

# Control de frecvență al ledurilor

## E-mailuri:

[sergiu-stefan.corban@student.tuiasi.ro](mailto:sergiu-stefan.corban@student.tuiasi.ro)

[florin-cristian.velovici@student.tuiasi.ro](mailto:florin-cristian.velovici@student.tuiasi.ro)

[razvan-andrei.olteanu@student.tuiasi.ro](mailto:razvan-andrei.olteanu@student.tuiasi.ro)

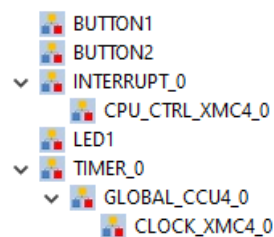
## Descriere și folosire:

Aplicație care aprinde ledul LED1 (P5.9) de pe placă XMC4700 cu o rata periodică inițială de 1.0s. Folosind butoanele, rata de aprindere poate fi ajustată în trepte de 100 ms. Folosim butonul Button1 (P15.13) pentru a crește frecvența de aprindere, butonul Button2 (P15.12) pentru a scădea frecvența de aprindere și butonul Reset pentru a reseta rata frecvenței înapoi la cea inițială.

## Componente și aplicații:

Componente software folosite:

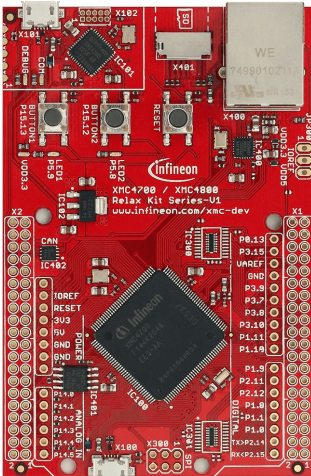
- Dave-4.4.2
  - Aplicații DAVE:



- Segger J-Link

Componente hardware folosite:

- XMC4700 Relax Kit



- USB 2.0 to USB Mini cable
- Windows PC

### Video demonstrativ:

<https://www.youtube.com/watch?v=KIOjN3auCtU>

### Cod:

main.c

```
#include <DAVE.h>

#define TIMER_MILLISECONDS 1000 * 100U

int main(void)
{
    DAVE_STATUS_t status;
    uint32_t timer_interval = 1000 * TIMER_MILLISECONDS;
    bool button_edge = true;

    status = DAVE_Init(); /* Initializarea aplicațiilor specifice Dave */

    if (status == DAVE_STATUS_FAILURE)
    {
        XMC_DEBUG("Eroare la initializarea aplicatiilor specifice Dave\n");
        while (1U)
```

```

    {
    }
}

while (1U)
{
    /* Verificam daca este vreun buton apasat*/
    if (!DIGITAL_IO_GetInput(&BUTTON1) || !DIGITAL_IO_GetInput(&BUTTON2))
    {
        if (button_edge == true)
        {
            /* Care buton a fost apasat? */
            if (!DIGITAL_IO_GetInput(&BUTTON1))
            {
                /* Butonul 1 a fost apasat. Scadem frecventa timpului de
aprindere a ledului. */
                timer_interval = (timer_interval > 100 * TIMER_MILLISECONDS) ?
(timer_interval - 100 * TIMER_MILLISECONDS) : timer_interval;
            }
            else
            {
                /* Butonul 2 a fost apasat. Crestem frecventa timpului de
aprindere a ledului. */
                timer_interval = (timer_interval < 1500 * TIMER_MILLISECONDS) ?
(timer_interval + 100 * TIMER_MILLISECONDS) : timer_interval;
            }
            TIMER_Stop(&TIMER_0);
            TIMER_SetTimeInterval(&TIMER_0, timer_interval);
            TIMER_Start(&TIMER_0);
        }
        button_edge = false;
    }
    else
    {
        button_edge = true;
    }
}

```

```
    }  
}  
  
void UserIRQHandler (void)  
{  
  
    DIGITAL_IO_ToggleOutput (&LED1);  
}
```