Lecture 33

processes v2 (w xv6)

1. the 'int' instruction (user to kernel)
2. sched / context swing & scheduler.
3. the new process procedure
4. the first process procedure
5. the scheduler (policy)

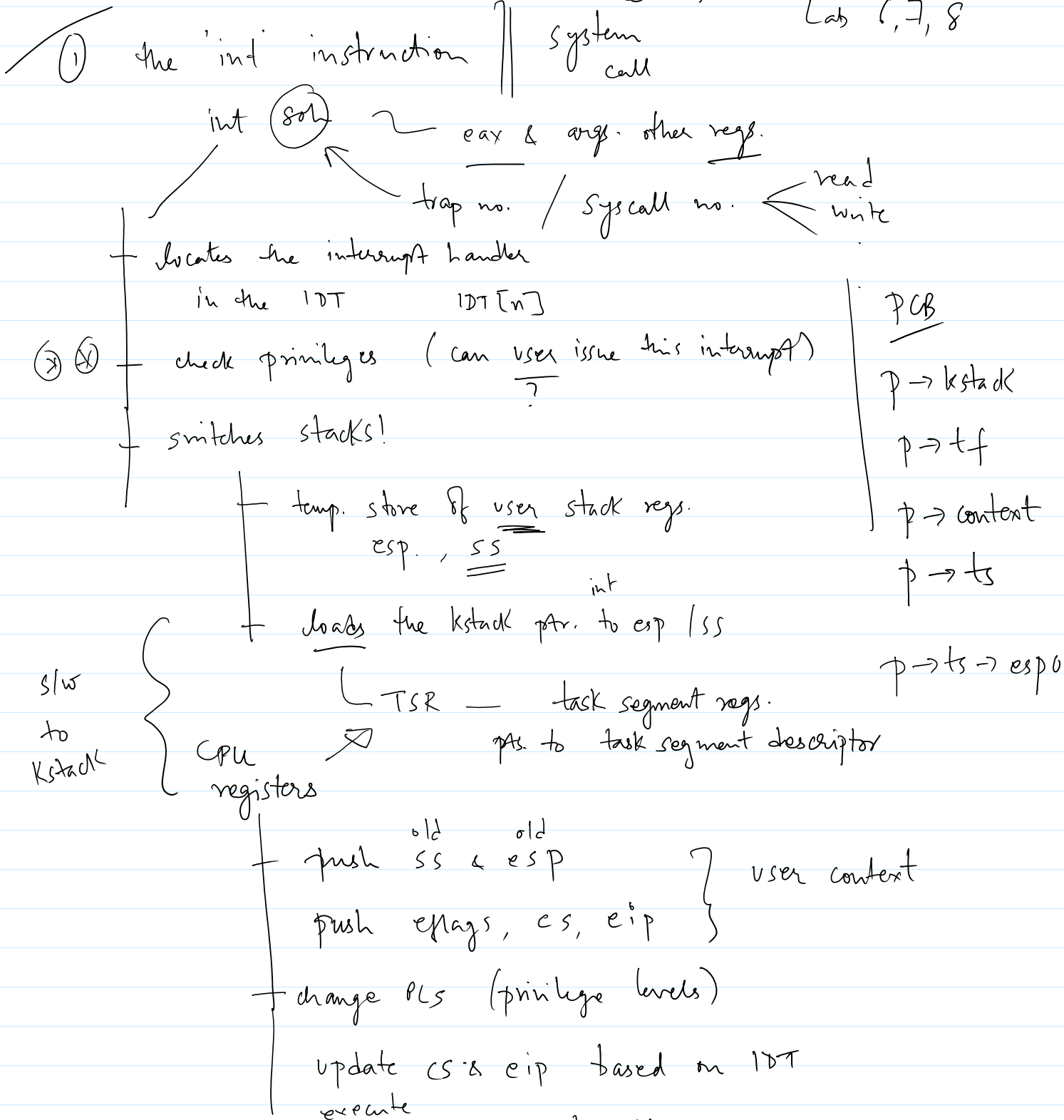Mon. 15th Oct.

Lab Quiz 3

at 1.55 pm

covers.

Lab 6, 7, 8

---

1. the 'int' instruction || invoke system call

int (80h) ~~~ eax & args. other regs.

↖ trap no. / syscall no. ← read write

- locates the interrupt handler in the IDT   IDT[n]

(?) (X) - check privileges (can <u>user</u> issue this interrupt?)

- switches stacks!

    - temp. store of user stack regs.
        esp. , ss

    - loads the kstack ptr. to esp / ss
                    int

        └ TSR — task segment regs.
                 ptr. to task segment descriptor

s/w to kstack {

CPU registers {

    - push old ss & old esp   } user context
    - push eflags, cs, eip   }

    - change PLs (privilege levels)

    - update cs & eip based on IDT
      execute

PCB
___

p → kstack

p → tf

p → context

p → ts

p → ts → esp0

— jmp to interrupt handler.

push error code

push trap no.

jmp all traps.

ts → esp0 ⟹   ← P → Kstack + KSTACKSIZE

SS
esp
eflags
cs
eip

h/w process
via int.

esp after int →

| error |
| trap no |

} start of trap handler

esp after
pre code →

| eax |
| ebx |
| ⋮ |

} all traps
push
context

esp after
all traps
———————
before trap.

| esp |

← esp

push esp
Call trap

Context of
user
process