

# 22. Analóg jelek feldolgozása, az ADC

Írta: Szomorú-Ozsváth Erzsébet

Lektorálta: Proksa Gábor, Lágler Gergely

## BEVEZETÉS

---

Gyakran szembesülünk a hétköznapi, valós életben olyan problémával, amikor valamilyen fizikai mennyiséget szeretnénk megmérni, például szeretnénk tudni, hogy hány fok van, milyen zajos az utca, milyen színű pontosan egy virágszirom, vagy éppen milyen magas is valaki. Ezek az értékek időben változóak. Ha ezeket az értékeket folyamatosan mérjük, akkor

ezek a mennyiségek analóg jelekkel írhatóak le. Az **analóg jel** egy folytonos fizikai mennyiség, amely folyamatos függvénye az időnek vagyis az analóg jel minden időpillanatban felvesz valamilyen értéket és akár minden időpillanatban más-más lehet ez az érték.

A mért értékeket szeretnénk a felhasználó számára érthető formában kijelezni, tárolni vagy más egységeknek, mint például számítógép, mikroprocesszor, FPGA, stb. további feldolgozás céljából átadni. Mivel az előbb említett eszközök digitális jelekkel működnek, ezért nélkülözhetetlen az analóg jelek feldolgozásához az analóg-digitális átalakítás. Általában az analóg jel egy feszültségérték, amihez egy számot rendelünk, tehát a jelet számmá alakítjuk.

A **digitális jel** egy olyan fizikai jel, amely diszkrét (egy adott egység többszöröse) értékek sorozatából áll, tehát csak véges sok értéket vehet fel.

Az analóg jelek digitális leképezéséhez a gyakorlatban mindig kompromisszumot kell kötnünk, azaz ki kell választanunk a tartományt, amelyben mérni szeretnénk, és a pontosságot. Természetesen a mérés során szeretnénk minél pontosabban megközelíteni a valós értéket, tehát szeretnénk minimalizálni a tényleges és a mért érték közötti különbséget, másszóval a **mérési hibát**.

Nem mindegy azonban, hogy egy ház magasságát szeretnénk megmérni méter pontossággal vagy pedig egy újszülött magasságát szeretnénk megmérni cm pontossággal. Az előbbi esetén nem állunk neki milliméterpapírral méricskélni, míg az utóbbi esetében értelmetlen egy méter hosszúra vágott bottal mérni. A tartomány nagyságából következik majd a mérés pontossága is, vagyis a **felbontás**. Látjuk tehát, hogy a megfelelő mérőeszköz kiválasztásához fontos a lehető legpontosabban ismerni a mérendő jelet.

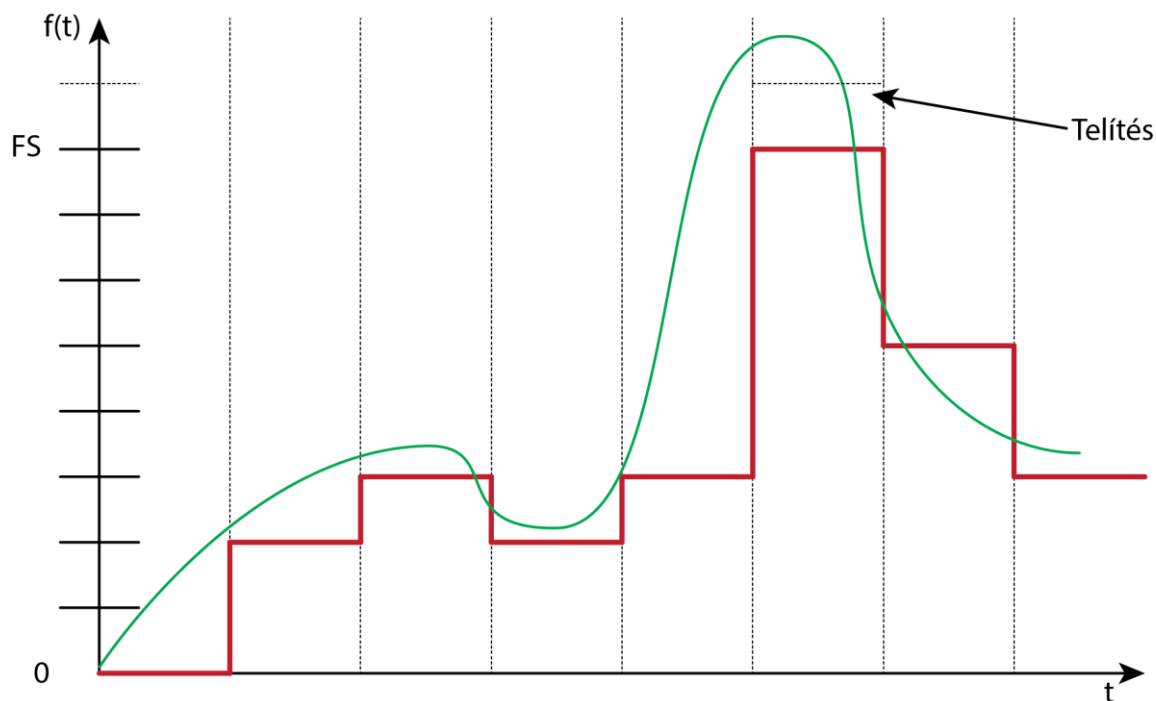
Ha például egy ember magasságát szeretnénk megmérni centiméteres osztású mérőszalaggal, akkor a legnagyobb mérési hiba, amit véthetünk (tegyük fel, hogy pontos, akkurátus mérést végzünk), mérőeszközünk pontatlanságából adódóan mm-es nagyságrendbe esik (holott tudjuk, hogy az ember folyamatosan nő, nem pedig centinként vagy miliméterenként), hiszen azt a nagyságrendet már nem tudjuk a szalagról leolvasni.

Egy másik példa lehet, hogy a hőmérséklet emelkedését egy hagyományos kültéri hőmérőn látjuk, hogy folyamatosan kúszik felfele a hőmérő higanyszála, mégis a beosztások miatt szabad szemmel számunkra a hőmérséklet leolvasása Celsius-fok vagy tized Celsius-fok pontossággal lehetséges. A folyamatos mennyiséget megpróbáljuk diszkrét értékkel jellemezni, azaz **kvantálni**.

Ahogy az előbbi példákban is látszott, sok jel nem érhető el villamos jel formájában, ilyenkor a megfelelő jelátalakítók, szenzorok segítségével villamos jellé kell alakítani (lásd részletesebben a szenzoros tananyagrészen). Ez a folytonos villamos jel kerül **analóg-digitális átalakító** (Analog-Digital Converter, továbbiakban ADC) bemenetére. Általában ez a jeltartomány csak pozitív lehet, de léteznek bipoláris átalakítók is, amelyek negatív tartományban is működnek.

## AZ ANALÓG-DIGITÁLIS ÁTALAKÍTÓ

Az ADC egy olyan átalakító, melynek bemenetére egy folytonos villamos jelet kötünk, kimenetén pedig az adott mintavételi időpontban mért érték digitális megfelelője jelenik meg. Az alábbi ábrán zölddel az ADC bemenetére kötött jel, pirossal pedig a kimenetén megjelenő jel látható. Az x-tengelyen az idő, az y-tengelyen az érték van ábrázolva. A maximális érték bejelölése után a tartományt felosztjuk 8 egyenlő részre. A mérési tartomány a legnagyobb és a legkisebb mérendő jel értéke közötti tartomány, ahol a mérést egy bizonyos pontossággal akarjuk elvégezni. Ezután a digitális kimenet azt mutatja meg, hogy a mért jel melyik ilyen kis tartományba, egységbe esik. Ha az érték például 3, az azt jelenti, hogy a kimenet a 3. egységben van.



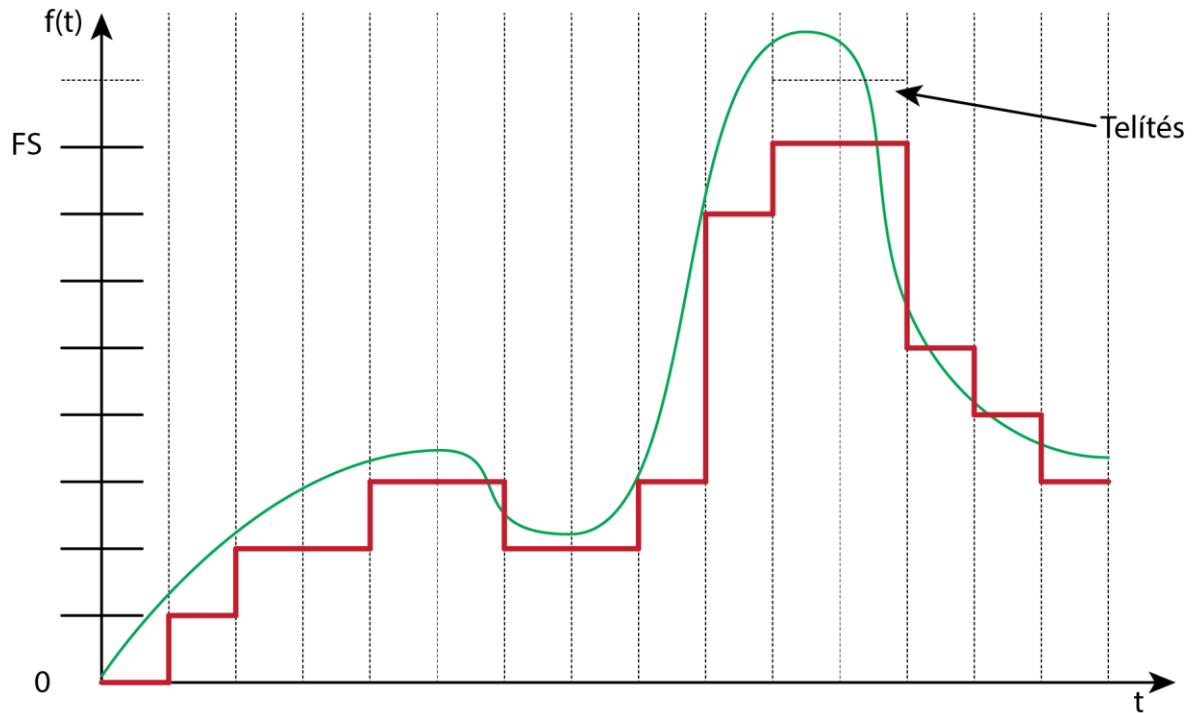
1. ábra - ADC bemeneti (zölddel) és kimeneti jele (pirossal)

Ez a példa többféle problémára is felhívja a figyelmet. Először is arra, hogy a kapott jelünk nem alakhű. Az 1. ábrán látszik, hogy a bemeneti jelünk amplitúdója nagyobb, mint a választott mérési tartomány, tehát a kimeneti jel **“betelített”** vagyis a lehető legmagasabb értéknél állt meg. Ez a jelenség azt mutatja, hogy nem megfelelő a kiválasztott ADC **kivezérlési tartománya** (Full Scale, FS), azaz az a bemeneti analóg tartomány, amelyben az átalakító telítés nélkül működik.

Míg az **unipoláris** ADC-k a referenciafeszültség és a földpotenciál között működnek, addig a **bipolárisak** a földpotenciál körül szimmetrikus tartományban.

A méréseink során használt Atmel mikrovezérlőkben is van ADC modul. Egyes lábain mérni, digitalizálni tudja az ott lévő feszültséget. A mikrokontroller adatlapjából indirekt kiderül, hogy unipoláris, mivel van egy "AREF" nevű lába, amely az ADC analóg referenciafeszültség bemenet lába, és erre a lábra az adatlapnak megfelelően csak pozitív érték kerülhet.

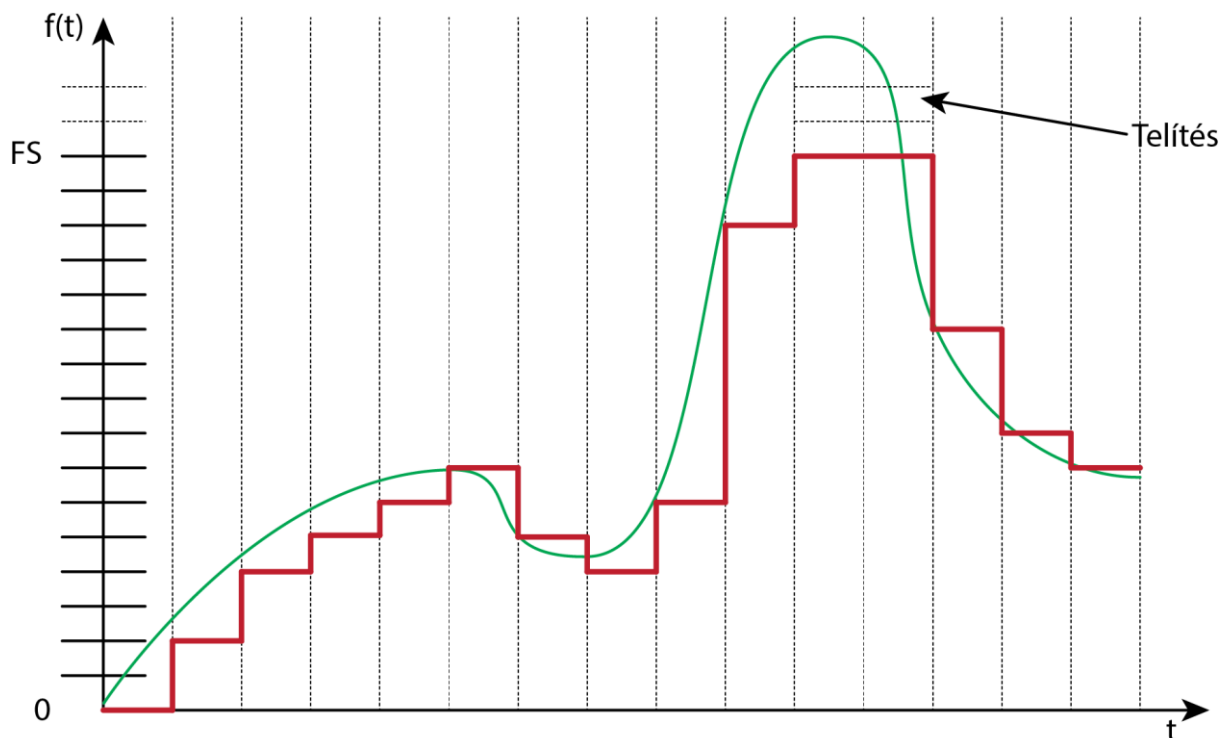
Az alakhússágot javíthatnánk, ha például gyakrabban mérnénk, tehát növelnénk a **mintavételi frekvenciát**, ezzel fordítottan arányosan csökkenne a mintavételi idő. A magasabb mintavételi frekvenciával készült kép a 2. ábrán látható.



*2. ábra - Növeljük a mintavételi frekvenciát*

Az ADC működéséről, így a mintavételi frekvencia állításának mikéntjéről is olvashatunk ennek a tananyagrésznek a második felében.

Az alakhúség javításának másik módja, hogyha kisebbre vesszük az értéktengelyen a lépésközöket, lásd 3. ábra.



3. ábra - Növeljük a felbontást

A lépésközök számát nevezzük **felbontásnak**, amely megmutatja, hogy az eszköz a bemenő analóg jelet hány különböző értéké tudja átalakítani. Ezt **bit-számban** szokták megadni, ami azt jelenti, hogy a mi példánkban adott 10 bites ADC **felbontása**  $2^{10}$ -en, azaz ez az ADC 1024 részre képes bontani a mérési tartományt. Ha tehát a referencia feszültségünk  $X$ , akkor  $X/1024$  V-os pontossággal tudunk mérni. Ezzel el is érteztünk a következő fontos fogalomhoz, az **LSB**-hez (Least Significant Bit, legkisebb helyiértékű bit). Az LSB megmutatja, hogy mekkora az a legkisebb átlagos eltérés a bemeneten, amely a kimenet legkisebb helyiértékű bit-jének a megváltoztatásához szükséges, vagyis hogy mekkora egy digitális lépcsőhöz tartozó feszültség. Kiszámítása a következő képlettel történik:  $LSB = KT / (2^n - 1)$ , ahol  $KT$  a kivezérlési tartomány,  $n$  a bitszám.

### Számítás

**Számítsuk ki, hogy 5 V-os referencia feszültség mellett mekkora a 10 bit-es ADC-nk LSB, azaz a legkisebb helyiértékű bit értéke.**

A kivezérlési tartományunk a 0-5V-ig terjed, tehát 5V-os, a bitszám pedig 10. Tehát az  $LSB = 5V / (2^{10} - 1) = 0,00489 \text{ V} = 4,89 \text{ mV}$

**Mekkora lenne a felbontás, ha a 10-ből csak 8 bit-es felbontást használnánk?**

Az előbbi példából jól látható, hogy az ADC helyes kiválasztása alapos körütekintést igényel. Valós életben egy mérnöknek mindig oda kell figyelnie az árra is. Minél pontosabb, minél nagyobb felbontású egy analóg-digitális átalakító, annál drágább. Gondoljunk csak bele, hogy millió nagyságrendben legyártott számítógéppalkatrész árának drágulása a gyártási összköltséget drágulás milliónak költségével emeli

meg. Ebből kifolyólag sorozatgyártásra szánt eszközöknél nem szoktak lényegesen **túlméretezni** (valamiből jobbat, pontosabbat, erősebbet választani). Sajnos az érzékeny vásárlás egyenes következménye, hogy a gyártók sokszor csak a minimális elvárásoknak akarnak megfelelni, nem törekednek a biztonsági tartalékokra, részben ennek köszönhetően az elektronikus eszközök gyakran pár éven belül tönkremennek és sajnos eszeveszett mértékben termeljük a szemetet.

## MŰKÖDÉSI MÓDOK

Nézzük meg az ADC-k különböző működési módjait, különös tekintettel az ATmega16A mikrokontrollerben található ADC-re (lásd részletesebben az adatlap 22. "Analog to Digital Converter" című fejezetét).

### ASZIMMETRIKUS MŰKÖDÉSI MÓD

Az **aszimmetrikus** működési mód azt jelenti, hogy a bemeneti csatornán (lábon) megjelenő feszültség a referenciafeszültséghez képest értendő. Ekkor a bemeneti lábak mindegyikére külön jeleket kapcsolhatunk. A tananyagrészen használt mérésekhez ezt az üzemmódot fogjuk használni.

### SZIMMETRIKUS/DIFFERENCIÁLIS MŰKÖDÉSI MÓD

A másik üzemmód a **differentiális vagy szimmetrikus üzemmód**, amikor a bemeneti jelek egy vezeték helyett két vezetéken futnak. Ez különösen a nagy frekvenciás jeleknél ( $> 1 - 10$  MHz) fontos a nemkívánt zajok elkerülése érdekében. Ennek a következménye viszont az, hogy nem tudjuk mindegyik vezetékét egyenként kihasználni, hanem csak páronként. Ha ennél több differentiális csatornára van szükség, akkor másik típusú vagy tokozású mikrokontrollert vagy külső ADC-t kell választani.

### Kitekintés

A vezetékekben a külső mágneses és elektromos tér hatására már kis távolságnál is viszonylag nagy zavarjelek keletkezhetnek. A vevő oldalon egy bizonyos nagyságú zaj mellett nehéz kiszűrni, hogy mi is volt az eredeti jel. Ezt elkerülendő a zajok kiküszöbölésének sokféle módja létezik.

#### Hardveres zajsűrés

Az egyik leggyakrabban használt a vezetékek árnyékolása. Ekkor a vezeték köré egy fém köpenyt húznak, ami Faraday-kalickaként hat a vezetékekre. Ilyen vezeték például közismert néven a coax(koaxiális) kábel.

Fontos zajkiküszöbölő módszer a már fent említett szimmetrikus jelvezetés is.

Csavart érpár két egymásra spirálisan feltekert vezetékből áll. Lényege, hogy a vezetékpárok jelei ne hassanak egymásra. Telefonkábeleknél, internetkábeleknél használjuk. Két fajtája van: a nem árnyékolt csavart érpár (UTP - Unshielded Twisted Pair) és az árnyékolt csavart érpárt (STP - Shielded Twisted Pair). Az árnyékolt már külső zavarjelektől is véd.

A nyomtatott áramkörök esetében külön földrétegeket szoktak a nagyfrekvenciás jelvezeték-rétegek közé tervezni, hogy a jelek ne hassanak egymásra. A vezetékek egymásra hatása tovább csökkenthető, hogyha a vezetékek a különböző rétegekben egymásra merőlegesek.

### Szoftveres zavarjelszűrés

Leggyakoribb szoftveres zajszűrés mód az átlagolás. Ilyenkor a szükségesnél több mérést végzünk, és a mért értékeket átlagoljuk:

$$a = \sum \frac{a_n}{n}$$

## ADC HASZNÁLATA A MIKROVEZÉRLŐN

Lássuk a mikrokontrollerbe beépített ADC számunkra releváns főbb jellemzőit:

- 8 csatornás aszimmetrikus üzemmódban
- 7 differenciális bemeneti csatorna szimmetrikus üzemmódban
- szukcesszív approximációs (magyarázatát lásd a kitekintésben)
- 10 bites
- 0-Vref feszültségig tartó mérési tartomány
- interrupthívás a konverzió elkészültekor
- maximum 15 ksps mintavételi frekvencia maximális felbontásnál

Ennél a mikrokontrollernél a differenciális üzemmód úgy valósítható meg, hogy a az ADC1 vezetékre tesszük a negatív jelet, a további 7 csatorna egyikére pedig a másikat. Viszont ez az üzemmód csak a TQFP tokozású mikrokontrollernél elérhető, a DIP tokosnál nem. A mi példamérésünkben azonban nincs szükség nagy zajimmunitásra, ahogy ez az alacsony frekvenciás mérésekben szokásos, használhatjuk az asszimmetrikus üzemmódot. Részletesebb magyarázathoz olvasd az adatlap 22. fejezetét.

Az ADC a pontos működéséhez 50 - 200 kHz közötti órajelet igényel. Amennyiben nincs szükség a 10 bites felbontásra, akkor az ADC órajele 200 kHz felett is lehet. Mi is megelégszünk csak a 8 bit-es felbontással. A jelátalakítás 13 ADC órajelet vesz igénybe (kivéve az első konverziónál, mert ott 25-öt).

### Számítás

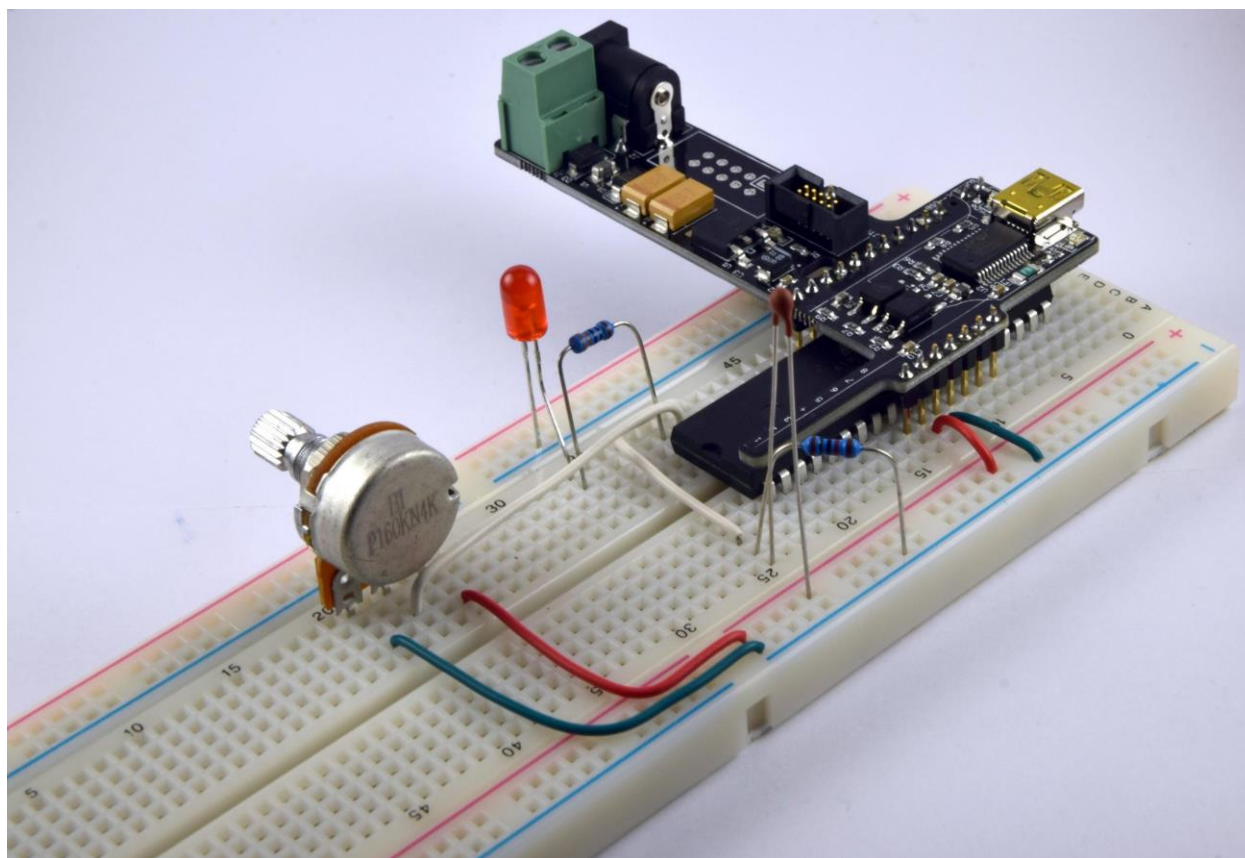
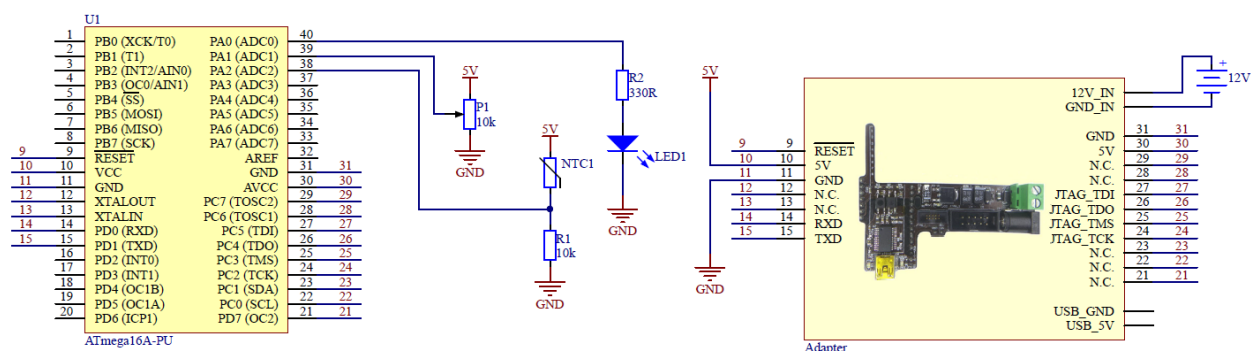
**Számítsuk ki, hogy mekkora maximális mintavételi sebességet lehet elérni 10 bit-es módban.**

A legnagyobb sebességet a lehető legnagyobb órajel mellett érhetjük el, ezért  $f = 200$  kHz-et választjuk órajelnek. Egy konverzió 13 órajelperiódus alatt zajlik le, tehát  $200/3 = 15,4$  kilo-sample per second-os (ksps) maximális mintavételi sebességet lehet elérni. Így egyúttal ellenőriztük is az adatlap által megadott értéket.

A példáinkban mikrokontroller órajelének előállításához a beépített RC oszcillátort használjuk. A működési frekvenciát ebben az üzemmódban a CKSEL3:0 lábakkal állíthatjuk be az adatlap 8.8 táblázata szerint. Mi 8 MHz-es órajelet szeretnénk beállítani, amit a CKSEL3:0 lábak 0100-ba állításával érünk el. Ahhoz, hogy az ADC számára megfelelő 50-200 kHz közötti órajelet előállítsuk, a mikrokontroller 8MHz-es órajelét leosztjuk, az úgynevezett prescaler segítségével (lásd később az ADCEnable() függvényt), 128-cal, ami 62,5 kHz-es órajelet jelent. Ebből a mintavételi sebesség kb. 4,8 ksps.

## MÉRÉS 1

Állítsuk össze a következő kapcsolást.



4. ábra - Mérési elrendezés

A tökéletes megértéshez érdemes az adatlapot párhuzamosan a megoldáshoz olvasni.

## FELADAT

A mikrokontroller PA1 (ADC1) lábán levő potenciométer feszültségét hasonlítsuk össze az NTC-vel létrehozott feszültségosztó feszültségével. A méréshez használjunk AD konverziót. Ha az NTC feszültsége nagyobb, mint a potenciométeren mérhető, akkor a PA0 lábon levő LED gyulladjon ki, ellenkező esetben ne világítson.

## MEGVALÓSÍTÁS

### Az IO port-ok inicializálása

```
void IOInit()
{
    //PORTA felhúzóellenállás kikapcsolása a bemeneteken
    PORTA = 0x00;
    DDRA = 0x00;
    sbi(DDRA, 0); //A LED a PA0-n van, PORTA0 legyen kimenet (0x01)

    //PORTB felhúzóellenállás kikapcsolása és minden pin bemenet
    PORTB = 0x00;
    DDRB = 0x00;

    //PORTC felhúzóellenállás kikapcsolása és minden pin bemenet
    PORTC = 0x00;
    DDRC = 0x00;

    //PORTD felhúzóellenállás kikapcsolása és minden pin bemenet
    PORTD = 0x00;
    DDRD = 0x00;
}
```

Első lépésben definiáljuk az IO port-ok működési módját, hogy be- vagy kimenetként fognak viselkedni, továbbá adunk nekik kiindulási értéket, szakkifejezéssel inicializáljuk őket, lásd IOInit() függvény az IO.c file-ban.

A PORTA, PORTB, PORTC, PORTD adatregisztereken kikapcsoljuk a felhúzóellenállásokat és bemenetnek definiáljuk a lábukat, kivéve a PORTA-n a PA0 lábát, ez kimenet, mert ez vezérli az állapotledet.

### Az órajel inicializálása

```
void TimerInit()
{
    //1x előosztás
    cbi(TCCR1B, CS12);
    cbi(TCCR1B, CS11);
    sbi(TCCR1B, CS10);

    //Timer1 interrupt be
    sbi(TIMSK, TOIE1);

    //Globális interrupt be
    sei();
}
```

Az órajel inicializálása a timer.c file-ban a timerInit() függvény segítségével történik. A TCCR1B (Timer Counter Control Register) az órajelvezérlő beállításokat tartalmazza. A regiszter bit-jeinek leírása az adatlap 16.11.2 fejezetében található.

A 16-6 táblázatból kiolvasható, hogy ha nincs szükségünk előosztóra, akkor a CS12=0, CS11=0, CS10=1- be



kell állítani a sorrendnek megfelelően. A regiszteren belüli többi bit értékét nem állítjuk, nincs rájuk szükségünk, értékük marad az alapértelmezett 0.

A 16.11.7 fejezetben leírtaknak megfelelően a Timer1 interruptját a TIMSK (Timer/Counter Interrupt Mask Register) regiszter, az órajel interruptjának beállításáért felelős regiszter TOIE1 lábának beállításával engedélyezzük. Végül engedélyezzük a globális megszakítást (globális interrupt-ot).

#### ADC inicializálása

```
void ADCInit()
{
    //AVCC referencia
    cbi(ADMUX, REFS1);
    sbi(ADMUX, REFS0);

    //ADC1 bemenet
    sbi(ADMUX, MUX0);
    ADC_state = poti;

    //Balra tolt eredmény -> felső 8 bit ADCRH-ban
    sbi(ADMUX, ADLAR);
}
```

Az ADC inicializálásához az ADCInit() függvényben először is be kell állítani a 22.9.1 fejezet alapján az ADMUX (ADC Selection Multiplexer Register) regiszter segítségével az ADC referencia feszültségét, mely esetünkben a külső AVCC feszültségével egyezik meg (REFS1=0,REFS0=1).

A 22.4 táblázat segítségével kiválasztjuk a beállítandó regiszterbiteket, hogy az egyes csatorna (ADC1) legyen a bemenet. Ehhez a MUX0-t kell 1-re állítani, a maradék MUX bitek maradhatnak az alapértelmezett 0 értéken, őket nem kell állítani.

ADC\_state állapotflag-et "poti"-ra állítjuk. Ez a változó fontos, hogy volatile legyen, mert csak az interrupt kezelő függvényben kerül módosításra és mivel oda nem kerül normál esetben a futás, a fordító a változót egyébként kioptimalizálja.

Az ADC konverzió eredménye az ADCH és az ADCL (ADC data register high és low) regiszterekbe kerül.

Az alábbi ábrán látható, hogy hogyan módosítja a konverzió eredményének elmentését az ADMUX regiszter ADLAR bit-je. Mi az egyszerűség kedvéért most megelégszünk a felső 8 bit-tel.

## The ADC Data Register – ADCL and ADCH

ADLAR = 0

Bit	15	14	13	12	11	10	9	8	
	–	–	–	–	–	–	ADC9	ADC8	ADCH
	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	ADCL
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

ADLAR = 1

Bit	15	14	13	12	11	10	9	8	
	ADC9	ADC8	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADCH
	ADC1	ADC0	–	–	–	–	–	–	ADCL
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

Forrás: ATmega16A adatlap 22.9 fejezet

5. ábra - ATmega16A ADC adat regiszterei

Emiatt praktikusabb az ADLAR = 1 módot használni, így az eredmény balra igazítva jelenik meg az ADCH regiszterben.

## ADC bekapcsolása

```
void ADCEnable()
{
    //128-as presc
    sbi(ADCSRA, ADPS2);
    sbi(ADCSRA, ADPS1);
    sbi(ADCSRA, ADPS0);

    //Interrupt bekapcsolása
    sbi(ADCSRA, ADIE);

    //AD konverter bekapcsolása
    sbi(ADCSRA, ADEN);
}
```

Az ADC bekapcsolása az ADCEnable() függvénnyel történik. Ebben a függvényben beállítjuk az ADC vezérlő és státusz bit-jeit az adatlap 22.9.2. fejezete alapján. Az előosztó (prescaler) 128-as előosztásra való beállítása a 22-5 Táblázat szerint az ADPS0, ADPS1, ADPS2 bit-jeinek 1-be való állításával történik. Engedélyezzük az interrupt-ot és az ADC-t.

```
while (1)
{
    //Ha az NTC osztó feszültsége nagyobb a potiénál,
    // akkor világítson a LED
    if (U_NTC > U_poti)
```

```

    {
        sbi(PORTA, 0);
    }

    //Egyébként ne világítson
    else
    {
        cbi(PORTA, 0);
    }
}

```

A fő ciklusban még elindítunk egy végtelen ciklust, amely a LED-et irányítja. Ha a potenciómter feszültsége nagyobb, mint az NTC osztó feszültsége, akkor a világítson a LED, egyébként ne.

```

ISR(TIMER1_OVF_vect)
{
    //Növelem a leosztó számlálót, ha még nem érte el a maximumot
    if (cntr < CNTR_MAX)
    {
        cntr++;
    }

    //Ha elérte
    else
    {
        cntr = 0;
        //AD konverzió indul (mintavétel)
        ADCStart();
    }
}

```

A megszakítás az ISR(TIMER1\_OVF\_vect) függvényben van implementálva (megvalósítva). A cntr számláló értékét addig növeljük, míg el nem éri a maximális értéket (CNTR\_MAX), amely egy 10 értékű állandóként (konstansként) van definiálva. Ha a számláló elérte ezt az értéket, akkor 0-ba állítjuk és meghívjuk az ADCStart() függvényt, amely nem csinál mást, mint az ADCSRA regiszter ADSC bitjét állítja be vagyis elindítja az ADC konverziót. A timer beállításai és a CNTR\_MAX értéke is lehetne más érték, viszont akkor ritkábban vagy gyakrabban indítanánk el a konverziót.

```

ISR(ADC_vect)
{
    //Az AD csatorna alapján a megfelelő változóba menti a mért feszültséget
    switch (ADC_state)
    {
        case poti:
            U_poti = ADCH * ADC_CONST;
            break;

        case NTC:
            U_NTC = ADCH * ADC_CONST;
            break;
    }
}

```

```

default:
    break;
}

//Váltás a következő AD csatornára
NextCH();
}

```

Ha a konverzió készen van, az meghív egy másik interrupt függvényt, az ISR (ADC\_vect) függvényt. Ebben a függvényben az ADC\_state állapot flag értékétől függően vagy a potenciométeren vagy az NTC-n mért feszültség értékét mentjük a megfelelő változóba. A feszültségértéket úgy kapjuk meg, hogy az ADC konverzió eredményét megszorozzuk a felbontással, ami a mi esetünkben 5 V-os tartomány felosztása 256 részre (csak az ADC felső 8 bitjét használjuk), lásd a main.h-ban definiált ADC\_CONST értékét. Ez 20 mV-os felbontásnak felel meg.

Ezután váltunk a következő csatornára a NextCH() függvény segítségével.

```

void NextCH()
{
    switch (ADC_state)
    {
        //ADC1 lábön a potenciométer van
        case poti:
            //A következő láb az ADC2, amin az NTC osztó van
            cbi(ADMUX, MUX0);
            sbi(ADMUX, MUX1);
            ADC_state = NTC;
            break;

            //ADC2 lábön az NTC-osztó van
        case NTC:
            //A következő láb az ADC1, amin a potenciométer van
            cbi(ADMUX, MUX1);
            sbi(ADMUX, MUX0);
            ADC_state = poti;
            break;

        default:
            break;
    }
}

```

Ilyenkor a bemeneti csatornát és az állapotflag értékét kell megváltoztatni.

Ha az ADC\_state állapot flag értéke "poti", tehát eddig a potenciométer feszültségét mértük, akkor az NTC-re kell váltani, ami azt jelenti, hogy adatlap 22-4 táblázata alapján az ADC2 (amin az NTC osztó van) csatornát kell bemenetnek választani és az állapotflag-et "NTC"-re kell állítani.

Az ADC\_State="NTC" esetében fordítva kell eljárni, bemeneti csatorna az ADC1 lesz (amin a Potenciométer van) és az állapotflag "poti" lesz.

Miután a technikai részleteken átrágtuk magunkat, élvezzük kicsit a munkánk eredményét. Tegyük breakpoint-ot a LED állító sorokhoz a main függvényben és vegyük fel a cntr számlálót, U\_poti, U\_NTC változókat a megfigyelendő változók közé (jobb klikk a változó nevéen a kódban és válasszuk az "Add Watch" menüpontot). Ekkor a watch ablakban megjelennek a változóink. Ha Debug módban futtatjuk a programunkat, akkor láthatjuk a potenciométeren és az NTC oszton mért feszültséget a watch ablakban.

Watch 1	
Name	Value
U_poti	4.316406
U_NTC	1.953125
cntr	2

6. ábra – Mikrovezérlő által mért feszültségek

Ha a számláló elérte a 10-et, akkor frissül az egyik mért feszültségérték. Ellenőrizzük az U\_poti és az U\_NTC értéke alapján az IO View ablakban a PORTA regiszter utolsó bit-jét (PA0). Láthatjuk, ha mindent jól csináltunk, hogy ha a potenciométer feszültsége (U\_poti) kisebb, mint az NTC oszton mért feszültség (U\_NTC), akkor a PA0=1, tehát a LED világít, egyéb esetben 0 és a LED nem világít.

IO View

Filter:

Name	Value
<div><div></div></div> AD_CONVERTER	
<div><div><div>ADD</div><div>OAC</div></div></div> ANALOG_COMPARATOR	
<div><div></div></div> BOOT_LOAD	
<div><div><div>MCU</div><div></div></div></div> CPU	
<div><div></div></div> EEPROM	
<div><div><div></div><div></div></div></div> EXTERNAL_INTERRUPT	
<div><div></div></div> JTAG	
<div><div><div>I/O</div><div></div></div></div> PORTA	
<div><div><div>I/O</div><div></div></div></div> PORTB	
<div><div><div>I/O</div><div></div></div></div> PORTC	
<div><div><div>I/O</div><div></div></div></div> PORTD	
<div><div><div></div><div></div></div></div> SPI	

Name	Address	Value	Bits
<div><div><div>I/O</div><div></div></div></div> PINA	0x39	0x03	<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>

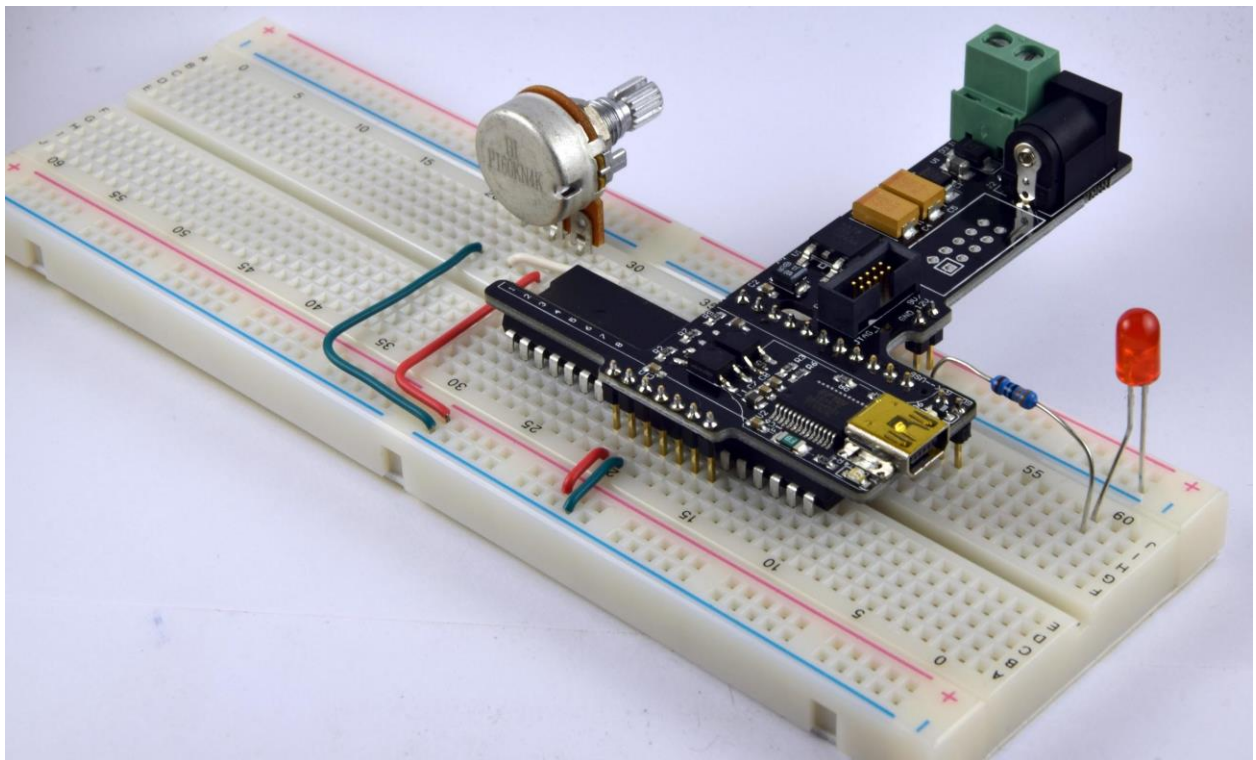
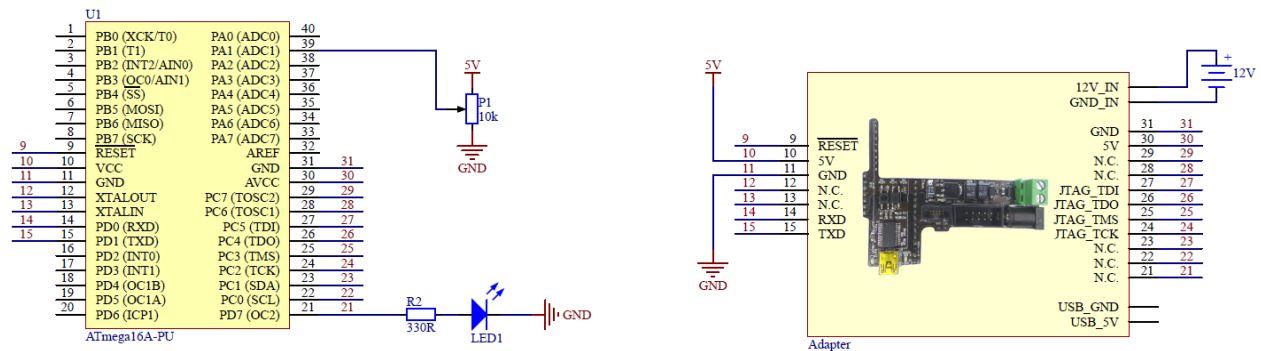
7. ábra – Mikrovezérlő regisztereinek állapota a program futása közben

Gyakorlásképpen még próbáljunk meg beállítani 2 V-ot a potenciométeren.

## MÉRÉS 2

### Pulzusszélesség-moduláció analóg-digitális átalakító segítségével (ADC\_PWM Projekt)

Állítsuk össze a következő kapcsolást.



8. ábra - Mérési elrendezés

### FELADAT:

A PA1 lábón (ADC1) lévő potenciométer AD értékének felső bájtját írjuk be a TMR2 PWM regiszterébe, vagyis a potenciométerrel legyen állítható a PD7 lábón megjelenő PWM jel kitöltési tényezője.

### Az IO Portok inicializálása

```
void IOInit()  
{  
    //PORTA felhúzóellenállás kikapcsolása a bemeneteken  
    PORTA = 0x00;  
}
```

Ez a példány Rába Ármin kizárólagos használatú példánya.

<http://kristalytisztalelektronika.hu> | Minden jog fenntartva Xtalin Mérnöki Tervező Kft.

```

DDRA = 0x00;
//A LED a PA0-n van, PORTA0 legyen kimenet (0x01)
sbi(DDRA, 0);

//PORTB felhúzóellenállás kikapcsolása és minden pin bemenet
PORTB = 0x00;
DDRB = 0x00;

//PORTC felhúzóellenállás kikapcsolása és minden pin bemenet
PORTC = 0x00;
DDRC = 0x00;

//PORTD felhúzóellenállás kikapcsolása és minden pin bemenet
PORTD = 0x00;
DDRD = 0x00;
//A PORTD PD7 pinje legyen kimenet,
//a timer2 PWM kimenete ezen a lábon van
sbi(DDRD, 7);
}

```

Először a be- és kimeneti lábakat inicializáljuk, lásd IOInit() függvény az IO.c file-ban.

A PORTA, PORTB, PORTC, PORTD-n kikapcsoljuk a felhúzóellenállásokat és bemenetnek definiáljuk a lábakat, kivéve a PORTA-n a PA0 lábat, mert ez vezérli az állapotledet, továbbá a PORTD-n a PD7-et, amelynek van egy speciális funkciója is, az adatlap 12.3.4 fejezete alapján a timer2 PWM kimenete ezen a lábon van.

#### Az órajel inicializálása

```

void TimerInit()
{
    //1x előosztás
    cbi(TCCR1B, CS12);
    cbi(TCCR1B, CS11);
    sbi(TCCR1B, CS10);

    //Timer1 interrupt be
    sbi(TIMSK, TOIE1);

    //Globális interrupt be
    sei();
}

```

Az órajel inicializálása a timer.c file-ban a timerInit() függvény segítségével történik. A TCCR1B regiszter bit-jeinek leírása az adatlap 16.11.2 fejezetében található.

A 16-6 táblázatból kiolvasható, hogy ha nincs szükségünk előosztóra, akkor a CS12, CS11, CS10 biteket 0,0,1 - be kell állítani a sorrendnek megfelelően. A maradék bit értékét nem állítjuk, nincs rájuk szükségünk, értékük marad az alapértelmezett 0.

A 16.11.7 fejezetben leírtaknak megfelelően a Timer1 interrupt-ját TOIE1 bit beállításával engedélyezzük és végül engedélyezzük a megszakítást (globális interrupt-ot).

## A pulzusszélesség-moduláció beállítása

```
void PWMInit()
{
    //TIMER2 (PWM)
    //1x előosztás (126.oldal)
    cbi(TCCR2, CS22);
    cbi(TCCR2, CS21);
    sbi(TCCR2, CS20);

    //TIMER2 (PWM)
    //1:24x előosztás (126.oldal)
    //sbi(TCCR2, CS22);
    //sbi(TCCR2, CS21);
    //sbi(TCCR2, CS20);

    //Phase correct PWM mód (125.oldal)
    cbi(TCCR2, WGM21);
    sbi(TCCR2, WGM20);

    //Ponált működés (126.oldal)
    sbi(TCCR2, COM21);
    cbi(TCCR2, COM20);
}
```

A pulzusszélesség-moduláció beállítása a pwm.c file pwmInit() függvényével, a TCCR2 regiszter beállításával történik.

A Timer2-re is a Timer1-nek megfelelően 1x-es előosztást alkalmazunk, tehát itt sincs szükség előosztóra (lásd 17.11.1 fejezetet az adatlapon). A 17.2 táblázat alapján az egyes üzemmódot választjuk, amely a hulláformát fázishelyes alakban állítja elő (WGM21 = 1, WGM20 = 0). A COM20 és COM21-es lábak működési módja függ az imént kiválasztott üzemmódtól. Egyes üzemmódban a 17.5 táblázat érvényes, melyben a ponált működést, azaz a COM21 és COM20 lábak 1 és 0-ba való állítását választjuk.

## ADC inicializálása

```
void ADCInit()
{
    //AVCC referencia
    cbi(ADMUX, REFS1);
    sbi(ADMUX, REFS0);

    //ADC1 bemenet
    sbi(ADMUX, MUX0);

    //Balra igazított eredmény -> felső 8 bit ADCRH-ban
    sbi(ADMUX, ADLAR);
}
```

Az ADC inicializálásához az ADCInit() függvényben először is be kell állítani a 22.9.1 fejezet alapján az ADMUX regiszter segítségével az ADC referencia feszültségét, mely esetünkben a külső AVCC feszültségével egyezik meg (REFS1,REFS0=0,1).



A 22.4 táblázat segítségével kiválasztjuk a beállítandó regiszterbiteket, hogy az egyes csatorna (ADC1) legyen a bemenet. Ehhez a MUX0-t kell 1-re állítani, a maradék MUX bitek maradhatnak az alapértelmezett 0 értéken, őket nem kell állítani.

Ha az ADLAR bit 1, az eredmény balra igazított, az ADCRH-ban az eredmény a felső 8 biten jelenik meg. A magyarázathoz lásd az 1. mérési példát.

#### ADC engedélyezése

```
void ADCEnable()
{
    //128-as presc
    sbi(ADCSRA, ADPS2);
    sbi(ADCSRA, ADPS1);
    sbi(ADCSRA, ADPS0);

    //Interrupt bekapcsolása
    sbi(ADCSRA, ADIE);

    //AD konverter bekapcsolása
    sbi(ADCSRA, ADEN);
}
```

Az ADC inicializálása után már engedélyezhető az ADC működése az ADCEnable() függvénnyel.

Az ADCSRA ADC vezérlő regiszterében állíthatjuk be, hogy az órajelhez képest mekkora legyen az ADC órajele. Mi ezt a prescaler-t az adatlap 22.9.2 fejezete alapján 128-ra állítjuk be. Engedélyezzük az interruptot és az ADC működését.

#### A main()-ben futó további programrészek

```
OCR2 = 0;

//Végtelen ciklus
while (1)
{
    //A PWM beállítása megszakításon keresztül történik
}
```

Kezdetben a PWM kitöltési tényezője legyen 0, azaz a PD7-re kötött LED ne világítson. Ezt úgy érjük el, hogy az Output Compare 8-bites regiszter (az OCR2 regiszter) értékét 0-ra állítjuk (lásd az adatlap 17.7.4 fejezetét).

Elindítunk egy végtelen ciklust, mivel a logikát az interrupt függvényben valósítjuk meg.

```
ISR(TIMER1_OVF_vect)
{
    //Ha még nem érte el a leosztó számláló a maximumot
    if (cntr < CNTR_MAX)
    {
        //Növelem a számlálót
        cntr++;
    }
}
```

```

    //Ha elérte
    else
    {

        cntr = 0;
        //AD konverzió indul (mintavétel)
        ADCStart();
    }
}

```

Az interrupt függvény nem csinál mást, mint elindít egy számlálót, amely 10-ig fut, ha ezt elérte, akkor töröljük (nullázzuk) a számlálót és elindítjuk az AD konverziót az ADCStart() függvénnyel.

```

void ADCStart()
{
    sbi(ADCSRA, ADSC);
}

```

Ez a függvény 1-re írja az ADCSRA vezérlőregiszter ADSC bitjét, mely elindítja a konverziót.

```

ISR(ADC_vect)
{
    //Az AD eredmény felső 8 bitjét beletölti a PWM timer regiszterébe
    OCR2 = ADCH;
}

```

Ha kész a konverzió, akkor egy másik interrupt függvény, az ISR(ADC\_vect) hívódik meg, amely a konverzió eredményének felső 8 bit-jét bemásolja a PWM Timer regiszterébe, amely már közvetlenül módosítja a kimeneti jel kitöltési tényezőjét.

Figyeljük meg az IO View-ban az AD\_Converter/ADC Prescaler Select-en belül az ADCSRA és az ADC regiszterek működését, breakpoint-ot téve az ADCEnable() és az ADC Start() függvények különböző parancsai mellé.

IO View

Filter:

Name	Value
<div><div></div><div></div></div> AD_CONVERTER	
<div><div></div><div></div></div> Reference Selection Bits...	<div>.. 0x01</div>
<div><div>ADC DAC</div></div> ADC Prescaler Select Bi...	<div>.. 0x07</div>
<div><div>+</div><div>ADC DAC</div></div> ANALOG_COMPARATOR	
<div><div></div><div></div></div> BOOT_LOAD	

Name	Address	Value	Bits
<div><div>ADC DAC</div></div> ADC	0x24	0xBD	<div><div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div></div>

9. ábra – Mikrovezérlő regisztereinek állapota a program futása közben

Tegyük Brakpoint-ot a main függvény és az interrupt függvény OCR2-t állító sorai elé. Figyeljük meg az IO View-ban a TIMER\_COUNTER\_2/Clock select bits-en belül az OCR2 regiszter működését és vessünk közben pillantást a kimeneti LED-ünkre is. Tekerjük el a potmétert és figyeljük meg a változást ismételten.

Futtatva a programot szabad szemmel ebből annyit figyelhetünk meg, hogy nagyobb kitöltési tényező mellett, tehát feltekerve a potmétert, a LED erősebben világít.

## GYAKORLÁS

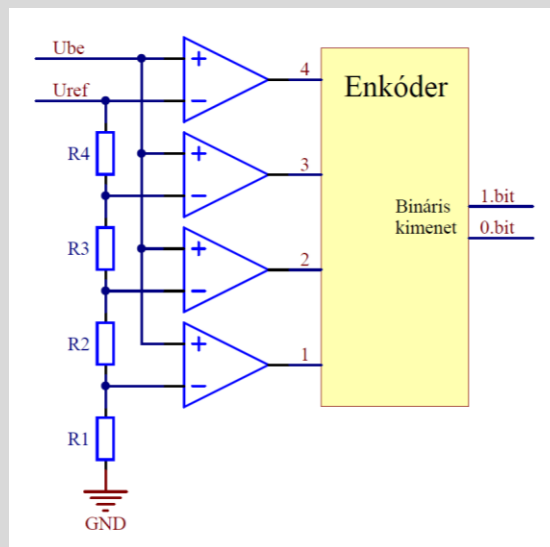
Módosítsuk a feladatot annyiban, hogy a Timer2 előosztását változtassuk 1024-re. Figyeljük meg, hogy mennyiben változik meg a kimenet. Ekkor kisebb kitöltési tényező esetén a LED már szaggatottan világít.

## Kitekintés

A témakör mélyebb megértése érdekében röviden áttekintjük az analóg-digitális átalakítók főbb fajtáit:

### Flash ADC

A flash ADC esetében a bemeneti feszültség közvetlenül, egy lépésben, a feszültséggel arányos számmá alakul. Egy sor párhuzamos komparátorból áll, mindegyik komparátor a bemeneti feszültséget hasonlítja össze a különböző, sorosan kapcsolt ellenállásokból felépített feszültségosztóból vett referencia feszültségekkel. Majd egy enkóder binárisan megadja a legnagyobb sorszámú magas kimenetű komparátor sorszámát. Bár a flash ADC előnye a gyorsasága, mégis a sok komparátorigénye az áramkört bonyolulttá teszi, nagy a teljesítményigénye, a bonyolultságához képest kicsi a felbontása és nagyon drága. Kis felbontásigényű, nagyfrekvenciás áramkörökben használják.



10. ábra - Flash ADC belső felépítése

### Sigma-delta átalakító

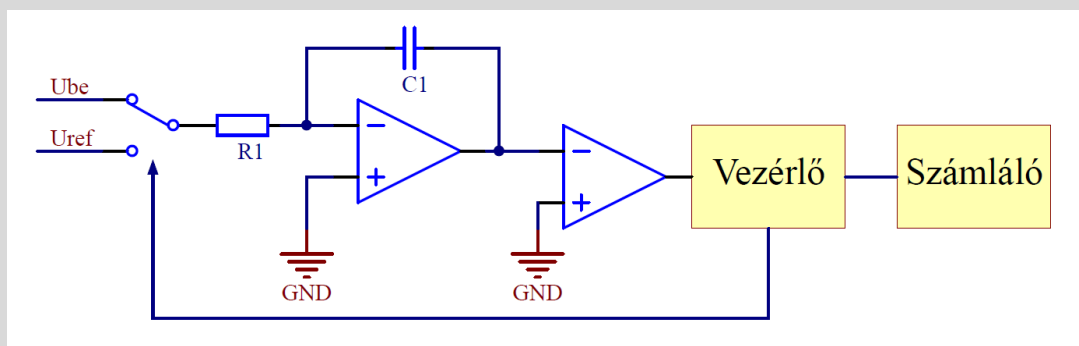
Működésének megértése túlnyúlik a tananyag keretein, ezért itt csak a főbb jellemzőit foglalkoztatjuk össze. Magas bitszámot az órajelnél jóval magasabb mintavételezéssel éri el. A túlmintavételezésnek köszönhetően lassú, cserébe viszont nagy felbontású és nem igényel külső precíziós alkatrészeket. Általában hangfrekvenciás alkalmazásokban használják.

### Dual slope/kétszeresen integráló/kettős meredekségű átalakító

A bemenő jel tölti a kapacitást egy határozott ideig. Az időbeli integrálással a zaj kiátlagolódik. Amíg a kapacitás kisül egy bizonyos fokig, egy számláló számlálja a kimeneti biteket. Minél nagyobb a kisülési idő, annál nagyobb a számláló állása.

A korábbi típusokhoz képest nagyobb a zaj immunitása, pontos, viszont lassú, és precíziós külső alkatrészekre (ellenállás, kondenzátor) van szükség a pontosság érdekében.

Manapság, amikor a mikrokontrollerekben már integrált ADC-k is elérhetőek, csökkent a szerepe, mégis zajos jelek méréséhez még mindig érdemes figyelembe venni.

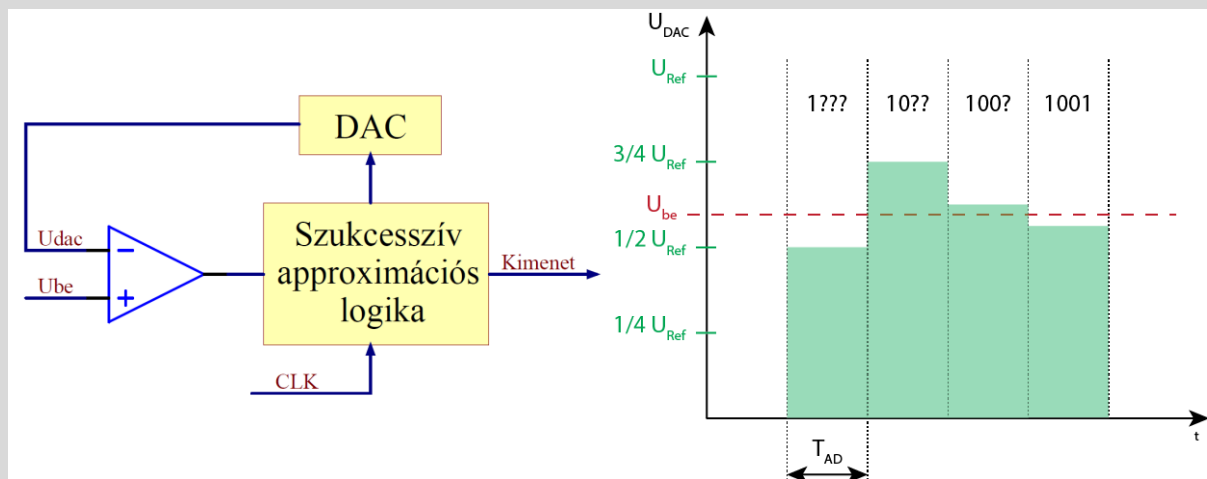


11. ábra -Sigma-delta átalakító belső felépítése

## Szukcesszív approximációs (fokozatos megközelítéses) ADC

Számunkra ez a legérdekesebb típus, mivel a választott mikrokontrollerben éppen egy ilyen szukcesszív approximációs típusú ADC van beépítve. Nem véletlen, ugyanis egyszerű a felépítése, pontossága és sebessége miatt pedig ez az egyik legelterjedtebb átalakító.

Itt az eredményt annyi lépésben kapjuk meg, amennyi a bitszám (ebből következik, hogy a mérés alatt nem szabad változtatni a bemenetet, különben hibás eredményt kapunk. Ezt elkerülendő a konverter elé mintavéveő-tartót kell rakni). Egyszerre mindig csak egy helyiértéket határozunk meg, a legnagyobb helyiértéktől kezdve. A komparátor azt vizsgálja meg, hogy a bemeneti feszültség nagyobb-e, mint a legnagyobb helyiértékhez tartozó referenciafeszültség. Ezt a referenciafeszültséget egy erre a célra beépített DAC generálja (A DAC működése fordított az ADC-hez képest, tehát egy digitális értéhez, számhoz rendel feszültséget). Ha a bemeneti feszültség kisebb, mint a DAC kimenetén lévő feszültség, akkor a SAR (Successive Approximation Register) soron következő bitje 0, egyébként 1. A maradékot összehasonlítjuk a következő helyiértékkel, és így tovább. Annyi referenciafeszültségre van szükség, mint amennyi a bitszám.



12. ábra - Szukcesszív approximációs ADC belső felépítése

## Önálló mérési feladat

### Hiszterézises komparátor szoftveresen

A hiszterézises komparátort meg lehet hardveresen, fizikai komponensek összeépítésével (lásd a komparátoros tananyag részben) és ADC-vel szoftveresen, programkódból is valósítani. Mi az utóbbi megoldást fogjuk kivitelezni.

A mikrokontrolleres alapkapcsolást egészítsük ki egy potméterrel, amely a tekeréssel arányosan változtatja az ellenállását.

Írjuk meg a kódot úgy, hogy az adc 600-as értéke felett egy kimeneti led világítson, az alatt pedig ne világítson, a potméteren pedig állítsunk be 3 V-ot.

Figyeljük meg, ha nagyon pontosan beállítjuk a 600-as ADC érték körülire a potmétert, akkor ugyanarra a bemenetre a LED néha világít, néha pedig nem. Ez a jelenség amiatt lehetséges, mert a jel és a táp is lehet zajos, a komparálás (a bemeneti két érték összehasonlításának a kiértékelése) pedig már nagyon kis különbség esetén is más kimenetet ad.

Ezt a jelváltási tartományt bővíthetjük ki a hiszterézises komparátorral, ha nem akarjuk, hogy a bemeneti kis változás azonnal megjelenjen a kimeneten, és ez a zaj-okozta változás "röcögést", azaz jelszint ide-oda ugrálását okozza. Próbáljuk ki, hogy ha a komparálási feltételt megváltoztatjuk úgy, hogy

```
if(ADC>700) output=1;
```

```
if(ADC<500) output=0;
```

akkor körülbelül 3,5V-nál és 2,5V-nál fog kapcsolni, ami pont egy 1V-os hiszterézis lesz.

## ÖSSZEFOGLALÓ

Ebben a tananyagban megismerkedhettük, hogy az analóg jeleket miként lehet digitális számmá alakítani, és azokat felhasználni a programunkban. Mivel a körülöttünk lévő világ analóg, azaz folytonos jelekből épül fel, így ennek nagy szerepe van a számítástechnikában, ahol csak digitális jeleken tudunk műveleteket végezni. A későbbi részekben építkezni fogunk az itt megszerzett tudásra.