## Design Description: MMU

### Intro

The Memory Management Unit (MMU) is designed with lookup tables. It was initially thought to be designed with linked lists, but that was soon was decided against. For full detail of the design changes please see (insert number here) Design decision and changes to the MMU.

### Description

The MMU is designed with a stack size of 256 Bytes, with two lookup tables as previously stated. Each which keeps track of the user stack (32 blocks) and the kernel stack. Both lookup tables are arrays of chars, char was used because it was the smallest possible variable type (1 Byte). The chars would take either the value of 0x0 or 0x1, which means free, and in use respectively.

#### Memory size

The 32 blocks of user stack is resided from 0x10006000 to 0x10008000. The block 0x10006000 is not used in case of future implementation that may require an extra flag. The kernel stack would start from 0x10005FFC down to Image$$RW\_IRAM1$$ZI$$Limit.

#### Memory – Lookup Table Mapping

Since we are using lookup tables, there has to be a mapping between the lookup table and the actual physical memory blocks. The mapping is illustrated in the Table 1 MMU Lookup Table Mapping. Having two lookup tables allows the kernel processes to continue function when the OS is out of user memory.

Table MMU Lookup Table Mapping

