Lecture 17: Introduction to Realtime Process Scheduling

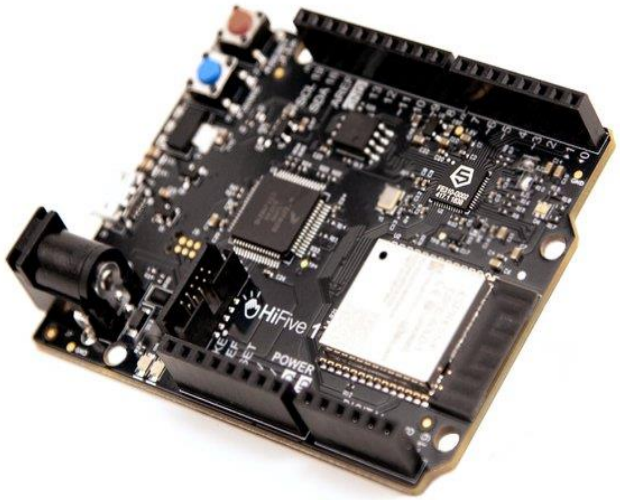
EECS 388 – Fall 2022

© Prof. Mohammad Alian
Lecture notes are based on slides created by Prof. Heechul Yun

Baremetal vs. Operating System

Recommended Reading: Chapter 12 Introduction to Embedded Systems - A Cyber-Physical Systems Approach, by E Lee and S. Seshia.

Process Management
Memory Management
Storage Management
Device Management
Protection and Security



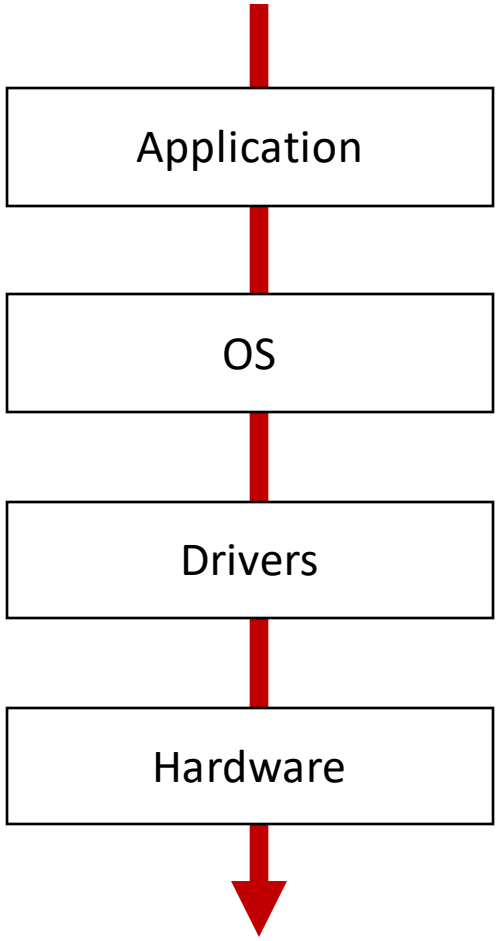
HiFive1 Microcontroller

Raspberry Pi-4 platform

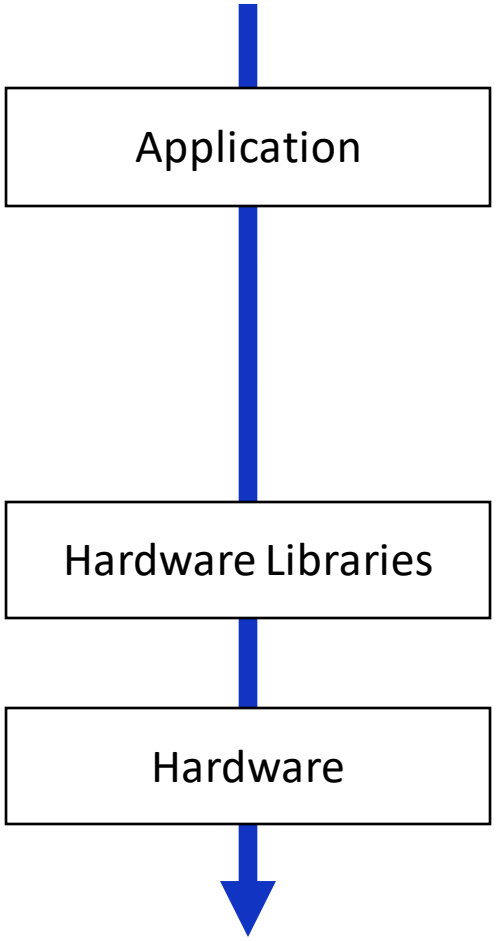
Outline

- Real-time operating systems
- Real-time CPU scheduling theory and practice

OS-Assisted Execution



Baremetal execution



Real-Time Operating System

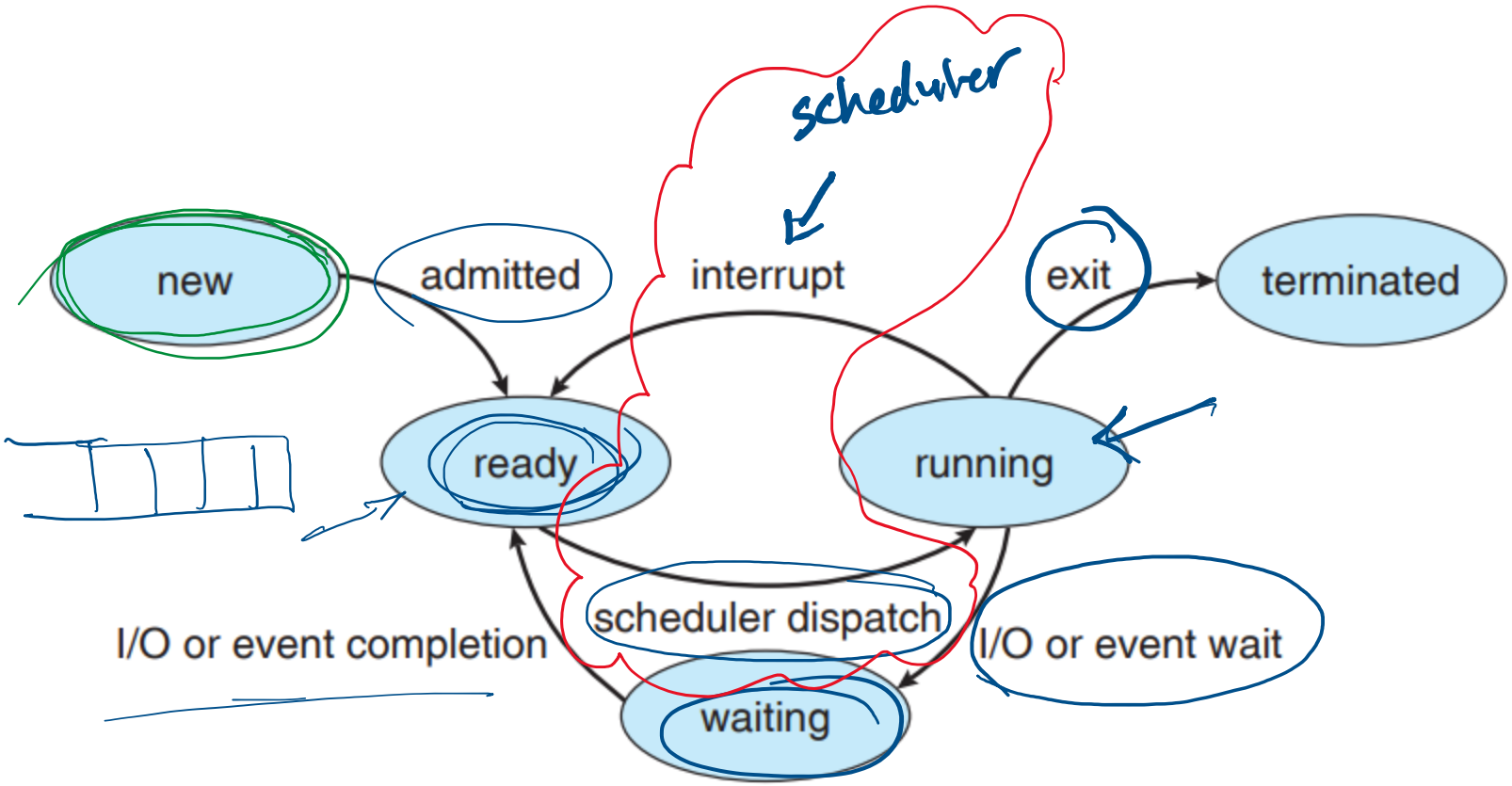
- Often refers to lightweight OS used in embedded systems
  - FreeRTOS, VxWorks, QNX, …
- Specialized to guarantee fast, deterministic real-time response to external events
  - Real-time (CPU) scheduling is key

Real-Time Operating System

- Process scheduling
  - Time share CPU with multiple processes
- Synchronization
  - When we have collaborative processes working on a task
- Input and output
  - Device drivers
- Memory management
- …

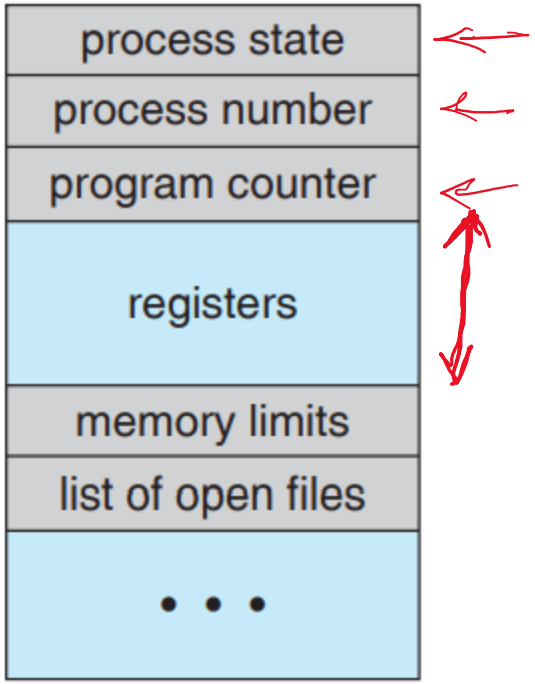
*Unit of CPU utilization

Process* States

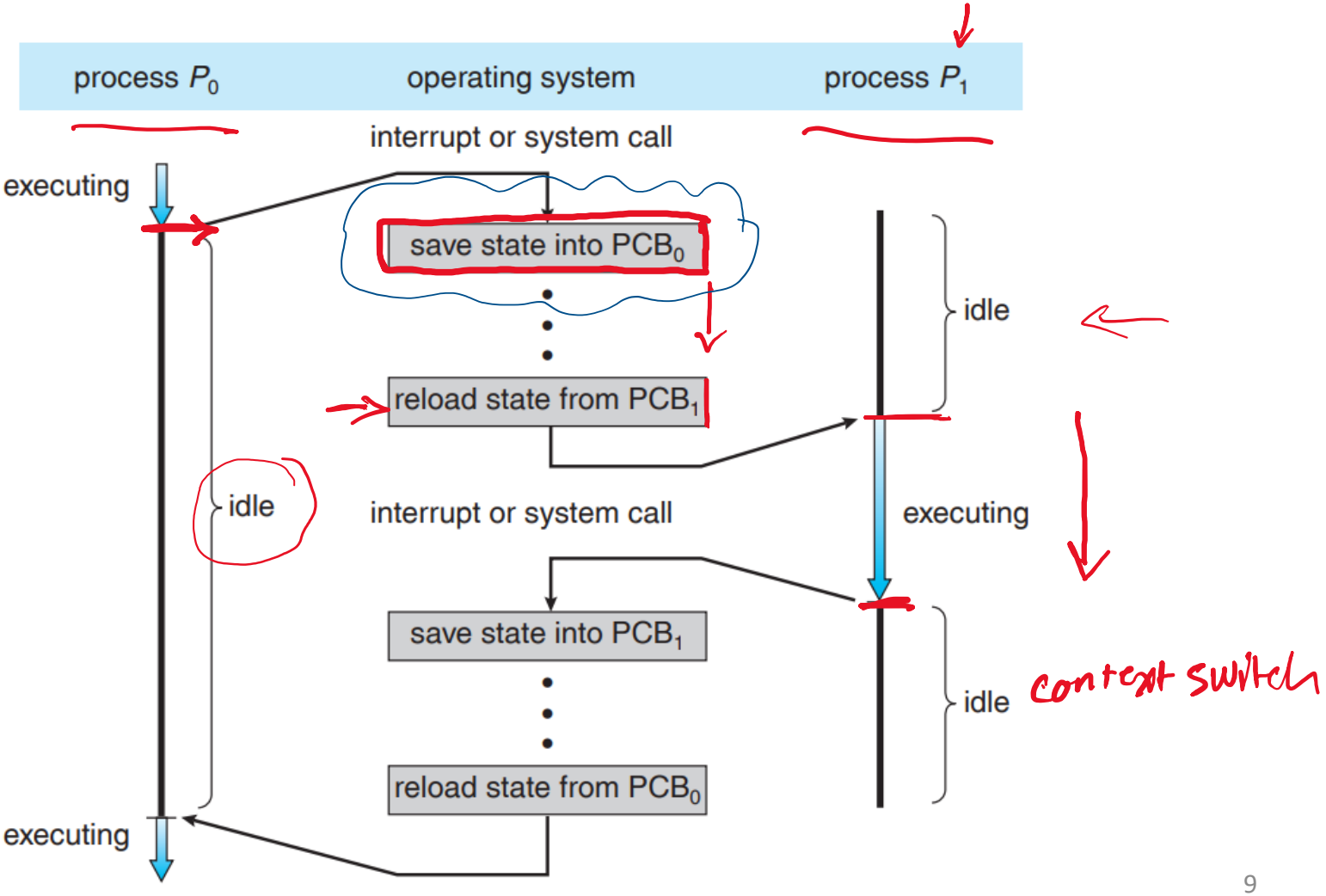


Process Control Block (PCB)

- To support multiple active processes, we need a data structure to keep process information
- In LC-3 we had only one process



Context Switches: Switch from one process to another



Real-Time Systems

- The correctness of the system depends on not only on the logical result of the computation but also on the time at which the results are produced
- **A correct value at a wrong time is a fault.**

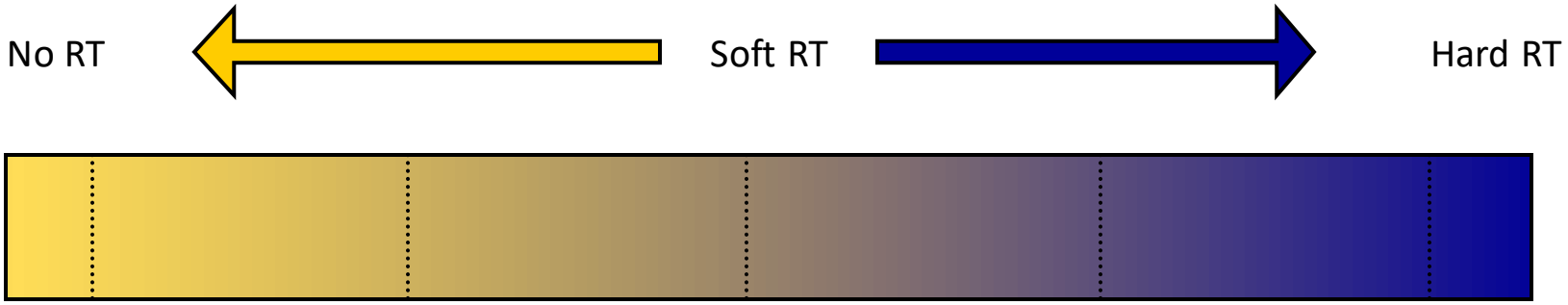
- Two requirements
  - Logical correctness: correct outputs
  - Temporal correctness: outputs at the right time

Soft vs. Hard Real-Time

- Soft real-time: missing deadlines is undesirable, but will not lead to catastrophic consequences
  - Related to the concept of "Quality of Service"
  - Typically interested in average-case response time
  - E.g., reservation systems, media players, phones, etc.

- Hard real-time: missing deadlines is not an option
  - Interested in worst-case response time
  - E.g., airplanes, nuclear plants, military systems, etc.

Real-Time Spectrum



Computer simulation	User interface	Internet video, audio	Tele communication	Flight control

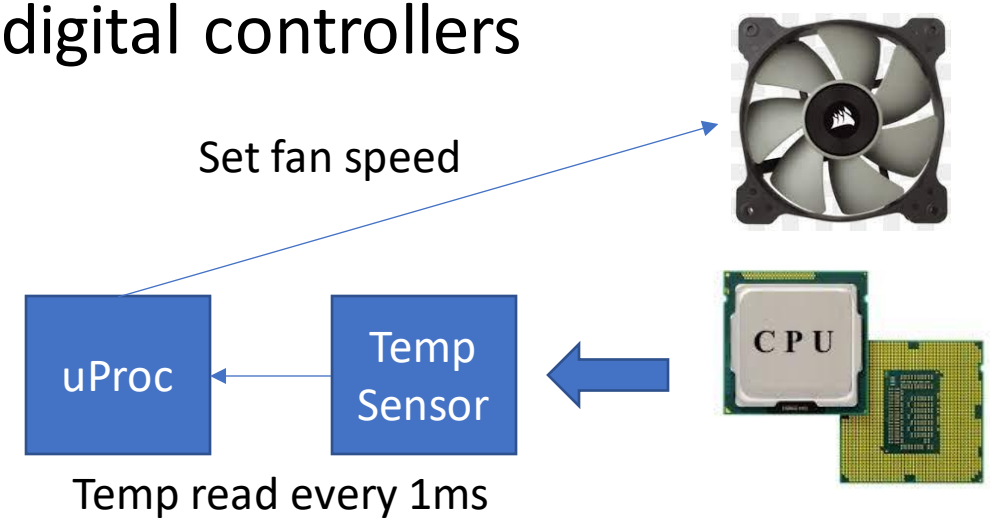
Some Terminologies: Jobs and Tasks

- A job is a unit of computation
  - E.g., serving one interrupt from a keyboard (read one character from keyboard)

- A task is a sequence jobs of the same type
  - E.g., task of serving interrupts from keyboard
    - Each key stroke activates a “job”
  - One task can have many “jobs”

Periodic Tasks

- Time-triggered computation
- Task is activated periodically every T time units
- Each instance of the task is called a job
- Each job has the same relative deadline (usually = to period).
- E.g., most digital controllers



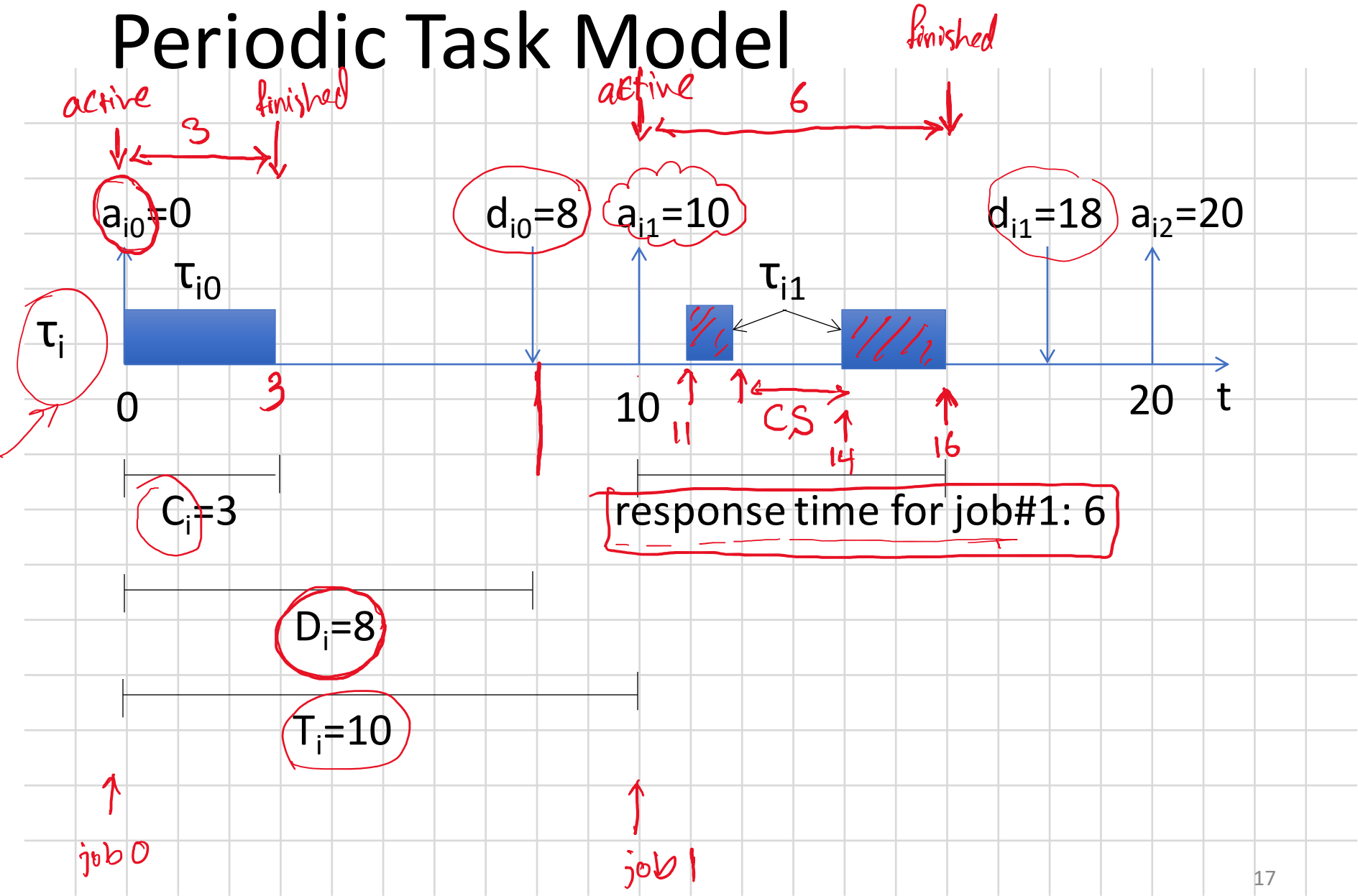
Periodic Task Model

- Task τ_i (N tasks in the system, τ_1 to τ_N)
  - Execution time C_i (sometimes e_i)
  - Relative deadline D_i
  - Period T_i (sometimes p_i)
- Each job τ_{ij} of τ_i (first job: τ_{i0})
  - Activation time $a_{ij} = a_{ij-1} + T_i$ (usually with $a_{i0} = 0$)
  - Absolute deadline $d_{ij} = a_{ij} + D_i$

Periodic Task Model

- Activation time: the instant at which the job becomes ready to execute
- Absolute deadline: specified in absolute time
  - Ex: train and airlines schedules
- Relative deadline: related to the release time
  - Ex: 8 milliseconds after the release time
- By convention, we will refer to an absolute deadline as “d”, and a relative deadline as “D”

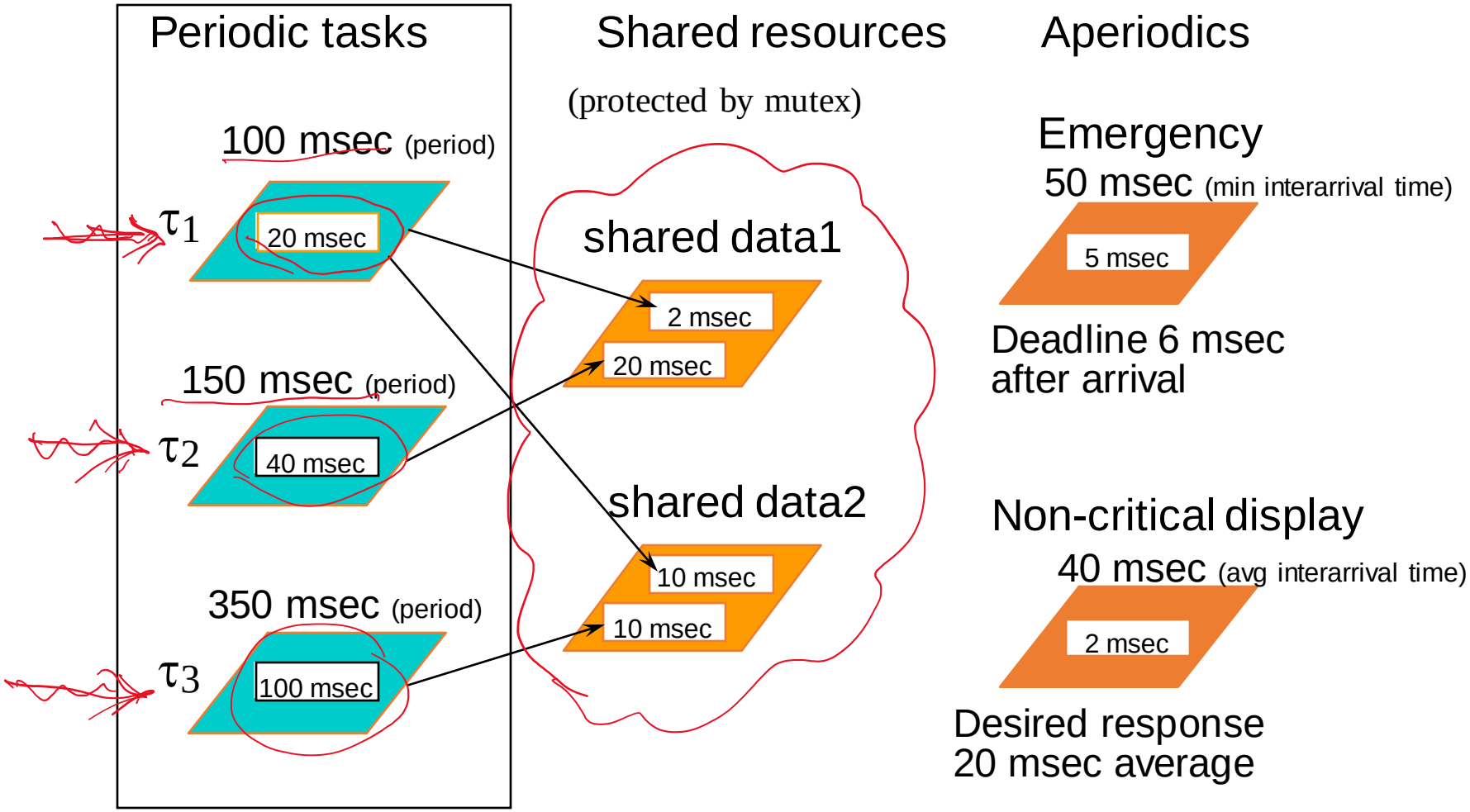
Periodic Task Model



Aperiodic Tasks

- Event-triggered computation
- Task is activated by an external event
- Task runs once to respond to the event
- Relative deadline D: available time to respond to the event
- Usually, minimal inter-arrival time T is assumed to be known

Example Real-Time System



Goal: guarantee that no real-time deadline is missed!!!

Acknowledgements

- These slides draw on materials developed by
  - Lui Sha and Marco Caccamo (UIUC)
  - Rodolfo Pellizzoni (U. Waterloo)
  - Edward A. Lee and Prabal Dutta (UCB) for EECS149/249A