# Mérési jegyzőkönyv a

3.

laboratóriumi gyakorlatról

**A mérés helyszíne:** QBP110

**A mérés időpontja:** 2016. *március 30.*

**A mérést végezték:** Herjavecz Zoltán, SDE19 mérőcsoport

**A mérést vezető oktató neve:** Széll András

sde19\_3.docx

## Az elvégzett mérési feladatok

# Sávlezárást jelző villanó fénysor

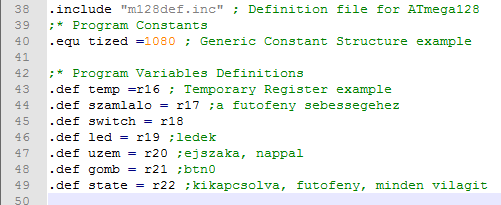
A fénysort a LED-oszlopok jelentik, egy oszlopban mindig csak egy LED világít, így fusson a fény végig rajtuk.

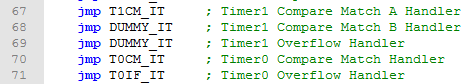
* a léptetés sebessége a SW0…SW3 kapcsolók állásától függően történjen 100 ms időalappal 15 lépésben (100ms…1,5 sec). A kapcsolókat „normál” sorrendben, sorrendhelyesen kezelje (SW0 = LSB).
* a világítás fényerejét az optikai bemenet függvényében fél fényerőre visszaveszi (éjszakai üzem). Nappal a két egymás melletti oszlop egyformán világít (tehát a LED-ek párosával gyulladnak ki és alszanak el), éjszakai üzem során csak a baloldali LED soron futnak végig a fények.
* a készüléket a BT0 nyomógomb segítségével lehet ki-bekapcsolni, három üzemmódja van: kikapcsolva, futófény, minden LED világít

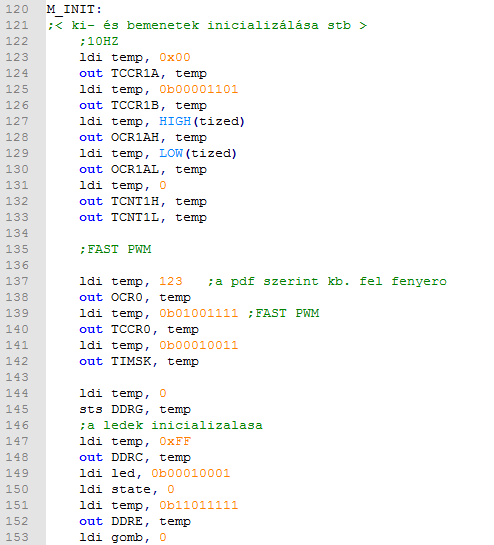
# Megoldás

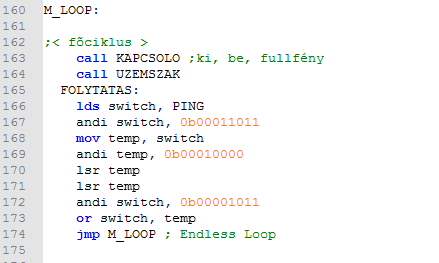
* A feladat első pontját úgy oldottam meg, ahogy az ml2\_avr jegyzetben le van írva a 16 bites Timer/counter1 perifériával, ezzel előállítottam egy 100 ms-os alapot, amin szoftveres időzítővel számolok a switchek miatt
* a világítás fényerejétől függően változik, hogy mennyi LED világít vagy sem. A fél fényerőt szintén a jegyzet alapján készítettem el, egy PWM jelet előállítva a Timer/Counter0 periférivál.
* A btn-nel történő ki/bekapcsolást a main loopban, interruptok használata nélkül implementáltam.

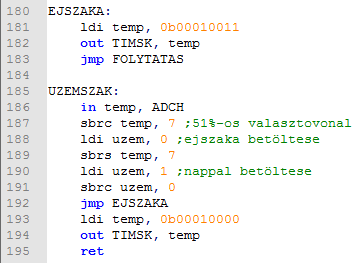
# Forráskód (annak magyarázata)

Létrehoztam egy konstanst, hogy amivel 10 Hz-re osztom le az órajelet, ezzel segítve a 0,1 sec-undumonként befutó IT-t, a többi regiszterért a kommentek beszélnek.

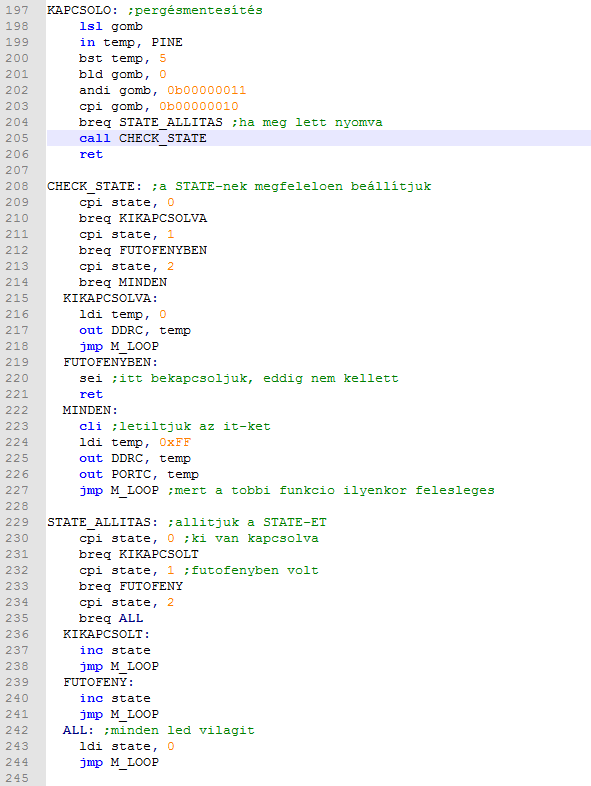
Bejegyeztem az IT szubrutinokat a megszakítási vektortáblába.

Először a jegyzetnek megfelelően konfigurálom a Timer0 perifériát, aztán a Fast PWM jelhez (amitől várhatóan fele olyan erősen világít a LED), aztán beállítom az I/O-kat, illetve inicializálom a regisztereket. A globális interruptot nem itt, hanem máshol engedélyezem, hogy a kikapcsolt illetve folyamatosan világító állapot tökéletesen működjön, ne fussanak közben be interruptok.

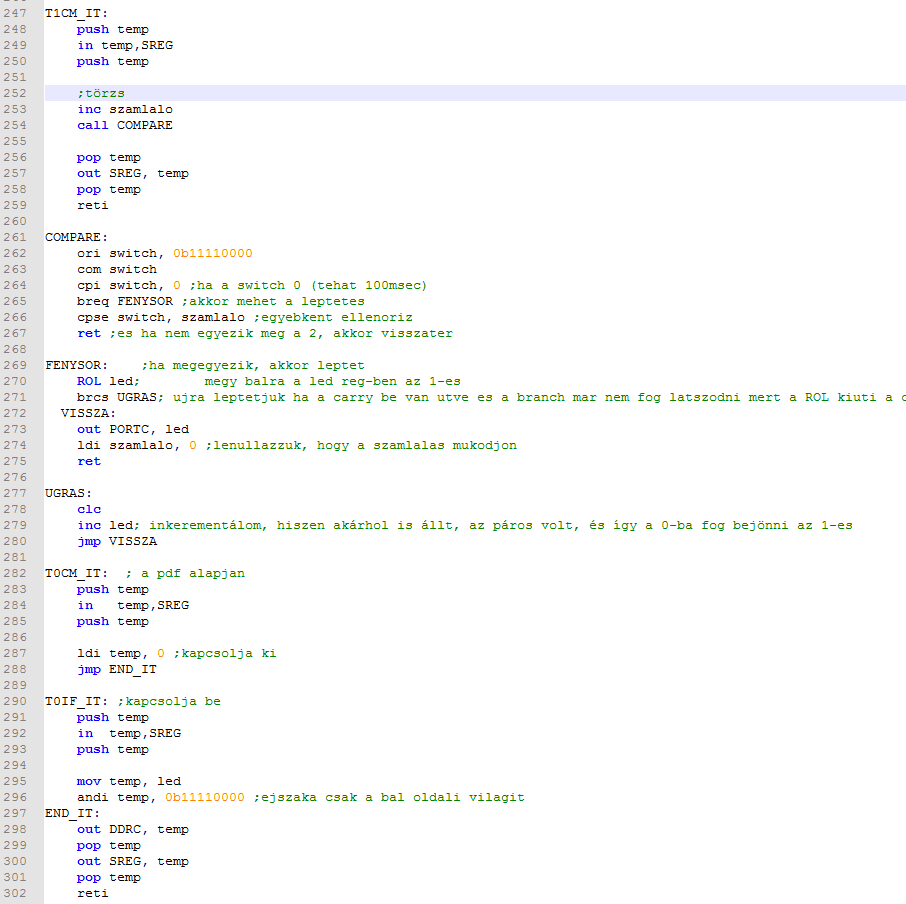
Mint látható a M\_LOOP-ból összesen két szubrutint hívok meg közvetlenül, ezeket ott elmagyarázom, illetve itt kezelem le a switchek beolvasását, megfelelő helyre helyezését.



Itt olvasom be a fényérzékelő állapotát, illetve töltöm be a regiszterbe, hogy éppen nappal vagy éjszaka van-e. Ha éjszaka van, akkor bekapcsolom a PWM megszakításait is, hogy a led gyengébben világítson, ha pedig nappal van, akkor kikapcsolom azt.

Itt pergésmentesítem a gombot első lépésben, kapcsolónak neveztem el a szubrutint, mert ez kapcsolja ki/be a programot kvázi. Ha meg lett nyomva a gomb, akkor beállítom a state regisztert, jelezve hogy most milyen funkciót kell produkáljon a program. Ez lefut minden egyes gombnyomás érzékelése után.

A CHECK\_STATE-ben konfigurálom be a programot a state regiszter alapján, hogy az adott funkciónak megfelelően működjön. A KIKAPCSOLVA illetve MINDEN címkék után azért ugrok rögtön vissza az M\_LOOP-ra, ugyanis akkor teljesen felesleges minden mást olvasni/állítani, ugyanis nem függ semmitől. A futófényben pedig engedélyezem a globális interruptokat, hogy az a funkció működjön, hiszen azt interruptok, hardver időzítő állítja elő.

A T1CM\_IT interrupt fut le 100 msec-onként, itt növelem a szoftver számlálót, és nullázom a switchek állásától függően, hiszen a switchek állásáig számol csak el. A switchekről beolvasott értékek komplementerét veszem, hiszen azok fordítva „működnek”, és a számlálós összehasonlítás miatt jobb így. A többihez pedig a kommentek bőségesen elegek. A következő ami érdekes lehet, az a Timer0 pedig a PWM jelet állítja elő, amivel a ledek fényereje lesz gyengébb. A bekapcsolásnál beletettem egy ANDI-t, hogy csak a bal oldali fénysor világítson, más érzésem szerint nem szorul magyarázatra.