

# 9. Mikrovezérlők I.

Írta: Lágler Gergely, Vincze Viktor

Lektorálta: Veréb Szabolcs

## BEVEZETÉS

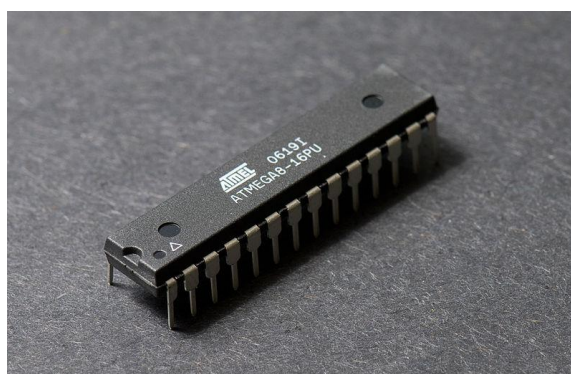
---

A tananyagrészt végére a mikrovezérlőnk képes lesz egy LED-et villogtatni egy előre megírt szoftver alapján.

Nagyon célratorően vágjunk is egyből a közepébe! A feje tetejére állítjuk itt is a szokványos tanulási rendet, és először próbáljuk ki az eszközt, hogy később, az alapok és a legbelső működés megértésékor már kézzelfogható tapasztalatokkal rendelkezünk. Az ATmega16A nevű mikrovezérlővel fogunk megismerkedni, és a tananyagban szereplő példákat is ennek a segítségével valósítjuk majd meg.

Azt tehát, hogy hogyan működik a mikrovezérlő, lépésről lépésre, gyakorlat közben fogjuk elsajátítani. Ez elegendő lesz arra, hogy valaki akár önállóan képes legyen egyszerűbb eszközöket tervezni és megvalósítani, azonban meg kell jegyeznünk, hogy a tananyag nem helyettesít semmilyen szakirányú képzést sem.

Mi is egy mikrovezérlő? Mire lehet használni? Hol használnak ilyet, vagy ehhez hasonló alkatrészeket? Mennyi féle létezik belőlük? Mi a különbség köztük? Miért éppen ezt használjuk?



Szerző: Pengo [CC BY-SA 3.0 (<https://creativecommons.org/licenses/by-sa/3.0/>)]  
[https://commons.wikimedia.org/wiki/File:ATmega8\\_01\\_Pengo.jpg](https://commons.wikimedia.org/wiki/File:ATmega8_01_Pengo.jpg)

**1. ábra** - ATmega16A mikrovezérlő

A mikrovezérlő is egy IC, azaz integrált áramkör. Ez tehát azt jelenti, hogy a belsejében lévő apró szilícium lapkán (chip) rengeteg parányi alkatrészt hoztak létre, melyek összessége bonyolult, összetett működésre képes.

## PC-S ANALÓGIA

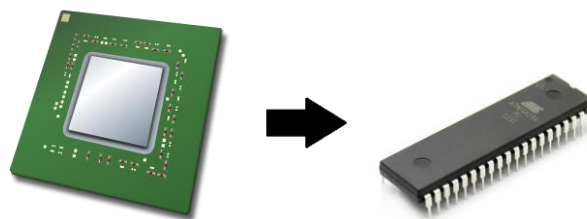
---

A mikrovezérlőt úgy ismerhetjük meg kicsit pongyola módon ugyan, de a leghatékonyabban, ha a személyi számítógépekhez (PC), vagy laptopokhoz hasonlítjuk, mert ezek ugyan nem mikroprogramozott eszközök, de sok mindenben hasonlítanak, illetve általánosan ismertek. A mikrovezérlők apró részei ugyanazokat a

feladatokat látják el, mint a PC belsejében az egyes nagyobb hardver alkatrészek, mindezt azonban egy apró tokba integrált módon (innen ered az integrált áramkör elnevezés). A PC-n is, valamint a mikrovezérlőn szintén (általunk is készíthető) program fut, és ez a szoftver határozza meg az eszköz működését.

## MŰVELETEKET VÉGZŐ EGYSÉG

A PC-ben a processzor végzi az alapvető műveleteket, mint például az összeadás, kivonás, szorzás, osztás. A mikrovezérlőben ugyanezen feladatokat a mikrovezérlő egy része; a központi műveletvégző egység (Central Processing Unit, vagyis CPU) látja el.

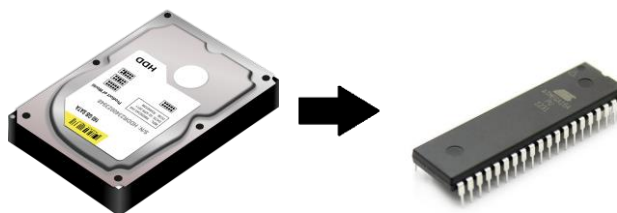


*2. ábra - egy processzor a PC-n belül, illetve egy mikrovezérlő*

## NEM FELEJTŐ MEMÓRIA

A PC-ben használatos nem felejtő memóriák például a winchesterek (HDD) és az SSD-k. Rengeteg adat tárolható rajtuk, mint például a programjaink, egyéb fájljaink. Ezek az adatok a gép kikapcsolását követően is megmaradnak, ezért hívjuk ezt nem felejtő memóriának.

A mikrovezérlőben is találhatók ilyen egységek. FLASH memóriának, vagy programmemóriának hívjuk azt a részt, ami a programot tartalmazza. EEPROM memóriának hívjuk azt a részt, amiben egyéb adatokat tárolhatunk.

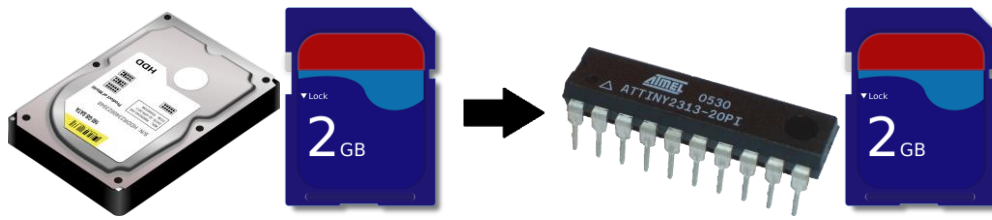


*3. ábra - egy winchester a PC-n belül, illetve egy mikrovezérlő*

## TOVÁBBI NEM FELEJTŐ MEMÓRIÁK

A PC-hez - ha további memóriára van szükségünk - lehet csatlakoztatni külső memóriákat, mint például külső winchester, pen-drive, SD-kártya.

Ezekhez hasonló lehetőségek a mikrovezérlőknél is léteznek, például külső "EEPROM" memória, vagy például szintén az SD-kártya.

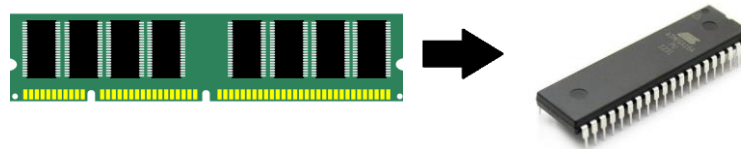


4. ábra - tipikus külső memóriák PC illetve mikrovezérlő esetében

## FELEJTŐ MEMÓRIA

A PC felejtő memóriájában tárolt adatok kikapcsoláskor ugyan elvesznek, de ez a memória sokkal gyorsabb működésre képes, mint a nem felejtők, így a processzor számára gyorsabb működést tesznek lehetővé. Ez az úgynevezett RAM memória.

Ugyanez megtalálható a mikrovezérlőben is.



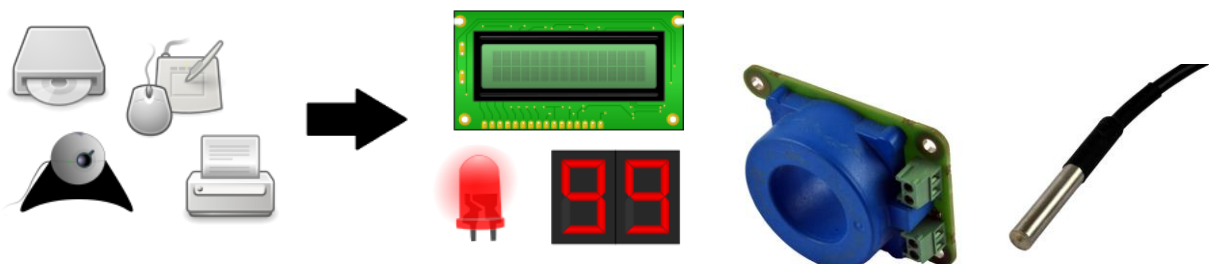
5. ábra - PC memória illetve mikrovezérlő

## PERIFÉRIÁK

A PC-re különböző adat beviteli, és kiviteli egységeket csatlakoztathatunk, úgynevezett perifériákat, például: monitor, billentyűzet, egér, nyomtató, kamera, szkennер, stb., ráadásul ezeket egyszerűen, egy-egy csatlakozó egyértelmű csatlakoztatásával, esetleg egy illesztőszoftver (driver) telepítésével működésre is tudjuk bírni.

A mikrovezérlőre is illeszthetők különböző perifériák, ám ezek legtöbbször jellemzően primitívebbek. A perifériák irányuk szerint csoportosítva lehetnek kimenetek (output), vagy bemenetek (input).

Például egyszerű kimenetek: kis kijelző, hangjelző, LED-sor. Jellemző bemenetek: nyomógombok, különböző szenzorok (hőmérsékletszenzor, nyomásszenzor, áramszensor, stb.).



6. ábra - perifériák összehasonlítása

Ahhoz, hogy ezek a perifériák az általunk kívánt feladatot végrehajtsák, tehát az adott fizikai jellemzőt kifejezni, vagy mérni, esetleg megváltoztatni tudják, úgymond illeszteniük kell őket a mikrovezérlőhöz, és olyan szoftverrészletet kell írniuk a mikrovezérlőbe, amely kezelni tudja az adott perifériát.

## ÖSSZEFOGLALÁS

Számos ok ad létjogosultságot a mikrovezérlőknek. Ezek az apró alkatrészek ugyan sokkal kisebb számítási kapacitással rendelkeznek, de nagyjából ugyanilyen mértékben kevesebb az energia fogyasztásuk is, nem tartalmaznak mozgó, kopó alkatrészeket, így élettartamuk rendkívül nagy, valamint működésük is sokkal megbízhatóbb. Természetesen a megbízhatósághoz többek között megbízható kiegészítő alkatrészek, és ugyanilyen szoftver is feltétel.

Ahhoz, hogy komoly ipari rendszerekbe tervezzen valaki mikrovezérlős áramkört, több év tanulásra és tapasztalatra van szükség, ebben a tananyagban csak az alap építőelemek kis szegmensével ismerkedünk meg épp annyira felszínesen, hogy rendelkezésre álljon belőlük néhány, és ezzel gyakorlati háttérrel biztosítsunk az egészen bonyolult elméleti alapok még hatékonyabb elsajátításához.

A PC és a mikrovezérlő sok hasonlósága mellett különbségek tehát többek közt a méret, az ár, a megbízhatóság, a kezelhetőség, és a megvalósítható feladat bonyolultsága.

## FELHASZNÁLÁSI TERÜLETEK

Mikrovezérlők, vagy nagyon hasonló eszközök valósítják meg a működést vezérlő, illetve szabályzó feladatokat a legtöbb körülöttünk található, elektronikával ellátott automatikus eszközben. Mikrovezérlő vezérelhet egy digitális mikrohullámú sütőt, egy mosógépet, mosogatógépet, intelligens otthon egy-egy szenzoregységét, vezérlőegységet, liftek kezelőfelületét, kártyás beléptetőrendszert, vonalkódolvasót, bankkártya leolvasót, a kórházakban és járművekben található legtöbb eszközt, mint például betegőrző készülékek, vérnyomásmérők, defibrillátorok, járművezérlők, ablaktörlő-, légszak-, ablakemelő-, tükör-, ABS-, klíma-elektronikát, stb. Hasonló eszköz található a mobiltelefonokban, modern rádiókban, tévékben is.

Látható, hogy a legtöbb körülöttünk lévő elektronikus eszköz, illetve biztonságkritikus eszköz tartalmazhat mikrovezérlőt, ami arra is utal egyben, hogy az ebben a tananyagban rendhagyó módon megismerhető ATmega16A mikrovezérlőt is a legkülönbözőbb célokra leszünk képesek használni, illetve ezen ismeretek birtokában önállóan elsajátítható lesz egy-egy bonyolultabb mikroprogramozott eszköz használata is.

### Kitekintés

#### A mikroprogramozott eszközök fajtái

A mikroprogramozott eszközök azon mikroelektronikai alkatrészek összefogó neve, melyekre a felhasználó, illetve a tervező programot kell írjon a működéshez. Fontos tisztában lenni vele, hogy ugyan a paletta egy komoly részét a mikrovezérlők teszik ki, léteznek szép számmal olyan mikroprogramozott eszközök is, melyek felépítésükben, vagy éppen működésükben kisebb vagy nagyobb mértékben eltérnek, és egy-egy speciális területen hatékonyabbak lehetnek. Például a digitális jelfeldolgozó processzorok (DSP, digital signal processor) a mérés-technikai alkalmazásokban képesek kiemelkedő teljesítményre. Hasonlók

még - sajnos egyenlőre csak említés szintjén - a CPLD-k, FPGA-k, Application processor-ok, stb. Ezek belső felépítésével, működésével, programozásával legkönnyebben egy villamosmérnöki képzés során találkozhat az ember.

### Különböző architektúrák

Az egyes mikroprogramozott eszközök belső felépítése, azaz rendszerarchitektúrája nagyon különböző lehet attól függően, hogy a főbb építőelemekből milyeneket tartalmaznak, vagy sem. Szintén maradv a PC-s analógiánál, az egyik ilyen legjellemzőbb eltérés lehet a műveletvégző egység szóhossza. A PC-ben a processzorokból léteznek olyanok, melyek 32 vagy olyanok, melyek például 64 bites adatokkal képesek egy lépésben számolni. Ez a tulajdonsága meghatározza a PC-re telepíthető operációs rendszert, a feltelepíthető szoftvereket, de még a PC hatékonyságát is. Ugyanígy a mikrovezérlők között is megtalálhatunk különböző szóhosszúságú CPU-kat. Nagyon elterjedtek még, és minden bizonnyal sokáig azok is maradnak a 8 bites mikrovezérlők, de találkozhatunk 16 bites, és egyre több eszközben nagyobb szóhosszúságú verziókkal is. Az ATmega16A egy 8 bites mikrovezérlő.

### KÜLÖNBÖZŐ MIKROVEZÉRLŐ GYÁRTÓK

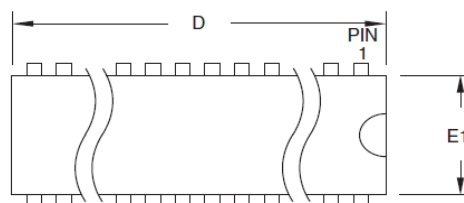
Mikrovezérlőket az 1970-es évektől kezdve, mára már évente több milliárd-számra gyártanak. A legismertebb gyártók a teljesség igénye nélkül: Microchip, ST Microelectronics, NXP Semiconductors, Texas Instruments, Analog Devices. Korábban nagy nevű gyártó volt az Atmel és a Freescale, de az utóbbi években felvásárolta őket a Microchip és az NXP.

## Az ATMEGA16A MIKROVEZÉRLŐ

### ATMEGA16A TOKOZÁSA

A tananyag internetes oldalán, ennél a résznél megtalálható az általunk most használt ATmega16A mikrovezérlő adatlapja, illetve egy hivatkozás a gyártó honlapjára (természetesen angolul.)

A tokozásról információt (Packaging information) a 325-327. oldalakon találhatunk. Ezek közül mi a furatszereltet választottuk, hogy könnyű legyen a breadboardon használni. Ezt a tokozást itt az Atmel a 40P6 kóddal jelöli, ami azt jelenti, hogy 40 lábú alkatrésze van szó, és a két lábsor között 0,6 hüvelyk (vagy inch, illetve col), vagyis 15,24 mm a távolság, tehát tökéletesen megegyezik a breadboard furattávolságaival, melyek 0,1 hüvelykre, azaz 2,54mm-re vannak egymástól.



7. ábra - a DIP tokozás

Ugyanezt a tokozást a legtöbb helyen DIP-40-nek nevezik, ami a 40 lábú, kétsoros tokozásra (Dual In-line Package) utal.

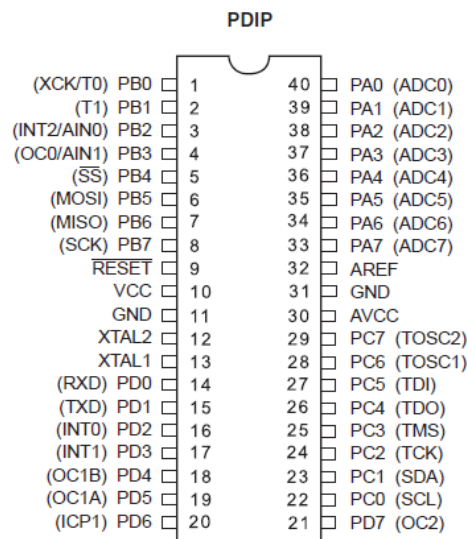
## Kitekintés

### A teljes gyártói jelölés

A pontos cikkszámot is kikereshetjük az adatlap 324. oldalán a rendelési információknál (Ordering information). Látható, hogy 40P6-os tokozással az ATmega16A-PU cikkszámú verzió rendelkezik.

## ATMEGA16A-PU LÁBKIOSZTÁS, ÉS ALAP BEKÖTÉS

Az adatlap 3. oldalán található a lábkiosztás (Pin configuration, vagy Pinout):



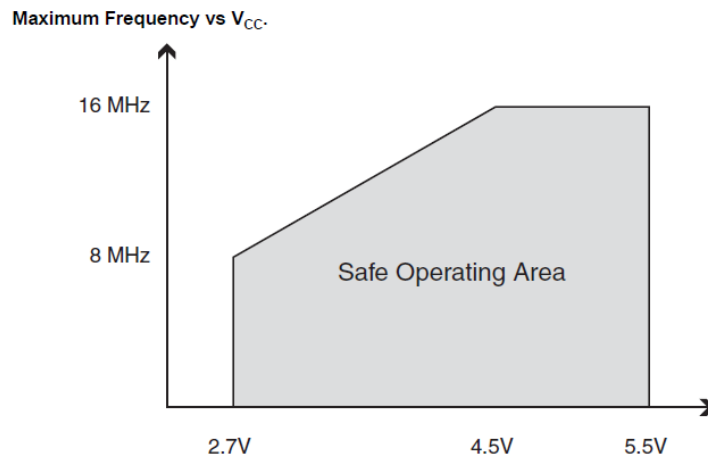
**8. ábra** - ATmega16A-PU lábkiosztása

Az adatlap szintén segít abban, hogy értelmezzük a fenti ábrát, az összes láb pontos leírása megtalálható az áttekintés (Overview) részen belül a lábak leírásánál (Pin description). Mivel lépésről lépésre fogunk megismerkedni az egyes funkciókkal, nem szükséges most megértenünk minden egyes láb, és a mögötte található összes áramkör (ezekre utalnak a lábak elnevezéseinél a zárójeles rövidítések) működését.

### Táplálbak

Mivel a mikrovezérlő is aktív alkatrész, így ahhoz, hogy működni tudjon, megfelelő tápfeszültségre van szüksége, melyet az erre a célra dedikált lábaira kell kötnünk. A legtöbb mikrovezérlőnek, így az ATmega16A-nak is több tápláló van, a hibátlan működés érdekében minden táplálóra megfelelő feszültséget fogunk kötni.

A GND lábai tehát a "föld" lábak, a VCC és az AVCC pedig a táplálbak. Minden mikrovezérlő adatlapjában megadják, hogy mekkora feszültségtartományban üzemképesek, itt ez az elektromos karakterisztikák (Electrical Characteristics) részben, a 281. oldalon található.



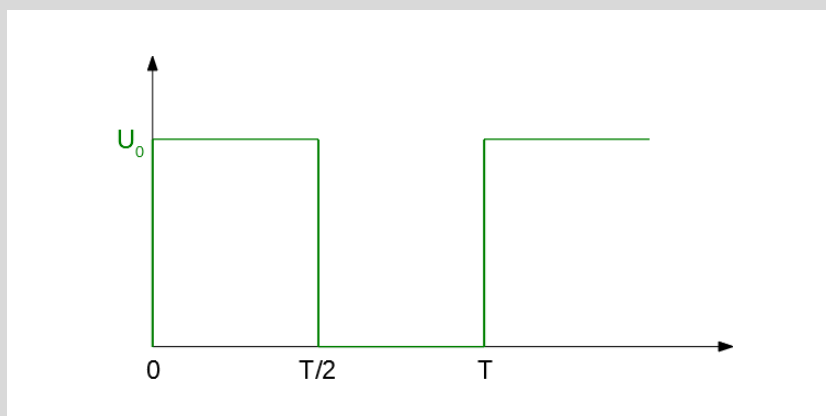
9. ábra - adatlap részlet - megbízható működési tartomány

Az ábra a biztonságos működési tartományt (safe operating area) adja meg, tehát az eszközünk 2,7 és 5,5V között fog helyesen működni. Az ábrán az is látszik, hogy az egyes tápfeszültségek mellett biztonságosan mekkora órajel frekvenciát állíthatunk be.

## Kitekintés

### Az órajel

Az órajel 0, 1, 0, 1 jelek egymásutánja, úgynevezett négyzögjel.



10. ábra - az órajel

A jelszintek közötti váltást a négyzögjel felfutó, illetve lefutó élének nevezzük. A legtöbb eszközünk a felfutó élre érzékeny, ilyenkor történik meg bennük a soron következő művelet. A mikrovezérlőnkben is pontosan így történik. Az órajel előállítását rendszerint egy külön erre a célra használt elektronika állítja elő, a mi esetünkben több választási lehetőségünk is van, ezek közül kettőt említünk meg.

A tananyag során végig a mikrovezérlő belsejébe épített úgynevezett belső RC oszcillátort (Calibrated Internal RC oscillator) fogjuk használni. Erről bővebben az adatlap 28. oldalán olvashatunk. Itt látható többek közt az is, hogy belső oszcillátor használata esetén maximum 8 MHz-es órajelet állíthatunk be, ez a maximum, amire a belső áramkör képes.

Lehetőségünk lenne egy megfelelő kvarckristályt kapcsolni a mikrovezérlő Xtalín és Xtalout (az xtal kifejezés a kristályra utal) lábaira, ekkor a mikrovezérlő a kristályt felhasználva állítaná elő az órajelét. Erre a célra különféle kristályokat gyártanak, egy-egy adott kristály adott frekvencia előállítására képes, többek között kapható 16 MHz-es is.

### A mikrovezérlő működési sebessége

Az órajel frekvenciát programozáskor fogjuk beállítani, és ez fogja megadni a mikrovezérlő működési sebességét. Nagyobb frekvencia nagyobb működési sebességet, gyorsabb műveletvégzést tesz lehetővé, minimális fogyasztásnövekedésért cserébe. 8MHz-es frekvencia tehát azt jelenti, hogy 1 másodperc alatt ilyen beállítással ez az eszköz például 8 millió összeadást tud elvégezni.

Láthatjuk azt is, hogy 16MHz-re is állítható, azonban ahhoz legalább 4,5V-os tápfeszültség előírt. Mivel egyelőre nem célunk minél kisebb fogyasztásra törekedni, ezért a tananyagban végig 8MHz-es frekvencián fogjuk használni a belső RC oszcillátort, és az egyszerűség kedvéért 5V-os tápfeszültséget adunk majd a mikrovezérlőnek.

### Abszolút maximum határok (Absolute maximum ratings)

Érdemes minden esetben megvizsgálni azt, hogy mik azok a maximum fizikai határok (feszültség, áram, stb), amit az éppen használni kívánt eszközeink táplába, egyéb lábai, vagy tokja elviselnek. Ezek az adatok a mikrovezérlő adatlapjának 279. oldalán találhatóak:

#### 27.1 Absolute Maximum Ratings\*

Operating Temperature .....	-55°C to +125°C
Storage Temperature .....	-65°C to +150°C
Voltage on any Pin except $\overline{\text{RESET}}$ with respect to Ground .....	-0.5V to $V_{CC}+0.5V$
Voltage on $\overline{\text{RESET}}$ with respect to Ground .....	-0.5V to +13.0V
Maximum Operating Voltage .....	6.0V
DC Current per I/O Pin .....	40.0mA
DC Current $V_{CC}$ and GND Pins .....	200.0mA PDIP and 400.0mA TQFP/MLF

11. ábra - adatlap részlet - abszolút maximum értékek

Számunkra mindegyik adat fontos lehet a későbbiekben, egy-egy periféria illesztésekor vissza kell majd tekintenünk ide, és kalkulálni velük:

- Maximum működési feszültség (maximum operating voltage): Látható, hogy ha a táplábra akár csak rövid ideig is 6V-nál magasabb feszültséget adunk, a mikrovezérlő maradandóan károsodhat, akár teljesen tönkre is mehet.
- Maximum feszültség az egyes lábakon (Voltage on any pin): egy-egy lábra a GND-hez viszonyítva nem köthetünk -0,5V-nál kisebb, és 5V-os tápfeszültség esetén 5,5V-nál többet.
- Maximum áram az egyes lábakon (DC current per I/O pin): egyik irányban sem folyhat 40mA-nél nagyobb áram.



- Maximum áram a táplálbakon (DC current VCC and GND pins): nem folyhat 200mA-nél több egyik táplálbon sem.

### Digitális bemenetek és kimenetek

A PA0...7, PB0...7, PC0...7, PD0...7 lábak mindegyike használható digitális bemenetként vagy digitális kimenetként is.

A digitális bemenet meg tudja állapítani, hogy a rá kapcsolt feszültség logikai alacsony, vagy logikai magas szintű. Logikai alacsony "0" értéket 0V, vagy közel 0V bemenet esetén kapunk, logikai magas "1" értéket pedig a tápfeszültség (VCC, például 5V) közelében kapunk.

A digitális kimenet logikai alacsony, vagy magas szintet kapcsol a lábra attól függően, hogy a szoftverben mit állítottunk be, illetve hogyan módosítjuk a kimeneti értéket működés közben.

## Kitekintés

### Logikai alacsony (logic low) és logikai magas (logic high) pontos definíciói

Joggal merülhet fel bárkiben a kérdés, hogy például 5V-os tápfeszültség esetén egy mikrovezérlő bemenet milyen feszültség szinteket fog biztosan 0-nak, és milyeneket biztosan 1-nek érzékelni, illetve egy kimenet esetén milyen feszültség szintek garantáltak. Erre választ szintén az "Electrical characteristics"-en belül, az adatlap 279. oldalán találhatunk választ:

27.2 DC Characteristics						
$T_A = -40^{\circ}\text{C}$ to $85^{\circ}\text{C}$ , $V_{CC} = 2.7\text{V}$ to $5.5\text{V}$ (Unless Otherwise Noted)						
Symbol	Parameter	Condition	Min	Typ	Max	Units
$V_{IL}$	Input Low Voltage except XTAL1 and RESET pins	$V_{CC}=2.7 - 5.5$	-0.5		$0.2 V_{CC}^{(1)}$	V
$V_{IH}$	Input High Voltage except XTAL1 and RESET pins	$V_{CC}=2.7 - 5.5$	$0.6 V_{CC}^{(2)}$		$V_{CC} + 0.5$	V
$V_{OL}$	Output Low Voltage <sup>(3)</sup> (Ports A,B,C,D)	$I_{OL} = 20\text{mA}$ , $V_{CC} = 5\text{V}$ $I_{OL} = 10\text{mA}$ , $V_{CC} = 3\text{V}$			0.7 0.5	V V
$V_{OH}$	Output High Voltage <sup>(4)</sup> (Ports A,B,C,D)	$I_{OH} = -20\text{mA}$ , $V_{CC} = 5\text{V}$ $I_{OH} = -10\text{mA}$ , $V_{CC} = 3\text{V}$	4.2 2.2			V V

12. ábra - adatlap részlet - bemenetek és kimenetek jellemző feszültség szintjei

A bemenet logikai 0-t fog jelezni, ha a rákapcsolt feszültség nagyobb, mint -0,5V, és kisebb, mint 0,2 VCC, esetünkben  $0,2 \times 5\text{V} = 1\text{V}$ .

A bemenet logikai 1-et fog jelezni, ha a rákapcsolt feszültség nagyobb, mint esetünkben 3V, és kisebb, mint 5,5V.

A kimeneten logikai 0 esetén 0,7V-nál garantáltan kevesebb fog megjelenni, ha 20mA-nél kevesebb áram folyik rajta.

A kimeneten logikai 1 esetén 4,2V-nál garantáltan nagyobb fog megjelenni, ha 20mA-nél kevesebb áram folyik rajta.

## A PROGRAMOZÓ ESZKÖZ

Ahhoz, hogy a mikrovezérlőnk utasítások sorozatát, azaz egy programot hajtson végre, előbb ezt a programot meg kell valósítani, majd úgymond "rátölteni" a mikrovezérlőre. Ezekhez a műveletekhez szükségünk lesz egy PC-re, egy megfelelő programozó eszközre és a megfelelő tápfeszültséggel ellátott mikrovezérlőnkre.

Többféle programozó eszköz kompatibilis ezzel a mikrovezérlővel, sőt némelyik a programozáson kívül is képes további hasznos feladatokat ellátni. Ismerjünk meg most röviden 3 ilyen programozó, és hibakereső (programmer and debugger) eszközt, továbbiakban: programozó, vagy debugger. Nekünk csak az egyikre lesz szükségünk a három közül.

Mindhárom eszköz többféle szabvány szerint is csatlakoztatható a mikrovezérlőhöz, mi ezek közül a JTAG-nek nevezettet fogjuk használni, mivel ezzel megvalósítható minden feladat, amit ebben a tananyagban tárgyalunk. A programozó eszköz JTAG csatlakozójának lábait fogjuk összekötni a mikrovezérlő megfelelő lábaival.

### AVR Dragon

A programozót USB kábelrel csatlakoztatni kell a PC-hez, vagy laptophoz. A következő képrészlet szerint a 10 pólusú "JTAG" felíratú csatlakozóhoz kell csatlakoztatni egy 10 pólusú szalagkábel, majd azt a mikrovezérlőhöz a később leírt módon. (Az AVR Dragon-ról részletes angol nyelvű anyag található a honlapon.)



13. ábra - AVR Dragon

A szalagkábel csatlakoztatásakor figyelni kell arra, hogy a programozón található tűskesor 1-essel jelölt lábához kerüljön a szalagkábel csatlakozó 1-es lába (ezt általában egy háromszöggel jelölik).

### ATATMEL-ICE-Basic

A programozót micro-USB kábelrel csatlakoztatni kell a PC-hez, vagy laptophoz. Az "AVR" felíratú 10 pólusú, dupla sűrűségű csatlakozóhoz kell csatlakoztatni egy megfelelő 10 pólusú szalagkábel, majd azt a mikrovezérlőhöz a később leírt módon.



**14. ábra** - ATATMEL-ICE-Basic

#### **ATATMEL-ICE-Basic-PCBA**

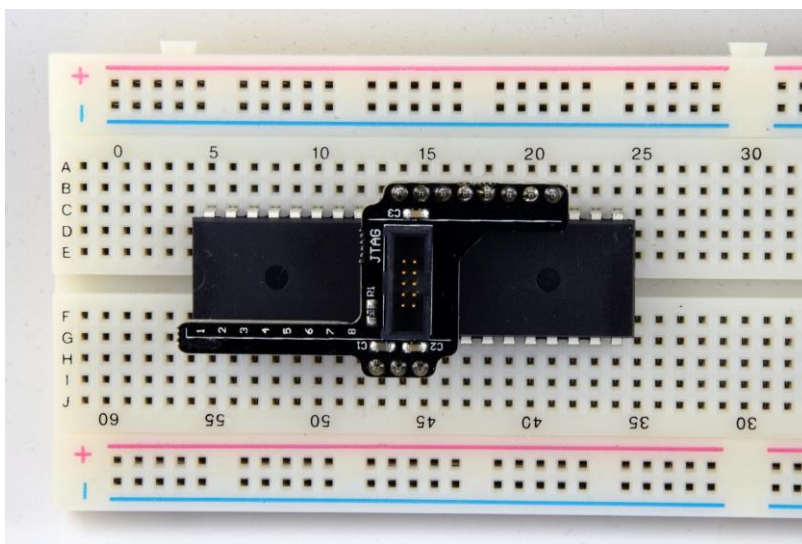
Az ATATMEL-ICE-Basic-PCBA az előző eszköz dobozolatlan verziója, így a csatlakozók kezelése megegyezik. A dobozolatlan elektronika sokkal kényesebb a különböző fizikai behatásokra, úgy mint mechanikai hatások, nedvesség, elektromos kisülések (ESD, azaz electrostatic discharge).

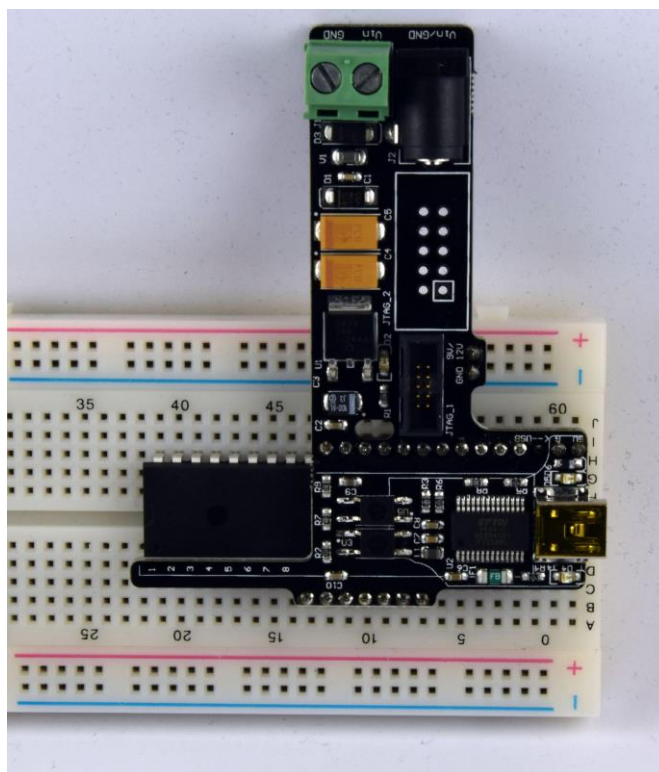


**15. ábra** - ATATMEL-ICE-Basic-PCBA

#### **A JTAG csatlakozó és a mikrovezérlő összekötése**

A JTAG programozó, és a mikrokontroller csatlakoztatását megkönnyíti a képen látható adapter:

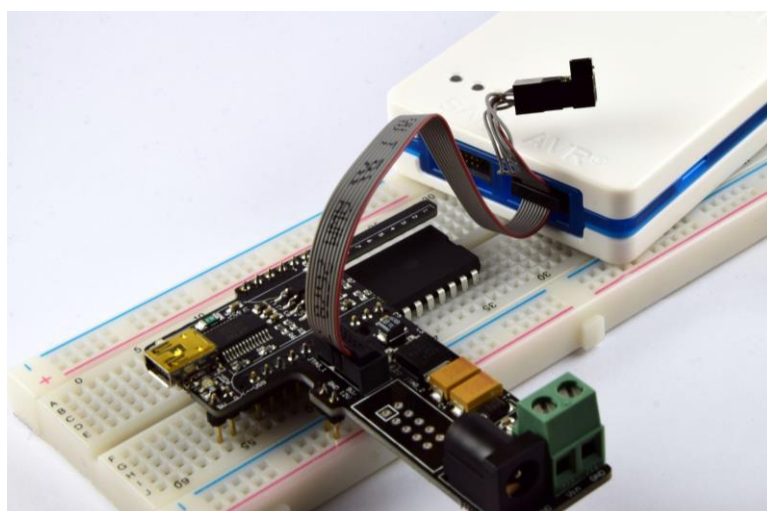




**16. ábra** - a különböző programozó adapterek használata

A jelöléseknek megfelelően nyomjuk a mikrovezérlő fölé a breadboardba az adaptert. Ügyeljünk arra, hogy az adapter 1-es felirata a mikrovezérlő 1-es lába fölé kerüljön! Az adapterből kiálló tűksorok pontosan a mikrovezérlő 5V, GND és JTAG lábaihoz csatlakoznak, és a nyomtatott áramkör vezetékei összekötik azokat az adapteren található szalagkábel-csatlakozókkal.

Csatlakoztassuk a megfelelő JTAG csatlakozóhoz a programozót. (A JTAG\_1 feliratú csatlakozóhoz lehet csatlakoztatni az ATATMEL-ICE-Basic programozót, a JTAG\_2 feliratúhoz pedig AVR Dragon csatlakoztatható.)

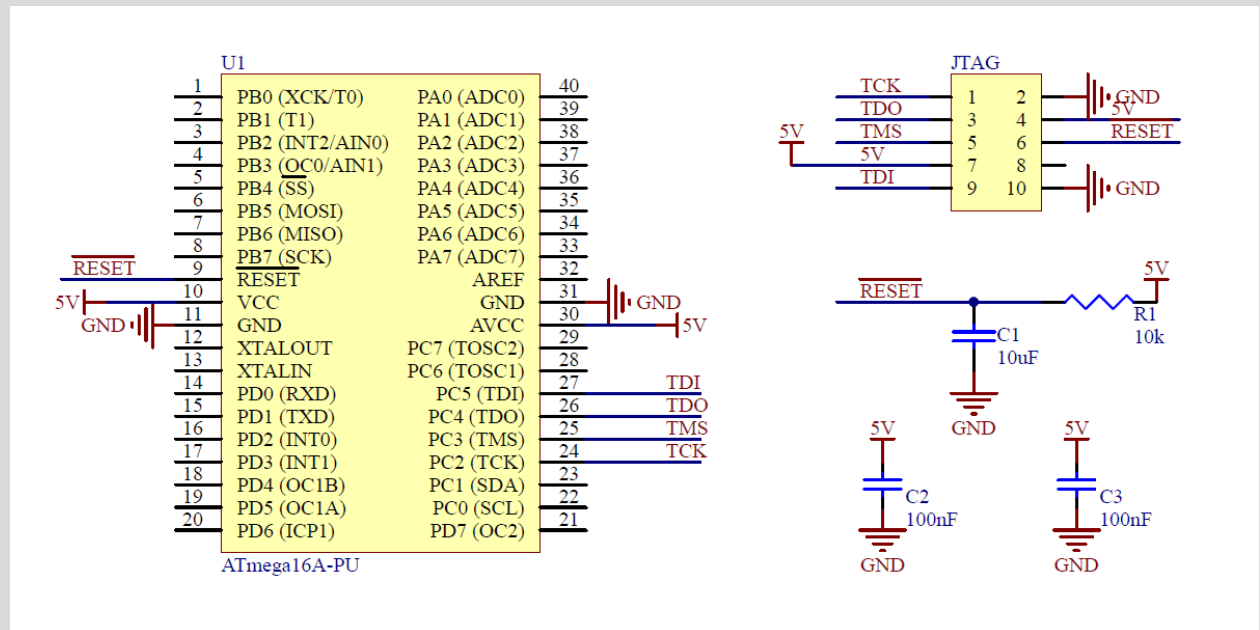


**17. ábra** - az ATATMEL-ICE-Basic programozó csatlakoztatása az adapterhez

## Kitekintés

### A programozó adapterek felépítése

Az adapter a mikrovezérlő lábait köti össze a programozó megfelelő lábaival. Mindössze néhány vezetékről van szó, de ezeken keresztül nem csak programozni tudjuk a mikrovezérlőt, hanem rengeteg adatot is képesek vagyunk kiolvasni belőle működés közben. Később látni fogjuk, hogy ez nagy mértékben képes megkönnyíteni a tervek megvalósítását, vagyis az implementálást.



18. ábra - a JTAG programozó adapter kapcsolási rajza

(A GND-eket azért kötjük össze az összes eszközben, mivel ez a referenciafeszültség, mindenhol ehhez képes számít egy-egy jel alacsony jelszintűnek, vagy magasnak.)

A VCC vezetéket azért kötjük össze a programozóval, hogy a programozó ellenőrizni tudja, hogy megfelelő tápfeszültséget biztosítunk-e a mikrovezérlőnek. (A mi esetünkben 5 Voltot.) Ha a tápfeszültség túl alacsony lenne, akkor a programozó károsíthatná a vezérlőt, ezért van szükség ennek automatikus ellenőrzésére.

A /RESET lábat például akkor használja a programozó, ha újra akarja indítani a vezérlőt. A TCK, TDO, TMS, TDI és TRST a JTAG-szabványban definiált, kommunikációra használt lábak, ezek közül az ATmega16A csak az első négyet használja.

## 5V TÁPFESZÜLTÉS ELŐÁLLÍTÁSA AZ ADAPTERREL

Az nagyobbik adapter további segítséget nyújt az elrendezés könnyű összeállításához, mivel található rajta a korábban megismert 7805-ös kapcsoláshoz hasonló áramkör, mely előállítja a mikrovezérlőnek szükséges 5V-ot, és a breadboardon keresztül a megfelelő lábakhoz kapcsolja. (Ezt a multiméterrel akár le is ellenőrizhetjük.)



A tápáramkör bemenetére többféleképp is adhatunk tápfeszültséget, a lényeg, hogy 7V és 20V közé essen, az ideális a 7-12V-os tartomány.

Csatlakoztathatunk egyenáramú adaptert - ha megfelelő a csatlakozója - a J2 jelű csatlakozóhoz. Köthetünk akár ehelyett egy 9V-os elemet is a J1 jelű zöld sorkapocsba. Ezeknek az eseteknek megvan az a hátránya, hogy nincs áramkorlát a rendszerben, így egy-egy nem kívánt rövidzár, vagy az alkatrészek helytelen bekötése tönkretehet valamit.

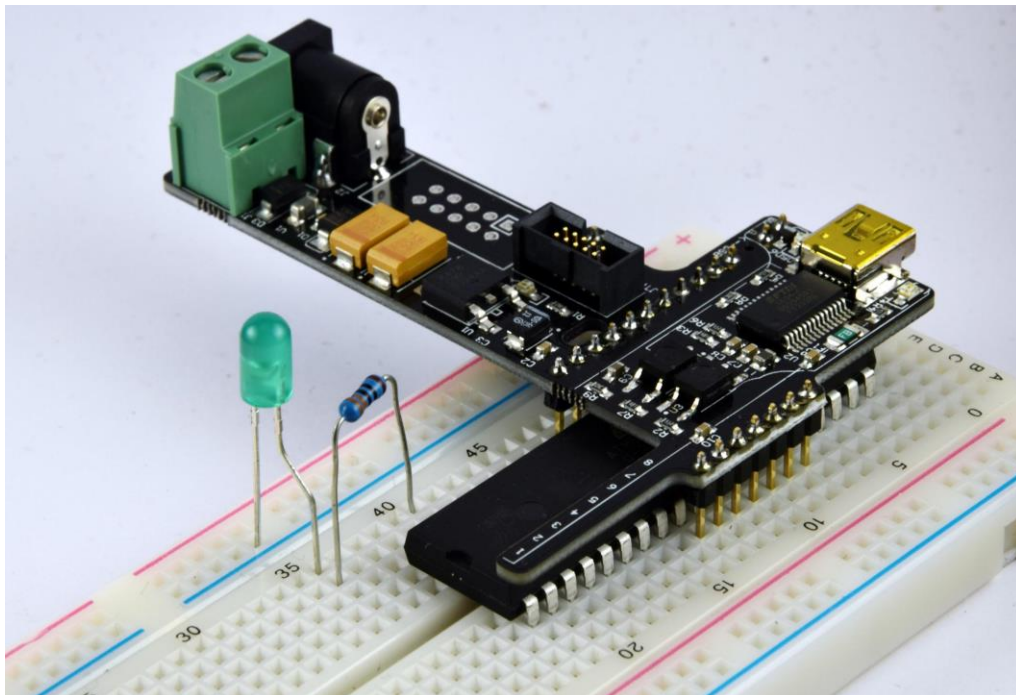
Az ideális, ha a J1 csatlakozónak laboratóriumi tápegységről tudunk feszültséget adni, és a labortáp áramkorlátját előzőleg kis értékre; körülbelül 70-100 mA-re állítjuk.

A tápvezetékek csatlakoztatásakor ugyan figyelni kell a polaritásra, de az adapter a tápbemeneten egy sorba kötött diódát is tartalmaz, így fordított bekötés esetén nem fog folyni áram, nem tud semmi tönkremenni, míg helyes bekötés esetén egy zöld LED fog világítani az adapteren.

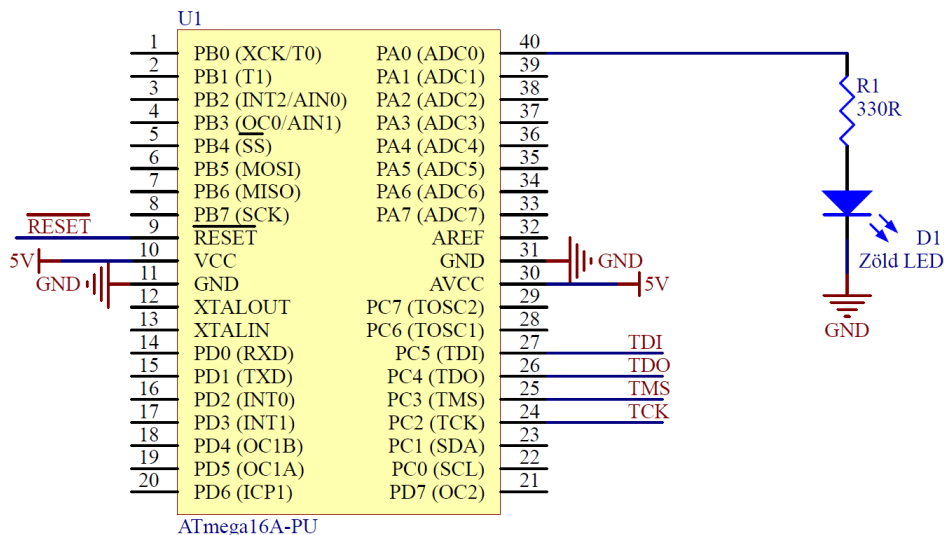
Az adapteren található USB-csatlakozót a tananyag későbbi részeiben fogjuk használni. Figyeljünk arra, hogy a tűkesornak az USB-csatlakozó melletti szakaszát egyenlőre ne kössük össze semmivel a breadboardon!

## LED BEKÖTÉSE

Csatlakoztassunk egy LED-et és egy 330 Ohm-os ellenállást a mikrovezérlő PA0-s lába és a GND közé úgy, hogy a LED a későbbiekben majd akkor világítson, ha a lábon 5V jelenik meg! (Fordított polaritás esetén a LED nem fog világítani.)



*19. ábra - LED bekötése a mikrokontroller PA0-s lábára*



20. ábra - LED-es kapcsolás rajza

## FEJLESZTŐKÖRNYEZET TELEPÍTÉSE

A hardverünk összeállt, viszont még rá kell töltenünk a kívánt működést eredményező firmware-t (mikrovezérlőn futó kisméretű szoftverek általános neve). Ehhez az Atmel cég biztosít számunkra a PC-re telepíthető, minden igényt kielégítő szoftvert, ezt hívjuk fejlesztőkörnyezetnek.

A tananyag során mi az Atmel Studio v6.2.1563-as verzióját fogjuk használni, amely letölthető a <https://www.microchip.com/mplab/avr-support/avr-and-sam-downloads-archive> weboldaltól. A Microchip folyamatosan fejleszti a szoftvert, aki érzi magában az erőt, használhatja a legújabb verziót is. Ekkor sajnos az esetleges kompatibilitási problémák miatt nem tudjuk garantálni a példaprogramok hibátlan működését. Amennyiben gondod támadna a letöltéssel vagy telepítéssel kapcsolatban, részletes útmutatót találsz a tananyag honlapján az aktuális résznél.

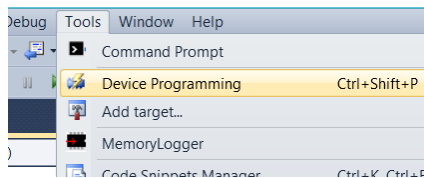
## LED-et villogtató szoftver feltöltése a mikrovezérlőre

Az egyszerűség kedvéért a legelső szoftvert tartalmazó projekt tömörítve letölthető a honlapról. Ennek a szoftverködnak a megértését, illetve saját programok írását csak a következő tananyagrészekben kezdjük, azonban most megtanuljuk, hogy hogyan lehet feltölteni a mikrovezérlő eszközre a kódot. Miután kicsomagoltuk a fájlt, nyissuk meg a *KE9\_1\_LED\_villog\_delay* nevű Atmel Studio solution fájlt.

Debug	2015.04.06. 10:46	Fájlmappa
Headers	2015.04.06. 10:46	Fájlmappa
Source	2015.04.06. 10:46	Fájlmappa
KE9_1_LED_villog_delay	2015.04.06. 9:56	ATMEL Studio 6.2 ... 1 KB
KE9_1_LED_villog_delay	2015.04.06. 9:56	ATMEL Studio 6.2 ... 7 KB

21. ábra - az első programunk projektjéhez tartozó fájlok

Győződjünk meg róla, hogy a mikrovezérlő táplálbai a helyes tápfeszültséget kapják, a programozó pedig helyesen van csatlakoztatva a mikrovezérlőhöz. Csatlakoztassuk a programozót a számítógépünkhöz. Ha először programozzuk az eszközt, mindenképp válasszuk ki a "Tools" menü "Device programming" menüpontját.

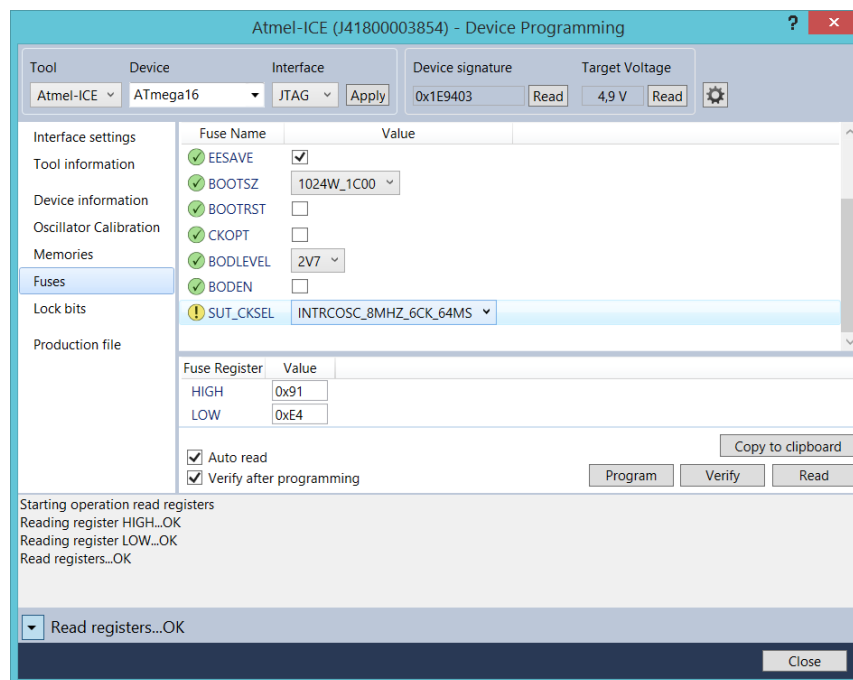


**22. ábra** - Itt található a Device programming menüpont

Itt állítsuk be a rendszerünk paramétereit.

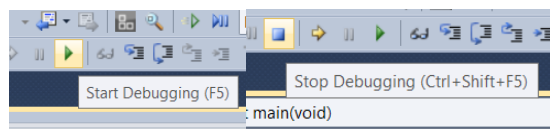
- Tool: Atmel-ICE
- Device: ATmega16
- Interface: JTAG

Ez után nyomjunk az “Apply” feliratú gombra, melynek hatására egyből ki is olvasunk néhány adatot a mikrovezérlőből. Ezen az ablakon belül a “Fuses”-ra kattintva láthatjuk a mikrovezérlő alap beállításait. Itt keressük meg az órajel frekvenciát (“SUT\_CKSEL”), és állítsuk át a belső oszcillátorról érkező 8MHz-es órajelre. Más értéket ne állítsunk át! Ahhoz, hogy a változtatás érvénybe lépjen, kattintsunk a “Program” gombra, majd zárjuk be a “Device programming” ablakot.



**23. ábra** - a Device programming ablak, és az órajel frekvencia átállítása

A már megírt szoftvert úgy tölthetjük legegyszerűbben az eszközre, ha először a “Start Debugging”, majd a “Stop Debugging” gombokra kattintunk.



**24. ábra** - a kód feltöltése és elindítása



Ha mindent jól oldottunk meg, akkor a számítógépünk nem dobott hibaüzenetet, a LED-ünk pedig 2 másodperces periódusidővel villog. Ez után szabad út áll előttünk a programozás megértéséhez, és a legkülönbözőbb perifériák illesztéséhez!

## **Kitekintés**

### **Hasznos tippek**

Soha ne írjunk ékezetes betűket, illetve speciális karaktereket a fájljaink nevébe, illetve elérési útjába. Ezek hatására a várttól eltérően működhet a fejlesztőkörnyezet, esetleg előfordulhat, hogy nem tudjuk letölteni a programunkat a mikrovezérlőre.

Fejlesztés során lehetőleg mindig kis lépésekben haladjunk. Ha mindig csak keveset változtatunk egy korábbi működő verzióhoz képest, akkor könnyebb lesz felfedezni az új részekben az esetlegesen bevitt hibákat, legyen szó akár hardverről, akár szoftverről.