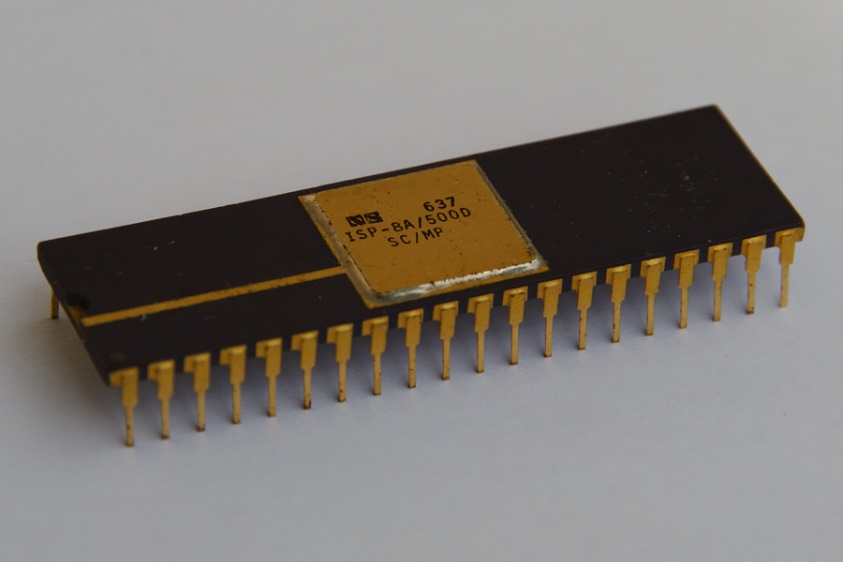
**Commit: Circuit Diagram, Datasheets and Languages**

Rather than photocopy the diagrams, I’ve translated them into KiCad format, and they should be in documentation/schematics. Any errors may be the translation, or may just be the original design…..

There are (were ?) actually two levels, of main processor board ; the “basic” level and the “advanced” level.

They are broadly speaking almost identical. The “basic board” was designed to support 128 bytes of RAM memory, and 1k of ROM memory as a minimum, up to 1k of RAM memory and 2k of ROM memory.

These probably sound really pathetic, but in the late 1970s this was a \*lot\* of memory. Writing 1,024 bytes worth of machine code is an awful lot in hexadecimal.

The basic circuit is in the file “basic-circuit.pdf” which should be on the github. If I haven’t forgotten.

The main change between the two, which I scribbled on later, is the “ZEROPAGE” line, which adds 64k addressing to the machine rather than 4k addressing – it allows you to decode the address properly (rather than just assuming 12 bits). An SC/MP only has 12 address lines, though there are 16 internally, and you have to latch the other 4.

The main CPU is an SC/MP which is running at about 1Mhz clock speed (hence the 560pf capacitor). Sense A and Sense B and Reset are connected to push buttons with pull up resistors.

There are a pair of 1k ROMS, IC2 and IC3 which are Motorola 6830 ROMs (the odd obsession with Motorola chips occurs throughout the design, probably because the original ETI articles all used Motorola stuff). IC2 is enabled for addresses 000-3FF and IC3 for 400-7FF (pin 13 uses A10 and not A10), and they both have connections to ZeroPage (which is pulled to ‘1’ at present). Additionally, NRDS has to be zero (e.g. reading, it’s inverse) to enable this chip. This is connected to the I/O, see later.

RAM is provided by the 128x8 SRAM 6810 chips, of which only one (IC4) is shown. This is decoded for A10/A11 to be at C00-FFF (e.g. A10 and A11 both logic 1) and also test ZeroPage. The decoding for lines A7,A8,A9 is done via IC5 (a 7442 - I should probably have used a 74138 !) binary to decimal decoder which drives the inverse CS5 line. This allows RAM in steps of 128 bytes, up to 1k in total.

The decoders for I/O are the main difference. These were originally 7400s, but they were updated to 7410s so I could test for zero page.

Input involves anding not-A10, A11 and zero page, e.g. maps on to 10xx xxxx or 800-BFF – the keyboard input. This is used to gate a signal onto the bus via IC5 which is what was then one of the brand new 74244 chips. Connected to this are 8 toggle switches, which I appear to have left floating … duh !

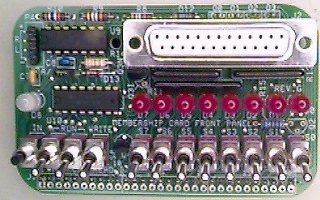
Output involves anding not A11, WRITE (e.g. Not NWDS) and ZERO page, thus latching the 7475s on write to the ROM area 000-7FF – not sure about the timing here tbh but I think it would work – these are originally connected to 8 LEDs (or 2 seven segment displays as an ‘expansion’)

The lines NWDS,NRDS,A0-A11,DB0-7,OUTPUT,NADS, the Flag registers and Serial In/Out are passed on to expansion.

In many ways, it’s an ordinary trainer board with LEDs and toggle switches. The oddity is having a 1k Monitor in there, many designs use direct control to fill memory.

Memory Map (mirrored 16 times through the 64k address space)

|  |  |  |
| --- | --- | --- |
| **Address** | **Access** | **Components** |
| 000-3FF | Read | 1k ROM (IC2) |
| 400-7FF | Read | 1k ROM (IC3) |
| 000-7FF | Write | Output to LED display |
| 800-BFF | Read | Read toggle switches. |
| C00-FFF | Read/Write | Up to 1k of RAM memory. (IC4 x 8) |



A modern equivalent – couldn’t find an SC/MP picture – this is Lee Hart’s 1802 “Membership Card”