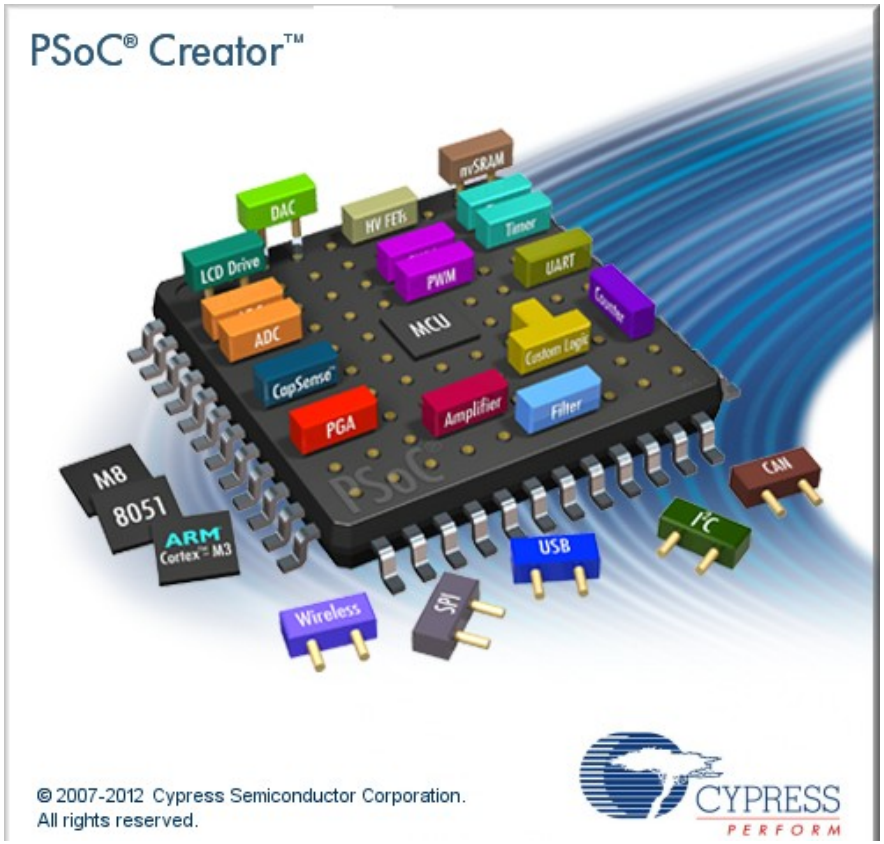
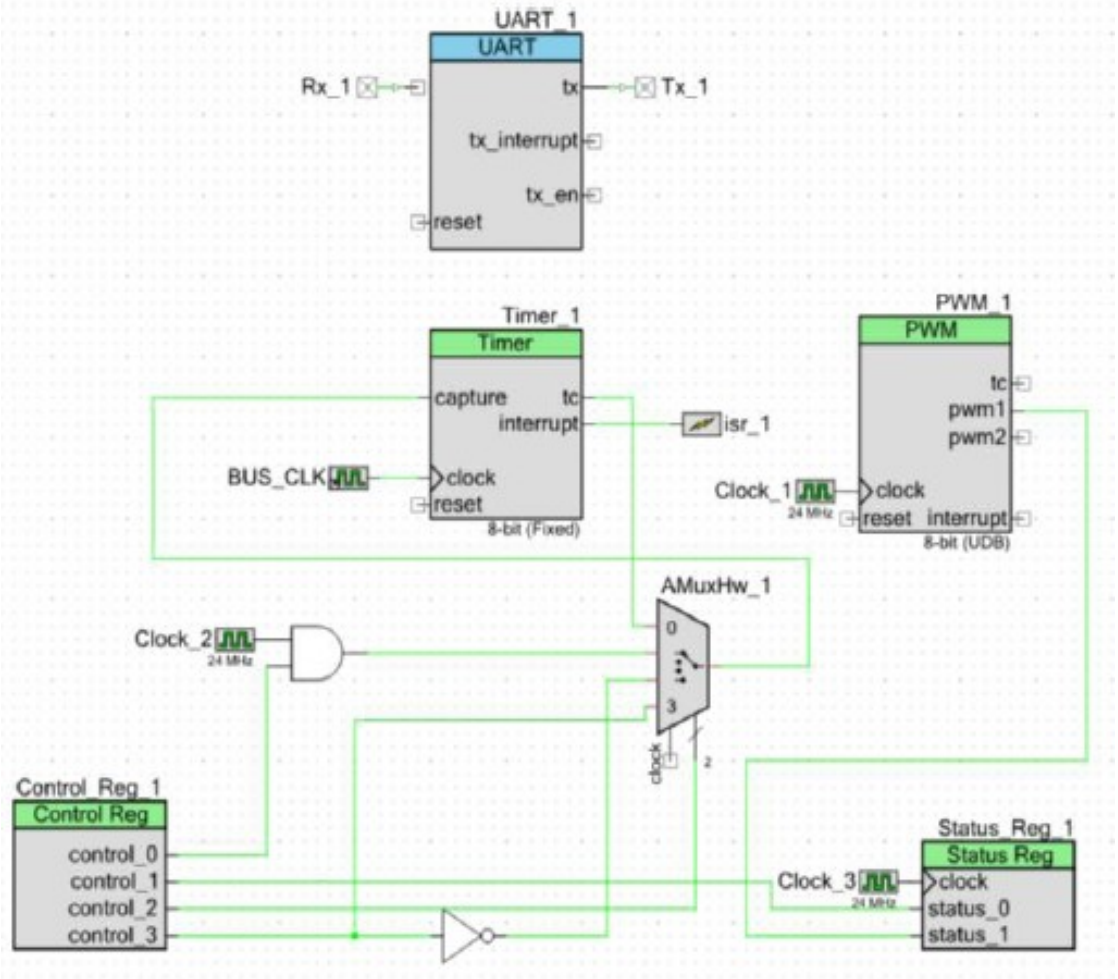
Újrakonfigurálható eszközök



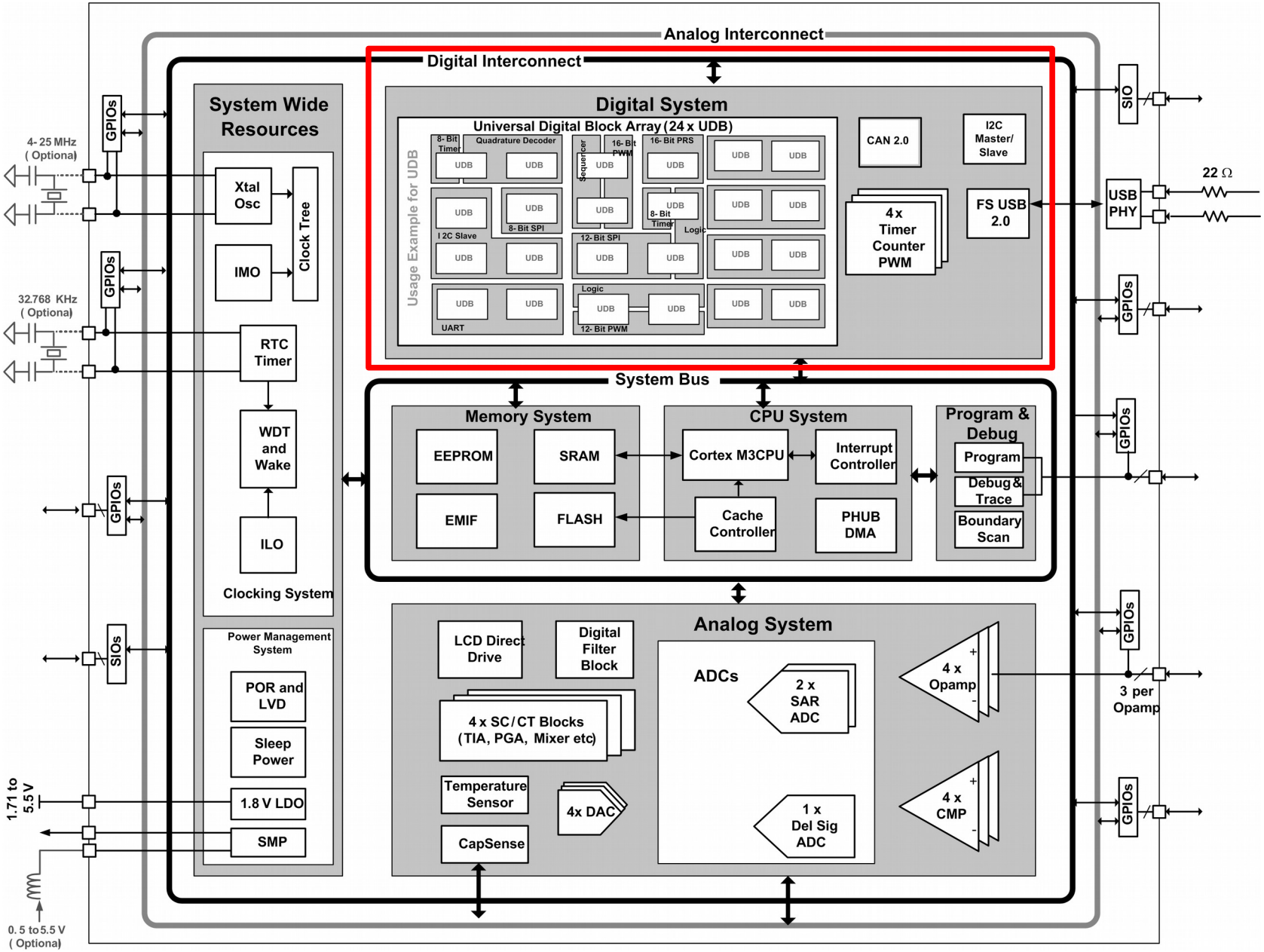


10. Cypress PSOC 5LP digitális perifériák

Felhasznált irodalom és segédanyagok

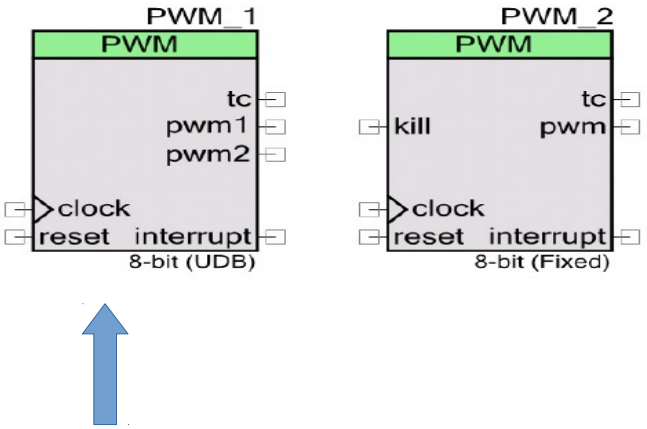
- Cypress: CY8C58LP FamilyDatasheet
- Cypress: PSOC 5LP Architecture Technical Reference Manual)
- Cypress: CY8CKIT-059 Prototyping Kit Guide
- Cypress: AN77 759: Getting Started with PSoC®5LP
- Cypress: PSoC®Creator™ User Guide
- Yuri Magda: Cypress PSoC 5LP Prototyping Kit Measurement Electronics
- Cserny István: PSOC 5LP Mikrokontrollerek programozása

PSOC 5LP Digitális alrendszer



Cypress Component Catalog

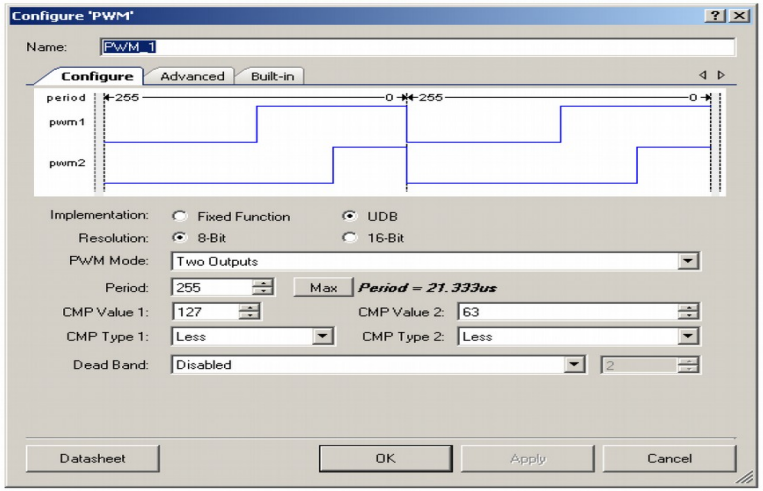
- A **PSOC Creator** integrált fejlesztői környezet számos „előregyártott” alkatrészt és áramköri modult kínál, melyek használatához a hardver komponensekhez adatlapot, grafikus konfigurálási lehetőséget és alkalmazásprogramozói függvényeket is biztosítanak



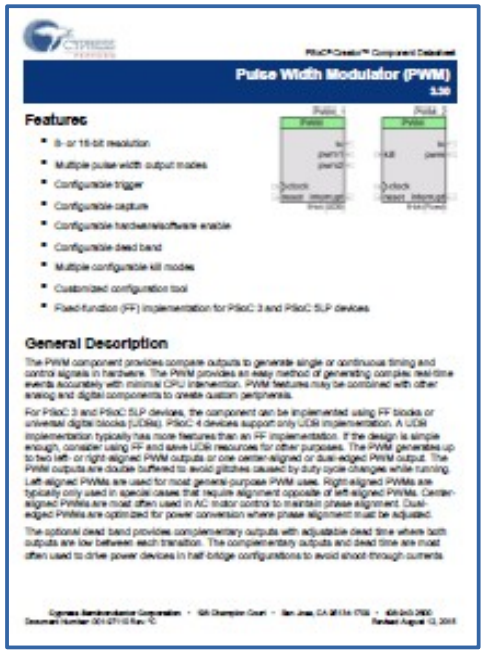
B_PWM_v3_30.v

(Verilog)

Konfigurációs panel



Adatlap



PWM.h

API header

PWM.c

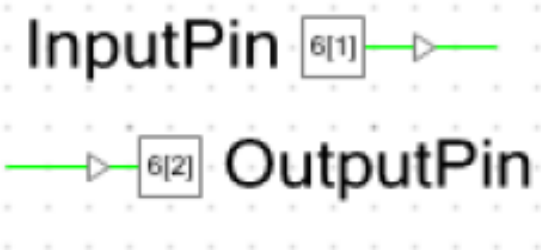
API függvények

Cypress Component Catalog

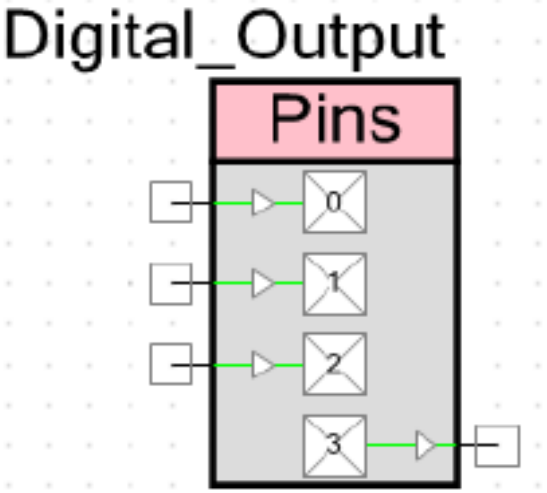
- **Analóg** (ADC, erősítő, komparátor, MUX, DAC, keverő, mintavevő, Vref)
- **Érintésérzékelés** (capsense gomb, csúszka)
- **Kommunikáció** (UART, IIC, SPI, I2S, LIN, CAN, USB)
- **Digitális**
  - Funkcionális egységek (Timer/Counter/PWM, PrISM, CRC, Quadrature decoder)
  - Logikai áramköri elemek (NOT, AND, OR, XOR, NAD, NOR, DFF, MUX, DeMUX)
  - Regiszterek (Control register, Status register) – a CPU-hoz biztosítanak kapcsolatot
  - Segédeszközök (éldetektor, pergésmentesítő, frekvenciaosztó/számláló, komparátor)
- **Megjelenítők** (karakteres és grafikus LCD, szegmens/mátrix kijelző vez.)
- **Szűrők** (digitális szűrő)
- **Port és tüske** (digitális/analóg ki és bemenetek)
- **Tápellátás felügyelet** (Power monitor, Voltage sequencer stb)
- **Rendszer** (bootloader, órajelforrás, DMA, EEPROM stb)
- **Hőmérséklet szabályozás** (vent. vezérlő, hőlem/termisztor kalkulátor)

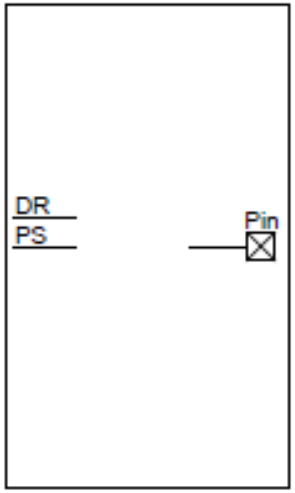
Digitális ki- és bemenetek

- A konfigurációs lapon beállítható a meghajtási üzemmód

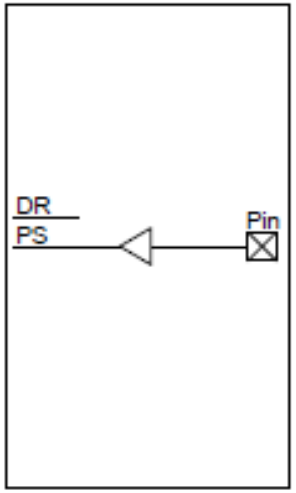


- Több kivezetés összefogható egy **Port** objektummá

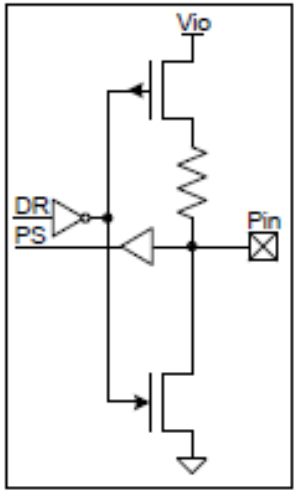




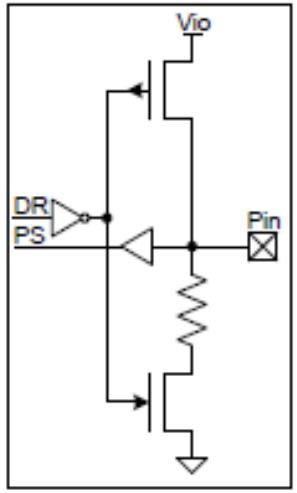
0. High Impedance Analog



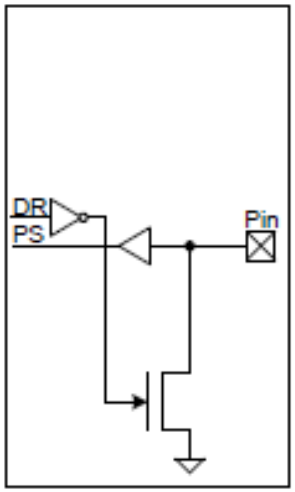
1. High Impedance Digital



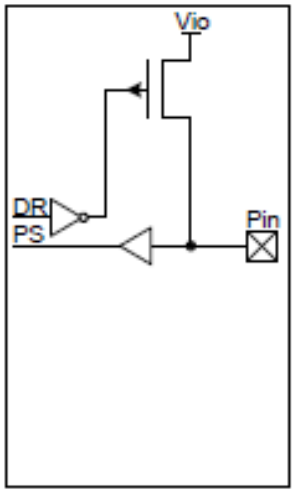
2. Resistive Pull-Up



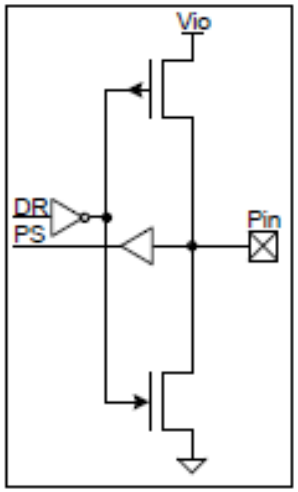
3. Resistive Pull-Down



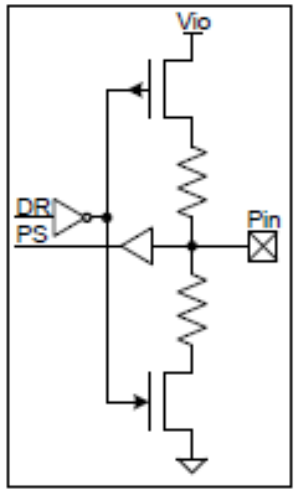
4. Open Drain, Drives Low



5. Open Drain, Drives High



6. Strong Drive



7. Resistive Pull-Up & Down

GPIO „Hello world”

- Az új projekthez adjunk egy digitális kimenetet (P2[1] kivezetés, ami **LED1**-hez van kötve a **CY8CKIT-059** kártyán)
- A főprogramban használjuk a **LED1_Write**(*value*) függvényt!

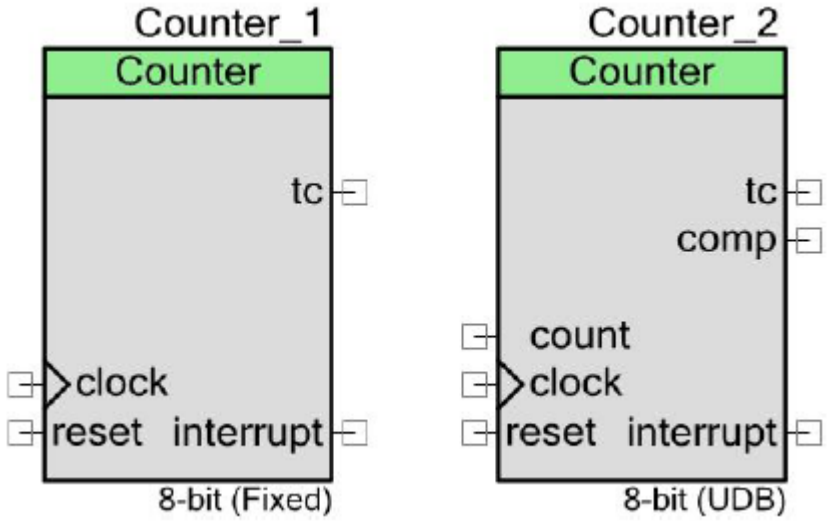
```
#include "project.h"
int main(void) {
    for(;;)    {
       LED1_Write(1);       // LED1 kimenet "magas"
       CyDelay(500);        // 500 ms késleltetés
       LED1_Write(0);       // LED1 kimenet "alacsony"
       CyDelay(500);        // 500 ms késleltetés
    }
}
```

- Így is írhatjuk:

```
#include "project.h"
int main(void) {
    for(;;)    {
       LED1_Write(!LED1_ReadDataReg());      // LED1 átbillentés
       CyDelay(500);                         // 500 ms késleltetés
    }
}
```

Counter – számláló

- **Fix-funkciójú számlálók:**
  - 4 db áll rendelkezésre
  - 8- vagy 16-bites
  - Csak visszaszámlálás lehetséges
  - Egyetlen capture regiszter
- **UDB-alapú számlálók:**
  - 8-, 16-, 24- or 32-bit
  - Sok lehetőség: engedélyezés, számlálás, capture, compare
  - 4 szintű capture FIFO
  - Megszakítás: tc, capture, compare



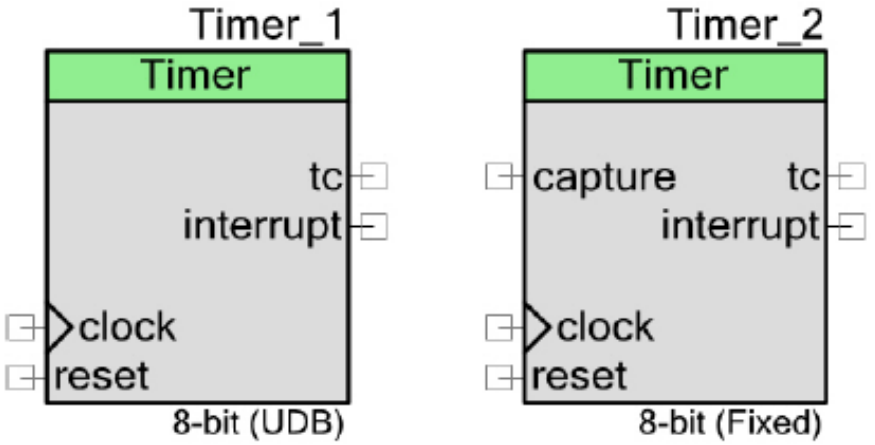
Tc – Terminal Count: amikor a számláló 0-ra vált

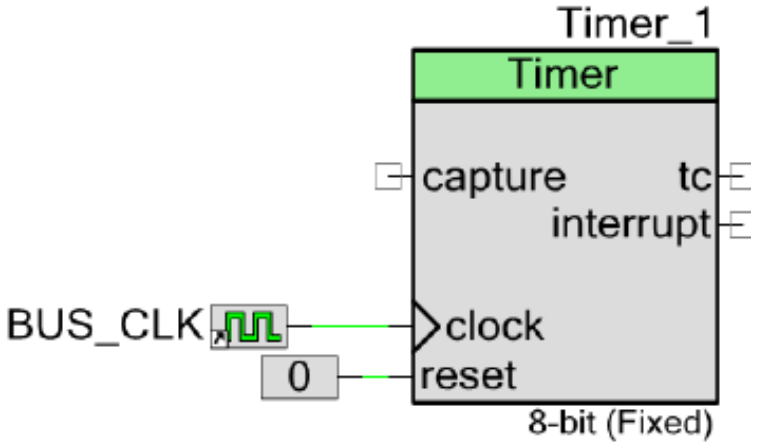
Capture: a számláló értékét eltárolja, amikor a *capture* bemenet aktiválódik

Compare: digitális komparátor, jelet ad, ha a számláló értéke egy előre megadott értékkel megegyezik

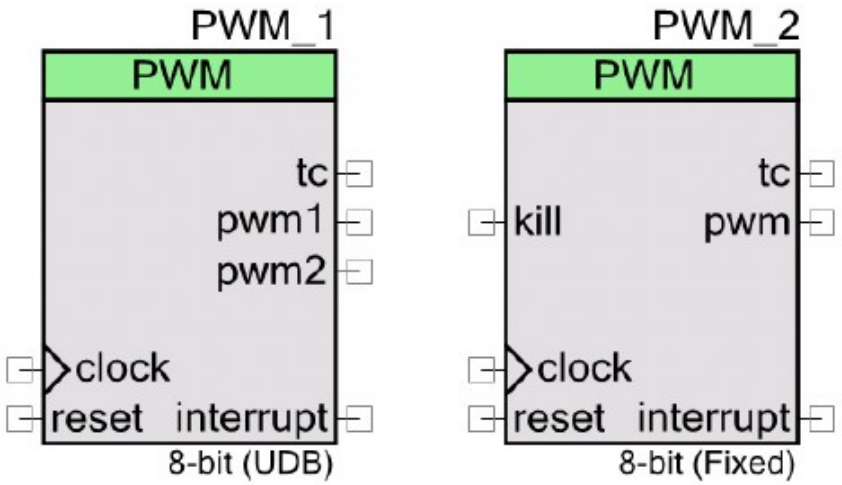
Timer – időzítő

- **Időzítő:** olyan számláló, amely fix frekvenciájú órajelet számlál
  - Hardver események között eltelt idő mérése
  - Események időpontjának rögzítése (capture) – logikai analizátor
  - Periodikus jelek vagy megszakítások keltése
  - Frekvencia osztó
- **Alapértelmezett bekötés →**

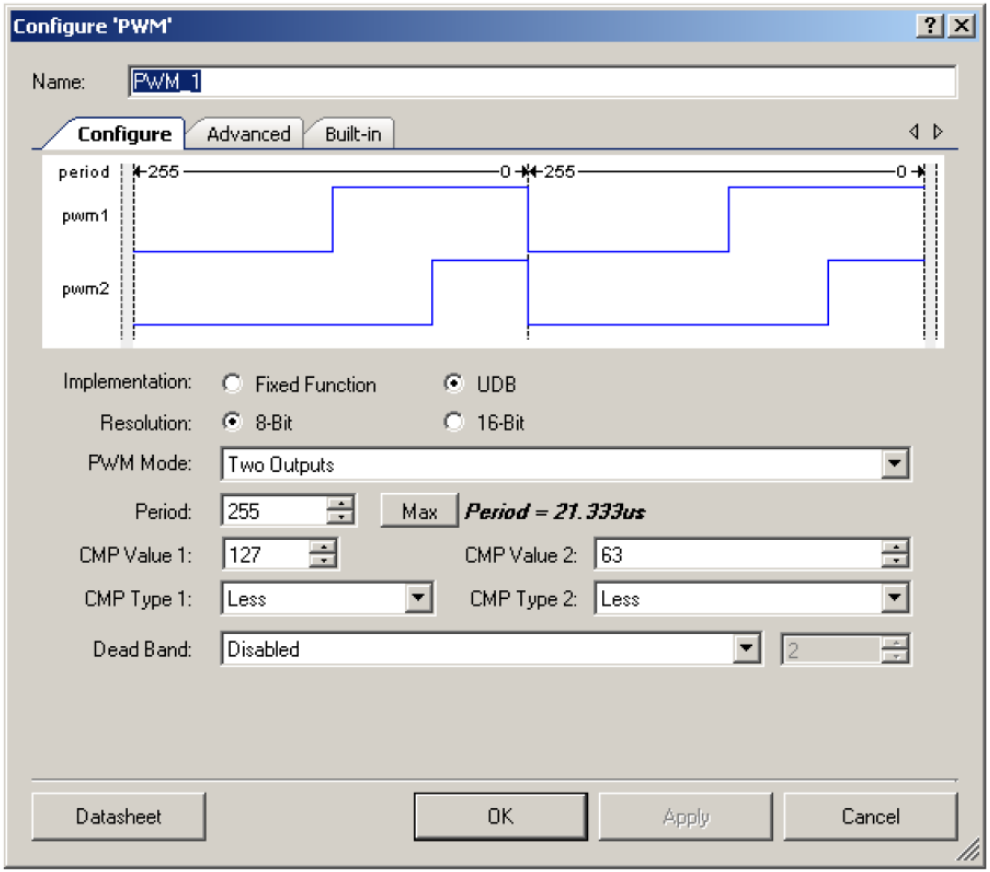




PWM – impulzusszélesség-moduláció



Konfigurálás



PWM: impulzus-szélesség moduláció

- 8- vagy 16-bites felbontás
- 1 vagy 2 kimenőjel
- Állítható holtsáv

Tipikus felhasználás:

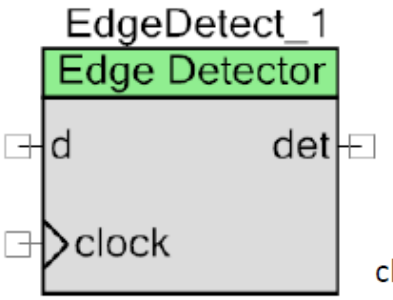
Periodikus hullámalak keltése, változtatható kitöltéssel.

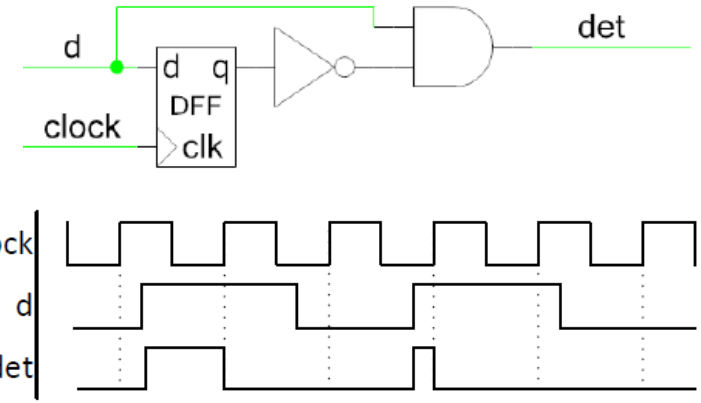
- Teljesítményszabályozás (LED, ventilátor fordulatszám)
- Szervo motorok vezérlése (az impulzusszélesség szabja meg a beállási irányt)

További digitális alkatrészek

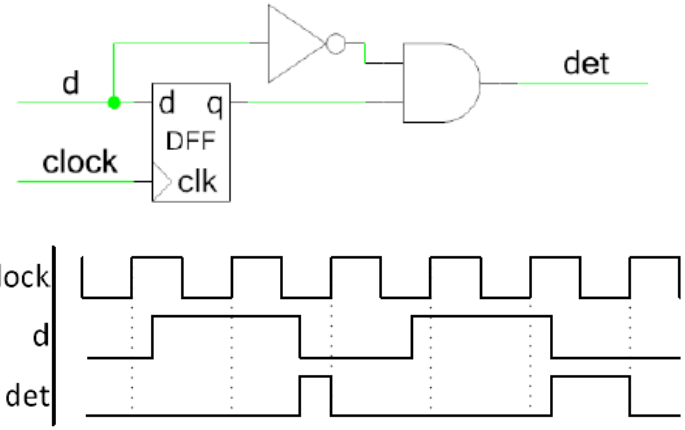
Jelszintváltozás detektálása

Felfutó él



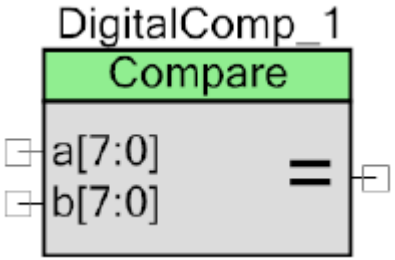


Lefutó él



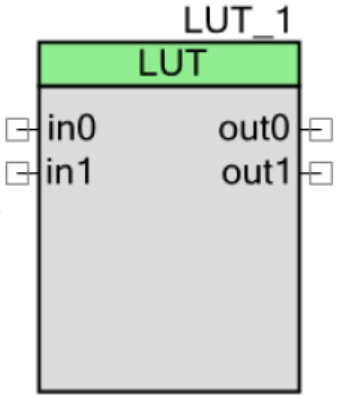
Digitális komparátor (1–32 bit)

- Összehasonlítási módok: =, ≠, <, ≤, >, ≥
- Alapértelmezett mód: **=**



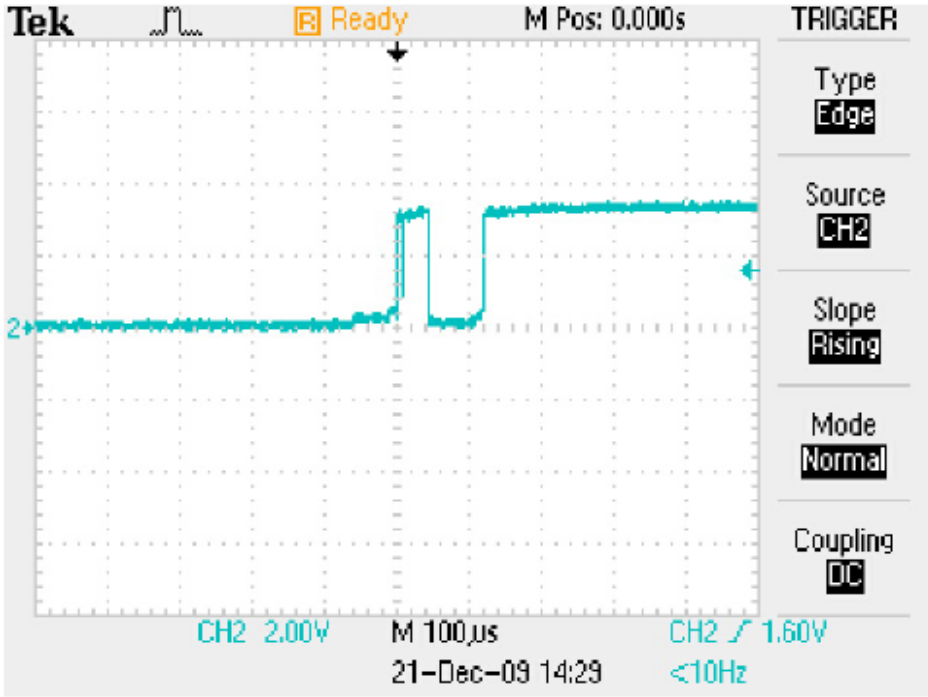
Kereső tábla (look-up table)

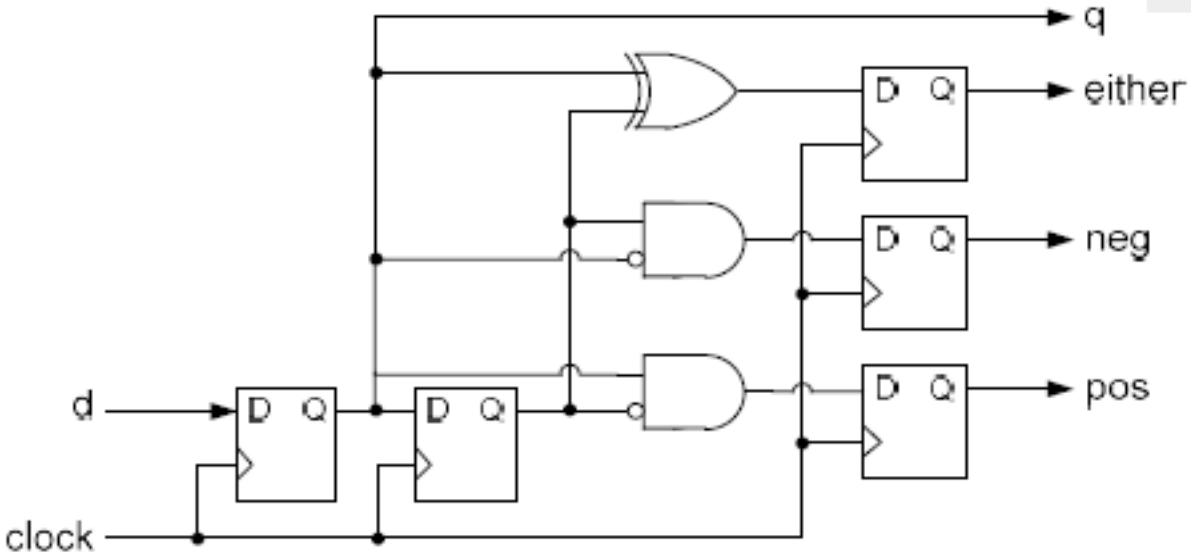
- 1-5 bemenet
- 1-8 kimenet
- Opcionálisan regiszteres kimenet (pl. státuszgép)

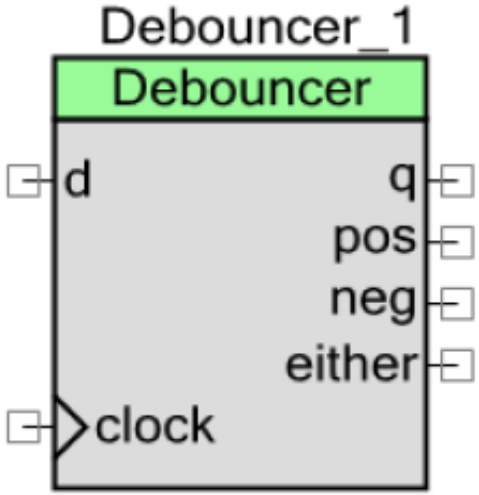


Debouncer (pergésmentesítő)

- A nyomógombok pergését szoftveresen vagy hardveresen ki kell küszöbölni
- A Debouncer komponens a hardveres pergésmentesítést alkalmasan megválasztott időközönkénti mintavételezéssel oldja meg (T = 10 – 20 ms javasolt)

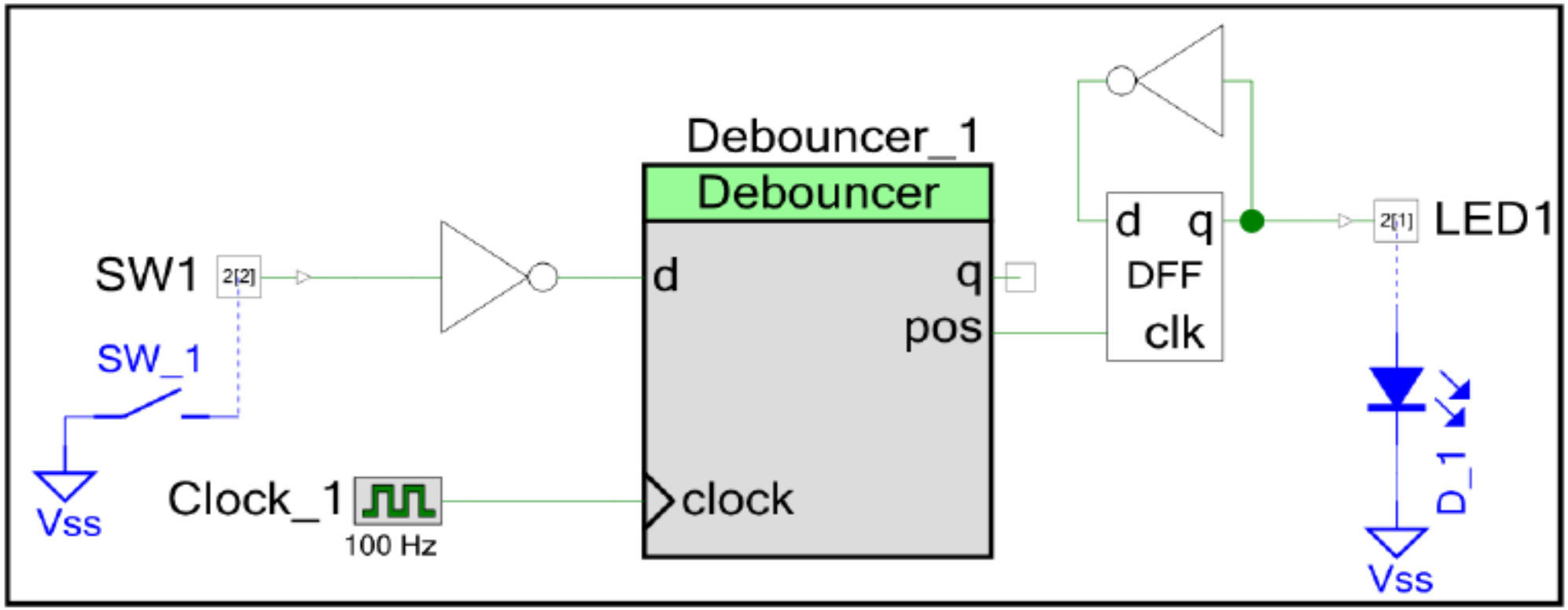






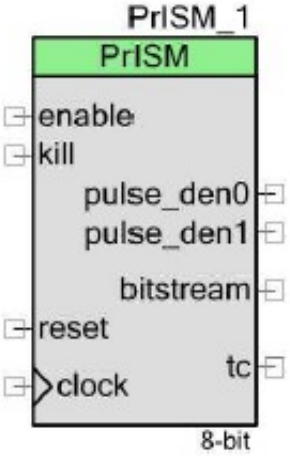
Debouncer projekt

- Az **SW1** nyomógomb jelét invertálás után pergésmentesítjük (100 Hz-es mintavételezés, felfutó él figyelés)
- Ezzel a pergésmentesítéssel egy T flip-flopot (visszacsatolt DFF) billegtetünk, melynek kimenete **LED1**-et kapcsolgatja ki-be
- A működés hardveres, API függvényhívásra nincs szükség

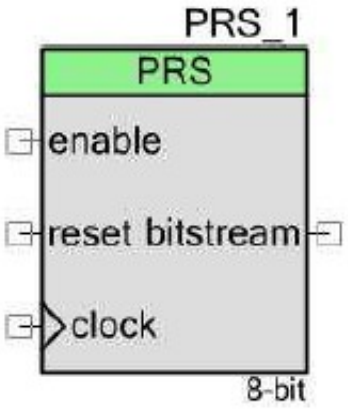


További digitális alkatrészek

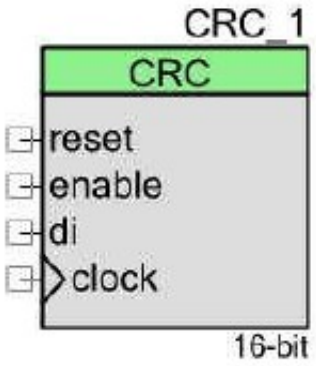
Precision Illumination Signal Modulation



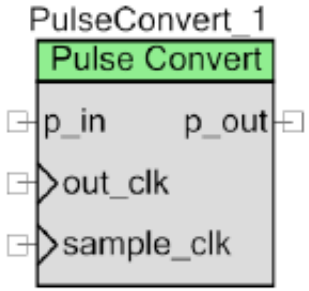
Pseudo Random Sequence (2-64 bits)



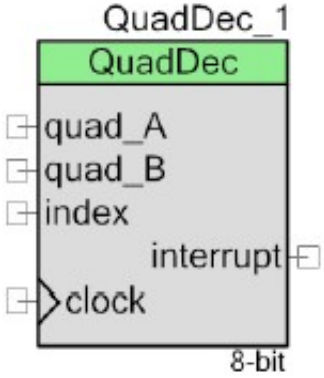
Cyclic Redundancy Check (CRC)



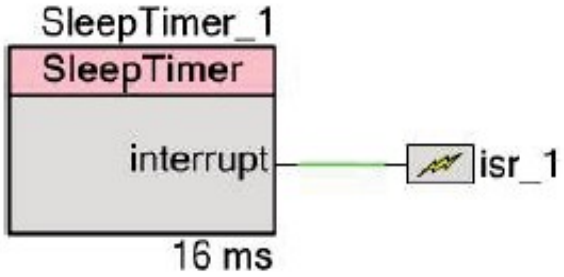
Pulse Converter



Quadrature decoder 8, 16, 32 bits



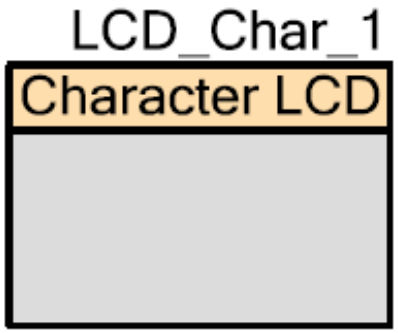
Sleep timer 2-4096 ms



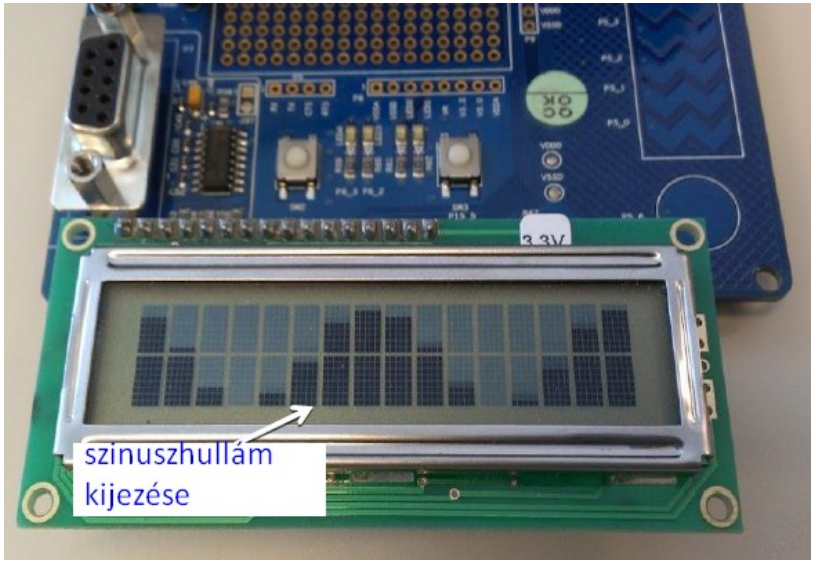
Alfanumerikus LCD vezérlő

Jellemzők:

- Az ipari szabványnak tekinthető Hitachi HD44780 LCD vezérlő protokollját valósítja meg.
- Csak 7 portlábat igényel egy I/O porton (4 bites üzemmód)
- A PSoC Creator beépített eszközt biztosít a felhasználó által definiált karakterek megszerkesztéséhez
- Vízszintes és függőleges oszlopdiagramok megjelenítését is támogatja

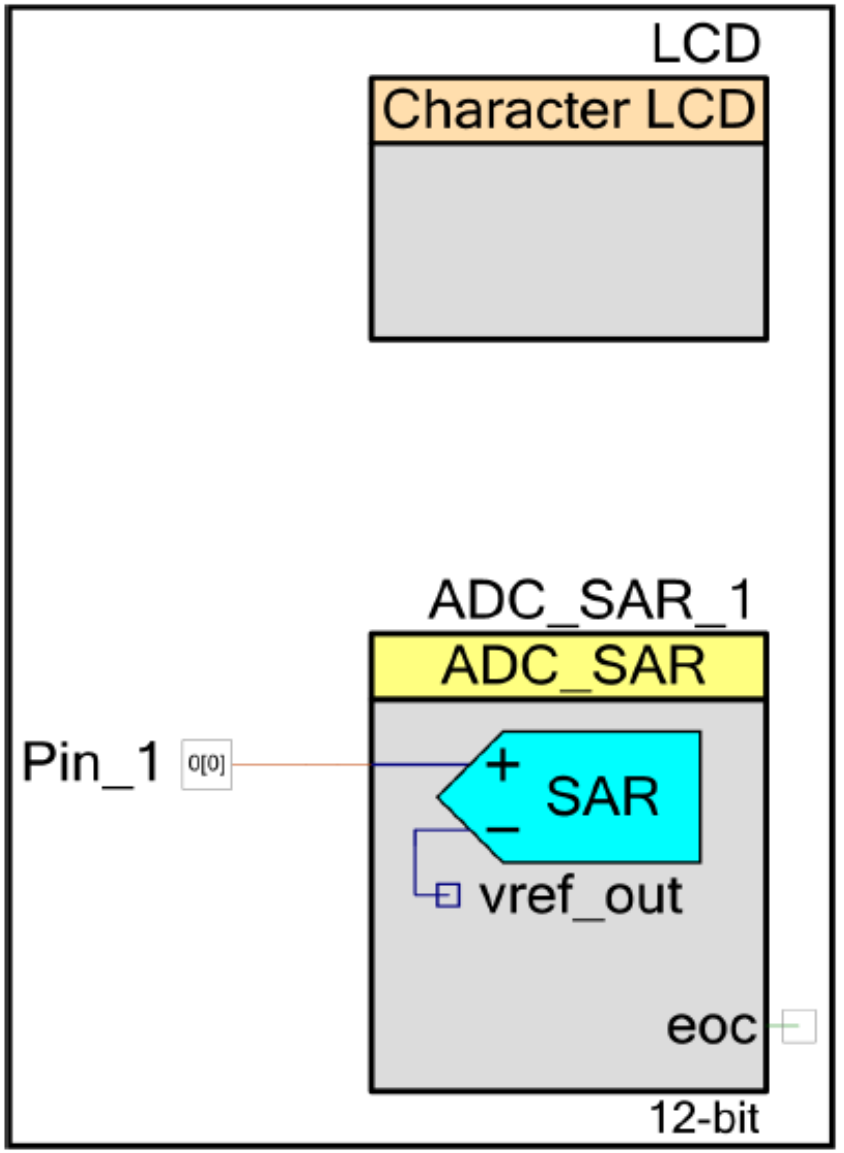



```
#include <device.h>
void main() {
    LCD_Char_Start();
    LCD_Char_PrintString("Hello World");
}
```



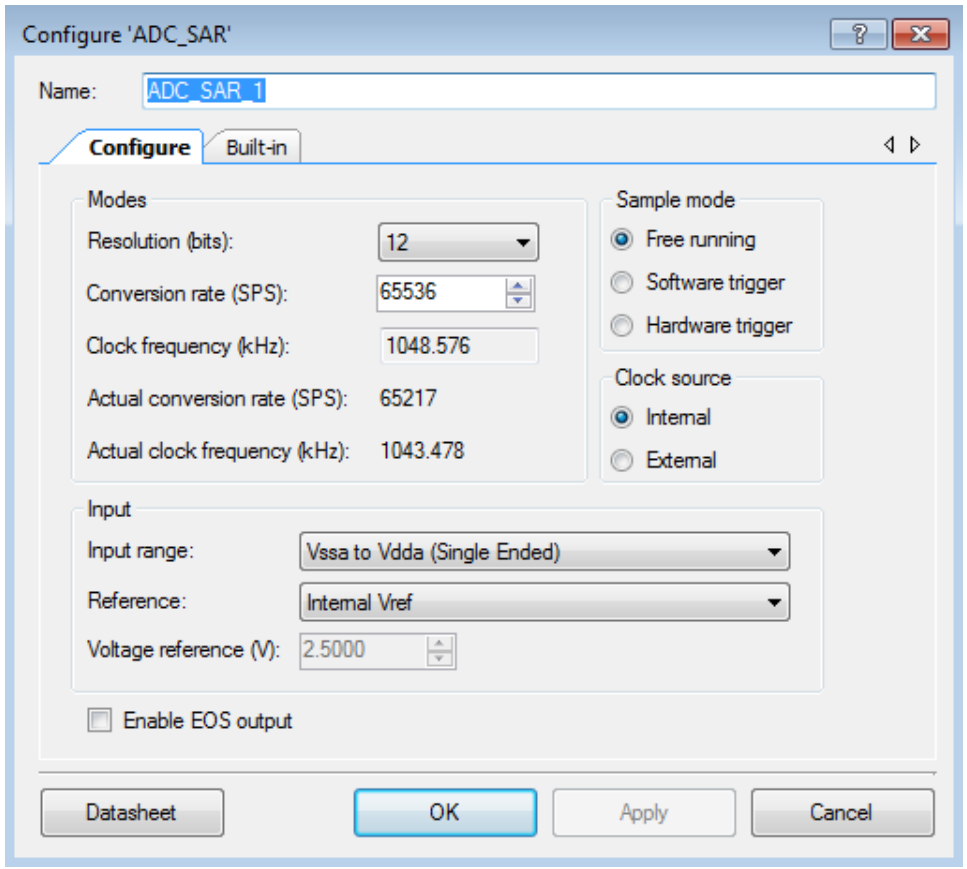
LCD_demo projekt

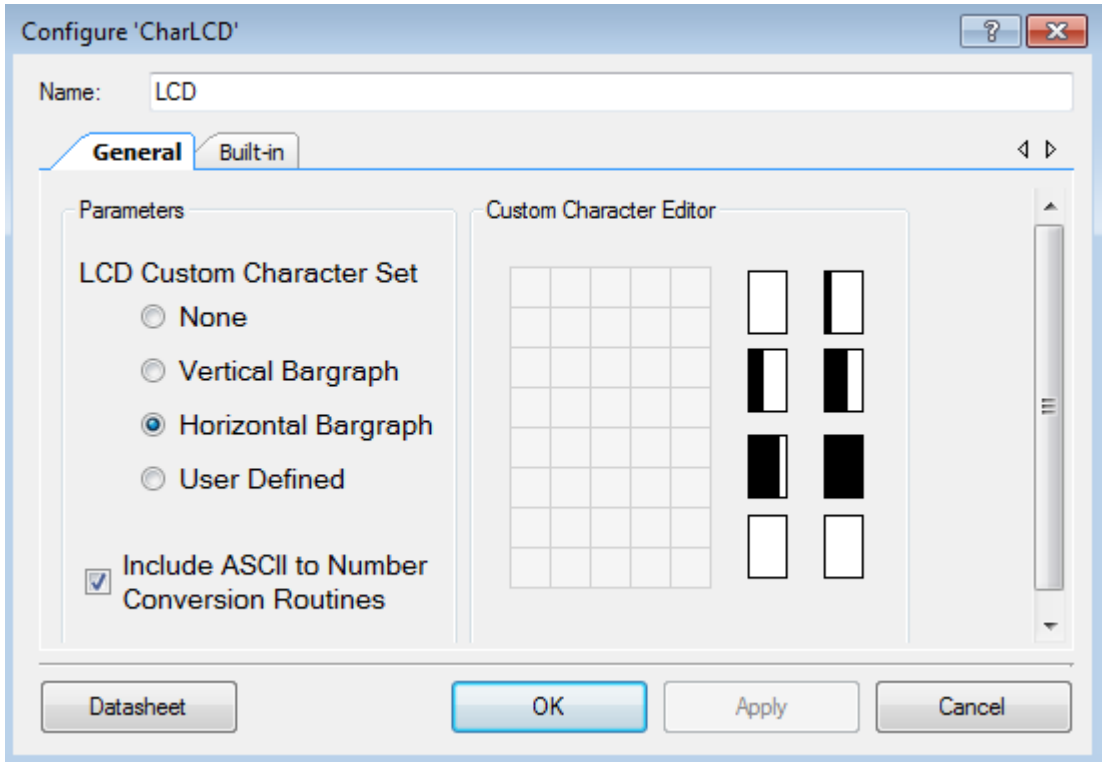
- Egyszerű mintaprogram, amelyben a 12 bites SAR ADC által mért feszültséget numerikusan és grafikusan is kijelezzük egy 16x2 karakteres LCD kijelzőn
- ADC bemenet: P0[0] (0 – 5 V)
- LCD kimenet: P2[7:1]
  P2_1 – LCD_D4
  P2_2 – LCD_D5
  P2_3 – LCD_D6
  P2_4 – LCD_D7
  P2_5 – LCD_E
  P2_6 – LCD_RS
  P2_7 – LCD_RW



LCD_demo projekt – konfigurálás

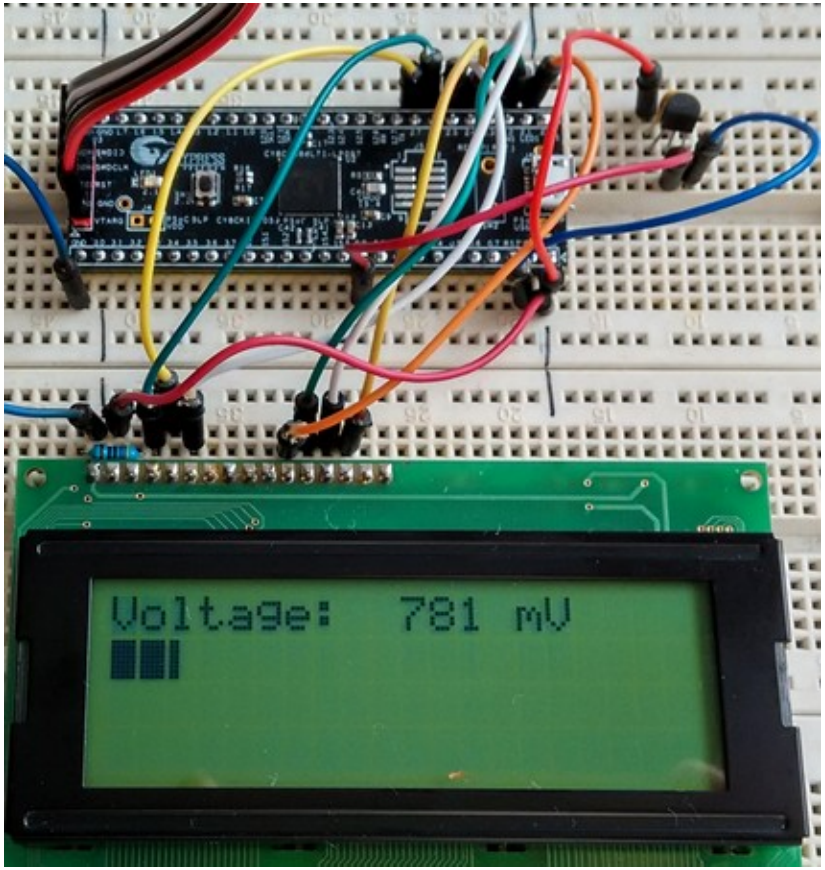
- **SAR ADC konfigurálás:** 12 bites mód, Vssa – Vdda tartomány, internal reference (belső referencia)
- **LCD konfigurálás:** válasszuk a Horizontal Bargraph opciót!



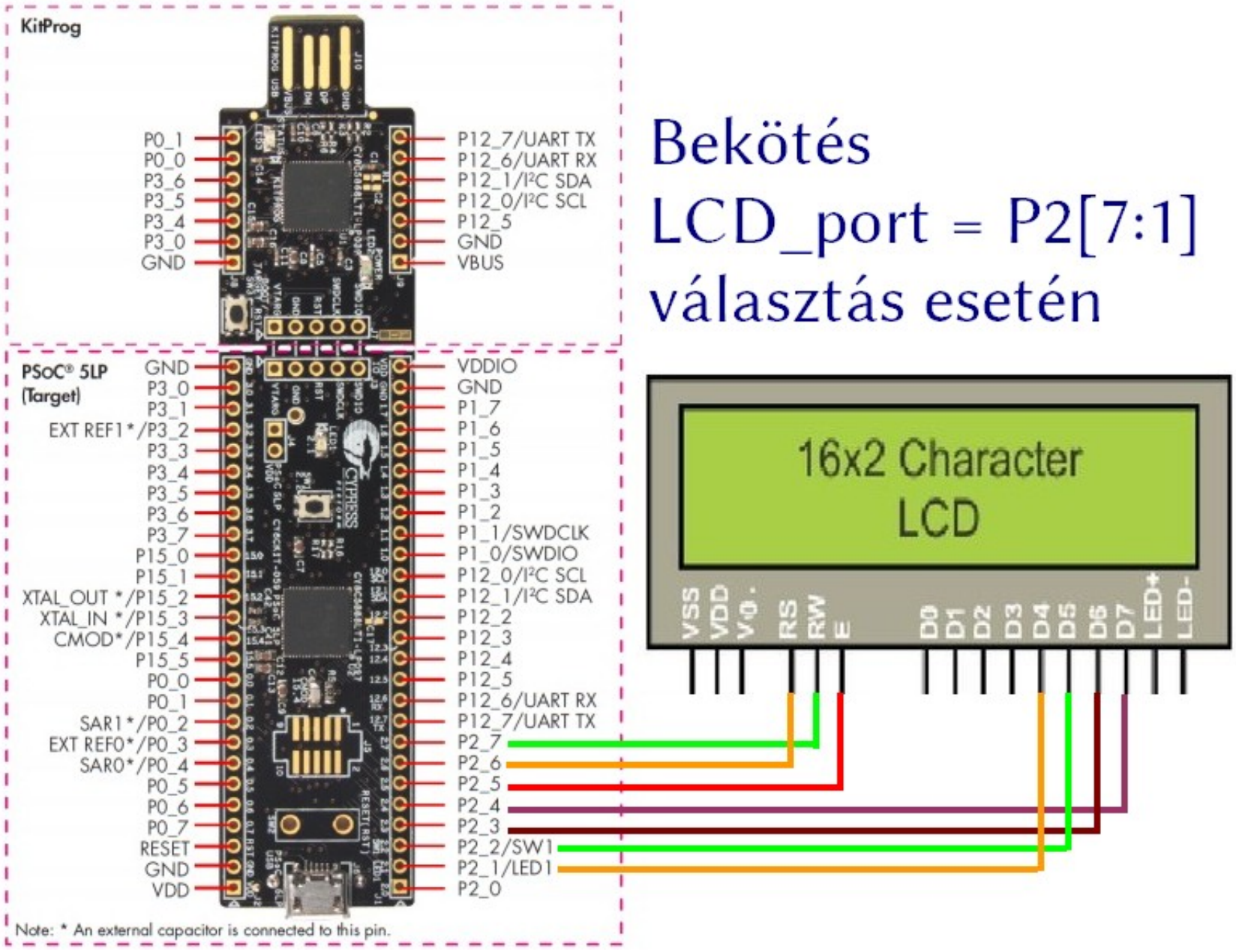


LCD_demo projekt – main.c

```
#include <project.h>
#include "stdio.h"

int main() {
    uint16 result, millivolts, bar;
    char textbuffer[20];
    ADC_SAR_1_Start();
    ADC_SAR_1_StartConvert();

    LCD_Start();
    LCD_LoadCustomFonts(LCD_customFonts);
    LCD_Position(0u, 0u);
    LCD_PrintString("CY8CKIT-059 PSOC");
    LCD_Position(1u, 0u);
    LCD_PrintString("ADC & LCD demo");
    CyDelay(2000u);
    LCD_ClearDisplay();

    for(;;) {
        result = ADC_SAR_1_GetResult16();
        millivolts = ADC_SAR_1_CountsTo_mVolts(result);
        bar = result/52;                    // Scale value to pixels
        sprintf(textbuffer, "Voltage: %4d mV",millivolts);
        LCD_ClearDisplay();
        LCD_PrintString(textbuffer);
        LCD_DrawHorizontalBG(1,0,16,bar);
        CyDelay(200u);
    }
}
```



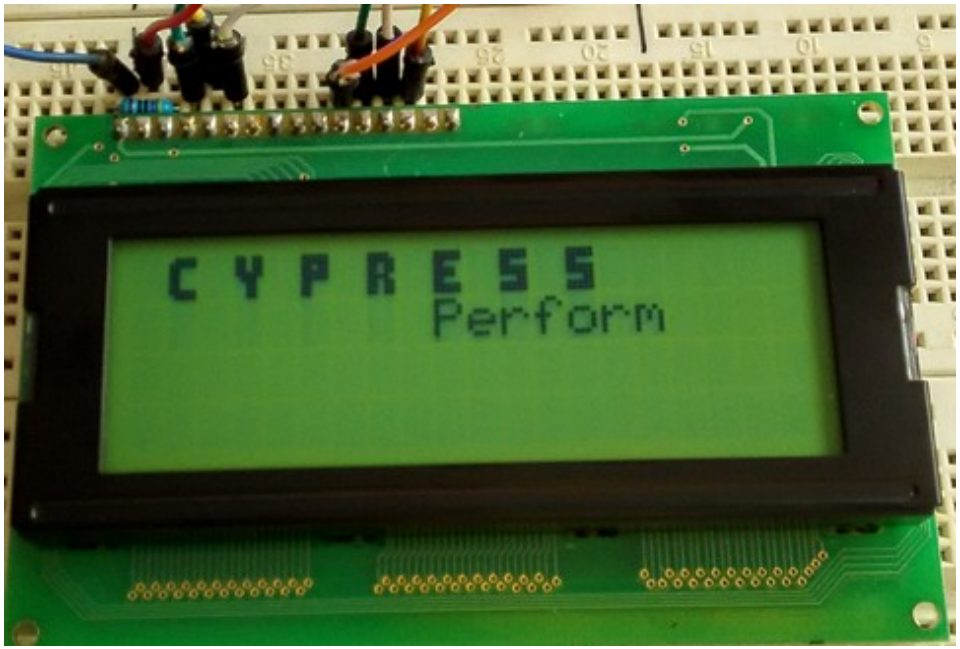
LCD_demo projekt – bekötési vázlat



LCD_test_1 projekt

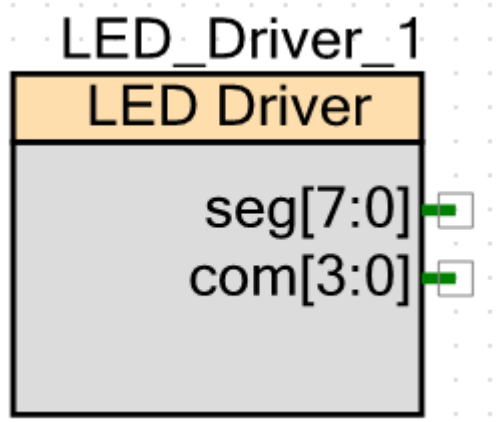
- Az egyszerű szövegkiírást és a felhasználó által definiált karakterek megjelenítését mutatjuk be karakteres LCD kijelzőn
- A mintaprogram eredetijének lelőhelye:
  CE95 290 - Character LCD with Custom Font
- A CY8CKIT-059 kártyára átdolgozott változat lelőhelye:
  Cypress Forum: Interfacing 16x2 LCD with PSoc5LP Kit-059

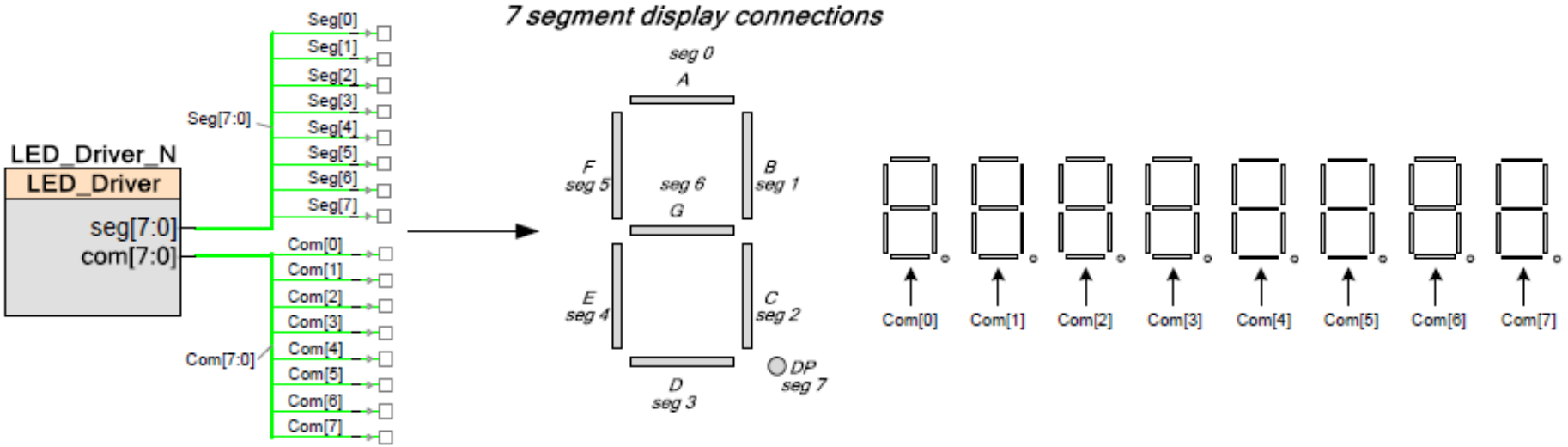
- Technikai okokból itt most a program futásának eredményét egy 20x4-es kijelzőn mutatjuk be
- A kijelző bekötése megegyezik az előző projektnél leírtakkal



Szegmens vagy mátrix kijelző meghajtó

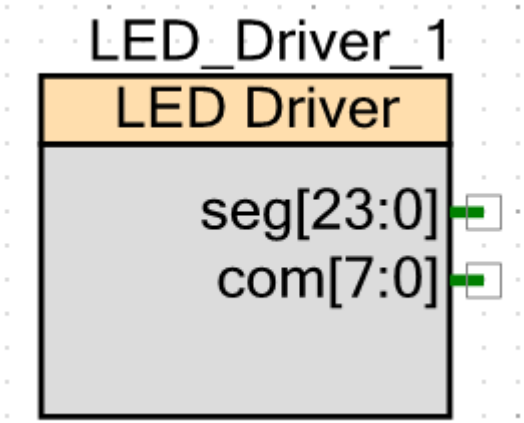
- Akár 8 RGB 7-szegmens számkijelző
- Akár 8 14 vagy or 16-szegmens számkijelző
- Akár 192 LED (8x8 RGB mátrix)
- Aktív magas vagy aktív alacsony vezérlés
- HW multiplexelés (sem CPU idő, sem megszakítás)
- 7-, 14-, vagy 16-szegmens dekódolás szám és karakter kiíráshoz
- Független fényerő vezérlés minden számjegyre/oszlopra

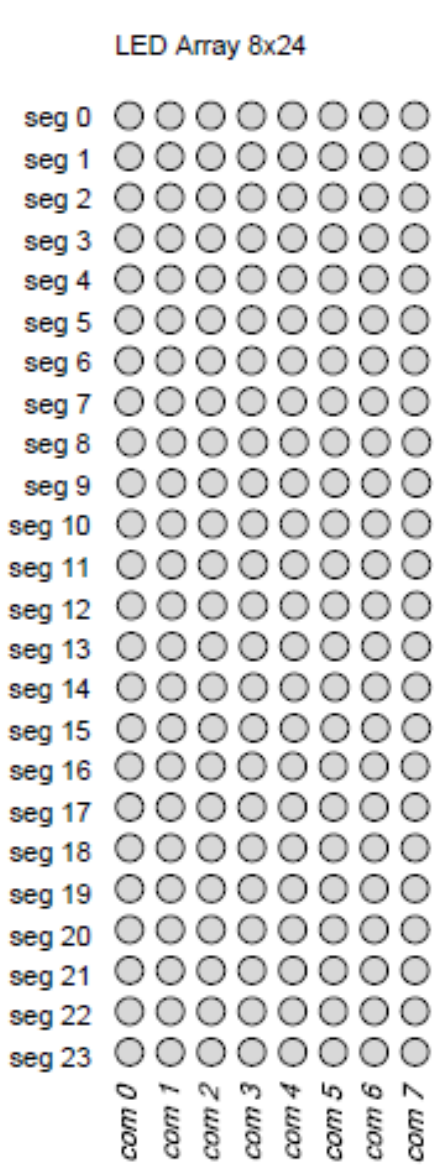


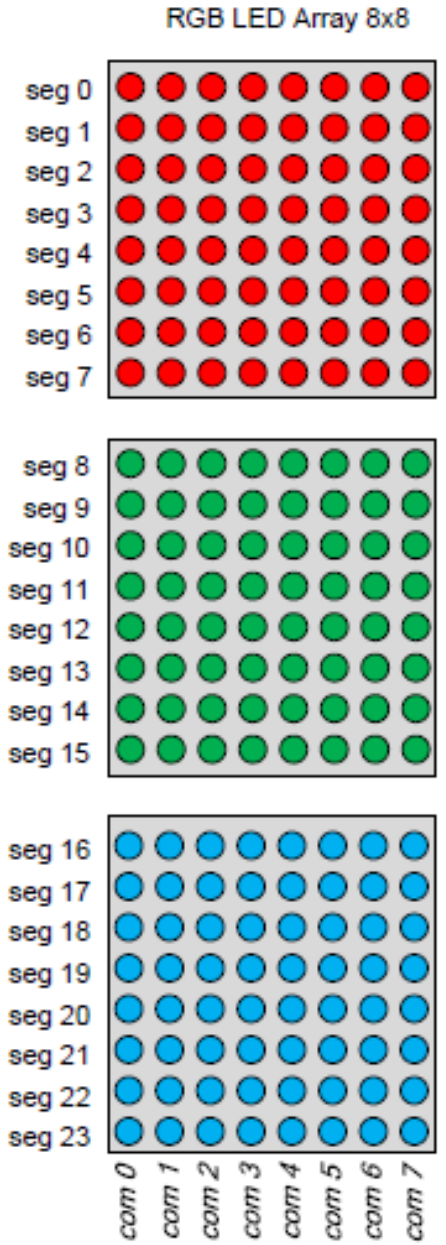


LED mátrix kijelző meghajtó

- Mátrix kijelzés lehetőségei
  - Akár 8x24 monokróm
  - 8x8 RGB mátrix

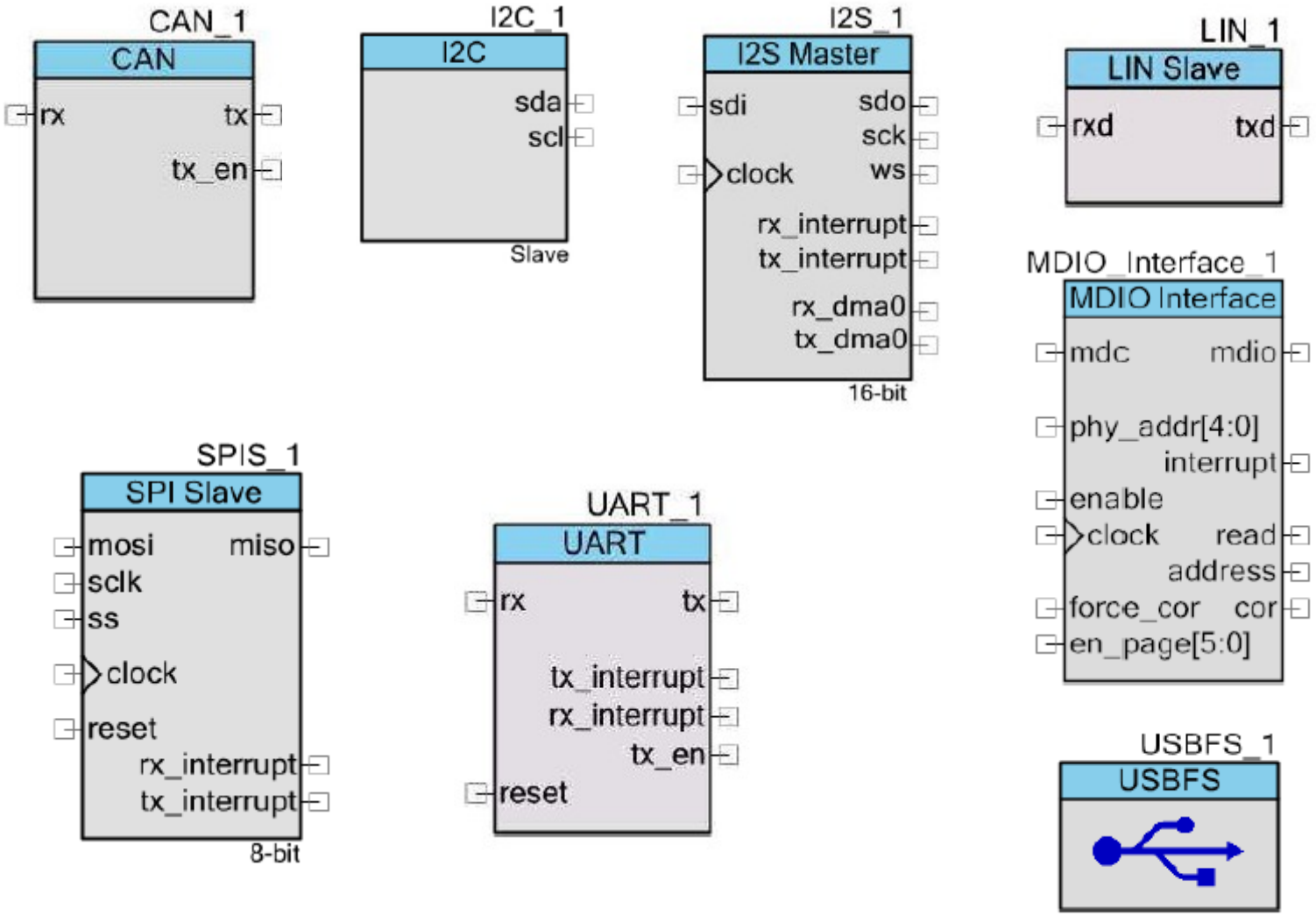






Kommunikációs modulok

- Külső eszközök és a CPU közötti kommunikációt szolgáló modulok



CY8CKIT-059 fejlesztői kártya

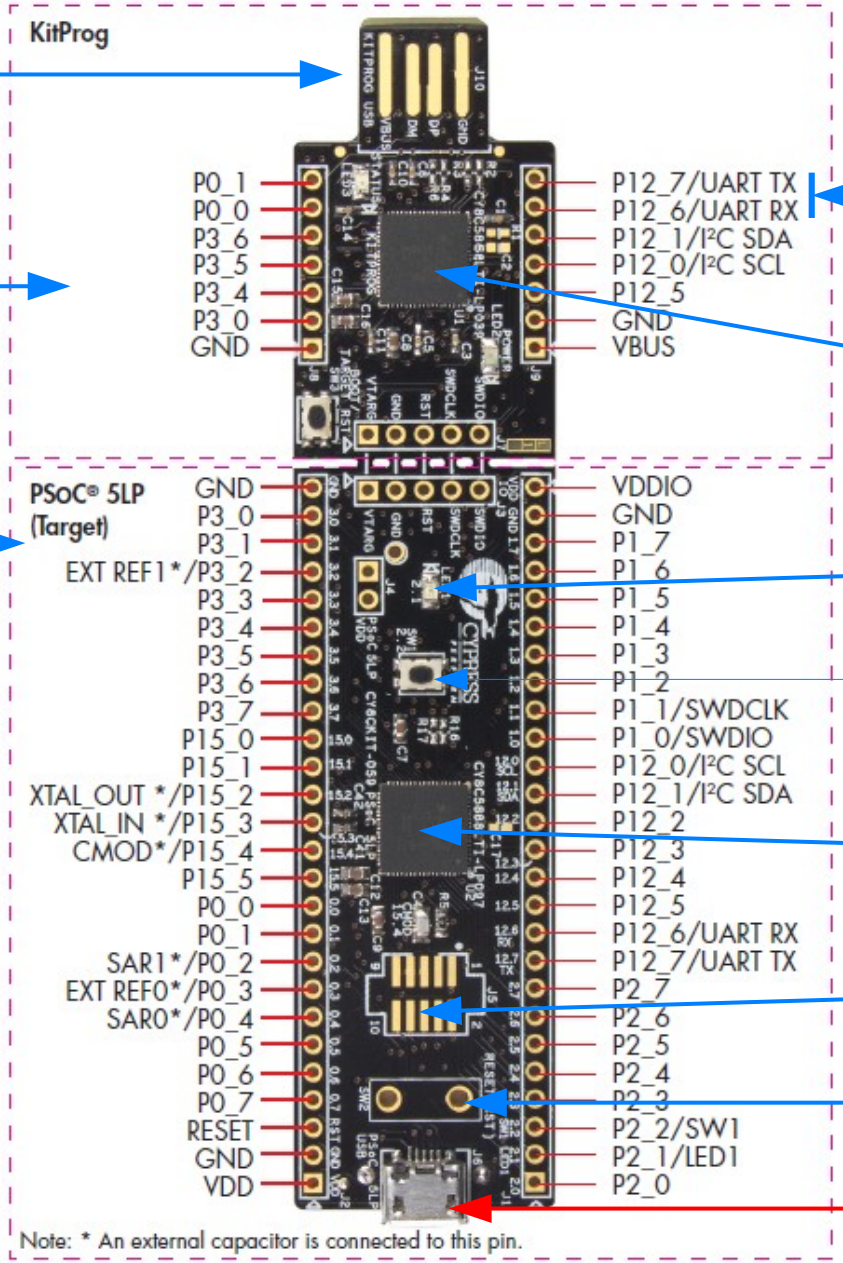
USB csatlakozás a PC-hez

KitProg programozó és hibavadász

PSOC 5LP Target áramkör

A tápellátás történhet a programozó felől (5V), Az alkalmazói USB csatlakozóról (5V), vagy a VDD csatlakozáson keresztül (3,3 – 5 V).

Utóbbi esetben a D1 és D2 diódákat el kell távolítani az USB-re csatlakozás előtt!



USB – UART Kivezetések

C8C5868LTI-LP039

LED1 (2.1 kivezetés)

SW1 (2.2 kivezetés)

CY8C5888LTI-LP097

JTAG csatlakozás

RESET gomb helye

USB alkalmazói csatl.

A céláramkör kapcsolási rajza