This document describes the Mutek virtual memory support.

CONFIG TOKEN:

Here is the list of new token add by virtual memory support:

CONFIG_HEXO_ARCH_MMU / CONFIG_HEXO_CPU_MMU

Type of MMU is used (processor or architecture)

CONFIG_HEXO_MMU

Enable MMU support (one of two previous token are require)

CONFIG_HEXO_MMU_PADDR

Physical address size in bits (Must be define if CONFIG_HEXO_MMU is enable)

CONFIG_HEXO_MMU_VADDR

Virtual address size in bits (Must be define if CONFIG_HEXO_MMU is enable)

CONFIG_HEXO_MMU_PAGESIZE

Virtual page size in bytes (Must be define if CONFIG_HEXO_MMU is enable)

CONFIG_VMEM_KERNEL_ALLOC

Enable Mutek virtual memory allocator.

CONFIG_VMEM_PHYS_ALLOC

Enable Mutek physical page allocator.

CONFIG_VMEM

Globaly enable: enable both MMU support and Mutek allocators.

CONFIG_SOCLIB_VMEM_MALLOC_REGION_SIZE

Preallocate memory size for malloc (see below).

Some Explanation

A memory region is initialy allocated for the memory allocator (malloc). The virtual memory is disabled when Mutek boot, but virtual memory allocators need to use malloc for their structures. So malloc must be available, with enought already mapped space.

Plateform modification

You must substitute the *vci_xcache_wrapper* by *vci_vcache_wrapper2* and the *vci_ram* by *vci_simple_ram* in the mutek tutorial plateform for virtual memory testing.