AMATEUR COMPUTER CLUB NEWSLETTER

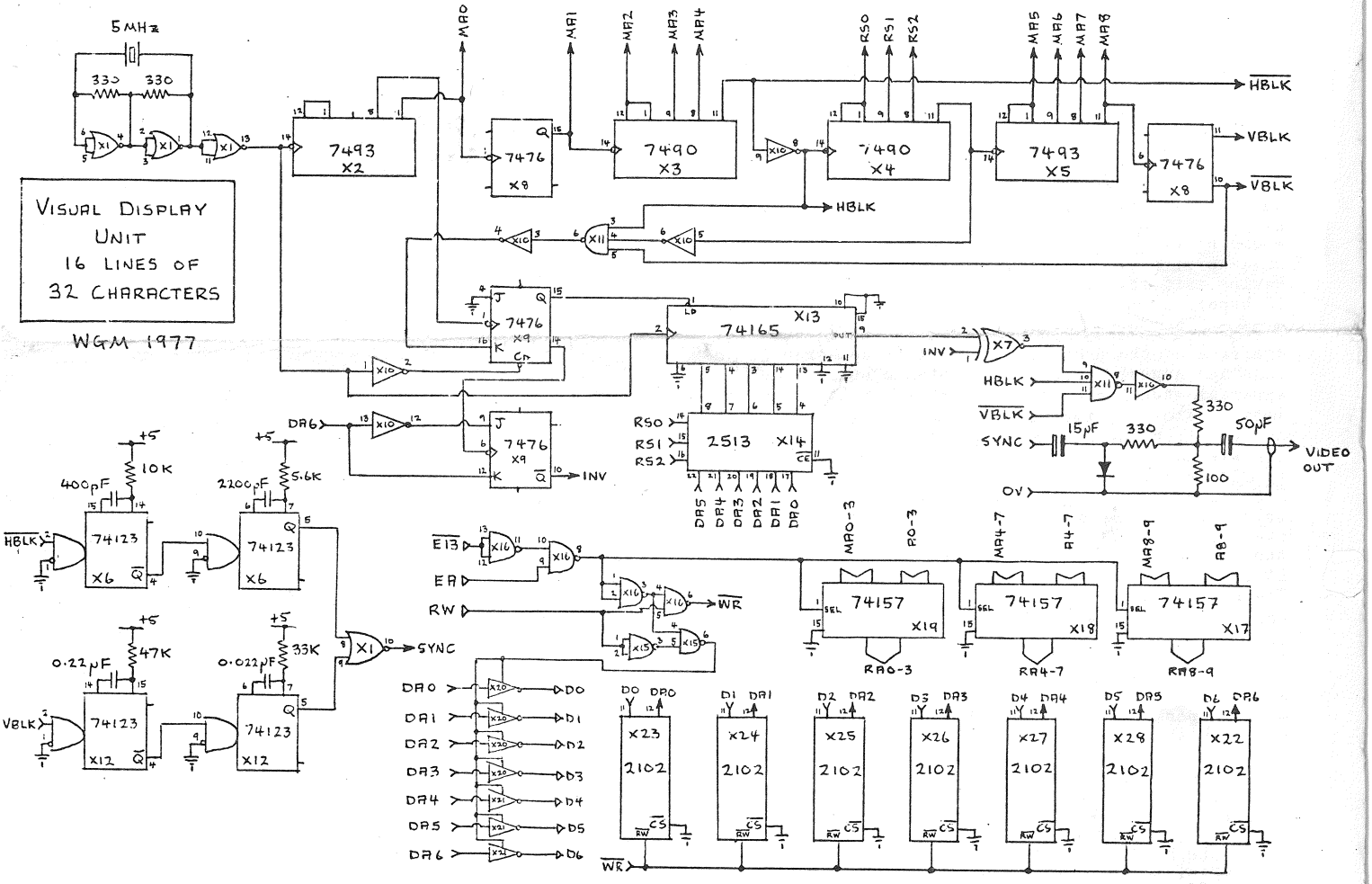
VOLUME 5 ISSUE 4 OCT 1977

~~SIMPLE~~ SC/MP'R DISPLAY

A VDU FOR SUPER SCAMP

Timing and Character Generation

The basic timing chain is adapted from a design by Jean-Daniel Nicould published recently. It consists of a string of counters, and produces the blanking pulses VBLK and HBLK. Monostables are used to produce the sync pulses within the blanking intervals. The outputs of these monostables are OR-ed to yield a combined signal SYNC.



X1	7402	X2	7493	X3-4	7490	X12	74123	X13	74165	X14	2513
X5	7493	X6	74123	X7	7486	X15-16	7400	X17-19	74157	X20-21	74125
X8-9	7476	X10	7404	X11	7410	X22-28	2102				

The character generator used is the widely available 2513 which provides the basic 64 character ASCII set. A 74165 converts the parallel output of the generator to the serial video signal and this, together with the blanking and sync signals, pass to the simple video mixer shown. The facility to invert the video signal for any character(s) has been added. One of the RAMs X22 holds the information which determines whether a character appears as white on black (normal) or black on white (inverted). This yields a useful cursor facility. However if the cursor is on the last character in a row, then the whole row will be underlined by two white lines. This is because the invert signal is latched by X9. Further logic would be required to remove this effect, but I find it to be a useful end-of-line warning. That's my excuse anyway.

Memory Access

Six 2102 1k RAMs hold the 6-bit ASCII code, the seventh holding the video invert information for each of the 1024 stored characters. Only 512 of these are displayed on the screen in 16 rows of 32

characters, so in order to view the second 'page' signal MA9 is connected to one of the spare keys on my Clare-Pendar keyboard. Holding this key down causes the second half of the VDU store to be displayed.
The RAMs are treated as part of the main store by the processor, and have addresses D000-D3FF. To write into the VDU store, the processor addresses a location in this range, and thus causes $\overline{E13}$ to go low and EA to go high. The address multiplexers switch control from the VDU to the processor address bus A0-9. The read/write signal RW going high takes the $\overline{RW}$ inputs of the RAMs low so that data on the system bus D0-6 is written into the desired location. The duration of the VDU loss of control is very short and the odd blips on the screen are insignificant. In a similar way the processor can read from the VDU store, except that this time RW is low and the tri-state buffers X20,21 allow RAM data onto the system data bus. If desired an additional RAM can be added so that the eighth bit is available if it is required to use the VDU RAMs as main store with the display switched off.

Setting Up

The values given for the monostable timing components are approximate and may need adjusting. However TVs seem to tolerate a wide variation in pulse widths. Likewise the 100 ohm resistor in the video mixer may need adjusting to get a stable picture. At the moment the modulator used is a home-made effort, but as this is not producing entirely satisfactory results, I may exchange it for a commercial product.

Conclusion

This is essentially a simple design with few frills, but it does give a good display in spite of there being no interlacing and no equalising pulses. I would be interested to hear from anyone who could come up with an equivalent display unit any cheaper! Having said that a host of better ideas will flow in (I hope).

I must now give some corrections to the circuit diagrams of the processor board that appeared in the June and August issues. The mistakes only became apparent when I read Mr.J.Knight's criticism in the latter issue. In the June article, on the diagram of the microprocessor board, the $\overline{E0}$-$\overline{E15}$ and $\overline{E16}$-$\overline{E31}$ signals have become transposed. Address bits A8-11 should be decoded by X9 to form enable signals $\overline{E16}$-$\overline{E31}$ and A12-15 are decoded by X10 to form page select signals $\overline{E0}$-$\overline{E15}$. This error has been carried over to the diagram of the microprocessor board in the August issue. Note also that the load signal LD should in fact go to X10 and not X9 as shown. Thanks Mr. Knight ! Please accept my apologies for any confusion generated.
On a general note, I would like to take up Mr. J Knight's point about a lack of IO. Firstly this is a a developing system and who knows what may yet be added !. Secondly, I am trying to keep costs down as much as possible and I don't want to provide hardware for a lot of peripherals I may never obtain. Does the home user, for instance, have any need for an elaborate interrupt system when only one operator is likely to use the machine at any one time ? Special requirements for interrupts from some peripherals can easily be added later, if desired, by using the SC/MP SENSA input.
Meanwhile, I like the idea of forming a SC/MP user group. So much bumph is produced about the 6800 and 8080, SC/MP material seems to be thin on the ground (outside the ACC newsletter).

Bill Marshall

'A GUIDE TO SC/MP PROGRAMMING' is a well written book which was produced by Kemitron to support Crofton's micro system, and is available separately for £3.50 from Kemitron Division of Drury Bros. (St Albans) Ltd., Unit 6, Beaumont Works,Hedley Rd., St Albans, Herts. tel St Albans 65094

SHOP

FOR SALE

Ex Language Lab solenoid controlled tape recorders, 2 R/W amps fully remote £10.0

Set of PDP8 cards (possibly complete, no memory)£30.

ICL punch verifier keyboards. New £10

1702A PROMs £4.00. Erasing & Programming service available.

DIDS 402-2A VDU's do not need control unit £125, plus copies of manual at cost.

300/600 baud cassette interface Kansas/Cuts built and tested £25.
All items plus P&P or carr by arrangement.
Dane Services. 2 Lupin Close,Hinckley,Leics LE10 2UJ tel 0455 35621 (ev + w'ends)

FOR SALE
One set (3 chips) of Intersil PROMs containing the control panel routines for the IM6100. As in Intersil Application note M006. Unused. £15

R.G.Cogliatti 6 Woodcote Lane,Purley,Surrey CR2 3HA 01 660 1880

Z80 COMPONENTS

Available to ACC members at a discount. Send 9x4" SAE. W.J.Whitehouse Flat 2, Heam Castle, Saundersfoot, Dyfed SA69 9AL

FOR SALE
An old POTTER mag tape transport (half inch), in working order except that there are no read/write electronics apart from the head. £15 ono.
E.R.Tippelt 206 Hammersmith Grove,London W6 7HG
01 743 6811

FOR SALE

ICL Equipment

10 off 8Mch. (6 bit) Disc Drives (CDC EDS 8) £90 each ONO.
3 off Controllers for above (max 8 drives) £90 each ONO (spares available)
1 off 300 lpm 120 col Printer (single phase) £50
2 off 1901 16K processor (single phase) £150 each ONO.
END
We are trying to make some space to put the next lot in.
A.M.Keene The Galdor Company, 52 Brighton Rd., Surbiton, Surrey KT6 5PL tel 01 399 1300

LOW PRICES FOR ACC MEMBERS
From Bear Microcomputer Systems Ltd., 24 College Rd., Maidenhead, Berks for;

DC1 WB-1 Design Manual	£3.80	0%VAT
DC2 77-68 Design Manual	£4.50	0%VAT
HC1 PCB for 77-68 CPU	£7.75	8%VAT

2102-1 £1.50, 2102L-1 £1.60, 6800P £14.25 8%VAT

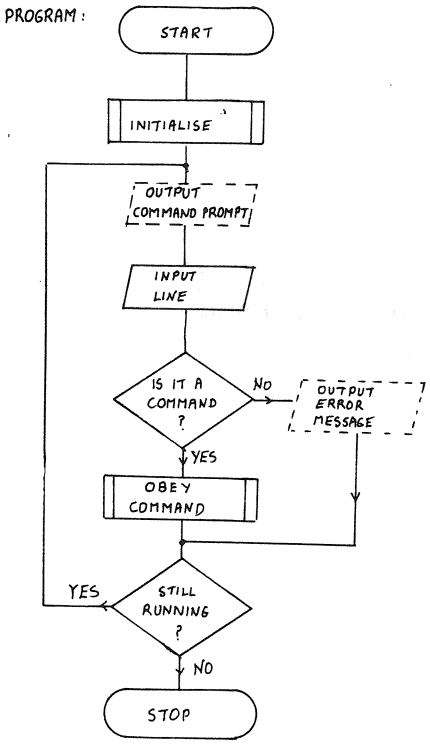
Plus Molex connectors for S-50 bus. Full range 74LS logic. P&P 30p. tel 0628 29073

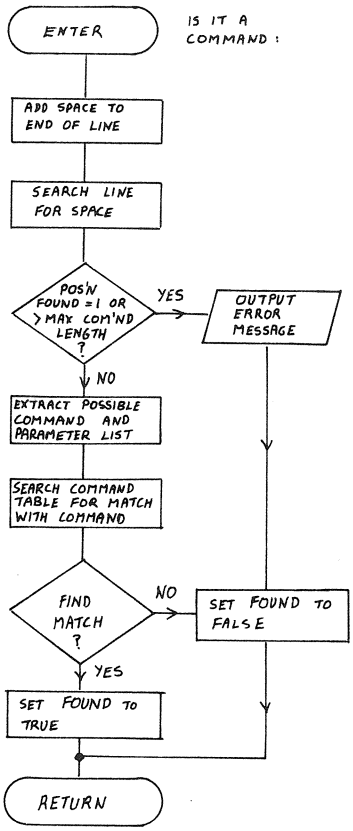
WANTED
5V power supply at 10A or more wanted. 240V in preferred though anything considered. Cash waiting to placate my Z80 system's appetite for amps !
Neil Harrison 15 Hill Rd., Watchfield,Swindon, Wilts.. (0793) 782551 x 289 (working hours).

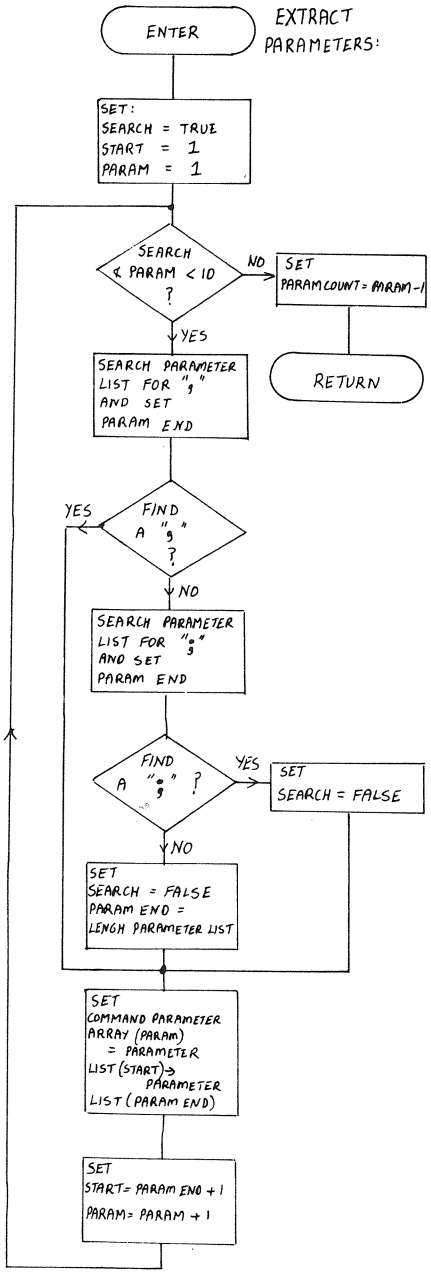
FOR SALE
SCMP Introkit & Keyboard fully built with edge connectors - £100.
P.A.Gibson Daw, 479 Wellingborough Rd., Abington Park,Northampton NN3 3HN tel 714821

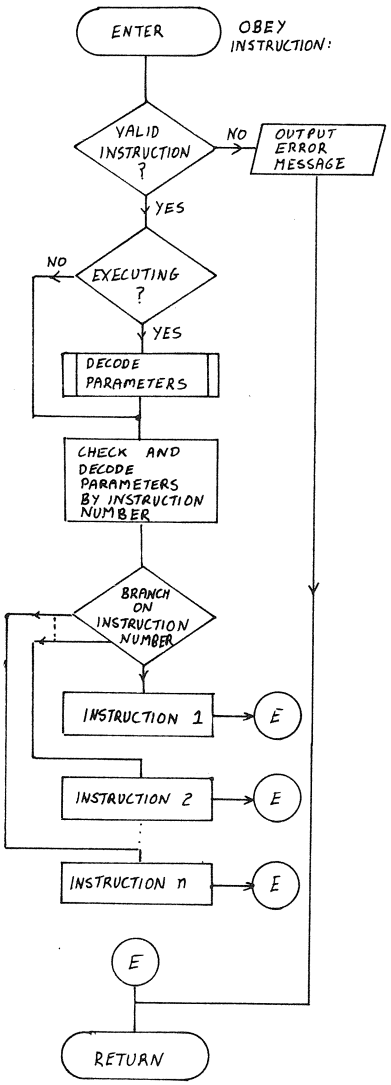
EMULATOR

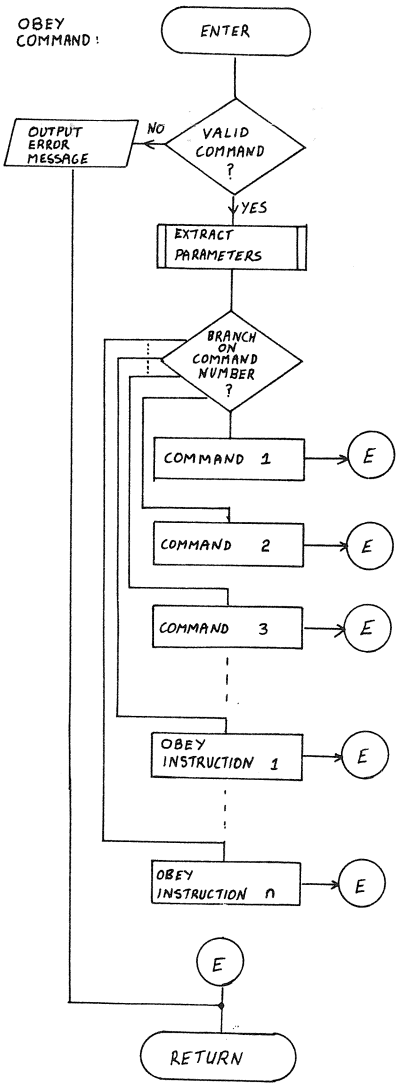
Software enabling the use of high level languages or indeed any language other than machine code falls into two types; compilers and interpreters. The latter include such things as monitors, executives, operating systems, and interactive BASIC facilities. My particular interests are executives and emulators and I include an outline flowchart of a working version of one of the latter.

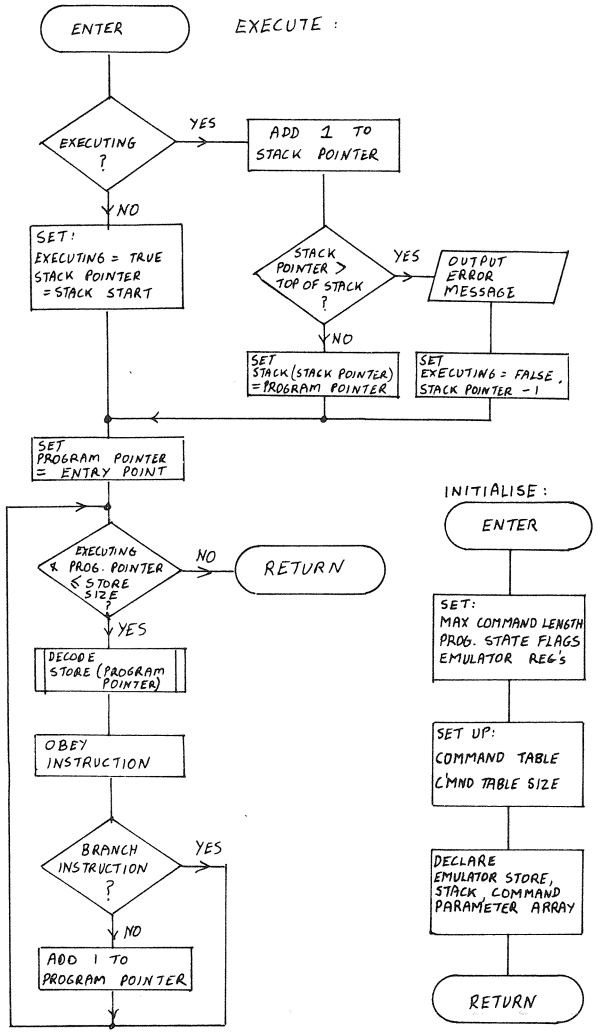












This flowchart may seem strange in places as it was drawn from a working BASIC PLUS program. As it is written, on starting, the program clears an area of memory (the emulator store, 256 16 bit locations), sets a stack pointer to the start of the stack (25 16 bit locations) and sets up the table of command (and instruction names). A line is then read with the presently required format;

⟨command prompt⟩ ⟨command⟩ space ⟨parameter list⟩

⟨parameter list⟩ consists of;

⟨parameter⟩ ,⟨parameter⟩... ⟨end of line⟩ or ;

This format allows for comments after the ; if it is used to terminate a line. Command prompt is optional and consists of '←', instructing the user to type something.

The commands display the contents of the store and generally carry out the functions of a monitor, e.g. PRINTREG prints the values in the registers, INITIALIZE clears the store and resets the registers, STOP halts the entire program.

Two other commands involve storing representations of instructions in the emulator store (SEGMENT converts a relocatable pseudo-assembler code and stores it one instruction and operand(s) per a 16 bit location) and printing out these instructions (DUMP prints out locations as an integer value and the appropriate mnemonic (by indexing the command table with the instruction code) followed by the operand(s)). The commands can be tailored to be precisely equivalent to the facilities provided by a front panel and monitor.

The instructions are at present the machine code of an imaginary computer but could be easily set up to be precisely equivalent to the machine code of almost any micro (at present programs are started in the emulator store by using an ENTER SUBROUTINE to the first or start location), also the facility is given of 'single stepping' instructions on the keyboard for immediate execution e.g. ADDN 1 would add 1 to the accumulator, GOTO's however have no effect. The programming of the ENTER SUBROUTINE (EXECUTE) instruction is thus complex as it carries out two functions.

Extensions to this system would allow it to fully simulate a MPU and possibly to dump tested code to paper tape, or cassette if available, to load on the real computer. A further possibility would be to write an assembler and/or compiler which would produce code which could be debugged using the emulator.

If anyone is interested, this program has been written in FORTRAN II, algol 68, CORAL 66 and finally BASIC PLUS (IBM 1130, ICL 1903 and PDP11/40 machines).

R.O.McLean 24 Barsby Dr.,Loughborough, Leics

HELP

HELP

Would anyone who has managed to get the system 68 VDU as recently published in ETI to work properly please contact me before I go completely round the bend. I have built a 7768 and it worked first time and would like to link it to the above VDU. Any information about linking to this VDU or the 560VDU would also be welcomed.
Mike Alger, The John Birbeck High School,Keeling St., North Somercotes, Louth LN11 7PN.

HELP

Any information available about a Univac sprocket feed printer No 2521103 type 0769-23 with printer drive board 2531028 would be much appreciated.
Bob Kellock,New Mills, Tisbury,Wilts (074 787 318).

HELP

A few months ago I described some Plessey 16k x 13 bit core store I bought from Servo & Electronic Sales. Quite a few people contacted me about getting data from from Plessey, Towcester.
Unfortunately my enquiries, both through 'official' and 'un-official' channels, have been unable to unearth anything more than a few drawings showing the outside of the core stack. Apparently the required data no longer exists. If anyone knows any other method of getting such core stores to work, I would be very interested to hear from them - and so would a number of other ACC members.
R.G.Cogliatti, 6 Woodcote Lane,Purley,Surrey CR2 3HA 01 660 1880

LOCAL GROUPS

NORTH WEST GROUP OF ACC

Three well attended and very successful meetings have been held to date in the centre of Manchester.

The dates for the next 4 meetings are;
Thursday 3rd November.
Thursday 15th December.
Thursday 12th January (provisional)
Thursday 2nd February (provisional)

The vanue for these meetings will be The National Computing Centre Ltd.,Oxford Rd.,Manchester 1. Members should arrive at 6.45 for a prompt 7.00 start. (No admittance after 7.30).

As the dates are not completely finalised, members should check with Dave Wade (061 980 2755 home, 061 236 9432 ext 211 work) or Ken Horton (061 799 0192 home, 061 228 6333 ext 371/2 work).

NORTHAMPTON GROUP ?

Anyone in the area interested in setting up a Computer Users Group in Northampton please contact P.A.Gibson Daw, 479 Wellingborough Rd., Abington Park, Northampton NN3 3HN tel 714821

ACC MIDLANDS GROUP

The meeting on 4th Sep was to have been a 7768 special, but although we had two 7768's on show, neither had their MPU fitted due to supply problems. However John Diamond showed a Hex keyboard he had set up to act as an input to his 7768. When the system is up & running we will publish the details. The next meeting will be on Sunday 13th November. Contact Roy Diamond, 27 Loweswater Rd.,Coventry, tel 454061

TREKFILE — Surak

News on the S-T fron this month;
1) The 'Great Bird of the Galaxy' otherwise known to us mortals as Gene Roddenberry, has begun filming a new series.
2) Galdor have a copy of the ICL FORTRAN version of THE game - but not a large enough memory, or the right compiler. (Pity).
3) A few club addresses

STAG: Janet Quarton, 15 Letter Daill,Cairnbaan, Lochgilphead,Argyll,Scotland (Dues; UK£1.50, Europe £2.50, USA $6 airmail, $4 surface mail).

BEYOND ANTARES: Sheila Hull, 49 Southampton Rd., Far Cotton, Northampton (UK £1.25, overseas please enquire).

STERB: John Hind, 14 Bingham Rd.,Radcliffe on Trent, Notts.

STARBASE 13: Brian Longstaff, 13 Woodfarm Drive, Sheffield, South Yorks.

EMPATHY: Catherine Owens, 30 Overdon Way,Halifax, Yorks. (£1)

WSF/STM: For Kirk fans in particular. Dee Smith, 25 Wolser Rd.,Caversham,Reading, Berks. (£1).

For info write STAR TREK WELCOMITTEE; Jerri Franz Foreign D.C. Rt.1, Box 79A,White Post, VA. 22663, USA, or STW's UK rep - Janet Quarton as above. Note; when writing for info to any of the above, please enclose an SAE or an International Reply Coupon as appropriate.

LIBRARY

ACC 2650 LIBRARY INDEX

Suffixes; G - General e.g. hardware + software items or general.
H - Hardware e,g. circuit diagrams.
M - Manufacturers data sheet
S - Software
T - Tapes ; punched paper or cassette.

G001 2650 Manual section. Full description 2650 registers, pin signals, timing- including some examples.

G002 'Introducing the 2650'.Small book covering fundamentals of 2650 and other chips.

G003 '2650 Introductory brochure and short form catalogue'

G004 'Designing with microcomputers'.Describes software from basics and 2650 instruction set. Describes computer terms and includes an 'Intelligent typewriter' listing.

G005 Electronic Design article; 'Using the 2650'. Describes 2650 and family devices.

G006 Electronic Design article; '2650 Intelligent Typewriter'. Circuit & Listing.

G007 Various articles; Signetics news sheets Mar '76 & Jan '77, sections on PC1001 and Pipbug, Promethus & ABC1500 card. Mullard Bulletin Nov '75; section on 2650.

G008 Technical notes 51 + 60. Description of Promethus card. (Promethus = 2650 assembler).

G009 Application memo AS50. 'Serial IO' using SENSE & FLAG. Hardware and softw re description.

G010 IBM book 'Data Communication Primer',describes in some detail communication links.

H001 Application memo SP50. PC1001 development PCB.

H002 Application memo SP51. 2650 demo system, for use with PC1001 or ABC1500.

H003 Application memo SP55. ABC1500 development system.

H004 MP51 2650 initialisation.

H005 MP52 low cost clock generators.

H006 MP53 address and data bus interface techniques.

H007 Data sheet '2651 programmable communications interface'. USART chip.

H008 Data sheet '2655 programmable peripheral interface. 3 x 8 bit parallel IO ports or 2 ports + timer or 2 ports + 8 bit serial.

H009 Data sheet 8T31. 8 bit bidirectional IO port.

H010 Data sheet 2652 USRT - multi protocol USRT.

S001 Section 2650 manual. Full instruction description with addressing modes and word formats.

S002 Section 2650 manual. Assembler language.

S003 SS50 Pipbug description & listing.

S004 SS51 description Pipbug paper tape object code format.

S005 - 9 Application memo's AS51-55.

M001 Quarndon - system 2650/8080

M002 Epsilon - system

M003 Central Data Company.

Any enquiries should be accompanied by a SAE either foolscap size for index only or A4 size for library material.

R A Munt 51 Beechwood Drive,Feniscowles,Blackburn, Lancs BB2 5AT tel (0254) 22341

ACC GENERAL LIBRARY

Some new items;

- 7768 CPU Design Manual	20p P&P
- 'A Guide To Baudot Machines' Articles from Byte April,May & June 1977 describing 5 level code teletypewriters.	20p P&P
- MUBUS data file	20p P&P
- 'A Guide To SC/MP Programming' donated by Kemitron.	30p P&P

NOTE: TELEPHONE NUMBER IS NOW : (0332) 513769

Frank Cato, 3 Rykneld Way,Derby DE3 7AT

ACC 8080/Z80 LIBRARY

The library has grown substantially with the addition of around 50 items from Intel's INSITE library. Thanks go to Bob Cottis who made the arrangements with Intel. There are many other software items including a range of monitors from 256 to 1.5k bytes in size, a Lunar Lander game, LIFE, a couple of BASIC's, routines to drive a TV display and lots more.

If you're just thinking of building an 8080 or Z80 micro the Hardware section can provide you with CPU, memory and interface circuit examples to help your design along. A 9" x 4" SAE will get you the latest library listing.
The trip to Zilog seems to have died the death, however Z80 enthusiasts should try not to miss the seminar to be organised by Lynx in November, see ETI for details.

Finally, I would be grateful if anyone who gets Interface Age, Kilobaud, or DDJ would let me know of articles that might be of use to the Library.

Neil Harrison 15 Hill Rd.,Watchfield, Swindon, Wilts SN6 8LA

LETTERS

CONVERSION ETC.

In response to the article by Steve Bailey in the April issue of ACCN, I give details below of a method of converting binary numbers to BCD format (or vica versa) called Couleur's technique (Refs 1 & 2).

FOR FRACTIONAL BINARY VALUES (With a leading decimal point)(Binary word length = n bits)

(1) Shift the binary word one bit right (LSB first) into the final result field.
(2) Look at the groups of four bits which will make up the BCD digits of the result;
If a group of four bits has a value greater than 0100, subtract 0011 ignoring any carry (or borrow) from the four bit field; go to (2)
Else; go to (2)

Repeat for all groups of four bits in the result field whose value is non-zero.

(3) If the total number of shifts is less than n, go to (1).

(4) Finish.

Example; convert .1001 binary (0.5625 decimal) to BCD.

	Final result field (initially zero)	
.1001		
100	1000	First shift right
	0011	Adjustment
100	0101	Result of adjustment
10	0010 1000	Second shift right
	0000 0011	Adjustment
10	0010 0101	Result of adjustment
1	0001 0010 1000	Third shift right
	0000 0000 0011	Adjustment
1	0001 0010 0101	Result of adjustment
	1000 1001 0010 1000	Fourth shift right
	0011 0011 0000 0011	Adjustment
	.0101 0110 0010 0101	Final result
	. 5 6 2 5	

Conversion from fractional BCD to binary may be done by exactly the reverse sequence; shifting the BCD bits left, and adding three to the groups of four bits if their value is greater than 4.(This may seem at first sight to be incorrect; a group of bits of decimal value 3 say could, in the conversion from binary to BCD have resulted from an initial value of 3 or 6, however a little thought will show that the value 6 (or 7,5,4) could never occur after a shift right in the binary-BCD conversion). The conversion from BCD to binary does have one slight problem in that the conversion must stop at some point in time (if a result to the conversion is ever to be obtained), and this can be done in two ways; by testing for all zeroes (in the field which initially contained the BCD value to be converted) before the shift left - obviously this is the end of the conversion because there is nothing left to convert -, or by counting up the number of bits shifted left into the result field (including leading zeroes which are, after all, significant) until the required word size is reached.
The best way of finishing the conversion is probably a combination of these or at least the second, because some decimal values do not have exact binary equivalents - 0.1 decimal for example, and would just generate an infinitely recurring binary output and the first method of termination would never work (try it and see).

FOR INTEGER BINARY VALUES of length n bits

The method is the same as for fractional binary values except that the binary value is shifted left (MSB first) into the result field. The result MSB is at the left of the result field. The adjustment is to add 0011 if the group of four digits has a value greater that 0100, ignoring carry out of the four bits.

For integer BCD to integer binary conversion, the above process is reversed, this time with no problem about infinite binary results.

The method described above is quite reasonable to program for purely fractional or integer binary values, but things get a bit complicated when it comes to writing a program to convert binary numbers of the mantissa-exponent form; the actual number to be converted may have the decimal point in effect located somewhere in the middle of the mantissa word, so the two parts - one after, the other before the decimal point - must be converted by the two different methods and the two results combined.

On the subject of FPA, it is usually desirable to extend the precision of calculations, but a limit is imposed by the size and number of words of data the FPA routines are handling.
One useful way of getting an extra bit into the mantissa word makes use of the fact that the mantissa is usually normalised so that its MSB is '1' and then there is a sign bit in front of that, so the format is, for say a 24 bit sign+mantissa; one sign bit plus 23 mantissa bits. However, since the MSB of the mantissa is always '1', it may be missed out completely, and the FPA routines altered to strip off and store the sign bit, and insert '1' instead of it. In this way, for a 24 bit word, you get one sign bit and, in effect, the full 24 bit precision for calculations.

ON OTHER SUBJECTS - mainly 'Whither' ACCN V5 Iss 2

It seems to me that the development of 'Flow-Ware' would not present too many problems, and I share M Kaleel's enthusiasm for the ACC to involve itself in research along the lines described. What I do not quite understand is how (or why) the overall FlowWare computer should be called a 'Language-less Entity'. Presumably 'Language-less' must be taken with a large dose of salt, unless he has found a way of communicating without language! Probably what was meant to be conveyed was that the Interactive language should be totally free format, with the computer dotting the i's and crossing the t's for the programmer. I would like to propose that this concept be called FREEFORALL, for FREE FORmat Adaptive Language. If fully implemented, this would allow the programmer to present his problem in any way he could think of, but a more practical method of doing something approaching a workable system would be to allow the programmer to define his own language, instruction set etc. to suit his own tastes. The definitions could be handled by a version of Instruction Set Processor (ISP) language (a well established system for defining operations right from the hardware of the computer up to the highest levels.)
This would solve problems in that a programmer could set up a system as complex or as simple as he liked, and could mould it exactly to his programming requirements, whether it be array programming or playing Star-Trek or LIFE. However the disadvantages would also be rather weighty because there would be as many languages as there were programmers, not a pleasant thought at all.

Refs;
1) IRE TRANSACTIONS ON ELECTRONIC COMPUTERS Dec '58 pages 313 - 316
2) RCA COSMOS DATA BOOK (1974 copy) pages 184 - 187 details of hardware binary to BCD converter.

I G Barnard

COMPUTER-LESS PROGRAMMING

I would like to say something that may be of interest to readers and members of ACC who cannot afford a computer or a micro processor kit.

A simple way to learn FORTRAN is to buy for about £50 a CASIO FX201P. This is a 127 step programmable calculator based on the FORTRAN method of programming with SUBROUTINES, LOOPS, MEMORY REGISTORS etc.

To show my point, below is a program for your

Random Number Generator (Vol 4 Iss 2).

ST* are statement numbers.
IF 2=1:2:3:4: means if memory 2 is less than memory 1 GOTO statement 2. If equal GOTO 3, if more than GOTO statement 4. This can be used with constants e.g.; IF 2=K12:2:3:4: so any conditional branching can be done.
Branching is just a GOTO 1 (statement number).
All constants are preceeded by a 'K' , and 'EE' follows exponent.
':' is statement end.
'ENT 1' means enter number (from keyboard) into memory location 1.

```
ENT1: ST*0: 1=1xK23:
2=K1EE9: 3=K1EE8: 4=K0: 5=K0:
ST*1: IF 1=2:3:3:
ST*3: 1=1-2: 4=4+K10: GOTO 1:
ST*2: IF 1=3:4:5:5:
ST*5: 1=1-3: 5=5+K1: GOTO 2:
ST*4: 1=1-4-5: ANS 1: GOTO 0:
```

If 12345678 is entered, first answer is 83950594, then the answer button is pressed for next random number; no more entries just answers. This as you notice is similar to a computer program.

T Turnbull

LARGE SCALE SOFTWARE

If anyone is interested in software for larger machines (not micros) I have a copy of a Pascal compiler which I could readily hand-code into assembler or FORTRAN or a similar lower level language, thus starting bootstrapping process to provide full Pascal implementations. I also have a fairly large collection of basic software written in Pascal. It might be possible to implement a subset of Pascal on a large 6800. If you own one and are interested get in touch and we'll see what can be done.
Ian Gooding 57A Erpingham Rd.,London SW15
(01) 788 1778

NASCOM I

NASCOM (suggested by some competitors to be an acronym for 'NASty COMputer') represents a significant advance at the low price end of the hobby computer kit market.

To be introduced by LYNX Electronics at their seminar at the Wembley Conference Centre on 26th November, the advance publicity indicates that it will be of great interest to many ACC members.

Based on the Z80 CPU chip, the NASCOM CPU is an 11" x 8" card carrying, as well as the MPU and a 1 or 2MHz clock, 2 k bytes of RAM,a 1K byte ROM monitor, TTY output (RS232 or 20mA) and cassette interface (unfortunately not CUTS but 5KHz tone on-off modulated). An additional 1K EPROM and 16 lines of IO may be added on the board, while the system may be expanded to 64k bytes of memory plus 256 input and 256 output ports.

So far this sounds like another D2 kit, or KIM - but NASCOM also includes a 46 key 'QWERTY' keyboard (instead of the usual 20 pad hexadecimal version) and a video display drive which gives a UHF output to display 16 lines of 48 characters on a normal domestic TV. The operating system takes advantage of the keyboard and TV VDU display, providing the following commands;
Execute, Set Breakpoint, Single Step, Tabulate on Screen, Examine/Modify Memory, Examine/Modify IO, Dump memory to serial IO, Load Memory from serial IO.
And all this for about £197 !
Details from Lynx Electronics (London) Ltd.,
92 Broad St.,Chesham,Bucks. Tel (02405) 75154

***won't be able to attend the seminar myself, so I would appreciate reports and observations on this kit from anyone who goes and/or buys a kit ed ***

SYSTEM 45 IS FOR THE SUPER HACKER

For anyone with a few bob to spare, Hewlett Packard's new desktop computer (System 45) might prove to be an acceptable Christmas present.
The basic package has a built in keyboard (full ASCII plus numeric pad plus special function keys), 24 line x 80 character VDU, with optional graphic mode, 16 k bytes of read/write memory (system software is stored on plug-in ROM's - up to 8 special purpose ROM's may be fitted),and one 217K byte miniature tape transport (a second may be fitted) costs only £8,384, but the keen amateur will probably wish to add a 80 column line printer (fits neatly under the VDU screen), a couple of floppy discs and a four colour plotter.

STARDATES

SURAK

As the title of this month's column implies, I am giving a rundown of the various 'Computer Dating' algorithms. No, nothing to do with marriage beareux, but the various interface algorithms e.g. Julian to Gregorian or Gregorian to Day of Week.
In all the following I have used the standard abbreviations; W = Day of Week (Sun = 0,Mon = 1)
D = Day of Month. J = Day of Year (i.e. Julian Day)
M = Month, Y = Year (Modulo 100), C= [Year/100]
[] brackets denote truncation towards zero. L = 1 if leap year, 0 otherwise.
Starting , then, with the Day of Week algorithms, we have;

(1) The Modified Heath-Baker Algorithm. This was first devised by conjuror Royal V Heath, but as originally published (Simon, Mathematical Magic; Allen & Unwin) it was riddled with bugs, plus a few ambiguities. I therefore devised an amended version which was published in ACCN V3 I5; however I did not catch all the bugs, as was pointed out in I6. I therefore present the final (?) version (hence the 'Modified'), written in FORTRAN for compactness, though it is very good for calculator users. (It should be pointed out, of course, that all vars are INTEGER).

```
      DATA A1/3,2,1,0,6,5,4/,A2/5,3,1,6/,B/1,4,4,0,
     +2,5,0,3,6,1,4,6/
      IF(100*C+Y.GT.0)GOTO 1
      Y=100*C+Y+701; C=Y/100; Y=Y-100*G
1     IF(C.GT.17.OR.(C.EQ.17.AND.(Y.GT.52.OR.(Y.EQ.52.
     +AND.(M.GT.(.OR.(M.EQ.9.AND.D.GE.14))))))GOTO 2
      W=A1(MOD(C,7)+1); GOTO 3
2     W=A2(MOD(C,4)+1)
3     IF(M.NE.1.AND.M.NE.2)L=0
      W=MOD(W+Y+Y/4+B(M)-L+D,7)
```

(2) Zeller's Congruence. This purely numeric (no table look-up) algorithm is unusual in that it uses the Roman year, which begins in March. (Which incidentally,is how September, October, November and December - meaning 7th, 8th, 9th and 10th - get their names.) Because of this, to use Zeller's Congruence you first have to shift the date 2 months backwards, e.g. 14/10/77 becomes 14/8/77; 14/2/77 becomes 14/12/76 etc. It is then simple;

$W=([2.6*M-.2]+D+[Y/4]+[C/4]-2*C)$, mod 7.

(3) Gregorian to Julian; if M=1, J=D. If M=2,J=D+31. If M.GT.2,J=[30.6*M-32.4]+D+L.

(4) Julian to Gregorian; If J.LT.32, M=1 and D=J, else if J.LT.60+L, M=2 and D=J-31, else M=[(J-L+32.21)/30.6], D=J-L-[30.6*M-32.4]

(5) Schecter's Algorithm; the number of days between 31/12/1899 and any date with C=19 is given by
S=[Y*365.25-.25] +J
W is then simply S mod 7, and the number of days between two dates is of course $S_2 - S_1$.

Schecter to Julian conversion is achieved using;
Y=[S/365.25] , J=S-[Y*365.25 - .25]

Incidentally, anyone who will be in London before February, and who wants to see an analogue computer (albeit of a specialised type) at work, could do worse than to go and see 'Laserium' at the London Planetarium. Personal booking only, not less than 3 days ahead, no 'on the night' tickets.

MOTOROLA

Product Preview

During the last couple of months, Motorola have been announcing new devices in the 6800 range.

The 6800 MPU itself is now available in 1.5MHz (MC68A00) and 2MHz (MC68B00) clock versions which are hardware and software compatible with the 1MHz MC6800.

A clock oscillator/driver (MC6875) is now available in sample quantities. For about £4 in small quantities, this also provides the necessary logic to perform a system reset, to operate DMA and dynamic memory refresh, and to interface with slow memory.

A new PIA (Peripheral Interface Adaptor), the 6821, is now available which overcomes several disadvantages of the older 6820; the Enable input is fully TTL compatible, operation is fully static, peripheral data, control and Int Req lines can drive two TTL loads, and the control lines needn't be held when the Reset input is active.

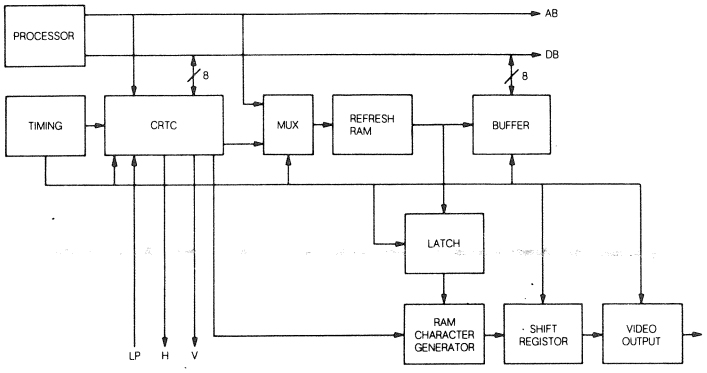
A new MPU, the 6802, is software compatible with the 6800 but includes an on-chip clock (4MHz XTAL for 1MHz CPU clock), and 128 bytes of on-chip RAM, 32 bytes of which can be powered from a standby supply. To squeeze this into a 40 pin package, the DMA facility has been dropped, otherwise the hardware interfaces are as the 6800. This chip has been designed for high volume users so will cost less than £20 even for one-off.

A companion chip (MC6846) combines 2k bytes of ROM, 10 IO lines (like half of a PIA), and a programmable 16 bit timer-counter. With the 6802, a 2 chip computer can be made. Motorola say that it is possible that they will market the 6846 pre-programmed with MIKBUG and an audio cassette interface program - to give a simple 2 chip amateur computer.

For a more powerful system, Motorola will be introducing the 6840; containing three programmable 16 bit timer counters.(Very useful for simulating a UART or cassette interface). Also planned are 6845 CRT controller, 6844 DMA controller, 6843 Floppy Disc controller, 6828 Priority Interrupt Controller, not to mention the fabled MC6801 next generation MPU, whatever that will turn out to be.

CRT controller (CRTC MC6845)

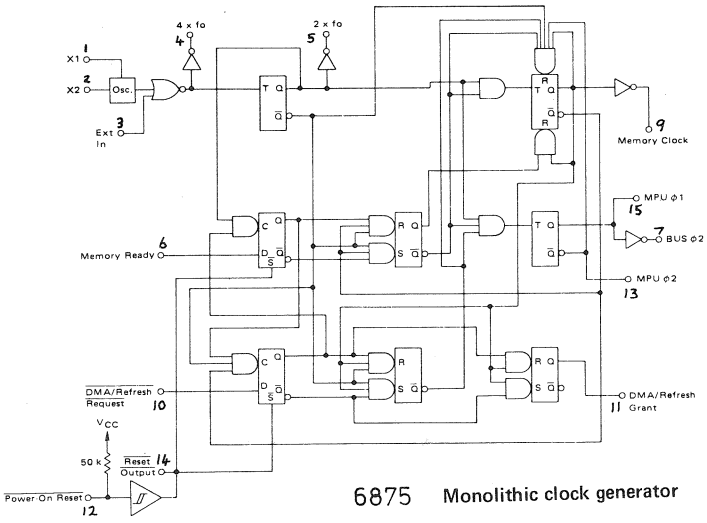
The MC6845 CRT controller performs the complex MPU to CRT interface function. Applications include TTY- and lineprinter-format displays, intelligent terminals, work processing. and information display devices.



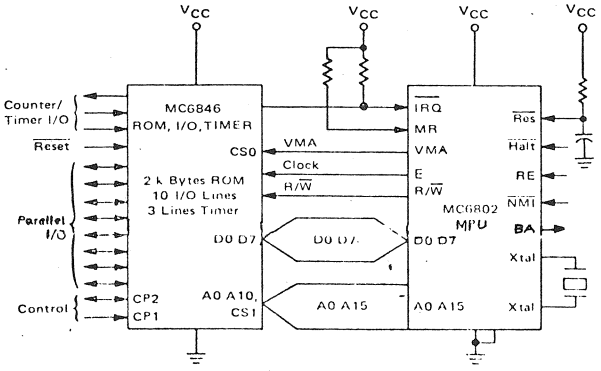
The controller has:

- 14-bit refresh address;
- hardware scrolling and paging;
- three interface modes;
- alphanumeric or full-graphic capability;
- programmable dots per character;
- programmable rasters per character;
- programmable characters per line;
- programmable lines per screen;
- programmable horizontal synchronization width and position;
- programmable vertical sychronization position;
- programmable cursor register;
- on-chip cursor register;
- on-chip light pen register;
- needs no line buffer;
- full TTL compatibility.

Moreover, it requires only a 5V supply and is compatible with MC6800 MPU.



6875 Monolithic clock generator



Random access memory RAM (static)

Static RAM	Access time
128 x 8-bit MCM6810	450ns (1.0MHz operation)
MCM68A10	360ns (1.5MHz operation)
MCM68B10	250ns (2.0MHz operation)
1024 x 4-bit MCM6614-6	200ns
MCM6614-7	300ns
MCM6614-8	450ns

Programmable read-only memory (EPROM, alterable)

1024 x 8-bit MCM68708
2048 x 8-bit MCM68716 (to be announced)

Electrically programmable, but may be erased with ultraviolet light and reprogrammed electrically by the user.

Motorola have also introduced the MMS68103, which is a reduced cost memory card which is EXORciser/PDS/Micromodule compatible (not as flexible as the present EXORciser card, and not suitable for speeds above 1MHz, but cheaper; 8k x 8 approx £317). They also plan to produce a cheap memory card for the D2 kit ; 16k x 8 retailing for just over £300, introduction due in October.

BUSSES *Part 4* BIG BUFFERS

Following the general description of Tri-State busses in the last newsletter, we now start to examine some of the more popular integrated circuits designed to act as Tri-State bus drivers and receivers.

They fall into three rough categories;

- MOS devices having limited drive capability, MPU bus outputs fall into this category, as do the bus lines of specialised interface chips e.g. 6850 ACIA, 2513 character generator.
- Bipolar devices having medium drive capability similar to normal TTL e.g. about 16mA sink at logic '0' level.
- High drive capability bipolar devices designed to drive large bus structures and high capacitance loads.

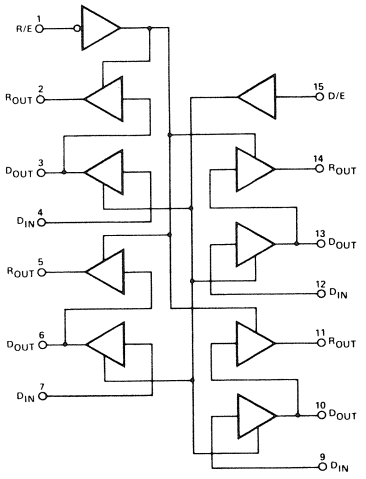
This issue we will concentrate upon the high drive devices.

The 8T26 (MC6880) consists of four separate receiver transmitter combinations, for use with a bi-directional bus system. Driver and receiver output current sink capabilities are 40 & 16mA respectively. Max input current is 200uA. '1' on D/E input enables the drivers, while '0' on R/E enables the receivers. Propogation delay is typically 15nS. 8T28 (MC6889) is similar but the receivers and drivers are non inverting.

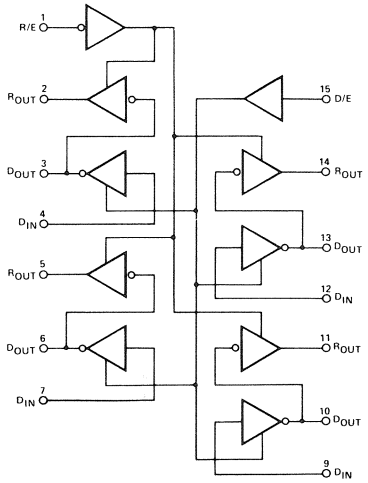
The 8T95 - 98 series (equivalent to DM 8095 - 98 and MC6885 - 6888) each have six devices per pack, arranged in different configurations (inverting or non inverting, common enable or grouped as 2 + 4) as shown below. Each output, when enabled, can sink 48mA or source (logic '1' output) 5.2mA. Input current is less than 400uA (enabled), 40uA (disabled). Propogation delay is typically 8nS.

QUAD BUS DRIVER/RECEIVER

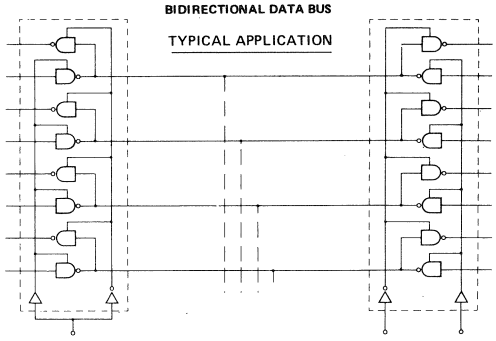
MC6889/MC8T28
8T28 – NON-INVERTING OUTPUT



MC6880/MC8T26,
8T26 – INVERTING OUTPUT

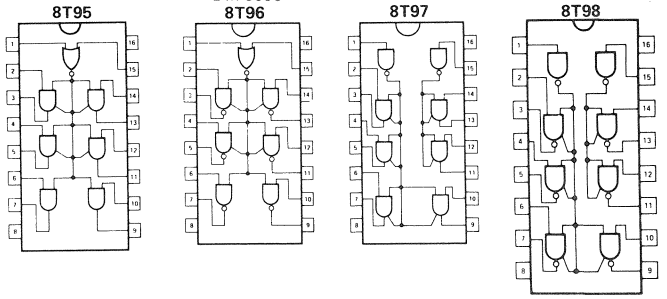


BIDIRECTIONAL DATA BUS
TYPICAL APPLICATION



Hex buffer/inverters

MC6885/MC8T96 through MC6888/MC8T98
DM 8095 DM 8096 DM 8097 DM 8098
8T95 8T96 8T97 8T98



7768 MICROCOMPUTER

First, a note about 7768's provenance; it has been designed for Bear Micro Systems, who have agreed to let details be published in the ACC Newsletter, and who also offer a discount to ACC members for parts and literature. (See page 2 of this issue).

It seems to have met a need, and at this time about 60 people have built or are in the process of building the CPU board. Having got the CPU board working, it is not long before the user wants to expand his system by adding;

- more memory
- some form of terminal (VDU or hard copy)
- some form of permanent storage (such as a cassette)
- system software; at least a monitor and preferably also a high level language such as BASIC.

To meet these needs, a 4k RAM board is now in the prototype stage and will be featured in the next issue of the ACCN. Also being designed is a 'Soft Monitor' board. This will carry;

- one or two serial asynchronous IO ports with provision for RS232 or 20mA or TTL level interfaces, also a crystal controlled baud rate generator (75 to 9600 baud).
- 1k bytes RAM which can be write protected.
- simple bootstrap loader, which can be used to load a system monitor into the write-protected RAM via one of the serial ports.

If Murphy turns a blind eye, we hope to describe this card also in the next newsletter.

Finally, a TV VDU card is planned (memory mapped store) which will also provide for a parallel input from an ASCII keyboard such as that described in Vol 4 of the ACCN.

CHESS NUTS

Douglas Panrod @ 1445 La Cima Rd.,Santa Barbara, California 93101 USA has started the Computer Chess Newsletter.

SC/MP GROUP DOES HAVE A HOME

In his article (J-2 Microsystem - V5 I3) J.Knight offered to start a SC/MP or 'small mpu' group, but your editor forgot to print his address ! It is; J.R.Knight Second Floor Flat, 6 Alfred St.,Bath, Avon BA1 2QU

ART PAGE

The Computer Arts Society publishes a bulletin called PAGE 8 times a year. Membership of the society is £2 a year, details from John Lansdown, 50/51 Russell Square, London WC1B 4JX

TUNE IT

I suppose the last issue of ACCN left a few people envious, all these new micros around and here am I with an ageing 6800 etc. The problem of modern electronic industries is the current development speed, and no matter where you get into the intrepid micro (or other) market, by the time you get settled you are getting the impression you just got off a high speed train with no chance of catching it (usually a financial constraint).

We have to remember that unlike cars micros don't rust, don't wear out (only the Return key does that) and generally will last for ever (seen any 8008's in Oxfam shops yet ?). The way out of our envy is to look at our system and say, hand on heart, am I really getting the full capacity use out of my system ? The answer will, in 99% of cases be ' well . . . not really'. So lets squeeze some more out of our system, but how ?

First lets see what the program is doing and where the activity is. I have come up with a cheap, by most standards, method of doing this. Assuming a 16 bit address bus you need 2 8 bit DAC's (Digital to Analogue Converters) and before anyone says 'too expensive' a simple 8 bit DAC only needs a few resistors. The idea is put one DAC on the first 8 address bits and the other DAC on the remaining 8 (less in some machines). The DAC o/p's are taken to an oscilloscope X and Y inputs and once set up the max address in the system will be top right of the screen and bottom left will be '0000'. You can see why cheap DACs or home brewed items can be used, since precision or stability are not really required as we are getting only an indication of current memory addresses being used by the system. With a program executing you will see where the main activity (or loops !!) are.

Having established the main activity you can decide your tune up method. If the hold up is IO say VDU, TTY or in some cases slower memories (causing wait states) then you can try to speed things up using buffering techniques and also interrupt processing can be very advantageous if a lot of slow IO is in use. If the slower memory is holding things up then alter your memory map and put less used program elements in the slower memory. I'm sure you are getting the drift by now so having done this what next ?.

Suppose you are doing a lot of number crunching, are your reiterative routines as efficient as possible ?. You may find a hardware number cruncher would pay off (anyone tried the new National chip ?) or a scientific calculator as an IO device, saving time and also some precious memory. Notably SWTPC's 8K BASIC version 2 is twice as fast as version 1 (I don't know how they did it).

Finally, will the hardware work any faster ? Most systems are optimised at say 1MHz clock but you can speed this up to say 1.3MHz before any noticeable problems occur. That is another 30% increase, add this to other increases and you don't need your Z80 yet ! One point to be careful of, if you change your system crystal check that there isn't anything alse dependant upon it, such as baud rate generators, as you may find wierd things happening.

I hope you will benefit from these thoughts and ideas and if you discover any good tips then put pen to paper and tell ACCN.

Postscript; I had many letters and phone calls concerning RT68. Since then I have obtained a listing of Smoke Signal's SMARTBUG, and the feature of this is a Software Trail facility giving a dump of registers as each instruction executes. Very clever.

Dave Goadby

PROM PROGRAMMING SERVICES

BYWOOD ELECTRONICS, 68 Ebberns Rd.,Hemel Hempstead, HP3 9RD tel 0442 62757

1p per byte plus £6 per master plus £1 per copy (including the first) plus the cost of the PROM from Hex listings. Thus for a full MM5204;

512 bytes	£5.12
Master	6.00
Copy 1	1.00
MM5204	10.95
	£23.07
VAT	1.85
	£24.92

MARSHALLS 42 Cricklewood Broadway,London NW2 3ET

Copying from Master PROM onto a blank PROM	£2.50
Copy from customers tape	£5.00
Making a tape from customers program	£12.50
Erasing	£2.00

5204 and 1702 Card sets available.

COMPUTABITS 41 Vincent St.,Yeovil,Somerset tel (0935) 26522

Paper tape generation from Hex code per 1k bits (e.g. 256 x 4 PROM)	£5.00
PROM programming from paper tape	£3.50

Facilities for 1702A,4702A or 3601

HOW TO GET FROM RESTART 0000 TO SOMEWHERE USEFUL

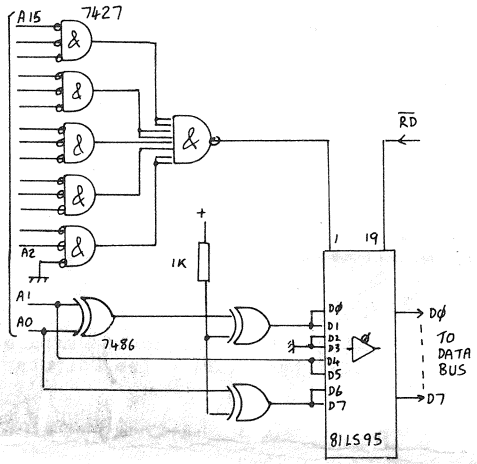
Although designed for the 'elite' Z80 users, the basic idea should prove of interest to other processor users.

Octal tri-state buffer (1) will only produce an output for addresses 0000 to 0004 (Hex). The two address lines A0 & A1 are decoded to produce the desired 8 bit output on the data bus. Here they are decoded according to the table;

A1	A0	Data	
0	0	C3	This in Z80 language performs an unconditional jump to F000 upon the 'reset to 0000' instruction, or connecting the reset pin to 0V.
0	1	00	
1	0	F0	
1	1	don't care	

Note the use of the spare 7486 gates as inverters. There is no reason why similar decoding could not be done for other instructions for other MPU's.

R Sinden



AMATEUR COMPUTER CLUB NEWSLETTER
Vol 5 Iss 4 October 1977
Editor; Mike Lord
7 Dordells, Basildon, Essex
tel ; (0268) 411125