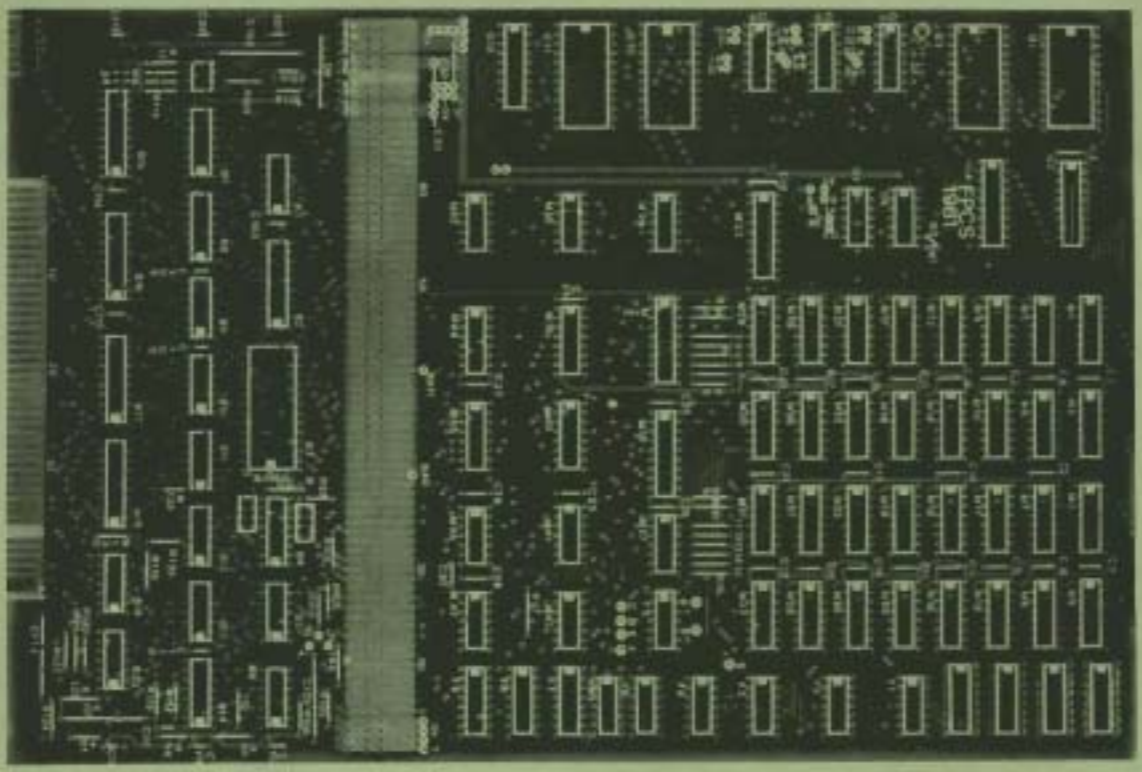
MICROPOWER VOLUME 2, NUMBER 4

A MAGAZINE FOR NASCOM USERS

nascom

SEPTEMBER 1982 £1.00



64 KILOBYTE RAM
and
BUFFER CARD
with PROGRAMMABLE GRAPHICS

This 64K RAM card is suitable for the Nascom 1 or 2. The double sided glass-fibre P.C.B., 302 mm (12 ins.) by 203 mm (8 ins.), holds up to 4 blocks of 16 Kb dynamic RAM (4116). When all four blocks are fitted the whole of the Z80 address field is occupied by RAM. The on board mapper allows parts of this address field to be selectively inhibited in either read or write mode, or both. The mapper divides the address field into 4K blocks, and any two selected blocks can be further subdivided into 2 x 2K blocks.

The graphics section is entirely separate from the dynamic RAM, but it can be mapped in at any chosen 2K boundary. It can use an EPROM (2716) to give a pre-programmed character set, or static RAM (2 x 4118, or 6116) to provide user-programmable characters.

For the Nascom 2 the memory and graphics section can be separated from the "buffer" section; the resulting 8 x 8 card can be plugged into a standard Nasbus (80-bus) edge connector For the Nascom 1 the bottom 8 x 4 ins. section of the card provides full buffering between the Nascom 1 43-way connector and Nasbus. In addition the following extra facilities are also provided:-

1. Power-on jump; this allows the processor to execute a program at any preset 4K boundary on power-on or reset.
2. Synchronised Reset; the reset pulse is synchronised with the processor M1 cycles, to prevent corruption of data in dynamic RAM
3. Wait state generator; one wait state can be added to memory or input/output access
4. ROM socket; a 28 pin or 24 pin socket can be placed at position B3, and via a series of links this can accommodate a 2716, 2732, 2764 or the standard Nascom Basic ROM
5. Input/output; a partial decode is provided which allows for 64 input/output addresses.

The 64K RAM card is available now, price £39.50, from

MICRO POWER Ltd.,
8/8A, Regent Street, Leeds LS7 4PE
Tel. (0532) 683186

Please add 55p p/p and V.A.T. at 15%.

Contents

Editorial	Page 2
Labels for Xtal BASIC	Page 3
Review of 32K CMOS Battery-backed RAM board	Page 8
A 64K EPROM Board for £4?	Page 11
Making a Nascom 1 into a Nascom 2 - so to speak	Page 14
CMOS Battery-backed RAM Update	Page 20
Letters	Page 24
2 Megabyte Nascom	Page 25
4Ks Via 2 Megs.	Page 27
Errata	Page 31
Hole in the Wall	Page 34

Editor - Ian J Clemmett

Published by - Micro Power Ltd., 8/8a Regent Street, Chapel Allerton, Leeds LS7 4PE

Printed by - Dataform Press, Leeds LS12 4JF

Editorial

I wonder how many people actually read these words of wisdom that are commonly referred to as Editorials. I must admit that I find them pretty heavy going in the glossy magazines. Well, the point of those last two lines of dribble is that if you do not read the rest of this Editorial you will not know what has happened to the front cover and what will be happening from the next issue onwards.

I proudly announce that from the next issue onwards this magazine will be known as the Nascom Newsletter and will be completely restructured to include a section from Nascom themselves (or rather, the Nascom bit of Lucas Logic). I am hoping to get news 'from the front' from Nascom so that I should be able to give you the latest developments, products from Lucas and possibly a section of hardware and software hints from them.

I also want to know what you want to see added to the magazine, removed, changed, etc. This is supposed to be your magazine but I can not do much unless I get some feedback from you. The more articles and letters that come in, the better the balance in the make-up and the higher the standard of the magazine in general. My current thoughts on the make-up of the magazine are one third software, one third hardware and one third letters, adverts, etc. This is the type of balance that I am aiming for at present but I will willingly change it on demand.

Other suggestions for the magazine are a club page and a witty corner (Corn Corner?). Also, it may be possible to set up a 'wanted' section where readers can get together to buy items in bulk and possibly get some discount from the suppliers. Any suppliers willing to make any offers?

The magazine has not got enough funds to supply you all with pens, envelopes and stamps to write to me but I am including a sheet for you to fill in, and memory plague to anyone who does not reply. This should give me some indication of what you want to see in the magazine which makes the editing easier.

In the next issue of the magazine, the first of the Nascom Newsletters, we are hoping to run a competition. The details have not been finalised yet but it will be co-sponsored by Micro Power (our owning company) and Microcode (Control) Ltd. with all sorts of prizes and goodies to be won. This should be enough incentive to ensure that you will all buy the next magazine.

Well, that's it for another editorial, the last in the Micropower series, next stop Nascom Newsletter. Many thanks to the readers who have been contributing to Micropower, keep the articles coming in, the more the merrier.

IJC

Labels for Xtal BASIC

by Stephen Hope

This article describes how labels can be added to Xtal BASIC. These labels can be used instead of line numbers in such commands as GOSUB and GOTO. This helps in producing easy to read programs, and allows subroutines to be referenced by meaningful names. Subroutines and other sections of a program can be moved around without the chore of searching the complete program to find any subroutine calls, or other references. It is surprising how such a relatively simple extension to BASIC improves the readability and ease of modification of the code.

The basic (no pun intended) idea is to first change the BASIC interpreter so that it will ignore a label on a line, and then alter the routines for GOSUB, GOTO etc. so that they will work with line numbers or labels. The first thing to do is to find a suitable format for the labels. I finally settled on a string of alphanumeric characters which starts and ends with a single quote mark. The quote marks make it relatively simple to distinguish between a label and a line number or anything else, in, for example, an IF..THEN statement.This will minimise the increase in execution time for the interpreter.

Next, a suitable point to patch the program. There is a section of code in Xtal which is used every time a statement finishes. This code checks for a keyword, a variable or a line terminator and executes the appropriate routine. Since all keywords are greater than 80H, any character less than this at the beginning of a line is assumed to be part of a variable name. Any illegal names for variables are detected later in the LET routine. Here I diverted the code to check for a single quote. If one is found, then my routine skips over any characters between this and the next. Note that this particular patch allows a label anywhere on a line at the end of a statement. Although labels will only be used if they occur at the beginning of a line, this allows comments to be embedded in a line, as long as they are enclosed in single quotes. A missing second quote gives a syntax error. Otherwise, the routine simply treats anything after the end of the label as the start of a new statement.

Finally, altering the GOTO and GOSUB statements. Here I again patched the original code in the interpreter. GOTO, GOSUB and RUN all use the same code to jump to a line number. The subroutine call to get the line number is changed so that it checks the first character after the keyword. If it is a quote, then it is assumed to be the start of a label. If not, something which is dealt with by the original routine. (This allows labels to be used with more complex types of statement, such as an IF statement, where almost anything is legal after the THEN

statement.) Any syntax errors are dealt with by the original code. I have also patched most of the other BASIC routines which use line numbers to make things nice and simple.

The characters of the label are then counted and a pointer to the first character kept for a search. The routine then searches the entire program, starting at the beginning, for an identical label. Both labels must be the same length for the search to be successful, and any embedded keywords in the label, such as ON, PRINT etc. will not affect the routine. Blanks within the label are significant. There is no limit to the length of a label, but Xtal will restrict the maximum number of characters in a line to 90+. If the label cannot be found, then this is taken to be an error, similar to a GOTO to a line which does not exist.

When a label is found, the relevant line number is returned, and the line is searched for in the normal way. This method has the great advantage of simplicity, although the program text is searched twice. I did this to make the patches to Xtal as simple as possible, and to allow other commands which use line numbers to use the code. The reduction in speed for finding a label instead of a line number in a large program is appox. 60%. In a small program, or where the line is near the beginning of a program, less. This does not seem to be a significant amount, since the label can always be replaced by the equivalent line number in speed-critical sections of code, (with some loss of clarity). The routine for skipping over labels slows the interpreter by only 1 - 2 %, so that a program with no labels will not seem slower than normal.

The only statement which will not work correctly with this modification is READ. A DATA statement after a label will be ignored by the interpreter, so that the data cannot be accessed. However, using the format 100'LABEL':DATA will fix this.

Now for the payoff. A list of the commands which work with the new labels.

1. GOTO 'LABEL'

2. GOSUB 'LABEL'

3. RUN 'LABEL'

4. ON X GOTO 'LABEL'

5. ON X GOSUB 'LABEL'

6. LIST 'LABEL'

7. EDIT 'LABEL'

8. RESTORE 'LABEL' (see note above)

9. IF..THEN 'LABEL'

10. IF..GOTO 'LABEL'

The places to put the patches are detailed in the listing of the machine code. I hope that these routines will help you next time that you have to debug a bigger and better version of STAR TREK, or whatever. Anyway, good luck!

Incidently, has anyone noticed a bug in Xtal? It surfaces when you give a statement like: IF A=1 GOTO X=2 which is treated by the interpreter as identical to IF A=1 THEN X=2, not as a syntax error. If anybody has noticed any other peculiarities of Xtal, or even that revolting version of BASIC that was sold by NASCOM, then a letter to the magazine might relieve a lot of other people of some head-scratching.

My final comment. The listing given here is from an assembler called Z2. This assembler is much faster than any other available for NASCOM that I have seen. For instance, the source code for a LISP interpreter which I have just finished runs from 2550 to C2B0 hex. This takes approx. 45 seconds to assemble, producing just under 8K of code. I won't even mention the speed of ZEAP on a similar size file. If there is anyone still out there who still markets this assembler, then perhaps they will write in and give their address for those of us who work on reasonable sized programs.

```
GETNM   EQU     178C ;normal number routine
CODE    EQU     3000 ;run adr.
DUMP    EQU     4000 ;adr. to put assembled code
        ORG     CODE
        LOAD    DUMP
ERROR1  EQU     1341 ;Xtal error routine
IGBLK   EQU     16D8 ;skip blanks S/R
        ;
        ; labels for jumps etc
        ; insert @ 16C2 JP BMPLAB to
        ; ignore labels in text
        ;
SYNERR  EQU     1308 ;syntax err routine
RUNLN   EQU     16B0 ;entry pt. to run BASIC func.
        ;
BMPLAB  CP      "'"-80 ;label marker
        JP      NZ,189A ;LET routine
MISSL   INC     HL
        LD      A,(HL) ;skip chars
        OR      A
        JP      Z,SYNERR ;no terminator
        CP      "'" ;label terminator
        JR      NZ,MISSL
        POP     DE ;tidy up stack
        JP      RUNLN
```

```
        ; New routine for GOTO etc.
        ; change routine to look for line no.
        ; to point here, and fetch no. or
        ; no. given a label.
        ;
        ; PATCH LOCATIONS
        ; put the adrress needed in the 2 bytes
        ; at these adrs.
        ;
        ; RUN,GOTO,GOSUB - 182F
        ; ON X ...       - 1909
        ; RESTORE        - 1734
        ; LIST           - 157F
        ; EDIT           - 2C04
        ;
LABC    DEC     HL ;HL=text ptr.
        CALL    IGBLK ;special entry pt. for ON X
        CP      "'"
        JP      NZ,GETNM ;normal number routine
        PUSH    HL
        LD      E,1 ;counter
NXTCHR  INC     E ;count chars in label
        INC     HL
        LD      A,(HL)
        OR      A
        JP      Z,SYNERR ;end of line in label
        CP      "'" ;check for terminator
        JR      NZ,NXTCHR
        EX      (SP),HL ;save ptr
        PUSH    HL ;ptr to label
        LD      A,E ;save counter (incl. quotes)
        LD      (COUNT),A
        EX      (SP),HL ;get label ptr
        PUSH    BC
        PUSH    HL
        LD      HL,(0C8C) ;scan from beginning of prog.
NXTLIN  LD      B,H ;for next line ptr
        LD      C,L
        LD      A,(HL) ;check for end
        INC     HL
        OR      (HL)
        JR      Z,LABERR
        INC     HL
        LD      E,(HL) ;save line no.
        INC     HL
        LD      D,(HL)
        CALL    IGBLK ;find 1st char on line
        CP      "'"
        JR      NZ,NOLAB
        EX      DE,HL
        EX      (SP),HL ;str ptr
        PUSH    HL
        PUSH    BC
        LD      A,(COUNT)
        LD      B,A
NXTMAT  LD      A,(DE) ;match 2 strs in line
        CP      (HL)
        JR      NZ,NOMTCH
```

```
        INC     HL
        INC     DE
        DJNZ    NXTMAT
        POP     AF ;success!
        POP     AF
        POP     DE ;line no.
        POP     BC
        POP     AF
        POP     HL ;old text ptr.
        JP      IGBLK ;ignore any blanks
NOMTCH  POP     BC ;restore regs
        POP     HL
        EX      (SP),HL
        EX      DE,HL
NOLAB   LD      H,B ;get next line
        LD      L,C
        LD      A,(HL)
        INC     HL
        LD      H,(HL)
        LD      L,A
        JR      NXTLIN ;next line ptr in HL
LABERR  DB      OEF,'Unknown Label',0
        JP      ERROR1
COUNT   DB      0
@*EOF
```


nascom

BRITISH AND BEST

SYSTEMS

48K NASCOM 3.	£549.00
8K NASCOM Micro Ed.	£399.00
NASCOM Single Drive.	£470.00
NASCOM Dual Drive.	£685.00
NASDOS Disc Operating System.	£60.00
CP/M Disc Operating System.	£126.00
SANYO Tape Recorder.	£25.00
12" Green Screen Plastic Monitor.	£99.00
12" Green Screen Metal Monitor.	£120.00
Epson MX80 F/T 3. Printer.	£350.00
TEC 1550 Printer.	£699.00

All Prices exc. VAT.

ADD-ONS

Castle Tape Interface.	£15.00
Milham A/D Converter.	£49.50
Screen Weave Eliminator.	£8.75
Screen Flash Eliminator.	£14.75
Programmer's Aid.	£28.00
Arfon Speech Board.	£85.00
W.S. Big Ears.	£45.00
PMG Soundbox.	£40.50
CMOS 2K Ramboard.	£95.00
CMOS 32K Ramboard.	£185.00
I.O. Systems Pluto high res. Graphics Board.	£399.00
Colourgraphics NASCOM AVC On Demonstration Now.	£185.00
RAM B 64K Upgrade	£30.00

TWO MORE BRITISH COMPUTERS

The new Welsh Dragon 32 with colour and sound. High resolution graphics and 32K user memory.

£199 inc VAT

Program tapes, cartridges and joysticks available.

The very popular Sinclair ZX 81.

£49.50 inc VAT

Printer and program tapes also available.

We will give a generous trade in on a ZX 81. Purchased from us if you later wish to upgrade to a Dragon or a Nascom.

NASCOM
Networking up to 32 Machines on one master.

SOFTWARE
Over 100 items in stock covering business, educational games.

NAS-SYS/NAS-DOS Suppliers -
NASCOM – applications packages.
Dove Computer Services – polished system, software.
Program Power – superb games and utilities.
Mike York – extensions to BASIC.

Business & Leisure Micro Computers

VISIT OUR GREATLY EXPANDED SHOW ROOM
16, The Square, Kenilworth, Warwickshire CV8 1EB.
Telephone: (0926) 512127.

Review of the 32K CMOS Battery-backed RAM Board.

A beautiful piece of work was the first thought that came to mind after seeing this board for the first time a couple of weeks ago. A battery-backed RAM board was one of those things that I used to dream about months ago when the cost was way out of my range. Now, a fully populated board (32K) is available from Microcode (Control) Ltd. for £185 (+ VAT) and well within the price range of all but the most miserly Nascom and Gemini micro owners.

The board arrived well packaged up in acres of corregated card board. The board that I was provided with to review was the fully populated one but it is possible to buy the board populated to 2K or 16K and then add additional 6116s as required (or can be afforded). The board is also available unpopulated but for the price it is hardly worth the hassle of construction which makes the 2K board a better bet.

I am not sure if everybody gets an appropriate piece of wire with which to make up all the link connections that are necessary to configure the board but I did which saved me the hours it would normally have taken me to hunt through our scrap box to find a suitable replacement. Depending on how nimble one is with ones fingers, a piece of wire between 24 inches and 4 foot would be required to completely configure a fully populated board.

The board itself is double-sided, through hole plated, solder resisted and silk-screen printed. It has nice, easy-to-follow tracks and is nicley laid out. The chip numbering (IC numbering) is of the format that that Nascom constructors have grown to love, totally haphazard. The unusual point about the board is that the components are orientated in the opposite direction to the printing and numbering on the board which could cause some confusion. This was evidently due, so I have been told, to the fact that when the board was originally designed, the machine it was designed for had all its boards in upside down, so, to make fault-finding easier in-situ the board was labelled in such a way that the labelling was the right way up when the board was upside down. A minor point really but sure to confuse some constuctors who choose to build up the bare board.

There are 5 different link blocks to wire up which configure the board in one of many ways.

LINK BLOCK A (LKSA) Address boundary option.

This link block is used to select where in the memory map the RAM is to sit. The RAM chips themselves are configured in 4K blocks and each of these blocks can be connected to any of the 16 block decode pins. More than one block of RAM can be connected to the same decode block which then leaves the final decoding to the Page control links. The decode linking is

similar to that used on the RAM A board and is very straight forward.

LINK BLOCK B (LKSB) Page configuration - page number selection.

Once you have decided where you want your RAM (there's no answer to that) the next thing is to decide whether you want the RAM as one 32K page or two 16K pages and then which pages to use (0 to 3). This is the decoding carried out at LKSB. If you choose to have two 16K pages these are assigned as page A and page B and may be placed into the memory map on pages 0 to 3. The explanation provided in the manual for this decoding is probably the hardest part to follow. It is easy to configure your board from the examples provided but the explanation is not very clear and it took a phone call to Microcode and a long hard look at the circuit diagram before the theory emerged. The page selection uses Port FFhex to feed a latch which is linked to the buffered data bus. The low nibble of the byte provides the page read enables and the high nibble, the page write enables.

	W Enable				R Enable		
Data Bus	!7	!6	!5	!4	!3	!2	!1	!0
Page	3	2	1	0	3	2	1	0

Therefore, to put your RAM as Read/Write on pages 0 and 2, connect 1D to bit 0, 2D to bit 4, 3D to bit 2 and 4D to bit 6. This leaves the pages free for other page compatible RAM or EPROM boards.

LINK BLOCK C (LKSC) Page configuration - page reset option

This link block is used to select which, if any, pages will be paged in on reset (including power up). If page B is required on reset then link WEB and REB to any of the Q dash pads. It is possible to have the selected page as Read only on reset in which case only the REB is connected to a Q dash pad. Hardwired Write protect (and Read protect if necessary) can be accomplished by wiring the appropriate WE and/or RE pads to the provided 0 volt pads.

LINK BLOCK D (LKSD) Power option.

This is the beauty of the board. Up until now it can have been regarded purely as 32K of paged RAM. The power option provides the battery-backed links required to turn the RAM into something close to easily programmable/eraseable/changeable EPROM. Each chip is provided with a link option allowing it to be either connected to the backup supply (provided by an on-board VARTEC MEMPAK 3.6v PCB Ni-cad Battery) or connected just to the normal +5v line of the micro. The battery can support the fully populated board for over 40 days and recharges

itself whenever the computer is on. The battery-backup takes over whenever the computer is switched off and the board can be removed, the page links reorganised, etc, etc. A very useful facility.

LINK BLOCK E (LKSE) Nascom I/O decode.

This is provided for Nascom users who have not got the Nas-IO signal decoded anywhere else in their system. This provides all the decoding necessary for port FFhex.

All the link connections made on the board are done so using pieces of wire pushed into gold plated socket pins. Once you have decided how you want your system configuring it is an easy matter (though fiddly at times) to cut the wire to appropriate lengths and link the pads. It is purely a matter of pulling out the appropriate links again in link blocks A, B and/or C to reconfigure the memory map, page layout and page-on-reset. As long as link block D connections, the power options, are not changed, the memory contents remain unchanged. I have carried the board around and left it unplugged for a couple of days without any corrupted data at all. The only thing that you have to be careful with is NOT placing the board on a conductive surface as this can short out the power lines and so cause the odd problem.

The possible uses for the board are endless. One application would be to use a page in Read only mode to develope or modify another monitor system. I have used the board as two 16K pages, one of which contains Wordease and some printer driver routines and the other page purely as normal RAM. It means that I can switch on my machine on a morning, flip a port and then leap enthusiastically into my latest Wordease file without having to reload anything from tape. Easier to use than disc (and faster, and cheaper) and far more versatile than EPROMs.

The board runs at 4Mhz without wait states and if a partially populated board is purchased, any 6116 chips can be used as long as they do not draw more than 100 micro Amps each. Alternatively NMOS RAMs and EPROMs can be put on the board but not connected into the battery-backed circuit.

In summary then, this is an excellent 8 by 8 board which is a viable alternative to an EPROM card as an EPROM programmer/eraser are unnecessary. It is easy to set up, very versatile and simple to use. A simple routine could easily be written to provide a menu driven system to select between the different pages and so utilities/monitors/programs.

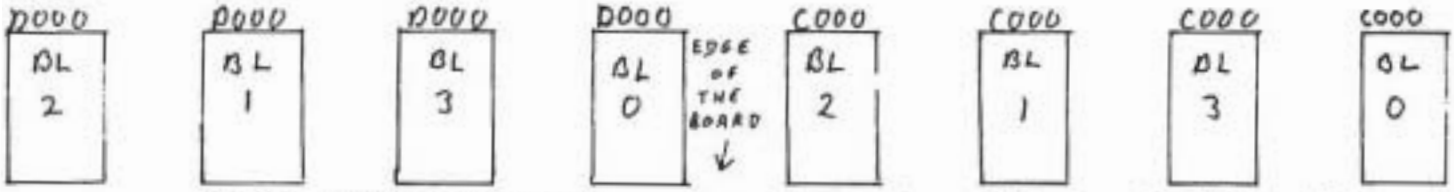
Finally, many thanks to Mr. L. Opit of Microcode for providing me with the board to review and helpfully answering all my questions.

A 64K EPROM Board for £4 ?

by Alf Want

My school has three languages for the Nascom, Nascom Pascal, Forth and Nascom BASIC. We use a 6K assembler, a 4K disassembler/debug, a micro-cassette monitor and our own control programs to make the use of these facilities as easy as possible. We would like to add a word processor, printer routines and perhaps Pilot. It looked as if in the end we would require 2 EPROM boards per machine. Then Mr. D. A. Boyd wrote his article on using 4K 2732 EPROMs. This would half solve our problems so I converted our EPROM burner and fitted Mr. Boyd's conversion. Fortunately it did not work. One pin was wrongly labelled on the diagrams. I say fortunately because it meant I had to work out how the circuit actually worked. In the process I realised that it really is easy to double up and convert the Nascom 2 plus RAM B board into a 44K RAM and 64K EPROM machine. It costs about £4 to do the job and saves the expense of two EPROM cards and two edge connectors. The disadvantage is that you only have 44K of RAM available on a standard Nascom 2 whereas EPROM board owners gloat unceasingly over their 64K.

The principle depends on the 2732 having, in effect, two chip select pins, /OE and /CE, pins 20 and 18. The Nascom 2 selects the 8 on-board memory sockets using the /CE signals generated by IC 46. Pins 3 and 13 of IC 46 can be used as a binary input giving 0 to 3 to select 4 banks of 2 sockets in the following manner.



IC 46 needs lows on pins 2 and 15 for this selection to work, and link switch 1,8 set to 8K. The logic lows are provided by the /XROM signal from the control board. Now is the time to study the circuit diagram with this article with the Nascom Memory circuit to hand.

It should be sufficient to explain the three output signals in detail leaving space for a few extra comments at the end.

1) /XROM When any address outside the EPROM block at C000-FFFF is selected, gate 3, the 'AND' gate, ensures that /XROM is inactive high. RAM is not disableJ and IC 46 is not enabled for bank select. When an EPROM address is selected, /XROM is active low. RAM at that address is disabled while IC 46 is enabled for both banks A and B.

2) /XROME When address E000-FFFF is active low and BS2 (Bank Select) is low (indicating any bank except 4 is selected) then gate 2 ensures that /XROME goes active low. It is looped along the /OE pin 20s of the 8 extra EPROMs. Whether one of these is actually enabled depends on the output of IC 46.

3) /XROMC When address C000-E000 is active low and BS2 is low (any bank selected except 4) then gate 4 ensures that /XROMC goes active low and is looped along link blocks 1 to 8 (NOT 9) pins 10, to connect to the /OE pin 20s of the original EPROMs in the eight Nascom sockets. Again, which one is actually selected depends on the output of IC 46.

4) /BROM When Port 3, Bank 4 is selected, BS2 goes high and gate 2 being an 'OR' gate /XROME must be deselected. Meanwhile, /BS2 is low and gate 1 is an 'OR' so, as long as address is E000+, /BROM is active and BASIC occupies bank 4 E000-FFFF.

5) BS1 & BS0 These send binary signals 0, 1, 2 or 3 to IC 46 to control the bank select. The output is steady depending on the last inputs to D0, D1 and D2 of the 3 flip-flops contained in the 74LS175.

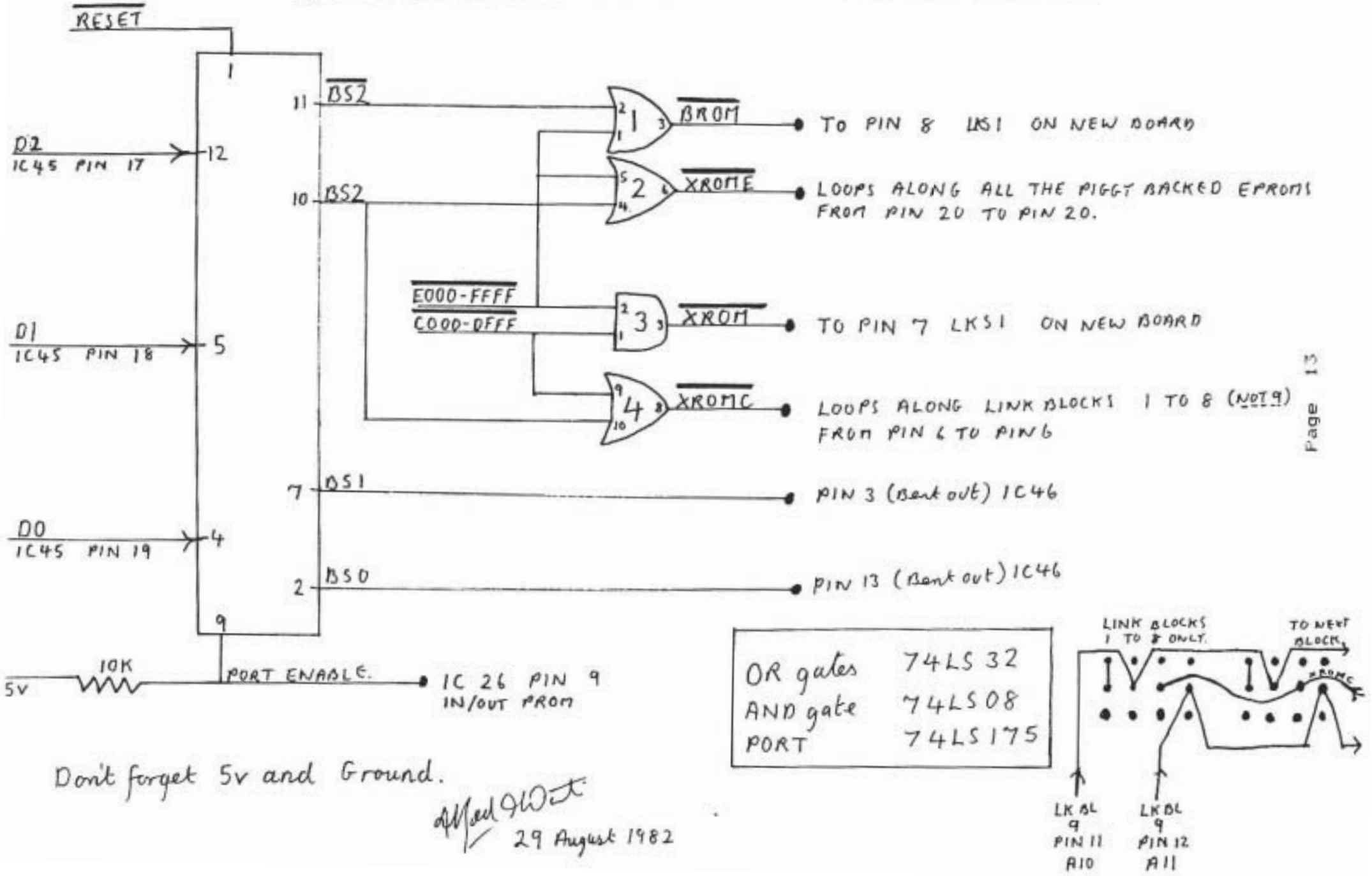
The reset line connected to pin 1 of the 74LS175 ensures that bank 0 is always selected on power-up or reset. Pin 9 may be regarded as a chip select pin, enabled by pin 9 of the IN/OUT PROM IC 26.

Construction takes one evening and is easy but painstaking. Wire wrap sockets are soldered carefully to the 8 EPROMs that are to be selected at C000-DFFF. These go into the eight Nascom sockets. The 8 EPROMs to be selected at E000-FFFF have pin 20 carefully bent out to take a seperate /OE signal. They then piggy-back into the 8 new sockets. Wire wrap sockets are not essential but give some cooling space between the mated EPROMs. I lost no EPROMs doing this.

The control board uses a wire wrap socket to plug into LKS1 in place of the header plug. It therefore holds 4 sockets, 2 fourteen pin and 2 sixteen pin sockets. It has 10 leads to the board. Pins 3 and 13 of IC 46 have to be bent out of their socket before connection. Do not forget the 5 volt and Ground.

I cannot claim to have tested the board fully. I cannot afford enough EPROMs. I have piggy-backed six 2732s and moved them around and can find no faults. Indeed, the board appears to work with no wait states at 4 Meg. I drew a wiring diagram but it was such a mess I thought you would do better without it. I have checked the circuit diagram very carefully to ensure it contains all necessary information. I can only marvel at the flexibility of these fantastic little computers.

CIRCUITS TO CONERT NASCOM 2 TO 64K EPROM



Don't forget 5v and Ground.

29 August 1982

Making a Nascom 1 into a Nascom 2 - so to speak

P. Crabb

This application note is intended as a guide for any soldering iron freak who happens to fall into the same way of thinking as myself. I will not guarantee that what has been accomplished on my system will work nor that it will not destroy what you have scrimped and saved lovingly to build over the last seven or eight years.

The Nascom 1 is an excellent machine even without the addition of Nas-sys, and a lot more powerful than many people realise, but in the light of the Nascom 2, it has somewhat paled, although with the time and effort lost in the 'bankrupcy' period, the total system has somewhat paled in the light of what you can now get for the same cost. In defence of the concept, I must add that what you lose in getting a complete working system from another source is more valuable to me than the money saved, but then I am funny that way. You probably are too.

I acquired my Nascom 1 many years ago and it served several purposes before being consigned to its box to gather dust while I set to work (paid) on a number of different systems. My work eventually moved on to the Nascom 2 as a base for an o.e.m. learning system and interest in the Nascom 1 was regenerated to the extent of acquiring a buffer board and a 16K RAM card.

During my original researches with the Nascom 1, I had expanded to the extent of CCsoft Tiny BASIC which I fitted to the standard machine using the Bits & P.C.s dual monitor board and I must commend this (if it is still available) to anyone who wants a cheap training system for BASIC. But, although I wrote a large number of useful programs (like calculating Pi to several thousand places) with the unexpanded Nascom/CCsoft system, the prospect of having so much extra storage did not appeal to my pioneering spirit, and I looked for a better vehicle to explore the wide-open spaces.

The answer was abviously the ROM BASIC from the Nascom 2 and ,quite by chance, one became available, as did a Nas-sys ROM. This was quite legal, decent and almost honest since when I bought a Nascom 2 CP/M system, these were on the board, and in the price, but quite inaccessible on the memory map.

The problem was incorporating them into the system. There were definitely no funds available for an EPROM board, and even if there had been, there would need to be modifications before the ROM BASIC could be accomodated. The answer to the problems came in a flash of inspiration (and a pool of perspiration) and used the Dual Monitor board.

Here there are four 2708 sockets, each providing access to the data bus, most of the address bus and also power lines.

Latter day ROMs work off 5 volts and take relatively little juice from bus lines, particularly as they have standby modes when not enabled. So, out with the hacksaw and down to work.

The first job is to remove the switches from the Monitor board and you will also need to remake the links to pin 20 (Output Enable) on the two sockets that are plugged through into the Nascom 1.

Next you need to cut the tracks to pin 19 and 21 of the other pair of sockets. These carry the +12 and -5 volts needed by 2708s but not by 2716s nor bigger ROMs.

The 2708 sockets provide all the data bus lines and address lines A0 to A9. To accomodate a 2716, you need to find A10 and connect it to pin 19. A 2732 needs A11 and that has to go to pin 18, and the BASIC ROM needs A12 on pin 21. Check with a data sheet, or any of the many articles on the subject more competent than this one, as to what to do with the pins that are now floating, but you can in practice pick up A10, A11 and A12 from anywhere on the Nascom 1. Suggested places in increasing order of 'safety' are the Z80 CPU socket (pins 40, 1 and 2), the Nascom 1 bus/43-way connector (pins 14, 13 and 9) and lastly the Nas/Gemini/80-bus connectors on the buffered motherboard (pins 40,41 and 42). These are also in increasing order of length, so a final decision must be made as a compromise between physical and electrical properties.

The last part of the modification depends on the RAM board that is fitted to the system and produces the correct enable lines for the ROMs. I use the Nascom RAM B card which has the Nascom 1 connection on the bank-select socket. Sixteen open-collector outputs are used to choose which part of each bank of RAM is mapped to which 4K memory area. The simplest solution, and the one that I tried first, was to connect pin 20 of the BASIC ROM to the E and F outputs of the bank-select. No surgery is required since there is a convenient hole where the switch was taken out of the board.

So, the N1 select goes to block 0, the 16K of RAM goes from 1000H to 4FFFH and the BASIC ROM resides from E000H to FFFFH. Nas-sys in the 2708 sockets and switch on.

The first task was to test the RAM and that produced no problems at all. Nas-sys appeared to work through all its many features and, very happily, when the area of memory from E000H upwards is tabulated, we can see that BASIC is there. Give it a 'J' and it crashes.

Wait states...?? Yes, I have heard of them, but this is not meant to be a technical article. Is the ROM fast enough to work? Can the Z80 read it at 2MHz? It obviously is because it can be tabulated and you can even do a fast copy of its contents into a RAM area with the 'C' command but the 'J' still does not work.

That gave me a clue anyway. Copy it into RAM at E000H and execute the RAM version. Off with the power, out with the soldering iron, check with the manual and swap the selects to the 3000H and 4000H blocks with the E000H and F000H blocks.

Switch on, copy 2000H bytes from 3000H to E000H using the 'C' and give it a 'J'. Garbage appears on the screen, Nas-sys functions irregularly and both the BASIC ROM and the area at E000H tabulate rubbish.

A puff of smoke bellows from each of my ears and I take a fortnight's well-earned nervous breakdown.

I return armed with a firm resolve, sixteen 4116s from another abandoned project that happens to be lying around and decide to populate the RAM B card fully.

The system now works but in the process several discoveries have brought out a little enlightenment.

The system is now configured as follows. Nas-sys is in two 2708s at 0000H with the usual screen and Nascom 1 workspace up to 0FFFH. There are two banks of 16K RAMs from 1000H to 8FFFH which gives me a 32K workspace for the BASIC. 9000H to 9FFFH is empty but is mapped to the 2716 socket on the Dual Monitor board to give two copies in the 4K space. The BASIC ROM resides from A000H to BFFFH and the last bank of RAM runs from C000H to FFFFH.

To work the system in BASIC, it is necessary to perform a 'C A000 E000 2000' followed by a 'J'. There is no reason why the 'J' command can not be doctored to carry out the copying procedure and I intend doing this as soon as possible.

As I had suspected, the ROM was not fast enough to work without wait states, even at 2 MHz, and it appeared that the data lines could not change fast enough during an opcode-fetch cycle to keep up. It was lucky that the Nascom 1 single-step function works in ROM areas as it clearly showed the effect. The first instruction at E000H is JMP E003 or C3 03 E0. When executed in single step mode, the program counter pointed to 0303 and similarly for all three-byte instructions. There is no problem in using the 'C' command at 2 MHz, however, as the LDIR command allows time for the data bus to settle and does not make such speed demands on the slower ROM chip. There is naturally no problem once the BASIC resides in RAM.

Two things will immediately stand out to those of you who have not allowed your computers to take over the art of adding up. There are holes in the memory map and there is usued RAM. Yes, but as you will see, these are features of the system that once were bugs.

The 16 decoded outputs from the memory board are only

partial bank decodes and are designed to activate the 16K bank of memory rather than a specific 4K block as one might have thought from the circuit and documentation. This means that any 16K block of memory can only be properly activated by four adjacent decodes since it requires the lower bits of the address bus to select the memory block. When I connected RAM to the 1000H, 2000H, E000H and F000H decode lines, then I was connecting two blocks of the same bank. A bit complex, but whenever I wrote to 2000H the result appeared in E000H and vice-versa. I shudder to think what happened when I read from 2000H! Nevertheless, as the first thing that BASIC does is clear the memory that it is going to work in, it was inadvertantly rubbing itself out as it went along.

It was also the problem of multi-mapping that caused the first attempt to copy into RAM to fail so miserably. The fact that the decodes are partial and rely on the lower address lines meant that the two decodes going to the BASIC ROM were reversing the two 4K blocks in the ROM. So the 4K block that was sitting at 3000H should have been at F000H and the block at 4000H was meant to be at E000H. Using the single 'C' command to copy the 8K block caused the disaster, but if I had copied each block seperately to the appropriate destination it would have worked fine (subject to the paragraph above).

Looking through the information above, you can probably deduce that the system I started with would have worked... If not, then this is what you should do, given just 16K.

The 16K of RAM needs to be split into two 4K blocks and an 8K block. The latter goes into the area E000H to FFFFH, but only one of the 4K blocks can sit at 1000H. The last block MUST reside at either 4000H, 8000H or C000H because of the partial decodes. The BASIC ROM must sit at 2000H, 6000H or A000H to make the two halves appear in the right order and be moveable with the 'C' command.

Of these, the choice of RAM at C000H and ROM at A000H allows for the greatest expansion of RAM being easily accomodated without changes of software, but it should not be forgotten that there is still a vacant socket (or two) on the Dual Monitor board. With the constant dropping of the price of EPROM, it is not outside the bounds of probability that a 2764 (or two) be installed therein, allowing up to 16K of dedicated programs and data. These could be used to hold ZEAP, NASPEN, DIS-DEBUG and the Bits and P.C.'s Toolkit, or Nascom's PASCAL system, for example.

Doing this would need further study of the layout of your memory, and the requirements of each bit of software used but the guidelines above should be sufficient for the task. If the requirements overlap to any extent, or if the EPROMs prove too slow, then a copy to RAM at the correct place should work.

For my own purposes, I do not so much require programs to be stored as data. In particular, for one application where I want to use big letters on the screen, I plan to place a 2732 version of the Nascom Character Generator into the spare 4K block and use that in place of the conventional DATA statements that take up an immense amount of storage in BASIC programs to perform this function. I will probably at the same time replace some of the less imaginative graphics, such as the teletext pixel patterns, with a set of subroutines that can be called from BASIC or Assembler programs.

One of the last worries that you may have is that your neat, flexible Nascom is getting ROM bound. There is absolutely no reason why this should happen, as long as you do not want to keep changing from one ROM program to another without switching off.

The full 48K of RAM can easily be accomodated alongside 26K of ROM and EPROM that will fit on the Dual Monitor board and the 2K of RAM on the Nascom 1 board. It cannot, however, be accomodated at the same time and, therefore, a simple switch is neede to disable either the RAM or the ROM. This is simply made with a few TTL chips and installed after the decode block to route the signal to either the RAM or ROM. What normally happens is that the ROM is initially enabled until a signal is sent to switch permanently to the RAM. This is called a PHANTOM line and is a common signal in S-100 systems. It can either be a decoded port signal or one of the PIO lines, in which case it can be used to toggle the selected chips and possibly alter the addresses - with a bit of logic.

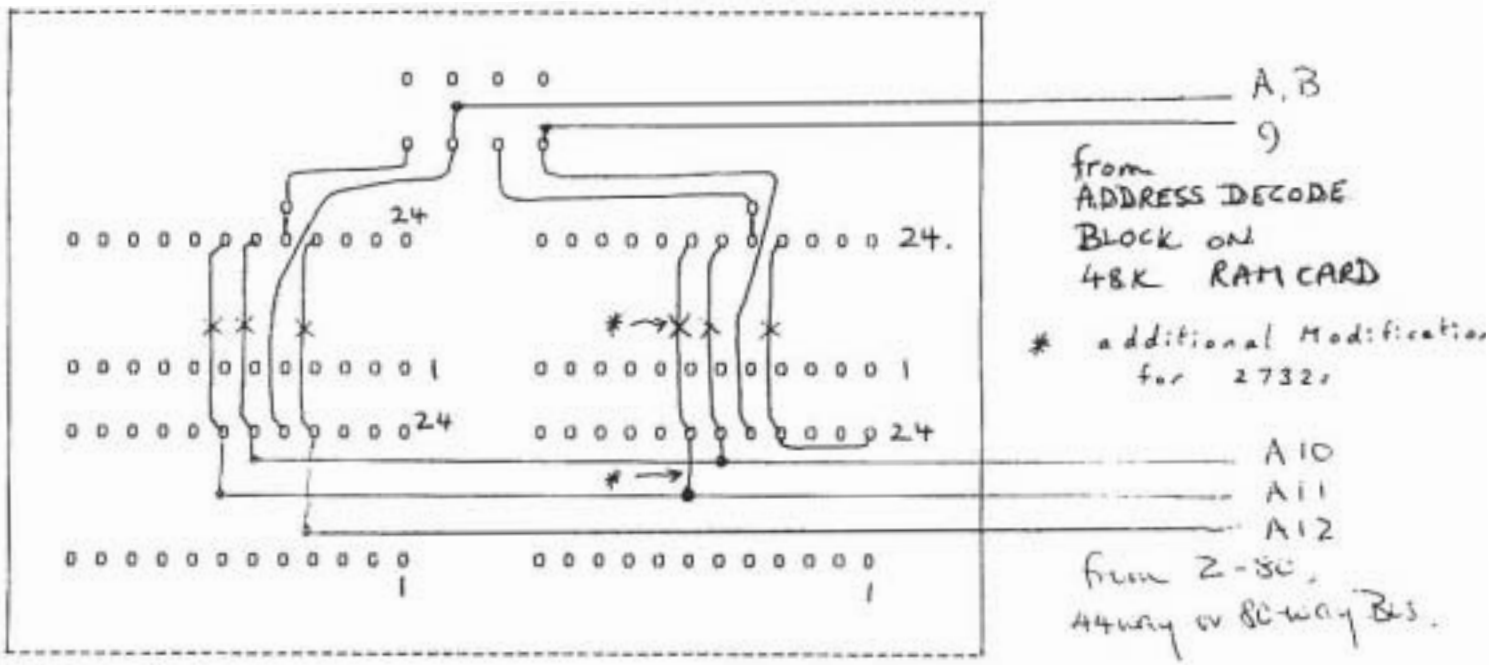
Unless you are content to enable the RAMs once and for all, you should check whether the disabled RAMs are refreshed, if you want to use their contents again anyway.

It is about here that the project gets passed back to you, since my own ideas are getting out of the mainstream of computing and into the specific areas of my interests. If you want to find out about light-pens, Teletext interfaces, piano-keyboards and embroidery designs, then each of these can make up a whole article...another time though.

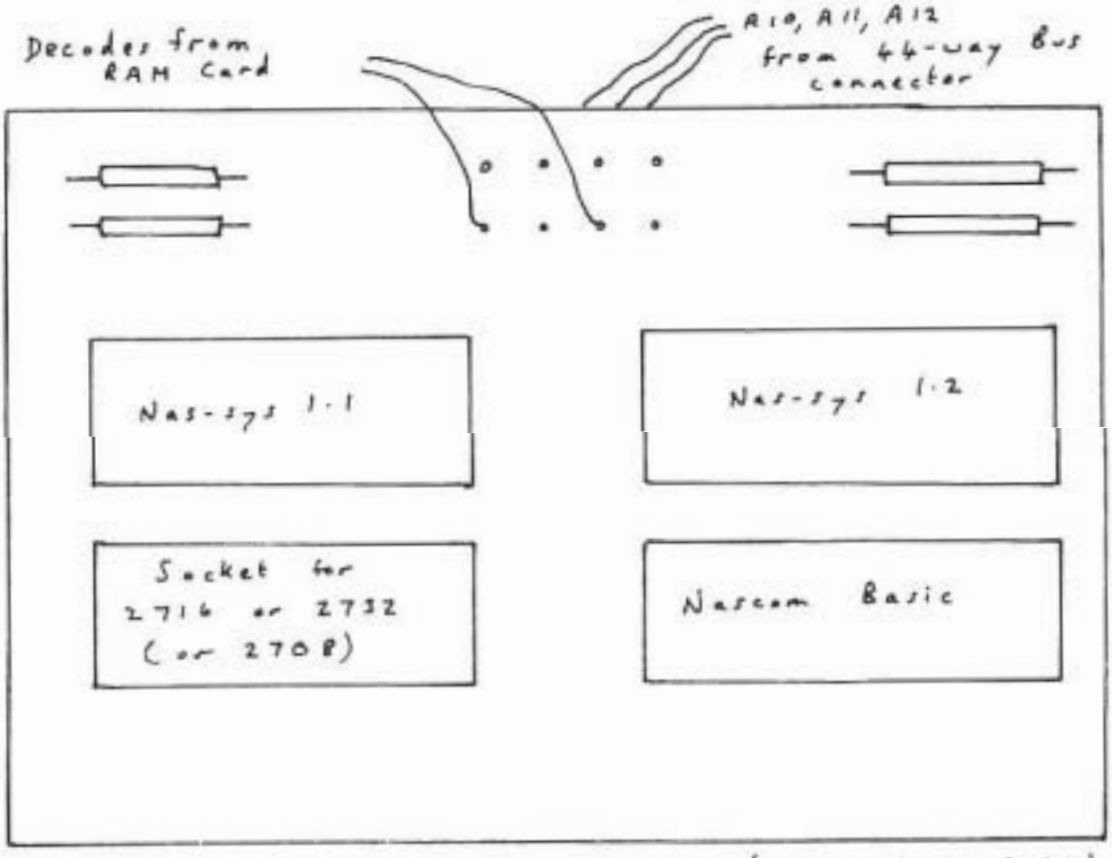
I hope that the above has been of use to you if your Nascom 1 was beginning to feel a little restricted and you thought the only way out was a Sinclair. For the price of a Spectrum, you can implement most of the above starting from a standard Nascom 1 and that will just be the start of a whole new lease of life.

Routine to relocate BASIC and perform cold start

```
21 00 A0  LD HL,£A000; Address of BASIC
11 00 E0  LD DE,£E000; Address to run BASIC
01 00 20  LD BC,£2000; Length of BASIC
DF ,C     ;Copy block
DF ,J     ;Cold start BASIC
```



Bits + P.C.s. Dual Monitor Board (circuit side)



Bits & PCs Dual Monitor Board (Component side)

CMOS Battery-backed RAM Update

Microcode (Control) Limited

Having read Paul Anderson's excellent article: '16K CMOS Memory Extension for the Nascom 2 Main Board', I would just like to mention a few points of interest.

Experience gained in manufacturing the 32K CMOS Battery-backed Board (see our advert. - will the Editor forgive us this 'plug'?) has taught us the following lessons with regard to the standby current of these amazing memory chips (HM6116--).

1. The standby current is extremely temperature dependant and for a particular memory chip this may vary over a ratio of 1000:1; we consider an average of 1 micro-amp per degree centigrade over the zone 15 to 25 degrees C as about typical for the standard 'P' series.

2. The best standby current is obtained by pulling the address, read, write and chip select signals up to V standby while in the standby mode. This obliquely hinted at in the specification sheets for these devices. This is most conveniently done using 10K DIL resistor networks. Without these pull-ups the 'in-circuit' standby current will often reach over 150 micro-amps.

3. In order that the pull-up scheme works, it is wise to use tristate buffers on these lines. However, this introduces yet another current sink, namely the leakage of the output transistor pairs in the tristate drives.

   We use 74LS126 quad tristate non-inverting buffers with active-high enable. Beware of the standard 74126 TTL part, as this has a wicked diode between the output pair and the Vcc to catch the unwary (me!). Also the S.G.S version of the 74LS126 seems to be leakier than other manufacturers.

4. If you want really long data retention periods the standby currents of the 'LP' versions are of at least an order of magnitude better.

   The 'LP' versions are functionally identical but have passed a special 'low standby current' test during the final stage of manufacture. The slightly higher price is well worth the cost in terms of this performance in our opinion.

5. Just recently the Toshiba equivalent device (TC5517AP) has dropped in price to compete with the Hitachi chip. The Toshiba chip does seem to be far superior in all parameters and our own tests would seem to verify the manufacturers claim.

With regard to data retention, the following points should be noted:

Data corruption is more frequent while Vcc is rising (during power-up) than while Vcc is dropping (during power-down), however it can happen during either transient phase. Corruption usually takes the form of one or two bits in a few (apparently random) locations being set high.

The causes are many and various but, as Paul Anderson points out, are predominantly due to the unpredictable behaviour of the Bus control signals during the transient phase of power-up and power-down. Corruption is not always easily detected, we ourselves use a check-summing routine after having initialised memory contents with the pattern AAhex 55hex.

Those readers who have purchased our memory boards may have wondered why on receipt your board is full of AA 55. If you really want to be smart you can run this program at 2000hex by selecting IC 11-12 to block 2 (if you did not overwrite it in your enthusiasm to use the board).

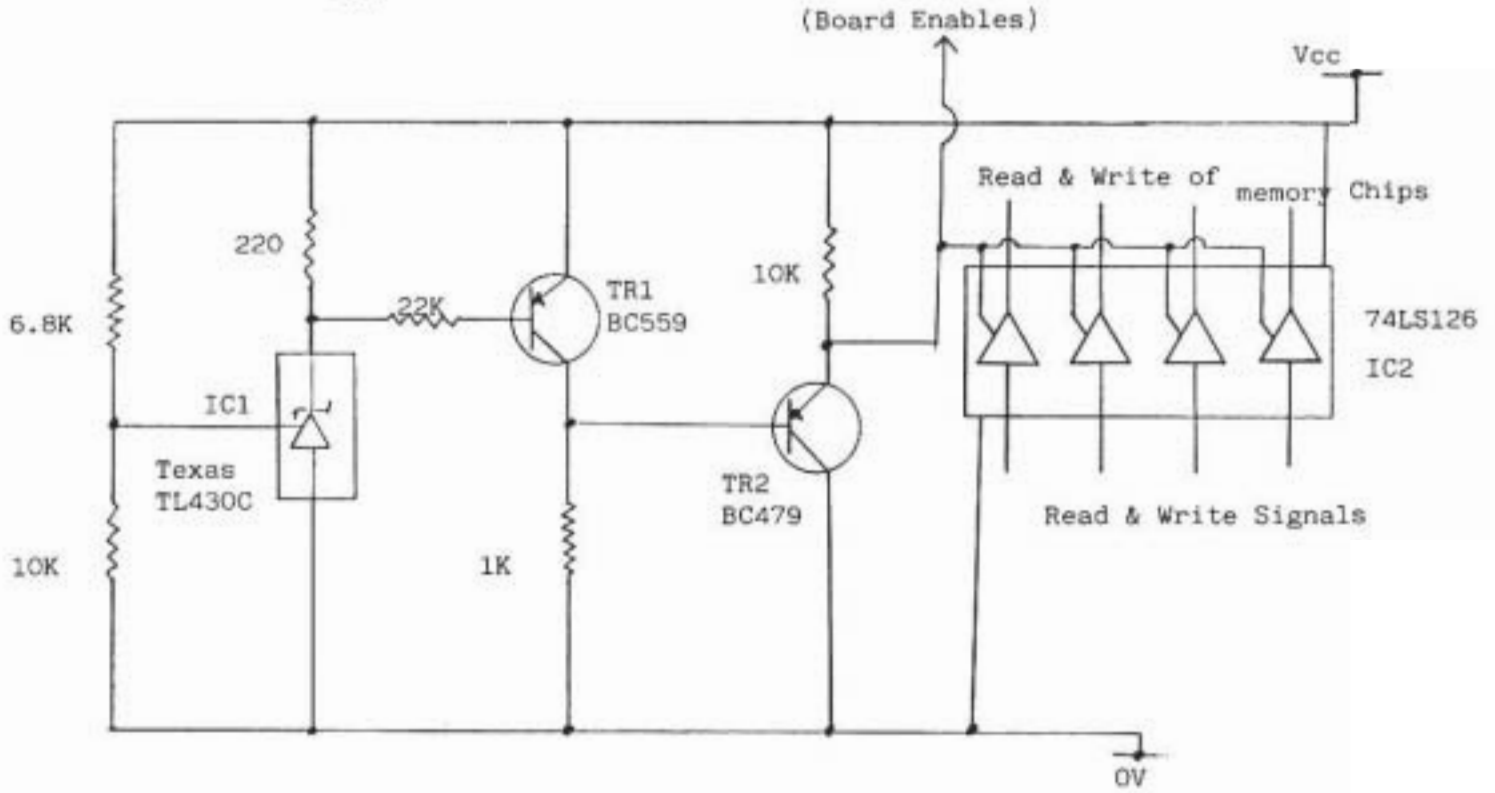
To overcome this danger of corruption, we developed a fast switch circuit to disable read and write lines whenever Vcc falls below 4.6 volts. (see diagram) The circuit is based about a programmable zener (IC 1) set to 4.6 volts. The reason we use this particular device in preference to a normal zener is due to the fact that its dynamic slope impedance is only 3 ohms. as a result, as soon as Vcc rises above the programmed zener voltage (4.6v) the zener draws current through TR1's base, switching TR1 on and TR2 off.

TR2 is switched on whenever Vcc is above 0,6 volts and below 4.6 volts and while switched on pulls the commoned enables of the tristate buffers low, thus disabling all read and writes. The circuit switches in under 2 micro seconds.

I hope this information will have been of some interest and, perhaps, use to your readers who intend utilising these delightful chips. Non-volatility is an asset that, once gained, forces one to think how on earth one survived without it.

Lastly, as a concession to Micro Power readers, we will offer the HM6116LP-3 (150nS access time) at £4.50 and the Toshiba TC5517AP (150nS access time) at £4.95 up until January 1983 to readers who enclose the cut-off section from our advert. in Micro Power.

© Microcode (control) Limited
1982



Low voltage detecting switch for battery backed memory systems

Nascom & Gemini USERS

NEW 32K C.M.O.S. BATTERY BACKED RAM BOARD

FEATURES:

EFFECTIVELY REPLACES EPROMS.
Does away with the inconvenience of EPROM programming and the compromise of assigning valuable address space to ROM.

ON BOARD RE-CHARGEABLE Ni-Cad BATTERY RETAINS MEMORY FOR OVER 1000 Hrs.
Battery is automatically charged during power-up periods.

HIGH SPEED OPERATION up to 6 MHz WITHOUT WAIT-STATES.

FULLY NASBUS[1] and GEMINI-80 BUS[2] COMPATIBLE.

GE MODE SCHEME SUPPORTED.
...e board can be configured to provide one 32k byte page or two completely independent 16k byte pages.
Complete pages of 64k bytes are simply implemented by adding more boards on to the bus.

SOFTWARE and/or HARDWARE READ/WRITE PROTECTION.
4K blocks in either page are link selectable to be aligned on any 4K boundary.

FULLY BUFFERED ADDRESS, DATA AND CONTROL SIGNALS.

MEMORY I.C. SOCKETS ARE LINK SELECTABLE TO SUPPORT ANY 24 PIN 2k byte MEMORY I.C.s.
Thus the board can support up to 32k bytes of any mixture of cmos, nmos rams or 2716/2516 eproms.
All options are link selectable using wire links plugged into gold-plated socket pins, avoiding the risk of damage and the inconvenience caused by soldering links directly to the board.
The printed circuit board is manufactured to the high quality demanded by industrial users and conforms to B.S.9000.
The board comes assembled and tested and is supplied with a minimum of 2k bytes of low-power cmos ram. Fully documented.

AVAILABLE NOW!

PRICES:

Board with 2k bytes	£99.95
Board with 16k bytes	£149.95
...oard with 32k bytes	£184.95
...are Boards (circuit diagram supplied)	£45.00
HM6116-LP/3 Very low power 2k cmos memory I.C.	£6.50

Please add 95p P&P and VAT @ 15%

[1]nasbus is a trademark of nascom microcomputers a division of LUCAS LOGIC
[2]trademark of GEMINI MICROCOMPUTERS LIMITED

cheques and P.O.s to:
MICROCODE (CONTROL) LTD.
41a MOOR LANE, CLITHEROE, LANCS. BB7 1BE.
For further information Phone (0200) 27890

Please supply me with the following: Price

Total enclosed £

Address:

POST CODE

256Kb RAM CARD

FOR NASCOM/GEMINI 80 BUS SYSTEMS

8" x 8" Plug in card with memory arranged in four 64k blocks. Buy it now as a standard 64k ram card and expand it later by simply adding more ram chips.

Versatile page mode options or more extensive memory mapping facilities.

Operation at 4 MHz without wait states.

Software support available for CP/M virtual disk - Speed up those programs that access the disk excessively.

NOW AVAILABLE IN KIT FORM

Bare PCB with full instructions £42-50
64k Kit complete £105
64k Card fully built and tested £150
64k Expansion kits P.O.A.

All prices are exclusive of VAT.
Add £1.50 for carriage and packing.

COMING SOON
Low cost 80 column video card for Nascom.

MAP 80 SYSTEMS LTD.,
333, Garret Lane, London, SW18
Tel: 01-874-2691

PARKSTONE ELECTRICS

NASCOM AUTHORISED DEALER

GEMINI PRODUCTS ALSO STOCKED

Computer Systems Built for Special Applications

Personal PEARL now available on NASCOM
Create all your business programs for less than £200

PARKSTONE ELECTRICS
18, Station Road,
Lower Parkstone,
Poole, Dorset, BH14 8UB.
Tel: Parkstone (0202) 746555

Here are two more 'pretty picture' plotting programs. The first is from Mr. D.P. Jackson of Harworth, Nr. Doncaster.

```
10 CLS:P=45:L=1:T=1:GOTO 30
20 P=180:T=2
30 Y=20:Z=40:FOR W=1 TO 20:FOR C=1 TO P
40 B=SIN(C)*Y+21:A=COS(C)*Z+45:ON T GOSUB 110,120
50 NEXT:Y=Y-1:Z=Z-2:NEXT:FOR D=16 TO 1 STEP -5
60 FOR A=17-D TO 72+D:FOR B=D-1 TO 42-D
70 GOSUB 110:NEXT B,A,D:L=L+0.5:IF L<2 GOTO 30
80 IF L<3 THEN L=4:GOTO 20
90 L=L-1:IF L=3 THEN L=1:T=1:P=45
100 GOTO 30
110 E=POINT(A,B):IF E=1 THEN RESET(A,B):RETURN
120 SET(A,B):RETURN
```

The second offering comes from Mr. N.J. Thomas of Kings Lynn and is as follows.

```
10 CLS:H=95:V=44
20 X1=H/2:X2=X1*X1:Y1=V/2:Y2=V/4
30 FOR X=0 TO X1:X4=X*X:M=-Y1
40 A=SQR(X2-X4)
50 FOR I=-A TO A STEP V/10
60 R=SQR(X4+I*I)/X1
70 F=(R-1)*SIN(R*12):REM THIS IS THE LINE TO CHANGE
80 Y=I/5+F*Y2
90 IF Y<=M THEN 120
100 M=Y:Y=Y1-Y
110 SET(X1-X,Y):SET(X1+X,Y)
120 NEXT I,X
130 END
```

Mr. Thomas also points out that anyone who has a Nascom 1 and Econographics will run into problems when trying to fit the Snowdinger (1) circuit. They both try and sit over the same place. He suggests that Snowdinger be brought out on ribbon cable to hang beside the PIO.

Finally, can anyone tie up the PIO pin outs of the Nascom 1 and 2 so that the EPROM Programmer circuit can be built by Nascom 1 owners.

Besides articles for the magazine, these are the only two letters that I have had since the last issue. I don't think that you are really trying. A nice feature to add to the magazine would be a permanent letters page in which we could set up a problem and answer series. Fire away.

2 Megabyte Nascom

Ian J Clemmett

The possibility of having 2 megabytes of RAM hung onto a Nascom 2 is remarkable but with the new 256K RAM boards from M.A.P.-80 Systems this becomes a reality. You will require a large motherboard of some desciption with at least 9 slots (8 for the RAM cards and 1 for the Nascom main board) but suprisingly little money.

When I was first contacted about reviewing this new board I imagined that it would be more a case of the RAM controlling the Nascom but I was pleasantly suprised to find that the 256K RAM cards were a convenient 8 x 8 with nothing more than 4 banks of 64K RAM chips and some decode and paging circuitry. I had been expecting at least an additional processor but, unless it was disguised as a 74LS series chip, there definitely wasn't one.

All products have a distinquishing feature of some kind or other and this board is no exception. Its blue. None of your common or garden green or grey but blue. (The writing paper that they sent me was blue as well.)

I was sent, or rather had delivered, the prototype of the board included in a Gemini disc system but I was also shown the card working in a Nascom (my own Nascom in fact). The Gemini system consisted of a processor card, video card, FDC and the 256K RAM card. I was also loaned 2 disc drives so that I could see the card under the control of CP/M.

As I said earlier, the RAM is arranged in four 64K blocks made up of 4864 chips. The minimum configuration is the board populated to only 64K but it is a simple matter of adding the additional RAM to make the board up to 256K but as the chips are quite pricey upgrading should be done carefully.

As the Z80 is only capable of directly addressing 64K paging must be used to allow access to the RAM. Its not much use having 2 Meg. if you can not use it. The page allocation is done using 2 link blocks which select which 64K block is on which page. The page control is performed via port FE which leaves port FF free for use by other cards eg. RAM B card. The normal Nascom paging system only allows 4 pages but the 256K card gives a five bit decode which gives the user 32 'pages' to play with which works out at 2 Megabytes. The RAM card works with most Nascom/Gemini ROM/RAM cards but, unfortunately, your faithful RAM A cards are incompatible though I would imagine that if a bit of paging could be added to them they could be reprieved. If you want to use your G802 board in the system in the 32K paged mode you need to do a minor hardware mod but nothing difficult or complicated.

The bits on port FE are assigned as follows:

```
bit 0   SET - selects upper 32K block
        RESET - selects lower 32K block
          (not operative if bit 7 is RESET)
bit 1 )
bit 2 )
bit 3 )  selects 64K page
bit 4 )    (bit 5 not used in 32K mode)
bit 5 )
bit 6   SET - select upper 32K of page 0 as permanent
        RESET - select lower 32K of page 0 as permanent
          (not operative if bit 7 is RESET)
bit 7   SET - select 32K mode
        RESET - select 64K mode
```

There are 3 modes of operation:
1) 64K paging - the memory is set up in 64K blocks which means that to change pages, a program must ensure that the Program Counter holds the value of the next instruction on the incoming page after the O FE XX is executed. Not the easiest of things.
2) 32K paging - in this mode each page is divided into two 32K pages and the user can then assign either the upper or lower 32K of page 0 as permanent and then page in any other 32K half page. This makes the controlling program much simpler and would be very us_ful for applications where large data files are manipulated by small programs.
3) Memory-mapping - this is the most powerful mode of all and it allows any 4K block from the total RAM to be 'mapped' into any 4K slot in the Z80 memory map. The 'source' block can be mapped into more than one 'target' block.

The switching from page to page on the RAM card can become very confusing and a memory management program would be very useful. This is exactly what M.A.P.-80 have done. If you buy a RAM board and send them your CP/M, they will modify the CP/M so that it treats any RAM over 64K as a virtual disk. This allows CP/M to support its normal disks plus an additional 'disk', P, which is accessed in exactly the same way as the real disks, only much faster. 'Disk' P has all the normal features of a CP/M disk, including a directory, but doesn't suffer as much from diskette wear and tear or head alignment. It makes making back-up copies of real disks easy when you can only afford one disk drive. Additional 32K blocks of RAM can be reserved from the CP/M to enable the user to have more than 64K of RAM.

Unfortunately, the board is incompatible with Nascom 1s but I think this is its only negetive point on an otherwise well designed and supported product. At £150 for the basic 64K board, it must be well within the reach of most users. It is purely a matter of plugging in the additional memory chips to move up to the 256K and beyond barrier.

Many thanks to Mr. M. Rothery for delivering the review system and showing me how it worked and also to Mr. P. Chance and Mr. T. Watkins for their technical help.

4Ks Via 2 Megs.

by A. Want

How to convert the popular Gemini/Bits and PCs EPROM Programmer to program 2732s and 2716s instead of 2708s and 2716s, at a cost of two 1M resistors.

That excellent article by D. A. Boyd (Micropower, Volume 2, Number 2) showing how to put 32K of EPROM on a Nascom 2 board was exactly what I needed for our school Nascom 2s. Trouble is that our Gemini/Bits and PCs EPROM programmer only blows 2708s and 2716s, or does it? This article explains how to convert this very popular 'blower' into a 2716/2732 model. The same warning applies as in D. A. Boyd's article, it will not work with 2532s, although making it do so would be easier than it was for the 2732. I chose the more difficult route because the 2732 is now available at just over £4 for the 350 nsec version. Having had timing problems with 2708s, I thought the extra 50-100 nsecs needed by the 2532 might be critical. (It shouldn't be I know).

The 2732 is programmed by:

1) Placing 25 volts, steady, on pin 18, known as /OE Vpp.
2) Putting the right address on the 12 address pins A0 to A11.
   Putting the eight bits of the required data on the eight data pins D0 to D7.
4) When all the above are stable, drop pin 20, known as /CE, to 0 volts for 50 milli-secs from its standby voltage of 5 volts.

Thanks to the superb documentation with the Gemini/Bits and PCs blower, I realised that this was very easily arranged. (Yes, I did say 'superb documentation'.) Mind you, it took a week of evenings to sort out, but then I am not an electrical engineer or a programmer.

Hardware Modifications.

Please do not write in saying how hamfisted these are, it is easy and it works. First study the switch diagram in Figure 1, the IC pin out diagram in Figure 2 and the voltage/pin out in Figure 3. Do this in conjunction with the circuit diagrams supplied with the board. Note that the pins of switch 2 are labelled as if it were an IC, orientated like the other ICs on the board. Now, work as follows...

1) On the top of the board, cut the two thick tracks between the two switches close to switch 2. On the underside of the board, close to switch 2, cut two thin tracks, one going to pin 6 of the switch (the furthest from the edge of the board) and the other to pin 3 (the track that crosses the switch). All four 2708 switch pins are now 'floating' and ready to become 2732 pins.

2) Short out the pulse capacitor, C10, from the old 2708 25 volt pulse circuit shown on sheet 3 of the circuit diagrams. Take a lead from 'A' of that diagram (I used the end of R5 furthest from TR2) to pin 3 of switch 2. This will provide 25 volts on pin 20 of the recipient EPROM.

3) Lift IC 2. Bend out pins 7 and 14. Use one of your two 1 Meg resistors to pull pin 7 down to 0 volts by connecting it between pin 7 and the wide track close to pin 7. Pin 7 must also be connected to Bit 6 of port 5. This is easiest done by using the through-plated hole next to the C7 legend. Pin 14 is left unconnected. Bit 6 of port 5 now controls a 25 volt/0 volt output to pin 20 of the recipient EPROM.

4) Connect together pins 8 and 9 of swotch 2 to put A10 on pin 19 of both EPROMs.

5) Fortunately, IC 3 produces an A11 signal on pin 1. I bent pin 1 of IC 3 out and connected it to pin 11 of the same switch to provide A11 on pin 21 of both the donor and recipient EPROMs.

6) Join Bit 2 of port 5 to pin 6 of switch 2. This is best achieved by using the plated hole just by the TR6 legend and close to the switch. I also pulled pin 6 of the switch to 5 volts with the second 1 Meg resistor. I used the legend end of R20 for my 5 volts.

Your board is now a 2732 programmer when switch 2 is pushed in and a 2716 programmer when switch 2 is out, but only when the software is correct.

Software Modifications.

Thank you Bits and PCs for your fully documented listing. I am most grateful. First, the functions of the bits 0 to 6 of the control word that is used to control the EPROM blower.

Bit	Function		Name
0	0 volts	Increments address counter clock	ACLOCK
0	5 volts	No increment	
1	0 volts	Enables address counter clock	RESET
1	5 volts	Resets address counter clock	
2	0 volts	2732 programming pulse of 0v	RP2732
2	5 volts	2732 standby, 5v on/CE	
3	0 volts	Enables donor ROM /OE	MROM
3	5 volts	Disables donor ROM	
4	0 volts	Disables 2716 5v programming pulse	TRIG
4	5 volts	Triggers 2716 5v programming pulse	
5	0 volts	Enables 2716 receptor ROM /OE	RW2716
5	5 volts	Disables 2716 output from /OE	
6	0 volts	Read enable 2732, 0v to /OE Vpp	V25TRG
6	5 volts	25 volts to 2732, 25v to /OE Vpp	

NB. Bit 7 of the control word is used by the software only. When it is set, 2716 EPROMs are in use, when reset, 2732 EPROMs are present. Bit 7 is called TBIT.

Now the listing alterations needed to work your board. I use the original line numbers, like BASIC, to insert extra lines where needed. Even without an assembler you should be able to follow the few changes needed and alter your code accordingly. If not, send me the original cassette and stamps, I will re-record it for you. I hope that this does not upset anybody as you can not be altering a board you do not have and I am not quite copying the original software.

```
0480 RP2732 EQU  £2
0570 V25TRG EQU  £6
0790        DEFM /ROM type 2716 or 2732/
0950        DEFM /11E8 Blow a ROM(fully erased)/
1030        DEFM /1343 Blow a ROM(not erased)/
1110        DEFM /12C0 Verify a ROM/
1200        DEFM /12A8 Load data from donor to RAM/
1260        DEFM /12A2 Check for fully erased ROM/
1520        LD   A,£32
1550        LD   DE,4096    ;No. of locations in ROM
1740        LD   A,£2F      ;0010 1111 CTRLWD
2450        LD   B,01       ;2732 cycles
2520 Delete line
2530 Delete line
2642        SET  V25TRG,A   ;25 volts for 2732
2643        LD   C,A        ;Save new control word in C
2685        RES  RP2732,A   ;Trigger 0v programming pulse 2732
2830        JP   NZ,DEL16
2834        LD   DE,£C00    ;C00 gives about 50 msec delay  for
2732
2836        JP   GODEL
2840 DEL16  LD   DE,£555    ;Delay for 2716 pulse
2965        LD   A,C        ;Get control word
2966        RES  V25TRG,A   ;25v off 2732
2967        LD   C,A        ;Save initial word again
3150 Delete line
3160        RES  RP2732,A   ;Set 2732 recipient to read
3170        RES  RW2716,A   ;Set 2716 recipient to read
3450 Delete line
3460        RES  RP2732,A   ;Set 2732 recipient to read
3560 Delete line
3570        SET  RP2732,A   ;Deselect 2732 recipient
```

It takes about four minutes to blow a 2732. I have burned eight 2732 EPROMs from RAM and copied the eight again. I have had no errors. I am really pleased at having saved my school the cost of two £60 EPROM boards and a £60 EPROM burner, at the expense of two 1 Meg resistors. Not bad eh! Incidently Mr. Boyd, how can I piggy-back another 4 banks of EPROM on top of my present four to give eight banks of 8K plus BASIC? I am serious, it would really be useful to us. Oh, and if you are ever in the

Maldon area it would be a pleasure to buy you a drink.

My address if you would like a tape of the new software doing is:-

8, St. Stephens Road,
Cold Norton,
Chelmsford,
Essex.

Blow away Peter, blow away Paul...

FIG 1

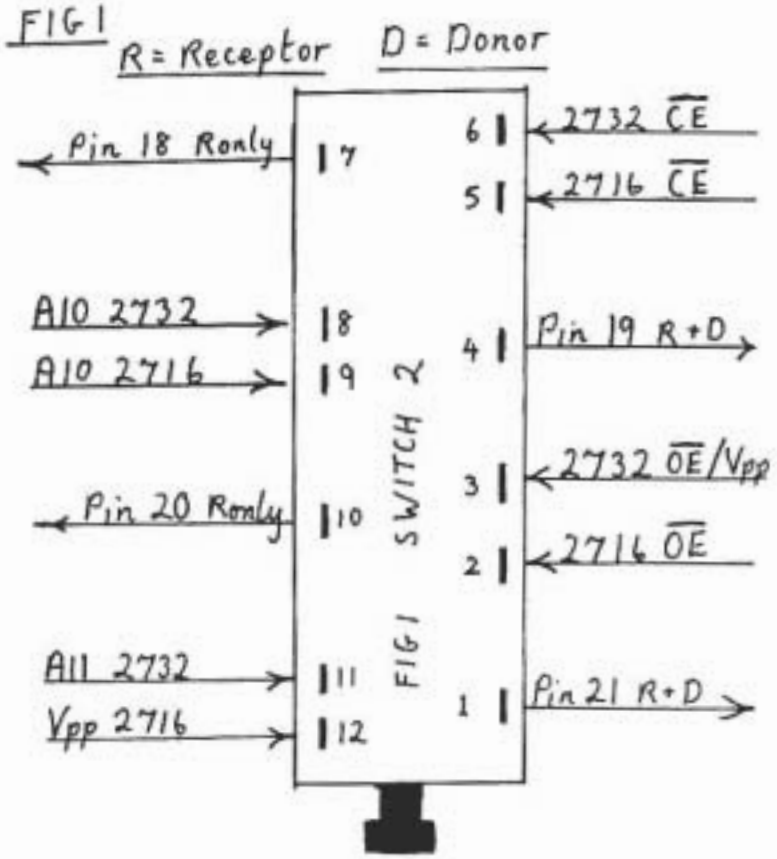


FIG 2

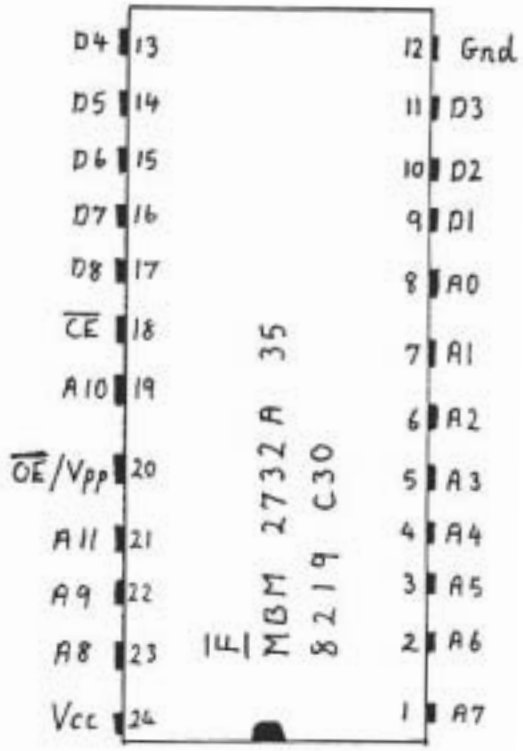


FIG 3

V25 must be less than 26v at all times
DAMAGE WILL BE DONE TO 2732 IF Vpp IS CHANGED WHILE $\overline{CE}$ IS 0V

CHIP	PIN	FUNCTION	READ	PROGRAM	STANDBY
2716	18	$\overline{CE}$	0V	5V pulse	0v
2732	18	$\overline{CE}$	0V	0V pulse	5v
2716	19	A10	0v/5v	0v/5v	
2732	19	A10	0v/5v	0v/5v	
2716	20	$\overline{OE}$	0V	5v	5v
2732	20	$\overline{OE}$ /Vpp	0v	25v	0v
2716	21	Vpp	25v	25v	25v
2732	21	A11	0v/5v	0v/5v	

Errata

1) Snowdinger 2

Mr. Williams of Shoeburyness, Essex found the 2 deliberate mistakes in the Snowdinger 2 circuit diagram contained in the last issue. The labelling on IC49 pins 2 and 6 somehow got reversed which means that pin 2 was really pin 6 and pin 6 was really pin 2. Also pin 1 of IC3 should have been connected to pin 6 of IC49 (2MHz and not 8MHz as shown).

Mr. Williams also points out that to run at 4MHz a 220ohm pull-up resistor is required on pin 9 of IC1.

Many thanks to Mr. Williams for pointing out these errors, its nice to get letters even if they only tell me that I made a botchup of the previous issue.

2) The Private Ads.

FOR SALE - Nascom 2 . 48K RAM . Gemini 64K + EPROM Card . ZEAP, NASDIS, DEBUG, EXT BASIC, TELETEXT COLOUR . 4.8K Tape . Verocase . Enormous Software & Firmware Library . Interested? Phone Peter Smith . Petersfield (0730) 4059 W/E and Evenings.

FOR SALE - Nascom 2, 48K, Boxed with Teletype, Monitor and Cassette. Over £250 worth of software including Pascal, Extension BASIC, Chess, Toolkit, Assembler, Disassembler, Word Processor and many Games.
Forced Sale - £450 the lot!
Tel. Leeds 742347

FOR SALE - Nascom 2 with ZEAP, NASDIS, standard graphics chip, RAM B card containing 48K of memory and power supply. £300
Telephone 01-540-0793 after 6pm.

STILL FOR SALE - After a fantastic response to my advert in the last issue of Micropower, I still have 2 RAM A cards for sale. Any offers? Tel. Ian on Leeds 683186

FOR SALE - EPROM programmer. Bits & PCs programmer for 2708s and 2716s. Fully assembled with application software. £20. Also 6 x 2708s fullu erased £6.
Ring Keith Brown on Colchester (0206) 841293

3) In the last issue of Micropower I included a Hex dump of an Othello program. I have since been advised that Othello is a registered trademark of Mine of Information. I apologise for any infringement.

4) I don't know if anyone actually noticed in the last issue but the Editor got changed. John Haigh who had been editing the magazine since it started has allowed me to take over the job and have the privilege of sitting up till all hours of the

night typing articles into Wordease. Having struggled to produce the last one and a half issues I can admire him for managing to keep the magazine going for the last year. I hope that I can do as well.

5) Nasprint Multiple Copies

This note came from Mr. S. Stubbs of Inverurie, Aberdeenshire.

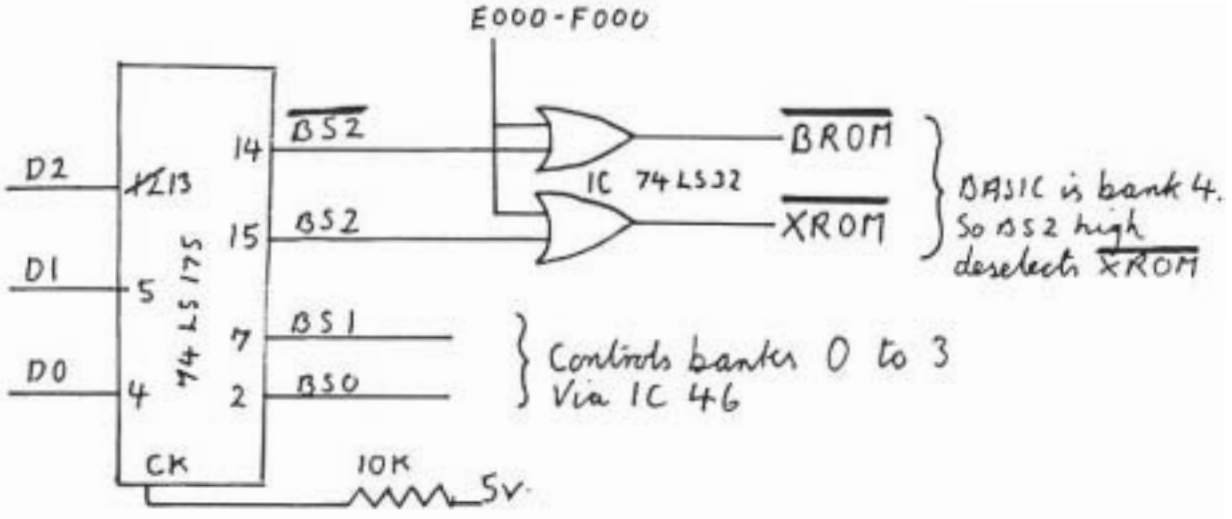
One of the more obvious ommissions in Naspen is the ability to print multiple copies of the same thing, without having to press 'P' every time. I produce about 100 copies of a 3000 word club newsletter every month so I use the following piece of relocateable code which holds in B the number (in hex) of copies required and calls the print routine in Naspen through a loop. The code can go anywhere except in Naspen workspace. I use 0C80 hex.

```
0C80  06 3A        LD B,£3A   ;58 copies wanted
0C82  CD C2 BA     CALL £BAC2 ;Naspen print routine
0C85  10 FB        DJNZ £0C82 ;Loop back 58 times
0C87  C3 06 B8     JP   £B806 ;Return to Naspen
```

To use, just enter this code having set the second byte for the number of copies in hex (up to 255 or FF hex) and execute at the address loaded. Ensure that when you exit from Naspen you leave the cursor pointing at the first character to be printed or use the 'Z' command before leaving if you want the whole text file printing. As the loop executes, you will see 'completed' printed on the screen after every copy. This is because it is within the print routine in Naspen and before the RETurn command.

6) Modifying the Nascom 2 for 2732 EPROMs Amendment

Mr. A. Want spotted this error in a drawing in Volume 2, Number 2 in the 2732 EPROM article. D2 going into the 74LS175 has been assigned to pin 12 in the diagram but should be assigned to pin 13 to work.



nascom MEANS SOLUTIONS nascom MEANS PERFORMANCE

Special Offer until October, 1982

Case Disc System – (includes: disc drive(s), case, power supply, cables and controller)

Single Disc (complete 0.35 MB £419 SAVE £69
Dual Disc (complete 0.70 Mb) £549 SAVE £136

Nascom have come a long way since their acquisition by Lucas. With the knowledge of over 30,000 units already in the field you can buy with confidence from NASCOM.

PRODUCTS:
We have kits, built and tested boards, and our fully assembled and tested NASCOM 3 system with a full choice of configuration either cassette or disc based. Alternative operating systems include NAS DOS and CP/M.

SOFTWARE:
We have a team of programmers who are writing software and courseware especially for UK educational business and domestic users.

FREE ADVICE:
We have appointed experts to advise on the specialist use of micro computers in U.K. schools, homes or businesses.

BACK-UP:
We have a nationwide dealer network giving full sales back-up and after sales service. From our head office we have a service line to sort out any problems.

SYSTEM EXPANSION:
NASCOM machines are designed to grow with users. Easily and simply NASCOM systems can be expanded by adding extra modules to the basic system.

LUCAS LOGIC LIMITED
NASCOM MICROCOMPUTERS DIVISION,
Welton Road, Wedgnock Industrial Estate,
Warwick CV34 5PZ, England.

Learn more about NASCOM now. Complete the coupon for further information and a full list of dealers.

Have you a NASCOM programme? If so send for an application form for inclusion in our programme book

Dealer Enquiries Welcome

£1,548 + VAT
A typical system as shown consisting of a Nascom 3 (48K), dual disks, monitor, and printer is available from your dealer now.

To Lucas Logic Ltd., Nascom Microcomputers Division, Welton Road, Wedgnock Industrial Estate, Warwick CV34 5PZ, England

Please send:
Literature ☐ Dealer List ☐ Prog. Book Form ☐
Name .....
Position .....
Establishment .....
Address .....
.....
Tel. No .....

PP2

Lucas Logic

Hole in the Wall

by Z.R. Lesiakowski

The offering of a program for this issue comes from Mr. Lesiakowski of Trowbridge, Wiltshire and is along the lines of a tennis type TV game. It is machine code (obviously) and requires Nas-sys 3.
Happy typing!

```
1000 31 00 10 CD A7 15 CD 76 13 CD 4B 18 CD A0 13 CD   1..M'.Mv.MK.M .M
1010 9D 11 06 00 CD 51 15 DF 5D CD F1 11 CD 9E 12 CD   ....MQ._]Mq.M..M
1020 FC 12 CD E0 13 28 07 CD E0 13 28 02 18 EB CD 51   ¦.M'.(.M'.(..kMQ
1030 15 F5 C5 ED 5B E2 19 CD 18 11 DF 5D ED 5B DE 19   .uEm[b.M.._]m[^.
1040 CD 59 11 ED 5B E0 19 CD 59 11 C1 F1 CA 62 10 CB   MY.m['.MY.AqJb.K
1050 00 DC BB 13 D4 C9 13 CD A0 13 AF 32 E4 19 DF 5D   .\¦.TI.M ./2d._]
1060 18 B7 21 DA 0B 22 29 0C 0E 14 CB 00 30 20 22 29   .7!Z.")...K.0 ")
1070 0C EF 50 4C 41 59 45 52 20 41 20 48 41 53 20 57   .oPLAYER A HAS W
1080 4F 4E 00 CD 05 11 CD EC 10 0D 20 E2 18 1E 22 29   ON.M..Ml.. b..")
1090 0C EF 50 4C 41 59 45 52 20 42 20 48 41 53 20 57   .oPLAYER B HAS W
10A0 4F 4E 00 CD 05 11 CD EC 10 0D 20 E2 DF 5D 3E 0C   ON.M..Ml.. b_]>.
10B0 F7 21 92 09 22 29 0C EF 57 6F 75 6C 64 20 79 6F   w!..").oWould yo
10C0 75 20 6C 69 6B 65 20 61 6E 6F 74 68 65 72 20 67   u like another g
10D0 61 6D 65 20 28 59 2F 4E 29 20 3F 00 CF 06 59 B8   ame (Y/N) ?.O.Y8
10E0 CA 06 10 06 4E B8 C2 B1 10 C3 99 12 22 29 0C EF   J...N8B1.C..").o
10F0 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20
1100 00 CD 05 11 C9 16 20 FF 15 20 FC C9 D5 CD 3C 11   .M..I. .. ¦IUM<.
1110 CD 33 11 CD 43 FF D1 C9 D5 CD 3C 11 CD 33 11 CD   M3.MC.QIUM<.M3.M
1120 58 FF D1 C9 D5 C5 CD 3C 11 CD 33 11 46 CD ED FF   X.QIUEM<.M3.FMm.
1130 C1 D1 C9 6A 26 00 E5 16 00 C3 31 FF 7B FE 2D 30   AQIj&.e..C1.{~-0
1140 05 3E 2C 93 5F C9 3E 5C 93 5F C9 CD 0C 11 1C CD   .>,._I>\._IM...M
1150 0C 11 1D 1D CD 0C 11 1C C9 CD 18 11 1C CD 18 11   ....M...IM...M..
1160 1D 1D CD 18 11 1C C9 3A DD 19 CB 47 28 02 1C C9   ..M...I:].KG(..I
1170 CB 4F 28 02 1D C9 CB 57 28 02 15 C9 CB 5F 28 02   KO(..IKW(..IK_(.
1180 14 C9 CB 67 28 03 15 1D C9 CB 6F 28 03 15 1C C9   .IKg(...IKo(...I
1190 CB 77 28 03 14 1C C9 CB 7F C8 14 1D C9 3E C9 21   Kw(...IK.H..I>I!
11A0 0A 08 CD CA 11 3E E4 21 8A 0B CD CA 11 3E FF 21   ..MJ.>d!..MJ.>.!
11B0 0A 08 CD D1 11 3E FF 21 39 08 CD D1 11 21 4A 09   ..MQ.>.!9.MQ.!J.
11C0 CD DB 11 21 79 09 CD DB 11 C9 06 30 77 23 10 FC   M[.!y.M[.I.Ow£.¦
11D0 C9 06 0F 11 40 00 77 19 10 FC C9 11 40 00 3E 20   I...@.w..¦I.@.>
11E0 06 05 77 19 10 FC C9 3E 2B BB C8 3E 01 BB C8 B7   ..w..¦I>+¦H>.¦H7
11F0 C9 DF 62 D0 DD 21 E4 19 1E 55 BB CA 98 12 21 F1   I_bPJ!d..U¦J..!q
1200 19 5E DD 56 00 BB 20 07 7A E6 0F DD 77 00 C9 21   .^]V.¦ .zf.]w.I!
1210 F0 19 5E BB 20 18 CB 6A 28 08 CB AA CB E2 DD 72   p.^¦ .Kj(.K*Kb]r
1220 00 C9 CB 62 C0 7A E6 0F CB EF DD 77 00 C9 21 F2   .IKb@zf.Ko]w.I!r
1230 19 5E BB 20 18 CB 72 28 08 CB B2 CB FA DD 72 00   .^¦ .Kr(.K2Kz]r.
1240 C9 CB 7A C0 7A E6 0F CB F7 DD 77 00 C9 21 F4 19   IKz@zf.Kw]w.I!t.
1250 5E BB 20 07 7A E6 F0 DD 77 00 C9 21 F3 19 5E BB   ^¦ .zfp]w.I!s.^¦
1260 20 18 CB 4A 28 08 CB 8A CB C2 DD 72 00 C9 CB 42    .KJ(.K.KB]r.IKB
1270 C0 7A E6 F0 CB CF DD 77 00 C9 21 F5 19 5E BB C0   @zfpKO]w.I!u.^¦@
1280 CB 52 28 08 CB 92 CB DA DD 72 00 C9 CB 5A C0 7A   KR(.K.KZ]r.IKZ@z
1290 E6 F0 CB D7 DD 77 00 C9 E1 3E 0C F7 DF 5B 3A E4   fpKW]w.Ia>.w_[:d
12A0 19 E6 60 28 0B 21 E5 19 35 C0 3A E7 19 77 18 13   .f'(.!e.5@:g.w..
12B0 3A E4 19 E6 90 C8 21 E5 19 35 35 CD D7 13 C0 3A   :d.f.H!e.55MW.@:
```

```
12C0 E7 19 77 3A E4 19 E6 30 28 19 ED 5B DE 19 1C CD   g.w:d.f0(.m[^..M
12D0 E7 11 C8 1D CD 59 11 1C CD 4B 11 ED 53 DE 19 CD   g.H.MY..MK.mS^.M
12E0 5A 13 C9 ED 5B DE 19 1D CD E7 11 C8 1C CD 59 11   Z.Im[^..Mg.H.MY.
12F0 1D CD 4B 11 ED 53 DE 19 CD 5A 13 C9 3A E4 19 E6   .MK.mS^.MZ.I:d.f
1300 06 28 0B 21 E6 19 35 C0 3A E7 19 77 18 13 3A E4   .(.!f.5@:g.w..:d
1310 19 E6 09 C8 21 E6 19 35 35 CD D7 13 C0 3A E7 19   .f.H!f.55MW.@:g.
1320 77 3A E4 19 E6 03 28 19 ED 5B E0 19 1C CD E7 11   w:d.f.(.m[`..Mg.
1330 C8 1D CD 59 11 1C CD 4B 11 ED 53 E0 19 CD 68 13   H.MY..MK.mS`.Mh.
1340 C9 ED 5B E0 19 1D CD E7 11 C8 1C CD 59 11 1D CD   Im[`..Mg.H.MY..M
1350 4B 11 ED 53 E0 19 CD 68 13 C9 7B F5 3C 3C 32 E9   K.mS`.Mh.I{u<<2i
1360 19 F1 3D 3D 32 EA 19 C9 7B F5 3C 3C 32 EB 19 F1   .q==2j.I{u<<2k.q
1370 3D 3D 32 EC 19 C9 01 06 00 21 D7 19 11 F0 19 ED   ==2l.I...!W..p.m
1380 B0 AF 32 E4 19 ED 5F 0F 47 3E 04 CB 07 05 28 07   0/2d.m_.G>.K..(.
1390 CB 0F 05 28 02 18 F4 CB 5F CC C9 13 C4 BB 13 C9   K..(..tK_LI.D;.I
13A0 16 0A 1E 16 ED 53 DE 19 CD 4B 11 CD 5A 13 16 55   ....mS^.MK.MZ..U
13B0 ED 53 E0 19 CD 4B 11 CD 68 13 C9 3E 08 32 DD 19   mS`.MK.Mh.I>.2].
13C0 1E 16 16 0B ED 53 E2 19 C9 3E 04 32 DD 19 1E 16   ....mSb.I>.2]...
13D0 16 54 ED 53 E2 19 C9 7E B7 C8 7E EE 01 C8 B7 C9   .TmSb.I~7H~n.H7I
13E0 21 ED 19 35 C2 A2 14 3A E8 19 77 ED 5B E2 19 CD   !m.5B".:h.wm[b.M
13F0 0B 15 CA A6 14 CD 67 11 CD 46 15 C8 CD 24 11 C2   ..J&.Mg.MF.HM$.B
1400 91 14 FD 21 DD 19 FD 7E 00 E6 34 FD 46 00 28 40   ..)!],)~.f4)F.(@
1410 3E 2C BB 20 10 1D CD 24 11 20 06 CB 00 CB 00 18   >,; ..M$. .K.K..
1420 6D CB 08 18 69 3E 00 BB 20 10 1C CD 24 11 20 06   mK..i>.; ..M$. .
1430 CB 00 CB 00 18 58 CB 00 18 54 CB 50 28 04 CB 00   K.K..XK..TKP(.K.
1440 18 4C CB 60 28 06 CB A0 CB F8 18 42 CB 00 18 3E   .LK`(.K Kx.BK..>
1450 3E 2C BB 20 10 1D CD 24 11 20 06 CB 08 CB 08 18   >,; ..M$. .K.K..
1460 2D CB 00 18 29 3E 00 BB 20 10 1C CD 24 11 20 06   -K..)>.; ..M$. .
1470 CB 08 CB 08 18 18 CB 08 18 14 CB 58 28 04 CB 08   K.K...K...KX(.K.
1480 18 0C CB 78 28 06 CB B8 CB E0 18 02 CB 08 FD 70   ..Kx(.K8K`..K.}p
1490 00 ED 5B E2 19 CD 18 11 CD 67 11 CD 0C 11 ED 53   .m[b.M..Mg.M..mS
14A0 E2 19 3E 01 B7 C9 21 DD 19 7E CB 60 20 46 CB 68   b.>.7I!].~K` FKh
14B0 20 2E CB 70 20 14 CB 57 28 04 EE 44 18 4A CB 67    .Kp .KW(.nD.JKg
14C0 28 04 CB 0F 18 42 CB 07 18 3E CB 57 28 04 EE 84   (.K..BK..>KW(.n.
14D0 18 36 CB 6F 28 06 CB 0F CB 0F 18 2C EE 90 18 28   .6Ko(.K.K..,n..(
14E0 CB 5F 28 04 EE 28 18 20 CB 7F 28 04 EE 84 18 18   K_(.n(. K.(.n...
14F0 CB 0F 18 14 CB 5F 28 04 CB 07 18 0C CB 77 28 04   K...K_(.K...Kw(.
1500 EE 44 18 04 EE 90 18 00 77 18 86 3E 0A 01 DD 19   nD..n...w..>..].
1510 BA 28 09 3E 55 BA 28 19 3E 01 B7 C9 0A E6 34 28   :(.>U:(.>.7I.f4(
1520 F7 3A E9 19 BB 06 80 C8 3A EA 19 BB 06 40 C8 18   w:i.;..H:j.;.@H.
1530 E7 0A E6 C8 28 E2 3A EB 19 BB 06 20 C8 3A EC 19   g.fH(b:k.;. H:l.
1540 BB 06 10 C8 18 D2 AF BA 06 40 C8 3E 5F BA 06 80   ;..H.R/:.@H>_:..
1550 C9 DD 21 EE 19 78 C5 B7 20 08 DD 77 00 DD 77 01   I]!n.xE7 .Jw.Jw.
1560 18 0C CB 17 30 05 DD 34 00 18 03 DD 34 01 01 CC   ..K.0.]4...]4..L
1570 0B ED 43 29 0C EF 53 63 6F 72 65 20 41 20 00 DD   .mC).oScore A .]
1580 7E 00 27 DF 68 01 EE 0B ED 43 29 0C EF 53 63 6F   ~.'_h.n.mC).oSco
1590 72 65 20 42 20 00 DD 7E 01 27 DF 68 3E 0F C1 DD   re B .]~.'_h>.A]
15A0 BE 00 C8 DD BE 01 C9 3E 0C F7 11 D5 0B ED 53 29   >.H]>.I>.w.U.mS)
15B0 0C EF 2A 2A 20 20 48 6F 6C 65 20 49 6E 20 54 68   .o**  Hole In Th
15C0 65 20 57 61 6C 6C 20 20 2A 2A 00 11 1B 08 ED 53   e Wall  **....mS
15D0 29 0C EF B1 20 5A 52 4C 20 31 39 38 32 20 B0 00   ).o1 ZRL 1982 0.
15E0 11 BC 08 ED 53 29 0C EF 54 68 69 73 20 67 61 6D   ...mS).oThis gam
15F0 65 20 69 73 20 66 6F 72 20 74 77 6F 20 70 6C 61   e is for two pla
1600 79 65 72 73 20 61 6E 64 20 72 65 71 69 72 65 73   yers and reqires
1610 20 75 73 65 20 20 20 6F 66 20 74 68 65 20 20 42    use   of the  B
1620 41 53 49 43 20 26 20 47 72 61 70 68 69 63 73 20   ASIC & Graphics
1630 52 4F 4D 53 2E 20 20 54 68 65 20 6F 62 6A 65 63   ROMS.  The objec
1640 74 20 6F 66 20 20 20 74 68 65 20 67 61 6D 65 20   t of   the game
```

PROGRAMPOWER PROGRAMPOWER PROGRAMPOWER PROGRAMPOWER PROGRAMPOWER PROGRAMPOWER

* * * NEWSFLASH * * *

NASCOM SOFTWARE PRICES are

T U M B L I N G

Look Special Low Prices

SPACE GAMES

Moon Raider	£ 6.95
Invasion Earth	£ 6.95
Startrek II	£ 6.95
Galaxian Attack	£ 6.95
Lunar Lander Supreme	£ 6.95
Jailbreak in Space	£ 6.95
New Fase	£ 6.95
Starship Command	£ 6.95
Super Startrek	£ 4.95
Cliff Invasion	£ 4.95
Alien Labyrinth	£ 4.95

OTHER GAMES

Sargon Chess Pack	£35.00
The Keys of Kraal	£ 6.95
Dungeon Quest	£ 5.95
Backgammon	£ 5.95
Draughts	£ 5.95
Graphic Golf	£ 4.95
Eldorado Gold	£ 4.95
Serpent	£ 4.95

APPLICATIONS SOFTWARE

Nasprint '80	£12.95
Wordease WP (2532/2732)	£19.50
(2716 £21.50/2708 £24.50)	
Basic File Handler	£12.95
Wirral Pilot	£12.50
XTAL BASIC	£35.00
Nascount Finance	£ 6.95
Club Membership	£ 6.95
Vortex	£ 6.95
Constellation	£ 4.95
Music Box	£ 4.95
Indexed File Handler	£ 4.95
Mini Toolbox	£ 4.95
Nascii	£ 4.95
Sound Chip Demo	£ 4.95
Mathspack	£ 5.95

HARDWARE etc.

Cottis Blandford	£14.90
AY-3-8910 Interface	£10.95
AY-3-8910 Sound Chip	£ 6.45
Games Graphics ROM	£ 7.45
Graphics Rom Adaptor	£ 7.45

N.B. Special Offer DEDUCT £1 per program when ordering two or more.

Please add VAT at 15% plus 55 pence per order for post & packing.

MICRO POWER Ltd., 8/8A Regent Street
Chapel Allerton, Leeds LS7 4PE

PROGRAMPOWER PROGRAMPOWER PROGRAMPOWER PROGRAMPOWER PROGRAMPOWER PROGRAMPOWER