CMSIS-RTOS

CMSIS-RTOS

- „wrapper“ za FreeRTOS, majhen in enostaven operacijski sistem za vgrajene sisteme
  - kreiranje opravil (tasks/threads)
  - komunikacija med procesi
    - semaforji
    - ključavnice
  - programski časovniki
- Dokumentacija za v2 API: https://www.keil.com/pack/doc/CMSIS/RTOS2/html/rtos_api2.html

Primer opravila

```
void StartDefaultTask(void *argument)
{
  /* Infinite loop */
  for(;;)
  {
    osDelay(1);
  }
}
```

Kreiranje opravil

```
osThreadId_t defaultTaskHandle;

const osThreadAttr_t defaultTask_attributes = {
  .name = "defaultTask",
  .priority = (osPriority_t) osPriorityNormal,
  .stack_size = 128 * 4
};

defaultTaskHandle =
       osThreadNew(StartDefaultTask, NULL,
                   &defaultTask_attributes);
```

Ogrodje programa

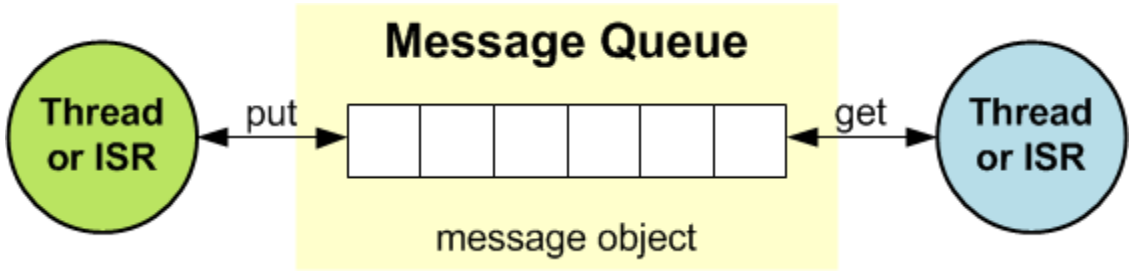
```
void main(){
        /* inicializacija naprav */

        /* kreiranje vseh opravil */
        handle1 = osThreadNew(task1, NULL, &attr1);
        handle2 = osThreadNew(task2, NULL, &attr2);
        handle3 = osThreadNew(task3, NULL, &attr3);

        /* pozenemo razvrščevalnik */
        osKernelStart();

        /* nikoli ne pridemo sem */
}
```

Čakalna vrsta



Čakalna vrsta

```
osMessageQueueId_t vrsta;
```

- kreiranje nove vrste

```
vrsta = osMessageQueueNew( ST_ELEMENTOV,
                           sizeof(ELEMENT),
                           NULL);
```

- zadnji argument so opcijski atributi vrste

Vstavljanje v vrsto

```
osMessageQueuePut(vrsta,
                  &kazalec_na_element,
                  0U,
                  0U);
```

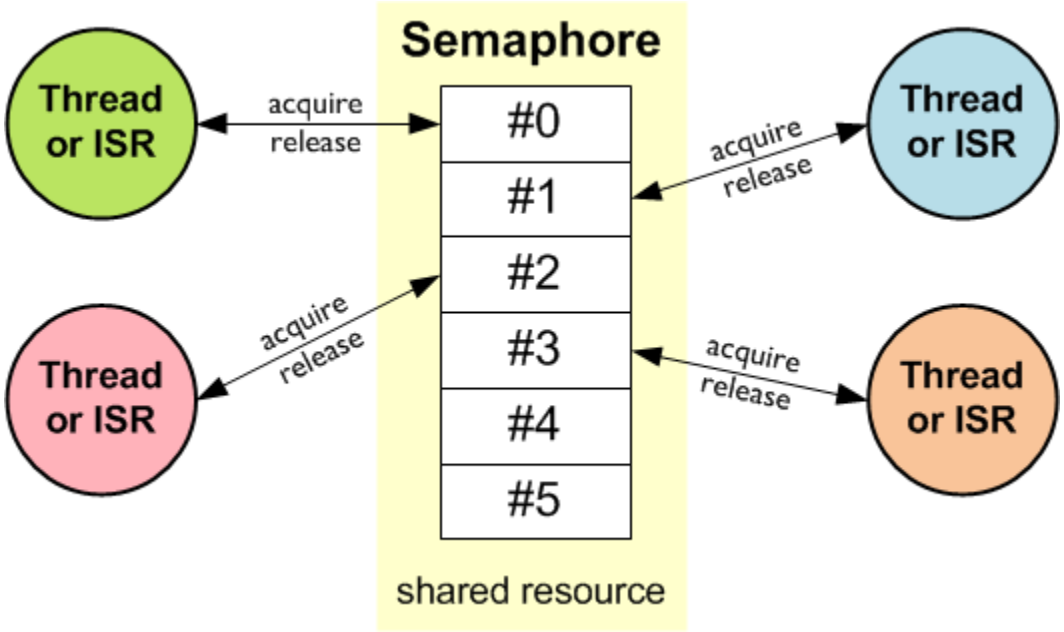
- tretji argument predstavlja prioriteto
- zadnji argument je timeout (v ms) za primer, ko je vrsta polna
  - znotraj PSP mora biti timeout vedno 0

Vstavljanje v vrsto

```
osMessageQueueGet(vrsta,
                  &kazalec_na_element,
                  NULL,
                  0U);
```

- kot tretji argument podamo spremenljivko v katero se zabeleži prioriteta elementa
- zadnji argument je timeout (v ms) za primer, ko je vrsta prazna

Semafor


Semaphore
Thread or ISR
acquire
release
#0
#1
#2
#3
#4
#5
Thread or ISR
acquire
release
Thread or ISR
acquire
release
Thread or ISR
acquire
release
shared resource

Semafor

```
osSemaphoreId_t semafor;
```

- kreiranje

```
semafor = osSemaphoreNew(2U,
                         2U,
                         NULL);
```

- prvi argument je maksimalno število žetonov
- drugi argument je začetno stanje žetonov

Delo s semaforjem

```
val = osSemaphoreAcquire(semafor,
                                10U);
```

- prevzame žeton, če je na voljo. Če ni, čaka toliko ms, kot je podano z drugim argumentom, osWaitForever čaka dokler ni prost
- vrne osOK, če je žeton semaforja prevzet

```
osSemaphoreRelease(semafor);
```

- vrne žeton semaforju

Naloga

- Rešite nalogo zadnje vaje (pošiljanje ukazov za prižiganje/ugašanje LED diod)
  - PSP bere prejete znake in jih vstavlja v čakalno vrsto
  - Task1 bere čakalno vrsto in čaka na znak za novo vrstico - „\n“. Ko se ta pojavi, pošlje ukaz za prižiganje/ugašanje in ID LED (preko druge čakalne vrste).
  - Task2 bere „ukazno“ čakalno vrsto in prižiga/ugaša ustrezne LED diode

  - Task1 naj tudi skrbi za pošiljanje znakov nazaj