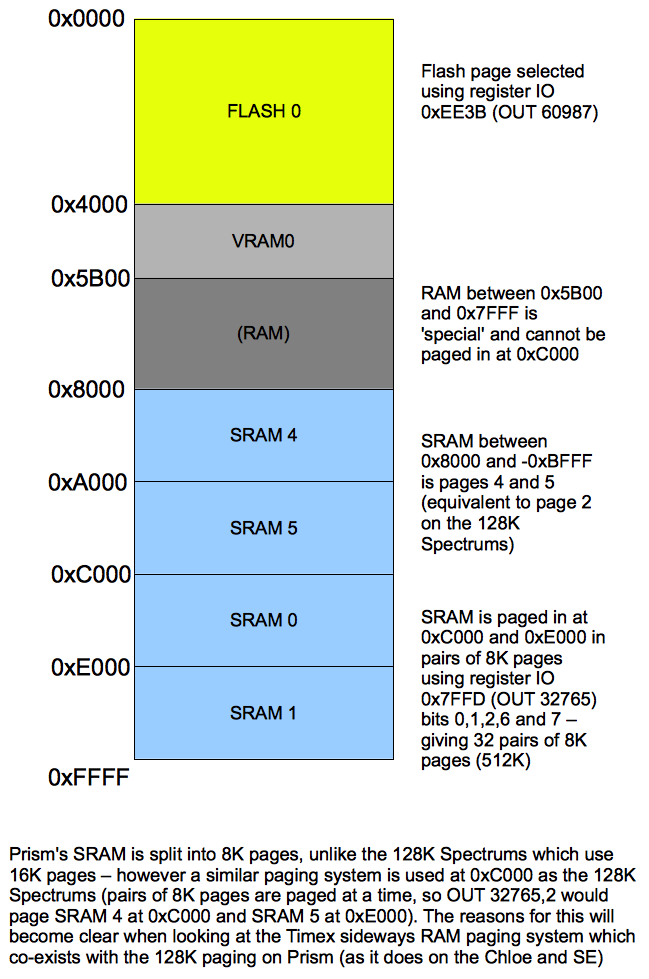
**Prism Memory - Memory Map**

Prism’s **default** memory map looks as follows:



**Compatibility Considerations**

There are two memory-related compatibility issues that Prism’s default configuration introduces, and both can be mitigated easily.

Firstly, Some 128K software doesn’t expect bits 6 and 7 of the register at 0x7FFD (332765) to affect memory paging (on the 128K machines, these bits do nothing at all) and therefore don’t care about whether they write to those bits or not. As these two bits are used as part of Prism’s memory paging, this can cause unexpected RAM pages to be presented at 0xC000 and 0xE000 and lead to a crash. Two games which do this are OutRun and Turbo OutRun. To fix this, Prism can be limited to 128K easily by setting the appropriate bit of ULA2’s MMU compatibility register (ULA2 4 bit register 1) :

OUT 36411,24 restricts memory to 128K.

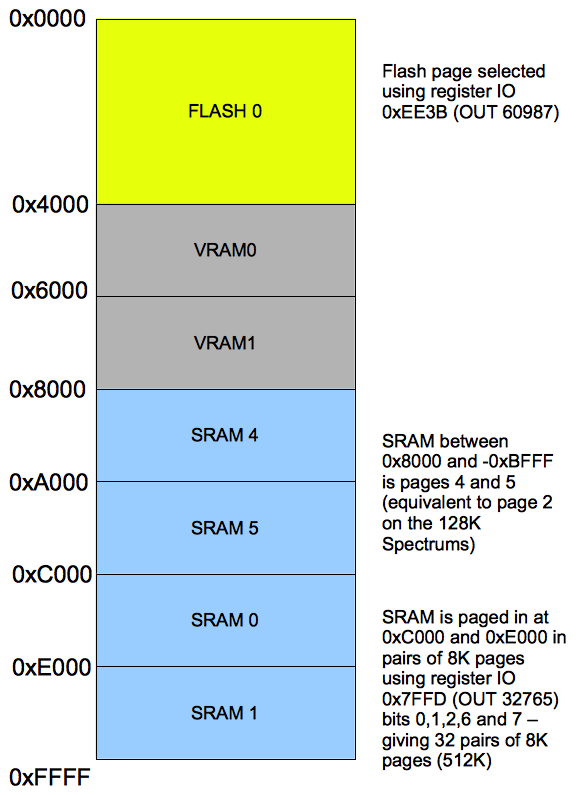
OUT 36411,16 removes the restriction.

Secondly, some Timex software (and some Chloe/Spectrum SE software) expects the Timex shadow screen (stored in VRAM1) to be presented to the CPU at memory address 0x6000. Prism doesn’t present VRAM past 0x5AFF by default, mainly so that the printer buffer and system variables don’t corrupt the screen in certain modes. This issue can also be rectified by setting a bit in ULA2’s MMU compatibility register (ULA2 4 bit register 1) to enable a 16K VRAM aperture . This cannot be done from BASIC (as all BASIC’s system variables would suddenly disappear!)

OUT 36411, 17 – set VRAM aperture to 16K

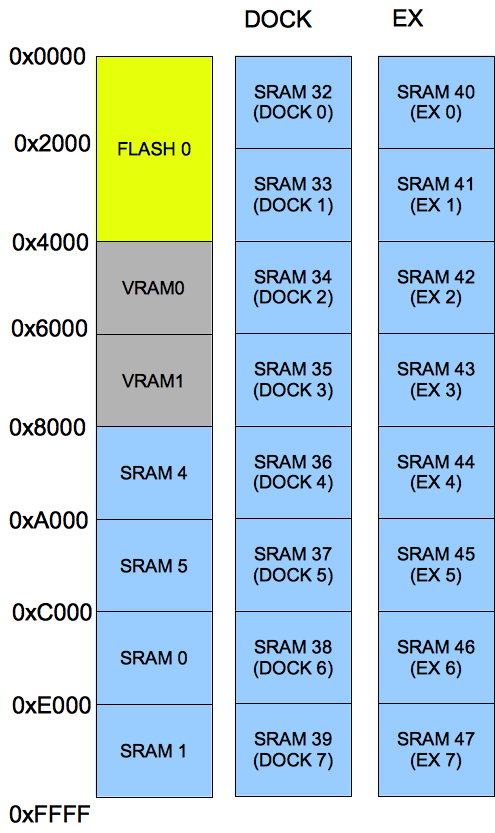
OUT 36411,16 – set VRAM aperture back to default (0x4000 – 0x5AFF)

The bits of the compatibility register are mutually exclusive, so if you wanted to both restrict memory to 128K AND have 16K of video RAM presented to the CPU between 0x4000 and 0x7FFF (like a real 128K Spectrum) then this can be done using OUT 36411,25.

****

**Memory map with (Timex/Chloe/SE compatible) 16K VRAM Apperture enabled.**

**Timex/Chloe/Spectrum SE Sideways RAM**

****