

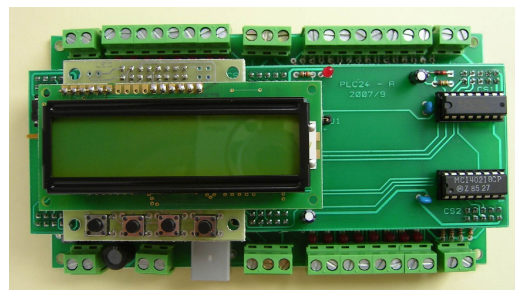
Digitális, analóg bemenetek, hőszenzorok olvasása, kimenetek vezérlése,
a PICPLC24 áramkörrel, és IBM PC-n futó kezelő programmal

A következők azoknak a "programozóknak" szólnak, akik egy olyan, saját, PC alapú vezérlőprogramot szeretnének írni, amivel bemeneteket figyelhetnek, analóg méréseket végezhetnek, és külső eszközöket tudnak vezérelni. (Pl. egy PC központú folyamatvezérléshez, szabályozáshoz kell egy be/kimeneti egység, vagy egy PC-ről vezérelt intelligens ház" programmal világítást, fűtést, redőny leeresztést, felhúzást kapcsolgatni, stb.)

Nyolc kimenet, egy analóg kimenet, és összesen 16 bemenet áll rendelkezésünkre. A bemenetek közül 8 digitális, 8 pedig lehet digitális, vagy analóg, vagy DS1820 hőszenzor bemenet, akár keverve is.

Tehát tudunk:

- 8 feszültség kimenetet vezérelni (A táptól függően 12, vagy 24 voltos)
- beolvasni 8-16 digitális bemenetet
- vagy 1-8 analóg bemenetet
- vagy 1-8 darab hőszenzort (DS1820)
- egy 0-10 voltos analóg kimenetet vezérelni



A hardwer:

A program a PICPLC24 mikrokontrolleres vezérlőpanelre, és 18F452 mikrokontrolleren fut.

A PICPLC24 panel részletes áramköri leírása a PICPLC24.PDF file-ban, most csak pár lényegi dolog:

A B1-8 bemenetek csak mint digitális bemenet használható.

A B9-B16 bemenet a mikrokontroller AN0-7 portjára csatlakozik. A mikrokontroller ezeket a kivezetéseit - portjait - tudja digitális, vagy analóg bemenetként, de akár kimenetként is kezelni. (Ez utóbbi a DS1820 szenzor beolvasásánál szükséges is.)

A B9-16 sorkapocsba tehát köthetünk digitális jeladókat, vagy analóg jeleket, vagy a DALLAS cég DS1820 egy vezetéken kommunikáló digitális hőszenzorát.

De honnan fogja a program tudni, hogy a B9-16 bemenetet mint analóg, vagy digitális bemenetként kezelje, vagy épp egy DS1820 szenzort próbáljon beolvasni? A PC-ről érkező parancs "Ax" vagy "DI", vagy épp "?x" karaktere határozza meg, hogy az adott bemenetet a mikrokontroller hogyan kezelje. (A bemenet beolvasó parancsokról majd később.)

Kimenő jelként 8 darab feszültség kimenet áll rendelkezésünkre.

És van egy 0-10 voltos analóg kimenetünk is.

A PLC24-PC program:

A program PIC 18F452 mikrokontrollerre készült, és 4 megás kvarcot igényel. A PIC-be égetett tartalom, a PLC24-PC.HEX a „mikroklub” CD-n is megtalálható.

A program fogadja a PC soros portjáról küldött vezérlőjeleket, és azok tartalmától függően kapcsolgatja a kimeneteket, beolvassa, és beküldi a bemenetek állapotát.

A program futását az L1 led villogása jelzi .

A kimenetek kapcsolgatása:

A PICPLC24 mikrokontrollerébe égetett program három 8 bites adatot vár a soros portján.

Két azonosító bájtot, amiből kiderül hogy most a kimenetek vezérlése következik, és persze a nyolc kimenet állapotát meghatározó adatot. (Amit a vétel után kirak a kimeneteket vezérlő 74HC4094-be.)

Pl. ha a 34 hexadecimális kódot küldjük - binárisan leírva ez 00110100 - akkor a 3., az 5., és a 6. relé lesz bekapcsolva.

A három bájtot 9600 baudon küldjük, 8 adat bit, 1 stop bit, paritás vizsgálat nincs.

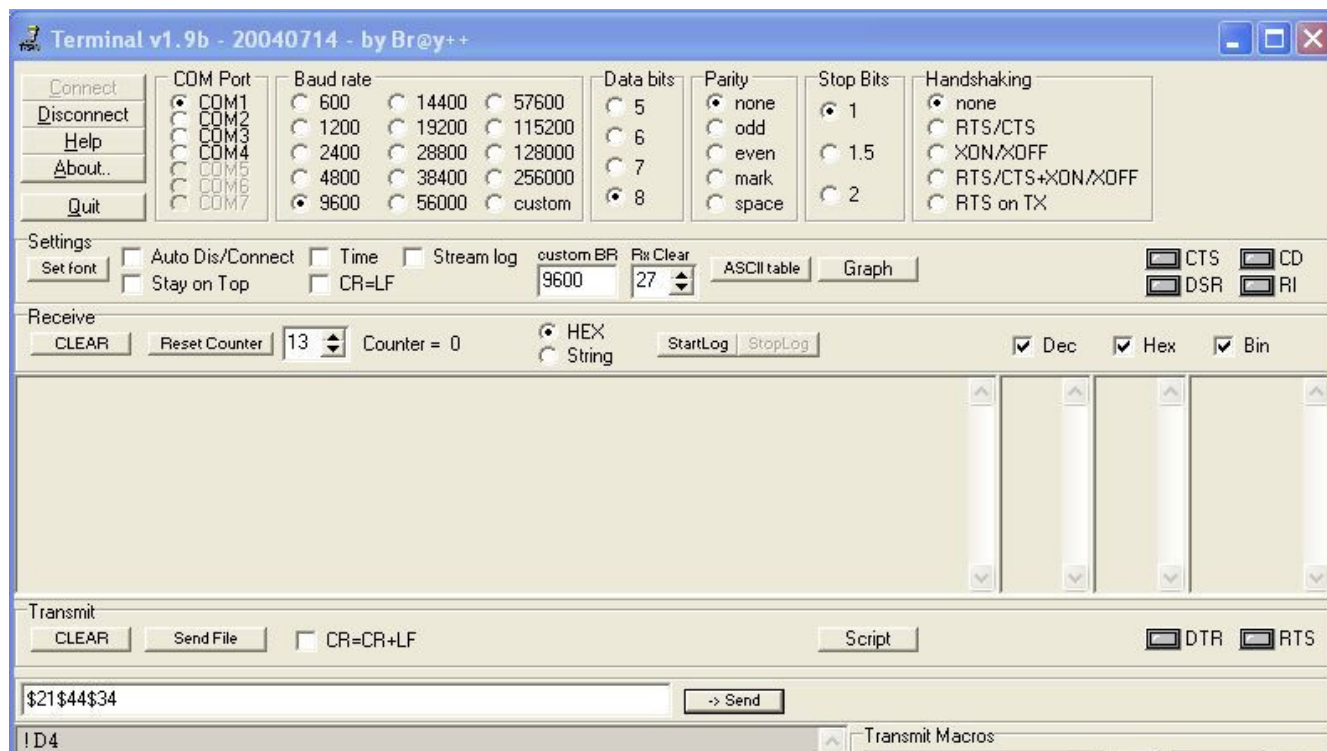
A lényeg, hogy 3 byte-ot kell küldeni, az alábbiak szerint:

1. bájt: 21 (hexa) ami egyébként a "!" kódja, ez egy azonosító byte
2. bájt: 44 (hexa) ami egyébként a "D" kódja, ez egy azonosító byte
3. bájt: a relék vezérlő adata (byte-ja)

Próbaképpen kapcsoljuk be a 3., az 5., a 6. kimenetet. Az előbbiek alapján ez a 34 hexadecimális vezérlőparancs értékeket jelenti. (Mert ugye ez legyen a kimenetek állapota: 00110100 , ez pedig 34h) Ezt kiegészítve az indító kóddal, tehát most az 21 , 44 , 34 hexadecimális kódokat kell kiküldeni.

Hogy néz ki ez a gyakorlatban? A működés egy terminál programmal lepróbálható. (Egy nagyon jó terminál program - ami az alábbi példákban is szerepel - rajt van a „mikroklub” CD-n, - shareware\terminal\terminal.exe - illetve letölthető a <http://bray.velenje.cx/avr/terminal/> címről.)

A teszthez állítsuk be a soros portot, és az előbb leírt adatátviteli paramétereket - 9600 baud, 8 adatbit, 1 stop bit, paritás vizsgálat nincs - majd egy kattintás a „Connect” gombra, és már küldhetjük is a vezérlő adatokat a mikroszámítógépbe!



Tehát a terminálprogramban gépeljük be a \$21\$44\$34 adatokat - a \$ jelzi hogy hexadecimális adatokról van szó - vagy az ezzel egyenértékű !D4 ASCII kódsorozatot. Eztán kattintsunk a Send gombra:



Ha minden rendben, bekapcsolnak a kiválasztott kimenetek.

Ha nem, ellenőrizzük le a beállításokat. (Soros port, baud, stb.) ellenőrizzük le a soros kábel csatlakozását, valamint hogy a megfelelő portot használjuk-e.

Ugyanígy, az 21h, 44h, 55h értéket kiküldve minden második - 1. 3. 5. 7. - a 21h, 44h, FFh pedig az összes kimenetet bekapcsolja.

A B1-8 és B9-16 bemenet mint digitális bemenetek bekérése:

Ha a "DI" parancs megy ki a PC-ről, akkor mint digitális bemenet van értelmezve a B1-8 , de a B9-16 is. (Ahogy arról szó volt, a B9-16 lehet analóg bemenet, és lehet a DS1820 hőszensorok érzékelője is.)

A "DI" parancsra válaszul 3 bájtot kapunk vissza. Az első bájt a B1-8, a második a B9-16 bemenet állapota. Mivel most ezek digitális bemenetként vannak értelmezve, 0 vagy 1-es értékű bit jelenti az adott bemenet állapotát.

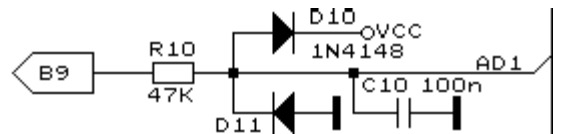
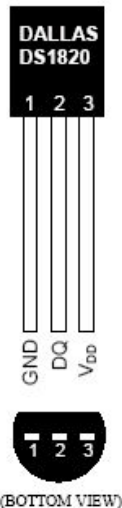
A harmadik bájt pedig egy ellenőrző bájt, ami az első két bájt összege.

A DS1820 hőszenzor beolvasása:

A DS1820 mindössze 3 kivezetéssel - ebből kettő a tápfeszültség - egy komplett hőmérő. A IC-be mindent beleintegráltak, - hőérzékelő, referenciafeszültség generátor, 9 bites A/D, stb. - ami egy 0.5 tized fokos pontosságú hőmérsékletméréshez szükséges. Az IC -55-től +125 C fokig tud mérni, és kifejezetten a mikrokontrolleres alkalmazásokhoz fejlesztették ki. (A teljes dokumentáció megtalálható a CD-n a MIKROKLUB/PDF/DALLAS könyvtárban.)

A DS1820 bekötése:

A PICPLC24 mikrogéphez a DS1820-at egy 3 eres kábelvel csatlakoztathatjuk. A negatív tápfeszültség kivezetést - GND - értelemszerűen egy G1 vagy G2, G3.. jelű, a pozitív tápot - Vdd - az 5V sorkapocsba kössük. A DQ adatkimenetet a B9...B16 bemenetre, ahova akarjuk. A kábel hossz akár több tíz méter is lehet. A kábel típus pár méterig lényegtelen, nagyobb távolságnál a sodrott kábelt ajánlják.



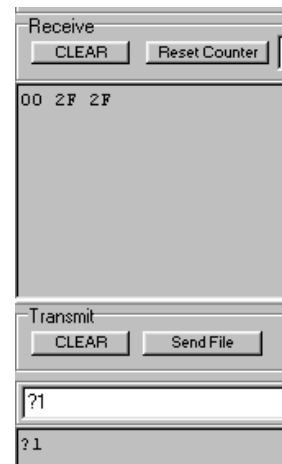
DS1820 beolvasásakor a PICPLC24 bemenetével soros ellenállás értéke 100 ohm - cseréljük ki a 47K ellenállást ! - a mikrokontroller panelon az R4 felhúzó ellenállás 4.7 Kohm legyen, a 100 nF kondenzátort pedig ki kell venni.

Lekérdezéskor "?" karaktert, majd a bemenet számát kell kiküldeni. Annyival bonyolódik a dolog, hogy mivel az áramkör B9-16 bemenete kapcsolódik a - maximum - nyolc DS1820 szenzor, így az első szenzor a B9-re megy, és a ?1 paranccsal kérdezhetjük le.

(Pl.: a ?5 kóddal kérdezhető le az ötödik - B13 sorkapocs - bemenetre kötött hőmérő adata.) Erre a mikro válaszként 3 byte-ot küld vissza:

1. bájt: előjel, ha 0 , akkor plusz, ha 1 akkor mínusz fok.
2. bájt: a hőmérséklet értéke 0.5 Celsius fokos egységben
3. bájt: egy ellenőrző összeg, az 1. és 2. bájt összege 1 byte-on

Egy példa: Mondjuk hogy a szenzor a B9 bemeneten van. A terminál programmal ?1 kódot küldünk, a mikrogép a 00, 2F, 2F hexadecimális értékeket küldi vissza. Az első bájt nulla, tehát a fagyypont felett vagyunk, a második érték 2Fh ami ugye decimálisan 47, 47×0.5 fok pedig adja az eredményt, ami most 23.5 Celsius fok.



Ha az előjel 1 (mínusz fok), akkor a 2. bájtban a hőmérséklet kettes komplemente van. (pl. 1.bájt=1, 2.bájt = FFh = 255 = -0.5C) Egy táblázat erről, az IC dokumentációjából:

TEMPERATURE	DIGITAL OUTPUT (Binary)	DIGITAL OUTPUT (Hex)
+85.0°C*	0000 0000 1010 1010	00AAh
+25.0°C	0000 0000 0011 0010	0032h
+0.5°C	0000 0000 0000 0001	0001h
0°C	0000 0000 0000 0000	0000h
-0.5°C	1111 1111 1111 1111	FFFFh
-25.0°C	1111 1111 1100 1110	FFCEh
-55.0°C	1111 1111 1001 0010	FF92h

Ha nincs, vagy nem olvasható a szenzor, akkor a program 2. és 3. bájton FFh értéket küld.

Az analóg kimenet vezérlése:

Ha az analóg kimenet feszültségét akarjuk állítani, akkor egy három bájtos adatsort kell küldenünk. Az `!Ax` a parancs, amiben `!A` jelzi hogy most az analóg kimenet van soron, és az `x` a kimeneti feszültséget állító 1 byte.

Tehát 3byte-ot kell küldeni, az alábbiak szerint:

1. bájt: 21 (hexa) vagy a `!"` karakter,
2. bájt: 41 (hexa) ami egyébként az `"A"` ASCII kódja,
3. bájt: végül a kimenet vezérlő adata (byte-ja)

Próbaképpen kapcsoljunk 5 voltot a kimenetre! Ahogy arról már szó volt - a kapcsolási rajzon szereplő értékeknél - 0-10 volt lehet a kimeneti feszültség. A vezérlő bájt értéke 0-255, vagy 00-FF ha hexában számoljuk. Az 5 volt pont a fele, azaz 80h.

Az előbbiek alapján tehát ez a 21h, 41h, 80h hexadecimális vezérlőparancs értékeket jelenti. Ezt így tudjuk kiküldeni a kedvenc terminálprogramunkban:



Vagy - keverve az ASCII és a hexa kódokat - így:



Ha minden rendben, a kimeneten az 5 voltot fogjuk mérni.

Az LCD/nyomógomb panel lekezelése:

Ha csatlakoztatjuk az LCD/nyomógomb panelt, akkor azt is tudjuk a PC-ről vezérelni, illetve a gombok nyomását érzékelni.

Ha az

LCDxxxxxxxxxxxxxxxxx

parancsot küldjük ki a soros porton, ahol az x a kiküldendő 16 karakter, az x pedig 1 vagy 2 lehet, attól függően, hogy az LCD első, vagy a második sorába akarjuk kiírni.

Pl. az LCD1****PICPLC24**** parancs az LCD első sorába a ****PICPLC24**** szöveget fogja kiírni.



Amire figyeljünk, hogy pontosan 16 karakter adjunk meg a parancsban, mert csak akkor hajtja végre a mikro. (Ha rövidebb a szövegünk, akkor töltsük fel a "maradék helyet" space karakterekkel.

Ha lenyomunk egy nyomógombot akkor a Kx - ahol az x=1, 2, 3 vagy 4 - kódokat küldi be a mikro, attól függően, hogy melyik gombot nyomtuk le a négy közül.

Kapcsolódó dokumentációk:

Hasonló, ki/bemenet kezelő program készült a PICPLC8 és PICPLC16 áramkörre is. Ezt a RELE16DS.PDF ismerteti.

A PICPLC24, PICPLC16, és a PICPLC8 áramkörökről egy-egy külön leírás szól. (PICPLC8.PDF, PICPLC16, és PICPLC24.PDF)

Az előbbi leírások letölthetőek a lenti honlapcímről, vagy megtalálhatóak a „mikroklub cd”-n.

Végül nincs más hátra mint hogy sok sikert kívánjak a használathoz. 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>