ELF Header, Real Mode Segmentation, Paging

[Paging Slides Adapted from Anton Burtsev’s Slides on System Boot for 143A Fall 17]

9 November 2018
Aftab Hussain
University of California, Irvine
Reading ELF Header during boot

ELF structure in xv6 to read in ELF header: (code)
Where the ELF header of the kernel is read: (code)

ELF Header Contents:
OS 0 to 1: Chapter 5: The Anatomy of a Program.
Review of Segmentation during boot in Real Mode
Segmentation in Real Mode

Logical address (also known as a 32-bit segment:offset pair)

CPU

Segment selector

Offset

32 bits

16 bits

16 bits

xv6 refers to this x86 logical address as a virtual address
Segmentation in Real Mode

Segment translation hardware

32 bits

16 bits

Segment selector

16 bits

Logical address
(also known as a 32-bit segment:offset pair)

CPU

16 bits

Segment selector

16 bits

Offset
Segmentation in Real Mode

Segment translation hardware

- 16 bits
- offset
- segment selector

32 bits

- Logical address
  (also known as a 32-bit segment:offset pair)

16 bits

- Segment selector
  left shift by 4

20 bits

- Segment selector

16 bits

- Offset
Segmentation in Real Mode

- **CPU**
- **segment selector** (16 bits)
- **offset** (16 bits)
- **logical address** (also known as a 32-bit segment:offset pair)
- **Segment translation hardware**
  - **segment selector** (16 bits)
  - **offset** (16 bits)
  - **segment selector** (20 bits)
    - **left shift by 4**
    - **add**
  - **linear address** (32 bits)
Segmentation in Real Mode

- Logical address (also known as a 32-bit segment:offset pair)

- Directly corresponds to the physical address

- Segment translation hardware
  - Segment selector (16 bits)
  - Offset (16 bits)

- CPU

- Linear address (20 bits)
  - Left shift by 4
  - Add
Segmentation in Real Mode

- Logical address (also known as a 32-bit segment:offset pair)
- Segment translation hardware
  - Segment selector
  - Offset
  - Linear address
    - Left shift by 4
    - Add

if paging is enabled, this address would go through a further translation process within the paging hardware to generate a physical address.
Segmentation in Real Mode

CPU

logical address

Segment translation hardware

xv6 configures this hardware such that logical and linear addresses are always the same.
Segmentation in Real Mode

It follows (without paging) in xv6, logical address = linear address = physical address

xv6 configures this hardware such that logical and linear addresses are always the same.
Review of Address Translation using Paging
mov (%EBX), EAX  # mov value from the location pointed by EBX into EAX
EAX = 0
EBX = 20 983 809

20 983 809 = 00 0000 0101 00 0000 0011 0000 0000 0001

page number

Virtual Address Space (or Memory) of the Process

page number = 5123
or (0b1 0100 0000 0011)

Physical Memory
mov (%EBX), EAX  # mov value from the location pointed by EBX into EAX  
EAX = 0  
EBX = 20 983 809  

\[20 \, 983 \, 809 = \underline{00 \, 0000 \, 0101\,00 \, 0000 \, 0011\,0000 \, 0000 \, 0001}\]  

Virtual Address Space (or Memory) of the Process  
page number  

Virtual Address Space (or Memory) of the Process  
page number = 5123  
or (0b1 0100 0000 0011)  

Physical Memory
mov (%EBX), EAX  # mov value from the location pointed by EBX into EAX
EAX = 0
EBX = 20 983 809

Virtual Address Space (or Memory) of the Process

CR3 = 0

Physical Memory

32 bits (4 bytes)

Level 1 (Page Table Directory)

page number = 5123
or (0b1 0100 0000 0011)

1M (1,048,575)
```
mov (%EBX), EAX  # mov value from the location pointed by EBX into EAX
EAX = 0
EBX = 20 983 809
```

![Diagram of virtual and physical memory addressing](image)

- **Virtual Address Space (or Memory) of the Process**
- **Physical Memory**
  - **Level 1 (Page Table Directory)**
  - **Level 2 (Page Table)**

Page number: 20 983 809

```
00 0000 0100 0000 0011 0000 0000 0001
```

- **Page number: 5123**
- **CR3 = 0**

1M (1,048,575)
mov (%EBX), EAX  # mov value from the location pointed by EBX into EAX
EAX = 0
EBX = 20 983 809

20 983 809 = 00 0000 0101 00 0000 0001 0000 0000 0001

Virtual Address Space (or Memory) of the Process

CR3 = 0

Physical Memory

32 bits (4 bytes)

Level 1 (Page Table Directory)

Level 2 (Page Table)

Page

Page number = 5123 or (0b1 0100 0000 0011)
mov (%EBX), EAX  # mov value from the location pointed by EBX into EAX

EAX = 0
EBX = 20 983 809

20 983 809 = 00 0000 0101 00 0000 0011 00 0000 0001

Virtual Address Space (or Memory) of the Process

CR3 = 0

page number = 5123 or (0b1 0100 0000 0011)

Physical Memory

Level 1 (Page Table Directory) Level 2 (Page Table) Page

32 bits (4 bytes)