

# RGB ledsor vezérlése

## Témalaboratórium jegyzőkönyv

2017/18 I. félév

Készítette: Seres Zsófia

Konzulens: Kovácsházy Tamás

# Tartalomjegyzék

<b>Tartalomjegyzék</b>	<b>1</b>
<b>Specifikáció</b>	<b>2</b>
A feladat	2
Az elvégzendő feladatrészek	2
<b>A kapcsolási rajz</b>	<b>3</b>
A bemenetek védelme	3
Az áram mérése	4
A feszültség mérése	5
I2C kommunikáció	5
Tápellátás	6
WS2812B RGB LED	6
<b>A nyomtatott áramkör</b>	<b>7</b>
<b>A szoftver</b>	<b>9</b>
<b>Felhasznált segédanyagok</b>	<b>9</b>

# Specifikáció

## A feladat

STK3700 Giant Gecko mikrokontrollerrel RGB ledek vezérlése és egyéb áramköri elemek működtetése mint például árammérő, feszültségmérő, hőmérsékletmérő, fénymérő.

## Az elvégzendő feladatrészek

- Alkatrész kiválasztása.
- Kapcsolás tervezése a specifikáció és az alkatrész alapján.
- Nyomtatott áramkör megtervezése.
- Szoftver írása a Simplicity Stúdió fejlesztőkörnyezetben a Geckora.

## A kapcsolás

### A bemenetek védelme

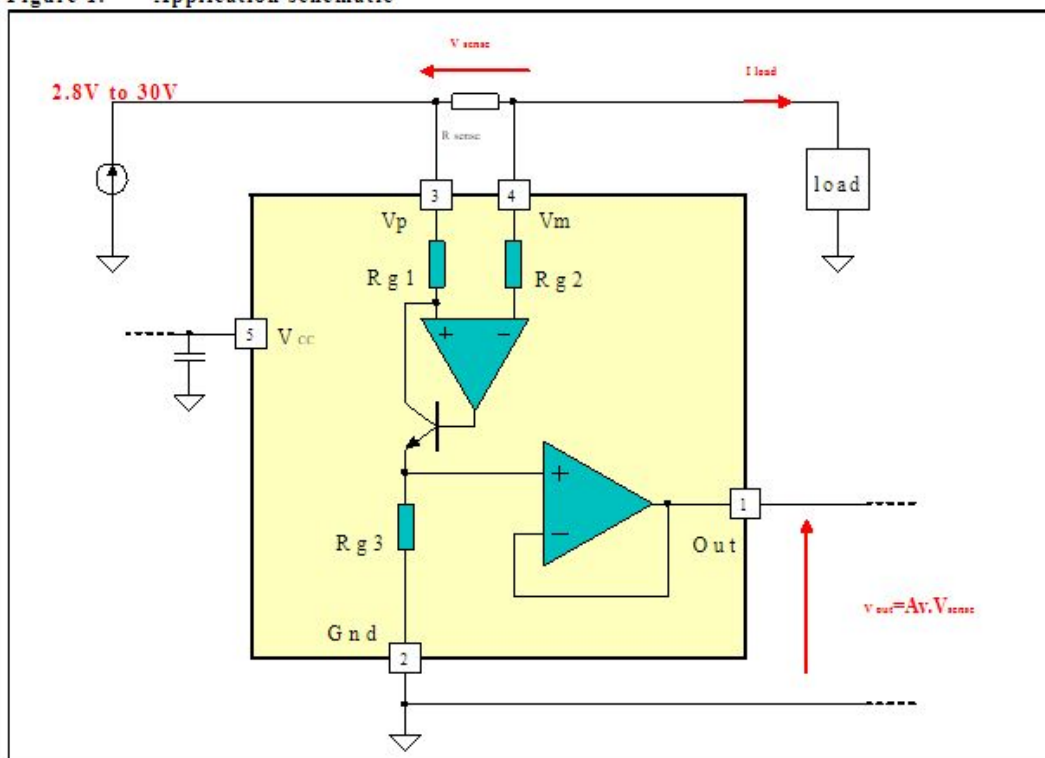
A felhasznált IC-k bemenetét fontos megszűrni, így minden bemenettel párhuzamosan egy 100nF-os értékű kondenzátort kötöttem illetve a LED-hez még külön egy 10uF-ost is.

A feszültségosztó kimenetére egy Zener diódát terveztem, ezzel védve a Gecko bemenetét az esetleges túlfeszültségtől. Ennek értéke 5V, mert a Geckó ennél több feszültséget nem bír el a bemenetén.

## Az áram mérése

Az áram mérésére egy “High side current sense amplifier”-t választottam. Ennek működése:

Figure 1. Application schematic

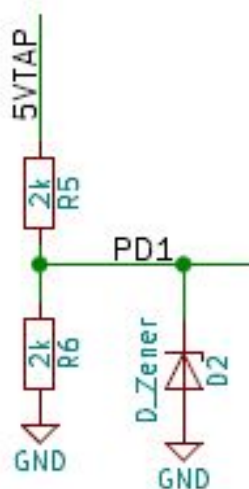


TSC101 árammérőkből van több fajta erősítési tényezőjű, én az 50-es erősítésűt használtam. Sönt ellenállást aszerint kellett választani, hogy ne essen rajta nagy feszültség hiszen a LED bemenetére közel 5V kell, hogy kerüljön. Illetve fontos, hogy a Gecko AD bemenetére ne kerüljön 5V-nál nagyobb feszültség. A kapcsolást 1A-es áram felvételre méreteztem.

A megválasztott  $0.1\ \Omega$  értékű sönt ellenálláson eső feszültséget méri az IC majd a fentebb látható kapcsolat segítségével felerősíti 50-szeresére és kiadja a jelet kimenetére, ami egy AD konverzióval kiolvasható a Gecko segítségével.

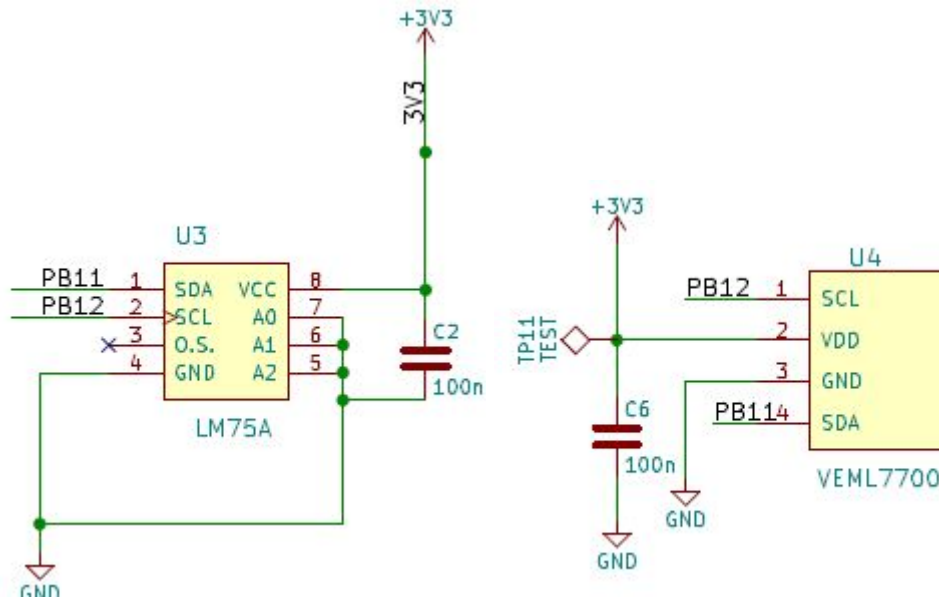
## A feszültség mérése

A feszültség mérésére egy feszültségosztót használok, melyben az ellenállások értéke  $2\ k\Omega$ . Ez is szintén egy AD lábra van rákötve.

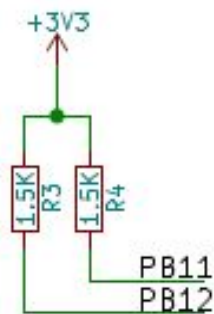


## I2C kommunikáció

A hőmérsékletérzékelő és a fénymérő I2C busz segítségével kommunikál a Geckoval.



Mind a két szenzor a PB11 és PB12-es lábakra van rákötve egy-egy 1.5 k $\Omega$ -os felhúzó ellenállással a következő módon:

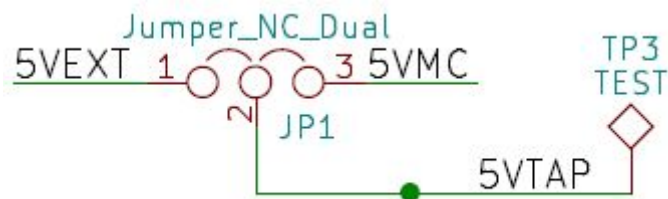


Mind a két szenzor a Gecko 1-es I2C-jét használja.  
A fénymérő kiválasztásánál fontos szempont volt a fizikai mérete, a kézi beforrasztás miatt.

## Tápellátás

A nyákon jumperrel lehet kiválasztani, hogy melyik tápellátást szeretnénk használni. A Gecko 5V-ja és egy külső táp közül. A külső tápot azért terveztem a nyákra mert ez teszi lehetővé azt, hogy több (~>10) LED-et is tudjak működtetni, mert a Gecko 5V-ja kevés ehhez.

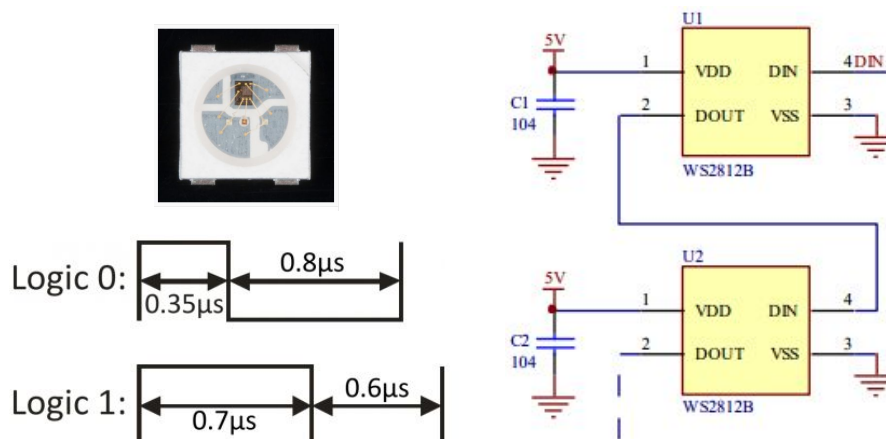
A jumper:



A jumpert 3 tűskével oldottam meg a nyákon. A középső tűske a kiválasztott tápot vezeti tovább. A nyákon azt a szélső tűskét kell kiválasztanunk amelyik tápforrást használni szeretnénk. A felső tűske a külső tápforrásé, az alsó a Gecko 5V-os tápjáé.



## WS2812B RGB LED



A feladathoz a WS2812B típusú LED-et használtam. Ez bal felső képen látható. 4 lába van: táp, föld, DataIn és DataOut.

1 db LED található a nyákon is, a többbit egy sorkapoccsal lehet a nyákhoz kötni. A jobb oldali képen látható, hogy hogyan lehet a LED-eket kaszkádosítani. Az első LED Dataout kimenetét kell a következő Datain bemenetére kötni.

A Gecko PC4-es lábáról kapja a PWM jelet. A bal alsó képen látható, hogy milyen PWM jelet szükséges beállítani a logikai 0 és 1-es értékekhez. A LED-ek 24 biten kapják az adatot ebből 8 bit R 8 bit G és 8 bit B.

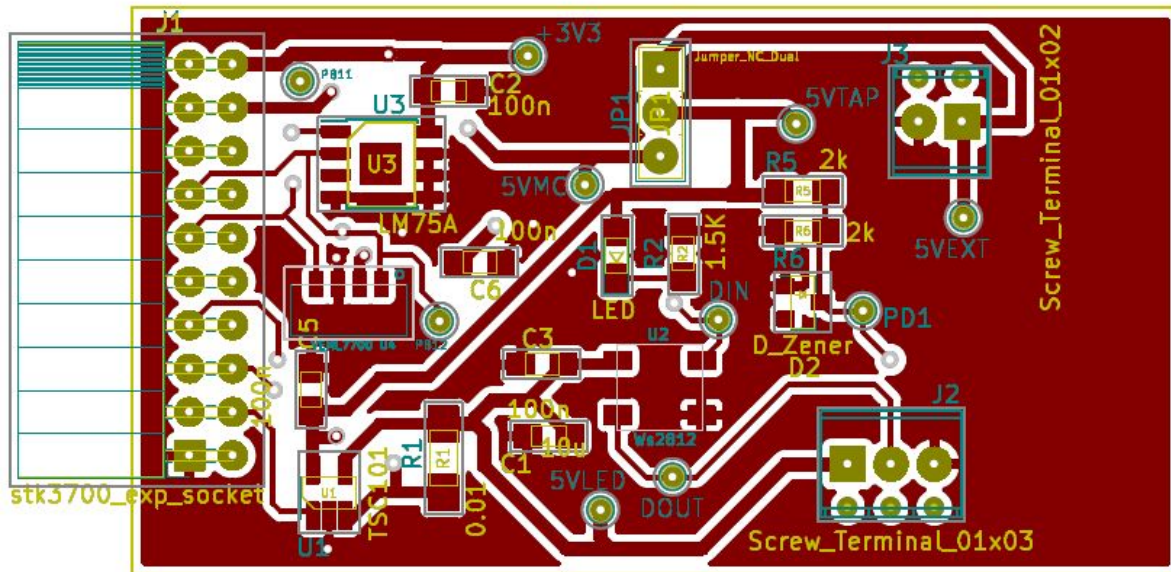
## A nyomtatott áramkör

A tervezéshez a kiCad programot használtam, amit órai keretek között tanultunk meg használni. Az elrendezésnél figyeltem arra, hogy a bemenet védelmére szolgáló kondenzátorok fizikailag is közel helyezkedjenek a bemenetekhez és ne zavarjanak be parazita tényezők. Nagyjából az összetartozó részeket is próbáltam egymáshoz közel helyezni pl.: az I2C-s elemeket egymás mellé, illetve a sorkapcsokat és a jumpert a nyák szélére helyeztem.

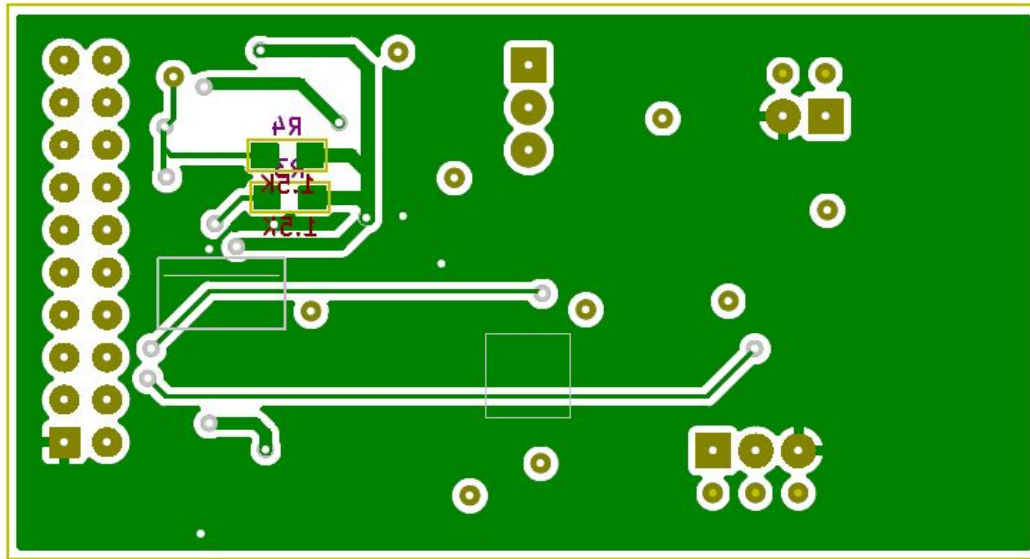


A nyomtatott áramkör terve a kiCad tervezőfelületén:

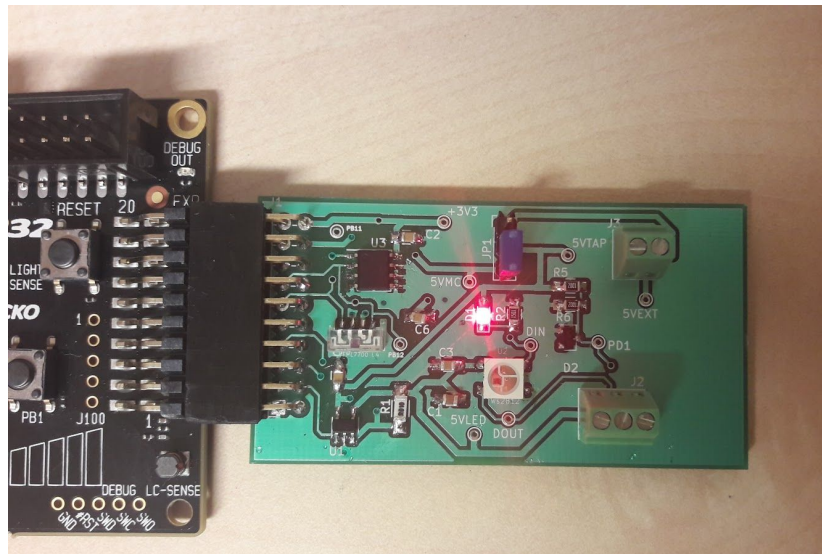
A felül lévő oldal:



Az alja:



Az alkatrészek beültetése után a nyomtatott áramkör így néz ki:



## A szoftver

A Gecko felprogramozásához a Simplicity Stúdió példa projektjeit használtam. Elsőként az I2C hőmérőt szerettem volna működtetni. Ez a DVK-ra lett kifejlesztve nem az STK-ra, így megpróbáltam módosítani, hogy le tudjon fordulni a Geckora is.

A módosítások után át kellett adni az LM75A címét a programnak amit az adatlapból ki kellett keresni, ez a 48h-nak adódott, mivel az A0,A1,A2 lábak határozzák meg a cím alsó 3 bitjét és ezek földre vannak kötve.

Beállítottam a példakódban az I2C1-es perifériát is az I2C0 helyett.

A kód lefordítása után a Gecko nem tudta kiolvasni a hőmérő regiszteréből az értéket így "error"-t írt ki az LCD kijelzőre.

Ez után Picoscope-ot használva próbáltam megnézni, hogy hardveres hiba lehetséges e. A felhúzó ellenállásoknál mértem a jelet amit a Gecko kiadott, brakepoint-ot használva. A kimeneten bemérhető volt a 3.3V és a 0V állapot is.

A LED működtetéséhez a PWM jelet kell előállítani GPIO lábra. A 0 jelszint előállításához 1,25 us/3 ideig szükséges logikai magas szintet kiadni. Először for ciklussal próbáltam előállítani majd a Delay() függvénnyel, de egyikkel sem sikerült ilyen magas frekvenciát elérni még.

## Felhasznált segédanyagok

- [WS2812B](#)
- [TSC101](#)
- [VEML7700](#)
- [LM75A](#)
- [Gecko user guide](#)
- [Gecko reference manual](#)
- [Simplicity Stúdió mintakód](#)

## Melléklet:

- Kapcsolási rajz

