

```

1 page_pde_offset = (__PAGE_OFFSET >> 20);
2
3     movl $(pg0 - __PAGE_OFFSET), %edi
4     movl $(swapper_pg_dir - __PAGE_OFFSET), %edx
5     movl $0x007, %eax           # 0x007 = PRESENT+RW+USER
6 10:
7     leal 0x007(%edi),%ecx       # Create PDE entry
8     movl %ecx, (%edx)          # Store identity PDE entry
9     movl %ecx, page_pde_offset(%edx) # Store kernel PDE entry
10    addl $4, %edx
11    movl $1024, %ecx
12 11:
13    stosl
14    addl $0x1000, %eax
15    loop 11b
16    # End condition: we must map up to and including INIT_MAP_BEYOND_END
17    # bytes beyond the end of our own page tables; the +0x007 is the attribute bits
18    leal (INIT_MAP_BEYOND_END+0x007)(%edi), %ebp
19    cmpl %ebp, %eax
20    jb 10b
21    movl %edi, (init_pg_tables_end - __PAGE_OFFSET)

```