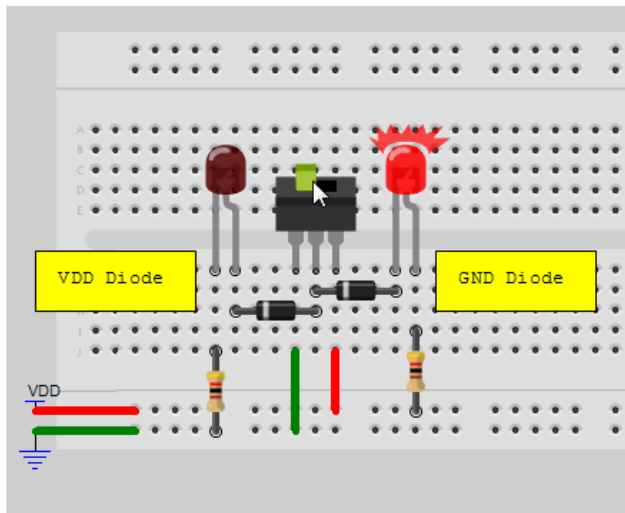
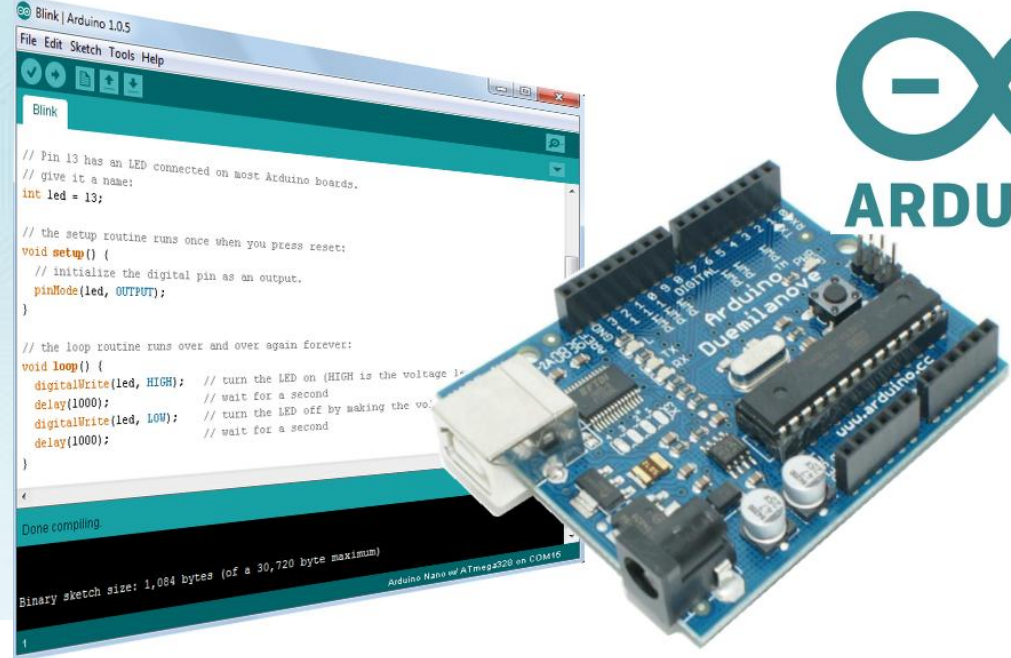


Hobbi Elektronika 2015/2016



1. Témakörök, célkitűzés, hozzávalók

Választható témakörök

Bevezetés az elektronikába

Alapfogalmak és összefüggések, tranzisztoros áramkörök építése dugaszolós próbapanelon, billenőáramkörök NE555-tel , tápáramkörök, logikai áramkörök.

Bevezetés az informatikába

Számrendszerek, Boole-algebra, igazságtáblázat, logikai kifejezések kiértékelése, ismerkedés a C programnyelvvvel, bitműveletek.

Mikrovezérlők programozása

ATmega328 (Arduino) programozása egyszerű programozói környezetben (Arduino, Wiring++).

Nagyteljesítményű (32 bites) mikrovezérlők programozása

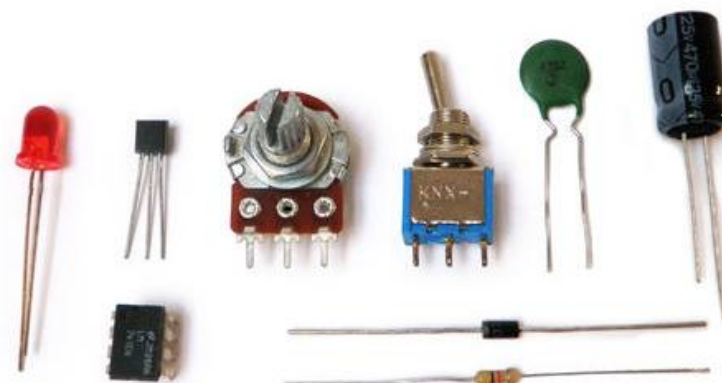
ARM Cortex-M0 mikrovezérlők programozása (mbed vagy C/C++)

IoT eszközök, kommunikáció, távvezérlés

ESP8266 WiFi eszköz, OpenWRT router, Androidos mobiltelefon stb.

1. Bevezetés az elektronikába

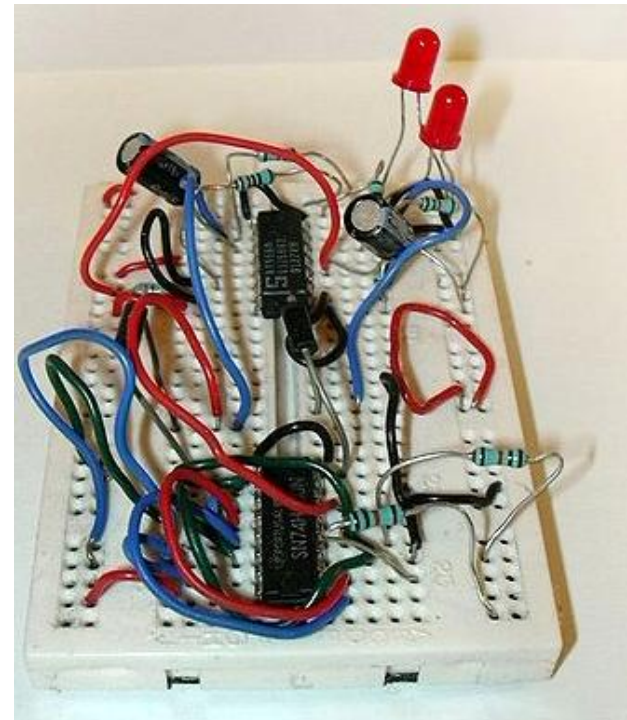
- **Alapfogalmak és a legegyszerűbb összefüggések tisztázása** (pl. Ohm törvény).
- **Egyszerű áramkörök építése és kipróbálása** (ellenállások, kondenzátorok, nyomógombok, kapcsolók, LED-ek, PNP és NPN tranzisztorok felhasználásával, dugaszolós próbapanelon)
- **Billenőáramkörök** (astabil, bistabil, monostabil áramkörök) ismertetése és felhasználása (pl. LED-es villogó, időzítők)
- **Egyszerű tápáramkörök** (feszültségstabilizátor, áramgenerátor) működésének ismertetése és méretezése.
- **Logikai kapuáramkörök** (ÉS, VAGY, NEM, stb.) ismertetése, egyszerű logikai áramkörök megépítése és vizsgálata.
- **Forrasztási gyakorlat**



Fiatalok is elkezdhetik...

Mire lesz szükségünk?

- ❑ **Dugaszolós próbapanel**
- ❑ **Alkatrészek** (ellenállás, kondenzátor, LED, tranzisztor, elemtartó vagy dugasztáp)
- ❑ **Vezeték** (kb. 0.6 mm átmérőjű merev rézhuzal, vagy készen kapható „jumper cable” készlet)
- ❑ **Integrált áramkörök** (bonyolultabb kapcsolások kiváltására)



Beszerzési források

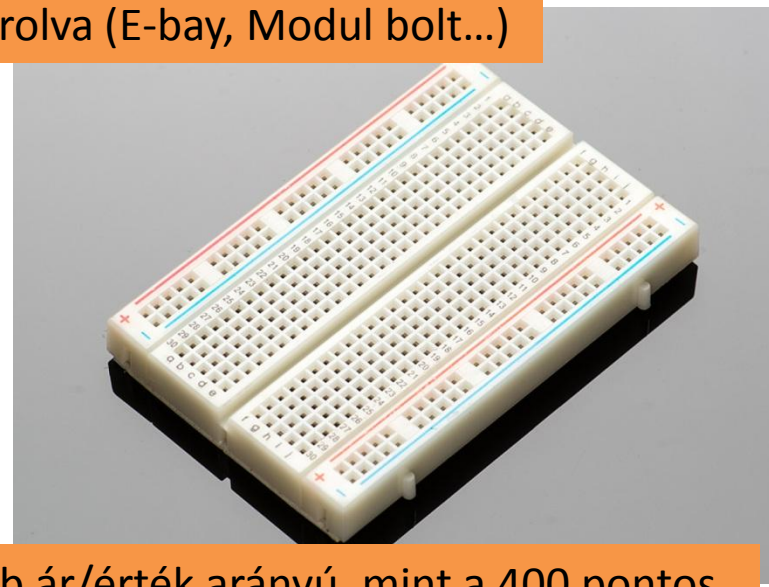
- Bontásból/adományokból (lásd: megtestesules.info/hobbielektronika/cuccok.html)
- Hobbi Elektronika üzlet (Kossuth u.) vagy a volt Elektromodul bolt (Sámsoni út)
- Hestore.hu, Ebay.com, Aliexpress.com vagy más webáruház



Conrad Basic 3964 építőkészlet



Olcóbb megoldás: Alkatrészként összevásárolva (E-bay, Modul bolt...)



A 830 pontos panel jobb ár/érték arányú, mint a 400 pontos.

Az ideális műhely eszköztára (mi azért ennyire nem merülünk bele!)



- A forrasztás mellőzhető dugaszpanel használatával. Csak akkor kell, ha valamit végleges formában is meg akarunk építeni...
- A fogók is mellőzhetők
- Feszültségmérő csak akkor kell, ha valami nem működik (a legolcsóbb kivitelű is megfelel ~ 2000 Ft)

Alternatív lehetőség: szimuláció

Áramkörszimulációs programok

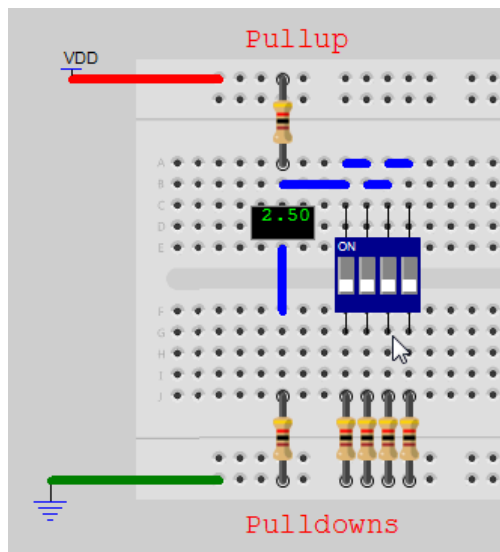
- Ami ingyenes, az többnyire korlátozott, tökéletlen
- Ami valamennyire használható, az többnyire fizetős

Virtual Breadboard: www.virtualbreadboard.com/

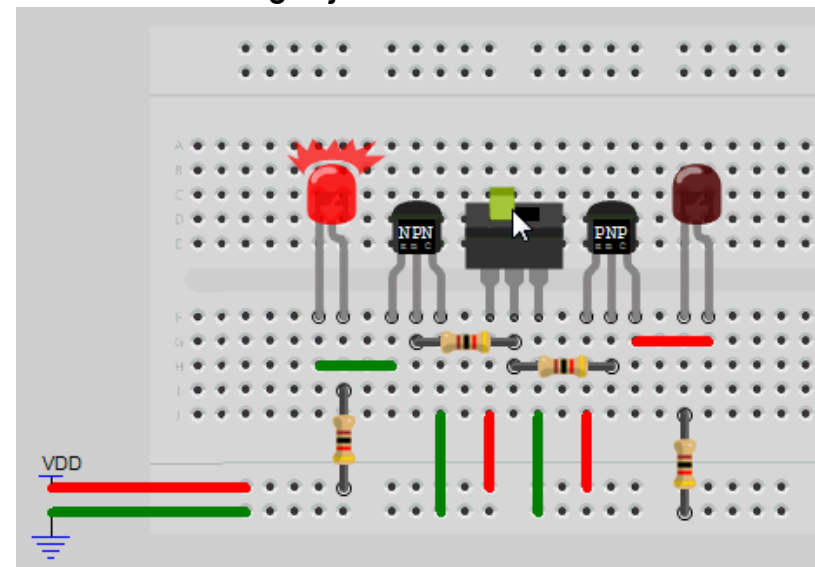
Falstad Circuit Simulator: www.falstad.com/circuit/

PHET Interactive Simulations: phet.colorado.edu/hu/simulation/circuit-construction-kit-dc

Feszültségosztó alsó tagjának változtatása



LED-ek meghajtása tranzisztorokkal



II. Bevezetés az informatikába

- Számrendszerek, átváltások, műveletek bináris, vagy hexadecimális számrendszerekben.
- Boole-algebra, igazságtáblázat, logikai kifejezések kiértékelése
- Ismerkedés a C programnyelvvvel
- Bitműveletek C nyelven megfogalmazva
- Mellékhatások és azok elkerülése a bitműveleteknél

Hozzávalók:

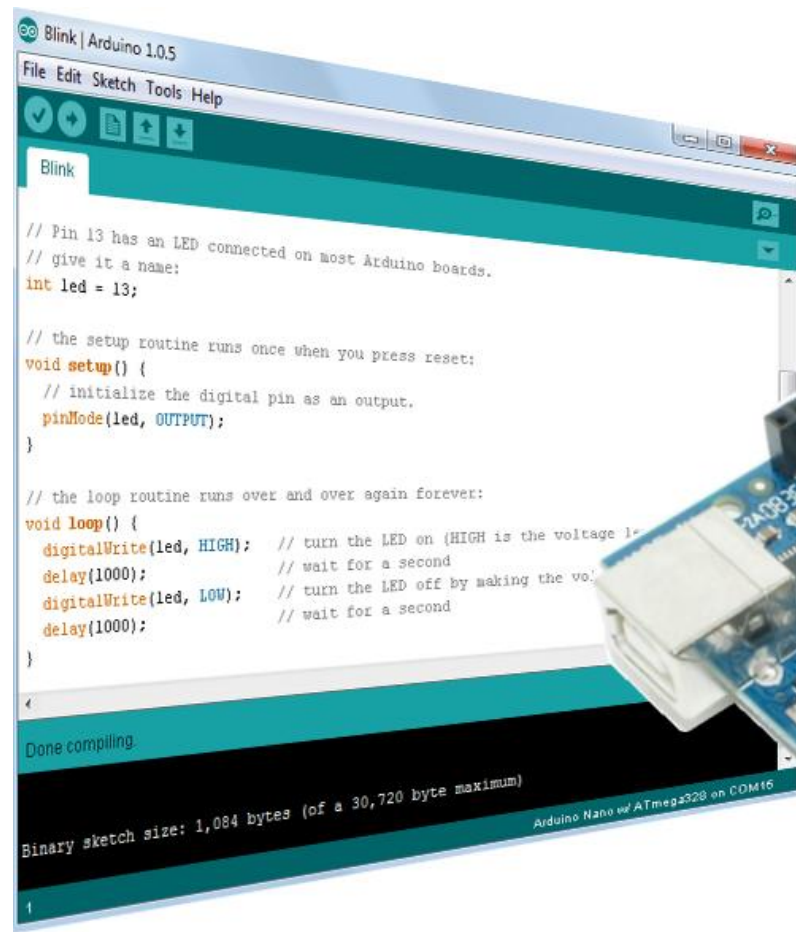
- Papír, ceruza, számítógép vagy zsebszámológép
- Szakirodalom az Interneten található

Célkitűzés:

- Alapozás a logikai áramkörök megértéséhez
- Alapozás a mikrovezérlők programozásához

III. Mikrovezérlők programozása

Kezdéshez az **Arduino** kártyát ajánlom, és az Arduino fejlesztői környezetet
A rendelkezésre álló mintaprogramok jól használhatók a részegységek megismeréséhez.
Elvárások: alapfokú elektronikai és informatikai ismeretek



```
Blink | Arduino 1.0.5
File Edit Sketch Tools Help
Blink
// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}

Done compiling.
Binary sketch size: 1,084 bytes (of a 30,720 byte maximum)
Arduino Nano w/ ATmega328 on COM16
```



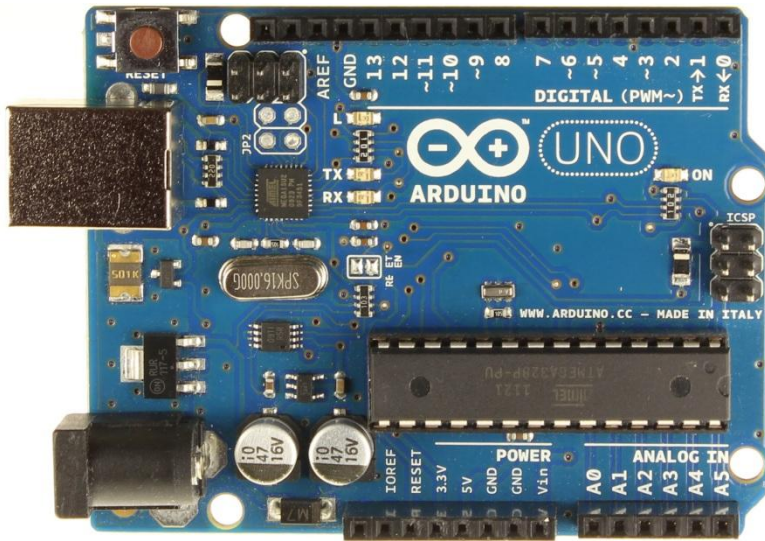
Arduino kártya

Jellemzők

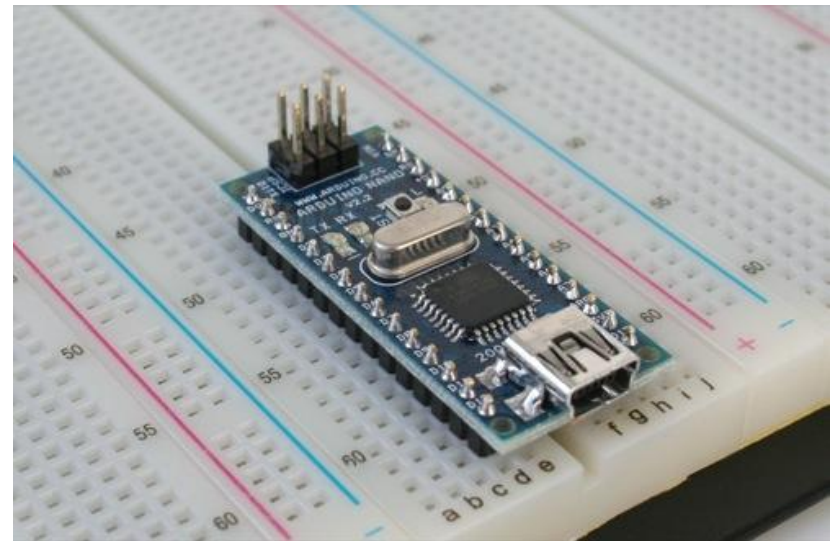
20 I/O kivezetés
32 kB flash, 2 kB RAM (ATmega328)
16 MIPS
Fejlett támogatás
Arduino IDE
5 V-os jelszintek

Perifériák

10 bites ADC
3 Timer
6 PWM csatorna
UART, SPI, I2C
A panelon USB-UART konverter
(FT232RL vagy CH340)



Arduino UNO kártya

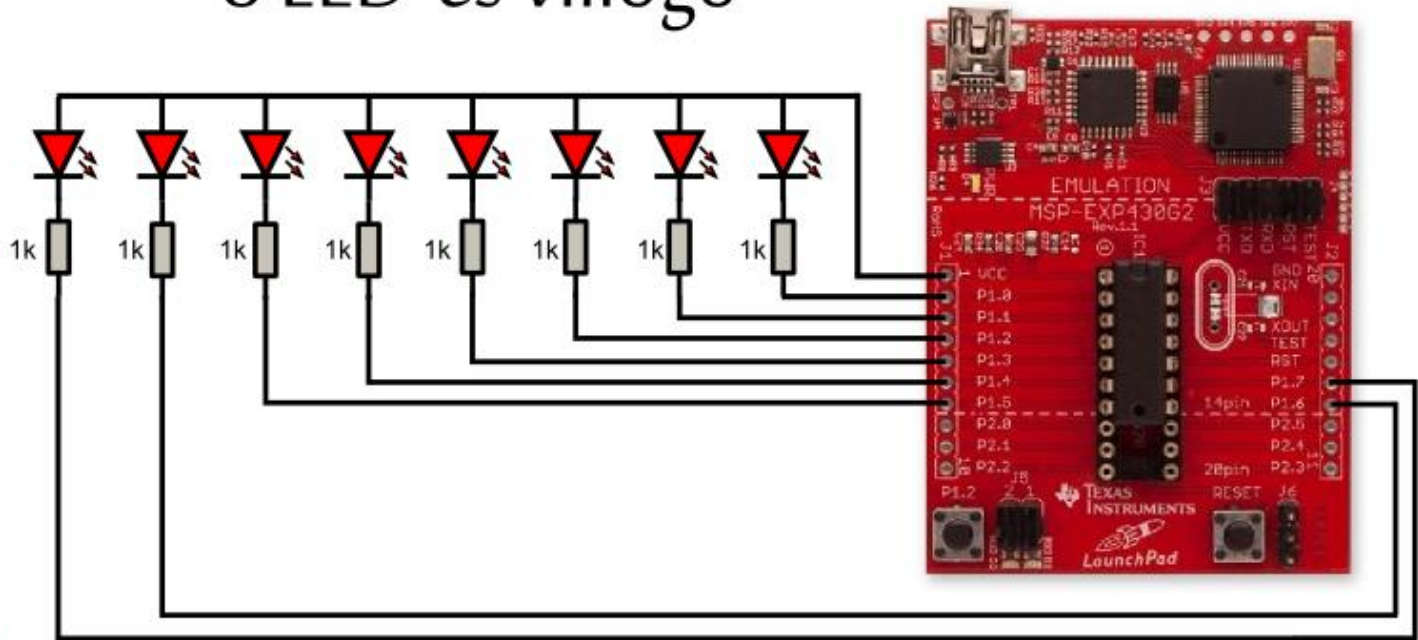


Arduino nano kártya

Néhány egyszerű, könnyen megvalósítható projekt (Arduino vagy MSP430 mikrovezérlővel)

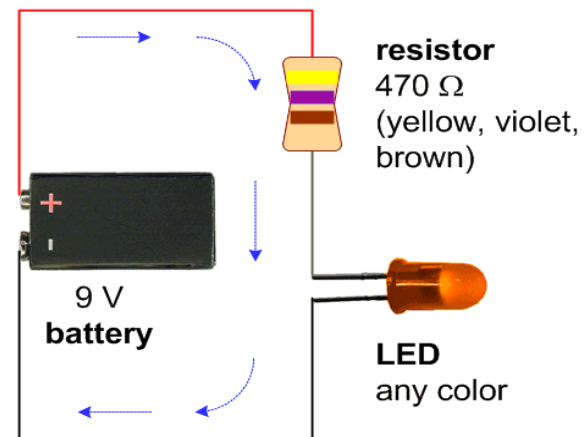
- 8 LED-es villogó (Knight Rider)
- Elektronikus dobókocka
- Ultrahangos távolságmérés
- Feszültségmérő LCD kijelzővel
- Hőmérséklet, relatív páratartalom, légnyomás mérése
- Hőmérő 7-szegmens LED kijelzővel
- Grafikus kijelzők vezérlése
- LED mátrix kijelző alkalmazás

8 LED-es villogó

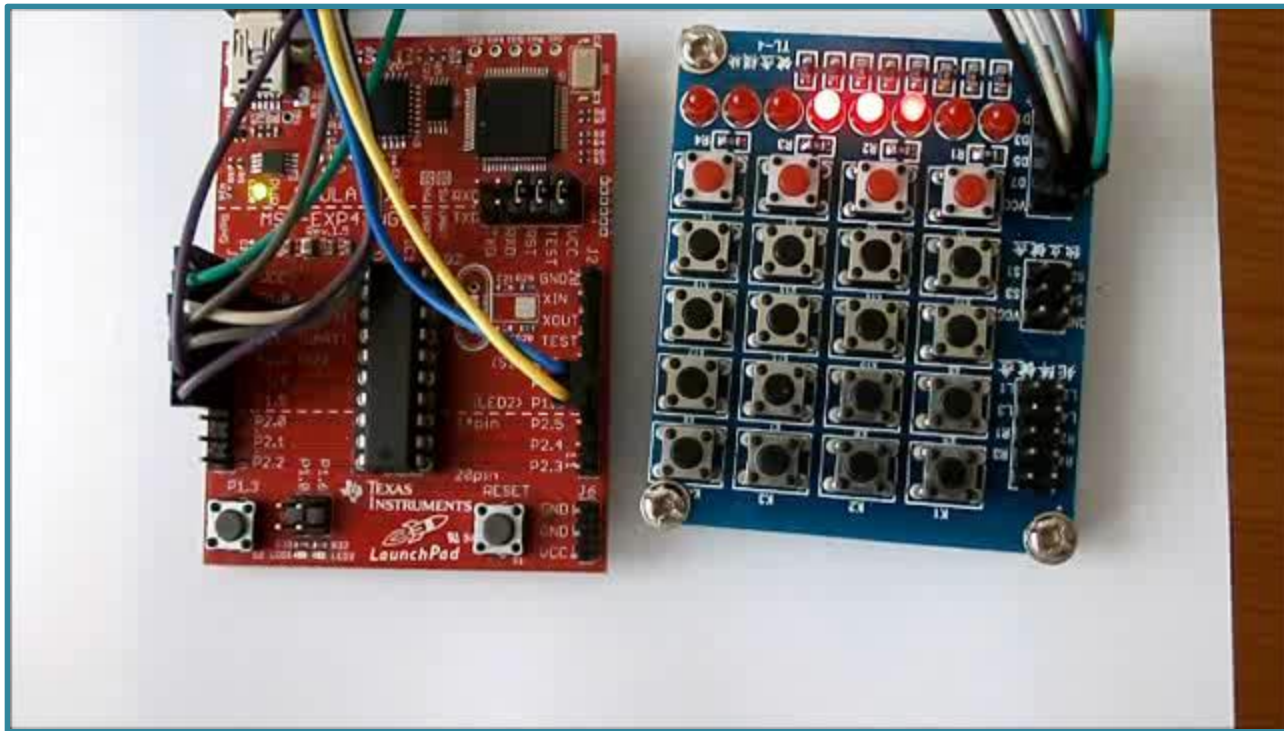


Ha a fényemittáló diódán (LED) nyitóirányú áram folyik, a dióda fényt bocsájt ki.

A fenti a kapcsolásban az MSP430 Launchpad kártya mikrovezérlője digitális kimeneteit alacsony szintre állítva kapcsolhatjuk be a LED-eket.

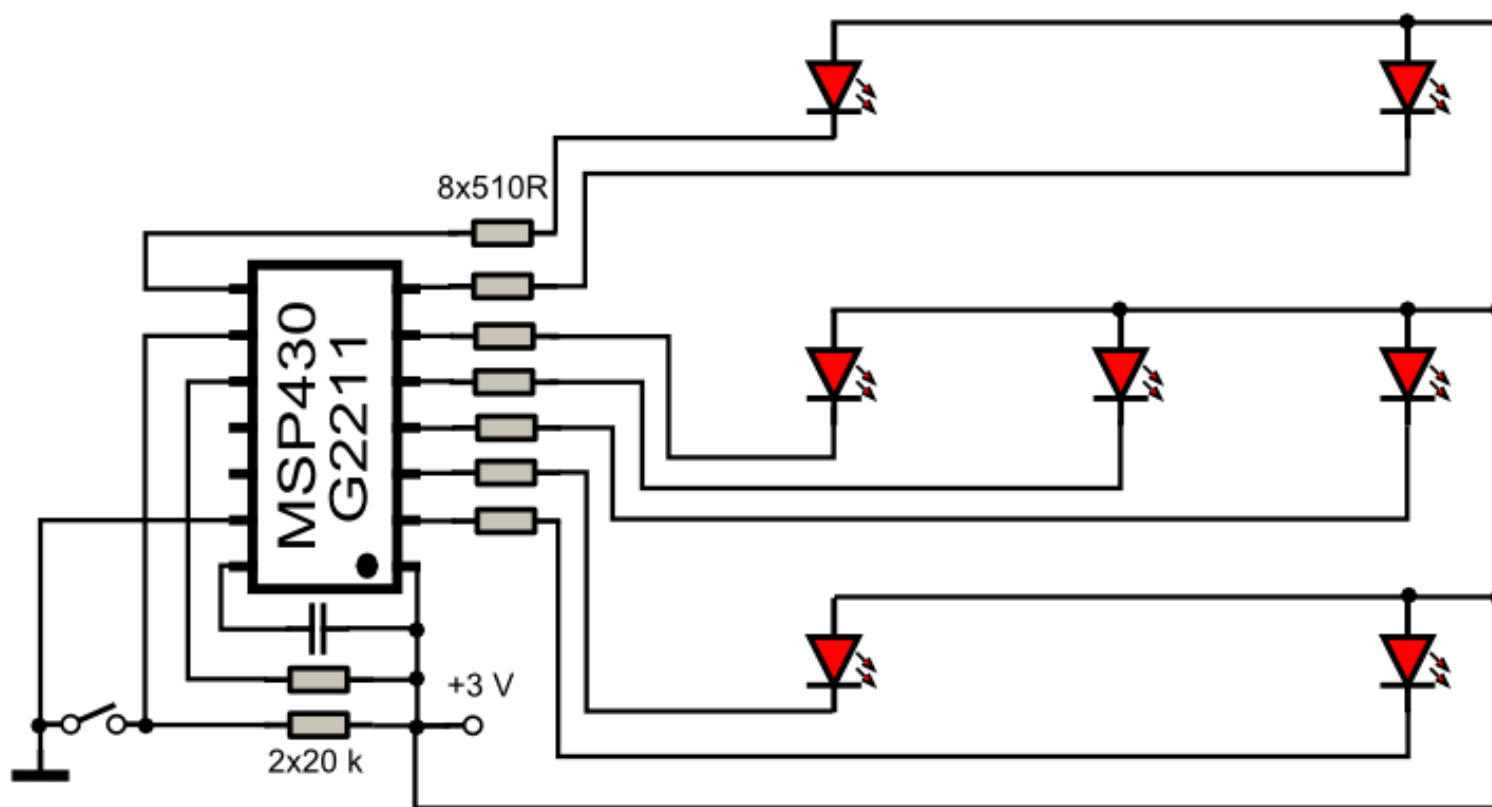


8 LED-es villogó

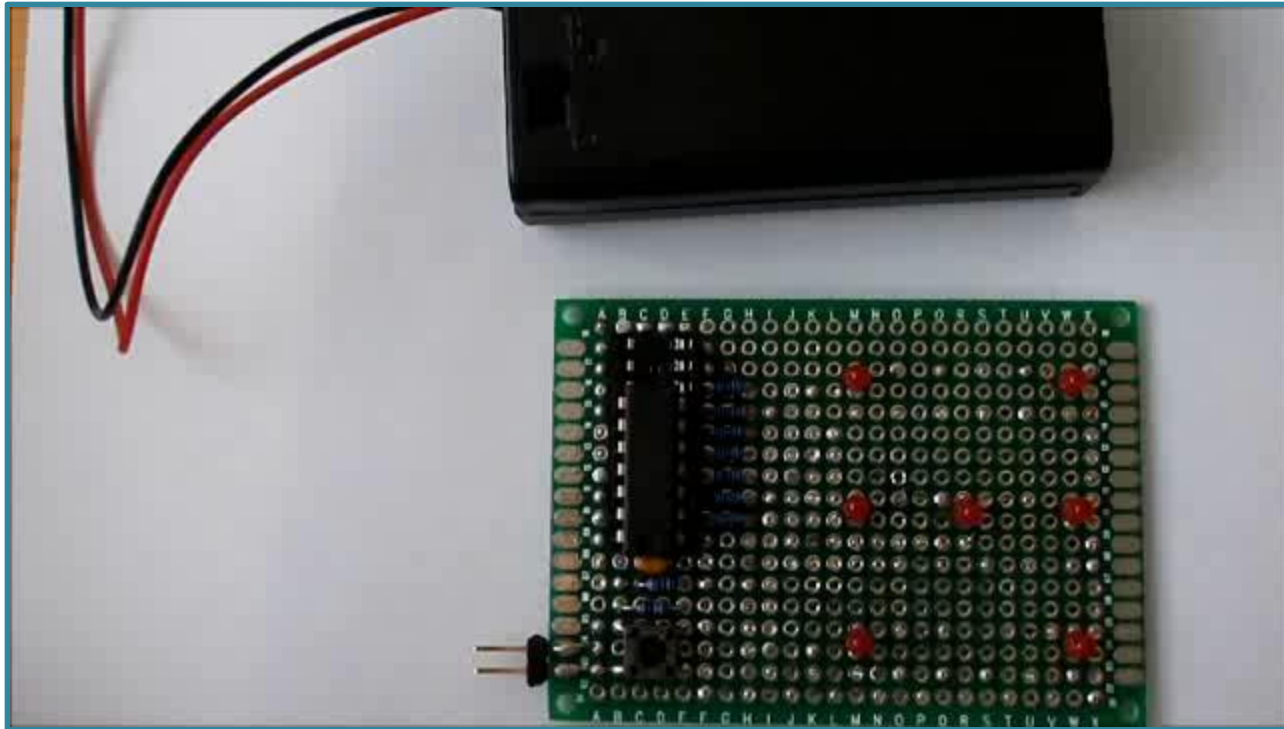


Elektronikus dobókocka

Elektronikus dobókocka



Elektronikus dobókocka

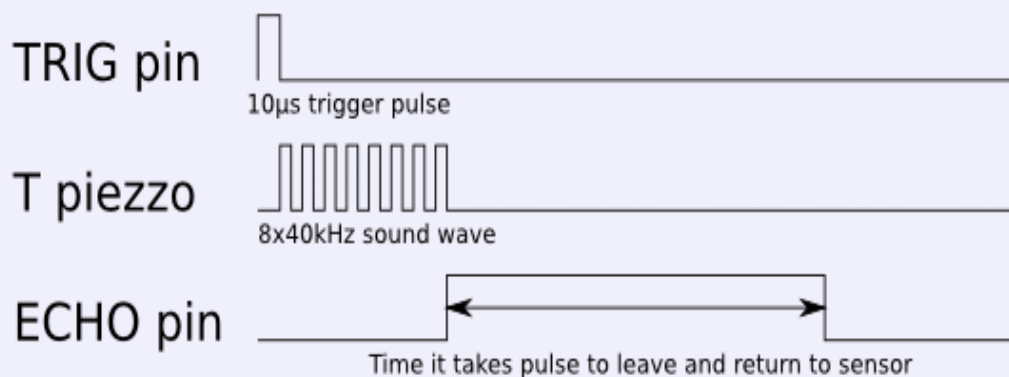


Ultrahangos távolságmérés

A **HC-SR04** modul piezo jeladója az indító impulzus hatására egy 40 kHz-es jelcsomagot sugároz ki. A modul digitális kimenő impulzusának szélessége megegyezik a visszaverődött hang terjedési idejével.



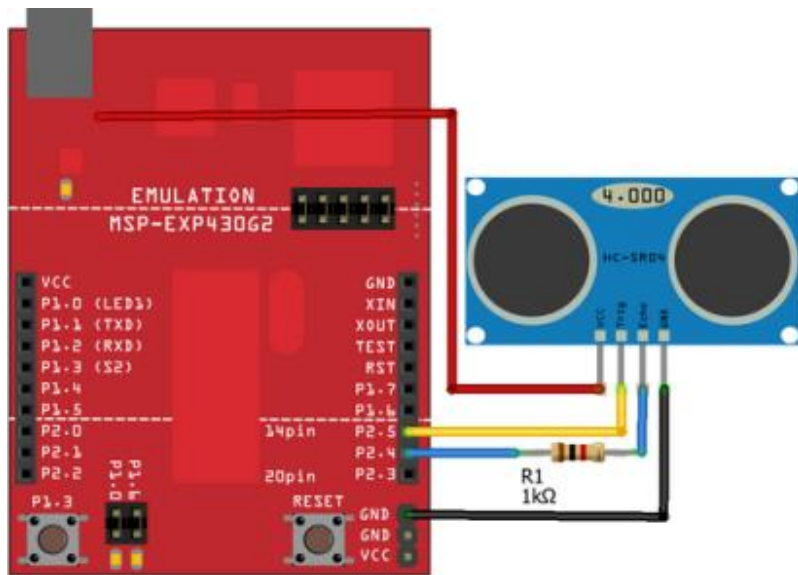
HC-SR04 Timing Chart



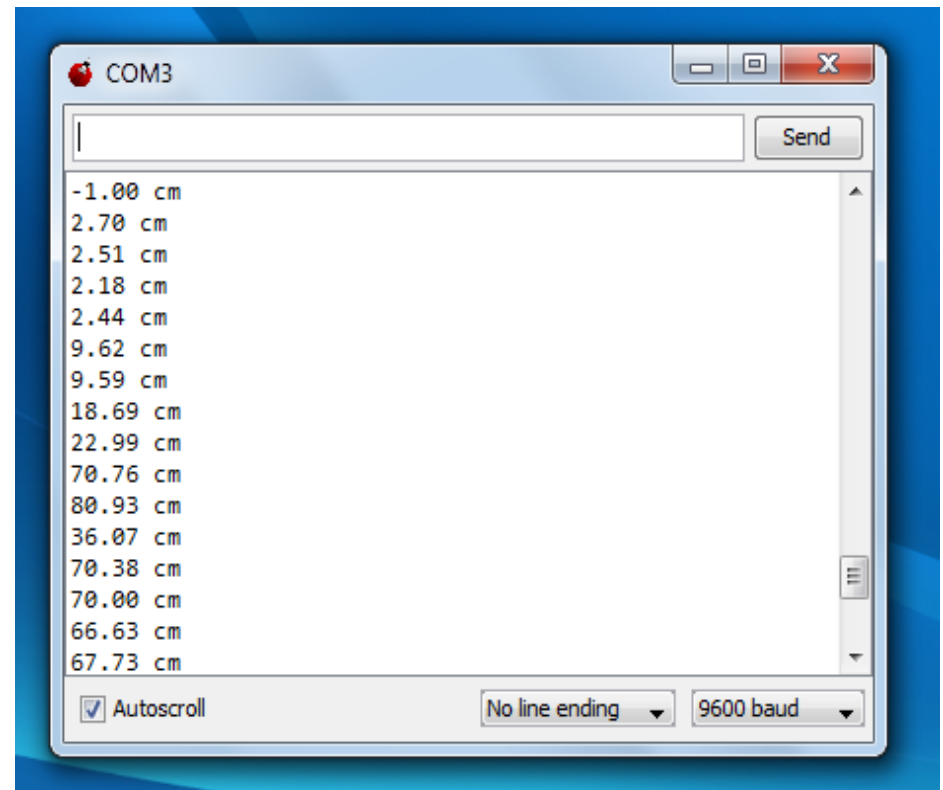
Főbb paraméterek

- Tápfeszültség: 4.5 V – 5.5 V
- Mérési tartomány: 2 cm – 4 m (gyakorlatban inkább 2 m)
- Érzékelési szögtartomány: $\sim 16^\circ$

Ultrahangos távolságmérés (Lab09: Sonar.ino)



Made with Fritzing.org



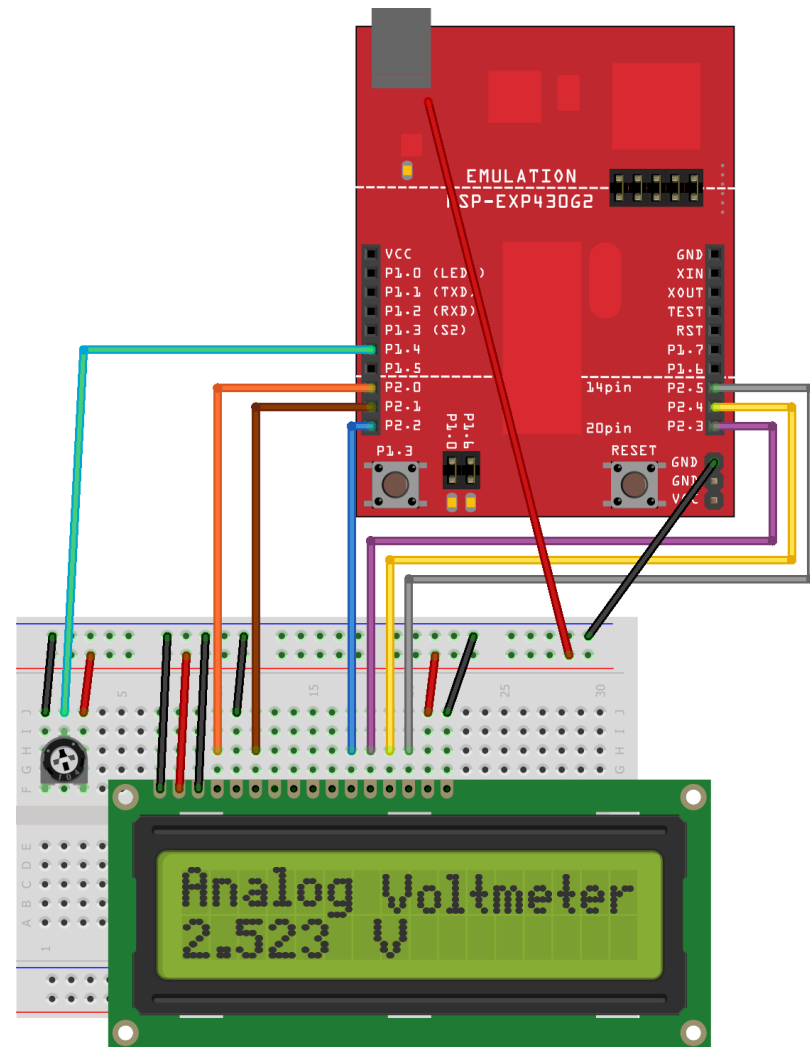
Feszültségmérő vagy hőmérő LCD kijelzéssel

Lab11: LCD16x2_voltmeter, LCD16x2_thermometer

Az Energia IDE beépített könyvtárai segítségével egyszerűen kezelhetjük a perifériákat.

Az **analogRead(A4)**; függvényhívással megmérjük az **A4** bemenetre kapcsolt feszültséget (0 – 3.5 V közötti feszültséget kapcsolhatunk rá).

Az eredményt voltokra számítjuk át, majd kijelezzük egy 16x2 karakteres LCD modul segítségével.



Made with  Fritzing.org

Hőmérséklet és relatív páratartalom mérése

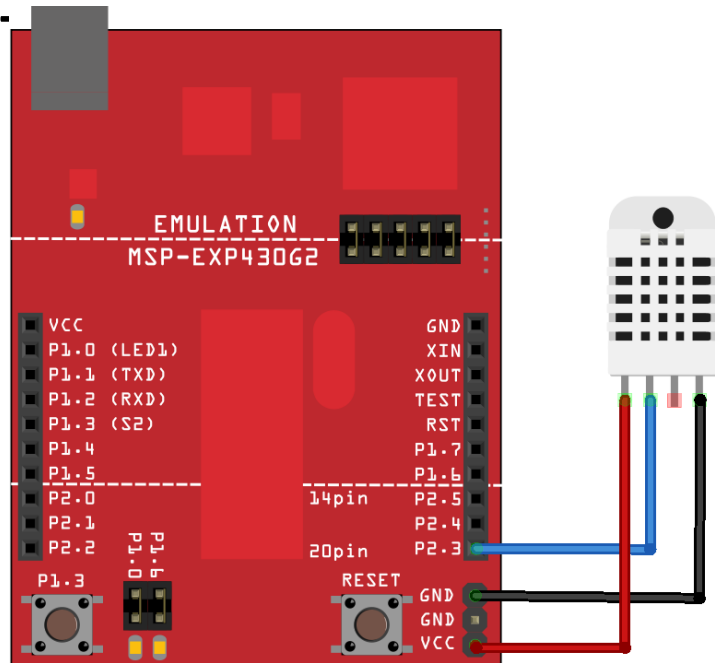
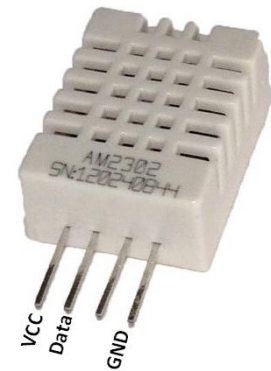
Lab09: TRHlogger.ino

Az AM2302 SZENZOR FŐBB JELLEMZŐI

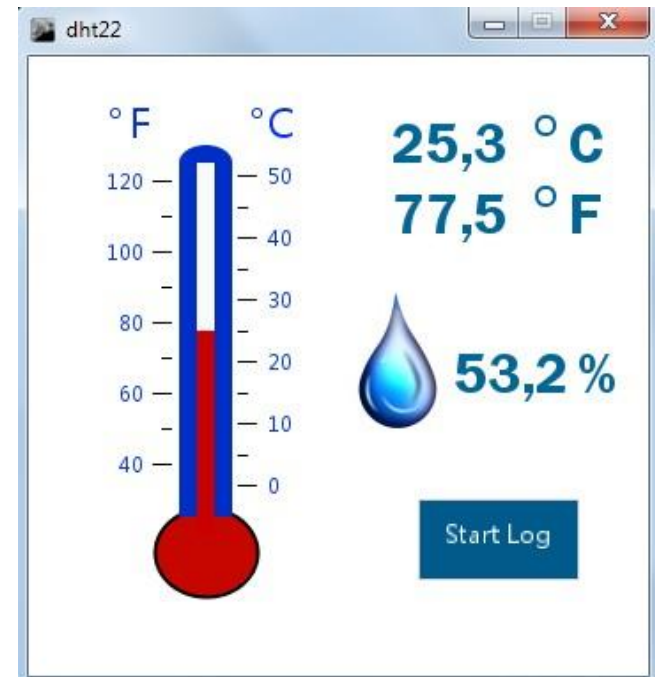
Felbontás: hőmérséklet 0.1 °C és rel. páratartalom 0.1 %

Kommunikáció: 1-wire, nem szabványos protokoll, 4 bájttal adat + 1 bájttal ellenőrző összeg.

Mintavételezési gyakoriság: 2 másodpercenként

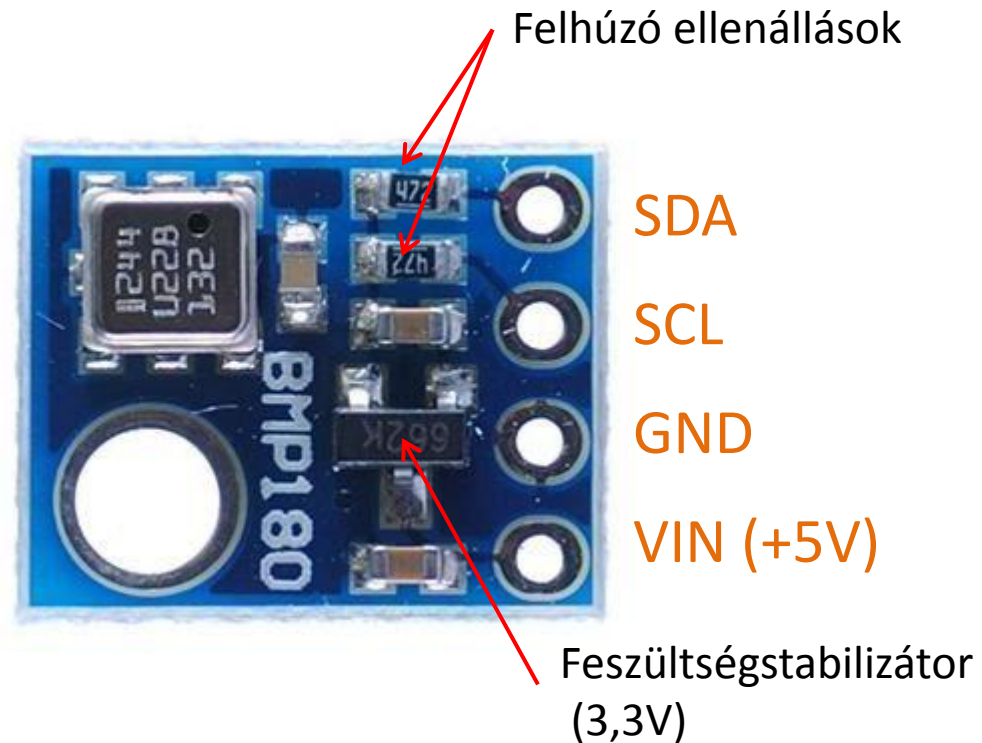
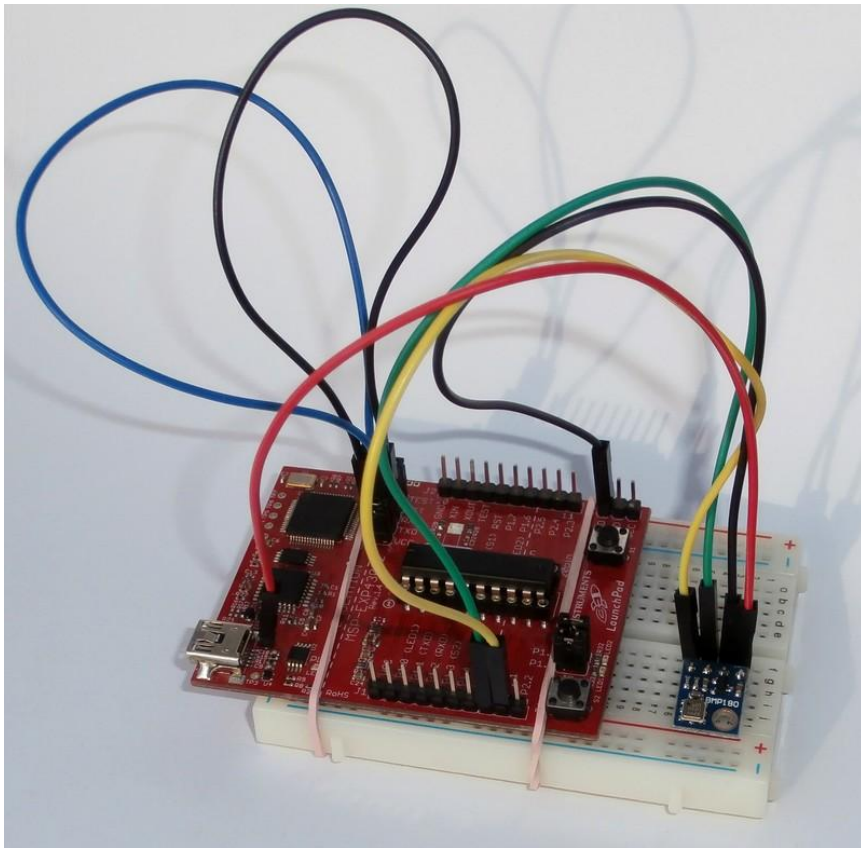


Made with  Fritzing.org



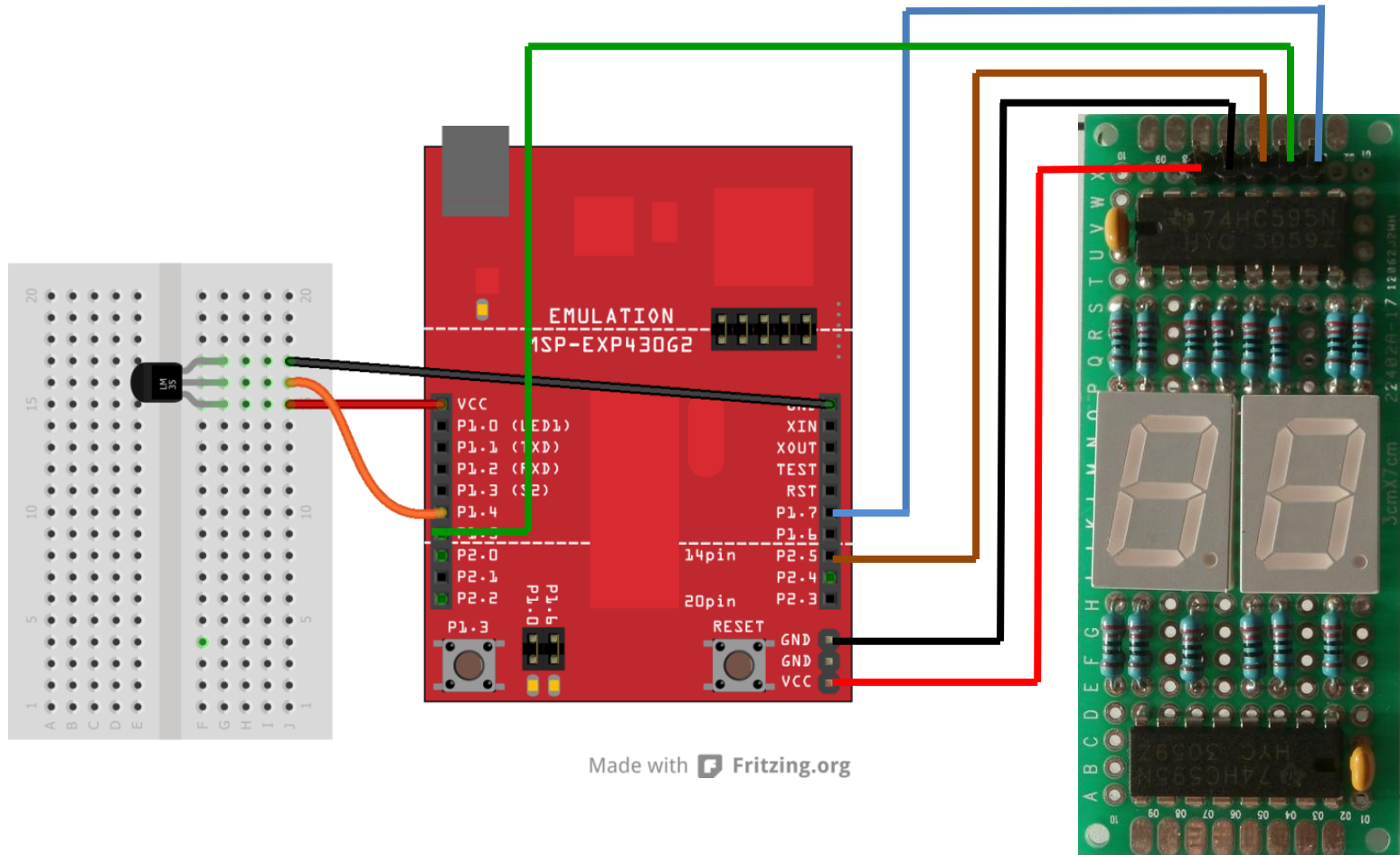
Légnyomás mérése BMP180 szenzorral

Lab10: PressureSensor_sw.ino



Hőmérő LED 7-segmens kijelzővel

Lab13: SPI_595_spec_thermometer



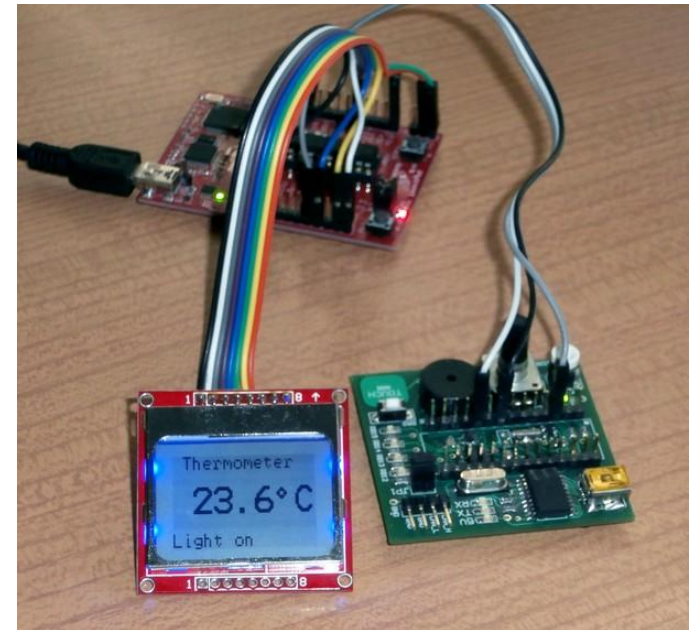
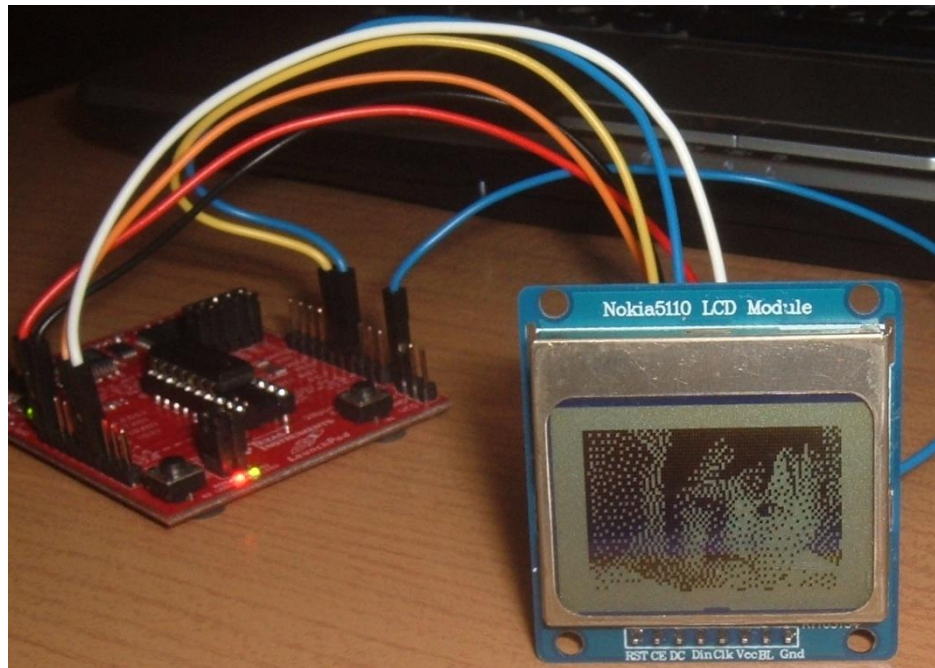
Grafikus kijelző vezérlése

LCD_5110_bitmap, LCD_5110_thermometer

lásd: Lab15

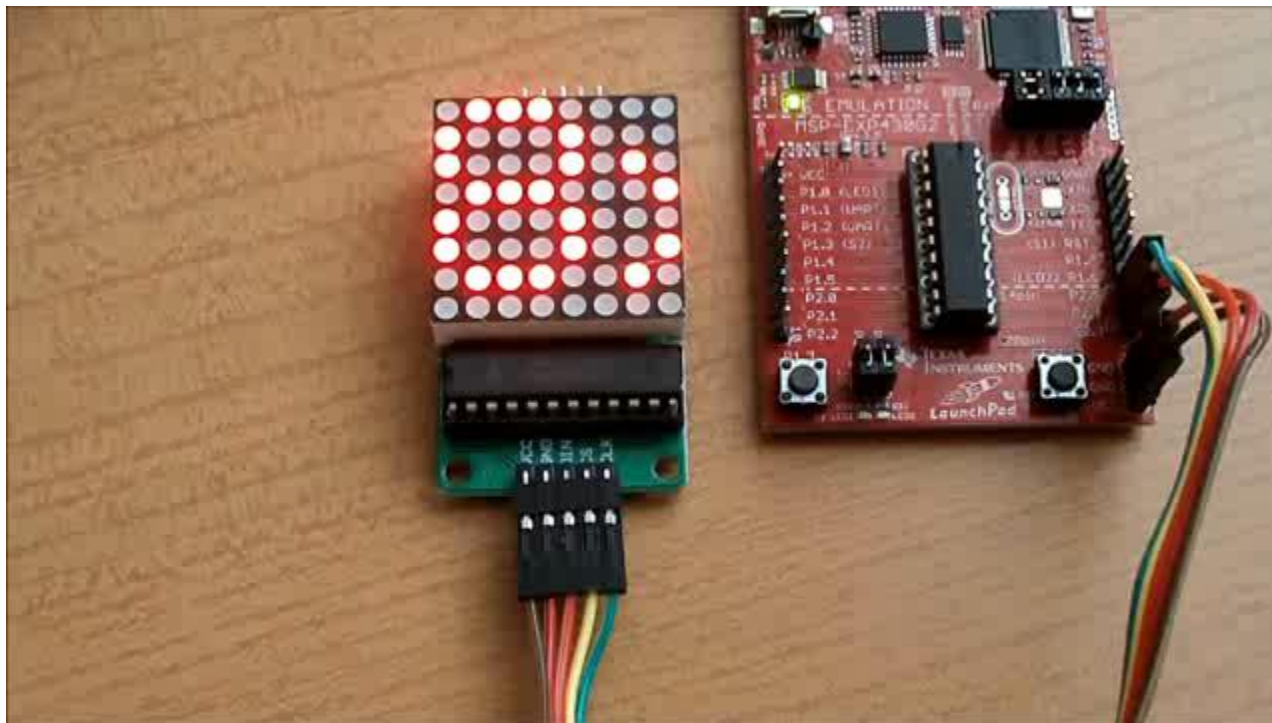
Bitmap kép kiküldése
(nem változtatható, esetleg animáció)

Grafikus kijelző programozott vezérlése
(Hőmérő alkalmazás)



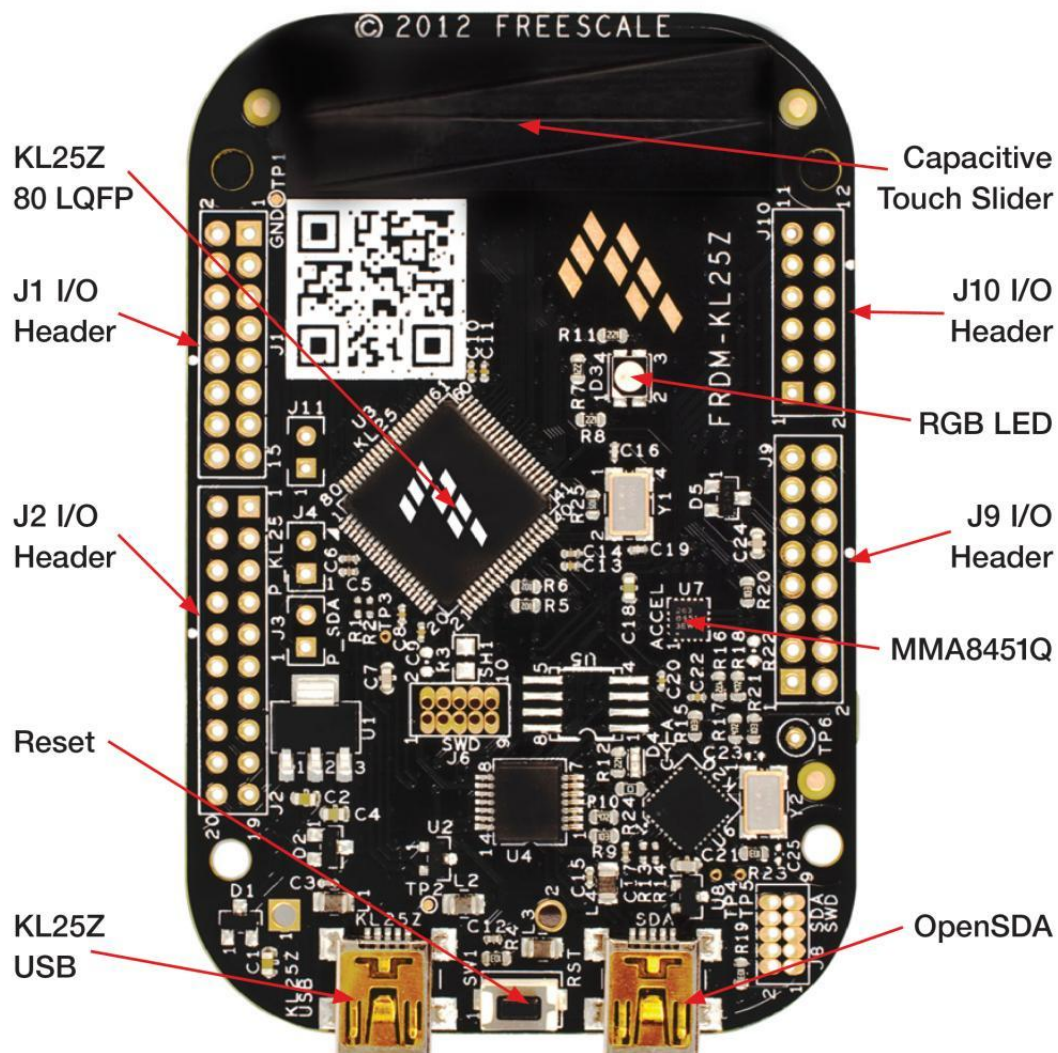
LED 8x8 mátrix alkalmazása

LAB16: LED8x8_MAX7219_LedControl.ino



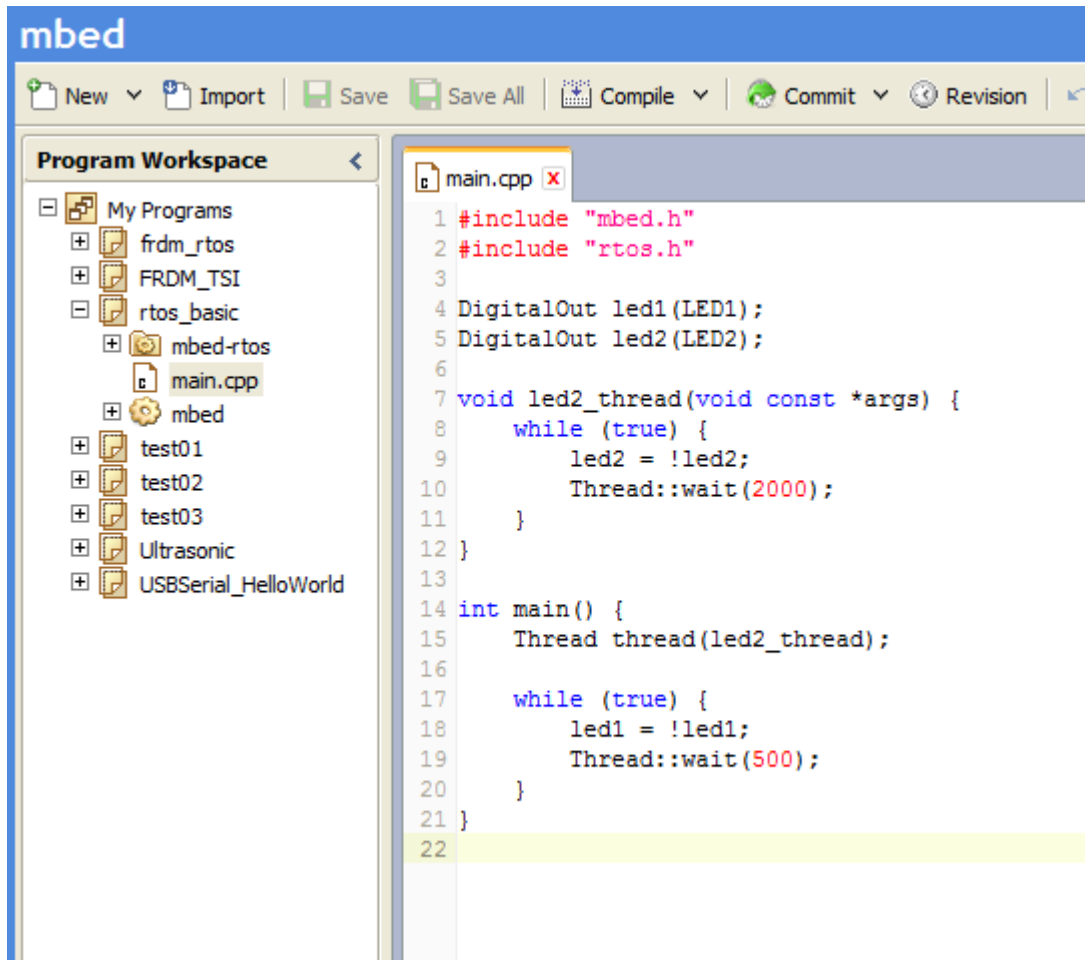
IV. 32 bites mikrovezérlők programozása

- ❖ ARM Cortex-M0+ CPU
- ❖ Nagyobb memória (8-16 kB RAM, 64-128kB ROM)
- ❖ Nagyobb sebesség (48 MHz)
- ❖ Több periféria
- ❖ Programozó és nyomkövető
- ❖ 3,3 V-os jelszint
- ❖ A kártyán Arduino kompatibilis kivezetések, RGB LED, és MMA8451Q gyorsulásérzékelő, kapacitív érintésérzékelő is van
- ❖ Programozás
 - C/C++
 - mbed
- ❖ RTOS támogatás



Programozás mbed környezetben

A webes környezet az mbed.org honlapján érhető el (ingyenes regisztráció szükséges)



```
1 #include "mbed.h"
2 #include "rtos.h"
3
4 DigitalOut led1(LED1);
5 DigitalOut led2(LED2);
6
7 void led2_thread(void const *args) {
8     while (true) {
9         led2 = !led2;
10        Thread::wait(2000);
11    }
12 }
13
14 int main() {
15     Thread thread(led2_thread);
16
17     while (true) {
18         led1 = !led1;
19         Thread::wait(500);
20     }
21 }
22
```

A mellékelt program többszálú futtatást tesz lehetővé az **rtos** könyvtár felhasználásával.

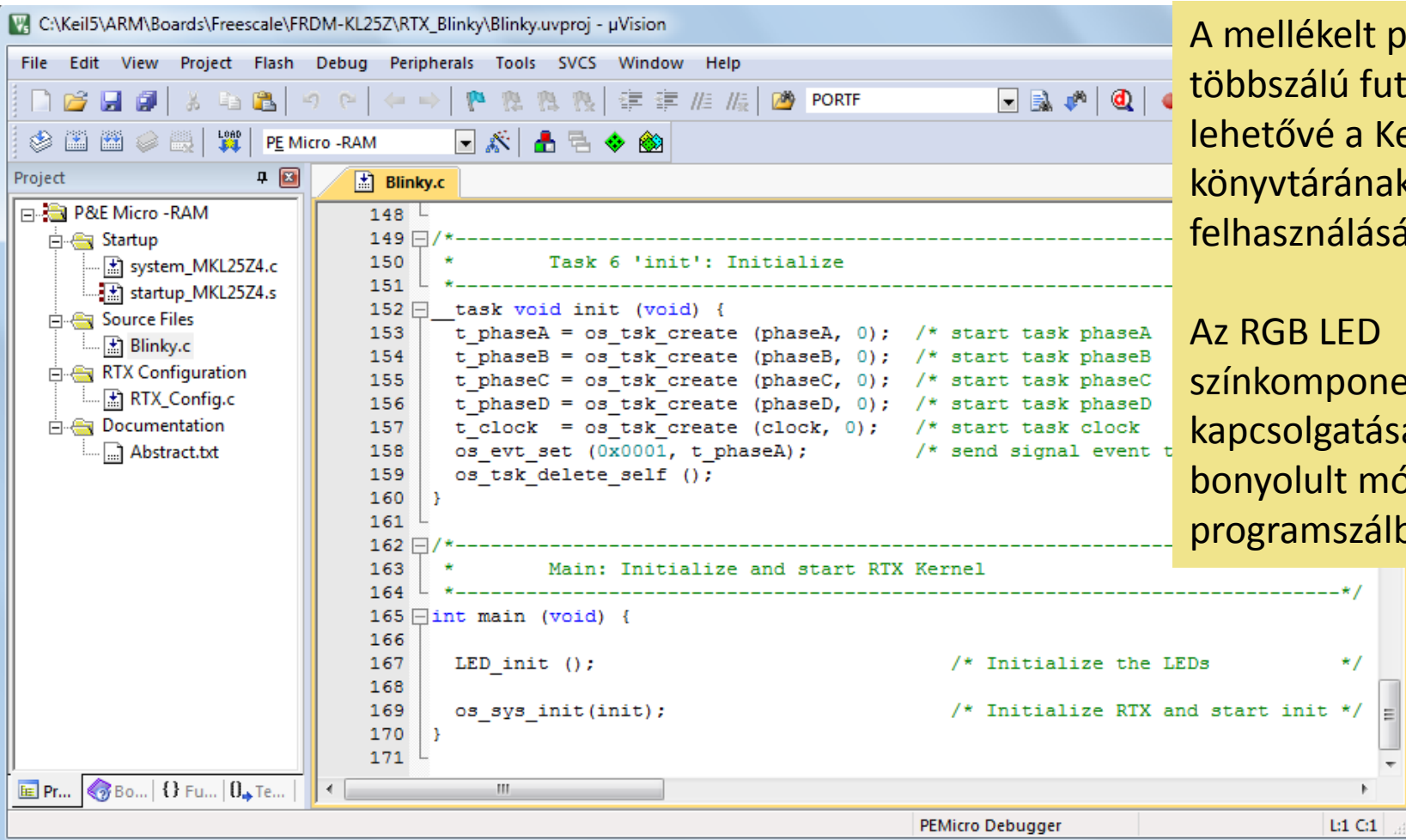
Két LED-et villogtatunk: LED1-et a fő programszálban, LED2-t pedig egy második programszálban.

Programozás Keil uVision környezetben

Az ingyenes változat 32 kB kódméretben limitált. A www.keil.com címről tölthető le.

A mellékelt program többszálú futtatást tesz lehetővé a Keil RTX könyvtárának felhasználásával.

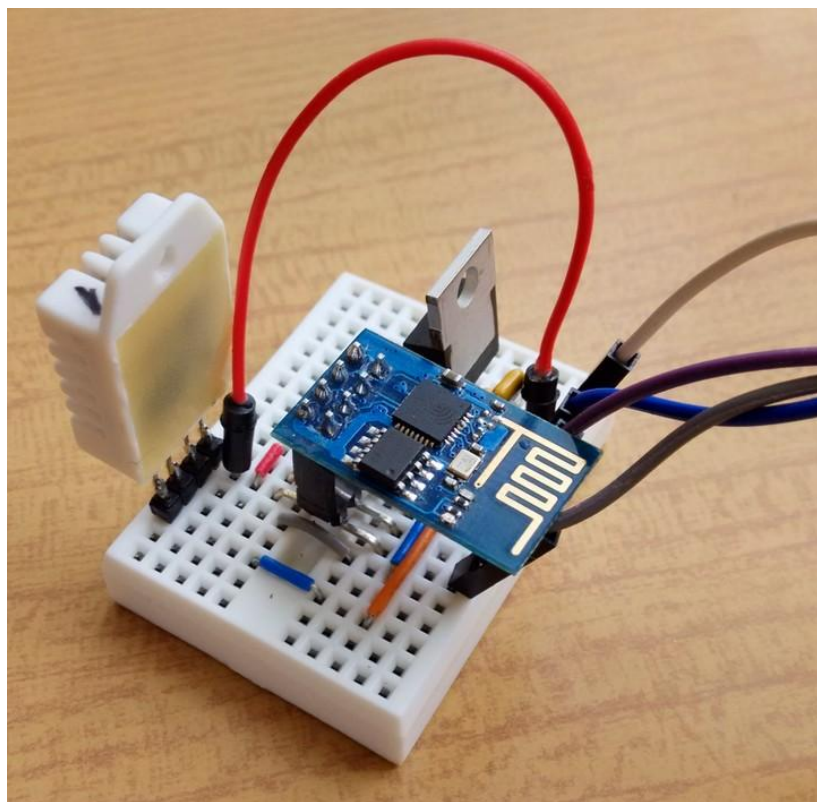
Az RGB LED színtelepeinek kapcsolgatása kicsit bonyolult módon, 6 programszálban történik.



The screenshot shows the Keil uVision IDE interface. The project tree on the left includes 'P&E Micro -RAM', 'Startup', 'Source Files', 'RTX Configuration', and 'Documentation'. The main editor displays the 'Blinky.c' file with the following code:

```
148 |
149 | /*-----*/
150 | *           Task 6 'init': Initialize
151 | *-----*/
152 | _task void init (void) {
153 |     t_phaseA = os_tsk_create (phaseA, 0); /* start task phaseA
154 |     t_phaseB = os_tsk_create (phaseB, 0); /* start task phaseB
155 |     t_phaseC = os_tsk_create (phaseC, 0); /* start task phaseC
156 |     t_phaseD = os_tsk_create (phaseD, 0); /* start task phaseD
157 |     t_clock = os_tsk_create (clock, 0); /* start task clock
158 |     os_evt_set (0x0001, t_phaseA); /* send signal event t
159 |     os_tsk_delete_self ();
160 | }
161 |
162 | /*-----*/
163 | *           Main: Initialize and start RTX Kernel
164 | *-----*/
165 | int main (void) {
166 |     LED_init (); /* Initialize the LEDs */
167 |
168 |     os_sys_init (init); /* Initialize RTX and start init */
169 | }
170 |
171 |
```

V. IoT alkalmazások, kommunikáció, webszerver, webkliens alkalmazások



Arduino nyelven programozott ESP8266
WiFi modul

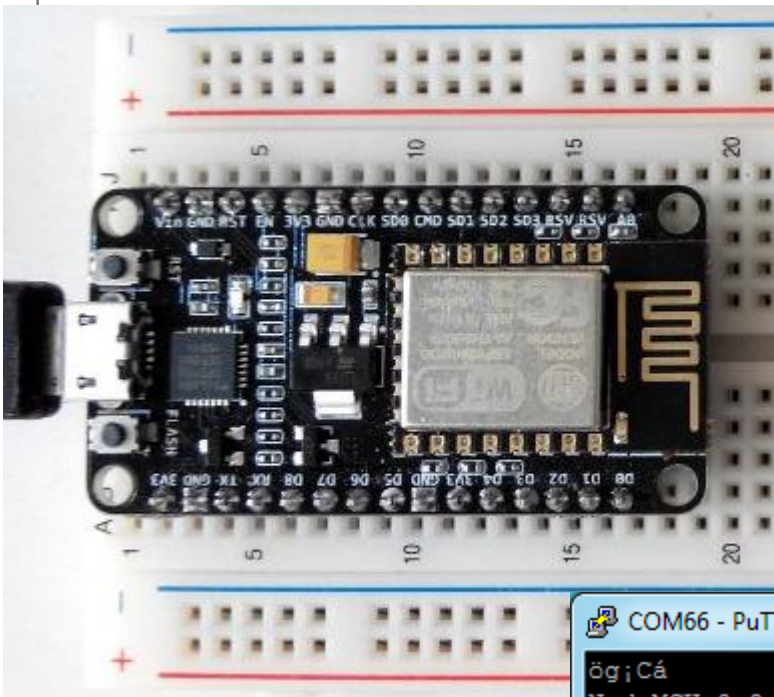


TP-Link WR-703N router OpenWRT
oprendszerrel, USB csatlakozással

OpenWRT alkalmazás Lua nyelven

```
local http = require "socket.http"
local ltn12 = require "ltn12"
io.input("/dev/ttyUSB0")
while true do
    h,t = io.read("*n", "*n")
    if h and h ~= "" and t and t ~= "" then
        local reqbody = string.format("1=%0.1f&2=%0.1f",t,h)
        local respbody = {}
        local body, code, headers, status = http.request {
            method = "POST",
            url = "https://api.thingspeak.com/update",
            source = ltn12.source.string(reqbody),
            headers =
                {
                    ["Host"] = "api.thingspeak.com",
                    ["Connection"] = "close",
                    ["X-THINGSPEAKAPIKEY"] = "XXXX*APIKEYXXXXXX",
                    ["Content-Type"] = "application/x-www-form-urlencoded",
                    ["content-length"] = string.len(reqbody)
                },
            sink = ltn12.sink.table(respbody)
        }
    end
end
```

NodeMCU – ESP8266 Lua interpreterrel



32 bites mikrovezérlő
beépített WiFi perifériával
~ 40 kB RAM, 4 Mbit SPI Flash
A kártya USB –soros átalakítót is tartalmaz
9 digitális I/O (3,3 V-os jelszint)
1 analóg bemenet (1,8 V max)

Lua mintaprogram LED villogtatáshoz

```
COM66 - PuTTY
ö;Cá
NodeMCU 0.9.5 build 20150318 powered by Lua 5.1.4
lua: cannot open init.lua
> while 1 do
>>  gpio.write(3, gpio.HIGH)
>>  tmr.delay(1000000)  -- wait 1,000,000 us = 1 second
>>  gpio.write(3, gpio.LOW)
>>  tmr.delay(1000000)  -- wait 1,000,000 us = 1 second
>> end
█
```

További lehetőségek

Mikrovérlő és Androidos mobil összekapcsolása
(USB, Bluetooth, WiFi)



Egyszerű robot építése

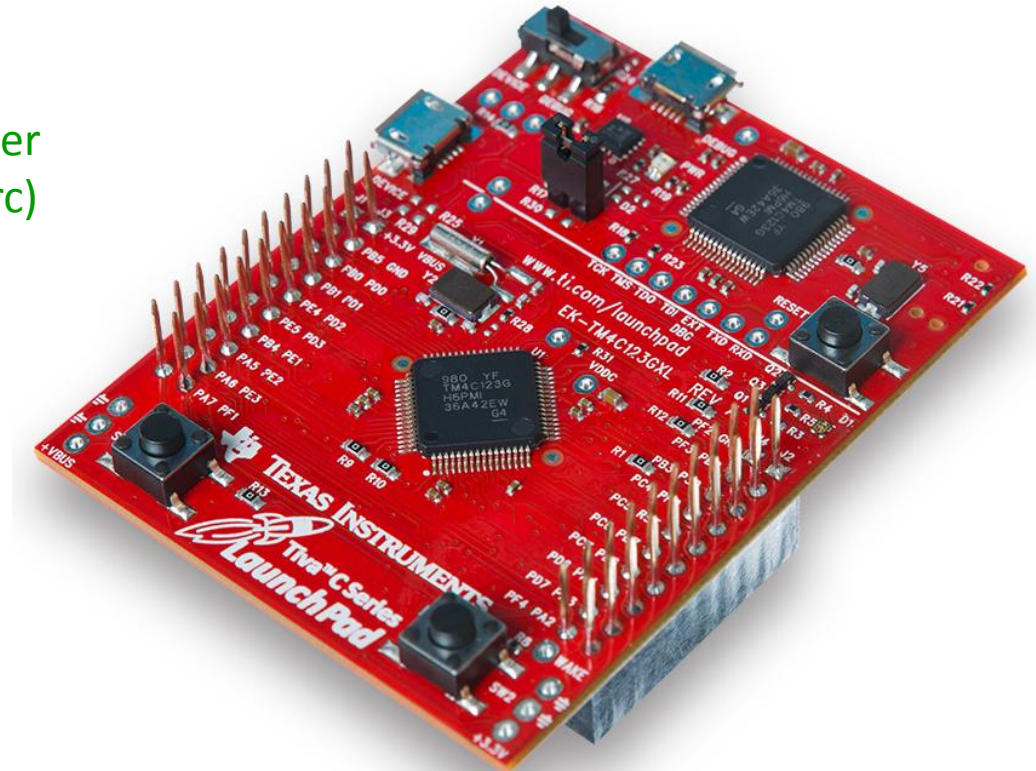
Tiva C Launchpad

Előnyök

- ARM Cortex-M4F 32 bites CPU
- Lebegőpontos műveletek
- ~40 I/O kivezetés
- 256kB flash, 32 kB RAM
- 80 MHz (~100 MIPS)
- USB kommunikáció
- Fejlettebb programozó/debugger
- Pontosabb órajel beállítás (kvarc)
- Gyorsabb UART kapcsolat
- RGB LED a kártyán

Hátrányok

- Sokkal bonyolultabb felépítés
- Sok kivezetés és periféria
- Valamivel költségesebb (\$13)
- Korlátozott Energia támogatás



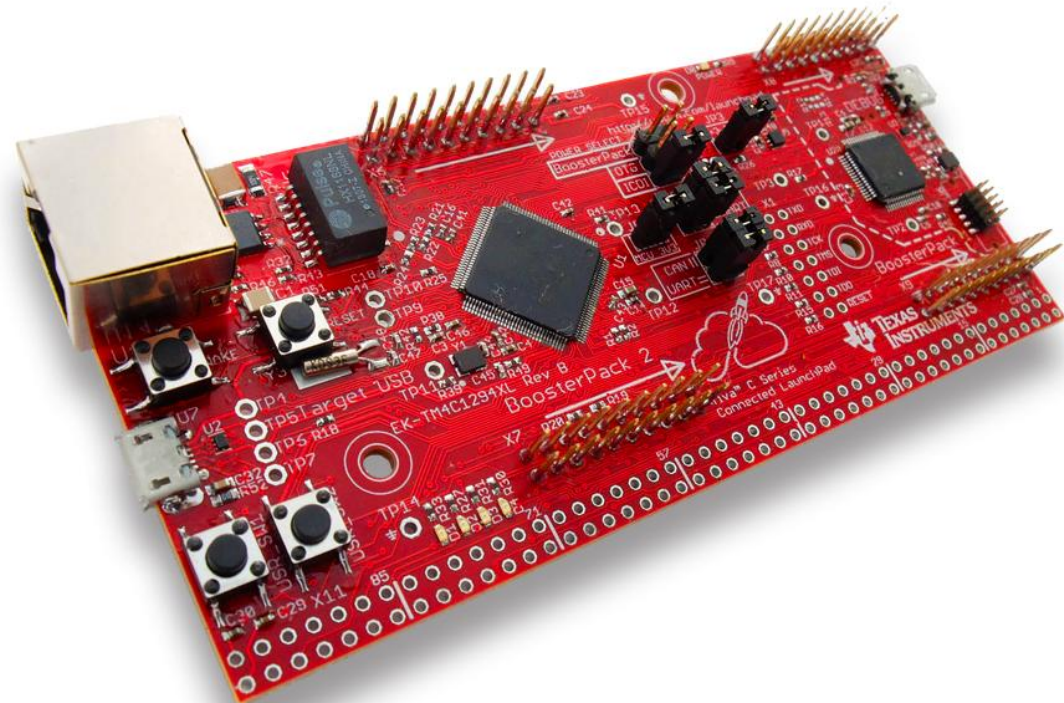
Connected Launchpad

Előnyök

- ARM Cortex-M4F 32 bites CPU
- Lebegőpontos műveletek
- ~80 I/O kivezetés
- 1MB flash, 256 kB RAM
- 120 MHz (~160 MIPS)
- 12 bites ADC
- USB host/device kommunikáció
- 10/100 Ethernet
- Fejlettebb programozó/debugger
- Pontosabb órajel beállítás (kvarc)
- Gyorsabb UART kapcsolat

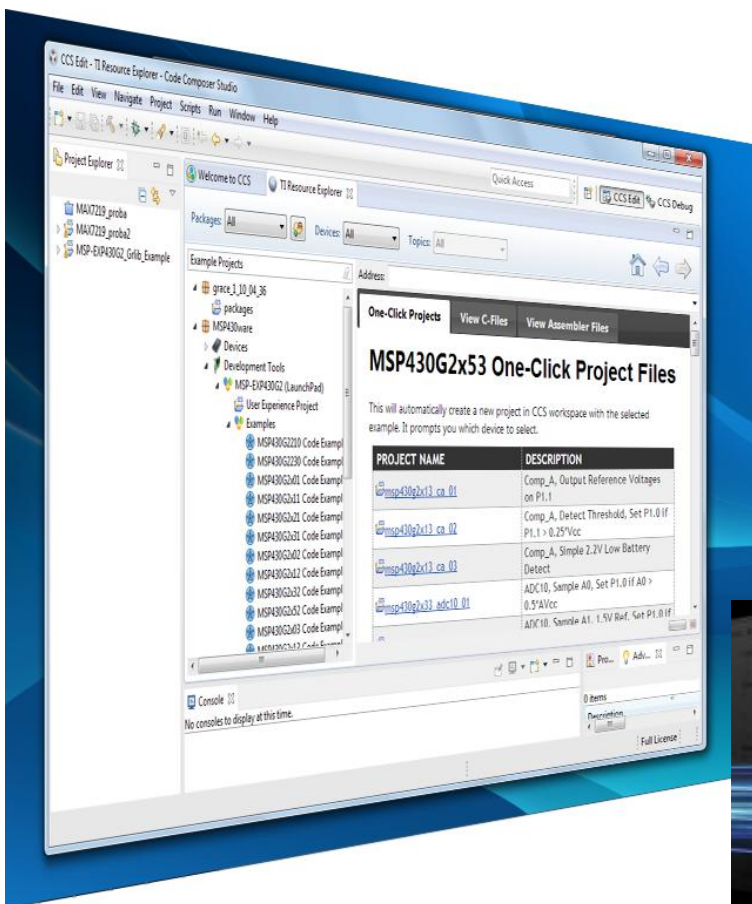
Hátrányok

- Sokkal bonyolultabb felépítés
- Sok kivezetés és periféria
- Valamivel költségesebb (\$20)
- Korlátozott Energia támogatás



Code Composer Studio 6.0

A haladó programfejlesztés eszköze...



- Eclipse alapú IDE
- C/C++ fordító (az ingyenes verzió korlátozott, de használható az mspgcc fordítóval is)
- **MSP430, Cortex-M4** (és sok más) CPU-hoz
- Saját periféria könyvtár (MSPware, Tivaware)
- Mintaprojektek
- Programletöltés
- **Hardveres nyomkövetés**
- Az új változat **importálni tudja az Energia projekteket**

