

SPI programozó ATMEL 90Sxxxx, AT MEGA, AT TINY mikrokontrollerekhez

Az ismertetett áramkör és program lehetővé teszi az ATMEL AVR mikrokontrollerek többségének a soros, "in circuit" kiolvasását, beégetését.

Röviden az ATMEL AVR mikrokontrollerekről:

Az ATMEL AVR mikrokontrollerek:

Az ATMEL 90S1200, 90S2313, 90S4414, 90S8515, valamint az ezeket felváltó AT MEGA 8515, AT TINY 2313 AVR mikrokontrollerek lábkompatibilisek az ATMEL 89-es sorozat 89C51, 89C2051, stb. tagjaival, de más a belső felépítésük, és úgynevezett RISC utasításkészlettel programozhatóak.



Az IC-k egy elektromosan törölhető, és újraprogramozható FLASH program és adat memóriával rendelkeznek.

A soros, és a párhuzamos programozás:

Az ATMEL a legtöbb AVR mikrokontrollerhez két programozási lehetőséget is biztosít, a soros, és a párhuzamos programozást.

A párhuzamos programozás esetén a beprogramozandó adatokat bájtanként írjuk be, azaz egy ütemben 8 bitet. Hátránya, hogy a programozáshoz sok kivezetésre van szükség, ami viszonylag bonyolult programozó áramkört igényel. Előnye hogy valamivel gyorsabb, mint a soros programozás. De talán ennél sokkal fontosabb, hogy bizonyos paraméterek csak a párhuzamos programozással állíthatóak be. (Amiből aztán problémák adódnak, de erről majd később...)

A soros programozási algoritmus szerint a beírandó adat bájtokat bitekre bontva, sorban egymás után, egy órajellel szinkronizálva kell beléptetni. Csak néhány kivezetés kell ez esetben, ezek egy SCK órajel, egy MOSI és MISO adatvonal, és egy RESET jel.

Az előbbiekből adódik a soros programozás egyik előnye, hogy lehetőséget a mikrokontroller áramkörben - in circuit - programozására, és a programozó áramkör hardvere is nagyon egyszerű lehet. Ha egy mikrokontrollert az áramkörben akarjuk programozni, akkor erre már jó, ha a tervezéskor is figyelünk! Érdemes a szükséges portokat egy csatlakozóra kivezetni, és vegyük figyelembe, hogy ezeken a kivezetéseken jönnek (mennek) majd a vezérlőjelek.

A konfigurációs, és „code protect” bitek:

A beégetett program működését alapvetően befolyásolják, az úgynevezett "fuse bitek". A korábbi kibocsátású IC-knél még nem, vagy csak 1-2 ilyen bit volt, az újabbaknál egész sor. Pl. a 90S8515-nél még

egy se, az "utódjánál", az AT MEGA 8515-nél már 16 is van. A probléma az, hogy ezek a bitek a forrásprogramban nem adhatóak meg, mint pl. a MICROCHIP PIC-eknél. (Szerintem ez óriási hiba! Az IC felprogramozása így sokkal körülményesebb, és nagy a hibalehetőség is.) Tehát ha ezek a konfigurációs bitek nem helyesen vannak beállítva, akkor a mikrokontrollerbe égetett program nem, vagy nem helyesen működik!

Mint minden újabb mikrokontrollernél, itt is lehetőség van a beprogramozott adatok kiolvasásának letiltására. Ha "Code protect" bitet beprogramozzuk, akkor beégetett tartalom nem olvasható vissza, helyesebben, ha kiadjuk az olvasás parancsot, akkor vagy csupa FF-et látunk - pl. a 90S2313 esetében - vagy látszólag "kijönnek" adatok - pl. a 90S8515 esetében - de azok "fals" értékek. A 90S8515-nél még a típus azonosítás is le van tiltva, ha titkosítottuk az IC-t. (Ez szerintem egy "gyári hiba", a többi típusnál a típus akkor is azonosítható, ha a code protect be van kapcsolva.)

(Gyakran kapom a kérdést, hogy akkor hogy lehetne mégis kiszedni az adatokat, egy titkosított chipből. Sehogy. Pont erre találták ki...) A törlés - erase - funkcióval törlődnek a titkosító bitek - de egyben a chip tartalom is.

Az SPI enable bit:

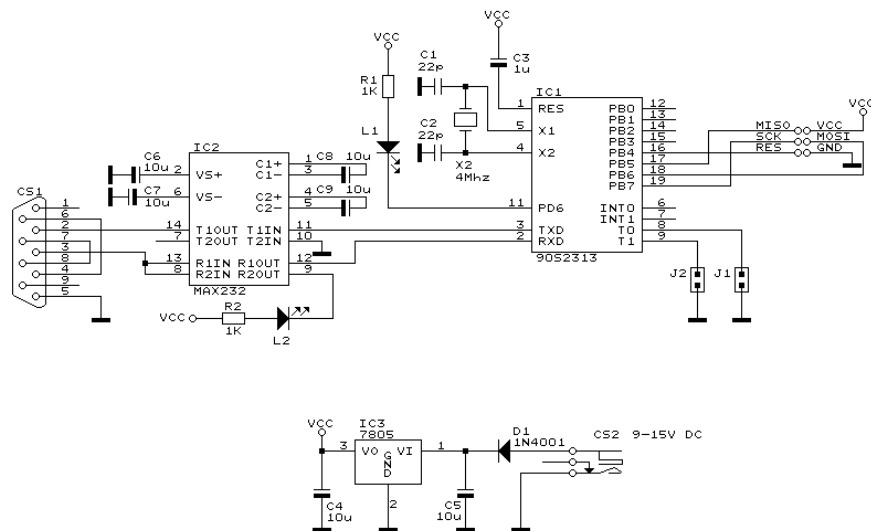
Az SPI konfigurációs bittel engedélyezhető, illetve tiltható le a soros programozás. Ezt a bitet azonban csak a párhuzamos programozási algoritmusban tudjuk átírni! Ha tehát ez bit törölt, akkor egy soros programozóval az IC nem írható, és nem is törölhető! Az újonnan vásárolt IC-kben az SPI engedélyezve van, tehát nincs gond a programozással. De egy korábban már beprogramozott IC újraindítását, törlését megakadályozhatja, ha az SPI bit törölve lett. (A megoldás ez esetben egy párhuzamos programozóval törölni az IC-t, majd az SPI bitet beírni.)

A programozó áramkör:

Ahogy látható, a felépítés nagyon egyszerű, egy MAX232-es illeszti a PC soros portját az égetőt vezérlő 90S2313-as mikrokontrollerhez. Ha a PC-ről adat érkezik, azt az L2 LED felvillanásai jelzik.

Az SPI jelek egy 3x2-es „tüskére” vannak kivezelve. Innen egy darab 6 eres szalagkábellet - felszerelve egy 10-es szalagkábeles anya csatlakozóval - kapcsolódhatunk a programozandó mikrokontrollert tartalmazó áramkörhöz.

A vezérlő mikrokontroller tápfeszültségét egy 7805 stabilizálja. A szükséges tápegység paraméterei: 9V-20V egyenfeszültség, 300mA. A fordított tápfesz ellen megvéd a D1-es. (Egy átlagos - stabilizálatlan - DC 9-12V-os hálózati adapter e fenti igényt általában kielégíti.)



A J1-J2 jumpert, és az L1 LED-et jelenleg nem használja a vezérlőprogram, ezek esetleges későbbi fejlesztéshez vannak fenntartva, ezért beültetésük felesleges!

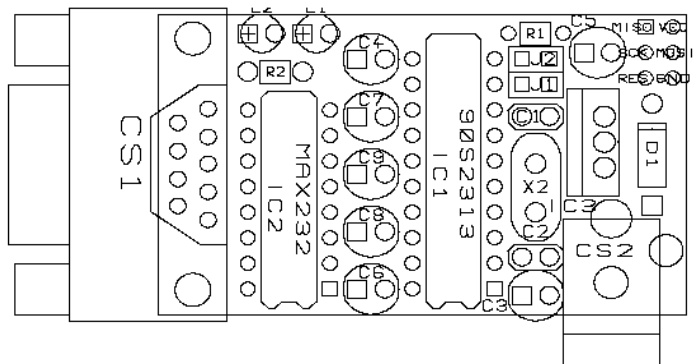
A 40 lábú IC-k kezelése:

Először is a programozó, és a programozandó DIL 40 tokos IC összekötése:

Programmer: IC

- | | | |
|------------|---|-----|
| 1. (MISO) | - | 7. |
| 2. (VCC) | - | 40. |
| 3. (SCK) | - | 8. |
| 4. (MOSI) | - | 6. |
| 5. (RESET) | - | 9. |
| 6. (GND) | - | 20. |

A 18 - 19 lábra egy kvarcot kell kötni!

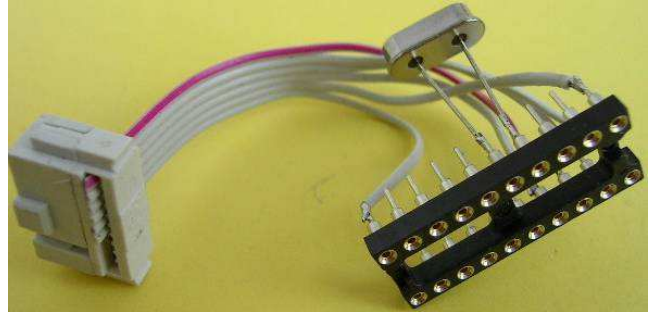


A 20 lábú IC-k kezelése:

A programozó, és a programozandó DIL 20-as tokozású IC összekötése:

Programmer: IC

- | | |
|--------------|-----|
| 1. (MISO) - | 18. |
| 2. (VCC) - | 20. |
| 3. (SCK) - | 19. |
| 4. (MOSI) - | 17. |
| 5. (RESET) - | 1. |
| 6. (GND) - | 10. |



A 4. - 5. lábra egy kvarcot kell kötni!

A 8 lábú IC-k kezelése:

A 8 lábú AT TINY12, 15 típusok két módon is égethetőek. Az LV - low voltage - az alacsonyfeszültségű égetésnél 5 voltos az égetőfeszültség, (Hasonlóan a soros EEPROM-okhoz.) Egy ábra erről a program HELP-jében is megtekinthető. A másik égetési mód a "HV" - high voltage - azaz magasfeszültségű mód, amikor az égetőfeszültség 12.5 voltos. Ha e módon akarjuk a TINY12, 15-öt égetni, akkor ahhoz egy adapter kell, ami a 40-es karos foglalatba kerül, az IC pedig az adapter 8 lábú foglalatába. Felmerül a kérdés: miért kellett az LV és HV programozási algoritmust is megcsinálni, plusz az adaptert a HV programozáshoz? Azért, mert bizonyos funkciók csak a HV programozásban érhetőek el, mint pl. a "Reset disable" bit átírása. Pl. ha a "Reset disable"-t beprogramozzuk, akkor ezzel az LV programozást is lehetetlenné tesszük, így ezután az IC-t LV módban se újra programozni, se törölni nem tudjuk! Ekkor kell a HV programozás az adapterrel... (Fogalmam sincs miért csinálta így az ATMEL, ez szerintem egy óriási baklövés a részükről ...)

Az AVRPROG.EXE program használata:

A program funkciói: olvasás, égetés, törlés, a beolvasott mikrokontroller tartalmának lemezre írása, vagy onnan az égetendő tartalom beolvasása. A program futtatható WIN98 és XP alatt is. Amúgy a PC program az „AVR Studio” fejlesztőrendszer része, így abba integrálható, abból közvetlenül futtatható, stb.

És akkor a használat, egy konkrét példán keresztül:

Legyen a feladat, egy „SPEED5.HEX” program - ez egy áramkörünk sebességmérő programja, de ez most lényegtelen - beégetése egy 90s2313-as mikrokontrollerbe!

- Csatlakoztassuk az égetőt a PC soros portjához, az égetendő IC-hez, és a táphoz! (Ha nem akarunk a számítógép hátsó csatlakozóin matatni, használjunk egy soros port toldó kábelt. Sokkal könnyebb a kezelés, és máskor is jól jön, ha kényelmesen elérhető a soros port.)

- Most kapcsoljuk be - dugjuk be - a tápot. A sorrend fontos! Ahogy a legtöbb számítógépre csatlakozó periféria leírása is tartalmazza, ne kössünk össze táp alatt lévő készülékeket!

- Indítsuk el a programot! A program megvizsgálja, hogy van-e csatlakoztatva égető. Ha nem talál, akkor ezt jelzi:

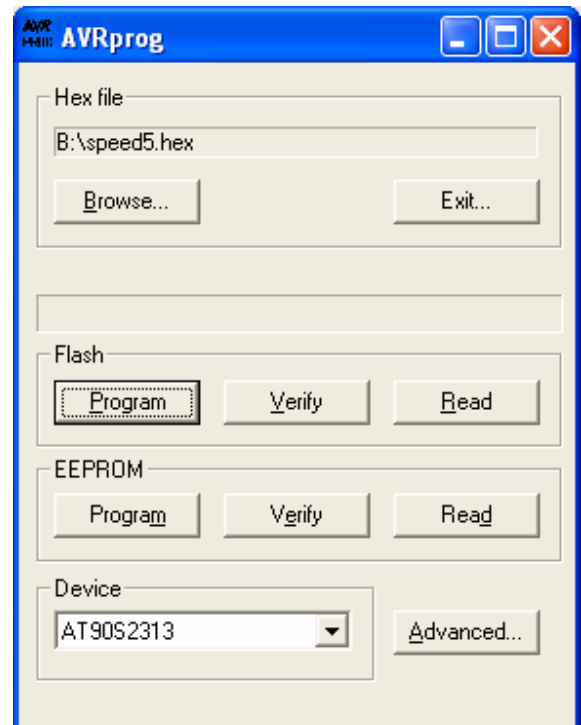
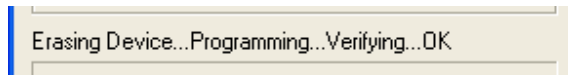


(Ez esetben nézzük át a csatlakoztatást, a tápot. Ha ez rendben, akkor próbáljuk kideríteni, nem él-e egy korábban telepített eszköz meghajtó programja a porton.)

Ha a program „talált” égetőt, megjelenik a bejelentkező kép:

- Állítsuk be a mikrokontroller típust, jelöljük ki az égetendő file-t !

- És most adjuk ki a „Program” parancsot. Az égető törli, programozza, majd leellenőrzi az égetést. Ha minden rendben, OK :

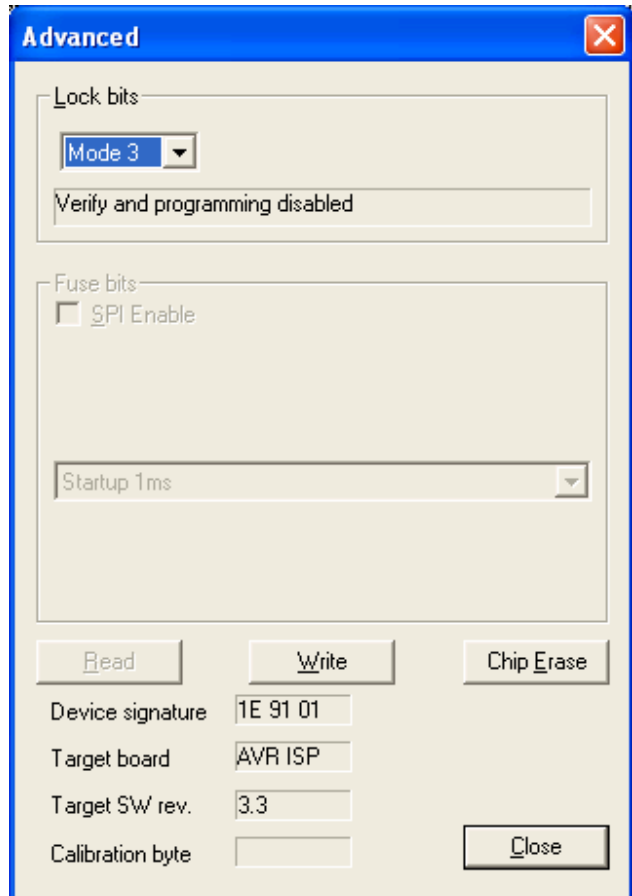


Az „Advanced” menüben tudjuk a code protect, és konfigurációs biteket állítani. Helyesebben azokat, amelyeket a soros programozás megenged...

A titkosítást a „Lock bits” beírásával érhetjük el.

Az SPI bit nem állítható. (Ez talán nem is baj, így véletlenül se tudjuk magunkat kicsukni a későbbi programozásból.)

Ugyanitt látható a 90s2313-ból kiolvasott IC azonosító, és az égetőt vezérlő programok verziószáma.



Használat USB portról:

Ha nincs szabad soros port a gépünkön, akkor használhatjuk az égetőt USB porton is, de ehhez szükségünk van egy USB/Serial port átalakítóra. A több gyártó is csinál ilyent, a kereskedelemben pár ezer forintért kaphatóak. (Én egy „noname” , 3500 forintos átalakítóval teszteltem – sikeresen.)

Epilógus:

- Még egyszer leírom - bár az előbbiekből is kiderül - hogy a működtető program(ok) fejlesztése (munkája, érdeme, stb.) tőlem (mikroklubtól) teljesen független, én csak a nyomtatott áramköri tervet, a hardvert, és ezt a dokumentációt készítettem hozzá.

Végül nincs más hátra mint hogy sok sikert kívánjak a használatához. Viszontlátásra: Torkos Csaba 8100 Várpalota Táncsics u. 7. Telefon: napközben: 88/473-784, egész nap: 06/30/9472-294, email: mikroklub@vnet.hu internet: <http://www.eprom.hu>, <http://www.mikroklub.hu>.