

Cuprins

Fișa de Asamblare	
1. Funcționare	2
2. Schema	3
4. Lista de componente	3
5. PCB	4
6. Introducere în μ Controllere	5

PLATFORMĂ DE EVALUARE 1 μ CONTROLLERE

- Avantaj Pret/Calitate
- Livrare rapida
- Design Industrial
- Proiecte Modificabile
- Adaptabile cu alte module
- Module usor de asamblat
- Idei Interesante

Idei pentru afaceri

Hobby & Proiecte Educationale

www.epsicom.com/kits.php

a division of EPSICO Manufacturing

Construcție On Board, numit și Single-Board computer pentru mult prea cunoscutul microcontroller PIC16F84, este poate cea mai comună placă cu utilizare directă, uneori finală, la aplicațiile noastre.

Board de evaluare a arhitecturilor realizate cu PIC-uri PIC16F877 sau PIC16F876, însă la fel de bine puteți face teste cu cele din seria:

PIC16C64(A), 16C65(B), 16C662, 16C67, 16C74(AB), 16C765, 16C77, 16C774, 16F74, 16F747, 16F77, 16F777, 16F871, 16F874, 16F874A, 16F877, 16F877A, 16F914, 16F917, 18C442, 18C452, 18F4220, 18F4320, 18F4331, 18F4410, 18F442, 18F4420, 18F4431, 18F4439, 18F4455, 18F448, 18F4480, 18F4510, 18F4515, 18F452, 18F4520, 18F4525, 18F4539, 18F4550, 18F458, 18F4580, 18F4585, 18F4610, 18F4620, 18F4680

Compatibile cu 40 de pini

Funcționare

O clasică platformă de evaluare a arhitecturilor realizate cu μ Controller cu 28 și 40 pini din seria celor fabricate de Microchip însa și a celor de la ATMEL. Aceasta conține: o sursă stabilizată de 5V, oscilatorul, circuitul de reset, circuite adiționale cu led-uri, comutatoare, un potențiomtru, un conector servo și o interfață RS232. Conține și o arie de lucru auxiliară și pentru alte aplicații decât cele prevăzute de noi. Toți pinii porturilor I/O ai μ Controllerului sunt legați la un conector tip header și permit cuplarea board-ului la module externe.

Programele pot fi scrise în assembler, C și PicBasic sau PicBasic Pro. Acesta include și conectorul de programare în-circuit astfel încât controllerul poate fi programat chiar pe acest board, folosind un programator.

Se recomandă startarea circuitului pentru familiarizare cu un program simplu de tip (ex. BLINKX2.BAS), cu programe de pe adresele:

1. <http://www.microengineeringlabs.com>
2. <http://www.techsystemembedded.com>
3. <http://www.microchip.com>
4. <http://melabs.com>

Boardul conține:

Reset –ul pentru inițializare.

Potențiometrul ce este conectat pe portul A0 , intrarea analogică pentru procesoare tip 16F876 și 16F877, însă acesta poate fi decuplat sau cuplat dupa caz prin jumperi.

și

PIC16C62(AB), 16C63(A), 16C642, 16C66, 16C72(A), 16C73(AB), 16C745, 16C76, 16C773, 16F72, 16F73, 16F737, 16F76, 16F767, 16F870, 16F872, 16F873, 16F873A, 16F876, 16F876A, 18C242, 18C252, 18F2220, 18F2320, 18F2331, 18F2410, 18F242, 18F2420, 18F2431, 18F2439, 18F2455, 18F248, 18F2480, 18F2510, 18F2515, 18F252, 18F2520, 18F2525, 18F2539, 18F2550, 18F258, 18F2580, 18F2585, 18F2610, 18F2620, 18F2680

Compatibile cu 28 de pini

Se recomandă testarea cu programele ADCX2.BAS, ADC8X2.BAS și ADC10X2.BAS.

ICSP – programare în circuit folosind un programator cuplat prin conectorul cu 10 pini.

Led-urile conectate pe port-urile B0,B1,B2. Se folosesc programele BLINKX2.BAS și BLINK3X2.BAS

Push butoanele conectate la porturile B4-B6 (pot fi testate cu BUTX2.BAS)

RC Servo, conector la care se cuplează un servo de ieșirea portului B3 ce trimite impulsuri de 1-2ms cu frecvența de 50-60Hz. Se testează cu programul SERVOX2.BAS

RS232 permite comunicarea asincronă cu alte dispozitive. Linia seriala TX (pin 2 pe J4) este conectată la portul C6 prin MAX232. Linia seriala RX (pin 3 pe J4) este conectată la portul C7 prin MAX232.

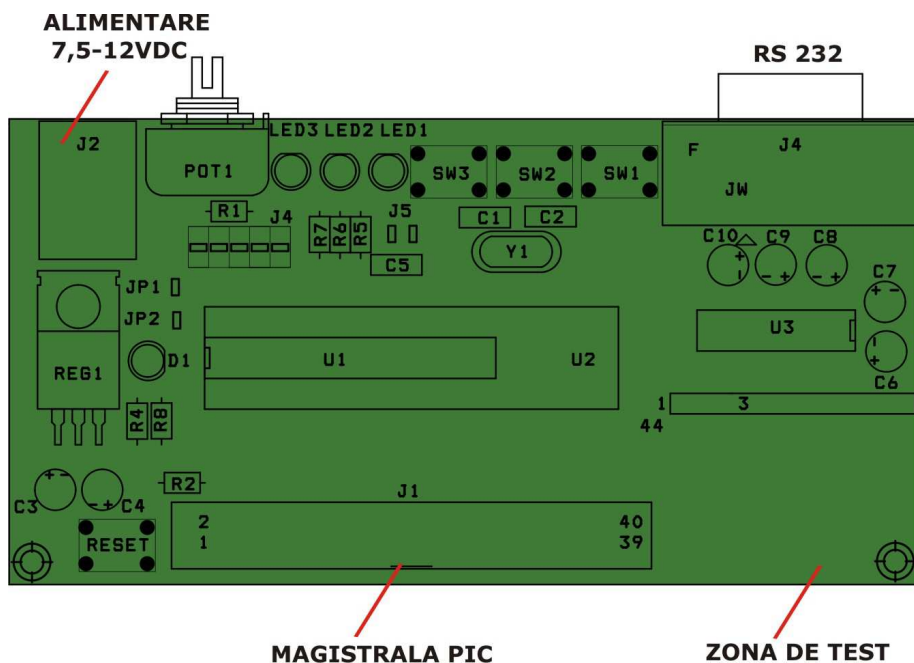
Se testează cu HSERX2.BAS

Programe

Sunt necesare câteva programe pentru inițiere și utilizare acestui sistem (programe de test pe care le găsim rapid pe internet). Astfel MPLAB.EXE (editor, assembler și simulator) și MPASM.EXE (assembler) și PSIM.EXE (simulator) sunt downloadabile free de pe internet, **NTPicprog**, de la adresa <http://tmw.dreamhosters.com/ntpicprog.zip> , **ICPROG.EXE** de la www.ic-prog.com e super.

Lista de componente

Nr.Crt.	Componenta	Denumire	Valoare	Cant
1	C1,C2	Condensator NP	22pF	2
2	C3	Capacitor POL	10µF/16V	1
3	C4,C6,C7,C8C9,C10	Capacitor POL	1µF/16V	6
4	C5	Condensator NP	0,1µF	1
5	D1,Led1,Led2,Led3	Led		4
6	J1	Conector		1
7	J2	Conector		1
8	J4	Conector		1
9	J5	Conector		1
10	JP1,JP2	Conector		2
11	P1	Semireglabil	5KΩ	1
12	R1	Rezistență	1ΩF	1
13	R2	Rezistență	100Ω	1
14	R4,R5,R6,R7	Rezistență	470Ω	4
15	REG1	C.I.	LM7805	1
16	RESET,SW1,SW2,SW3	Pushbutton		4
17	U1	C.I.	µC 40 pini	1
18	U2	C.I.	µC 28 pini	1
19	U3	C.I.	MAX232	1
20	Y1	Cuarț	4.000Mhz	1



Header

PORT A A/D

A0 POT1

PORT B - LEDs, RC Servo, Push buton

B0 LED1

B1 LED2

B2 LED3

B3 RC Servo, J5

B4 SW1 (enable internal pull-ups)

B5 SW2 (enable internal pull-ups)

B6 SW3 (enable internal pull-ups)

/ ICSP SCLOCK

B7 ICSP SDATA

PORT C - Serial

C6 RS232 TX

C7 RS232 RX

Amplasarea componentelor

Acest produs se livrează în varianta circuit imprimat, circuit imprimat + componente sau în varianta asamblată în scopuri educaționale și va fi însoțit de documentația completă de asamblare pe CD.

Dacă doriți să aflați mai multe despre produsele noastre, vizitați situl www.epsicom.com

Dacă ați întâmpinat probleme cu oricare dintre produsele noastre sau dacă doriți informații suplimentare, contactați-ne prin e-mail office@epsicom.com

Pentru orice întrebări, comentarii sau propuneri de afaceri nu ezitați să ne contactați pe adresa office@epsicom.com

31 Sararilor Street | 200570 Craiova, Dolj, Romania | 0723.377.426, 0743.377.426

Câteva cuvinte despre structura microcontrollerelor

Cu ajutorul acestor prime informații veți intra în superba lume a microcontrollerelor.

În primul rând să știm cu ce lucrăm, cum funcționează și cum putem realiza programe pentru ele.

Câteva linii de program reușite și câteva exemple revelatoare vă vor da impulsul la a transforma un chip într-o jucărie teribilă.

Ce ne trebuie pentru început ? Curajul abordării, în fond nu este complicat deloc ! Totul are un început iar apoi vine și performanța.

Schema bloc simplificată

La PIC16F84 sau PIC16F870 vom găsi și vom lucra cu acest grup de patru blocuri.

Memoria program (verde) va conține programul pe care îl înscrinem. Programul (Software-ul) conține un set de date și instrucțiuni pe care microcontrollerul le va prelucra. Ele vor fi scrise într-un program cu ajutorul unui PC, instrucțiuni pe care le înscrinem apoi în controller, adică le programăm cu un programator în "memoria program".

Această memorie este de tip EEPROM care poate fi rescrisă de mii de ori. Cum o înscrinem ? Cu un programator pe care îl veți putea găsi în colecția IT ([EP0050](#) de exemplu pentru o gamă largă de microcontrolere produse de firma MICROCHIP).

Regiștrii și blocul RAM (portocaliu) conține toate registrele interne precum și o mică memorie RAM (64-128 octeți) unde puteți stoca date temporare. Există mai mulți regiștrii cu funcții diferite ce vor fi explicați în continuare.

Pentru ce sunt necesari ? Să luăm un exemplu: dacă faci o buclă program, atunci ai nevoie de o variabilă pentru a schimba valoarea de fiecare dată când bucla se execută și apoi folosim o variabilă definită într-o adresă RAM pentru a păstra valoarea contorului. (Vom reveni mai târziu cu un exemplu edificator, nu este complicat). Conținutul registrului și datele din RAM vor dispărea la oprirea tensiunii de alimentare, sunt doar date provizorii, temporare.

O altă zonă de memorie, care va funcționa la fel ca și RAM-ul, este memoria **EEPROM (galben)** Aceasta este o memorie mică unde puteți citi precum și scrie date, însă datele nu vor dispărea atunci când tensiunea dispăre. Datele din această memorie pot fi adresate, citite, modificate sau descărcate, însă pot fi și protejate la citire cu un cod (lock).

Ultimul bloc este cel cu **porturi**, adică pinii de intrare sau ieșire pe care îi definim prin program ca intrări sau ieșiri cu funcțiile pe care le dorim (să citim anumite date prin pinii-port de intrare și să oferim datele rezultate din program către anumite circuite cum ar fi de comandă unor elemente de forță, ...). În exemplele ce vor urma afla cum și la ce se folosesc.

PIC16F870 este o versiune apărută ulterior procesorului PIC16F84, ambele circuite conțin mai multe blocuri decât cele descrise. Trecând de la prezentarea generală trecem la citirea fișelor tehnice ale celor două circuite pentru a obține detaliile fiecărui bloc, pentru a le utiliza numai dacă avem strict nevoie de ele. Ambele controlere au timere, watchdogs, sistem de întreruperi și multe alte blocuri.

Să luăm un exemplu: circuitul PIC16F870

În figura alăturată sunt notați pinii circuitului.

Pentru o mai bună selecție pe categorii a acestora, s-au colorat.

Astfel:

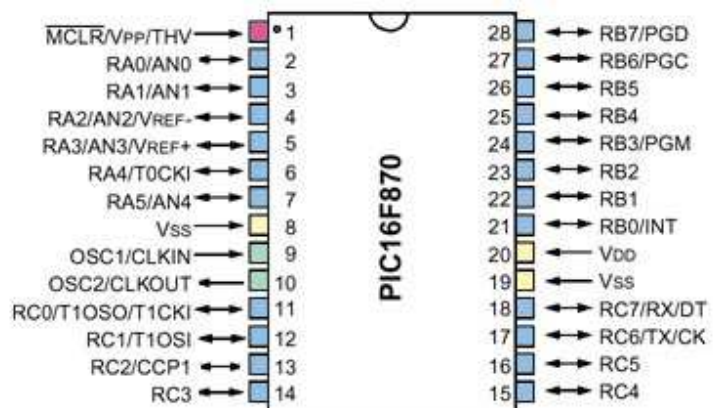
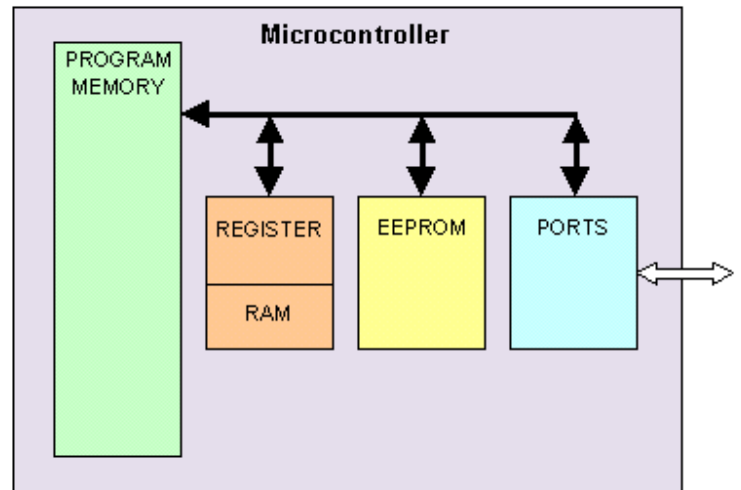
- Pinii desenați cu albastru sunt porturi (pini de intrare/ieșire).

Deoarece nu există multe extra funcții în acest microcontroller, unii pini pot fi folosiți în mai multe scopuri: ca intrări la un convertor AD (analog/numeric) intern iar alții pini pot fi conectați la un contor intern, etc. Prima oară vom afla cum utilizăm pinii albaștrii ca simple intrări și ieșiri.

- Cei doi pini verzi ar trebui să fie conectați la un cuarț pentru a obține un semnal de ceas intern, generatorul de impulsuri de tact.

- Pinii galbeni sunt dedicați alimentării circuitului;

- Pinul roșu este intrarea Reset care va inițializa funcționarea circuitului.



Până aici a fost relativ simplu de înțeles.

Poate părea uneori greu și plictisitor, însă observațiile din exemple ne scapă mai tot timpul de explicarea detaliilor.

Vrem să învățăm să programăm și nu știm cu ce să începem? De ce anume mai avem nevoie pentru a ne pune la treabă ?

Etapa I

- În primul rând avem nevoie de o platformă de evaluare/dezvoltare a unor circuite realizate cu microcontrollere.
- Luăm un exemplu de circuit din manual sau de pe net, configurăm circuitul și testăm programele deja realizate. Exemplele sunt un mod rapid de învățare. Deschidem fișierele ASM, citim comentariile făcute pentru fiecare linie și vom înțelege foarte repede modul în care funcționează microcontrolerul, vom învăța de asemenea un set minim de instrucțiuni. Foile de catalog ale fiecărui procesor conțin detalii referitoare la instrucțiuni.

Etapa II

Iată o carte EBOOK de bază despre PIC16F84.

[Aici găsim o documentație excelentă pentru înțelegerea funcționării și elemente de programare pentru PIC16F84.](#)

Merită citită, este gratuită.

Ce putem face, de exemplu, pentru a aprinde un LED în primă fază și să îl facem apoi să clipească ?

- vom scrie câteva linii de program și le vom salva într-un fișier cu extensia .asm ([le găsim în cartea de mai sus](#)). Pentru o înțelegere rapidă a procedurii și o corectare rapidă, liniile de program din .asm sunt și vor fi însoțite de comentarii.
- îl asamblăm cu un program numit MPASM (vezi mai jos),
- îl compilăm rezultând un fișier .hex în cod mașină și un fișier .lst de verificare (erorile apărute în limbajul de asamblare pe linia de program vor fi regăsite ca mesaje în fișierul .lst). Fișierul .hex va fi înscris în format recunoscut de procesor.
- Verificăm liniile de program și apoi funcționarea aplicației programând controllerul (vezi mai jos).

Etapa III

Ce mai rămâne de făcut ?

Dacă ați ajuns la acest rând înseamnă că ați făcut deja foarte multe. De aici înainte folosiți imaginația pentru a realiza aplicații cât mai interesante. Mult succes cu proiectele voastre și nu uitați că cei mai buni profesioniști au fost autodidacții !

Asamblorul

Există multe asambleoare pentru PIC16F84 și 16F870 însă ce poate fi mai la îndemână decât programul oferit gratuit de producătorul [MICROCHIP](#) , ultima versiune de [MPLAB](#) ? Microchip a realizat un mediu software complet unde puteți simula software-ul vostru înainte de a realiza circuitul. Acest mediu este numit MPLAB. [Development software \(MPLAB® IDE\)](#)

Programatorul

Dacă ați asamblat codul veți obține un fișier hex. pe care va trebui să îl înscriseți în microcontroller, adică să programați microcontrollerul.

Pentru a face acest lucru avem nevoie de o interfață (Programator PIC) de la computer la microcontroler și de un program software cu care să transferăm fișierul din PC în microcontroller.

Iată un programator util pentru o gamă largă de microcontrollere din seria MICROCHIP, cod **EP0050** realizat și de firma noastră. După instalarea programului și cuplarea programatorului pe portul serial, se lansează programul, se selectează controllerul PIC16F870 și se setează opțiunile de programare (configurare):

Fuserele:

- Watchdogtimer (pagina 101 datasheet)
- Poweruptimer (pagina 89 datasheet)
- Brown out resetare permite (pagina 94 datasheet)
- Low Voltage Programming Enable (Activare tensiune mica de programare)
- Code Protection Data Enable (Activare Codul de protecție a datelor)
- Flash Program Memory Write Enable
- Brown out reset enable (pagina 103 datasheets)

Fișierele necesare de la Microcip: [PIC16F870 Datasheet](#) , [Programarea specifică pentru PIC16F87X](#)