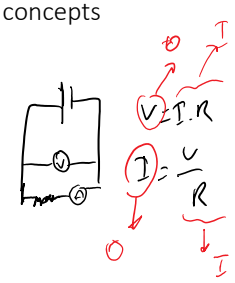
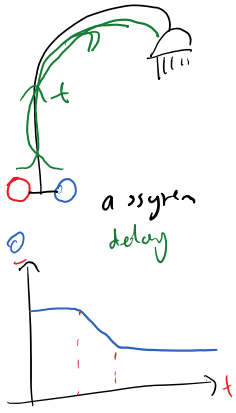
System and real time concepts

11 Eylül 2017 Pazartesi 11:59

System=f(I,O)

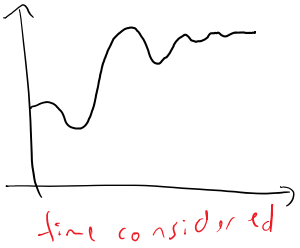


what is output and what is input? Depends on your approach



In shower I is hot and cold water O is average of it

If you dont consider time taken to effect you will be adjusting water forever



RealTime Systems

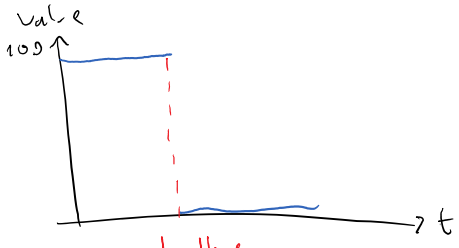
If a system comes with a delay by definition realtime systems is ? Need a new definition:

RT systems different from systems in a way that you have to produce logically correct output in time.

↳ Soft RT

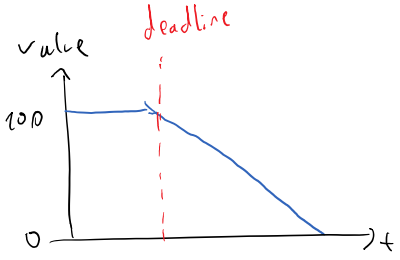
↳ Hard RT ⇒ usually hard RT systems used in class or scheduling

If you miss a deadline you will pay penalty



Hard Realtime System

You are not allowed to miss deadline



Soft Realtime system

After passing deadline value decreasing according to delay

to delay

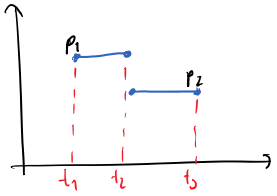
RT Processes

- Periodical
-

Scheduling

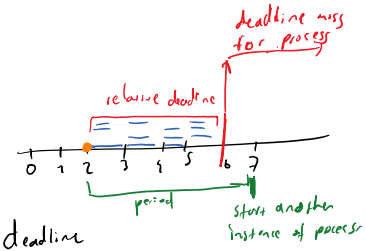
Task / Process

1) Period of the task ($P_1: t_2 - t_1$, $P_2: t_3 - t_2$)
2) Expected (max) execution time
3) Deadline (relative) (usually period) ($P_1: t_2$, $P_2: t_3$)
4) Initial Delay (usually 0) ($P_1: t_1$, $P_2: t_2$)



Usually same tasks multiple instances

2, 5, 3, 4
- relative deadline
- expected time
- period
- delay



(5, 3) = period, ex. t

(5, 3, 4) = period, ex. t, relative deadline

$T_1 = (4, 1)$
$T_2 = (5, 2)$
$T_3 = (20, 5)$

The aim is to calculate feasible schedule

Usually there is 1 resource

Context switch is ignored

Hyper period: smallest common multiple

(4 1) # 5 => 5
(5 2) # 4 => 8
(20 5) # 1 => 5

instances → 18 => Total time to be processing

Hyperperiod

18 | 20 => can be run (feasible)

Which process starts first?

↳ Rate Monotic (RM)
- Definitions are static
- Preemptive (kesintili)
  - important for not missing deadline
- Checking for periods

↳ Deadline monotic
- Checking for deadlines
  - If deadline is not given period will be accepted as deadline and it will give some result with RM