Virtual Memory Guide

For this guide, let the virtual address be represented by ***V.***

Basic Virtual Memory Math

Physical Address are p bits -> 2^p bytes of physical memory.
Virtual Addresses are v bits -> 2^v bytes of virtual memory.

1. **Dynamic Relocation**

MMU Registers:
Relocation Register R
Limit Register L

Translation:
If V >= L then EXCEPTION
Else
Physical Address = V + R

Example 1:
V = 0x1084
R = 0x1000 1000
L = 0x2000

Physical Address = 0x1084 + 0x1000 1000 = 0x1000 2084

Example 2:
V = 0x294B
R = 0x4010 0000
L = 0x2400

Physical Address = EXCEPTION -> V > L

2. **Segmentation**

Number of Segments: S
Number of Bits required for Segment Number = log(S)
Segment Offset Bits = v - log(S)

Virtual Address Format (Binary)= SSS...S OOO … O
Where S is the segment bits, O is the offset bits.

Example 1:
v = 16 bits
Number of Segments: 2

Number of Bits for Segment Number: log(2) = 1
Segment Offset Bits: 16 - log(2) = 16 - 1 = 15
Virtual Address Format: SOOO OOOO OOOO OOOO

Example 2:
v = 32 bits
Number of Segments: 4
Number of Bits for Segment Number: log(4) = 2
Segment Offset Bits: 32 - log(4) = 32 - 2 = 30
Virtual Address Format: SSOO OOOO OOOO OOOO OOOO OOOO OOOO OOOO

MMU Registers:
Relocation Register for Each Segment i: Ri
Limit Register for Each Segment i: Li

Translation:
If Segment Offset >= Li then EXCEPTION
Else
Physical Address = Segment Offset + Ri

Example 3:
v = 16 bits
R0 = 0x1000
R1 = 0x2000
L0 = 0x500
L1 = 0x800
V = 0x01AF
Segment: 0
Offset: 0x01AF
Physical Address: 0x01AF + R0 = 0x01AF + 0x1000 = 0x11AF

(Recall v is the number of bits for the virtual address, p, the number of bits for the physical address)

3. Single Level Paging

Frame Size: F
Page Size: P
P should equal F

Number of Frames = 2^p / F
Number of Pages = 2^v / P
Number of Page Table Entries (PTEs) = Number of Pages = 2^v / P

Example 1:
Page Size: 4KB (2^12 bytes)

Frame Size: 4KB
v: 24 bits
p: 32 bits
Number of Frames: 2^32 / 2^12 = 2^20
Number of Pages: 2^24 / 2^12 = 2^12
Number of Page Table Entries: 2^12

Size of PTE: S
Size of Page Table = Number of Page Table Entries * S = 2^v / P * S

Example 2:
Number of Page Table Entries: 2^12
Size of PTE: 4 bytes
Size of Page Table: 2^12 * 4 bytes = 2^2 * 2^12 = 2^14 bytes

Number of Bits for Page Number: log(Number of Pages)
Number of Bits for Frame Number: log(Number of Frames)
Number of Bits for Page Offset: log(Page Size)
Number of Bits for Frame Offset: log(Frame Size)
Virtual Address Format (binary): PP...POO...O

Example 3:
Page Size: 4KB (2^12 bytes)
Frame Size: 4KB
v: 24 bits
p: 32 bits
Number of Pages: 2^12
Number of Frames: 2^20
Number of Bits for Page Number: log(2^12) = 12
Number of Bits for Frame Number: log(2^20) = 20
Number of Bits for Page Offset: log(2^12) = 12
Number of Bits for Frame Offset: log(2^12) = 12
Virtual Address Format (binary): PPPP PPPP PPPP OOOO OOOO OOOO

Translation:
1. Extract Page Number
    a. Manually extract upper PageNumberBits, or,
    b. V / Page Size
2. Extract Page Offset
    a. Manually extract lower PageOffsetBits, or,
    b. V modulo Page Size
3. Lookup Frame Number using Page Number
    a. If NOT VALID, EXCEPTION
    b. Otherwise,
    Physical Address = Frame Number * Frame Size + Page Offset

Example 4:
Page Size: 4KB (2^12 bytes)
Frame Size: 4KB
v: 24 bits
p: 32 bits
Number of Pages: 2^12
Number of Frames: 2^20
Number of Bits for Page Number: log(2^12) = 12
Number of Bits for Frame Number: log(2^20) = 20
Number of Bits for Page Offset: log(2^12) = 12
Number of Bits for Frame Offset: log(2^12) = 12
V = 0x120F1A

Page Number: 0x120
Page Offset: 0xF1A

Lookup in page table yields Frame Number = 0x33000, Valid = 1

Physical Address: 0x33000 * 0x1000 + 0xF1A = 0x3300 0F1A

(Recall v is the number of bits for the virtual address, p, the number of bits for the physical address)

4. Multi-Level Paging

Please refer to Single-Level Paging for steps regarding number of frames, pages, etc.

PTE Size: S bytes
Page Size: P bytes

Number of PTEs per Page = P / S
Bits for Page Number = log(Number of PTEs per Page) = log(P / S)

Example 1:
PTE Size: 8 bytes
Page Size: 4KB (2^12)
Number of PTEs per Page: 2^12 / 8 = 2^12 / 2^3 = 2^9

Number of Levels = CEIL((v - PageOffsetBits) / PageNumberBits)
Virtual Address Format (binary): A..AB...BC...C …. OOO...OO
Where letters A-Z except O are page numbers for each level of the table, and O is the page offset bits

Example 2:
v: 32 bits
Page Size: 4KB (2^12 bytes)

Page Offset Bits: 12
PTE Size: 8 bytes
Number of PTEs per Page: 2^9
Number of Levels: CEIL((32 - 12) / 9) = CEIL(20 / 9) = 3
Virtual Address Format: 1122 2222 2223 3333 3333 OOOO OOOO OOOO

Physical Address = Frame Number * Frame Size + Page Offset

(Recall v is the number of bits for the virtual address, p, the number of bits for the physical address)