## 5. System issues

## Context switch

- Preemption causes time and space overheads which should be duly accounted for in realistic schedulability tests
- Under preemption every single job incurs at least two context switches
  - One at activation to install its execution context
  - One at completion to clean up
- The resulting costs should be charged to the job
  - Knowing the timing behavior of the run-time system we could incorporate overhead costs in schedulability tests

Real-Time Systems

2014/15 UniPD / T. Vardanega

262 of 492











- Let S(i) denote the set of jobs J<sub>{j}</sub> with π<sub>j</sub> = π<sub>i</sub>, excluding J<sub>i</sub> itself
- The time demand equation for  $J_i$  to study in the interval  $0 < t \le \min(D_i, p_i)$  then becomes

$$\Box \ \omega_{i_1}(t) = e_i + B_i + \sum_{S(i)} e_i + \sum_{k=1,..,i-1} \left[ \frac{\omega_{i_1}(t)}{p_k} \right] e_k$$

- This obviously worsens  $J_i$ 's response time
  - But the impact in terms of *schedulability loss* at system level may not be as bad (see later ...)

Real-Time Systems

```
2014/15 UniPD / T. Vardanega
```

267 of 492













2014/15 UniPD / T. Vardanega

Real-Time Systems

273 of 492











## Tick scheduling /2

- The tick scheduler may acknowledge a job's release time 1 (scheduling) tick later than it arrived
  - □ This delay has negative impact on the job's response time
  - □ We also need to assume that a logical place exists where jobs in the "release time arrived but not yet acknowledged" state are held
  - □ The time and space overhead of transferring jobs from that logical place to the ready queue is not null and must be accounted for in the schedulability test together with the time and space overhead of handling clock interrupts

Real-Time Systems

```
2014/15 UniPD / T. Vardanega
```

280 of 492

279 of 492

Dool Time Systems







































