# 'HI-RES' DISPLAY MOD.

M.J. Bauer
Division of Computing & Mathematics
Deakin University, Victoria, Australia

In the editorial of DREAMer No 10, Graeme suggested that I had abandoned the idea of higher resolution graphics due to problems with hardware design. Well, this is really not the case at all; (and I think he knows this, but he thought he might be able to provoke some action from me. The trick worked, Graeme! )

The facts are these: (1) the hardware modification exists (I developed it ages ago) to give a 128 x 64 dot display. It's a somewhat messy modification requiring quite a bit of PCB track cutting and rewiring, but not too difficult for an experienced enthusiast. But I was very reluctant to release the modifications because NONE of the programs written for the DREAM, including the CHIPOS monitor, will run on a modified version! A new EPROM is required which contains modified display routines. A DREAM-6800 which has a modified 128x64 display (as given here) will not function at all with the original CHIPOS EPROM. Even when you have a suitable EPROM installed for use with the extended graphics, none of your existing programs (games, etc) will run properly, because of the new screen format. Some CHIP-8 programs might work in the upper LHS of the screen, but most programs will go completely haywire! For example, DREAM INVADERS would require a complete rewrite, not just a few simple patches, to work on a 128x64 grid.

So, before you embark on any hardware mods, you should convince yourself that your standard DREAM-6800 has outlived its usefulness! Further, it is recommended that you do not attempt this upgrade unless you are reasonably experienced and have a good understanding of how the hardware works (because you might have to trace a fault).

## POST SCRIPT

Having performed the conversion and tested it myself (including the EPROM patches), I am now a lot more enthusiastic about it than before. (The article was scheduled for the August issue, but there were bugs in it.) It was an unexpected pleasure when the 4014 was seen to operate at 4 MHz! The 74LS165 works too, but the dot widths are not as consistent as the 4014 because the latter is synchronous loading (a very minor point indeed).

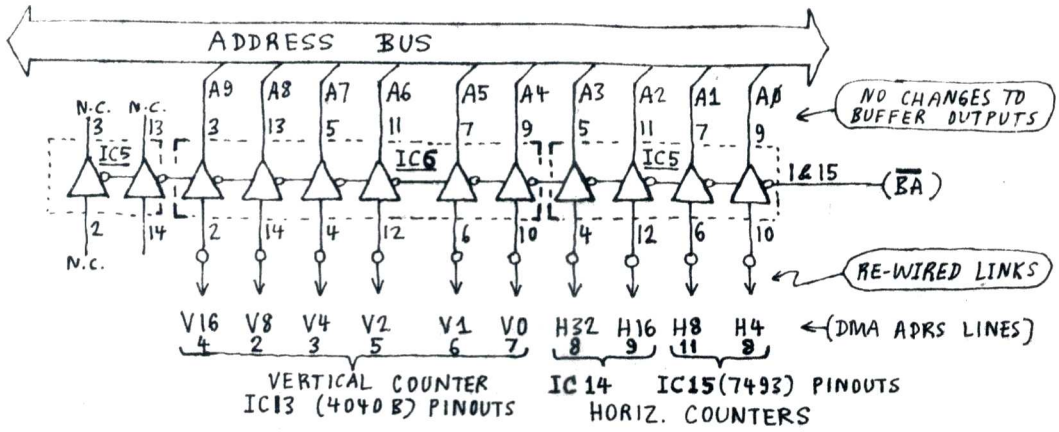
If there is sufficient demand, I could be persuaded to write a new operating-system EPROM for hi-res DREAMs, which would contain a 'software switch' to select the desired display format — the old 64 x 32, or the new 128 x 64. Thus, the modified system would be able to run CHIP-8 programs written for the standard Dream or the 'hi-res' Dream (simply by using a different command for 'GO'). However, machine-code programs which used the display would, in general, NOT be transportable from one system to the other without extensive modification. This probably won't be of major concern. The new operating-system EPROM — let's call it 'HIDIOS' (HI-res Dream Improved Operating System) — would use a 2716 (2K) and would incorporate several enhancements to the original CHIPOS.

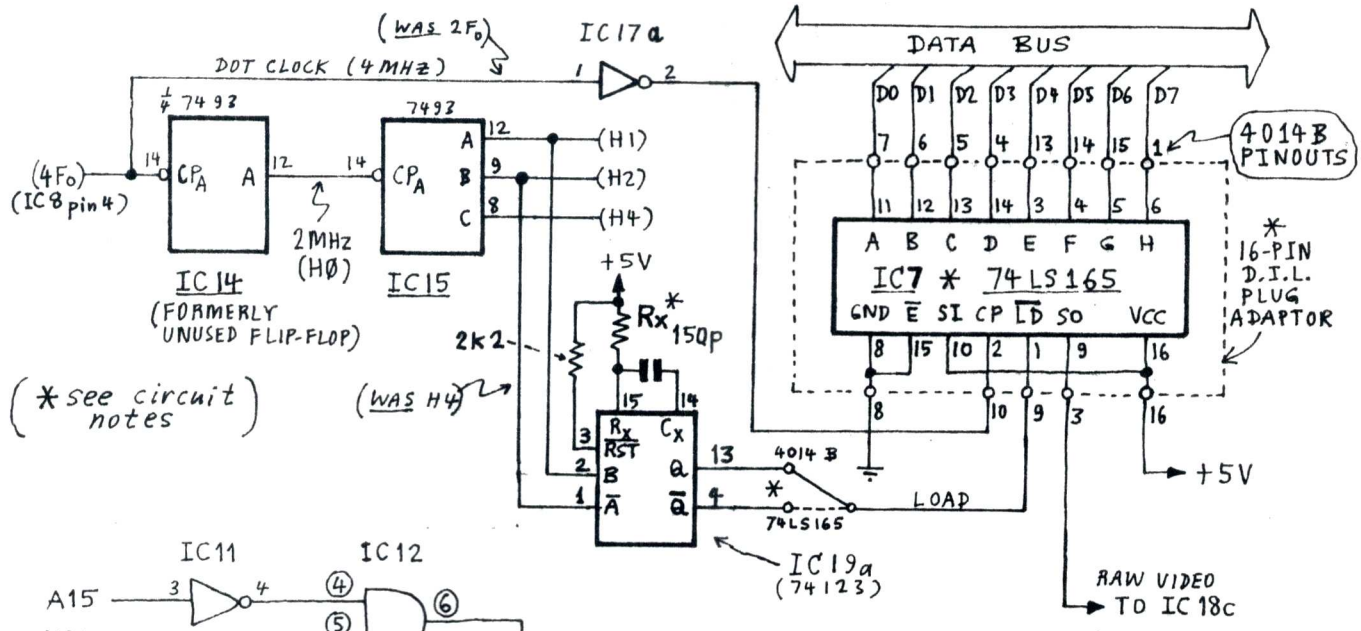
*NB: The last paragraph is reproduced for historical interest only... The EPROM upgrade is no longer available.

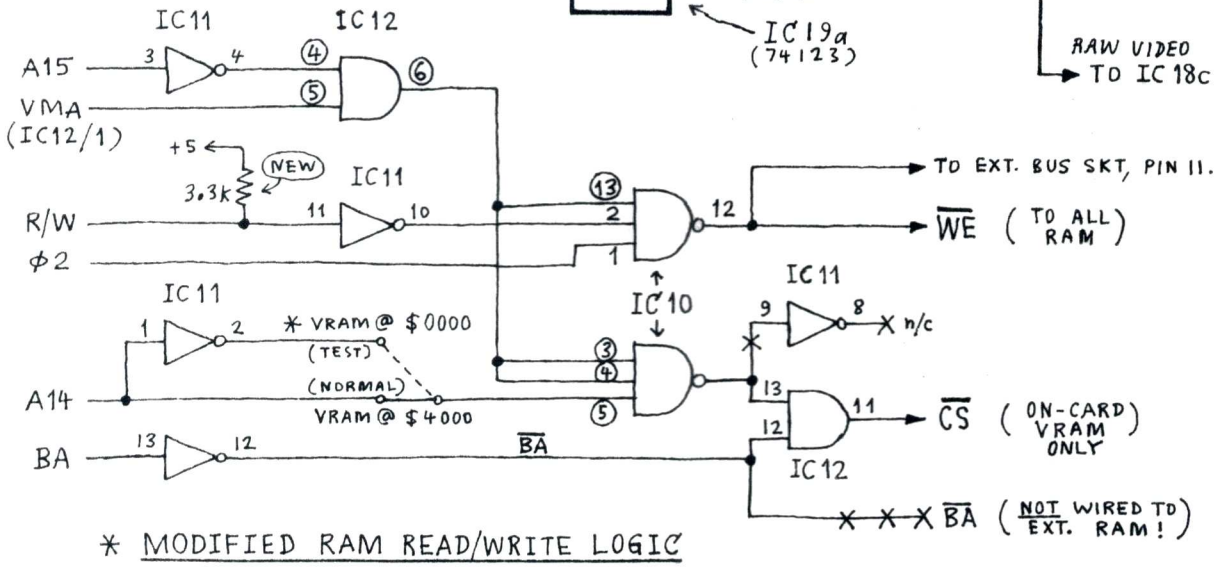
# CIRCUIT MODIFICATIONS

## FOR DOUBLE-RESOLUTION DISPLAY ON DREAM-6800

(124 H x 64 V DOTS)

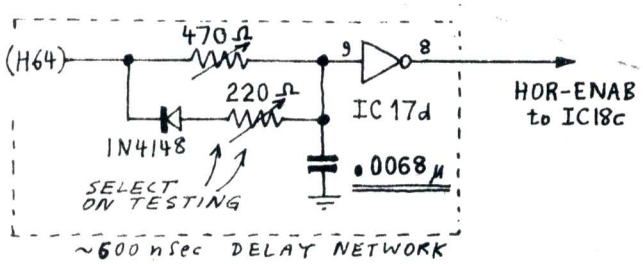


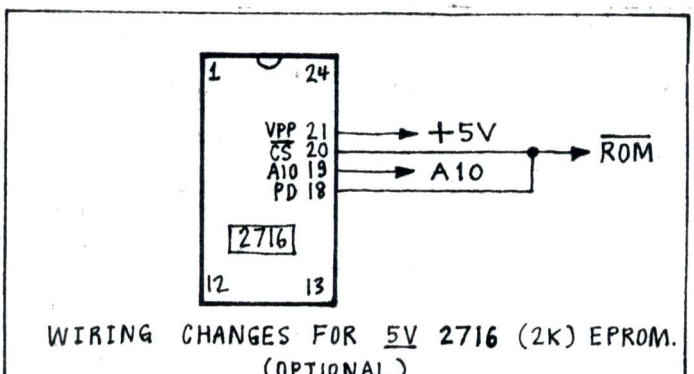




* MODIFIED RAM READ/WRITE LOGIC

( ⓝ = NEW CONNECTION )



## CIRCUIT NOTES

(1) The 'dot clock' freq. is doubled to 4.00 MHz (6875/pin 4).

(2) The video memory buffer (VRAM) is now all of the on-board 1K, which has been relocated to $4000 - $4400. (NB: The RAM space from $4400 - $8000 is not available for expansion.)

(3) At least 1K of external RAM is required at $0000 for user programs and scratch. 4K - 8K is recommended. The E/A or J-R expansion boards may be used, provided they are modified so that the '$\overline{BA}$' signal (Bus Available) does NOT select the lowest 1K block; (see note elsewhere).

(4) The original 4014 shift-register could well perform satisfactorily at 4 MHz (mine did), so try it first. The 4014 requires a 300 nSec (± 50 nSec) positive-going LOAD pulse. Use Rx = 4.7k. It is strongly recommended to wire 8 x 10k pullup resistors on the data lines (D0-D7). Also, tie 4014 pin 11 to Vcc.

(5) If your 4014 can't hack the pace, then a 74LS165 may be substituted, as shown on the circuit. The pinouts are different, so the 74LS165 must be piggy-backed onto a 16 pin DIL plug and jumper-wired using (preferably) 30 guage 'Kynar' wire. In this case, the LOAD pulse required is about 125 nSec (± 50 nSec) negative-going. Use Rx = 1k in the one-shot (IC19).

(6) The 'horizontal-enable' delay network will no doubt need to be adjusted. Use trimpots as shown. This is not a critical adjustment, but it's nice to have only the wanted part of the display showing. The ideal test pattern is a rectangular border, one dot thick, around the extremities of the display. Write a CHIP-8 program to do it.

(7) If you have already modified your Dream board (e.g. to take a RAM expansion board), then first put it back to its original form. Then rewire it as shown. Note that $\overline{WE}$ (Write Enable, to all RAM) is still available for use by external RAM. N.B: $\overline{CS}$ must now only be used to select the on-card 1K of RAM. $\overline{WE}$ should now be routed to pin 11 on the expansion bus socket (formerly $\overline{BA}$). $\overline{BA}$ is no longer required for any expansion of memory.

(8) Special note for use of J-R expansion board: (i) $\overline{CS}$ and $\overline{BA}$ signals must be disconnected; (ii) the 74LS08 pins 2,5&12 ($\overline{BA}$) are now tied high; (iii) the 74LS155 pin 2 wired to A15, and pin 14 wired to A14; (ensure pins 2 & 14 not shorted).

## TRACKS TO BE CUT and LINKS TO BE REMOVED (at specified pins):-

( ) IC8 (6875 skt) pin 5; use PCB track for 4 MHz.
( ) IC19 (74123) pins 1,2 & 3.
( ) IC6 (74LS367) pin 14.
( ) IC15 (7493) pin 14.
( ) Only if using 74LS165; IC7 (4014 skt) pin 9.
( ) IC10 (74LS10) pins 3 & 13.
( ) Remove links from IC10 pins 4 & 5.
( ) Remove links from IC 5 pins 4, 6, 10 & 12.
( ) Remove links from IC 6 pins 2, 4, 6, 10 & 12.

It is recommended to make all new connections on the underside of the PCB, with 30G 'Kynar' (wire-wrap) wire, for neat and reliable results. Take care not to nick the wire when stripping insulation.

Changes to the CHIPOS EPROM (Refer to your CHIPOS manual)

The subroutine 'DISLOC' computes the 16 bit address of the byte to be altered during a display operation. (In general, two such bytes adjacent to each other will be altered during execution of a SHOW instruction, because any arbitrary pattern byte will overlap the boundary of two adjacent video RAM locations. Or, put another way, the X coordinate of a symbol to be displayed will not, usually, correspond with a V-RAM byte boundary.) DISLOC builds up the 16 bit address 'BLOC' (Buffer LOCation) given the symbol coords 'VX' and 'VY' as input parameters; (VX is passed via acc-B). The resulting BLOC is moved to the X-reg for use by subr 'SHOWUP'.

The old 64x32 display format consisted of 256 bytes of video RAM, which meant that DISLOC only had to compute the low order byte of the buffer location, BLOC, while the high order byte remained constant at 01. The new 128x64 format requires 1K of RAM. Therefore, ten bits are required to specify a given RAM location, so DISLOC must now compute the high order byte as well.

```
C07B  40 00          CHIPOS PATCHES
C081  44 00          for 'Hi-res'
C3B6  43 90
C3E3  3A

C220  29 7F 00 3F  DE 26 C4 0F  26 02 C6 10  37 DF 14 A6
C230  00 97 1E 7F  00 1F D6 2E  C4 07 27 09  74 00 1E 76
C240  00 1F 5A 26  F5 D6 2E 8D  28 96 1E 8D  15 D6 2E CB
C250  08 8D 1E 96  1F 8D 0B 7C  00 2F DE 14  08 33 5A 26
C260  CB 39 16 E8  00 AA 00 E7  00 11 27 04  86 01 97 3F
C270  39 96 2F 84  3F 58 44 56  44 56 44 56  44 56 8A 40
C280  97 1C D7 1D  DE 1C 39
```

Relocated routines:-

| | |
|---|---|
| SHOW2 | C22C |
| SHOWUP | C262 |
| DISLOC | C271 |

TESTING

Just in case there is something wrong with the way you have your expansion RAM board wired, I recommend that you first test your 'hi-res' system without it. This is possible by temporarily locating the 1K Video RAM at $0000. Simply jumper the A14 line to IC10/pin5 (instead of A14; see circuit). Astute readers will realize that the system's scratch area and stack ($0000 - $0080) will actually be visible at the top of the display window, so you will be able to observe the bit patterns changing when you use 'memod'. Notice the 2 bytes toward the top left of the display counting in binary? These are the relative-time-clock (RTC) counters ($0020,21). Of course with the old CHIPOS EPROM, nothing sensible will be displayed, but you can still use memod 'blindly' to write stuff into the display buffer (now at $0080 - $0400); try it. If all goes well, jumper the VRAM back to $4000, plug in your expander board and use memod again to check writing data into the video buffer (at $4000). If that works, plug in a re-programmed EPROM, and your new system is ready for use. The 128 x 64 graphics format is quite respectable and should inspire lots of practical applications.