



Seconda prova pratica in itinere: diagnostic voter con TBS

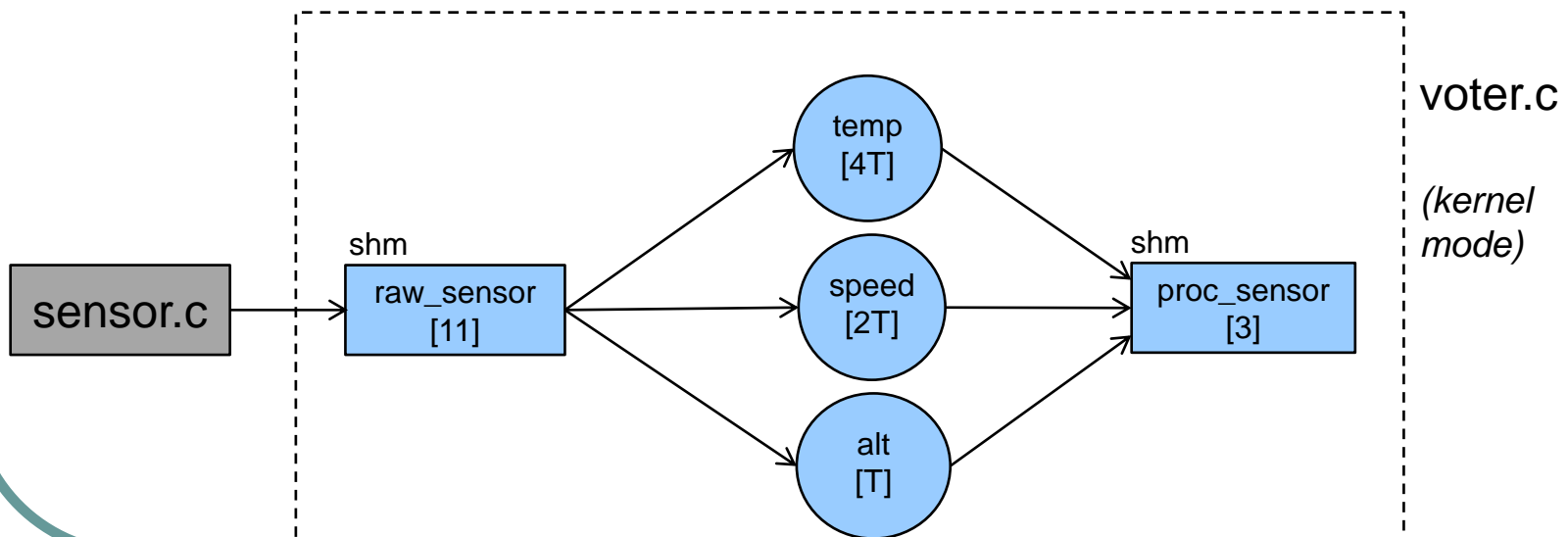
Corso di
Progetto e Sviluppo di
Sistemi in Tempo Reale
a.a. 2020/21

Marcello Cinque

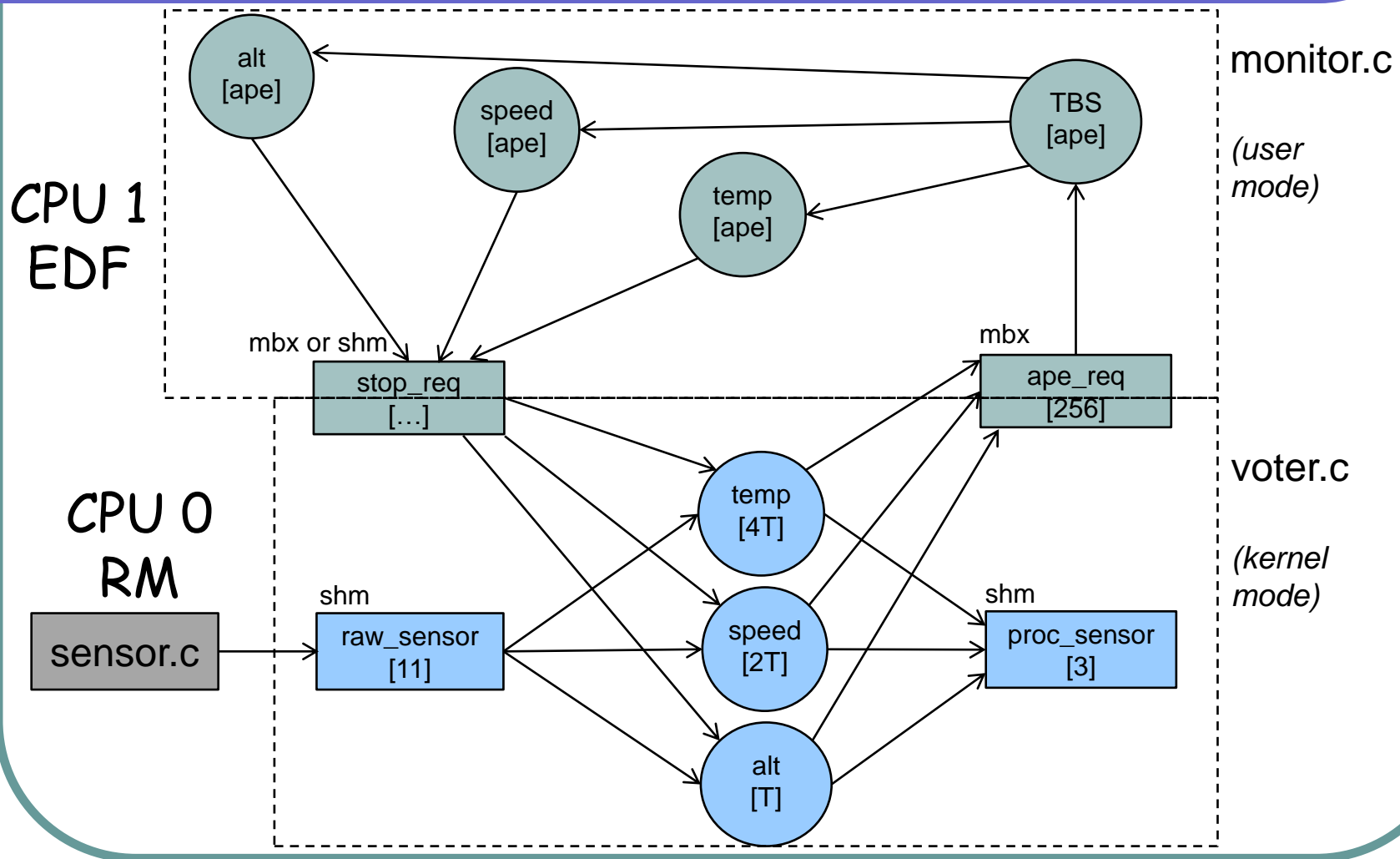
Traccia

- A partire dall'esercizio della prima prova, è richiesto di:
 - Proteggere l'accesso alle risorse condivise con semafori
 - Aggiungere un meccanismo di diagnostica asincrona basata su un Total Bandwidth Server e task aperiodici
- Il TBS e i relativi task aperiodici sono da implementare come task di livello utente in un sorgente a parte, denominato *monitor.c*
- Il TBS e i task aperiodici (questi ultimi schedulati con EDF) sono hard real-time e vanno eseguiti sulla CPU 1.
 - I task hanno funzione diagnostica e comandano lo stop del voter quando accumulano un numero significativo di errori

Schema di partenza



Schema da realizzare



Traccia

- Se uno dei tre task del voter rileva un'anomalia, invia un messaggio, tramite mailbox, al TBS, specificando un codice per il sensore
 - Ad es, 1 per temperatura, 2 per velocità, 3 per altitudine
- Il TBS riceve dalla mailbox e, in base al codice ricevuto, risveglia il task aperiodico corrispondente (*temp*, *speed*, o *alt*, tramite invio di un messaggio diretto); poi torna ad attendere sulla mailbox

Traccia

- Il task aperiodico hard-real time risvegliato:
 - Setta la propria deadline
 - Utilizzando la primitiva per EDF con primo parametro «now» e secondo pari alla deadline calcolata secondo la regola del TBS (come banda del TBS, 50% è un valore ragionevole)
 - Preleva dalla shared memory lo stato corrente di *raw_sensor* e *proc_sensor* per il sensore di competenza (in mutua esclusione) e li salva in una variable locale
 - Scrive a video le informazioni lette
 - Diventando temporaneamente «soft»
 - Oppure tramite un buddy task Linux, risvegliato tramite messaggio diretto
- Dopo 10 notifiche (20 per altitudine), il task manda un messaggio al corrispondente task in voter (tramite la mailbox o shared memory *stop_req*) per fermarlo. Il corrispondente task voter scriverà -1 nel relativo *proc_sensor* e terminerà

Traccia

- Per il tempo dei calcoli dei task aperiodici (da utilizzare per settare le deadline) è possibile:
 - Provare a misurarli, prendendo il massimo osservato
 - Assumere dei valori convenzionali, con l'assunzione ulteriore che il task aperiodico «alt» duri il doppio di «temp» e «speed»

Note

- E' necessario modificare gli script `ldmod` e `remod` presenti nel folder di `sensor` per inserire/rimuovere i moduli necessari ai semafori, messaggi e mailbox
- E' opportuno modificare il sorgente `sensor.c` per far eseguire tutti i task sulla CPU 0.