

3. ábra

fér be. Ennyire biztosan nem lesz szükség, még úgy sem, hogy a 16 bit kivitelére kissé pazarló módon elhasználnunk kb. 200-at. Ebben a mikrovezérlőben nincs SPI periféria. Az I/O port kiosztása: $RA0$ = soros óra, $RA1$ = átíró óra, $RA2$ = soros adat, $RA3$ = bemenet (a pozitív multiplexelés miatt a szokásos felhúzó helyett lehúzó ellenállást igényel!), $RA4$ üres kimenet, $RA5$ relékimenet. Az inicializáláshoz tartozik még az időzítések előbányászása a flashból.

A program az inicializálás után két jól elkülönülő részből áll: A megszakítás az időzített feladatokat végzi, a főhurokban pedig a megszakítástól kapott gombadatok feldolgozása történik. A tárgyalást a megszakítással kezdjük, pl. azért is, mert ez gyűjti össze a főprogram számára a feldolgozandó információt.

A megszakítás legelső feladata a kérés bit törlése, hogy a programozónak ne kelljen észben tartania ezt a kötelezettséget. Ezután az a feladat következik, aminek az időzítésén múlik a kijelző egyenletes fényessége: az átíró órajel kiküldése. Az adatot az előző megszakítás írta ki a két 8 bites léptető regiszterbe, most csak átírjuk azok kimenő regiszterébe.

Egy bekapcsolás után csak egyszer fordul elő, hogy korábbról nincs érvényes adat a léptető regiszterekben, és ez az állapot is csak 2 ms-ig tart. Az összes többi feladat időzítése szinte tetszőleges lehet a megszakításon belül, így az egyes ágak időkiegyenlítés nem igényelnek. Az átírás hatására kijelölődik az aktuális szám-

jegy és az aktuális nyomógomb. A számjegy értékét az előző megszakítás már kivitte, most csak az aktuális gombot kell beolvasni.

Egy teljes ciklus (négy gomb) után a főprogramot egy jelzőbit állításával értesíteni kell a gyűjtés megtörténtéről, hogy az elemzést elvégezhesse. Ezután elő kell készíteni és ki kell írni a léptető-regiszter-párba az új számjegyet és annak pozícióját. Ennek első lépése a számjegy szegmensképének kikeresése egy táblázatból. A második lépés a kép elhelyezése „kettőtörve” a felső kimenő regiszter alsó és az alsó kimenő regiszter felső négy bitjén. A táblázat fordított sorrendben tartalmazza az alsó és a felső 4 bitet, ezzel segítve a kettőtörést. Ezt követi a minden megszakításban inkrementált multiplexer számláló utolsó két bitjének táblázatos dekódolása 4-ből 1 alakra, és az eredmény elhelyezése az alsó kimenő szó alsó négy bitjén.

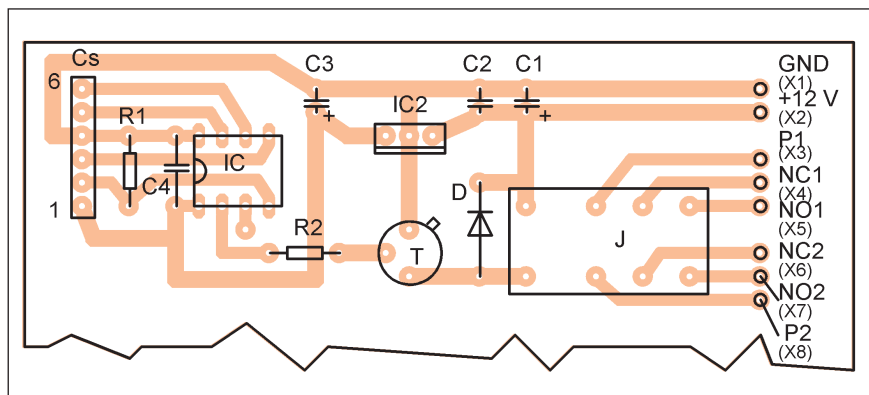
A számjegyek száma 2 hatvány, így a számlálót nem kell törölni, az utolsó két bitje mindig felhasználható. A tizedespont még

igényel némi kezelést, utána a 16 bites adat kivihető. Az utolsó négy bitnek sötét kijelző mellett is pörögnie kell a gombok beolvashatósága érdekében, ezért a sötétítés úgy történik, hogy a táblázatból nyert szegmenskódot felülírjuk nullával. A tizedespont helyét az aktív félperiódus jelzőbitje jelöli ki (=0: jobb mező, =1: bal mező). Villogó ill. stabil jellegét az üzemmód (számlálás / beállítás) határozza meg.

A továbbiakban 2 Hz-nél sűrűbb időzítésre nincs szükség, beiktatunk hát egy 250-es osztót, ami csak minden 250-edik megszakítás számára teszi lehetővé a folytatást, a többi számára itt a megszakítás vége.

A 2 Hz-et használjuk a tizedespont aktív idejének időzítésére (a fény kikapcsolására). A tizedespontot az ezután következő, a félmásodpercek számláló, de a perces osztóláncban részt nem vevő 10-es frekvenciaosztó kapcsolja be 5 másodpercenként. Ez a bekapcsolás a megszakításban a kikapcsolás után következik, így mindaddig érvényben marad, ameddig a következő 0,5 s le nem jár. A 2 Hz-es ágba egy, a főprogramból indítható független 1 perces időzítő is működik. Ez kapcsolja le a kijelzőt.

A szintén a 2 Hz-es ágba elhelyezett perces osztólánc már csak azzal a feltétellel működik, hogy számlálás üzemmód van kijelölve és egyik programozott idő sem nulla. A percenkénti feladat a visszaszámlálásból és a beállított számú perc leteltkor a lejáratkori teendőkből (a másik félperiódus elindításából) áll.



4. ábra