Exercise sheet 9 – Memory

Goals:

- MMU
- Virtual memory

Exercise 9.1: Intel x86/32 bit 1 level page table

Consider an Intel x86/32 bit architecture with a 1 level page table, similar to the 1 level page table in the lecture.
Given are:

- 32 bit architecture
- 4 KiB page/frame size
- 1 level page table
- Virtual address: 0x1202F494
- Real address: 0x00014494

The virtual address is mapped to the real address.

(a) State the *offset* part of the given addresses.

Proposal for solution: Offset: 0x494
32 bits for the whole address, 20 bits for page/frame and 12 bits for the offset.

(b) State the *page number* and the page base address.

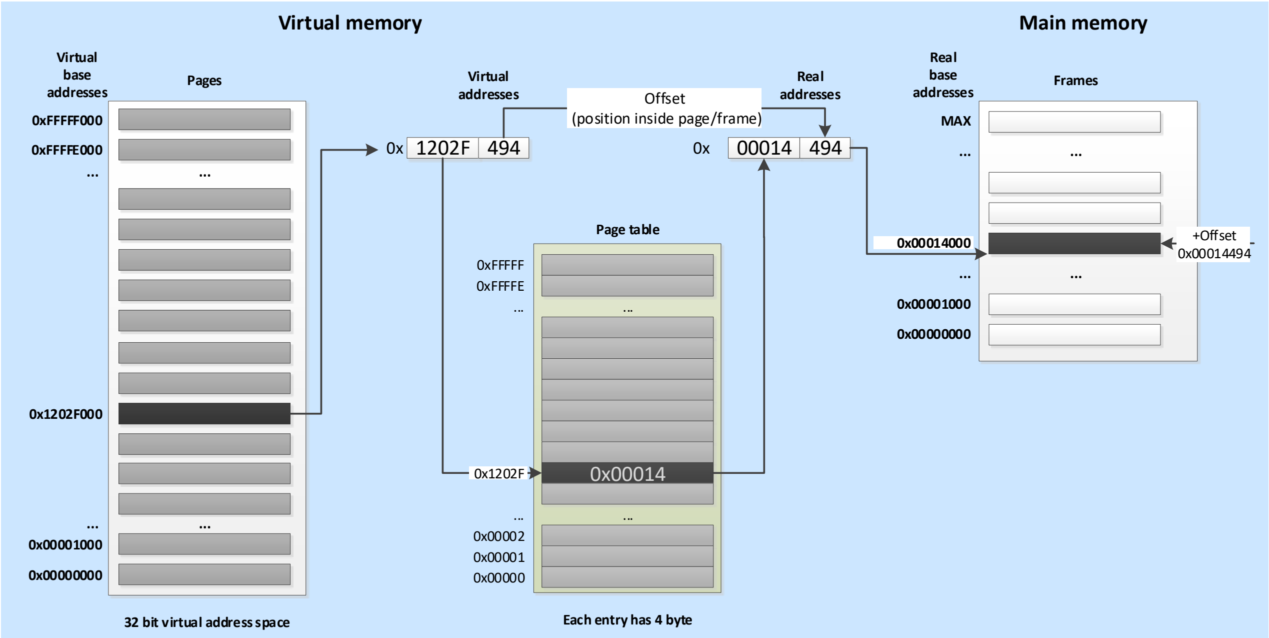
Proposal for solution: *Page: virtual memory*
Page number: 0x1202F
Page base address: 0x1202F000

(c) State the *frame number* and the frame base address.

Proposal for solution: *Frame: real memory*
Frame number: 0x00014
Frame base address: 0x00014000

(d) State the entry and its position in the page table for the given situation. *Hint: You may want to draw a sketch, similar to the lecture. Specify as much details as possible with explicit numbers.*

Proposal for solution:



(e) How many pages are possible for the given 32 bit architecture?

Proposal for solution: $2^{20} = 1048576$ pages

(f) Estimate the size in bytes for the page table. *Hint: You may simplify the address calculation by only considering to use a full word (address word) for each entry.*

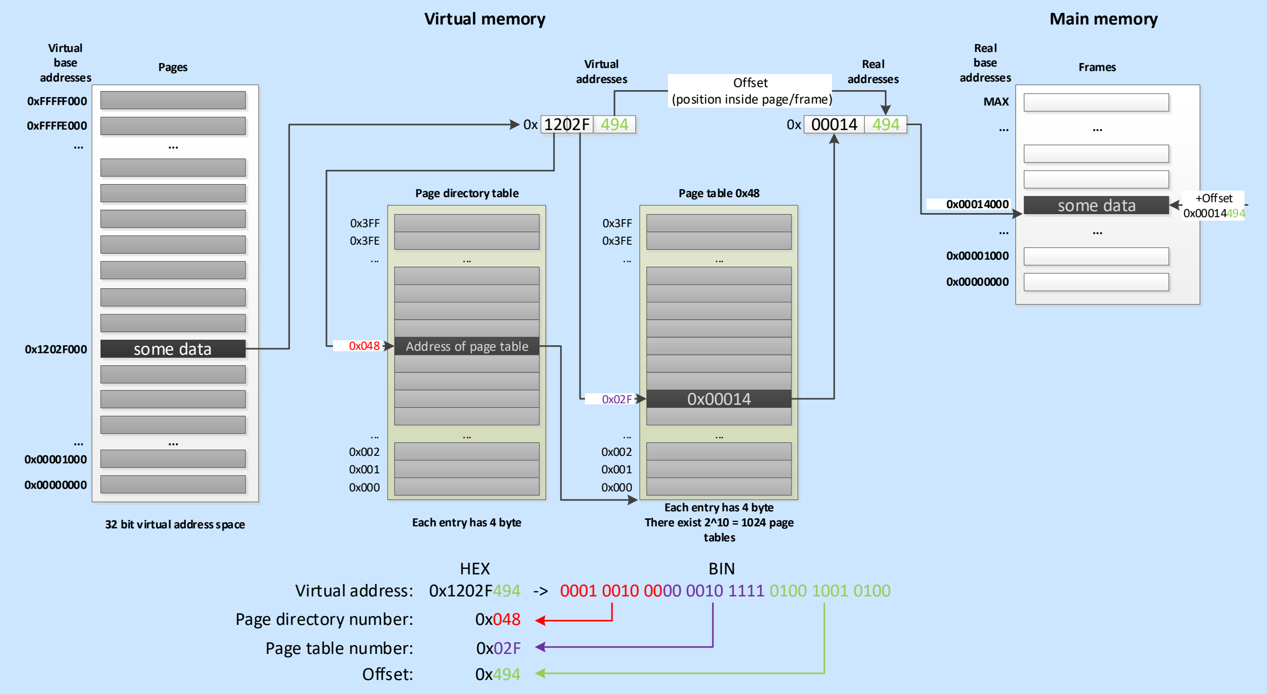
Proposal for solution: Each entry needs 4 byte, so the whole page table consumes about $2^{20} * 4$ bytes = 4194304 bytes (4 MiB)

Exercise 9.2: Intel x86/32 bit 2 level page table

Now consider an Intel x86/32 bit architecture with a 2 level page table (without segmentation). Use the same addresses as given in exercise 9.1.

(a) Draw a sketch with the situation. Use as much details as possible with explicit numbers (*you may calculate them and make certain assumptions*).

Proposal for solution:



Exercise 9.3: Intel x86/64 bit architecture with 3 level page table

Hint: Use the »AMD64 Architecture Programmer's Manual Volume 2: System Programming« to answer that question.

Given is:

- Virtual address: 0x0000 FF00 1232 F494
- Real address: 0x0000 0078 0012 F494
- Long-Mode Page Translation
- Page size: 2 MiB
- Bytes used for sign extension: 2
- Maximum bits for real memory addresses: 52 bits
- 3 level page table: *Page-Map, Page-Directory-Pointer, Page-Directory*

(a) Have a look on `RA_exercises/sheet_09/AMD64 Architecture Programmer's Manual.pdf` file page 134 (PDF: 186).

(b) How many bits are used for the offset? Show the offset in the given addresses.

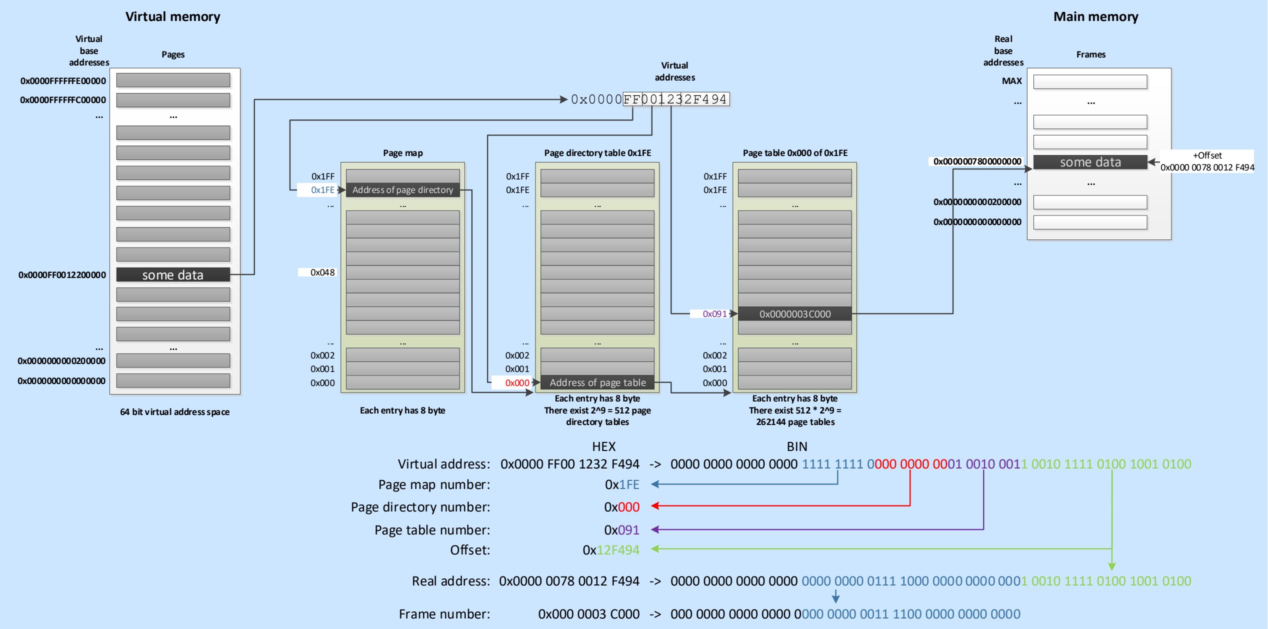
Proposal for solution: The right 21 bits are used for the offset: 0x12F494

(c) How many bits are used for each page table level?

Proposal for solution: For each page table level 9 bits are used. Bits 21 - 29 for Page-Directory Offset, bits 30 - 38 for Page-Directory-Pointer Offset, and bits 39 - 47 for Page Map.

(d) Draw a scheme for the page table situation. Use as much details as possible with explicit numbers (you may calculate them and make certain assumptions).

Proposal for solution:



	HEX		BIN
Virtual address:	0x0000 FF00 1232 F494	->	0000 0000 0000 0000 1111 1111 0000 0000 0001 0010 0011 0010 1111 0100 1001 0100
Page map number:	0x1FE		
Page directory number:	0x000		
Page table number:	0x091		
Offset:	0x12F494		
Real address:	0x0000 0078 0012 F494	->	0000 0000 0000 0000 0000 0000 0111 1000 0000 0000 0001 0010 1111 0100 1001 0100
Frame number:	0x000 0003 C000	->	000 0000 0000 0000 0000 0000 0011 1100 0000 0000 0000

(e) Calculate the maximum addressable real memory.

Proposal for solution: $2^{52} = 4,503599627 \times 10^{15}$ Bytes $=$ 4096 TiB $=$ 4 PiB