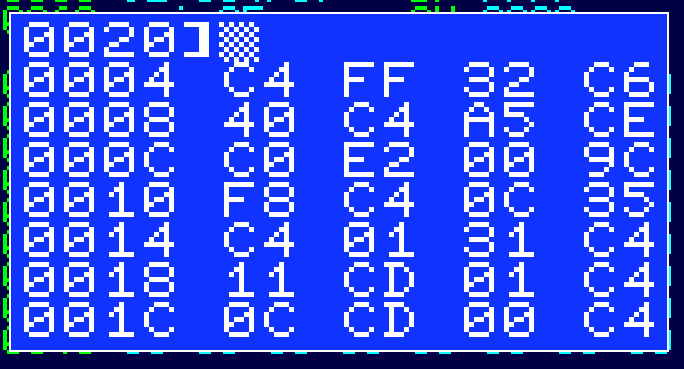
**Release : code for post 13**

I’m quite pleased with progress on this. I was worrying about fitting the assembler in, but the other bits of the monitor are written and working properly – well, I haven’t tested the tape read ☺ but it probably works.

The monitor commands are built around the concept of current address – on the screen you can see this is currently $0020. The cursor is at the top of the screen because it loops round from the bottom rather than scrolling, part of my (rather unnecessary) requirement that the monitor does not require extra RAM beyond the 128 bytes. It will probably end up using about half that 128 byte minimum in variables, input buffers, labels and the stack.

The current supported commands are:

A [aaaa] Set current address to aaaa

B [bb] [bb] [bb] …. Enter bytes – as many as will fit on the line.

C Clear screen.

D [aaaa] Dump from current address, updating current address if aaaa is present

G aaaa Run program from aaaa – the address is mandatory here ; XPPC P3 exits.

GET [aaaa] Read tape into memory at current address, updating if aaaa is present

PUT nnnn Write nnnn bytes starting at the current address.

So the next bits are – adding in the micro assembler so you can type code straight in, then the label allocation code.

Then I’ll see if I can fit a disassembler in as well. It looks quite promising. If there is still space left I might put some 16 bit multiply and divide routines in.