Interrupt

MicLab-02

Név:

Dátum:

Mérőhely:

# Bevezetés

Az interrupt használatának megismerése.

## Ajánlott irodalom

<http://www.inf.u-szeged.hu/noise/Education/MicLab/>

## 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.03.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 02.

# 1. feladat – LED vezérlése nyomógombbal

Írjon egy programot, mely a LED0-át bekapcsolja, ha a BTN0 le van nyomva és kikapcsolja, ha a BTN0 fel van engedve.

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

Az elkészült programot be kell mutatni!

A gyakorlatvezető ellenőrizte:

* Igen
* Nem

A program működött:

* Igen
* Nem

# 2. feladat – LED vezérlése interrupttal I

Írjon egy programot, mely a LED0-át 10 Hz frekvenciával villogtatja, a vezérlési feladatot a Timer 2 interrupt rutin lássa el.

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

Az elkészült programot be kell mutatni!

A gyakorlatvezető ellenőrizte:

* Igen
* Nem

A program működött:

* Igen
* Nem

# 3. feladat – LED állapotának billentése nyomógombbal

Írjon egy programot, mely a LED0-át átbillenti az aktuális állapotának az ellenkezőjére, ha megnyomjuk a BTN0-át. A nyomógomb lenyomásakor a véletlenszerű, ún. „pergése” okozta többszöri detektálás elkerüléséhez pergésmentesíteni is kell a nyomógombot.

A megvalósítás részletei:

* A nyomógombot a Timer 2 használatával pergésmentesítse, mely 10 ms-os periódusidővel generál egy interruptot és léptet egy változót (extern uint8\_t legyen), amivel az időt mérhetünk.
* A nyomógomb állapotát a főprogramban ellenőrizze. Amennyiben a gomb 50 ms-ig lenyomott állapotban van, úgy minősítse lenyomottnak és billentse át a LED0 állapotát az ellenkezőjére!

Tippek a pergésmentesítés algoritmusához: Ha a gombot lenyomtuk, akkor a program várakozzon 50 ms-ig (tartsa fel a főprogramot a megfelelő C vezérlési szerkezettel), billentse át a LED0-át, majd ismét várakozzon, egészen addig, amíg a gomb lenyomott állapotban van (ezzel el lehet kerülni a LED0 folyamatos billegtetését, ha a gomb sokáig van lenyomva). Ezután ismét várjon 50 ms-ot, hogy a gomb felengedése is pergésmentesítve legyen.

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

Az elkészült programot be kell mutatni!

A gyakorlatvezető ellenőrizte:

* Igen
* Nem

A program működött:

* Igen
* Nem

# Szorgalmi feladat – LED vezérlése interrupttal III

Írjon egy programot, mely a LED0-ás LED-et 5 Hz, 20 Hz és 100 Hz frekvenciával villogtatja. A vezérlési feladatot a Timer 0 interrupt rutin lássa el, a villogási frekvenciák között a BTN0-ás nyomógomb segítségével lehessen váltani. A nyomógombot pergés mentesíteni is szükséges.

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

Az elkészült programot be kell mutatni!

A gyakorlatvezető ellenőrizte:

* Igen
* Nem

A program működött:

* Igen
* Nem

# Megjegyzések