

## Despre programarea microcontrollerului Atmel AVR

Acest document va explica cum functioneaza programatoarele ale caror scheme se gasesc pe internet, pentru a permite intelegerea schemelor si alegerea in cunostinta de cauza.

Programarea ISP (“in-system programming”) inseamna posibilitatea de a programa procesorul fara a-l demonta din circuit; exista si varianta programarii separate, prin scoaterea din soclu si montarea in soclul unui programator dedicat.

Programarea ISP presupune existenta a 2 componente:

- 1) circuitul de programare (programatorul), care se conecteaza la mufa ISP a placutei si la portul serial/paralel/USB al PC-ului
- 2) soft-ul de programare, care preia fisierul utilizator si il transmite programatorului. Trebuie sa fie compatibil cu acesta din urma.

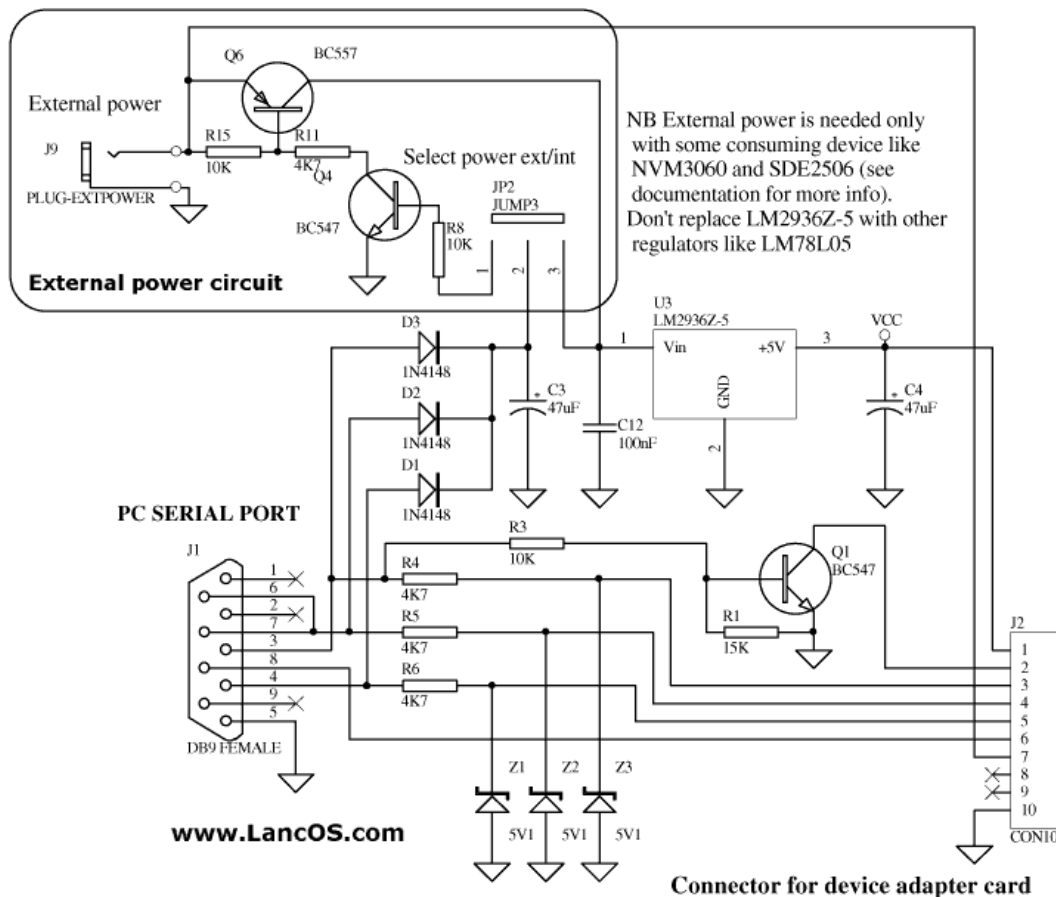
Exista mai multe variante:

a) programarea direct din CodeVision; in acest caz, trebuie folosit un programator suportat de CV, configurat din meniul Settings->Programmer; dintre acestea, posibilitatile cele mai intilnite sint:

- STK500/AVRISP; acesta este un programator “industrial” care contine el insusi un microcontroller care implementeaza toata logica de programare; se poate construi (exista scheme pe internet) dar este mai scump; se conecteaza la PC pe portul serial (sau eventual USB). Se poate procura din comert, de exemplu de la Comet Electronic.
- STK200/300; cu acesta este compatibil un programator simplu care se conecteaza pe portul paralel; programatorul nu contine “inteligenta” deci toate semnalele, cu timingurile lor precise, sint generate de catre softul de programare.

b) programarea folosind un soft extern fata de CodeVision. Un exemplu este Pony Prog ([www.lancos.com](http://www.lancos.com)). Suporta atat programatorul paralel, cit si unul serial.

Schema programatorului serial asa cum apare pe [www.lancos.com](http://www.lancos.com):



### Explicatii:

Aceasta schema este mai complicata decit ne trebuie la proiect, intrucit placuta noastra are deja alimentare – nu este nevoie de aplicarea unei alimentari externe pe pinul Vcc al portului ISP! Ori, o mare parte din piesele de pe schema exact asta fac – extrag alimentarea din portul serial pentru a o aplica procesorului prin ISP! este vorba de diodele D1-D3, C3, C12, C4, stabilizatorul U3, si piesele din dreptunghiul din stnga sus.

Cele 3 diode sint conectate pe pini nefolositi ai portului serial; ele servesc la preluarea “parazita” a alimentarii direct din portul serial, care nu e conceput sa alimenteze periferice, asa cum e cazul la USB: acesti pini, care sint cei de flow control, sint programati de catre soft pentru a avea nivel de +12V pe ei; aceasta tensiune pozitiva trece prin diode, incarca condensatorul C3, si apoi este stabilizata la valoarea de 5V de catre stabilizatorul U3, similar cu 7805, dar cu o cadere de tensiune mai mica.

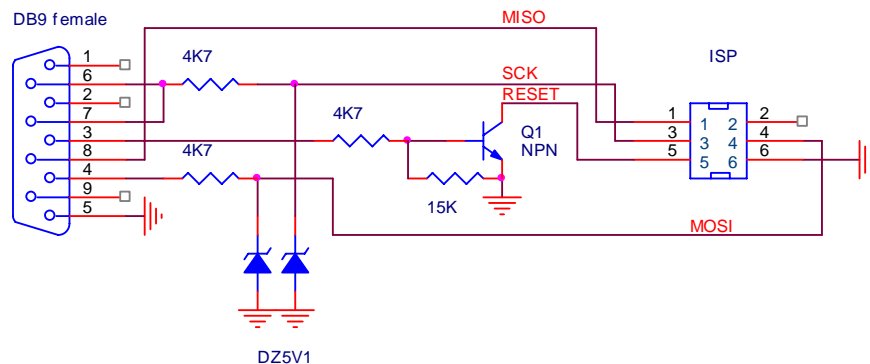
Semnificatia pinilor de la J2, care se conecteaza la mufa ISP de pe placuta noastra (*Atentie!* noi folosim alt pinout de conector), este: 1=Vcc, 2=Reset, 4=SCK, 5=MOSI, 6=MISO, 10=GND. Se observa ca pinul 3 nu este conectat la nimic deci R4 si dioda Zener Z3 vor lipsi.

### Functionare:

Tranzistorul Q1, cind primeste tensiuni pozitive pe baza, conduce si “trage” in 0 pinul reset al procesorului; este necesara aceasta metoda pentru a asigura un 0 “ferm”, caci la mufa DB9 ajung fie tensiuni pozitive (8..12V), fie negative (-8..-12), niciodata 0V. Rezistenta R1 tine blocat Q1 atunci cind mufa DB9 nu este conectata.

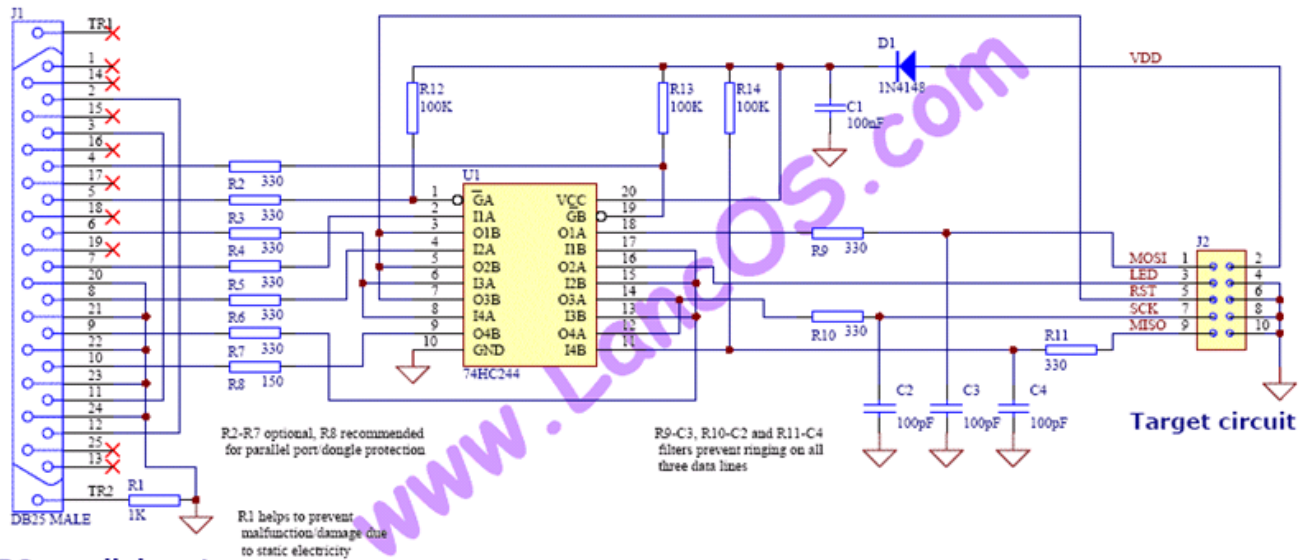
Rezistentele R5, R6 si diodele Zener corespunzatoare au rolul de a transforma tensiunile de +/-12V in cca. 5V/0V astfel: tensiunile pozitive mai mari de 5V sint limitate la 5V de dioda Zener, diferenta de tensiune cade pe rezistenta (a carei valoare deci nu e critica), iar tensiunile negative sint suntate de diodele Zener, care devin polarizate direct si se deschid, lasind maxim -0.6V pe ele (in conductie directa, diodele Zener se comporta ca niste diode obisnuite). Astfel se obtine un “0” dar mai putin “ferm” decit in varianta cu tranzistor.

Prin urmare, o varianta simplificata de schema, cu mufa ISP in varianta folosita de noi pe placa de test, este:



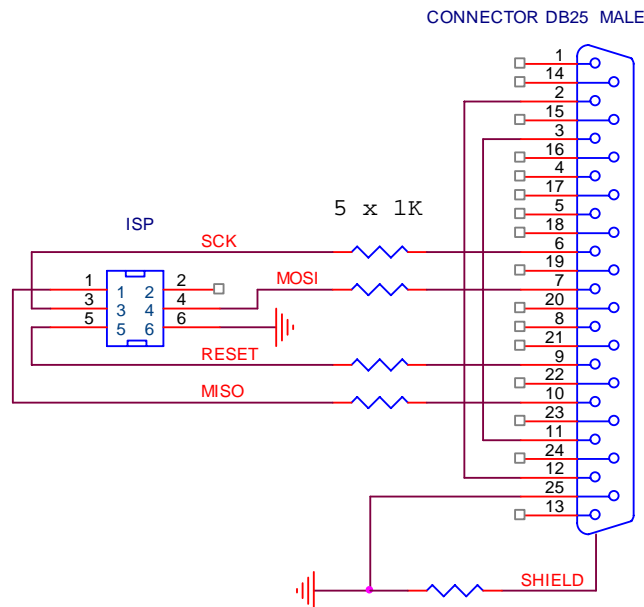
Valorile rezistentelor nu sint critice. Q1 poate fi orice tranzistor NPN de uz general: BC107-109, BC170-172, 2N2222, etc. Diodele Zener pot fi in gama 4V..5V1, de mica putere.

## Schema programatorului paralel



Componenta principala este U1 (74HC244), care este un buffer, un circuit cu rolul de separare si amplificare (semnalul logic TTL de la intrarea In este amplificat in curent si apare la iesirea On). Acest buffer si rezistentele R2-R8 au rolul de a separa si a proteja atat portul paralel, cit si microcontrollerul; semnalele GA, GB au rolul de "enable" pentru grupurile A si B de pini. Este uzuala folosirea unui buffer de acest tip de fiecare data cind se conecteaza un periferic cu interfata de tip paralel pe un bus de date de lungime mare, sau in general cind cele 2 circuite care se conecteaza sînt in aparate diferite, sau atunci cind trebuie conectati mai multi receptori la un singur emitor (si deci cererea de curent e mai mare decit ceea ce poate asigura emitorul).

Dezavantajele folosirii acestui buffer sînt complexitatea mai mare a schemei si necesitatea alimentarii acestuia (realizata de D1 si C1). Prin renuntarea la el si la filtrele RC formate de R10,C2 etc, se obtine schema simplificata:



Observatii despre folosirea schemelor simplificate:

- Utilizati aceste scheme pe riscul vostru. Nu pot fi facut raspunzator de eventualele pagube ce pot apare!

- In cazul schemei seriale, simplificarea nu a eliminat nici un element de siguranta, ci doar sursa de alimentare. In cazul schemei paralele, am eliminat *exact* elementele de siguranta in scopul simplificarii! Ati fost preveniti.
- Schema seriala are mai multe piese, deci la prima vedere este preferabila cea paralela (plus ca aceasta este suportata direct de CVAvr ca varianta de programator STK200/300). Totusi, trebuie retinut ca portul serial al PC-ului este mai "solid" decit cel paralel, deci este mai putin riscanta folosirea variantei seriale (mai ales cind la cea paralela se omite bufferul).
- Nu conectati intre ele placuta si PC-ul cind acestea sint alimentate! Daca PC-ul este pornit, se conecteaza intii mufa DB25 a programatorului la acesta, apoi mufa ISP la placuta *ne-alimentata*, si abia apoi se alimenteaza placuta! Aceasta indicatie este valabila si in cazul conectarii unui alt periferic la PC (mai ales imprimantele pe portul paralel).
- Pentru a reduce riscul, este indicat ca pinii de masa ai celor 2 montaje sa fie conectati inaintea conectarii altor semnale. In acest scop, se poate eventual conecta un fir suplimentar care sa lege cele 2 mase inaintea conectarii mufei ISP (mai ales in cazul schemei paralele).
- Alimentatorul placutei si PC-ul vor fi conectate in aceeasi priza de 220V
- Folositi cablu de lungime minima posibila (mai ales in cazul portului paralel).
- Cei care nu au porturi seriale/paralele native si doresc folosirea de convertoare USB/serial sau USB/paralel pot avea surpriza ca unele din aceste convertoare nu vor functiona. Mai ales in cazul convertorului paralel, acesta e gindit pentru imprimante, nu pentru programatoare, si e posibil sa strice "timing-urile".

Alte observatii:

- Se observa ca nu am conectat pinul 2 al mufei ISP (Vcc); placuta se alimenteaza din propriul alimentator. Pinul 2 este prevazut pentru acele programatoare care necesita alimentare, cum ar fi cel paralel cu buffer, sau AVRISP.
- In general, daca folositi un soft de programare serial extern (deci nu programatorul din Code Vision), poate apare conflict la portul serial. De aceea, fie inchideti terminalul din CV, fie macar apasati butonul "Disconnect" al acestuia, inainte de a incepe programarea.