

Primer1: Korišćenje flegova (notifikacija)

U ovom primeru taster RA0 podešen je na spoljašnji interapt i uključeno je njegovo generisanje. Takođe, uključen je UART2. Na klik tasterom šalje se flag iz interapta koji se očitava iz blokiranog zadatka koji čeka na flag.

```
void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
{
    osThreadFlagsSet(task1Handle,0x11);
}
-----
void task(void *argument)
{
    /* USER CODE BEGIN 5 */
    /* Infinite loop */

    char txt[] = "Taster je pritisnut!\n";
    for(;;)
    {

        osThreadFlagsWait(0x21, osFlagsWaitAny, osWaitForever);//podesiti
osFlagsWaitAll
        HAL_UART_Transmit(&huart2, txt, strlen(txt), 1000);
    }
}
```

Primer2: Korišćenje ADC

U ovom primeru na analogni ulaz RA0 dovodi se analogni signal koji se očitava ADC1. Nakon završetka konverzije ADC1 generiše prekid, nakon čega se generiše notifikacija sa vrednostima konverzije.

```
void HAL_ADC_ConvCpltCallback(ADC_HandleTypeDef* hadc){
    char txt[] = "Taster je pritisnut!\n";
    uint32_t value = HAL_ADC_GetValue(&hadc1);
    value = (value<<1)|0x01;
    osThreadFlagsSet(task1Handle,value);
}

void task(void *argument)
{
    char txt[20];
    for(;;)
    {
        HAL_ADC_Start_IT(&hadc1);
        uint32_t ret = osThreadFlagsWait(0x1FFFU, osFlagsWaitAny, 5000);
        sprintf(txt,"ADC vrednost je: %lu \n", (ret>>1));
        HAL_UART_Transmit(&huart2, txt, strlen(txt), 100);
        osDelay(100);
    }
}
```