

UPP628 KÉZIKÖNYV

A programozó beüzemeléséhez először telepítsd a kezelőprogramot, amit innen tudsz letölteni:

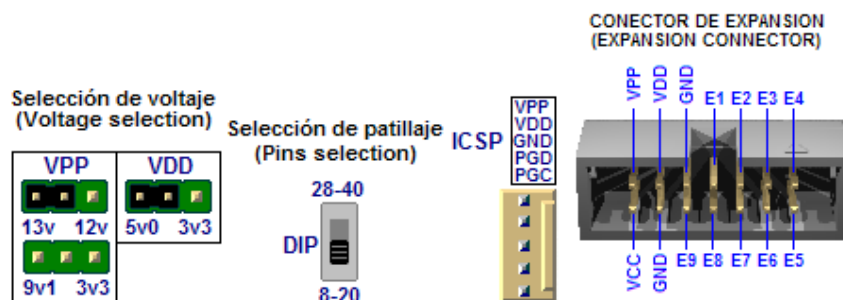
<http://www.epromirok.hu/downloads/UPP628.zip>

[Általános információk]

Mielőtt a programozóra csatlakoztatnál vagy a foglalatba helyeznél egy programozandó eszközt, olvasd el az eszközre vonatkozó beállításokat, és győződj meg róla, hogy a programozó annak megfelelően lett beállítva (táp feszültség VDD, programozási feszültség VPP és a foglalatba helyezés megfelelő módja/iránya: (A kapcsoló 8-20 állásban legyen 8, 14, 16, 18 és 20 lábú chipek esetén; 28-40 állásban legyen 28 vagy 40 lábú chip esetén) ha a programozás a programozó ZIF foglalatában történik.

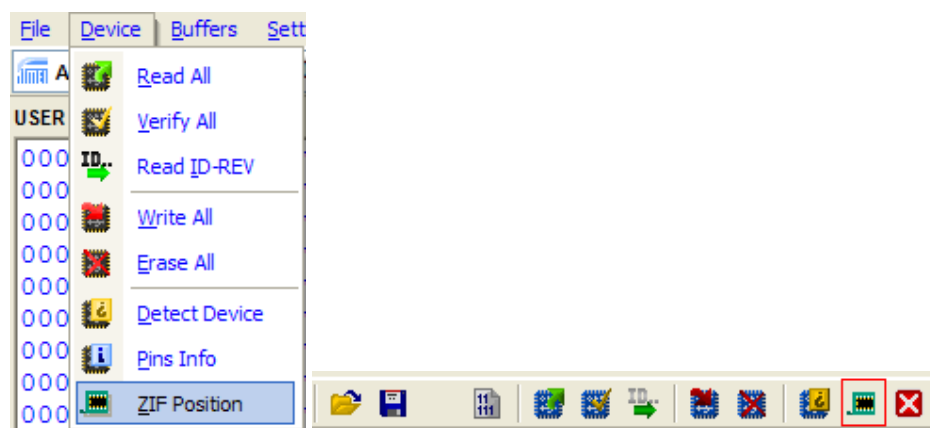
ICSP (rendszerben történő programozás) esetén ennek a kapcsolónak az állása mindegy.

FONTOS, HOGY AZ ALÁBBI UTASÍTÁSOKAT BETARTSD A PROGRAMOZÓ ÉS A PROGRAMOZANDÓ ESZKÖZ VÉDELME ÉRDEKÉBEN!



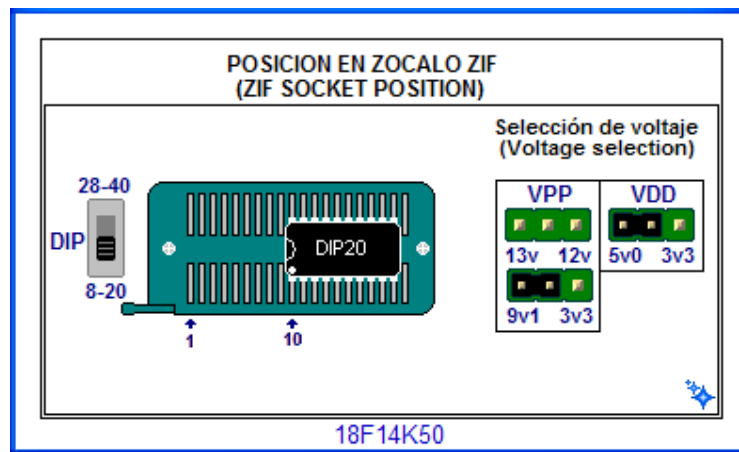
[Behelyezés a ZIF foglalatba és a szükséges beállítások]

Nem minden eszköz programozható a ZIF foglalatba helyezve, illetve amelyik a foglalatba helyezve programozható annak sem mindegy a behelyezés iránya, pozíciója. ICSP kapcsolat esetén is szintén az alábbi menüpont használata szükséges a megfelelő beállítás elvégezhetőségéhez: (Eszköz -> Programozóra helyezés módja)

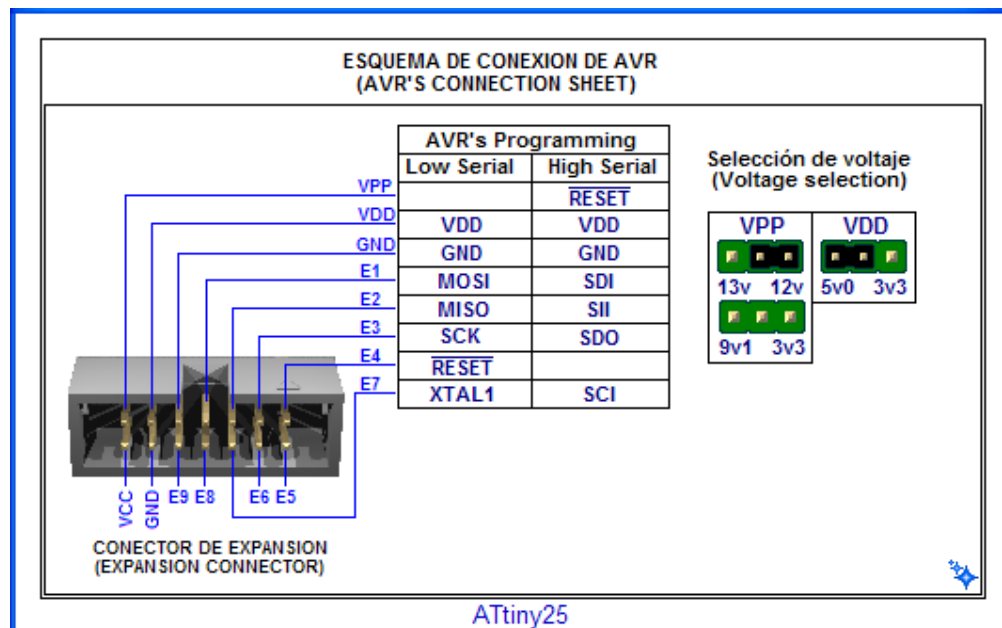


Itt nézhető meg az adott eszköz behelyezési/bekötési módja és a szükséges jumper beállítások a megfelelő VPP és VDD beállításához.

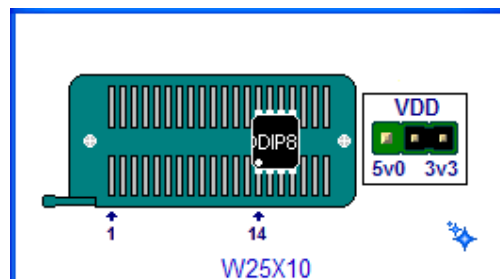
Alább látható, hogy pl egy **PIC18F14K50** (Microchip) hogyan helyezhető a foglalatba és hogyan kell a VPP és VDD értékeket beállítani, illetve a DIP kapcsoló segítségével a megfelelő chip láb számot meghatározni (Ez a chip 20 lábú):



Az alábbi ábrán megtekinthető pl az **ATtiny25** (Atmel) bekötési módja:



Egy másik példa a **W25X10** (Winbond) eeprom behelyezésére és beállítására:



[Microchip PIC és dsPIC]

Nem azonosíthatók.-

Az alábbi eszközök automatikusan nem azonosíthatók mert nem rendelkeznek eszköz azonosítóval, így manuálisan kell kiválasztani őket:

PIC10F200, PIC10F202, PIC10F204, PIC10F206

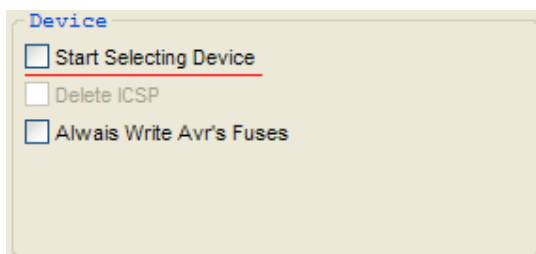
PIC10F220, PIC10F222

PIC12F508, PIC12F509, PIC12F510, PIC12F519

PIC16F505, PIC16F506, PIC16F526, PIC16F54, PIC16F57, PIC16F59

PIC16F83, PIC16F84

Mielőtt behelyeznéd ezen eszközök egyikét a ZIF foglalatba, vedd ki a pipát a Beállítások->Eszköz kiválasztás menüpont rubrikájából, hogy az érzékelő algoritmus megváltoztassa a flash értékeket.



ICSP Programozhatóság.-

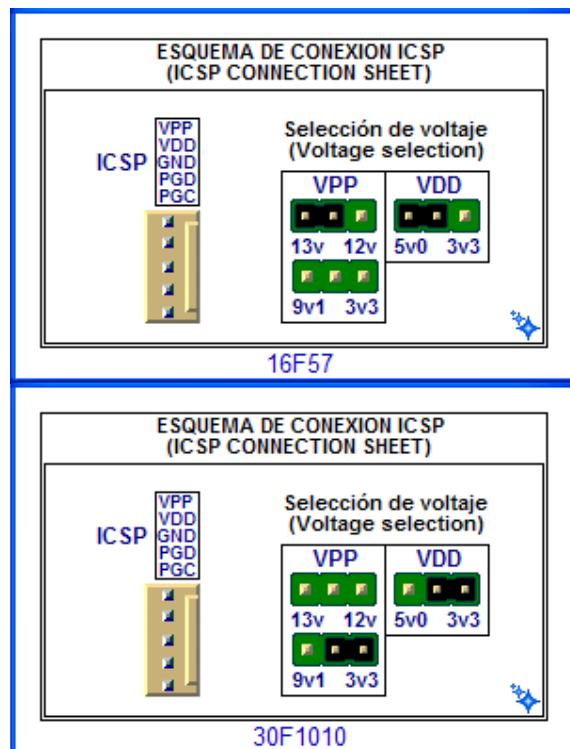
Az alábbi eszközök programozhatók az 5lábú ICSP csatlakozó segítségével.

Minden táp és föld láb a Microchip specifikációk szerint kell, hogy bekötésre kerüljön.

A VPP és VDD értékeket pontosan be kell állítani.

PIC16F57, PIC16F59, PIC18FxxJxx, PIC18LFxxJxx

dsPIC30Fxxxx, dsPIC33Fxxxx

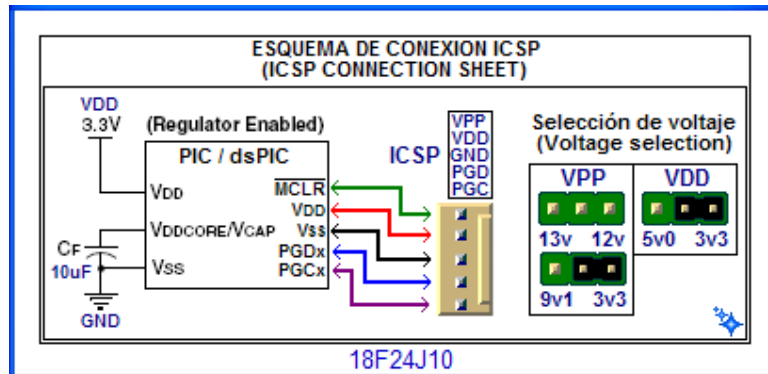


VPP = 13v, VDD = 5v

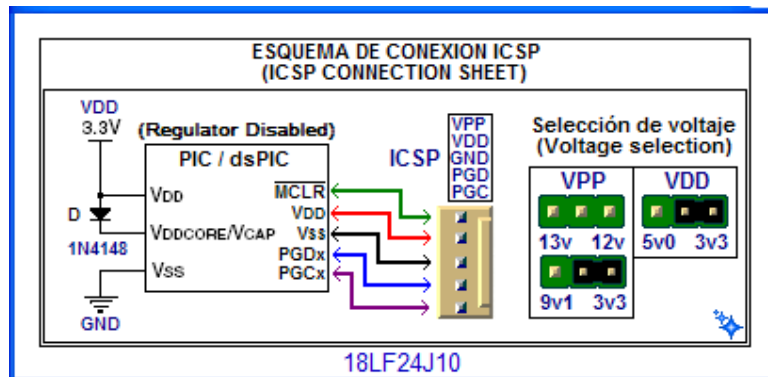
VPP = 3v3, VDD = 3v3

Kettős feszültség értékkel.-

A **PIC18FxxJxx**, **PIC18LFxxJxx** és **dsPIC33Fxxxx** két tápot igényel a működéséhez, VDD (3.3v) és VDDCORE/VCAP (2.5v). Ez az IC család ICSP módban programozható és az alábbi ábra szerint kell csatlakoztatni a VDDCORE/VCAP lábakat:



PIC18FxxJxx esetén köss egy 10uFkondenzátort a föld és VDDCORE/VCAP közé



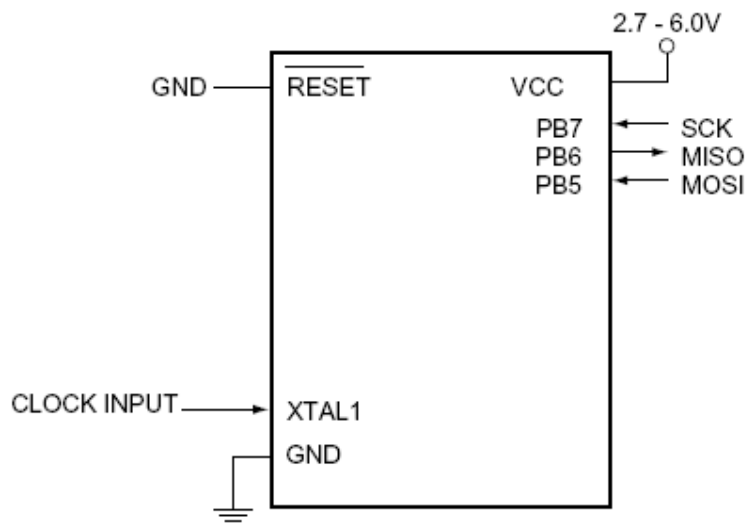
PIC18LFxxJxx esetén köss egy 1N4148 (vagy azzal egyenértékű) diódát a VDD és VDDCORE/VCAP közé

[Atmel AT89xxx és AVR]

Az AT89xxx és AVR eszközök számos programozási módot támogatnak, mely a ténylegesen programozandó eszköztől függ.

Alacsony feszültségű soros programozás.-

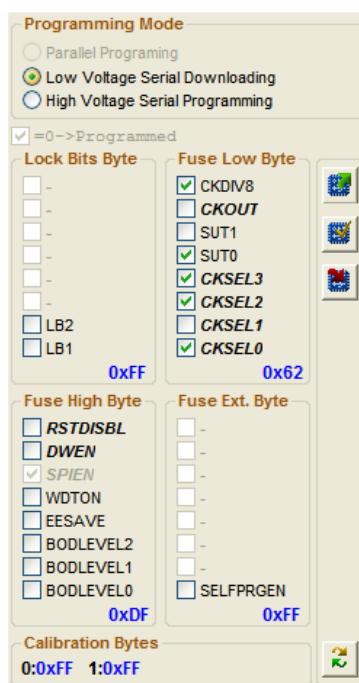
Ez a programozási mód minden AT89xxx és AVR esetén támogatott az alábbi bekötés esetén:



Ha az eszköz belső órajele lett állítva, nem szükséges csatlakoztatni az órajelet az XTAL1 lábra. Minden más esetben elengedhetetlen.

A gyártó által beállított alapértelmezett állapotban az eszköz ezt a programozási módot támogatja. Ennek a programozási módnak vannak megszorításai, mely a programozandó eszköztől függ. Ezek megismeréséhez mindenképp olvasd el az adott chip adatlapját.

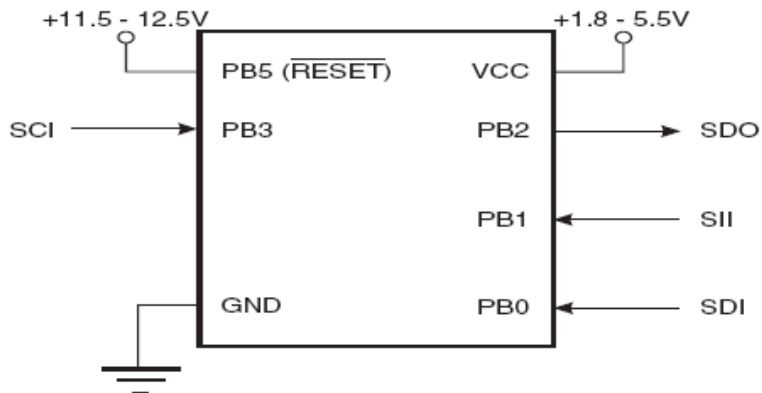
Minden esetben nagyon körültekintően bánj a fuse bitekkel, mert módosításukkal akár le is tiltható ez a programozási mód. Ezek a fuse bitek vastag dőlt betűkkel vannak jelölve, mint ahogy a lenti képen is látható:



A fuse bitek módosítása előtt mindig tanulmányozd át az eszköz adatlapját, hogy megismerhesd azok funkcióját.

Nagy feszültségű soros programozás

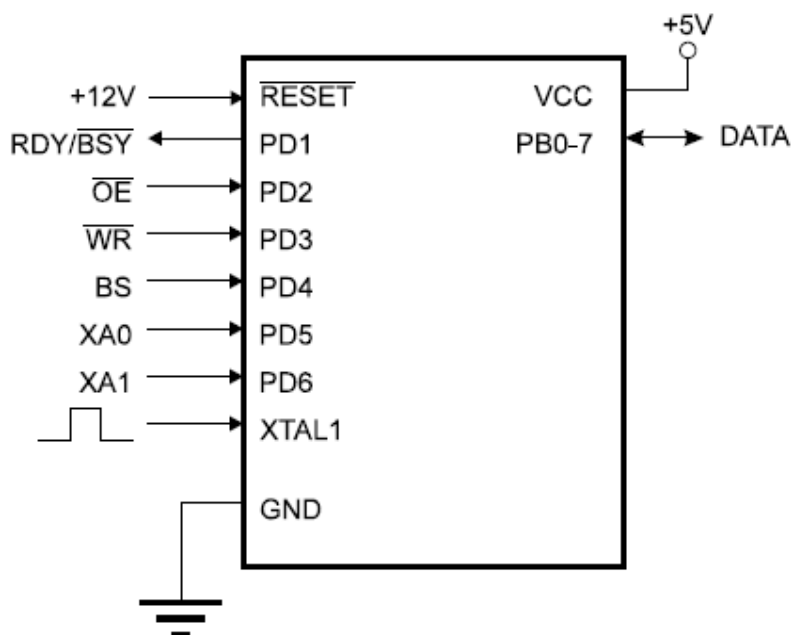
Számos 20 vagy több lábbal rendelkező AVR támogatja ezt a programozási módot. A fuse bit beállításokkal kapcsolatban nem rendelkeznek megszorításokkal. Példa:



Párhuzamos programozás

Általában ez a programozási mód 20 vagy több lábbal rendelkező eszközök esetén támogatott, de **jelenleg a programozó nem támogatja. Várhatóan egy későbbi programozó adapter segítségével támogatottá fog válni.**

Egy bekötési példa:



[Freescale M68HC908]

Bemutató

Ezek az eszközök beépített ROM-mal rendelkeznek, mely rutinjai elérhetik a flash memóriát olvasásra, írásra, törlésre és ellenőrzésre. Ezek más néven 'monitor ROM' vagy MON08.

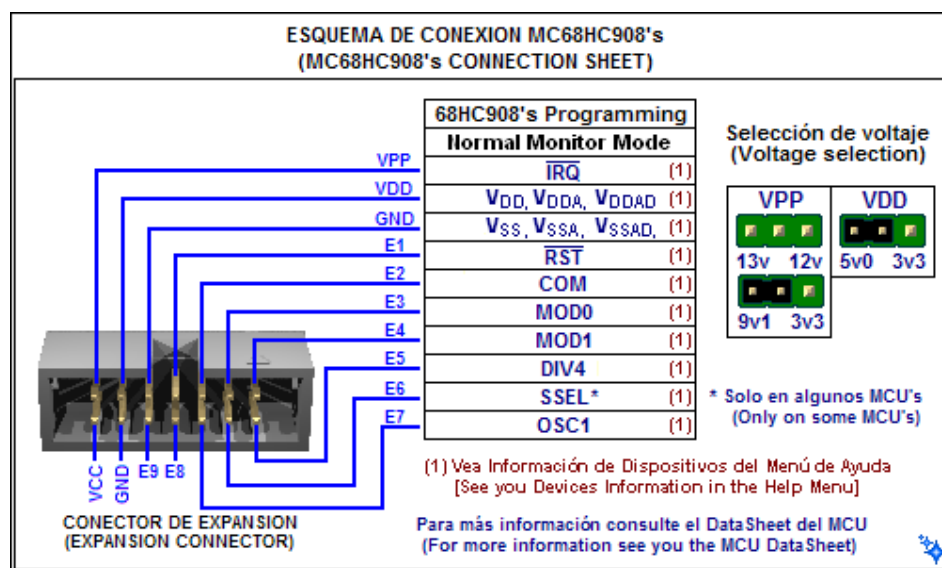
A flash memória 8 biztonsági byte-tal védett, hogy egyszerű felhasználók engedély nélkül ne módosíthassák annak tartalmát. A biztonsági kulcs megkerülése nélkül csak a teljes törlés 'mass erase' engedélyezett, mellyel az MCU flash területe 0xFF értékre állítódik.

Ennek az MCU családnak a programozásához monitor módba kell lépni (monitor rom vagy MON08) az alábbi láb beállításokkal:

- VDD : Táp feszültség (5v.)
- VSS : Föld (0v.)
- IRQ : Normál monitor módban történő programozás (9v.)
- RST : Reset
- COM : Két irányú, fél duplex kommunikáció
- MOD0 : Programozási mód 0.
- MOD1 : Programozási mód 1.
- DIV4 : Működési frekvencia osztó(2-vel vagy 4-el).
- SSEL : Biztonsági byte bevitel soros vagy párhuzamos mód kiválasztása.
- OSC1 : Külső órajel.

Csatlakoztatás

Az alábbi ábrán megtekinthető az MCU bekötése és a beállítandó VPP ill. VDD értékek ...



Az IRQ láb jel értékétől és a reset vektorok értékétől (0xFFFE és 0xFFFF) függően több módon is beléphetünk 'Monitor Módba'.

Kényszerített mód: Ha a reset vektor 0xFF értékű, lehetséges belépni a kényszerített monitor módba ('Forced Monitor Mode') az IRQ láb feszültség szintjétől függően (ez lehet a VDD vagy VSS az MCU-tól függően).

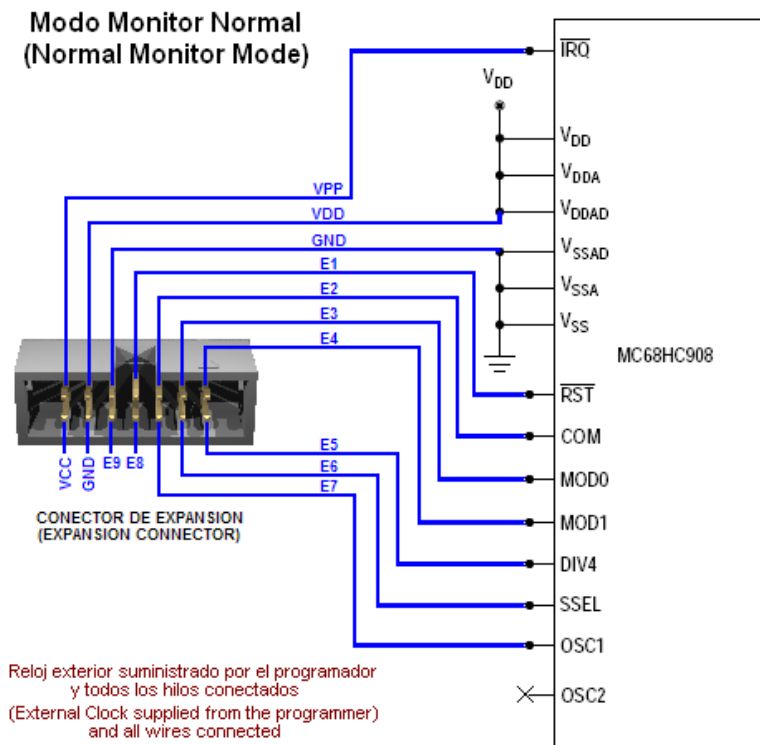
Normál mód: Ha az IRQ láb feszültség szintje a VDD-nél magasabb + 2.5v-tal (kb 9v) beléphetünk a normal monitor módba a reset vektorok értékétől függetlenül.

Nagyon fontos: Minden MCU esetén egyedi láb konfigurációkkal lehet belépni monitor módba, ezért ajánlatos áttanulmányozni az adott eszköz adatlapjában a 'Monitor ROM' részt, hogy megismerhető legyen annak láb konfigurációja, és hogy pontosan beazonosítható legyen minden támogatott monitor mód és a programozóra csatlakoztatás módja.

Alább megtekinthető néhány példa:

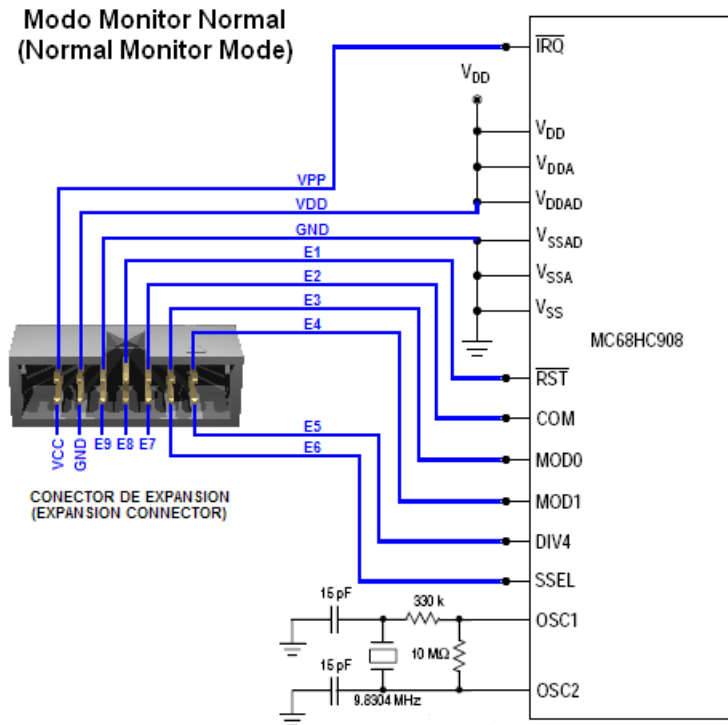
(A) Minden vezeték csatlakoztatva

Ebben az esetben a chip nincs az áramkörre ültetve, vagy kiemelhető a foglalatból vagy a cél áramkör rendelkezik olyan kivezetett csatlakozókkal melyen az összes szükséges láb elérhető és az áramkör többi részéről megfelelő módon le vannak választva ...



(B) Az órajel kivételével minden vezeték csatlakoztatva

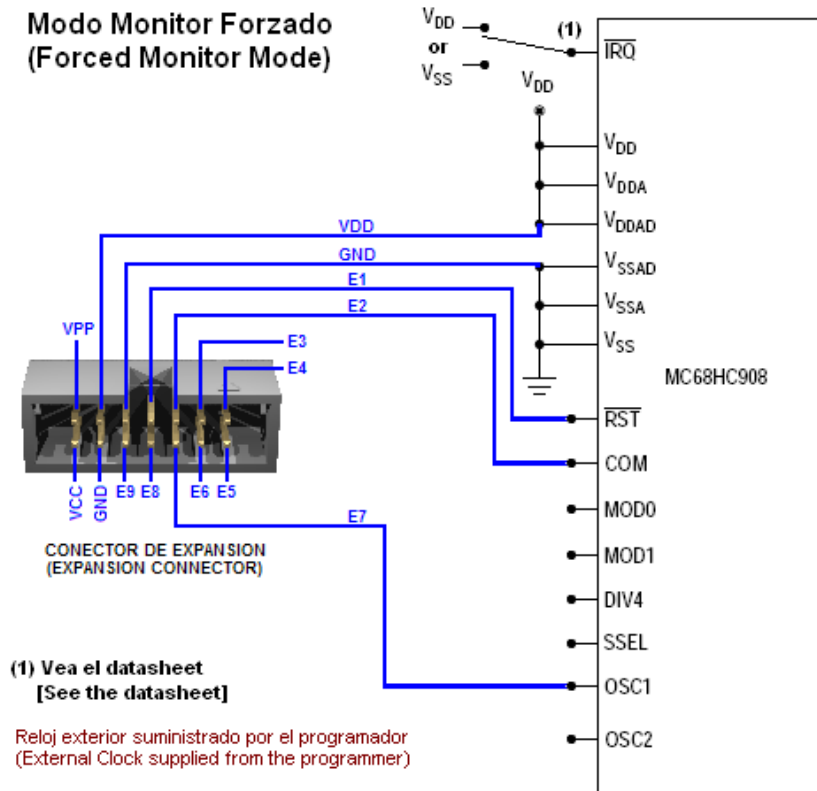
Hasonló a fenti módhoz csak ebben az esetben az MCU órajele a cél áramkörön kerül előállításra. Például olyan alaplap esetén ahol minden szükséges láb az órajel kivételével egy csatlakozóra ki lett vezetve...



(C) Minimális csatlakoztatás

Ebben a kényszerített módban beléphetünk monitor módba a legkevesebb csatlakoztatási pont segítségével. Ebben az esetben az IRQ láb ráköthető a VDD-re, VSS-re vagy NC-re (Nem bekötött). Kérlek olvasd át az eszköz adatlapjában az IRQ csatlakoztatást kényszerített mód esetén...

Modo Monitor Forzado (Forced Monitor Mode)



Láb kiosztás és kommunikáció sebesség

Az alábbi táblázatban megtekinthetők a jelölés kiosztások a programozó és láb nevek között:

Láb név/ szám a programozón	ESZKÖZ CSALÁD											
	AP	BD/GP/GT	EY	GR	GZ	JB/JG	JK/JL/KL	KX	LJ/SR	MR8	MR16/32	QT/QY
VPP	IRQ	IRQ	IRQ	IRQ	IRQ	IRQ	IRQ	IRQ	IRQ	IRQ	IRQ	IRQ
VDD	VDD(*)	VDD(*)	VDD(*)	VDD(*)	VDD(*)	VDD(*)	VDD(*)	VDD(*)	VDD(*)	VDD(*)	VDD(*)	VDD(*)
GND	VSS(*)	VSS(*)	VSS(*)	VSS(*)	VSS(*)	VSS(*)	VSS(*)	VSS(*)	VSS(*)	VSS(*)	VSS(*)	VSS(*)
E1 (RST)	RST	RST	RST	RST	RST	RST	RST	RST	RST	RST	RST	RST
E2 (COM)	PTA0	PTA0	PTA0	PTA0	PTA0	PTA0	PTB0	PTA0	PTA0	PTB1	PTA0	PTA0
E3 (MOD0)	PTA1	PTC0	PTB4	PTB0	PTB0	PTA1	PTB1	PTB0	PTA1	PTB0	PTC3	PTA1
E4 (MOD1)	PTA2	PTC1	PTB3	PTB1	PTB1	PTA2	PTB2	PTB1	PTA2	NC	PTC4	PTA4
E5 (DIV4)	PTB0	PTC3	PTB5	NC	PTB4	PTA3	PTB3	NC	PTC1	NC	PTC2	NC
E6 (SSEL)	NC	PTA7	PTA1	PTA1	PTA1	NC/PTE3	NC	PTA1	NC	NC	PTA7	NC
E7 (OSC1)	OSC1	OSC1	OSC1	OSC1	OSC1	OSC1	OSC1	OSC1	OSC1	OSC1	OSC1	OSC1

(*) Mindegyik láb be kell kötve legyen: Vdd, Vdda, Vddad, Vss, Vssa, Vssad

OSC1 érték (MHZ) a programozó óról	12	12	12	12	12	12	12	12	12	6	12	12
Kommunikáció sebesség (BPS)	AUTO	AUTO	AUTO	AUTO	AUTO	AUTO	AUTO	AUTO	AUTO	14400	AUTO	AUTO

[Texas Instruments MSP430Fxxx]

Bemutató

Ezek az eszközök a JTAG néven ismert programozási módot használják, melyhez az alábbi jelekre van szükség:

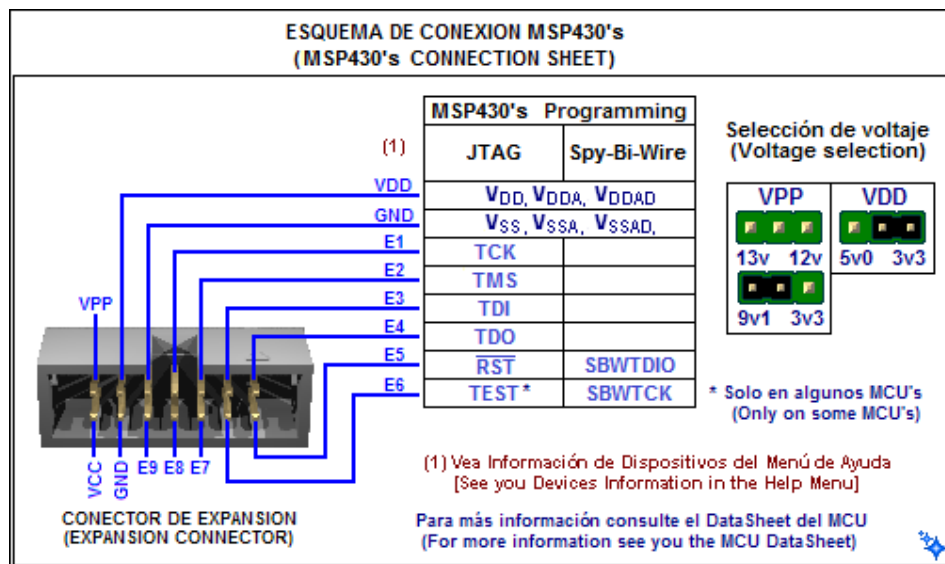
TCK, TMS, TDI, TDO, RST és TEST (hogya a programozás felület megosztásra kerüljön a közös I/O-kkal; VDD (táp) és GND (föld) csatlakoztatással együtt.

Néhány, de nem minden eszköz támogatja a Spy-Bi-Wire (SBW) programozási módot is, ami csak két jelet használ, de sokkal lassabb:

SBWTDIO és SBWTCK; VDD (táp) és GND (föld) csatlakoztatással együtt.

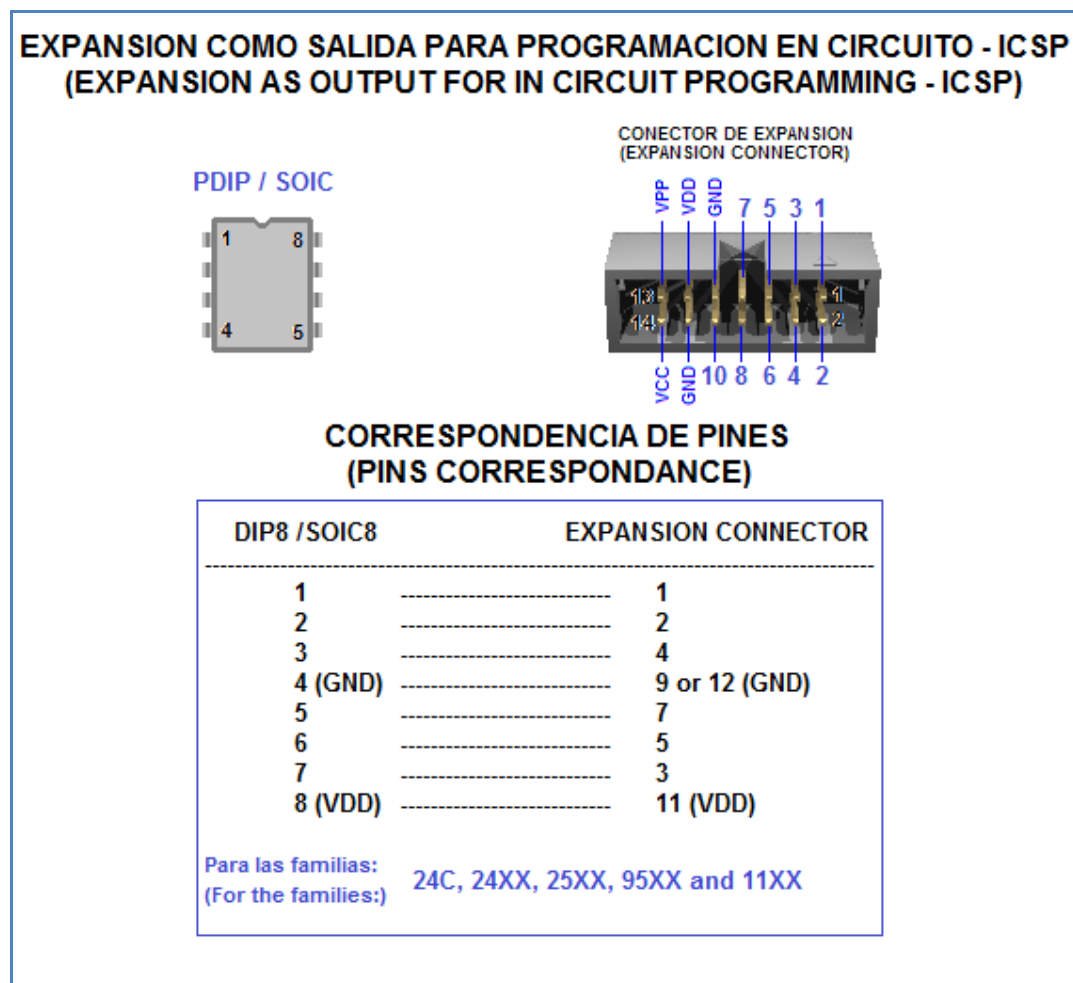
MSP430Fxxx bekötése a programozóra

A programozáshoz a programozó bővítő csatlakozóját kell használni. JTAG vagy Spy-Bi-Wire mód kiválasztható a szoftverben:



[EEPROM áramkörön történő programozása (ISP)]

Hogyan használjuk áramkörön történő (ISP) Eeprom programozásra a kiegészítő csatlakozót
Az alábbi ábra mutatja be a lábak bekötését a programozó és a cél eszköz között 24C, 24XX, 25XX, 95XX és 11XX eeprom családok esetén:



Az alábbi ábra mutatja be a lábak bekötését a programozó és a cél eszköz között 93XX eeprom család esetén:



Sikeres áramkörön történő programozáshoz be kell tartanunk néhány dolgot:

1. lehetőség

- a) A programozó csak kb 20 mA-t tud biztosítani a cél eszköznek. Ha a cél eszköznek nagyobb áramerősségre van szüksége a programozó VDD kimenet értéke leeshet, ami hibás működést fog eredményezni. Automatikus külső tápegység használata esetén hagyjuk a programozó VDD lábát bekötetlenül.
A GND (föld) láb mindig be kell kötve legyen.
- b) A cél alaplapon az eepromot más chipek is elérhetik a programozóval egy időben, így azok egyszerre próbálhatnak kommunikálni vele. Ezt a körülményt el kell kerülni. Ehhez távolítsuk el azokat a kristály oszcillátorokat amik az órajelet biztosítják ezeknek a chipeknek.
- c) Meg kell győződni róla, hogy az eeprom mely lábai vannak használva/ bekötve a cél alaplapon. PI 93XX chip család esetén az ORG láb hova van bekötve (a VDD-re vagy a GND-re). Ebben az esetben csatlakoztassuk le a programozóról és a szoftver menüjében az ORG-re vonatkozóan a megfelelő rublikát kell kijelölni. (Ha az ORG láb be a VDD-re van kötve, válaszd az "ORG 16 bit" pontot, ha a GND-re (föld) akkor az "ORG 8 bit" pontot).

95XX család esetén a Write protect (WP) és Hold (HOLD) lábakat nem szabad a programozóra csatlakoztatni ha azok a cél alaplapon a VDD-re vannak kötve. Ha a GND-re (föld) vannak kötve, az eeprom nem érhető el (hardveres írás védelem). Ha ezek lecsatlakoztatásra kerülnek a cél alaplapról akkor ezeket a lábakat be kell kötni a programozóra.

- d) A legideálisabb esetben minden lábat le kellene választani a cél alaplapról a könnyebb programozhatóság érdekében, de a legtöbb esetben ez nem életszerű.

2. lehetőség

Forraszd ki a az eeprom VDD lábát a cél alaplapról. Ezután egy so8 teszt csipesz segítségével csatlakoztathatod azt a programozóra. Ebben az esetben a programozó biztosítja a chip tápellátását.

3. lehetőség

A legrosszabb esetben leforrasztható az egész chip az alaplapról; Kiolvasás, programozás után vissza kell forrasztani.

Fontos megjegyzések:

Ha a cél alaplapon nem felel meg a fenti feltételeknek, programozáskor előfordulhatnak problémák, ezért fejlesztéskor mindig szem előtt kell ezeket tartani ha szükséges az ISP programozás is.

Eeprom írása előtt érdemes többször kiolvasni annak tartalmát, hogy mindig ugyanaz az adat került-e kiolvasásra.. Ha ez rendben van, akkor valószínűsíthető, hogy a csatlakoztatás megfelelő.

Először mentsd el a kiolvasott adatokat és csak utána nyisd meg a felprogramozandó adat fájlt. Végül a felprogramozott adat és a tároló tartalmát érdemes ellenőriztetni / összehasonlíttatni, hogy az eeprom programozása sikeresen megtörtént-e.