IF THIS NOT USED.

PANEL BUFFER BOARD



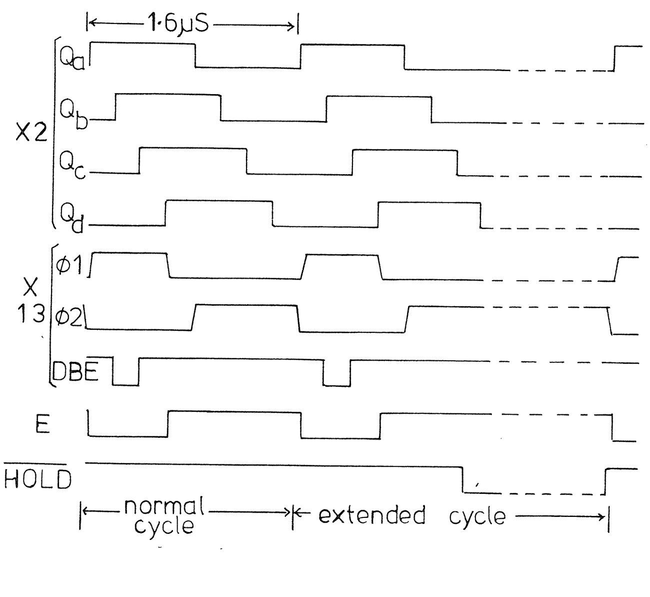
77-68 ; THE STORY SO FAR

At least three boards are working.

Some minor mistakes on the schematics in ACCN Vol 5 Iss 1;

- 'MRW' comes from X14 pin 6
- small circles should have been drawn on the inputs of X24 at the bottom of page 2 (pins 9,10,12,13)
- the data bus edge connector pin numbering given on page 3 of Iss 1 is incorrect, the correct version is on page 2

The clock circuit waveforms are as below;



SC/MP KITBUG BOOTSTRAP LOADER

T M Spencer

The program below may be typed in under KITBUG to act as a very simple bootstrap for loading a program previously dumped out on to the teletype paper tape punch as a series of 8 bit words. It uses the GECO routine in KITBUG and restores the 8th bit by reference to the GECO stack.

```
T200
0200 C4        LDI        ;load addr of GECO to P3
0201 01        01         ;reg ready for call
0202 37        XPAH P3
0203 C4        LDI
0204 85        85
0205 33        XPAL P3
0206 3F    L1: XPPC P3    ;call GECO
0207 C2        LD (P2)    ;retrieve 'MSB' marker
0208 FE        -2         ;from stack
0209 1E        RR         ;put it in MSB and combine
020A 58        ORE        ;it with rest of character
020B CD        ST @(P1)1  ;put it in memory
020C 01
020D 90        JMP L1     ;get the next character
020E F7
020F 08
XX
 ?
-M3F7
03F7 00 02  PC, where loader is put
03F8 00 00
03F9 00 02  P1, where loaded prog is to start
03FA 00 10
03FB 00 03  P2, stack pointer, invariant for your
03FC 00 EB  your machine (this one has 512 words)
03FD 00 X
 ?
-G
D4/F)K L S ....  GO, and load from tape.
```

RUNNING INSTRUCTIONS Type in loader, set registers, hit G & C/R. Start PTR on teletype on first word required. Stop reader when program in, and regain control by using RESTART to re initialise KITBUG.

WHITHER ?

FURTHER COMMENTS ON SUBJECTS IN FEB 77 ACCN

Special Interest Groups

Len Warner has practically taken the words from my mouth,but maybe the following thoughts would be of some use;

Group Classification (confirmable on minimum quorum?) How about a provisional organisation as follows;

(a) OLD TIMERS & RENOVATORS: Specific machine enthusiasts e.g.Elliott 803 fans;(dare I say it)...WBers(!)
(b) Dedicated-Applications enthusiasts;e.g. Amateur(HAM) station control; Hi-fi equipment control; Home systems (heating,burglar-alarm etc.,) control; Motoring " aids Musical systems control & generation;
(c) "Professional Amateurs" - ie those involved in some aspect of computers, professionally
(d) Small Business System-ers
(e) Aeromodellers(autopilot oriented?),Railwaymodellers
(f) Students' activities(establishing Campus Groups)
(g) Games enthusiasts:
i) Video games("Pong-ers");ii) Non-video games("LIFE"ers)
(h) Robotics; Pattern Recognition; AI
(i) Simulators & Wargamers
(j) Micro-addicts:i)Calc.-chips by manufacturer;ii)Single-chippers;iii)Bit-slicers
(k) Sci-fi watchers & Visionaries
(l) Book-collectors,non-doers,progress-watchers,dreamers
(m) Socialisers, trip-organisers, PR officers,exhibitioners
(n) Educationalists,"Do-good"ers:teaching youngsters
(o) Cheap parts organisers,Rep's

Which way to go

Graham Cobb's idea of involving ourselves in Research is, I feel, of genius. Consider just some of the advantages that come to mind;

* We are our own masters and are accountable to only ourselves.
* Profitability is not a criterion.
* No time constraint(except for exhibitions)for the solution of problems, which can therefore be iterative.
* Duplicating the efforts of other organisations is not a waste - it is valuable hands-on training for those who are involved.
* We can tap resources having the greatest potential!
  (a) We can utilise talent uncorrupted by the jaundiced views of established(and enforced!) practice
  (b) We can use modifications and redesigns of traditional and also inadequate designs(eg where they are conceived by frustrated forced-conformers)
  (c) We have the enthusiasm and ingenuity required for innovation in its most potent form! - MEMBERS! WE includes young blood which will in the future be the lifeblood of technocratic society
  (d) We do not have such potentially counter-productive manifestations of industry as Job-frustration, non-promotion, Unions, Management interference, "Jenny-bashing",Competition for funds, and having to drop near-and-dear projects,Job specification, uncooperativeness,envy,jealousy etc., to deter us
  (e) We can obtain a constructive diversity of opinion and approach to guide us with our designs
  (f) We already exist as a fully-fledged organisation!!

Which way to go(cont'd): what research

May I 1st quote John Backus of BNF & Fortran fame (as quoted in Datamation, Jan '77 Pp 142-3,146)

""I have this view that modern computers are really very strange if you look at them very closely. You can think of them as being a CPU and a store, and a narrow tube connecting them. What you're trying to do is change the contents of that store in some major and significant way. And it all takes place by sort of huffing and puffing one little word at a time through that damned little tube, which I call the von Neumann bottleneck..."

" What he wants to do, he explains, is to provide an alternative to the von Neumann machine and to programming languages that mirror it!"(Unquote)

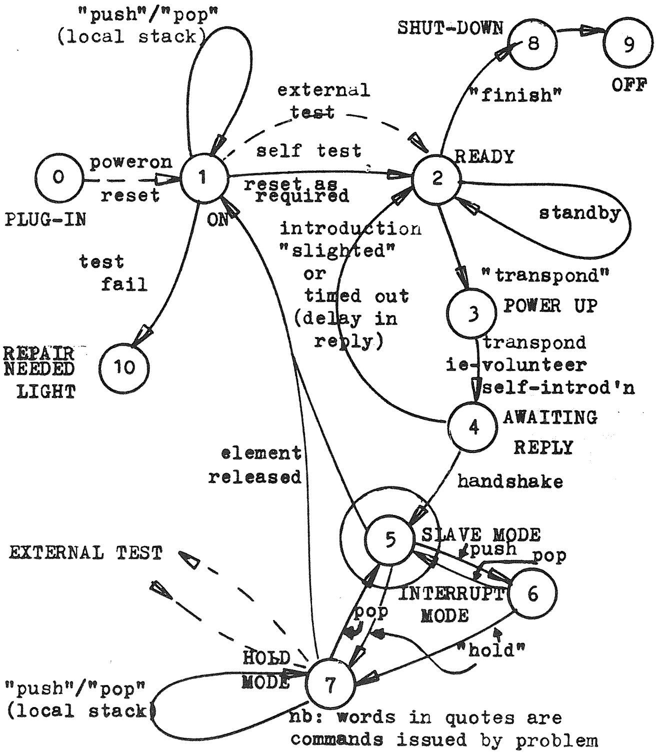
I believe that the solution I have in mind will also satisfactorily solve the "parallelism" problem pointed out by Mr. Cobb:

Organise the functional units into modules(in like fashion to analogue computers) - which bear correspondence to FLOWCHARTING-type blocks!! eg: decision-box type blocks, non-decisional sequences, index-incrementers etc.

The machine is designed to operate in semi finite-(state-) machine fashion, the various "states" in reality being the modular blocks. Each block in its turn can be logically organised as a state-machine too!

If each block is implemented as a separate entity which displays AUTONOMOUS behaviour, control links such as "Device ENABLE" can be largely dispensed with. Each available unit then must be searching for the problem! Imagine then, a given assortment of free-roving units, in a low (power)-consumption or standby mode to start with. Along comes an application, and BROADCASTS a request for the use of given units of at least a given capability (where there are units present of varying speeds etc.,). In systems where there are limited resources (within the context of the problem) if no free units are available , the lowest-priority unit of the necessary capability drops what it was doing(into a stack, much as in normal prioritised-interrupt structures), returning to its former task when the more critical job is over. I have specified autonomous behaviour ALSO so that problems and block elements may effectively meet HALFWAY, thus reducing "handshake" time; further, "garbage collection" as necessitated in implementations of LISP(to free resources) is avoided(which is not to say, however,that List Processing methodology cannot usefully be used (in hardware-implemented form, of course) to mechanise control-linkages). The control linkage structures will need to be organised so as to minimise inter-module information exchange(remember Backus' words) - ideally this would be in the form of just one pulse down a line, but since we need to exchange somewhat more information, the new-fangled multi-level("threshold"?) logic is sure to come in handy(for further information consult the M&B Monograph entitled THRESHOLD LOGIC and also - though I am not too sure that it is the same thing that is being written about - investigate the report in Practical Wireless, Jan '77 issue P772 HOTLINES entitled "Quite illogical". The former is a reasonable(both in price as well as presentation) introduction(which, unfortunately does not appear to be in print, but is available at many libraries - or can be obtained externally for you by them from the nearest British Library). I have a vague feeling that the manufacturer being referred to in the latter was in fact SIGNETICS(California) but I cannot recall where I might have read about it).

Let me try, however to illustrate my approach using state-machine format (the arrows represent stimulae to invoke change in state(each state is represented by a circle) or transformation(s) undergone in order to change state).



nb: words in quotes are commands issued by problem

State 5 (Slave Mode) is the problem-solution mode in which the resources belonging to the element are directly under the occupancy of the "PROBLEM-PATH", and algorithm control MAY ONLY be passed to an element if it is either in this or in INTERRUPT mode.

The latter, (State 6) is invoked, as in normal practice, by a higher-priority-user displacing the current user, and similarly, lower-priority users are re-invested with control when the higher-status user finishes with that element.

State 7 (Hold Mode) is used where a process (ie a user problem) is not prepared to release the element for fear of losing control over it when it is needed in a fairly short time. It is useful for "single stepping" and externally testing the results either while debugging using test-data or "on-the-fly", to hold intermediate results.

At POWER-ON (Transition 0-1) the element performs a RESET cycle in which all random patterns set up due to power transients etc.,(during switch-on) are cleared and then the element may either test itself(using a predetermined schedule) or test states/data are loaded into the element's local stack and a test sequence externally initiated, either of which drives the element to the READY(2) state, in which the element remains in a standby state until the problem announces it has finished(Transitions 2-8-9) or the problem requires attention from the element(2-3..). Here the element resPONDs to the TRANSmission - ie TRANSPONDS - with a predetermined message to the effect that it is available and has such-and-such resources available to the problem for its own use. The problem may now reply favourably (4-5) or may have found an alternative element more suitable for its needs(4-2). The latter leads to a resumption of executing "standby" states, whereas the former now enables the problem to use the element. When the problem has finished with its immediate need for the element, it either relinquishes control of it (after ALGORITHMIC control for the problem has been separately passed on to another element), or asks that the element remain as it is until it has been examined.

The above general scheme is equally applicable to most elemental structures, differing only in the details of what it does at state 5/6.

I must hasten to state that the above scheme is not for sticking to in its entirety, but merely as a guide for thought, which I am sure will be forthcoming from lots of members, so that a viable scheme is ultimately achieved.

Can I now take the opportunity of naming this new concept FLOW-WARE and the new computer-entity AGILE for:- Adaptive-Geometry Interactive Languageless* Entity?!!

* Yes! I DO mean it!! I have come to the conclusion that software, particularly LANGUAGES is the root of much disheartenment and misery and is best done without!! - After all,

"..Today in 1977, there is no theoretical way to prove programs correct...software correctness still remains the most elusive goal of Computer Science.

"As a result, software is the most unsafe, the least understood, and the most expensive component of total computer system costs...software design, development and testing is the most highly labor-intensive component of computer system products. The really useful and exciting advances in computing will probably only proceed at the same pace as advances in software engineering. And, this will be distressingly slow..." (SCIENCE, 18th Mar 77 Vol95 No. 4283 - American Assoc. for the Advancement of Science Publication - underlines mine)

(WE could change all that!!!)

My suggestion is that the FLOW÷WARE is interactively linked(using...er...light-pems, symbol call-up and actual video, graphics-wise scroll-draw-erase-modify designing!!!). In this context, actions do speak louder than words, and as many coordination/design functions for the "compilation" should be in the form of specially designed functions - KEYBOARD functions, that is to say, - as can be reasonably avoided being implemented through structures which reek of the inefficiency and complication of O/S and other software being proliferated today.

Is it too much to expect that the headlong grasp for, (once the lowlevel flow-ware is implemented) the equivalents of present-day high-level languages,be on a much more cautionary and conscious level, to prevent the present state of disarray from being repeated?

I suggest that we take, even the design exercise for the specification and implementation of flow-ware, with the right sense of thoroughness and elegance in concept. If we don't come up with the goods, our results, if proliferated nevertheless, will be the bane of many generations to come!! Perhaps the following books will help in the search of state-of-the-art concepts for incorporating into a design:

Microcircuit Learning Computers (also a Mills&Boone Monograph)
Content Addressable Parallel Processors by C.C.Foster - Van Nostrand Reinhold

I hope we can get started on this project fairly soon! I am certain most modules of the type suggested can be mechanised using presentday components already being used by ACC members - MPU chips & bit-slices - not to mention WB's!! As speed is not the prime concern at present, even the 'scampiest' of SC/MP's would fit the bill for most modules - especially since the designs involved are not clock-bound (but are asynchronous in operation). So even the most scanty homebrewed computer can be pressed into service - which means that an ACC member with the tightest of budgets could be of material help in REAL, FUNDAMENTAL RESEARCH!!

I hope a correspondence column will be opened in the wake of this letter, with the responses which members will, I hope, have - I do not think a shooting down of any or all of the above ideas will be that bad a thing either, if something eventually comes of it. Unfortunately I won't be able to involve myself in any frays for the time being - studying for exams and all that!

I may of course barge in if the going gets too exciting! Anyway, I hope any ideas will end up on ACCN so that everybody has a full transcript!

GOOD HUNTING!!
M Shiraz kaleel

WB1 & 77-68 USER GROUPS

To help share more detailed information and software, user groups have been formed for both WB-1 and 77-68. If you want to join write to;
B.M.S. 24 College Rd.,Maidenhead,Berks SL6 6BN

PUZZLES

In a club, the fat members outnumber the thin members by sixteen. Seven times the number of fat members exceeds nine times the number of thin men by thirty two. Find the number of fat and thin members in the club. (Answer on page 15)

P Rutherford

BE DISCRETE

The monstrous germanium computers are still with us, and often can be obtained for the price of the scrap metal they contain. All that is needed is a shed or barn big anough, as one ACC member (Roger Holmes) has found.

In an outbuilding at his father's farm, Roger has started installing the last working machine of the ICT 1300 series. He expects to spend four months re-commissioning it and would welcome help from other ACC members.

This particular machine has an interesting history. It was built in 1960 and spent 10 years at London University printing GCE pass-slips. Then it moved to the Galdor Centre at Surbiton, where extra instructions and peripherals were added, and now it is in Kent, and looks like going on for ever.

R. Holmes Bethersden 316

BUSSES

- CONCENTRATE ON THE MEMORY

As more and more microprocessors become available to experimenters, the desirability of having a set of micrococomputer building blocks to provide storage and I/O facilities for the processor in question increases. In particular, it would be very nice indeed if a standard memory design could be used with many different systems. Memory design is inherently an uninteresting task compared with the job of building and designing other components of micro systems and I, for one, would rather do without the job !

I was very interested, therefore, to hear at the ACC AGM that a proposal for a standard microprocessor bus has been put forward for discussion by two ACC members, Messrs. Crowe & Howden. I gather that one of them is engaged in building a 6800 based system and the other a Z80 based system, and they felt that it would be nice to share the development cost/effort of the memory cards.

There have been several proposals in the past for standard busses - perhaps the most notable are the UNIBUS (heart of the Digital Equipment Corporation's PDP-11 minicomputer family), the S-100 bus (a de facto standard resulting from the enormous popularity of the MITS Altair computers in the USA) and the MUBUS (proposed by J.Nicoud at Lausanne Polytechnic in Switzerland).

Though much thought went into the UNIBUS design (it is still in use in current PDP-11's and has hardly been changed at all since DEC produced their first PDP-11 around 1969),it is unsuitable for micro use since micros cannot exploit all the facilities. Evidence for this is provided by DEC's decision to adopt a simpler bus (the QBUS) for their LSI-11 microcomputer.

The S-100 bus happened rather than was designed and although there is an incredible number of boards (I/O,RAM,PROM) plug compatible with this bus, it has two drawbacks. First, the edge connectors used are spaced at 0.156" centres. This runs contrary to all European equipment practices and appropriate connectors are not easy to get in Eurpoe. Secondly, if a designer wishes to interface anything other than an 8080 or Z80 to the bus, he has to work quite hard to achieve compatibility. This is mainly due to MITS having oriented the bus around the 8080 and because they did not anticipate that the tremendous success of their Altair range would make the bus into a standard.

MUBUS, too, has its disadvantages, although it comes nearer the goal of a general micro bus than those mentioned so far.

Messrs. Crowe & Howland set out to define a bus based on double Eurocard board sizes and connectors which would enable not only memories but also peripheral interfaces to be plugged between systems. My criticisms of their efforts are;

1) If it is desired to be able to have a standard memory card for systems then only those signals sufficient for a memory interface should be defined.

2) The major external differences between micros arise in the way they treat I/O devices. Since most microprocessors belong to a family of chips which also include specially tailored I/O chips, I feel it is not worth while trying to achieve compatibility of I/O cards between systems.

3) The power distribution they propose is

+12V	2 tracks
+5V	2 tracks
0V	2 tracks
-5V	2 tracks
-12V	2 tracks

My experience is that whilst +12V and +5V are fairly well loaded in systems, -5V and -12V are normally only used for very low current substrate bias purposes or for providing power to a few EIA line interface chips. On the basis that each track can pass up to 4 amps, I would rather see

+12V	2 tracks	max 8A
+5V	3 tracks	max 12A
0V	4 tracks	max 16A
-5V	1 track	max 4A
-12V	1 track	max 4A

Obviously, power tracks can be made much thicker than ordinary signal tracks on the backplane, but there is also the question of providing a low impedance path via the connectors onto the board.

4) Since double-Eurocard (fairly large) is the proposed board size, might it not be easier to provide on-board voltage regulation thus cutting down on the complexity of the power supply and isolating (from the noise point of view) each board from the others more effectively ? Only the regulators for the voltages used on a board need be installed. This is the approach used on the S-100 bus.

How about the good points then ?

1) For amateurs, the ability to solder a ready made indirect connector onto a board to plug into the backplane with is very attractive since it means that boards are cheaper and easier to produce; they don't require gold-plated edge connector fingers.

2) The idea of having all the bus on one connector and leaving the other one free for the user to use if he wishes again allows savings to be made.

3) A standard printed circuit backplane could be designed and supplied to all ACC members wishing to use the bus. This would save much wiring effort and if the number of people interested was high enough, the backplane could be very cheap.

4) There could be an ACC memory board for all such systems which, if the club bought the chips in bulk, would be very cheap.

In conclusion, then, more thought has got to go into the design of the bus, but I feel that the major advantages are to do with exchanging memory cards and that the bus should only provide a memory interface and power distribution. A possible set of lines might be;

Various	power lines, as above
AO-A15	16 address lines
DO-D7	8 data lines
R/$\overline{W}$	Read/Write line
MAV	Memory Address Valid
MDV	Memory Data Valid
REFRESH	Perform refresh cycle
HOLD	Extend current cycle for slow memories
DMARQ	Request DMA
DMAGRANT	Start DMA

Tony West

COMPUTER COMICS CONTINUED

UK subscription to KILOBAUD via J. Coote, at 56 Dinsdale Ave, Wallsend, Tyne & Wear NE28 9JD, has been raised to £11 for 2nd class sea-mail for one year. Back issues are £2.35.

CREATIVE COMPUTING mag, and some books, may be obtained via Creative Computing, 60 Porchester Rd., Southampton S02 7JD

MECHANICAL MACHINES

It seems that a reaction is setting in against the use of anonymous LSI chips , some constructors have been heard to say that they like projects with lots of wiring, one microcomputer system is even advertised as being boring. So, for chassis-bashers and other bottom-up constructors, the following two items may be of interest;

MECCANO have a design for a Differential Analyser available; their 'Supermodel Booklet No. 4' at 80p + 15p P&P from M.W.Models, 165 Reading Rd., Henley-on Thames, RG9 1DP.

A circuit for a small relay 'computer' is printed in the book 'Digital Computers, a Practical Approach' second edition 1969, by Marchant & Pegg, published by Blackie.

HELP

PLESSEY CORE & IBM I/O

Can I get in touch with someone having information about a Plessey core store type 68/1339 (13 bits) and an IBM 735 I/O writer. I have these items but am in need of details about the necessary drive circuits.
J Larsen Valmuevej 12, Breum, 7870 Roslev, DENMARK

FRIDEN F1Ø

We were recently given a Friden F10 Flexowriter, but although the mechanics are working, the punch (except parity bit relay) and sense sections seem to be dead (although the bulb seems to be at a fairly high potential both sides relative to earth). I wonder if you know where I could get hold of some documentation e.g. User Manual or circuit diagram /service manual.

The Flexowriter came without a PT reader, could this have anything to do with the problem ?
Simon Garth 67 De Parys Ave.,Bedford MK40 2TR

WATSIT

I have a 'Viatron Robot Printer Board No 109801 PCC2. It contains some rather interesting IC's, which I cannot identify. Can anyone help ?
R.G.Cogliatti, 6 Woodcote Lane, Purley, CR2 3HA
tel; 01 660 1880

WHICH GOLFBALL ?

I have recently obtained an IBM golfball I/O typrwriter and wonder if anyone can help me with the following; how do you identify what model you have, I can see no easily spottable identification. The machine has the motor missing, are these available anywhere in this country, also circuit diagrams ?. Finally, has anyone managed to successfully interface one of these machines with a microprocessor ? Any information at all would be gratefully received.
A W Nicoll, 257 Lutterworth Rd., Nuneaton,CV11 6PU

TRADE WITH FINLAND

I would like to find a member who could take the trouble to send me future issues of ETI as they are published. I will pay cover price and postage and furnish him (her) with envelopes and mailing labels so the trouble shouldn't be too great.
K Soderstrom, Gammelbackantie 3E33, SF-06100
Porvoo 10, FINLAND

MACHINE'S MATE

It has been my long-standing ambition to play chess against a computer, but to date have not been able to locate or obtain a computer chess program.

I would be most grateful if you could let me know of any implementations (high or low level language) which are available to ACC members.
R Grant 514 Finchley Rd., London NW11 8DD

CREED SWITCH

We've a couple of old Creed 5 channel TT, but I can't get them to work, in spite of having a manual from Creed, and unfortunately the people who originally worked on these machines seem to have retired from Creed.

My problem seems to be the Mark-Space ratio switch on the front. Can anybody tell me what the adjustment should be ?
P Foster 10 Benson Close, Lichfield, Staffs

SHOP

Note; This section is included for the general interest of ACC members, and does not imply any approval by the ACC of the goods or services offered.

A BETTER EXCHANGE RATE

Interam Computer Systems is a new company with the stated aim of selling goods in the UK at the lowes feasible price. As a special introductory offer to all members, Interam are offering the following manufacturers' equipment at an exchange rate of $1.7/£1 including import duty but excluding freight and 8% VAT;

Cromenco Inc. Technico Inc.
HAL Communications Corp. ECD Corp.

They also hope to sell the following manufacturers equipment under the same terms;

Ohio Scientific Instruments
Apple Computer Inc
IMSAI Manufacturing Corp
Processor Technology Corp
Technical Design Labs

For further imformation contact John Lagan at 59 Moreton St., Pimlico, London SW1, tel 01 638 6200 ext 136 or 01-834 0261 after 5.30 weekdays or any time weekends.

MONITORS & MODULATORS

For the DIY VDU enthusiast, UHF modulators are available from Maplin & GDS (Sales) Ltd., A range of video monitors at around £200 is made by Bluebest Electronics Ltd., 18 Longleat Gardens, Maidenhead, Berks.

FOR SALE

Ampex TM4 tape deck. ½" tape. 19" rack mounting. With most of the control electronics and power supplies. Buyer collects.

ICL card verifier mechanism, works OK, buyer collects.

Oil filled transformer, suitable core store PSU. 35 - 0 - 35 V @ approx 8A. Buyer collects.

Offers for the above to Bob Selby,
01 751 2872 between 6pm & 7 pm.

BEAR MICROCOMPUTER SYSTEMS

Offer discounts to ACC members on certain items;

Ref	DC2	77-68 design manual	£4.50
	DC2A	77-68 spare diagrams set	£1.00
	HC1	77-68 PC Board	£7.75
	HC100	25 way submin D type plug	£0.70
	HC101	25 way submin D type socket	£1.00
	HC102	Cover for 25 way D type	£0.50

SAE for complete catalogue
B.M.S. 24 College Rd.,Maidenhead,Berks SL6 6BN
tel 0628 29073

10% REDUCTION TO ACC MEMBERS

COMART (Computer Mail Order & Retail) Ltd. have taken on the UK representation of IMSAI products including their 8080 microcomputer system and the 8048 control microcomputer. They will also soon be announcing representation of other Altair/Imsai S-100 bus compatible products including low cost peripherals and memories.

COMART have decided to offer a discount of 10% to ACC members for pre-paid orders, and are interested in hearing from any members who own IMSAI processors and would like to start a user group.

For further information contact David A Broad
39 Gordon Rd., Little Paxton, Cambs PE19 4NJ
tel Hunt (0480) 74356

FOR SALE

One only, 128 x 8 static RAM 450nS Motorola MCM6810AL ceramic package. I used it in my 6800 system but it is now surplus to my requirements. Maker's price is £6-71, but my price is only £4.50. Supplied in conducting plastic foam.
T.G.Gardner 46 Woodcote Green Rd.,Epsom,Surrey

SURPLUS ELECTRONICS

May I recommend a surplus electronics shop at 230 High St., Harlington, Middx. Ian Cullen, who recently opened the business, has a wide range of bits and pieces. Occasionally he breaks up computer units. If you live in the area or are visiting he is worth a call. Normal shop opening hours, but he is closed on Mondays & Wednesdays.

Bob Warren

FOR SALE

19" rack, 30A thermal cut-out, 5,10,20V PU,7 track Pertec tape drive (not working), 7 hole FRIDEN, 7/8" & 1" paper tape, set of ASR33 documentation.

WANTED

Small, working, disc drive - anything considered, microfloppy upwards.

V.B.Coen 313 Kingston Rd.,Ilford,Essex IG1 1PJ

PUZZLE ANSWER

56 fat & 40 thin members.

LETTERS

TREKFILE by 'Surak' (RJB)

First, I must apologise to Andy and Stuart at Galdor for giving the wrong impression. Of course, my letter was written several months ago, and at the time MAC was the only high level language available. The alternatives were an assembler called MPL3, and something called Rapidwrite which, being a form of COBOL, is not what I'd call 'High Level'. Of course, since then they have got their 1900 going, and I have (thankfully) gone back to FORTRAN programming. Incidentally, I was amused by their statement that they have Manchester, not Mercury, Autocode. What's the difference ? Manchester/Mercury Autocode is so called because it was developed at Manchester University. Both names are equally 'correct'.

Secondly, a couple of corrections for printing mistakes in my 'Fortran' article. The line "The string is then output..." should begin a new paragraph, as of course the string is output regardless of what character was first. "IN" format would probably be better as (${}_2$NUMBER). Perhaps the most serious errors, however, are the two in the 'CTRL' section. 'Delete last line' should read 'Delete last character';'delete life' should, of course, read 'delete line'. There are also a few minor errors in the listing, but they should be easy to spot and correct. In fact, I believe that a lot of the errors were due to me rather than Mike Lord, or whoever transcribed the article.

Talking of articles, it would seem that a lot of my stuff is getting into the ACCN. This is probably indicative that the volume of 'other' contributions is not as high as it could be. So, if you have something worthwhile to say, don't waste time - get pen/typewriter to paper and follow my example.It's your newsletter, after all.

Finally, for the Trekkies amongst us, a few clubs;

STAR TREK Action Group (STAG) - Janet Quarton, 15 Letter Drive, Cairnboon Lochgilpheal,Argyll,SCOTLAND (£1- p.a.)

STAR TREK Welcomittee, UK as above

STAR TREK Welcomittee - Jerri Franz, Foreign D.C. Rt.1, Box 79A, White Post, VA. 22663, USA. Enclose a self-addressed airmail envelope and an International Reply Coupon for info.

STERB - John Hind, 14 Bingham Rd.,Radcliffe-on-Trent Notts (£1-)

BEYOND ANTARES - Sheila Hull, 49 Southampton Rd., Far Catton, Northants (£1-)

EMPATHY - Catherine Owens, 30 Overden Way, Halifax, Yorks (£1-)

STAR TREK Correspondence Club - Jackie Dunham, 105 Sommerleighton Gardens, Norwich (75p)

I apologise for any spelling mistakes that may have crept/galloped into that list, as most of it was handwritten i.e. hardly legible.

Keep on Trekking - Live Long and Prosper.

Z80 GROUP

Without wishing to tread on Neil Harrison's (or anyone elses) toes, I'm interested in forming a very informal user group for the Z80 (MK3880). Just to exchange ideas and chat over problems basically.

Roger Sinden 18 Percival Rd.,East Sheen SW14 7QE
tel 01 878 5374

SOME PROJECTS

Now my 6800 is working I can start to take an interest in other things. For instance, since I need an interface for cassette, 7 track tape and PTR & PTP plus, I hope, a disc unit, I decided to standardise on five cards which can be permuted to interface to anything. Since I have the facilities at home to silk-screen print & etch PCB's this is the easiest approach. If members are interested I'll write a description and print boards for anyone else interested.

EBL are selling Cossor 402-2A & 401-2 VDU's (2000 char). My teletype is a 401 (The original lab prototype !) with home built timing and serial interfacing. Again, since I have solved the problems involved, I can offer my solution to other members. It works well !

Since I work for Cossor, I can probably help others with information on these VDU's.

Projects;

Hardware; Standard Interface for Anything

1) One control card, 1 per system.
   2 PIA's handling all control signals,decoding selects for up to 16 devices, 6-10 chips,can drive several at once with care.
2) Read-Write card, 1 for cassette, 7+ for tape, disc, drum etc. 4x741, 4-6 other chips.
3) Timer card, 1 per device, osc., PLL, gating, 5 chips.
4) Buffer card 2-6 chips, latches, tri-state buffers.
5) Serial/Parallel converter, 4 chips.

For instance, 7 track tape would require 7 x (2), 1 x (3), 1 x (4)

Punch/reader would use only 1 x (4)

Software;

In the long term, I'm interested in software for a voice coder program, book reader, and any form of pattern recognition. Given long enough, I'll write it myself, I'm a reasonable self-taught programmer. Short term, I need an assembler and a FORTRAN compiler.

Since I intend to move from 6800 to multiple 9900, any programs I write other than device handlers will be in a standard format, FORTRAN or Assembler Symbolic.

I suggest that contributors to Software Libraries make their masterpieces relocatable. This is particularly easy with the 6800 because of all the relative addressing instructions, and avoids their clashing

with anything else - indeed it permits loading more than one at a time, and using combinations.
What we need is a relocating loader and monitor. Any offers ?

I have noticed that contributors are either Hadrware men who write simple programs, or Software men who write complex ones (in different languages)
If we can standardise an interface - I've suggested one, not necessarily THE one, can we then as a group standardise cross-assemblers so that we can allwrite the same languages ?. This would facilitate group software and hardware projects and let us all head in the same general direction.

Dave Morton

SOFTWARE SURVEY WANTED

A suggestion you might like to include in the ACCN is that someone with the info does a software survey for the popular MPU's stating where it can be obtained and the price. This would make life easier for people selecting a MPU chip.
Eric Singleton

ALGORITHMIC LIBRARY

I would be willing to organise an algorithmic library (of flow charts as distinct from coded programs). I could liaise with the software libraries WB, 8080, 6800 etc., who could arrange coding (and testins) in their particular instruction set.
T F Hobson 44 Milton Ave.,Eaton Ford, Huntingdon, Cambs PE19 3LE tel Huntingdon 74090

RADIO DATA LINK

The Amateur Radio Licence has now been changed to permit data transmission on VHF and up.
I would like to suggest that a standard is made to be the same as is commercially used and/or the same as the PO modems use at 600/1200 baud. This would enable similar equipment to be used through acoustic couplers on telephone lines and also allow surplus equipment to be pressed into service.
P C Staton G4FXY

LANGUAGE

The really good programming languages cannot be attributed to one person, nor are they promoted by a single organisation. In this context I am referring to FORTRAN, COBOL, Algol 60 and Algol 68.
It is noteworthy that of the four good languages, three are in widespread use. Algol 60, despite its lack of transput definition, has attained working status on most mainframe computers; it represented a leap forward in the design of high-level programming languages just as FORTRAN did over assembler languages.
Algol 68 represents a similar leap forward in the specification of programming languages, and hence in their design. Unlike Algol 60, Algol 68 has transput fully defined, and incorporates facilities for parallel processing. To the commercial computer user, the design of programming languages may seem far removed from the problem of updating the purchase ledger online, but were it not for the efforts of computer scientists and mathematicians who seek to systemise the design of programming languages, he would not be able to reap the benefits of Pascal, Algol W, PL/1, APL etc. etc.
A programming language differs from a human language in important respects, not the least of which is that it is used to communicate with a logic machine instead of another human being (we are not machines). Hence, a programming language gains in power from explicit references to values, while conversely, a human language gains from oblique references; although a programming language may be used to write poetry, the programs themselves are not poetry except insofar as we perceive beauty in their form.
One cannot do justice to a good programming language by excerpts from theoretical programs. Only a systematic perusal of such languages coupled with actual programming experience will help one to appreciate why these languages are so widely used. For the benefit of those who do not agree with this assertion, I append a fragment of a working program in Algol 68-R, which constructs two linked lists.
The program analyses the usage of a suite of 40-odd programs. On each occasion one of these programs is used, the names of the user and the program are appended to a file. After creating lists of user and program names, each name being unique in the list and each user referencing at least one program, and vice versa, the program sorts the lists into alphabetical order, and then prints the suite analysis. To avoid sorting strings, the lists involve creation of references to strings, giving rise to a tag-sort, which is faster.

Since the number of users and program names is not known, even at run time, the program creates flexible lists which are expanded as necessary.

```
¢ Fragment of plusanal ¢
mode unit = struct ( string s,
                     [1:Øflex]ref unit ru,int t);
[1:Øflex]ref unit
     program list:= loc unit := ("",nil,Ø),
     userlist := loc unit := ("",nil,Ø);
(ru of program list[1]):= userlist[1];
(ru of userlist[1]):= programlist[1];
op += (ref[] ref unit a, ref unit b);
     (int up = upb a; [1: up +1]ref unit uu;
      uu[:up]:= a; uu[up + 1]:= b; a:=uu);

proc table =(ref[]ref unit u,ref string s)int:
    (int p,q :=1;
    for i from 2 to p:= upb u
    while s of u[i] ≠ s do q:=i;
    ¢ to see whether we have already met s ¢
    if q ≠ p then q plus 1
    else u + (unit := (s,u[1],Ø));
    ¢ s is a newcomer, so a copy has been made on
      the heap and u has been enlarged to contain
      its name  ¢
       q := p+1
    fi ) comment procedure delivers index of s in
                 u comment ;

proc refer = (ref ref unit u,v):
    (¢ check if u refers to v. If not, enlarge ru
     of u and add 1 to t of u ¢
    int m :=1,n := upb (ru of u);
    for i to n
    while s of (ru of n)[i]≠ s of v do m:=i;
    if m=n then (ru of u)+(unit:=v)fi;
    t of u plus 1);

proc printlist=(ref[]ref unit u)int:
    ¢ delivers total of all t in list ¢
    (int j := Ø;
    for i from 2 to upb u do
      (ref ref unit rru=u[i];
      outf(standout,$(11kg,29k3uv,41k3uv$,
        (s of rru,t of rru,upb(ru of rru)-1));
      j plus t of rru);
    j);
end ¢ of fragment ¢
```

If you are interested in any of the good programming languages, Professor D.W.Barron in his 'An Introduction to the Study of Programming Languages' (CUP 1977) gives a useful bibliography. See also Pagan, Frank C :'A Practical Guide to Algol 68' (Wiley 1976).

J Hamilton

AMATEUR COMPUTER CLUB NEWSLETTER

Vol 5 Iss 2 June 1977

Editor; Mike Lord
7 Dordells, Basildon, Essex
0268 411125