Embedded Operating System
RTOS static priority servers

Graziano Pravadelli

Introduction

- In most real-time applications there are
  - Both periodic and aperiodic tasks
    - Typically periodic tasks are time-driven, hard real-time
    - Typically aperiodic tasks are event-driven, soft or hard RT
- Objectives:
  - Guarantee hard RT tasks
  - Provide good average response time for soft RT tasks

Handling periodic and aperiodic tasks

Immediate service

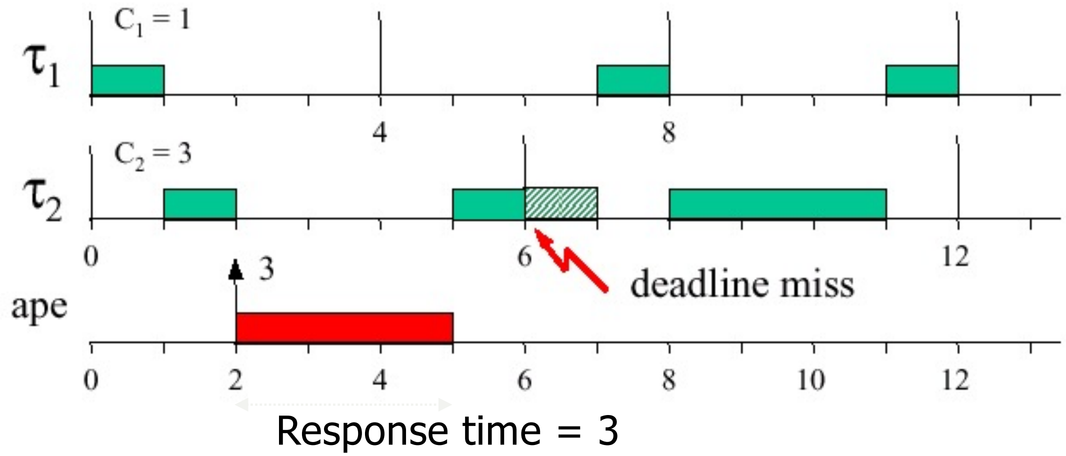
Background scheduling

Aperiodic servers

- Static priority servers
- Dynamic priority servers

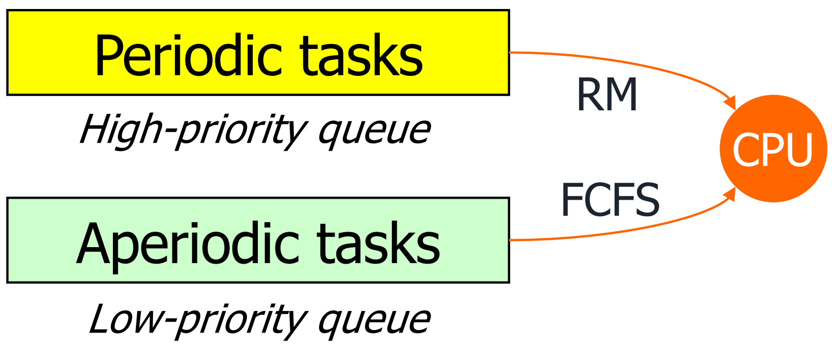
Immediate service

- Aperiodic requests are served as soon as they arrive in the system
- Minimum response times for aperiodic requests
- Low guarantee of periodic tasks



Background scheduling

- Soft aperiodic tasks in the background behind periodic tasks, that is, in the processor time left after scheduling all periodic tasks
- Aperiodic tasks get assigned a priority lower than any periodic one



Background scheduling – Example

	a_i	C_i	T_i
τ_1	0	2	6
τ_2	0	4	10
ape_1	2	1	-
ape_2	12	2	-

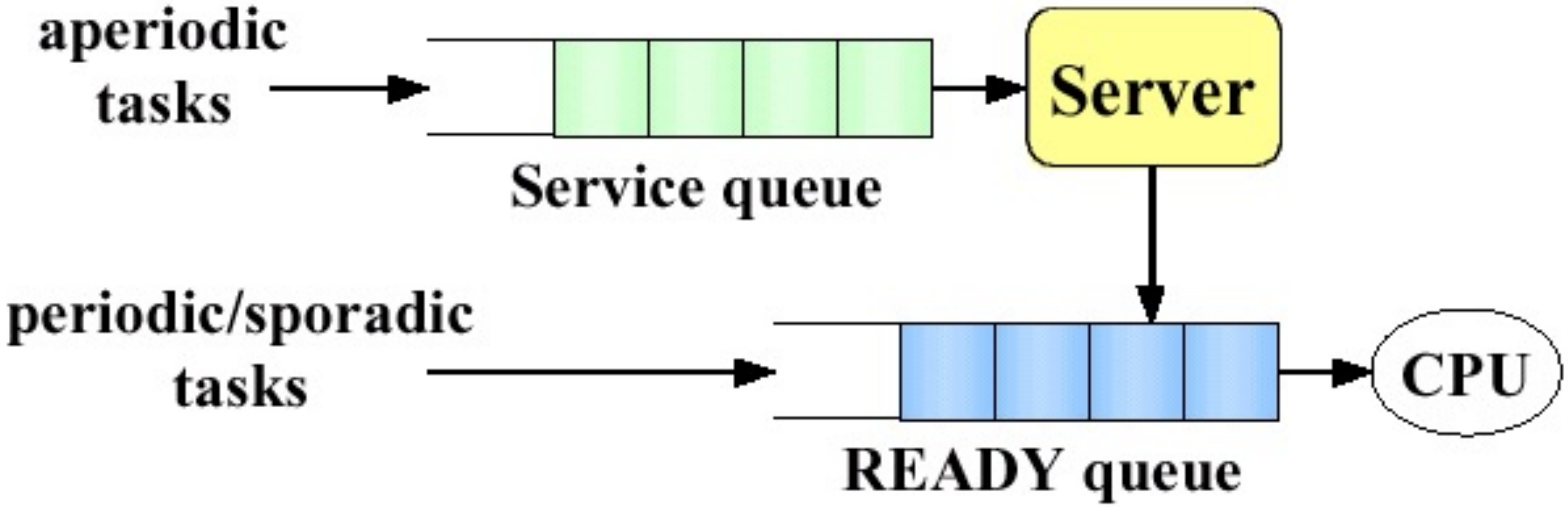
RM for periodic tasks

Background scheduling

- Utilization factor under RM < 1
  - ➡ some processor time is left, it can be used for aperiodic tasks
- High periodic load
  - ➡ bad response time for aperiodic tasks
- Applicable only if no stringent timing requirements for aperiodic tasks
- Major advantage: simplicity

Priority servers

- To achieve more predictable aperiodic task handling
  - A specific periodic task (server) services aperiodic requests
  - The server is assigned
    - a period T_s
    - a computation time C_s (capacity of the server)
  - The server is scheduled like any other periodic task



Priority servers

Classified according to the priority scheme of the periodic scheduler

Static priority servers

- Polling Server
- Deferrable server
- Priority exchange
- Sporadic server
- Slack stealing

Dynamic priority servers

- Dynamic Polling Server
- Dynamic Deferrable Server
- Dynamic Sporadic Server
- Total Bandwidth Server
- Constant Bandwidth Server

Polling Server (PS)

At the beginning of its period	PS is (re)-charged at its full value C_s	
	PS ready to serve any pending aperiodic requests	within the limits of its capacity C_s
If no aperiodic request pending	PS "suspends" itself until beginning of its next period	C_s is discharged to 0
		Processor time is used for periodic tasks
		If aperiodic task arrives after suspension of PS, it is served in the next period
If aperiodic requests pending	PS serves them until C_s>0	

Polling server - example

	C_i	T_i
τ_1	1	4
τ_2	2	6

Server

$C_s = 2$

$T_s = 5$

	C_i	a_i
ape1	2	2
ape2	1	8
ape3	2	12
ape4	1	19

Polling server analysis

- In the worst-case, PS behaves as a periodic task with $U_s = C_s/T_s$

$$U_p + U_s \le U_{lub}(n+1) \qquad \sum_{i=1}^{n} \frac{C_i}{T_i} + \frac{C_s}{T_s} \le (n+1)[2^{1/(n+1)} - 1]$$

- More precise schedulability analysis

$$U_{\text{lub}}^{RM+PS}(n) = U_s + n\left[\left(\frac{2}{U_s+1}\right)^{1/n} - 1\right] \Rightarrow U_p \le n\left[\left(\frac{2}{U_s+1}\right)^{1/n} - 1\right]$$

Deferrable Server

Like PS but	DS preserves its capacity if no requests are pending at invocation of the server	
	Capacity is maintained until server period	Aperiodic requests arriving at any time are served as long as the capacity has not been exhausted
At the beginning of any server period	Capacity is replenished at its full value	But no cumulation

Deferrable server – Example

	C_i	T_i
τ_1	1	4
τ_2	2	6

Server

$C_s = 2$

$T_s = 5$

	C_i	a_i
ape1	2	2
ape2	1	8
ape3	2	12
ape4	1	19

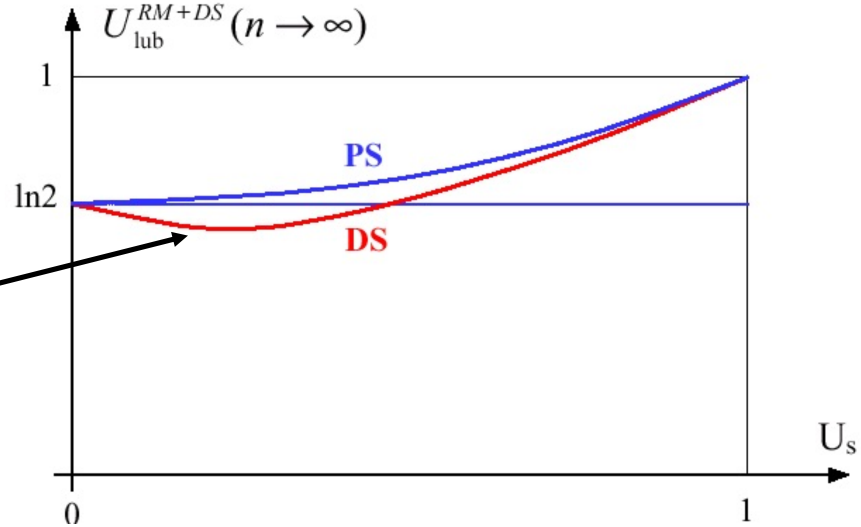
Priority server – PS vs. DF

- Utilization

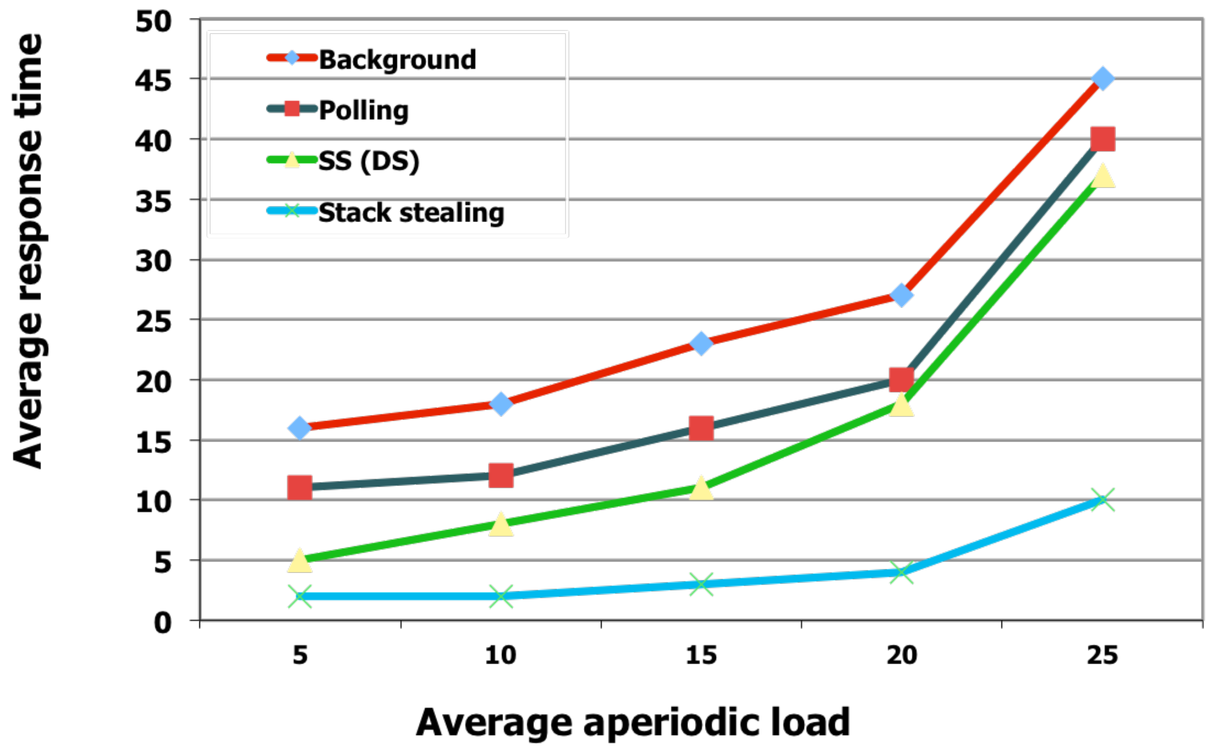
$$U_{\text{lub}}^{RM+DS}(n) = U_s + n\left[\left(\frac{U_s+2}{2U_s+1}\right)^{1/n} - 1\right]$$

- Comparing PS and DS

Keeping the budget improves responsiveness, but decreases the utilization bound.



Comparison of fixed priority servers (1)



(U_p=0.69)

Comparison of fixed priority servers (2)

	Performance	Computational complexity	Memory requirement	Implementation complexity
Background server	**C**	**A**	**A**	**A**
Polling Server	**C**	**A**	**A**	**A**
Deferrable Server	**B**	**A**	**A**	**A**
Priority Exchange	**B**	**B**	**B**	**B**
Slack Stealing	**A**	**C**	**C**	**C**

A=excellent B=good C=poor