1 Getting Started

Your job in this activity is to set up the AARCH64 page table to identity map the first 2 mbyte page of memory (addresses 0x00000000-0x00200000).

To Do:

1. Download `textttmmusetup.S` from the course website into your kernel tree. Add it to the Makefile’s OBJS list.

2. Call `mmu_on()` from `kernel_main` (`mmu_on()` takes no arguments).

3. Test your code in qemu and make sure it works.

4. Translate `mmu_on()` to C.

Below is a C struct that implements the L1 descriptor format:

```c
struct table_descriptor_stage1 {
    unsigned int type :2; // Least significant bits
    unsigned int ignored :10;
    unsigned long next_lvl_table :36;
    unsigned int res0 :11;
    unsigned int pxn_table :1;
    unsigned int xn_table :1;
    unsigned int ap_table :2;
    unsigned int ns_table :1; // Most significant bit
};
```

Below is a C struct that implements the L2 descriptor format:

```c
struct page_descriptor_stage1 {
    // lower attributes (bits 16:2)
    unsigned int type : 2; // block (2'b01) or table (2'b11)
    unsigned int attrindx : 3; // stage 1 memory attr index field
    unsigned int ns : 1; // non-secure
    unsigned int ap : 2; // data access permissions
    unsigned int sh : 2; // sharability field
    unsigned int af : 1; // accessed flag
    unsigned int ng : 1; // not global
    unsigned int oa : 4; // ?
    unsigned int nt : 1; // block translation entry
    unsigned int output_addr : 18; // output address. size config'd by T0SZ field in tcr_el1
}
```

![Table Entry Structure](image)
unsigned int res01 : 13; // another res0 field
unsigned int res00 : 2; // reserved, 0

// upper attributes (bits 63:50)
unsigned int gp : 1; // guarded page
unsigned int dbm : 1; // dirty bit modifier
unsigned int contiguous : 1; // translation table entry is one of a
    // contiguous set of entries that might be
    // cached in a single TLB entry
unsigned int pxn : 1; // privileged execute never
unsigned int xn : 1; // execute never
unsigned int ignored2 : 4;
unsigned int pbha : 4; // page-based hardware attributes

unsigned int ignored1 : 1;
}__attribute__((packed));