Writing kernels for fun ~~and profit~~

Γιάννης Τσιομπίκας

nuclear@member.fsf.org

23 Μαρτίου 2011

Γιατί;

- It's FUN!
- Εξοικείωση με το hardware.
- Εμβάθυνση στον θαυμαστό κόσμο των λειτουργικών συστημάτων.
- Μια καλή δικαιολογία να γράψουμε assembly (FUN!!!).
- Post-apocalyptic computing syndrom...

Πώς;

Πλατφόρμα

IBM PC-συμβατοί με 32bit intel CPU (80386 και άνω).

Εργαλεία

- GNU C compiler
- GNU binutils (assembler, linker, etc)
- GNU make
- GNU debugger
- GNU GRUB (legacy)
- QEMU

Booting

Η εκτέλεση ξεκινάει από την ROM (BIOS) σε 16 bit real mode.

BIOS

- Φορτώνει το πρώτο sector (512 bytes) στην διεύθυνση 7c00.
- jump στην διεύθυνση 7c00.

Boot Loader

- Φορτώνει τον kernel στην μνήμη.
- Ενεργοποιεί το A20 line.
- Βάζει τον επεξεργαστή σε protected mode.
- jump στο entry point του kernel.

Booting

Η εκτέλεση ξεκινάει από την ROM (BIOS) σε 16 bit real mode.

BIOS

- Φορτώνει το πρώτο sector (512 bytes) στην διεύθυνση 7c00.
- jump στην διεύθυνση 7c00.

Boot Loader

- Φορτώνει τον kernel στην μνήμη.
- Ενεργοποιεί το A20 line.
- Βάζει τον επεξεργαστή σε protected mode.
- jump στο entry point του kernel.

Booting

Η εκτέλεση ξεκινάει από την ROM (BIOS) σε 16 bit real mode.

BIOS

- Φορτώνει το πρώτο sector (512 bytes) στην διεύθυνση 7c00.
- jump στην διεύθυνση 7c00.

Boot Loader

- Φορτώνει τον kernel στην μνήμη.
- Ενεργοποιεί το A20 line.
- Βάζει τον επεξεργαστή σε protected mode.
- jump στο entry point του kernel.

Multiboot

Multiboot header:

offset	μέγεθος	πεδίο
0	4	magic identifier
4	4	flags
8	4	checksum
12	4	header address
16	4	load address
20	4	load end address
24	4	bss end address
28	4	entry address
32	4	video mode type
36	4	video mode width
40	4	video mode height
44	4	video mode color depth

Multiboot code

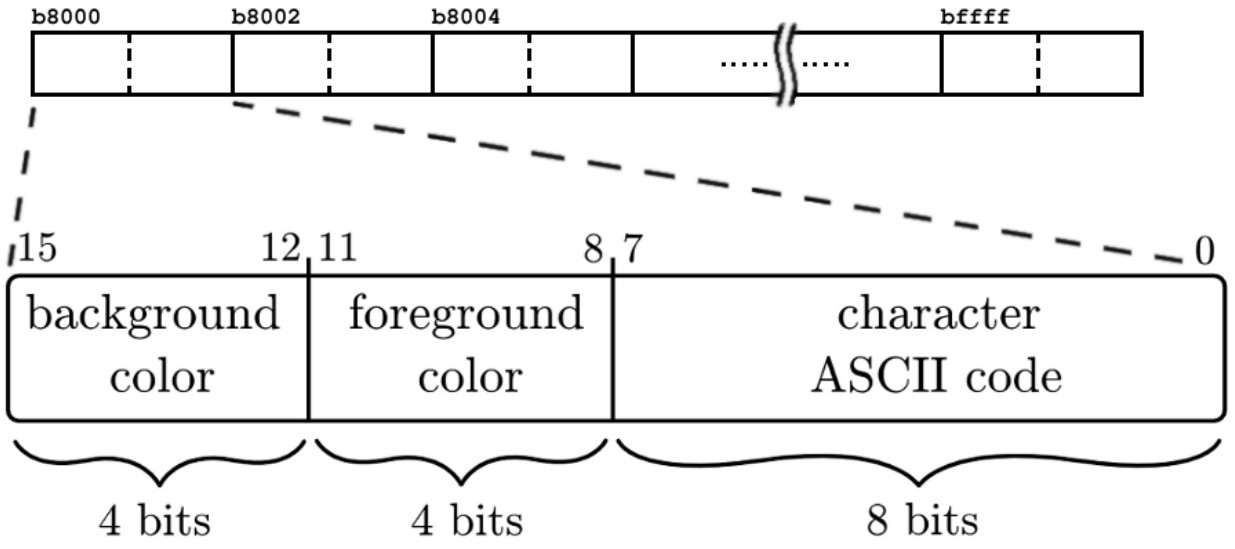
```
#define MAGIC       0x1badb002
#define FLAGS       0

    .align 4
    /* multiboot header */
    .long MAGIC
    .long FLAGS
    .long -(MAGIC + FLAGS)     /* checksum */
```

Entry code

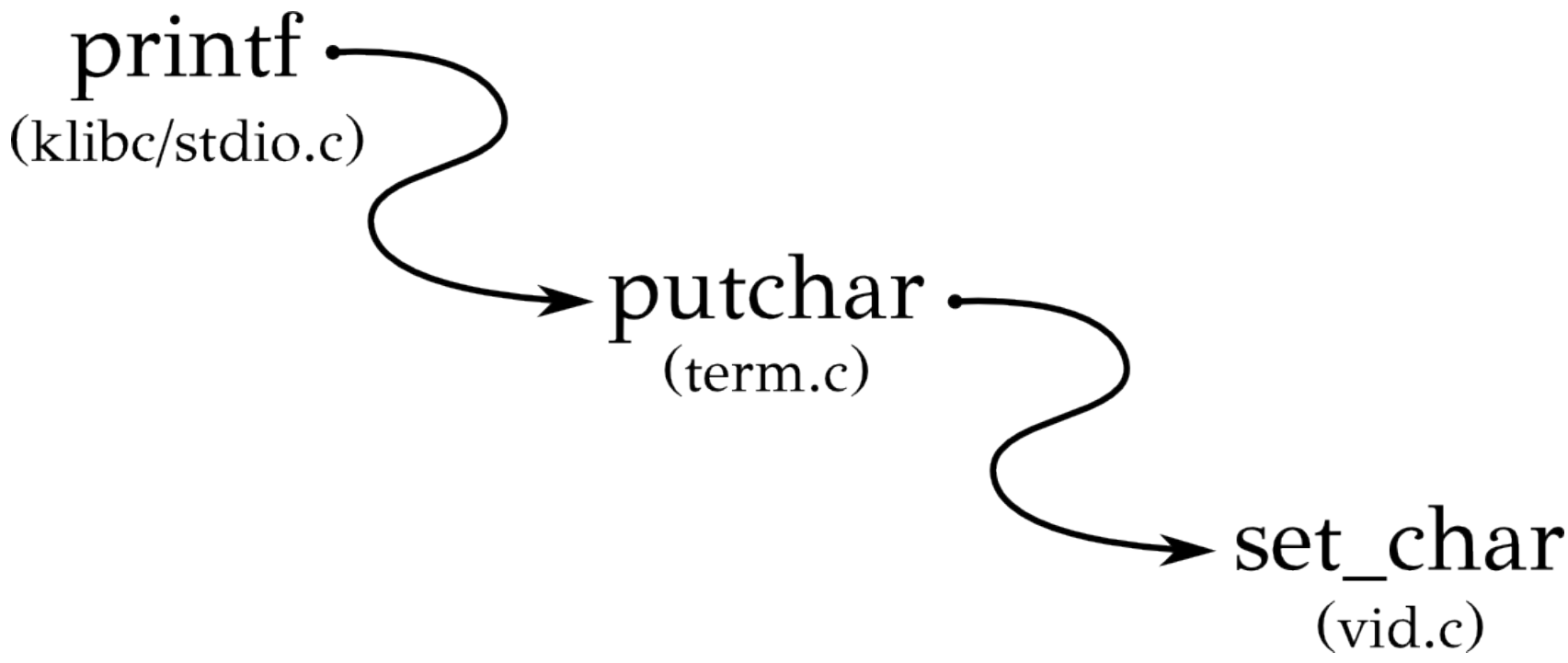
```
#define STACK_SIZE 0x4000
    .text
    .globl kentry
kentry:
    /* setup a temporary kernel stack */
    movl $(stack + STACK_SIZE), %esp
    /* reset eflags */
    pushl $0
    popf
    /* call the kernel main function */
    call kmain
    /* dropped out of main, halt the CPU */
    cli
    hlt
    /* space for the temporary kernel stack */
    .comm stack, STACK_SIZE
```

Text output

VGA text mode video memory: b8000

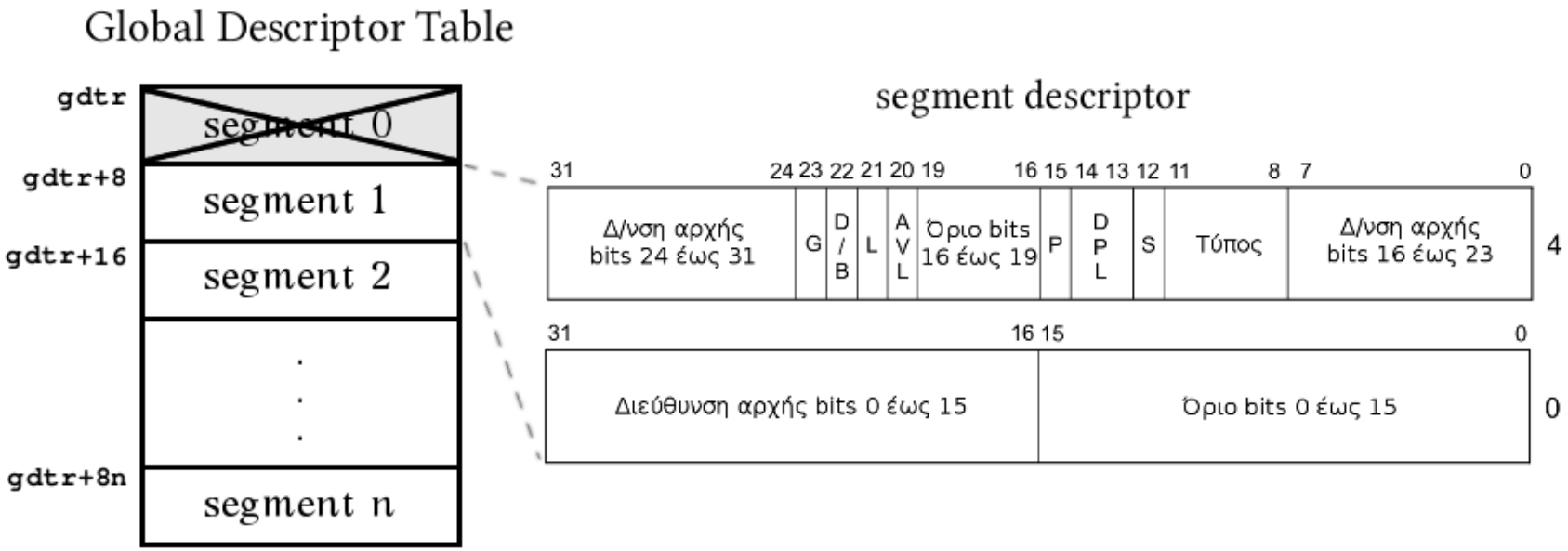


Text output - call graph

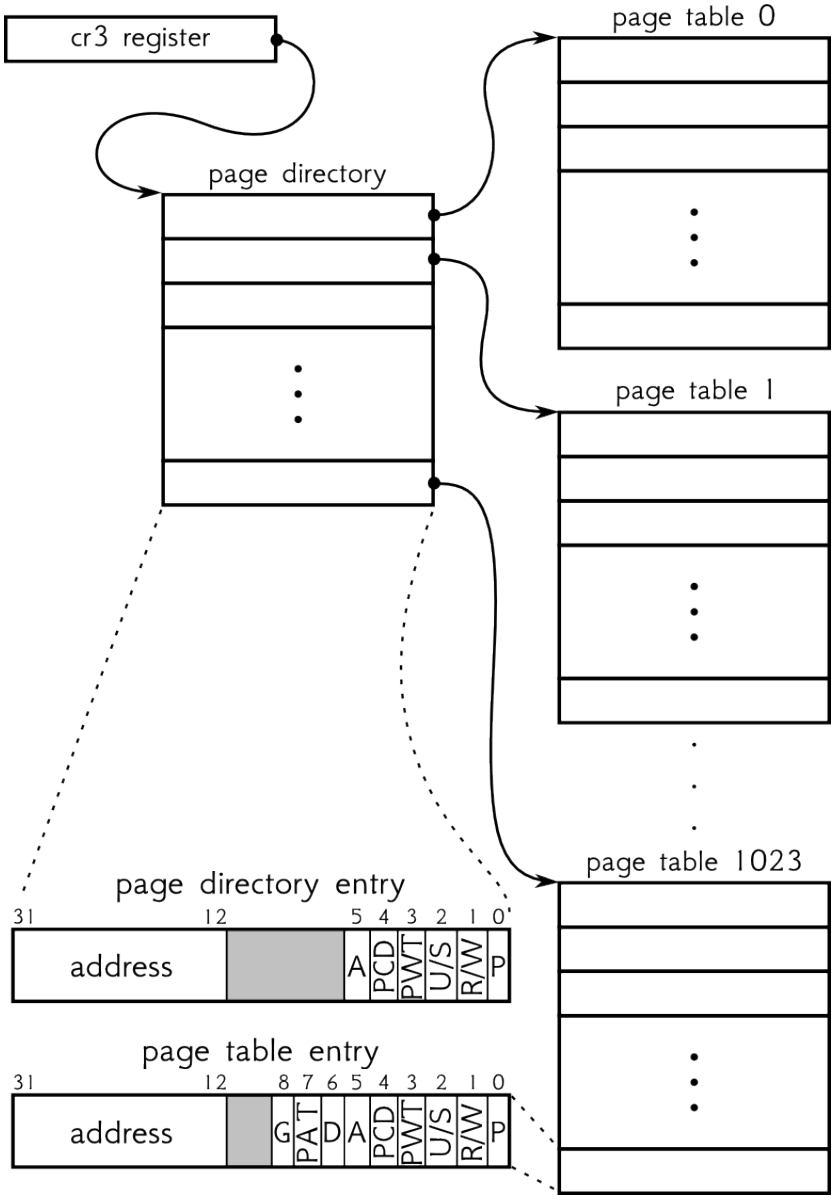


x86 memory model

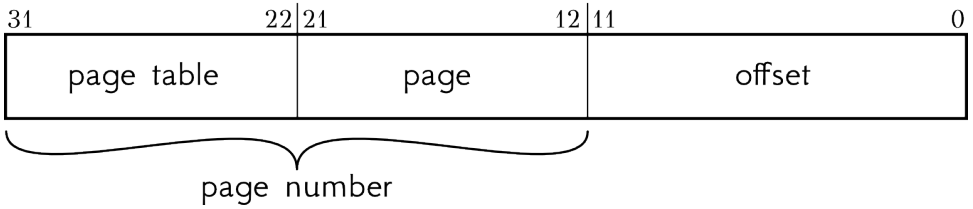
- Segments ορίζονται από 8-byte descriptors στον GDT (ή LDT).
- Λογικές διευθύνσεις αποτελούμενες από segment και offset. Segment selector registers επιλέγουν ποιο segment χρησιμοποιείται σε κάθε περίπτωση.
- Προαιρετικό paging με page tables 2 επιπέδων.



Paging



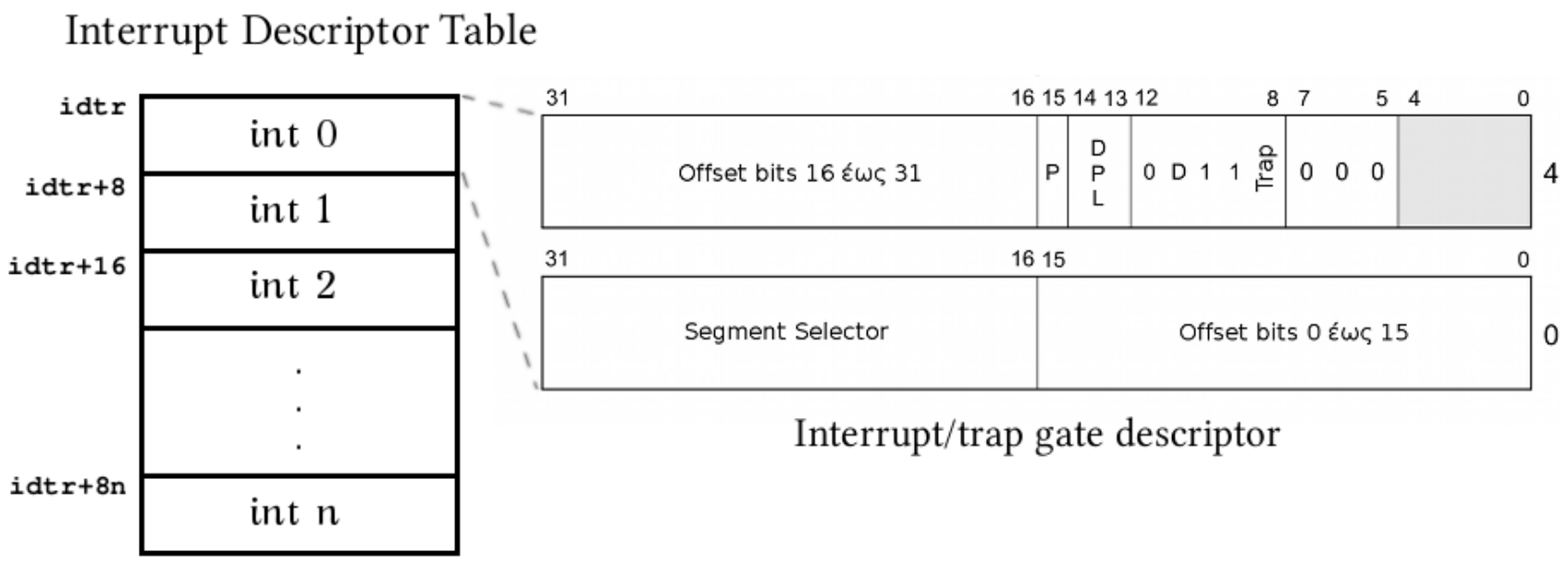
Virtual address translation



- Τα 10 ανώτερα bits $[22, 31]$ είναι index στο page directory (διαλέγουν page table).
- Τα επόμενα 10 bits $[12, 21]$ είναι index στο επιλεγμένο page table (διαλέγουν page).
- Τα κατώτερα 12 bits είναι το offset μέσα στο page.

Τύποι interrupts

- Hardware interrupts
- Software interrupts
- Exceptions



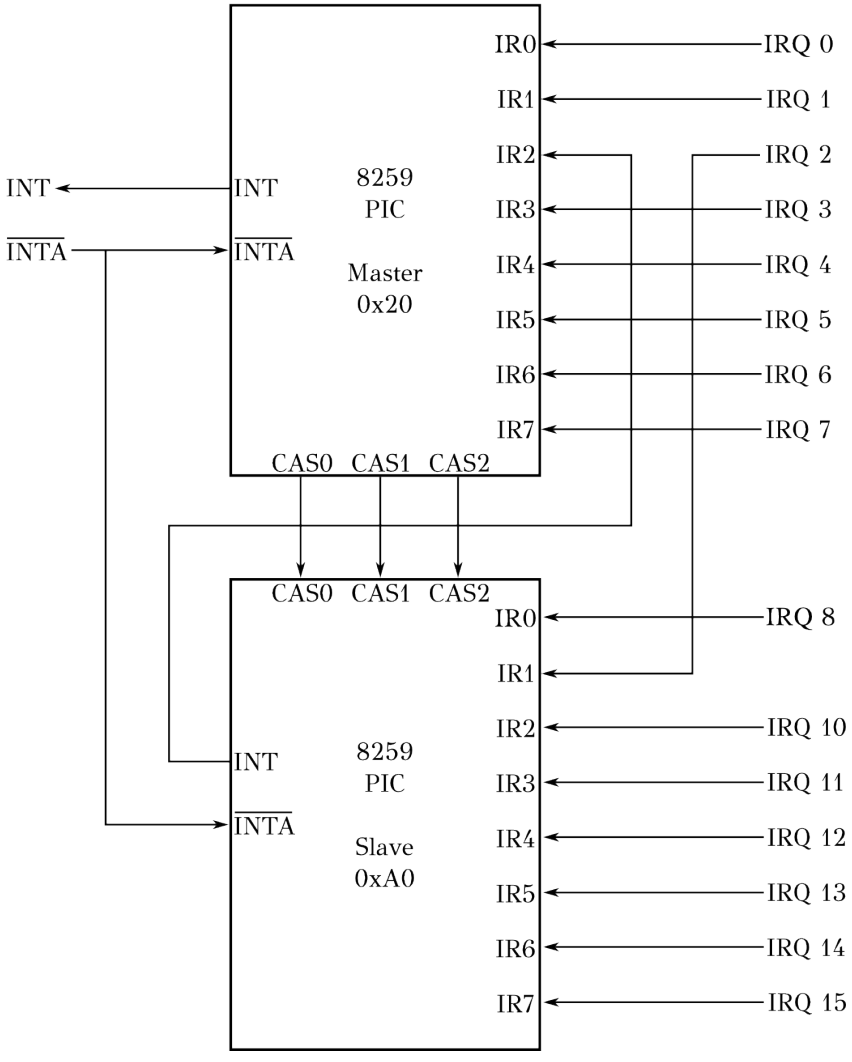
Exceptions

#	name	type	error code
0	divide error	fault	no
1	debug	trap/fault	no
2	NMI	N/A	N/A
3	breakpoint	trap	no
4	overflow	trap	no
5	bound range exceeded	fault	no
6	invalid opcode	fault	no
7	device not available	fault	no
8	double fault	abort	0
9	co-proc segment overrun	abort	no
10	invalid TSS	fault	selector
11	segment not present	fault	selector
12	stack fault	fault	selector or 0
13	general protection	fault	selector or 0
14	page fault	fault	special flags
15	reserved	N/A	N/A
16	floating point	fault	no
17	alignment check	fault	EXT bit
18	machine check	abort	no
19	SIMD floating point	fault	no

Τα interrupts 0 έως 31 είναι reserved για CPU exceptions.

Hardware interrupts

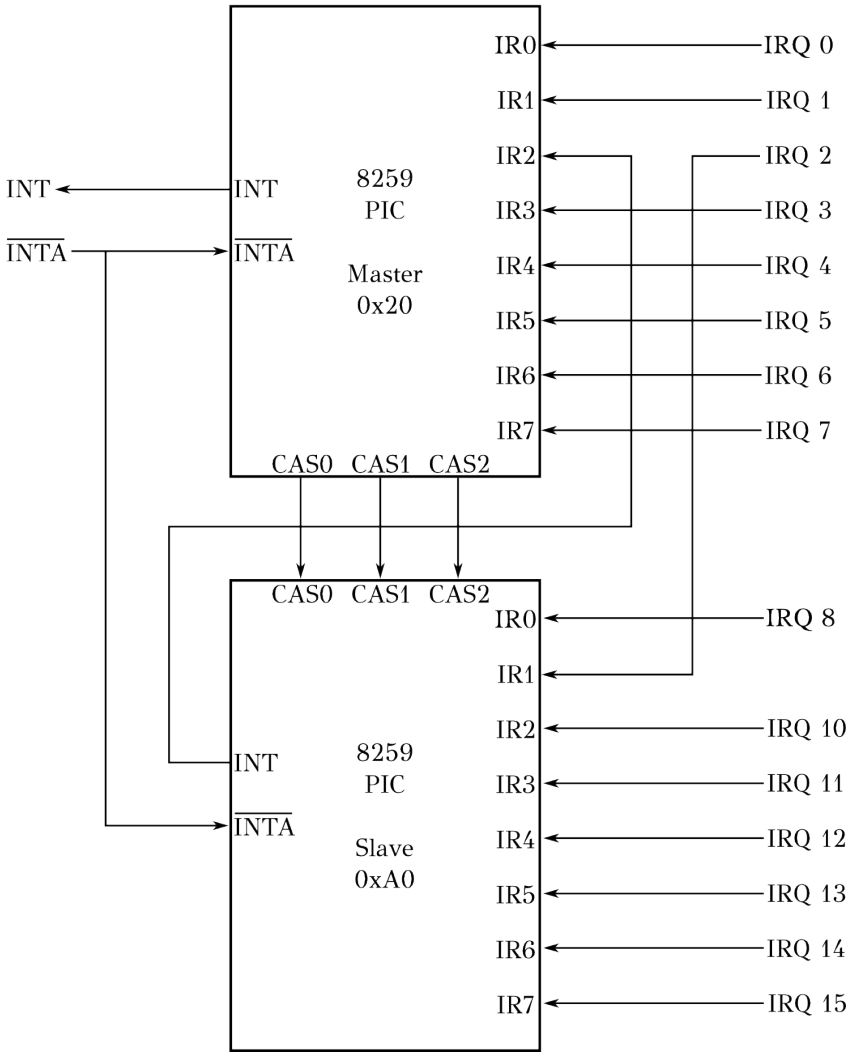
2 cascaded intel 8259A PIC chips




```
static void init_pic(int offset)
{
    /* send ICW1 saying we'll follow with ICW4
       later on */
    outb(ICW1_INIT | ICW1_ICW4_NEEDED, PIC1_CMD);
    outb(ICW1_INIT | ICW1_ICW4_NEEDED, PIC2_CMD);

    /* send ICW2 with IRQ remapping */
    outb(offset, PIC1_DATA);
    outb(offset + 8, PIC2_DATA);

    /* send ICW3 to setup the master/slave rel */
    /* ... set bit3 = 3rd int pin cascaded */
    outb(4, PIC1_DATA);
    /* ... set slave ID to 2 */
    outb(2, PIC2_DATA);

    /* send ICW4 to set 8086 mode (no calls) */
    outb(ICW4_8086, PIC1_DATA);
    outb(ICW4_8086, PIC2_DATA);

    /* done, reset the data port to 0 */
    outb(0, PIC1_DATA);
    outb(0, PIC2_DATA);
}
```

Hardware interrupts

2 cascaded intel 8259A PIC chips




```
static void init_pic(int offset)
{
    /* send ICW1 saying we'll follow with ICW4
       later on */
    outb(ICW1_INIT | ICW1_ICW4_NEEDED, PIC1_CMD);
    outb(ICW1_INIT | ICW1_ICW4_NEEDED, PIC2_CMD);

    /* send ICW2 with IRQ remapping */
    outb(offset, PIC1_DATA);
    outb(offset + 8, PIC2_DATA);

    /* send ICW3 to setup the master/slave rel */
    /* ... set bit3 = 3rd int pin cascaded */
    outb(4, PIC1_DATA);
    /* ... set slave ID to 2 */
    outb(2, PIC2_DATA);

    /* send ICW4 to set 8086 mode (no calls) */
    outb(ICW4_8086, PIC1_DATA);
    outb(ICW4_8086, PIC2_DATA);

    /* done, reset the data port to 0 */
    outb(0, PIC1_DATA);
    outb(0, PIC2_DATA);
}
```

Επόμενα βήματα;

- Διαχείριση μνήμης (kernel malloc).
- Processes.
- ATA driver & filesystem.
- Διάφορα: tty, timers, rtc.
- ...

Ερωτήσεις;

Links

- `http://nuclear.sdf-eu.org/articles/kerneldev`
- `http://codelab.wordpress.com`
- `http://www.linuxinside.gr`
- IRC: `#osdev` στο GRnet.