# PCA - PWM out

#### **MICLAB**

Név: Pilter Zsófia, Vad Avar

Dátum: 2024.11.18. Mérőhely: 1 jobb és bal

#### Jegyzőkönyv készítése

A jegyzőkönyvek az órán végzett munka dokumentálására szolgálnak. A letölthető minta jegyzőkönyvet kell kiegészíteni a megfelelő információkkal: név, dátum, mérőhely (pl. 3. jobb), a feladatokhoz tartozó esetleges kifejtendő válaszokkal, valamint a kódok lényeges részével.

A jegyzőkönyveket a CooSpace-en kell feltölteni, külön pdf formátumban csatolni kell a jegyzőkönyvet (a fájl neve a következő mintát kövesse: NagyJ.KissB.o3.pdf), egy külön zip fájlban pedig a kódokat (\*.c, \*.cwg). Amennyiben probléma merül fel a beadás során, az anyagokat az oktató e-mail címére kell elküldeni, levél tárgya legyen pl. MicLab o3.

### 1. feladat – PCA konfigurálása, LED vezérlése PWM módban

A Port I/O Mapping ablakban engedélyezze a PCAo\_CEXo kimenetet. Pin-skipp segítségével tolja el a PCAo\_CEXo-át a P1.4-be. Konfigurálja be a PCA perifériát 8 bites módban. Az órajelét a Timer o overflow adja. A Cannel o legyen Edge-aliagned PWM módban.

Készítsen egy programot, ami a PCA segítségével 1 másodperces periódusidővel villogtatja az EFM8BB1LCK panelen lévő LEDo-át 25%-os kitöltési tényezővel. Ehhez először a Timer o-át konfigurálja be, majd a programban állítsa be PCA capture/compare regiszterét a megfelelő értékre.

Tipp: az 1 s-os periódusidőt el lehet érni a megfelelő beállításokkal a konfigurátorban.

A program részekre bontott forráskódja (Config, Main.c, Interrupts.c, ha van):

# Config:

```
cproperty object="P1.3" propertyId="ports.settings.skip" value="Skipped"/>
    cproperty object="P1.4" propertyId="ports.settings.iomode" value="Digital"
Push-Pull Output"/>
    <property object="P1.4" propertyId="ports.settings.outputmode" value="Push-</pre>
pull"/>
    cproperty object="PBCFG_0" propertyId="pbcfg.settings.enablecrossbar"
value="Enabled"/>
    <property object="PCACH_0" propertyId="ABPeripheral.included" value="true"/>
<property object="PCACH_0"</pre>
propertyId="pcach.pcachannel.capturecomparehighbyte" value="65"/>
    cproperty object="PCACH_0" propertyId="pcach.pcachannel.capturecomparelowbyte"
value="65"/>
    cproperty object="PCACH_0"
\verb|propertyId="pcach.pcachannelcapturecomparemode.enable channelcomparator function"|
value="Enabled"/>
    cproperty object="PCACH 0"
propertyId="pcach.pcachannelcapturecomparemode.enablechannelpulsewidthmodulationmo
den" value="Enabled"/>
    cproperty object="PCACH 0"
propertyId="pcach.pcacontrol.channelcapturecomparemode" value="Predefined 8~11-bit
pulse modulator"/>
    <property object="PCACH_1" propertyId="ABPeripheral.included" value="true"/>
<property object="PCACH_2" propertyId="ABPeripheral.included" value="true"/>
    propertyId="pca.pcacountertimerconfiguration.pcaclockfrequency" value="256.000
    cproperty object="PCA_0"
propertyId="pca.pcacountertimerconfiguration.pcaclockfrequencyintegervalue"
value="256"/>
    cproperty object="PCA_0"
propertyId="pca.pcacountertimerconfiguration.pcaclockperiod" value="3.906 mS"/>
    cproperty object="PCA_0"
propertyId="pca.pcacountertimerconfiguration.selectpcacountertimerpulse"
value="Timer 0 overflow"/>
    <property object="PCA_0" propertyId="pca.pcacountertimerruncontrol.pcarun"</pre>
value="Start"/>
    cproperty object="TIMER01_0" propertyId="ABPeripheral.included" value="true"/>
    propertyId="timer01.timer0highbyte.timer0highbyte" value="7"/>
    cproperty object="TIMER01_0"
propertyId="timer01.timer0mode2:8bitcountertimerwithautoreload.targetoverflowfrequ
ency" value="256"/>
    cproperty object="TIMER01_0"
propertyId="timer01.timer0mode2:8bitcountertimerwithautoreload.timerreloadvalue"
value="7"/>
    cproperty object="TIMER16_2" propertyId="ABPeripheral.included" value="true"/>
    cproperty object="TIMER_SETUP_0" propertyId="ABPeripheral.included"
value="true"/>
    cproperty object="TIMER_SETUP_0"
propertyId="timer_setup.clockcontrol0.timer01prescale" value="SYSCLK / 48"/>
    cproperty object="TIMER_SETUP_0" propertyId="timer_setup.timer0.mode"
value="Mode 2, 8-bit Counter/Timer with Auto-Reload"/>
```

```
cproperty object="TIMER_SETUP_0"
propertyId="timer_setup.timer0.timerrunningstate" value="Timer is Running"/>
   cproperty object="TIMER_SETUP_0"
propertyId="timer setup.timer0.timerswitch1:runcontrol" value="Start"/>
   cproperty object="TIMER_SETUP_0"
propertyId="timer_setup.timer01control.timer0runcontrol" value="Start"/>
   cproperty object="WDT_0" propertyId="ABPeripheral.included" value="true"/>
   cproperty object="WDT_0" propertyId="wdt.watchdogcontrol.wdtenable"
value="Disable"/>
 </mode>
 <modeTransition>
   cproperty object="RESET → DefaultMode"
propertyId="modeTransition.source" value="RESET"/>
   cproperty object="RESET → DefaultMode"
propertyId="modeTransition.target" value="DefaultMode"/>
 </modeTransition>
</device:XMLDevice>
Main.c:
// src/feladat11-01_main.c: generated by Hardware Configurator
// This file will be updated when saving a document.
// leave the sections inside the "\{[...]" comment tags alone
// or they will be overwritten!!
// Includes
//-----
#include <SI EFM8BB1 Register Enums.h>
                                         // SFR declarations
#include "InitDevice.h"
// $[Generated Includes]
// [Generated Includes]$
//-----
// SiLabs_Startup() Routine
// -----
// This function is called immediately after reset, before the initialization
// code is run in SILABS_STARTUP.A51 (which runs before main() ). This is a
// useful place to disable the watchdog timer, which is enable by default
// and may trigger before main() in some instances.
//-----
void SiLabs_Startup (void)
 // $[SiLabs Startup]
 // [SiLabs Startup]$
//-----
// main() Routine
// -----
int main (void)
 // Call hardware initialization routine
 enter_DefaultMode_from_RESET();
 while (1)
 {
```

```
// $[Generated Run-time code]
   // [Generated Run-time code]$
}
```

Az elkészült programot be kell mutatni!

A gyakorlatvezető ellenőrizte:

- <u>Igen</u>
- Nem

A program működött:

- Igen
- Nem

# 2. feladat – LED fényerejének változtatása nyomógombbal

A kiegészítő panelen lévő SW1 nyomógombbal 0%-os kitöltési tényezőtől 75%-os kitöltési tényezőig 4 lépésben növelje a LEDo fényerejét gombnyomásra. Az utolsó utáni lépés után ismét az első következzen. A PWM frekvenciája legyen 100 Hz feletti. Amig az SW1 le van nyomva, addig folyamatosan léptesse a fényerőt egy timer interruptban. A 4 lépés megtételéhez szükséges idő legyen 1 másodperc.

A program részekre bontott forráskódja (Config, Main.c, Interrupts.c, ha van):

# Config:

```
<?xml version="1.0" encoding="ASCII"?>
<device:XMLDevice xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"</pre>
xmlns:device="http://www.silabs.com/ss/hwconfig/document/device.ecore"
name="EFM8BB10F8G-A-QSOP24" partId="mcu.8051.efm8.bb1.efm8bb10f8g-a-qsop24"
version="4.0.0" contextId="%DEFAULT%">
  <mode name="DefaultMode">
    <property object="CROSSBAR0" propertyId="xbar0.pca0.cex" value="CEX0"/>
    <property object="DefaultMode" propertyId="mode.diagramLocation" value="100,</pre>
100"/>
    cproperty object="INTERRUPT_0" propertyId="ABPeripheral.included"
value="true"/>
    cproperty object="INTERRUPT 0"
propertyId="interrupt.interruptenable.enableallinterrupts" value="Enabled"/>
    cproperty object="INTERRUPT_0"
propertyId="interrupt.interruptenable.enabletimer2interrupt" value="Enabled"/>
    cproperty object="P0.0" propertyId="ports.settings.skip" value="Skipped"/>
cproperty object="P0.1" propertyId="ports.settings.skip" value="Skipped"/>
cproperty object="P0.2" propertyId="ports.settings.skip" value="Skipped"/>
cproperty object="P0.2" propertyId="ports.settings.skip" value="Skipped"/>

    cproperty object="P0.3" propertyId="ports.settings.skip" value="Skipped"/>
    cproperty object="P0.4" propertyId="ports.settings.skip" value="Skipped"/>
    cproperty object="P0.5" propertyId="ports.settings.skip" value="Skipped"/>
    cproperty object="P0.6" propertyId="ports.settings.skip" value="Skipped"/>
    cproperty object="P0.7" propertyId="ports.settings.skip" value="Skipped"/>
    cproperty object="P1.0" propertyId="ports.settings.skip" value="Skipped"/>
```

```
<property object="P1.4" propertyId="ports.settings.iomode" value="Digital</pre>
Push-Pull Output"/>
    <property object="P1.4" propertyId="ports.settings.outputmode" value="Push-</pre>
pull"/>
    <property object="PBCFG_0" propertyId="pbcfg.settings.enablecrossbar"</pre>
value="Enabled"/>
    cproperty object="PCACH_0" propertyId="ABPeripheral.included" value="true"/>
    cproperty object="PCACH 0"
propertyId="pcach.pcachannel.capturecomparehighbyte" value="1"/>
    <property object="PCACH_0" propertyId="pcach.pcachannel.capturecomparelowbyte"</pre>
value="1"/>
    cproperty object="PCACH 0"
propertyId="pcach.pcachannelcapturecomparemode.enablechannelcomparatorfunction"
value="Enabled"/>
    cproperty object="PCACH 0"
propertyId="pcach.pcachannelcapturecomparemode.enablechannelpulsewidthmodulationmo
den" value="Enabled"/>
    cproperty object="PCACH_0"
propertyId="pcach.pcacontrol.channelcapturecomparemode" value="Predefined 8~11-bit
pulse modulator"/>
    <property object="PCACH_1" propertyId="ABPeripheral.included" value="true"/>
    cproperty object="PCACH_2" propertyId="ABPeripheral.included" value="true"/>
    cproperty object="PCA_0" propertyId="ABPeripheral.included" value="true"/>
    property object="PCA_0"
propertyId="pca.pcacountertimerconfiguration.pcaclockfrequency" value="255.208
kHz"/>
    cproperty object="PCA_0"
propertyId="pca.pcacountertimerconfiguration.pcaclockfrequencyintegervalue"
value="255208.333333"/>
    cproperty object="PCA_0"
propertyId="pca.pcacountertimerconfiguration.pcaclockperiod" value="3.918 uS"/>
    <property object="PCA_0" propertyId="pca.pcacountertimerruncontrol.pcarun"</pre>
value="Start"/>
    <property object="TIMER01_0" propertyId="ABPeripheral.included" value="true"/>
    cproperty object="TIMER16_2" propertyId="ABPeripheral.included" value="true"/>
    cproperty object="TIMER16_2" propertyId="timer16.control.runcontrol"
value="Start"/>
    value="Timer is Running"/>
    cproperty object="TIMER16 2"
propertyId="timer16.initandreloadvalue.targetoverflowfrequency" value="4"/>
    cproperty object="TIMER16_2"
propertyId="timer16.initandreloadvalue.timerreloadvalue" value="1734"/>
    cproperty object="TIMER16 2"
propertyId="timer16.reloadhighbyte.reloadhighbyte" value="6"/>
    roperty object="TIMER16_2" propertyId="timer16.reloadlowbyte.reloadlowbyte"
value="198"/>
    cproperty object="TIMER16_3" propertyId="ABPeripheral.included" value="true"/>
    cproperty object="TIMER_SETUP_0" propertyId="ABPeripheral.included"
value="true"/>
    cproperty object="WDT_0" propertyId="ABPeripheral.included" value="true"/>
    cproperty object="WDT_0" propertyId="wdt.watchdogcontrol.wdtenable"
value="Disable"/>
  </mode>
  <modeTransition>
    cproperty object="RESET → DefaultMode"
propertyId="modeTransition.source" value="RESET"/>
    cproperty object="RESET → DefaultMode"
propertyId="modeTransition.target" value="DefaultMode"/>
```

```
</modeTransition>
</device:XMLDevice>
```

#### Main.c:

```
// src/feladat11-02_main.c: generated by Hardware Configurator
// This file will be updated when saving a document.
// leave the sections inside the "\{[\ldots]" comment tags alone
// or they will be overwritten!!
//-----
//-----
// Includes
//-----
#include <SI_EFM8BB1_Register_Enums.h>
                             // SFR declarations
#include "InitDevice.h"
// $[Generated Includes]
// [Generated Includes]$
#define SW1 BTN P0 B1
                  _____
// SiLabs_Startup() Routine
// -----
// This function is called immediately after reset, before the initialization
// code is run in SILABS STARTUP.A51 (which runs before main() ). This is a
// useful place to disable the watchdog timer, which is enable by default
// and may trigger before main() in some instances.
//-----
void SiLabs_Startup (void)
 // $[SiLabs Startup]
 // [SiLabs Startup]$
}
//-----
// main() Routine
              _____
// -----
int main (void)
 // Call hardware initialization routine
 enter_DefaultMode_from_RESET();
 while (1)
  // $[Generated Run-time code]
  // [Generated Run-time code]$
}
Interrupts.c:
// src/Interrupts.c: generated by Hardware Configurator
```

```
//
// This file will be regenerated when saving a document.
// leave the sections inside the "\{[\ldots]" comment tags alone
// or they will be overwritten!
//-----
// USER INCLUDES
#include <SI_EFM8BB1_Register_Enums.h>
#define SW1 BTN P0 B1
#define STATE1 64u
#define STATE2 128u
#define STATE3 191u
#define STATE4 1u
//-----
// TIMER2 ISR
//-----
//
// TIMER2 ISR Content goes here. Remember to clear flag bits:
// TMR2CN0::TF2H (Timer # High Byte Overflow Flag)
// TMR2CN0::TF2L (Timer # Low Byte Overflow Flag)
//-----
uint8_t pca_register;
SI_INTERRUPT (TIMER2_ISR, TIMER2_IRQn)
 if (!SW1 BTN)
    pca_register = PCA0CPH0;
    switch (pca_register)
      case STATE4:
       PCA0CPH0 = STATE1;
       break;
      case STATE1:
       PCA0CPH0 = STATE2;
       break;
      case STATE2:
       PCA0CPH0 = STATE3;
       break;
      case STATE3:
       PCAOCPHO = STATE4;
       break;
      }
   }
}
```

Az elkészült programot be kell mutatni!

A gyakorlatvezető ellenőrizte:

- <u>Igen</u>
- Nem

A program működött:

- Igen
- Nem

# 3. feladat – LED fényerejének állítása potenciométerrel

Mérje folyamatosan az ADC-vel a kiegészítő panelen lévő potenciométert interrupt módban és a mért értékkel arányosan változtassa a LEDo kitöltési tényezőjét o ás ~100% között. A PWM frekvenciája legyen 100 Hz feletti. Az ADC kód legyen 8 bites és legyen balra igazítva. a Voltage Reference legyen 3,3 V. Elég a felső 8 bitet kiolvasni (ADCoH regiszter).

A program részekre bontott forráskódja (Config, Main.c, Interrupts.c, ha van):

#### Config:

```
<?xml version="1.0" encoding="ASCII"?>
<device:XMLDevice xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"</pre>
xmlns:device="http://www.silabs.com/ss/hwconfig/document/device.ecore"
name="EFM8BB10F8G-A-QSOP24" partId="mcu.8051.efm8.bb1.efm8bb10f8g-a-qsop24"
version="4.0.0" contextId="%DEFAULT%">
  <mode name="DefaultMode">
    cproperty object="ADC_0" propertyId="ABPeripheral.included" value="true"/>
    cproperty object="ADC 0"
propertyId="adc.accumulatorconfiguration.accumulatorshiftandjustify" value="Left
justified"/>
    cproperty object="ADC_0" propertyId="adc.configuration.gaincontrol" value="1x
gain"/>
    <property object="ADC 0" propertyId="adc.configuration.sarclockdivider"</pre>
value="0"/>
    <property object="ADC_0" propertyId="adc.control.enableadc" value="Enabled"/>
    cproperty object="ADC_0" propertyId="adc.control.startofconversion"
value="Timer 2 overflow"/>
    cproperty object="ADC 0"
propertyId="adc.multiplexerselection.positiveinputselection" value="ADC0.15
    <property object="ADC 0" propertyId="adc.view.view" value="Advanced"/>
    <property object="CROSSBAR0" propertyId="xbar0.pca0.cex" value="CEX0"/>
    roperty object="DefaultMode" propertyId="mode.diagramLocation" value="100,
100"/>
    cproperty object="INTERRUPT_0" propertyId="ABPeripheral.included"
value="true"/>
    cproperty object="INTERRUPT 0"
propertyId="interrupt.extendedinterruptenable1.enableadc0conversioncompleteinterru
pt" value="Enabled"/>
    cproperty object="INTERRUPT 0"
propertyId="interrupt.interruptenable.enableallinterrupts" value="Enabled"/>
    <property object="P0.0" propertyId="ports.settings.skip" value="Skipped"/>
    <property object="P0.1" propertyId="ports.settings.skip" value="Skipped"/>
    cproperty object="P0.2" propertyId="ports.settings.skip" value="Skipped"/>
    cproperty object="P0.3" propertyId="ports.settings.skip" value="Skipped"/>
   cproperty object="P1.1" propertyId="ports.settings.skip" value="Skipped"/>
    cproperty object="P1.2" propertyId="ports.settings.skip" value="Skipped"/>
    cproperty object="P1.3" propertyId="ports.settings.skip" value="Skipped"/>
```

```
<property object="P1.4" propertyId="ports.settings.iomode" value="Digital</pre>
Push-Pull Output"/>
      <property object="P1.4" propertyId="ports.settings.outputmode" value="Push-</pre>
pull"/>
      cproperty object="P1.7" propertyId="ports.settings.inputmode" value="Analog"/>
      cproperty object="P1.7" propertyId="ports.settings.iomode" value="Analog
I/0"/>
      cproperty object="P1.7" propertyId="ports.settings.skip" value="Skipped"/>
      <property object="PBCFG 0" propertyId="pbcfg.settings.enablecrossbar"</pre>
value="Enabled"/>
      <property object="PCACH_0" propertyId="ABPeripheral.included" value="true"/>
      cproperty object="PCACH 0"
propertyId="pcach.pcachannelcapturecomparemode.enablechannelcomparatorfunction"
value="Enabled"/>
      cproperty object="PCACH 0"
\verb|propertyId="pcach.pcachannelcapturecomparemode.enable channel pulse width \verb|modulation|| modulation modula
den" value="Enabled"/>
      cproperty object="PCACH_0"
propertyId="pcach.pcacontrol.channelcapturecomparemode" value="Predefined 8~11-bit
pulse modulator"/>
      <property object="PCACH_1" propertyId="ABPeripheral.included" value="true"/>
      cproperty object="PCACH_2" propertyId="ABPeripheral.included" value="true"/>
      cproperty object="PCA_0" propertyId="ABPeripheral.included" value="true"/>
      cproperty object="PCA_0"
propertyId="pca.pcacountertimerconfiguration.pcaclockfrequency" value="255.208
kHz"/>
      cproperty object="PCA 0"
propertyId="pca.pcacountertimerconfiguration.pcaclockfrequencyintegervalue"
value="255208.333333"/>
      cproperty object="PCA_0"
propertyId="pca.pcacountertimerconfiguration.pcaclockperiod" value="3.918 uS"/>
      <property object="PCA_0" propertyId="pca.pcacountertimerruncontrol.pcarun"</pre>
value="Start"/>
      <property object="TIMER01_0" propertyId="ABPeripheral.included" value="true"/>
      cproperty object="TIMER16_2" propertyId="timer16.control.runcontrol"
value="Start"/>
      value="Timer is Running"/>
      cproperty object="TIMER16 2"
propertyId="timer16.initandreloadvalue.targetoverflowfrequency" value="4"/>
      cproperty object="TIMER16 2"
propertyId="timer16.initandreloadvalue.timerreloadvalue" value="1734"/>
      cproperty object="TIMER16 2"
propertyId="timer16.reloadhighbyte.reloadhighbyte" value="6"/>
      <property object="TIMER16_2" propertyId="timer16.reloadlowbyte.reloadlowbyte"</pre>
value="198"/>
      cproperty object="TIMER16_3" propertyId="ABPeripheral.included" value="true"/>
      cproperty object="TIMER_SETUP_0" propertyId="ABPeripheral.included"
value="true"/>
      cproperty object="VREF_0" propertyId="ABPeripheral.included" value="true"/>
      cproperty object="VREF 0" propertyId="vref.hidden.voltagereferenceselect"
value="VDD pin"/>
      cproperty object="VREF 0"
propertyId="vref.voltagereferencecontrol.selectvoltagereference"
value="Unregulated VDD"/>
      <property object="WDT_0" propertyId="ABPeripheral.included" value="true"/>
      cproperty object="WDT_0" propertyId="wdt.watchdogcontrol.wdtenable"
value="Disable"/>
```

```
</mode>
 <modeTransition>
   cproperty object="RESET → DefaultMode"
propertyId="modeTransition.source" value="RESET"/>
   cproperty object="RESET → DefaultMode"
propertyId="modeTransition.target" value="DefaultMode"/>
 </modeTransition>
</device:XMLDevice>
Main.c:
//----
// src/feladat11-03_main.c: generated by Hardware Configurator
// This file will be updated when saving a document.
// leave the sections inside the "\{[\ldots]" comment tags alone
// or they will be overwritten!!
//-----
         _____
// Includes
//-----
#include <SI_EFM8BB1_Register_Enums.h>
                                     // SFR declarations
#include "InitDevice.h"
// $[Generated Includes]
// [Generated Includes]$
//-----
// SiLabs_Startup() Routine
// -----
// This function is called immediately after reset, before the initialization
// code is run in SILABS_STARTUP.A51 (which runs before main() ). This is a
// useful place to disable the watchdog timer, which is enable by default
// and may trigger before main() in some instances.
//-----
void SiLabs_Startup (void)
 // $[SiLabs Startup]
 // [SiLabs Startup]$
//-----
// main() Routine
int main (void)
 // Call hardware initialization routine
 enter DefaultMode from RESET();
 while (1)
   // $[Generated Run-time code]
   // [Generated Run-time code]$
Interrupts.c:
// src/Interrupts.c: generated by Hardware Configurator
//
```

Az elkészült programot be kell mutatni!

A gyakorlatvezető ellenőrizte:

- <u>Igen</u>
- Nem

A program működött:

- <u>Igen</u>
- Nem

## Megjegyzések

A második feladatban a config elvileg helyes, mégis azt tapasztaltuk, hogy nem azt tapasztaltuk, amit akartunk.