

# Laboration 4

**Miniprojekt**

**Utvärdering av sensorer i FreeRTOS**

Kurs

Realtidssystem IS16

2017-10-17

**Johan Kämpe**



Mjukvaruutvecklare Inbyggda System

[MÖLK Utbildning AB](#)

## Laborationsinstruktioner

### Laborationsserie 4 – Realtidssystem IS16

-----

Laborationsserie 4 är ett miniprojekt. Sladda upp två dimmande LED:ar på PA9 och PA10. Använd programmet som jag lade upp på slack (och moodle) 171016.

Implementera ett ytterligare task för en sensor som du väljer själv, detta task skall polla av sensorn och få dimmandet att sluta över ett visst tröskelvärde på sensorn som du sätter själv.

När du stoppar dimmandet skall du använda en semafor, enligt lektionen den 171016. Om du inte var närvarande får du kolla upp min slideshow för 16:e oktober.

Utvärdera hur det går: skriv en liten redogörelse för

1. hur du gjorde, hur du sladdade upp sensorn,
2. koden som du gjorde,
3. fungerade det eller fungerade det inte? Om du vet varför, skriv ner vad du vet.

Jag har för avsikt att lägga upp en wiki med era redogörelser när jag hinner.

## Val av sensor

Sensorn som valdes är ett räkneverk med ett siffervärdespann på 0 till 9.

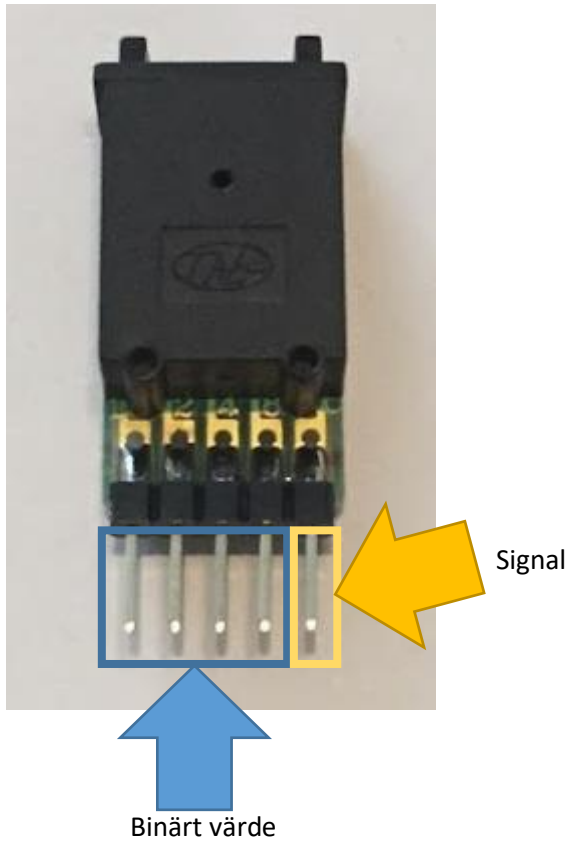
Den ena av knapparna ökar siffervärdet, den andra sänker det. Efter 9 rullar värdet över till 0, och vice versa.



## Räkneverkets anslutningar

De första fyra benen på räkneverket representerar dess aktuella siffervärde i binär form. Ben 1 är den minst signifikanta siffran.

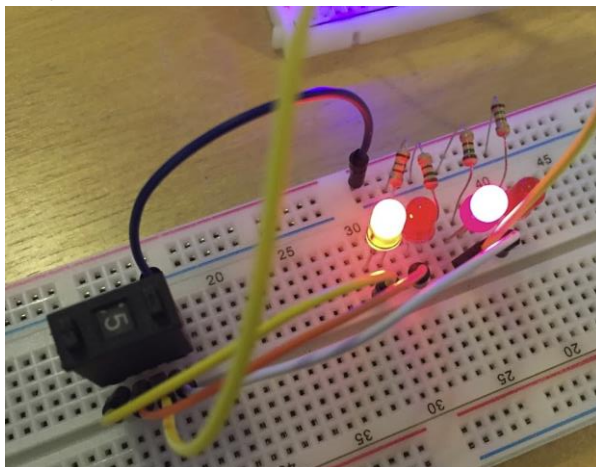
Ben 5 är signalen som släpps igenom vid en binär etta på ett ben.



Exempel med lysdioder kopplade till ben 1-4:

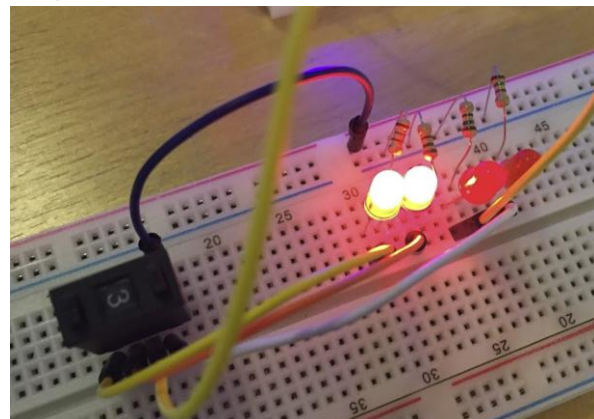
Värde: 5

Ben/binärt värde: 1010



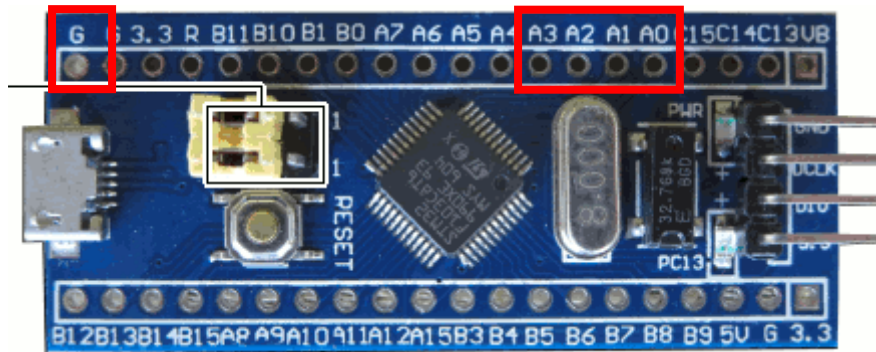
Värde: 3

Ben/binärt värde: 1100



## Uppkoppling av räkneverk till STM32F103-devboard

| Räkneverk Pin | STM32F103 Pin |
|---------------|---------------|
| 1             | PA0           |
| 2             | PA1           |
| 3             | PA2           |
| 4             | PA3           |
| 5             | GND           |



MCUns pins sattes till [pinMode](#) INPUT\_PULLUP, och lästes av med [digitalRead](#).

Vid en etta på räkneverkets pins jordas MCUns pin, och dess värde blir LOW.

## Kod

Notera att en "etta" på en bit från räkneverket blir ett LOW-värde på MCUs pins, då intern pull-up används, och GND som signal.

```
#define COUNTER_CLICKER_PIN_AMOUNT 4
#define COUNTER_CLICKER_TRESHOLD 6
```

Figur 1 Antal pins och tröskelvärde för att stoppa LED-dimming

```
int counter_pins[COUNTER_CLICKER_PIN_AMOUNT] = {
    PA0, PA1, PA2, PA3
};
```

Figur 2 Array med pins som är kopplade till räkneverket

```
xTaskCreate(pollClickerCounter,
    "Poller",
    configMINIMAL_STACK_SIZE,
    NULL,
    tskIDLE_PRIORITY + 2,
    NULL);
```

Figur 3 Task för pollning av räkneverk

```
/* Check clicker counter value */
static void pollClickerCounter(void *pvParameters)
{
    static int counter_value = 0;

    while(1)
    {
        if(xSemaphoreTake(semaphore, (TickType_t)100) == pdTRUE)
        {
            /* Keep semaphore while counter value is under treshold,
            LEDs will stop fading. */
            do{
                counter_value = 0;
                for(int i = 0; i < COUNTER_CLICKER_PIN_AMOUNT; i++)
                {
                    counter_value |= (!digitalRead(counter_pins[i]) ?
                        (1 << i) : 0);
                }
            }while(counter_value < COUNTER_CLICKER_TRESHOLD);

            xSemaphoreGive(semaphore);
        }
        else;
    }
}
```

Figur 4 Pollningsfunktionen, släpper ej semafor om tröskelvärdet ej är nått

```

for (;;)
{
    /* Do one full fade cycle if semaphore is available */
    if(xSemaphoreTake(semaphore, (TickType_t)0) == pdTRUE)
    {
        xSemaphoreGive(semaphore); // Give semaphore and complete cycle

        int ix;
        vTaskDelay(Lfreq->longDelay);

        for (ix = 0; ix < 256; ix++)
        {
            analogWrite(Lfreq->LED, ix);
            vTaskDelay(Lfreq->shortDelay);
        }

        vTaskDelay(Lfreq->longDelay);

        for (ix = 255; ix >= 0; ix--)
        {
            analogWrite(Lfreq->LED, ix);
            vTaskDelay(Lfreq->shortDelay);
        }
    }
}

```

Figur 5 Modifierad vLEDFlashTask, slutar dimma om semafor är upptagen.

## Slutsats

Räkneverket fungerar med FreeRTOS